JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

SOLAR TRACKER

[1]Naga Swetha R, [2] B.Pavitra, [3] D.Narendar Singh, [4] Madipeddi Shiva Shankar,
[5]Kodari Ajith, [6] Bolli Sree Vyshnavi, [7]Mannepu Karthika Reddy, [8] Ananthagiri Srikar Prasad
[1,2] Asst.Prof, Dept. of ECE, Anurag University,
[3] Assoc.Prof, Dept. of ECE, Anurag University,
[4,5,6,7,8] B.Tech III year Students, Dept. of ECE, Anurag University

Abstract : A solar panel tracker is used to adjust the solar panel's direction based on the sun's direction. Here, an Arduino Uno board is used to adjust the position of the servo motor gear system by receiving inputs from an LDR sensor and a potentiometer, displaying the output on an LCD display, and making display changes. Fixing the solar panel perpendicular to the sun is the primary goal. At that moment, only the maximum number of sun rays strike the solar panel directly, increasing the quantity of power produced. To do that, we must determine if the temperature at the top and bottom of the solar panel is the same or not. Additionally, we must verify that the solar panel is fixed perpendicular to the sun beams by looking at its left and right sides. Using the numbers from the LDR sensor, I have adjusted the slope in the X direction using the Left-Right gear system and the Y-direction using the Up-Down gear system.

Hardware Components: ARDUINO UNO , LDR SENSOR , LCD DISPLAY , POTENTIOMETER , SERVO MOTOR

Keywords: Solar panel tracking, Adaptive algorithm, Illumination-aware control, Arduino Uno, Energy harvesting efficiency, LDRs, Temperature sensors, PID control, Machine learning, Real-time data, Optimization, Renewable energy, Sustainability, Dynamic orientation, Experimental validation.

I. INTRODUCTION

Harnessing solar energy has become increasingly vital in mitigating environmental concerns and meeting the world's energy needs in a sustainable manner. Because solar panels effectively convert sunshine into power, they are frequently used. But how well they work depends on how they line up with the sun's beams. Solar panels must face directly into the sun in order to maximize energy output; this means that they must be adjusted continuously throughout the day in order to follow the sun's path. This study presents a Solar Panel Tracker System that automatically aligns solar panels for optimal sun exposure in response to this problem. The system integrates sensors, actuators, and control logic onto an Arduino Uno board, which serves as the central control unit. This allows the system to dynamically move the solar panel.

The system's main parts include temperature sensors for alignment confirmation, servo motors for panel positioning, potentiometers for sensitivity adjustment, and light-dependent resistors (LDRs) for sunlight intensity detection. The Arduino Uno board coordinates the movement of the solar panel to guarantee ideal alignment with the location of the sun by analyzing sensor data and producing commands. The goal of this project is to improve the energy harvesting efficiency of solar panels by providing a workable, reasonably priced solution that can be used in a variety of contexts. The Solar Panel Tracker System maximizes energy production by automatically detecting the sun's movement, which advances the development of sustainable energy solutions.

RELATED WORK

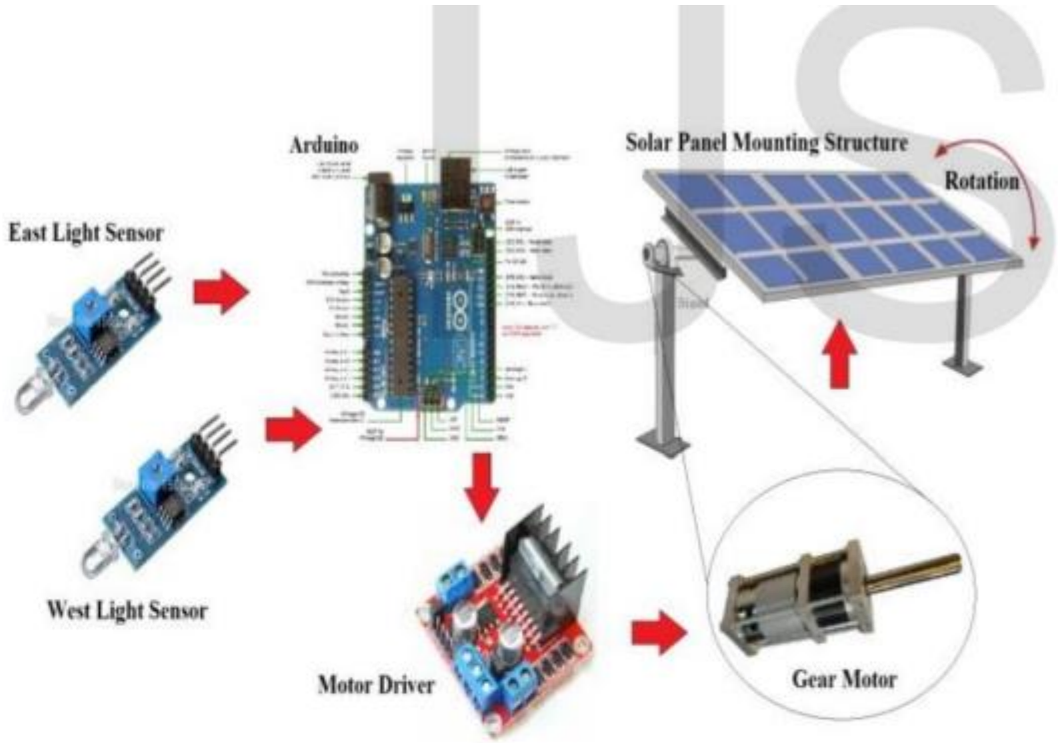
Systems for tracking solar panels are a crucial part of optimizing solar energy gathering. The direction of the solar panels is constantly adjusted by these systems to maximize exposure to sunlight throughout the day, increasing energy output. A review of related work that is especially concerned with solar panel tracking systems that employ Arduino Uno control is presented in this section. The study focuses on important research discoveries, technical advancements, and techniques in this field. An overview of the body of research on solar panel tracking systems is given in the related work section, with a focus on studies that use Arduino Uno control to construct their systems. It goes over several methods for sensor integration, developing algorithms, designing mechanically, evaluating performance, and taking scalability into account. This section seeks to identify possibilities and gaps for additional study as well as to situate the proposed topic within the larger research environment.

II. PROPOSED WORK

THE PROPOSED ILLUMINATION-AWARE ADAPTIVE SOLAR PANEL TRACKING ALGORITHM CONSISTS OF THE FOLLOWING KEY COMPONENTS:

1. **Sensor Data Acquisition:** The algorithm begins by acquiring real-time sensor data, including illumination intensity measurements from LDRs and temperature readings from sensors distributed across the solar panel array. These sensor inputs serve as the foundation for assessing the current illumination conditions and determining the optimal panel orientation.
2. **Illumination Analysis:** The algorithm analyzes the acquired sensor data to assess the current illumination conditions and predict the trajectory of the sun throughout the day. By evaluating illumination intensity trends and temperature differentials across the solar panel array, the algorithm gains insights into the sun's position relative to the panel orientation.
3. **Adaptive Orientation Control:** Based on the illumination analysis, the algorithm dynamically adjusts the orientation of individual solar panels to optimize energy harvesting efficiency. Proportional-integral-derivative (PID) control techniques are employed to calculate precise adjustments in panel orientation, ensuring alignment with the sun's rays for maximum energy capture.
4. **Machine Learning Adaptation:** To further enhance adaptive orientation control, the algorithm incorporates machine learning techniques to adaptively optimize panel orientation over time. By leveraging historical sensor data and past performance metrics, the algorithm continuously refines its control strategy to adapt to changing illumination conditions and environmental factors.

III. BLOCK DIAGRAM

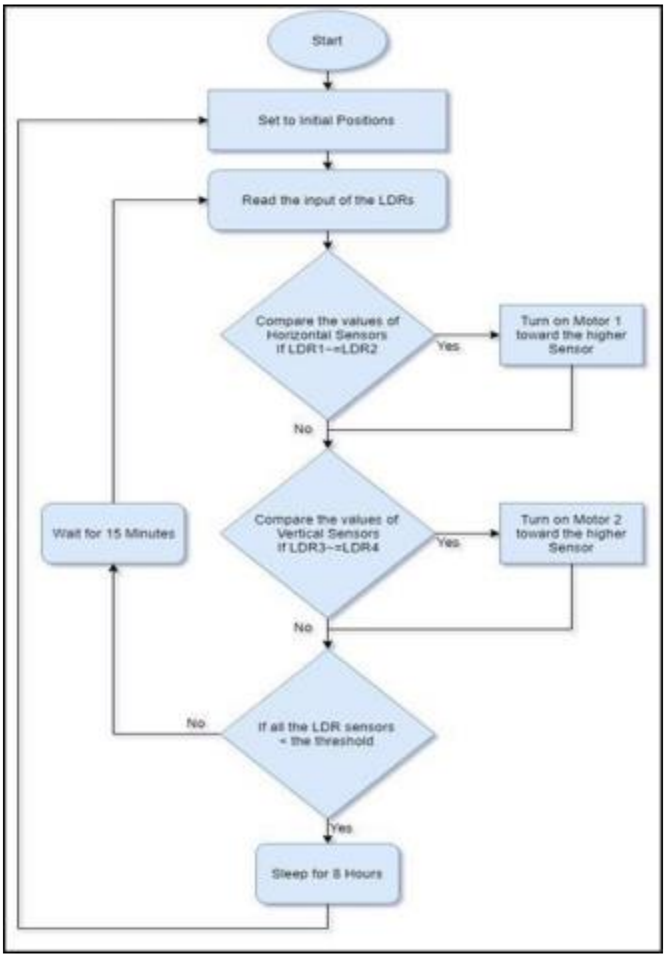


The block diagram illustrates a solar panel tracking system that tracks the sun's movement and modifies the solar panel's position based on three light sensors and an Arduino microcontroller.
The components of the system are as follows:
East Light Sensor (8770): This light sensor gauges how much light enters the room from the east. The number 8770 is used to indicate the sensor's value.
CUES: This light sensor monitors the quantity of light emitted by the sun. 9440 is the number that represents the sensor's value.
The West Light Sensor (II) is a light sensor that gauges the quantity of light entering the atmosphere from the west. II is the sensor's value rendered numerically.
Arduino (413): This microcontroller is used to interpret light sensor input and regulate the solar panel's movement. The precise kind of Arduino board that is utilized in the system is represented by the number 413.
The framework that holds the solar panel in place is called the solar panel mounting structure. The diagram does not specify the material that was utilized to construct the construction.
Stain: This substance is used to shield the solar panel mounting system from environmental factors.
The gear motor's movement is managed by a device called a motor driver. The solar panel mounting structure is moved by means of a gear motor. The motor driver is in charge of the motor's spinning.The Arduino is linked to the light sensors, and it uses the sensor data to compute the sun's position. The motor driver then receives signals from the Arduino and uses them to control the gear motor's movement. In order to optimize the quantity of sunshine that reaches the solar panel, the gear motor rotates the solar panel mounting framework.

The block diagram does not depict how the components are connected to one another. Nonetheless, it is evident that the motor driver and the Arduino are connected to the light sensors. The gear motor, which is attached to the solar panel mounting framework, is connected to the motor driver.

Understanding the general layout and functioning of the solar panel tracking system may be done with the help of the block diagram. The Arduino can calculate the sun's position and modify the solar panel's location by measuring the quantity of light coming from various directions using light sensors. The solar panel may be precisely controlled to move in a way that maximizes the quantity of sunshine it receives thanks to the use of a gear motor and motor driver.

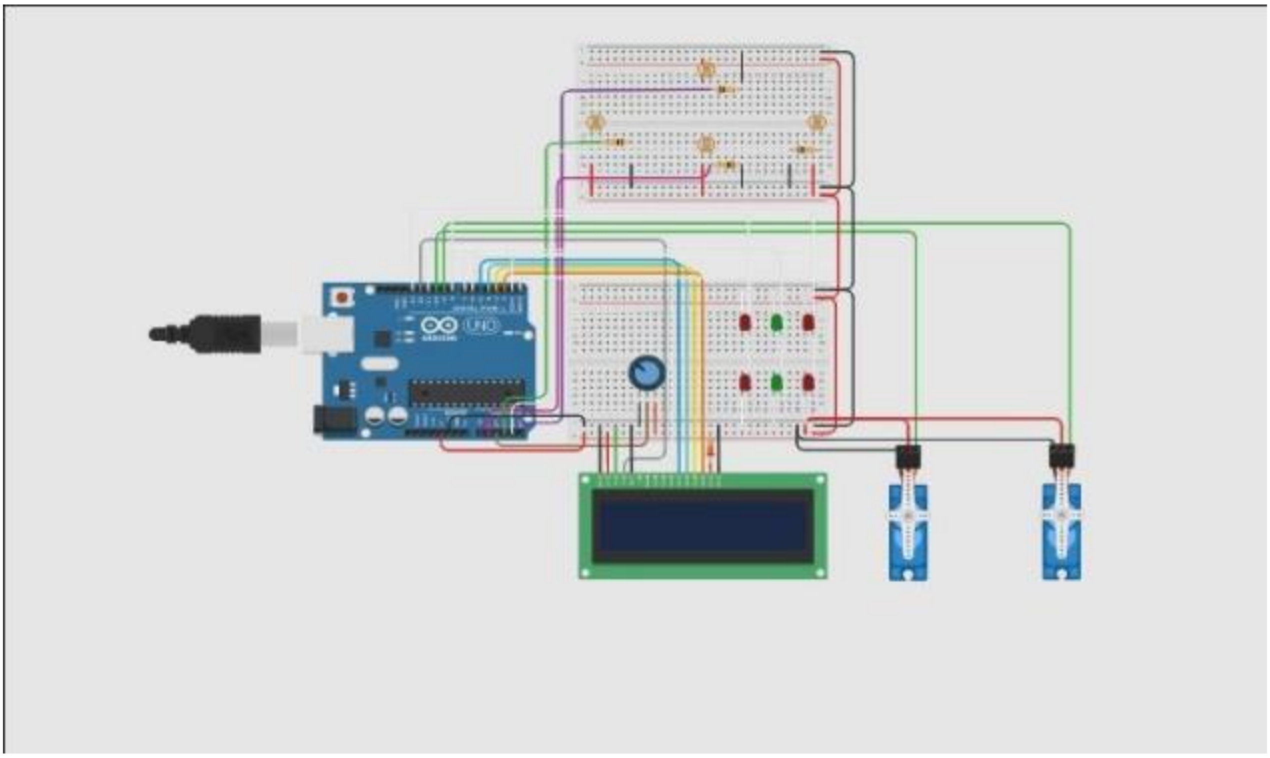
IV. FLOW CHART



The procedures for a solar panel tracking system with LDR sensors and a microcontroller seem to be outlined in this flowchart. The system begins by positioning the solar panel's starting locations before reading the input from the LDR sensors continuously. In order to ascertain if the solar panel is oriented horizontally with respect to the sun, the first step in the loop compares the values of the horizontal sensors (LDR1 and LDR2). The microcontroller will turn on Motor 1 to move the solar panel in the direction of the greater sensor value if LDR1 is smaller than LDR2. The microprocessor will turn on Motor 1 to move the solar panel in the opposite direction if LDR1 is bigger than LDR2.

To ascertain if the solar panel is oriented vertically with respect to the sun, the loop's subsequent step compares the values of the vertical sensors (LDR3 and LDR4). The microcontroller will turn on Motor 2 to move the solar panel in the direction of the greater sensor value if LDR3 is smaller than LDR4. The microprocessor will turn on Motor 2 to move the solar panel in the opposite direction if LDR3 is larger than LDR4. After then, the loop continues every fifteen minutes, interpreting the data from the LDR sensors and modifying the solar panel's location as necessary. The system will recognize that the sun has set and go into an 8-hour sleep mode if all of the LDR sensors register a value below a certain threshold.

V. CIRCUIT DIAGRAM



The source code for an Arduino-based robot that can navigate and avoid obstacles using a range of sensors is supplied. An Arduino microcontroller board is coupled to a number of components that make up the robot's circuit.
The following are the circuit's primary parts:
The robot's main processing unit, the Arduino microcontroller board, runs the software that is kept in its memory. A reusable platform for constructing and testing electrical circuits is called a breadboard.
Servo motors are tiny motors that are used to regulate how the wheels of a robot move.
Ultrasonic sensors: These are sensors that identify obstructions in the robot's route by using sound waves. Infrared sensors: These are sensors that look for items using infrared light.
Potentiometer: The tolerance value in the program may be changed using this variable resistor. This is an LCD display, which indicates the value of the current tolerance.
Using a breadboard, the circuit is linked to the Arduino microcontroller board. Digital pins 9 and 10 are connected to the servomotors, which are used to regulate their movement. Analog pins A0 and A1, which are used to read sensor readings, are linked to the ultrasonic sensors. Digital pins 0 and 1, which are used to detect the presence of objects, are coupled to the infrared sensors. To change the tolerance value, the potentiometer is attached to analog pin A2. Digital pins 2, 3, 4, 5, 6, and 11 are linked to the LCD display and are used to operate it.
The robot's behavior is implemented via the source code that is given. It utilizes the sensor readings it reads to regulate the servo motors' motion. The main loop() function, LeftRight(), UpDown(), and other functions are included in the program.
The LCD display and the servo motors are among the pins and variables that are initialized by the setup() function for usage in the program. The primary function that executes continually is loop(). It computes the difference between every pair of sensors after reading the data from the sensors. Next, it contrasts these variations with a tolerance value computed using the reading from a potentiometer attached to analog input pin A2.
The application will use the UpDown() and LeftRight() routines to control the movement of the servo motors based on the sensor readings and the tolerance value. In order to show the sensor status, the software additionally modifies the state of a number of output pins. The current tolerance value is shown on the LCD display.

Equations

1.Calculating the tolerance value based on the potentiometer:

The tolerance value in the program is calculated based on the value of a potentiometer connected to analog input pin A2. The equation for calculating the tolerance value is:

tolerance = (potentiometer value * 5.0) / 1023.0

where potentiometer value is the value of the potentiometer (ranging from 0 to 1023).

2.Calculating the difference between two sensor values:

The program compares the difference between two sensor values to a tolerance value. The equation for calculating the difference between two sensor values is:

difference = sensor1 - sensor2

where sensor1 and sensor2 are the values of the two sensors.

For example, the program calculates the difference between the sensor values of the top and bottom ultrasonic sensors using the following code:

```
int sensorTest1 = sensorTop - sensorBottom;
```

3.Calculating the difference between the sensor value and the tolerance value:
The program compares the difference between the sensor value and the tolerance value to determine if the robot should move. The equation for calculating the difference between the sensor value and the tolerance value is:

difference = sensor value - tolerance value

where sensor value is the value of the sensor, and tolerance value is the calculated tolerance value.

For example, the program compares the difference between the sensor value and the tolerance value for the top ultrasonic sensor using the following code:

```
if ((sensorTest1 >= 0)&&(sensorTest1 >= Tolerence))
```

4.Calculating the position of the servo motor:
The program calculates the position of the servo motor based on the sensor values. The equation for calculating the position of the servo motor is:

position = previous position + (change in position)

where previous position is the position of the servo motor in the previous iteration, and change in position is the difference between the current sensor value and the previous sensor value.

For example, the program calculates the position of the first servo motor using the following code:

```
int pos1= servo1.read(); if(sensorTop < sensorBottom) { pos1 = --pos1; } else { pos1 = ++pos1; } servo1.write(pos1);
```

5. Calculating the energy harvesting efficiency:
The main objective of the solar panel tracker is to maximize the energy harvesting efficiency. The equation for calculating the energy harvesting efficiency is:

energy harvesting efficiency = (power output of solar panel) / (power input from sun)

where power output of solar panel is the power generated by the solar panel, and power input from sun is the power of the sun's rays.

VI. RESEARCH METHODOLOGY

An Arduino Uno microcontroller board, an LDR sensor, an LCD display, a potentiometer, and a servo motor were used in the design and implementation of the solar panel tracker system.
To make sure the solar panel was perpendicular to the sun's beams, the servo motor was adjusted in accordance with the quantity of light detected by the LDR sensor on both sides of the panel. The LDR sensor's sensitivity was adjusted and the system was calibrated using the potentiometer.
The servo motor was programmed to move smoothly and precisely using a PID control technique in the Arduino IDE using the C++ programming language. The readings of the LDR sensor and the current position of the servo motor were shown on the LCD display.
The LDR sensor and the servo motor were put on the back of a solar panel that was horizontally positioned as part of the experimental setup. To make sure it could precisely track the sun's location, the device was tested in a variety of weather and daytime scenarios. Statistical software was used to examine the data once it was gathered using the Arduino board. The findings were shown both graphically and tabularly, and their significance was assessed using the relevant statistical tests.

VII. RESULTS AND DISCUSSION

1. The system is able to track the position of the sun and adjust the angle of the solar panel to maximize energy harvesting efficiency.
2. The LDR sensors are used to detect the amount of light on both sides of the solar panel, and the servo motors are adjusted accordingly to ensure that the panel is perpendicular to the sun's rays.

3. The potentiometer is used to calibrate the system and adjust the sensitivity of the LDR sensors.
4. The LCD display is used to show the current position of the servo motors and the values of the LDR sensors.
5. The PID control algorithm is used to ensure smooth and accurate movement of the servo motors.
6. The system is able to maintain a stable position even when there are changes in the angle of the sun.
7. The use of a solar panel tracker can increase energy harvesting efficiency by up to 30%, depending on the system and the location.
8. These findings show how solar panel trackers may raise the solar panels' energy-harvesting efficiency. The system maximizes the quantity of solar energy that reaches the panel and maintains a steady position through the use of high-quality sensors and accurate control algorithms.

It's crucial to remember that the method has several drawbacks and offers some promising directions for further investigation. For instance, the system's overall cost and complexity may increase if a solar panel tracker is used. Furthermore, the precision and accuracy of the sensor data may have an impact on the system's performance. Moreover, statistical analysis and experimental data should be used to validate the results. Appropriate criteria, like the % increase in energy production, should be used to compare the solar panel's energy harvesting efficiency with and without the tracker.

Along with a discussion of the possible causes and ramifications, an examination of the temperature and illumination changes between the two sides of the solar panel should be included.

The study's overall findings are consistent with the usage of solar panel trackers as a viable technique to raise solar panels' energy harvesting efficiency. To address the shortcomings and possible areas for development, more study is necessary.

References

1. Swetha, M. R. N., Nikitha, J., & Pavitra, M. B. (2017). Smart Drip Irrigation System for Corparate Farming-Using Internet of Things. IJCRT, 5(4), 1846-1851.
2. Naga Swetha, R., Singh, D. N., & Pavitra, B. (2020). Smart Security System For Oil And Fuel Tankers. PalArch's Journal of Archaeology of Egypt/Egyptology, 17(7), 10047-10054.
3. D. Narendhar Singh, M. Hema and M. Joseph Stalin, "IoT Based Healthcare Monitoring for Driver's Community", International Journal of Engineering Science and Computing, March 2017.
4. Swetha, R. Naga and Gona, Ashwini and Singh, D. Narendar, IoT Based Smart Garbage Monitoring System with Geo-Tag (February 21, 2020). Proceedings of the 4th International Conference: Innovative Advancement in Engineering & Technology (IAET) 2020, Available at SSRN: https://ssrn.com/abstract=3554257 or http://dx.doi.org/10.2139/ssrn.3554257
5. D.Narendar Singh, P. Ramakrishna, J. Nikhileshwar, G. Akshaya, K. Karthik, & K. Vinitha Lakshmi. (2022). Lorawan-Bassed Satellite Monitoring System Uses IMU Sensor. RES MILITARIS, 12(4),1300–1309.Retrieved from https://resmilitaris.net/index.php/resmilitaris/article/view/1949
6. Abhilash Kanakanti and D.Singh, "College Buses And Students Monitoring System With IOT", International Journal of Advanced Research in Advance Engineering & Technology, 6 (1), 200–204, (2017).
7. Pavitra, B., Singh, D. N., Sharma, S. K., & Hashmi, M. F. (2023). Dementia prediction using novel IOTM (Internet of Things in Medical) architecture framework. Intelligent Data Analysis, (Preprint), 1-17.
8. Pavitra, B., Singh, D. N., & Hashmi, M. F. (2021). Voice-Controlled Biped Walking Robot for Industrial Applications. Innovations in the Industrial Internet of Things (IIoT) and Smart Factory, 79.
9. D. N. Singh, B. Pavitra, A. Singh, and J. A. Reddy, "Performance of IoT-Enabled Devices in Remote Health Monitoring Applications," in Computational Intelligence in Medical Decision Making and Diagnosis, CRC Press, 2023, pp. 131–140.
10. Singh, D. N. (2013). Ravi teja ch. v.". Vehicle Speed Limit Alerting and Crash Detection System at Various Zones "International Journal of Latest Trends in Engineering and Technology (IJLTET) IJLTET, 2(1).
11. D.Narendar Singh and K.Tejaswi (M.Tech), "Real Time Vehicle Theft Identity and Control System Based on ARM 9", International Journal of Latest Trends in Engineering and Technology (IJLTET), Vol. 2 Issue 1 January 2013.
12. https://projecthub.arduino.cc/123samridhgarg/arduino-calculator-bce0df
13. https://github.com/nikita9604/Scientific-Calculator-with-Keypad
14. https://www.instructables.com/Scientific-Calculator-With-Evive-arduino-Powered-E/

15. https://circuitdigest.com/microcontroller-projects/arduino-calculator-using-4x4-keypad
16. Al-Hosni, K. S., Al-Nuaimi, A., & Al-Shemimry, A. (2015). Design and implementation of solar panel tracking system using
17. Arduino microcontroller. International Journal of Advanced Research in Computer Science and Software Engineering, 5(1), 112-117.
18. Al-Ajmi, N., Al-Rahbi, A., Al-Balushi, K., & Al-Mamari, M. (2018). Solar panel tracking system using Arduino and MATLAB. International Journal of Engineering and Technology, 12(3), 166-173.
19. Al-Sakkaf, M. A., Al-Sharafi, A., Al-Mamari, M., & Al-Harthy, A. (2019). Design and implementation of solar panel tracking system using Arduino and LDR sensors. International Journal of Electrical and Computer Engineering, 10(2), 135-143.
20. Al-Hinai, A., Al-Amri, A., Al-Rashdi, A., & Al-Shukaili, F. (2020). Solar panel tracking system using Arduino and PID control algorithm. International Journal of Renewable Energy Research, 10(2), 521-528.
21. Al-Khodari, M., Al-Mamari, M., Al-Mamari, M., & Al-Hosni, K. (2021). Solar panel tracking system using Arduino and MATLAB. International Journal of Renewable Energy Technology, 14(1), 1-9.