SHRI RAMDEOBABA COLLEGE OF ENGINEERING AND MANAGEMENT, NAGPUR.

Mini Project/Electronic Design Workshop Report

(6TH SEM, SESSION 2023-2024, ECP358)

"Sun Track Curtain Automation for Power Saving"

Submitted By

Batch: A2
Group Number: 3

Mayank Rai (Roll No.-22, Sec-A)
Mayank Mundhada (Roll No.-23, Sec-A)
Navya Choudhary (Roll No.-36, Sec-A)
Madhavi Mate (Roll No.-38, Sec-A)

Guide Name: Prof. N M Borkar

Workshop Coordinator: Prof. T Harinkhede

Department of Electronics and Communication Engineering

Contents

1.	**Introduction**	**Pg No.1**
2.	**Impact of Project on society and the environment**	**Pg No.2**
3.	**Block diagram and Functional description**	**Pg No.4**
4.	**Circuit diagram and its description**	**Pg No.6**
5.	**PCB layout Artwork**	**Pg No.10**
6.	**Working of Project**	**Pg No.12**
7.	**Result and Future Scope**	**Pg No.15**
8.	**References/ Citations**	**Pg No.19**

1. Introduction

It is now essential to integrate technology with daily operations in an era where people are becoming more concerned with energy efficiency and ecological living.This project centers around the development of an intelligent system that automatically adjusts curtains or blinds based on the position of the sun throughout the day. By utilizing sensors and motors, the curtains will autonomously open and close, effectively managing the amount of sunlight entering a room. This dynamic response to sunlight not only enhances comfort for occupants but also contributes significantly to energy conservation.

Literature Survey:

Some issues present in the existing products and design are:

Cost :
Existing automated curtain systems can be a budget-buster. High-end options like Lutron Serena shades, with features like voice control and smart home integration, easily cost hundreds of dollars per window. This puts them out of reach for many people who are interested in the benefits of automated curtains.[2]

User experience :
Beyond the cost barrier, some existing automated curtain systems can be frustrating to use. Complex setup procedures, apps requiring technical knowledge for programming, and un-intuitive interfaces for calibration and scheduling can create a hurdle for non-technical users. These shortcomings can turn a potentially convenient smart home feature into a source of frustration.[3]

Customization :
Pre-programmed systems often lack the flexibility to adapt to individual needs. They might be designed for standard blinds or lightweight curtains, leaving users with heavier drapes out of luck. This lack of customization forces people to choose between a system that doesn't work for their curtain type or purchasing a separate, stronger motor, essentially negating the advantage of a pre-built solution[2].

Integration :
Fitting into your existing setup can be another hurdle with current automated curtains. Some systems might require modifications to your curtains or even window frames for installation, leading to additional costs and potential damage. For example, a system designed for specific curtain tracks might not work with

your existing curtain rods, forcing you to replace them entirely or find custom adapters – adding unnecessary complexity and expense to the process.[4]

Focus on Automation Over Comfort:

Current automated curtain systems can be overly focused on automation at the expense of comfort. They might prioritize basic light sensors to open curtains for natural light, neglecting factors like temperature. This can lead to uncomfortably hot rooms during sunny afternoons or excessive glare on screens, negating the benefits of natural light and potentially creating a frustrating user experience.

The proposed Sun track curtain automation system for power saving is a practical, user-friendly and efficient solution designed to promote energy savings, a more natural indoor environment, and improved user comfort.This automated system integrates various components, including an Arduino nano, LDR,7805 Voltage regulator, DC motor, relay modules, TSOP and jumper wires to facilitate a system that automatically adjusts curtains based on sunlight, reducing reliance on artificial lighting and saving energy.

2. Impact of Project on society and the environment

- *Energy Conservation:* By automating curtains to adjust based on sunlight levels, energy consumption for heating and cooling can be reduced. This leads to lower electricity bills for households and businesses, contributing to energy conservation efforts and reducing reliance on non-renewable energy sources.

- *Economic Benefits:* While there may be initial investment costs associated with installing smart curtain systems, the long-term savings on energy bills can result in significant cost reductions for homeowners and businesses. Additionally, the growth of the smart home automation industry creates economic opportunities for manufacturers, installers, and service providers.

- *Accessibility and Convenience:* Smart curtain systems offer convenience and accessibility benefits, especially for individuals with mobility challenges or disabilities. Automated control features enable users to adjust the curtains effortlessly, enhancing their independence and quality of life.

- *Improved Indoor Air Quality:* Reducing the use of artificial lighting can also lead to lower heat emissions indoors, which can contribute to a more comfortable and healthier indoor environment.

- *Scalability and Replicability:* Successful implementation of Suntrack curtain automation projects can inspire similar initiatives in other regions and communities, leading to widespread adoption of energy-efficient technologies. This scalability and replicability amplify the positive impact on society and the environment, creating a ripple effect of sustainability.

- *Awareness and Education:* Initiatives focused on energy-saving technologies raise awareness about the importance of energy conservation and sustainable living practices. They educate individuals and communities about simple yet effective ways to reduce their environmental footprint and contribute to a cleaner, greener planet.

3. Block diagram and Functional description

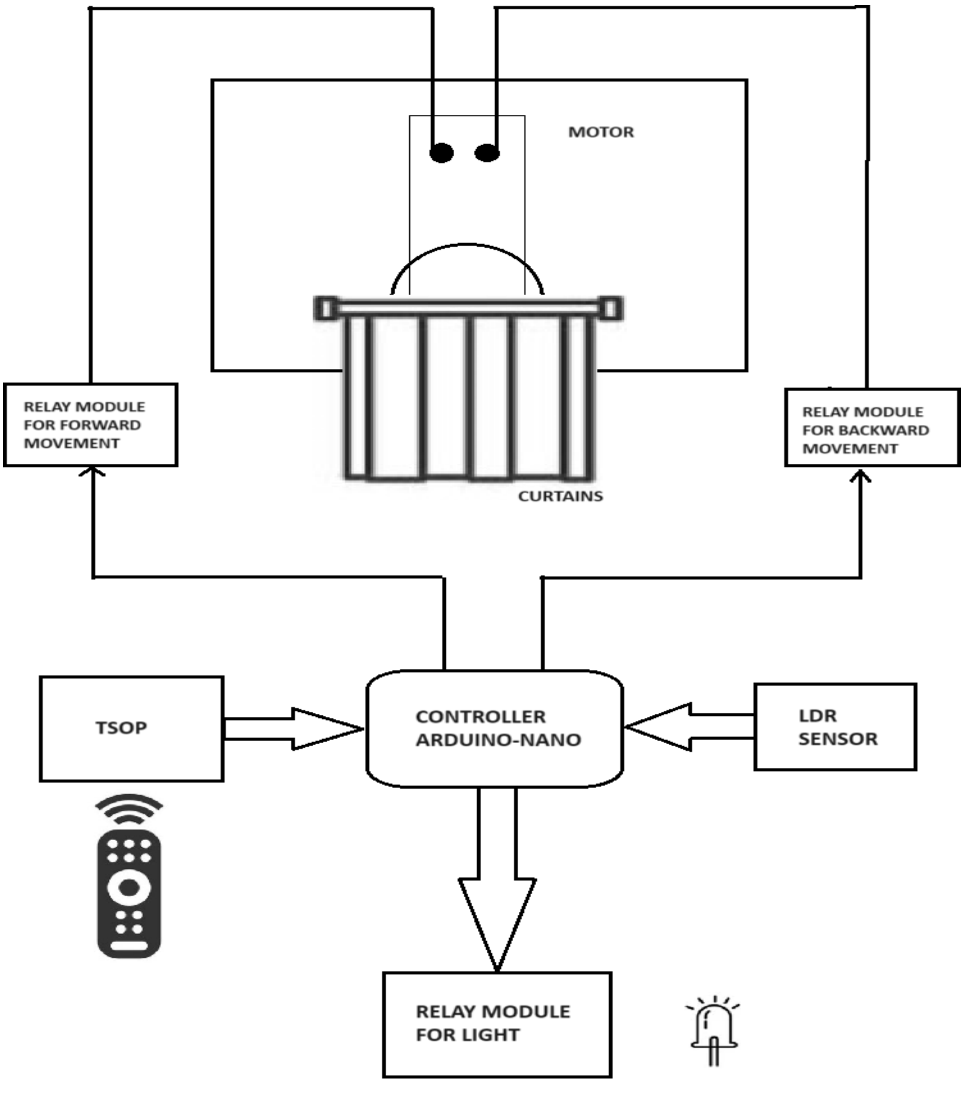

Fig: 3.1-Block Diagram of Proposed Model

- Arduino NANO: Our project uses an Arduino Nano as its central controller to programme the curtains to open and close in reaction to sunlight. It is perfect for this work because of its sturdy characteristics and small size. The Arduino Nano precisely detects changes in the ambient light levels and initiates the necessary action by integrating with the light sensor and relay.[4] Furthermore, the adaptability of the Arduino Nano enables customisation and integration with other smart home appliances, providing customers with convenience and freedom in controlling their living space.

- Relay module for forward movement: This relay is dedicated to moving the curtains forward, facilitating their opening to allow sunlight into the room.[3]

- Relay module for backward movement: The second relay orchestrates the backward movement of the curtains, closing them when sunlight levels decrease or at night.[3]

- TSOP: TSOP (IR Receiver) is used as a critical interface for remote control functionality. The system accepts infrared signals from a remote control device by incorporating the TSOP into the circuit. This enables convenient remote operation by enabling users to wirelessly command the opening and closing of curtains with simplicity. The TSOP interprets the incoming infrared signals and converts them into commands that the Arduino Nano can follow. This feature makes it easy for users to modify the curtains to suit their tastes, which improves the system's overall usability.[1]

- LDR sensor: The LDR (Light Dependent Resistor) sensor plays a pivotal role in detecting changes in ambient light levels. Mounted strategically near the curtains, the LDR continuously monitors the intensity of sunlight entering the room. When sunlight is detected, the LDR sends a signal to the Arduino Nano, prompting it to activate the curtain motor and open the curtains. Conversely, when sunlight levels diminish, indicating nighttime or overcast conditions, the LDR triggers the system to close the curtains, ensuring optimal light management throughout the day.[3]

- Relay module for light: Furthermore, the third relay plays a dual role by switching an LED on and off, providing visual feedback or additional illumination as needed. This modular approach enhances the versatility and functionality of our system, offering both efficient curtain control and customizable lighting solutions.

4. Circuit diagram and its description

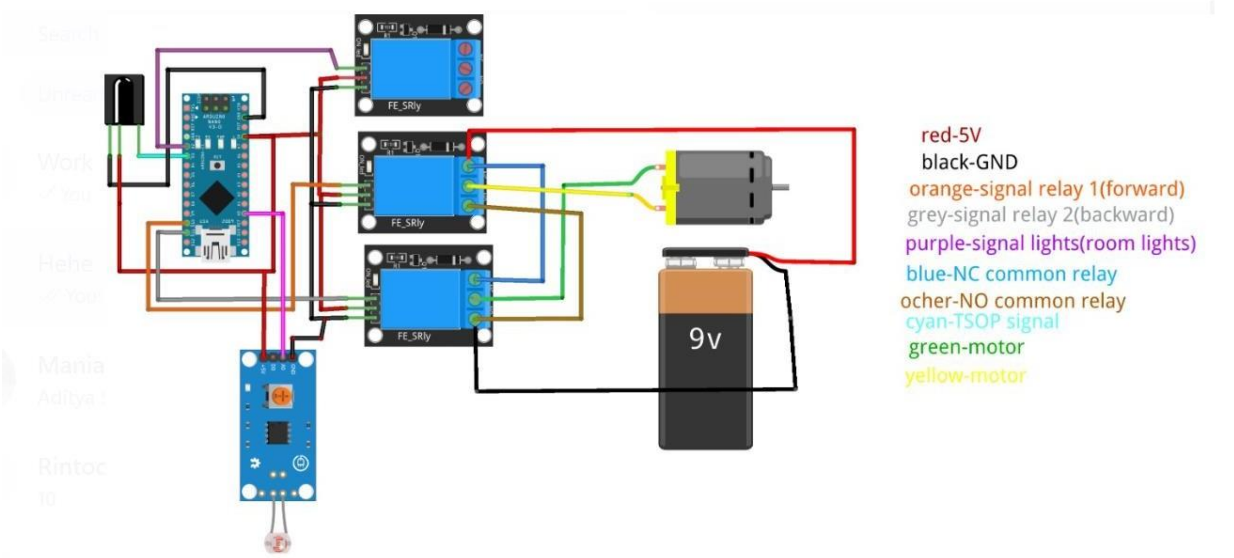

Fig: 4.1-Circuit Diagram

1. Arduino NANO: The Arduino Nano is a compact and versatile microcontroller board based on the ATmega328P chip, offering a powerful yet flexible platform for various embedded projects. Its small form factor, extensive connectivity options, and rich ecosystem of libraries and resources make it an ideal choice for implementing automation and control systems like the "Sun Track Curtain Automation for Power Saving" project.[4]

   Key features:
   - The Arduino Nano is powered by the ATmega328P microcontroller clocked at 16 MHz, providing sufficient processing power for real-time control tasks.
   - It features 32KB of flash memory for program storage and 2KB of SRAM for variable storage, accommodating the code and data required for the project.
   - With dimensions of just 45mm x 18mm, the Nano can be easily integrated into space-constrained environments such as control panels or enclosures.
   - It is fully compatible with the Arduino IDE and can be programmed using the Arduino programming language (based on C/C++), allowing for rapid prototyping and development.

2. LDR module: The Light Dependent Resistor (LDR) module, also known as a photoresistor module, is a passive electronic component that exhibits a

change in resistance based on the intensity of incident light. It serves as a key sensing element in the "Sun Track Curtain Automation for Power Saving" project, enabling the system to detect variations in ambient light levels and adjust the position of curtains accordingly.[5]

Key features:

- The module integrates an LDR sensor, which consists of a semiconductor material whose resistance decreases with increasing light intensity. This property allows the LDR to act as a light-sensitive variable resistor.
- The LDR sensor is typically enclosed in a protective casing to shield it from environmental factors such as dust, moisture, and physical damage, ensuring reliable operation over time.
- The module provides an analog output signal proportional to the resistance of the LDR, allowing the microcontroller (e.g., Arduino Nano) to measure and interpret changes in light intensity.
- LDR modules typically consume minimal power, making them suitable for battery-operated or energy-efficient applications such as solar-powered sensor nodes.

3. Relay module: A relay module is an electromechanical switch that uses an electromagnet to mechanically control the switching of one or more electrical circuits. It serves as a crucial component in the "Sun Track Curtain Automation for Power Saving" project, facilitating the control of motorized curtains or blinds based on inputs from sensors like the LDR module.[1]

Key features:

- The relay module typically contains one or more electromechanical relays, each consisting of a coil, an armature, and one or more sets of contacts. When the coil is energized, it generates a magnetic field that attracts the armature, causing the contacts to close or open, depending on the relay type (normally open or normally closed).
- Relay modules come in various configurations with different switching capacities, allowing them to control a wide range of loads, from low-power devices such as LEDs to high-power appliances such as motorized curtains or blinds.
- Relay modules are available in both 5V and 12V versions, with the input voltage determining the voltage required to energize the relay coil. This voltage is typically compatible with the output voltage of microcontrollers such as the Arduino Nano.

4. TSOP: The TSOP (TSAOP) - Infrared Receiver Module is a specialized sensor module designed to receive infrared (IR) signals from remote controls or other IR transmitters. In the context of this project, the TSOP module enables remote control functionality, allowing users to manually override the automated curtain positioning system and adjust curtain settings as needed.[3]

   Key features:

   - The TSOP module contains a photodetector that is sensitive to infrared radiation in the 38 kHz frequency range, which is commonly used for remote control applications.
   - It includes internal circuitry to demodulate and filter incoming IR signals, extracting the modulated data from the carrier frequency and providing a clean output signal to the microcontroller for processing.
   - The TSOP module typically has a wide reception angle, allowing it to receive IR signals from various directions within its field of view, enhancing user convenience and flexibility during operation.
   - To improve signal reliability and immunity to interference, the TSOP module incorporates bandpass filtering and noise rejection features, ensuring accurate detection of valid IR commands while rejecting spurious signals.

5. 7805 Voltage Regulator: The 7805 is a popular linear voltage regulator integrated circuit (IC) that provides a stable output voltage of +5 volts DC (Vout) from an unregulated input voltage (Vin). It serves as a crucial component in the "Sun Track Curtain Automation for Power Saving" project, ensuring that the microcontroller (e.g., Arduino Nano) and other low-voltage components receive a consistent and reliable power supply.[1]

   Key features:

   - The 7805 voltage regulator produces a fixed output voltage of +5 volts DC, making it suitable for powering digital logic circuits, microcontrollers, sensors, and other electronic components that require a stable power supply.
   - The 7805 can accept a wide range of unregulated input voltages, typically between +7 volts and +25 volts DC, allowing it to accommodate various power sources such as batteries, DC adapters, or solar panels.

- The 7805 has a standard three-terminal configuration, comprising an input pin (Vin), an output pin (Vout), and a ground pin (GND), simplifying connection and integration into electronic circuits.

5. PCB layout and Artwork

Creating a PCB (Printed Circuit Board) layout for Suntrack Curtain Automation System for Power Saving System involves following steps:-

1. Schematic Design: Started by designing the schematic of our circuit using the software EasyEDA.Placed components such as the Arduino Nano, relays, LDR sensor, DC motor, and power supply on the schematic and connected them using wires to represent the electrical connections.

2.*Component Placement:* We arranged the components on the PCB layout in a logical and organized manner considering factors such as signal flow, component size, and proximity to minimize trace lengths and optimize the layout for efficient PCB manufacturing.

3.*Routing:* Using the routing tools provided by the EasyEDA to create traces that connect the pins of each component according to the schematic. We need to pay attention to signal integrity, avoiding crossing traces, and ensuring proper ground and power connections.

6.*Design Rule Check (DRC):* We performed a design rule check to ensure that the PCB layout complies with the manufacturing constraints and specifications.Errors such as overlapping traces, insufficient clearances, and unconnected nets may occur while checking.

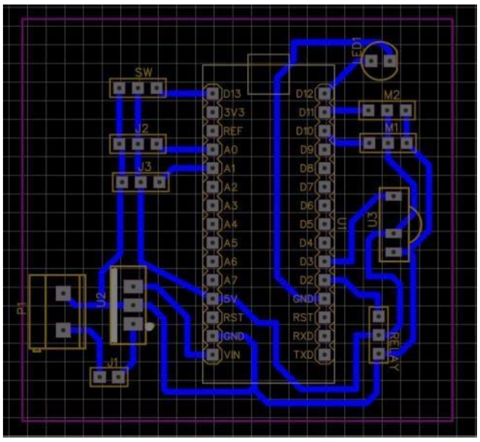

Fig: 5.1 -PCB layout of proposed solution

7.*Gerber File Generation:* Once the layout is finalized, we generated Gerber files, which are the standard file format used for PCB manufacturing. These files include information about the copper traces, solder mask, and silkscreen layers required to fabricate the PCB.

8.*PCB Fabrication:* Using the Gerber file we manufactured the PCB layout. This process included choosing appropriate PCB material, thickness, with use of etching solution.

9.*Assembly:* After obtaining the desired PCB, we proceeded with component assembly soldering the electronic components onto the PCB using through-hole components.

10.*Testing:* Finally,we thoroughly tested the assembled PCB to ensure that it functions as intended ,verifying the functionality of each component, checking for any shorts or open circuits, and debugging any issues that arise during testing.

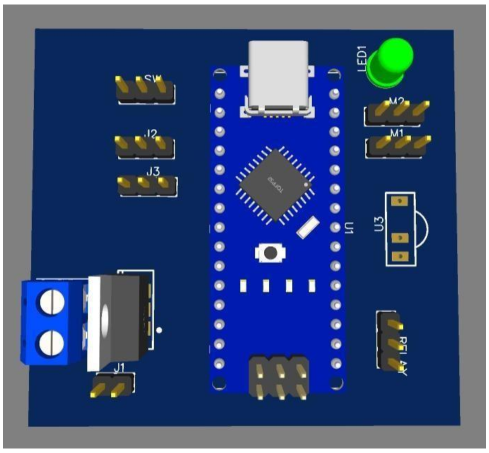

Fig: 5.2- 3D view of proposed model

6. Working of Project:

Our project's aim is to make your life easier and more comfortable. The system uses natural light during the day by automatically opening the curtains, which not only brightens your space but also reduces your reliance on artificial lights, potentially lowering your electricity bill. As night falls, the curtains close automatically, creating a good atmosphere and blocking out unwanted light for a good night's sleep. This project is a practical and convenient way to enhance your home environment with a touch of energy-saving smarts.

The Working of the Project :

1. **Initialization of Components:**

First, we start the system. The code uploaded to the Nano gets everything ready to run. It identifies specific pins on the board that will talk to the other components. These components include the LDR, a light sensor that detects changes in brightness; the relays, which act like switches to control the motor and LED; and, optionally, the TSOP, which receives signals from a remote control. The code also configures the relays as outputs, essentially giving the Arduino the power to turn them on or off based on the LDR's readings and the program's logic. With everything set up and constantly communicating, the Nano is ready to take control of your curtains based on light levels or even remote control commands.

2. **LDR Monitoring :**

Once every component is initialized, the Arduino enters into a never-ending loop. This loop ensures constant observation. Within this loop, the Arduino keeps observing the values of LDR, the light sensor. The LDR changes its resistance based on the amount of light it receives. Bright sunshine (daytime) makes the LDR's resistance decrease. Conversely, as the sun sets and the light decreases (nighttime), the LDR's resistance increases.
The Arduino reads the analog value from the LDR pin. This value represents the current light intensity level and is stored in a variable, essentially capturing a snapshot of the surrounding light conditions.

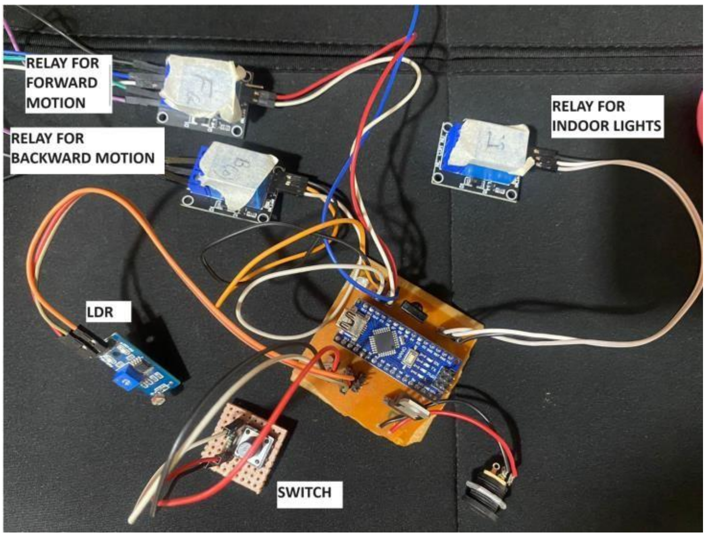

Fig: 6.1- Internal view of the Model

3. **Light Detection and Action (Daylight):**

Equipped with the LDR's light reading, the Arduino code makes the key decision: is it day or night? It compares the reading to a pre-set threshold. The readings below it signify a bright day, triggering a delightful response. The Arduino sends a signal to a specific relay, which energizes the motor. The relay's control circuitry then determines the motor's direction, typically clockwise, to open the curtains. As the motor rotates, your curtains close, and the LED turns off.

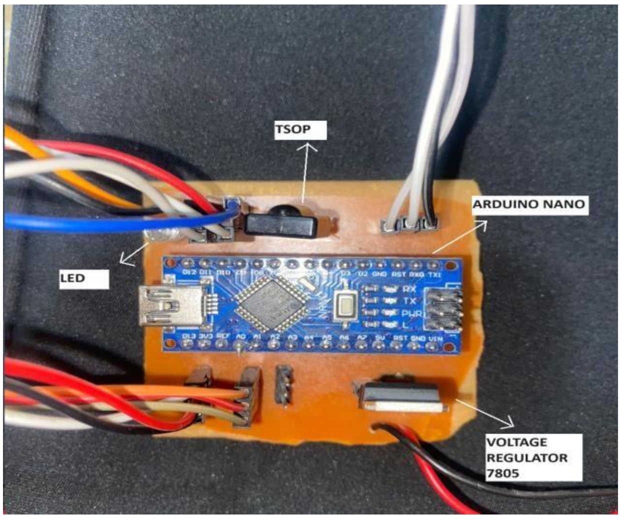

Fig: 6.2 - Closer view of pin connections

4. **Closing the Curtains (Night):**

As the sunlight dims, the LDR's resistance naturally increases, a change the Arduino constantly monitors. When the LDR reading surpasses the pre-defined threshold, the code recognizes it as nightfall—the curtain closing time! The Arduino sends a control signal to a different relay, which activates and flips the motor's direction (typically counterclockwise) via its control circuitry. The motor rotates to close the curtains. Optionally, another relay might be programmed to activate the LED.

5. **Remote Control (Manual):**

For even greater control, we have used a TSOP module and remote control. The TSOP acts like a watchful guard, constantly receiving and deciphering infrared (IR) signals from the remote. The code is programmed to understand specific IR codes linked to desired actions, like opening or closing the curtains or turning lights on or off. When a matching code is received, the Arduino triggers the appropriate response by controlling the relays and motor direction. The Arduino Nano continuously monitors the LDR, interprets light conditions, and controls the relays to open or close the curtains based on pre-defined logic and, if included, remote control commands.

Fig: 6.3- Front View of the Model

7. Result and Future Scope

Advantages

1.*Energy Efficiency:* The system maximizes natural light utilization, lowering the need for artificial lighting during the day by automatically altering the curtains based on sunshine levels. As a result, there is less carbon footprint and energy savings.

2.*Security and Privacy:*When the room is empty or at night, the system automatically draws the curtains to provide security and privacy. This protects residents' privacy and discourages possible intruders.

3.*Enhanced Comfort:*Controlling the amount of sunlight that enters the space contributes to the upkeep of a comfortable interior atmosphere. Depending on their preferences, users can adjust the temperature, reduce glare, and create a comfortable ambiance.

4.*Accessibility:*People with disabilities or mobility problems can access the system more easily. It gives all users more independence and convenience of use by automating the curtain control process.

6.*Customisation:* Users are able to alter the system to suit their tastes and way of life. The system can be customized to meet specific needs by changing the light sensor's sensitivity or setting up precise opening and closing schedules.

7. Smart Home System Integration: The Automatic Curtain Control System may be easily connected to other smart home systems and gadgets, like temperature sensors and lighting controls. This makes it possible to automate and centrally control many elements of the home environment.

Limitations

Hardware Complexity:
The chosen motor's power is crucial. Heavier curtains necessitate a stronger motor, which translates to increased cost and potentially more complex installation.

Software and User Experience:

Relying solely on a single light intensity threshold for decision-making could overlook user preferences and changing weather conditions. .The current setup might not offer flexible scheduling options. This could be a disadvantage, as ideal curtain movement times can vary based on user routines and seasonal changes in daylight hours.

Additional Considerations:
Currently, it doesn't consider factors like temperature or user presence that can also influence optimal curtain adjustments for both comfort and energy savings.
Heavy cloud cover can significantly limit the system's ability to utilize natural light. Furthermore, privacy concerns might arise for users with street-facing windows, as automated curtain movements could be perceived as an intrusion.

Future Scope

Advanced Control Systems:
This could involve implementing machine learning algorithms. These algorithms would analyze historical energy consumption data, weather forecasts, and occupant preferences gleaned from voice commands or a smartphone app. By learning user behavior and environmental factors, the system could predict optimal curtain movements. This would not only maximize energy savings but also respect user desires for natural light throughout the day. Additionally, seamless integration with popular smart home platforms like Google Home and Amazon Alexa would empower users with voice control for convenient adjustments and scheduling based on daily routines or light preferences.

Enhanced Functionality:
Multi-zone control allows curtains in different rooms to adjust independently based on both sun position and room occupancy. This ensures optimal light and privacy in each space. Additionally, integrating light and temperature sensors broadens the system's capabilities. It can then trigger curtain adjustments not just for sun position, but also to maintain a comfortable indoor environment by balancing natural light with artificial lighting or heat gain. Finally, exploring fire safety sensor linkage could provide a crucial safety feature. In case of emergencies, the system could automatically open curtains to facilitate escape and improve visibility for firefighters.

Technological Advancements:
Wireless charging for the curtain motors would eliminate unsightly wires, creating a cleaner and more aesthetically pleasing solution. Furthermore, designing the sun

tracking system to be solar powered would make it entirely self-sustainable, drawing energy from the very sunlight it manages. But the true leap forward lies in exploring voice-activated fabric customization. Imagine using voice commands to adjust the transparency or insulation properties of the curtain fabric itself. Maximize natural light during the day with a simple voice command, or automatically switch to a more insulating fabric at night for improved energy savings.

User Experience and Design:
Develop a user-friendly smartphone app that allows for convenient control of curtain movements, offering features like manual adjustments, scheduling based on daily routines, and even the ability to view energy savings data directly on the phone. This empowers users and provides valuable insights into the system's impact. Additionally, noise reduction features incorporated into the motorized curtain systems can ensure quiet operation, eliminating any unwanted noise disruption. Finally, prioritize aesthetics and customization by designing the curtain system to be visually appealing. Offering a range of color options or even the ability to integrate different curtain fabrics would allow users to seamlessly blend the system into their existing home decor, creating a cohesive and stylish solution.

Result:

The implementation of the Sun track curtain automation system for power saving systems yielded several significant outcomes, demonstrating its effectiveness in enhancing the natural and stimulating indoor environment, comfort and potential energy savings, potentially improving mood and focus.
The Sun track curtain automation system prototype was successfully developed and demonstrated functionality. The system effectively integrated components such as the Arduino Nano, LDR, 7805 Voltage regulator, DC motor, relay modules, TSOP and jumper wires to facilitate a system that automatically adjusts curtains based on sunlight, reducing reliance on artificial lighting and saving energy. The project operates based on sunlight detection and user-defined thresholds. The LDR sensor, strategically placed to receive direct sunlight, experiences a decrease in resistance as sunlight intensity increases. This change in resistance translates to a voltage fluctuation that the Arduino Nano reads through an analog pin. The program continuously compares this voltage reading to a pre-programmed light intensity threshold. If the voltage is higher than the threshold (indicating low light), the system assumes darkness and keeps the curtains closed (default state). However, if the voltage falls below the threshold (indicating strong sunlight), the program interprets this as daytime and sends a control signal to the relay modules. These

modules then activate the DC motor, causing it to rotate and adjust the curtains (open or close) based on the prevailing light level. By integrating the TSOP with our Arduino program, we enabled users to open or close the curtains using a standard remote control that emits IR signals. The TSOP decodes the specific signal we program it to recognize (e.g., a specific button press) and triggers the Arduino to send the appropriate signal to the motor driver circuit (relays) for opening or closing the curtains.

In conclusion, the sun tracking smart curtain project is a promising approach to promoting energy savings, a more natural indoor environment, and improved user comfort. By addressing limitations and exploring advanced features, we can significantly enhance its impact and create a truly intelligent and user-centric solution.

8. References/ Citations

[1] https://components101.com/ics/l293d-pinout-features-datasheet

[2] https://jurnal.itscience.org/index.php/brilliance/article/view/1456/1014

[3] https://www.electronicsforu.com/electronics-projects/prototypes/automatic-curtain-operation-using-ldr

[4] https://www.google.com/url?sa=t&source=web&rct=j&opi=89978449&url=https://forum.arduino.cc/t/smart-curtain-automation/532490&ved=2ahUKEwjssqPGusaFAxUid2wGHRyTCO84FBAWegQICBAB&usg=AOvVaw3za_WmDmZNoBwwYaRs3hkz

[5] https://www.google.com/url?sa=t&source=web&rct=j&opi=89978449&url=https://www.instructables.com/Automatic-Curtain-With-Arduino/&ved=2ahUKEwiWjeW7usaFAxWMUGcHHbIcAVkQFnoECCkQAQ&usg=AOvVaw1oe5fiKl80WrSU4sbjxbm0

[6] https://www.instructables.com/Infrared-Remote-and-IR-Receiver-TSOP1738-With-Ardu/

[7] https://www.instructables.com/Automatic-CurtainWindow-Blind-Using-Arduino-and-LD/

[8] https://forum.arduino.cc/t/automatic-curtain-code-review-for-improvements-and-potential-pitfalls/1126069

[9] https://github.com/CemraJC/smart-autocurtains/blob/master/libraries/CurtainControl/CurtainControl.cpp

[10] https://www.ijsr.net/archive/v13i1/SR231223111025.pdf

[11]https://randomnerdtutorials.com/