TOUCH FREE SWITCH CONTROL PANEL FOR ELEVATORS

AIM:

The aim of the project is to develop and enhance the touch Free Switch Control Panel for Elevators

PROBLEM STATEMENT:

In the context of elevator control panels, traditional push-button switches require physical contact, which can pose hygiene concerns, especially in high-traffic areas. Additionally, frequent use of these switches can lead to wear and tear, resulting in maintenance issues and potential downtime of the elevator system. To address these challenges, there is a need for a touch-free switch control panel for elevators that offers hands-free operation, improved hygiene, and reduced maintenance requirements.

SOLUTION:

The main aim of this proposed system is to develop hand gesture-based switch control panel for touch-free user interface of elevator/lift to minimize risk of spreading of corona virus by touching the switches, and which can also be scalable to any kind of applications to control its various operations using simple hand gestures so that any common man can use it easily without having any specific knowledge of its operations.

The proposed system is built on Arduino UNO and has OLED module for the user interface, Gesture sensor for user input and Motor controller to control Motor of the Elevator/Lift. Functional block diagram of the proposed system is

Touch-free elevator control panels are becoming increasingly essential due to the growing emphasis on hygiene and contactless technology.

Several effective solutions are available:

Considerations for Implementation:

- **Reliability and Durability:** Ensure the chosen technology is robust and can withstand heavy use.
- **User Experience:** Design the interface for intuitive and easy operation.
- **Hygiene:** Consider materials that are easy to clean and disinfect.
- **Compatibility:** Ensure compatibility with existing elevator systems.

- **Cost-Effectiveness:** Balance the desired features with the budget.

Leading Providers:

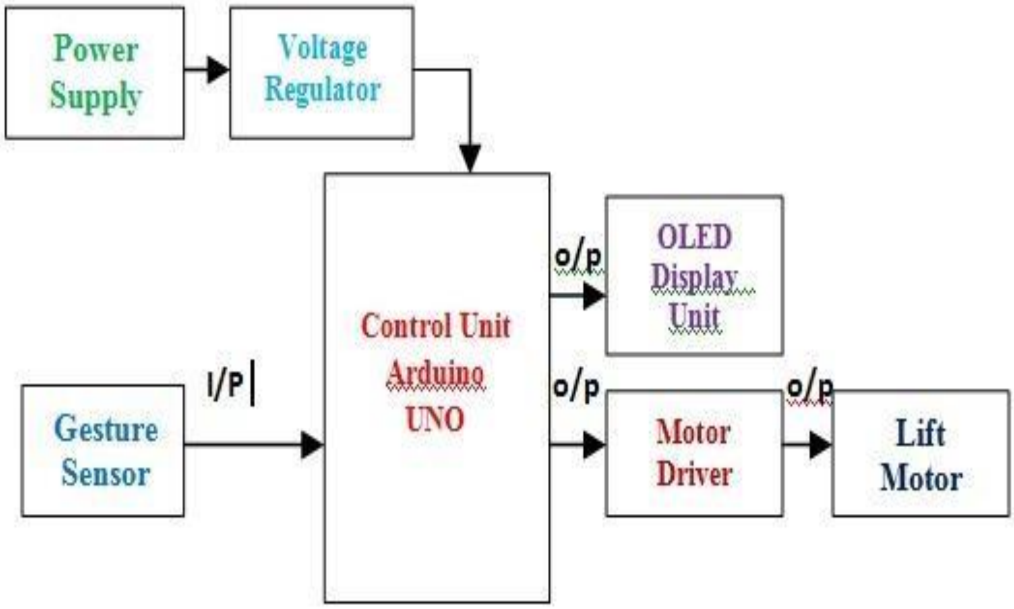
Several companies specialize in touch-free elevator control solutions:

- **Schindler:** Offers kNOw Touch, a popular infrared-based system.
- **Otis:** Provides touch-free options as part of their advanced elevator packages.
- **Kone:** Offers touch-free solutions with various sensor technologies.

PROJECT DESIGN SPECIFICATION AND ARCHITECTURE:

The goal of this project is to design and develop a touch-free control panel for elevators, aiming to enhance user experience, improve hygiene, and comply with modern contactless technology trends. This document outlines the project's design specifications and architectural framework.

Users can perform hand gesture by simply moving their hand over the sensor in the required direction. The APDS9960 Sensor reads the hand gestures. The sensor supports UP, DOWN, LEFT and RIGHT gestures. Make sure each gesture begins outside of the range of the sensor, moves into the range of the sensor, and ends outside the range of the sensor. In this project UP & DOWN gestures are used to set the floor number where the user wants to go to, The LEFT gesture is to close the Elevator/lift door and move the lift according to the floor number and the RIGHT gesture is used to stop the Elevator/lift & open the door.



If a user made a gesture it is detected by the gesture sensor and informs Arduino to read the gesture. Arduino compares the detected gesture and set the function for it. If the detected gesture (UP) indicates going in the upward direction, then set the function for it. The function will allow an increase in the number of floors. So suppose user want to reach the 3rd floor. To go there, simply make a hand gesture for going upwards three times. On doing this, the control panel will capture your gesture and set the Elevator/lift mechanism to go to the third floor. After deciding which floor user wants to go to, make a leftward gesture so that the lift control panel can acknowledge the decision. If you want to get off at an earlier floor, say 2nd floor due to an emergency, then make a rightwards gesture to stop the lift. All these gestures resemble the functions like up, down, start and stop that are normally present in a Elevator/Lift.

Design Specifications

Functional Requirements

- Touch-free operation: Utilize infrared, capacitive, or ultrasonic sensors to detect user proximity and gestures.
- Floor selection: Enable users to select desired floors without physical contact.

- Call and hold functions: Implement functionality to call the elevator and hold it at the current floor.
- Door open/close: Provide options to open and close elevator doors without physical buttons.
- Emergency stop: Maintain a physical emergency stop button for safety.
- Visual feedback: Display clear visual indicators for selected floor, elevator status, and door operation.
- Audio feedback: Provide audible confirmation for user actions.
- Accessibility: Ensure the system is accessible to people with disabilities, including visual and auditory impairments.

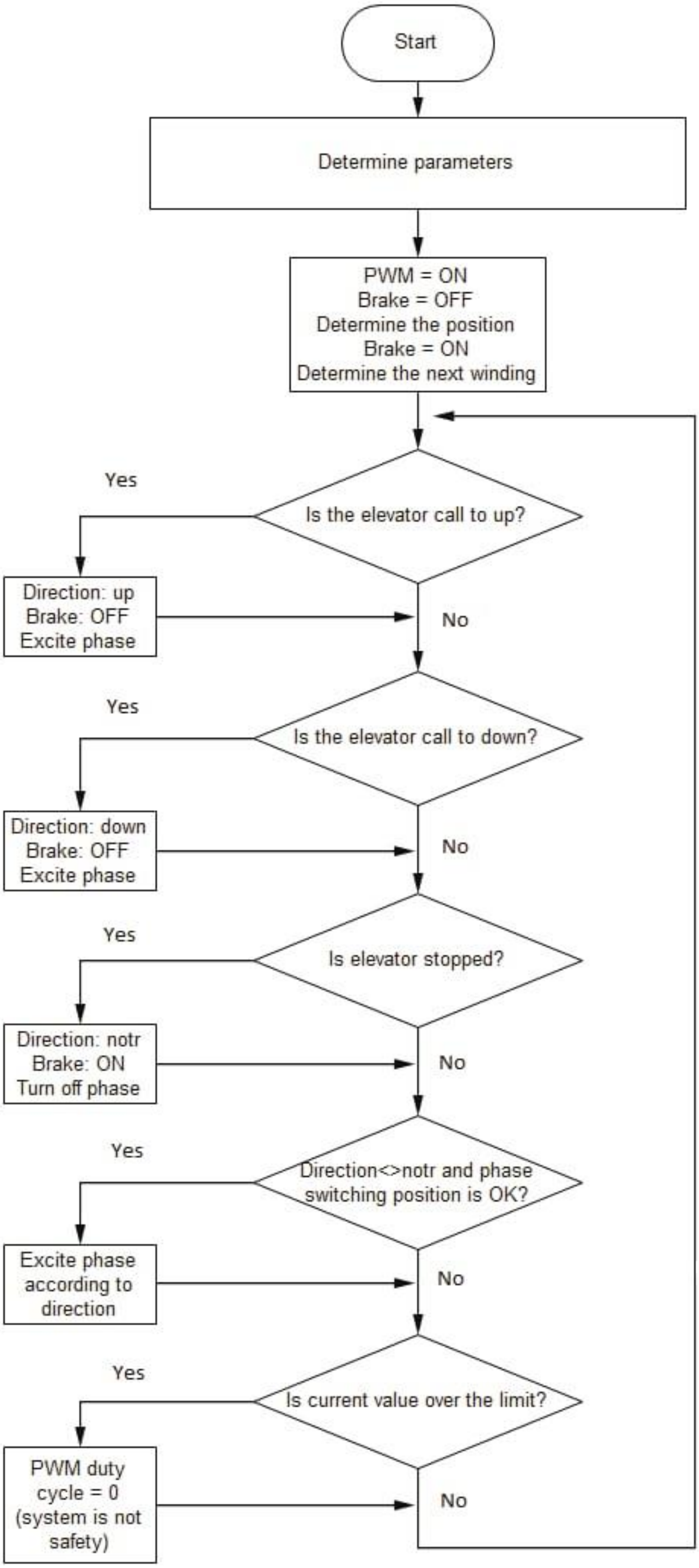
Non-Functional Requirements

- Reliability: The system should operate reliably under various environmental conditions.
- Security: Implement measures to protect user data and prevent unauthorized access.
- Maintainability: Design the system for easy maintenance and troubleshooting.
- Energy efficiency: Optimize power consumption to reduce environmental impact.
- Compatibility: Ensure compatibility with existing elevator systems and infrastructure.

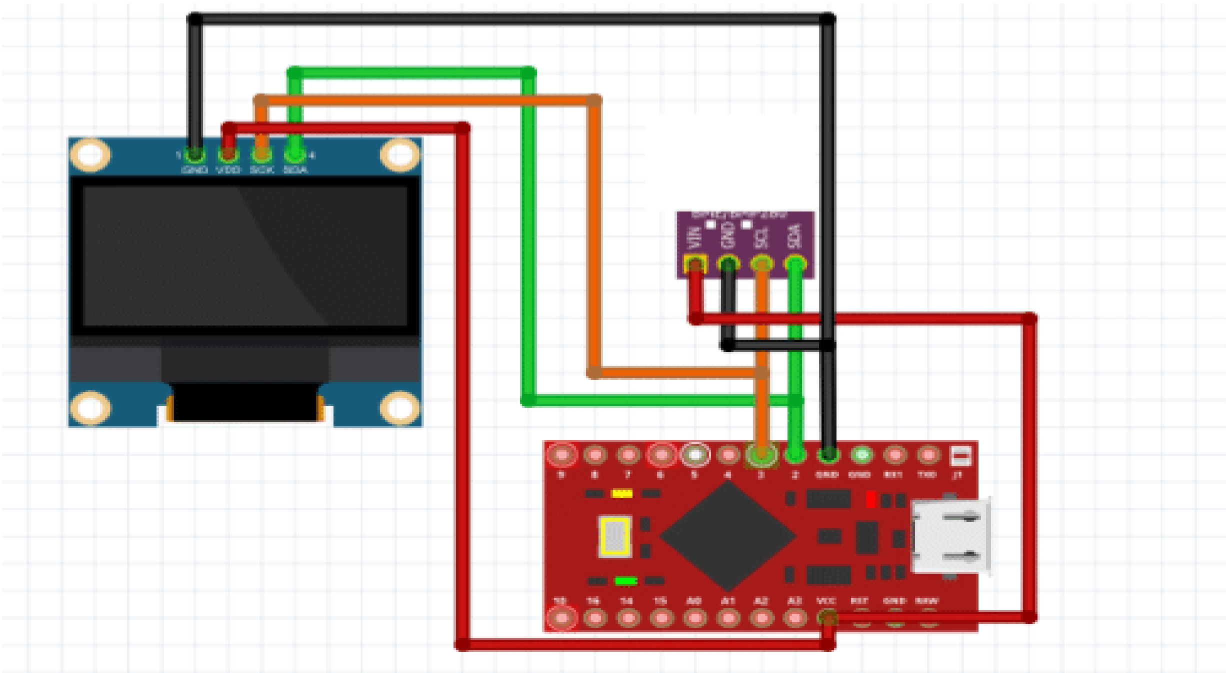
Design Considerations

- Sensor selection: Choose the most suitable sensor technology based on factors like accuracy, response time, cost, and environmental conditions.
- User interface: Create an intuitive and user-friendly interface with clear visual and auditory feedback.
- Error handling: Implement robust error handling mechanisms to address system failures and unexpected user inputs.
- Safety: Prioritize safety by incorporating redundant systems and emergency stop features.

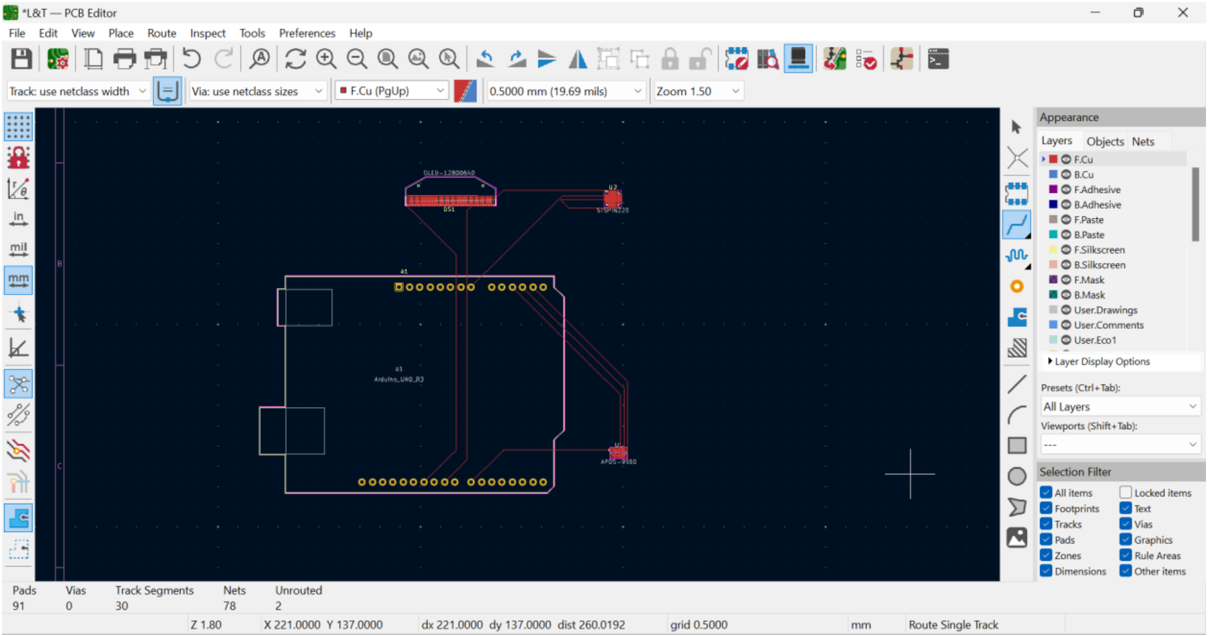
- Accessibility: Adhere to accessibility standards to accommodate users with disabilities.

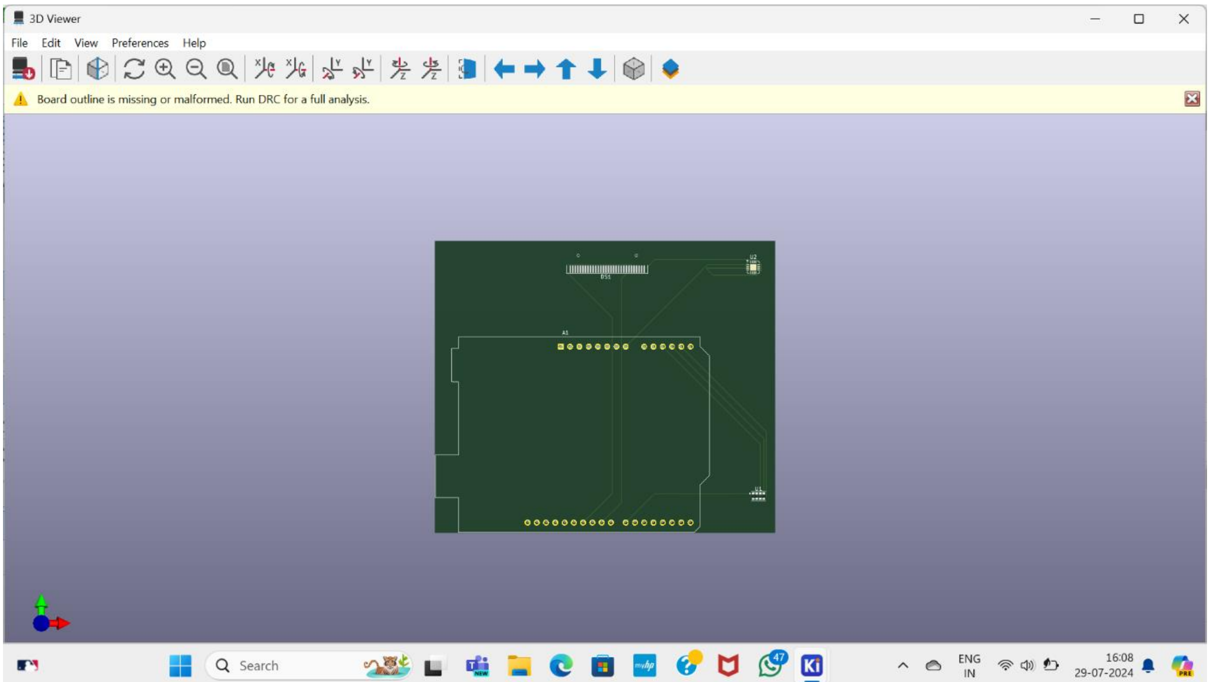

Start
Determine parameters
PWM = ON
Brake = OFF
Determine the position
Brake = ON
Determine the next winding
Is the elevator call to up?
Yes
Direction: up
Brake: OFF
Excite phase
No
Is the elevator call to down?
Yes
Direction: down
Brake: OFF
Excite phase
No
Is elevator stopped?
Yes
Direction: notr
Brake: ON
Turn off phase
No
Direction<>notr and phase switching position is OK?
Yes
Excite phase according to direction
No
Is current value over the limit?
Yes
PWM duty cycle = 0 (system is not safety)
No

WIRING DIAGRAM:



KICAD PCB DESIGN& GERBER FILE SUBMISSION:





COMPONENTS WORKING PRINCIPLES/ FUNCTIONALITY:

Arduino UNO

Arduino is an open-source platform used for building electronics projects. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board.

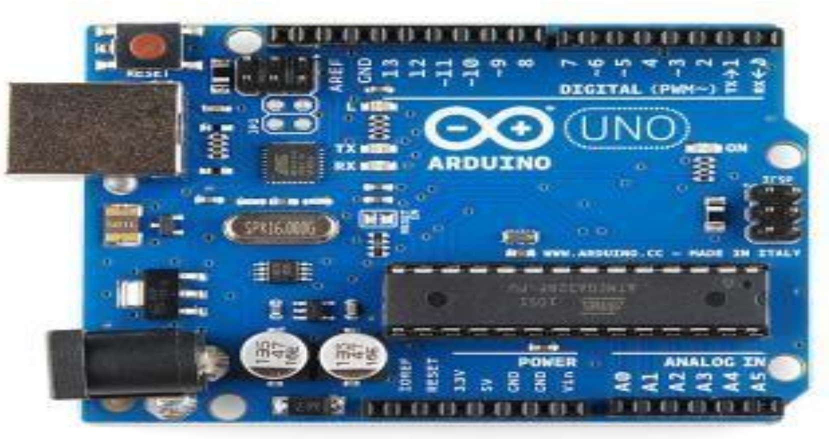


Figure 2: Arduino Board

Key Features:

- **Microcontroller:** Based on the Atmega328P chip.
- **I/O Pins:** 14 digital input/output pins, 6 analog inputs.
- **Power:** Can be powered via USB or an external power supply.
- **Easy to Use:** Simple setup, extensive community support, and a vast array of tutorials.
- **Open-Source:** Hardware and software are freely available.

Gesture Sensor APDS9960

The APDS9960 RGB & Gesture Detection Module is a small breakout board that comes with a built-in APDS-9960 sensor. The APDS9960 supports Colour detection, Gestures detection, Ambient light detection through a single module. It can detect up to six gestures which include Left, Right, Up, Down, Far and Near. The range of detection is about 10 to 20 cm. The sensor has on-chip UV and IR blocking filters, four separate diodes sensitive to different directions and I2C compatible interface for communicating with embedded controllers. The sensor also contains an interrupt pin to notify the microcontroller when there is a gesture data available for processing.

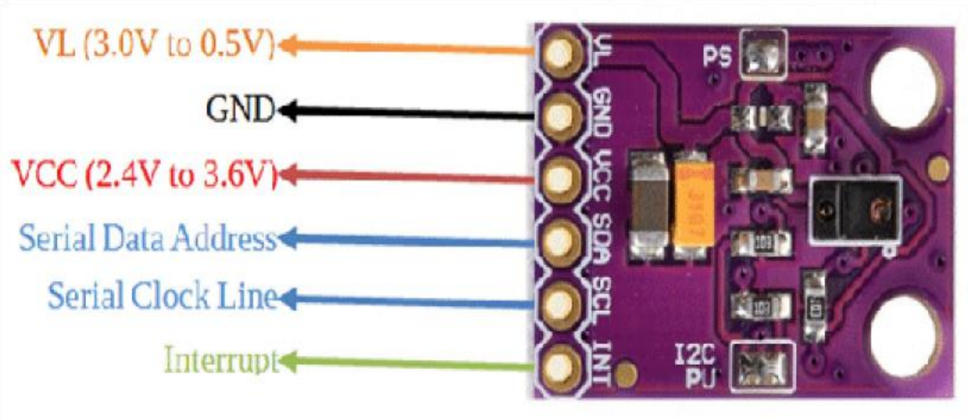


Figure 3: APDS 9960 Breakout Board

Key Features

- **Compact size:** Easy integration into various devices.
- **Low power consumption:** Ideal for battery-powered applications.
- **I2C interface:** Simple communication with microcontrollers.
- **Gesture detection:** Enables touchless control and user interaction.
- **Ambient light and color sensing:** Supports adaptive display brightness and color correction.
- **Proximity detection:** Contributes to power-saving features and user experience enhancement.

OLED Display Unit

SSD 1306 OLED display is used in this project to display various gestures performed by the user. This is a 0.96 inch Blue-Yellow OLED display module. The display module can be interfaced with any microcontroller using SPI/IIC protocols. It is having a resolution of 128x64. The package includes display board, display, 4 pin male header presoldered to board as shown in the fig. 4

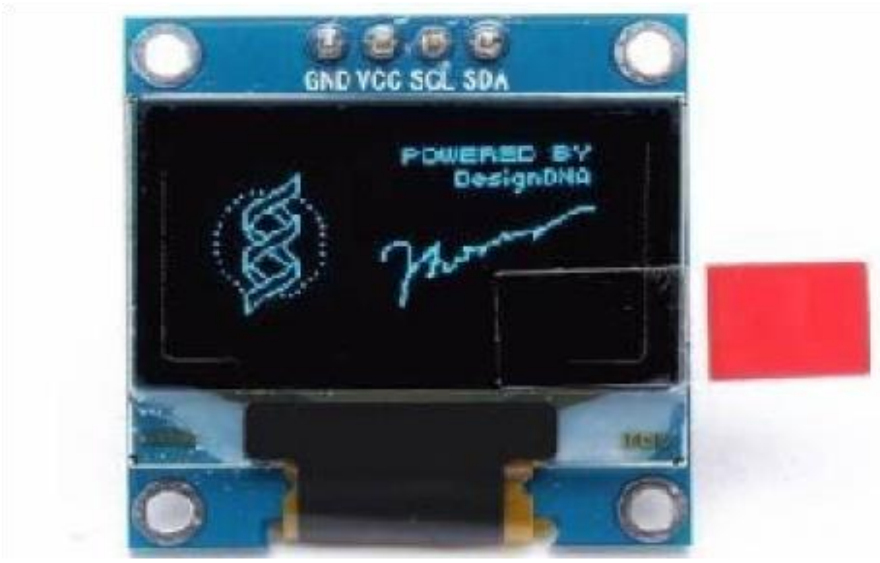


Figure 4: SSD 1306 OLED

Key Features of OLED Displays

- **Superior Contrast Ratio:** OLEDs can achieve perfect blacks by turning off individual pixels, resulting in incredibly deep contrasts and vibrant colors.
- **Wide Viewing Angles:** OLED displays offer consistent color and brightness from virtually any angle.
- **Faster Response Times:** OLED pixels can turn on and off much faster than LCDs, leading to reduced motion blur and improved image quality.
- **Thinner and Lighter:** Without the need for a backlight, OLED displays can be made thinner and lighter than their LCD counterparts.
- **Flexibility:** OLED technology allows for the creation of flexible and curved displays, opening up new design possibilities.

- Work perfectly well without the need of back light.
- 128*64 high resolution, ultra wide viewing angle
- Super low power consumption–only 0.08W when the whole screen lights up and 0.06W when displaying characters

•	Fully compatible with multiple controlling chips including Arduino and more. Support a wide range of voltage input

Bill of Materials

Component Name	Quantity	Description	Cost Approx. In INR
Arduino Pro Micro/Nano	1	For Programming	400
APDS 9960	1	Gesture Sensor	200
OLED SSD1306	1	OLED DISPLAY	300
Wires	--	For Connection	30
Total Cost			**930**

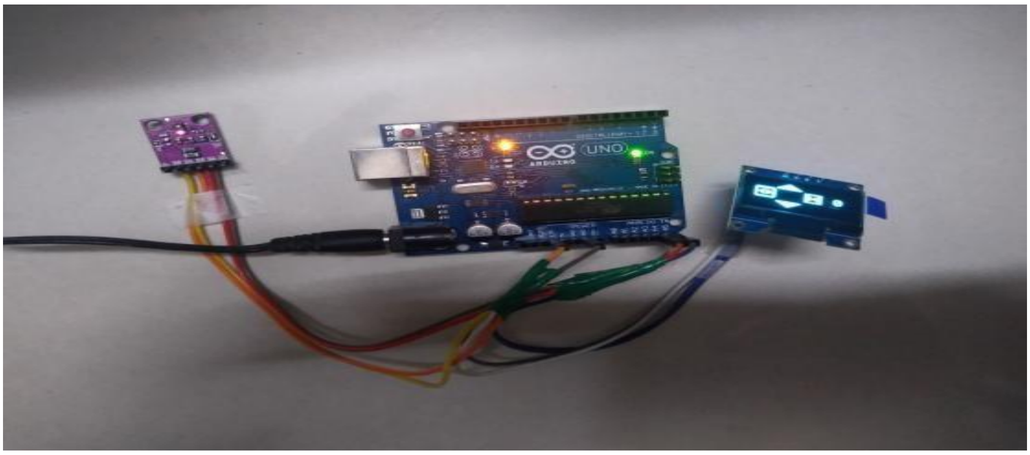
CODING

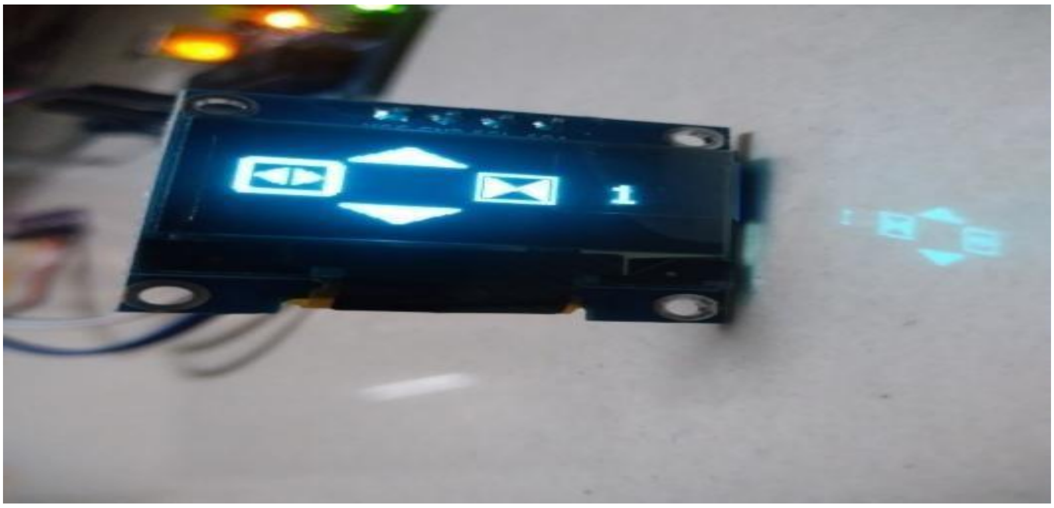
```
from machine import Pin, I2C
from ssd1306 import SSD1306_I2C
import time

# Initialize I2C and OLED display
i2c = I2C(0, scl=Pin(22), sda=Pin(21))
oled_width = 128
oled_height = 64
oled = SSD1306_I2C(oled_width, oled_height, i2c)

floornum = 0
currentfloor = 0

# Placeholder functions for gesture sensor and motor control
def init_gesture_sensor():
    pass

def gesture_available():
    # Simulate gesture availability for testing
    return True

def read_gesture():
    # Simulate reading a gesture
    return input("Enter gesture (up, down, left, right): ").strip().lower()

def liftup():
    print("Lifting up...")

def liftdown():
    print("Lifting down...")

def stop():
    print("Stopping...")

# Initialize gesture sensor
init_gesture_sensor()
print("Detecting gestures ...")

while True:
    if gesture_available():
        oled.fill(0)
        gesture = read_gesture()

        if gesture == "up":
            floornum += 1
            print("Detected UP gesture")
            oled.text("UP", 90, 17)
            oled.text("FLOOR GO=", 0, 17)
```

```
        oled.text(str(floornum), 0, 27)
        oled.show()

    elif gesture == "down":
        floornum -= 1
        print("Detected DOWN gesture")
        oled.text("DOWN", 90, 17)
        oled.text("FLOOR GO=", 0, 17)
        oled.text(str(floornum), 0, 27)
        oled.show()

    elif gesture == "left":
        print("Detected LEFT gesture")
        oled.text("OK", 90, 17)
        oled.text("FLOOR=", 10, 27)
        oled.text(str(currentfloor), 10, 37)
        oled.show()

    elif gesture == "right":
        print("Detected RIGHT gesture")
        oled.text("STOP", 90, 17)
        oled.text("FLOOR=", 10, 27)
        oled.text(str(currentfloor), 10, 37)
        stop()
        oled.show()

    if gesture == "left":
        if floornum > currentfloor:
            print("Going UP")
            liftup()
            time.sleep(2)
            currentfloor += 1
            if floornum == currentfloor:
                print("Reached")
                stop()
        elif floornum < currentfloor:
            print("Going DOWN")
            liftdown()
            time.sleep(2)
            currentfloor -= 1
            if floornum == currentfloor:
                print("Reached")
                stop()
```

OUTPUT:

Once the hardware and code are ready, c and upload the complete code. As you can see by default OLED will display the Elevator UI as shown in fig.5. Now wave off your hand up or down, to set the floor where you want to go. Then make the left gesture for acknowledging the lift to go to that floor. If you want to stop the lift, then make the right gesture with your hand as shown in fig. 6.





Gesture based user interface is an interesting and interactive user interface. It has gained lot of scope in Electronics automation. One of the many advantages of using hand gesture recognition system is that the user can interact from a distance which was not possible in the case keyboard and mouse input methods. Thus there by installing a gesture recognition system, it increases the longevity as it need not be touched. This eliminates major drawbacks like seen in touch screen kiosks where the screen gets worn out in short time period and has to be replaced often.

In this Project we are detecting only four gestures UP, Down, Left, Right to control the operations of the Elevator/Lift, but the Gesture sensor APDS9960 can also detect some other gestures like Far, Near and None which can be used to control other operations like to switch On and Off Fan in the Elevator/Lift. Apart from the gesture detection APDS9960 sensor supports Colour detection, Proximity detection and Ambient light detection through a single module, so a developer may use these features of the sensor to develop some other applications.

CONCLUSION:

The imperative for touch-free solutions in public spaces, particularly in high-traffic areas like elevators, has become increasingly evident. This design exploration delved into the development of a touch-free elevator control panel, focusing on hygiene, accessibility, reliability, user experience, security, and aesthetics.

Several technologies were evaluated, including gesture recognition, voice control, proximity sensors, and foot pedals. Each technology presented distinct advantages and challenges. Gesture recognition offered intuitive interaction but faced accuracy concerns, while voice control, though accessible, raised privacy and noise interference issues. Proximity sensors provided simplicity but were limited in functionality, and foot pedals, while hygienic, had restricted applications.

Optimal Design Approach

A combination of gesture and proximity sensors emerged as a promising approach. Gesture recognition for floor selection coupled with proximity sensors for panel activation offers a balanced solution. This combination ensures intuitive interaction, hygienic operation, and reliable performance.

Impact and Implications

The implementation of touch-free elevator control panels has the potential to significantly enhance public health by minimizing the spread of germs. Additionally, it promotes inclusivity by accommodating users with disabilities. By prioritizing user experience, these panels can contribute to a more comfortable and efficient elevator experience.

Future Directions

While this design provides a robust foundation, there is ample scope for further innovation. Integration with building management systems, incorporating biometric authentication, and developing mobile app compatibility can elevate the system's capabilities. Continuous research and development in sensor technology and artificial intelligence can further refine gesture recognition and voice control accuracy.

Conclusion

The development of a touch-free elevator control panel is a critical step towards creating healthier, more accessible, and user-centric environments. By carefully considering the interplay of technology, design, and user needs, it is possible to develop solutions that not only meet current demands but also anticipate future challenges.