IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⧖ Peer-reviewed & Refereed journal ⧖ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125338

PORTABLE VENTILATOR

Mrs. Subhashini R[1], Ms. Bhoomika K S[2], Mr. Darshan M K[3], Mr. Veeresh Rajesh Hiremath[4]

Assistant Professor, Dept. of Electronics and Communication Engineering, Maharaja Institute of Technology Thandavapura[1]

Students, Dept. of Electronics and Communication Engineering, Maharaja Institute of Technology Thandavapura[2-4]

Abstract: COVID-19 pandemic highlighted the urgent need for affordable respiratory aid systems, particularly in low-resource areas. This article introduces the design and development of an affordable, Arduino-based intelligent ventilator with the ability to monitor real-time SpO_2 and heart rate. The system makes airflow control changes in real time through fuzzy logic and sends emergency notifications through a GSM module when oxygen saturation falls below 90%. It integrates a MAX30100 pulse oximeter, R385 diaphragm pump, SIM800L GSM module, and uses a dual-cell lithium-ion battery with buck conversion for voltage regulation. It showcases how affordable components and smart control can facilitate efficient, portable respiratory assistance.

Keywords: Portable ventilator, Arduino Uno R4, SpO_2 monitoring, fuzzy logic control, GSM alert system, MAX30100, emergency healthcare, embedded systems

I. INTRODUCTION

The recent spate of global health crises, particularly the COVID-19 pandemic, has highlighted the vulnerability of healthcare systems, particularly in resource-poor settings. Among the key shortages that were highlighted was the critical shortage of mechanical ventilators, which are required to support patients with respiratory failure or distress. Traditional ventilators, although effective, are too expensive and require highly sophisticated facilities and trained staff for use. This renders them unavailable to rural and developing regions, with the consequence of preventable death due to the lack of timely respiratory therapy.

To meet this need, the design of an affordable, automated, and portable ventilator is a must. The system utilizes an Arduino Uno R4 microcontroller to control different components like the MAX30100 sensor for real-time heart rate and SpO_2 monitoring, an R385 diaphragm pump for controlled airflow, and a SIM800L GSM module for emergency notification. The system uses fuzzy logic control for smooth and variable motor speed adjustments depending on the patient's oxygen saturation.

This solution not only reduces cost but also enhances accessibility and convenience and is therefore deployable in homes, ambulances, and rural clinics. It offers a low-cost alternative to high-end ventilators by providing the necessary functionalities required to treat patients with mild to moderate respiratory distress. Its autonomous operation and real-time alerting capability also ensure that caregivers receive timely notifications in the event of critical conditions so that timely medical care can be provided.

Here, we present the design, development, and implementation of this smart portable ventilator. We outline the system architecture, hardware-software integration, fuzzy control strategy, and performance results achieved through simulation and testing. The goal is to present a realistic, affordable respiratory support device for saving lives in emergency and under-served situations.

II. PROBLEM STATEMENT AND OBJECTIVE

Problem statement:
The global shortage of mechanical ventilators during respiratory pandemics, such as the recent COVID-19 outbreak, has reiterated the constraints in the access and cost of life support. Persons with ARDS or other pulmonary diseases will need continuous ventilatory support to ensure adequate oxygenation and avoid respiratory failure. But commercial ventilator systems are costly, complicated and commonly not available in developing countries.

Objective:
The overall aim of the project is to create a low-cost, mobile, automation-based ventilator system that can be used for assisting patients with respiratory problem. The device is intended to provide continuous monitoring of the blood

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125338

oxygen saturation of arterial hemoglobin (SpO_2) and the pulse rate of the patient, with a non-invasive optical sensor. It adopts fuzzy logic control to regulate continuous airflow produced by a diaphragm pump according to SpO_2 values and keep smooth and adaptive operation of the device. If there's a drastic decrease in oxygen, the system will instantly inform the caregivers or whoever in charge with the patient via text message (SMS) through the SIM800L GSM module. Live health data can be viewed on 16x2 LCD screen continuously.

The battery-based setup is controlled by a buck converter, which provides a constant supply voltage to all devices. This provides robust operation even during emergencies or in low-resource settings where line voltage may be erratic and ensures constant operation of vital components such as the GSM-based alert system and the motorized airflow device.

III. SYSTEM DESIGN

The system design of the proposed portable ventilator consists of multiple hardware and software components that enable real-time monitoring, adaptive delivery of respiratory assistance, and emergency alerting functionality. The Arduino Uno R4 WiFi is the central processing unit of the system which connects with the MAX30100 pulse oximeter and heart rate sensor, which continually measure SpO_2 and pulse rate respectively. The continuous data stream is processed in real-time to determine patient status. When the SpO_2 drops below the normal limit, the Arduino activates a R385 diaphragm pump motor that supplies auxiliary airflow. In order to deliver oxygen in a gradual and precision manner, the implemented fuzzy logic control system was used to change the pump's speed (and thus the oxygen delivery). The pump speed was adjusted from maximum, when SpO_2 was 100%, to minimum as the SpO_2 dropped to 90%. This unique method of controlling the oxygen delivery is crucial in order to provide a smooth delivery, rather than a sudden rush of oxygen.

In critical condition management, a SIM800L GSM module is used in the system to send emergency SMS alert if the oxygen saturation drops below 90%. By sending alerts to volunteers or caregivers, a response can take place rapidly. The patient values of SpO_2, heart rate, and system status are displayed live on the 16x2 I2C LCD screen, so onsite staff can monitor the patient. The system is powered by two 7.4V lithium-ion cells, which are connected to a buck converter (LM2596), giving a constant output voltage of 4.2V for the GSM module. The pump is controlled using an IRF520 MOSFET driver receiving PWM signals from the Arduino. The modules are connected using a JUMPER wire and breadboard. Thus the system is compact, functional and good for ventilator operation both in a clinical setting and in remote/emergency situations.

IV. METHODOLOGY

The approach used to create the portable ventilator system is based on four degree of freedoms (DOF): Real-Time Monitoring, Adaptive Control, Emergency Alerting, and Power Management. Each DOF is applicable independently or collectively, and the relationship between the different DOF is built for reliable and autonomous respiratory support both in a clinical setting and in resource-poor setting.

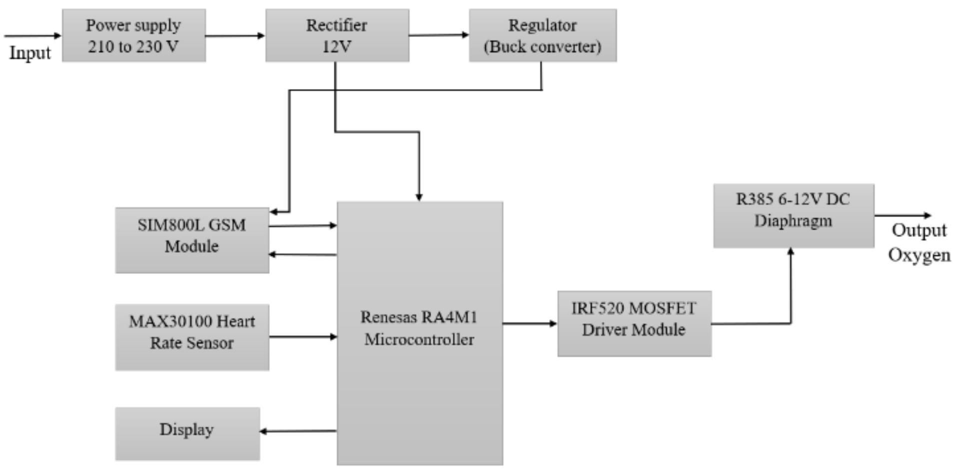

Fig.1 Block Diagram

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⨳ Peer-reviewed & Refereed journal ⨳ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125338

1. **Real-time Observation:** The system starts to run by initiating key hardware components, including a MAX30100 pulse oximeter and heart rate monitor, an LCD display, a GSM module, and a MOSFET driver. The MAX30100 pulse oximeter continuously measures the patients blood oxygen saturation (SpO_2) and pulse rate with non-invasive optical sensing and sends the values via I^2C to the Arduino Uno R4, which processes the readings and provides real-time processing health parameters. The resulting health parameters will be displayed immediately in front of the user on a 16x2 LCD.

2. **Adaptive Motor Control through Fuzzy Logic:** After accessing the real-time data, the system calculates the SpO_2 levels. If the SpO_2 level remains above 90% but is less than 100%, the system represents this as mild oxygen deficiency. The system engages the R385 diaphragm air pump and modulates the speed of the pump using a fuzzy logic-based algorithm. The motor speed is increased linearly as SpO_2 decreases from 100% to 90%, which maintains a smooth and gradual increase in flow. Linear actuation preserves the positive aspects of not having sharp motor actions that put patients at risk or discomfort patients by delaying patient's Oxygen availability. Linear actuation allows the system to be adaptive and replicate natural breathing.

3. **Emergency Alerting Using GSM:** If the SpO_2 level is less than 90%, which could indicate a potentially serious situation, the system will enter into emergency mode. This includes triggering the buzzer to alert caregivers in close proximity, as well as sending an SMS alert to set contacts using the SIM800L GSM module. The GSM module can receive AT commands from the Arduino via SoftwareSerial connection. The system will only send one SMS for an event to avoid spamming recipients with alerts. It will continue to send alerts if the SpO_2 level normalizes and becomes abnormal a second time.

4. **Power Management System:** The power requirements for the system were provided by a 7.4V dual-cell lithium-ion battery or 12v adaptor to support portability. Given the potential for battery or connector instability, we implemented a buck converter (LM2596) to convert the voltage down to a regulated voltage of 4.2V to protect sensitive components (e.g. GSM module) within the system. Adopting this approach assures stable operation, which is necessary to prepare for an emergency situation, despite prolonged runtime or unstable power conditions.

The system is powered and functions under continual monitoring so the system provides event-based input responses in real-time. This includes monitoring the system, displaying values, monitoring ventilation volumes and rates through state-change outputs, and sending alerts whenever necessary to adjust ventilator operation. This design approach allows the ventilator to maintain autonomy, efficiency, and the ability to react to unwanted physiological or environmental conditions.

V. ALGORITHM

The code acts as a real-time ventilator monitoring and control system, which measures and tracks oxygen saturation (SpO_2) and heart rate in real-time, and adjust ventilator speed as necessary. The code also sends an alert in case of emergency. Below is a step-wise flow of how the code works and functions.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

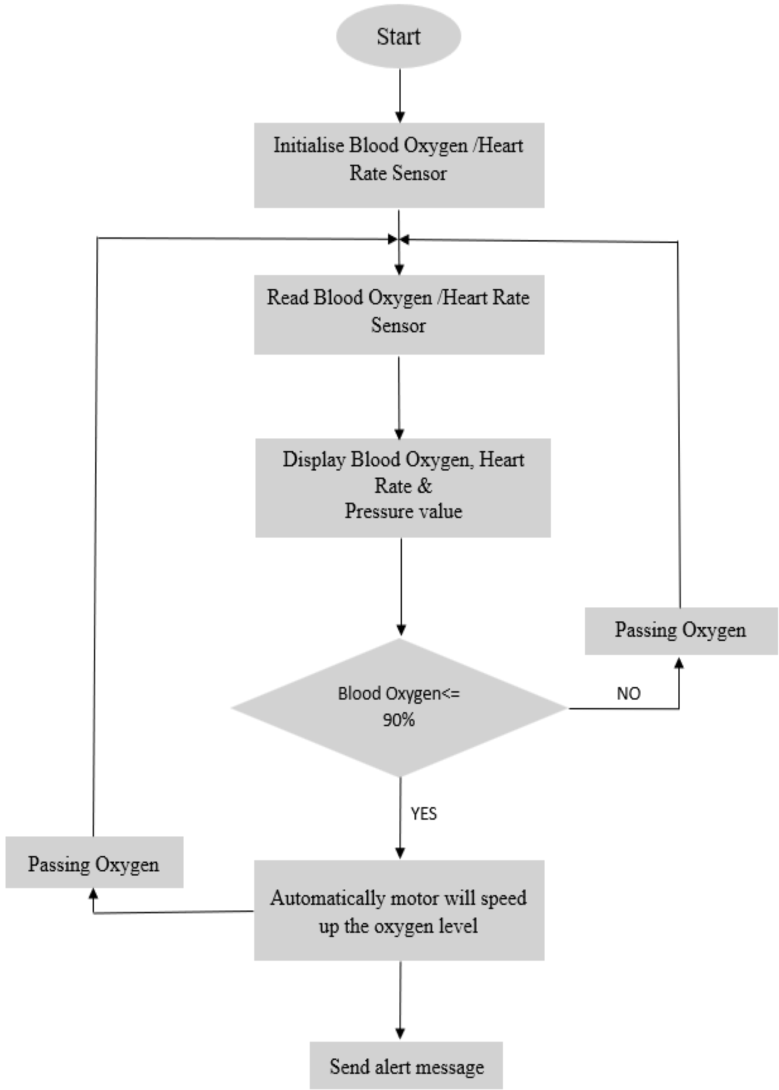

Fig. 2 Flow Chart

1. The system boots and initializes all components in accordance with process parameters.
2. The LCD is initialized and the backlight is turned on so that it is visible.
3. The MAX30100 pulse oximeter sensor is configured to detect blood oxygen (SpO_2) and the heart rate.
4. The MOSFET module is set up as an output device, which will allow it to control an adjustable ventilator motor speed.
5. The GSM module is initialized, which will enable the system to send an emergency SMS, if needed.
6. The system continuously checks the SpO_2 and heart rate values from the pulse oximeter sensor.
7. The values for SpO_2 and heart rate that were measured will be displayed on the LCD in real-time for monitoring.
8. The system continues to run in a loop to ensure that the oxygen levels are continuously monitored.
9. If the SpO_2 levels recorded are between 90% and 100%, the system will calculate the motor speed from the mapping function results.
10. The MOSFET module will then supply appropriately adjusted power to the ventilator motor based on the declared speed, either increasing or decreasing when needed.
11. When the SpO_2 level drops under 90% for more than 10 seconds, the system will prepare to send an emergency SMS alert.
12. The GSM module switches to text mode and sends a saved emergency message to a specified phone number.
13. To avoid delivery of multiple messages only one alert message is sent per low-oxygen event.
14. The system has to constantly check the SpO_2 level and update the ventilator motor speed.
15. If the SpO_2 level exceeds 90%, the normal speed of the ventilator is restored.
16. The system has to run in infinite loop to properly monitor and take the required actions when necessary.

DOI: 10.17148/IARJSET.2025.125338

VI. HARDWARE DESIGN

The ventilator system includes sensors, actuators, a microcontroller, and communication module, and power management unit to allow for real-time monitoring, adaptive oxygen control, and emergency alerting. Below are the different parts and their functions.

1.Arduino Uno R4: The Arduino Uno R4 acts as the central processing unit. It reads the information from the MAX30100 pulse oximeter sensor, parses the information through the Configuration Bluetooth app, and makes decisions based on the data with respect to SpO_2 levels, Verifies the oxygen pump speeds, and sends alerts based on input received through the SIM800L GSM communication module.

2.MAX30100 Obesity classes: The MAX30100 continuously measures SpO_2 (oxygen saturation) and heart rate. The Arduino Uno R4 is used to communicate with the oximeter sensor through the I2C protocol and receive information from the oximeter sensor. The data is received for real-time monitoring from the oximeter sensor and sent to a 16x2 I2C LCD Display.

3.R385 Diaphragm Air Pump & IRF540N MOSFET: The R385 6-12V DC Diaphragm Pump regulates oxygen supply. The pump is actuated by an IF540N MOSFET used as a switch. The Arduino Uno is programmed to transmit Pulse Width Modulation (PWM) signals to the MOSFET to control the pump's speed. The pump's speed would increase (from 0% speed to 100%) to adjust for the decreased SpO_2 levels from 100% to 90% (i.e. oxygen delivery system).

4. SIM800L GSM module: If the SpO_2 level is below 90%, an emergency SMS alert will be sent using the SIM800L GSM module. The Arduino Uno R4 communicates with the SIM800L GSM module using a serial interface (Software Serial). An alert SMS will be sent to a numbers that have been pre-defined, it includes a list of low oxygen level message.

5. 16x2 I2C LCD module: The real- time readings of SpO_2 and heart rate readings of the user are shown using a 16x2 I2C LCD Display. The animation display of messages will show system activity such as 'Initializing' or 'Motor Speed changing' with the user's SpO_2 and heart rate at the same time. In addition, any Emergency alert sent.

6. Lithium-Ion Battery & Buck Converter: The whole system is powered by a 7.4V Dual-Cell Lithium-Ion Battery. The SIM800L GSM module requires 4.2V, therefore a buck converter (DC-DC step-down module) is used to step down this voltage.

The above mentioned components were interconnected using jumper wires, PCB and/or breadboard. The Arduino Uno R4 would process the sensory input for the heart rate sensor and SpO_2 sensor, adjust the motor speed using the MOSFET, and control which display messages will be shown, while communicating with the GSM module to send alerts. It is very important to have managed all power connections and components to ensure the stable operation of the system.

VII. SOFTWARE DESIGN

The ventilator system utilizes the MAX30100 Pulse Oximeter Sensor for continuous monitoring of SpO_2 levels and heart rate. In response to SpO_2 values, the ventilator system will make adjustments to the air pump motor (via a MOSFET driver) and will send out an SMS (GSM module) message to alert caregivers (doctors) for emergencies.

1. Libraries: During code-processing, libraries are required such as Wire.h (I2C communication), LiquidCrystal_I2C.h (LCD display), MAX30100_PulseOximeter.h (MAX30100 sensor), and Software Serial.h (GSM module, soft serial).
2. Initialization: When the code is processed, the libraries and necessary components are initialized properly. This consists of: (a) configuring the required libraries, and some components (I2C interface with the LCD display and MAX30100 sensors), GSM module (software serial), configuring the GPIO pins for a MOSFET (driver) that controls the air pump; and displaying something on the LCD.
3. Loop: the or reading the SpO_2 & heart rate data from the MAX30100 sensor, and in response, adjusting air pump speed based on spit values utilizing a fuzzy logic(MOSFET control). In addition, the system has a display (LCD) that shows the real-time SpO_2 and heart rate, while sending out an ambulance alert, via the (GSM) modem, should the SpO_2 babysit levels fall below 90 percent.
4. SpO_2 and Heart Rate Monitoring: The system consistently reads SpO_2 with the MAX30100 Pulse Oximeter Sensor. When SpO_2 is 100%, the air pump will be off. As SpO_2 decreases, the speed of the pump gradually increases. Once SpO_2 is below 90%, the system will send an emergency alert via SMS.

5. GSM Module for Emergency Alerting: The SIM800L GSM module is used to send an SMS alert in the case of SpO_2 is below 90%. The system uses AT commands to set the GSM module into text mode. The system will send a text to a predetermined number to alert caregivers to low oxygen levels. Then, once the SMS is sent, the system will not send another SMS until the first one was received.

6. LCD Display for Real-Time Monitoring: The LCD display (16x2 I2C) is used to maintain an actual count of. Heart rate, in BPM (HR). SpO_2 level in %, Oxygen pump power, in % and emergency alert status when the message was sent.

7. MOSFET-Based Pump Control: The MOSFET driver controls the speed at which the air pump motor will operate based on the SpO_2 levels. The pump operations are mapped linearly between 100% SpO_2 (Off) and 90% SpO_2 (Max Speed). The signal was sent from Arduino Pin 3 to the MOSFET gate, in the form of a PWM signal.

8. MAX30100 Sensor for SpO_2 and Heart Rate: The MAX30100 Pulse Oximeter Sensor is responsible for non-invasive. The MAX30100 is a pulse oximeter, which utilizes infrared and red LEDs to monitor oxygen saturation levels in a non-invasive manner. The MAX30100 provides continuous updates of heart rate and SpO_2, or oxygen saturation, values to the system. The current of the IR LED is adjusted to be as accurate as possible.

9. Power Management: A buck converter is used to power the sensor system in a stable manner. The buck converter allows for a regulated voltage of 4.2V from a dual-cell lithium-ion battery. This provides stable power and ensures the GSM module operates correctly, without unexpected fluctuations in voltage.

VIII. RESULT

The system initializes components, including the LCD, MAX30100 sensor, MOSFET module, and GSM module.

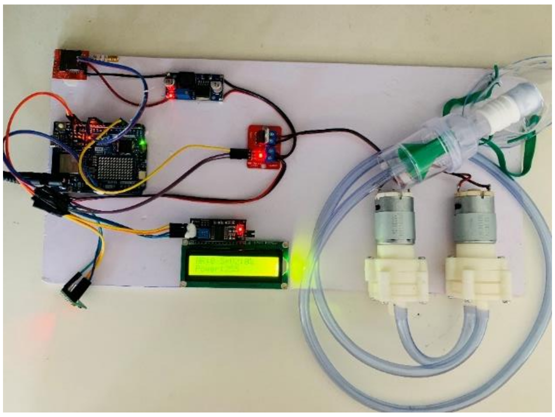

Fig. 3 System initializes components.

It continuously monitors SpO_2 and heart rate, displaying values on the LCD and adjusting the ventilator speed based on SpO_2 levels.

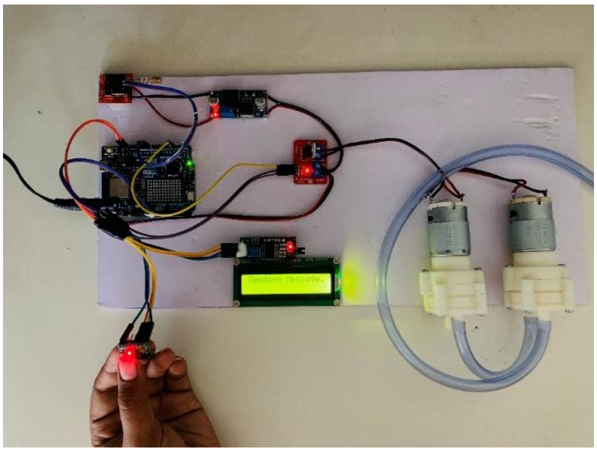

Fig. 4 Continuously monitors SpO_2 and heart rate.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⪤ Peer-reviewed & Refereed journal ⪤ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125338

If SpO_2 drops below 90% for over 10 seconds, an emergency SMS alert is sent via the GSM module. The system runs in a loop, ensuring real-time monitoring and adaptive ventilator control.

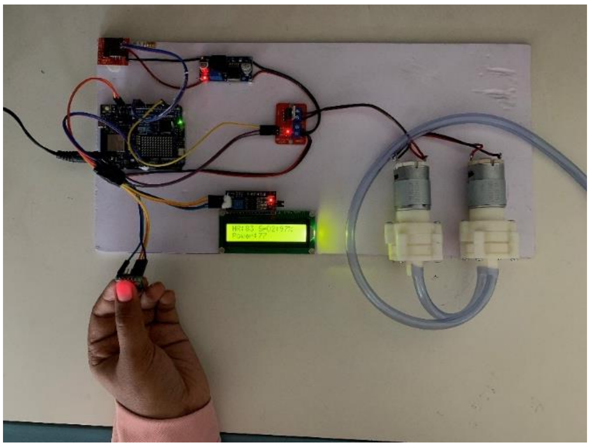

Fig. 5.3 Sending alert SMS.

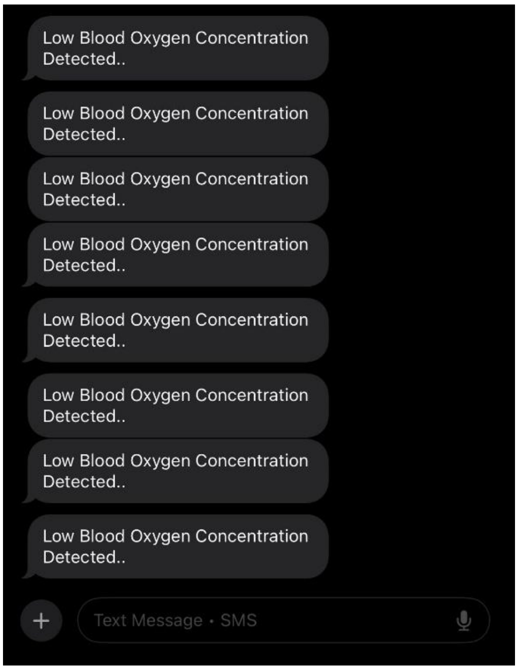

Fig. 5.4 Received alert SMS.

IX. FUTURE ENHANCEMENTS

1. Integration of Advanced Ventilation Modes- The system in its future versions will support advanced ventilator modes such as pressure control, volume control, and BiPAP/CPAP modes, which will make the ventilator appropriate for patients with severe respiratory disease and allow the use of ICUs.
2. Redundancy and Increased Accuracy of Sensors- We can switch to more accurate and more reliable biomedical sensors (such as the MAX30102) or medical-grade pulse oximeters, which will minimize the effects of motion artifacts and environmental noise. We can also include redundant sensors to make it more reliable by verifying critical data.
3. Solar Charging and Battery Health Monitor Systems- Future versions can include a battery management system (BMS) to monitor battery health and discharge and charge cycle counts. Solar charging would extend the system's usability in rural or off-grid areas.
4. Customizable and modular- a modular design would allow for serviceability and upgradability for example to replace sensors or to other power sources, allowing flexibility to support different interventions or environments.

International Advanced Research Journal in Science, Engineering and Technology
Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025
DOI: 10.17148/IARJSET.2025.125338

X. CONCLUSION

The low-cost ventilator system provides a simple yet effective solution for automated respiratory support. By integrating SpO2 monitoring, fuzzy logic ventilator assistance, and GSM-based emergency alerts, the system creates improved patient care, along with reduced reliance on valuable hospital ventilators. Its cost-effectiveness, portability, and automation make it most desirable for health care facilities in developing world locations and emergency situations. Future improvements, IoT connectivity and AI-style analytics, could enhance functionality and adaptability.

REFERENCES

[1]. S. Q. Bless, J. K. Abed, and M. J. Mnati, “Developing a portable smart ventilator: A prototype that balances cost and functionality,” in Proc. 7th Int. Symp. on Innovative Approaches in Smart Technologies (ISAS), 2023, doi: 10.1109/ISAS60782.2023.10391355.
[2]. R. B. Kale and S. K. Singh, “DIY ventilator using Arduino with blood oxygen sensing for COVID pandemic,” Int. Res. J. Eng. Technol. (IRJET), vol. 9, no. 7, pp. 1616–1619, Jul. 2022.
[3]. P. Sivaranjani, S. Sasikala, S. K. Soumiya, M. S. Subhashree, and S. R. Suriya, “IoT based smart ventilator for automatic oxygen flow,” in Proc. 7th Int. Conf. Trends in Electronics and Informatics (ICOEI), 2023, pp. 463–471, doi: 10.1109/ICOEI56765.2023.10125803.
[4]. R. Raman and A. Sapatnekar, “Smart ventilators in intensive care: IoT-connected respiratory support,” in Proc. IEEE Int. Conf. System, Computation, Automation and Networking (ICSCAN), Puducherry, India, 2023, pp. 1–6, doi: 10.1109/ICSCAN58655.2023.10395430
[5]. Thitirattanapong, A., Ongtrakul, S., & Pintavirooj, C. (2021). Low-Cost Blower-Based Ventilator (pp. 1–3).
[6]. Awad, M. E., El-Garhy, A. M., Eldosoky, M. A., & Soliman, A. M. (2021). Modeling and Simulation of Respiratory System for Acute Respiratory Distress Syndrome (ARDS) Associated with COVID-19 (pp. 232–242).
[7]. Low Cost Portable Ventilator for COVID-19.
[8]. Thitirattanapong, A., Ongtrakul, S., & Pintavirooj, C. (2021). Low-Cost Blower-Based Ventilator, 2021 13th Biomedical Engineering International Conference (BMEiCON), (pp. 1–3).
[9]. Jadav, V., Machhi, N., Biawat, D., Das, R., & Mali, J. (2022). Review for Arduino-Based Portable Ventilator for COVID-19. International Journal of Creative Research Thoughts (IJCRT), 10, d214–d217.
[10]. Instrumentation and Telecommunication Engineering. Low Cost Portable Ventilator for COVID-19 (pp. 1–14).
[11]. Corey, R. M., Widloski, E. M., Null, D., Ricconi, B., Johnson, M. A., White, K. C., et al. (2020). Low-complexity system and algorithm for an emergency ventilator sensor and alarm. IEEE Transactions on Biomedical Circuits and Systems, 14, 1088–1096.
[12]. Christou, A., Ntagios, M., Hart, A., & Dahiya, R. (2020). GlasVent—The Rapidly Deployable Emergency Ventilator. Global Challenges, 4, 2000046.
[13]. Srinivasan, S., Ramadi, K. B., Vicario, F., Gwynne, D., Hayward, A., Langer, R., et al. (2020). Individualized system for augmenting ventilator efficacy (iSAVE): A rapidly deployable system to expand ventilator capacity. BioRxiv, 2020.03.28.012617.
[14]. Kordulasiński, R., Królewska, M., Głowacz, B., Mikowska, L., Olejniczak, Z., Pałasz, T., et al. (2020). A Novel System for Patient Ventilation and Dosing of Hyperpolarized 3He for Magnetic Resonance Imaging of Human Lungs. Applied Sciences, 10, 3163.
[15]. Gusti, V., Wu, W. J., Grover, A., Chiu, S., Su, K.-W., Ma, E., et al. (2020). The COSMIC Bubble Helmet: A Non-Invasive Positive Pressure Ventilation System for COVID-19. IEEE Open Journal of Engineering in Medicine and Biology, 1, 312–315.
[16]. Chharia, A., Chauhan, S., Basak, S., & Sharma, B. (2021). NeT Vent: Low-Cost, Rapidly Scalable and IoT-Enabled Smart Invasive Mechanical Ventilator with Adaptive Control to Reduce Incidences of Pulmonary Barotrauma in SARS-CoV-2 Patients (pp. 1–10).
[17]. Pristianto, E. J., Kurniadi, D. P., Arisesa, H., Kurniawan, D., Kurniawan, E. D., Wijayanto, Y. N., et al. (2021). High-Flow and High-Pressure Oxygen Mixing for Ventilator System (pp. 201–204).
[18]. "Arduino Uno." Available at: https://andprof.com/tools/what-is-arduino-ide-and-how-use-it/
[19]. "MAX30100 Pulse Oximeter and Heart Rate Sensor." Available at: https://lastminuteengineers.com/max30100-pulse-oximeter-heart-rate-sensor-arduino-tutorial/
[20]. "MLX90614 IR Temperature Sensor." Available at: https://www.apogeeweb.net/circuitry/MLX90614-temperature-sensor.html
[21]. "HX710B Air Pressure Sensor." Available at: https://www.homemade-circuits.com/hx710b-air-pressure-sensor-datasheet-how-to-connect/