DEVELOPMENT OF IOT BASED CONTINUOUS POSITIVE AIRWAY PRESSURE (CPAP) SYSTEM

A PROJECT REPORT

Submitted by

THAARANI S 211419105152

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

ELECTRICAL AND ELECTRONICS ENGINEERING

PANIMALAR ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

APRIL 2023

PANIMALAR ENGINEERING COLLEGE

(An Autonmous Institution, Affiliated to Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report "**DEVELOPMENT OF IOT BASED CONTINUOUS POSITIVE AIRWAY PRESSURE (CPAP) SYSTEM"** is the bonafide work of **THAARANI S (211419105152)** who carried out the project work under my supervision.

Signature of the HOD with date	Signature of the Supervisor with date
DR.S.SELVI, M.E., Ph.D.,	**MRS.S.PUSHPA**
Professor and Head	**Assistant Professor**
Department of Electrical and	Department of Electrical and
Electronics Engineering	Electronics Engineering
Panimalar Engineering College	Panimalar Engineering College
Chennai – 123	Chennai – 123

INTERNAL EXAMINER **EXTERNAL EXAMINER**

BHARAT HEAVY ELECTRICALS LIMITED

(A Government of India Undertaking)

BOILER AUXILIARIES PLANT

RANIPET 632 406 TAMIL NADU INDIA

HUMAN RESOURCE DEVELOPMENT CENTRE

Certificate

Shri/Ms. **THAARANI S** *Student of* **B.E (Electrical & Electronics) PANIMALAR ENGINEERING COLLEGE, CHENNAI** *undergone a Project Training on the title* **DEVELOPMENT OF IOT BASED CONTINUOUS POSITIVE AIRWAY PRESSURE (CPAP) SYSTEM** *from* **02-01-2023** *to* **30-01-2023** *(Inclusive of both days) at Boiler Auxiliaries Plant, BHEL, Ranipet - 632 406, Tamil Nadu.*

Date:

HUMAN RESOURCE DEVELOPMENT CENTRE ★ BHEL/BAP/RANIPET-6 ★

30 JAN 2023

D. Ponsankar

DY. MANAGER / HRDC

D. PONSANKAR
Dy. MANAGER / HRM - HRDC
BAP/BHEL/RANIPET-632 406

ACKNOWLEDGEMENT

My sincere thanks to the Honourable Founder and Chairman, **Dr.JEPPIAAR,M.A.,B.L.,Ph.D.,** for his sincere endeavour in educating us in his premier institution.

I would like to express my deep gratitude to the beloved **Secretary and Correspondent, Dr.P.CHINNADURAI, M.A.,M.Phil.,Ph.D** for his enthusiastic motivation which inspired me a lot in completing this project and my sincere thanks to our Directors **Mrs.C.VIJAYA RAJESWARI, Dr.C.SAKTHI KUMAR, M.E.,Ph.D** and **Dr.SARANYASREE SAKTHIKUMAR,B.E,M.B.A,Ph.D** for providing me with the necessary facilities for the completion of this project.

I would like to express my sincere thanks to the Principal, **Dr. K. Mani M.E., Ph.D.,** for having extended his guidance and cooperation.

I would also like to thank the **Head of the Department**, **Dr.S. Selvi, M.E., Ph.D., Professor and Head, Department of Electrical and Electronics Enginering** for her encouragement.

Personally, I thank my Guide **Mrs.S.Pushpa, Assistant Professor in Department of Electrical and Electronics engineering** for the persistent motivation and support for this project, who at all times was the mentor of germination of the project from a small idea.

I express my sincere thanks to the project coordinators **Dr.S.Deepa & Dr.N.MANOJ KUMAR,M.E.,Ph.D., in Department of Electrical and Electronics Engineering** for the Valuable suggestions from time to time at every stage of my project.

Finally, I would like to take this opportunity to thank my family members, faculty and non-teaching staff members of our department, friends, well-wishers who have helped me for the successful completion of my project.

ABSTRACT

The aim of this project is to clear the throat and nasal passage for the subject to get rid of sleep apnea and snoring problems which is harmful on a long run. In addition, we implement real time monitoring and storing of the parameters monitored. Continuous positive airway pressure therapy (CPAP) is a treatment of choice to help a person who has obstructive sleep apnea (OSA) to breathe more easily during sleep. A CPAP machine increases air pressure in the throat so that the airway does not collapse when a person breathes in. Obstructive sleep apnea (OSA) is one of the main underdiagnosed sleep disorder where the upper airways collapse intermittently during sleep causing cessations in breathing. It is an aggravating factor for several serious cardiovascular diseases, including stroke. A lot of people die during sleep because of uneven body changes in the body during sleep. Treatment for sleep apnea are mostly clinical based and are not suitable for normal people to affort. In this project, we implement a compact, portable and cost-effective micro-controller based sleep apnea monitor with Respiration sensor for monitoring the breath condition, SPO2 and Pulse Sensor for monitoring the heart rate and blood oxygen saturation level. All the parameters are monitored by the microcontroller, once it detects any difficulties in breathing condition, it automatically turns on the supportive system for the patient to breathe stably. The data are continuously monitored and stored on the ThingSpeak platform using IoT to know the recovery.

TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE NO.
	ACKNOWLEDGEMENT	**iv**
	ABSTRACT	**v**
	TABLE OF CONTENTS	**vi**
	LIST OF TABLES	**ix**
	LIST OF FIGURES	**x**
	LIST OF ABBREVIATIONS	**xii**
1	**INTRODUCTION**	**1**
	1.1 Sleep Apnea	1
	1.1.1 Causes	2
	1.2 CPAP	3
2	**LITERATURE REVIEW**	**4**
3	**SYSTEM ANALYSIS**	**8**
	3.1 Existing System	8
	3.1.1 Drawbacks	8
	3.2 Proposed System	9
	3.2.1 Advantages	9
	3.2.2 Application	9
4	**SYSTEM DESCRIPTION**	**10**
	4.1 Hardware Requirements	10
	4.2 Software Requirements	10
	4.3 Block Diagram of the proposed system	11
	4.3.1 Block Diagram Description	12
	4.3.2 Flow chart for auto mode	13
	4.3.3 Schematic Diagram	14

5	**MODULES ELUCIDATION**	**15**
	5.1 Power Supply	15
	5.1.1 Transformer	15
	5.1.2 IC voltage regulator	16
	5.2 Arduino UNO	17
	5.2.1 Power supply of Arduino	17
	5.2.2 ATmega328p	19
	5.2.3 Communication	21
	5.3 Node MCU	21
	5.3.1 Node MCU GPIO Layout	23
	5.4 Respiration sensor	24
	5.5 MAX30100	25
	5.5.1 Benefits and features	25
	5.5.2 Pin configuration	26
	5.5.3 System block diagram	27
	5.6 LCD Display	28
	5.6.1 Power supply of LCD	29
	5.6.2 Address counter	29
	5.7 Relay	30
	5.7.1 Module pin description	30
	5.8 Centrifugal compressor	32
	5.8.1 Important elements	33
	5.9 CPAP Mask	34
	5.10 Push Button	35
6	**SOFTWARE DESCRIPTION**	**36**
	6.1 ThingSpeak platform	36
	6.1.1 IoT	37
	6.2 Arduino IDE	38

7	**RESULT AND DISCUSSION**	**39**
	7.1 Final Module	40
	7.2 Output	41
	7.3 Software output	43
	7.3.1 Mobile view	44
8	**CONCLUSION**	**46**
	FUTURE SCOPE	**47**
	APPENDIX	**48**
	REFERENCES	**55**

LIST OF TABLES

TABLE No.	TITLE	PAGE No.
4.1	Hardware Requirements	10
5.1	Relay pin description	30

LIST OF FIGURES

FIGURE No.	TITLE	PAGE No.
1.1	Obstructive sleep apnea	2
4.1	Block diagram of the proposed work	11
4.2	Flow chart for auto mode	13
4.3	Schematic diagram	14
5.1	Block diagram for power supply	15
5.2	Step-down Transformer	16
5.3	IC voltage regulator	16
5.4	Arduino UNO	18
5.5	Node MCU	22
5.6	Node MCU GPIO Layout	23
5.7	Respiration sensor	24
5.8	MAX30100	25
5.9	Pin configuration of MAX30100	26
5.10	System block diagram	27
5.11	Lcd display	28
5.12	Single- Channel Relay	30
5.13	Compressor Motor	32
5.14	CPAP Mask	34
5.15	Push Button	35
6.1	ThingSpeak	36
6.2	ThingSpeak IoT	37
6.3	Arduino IDE	38
7.1	Final Module	40

7.2	Manual Mode	41
7.3	Airflow ON	41
7.4	Auto Mode	41
7.5	Snoring Event < 2	41
7.6	Stable Condition	41
7.7	Snoring Event > 2	41
7.8	Snoring Detected	41
7.9	Airflow ON	41
7.10	Software Output	43
7.11	Heart Rate in Mobile view	44
7.12	Spo2 level in Mobile view	44
7.13	Compressor status in Mobile view	45

LIST OF ABBREVIATIONS

CPAP	Continuous Positive Airway Pressure
SPO2	Saturation of Peripheral oxygen
IoT	Internet of Things
OSA	Obstructive Sleep Apnea
CSA	Central Sleep Apnea
SIDS	Sudden Infant Death Syndrome
PPG	Photoplethysmogram
MoT	Medical of Things
PID	Proportional Integral Derivative
COPD	Chronic Obstructive Pulmonary Disease
IDE	Integrated Development Environment
FIFO	First In First Out
LED	Light Emitting Diode
LCD	Liquid Crystal Display
EEPROM	Electrically Erasable Programmable Read Only Memory
SRAM	Static Random Access Memory
RAM	Random Access Memory
ROM	Read Only Memory
AVR	Aortic Valve Replacement

RISC	Reduced Instruction Set Computer
GPIO	General Purpose Input Output
BiPAP	Bilevel Positive Airway Pressure

CHAPTER 1

INTRODUCTION

1.1 SLEEP APNEA

Sleep apnea is a disorder which is parallelly connected to the human respiratory system and our brain. It indicates a problem with breathing and respiration obstacles during sleep. There are two kinds of sleep apnea: obstructive sleep apnea (OSA) and central sleep apnea (CSA). Sometimes they both take place at the same time. That can be said as complex sleep apnea, but it is too rare to be called an apnea type. Obstructive sleep apnea (OSA) is basically upper airway congestion. It occurs due to relaxation of our throat muscles and lack of oxygen passing through our nasal-throat passage that causes irritation in breathing and sometimes results in serious complications. Central sleep apnea is less popular but more alarming than OSA. The human brain is the key maintenance center of the whole body, as every organ and system runs on its directed signal or instruction, and its central sleep apnea comes into major exposure. Central sleep apnea occurs when the brain fails to send the necessary instructions or signals to the system which controls our breathing and respiration. Here, neurons fail to transmit signals to the breathing muscle, which pauses the breath for a good amount of time, maybe near 10 seconds . And in rare cases, they can both take place, which is more than an emergency medical treatment issue, and alarming also.

The symptoms are as follows:

- Loud snoring
- Episodes in which you stop breathing during sleep — which would be reported by another person
- Gasping for air during sleep
- Excessive daytime sleepiness (hypersomnia)

1.1.1 CAUSES

As shown in Fig.1.1 , Sleep Apnea occurs when the muscles in the back of your throat relax. These muscles support the soft palate, the triangular piece of tissue hanging from the soft palate (uvula), the tonsils, the side walls of the throat and the tongue. When the muscles relax, your airway narrows or closes as you breathe in. You can't get enough air, which can lower the oxygen level in your blood. Your brain senses your inability to breathe and briefly rouses you from sleep so that you can reopen your airway. This awakening is usually so brief that you don't remember it. You might snort, choke or gasp. This pattern can repeat itself five to 30 times or more each hour, all night, impairing your ability to reach the deep, restful phases of sleep.

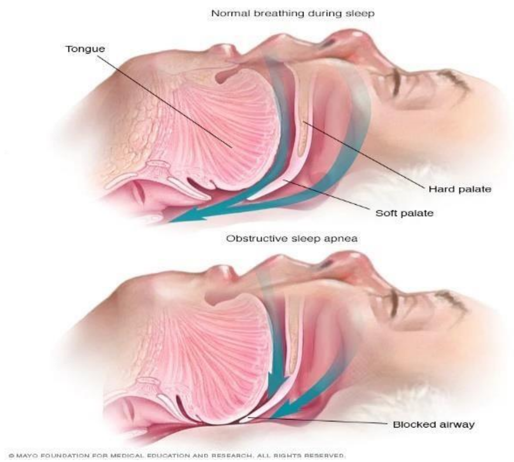

Fig. 1.1 Obstructive Sleep Apnea

About 1 billion people around World-wide suffer from sleep apnea disorder. The prevalence of obstructive sleep apnea disorder is 2.4% to 4.96% in men and 1% to 2 % in women in the Indian sub-continent. The general factors that cause sleep disorders are physical, medical, environmental and psychiatric problems and other factors such as genetics, night shift duty, medicines, and aging are the major causes of the sleep disorder. The risk factors associated with sleep apnea are the acute stroke, neurological problems, congestive heart failure, etc. This sleep apnea disorder may be lethal if it is left untreated.

1.2 CONTINUOUS POSITIVE AIRWAY PRESSURE (CPAP)

Continuous positive airway pressure (CPAP) is the gold standard of care for mild to extreme OSA patients. A CPAP machine increases air pressure in the throat so that the airway does not collapse when a person breathes in. Good pressure stability and pressure reduction of CPAP therapy are required for easy sleeping. During an obstructive sleep apnea episode, the diaphragm and chest muscles work harder to open the obstructed airway and pull air into the lungs. Breathing usually resumes with a loud gasp, snort, or body jerk. These episodes can interfere with sound sleep. They can also reduce the flow of oxygen to vital organs and cause irregular heart rhythms. Here, experimental CPAP machine using an application CPAP blower (motor) assembly and a microcontroller are designed with the objectives to help patients to have easy accessibility both in terms of price and usability (user-friendly).

The proposed work for measures and monitors the breath condition, heart rate and blood oxygen saturation level. All the parameters are monitored by the microcontroller, once it detects any difficulties in breathing conditions it'll automatically turn on the supportive system for the patient. By using the proposed supportive system patients would be able to breathe stably. The data are continuously monitored and stored on the cloud platform to know the recovery.

CHAPTER 2

LITERATURE REVIEW

Salvatore Andrea Pullono, Ifana Mahbub, Maria Giovanna Bianco, Samira Shamsir, Syed Kamrul Islam, Mark S.Gaylord, Vichien Lorch, Antonino S.Fiorilla (2017), [1] as discussed on Medical Devices for Pediatric Apnea Monitoring and Therapy: Past and New Trends. This paper presents a review of the state of the art of clinical technologies and methodologies for sleep apnea detection and their pros and cons, with particular focus on their working principles and relevance to pediatrics. The point-of-care diagnosis of this sleep disorder are focused on the design of integrated devices for less complex and noninvasive monitoring and envisioned to be integral parts of the daily home-based applications is included in the paper. Despite the fact that it is possible to monitor infants at home, most caregivers are not professionally trained and in the rare cases in which they are, clinical studies have shown that monitors cannot be used to detect adverse collateral events such as sudden infant death syndrome (SIDS).

Remo Lazazzera, Margot Deviaen Carolina Varon , Bertien Buyse, Dries Testelmans, Pablo Laguna , Eduardo Gil , Guy Carrault (2021), [2] discussed on Detection and Classification of Sleep Apnea and Hypopnea Using PPG and SPO2 Signals.The system proposed in this work is designed for feasible ambulatory monitoring, based on oximetry devices providing both SpO2 and PPG signals. The presented detection tool, exploiting PPG and SpO2 signals, can be used as first screening for apnea/hypopnea monitoring. This can validate the possibility to implement the method in a non-invasive, home sleep monitoring device. The results suggest that such approach can be used as an initial report to better select patients at home, before clinical sleep center observation. This perspective, lights the way for an embedded system for sleep apnea/hypopnea home monitoring, in a Medical of Things (MoT) device.

Gr´egoire Surrel , Amir Aminifa , Francisco Rinc´on, Srinivasan Murali, David Atienza (2018) ,[3] as discussed on Online Obstructive Sleep Apnea Detection on Medical Wearable Sensors. In this paper, we propose a accurate, and energy efficient system for monitoring obstructive sleep apnea on a long-term basis. It is designed as an online ultra-low power wearable obstructive sleep apnea monitoring system. It concludes that using our proposed patient-specific technique, it is possible to achieve high classification accuracy for OSA detectionIt is based on monitoring the patient using a single-channel electrocardiogram signal. Bluetooth link, this wearable sensor can upload its analysis to an online webservice for a continuous monitoring, tracking the evolution of the disease - Shorter distance.

Vishal V Kurup, Rahul K R, Rahul G Kartha, Elna T Thampi, Gylson Thomas, Ajish P J (2021), [4] as discussed on Low-Cost Portable Emergency Mechanical Ventilator Proposed for Covid-19 Patients. We present the design and validation of a simple, compact, and low-cost ventilator with features like advanced closed loop feedback control for the intelligent adult respiratory circuit .It involves the design and making of a prototype that will provide oxygen-regulated, volume and pressure controlled air for mechanical ventilation with sensors providing feedback signals for a closed loop control. Using this technology, we can simply manage the ventilator, making it more pleasant for elderly and handicapped folks. Manual ventilation is a short-term solution in critical care situations. Respiratory assistance may be required in a variety of situations: patients may be awake or asleep, sedated or drugged and paralysed.

Athra'a sabeeh Mikha, Hadeel K.Aljobouri (2021) , [5] as discussed on A Simplified Design of CPAP Device Construction by using Arduino NANO for OSA Patients. This study aims to build an easy hardware design for the CPAP experimental device of a PID controlling Brushless DC motor using an Arduino Nano board to be used for educational purposes. The safety is achieved via the increment of the pressure at the inhalation event and decrease it at the exhalation event in order to treat the OSA patients. The results of tests show that the CPAP controller has good reliability and stability with less pressure error. This prototype CPAP design might be motivated by adding another pressure sensor for expiration event setting mode . Also, an oximeter sensor inserts to check up patient O2 level.

Joey Reinders , Bram Hunnekens Frank Heck, Tom Oomen , Nathan van de Wouw (2021), [6] as discussed on Adaptive Control for Mechanical Ventilation for Improved Pressure Support. The main contribution of this article is the design of a control strategy for mechanical ventilation, which ensures the exact tracking of the airway pressure independent of the patient, hose, leakage, patient effort, and set point. Key advantages of the proposed approach include that it allows for a fast and accurate pressure response, even for large lungs and big leaks, prevents overshoot in the patient flow.Furthermore, accurate pressure tracking results in better patient-ventilator synchrony.A drawback of this approach is that it is limited to repeated sequences of the set point and initial conditions. Therefore, the performance of the iterative learning control framework proposed in degrades when patients are breathing spontaneously.

Udaya Dampage, Malmindi Ariyasinghe, Samanthi ,Pulleperuma (2021),[7] as discussed on An Automated Jet Nebulizer with Dynamic Flow Regulation. The proposed nebulizer is composed of two modes as “Compressed Air” mode and the “Oxygen Therapy” mode. The automated triggering from one mode to another will be dependent upon the percentage of oxygen saturation of the patient, monitored from the SpO2 sensor. Oxygen therapy mode identifies the patient’s desaturation and important where the patient can be already hypoxic or have a ventilation-perfusion mismatch. Disadvantageous in severe COPD patients where this can increase the partial pressure of carbon dioxide and exacerbate the respiratory failure.

Santhoshini Arulvallal, Snekhalatha U, Rajalakshmi.T (2019) , [8] as discussed on Design and Development of Wearable Device for Continuous Monitoring of Sleep apnea Disorder. In this proposed study, wearable continuous sleep apnea monitoring system was developed which can measure spo2, heart rate and blood pressure by cuffless means in real time. The observational values obtained through the device shows high compactness with polysomnography results. This brought the capability of sleep apnea monitoring to be used as a wrist-based wearable in a cost-effective, powerefficient manner. The developed smart watch extents the screening capacity of sleep apnea syndrome which can be used for the further treatment process. It involves the complete and continuous monitoring of the certain parameters and body functions of the patient. Real-time monitoring cannot be viewed through Bluetooth or wi-fi.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

Treatment for sleep apnea are mostly clinical based and the existing techniques either detect sleep apnea events or provide solution manually. The vast majority of the gadgets to treat sleep apnea patients are massive and are not practical for typical individuals. There is no mindfulness about sleep apnea to the greater part of individuals. Cost of the existing systems are too high for normal people to afford. The existing systems to treat sleep apnea patients in health care units does not have a controlled air pressure at every time of instance, the pressurized air is supplied with the help of care takers manually.

3.1.1 DRAWBACKS

- High Cost
- Not Portable
- Less Awareness
- Manual Operation

3.2 PROPOSED SYSTEM

In this project we have focused on the problem encountered by OSI patients. In our system the pricing is less than 1/3rd the amount of the actual one. Portable in size. The data are continuously monitored and stored on the cloud platform to know the recovery. This proposed system consists of a Respiration sensor for measuring and monitoring the breath condition, SPO2 and Pulse Sensor for monitoring the heart rate and blood oxygen saturation level. All the parameters are monitored by the microcontroller once the microcontroller detects any difficulties in breathing conditions it will automatically turn on the supportive system for the patient. By using our proposed supportive system patients can able to breathe stably.

3.2.1 ADVANTAGES

1. Cost effective
2. Portable
3. Dual mode of operation - is auto mode and timer mode or manual mode, smart mode offers decent life for the machine.
4. Real-time monitoring of breath condition, heart rate and SPO2 level

3.2.2 APPLICATIONS

1. It could be used for people with unhealthy snoring.
2. It could be used for people with sleep apnea related problems.
3. It could be used for people with asthma for passage clearing at PoC level.
4. It could be used Neonatal Intensive Care Units.
5. It could be used as instant solution for COVID-19 patients.

CHAPTER 4

SYSTEM DESCRIPTION

4.1 HARDWARE REQUIREMENTS

The hardware requirements for the proposed system are shown in Table.4.1.

Table.4.1 Hardware requirements

S.No.	Component Name	Quantity
1.	Arduino UNO	1
2.	Node MCU	1
3.	Liquid Crystal Display	1
4.	Respiration Sensor	1
5.	MAX30100 Sensor	1
6.	Driver Unit	1
7.	Airflow Unit	1
8.	Mask	1
9.	12v/1A Power Supply	1
10.	Power Chord	1

4.2 SOFTWARE REQUIREMENTS

- Language : Embedded C
- Software : Arduino IDE
- Cloud platform : ThingSpeak

4.3 BLOCK DIAGRAM

The block diagram for the proposed work is shown below

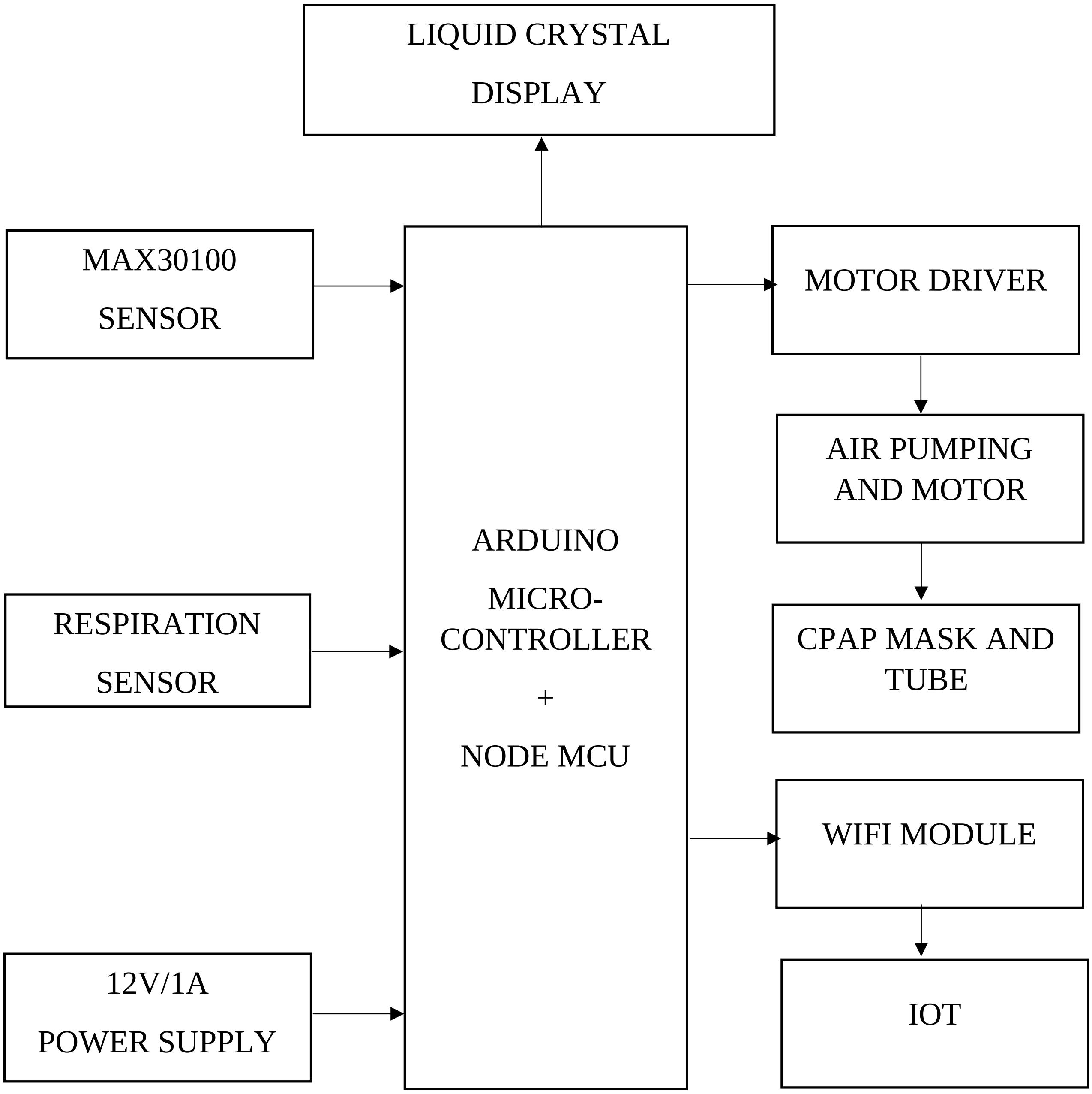

Fig. 4.1 Block Diagram of the proposed system

4.3.1 BLOCK DIAGRAM DESCRIPTION

An Arduino Microcontroller acts like brain of the project where all sensors and the motor via relay are integrated to provide the solution for OSA patients as shown in Fig.4.1. Arduino UNO is used to perform the specified operations and Node MCU is used to transfer data to the cloud using wi-fi protocol . Here, the power supply of 230V is given to the step down transformer and then to the regulators, a voltage of 5V is given to the microcontroller. It is used to monitor the sensors like respiratory sensor and max 30100 sensor connected. MAX30100 monitors the heart rate and blood oxygen saturation level of the subject. Respiratory sensor it is placed inside the mask and it monitors the breath condition. Liquid Crystal Display (LCD) displays the values of heart rate , Spo2 and snoring events and the status of the compressor when changed.

Driver unit has the compressor motor to supply pressurized air to the Airflow unit. The Air flow unit consists of mask. The masks connect to a hose , supplying air through the nasal passage of the patients. The two push buttons are used to change to Manual mode and Auto mode accordingly via relay. In manual mode the compressor motor is turned ON and OFF accordingly. In auto mode, the supporting system is turned ON according to the snoring events monitored . If the snoring events are greater than 2 , the supporting system automatically turns ON to help patients breath stably as shown in Fig. 4.2.

The data is obtained in real-time via a pre-programmed Arduino board; the results are relayed back the ThingSpeak cloud for real-time monitoring and to know the recovery.

4.3.2 FLOW CHART FOR AUTO MODE

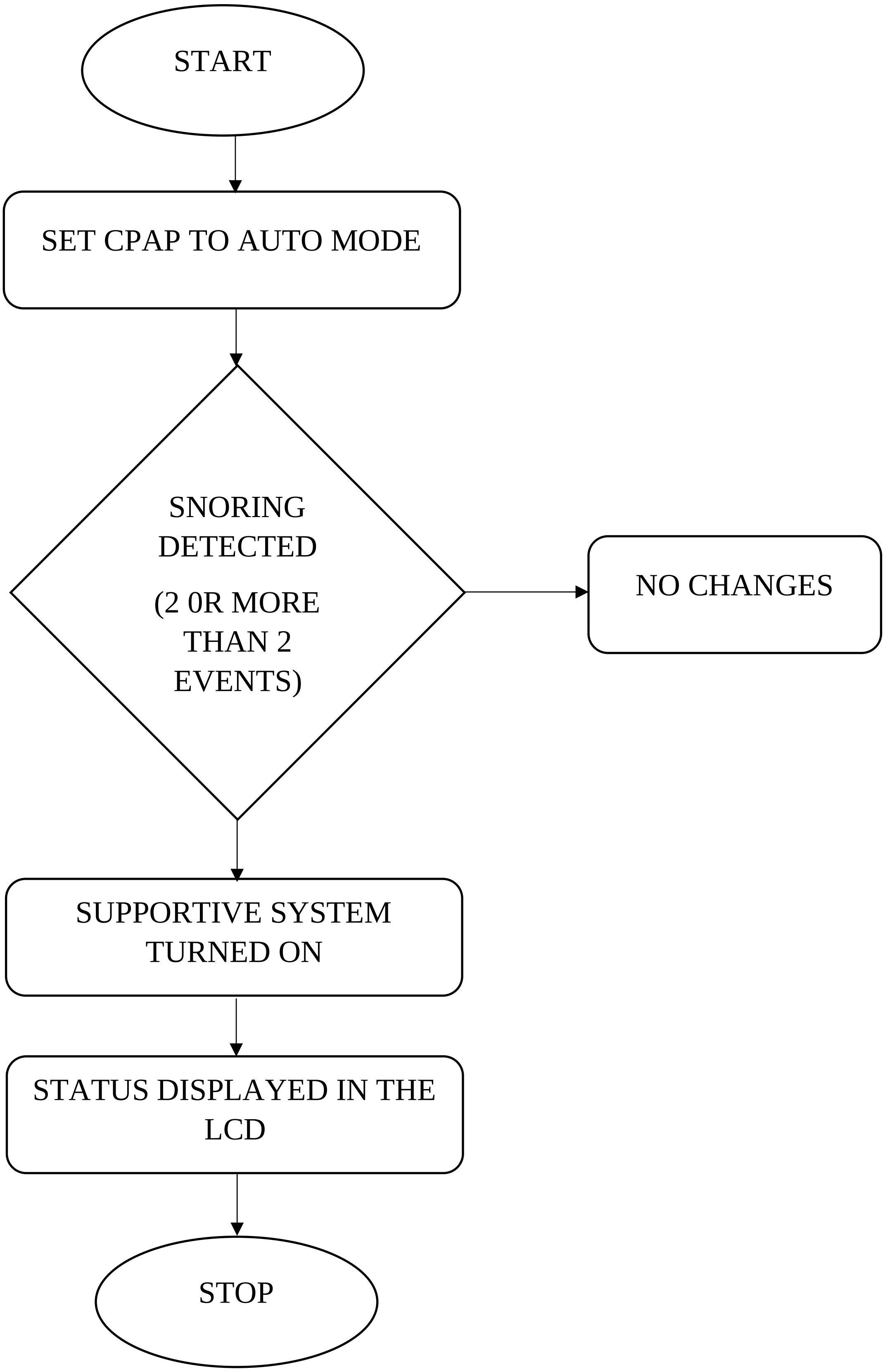

Fig 4.2 Flow chart for auto mode

4.3.3 SCHEMATIC DIAGRAM

The schematic diagram of the proposed system is shown in Fig. 4.3.

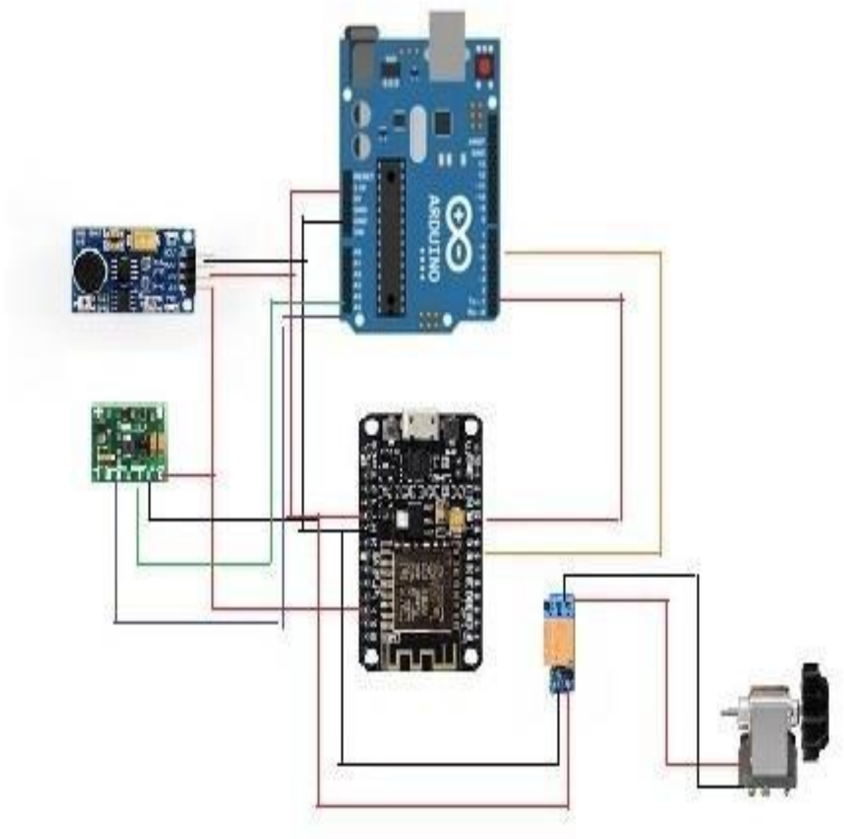

Fig. 4.3 Schematic Diagram

CHAPTER 5

MODULES ELUCIDATION

5.1 POWER SUPPLY

The AC voltage, typically 220 V rms, is connected to a transformer, which steps that AC voltage down to the level of the desired DC output. A diode rectifier then provides a full-wave rectified voltage that is initially filtered by a simple capacitor filter to produce a DC voltage. This resulting dc voltage usually has some ripple or AC voltage variation.

Fig. 5.1 Block diagram of power supply

A regulator circuit removes the ripples and also remains the same DC value even if the input DC voltage varies, or the load connected to the output DC voltage changes as shown in Fig. 5.1 . This voltage regulation is usually obtained using one of the popular voltage regulator IC units.

5.1.1 TRANSFORMER

The potential transformer will step down the power supply voltage (0-230V) to (0-15V) level. If the secondary has less turns in the coil then the primary, the secondary coil's voltage will decrease and the current will increase or decreased depend upon the wire gauge.

As shown in Fig.5.2 this is called a Step-down transformer. Then the secondary of the potential transformer will be connected to the rectifier.

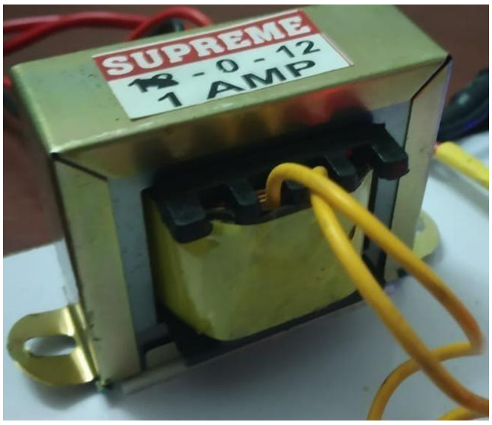

Fig. 5.2 Step-down Transformer

5.1.2 IC VOLTAGE REGULATOR

A fixed three-terminal voltage regulator has an unregulated DC input voltage, it is applied to one input terminal, a regulated DC output voltage from a third terminal, with the second terminal connected to ground as shown in Fig.5.3.

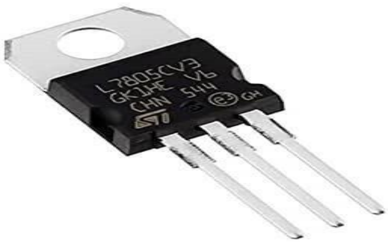

Fig.5.3 IC voltage regulator

In this circuit we are using 7805 regulator so it converts variable DC into constant positive 5V power supply. If the input voltage goes to below 14.6 Volt, the output is varied. Hence, we use 230/15V step-down transformer. Transformer output is higher than the regulator minimum input level.

5.2 ARDUINO UNO

The Arduino UNO is a standard board of Arduino. Arduino UNO is based on an ATmega328P microcontroller. It is easy to use compared to other boards, such as the Arduino Mega board, etc. The board consists of digital and analog Input/Output pins (I/O), shields, and other circuits as shown in Fig. 5.4.

The Arduino UNO includes 6 analog pin inputs, 14 digital pins, a USB connector, a power jack, and an ICSP (In-Circuit Serial Programming) header. It is programmed based on IDE, which stands for Integrated Development Environment. It can run on both online and offline platforms. It can be powered by the USB cable or by an external 9-volt battery, though it accepts voltages between 7 and 20 volts. It is similar to the Arduino Nano and Leonardo. The hardware reference design is distributed under a Creative Commons Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available. The preinstalled flash has a bootloader, which takes the memory of 0.5 Kb.

Here, SRAM stands for Static Random Access Memory, and EEPROM stands for Electrically Erasable Programmable Read-Only Memory.The word "uno" means "one" in Italian and was chosen to mark the initial release of Arduino Software. The Uno board is the first in a series of USB-based Arduino boards; it and version 1.0 of the Arduino IDE were the reference versions of Arduino, which have now evolved to newer releases. The ATmega328 on the board comes pre programmed with a bootloader that allows uploading new code to it without the use of an external hardware programmer.
The IDE is common to all available boards of Arduino.

5.2.1 POWER SUPPLY OF ARDUINO

The Arduino Uno board can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack.

Leads from a battery can be inserted in the GND and Vin pin headers of the POWER connector. The board can operate on an external supply from 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may become unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7-12V.

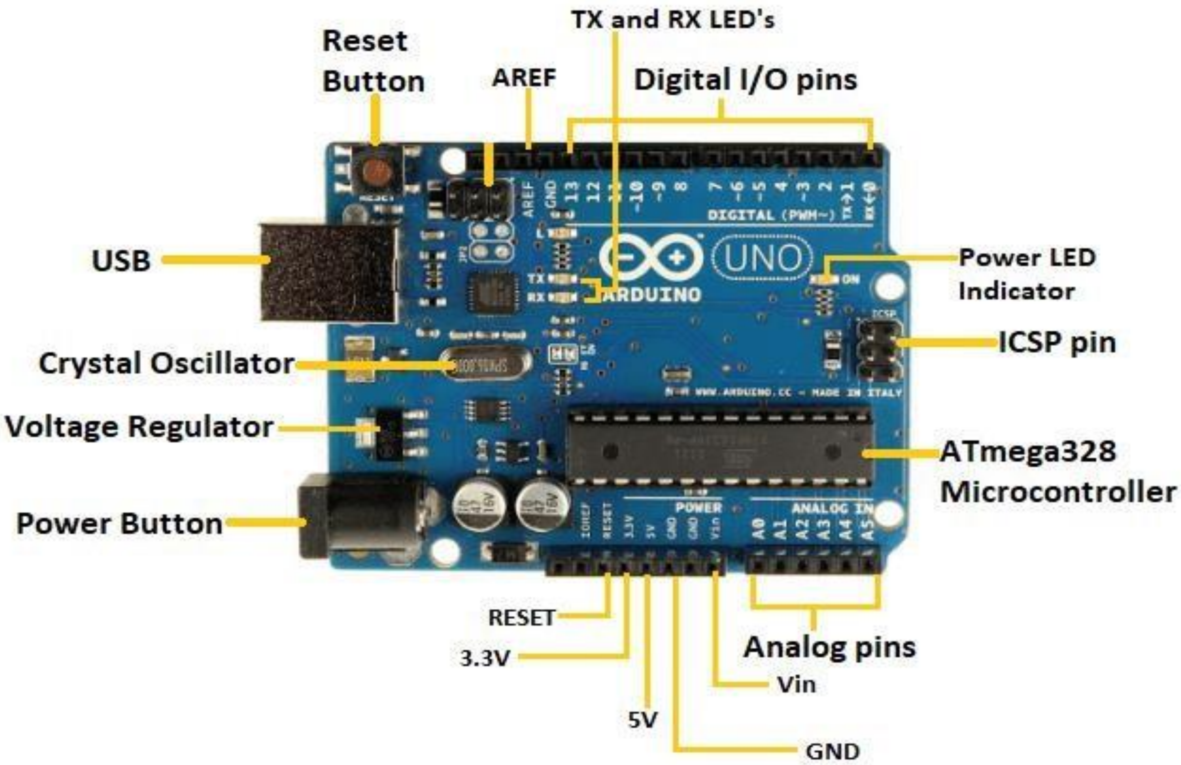

Fig . 5.4 Arduino UNO

- **ATmega328 Microcontroller-** It is a single chip Microcontroller of the ATmel family. The processor code inside it is of 8-bit. It combines Memory (SRAM, EEPROM, and Flash), Analog to Digital Converter, SPI serial ports, I/O lines, registers, timer, external and internal interrupts, and oscillator**.**
- **ICSP pin -** The In-Circuit Serial Programming pin allows the user to program using the firmware of the Arduino board.

- **Power LED Indicator**- The ON status of LED shows the power is activated. When the power is OFF, the LED will not light up.
- **Digital I/O pins**- The digital pins have the value HIGH or LOW. The pins numbered from D0 to D13 are digital pins.
- **TX and RX LED's-** The successful flow of data is represented by the lighting of these LED's.
- **AREF-** The Analog Reference (AREF) pin is used to feed a reference voltage to the Arduino UNO board from the external power supply.
- **Reset button**- It is used to add a Reset button to the connection.
- **USB-** It allows the board to connect to the computer. It is essential for the programming of the Arduino UNO board.
- **Crystal Oscillator**- The Crystal oscillator has a frequency of 16MHz, which makes the Arduino UNO a powerful board.
- **Voltage Regulator**- The voltage regulator converts the input voltage to 5V. ○ **GND**- Ground pins. The ground pin acts as a pin with zero voltage. ○ **Vin-** It is the input voltage.
- **Analog Pins**- The pins numbered from A0 to A5 are analog pins. The function of Analog pins is to read the analog sensor used in the connection. It can also act as GPIO (General Purpose Input Output) pins.

5.2.2 ATmega328p :

The AVR (Aortic Valve Replacement) core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the arithmetic logic unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers. The Atmel ATmega328P provides the following features: 32K bytes of in-system programmable flash with read-while-write capabilities, 1K bytes

EEPROM, 2K bytes SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte oriented 2-wire serial interface, an SPI serial port, a 6-channel 10-bit ADC (8 channels in TQFP and QFN/MLF packages), a programmable watchdog timer with internal oscillator, and five software selectable power saving modes. The idle mode stops the CPU while allowing the SRAM, Timer/Counters, USART, 2-wire serial interface, SPI port, and interrupt system to continue functioning. The power-down mode saves the register contents but freezes the oscillator, disabling all other chip functions until the next interrupt or hardware reset. In power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC noise reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC, to minimize switching noise during ADC conversions. In standby mode, the crystal/resonator oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low power consumption.

The device is manufactured using Atmel high density non-volatile memory technology. The on-chip ISP flash allows the program memory to be reprogrammed insystem through an SPI serial interface, by a conventional non-volatile memory programmer, or by an on-chip boot program running on the AVR core. The boot program can use any interface to download the application program in the application flash memory. Software in the boot flash section will continue to run while the application flash section is updated, providing true read-while-write operation. By combining an 8-bit RISC CPU with in-system self-programmable flash on a monolithic chip, the Atmel ATmega328P is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications. The ATmega328P AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, incircuit emulators, and evaluation kits.

Features of ATmega328p are

- High performance, low power AVR® 8-bit microcontroller.
- Advanced RISC architecture.
- High endurance non-volatile memory segments

5.2.3. COMMUNICATION

The Arduino Uno ATmega328 offers UART TTL-serial communication, and it is accessible on digital pins like TX (1) and RX (0). The software of an Arduino has a serial monitor that permits easy data. There are two LEDs on the board like RX & TX which will blink whenever data is being broadcasted through the USB.

A Software Serial library permits for serial communication on Arduino Uno digital pins and the ATmega328P supports TWI (I2C) as well as SPI-communication. The Arduino software contains a wired library for simplifying the utilization of the I2C bus.

The physical characteristics of an Arduino board mainly include length and width. The printed circuit board of the Arduino Uno length and width are 2.7 X 2.1 inches, but the power jack and the USB connector will extend beyond the previous measurement. The board can be attached on the surface otherwise case with the screw holes.

5.3 NODE MCU

Node MCU is an open source IoT platform. It includes firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware, which is based on the ESP-12 module as shown in Fig.5.5. The term “Node MCU” by default refers to the firmware rather than the dev kits. The firmware uses the Lua scripting language. It uses many open source projects, such as Lua-c json and spiffs. LUA based interactive firmware for Expressif ESP8622 Wi-Fi SoC, as well as an open source hardware board that contrary to the ESP8266 Wi-Fi modules includes a CP2102 TTL to USB chip for programming and debugging, is breadboard-friendly, and can simply be powered via its micro USB port.

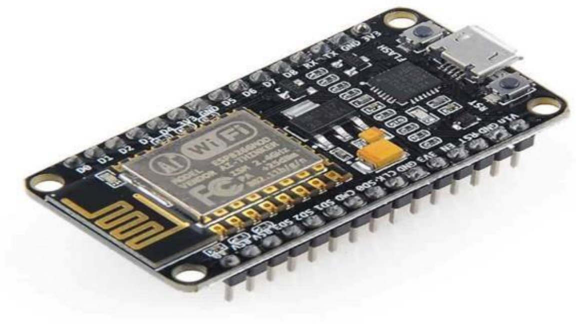

Fig 5.5 Node MCU

Features of Node MCU are

- Open-source
- Interactive
- Programmable
- Low cost
- Simple
- Smart
- Wi-Fi enabled

Nodes style network API- Event-driven API for network applications, which facilitates developers writing code running on a 5mm*5mm sized MCU in Nodejs style. Greatly speed up your IOT application developing process.

5.3.1. NODE MCU GPIO LAYOUT

Node MCU GPIO Layout is shown in Fig.5.6.

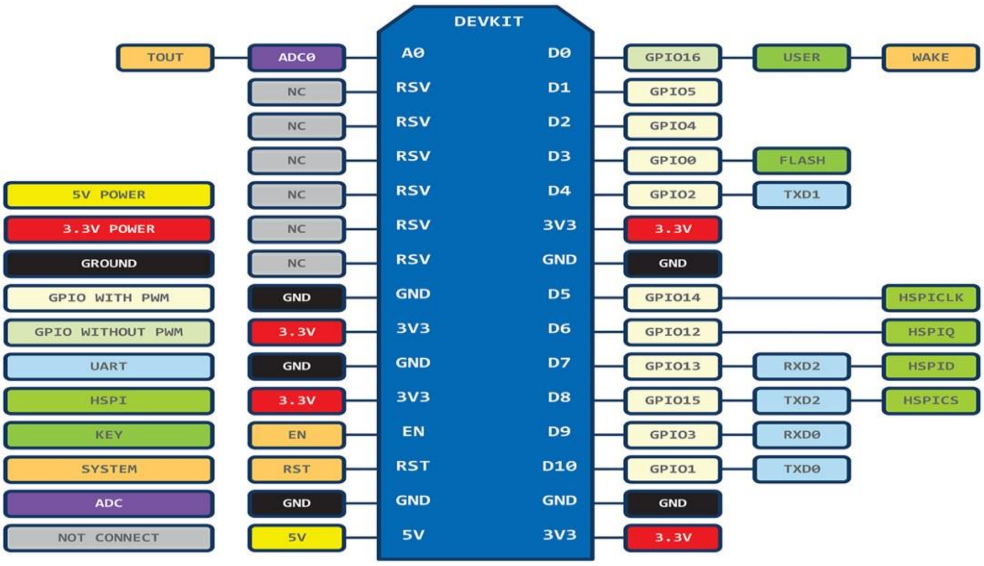

Fig. 5.6. Node MCU GPIO Layout

The ESP8266 chip requires 3.3V power supply voltage. It should not be powered with 5 volts like other arduino boards.

- NodeMCU ESP-12E dev board can be connected to 5Vusing micro USB connector or Vin pin available on board.

- The I/O pins of ESP8266 communicate or input/output max 3.3V only. i.e. the pins are NOT 5V tolerant inputs. In case you have to interface with 5V I/O pins, you need to use level conversion system (either built yourself using resistor voltage divider) .

5.4 RESPIRATION SENSOR

The sound sensor module (as shown in Fig.5.7) provides an easy way to detect sound and is generally used for detecting sound intensity. This module can be used for security, switch, and monitoring applications. Its accuracy can be easily adjusted for the convenience of usage. It uses a microphone which supplies the input to an amplifier, peak detector and buffer.

Fig 5.7 Respiration sensor

When the sensor detects a sound, it processes an output signal voltage which is sent to a microcontroller then performs necessary processing. Sound detection sensor module for arduino detects whether sound has exceeded a threshold level set point is adjusted via an on board potentiometer. When the sound level exceeds the set point, an LED on the module is illuminated and the output is set low.

Specifications of sound detection sensor module are

- Working voltage: DC 3.3-5V
- Adjustable Sensitivity
- Dimensions: 32 x 17 mm
- Signal output indication
- Single channel signal output
- Outputs low level and the signal light when there is sound
- Output in the form of digital switching outputs (0 and 1 high and low)

5.5 MAX30100

The MAX30100 is an integrated pulse oximetry and heart-rate monitor sensor solution as shown in Fig.5.8. It combines two LEDs, a photodetector, optimized optics, and low-noise analog signal processing to detect pulse oximetry and heart-rate signals. The MAX30100 operates from 1.8V and 3.3V power supplies and can be powered down through software with negligible standby current, permitting the power supply to remain connected at all times.

Fig 5.8 Max30100

5.5.1. BENEFITS AND FEATURES

● Complete Pulse Oximeter and Heart-Rate Sensor Solution Simplifies Design

- Integrated LEDs, Photo Sensor, and
  High-Performance Analog Front -End
- Tiny 5.6mm x 2.8mm x 1.2mm 14-Pin Optically
  Enhanced System-in-Package

● Ultra-Low-Power Operation Increases Battery Life for Wearable Devices

- Programmable Sample Rate and LED Current for Power Savings
- Ultra-Low Shutdown Current (0.7µA, typ)
- Advanced Functionality Improves Measurement Performance
- High SNR Provides Robust Motion Artifact Resilience
- Integrated Ambient Light Cancellation
- High Sample Rate Capability
- Fast Data Output Capability

5.5.2 PIN CONFIGURATION

The pin configuration of MAX30100 is shown in Fig. 5.9.

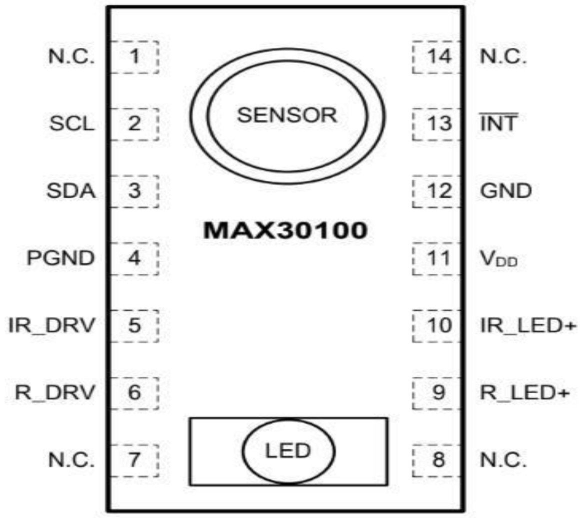

Fig. 5.9 Pin Configuration of MAX30100

The pin description is as follows

1, 7, 8, 14 - N.C. - No Connection. Connect to PCB Pad for Mechanical Stability.

2 -SCL -I2C Clock Input

3- SDA -I2C Clock Data, Bidirectional (Open-Drain)

4- PGND- Power Ground of the LED Driver Blocks

5- IR_DRV- IR LED Cathode and LED Driver Connection Point. Leave floating in circuit.

6- R_DRV- Red LED Cathode and LED Driver Connection Point. Leave floating in circuit.

9- R_LED+ - Power Supply (Anode Connection) for Red LED. Bypass to PGND for best performance. Connected to IR_LED+ internally.

10- IR_LED+ - Power Supply (Anode Connection) for IR LED. Bypass to PGND for best performance. Connected to R_LED+ internally.

11- VDD - Analog Power Supply Input. Bypass to GND for best performance.

12 – GND - Analog Ground

13 – INT - Active-Low Interrupt (Open-Drain)

5.5.3. SYSTEM BLOCK DIAGRAM

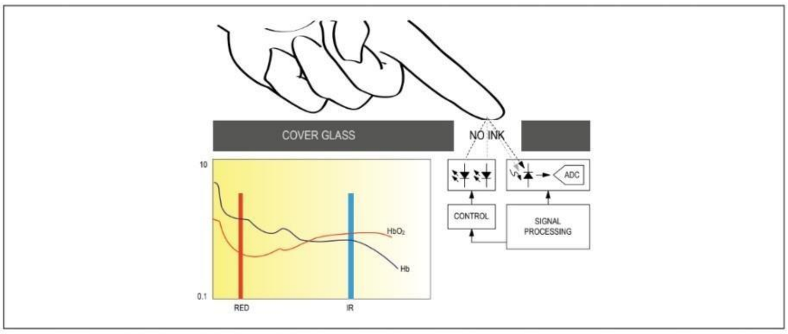

Fig.5.10 System Block Diagram

The MAX30100 is a complete pulse oximetry and heart rate sensor system solution designed for the demanding requirements of wearable devices. The MAX30100 provides very small total solution size without sacrificing optical or electrical performance. Minimal external hardware components are needed for integration into a wearable device. The MAX30100 is fully configurable through software registers, and the digital output data is stored in a 16-deep. FIFO within the device. The FIFO allows the MAX30100 to be connected to a microcontroller or microprocessor on a shared bus, where the data is not being read continuously from the device's registers.

SpO2 Subsystem -The SpO2 subsystem in the MAX30100 is composed of ambient light cancellation (ALC), 16-bit sigma delta ADC, and proprietary discrete time filter as shown Fig.5.10.The SpO2 ADC is a continuous time oversampling sigma delta converter with up to 16-bit resolution. The ADC output data rate can be programmed from 50Hz to 1kHz. The MAX30100 includes a proprietary discrete time filter to reject 50Hz/60Hz interference and low-frequency residual ambient noise.

The applications are

● Wearable Devices

● Fitness Assistant Devices

● Medical Monitoring Devices

5.6 LCD DISPLAY :

Liquid crystal displays (LCDs) have materials which combine the properties of both liquids and crystals.

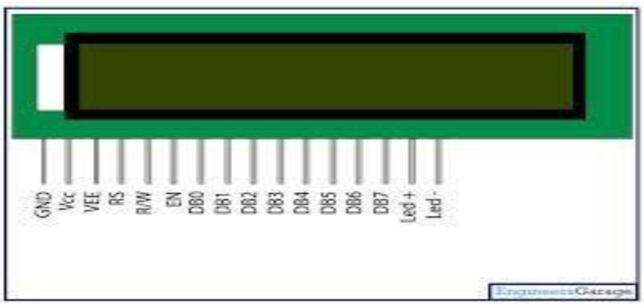

Fig. 5.11 LCD Display

An LCD consists of two glass panels (as shown in Fig. 5.11), with the liquid crystal material sand witched in between them. The inner surface of the glass plates are coated with transparent electrodes which define the character, symbols or patterns to be displayed polymeric layers are present in between the electrodes and the liquid crystal, which makes the liquid crystal molecules to maintain a defined orientation angle.

5.6.1 POWER SUPPLY OF LCD

The power supply should be of +5V, with maximum allowable transients of 10mv. To achieve a better / suitable contrast for the display, the voltage (VL) at pin 3 should be adjusted properly.

- Develop a uniquely decoded ‘E’ strobe pulse, active high, to accompany each module transaction. Address or control lines can be assigned to drive the RS and R/W inputs.

- If a parallel port is used to drive the RS, R/W and ‘E’ control lines, setting the ‘E’ bit simultaneously with RS and R/W would violate the module’s set up time. A separate instruction should be used to achieve proper interfacing timing requirements.

5.6.2 ADDRESS COUNTER

The address counter allocates the address for the DD RAM and CG RAM read/write operation when the instruction code for DD RAM address or CG RAM address setting, is input to IR, the address code is transferred from IR to the address counter. After writing/reading the display data to/from the DD RAM or CG RAM, the address counter increments/decrements by one the address, as an internal operation. The data of the address counter is output to DB0 to DB6 while R/W = 1 and RS = 0.

5.7 RELAY

Relay is an electromechanical device that uses an electric current to open or close the contacts of a switch.

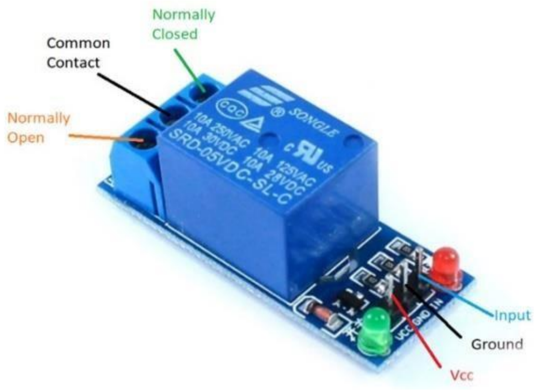

Fig.5.12 Single-Channel Relay

The single-channel relay module (as shown in Fig. 5.12) is much more than just a plain relay, it comprises of components that make switching and connection easier and act as indicators to show if the module is powered and if the relay is active or not.

5.7.1 MODULE PIN DESCRIPTION

The pin description of the single-channel relay module is shown in Table. 5.1.

Table. 5.1 Relay Pin Description

Pin Number	Pin Name	Description
1	Relay Trigger	Input to activate the relay
2	Ground	0V reference
3	VCC	Supply input for powering the relay coil

4	Normally Open	Normally open terminal of the relay
5	Common	Common terminal of the relay
6	Normally Closed	Normally closed contact of the relay

Single-Channel Relay Module Specifications

- Supply voltage – 3.75V to 6V
- Quiescent current: 2mA
- Current when the relay is active: ~70mA
- Relay maximum contact voltage – 250VAC or 30VDC
- Relay maximum current – 10A

The relay uses an electric current to open or close the contacts of a switch. This is usually done using the help of a coil that attracts the contacts of a switch and pulls them together when activated, and a spring pushes them apart when the coil is not energized. There are two advantages of this system – First, the current required to activate the relay is much smaller than the current that relay contacts are capable of switching, and second, the coil and the contacts are galvanically isolated, meaning there is no electrical connection between them. This means that the relay can be used to switch mains current through an isolated low voltage digital system like a microcontroller.

Single-Channel Relay Module Applications are

- Mains switching
- High current switching
- Isolated power delivery
- Home automation

5.8 CENTRIFUGAL COMPRESSOR

Power absorbing turbomachines used to handle compressible fluids like air, gases etc, can be broadly classified into: (i) Fans (ii)Blowers and (iii)Compressors.

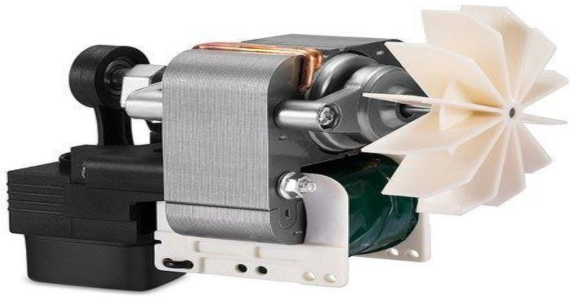

Fig 5.13 Compressor motor

These machines produce the head (pressure) in the expense of mechanical energy input. The pressure rise in centrifugal type machines are purely due to the centrifugal effects. A fan usually consists of a single rotor with or without a stator. It causes only a small pressure rise as low as a few centimeters of water column. Generally it rises the pressure upto a maximum of 0.07 bar (70 cm WG). In the analysis of the fan, the fluid will be treated as incompressible as the density change is very small due to small pressure rise. Fans are used for air circulation in buildings, for ventilation, in automobiles in front of engine for cooling purposes etc.

Blower may consists of one or more stages of compression with its rotors mounted on a common shaft. The air is compressed in a series of successive stages and is passed through a diffuser located near the exit to recover the pressure energy from the large kinetic energy. The overall pressure rise may be in the range of 1.5 to 2.5 bars. Blowers are used in ventilation, power station, workshops etc. Compressor is used to produce large pressure rise ranging from 2.5 to 10 bar or more. A single stage compressor can generally produce a pressure rise up to 4 bar. Since the velocities of air flow are quite high, the Mach number and compressibility effects may have to be taken into account in evaluating the stage performance of a compressor.In general the centrifugal compressor may be known as a fan, blower, supercharger etc, depending on the need to be served as shown in Fig.5.13 . Broadly speaking, fans are the low-pressure compressors; blowers are the medium pressure compressors. It is therefore the analysis of one, say centrifugal compressor, will also holds good to the other machines like blower, fans.

5.8.1. IMPORTANT ELEMENTS

It mainly consists of : (i) inlet casing with the converging nozzle (ii) the impeller (iii) the diffuser and (iv) the outlet casing.

The impeller convert the supplied mechanical energy into fluid energy whereby the fluid kinetic energy and the static pressure rises. An impeller is made of radial blades which are brazed to the shroud. It can be made from a single piece consisting of both the inducer and a largely radial portion. The inducer receives the flow between the hub and tip diameters (dh and dt) of the impeller eye and passes on to the radial portion of the impeller blades. The flow approaching the impeller may be with or without swirl. The inlet diameter 4.2 Important Elements of a Centrifugal Compressor. The function of the inlet casing with the conversant nozzle is to accelerate the entering fluid to the impeller inlet. The inlet nozzle accelerate the fluid from the initial condition (state 0) to the entry of the Inlet Guide Vanes (IGV) which direct the flow in the desired direction at the inlet of the impeller (state 1).

5.9 CPAP MASK

Patients wear a CPAP mask during the night over their nose and mouth. The masks connect to a small machine with a hose as shown in Fig.5.14. The machine pumps pressured air into the patient’s airway to keep it open while they're sleeping.

Fig 5.14 CPAP Mask

The nasal CPAP mask covers the patient's nose from the bridge to their upper lip area. This delivers an indirect airflow to the airway via the nasal mask and works well for patients who need higher pressure settings.The nasal CPAP mask offers your patients many versatile options and is a popular compromise between the bulky full face CPAP mask and the lightweight nasal pillow. Nasal masks are popular among patients who wear CPAP mask since there are many fits and sizes available.

A physician may recommend nasal CPAP masks to patients who:

- Move around in their sleep a lot.
- Need a higher pressure setting on their CPAP machine.
- Want a good selection of mask options they can pick from.
- Prefer a more natural airflow.

5.10 PUSH BUTTON

A push button switch is a small, sealed mechanism that completes an electric circuit when you press on it as shown in Fig.5.15 . When it's on, a small metal spring inside makes contact with two wires, allowing electricity to flow. When it's off, the spring retracts, contact is interrupted, and current won't flow. Push buttons can be explained as simple power controlling switches of a machine or appliance. These are generally metal or thermoplastic switches that are intended to grant easy access to the user. These switches are ordinarily used in calculators, push-button telephones, kitchen appliances, magnetic locks, and several other mechanical and electronic devices used across homes or industries. The design of the push button is such that it can accommodate a human finger to control the system easily. For machinery with complicated operations and various buttons, there are donations for separate colours of the push button. This enables users to identify what function the switch is bound to perform. It comes in two basic kinds like momentary and non-momentary.

Fig.5.15 Push Button

Momentary Contact - Momentary switches work only as long as you press on them, like the buttons on a phone, calculator or door buzzer. They can be subdivided into normally-on and normally-off types.

Non-Momentary Contact - Non-momentary switches take one push to turn on, another to turn off. TVs and stereos use non-momentary switches for their power buttons.

CHAPTER 6

SOFTWARE DESCRIPTION

6.1 THINGSPEAK PLATFORM

ThingSpeak is an open-source software written in Ruby which allows users to communicate with Internet enabled devices as shown in Fig. 6.1. It facilitates data access, retrieval and logging of data by providing an API to both the devices and social network websites. ThingSpeak was originally launched by ioBridge in 2010 as a service in support of IoT applications. ThingSpeak has integrated support from the numerical computing software MATLAB from MathWorks, allowing ThingSpeak users to analyze and visualize uploaded data using MATLAB without requiring the purchase of a MATLAB license from MathWorks.

ThingSpeak™

Fig 6.1 ThingSpeak

MATLAB helps you develop IoT algorithms. MATLAB and ThingSpeak helps you collect and analyse IoT data, quickly and easily. MATLAB and Simulink help you develop smart connected devices. MATLAB supports cloud deployment for small to medium scale IoT systems, up to large enterprise systems.

6.1.1. IoT

The Internet of things (IoT) describes physical objects (or groups of such objects) with sensors, processing ability, software, and other technologies that connect and exchange data with other devices and systems over the Internet or other communications networks. Internet of things has been considered a misnomer because devices do not need to be connected to the public internet, they only need to be connected to a network and be individually addressable.

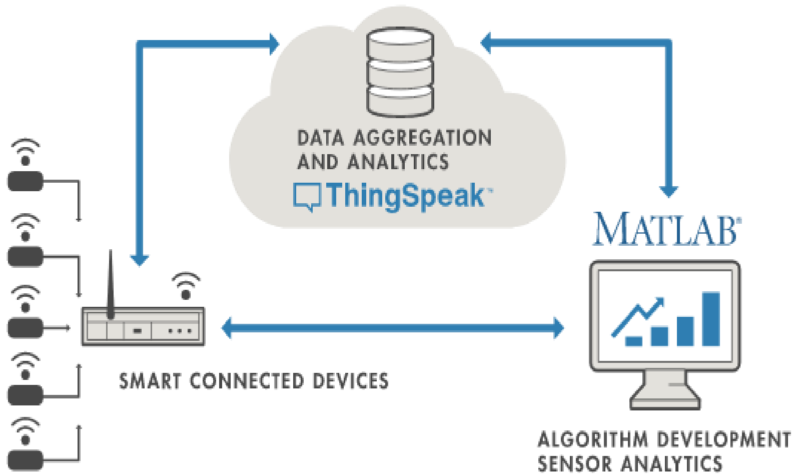

Fig. 6.2 ThingSpeak IOT

In ThingSpeak , IoT analytics platform service that allows you to aggregate, visualize and analyze live data streams in the cloud as shown in Fig.6.2. ThingSpeak provides instant visualizations of data posted by your devices to ThingSpeak. With ThingSpeak, your data is stored in channels. Each channel stores up to 8 fields of data. Users of the free license will also be limited to 4 channels. For users of the free option, the message update interval limit remains limited at 15 seconds.

6.2 ARDUINO IDE

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board as shown in Fig.6.3. This software can be used with any Arduino board. It is a text editor like a notepad with different features. It is used for writing code, compiling the code to check if any errors are there and uploading the code to the Arduino.

Fig.6.3 Arduino IDE

Arduino boards are programmed in “C.” C is a popular system programming language that has minimal execution time on hardware in comparison to other highlevel programming languages. It’s the reason most of the operating systems and several programming languages are built on C. Much like other microcontrollers, the AVR microcontrollers housed in Arduino boards are programmed in a subset of C. A general term for such subsets is “Embedded C” because they apply to programming embedded controllers. The language in which Arduino is programmed is a subset of C and it includes only those features of standard C that are supported by the Arduino IDE.

CHAPTER 7

RESULTS AND DISCUSSION

In this project, an effective CPAP design is proposed to treat the Obstructive Sleep Apnea (OSA) patients. It is used to monitor the sensors like respiratory sensor and max 30100 sensor connected. MAX30100 monitors the heart rate and blood oxygen saturation level of the subject. Respiratory sensor it is placed inside the mask monitors the breath condition. Liquid Crystal Display (LCD) displays the values of heart rate , Spo2 and snoring events and the status of the compressor when changed. The two push buttons are used to change to Manual mode and Auto mode accordingly. In manual mode the compressor motor is turned ON and OFF to the comfort of patient by the care takers or the patient themselves and status is displayed in LCD as shown in Fig.7.2 & Fig.7.3.This is the method used in health care units where the physicians will monitor the OSA patient.

In auto mode , if the patient is sensed to have any difficulties in breathing the compressor motor turns ON automatically to provide the pressurized air through the mask of the patient, providing instant solution by clearing the nasal passage as shown in Fig.7.4. The subject is sensed to have cessations in breathing by measuring the number of respiratory events. Considering this prototype if the detected event is less than 2 as shown in Fig.7.5 , the LCD displays the condition to be stable as shown in Fig.7.6. But if the snoring events is detected to be 2 or more than 2 as shown in Fig. 7.7, the snoring gets detected as shown in Fig.7.8 and compressor motor turns ON and the air flows into the mask as shown in Fig.7.9.The final module is shown in Fig.7.1. The parameters displayed are sent to the ThingSpeak cloud platform. The data can be monitored in the form of graph as shown in Fig.7.10 and it can also be viewed in mobile as shown in Fig.7.11,Fig.7.12,Fig.7.13. It can also be stored in the form excel sheet as shown in Fig.7.14 to know the recovery.

7.1. FINAL MODULE

The final is shown in figure 7.1

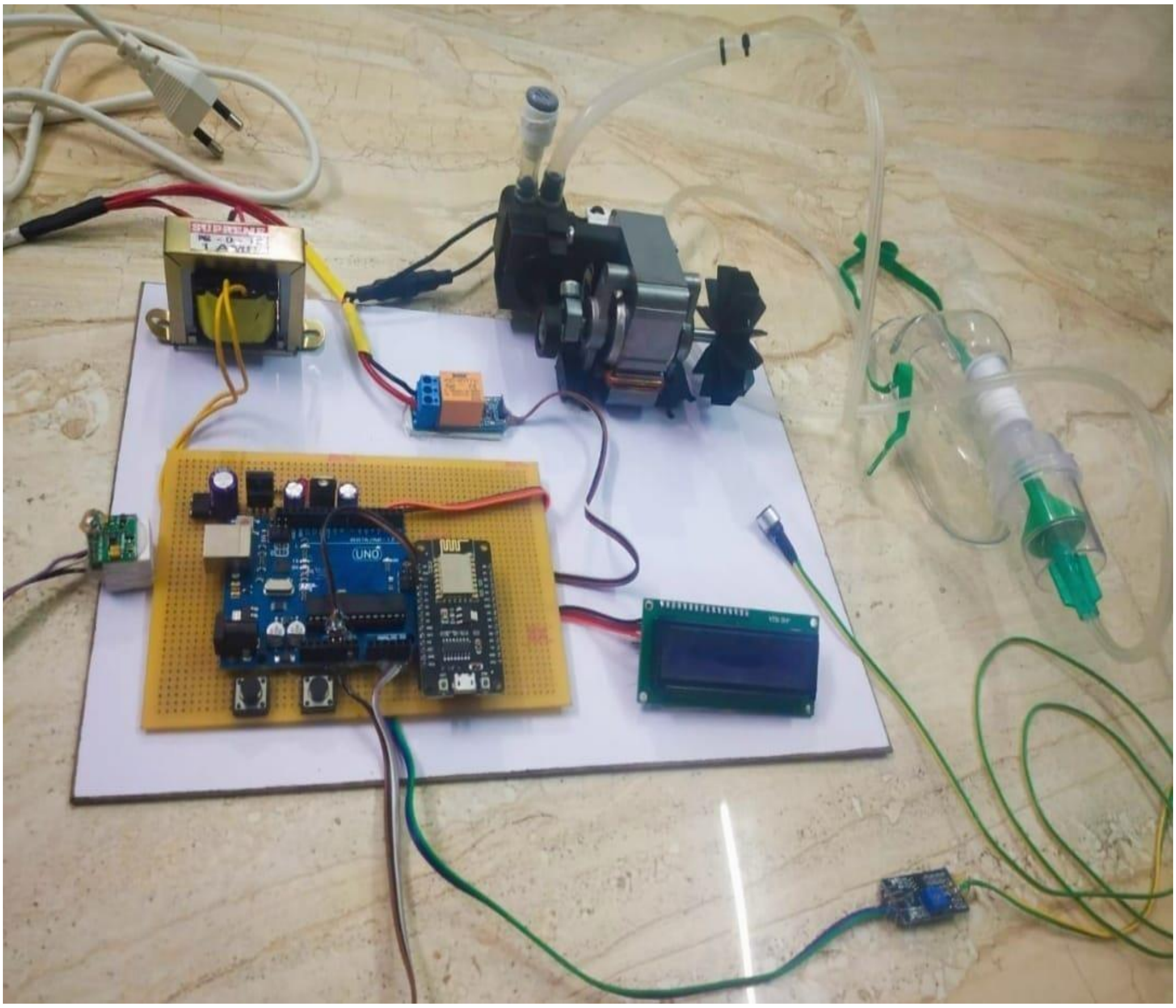

Fig 7.1 Final module

7.2. OUTPUT

Fig 7.2 Manual Mode

Fig 7.3 Air Flow ON

Fig 7.4 Auto Mode

Fig 7.5 Snoring Events < 2

Fig 7.6 Stable Condition

Fig 7.7 Snoring Events > 2

Fig 7.8 Snoring Detected

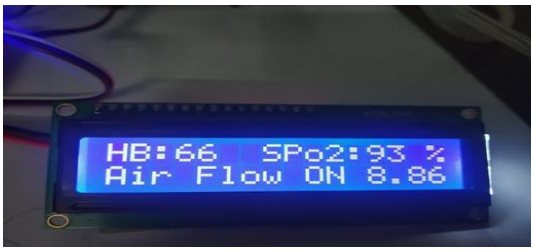

Fig 7.9 Air Flow ON

In figure 7.2, Auto mode off is displayed by LCD whenever we press the push button of manual mode, the automatic mode gets turn off and manual mode turns on. And the motor pumps the air and the air flows into the CPAP mask via tube to treat the person which is displayed by the lcd as Airflow on as shown in figure 7.3.

Whenever we press the push button of automatic mode, the manual mode gets turn off and the automatic mode gets turn on which is displayed by the LCD as Auto mode on as shown in figure 7.4. In figure 7.5, the LCD displays s:1which means that the snoring event less than two. So the person is in stable condition. The LCD displays as Stable as shown in figure 7.6. In figure 7.7, the LCD displays s:3 which means that the snoring level is greater than two. The LCD displays Snoring detected as shown in figure 7.8. So the motor gets turned on automatically and pumps the air and the air flows into CPAP mask via tube to treat the person. Now the LCD displays Airflow on as shown in figure 7.9.

The maximum snoring limit is set as for 2 seconds. If the snoring limit is less than two sceonds, then the person is in stable condition. The motors does not pump the air. If the snoring limit is greater than two seconds, then the person is not in stable condition. The snoring gets detected. The motor starts to pump the air and the air flows into the CPAP mask via the tube connected to the motor and the person gets treated.

At each and every point, the datas which are displayed in the LCD are sent and stored in Thingspeak platform. It is an IOT based platform where the datas are continuously monitored and stored to know about the recovery of the person.

7.3. SOFTWARE OUTPUT

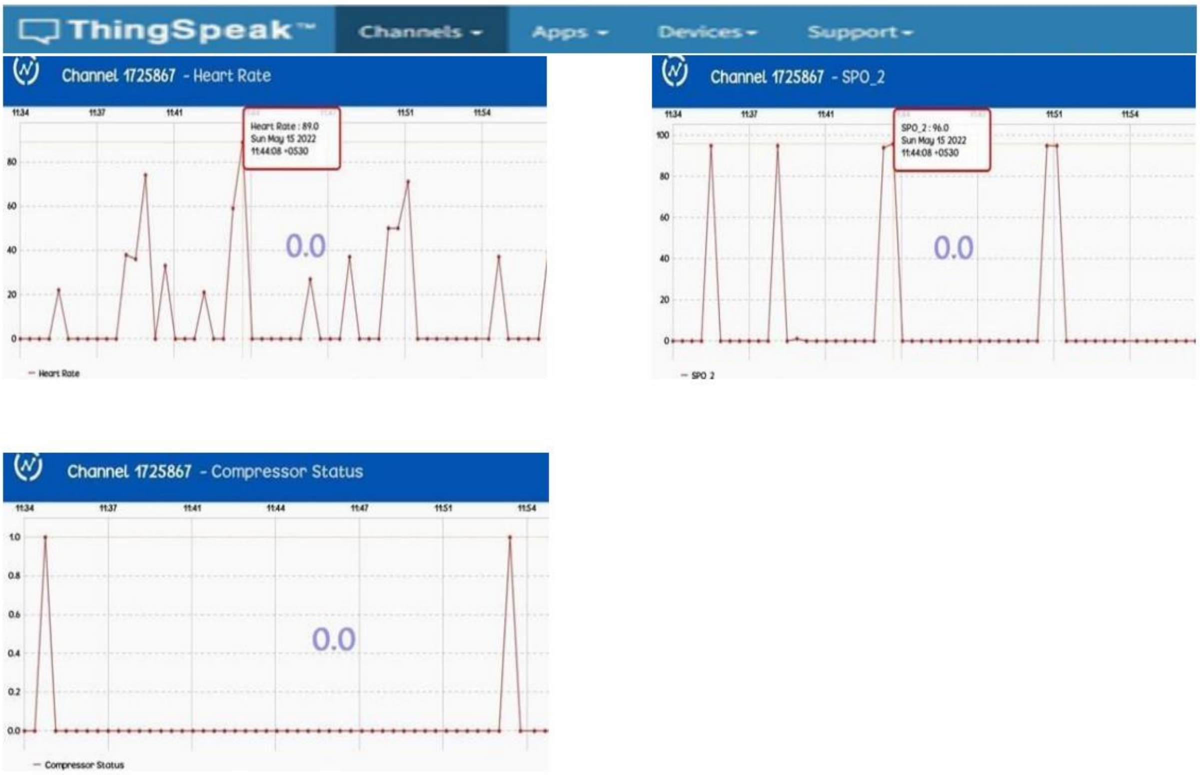

Fig.7.10 Software Output

The software output displayed in the Thingspeak platform is shown in fig.7.10. Here the various parameters are monitored and stored. The first graph depicts the heart rate of the person. The second graph depicts the SPO2 level of the person. The third graph shows the compressor status. All the details are stored in the Thingspeak platform to know about the recovery of the person.

7.3.1. MOBILE VIEW:

Fig.7.11 Heart rate in mobile view

The heart rate of the person who is wearing the CPAP mask is stored in the Thingspeak platform as shown in fig. 7.11.

Fig.7.12 SPO2 level in mobile view

The SPO2 level of the person who is wearing the CPAP mask is stored in the Thingspeak platform as shown in fig. 7.12.

Fig.7.13 Compressor status in mobile view

The compressor status is stored in the Thingspeak platform as shown in fig.7.13.

CHAPTER 8

CONCLUSION

In this project we have focused on the problem encountered by OSA patients . Most of the devices to treat sleep apnea patients are bulky and are not cost effective for normal people. There is no awareness about sleep apnea to most of the people. The basic idea of this CPAP is to design more compactable, portable and cost effective system. It is also used to treat and provide solution for sleep apnea patients.

Continuous Positivity Airway Pressure (CPAP) using IOT, uses Respiration sensor for monitoring the breath condition, SPO2 and Pulse Sensor for monitoring the heart rate and blood oxygen saturation level. All the parameters are monitored by the microcontroller, once the microcontroller detects any difficulties in breathing conditions it will automatically turn on the supportive system for the patient. The data are continuously monitored and stored on the ThingSpeak cloud to know the recovery. In this system constant flow of pressurised air in given to the subject for stable breathing during sleep.

FUTURE SCOPE

In the future the pressure can be regulated according to the age and condition of the patients. Nowadays, sleeping disorder rates are higher than before. More people are suffering from hypertension, stress-related problems, and heart-related problems which cause sleep apnea. The proposed system will be essential in the treatment of sleep apnea.

We can also incorporate Bilevel – Positive Airway Pressure (BiPAP). The main difference between CPAP and BiPAP is that the pressure delivered by a CPAP machine is same during the inhalation and exhalation but the pressure delivered by a BiPAP machine is different during inhalation and exhalation.

We can also adapt to an advance form of CPAP technique by incorporating Artificial Intelligence instead of Internet Of Things in future.

APPENDIX

CPAP_CODE

```
#include <ESP8266WiFi.h>
#include <LiquidCrystal_PCF8574.h>
#include <Wire.h>
#include <SoftwareSerial.h>
#include "ThingSpeak.h"
#define compres D0
LiquidCrystal_PCF8574 lcd(0x27);
#define RX D5
#define TX D6
SoftwareSerial ard_node(RX, TX, false, 256);
//SoftwareSerial ard_node;
#define splash splash1
#define resp A0
#define ky1 D3
#define ky2 D4
int Cpap;
int a, b;
char ssid1[] = "Project";
char password1[] = "1234567890";
unsigned long channelID = 1725867;
const char * writeAPIKey = "4OT8MAEBR6SUWMZF"; // write API key for your
ThingSpeak Channel
const char* server = "api.thingspeak.com";
String th;
String modefn = "Auto";
int tup = 20;
WiFiClient client;
int H_B, SPO_2;
int res_r, resfn;
int res_r_t = 580;
float res_tim = 0.0;
int C_Status;
int C_tim = 10;
float cc_tim = C_tim;
int k1_r, k2_r;
```

```
int C_auto;
int DispDelay=20;
byte fade = 1;
byte state = 1;
void setup() {
 // put your setup code here, to run once:
 Serial.begin(115200);
 ard_node.begin (115200);
 // ard_node.begin (115200, SWSERIAL_8N1, RX, TX, false, 256);
 ard_node.println("");
 pinMode(ky1, INPUT_PULLUP);
 pinMode(ky2, INPUT_PULLUP);
 pinMode(resp, INPUT);
 pinMode(compres, OUTPUT);
 digitalWrite(compres, LOW);
 Wire.beginTransmission(0x27);
 LcDSet();
 k1_r = digitalRead(ky1);
 k2_r = digitalRead(ky2);
 delay(2000);
 lcd.clear();
 setupWifi();
 ThingSpeak.begin(client);
 lcd.clear();
}
void LcDSet() {
 lcd.begin(16, 2);
 // Print a message to the LCD.
 lcd.clear();
 lcd.setBacklight(1);
 splash(0, "CPAP");
 splash(1, "Machine");
 delay(1000);
}
void setupWifi() {
 Serial.println();
 Serial.println();
 Serial.print(ssid1);
 WiFi.begin(ssid1, password1);
```

```
Serial.print("Connecting to ");
 // while (WiFi.status() != WL_CONNECTED)
 // {
 //
 // if (b <= 0) {
 // b = 2;
 // }
 // delay(800);
 // Serial.print(".");
 // lcd.setCursor(0, 0);
 // lcd.print("I'm Connecting ");
 // lcd.setCursor(0, 1);
 // lcd.print("to");
 // lcd.setCursor(b, 1);
 // lcd.print(".");
 //
 // b = b + 1;
 // if (b >= 16)
 // {
 // lcd.clear();
 // b = 2;
 // }
 //
 // }
 if (WiFi.status() != WL_CONNECTED) {
 lcd.clear();
 lcd.setCursor(0, 0);
 lcd.print("Connected !!" );
 delay(1000);
 delay(1000);
 lcd.clear();
 }
 Serial.println("");
 Serial.println("WiFi connected");
 Serial.println("IP address: ");
 Serial.println(WiFi.localIP());
}
void loop() {
 getArd();
```

```
 lcd.home();
 setDisp();
 k1_r = digitalRead(ky1);
 k2_r = digitalRead(ky2);
 if (k1_r == 0 and C_auto == 0) {
 splash(1, "Auto Mode OFF");
 modefn = "Manual";
 delay(200);
 C_auto = 1;
 }
 else if (k1_r == 0 and C_auto == 1) {
 splash(1, "Auto Mode ON");
 modefn = "Auto";
 delay(200);
 C_auto = 0;
 }
 if (C_auto == 0) {
 getres();
 }
 else {
 if (k2_r == 0 and C_Status == 0) {
 digitalWrite(compres, HIGH);
 splash(1, "C ON");
 delay(100);
 C_Status = 1;
 }
 else if (k2_r == 0 and C_Status == 1) {
 delay(200);
 splash(1, "C OFF");
 digitalWrite(compres, LOW);
 delay(100);
 C_Status = 0;
 }
 }
 UpdateServer();
 delay(400);
}
void getArd() {
 String SS = ""; String SSs = "";
```

```
while (ard_node.available() > 0) {
 char S = ard_node.read();
 SS += S;
 delay(10);
 }
 if (SS.length() > 0) {
 Serial.println(SS);
 String sys = getSplitValue(SS , ',', 0);
 String dia = getSplitValue(SS , ',', 1);
 String ff = getSplitValue(SS , ',', 2);
 H_B = sys.toInt();
 SPO_2 = dia.toInt();
 f_Status = ff.toInt();
 ard_node.flush();
 }
}
void getres() {
res_tim += 0.40;
 res_r = analogRead(resp);
 Serial.print("res ");
 Serial.println(res_r);
 //res_r=map(res_r,0,1024,1024,0);
 //
 if (C_Status == 0) {
 if (res_r > res_r_t) {
 resfn++;
 splash(0, "Snoring");
 splash(1, "Detected");
 }
 }
 if (res_tim > 8 and resfn >= 2 and C_Status == 0) {
 Serial.println("Comp ON");
 digitalWrite(compres, HIGH);
 C_Status = 1;
 resfn = 0;
 res_tim = 0;
 }
 else if (res_tim > C_tim and C_Status == 1) {
 C_Status = 0;
```

```
 Serial.println("Comp OFF");
 lcd.setCursor(0, 1);
 lcd.print("Air Flow OFF ");
 digitalWrite(compres, LOW);
 resfn = 0;
 res_tim = 0;
 }
 else if (res_tim > 10) {
 C_Status = 0;
 Serial.println("Normal");
 lcd.setCursor(0, 1);
 lcd.print("Stable ");
 digitalWrite(compres, LOW);
 resfn = 0;
 res_tim = 0;
 }
}
void setDisp() {
 if (DispDelay > 2 ) {
 display_ard(H_B, SPO_2);
 DispDelay = 0;
 }
 DispDelay++;
}
void display_ard(int val1 , int val2) {
 lcd.setCursor(0, 0);
 lcd.print("HB: ");
 lcd.setCursor(3, 0);
 lcd.print(val1);
 lcd.setCursor(7, 0);
 lcd.print("SPo2: ");
 lcd.setCursor(12, 0);
 lcd.print(val2);
 lcd.setCursor(15, 0);
 lcd.print("% ");
 if (C_Status == 0) {
 lcd.setCursor(0, 1);
 lcd.print("S : ");
 lcd.setCursor(3, 1);
```

```
 lcd.print(resfn);
 lcd.setCursor(12, 1);
 lcd.print(res_tim);
 cc_tim = C_tim;
 }
 else if (C_Status == 1 and C_auto == 1 ) {
 lcd.setCursor(0, 1);
 lcd.print("Air Flow ON ");
 }
 else {
 lcd.setCursor(0, 1);
 lcd.print("Air Flow ON ");
 lcd.setCursor(12, 1);
 lcd.print(cc_tim -= 0.38);
 }
}
void UpdateServer() {
 tup++;
 if (tup >= 50) {
 splash(1, "Sending Data");
 if (client.connect(server, 80)) {
 ThingSpeak.setField(1, String(H_B));
 ThingSpeak.setField(2, String(SPO_2));
 ThingSpeak.setField(3, String(C_Status));
 ThingSpeak.setField(4, modefn);
 // write to the ThingSpeak channel
 int x = ThingSpeak.writeFields(channelID, writeAPIKey);
 if (x == 200) {
 Serial.println("Channel update successful.");
 splash(1, "Data Sent");
 }
 else {
 Serial.println("Problem updating channel. HTTP error code " + String(x));
 splash(1, "Sending Failed");
 }
 }
 tup = 0;}
}
```

REFERENCES

[1] S. A. Pullano et al., "Medical Devices for Pediatric Apnea Monitoring and Therapy: Past and New Trends," in IEEE Reviews in Biomedical Engineering, vol. 10, pp. 199-212, 2017, doi: 10.1109/RBME.2017.2757899.

[2] R. Lazazzera et al., "Detection and Classification of Sleep Apnea and Hypopnea Using PPG and SpO$_2$ Signals," in IEEE Transactions on Biomedical Engineering, vol. 68, no. 5, pp. 1496-1506, May 2021, doi: 10.1109/TBME.2020.3028041.

[3] G. Surrel, A. Aminifar, F. Rincón, S. Murali and D. Atienza, "Online Obstructive Sleep Apnea Detection on Medical Wearable Sensors," in IEEE Transactions on Biomedical Circuits and Systems, vol. 12, no. 4, pp. 762-773, Aug. 2018, doi: 10.1109/TBCAS.2018.2824659.

[4] V. V. Kurup, K. R. Rahul, R. G. Kartha, E. T. Thampi, G. Thomas and P. J. Ajish, "Low-Cost Portable Emergency Mechanical Ventilator Proposed for Covid-19 Patients," 2021 IEEE 9th Region 10 Humanitarian Technology Conference (R10-HTC), 2021, pp. 1-5, doi: 10.1109/R10-HTC53172.2021.9641670.

[5] A Simplified Design of CPAP Device Construction by Using Arduino NANO for OSA Patients- https://www.researchgate.net/publication/357811669

[6] J. Reinders, B. Hunnekens, F. Heck, T. Oomen and N. van de Wouw, "Adaptive Control for Mechanical Ventilation for Improved Pressure Support," in IEEE Transactions on Control Systems Technology, vol. 29, no. 1, pp. 180-193, Jan. 2021, doi: 10.1109/TCST.2020.2969381.

[7] Journal of Pharmaceutical Innovation -https://doi.org/10.1007/s12247-021-095572- An Automated Jet Nebulizer with Dynamic Flow Regulation

[8] Arulvallal, Santhoshini; U., Snekhalatha; T., Rajalakshmi (2019). [IEEE 2019 International Conference on Communication and Signal Processing (ICCSP) – Chennai, India (2019.4.4-2019.4.6)] 2019 International Conference on Communication and Signal Processing (ICCSP) - Design and Development of Wearable Device for Continuous Monitoring of Sleep APNEA Disorder., (), 0050–0053. doi:10.1109/ICCSP.2019.8697961.

[9] M. Młyńczak, T. A. Valdez and W. Kukwa, "Joint Apnea and Body Position Analysis for Home Sleep Studies Using a Wireless Audio and Motion Sensor," in IEEE Access, vol. 8, pp. 170579-170587, 2020, doi: 10.1109/ACCESS.2020.3024122.

[10] M. M. Hasan et al., "Cost Effective Bluetooth Technology Based Emergency Medical Ventilator for Respiratory Support," 2021 International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI), 2021, pp. 1-5, doi: 10.1109/ACMI53878.2021.9528262.

[11] W. Senavongse, P. Boonchoo, N. Nimitsantiwong and A. Rangsivaranan, "Design and development of continuous positive airway pressure machine for snoring," 2016 9th Biomedical Engineering International Conference (BMEiCON), 2016, pp. 1-4, doi: 10.1109/BMEiCON.2016.7859645.

[12] V. Gusti et al., "The COSMIC Bubble Helmet: A Non-Invasive Positive Pressure Ventilation System for COVID-19," in IEEE Open Journal of Engineering in Medicine and Biology, vol. 1, pp. 312-315, 2020, doi: 10.1109/OJEMB.2020.3036742.