# “COVID-19 SOCIAL DISTANCE MONITORING AND CONTACT TRACING”

*Submitted in partial fulfillment of the requirement for the degree of*

**Bachelor of Engineering**

*in*

**Electronics & Communication Engineering**

*by*

| | |
|---|---|
| **1DS17EC121** | **SUHAS Swamy** |
| **1DS17EC433** | **PRATIBHA** |
| **1DS18EC419** | **LAXMI** |
| **1DS18EC413** | **HARISH HL** |

Under the Guidance of
**Dr. Kiran Gupta**
Project Guide
Professor, ECE Dept., DSCE, Bengaluru


**DAYANANDA SAGAR COLLEGE OF ENGINEERING**
**Department of Electronics & Communication Engineering**
(An Autonomous College affiliated to VTU Belgaum, accredited by NBA & NAAC)
Shavige Malleshwara Hills, Kumaraswamy Layout,
Bengaluru-560078, Karnataka, India


**2020-21**

## Certificate

Certified that the project work entitled **“ COVID-19 SOCIAL DISTANCE MONITORING AND CONTACT TRACING”** carried out by **SUHAS (1DS17EC121), PRATIBHA (1DS18EC433), LAXMI (1DS18EC419), HARISH H L(1DS18EC413)** are bonafide students of Dayananda Sagar College of Engineering, Bangalore, Karnataka, India in partial fulfillment for the award of Bachelor of Engineering in Electronics & Communication Engineering of the Visvesvaraya Technological University, Belagavi, Karnataka during the academic year 2020-21. It is certified that all corrections / suggestions indicated for project work have been incorporated in the report deposited to the ECE department, the college central library & to the university. This project report **(17EC8DCPR2)** has been approved as it satisfies the academic requirement in respect of project work prescribed for the said degree.

**Acknowledgement**

This work was supported by a Dayananda Sagar College of Engineering, without their support, this project would not have been possible.

We would like to thanks my advisor, Dr. Hemachandra Sagar (Chairman) and Dr. Premachandra Sagar (Vice Chairman) for giving me the support to explore the challenging project.

We convey our respectable thanks to “Galiswamy” Secretary of Dayananda Sagar College of Engineering.

We would also like to extend our gratitude to the Principal “ Dr. C.P.S. Prakash” and Head of the Department “ Dr. T.C.Manjunath” for giving continuous support throughout the project.

We would like to express special thanks of gratitude to guide “Dr. KIRAN GUPTA” for their able guidance and support in completing the Project.

We would like to express special thanks of gratitude to the Project Convener “ Dr. K.N. Pushpalatha” for supporting in the Project.

We would like to express special thanks of gratitude to the Department of Project Coordinators Dr. P.Vimala, Dr. A Rajagopal, Prof. Deepa N P, Prof. Manasa R K for their support in the Project.

We would like to express our heartfelt thanks to all supporting staff members and External guide.

We would like to thank my parents and friends who have helped me with their valuable suggestions and guidance has been very helpful in various phases of the completion of the project.

Last but not the least I would like to thank my classmates who have helped me a lot

**ABSTRACT**

The COVID-19 pandemic forced governments across the world to impose lockdowns to prevent virus transmissions. This resulted in the shutdown of all economic activity and accordingly the production at manufacturing plants across most sectors was halted. While there is an urgency to resume production, there is an even greater need to ensure the safety of the workforce at the plant site. Reports indicate that maintaining social distancing and wearing face masks while at work clearly reduces the risk of transmission. However, with ongoing COVID- 19 outbreaks and the threat of a second wave still present, the need to continue observing social distancing and contact tracing will be an important element in any plan for the safe reopening of colleges and businesses and operational facilities. The project will focus specifically on 1) wearable devices suitable for monitoring the populations at risk and those in quarantine, both for evaluating the health status of caregivers and management personnel, and for facilitating triage processes for admission to hospitals; 2) unobtrusive sensing systems for detecting the disease and for monitoring patients with relatively mild symptoms whose clinical situation could suddenly worsen in improvised hospitals; and 3) telehealth technologies for the remote monitoring and diagnosis of COVID-19 and related diseases. Finally, further challenges and opportunities for future directions of development are highlighted.

**Table of Contents**

Title Sheet i
Certificate ii
Declaration iii
Acknowledgement iv
Abstract v
Table of Contents vi
List of Figures
Chapter 1 Introduction
1.1 Overview
1.2 Objectives / Scope / Aim of the project work
Chapter 2 Literature Survey
Chapter 3 Structure of the project work
Chapter 4 Methodology
Chapter 5 Analytical Modelling
Chapter 6 Software tools used/ Results and discussion
Chapter 7 Advantages and Applications.
Chapter 7 Conclusions
Chapter 9 References

***CONTENTS***

Project work Phase-1 Report Declaration ii
Abstract ………………………………………………………………........................................ iii
Introduction 1

Overview
Objectives of the project work/ Problem statement definition
Motivation
Methodology
Literature survey
Overall Block Diagram/ Schematic/Circuit diagram/ Working principle/ related
Proposed Methodology/Flow diagram/ Pseudo code/Algorithm/related
Tools used for the project work
Results and Discussions
Work carried out till now, Simulation Results or Experimental Results
Applications/Advantages/Limitations
Conclusions References
Month-wise flow of events / Project Schedule PO-PSO Mapping
Budget Estimation

# 1. INTRODUCTION

**Overview**

Corona virus disease 2019 (COVID-19) has emerged as a pandemic with serious clinical manifestations including death. A pandemic at the large-scale like COVID-19 places extraordinary demands on the world's health systems, dramatically devastates vulnerable populations, and critically threatens the global communities in an unprecedented way. While tremendous efforts at the frontline are placed on detecting the virus, providing treatments and developing vaccines, it is also critically important to examine the technologies and systems for tackling disease emergence, arresting its spread and especially the strategy for diseases prevention. The objective of this article is to review enabling technologies and systems with various application scenarios for handling the COVID-19 crisis. The article will focus specifically on 1) wearable devices suitable for monitoring the populations at risk and those in quarantine, both for evaluating the health status of caregivers and management personnel, and for facilitating triage processes for admission to hospitals; 2) unobtrusive sensing systems for detecting the disease and for monitoring patients with relatively mild symptoms whose clinical situation could suddenly worsen in improvised hospitals; and 3) telehealth technologies for the remote monitoring and diagnosis of COVID-19 and related diseases. Finally, further challenges and opportunities for future directions of development are highlighted.

COVID-19 is a chronic condition or disease that causes inflammation and narrowing of the tubes, the passageways that allow air to enter and leave the lungs, making it harder to breathe [, If people with COVID-19 are exposed to a substance to which they are sensitive or a situation that changes their regular breathing patterns, the symptoms can become more severe. According to the latest World Health Organization (WHO) estimates, approximately 250 million people suffer from COVID-19worldwide, and almost 250 million Americans are affected by this disease according to AAFA, This disease is a public health problem in both rich and poor countries. Although there is no cure for COVID-19, effective treatments are available. The best way to manage COVID-19 is to avoid triggers, take medications to prevent symptoms and prepare to treat COVID-19 episodes if they occur.

COVID-19 is one of the most common chronic diseases and the third leading cause of hospitalization among adolescents. It is a medical condition that causes coughing, wheezing, and difficulty in breathing. During the period from 2008–2010, the prevalence of COVID-19 was higher among children than adults. According to the Center for Disease Control, it affects 7.1 million (1:11) children and its rampancy has increased by 15% in the last decade.Records obtained from the Center for Disease Control and Prevention also indicate that in 2013, about 21% of high school students (grades 9-12) had COVID-19. COVID-19 demands a great deal of health care utilization andentails a lot of missed days of school and work. COVID-19 is characterized by episodic respiratory symptoms and intermittent exacerbations.1 The symptoms, airflow obstruction, and exacerbations in COVID-19 vary greatly in both frequency of occurrence and severity. Monitoring these events is crucial to the care of patients with COVID-19and is directed at the early detection of exacerbations and monitoring of the day-to-day control of COVID-19. Monitoring can also be extended to investigate reasons for poor control and reasons for exacerbations, such as noncompliance and exposure to triggers.It is important to identify who will perform the monitoring because this has implications for the type of data that are collected, their validity, and their accuracy. COVID- 19can be monitored by the following people:

The patient withCOVID-19 because self-monitoring allows the early detection of exacerbations;

The treating physician to assess control of COVID-19and investigate reasons for poor control; and

Health care managers to assess the quality and cost of care for patients with COVID-19.

This article reviews COVID-19 monitoring from each of these perspectives. People with COVID-19 have sensitive airways which react to environmental triggers, causing 'flare ups'. This is when muscles in the wall of airways tighten and swell, narrowing the airway itself. This, in combination with the production of mucus can block the airway to varying degrees. Resulting in symptoms such as coughing, wheezing, tightness in the chest and shortness of breath, making it extremely difficult to breathe.

## 2. PROBLEM STATEMENT

COVID-19, a chronic health condition prevalent in children can be characterized by breathlessness, chest tightness and coughing. ACOVID-19 attack can be triggered by a variety of factors including environmental conditions, intense physical activity, humidity and dust. In the United States, as of February 2020, 17 million children (10%) were reported to be suffering from COVID-192. This condition is generally more prevalent among adolescents in the age group of 41-67. Due to the high prevalence of COVID-19 in children and the difficulty involved in diagnosing the condition it becomes imperative to come up with technological solutions for continuous care and management of patients with this chronic disease.

## 3. OBJECTIVE

The objective of this project is to develop a system to analyse the trigger factor of COVID-19 and a device that can use by COVID-19 patients, which can perform multiple functions that enable a physician to monitor the patient's condition and to provide continuous care. The different functionalities in a point of care device that are considered vital in caring for patients with COVID-19 are discussed below. The idea behind this work is to develop a point of care device which can ideally be integrated with an alarm to remind the patients to take their medication and also a memory which stores all the test results, patients' perception of their condition and the regularity with which the inhaler is used so that the physicians can later use these data to accurately assess the condition of the patient.

## 4. MOTIVATION

COVID-19 is one of the widespread chronic diseases. Rising prevalence increases the burden of personal disease management, financial expenditures and workload, both on sides of patients and healthcare systems. According to the World Health Organization COVID-19 is a serious public health problem with over 100 million sufferers worldwide. It continues to be one of the major causes of hospitalization of children in many countries. COVID-19 is the leading cause of absenteeism from school and the third leading cause of work loss

## 5. EXISTING SYSTEM

The implementation of a portable system for people suffering from Broncho pulmonary diseases is proposed, which is a combination of a patient activity sensor, a Temperature sensor andan air pollution sensor. The sensors are installed in a compact case, which is fixed on the patient's body and transmits data to the central station, which can be implemented on the basis of a smartphone. Such a device is designed to inform the patient about the need to reduce activity in the case of determining a high degree of air pollution to prevent an attack of the disease. It is also proposed to form a map of air pollution by analyzing data obtained from a variety of similar devices.

This willdetermine areas of excessive air pollution and inform patientsabout the need to refrain from visiting selected areas. The disadvantage of this system is that it is not able to determine the current state of health of the patient. However, the proposed device provides undoubtedly important information and is promising in the case of the integration of such sensors into a comprehensive patient monitoring system.

## 6. LITERTURE REVIEW

Comparative replication and immune activation profiles of SARS-CoV-2 and SARS-CoV in human lungs: an ex vivo study with implications for the pathogenesis of COVID-19

**Authors: HinChu,JasperFuk-Woo Chan Published in: IEEE 2020**

Severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) is an emerging coronavirus that has resulted in nearly 1,000,000 laboratory-confirmed cases including over 50,000 deaths. Although SARS-

CoV-2 and SARS-CoV share a number of common clinical manifestations, SARS-CoV-2 appears to be highly efficient in person-to-person transmission and frequently cause asymptomatic infections. However, the underlying mechanism that confers these viral characteristics on high transmissibility and asymptomatic infection remain incompletely understood. Methods: We comprehensively investigated the replication, cell tropism, and immune activation profile of SARS-CoV-2 infection in human lung tissues with SARS-CoV included as a comparison. Results: SARS-CoV-2 infected and replicated in human lung tissues more efficiently than that of SARS-CoV. Within the 48-hour interval, SARS-CoV-2 generated 3.20 folds more infectious virus particles than that of SARS-CoV from the infected lung tissues ($P<0.024$). SARS-CoV-2 and SARS-CoV were similar in cell tropism, with both targeting types I and II pneumocytes, and alveolar macrophages. Importantly, despite the more efficient virus replication, SARS-CoV-2 did not significantly induce types I, II, or III interferons in the infected human lung tissues. In addition, while SARS-CoV infection upregulated the expression of 11 out of 13 (84.62%) representative pro-inflammatory cytokines/chemokines, SARS-CoV-2 infection only upregulated 5 of these 13 (38.46%) key inflammatory mediators despite replicating more efficiently. Conclusions: Our study provided the first quantitative data on the comparative replication capacity and immune activation profile of SARS-CoV-2 and SARS-CoV infection in human lung tissues. Our results provided important insights on the pathogenesis, high transmissibility, and asymptomatic infection of SARS-CoV-2.

Potential applications of wearable sensors in closed-loop management of STEMI patients during pandemics

**Authors: Xiaorong Ding**

**Published in: 2020May 2020IEEE Reviews in Biomedical Engineering**

Coronavirus disease 2019 (COVID-19) has emerged as a pandemic with serious clinical manifestations including death. A pandemic at the large-scale like COVID-19 places extraordinary demands on the world's health systems, dramatically devastates vulnerable populations, and critically threatens the global communities in an unprecedented way. While tremendous efforts at the frontline are placed on detecting the virus, providing treatments and developing vaccines, it is also critically important to examine the technologies and systems for tackling disease emergence, arresting its spread and especially the strategy for diseases prevention. The objective of this article is to review enabling technologies and systems with various application scenarios for handling the COVID-19 crisis. The article will focus specifically on 1) wearable devices suitable for monitoring the populations at risk and those in quarantine, both for evaluating the health status of caregivers and management personnel, and for facilitating triage processes for admission to hospitals 2) unobtrusive sensing systems for detecting the disease and for monitoring patients with relatively mild symptoms who's clinical situation could suddenly worsen in improvised hospitals and 3) telehealth technologies for the remote monitoring and diagnosis of COVID-19 and related diseases. Finally, further challenges and opportunities for future directions of development are highlighted.

Prospects for Designing a Portable System for Monitoring of the Patient's Condition with

**COVID-19**

**Authors: Ivan V. Semernik, Alexander V. Dem'yanenko**

**Published in: 2019 IEEE**

In this article the prospects and possibilities for creating an individual wearable system for monitoring the condition of a patient suffering from COVID-19 and preventing attacks of the disease are discussed. As the basic method of determining the condition of the patient is considered the technique for determining the transmission coefficient of a certain frequency microwave signal through the chest. The proposed method is non-invasive and harmless and can be used for patients of all age groups.

Detection and Monitoring of COVID-19 Trigger Factor using Zigbee Authors: Miss. AnumehaLal , Mr. Girish A. Kulkarni

**Published in: International Advanced Research Journal in Science, Engineering and Technology,Vol. 3, Issue 7, July 2016**

COVID-19 is one of the widespread chronic diseases. Firstly, the medical background of COVID-19 is given. Pathology and symptoms are presented. COVID-19 is a chronic condition that mostly affects adolescents. It is a condition that requires continuous monitoring of the symptoms in order to provide an effective course of treatment. It also requires a strict adherence to medication prescribed by the physician.

However, the aim of this study is to develop a system, which is based on a periodical data collected by the different sensors. There is no cure for COVID-19. Symptoms can be prevented by monitoring factors which can trigger COVID-19 attack. So it is very much needed that there should be a system which can monitor air parameter on regular basis and warn the patient when these factor can trigger their COVID-19 attack.

Wireless sensor networks in monitoring of COVID-19 Authors: DinkoOletic
**Published in: IJRSE 2013**
COVID-19 is one of the widespread chronic diseases. Rising prevalence increases the burden of personal disease management, financial expenditures and workload, both on sides of patients and healthcare systems. Firstly, the medical background of COVID-19 is given. Pathology and symptoms are presented. Afterwards, the problem of persistent COVID-19 management is introduced with a short overview of traditional disease management techniques. A review on approaches to COVID-19telemonitoring is made. Effectiveness of home peakflowmetry is analysed. Employment of low power wireless sensor networks (WSN) paired with smartphone technologies is reviewed as a novel COVID-19 management tool. Using the technology, the aim is to retain the disease in a controlled state with minimal effort, invasivenessandcost,andassesspatient's condition objectively. WSN-s for sensing of both COVID-19 triggers in the environment, and continuous monitoring of physiological functions, in particular respiratory function are reviewed. Sensing modalities for acquiring respiratory function are presented. Signal acquisition prerequisites and signal processing of respiratory sounds are reviewed. Focus is put on low-power continuous wheeze detectiontechniques. At the end, research challenges for further studiesare identified.
Monitoring the patient with COVID-19: An evidence-based approach Authors: Harold S. Nelson, MD
**Published in: Apr 17, 2000**

The monitoring of symptoms, airflow obstruction, and exacerbations is essential to COVID-19 management. Patients who practice self-monitoring in conjunction with use of a written action plan and regular medical review have significantly fewer hospitalizations, emergency department visits, and lost time from work. Either symptom monitoring or peak expiratory flow monitoring is satisfactory, provided the results are interpreted with reference to the patient's own baseline COVID-19 status. Regular monitoring by physicians also improves health outcomes for patients, provided the physician is systematic and monitors control, medications, and skills at regular intervals. Additional monitoring tools are under evaluation, and these include measures of airway responsiveness, airway inflammation, and Internet- based monitoring systems. Administrators need to monitor the quality and cost of care, as well as compliance with national management guidelines. Assessment of the hospitalization rate and regular audit may achieve these aims in the hospital setting. The best way to assess and monitor COVID-19 in primary care remains an unresolved yet crucial issue because primary care physicians manage the vast burden of illness caused by COVID-19. Monitoring COVID-19 outcomes is an essential step toward the successful implementation of national guidelines for the management of COVID-19.

## 7. METHODOLOGY

The COVID-19 Monitoring System is designed around a microcontroller for gathering, sending and receiving information from different sensors and external servers. The aim of the architecture design is to provide an easier access to information and services, better patient healthcare services, transparent and efficient use of healthcare resources, and a fast response by the hospital side in case of COVID-19 attack. Symptoms can be prevented by monitoring factors which can trigger COVID-19 attack. So it is very much needed that there should be a system which can monitor air parameter on regular basis and warn the patient when these factor can trigger their COVID-19 attack.A portable system for non-invasive diagnosis of Broncho pulmonary diseases and continuous monitoring of the patient's condition is a combination of two compact modules radiating and receiving, located on the side of the chest and back, respectively. The position of each module is fixed and does not change over time. The fixation point of the modules is determined based on the individual characteristics of the patient's body.

The COVID-19 patient can monitor his own condition at any time, though this he can save the life.
Sensor technology is to be used for monitoring the COVID-19 patient condition easily. Temperature: For the temperature we have 2 domains, the cold air and the hot air. The patients are mostly exposed to cold air after exercising. It is advised to avoid temperatures below 18°C. The temperature of 15°C and below is considered risky. The hot air is by itself dangerous. It also helps contain pollen and air pollution. It is advised to avoid temperatures above 27°C. The temperature of 30°C and above is considered also risky for the COVID-19tic patients.
The system uses microcontroller. A SPO2 sensor ,and Temperature sensor are connected to the Microcontroller. The temperature sensor gives the temperature value in degree Celsius. To measure the heart rate, the heart beat/pulse is detected and the number of pulses for one minute is counted to get the beats per minute. Light (using an LED) is passed from one side of the finger and the intensity of light received on the other side is measured (using an LDR).The GPS and Nodemcu modules are interfaced with the microcontroller. The GPS module finds out the latitude and longitude of the patient. The temperature and Spo2 values are measured and compared with a configurable threshold to be classified as "low", "normal" or "high". The Nodemcu module is used to send a message to the doctor's mobile in case of emergencies. The message contains the temperature, Spo2 values and the patient's latitude and longitude. The doctor can thus take immediate action with the help of this alert system and if in case of changing the position of covid Patient also detect by using GPS value and send alert to the concern persons.
While the symptoms described in the above are typical indicators of COVID-19, not all people suffer in the same way or the same combination of these symptoms. Research shows that some people may have the coughing, wheezing, chest tightness and shortness of breath, while others may have a different combination of the symptoms at different times [10]. Sometimes during an attack, some of the symptoms will be worse than others, and even vary from one episode to another. Some are mild and generally more common, while some are more serious. The life-threatening attacks may be less common, but they also may last longer in length and require emergency medical care

The measurement results are transmitted via a wireless interface to a PC, tablet or smartphone and are recorded in an electronic diary or, for example, can be used to train a neural network. This will allow for the accumulation of data to adapt the program of processing results for a specific patient and more accurately monitor the change in its health. When the measurement results exceed the set limits, an alarm is generated, which is displayed as a message on the screen of the mobile device and can be sent to the email address of the medical center.
The described individual system can be useful for continuous express monitoring of the condition of a person suffering from COVID-19 during the day and warning him about the need to take medicine. In addition, it can be useful in medical institutions for monitoring the condition of a patient in hospital, and monitoring the effects of drugs.

- **SYSTEM REQUIREMENTS**

**HARDWARE REQUIREMENTS**

- Arduino NANO
- SPO2 Sensor
- Temperature Sensor
- ESP 8266
- Load cell
- Power Supply

Ultrasonic sensor

**SOFTWARE REQUIREMENTS**

 Embedded C
 Arduino Suite

Functional Requirements:

System should scan & Detect the COVID-19 patient condition
System should measure the heart beat & temperature.
System should self-monitoring allows the early detection of exacerbations.
System should automatically investigate reasons for poor control.

## 8. ARDUINO NANO (ATMEGA 328P)

**Fig. Arduino NANO**

The Arduino NANO is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino. The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards and other circuits. The board has 14 Digital pins, 6 Analog pins, and programmable with the Arduino IDE (Integrated Development Environment) via a type B USB cable. It can be powered by a USB cable or by an external 9 volt battery, though it accepts voltages between 7 and 20 volts. It is also similar to the Arduino Nano and Leonardo. "Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform.
The ATmega328 on the Arduino Uno comes pre- programmed with a bootloader that allows uploading new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol.[1] The Uno also differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2 programmed as a USB-to-serial converter.

**TECHNICAL SPECIFICATIONS**

- Microcontroller: Microchip ATmega328P
- Operating Voltage: 5 Volts
- Input Voltage: 7 to 20 Volts
- Digital I/O Pins: 14 (of which 6 provide PWM output)
- Analog Input Pins: 6
- DC Current per I/O Pin: 20 mA
- DC Current for 3.3V Pin: 50 mA
- Flash Memory: 32 KB of which 0.5 KB used by boot loader
- SRAM: 2 KB
- EEPROM: 1 KB
- Clock Speed: 16 MHz
- Length: 68.6 mm
- Width: 53.4 mm

- Weight: 25 g

Vin is the input voltage of the board, and it is used when an external power source is used from 7V to 12V.

- 5V is the regulated power supply voltage of the nano board and it is used to give the supply to the board as well as components.
- 3.3V is the minimum voltage which is generated from the voltage regulator on the board.
- GND is the ground pin of the board
- RST Pin( Reset): This pin is used to reset the microcontroller.
- I/O Pins (Digital Pins from D0 – D13): These pins are used as an i/p otherwise o/p pins. 0V & 5V
- Serial Pins (Tx, Rx): These pins are used to transmit & receive TTL serial data.
- External Interrupts (2, 3): These pins are used to activate an interrupt.
- PWM (3, 5, 6, 9, 11): These pins are used to provide 8-bit of PWM output.
- SPI (10, 11, 12, & 13): These pins are used for supporting SPI communication.
- Inbuilt LED (13): This pin is used to activate the LED.
- IIC (A4, A5): These pins are used for supporting TWI communication.
- AREF: This pin is used to give reference voltage to the input voltage

**ARDUINO SOFTWARE**

**INTRODUCTION TO THE ARDUINO IDE**

- The Arduino is a single-board microcontroller solution for many DIY projects, we will look at the Integrated Development Environment, or IDE, that is used to program it. Once the installer has downloaded, go ahead and install the IDE. Arduino IDE is an open source software that is mainly used for writing and compiling the code into the Arduino Module.
- It is an official Arduino software, making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process.

**DOWNLOAD THE IDE**

First, you must download the IDE and install it. Start by visiting Arduino‘s software page. The IDE is available for most common operating systems, including Windows, Mac OS X, and Linux, so be sure to download the correct version for your OS. If you are using Windows 7 or older, do not download the Windows app version, as this requires Windows 8.1 or Windows 10.

**The ARDUINO IDE**

The Arduino IDE is incredibly minimalistic, yet it provides a near-complete environment for most Arduino-based projects. The middle section of the IDE is a simple text editor that where you can enter the program code. The bottom section of the IDE is dedicated to an output window that is used to see the status of the compilation, how much memory has been used, any errors that were found in the program, and various other useful messages.

**Arduino program dumping window**

Projects made using the Arduino are called sketches, and such sketches are usually written in a cut-down version of C++ (a number of C++ features are not included). Because programming a microcontroller is somewhat different from programming a computer, there are a number of device-specific libraries (e.g., changing pin modes, output data on pins, reading analog values, and timers).

This sometimes confuses users who think Arduino is programmed in an ―Arduino language.‖ However, the Arduino is, in fact, programmed in C++. It just uses unique libraries for the device.

**THE 6 BUTTONS**

While more advanced projects will take advantage of the built-in tools in the IDE, most projects will rely on the six buttons found below the menu bar.

**The button bar**

1. The check mark is used to verify your code. Click this once you have written your code.
2. The arrow uploads your code to the Arduino to run.
3. The dotted paper will create a new file.
4. The upward arrow is used to open an existing Arduino project.
5. The downward arrow is used to save the current file.
6. The far right button is a serial monitor, which is useful for sending data from the Arduino to the PC for debugging purposes.

**ARDUINO HARDWARE**

Arduino is open-source hardware. The hardware reference designs are distributed under a Creative Commons Attribution Share-Alike 2.5 license and are available on the Arduino website. Layout and production files for some versions of the hardware are also available. The boards use single or double-row pins or female headers that facilitate connections for programming and incorporation into other circuits. These may connect with add-on modules termed shields. Multiple and possibly stacked shields may be individually addressable via an $I^2C$ serial bus. Most boards include a 5 V linear regulator and a 16 MHz crystal oscillator or ceramic resonator. Some designs, such as the Lily Pad, run at 8 MHz and dispense with the onboard voltage regulator due to specific form-factor restrictions. Most Arduino boards consist of an Atmel 8-bit AVR microcontroller (ATmega8, ATmega168, ATmega328, ATmega1280, ATmega2560) with varying amounts of flash memory, pins, and features.

Arduino microcontrollers are pre-programmed with a boot loader that simplifies uploading of programs to the on-chip flash memory. The default bootloader of the Arduino UNO is the opti boot boot loader. Boards are loaded with program code via a serial connection to another computer. Some serial Arduino boards contain a level shifter circuit to convert between RS-232 logic levels and transistor–transistor logic (TTL) level signals. Current Arduino boards are programmed via Universal Serial Bus (USB), implemented using USB-to-serial adapter chips such as the FTDI FT232. Some boards, such as later-model Uno boards, substitute the FTDI chip with a separate AVR chip containing USB-to-serial firmware, which is reprogrammable via its own ICSP header. Other variants, such as the Arduino Mini and the unofficial Boarduino, use a detachable USB-to- serial adapter board or cable, Bluetooth or other methods. When used with traditional microcontroller tools, instead of the Arduino IDE, standard AVR in-system programming (ISP) programming is used.

The Arduino board exposes most of the microcontroller's I/O pins for use by other circuits. Uno provide 14 digital I/O pins, six of which can produce pulse-width modulated signals, and six analog inputs, which can also be used as six digital I/O pins. These pins are on the top of the board, via female 0.1-inch (2.54 mm) headers. Several plug-in application shields are also commercially available. The Arduino Nano, and Arduino-compatible Bare Bones Board and Boarduino boards may provide male header pins on the underside of the board that can plug into solderless breadboards.

There are several I/O digital and analog pins placed on the board which operates at 5V. These pins come with standard operating ratings ranging between 20mA to 40mA. Internal pullup resistors are used in the board that limits the current exceeding from the given operating conditions. However, too much increase in current makes these resisters useless and damages the device.

**POWER SUPPLY**

The transformer 230Volts will be stepped down to 12-0-12 one side of the 12V is given to the 7805 and Lm317.In this project the microcontroller requires +5V power supply. The design description of power supply is given below.

The +5 Volt and 3.8V power supply is based on the commercial 7805 & Lm317 voltage regulator IC. This IC contains all the circuitry needed to accept any input voltage from 8 to 18 volts and produce a steady +5 volt & 3.8volt output, accurate to within 5% (0.25 volt). It also contains current-limiting circuitry and thermal overload protection, so that the IC won't be damaged in case of excessive load current; it will reduce its output voltage instead.

The 1000µf capacitor serves as a "reservoir" which maintains a reasonable input voltage to the 7805 throughout the entire cycle of the ac line voltage. The bridge rectifier (WM04) keep recharging the reservoir capacitor on alternate half-cycles of the line voltage, and the capacitor is quite capable of sustaining any reasonable load in between charging pulses.

The LED and its series resistor (220ohm) serve as a pilot light to indicate when the power supply is on and also helps to the reservoir capacitor is completely discharged after power is turned off. Then I know it's safe to remove or install components for the next experiment

## 9. TEMPERATURE SENSOR

The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 does not require any external calibration or trimming to provide typical accuracies.

**GENERAL DESCRIPTION**

The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in° Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient Centigrade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of ±1⁄4°Cat room temperature and ±3⁄4°C over a full −55 to +150°Ctemperature range. Low cost is assured by trimming and calibration at the wafer level.

The LM35's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with single power supplies, or with plus and minus supplies. As it draws only 60 µA from its supply, it has very low self-heating, less than 0.1°C in still air. The LM35 is rated to operate over a −55° to +150°C temperature range, while the LM35C is rated for a −40° to +110°C range (−10° with improved accuracy). The LM35 series is available packaged in hermetic TO-46 transistor packages, while theLM35C, LM35CA, and LM35D are also available in the plastic TO-92 transistor package. The LM35D is also available in an 8-lead surface mount small outline package and aplastic TO-220 package.

4.5.2 FEATURES

- Calibrated directly in ° Celsius (Centigrade)
- Linear + 10.0 mV/°C scale factor
- 0.5°C accuracy guarantee able (at +25°C)
- Rated for full −55° to +150°C range
- Suitable for remote applications
- Low cost due to wafer-level trimming

Temperature Sensor

- Less than 60µA current drain
- Operates from 4 to 30 volts
- Low self-heating, 0.08°C in still air
- Low impedance output, 0.1 W for 1 mA load

**APPLICATIONS**

The LM35 can be applied easily in the same way as other integrated-circuit temperature sensors. It can be glued or cemented to a surface and its temperature will be within about 0.01§C of the surface temperature. This presumes that the ambient air temperature is almost the same as the surface temperature; if the air temperature were much higher or lower than the surface temperature, the actual temperature of the LM35 die would be at an inter- mediate temperature between the surface temperature and the air temperature. This is especially true for the TO-92 plastic package, where the copper leads are the principal thermal path to carry heat into the device, so its temperature might be closer to the air temperature than to the surface temperature. To minimize this problem, be sure that the wiring to the LM35, as it leaves the device, is held at the same temperature as the surface of interest. The easiest way to do this is to cover up these wires with a bead of epoxy which will insure that the leads and wires are all at the same temperature as the surface, and that the LM35 die's temperature will not be affected by the air temperature.
The TO-46 metal package can also be soldered to a metal surface or pipe without damage. Of course, in that case the Vb terminal of the circuit will be grounded to that metal. Alternatively, the LM35 can be mounted inside a sealed-end metal tube, and can then be dipped into a bath or screwed into a threaded hole in a tank. As with any IC, the LM35 and accompanying wiring and circuits must be kept insulated and dry, to avoid leakage and corrosion. This is especially true if the circuit may operate at cold temperatures where condensation can occur. Printed-circuit coatings and vanishes such as Humiseal and epoxy paints or dips are often used to insure that moisture cannot corrode the LM35 or its connections. These devices are sometimes soldered to a small light- weight heat fin, to decrease the thermal time constant and speed up the response in slowly-moving air. On the other hand, a small thermal mass may be added to the sensor, to give the steadiest reading despite small deviations in the air temperature.

**10. ESP8266 WiFi Module**

ESP8266 is Wi-Fi enabled system on chip (SoC) module developed by Espressif system. It is mostly used for development of IoT (Internet of Things) embedded applications.ESP8266 comes with capabilities of

- 2.4 GHz Wi-Fi (802.11 b/g/n, supporting WPA/WPA2),
- general-purpose input/output (16 GPIO),
- Inter-Integrated Circuit ($I^2C$) serial communication protocol,
- analog-to-digital conversion (10-bit ADC)
- Serial Peripheral Interface (SPI) serial communication protocol,
- $I^2S$ (Inter-IC Sound) interfaces with DMA(Direct Memory Access) (sharing pins with GPIO),
- UART (on dedicated pins, plus a transmit-only UART can be enabled on GPIO2), and
- pulse-width modulation (PWM).
- It employs a 32-bit RISC CPU based on the TensilicaXtensa L106 running at 80 MHz (or overclocked to 160 MHz). It has a 64 KB boot ROM, 64 KB instruction RAM and 96 KB data RAM. External flash memory can be accessed through SPI.

To communicate with the ESP8266 module, microcontroller needs to use set of AT commands. Microcontroller communicates with ESP8266-01 module using UART having specified Baud rate.
There are many third-party manufacturers that produce different modules based on this chip. So, the module comes with different pin availability options like,
ESP-01 comes with 8 pins (2 GPIO pins) – PCB trace antenna. (shown in above figure)

ESP-02 comes with 8 pins, (3 GPIO pins) – U-FL antenna connector.
ESP-03 comes with 14 pins, (7 GPIO pins) – Ceramic antenna.
ESP-04 comes with 14 pins, (7 GPIO pins) – No ant.

**ESP8266-01 Module Pins**
3V3: - 3.3 V Power Pin.
GND: - Ground Pin.
RST: - Active Low Reset Pin.
EN: - Active High Enable Pin.
TX: - Serial Transmit Pin of UART.
RX: - Serial Receive Pin of UART.

GPIO0 & GPIO2: - General Purpose I/O Pins. These pins decide what mode (boot or normal) the module starts up in. It also decides whether the TX/RX pins are used for Programming the module or for serial I/O purpose.
To program the module using UART, Connect GPIO0 to ground and GPIO2 to VCC or leave it open. To use UART for normal Serial I/O leave both the pins open (neither VCC nor Ground).
Now let's connect ESP8266 module to computer with RS232 standard serial port (using USB to Serial converter in case of laptop) as shown in below figure.

**ESP8266 Module Serial Connection with PC**
Note that, to put ESP8266 in flash mode, make connections as per above figure (in between ESP8266 and USB to Serial converter) and then only connect it to PC/laptop. Do not forget to connect GPIO0 pin to ground Then click on START tab in ESP8266 DOWNLOAD TOOL, and wait till it finishes. After finishing flash process, disconnect ESP8266 module from PC/laptop and remove ground connection at GPIO0 pin.

## 11. NODEMCU

NodeMCU is an open source LUA based firmware developed for ESP8266 wifi chip. By exploring functionality with ESP8266 chip, NodeMCU firmware comes with ESP8266 Development board/kit i.e. NodeMCU Development board. NodeMCUDev Kit/board consist of ESP8266 wifi enabled chip. The ESP8266 is a low-cost Wi-Fi chip developed by Espressif Systems with TCP/IP protocol. For more information about ESP8266, you can refer ESP8266 WiFi Module. There is Version2 (V2) available for NodeMCUDev Kit i.e. NodeMCU Development Board v1.0 (Version2), which usually comes in black colored PCB. NodeMCUDev Kit has Arduino like Analog (i.e. A0) and Digital (D0-D8) pins on its board.
It supports serial communication protocols i.e. UART, SPI, I2C etc.
Using such serial protocols we can connect it with serial devices like I2C enabled LCD display, Magnetometer HMC5883, MPU-6050 Gyro meter + Accelerometer, RTC chips, GPS modules, touch screen displays, SD cards etc.

**Pin Definition**

| Label | GPIO | Input | Output | Notes |
|---|---|---|---|---|
| **D0** | **GPIO16** | no interrupt | no PWM or I2C support | HIGH at boot used to wake up from deep sleep |
| **D1** | **GPIO5** | OK | OK | often used as SCL (I2C) |
| **D2** | **GPIO4** | OK | OK | often used as SDA (I2C) |
| **D3** | **GPIO0** | pulled up | OK | connected to FLASH button, boot fails if pulled LOW |
| **D4** | **GPIO2** | pulled up | OK | HIGH at boot connected to on-board LED, boot fails if pulled LOW |
| **D5** | **GPIO14** | OK | OK | SPI (SCLK) |
| **D6** | **GPIO12** | OK | OK | SPI (MISO) |
| **D7** | **GPIO13** | OK | OK | SPI (MOSI) |
| **D8** | **GPIO15** | pulled to GND | OK | SPI (CS) Boot fails if pulled HIGH |
| **RX** | **GPIO3** | OK | RX pin | HIGH at boot |
| **TX** | **GPIO1** | TX pin | OK | HIGH at boot debug output at boot, boot fails if pulled LOW |
| **A0** | **ADC0** | Analog Input | X | |

**How to start with NodeMCU?**

NodeMCU Development board is featured with wifi capability, analog pin, digital pins and serial communication protocols.
To get start with using NodeMCU for IoT applications first we need to know about how to write/download NodeMCU firmware in NodeMCU Development Boards. And before that where this NodeMCU firmware will get as per our requirement.
There is online NodeMCU custom builds available using which we can easily get our custom NodeMCU firmware as per our requirement.

## 12. TELEGRAM BOTS

**Bots: An introduction for developers**
Bots are third-party applications that run inside Telegram. Users can interact with bots by sending them messages, commands and inline requests. You control your bots using HTTPS requests to our Bot API.

**What can I do with bots?**

A chat with @TechCrunchBot also showing search results from the @gif inline-bot
To name just a few things, you could use bots to:
Get customized notifications and news. A bot can act as a smart newspaper, sending you relevant content as soon as it's published. Integrate with other services. A bot can enrich Telegram chats with content from external services. Gmail Bot, GIF bot, IMDB bot, Wiki bot, Music bot, Youtube bot, GitHub bot
Accept payments from Telegram users. A bot can offer paid services or work as a virtual storefront.

Demo Shop Bot Create custom tools. A bot may provide you with alerts, weather forecasts, translations, formatting or other services. Markdown bot, Sticker bot, Vote bot, Like bot Build single- and multiplayer games. A bot can offer rich HTML5 experiences, from simple arcades and puzzles to 3D-shooters and real-time strategy games. GameBot, Gamee Build social services. A bot could connect people looking for conversation partners based on common interests or proximity. Do virtually anything else. Except for dishes — bots are terrible at doing the dishes.

**How do bots work?**
At the core, Telegram Bots are special accounts that do not require an additional phone number to set up. Users can interact with bots in two ways:
Send messages and commands to bots by opening a chat with them or by adding them to groups. This is useful for chat bots or news bots like the official TechCrunch bot.
Send requests directly from the input field by typing the bot's @username and a query. This allows sending content from inline bots directly into any chat, group or channel.
Messages, commands and requests sent by users are passed to the software running on your servers. Our intermediary server handles all encryption and communication with the Telegram API for you. You communicate with this server via a simple HTTPS-interface that offers a simplified version of the Telegram API. We call that interface our Bot API.

**How do I create a bot?**

The Botfather.
There's a… bot for that. Just talk to BotFather (described below) and follow a few simple steps. Once you've created a bot and received your authorization token, head down to the Bot API manual to see what you can teach your bot to do.

## 13. SPO2 SENSOR (MAX30100)

The MAX30100 is an integrated pulse oximetry and heart rate monitor sensor solution. 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.
**Features**
Complete PulseOximeter 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

14. FINAL Module Arduino Code

```
#include <TinyGPS++.h>

#include <SoftwareSerial.h>

/* Create object named bt of the class SoftwareSerial */

SoftwareSerial GPS_SoftSerial(6, 5);/* (Rx, Tx) */

/* Create an object named gps of the class TinyGPSPlus */

TinyGPSPlus gps;

long duration;

int distance;


#include <Wire.h>

#include "MAX30100_PulseOximeter.h"


#define REPORTING_PERIOD_MS     2000

// unsigned long temp=0;


int triPin = 8;

int echoPin = 9;

//long microsecondsToCentimeters(long microseconds);

PulseOximeter pox;

 uint32_t tsLastReport = 0;
```

```
float tempc;
float vout;
const int sensor = A3;
const int SW=4;
int pins=12;
int buzz=11;
// Callback (registered below) fired when a pulse is detected
void onBeatDetected()
{
    Serial.println("Beat!");
}

void setup()
{
  //Serial.begin(9600); /* Define baud rate for serial communication */
  GPS_SoftSerial.begin(9600);
    Serial.begin(9600);
 pinMode(SW,INPUT_PULLUP);
 pinMode(pins,INPUT);
 pinMode(buzz,OUTPUT);
 pinMode(triPin,OUTPUT);
pinMode(echoPin,INPUT);

 digitalWrite(buzz,LOW);
 Serial.print("Initializing pulse oximeter..");

    if (!pox.begin()) {
        Serial.println("FAILED");
        for(;;);
```

```
  } else {
    Serial.println("SUCCESS");
    Serial.println("COVID TESTING");
  }


  pox.setOnBeatDetectedCallback(onBeatDetected);


}

void loop()
{

//if(digitalRead(pins)==LOW)
//{
//  Serial.println("$Band Removed#");
//  delay(1000);
//
//}
////else
////{
////
////Serial.print("Initializing pulse oximeter..");
////
////    if (!pox.begin()) {
////        Serial.println("FAILED");
////        for(;;);
////    } else {
```

```
////        Serial.println("SUCCESS");
////        Serial.println("COVID TESTING");
////    }
////
////
////    pox.setOnBeatDetectedCallback(onBeatDetected);
  int i=0;
  while(i<20000)
  {
    pox.update();
    if (millis() - tsLastReport > REPORTING_PERIOD_MS) {
        Serial.print("$ HEARTRATE: ");
        Serial.print(pox.getHeartRate());
        //Serial.println("mm Hg");
        Serial.println("#");
       Serial.print("$ SPO2:");
        Serial.print(pox.getSpO2());
        Serial.println("#");


        if(pox.getHeartRate()>120)
        {
          Serial.println("$ HEARTBEAT IS HIGH #");
         // delay(2000);
        }
        if(pox.getSpO2()<90)
        {
          Serial.println("$ OXIDE_LEVEL IS LOW #");
         // delay(2000);
        }
```

```
        tsLastReport = millis();
    }
    i++;
}
//HEART_BEAT_MONITOR();
uv_reading1();
TEMP_MONITOR();

serialEvent();


}
void TEMP_MONITOR()
{
 // serialEvent();
   // tempc=vout;
    vout=analogRead(sensor);
    vout=(vout*500)/1023;
    tempc=vout;

    Serial.print("$TEMP:");
    Serial.print(tempc);
    Serial.println("C#");
    delay(2000);

    if(tempc>40)
    {
      Serial.println("$MORE TEMPERATURE#");
    }
    else if (tempc<35)
```

```
    {
      Serial.println("$LESS TEMPERATURE#");
    }
    else
    {
      Serial.println("$NORMAL TEMPERATURE#");
    }
    setup();
}

void serialEvent()
{


  if(digitalRead(SW)==LOW)
  {
    Serial.println("EMERGENCY...");
    Serial.println("$EMERGENCY...#");
    delay(2000);
//    Manual();
    GPS();

   }




  }
void GPS()
{
```

```
  smartDelay(1000); /* Generate precise delay of 1ms */
        unsigned long start;
        double lat_val, lng_val, alt_m_val;
       // uint8_t hr_val, min_val, sec_val;
        bool loc_valid, alt_valid, time_valid;
        lat_val = gps.location.lat(); /* Get latitude data */
        loc_valid = gps.location.isValid(); /* Check if valid location data is available */
        lng_val = gps.location.lng(); /* Get longtitude data */
        alt_m_val = gps.altitude.meters();  /* Get altitude data in meters */
        alt_valid = gps.altitude.isValid(); /* Check if valid altitude data is available */
//        hr_val = gps.time.hour(); /* Get hour */
//        min_val = gps.time.minute();  /* Get minutes */
//        sec_val = gps.time.second();  /* Get seconds */
        time_valid = gps.time.isValid();  /* Check if valid time data is available */
        if (!loc_valid)
        {
          Serial.print("Latitude : ");
          Serial.println("*****");
          Serial.print("Longitude : ");
          Serial.println("*****");
        }
        else
        {
        //  DegMinSec(lat_val);
          Serial.print("Latitude in Decimal Degrees : ");
          Serial.println(lat_val, 8);

        //  DegMinSec(lng_val); /* Convert the decimal degree value into degrees minutes seconds form */
          Serial.print("Longitude in Decimal Degrees : ");
```

```
        Serial.println(lng_val, 8);



String one = "PERSON LOCATION AT:https://www.google.com/maps/?q=";
  String two = "," ;
  String message = one +lat_val +two + lng_val;

  int str_len = message.length() + 1;
  char textmessage[str_len];
  message.toCharArray(textmessage,str_len);
  Serial.println(textmessage);
  delay(3000); // wait for 3 seconds
    Serial.println(textmessage);
      Serial.print("$T");
      Serial.print(textmessage);
      Serial.println("#");
      }
      if (!alt_valid)
      {
      //  Serial.print("Altitude : ");
        Serial.println("*****");
      }
      else
      {
     //   Serial.print("Altitude : ");
        Serial.println(alt_m_val, 6);
      }
      if (!time_valid)
```

```
        {
        //  Serial.print("Time : ");
          Serial.println("*****");
        }
        else
        {
//          char time_string[32];
//          sprintf(time_string, "Time : %02d/%02d/%02d \n", hr_val, min_val, sec_val);
//         // Serial.print(time_string);
        }


}

static void smartDelay(unsigned long ms)
{
  unsigned long start = millis();
  do
  {
    while (GPS_SoftSerial.available())  /* Encode data read from GPS while data is available on serial port
*/
      gps.encode(GPS_SoftSerial.read());
/* Encode basically is used to parse the string received by the GPS and to store it in a buffer so that
information can be extracted from it */
  } while (millis() - start < ms);
}
//void DegMinSec( double tot_val)   /* Convert data in decimal degrees into degrees minutes seconds
form */
//{
//  degree = (int)tot_val;
```

```
//  minutes = tot_val - degree;
//  seconds = 60 * minutes;
//  minutes = (int)seconds;
//  mins = (int)minutes;
//  seconds = seconds - minutes;
//  seconds = 60 * seconds;
//  secs = (int)seconds;
//}
void uv_reading1()
{
  digitalWrite(triPin, LOW);
  delayMicroseconds(2);
  digitalWrite(triPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(triPin,LOW);
  duration=pulseIn(echoPin, HIGH);
  distance=duration*0.034/2;
  Serial.print("Distances:");
  Serial.println(distance);
  delay(1000);
if(distance<20)
{
  Serial.println("object detected..");
  digitalWrite(buzz,HIGH);
  delay(2000);
  digitalWrite(buzz,LOW);
  delay(1000);
}
}
```

RESULTS IN TELEGRAM

**Results:**
Monitoring all the sensors like temperature, heartbeat and spo2 values.
Track the person location if any gps values change.
Send intimation to the doctor if any sensor value crosses the threshold.
Send intimation message to nearest police station when person move from quarantine location.

**Advantages**
Easy Implementation
Best Method for Detection of Covid Symptoms.
Early Detection of Disease
Cost Effective Method
Easy to track the person.
Maintain everyone Health record.

**CONCLUSION**
When the engineer will design a new product he must study everything related to his idea to avoid its errors and he must interest to introduce high quality, low cost, high accuracy, small size and easy to use product, then he should take customers opinions and suggestions to improve his skills in the next design.Thus, this paper discusses the prospects for introducing a portable system for diagnosing COVID-19. A block diagram of the system is presented on the basis of a cheap patient status sensor in combination with a portable computing device - a smartphone, tablet, etc. Such a structure will significantly reduce the cost of the device, which will contribute to its wider distribution. As the main method of state control, it is proposed to use the method of measuring the transmission coefficient of the microwave signal through the patient's chest. In this case, measurements are carried out at a single point, but for a long time, for example, when the device is continuously worn during the day. The advantages of using microwave technologies allow us to apply the proposed structure to monitor the condition of patients of all age groups, including young children. The integration of additional sensors for the patient's vital activity and the state of the environment, together with the use of modern IT technologies, will enable the creation of a comprehensive system for monitoring the patient's condition and informing him of the necessary actions in a timely manner.

REFERENCE

Y. N. Mi*et al.*, "Estimating instant case fatality rate of COVID-19 in China," *International Journal of Infectious Diseases,* 2020.
G. Onder*et al.*, "Case-fatality rate and characteristics of patients dying in relation to COVID-19 in Italy," *JAMA,* 2020.
*Coronavirus (COVID-19)*. [Online]. Available: https://covid19.who.int/
H. Chu *et al.*, "Comparative replication and immune activation profiles of SARS-CoV-2 and SARS-CoV in human lungs: an ex vivo study with implications for the pathogenesis of COVID-19," *Clinical Infectious Diseases,* 9 April 2020.
N. Ji*et al.*, "Potential applications of wearable sensors in closed-loop management of STEMI patients during pandemics (submitted)," in *The 42nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Montreal, Canada, 2020

PO and PSO Mapping for Project - 2020-21

**Justification for PO & PSO mapping for Project**

| **Project Title** | | |
|---|---|---|
| **PO** | **Levels 3/2/1** | **Justification** |
| **PO1** | 3 | Apply conceptual knowledge of Electronics and communication courses to choose the relevant technical domain. Using the basic knowledge of mathematical models, data structures, and algorithms involved in Machine learning techniques to solve the problem. |
| **PO2** | 2 | Reputed e-journals and IEEE research papers are referred to select the topic for the benefit of society. |
| **PO3** | 2 | Different algorithms, advantages, disadvantages of methods, comparison of results, and scope of the topic is presented with appropriate consideration for the public safety, and the cultural, societal, and environmental considerations. |
| **PO4** | 1 | Study of corona virus |
| **PO5** | 3 | By reviewing research literature, identifying and comparing different methods and approaches used in Airdrops and Machine learning techniques to attain substantiated conclusions for designing the solutions for the problem statement. |
| **PO6** | 3 | Implemented relevant engineering practices to formulate a better solution to manage safety, health (med kit airdrops), and environmental needs. |
| **PO7** | 2 | No data selling is done |

| | | |
|---|---|---|
| **PO8** | 1 | To ethically gather the datasets from social media for analysis/dataset creation committing to professional ethics and responsibilities. |
| **PO9** | 3 | To individually contribute and effectively work on the identified problem with proper coordination. |
| **PO10** | 3 | Demonst Srating and making effective presentations, justifying the cause and need of the project and progress of the project at each stage. |
| **PO11** | 2 | Demonstrated the knowledge of project management principles for planning and implementing the project in the given schedule. |
| **PO12** | 2 | Exploring the design of tracing of covid patients and tracing is done |
| **PSO1** | 3 | Current trend Arduino and processing have been used. |
| **PSO2** | 2 | Integrating both hardware and software technologies in this project design. |

**Budget Estimation**

| **Sl. No.** | **Particulars** | **Estimated Cost in Rs.** |
|---|---|---|
| 1 | Arduino Nano | 350 |
| 2 | Temperature Sensor | 200 |
| 3 | Heartbeat Sensor | 550 |
| 4 | Spo2 Sensor | 1100 |
| 5 | Ultrasonic Sensor | 250 |
| 6 | Buzzer | 15 |
| 7 | Nodemcu | 525 |
| 8 | GPS | 1000 |
| 9 | Wires | 300 |
| 10 | Others | 500 |
| 11 | | |
| 12 | | |
| Total | 4790 | |