Project Report

on

DESIGN OF IoT BASED SMART HEALTHCARE SYSTEM

Submitted in partial fulfillment of the requirement of

Jawaharlal Nehru Technological University for the award of degree of

Bachelor of Technology
in
Electronics and Communication Engineering

By

K. SAI PRASAD	19315A0431
M. PRAVAL GOUD	19315A0432
B. VILAS YADAV	19315A0435

UNDER THE GUIDANCE OF

Mr. S K SATYANARAYANA
Internal Guide, Assistant Professor, Dept. of E.C.E.

Dr. C.N. SUJATHA
Project Coordinator, Professor, Dept. of E.C.E

Department of Electronics and Communication Engineering

SREENIDHI INSTITUTE OF SCIENCE AND TECHNOLOGY

(Affiliated to JNT University, Hyderabad)

Yamnampet, Ghatkesar, Hyderabad – 501 301.

2022

Department of Electronics and Communication Engineering

SREENIDHI INSTITUTE OF SCIENCE AND TECHNOLOGY

Yamnampet, Ghatkesar, Hyderabad – 501 301.

CERTIFICATE

This is to certify that the project work titled **"DESIGN OF IoT BASED SMART HEALTHCARE SYSTEM"** submitted by

K. SAI PRASAD	**19315A0431**
M. PRAVAL GOUD	**19315A0432**
B. VILAS YADAV	**19315A0435**

in partial fulfillment for the award of **Bachelor of Technology** Degree in **Electronics and Communication Engineering, Jawaharlal Nehru Technological University, Hyderabad,** is a record of the bonafied work carried out by them during the academic year 2021-2022 under our guidance and supervision.

Mr. S K SATYANARAYANA	**Dr. C.N. SUJATHA**	**Dr. S.P.V SUBBARAO**
Assistant Professor	**Professor**	**Professor**
Internal Guide	**Project Co-Ordinator**	**Head of ECE dept**

Declaration

This is to certify that the work reported in the present project titled **"DESIGN OF IoT BASED SMART HEALTHCARE SYSTEM"** is a record work done by me/us in the **Department of Electronics and Communication Engineering, Sreenidhi Institute of Science and Technology, Yamnampet, Ghatkesar.**

No part of the project is copied from books/ journals/ internet and wherever the portion is taken, the same has been duly referred in the text. The report is based on the project work done entirely by me/ us and not copied from any other source.

K. SAI PRASAD	**19315A0431**
M. PRAVAL GOUD	**19315A0432**
B. VILAS YADAV	**19315A0435**

ACKNOWLEDGEMENT

We wish to express our deep sense of gratitude to our Internal guide **Mr. S K SATYANARAYANA** and project coordinator **Dr. C.N. SUJATHA, Professor (ECE)** who have taken great care in the Project Work undertaken, by devoting her valuable time in advising and guiding us at each phase, leading to successful completion of our Project Work. It has been an educative and enlightening experience working with her.

We are greatly indebted to Prof. **Dr. S.P.V SUBBA RAO,** Head of the department of Electronics and Communication for providing valuable guidance at every stage of this Project Work.

We are thankful to our principal **Dr. T. CH. SIVA REDDY** for providing valuable guidance and support at every stage of this Project Work.

We are profoundly grateful to our director **Dr. C. V. TOMY** and college authorities for helping us in completing our Project Work.

We would also like to thank all our other faculty members and staff who supported us completing the project work.

CONTENTS

Abstract **i**

Contents **ii**

List of figures **a-b**

CHAPTER 1 INTRODUCTION **9-10**

1.1 Problem Statement

1.2 Aim and Objective

1.3 Scope and Limitations

CHAPTER 2 LITERATURE SURVEY **11-12**

2.1 Introduction

2.3 Motivation

2.3 Objectives

2.4 Block Diagram

CHAPTER 3 PROPOSED METHODOLOGY **13-49**

3.1 Module

3.1.1 Overview

3.1.2 History

3.1.3 NodeMCU ESP8266 Features and Technical Specifications

3.1.4 Types

3.1.5 NodeMCU Pinout and Functions Explained

3.1.6 ModeMCU Compatibility with Arduino IDE

3.1.7 The NodeMCU IoT Experiment

3.1.8 NodeMCU Carrier Board

3.1.9 SP8266 Development Platforms

3.2 Sensors

3.2.1 DHT-11

3.2.2 Humidity

3.2.3 Heart Beat Sensor

3.2.4 Buzzer

3.2.5 MEMS Sensor

3.2.6 I2C 1602 Serial LCD Module

CHAPTER 4 SOFTWARE USED **50 - 58**

4.1 ThingsSpeak

4.1.1 Introduction
4.1.2 ThingsSpeak Key Features
4.1.3 Work Flow
4.1.4 Getting Started with ThingsSpeak

Page | 6

CHAPTER 5 TECHNOLOGY **59 - 62**

5.1 IoT

5.1.1 Brief on IoT
5.1.2 Components of IoT
5.1.3 IoT Technologies
5.1.4 Types of IoT

CHAPTER 6 RESULTS AND DISCUSSIONS **63 - 65**

CHAPTER 7 CONCLUSION AND FUTURE SCOPE **66**

CHAPTER 8 REFERENCES **67**

LIST OF FIGURES

Figure 1.1 Block Diagram
Figure 2.1 NodeMCU
Figure 2.2 Amica NodeMCU
Figure 2.3 Lolin NodeMCU
Figure 2.4 Pin Diagram
Figure 2.5 Open source arduino
Figure 2.6 NodeMCU for IOT
Figure 2.7 Carriar Board
Figure 2.8 Carriar Board Schematic
Figure 2.9 NodeMCU jumper position
Figure 2.10 Parts of NodeMCU
Figure 2.11 Multiplexing Pins
Figure 2.12 Basic pins and LED indicator
Figure 2.13 NodeMCU sreial communication
Figure 2.14 DTH-11
Figure 2.15 Thermister of DTH-11
Figure 2.16 Humidity Pulse
Figure 2.17 Start Pulse
Figure 2.18 Response
Figure 2.19 Output Representation
Figure 2.20 End Pulse/frame
Figure 2.21 DTH-11 Pins
Figure 2.22 Pulse Sensor
Figure 2.23 Buzzer
Figure 2.24 piezoelectric
Figure 2.25 Piezoelectric disk beeper
Figure 2.26 Buzzer pin configuration
Figure 2.27 Oximeter
Figure 2.28 Oximeter Pins
Figure 2.29 Oximeter working
Figure 2.30 Graph of position
Figure 2.31 Oximeter sensor with arduino
Figure 2.32 MEMS Sensor
Figure 2.33 MEMS Working
Figure 2.34 Acceleration calculation-capacitor type mems acceleration
Figure 2.35 LCD Module
Figure 3.1 Analyze and visualize your data with matlab
Figure 3.2 Trigger a reaction
Figure 3.3 Send sensor data privately to the cloud
Figure 3.4 Analyze and visualize your data in things speak
Figure 3.5 Sent data to doctor
Figure 3.6 Profile creation
Figure 3.7 Profile creation
Figure 3.8 Channel settings
Figure 3.9 Creating fields
Figure 3.10 Private view
Figure 3.11 Entering key
Figure 3.12 Data Import/Export
Figure 4.1 IOT working module
Figure 5.1 Creating channel and field

Figure 5.2 API key
Figure 5.3 Core Code
Figure 5.4 Humidity and Temperature
Figure 5.5 Graph of MEMS and heartbeat
Figure 5.6 Room temperature

ABSTRACT

Wellbeing is vital for human existence. In this advanced age, event of sicknesses turned out to be more contrasted with before so it became important to deal with wellbeing. In nowadays coronary episode cases are expanding and furthermore different sicknesses, in order to forestall these, ongoing wellbeing checking is required. In current medical care framework, the patient requirements successive visit to clinic to realize their ailment, which is certainly not a simple assignment to accomplish for patient. Our undertaking will give a wellbeing model which help to screen patient by 24*7 which maintains a strategic distance from the need to as often as possible visit clinic. The idea of Internet of Things and Machine Learning are widely utilized in the field of clinical conclusion and medical services to screen the state of a patient. Here we present the Healthcare Monitoring System, an IoT application structure that utilizes various sensors and thingspeak for monitoring the present condition of the patient and provide the treatment according to the patient condition

Key words: Node MCU, Cloud Platform, DHT11 and Heartbeat sensor.

CHAPTER-1

INTRODUCTION

IoT is a hot topic with societal, financial, and technological implications. The Web of Things is a collection of sensors, processors, and microcontrollers mostly with add-ons that are used for web-based communication and have become an internet phenomenon. It's also designed with a standard that facilitates interfacing and communicating with one another as well as with individual clients. This web-based communication aids in the discovery of a variety of applications developed in light of IoT innovation, in which each physical object, such as sensor devices, is linked to the internet. Healthcare plays an important role in the IoT, reducing the amount of problem faced by patients and experts. The needed care is given rather than the costly medical consideration and avoidance is given by the productive medical care administration. This assistance will help each person by following fundamental medical services.

Unrest & rapid online advancement, as well as innovation, such as the Internet of Things, have emerged and are compounding. With distributed computing and edge registration, the Web of Things knows a new and more productive means of exchanging and communicating information. The Internet of Things will revolutionise medical treatment and contribute to humanity's well-being and prosperity. Patients are often expected to attend a facility or an emergency clinic for clinical exams, which is inconvenient & inefficient. The IoT is fit for understanding a continuous wellbeing observing framework that includes sensors to gauge pulse and internal heat level of patients and imagine the information dynamically. By such, individuals can have better control of their medical issue. As opposed to relying upon conflicting visits to focuses or clinical facilities for various tests, individuals can get to their wellbeing information by the web and starts to follow their illnesses. Using open-source administrations like Google Help and IFTTT, the Internet of Things, which recognizes the relationship between devices (Tao et al., 2014), activities like sending an alarm email and messages during a crisis are possible. Additionally, geolocation may be used to track the client's location.

1.1 Problem Statement

The conventional medical services framework, individuals are expected to visit facilities or clinical focuses consistently for clinical tests, which is less successful and tedious. The high clinical expense and long holding up will deter individuals from performing clinical tests 2 consistently. A wellbeing observing framework that gathers and screens the wellbeing status of the client continuously will helps individuals by setting aside their cash and season of visiting facilities and clinical focuses except if there is a requirement for it. Additionally, the security of the wellbeing framework is indispensable to defend the protection of the client. Individuals might stay away from medical care in touchy regions because of wellbeing data security concerns. Brilliant wearable contraptions, for example, Watches of Apple and Samsung are putting away the gathered wellbeing information in the cloud. Distributed

storage permits clients to appreciate excellent administrations with next to no weight of capacity support. Nonetheless, cloud clients are more defenseless against issues like burglary, secrecy, and data spilled to the outsider contrasted with neighborhood capacity clients (Romero, 2012). Putting away private wellbeing data in the wellbeing framework itself will assist with diminishing the opportunity of data spillage, as the wellbeing data may be available by approved clients which works on the security and protection of the framework. The area following capacity assumes an indispensable part in the wellbeing observing framework as it permits individuals to follow the whereabouts of the client. Also, the direction of the client ought to be recorded as it permits individuals to follow the whereabouts of the client. The capacity will prove to be useful when there is a need to find an individual, like the COVID-19 flare-up. The location of the patients is important in tracing the origins of the illness. Furthermore, the information will aid in the development of countermeasures and exit strategies that will stop or slow the spread of the disease.

1.2 Aim and Objective

The goal of the project is to create a wearable health monitoring system that records a user's position, measures their heart rate and body temperature, and displays the data in real time. To maintain track of the patient's health data, authorized users will have access to the gathered data kept in the database. When the user wearing the wellness framework experiences unusual internal heat levels or sudden variations in viewpoint rate, the framework is designed to send quick alerts to phones and texts. The task's objectives are to present an Internet of Things (IoT)-based wellness monitoring framework that continually gathers and displays the client's interior heat level and pulse. Propose a system that uses geolocation to track the user's position while visualizing and storing health data in a database.

1.3 Scope and Limitations

The undertaking centres around the advancement of an IoT based wellbeing checking framework that includes both equipment and programming. The model framework comprises of sensors and an information handling specialist. The model permits approved clients to screen the wellbeing information and the area of the wellbeing framework client through the web. Moreover, the approved client can undoubtedly follow back to the past wellbeing information by getting to the data set that stores all the client's wellbeing information. Since the focal point of this venture is on the execution and association of IoT in the wellbeing observing framework and because of spending plan constraint, the precision of the sensors utilized in this framework won't be thought about as the sensors utilized in this framework are not therapeutically checked and are not appropriate to be utilized for any genuine clinical investigation using any and all means.

CHAPTER-2

LITERATURE SURVEY

2.1 Introduction

Ravi Kishore Kodali proposed medical care observation, which is carried out to examine the patient's temperature. Where the patient's 24-hour care records are being watched, the Zig Bee network convention is used. The cloud is used to keep track of emergency clinic records. IoT-enabled devices improve the quality of care by performing routine checks and gathering data efficiently, while also lowering the cost of care and evaluation of the same. In the second paper, Jasmeet Chhabra et al. suggest the planning and implementation of crisis clinical benefits based on the IoT wellbeing checking framework. The expense of medical services and patient well-being are reduced in this assignment. The web allows for the collection, recording, dissecting, and sharing of information, which eliminates the need for patients to contact a specialist every time they want to check their importent signs like heart rate, temperature, and circulatory strain. Thirumalasetty Sivakanth et al. offer a reconfigurable sensor network for basic wellness checks in Thirumalasetty Sivakanth et al. In a pleased and continuous wellbeing monitoring system, the risk of patient breakdown and harmful consequences is reduced.

2.2 MOTIVATION

In country medical clinics, the offices for wellbeing caring are restricted. The low quality of wellbeing the executives empowers issues in medical care framework Everyone ought to get the information on own wellbeing as simple and right on time as could really be expected. Likewise, it ought to be worth for every. Latest report of The India Spend examination of information says that the 500,000 specialists' deficiency in India. WHO characterizes the specialist patient proportion will be 1:1000 which has been bombed in India? In agricultural nations there is absence of assets and the board to connect the issues of people. An everyday person can't bear the cost of the costly and day to day checkup for his wellbeing. For this reason, different frameworks which give simple and guaranteed caring unit has been created. Proposals framework lessens time with securely took care of hardware.

2.3 Objectives

- To foster constant Health checking framework.
- To examine the patients' real-time data.
- Decreases the intricacy because of the wires.
- Specialists focus on patients and give pressing consideration who are in the most peril in this manner saving lives.
- To break down and foresee soundness of the patients by means of Machine Learning Algorithms.

2.4 Block Diagram

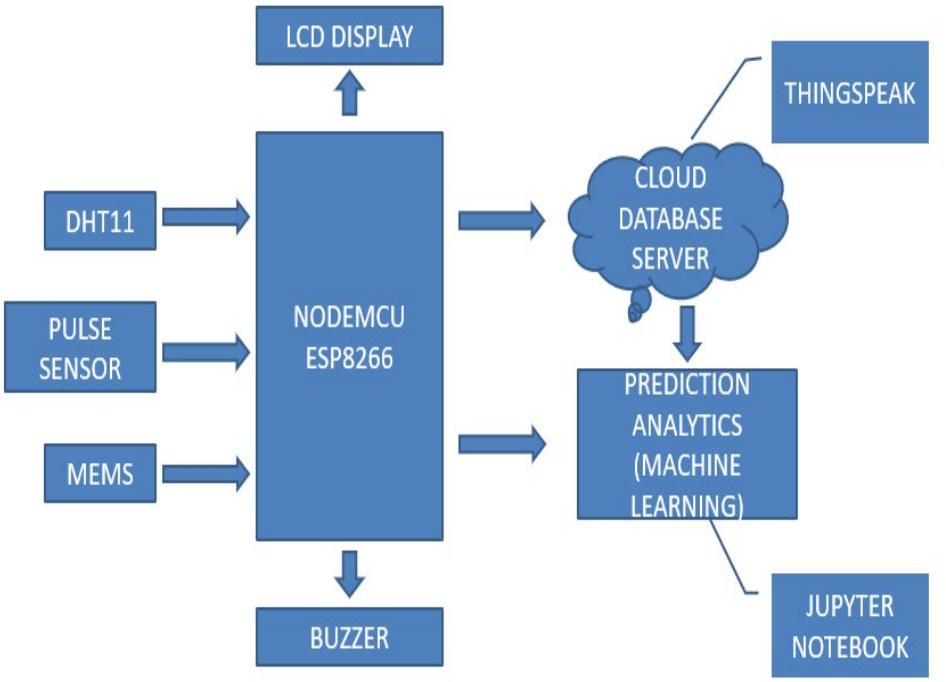


Fig 1 shows the proposed framework. The wellbeing checking sensors are utilized to gather wellbeing related information for example for information securing. Correspondence ought to be conceivable by controller for sending data on web from a distance. Data dealing with has been done at server. All data assembled and amassed at server point. To get wellbeing related data in justifiable configuration it tends to be displayed on page for example information the board.

Fig 2: Working of framework Fig 2 shows the functioning progression of framework. The outcomes gathered from sensor are examined for example on the off chance that strange conduct has been identified, crisis plan enacted to illuminate the Doctor about quiet's wellbeing. So, it diminishes basic circumstances in Hospital.

CHAPTER-3

PROPOSED METHODOLOGY

SENSORS AND MODULE

Hardware requirements:

- NODEMCU ESP8266.
- Dht11.
- Pulse(or)Heart beat sensor.
- Buzzer.
- LCD Display.
- MEMS.

3.1 MODULE

NODEMCU ESP8266.

NodeMCU is a low-cost open source IoT platform. It began with firmware and equipment based on Espressif Systems' ESP8266 Wi-Fi SoC and the ESP-12 module. Later, support for the ESP32 32-bit MCU was introduced. The NodeMCU (Node MicroController Unit) is an open-source programming and equipment update platform based on the ESP8266, a tiny System-on-a-Chip (SoC). The ESP8266 was conceived and produced by Espressif Systems, and it has all of the key components of a PC, including a CPU, RAM, networking (WiFi), and, surprisingly, a cutting-edge working framework and SDK. If all other factors are equal, this is a fantastic solution for Internet of Things (IoT) businesses.

On the other side, the ESP8266 is difficult to access and utilise as a chip. For the most basic tasks , like as making it on or sending a keystroke to the chip "PC," you should connect wires to its pins by the appropriate simple voltage. You'll also need to programme it in low-level machine directions that the chip equipment can know . When the ESP825 is used like a built-in regulator chip in well-made devices, this level of mix isn't an issue. It is a significant burden for experts, programmers, or understudies who need to investigate various options in their own IoT initiatives.

What can be said, on the other hand, regarding Arduino? The Arduino project built an open-source equipment design and programming SDK for its flexible IoT regulator. The NodeMCU is a microcontroller board featuring a USB connector, LED lights, and standard data connections, similar to the Arduino equipment. It also shows where sensors or other sheets should be connected in the traditional way. Unlike NodeMCU, however, the Arduino board may have a variety of CPU chips (often an ARM or Intel x86 chip), memory chips, and

programming conditions. There is also an Arduino reference design for the ESP8266 chip. Arduino's adaptability, on the other hand, implies that it may be utilised by a wide range of enterprises. Most Arduino sheets, for example, don't have WiFi capabilities, and others don't even have USB ports., lack WiFi capabilities, and some even lack a USB connector in favour of a sequential information port.

3.1.1 Overview

Prototype that is open source board layouts are there for NodeMCU, an firmware that open-source. The term "NodeMCU" combines the terms "hub" and "MCU" (miniature regulator unit). The term "NodeMCU" is used to refer to the firmware rather than the associated advancement units.

Both the firmware and the prototype board designs are open source.

The firmware makes use of the Lua prearranging language. The firmware is based on the eLua project, which is compatible with the Espressif Non-OS SDK for ESP8266. It incorporates lua-cjson and SPIFFS, among other open source projects. Clients must pick the components required for their project and create a firmware customised to their individual demands due to budget constraints. Furthermore, support for the 32-digit ESP32 has been included.

Prototype equipment often consists of a circuit board that functions as a double in-line pack (DIP), which combines a USB controller with a more unassuming surface-mounted board carrying the MCU and radio wire. The DIP configuration was chosen to enable for easy prototyping on a breadboard. Initially, the ESP8266's ESP-12 module was explored., which is a Wi-Fi SoC containing a Tensilica Xtensa LX106 focus mostly usefull in IoT applications.

3.1.2 History

Not long after the ESP8266 became popular, the NodeMCU was born. Espressif Systems released the ESP8266 on December 30, 2013. When Hong posted the nodemcu-firmware master record to GitHub on October 13, 2014, NodeMCU was born. Developer Huang R contributed the gerber record of an ESP8266 board named devkit v0.9 after two months, extending the project to include an open-gear stage. Tuan PM focused on the NodeMCU project and swiftly ported the MQTT client library from Contiki to the ESP8266 SoC stage. The MQTT IoT display could then be assisted by NodeMCU, which used Lua to connect to the MQTT agent. Devsaurus modified the u8glib to the NodeMCU project on January 30, 2015, allowing NodeMCU to run LCD, Screen, OLED, and even VGA.

The primary makers abandoned the firmware effort in the middle of 2015, and a coalition of free allies see over. In the early winter of 2016, the NodeMCU has amassed a collection of forty distinct modules.

3.1.3 NodeMCU ESP8266 Features and Technical Specifications.

Features

- At last, programable Wi-Fi module.
- Programming characterised equipment similar to Arduino IO.

- The Arduino IDE or the simple and powerful Lua programming language may be used to make changes.
- Plug and play with USB-TTL included.
- 10 GPIOs D0-D10, PWM functionality, IIC and SPI compatibility.
- Wi-Fi organisation (may be used as a tunnel as well as a station, has a web server), connects to the internet to get or send data.
- Occasion driven API to network applications.
- PCB radio wire.

Note: Complete specialized data can be found in the NodeMCU ESP8266 Datasheet, connected at the lower part of this page.

Specifications

- Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106
- Operating Voltage: 3.3V
- Input Voltage: 7-12V
- Digital I/O Pins (DIO): 16
- Analog Input Pins (ADC): 1
- UARTs: 1
- SPIs: 1
- I2Cs: 1
- Flash Memory: 4 MB
- SRAM: 64 KB
- Clock Speed: 80 MHz
- USB-TTL based on CP2102 is included onboard, Enabling Plug n Play
- PCB Antenna
- Small Sized module to fit smartly inside your IoT projects

	Official NodeMCU	NodeMCU Carrier Board	LoLin NodeMCU
Clock Speed	80 MHz	80 MHz	80 MHz
USB to Serial	CP2102	**CP2102**	CH340G
USB Connector	Micro USB	Micro USB	Micro USB
Operating Voltage	3.3V	3.3V	3.3V
Input Voltage	4.5V-10V	4.5V-10V	4.5V-10V
Flash Memory/SRAM	4 MB / 64 KB	4 MB / 64 KB	4 MB / 64 KB
Digital I/O Pins	11	11	11
Analog In Pins	1	1	1
ADC Range	0-3.3V	0-3.3V	0-3.3V

3.1.4 Types

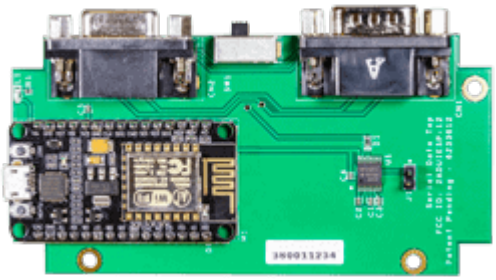
There are two accessible variants of NodeMCU as rendition 0.9 and 1.0 in the adaptation 0.9 their ESP-12 and form 1.0 their ESP-12e where e means "Upgraded".

Official Amica NodeMCU

The Amica NodeMCU is 49mm x 26mm in size, with a pin spacing of 0.1″ and a column spacing of 0.9″.

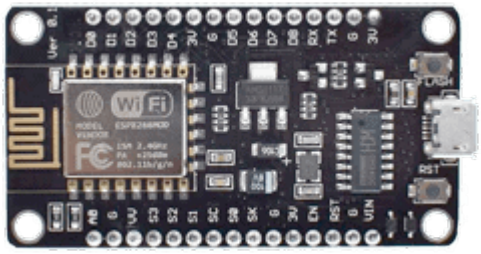
The Amica NodeMCU is about 25% lesser than a LoLin-style NodeMCU, but it is still functional.

Official Amica NodeMCU
on Carrier Board



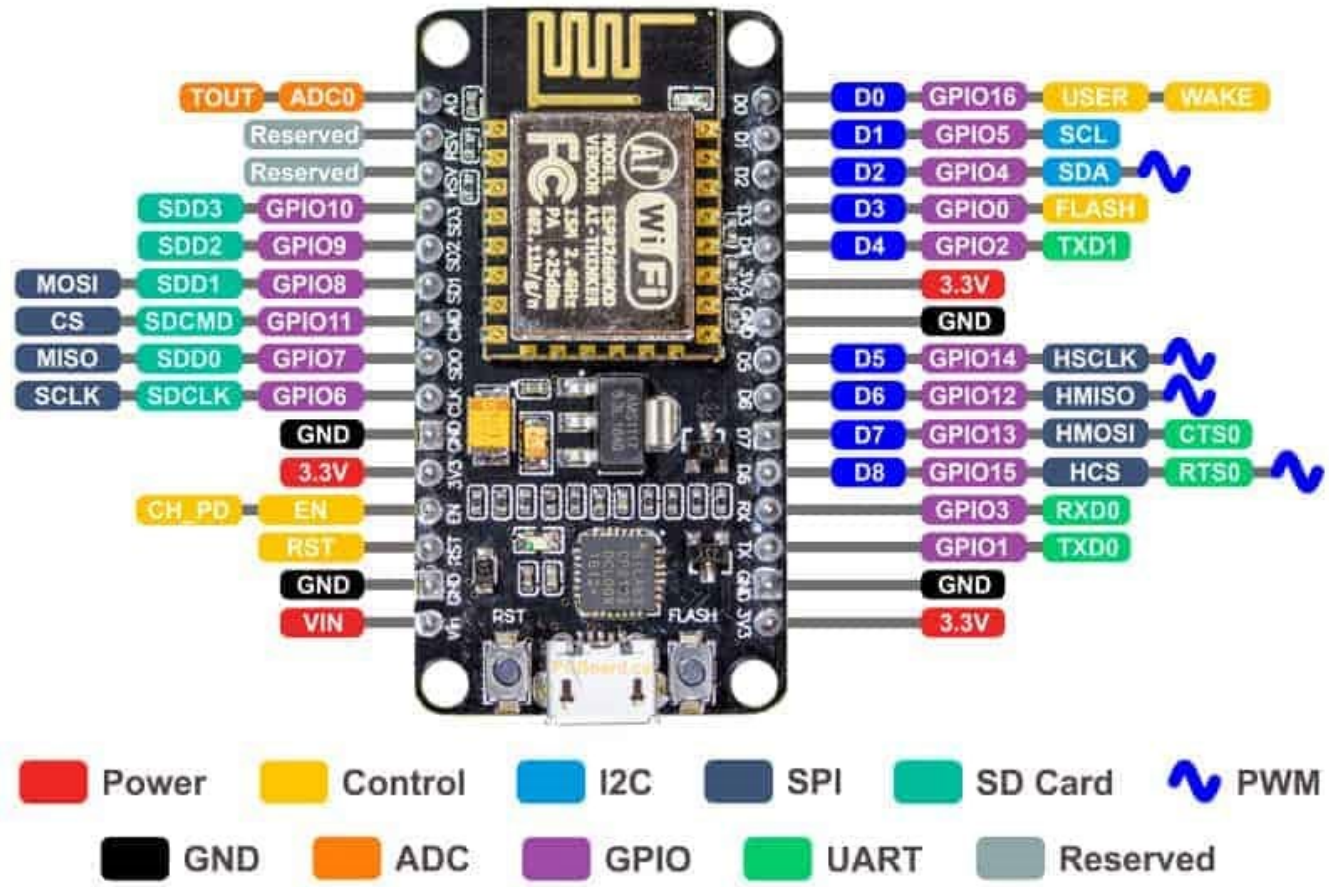
Using two DB-09 female/male connections, the Amico NodeMCU is installed on a 102mm x 51mm transporter board.

Lolin
NodeMCU



LoL in style Node MCU contains 58mm x 32mm with a pin separating of 0.1″ among pin and 1.1″ b/w lines.

1.5 Node MCU Pins and it's Functions .



Contains 4 power pins on the board and three 3.3v pins and a Vin pin

VIN may usefull for feed the Node MCU/ESP 8266 and its outskits directly. The power sent to VIN is controlled by the NodeMCU module's locally accessible controller; you may also send 5V managed to Vin pin 3. The installed voltage controller produces 3V pins, which may be used to deliver capacity to external devices.

The GND pins for the Node MCU/ESP 8266 are **GND**.

I2C pins are used to link together I2C sensors and peripherals. Both I2C Master and Slave are supported. The I2C interface is well-known, and the clock repetition is limited to 100 kHz at maximum. The I2C clock repetition should be greater than the slave contraption's slowest clock repetitions, as should be evident.

GPIO (General Purpose Input/Output) Pins The NodeMCU/ESP8266 has 17 GPIO pins that may be allocated to I2C, I2S, UART, PWM, IR Remote Control, LED Light, and Button functionalities automatically. Each advanced-enabled GPIO may be set to pull-up or pull-down on the inside or outside, or to high impedance. It can also be set to edge-trigger or level-trigger to create CPU incursions when setup as an information**.**

ADC (Audio/Digital Converter) Channel A 10-bit SAR ADC is contains with in the Node MCU. ADC may be used to do both of these functions. VDD3P3 pin's power supply voltage is tested, as is TOUT pin's input voltage. Regardless, they will not be completed at the equal time.

UART connector pins The NodeMCU/ESP8266 has two UART ports (UART0 and UART1) that can transmit at up to 4.5 Mbps and allow nonconcurrent (RS232 and RS485) connection. Communication may be done via UART0 (pins TXD0, RXD0, RST0, and CTS0). UART1 (TXD1 pin) on the other hand, includes only information delivered over these lines and is typically used for printing logs.

Pins for SPI The NodeMCU/ESP8266 emphasises two SPIs in slave and expert modes (SPI and HSPI). These SPIs also strongly support the following applicable SPI features:

•Up to 80 MHz and isolated tickers of 80 MHz
•Up to 64-Byte First In Firt Out
•4 timing techniques of the SPI design move

Pins for **SDIO** Secure Digital Input/Output Interface (SDIO) is a feature of the NodeMCU/ESP8266 that is used to connect SD cards easily. The 4-digit 25 MHz SDIO v1.1 and 4-bit 50 MHz SDIO v2.0 standards have been maintained.

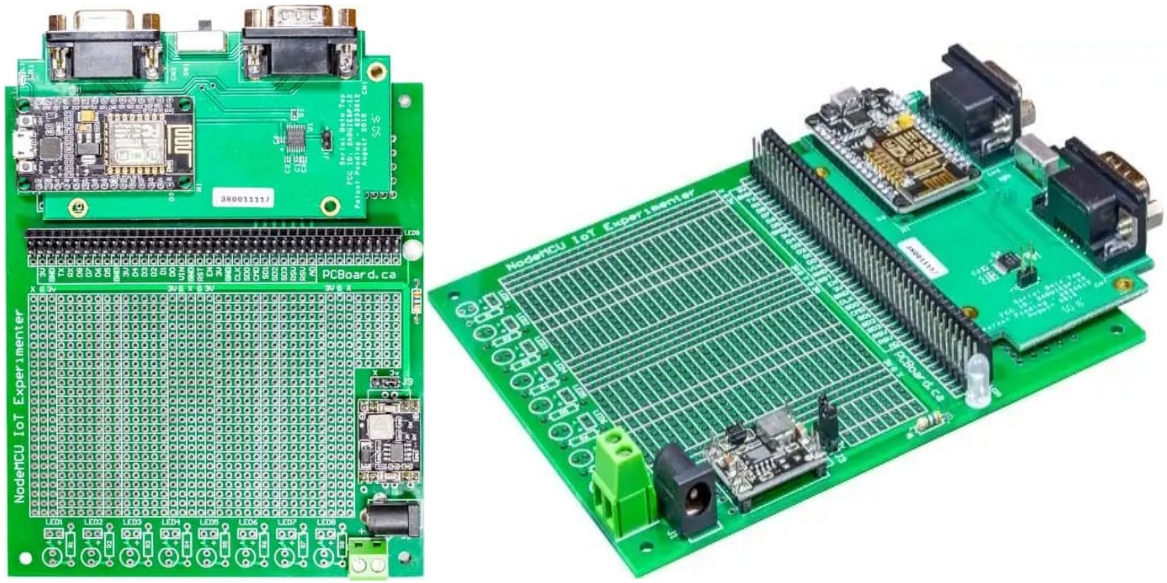
PWM (Pulse Width Modulation) pins are accessible on four of the board's channels (PWM). Automatically produced PWM output can be utilised to power complicated engines and LEDs. The PWM recurrence range is adjustable from 1000 to 10000 seconds (100 Hz and 1 kHz).
Control pins are used to control the Node MCU/ESP 8266. Chip These pins house the enable (EN), reset (RST), and wake (WAKE) pins.

The ESP8266 chip is activated when the EN pin is pushed HIGH. When the chip is pushed LOW, it uses the least amount of power possible.
RST: Use the RST pin to reset the ESP8266 chip.
The rouse pin is used to jolt the chip up from its slumber.
Control pins are used to control the Node MCU/ESP 8266 chip is enabled. When the pin is pushed LOW, the chip operates at the lowest possible power.
RST: To reset the ESP 8266 chip, use the RST pin.
WAKE: The rouse pin is used to Wake the chip from i sleep.

3.1.6 ModeMCU With The Arduino IDE

The Node MCU has a number of development options, including Arduino IDE compatibility (Integrated Development Environment). The Node MCU/ESP 8266 people group took the IDE option a step further by building an Arduino add-on. If you're just starting started programming the ESP8266 or even a laid-out designer, this is the energetically recommended environment. To learn how to use the Arduino IDE with a Node MCU/ ESP 8266, go to our dedicated website..

3.1.7 The NodeMCU Experiment



The NodeMCU IoT Experimenter is a flexible prototype platform that works with a wide range of NodeMCU modules, including the NodeMCU Carrier Board. For IoT applications, sophisticated or clear interaction, and development phases, this is excellent. When paired with this prototype board, the NodeMCU's adaptability, which includes the ability to be adjusted and utilised from the Arduino IDE, making it the ultimate experimenter's answer.
The NodeMCU IoT Experimenter is 5 5/16″ x 4.5″ and has a bind veil on each side, plated apertures, a high-contrast silk-screen name part, and prototyping areas (135mm x 115mm).
The board contains a mounting attachment zone that can accommodate NodeMCU modules with a 1.1″ or 0.9″ pitch.That includes possible versions such as the LoLin Node MCU models, as well as the Amica Node MCU transporter board (slim pin dispersion). The underlying USB port of your Node MCU module may be used to provide power directly to it. Power, on the other hand, may be supplied to the IoT Experimenter board, which is equipped with a controlled power supply module.

On the model surface, there are over 1,000 plated-through apertures for installing eight status pointer LEDs, as well as dropping resistors and a power marker LED. Power bus for the Ground (G), +3.3v (3v) power rail, and a 3rd rail X are available in the prototyping region. A mounting attachment location on the board accepts NodeMCU modules with either a broad 1.1″ pitch or 0.9″ pitch. This contains the Amica NodeMCU transporter board (slim pin dispersion) as well as other suitable options such as the LoLin. Outer voltages, such as 5V, can be used on the third rail.

Interacting with the Node MCU is accomplished using a continues of headers that stretch every pin of the Node MCU into 4-column columns. The appropriate pins from the Node MCU are labelled on each port. The header area is located beneath the Node MCU and is made up of standard 40-pin headers to allow for easy attachments and header pin connections.

Related projects

ESP8266 Arduino Core

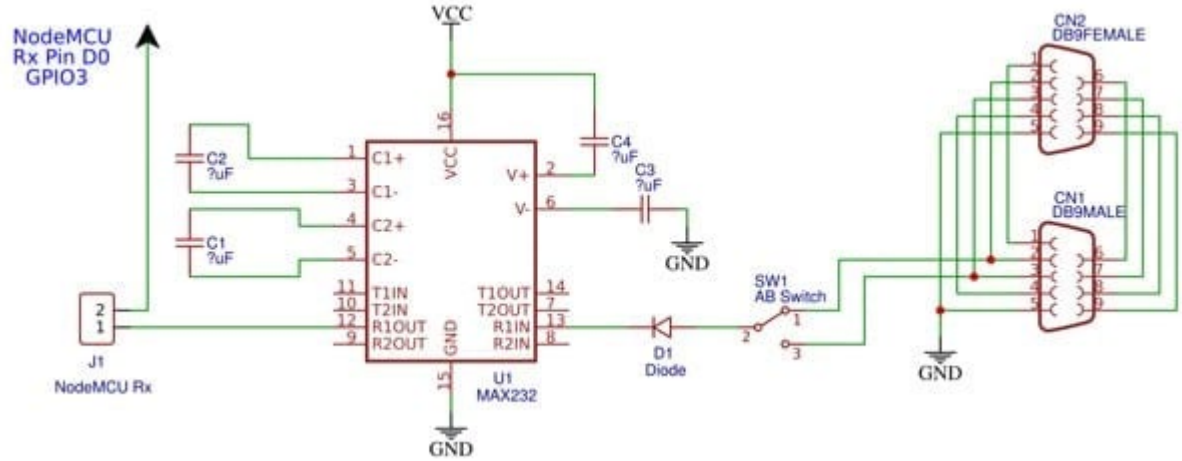
As Arduino.cc began developing new MCU sheets in light of non-AVR processors such as the ARM/SAM MCU, which is used in the Arduino Due, they expected to make changes to the Arduino IDE so that it would be relatively simple to replace toolchains and enable Arduino C/C++ to be prepared for these new processors. They accomplished this with the Board Manager's and SAM Core's presentations. "Deeply" refers to the set of programming components that the Board Manager and the Arduino IDE anticipate to include in Arduino C/C++ source file for the target MCU's machine language. Some ESP8266 fans created an Arduino centre dedicated to the ESP8266 WiFi SoC, dubbed the "ESP8266 Core for the Arduino IDE". This has become the primary programming development step for ESP8266-based modules and advancement sheets, including NodeMCUs.

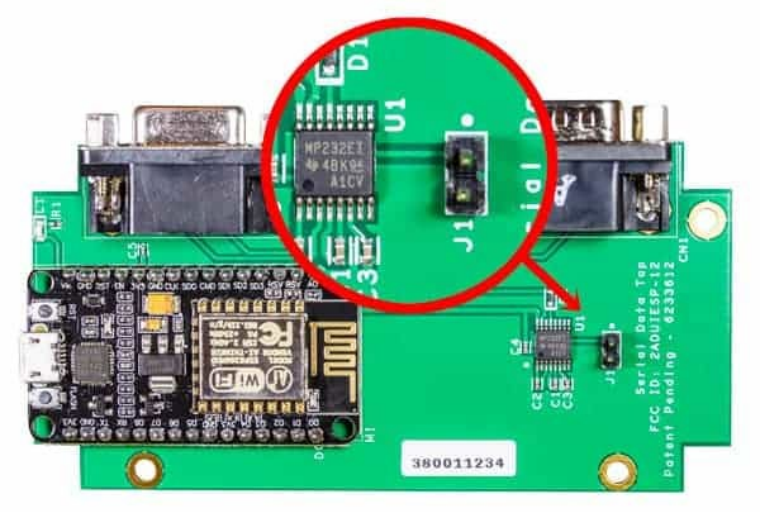
3.1.8 NodeMCU Carrier Board

The Amica NodeMCU ESP8266 processor is included in the NodeMCU Carrier Board, as well as a DB09 male and female connection and an RS-232 level converter.

The Carrier Board's early design was for a WiFi application, and the sequential ports allowed RS-232 data to be sent to the NodeMCU via the chronic connections using a MAX232 viable level converter. The level converter allows authentic RS-232 signals to be sent from the NodeMCU without having to worry about large voltage fluctuations.

NodeMCU Carrier Board Schematic



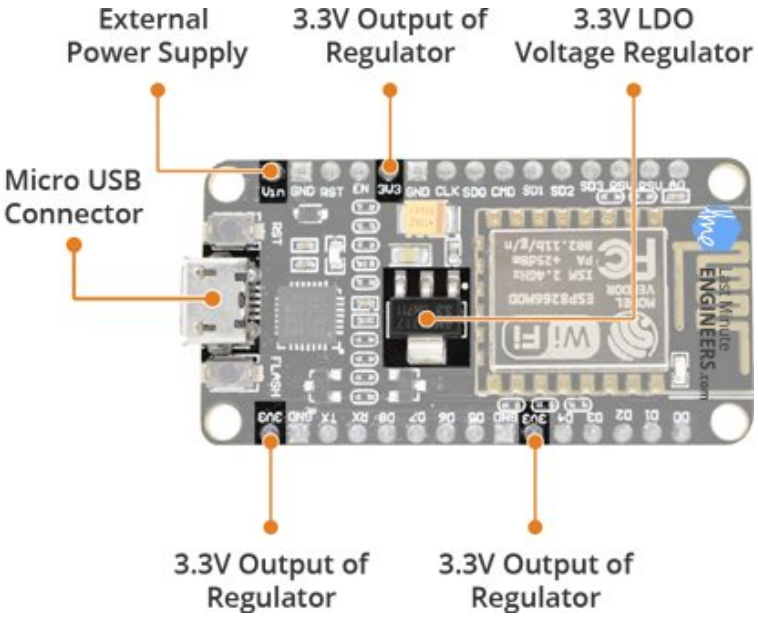


The two DB-09 connections, as well as the switch at SW1, are shown on the NodeMCU Carrier Board schematic. The information from either DB-09 is flipped between Pin 2 and Pin 3 to the level converter when the switch is made.

Finally, at J1, there is a jumper position. This allows sequential data from either DB-09 to appear on the Rx pin of the NodeMCU.

Power Requirement

Because the ESP8266's operating voltage range is 3V to 3.6V, the board includes an LDO voltage controller to keep the voltage stable at 3.3V. It can reliably provide up to 600mA, which should be plenty when the ESP8266 draws up to 80mA during RF communications. The controller's output is also separated off to one of the board's sides and labelled 3V3. This pin can be used to provide capacity to external elements.

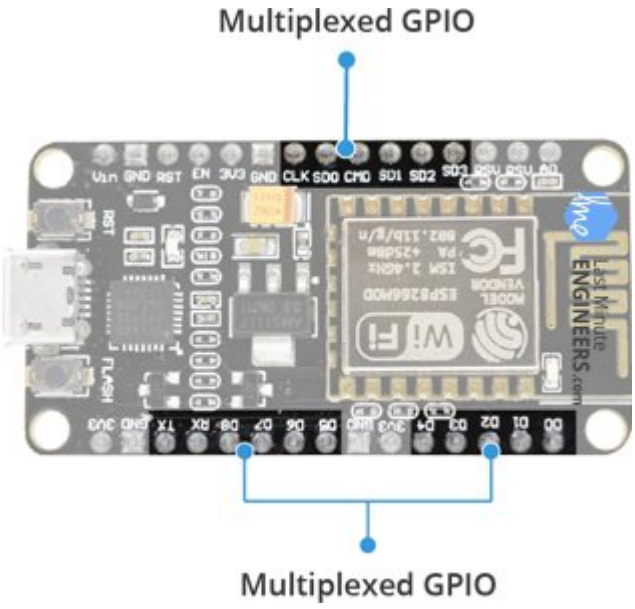


The on-board MicroB USB port provides capacity to the ESP8266 NodeMCU. If you have a controlled 5V power source, the VIN pin may be used to directly feed the ESP8266 and its peripherals.

I/O and peripherals

The ESP8266 NodeMCU includes 17 GPIO pins on both sides of the development board, which are split off to pin headers. These pins may be used for a variety of tasks around the house, including:

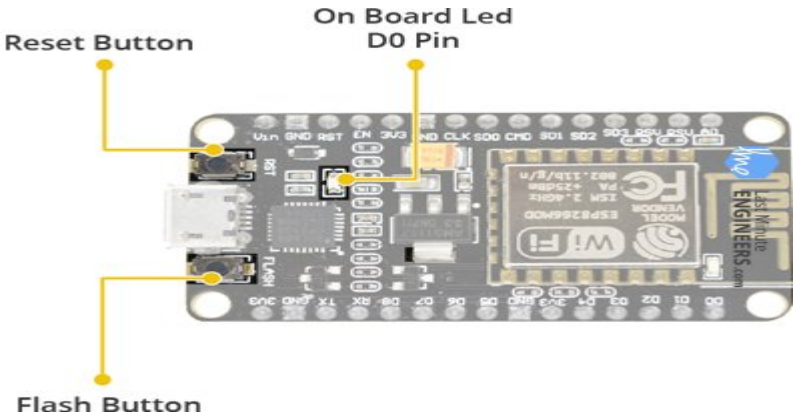
- **A 10-bit ADC channel is referred to as an ADC channel.**
- **UART interface — The UART interface is used to serially load code.**
- **PWM outputs — PWM pins for controlling motors or dimming LEDs.**
- **SPI, I2C, and I2S interfaces — SPI and I2C interfaces are used to connect a variety of sensors and peripherals.**
- **If you wish to add sound to your project, use the I2S interface.**



The ESP8266's pin multiplexing functionality comes in handy (Multiple peripherals multiplexed on a single GPIO pin). As a result, a single GPIO pin may perform the functions of PWM, UART, and SPI.

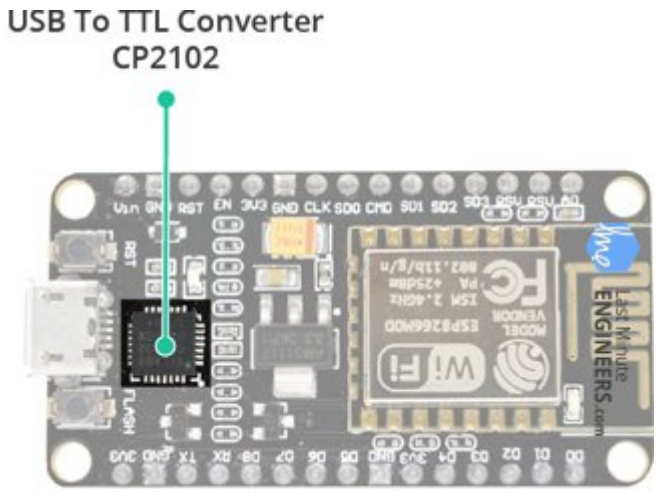
LED Indicator & On-Board Switches

Two buttons are highlighted on the ESP8266 NodeMCU. The Reset button, labelled RST and located in the upper left corner, is evidently used to reset the ESP8266 chip. The download button on the bottom left corner of the FLASH button is used while updating firmware.



The board likewise has a LED pointer which is client programmable and is associated with the D0 pin of the board.

Serial Communication



The board includes a Silicon Labs CP2102 USB-to-UART Bridge Controller, which converts the USB signal to sequential and allows your PC to program and communicate with the ESP8266 chip.
Assuming you have a more established adaptation of CP2102 driver introduced on your PC, we suggest overhauling now.

3.1.9 SP8266 Development Platforms

Presently, we should continue on toward the fascinating stuff.

The ESP8266 may be programmed using a number of different advancement schemes. You can utilize Espruino - JavaScript SDK and firmware swiftly replicating Node.js, Mongoose OS - A working structure for IoT contraptions (suggested stage by Espressif Systems and Google Cloud IoT), or a Espressif item headway unit (SDK), or one of the stages documented on WiKiPedia.

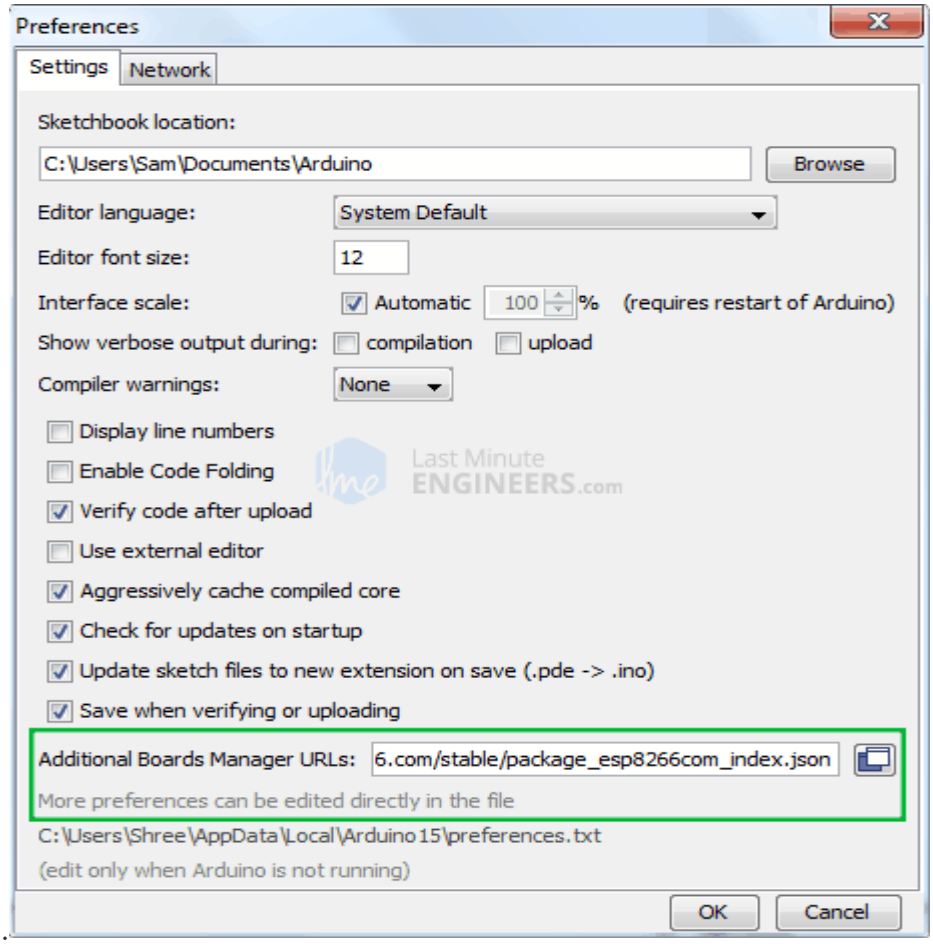
Fortunately, the incredible ESP8266 people group took the IDE decision a step further by creating an Arduino add-on. This is the climate we recommend beginning with if you're new to programming the ESP8266, and it's the one we'll record in this tutorial.

This Arduino ESP8266 add-on is based on Ivan Grokhotkov's and the rest of the ESP8266 people group's incredible work. For further information, visit the ESP8266 Arduino GitHub repository.

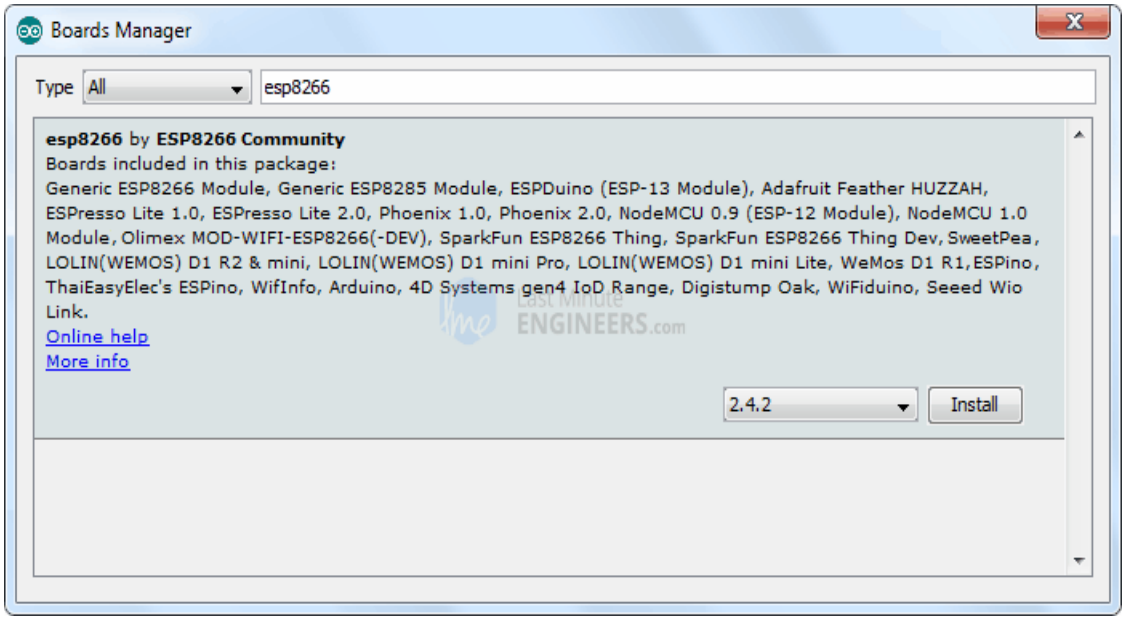
Installing the ESP8266 Core on the Windows Operating System

Let's continue with the introduction of the ESP8266 Arduino centre. The most important thing is to have the latest Arduino IDE (Arduino 1.6.4 or above) installed on your computer. If you don't have it, we recommend that you get it right now.

To begin, we'll need to provide a special URL to the board chief. Go to File > Preferences in the Arduino IDE. Then, in the Additional Board Manager URLs text box in the lower portion of the window, copy and paste the following URL:



Select OK. Then click to Tools > Boards > Boards Manager to find the Board Manager. Aside from the regular Arduino sheets, there should be a few of new sections. Compose esp8266 to channel your quest. Select Install after clicking on that paragraph.



The ESP8266 board definitions and instruments have a distinct set of gcc, g++, and other quite large, collected parallels, so downloading and installing them may take a few seconds (the documented record is 110MB). When the installation is complete, a little INSTALLED text will appear beside the paragraph. The Board Manager is now complete.

How can I write NodeMCU code?

We should see the IDE (Integrated Development Environment) intended for NodeMCU enhancement after setting up ESP8266 with Node-MCU firmware.

NodeMCU with ESPlorer IDE

Lua scripts are by and large used to code the NodeMCU. Lua is an open-source, lightweight, embeddable prearranging language based on top of C programming language.

For more data about how to compose Lua script for NodeMCU allude to Getting begun with NodeMCU utilizing ESPlorerIDE

Arduino IDE and NodeMCU

Here's another way to make NodeMCU using a well-known IDE, such as Arduino IDE. We may also use the Arduino development environment to create NodeMCU apps. This saves Arduino designers the trouble of learning a new dialect and IDE for NodeMCU.

Refer to Getting Started with NodeMCU Using ArduinoIDE for further information on how to construct an Arduino sketch for NodeMCU.

What's the difference between ESPlorer and the Arduino IDE?

While developing an application for NodeMCU using ESPlorer IDE and Arduino IDE, there is a programming language distinction that may be made.

If we're using Arduino IDE to create NodeMCU programmes, we'd want to code in C++, and if we're using ESPlorer IDE, we'd like to code in Lua.

Because NodeMCU is a Lua interpreter at its core, it can easily understand Lua script. When we create Lua scripts for NodeMCU and send/transfer them to NodeMCU, they are executed in a sequential order. NodeMCU will not be able to build a duplicate firmware record of code. It will, for all intents and purposes, transmit a Lua script to NodeMCU for execution.

When we compose and organise code in the Arduino IDE, the ESP8266 toolchain creates a paired firmware record of the code we constructed behind the scenes. Furthermore, when we transfer it to NodeMCU, the newly developed twofold firmware code will be streaked throughout every NodeMCU firmware. To be honest, it makes up the whole firmware.

That is the reason why NodeMCU will not accept any further Lua scripts/code once it has been processed by Arduino IDE. After being streaked by Arduino sketch/code, there will be no more Lua mediator, and attempting to transfer Lua scripts would result in an error. We'll start with the Lua script again, this time with NodeMCU firmware.

Since Arduino IDE assembles and transfer/composes total firmware, it requires more investment than ESPlorer IDE.

Using the Arduino IDE to programme the Nodemcu

Because the Arduino IDE is simple to use, it may easily be used to modify the Nodemcu Development Board. It will take no more than 10-20 minutes to programme Nodemcu using the Arduino IDE. All you'll need is the latest version of the Arduino IDE, a USB cable, and the Nodemcu board. Set up your Arduino IDE for NodeMCU using our Getting Started Tutorial for NodeMCU. To set up your Arduino IDE for the ESP32 module, see my getting everything rolling tutorial exercise for ESP32 Wifi + Bluetooth Module..

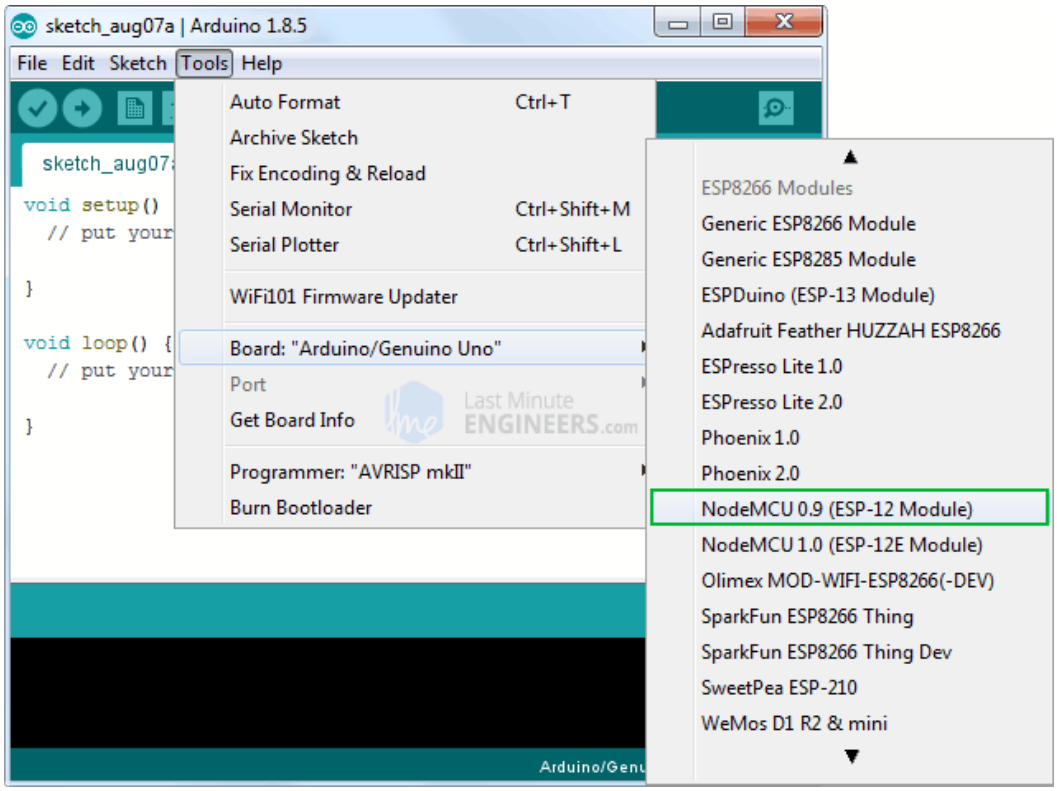
Blink is an Arduino example.

We'll upload the most basic sketch of all - The Blink! - to guarantee the ESP8266 Arduino centre and the NodeMCU are properly set up.

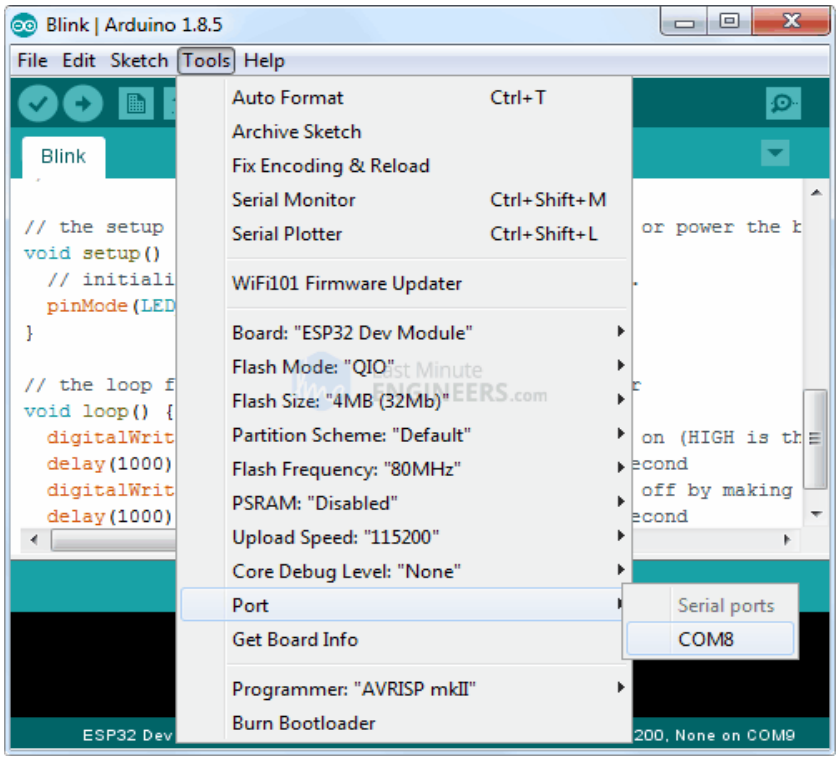
For this test, we'll use the on-board LED. The D0 pin of the board is coupled with the on-board Blue LED and is client programmable, as mentioned before in this educational exercise. Great!

Before we start transferring the programme and playing with the LEDs, we need to make sure that the board is selected correctly in the Arduino IDE. Open the Arduino IDE and go to the

Tools > Board menu and choose NodeMCU 0.9 (ESP-12 Module).



Connect your ESP8266 NodeMCU to your PC via the tiny B USB connector. When the board is attached, it should be assigned a unique COM port. This will be something like COM# on Windows computers, and /dev/tty.usbserial- Under the Arduino IDE > Tools > Port menu, choose this sequential port. Select the 115200 Upload Speed option as well.



Caution:

More thought should be paid to board selection, COM port selection, and upload speed selection. When uploading fresh files, you can encounter the espcomm upload mem error if you forget to do so.

After we've finished, check out the sample drawing below.

```
void setup ()

{

        pinMode (D0, OUTPUT);

} void loop()

{

        digitalWrite (D0, HIGH);

        delay (500);

        digitalWrite (D0, LOW);

        delay (500);

}
```

The LED will begin flashing after the code has been uploaded. To get your ESP8266 to start running the code, you may need to press the RST button.

• IoT device prototyping

• Low-power battery-powered applications

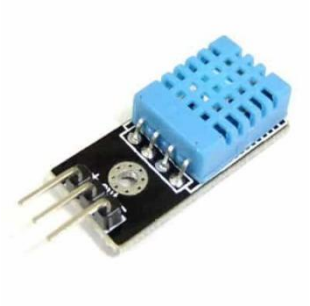
• Network projects

• Projects needing numerous I/O interfaces with Wi-Fi and Bluetooth capabilities

3.2 Sensors

3.2.1 DHT-11

The DHT11 moisture and temperature sensor may be used as a sensor or a module. The draw up resistor and a power-on LED distinguish this sensor and module. The DHT11 is a general-purpose moisture sensor. This sensor uses a thermistor and a capacitive dampness sensor to measure the surrounding air. This sensor is used to monitor the moisture variation of the climate in which the crops are grown. This is a digital sensor that evaluates rate design stickiness esteem.

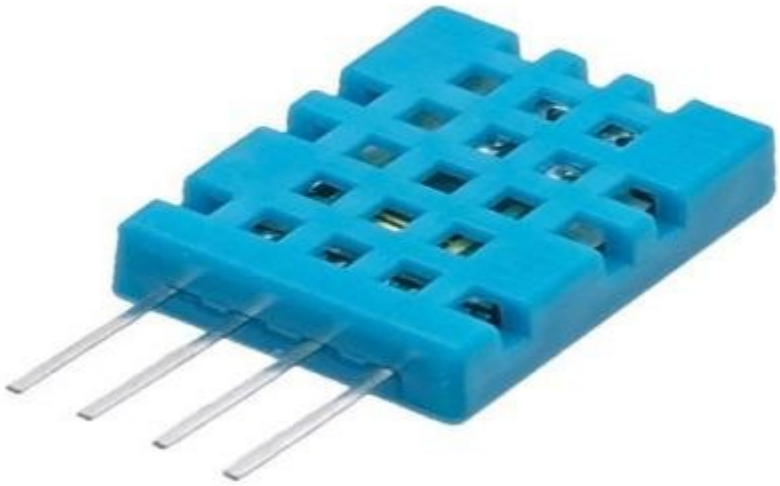


The DHT11 is a low-cost electronic sensor that can sense temperature and stickiness. This sensor may interface with any small regulator, such as an Arduino, Raspberry Pi, or other microcontroller, to measure stickiness and temperature in real time.

A capacitive moisture detecting component and a temperature detecting thermistor make up a working DHT11 sensor. A dampness retaining substrate acts as a dielectric between the two terminals of the dampness detecting capacitor. With the changing of dampness levels, the capacitance value changes. The IC measures, processes, and converts the altered obstruction values into advanced structures.

This sensor employs a Negative Temperature Coefficient Thermistor to estimate temperature, which causes a decrease in its opposition esteem as temperature rises. This sensor is usually made of semiconductor ceramics or polymers to achieve a higher opposition esteem in any case, even for the smallest variation in temperature.
DHT11 has a temperature range of 0 to 50 degrees Celsius with a 2-degree precision. The stickyness range of this sensor is 20 to 80 percent, with a precision of 5%. This sensor's testing rate is 1Hz, which means it does one reading per second. DHT11 is a tiny transistor with a 3-to-5-volt operating voltage. The maximum current used for estimation is 2.5mA.

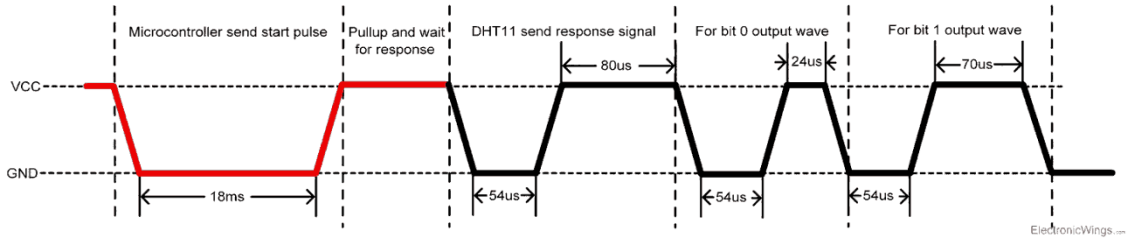
The DHT11 sensor contains four pins: VCC, GND, Data Pin, and a pin that isn't connected to anything. A 5k to 10k ohm pull-up resistor is used to connect the sensor and the tiny regulator.



3.2.2 HUMIDITY (DHT11)

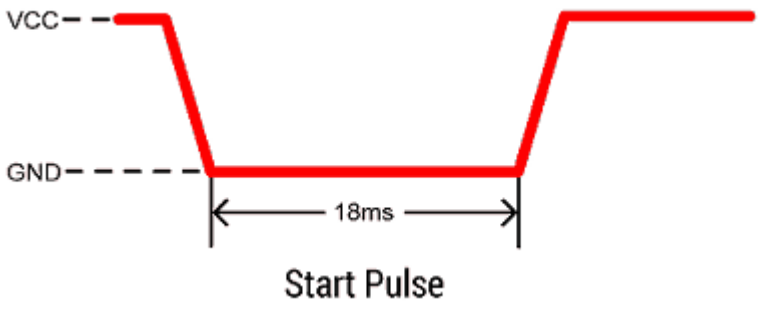
DHT11 uses only a single wire for communication. The reasoning one or rationale zero on this pin is defined by voltage levels with certain time esteem.

The correspondence cycle is separated into three stages: first, a solicitation is sent to the DHT11 sensor, then the sensor sends a response heartbeat, and finally, the information of absolute 40 pieces is sent to the microcontroller.



Communication process

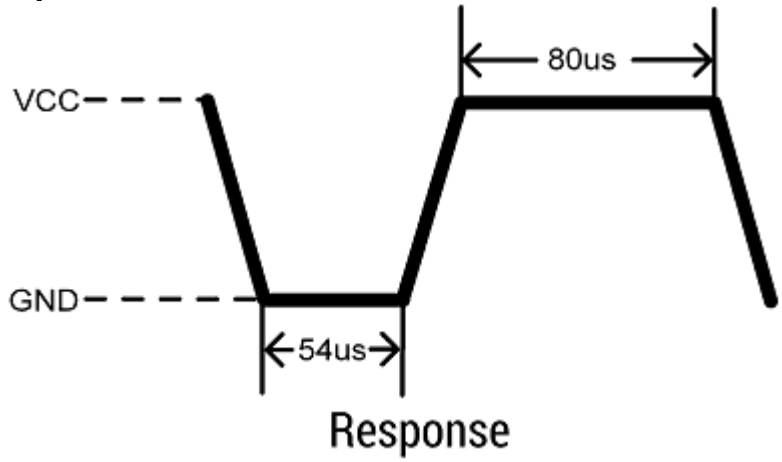
Start pulse (Request)



- •In order to communicate with DHT11, we must first transmit the initial heartbeat to the sensor.
- •To deliver a start beat, pull the information pin down (low) for at least 18ms and then lift up, as shown in the diagram.

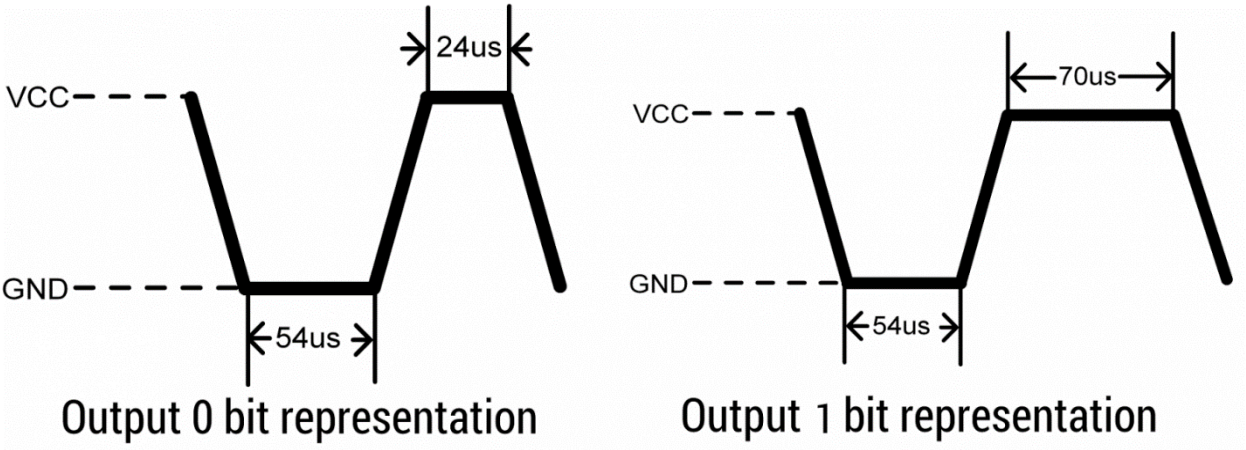
- Reaction

Response



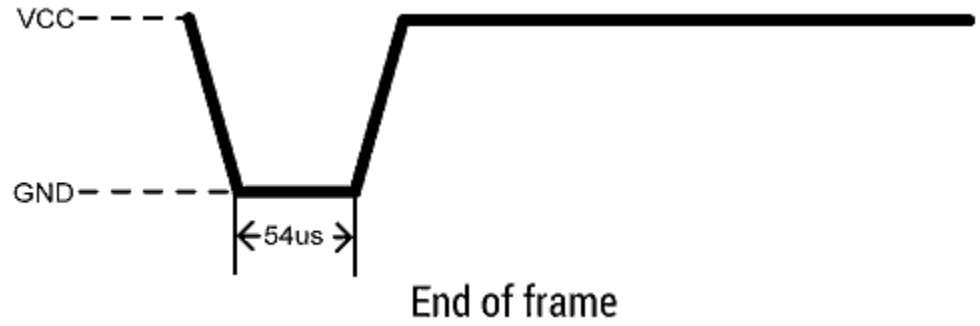
- In the wake of getting start beat from, DHT11 sensor sends the reaction beat which demonstrates that DHT11 got start beat.
- The reaction beat is low for 54us and afterward goes high for 80us.

Data



- After transmitting the reaction beat, the DHT11 sensor provides the data, which includes wetness and temperature values as well as a checksum.
- The information outline is made up of 40 parts, each of which is 8 cycles long and comprises 5 fragments (bytes).
- The first two fragments in these five parts contain mugginess esteem in decimal whole number structure. Relative Percentage Humidity is calculated using this value. The first eight pieces are whole number parts, whereas the latter eight pieces are partial parts.
- Next two fragments contain temperature esteem in decimal whole number structure. This worth gives us temperature in Celsius structure.
- Last section is the checksum which holds checksum of initial four portions.
- Here checksum byte is immediate expansion of stickiness and temperature esteem. Also, we can confirm it, regardless of whether it is same as checksum esteem. On the off chance that it isn't equivalent, then there is some blunder in the got information.
- When information got, DHT11 pin goes in low power utilization mode till next start beat.

End of frame

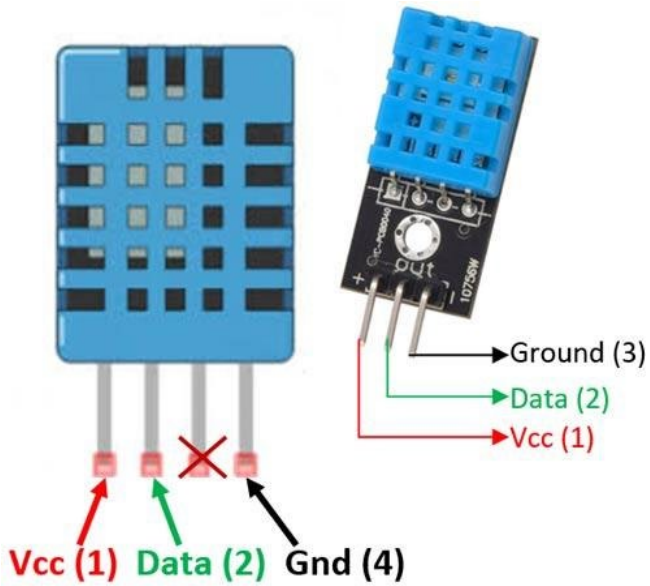


- After delivering 40 pieces of data, the DHT11 sensor delivers a 54us low level and then jumps high. DHT11 then enters a state of repose.

DHT11 versus DHT22 (DHT11 vs DHT22)

There are two types of DHT sensors; they seem similar and use the same pinout, but they have different quality and features:

•3.5 to 5V power and I/O
•2.5mA max current use during change (while mentioning information)
•Great for 20-80 percent stickiness readings with 5% precision
•Great for 0-50°C temperature readings with 2°C precision
•Something like 1 Hz testing rate (when consistently)
•Body size 15.5mm x 12mm x 5.5mm
•4 pins with 0.1" spacing.



Applications

This sensor is used in a variety of applications, including assessing wetness and temperature values in heating, ventilation, and air conditioning systems. These sensors are often used by weather stations to forecast weather trends. In households where people are affected by

stickiness, the mugginess sensor is used as a preventative strategy. This sensor is used in offices, automobiles, historical centres, nurseries, and endeavours to estimate mugginess values and as a security measure.

3.2.3 HEARTBEAT SENSOR:

The heartbeat sensor is relies on light plethysmography principles. It detects variations in blood volume in any organ in the body and produces changes in that organ's strength (the area of blood vessels). Pulse timing is particularly crucial in applications that track cardiac vital signs. Because the flow of blood volume is governed by the pace of the heart's pulse and light is absorbed by the blood, the signal's pulse is comparable to the heartbeat's pulse.

Photo plethysmography is available in two forms. Transmission: The solar radiation device's light penetrates through all of the body's blood vessels, including the earlobe, and is detected by the detector.

Reflection: The region reflects the light emitted by the sunshine emitting gadget.

A lightweight light emitting diode and detector, a type of light sensing resistor or photodiode, make up the basic heart rate sensor. The intestines' pulse generates variations in blood flow to various areas of the body. When a solar light source, such as an LED, illuminates tissue, the tissue either reflects (finger tissue) or transmits sunlight (ear bud). A portion of the sunlight is absorbed by the blood, and the photodetector also receives transmitted or reflected light. The quantity of blood in the tissue determines how many solar rays are absorbed. The detector's output is an electrical signal proportional to the average heart rate. In terms of tissue and blood volume, this signal is really a DC signal that synchronises with intestinal stroke and also superimposes the AC component induced by pulsatile changes in blood volume on the DC signal.

3.2.4 Buzzer

A buzzer or beeper is a mechanical, electromechanical, or piezoelectric audio instrument (piezo for short). Alarm clocks, timers, and confirmation of user input such as a depression or keyboard are all common uses for buzzers and beepers.

How to use a Buzzer

A buzzer might be a modest but inexperienced addition to our venture/sound system's capabilities. Because of its tiny and compact 2-pin form, it can be utilised on a breadboard, a Perf Board, or even PCBs, making it a widely used element in most virtual packages.

There are two sorts of buzzers that are regularly offered. The only confirmed right here is a simple buzzer that, when energised, makes a continual Beeeeeeppp sound; the other type is a premade buzzer that can seem bigger than this and can create a beep. Beep. Beep. It produces sound as a result of the oscillating circuit inside it. However, the only proved here is the most often used since it can be customised with the help of many circuits to fit perfectly in our application. This buzzer may be used by supplying it with a DC power supply ranging from 4 to 9 volts. A reliable 9V battery can also be utilised, however a controlled +5V or +6V DC supply is recommended. The buzzer is usually connected to a switching circuit that turns on or off the buzzer at predetermined times and intervals.

Types

Electromechanical

Early systems used an electromechanical mechanism that acted as a dead ringer for an electrical bell without the use of a metal gong. In the same way, a relay may be wired to stop its own actuating current, causing the contacts to buzz. These devices are frequently affixed to a wall or ceiling to serve as a sounding board**.**

Mechanical

A buzzer is an example of a purely mechanical buzzer that requires drivers to operate. Doorbells are another example of them.

Piezoelectric

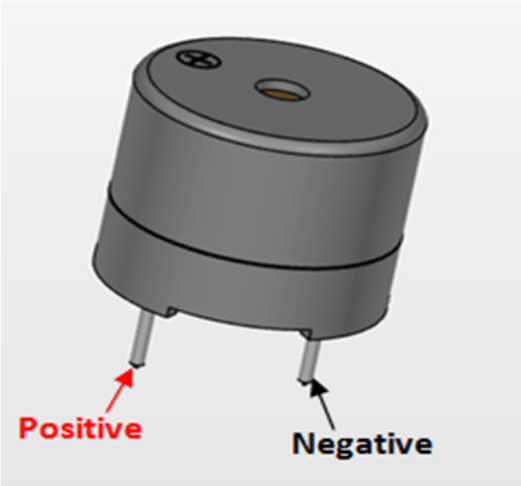
Piezoelectric disk beeper

An oscillating electrical circuit or another audio signal source can also drive a piezoelectric element, which is then powered by a piezoelectric amplifier. A click, a hoop, or a beep are frequent sounds that indicate that a button has been pressed.

inside a ready-made loudspeaker, with a piezoelectric-disk-beeper (With 3 electrodes... including 1 remarks-electrode (the significant, little electrode linked with purple twine for the duration of this photo) and an oscillator to self-power the buzzer

To produce an audible beep, a piezoelectric buzzer/beeper also uses acoustic hollow space resonance or Helmholtz resonance.

Buzzer Pin Configuration



Pin Number	Pin Name	Description
1	Positive	Identified by (+) symbol or longer terminal lead. Can be powered by 6V DC
2	Negative	Identified by short terminal lead. Typically connected to the ground of the circuit

Specifications

• 6V DC Rated Voltage

• Voltage range: 4 to 8 volts DC

• Sound Type: Continuous Beep

• Resonant Frequency: 2300 Hz

• Small and clean sealed box

• Rated current: 30mA

• Compatible with breadboards and perf boards

Application

• Communication gear

• Alarming circuits, where the user must be warned about anything

• Electronics in automobiles

• Due to its small size, portable gear

Pulse Oximeter

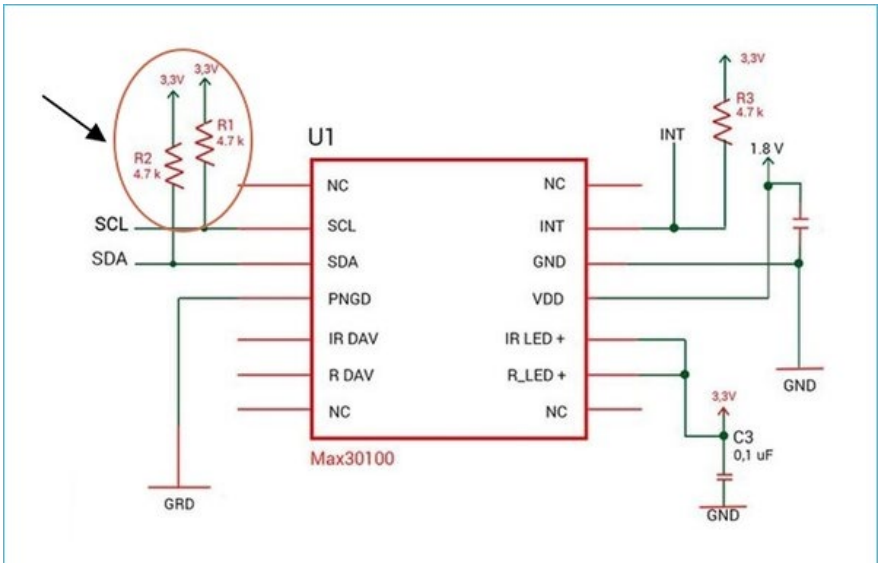
Pulse oximeters are low-cost, non-invasive medical sensors that detect the oxygen saturation (SPO2) of haemoglobin in the blood on a continuous basis. It shows the percentage of blood that is oxygenated.

The sensor combined with blood oxygenation, oxygen saturation and coronary transducer for an electrocardiogram (ecg). It is used to detect the coronary heart price signals and pulse and it also includes two LEDs, an image detector, enhanced optics and coffee noise analog signal sign progressing. It is running on 1.8V and 3.33V Software can cut down the energy sources with very minimal backup power, enabling its facilities source to stay in contact for as long as feasible.

Pulse oximeter Principle

The pulse oximeter's premise was founded upon differing penetration properties of the oxygenated and deoxygenated haemoglobin. Extra infrared emission is absorbed by oxygenated haemoglobin, allowing more red light to enter. Deoxygenated haemoglobin, on the other hand, it enables higher infrared rays to flow through while absorbing more red light.

Circuit diagram



What is in sensors interior?

Every detector probe for a pulse oximeter is made up of two small emitting diodes, one of which emits a photo detector as well as red and near-infrared light, are used. The amount of light that is transmitted through each spectrum is measured by photograph-detector. Blood contains oxygen material it deliberate employing fluctuations within the analysis. The probes are put over an appropriate person's body, most typically an ear lobe or a fingertip.

Unique approaches are conveying mild across the transmission media are employed to track oxygen saturation in the blood.

Transmission Method

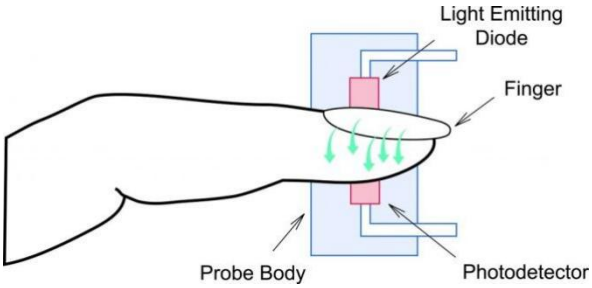
The transmitter, which is inserted on the other side of the finger, is used in the transmission technique. The finger is put in btwn the photo detector and the LEDs during this method. When this finger positioned, in light of the portion is take in the finger and a few elements passed by the image detector. To each cardiac heartbeat, the amount of blood float increases, which may cause high the fingertip will absorb radiation., resulting in low light reach a picture-detector.

As an outcome, the acquired mild signal's waveform will include highs during cardiac cycle and valleys (backsides) during every heartbeat. This mirrored image price allows blood to float deep down beat, which is the difference between the trough and the height fee.

Reflectance Method:

Light Emitted Detector and a picture in the Reflection technique, the detectors are situated along the same surface, adjacent to one another. Due to the finger, there will be a few consistent

weak reflections lower back to the sensor in the reflecting approach. With each heartbeat, the blood volume inside the finger expands, resulting in a brighter reflected picture being returned to the sensor.



The qualities of oxygen saturation hemoglobin in terms of red and Infrared light absorption are used in pulse oximetry. The proportion of red as well as IR light that hemoglobin absorbs will be used to measure blood oxygen content. The change in blood volume across the finger detects the pulse, which is also measured through the quantity of sunlight passing by fingertip.

A Chip has MAX30100 detects pulse oximetry and rate signals by combining the LEDs two are red and (infrared)IR a low noise signal processing and photo detector. The compressed data of each infrared and red mild saved as 64-byte incredibly buffer fifo. It has two modes of operation: guts fee mode and oxygen saturation and heartbeat based. The pulse price mode, just the infrared LED turned on, however the both Mid - infrared and red Led are switched on in dual mode. A 60 Hz reduced filter is also included built in. Even it was distinguish between current noise and ambient noise and variations, it cannot account for them.

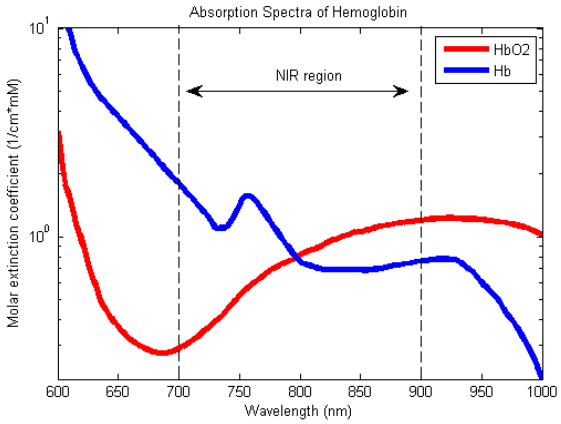
Light of red and ir are moved through the finger from LEDs, and the picture device incorporated into a chips detects combined uptake of a specific wavelength inside the sun. With the MAX30100, we employed both the oximetry and critical sign detection activities during this project. So we're able to locate both pulse fee and oxygen saturation on the equal time.

As a result, if we look at the waveform of the captured mild signal, we can see that it has peaks at every heartbeat. A hard and fast low-price reading is that this value may be considered stable reflection in between the centre beats, and the differ by the height deducted The reflecting charge due to severe drift far down beat is calculated from the steady reflecting value.

The troughs/peaks by reflected light appear at every pulse in each of the examples above, and the length between two spikes is commonly used to estimate the men and women's fee. As a result, a conventional heart beat sensor module consists of one photo-detector and one transmitter LED (mainly infrared).

So, how is oxygen saturation determined?

Now that you've grasped the various techniques of sensor placement for detecting the oxygen content in blood (SPO2), let's look at how it works in detail. As previously stated, the sensor has two transmitting LEDs: a red LED with a wavelength of 650 nanometers and an infrared LED with a wavelength of 950 nanometers.



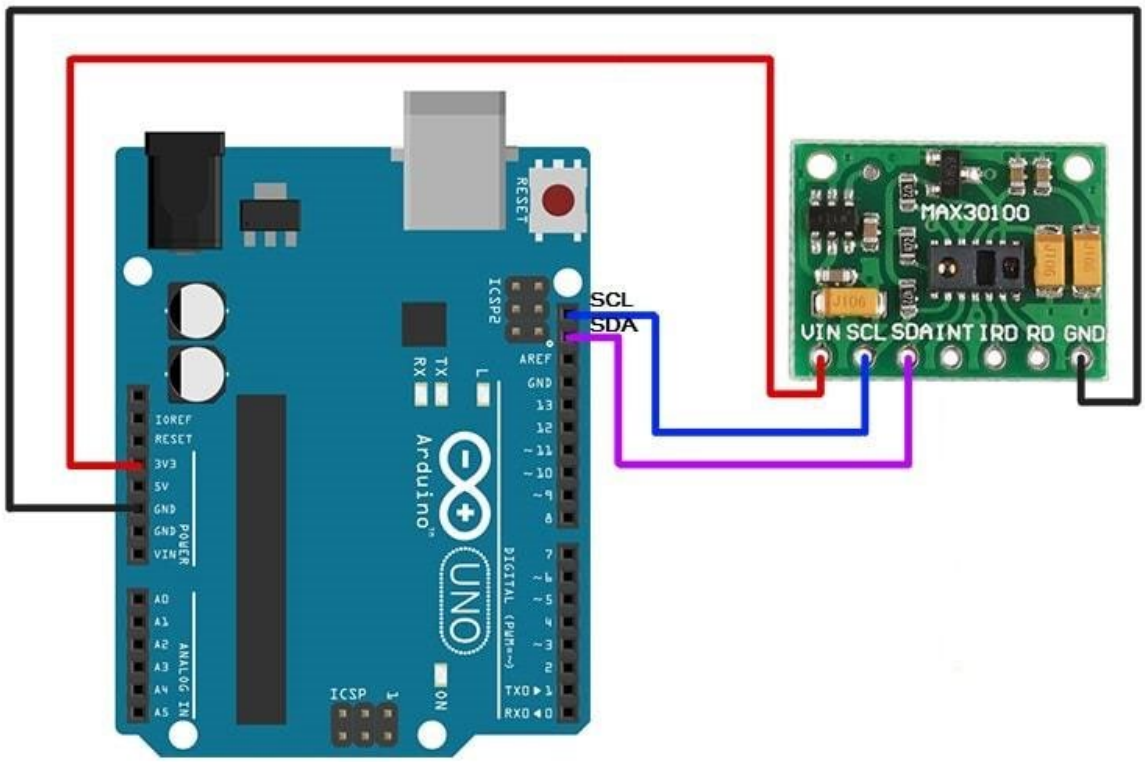
The chart above shows the absorption rate of oxygenation hemoglobin to different lighting wavelengths. It's possible to observe that the red line represents hydrated haemoglobin, which collects more ultraviolet than the blue line represents deoxygenated haemoglobin purple moderate haemoglobin. Furthermore, as compared to actinic rays, deoxygenated haemoglobin (blue line) absorbs more pink lights. It contains oxygen material The amount of sugar in the blood may be simply evaluated by determining what further sugar there is in the hemoglobin purple mild r is absorb in comparison the infrared mild. The proportion of oxygenated to deoxygenated haemoglobin determines the quantity of oxygenation or deoxygenated haemoglobin crimson mild absorbed versus infrared purple light RIR absorbing was fluctuate, it may be able to have a glance cup table that ration may be converted to spo2 cost maximum producers has the very our appearance to desk generally rir ratio of 0 five equates to approximately a hundred spo2 a ratio of one zero to about 82 spo2 even as a ratio of two zero equals to 0 spo2

Which sensor should be used to determine saturation oxygen?

So readers could see from the previous theory, we could simply create our personal oxygen saturation sensor by connecting an infra sender, a red light, and a mild determined resistors light detector to a controller, and calculating the balance of red moderate absorbing vs infrapink modest absorbed, as well as the spo2 charge.

Using an Arduino to interface with a max30100 vital sign sensor

Let us just attach the max30100 oxygen saturation sensor to with an Uno and view the data on a serial port. The connections and circuit diagram are provided here. Connect the max30100's vin pin to the Arduino's 5v or 33v pin ground pins to gnd, and the max30100's i2c connector scl sda to Arduino's A5 and A4.



3.2.5 MEMS SENSOR

Microelectromechanical systems (MEMS) are a type of microelectro Mems generation is one of the most advanced technologies used in the creation of many fashionable products, such as automobile impact airbags detectors and projector with processing chips. Professor R Howe initially outlined this concept in 1989, and numerous prototype was launched also changed, therefore it was essential component of a most recent items are accessibility in the today's market.

The mems chip had two components in its early stages: one included the chip's most important shape, and the other covered everything needed for signal conditioning. This approach didn't work out because the tool's overall space became larger, and the numerous components of one chip required multisampling techniques. The result of this tool has been less precise, and attaching a really machine was problematic. Main concept behind the era was used a set bases chips such as cantilever also diaphragm obtain to equal characteristics to an electronic circuit. Even with electrical technology, mems devices cannot be considered electronics circuits, thus the microfabrication method must be rewarded to achieve such a notion. Mechanical sides and holes are replicated by mems. However, it has a electronic circuits was fixed component foam is generate mems from silicon producers to has been thorough understanding of electronic mechanics as well as the materials used in the process.

MEMS Accelerometer

The piezoelectric accelerometer is one of the most commonly used, but it is cumbersome and not available for all operations. There are several different types of accelerometers on the market today, distinguished by whether the pressure is dynamic or static. The piezoelectric accelerometer is still one of the most often used nowadays. It may be further enhanced for multi-axis detection due to its tiny size and strong detection features. First one of this type has been further improved 25 years ago, but because to its tiny size and robustness, it was uncommon until lately, when there was a scarcity of large-scale industrial applications. detection characteristics.

The capacitive type mems accelerometer is one of the most often used mems accelerometers. Capacitive accelerometers are noted for their high sensitivity and temperature correctness. If 2

pieces are kept equal to one another and divided by a length of d, where e is the separators material's permeability and C is the capacitance created. is sometimes called capacitance.

A/d = EA/d C0 = E0.E

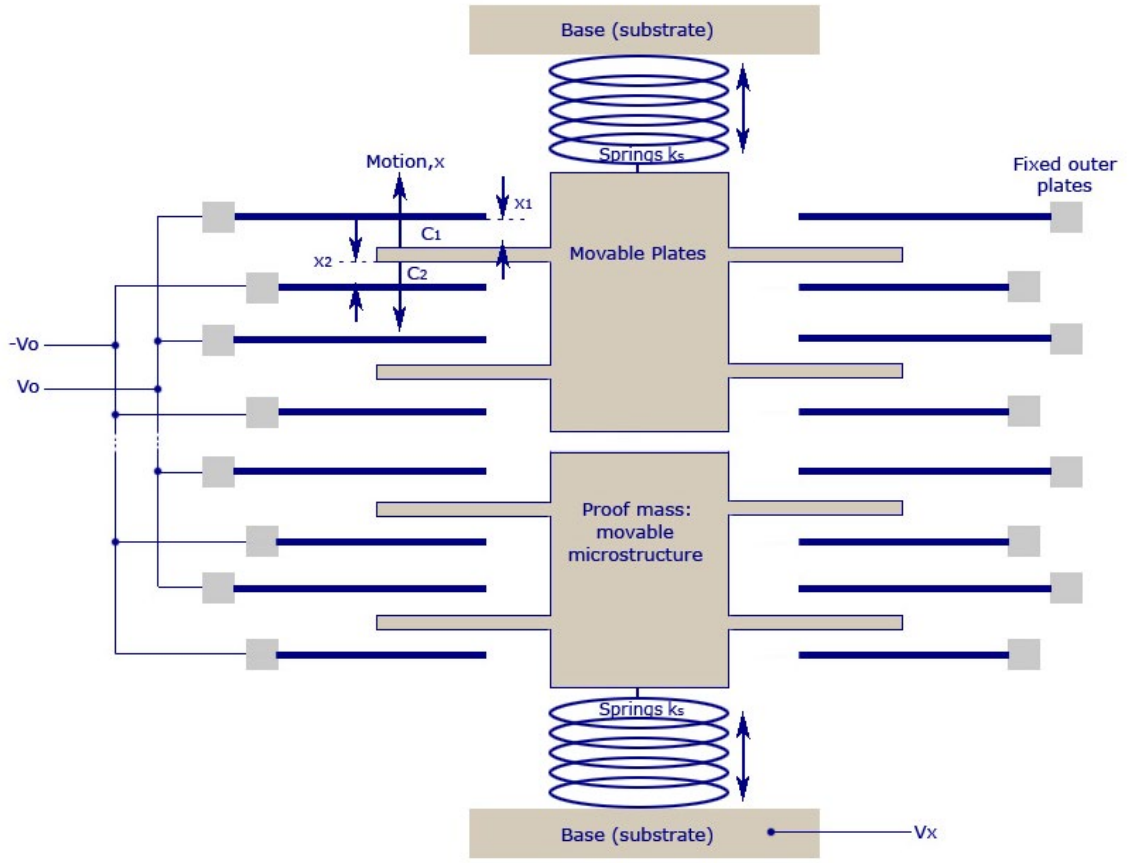
EA = E0EA = E0EA = E0EA = E0E

A – Electrode surface area

Changing the quantities of E, A, or d will help determine the capacitance change and, as a result, the MEMS transducer's performance. operation. The change in the values of d or A is what determines the accelerometer's readings.

A conventional mems accelerometer, commonly known as a simple single-axis accelerometer, may be made into a 2-axis or 3-axis accelerometer by holding additional capacitor sets at 90 degrees to each other.

Because they are connected to a mechanical suspension system, simple mems transducers consist primarily of movable microstructures or proof masses and are connected to a frame of reference. The movable plate and fixed outer plate function as a condenser plate as you accelerate as a result, the proof mass shifts, causing a capacitor in between moveable and fixed skins. The distance between adjacent plates varies as acceleration is applied x1 and x2, and we see that the distance between the poles.

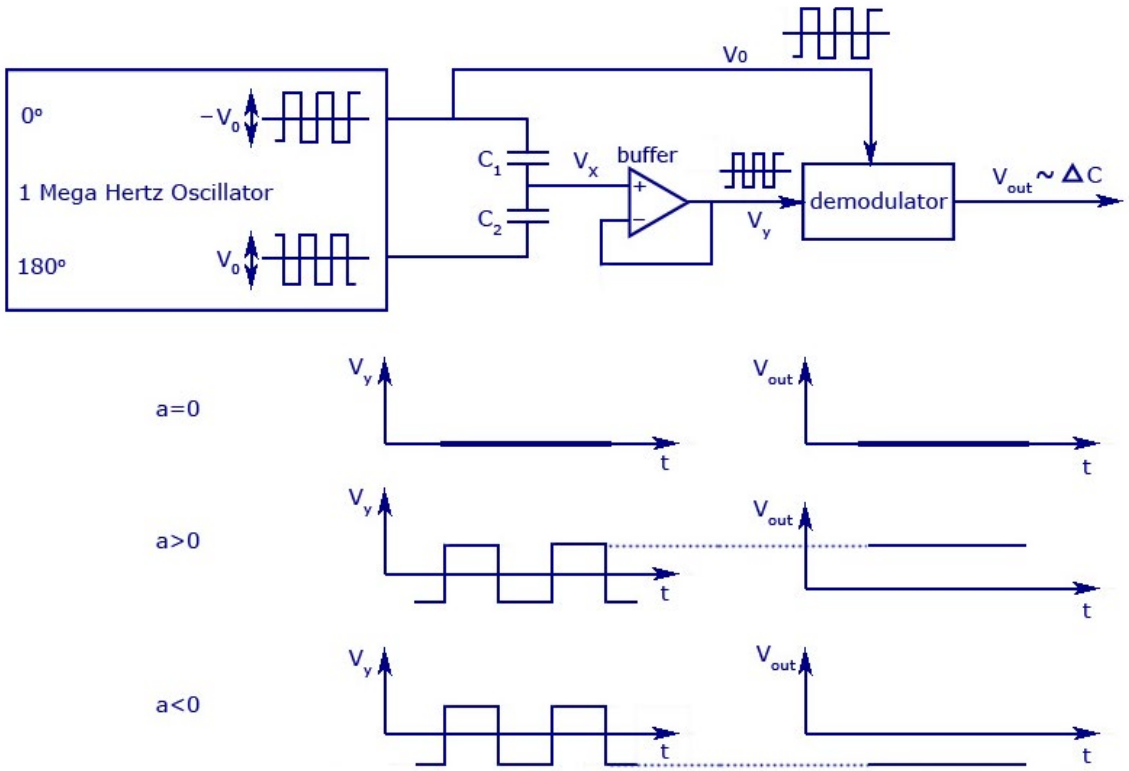


All sensors have numerous capacitor sets, as seen in the figure above. The higher capacitors are coupled in parallel to provide C1 total capacitance, while the lower capacitors produce C2.

If Vx is the proof mass's output voltage and V0 is the output voltage generated between the plates, then (Vx +V0) C1 + (Vx -V0) C2 = 0.

Vx =V0 [(C2-C1)/(C2+C1)] = (x/d) is another option. V0

The circuit that is used to compute acceleration by changing the spacing between capacitor plates is shown in the diagram below. The result generated for various acceleration settings is also graphically shown.



Acceleration Calculation - Capacitor Type MEMS Accelerometer

When then If there is no acceleration (a=0), the voltage output will be zero as well. Its quantity of Vx changes proportionately to the amount of V0 whenever accelerating is given, such as (a>0).

The signals Vx and Vy become negative when a deceleration is applied, such as (a0). Because It combines the quantities of Vy and V0 to get VOUT, that has the requisite accelerating sign and amplitude, and even the needed demodulator generates an output that is appropriate for the sign of the acceleration.

The proof mass weigh is unexpectedly modest due to the d the length of the chasm The output capacitance is roughly 20 aF, as well as the plates width is less than 1.3mm since the proving weight is much less than 0.1 microgram.

The gadget be chosen based on its noise characteristics. The noise characteristics might easily impair the accuracy of determining the acceleration value at low gravity conditions. The outputs, the voltage regulator circuit, and the oscillations caused by the springtime

measurement device are In a MEMS accelerometer, there are three noise-producing parameters.

Applications for MEMS Accelerometers

• MEMS sensors are being employed as step counts, programme control, and switching between multiple modes in the latest mobile phones and gaming joysticks.

• •The Nokia 5500 uses a 3D accelerometer to enable an easier tap and change function, which allows you to change mp3s by tapping on the phone while it's in your pocket

• Used to protect laptop disc drives from being destroyed if the computer falls to the floor. When the gadget detects a free fall, it instantly turns off the hard disc.

•where it detects abrupt negative acceleration and decides the appropriate moment to deploy the airbag.

Advantages

1. MEMS devices are extremely tiny and may be suitable for a variety of mechanical applications requiring big measurements.

2. The device's small size is help to reduce the cost.

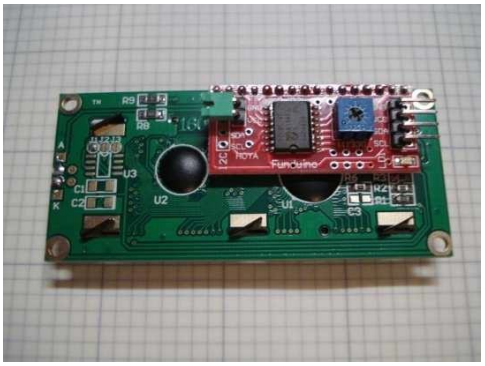
3. Two to Three distinct chips required the developed by certain things, microelectronics may typically combine them all onto a single MEMS chip. As a result, a single chip handles data receipt, filtering, storing, transmission, interface, and all other processes.

Applications

1. As an accelerometer, the gadget is a wide range of application, including airbag sensors and picture stabilisation in digital cameras.
2. Can be used as a pressure transducer to determine heart rate, multifarious stress (MAP), and tread pressure differences.
3. This is found in a wide variety of places, including gyros, DNA bits, and ink jet nozzles.
4. Devices, fibre optic switching, and other applications use optical MEMS.
5. RFMEMS is used to make transmitters, filtering, buttons, circuits, Random access memory

speakers, and other electronic devices other electronic devices.

3.2.6 Serial LCD Module I2C 1602:



PRODUCT FEATURES:

The I2C 1602 Liquid crystal display is a two-line, 16-character display with an I2C daughter board interface. To operate, the I2C interface simply requires two data lines, plus +5 VDC and GND.

CHAPTER-4
SOFTWARES USED

4.1 ThingSpeak

4.1.1 Introduction

ThingSpeak is an Internet of Things cloud platform that helps to transfer sensor data to cloud. Furthermore, you can ThingSpeak is indeed a cloud-based Internet of Things platform that allows you to send sensor data to the cloud. You'll also use MATLAB or other tools to process and analyse your data, as well as create your own apps.

The Thing Speak program is managed by MathWorks. You must either create a new MathWorks Account or log in to your current MathWorks Account to join up for ThingSpeak. ThingSpeak is available for no cost for small, non-profit applications.

ThingSpeak has a REST API for receiving and storing sensor cloud data as well as constructing IoT apps. It has pre-built libraries and APIs for Arduino, Raspberry Pi, and MATLAB, but because it uses a REST API and HTTP, it should work with any programming language that supports HTTP.

ThingSpeak TM is indeed a cloud-based IoT framework that helps you to gather, display, and analyse real-time data streams. ThingSpeak provides real-time representations of data delivered to it by your devices. Thanks to ThingSpeak's ability to run MATLAB code, you'll be able to study and process data in real time. ThingSpeak is a popular language for building IoT devices and prototypes that require insights.

4.1.2 ThingSpeak Key Features

ThingSpeak is perhaps a cloud-based framework for aggregating, visualising, and analysing real-time datasets. ThingSpeak includes a number of helpful features, including the ability to easily configure devices to transmit messages to ThingSpeak via common IoT protocols.

• Visualize data in real time from your sensors.

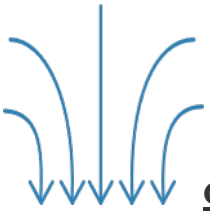
• On-demand data collection via other sources.

• Seem like you're using MATLAB to analyse your IoT data.

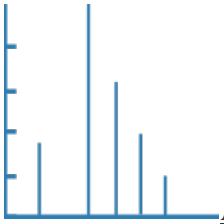
• Run IoT analytics automatically depending on scheduling or events.

•You may explore and construct IoT solutions without needing to set up servers or write web applications.

• React to your data and communicate automatically using third-party networks like Twilio® or Twitter®. Explore the subjects below to discover how to use ThingSpeak to gather, analyse, and act upon your IoT data:



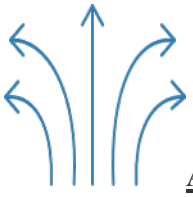
Collect



Analyze

Analyze and visualize your data with MATLAB.

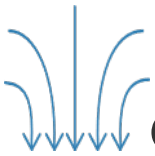
Send sensor data privately to the cloud.



Act

Trigger a reaction.

4.1.3 Work Flow



Collect

Transmit sensor data to the cloud in a secure manner.

Our houses, smartphones, autos, urban infrastructure, and machinery all have sensors. Temperature, humidity, and pressure are just a few of the variables that sensors measure. The information is then sent in some form, such as a numerical number or an electrical circuit.

Examine and display your data with MATLAB.

Data saved in the cloud may be accessed from any location. • You may use online analytical models to examine and visualise data.

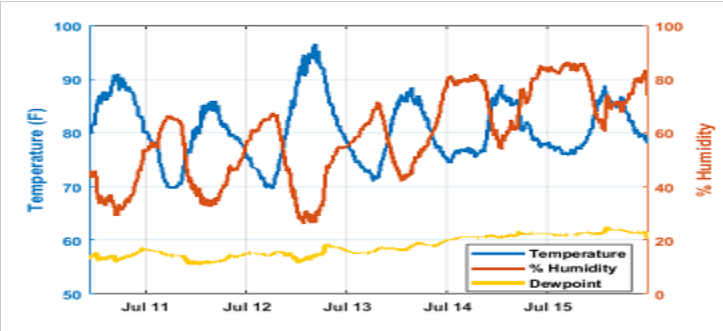
• You can uncover relationships, patterns, and trends in data.

• You have the ability to develop new information may visualise it using graph, chart, and indicators.

Why would you want to examine and display data in Thing Speak?

Thing Speak provides you access to MATLAB, that can help you with:

- Data analysis You can explore data linkages and transform, merge, and produce new data using built-in charting tools.
- You may even programme computations to take place at specified times.
- To generate a more complicated study, combine data from many sources.



Act

Trigger a reaction.

Acting on data can include receiving a tweet anytime the temperature you're monitoring gets over 70° F. You may even make a much more intricate operation, such as turning on a motor whenever the level of water in the storage tank falls below a certain threshold. The Talk Back app may also be used to remotely control items like rechargeable batteries and door locks.

Why do you want to react on data with Thing Speak?

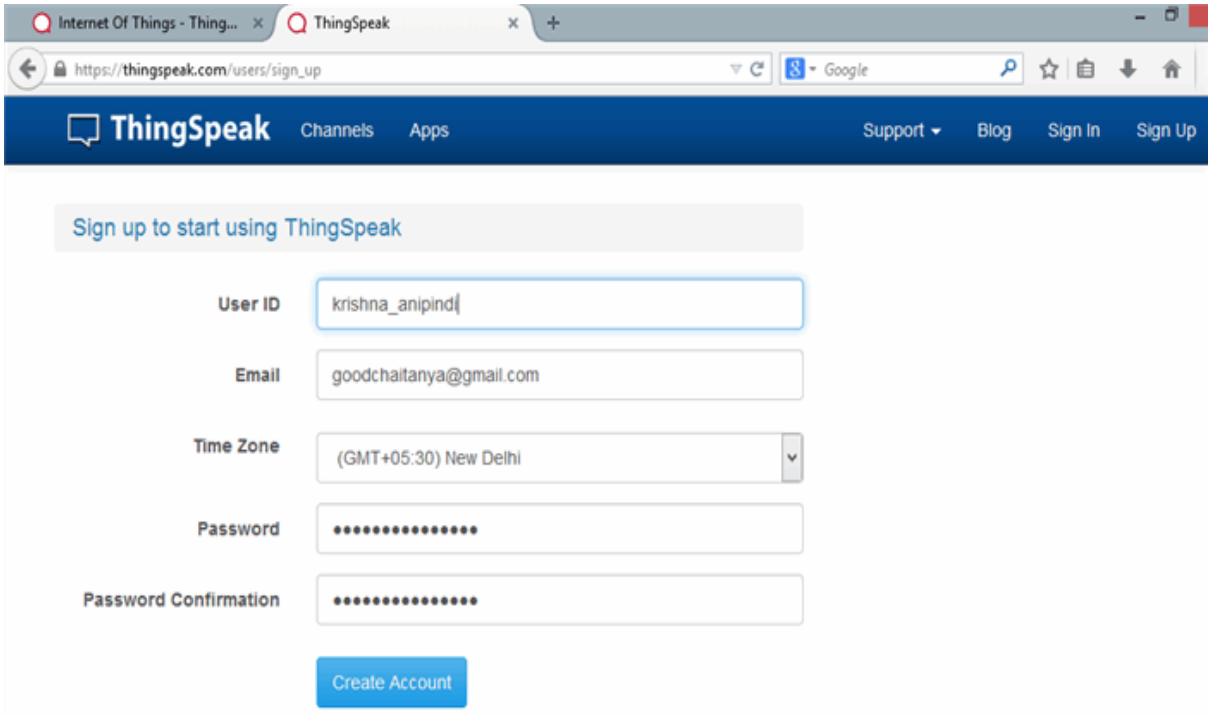
For all of these reasons and more, Thing Speak allows gadgets to interact with one another. You can:

• React to information as it reaches a channel, including actual data and calculated data; and

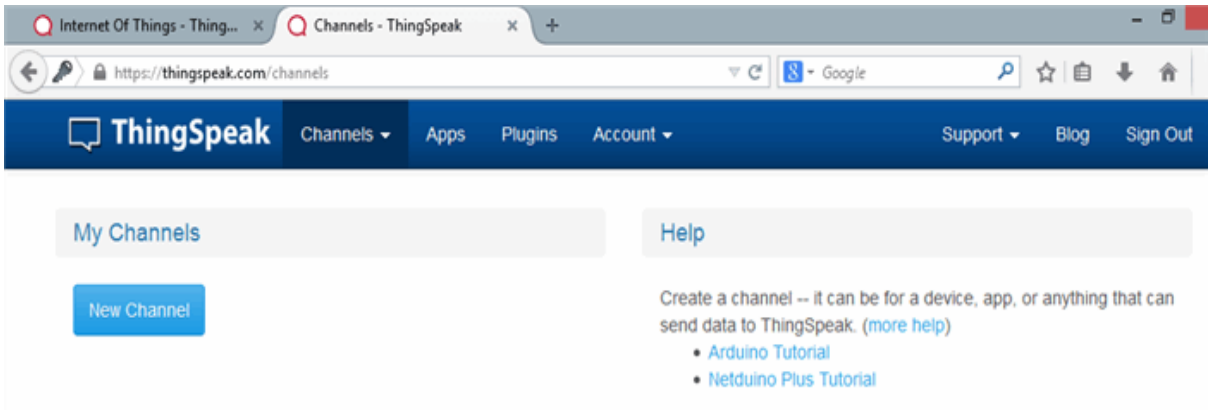
• Join the queue commands for a module to execute.

4.1.4 Getting Started with thingspeak

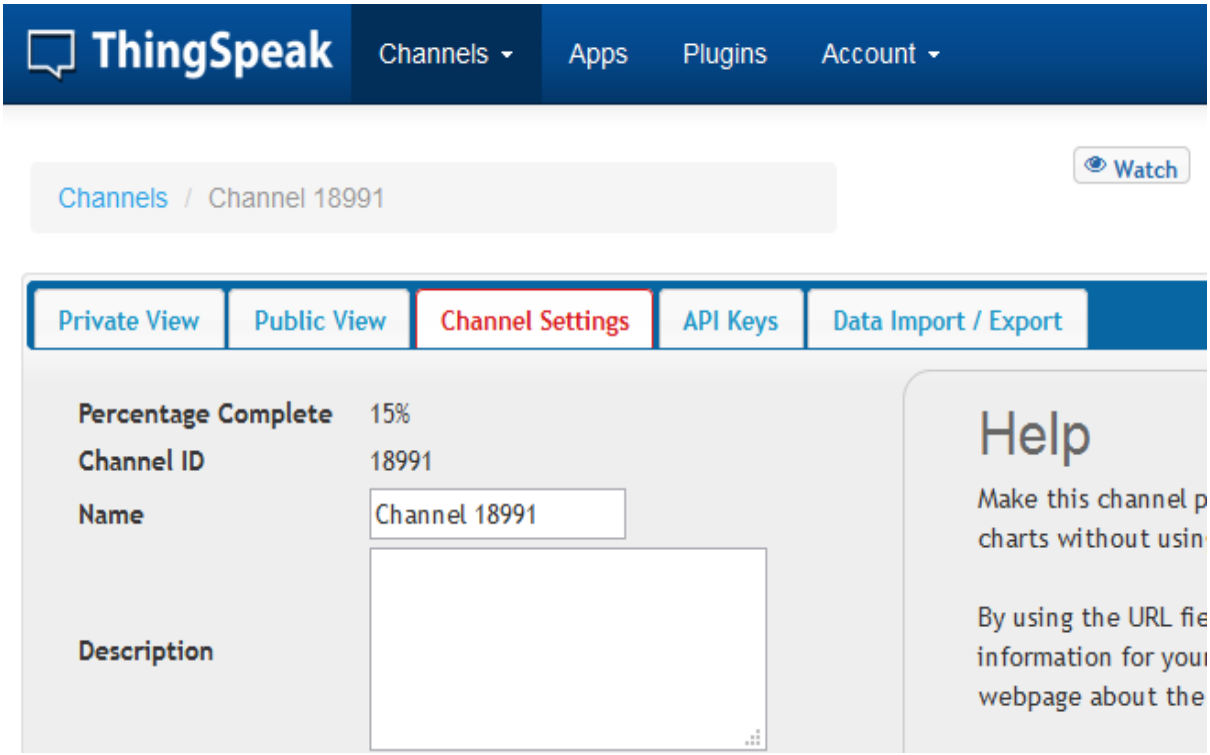
To access the sign-up page, go to https://thingspeak.com/ and select the 'Get Began Now' button inside the centre of the picture (the 'Sign Up' button on the right wing will lead you to the same page). Click on 'Create Account' button after entering the needed information.



You should be brought to a page confirming that the profile was created successfully. The message will disappear after several seconds, and the final part should appear like the one below.:

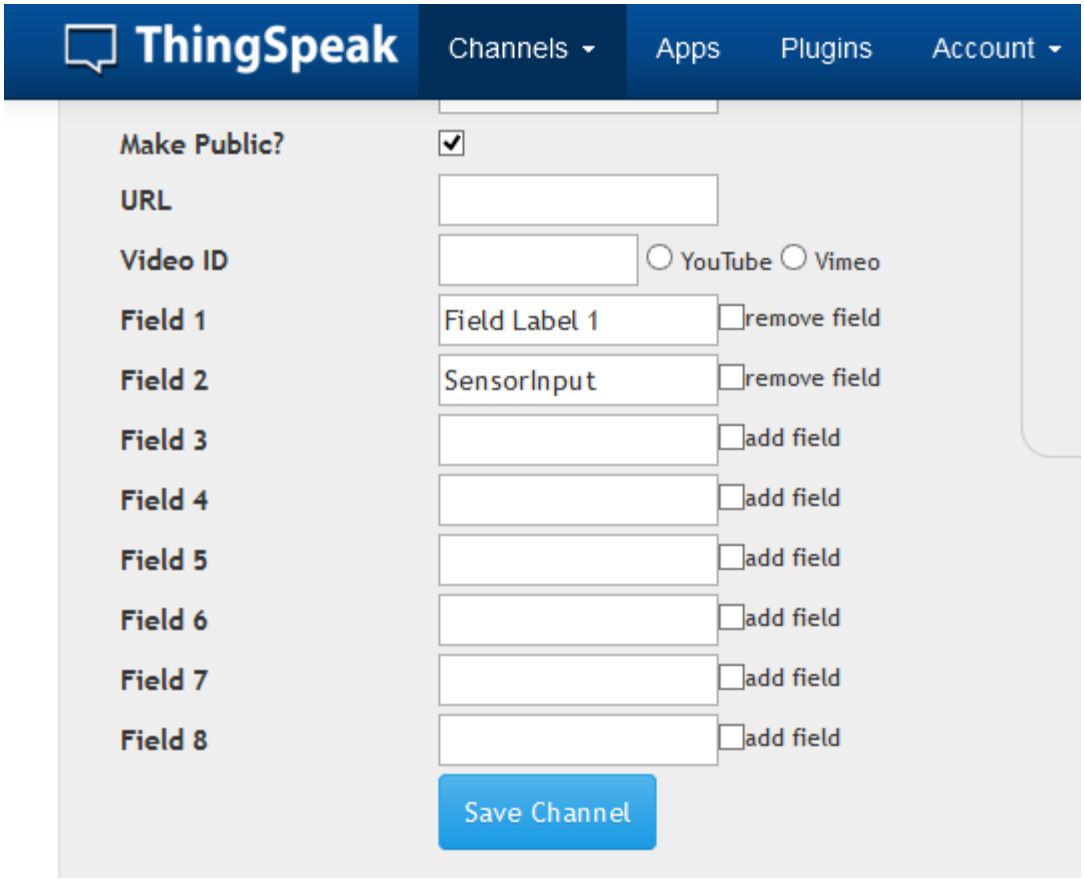


Go ahead and click on ‘New Channel’. You should see a page like the below:

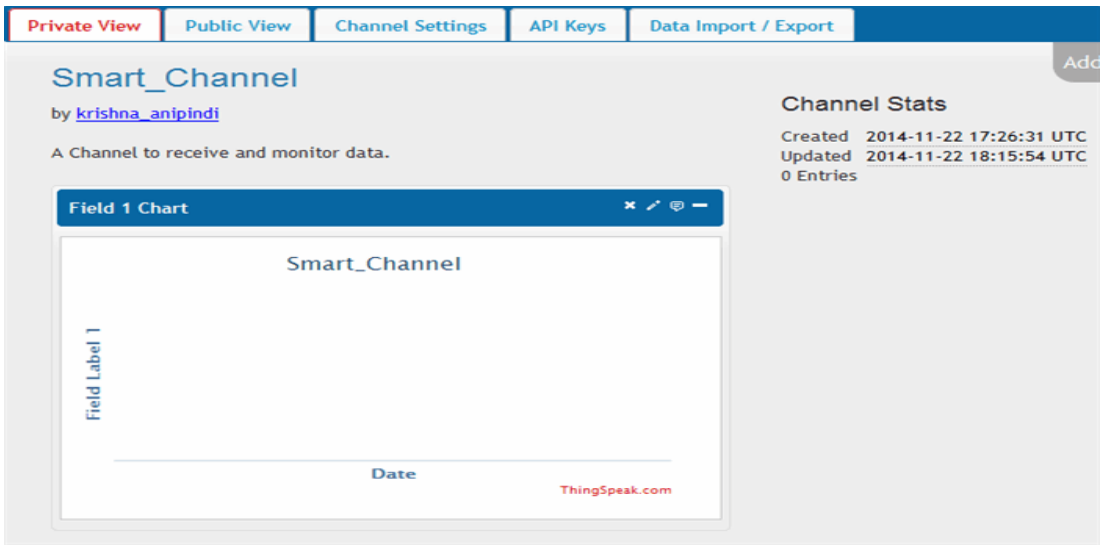


You may change the name and add an explanation to the channel to suit your needs. You can add any other relevant information to the metadata section. Latitude, Longitude, and Elevation fields should all be on the same page. As you scroll down, you should see a check box labelled 'Make Public?' Take into account the following consequences of the various fields and tabs:

- Latitude, longitude, and elevation - Those coordinates identify the position of a 'thing,' and are very useful when transporting goods.
- Is the channel open to the public? - If the channel is open to the public, anybody can view the stream of data and visualisations. If this checkbox is not checked, the channel would be private, and each read english operation will require an API key.
- URL - If you provide this, the URL of you website or blog will be exposed to the show's public view.Video ID - This is your video's unique identifier on Youtube. If the video is specified, it is broadcast to the whole globe via the stream. The data provided by a sensors or a 'object' is represented by fields 1 to 8.
- A field must be added before it can be used to hold data. Field 1 is automatically inserted. Your demand still will succeed if you try and post to parameters that your haven't added, but you can’t be capable to see the field within the graphs or the data connected with it. To add another field, click the little box beside the field's 'add field' text. The 'add field' wording changes to'remove field' when you select the 'add field' button, and a default labelling comment appears in the comments box corresponding to each field.



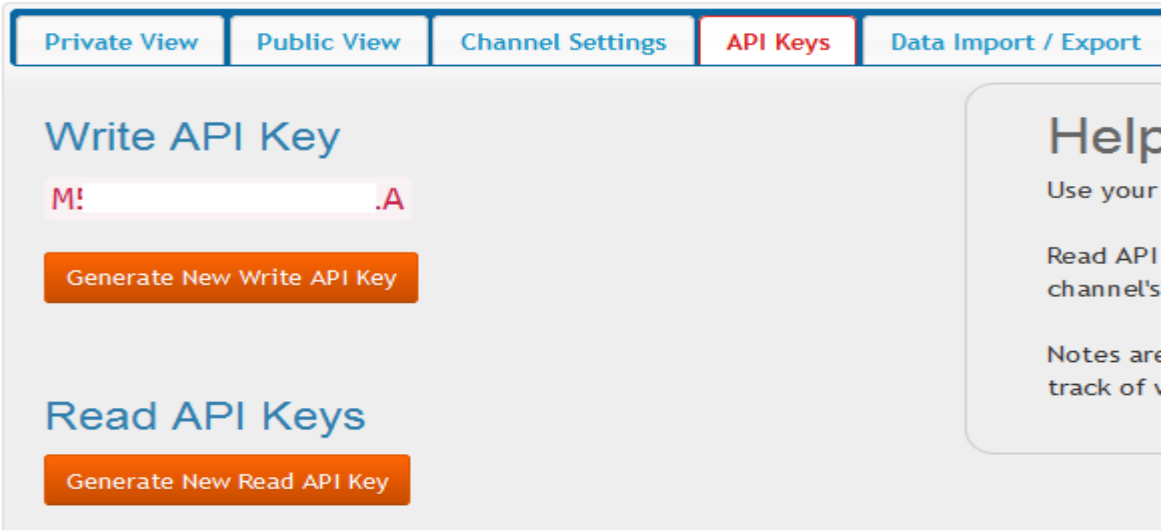
After you've finished editing the fields, hit the 'Save Channels' button. You should see a page similar to the one below, with the 'Secret View' tab selected by default:



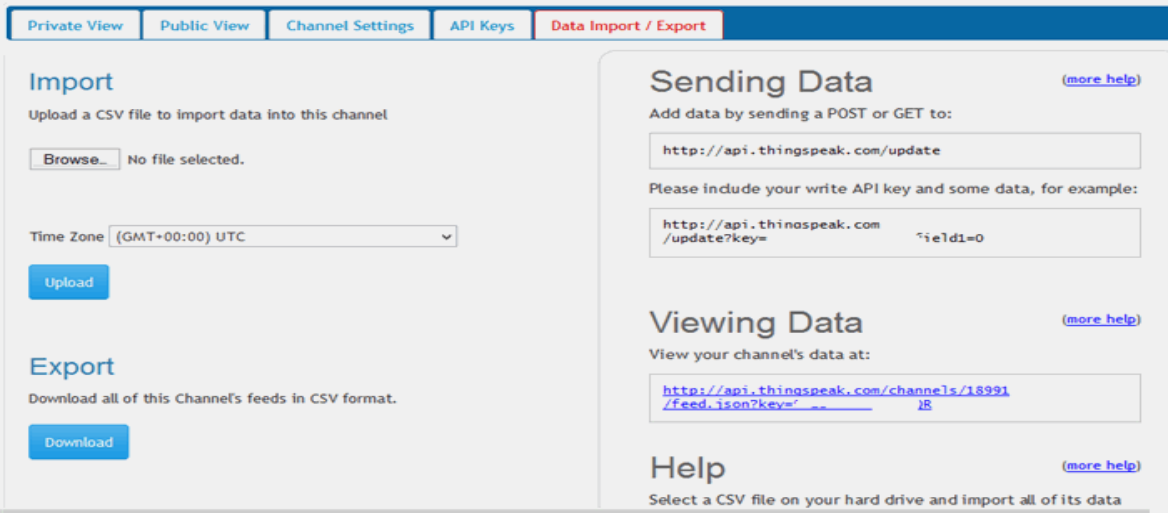
In the Private View, each of the variables we've added is represented by a chart. Then choose 'Public View' from the drop-down menu. Since our channel is public, this should look just like what we see on the 'Private View' tab. If the stream isn't truly public (the'make open' check box inside the 'channel settings' tab isn't selected), the notice 'This stream is not public' appears on the public view page.

Then choose 'API Keys' from the drop-down menu. A screen similar to this one should display. The receiving API key(s) are used to receive data from the channel, while the write API key(s) are used to submit data towards the channel.

The existing key will be overwritten if you click the 'Create New Write API Key' button. You will only have one Write API key at any one moment. Furthermore, only a Get API key may be used to access the show's feed and charts if the channel is private. Please only share your channel's Read API keys to people who have been vetted and given permission to watch it.



Now go to the 'Data Import/Export' panel and you'll see something like this. The 'Comma Separated Values (CSV)' data from a file is imported into the channel using this tab. You may also get the channel's feed in Csv file format from this page. This page also explains how to transmit and see data by using the send & view APIs' URIs.



Cons:

- Lack of powerful features which are included in some IDE's.

CHAPTER-5

TECHNOLGYS

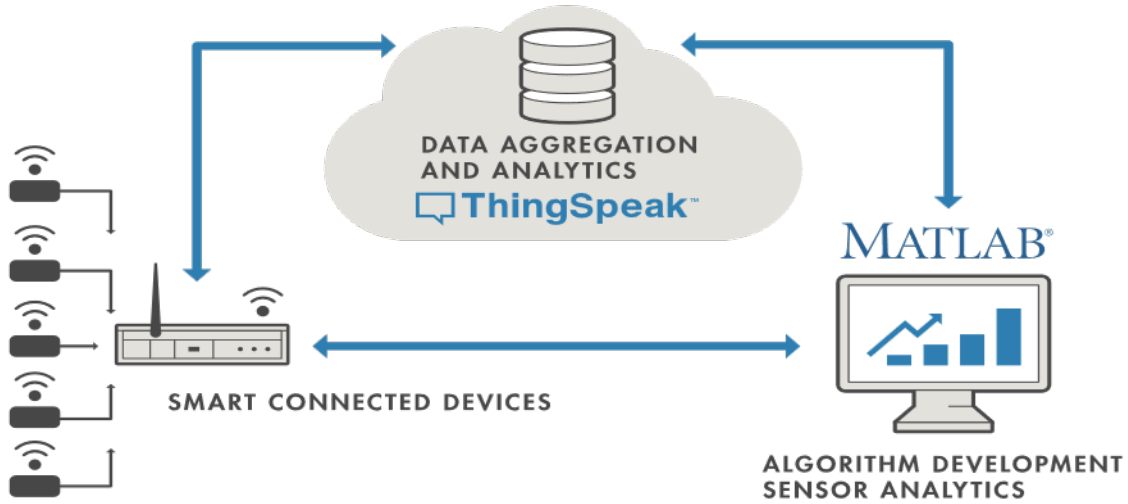
5.1 IOT

5.1.1Brief on IOT

The Internet of Things is the most recent fad where a very big portion of embedded systems (things) are linked to the Internet. These networked devices exchange data with people and other devices, and they frequently send sensor information to the cloud storage and internet computing power, where it is collected and analyzed for meaningful data. This trend is being aided by low-cost cloud computing services, as well as rising cloud computing usage.

To mention a few applications, IoT systems may be used to monitor systems, equipment, traffic tracking, commercial measuring and reviewing, and smart appliances.

To a large extent, the graphic below may be utilised to represent numerous IoT devices.



The devices that live at the network's edge may be seen on left. Portable gadgets, digital temperature monitoring, heartbeat monitors, hydrostatic pressure sensors, & manufacturing floor equipment are all examples of data collection devices. This cloud is at the centre of the system, gathering and analysing data from a variety of sources, generally using IoT advanced analytics built expressly for this purpose.

The algorithm development for the IOT platform is shown on the right side. By doing historic analysis of data, a scientist or data analyst attempts to grasp it. Data from IoT network is provided into a computer runtime environment in this scenario, allowing the architect or scientist to work with it.

5.1.2 Components of IOT

In a conventional Iot device, data is collected and exchanged in live time. An Iot device is made up of three components:

Smart Devices :

This is a device, including a television, surveillance camera, or exercise equipment, that's been given computational capabilities. It collects data from the environment, user input, and use trends, and transmits it via the internet to and from its Iot platform.

Internet of Things:

An Internet of Things application is a collection of functions and technology that brings together data from a variety of IoT devices. It uses deep learning or artificial intelligence tool to detect data and make informed decisions. These selections are communicated to the Iot system, which responds to the data smartly.

A graphical user interface:

A GUI can also be used to control one IoT system or a fleet of systems. A smartphone app or website for registering and controlling smart devices is a common example.

5.1.3 IoT technologies

IoT systems may employ the following technologies:

Edge computing

Edge computing enables smart devices to do much more than just transmit and receive information to their IoT platform. It increases processing capability at the edges of an IoT network, reducing transmission delay and accelerating response times.

Cloud computing

Cloud computing is used for distant data storage and IoT device management, allowing data to be accessed by many devices on the internet.

Machine learning

Machine learning refers to the software and processes that are used to evaluate data and make critical decisions based on that data. Such machine learning approaches may be applied in the cloud as well as at the network's edge.

5.1.4 Types of IoT

Consumer Internet of Things (CIoT)

Consumer IoT refers to the use of IoT in apps and systems (CIoT). Mobile phones, smartwatches, smart chatbots, home automation, and other IoT devices are widely used. In CIoT systems, connection is often offered by Wi-Fi, Bluetooth, as well as ZigBee. These technologies provide short-range communication, making them perfect for usage in smaller environments such as homes and offices.

Internet of Things for Business:

CIoT pushes things one step beyond by offering IoT benefits to larger venues, whereas CIoT aims to improve home and personal surroundings. Corporate office buildings, markets, department shops, resorts, health clinics, and amusement parks are really just just few cases. CIoT use applications include tracking climatic conditions, managing access to firm premises, and reducing energy and consumption costs in resorts and other large venues. Many industrial Internet of Things technologies attempt to improve user experiences and business transformations.

Industrial Internet of Things (IIoT)

The (IIoT) is the dynamic branch of the Internet of Things business. Its main purpose is really to boost existing industrial systems' effectiveness and competitiveness. IIoT deployments are most popular in big factories and industrial plants, and are frequently linked to industries including healthcare, farming, automotive, and logistics.Perhaps most well-known instance of IIoT is the Industrial Internet.

IoT Infrastructure:

Infrastructure IoT is related to the development of smart infrastructure to make use of IoT technology to cost cutting, efficiency gains, and management, among other things. This involves the capacity to track and regulate the functioning of rural facilities such as roads, rail lines, and floating offshore wind farms.

IoT in infrastructure is a subclass of IIoT. However, because of its importance, it's worth mentioning.

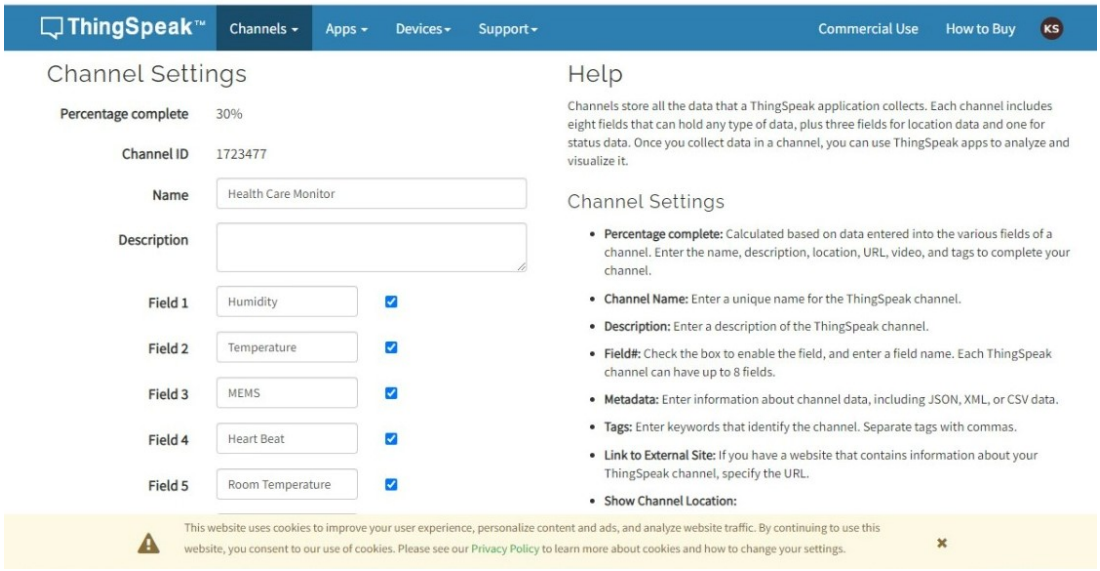
Internet of Military Things (IoMT)

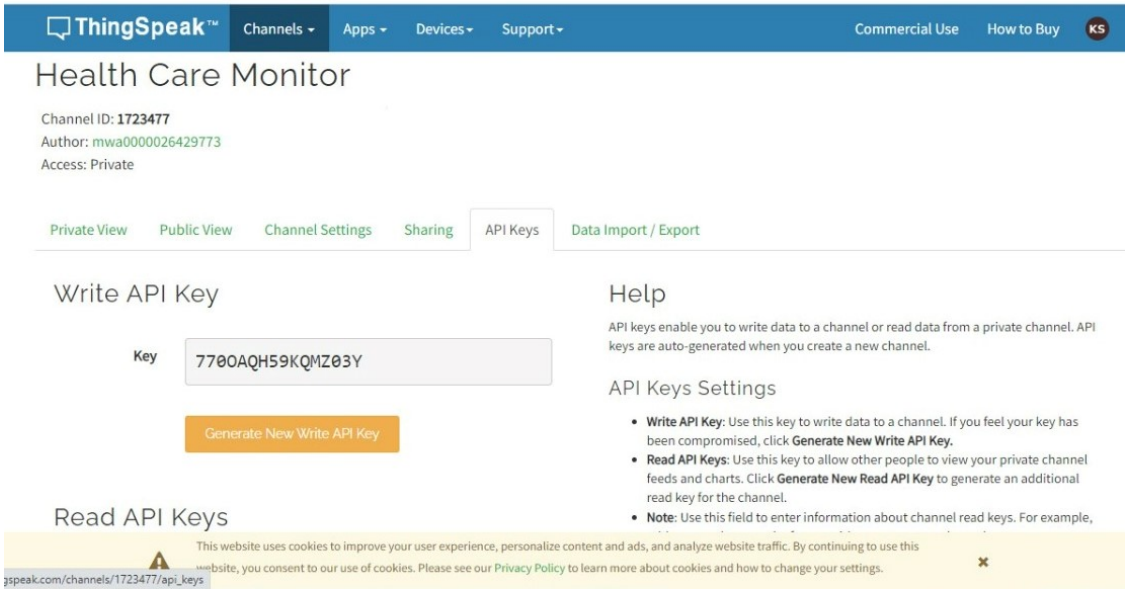
The Internet of Military Things (IoMT), also called as Battlefield IoT, Internet of Battle Things, or merely IoBT, is by far the most recent type of Internet of Things. The acronym IoMT stands for "Internet of Things in Military and Battlefield Situations," and it means precisely that. The organization's main goals are to increase tactical awareness, threat assessment, and reaction speed.

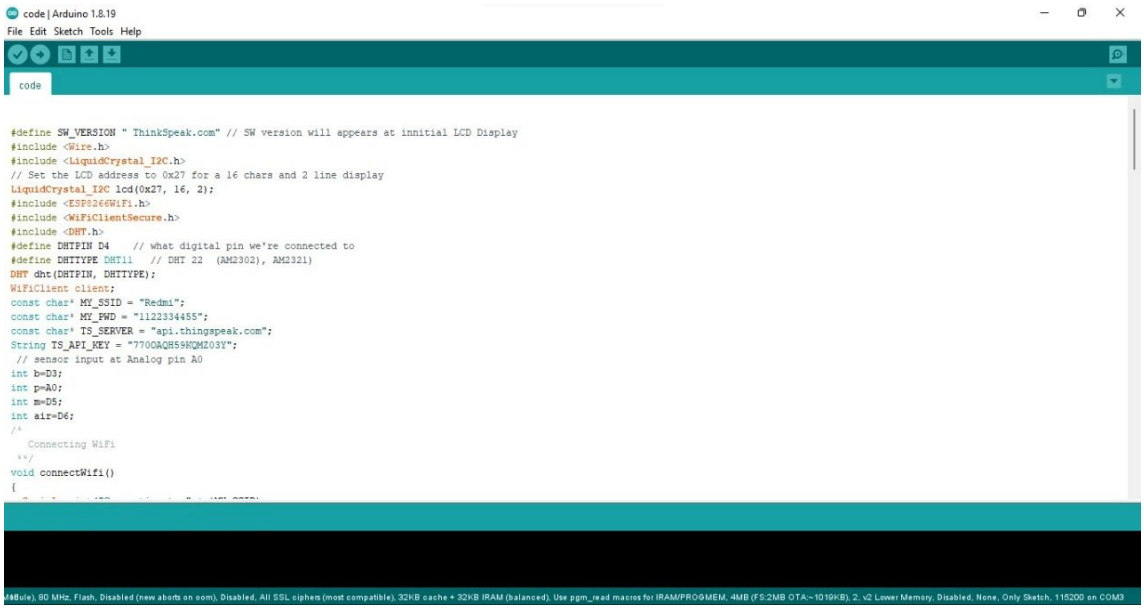
Using an integrated system to connect warships, aircraft, armored vehicles, troops, unmanned aerial vehicles, and sometimes even Advanced Airbases is a frequent IoMT use.. In addition, the Internet of Military Things generates data that can be used to improve military operations, systems, weapons, and strategy.

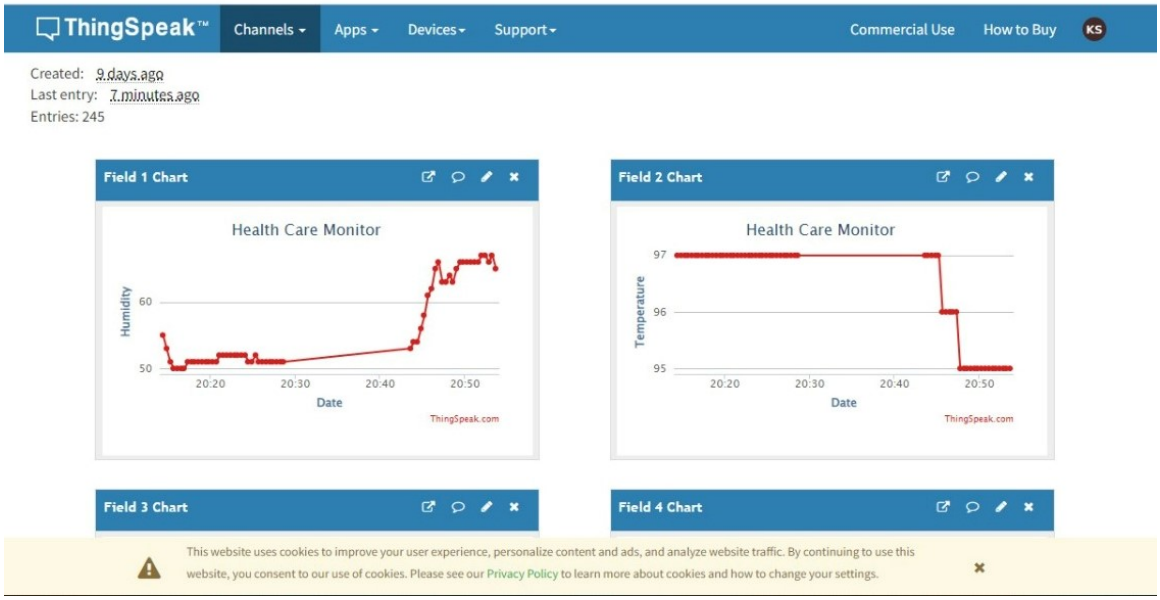
CHAPTER-6

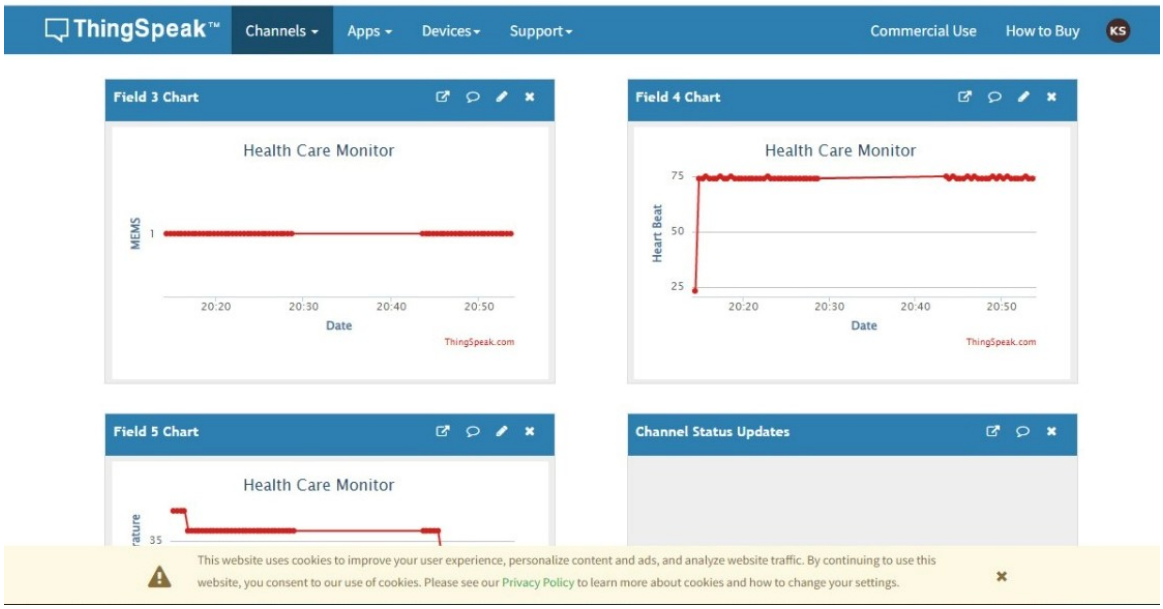
RESULTS AND GRAPHS

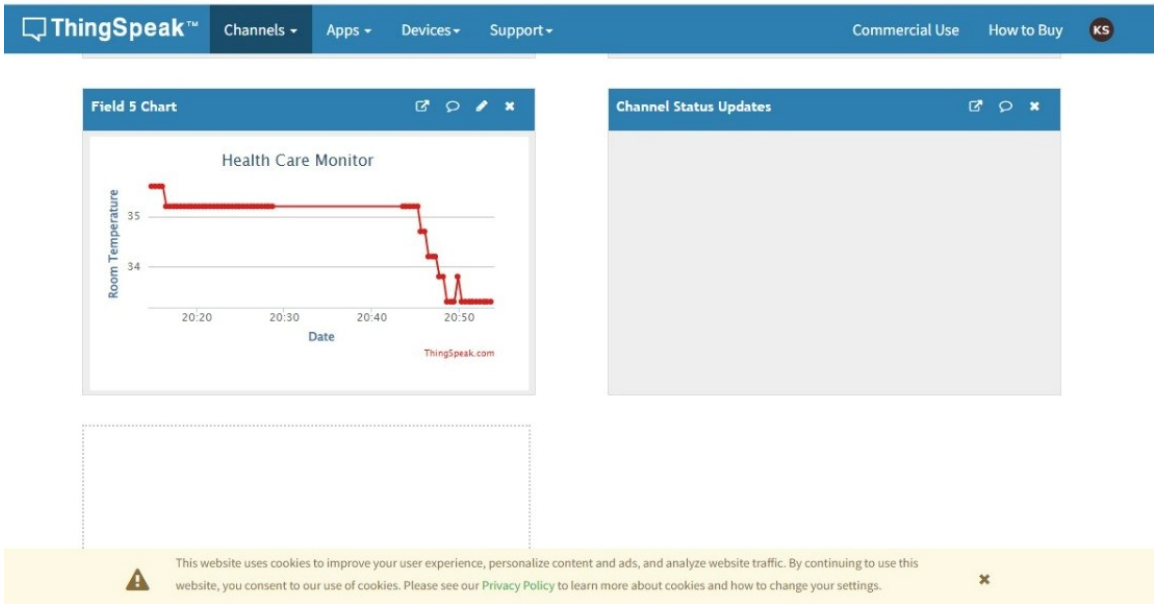





code | Arduino 1.8.19
File Edit Sketch Tools Help
code
#define SW_VERSION " ThinkSpeak.com" // SW version will appears at innitial LCD Display
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 16, 2);
#include <ESP8266WiFi.h>
#include <WiFiClientSecure.h>
#include <DHT.h>
#define DHTPIN D4 // what digital pin we're connected to
#define DHTTYPE DHT11 // DHT 22 (AM2302), AM2321)
DHT dht(DHTPIN, DHTTYPE);
WiFiClient client;
const char* MY_SSID = "Redmi";
const char* MY_PWD = "1122334455";
const char* TS_SERVER = "api.thingspeak.com";
String TS_API_KEY = "7700AQH59KQMZ03Y";
// sensor input at Analog pin A0
int b=D3;
int p=A0;
int m=D5;
int air=D6;
/*
Connecting WiFi
**/
void connectWifi()
{


ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
KS
Created: 9 days ago
Last entry: 7 minutes ago
Entries: 245
Field 1 Chart
Health Care Monitor
Humidity
60
50
20:20
20:30
20:40
20:50
Date
ThingSpeak.com
Field 2 Chart
Health Care Monitor
Temperature
97
96
95
20:20
20:30
20:40
20:50
Date
ThingSpeak.com
Field 3 Chart
Field 4 Chart
This website uses cookies to improve your user experience, personalize content and ads, and analyze website traffic. By continuing to use this website, you consent to our use of cookies. Please see our Privacy Policy to learn more about cookies and how to change your settings.


ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
KS
Field 3 Chart
Health Care Monitor
MEMS
1
20:20
20:30
20:40
20:50
Date
ThingSpeak.com
Field 4 Chart
Health Care Monitor
Heart Beat
75
50
25
20:20
20:30
20:40
20:50
Date
ThingSpeak.com
Field 5 Chart
Health Care Monitor
35
Channel Status Updates
This website uses cookies to improve your user experience, personalize content and ads, and analyze website traffic. By continuing to use this website, you consent to our use of cookies. Please see our Privacy Policy to learn more about cookies and how to change your settings.


ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
KS
Field 5 Chart
Health Care Monitor
Room Temperature
35
34
20:20
20:30
20:40
20:50
Date
ThingSpeak.com
Channel Status Updates
This website uses cookies to improve your user experience, personalize content and ads, and analyze website traffic. By continuing to use this website, you consent to our use of cookies. Please see our Privacy Policy to learn more about cookies and how to change your settings.

CHAPTER-7

CONCLUSION AND FUTURE SCOPE

The real time wireless health care monitoring system is described in this thesis as a result of the pursued research effort. This Real Time Health Care Monitoring System prototype was created and manufactured to wirelessly monitor people's health at any time and take preventative steps to avert death or disease. Wearable technology is used in this system so that individuals may feel comfortable without having to add anything to their regular attire that might make them feel self-conscious.

The suggested device has great sensitivity when measuring the patient's oxygen saturation and pulse rate, with an accuracy of up to 95% when tested on healthy people. It was double-checked with a commercially accessible pulse oximeter. During a surgery or while unconscious, these gloves can be used to monitor blood oxygen saturation and pulse rate. This system may also be utilised by the doctor to access the above-mentioned data while he is not in the hospital and medication the patient, and the records can be digitalized and accessible at any time. For those choices available on the Internet, the proposed framework permits a person to digitally preserve all documents/records related to health from birth to death. A corresponding graph is built at the backend, which is also available at the front end in the patient's account and may be shown to the physician for accurate diagnosis. The present application visualises data from the Firebase database in real time. The app might feature a model that can be trained on the fly utilising database data to forecast a patient's vitals for a future date. This allows the user to spot any abnormalities in their vital signs and take appropriate precautions. Future advancements in the realm of IoT devices and medical equipment advancements can be merged into this system to make it more generic and accessible to everybody.

By incorporating a global positioning system into this patient monitoring, the location information of a user may be gathered, which also aids in the safety monitoring of dementia sufferers. The goal for future development is to create a mobile application that allows users to conveniently access history data or reports from within the application. During a surgery or while unconscious, these gloves can be used to monitor blood oxygen saturation and pulse rate. This system may also be utilised by the doctor to access the above-mentioned data while he is not in the hospital and medication the patient, and the records can be digitalized and accessible at any time.

CHAPTER-8

REFERENCES

- 1.Rahaman A, Islam M, Islam M, Sadi M, Nooruddin S. Developing IoT based smart health monitoring systems: a review. Rev Intell Artif. 2019;33:435–40
- 2.Riazul Islam SM, Kwak Daehan, Humaun Kabir M, Hossain M, Kwak Kyung-Sup. The Internet of Things for health care: a comprehensive survey.
- 3.Lin T, Rivano H, Le Mouel F. A survey of smart parking solutions. IEEE Trans Intell Transp
- 4.Al-Ali AR, Zualkernan IA, Rashid M, Gupta R, Alikarar M. A smart home energy management system using IoT and big data analytics approach

5 .Zanella A, Bui N, Castellani A, Vangelista L, Zorzi M. Internet of Things for smart.

B20

ORIGINALITY REPORT

24%	16%	5%	17%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.make-it.ca Internet Source	3%
2	lastminuteengineers.com Internet Source	2%
3	Submitted to Gitam University Student Paper	2%
4	www.sersc.org Internet Source	1%
5	Submitted to Al Musanna College of Technology Student Paper	1%
6	Submitted to Institute of Research & Postgraduate Studies, Universiti Kuala Lumpur Student Paper	1%
7	www.pareteum.com Internet Source	1%
8	Submitted to Universiti Teknologi MARA Student Paper	1%

9	Submitted to Softwarica College Of IT & E-Commerce Student Paper	1 %
10	en.wikipedia.org Internet Source	1 %
11	Submitted to Arab Open University Student Paper	1 %
12	Submitted to International University - VNUHCM Student Paper	1 %
13	Submitted to Icon College of Technology and Management Student Paper	1 %
14	Submitted to Pennsylvania State System of Higher Education Student Paper	1 %
15	Submitted to Liverpool John Moores University Student Paper	1 %
16	www.instrumentationtoday.com Internet Source	<1 %
17	wikimili.com Internet Source	<1 %
18	Submitted to Amrita Vishwa Vidyapeetham Student Paper	<1 %

19	www.ijmetmr.com Internet Source	<1 %
20	Submitted to Vet Nurse Plus Student Paper	<1 %
21	Submitted to Amsterdam International Community School Student Paper	<1 %
22	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	<1 %
23	Submitted to SVKM International School Student Paper	<1 %
24	Submitted to University of Huddersfield Student Paper	<1 %
25	Submitted to Gulf College Oman Student Paper	<1 %
26	Submitted to Harare Institute of Technology Student Paper	<1 %
27	Submitted to Athlone Institute of Technology Student Paper	<1 %
28	Balaji V R, Jayanth A, Jaganath P, Durkesh Kumar R. "Implementation of IoT based Smart Remote for Disabled people", 2022 8th International Conference on Advanced	<1 %

	Computing and Communication Systems (ICACCS), 2022 Publication	
29	R. Suganya, N. Tamilarashan, A. Sharan, K. Vishnu. "Chapter 38 Wield NodeMCU IOT to Shelter Coal Mining", Springer Science and Business Media LLC, 2022 Publication	<1 %
30	Submitted to Teaching and Learning with Technology Student Paper	<1 %
31	Submitted to Universiti Teknologi Petronas Student Paper	<1 %
32	Submitted to Fr Gabriel Richard High School Student Paper	<1 %
33	Submitted to Multimedia University Student Paper	<1 %
34	Submitted to Polytechnics Mauritius Student Paper	<1 %
35	www.electroniclinic.com Internet Source	<1 %
36	Submitted to University of Salford Student Paper	<1 %
37	www.rcciit.org Internet Source	<1 %

38	Amit Kumar Kaiborta, Sarita Samal. "IoT based Voice Assistant for Home Automation", 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT), 2022 Publication	<1 %
39	Submitted to Nottingham Trent University Student Paper	<1 %
40	Submitted to University of Limerick Student Paper	<1 %
41	Submitted to University of Makati Student Paper	<1 %
42	www.dnatechindia.com Internet Source	<1 %
43	Submitted to Durban University of Technology Student Paper	<1 %
44	Submitted to Kakatiya Institute of Technology and Science Student Paper	<1 %
45	Submitted to University of Central Lancashire Student Paper	<1 %
46	www.makerguides.com Internet Source	<1 %

47	Submitted to Birla Institute of Technology and Science Pilani Student Paper	<1 %
48	Submitted to University Tun Hussein Onn Malaysia Student Paper	<1 %
49	Submitted to University of Newcastle Student Paper	<1 %
50	Submitted to Galileo Global Education Student Paper	<1 %
51	Submitted to International Hellenic University Student Paper	<1 %
52	dc-headshots.com Internet Source	<1 %
53	Submitted to BRAC University Student Paper	<1 %
54	Md Julhas Hossain, Md. Amdadul Bari, Mohammad Monirujjaman Khan. "Development of an IoT Based Health Monitoring System for e-Health", 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC), 2022 Publication	<1 %
55	agri.ckcest.cn Internet Source	<1 %

56	repository.sustech.edu Internet Source	<1 %
57	Submitted to Universiti Malaysia Perlis Student Paper	<1 %
58	Submitted to University of Wolverhampton Student Paper	<1 %
59	create.arduino.cc Internet Source	<1 %
60	www.makerstore.com.au Internet Source	<1 %
61	Submitted to Lovely Professional University Student Paper	<1 %
62	en.m.wikipedia.org Internet Source	<1 %
63	Submitted to Higher Education Commission Pakistan Student Paper	<1 %
64	www.igi-global.com Internet Source	<1 %
65	Submitted to Southampton Solent University Student Paper	<1 %
66	Submitted to Bayard Rustin High School Student Paper	<1 %
67	Submitted to University of South Australia	

	Student Paper	<1%
68	journals.pen2print.org Internet Source	<1%
69	"Advances in Computing and Data Sciences", Springer Science and Business Media LLC, 2019 Publication	<1%
70	Submitted to Sri Lanka Technological Campus Student Paper	<1%
71	components101.com Internet Source	<1%
72	link.springer.com Internet Source	<1%

Exclude quotes	On	Exclude matches	< 5 words
Exclude bibliography	On		