**IOT BASED ICU PATIENT MONITORING SYSTEM**

**ABSTRACT**

In contemporary healthcare, the integration of Internet of Things (IoT) technology has proven instrumental in enhancing patient care and monitoring. This paper introduces an innovative IoT-based ICU Patient Monitoring System designed for real-time tracking of critical health parameters such as temperature, body humidity, pulse rate, and blood pressure. The system utilizes Arduino microcontrollers for data acquisition and processing, coupled with the powerful cloud-based platform ThingSpeak for seamless data storage and analysis.

ThingSpeak's robust analytics capabilities allow healthcare professionals to visualize and analyze patient data in real-time. Alerts and notifications can be set up to notify medical staff of any anomalies or critical deviations from normal health parameters. The system provides a user-friendly interface accessible through web or mobile applications, facilitating remote monitoring and improving the efficiency of healthcare delivery.

**CHAPTER 1**

**INTRODUCTION**

In the rapidly evolving landscape of healthcare, the integration of cutting-edge technologies has become paramount in providing enhanced and personalized patient care. One such advancement is the Internet of Things (IoT), which has found profound applications in the realm of medical monitoring. This paper introduces an innovative IoT-Based ICU Patient Monitoring System designed to comprehensively assess and monitor critical health parameters in real-time, leveraging the capabilities of Arduino microcontrollers and the cloud-based platform ThingSpeak.

As patients in Intensive Care Units (ICUs) necessitate continuous and meticulous monitoring, the proposed system aims to address this imperative with a holistic approach. By incorporating sensors to measure key vital signs, including temperature, body humidity, pulse rate, and blood pressure, the system provides healthcare professionals with a comprehensive dataset for continuous health evaluation. The integration of Arduino microcontrollers not only facilitates efficient data acquisition and processing but also ensures the system's scalability and cost-effectiveness.

ThingSpeak, a robust cloud platform, serves as the backbone for data storage, analysis, and visualization. The platform's intuitive interface allows for seamless data management, enabling medical practitioners to monitor patients remotely and make informed decisions promptly. Moreover, the system is designed to generate alerts and notifications, ensuring that medical staff can respond swiftly to any deviations from normal health parameters.

This IoT-based ICU Patient Monitoring System holds the promise of transforming traditional healthcare practices by ushering in a new era of proactive and data-driven patient care. Through the amalgamation of Arduino's hardware capabilities and ThingSpeak's cloud-based analytics, the system not only ensures accurate and real-time health data but also opens avenues for telemedicine and remote healthcare management. As we delve into the details of this innovative solution, it becomes evident that its implementation could significantly contribute to the improvement of healthcare services, ultimately leading to better patient outcomes and a more efficient healthcare ecosystem.

**OBJECTIVE**

The objective of this paper is to introduce an innovative IoT-based ICU Patient Monitoring System, utilizing Arduino microcontrollers and ThingSpeak platform, aimed at real-time tracking of critical health parameters. It emphasizes enhancing patient care and monitoring through seamless data acquisition, storage, analysis, and visualization. The system's user-friendly interface enables remote monitoring, while alerts and notifications ensure timely intervention by medical staff in response to abnormal health parameter deviations.

**CHAPTER 2**

**LITERATURE SURVEY**

**2.1 Patient Health Monitoring Using Arduino Through Iot**

<https://www.researchgate.net/publication/341958678_PATIENT_HEALTH_MONITORING_USING_ARDUINO_THROUGH_IOT>

**ABSTRACT:** Health has prime importance in our day-to-day life. Sound health is necessary to do the daily work properly. Monitoring your beloved ones becomes a difficult task in the modern day life. Keeping track of the health status of the patient at home is a difficult task. Especially old age patients should be periodically monitored and their loved ones need to be informed about their health status from time to time while at work. An innovative system that automated this task with ease. System puts forward a smart patient health monitoring system that uses Sensors to track patient health and uses internet to inform their loved ones in case of any issues. This project aims at developing a system which gives body temperature, blood pressure and heart rate using LM35, pressure sensor and pulse sensor respectively.These sensors are interfaced with Micro Controller Arduino ATMEGA328. This Micro- Controller Arduino ATMEGA328 is provided with a programming written in embedded c language. The data is sent to the server through wireless data transmission. This wireless data transmission is done by ESP8266 module. Data is transmitted on IOT platform i.e. thing speak which generates the graphs of Temperature, Blood Pressure and Heart Beat over a period of time. Data visualization is done on Thing speak, so that record of data can be stored over period of time. This data is stored on web server, so that it can be seen to user who logged into the website using his username and its corresponding password.

**2.2** **IoT-based Portable ECG Monitoring System for Smart Healthcare**

[**https://ieeexplore.ieee.org/document/8934622**](https://ieeexplore.ieee.org/document/8934622)

**ABSTRACT:** Sudden and unexpected death due to heart failure is a major cause of mortality among middle aged and elderly people. An efficient heart monitoring system can find out the malformation of heart conditions and that can also be helpful in diagnose at critical ambience. Sometimes the distance between patients and doctors is the main barrier that people do not have access to quality health services and thus having trouble for their regular health examine. IoT-based healthcare monitoring system is one of the manifested application areas in medical science. As a significant access in diagnose heart disease ECG observing system is widely used. In this paper, a progressive method for ECG monitoring system based on Internet of Things (IoT) has been proposed. In this study, a system is designed to frequently monitor the Electrocardiogram (ECG) signal collected from patient's body using wearable sensors and the data is stored into the database which can be accessed by authorized personnel only. When any malformation is found an automatic email is sent to the users and doctors for analyzing about the critical conditions of the patients and provides emergency health assistances. In order to verify the authenticity of this system tests have been implemented on several patients and the report shows that, this system is dependable and efficient for collecting real time ECG data which can be very helpful in diagnose heart diseases. This IoT-based low cost device can be reliably used to reduce the risk of disability and mortality rate due to cardiovascular diseases.

**2.3** **IoT Based Health Monitoring System using Blynk App**

[**https://www.ijeat.org/wp-content/uploads/papers/v8i6/E7467068519.pdf**](https://www.ijeat.org/wp-content/uploads/papers/v8i6/E7467068519.pdf)

**ABSTRACT:** The common healthcare benefits with crafty decisions, execute brilliant communications technologies that can produce resourceful communities. Health plays a major role in our daily routine. Real-time health monitoring for initial detection of life alarming diseases through advanced sensing and communication technology usually contribute advance treatment to save the lives of patients. The main aim of this project is to develop a system which will give body temp, heart rate using DS18b20 and pulse sensor as well. Plus the Controller ESP32 Dev board is interfaced with the sensors. Wireless data transmission is processed using Arduino through the Wi-Fi module. The controller ESP32 is utilized for Wireless data transmission on IOT technique using an android app BLYNK. Visually digitalizing the data on a Blynk Android App. Patient’s record of data will be stored over a period of time. The information is stored using an Android app in order to control the device using the app

**2.4** **IoT-based Low Cost Healthcare System for Cardiac Patient of Bangladesh using Bio-sensors**

[**https://ieeexplore.ieee.org/document/9290531**](https://ieeexplore.ieee.org/document/9290531)

**ABSTRACT:** From recent statistics, it is observed that the number of patients due to cardiovascular diseases (CVD) are increasing significantly. Specially, peoples of under developed and developing countries are greatly affected by CVD. Due to the incremental rising of cardiac patients for CVD in Bangladesh, death from sudden heart failure also increased considerably. Therefore, for ensuring rapid treatment and monitoring of a CVD patient, this paper proposes a low cost healthcare system for the CVD patients of Bangladesh based on the Internet of Things (IoT) and Bio-sensors which will help the doctors to monitor a cardiac patient constantaniously in hospital or home from remote location. In this system, the hardware part consists of three different sensors for collecting the temperature, ECG, and pulse rate data from patient body. Then, the collected data can be sent to a cloud server for storage using local Wi-Fi network, and also a website is designed to monitor the measured data by the doctor. In case of any critical condition, the doctor and some family members of the patient can be notified using a messaging mechanism. After successful implementation of the proposed system, it works properly for CVD patients. From cost analysis, it can be seen that the cost of the proposed system looks cheaper than some other existing devices, and it will cost within 80to100 which is comparatively cheaper for a low income country like Bangladesh.

**2.5** **Design and Development of Patient Monitoring System**

[**https://iopscience.iop.org/article/10.1088/1757-899X/226/1/012094#:~:text=The%20system%20consists%20of%20a,to%20PC%20via%20Arduino%20microcontroller**](https://iopscience.iop.org/article/10.1088/1757-899X/226/1/012094#:~:text=The%20system%20consists%20of%20a,to%20PC%20via%20Arduino%20microcontroller)**.**

**ABSTRACT:** Patient monitoring system allows continuous monitoring of patient vital signs, support decision making among medical personnel and help enhance patient care. This system can consist of devices that measure, display and record human's vital signs, including body temperature, heart rate, blood pressure and other health-related criteria. This paper proposes a system to monitor the patient's conditions by monitoring the body temperature and pulse rate. The system consists of a pulse rate monitoring software and a wearable device that can measure a subject's temperature and pulse rate only by using a fingertip. The device is able to record the measurement data and interface to PC via Arduino microcontroller. The recorded data can be viewed as a historical file or can be archived for further analysis. This work also describes the preliminary experimental results of the selected sensors to show the usefulness of the sensors for the proposed patient monitoring system.

**CHAPTER 3**

**EXISTING SYSTEM**

The present ICU patient monitoring system incorporates RF (Radio Frequency) communications to enable wireless data transmission between patient sensors and central monitoring stations. These systems utilize a network of sensors placed on the patient's body to continuously monitor vital signs such as heart rate, blood pressure, oxygen saturation, and respiratory rate. The collected data is transmitted via RF signals to centralized monitoring stations, providing real-time updates to healthcare providers. While RF communication offers increased mobility and flexibility, drawbacks include potential interference from other electronic devices, limited range in some environments, and susceptibility to signal loss or degradation in certain conditions like electromagnetic interference or structural barriers. Additionally, concerns about data security and privacy may arise due to the wireless transmission of sensitive patient information. Despite these drawbacks, RF-enabled ICU patient monitoring systems enhance patient care by providing timely access to vital information and facilitating remote monitoring.

**DRAWBACKS**

* RF signals may face interference from other electronic devices, potentially leading to signal disruption and inaccurate data transmission.
* RF communication has a finite range, restricting mobility and requiring additional infrastructure for coverage in larger facilities.
* Environmental factors like electromagnetic interference or structural barriers can cause signal loss, compromising data transmission reliability.

**CHAPTER 4**

**PROPOSED SYSTEM**

The proposed system incorporates various sensors to continuously monitor the patient's vital signs, ensuring a comprehensive health assessment. A temperature sensor measures body temperature, a humidity sensor records body humidity levels, a pulse rate sensor captures heart rate, and a blood pressure sensor monitors blood pressure variations. The Arduino microcontroller processes the acquired data and transmits it securely to the ThingSpeak cloud platform using Wi-Fi connectivity.

To facilitate proactive and quick treatment by enabling rapid communication between healthcare staff. To enhance patient safety and reduce the possibility of permanent disability or death due to delayed decision-making. To provide a cost-effective solution compared to other healthcare systems by utilizing IoT technology and existing ICU devices. To enable remote monitoring and communication, allowing healthcare staff to access patient data and make informed decisions from anywhere. To collect and analyze data from both healthy and sick patients to validate the effectiveness of the system in real-world scenarios. Overall, the proposed work aims to leverage IoT technology to create a smart patient monitoring system that improves the quality of care in the ICU and enhances patient outcomes.

**ADVANTAGES:**

* Cost-effectiveness: The proposed system is designed to be cost-effective compared to other healthcare systems. By utilizing existing ICU devices and IoT technology, it offers a more affordable solution for patient monitoring and management.
* Enhanced patient safety: With continuous monitoring and timely alerts, the proposed system enhances patient safety in the ICU. It reduces the possibility of errors or delays in decision-making, ensuring that patients receive the necessary care promptly.

**METHODOLOGY:**

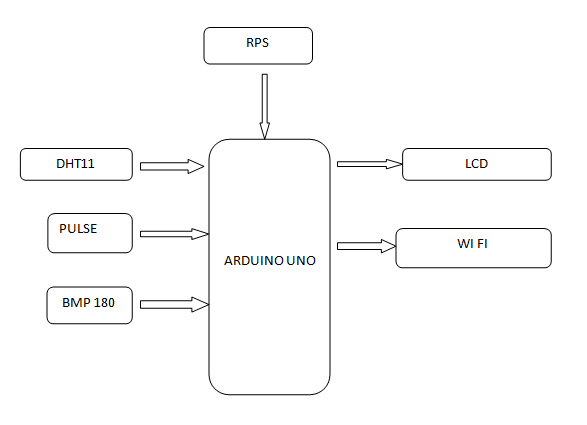
The methodology for developing the IoT-based ICU Patient Monitoring System involves several key steps. Firstly, the system architecture is designed, outlining the integration of hardware components such as the Arduino Uno microcontroller, DHT11 sensor for temperature and humidity monitoring, pulse sensor for pulse rate measurement, BMP180 sensor for blood pressure monitoring, WiFi module for internet connectivity, LCD display for real-time data visualization, and a reliable power supply (RPS). Secondly, the hardware setup is executed, ensuring proper connections and wiring between the components to enable seamless data transmission and processing.

Next, the software implementation phase involves programming the Arduino Uno to interface with each sensor and collect data at regular intervals. This includes utilizing libraries and code snippets for sensor communication and implementing error handling mechanisms for data accuracy and reliability. Data acquisition and processing algorithms are developed to convert raw sensor data into meaningful health parameters, such as temperature, humidity, pulse rate, and blood pressure.

The WiFi module is integrated with the Arduino Uno to establish a connection to the internet, enabling data transmission to the cloud-based platform ThingSpeak. A ThingSpeak channel is set up to receive and store patient data, and the Arduino Uno is configured to send collected data to the channel using appropriate APIs or protocols. Real-time monitoring and analysis capabilities are implemented using ThingSpeak's analytics tools, allowing healthcare professionals to visualize and analyze patient data in real-time. Alerts and notifications are configured to notify medical staff of any anomalies or critical deviations from normal health parameters.

A user-friendly interface is developed for accessing and interacting with the patient monitoring system, accessible through web or mobile applications. Compatibility with different devices and screen sizes is ensured, facilitating remote monitoring and improving the efficiency of healthcare delivery. Rigorous testing and validation are conducted to ensure functionality, reliability, and accuracy, with comprehensive documentation prepared covering system architecture, hardware setup, software implementation, and user guidelines. Finally, the IoT-based ICU Patient Monitoring System is deployed in real-world healthcare settings, with proper training and support provided to medical staff.

**4.1 BLOCK DIAGRAM**

****

**CHAPTER 5**

**HARDWARE TOOLS**

* ARDUINO UNO
* DHT11
* PULSE SENSOR
* BMP 180
* LCD
* WI FI MODULE
* REGULATED POWER SUPPLY

**5.1 DESCRIPTION OF TOOLS**

**1. ARDUINO UNO**

The most common version of Arduino is the Arduino Uno. This board is what most people are talking about when they refer to an Arduino. The Uno is one of the more popular boards in the Arduino family and a great choice for beginners. There are different revisions of Arduino Uno, below detail is the most recent revision (Rev3 or R3).

The Arduino Uno is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

Microcontroller   : ATmega328

Operating Voltage     : 5V

Input Voltage (recommended)             : 7-12V

Input Voltage (limits)                           : 6-20V

Digital I/O Pins                                     : 14 (of which 6 provide PWM output)

Analog Input Pins                                 : 6

DC Current per I/O Pin                         : 40 mA

DC Current for 3.3V Pin                       : 50 mA

Flash Memory                              : 32 KB (ATmega328) of which 0.5 KB used by  bootloader

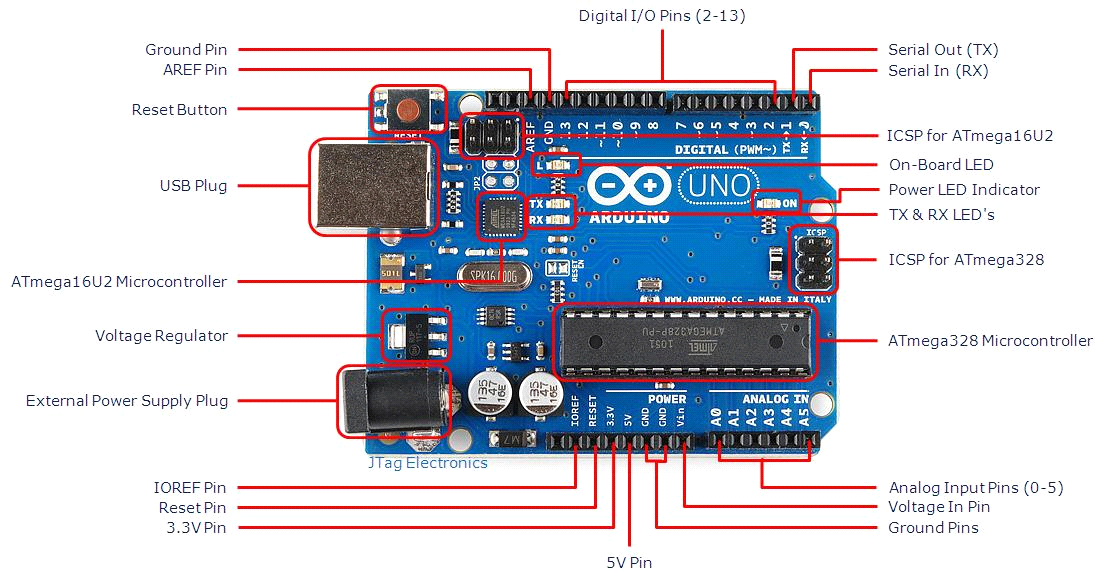
SRAM                                                  : 2 KB (ATmega328)

EEPROM                                             :  1 KB (ATmega328)

Clock Speed                                         : 16 MHz

Length                                                 :  68.6 mm

Width                                                    : 53.4 mm

fig 3.1 ArduinoUno R3 Board

* **USB Plug & External Power Supply Plug**

Every Arduino board needs a way to be connected to a power source. The Arduino Uno can be powered from a USB cable coming from your computer or a wall power supply that is terminated in a barrel jack. The power source is selected automatically. The USB connection is also how you will load code onto your Arduino board. Please on my other post on how to program with Arduino can be found in Installing and Programming Arduino.

NOTE: The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V,

however, the 5V pin may supply less than five volts and the board may be unstable.If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts

.

* **Voltage Regulator**

The voltage regulator is not actually something you can (or should) interact with on the Arduino. But it is potentially useful to know that it is there and what it’s for. The voltage regulator does exactly what it says – it controls the amount of voltage that is let into the Arduino board. Think of it as a kind of gatekeeper; it will turn away an extra voltage that might harm the circuit. Of course,it has its limits, so don’t hook up your Arduino to anything greater than 20 volts.

* **Power Pins**

Voltage In Pin – The input voltage to the Arduino board when it’s using an external power source(as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

5V Pin – This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 – 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. It’s not recommended.3.3V Pin – A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

* **Ground Pins**

There are several GND pins on the Arduino, any of which can be used to ground your circuit.

* **IOREF Pin**

This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs for working with the 5V or 3.3V.

* **Input and Output Pins**

Each of the 14 digital pins on the Uno can be used as an input or output. They operate at 5 volts. These pins can be used for both digital input (like telling if a button is pushed) and digital output (like powering an LED). Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-5k Ohms. In addition, some pins have specialized functions.

* **Serial Out (TX) & Serial In (RX)**

Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.

* **External Interrupts**

Pins 2 and 3 can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.

PWM– You may have noticed the tilde (~) next to some of the digital pins (3, 5, 6, 9, 10, and 11). These pins act as normal digital pins, but can also be used for something called Pulse-Width Modulation (PWM).  Think of these pins as being able to simulate analog output (like fading an LED in and out).

SPI – Pins 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). SPI stands for Serial Peripheral Interface. These pins support SPI communication using the SPI library.

Analog Input Pins – Labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). These pins can read the signal from an analog sensor (like a temperature sensor) and convert it into a digital value that we can read. By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF Pin(Stands for Analog Reference. Most of the time you can leave this pin alone). Additionally, some pins have specialized functionality:

TWI – Pins A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library.

* **Reset Pin**

Bring this line LOW to reset the microcontroller.Typically used to add a reset button to shields which block the one on the board.

* **LED Indicators**

Power LED Indicator – Just beneath and to the right of the word “UNO” on your circuit board, there’s a tiny LED next to the word ‘ON’. This LED should light up whenever you plug your Arduino into a power source. If this light doesn’t turn on, there’s a good chance something is wrong. Time to re-check your circuit!

On-Board LED – There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it’s off. This useful to quickly check if the board has no problem as some boards has a pre-loaded simple blinking LED program in it.

TX & RX LEDs – These LEDs will give us some nice visual indications whenever our Arduino is receiving or transmitting data (like when we’re loading a new program ontotheboard).

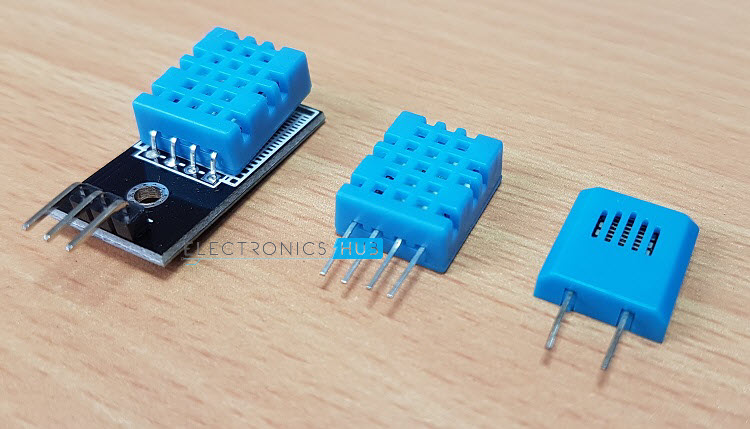
**Reset Button:**Pushing the reset button temporarily connect the reset pin to ground and restart any code that is loaded on the Arduino. This can be very useful if your code doesn’t repeat, but you want to test it multiple times.

**2. DHT11 (TEMPERATURE & HUNIDITY SENSOR):**

Humidity Sensor is one of the most important devices that has been widely in consumer, industrial, biomedical, and environmental etc. applications for measuring and monitoring Humidity.

Humidity is defined as the amount of water present in the surrounding air. This water content in the air is a key factor in the wellness of mankind. For example, we will feel comfortable even if the temperature is 00C with less humidity i.e. the air is dry.

But if the temperature is 100C and the humidity is high i.e. the water content of air is high, then we will feel quite uncomfortable. Humidity is also a major factor for operating sensitive equipment like electronics, industrial equipment, electrostatic sensitive devices and high voltage devices etc. Such sensitive equipment must be operated in a humidity environment that is suitable for the device.

[](https://www.electronicshub.org/wp-content/uploads/2017/06/Humidity-Sensors.jpg)

Hence, sensing, measuring, monitoring and controlling humidity is a very important task. Some of the important areas of application for sensing, measuring and controlling Humidity are mentioned below.

**Domestic**: Sensing and controlling humidity in our homes and offices is important as higher humidity conditions will affect the blood flow. Other areas include cooking, indoor plantation etc.

**Industrial**: In industries like refineries, chemical, metal, or other industries where furnaces are used, high humidity will reduce the amount of oxygen in the air and hence reduces the firing rate. Other industries like food processing, textile, paper etc. also need control of humidity.

**Agriculture**: Irrigation techniques like drip irrigation need accurate moisture content for plants. Also, the moisture in the soil plays an important role in the proper growth of the plant. Other areas where humidity control is required is indoor vegetation.

**Electronics and Semiconductor**: Almost all electronic devices are rated with a range of humidity values in which they work as expected. Generally, this value will be something like 10% – 50% Humidity. Semiconductor Fabs (Fabrication Plants) should maintain very precise temperature and humidity values as even minute difference can show a huge impact in the production.

**Medical**: Medical equipment like ventilators, incubators, sterilizers etc. need humidity control. It is also used in pharmaceutical plants and biological processes.

All the above mentioned and many other applications need sensing of Humidity and is done using Humidity Sensors. Before discussing about Humidity Sensors, its types and working principle, we will first see some important terms and definitions related to Humidity.

**Important Terms Related to Humidity**

**Moisture**: Generally, the term Moisture means water content of any material or substance. But practically, the term Moisture refers to the water content in solids and liquids. The term Humidity refers to the water content in gases (air).

Absolute Humidity: Absolute Humidity (AH) is the ratio of mass of the water vapour to the volume of the air. If m is the mass of the water vapour and V is the total volume i.e. volume of air and water vapour mixture, then Absolute Humidity AH is given by

**AH = m/V**

Absolute Humidity doesn’t take temperature in to account but it changes with temperature and pressure.  
Relative Humidity: Whenever we talk about measuring Humidity, it usually Relative Humidity that we are talking about (unless otherwise specified).

Relative Humidity or RH is the ratio of the actual water vapour pressure present in the air at a temperature to the maximum water vapour pressure present in the air at the same temperature.

In weather reports and forecasts, the probability of precipitation or dew or fog is indicated using Relative Humidity and hence, it is considered an important metric.

Relative Humidity takes both temperature and pressure in to consideration. Hence, the Humidity Sensors which measure Relative Humidity, measure both the moisture content as well as the air temperature.

**NOTE**: For temperatures above 1000C, measuring Relative Humidity (RH) is of no use as it would deliver misleading values.

Specific Humidity: Specific Humidity (SH) is the ratio of mass of the water vapor to the total mass of the air.  
Mixing Ratio or Humidity Ratio: Mixing Ratio is the ratio of mass of the water vapor to mass of the dry air.

Dew Point Temperature: Dew Point Temperature is the temperature at with the water vapor content is saturated in the air. At Dew Point temperature, the Relative Humidity RH is 100%. In other words, for the air to hold maximum amount of water vapor (or moisture), it has to reach Dew Point Temperature.

**Humidity Sensors – Classification and Working Principles**

Humidity Sensors are very important devices that help in measuring the environmental humidity. Technically, the device used to measure the humidity of the atmosphere is called Hygrometer. Humidity Sensors or Hygrometers can be classified based on the type of humidity it is used for measuring i.e.

Absolute Humidity (AH) sensors or Relative Humidity (RH) sensors. Humidity Sensors can also be classified based on the parameter used for measuring Humidity i.e. Capacitive Humidity Sensors, Electrical Conductivity (or Resistive) Humidity Sensors and Thermal Conductivity Humidity Sensors.

There are other types of Humidity Sensors or Hygrometers like Optical Hygrometer, Oscillating Hygrometer and Gravimetric Hygrometer.

Let us see about different types of Humidity Sensors or Hygrometers along with their working principles.

**Capacitive Humidity Sensors**

Humidity Sensors based on capacitive effect or simply Capacitive Humidity Sensors are one of the basic types of Humidity Sensors available.

They are often used in applications where factors like cost, rigidity and size are s of concern. In Capacitive Relative Humidity (RH) Sensors, the electrical permittivity of the dielectric material changes with change in humidity.

**3. PULSE SENSOR**

The Heartbeat rate information knowing is very useful while doing exercise, studying, etc. But, the heartbeat rate can be complicated to calculate. To overcome this problem, the pulse sensor or heartbeat sensor is used. This is a plug & play sensor mainly designed for Arduino board which can be used by makers, students, developers, artists who can utilize the heartbeat information into their projects. This sensor uses an easy optical pulse sensor along with amplification & cancellation of noise to make a circuit. By using this circuit, we can get fast and reliable heartbeat readings. This circuit can be operated with 4mA current and 5V voltage to use in mobile applications.

An alternate name of this sensor is heartbeat sensor or heart rate sensor. The working of this sensor can be done by connecting it from the fingertip or human ear to Arduino board. So that heart rate can be easily calculated.

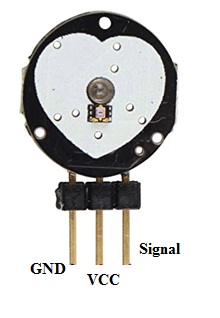


The pulse sensor includes a 24 inches color code cable, ear clip, Velcro Dots-2, transparent stickers-3, etc. A color code cable is connected to header connectors. So this sensor is easily connected to an Arduino into the project without soldering. An ear clip size is the same as a heart rate sensor and it can be connected using hot glue at the backside of the sensor to wear on the earlobe. Two Velcro dots are completely sized toward the sensor at the hook side. These are extremely useful while making a Velcro strap to cover approximately a fingertip. This is used to cover the Sensor around the finger. Transparent strikers are protection layers used to protect the sensor from sweaty earlobes and fingers. This sensor includes three holes in the region of the external edge so that one can easily connect anything to it.

The main specifications of this sensor mainly include the following. This is a hear beat detecting and biometric pulse rate sensor Its diameter is 0.625 Its thickness is 0.125 The operating voltage is ranges +5V otherwise +3.3V This is a plug and play type sensor The current utilization is 4mA Includes the circuits like Amplification & Noise cancellation This pulse sensor is not approved by the FDA or medical. So it is used in student-level projects, not for the commercial purpose in health issues applications.

**Pin Configuration**

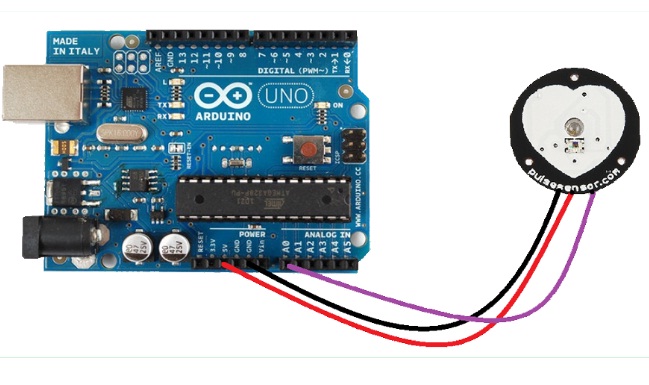
The heartbeat sensor includes three pins which discussed below.



Pin-1 (GND): Black Color Wire – It is connected to the GND terminal of the system. Pin-2 (VCC): Red Color Wire – It is connected to the supply voltage ( +5V otherwise +3.3V) of the system. Pin-3 (Signal): Purple Color Wire – It is connected to the pulsating o/p signal.

**Pulse Sensor Circuit Diagram**

The pulse sensor working principle is very simple. This sensor has two surfaces, on the first surface, the light-emitting diode & ambient light sensor is connected. Similarly, on the second surface, the circuit is connected which is accountable for the noise cancellation& amplification. The LED is located above a vein in a human body like ear tip or fingertip, however, it must be located on top of a layer directly. Once the LED is located on the vein, then the LED starts emitting light. Once the heart is pumping, then there will be a flow of blood within the veins. So if we check the blood flow, then we can check the heart rates also. If the blood flow is sensed then the ambient light sensor will receive more light as they will be reproduced by the flow of blood. This small change within obtained light can be examined over time to decide our pulse rates. This sensor used in straight forward, however connecting it in the correct way matters. Because all types of electronic components are directly exposed to the sensor. So, it is mandatory to envelop this sensor by using hot glue, vinyl strip otherwise other types of non-conductive materials. These sensors cannot be operated with wet hands. The sensor’s smooth side must be located on the pinnacle of the vein & press it. Generally, Velcro tapes or clips are utilized to get this force.



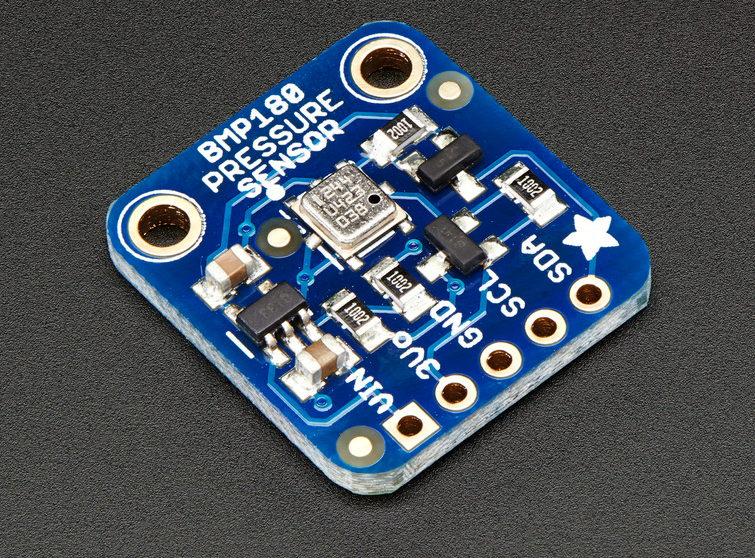
This sensor can be used by connecting it to the Arduino board. Once it is connected, then give the power supply with the help of VCC pin and GND pins. The operating voltage of this sensor is +5V or 3.3V. Once the sensor is connected to the development board such as Arduino, then we can use the readily accessible Arduino code to make things easier.

**Applications of Pulse Sensor**

The applications of pulse rate sensor include the following. This sensor is used for Sleep Tracking This sensor is used for Anxiety monitoring This sensor is used in remote patient monitoring or alarm system This sensor is used in Health bands This sensor is used in complex gaming consoles Thus, this is all about Pulse Sensor (Heartbeat / Heartrate Sensor). it is open-source and plug-and-play hardware. This sensor can easily include live heartbeat information into their projects. This sensor includes two circuits like an optical amplifying & a noise eliminating. The connection of this sensor on earlobe otherwise fingertip can be done using a Clip, and connect it to Arduino board. So that heart rate can be easily measured. These sensors are used by developers, students, makers, athletes, artists, etc.

**4. BMP 180 PRESSURE**

The precision sensor from Bosch, the BMP180, stands out as an exceptional and cost-effective solution for measuring both barometric pressure and temperature. Its versatility extends to functioning as an altimeter, leveraging the correlation between pressure changes and altitude. The integration of this sensor onto a printed circuit board (PCB) with essential features makes it user-friendly for a variety of applications.



This sensor, succeeding the BMP085, maintains complete compatibility with the BMP085 in terms of firmware and software. This means that users can seamlessly transition from BMP085-based projects to BMP180 without having to modify existing code or libraries. The absence of the XCLR pin in the BMP180 is notable; hence, users wanting to determine data readiness must rely on querying the I2C bus.

The board's 5V compliance is a noteworthy feature, ensuring compatibility with systems operating at this voltage. To facilitate this, a 3.3V regulator and an I2C level shifter circuit are incorporated into the board. These components enable safe utilization of the sensor with 5V logic and power, expanding its compatibility and ease of integration into diverse projects.

The ease of use with this sensor is exemplified in its connectivity, particularly with popular development platforms like Arduino. For Arduino users, connecting the VIN pin to the 5V voltage pin, GND to ground, SCL to I2C Clock (Analog 5), and SDA to I2C Data (Analog 4) is all that's required. To get started, downloading the BMP085/BMP180 Arduino library and accompanying example code for temperature, pressure, and altitude calculations is recommended. Once the library is installed, loading the example sketch immediately provides users with precise temperature, pressure, and altitude data.

In summary, Bosch's BMP180 sensor emerges as an optimal choice for those seeking a reliable, low-cost solution for measuring barometric pressure and temperature, with the added bonus of functioning as an altimeter. Its seamless compatibility with the BMP085, 5V compliance, and straightforward integration with popular development platforms make it an excellent choice for a wide range of applications, from weather monitoring to altitude tracking in various projects.

**5. LCD**

LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over seven segments and other multi segment LEDs. The reasons being: LCDs are economical; easily programmable; have no limitation of displaying special & even custom characters (unlike in seven segments), animationsand so on.

**3.6.1 Introduction of LCD**

The most commonly used Character based LCDs are based on Hitachi's HD44780 controller or other which are compatible with HD44580.

A **16x2 LCD** means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data.

The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD.

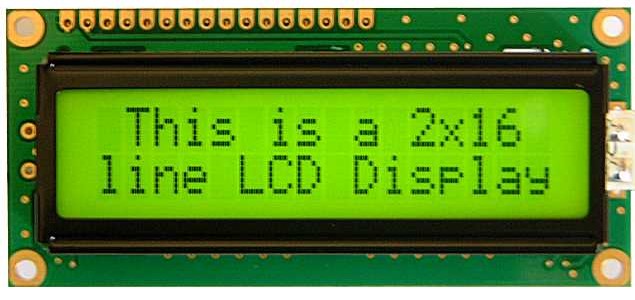


Fig3.17: 16x2 LCD display

**3.6.2Pin Description of LCD**

Most LCDs with 1 controller has 14 Pins and LCDs with 2 controller has 16 Pins (two pins are extra in both for back-light LED connections). Pin description is shown in the table below.

**Pin Configuration table for a 16X2 LCD character display:-**

|  |  |  |
| --- | --- | --- |
| **Pin Number** | **Symbol** | **Function** |
| **1** | Vss | Ground Terminal |
| **2** | Vcc | Positive Supply |
| **3** | Vdd | Contrast adjustment |
| **4** | RS | Register Select; 0→Instruction Register, 1→Data Register |
| **5** | R/W | Read/write Signal; 1→Read, 0→ Write |
| **6** | E | Enable; Falling edge |
| **7** | DB0 | Bi-directional data bus, data transfer is performed once, thru DB0 to DB7, in the case of interface data length is 8-bits; and twice, through DB4 to DB7 in the case of interface data length is 4-bits. Upper four bits first then lower four bits. |
| **8** | DB1 |
| **9** | DB2 |
| **10** | DB3 |
| **11** | DB4 |
| **12** | DB5 |
| **13** | DB6 |
| **14** | DB7 |
| **15** | LED-(K) | Back light LED cathode terminal |
| **16** | LED+(A) | Back Light LED anode terminal |

Table3.1: Pin Description of LCD

**3.6.3 Data/Signals/Execution of LCD**

LCD accepts two types of signals, one is data, and another is control. These signals are recognized by the LCD module from status of the RS pin. Now data can be read also from the LCD display, by pulling the R/W pin high. As soon as the E pin is pulsed, LCD display reads data at the falling edge of the pulse and executes it, same for the case of transmission.

LCD display takes a time of 39-43µS to place a character or execute a command. Except for clearing display and to seek cursor to home position it takes 1.53ms to 1.64ms. Any attempt to send any data before this interval may lead to failure to read data or execution of the current data in some devices. Some devices compensate the speed by storing the incoming data to some temporary registers.

**3.6.4 Commands and Instruction set**

Only the instruction register (IR) and the data register (DR) of the LCD can be controlled by the MCU. Before starting the internal operation of the LCD, control information is temporarily stored into these registers to allow interfacing with various MCUs, which operate at different speeds, or various peripheral control devices. The internal operation of the LCD is determined by signals sent from the MCU. These signals, which include register selection signal (RS), read/write signal (R/W), and the data bus (DB0 to DB7), make up the LCD instructions.

There are four categories of instructions that:

* Designate LCD functions, such as display format, data length, etc.
* Set internal RAM addresses
* Perform data transfer with internal RAM
* Perform miscellaneous functions

Although looking at the table you can make your own commands and test them. Below is a brief list of useful commands which are used frequently while working on the LCD.

**6. WIFI MODULE**

The **ESP8266** is a low-cost [Wi-Fi](https://en.wikipedia.org/wiki/Wi-Fi) chip with full TCP/IP stack and [MCU (microcontroller unit)](https://en.wikipedia.org/wiki/Microcontroller) capability produced by Shanghai-based Chinese manufacturer, [Espressif Systems](https://en.wikipedia.org/w/index.php?title=Espressif_Systems&action=edit&redlink=1" \o "Espressif Systems (page does not exist)).[[1]](https://en.wikipedia.org/wiki/ESP8266#cite_note-1)

The chip first came to the attention of western [makers](https://en.wikipedia.org/wiki/Maker_culture) in August 2014 with the ESP-01 module, made by a third-party manufacturer, AI-Thinker. This small module allows microcontrollers to connect to a Wi-Fi network and make simple TCP/IP connections using [Hayes](https://en.wikipedia.org/wiki/Hayes_command_set)-style commands. However, at the time there was almost no English-language documentation on the chip and the commands it accepted.[[2]](https://en.wikipedia.org/wiki/ESP8266#cite_note-2) The very low price and the fact that there were very few external components on the module which suggested that it could eventually be very inexpensive in volume, attracted many hackers to explore the module, chip, and the software on it, as well as to translate the Chinese documentation.[[3]](https://en.wikipedia.org/wiki/ESP8266#cite_note-3)

The **ESP8285** is an ESP8266 with 1 MiB of built-in flash, allowing for single-chip devices capable of connecting to Wi-Fi

**Features**

* 32-bit RISC CPU: TensilicaXtensa L106 running at 80 MHz\*
* 64 KiB of instruction RAM, 96 KiB of data RAM
* External QSPI flash: 512 KiB to 4 MiB\* (up to 16 MiB is supported)
* IEEE 802.11 b/g/n Wi-Fi
* Integrated TR switch, balun, LNA, power amplifier and matching network
* WEP or WPA/WPA2 authentication, or open networks
* 16 GPIO pins
* SPI
* I²C
* I²S interfaces with DMA (sharing pins with GPIO)
* UART on dedicated pins, plus a transmit-only UART can be enabled on GPIO2
* 10-bit ADC



**7. REGULATED POWER SUPPLY:**

All digital circuits require regulated power supply. In this article we are going to learn how to get a regulated positive supply from the mains supply.

**STEP DOWN**

**TRANSFORMER**

**BRIDGE**

**RECTIFIER**

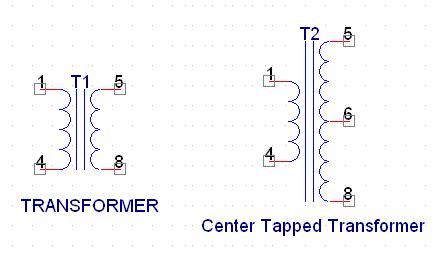
**FILTER**

**CIRCUIT**

**REGULATOR SECTION**

**shows the basic block diagram of a fixed regulated power supply**

**TRANSFORMER**



**Types of transformer**

A transformer consists of two coils also called as “WINDINGS” namely PRIMARY & SECONDARY.

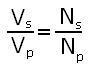
They are linked together through inductively coupled electrical conductors also called as CORE. A changing current in the primary causes a change in the Magnetic Field in the core & this in turn induces an alternating voltage in the secondary coil. If load is applied to the secondary then an alternating current will flow through the load. If we consider an ideal condition then all the energy from the primary circuit will be transferred to the secondary circuit through the magnetic field.

So Image

Image

The secondary voltage of the transformer depends on the number of turns

in the Primary as well as in the secondary.

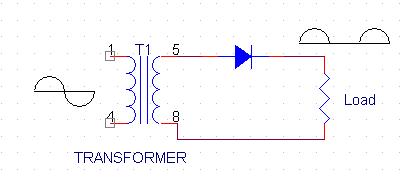


**Rectifier**

A rectifier is a device that converts an AC signal into DC signal. For rectification purpose we use a diode, a diode is a device that allows current to pass only in one direction i.e. when the anode of the diode is positive with respect to the cathode also called as forward biased condition & blocks current in the reversed biased condition.

 Rectifier can be classified as follows:

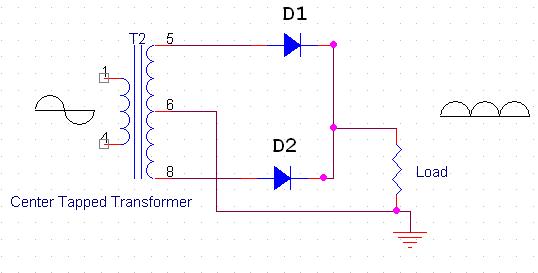
**1)      Half Wave rectifier.**



**Half wave rectifier**

This is the simplest type of rectifier as you can see in the diagram a half wave rectifier consists of only one diode. When an AC signal is applied to it during the positive half cycle the diode is forward biased & current flows through it. But during the negative half cycle diode is reverse biased & no current flows through it. Since only one half of the input reaches the output, it is very inefficient to be used in power supplies.

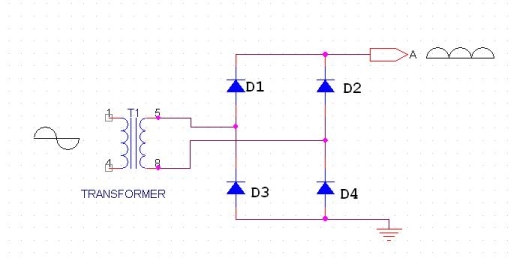
**2)      Full wave rectifier.**



**Full wave rectifier**

Half wave rectifier is quite simple but it is very inefficient, for greater efficiency we would like to use both the half cycles of the AC signal. This can be achieved by using a center tapped transformer i.e. we would have to double the size of secondary winding & provide connection to the center. So during the positive half cycle diode D1 conducts & D2 is in reverse biased condition. During the negative half cycle diode D2 conducts & D1 is reverse biased. Thus we get both the half cycles across the load.

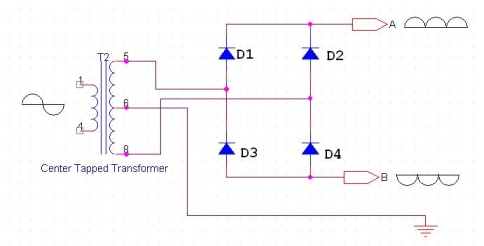
One of the disadvantages of Full Wave Rectifier design is the necessity of using a center tapped transformer, thus increasing the size & cost of the circuit. This can be avoided by using the Full Wave Bridge Rectifier.

**3) Bridge Rectifier.**

**Bridge rectifier**

As the name suggests it converts the full wave i.e. both the positive & the negative half cycle into DC thus it is much more efficient than Half Wave Rectifier & that too without using a center tapped transformer thus much more cost effective than Full Wave Rectifier.

Full Bridge Wave Rectifier consists of four diodes namely D1, D2, D3 and D4. During the positive half cycle diodes D1 & D4 conduct whereas in the negative half cycle diodes D2 & D3 conduct thus the diodes keep switching the transformer connections so we get positive half cycles in the output.



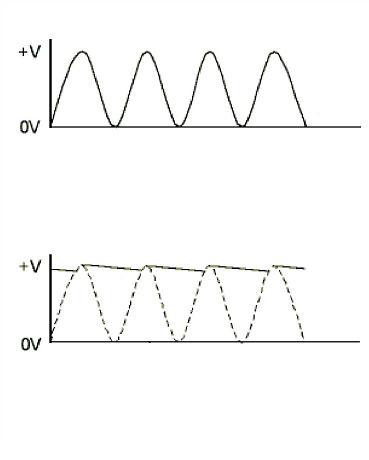
**Bridge rectifier with center tapped**

If we use a center tapped transformer for a bridge rectifier we can get both positive & negative half cycles which can thus be used for generating fixed positive & fixed negative voltages.

**FILTER CAPACITOR**

Even though half wave & full wave rectifier give DC output, none of them provides a constant output voltage. For this we require to smoothen the waveform received from the rectifier. This can be done by using a capacitor at the output of the rectifier this capacitor is also called as “FILTER CAPACITOR” or “SMOOTHING CAPACITOR” or “RESERVOIR CAPACITOR”. Even after using this capacitor a small amount of ripple will remain.

We place the Filter Capacitor at the output of the rectifier the capacitor will charge to the peak voltage during each half cycle then will discharge its stored energy slowly through the load while the rectified voltage drops to zero, thus trying to keep the voltage as constant as possible.



**output waveforms**

If we go on increasing the value of the filter capacitor then the Ripple will decrease. But then the costing will increase. The value of the Filter capacitor depends on the current consumed by the circuit, the frequency of the waveform & the accepted ripple.



Where,

Vr= accepted ripple voltage.( should not be more than 10% of  the voltage)

I= current consumed by the circuit in Amperes.

F= frequency of the waveform. A half wave rectifier has only one peak in one cycle so F=25hz

Whereas a full wave rectifier has Two peaks in one cycle so F=100hz.

**VOLTAGE REGULATOR**

A Voltage regulator is a device which converts varying input voltage into a constant regulated output voltage. Voltage regulator can be of two types

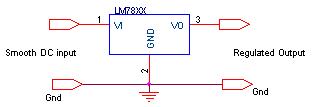
1)      Linear Voltage Regulator

      Also called as Resistive Voltage regulator because they dissipate the excessive voltage resistively as heat.

2)      Switching Regulators.

They regulate the output voltage by switching the Current ON/OFF very rapidly. Since their output is either ON or OFF it dissipates very low power thus achieving higher efficiency as compared to linear voltage regulators. But they are more complex & generate high noise due to their switching action. For low level of output power switching regulators tend to be costly but for higher output wattage they are much cheaper than linear regulators.

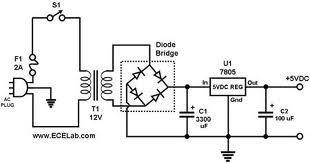
The most commonly available Linear Positive Voltage Regulators are the 78XX series where the XX indicates the output voltage. And 79XX series is for Negative Voltage Regulators.



**Switching regulator**

After filtering the rectifier output the signal is given to a voltage regulator. The maximum input voltage that can be applied at the input is 35V.Normally there is a 2-3 Volts drop across the regulator so the input voltage should be at least 2-3 Volts higher than the output voltage. If the input voltage gets below the Vmin of the regulator due to the ripple voltage or due to any other reason the voltage regulator will not be able to produce the correct regulated voltage.

#### 3 Circuit diagram:



**Circuit Diagram of power supply**

#### IC 7805:

7805 is an integrated three-terminal positive fixed linear voltage regulator. It supports an input voltage of 10 volts to 35 volts and output voltage of 5 volts. It has a current rating of 1 amp although lower current models are available. Its output voltage is fixed at 5.0V. The 7805 also has a built-in current limiter as a safety feature. 7805 is manufactured by many companies, including National Semiconductors and Fairchild Semiconductors.

The 7805 will automatically reduce output current if it gets too hot. The last two digits represent the voltage; for instance, the 7812 is a 12-volt regulator. The 78xx series of regulators is designed to work in complement with the 79xx series of negative voltage regulators in systems that provide both positive and negative regulated voltages, since the 78xx series can't regulate negative voltages in such a system.

The 7805 & 78 is one of the most common and well-known of the 78xx series regulators, as it's small component count and medium-power regulated 5V make it useful for powering TTL devices.

**Table Specifications of IC7805**

|  |  |
| --- | --- |
| **SPECIFICATIONS** | **IC 7805** |
| Vout | 5V |
| Vein - Vout Difference | 5V - 20V |
| Operation Ambient Temp | 0 - 125°C |
| Output Imax | 1A |

**CHAPTER 6**

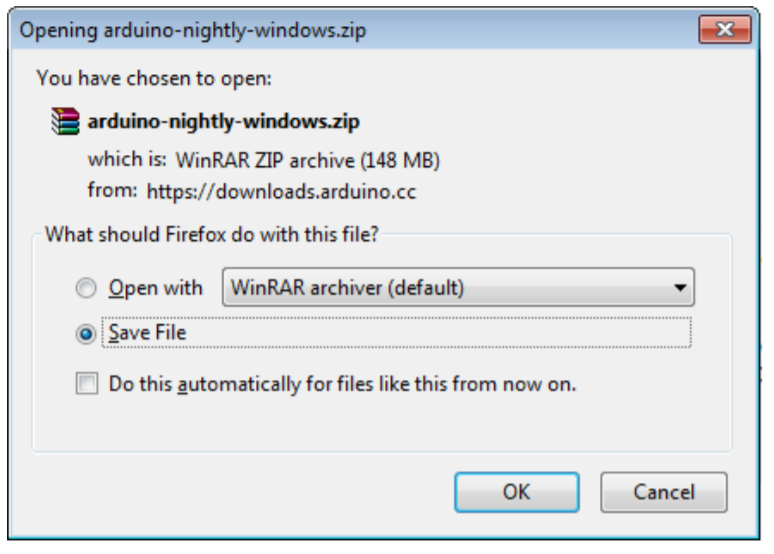
**SOFTWARE TOOLS**

1. Arduino ide
2. Embedded c programming
3. Windows OS

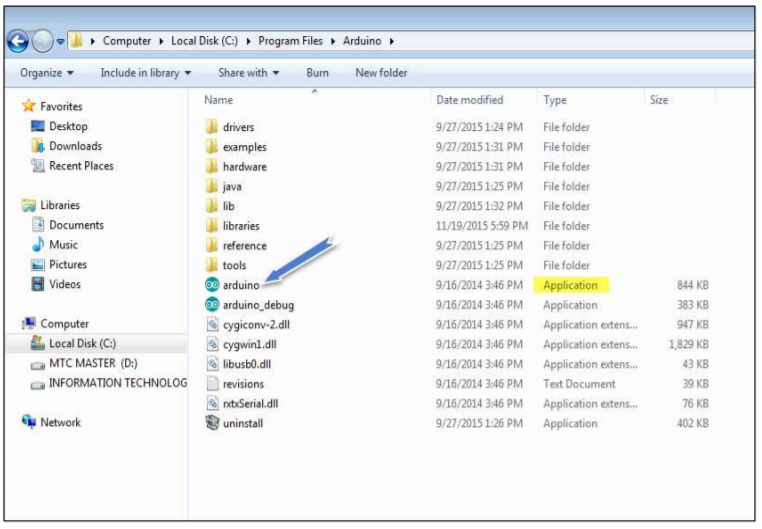
**6.1 DESCRIPTION OF TOOLS**

**1. Arduino IDE:**

Arduino IDE Software. You can get different versions of Arduino IDE from the Download page on the Arduino Official website. You must select your software, which is compatible with your operating system (Windows, IOS, or Linux). After your file download is complete, unzip the file.



Launch Arduino IDE. After your Arduino IDE software is downloaded, you need to unzip the folder. Inside the folder, you can find the application icon with an infinity label (application.exe). Doubleclick the icon to start the IDE.

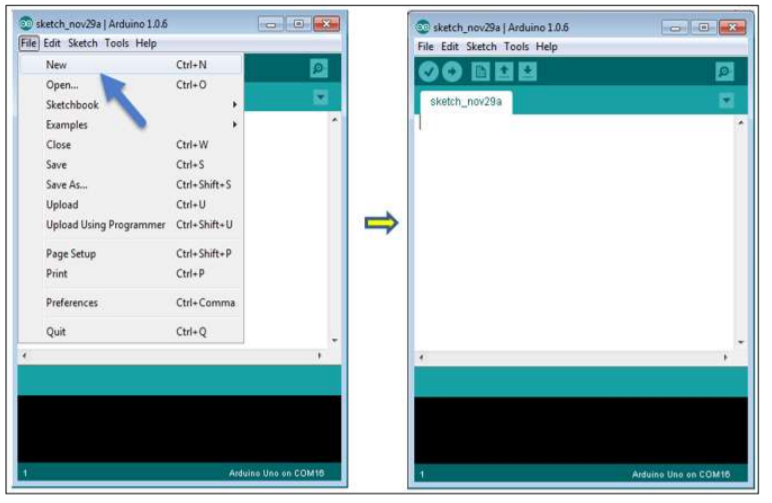


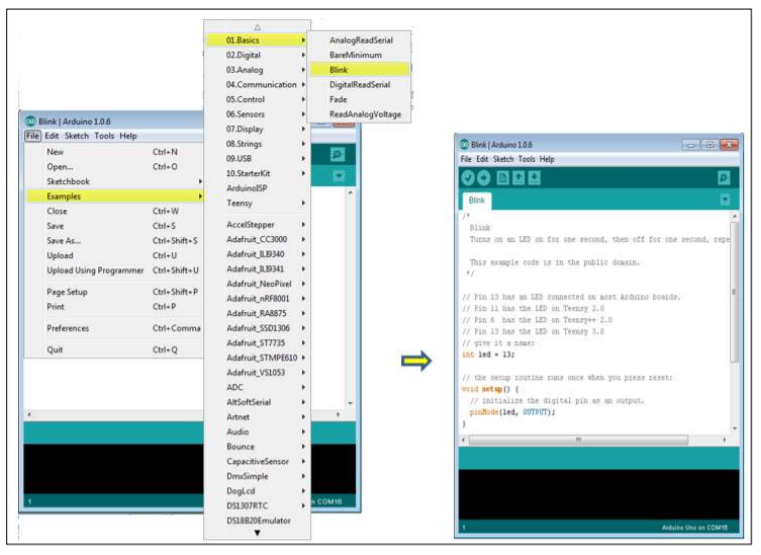
Open your first project. Once the software starts, you have two options:

• Create a new project.

• Open an existing project example.

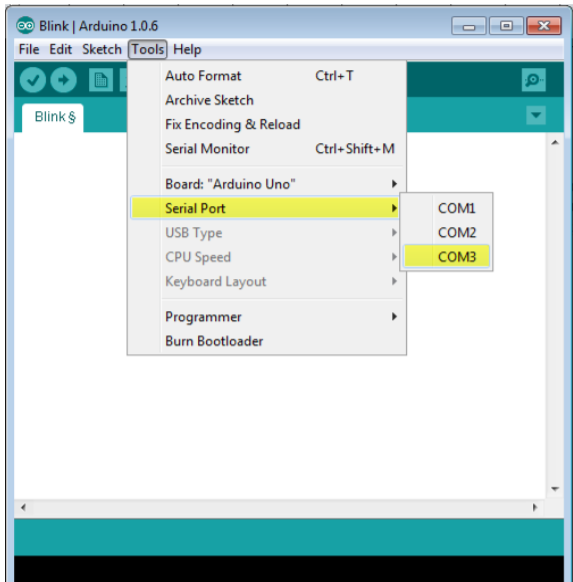
To create a new project, select File --> New





Here, we are selecting just one of the examples with the name Blink. It turns the LED on and off with some time delay. You can select any other example from the list

Select your serial port. Select the serial device of the Arduino board. Go to Tools -> Serial Port menu. This is likely to be COM3 or higher (COM1 and COM2 are usually reserved for hardware serial ports). To find out, you can disconnect your Arduino board and re-open the menu, the entry that disappears should be of the Arduino board. Reconnect the board and select that serial port.

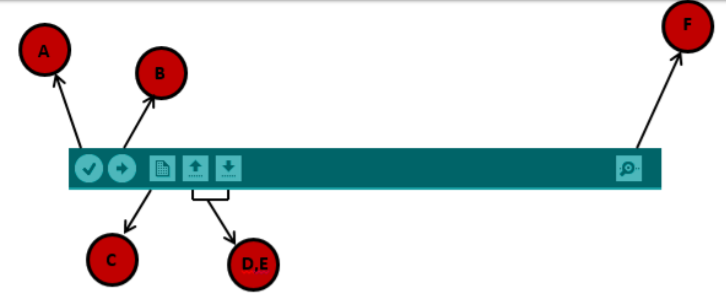


Before explaining how we can upload our program to the board, we must demonstrate the function of each symbol appearing in the Arduino IDE toolbar.

1. Used to check if there is any compilation error.
2. Used to upload a program to the Arduino board.
3. Shortcut used to create a new sketch.
4. Used to directly open one of the example sketch.
5. Used to save your sketch.
6. Serial monitor used to receive serial data from the board and send the serial data to the board.

Now, simply click the "Upload" button in the environment.

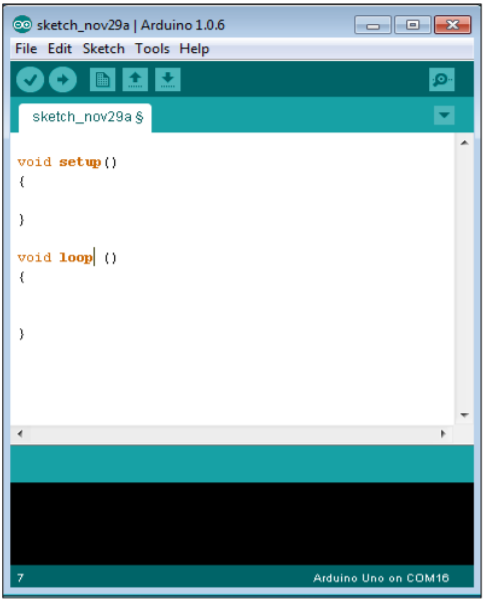
Wait a few seconds; you will see the RX and TX LEDs on the board, flashing. If the upload is successful, the message "Done uploading" will appear in the status bar.



In this chapter, we will study in depth, the Arduino program structure and we will learn more new terminologies used in the Arduino world. The Arduino software is open-source. The source code for the Java environment is released under the GPL and the C/C++ microcontroller libraries are under the LGPL. Sketch: The first new terminology is the Arduino program called “sketch”. Structure Arduino programs can be divided in three main parts: Structure, Values (variables and constants), and Functions. In this tutorial, we will learn about the Arduino software program, step by step, and how we can write the program without any syntax or compilation error. Let us start with the Structure. Software structure consist of two main functions:

•Setup( ) function

•Loop( ) function



Data types in C refers to an extensive system used for declaring variables or functions of different types. The type of a variable determines how much space it occupies in the storage and how the bit pattern stored is interpreted. The following table provides all the data types that you will use during Arduino programming.

**CHAPTER 7**

**RESULTS AND DISCUSSION**

**CHAPTER 8**

**CONCLUSION**

In conclusion, the integration of IoT technology in healthcare, as exemplified by the IoT-based ICU Patient Monitoring System, presents a significant advancement in patient care and monitoring. By leveraging Arduino microcontrollers and the cloud-based platform ThingSpeak, healthcare professionals can access real-time data on critical health parameters, enabling timely intervention and personalized patient care. The system's ability to visualize and analyze data, coupled with customizable alerts, enhances medical decision-making and improves patient outcomes. Moreover, its user-friendly interface facilitates remote monitoring, thereby increasing the efficiency and accessibility of healthcare delivery. As IoT continues to evolve, such innovative solutions hold promise for revolutionizing healthcare practices, promoting proactive patient management, and ultimately, saving lives.

**FUTURE SCOPE:**

Looking ahead, the future scope of IoT-based patient monitoring systems is promising. Advancements in sensor technology, data analytics, and artificial intelligence will enable more sophisticated and personalized healthcare solutions. Integration with wearable devices and predictive analytics could facilitate early detection of health issues and proactive interventions. Additionally, interoperability standards and cybersecurity measures will be crucial for seamless data exchange and ensuring patient privacy and security. Overall, IoT holds immense potential for transforming healthcare delivery and improving patient outcomes in the future.

**REFERENCES**

[1] H. R. Patil and D. S. Garge, "Patient Monitoring System", International journal of advance research in science and engineering, vol. 7, no. 03, pp. 23-32, 2018. [Accessed 6 March 2019].

[2] Sardey, P., Pratik B., D., Shalvari R., D. and Shubham, G.."IOT BASED PATIENT HEALTH MONITORING SYSTEM", International Journal of Advance Engineering and Research Development, vol. 4, no. 06, 2017. Available: 10.21090/ijaerd.46601.

[3] S. rani and B. Hari, "Iot Patient Health Monitoring System", Indian Journal of Public Health Research and Development, vol. 8, no. 4, 2017. [Accessed 11 August 2019].

[4] Nevon Projects. 2020. IOT Based Heart Defect Monitoring System Using ECG. [online] Available at: [Accessed 10 November 2019].

[5] "IOT & Arduino based patient health monitoring system project", Microtronics Technologies, 2018. [Online]. Available: https://www.projectsof8051.com/arduino-and-iot-based-patient-healthmonitoring-system-project/. [Accessed: 18- Jun- 2019].

[6] R.Jain, https://circuitdigest.com/, 2018. [Online]. Available: https://circuitdigest.com/microcontroller-projects/iot-based-patientmonitoring-system-using-esp8266-and-arduino. [Accessed: 18- Dec- 2019].

[7] M. Sathya, S. Madhan and K. Jayanthi, "Internet of things (IoT) based health monitoring system and challenges", International Journal of Engineering & Technology, vol. 7, no. 17, p. 175, 2018. Available: 10.14419/ijet.v7i1.7.10645 [Accessed 10 June 2019].

[8] M. Muqeet and M. Quadri, "IoT based Patient Monitoring System Using Raspberry Pi", IJRECE VOL. 7 ISSUE 2 (APRIL- JUNE 2019) ISSN: 2393- 9028 (PRINT) | ISSN: 2348-2281 (ONLINE) INTERNATIONAL JOURNAL OF RESEARCH IN ELECTRONICS AND COMPUTER ENGINEERING, vol. 7, no. 2, pp. 2976-2978, 2019.

[9] DHT11 Sensor Pinout, Features, Equivalents & Datasheet. (2018). Retrieved 9 October 2019, from https://components101.com/dht11- temperature-sensor.

[10] "DHT11 Sensor Definition, Working and Applications", ElProCus - Electronic Projects for Engineering Students. [Online]. Available: <https://www.elprocus.com/a-brief-on-dht11-sensor/>.

[11] A. Azizulkarim, M. Abdul Jamil and R. Ambar, "Design and Development of Patient Monitoring System", IOP Conference Series: Materials Science and Engineering, vol. 226, p. 012094, 2017. Available: 10.1088/1757- 899x/226/1/012094.

[12] "Pulse Sensor Pinout, Configuration & How Pulse Sensor Works", Components101.com,2018.[Online].Available:https://components1 01.com/sensors/pulse-sensor. [Accessed: 13- Apr- 2019].

[13] M. Rashidul, S. Khan and M. Osmani, "IoT-based Low Cost Healthcare System for Cardiac patient of Bangladesh using Bio-sensors", 2019.

[14] "ADS1115 analog-to-digital converter and ESP8266-esp8266 learning", esp8266learning.[Online].Available:http://www.esp8266learning. com/ads1115-analog-to-digital-converter-and-esp8266.php.[Accessed: 08- Apr- 2019].

[15] Priyanka, R., and M. Reji. 2019. "IOT Based Health Monitoring System Using Blynk App". International Journal Of Engineering And Advanced Technology 8 (6): 78-81. doi:10.35940/ijeat.e7467.088619.

[16] T. Shaown, I. Hasan, M. M. R. Mim and M. S. Hossain, "IoT-based Portable ECG Monitoring System for Smart Healthcare," 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), Dhaka, Bangladesh, 2019, pp. 1-5, doi: 10.1109/ICASERT.2019.8934622.

[17] Dr. B.Srikanth, P.Divya, P.Nandini, Sk.Sabira and T.Bharathi., "PATIENT HEALTH MONITORING USING ARDUINO THROUGH IOT", EPRA International Journal of Research & Development (IJRD), pp. 619-633, 2020. Available: 10.36713/epra4554.

[18] P.Kalaivani,P.Priyanka,S.L.Priyanka, ,M.V.Devi, Patient Monitoring System Using Wi-Fi Technology", International Journal of Recent Trends in Engineering and Research, pp. 23-30, 2018. Available: 10.23883/ijrter.conf.02180328.004.k1mgc.