**CHAPTER -1**

**INTRODUCTION**

Pollution is an important and major factor that is negatively affecting the quality of the lives of millions. Most of the pollutants in the environment are a result of untreated emissions and the release of industrial wastes from factories to the environment, with the growing world population and industry advancement, Environmental pollution became a big concern. Many Processing and manufacturing industries majorly contribute to 4 types of pollution

* Air pollution.
* Water pollution.
* Soil pollution.
* Noise pollution

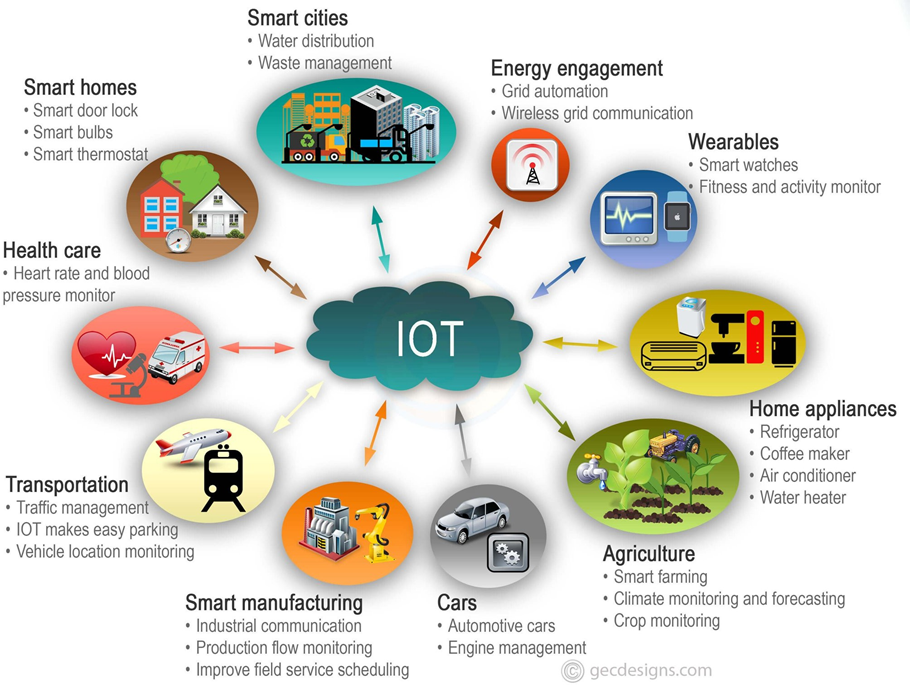
Air pollution is caused by the emission of a high amount of untreated industrial waste such as carbon dioxide, carbon monoxide, sulphurous oxides, nitrous oxides, minute particulate matters like optical dust, and also a considerable amount of vapours of polymers such as methane’s and butanes which are formed as a by-product due to burning coal, burning natural oils, fossil fuels and petroleum products to perform industrial processing activities. As per estimation made by the Indian pollution control authority that every year nearly a 1.2million Indians die because of airborne diseases Similarly, the Major reason for water pollution through industries is caused by discharging the untreated industrial waste that is generated during various processing activities, the industrial wastes include asbestos, lead, mercury, nitrates, phosphates, sulphur, and other petrochemicals, etc. The untreated acidic effluents which are released to natural reservoirs decrease the pH value of the reservoir water resulting in a decreased microbial activity affecting the growth of the algae and other aquatic plants resulting in a decrease in the level of BOD (biological oxygen demand) that is the level of dissolved oxygen in the water which affects in the breathing of the aquatic animals resulting in their death. Likewise, Thermal pollution can be defined as the sudden increase or decrease in the temperature of a natural water reservoir, which may be oceans, ponds, rivers, lakes, etc. due to human activities. Thermal pollution usually occurs when the processing plants take the water from the natural reservoir and let out the used water back to the reservoir with extreme differences in temperatures. The sharp increase in the temperature of the water reservoir causes a decrease in the level of dissolved oxygen in the water affecting the life of aquatic plants and animals.

**1.1: IoT**

Internet of things (IoT) describes physical objects (or groups of such objects) with sensors, processing ability, software and other technologies that connect and exchange data with other devices and systems over the Internet or other communications networks.

The Internet of things (IoT) falls under the Electronics & Communication and Computer Science Engineering. Internet of things has been considered a misnomer because devices do not need to be connected to the public internet, they The only need to be connected to a network, and be individually addressable.[7][8]

**Figure 1.1: A Modern Example of IOT**



**1.1.1: HISTORY**

Such as Electronic Route Guidance System (ERGS) and CACS in the United States and Japan respectively. While the term Inter-Vehicle Communications (IVC) began to circulate in the early 1980s. Various media were used before the standardization activities began, such as lasers, infrared, and The beginnings of vehicular communications go back to the 1970s. Work began on projects radio waves.

The PATH project in the United States between 1986 and 1997 was an important breakthrough in vehicular communications projects. Projects related to vehicular communications in Europe were launched with the PROMETHEUS project between 1986 and 1995.Numerous subsequent projects have been implemented all over the world such as the Advanced Safety Vehicle (ASV) program, CHAUFFEUR I and II, Fleet Net, Car TALK 2000,etc.

In the early 2000s, the term Vehicular Ad Hoc Network (VANET) was introduced as an application of the principles of Mobile Ad-Hoc Networks (MANETs) to the vehicular field. The terms VANET and IVC do not differ and are used interchangeably to refer to communications between vehicles with or without reliance on roadside infrastructure, although some have argued that IVC refers to direct V2V connections Only. Many projects have appeared in EU, Japan, USA and other parts of the world for example, ETC, SAFESPOT, PREVENT, COME Safety, NOW, IVI.

**1.1: APPLICATIONS OF IOT**

* The number of connected devices now dwarfs the number of humans on earth.
* Researchers at Frost & Sullivan put the number of active IoT-connected devices at 41.76 billion in 2023. IoT Analytics researchers estimate the number at 16.7 billion active endpoints in 2023, while Statist a estimates 15.14 billion
* Despite variations in the actual figures and what's included in the count, one thing is clear: There's a mind-blowing number of IoT devices in the world.
* That might not be surprising, though, considering the multiple areas where IoT is being used. Those IoT connections span the globe and permeate nearly all places: homes, offices, factories, farms, vehicles and even space.

**Here's a detailed look at the top 5 use cases of IoT.**

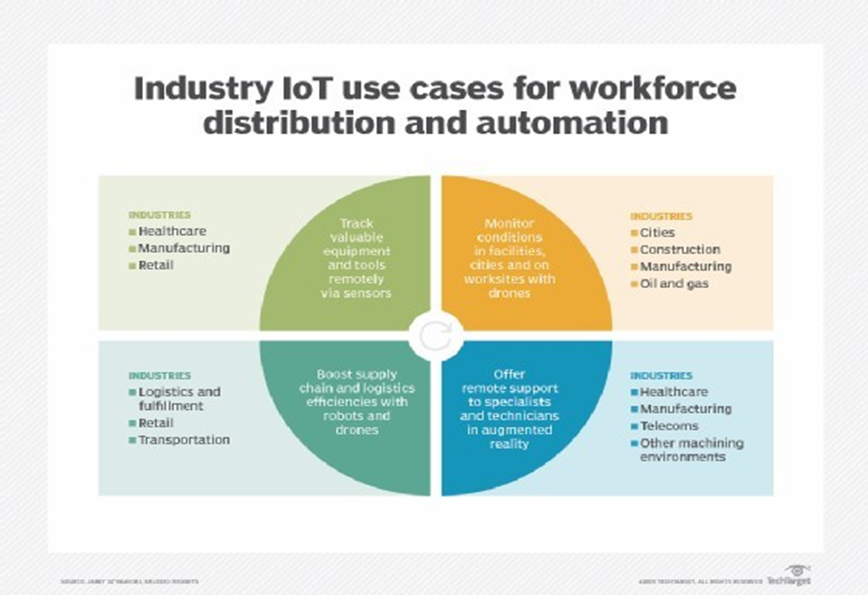
**1. Self-driving and connected vehicles**

Autonomous vehicles are one of the most notable examples of IoT in action, with long time automotive companies such as BMW Group, Ford Motor Company and General Motors along with newer entries such as Tesla, all working on self-driving vehicles.

**2. Logistics and fleet management**

Companies are using sensors, telematics, GPS and analytics to see where their vehicles are at any given moment, estimate when they'll arrive at their destination and whether external conditions warrant updating routes or expected arrival times.

This technology ecosystem also enables companies to identify ways to improve operations through predictive maintenance, more driver training and route optimization



**Figure 1.1: Cases for workforce**

**3. Traffic management**

Part of what enables self-driving cars is smart traffic management, which is also powered by IoT.

Like the vehicles themselves, roadway infrastructure has become more connected during the past decade, with cameras, sensors, traffic light controls, parking meters and even smartphone traffic apps transmitting data that's used to help avert traffic jams, prevent accidents and ensure smooth travel.

Sensors on traffic signals can detect varying levels of light in the sky and adjust the brightness of the signals, helping ensure they're always visible to drivers.

Connected devices can be used to detect open parking spaces and transmit that information to kiosks or apps to alert drivers.

**4. Smart grids, including smart meters**

Utilities are also using IoT to bring efficiency and resiliency to their energy grids.

Historically, energy flowed one way along the grid: from the generation site to the customer. However, connected devices now enable two-way communication along the entire energy supply chain: from generation through distribution to use, thereby improving the utilities' ability to move and manage it.

Utilities can analyse real-time data transmitted by connected devices to detect blackouts and redirect distribution and respond to changes in energy demand and load.

**5. Environmental monitoring**

Connected devices can collect data that indicates the health and quality of air, water and soil, as well as fisheries, forests and other natural habitats. They can also collect weather and other environmental data.

As such, IoT delivers the ability to not only access more real-time data about the environment at any given time and place, but it also enables a range of organizations in various industries to use that data to glean actionable insights.

**1.2: TOOLS**

Keeping track of all the IoT devices on a network, ensuring they are secure and operating correctly are critical -- and challenging -- activities. IoT monitoring tools simplify keeping devices updated with the latest software and firmware, automatically provide status information and identification, and are part of a larger cybersecurity strategy.

IT administrators can deploy and use IoT monitoring tools on their own or in conjunction with IoT analytics applications, security software, and identity and access management platforms.

**Features to look for in IoT monitoring software remote monitoring**

**Remote monitoring**

of IoT devices is a must; devices are often deployed in inaccessible or hard-to-reach locations. It's not practical or economical to send IT teams on location to physically inspect or monitor the devices. It's also not a good use of time to have them switch between several monitoring applications to check in on the various devices. An IoT monitoring application with remote monitoring capabilities eliminates the extra work and centralizes the information IT teams need to do their work.

**Automated discovery and alerting**

Automation for IoT device monitoring doesn't need to be sophisticated. Setting baseline performance limits that trigger alarms when an IoT device crosses the threshold is enough. This level of automation reduces the workload for IT teams by engaging them only when there is a true issue that requires their attention. Most monitoring software comes with default performance thresholds that IT can use as they are or tweak them to meet their environment's needs.

**Connection security**

Whether they're connected through a private network or the internet, IoT devices can be vulnerable to cyber-attacks. Hackers can take advantage of default device passwords to enter a network and then move about at will. Many companies fail to change these default passwords because there are so many devices to manage.

**IoT monitoring software options to consider**

**AWS IoT Device Management**

AWS IoT Device Management is a specialized module in the Amazon family of cloud services. It's easy to set up, and it's accessible via a web browser. It can handle bulk and individual IoT device registration, and it groups devices to make monitoring and updating more efficient. It also offers superior intrusion prevention and detection. AWS prices by number of devices and usage.

**Splunk for Industrial IoT**

Splunk for Industrial IoT was designed for industrial markets but is useful for any industry. It uses built-in machine learning and AI features to predict performance and device health and enforce security. It offers a wide range of integrations, and it scales easily across enterprise- sized networks. It's available for a 60-day free trial as part of Splunk Enterprise.

**1.3: ADVANTAGES**

**1. Smart home devices**

Smart home devices are the most common consumer IoT products. AI assistant speakers, smart locks and connected appliances are just some smart home device examples that help people live more efficient, connected lives.

An IoT-connected fridge can tell homeowners when they have low stock of specific groceries. Similarly, an IoT doorbell could sense when someone approaches the front door, even before they ring the bell.

**2. Industrial sensors**

Industrial IoT (IIoT) supports use cases in automation, safety and data collection. With industrial sensors, companies can gain valuable insights and capabilities in the workplace, warehouses, manufacturing plants and construction sites.

Industrial vehicles and machinery can have IIoT sensors that track performance and output. Such sensors also enable predictive maintenance, which prevents expensive breakdowns, costly repairs and downtime.

##### 3. Industrial robots

Industrial organizations also use IoT technology in robots. For example, many warehouses use autonomous courier robots. These robots have IoT sensors that report their location and performance on the warehouse floor. IoT sensors help with navigation and can scan QR codes throughout the warehouse to guide their route.

##### 4. Healthcare devices

In healthcare facilities, IoT devices are crucial to patient care. IoT medical devices help nurses remotely monitor patient vitals so that nursing teams can provide better care to their departments.

**CHAPTER -2**

**LITERATURE SURVEY**

Zumyla Shanaz, Prem Kumar S investigated ‘IoT based Industrial Pollution Monitoring System’. The Author proposed to build a robust system that continuously monitors the air quality around the industry by monitoring the level of various pollutants released during the industrial process with less human. intervention with the aim of providing a healthy environment to workers of the industries. Here the author used MQ-6 and MQ2 sensors for analyzing the level of CO, CO2, and quality of smoke released in the atmosphere with the help of GSM technology for the exchange of the data from sensors to the monitoring authorities.

Ms.Aarthi, Karan Kapoor addressed ‘Air and Sound Pollution Monitoring System Using IoT’.In this research, the author develops a system that monitors both air quality and intensity of noise produced during the industrial process using various sensors like MQ135 to detect the level of CO2 in the atmosphere, DHT11 sensor to monitor both temperatures and humidity, LM393 sensor to monitor sound intensity. The system is integrated using Raspberry Pi 3B module, Raspberry Pi 3B is an ARM-based credit card sized SBC (Single Board Computer) which has an inbuilt Wi-Fi and Bluetooth module in it. The author used GPRS technology to exchange the data from the sensors to the specified locations through IoT.

Kavitha.B.C. B.C, Deepa Jose proposed ‘IoT based pollution monitoring system using raspberry – pi’ Here the author develops a system that comprises various sensors like MQ-6, MQ-7, MQ-135, LDR, and DHT11 sensors to monitor the presence of various pollution causing parameters like carbon monoxide, carbon dioxide, smoke and butane, the system also monitors the increase in the atmospheric temperature and humidity due to pollutants released. The microcontroller used is raspberry pi with an inbuilt Wifi module. The system continuously keeps an eye on data of the quantity of pollutants in the atmosphere and sends alarming messages to the governing authorities when the emission of pollutants exceeds the pre-set values by the pollution board.

**CHAPTER -3**

**BLOCK DIAGRAM**

VEHICLE ‘A’

**POWER SUPPLY**

**ARDUINO MEGA 2500**

**LCD DISPLAY (16\*8)**

**SOUND SENSOR**

**BUZZER**

**TEMP SENSOR**

**LED**

**MQ6 SENSOR**

**MQ9 SENSOR**

**Figure 3.0: Block diagram**

**3.1 Arduino UNO**

The Arduino UNO is a standard board of Arduino. Here UNO means 'one' in Italian. It was named as UNO to label the first release of Arduino Software. It was also the first USB board released by Arduino. It is considered as the powerful board used in various projects. Arduino.cc developed the Arduino UNO board.



**Figure 3.1:Arduino uno**

The Arduino UNO includes 6 analog pin inputs, 14 digital pins, a USB connector, a power jack, and an ICSP (In-Circuit Serial Programming) header. It is programmed based on IDE, which stands for Integrated Development Environment. It can run on both online and offline platforms.

**Technical Specifications of Arduino UNO**

* There are 20 Input/Output pins present on the Arduino UNO board. These 20 pis include 6 PWM pins, 6 analog pins, and 8 digital I/O pins.
* The PWM pins are Pulse Width Modulation capable pins.
* The crystal oscillator present in Arduino UNO comes with a frequency of 16MHz.
* It also has a Arduino integrated WiFi module. Such Arduino UNO board is based on the Integrated WiFi ESP8266 Module and ATmega328P microcontroller.
* The input voltage of the UNO board varies from 7V to 20V.
* Arduino UNO automatically draws power from the external power supply. It can also draw power from the USB.

**IMPORTANT NOTE:**



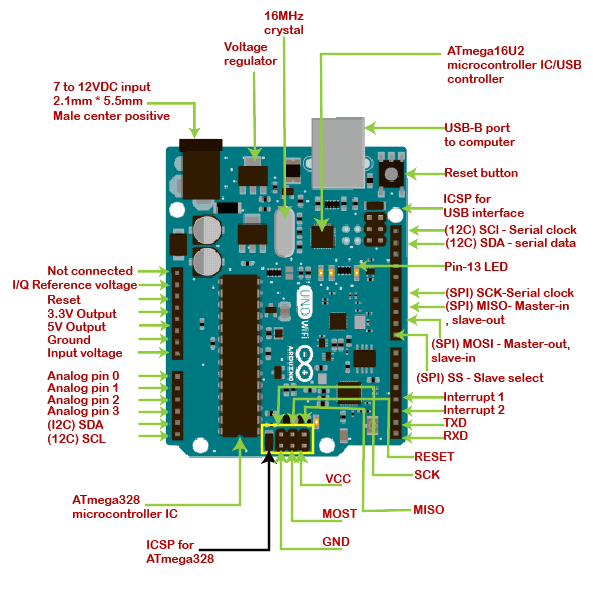
**Figure 3.1: Arduino uno naming**

* **ATmega328 Microcontroller**- It is a single chip Microcontroller of the ATmel family. The processor code inside it is of 8-bit. It combines **Memory (SRAM, EEPROM, and Flash), Analog to Digital Converter, SPI serial ports, I/O lines, registers, timer, external and internal interrupts, and oscillator.**
* **ICSP pin**- The In-Circuit Serial Programming pin allows the user to program using the firmware of the Arduino board.
* **Power LED Indicator**- The ON status of LED shows the power is activated. When the power is OFF, the LED will not light up.
* **Digital I/O pins**- The digital pins have the value HIGH or LOW. The pins numbered from D0 to D13 are digital pins.
* **TX and RX LED's**- The successful flow of data is represented by the lighting of these LED's.
* **AREF-**The Analog Reference (AREF) pin is used to feed a reference voltage to the Arduino UNO board from the external power supply.
* **Reset button**- It is used to add a Reset button to the connection.
* **USB**- It allows the board to connect to the computer. It is essential for the programming of the Arduino UNO board.
* **Crystal Oscillator**- The Crystal oscillator has a frequency of 16MHz, which makes the Arduino UNO a powerful board.
* **Voltage Regulator**- The voltage regulator converts the input voltage to 5V.
* **GND**- Ground pins. The ground pin acts as a pin with zero voltage.
* **Vin**- It is the input voltage.
* **Analog Pins**- The pins numbered from A0 to A5 are analog pins. The function of Analog pins is to read the analog sensor used in the connection. It can also act as GPIO (General Purpose Input Output) pins.

**Arduino UNO Pinout**

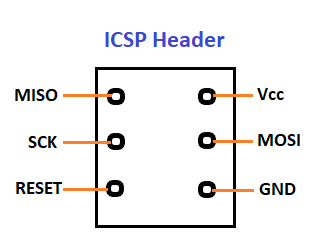
The Arduino UNO is a standard board of Arduino, which is based on an **ATmega328P** microcontroller. It is easier to use than other types of Arduino Boards.

The Arduino UNO Board, with the specification of pins, is shown below:



**Figure 3.1: Arduino uno pinout**

* **ATmega328 Microcontroller**- It is a single chip Microcontroller of the ATmel family. The processor core inside it is of 8-bit. It is a low-cost, low powered, and a simple microcontroller. The [Arduino UNO](https://www.javatpoint.com/arduino-uno) and Nano models are based on the ATmega328 Microcontroller.
* **Voltage Regulator**-The voltage regulator converts the input voltage to 5V. The primary function of voltage regulator is to regulate the voltage level in the [Arduino board](https://www.javatpoint.com/arduino-boards). For any changes in the input voltage of the regulator, the output voltage is constant and steady.
* **GND** - Ground pins. The ground pins are used to ground the circuit.
* **TXD and RXD**-TXD and RXD pins are used for serial communication. The TXD is used for transmitting the data, and RXD is used for receiving the data. It also represents the successful flow of data.
* **USB Interface**-The USB Interface is used to plug-in the USB cable. It allows the board to connect to the computer. It is essential for the programming of the [Arduino](https://www.javatpoint.com/arduino) UNO board.
* **RESET**-It is used to add a Reset button to the connection.
* **SCK**-It stands for **Serial Clock**. These are the clock pulses, which are used to synchronize the transmission of data.
* **MISO**-It stands for **Master Input/ Slave Output**. The save line in the MISO pin is used to send the data to the master.
* **VCC**-It is the modulated DC supply voltage, which is used to regulate the IC's used in the connection. It is also called as the primary voltage for IC's present on the Arduino board. The Vcc voltage value can be negative or positive with respect to the GND pin.
* **Crystal Oscillator-** The Crystal oscillator has a frequency of 16MHz, which makes the Arduino UNO a powerful board.
* **ICSP**-It stands for **In-Circuit Serial Programming**. The users can program the Arduino board's firmware using the ICSP pins.
* The program or firmware with the advanced functionalities is received by microcontroller with the help of the ICSP header.
* The ICSP header consists of 6 pins.
* The structure of the ICSP header is shown below:



**It is the top view of the ICSP header.**

* **SDA**-It stands for **Serial Data**. It is a line used by the slave and master to send and receive data. It is called as a **data line,** while SCL is called as a clock line.
* **SCL**-It stands for **Serial Clock**. It is defined as the line that carries the clock data. It is used to synchronize the transfer of data between the two devices. The Serial Clock is generated by the device and it is called as master.
* **SPI**-It stands for **Serial Peripheral Interface**. It is popularly used by the microcontrollers to communicate with one or more peripheral devices quickly. It uses conductors for data receiving, data sending, synchronization, and device selection (for communication).
* **MOSI**-It stands for Master Output/ Slave Input.The MOSI and SCK are driven by the Master.ADVERTISEMENT
* **SS**-It stands for **Slave Select**. It is the Slave Select line, which is used by the master. It acts as the enable line.
* **I2C**-It is the two-wire serial communication protocol. It stands for Inter Integrated Circuits. The I2C is a serial communication protocol that uses SCL (Serial Clock) and SDA (Serial Data) to receive and send data between two devices.

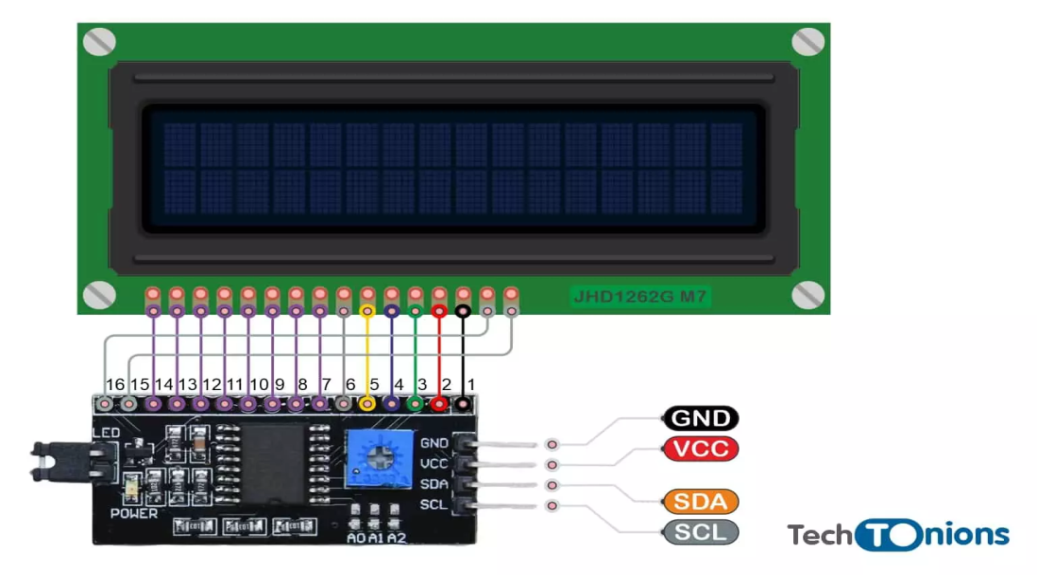
**3.2: LIQUID CRYSTAL DISPLAY (LCD) WITH I2C MODULE**

This LCD screen is a 16x2 character LCD display with an I2C interface. It features two rows for displaying text, with each row capable of displaying up to 16 characters. The white characters will be displayed on a blue background, providing a clear and visually appealing display.

In traditional Arduino LCD display projects, there are often limitations in terms of available pins, especially when using Arduino Uno. Additionally, wiring and connections can become quite complex. To address these issues, we introduce this I2C 16x2 Arduino LCD display, which utilizes the I2C communication interface.

This means that it only requires 4 pins to connect the LCD display, including VCC, GND, SDA, and SCL. By adopting the I2C interface, we can save at least 4 digital/analog pins on the Arduino, making the project's connections simpler and more convenient.

**The pin outs are as follows:**



**Fig 3.2: LCD Pin out**

**SPECIFICATION**

* Compatible with Arduino/Genuino [UNO](https://www.dfrobot.com/product-838.html), [Leonardo](https://www.dfrobot.com/product-832.html), [Mega](https://www.dfrobot.com/product-1175.html), 101 (Intel Curie), Micro, [Nano](https://www.dfrobot.com/product-1122.html), Mini
* I2C Address:0x20-0x27(0x20 default)
* Back lit (Blue with white char color)
* Supply voltage: 5V
* Interface:I2C/TWI x1,Gadgeteer interface x2
* Adjustable contrast
* Size: 80x36x20mmz(3.1x1.4x0.7in)

**3.3 MQ-6 sensor**

The MQ-6 Sensor is used to detect the concentration of various combustible gases like propane, Butane, and LPG in the atmosphere; it also detects the presence of natural gases. The MQ-6 sensor can detect gas concentration nearly from 200 to 10000 ppm. The output of the sensor is dependent on the analog resistance. The maximum operating voltage is 5V.



**Figure : 3.3 MQ -6 sensor**

**SPECIFICATIONS:-**

|  |  |  |  |
| --- | --- | --- | --- |
| Model No. | | MQ-6 | |
| Sensor Type | | Semiconductor | |
| Standard Encapsulation | | Bakelite (Black Bakelite) | |
| Detection Gas | | Isobutane, Butane, LPG | |
| Concentration | | 300-1000ppm  (Butane, Propane, LPG) | |
| Circuit | Loop Voltage | VC | <24V DC |
| Heater Voltage | VH | 5.0+0.2V ACor DC |
| Load Resistance | RL | Adjustable |
| Character | Heater resistance | RH | 31+3room Tem. |
| Heater consumption | PH | 900mw |
| Sensing Resistance | RS | 2k-20V9(in 2000ppm C3H8 |
| Sensitivity | S | Rs(in air)/Rs (1000ppm C4H10)>5 |
| slope |  | <0.6(R2000ppm/R1000ppmLPG) |
| Condition | Tem. Humidity | 20+265%+5%RH |
| Standard test circuit | VC:5.0v+0.1v  VH:5.0v+0.1v |
| Preheat time | Over 48 hours |

**Table 3.3: Specification**

**FEATURES**

* Voltage: 5V
* Detecting concentration :
* 200-10000 ppm LPG
* iso-butane
* propane
* LNG
* Analog and Digital Output
* Digital Out is High or Low based on a adjustable preset threshold.

**APPLICATIONS**

* Domestic gas leakage detector
* Industrial Combustible gas detector
* Portable gas detector

**3.4 MQ-9 SENSOR**

**Figure 3.4:MQ-9 Sensor**

**T**he MQ-9 Sensor is used to detect the concentration of CO in the atmosphere. The MQ-9 sensor is very sensitive to CO; the MQ-9 sensor can also be used to detect various other gases containing CO in them. Initially, the conductivity of the sensor is low in the clean or neat air, when the concentration of the gases increases the conductivity of the sensor increases the maximum operating voltage is 5V

**Specfication**

|  |  |
| --- | --- |
| Model | MQ-9 |
| Operating voltage (VDC) | 5 |
| Current consumption(MA) | 150 |
| D0 output | TTL digital 0 and 1(0.1 and5V) |
| A0 output | 0.1-0.3(relatively clean) |
| Length(mm): | 32 |
| Width(mm): | 20 |
| Height | 16 |
| Weight(g): | 5 |
| Shipping Dimensions | 5\*5\*4cm |

**Table 3.4: Specfication**

**Features :**

1. Good sensitivity to CO/Combustible Gas
2. High sensitivity to Methane, Propane, and CO
3. Long life and low cost
4. Simple drive circuitnsor increases. The maximum operating voltage is 5V.

**Applications**

1. The domestic gas leakage detector
2. Industrial gas detector
3. vPortable gas detector

**3.5: LM393 Sensor**

****

**Figure 3.5: LM393 Sensor**

LM393 sensor is used to detect the intensity of the sound in the environment, it converts the difference in the air pressure into electrical signals. The sound waves produced will make the diaphragm of the sensor to vibrate, intern making the tiny magnets to vibrate which is present inside the sensor .this vibration induces the current in the coil which is a direct measure of the intensity of the sound. It develops a binary indication of the sound and also the analog representation of the sound obtained. The operating voltage is between 4-5V.

**Features :**

1. Signal output indication.
2. The output is digital.
3. Single-channel signal output.
4. With the retaining bolt hole, convenient installation.
5. Output low level and the signal light when there is sound.

**Specfication:-**

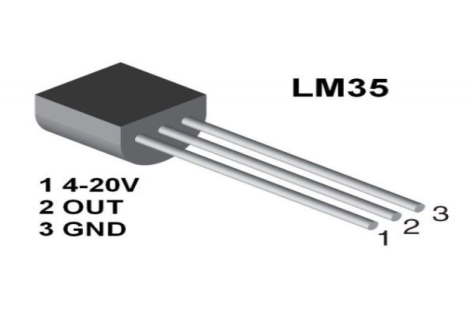
|  |  |
| --- | --- |
| Operating voltage(VDC) | 33-5 |
| IC chip | LM393 |
| Length(mm): | 36.5 |
| Width(mm): | 17 |
| Height(mm): | 10 |
| Weight(g): | 4 |
| Shipping Weight | 0.005kg |
| Shipping Dimensions | 5\*3\*2cm |

**Table 3.5: Specification**

**Application:-**

* Consumer electronics such as phones, computers, music systems
* Security and Monitoring systems such as burglar alarms, door alarm, etc.
* Home automation such as lighting your house by detecting whistle/clap instead of physically turning the light switch
* Ambient sound recognition and sound level recognition

**3.6: LM35 Temperature sensor**



**Figure 3.6: LM35 Sensor**

The LM35 sensor is used to detect the ambient temperature. When there is a 1 degree change in the temperature the sensor shows a voltage difference of 10mV. Generally, an LM35 sensor has 3 terminals, an input terminal, an output terminal, and a terminal connected to the ground. It works on the principle that temperature is directly proportional to the difference in the voltage between the two terminals. It does not require any external calibration. It measures about -55 degrees to 150 degrees. The system uses an LM35 sensor to detect the temperature of the heat-treated water before letting it into the natural reservoir. The maximum output voltage is between 1.5V.

**Specification:-**

|  |  |
| --- | --- |
| **Local sensor accuracy (max)** | **0.5, 1** |
| **Operating temperature range (°C)** | **-55 to 150** |
| **Supply voltage (min) (V)** | **4** |
| **Supply voltage (max) (V)** | **30** |
| **Supply current (max) (µA)** | **114** |
| **Interface type Analog** | **Output** |
| **Sensor gain (mV/°C)** | **10** |
| **Rating** | **Catalog** |
| **Features** | **UL recognized** |

**Table 3.6: Specification**

**Features:-**

* Calibrated Directly in Celsius (Centigrade)
* Linear + 10-mV/°C Scale Factor
* 0.5°C Ensured Accuracy (at 25°C)
* Rated for Full −55°C to 150°C Range
* Suitable for Remote Applications
* Low-Cost Due to Wafer-Level Trimming
* Operates From 4 V to 30 V
* Less Than 60-µA Current Drain
* Low Self-Heating, 0.08°C in Still Air
* Non-Linearity Only ±¼°C Typical
* Low-Impedance Output, 0.1 Ω for 1-mA Load

**Applications:-**

* It's used for measuring the temperature of a particular environment.
* It provides thermal shutdown for a circuit or component used in a specific project.
* It can be used for battery temperature measurement. It provides battery protection from overheating.
* It can be used in HVAC applications as a temperature measurement device.

**CHAPTER -4**

**SOFTWARE**

**4.1: Software Introduction:**

Arduino IDE is an open-source software, designed by Arduino.cc and mainly used for writing, compiling & uploading code to almost all Arduino Modules.It is an official Arduino software, making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process.It is available for all operating systems i.e. ( MAC, Windows, Linux and runs on the Java) Platform that comes with inbuilt functions and commands that play a vital role in debugging, editing and compiling the code.A range of Arduino modules available including Arduino Uno,



**Figure 4.1: Software**

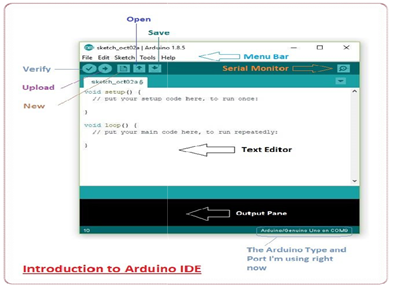
**4.1.1: Definition of Arduino IDE**

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online.

ABOUT IDE

The IDE environment is mainly distributed into three sections :

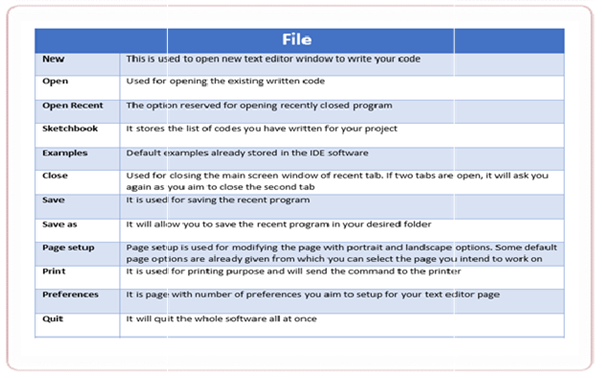
\*Menu Bar \*Text Editor \*Output Pane



**Table 4.1.1:Arduino IDE**

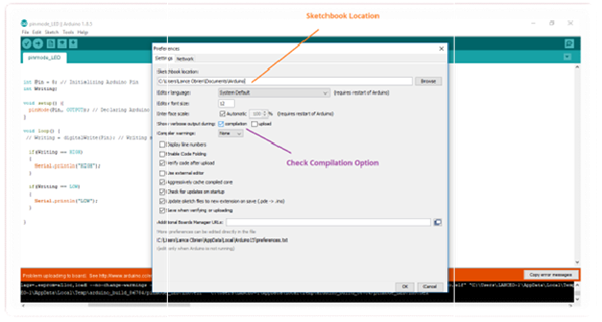
The bar appearing on the top is called **Menu Bar** that comes with five different options as follow

File - You can open a new window for writing the code or open an existing one. The following table shows the number of further subdivisions the file option is categorized into



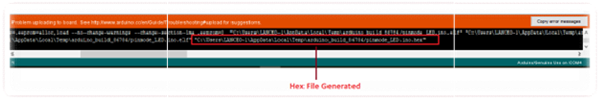
**Table 4.1.1: File**

As you go to the preference section and check the compilation section, the Output Pane will show the code compilation as you click the upload button.



**Table 4.1.1: Preference**

And at the end of the compilation, it will show you the hex file it has generated for the recent sketch that will send to the Arduino Board for the specific task you aim to achieve.



**Table 4.1.1: Hex file**

**Edit** - Used for copying and pasting the code with further modification for font

**Sketch** - For compiling and programming

**Tools** - Mainly used for testing projects. The Programmer section in this panel is used for burning a bootloader to the new microcontroller.

**Help** - In case you are feeling skeptical about software, complete help is available from getting started to troubleshooting.

**4.2: PROGRAM**

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27,16,2); //for 16x2 lcd display

int buzzer1 = 13;

int GASA0 = A0;

int gasvalue1;

int buzzer2 = 13;

int GASA1 = A1;

int gasvalue2;

int num\_Measure = 128 ; // Set the number of measurements

int pinSignal = A0; // pin connected to pin O module sound sensor

int redLed = 5;

long Sound\_signal; // Store the value read Sound Sensor

long sum = 0 ; // Store the total value of n measurements

long level = 0 ; // Store the average value

int soundlow = 40;

int soundmedium = 500;

#define LM35PIN A0

int adcValue;

float milliVolt;

float tempInC;

float tempInF;

byte degree\_symbol[8] =

{

0b00111,

0b00101,

0b00111,

0b00000,

0b00000,

0b00000,

0b00000,

0b00000

};

void setup() {

lcd.init(); //initiute the lcd1

lcd.init();

lcd.backlight();

pinMode(buzzer1, OUTPUT);

lcd.setCursor(2,0);

lcd.print("LPG LEAKAGE1");

lcd.setCursor(1,1);

lcd.print("LEVEL DETECTOR1");

delay(5000);

lcd.init(); // initiate the lcd

lcd.init();

lcd.backlight();

Serial.begin(9600);

pinMode(buzzer2, OUTPUT);

lcd.setCursor(3,0);

lcd.print("welcome to");

lcd.setCursor(1,1);

lcd.print("JOSH ARDUINO");

delay(5000);

lcd.init();

lcd.backlight();

lcd.setCursor(0,0);

lcd.print("LM35 Temperature");

lcd.setCursor(0,1);

lcd.print("Sensor - Arduino");

lcd.createChar(1, degree\_symbol);

delay(2000);

lcd.clear();

}

void loop() {

int analogSensor1 = analogRead(GASA0);

int gasvalue1=(analogSensor1-50)/25; //gas module sensitivity

lcd.setCursor(0,0);

lcd.print("LPG LEVEL1:");

lcd.setCursor(10,0);

lcd.print(gasvalue1);

lcd.setCursor(12,0);

lcd.print("%");

//check if it has reached the threshold value

if (gasvalue1 >= 40) //gas percentage alert

{

lcd.setCursor(0,1);

lcd.print("DANGER1");

tone(buzzer1, 5000, 3000);

}

else

{

lcd.setCursor(0,1);

lcd.print("NORMAL1");

noTone(buzzer1);

}

delay(500);

lcd.clear();

int analogSensor2 = analogRead(GASA1);

int gasvalue2=(analogSensor2-50)/35; //gas module sensitivity

lcd.setCursor(0,0);

lcd.print("GAS Level2:");

lcd.setCursor(10,0);

lcd.print(gasvalue2);

lcd.setCursor(12,0);

lcd.print("%");

// Checks if it has reached the threshold value

if (gasvalue2 >= 10) //gas percentage alert

{

lcd.setCursor(0,1);

lcd.print("DANGER2");

tone(buzzer2, 1000, 10000);

}

else

{

lcd.setCursor(0,1);

lcd.print("NORMAL2");

noTone(buzzer2);

}

delay(500);

lcd.clear();

// Performs 128 signal readings

for ( int i = 0 ; i <num\_Measure; i ++)

{

Sound\_signal = analogRead (pinSignal);

sum =sum + Sound\_signal;

}

level = sum / num\_Measure; // Calculate the average value

Serial.print("Sound Level: ");

lcd.print("Sound Level= ");

Serial.println (level-33);

lcd.print(level-33);

if(level-33<soundlow)

{

lcd.setCursor(0,2);

lcd.print("Intensity= Low");

digitalWrite(redLed,LOW);

}

if(level-33>soundlow && level-33<soundmedium)

{

lcd.setCursor(0,2);

lcd.print("Intensity=Medium");

digitalWrite(redLed,LOW);

}

if(level-33>soundmedium)

{

lcd.setCursor(0,2);

lcd.print("Intensity= High");

digitalWrite(redLed,HIGH);

}

sum = 0 ; // Reset the sum of the measurement values

delay(200);

lcd.clear();

adcValue = analogRead(LM35PIN);

milliVolt = ((adcValue \* 4870.0) / 1024);

tempInC = milliVolt/10; /\* Temperature in Degree Celsius \*/

tempInF = ((tempInC \* 9/5) + 32); /\* Temperatue in Degree Fahrenheit \*/

lcd.setCursor(0,0);

lcd.print("Temp = ");

lcd.setCursor(7,0);

lcd.print(tempInC);

lcd.write(1);

lcd.print("C");

lcd.setCursor(0,1);

lcd.print("Temp = ");

lcd.setCursor(7,1);

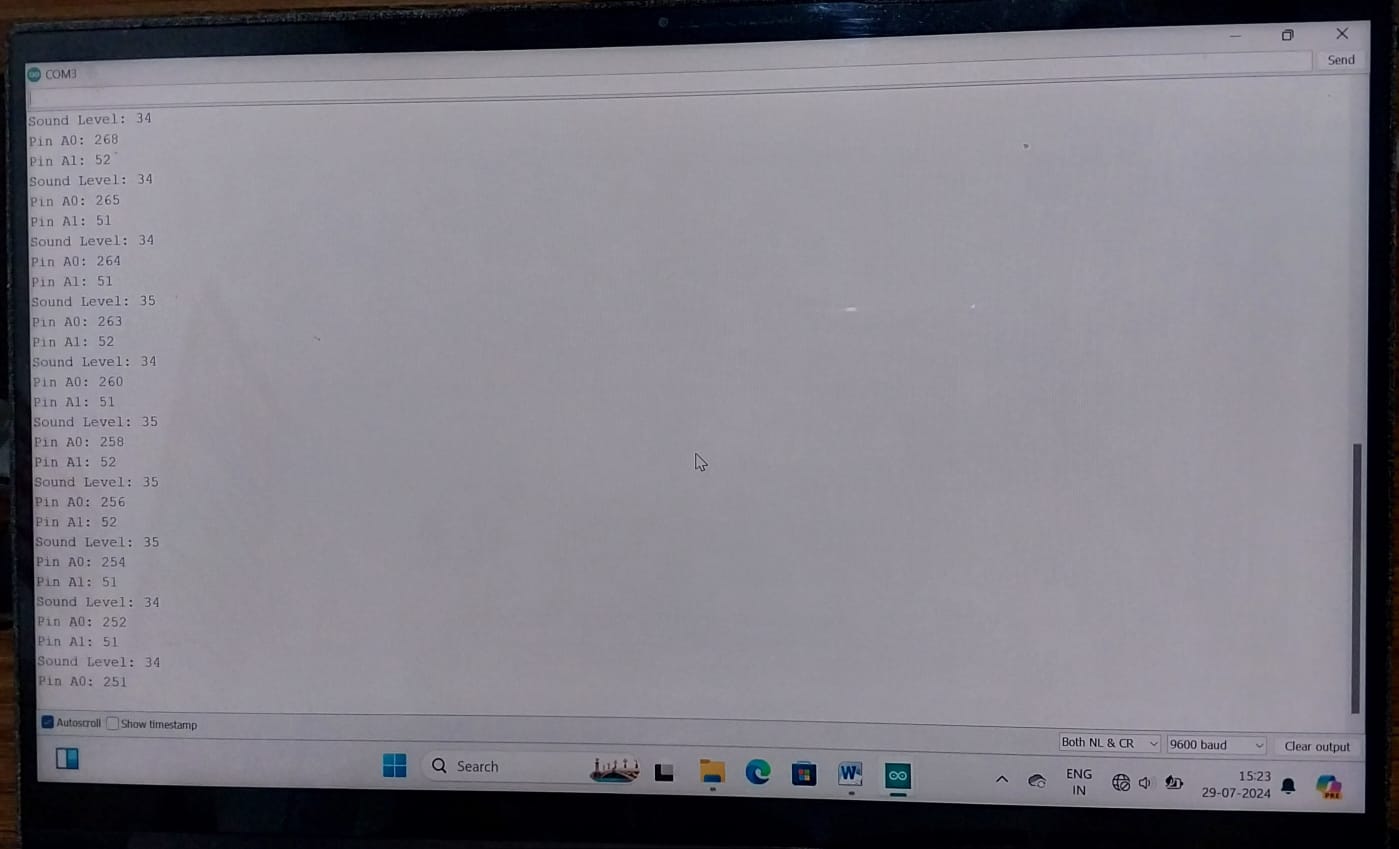
lcd.print(tempInF);

lcd.write(1);

lcd.print("F");

delay(500);

}

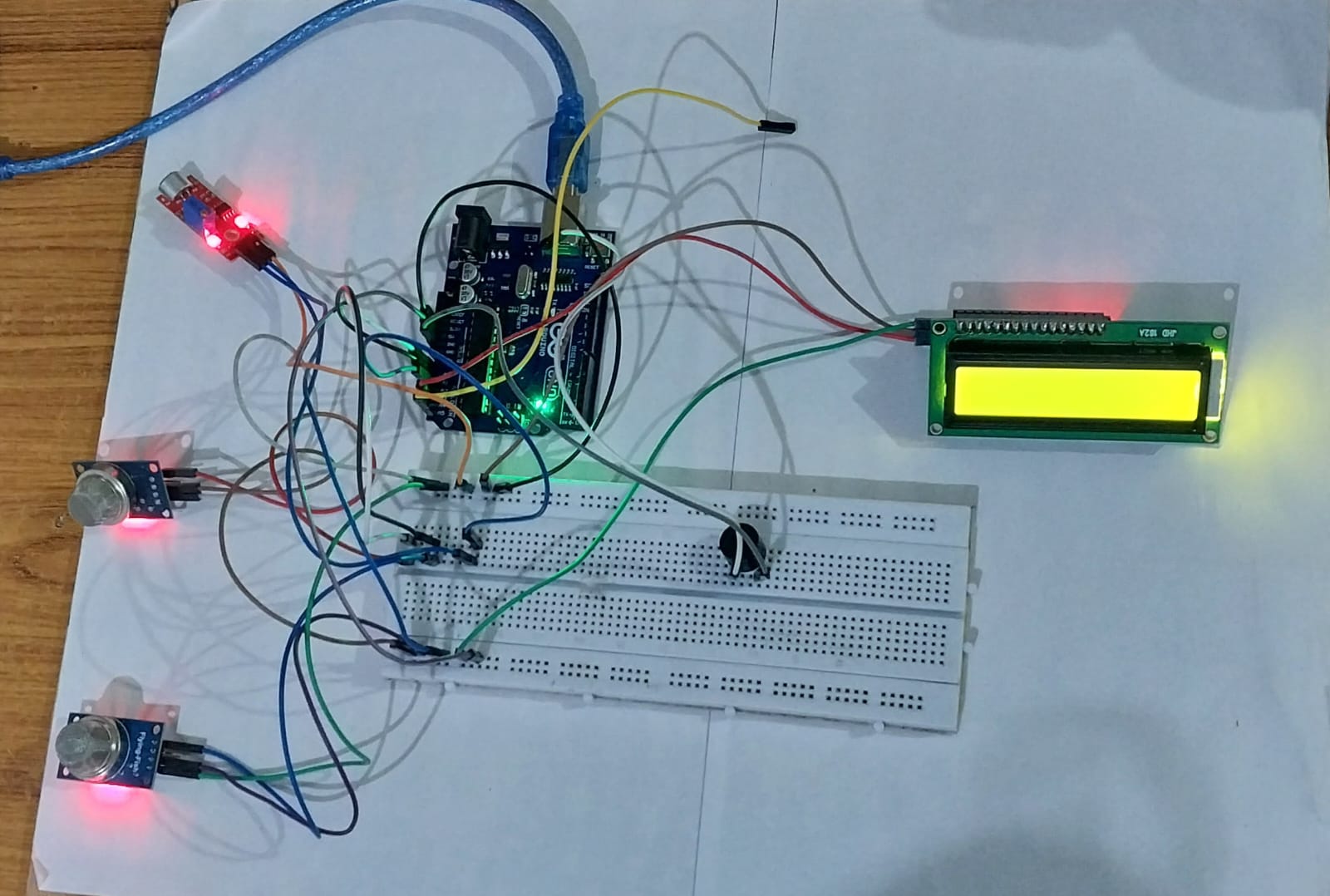


**Figure 4.2: output**

**CHAPTER-5**

**RESULT**

The IOT concept can be applied to a wide range of application. We implemented the use of IOT in Industrial pollution monitoring and this project, real time air pollution monitoring system based on IOT is presented. Real time monitoring of air quality parameters ensures that the industrial emissions levels are maintained throughout and helps us to track all the data in a single place (i.e) cloud and reduce the pollution based on the collected data.The implementation cost is very economical as the sensors and the microcontrollers are easily available. The online database system has increased the flexibility by updating all current parameters of the industries over a common server. The camera can be upgraded in future with image processing algorithms to enable a fully automated system for safety, such as fire alarm, gas leakage. This enables to generate automated control action in the absence of the authorized user.

****

**Figure5.0: Result On IOT BASED INDUSTRIAL AIR, WATER AND NOISE POLLUTION MONITORING SYSTEM**

**CONCLUSION**

The “Iot based industrial Air, water, and Noise pollution monitoring system” is designed to monitor and control the pollution caused by the release of harmful, untreated industrial pollutants in a costeffective and highly secured manner. This system is more effective than the existing system since it helps the environmental pollution control authorities in monitoring the industrial environmental conditions with the help of 7 sensors interfaced to the advanced controller which collects and records the real-time data of various pollutants released by the industries through various means, which helps the authorities to keep control and decrease Air, Water, Thermal, and noise pollution caused by the industries. Also, the project webpage continuously records the data in detail along with its data and time of capture which cannot be erased or deleted and can get accessed to the sensor data on any date and time and Based upon the collected data, the respective action can be taken on controlling the environmental pollution.

**REFERENCES**

[1] Zumyla Shanaz, Prem Kumar S, Rahul R, Rajesh Kumar,and Santhosh Kumar. "IoT based Industrial Pollution Monitoring System” (2019).

[2] Deshmukh, Sarika, Saurabh Surendran, and M. P. Sardey. "Air and Sound Pollution Monitoring System using IoT." International Journal on Recent and Innovation Trends in Computing and Communication 5.6 (2017).

[3] BC, Kavitha, and Deepa Jose. "IoT Based Pollution Monitoring System using Raspberry-PI." International Journal of Pure and Applied Mathematics 118.24 (2018).

[4] Nasution, T. H., M. A. Muchtar, and A. Simon. "Designing an IoT-based air quality monitoring system." IOP Conference Series: Materials Science and Engineering. Vol. 648. No. 1. IOP Publishing, 2019.

[5] Mahammad, D. V. "Design and Implementation of IoT based Portable Outdoor Dust Density Monitoring System." (2019).

[6] Sai, Palaghat Yaswanth. "An IoT Based Automated Noise and Air Pollution Monitoring System." International Journal of Advanced Research in Computer and Communication Engineering 6.3 (2017): 419-423.

[7] Madhavireddy, Vennam, and B. Koteswarrao. "Smart water quality monitoring system using IoT technology." International Journal of Engineering and Technology (UAE) 7 (2018): 636-639.

[8] Anuradha, T., Chaitra R. Bhakti, and D. Pooja. "IoT based low cost system for monitoring of water quality in real time." Int. Res. J. Eng. Technol.(IRJET) 5.5 (2018).

[9] Daigavane, Vaishnavi V., and M. A. Gaikwad. "Water quality monitoring system based on IoT." Department Electronics & Telecommunication Engineering, Advances in Wireless and Mobile Communications 10.5 (2017): 1107-1116