CHAPTER 1

INTRODUCTION

Liquefied Petroleum Gas (LPG) is a commonly used energy source in households, industries, and commercial establishments due to its high calorific value and ease of use. Despite its widespread adoption, the handling and monitoring of LPG cylinders pose significant challenges, particularly in terms of ensuring uninterrupted supply and addressing potential safety hazards. Traditional methods of manually monitoring LPG levels are often inconvenient, prone to oversight, and unable to address emergency situations effectively. Moreover, accidental LPG leaks can lead to catastrophic incidents, underscoring the need for a reliable and automated monitoring solution.

The Internet of Things (IoT) has emerged as a transformative technology, enabling smart and connected systems that can address these challenges. This project proposes an IoT-based LPG Leakage and Level Detection System integrated with an auto-booking mechanism using the ThingSpeak platform. The system leverages an Arduino microcontroller, a pressure sensor, and a gas sensor to continuously monitor the gas level in the cylinder and detect leakage. The ESP8266 Wi-Fi module facilitates seamless communication with the ThingSpeak platform, enabling real-time alerts and automatic booking when the LPG level falls below a predefined threshold. Additionally, a buzzer provides an immediate audible alert in case of leakage, and SMS notifications are sent to users for prompt action.

By integrating safety and convenience, this system offers a smart and efficient solution for LPG monitoring and management. It ensures a continuous supply of gas, enhances user safety, and significantly reduces risks associated with LPG usage, making it an essential tool for modern households and industries.

OBJECTIVE

- **Aim**: Develop an IoT-based system for efficient and safe management of LPG cylinders.

- **Monitoring LPG Levels**: Uses a pressure sensor to monitor the gas levels in the cylinder.

- **Leakage Detection**: Detects LPG gas leakage using a gas sensor to enhance safety.

- **Real-Time Alerts**: Provides instant notifications and alerts to users in case of issues, such as leakage or low gas levels.

- **Buzzer Notification**: Alerts users with a buzzer for immediate attention.

- **Automated Booking**: Automates the LPG cylinder booking process via the ThingSpeak platform using the ESP8266 Wi-Fi module.

- **Safety & Convenience**: Ensures uninterrupted gas supply, reduces manual intervention, minimizes risks, and improves user experience.

CHAPTER 2

LITERATURE SURVEY

TITLE:

[1] B. D. Jolhe, P. A. Potdukhe, N. S. Gawai, "Automatic LPG Booking, Leakage Detection And Real TimeGas Measurement Monitoring System", International Journal of Engineering Research & Technology (IJERT) ,April 4, 2013.

[2] S.Rajitha, T.SWAPNA, "A Security Alert System Using Gsm For Gas Leakage", International Journal Of VLSI And Embedded Systems,2012.

1. **Rising Demand**:
   a. LPG cylinders are increasingly used in urban, suburban, and rural areas in the Indian Subcontinent.
   b. The demand for LPG cylinders has also grown in the industrial sector.
2. **Safety Concerns**:
   a. LPG is a highly flammable gas, and with the rising usage of cylinders, there has been an increase in gas leakage accidents.
   b. LPG leakage can lead to risks such as building fires, suffocation, and explosions.
   c. Many accidents occur due to negligence in detecting gas leaks.
3. **Challenges in Monitoring**:
   a. Constant monitoring of gas leakage is difficult and requires diligent effort.
4. **Proposed IoT System**:
   a. The IoT-based system offers precise and accurate gas leakage monitoring.
   b. **Gas Leakage Detection**: The MQ-6 gas sensor detects leakage and activates the alarm circuitry.
   c. **Alerts**: The system sends an SMS notification to the customer when a gas leak is detected.
5. **Cylinder Usage Monitoring**:
   a. The system continuously monitors the weight of the LPG cylinder to prevent untimely exhaustion.
   b. This helps customers know when to replace or refill their cylinder.

CHAPTER 3

PROPOSED SYSTEM

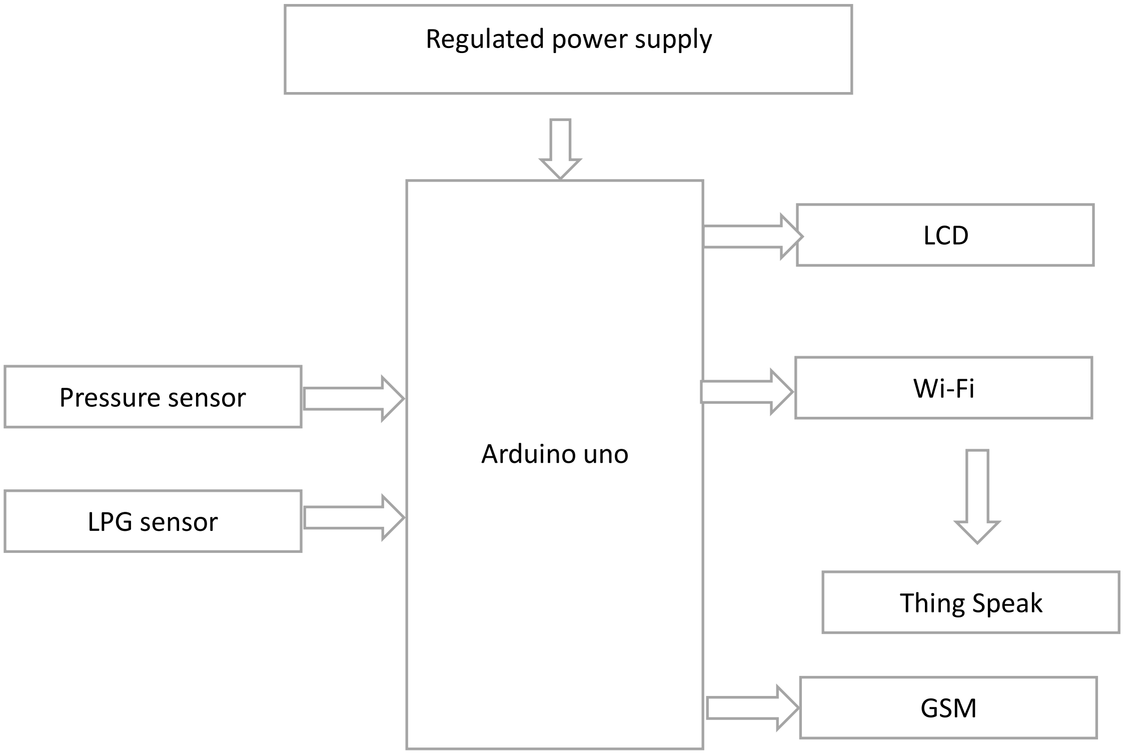
The proposed system is an IoT-enabled solution for LPG level detection, leakage monitoring, and automated booking. It utilizes an Arduino microcontroller as the central processing unit, interfaced with a pressure sensor to monitor the LPG level in the cylinder and a gas sensor to detect leakage. When the gas level drops below a predefined threshold, the system automatically sends an alert and initiates a refill booking request through the ThingSpeak IoT platform using the ESP8266 Wi-Fi module. In case of a gas leak, the system triggers a buzzer for immediate attention and sends an SMS notification to the user. This integrated approach ensures real-time monitoring, improves safety by addressing leaks promptly, and enhances convenience by eliminating manual level checks and booking processes.

BENEFITS OF PROPOSED SYSTEM

1. **Enhanced Safety:** Detects LPG leakage promptly, triggering buzzer alerts and SMS notifications to minimize potential hazards.
2. **Convenience:** Automates gas level monitoring and refill booking, eliminating the need for manual intervention.
3. **Uninterrupted Supply:** Ensures timely cylinder refills by sending alerts when gas levels fall below a predefined threshold.
4. **IoT Integration:** Enables real-time monitoring and communication through the ThingSpeak platform for efficient and remote system management.

CHAPTER 4

BLOCK DIAGRAM



Working Principle

1. **Startup**:
   - The Arduino initializes the pressure sensor and ESP8266 Wi-Fi module during startup.
   - The system connects to the configured Wi-Fi network and ThingSpeak platform.
2. **Gas Level Monitoring**:
   - The pressure sensor continuously measures the weight of the LPG cylinder.

- The Arduino processes the data and calculates the remaining gas level based on the weight.

3. **Threshold Detection**:
   - The gas level is compared with a predefined threshold and leakage detection.
   - If the level falls below the threshold, the system triggers an action.
4. **Data Transmission**:
   - The ESP8266 Wi-Fi module sends the gas level data to the ThingSpeak cloud platform.
   - ThingSpeak processes the data and sends a booking request to the LPG supplier.
5. **Alert Notification**:
   - If the gas level is critically low, the system triggers a buzzer alert for immediate attention.
   - The user is notified via the ThingSpeak platform or a mobile application.
6. **Automatic Booking**:
   - The ThingSpeak platform sends a preconfigured booking request to the supplier. And send sms.
7. **Real-time Monitoring:**
   - The system continuously monitors the gas level in real-time and updates the data on the ThingSpeak platform periodically.
   - Users can access the real-time gas level information remotely through a mobile app or web interface.
8. **Data Logging and Analysis:**
   - The system logs all gas usage data (e.g., daily consumption) on ThingSpeak for future analysis and usage patterns.
   - The data can be analyzed to predict the next refill date based on consumption trends.

CHAPTER 5

HARDWARE TOOLS

1. ARDUINO UNO
2. PRESSURE SENSOR
3. LPG SENSOR
4. LCD
5. WIFI
6. GSM
7. THING SPEAK
8. REGULATED POWER SUPPLY

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

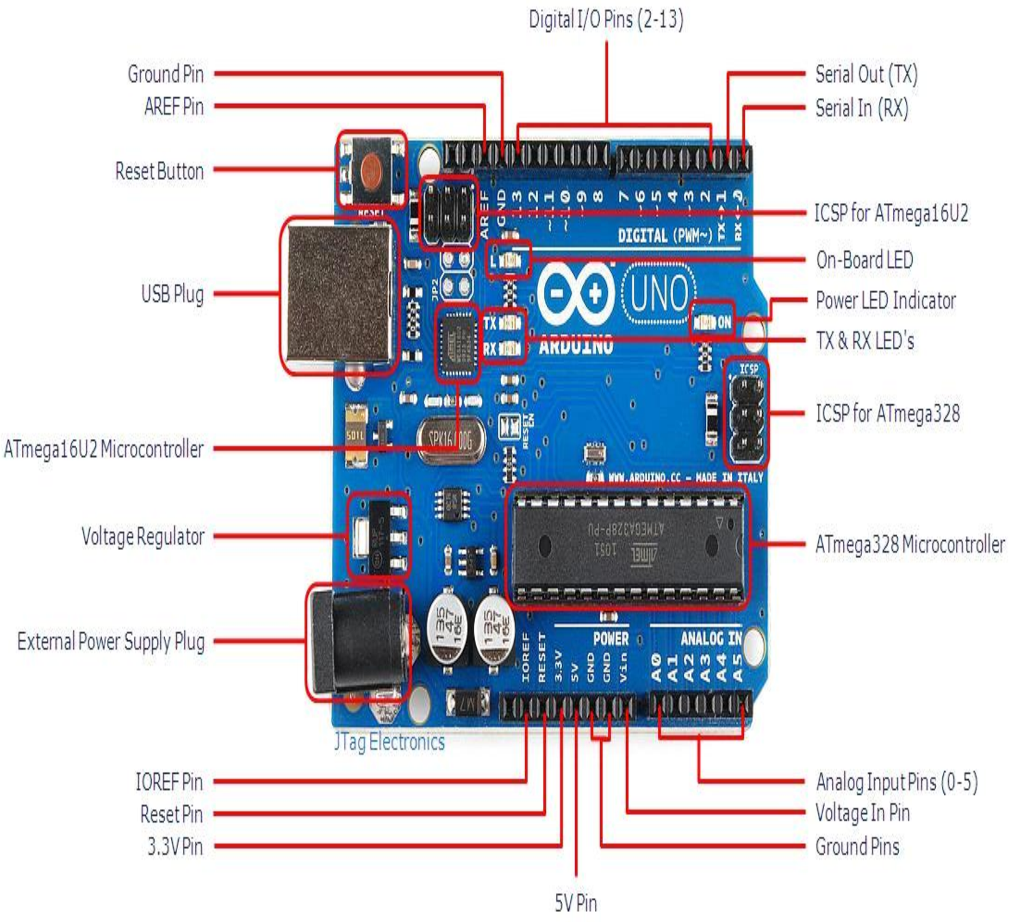


fig : Arduino Uno R3 Board

➢ 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.

2. GSM

Global System for Mobile Communications (GSM) modems are specialized types of modems that operate over subscription based wireless networks, similar to a mobile phone. A GSM modem accepts a Subscriber Identity Module (SIM) card, and basically acts like a mobile phone for a computer. Such a modem can even be a dedicated mobile phone that the computer uses for GSM network capabilities.

Traditional modems are attached to computers to allow dial-up connections to other computer systems. A GSM modem operates in a similar fashion, except that it sends and receive data through radio waves rather than a telephone line. This type of modem may be an external device connected via a Universal Serial Bus (USB) cable or a serial cable. More commonly, however, it is a small device that plugs directly into the USB port or card slot on a computer or laptop.

It is widely used mobile communication system in the world. GSM is an open and digital cellular technology used for transmitting mobile voice and data services operates at the 850MHz, 900MHz, 1800MHz and 1900MHz frequency bands.

Features

- Improved spectrum efficiency
- International roaming
- Compatibility with integrated services digital network (ISDN)
- Support for new services.
- SIM phonebook management
- Fixed dialing number (FDN)
- Real time clock with alarm management

GSM Modem:

A GSM modem is a wireless modem that works with a GSM wireless network. A wireless modem behaves like a dial-up modem. The main difference between them is that a dial-up modem sends and receives data through a fixed telephone line while a wireless modem sends and receives data through radio waves.

A GSM modem can be an external device or a PC Card / PCMCIA Card. Typically, an external GSM modem is connected to a computer through a serial cable or a USB cable. A GSM modem in the form of a PC Card / PCMCIA Card is designed for use with a laptop computer. It should be inserted into one of the PC Card / PCMCIA Card slots of a laptop computer. Like a GSM mobile phone, a GSM modem requires a SIM card from a wireless carrier in order to operate.

As mentioned in earlier sections of this SMS tutorial, computers use AT commands to control modems. Both GSM modems and dial-up modems support a common set of standard AT commands. You can use a GSM modem just like a dial-up modem.

In addition to the standard AT commands, GSM modems support an extended set of AT commands. These extended AT commands are defined in the GSM standards. With the extended AT commands, you can do things like:

- Reading, writing and deleting SMS messages.
- Sending SMS messages.
- Monitoring the signal strength.
- Monitoring the charging status and charge level of the battery.
- Reading, writing and searching phone book entries.

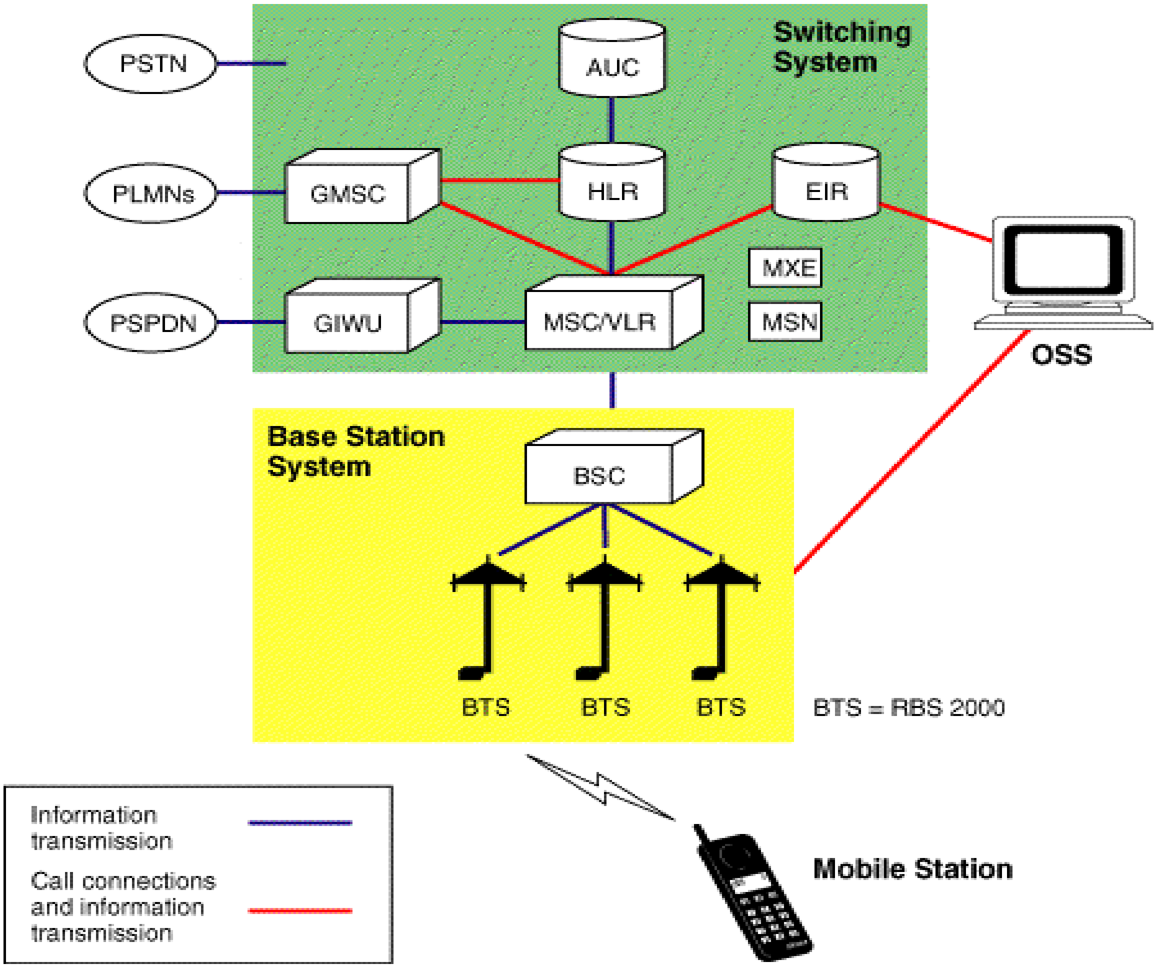
The number of SMS messages that can be processed by a GSM modem per minute is very low ,only about six to ten SMS messages per minute.

GSM Modem Application

The GSM Network

GSM Network

GSM provides recommendations, not requirements. The GSM specifications define the functions and interface requirements in detail but do not address the hardware. The reason for this is to limit the designers as little as possible but still to make it possible for the operators to buy equipment from different suppliers. The GSM network is divided into three major systems: the switching system (SS), the base station system (BSS), and the operation and support system (OSS).



Block diagram of GSM system

3. GAS SENSOR

A **gas sensor** is a device designed to detect and measure the presence of gases in the environment, playing a crucial role in safety, health, and environmental monitoring. It is an essential component in various applications, including residential, industrial, and environmental protection. Gas sensors detect hazardous gases by measuring changes in specific physical or chemical properties, ensuring early detection and prevention of accidents or health hazards.

Gas sensors work through different mechanisms. **Semiconductor-based sensors** change resistance when exposed to a target gas, while **electrochemical sensors** generate an electrical current as a result of a chemical reaction with the gas. **Infrared sensors** measure the absorption of infrared light by gas molecules, and **catalytic sensors** detect flammable gases by measuring the heat produced during the oxidation process. Each type of sensor is selected based on the application and the type of gas it needs to detect.

The applications of gas sensors are diverse. In **residential settings**, they are often integrated into gas leak detectors to identify leaks of gases like LPG or methane, preventing potential disasters. In **industrial environments**, gas sensors monitor harmful substances such as carbon monoxide, hydrogen sulfide, and ammonia, ensuring worker safety. In **environmental monitoring**, sensors measure air quality by detecting pollutants like ozone and nitrogen dioxide. Additionally, in **automotive systems**, gas sensors monitor exhaust emissions, ensuring compliance with environmental regulations.

In the **Electronic Eye Dial-Up System**, the gas sensor plays a vital role in ensuring safety by detecting harmful gases. It enables the system to send alerts through a GSM modem when gas hazards are detected, allowing for quick intervention. The integration of gas sensors with other components highlights their importance in providing a reliable, cost-effective, and scalable safety solution across various applications.

4. 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), animations and so on.

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.

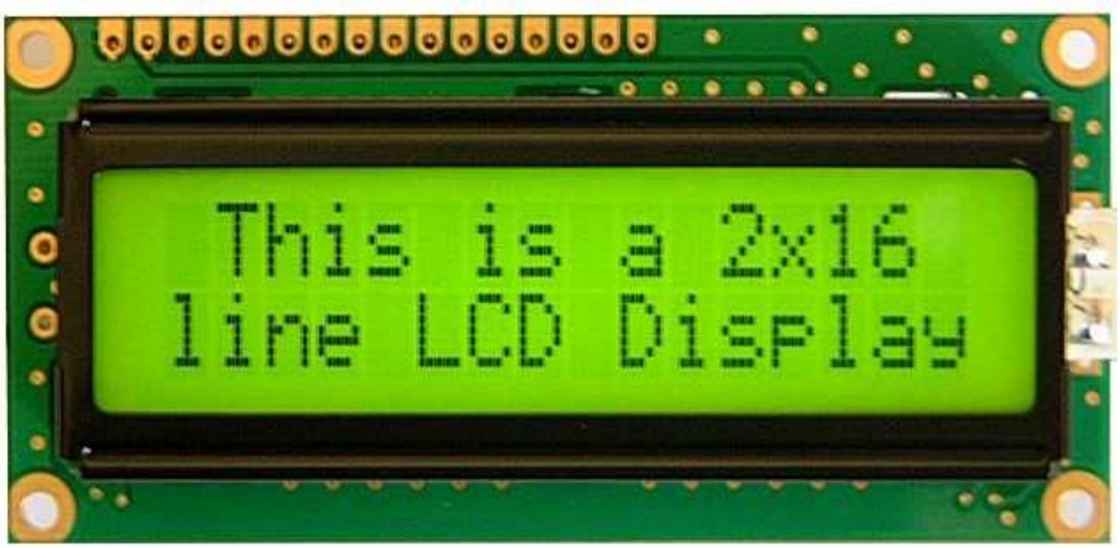


Fig: 16x2 LCD display

Pin 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

Table: Pin Description of LCD

Commands and Instruction set

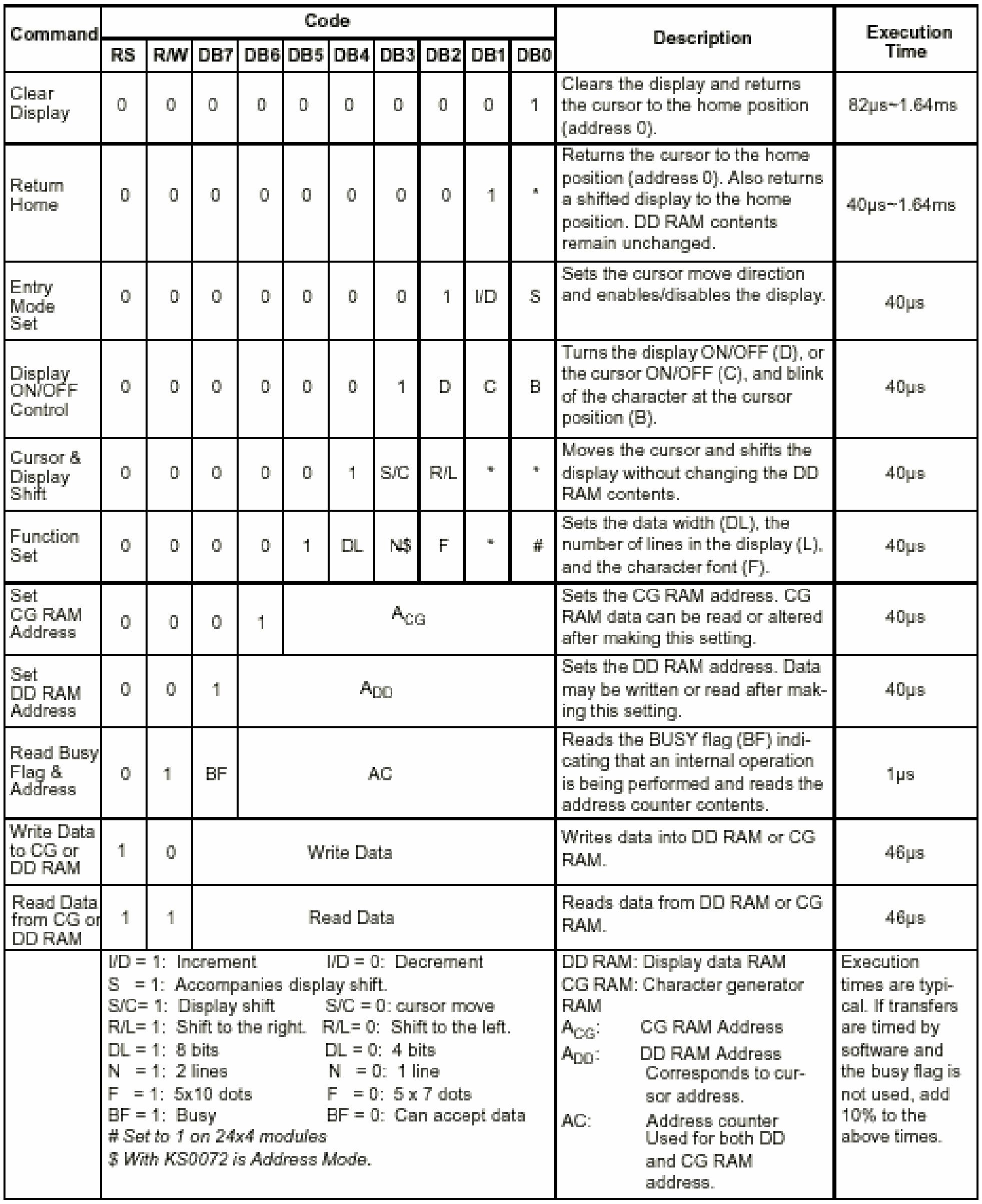
Only the instruction register (IR) and the data register (DR) of the LCD can be controlled by the MCU. Before starting the internal operation of the LCD, control information is temporarily stored into these registers to allow interfacing with various MCUs, which operate at different speeds, or various 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.



Command	Code										Description	Execution Time
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Clear Display	0	0	0	0	0	0	0	0	0	1	Clears the display and returns the cursor to the home position (address 0).	82μs~1.64ms
Return Home	0	0	0	0	0	0	0	0	1	*	Returns the cursor to the home position (address 0). Also returns a shifted display to the home position. DD RAM contents remain unchanged.	40μs~1.64ms
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S	Sets the cursor move direction and enables/disables the display.	40μs
Display ON/OFF Control	0	0	0	0	0	0	1	D	C	B	Turns the display ON/OFF (D), or the cursor ON/OFF (C), and blink of the character at the cursor position (B).	40μs
Cursor & Display Shift	0	0	0	0	0	1	S/C	R/L	*	*	Moves the cursor and shifts the display without changing the DD RAM contents.	40μs
Function Set	0	0	0	0	1	DL	N$	F	*	#	Sets the data width (DL), the number of lines in the display (L), and the character font (F).	40μs
Set CG RAM Address	0	0	0	1	A_{CG}						Sets the CG RAM address. CG RAM data can be read or altered after making this setting.	40μs
Set DD RAM Address	0	0	1	A_{DD}							Sets the DD RAM address. Data may be written or read after making this setting.	40μs
Read Busy Flag & Address	0	1	BF	AC							Reads the BUSY flag (BF) indicating that an internal operation is being performed and reads the address counter contents.	1μs
Write Data to CG or DD RAM	1	0	Write Data								Writes data into DD RAM or CG RAM.	46μs
Read Data from CG or DD RAM	1	1	Read Data								Reads data from DD RAM or CG RAM.	46μs
	I/D = 1: Increment I/D = 0: Decrement S = 1: Accompanies display shift. S/C= 1: Display shift S/C = 0: cursor move R/L= 1: Shift to the right. R/L= 0: Shift to the left. DL = 1: 8 bits DL = 0: 4 bits N = 1: 2 lines N = 0: 1 line F = 1: 5x10 dots F = 0: 5 x 7 dots BF = 1: Busy BF = 0: Can accept data *# Set to 1 on 24x4 modules* *$ With KS0072 is Address Mode.*										DD RAM: Display data RAM CG RAM: Character generator RAM A_{CG}: CG RAM Address A_{DD}: DD RAM Address Corresponds to cursor address. AC: Address counter Used for both DD and CG RAM address.	Execution times are typical. If transfers are timed by software and the busy flag is not used, add 10% to the above times.

Table: Showing various LCD Command Description

List of Commands

No.	Instruction	Hex	Decimal
1	Function Set: 8-bit, 1 Line, 5x7 Dots	0x30	48
2	Function Set: 8-bit, 2 Line, 5x7 Dots	0x38	56
3	Function Set: 4-bit, 1 Line, 5x7 Dots	0x20	32
4	Function Set: 4-bit, 2 Line, 5x7 Dots	0x28	40
5	Entry Mode	0x06	6
6	Display off Cursor off (clearing display without clearing DDRAM content)	0x08	8
7	Display on Cursor on	0x0E	14
8	Display on Cursor off	0x0C	12
9	Display on Cursor blinking	0x0F	15
10	Shift entire display left	0x18	24
12	Shift entire display right	0x1C	30
13	Move cursor left by one character	0x10	16
14	Move cursor right by one character	0x14	20
15	Clear Display (also clear DDRAM content)	0x01	1
16	Set DDRAM address or coursor position on display	0x80+add*	128+add*
17	Set CGRAM address or set pointer to CGRAM location	0x40+add**	64+add**

Table: Frequently Used Commands and Instructions forLCD

Liquid crystal displays interfacing with Controller

The LCD standard requires 3 control lines and 8 I/O lines for the data bus.

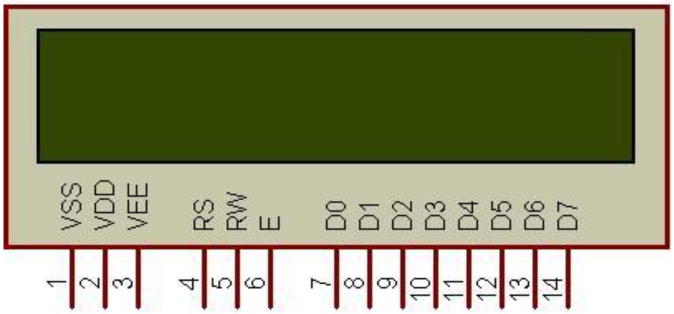


Fig: Pins of LCD

- **8 data pins D7:D0**

Bi-directional data/command pins

Alphanumeric characters are sent in ASCII format.

- **RS: Register Select**

RS = 0 -> Command Register is selected

RS = 1 -> Data Register is selected

- **R/W: Read or Write**

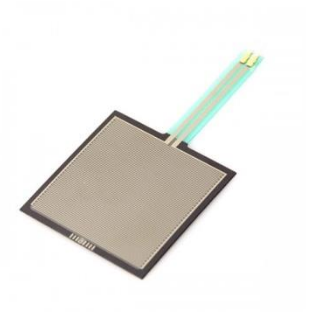
0 -> Write,

1 -> Read

- **E: Enable (Latch data)**

Used to latch the data present on the data pins. A high-to-low edge is needed to latch the data.

5. PRESSURE SENSOR



A pressure sensor is a device designed to measure the pressure of gases or liquids. Pressure is an essential parameter in various applications, and these sensors convert the pressure data into an electrical signal that can be read and processed. The basic working principle involves the deformation of a diaphragm or other sensing element when pressure is applied. This deformation is then converted into an electrical signal through various methods, such as piezoelectric, capacitive, or resistive effects.

Pressure sensors are widely used in industrial, automotive, and consumer applications. In industrial settings, they monitor and control processes involving fluids and gases, ensuring safety and efficiency. In automotive applications, they measure tire pressure, engine oil pressure, and fuel pressure to maintain optimal performance and safety. Consumer electronics, such as smartphones and wearable devices, also utilize pressure sensors for functions like altitude measurement and touch sensitivity.

The accuracy, range, and sensitivity of pressure sensors can vary based on their design and the materials used. Advances in microelectromechanical systems (MEMS) technology have led to the development of miniature pressure sensors that are highly accurate and reliable, facilitating their integration into compact and complex systems. These advancements continue to expand the applications and capabilities of pressure sensors in various fields.

6. WIFI

The **ESP8266** is a low-cost Wi-Fi chip with full TCP/IP stack and MCU (microcontroller unit) capability produced by Shanghai-based Chinese manufacturer, Espressif Systems.

The chip first came to the attention of western makers 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-style commands. However, at the time there was almost no English-language documentation on the chip and the commands it accepted.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.

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: Tensilica Xtensa 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. THING SPEAK

ThingSpeak is an open-source Internet of Things (IoT) platform that enables users to collect, analyze, and visualize data from sensors and devices connected to the internet. It is primarily used for building IoT applications, offering easy integration with hardware devices such as microcontrollers (e.g., Arduino, ESP8266, Raspberry Pi) and sensors (e.g., temperature, humidity, gas sensors). ThingSpeak provides an intuitive interface for real-time data logging and processing, making it ideal for projects like remote monitoring and automated systems.

Key features of ThingSpeak include:

1. **Data Collection:** ThingSpeak allows users to collect sensor data through HTTP requests or MQTT. It stores this data in "channels," each containing fields for different types of information (e.g., temperature, gas levels, humidity).

2. **Data Analysis:** ThingSpeak offers built-in tools for analyzing data in real-time, including MATLAB integration. Users can write custom scripts to perform operations on incoming data, like data smoothing or applying thresholds.

3. **Visualization:** It offers various types of data visualization, including graphs, charts, and maps, to help users interpret their data more easily.

4. **Alerts & Notifications:** ThingSpeak supports real-time alerts based on predefined conditions, such as sensor values exceeding a threshold. It can trigger notifications via email, SMS, or external services like Slack.

5. **Integration:** ThingSpeak supports API integration, allowing users to interface with other platforms or control devices remotely. It also integrates with various cloud services like Google Sheets, allowing data export for further analysis.

6. **Security:** ThingSpeak ensures secure communication via encrypted connections and offers access control features for managing user permissions on a channel.

7. **Free Tier:** ThingSpeak provides a free tier that allows limited but sufficient functionality for small-scale IoT projects, while also offering premium plans for advanced features.

Due to its simplicity, flexibility, and rich set of features, ThingSpeak is widely used for IoT projects such as home automation, environmental monitoring, smart agriculture, and industrial applications.

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

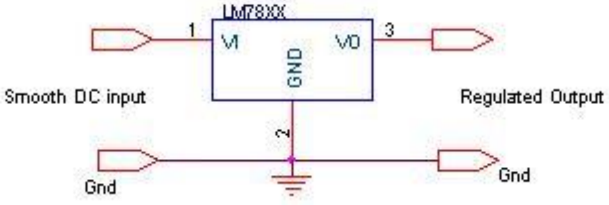
1) Linear Voltage Regulator

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.

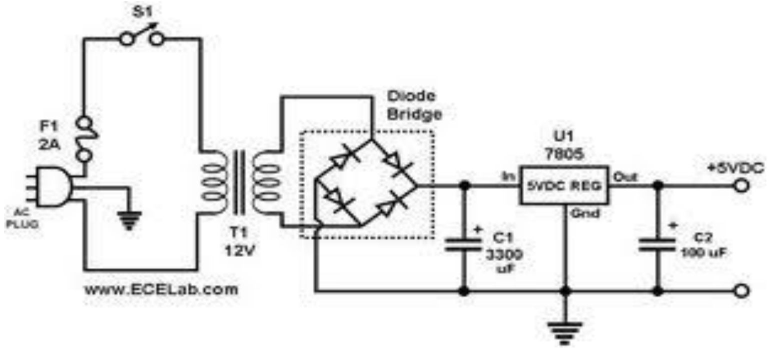
The most commonly available Linear Positive Voltage Regulators are the 78XX series where the XX indicates the output voltage. And 79XX series is for Negative Voltage Regulators.



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**
V_{out}	5V
V_{ein} - V_{out} Difference	5V - 20V
Operation Ambient Temp	0 - 125°C
Output I_{max}	1A

CHAPTER 6

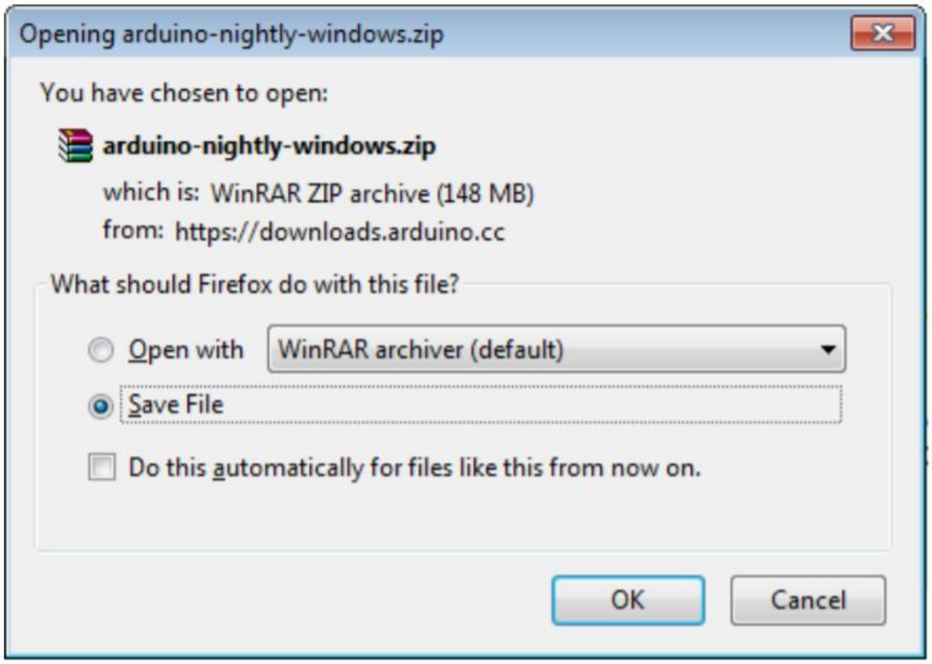
SOFTWARE TOOLS

1. Arduino ide
2. Embedded c programming

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.



Opening arduino-nightly-windows.zip

You have chosen to open:

arduino-nightly-windows.zip

which is: WinRAR ZIP archive (148 MB)

from: https://downloads.arduino.cc

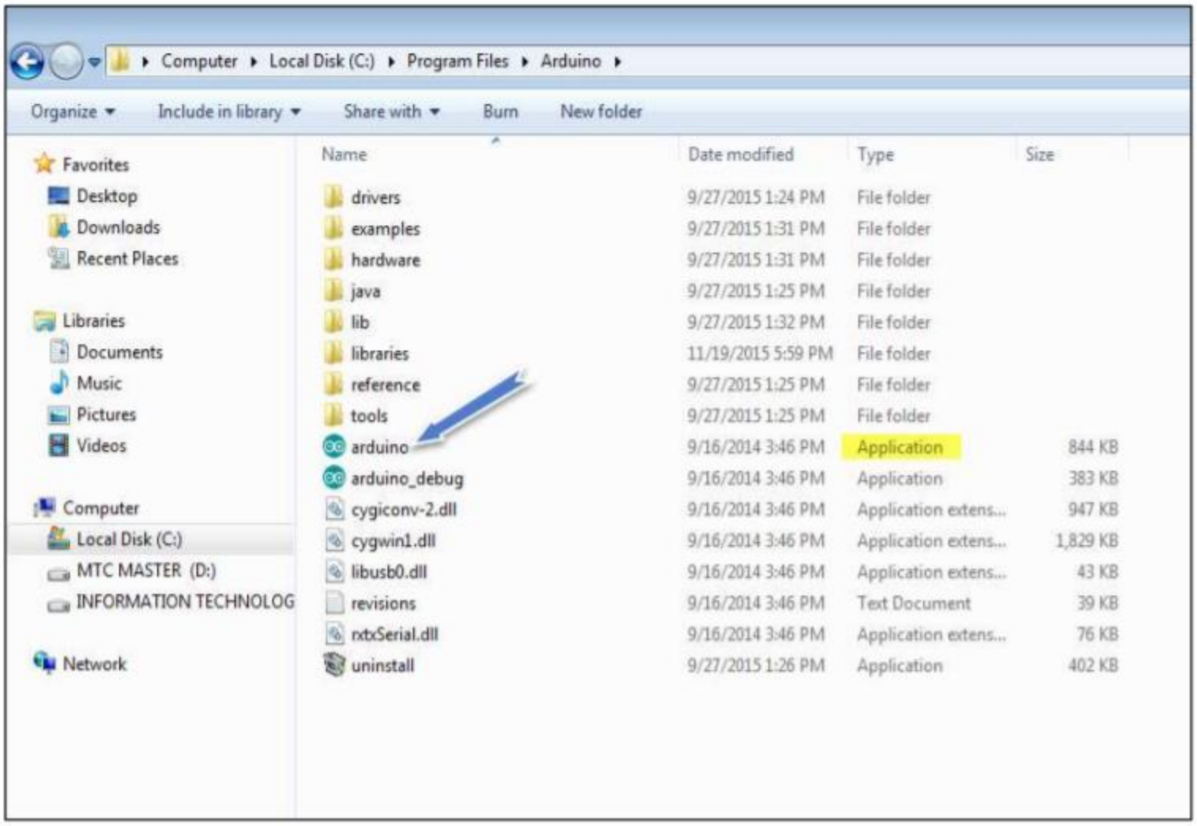
What should Firefox do with this file?

○ Open with WinRAR archiver (default)

● Save File

□ Do this automatically for files like this from now on.

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). Double click the icon to start the IDE.



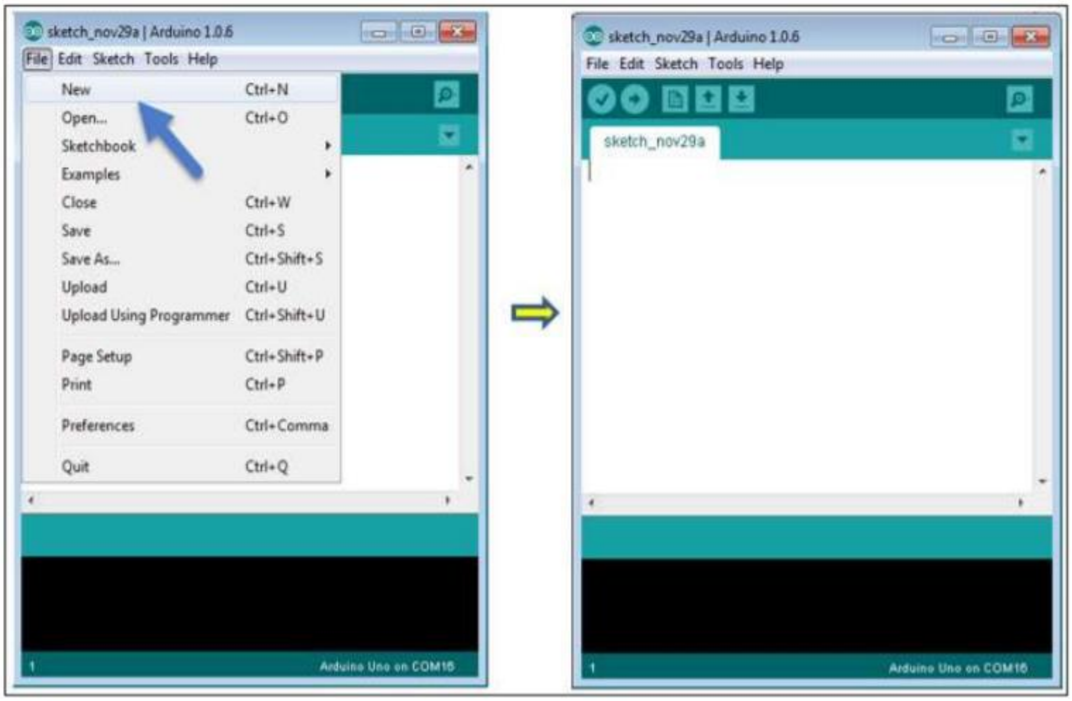
Name	Date modified	Type	Size
drivers	9/27/2015 1:24 PM	File folder	
examples	9/27/2015 1:31 PM	File folder	
hardware	9/27/2015 1:31 PM	File folder	
java	9/27/2015 1:25 PM	File folder	
lib	9/27/2015 1:32 PM	File folder	
libraries	11/19/2015 5:59 PM	File folder	
reference	9/27/2015 1:25 PM	File folder	
tools	9/27/2015 1:25 PM	File folder	
arduino	9/16/2014 3:46 PM	Application	844 KB
arduino_debug	9/16/2014 3:46 PM	Application	383 KB
cygiconv-2.dll	9/16/2014 3:46 PM	Application extens...	947 KB
cygwin1.dll	9/16/2014 3:46 PM	Application extens...	1,829 KB
libusb0.dll	9/16/2014 3:46 PM	Application extens...	43 KB
revisions	9/16/2014 3:46 PM	Text Document	39 KB
rxtxSerial.dll	9/16/2014 3:46 PM	Application extens...	76 KB
uninstall	9/27/2015 1:26 PM	Application	402 KB

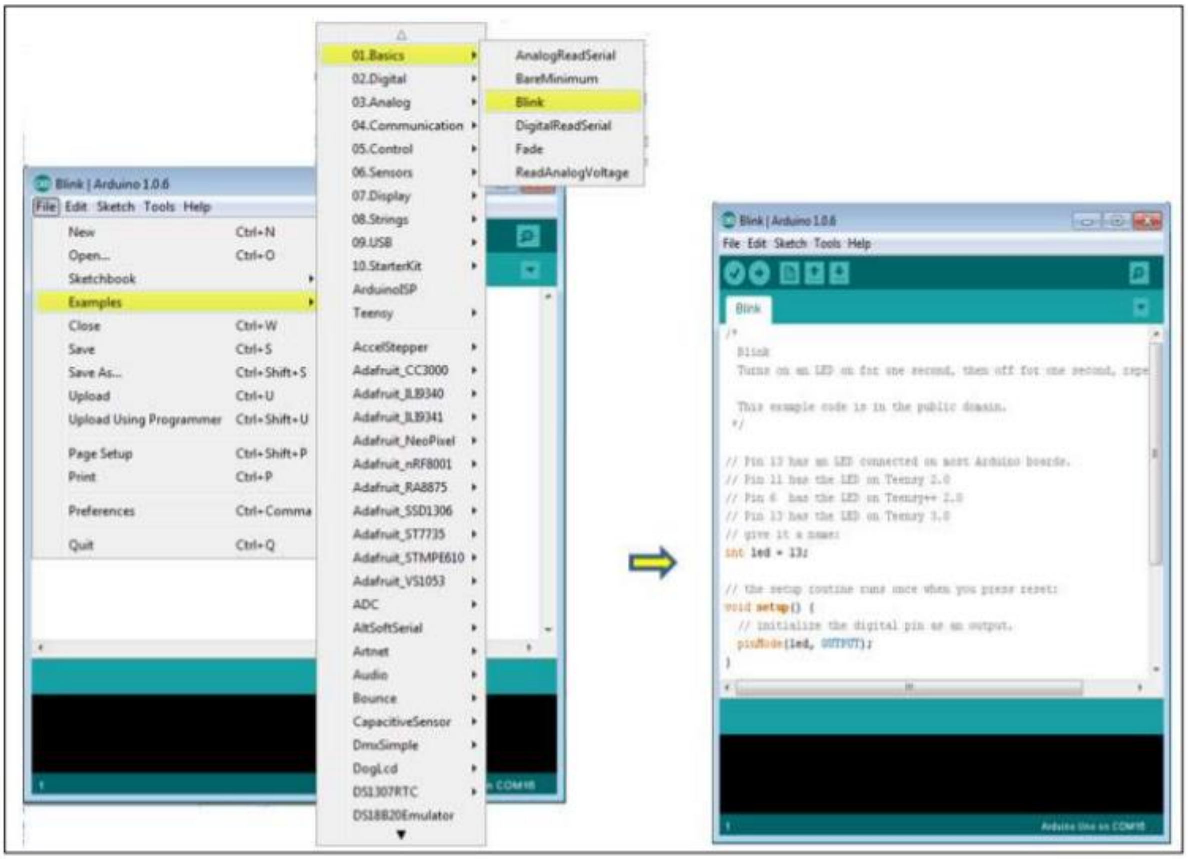
Open your first project. Once the software starts, you have two options:

- Create a new project.

- Open an existing project example.

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

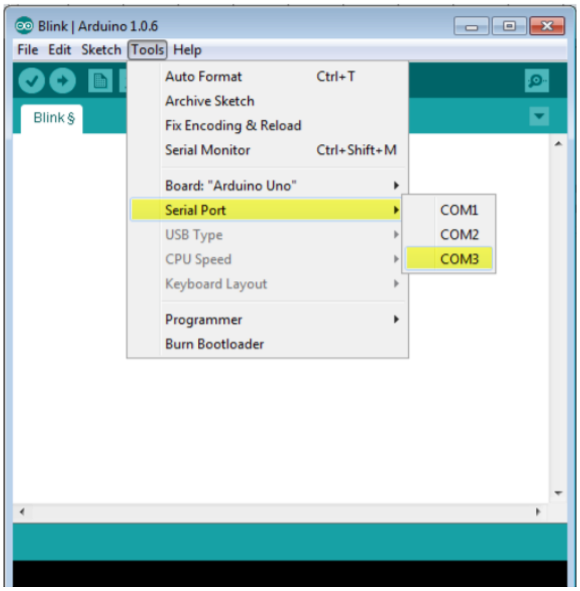




```
/*
  Blink
  Turns on an LED on for one second, then off for one second, repe

  This example code is in the public domain.
 */

// Pin 13 has an LED connected on most Arduino boards.
// Pin 11 has the LED on Teensy 2.0
// Pin 6  has the LED on Teensy++ 2.0
// Pin 13 has the LED on Teensy 3.0
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}
```

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.

A- Used to check if there is any compilation error.

B- Used to upload a program to the Arduino board.

C- Shortcut used to create a new sketch.

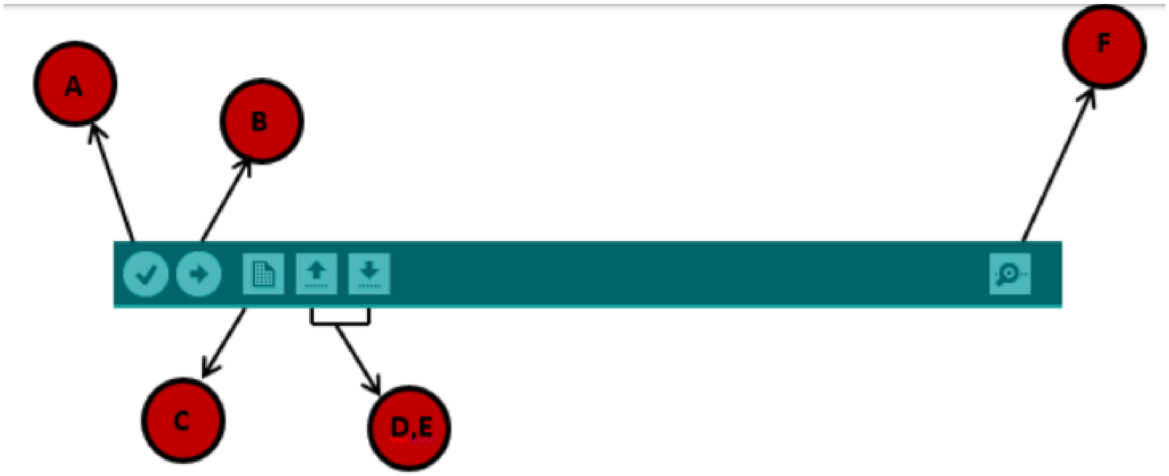
D- Used to directly open one of the example sketch.

E- Used to save your sketch.

F- 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.

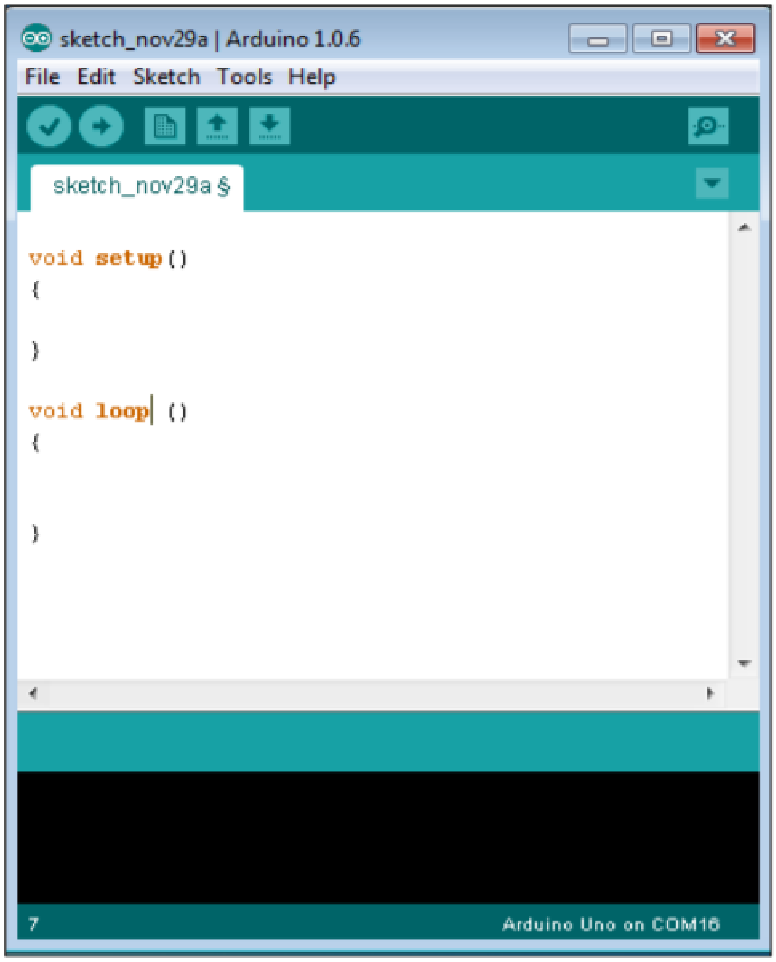
Wait a few seconds; you will see the RX and TX LEDs on the board, flashing. If the upload is successful, the message "Done uploading" will appear in the status bar.



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

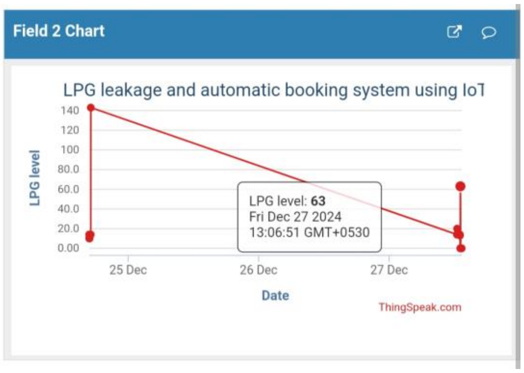


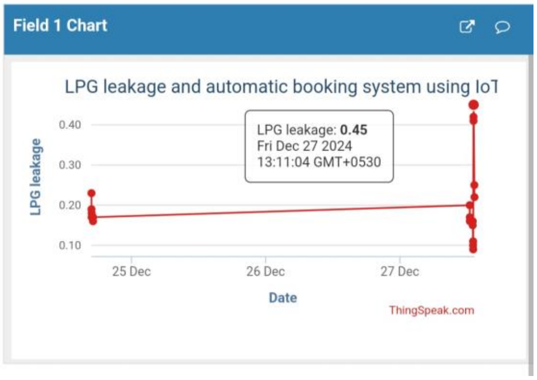
```
void setup()
{

}

void loop ()
{


}
```

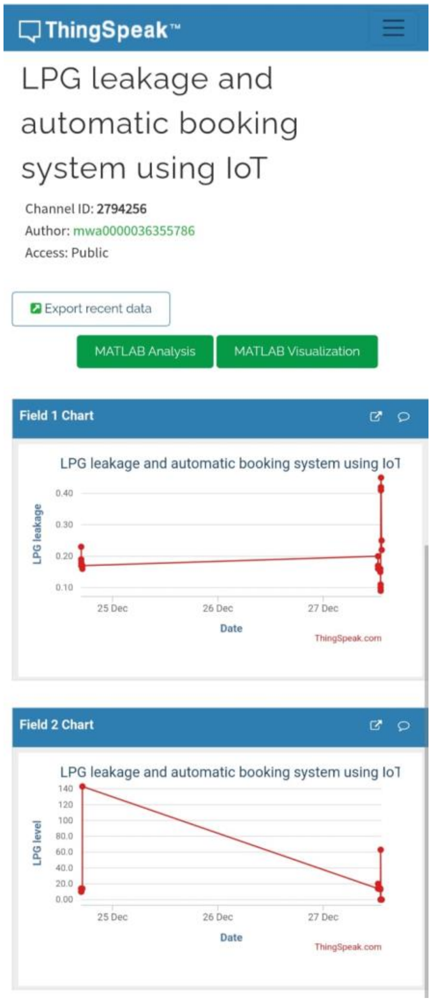
Data types in C refers to an extensive system used for declaring variables or functions of different types. The type of a variable determines how much space it occupies in the storage and how the bit pattern stored is interpreted. The following table provides all the data types that you will use during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

CHAPTER 7

RESULTS

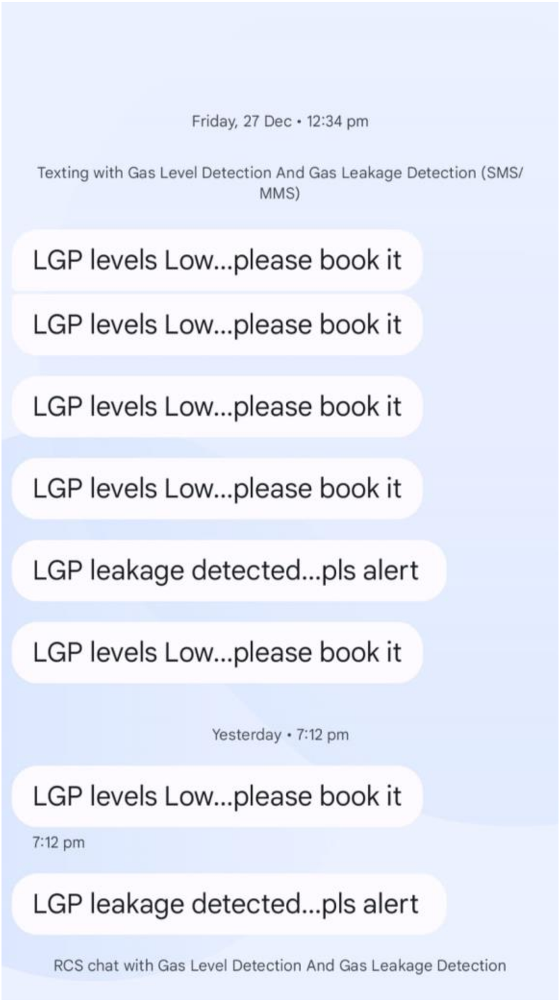






LPG leakage and automatic booking system using IoT

Channel ID: 2794256
Author: mwa0000036355786
Access: Public



Friday, 27 Dec • 12:34 pm

Texting with Gas Level Detection And Gas Leakage Detection (SMS/MMS)

LGP levels Low...please book it

LGP levels Low...please book it

LGP levels Low...please book it

LGP levels Low...please book it

LGP leakage detected...pls alert

LGP levels Low...please book it

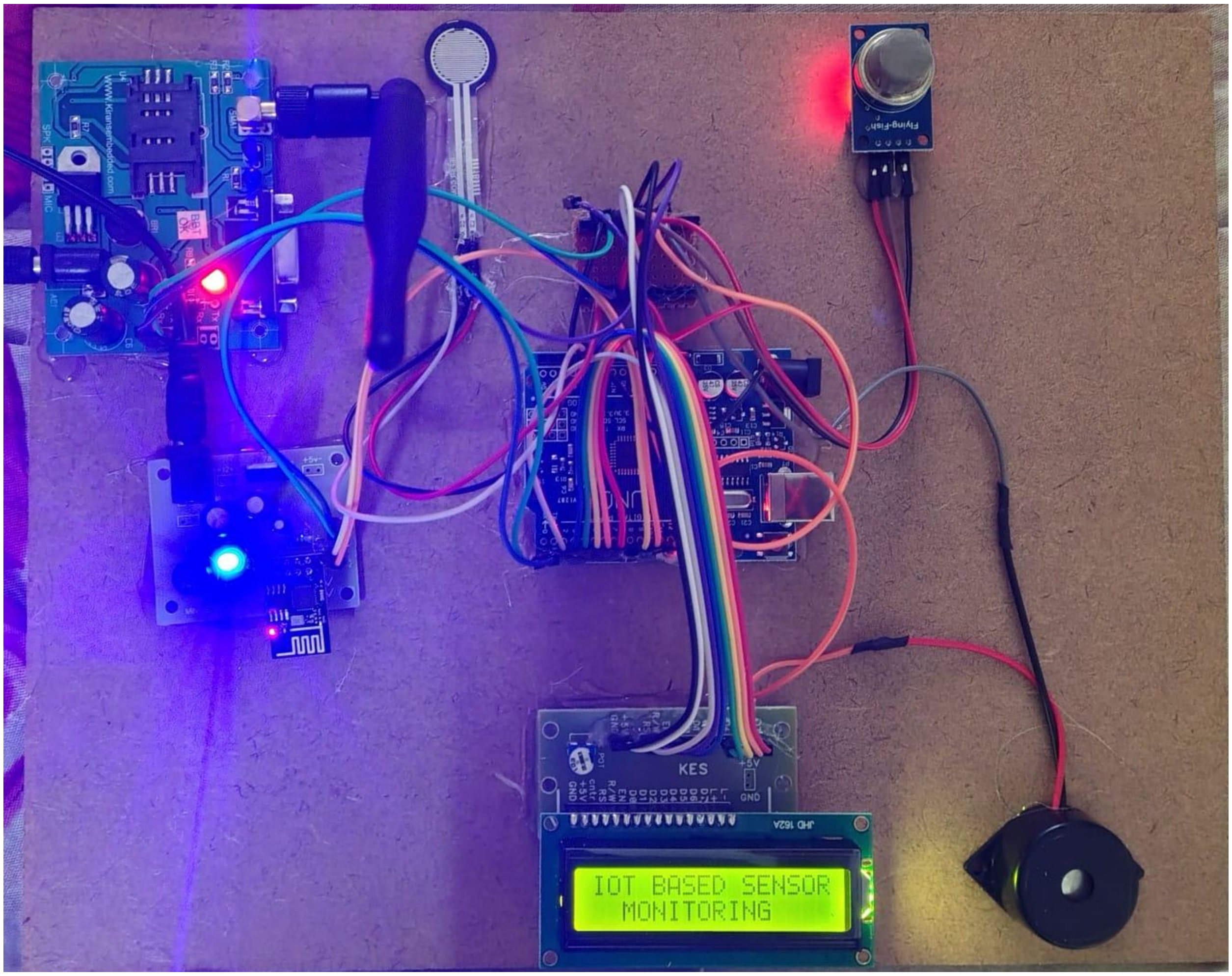
Yesterday • 7:12 pm

LGP levels Low...please book it

7:12 pm

LGP leakage detected...pls alert

RCS chat with Gas Level Detection And Gas Leakage Detection


IOT BASED SENSOR
MONITORING
KES
JHD 162A
Flying-Fish
www.Kiransembedded.com

CHAPTER 8

CONCLUSION

The IoT-based LPG Leakage and Level Detection System with auto-booking provides an innovative, efficient, and user-friendly solution for managing LPG cylinders. By leveraging advanced sensors and IoT technology, the system ensures real-time monitoring of gas levels and leakage detection, enhancing safety and convenience for users. The integration with the ThingSpeak platform enables seamless automation of refill bookings, reducing manual effort and ensuring uninterrupted LPG supply. Additionally, the buzzer alert and SMS notification system provide timely warnings in case of gas leakage, minimizing risks associated with accidents. This comprehensive approach not only addresses the safety concerns of LPG usage but also simplifies cylinder management, making it an essential tool for households and industries alike. The proposed system demonstrates the significant potential of IoT in improving everyday life.

FUTURE SCOPE

The system can be enhanced with AI-driven analytics for predicting gas consumption trends and optimizing refill schedules. Mobile app integration can provide real-time monitoring and control, improving user interaction. Expanding its application to industrial and commercial LPG setups can extend its benefits to broader domains. Incorporating solar-powered modules will enhance sustainability, while integration with smart home systems can improve overall automation and safety, making the system more versatile and future-ready.

REFERENCES

[1] Kumar Keshamoni, Sabbani Hemanth, “Smart Gas Level Monitoring, Booking and Gas Leakage Detectorover IoT”, IEEE 7th International Advance Computing Conference,2017.

[2] Chaitali Bagwe, Vidya Ghadi, Vinayshri Naik, Neha Kunte, “IOT Based Gas Leakage Detection Systemwith Database Logging, Prediction and Smart Alerting- Review”,March 2018.

[3] B. D. Jolhe, P. A. Potdukhe, N. S. Gawai, “Automatic LPG Booking, Leakage Detection And Real TimeGas Measurement Monitoring System”, International Journal of Engineering Research & Technology (IJERT) ,April 4, 2013.

[4] R.Naresh Naik, P.Siva Nagendra Reddy, S.Nanda Kishore, K.Tharun Kumar Reddy. “Arduino Based LPGgas Monitoring & Automatic Cylinder booking with Alert System”, IOSR Journal of Electronics and Communication Engineering (IOSR-JECE), August 2016.

[5] S.Rajitha, T.SWAPNA, “A Security Alert System Using Gsm For Gas Leakage”, International Journal OfVLSI And Embedded Systems,2012.

[6] Shruthi Unnikrishnan, Mohammed Razil, Joshua Benny, Shelvin Varghese and C.V. Hari,“LPGMonitoring and Leakage Detection System”, IEEE WiSPNET, 2017.

[7] Arun Raj, Athira Viswanathan, “LPG gas monitoring system”, INTERNATIONAL JOURNAL OFINNOVATIVE TECHNOLOGY AND RESEARCH, Volume No.3, 2015.