*Project Work Report On*

### “WIRELESS NOTICE BOARD WITH SPEAKER USING GSM AND ARDUINO”

done by

Chinmaya D R

Under the guidance of

**Prof. Prathima A** Assistant Professor



**Department of Electronics & Communication Engineering VEMANA INSTITUTE OF TECHNOLOGY**

**(Affiliated to VTU & Approved by AICTE)**

the prediction accuracy of battery state-of-charge (SoC), state-of-health (SoH), and energy demand patterns. These predictive insights enable real-time decision-making for efficient energy storage and distribution, prolong battery lifespan, and reduce energy losses. The system also adapts to changing solar irradiance and load conditions, making it robust and scalable for various solar energy applications. Simulation results demonstrate improved performance over conventional methods, validating the effectiveness of ML in energy management. The study highlights the potential of intelligent BMS to advance sustainable energy solutions by ensuring smarter integration of solar power systems.

|  |  |  |
| --- | --- | --- |
| **Chapter** |  | **Page** |
| **No.** | **Title** | **No.** |
|  | Abstract | i |
|  | Acknowledgement  Table of Contents List of Figures | ii iii- iv  v |
| **1** | **INTRODUCTION** | **1-3** |
|  | 1.1 [Scope](#_bookmark0) | [**2**](#_bookmark0) |
|  | 1.2 [Importance](#_bookmark1) | [**2**](#_bookmark1) |
|  | 1.3 [Objectives](#_bookmark2) | [**3**](#_bookmark2) |
| **2** | **LITERATURE SURVEY** | **4-5** |
| **3** | **DESIGN METHODOLOGY** | **6-23** |
|  | 3.1 [Aim of the Project](#_bookmark3) | [**7**](#_bookmark3) |
|  | 3.2 [Integration](#_bookmark4) | [**7**](#_bookmark4) |
|  | 3.2.1 [Workflow](#_bookmark5) | [**8**](#_bookmark5) |
|  | 3.2.2 [Circuit Integration](#_bookmark6) | [**8**](#_bookmark6) |
|  | 3.3 [Hardware and Software Requirements](#_bookmark7) | [**8-9**](#_bookmark7) |
|  | 3.3.1 [Hardware Requirements](#_bookmark8) | [**9-21**](#_bookmark8) |
|  | 3.3.2 Software Requirements | **22-23** |
| **4** | **DESIGN IMPLEMENTATION** | **24-27** |
|  | 4.1 [Hardware Implementation](#_bookmark9) | [**24**](#_bookmark9) |
|  | 4.2 [Data Processing and Feature Extraction](#_bookmark10) | [**25**](#_bookmark10) |
|  | 4.3 [Machine Learning Implementation](#_bookmark11) | [**26-27**](#_bookmark11) |
| **5** | **Results** | **28-30** |
|  | [**CONCLUSION**](#_bookmark12) | [**31**](#_bookmark12) |
|  | [**REFERENCES**](#_bookmark13) | [**32**](#_bookmark13) |

|  |  |  |
| --- | --- | --- |
| **Figure**  **no.** | **Title** | **Page**  **no.** |
| 3.1 | ESP32 Board Highlights | 9 |
| 3.2 | ESP32 Pinout Diagram | 11 |
| 3.3 | GSM MODULE (SIM800L) | 13 |
| 3.4 | LCD Display | 16 |
| 3.5 | Speaker | 17 |
| 3.6 | Breadboard | 19 |
| 3.7 | Jumper wires | 19 |
| 3.8 | Antenna | 20 |
| 3.9 | DFPlayer | 21 |
| 4.1 | Block Diagram of Wireless Notice Board with Speaker Using GSM and Arduino. | 24 |
| 5.1 | Circuit Connections | 28 |
| 5.2 | Out in Display | 29 |
| 5.3 | Final Output | 30 |

**CHAPTER 1**

# INTRODUCTION

The "Wireless Notice Board with Speaker Using GSM and Arduino" project presents an innovative approach to modernizing traditional notice boards by integrating wireless communication and audio functionalities. This system enables authorized users to remotely update messages and broadcast audio alerts, thereby enhancing real-time information dissemination in various environments such as educational institutions, transportation hubs, and public venues.

At the core of this system is the Arduino microcontroller, which interfaces with a GSM module (e.g., SIM900A) to receive SMS messages sent from authorized mobile phones. Upon receiving a message, the Arduino processes the content and displays it on an LCD screen. The inclusion of a speaker allows for audio notifications, ensuring that critical messages capture immediate attention. This combination of visual and auditory alerts ensures that important information is effectively communicated to the intended audience.

The system's design emphasizes user authentication by configuring the GSM module to accept messages only from predefined numbers, thereby enhancing security and preventing unauthorized access. Additionally, the use of specific prefixes and suffixes in the SMS content allows the Arduino to accurately parse and display the intended message.

This project offers a efficient solution for dynamic information sharing, reducing reliance on manual updates and printed materials. Its scalability allows for adaptation to larger displays or integration with other communication modules as needed. By leveraging GSM technology and Arduino's versatility, this wireless notice board system significantly enhances the way information is communicated in various settings.

This project demonstrates the practical application of combining GSM technology with microcontroller-based systems to create a dynamic, real-time information display solution. By enabling remote updates and incorporating audio announcements, the wireless noticeboard enhances communication efficiency, making it an ideal tool for various environment.

* 1. **SCOPE**
     + Enable wireless updates to the notice board through GSM, allowing authorized users to send messages via SMS.
     + Display received messages instantly on an LED or LCD screen.
     + Convert text messages into audio output using a speaker, ensuring accessibility for a broader audience.
     + Design the system to be easily expandable for larger displays or additional features such as multiple language support.
  2. **IMPORTANCE**

The importances of your wireless noticeboard with a speaker using GSM and Arduino, presented in points:

* + - **Real-Time Communication:** Enables instant updates to the noticeboard from any location via GSM, ensuring timely information dissemination.
    - **Remote Accessibility:** Allows authorized users to update the noticeboard without being physically present, enhancing convenience and flexibility.
    - **Enhanced Accessibility:** The integrated speaker provides audio announcements, making information accessible to individuals with visual impairments or those not directly facing the noticeboard.
    - **Efficiency:** Reduces the need for manual updates, saving time and effort in environments where the information changes frequently.
    - **Versatility:** Can be used in various settings, including schools, offices, hospitals, and public spaces, to broadcast important announcements effectively.
    - **Inclusivity:** Ensures that all individuals, regardless of their ability to read the display, can receive important messages through audio output.
    - **Scalability:** The system can be expanded or customized to include additional features, such as multilingual support or larger display screens, making it adaptable to different needs.
  1. **OBJECTIVES**
* **Display Messages:** Show text messages on the notice board using a GSM module to receive them.
* **Audio Announcements:** Play audio messages or announcements through a speaker
* **Remote Control:** Allow sending and updating messages remotely via SMS or other wireless communication.
* **Ease of Use:** Provide a user-friendly interface for easy operation and message management.

**CHAPTER 2**

# LITERATURE SURVEY

#### [1.]F. Hamid and N. H. Shah, "Wireless Notice Board Based on Arduino and GSM Technology," International Journal of Engineering Sciences & Research Technology, vol. 7, no. 2, pp. 141–144, Feb. 2018.

The authors propose a system where authenticated users can send SMS messages to a GSM module connected to an Arduino Nano. The message is then displayed on an LCD screen. This approach aims to replace traditional notice boards, reducing paper usage and manual effort.

#### [2.]M. Abila Mary, B. Pavithra, R. Sangeetha, and T. C. Subbu Lakshmi, "GSM Based Wireless Notice Board Using Arduino," International Journal of Advanced Research Trends in Engineering and Technology, vol. 6, no. 4, pp. 1–4, Apr. 2019.

This paper presents a GSM-based wireless notice board system utilizing Arduino technology. It outlines the integration of components such as GSM modems and LED displays to facilitate wireless message display, aiming to enhance communication within educational institutions and organizations.

#### [3.]A. Ibrar, H. Khalid, U. Ali, U. Safdar, and I. Khan, "Wireless Electronic Notice Board Using GSM Technology," presented at MD SRIC 2019, Wah, Pakistan, Nov. 2019.

The project focuses on developing a wireless electronic notice board that receives text messages via GSM and displays them on an LED screen. The system is designed to facilitate efficient communication in institutions and public places by allowing remote updates to the noticeboard.

**[4.]M. C. Baba, J. B. Grado, J. Dellosa, and D. J. Solis, "Digital Electronic Board Using GSM Module and Database," International Journal of Research in Engineering, Science and Management, vol. 4, no. 7, pp. 186–189, July 2021.** The paper discusses the design and development of a digital electronic board that utilizes a GSM module and database to display messages. The system allows for remote message updates, aiming to improve information dissemination in various settings.

#### [5.]S. R. Patil and A. S. Patil, "Wireless Electronic Notice Board Using GSM Technology," International Journal of Advanced Research in Computer Engineering & Technology, vol. 3, no. 4, pp. 1–4, Apr. 2014.

The authors present a GSM-based wireless electronic notice board that enables authorized users to send messages via SMS, which are then displayed on an LED screen. This system aims to streamline the process of updating notices, making it more efficient and less labor-intensive.

#### [6.]P. S. Khandare and S. S. Gawande, "GSM Based Wireless Notice Board with Voice Output," International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 3, no. 5, pp. 1–4, May 2014.

The paper introduces a GSM-based wireless noticeboard system that incorporates voice output. The system allows messages sent via SMS to be displayed on an LED screen and simultaneously converted to speech, enhancing accessibility for visually impaired individuals.

#### [7.]P. S. Asutkar, P. S. Khandare, and S. S. Gawande, "GSM Based Wireless Electronic Notice Board," International Journal of Electrical, Electronics and Computer Systems, vol. 2, no. 4, pp. 1–4, Apr. 2014.

The authors describe a GSM-based wireless electronic notice board system that allows messages to be sent via SMS and displayed on an LED screen. The system is designed for use in educational institutions and other organizations to facilitate efficient communication.

#### [8.]R. K. Sharma and A. K. Sharma, "GSM Based Wireless Notice Board with Voice Output," International Journal of Engineering Research and Applications, vol. 4, no. 4, pp. 1–4, Apr. 2014.

The paper presents a GSM-based wireless notice board system that features voice output. Messages sent via SMS are displayed on an LED screen and simultaneously converted to speech, aiming to improve information dissemination in public places.

**CHAPTER 3**

# DESIGN METHODOLOGY

The design methodology for a wireless notice board using GSM and Arduino involves a systematic approach encompassing hardware integration, software development, and system implementation. The primary objective is to enable remote updating of messages on a digital display via SMS, facilitating efficient communication in various settings such as educational institutions, offices, and public spaces.

**System Design and Component Selection:** The system architecture comprises an Arduino microcontroller, a GSM module (e.g., SIM900A), a display unit (such as a 16x2 LCD or LED matrix), and a power supply. The Arduino serves as the central controller, interfacing with the GSM module to receive SMS messages and with the display unit to present the messages. A real-time clock (RTC) module can be integrated to schedule message displays, enhancing the system's functionality.

**Software Development:** Programing is conducted using the Arduino Integrated Development Environment (IDE). The software includes routines to initialize the GSM module, parse incoming SMS messages, and update the display accordingly. Additional features such as message validation, user authentication, and scheduled displays can be implemented to enhance security and usability.

**Circuit Design:** The hardware components are interconnected following a schematic that ensures proper communication and power distribution. The GSM module communicates with Arduino via serial communication, while the display unit is connected to Arduino’s digital I/O pins. The power supply is configured to meet the voltage and current requirements of all components.

**Implementation and Testing:** After assembling the hardware and uploading the software, the system undergoes rigorous testing to ensure reliable operation. Test cases include sending SMS messages to the system and verifying accurate and timely updates on the display. Any issues identified during testing are addressed through iterative debugging and refinement of both hardware connections and software code

## Aim of the Project

The primary aim of the "Wireless Notice Board Using GSM and Arduino" project is to develop an efficient, real-time communication system that allows authorized users to remotely update digital noticeboards via SMS. This system seeks to replace traditional manual notice boards, thereby reducing paper usage and the time-consuming process of physically updating information. By integrating GSM technology with Arduino microcontrollers, the project facilitates the reception and display of messages on an LCD or LED screen, ensuring timely dissemination of information in various settings such as educational institutions, offices, and public spaces. The overarching goal is to create a user- friendly, cost-effective, and eco-friendly solution that streamlines the process of information sharing and enhances operational efficiency.

## Integration

The overall system integrates as follows:

* + - **Arduino Uno**: Acts as the central microcontroller to process incoming messages and control peripherals.
    - **GSM Module (e.g., SIM900/SIM800L)**: Receives SMS messages sent from mobile phones.
    - **16x2 LCD Display**: Displays the received messages.
    - **Speaker**: Outputs audio notifications or reads out messages.
    - **Audio Module (e.g., DFPlayer Mini)**: Plays audio files stored on an SD card.
    - **Power Supply**: Provides regulated 5V DC power to the system.
    - **Connecting Wires and Breadboard**: For assembling the circuit.
    - **Amazon Lex**: A natural language understanding (NLU) service for building conversational interfaces.
    - **AWS Lambda**: Executes backend logic in response to events.
    - **Amazon DynamoDB**: A fully managed NoSQL database. Stores user sessions, preferences, or chatbot context.
    - **Twilio API (WhatsApp Integration)**: Sends and receives WhatsApp messages via webhook callbacks and REST API.

## Workflow

#### Initialization:

* + - Set up serial communication between Arduino and GSM module.
    - Initialize the LCD display.
    - Initialize the audio module.

#### Receiving SMS:

* + - The GSM module receives an SMS sent from a mobile phone.
    - Arduino reads the SMS content via serial communication.

#### Processing Message:

* + - Extract the message content from the SMS.
    - Display the message on the LCD.
    - Trigger the audio module to play a notification sound or read out the message.

#### Loop:

* + - Continuously check for new messages and repeat the process.

## Circuit Integration

#### GSM Module:

* + - Connect TX and RX pins to Arduino's digital pins via SoftwareSerial.
    - Ensure proper power supply (usually 12V for SIM900 modules).

#### LCD Display:

* + - Connect RS, E, D4-D7 pins to Arduino's digital pins.
    - Use a potentiometer to adjust contrast.

#### Audio Module:

* + - Connect TX and RX pins to Arduino's digital pins via SoftwareSerial.
    - Connect speaker to the audio module's output.
    - Ensure a separate power supply if required.

## Hardware and Software Requirements

The Wireless Notice Board with Speaker using GSM and Arduino is a compact system designed to receive, display, and announce messages wirelessly. At its core, an Arduino microcontroller manages all connected components. A GSM module like SIM800L allows

message reception via SMS, with an antenna improving signal quality. Messages are shown

on a display unit such as an LCD or OLED screen, while a DF Player Mini and speaker provide audio output by playing pre-recorded voice messages from a microSD card. A 5V power supply ensures stable operation, and breadboards with jumper wires assist in easy circuit connections during prototyping.

On the software side, the system leverages cloud services to process WhatsApp messages sent by users. Twilio API enables message exchange over WhatsApp. Incoming messages trigger AWS Lambda, which routes them to Amazon Lex for natural language understanding. Lex identifies user intent, and Lambda then performs backend actions like retrieving data from Amazon DynamoDB, a NoSQL database used for storing messages and user details. The final response is returned to the user via Twilio. This integration creates a responsive, smart notice board that offers both visual and audible communication, ideal for schools, offices, and public environments.

### Hardware Requirements

1. **ESP32 Microcontroller**

As shown in Fig3.1 ESP32 comes with an on-chip 32-bit microcontroller with integrated Wi-Fi + Bluetooth + BLE features that targets a wide range of applications. It is a series of low- power and low-cost developed by [Espressif Systems](https://en.wikipedia.org/w/index.php?title=Espressif_Systems&action=edit&redlink=1).

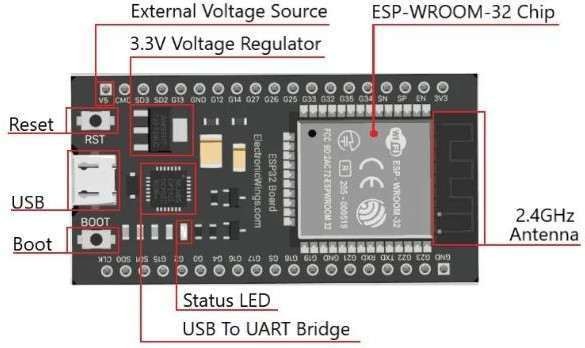


Fig 3.1 ESP32 Board Highlights

ESP-WROOM-32 development board highlights as shown in Fig 3.1 containing Tensilica Xtensa® Dual-Core 32-bit LX6 microprocessor operates at 80 to 240 MHz adjustable clock frequency. It comes with 448 KB of ROM, 520 KB of on-chip SRAM, and 4MB of Flash Memory.

####  Wireless Connectivity

* + On-chip Wi-Fi supports 802.11b/g/n standard.
  + Operates at 2.4 GHz band, up to 150 Mbps.
  + It also supports Bluetooth v4.2 BR/EDR and Bluetooth LE specifications this dual mode of Bluetooth makes it even more versatile.

####  Reset/Boot buttons

* + In ESP32 board comes with two main push buttons one is the Reset (RST/EN) button another is the BOOT button.
  + The reset button is used to reset the ESP32 Chip.
  + The use of the boot button is to enter in boot mode to upload the new sketch or program.

 **Power**

* + ESP32 development board power up from two sources one is from an external 5 Volt source and another from the USB power source.
  + The ESP32 Board operates at a 2.3 Volt to 3.6 Volt voltage source and its recommended output current is 600 mA or more.
  + Most of the ESP32 development board has an inbuilt 1117 voltage regulator which converts the 5-volt input to 3.3 volts at the output.

 **ESP32 Development Board Pinout**

The ESP32 development board has a total of 38 pinouts, the pin mapping that are as shown in Fig 3.2. These development boards come with all the needed circuitry to power and program the chip, connect it to your computer, pins to connect peripherals, built-in power and control LEDs, an antenna for wi-fi signal, and other useful features. Others even come with extra hardware like specific sensors or modules, displays, or a camera in the case of the ESP32-CAM.

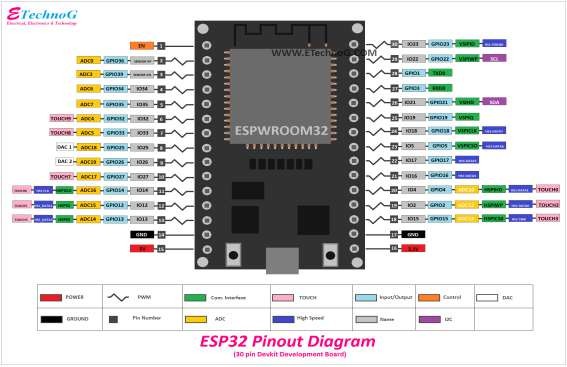


Fig 3.2 ESP32 Pinout Diagram

####  Power Pins

As shown in Fig.3.2 all boards come with power pins: 3V3, GND, and VIN. One can use these pins to power the board (if not providing power through the USB port), or to get power for other peripherals (if powering the board using the USB port).

####  General Purpose Input Output Pins (GPIOS)

Almost all GPIOs have a number assigned and that’s how one should refer to them by their number. With the ESP32 one can decide which pins are UART, [I2C,](https://randomnerdtutorials.com/esp32-i2c-communication-arduino-ide/) or [SPI](https://randomnerdtutorials.com/esp32-spi-communication-arduino/) , just need to set that on the code. It is possible due to the ESP32 chip’s multiplexing feature that allows to assign multiple functions to the same pin.

If the code isn’t set, the pins will be configured by default as shown in the figure above in Fig 3.2 (the pin location can change depending on the manufacturer). Additionally, there are pins with specific features that make them suitable or not for a particular project.

The ESP32 microcontroller is a powerful and flexible platform for building smart, connected Battery Management Systems. Its unique blend of real-time control, wireless connectivity, and low power consumption makes it an excellent choice for modern energy applications from electric vehicles and solar storage to portable IoT devices.

### SIM800L GSM Module

The **SIM800L GSM module** is a compact and cost-effective solution for adding mobile communication capabilities to projects. This versatile module supports quad-band GSM (850/900/1800/1900 MHz), making it suitable for global use. It allows an Arduino or any other microcontroller to send and receive SMS messages, make voice calls, and access the internet via GPRS for basic data applications. The SIM800L communicates with the Arduino through UART, typically using the TX and RX pins for serial communication. It operates at a voltage range of 3.7V to 4.2V, requiring a stable power supply, such as a Li- Po battery, to ensure reliable operation.

The module's small size makes it easy to integrate into projects, but it demands careful consideration of power management due to its high current consumption during transmission. The SIM800L is controlled through AT commands, which are sent over the serial interface, allowing users to perform tasks such as sending text messages, making calls, and checking network status. For better signal reception, an external antenna is connected, and for optimal performance, especially in power-hungry operations, additional capacitors may be needed to stabilize the power supply. The SIM800L is a powerful tool for IoT projects that require mobile connectivity, providing a bridge between the microcontroller and the mobile network.

#### Key Features of SIM800L:

* + **GSM/GPRS Support**: Operates on 2G GSM networks, supporting communication via SMS, voice calls, and GPRS data.
  + **Frequency Bands**: Works on 850/900/1800/1900 MHz frequency bands.
  + **Low Power Consumption**: Suitable for battery-powered applications, offering low power modes for energy-efficient operations.
  + **AT Command Interface**: Communication with the module is controlled through standard AT commands, which are sent via UART (Universal Asynchronous Receiver-Transmitter).
  + **Small Form Factor**: Compact size that is ideal for integration into space- constrained embedded systems.



Fig 3.3 GSM MODULE (SIM800L)

#### Pin Configuration:

The module has several key pins for communication and control:

* + - **VCC:** Power supply input (3.7V to 4.2V). It's crucial to provide stable power, preferably from a Li-Po battery or a well-regulated power supply.
    - **GND:** Ground.
    - **TXD:** Transmit Data Pin (connected to Arduino's RX).
    - **RXD:** Receive Data Pin (connected to Arduino's TX).
    - **RST:** Reset Pin (active low).
    - **Net light:** Network status LED (can be connected to an external LED).
      1. **DTR:** Sleep mode control (optional).
    - **RING:** Indicates an incoming call (optional).
    - **MICP/MCIN:** Microphone pins for voice input (optional if voice call is used).
    - **SPKP/SPKN:** Speaker output pins (optional if voice call is used).

#### Applications:

* **Remote Monitoring**: Allows devices to send status reports or alerts via SMS or data over the internet.
* **SMS-based Control**: Enables remote operation of devices using SMS commands.
* **Data Transmission**: Facilitates low-speed internet communication via GPRS, making it suitable for IoT and remote sensing applications.

#### Working with Arduino:

The SIM800L GSM module is commonly used with platforms like Arduino for projects involving wireless communication.

* **Communication**: It uses UART communication, with the TXD and RXD pins of the module connected to the Arduino's serial pins.
* **Power Supply**: The module requires a stable power supply, typically 3.7V to 4.2V, which may need to be regulated depending on the system's requirements.
* **Libraries**: Standard libraries like Software Serial can be used for setting up communication between the Arduino and SIM800L. A DFRobotDFPlayerMini or Adafruit\_FONA library can also be utilized for easier implementation of SMS, voice calls, and data transmission.

#### Example Applications:

* **Wireless Notice Board with Voice Alerts**: The SIM800L can receive messages via SMS and trigger a display or audio output, such as playing a voice message via an attached speaker.
* **Remote Control Systems**: Users can send SMS commands to control home appliances or other devices remotely.
* **Data Reporting System**: IoT systems can use the SIM800L to transmit sensor data to a remote server or receive updates through SMS.

### LCD Display Unit

The LCD (Liquid Crystal Display) unit used in this project is a 16x2 alphanumeric display, capable of showing 16 characters per line on two lines. It serves as the main visual interface between the system and the user, displaying messages such as text received via the GSM module, system status updates, or alert notifications. The display operates using a built-in HD44780 controller and can be interfaced with microcontrollers like Arduino either directly through digital pins or using an I2C interface module for simplified wiring. The I2C interface reduces the number of required connections, allowing communication through just two pins (SDA and SCL), and is supported by user-friendly libraries such as LiquidCrystal\_I2C.h. The backlit LCD ensures clear visibility in various lighting conditions, making it an efficient and reliable component for message display in embedded applications like wireless notice boards and remote monitoring systems.

#### Key Features of 16x2 LCD:

* + **Display Size**: 16 columns × 2 rows (16x2).
  + **Display Type**: Alphanumeric – can display letters, numbers, and symbols.
  + **Interface**: Operates in 4-bit or 8-bit mode using digital I/O pins.
  + **Backlight**: Comes with an LED backlight for better visibility.
  + **Controller**: Built-in HD44780 controller compatible with most microcontrollers.

#### I2C Interface (Optional):

* + To simplify wiring and save microcontroller pins, an I2C module can be attached to the LCD.
  + Reduces connection from 16 pins to just 4 (VCC, GND, SDA, SCL).
  + Allows easier programming using libraries like LiquidCrystal\_I2C.h.

#### Working with Arduino:

* + The LCD connects to Arduino via digital pins or I2C.
  + Libraries like or LiquidCrystal\_I2C.h are used to send data in LCD.
  + Can display dynamic content like sensor data, SMS messages, or system status.

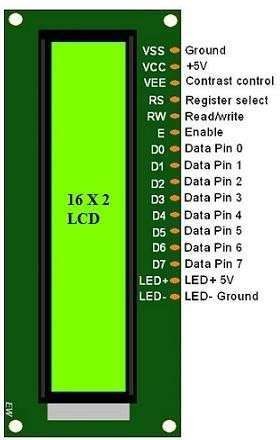


Fig 3.4 LCD Display

#### Applications in Project:

* **Message Display**: Shows messages received via GSM (e.g., notice board text).
* **System Status**: Displays operational status, error messages, or alerts.
* **User Feedback**: Provides visual feedback during operation (e.g., “Message Received”, “Playing Audio”).

In embedded applications, this LCD is typically controlled via a parallel interface, which requires multiple digital pins from a microcontroller. To reduce pin usage, an I2C adapter can be attached to the LCD, allowing communication over just two data lines (SDA and SCL). This approach not only simplifies circuit design but also frees up microcontroller pins for other tasks. With the support of readily available libraries like LiquidCrystal\_I2C.h, developers can initialize the display and write content with just a few lines of code.

For power and visibility, the module includes an LED backlight and an adjustable contrast pin (usually controlled by a potentiometer). In your project, the LCD plays a critical role in displaying the SMS content received via the SIM800L GSM module, as well as providing status messages like "Message Received" or "System Ready", giving real-time feedback to users.

### Speaker

A small speaker is used to output audio message. It connects to the DFPlayer Mini and plays sound files stored on a microSD card. A speaker is a transducer that converts electrical signals into sound, allowing audio to be heard. In electronics projects, small speakers are often used to provide voice prompts, music, or sound effects. These speakers come in various sizes and power ratings, typically measured in watts, and are selected based on the desired volume and sound quality. They are driven by an audio amplifier, which boosts the signal from a source like an MP3 player or microcontroller. Compact and versatile, speakers are essential in any project that requires audible output.

The speaker unit in this wireless noticeboard project serves as an audio output device that plays voice alerts or audio messages in response to SMS commands received via the GSM module. This adds an important layer of accessibility and functionality, especially in environments where visual attention to the noticeboard is not always possible. The speaker is typically connected through an audio playback module, such as the DFPlayer Mini, which can read pre-recorded audio files (usually in MP3 or WAV format) from a microSD card and playing them through a connected speaker.

The DFPlayer Mini is controlled by the Arduino using serial communication, often through the Software Serial library. When an SMS is received by the SIM800L GSM module, the Arduino processes the content and, if required, triggers an associated audio file to play through the speaker.



Fig 3.5 Speaker

This process allows the system to read out the notice or provide audio feedback such as alerts, welcome messages, or instructions. The volume, playback control, and audio file selection are all managed through simple serial commands sent from the Arduino to the DFPlayer Mini..

The speaker used can be a small 3W or 5W audio speaker suitable for indoor use. It is powered either directly from the DFPlayer Mini or through an external amplifier if higher volume is required. The integration of a speaker significantly enhances the usability of the system, making the notice board more interactive and inclusive, especially for users who may be visually impaired or for announcements in noisy environments.

### Power Supply

The power supply is a crucial part of the Wireless Notice Board using GSM and Arduino with Speaker system. It ensures that each component receives the appropriate voltage and current required for proper operation. As shown in the diagram, the system uses a regulated power supply to convert AC mains voltage to a stable DC voltage. The process involves several stages:

* + **Step-down Transformer**: Reduces the high-voltage AC (typically 230V) to a lower AC voltage (e.g., 12V AC).
  + **Bridge Rectifier**: Converts the AC voltage into pulsating DC voltage.
  + **Filter Capacitor**: Smooths out the pulsating DC to a more stable DC voltage.
  + **Voltage Regulator (e.g., 7805)**: Provides a fixed output voltage (such as 5V) needed for the Arduino, GSM module (via regulator or converter), LCD display, and DFPlayer Mini audio module.
  + **Output Terminals**: Distribute regulated DC power to different modules like the Arduino board, SIM800L GSM module, LCD display, and speaker.

Special care is taken for the SIM800L module, which requires a stable 3.7V–4.2V supply and may draw peak currents up to 2A during communication. This may be achieved using a Li-ion battery or a DC-DC buck converter to ensure reliable operation.

The power supply unit is the backbone of any embedded system, providing the necessary electrical energy to each component in a safe and stable manner. In this project, which involves components like an Arduino, SIM800L GSM module, LCD display, DFPlayer Mini, and a speaker, it is crucial to design a power supply that delivers the correct voltage and sufficient current to each part.

### Breadboard & Jumper Wires

The breadboard is a solderless prototyping platform used to construct and test electronic circuits quickly and efficiently. In this project, it plays a critical role in allowing temporary connections between components such as the Arduino, SIM800L GSM module, LCD, DFPlayer Mini, and power supply without the need for soldering. The breadboard consists of a grid of interconnected holes arranged in rows and columns, enabling easy insertion of components and wires.

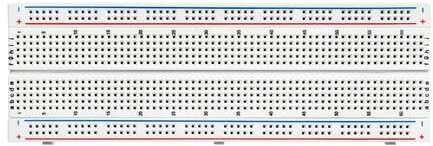


Fig 3.6 Breadboard

Jumper wires are used to establish electrical connections between components on the breadboard or between the breadboard and external modules. They come in male-to-male, male-to-female, and female-to-female types, depending on the type of pin connections required. In this project, jumper wires are used to connect signal and power lines between the Arduino board, GSM module, LCD display, audio module, and the speaker.



Fig 3.7 Jumper wires

Using breadboard and jumper wires greatly speeds up development and troubleshooting during the project’s hardware setup phase. Components can be easily rearranged or replaced without damaging the hardware, making it ideal for educational and prototype- level projects like the Wireless Notice Board using GSM and Arduino with Speaker.

### Antenna

The antenna is a vital component in the Wireless Notice Board using GSM and

arduino system, particularly for enabling wireless communication through the SIM800L GSM module. It enhances the signal reception and transmission capabilities of the GSM module, ensuring reliable connectivity to the cellular network. The SIM800L operates on GSM frequencies (850/900/1800/1900 MHz), and the antenna ensures that these signals are properly captured and transmitted.

There are various types of antennas that can be used, such as PCB antennas, spring antennas, or external SMA antennas, depending on the range and signal strength required. A properly connected antenna improves the stability of SMS reception, reduces communication errors, and increases the overall performance of the GSM-based system. In poor signal areas, a high-gain antenna is recommended to maintain consistent network access for receiving messages that will be displayed and announced by the system.



Fig 3.8 Antenna

### DFPlayer

The DFPlayer Mini is a compact and low-cost MP3 audio playback module used in this project to provide voice or audio output based on the received messages. It is capable of reading audio files (in MP3 or WAV format) from a microSD card and playing them through a connected speaker. The module has a built-in audio decoder and amplifier, allowing it to directly drive small speakers without the need for external amplification. In the wireless notice board system, the DFPlayer Mini is connected to the **Arduino** using **serial communication (TX and RX pins)**, typically through the SoftwareSerial library. The Arduino sends simple commands to the module, such as play, pause, stop, or select a specific audio file. This allows the system to **play pre-recorded voice messages** or alerts when a new SMS is received via the SIM800L GSM module.

### DFPlayer

The DFPlayer Mini is a compact and low-cost MP3 audio playback module used in this project to provide voice or audio output based on the received messages. It is capable of reading audio files (in MP3 or WAV format) from a microSD card and playing them through a connected speaker. The module has a built-in audio decoder and amplifier, allowing it to directly drive small speakers without the need for external amplification. In the wireless notice board system, the DFPlayer Mini is connected to the **Arduino** using **serial communication (TX and RX pins)**, typically through the SoftwareSerial library. The Arduino sends simple commands to the module, such as play, pause, stop, or select a specific audio file. This allows the system to **play pre-recorded voice messages** or alerts when a new SMS is received via the SIM800L GSM module.

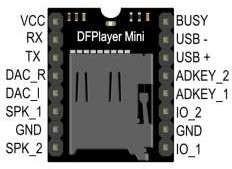


Fig3.9 DFPlayer

The module operates at 3.2V to 5V, making it compatible with most microcontrollers. It includes options for volume control, loop playback, and supports up to 32GB microSD cards formatted in FAT16 or FAT32. Its ease of use, small size, and reliable audio output make it an ideal choice for embedded systems that require sound output, enhancing both functionality and user interaction in the project.

### Software Requirements

* **AWS Lex**
  + **Purpose**: Allows users to interact with the system using natural language (text or voice).

#### Role in Project:

* + - Acts as a chatbot interface.
    - Users can type or speak commands like “Send message to board” or “Announce update”.

#### Features:

* + - Pre-trained NLP (Natural Language Processing).
    - Easily integrated with Lambda.
    - Can be deployed on web, mobile, or messaging apps like WhatsApp.
  + **Example Utterance**: *"*Display meeting at 3 PM on board".

### AWS Lambda (Serverless Backend Logic)

* + Purpose: Executes backend logic in response to triggers (e.g., Lex input or Twilio message).
  + Role in Project:
    - Parses the input message from Lex or WhatsApp.
    - Sends command to GSM via a cloud-connected module or queues SMS through an API.
  + Logs data or status to DynamoDB.
  + Features:
    - No server management needed.
    - Scales automatically.
    - Integrates seamlessly with Lex, DynamoDB, and Twilio.
  + Example Function: When Lex receives a message, Lambda formats it and initiates an SMS send process to the GSM module.

### Amazon DynamoDB (Cloud Database)

* + **Purpose**: Stores data like messages, timestamps, sender details, and status.

#### Role in Project:

* + - Maintains a history of all notice board messages.
    - Stores logs of commands and acknowledgments.

#### Features:

* + - NoSQL database – fast and scalable.
    - Works well with Lambda using AWS SDK.
  + **Example Usage**: Storing message content: {“message": "Meeting at 3 PM", "time": "2025-05-07 14:00", "status": "Sent”}

### Twilio API (WhatsApp Integration)

* + **Purpose**: Enables communication via WhatsApp using Twilio's messaging platform.

#### Role in Project:

* + - Users send messages from their phone to the notice board.
    - Twilio forwards the message to AWS Lambda for processing.
    - Lambda can send acknowledgment back through Twilio.

#### Features:

* + - Reliable and secure messaging.
    - Supports media, templates, and multi-language messages.
    - Easy integration with webhooks and serverless workflows.

**CHAPTER 4**

# DESIGN IMPLEMENTATION

The Wireless Notice Board with Speaker using GSM and Arduino is designed with a combination of embedded hardware and cloud-based software to enable remote message broadcasting with voice output. The hardware components include an Arduino UNO microcontroller that controls a 16x2 LCD for displaying text, a GSM module (such as SIM800L) for wireless message reception, and a DFPlayer Mini with a speaker for text-to- speech audio announcements. These components are powered through a regulated 5V supply and connected via standard serial and I2C communication. On the software side, the system uses AWS Lex as a chatbot interface for voice or text command input, and AWS Lambda functions for backend logic that processes user input and stores messages in Amazon DynamoDB, which logs the content and its status. Integration with Twilio’s API allows messages to be sent directly from WhatsApp, which are captured via webhook, processed through Lambda, and forwarded to the GSM module for Arduino display and announcement. This combination of local embedded hardware with scalable, serverless cloud software enables real-time, wireless, and voice-supported digital notices accessible from anywhere.

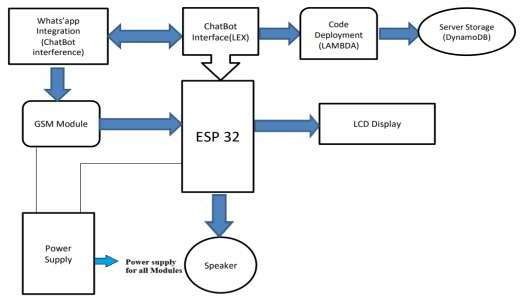


Fig 4.1 Block Diagram of Wireless Notice Board with Speaker Using GSM and Arduino.

The Fig 4.1 illustrates the overall architecture of the Wireless Notice Board with Speaker using GSM and ESP32, integrated with cloud-based chatbot services. At the core of the system is the ESP32 microcontroller, which interfaces with key components including a GSM module, an LCD display, and a speaker. The ESP32 is powered by a centralized power supply that also provides regulated voltage to all other connected modules. For user interaction, WhatsApp messages are used as input through Twilio integration, enabling communication with a chatbot built using Amazon Lex. The chatbot processes user commands, which are further handled by AWS Lambda functions for code execution. These Lambda functions manage the logic to store messages into a cloud database, specifically Amazon DynamoDB, which serves as the system's backend storage. Once a message is validated and stored, it is transmitted to the ESP32 via the GSM module. The ESP32 then displays the message on the LCD and converts the text to audio using the connected speaker module. This architecture enables real-time, remote, and voice-enabled messaging functionality using a combination of embedded and cloud technologies.

## Hardware Implementation

The hardware implementation of the Wireless Notice Board with Speaker using GSM and ESP32 is centered around the ESP32 microcontroller, which acts as the primary processing unit for interfacing and controlling various peripherals. The ESP32 is chosen for its built-in Wi-Fi and Bluetooth capabilities, as well as its compatibility with cloud integration and peripheral communication. It is connected to a GSM module (such as SIM800L), which enables wireless communication and message reception via SMS. The received message is displayed on a 16x2 LCD, which is interfaced with the ESP32 using I2C protocol for efficient data transmission and minimal pin usage.

For voice output, a DFPlayer Mini MP3 module is connected to the ESP32 and linked to a speaker, allowing messages to be converted into audio announcements. A regulated power supply is provided to ensure stable operation for all components, including the ESP32, GSM module, LCD, and audio system. The power supply module converts input voltage (e.g., 12V DC) into 5V required by the connected modules. The hardware setup ensures that each component works in sync, with the ESP32 coordinating data flow from cloud- based services to physical output devices. This design supports wireless, remote-controlled

message delivery and playback, making the system efficient, scalable, and suitable for real- time communication.

## Data Processing and Feature Extraction

In the Wireless Notice Board with Speaker using GSM and ESP32 system, data processing begins at the point of user interaction through WhatsApp or chatbot interfaces. When a user sends a message via WhatsApp or speaks to the chatbot (Amazon Lex), the input is first processed by natural language understanding (NLU) tools within Lex. Lex extracts the relevant message content by identifying key phrases or intent through pre-defined utterances and slot values.

Once the message is recognized, it is forwarded to AWS Lambda, which performs further processing such as validating the message, checking for command structure, filtering inappropriate content, and formatting it for output. This stage ensures that only valid and safe messages are passed on for display and announcement. The processed data is then stored in Amazon DynamoDB, where timestamps, sender details, and message status (e.g., pending, sent) are logged. This structured storage allows for future retrieval, message tracking, and system monitoring.

For feature extraction, the system isolates the core message text to be displayed on the LCD and prepares it for speech synthesis. The ESP32 extracts the clean text from the GSM module or directly from the cloud input (in case of Wi-Fi connectivity), and sends it to the DFPlayer Mini module. The text is matched with pre-recorded MP3 audio files or synthesized in real-time for playback through the speaker. Through this layered approach, the system ensures accurate processing, secure storage, and clear delivery of messages, both visually and audibly.

## Machine Learning Implementation

While the core functionality of the Wireless Notice Board with Speaker using GSM and ESP32 is primarily rule-based, a basic level of machine learning is implemented through the use of Amazon Lex for intelligent chatbot interaction. Amazon Lex utilizes pre-trained machine learning models for natural language understanding (NLU) to recognize user intents and extract key information from conversational input. This allows users to send

messages in a flexible and natural language format, which the system can interpret accurately without rigid command structures.

Lex uses pattern matching and intent classification to distinguish between different user queries—such as sending a new message, requesting help, or checking message status. These models improve over time based on usage data and user interactions, making the chatbot smarter and more adaptive. Additionally, optional enhancements using machine learning could include sentiment analysis on messages (using AWS Comprehend) to flag inappropriate content or prioritize urgent announcements.

Although the embedded hardware like ESP32 does not run machine learning models locally due to resource limitations, cloud-based inference handled by AWS ensures that machine learning tasks are performed efficiently and seamlessly. This lightweight integration of AI adds intelligence to the system without increasing hardware complexity.

**CHAPTER 5**

**RESULTS & DISCUSSION**

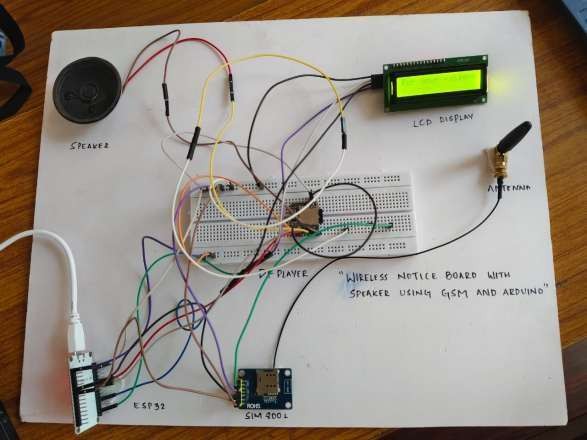
****

Fig 5.1 Experimental Setup

The project titled **"Wireless Notice Board with Speaker Using GSM and Arduino"** is a practical and innovative system designed to display messages remotely via SMS and enhance communication through audio playback. This setup integrates an ESP32 microcontroller, a SIM800L GSM module, a 16x2 LCD display, a DFPlayer Mini MP3 module, and a speaker, all mounted on a prototype board. When a user sends an SMS to the GSM module, the ESP32 reads the incoming message, processes its content, and displays the text on the LCD screen. If the message contains predefined keywords or commands, the ESP32 activates the DFPlayer Mini, which plays a corresponding audio file stored on a microSD card. The connected speaker then outputs the audio, allowing the message to be both seen and heard. This dual-mode communication—visual and auditory—makes the system highly effective for use in public places such as schools,

offices, and transport terminals, where timely and clear information dissemination is critical. The visible message in your setup, "Thank you for your patience," demonstrates the successful reception and display of a message, indicating that the system is functioning as intended.

Fig 5.2 Output in Display

The result of an LCD display in a wireless notice board using a speaker with GSM and Arduino would be a system capable of receiving messages (potentially via SMS through GSM), displaying them on an LCD screen, and possibly announcing them audibly through a speaker.

For example, as shown in Fig 5.2 if the system receives an SMS saying, "Train arriving in Platform C", it could display "Arriving in Platform C" on the LCD and announce it through the speaker. The image shows a partial message "dy in Platform C", which could be part of such an announcement.

Result: The system can display and potentially announce messages received via GSM on the LCD display and through the speaker, making it a functional wireless notice board.

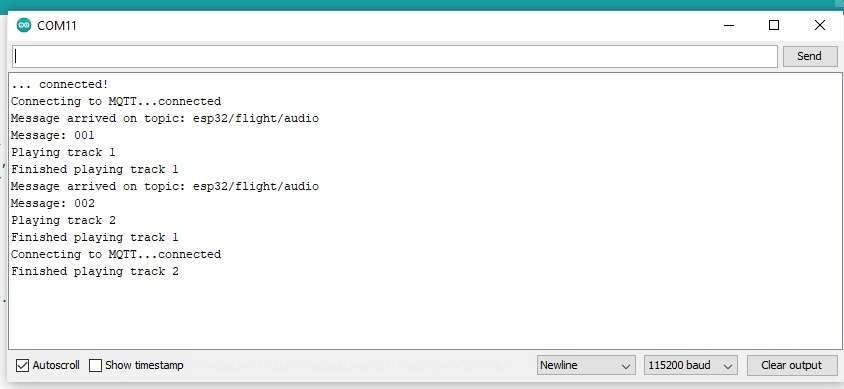


Fig 5.3 Final Output

The result shown in Fig 5.3 is a serial monitor output from an ESP32 board. The output indicates that the board is:

1. Connecting to WiFi and successfully establishing a connection.
2. Initializing a DFPlayer, which is a module used for playing audio files.
3. Connecting to an MQTT (Message Queuing Telemetry Transport) broker and receiving messages on the topic "esp32/flight/audio".
4. Playing audio tracks based on the received messages, with track numbers corresponding to the message content (e.g., "002" plays track 2, "003" plays track 3).

# CONCLUSION

The Wireless Notice Board with Speaker using GSM and Arduino is an innovative and practical solution for remotely delivering important messages with both visual and audible outputs. By integrating a GSM module, the system enables real-time message updates from any location via SMS or cloud-based platforms. The Arduino microcontroller acts as the central control unit, managing communication between the GSM module, LCD display, and audio playback system. The inclusion of a speaker, controlled through modules like the DFPlayer Mini, allows the system to convert text messages into voice, making it especially useful for visually impaired users or public announcement environments.

Furthermore, the integration of cloud services such as Amazon Lex, AWS Lambda, and DynamoDB significantly enhances the system's scalability and intelligence. These components enable chatbot-based communication, serverless execution, and structured data storage, respectively. The addition of WhatsApp integration via Twilio makes the system more accessible and user-friendly, allowing messages to be sent through commonly used communication platforms.

Overall, the project demonstrates a successful blend of embedded systems and cloud technologies to achieve an efficient, low-cost, and user-centric wireless notice board. It offers great potential for deployment in schools, offices, railway stations, hospitals, and other public places where centralized and audible message delivery is essential. The modular nature of the design also allows for future enhancements such as multilingual support, AI-driven message filtering, and scheduling features, making it a forward-compatible and impactful communication tool.

# REFERENCES

[1]. Shereen, N. Z. a nd Rozumah, B. “Mobile Phone use Amongst Student in University in Malaysia: It correlates and relationship to Psychological Health”. European Journal of Scientific Research. Vol. 37. No. 2. pp. 206 – 218, 2009.

1. Sangami.D , Nivetha.M , Rahmankhan.R , Vaitheesh.S , “A MODERN GSM BASED ANNOUNCEMENT AND STUDENT FEEDBACK SYSTEM”,

International Conference on Breakthrough in Engineering, Science & Technology– 2016 (INC-BEST’16).

1. Theodore S. Rappaport, “Wireless Communications: Principles and Practice”, Prentice Hall Communications Engineering and Emerging Technologies Series, (2002), Chapter 10, p500 - 502.
2. John Scourias, “Overview of GSM: The Global System for Mobile Communications”, University of Waterloo, March 13, 1996.
3. Hardik Gupta, Pooja Shukla, Ankita Nagwekar “GSM based LED scrolling display “. International Journal of Students Research in Technology and Management , vol (3) , may 2013 , ISBN [978-93-83006-01-4], pg 278-291.
4. Building Wireless sensor Networks using Arduino by Robert Faludi.
5. Emic 2 Text-to-Speech Module (#30016), Parallex.com
6. Emic2 Open Schematic (30016) - Parallex.com