Home Automation System: Voice Controlled Using Google Assistant

[Project Report]

Bachelor of Technology
in
Electronics and Communication Engineering

Course : - Microcontroller and Interfacing

By:
Rashi Aggrawal
Jaydevsinh Rathod
[23BEC158 and 23BEC159]

1 Intro duction

This project presents a voice-controlled home automation system designed to enhance the comfort and convenience of daily living. By integrating Google Assistant with wireless communication and microcontroller-based control, the system allows users to operate household appliances using simple voice commands. The solution emphasizes accessibility, affordability, and ease of implementation, making it a suitable option for smart home applications. The growing demand

for intelligent home systems has made automation an essential part of modern living spaces. This project addresses that need by providing a seamless interface between user input and appliance control, eliminating the reliance on physical switches. The use of voice as an input method also makes the system particularly helpful for elderly or physically challenged individuals, promoting inclusivity in technology-driven environments. Overall, the project highlights how everyday

technology can be repurposed and integrated to create meaningful, real-world solutions. It serves as a foundational model that can be further expanded to control multiple devices, monitor power usage, and even integrate with secu- rity systems, thereby laying the groundwork for a comprehensive smart home ecosystem.

Circuit Diagram

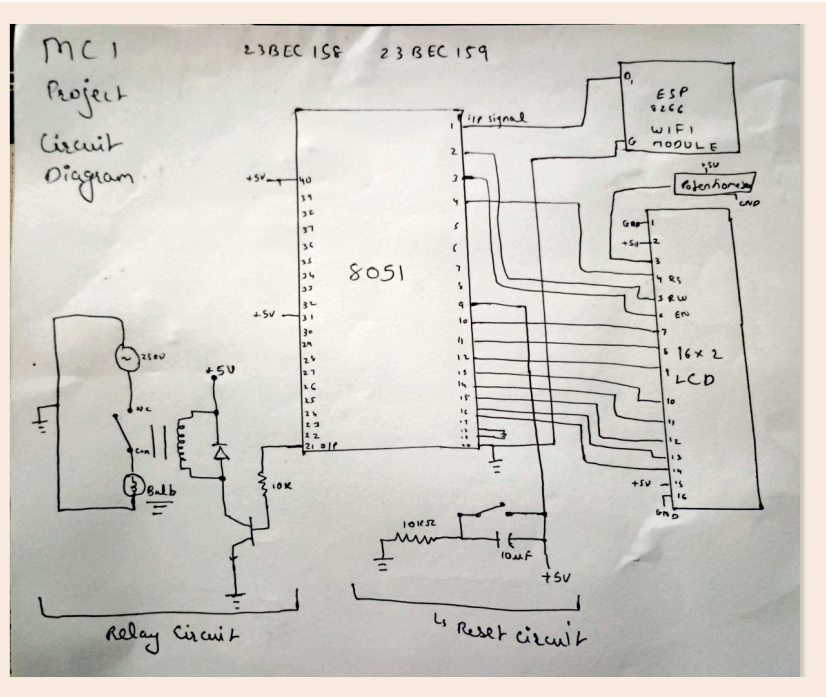
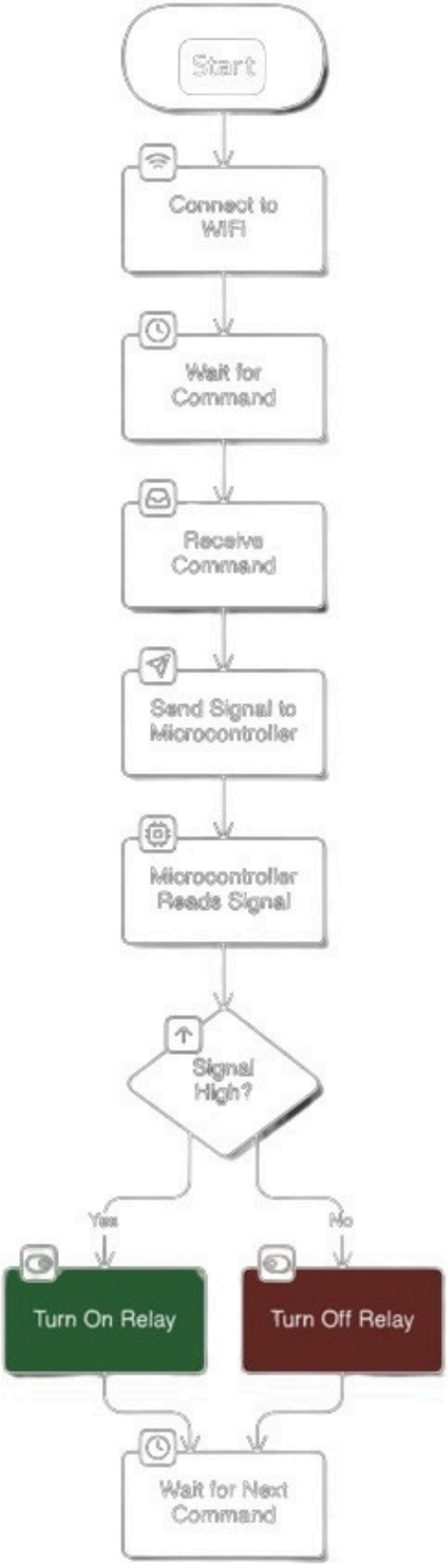


Figure 1: Circuit/Block Diagram

3 System Operation



4 Working

Use of NodeMCU (ESP8266) in the Project:

In this project, the NodeMCU (ESP8266) acts as the communication bridge between Google Assistant and the 8051 microcontroller, enabling voice-controlled automation of a home appli- ance. It handles Wi-Fi connectivity, cloud interaction, and the transmission of control signals based on voice commands. The actual switching of the appliance (a bulb) is done through a relay connected to the 8051, and the bulb's current status is displayed in real-time on an LCD screen.

1. Wi-Fi and Cloud Connectivity

The ESP8266 connects to a Wi-Fi network, enabling:

- •Communication with a cloud IoT platform (e.g., Adafruit IO)
- •Reception of commands from Google Assistant via IFTTT
- •Real-time control of the appliance over the internet

2. Voice Command Processing using Google Assistant & IFTTT

- The user issues voice commands like "Turn on the bulb" to Google Assistant.
- IFTTT interprets this and updates a virtual feed (like "bulb-control") on Adafruit IO with values such as "ON" or "OFF".
- This update triggers the automation process.

3. Data Handling via MQTT Protocol

- The ESP8266 is programmed to subscribe to the relevant feed using the MQTT protocol.
- It continuously listens for updates from Adafruit IO.
- When a new command is received (e.g., "ON"), the ESP8266 changes the logic level on one of its GPIO pins accordingly.

4. Interaction with the 8051 Microcontroller

- The digital signal from the ESP8266's GPIO is sent to an input pin of the 8051 microcontroller.
- The 8051 reads the logic level:
  - If the signal is HIGH (command = ON):
    - It activates a GPIO pin on Port 0, sending a HIGH signal to the relay module.
    - The relay closes its contacts and powers the bulb, turning it ON.
    - Simultaneously, the 8051 sends a message like "Bulb ON" to a connected 16x2 LCD.
  - If the signal is LOW (command = OFF):
    - The 8051 deactivates the relay by sending a LOW signal.
    - This disconnects power to the bulb, turning it OFF.
    - The LCD is updated to show "Bulb OFF".
- The LCD is connected to a different port of the 8051 and is interfaced using standard 4-bit or 8-bit mode, with appropriate control and data lines.

5. Optional Manual Control via Dashboard

- Apart from voice commands, the user can manually control the bulb using buttons or toggles available on the Adafruit IO dashboard, accessible from a smartphone or computer.
- These dashboard actions also update the feed and follow the same process as the voice commands.

Why NodeMCU is Crucial in This Project

- Provides seamless integration between cloud services (Adafruit IO), voice commands (Google Assistant), and hardware (8051 microcontroller)
- Acts as a real-time data handler and bridge for command execution
- Eliminates the need for complex external Wi-Fi modules or dedicated servers
- Enables a smooth, responsive, and interactive smart home experience

6 Bill of Materials (with Costing)

Sr. No.	Component Name	Quantity	TotalCost(INR)
1	NodeMCU (ESP8266) 8051	1	180
2	Microcontroller (AT89S52) Relay	1	100
3	Module (5V) 16x2 LCD Display	1	50
4	Power Supply Module (7805 or USB)	1	120
5	General Purpose PCB (GPPB board)	1	50
6	Female Header Pins Potentiometer	1	40
7	(10k for LCD) Connecting Wires /	1set	20
8	Jumper Cables 230V Bulb (Load)	1	10
9	Resistors, Capacitors, etc. Crystal	1set	30
10	Oscillator & IC Base	1	50
11		Assorted	30
12		1set	40
	Total Estimated Cost		750

7 Applications

This voice-controlled home automation system presents a meaningful step toward building smarter, more responsive living environments. By combining the capabilities of Google Assis- tant, ESP8266, and the 8051 microcontroller, the system is not only functional but also scalable and adaptable for various real-world scenarios. Below are several domains where this technology can be effectively implemented:

1. Smart Homes and Modern Living Spaces

- Enables users to operate home appliances like lights, fans, air conditioners, and coffee machines using voice commands.
- Enhances user convenience by eliminating the need for physical switches or remote controls.
- Supports integration into smart home ecosystems with routines, schedules, or triggers (e.g., turning on the lights at sunset).

2. Assistive Technology for Elderly and Disabled Individuals

- Improves accessibility for individuals with mobility, vision, or dexterity challenges.
- Allows hands-free control of lighting, fans, or emergency alarms.
- Can be extended to work with medical reminders and smart beds.

3. Energy Management and Sustainability

- Reduces energy waste by enabling remote control of appliances.
- Can be integrated with sensors for condition-based appliance control.
- Promotes eco-friendly living by enhancing energy transparency and efficiency.

4. Security and Surveillance Systems

- Enables voice-activated control of floodlights, door locks, and surveillance cameras.
- Can be linked with motion sensors for real-time security response.

5. Hospitality Industry (Hotels, Guest Houses, Airbnb)

- Enhances guest experience by enabling voice-operated room control.
- Improves hygiene and convenience by eliminating remotes and switches.
- Offers a tech-savvy edge that appeals to modern travelers.

6. Agricultural and Industrial Automation (Scalable Adaptation)

- Can control irrigation systems, lighting, and fans in farms and warehouses.
- Reduces manual effort in physically demanding environments.

7. Education and Demonstration Projects

- Ideal for students and hobbyists learning about IoT and microcontrollers.
- Demonstrates MQTT, cloud automation, and embedded interfacing.

This project not only addresses current needs for convenience and control but also opens doors to future automation possibilities. With minor modifications, it can evolve into a complete smart ecosystem adaptable to diverse sectors such as healthcare, agriculture, education, and hospitality.

8 Summary

This project successfully demonstrates a compact and cost-effective voice-controlled home automation system using Google Assistant, the ESP8266 (NodeMCU), and the 8051 microcontroller. The integration of cloud services like Adafruit IO and IFTTT enables seamless communication between voice commands and physical device control, showcasing the real power of IoT in everyday applications. By replacing traditional switches with voice commands, the system offers improved conve- nience,
especially for elderly or differently-abled individuals, while also promoting energy-efficient living. The use of a relay-controlled load and real-time feedback via an LCD adds practicality and user awareness to the setup.
This project not only fulfills academic requirements but also presents a scalable solution that can be adapted for a wide range of real-world applications — from smart homes to commercial buildings and assistive technologies. With further enhancements, the same framework can support additional appliances, sensor-based automation, and mobile dashboard integration, laying the foundation for more advanced smart environments.

5 Code

```
ORG0000H
        SJMP MAIN
; Memory Locations
        ORG0030H

; Constants
        ORG0050H MOVSP,#60H
MAIN :  ACALLLCDStart
                                ;  Initialize  stackpointer
        MOV DPTR, #welcome project
        ACALL LCD STRING
        SETB EA
        CLR P1.0
        CLR P2.0



MAIN LOOP :
        JBP1.0,BULBON           ;  Check  if  signal  atP1.0  is  high
        SJMPBULBOFF             ;  Iflow,jumptoBULBOFF

BULB ON :
        SETBP2.0                ;  Turnon  the  bulb/load  (set  P2.0  high)
        MOVDPTR,#bulbonmsg
        ACALL LCD CMD
        ACALL LCD STRING
        SJMPMAINLOOP            ;ReturntoMAINLOOP

BULB OFF :
        CLRP2.0                 ;  Turn  off  the  bulb/load  (clear  P2.0)
        MOV DPTR, #bulb off msg
        ACALL LCD CMD
        ACALL LCD STRING
        SJMPMAINLOOP            ;ReturntoMAINLOOP

; LCD Initialization
LCD Start:
        MOVP3, #00H             ; Clear data pins on Port 3
        CLRP1.3                 ;RS(Register  Select)  pin  initialized
        CLRP1.1                 ;RW(Read/Write)pininitialized
        MOV A, #38H
        ACALL LCD CMD
        MOV A, #0EH
        ACALL LCD CMD
```

```
        MOV A, #01H
        ACALL LCD CMD
        MOV A, #06H
        ACALL LCD CMD
        RET

; LCD Command
LCD CMD :
        MOV P3, A           ; Send command to data pins ; RSon Port 3
        CLR P1.3            = 0 (Instruction mode) ; Enable
        SETB P1.2           pin high
        ACALL DELAY
        CLR P1.2            ; Enable pin low
        ACALL DELAY
        RET

; LCD Data
LCD STRING:
        CLRA
        MOVC A, @A+DPTR
        JZ  LCD_STR_END
        ACALL LCD DATA INC
        DPTR   SJMP   LCD
        STRING

LCD STR END:
        RET

LCD DATA :
        MOV P3, A           ; Send data to Port 3
        SETB P1.3           ; RS = 1 (Data mode)
        CLR P1.1
        SETB P1.2           ; Enable pin high ;
        ACALL DELAY
        CLR P1.2            Enable pin low ; RS
        ACALL DELAY
        CLR P1.3            pin reset
        RET

; Delay Routine
DELAY :
DELAY1 :
DELAY2 :  MOV R6, #50
        MOV R7, #100

        DJNZ  R7,  DELAY2
        DJNZ  R6,  DELAY1
        RET

; Messages
welcome project:
        DB ”smart home automation”,
bulb on msg:
        DB ”Bulb is ON”, 0
bulb off msg:
        DB ”Bulb is OFF”, 0
        END
```