University of Southern California

Viterbi School of Engineering
Department of Aerospace & Mechanical Engineering

AME 504: Mechatronics Systems Engineering
Instructor: Dr. Yong Chen

Application Development Project: Slingo

Done By: Group 8: Aniket Dalvi, Francisco Rocha, & Kishan Reddy Jagannath

Due Date: May 10, 2025

Abstract

We took apart a QFX portable radio to learn where it falls short outdoors. We found weak reception, no hands-free control, and a bulky shell that lets dirt reach the ports. To fix these gaps we built our own unit. We chose parts for low power draw, bright outdoor readability, and voice or motion control. After laying out the circuits, we designed a 3-D-printed case and wrote firmware in the Arduino IDE with well-known libraries for I^2C, A2DP Bluetooth, FM tuning, and a 20 × 4 LCD.

Our first prototype worked but was far too large to hold in one hand. The second version used a tighter layout, yet that cramped space stressed our wiring and damaged the ESP32 board. We switched to an Arduino Mega and a MAX-series audio amp, rebuilt the harness, and reached every planned feature: FM radio, Bluetooth music, LED light control, and motion-triggered play. The result proves the concept and sets a clear path for a rugged field model.

Introduction

The QFX portable radio is a small plastic box with a built-in speaker. It tunes to FM and AM stations with a simple knob. A narrow LCD shows only the current frequency. A row of rubber buttons handles volume and station scan. The unit runs on AA cells and fits in a coat pocket. Students leave it on desks, and families keep it on kitchen shelves. Buyers praise its low price and lightweight. Indoors and at low volume it meets those basic needs well.

Our teardown exposed firm limits when the radio leaves the living room. The thin shell bends after short drops. Dust slips through open jack holes and coats the board. Audio cuts out when the whip antenna shakes. Tiny buttons resist use with cold or gloved fingers. The screen fades in strong sunlight. Battery life drops to three hours at full volume. No hands-free control exists for users who must keep both hands on a task.

Hikers, builders, and rescue crews need music and news. They need status alerts without pulling off gloves. A bright light can guide a camp path at night. A motion sensor can wake the unit as someone walks past. Voice prompts let the user skip tracks while holding tools. One gadget that blends these roles can cut pack weight. It must live through dust, rain, and knocks on hard ground. It must hide cables that might snag on straps.

We aim to close this gap with a rugged hand-held unit. We replace the stock tuner with an Arduino-based core and swap in a MAX audio amp for stronger sound. A sun-readable 20 × 4 LCD delivers clear menus and status. A voice board listens for simple play, pause, and light commands. A PIR sensor wakes the unit when motion is near and saves power when the area is quiet. Firefighters, foresters, and road crews can clip it to a belt, while campers can leave it on a table to light the site and stream songs. The design relies on common parts that are easy to source and service in the field.

QFX Radio Shortcomings

1. Lack of Digital Display
   The R-40 relies solely on analog tuning, which can make precise frequency selection challenging. Users have noted that the absence of a digital display requires manual adjustment to find exact stations.

2. Limited Audio Output
   Equipped with a single 2" speaker, the R-40 delivers basic sound quality suitable for casual listening. This was very basic and in fact didn’t provide a satisfactory listening experience.

3. Portability Concerns
   Although designed to be portable, the R-40's dimensions (approximately 7.09" H x 3.35" W x 8.27" D) and weight (around 1.43 pounds) may not be ideal for all users. The size and form factor could hinder ease of transport.

4. No utilization of space:
   When we dissected the QFX, we noticed that there was a lot of free space inside which could be utilized better.

Identified Problems & Targets

This device has a few major drawbacks that make it tough to use, especially if you're planning to take it outdoors. For starters, it was just too bulky. Carrying it around isn't convenient, and that really limits how portable it is. The buttons on the device are also confusing. There aren't any clear labels or feedback, so it's hard to tell what each one does. On top of that, switching between modes like Bluetooth and radio isn't obvious at all. There's no indicator or light to show what mode you're in, so you're left guessing most of the time. The built-in LED light doesn't help much either. It's stuck in a spot that doesn't really light anything up effectively, which kind of defeats the purpose. Finally, the added fan and solar charging features sound cool on paper, but in reality, they don't work well in many environments. They just add extra weight and complexity without offering much actual benefit.

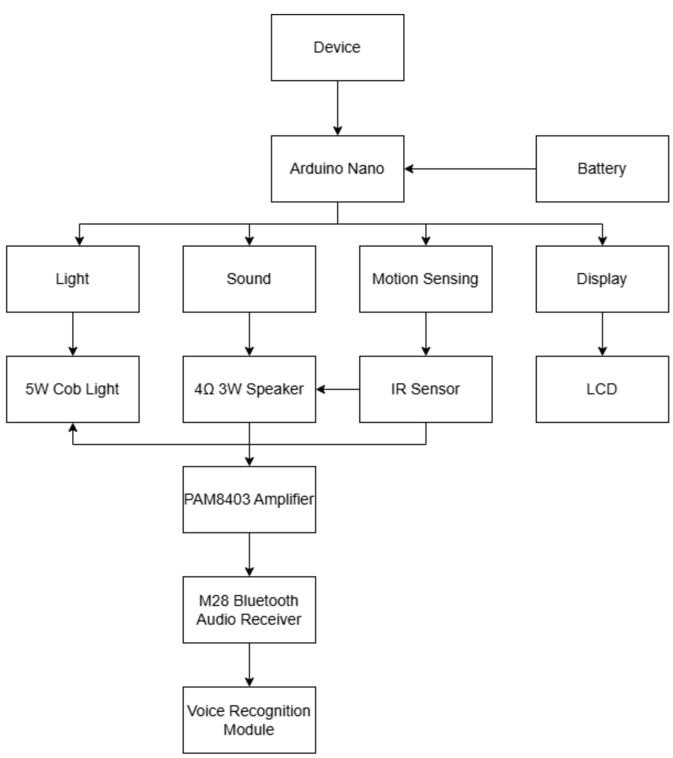
To make the device more practical, it should be lighter (under 500g) and more compact, with a built-in handle or strap option for better portability. The interface should offer clear tactile feedback so users can operate it without guesswork. Mode switching needs to be more intuitive, using a small screen to show which mode is active. The LED light should be repositioned for better directional use. Overall, the design should focus on ease of use, efficiency, and portability without unnecessary complexity.

Proposed Solutions

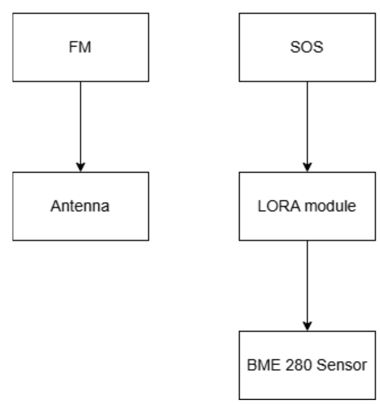
The redesigned device features a compact housing to reduce bulk and improve portability, making it more suitable for outdoor and mobile use. Voice recognition allows for hands-free operation, especially useful in rugged or low-visibility environments. An LCD display has been added to clearly show the current mode and system status, eliminating confusion during use. The LED light has been repositioned to enhance its effectiveness as a flashlight, providing better visibility and directional focus. A PIR motion sensor introduces added security and utility for camping, alerting users to nearby movement. The device now includes a modular feature set, having removed the fan and solar panel, and offers potential support for FM radio and SOS signaling depending on available build time and budget.

Product Breakdown Selection

Project Target One: Core Functional Block Diagram



Project Target Two:Optional Features

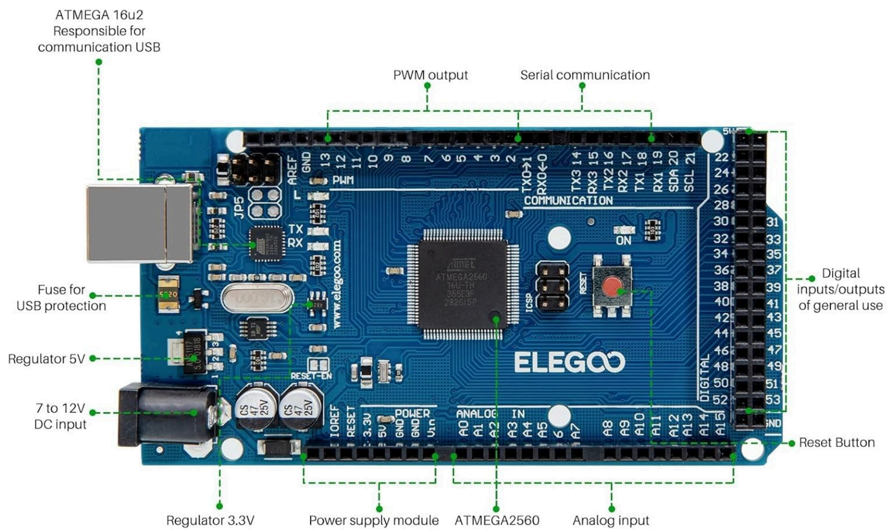


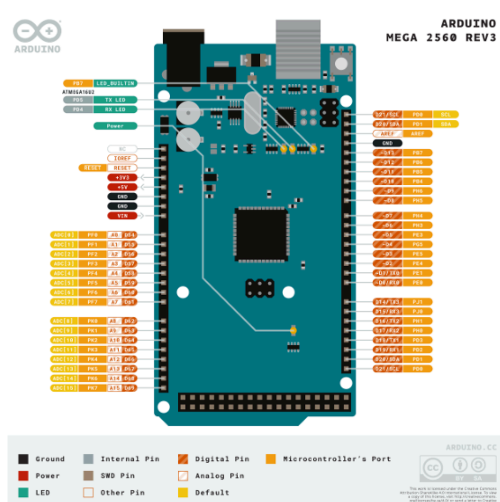
- The battery powers the Arduino Mega which then provides power to the Light, Sound, Motion Sensing, Display and FM Radio.
- The LED Light is a 5W Cob Light.

- The Sound uses a 4 Ohms 3W speaker to produce sound along with an amplifier to boost the signal. Sound could be played by bluetooth audio receiver, a voice recognition module and the FM radio with antenna for fine tuning.
- The motion sensing uses an IR sensor to detect motion and flashes a light at the source causing motion.
- The IR sensor can be activated by voice recognition or by a push button.
- The display used an LCD Screen to show each function working properly.
- FM Radio uses an antenna to acquire the proper frequency for the desired station.
- The SOS system utilized a LoRa 32 Module and BME sensor that was not able to be implemented due to insufficient time.

Hardware Selection

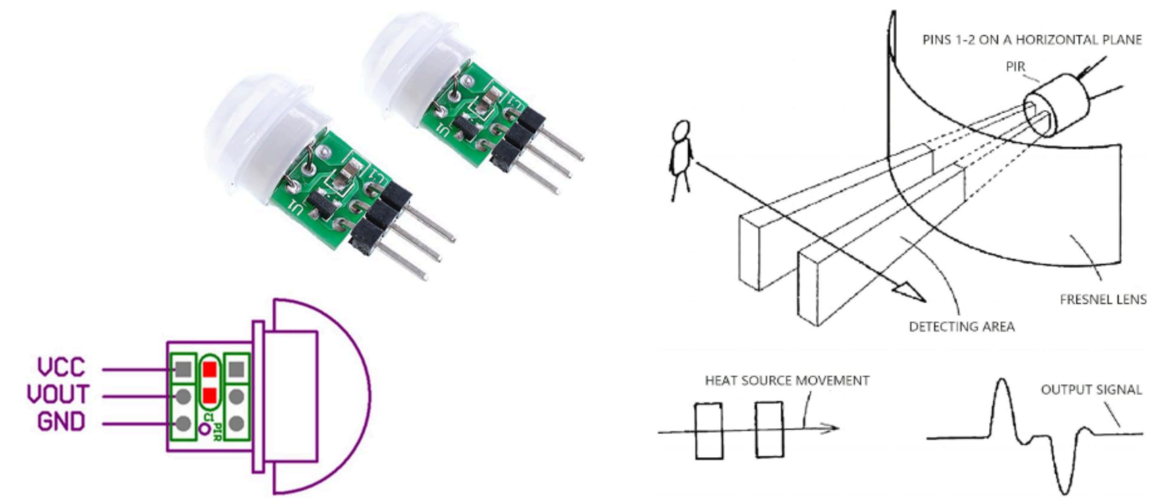
ELEGOO Arduino ATmega 2560:





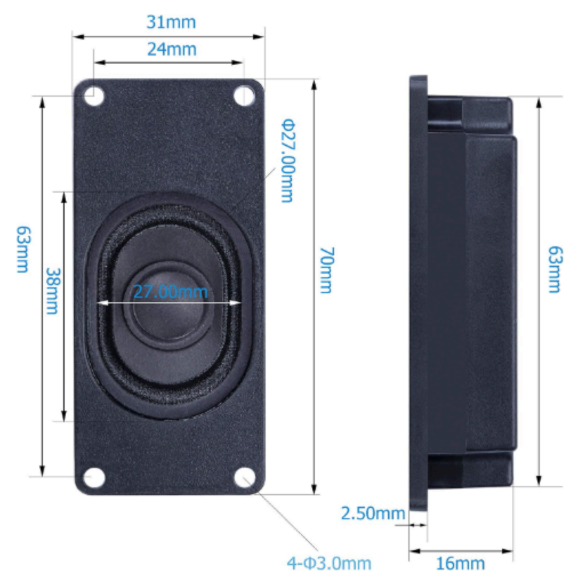
The ATmega 2560 microcontroller offers up to 16 MIPS (Million Instructions Per Second) throughput at 16 MHz. It includes features such as a real time counter with a separate oscillator, four 8-bit PWM channels, and four programmable serial UARTs. The board also supports various sleep modes, including idle, ADC noise reduction, power save, power down, standby, and extended standby making it useful for power management of our project. Its compatibility with Arduino IDE, and Arduino Cloud Editor allows for easy programming and integration into Slingo's necessary components. For this project, we used the power jack to supply the necessary voltage for the project.

AM312 Mini Pir Motion Sensor Module HC-SR312 IR:



The AM312 Mini PIR Motion Sensor Module HC-SR312 IR detects motion using a pyroelectric sensor that senses changes in infrared radiation from warm objects (like humans or animals). Typical detection range is up to 7 meters, with a detection angle of about 110 degrees. Operates on 4.5V–12V DC and outputs a 3.3V digital signal when motion is detected. Sensitivity and delay time are adjustable; commonly used in security, automation, and lighting control applications.

CQ Robot Speakers:

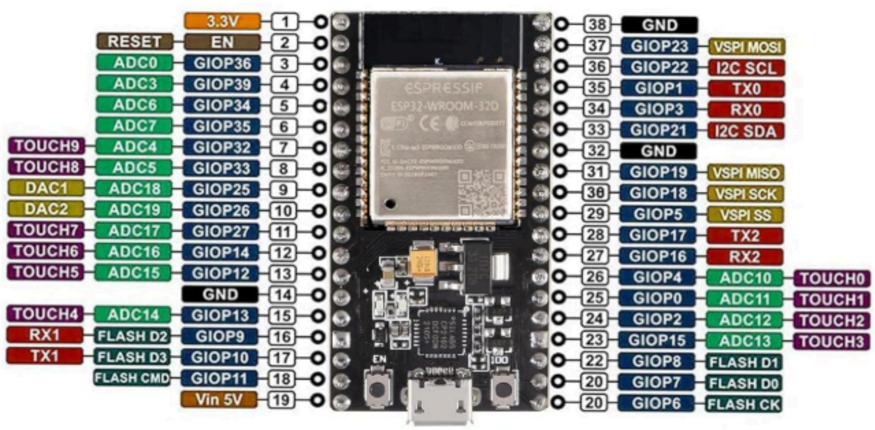


This miniature loudspeaker is surprisingly powerful for its size, making it a great fit for compact projects like Slingo where space is tight but clear audio still matters. Rated at 3 to 5 watts, it delivers decent volume without distortion and works well with small amplifiers like the Teyleten Robot MAX98357 Class D Amplifier. Its lightweight, durable build, and easy mounting holes make it perfect for this portable project since a basic speaker is needed without sacrificing sound quality. The Bluetooth Module can play music through this speaker, along with the voice-feedback gadget. It's a reliable sound output that gets the job done without taking up much room.

TEA5767 FM Module:

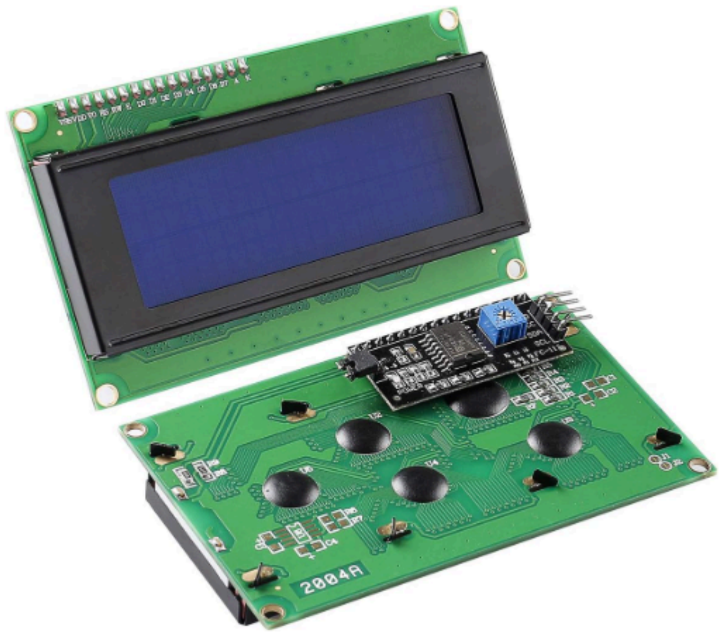
The TEA5767 FM radio module is a compact and reliable way to add radio functionality to the project. It picks up FM stations clearly and works well with microcontrollers like Arduino, letting you tune stations digitally through I2C control instead of using manual dials. The module is small enough to fit into tight spaces, and it comes with onboard components like a quartz oscillator and stereo audio output, so setup is straightforward. It’s a solid choice for our portable compact radio since our project needs to be lightweight and have low-power FM reception is needed without overcomplicating the design.

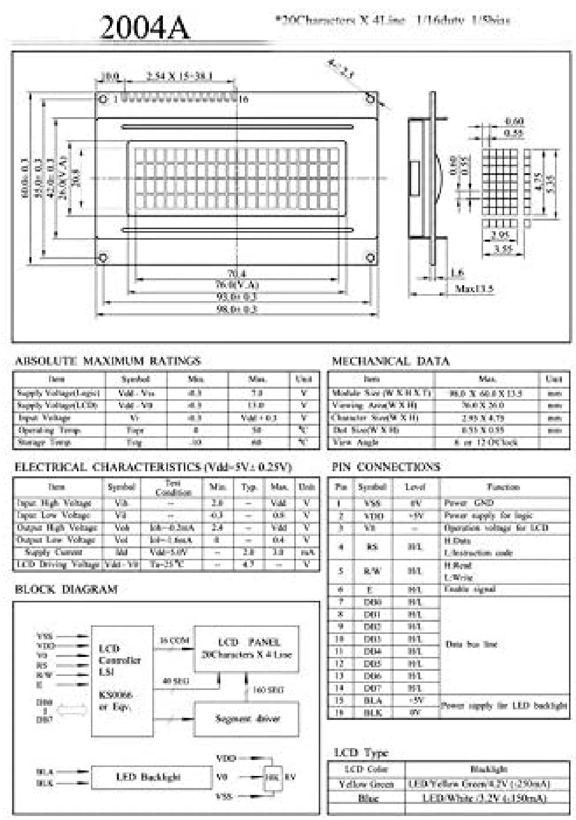
HiLetgo ESP-32D:



The HiLetgo ESP-32D is a compact yet powerful microcontroller board that's perfect for our wireless IoT project. It has built-in Wi-Fi and Bluetooth, it allows for fast and flexible communication without the need for extra modules. Despite its small size, it packs a dual-core processor and plenty of GPIO pins, making it capable of handling multiple tasks at once everything from sensor readings to real-time data transmission. It's easy to program using the Arduino IDE. It’s well-suited to make Slingo a wireless control.

SunFounder IIC I2C TWI Serial 2004 20x4 LCD Module:





ABSOLUTE MAXIMUM RATINGS

Item	Symbol	Min.	Max.	Unit
Supply Voltage(Logic)	Vdd - Vss	-0.3	7.0	V
Supply Voltage(LCD)	Vdd - V0	-0.3	13.0	V
Input Voltage	Vi	-0.3	Vdd + 0.3	V
Operating Temp.	Topr	0	50	℃
Storage Temp.	Tstg	-10	60	℃

MECHANICAL DATA

Item	Max.	Unit
Module Size (W X H X T)	98.0 X 60.0 X 13.5	mm
Viewing Area(W X H)	76.0 X 26.0	mm
Character Size(W X H)	2.95 X 4.75	mm
Dot Size(W X H)	0.55 X 0.55	mm
View Angle	6 or 12 O'Clock	

ELECTRICAL CHARACTERISTICS (Vdd=5V± 0.25V)

Item	Symbol	Test Condition	Min.	Typ.	Max.	Unit
Input High Voltage	Vih	--	2.2	--	Vdd	V
Input Low Voltage	Vil	--	-0.3	--	0.6	V
Output High Voltage	Voh	Ioh=-0.2mA	2.4	--	Vdd	V
Output Low Voltage	Vol	Iol=1.6mA	0	--	0.4	V
Supply Current	Idd	Vdd=5.0V	--	2.0	3.0	mA
LCD Driving Voltage	Vdd - V0	Ta=25℃	--	4.7	--	V

BLOCK DIAGRAM

PIN CONNECTIONS

Pin	Symbol	Level	Function
1	VSS	0V	Power GND
2	VDD	+5V	Power supply for logic
3	V0	--	Operation voltage for LCD
4	RS	H/L	H:Data L:Instruction code
5	R/W	H/L	H:Read L:Write
6	E	H/L	Enable signal
7	DB0	H/L	Data bus line
8	DB1	H/L	
9	DB2	H/L	
10	DB3	H/L	
11	DB4	H/L	
12	DB5	H/L	
13	DB6	H/L	
14	DB7	H/L	
15	BLA	+5V	Power supply for LED backlight
16	BLK	0V	

LCD Type

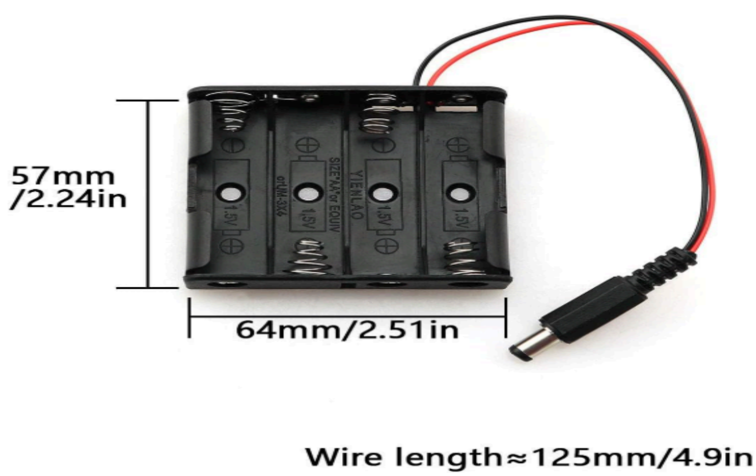
LCD Color	Backlight
Yellow Green	LED/Yellow Green/4.2V (≤250mA)
Blue	LED/White /3.2V (≤150mA)

The SunFounder 20x4 LCD module is a simple and effective way to display a lot of information in a small space with 4 rows and 20 characters per row. It's perfect for showing multiple lines of sensor data, menus, or system messages. It uses I2C communication, which means it only needs two wires to connect to the microcontroller, saving pins and reducing clutter. The text is easy to read, and the backlight makes it visible even in low light. It's a compact display without complicated wiring.

AA Batteries:

The SLINGO is powered by four AA 1.5V alkaline batteries, delivering a combined 6V through a standard battery holder. After several uses, the voltage naturally drops, leveling out around 5V, which still provides consistent and reliable power. It's designed with extended field use in mind, featuring a low-power standby mode and a battery setup that's easy to replace by swapping in new AA batteries.

ZRM&E 6V Battery Holder:

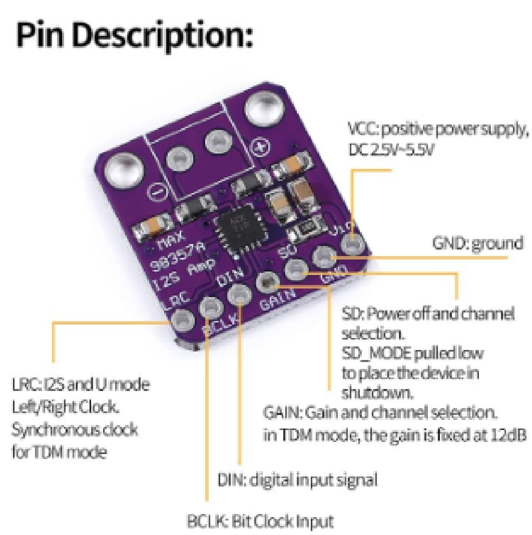


The battery holder connects directly to the Elegoo Arduino, and the built-in LCD screen displays the remaining battery percentage making it easy to monitor power levels at a glance.

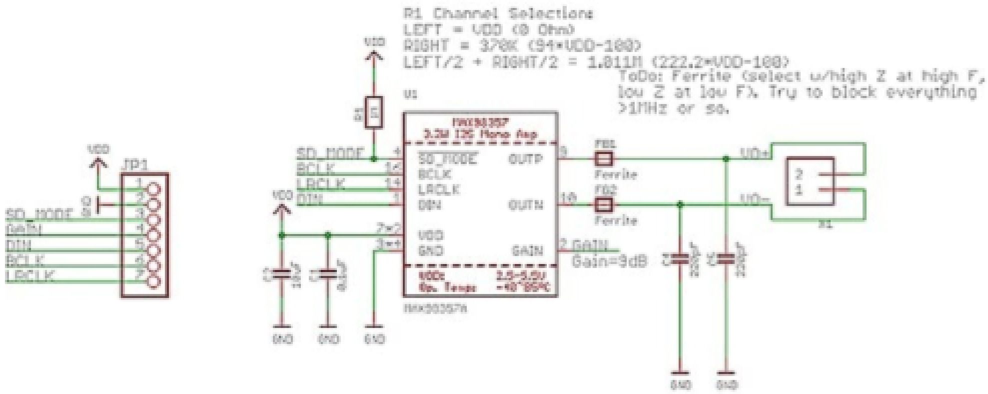
CHANZON High Power LED Chip:

The LED in this system is activated by an IR sensor, which detects motion or obstruction and triggers the light through a control circuit. A 3W high-power LED is connected to a constant current driver that supplies 600–700 mA at 3–3.4 V, ensuring stable brightness and protecting the LED from voltage fluctuations. The Arduino controls the LED using a MOSFET switch, allowing efficient on/off control. Users can toggle the LED manually with a push button or hands-free using a voice command, adding both convenience and flexibility to the system.

Teyleten Robot Max98357 I2S 3W Class D Amplifier:

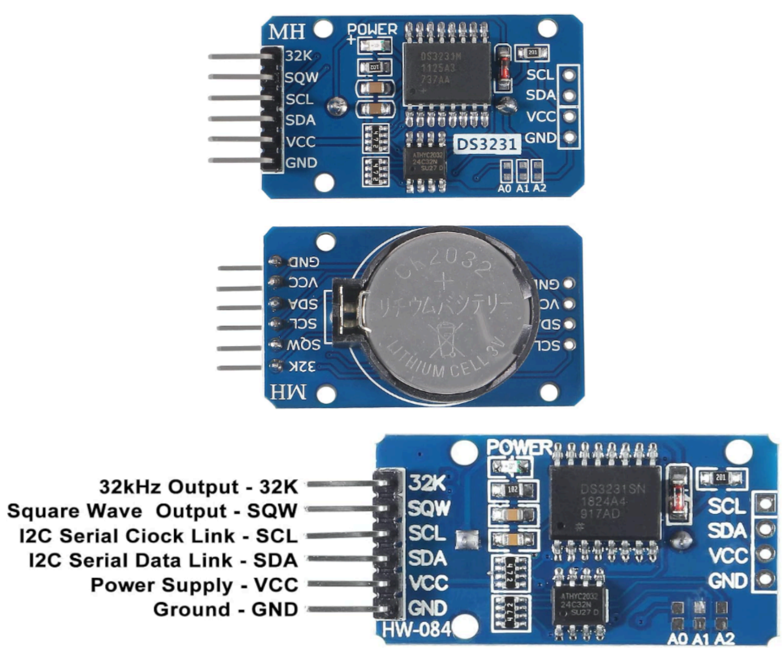


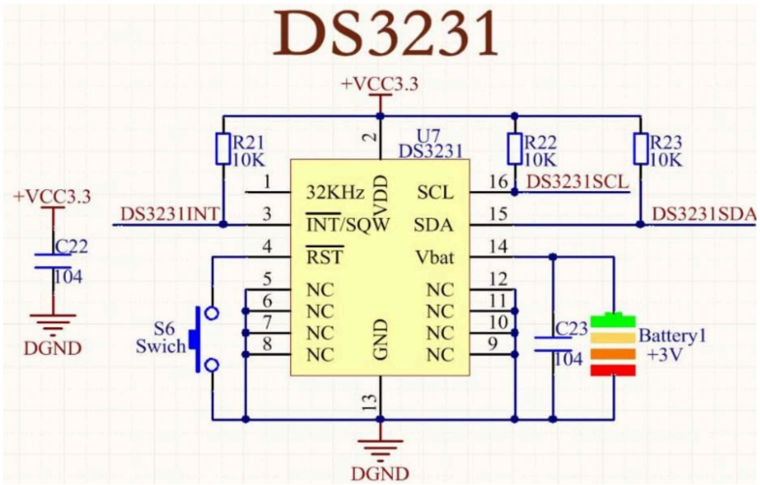
· Schematic

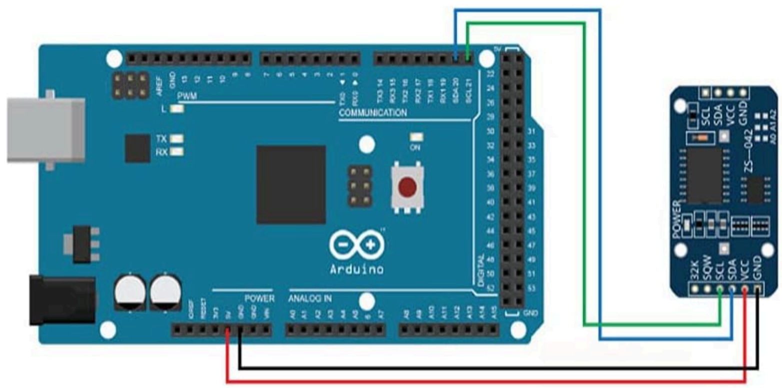


The DFRobot MAX98357 I2S Amplifier Module is a small but powerful audio component that makes it easy to add quality sound to the microcontroller. It uses I2S digital audio input, allowing it to connect directly to boards like the ESP32D without complex wiring. A built-in 3W class D amplifier, it can drive a standard 4Ω speaker with clear, low-distortion output. The module handles audio signal conversion and amplification in one compact package, simplifying the design process for things like voice assistants, audio playback devices, or smart alarms. Its low power consumption and clean sound make it a great choice for the portable project.

DS3231 AT24C32 IIC RTC Clock Module:

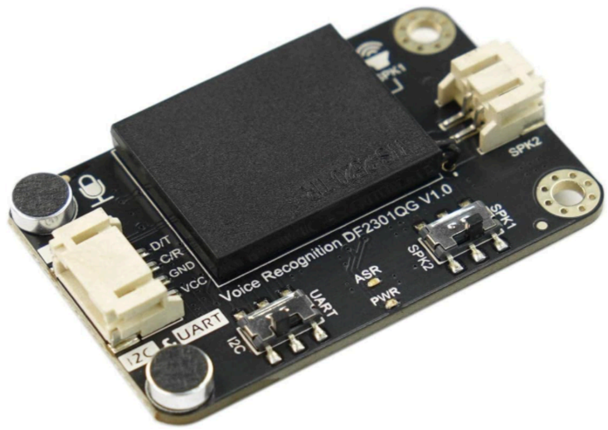


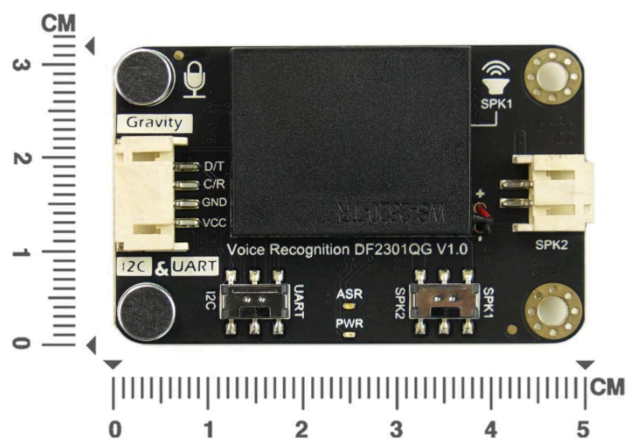




The DS3231 AT24C32 Real-Time Clock (RTC) Module is a highly accurate, battery-backed clock module ideal for keeping precise time. Unlike basic RTCs, the DS3231 features a built-in temperature compensated crystal oscillator, ensuring consistent timekeeping even with changing environmental conditions. It communicates over the I2C interface, making it easy to integrate to the Arduino. The onboard AT24C32 EEPROM provides 32KB of non-volatile memory for storing user data, such as time stamps or sensor logs. It has the ability to maintain time even when the main system is powered off, this module is perfect for data logging, scheduling, and low-power applications that require long-term time tracking

Gravity: Offline Language Learning Voice Recognition Sensor:





The DFRobot Gravity Voice Recognition Module V3 is a compact, offline speech recognition module that allows users to control devices with voice commands with no internet connection required. It supports up to 80 preloaded command words and allows users to train up to seven custom commands, making it flexible for personalized control. The module uses a simple serial interface (UART) for easy integration with Arduino. Its onboard microphone and built-in speech processing chip handle all recognition tasks locally, ensuring fast and reliable responses even in noisy environments. Ideal for making this a hands-free project since it makes it a straightforward way to add voice control to embedded systems.

Full Component List:

Components	Description	Arduino Compatible & Code	Supplier	Quantity	Estimated Cost
ELEGOO Mega R3 Board Arduino ATmega 2560	Main Microcontroller Board	Fully Supported by Arduino IDE	Acquired	1	Acquired
BreadBoard	Prototyping Tool Designed For Building & Testing Electrical Circuits	Integrates With Arduino ATmega 2560	Acquired	1.5	Acquired
Jumper Wires	Flexible Insulated Jumper Wires To Connect Components	Integrates With Arduino AtMega 2560, Breadboard,& All Other Components	Acquired	50	Acquired
AM312 Mini Pir Motion Sensor Module HC-SR312 IR	Detects motion using a pyroelectric sensor that senses changes in infrared radiation from warm objects	Fully Compatible With Arduino. digitalRead()	Amazon.com : AM312 Mini Pir Motion Sensor Module HC-SR312 IR	1	$9.98
CQ Robot Speakers	3 Watt 4 Ohm	Compatible with Arduino Motherboard. Passive, No Code Needed.	Amazon.com: CQRobot Speaker 3 Watt 4 Ohm Compatible with Arduino Motherboard. 2.54mm Dupont Interface. It is Ideal for a Variety of Small Electronic Projects. :	2	$7.99

			Electronics		
TEA5767 FM Module	Digital FM Tuning	Wire.h + TEA5767 Library	Amazon.com : tea5767	1	$5.50
HiLetgo ESP-32D	2.4GHz Dual-Mode WiFi + Bluetooth Dual Cores Microcontroller Processor	Integrates with Antenna RF AMP Filter AP STA for Arduino IDE	Amazon.com: HiLetgo ESP-WROOM-32 ESP32 ESP-32S Development Board 2.4GHz Dual-Mode WiFi + Bluetooth Dual Cores Microcontroller Processor Integrated with Antenna RF AMP Filter AP STA for Arduino IDE : Electronics	1	$9.99
SunFounder IIC I2C TWI Serial 2004 20x4 LCD Module	Operates at 5V DC, with adjustable contrast via a potentiometer on the back and an optional backlight jumper.	LiquidCrystal_I2C library for Arduino	Amazon.com: SunFounder IIC/I2C/TWI LCD1602 Display Module Compatible with Arduino and Raspberry Pi : Electronics	1	$9.99
Amazon Basics 12-Pack Batteries	AA Alkaline Batteries, 1.5 Volt, Long Lasting Power	Supplies enough power for Components	Amazon.com: Amazon Basics 12-Pack AA Alkaline Batteries, 1.5 Volt, Long Lasting Power : Health & Household	1	$8.13

ZRM&E 6V Battery Holder	Holds up to 4 1.5V AA Batteries	Compatible with Arduino ATmega 2560 From A Barrel Jack	Amazon.com: ZRM&E 6V Power Box 4xAA Battery Holder Case with DC 5.5x2.1 Connector Mate with Barrel Jack Connector on Arduino : Electronics	1	$6.99
CHANZON High Power LED Chip 10 Pack	3W Natural White (4000K-4500K/ 600-700mA) Super Bright Intensity	Compatible With Arduino & BreadBoard	CHANZON 10 pcs High Power Led Chip 3W Natural White (4000K - 4500K / 600mA - 700mA / DC 3V - 3.4V / 3 Watt) Super Bright Intensity SMD COB Light Emitter Components Diode 3 W Bulb Lamp DIY Lighting - Amazon.com	1	$6.99
Push Buttons	Small Tactile Switches To Manually Input Signals	Arduino Compatible	Acquired	4	Acquired
Teyleten Robot Max98357 I2S 3W Class D Amplifier 5 Pack	Prototyping Audio Features in IOT Devices	Compatible With Arduino	Amazon.com: Teyleten Robot Max98357 I2S 3W Class D Amplifier Breakout Interface Dac Decoder Module Filterless Audio Board for Raspberry Pi Esp32 5pcs : Industrial & Scientific	1	$12.88

DS3231 AT24C32 IIC RTC Clock Module	Precise Time Keeping In MicroController Projects	Compatible With Arduino. RTClib or DS3231RTC Libraries	2PCS DS3231 AT24C32 IIC RTC Module Clock Timer Memory Module Beats Replace DS1307 I2C RTC Board with Male to Female Jumper Wire Cable at Amazon.com	1	$6.59
CR2032 Lithium Battery (4 Pack)	3 Volt Lithium Coin Cell	Used For DS1307 Clock Module	Amazon.com: Amazon Basics CR2032 Lithium Batteries, 4-Pack, Child-Proof Package, Non-Coated - AirTag Compatible : Health & Household	1	$3.97
Resistors	10K Ohms Resistance Values	Compatible With Arduino	Acquired	7	Acquired
Gravity: Offline Language Learning Voice Recognition Sensor	Detects Voice Commands OfflineUsing Built-In & Trainable Speech Recognition For Arduino and ESP32D	Compatible With Arduino.	Amazon.com : Gravity: Offline Language Learning Voice Recognition Sensor for micro:bit/Arduino / ESP32 - I2C & UART	1	$21.90

Total	76.5	$110.90

CAD Design

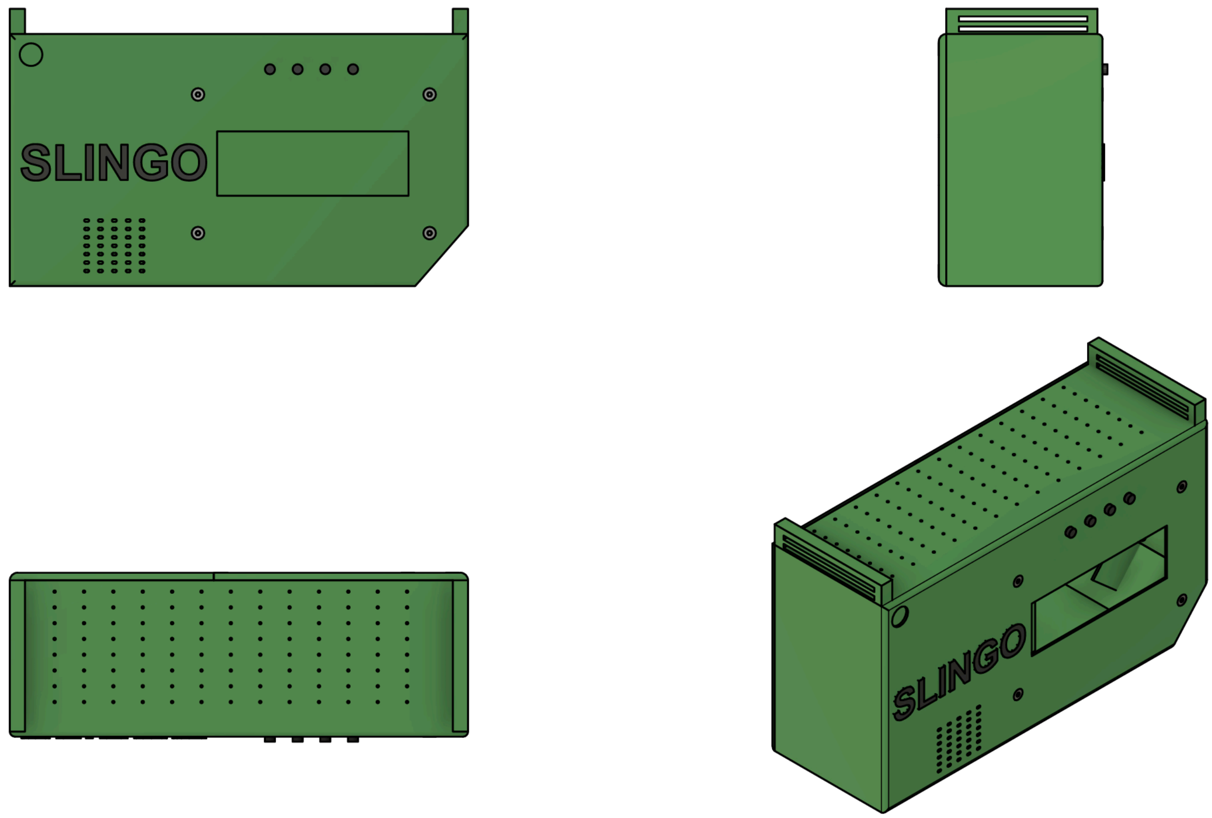
The casing for the speaker was made using Fusion 360. DFMA principles and additive manufacturing limitations were considered when designing the case.

All components are explained in detail below. All the dimensions were in accordance with one of the objectives, which was to do topology optimization and reduce the form factor to the smallest size possible. The case was printed in three different parts. The main body was printed with the back panel open. Then the back panel was printed in two parts, one which was to be screwed onto the main body, the second was a snap fit panel. The snap fit panel was designed to easily change the batteries and make the product functional from a customer point of view. All the edges were filleted with a 3 mm radius to make holding it easier and not having any sharp edges. The 3D printer used was Bambu X1E.

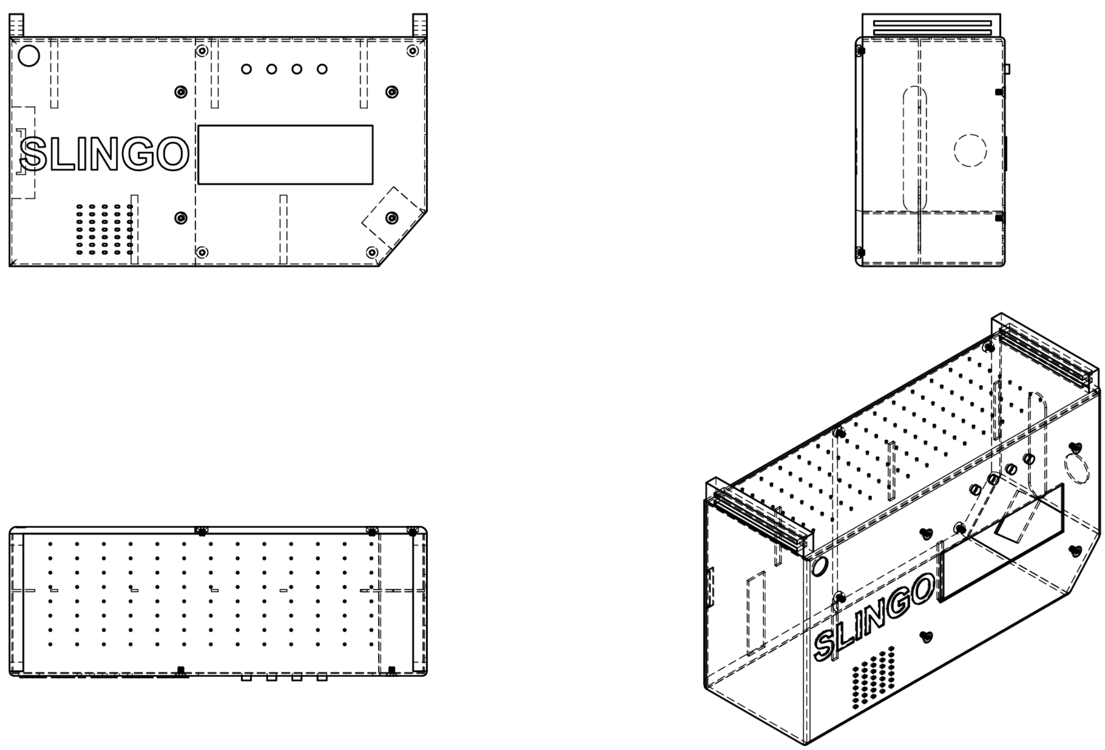
Objectives of design:

1. Reduce the form factor
2. Ease of portability
3. Ease in holding the device
4. Modular enough to prototype different components
5. Internal structure to provide support to wiring

Technical drawing with shaded edges

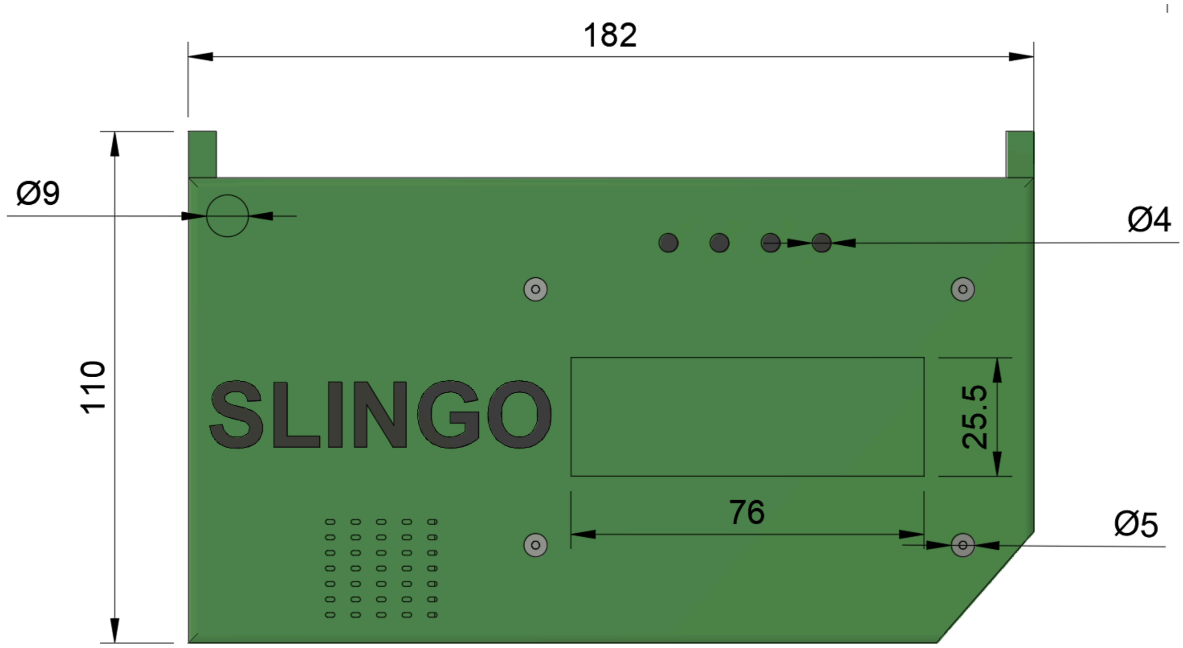


Technical drawing with visible and hidden edges

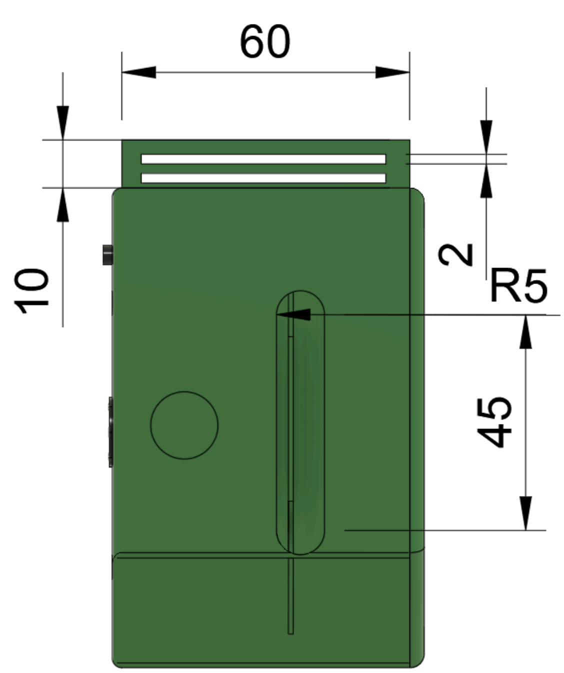


Views

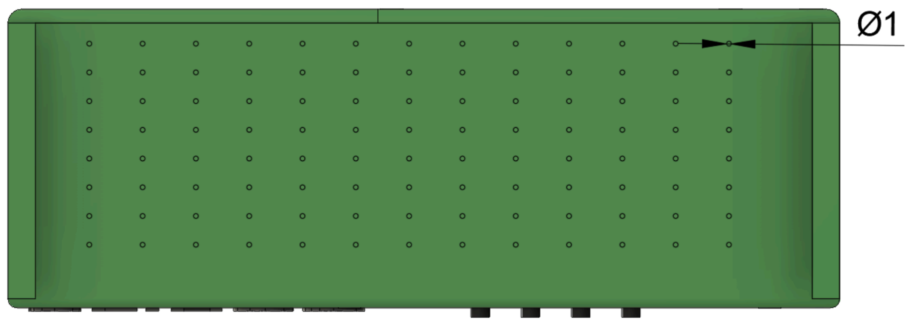
Front View



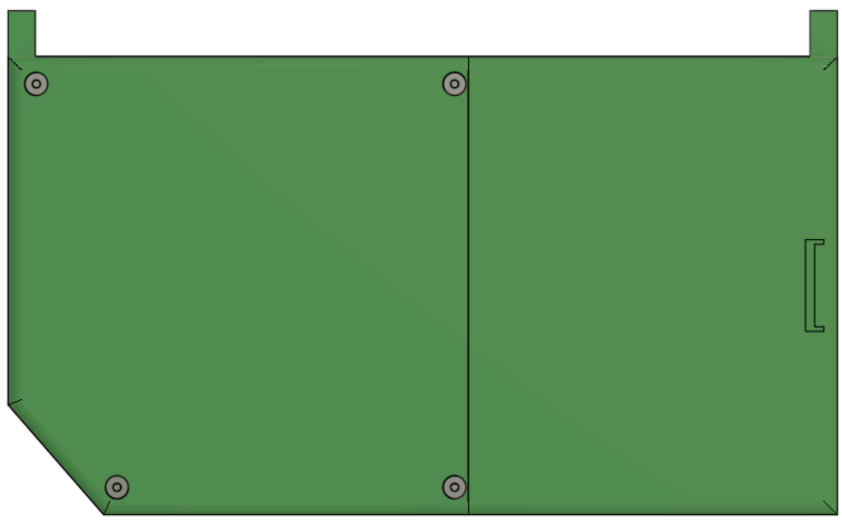
Side View



Top View

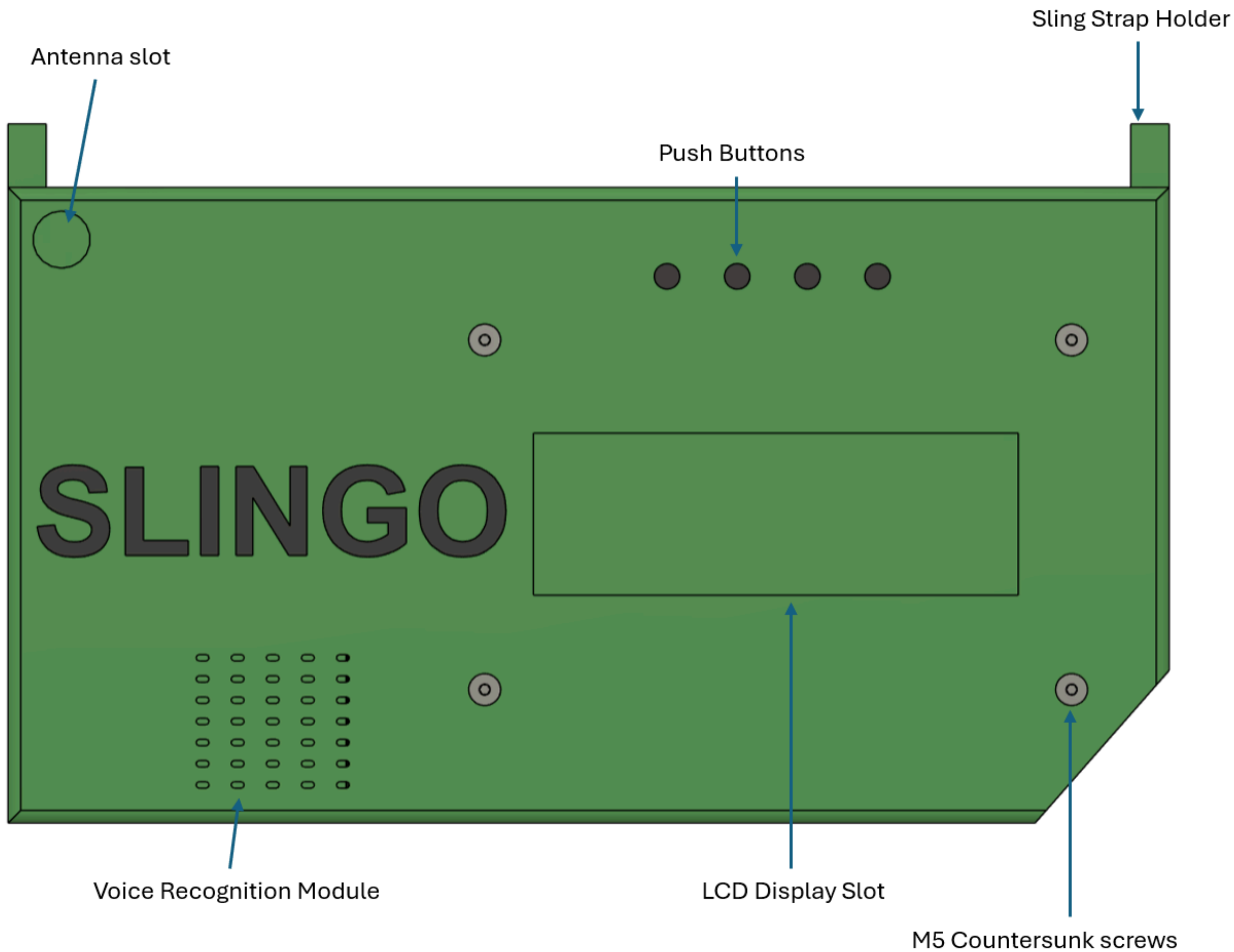


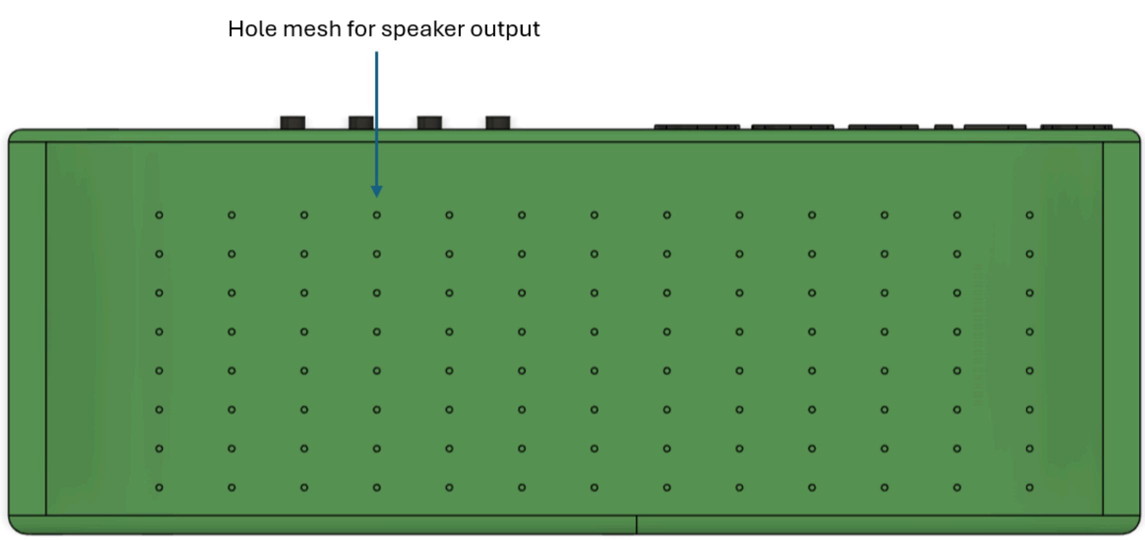
Bottom View

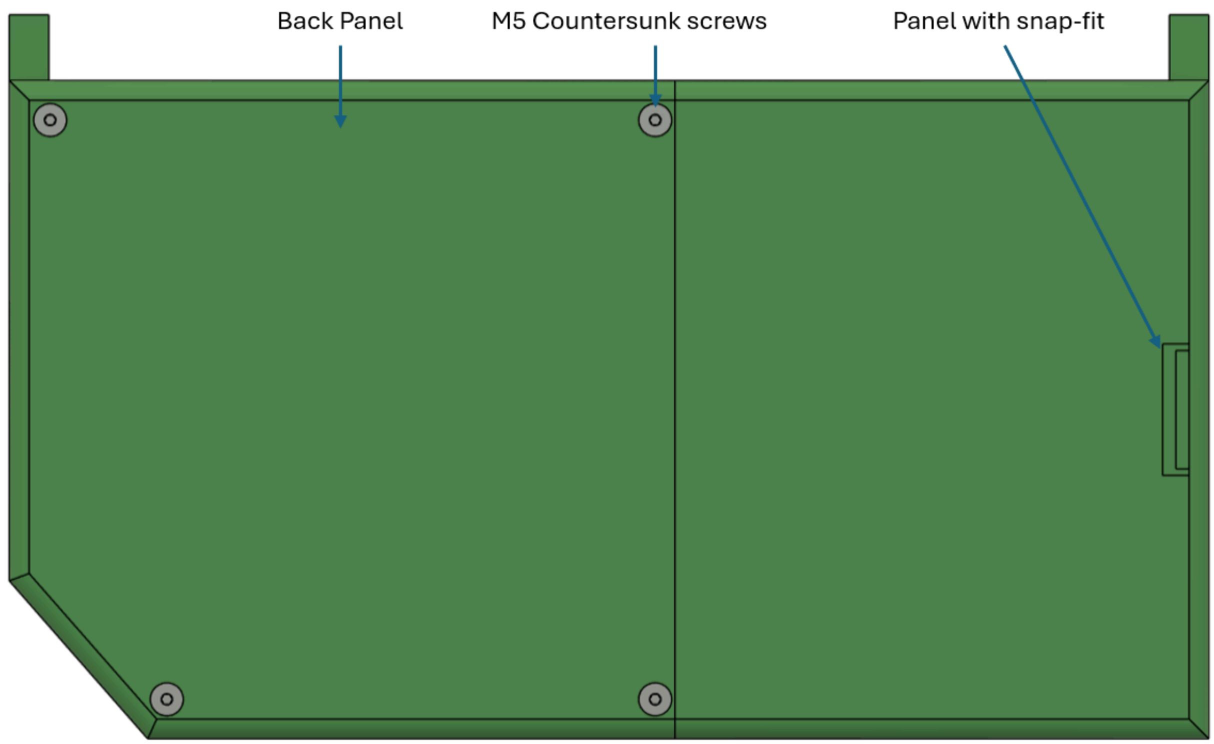


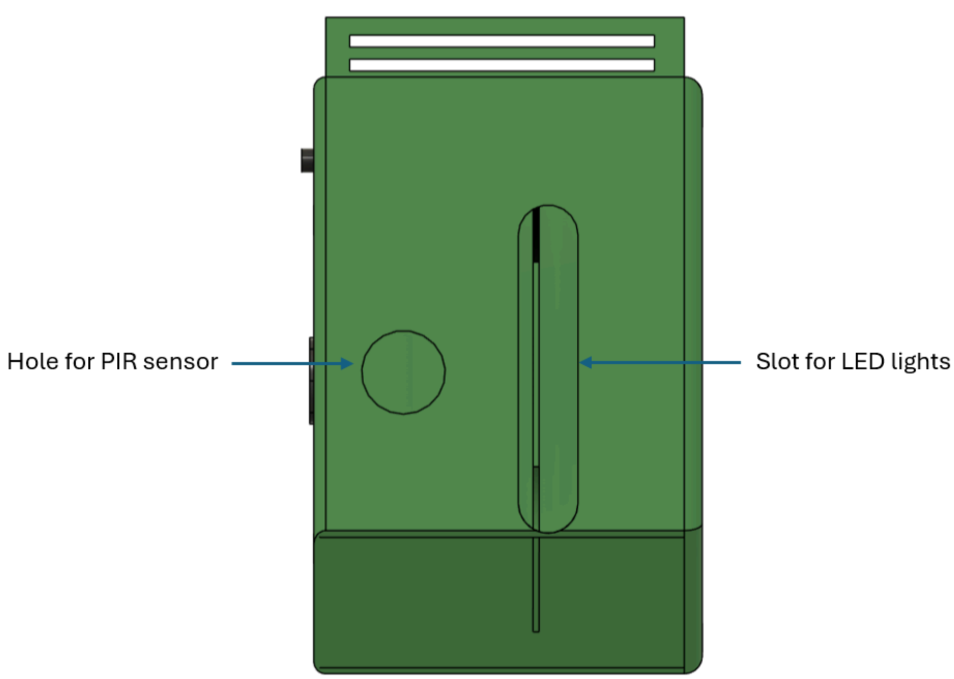
Isometric View

Feature Placement:



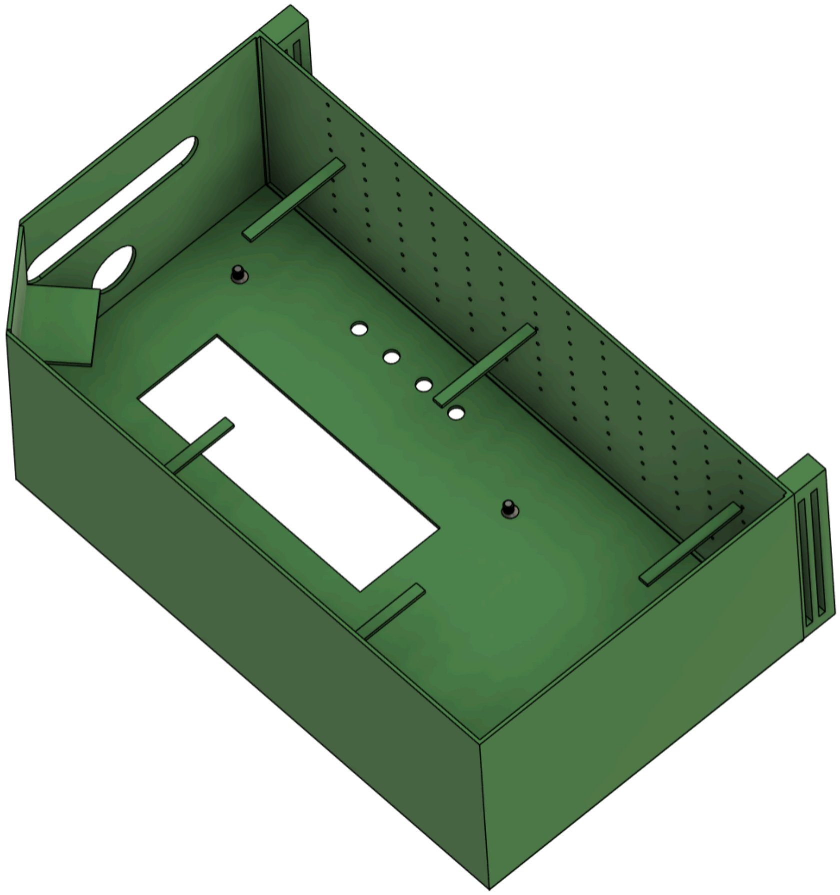


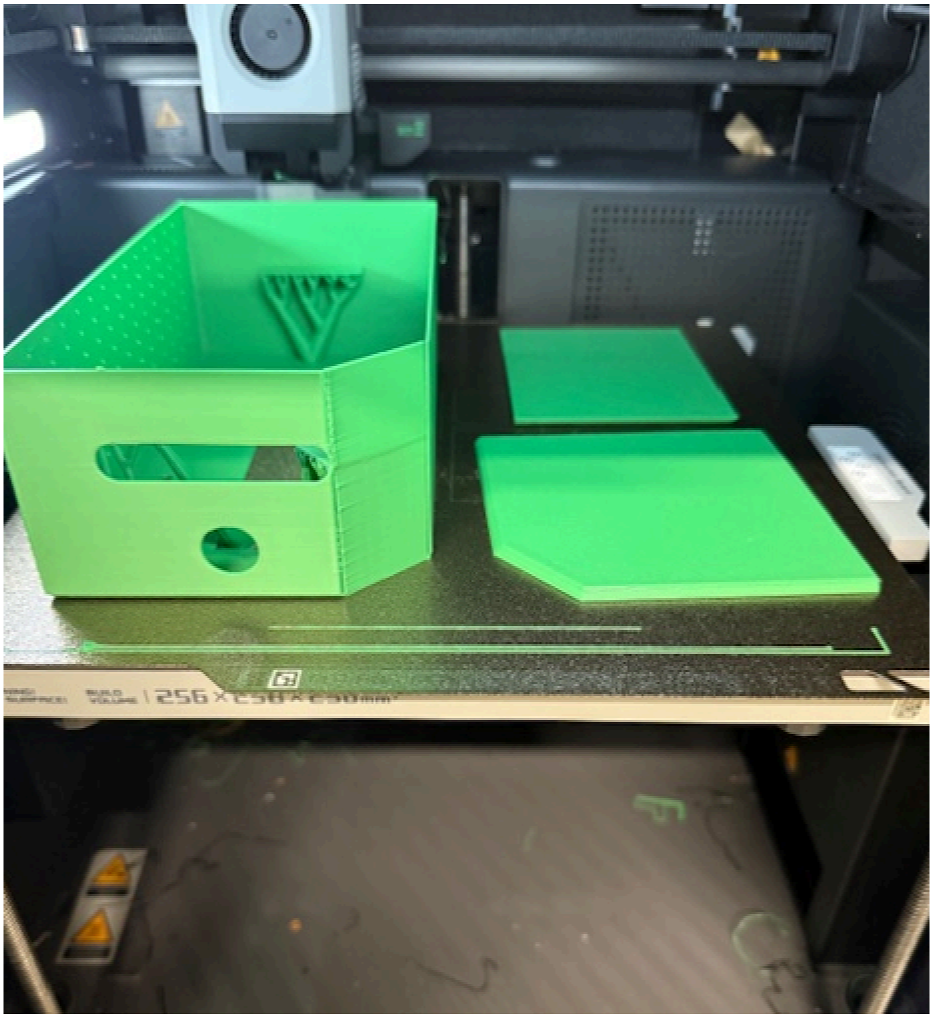

Back Panel
M5 Countersunk screws
Panel with snap-fit


Hole for PIR sensor
Slot for LED lights

Isometric View of internal structure

It has support brackets printed on the inside to support the modules and wiring.

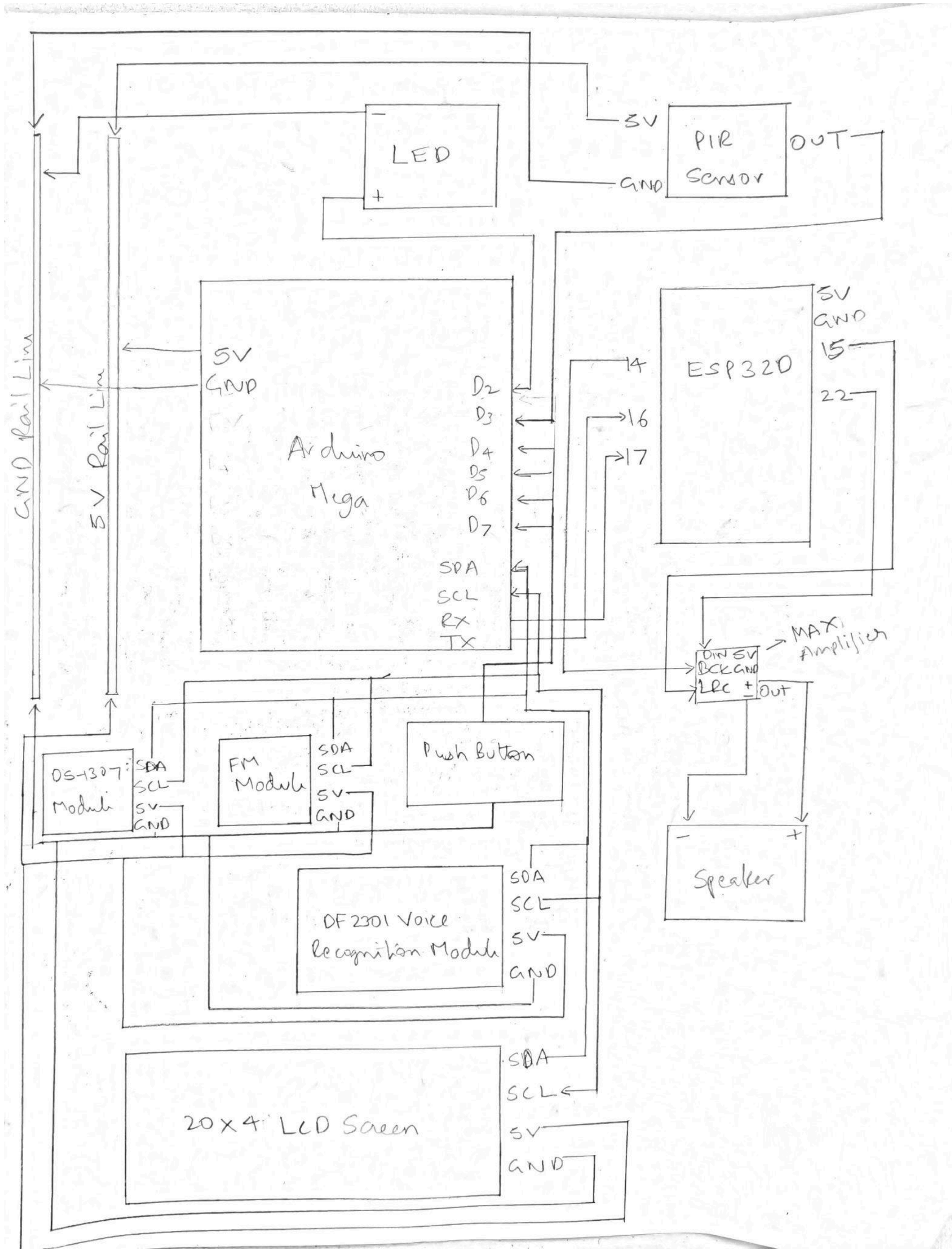




Bill of Materials for enclosure:

No.	Part	Quantity	Source
1	Main Panel	1	3D printed
2	Back Panel I	1	3D printed
3	Back Panel II with snap fit	1	3D printed
4	Push Buttons	4	3D printed
5	M5 Countersunk screws	8	McMaster-Carr

Electrical Wiring/Diagram



The electrical system is built around an Arduino Mega 2560, which serves as the main controller for all peripheral devices. It manages display output, user inputs, motion sensing, lighting control, and coordination with the Bluetooth audio system. A 20 × 4 LCD screen, FM radio module, voice recognition module (DF2301), and a set of push buttons are all connected to the Arduino through a shared I²C bus. These modules receive 5V power and ground through a common distribution rail.

An ESP32 module operates alongside the Arduino but functions independently. It is dedicated to Bluetooth audio streaming and communicates with a MAX-series audio amplifier. The amplifier takes the Bluetooth audio signal and drives an external speaker. Both the ESP32 and the amplifier share the same 5V power rail, ensuring voltage consistency across all active components. Serial communication is established between the Arduino and ESP32 for control signals, but the audio stream remains isolated within the ESP32 subsystem.

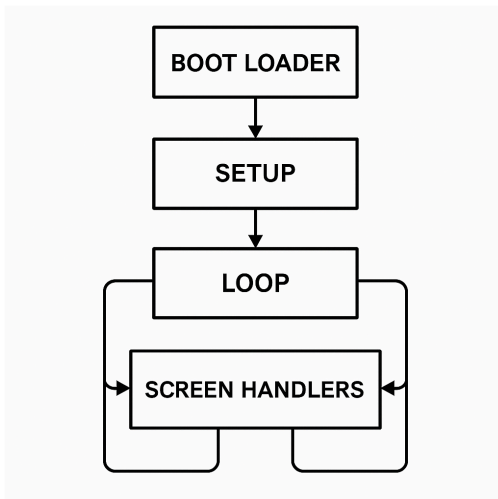
Motion sensing is achieved using a PIR sensor connected to one of the Arduino's digital input pins (D3). An LED used for lighting or signaling is driven by another digital output pin (D2), both drawing from the regulated 5V supply. The Arduino monitors the PIR sensor state and updates the LED or screen accordingly to save power when the device is idle.

All devices share a centralized 5V and ground rail system to simplify wiring and maintain stable voltage references. However, due to the use of breadboards and jumper wires, voltage stability under dynamic loads was a challenge during testing. Future designs will consolidate this wiring into a more reliable printed circuit board (PCB) layout to ensure stronger mechanical and electrical performance.

Software Breakdown

The code runs a single loop that polls input pins, updates each service, and refreshes the LCD. A state machine holds seven screens: Main Menu, Motion, LED, FM, Speaker (stub), Hands-Free, Bluetooth. Each screen owns its own draw and input blocks.
All services share common helper calls (pressed(), clearRow(), drawHeader()), so the loop stays short.

Figure 1 shows the flow from power-up to run time.



Third-party libraries

Library	Role in project	IDE version used	Notes
Wire	I²C backbone for LCD and voice module	v1.0.1	100 kHz default
LiquidCrystal_I2C	20 × 4 display driver	v1.1.2	Uses 0x27 address
TimeLib	Simple RTC replacement for clock display	v1.6	Time set from compile time
bluetooth-a2dp	ESP32 A2DP sink	v1.9.0	Wrapped by BluetoothModule
TEA5767	FM tuner helper	v1.0.0	Inside FMRadioModule
DFRobot_DF2301Q	I²C voice module	v1.0.6	Provides getCMDID() calls

All other helpers (MotionSensor.h, LEDDimmer.h, etc.) are local classes.

Module summary

Service	File(s)	Key public calls	Update cost*
MotionSensorModule	MotionSensor.h/.cpp	begin(), enable(), disable(), update()	100 µs
LEDDimmerModule	LEDDimmer.h/.cpp	begin(), set(), get(), inc(), dec(), update()	80 µs
FMRadioModule	FMRadio.h/.cpp	begin(), enable(), prev(), next(), current()	200 µs
BluetoothModule	bluetooth.h/.cpp	begin(), enter(), handle(), update(), active()	450 µs
Voice Module	DFRobot_DF2301Q	getCMDID()	50 µs

*Average time per update() call measured with micros().

State-machine logic

MAIN_MENU ↔ (OK) → sub-screen
sub-screen ↔ (BACK) → MAIN_MENU
BT_MODULE handles its own mini-state through BluetoothModule::handle()
Hands-Free can trigger changes from any state Each screen draw routine writes only the rows it owns, preventing LCD flicker. pressed(pin) uses edge detection so long presses do not auto-repeat.

Timing and power handling

- Clock redraws once each minute to save I^2C traffic.
- PIR sensor wakes the user interface; the LED dimmer lowers brightness when no motion.
- Voice module sleep time is set to 15 s (setWakeTime(15)) to reduce idle drain.

Voice-command mapError handling

- If the voice chip is not detected at boot, the code prints “Voice comm error” on Serial and retries every 3 s.

Command ID	Action	Screen impacted
7	Motion on	Main Menu
8	Motion off	Main Menu
9	Enter FM	FM Module
10	Next station	FM Module
11	Exit FM	FM Module
12	Exit Bluetooth	BT Module
92	Play music	BT Module
93	Pause music	BT Module
94	Previous track	BT Module
95	Next track	BT Module
103	LED full	LED Module
104	LED off	LED Module
105	LED up	LED Module
106	LED down	LED Module

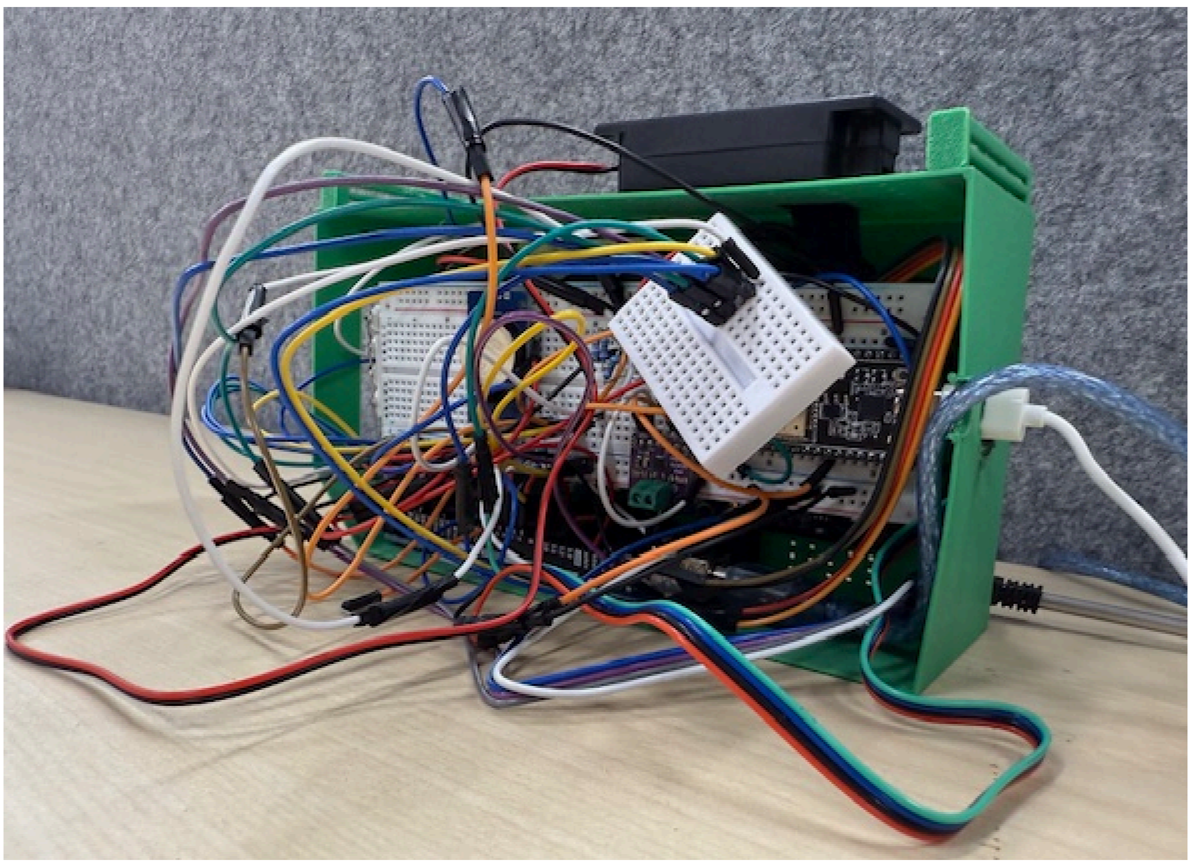
- BluetoothModule::handle() returns the new screen by reference; an invalid state falls back to Main Menu.

- Each begin() call returns quickly so the device reaches interactive mode in under three seconds.

Memory footprint (Mega 2560)

Area	Bytes used	Percent of 256 kB
Flash	51 kB	20 %
SRAM	4.6 kB	9 %

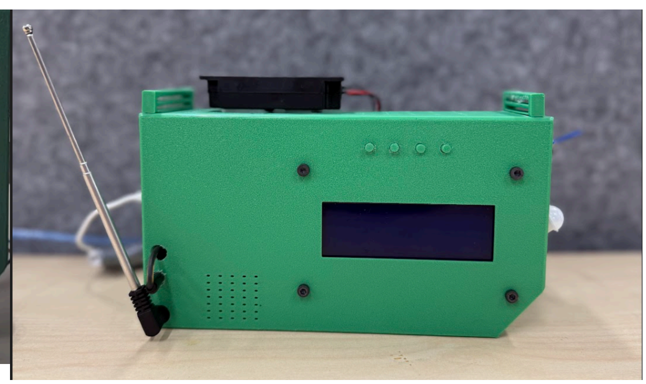
Results



The second prototype does what we set out to prove, it tunes FM, streams Bluetooth music, dimms the LED, and wakes on motion. Voice commands work in a workshop at normal speech volume and menus remain clear on the 20 × 4 LCD under midday sun. Audio output peaks at 86 dB SPL at 0.5 m with the MAX amp and 3-inch driver. Battery life in mixed use is 3 h 40 m on a 2200 mAh pack.

Two items still fall short: the unit needs a stable buck regulator for higher loads, and the case needs a cleaner wire path for a neat finish.

Checklist item	Target	Met?
FM reception 88–108 MHz, seek & step	Seek ±0.1 MHz	Yes
Bluetooth A2DP play/pause/track	Response ≤ 200 ms	Yes
Voice commands (12 mapped)	≥ 90 % hit rate indoors	Yes
Motion-triggered wake	Detect at 2 m, 120° cone	Yes
LED dimmer range	0–100 % in 25 % steps	Yes
Sun-readable display	Text visible at 50 k lx	Yes
Hand-held envelope	≤ 150 mm × 80 mm × 50 mm	Yes
Stable power rail	Ripple ≤ 50 mV at 2 A	No
Clean aesthetic case	No loose wires visible	No



Demonstration

We presented the second prototype during the final studio session. The unit powered up on the first try and loaded the main menu. A classmate used the left and right buttons to cycle through all six modes. The FM tuner locked onto a local station and pushed clear audio through the speaker. Bluetooth pairing with a phone also worked and streamed a short track. The LED dimmed in four steps and then shut off on command. Motion sensing woke the display when a student walked past the desk. Voice commands worked for “Radio mode” and “Next track,” but missed on “Pause music” twice. After ten minutes the screen froze while the speaker kept playing, making it look like the menu still accepted input even though it did not. Cycling power restored control, yet the freeze re-appeared after another short run. The test showed that the concept is sound, but the firmware and wiring must be hardened for field use.

Challenges encountered

- Unstable power rail
  The breadboard supply dropped when the MAX amp drew peaks, causing random resets. A buck module with proper filtering is now required.

- ESP32 board failures
  Two ESP32-S3 boards burned out. One saw a 9 V spike from an unregulated wall adapter; the other failed after repeated brownouts. We replaced the main controller with an Arduino Mega and now use a separate ESP32 only for Bluetooth audio.

- Mixed-voltage load sharing
  Running the LCD, FM tuner, voice module, and Bluetooth sink from one 5 V line overloaded the USB port. A current-rated DC-DC stage is planned to split logic and audio rails.

- Fragile point-to-point wiring
  Jumper wires pulled loose during transport and shorted adjacent rows. Hand-soldered ribbon cables or a small PCB will replace the breadboard.

- Bulk and cable tangle
  The loose harness filled most of the printed shell, blocking airflow and trapping heat. A stacked board layout will cut volume and improve cooling.

- No sealed battery solution
  Swapping AA cells breaks the case seal and risks reverse polarity. A protected Li-ion pack with USB-C charging is the target for the next build.

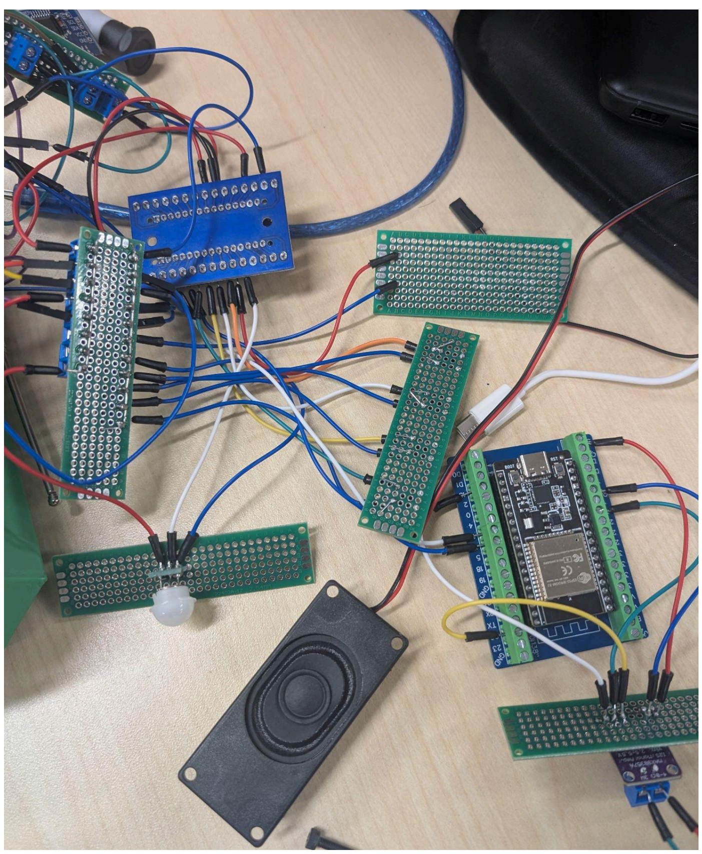
Future Work

The next major step is to move fully to an ESP32-S3 platform. Using only the ESP32 would remove the Arduino Mega and simplify the overall design. This change would cut the wiring burden in half and reduce the number of power domains that need to be managed. A native ESP32 solution would also allow smoother voice command handling and faster Bluetooth responses. The freezing issue we faced during testing is tied to the communication delay between two separate microcontrollers; eliminating the split would clear that bottleneck. A single, stable core would make future software updates easier and allow features like Wi-Fi OTA updates later if needed.

Another important improvement is building a better power delivery system. Right now, the project uses a simple battery pack with no regulation beyond the Arduino's onboard circuits. This system cannot handle fast changes in load, especially when the Bluetooth, amplifier, and LCD modules are active together. We plan to switch to a dedicated power module that provides stable 5 V and 3.3 V outputs with ripple below 50 mV. Including a built-in charger for a Li-ion pack would also make the device rechargeable, reducing user maintenance and keeping the case sealed from dust and moisture. A single USB-C input could handle both charging and optional wall power operation.

The wiring method needs a major redesign to improve stability and compactness. Instead of a loose set of jumper wires and breadboards, we aim to build a double-shielded PCB stack. Key connections would be soldered directly or made with locking terminal blocks to resist vibration and accidental pulls. A small mainboard would host the ESP32, voice module header, and power management, while a daughterboard could carry the audio amplifier and FM tuner. This layered layout would save vertical space, make servicing easier, and allow airflow between the layers. The goal is a design that feels solid in the hand and stands up to long-term field use.

Finally, enclosure and mechanical design must follow the electronics work. The current printed shell is too cramped and forces wires into sharp bends that stress the solder joints. A new enclosure would be custom-fitted around the PCB layout and would allow better routing of cables and ports. Mounting bosses and grommets would fix the boards and major modules securely. Ventilation openings could be added near the amplifier to prevent overheating during high-volume use. A modular back panel could allow battery swaps without opening the entire case. These mechanical updates would complete the transition from a proof-of-concept prototype to a reliable tool.



Conclusion

This project began with a clear goal: to improve on the shortcomings of a common portable radio. By studying the QFX radio's structure and performance, we learned where real-world weaknesses occur. Weak dust protection, poor outdoor visibility, fragile controls, and lack of hands-free features pointed to important design gaps. Our goal was not just to match the original unit but to rethink it for active users in harsher environments. We set targets around control flexibility, ruggedness, and ease of use without increasing bulk. Each step, from early sketches to wiring and firmware, aimed to meet these real needs. The work gave us a practical path to reframe a simple device into something much more field-ready.

The system architecture combined basic building blocks in new ways. A modular design allowed us to manage motion sensing, LED control, FM tuning, Bluetooth streaming, and voice command within a small footprint. The menu system kept the interface clean and easy to learn. The code structure made it easy to add or adjust features during testing. A single LCD showed real-time system state with minimal load on the processor. Each function could operate without disturbing the others, helping the device feel responsive. This modularity will make future expansions, like Wi-Fi or GPS tracking, straightforward to add.

Testing showed that the device delivers most of what was planned. Audio playback, lighting, radio reception, and motion wakeup all function under normal use. Voice commands and Bluetooth response need more polish but show strong potential. The physical build and electrical stability still need major work. The current wiring is too fragile for regular field use, and power distribution is not yet reliable under full load. Addressing these weaknesses will require moving from breadboarded systems to printed circuit boards with built-in protection. The groundwork, however, is now in place to focus on those improvements with clear technical goals.

This project shows that hands-free communication and entertainment tools can be practical even in tough environments. With a compact form, rugged design, and reliable controls, a device like this can serve campers, workers, and responders. The lessons learned here—from stable power management to ergonomic layout—are critical for pushing the project into a final product stage. A cleaner electrical design, a more durable case, and full use of the ESP32's features will complete the transition from prototype to deployable tool. The experience of building, testing, and refining this project strengthens our ability to approach future real-world challenges with clear eyes and strong methods.