# AN INDUSTRIAL TRAINING REPORT

## EMBEDDED TESTING

**Submitted by:**

**NAME : EZHILARASI M**
**REG NO : 513121106025**

**In partial fulfilment for the award of the degree**
**Of**

**BACHELOR OF ENGINEERING**

**in**

**ELECTRONICS AND COMMUNICATION ENGINEERING**

**THANTHAI PERIYAR GOVERNMENT INSTITUTE OF TECHNOLOGY ,VELLORE-02.**

**(AFFILIATED TO ANNA UNIVERSITY – CHENNAI)**

**VAAYUSATRA AEROSPACE PRIVATE LIMITED**

**IITM RESEARCH PARK, CHENNAI-600 113.**

## BONAFIDE CERTIFICATE

This is to certify that the internship report titled **"EMBEDDED TESTING"** is the Bonafide record of work done by **EZHILARASI M (513121106025)** in partial fulfillment of the requirement of the degree in Electronics and Communication Engineering during the year 2024 -2025.

Internship completed for 25 days from **18.07.2024 to 24.08.2024.**

**Mr. JAGADEESH KANNA,**
**M.E.,FOUNDER & CEO**,
Vaayusastra Aerospace Pvt. Ltd,
IITM Research
Park,Tharamani,
Chennai-600113.

**Mr. P. GOWTHAM,**
**R&D ENGINEER,**
Vaayusastra Aerospace Pvt.
Ltd,IITM Research Park,
Tharamani,
Chennai-600113.

Submitted for the Industrial Training Report and Evaluation held on ___________.

# ACKNOWLEDGEMENT

We feel glad to take this opportunity to cordially acknowledge a number of people who provided us a great support during our internship.

We would like to express our deep sense of gratitude to our respected Principal **Dr. P. K. PALANI, B.E (Hons), M.E., Ph.D**., who has bestowed his kind grace and affection of us in accomplishing this internship.

I am highly indebted to pay my sincere gratitude to our Head of the Department **Dr.S.LETITIA,M.E., Ph.D.,** for her guidance, constant supervision and also her support in completing the project.

We would like to thanks our faculty adviser **Dr.B.SENTHIL MURUGAN**, **M.E., Ph.D.,** Assistant Professor, Department of Electronics and Communication Engineering, who guided us throughout the entire phase of this internship. It is his motivation and guidance which made us explore this internship.

We would like to thank **Mr. JAGADEESH KANNA**, Founder & CEO, Vaayusastra Aerospace Pvt. Ltd., for giving us this opportunity and necessary advice and guidance and arrangement of all facilities to provide us wonderful learning experience.

We are highly indebted to express our sincere gratitude to our Training Guide & Coordinator, **Mr.P.GOWTHAM**, for his guidance, constant supervision, and support in completing the training.

Finally, we thank our parents, teaching, non-teaching staff and friends who helped us in completing this internship successfully.

## TABLE OF CONTENTS

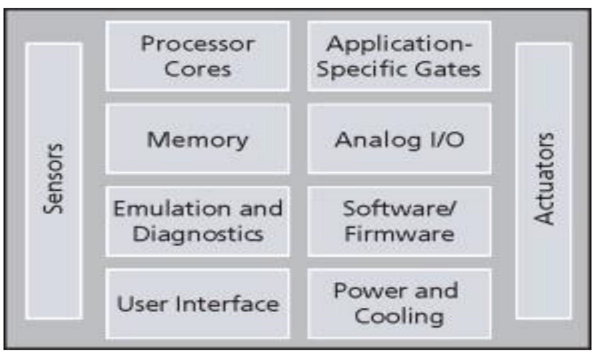
# OVERVIEW OF EMBEDDED SYSTEMS

## 1.1 EMBEDDED SYSTEMS

Embedded means something that is attached to another thing. An embedded system can be thought of as a computer hardware system having software embedded in it. An embedded system can be an independent system or it can be a part of a large system. An embedded system is a microcontroller or microprocessor-based system which is designed to perform a specific task. For example, a fire alarm is an embedded system; it will sense only smoke.

An embedded system has three components:

1. Hardware
2. Software
3. Firmware




## 1.2 CHARACTERISTICS OF EMBEDDED SYSTEMS

- **Specific function:** An embedded system is usually designed for a specific function.
- **Tightly constrained**: An embedded system is tightly resource- and time-constrained. For example, an embedded system has to be fast and tasks-tolerant of slight variations in reaction time (in real- time or near real-time manner), with limited memory and minimum power consumption.

- **Real-time and reactive:** Real-time or near real-time manner has to be served in many environments.

For instance, a global positioning system (GPS) navigator needs to continually provide road and location information, and to send driver alerts to increase situation awareness in a near real-time manner or sometimes real-time manner. Likewise, a car cruise controller is required to continually monitor and react to speed and brake sensors, and also compute the acceleration or deacceleration in a real-time manner. Any delay would make the car out of control, which could give rise to catastrophic results.

- **Hardware/Software codesign:** An embedded system is typically a computer hardware system with software embedded in it. Hardware is designed for performance and security, while software is designed for more features and flexibility.
- **Microprocessor-/Microcontroller-based:** A microprocessor or microcontroller is often deployed at the heart of the embedded system and designed to perform operations.
- **Memory:** A memory is required for an embedded system since programs and operating systems are generally loaded and stored in the memory.
- **Connected peripherals:** Peripherals are needed to connect input and output devices.

## 1.3 CAREERS IN EMBEDDED SYSTEMS

Careers in embedded systems are diverse and offer opportunities in various industries, including automotive, healthcare, consumer electronics, and aerospace. Professionals in this field work on designing, developing, and maintaining both the hardware and software components of systems that perform dedicated functions within larger products. Roles such as Embedded Software Engineer, Embedded Hardware Engineer, and Firmware Engineer are common, each requiring specialized skills like proficiency in C/C++, understanding of microcontrollers, and experience with real-time operating systems (RTOS).

In addition to these core roles, the field of embedded systems also includes positions like System Validation Engineer, IoT Developer, and Embedded Systems Architect, where professionals focus on testing, connectivity, and overall system design.

# COMPANY PROFILE

## 2.1 VAAYUSASTRA AEROSPACE

Vaayusastra Aerospace is an educational and research-focused organization aiming to democratize aerospace education, particularly in underserved regions. Founded on the principles of innovation, accessibility, and practical learning, the company merges artistic methods with scientific rigor to engage students and foster a deep interest in aeronautics. They undertake advanced R&D projects, including work on military telemetry, astronaut gear, and space communication systems. Vaayusastra collaborates with industry veterans and top institutions like IIT, ISRO, and DRDO to drive their initiatives.

The company's mission is to make aerospace education accessible to all, particularly in rural areas, through a unique blend of creative arts and scientific education. By integrating theater arts with practical aeronautical science, Vaayusastra seeks to inspire young minds and encourage them to pursue careers in the aerospace sector. Their innovative approach ensures that even students in remote areas can access high-quality aerospace education.

Vaayusastra's research and development efforts are focused on cutting-edge aerospace technologies, including telemetry in military suits, astronaut suits, wireless communication for aircraft, and CubeSat attitude control. Their work is supported by a team of experts from renowned institutions such as IIT, ISRO, and DRDO, who provide mentorship and guidance. The company's vision is to foster a culture of innovation and hands-on learning in aerospace, bridging the gap between theoretical knowledge and practical application.

## 2.2 VISION OF THE INSTITUTE

The vision of Vaayusatra Aerospace is to be a leading institute in aerospace innovation and education.

1. To Develop new and advanced aerospace technologies.
2. To Promote environmentally friendly practices in aviation.
3. To Provide quality education and training for future aerospace professionals.
4. To Work with industry partners and research organizations.
5. To Enhance the competitiveness of the aerospace sector.

## 2.3 MISSION OF THE INSTITUTE

The mission of Vaayusatra Aerospace is to:

1. To Provide high-quality education and training in aerospace engineering.
2. To Foster innovation by conducting cutting-edge research and development in aerospace technologies.
3. To Promote sustainable practices in the aerospace industry to protect the environment.
4. To Collaborate with industry leaders and academic institutions for knowledge sharing and advancements.
5. To Support entrepreneurship and the growth of startups in the aerospace sector.
6. To Ensure the highest standards of safety and reliability in all aerospace projects.

## 2.4 RESEARCH AND DEVELOPMENT

At Vaayusatra Aerospace, the latest research and development initiatives focus on several cutting-edge areas to advance aerospace technology. The institute is exploring electric and hybrid propulsion systems aimed at reducing emissions and improving fuel efficiency in aircraft. Research into advanced composite materials is underway to create lightweight and durable components that enhance aircraft performance.

Vaayusatra Aerospace is also investigating sustainable aviation fuels (SAFs) to minimize the environmental impact of air travel and is working on eVTOL (electric vertical takeoff and landing) vehicles to address urban transportation challenges. Innovations in noise reduction technologies aim to mitigate noise pollution from aircraft, particularly in urban settings. Furthermore, the institute is implementing smart maintenance systems utilizing IoT and advanced sensors for real-time monitoring of aircraft, enhancing safety and maintenance efficiency. Finally, research into hypersonic flight technologies is being pursued, focusing on speed and efficiency for both commercial and military applications.

# DEPARTMENT PROFILE

## 3.1 AEROSPACE ENGINEERING DEPARTMENT

This department focuses on the design, development, and testing of aircraft and spacecraft. It integrates principles from various engineering disciplines to create innovative and efficient aerospace solutions.

## 3.2 PROPULSION SYSTEMS DEPARTMENT

Specializing in the research and development of advanced engine technologies, this department aims to improve propulsion efficiency and reduce emissions in aviation.

## 3.3 AVIONICS DEPARTMENT

Experts in this department work on the electronic systems used in aviation, including navigation, communication, and control systems, ensuring that aircraft operate safely and effectively.

## 3.4 MATERIALS SCIENCE AND ENGINEERING DEPARTMENT

This department is dedicated to studying and developing advanced materials that enhance the strength, durability, and performance of aerospace components.

## 3.5 AERODYNAMICS DEPARTMENT

Researchers in this department analyze the behavior of air as it interacts with solid objects, optimizing aircraft designs for better performance and fuel efficiency.

## 3.6 RESEARCH AND DEVELOPMENT DEPARTMENT

Engaging in innovative projects, this department focuses on pushing the boundaries of aerospace technology through cutting-edge research and experimentation.

# DAILY REPORT

## ORIENTATION PROGRAM 18/07/2024

During the orientation program, the class timings were discussed, and the importance of a well-crafted resume was emphasized. Feedback was provided on our resumes, and a resume-building class was scheduled for those interested in improving their skills. Key instructions for creating an effective resume includes ensuring your full name is clearly stated, providing a short address, avoiding photos, listing completed projects, including your email and LinkedIn profile, limiting extracurricular skills, adding relevant certifications, and keeping the resume to one page. Additionally, a special resume class was planned for those who did not perform well in the initial evaluation. These guidelines and additional support aimed to help us present our qualifications and experiences effectively.

## BASICS OF C PROGRAMMING 22/07/2024

Embedded C is essential for controlling hardware efficiently in embedded systems. It builds on standard C with features suited for real-time, low-level programming.

- **Statements and Loops:** These control the flow of operations and manage repetitive tasks, like continuously reading sensor data or controlling an actuator.
- **Functions:** Functions help modularize code, making complex embedded systems easier to develop and maintain by dividing tasks into smaller, reusable blocks.
- **Header Files:** They enable code reuse and manage hardware-specific instructions, simplifying development by organizing function declarations, macros, and constants.
- **Delays:** Delays are used to control timing-sensitive operations, such as managing communication protocols or synchronizing tasks, which are crucial for real-time performance.

## DELAY PROGRAMS AND ONLINE COMPILATION 23/07/2024

**Delay Generation:** Delays are essential for controlling timing in embedded systems. They can be implemented through:

- **Software Loops:** Easy to set up and use but may be imprecise as they depend on processor speed and can be affected by system interruptions. This method involves simple counting loops to introduce delays.
- **Hardware Timers:** More precise and reliable. These use the microcontroller's internal timers to count clock cycles or generate interrupts, ensuring accurate delays even under varying operational conditions.

**Online Compilation:** Online compilation tools such as **OnlineGDB** and **Programiz** streamline embedded C development by providing web-based environments. These platforms enable users to write, compile, and debug code directly from a browser, offering real-time feedback and reducing the need for local setup. This approach simplifies the development process and allows for easier testing and debugging of embedded applications.

## TIMERS AND COUNTER PROGRAMS 24/07/2024

**Timers:** Timers are crucial for precise timing and scheduling in embedded systems. They work by counting clock pulses from a timer source, allowing you to:

- **Generate Delays:** Create accurate time delays by counting up to a preset value and triggering interrupts when the count is reached.
- **Schedule Events:** Trigger periodic tasks or events at regular intervals, essential for time-sensitive applications.

**Counters:** Counters are used to monitor and measure external events or pulses. They help:

- **Count Events:** Track occurrences of external signals or pulses, such as counting the number of rotations or button presses.
- **Measure Frequency:** Determine the frequency of input signals by counting pulses over a specific time period, providing insight into signal characteristics.

## DELAY CONCEPTS AND ONLINE WOKWI SIMULATION 25/07/2024

**Delay Concepts:**

- **Software Delays:** Achieved with loops, but less precise due to processor speed.
- **Hardware Delays:** Utilizes timers or counters for more accurate timing.

**Basic Programs:**

1. **LED Blinking (Wokwi Simulation):**
   - **Objective:** Blink an LED using delays to control timing.
2. **Number Counting on a 7-Segment Display:**
   - **Objective:** Show numbers in sequence with controlled delays.
3. **Sensor Integration:**
   - **Objective:** Read and process sensor data with timed intervals.

These examples demonstrate practical applications of delay concepts and sensors.

## MICROCONTROLLERS AND MICROPROCESSORS 26/07/2024

**Microcontrollers:** Microcontrollers are integrated circuits with a processor, memory, and I/O on one chip, ideal for control tasks. Examples include:

- **8051:** An 8-bit microcontroller with 4 KB ROM and 128 bytes RAM, used for simple control applications.
- **8055:** An enhanced 8051 with additional RAM and I/O, supporting more complex functions.

**Microprocessors:** Microprocessors focus on processing tasks and need external components for memory and I/O. They are suited for complex applications requiring more processing power.

Microcontrollers are used for specific tasks, while microprocessors handle more demanding processing needs.

## SOC, MEMORY MANAGEMENT, RTOS 29/07/2024

**System on Chip (SoC):** SoCs integrate processors, memory, and peripherals into a single chip, enhancing performance and reducing size and power consumption. They are used in complex applications requiring compact designs.

**Memory Management:**

- **Dynamic Allocation:** Manages memory at runtime with functions like malloc and free.
- **Memory Protection:** Prevents interference between tasks, ensuring system stability and security.

**Real-Time Operating System (RTOS):** RTOS is crucial for applications with strict timing requirements, offering features like precise task scheduling and efficient inter-task communication.

**LCD Simulation in Wokwi:** Wokwi simulates LCD displays, allowing users to test and debug LCD code effectively. This simulation supports developing and validating display interactions without the need for physical hardware.

## ELECTRONIC COMPONENTS 08/08/2024

**Diodes:** Diodes allow current to flow in one direction, essential for rectification and circuit protection.

**Transistors:** Transistors are active devices used for amplification and switching, with types including BJTs and FETs.

**Resistors:** Resistors limit current flow and divide voltage, crucial for signal control in circuits.

**Active vs. Passive Components:**

- **Active Components:** Include diodes and transistors, capable of amplification and switching.
- **Passive Components:** Include resistors, capacitors, and inductors, modifying signal behavior without amplification.

These components are fundamental for designing and managing electronic circuits.

## LOGIC GATE SIMULATION USING TINKERCAD 09/08/2024

**NOT Gate:** The NOT gate, also known as an inverter, outputs the opposite of its input signal. If the input is high (1), the output will be low (0), and if the input is low (0), the output will be high (1). In Tinkercad, you can simulate a NOT gate to observe this inversion effect, which is fundamental in digital logic circuits. This gate is crucial for implementing functions that require signal negation and for creating more complex logical operations.

**Buffer Gate:** A buffer gate transfers its input signal directly to its output without modification, effectively amplifying or isolating the signal. It is used to strengthen weak signals or to separate different stages of a circuit to prevent interference. Simulating a buffer in Tinkercad allows you to see how it preserves the signal's integrity while providing the necessary drive capability for subsequent circuit stages. This is essential in ensuring that signals remain strong and clear throughout the circuit. Simulating these gates in Tinkercad provides a hands-on way to understand their fundamental operations and their roles in managing and processing signals.

## LOGIC GATE SIMULATION USING TINKERCAD 10/08/2024

**OR Gate:** The OR gate outputs high (1) if at least one input is high. In Tinkercad, simulate this gate to see how it combines signals, providing a high output when any input is active.

**NOR Gate:** The NOR gate outputs high (1) only when all inputs are low, inverting the OR gate's output. Simulating it in Tinkercad shows how it negates the OR function.

**NAND Gate:** The NAND gate, the inverse of the AND gate, outputs high (1) unless all inputs are high. Simulating it in Tinkercad demonstrates how it performs a negated AND operation.

**AND Gate:** The AND gate outputs high (1) only when all inputs are high. In Tinkercad, simulate this gate to understand how it requires all inputs to be active for a high output.

## SENSORS AND ACTUATORS 12/08/2024

## Sensors:

- Ultrasonic Sensor: Measures distance by timing the echo of ultrasonic waves. Used for obstacle detection and distance measurement.
- IR Sensor: Detects objects and measures distance using infrared light. Common in object detection and remote controls.
- Moisture Sensor: Measures soil or material moisture. Useful in agriculture for monitoring soil conditions.
- Temperature Sensor: Measures temperature through thermocouples or thermistors. Essential for climate control.
- Humidity Sensor: Measures air humidity. Key for weather stations and humidity control.
- Blood Pressure Sensor: Monitors blood pressure. Used in medical diagnostics.

## Actuators:

- Motor: Converts electrical energy into mechanical motion. Used in robotics and automotive systems.
- Servo: Provides precise control of position. Common in robotics and automation.
- Solenoid: Converts electrical energy into linear motion. Used in locks and valves.

These components are vital for systems that interact with and control the physical world.

## SENSOR TESTING 13/08/2024

**Objective:** Get practical experience with Arduino and ESP32 by configuring pins, blinking LEDs, and testing sensors.

**Activities:**

1. **Pin Configuration:**
   - **Arduino:** Identify digital/analog pins; use pin 13 for onboard LED tests.
   - **ESP32:** Learn GPIO pin functions; test onboard LED.
2. **Onboard LED Blinking:**
   - **Arduino:** Write code to blink the LED on pin 13.
   - **ESP32:** Create similar code for the onboard LED.
3. **Breadboard LED Blinking:**
   - Connect an external LED to a digital pin through a resistor.
   - Write code to blink the external LED.
4. **Sensor Testing:**
   - Connect sensors to Arduino/ESP32 and check readings on the serial monitor.

## TESTING WITH ARDUINO NANO 14/08/2024

**Objective:** Practice basic functionality by implementing ultrasonic sensor testing and working with the ESP32 camera module.

**Activities:**

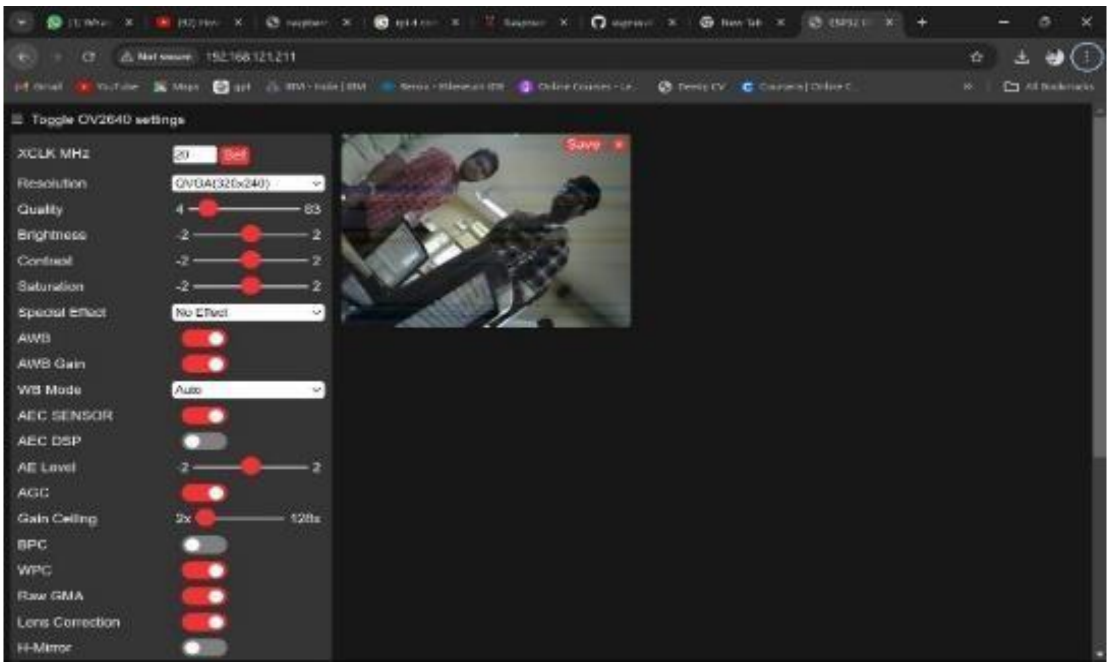
1. ESP32 Camera Module:
   - Setup: Connect the ESP32 camera module to the Arduino Uno (or directly to a compatible ESP32 board) according to the module's pinout.

- Programming: Write code to initialize the camera and capture images. Ensure the module is correctly interfaced and can handle image capture.

2. Ultrasonic Sensor Testing:

    - Setup: Connect the ultrasonic sensor's trig and echo pins to digital pins on the Arduino Uno.

    - Programming: Use a sketch to send pulses from the trig pin and measure the echo time to calculate distance. Display the distance readings on the Serial Monitor.




## TESTING WITH ESP32 16/08/2024

**Objective:** Learn to interface the ESP32 with BMP180 and DHT11 sensors to measure environmental data.

**Activities:**

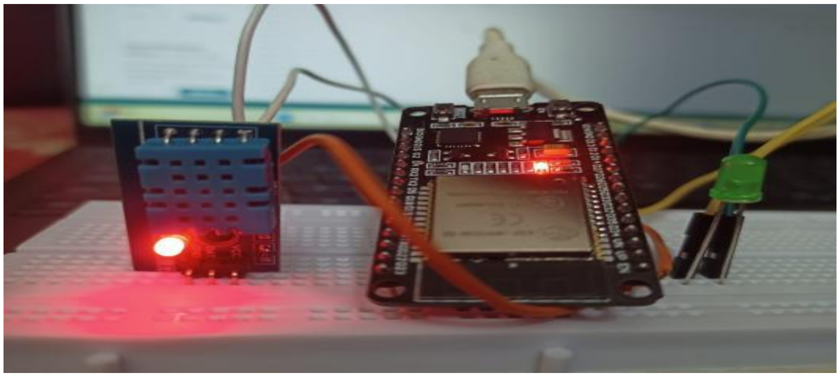
1. BMP180 Sensor:

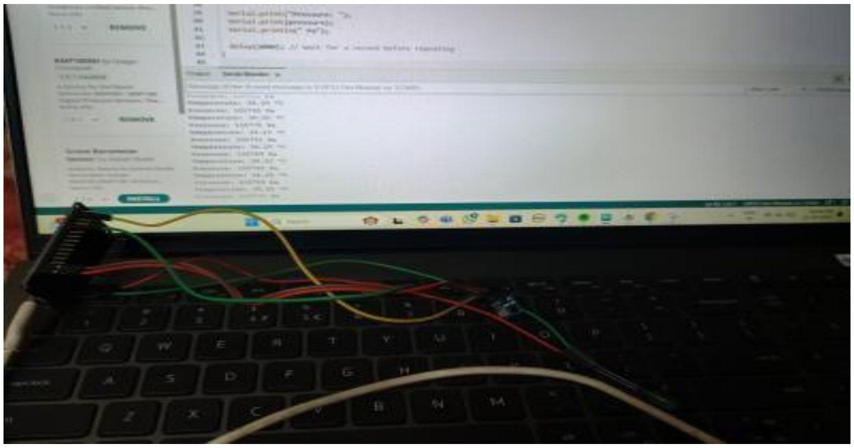
    - Setup: Connect BMP180 to ESP32 using I2C (SDA to A4, SCL to A5).

- Programming: Develop a sketch to read temperature and pressure from the BMP180 and display the results on the Serial Monitor using a suitable library.

2. DHT11 Sensor:

   - Setup: Connect DHT11 to a digital pin on ESP32.
   - Programming: Write code to read temperature and humidity from the DHT11 sensor and print the data to the Serial Monitor.



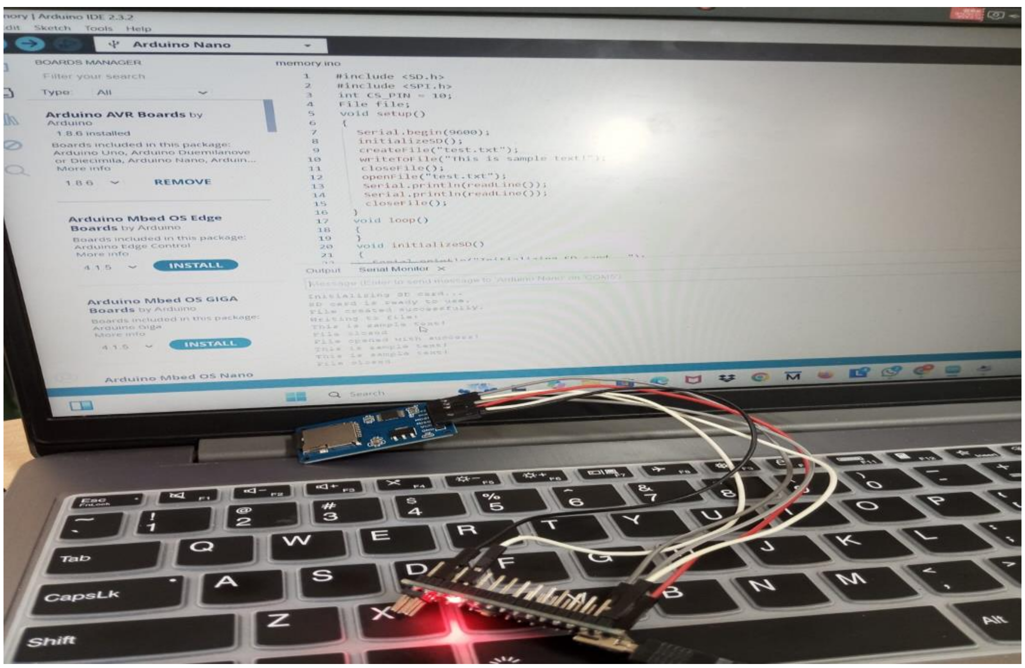


## INTERFACING SDCARD MODULE WITH ARDUINO 17/08/2024

### Activities:

**SD Card Module:**

- Setup: Connect the SD card module to the Arduino Nano by wiring VCC to 5V (or 3.3V if required), GND to GND, MISO to the Nano's MISO pin, MOSI to MOSI, SCK to SCK, and CS to a digital pin (e.g., D4). Ensure that connections are secure and properly aligned to avoid communication issues.
- Programming: Write a sketch to initialize the SD card module and perform basic file operations, such as creating, reading, and writing files. Include error handling to manage any issues during initialization or file operations. Use the Serial Monitor to display status messages and data, allowing you to verify that the SD card is functioning correctly.

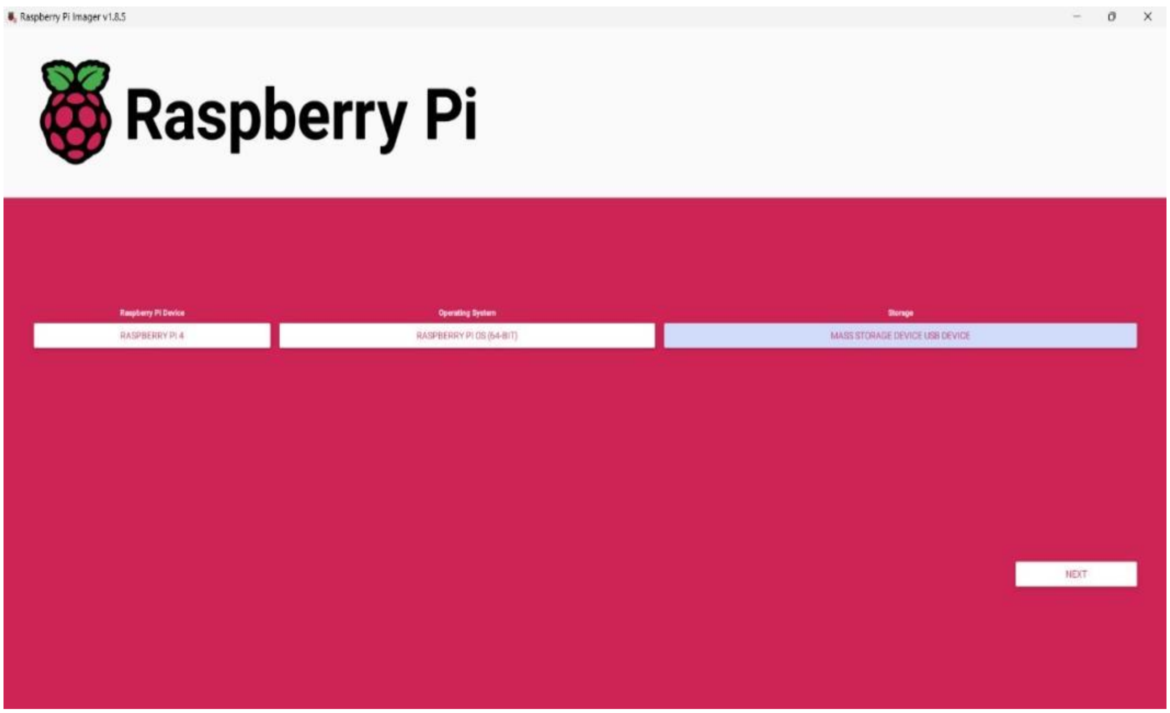


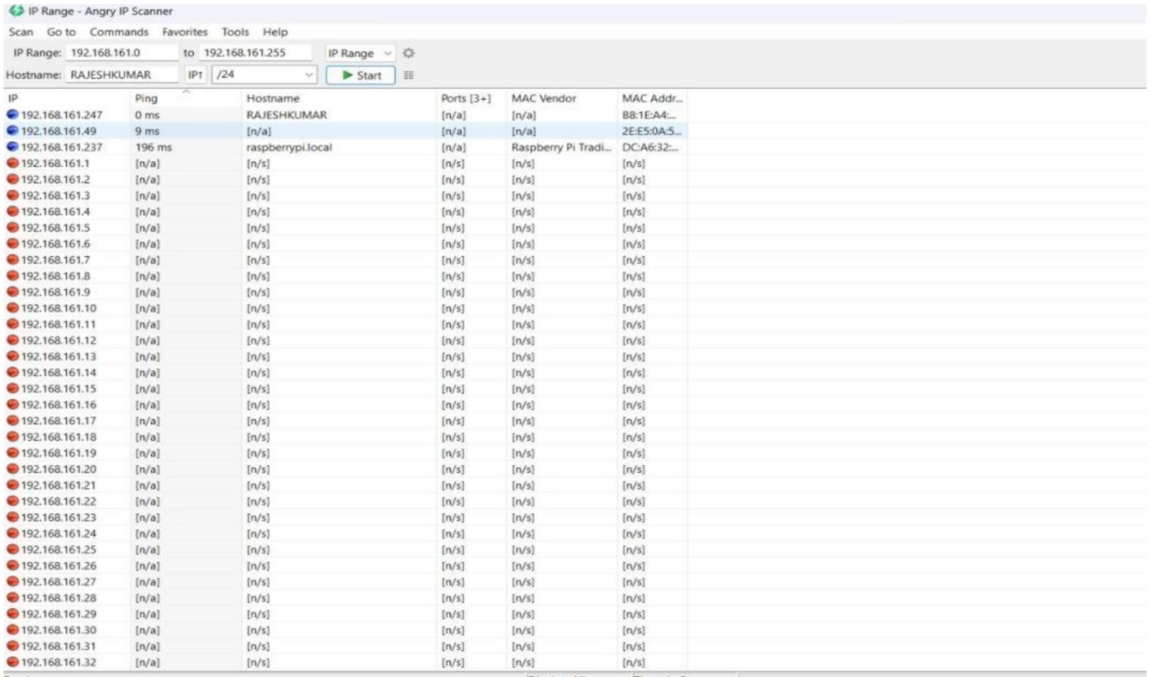
## RASPBERRY PI INTRODUCTION AND INSTALLATION 19/08/2024

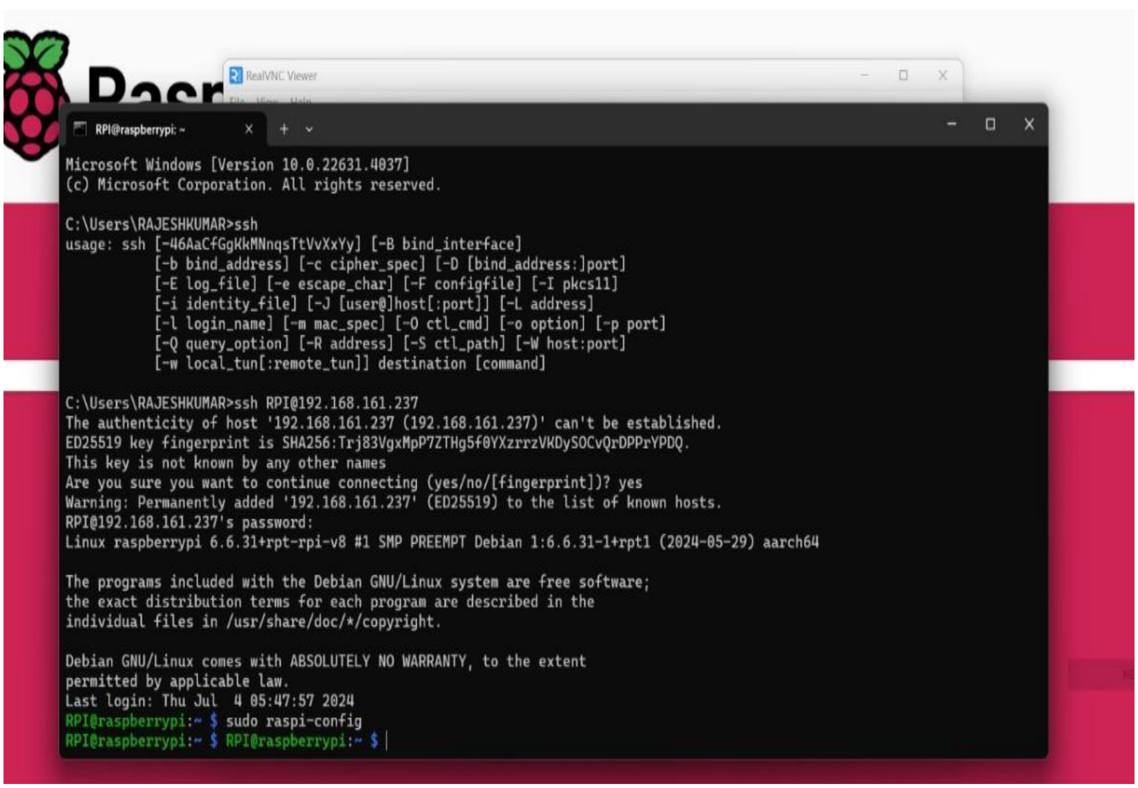
To set up your Raspberry Pi, start by downloading and installing Raspberry Pi Imager from the official website. Once installed, prepare your storage device by inserting the USB drive or microSD card into your computer. Open Raspberry Pi Imager and choose the operating system appropriate for your model, like “Raspberry Pi (64-bit)” for the Raspberry Pi 4.

Select your storage device from the available options and, if desired, configure advanced settings such as hostname, username, password, Wi-Fi network, and SSH. Click “Write” to start the installation process, then wait for it to complete.

After writing the OS, install RealVNC Viewer, input your Raspberry Pi’s IP address, and log in with the credentials set earlier. For additional configuration, use SSH by running ssh rpi@your_ip in your terminal, enter the password, and run sudo raspi-config to adjust settings. Complete the setup by following the prompts in raspi-config and rebooting if needed.




IP Range - Angry IP Scanner
Scan Go to Commands Favorites Tools Help
IP Range: 192.168.161.0 to 192.168.161.255 IP Range
Hostname: RAJESHKUMAR IP↑ /24 Start
IP Ping Hostname Ports [3+] MAC Vendor MAC Addr...
192.168.161.247 0 ms RAJESHKUMAR [n/a] [n/a] B8:1E:A4:...
192.168.161.49 9 ms [n/a] [n/a] [n/a] 2E:E5:0A:5...
192.168.161.237 196 ms raspberrypi.local [n/a] Raspberry Pi Tradi... DC:A6:32:...
192.168.161.1 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.2 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.3 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.4 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.5 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.6 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.7 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.8 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.9 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.10 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.11 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.12 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.13 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.14 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.15 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.16 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.17 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.18 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.19 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.20 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.21 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.22 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.23 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.24 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.25 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.26 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.27 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.28 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.29 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.30 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.31 [n/a] [n/s] [n/s] [n/s] [n/s]
192.168.161.32 [n/a] [n/s] [n/s] [n/s] [n/s]


RealVNC Viewer
RPI@raspberrypi: ~
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.
C:\Users\RAJESHKUMAR>ssh
usage: ssh [-46AaCfGgKkMNnqsTtVvXxYy] [-B bind_interface]
[-b bind_address] [-c cipher_spec] [-D [bind_address:]port]
[-E log_file] [-e escape_char] [-F configfile] [-I pkcs11]
[-i identity_file] [-J [user@]host[:port]] [-L address]
[-l login_name] [-m mac_spec] [-O ctl_cmd] [-o option] [-p port]
[-Q query_option] [-R address] [-S ctl_path] [-W host:port]
[-w local_tun[:remote_tun]] destination [command]
C:\Users\RAJESHKUMAR>ssh RPI@192.168.161.237
The authenticity of host '192.168.161.237 (192.168.161.237)' can't be established.
ED25519 key fingerprint is SHA256:Trj83VgxMpP7ZTHg5f0YXzrrzVKDySOCvQrDPPrYPDQ.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.161.237' (ED25519) to the list of known hosts.
RPI@192.168.161.237's password:
Linux raspberrypi 6.6.31+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.6.31-1+rpt1 (2024-05-29) aarch64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Jul 4 05:47:57 2024
RPI@raspberrypi:~ $ sudo raspi-config
RPI@raspberrypi:~ $ RPI@raspberrypi:~ $

## PROJECT DISCUSSION AND SELECTION 20/08/2024

**Objective:** Develop a system that captures images, extracts text, and converts it to speech for accessibility or automation.

**Components:**

- Camera Module: Captures images (e.g., ESP32 camera, Raspberry Pi camera).
- Image Processing: Use Tesseract for OCR to convert images to text.
- Text-to-Speech (TTS): Convert text to speech with espeak or Festival.

**Setup:** Connect the camera to a microcontroller or Raspberry Pi, write scripts for image capture, text extraction, and speech synthesis.

Considerations:

- Accuracy: Ensure precise text extraction and clear speech output.
- Hardware: Use suitable components within budget.

**Next Steps:** Define the project scope, prototype, and refine based on testing and feedback.

## TESTING OF CAMERA MODULE AND SPEAKER 21/08/2024

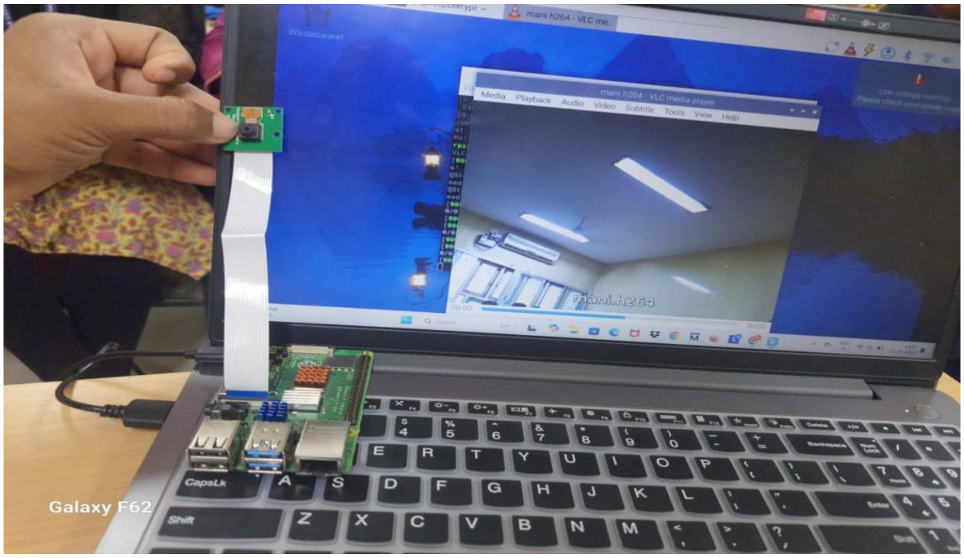
**Camera Module:**

1. **Update Packages:** Run sudo apt update and sudo apt upgrade.
2. **Install Utilities:** Install libcamera with sudo apt install libcamera-apps.
3. **Enable Camera:** Use sudo raspi-config to enable the camera under Interface Options.
4. **Verify Camera:** Check with libcamera-hello.
5. **Record Video:** Use libcamera-vid -t 10000 -o video.h264 for a 10-second video.
6. **View Video:** Open with VLC using vlc video.h264.
7. **Capture Images:** Use libcamera-still -o image.jpg.

**Speaker:**

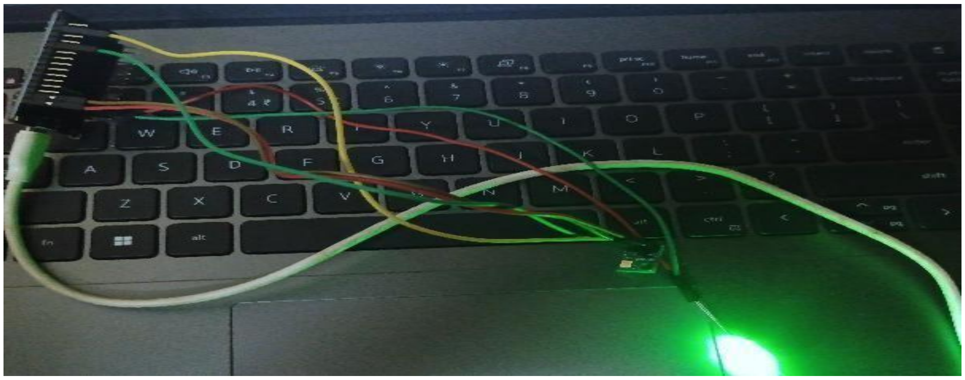
- **Setup:** Connect the speaker and test audio playback with TTS commands (e.g., espeak).
- **Verify Output:** Ensure clear sound and adjust settings as needed.

**Integration:** Test the full process: image capture, text extraction, and speech output.




## BAROMETRIC PRESSURE SENSOR:

BMP180 is a digital sensor used to measure atmospheric pressure and temperature.When connected to an ESP32 microcontroller via the I2C interface.

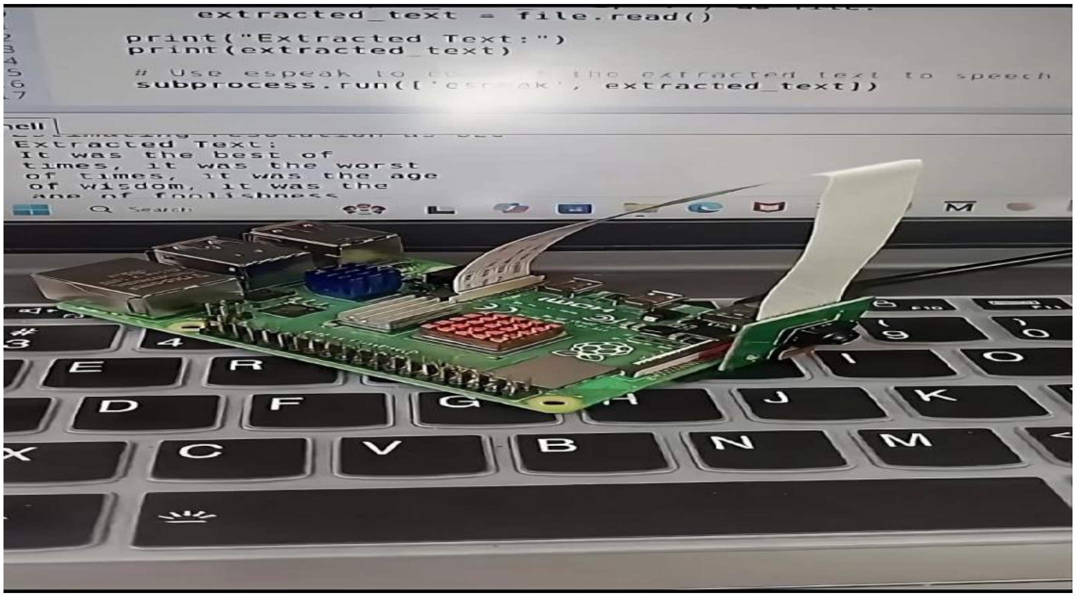


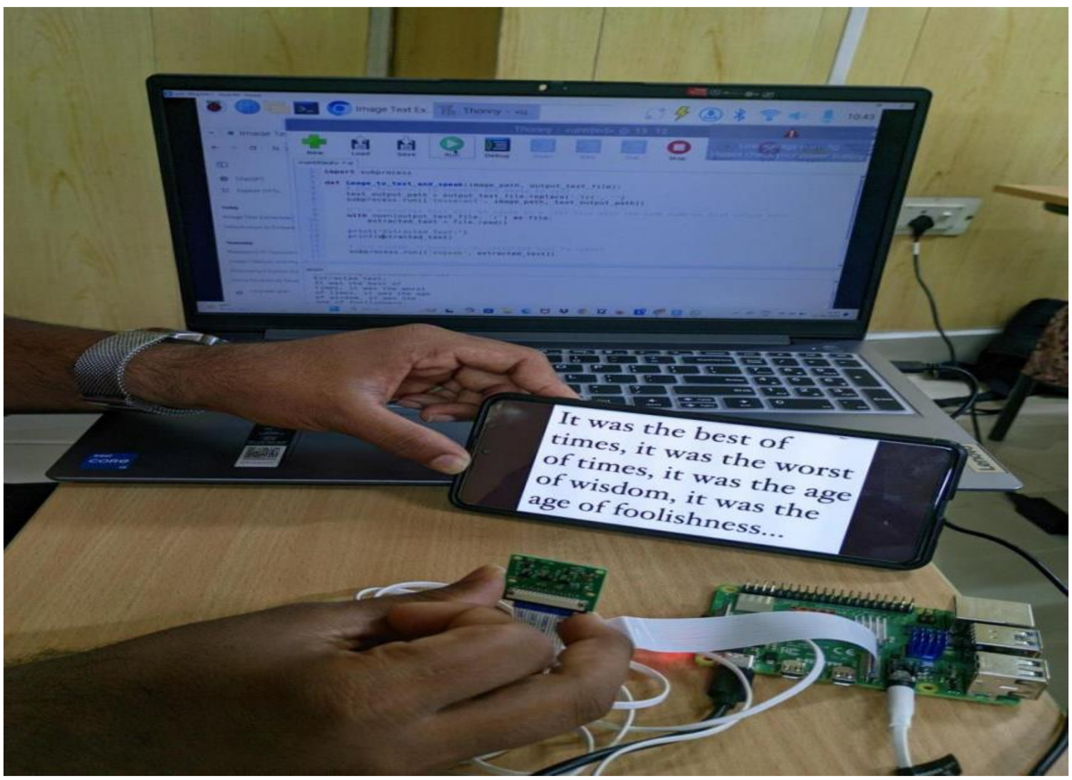
**FINAL PROJECT: IMAGE TEXT-TO-SPEECH CONVERSION FOR THE VISUALLY IMPAIRED 22/08/2024**

**Objective:** Develop a system that captures an image, extracts text from it, and converts the text to speech to assist visually impaired individuals.

**Steps:**

1. Capture Image:
   - Use a camera module to capture a photo of the text.
2. Extract Text:
   - Process the image using Optical Character Recognition (OCR) with Tesseract or a similar tool to convert the text in the image to digital format.
3. Convert Text to Speech:
   - Use text-to-speech (TTS) software like espeak or Festival to read the extracted text aloud.
4. Integration and Testing:
   - Combine all components to ensure the system works seamlessly from image capture to audio output.
   - Test the system with different text images to verify accuracy and clarity.
5. User Interface:
   - Create a simple interface (e.g., a button) to trigger the image capture and processing, making it user-friendly for visually impaired users.


extracted_text = file.read()
print("Extracted Text:")
print(extracted_text)
Extracted Text:
It was the best of
times, it was the worst
of times, it was the age
of wisdom, it was the


It was the best of
times, it was the worst
of times, it was the age
of wisdom, it was the
age of foolishness...

## PROJECT TESTING AND DOCUMENTATION 23/08/2024

**Testing:**

- Functional: Verify each component (camera, OCR, TTS) and the integrated system for accuracy and reliability.
- Performance: Assess response times and system performance under various conditions.
- User: Test with actual users (e.g., visually impaired) to ensure usability and gather feedback.

**Documentation:**

- Overview: Summarize project goals and functionalities.
- Architecture: Describe hardware and software setup.
- Code: Include comments and a summary of the code.
- User Guide: Provide setup and usage instructions.
- Testing Results: Document findings and issues.
- Improvements: Suggest future enhancements.

## PROJECT AND REPORT SUBMISSION 24/08/2024

**Project Submission:**

- **Compile All Components:** Ensure all hardware, software, and documentation are complete and organized.
- **Prepare a Presentation:** Summarize key aspects of the project, including objectives, implementation, and results.
- **Submit:** Follow the guidelines for submission, including deadlines and format requirements.

# CASE STUDY

# ASSISTIVE TECHNOLOGY FOR VISUAL IMPAIRMENT - ADAPTATIONS IN HOUSEHOLD APPLIANCES

## Introduction:

Visual impairment significantly affects daily activities, and household tasks such as laundry can be particularly challenging without proper assistance. Technological advancements in embedded systems have led to the development of assistive devices that help visually impaired individuals perform these tasks more independently. This case study focuses on an embedded system designed to assist users in operating a washing machine, providing features like real-time color recognition and voice synthesis to guide users through the process.

The primary objective of the system is to ensure that users can sort their laundry by color and operate the washing machine without requiring external help. This system's design leverages an ARM STR711FR0 microcontroller, which serves as the core processing unit, managing inputs from sensors and driving the output devices that deliver auditory feedback to the user.

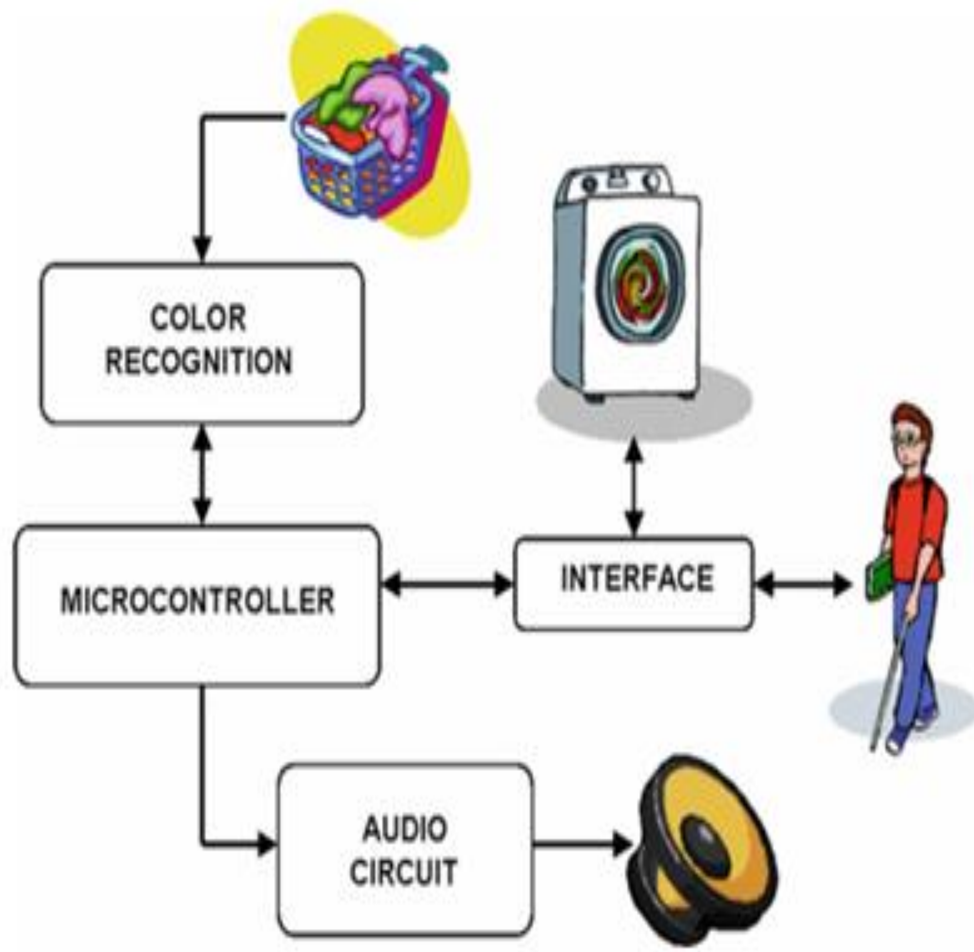
## System Overview:

The embedded system integrates several components, each chosen for its specific role in making the washing machine accessible to visually impaired users. The key components include:

Microcontroller*: The ARM STR711FR0 microcontroller is selected for its balance between performance and resource availability, with 16kB of RAM and 64kB of flash memory. It is responsible for processing data from the color sensor and controlling the voice synthesis subsystem.

Color Sensor: The system uses a TCS230 color sensor, which can detect the red, green, and blue components of reflected light. This sensor is essential for distinguishing between different colors of clothing, ensuring that users do not mix incompatible colors in the wash.

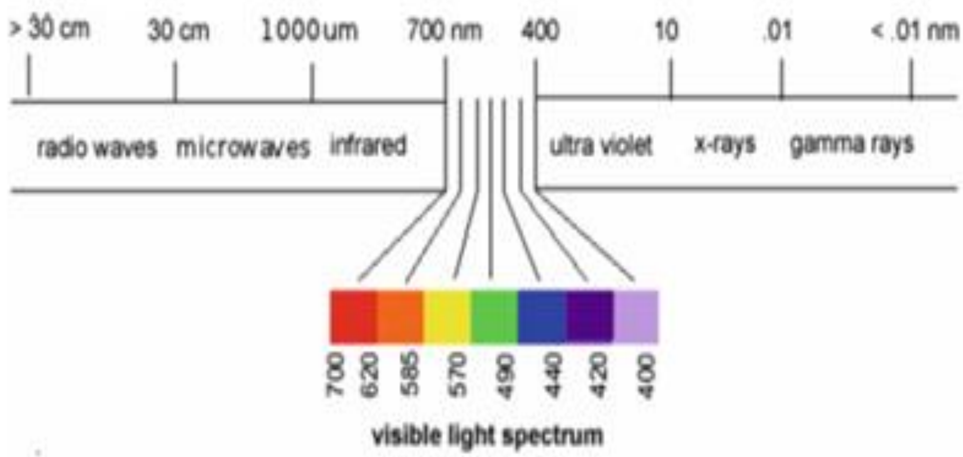
Voice Synthesis: The auditory feedback mechanism is built on phonetic voice synthesis, allowing the system to provide real-time, spoken instructions to the user, thus making the washing machine's operation more intuitive.



## Color Recognition:

The device employs the HSV color model, which is more suited for color recognition tasks than the traditional RGB model due to its separation of chromatic content (hue) from intensity (saturation and value). This model allows the system to more accurately distinguish between different colors, a crucial feature for users who need to prevent color mixing during laundry.

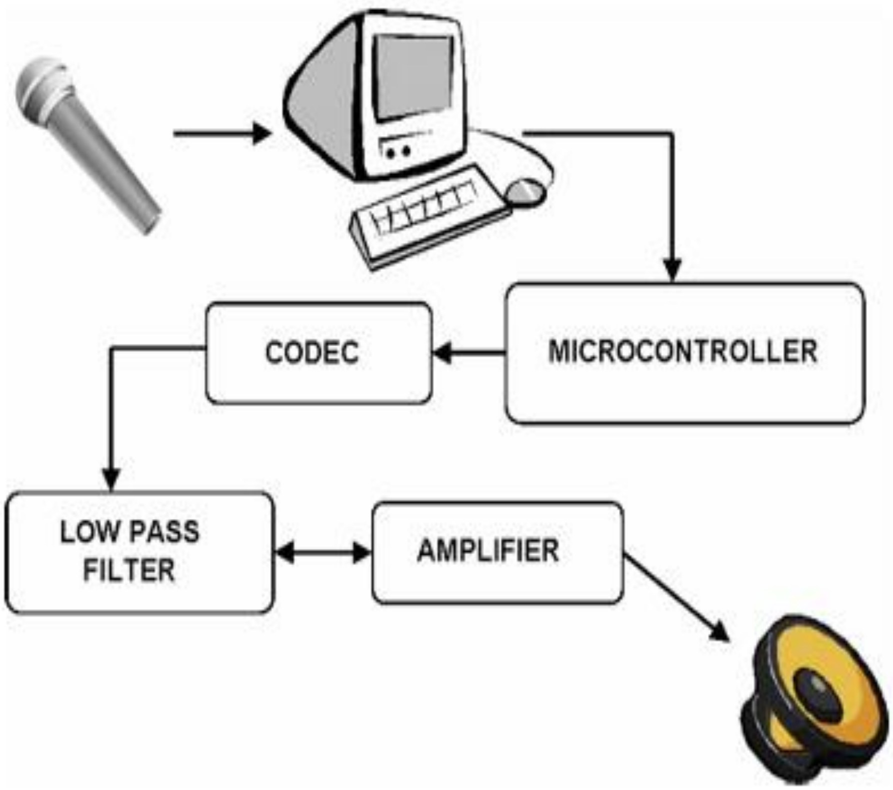
A TCS230 color sensor is used to measure the red, green, and blue components of the reflected light from clothing items. The microcontroller converts the sensor's RGB outputs into the HSV model and matches the detected hue against a predefined color sample set.




## Voice Synthesis:

Voice synthesis is implemented using pre-recorded messages and phonetic synthesis. Phonetic synthesis was chosen for its memory efficiency, allowing the system to generate a wide range of spoken feedback without consuming excessive memory space.

The microcontroller retrieves and processes audio data from external flash memory, converting it into an analog signal via a codec for playback. This system provides real-time, clear auditory instructions and feedback to the user, improving the accessibility of the washing machine.




## Result:

This case study demonstrates the successful integration of color recognition and voice synthesis in an embedded system designed for visually impaired users. The careful selection of components and algorithms results in a cost-effective, reliable device that significantly enhances the usability of a standard household appliance.

# CONCLUSION

The internship at Vaayusastra Aerospace, focused on embedded testing, offered a unique opportunity to apply theoretical knowledge in a practical setting. During the course of the internship, I gained hands-on experience in testing embedded systems, understanding the critical role they play in aerospace applications. The exposure to real-world projects, coupled with an in-depth understanding of testing methodologies, enhanced my skills in debugging, validating, and ensuring the reliability of embedded systems.

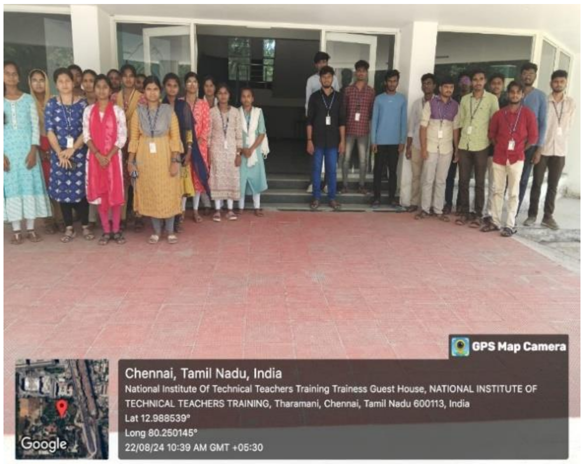
This experience not only deepened my technical expertise but also honed my problem-solving abilities and attention to detail. Overall, the internship has been instrumental in shaping my understanding of embedded systems testing and has significantly contributed to my professional growth in this field.

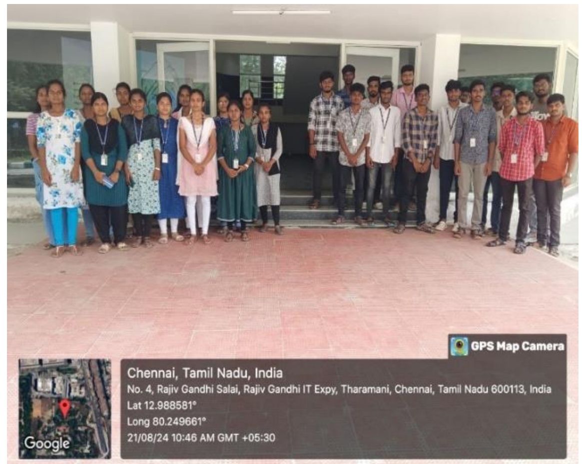
By combining Arduino's real-time control with Raspberry Pi's processing capabilities, the embedded system can leverage the strengths of both platforms. This integration allows for a scalable and modular approach, where Arduino handles tasks that require immediate responsesand physical interactions, while Raspberry Pi manages tasks that demand higher computationalresources and advanced functionalities.

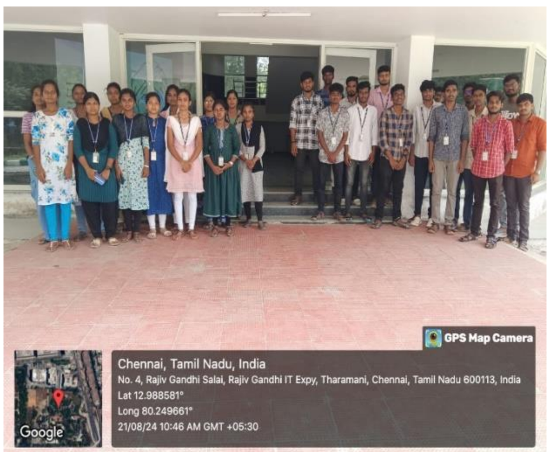
# REFERENCES

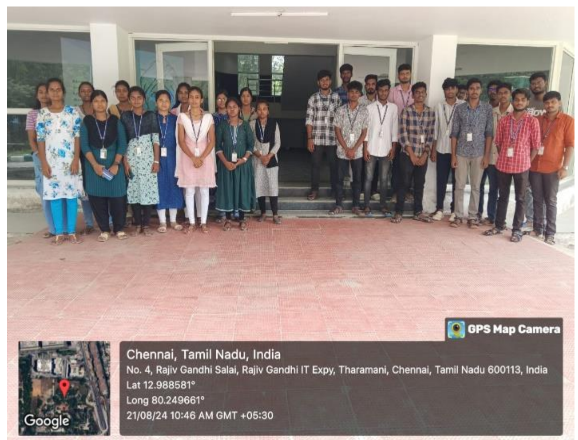

https://www.arduino.cc/reference/en/

https://www.raspberrypi.com/documentation/https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-reference/api-conventions.html


GPS Map Camera
Chennai, Tamil Nadu, India
National Institute Of Technical Teachers Training Trainess Guest House, NATIONAL INSTITUTE OF TECHNICAL TEACHERS TRAINING, Tharamani, Chennai, Tamil Nadu 600113, India
Lat 12.988539°
Long 80.250145°
22/08/24 10:39 AM GMT +05:30
Google


GPS Map Camera
Chennai, Tamil Nadu, India
No. 4, Rajiv Gandhi Salai, Rajiv Gandhi IT Expy, Tharamani, Chennai, Tamil Nadu 600113, India
Lat 12.988581°
Long 80.249661°
21/08/24 10:46 AM GMT +05:30
Google


GPS Map Camera
Chennai, Tamil Nadu, India
No. 4, Rajiv Gandhi Salai, Rajiv Gandhi IT Expy, Tharamani, Chennai, Tamil Nadu 600113, India
Lat 12.988581°
Long 80.249661°
21/08/24 10:46 AM GMT +05:30
Google


GPS Map Camera
Chennai, Tamil Nadu, India
No. 4, Rajiv Gandhi Salai, Rajiv Gandhi IT Expy, Tharamani, Chennai, Tamil Nadu 600113, India
Lat 12.988581°
Long 80.249661°
21/08/24 10:46 AM GMT +05:30
Google


GPS Map Camera
Chennai, Tamil Nadu, India
National Institute Of Technical Teachers Training Trainess Guest House, NATIONAL INSTITUTE OF TECHNICAL TEACHERS TRAINING, Tharamani, Chennai, Tamil Nadu 600113, India
Lat 12.988547°
Long 80.250139°
22/08/24 03:58 PM GMT +05:30
Google


GPS Map Camera
Chennai, Tamil Nadu, India
National Institute Of Technical Teachers Training Trainess Guest House, NATIONAL INSTITUTE OF TECHNICAL TEACHERS TRAINING, Tharamani, Chennai, Tamil Nadu 600113, India
Lat 12.988547°
Long 80.250139°
22/08/24 03:58 PM GMT +05:30
Google


GPS Map Camera
Chennai, Tamil Nadu, India
49, T. Nagar, Chennai, Tamil Nadu 600017, India
Lat 12.989403°
Long 80.249011°
17/08/24 10:24 AM GMT +05:30
Google


GPS Map Camera
Chennai, Tamil Nadu, India
National Institute Of Technical Teachers Training Trainess Guest House, NATIONAL INSTITUTE OF TECHNICAL TEACHERS TRAINING, Tharamani, Chennai, Tamil Nadu 600113, India
Lat 12.988547°
Long 80.250139°
22/08/24 03:58 PM GMT +05:30
Google


GPS Map Camera
Chennai, Tamil Nadu, India
No. 4, Rajiv Gandhi Salai, Rajiv Gandhi IT Expy, Tharamani, Chennai, Tamil Nadu 600113, India
Lat 12.988581°
Long 80.249661°
21/08/24 10:46 AM GMT +05:30
Google


GPS Map Camera
Chennai, Tamil Nadu, India
National Institute Of Technical Teachers Training Trainess Guest House, NATIONAL INSTITUTE OF TECHNICAL TEACHERS TRAINING, Tharamani, Chennai, Tamil Nadu 600113, India
Lat 12.988547°
Long 80.250139°
22/08/24 03:58 PM GMT +05:30
Google


GPS Map Camera
Chennai, Tamil Nadu, India
National Institute Of Technical Teachers Training Trainess Guest House, NATIONAL INSTITUTE OF TECHNICAL TEACHERS TRAINING, Tharamani, Chennai, Tamil Nadu 600113, India
Lat 12.988547°
Long 80.250139°
22/08/24 03:58 PM GMT +05:30
Google


GPS Map Camera
Chennai, Tamil Nadu, India
ABDUL KALAM ACADEMIC BLOCK, NATIONAL INSTITUTE OF TECHNICAL TEACHERS TRAINING, Tharamani, Chennai, Tamil Nadu 600113, India
Lat 12.987878°
Long 80.248733°
09/08/24 04:04 PM GMT +05:30
Google