ENEL 351 - Microcontrollers | Winter 2025

SMART DISTANCE WARNING AID

Final Project Report

Student: Gursharan Singh Rehal

S ID: 200480626

TABLE OF CONTENTS

1. PROJECT OVERVIEW

The Smart Distance Warning Aid is a microcontroller-based embedded system developed as a final project for the ENEL 351 course at the University of Regina. Its primary objective is to provide real-time proximity alerts through a combination of visual, auditory, and tactile cues. Designed with accessibility and real-world usability in mind, the project leverages widely available sensors and a structured firmware model to achieve accurate environmental awareness.

The core of the system is the STM32F103RB microcontroller, which interfaces with multiple peripherals to detect and respond to distance data. The HC-SR04 ultrasonic sensor is employed to calculate the distance to nearby objects, while an IR proximity sensor serves as a secondary input to validate proximity in the immediate vicinity. This dual-sensor approach improves reliability, especially in dynamic or cluttered environments. To alert the user, the system uses an LED (PA6) and a piezo buzzer (PA7), both of which activate in pre-defined patterns based on how close an object is detected. Additionally, a vibration motor (PA4) is integrated to provide tactile feedback—originally meant to serve as a power-on confirmation, it was later refined to be part of the system's startup sequence.

The UART interface allows for continuous debugging and live monitoring through a terminal (e.g., PuTTY). The firmware includes non-blocking delay logic using SysTick, which ensures real-time responsiveness without the use of blocking loops. Filtering mechanisms such as sudden-drop suppression, IR gating, and glitch rejection improve measurement reliability.

Key Highlights:

- Real-time distance monitoring using ultrasonic and IR sensors
- LED, buzzer, and vibration feedback system
- Built with STM32 HAL functions, SysTick timing, and TIM2 input capture
- Optimized for accessibility, system responsiveness, and noise filtering

2. SYSTEM ARCHITECTURE

The Smart Distance Warning Aid is architected around the STM32F103RB Nucleo-64 development board, utilizing a modular approach to integrate input, output, processing, and communication components. Each sensor and actuator is connected via specific GPIO pins on Port A, with critical functions such as timing and delay handled using dedicated STM32 peripherals (SysTick and Timer2).

The system includes two sensor inputs: an ultrasonic distance sensor (HC-SR04) and an infrared proximity sensor. The HC-SR04 is responsible for measuring the distance to objects using a sonar-based pulse-echo method. It is connected to GPIO PA0 (trigger) and PA1 (echo). PA1 leverages the input capture capability of TIM2 to measure the pulse duration returned from the sensor. The IR sensor connects to PA5 and is used to gate the ultrasonic measurements—if the IR sensor does not detect a nearby object, distant ultrasonic reflections are filtered out.

The outputs of the system include an LED on PA6 for visual feedback, a buzzer on PA7 for auditory alerts, and a vibration motor on PA4 for tactile feedback. Each output is controlled using non-blocking toggling driven by SysTick-based timing logic. The LED and buzzer blink in synchronized patterns based on distance thresholds, while the vibration motor provides a brief tactile pulse on system startup to indicate readiness.

Serial communication is achieved via UART on PA9 (TX), allowing the device to send real-time data, debugging messages, and sensor readings to a terminal such as PuTTY or Tera Term.

Architectural Design Summary:

- Inputs: HC-SR04 (PA0, PA1), IR sensor (PA5)
- Outputs: LED (PA6), Buzzer (PA7), Vibration Motor (PA4), UART TX (PA9)
- Processing: Distance computation, zone mapping, filtering
- Timing: TIM2 for pulse capture, SysTick for LED/Buzzer/motor delays
- Communication: USART1 configured at 9600 baud for live data logging

3. SYSTEM FEATURES

The Smart Distance Warning Aid distinguishes itself through a wide array of features tailored to meet the functional and real-world needs of an interactive microcontroller project. At its core, the system blends distance sensing and multi-modal feedback using a combination of interrupt-driven programming and robust signal conditioning. The integration of auditory, visual, and tactile outputs ensures the device is accessible to a wide range of users, including individuals with visual impairments or those operating in low-light environments.

A notable software feature is the use of SysTick for non-blocking timing operations. Unlike traditional delay functions, which halt program execution, the SysTick-based structure allows concurrent sensor monitoring and output control. This ensures real-time responsiveness even under frequent signal changes or movement near the sensor. The main loop is tightly optimized to poll the IR sensor, trigger the ultrasonic sensor, capture echo timing using TIM2, and then map the measured pulse duration to real-world distance values.

Software filtering is another critical highlight. Glitch suppression mechanisms avoid false readings caused by electrical noise or sensor anomalies. Additionally, a sudden-drop filter detects and suppresses unexpected drops in distance unless repeated across multiple cycles, ensuring that brief signal losses or false echoes do not disrupt the feedback cycle. IR-based gating further improves reliability by validating ultrasonic results only when a nearby presence is detected.

The LED and buzzer operate in synchronized pulse patterns that change with proximity levels, reinforcing the feedback in both sensory domains. The vibration motor provides a subtle haptic signal on startup, confirming that the device has successfully powered on without requiring visual or audio confirmation.

Feature Summary:

- SysTick-based non-blocking timing for efficient multitasking
- Real-time distance feedback mapped to 6 levels of output intensity
- IR gating and echo suppression for accurate measurement filtering
- Glitch, timeout, and sudden drop filtering for reliable readings
- UART debugging and terminal-based logging
- Tactile power-on signal using vibration motor

4. DIFFICULTIES FACED

During the development of the Smart Distance Warning Aid, several technical, conceptual, and practical challenges were encountered. One of the primary hurdles was configuring the ultrasonic sensor with consistent echo timing capture using Timer2's input capture mode. Due to the nature of ultrasonic waves, reflections from multiple surfaces or from unintended angles led to unstable readings, necessitating the development of advanced software filters.

Another challenge was ensuring reliable non-blocking feedback synchronization. Managing the blinking LED and buzzing patterns required precise time-based toggling without using blocking delays, which would interfere with sensor readings. Implementing SysTick-based timing logic helped resolve this, but required careful calibration and debugging.

The IR sensor, while simple in design, occasionally introduced complications in detection reliability, particularly in changing lighting environments or with reflective surfaces. This prompted the inclusion of IR gating logic and internal pull-up configuration to maintain stability.

An additional difficulty was integrating the vibration motor without damaging the STM32's GPIO circuitry. The motor's inrush current, especially during startup, presented a risk of overstressing PA4. To mitigate this, its use was limited to a short power-on burst to act as a tactile system readiness signal.

Lastly, serial communication over UART initially posed problems with inconsistent baud rate configurations and unstable COM port detection. Once synchronized at 9600 baud and paired with PuTTY, it became a reliable debugging and monitoring tool.

Key Difficulties:

- Echo timing instability and multi-reflection noise
- Non-blocking feedback control and timing synchronization
- IR sensor reliability in variable environments
- Safe GPIO control of high-draw components (vibration motor)
- UART connection inconsistencies across development machines

5. ACCOMMODATIONS MADE

In response to the technical challenges and hardware limitations faced throughout the project, a series of key accommodations were introduced to ensure successful implementation and safe operation of the Smart Distance Warning Aid. These adaptations reflect both design revisions and workflow improvements based on real-world constraints.

One major accommodation involved the planned capacitive touch sensor. Initially proposed to activate or deactivate the system, the sensor was later removed due to complexity and limited responsiveness in practical tests. Instead, a more intuitive and simpler feedback mechanism was adopted: a vibration motor that signals system power-on. This solution provided a clearer tactile response while reducing GPIO usage and simplifying firmware logic.

To safely integrate the vibration motor without a transistor driver, the code was modified to limit motor activation to a short, controlled pulse at system startup. This approach protected the STM32's GPIO (PA4) from overcurrent conditions while still maintaining the intended tactile user feedback.

Another accommodation was made in the distance filtering logic. Originally, sudden distance drops were processed immediately, often resulting in false alarms. This behavior was mitigated by adding a counter-based confirmation mechanism—sudden drops were only accepted if they persisted over three consecutive cycles. This significantly reduced false triggers.

In terms of UART communication, alternative tools like Tera Term were explored, but PuTTY was chosen for its stability and simplicity. Baud rate mismatches were resolved through consistent configuration across hardware and software environments.

Accommodation Highlights:

- Capacitive sensor removed in favor of tactile motor signal
- Controlled motor current by limiting duration of activation
- Implemented counter-based glitch filtering for smoother output
- Streamlined UART settings and standardized terminal interface

6. BUDGET AND RESOURCES

The Smart Distance Warning Aid project was designed with accessibility and affordability in mind. From the outset, one of the project goals was to use low-cost, widely available components to demonstrate the core principles of real-time embedded systems while staying within a tight budget. To ensure feasibility and replicability, off-the-shelf components were chosen for all major subsystems.

The core controller, the STM32F103RB Nucleo-64 board, was provided through course resources and served as the central development platform. Its built-in debugging tools, USB interface, and accessible pin headers made it ideal for prototyping. The HC-SR04 ultrasonic sensor was selected for its affordability and reliability in indoor range measurements. The IR proximity sensor was also a low-cost digital sensor commonly found in obstacle avoidance projects, costing less than $1 CAD.

The buzzer, LED, and associated resistors were salvaged from lab supplies. A small 3V vibration motor was procured for under $3 CAD and directly integrated into the breadboard prototype. Additional wiring, jumper cables, and breadboard space were already available as part of the standard lab kit.

Tools such as Keil uVision IDE (for C programming and debugging), PuTTY (for serial monitoring), and CAD software (for schematic design) were all free to use or provided through university licensing. No specialized equipment beyond the ADALM2000 logic analyzer was required for testing signal behavior.

Budget Summary:

- STM32F103RB Nucleo Board: Provided (course resource)
- HC-SR04 Ultrasonic Sensor: ~$2.00 CAD
- IR Proximity Sensor: ~$0.80 CAD
- Buzzer: ~$1.00 CAD
- LED + Resistor: ~$0.20 CAD
- Vibration Motor: ~$2.50 CAD
- Misc wiring/breadboard: Provided in lab
- Software Tools: Free (university licensed or open source)

7. SCHEMATIC DESIGN AND EXPLANATION

The Smart Distance Warning Aid's final schematic was developed using KiCad EDA and is included in the deliverables as "Final-Schematic.pdf". The schematic visually represents all electrical connections between the STM32F103RB microcontroller and the peripheral components used in the project. It was designed to match the final prototype implementation and ensure clarity in both hardware setup and documentation.

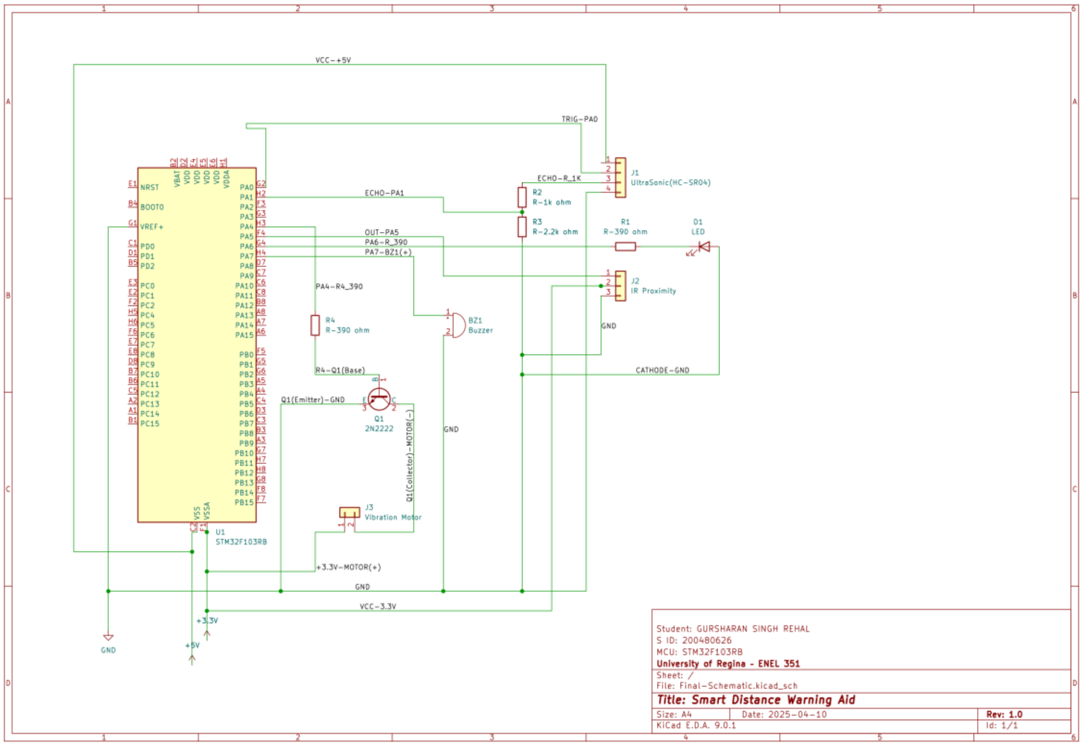
The microcontroller (U1) is placed centrally and acts as the hub for all I/O interactions. Key GPIO pins on Port A are routed to various components. The ultrasonic sensor (J1) connects through PA0 for the TRIG signal and PA1 for the ECHO, with a 1kΩ resistor (R2) in series to protect the echo line. The IR proximity sensor (J2) is connected to PA5 and also protected with a 2.2kΩ resistor (R3). The buzzer (BZ1) is wired directly to PA7 and powered via 3.3V.

The LED (D1), connected to PA6, is driven through a 390Ω current-limiting resistor (R1). The vibration motor, connected via J3, is controlled using a 2N2222 transistor (Q1). The motor's base is connected to PA4 through a 390Ω resistor (R4), while its emitter is grounded. A flyback diode (not shown explicitly but implied) is assumed to be placed across the motor terminals to protect against voltage spikes.

The schematic provides a complete and accurate representation of the real-world connections and serves as a blueprint for system replication, debugging, and presentation.

Key Schematic Features:

- Clear GPIO mapping for all inputs and outputs
- Protection resistors on signal lines
- Transistor-controlled motor circuit with PA4 input
- Legend and title block with student name, course, and project title



---

8. LESSONS LEARNED

Throughout the development of the Smart Distance Warning Aid, I encountered numerous challenges that ultimately shaped my growth as both an embedded systems designer and problem solver. One of the most significant takeaways was the importance of planning around hardware limitations. Initially, I assumed that I could directly drive a vibration motor using a GPIO pin. I learned that even a seemingly simple addition can introduce electrical risks and the need for current management, prompting a complete reassessment of the power delivery strategy.

In terms of software, implementing non-blocking logic using SysTick gave me a deep appreciation for real-time responsiveness in embedded systems. Blocking delays, while simple to implement, interfered with feedback consistency. Migrating to an interrupt-driven approach was a turning point in maintaining reliable multi-modal output while still ensuring timely sensor polling.

Another important lesson was modular coding. By structuring my main logic into clean segments for input capture, data filtering, and output control, debugging became significantly easier. This practice also made the firmware more readable and extendable, which is essential in collaborative or long-term projects.

The importance of filtering became even clearer during field testing. While sensors performed well in ideal conditions, real-world usage involved erratic lighting, odd reflections, and false positives. Tackling these challenges through IR gating, glitch suppression, and timeout detection taught me to always plan for noise and exceptions.

Lessons Summary:

- Consider hardware limitations early in the design phase
- Non-blocking SysTick logic ensures reliable multitasking
- Modular firmware architecture simplifies debugging and extensions
- Real-world testing is essential to reveal edge cases and signal noise
- Tactile and sensory feedback improves accessibility and usability

---

9. FUTURE APPLICATIONS

The Smart Distance Warning Aid project has laid a robust foundation that opens the door to a wide range of potential enhancements and practical applications. While the current implementation targets proximity detection and accessibility use cases, the underlying architecture and modular design allow it to be extended into more advanced systems for diverse real-world scenarios.

One immediate future direction involves integrating a PWM controller or motor driver IC to allow programmable vibration feedback, including variable intensity or rhythmic pulses. This would make the tactile feedback more expressive and potentially better suited for different user profiles, including those with limited hearing or vision.

The addition of a small OLED or LCD screen could provide direct visual output, making the device useful in robotics, educational kits, or even compact measurement tools for maintenance personnel. With minimal hardware additions, the system could also store peak or average distance values and present historical trends on screen.

On the connectivity front, a BLE or Wi-Fi module (such as ESP8266 or ESP32) could be added for wireless telemetry or IoT integration. This would allow the device to send alerts or readings to a mobile app, server, or cloud dashboard, thereby increasing its relevance in smart home or industrial safety applications.

Further, the use of a rechargeable battery and compact enclosure could convert the prototype into a wearable haptic proximity assistant—ideal for personal navigation or safety in crowded spaces.

Potential Future Extensions:

- Add LCD or OLED for real-time display of distance
- Integrate wireless modules (BLE, Wi-Fi) for remote logging or mobile alerts
- Introduce rechargeable battery + charger module for portability
- Reinstate capacitive touch or gesture control for enable/disable functionality
- Expand filtering logic with machine learning to adapt to usage environment
- Develop companion mobile app for live tracking, alerts, and logging

---

10. CONCLUSION

The Smart Distance Warning Aid stands as a fully realized demonstration of practical embedded system design with real-world applicability. From conceptualization through to physical prototyping and final testing, this project showcases the integration of multiple hardware components—ultrasonic sensor, IR detector, buzzer, LED, and vibration motor—coordinated by a modular and highly responsive firmware environment running on the STM32F103RB.

The strength of the system lies not only in its multi-modal feedback loop but also in its attention to signal reliability and user accessibility. Its modular structure, effective use of interrupts and timers, and robust software filtering ensure that it can operate stably in dynamic environments. These characteristics make it suitable for diverse applications ranging from assistive technology to robotic obstacle detection systems.

The project also provided invaluable educational experiences in microcontroller interfacing, low-level GPIO control, non-blocking programming strategies, and practical limitations of hardware integration. From the initial challenges of pulse-width timing and IR noise filtering to creative accommodations like tactile startup signaling, each obstacle helped refine the final solution.

Most importantly, the completed system is affordable, efficient, and reproducible—core principles that make it well-aligned with modern engineering practices for scalable, real-world embedded applications.

Final Takeaways:

- Successfully developed an affordable and reliable proximity alert system
- Demonstrated effective use of STM32 timers, SysTick, and interrupts

- Emphasized system responsiveness and output feedback clarity
- Delivered a comprehensive final report, working prototype, and complete schematic
- Gained hands-on experience in debugging, hardware safety, and feature optimization

---

12. APPENDIX B: GLOSSARY OF TERMS

- STM32F103RB – ARM Cortex-M3-based microcontroller from STMicroelectronics, used as the system's core controller.
- GPIO – General Purpose Input/Output; microcontroller pins used for digital signals.
- SysTick – A system timer in ARM Cortex-M cores used for precise, non-blocking delays.
- TIM2 Input Capture – A timer feature that records the timer count when a signal edge is detected, used to measure pulse widths.
- HC-SR04 – Ultrasonic distance sensor used to measure proximity by emitting and detecting echo pulses.
- IR Sensor – Infrared proximity sensor that detects nearby presence and outputs a digital LOW signal.
- UART – Universal Asynchronous Receiver/Transmitter, a protocol used for serial communication.
- PuTTY – A terminal emulator commonly used for UART serial monitoring on Windows.
- Glitch Filtering – Software method to reject erroneous sensor data caused by interference or sudden environmental changes.
- Flyback Diode – A diode placed across inductive loads to prevent voltage spikes that could damage transistors or microcontroller pins.