WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

A Technical Review : On Industrial Training

Anitosh Dash

ECE, Dayananda sagar college of enginnering
Banguluru, India
anitosh2003@gmail.com

ABSTRACT—

This paper presents a comprehensive review of the technical exposure and experiential learning gained during an internship at HCLTech. The internship focused on Embedded Systems Engineering and involved both theoretical training and hands-on projects. This review outlines the development of a vehicle that detects obstacles using sensor technology, the practical implementation of C programming in an industrial environment, and the use of GDB for debugging. The paper also reflects on the integration of classroom knowledge with real-world engineering practices

Keywords— Theoretical training, sensor technology, GDB.

Introduction

Internships provide vital industrial exposure that bridges the gap between academic learning and practical implementation. This review summarizes the key components of an internship undertaken at HCLTech, including technical training, tool usage, and a project involving obstacle detection in autonomous systems. The internship was structured to provide exposure to real-time industry practices, where learning was not limited to coding, but extended into areas such as system design, testing, debugging, and performance optimization. HCLTech's comprehensive program allowed for a well-rounded experience combining mentorship, guided learning modules, and self-driven exploration. Interns were encouraged to engage in hands-on practice, critically analyze problem statements, and participate in developing solutions that resemble those used in actual product development cycles. This paper aims to document and reflect upon the various facets of this technical journey, the integration of software principles with hardware systems, and the skills acquired that contribute to an aspiring embedded systems engineer's professional toolkit.

Company Overview

HCLTech is a leading global technology company that provides a wide range of services including IT consulting, enterprise transformation, remote infrastructure management, engineering, and R&D services. With a strong presence in the field of digital, engineering, and cloud services, HCLTech is known for driving innovation and delivering scalable solutions across industries such as healthcare, finance, automotive, aerospace, and telecommunications.

During the internship period, HCLTech provided a structured environment that fostered both technical growth and professional development. Interns were given access to well-designed training modules, mentorship by experienced professionals, and the opportunity to work on real-time projects. The company emphasizes practical exposure, teamwork, and a problem-solving mindset—enabling interns to understand how large-scale embedded systems and applications are developed and maintained in the real world. The experience at HCLTech not only enhanced technical competencies but also helped in developing a deeper understanding of the expectations and workflows in a professional engineering environment.

Training Overview

The internship at HCLTech began with a structured and intensive training phase designed to provide foundational and advanced knowledge required for real-world embedded system development. The training primarily focused on the **C programming language** as used in industrial embedded environments and the use of **GDB (GNU Debugger)** for professional-grade debugging

Industrial C Programming for Embedded Systems: C remains the dominant language in embedded systems due to its low-level hardware access, deterministic performance, and portability. The training at HCLTech was tailored to equip interns with both fundamental knowledge and specialized techniques used in real-world projects.

Topics covered included:
Memory Architecture Awareness: Interns were trained to understand the memory segmentation of embedded systems, including stack, heap, data, and code segments. Concepts like static vs. dynamic memory allocation, pointer dereferencing, and array vs. pointer relationships were covered in-depth.
Peripheral Register Access and Bitwise Operations: Understanding how to manipulate hardware registers using bit masking and shifting was emphasized. Examples included controlling GPIOs and timers through register-level programming using macros and pointers.
Structured Code Design: The training emphasized modularity, reusability, and readability of C code.
Compiler Optimizations and Warnings: Interns were trained to enable and interpret compiler flags like -Wall, -O1, -O2, etc., and understand how compiler optimizations affect code execution in embedded systems.

Preprocessor Directives and Build Systems

In industrial settings, the C preprocessor is heavily used to manage complex codebases and enable platform-specific configurations.
Topics covered included:
Conditional Compilation: Use of #ifdef, #ifndef, #define, and #endif to selectively compile sections of code based on build targets or debug modes.
Macros and Constants: Creation of reusable macros for setting and clearing bits, parameterized macros for peripheral control, and safe macro practices using parentheses.
Header Guards and Include Hierarchies: Proper header file management to avoid multiple inclusion errors and maintain project portability.
Introduction to Makefiles: Although advanced build systems like CMake were briefly mentioned, emphasis was placed on traditional Makefile usage to automate compilation, define build rules, and handle dependencies effectively.

This training laid a solid technical foundation for implementing the obstacle detection vehicle project and simulated real-world conditions where tight memory constraints, low-level hardware control, and efficient debugging are critical.

Project Objective

To design and implement a vehicle system capable of detecting obstacles in its path and halting or diverting movement accordingly—supporting the use-case of automated or semi-autonomous navigation.

C Software Development

The control logic was written in C/C++ using the Arduino IDE and compiled with GCC. Key logic included:
- Real-time distance measurement using echo and trigger pins
- Conditional logic to stop or reroute when an object is within a threshold distance
- Serial output for monitoring system behavior.

Integration of Training and Project Work

The C training directly contributed to efficient memory handling, better modularity, and debugging skills using GDB during the development of the vehicle system. The lessons on pointer operations and interrupt handling were crucial when working with sensor inputs and motor controls.

Challenges faced

During the internship project focused on obstacle detection in vehicles, several technical and practical challenges emerged, which required a systematic and analytical approach to resolve.
One of the primary issues encountered was inconsistent and noisy sensor data from the ultrasonic distance sensors. These sensors, while economical and easy to interface, are highly sensitive to environmental conditions such as surface texture, angle of incidence, and ambient noise. Initially, the vehicle would react unpredictably due to erratic distance readings. To overcome this, a software-based filtering technique was implemented, where multiple sensor readings were taken over a short time window and the median value was used. This approach significantly stabilized the vehicle's behavior by reducing the impact of outlier readings.

Learning Outcomes

Exposure to embedded system development cycles. Enh
anced coding efficiency and debugging capability with GDB. Real-time problem-solving and cross-functional integration of electronics and programming.

Conclusion

The internship at HCLTech proved to be an invaluable experience that bridged the gap between theoretical knowledge and industrial application. Through

a structured blend of technical training and hands-on project work, I gained deeper insights into the workings of embedded systems, the industrial use of the C programming language, and the critical role of debugging tools such as GDB in ensuring reliable software performance.

The obstacle detection vehicle project not only enhanced my understanding of sensor integration, real-time processing, and hardware-software interfacing but also taught me how to handle real-world engineering challenges such as sensor noise, power instability, and hardware communication failures. This practical experience reinforced the importance of careful system design, modular coding practices, and robust testing methodologies.

Acknowledgments

I would like to express my heartfelt gratitude to HCLTech for providing me with the opportunity to undertake this enriching internship. The structured training, technical mentorship, and practical exposure have significantly contributed to my professional growth and understanding of embedded systems and industrial software development.

I am especially thankful to the Embedded Systems and ED&A Testing team at HCLTech for their continuous support, guidance, and valuable feedback throughout the internship period. Their willingness to share knowledge and encourage hands-on learning was instrumental in the successful completion of my project.

I would also like to thank my academic mentors and faculty members for their encouragement and for equipping me with the foundational knowledge that prepared me to face real-world challenges confidently.

Finally, I am grateful to my fellow interns and peers, whose collaboration and shared experiences made the journey more insightful and enjoyable

REFERENCES

[1] [1] B. W. Kernighan and D. M. Ritchie, *The C Programming Language*, 2nd ed. Englewood Cliffs, NJ, USA: Prentice Hall, 1988.

[2] [2] GNU Project, "GDB: The GNU Project Debugger," [Online]. Available: https://www.gnu.org/software/gdb/. [Accessed: May 18, 2025].

[3] [3] Arduino, "Arduino UNO Documentation," [Online]. Available: https://docs.arduino.cc/hardware/uno-rev3. [Accessed: May 18, 2025].

[4] [4] HCLTech, *Embedded Systems and C Programming Modules*, Internal Training Resource, 2025.

[5] [5] Seeed Studio, "HC-SR04 Ultrasonic Sensor Datasheet," [Online]. Available: https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf. [Accessed: May 18, 2025].

[6] [6] Espressif Systems, *ESP32 Technical Reference Manual*, [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf. [Accessed: May 18, 2025].

[7] [7] J. Ganssle, *The Art of Designing Embedded Systems*, 2nd ed. Burlington, MA, USA: Newnes, 2008.

[8] [8] D. Molloy, *Exploring C for Microcontrollers: A Hands-on Approach*, Hoboken, NJ, USA: Wiley, 2019