SIT210

EMBEDDED SYSTEMS DEVELOPMENT

Learning Summary Report

Aruj Punia
2310994767

Self-Assessment Details

The following checklists provide an overview of my self-assessment for this unit.

		Pass (D)	Credit (C)	Distinction (B)	High Distinction (A)
Self-Assessment				✓	

Declaration

I declare that this portfolio is my individual work. I have not copied from any other student’s work or from any other source except where due acknowledgment is made explicitly in the text, nor has any part of this submission been written for me by another person.

Signature: **ARUJ PUNIA**

Portfolio Overview

This portfolio includes work that demonstrates that I have achieve all Unit Learning Outcomes for SIT210 Unit Title to a **Distinction** level.

Portfolio Summary

In my portfolio, I demonstrate comprehensive achievement of all Unit Learning Outcomes for the SIT210 Embedded Systems Development course, with work that aligns strongly with a Distinction grade. Throughout the course, I consistently demonstrated advanced knowledge of embedded systems architecture and hardware interactions, while my projects exceeded expectations by showcasing innovative and in-depth solutions to complex challenges. Beyond course requirements, I pursued additional projects in embedded systems, further developing my skills and underscoring my commitment to this field. Based on these accomplishments, I am confident that my performance meets the Distinction standard. I respectfully request that the assessment panel consider these contributions when determining my final grade. Thank you.

ULO1: Software Development
(Develop software necessary for controlling devices within an embedded system.)

- Task 1.1 (Pass): I developed foundational skills in setting up and programming an Arduino Nano 33 IoT, configuring it for the Arduino IoT Cloud, and enabling Wi-Fi connectivity—key skills for embedded systems and IoT development.

- Task 2.1 (Pass): Through this task, I gained hands-on experience in embedded systems programming on the Arduino Nano 33 IoT, focusing on client requirement analysis, structured software development, resource integration, troubleshooting, and applying a client-centered approach. I also used IFTTT to send an email.

- Task 3.3 (Distinction): I created an embedded system using an Arduino and an LED to respond to virtual 'wave' signals from a simulated device, gaining experience with Arduino hardware, setting up a publish-subscribe structure, and developing a responsive embedded system.

- Task 3.1 (Pass): I enhanced my understanding of IoT connectivity by programming my Arduino device to send data to the web. I assembled a basic circuit board with multiple sensor options, wrote Arduino code to capture sensor data into a variable, and published it on ThingSpeak.com. Additionally, I set up a webhook in the Arduino Console for seamless data transmission.

- Task 3.2 (Credit): I designed an embedded system with Arduino to monitor sunlight levels in a terrarium, using IFTTT to trigger notifications. This task strengthened my skills in hardware setup, coding, testing, and integrating with external services.

- Task 7.2 (Distinction): In this assignment, I implemented audio processing on the Raspberry Pi (RPi) to detect and interpret audio signals, which can be used to trigger actions, monitor audio events, or enable voice control. This project helped me develop foundational skills and tools for the RPi to recognize sounds or voice commands, with potential applications in controlling devices within an embedded system context.

- Task 11.1 (Pass): For this task, we presented our project work through a video demonstration.

ULO2: Embedded System
(Develop an embedded system by interfacing sensing, computation and communication components to solve real word problems.)

- Task 1.1 (Pass): I developed foundational skills in setting up and programming an Arduino Nano 33 IoT, configuring it for the Arduino IoT Cloud, and enabling Wi-Fi connectivity—key skills for embedded systems and IoT development.

- Task 3.1 (Pass): I enhanced my understanding of IoT connectivity by programming my Arduino device to send data to the web. I assembled a basic circuit board with multiple sensor options, wrote Arduino code to capture sensor data into a variable, and published it on ThingSpeak.com. Additionally, I set up a webhook in the Arduino Console for seamless data transmission.

- Task 7.1 (Credit): I evaluated project pitch presentations from classmates, providing feedback on project clarity, content structure, enthusiasm, speech clarity, and overall organization.

- Task 7.2 (Distinction): In this task, I built a system utilizing the audio processing capabilities of the Raspberry Pi (RPi) with available libraries. The system processes audio inputs to allow interaction via sound, such as voice commands or sound-triggered actions, enabling hands-free control of connected devices. By analyzing audio data to detect keywords or sounds, this system supports real-world applications in areas like home automation and security, where audio processing adds interactivity and adaptability to an embedded control setup.

- Task 11.1 (Pass): In this task we have presented our task through video.

- Task 4.1 (Pass): This task supports the Unit Learning Outcome (ULO) of developing an embedded system by integrating sensing, computation, and communication components to address practical problems. Installing Raspberry Pi OS provided a foundational operating environment for embedded system development. Learning remote access to the RPi introduced a communication component, enabling control and monitoring from a distance—essential for many embedded applications. Additionally, programming the RPi to perform basic operations established foundational coding skills, preparing me to integrate sensors and other components. These steps collectively establish the groundwork for creating responsive, problem-solving embedded systems.

- Task 5.2 (Credit): I developed skills in creating graphical user interfaces (GUIs) on a Raspberry Pi using Python. This included building a circuit to control three LEDs and designing a GUI with radio buttons for each LED and an "Exit" button.

ULO3: Sensing
(Design sensing architectures that satisfy performance constraints related to robustness, fault tolerance and responsiveness.)

- Task 1.1 (Pass): I gained foundational skills in setting up and programming an Arduino Nano 33 IoT, configuring it for the Arduino IoT Cloud, and establishing Wi-Fi connectivity—key competencies in IoT and embedded systems development.

- Task 3.2 (Credit): I designed and implemented an Arduino-based embedded system to monitor sunlight exposure in a terrarium and trigger notifications through IFTTT. This task enhanced my skills in hardware configuration, coding, testing, and integration with external services.

- Task 4.1 (Pass): In Part 1, I learned to set up a Raspberry Pi with a new operating system, including formatting the SD card, installing the OS, and managing files. Part 2 involved exploring remote access methods, expanding my understanding of the Raspberry Pi's remote functionalities.

- Task 6.1 (Pass): This task involved creating a video project pitch as the initial step in developing an embedded system prototype. The pitch outlined an understanding of the problem domain, industry practices, and design challenges, identifying gaps for an embedded solution. It presented an overview of the proposed system, emphasizing key components, interconnections, protocols, and user interface plans, using block and flow diagrams to clarify system structure and performance goals.

- Task 11.1 (Pass): We demonstrated our work for this task through a video presentation.

ULO4: Experiential Learning
(Reflect upon learning achieved through experiences within the unit, and assessment tasks to highlight development of technical expertise in relation to career opportunities.)

- Task 2.1 (Pass): I developed hands-on skills in embedded systems programming on the Arduino Nano 33 IoT, focusing on understanding client requirements, structured software development, integrating external resources, troubleshooting, and adopting a client-centered approach.

- Task 3.2 (Credit): I created a tutorial video for Task 3.1, enhancing my communication skills, technical expertise in Arduino and embedded systems, attention to detail, and empathy for learners' needs.

- Task 3.1 (Pass): I gained insights into IoT connectivity by programming my Arduino device to send data to the web. I built a basic circuit board with multiple sensor options, wrote

Arduino code to capture sensor data, and published it on ThingSpeak.com. Additionally, I set up a webhook integration in the Arduino Console for data transmission.

- Task 6.1 (Pass): I developed a project proposal for a "Smart Inventory Management System," outlining a system that uses RFID for adding products to the inventory and facilitates communication between multiple microcontrollers.

- Task 7.2 (Credit): I evaluated classmates' project pitch presentations, providing feedback on clarity, enthusiasm, content structure, speech clarity, and overall organization.

- Task 3.3 (Distinction): I deepened my expertise in IoT by implementing publish-subscribe communication between two Arduino devices, advancing my independent problem-solving skills, which are essential for career opportunities in embedded systems and scalable networking.

- Task 4.1 (Pass): In Part 1, I set up a Raspberry Pi with a new OS, including formatting the SD card, OS installation, and basic file management. Part 2 involved exploring remote access methods, expanding my understanding of the Raspberry Pi's remote capabilities.

- Task 5.1 (Pass): I gained a foundational understanding of coding on the Raspberry Pi using the GPIO library, creating an interface to control an LED, writing Python scripts to manage the LED, and running the code on the Raspberry Pi.

- Task 11.1 (Pass): For this task, we demonstrated our work through a video presentation.

Reflection

The most important things I learnt:

In the course “Embedded Systems Development” (SIT210), I have gained an advanced understanding and high-level skills, especially in working with the Raspberry Pi 4 and Arduino Nano 33 IoT. I developed a deep comprehension of microcontrollers, programming languages, and real-time operating systems, equipping me to navigate complex aspects of embedded systems design and application. This course enabled me to translate theory into sophisticated, functional systems by applying best practices in programming, debugging, and optimizing resources for efficient design.

I refined my skills in sensor interfacing and mastering communication protocols to create seamless, integrated systems. Working with the Raspberry Pi 4 and Arduino Nano 33 IoT, I enhanced my ability to design scalable, responsive solutions, highlighting my adaptability in tackling real-world challenges. Engaging in collaborative projects allowed me to take on leadership roles, fostering effective teamwork and project management skills. This experience has underscored the broad, real-world significance of embedded systems and the ongoing need for skill enhancement in this dynamic field.

The things that helped me most were:

The most beneficial aspects of my learning experience were the hands-on projects, which played a crucial role in deepening my understanding of embedded systems. These projects allowed me to apply theoretical knowledge in real-world scenarios, helping me develop the critical skills necessary for designing and implementing embedded systems. The collaborative nature of these assignments also improved my teamwork abilities, mirroring real-world dynamics where clear communication and effective coordination are vital for success.

Additionally, the guidance and expertise of the instructors were invaluable. Their clear explanations, readiness to address questions, and practical insights were instrumental in solidifying my understanding of complex topics. Their support ensured that I could fully grasp the material and apply it effectively in both individual and collaborative settings.

I feel I learnt these topics, concepts, and/or tools really well:

Throughout my course, I have gained a strong understanding of microcontrollers, exploring their complex architecture and versatile capabilities. This knowledge has enabled me to choose the most suitable microcontroller for specific applications, configure them to meet particular requirements, and develop customized software that aligns with their specifications.

Additionally, I gained proficiency in using Real VNC to remotely access and control my Raspberry Pi, improving my ability to troubleshoot and code efficiently. My programming skills in Python also expanded significantly, as I developed optimized code for embedded systems. Furthermore, I took the initiative to help half of my class by teaching them how to remotely connect and control their Raspberry Pi, as well as how to install the OS, which further strengthened my understanding and communication skills.

I still need to work on the following areas:

I still need to work on a few areas to enhance my embedded systems skills. Although I have a solid foundation in debugging, I aim to improve my advanced debugging techniques, particularly for complex real-time embedded systems that involve intricate timing issues.. Additionally, I'm working to improve my understanding of advanced memory management, including techniques like memory pooling and efficient data structure allocation. Lastly, I recognize the need to further develop my coding skills to ensure more efficient, optimized, and robust code for embedded systems.

This unit will help me in the future:

This unit has provided a solid foundation in embedded systems, which will serve as a key stepping stone for future technologies. The knowledge I've gained is essential for advancing in fields such as electrical engineering, computer science, and related disciplines. Beyond academics, the skills and insights I've developed open up numerous career opportunities. Embedded systems are critical in industries like automotive, healthcare, aerospace, and the Internet of Things (IoT), making my skills highly valuable in these areas. Furthermore, the problem-solving abilities I've strengthened during this unit are transferable, benefiting me in various professional roles and future academic endeavors.

I found the following topics particularly challenging:

One of the most challenging tasks I encountered was setting up Firebase and creating a web UI to integrate and sync my Arduino Nano 33 IoT-based setup with three LEDs. This involved configuring real-time data communication between the Arduino and the web interface, which required careful attention to both hardware and software integration.

In addition, I faced challenges in memory management, requiring precise attention to ensure efficient allocation and avoid memory leaks or overflows. Mastering advanced communication protocols like SPI and I2C was another hurdle, as it involved troubleshooting and configuring multi-device communication. Low-level programming in Assembly also posed a significant challenge, as optimizing code for microcontrollers demanded a deep understanding of hardware and the ability to write efficient, low-level instructions. These challenges helped deepen my understanding of embedded systems development and refined my problem-solving abilities.

I found the following topics particularly interesting:

I found several aspects of embedded systems development particularly fascinating. The practical experience of sensor interfacing stood out as especially captivating. Learning how to collect data from sensors like accelerometers, temperature sensors, and gyroscopes for real-world applications presented both challenges and rewards. I was also intrigued by the concept of resource optimization, especially in environments with limited resources. Finding the right balance between functionality and performance while managing constraints such as power and memory limitations was a thought-provoking challenge.

Additionally, the process of developing an embedded project from concept to execution was highly rewarding. It allowed me to apply my knowledge in a hands-on way, creating tangible solutions and fostering a problem-solving mindset that kept me engaged and motivated throughout the course.

If I did this unit again, I would do the following things differently:

If I were to take this unit again, I would approach my learning and coursework differently in a few key ways. First, I would prioritize starting assignments and projects early to allow for deeper exploration and problem-solving, reducing the stress of last-minute work. Second, I would improve my time management by creating a more structured study schedule to balance my workload and avoid the need for cramming. Lastly, I would focus on consistent practice, dedicating regular time to coding, debugging, and applying concepts to reinforce my learning and ensure better retention.