21ST CENTURY LEARNING DESIGN

Interactive Guide for Lecturers – Fun, Skills & Innovation

ZAIN RETAS

21ST CENTURY LEARNING DESIGN

Interactive Guide for Lecturers – Fun, Skills & Innovation

First Edition 2025

ZAIN RETAS

Publish by:
Politeknik Merlimau Melaka
KM 15, Jalan Jasin
77300 Merlimau Melaka

Tel : 06-2636687
Fax: 06-2636678
Website: www.pmm.mypolycc.edu.my

EDITORIAL BOARD

ACKNOWLEDGEMENT

Alhamdulillah, with the grace of Allah SWT, this eBook 21st Century Learning Design (21CLD) has been successfully completed.

I would like to extend my sincere gratitude to the Unit Centre for Artificial Intelegence Digital Learning (AiDEC) for their continuous support in the development of this material. My appreciation also goes to fellow colleagues and subject matter experts who have contributed ideas, feedback, and encouragement throughout the preparation of this eBook.

Special thanks to my students, whose active participation and enthusiasm have inspired the integration of interactive content, skill-building activities, and workshops featured in this eBook. Their curiosity and creativity continue to be the driving force behind the design of meaningful and engaging learning experiences.

Finally, I wish to acknowledge my family and friends for their unwavering support and motivation. Without their prayers and encouragement, this effort would not have been possible.

It is my hope that this eBook will serve as a useful guide and resource for lecturers, helping them to explore innovative approaches in teaching and learning while nurturing the skills needed for the 21st century.

PREFACE

All praise to Allah Almighty, for with His will this eBook **21st Century Learning Design (21CLD)** has been successfully developed as a practical resource to support lecturers in creating more meaningful, engaging, and impactful teaching and learning experiences.

This eBook is designed to guide lecturers through interactive content, fun activities, skill-building exercises, workshops, and best practices that can be directly applied in the classroom. Each section is structured to help educators understand the 21CLD framework, integrate 21st-century skills, and strengthen innovative and collaborative pedagogical approaches.

In addition, this eBook provides examples of activities, case studies, and reflective tasks aimed at fostering higher-order thinking, collaboration, effective communication, creativity, and self-directed learning. Through a balance of theory, hands-on practice, and workshop-style learning, lecturers are empowered to enhance the effectiveness of their teaching while creating enjoyable learning experiences for their students.

It is my sincere hope that this eBook will serve as a valuable guide, inspire new explorations, and support lecturers in nurturing learners who are not only knowledgeable but also competitive and well-prepared for the challenges of the future.

TABLE OF CONTENTS

21st Century Learning Design (21CLD)

Week 1

WEEK 1

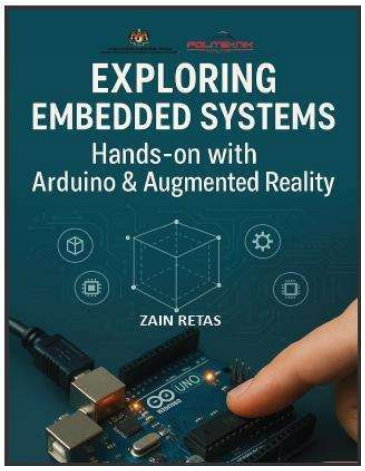
Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	none
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	Microcontroller and Embedded System 1.1 Explain microcontroller based system.
Learning :	Before class: Student should be able to: 1. Access and review provided digital resources (eBook, videos, or notes) on: • Definition of Microcontroller • Examples of Microcontroller applications • Difference between Microcontroller and Microprocessor • Basic concept of Embedded Systems 3. Identify key components commonly found in a Microcontroller (CPU, Memory, I/O ports, etc.). 4. Explore real-world examples of embedded systems in daily life (e.g., home appliances, automotive, medical devices) During class: Student should be able to: 1. Explain in their own words the definition and concept of Microcontroller and Embedded System 2. Compare Microcontroller vs. Microprocessor through class discussion 3. Identify and label the main parts of a typical Microcontroller using diagrams 4. Discuss and present examples of Embedded Systems based on their pre-class exploration 5. Participate in a short quiz or interactive activity to reinforce understanding
Panel Industry Details:	PROF. MADYA IR. DR. AHMAD ZAKI BIN HAJI SHUKOR
Panel Lecturer Details:	Zain binti Retas

WEEK 1

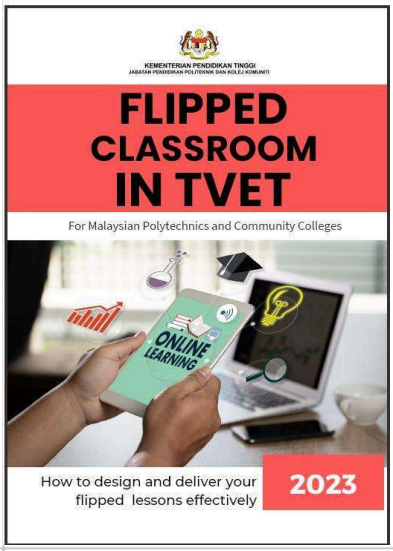
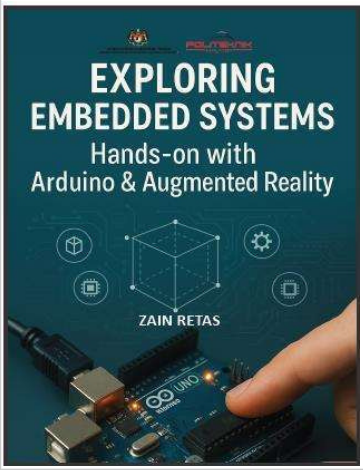
Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	**15 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
1. Lecturer share ebook.	• Definition of Microcontroller • Examples of Microcontroller applications • Difference between Microcontroller and Microprocessor • Basic concept of Embedded Systems	Padlet	Student read a ebook and watch a video
Phase	**Demonstration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
1. Lecturer share a Interactive Comparison with Microprocessor. 2. Lecturer ask the student, identify Real-world Embedded System Examples.	Note • compare Microcontroller vs. Microprocessor • Highlight size, integration, and application differences • Bring small like a digital thermometer, remote control, or simple automation device • Explain briefly how a Microcontroller is working inside •	Padlet	• Ask students to look around the classroom or think of examples from their home • Each group suggests 1 everyday item they believe contains a Microcontroller • Example answers: Microwave, Smartwatch, Washing Machine, Car remote key •

WEEK 1

Phase	Integration	Duration	30 minutes
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**

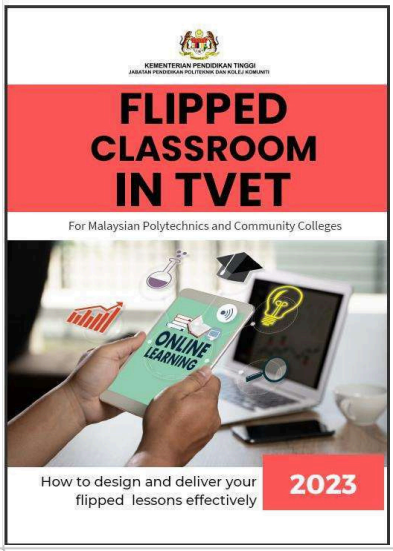
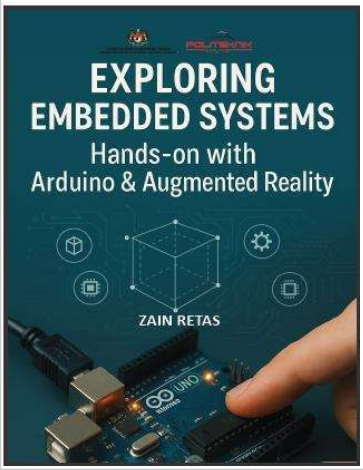
Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**
1. By the end of this activity, students should: • Clearly differentiate Microcontroller vs. Microprocessor • Understand basic Embedded System applications • Relate theoretical knowledge to real-life devices	1. eBook  2.	1.	i. Post a note from video in i. https://padlet.com/zainretas2018/djm50122-embedded-system-application-wymq1cusjjah i.

WEEK 1

References:		https://polycc.cidos.edu.my/download	/.
		https://anyflip.com/ezvdj/xcaq/	/.

Week 2

WEEK 1

References:		https://polycc.cidos.edu.my/download	/.
		https://anyflip.com/ezvdj/xcaq/	/.

WEEK 2

Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	none
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	Microcontroller and Embedded System 1.2 Explain the concept of embedded system
Learning Outcome:	Before class: Student should be able to: 1. Watch a short video or animation explaining: 2. What is an Embedded System 3. Simple real-life examples (e.g., washing machine, digital thermometer, remote control) 4. Read 1-2 pages from the provided eBook or notes covering: 5. Definition and characteristics of Embedded Systems 6. Examples of everyday devices containing Embedded Systems During class: Student should be able to: Step 1: Quick Recap (5 mins) 1. Instructor briefly reviews the concept based on students' pre-class materials. 2. Show 2-3 real or image examples Step 2: Group Activity – "Where's the Embedded System?" (15 mins) • Students work in pairs or small groups • Instructor shows 5 devices (pictures) • For each device, students discuss: o Does it have an Embedded System? o What task does it perform?

WEEK 2

Panel Industry Details:	PROF. MADYA IR. DR. AHMAD ZAKI BIN HAJI SHUKOR
Panel Lecturer Details:	Zain binti Retas

Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	**15 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
1. Lecturer share ebook.	• Definition of Microcontroller • Examples of Microcontroller applications • Difference between Microcontroller and Microprocessor • Basic concept of Embedded Systems	Padlet	Student read a ebook and watch a video

WEEK

Phase	Demonstration	Duration	30 minutes
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
1. Lecturer share a Interactive Comparison with Microprocessor. 2. Lecturer ask the student, identify Real-world Embedded System Examples.	Note **Step 1: Quick Recap (5 mins)** • Instructor briefly reviews the concept based on students' pre-class materials • Show 2-3 real or image examples **Step 2: Group Activity – "Where's the Embedded System?" (15 mins)** • Students work in pairs or small groups • Instructor shows 5 devices (physical or pictures) • For each device, students discuss: a. Does it have an Embedded System? b. What task does it perform? •	Padlet	
Phase	**Integration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Step 3: Class Discussion & Clarification (15 mins) i. Tinkercad Circuits Simulation i. Physical Arduino Board	• Simulate a simple Arduino (Microcontroller) • Show how inputs (e.g., button press) and outputs (e.g., LED light) work • Demonstrate code upload and observe real-time response • Show the actual Arduino UNO or similar Microcontroller • Point out key components (CPU, I/O		Share in padlet week 2 Tinkercad Circuits Simulation Physical Arduino Board Demonstration

Demonstration	pins, power supply) • Run a basic program like LED blink to demonstrate embedded system behavior		

Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**
By the end of this activity, students should: • Show schematic diagram of an Embedded System • Highlight Microcontroller as the control "brain" i.	Quick reflection task: • List 2 devices at home they think contain an Embedded System (E.g., Microwave, Car Remote, Smartwatch)	1.	i. Post a video schematic to padlet https://padlet.com/zainretas2018/djm50122-embedded-system-application-wymq1cusjjah

Week 3

WEEK 3

Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	none
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	Microcontroller Architecture 2.1 Describe the PIC16 Architecture
Learning Outcome :	Before class: Student should be able to: • Manipulate Circuit Simulation Fundamental During class: Student should be able to: • Recognize basic microcontroller circuit components & Sketch schematic using simulation software After Class: • Simulate the running circuit
Panel Industry Details:	PROF. MADYA IR. DR. AHMAD ZAKI BIN HAJI SHUKOR
Panel Lecturer Details:	Zain binti Retas

WEEK 3

Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	**15 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
1 Lecturer share video about Circuit Simulation Fundamental : LEDs & Breadboards With Arduino in Tinkercad		ThinkerCAD	Student watch the video and take a note from the video.
2. Lecturer share a video and student , identify the Blink component.		**ThinkerCAD**	Student watch the video and take a note from the video.
Phase	**Demonstration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation		**Platform/ Tools/ Equipment/Application**	**Student Activity**
Lecturer ask the student **to do the** Simulate the running simulation function in ThinkerCAD based on video given		ThinkerCAD	Student screen recording the circuit they to in Week 3 Padlet https://padlet.com/zainretas2018/djm50122-embedded-system-

<table>
<tr><td></td><td></td><td></td><td>application-wymq1cusjjah</td></tr>
<tr><td></td><td></td><td>ThinkerCAD</td><td>Student screen recording the circuit they to in Week 3 Padlet
Sesi II 2021/2022 DGI40122 EMBEDDED ROBOTIC (padlet.com)</td></tr>
<tr><td>Phase</td><td>Integration</td><td>Duration</td><td>30 minutes</td></tr>
<tr><td>Instructor Role/ Steps/ Preparation</td><td>Materials/ Resources</td><td>Platform/ Tools/ Equipment/Application</td><td>Student Activity</td></tr>
<tr><td>Step 3: Class Discussion & Clarification
a. Physical Arduino Board Demonstration</td><td>• Arduino UNO (or similar Arduino board) USB cable for power connection
• LED, Resistor, Breadboard (optional for LED blink demo)
• Projector or document camera (for large classroom visibility)</td><td>(15 mins)</td><td>Physical Arduino Board Demonstration</td></tr>
<tr><td>b. Identify & Explain Key Components</td><td>Component — Explanation
Microcontroller IC — The "brain" – controls the board (ATmega328/32)
Digital & Analog Pins — Inputs/Outputs for sensors, buttons, LEDs
Power Supply Pins — For powering external components
Reset Button — Restarts the board
USB Port — For programming and power connection</td><td>(5 mins)</td><td>Physical Arduino Board Demonstration</td></tr>
<tr><td>c. Simple Working Demo – LED Blink</td><td>a. Connect Arduino to computer or power bank
b. Load basic Blink program (pre-prepared or live upload)
c. Explain that the Microcontroller reads instructions and turns LED on/off
d. Highlight: "This is a simple example</td><td>(5-8 mins)</td><td></td></tr>
</table>

	of how Microcontrollers control outputs based on program code."		

Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**
By the end of this activity, students should: Real-World Connection	Quick reflection task: • Link to daily devices: "Just like this LED is controlled, similar Microcontrollers work inside your Smartwatch, Remote Control, or Thermometer.	(2-3 mins)	Let a volunteer student press the reset button or connect/disconnect the LED

Week 4

WEEK 4

Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	none
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	Microcontroller Architecture 2.1 Describe the PIC16 Architecture
Learning Outcome:	Before class: Student should be able to: • Manipulate Circuit Simulation Fundamental During class: Student should be able to: • Recognize basic microcontroller circuit components & Sketch schematic using simulation software After Class: • Simulate the running circuit
Panel Industry Details:	PROF. MADYA IR. DR. AHMAD ZAKI BIN HAJI SHUKOR
Panel Lecturer Details:	Zain binti Retas

WEEK 4

Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	**15 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
1 Lecturer share video about Arduino in Tinkercad how to use seven segment displays with arduino in proteus	https://www.youtube.com/watch?v=hGj9aePKvHw Watch pre-recorded tutorial video: "Introduction to Seven Segment Display & Arduino Simulation in Proteus" Read provided notes explaining: • Types of Seven Segment Displays (Common Cathode/Anode) • Pin configuration and basic working principle	ThinkerCAD	Student watch the video and take a note from the video. Identify pin names (a-g, dp) Briefly sketch a Seven Segment Display with labeled pins
Phase	**Demonstration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation		**Platform/ Tools/ Equipment/Application**	**Student Activity**
Lecturer ask the student **to do** the Simulate the running simulation function in Proteus based on video given	https://www.youtube.com/watch?v=hGj9aePKvHw	Proteus	Student screen recording the circuit they to in Week 3 Padlet Sesi II 2021/2022 DGI40122 EMBEDDED ROBOTIC (padlet.com)

WEEK 4

Quick recap on Seven Segment Display principles	Instructor Introduction	5 mins	
Instructor shows full wiring in Proteus	Demonstration (Proteus Simulation)	10 mins	
Explain basic code to display digit "3"	Arduino Code Walkthrough	5 mins	
Students: Build circuit in Proteus, load code, test	Student Hands-On Activity	25 mins	
Instructor assists and clarifies doubts	Q&A and Troubleshooting	10 mins	
Phase	**Integration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Lecturer's Preparation: Prepare Proteus project file with Arduino and Seven Segment Display pre-configured.	For Simulation: • Proteus Software • Arduino IDE • Arduino UNO model in Proteus • Seven Segment Display (Common Cathode recommended) • Resistors (virtual in Proteus) For Classroom: Projector Visual aids (pin configuration diagrams) Pre-class tutorial video	15 minutes	Modify the Arduino code to display digits "0-9" sequentially
Prepare Arduino source code (.ino) and compiled hex file.		10 - 15 minutes	Run and test simulation in Proteus
Print and share pre-class notes and video link with students.		5 - 10 minutes	Submit screenshot or short video as proof of working simulation
Ensure Proteus and Arduino IDE are properly installed and functional on all lab computers.			
Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**

WEEK 4

Lecturer Preparation Tasks Prepare Proteus project file with Arduino and Seven Segment Display ready for simulation.	For Simulation: • Proteus Software • Arduino IDE • Arduino UNO model in Proteus • Seven Segment Display (Common Cathode recommended) • Resistors (virtual in Proteus) For Classroom: • Projector • Visual aids (pin configuration diagrams) • Pre-class tutorial video		Modify the Arduino code to display digits "0-9" sequentially
Prepare Arduino source code (.ino) and compile the corresponding .hex file.			Run and test simulation in Proteus
Print and share pre-class notes and video link with students.			Submit screenshot or short video as proof of working simulation in padlet. Sesi II 2021/2022 DGI40122 EMBEDDED ROBOTIC (padlet.com)
Ensure Proteus software and Arduino IDE are installed and functioning on all laboratory computers.			

Week 5

WEEK 4

Lecturer Preparation Tasks Prepare Proteus project file with Arduino and Seven Segment Display ready for simulation.	For Simulation: • Proteus Software • Arduino IDE • Arduino UNO model in Proteus • Seven Segment Display (Common Cathode recommended) • Resistors (virtual in Proteus) For Classroom: • Projector • Visual aids (pin configuration diagrams) • Pre-class tutorial video		Modify the Arduino code to display digits "0-9" sequentially
Prepare Arduino source code (.ino) and compile the corresponding .hex file.			Run and test simulation in Proteus
Print and share pre-class notes and video link with students.			Submit screenshot or short video as proof of working simulation in padlet. Sesi II 2021/2022 DGI40122 EMBEDDED ROBOTIC (padlet.com)
Ensure Proteus software and Arduino IDE are installed and functioning on all laboratory computers.			

WEEK 5

Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	none
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	PIC Programming in C 3.1 Explain C programming for embedded system
Learning Objective:	Before class: Student should be able to: • Manipulate Circuit Simulation Fundamental During class: Student should be able to: • Recognize basic microcontroller circuit components & Sketch schematic using simulation software After Class: • Simulate the running circuit
Panel Industry Details:	
Panel Lecturer Details:	Zain binti Retas

WEEK 5

Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	**15 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
1 Lecturer share Prepare and share pre-class materials (eBook, notes, video link)	Pre-recorded video: Introduction to C Programming for Embedded System - PDF notes on C syntax basics (variables, functions, structure) - Example Arduino Blink code		
Phase	**Demonstration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation			
Explain C programming structure for embedded system (specific to Arduino)	- Arduino IDE - Proteus simulation (if applicable) - Sample C code for embedded system (e.g., LED blink, simple counter) - Projector for live coding demo https://www.youtube.com/watch?v=FGPyBrgruBw&t=1s **How to generate Hex file in Arduino for Proteus Simulation** https://www.youtube.com/watch?v=akosjzrTmlg**How to simulate Arduino in Proteus, generate .hex file from Arduino IDE**	55 - 60 minutes	- Participate in code walkthrough - Identify C code Hex File components in provided example

Phase	**Integration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
- Provide optional challenge task (e.g., write code to blink LED in sequence)	- Challenge task instruction (online or printed)		- Modify provided code (e.g., change LED blinking pattern) - Test code on Arduino or in Proteus
			- Ask questions during troubleshooting
Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**
- Review submitted work (if applicable)	- Access to Arduino IDE/Proteus (if students wish to practice at home)	30 - 45 minutes	- Complete challenge task (e.g., modify code to create LED running pattern)
- Provide feedback or clarification in next class |			- Submit screenshot/video as proof (optional)
			- Prepare questions for next class if facing difficulties

Week 6

WEEK 6

Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	none
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	PIC Programming in C 3.1 Explain C programming for embedded system
Learning Objective:	Before class: Student should be able to: • Manipulate Circuit Simulation Fundamental During class: Student should be able to: • Recognize basic microcontroller circuit components & Sketch schematic using simulation software After Class: • Simulate the running circuit
Panel Industry Details:	
Panel Lecturer Details:	Zain binti Retas

WEEK 6

Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	**15 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
1 Lecturer share Prepare and share pre-class materials (eBook, notes, video link)	Pre-recorded video: Introduction to C Programming for Embedded System - EBook topic 7 Segment and LCD Display - PDF notes on C syntax basics (variables, functions, structure) - - Notes/diagrams showing 7 Segment Display pins, LCD (16x2) pins, and examples - Proteus circuit example files	20 - 30 minutes	- Watch video and read notes - Sketch basic pinout for 7 Segment and LCD Displays - Reflect: List 2 devices at home that use these displays (e.g., microwave, calculator)
Phase	**Demonstration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation			
Explain C programming structure for embedded system (specific to Arduino) - Provide pre-class video explaining basic concepts of 7 Segment and LCD Displays - Recap working principles of 7 Segment and LCD Displays	- Arduino IDE - Proteus simulation (if applicable) - Sample C code for embedded system - Projector for live coding demo https://www.youtube.com/watch?v=V2neQEZWK_w&list=PLXY4_qxp8fUfco1iyezibCLPIS_VLRh66 Proteus arduino https://www.youtube.com/watch?v=ZXgXuFWsdEM&list=PLXY4_qxp8fUfco1iyezibCLPIS_VLRh66&t=3s Arduino to 16*2 LCD Display Proteus - Arduino Proteus Simulation tutorial # 5	55 - 60 minutes	- Participate in code walkthrough - Identify C code Hex File components in provided example Observe live demonstration of 7 Segment and LCD Displays - Participate in discussion and Q&A - Hands-on task: . Build 7 Segment circuit in Proteus .Upload and test

			Arduino code to display digits .Build LCD circuit in Proteus .Upload and test code to display custom message -.Troubleshoot and modify code if needed
Phase	**Integration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
- Instructor explains how both displays can work together in one embedded system	- 7 Segment shows numeric output (e.g., counter) - LCD shows detailed message (e.g., "Counter Running")		- Instructor shows combined project in Proteus or hardware - Students observe and discuss real-world applications (e.g., digital clock, temperature display)
Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**
- Provide practice task:	- Task sheet with clear instructions - Access to Proteus and Arduino IDE (optional at home or in next lab)	30 - 45 minutes	- Complete practical task: . Modify code to run 7 Segment & LCD together . Submit screenshot or video proof - Reflect on challenges faced and prepare questions for next class

Week 9

WEEK 7

Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	none
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	4.0 Input Output ports programming 4.1 Explain the Features and Functions the Input Output (I/O) ports
Learning Objective:	Before class: Student should be able to: • Manipulate Circuit Simulation Fundamental During class: Student should be able to: • Recognize basic microcontroller circuit components & Sketch schematic using simulation software After Class: • Simulate the running circuit
Panel Industry Details:	
Panel Lecturer Details:	Zain binti Retas

WEEK 7

Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	**15 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
1 Lecturer share Prepare and share pre-class materials (eBook, notes, video link)	- Share video on I/O ports, DC Motor control using L293D Motor Driver, and LDR sensor basics - Pre-recorded video: Intro to I/O Ports & Basic Sensor/Actuator Control - Notes/diagrams of L293D wiring, LDR connection to Arduino - Optional Proteus preview files	20 - 30 minutes	- Watch video and read notes - Sketch basic I/O connection for DC Motor (with L293D) and LDR - Answer reflection questions: 1. What type of I/O ports control DC Motor? 2. How does the LDR provide input to Arduino?
Phase	**Demonstration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation			
Recap Arduino I/O port functions	- Arduino Board - L293D Motor Driver IC - DC Motor - LDR (Light Dependent Resistor) - Breadboard, jumper wires - Proteus software & Arduino IDE - Sample Arduino code (Motor control & LDR reading)	60 - 90 minutes	- Observe live demonstration of motor control using L293D - Identify I/O pins used for motor control and sensor input - Hands-on task: . Build DC Motor circuit with L293D in Proteus or physical hardware . Test code to control motor (e.g., motor ON/OFF, direction control)

			. Connect LDR, test code to read LDR value via Serial Monitor - Troubleshoot and ask questionsArduino code to display digits .Build LCD circuit in Proteus .Upload and test code to display custom message -.Troubleshoot and modify code if needed
Phase	**Integration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
.	- Challenge task instructions - Access to Proteus, Arduino IDE (optional for practice)	30 - 45 minutes	- Complete practical task: . Modify code to create light-controlled motor system . Submit screenshot/video as proof - Reflect on learning and challenges
Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**
Optional Real-World Applications for Student Discussion	.Automatic fan or window system based on sunlight .Smart car headlight control using LDR .Motorized blinds responding to light intensity	30 - 45 minutes	

Week 10

WEEK 8

Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	none
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	4.0 Input Output ports programming 4.1 Explain the Features and Functions the Input Output (I/O) ports Sensor and Actuator
Learning Objective:	Before class: Student should be able to: • Manipulate Circuit Simulation Fundamental During class: Student should be able to: • Recognize basic microcontroller circuit components & Sketch schematic using simulation software After Class: • Simulate the running circuit
Panel Industry Details:	
Panel Lecturer Details:	Zain binti Retas

WEEK 8

Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	**15 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
- Provide pre-class video explaining:	. Basic concept of I/O ports on Arduino . Difference between Input (Sensor) and Output (Actuator) - Share short notes with simple diagrams (sensor to input pin, actuator to output pin) - Pre-recorded video: Introduction to Arduino I/O Ports — Sensors & Actuators - Short notes/handout: I/O port functions, examples (LED, LDR, Motor)	20 - 30 minutes	- Watch video and read notes - Sketch examples of at least one sensor and one actuator connection to Arduino - Reflect: Identify 2 devices at home that use both sensors and actuators (e.g., washing machine, automatic fan)
Phase	**Demonstration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation			
Recap Arduino I/O port functions - Explain how sensors provide input (e.g., LDR, push button, temperature sensor) - Explain how actuators receive output (e.g., LED, Buzzer, Motor) - Demonstrate simple Arduino circuit: .Sensor provides input	- Arduino Board - Sensors (LDR, Push Button, etc.) - Actuators (LED, Buzzer, DC Motor) - Breadboard, jumper wires - Arduino IDE - Proteus software (optional for simulation) - Example Arduino code (simple I/O control)	60 - 90 minutes	- Observe live demonstration - Participate in discussion: Identify I/O pins in the circuit - Hands-on Practical Task: . Build circuit with one sensor & one actuator . Upload and test Arduino code . Modify code to change actuator behavior based on sensor input

WEEK 8

Arduino processes input .Actuator responds accordingly - Guide students in hands-on practical tasks			- Troubleshoot and clarify doubts
Phase	**Integration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
.- Lecturer demonstrates combining sensor input and actuator output into a basic control system	- LDR controls LED or Buzzer based on light level	10 - 15 mins (within class)	- Instructor shows project in Proteus or real hardware - Students discuss other real-world systems applying similar logic (e.g., alarm system, automatic lighting)
Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**
- Provide practice task: .Modify code so actuator behavior varies based on different sensor conditions - Collect proof of completion (screenshot or video) - Prepare feedback for next class |	. - Practice task instruction sheet - Access to Arduino IDE & Proteus (optional for practice) Example Real-World Systems for Student Discussion - Automatic light based on motion or light sensor - Smart fan triggered by temperature sensor - Door alarm activated by magnetic sensor	30 - 45 minutes	- Complete practical task: . Modify code to enhance sensor-actuator interaction . Submit screenshot or short video - Reflect on learning and challenges

Week 11

WEEK 9

Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	none
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	5.0 Hardware interfacing 5.1 Explain LCD interfacing
Learning Objective:	Before class: Student should be able to: • Manipulate Circuit Simulation Fundamental During class: Student should be able to: • Recognize basic microcontroller circuit components & Sketch schematic using simulation software After Class: • Simulate the running circuit
Panel Industry Details:	
Panel Lecturer Details:	Zain binti Retas

WEEK 9

Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	**15 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
- Provide short video introduction to LCD (16x2) interfacing with Arduino - Share notes covering: .LCD 16x2 pinout .Basic working principle .Typical applications of LCD displays - Optional: Share example Arduino code for LCD display (e.g., “Hello World”):	- Pre-recorded video: Introduction to LCD Interfacing with Arduino https://www.youtube.com/watch?v=ZXgXuFWsdEM&list=PLXY4_qxp8fUfco1iyezibCLPIS_VLRh66&t=8s - Notes with LCD pin diagram and connection example - Arduino example code for LCD display	20 - 30 minutes	- Watch video and read notes - Sketch LCD 16x2 pin configuration - Reflect: Identify 2 real-world devices that use LCD displays (e.g., microwave, digital clock))
Phase	**Demonstration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation			
- Recap LCD function and pin description - Explain connection between Arduino and LCD (focus on RS, EN, D4-D7 pins) - Demonstrate live LCD interfacing using: .Proteus simulation or .Physical Arduino and LCD hardware - Explain Arduino	- Arduino UNO or similar board - LCD 16x2 Display - Potentiometer (for contrast adjustment) - Breadboard and jumper wires - Arduino IDE - Proteus simulation software (optional) - Example Arduino code to display text on LCD	60 - 90 minutes	- Observe live demonstration of LCD connection and display output - Identify correct pins for LCD control on Arduino - Hands-on Practical Task: . Connect LCD to Arduino (either in Proteus or physically)

code structure for LCD control - Guide students through hands-on practical tasks			. Upload and test basic LCD code (display name, “Hello World”, etc.) Modify code to display custom message - Troubleshoot and clarify doubts
Phase	**Integration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
- Lecturer demonstrates integrating LCD with other components	- Example: Display sensor reading (e.g., LDR or temperature sensor value) on LCD	10 - 15 mins (within class)	- Students observe, discuss potential real-world applications (e.g., digital thermometer, home automation display)
Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**
- Provide practice task: . Modify LCD code to create scrolling text or display sensor value - Collect proof of task completion (screenshot or video) - Provide feedback in next class |	- Practice task instructions - Arduino IDE and Proteus simulation access	30 - 45 minutes	- Complete practical task: . Modify code to enhance LCD output (scrolling text or sensor integration) . Submit screenshot or video proof - Prepare questions for next class

Week 12

WEEK 10

Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	none
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	5.0 Hardware interfacing 5.2 Explain AD C, DAC and sensor interfacing
Learning Objective:	Before class: Student should be able to: • Manipulate Circuit Simulation Fundamental During class: Student should be able to: • Recognize basic microcontroller circuit components & Sketch schematic using simulation software After Class: • Simulate the running circuit
Panel Industry Details:	
Panel Lecturer Details:	Zain binti Retas

WEEK 10

Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	**15 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
- Prepare and share pre-class materials: .Short video explaining ADC, DAC, and basic sensor interfacing .Notes on: .Difference between analog and digital signals .Function of Arduino's Analog Input (ADC) .Concept of DAC (using PWM in Arduino) .Example of sensor-based applications - Optional: Share Proteus preview file of LDR or potentiometer connected to Arduino | |	- Pre-recorded video: Introduction to ADC, DAC, and Sensor Interfacing https://www.youtube.com/watch?v=RXtWqA6BmF0 - Notes/handouts with clear diagrams of Analog Input pins and wiring examples - Optional Proteus preview simulation file	20 - 30 minutes	- Watch video and read notes - Sketch a basic diagram showing an analog sensor (e.g., LDR, potentiometer) connected to Arduino - Reflect on: ✔ What is the difference between analog and digital signals? ✔ Identify 2 daily devices that involve analog sensors (e.g., light dimmer, temperature sensor)
Phase	**Demonstration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation			
- Recap key points from pre-class material - Explain: How ADC works in Arduino .Basic concept of DAC (focus on PWM) .Real-life examples of analog sensor interfacing - Demonstrate: .Analog sensor	- Arduino UNO or similar board - LDR or Potentiometer (for analog input) - LED, Buzzer, or Motor (for output) - Breadboard, jumper wires - Arduino IDE - Proteus simulation software (optional) - Example Arduino code for	60 - 90 minutes	- Observe demonstration of sensor interfacing and ADC/DAC application - Identify correct analog pins on Arduino - Hands-on Practical Task:

(LDR/Potentiometer) connected to Arduino .Reading analog values using `analogRead()` .Controlling output (e.g., LED brightness) using PWM with `analogWrite()` - Guide students through practical tasks in Proteus or physical setup |	analog reading and PWM output		.Connect LDR or Potentiometer to Analog Input .Upload and test Arduino code to read analog values .Modify code to control LED brightness based on sensor input (PWM output) - Troubleshoot and ask questions
Phase	**Integration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
| - Lecturer demonstrates integrating analog sensor input with output device using ADC/DAC principle	- Instructor shows integrated project in Proteus or hardware - Example project: ✔ LDR controls LED brightness ✔ Potentiometer controls motor speed	10 - 15 mins (within class)	- Students discuss other real-world applications (e.g., automatic fan speed, smart dimmers)
Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**
- Provide practice task: .Modify code to enhance sensor-output interaction .Example: LDR controls LED brightness with threshold detection .Optional: Include serial monitor output for debugging - Review submissions (screenshot/video) - Provide feedback in next session |	- Practice task instructions - Arduino IDE and Proteus simulation access Example Real-World Applications for Student Discussion .Smart lighting system (automatic brightness control). Temperature-based fan speed control Sound-controlled LED brightness or motor speed Light-sensitive alarm system	30 - 45 minutes	- Complete practical task: . Modify code to enhance LCD output (scrolling text or sensor integration) . Submit screenshot or video proof - Prepare questions for next class

Week 13

WEEK 11

Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	none
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	5.0 Hardware interfacing 5.2 Explain AD C, DAC and sensor interfacing -Building and Programming a Line Follower Robot using ESP32/Arduino UNO with Arduino IDE
Learning Objective:	Before class: Student should be able to: • Manipulate Circuit Simulation Fundamental During class: Student should be able to: • Recognize basic microcontroller circuit components & Sketch schematic using simulation software After Class: • Simulate the running circuit
Panel Industry Details:	
Panel Lecturer Details:	Zain binti Retas

WEEK 11

Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	20 - 30 minutes
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
- Provide pre-class video tutorial on Line Follower Robot principles	Pre-recorded video: Introduction to Line Follower Robot - Notes/handout with block diagram and component explanation - Arduino IDE download and ESP32 library guide		- Watch video and read notes - Sketch basic block diagram of Line Follower Robot - Reflect: ✔ What is the function of IR sensors? ✔ How does the robot decide to move left or right?
Phase	**Demonstration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation			
- Recap hardware components: ESP32/Arduino UNO, IR sensors, Motor Driver, DC motors - Explain and demonstrate: .Wiring of IR sensors, motor driver, and motors .Arduino IDE code structure .How sensor input affects motor control logic - Guide students through: .Circuit assembly .Code upload and	- ESP32/Arduino UNO - L298N Motor Driver - IR sensors - DC Motors - Robot chassis with wheels - Breadboard, jumper wires, batteries - Arduino IDE installed on laptops - Pre-defined black line track		- Assemble robot hardware - Wire sensors and motor driver according to instructions - Upload provided sample code to microcontroller - Test and observe robot movement on the track - Troubleshoot if robot fails to follow the line

troubleshooting .Testing the robot on the track			- Ask questions and modify code if needed
Phase	**Integration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
- Lecturer emphasizes integration of sensor input with motor control logic in real time - Demonstrate: ✔ How IR sensor feedback controls motor direction ✔ Robot behavior during left/right turns and stops	- Instructor shows real-time robot adjustments based on sensor data		- Students observe and explain how programming affects robot performance - Discussion on sensor placement, motor speed, and environmental impact
Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**
- Provide improvement challenge: .Students modify the code or sensor placement for better performance .Example: Adjust turning logic or increase sensor count - Collect student videos or photos showing improved robot performance - Provide feedback in next class |	- Challenge task instructions - Access to Arduino IDE and necessary hardware ☑ Example Learning Outcomes Reinforced ✔ Ability to program and control robot using Arduino IDE ✔ Understanding of sensor and motor interfacing ✔ Practical troubleshooting and improvement of autonomous robot behavior	30 - 60 minutes	- Modify code or hardware based on test results - Re-test robot on track - Record video showing improved performance - Submit video and short explanation

Week 14

WEEK 12

Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	none
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	5.0 Hardware interfacing 5.3 Describe Relay interface and Motor Control Relay Interface and Motor Control using Microcontroller (Arduino/ESP32)
Learning Objective:	Before class: Student should be able to: • Understand basic relay operation and motor control principles. During class: Student should be able to: • Observe a live demonstration, assemble the relay circuit, and control a motor using Arduino code. After Class: • Complete a practical task by modifying the code to control the motor with a sensor input and submit proof of their work.
Panel Industry Details:	
Panel Lecturer Details:	Zain binti Retas

WEEK 12

Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	20 - 30 minutes
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
- Provide pre-class video or animation explaining: ✓ What is a relay and how it works ✓ Basic motor control using a relay ✓ Safety considerations for relay circuits - Share notes on: ✓ Relay pin configuration and wiring ✓ Motor control principles using relay interface - Optional: Share example Proteus circuit diagram	- Pre-recorded video: *Introduction to Relay and Motor Control* - Notes/handouts with relay circuit diagrams and applications - Optional Proteus simulation file preview		- Watch video and read notes - Sketch basic relay circuit for motor control - Reflect: ✓ Why use a relay to control motors? ✓ Mention at least one real-world device that uses a relay for motor control (e.g., washing machine, industrial machine)
Phase	**Demonstration**	**Duration**	**minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**60 - 90 minutes**
Instructor Role/ Steps/ Preparation			
- Recap relay function and importance of electrical isolation - Explain relay connection to Arduino (low voltage control for high voltage device) - Demonstrate: ✓ Relay control using Arduino digital output ✓ Motor connected to relay, turning ON/OFF via code - Discuss safety	- Arduino UNO or ESP32 - Relay Module - DC Motor or other load - Breadboard, jumper wires - Arduino IDE - Proteus simulation software (optional) - Example Arduino code for relay control	60 - 90 minutes	- Observe demonstration of relay and motor control circuit - Identify correct wiring of relay to Arduino and motor - Hands-on Practical Task: ✓ Connect relay to Arduino output pin

procedures for relay handling - Guide students through practical tasks .Code upload and troubleshooting .Testing the robot on the track			✓ Connect motor to relay as load ✓ Upload code to turn motor ON/OFF via relay ✓ Modify code for different control patterns - Troubleshoot and ask questions
Phase	**Integration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
- Lecturer demonstrates integrating relay-controlled motor with other inputs (e.g., sensor or button)	Example project: ✓ Button or sensor triggers relay to turn motor ON/OFF		- Instructor shows integrated project in Proteus or hardware - Students discuss other real-world relay applications (e.g., pump control, smart home appliances)
Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**
- Provide improvement challenge: .✓ Modify code to control motor using relay based on sensor input (e.g., LDR or temperature sensor) - Review student submissions (video or photo proof) - Provide feedback in next session	- Challenge task instructions - Access to Arduino IDE and necessary hardware ✅ **Example Real-World Applications for Discussion** • Automatic pump control • Smart home fan or appliance switching • Industrial motor or machine control with electrical isolation • Vehicle starter motor control	30 - 45 minutes	- Complete practical task: ✓ Modify and test relay-motor control with sensor input ✓ Submit screenshot or short video - Reflect on learning and prepare questions

Week 7

WEEK 13

Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	None
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	6.0 Embedded System application 6.1 Discuss embedded system application
Learning Objective:	Before class: Student should be able to: • Explore real-world examples of embedded systems by watching a video and identifying at least three applications from daily life. During class: Student should be able to: • Participate in a discussion on various embedded system applications across industries and present one application with its components and functions. After Class: • Complete a short reflective task or case study describing an embedded system they use daily and suggest possible improvements.
Panel Industry Details:	
Panel Lecturer Details:	Zain binti Retas

WEEK 13

Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	20 - 30 minutes
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
- Share pre-class video and notes introducing real-world embedded system applications Provide reflection questions to guide student understanding (e.g., "Where do you see embedded systems in your daily life?")	- Video: *Introduction to Embedded System Applications* - Notes or infographic with application domains (automotive, home, medical, industrial, etc.)		- Watch video and read notes - Identify and write down at least 3 examples of embedded systems from their surroundings
Phase	**Demonstration**	**Duration**	**Minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**60 - 90 minutes**
Instructor Role/ Steps/ Preparation			
- Facilitate a group discussion or Think-Pair-Share session - Guide students to categorize applications based on industry (e.g., healthcare, automotive, smart home) - Highlight real projects or devices (e.g., smartwatches, traffic light control, medical monitors) - Encourage student sharing and brainstorming	- Projector/slides showing case studies and real-world systems - Whiteboard or digital board for listing ideas - Sample embedded system hardware	60 - 90 minutes	- Participate in class discussion - Share examples from reflection activity - Work in small groups to present a chosen application, explaining: ✓ Purpose ✓ Key components ✓ Expected behavior ✓ Industry used

WEEK 13

Phase	Integration	Duration	30 minutes
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
- Instructor demonstrates how various embedded systems share common architecture (input → process → output)			- Students map their selected example to the embedded system block diagram - Discuss similarities and differences between applications across industries
Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**
- Assign a case-based reflection task - Provide feedback on group presentations or individual responses	- Case study template or online submission form - Optional reading/resource links for deeper exploration	30 - 45 minutes	- Submit a short report or video explaining one embedded system they use often - Reflect on how the system works and propose one possible improvement or upgrade

Week 8

WEEK 14

Lesson plan template	
Programme Name	DIPLOMA IN MECHATRONIC ENGINEERING
Course Code Course Name	DJM 50122 | EMBEDDED SYSTEM APPLICATION
Justification for flipping the module	1. To teaching and learning this topic is more interesting because there are many new terms and need to be understood there is no misinterpretation. 2. To increase the level of understanding of students for this particular topic and relation to real life situation
Prerequisite skills or knowledge:	None
Lesson Synopsis:	EMBEDDED SYSTEM APPLICATION covers the basic concept and application of microcontroller system and embedded system. Students will be able learn programming and hardware on embedded development system and understand how to interface.
Lesson Title	6.0 Embedded System application 6.1 Discuss embedded system application
Learning Objective:	Before class: Student should be able to: • Explore real-world examples of embedded systems. During class: Student should be able to: Participate in a discussion on Prepare Mini Sumo Robot Tournaments. After Class: • Complete a short reflective task or case study describing an embedded system they use daily and suggest possible improvements.
Panel Industry Details:	
Panel Lecturer Details:	Zain binti Retas

WEEK 14

Lesson Structure (LESSON 1)			
Component	Theoretical/Hands on Practical		
Session	**Before Class**		
Phase	**Activation**	**Duration:**	20 - 30 minutes
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/Application**	**Student Activity**
- Prepare Mini Sumo Robot Tournaments.	- Video: *Introduction to Embedded System Applications* - Notes or infographic with application domains (automotive, home, medical, industrial, etc.)		- Prepare Mini Sumo Robot Tournaments. - Identify and write down at least 3 examples of embedded systems from their surroundings
Phase	**Demonstration**	**Duration**	**Minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**
Phase	**Application**	**Duration**	**60 - 90 minutes**
Instructor Role/ Steps/ Preparation			
- Facilitate a group discussion or Think-Pair-Share session - Guide students to categorize applications based on industry (e.g., healthcare, automotive, smart home) - Highlight real projects - Encourage student sharing and brainstorming	- Projector/slides showing case studies and real-world systems - Whiteboard or digital board for listing ideas - Sample embedded system hardware	60 - 90 minutes	- Participate in Mini Sumo Robot Tournaments.
Phase	**Integration**	**Duration**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/ Resources**	**Platform/ Tools/ Equipment/Application**	**Student Activity**

WEEK 14

- Instructor demonstrates how various embedded systems share common architecture (input → process → output)			- Students map their selected example to the embedded system block diagram - Discuss similarities and differences between applications Mini Sumo Robot Tournement.
Lesson Structure(LESSON 1)			
Component	**Theoretical**/Hands on Practical		
Session	**After Class**		
Phase	**Reflection**	**Duration:**	**30 minutes**
Instructor Role/ Steps/ Preparation	**Materials/Resources**	**Platform/Tools/ Equipment/ Application**	**Student Activity**
- Assign a case-based reflection task - Provide feedback on group presentations or individual responses	- Case study template or online submission form - Optional reading/resource links for deeper exploration	30 - 45 minutes	- Submit a short report Mini Sumo Robot Tournement.

References

1.AnyFlip. (n.d.). 21st century learning design. https://anyflip.com/ezvdj/xcaq/

2.PolyCC CIDOS. (n.d.). Course learning materials. https://polycc.cidos.edu.my/download

3.YouTube. (2023). How to generate hex file in Arduino for Proteus simulation [Video]. YouTube. https://www.youtube.com/watch?v=FGPyBrgruBw

4.YouTube. (2023). Arduino to 16x2 LCD Display Proteus - Arduino Proteus Simulation tutorial #5 [Video]. YouTube. https://www.youtube.com/watch?v=ZXgXuFWsdEM

5.YouTube. (2023). Introduction to seven segment display & Arduino simulation in Proteus [Video]. YouTube. https://www.youtube.com/watch?v=hGj9aePKvHw

6.YouTube. (2023). Introduction to ADC, DAC, and sensor interfacing [Video]. YouTube. https://www.youtube.com/watch?v=RXtWqA6BmF0