Module Definition Form (MDF)

Module code: MOD009174	**Version:** 1 **Date Amended:** 07/Feb/2023

1. Module Title
Programming Methods for Robotics

2a. Module Leader
Shabnam Sadeghi Esfahlani

2b. School
School of Engineering and the Built Environment

2c. Faculty
Faculty of Science and Engineering

3a. Level
7

3b. Module Type
Standard (fine graded)

4a. Credits
15

4b. Study Hours
150

5. Restrictions

Type	Module Code	Module Name	Condition
Pre-requisites:	None		
Co-requisites:	None		
Exclusions:	None		
Courses to which this module is restricted:			

LEARNING, TEACHING AND ASSESSMENT INFORMATION

6a. Module Description

Robot Operating System (ROS) is a framework of tools, libraries, and software enabling robot software development. It is an adaptable system for programming robots and managing robotic platforms. In this module, you will learn to use Matlab and Simulink software to create a platform and practice the applications of operating hardware, robotic simulation models, path planning, localisation and mapping, and other algorithms. You will use ROS Toolbox, which allows them to access ROS functionality in MATLAB; MATLAB enables communication with a ROS network, interactively analyse robot abilities, and envision sensor data. You will develop robotics applications by exchanging data with ROS-enabled robots and robot simulators such as Gazebo. You will create Simulink models that exchange messages with a ROS network. Verify yourmodel within the Simulink environment by receiving and sending messages to ROS-enabled robots and robot simulators. You will also generate C++ code for a standalone ROS application.

6b. Outline Content

- Learning MATLAB functions and Simulink blocks to import, analyse, and playback recorded ROS data.
- Configure ROS Network addresses and connect to a ROS-enabled Robot from Simulink
- Connect to a ROS network and exchange data.
- Enable communication between devices and data exchange.
- Connect and access messages.
- Verify ROS nodes via simulation
- Connect to external robot simulators (i.e., Gazebo)
- Perform C++ code generation (with Simulink Coder) in Simulink and deploy them to simulated or physical hardware.
- Visualise messages and debug parameters while the model is running on hardware.

6c. Key Texts/Literature

The reading list to support this module is available at: https://readinglists.aru.ac.uk/

6d. Specialist Learning Resources

Matlab software and Simulink toolbox, sensors and actuators

7. Learning Outcomes (threshold standards)		
No.	**Type**	**On successful completion of this module the student will be expected to be able to:**
1	Knowledge and Understanding	Understand different classifications and applications of programming in the robotics industry
2	Intellectual, practical, affective and transferrable skills	Analyse complex robotic problems via programming and practice the skill to reach corroborated conclusions using mathematics and engineering principles.
3	Intellectual, practical, affective and transferrable skills	Identify and implement relevant computational and analytical techniques to benchmark complex robotic problems.
4	Intellectual, practical, affective and transferrable skills	Incorporate engineering judgment to perform with information uncertainties and discuss the employed techniques' limitations.

8a. Module Occurrence to which this MDF Refers				
Year	**Occurrence**	**Period**	**Location**	**Mode of Delivery**
2026/7	ZZF	Template For Face To Face Learning Delivery		Face to Face

8b. Learning Activities for the above Module Occurrence			
Learning Activities	**Hours**	**Learning Outcomes**	**Details of Duration, frequency and other comments**
Lectures	18	1-2	Three hours per week for 6 weeks
Other teacher managed learning	18	2-4	Three hours per week for 6 weeks
Student managed learning	114	1-4	Student-managed study, engage with team members and work towards the assessment elements
TOTAL:	150		

9. Assessment for the above Module Occurrence

The assessment for some modules includes a presentation (see below for details on how this particular module is assessed). We recognise that presenting, especially to a large group, can cause significant anxiety or distress for some students. If this may affect you, please contact your Module Leader or the Disability and Neurodiversity Service as early as possible. We will work with you to consider appropriate reasonable adjustments. Depending on your individual needs, these might include, for example, presenting to a smaller group or presenting only to a member of academic staff.

Assessment No.	Assessment Method	Learning Outcomes	Weighting (%)	Fine Grade or Pass/Fail	Qualifying Mark (%)
010	Coursework	3-4	50 (%)	Fine Grade	40 (%)
Individual Coursework on practical work (max 1500 words), maps to Engineering Council Learning Outcomes M3					

Assessment No.	Assessment Method	Learning Outcomes	Weighting (%)	Fine Grade or Pass/Fail	Qualifying Mark (%)
011	Coursework	1-2	50 (%)	Fine Grade	40 (%)
Individual Report on theoretical underpinning (max 1500 words), maps to Engineering Council Learning Outcomes M2					

In order to pass this module, students are required to achieve an overall mark of 40% (for modules at levels 3, 4, 5 and 6) or 50% (for modules at level 7*).

In addition, students are required to:
(a) achieve the qualifying mark for each element of fine graded assessment as specified above
(b) pass any pass/fail elements

[* the pass mark of 50% applies for all module occurrences from the academic year 2024/25 – see Section 3a of this MDF to check the level of the module and Section 8a of this MDF to check the academic year]