

Subject: 48623 – Mechatronics 2

Assessment #: 3

Assessment Title: *Embedded integration*

Student Number:

Student Name:

Declaration of Originality

The work contained in this assignment, other than that specifically attributed to another source, is that of the author(s). It is recognised that, should this declaration be found to be false, disciplinary action could be taken and the assignment of the student involved will be given zero marks. In the statement below, I have indicated the extent to which I have collaborated with other students, whom I have named.

Statement of Collaboration

--

Signature

--

Marks

LCD	/4
Control	/5
Sweep	/7
Wall follow	/9
System knowledge	5
TOTAL	/30

Aim

The aim of this assessment is to integrate the Assessment 2 with a robotic simulation tool to carry out basic robotic navigation tasks in a maze.

Requirements

You will be required to write source code for the micro-controller to carry out robotic tasks in a hardware in the loop simulation set up. The map is 20x20m and will be given to you (See Figure 1). The white spaces are empty (obstacle free) and dark parts are walls (obstacles). You will be provided with MATLAB interface and full documentation of the simulator.

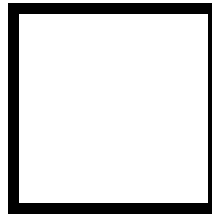


Figure 1: Map (box.png)

You are required to carry out the following operations through the Arduino kit.

- **Main menu**

When powered ON, the first line of the LCD should display the Student ID. The second line should display “Main menu”. The robot should be designed to perform the following operations: Control, Sweep and Wall follow. User should be able to select the above options through the second line of the LCD. The cursor should be in the second line. The Menu should be able to cycle through the ‘DOWN’ button. A particular mode should be able to select through the ‘SELECT’ button.

- **Control**

In the ‘Control’ mode, the system should take in the user input from the button press to drive the robot in the simulator map. LEFT and RIGHT buttons should rotate the robot in left and right directions respectively. UP and DOWN buttons should drive the robot forward and backward respectively. The robot should drive/rotate continuously on the chosen direction while the button is being pressed. The second line of the LCD should display “Control”. By pressing the SELECT button, the users should be able to exit this mode and return to the main menu.

- **Sweep**

Once in the Sweep mode, the second line of the LCD should display “Sweep”. Pressing the UP button should start the sensor autonomously to do a 360° clockwise (CW) scan of the surrounding area. The robot should be stationary. The robot then should find the nearest object and rotate itself counter clockwise (CCW) to face towards the identified shortest distance to the object. The robot will be tested in the given map may be with some random obstacles (see Figure 2).

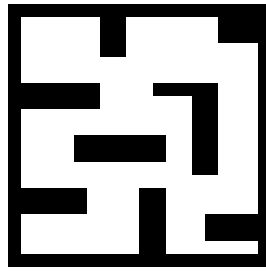


Figure 2: Map (map3_1.png)

Once the task is completed, pressing SELECT button should allow the users to exit this mode and return to the main menu.

- **Wall follow**

Robot should be in the box map (Figure 1) and when this option is triggered the robot needs to find the nearest wall using the ‘Sweep’ functionality. The robot should then autonomously move towards the wall and stop at 2 meters from the wall. The robot must then travel parallel to the wall while maintaining the 2m distance. The robot needs to keep moving while maintaining the 2m distance from all the walls until the user press the UP button to stop the robot (wall follow as well as sensor scanning). The second line of the LCD should display “Wall follow”. By pressing the SELECT button, the users should be able to exit this mode and return to the main menu. **Important:** The robot must not collide with any walls.

Notes

- The only equipment permitted for this assignment is:
 - 1x Arduino Uno
 - 1x 16x2 LCD Shield
 - MATLAB robot simulator
- The MATLAB simulator is provided with the user guide for you to test your implementation and complete the assignment.
 - MazeColliders simulator: <https://github.com/cpu191/MazeColliders>
 - MazeColliders user guide: https://github.com/cpu191/MazeColliders/blob/master/MazeColliders_User_Guide.pdf
- **YOU ARE NOT PERMITTED TO USE THE ARDUINO LIBRARIES AND FUNCTIONS (Refer to ban list on UTS online);** however, you are permitted to use the LiquidCrystal.h library for the LCD, Serial.h for serial communication, and progmem library (including F()) for better memory handling. You are permitted to use the C standard libraries and AVR i/o and interrupt libraries.
- You may also use the C++ String variable types for this assignment.
- You may also use these variable conversion function part of the C++ variable and Arduino function:
 - <https://www.arduino.cc/reference/en/language/variables/data-types/string/functions/tofloat/>
 - <https://www.arduino.cc/reference/en/language/variables/data-types/stringobject/>
 - <https://www.arduino.cc/reference/en/language/variables/data-types/string/functions/toint/>

- A well commented complete PDF of your source code should be submitted in TurnItIn by the due date. And the same code needs to be uploaded as buildable “.ino” file separately for testing the functionalities.
- In the PDF document you must include the specific serial monitor setting you used. By default, we will be marking with the baud rate at 9600 and line ending as newline.

Marking Scheme

	Item	Mark Allocation	Marks Awarded
LCD	<ul style="list-style-type: none"> • Correctly displays the student ID in the first line 	1	
	<ul style="list-style-type: none"> • LCD displays correct information on second line allows user to choose specific modes of operation and the mode name clearly should display during operation. 	3	
Control	<p>The simulation robot should accurately move inside the map based on the user button inputs and it has to be easy to control.</p> <ul style="list-style-type: none"> - LEFT: robot rotating anticlockwise - RIGHT: robot rotating clockwise - UP: robot driving forward - DOWN: robot driving backward - Functioning user interface 	5	
Sweep	<p>The robot should scan (CW) and find the closest object</p> <ul style="list-style-type: none"> - UP: the sensor rotates 360° clockwise (CW) while taking measurements - Stationary robot - Functioning user interface 	3	
	<p>The robot turns (CCW) to face the identified object (shortest distance)</p>	4	
Wall follow	<p>Robot should follow the closest wall without collision</p> <ul style="list-style-type: none"> - robot finds the closest wall - Robot moves towards the wall (close to 2m) - Functioning user interface 	5	
	<p>Robot should maintain the set distance while continuously navigating until the UP button is pressed</p>	4	
System knowledge	<p>Questions during the demonstration</p>	5	
Total (30 Marks)		30	

**** If any student could not complete the whole task, they have the option of demonstrating parts those they have completed with reduced marks. As it is a complex mapping, the marks are decided by the discretion of the markers.**

Rules for Submission

Due Date:

3rd May 2021 at 6pm for the Mondays 6pm- 7:30pm class

3rd May 2021 at 7:30pm for the Mondays 7:30pm- 9:00pm class

5th May 2021 at 12pm for the Wednesday 12pm- 1:30pm class

Code

The code must be submitted as a **PDF Document** and in a **single .ino file**. Students must submit **all their code** into Canvas (Two submission section for PDF and .ino code). The code must be in text form and must be able to be built in a single file. Your code must be submitted to canvas at the due time (as given in the above ‘Due Date’ section) and before your demonstration (we will start marking after the due time based on your submitted code).

Demonstration

Students doing the subject on campus: We will mark based on in class, face to face demonstration for all the students. You will need to set up your system and get it ready for marking at the beginning of the class and your Canvas submission ready for downloading. We will be randomly approaching the students and mark the work. During marking, you will be asked to download your code from Canvas and upload it to your set up. You should demonstrate during your allocated class time.

Students doing the subject online: This option is only open for the students who are overseas, in another state or medically recommended not to be on campus. You will need to set up your system and get it ready for marking at the beginning of the class. You will need to have a camera set up to demonstrate various aspects and you will be asked to share your computer screen. We will be randomly approaching the students and mark the work. If the video feed is of poor quality, we will be marking based on your submitted code. You should demonstrate during Monday 6pm class time.

Students found using other student’s code will receive no marks for this assignment and have disciplinary actions taken against them by the university.

Support and Assistance

Support and assistance for this assignment will be available by posting questions on the “Discussion” forum on Canvas. This forum is monitored electronically and as such will have the same response time as a direct email. Please use the forum so that other students may benefit from the answers given.

Face to face support is available during the lecture and/or tutorial timeslot. Please email to make an appointment.

Students with difficulty meeting assessment requirements

Students who experience **significant** difficulty, or anticipate that they will experience significant difficulty, in meeting assessment requirements must submit an “Application for Special Consideration form” (available at <http://www.sau.uts.edu.au/assessment/consideration/online.html>) to the Registrar **before** the due date of the assessment item. Significant difficulty means

- i. Serious illness or psychological condition.
- ii. Loss or bereavement
- iii. Hardship/trauma

Note also that students may apply for special consideration because of illness or other circumstances (**not work related**) beyond their control. The “Application for Special Consideration form” has a section that must be filled in by a doctor, counsellor or other relevant professional authority. A medical certificate alone is not adequate and will not be accepted. Note that it is up to the students to provide adequate information about their circumstances. University staff will not chase additional information and the Subject Coordinator has the right to reject applications that lack sufficient information.

It is the student’s responsibility to contact the Subject Coordinator to find out what action has been taken and to obtain details of any additional assessment required or learning and assessment special arrangements.

For further details please refer to section 4.6 of the “Coursework Assessment Policy and Procedures Manual”.