CG2271: Real-Time Operating Systems

Mini Project

LuminNUS Team Number: _______________ HP Number: ______________________

The aim of the mini-project is to add some excitement to this module and allow you to develop an RTOS-based system beyond the structured labs. The project is expected to be completed at your own time, beyond the lab slots. You are free to make use of the Makers Lab to develop your project.

The aim of the project is to design a RTOS-based robotic car that will be controlled through an Android App. The robotic car must be able to fulfil the following features:

1. Establish a BT connection with the Android App
2. Receive commands from the Android App and execute the correct response
3. Move the car in multiple directions.
4. Control the various LED's according to the car's status
5. Play different sounds/tunes according to the cars status.

Grouping: The project can be completed in groups of 2 / 3. You do not need to be in the same group as your structured lab slot.

Component List per Group:

Item	Qty
DRV8833 Dual Motor Driver Carrier	2
Robot Smart Car Chassis Kit	1
BT06 Module	1
Red LED 5mm	12
Green LED 5mm	12
Prototyping Breadboards	3
Jumper Wire Bundle (40 pcs)	1
Battery Holder AA x 6	2
Piezo Buzzer	1
9V Battery Connector	3
220 ohms Resistors	24
Duracell Batteries (Pack of 12)	1
GP 9V Batteries	1
7805 Regulator	1
Push Button Switch	1
2-Way Connector Strip	1
FRDM KL25Z Board	1

Assembly Instructions:

1. The assembly of the chassis is given within the package.
2. The H-bridge motor drivers are similar to what you have used before in EPP2. Their interface details can be found here: https://www.pololu.com/product/2130
3. The interface of the BT module and the Android App Interface will be elaborated through a video tutorial.

Requirements Checklist:

A. BT Connectivity

Requirement	Level of Achievement
1. Develop a User Interface Button to establish BT connectivity with the Robot	
2. Robot must respond with TWO LED Flashes at the Front (Green LED's) to indicate that the connection has been established.	
3. Robot must play any unique tone sequence to indicate that connection has been established.	

B. Motor Control

Requirement	Level of Achievement
1. The robot must be able to move in all FOUR directions, Forward, Left, Right and Back.	
2. The robot must be able to perform curved turns while moving.	
3. The robot must stop all movement if no command is being sent.	

C. LED Control

Requirement	Level of Achievement
1. The front 8 Green LED's must be in a Running Mode (1 LED at a time) whenever the robot is moving (in any direction).	
2. The front 8 Green LED's must all be lighted up continuously whenever the robot is stationery.	
3. The rear 8 Red LED's must be flashing continuously at a rate of 500ms ON, 500ms OFF, while the robot is moving (in any direction).	
4. The rear 8 Red LED's must be flashing continuously at a rate of 250ms ON, 250ms OFF, while the robot is stationery.	

D. Audio Control

Requirement	Level of Achievement
1. The robot must continuously play a Song tune from the start of the challenge run till the end.* There should not be any break in the song even if the robot is not moving.	
2. When the robot completes the challenge run, the robot must play a unique tone to end the timing.	

*You are free to select any Song Tune. For this test, you must play the actual audio clip of the song and demonstrate that you are able to replicate a similar tune using the buzzer.

RTOS Architecture Minimum Requirements:

The architecture should have a minimum of 4 tasks and 1 Interrupt

tBrain: Decode the data from the Serial Port and perform the necessary action

tMotorControl: Control the Action of the Motors

tLED: Control the LED's

tAudio: Provide Audio Output

Serial_ISR: The Serial Data coming from the BT06 device. The serial data **MUST** be captured through the use of Interrupts.

This is a general guideline. You can have more tasks if you wish but not less.

You are free to decide the way in which the tasks will communicate and synchronize with each other. You must ensure that shared data is protected using appropriate RTOS constructs.

RTOS Architecture Report + Code Submission:

Each team is to submit a **SINGLE-PAGE Double-Sided** report explaining the RTOS architecture and the usage of Global Variables. Font Size should be 10 or more. Any submission with more pages will get **ZERO** marks.

Please zip up only the files with your implementation. You do not need to submit the RTOS library files.

Please upload your report to LumiNUS <Project Report> Folder with the format 'TEAM-XX.pdf'

Please upload your code to LumiNUS <Project Code> Folder with the format 'TEAM-XX.zip'

LumiNUS Video Submission:

Each team is to submit a 2-3 min video about their entire journey doing this project. Videos longer than 3 minutes will not be graded and will be given **ZERO** marks. The marks distribution for the video is as shown below.

Creativity – 8 marks

Technical Overview – 8 marks

Total = 16 marks

Please upload to LumiNUS <Project Video> Folder with the format 'TEAM-XX.mp4'. Other video formats like .mov are also acceptable.

Grading Criteria:

Item	Marks
Fulfilment of Requirements (12 x 2)	24
Usage of Global Variables	10
RTOS Architecture Report	10
Leaderboard Ranking	20
Video Submission	16

Total Marks for the Project = 80

Contribution to the Final Grade = 30%

CHALLENGE RUN

The challenge run will require you to control your robot and navigate it through a simple maze while fulfilling all the requirements in the checklist above. The actual challenge run maze is still being finalized, so stay tuned for an announcement on that.

IMPORTANT POINTS TO NOTE:

- The Leaderboard Ranking will be based on how fast your robot is able to traverse the maze.
- There will be Penalty timings imposed if certain events happen, e.g. hitting the wall/block.
- Your final Leaderboard rank will only be known after all the groups have completed the challenge.
- Each group is given **ONLY TWO ATTEMPTS** at the challenge run. The second attempt must be taken **immediately** after the first attempt. You will not be given any additional time in-between attempts.
- The **BEST** timing out of the 2 attempts will be taken.

Attempt	Timing	Hits	Final Timing
1			
2			