

Real Time Computing Assignments 2015/16

1. LEDs (1 to 2 sessions)

- a) Write a program to flash LED number 16 at a rate of around 2 Hz (exact frequency not important).
- b) Modify your program so that LED 0 is on for about $\frac{1}{2}$ a second, then LED 1, then LED2 etc up to LED31, then repeat the cycle for ever.
- c) Produce a program which allows the user to set any desired pattern on the LEDs by using the navigation switches. The kind of pattern achievable and how to achieve it is up to you.

This is a formative exercise. The assignment must be approved by a demonstrator.

2. Motor Control (4 sessions)

In the completed program, the following **basic requirements** shall be met:

- A user will be able to set the speed of the motor via a suitable user interface.
- The speed will either be chosen from a number of fixed speeds (at least 4) or be continuously variable.
- Either way, the target user selected speed will be displayed on the screen. The speed displayed will be in units of revolutions per second.
- The actual motor speed will be monitored and also displayed on the screen.

In an **advanced** implementation, the speed of the motor will be controlled using feedback (either a P or a PI controller) to keep it as constant as possible.

Since there are several tasks to achieve here, it is suggested that a staged approach to development is followed, although you do not have to follow this (only the final submitted code is assessed):

- a) Write a program that will cause the motor to run at a particular speed using PWM. Some experimentation will be necessary to find the right values to use in your program to achieve particular speeds.
- b) Design and implement a user interface that allows the speed to be set by the user, with display of the speed they have selected on the LCD screen.
- c) Add code to monitor the speed of the motor and display it on the screen alongside the speed the user has selected.
- d) (advanced, if time) Add feedback to control the speed of the motor and keep it as constant as possible (some experimentation will be necessary).

This is a summative exercise which will carry half the marks for the Real Time Computing element of the module. The final program code you submit will have reached one of the steps a) to d) above. You will be required to demonstrate your code during the lab session of week 7, at which time you will be assessed based on the observed performance of your code.

3. Sound Recorder

In the completed program, the following **basic requirements** shall be met:

- The user will be able to record sound from the microphone provided and later play that sound back through the headphones provided.
- A user interface will make it easy to control the operation of the recorder.
- The system should be capable of recording and playing back at least 5 seconds of sound at reasonable quality.

An **advanced** implementation would provide additional controls to vary playback speed and direction and allow multiple separate recordings to be made and played back.

This is a summative exercise which will carry half the marks for the Real Time Computing element of the module. You will be assessed on the performance of your system, the features which you implement and the extent to which you follow the required coding style. You are not required to demonstrate your code for this assignment.

Submission

Assignment 1 (formative) must be approved by a demonstrator before you proceed to the next assignment.

Assignment 2 source code (C code) and makefile must be submitted to **Duo by 2pm on 24th November**. The performance of your program will be assessed during your lab session in week 7.

Assignment 3 source code and makefile must be submitted to **Duo by 2pm 15th December**. There will be no lab session in week 10.

Assessment

For each summative assessment you must submit the source code for a program that compiles and runs and meets at least some of the requirements detailed in the assignment description. The assignments are intended to be very challenging and it is quite probable that you will not be able to implement them completely in the given time. Marks awarded will partly depend on what proportion of the complete assignment you are able to achieve, and on how well the function performs.

For assignment 2 you will be assessed purely on how well your program meets the requirements and their performance.

For assignment 3 you will be assessed both on how well your program meets the requirements and their performance, and on the quality of your code. You will be given guidelines on the expected coding style.

Plagiarism

The work you submit must be your own. The University takes plagiarism very seriously and ALL submitted code is subjected to a sophisticated plagiarism check.