EXAMINATION PAPER NO: 2302T

SHEFFIELD HALLAM UNIVERSITY

SCHOOL OF ENGINEERING

Full Course Title: TARC

Unit Title: Computer Engineering

Date: Tuesday 4 September 2001

Time allowed: 2 Hours
From: 1400 to 1610
(Including 10 minutes reading time)

INSTRUCTIONS TO CANDIDATES:

1. **The normal University examination regulations apply (see script answer book)**
2. **Do not start writing until instructed to do so by the Invigilator.**
3. **Answer THREE questions.**
4. **All questions carry equal marks.**

	PLEASE TICK	PLEASE SPECIFY TYPE/NUMBER REQUIRED
TABLES ATTACHED		
DIAGRAMS ATTACHED		
FORMULAE BOOKS REQUIRED		
16 - PAGE BOOKLETS	✓	
8 - PAGE BOOKLETS		
SUPPLEMENTARY BOOKS	✓	
GRAPH PAPER REQUIRED		

UNIT TUTOR(S): P Davies/A Goude

1. **(a)** Given that concurrent program execution is possible using a single CPU in conjunction with a real time clock and a host operating system to effect process swapping, explain how both response time and CPU utilisation are affected by the real time clock period 'T' when the operating system adopts a 'polled' approach to interacting with external IO. Derive equations for both average response time and CPU utilisation.

[8 Marks]

By choosing a suitable value for 'T', explain how you would: -

1. Optimise the system's response time to polling external IO.
2. Optimise the system utilisation for dealing with numerically intensive programs.
3. Optimise the system for dealing with a mixture of both numerically intensive and interactive programs.

[3 Marks]

(b) Explain, with the aid of suitable graphs illustrating process scheduling, why an 'interrupt' driven approach to Input/Output can, in conjunction with a longer than normal real time clock period 'T', give both better 'effective utilisation' and better response time figures than a 'polled' approach.

[12 Marks]

(c) You have been asked by your employer to design a system to generate a simple analogue waveform in the form of a ramp as shown in Fig Q1 below. The intention is to use a computer in conjunction with a digital to analogue converter (DAC) and a suitable program to generate the digital values that can be sent to the DAC. However, the computer system you have been given to carry out this task is one that is capable of multi-tasking with the aid of a Real-Time Clock (RTC). Explain with the aid of a diagram the shape of the typical resultant waveform that you would expect to see in reality when only your program is run on the system.

[5 Marks]

If a second, numerically intensive process were also to run on your computer at the same time, and with the same priority as your waveform generating process, how would this affect the shape of the waveform?

[5 Marks]

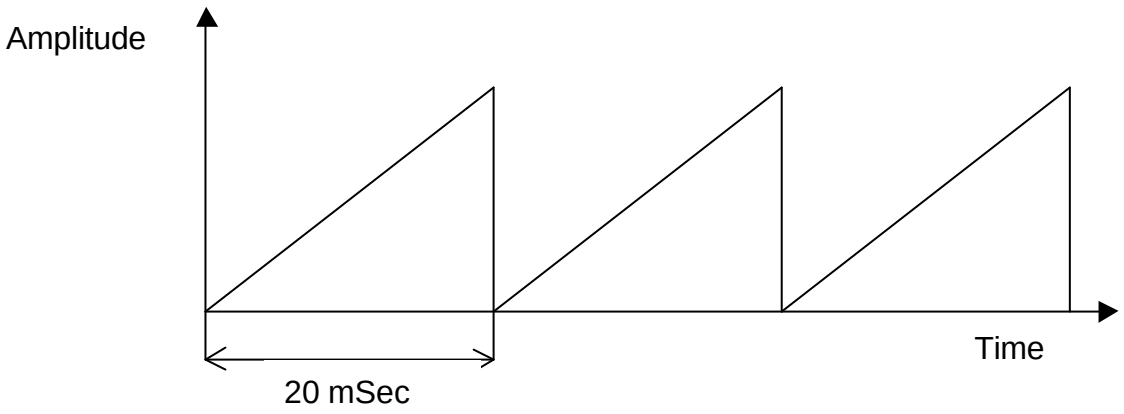

Fig Q1

2. (a) One of the most important roles played by the Memory Management Unit (MMU) in a concurrent system is that of process protection, i.e. preventing a process from accessing resources that do not belong to it, such as the operating system memory, reserved hardware and of course other processes that happen to be running alongside it in memory.

(i) Describe the interface between the MMU of the Motorola 68000 CPU and system main memory and explain the role and purpose of each of the interface signals.

[8 Marks]

(ii) Explain the purpose of the 'supervisor' and 'user' modes of operation within the Motorola 68000 Microprocessor and explain how each mode affects the operation of the MMU. Under what circumstances does the CPU switch from one mode to the other?

[8 Marks]

(iii) Explain how process protection is carried out by the MMU, i.e. how the MMU validates a process's access to main memory in conjunction with the CPU's mode of operation, and what happens when an illegal access is detected. What reprogramming of the MMU takes place with each time-slice.

[8 Marks]

2. (b) A further problem faced by any concurrent operating system is **how** and **where** to place programs in memory for execution. In order to maximise CPU utilisation, it is desirable to have **all** active processes resident in memory at the same time. Given that some of them may have been written in such a way that they make assumptions about where in memory they will be loaded and run, and that it may not be possible to load a program at that address (because that section of memory is in use by another process) explain in detail how the operating system copes with and resolves this problem using an MMU.

[5 Marks]

Explain what is meant by Position Independent Code and explain how the use of this can overcome the problem identified above.

[4 Marks]

3. **(a)** Fig Q3 below shows two processes, P1 and P2 that independently access an 8 channel Analogue to Digital Converter (ADC). To perform a conversion, a process first has to write a channel number into the ADC's 'Channel Select Register', and then simply has to perform a write to the ADC's control register to start the conversion. The process would then 'poll' the Status register while the device was busy waiting for the ADC to indicate that the conversion has been completed. Identify, using suitable pseudo-code to back up your explanation, what possible problems could occur if both processes accessed the device at or near the same point in time.

[15 Marks]

(b) Explain in detail why you cannot simply use a 'flag' (a single bit variable with the values 0 or 1) to protect the ADC in Fig Q3 from access by both processes at the same time. Your answer should be supported by pseudo-code listings.

[10 Marks]

(c) Explain how you would overcome the problems identified in 3a) above with a semaphore. Your answer should include full pseudo-code listings for each process to illustrate how it would use the semaphore and an explanation of **why** it works. Also, explain the significance of the numerical value assigned to the semaphore and outline what would happen if any other value were used.

[8 Marks]

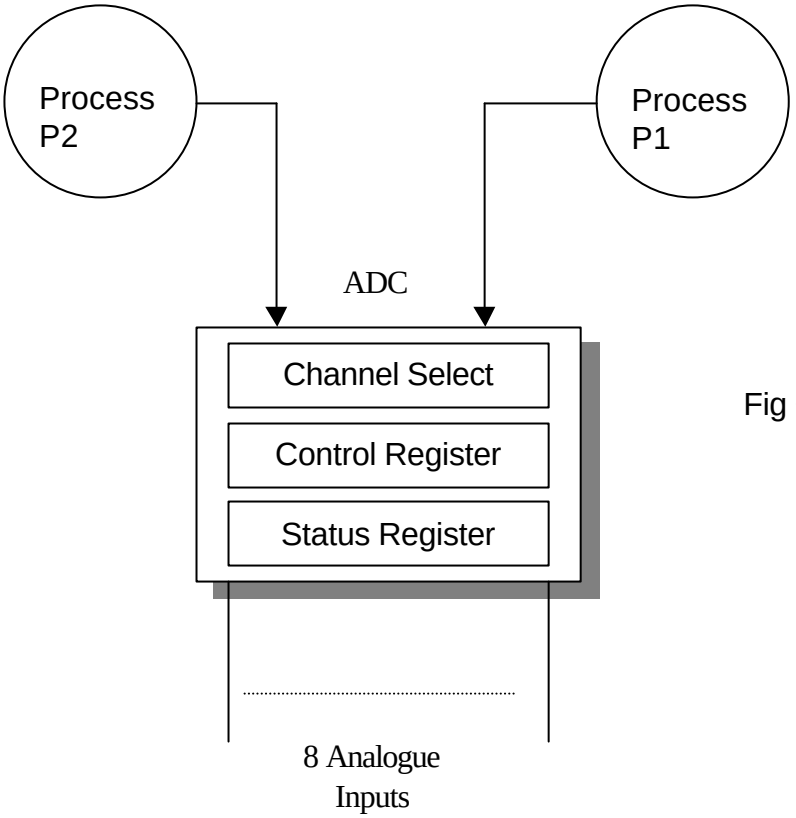

Fig Q3

4. (a) (i) Real time systems are often classified on the basis of their response time and upon how they are driven by external stimuli. Explain what is meant by the following terms

- Hard and soft real time systems
- Event and time driven systems

[4 Marks]

(ii) What is meant by a "deterministic response time", and why is this important in a real time system. Discuss what factors, both software and hardware, affect whether a system is deterministic or not. Suggest further why Windows NT may not be the most appropriate operating system to use in a 'hard' real-time system.

[8 Marks]

(iii) What are meant by the following terms in relation to process scheduling

- Pre-emptive scheduling
- Volatile Environment
- Process Descriptor

[6 Marks]

4. (b) (i) Fig Q4 Below shows four processes P1 - P4 cooperating in a multi-tasking system. Process P1 reads data from two pipelines, Pipe1 and Pipe2, as and when 1 byte of data has been written into it by processes P2 and P3. P1 also reads a single byte of data from the data pool DP1. Process P1 processes the data it reads and displays the results on the VDU.

Processes P2, P3 and P4 each read a single character from their respective VDUs. The semaphores PS1 and CS1 are there to handle the producer/consumer arrangement that exists between processes P4 and P1. Given this arrangement, design the pseudocode only, for process P1, making full use of suitable software primitives where necessary, that would allow it to deal with the arrival of data from either of the pipelines or the datapool as and when such data arrive. Assume that P1 adopts a **time driven**, (i.e. polling**)** approach to determining if data is available to be read. Remember P1 should not get suspended waiting for data that might not arrive.

(Note: You may assume that all pipelines, data pools, semaphores etc. have been created and that where relevant all processes have linked/opened paths to them prior to your code, i.e. there is no need to provide this level of detail in your pseudo-code).

[10 Marks]

QUESTION 4 CONTINUES OVERLEAF

QUESTION 4 CONTINUED

4. (b) (ii) Because polling is wasteful of CPU resources, outline briefly (there is no need to go into the detail of the pseudocode) how the utilisation of the CPU could be improved by using an event driven approach, in particular how you would ensure that 'P1' is only active when data is available to be read from Pipe1, Pipe2 or DP1

[5 Marks]

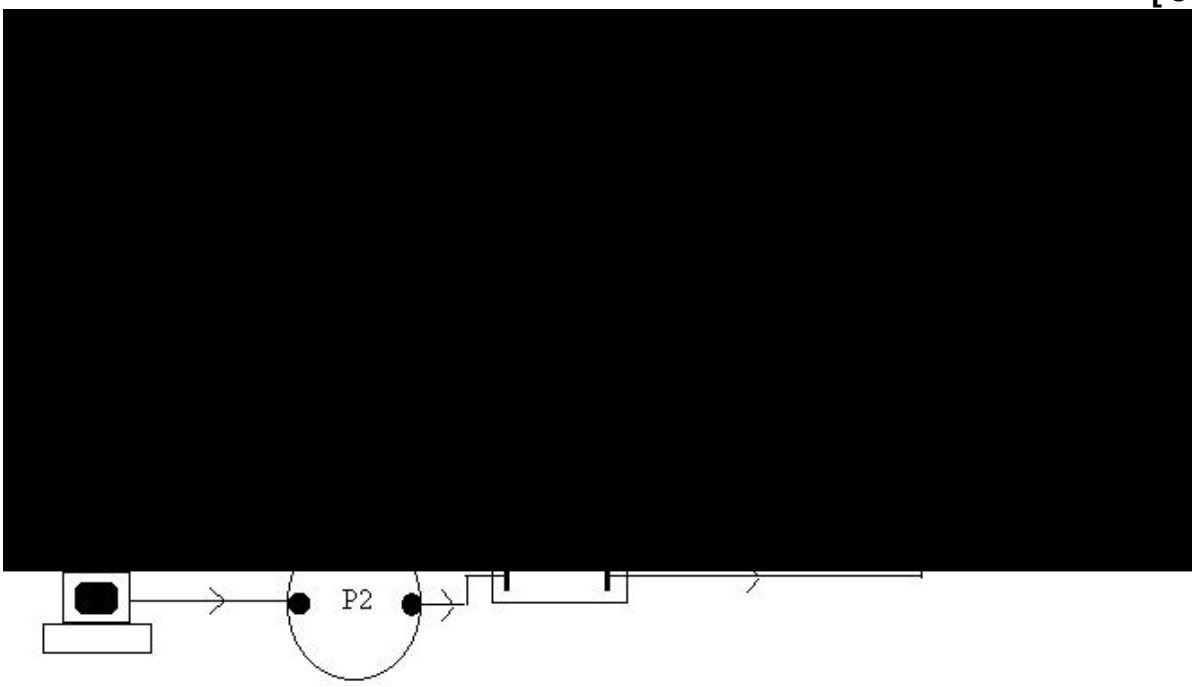

Fig Q4