UNIVERSITY OF ALBERTA

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

ECE 315 – Computer Interfacing

Final Assessment

Instructor: Dr. Bruce F. Cockburn
Due date & time: Friday, April 17, 2020, 11:00 am sharp (upload to eClass using the link)
As a back-up method only, e-mail your booklet to: cockburn@ualberta.ca

Aids permitted: The lecture slides and any reputable references can be freely consulted.
Internet references, including manufacturer documentation, can be consulted.
Past model solutions from eClass can also be freely consulted.
References (not slides & solutions), must be clearly referenced using a note that gives the author and/or organization, the full document title, and other available publication details (such as publisher, date, document number, URL).
Electronic calculators of any kind can be used.

Instructions:
1. The problems must be completed by each student with no help from others.
2. Enter your answers directly into the MS Word file. The Word file can also be converted into another file format, but must then be uploaded in .pdf.
3. The .pdf version can be printed and completed by hand, but the finished document must then be scanned into .pdf and uploaded into eClass.
4. Enter your full name and I.D. number on this page.
5. This final assessment consists of seven questions, some with multiple parts.
6. Neatly type your answers into the space provided (adjustable within a page).
7. The number of pages must be kept to 13, using the original page structure.
8. The answers must be in Times font, at least 11 point, single-line spaced.
9. Take into account the marks per question when budgeting your time.

Student name: _____, _____
Family name **Given name**

Student I.D.: _____

Question	Worth	Mark	Subject
1.	20		Basic Concepts
2.	15		Preemptive Multitasking Architecture
3.	10		Serial Peripheral Interface
4.	15		TCP/IP-based Computer Communications
5.	10		FILO Buffers and FIFO Stacks
6.	15		Stepper Motor Control
7.	15		Direct Memory Access
Total	**100**		**—**

Question #1 (Basic Concepts)

(a) Briefly explain what is meant by "buffer overflow". How could a buffer be designed so that buffer overflow is avoided? [5 marks]

(b) An embedded system that polls a variety of sensors at different harmonically related frequencies could be implemented either using one bare-metal thread triggered by a single hardware timer, or as a multitasking system that supports state-driven code. What major considerations should a designer keep in mind when selecting one of these two software architectures over the other? [5 marks]

Question #1 (Basic Concepts, cont'd)

(c) Cross-over cables have arisen in a number of communication standards, including RS-232C and twisted-pair Ethernet. What is the problem that is solved by using a cross-over cable? How have more recent versions of Ethernet avoided the need for crossover cables? [5 marks]

(d) The Fast Ethernet Controller (FEC) block of the MCF5234 microcontroller uses what at first might appear to be a rather complex system of buffers for handling the data that is transmitted and received. What would be the major advantages of using such a system, where the transmit and receive directions have independent buffer systems and where the transmit buffers are variable in size, but the receive buffers all have the same size? [5 marks]

Question #2 (Preemptive Multitasking Architecture)

(a) In your own words briefly describe the priority inversion problem. Also briefly describe and justify the common strategy that is used in MicroC/OS to avoid the priority inversion problem. [7 marks]

Question #2 (Preemptive Multitasking Architecture, cont'd)

(b) What are at least three different common mechanisms that can force a running task in a preemptive multitasking architecture to give up the CPU? [6 marks] What prevents a running task in a preemptive multitasking architecture from consuming more than some maximum percentage of the CPU's time? [2 marks]

Question #3 (Serial Peripheral Interface)

The Serial Peripheral Interface (SPI) standard uses a circular shift register architecture to interface a controlling device (e.g., an SPI master device) to one or more peripheral devices (e.g., SPI slave devices). Briefly explain how the shift register is used to both write information from the master to multiple slaves in the daisy-chained configuration, and to read information from the same slaves to the master. [6 marks] Comment on the likely advantages and disadvantages of the daisy-chained configuration for connecting multiple slaves devices to one master, compared with the alternative of having separately enabled/addressed slave devices. [4 marks]

Question #4 (TCP/IP-based Computer Communications)

(a) An 11-state finite state machine is implemented (typically in software) to record the status of the "TCP entity" at each end of a TCP connection. What are the likely advantages of recording the present status of the TCP entity using the relatively small number of states in a standardized state machine instead of using less strictly structured software algorithms? [3 marks] What kinds of events cause the TCP entity to advance from state to state? [2 marks] What kinds of outputs can be produced as a result of these state transitions? [2 marks]

Question #4 (TCP/IP-based Computer Communications, cont'd)

(b) Briefly explain what is mean by the client-server model and the socket for communication over TCP/IP. What are the benefits of the client-server model and the socket given that using sockets restricts the kinds of interactions that a user application can have with TCP. [8 marks]

Question #5 (FILO Buffers and FIFO Stacks)

Both first-in first-out (FIFO) buffers and first-in last-out (FILO) stacks require a mechanism that allocates memory space when new data needs to be written, and that de-allocates memory space when data is read and then "removed" from further access. Briefly explain how this memory allocation and de-allocation is efficiently implemented in FIFO buffers and FILOs stacks. [10 marks]

Question #6 (Stepper Motor Control)

(a) What are the main reasons why a stepper motor is a good choice for an actuator that produces physical rotation under the command of the digital computer? [4 marks] What would be some of the major disadvantages of using a stepper motor? [3 marks] How could those disadvantages be controlled or at least minimized? [3 marks]

Question #6 (Stepper Motor Control, cont'd)

(b) A stepper motor control algorithm must take into account several constraints including the maximum acceleration and deceleration rates and the maximum slew rate. What would be the likely consequences if a control algorithm repeatedly violating these constraints? Be sure to explain your answers. [5 marks]

Question #7 (Direct Memory Access)

(a) Briefly describe what is meant by "Direct Memory Access (DMA)"? [4 marks] What are suitable situations for using DMA in an embedded system? [2 marks] In which situations would DMA not be advisable to use compared to operation without using DMA? [2 marks]

Question #7 (Direct Memory Access)

(b) It is necessary to provide a mechanism for sharing the system bus among the CPU(s) and DMAC(s). Briefly explain what is meant by cycle-stealing mode and the continuous/burst/exclusive mode for sharing a bus. [3 marks] What the main advantages and disadvantages of these two modes with respect to each other? [4 marks]