UNIVERISTY OF VICTORIA

CENG 355 MICROPROCESSOR-BASED SYSTEMS

MIDTERM EXAMINATION

27 OCTOBER 2017

NAME:_____ STUDENT NO._____

INSTRUCTOR: D.N.RAKHMATOV DURATION: 80 MINUTES

TO BE ANSWERED IN THE BOOKLET.

STUDENTS MUST COUNT THE NUMBER OF PAGES IN THIS EXAMINATION PAPER, AND REPORT ANY DISCREPANCY IMMEDIATELY TO THE INVIGILATOR.

THIS EXAMINATION PAPER HAS **4** PAGES AND **3** QUESTIONS.

In taking this examination, you agree that all work recorded herein is your own. A student caught in the act of cheating will be given a grade of **F** on this examination.

Show your work.

Read the questions carefully. If something appears ambiguous, write down your assumptions.

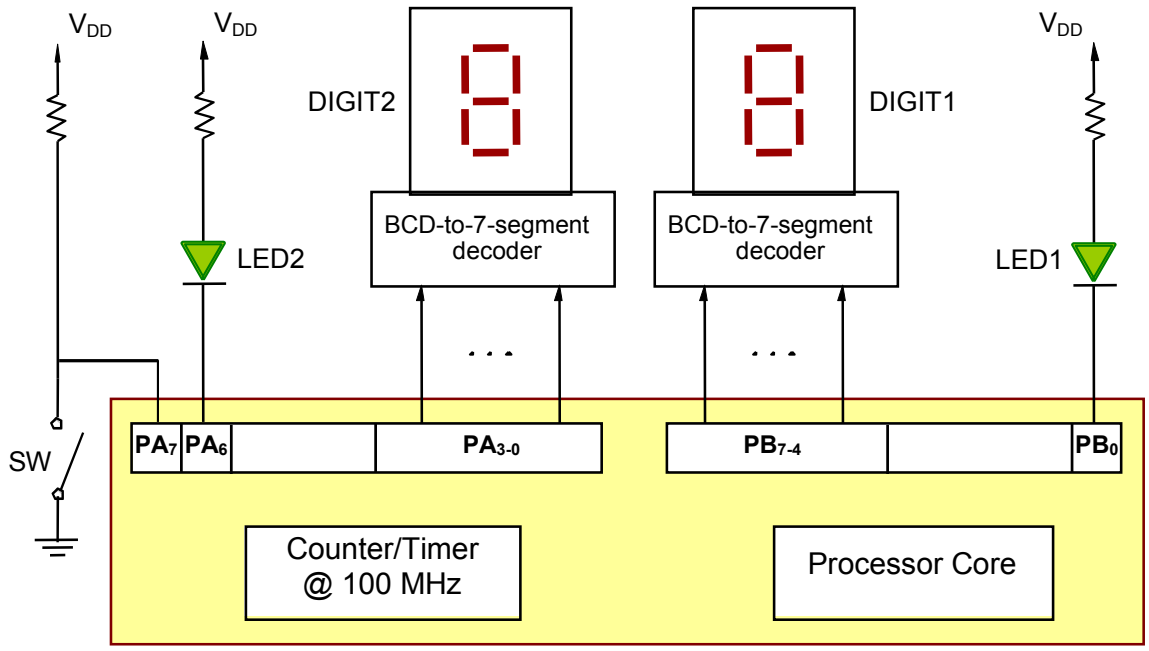
You are allowed to use books, notes, and/or any other printed materials during this examination.

GOOD LUCK!

1. *[15 points]* The textbook's microcontroller, used in a system below, is responsible for two tasks: (1) incrementing either **DIGIT1** (if **LED1** is on) or **DIGIT2** (if **LED2** is on) every second, and (2) alternating between **LED1** and **LED2** being on, whenever **SW** is hit (i.e., pressed and then released). Write the corresponding C program, assuming that the **first task** is an ISR, whose address is stored at location **0x20**, and the **second task** is the main program. Also, assume that bit **6** of the processor status register (i.e., **PSR[6]**) is the processor's interrupt-enable bit, and **Ports A** and **B** are always ready to be accessed by the processor. Initially, **LED1** is on, **LED2** is off, and both **DIGIT1** and **DIGIT2** show **0**.

• *Main Program*: Every time the **SW** key is hit, i.e., pressed and then released (**PA_7** must first become 0 and then 1 again), **LED1** and **LED2** must swap their states: if **LED1** is on and **LED2** is off, then **LED1** becomes off and **LED2** becomes on, and vice versa. (Note: **LED1** and **LED2** are never both on, or both off.)

• *ISR*: The 100-MHz Counter/Timer must be configured to generate interrupts every second, and its ISR must increment **DIGIT1** if **LED1** is on, or increment **DIGIT2** if **LED2** is on. (Note: incrementing **9** gives **0**.)



2. *[15 points]* Table below specifies a set of **4** independent pre-emptive tasks to be executed by a single processor. Show the task schedules based on:

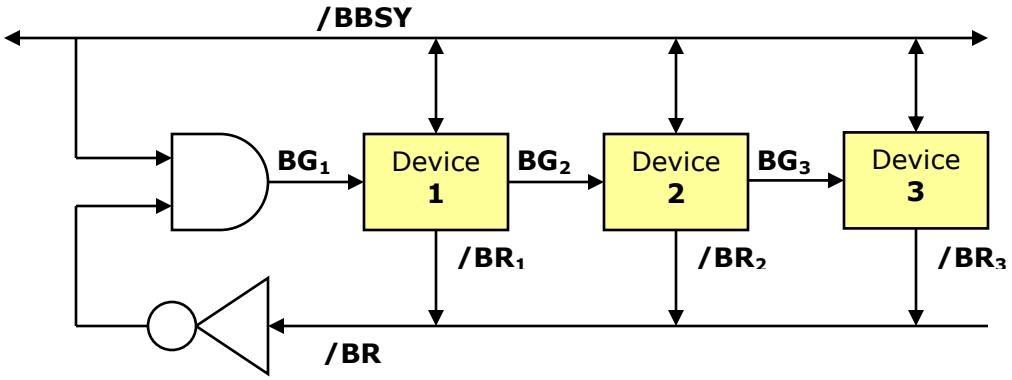
(a) *Rate Monotonic* (**RM**) priority assignment;

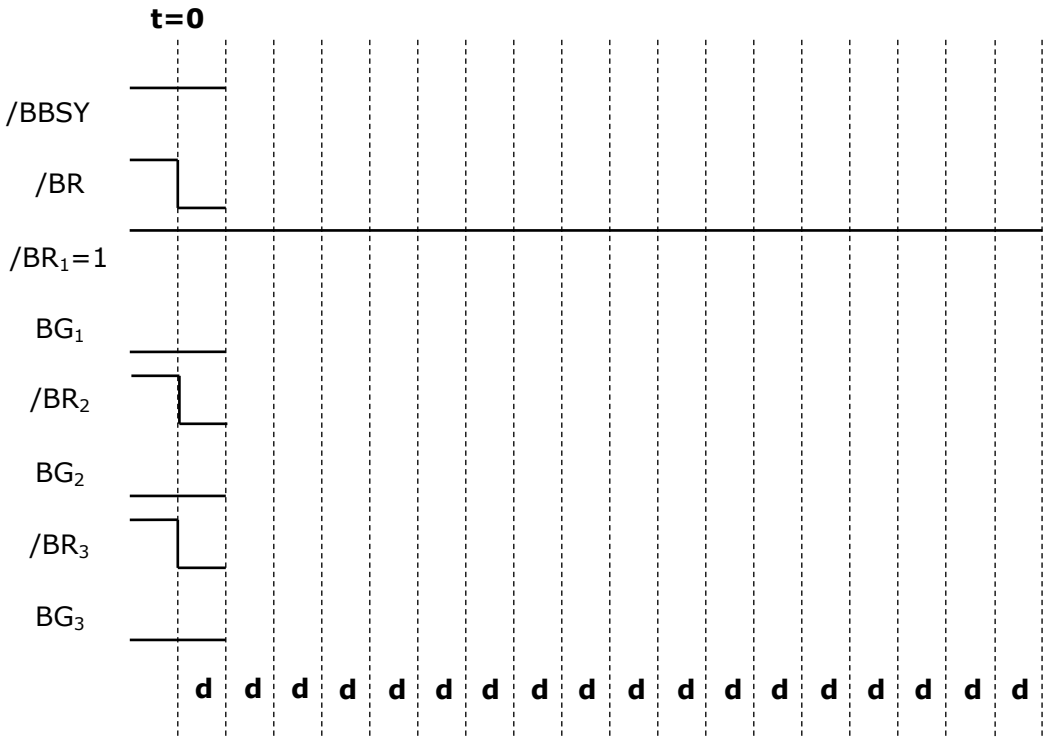
(b) *Earliest Deadline First* (**EDF**) priority assignment.

If needed, break any prioritization ties as you wish.

Task $\mathbf{T_i}$	Period $\mathbf{P_i}$	WCET $\mathbf{C_i}$	Deadline $\mathbf{D_i}$	Initial Delay ϕ_i
T1	30	5	20	0
T2	30	5	30	0
T3	45	15	25	0
T4	90	15	80	0

3. *[10 points]* Consider the daisy-chain arbitration scheme shown below. Assume that the input-to-output signal propagation delays are the same and equal to **d** for all three devices, the inverter, and the **AND** gate. Also, assume that device **x** is able to start using the bus (making **/BRx = 1** and **/BBSY = 0**) only when it receives a **0-1** transition on its bus-grant input **BGx** and detects that the bus is not currently busy (i.e., **/BBSY = 1**). Also, assume that device **x** lets the bus-grant propagate through only when it is neither requesting nor using the bus. Finally, assume that any of the three devices will need to use the granted bus for only **3d** time units. Complete the timing diagram shown on the next page, where Device 2 and Device 3 request the bus at the same time **t = 0**.



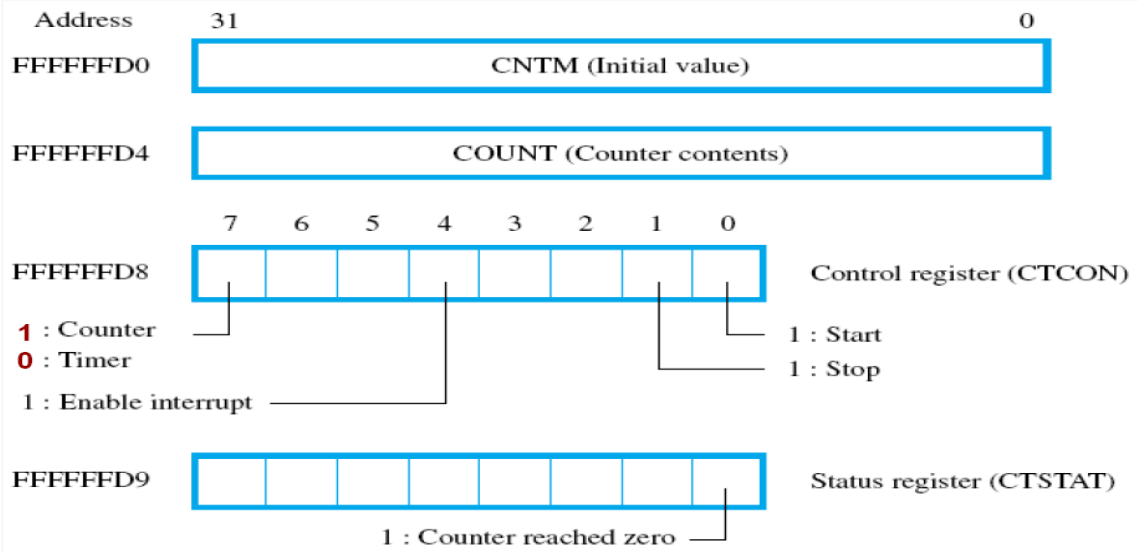


NOTE: If answering Question 3 here, please write down your name below and insert this page in your Booklet.

NAME: _____

END

Counter/Timer Registers



Parallel Port Registers