

EECS388 Embedded Systems Practice Questions

True/ False:

1. L&L bound can prove that a task set is not schedulable with RMS.
False.
2. The exact schedulability test calculates the total interference caused by higher priority tasks for a lower priority task.
True.
3. It is impossible to have priority inversion in a system with two tasks.
True
4. Priority ceiling protocol prevents certain deadlocks.
True

Short Answer:

1. Implement a subroutine that multiplies the value of memory stored in Label1 by the value stored in Label1.

```
                .ORIG x3000
                JSR  MUL
                HALT
MUL             // your code here
                // R3 = Mem[Label1] * Mem[Label2]
Label1         .BLKW 1
Label2         .BLKW 1
                .END
```

```
                LD      R1, Label1
                LD      R2, Lable2
LOOP           BRz     DONE
                ADD     R1, R1, R1
                ADD     R2, #-1
                JMP     LOOP
DONE           RET
```

Sine we don't use R1 and R2 in the main function, we don't need to save or restore them.

2. What are the data that the following bit stream is transmitting? Assume RS-232 serial protocol. Is there any error in the transmission?

1111111011010111111111111111000011000011111111

1111111 0 (start) 11010111 (first data reverse) 1 (p)
11 (stop) 11111111 0 (start) 00011000 (second data
reverse) 0 (p) 11 (stop) 111111

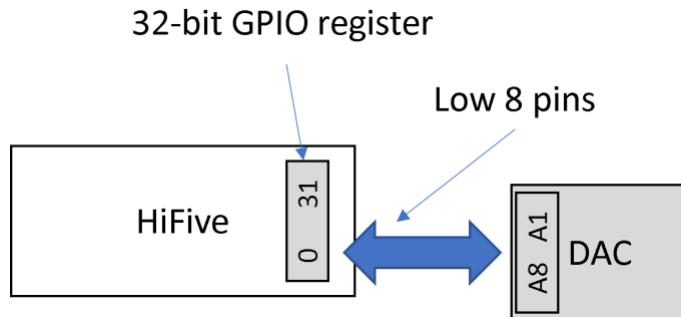
First data = reverse of (0b11010111) = 0b11101011

Parity is 1 but there are even number of "1"s in the data => error

Second data = reverse of (0b00011000) = 0b00011000

Parity is 0 and there are even number of "1"s in the data => no error

3. Complete the following code to set the output voltage to 10/256 volts. Assume that the DAC is a +10V output DAC.



```
void set_dac()
{
    uint32_t val = *(volatile uint32_t *) (GPIO_CTRL_ADDR +
    GPIO_OUTPUT_VAL);

    // YOUR CODE HERE

    *(volatile uint32_t *) (GPIO_CTRL_ADDR + GPIO_OUTPUT_VAL) = val;
}
```

$$V_o = 10 * (A_1/2 + A_2/4 + A_3/8 + A_4/16 + A_5/32 + A_6/64 + A_7/128 + A_8/256)$$
$$= 40/256$$

When you solve the equation you get $A_6 = 1$ and all other $A_s == 0$.

CODE:

```
val = val & 0xFFFFFFF0;
```

```
val = val | 0x00000004;
```

4. Assume that you have a processor that is connected over a network adapter to the Internet. The network adapter receives packets with the following inter-arrival time distribution:



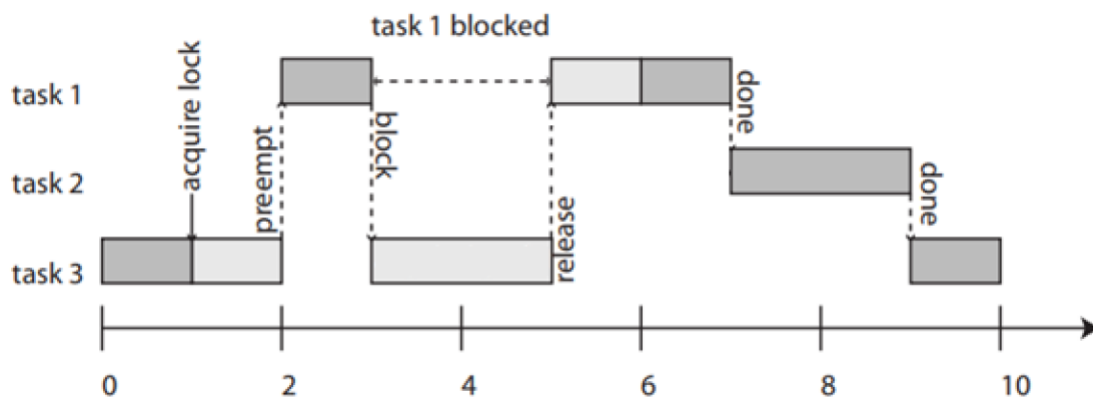
One of your EECS388 classmates developed a protocol called "adaptive interrupts" for notifying the microcontroller of packet arrivals at the network adapter:

By default, the network adapter is interrupt-driven, meaning that once a packet is received, it sends an interrupt to the processor immediately. After the first Interrupt is received, we switch to polling, meaning that the processor starts polling the network adapter for a short interval and then switch back to interrupts again.

Do you endorse your classmate's protocol? Why?

Yes! Polling work best when we have a burst of events, while interrupt saves CPU cycles when we have low activity. The adaptive interrupt protocol outlined above enjoys the best of both Polling and Interrupt.

5. What is the priority of task 3 at time 1, 4, and 6 with and without priority inheritance protocol? Assume that the initial priority of the tasks is: (highest priority) task#1 = 3 > task#2 = 2 > task#3 = 1 (lowestst priority)



time	1	4	6
Without priority inheritance	1	1	1
With priority inheritance	1	3	1

6. Consider the following real-time tasks and a single-core system.

Task	Compute Time (Seconds)	Period (seconds)
t1	3	10
t2	6	15
t3	10	35

L&L Bound

$$UB(1) = 1.0$$

$$UB(2) = 0.828$$

$$UB(3) = 0.779$$

$$UB(n) = 0.693$$

1.1. Which task has the highest priority?

Under RMS: t1

Under EDF the priority changes over time.

1.2. What is the total CPU utilization of the taskset?

0.9857

1.3. Is this taskset schedulable based on Liu & Layland bound?

We don't know!

1.4. Is this taskset schedulable under the rate monotonic scheduler? Use the exact analysis for your answer. The equation is given as follows where r denotes the response time (ready to completion) of the task. You do not need to upload your steps.

$$r_i^{k+1} = c_i + \sum_{j=1}^{i-1} \left\lceil \frac{r_i^k}{p_j} \right\rceil c_j, \quad \text{where } r_i^0 = \sum_{j=1}^i c_j$$

- (A) Yes it is schedulable because $r_3^3 == r_3^2$ and $r_3^3 < 35$
- (B) Yes it is schedulable because $r_3^4 == r_3^3$ and $r_3^4 < 35$
- (C) No it is not schedulable because $r_3^3 > 35$
- (D) No it is not schedulable because $r_3^2 > 35$

7. Draw a timeline (from time 0 to 40) of the taskset under the EDF scheduling on a piece of paper and answer the following questions.

7.1. Which task is running at time 9 and for how long?

- (A) t1 is running until time 13
- (B) t2 is running until time 10
- (C) t3 is running until time 10

7.2. Which task is running at time 20 and for how long?

- (A) t1 is running until time 23
- (B) t2 is running until time 21
- (C) t1 is running until time 21
- (D) Both (A) and (B) are possible

7.3. Select all the correct statements?

- (A) There is no missing deadline until time 40 using EDF scheduler
- (B) t3 only runs for 1 seconds until time 20
- (C) t1 is running at time 15
- (D) t3 is running at time 24
- (E) t3 is running at time 30

(D) None of the above

