Name:____ Email:__________________@ncsu.edu

ECE 461/561, Spring 2023: Test 2 Solutions (rev. 3)

This quiz is closed-computer, closed-book, closed-notes. You may use two 8.5" x 11" sheets of paper with anything you want written or printed on its two sides. Calculators are allowed, but not cellphones.
Unless otherwise stated, assume the code is built using MDK-ARM (AC6 compiler, armlink linker, all settings for maximum optimization for time) for the Kinetis KL25Z128 MCU used on the FRDM-KL25Z board and the core clock frequency is fixed at 48 MHz.

All questions are equally weighted.

Grading basics:

- Blank: 0 pts
- Something relevant written: 1 pt

Please read and sign this statement: I have not received assistance from anyone nor assisted others while taking this test. I will not use my cellphone on this test. I have also notified the test proctor of any violations of the above conditions.

Signature ____

#	Notes	Score
1		
2		
3		
4		
5		
6		
7		

8		
9		
10		
11	561 Only	
12		
13		
14		
15		

16		
17		
18		
19		
20	561 Only	
Total		

Possibly Useful Reference Information

Condition Flag	Meaning if 0	Meaning if 1
Z Zero	Result not zero 0x0000	Result was zero
V Overflow	Result did not overflow	Result overflowed
N Negative	Not negative, MS bit of result is 0	Negative, MS bit of result is 1
C Cary	No carry out or borrow in	Carry out or borrow in occurred

Mnemonic Extension	Meaning	Condition Flags		Mnemonic Extension	Meaning	Condition Flags
EQ	Equal	Z == 1		VC	No overflow	V == 0
NE	Not equal	Z == 0		HI	Unsigned higher	C == 1 and Z == 0
CS	Carry set	C == 1		LS	Unsigned lower or same	C == 0 or Z == 1
CC	Carry clear	C == 0		GE	Signed greater than or equal	N == V
MI	Minus, negative	N == 1		LT	Signed less than	N != V
PL	Plus, positive or zero	N == 0		GT	Signed greater than	Z == 0 and N == V
VS	Overflow	V == 1		LE	Signed less than or equal	Z == 1 or N != V

Power and Energy – Version A

Using Sleep Mode

Consider an MCU which runs at 40 MHz with a 3.0 V supply, drawing 4 mA. It needs to perform 80,000 cycles of computation every 20 ms. It is normally sleeping, drawing 100 uA. Every 20 ms it wakes up, performs the computations, and goes back to sleep. Assume it takes 800 us to wake up, and draws 5 mA while waking up.

Duty Cycles

5: Correct (matching key)

4: Almost correct, only one missing/wrong factor: E.g. period is not 20 ms, or 20 ms * 40 MHz = 800,000 cycles

3: Period missing, or multiple missing/wrong factors.

1. What is the fraction of time that the MCU is waking up?
   800 us/ 20 ms = 0.04

2. What is the fraction of time that the MCU is performing computations?
   (80,000/40MHz)/ 20 ms = 0.10

3. What is the fraction of time that the MCU is sleeping?
   (1 – 0.04 – 0.1) = 0.86
   Full credit if result is correct given previous answers: #3 = 1 - #1 - #2

Average Power Consumption

Almost correct penalty: -1 if missing time fraction (duty cycle) term

4. What is the average power consumption (in mW) due to the MCU waking up?
   5 mA * 3 V * 0.04 = 0.6 mW

5. What is the average power consumption (in mW) due to the MCU performing computations?
   4 mA * 3 V * 0.10 = 1.2 mW

6. What is the average power consumption (in mW) due to the MCU sleeping?
   0.1 mA * 3 V * 0.86 = 0.258 mW

7. What is the MCU's total average power consumption (in mW)?
   2.058 mW
   Full credit if result is correct given previous answers: #7 = #4 + #5 + #6

Power and Energy – Version B

Using Sleep Mode

Consider an MCU which runs at 40 MHz with a 3.0 V supply, drawing 4 mA. It needs to perform 8,000 cycles of computation every 20 ms. It is normally sleeping, drawing 200 uA. Every 20 ms it wakes up, performs the computations, and goes back to sleep. Assume it takes 800 us to wake up, and draws 5 mA while waking up.

Duty Cycles

5: Correct (matching key)

4: Almost correct, only one missing/wrong factor: E.g. period is not 20 ms, or 20 ms * 40 MHz = 800,000 cycles

3: Period missing, or multiple missing/wrong factors.

1. What is the fraction of time that the MCU is waking up?
   800 us/ 20 ms = 0.04

2. What is the fraction of time that the MCU is performing computations?
   (8,000/40MHz)/ 20 ms = 0.01

3. What is the fraction of time that the MCU is sleeping?
   (1 – 0.04 – 0.01) = 0.95
   Full credit if result is correct given previous answers: #3 = 1 - #1 - #2

Average Power Consumption

Almost correct penalty: -1 if missing time fraction (duty cycle) term

4. What is the average power consumption (in mW) due to the MCU waking up?
   5 mA * 3 V * 0.04 = 0.6 mW

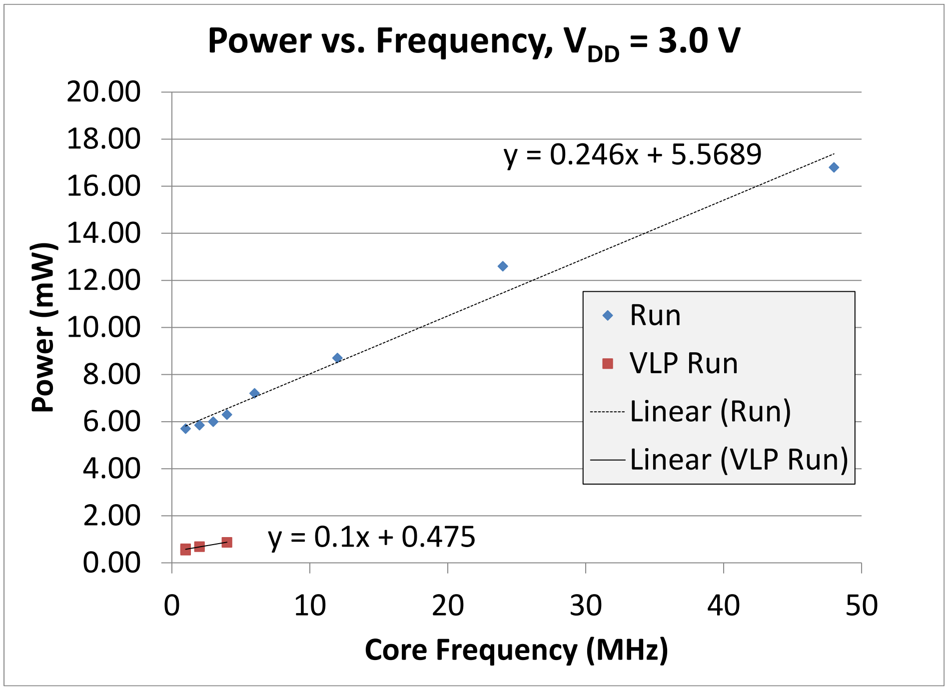
5. What is the average power consumption (in mW) due to the MCU performing computations?
   4 mA * 3 V * 0.01 = 0.12 mW

6. What is the average power consumption (in mW) due to the MCU sleeping?
   0.2 mA * 3 V * 0.95 = 0.57 mW

7. What is the MCU's total average power consumption (in mW)?
   1.29 mW
   Full credit if result is correct given previous answers: #7 = #4 + #5 + #6

Voltage and Frequency Scaling – Version A

Selected power vs. MCU core frequency characteristics of a KL25Z128 MCU are shown below.



8. Consider a system (different from the previous questions) using a KL25Z128 MCU operating at 28 MHz in Run mode at 1.9 V.
   a. What is the power consumption in (mW)?
   (0.246 * 28 MHz + 5.5689 mW)*(1.9 V / 3.0 V) = 7.889 mW
   1.5 pts 1 pt

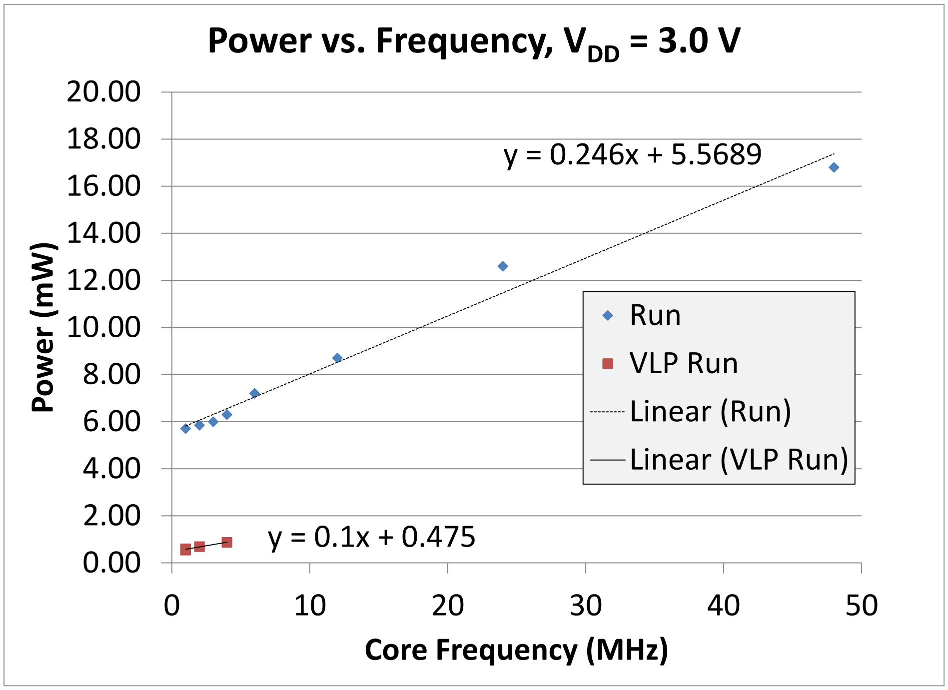
   b. How much energy (in pJ) is used per clock cycle?
   7.889 mW/28 MHz = 0.282 nJ = 282 pJ

9. Consider a system (different from the previous questions)using a KL25Z128 MCU operating at 3 MHz in VLP Run mode at 2.8 V.
   a. What is the power consumption in (mW)?
   (0.1 * 3 MHz + 0.475 mW)*(2.8 V / 3.0 V) = 0.723mW
   1.5 pts 1 pt

   b. How much energy (in pJ) is used per clock cycle?
   0.723 mW/3 MHz = 0.241 nJ = 241 pJ

Voltage and Frequency Scaling – Version B

Selected power vs. MCU core frequency characteristics of a KL25Z128 MCU are shown below.



8. Consider a system (different from the previous questions) using a KL25Z128 MCU operating at 18 MHz in Run mode at 2.9 V.
    a. What is the power consumption in (mW)?

    (0.246 * 18 MHz + 5.5689 mW)*(2.9 V / 3.0 V) = 9.664 mW

    1.5 pts 1 pt

    b. How much energy (in pJ) is used per clock cycle?

    9.664 mW/18 MHz = 0.537 nJ = 537 pJ

9. Consider a system (different from the previous questions)using a KL25Z128 MCU operating at 2 MHz in VLP Run mode at 2.2 V.
    a. What is the power consumption in (mW)?

    (0.1 * 2 MHz + 0.475 mW)*(2.2 V / 3.0 V) = 0.495 mW

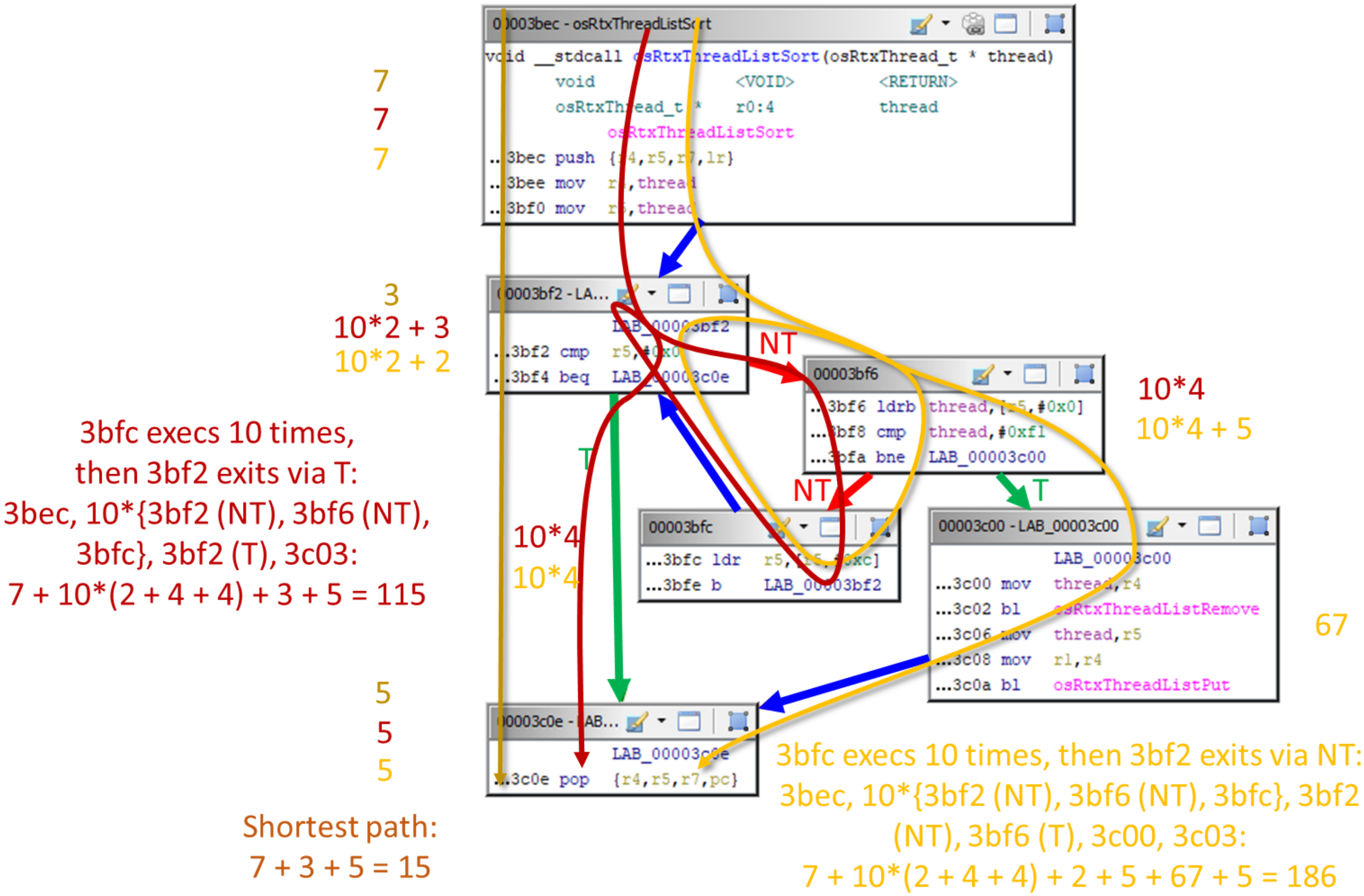
    1.5 pts 1 pt

    b. How much energy (in pJ) is used per clock cycle?

    0.495 mW/2 MHz = 0.248 nJ = 248 pJ

Responsiveness

Static Timing Analysis



Consider the CFG above. Assume the following:

- Instruction execution times (in CPU cycles):
  - Push/Pop: # of registers pushed/popped + 1
  - Load: 2
  - Store: 1
  - Unconditional branch and bl: 2
  - Conditional branch: 1 if not taken (NT), 2 if taken (T)
- Subroutines execution times (measured in CPU cycles):
  - osRtxThreadListRemove/osRtxthreadListPut : 30 each
- Basic block 00003bfc may be executed from 0 to 10 times.

10. Complete this table with the minimum and maximum time (in cycles) to execute each basic block (including called subroutines) **once**. 5*(1/12) point per box

Basic Block	Min Exec Time C	Max Exec Time C
00003bec	7	7
00003bf2	2	3
00003bf6	4	5
00003bfc	4	4
00003c00	67	67
00003c0e	5	5

11. **ECE 561 Only:** What is the **minimum** execution time (measured in cycles) for the entire function (osRtxThreadListSort), including any subroutine calls?
15
Full credit if correct given #10 responses:

Basic Block	Min Exec Time C	Max Exec Time C
00003bec	**?*1**	
00003bf2	?	**?*1**
00003bf6	?	?
00003bfc	?	
00003c00	?	
00003c0e	**?*1**	

12. What is the **maximum** execution time (measured in cycles) for the entire function (osRtxThreadListSort) , including any subroutine calls?
186
1 point off if total doesn't include 10 iterations of 3bfc.
Full credit if correct given #10 responses:

Basic Block	Min Exec Time C	Max Exec Time C
00003bec	**?*1**	
00003bf2	**?*11**	?
00003bf6	**?*10**	**?*1**
00003bfc	**?*10**	
00003c00	**?*1**	
00003c0e	**?*1**	

System Concepts

13. Consider implementing a real-time system with ten threads. You can choose either preemptive or non-preemptive scheduling. Which will use less memory, and why?

The one with non-preemptive scheduling, as it only needs enough space for one stack: the deepest possible stack across all of the threads. The preemptive scheduling approach requires enough space to hold the deepest possible stack for **each of the threads simultaneously**.

14. A task's response time is the sum of its execution time, interference by other tasks and blocking by other tasks.
    a. Which other tasks can cause interference?
    Higher priority tasks

    b. Which other tasks can cause blocking?

    Preemptive system: Lower priority tasks on which this task is dependent: e.g. share a mutex, waiting on a semaphore/message/event.

    Non-preemptive system: all lower priority tasks.

Response Time and Schedulability – Version A

Consider a real-time system consisting of the following set of independent periodic tasks and the RTX5 RTOS configured for preemptive scheduling.

Task	Execution Time C (ms)	Period T (ms)
Fee	3	5
Fi	1	9
Fo	4	28

15. Calculate the utilization of the task set.
U = 3/5 + 1/9 + 4/28 = 0.854

16. Using the rate-monotonic approach to assign a priority (high (A) to low (C)) to each task.
    Fee: A. Fi: B. Fo: C.
17. Assume that each task has a deadline equal to its period. Does the utilization bound test show that this task set is **always schedulable** using fixed-priority **preemptive** scheduling with these priorities? Why or why not?

    RM Utilization Bound for three tasks is $U_{max}(3) = 3(2^{1/3}-1)=0.780$ and the task set utilization U = 0.854 is larger. So the test is inconclusive and doesn't show the task set is always schedulable.

18. Find the worst-case response time of the **highest** priority task when using **preemptive** fixed-priority scheduling.
    $R = C_{Fee} = 3$ ms
19. Find the worst-case response time of the **highest** priority task when using **non-preemptive** fixed-priority scheduling.
    $R = C_{Fee} + C_{Fo} = 7$ ms

20. **ECE 561 Only:** Find the worst-case response time of the **lowest** priority task when using **preemptive** fixed-priority scheduling.

$$R_i^0 = C_i + \sum_{j \in hp(i)} C_j = 4 + 3 + 1 = 8\ ms$$

$$R_i^1 = C_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i^0}{T_j} \right\rceil C_j = 4 + \left\lceil \frac{8}{5} \right\rceil 3 + \left\lceil \frac{8}{9} \right\rceil 1 = 4 + 2 * 3 + 1 * 1 = 4 + 6 + 1 = 11\ ms$$

$$R_i^2 = C_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i^1}{T_j} \right\rceil C_j = 4 + \left\lceil \frac{11}{5} \right\rceil 3 + \left\lceil \frac{11}{9} \right\rceil 1 = 4 + 3 * 3 + 2 * 1 = 4 + 9 + 2 = 15\ ms$$

$$R_i^3 = C_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i^2}{T_j} \right\rceil C_j = 4 + \left\lceil \frac{15}{5} \right\rceil 3 + \left\lceil \frac{15}{9} \right\rceil 1 = 4 + 3 * 3 + 2 * 1 = 4 + 9 + 2 = 15\ ms$$

The worst-case response time is 15 ms.

Response Time and Schedulability – Version B

Consider a real-time system consisting of the following set of independent periodic tasks and the RTX5 RTOS configured for preemptive scheduling.

Task	Execution Time C (ms)	Period T (ms)
Fee	3	6
Fi	1	7
Fo	5	28

15. Calculate the utilization of the task set.
    U = 3/6 + 1/7 + 5/28 = 0.8214

16. Using the rate-monotonic approach to assign a priority (high (A) to low (C)) to each task.
    Fee: A. Fi: B. Fo: C.

17. Assume that each task has a deadline equal to its period. Does the utilization bound test show that this task set is **always schedulable** using fixed-priority **preemptive** scheduling with these priorities? Why or why not?

    RM Utilization Bound for three tasks is $U_{max}(3) = 3(2^{1/3}-1)=0.780$ and the task set utilization U = 0. 8214 is larger. So the test is inconclusive and doesn't show the task set is always schedulable.

18. Find the worst-case response time of the **highest** priority task when using **preemptive** fixed-priority scheduling.
    $R = C_{Fee} = 3$ ms

19. Find the worst-case response time of the **highest** priority task when using **non-preemptive** fixed-priority scheduling.
    $R = C_{Fee} + C_{Fo} = 8$ ms

20. **ECE 561 Only:** Find the worst-case response time of the **lowest** priority task when using **preemptive** fixed-priority scheduling.

$$R_i^0 = C_i + \sum_{j \in hp(i)} C_j = 5 + 3 + 1 = 9\ ms$$

$$R_i^1 = C_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i^0}{T_j} \right\rceil C_j = 5 + \left\lceil \frac{9}{6} \right\rceil 3 + \left\lceil \frac{9}{7} \right\rceil 1 = 5 + 2 * 3 + 2 * 1 = 5 + 6 + 2 = 13\ ms$$

$$R_i^2 = C_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i^1}{T_j} \right\rceil C_j = 5 + \left\lceil \frac{13}{6} \right\rceil 3 + \left\lceil \frac{13}{7} \right\rceil 1 = 5 + 3 * 3 + 2 * 1 = 5 + 9 + 2 = 16\ ms$$

$$R_i^3 = C_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i^2}{T_j} \right\rceil C_j = 5 + \left\lceil \frac{16}{6} \right\rceil 3 + \left\lceil \frac{16}{7} \right\rceil 1 = 5 + 3 * 3 + 3 * 1 = 17\ ms$$

$$R_i^4 = C_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i^{n3}}{T_j} \right\rceil C_j = 5 + \left\lceil \frac{16}{6} \right\rceil 3 + \left\lceil \frac{16}{7} \right\rceil 1 = 5 + 3 * 3 + 3 * 1 = 5 + 9 + 3 = 17\ ms$$

The worst-case response time is 17 ms.