Name:_____ Email:_____@ncsu.edu

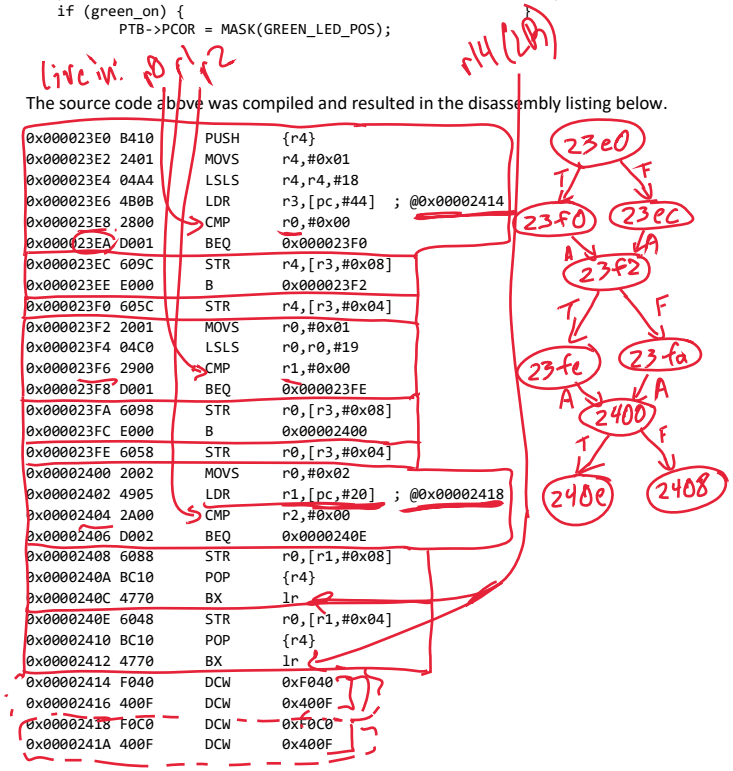
ECE 461/561, Spring 2022: Quiz 1: Examining Object Code

This quiz is closed-computer, closed-notes. You are allowed to bring in the ESF textbook, as Chapters 4 & 5 may be useful references. You may use one 8.5" x 11" sheet of paper with anything you want written or printed on its two sides.

```
void Control_RGB_LEDs(unsigned int red_on,
unsigned int green_on, unsigned int blue_on) {

    if (red_on) {
          PTB->PCOR = MASK(RED_LED_POS);
    } else {
          PTB->PSOR = MASK(RED_LED_POS);
    }
    if (green_on) {
          PTB->PCOR = MASK(GREEN_LED_POS);
    }  else {
          PTB->PSOR = MASK(GREEN_LED_POS);
    }
    if (blue_on) {
          PTD->PCOR = MASK(BLUE_LED_POS);
    }  else {
          PTD->PSOR = MASK(BLUE_LED_POS);
    }
}
```

The source code above was compiled and resulted in the disassembly listing below.



```
0x000023E0 B410      PUSH    {r4}
0x000023E2 2401      MOVS    r4,#0x01
0x000023E4 04A4      LSLS    r4,r4,#18
0x000023E6 4B0B      LDR     r3,[pc,#44]  ; @0x00002414
0x000023E8 2800      CMP     r0,#0x00
0x000023EA D001      BEQ     0x000023F0
0x000023EC 609C      STR     r4,[r3,#0x08]
0x000023EE E000      B       0x000023F2
0x000023F0 605C      STR     r4,[r3,#0x04]
0x000023F2 2001      MOVS    r0,#0x01
0x000023F4 04C0      LSLS    r0,r0,#19
0x000023F6 2900      CMP     r1,#0x00
0x000023F8 D001      BEQ     0x000023FE
0x000023FA 6098      STR     r0,[r3,#0x08]
0x000023FC E000      B       0x00002400
0x000023FE 6058      STR     r0,[r3,#0x04]
0x00002400 2002      MOVS    r0,#0x02
0x00002402 4905      LDR     r1,[pc,#20]  ; @0x00002418
0x00002404 2A00      CMP     r2,#0x00
0x00002406 D002      BEQ     0x0000240E
0x00002408 6088      STR     r0,[r1,#0x08]
0x0000240A BC10      POP     {r4}
0x0000240C 4770      BX      lr
0x0000240E 6048      STR     r0,[r1,#0x04]
0x00002410 BC10      POP     {r4}
0x00002412 4770      BX      lr
0x00002414 F040      DCW     0xF040
0x00002416 400F      DCW     0x400F
0x00002418 F0C0      DCW     0xF0C0
0x0000241A 400F      DCW     0x400F
```

Use the object code listing to answer the following questions.

All Students (ECE 461 and ECE 561)

1. Identify each basic block of code. If you have a printed quiz, draw a rectangle around each basic block in the object code listing above. If you do not have a printed quiz, you may list the starting address of each basic block.

2. Draw the code's control flow graph next to the object code listing above. If you do not have a printed quiz, draw the graph on your answer paper.
   - Represent each basic block by a rectangle labeled with the last four digits of its starting address.
   - Use arrows to show the control flow edges (arcs) between basic blocks. Label each control flow edge with A, T, or F to indicate under which condition the edge is taken (always, condition true, or condition false). Do not include control flow edges for subroutine returns.
3. List the address for each prolog instruction.

0x000023E0

4. How much stack space (in bytes) does the function use, and what is it used for?

4 bytes to save r4 so it can be restored in the epilog.

5. Identify which register is used to pass each of these arguments:
   - red_on r0
   - green_on r1
   - blue_on r2

6. Identify the address of the instruction which tests each of these arguments:
   - red_on 0x000023e8
   - green_on 0x000023f6
   - blue_on 0x00002404

[Note: Compare instruction addresses given above, but OK if you put BEQ instruction addresses instead (0x000023ea, 0x000023f8, 0x00002406).

7. List the address for each epilog instruction.

0x0000240a, 0x0000240c, 0x00002410, 0x00002412. There are two epilogs, since there are two subroutine returns.

8. Which register holds the function's return address?

LR (r14)

9. There are eight bytes of data located at 0x00002414 to 0x0000241B. What do their values represent?

0x00002414 holds the address of PTB (0x400ff040). 0x00002418 holds the address of PTB (0x400ff0c0)

10. Which register is used to hold MASK(GREEN_LED_POS), and which bit (from 0 to 31) in that register is a 1?

r0, bit 19. [Note that the value of red_on is not needed any more at this point in the program, so r0 is reused]

ECE 561 Students Only

For the next questions, consider the two STR instructions at 0x00002408 and 0x0000240E and the source code they implement.

11. What are r0 and r1 used for? Explain what parts of the source code they implement.

PTD->PCOR = MASK(BLUE_LED_POS);
PTD->PSOR = MASK(BLUE_LED_POS);
r0 holds the mask value (MASK(BLUE_LED_POS)), and r1 holds the starting address of PTD's control registers.

12. How are #0x08 and #0x04 used? Explain what parts of the source code they implement.

These are the offsets of PCOR and PSOR from the base address of PTD, used in the source code PTD->PCOR and PTD->PSOR.

PCOR has an 8 byte offset. PSOR has a 4 byte offset. These are added with r1 in the store instructions (e.g. STR r0, [r1, #0x08]) to form the address actually accessed (r1+8 or r1+4).