**EL 426 Midterm 1 practice test**

**25th Jan, 2019**

Note: I generally ask questions on ARM which take an average of about 1 minute to solve each question. So in one hour exam, you can expect 40 to 50 such questions. You should practice solving it using ARM quick reference.

1. Register r0 holds 0x01020304 and r1 holds the value 0x1000, then what would be the content of register r2 after following instruction. Assume big-endian memory system.

STR r0, [r1]

LDRH r2, [r1]

1. What constant would be loaded into register r5 by following instruction?

MVN r5, #0x0F, 6

1. Can an immediate value 0xAA000000 be loaded into any register using MOV instruction or not? Why?
2. Between SWI, FIQ, Data Abort and Pre-fetch abort, which exception has the highest priority?
3. If instruction LDMIB r5, {r0-r4} is executed then what will be loaded in register R1?
4. If r0 has the value 0x24, what is the content of r12 after executing the following instruction?

LDRB r12, [r0], #2

**Address Contents**

0x24 0x06

0x25 0xFC

0x26 0x03

0x27 0xFF

1. What is the significance of “!” in a load/store instruction?
2. If someone wants to copy the contents of any program status register in a register then which instruction needs to be used?
3. Register r9 contains 0x1000 initially. After executing following instruction, register r1 is loaded from which memory location?

LDMIA r9, {r2, r0, r1, r4}

1. If you want to store 1 to 250 decimal numbers in the memory in consecutive locations, then which instruction would be most memory efficient?
2. Write a single instruction which can multiply contents of register r4 by (128)10 and store the result to register r6, other than any multiply instruction.

1. What will be the value of register r1 after execution of following code? Assume code is saved form 0x0 memory location.

ldr r0, =0x4020

ldr r1, =const

str r1, [r0]

const DCD 0xFFFF0000

1. What is wrong with the following instruction?

STR r0, [r1]!, #10

1. Initially, r5 contains the value 0x1000. What will be the content of register r5 after following instruction?

STR r4, [r5, #10]

1. Branch instructions have the range of + 32 Mbytes, how many offset bits it requires? Why?

1. What will be the contents of register r0 after executing this code?

LDR R0,=1;

loop MOV R0,R0,LSL#1

BCC loop

1. When will the overflow flag be set? Give an example.