Department of Electrical and Computer Engineering
University of Alabama in Huntsville

CPE 323 – Introduction to Embedded Computer Systems
Midterm Exam

Instructor: Dr. Aleksandar Milenkovic

Date: October 07, 2015

Place: EB 207

Time: 2:20 PM – 03:40 PM

Note: Work should be performed systematically and neatly. This exam is closed books and closed neighbor(s). Allowable items include exam, pencils, straight edge, calculator, and materials distributed by the instructor (MSP430 ISA sheet). Good luck!

Question	Points	Score
1	15	
2	25	
3	20	
4	20	
5	20+5	
Sum	100+5	

Please print in capitals:

Last name: _____

First (Banner) name: _____

1. (15 points) Misc, MSP430

Circle the correct answer for A-E and type in the answers for F and G.

1.A. (True | False) (2 points) Assembly language directive "DC16 4" allocates 8 bytes in memory and initializes their values to 0x00.

1.B. (True | False) (2 points) Assembly language directive "DS32 8" allocates 32 bytes in memory.

1.C. (True | False) (2 points) Register R1 serves as the stack pointer (SP).

1.D. (True | False) (2 points) Flags C, V, Z, and N residing in the status register (R2) can be modified by software developers using assembly language instructions (e.g., BIC and BIS instructions).

1.E. (True | False) (2 points) A push onto the program stack will increase the value of the stack pointer.

decreases

1.F. (2 points) How many memory operations (reads from memory and writes to memory) occurs during execution of the instruction MOV.W #4523, &WDTCTL?

3 instruction reads
2 content reads
1 write

1.G. (3 points) What is the address range of a 4 KB block of data placed in memory at the address 0x0C00? Fill in the blanks.

_____ - _____

1kB = 2^{10} bytes

2. (25 points) Assembler (Directives, Instructions, Addressing Modes)

2.A. (5 points) Show the word-wide **HEXADECIMAL** content of memory corresponding to the following sequence of assembler directives. ASCII code for character 'A' is 65 decimal / 41 hex, and for character '0' is 48 decimal / 30 hex. Note: the number of rows does not reflect the number of words allocated by the directives.

```
// suffix q stands for octal, suffix b stands for binary, prefix 0x for hex
            ORG 0xE000
C1          DC8 0137q, 0xC8, 11000011b
            EVEN
C2          DC8 'A', '0', "BC"
            EVEN
C3          DC32 -6
```

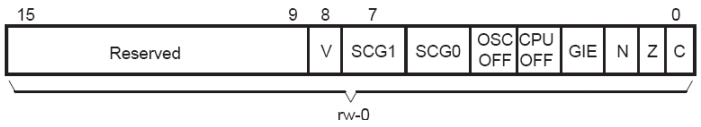
0000 0
0001 1
0010 2
0011 3
0100
0101
0110
0111

1 1 1 4
DC8 'A' '0' "BC"

Label	Address [hex]	Memory[15:0] [hex]
C1	E000	C8 | 5F
	E002	?? | C3
C2	E004	30 | 41
	E006	43 | 42
	E008	?? | 00
C3	E00A	FF | FA
	E00C	FF | FF
	E00E	

2.B. (20 points) Consider the following instructions given in the table below. For each instruction determine addressing modes of the source and destination operands, source and destination addresses, and the result of the operation. Fill in the empty cells in the table. The initial content of memory is given in the table. The initial value of registers R2, R5, and R6 is as follows: SR=R2=0x0001 (V=0, N=0, Z=0, C=1), R5=0xC003, R6=0xC006. Assume the starting conditions are the same for each question, i.e., always start from the initial conditions in memory and given register values.

Indirect / auto



Label	Address [hex]	Memory[15:0] [hex]
	0xC000	0x0504
	0xC002	0xFEEE
TONI	0xC004	0x9862
	0xC006	0x3344
	0xC008	0xF014
DEN	0xC00A	0x2244
EDE	0xC00C	0xCDDA
	0xC00E	0xEFDD

3344
+ 9862

	Instruction	Instr. Size in Words	Source Operand Addressing Mode	Destination Operand Addressing Mode	Source Address	Dest. Address	Result (content of a memory location or a destination register;
(a)	DADD.W @R6+, TONI	2	auto increment	Symbolic	C006	C004	3344 + 9862 = 3207 M[C004] ← 3207
(b)	SUBC &DEN, 8(R6)	3	Abs	Indexed	C00A	C00E	2244 EFDD 0001
(c)	XOR.W @R6, R5	2	Indirect	register	C006	—	

3. (20 points) Analyze assembly program

Consider the following code segment.

```
01        SUB     #4, SP    ; allocate 4 bytes (2 words on the stack)
02        MOV     mylw, 0(SP)           ;
03        MOV     mylw+2, 2(SP)         ;
04        BIT     #8000, 2(SP)          ;
05        JZ      lskip                 ;
06        INV     2(SP)                 ;
07        INV     0(SP)
08        ADD     #1, 0(SP)
09        ADDC    #0, 2(SP)
10 lskip:NOP
11
12 mylw DC32 0xFFFFFFFA
```

3.A. (2 points) How many bytes is allocated by the assembly directive in line 12?

4

3.B. (2 points) What is the content of register SP after the instruction in line 01 is completed? The initial value of SP is 0x1200.

SP = 0x11FC

3.C. (3 points) What is the content of memory locations at addresses [SP+0] and [SP+2] after the instructions in lines 02 and 03 are completed, respectively?

Mem[SP+0] = FFFA

Mem[SP+2]= FFFF

3.D. (3 points) What is the content of memory locations at addresses [SP+0] and [SP+2] after the program is executed (line 10)?

Mem[SP+0] = _____

Mem[SP+2]= _____

3.E. (8 points) What does this code segment do? Explain your answer.

3.F. (2 points) Calculate the total execution time in seconds for the code sequence from above (line 01 – line 10). We know the following: the average CPI is 2 clocks per instruction. Assume the clock frequency is 1 MHz. What is MIPS rate for this code?

4. (20 points, C language) Consider the following C program. Assume that the register SP at the beginning points to 0x0A00. Answer the following questions. Assume all variables are allocated on the stack, and in the order as they appear in the program. ASCII code for character '0' is 48 (0x30).

```
1  int main( void )  {
2    volatile int a = 4;
3    volatile long int c = -2, d = 2;
4    volatile char mych = {'4', '3', '2', '1'};
5    volatile long int *pli = &c;
6    Volatile int *pi = &a;
7      pli = pli - 1;   // lpa = lpa - 1*sizeof(long int)
8      pi = pi - 6;   //
9      *pi = a + *pi;
10 }
```

0x0A00	TOS	
0x09FE	0004	
0x09FC	FFFF	
0x09FA	FFFE	
0x09F8	0000	
0x09F6	0002	
0x09F4	31	32
0x09F2	33	34
0x09F0	0x09FA	
0x09EE	09FE	

->

Fill in the following table by determining the values/addresses given below.

9FE - C

#	Question?	Value/Address
1	The number of bytes allocated on the stack for the variable declared in line 2.	2
2	The number of bytes allocated on the stack for the character array declared in line 4.	4
3	The number of bytes allocated on the stack for all variables declared in lines 2-6.	18
(4)	Value of mych[0] after initialization performed in line 4.	34 '4'
5	Address of variable a (&a).	0x09FE
7	Value of pli at the moment after the statement in line 5 is executed.	0x09FA
8	Value of pli at the moment after the statement in line 7 is executed.	0x09F6
9	Value of pi at the moment after the statement in line 8 is executed.	0x09F2
10	Value of mych[0] at the moment after the statement in line 9 is executed.	38 '8'

Address	15-0	Comment

5. (20 points + 5 points bonus) Design and implement an MSP430 assembly language subroutine *int hex_alpha (int myw)* that processes an integer to determine the number of alphabetical symbols (A to F) in its hexademical representation. For example, hex_alpha(0xABBA)=4, and hex_alpha(0x345A)=1. The main program stores the input myw in the register R12 and the subroutine returns the result in the register R13.

(Bonus 5 points) Design and write a main program that calls the subroutine and displays the result on port 1.

```
#include <msp430.h>                 ; #define controlled include file
           PUBLIC  hex_alpha
           RSEG    CODE             ; place program in 'CODE' segment
hex_alpha:                          ; write your code below
```