ENGI 4862
Microprocessors

Assignment 2

Due Wednesday, 14 June 2017, 4:00 PM

1 MCU Families [6 marks, 3 each]

(a) For ATmega32, give address ranges of its general purpose registers, I/O registers, and internal RAM.

(b) Do these values differ for the ATmega8 and the ATmega 128? If so, how?

2 Valid Instructions [4 marks]

Which of the following are not valid instructions for the AVR ATmega32? (Use the instruction set reference if you are unsure.)

(a) `MUL R4, R5`

(b) `LDI R15, 0x1f`

(c) `SEZ`

(d) `ANDI R10, 0x13`

(e) `PUSH R6`

(f) `LDI R30, 200`

(g) `XOR R0, R1`

(h) `LDS 0x300, R15`

3 Status Flags [8 marks, 4 each]

Find the status of the C, H, Z, and N flags after the following operations.

(a)
```
LDI R16,0x1F
LDI R17,0x25
ADD R16, R17
```

(b)
```
LDI R18,0x73
LDI R19,0xF9
ADD R18, R19
```

4 Memory and I/O [10 marks, 5 each]

Write two versions of simple assembly programs to configure Port B for input and copy data from PINB to location 0x2ff in the data memory via R15 for ATmega32.

(a) For version (a), you may not use the LDS instruction.

(b) For version (b), you may not use the IN instruction.

5 Coding with Higher-Value Arithmetic [15 marks]

Write a short program to complete the following tasks (in the specified sequence):

- Use the .EQU directive to assign the names AH, AL, BH, and BL to the addresses $100, $101, $102, $103 respectively.
- Load the 16 bits stored in locations AH, AL into registers. (They represent a single 16-bit value.)
- Load the 16 bits stored in BH, BL into registers.
- Add the 16 bit values.
- Store the result in locations $200 and $201

6 Instruction Sizes [12 marks, 2 each]

Find the number of bytes that each of the following instructions takes (by checking the "Words" section of each instruction in the 8-Bit AVR Instruction Set available in the Course Manual (Part 1)):

(a) `CALL LABEL`

(b) `DEC R10`

(c) `RCALL LABEL`

(d) `OUT 0x18, R16`

(e) `STS 0x38, R16`

(f) `SUB R10, R1`

7 Delay Loops [10 marks]

Write simple code for a nested loop to perform a delay by decrementing 100,000 times.

8 Stack Manipulation [10 marks]

Assume that initially SPH=$08 and SPL=$5F, and that R20 = 10, R21 = 30, R22 = 40, and R23 = 50.

Find the contents of these registers, the stack, and the stack pointer after the execution of each of the following instructions:

```
PUSH R20
PUSH R22
POP R21
PUSH R23
POP R20
```

9 Swap Program [10 marks]

Write a short assembly program that swaps the contents of two registers without leaving any other registers in a modified state at the end of your program.

10 Subroutines and the Stack [15 marks]

It is common practice to save all the used registers at the beginning of a subroutine. Assume that we have SPH=$08, SPL=$5F before statement CALL PROC1 in the program below. Show the contents of the stack pointer and the exact memory contents of the stack after statement POP R0. You may choose initial values for the registers, but you must clearly state your assumptions.

Address	*Code*
0xAB00	`CALL PROC1`
0xAB02	`DEC R2`
...	...
PROC1:	`PUSH R16`
	`PUSH R17`
	`PUSH R0`
	`PUSH R1`
	`LDI R16, 0x7f`
	`LDI R17, 0xAC`
	`MUL R16, R17`
	`MOV R16, R0`
	`MOV R17, R1`
	`POP R1`
	`POP R0`
	...
	`RET`