

ENGI 4862

Microprocessors

Assignment 3

Due Wednesday, 05 July 2017, 4:00 PM

1 Flag Register [5 marks]

Determine the contents of register R20 and the six conditional/status flags (C, H, Z, N, V, S) after each of the following instructions executes. If a flag or register contents are unknown, indicate with an 'x'.

LDI	R20 , \$53
LDI	R16 , \$AC
DEC	R16
SEC	
ADC	R20 , R16
ANDI	R20 , \$30

2 Masking and Bit Manipulation [24, 8 marks each]

- (a) Write a program that configures Port B as an input and Port C as an output, then continuously reads from Port B, toggles the upper nibble, then swaps the upper and lower nibbles of that value, then writes the result to Port C.
(Toggling data can be accomplished using one's complement.)
- (b) Write a program to toggle the lower four bits of PORTD continuously without disturbing the rest of the bits. Among the PORTD lower four bits, make the ON/OFF pattern of the odd bits the same, but opposite from that of the even bits.
- (c) Write a program to monitor the PORTA.4 bit. When it is LOW, make a LOW-to-HIGH-to-LOW pulse on PORTB.7.

3 Sum of Products [16]

There are two arrays of 50 8-bit numbers located starting at memory locations 0x0400 and 0x0500, respectively. Write an assembly program that:

- Reads corresponding 8-bit elements of each array.
- Multiplies the corresponding elements together to produce a 16-bit result.
- Adds the 16-bit product to a running sum.
- When the whole array has been processed, store the 16-bit result to memory location 0x0300.

4 Division [10]

Write a subroutine called DIV to perform integer division by repeated subtraction. To maintain consistency with the MUL instruction, your operands will be passed in via R0 and R1 (i.e. you are calculating $R0 / R1$) and your result will be stored in those same registers: R0 will hold the quotient, and R1 the remainder.

5 Addressing Modes [10 marks, 2 each]

Identify the addressing mode used in each of the following instructions:

- (a) LD R20, X+
- (b) STS 0x0200, R3
- (c) ORI R16, 0x66
- (d) STD Y+4, R4
- (e) OUT PORTA, R2

6 Data in Program ROM [10, 2 marks each]

The following is defined in program ROM. Find the precise offset location in memory of each ASCII character or data (Textbook: Appendix-F for ASCII codes):

```
.ORG 0X200
Data1: .DB      12, 0X34
Data2: .DB      "24 468" ; note that a space is between two 4s.
      .ORG 0X270
Data3: .DW      0X1234, 0B101001111000
      .ORG 0X350
Data4: .DD      0X12345678
Data5: .DQ      0X12345678ABCDEF
```

7 Signed Operations [10, 2 marks each]

Write a sequence of code for each of the following operations and indicate the status of the V, N, and S flags for each. Hint: LDI R16, -12 is a legal instruction as -12 will be automatically converted to its two's complement by the assembler. You may want to verify this within AVR Studio.

- (a) (127) + (3)
- (b) (-113) + (+113)
- (c) (-100) + (-30)
- (d) (100) + (120)
- (e) (-125) + (-30H)

8 Type Conversion [15 marks]

Write an ATmega32-compatible program that reads the contents of the PORTC (which is connected to DIP switches), and then converts the 8-bit hexadecimal value into two 8-bit ASCII values representing each hex digit.

- “1010” to “1111” should be converted to ASCII values of the corresponding small letters.
- Store the lower digit in R30, and the upper digit in R31.
- This should be a full program, so include all the necessary statements as well as comments.