ECED 3204 Microprocessor

Final

Date: December 10 2016 **Room: Sexton Gym**

Time: 12:00pm-3:00pm

Non communication calculator is allowed

Student name	
ID	

1) Implement Macro for following subroutine and write the macro call as well.

[10points]

```
SwapRegs:   push    r16
            mov     r16,r17
            pop     r17
            ret
```

2) Write an ASM program to find the total elements of an array of 8-bit integers that have bit 1,3,5,7 set to 1. Assume the array is in the program memory.

[10points]

3) Read following program and determine the value of r0, r16, 17, r18, r19 when the program is done.

[10points]

```
            .include <m644Adef.inc>
            .equ    N1 = 0
            .equ    N2 = 30
            .def    ii = r16
            .def    count = r17
            .def    tmp = r18
            .macro setPointer
            ldi     @0,low(@2)
            ldi     @1,high(@2)
            .endmacro
            .cseg
            .org    0x00
            rjmp    start
            .org    0xF6
start:      clr     ii              ; initialize ii to N1
            clr     count           ; initialize count to 0
            ldi     tmp,50          ; value to be compared
            setPointer  ZL,ZH,(array<<1); use the Z register as a pointer to the array
floop:      cpi     ii,N2           ; check looping condition
            breq    done
            lpm     r0,z+           ; read the next array element
            cp      r0, tmp         ; is the element less than 100?
            brge    next            ; if yes, then branch to increment count
            inc     count
next:       inc     ii
            rjmp    floop
done:       rjmp    done
```

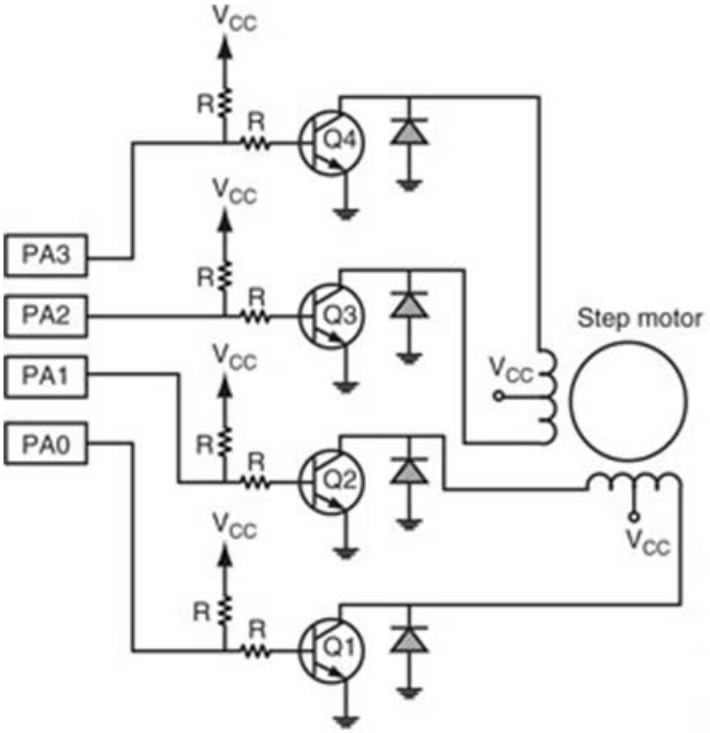
```
array:      .db     10,11,12,13,14,15,16,17,18,19
            .db     50,21,22,23,24,25,26,27,28,29
            .db     30,31,32,33,34,35,36,37,38,39
// End of program
```

4) Write a sequence of C statements to configure PCINT1, PCINT3, PCINT5, PCINT11, PCINT13, PCINT15, PCINT22 and PCINT23 to interrupt the MEGA CPU.

[10points]

5) Write a subroutine that rotates the stepper motor shown in figure clockwise rotation in half step sequence.
Assume the subroutine for delay is available **delayby10ms.**

[10points]



6) Write an AVR assembly program to generate a periodic square wave with 2-kHz frequency and 50% duty cycle using the Timer 1 CTC mode of the ATmega644 device. Square wave is on OC1A pin and use ICR1 to hold the TOP value assuming that fclk_I/O = 16 MHz. select clk_IO as clock input to Timer1 and prescaler to 1.

[10points]

7) Write a subroutine and a C function to configure the USART0 on Port D of the MEGA644A to operate with the following setting assuming that f_{OSC} = 32 MHz:

[10points]

- one start bit, eight data bits, one stop bit, no parity
- 38,400 baud rate
- Enable transmission and reception
- Disable multiprocessor mode
- Asynchronous double speed mode operation
- Disable interrupts

8) Configure the MEGA SPI module with the specified setting:

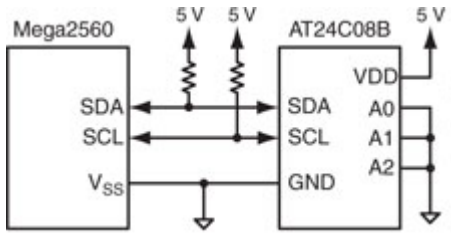
[10points]

Master mode with all interrupts disabled
SCK signal idle high and sample data on the leading edge
Data shifted least significant bit first
Baud rate set to 1MHZ assuming f_{SYS} = 16 MHz

9) For following connection between Mega250 and AT2408B, write a ASM subroutine to configure the TWI module to operate with the following parameters, assuming that the Mega device is running with a 10MHz

[10points]

200 KHz shift rate
Enable TWI
Enable TWI interrupt



10) Write a sequence of AVR instructions to configure the AVR MEGA2560 ADC to operate with the following settings:

[10points]

- Select differential mode input
- Select ADC9 and ADC8 as positive and negative inputs with gain set to 10
- Disable auto-triggering
- Set ADC clock prescaler to 32 assuming that the MEGA2560 uses a 16-MHz crystal oscillator to generate its system clock
- Select AVCC as its reference voltage
- Result left-justified
- Disable digital input buffers of the unused analog inputs
- Enable ADC interrupt