American International University – Bangladesh

Faculty of Engineering

Department of Electrical and Electronic Engineering

Course Name:	Microprocessor and Embedded Systems	**Course Code:**	EEE 4103
Semester:	Spring 2023-2024	**Section:**	
Faculty Name:	Protik Parvez Sheikh		
Assignment No:	3 **(individual submission consisting of 30 marks)**		
Student Name:			
Student ID:		**Program Name:**	**BSc in EEE**
Submission Date:		**Due Date:**	**11/05/2024**

Assessment Rubrics:

COs-POIs	Excellent [28-30]	Proficient [25-27]	Good [20-24]	Acceptable [10-19]	Unacceptable [1-9]	No Response [0]	Secured Marks
CO3 P.a.4.C.3	All the problems are solved correctly. The simulation processes are clearly described, and results are generated by combining all possible input patterns with appropriate outcomes. All necessary drawings and computations are shown.	All the problems are solved correctly. The simulation processes are clearly described, and results are generated by combining all possible input patterns with appropriate outcomes. A few necessary drawings and computations are missing.	All the problems are solved correctly. The simulation processes are not clearly described, and results are generated by combining all possible input patterns with appropriate outcomes. Some necessary drawings and computations are missing.	All the problems are not solved correctly. The simulation processes are not clearly described, and results are generated by combining several wrong input patterns with appropriate outcomes. Some necessary drawings and computations are missing.	All the problems are not solved correctly. The simulation processes are not described, and results are generated by combining mostly wrong input patterns with inappropriate outcomes. Almost all the necessary drawings and computations are missing.	No responses at all	
Comments					**Total marks (30)**		

Questions:

1. Find the baud rate for the asynchronous normal operating mode when the oscillator frequency, fOSC = 24 MHz, and register data is, UBRRn = 111010100101. Calculate the baud error and comment on whether there will be any communication errors or not.
   [For Arduino Uno, standard Baud rates maybe 300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, etc.]

2. Determine the necessary register setup to operate a microcontroller in the fast PWM mode in inverting mode. The counter should count to a maximum value of 235 and then reset to the BOTTOM and repeat. Draw the necessary timing waveform. Use a Timer0 of the Arduino Microcontroller.

3. Compute the duty cycle and sketch the waveforms obtained at port D of the Arduino. Identify the modes of operation and compute the operating frequency of that mode based on the following program segment. Identify the Timer of the Arduino Microcontroller. The system clock frequency is 8 MHz.

```
DDRD |= (1<<PD5);
pinMode(5, OUTPUT);
OCR0A = 200; // Load OCR0A register for setting the PWM frequency
OCR0B= 141; // Load OCR0B register for setting the duty cycle
// Configure TCCR0A and TCCR0B registers for the mode and pre-scaler
TCCR0A |= (1 << COM0B1) | (1 << COM0A0) | (1<<WGM01) | (1<<WGM00);
TCCR0B |= (1<<WGM02) | (1<<CS01) | (1<<CS00);
```

Table 1: Clock select function bits and corresponding pre-scaler values (L) and Compare output mode setting bits (R)

CSn2	CSn1	CSn0	Pre-scaler	COMnA1	COMnA0	Description
0	0	1	1	0	0	The normal port operation, OC0A disconnected
0	1	0	8	0	1	WGM02 = 0; Normal port operation, OC0A disconnected WGM02 = 1; Toggle OC0A on Compare Match
0	1	1	64	1	0	Clear OC0A on Compare Match, Set OC0A at BOTTOM (non-inverting mode)
1	0	0	256	1	1	Set OC0A on Compare Match, Clear OC0A at BOTTOM (inverting mode)
1	0	1	1024			

4. Design an adder/subtractor circuit with one selection variable ‘*S*’ and two inputs ‘*A*’ and ‘*B*’: when $S = 0$ the circuit performs $A + B$. When $S = 1$ the circuit performs $A – B$ by taking the 2’s complement of *B*.

5. Develop the control words in binary and hexadecimal formats using the information provided in Table 2 for the following micro-operations:
   i. R7←R3+R4
   ii. R3←SHL R3
   iii. R5←R1
   iv. R2←SHR R5
   v. R3←CRC R7

Table 2: Functions of control variables

Binary Code	**Functions of selection variables**					
	A	***B***	***D***	***F* with C_{in} = 0**	***F* with C_{in} = 1**	***H***
0 0 0	Input Data	Input Data	None	A-1	A	1’s to the output Bus
0 0 1	R1	R1	R1	A+B	A+B+1	Shift Left with $I_L = 0$
0 1 0	R2	R2	R2	A-B-1	A-B	No Shift
0 1 1	R3	R3	R3	A	A+1	Circulate Left with Carry
1 0 0	R4	R4	R4	$\bar{A}$	X	0’s to the output Bus
1 0 1	R5	R5	R5	AX**OR** B	X	-
1 1 0	R6	R6	R6	A **AND** B	X	Circulate-Right with Carry
1 1 1	R7	R7	R7	A **OR** B	X	Shift Right with $I_R = 0$

One example is shown as follows:

Micro-operation	*A*	*B*	*D*	*F*	C_{in}	*H*	In Hex
R5← CRC (R3+R4)	011	100	101	001	0	110	7296h

6. The microinstructions from **question 5** will be saved in a ROM, with each instruction being 14 bits long and stored sequentially starting from ROM location 000. To illustrate this, create a table (Table 3) detailing the contents of the control memory, adding rows as necessary.

Table 3: 14-bit Control Logic Output with ROM locations for Question 5

ROM Address			Control Word									Address			Mux Select	
A2	A1	A0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
			x	s2	s1	s0	Cin	L	y	z	w	A2	A1	A0	H1	H0
0	0	0														

7. Sketch a 4-bit status register with four flags: carry, overflow, zero, and sign. Predict the values of these flags after performing the operations outlined in **question 5**.

8. Prepare a flow chart that will count the number of 0's in register, R2, and then store the counts in register R5. Determine the outputs of the R5 (in binary) and R2 (in decimal) registers as well as of the carry flag after each clock cycle or timing state. Determine the number of states that are required to complete the operation.

Timing States	**R2**								**C**	**R5 (Decimal)**	**R5 (Binary)**
	1	1	0	1	0	0	1	0	1	1	00000001
T1											
T2											
T3											
T4											
T5											
T6											
T7											
T8											