ECE 241 **Home Work 4** **Spring 2017**

Due Date: May 3, 2017 **Points: 40**

This assignment will have to be done as a pdf or word document and uploaded to the course webpage.

1) Number Conversion: Convert the follow number to the requested base. (8 points)

A) $10100101_2 \rightarrow$ Four digit Hexadecimal
0x00A5

B) $1010_{10} \rightarrow$ 10 (16 bit signed number)
10

C) $55AA_{16} \rightarrow$ (16 bit binary)
21930

D) $2222_{10} \rightarrow$ Four digit Hexadecimal
0x08AE

2) Binary Addition: Show the results of the following binary additions, being sure to ***show all carries.*** Assume a zero carry in. (6 points)

A)

	0	0^1	1	0^1	1	0^1	1	0^0
+	0	0	1	0	1	0	1	0
	0	1	0	1	0	1	0	0

B)

	0^1	1^1	1	0	0	0	1	0^0
+	0	0	1	0	1	0	0	0
	1	0	0	0	1	0	1	0

C) Overflow: Interpreting the additions in question 2 as 8-bit signed numbers, do either have an overflow and why? (2 points)

No because the last digit is a zero carry.

3) Masking: Write a line of c code that performs the following operations, you can not use the bitSet or bitClear functions. (6)

A) Force bit 4 in the variable A, high.

```
A |= 0x01;
```

B) Force bit 9 in the variable B, low.

```
B &= ~0x002;
```

C) Toggle bit 6 in the variable A.

```
A ^= 0x06;
```

4) Conditionals: For what values of A do the following logical equations evaluate as true. (8)

A) if(A > 5 && A < 0)

A = is always false

B) if(A)

A = anything but 0

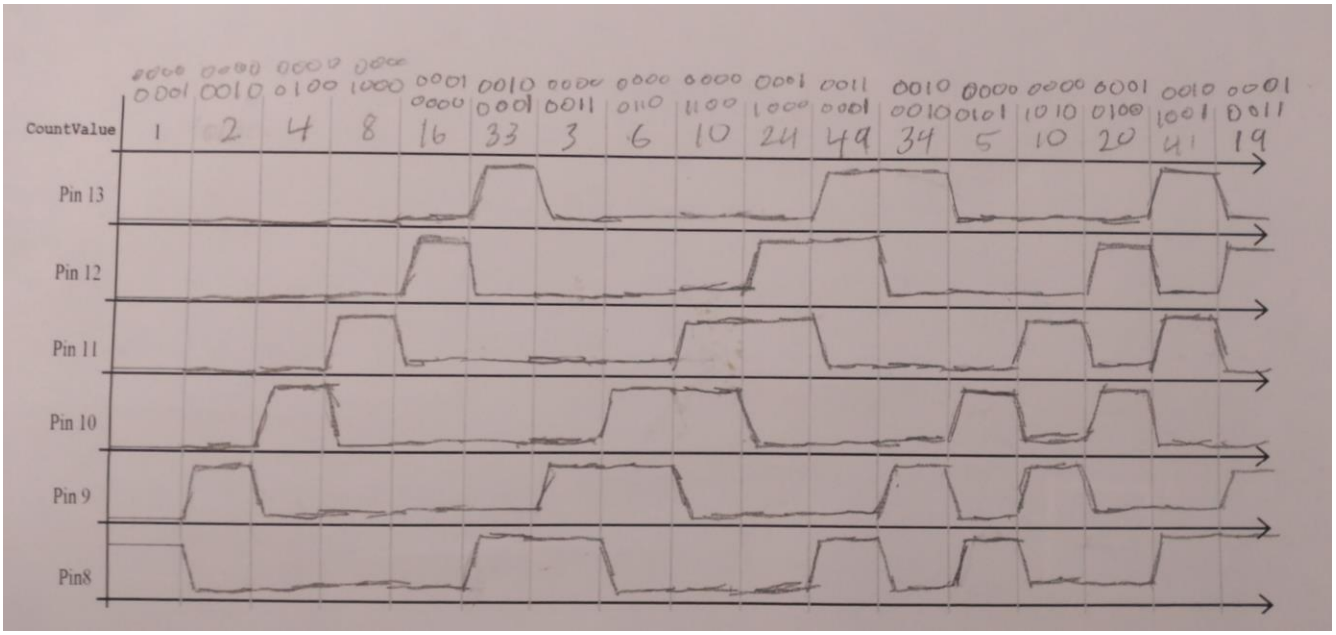
C) if(A < 10 && A > -10)

A = -9 through 9

D) if(A & 0x0042)

A = bits 9 or 12 are high

5) Code Reading/Timing: Fill in the timing diagram, as it would appear if the code in Appendix A were running. (10)



Appendix A: Output Function Generator.

```
#include <MsTimer2.h>
int CountValue = 1;
void CountValueISR()
{
      // Compute next output value.
      // if bit 5 set is equal to bit 4
      if (bitRead(CountValue, 5) == bitRead(CountValue, 4))
            CountValue = CountValue << 1; // shift to the left
      else
            CountValue = (CountValue << 1) // shift to the left
            + 1; // and set bit 0.
      CountValue &= 0x3f;
      // Send output to portb, or pins 13-8.
      PORTB = CountValue;
} // End of ISR

// Runs at start up to initialize system.
void setup()
{
      DDRB |= 0x3f; // Set pins 13-8 as outputs.
      // set up Timer2 for 50 millisecond interval.
      MsTimer2::set(50, CountValueISR);
      MsTimer2::start();
} // End of setup()

// loop called continuously.
0void loop()
{
      // Nothing in loop.
} // End of loop.
```