CPE301 - SPRING 2019

Design Assignment 2C

Student Name: Ron Joshua Recrio

Student #: 5003825419

Student Email: recrio@unlv.nevada.edu

Primary Github address: https://github.com/recrio

Directory: /DesignAssignments/DA2C

Submit the following for all Labs:

1. In the document, for each task submit the modified or included code (only) with highlights and justifications of the modifications. Also, include the comments.

- 2. Use the previously create a Github repository with a random name (no CPE/301, Lastname, Firstname). Place all labs under the root folder ESD301/DA, sub-folder named LABXX, with one document and one video link file for each lab, place modified asm/c files named as LabXX-TYY.asm/c.
- 3. If multiple asm/c files or other libraries are used, create a folder LabXX-TYY and place these files inside the folder.
- 4. The folder should have a) Word document (see template), b) source code file(s) and other include files, c) text file with youtube video links (see template).

1. COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS

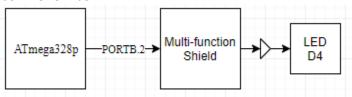
List of Components used

ATmega328p Xplained Mini

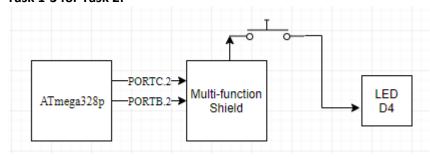
Multifunction Shield

Block diagram with pins used in the Atmega328P

Task 1-3 for Task 1:



Task 1-3 for Task 2:



2. INITIAL CODE OF DA2A

}

```
DA2A Task 1:
#include <stdio.h>
#include <avr/io.h>
int main(void)
    DDRB = (1<<2); //Make PB2 Output
      TCCR1B = 5; // set prescaler to 1024
    while (1)
             TCNT1 = 0; // set timer/counter to 0
             while (TCNT1 != 6796) {
              //do nothing
              PORTB ^= (1<<2); // toggle PB2 using xor
             TCNT1 = 0; // reset again
             while (TCNT1 != 4531) {
              // do nothing
             PORTB ^= (1<<2); // toggle PB2 using xor
    }
}
DA2A Task 2:
#define F_CPU 16000000UL //Change frequency to 16MHz
#include <avr/io.h>
#include <util/delay.h>
int main(void)
{
       DDRB = (1 << 2);
                           //PB2 is an output, XXXX X1XX
       PORTB |= (1 << 2);
                           //PB2 set as high to turn off LED, XXXX X1XX
       DDRC &= (0 << 2);
                           //PC2 is an input, XXXX X0XX
                           //PC2 set as low or unpressed, XXXX X0XX
       PORTC = (0 << 2);
      while (1) {
              if (!(PINC & (1 << PINC2))){ // If button is pressed</pre>
             PORTB &= \sim(1<<2);
                                  // turn on LED
                                  // delay for 1250ms
           else {
                                  // if button is not pressed
                 PORTB |= (1<<2); // set PB2 to high or LED off
              }
    }
```

3. DEVELOPED CODE OF TASK 1/C FROM DA2A

```
DA2C Task 1 for Task 1:
#define F_CPU 1600000UL
#include <avr/io.h>
#include <stdio.h>
int main(void)
       DDRB = (1<<DDB2); // Make PB2 Output
       PORTB = (0<<DDB2); // Turn on LED
       TCCR0A = 0; // Normal Mode
       TCCR0B = 5; // Set prescaler to 1024
       int ovrflow = 0; // overflow counter
    while (1)
    {
              TCNT0 = 0; // Reset counter
              ovrflow = 0; // Reset overflow counter
              // Delay for 6796
              while (ovrflow < 26) { // Gets to 6656
                     while ((TIFR0 & 0x01) == 0) {}
                     ovrflow++; // increment ovrflow
                     TCNT0 = 0; // reset counter
                     TIFR0 = 1; // reset ovf flag
              while (TCNT0 < 140) {} // 6656+140 = 6796
              PORTB ^= (1<<DDB2); // Turn off LED
              TCNT0 = 0; // reset counter
              ovrflow = 0; // reset ovrflow counter
              // Delay for 4531
              while (ovrflow < 17) { // 4352</pre>
                     while ((TIFR0 & 0x01) == 0) {}
                     ovrflow++; // increment ovrflow
                     TCNT0 = 0; // reset counter
                     TIFR0 = 1; // reset ovf flag
              while (TCNT0 < 179) {} // 4352+179 = 4531
              PORTB ^= (1<<DDB2); // Turn on LED
    }
}
```

DA2C Task 1 for Task 2:

```
#define F_CPU 16000000UL
#include <avr/io.h>
#include <stdio.h>
int main(void)
       DDRB = (1<<DDB2); // Make PB2 Output</pre>
       DDRC = (0<<DDC2); // Make PC2 Input
       PORTB = (1<<DDB2); // Turn off LED</pre>
       PORTC = (1<<DDC2); // Turn on pull-up transistor</pre>
       TCCR0A = 0; // Normal Mode
       TCCR0B = 5; // Set prescaler to 1024
       int ovrflow = 0; // overflow counter
       while (1)
              if (!(PINC & (1 << PINC2))) {      // if button pressed</pre>
              PORTB ^= (1<<DDB2); // Turn on LED
              TCNT0 = 0; // Reset counter
              ovrflow = 0; // Reset overflow counter
              // Delay for 1.25 sec (19531 TCNT)
              while (ovrflow < 76) { // Gets to 19456</pre>
                     while ((TIFR0 & 0x01) == 0) {}
                     ovrflow++; // increment ovrflow
                     TCNT0 = 0; // reset counter
                     TIFR0 = 1; // reset ovf flag
              while (TCNT0 < 75) {} // 19456+75 = 19531
              PORTB ^= (1<<DDB2); // Turn off LED
              }
       }
}
```

4. DEVELOPED CODE OF TASK 2/C FROM DA2A

```
DA2C Task 2 for Task 1:
#define F_CPU 1600000UL
#include <avr/io.h>
#include <stdio.h>
#include <avr/interrupt.h>
int ovrflow = 0; // global ovrflow counter
int main(void)
{
       DDRB = (1<<DDB2); // Make PB2 Output
       PORTB = (0 << DDB2); // Turn on LED
       TIMSK0 |= (1<<TOIE0); // Set up interrupt
       TCCR0A = 0; // Normal Mode
       sei(); // interrupt enable
       TCCR0B = 5; // Set prescaler to 1024
       while (1)
       {
       }
}
ISR (TIMER0 OVF vect) {
       ovrflow++; //increment ovrflow
       if (ovrflow == 26) { // delay for .435s
              TCNT0 = 0;
              while (TCNT0 < 140) {}
              PORTB ^= (1<<DDB2); // Turn OFF
              TCNT0 = 0; // reset counter
       else if (ovrflow == 43) { // delay for .29s
              TCNT0 = 0;
              while (TCNT0 < 179) {}
              PORTB ^= (1<<DDB2); // Turn ON
              ovrflow = 0; // reset ovrflow
              TCNT0 = 0; // reset counter
       TCNT0 = 0;
}
```

DA2C Task 2 for Task 2:

```
#define F_CPU 16000000UL
#include <avr/io.h>
#include <stdio.h>
#include <avr/interrupt.h>
int ovrflow = 0; // global ovrflow counter
int main(void)
       DDRB = (1<<DDB2); // Make PB2 Output
       DDRC = (0<<DDC2); // Make PC2 Input
       PORTB = (1<<DDB2); // Turn off LED</pre>
       PORTC = (1<<DDC2); // Turn on pull-up transistor</pre>
       TIMSK0 |= (1<<TOIE0); // Set up interrupt
       TCCR0A = 0; // Normal Mode
       sei();
       TCCR0B = 5; // Set prescaler to 1024
       while (1)
       {
              if (!(PINC & (1 << PINC2))) { // if button pressed</pre>
                     PORTB ^= (1 << DDB2); // Turn on LED
                     TCNT0 = 0;
                     ovrflow = 0;
                     while (!(PORTB & (1<<PORTB2))) {} // while on
              ovrflow = 0;
       }
}
ISR (TIMER0 OVF vect) {
       ovrflow++; //increment ovrflow
       if (ovrflow == 76) {
              TCNT0 = 0;
              while (TCNT0 < 75) {}
              PORTB ^= (1<<DDB2); // Turn off LED
       TCNT0 = 0; //reset counter
}
```

5. DEVELOPED CODE OF TASK 3/C FROM DA2A

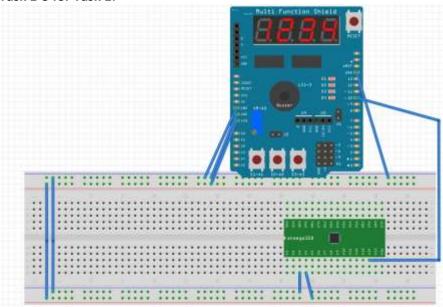
```
DA2C Task 3 for Task 1:
#define F_CPU 16000000UL
#include <avr/io.h>
#include <stdio.h>
#include <avr/interrupt.h>
int ovrflow = 0; // global ovrflow counter
int main(void)
{
       DDRB |= (1<<DDB2); // Make PB2 Output
       PORTB &= (0<<DDB2); // Turn on LED
       TIMSK0 |= (1<<OCIE0A); // Set up interrupt
       TCCR0A |= (1<<WGM01); // Normal Mode
       OCR0A = 0xFF;
       TCNT0 = 0;
       sei(); // interrupt enable
       TCCR0B |= (1<<CS02) | (1<<CS00); // Set prescaler to 1024
       while (1)
       {
       }
}
ISR (TIMERO_COMPA_vect) {
       ovrflow++; //increment ovrflow
       if (ovrflow == 26) { // delay for .435s
              TCNT0 = 0;
              while (TCNT0 < 140) {}
              PORTB ^= (1<<DDB2); // Turn OFF
              TCNT0 = 0; // reset counter
       }
       else if (ovrflow == 43) { // delay for .29s
              TCNT0 = 0;
              while (TCNT0 < 179) {}
              PORTB ^= (1<<DDB2); // Turn ON
              ovrflow = 0; // reset ovrflow
              TCNT0 = 0; // reset counter
       }
}
```

DA2C Task 3 for Task 2:

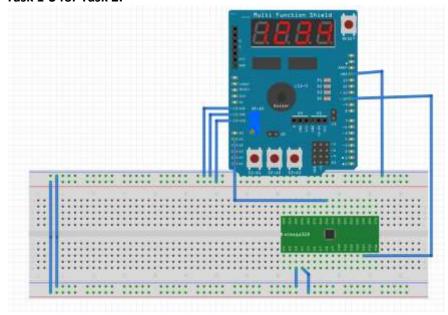
```
#define F_CPU 16000000UL
#include <avr/io.h>
#include <stdio.h>
#include <avr/interrupt.h>
int ovrflow = 0; // global ovrflow counter
int main(void)
       DDRB |= (1<<DDB2); // Make PB2 Output
       DDRC &= (0<<DDC2); // Make PC2 Input
       PORTB |= (1<<DDB2); // Turn off LED
       PORTC |= (1<<DDC2); // Turn on pull-up transistor
       TIMSK0 |= (1<<OCIE0A); // Set up interrupt
       TCCR0A |= (1<<WGM01); // Normal Mode
       OCR0A = 0xFF;
       TCNT0 = 0;
       sei();
       TCCR0B |= (1<<CS02) | (1<<CS00); // Set prescaler to 1024
       while (1)
       {
              if (!(PINC & (1 << PINC2))) \{ // \text{ if button pressed} \}
                     PORTB ^= (1 << DDB2); // Turn on LED
                     TCNT0 = 0;
                     ovrflow = 0;
                     while (!(PORTB & (1<<PORTB2))) {} // while on</pre>
              ovrflow = 0;
       }
}
ISR (TIMERO_COMPA_vect) {
       ovrflow++; //increment ovrflow
       if (ovrflow == 76) {
              TCNT0 = 0;
              while (TCNT0 < 75) {}
              PORTB ^= (1<<DDB2); // Turn off LED
       TCNT0 = 0; //reset counter
}
```

6. SCHEMATICS

Task 1-3 for Task 1:



Task 1-3 for Task 2:



7. SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT) DA2C Task 1_1 (BEFORE):

```
#define F CPU 16000000UL
      #include <avr/io.h>
      #include <stdio.h>
     □int main(void)
      {
                                                                          Processor Status
          DDRB = (1<<DDB2); // Make PB2 Output
                                                                               Name
                                                                                               Value
          PORTB = (0<<DDB2); // Turn on LED
                                                                           Program Counter 0x00000048
          TCCR0A = 0; // Normal Mode
                                                                           Stack Pointer
                                                                                          0x08FD
          TCCROB = 5; // Set prescaler to 1024
                                                                           X Register
                                                                                          0x0000
          int ovrflow = 0; // overflow counter
                                                                                          0x08FF
                                                                           Y Register
          while (1)
                                                                           Z Register
                                                                                          0x0000
                                                                           Status Register
                                                                                          ITHSVNZC
               TCNT0 = 0; // Reset counter
                                                                           Cycle Counter
               ovrflow = 0; // Reset overflow counter
                                                                                          16.000 MHz
                                                                           Frequency
               // Delay for 6796
                                                                           Stop Watch
                                                                                          1.25 µs
               while (ovrflow < 26) { // Gets to 6630
                                                                           ■ Registers
                   while ((TIFR0 & 0x01) == 0) {}
                                                                            R00
                                                                                          0x00
                   ovrflow++; // increment ovrflow
                                                                            R01
                                                                                          0x00
                   TCNT0 = 0; // reset counter
                   TIFR0 = 1; // reset ovf flag
               while (TCNT0 < 140) {} // 6656+140 = 6796
               PORTB ^= (1<<DDB2); // Turn off LED
               TCNT0 = 0; // reset counter
               ovrflow = 0; // reset ovrflow counter
               // Delay for 4531
               while (ovrflow < 17) { // 4352
                   while ((TIFR0 & 0x01) == 0) {}
                   ovrflow++: // increment ovrflow
110 % +
```

DA2C Task 1_1 (AFTER):

```
#define F CPU 16000000UL
        #include <avr/io.h>
        #include <stdio.h>
      □int main(void)
        {
                                                                              Processor Status
                                                                                                            ▼ 🗖 X
            DDRB = (1<<DDB2); // Make PB2 Output
                                                                                                    Value
                                                                                   Name
            PORTB = (0<<DDB2); // Turn on LED
                                                                               Program Counter 0x00000057
            TCCR0A = 0; // Normal Mode
                                                                               Stack Pointer
                                                                                              0x08FD
            TCCR0B = 5; // Set prescaler to 1024
                                                                               X Register
                                                                                              0x0000
            int ovrflow = 0; // overflow counter
                                                                               Y Register
                                                                                              0x08FF
            while (1)
                                                                                              0x0000
                                                                               Z Register
                                                                               Status Register
                                                                                              UTHSVNZC
                                                                                              6959129
                                                                               Cycle Counter
                ovrflow = 0; // Reset overflow counter
                                                                               Frequency
                                                                                              16.000 MHz
                // Delay for 6796
                                                                               Stop Watch
                                                                                              434.95 ms
                while (ovrflow < 26) { // Gets to 6630
                                                                               ■ Registers
                     while ((TIFR0 \& 0x01) == 0) {}
                                                                                R00
                                                                                               0x00
                     ovrflow++; // increment ovrflow
                                                                                R01
                                                                                               0x00
                     TCNT0 = 0; // reset counter
                     TIFR0 = 1; // reset ovf flag
                while (TCNT0 < 140) {} // 6656+140 = 6796
                PORTB ^= (1<<DDB2); // Turn off LED
                 TCNT0 = 0; // reset counter
                ovrflow = 0; // reset ovrflow counter
                 // Delaw fem 4534
      □int main(void)
                                                                             Processor Status
           DDRB = (1<<DDB2); // Make PB2 Output
                                                                                 Name
                                                                                                  Value
            PORTB = (0<<DDB2); // Turn on LED
                                                                              Program Counter
                                                                                             0x00000048
            TCCR0A = 0; // Normal Mode
                                                                              Stack Pointer
                                                                                             0x08FD
            TCCROB = 5; // Set prescaler to 1024
                                                                              X Register
                                                                                             0x0000
            int ovrflow = 0; // overflow counter
                                                                              Y Register
                                                                                             0x08FF
            while (1)
                                                                                             0x0000
                                                                              Z Register
                                                                              Status Register
                                                                                             ITHSVNZC
                TCNT0 = 0; // Reset counter
                                                                              Cycle Counter
                                                                                             11598874
                ovrflow = 0; // Reset overflow counter
                                                                              Frequency
                                                                                             16.000 MHz
                // Delay for 6796
                                                                              Stop Watch
                                                                                             724.93 ms
                while (ovrflow < 26) { // Gets to 6630
                                                                             Registers
                    while ((TIFR0 & 0x01) == 0) {}
                                                                              R00
                                                                                             0x00
                    ovrflow++; // increment ovrflow
                                                                              R01
                                                                                             0x00
                    TCNT0 = 0; // reset counter
                    TIFR0 = 1; // reset ovf flag
                while (TCNT0 < 140) {} // 6656+140 = 6796
                PORTB ^= (1<<DDB2); // Turn off LED
TCNTA = 0: // reset counter
DA2C Task 1 1 Waveform:
                     — W 0.435 s ----- II 1.379 Hz → 🚺 0.725 s
```

DA2C Task 1_2 (BEFORE):

```
#include <avr/io.h>
 #include <stdio.h>
mint main(void)
 1
      DDRB - (1<<DDB2); // Make P82 Output
      DDRC = (0<<DDC2); // Make PC2 Input
                                                                       Program Counter 0x0000004F
      PORTB - (1<<DDB2); // Turn off LED
                                                                       Stack Pointer
                                                                                      CxO8FD:
      PORTC + (1<<DDC2); // Turn on pull-up transistor
                                                                       X Register
                                                                                      0x0000
      TCCR0A = 0; // Normal Mode
                                                                       Y Register
                                                                                      0x08FF
      TCCR08 = 5; // Set prescaler to 1024
                                                                                      Dx0000
                                                                       Z Register
      int ovrflow = 0; // overflow counter
                                                                       Status Register
                                                                                      00820860
      while (1)
                                                                       Cycle Counter
          if (!(PINC & (1 << PINC2))) { // if button pressed
                                                                                      16.000 MHz
                                                                       Frequency
          PORTS ~= (1<<0082); // Turn on LED
                                                                       Stop Watch
                                                                                      1:69 us
          TCNT0 = 0; // Reset counter
ovrflow = 0; // Reset overflow counter
                                                                       Registers
                                                                        Rost
                                                                                       0,00
                                                                        R01
                                                                                       0x00
          // Delay for 1.25 sec (19531 TCNT)
          while (ovrflow < 76) ( // Gets to 19456
              while ((TIFR0 & 0x01) == 0) {}
              ovrflow++; // increment ovrflow
              TCNT0 = 0; // reset counter
              TIFRO - 1; // reset ouf flag
          while (TCNT0 < 75) {} // 19456+75 = 19531
         PORTS ~= (1ckDOS2); // Turn off LED
```

DA2C Task 1_2 (AFTER):

```
Eint main(void)
     DDR8 - (1<<DD82); // Make P82 Output
                                                                         Name
                                                                                          Value
     DDRC = (0<<DDC2); // Make PC2 Input
                                                                      Program Counter 0x00000058
     PORTB - (1<<DOB2); // Turn off LED
                                                                      Stack Pointer
                                                                                     O<sub>V</sub>ORFO
     PORTC - (1<<DOC2); // Turn on pull-up transistor
                                                                                     040000
                                                                      X Register
     TCCR0A = 0; // Normal Mode
                                                                                     OxOBFF
                                                                      V Register
      TCCR08 - 5; // Set prescaler to 1024
      int ovrflow - 0; // overflow counter
                                                                      Z Register
                                                                                     0x0000
     while (1)
                                                                      Status Register
                                                                                    Cycle Counter
                                                                                     19999770
          if (!(PINC & (1 << PINC2))) { // if button pressed
                                                                                     16,000 MHz
                                                                      Frequency
         PORTS *= (1<<DDB2); // Turn on LED
                                                                      Stop Watch
                                                                                     1,249.99 ms
         TCNT0 = 0; // Reset counter
                                                                      iii Registers
          ovrflow = 0; // Reset overflow counter
                                                                       R00
                                                                                     0400
                                                                       R01
                                                                                     0x00
          // Delay for 1.25 sec (19531 TCNT)
         while (ovrflow < 76) { // Gets to 19456
              while ((TIFR0 & 0x01) == 0) {}
              ovrflow++; // increment ovrflow
              TCNT0 = 0; // reset counter
              TIFR0 = 1; // reset ovf flag
         while (TCNT8 < 75) {} // 19456+75 = 19531
          BORTE A. (16/0082): // Turn off LED
```

DA2C Task 1_2 Waveform:

DA2C Task 2_1 (BEFORE):

```
#define F_CPU 16000000UL
       #include <avr/io.h>
      #include <stdio.h>
      #include <avr/interrupt.h>
      int ovrflow = 0; // global ovrflow counter
                                                                                                                     ▼ □ ×
                                                                                     Processor Status

    int main(void)

                                                                                          Name
                                                                                                             Value
                                                                                      Program Counter 0x00000052
           DDRB = (1<<DDB2); // Make PB2 Output
                                                                                      Stack Pointer
                                                                                                       0x08FD
           PORTB = (0<<DDB2); // Turn on LED
           TIMSK0 |= (1<<TOIE0); // Set up interrupt
                                                                                      X Register
                                                                                                       0x0102
           TCCR0A = 0; // Normal Mode
                                                                                      Y Register
                                                                                                       0x08FF
           sei(); // interrupt enable
                                                                                                       0x006E
                                                                                      Z Register
           TCCR0B = 5; // Set prescaler to 1024
0
                                                                                      Status Register
                                                                                                       THSVNZC
           while (1)
                                                                                      Cycle Counter
           {
                                                                                                       16.000 MHz
                                                                                      Frequency
                                                                                      Stop Watch
                                                                                                       2.75 µs
      }
                                                                                     ■ Registers
     ☐ ISR (TIMERØ_OVF_vect) {
                                                                                       R00
                                                                                                       0x00
           ovrflow++; //increment ovrflow
                                                                                       R01
                                                                                                       0x00
           if (ovrflow == 26) \{ // \text{ delay for .435s} \}
               TCNT0 = 0;
               while (TCNT0 < 140) {}
              PORTB ^= (1<<DDB2); // Turn OFF
TCNT0 = 0; // reset counter
           else if (ovrflow == 43) { // delay for .29s
               TCNT0 = 0;
               while (TCNT0 < 179) {}
               PORTB ^= (1<<DDB2); // Turn ON
               ovrflow = 0; // reset ovrflow
               TCNT0 = 0; // reset counter
```

DA2C Task 2_1 (AFTER):

```
Processor Status
                                                                                           ▼ 🗖 X
                                                                 Name
                                                                                   Value
                                                             Program Counter
                                                                             0x00000070
 □ISR (TIMERØ_OVF_vect) {
       ovrflow++; //increment ovrflow
                                                             Stack Pointer
                                                                             0x08F6
       if (ovrflow == 26) { // delay for .435s
                                                             X Register
                                                                             0x0102
           TCNT0 = 0;
                                                             Y Register
                                                                             0x08FF
           while (TCNT0 < 140) {}
           PORTB ^= (1<<DDB2); // Turn OFF
                                                             Z Register
                                                                             0x006E
           TCNT0 = 0; // reset counter
                                                             Status Register
                                                                             ITHSVNZC
                                                             Cycle Counter
                                                                             6959159
       else if (ovrflow == 43) { // delay for .29s
           TCNT0 = 0;
                                                             Frequency
                                                                             16.000 MHz
           while (TCNT0 < 179) {}
                                                             Stop Watch
                                                                             434.95 ms
           PORTB ^= (1<<DDB2); // Turn ON
                                                             Registers
           ovrflow = 0; // reset ovrflow
           TCNT0 = 0; // reset counter
                                                              R00
                                                                              0x00
                                                                              0x00
                                                              R01
       TCNT0 = 0;
      }
                                                               Processor Status
                                                                    Name
                                                                                       Value
                                                                Program Counter
                                                                                 0x00000080
□ISR (TIMER0_OVF_vect) {
      ovrflow++; //increment ovrflow
                                                                Stack Pointer
                                                                                 0x08F6
      if (ovrflow == 26) { // delay for .435s
                                                                                 0x0102
                                                                X Register
          TCNT0 = 0;
                                                                                 0x08FF
                                                                Y Register
          while (TCNT0 < 140) {}
          PORTB ^= (1<<DDB2); // Turn OFF
                                                                Z Register
                                                                                 0x006E
          TCNT0 = 0; // reset counter
                                                                Status Register
                                                                                 Cycle Counter
                                                                                 11598907
      else if (ovrflow == 43) { // delay for .29s
          TCNT0 = 0;
                                                                Frequency
                                                                                 16.000 MHz
          while (TCNT0 < 179) {}
                                                                Stop Watch
                                                                                 724.93 ms
          PORTB ^= (1<<DDB2); // Turn ON
                                                               Registers
          ovrflow = 0; // reset ovrflow
          TCNT0 = 0; // reset counter
                                                                 R00
                                                                                  0x00
                                                                 R01
                                                                                  0x00
      TCNT0 = 0;
```

DA2C Task 2 1 Waveform:

DA2C Task 2_2 (BEFORE):

```
UUKC = (שללטטכב); // make PCZ Impuc
     PORTB = (1<<DDB2); // Turn off LED
                                                         Processor Status
                                                                                         ▼ 🗖 X
     PORTC = (1<<DDC2); // Turn on pull-up transistor
                                                              Name
                                                                                Value
     TIMSK0 |= (1<<TOIE0); // Set up interrupt
                                                          Program Counter
                                                                          0x00000056
     TCCR0A = 0; // Normal Mode
                                                          Stack Pointer
                                                                           0x08FD
     TCCR0B = 5; // Set prescaler to 1024
                                                                           0x0102
                                                          X Register
     while (1)
                                                          Y Register
                                                                           0x08FF
                                                                           0x006E
          if (!(PINC & (1 << PINC2))) { // if button p
                                                          Z Register
              PORTB ^= (1 << DDB2); // Turn on LED
                                                          Status Register
                                                                           THSVNZC
              TCNT0 = 0;
                                                          Cycle Counter
                                                                           48
              ovrflow = 0;
                                                                           16.000 MHz
              while (!(PORTB & (1<<PORTB2))) {} // whi Frequency
                                                          Stop Watch
                                                                           3.00 µs
          ovrflow = 0;
                                                          Registers
     }
                                                           R00
                                                                           0x00
 }
                                                                           0x00
                                                           R01
⊟ISR (TIMERØ OVF vect) {
     ovrflow++; //increment ovrflow
```

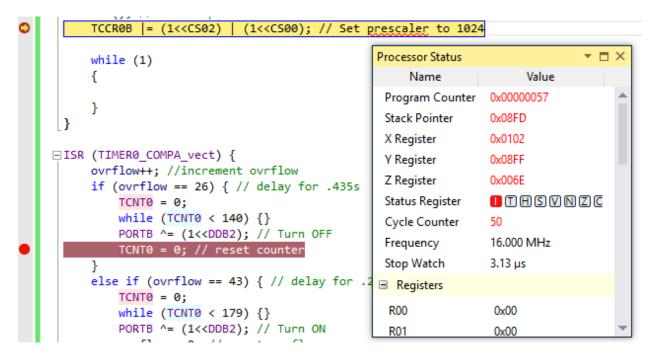
DA2C Task 2_2 (AFTER):

```
while (!(PORTB & (1<<PORTB2))) {} //
                                                       Processor Status
                                                                                        ▼ 🗖 X
                                                                               Value
                                                            Name
          ovrflow = 0;
                                                        Program Counter
                                                                         0x0000007F
                                                        Stack Pointer
                                                                         0x08F6
      }
 }
                                                        X Register
                                                                         0x0102
                                                        Y Register
                                                                         0x08FF
□ISR (TIMER0_OVF_vect) {
                                                        Z Register
                                                                         0x006E
      ovrflow++; //increment ovrflow
                                                                         THSVNZC
      if (ovrflow == 76) {
                                                        Status Register
          TCNT0 = 0;
                                                                         19999796
                                                        Cycle Counter
          while (TCNT0 < 75) {}
                                                                         16.000 MHz
                                                        Frequency
          PORTB ^= (1<<DDB2); // Turn off LED
                                                        Stop Watch
                                                                         1,249.99 ms
      TCNT0 = 0; //reset counter
                                                       Registers
                                                         R00
                                                                          0x02
                                                         R01
                                                                          0x00
```

DA2C Task 2_2 Waveform:

```
₩ 1.25 s 0.7999 Hz № 1.25 s H 1.25 s
```

DA2C Task 3_1 (BEFORE):



DA2C Task 3_1 (AFTER):

```
Processor Status
                                                                                       ▼ 🗆 ×
          while (1)
                                                            Name
                                                                              Value
                                                        Program Counter 0x00000076
                                                        Stack Pointer
                                                                        0x08F6
      }
                                                        X Register
                                                                        0x0102
     □ ISR (TIMER0_COMPA_vect) {
                                                        Y Register
                                                                        0x08FF
          ovrflow++; //increment ovrflow
                                                        Z Register
                                                                        0x006E
           if (ovrflow == 26) { // delay for .435s
                                                        Status Register
                                                                        ITHSVNZC
               TCNT0 = 0;
               while (TCNT0 < 140) {}
                                                        Cycle Counter
                                                                        6959165
               PORTB ^= (1<<DDB2); // Turn OFF
                                                        Frequency
                                                                        16.000 MHz
0
               TCNT0 = 0; // reset counter
                                                        Stop Watch
           else if (ovrflow == 43) { // delay for .
                                                       Registers
               TCNT0 = 0;
                                                         R00
                                                                         0x00
               while (TCNT0 < 179) {}
               PORTB ^= (1<<DDB2); // Turn ON
                                                         R01
                                                                         0x00
               sunflow - Qr // posst sunflow
                                                      Processor Status
                                                                                      ▼ 🗖 X
          while (1)
                                                           Name
                                                                             Value
                                                                        0x00000086
                                                       Program Counter
                                                       Stack Pointer
                                                                        0x08F6
     }
                                                       X Register
                                                                        0x0102
    □ISR (TIMERØ COMPA vect) {
                                                                        0x08FF
                                                       Y Register
          ovrflow++; //increment ovrflow
                                                       Z Register
                                                                        0x006E
          if (ovrflow == 26) { // delay for .435s
                                                       Status Register
                                                                        TCNT0 = 0;
              while (TCNT0 < 140) {}
                                                       Cycle Counter
                                                                        11598914
              PORTB ^= (1<<DDB2); // Turn OFF
                                                                        16.000 MHz
                                                       Frequency
              TCNT0 = 0; // reset counter
                                                                        724.93 ms
                                                       Stop Watch
          else if (ovrflow == 43) { // delay for .2
                                                      ■ Registers
              TCNT0 = 0;
                                                        R00
                                                                        0x00
              while (TCNT0 < 179) {}
              PORTB ^= (1<<DDB2); // Turn ON
                                                        R01
                                                                        0x00
              ovrflow = 0; // reset ovrflow
              TCNT0 = 0; // reset counter
```

DA2C Task 3_1 Waveform:

DA2C Task 3_2 (BEFORE):

```
Processor Status
                                                                               ▼ 🗖 X
          if (!(PINC & (1 << PINC2))) { //
                                                   Name
                                                                      Value
               PORTB ^= (1 << DDB2); // Tur
                                               Program Counter
                                                                0x0000005C
               TCNT0 = 0;
                                               Stack Pointer
                                                                0x08FD
              ovrflow = 0;
                                               X Register
                                                                0x0102
              while (!(PORTB & (1<<PORTB2)
                                               Y Register
                                                                0x08FF
          ovrflow = 0;
                                               Z Register
                                                                0x006E
                                               Status Register
                                                                THSVNZC
      }
 }
                                                Cycle Counter
                                                                57
                                               Frequency
                                                                16.000 MHz
□ISR (TIMER0_COMPA_vect) {
                                               Stop Watch
                                                                3.56 µs
      ovrflow++; //increment ovrflow
      if (ovrflow == 76) {
                                               Registers
          TCNT0 = 0;
                                                R00
                                                                 0x00
          while (TCNT0 < 75) {}
          PORTB ^= (1<<DDB2); // Turn off
                                                R01
                                                                 0x00
```

DA2C Task 3_2 (AFTER):

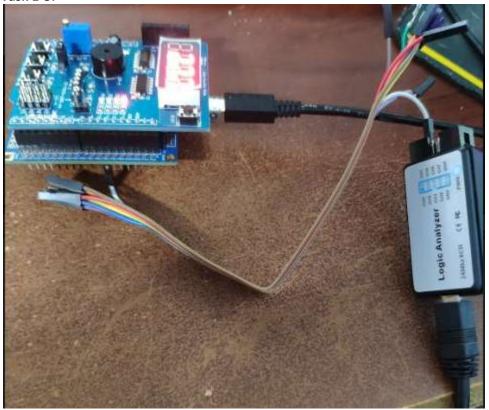
```
if (!(PINC & (1 << PINC2))) { //
                                                                    Value
                                                  Name
              PORTB ^= (1 << DDB2); // Tur
                                              Program Counter
                                                               0x00000085
              TCNT0 = 0;
                                               Stack Pointer
                                                               0x08F6
              ovrflow = 0;
              while (!(PORTB & (1<<PORTB2)
                                              X Register
                                                               0x0102
                                               Y Register
                                                               0x08FF
          ovrflow = 0;
                                               Z Register
                                                               0x006E
                                               Status Register
                                                               }
                                               Cycle Counter
                                                               19999805
                                              Frequency
                                                               16.000 MHz
□ ISR (TIMER0_COMPA_vect) {
                                                               1,249.99 ms
                                               Stop Watch
     ovrflow++; //increment ovrflow
     if (ovrflow == 76) {
                                              Registers
          TCNT0 = 0;
                                               R00
                                                               0x02
          while (TCNT0 < 75) {}
          PORTB ^= (1<<DDB2); // Turn off
                                               R01
                                                               0x00
```

DA2C Task 3_2 Waveform:

```
→ |← W 1.25 s 1 0.7999 Hz (duty) 0.1467 m% 1 1.25 s → |
```

8. SCREENSHOT OF EACH DEMO (BOARD SETUP)

Task 1-3:



9. VIDEO LINKS OF EACH DEMO

https://youtu.be/hjMDncpwhul

10. GITHUB LINK OF THIS DA

https://github.com/recrio/submissions/tree/master/DesignAssignments/DA2C

Student Academic Misconduct Policy

http://studentconduct.unlv.edu/misconduct/policy.html

"This assignment submission is my own, original work".

Ron Joshua Recrio