Temple University

College of Engineering

Department of Electrical and Computer Engineering (ECE)

Student Lab Report Cover Page

Course Number : 3613

Course Section : 002

Experiment # : Lab # 12

Student Name (print) : Robert Bara

TUid# : 915614617

Date : 12/4/2020

Grade : ______ **/100**

TA Name : Sung Choi

LAB 12 ACTIVITIES:

Activity 1

1. Code and Description

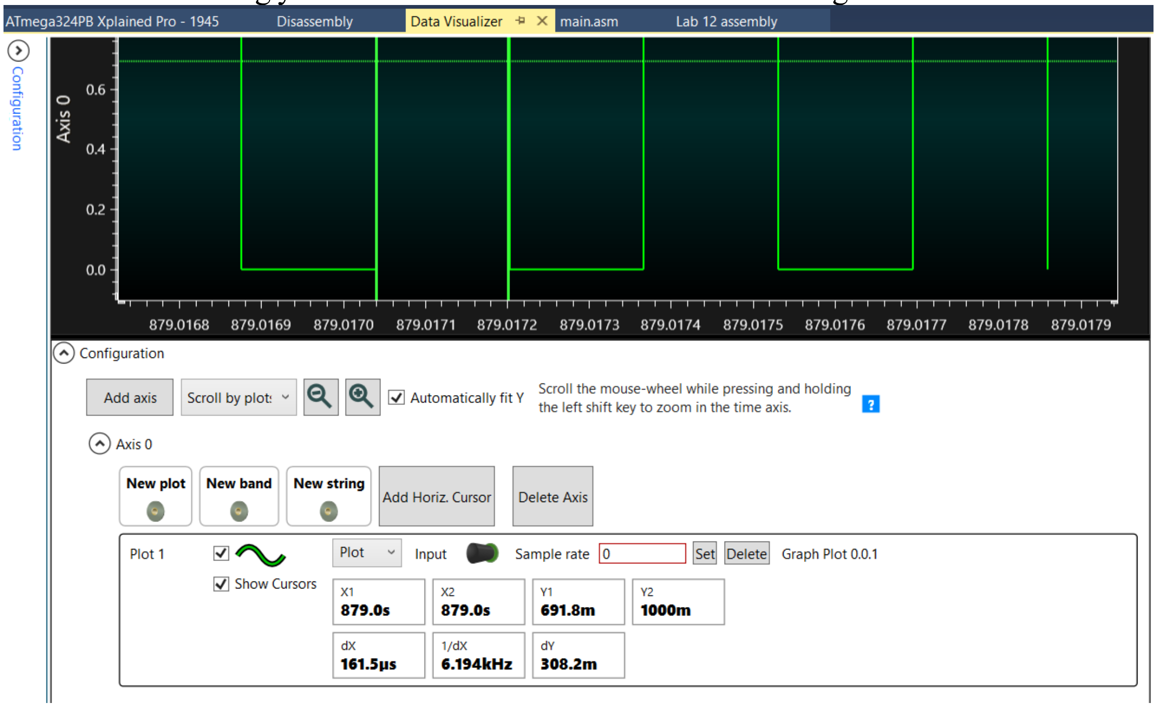
Code (Full-Comment):

```
//Activity 1: Robert Bara

//Timer Period=1/3kHz=333.33usec, 1/16MHz=0.0625us from Stopwatch
//tick=333.33us/0.0625us=5334us
//For 50% Duty Cycle, 5334us/2=2667us
//On for 333.33us/2=166.67us
//Prescaler 2667/64=41.67=41 for TCNT0

//Timer 0 overflow interrupt 3kHz, normal mode
.ORG 0x0;location for reset
JMP MAIN
.ORG 0x24;location for Timer0 oveflow (use Interrupt vector table for Atmega324PB)
JMP T0_OV_ISR //activity 2 use 0x1E

;----main program for initialization
.ORG 0x100
MAIN:LDI R20,HIGH(RAMEND)
OUT SPH,R20
LDI R20,LOW(RAMEND)
OUT SPL,R20
LDI R20,0B00000000
OUT DDRB, R20
LDI R20,$FF
OUT DDRA,R20
OUT PORTB,R20

LDI R20,0xFF
OUT DDRD, R20

//Part A starts, Timer 0 interrupt
LDI R20,(1<<TOIE0)
STS TIMSK0,R20 ;Timer0 mask register
SEI
LDI R20,-41   ;timer value
OUT TCNT0,R20
LDI R20,0x03  ;prescaler 64
OUT TCCR0B,R20

//MAIN OPERATION, forever loop
HERE:  IN     R20,PINB
              OUT    PORTA,R20
              JMP    HERE

//Interrupt Service Routine for Timer 0 Interrupt
T0_OV_ISR:
IN R16,PORTD
LDI R17,0xFF
EOR R16,R17
OUT PORTD,R16
```

```
LDI R20,-41  ;reset the timer value
OUT TCNT0,R20; load for next round

RETI
```

2. Result

- Calculations for timing signal.
- //Timer Period=1/3kHz=333.33usec, 1/16MHz=0.0625us from Stopwatch
- //tick=333.33us/0.0625us=5334us
- //For 50% Duty Cycle, 5334us/2=2667us
- //On for 333.33us/2=166.67us
- //Prescaler 2667/64=41.67=41 for TCNT0
- Screenshots showing your results need several to show the PD7 signal "on" and "off" times



☑ Automatically fit Y

Scroll the mouse-wheel while pressing and holding the left shift key to zoom in the time axis.

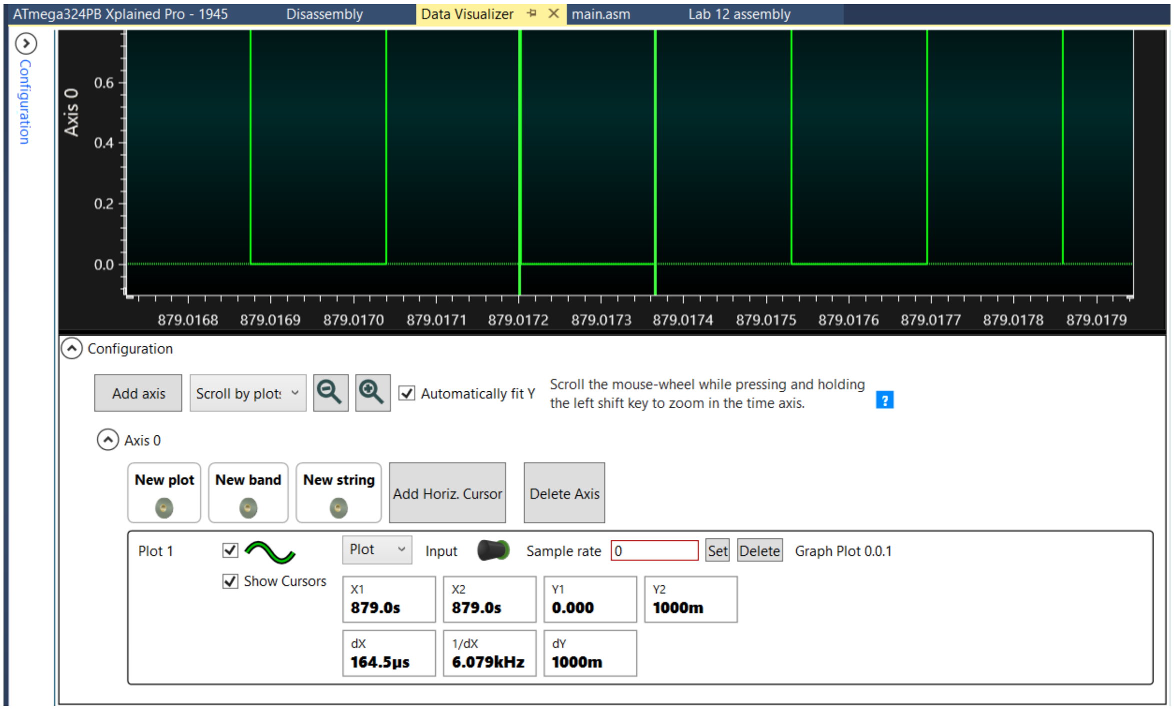
☑ Show Cursors

Sample rate 0

X1	X2	Y1	Y2
879.0s	879.0s	691.8m	1000m

dX	1/dX	dY
161.5µs	6.194kHz	308.2m

Figure 1. On Signal for 50% duty cycle



☑ Automatically fit Y

Scroll the mouse-wheel while pressing and holding the left shift key to zoom in the time axis.

☑ Show Cursors

Sample rate 0

X1 879.0s
X2 879.0s
Y1 0.000
Y2 1000m
dX 164.5µs
1/dX 6.079kHz
dY 1000m

Figure 2. Off Signal, 50% Duty Cycle

Activity 2

1. Code and Description

Code (Full-Comment):

2. Result

- Calculations for timing signal.

```
//Activity 2: Robert Bara
//Timer Period=1/1kHz=0.001usec, 1/16MHz=0.0625us
//tick=0.001us/0.0625us=62500us
//For 50% Duty Cycle, on signal=.001us/2=.0005us
//Prescale 62500/1024=61for TCINT1
```

- Screenshots showing your results need several to show the PD7 signal “on” and “off” times

Activity 3

1. Code and Description

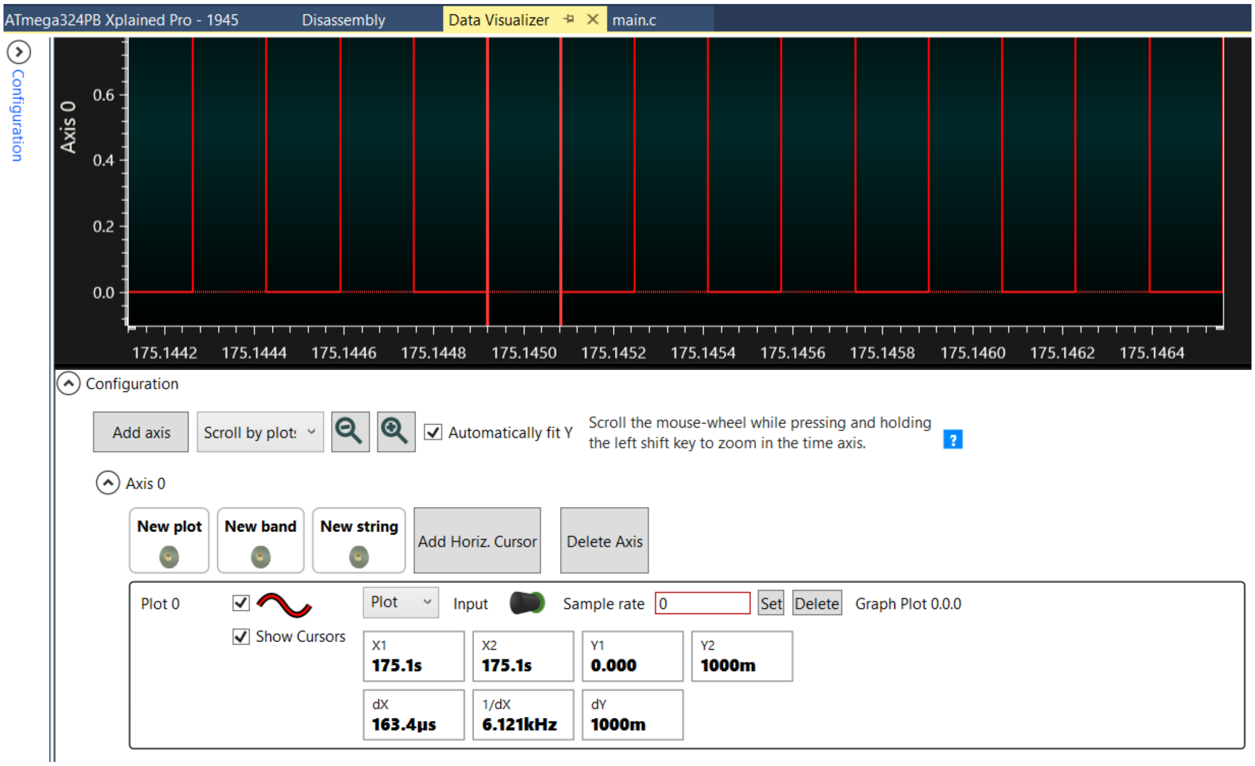
Code (Full-Comment):

```
//Activity 3 Robert Bara, it is activity 1 in C
#include "avr/io.h"
#include "avr/interrupt.h"

ISR (TIMER0_OVF_vect)       //ISR for Timer0 overflow
{
       TCNT0 = -41;         //timer value for .16ms
       PORTD ^= 0xFF;       //toggle PORTD
}
```

```
int main ()
{
        DDRD =0xFF;            //DDRD = output

        TCNT0 = -41;           //timer value
        TCCR0A = 0x00;
        TCCR0B = 0x03;         //Normal mode,  prescaler 64

        TIMSK0 = (1<<TOIE0); //enable Timer0 overflow interrupt
        sei ();                //enable interrupts

        DDRB = 0x00;           //Make portB input
        PORTB = 0xFF;          //Enable Pullup Resistors
        DDRA = 0xFF;           //Make portA output

        while (1)                     //wait here
        PORTA = PINB;
}
```

2. Result

- Screenshots showing your results need several to show the PD7 signal “on” and “off” times



☑ Automatically fit Y

Scroll the mouse-wheel while pressing and holding the left shift key to zoom in the time axis.

☑ Show Cursors

Sample rate 0

X1 175.1s

X2 175.1s

Y1 0.000

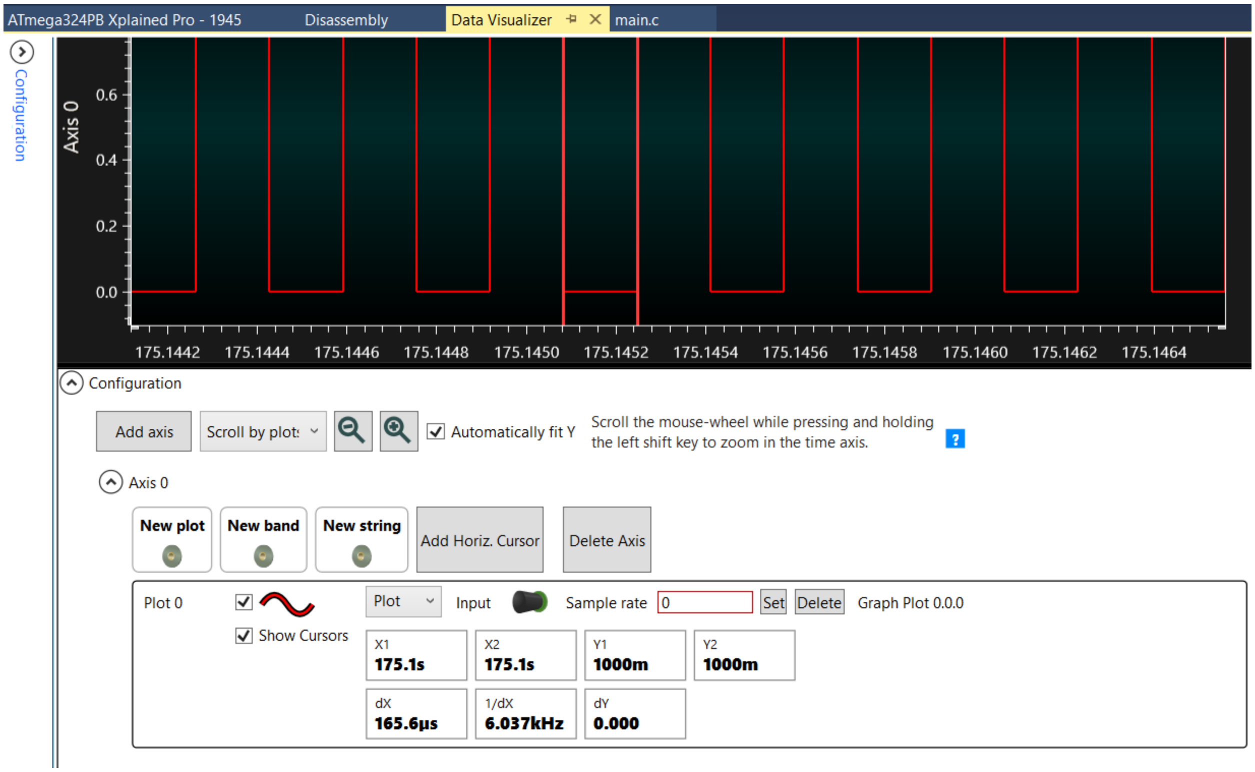
Y2 1000m

dX 163.4µs

1/dX 6.121kHz

dY 1000m

Figure 3. On Time for 50% Duty Cycle



☑ Automatically fit Y

☑ Show Cursors

Sample rate 0

Graph Plot 0.0.0

X1 175.1s | X2 175.1s | Y1 1000m | Y2 1000m

dX 165.6µs | 1/dX 6.037kHz | dY 0.000

Figure 4. Off for 50% Duty Cycle

Activity 4

1. Code and Description

Code (Full-Comment):

```
//Activity 4
#include <avr/io.h>
#include <avr/interrupt.h>
// Declare Global Variables
volatile int counter;
int main(void)
{
       counter = 0;// Initialize counter
       DDRA = 0xFF;//Port A has 8 Leds as outputs
       DDRD = 0;// Port D all inputs
       PORTD |= (1<<2);// Set Pull up resistor PD2 (INT0)
       EIMSK=(1<<INT0);
       EICRA=0X02; //binary 0b10 INT0 WILL ACTIVATE Falling EDGE TRIGGER
       sei();// Set Interrupt Flag
while(1)
{
       // Wait For Interrupts and Update LEDs
       PORTA = counter;
}
}
// External Interrupt on Port D Pin 2
ISR(INT0_vect, ISR_BLOCK)
{
        counter++;// Increment counter when pin changes
    }
```

2. Result

- Result Video Link (YouTube video of the full operation)
- https://www.youtube.com/watch?v=OPsf_I5M5ZA&ab_channel=RobertBara

Activity 5

1. Code and Description

Code (Full-Comment):

```
#define F_CPU 16000000UL
#include <util/delay.h>
#include <avr/interrupt.h>
#include "SSD1306.h"
//#include "SSD1306.c" //Not sure why I need to include this file here
// Declare Global Variables
volatile int counter=0;
int main(void)
{

        //CONFIGURATION AND SET INTERRUPT
        DDRD = 0;// Port D all inputs
        PORTA=0xff; //initialize LED ports
        PORTD  |= (1<<2)|(1<<3);// Set Pull up resistor PD2 (INT0) and PD3 (INT1)
        EIMSK=(1<<INT0)|(1<<INT1); //unmask INT0 and INT1
        EICRA=0X0E; //INT0 and INT1 WILL ACTIVATE Falling EDGE TRIGGER
        sei();// Set Interrupt Flag

        //OLED Display
        OLED_Init();  //initialize the OLED
        _delay_ms(1);
        OLED_Clear(); //clear the display (for good measure)
        OLED_SetCursor(0, 0);        //set the cursor position to (0, 0)
        OLED_Printf("Count UP PD2-PB2:   \n"); //Print out some text
        OLED_Printf("Count DW PD2-PB3:   ");
        OLED_SetCursor(4, 0);
        OLED_Printf("PortA:   ");
        OLED_SetCursor(6, 0);
        OLED_Printf("Robert Bara   ");

        while(1)
        {
                // Wait For Interrupts and Update LEDs
                        if(counter>10)
                        {
                                OLED_SetCursor(5, 0);
                                OLED_DisplayNumber(C_DECIMAL_U8,counter,3);
                                _delay_ms(1000);
                                PORTA=0b1000000;
                        }
                        else
                        {
                                OLED_SetCursor(5, 0);
                                OLED_DisplayNumber(C_DECIMAL_U8,counter,3);
                _delay_ms(1000);
                PORTA=0;
            }

    }
}
// External Interrupt on Port D Pin 2
ISR(INT0_vect, ISR_BLOCK)
{
    counter++;// Increment counter when pin changes
}
// External Interrupt on Port D Pin 3
ISR(INT1_vect, ISR_BLOCK)
{
    counter--;// Decrement counter when pin changes
}
```

2. Result

- Result Video Link (YouTube video of the full operation)

https://www.youtube.com/watch?v=OPsf_I5M5ZA&ab_channel=RobertBara (1:23)

ECE3613 Processor System Laboratory Rubric
Lab #: 12 Section: 001 / 002
Name: ________________________

Activity	Contents	Full Points	Earned Points	Comment
1	Code	10		• Code with full comments
	Result	10		• Calculations for timing signal. • Screenshots showing your results need several to show the PD7 signal "on" and "off" times.
2	Code	10		• Code with full comments
	Result	10		• Calculations for timing signal. • Screenshots showing your results need several to show the PD7 signal "on" and "off" times.
3	Code	10		• Code with full comments
	Result	10		• Screenshots showing your results need several to show the PD7 signal "on" and "off" times.
4	Code	10		• Code with full comments
	Result	10		• YouTube video of the full operation
5	Code	10		• Code with full comments
	Result	10		• YouTube video of the full operation
Total		100		