Microprocessor Systems and Interfacing

Lab Report

Lab06

Group Members Name & Reg #:	**Muhammad Haris Irfan** **(FA18-BCE-090)**
Class	Microprocessor Systems and Interfacing CPE342 (**BCE-6B**)
Instructor's Name	Dr. Omer Ahmad

Pre-Lab Tasks

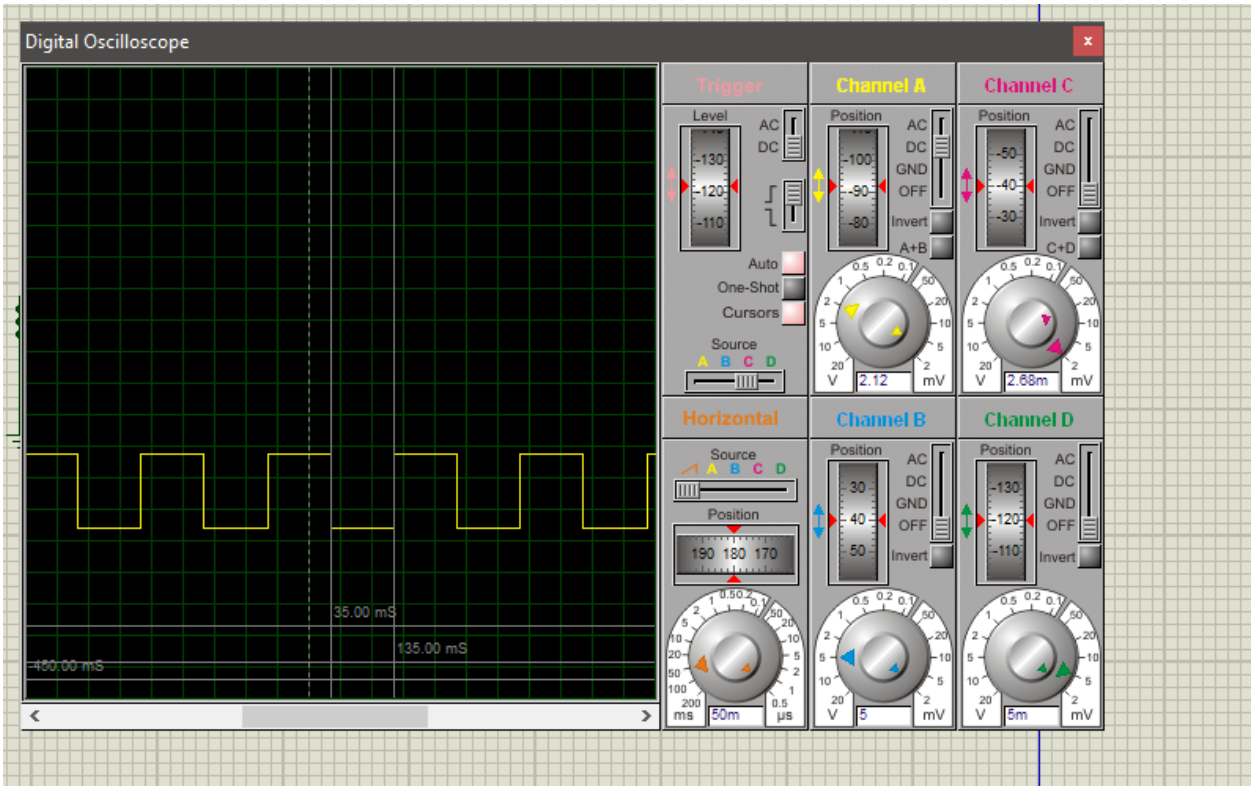
Task-1

Consider the following C code. Your task is to write an AVR Assembly program with the same functionality.

Code

```
ldi R18,0xFF
OUT DDRB, R18

start:

sbi PORTB, 3
rcall delay_ms_100

cbi PORTB,3
rcall delay_ms_100

jmp start



delay_ms:

ldi r20,8

a:
ldi r17,250
loop:
nop
dec r17
brne loop

dec r20
brne a
ret



delay_ms_100:
ldi r22,100

loop1:
rcall delay_ms
dec r22

brne loop1
ret
```

Simulation:


Digital Oscilloscope
Trigger
Level
AC
DC
Auto
One-Shot
Cursors
Source
A B C D
Channel A
Position
AC
DC
GND
OFF
Invert
A+B
2.12
mV
Channel C
Position
C+D
2.68m
mV
Horizontal
Source
Position
190 180 170
50m
Channel B
Position
5
mV
Channel D
Position
5m
mV
35.00 mS
135.00 mS
-150.00 mS

In Lab Tasks

Task 1:

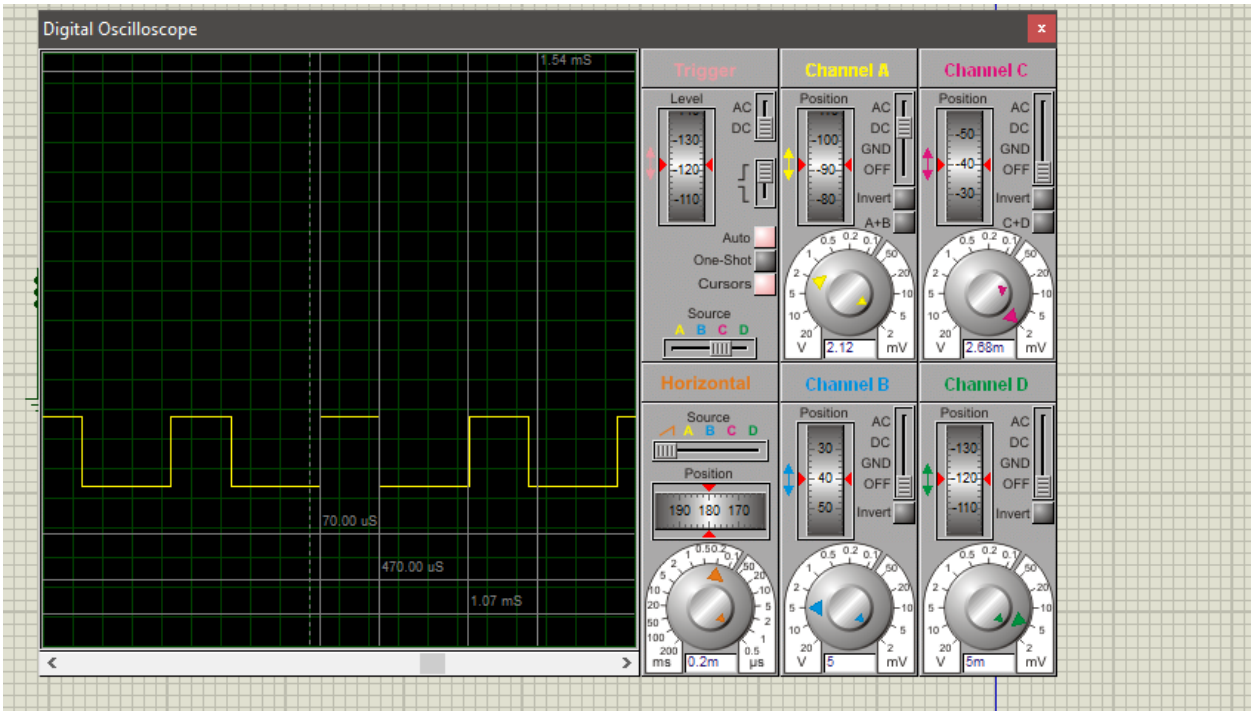
Generate a square wave signal of frequency 1 KHz and 40% duty cycle on a digital I/O pin of ATmega328P.

Code:

```
;
; prelabTask6.asm
;
; Created: 10-Apr-21 11:10:41 PM
; Author : HP
;
; Replace with your application code


ldi R18,0xFF
OUT DDRB, R18

start:

sbi PORTB, 3
rcall find_delay:


cbi PORTB,3

rcall delay_us_600

jmp start



find delay:

ldi r16, 0x00
cp r16, PORTB
breq




delay_us_400:

ldi r23, 5
a:
```

```
ldi r22,200

loop2:
nop
dec r22
brne loop2

dec r23
brne a

ret

delay_us_600:
ldi r23,6
b:
ldi r22,200

loop1:
nop
dec r22
brne loop1

dec r23
brne b
ret
```

Simulation:



Task 2:

Generate a square wave signal of frequency 1 KHz and 40% duty cycle on a digital I/O pin of ATmega328P.

Code:

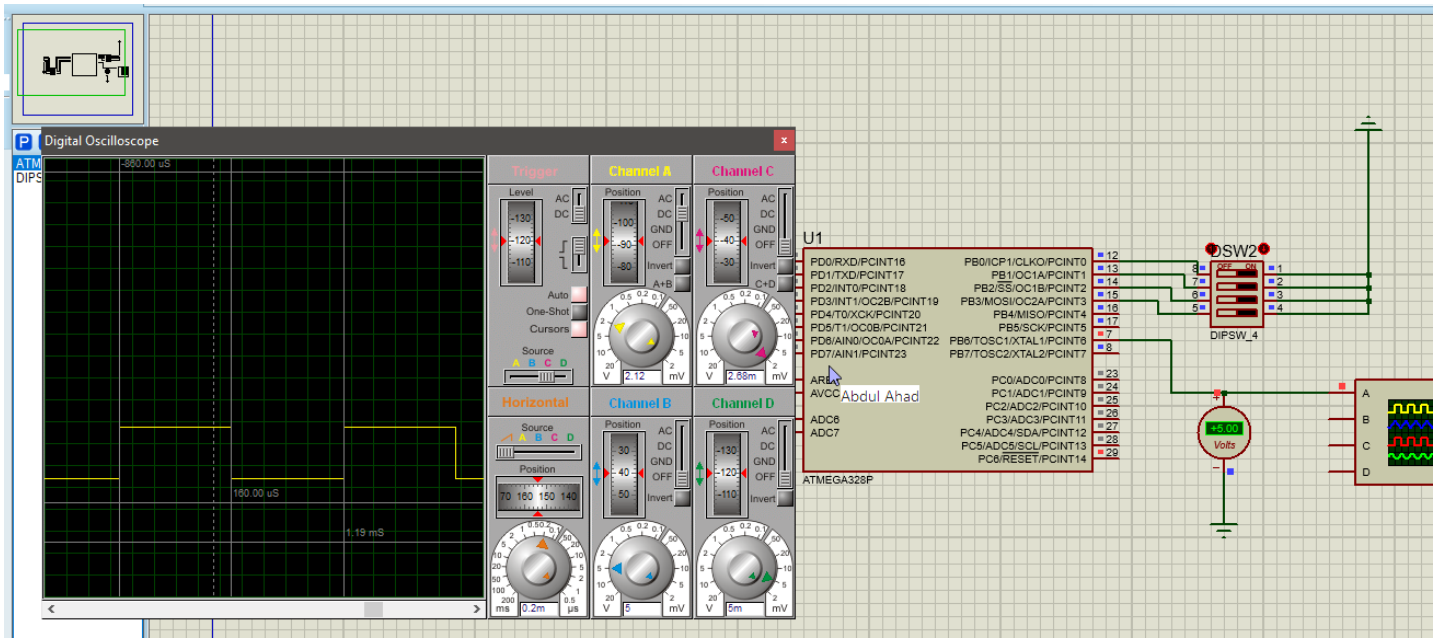
```
ldi r18,0xF0
OUT DDRB,r18

ldi r18, 0x0F  // input
OUT PORTB,r18 // pull up


start:

in r20,PINB
ANDI r20,0x0F


sbi PORTB, 6
rcall _delayFind
cbi PORTB,6
rcall _delayFind

jmp start



_delayFind:

ldi r21, 21
cpi r20, 15
breq loop


ldi r21, 22
cpi r20, 14
breq loop


ldi r21, 23
cpi r20, 13
breq loop


ldi r21, 26
cpi r20, 12
breq loop
```

```
//////////////////

ldi r21, 28
cpi r20, 11
breq loop


ldi r21, 30
cpi r20, 10
breq loop


ldi r21, 34
cpi r20, 9
breq loop


ldi r21, 37
cpi r20, 8
breq loop

////////////////////////////////////



ldi r21, 42
cpi r20, 7
breq loop


ldi r21, 48
cpi r20, 6
breq loop


ldi r21, 56
cpi r20, 5
breq loop


ldi r21, 68
cpi r20, 4
breq loop

//////////////////////////////////////////


ldi r21, 84
cpi r20, 3
breq loop
//////////////////////////////

ldi r21, 114
cpi r20, 2
breq loop
////////////////////////////
```

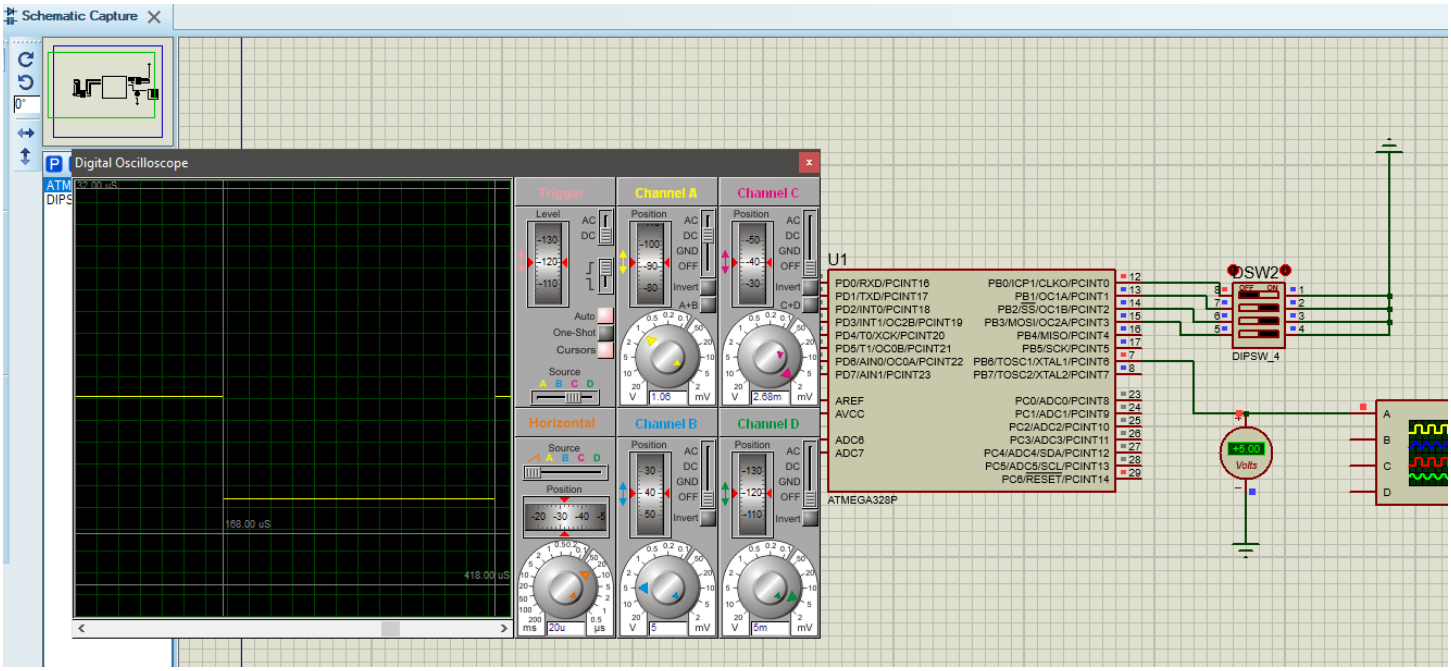
```
ldi r21, 177
cpi r20, 1
breq loop
////////////////////////////


ldi r21, 125
cpi r20, 0
breq forone
////////////////////////////


loop:

rcall delay_1us
dec r21
brne loop
ret



delay_1us:
       nop

ret

forone:
ldi r16, 18
rcall delay_1us


loopaa:


dec r16
brne loopaa
dec r21
brne forone

ret
```

Simulation:

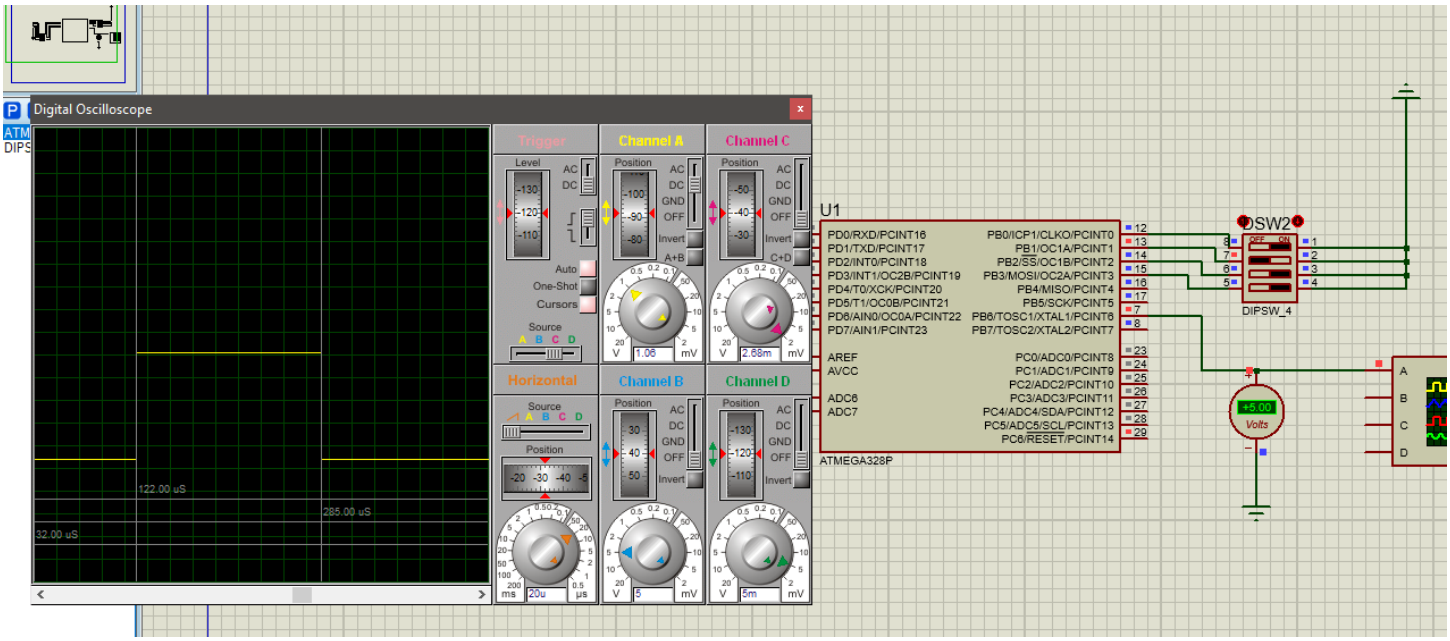
- **For 1KHz:**



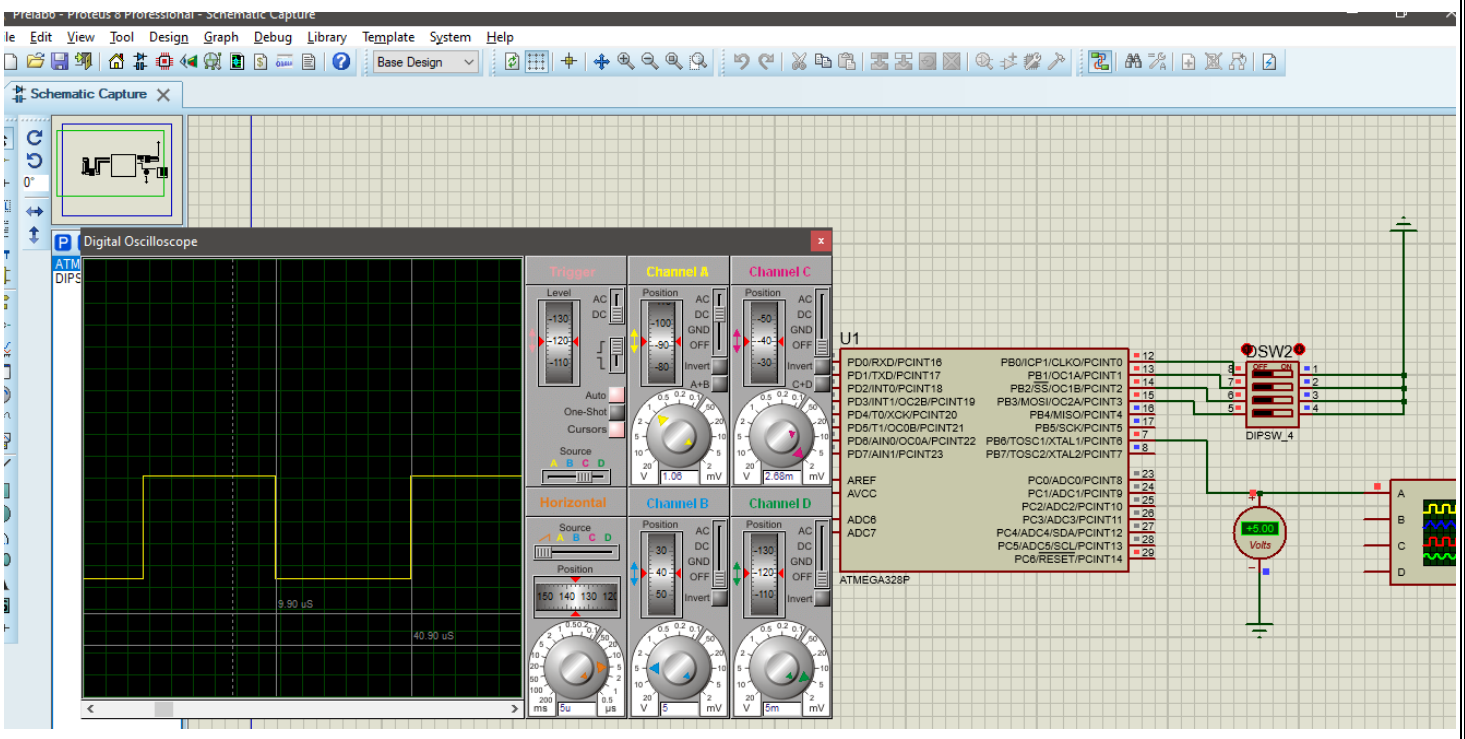
- **For 2KHz:**



- **For 3KHz:**



- **For 16KHz:**



Post Lab Tasks

Task 1:

Write a report explaining the problems faced in implementing the in-lab tasks and provide their solutions.

Answer:

Initially I tried making delays for each signal, but that code was very lengthy, so I made a generic code with a single delay that was called multiple times, the main problem that I faced was to make accurate delay for each frequency signal as my delay was of 31us at start, to make my code accurate I used 1us delay.

Nevertheless, after many calculations, my signal waves got accurate.