CPE301 – SPRING 2020

Design Assignment 2A

Student Name: Xianjie Cao

Student #: 5004222179

Student Email: caox2@unlv.nevada.edu

Primary Github address: <https://github.com/c1029324620/Mocha.git>

Directory: Mocha/DesignAssignments/LAB2/DA2A

The goal of the assignment is use GPIO and delays:

1. Design a delay subroutine to generate a waveform on PORTB.3 with 55% DC and 0.75

sec period.

2. Connect a switch to PORTC.3 (active high - turn on the pull up transistor) to poll for an

event to turn on the led at PORTB.2 for 2 sec after the event.

1. **COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS**

Atmel Studio 7: Simulator, debugger and Assembler.

Atmega328PB-Xmini PC

Arduino Multi-function shield: LED, switch

Logic Analyzer

1. **INITIAL/MODIFIED/DEVELOPED CODE OF TASK 1/A**

**Assembly Code for Task 1:**

;

; DA2AT1.asm

;

; Created: 2/21/2020 11:50:06 PM

; Author : c1029

;

; design a delay subroutine to generate waveform on PORTB.3 with 55% Duty Cycle and the period is 0.75s.

; Replace with your application code

start:

ldi r16, 0xff ;r16 = 0xff, it will be used to set DDRB as output

out DDRB, r16 ;set DDRB as output

again:

sbi PORTB, 3 ;set bit 3 in PORTB high.

rjmp delay\_high ;jump to delay subroutine, should wait around 0.4s

high\_delay\_done:

cbi PORTB, 3 ;after 0.41 sec delay, make PORTB.3 Low

rjmp delay\_low

low\_delay\_done:

rjmp again ;after 0,33 sec delay, set PORTB.3 High again, then we have generated a waveform.

; 0.4125s delay if it runs at 16 MHz

delay\_high:

ldi r18, 34

ldi r19, 124

ldi r20, 86

L1: dec r20

brne L1

dec r19

brne L1

dec r18

brne L1

rjmp high\_delay\_done

; 0.3375 delay if it runs at 16 MHz

delay\_low:

ldi r18, 28

ldi r19, 101

ldi r20, 233

L2: dec r20

brne L2

dec r19

brne L2

dec r18

brne L2

rjmp low\_delay\_done

done:

rjmp done

**C code for Task 1:**

/\*

\* DA2AT1\_C.c

\*

\* Created: 2/22/2020 1:02:46 AM

\* Author : c1029

\*/

#define *F\_CPU* 16000000UL

#include <avr/io.h>

#include <util/delay.h>

int main(void)

{

DDRB = 0xff; //set PORTB as output

/\* Replace with your application code \*/

while (1)

{

PORTB |= (1 << 3); //set bit 3 in PORTB to HIGH.

*\_delay\_ms*(413); //wait for 0.4125 sec.

PORTB &= ~(1 << 3); //set bit 3 in PORTB to LOW.

*\_delay\_ms*(338); //wait for delay of 0.3375 sec

}

}

1. **DEVELOPED MODIFIED CODE OF TASK 2/A from TASK 1/A**

**Assembly Code for Task 2**：

;

; DA2AT2.asm

;

; Created: 2/24/2020 8:02:35 PM

; Author : c1029

;

; connect a switch to PORTC.3 to poll for an event to turn on the LED at PORTB.2 for 2 sec.

; Replace with your application code

start:

cbi DDRC, 3 ;set PORTC.3 as input

ldi r16, 0x04;set PORTB as output

out DDRB, r16 ;set PORTB.2 as output

again:

sbis PINC, 3 ;becasue the push button is active high, if the button is push, then will set it low

rjmp delay\_2s ;jump to delay and will turn LED on

sbi PORTB, 2 ;set bit 2 in PORTB high will set LED off

rjmp again

delay\_2s:

cbi PORTB, 2 ;this will turn LED on

ldi r18, 163

ldi r19, 87

ldi r20, 3

L1: dec r20

brne L1

dec r19

brne L1

dec r18

brne L1

rjmp again

**C code for Task 2:**

/\*

\* DA2AT2\_C.c

\*

\* Created: 2/24/2020 10:07:04 PM

\* Author : c1029

\*/

#define *F\_CPU* 16000000UL

#include <avr/io.h>

#include <util/delay.h>

int main(void)

{

DDRB |= (1 << 2); //set PORTB.2 as output.

PORTB |= (1 << 2);

DDRC &= (0 << 3); //set PORTC.3 as input.

PORTC |= (1 << 3); //enable pull-up

/\* Replace with your application code \*/

while (1)

{

if(!(PINC & (1 << PINC3))) // if PINC.3 is pressed.

{

PORTB &= ~(1 << 2);

*\_delay\_ms*(2000);

}

else

{

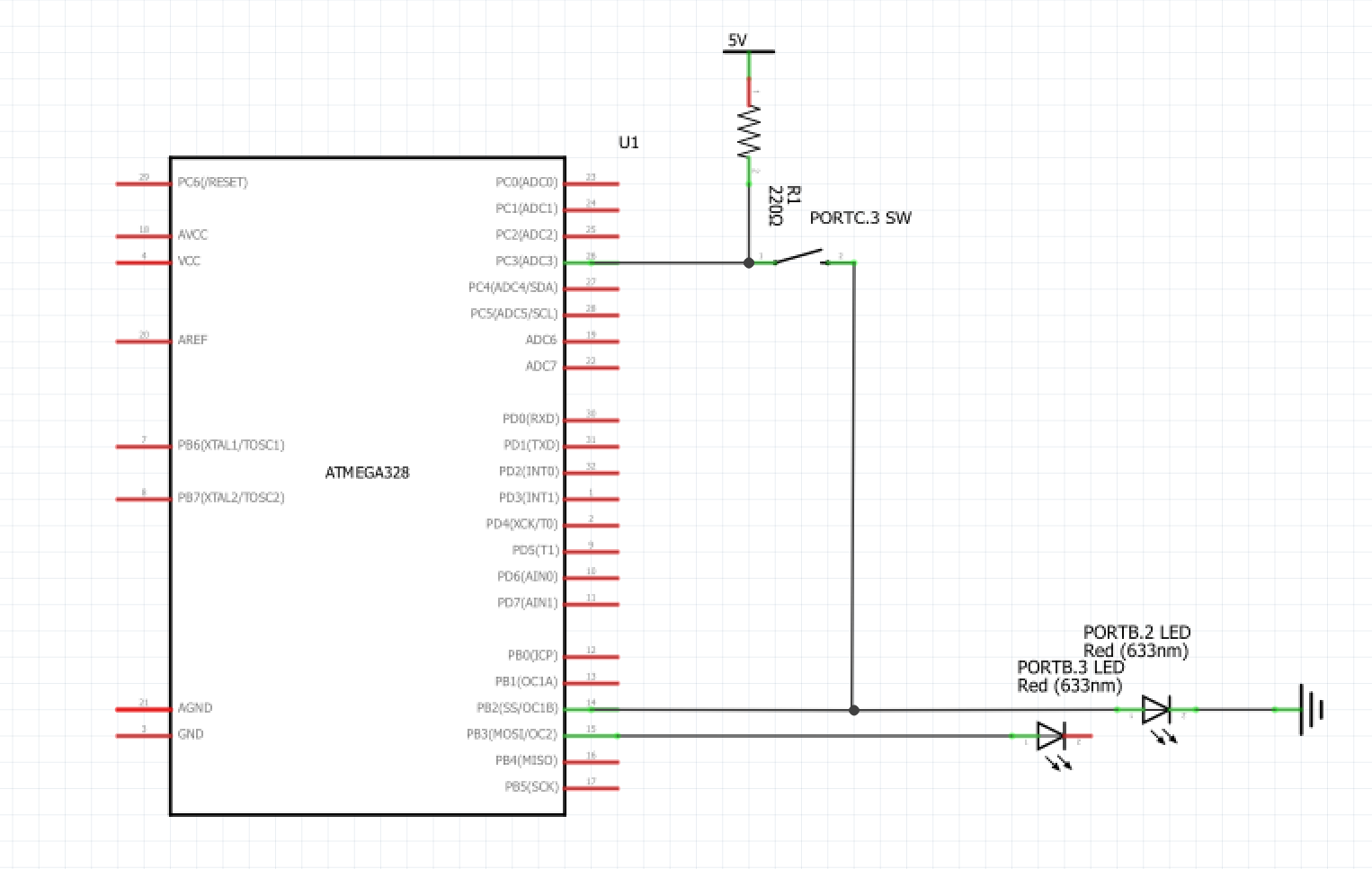
PORTB |= (1 << 2);

}

}

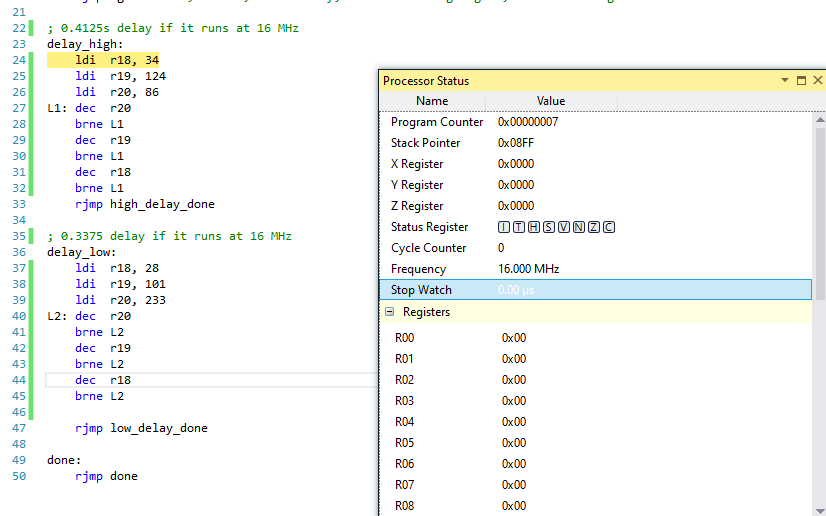
}

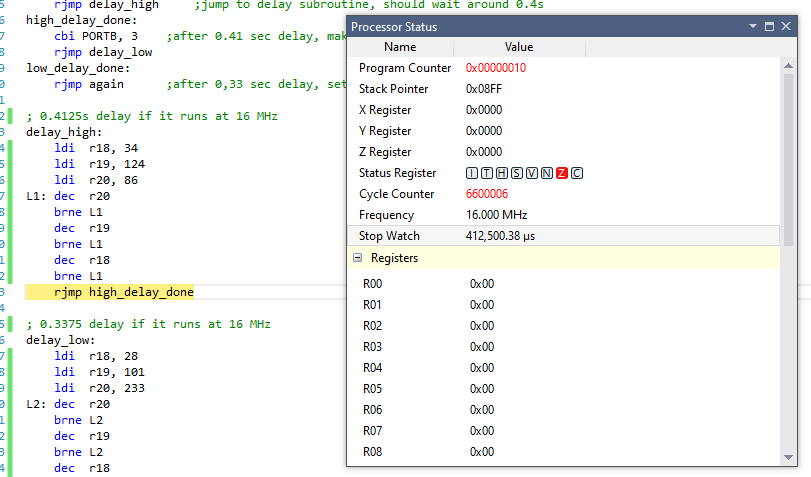
1. **SCHEMATICS**



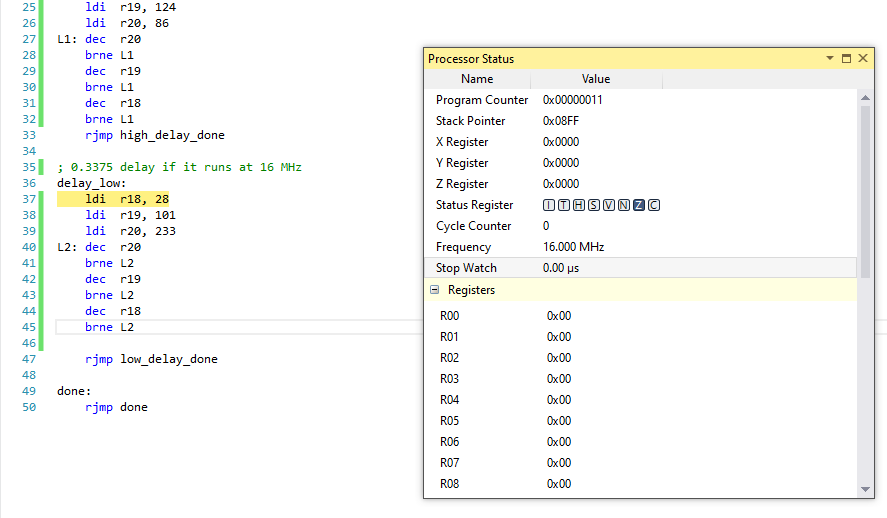
1. **SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)**

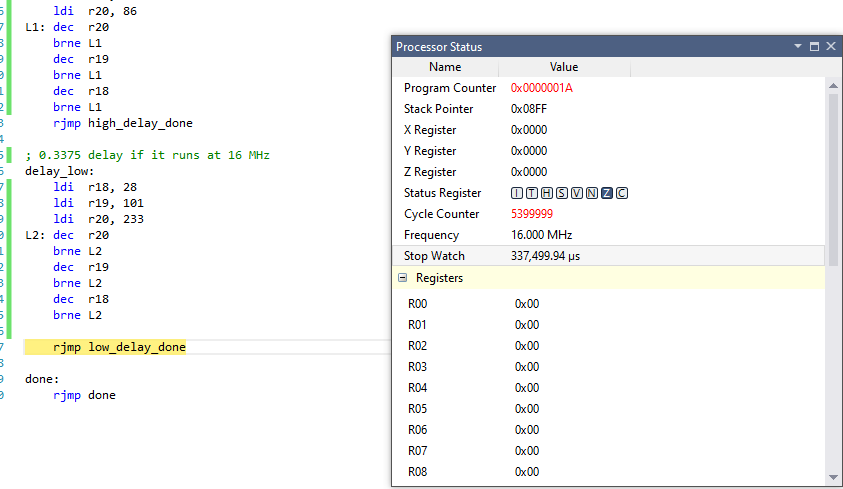
**Delay of 0.415 sec:**



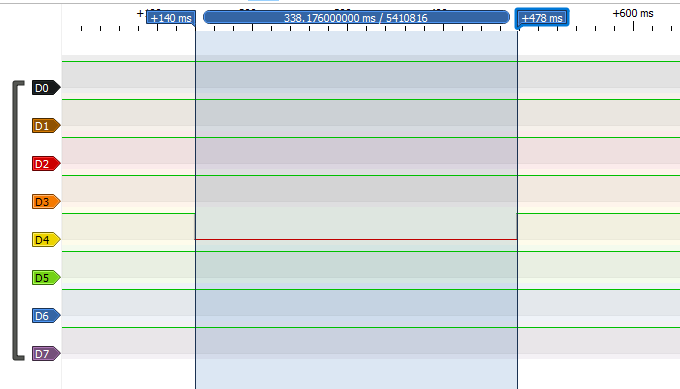


Delay of 0.3375 sec:

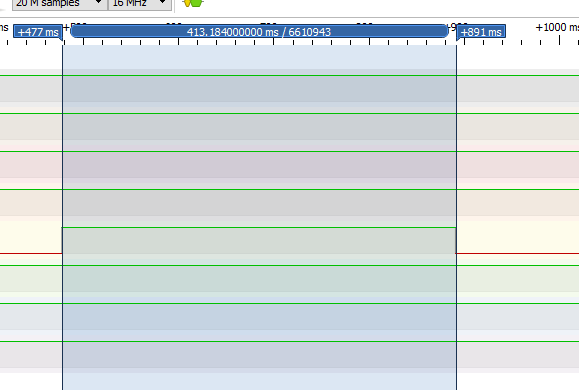




The Waveform generated on PlusView:

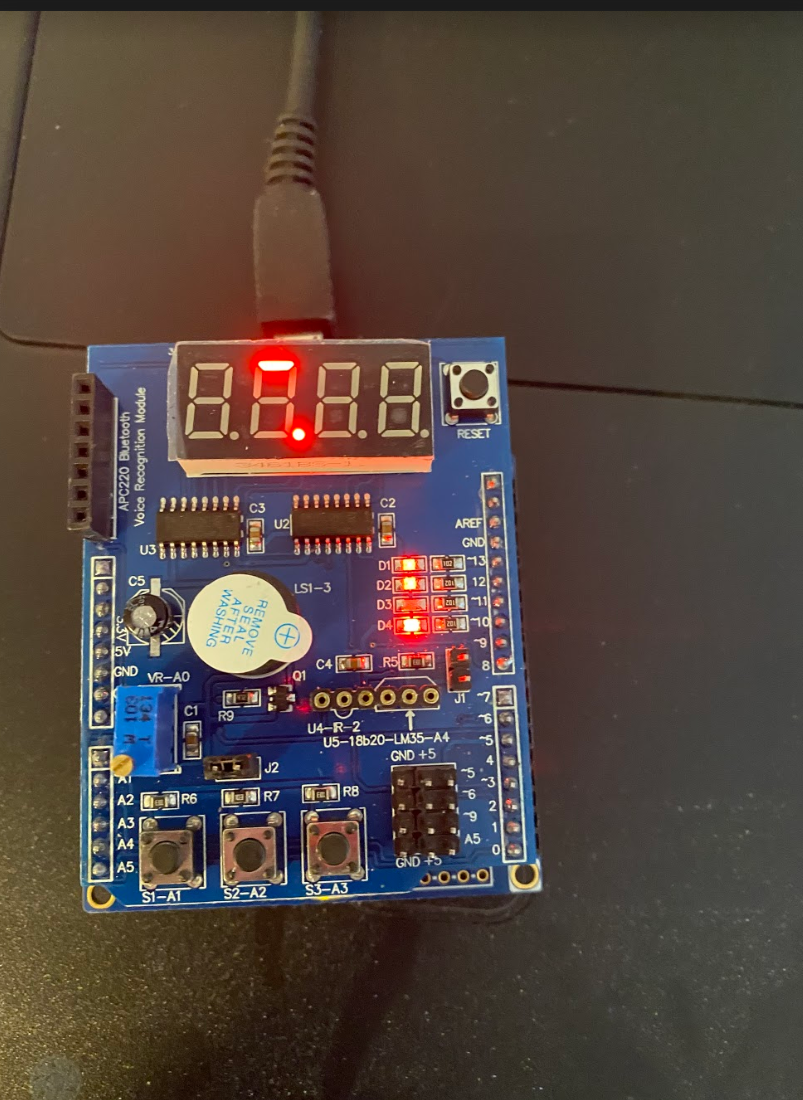
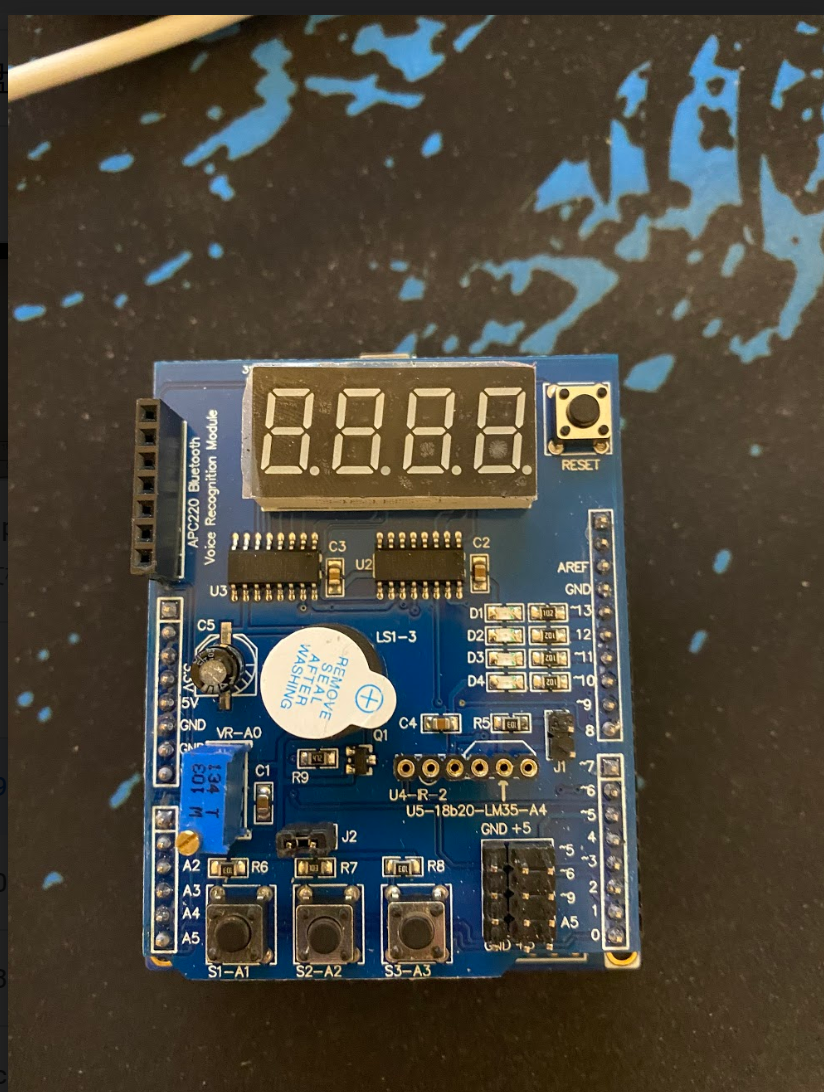


OFF: 478ms – 140ms = 338ms



ON: 891ms – 477ms = 414ms

1. **SCREENSHOT OF EACH DEMO (BOARD SETUP)**



1. **VIDEO LINKS OF EACH DEMO**

Assembly Demo:

<https://youtu.be/VFOdS4pv3yg>

C Demo:

<https://youtu.be/8GLdofv3Kvw>

1. **GITHUB LINK OF THIS DA**

<https://github.com/c1029324620/Mocha/tree/master/DesignAssignments/LAB2/DA2A>

**Student Academic Misconduct Policy**

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

“This assignment submission is my own, original work”.

Xianjie Cao