CPE301 – SPRING 2019

Design Assignment 2A

Student Name: Victor Martinez

Student #:5003131674

Student Email: martiv6@unlv.nevada.edu

Primary Github address: https://github.com/martiv6

Directory: https://github.com/martiv6/submissions\_da/tree/master/DesignAssignment/DA2

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**

TASK 1

LED

SHIELD

328P

Mini

POWERSUPPLY

TASK 2

LED

SWITCH

SHIELD

328P

Mini

POWERSUPPLY

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

;

; 2A\_task\_1.asm

;

; Created: 3/1/2019 8:18:16 PM

; Author : victor

;

.org 0

LDI r16, 0b0111 ; setting pd2

OUT DDRB, r16 ; enableing pd2 as output

LDI r17, 0b0000 ; used to set/reset pd5

LDI r20, 0b0101 ; setting up TCCR1B = 00000101 for 1024

STS TCCR1B, r20 ; storing the value in TCCR1B

INTRO:

LDI r20, 0b0000 ; setting up timer counter to 0

STS TCNT1H, r20 ; Timer counter is 16 bits

STS TCNT1L, r20 ; Thus, requires 2 8 bit regs

RJMP SLOWDOWN ; jumps back to SLOWDOWN

COMEBACK:

EOR r17, r16 ; xor to toggle LED

OUT PORTB, r17 ; enable pd5

LDI r20, 0b0000 ; setting up timer counter to 0

STS TCNT1H, r20 ; Timer counter is 16 bits

STS TCNT1L, r20 ; Thus, requires 2 8 bit regs

RJMP FINAL\_STOP ; Jumps to FINAL\_STOP

COMEBACK\_DOS:

EOR r17, r16 ; xor to toggle LED

OUT PORTB, r17 ; enable pd5

RJMP INTRO ; repeat main loop

SHOWDOWN\_1:

CPI r29, 0x11 ; check if upper timer counter have reached desired value

BRLT SLOWDOWN ; otherwise recheck the upper bytes

RJMP COMEBACK ; jumps to COMEBACK

SLOWDOWN:

LDS r29, TCNT1H ; load upper bytes of timer counter to r29

LDS r28, TCNT1L ; load lower bytes of timer counter to r28

CPI r28, 0xB2 ; check to see if lower 8 bits of timer counter

BRSH SHOWDOWN\_1 ; if lower bit not

RJMP SLOWDOWN ; otherwise keep checking lower bytes

SHOWDOWN\_2:

CPI r29, 0x1A ; check if upper timer counter have reached desired value

BRLT FINAL\_STOP ; otherwise recheck the lower bytes

RJMP COMEBACK\_DOS ; jumps to COMEBACK\_DOS

FINAL\_STOP:

LDS r29, TCNT1H ; load upper bytes of timer counter to r29

LDS r28, TCNT1L ; load lower bytes of timer counter to r28

CPI r28, 0x8C ; check to see if lower 8 bits of timer counter are 0x08

BRSH SHOWDOWN\_2 ; keep checking lower bytes

RJMP FINAL\_STOP ; jumps to FINAL\_STOP

/\*

\* 2A\_Task\_1\_C.c

\*

\* Created: 3/1/2019 12:09:53 AM

\* Author : victor

\*/

#include <avr/io.h>

#define *F\_CPU* 16000000UL

#include <util/delay.h>

int main(void)

{

DDRB = 0b0100; //makes pd2 an output

while (1)

{

PORTB = (1 << PORTB2); //turns on the led we want

*\_delay\_ms*(435); //time the led will be off to get the 60% DC with the period of 0.725

PORTB= ~(1 << PORTB2); //will turn on led

*\_delay\_ms*(290); //the time the led will be on to get the 40% DC with the period of 0.725

}

return 0;

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

;

; 2A\_Task2\_asm.asm

;

; Created: 3/2/2019 12:20:24 PM

; Author : victor

;

.org 0

SBI DDRB, 0b0010 ; makes PB2 the output

CBI DDRC, 0b0000 ; makes A1/S1 the button to push

SBI PORTB, 0b0010 ; makes PB2 turn off

THEWAIT:

SBIC PINC, 0b0001 ; if pressed do this

RJMP THEWAIT ; if not pressed jump back to THEWAIT

RCALL THEPUSH ; keeps led to 1.25s

SLOWDOWN:

LDS R29, TCNT1H ; load upper bytes of timer counter to r29

LDS R28, TCNT1L ; load lower bytes of timer counter to r28

CPI R28, 0x4A ; check to see if lower 8 bits of timer counter are valid

BRSH WELP ; if same or higher if not goes back to welp

RJMP SLOWDOWN ; otherwise keep checking lower bytes

THEPUSH:

CBI PORTB, 0b0010 ; if pressed it will light up the led

LDI R20, 0b0101 ; set the prescalar to 1024

STS TCCR1B, R20 ; loads value to R20

LDI R20, 0b0000 ; resets timer

STS TCNT1H, R20 ; clears value in TCNT1H

STS TCNT1L, R20 ; clears value in TCNT1L

RJMP SLOWDOWN ; jumos down to SLOWDOWN to start again

COMINGHOME:

RJMP THEWAIT ; goes to back to see in theres an input

SBI PORTB, 0b0010 ; if on display on LED

RJMP THEWAIT ; goes back to see if there is an input

WELP:

CPI R29, 0x4C ; check if upper timer counter have reached desired value

BRLT SLOWDOWN ; otherwise recheck the lower bytes

SBI PORTB, 0b0010 ; TURN OFF LED

RET

/\*

\* 2A\_Task\_2\_C.c

\*

\* Created: 3/2/2019 6:16:17 PM

\* Author : victor

\*/

#include <avr/io.h>

#define *F\_CPU* 16000000UL

#include <util/delay.h>

int main(void)

{

DDRB = 0b0100; // set PB2 as output when pressed

PORTB = (0b0001<<0b0101); // turn off PB5 LED

DDRC = (0b0000<<0b0001); // reads data entered

PORTC = (0b0001<<0b0001); // will cause it to light up and display

while (1)

{

if(PINC & (PINC1<<PINC1)) //to see if the switch is pressed

{

PORTB = (PORTB1<<PORTB2); // if there is no input keep led off

}

else

{

PORTB &= ~(PORTB1<<PORTB2); // toggle the led

*\_delay\_ms*(1250); // keep the light on for 1.25 SECONDS

}

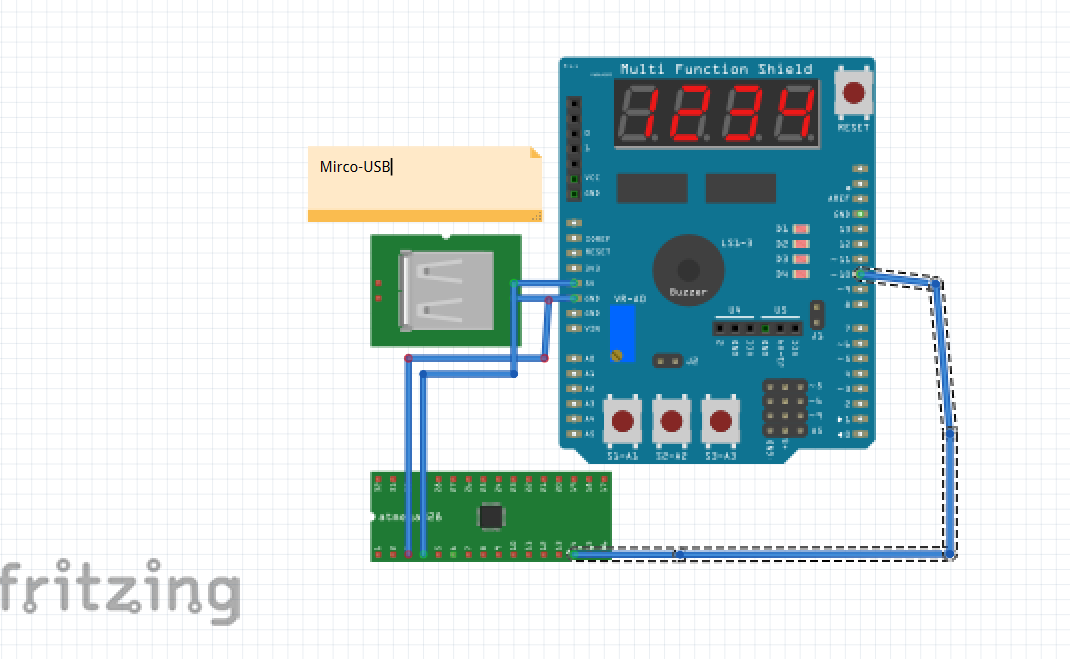
}

return 0;

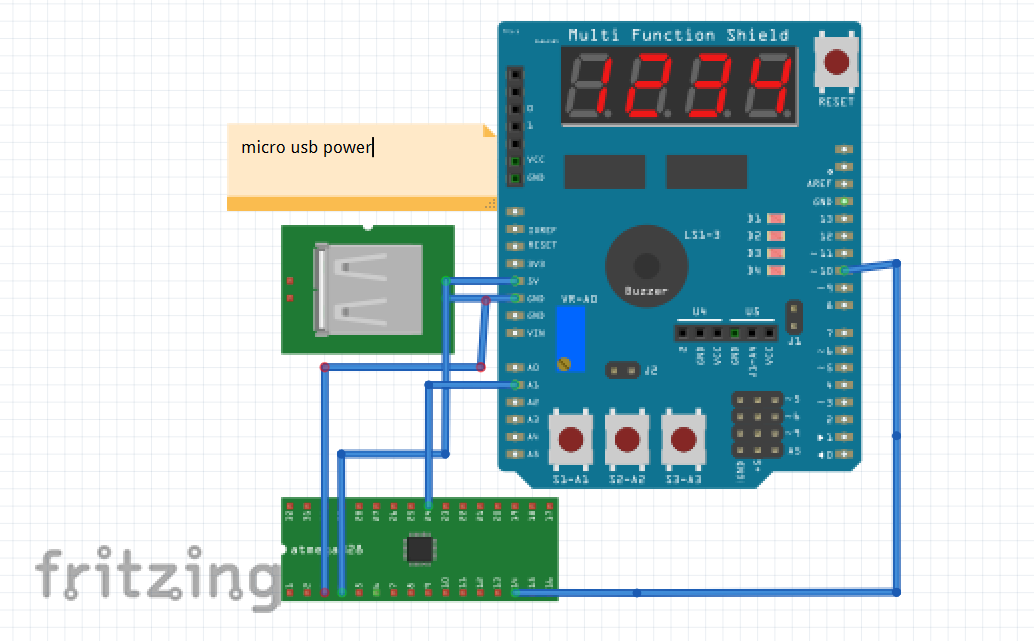
}

1. **SCHEMATICS**

TASK 1.



TASK 2.



1. **SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)**
2. **SCREENSHOT OF EACH DEMO (BOARD SETUP)**

TASK 1&2



1. **VIDEO LINKS OF EACH DEMO**

TASK 1 – <https://youtu.be/C1VlHoTSl8A>

C- <https://youtu.be/LN9se80oZAE>

TASK 2 - <https://youtu.be/xiePFdgSiS8>

C- <https://youtu.be/eeDtTQRLvYc>

1. **GITHUB LINK OF THIS DA**

https://github.com/martiv6/submissions\_da/tree/master/DesignAssignment/DA2

**Student Academic Misconduct Policy**

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

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

Victor Martinez