

Design Assignment 2

Student Name: Abraham Garcia

Student #: 5005262049

Student Email: garci11@unlv.nevada.edu

Primary Github address: https://github.com/SON-Abe/submission_da.git

Directory: C:/Users/abrah/OneDrive/Desktop/General/Uni/2024/CPE 301/Design_Assignments

Video Playlist: [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

- Microchip Studio Debugger
- Microchip Studio Simulator
- Male-to-Male Wire
- Female-to-Male Wire
- ATmega328PB Microcontroller
- Logic Analyzer
- Saleae Logic Analyzer Software

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

```
// DA2_2C.c
// Created: 3/10/2024 12:07:15 AM
// Author : SON

#define F_CPU 16000000UL      //16MHz clock
#include <avr/io.h>          //AVR io header file
#include <util/delay.h>        //delay header file

int main(void)                //main
{
    DDRB |= (1<<5);           //set PB5 as output
    PORTB &= ~(1<<5);         //set PB5 to '0'
    DDRC &= ~(1<<2);         //set PC2 as input
    PORTC |= (1<<2);          //enable pull-up resistor
    while(1)                   //forever loop
    {
        if (!(PINC & (1<<2))) //if PC2 is GND
        {
            PORTB |= (1<<5);   //turn on PB5
            _delay_ms(2000);     //for 2000ms
            PORTB &= ~(1<<5);  //turn off PB5
            _delay_ms(2000);     //for 2000ms
        }
    }
}

; DA2_2A.asm
; Created: 3/10/2024 2:23:43 AM
; Author : SON
```

```

#define F_CPU 16000000UL ;16MHz clock
.include "m328pbdef.inc" ;m328pb device header file

.org 0x00                                ;set program counter the start of the code
    jmp main                             ;jump to main

main:
    cbi DDRC, 2                         ;set PC2 as input
    sbi DDRB, 5                         ;set PB5 as output
    cbi PORTB, 5                        ;enable pull-up resistor

loop:
    sbic PINC, 2                       ;skip next lin if PINC 2 clear
    jmp loop                            ;loop
    sbi PORTB, 5                        ;turn on PB5
    call delay                           ;for 2 seconds
    cbi PORTB, 5                        ;turn off PB5
    call delay                           ;for 2 seconds
    jmp loop                            ;loop

delay:
    ldi r18, 163                        ;loaded value to help delay for 2000ms
    ldi r19, 87                          ;loaded value to help delay for 2000ms
    ldi r20, 3                           ;loaded value to help delay for 2000ms
L1: dec r20                            ;dec value to help delay for 2000ms
    brne L1                            ;loop till 0
    dec r19                            ;dec value to help delay for 2000ms
    brne L1                            ;loop till 0
    dec r18                            ;dec value to help delay for 2000ms
    brne L1                            ;loop till 0
    nop
ret                                     ;return when done with delay

```

3. DEVELOPED/MODIFIED CODE OF TASK 3/A

```

// DA2_3C.c
// Created: 3/10/2024 4:27:09 AM
// Author : SON

#define F_CPU 8000000UL                  //8MHz clock
#include <avr/io.h>                   //AVR io header file
#include <util/delay.h>                //delay header file
#include <avr/interrupt.h>              //interrupt header file

```

```

ISR(INT1_vect)                                //interrupt subroutine for
INT1_vect
{
    PORTB &= ~(1 << 4);                      //turn off PB4
    _delay_ms(3000);                          //delay for 3000ms
}

int main(void)                                //main code
{
    DDRB |= (1 << 4);                      //set PB4 as output
    PORTD |= (1 << 3);                      //enable pull-up resistor for
PD3

    EICRA |= (1 << ISC11) | (1 << ISC10);   //rising edge trigger
    EIMSK |= (1 << INT1);                   //enable INT1
    sei();                                    //enable global interrupts

    while (1)                                //forever loop
    {
        PORTB |= (1 << 4);                  //turn on PB4
    }

    return 0;                                //return 0
}

```

```

; DA2_3A.asm
; Created: 3/10/2024 12:52:21 PM
; Author : SON

#define F_CPU 8000000UL                      ;8MHz clock
.include "m328pbdef.inc"                    ;m328pb device header file
.org 0x0000                                ;set program counter
the start of the code
    jmp main                                ;to main
.org 0x0004                                ;set to the address of
INT1
    jmp INT1_ISR                            ;jmp to INT1_ISR for INT1
subroutine

main:                                         ;main

```

```

ldi r16, 1 << 4 ;set PB4
out DDRB, r16 ;as the output

ldi r17, 1 << 3 ;set PD3
out PORTD, r17 ;as the input

ldi r18, (1 << ISC11) | (1 << ISC10) ;load 1 << ISC11 or 1 << ISC10 for rising edge
trigger for IN1
sts EICRA, r18 ;in register EICRA

ldi r19, 1 << INT1 ;enable INT1
sts EIMSK, r19 ;in register EIMSK

sei ;enable global
interrupts

loop:
    in r20, PIND ;Read PD3
    andi r20, (1 << PIND3) ;Mask out all bits except for PD3
    cpi r20, (1<<PIND3) ;check if PD3 on
    brne INT1_ISR ;break if PD3 is GND
    ldi r21, (1 << 4) ;load '1' to PB4
    out PORTB, r21 ;output to PB4
    rjmp loop ;loop

INT1_ISR: ;INT1 subroutine
    ldi r22, ~(1 << 4) ;load '0' to PB4
    out PORTB, r22 ;output to PB4
    rcall delay ;call for delay
    jmp loop ;jmp back to loop

when done

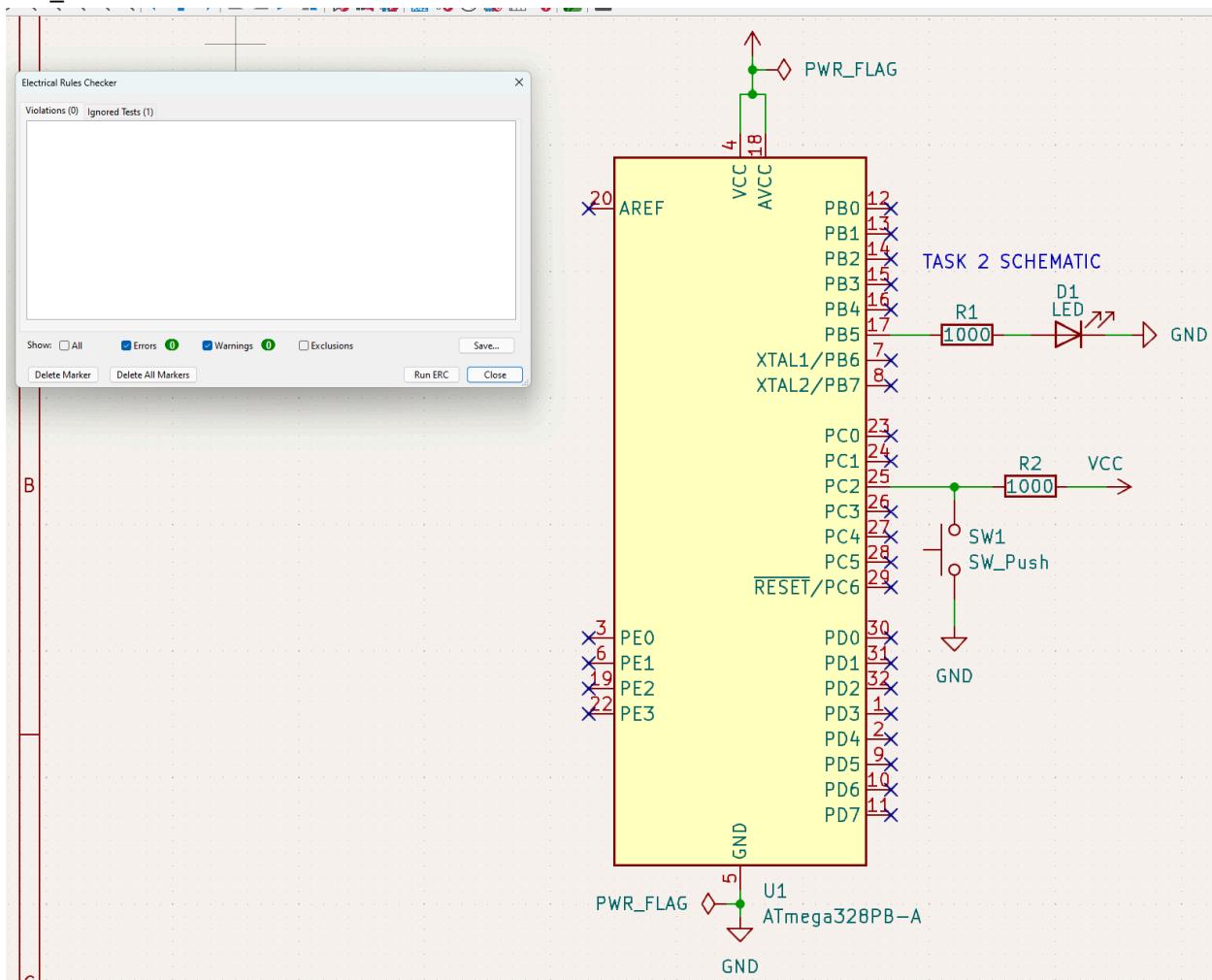
delay: ;delay
    ldi r18, 244 ;loaded value to help delay for
3000ms
    ldi r19, 130 ;loaded value to help delay for
3000ms
    ldi r20, 6 ;loaded value to help delay
for 3000ms
L1: dec r20 ;dec value to help delay for
3000ms
    brne L1 ;repeat till 0
    dec r19 ;dec value to help delay for
3000ms

```

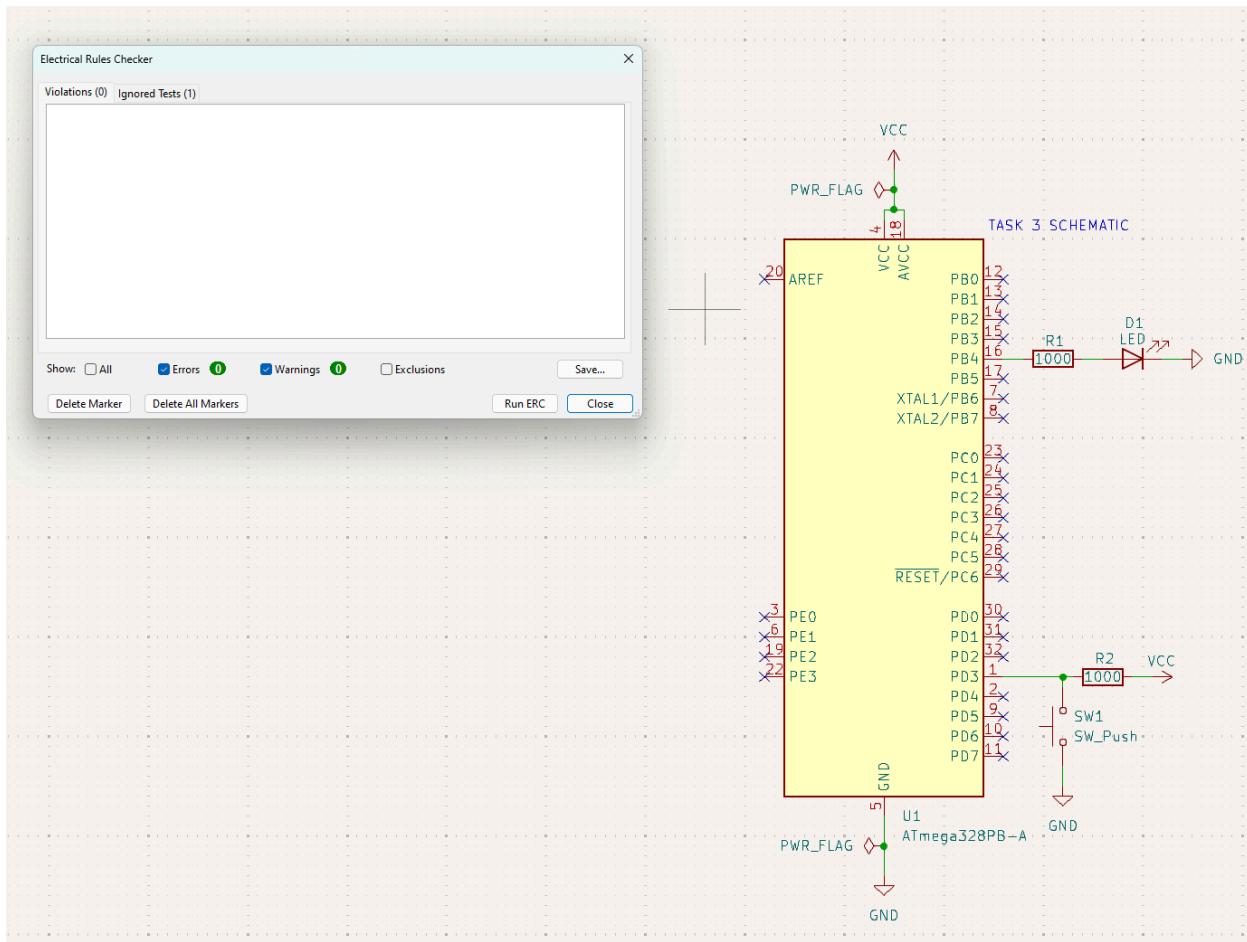
```
    brne L1                                ;repeat till 0
    dec r18                                 ;dec value to help delay for
3000ms
    brne L1                                ;repeat till 0
```

4. SCHEMATICS

DA2_2



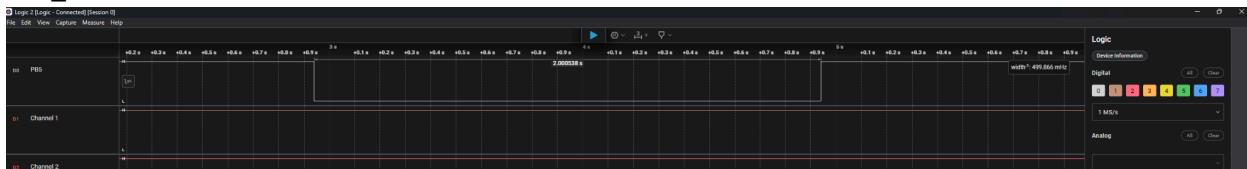
DA2_3



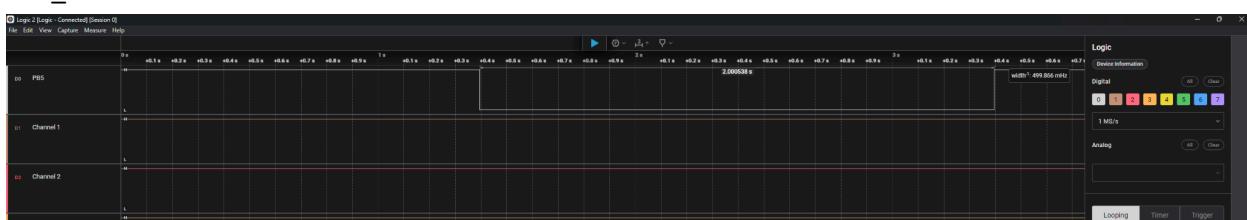
5. SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)

Task 2:

DA2_2C

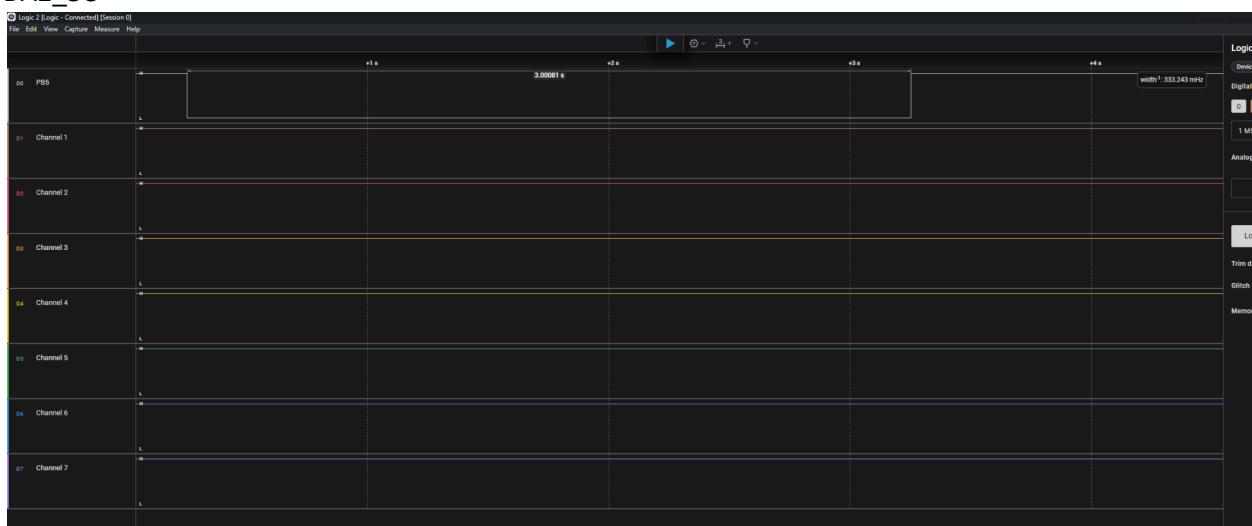


DA2_2A:

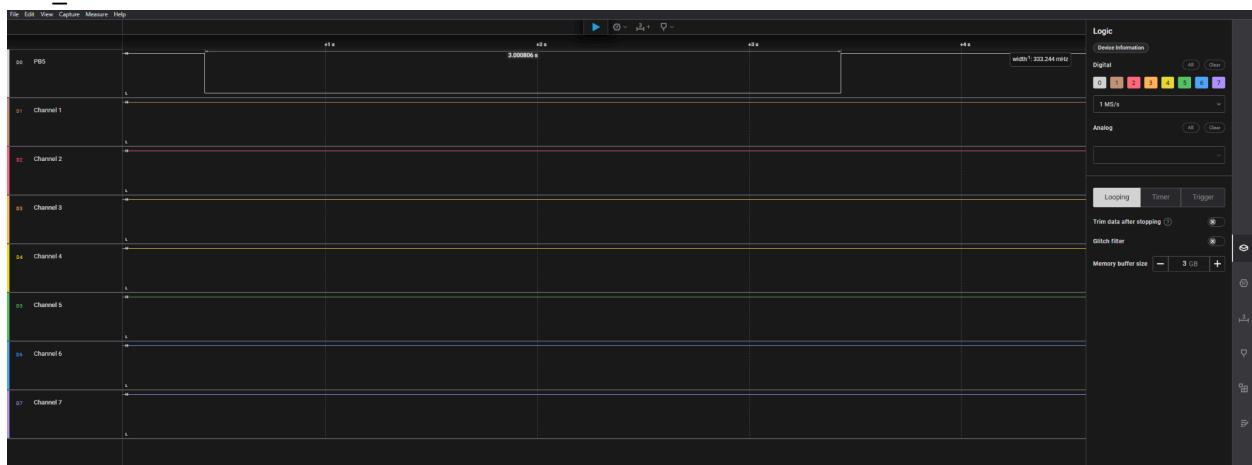


Task 3:

DA2_3C

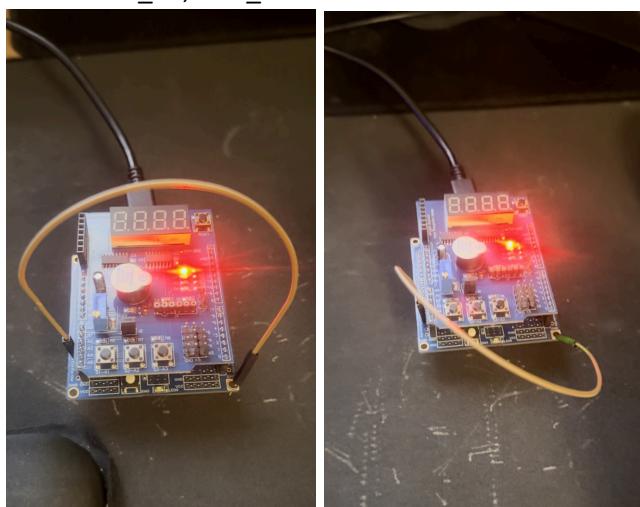


DA2_3A

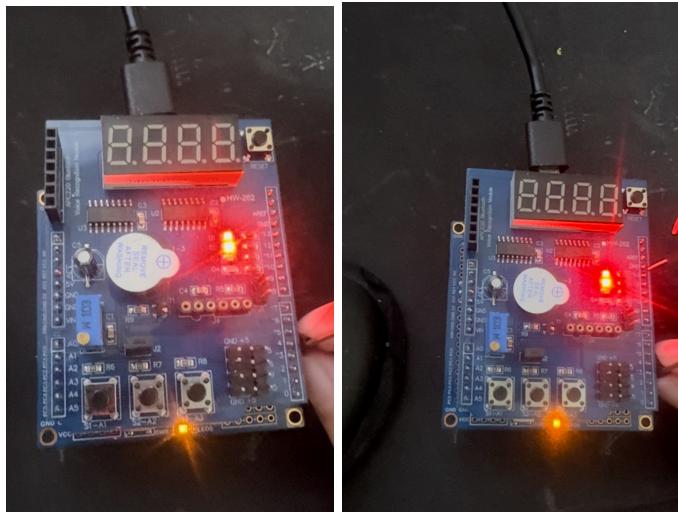


6. SCREENSHOT OF EACH DEMO (BOARD SETUP)

Task 2: DA2_2C, DA2_2A



Task 3: DA2_3C, DA2_3A



7. **VIDEO LINKS OF EACH DEMO**

[DA2_2C](#)

[DA2_2A](#)

[DA2_3C](#)

[DA2_3A](#)

8. **GITHUB LINK OF THIS DA**

https://github.com/SON-Abe/submission_da/tree/c3216529f2c10f97d77ef79090e4a20d41453c48/Design_Assignments/DA2

Student Academic Misconduct Policy

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

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

Abraham Garcia