CPE301 – SPRING 2019

Design Assignment DA2B

Student Name: Jett Guerrero

Student #: 5002377713

Student Email: guerrj1@unlv.nevada.edu

Primary Github address: <https://github.com/guerrj1/Submission_DA.git>

Directory: DA2B – <https://github.com/guerrj1/Submission_DA/tree/master/DA2B>

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).

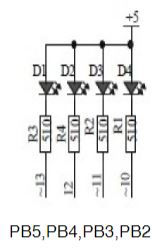
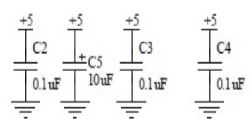
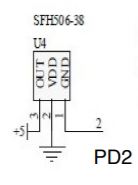
1. **COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS**

-ATMega328p

-Arduino Shield

-Micro-USB Cable

-Male to Male Wire

1. **DEVELOPED CODE OF TASK 1 ASSEMBLY CODE**

; DA2BT1.asm

.INCLUDE "M328PDEF.INC"

.org 0 ;location reset

jmp main

.org 0x02 ;location for external interrupt 0

jmp ex0\_isr ;external interrupt label

main:

ldi r20, high(ramend) ;loads the upper ram address

out sph, r20

ldi r20, low(ramend) ;loads lower ram address

out spl, r20 ;initialize the stack

sts eicra, r20 ;external interrupt control register

sbi portd, 2 ;pull-up activated

sbi ddrb, 2 ;set portb2 as output

cbi portb, 2 ;clears portb2

sbi ddrb, 5 ;sets pb5 as output

sbi portb, 5 ;turns off pb5

ldi r20, (1<<int0) ;enable INT0

out eimsk, r20 ;store EIMSK

LDI R20,5 ;set clock prescaler to 1024

STS TCCR1B, R20

sei ;enable interrupts

ex0\_isr: ;interrupt label

in r21, portb ;portb value stored into r21

ldi r22, (1<<2) ;sets a one in bit 2

eor r21, r22 ;toggles the led

out portb, r21 ;outputs the led

STS TCNT1H, R20

STS TCNT1L, R20

rcall delay\_on ;calls the delay to keep led on

reti

delay\_on:

LDS R29, TCNT1H ;loading upper bit of counter to R29

LDS R28, TCNT1L ;loading lower bit of counter to R28

CPI R28,0x4A ;comparing if lower 8 bits of timer is 0x08

BRSH body\_on ;if lower bits of timer have reached desired amount, check the upper bits

RJMP delay\_on ;otherwise, keep checking lower bits

body\_on:

CPI R29,0x4C ;check to see if upper timer bits have reached the desired value

BRLT delay\_on ;if not, recheck the lower bits

RET ;once the timer reached the desired value, toggle the LED

1. **DEVELOPED CODE FOR TASK 1 C CODE**

//DA2BT1C.c

#define *F\_CPU* 16000000UL

#include <avr/io.h>

#include <avr/interrupt.h>

#include <util/delay.h>

int main(void)

{

DDRB |= (1<<2); //sets pb2 as an output

DDRB |= (1<<5); //sets pb5 as an output

PORTB |= (1<<2); //sets pb2 led

PORTB |= (1<<5); //sets pb5 led off

DDRC &= (0<<2); //for push button output

PORTC |= (0<<2); //for push button

PORTD |= (1<<2); // enable PORTD.2 pin pull up resistor

EIMSK = (1<<INT0); //enable external interrupt 0

EICRA = 0x02; //interrupt on falling edge

sei();

while(1) //while running

{}

}

ISR(INT0\_vect)

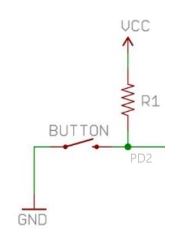
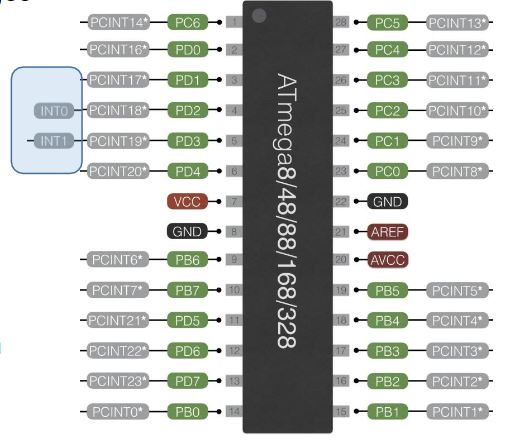
{

PORTB ^= (1<<2); //toggle pb2 led

*\_delay\_ms*(1250); //keep led on for 1250ms

}

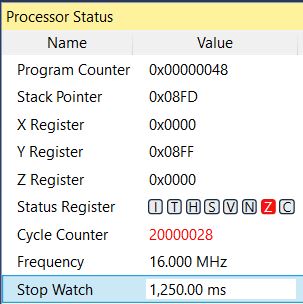
1. **SCHEMATICS**

****

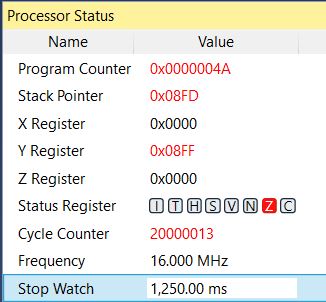
In this schematic, we are using PD2 for INT0. To activate PD2, we can use a pushbutton to trigger a 0 which will cause an interrupt in the system. The schematic on the right shows a simple schematic of the logic.

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

Assembly Processor Status Simulation for 1.25 sec Delay

****

C Code Processor Status Simulation for 1.25 sec Delay

****

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

****

LED is initially off. Turns on after pressing the push button and stay on for 1.25 secs.

1. **VIDEO LINKS OF EACH DEMO**

**Assembly Code Demo**

[**https://youtu.be/neLtCoso4XY**](https://youtu.be/neLtCoso4XY)

**C Code Demo**

[**https://youtu.be/PS8ZCJxHJcA**](https://youtu.be/PS8ZCJxHJcA)

1. **GITHUB LINK OF THIS DA**

<https://github.com/guerrj1/Submission_DA/tree/master/DA2B>

**Student Academic Misconduct Policy**

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

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

Jett Guerrero