CPE301 – SPRING 2020

Design Assignment 2C

Student Name: Jason Villanueva

Student #: 2001255420

Student Email: villaj21@unlv.nevada.edu

Primary Github address: <https://github.com/jasonvillanuevagit/submission_designAssignments->

Directory: <https://github.com/jasonvillanuevagit/submission_designAssignments-/tree/master/DesignAssignment2C>

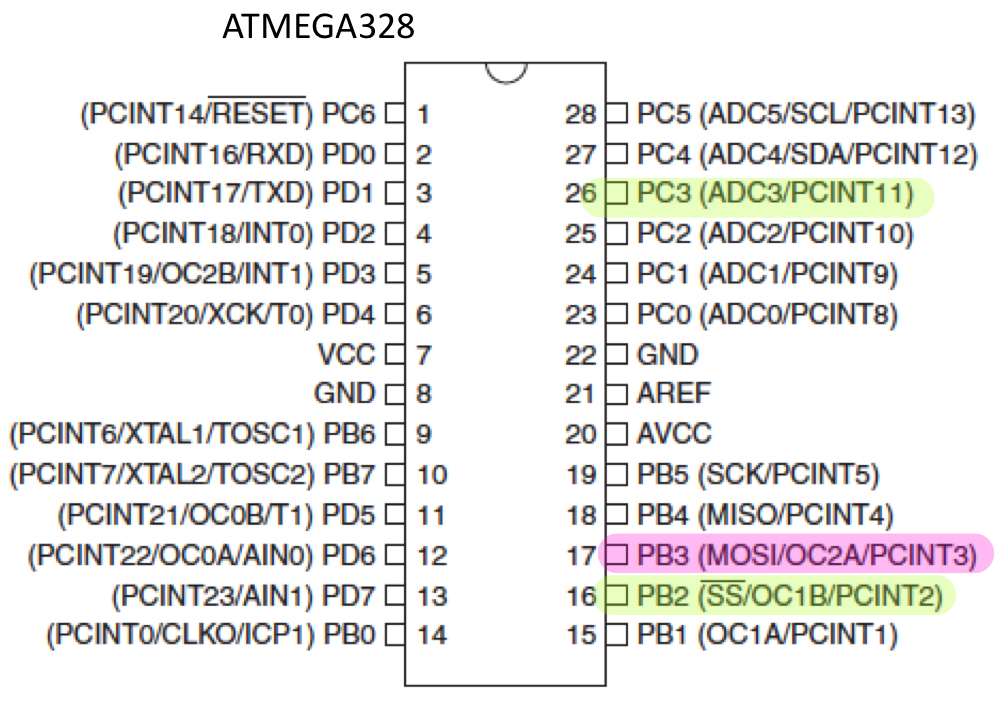
1. **COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS**

Atmel Studio 7.0 Atmega328PB-Xmini PC Multi-Function Shield Button

- Assembler -LEDs

- Simulator

- Debugger



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

Task 1 Code

#define F\_CPU 16000000UL

#include <avr/io.h>

#include <util/delay.h>

void set(){

TCCR0A = 0;//NORMAL MODE OPERATION

TCCR0B = 0X05;//THE PRESCALER SET TO 1024

TCNT0 = 0X00;//COUNTER VALUE = 0

}

void On\_413(){

int cycle = 25;//CYCLE AMOUNT FOR 413ms

DDRB |= (1<<3);//SET PORTB.3 "ON"

while (cycle != 0){

cycle --;

while((TIFR0 & 0X01) == 0);

TIFR0 = 0X01;//RESET OVERFLOW FLAG

}

}

void Off\_337(){

int cycle = 20;//CYCLE AMOUNT FOR 337ms

DDRB = (0<<3);//SET PORTB.3 "OFF"

while (cycle != 0){

cycle --;

while ((TIFR0 & 0X01) == 0);

TIFR0 = 0X01;//RESET OVERFLOW FLAG

}

}

void Button(){

int cycle = 122;//CYCLE AMOUNT FOR 2s

DDRB = (0<<3);//SET PORTB.3 "OFF"

DDRB |= (1<<2);//SET PORTB.2 "ON"

while (cycle != 0){

cycle --;

while ((TIFR0 & 0X01) == 0);

TIFR0 = 0X01;//RESET OVERFLOW FLAG

}

DDRB = (0<<3);//SET PORTB.3 "OFF"

}

int main(void){

DDRC &=~ (1<<3);//SET PORTC.3 INPUT

PORTC |= (1<<3);//PULL-UP RESISTOR ACTIVATE

while (1){

set();//CALL SET FCT

if (!(PINC & (1 <<3))){//IF BUTTON PRESS @ PINC.3

Button();//CALL BUTTON FCT

}

On\_413();//CALL On\_413 FCT

Off\_337();//CALL Off\_337 FCT

}

return 1;

}

Task 2 Code

#define F\_CPU 16000000UL

#include <avr/io.h>

#include <avr/interrupt.h>

volatile uint8\_t Overflow;

void set(){

TCCR0A = 0;//NORMAL MODE OPERATION

TCCR0B = 0X05;//THE PRESCALER SET TO 1024

TCNT0 = 0X00;//COUNTER VALUE = 0

TIMSK0 = (1<<TOIE0);//ENABLE TIMER INTERUPT

sei();//ENABLE GLOBAL INTERUPT

}

ISR(TIMER0\_OVF\_vect)

{

Overflow++;//INCREMENT OVERFLOW

}

void Button(){

DDRB = (0<<3);//SET PORTB.3 "OFF"

Overflow = 0;//RESET OVERFLOW

set();//CALL SET FCT

DDRB |= (1<<2);//SET PORTB.2 "ON"

while (Overflow < 122);

DDRB = (0<<2);//SET PORTB.2 "OFF"

Overflow = 0;

}

int main(void)

{

DDRC &=~ (1<<3);//SET PORTC.3 INPUT

PORTC |= (1<<3);//PULL-UP RESISTOR ACTIVATE

set();//CALL set FCT

DDRB = (1<<3);//SET PORTB.3 "ON"

while (1){

//CONTROL ENTERS IF BUTTON PRESSED @ PORTB.3

if (!(PINC & (1<<3))){

Button();//CALL Button FCT.

}

//CONTROL ENTERS IF OVERFLOW VALUE BETWEEN 20 & 25

if ((Overflow >= 20) & (Overflow < 25)){

DDRB = (0<<3);//SET PORTB.3 "OFF"

}

//CONTROL ENTERS IF OVERFLOW VALUE IS GREATER THAN 25

if ((Overflow > 25)){

DDRB = (1<<3);//SET PORTB.3 "ON"

Overflow = 0;//RESET OVERFLOW TO 0

}

}

return 1;

}

Task 3 Code

#define F\_CPU 16000000UL

#include <avr/io.h>

#include <avr/interrupt.h>

volatile uint8\_t Overflow;

void set(){

TCCR0A = (1<<WGM01);//CTC MODE OPERATION

TCCR0B = 0X05;//THE PRESCALER SET TO 1024

TCNT0 = 0X00;//COUNTER VALUE = 0

OCR0A = 0X80;//COMPARE REGISTER VALUE OF 128

TIMSK0 = (1<<OCIE0A);//ENABLE TIMER COMPARE INTERUPT

sei();//ENABLE GLOBAL INTERUPT

}

ISR(TIMER0\_OVF\_vect){

Overflow++;//INCREMENT OVERFLOW

}

void Button(){

DDRB = (0<<3);//SET PORTB.3 "OFF"

Overflow = 0;//RESET OVERFLOW

set();//CALL SET FCT

DDRB |= (1<<2);//SET PORTB.2 "ON"

while (Overflow < 243);

DDRB = (0<<2);//SET PORTB.2 "OFF"

Overflow = 0;

}

int main(void){

DDRC &=~ (1<<3);//SET PORTC.3 INPUT

PORTC |= (1<<3);//PULL-UP RESISTOR ACTIVATE

set();//CALL set FCT

DDRB = (1<<3);//SET PORTB.3 "ON"

while (1){

//CONTROL ENTERS IF BUTTON PRESSED @ PORTB.3

if (!(PINC & (1<<3))){

Button();//CALL Button FCT.

}

//CONTROL ENTERS IF OVERFLOW VALUE BETWEEN 50 & 73

if ((Overflow >= 50) & (Overflow < 73)){

DDRB = (0<<3);//SET PORTB.3 "OFF"

}

//CONTROL ENTERS IF OVERFLOW VALUE IS GREATER THAN 73

if ((Overflow > 73)){

DDRB = (1<<3);//SET PORTB.3 "ON"

Overflow = 0;//RESET OVERFLOW TO 0

}

}

return 1;

}

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

N/A

1. **SCHEMATICS**

N/A

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

N/A

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

A circuit board

Description automatically generatedA circuit board

Description automatically generated

1. **VIDEO LINKS OF EACH DEMO**

Task1

<https://youtu.be/rz4ezrzJFi8>

Task2

<https://youtu.be/348mFFe85oU>

Task3

<https://youtu.be/eAuFMcSwNGA>

1. **GITHUB LINK OF THIS DA**

<https://github.com/jasonvillanuevagit/submission_designAssignments-/tree/master/DesignAssignment2C>

**Student Academic Misconduct Policy**

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

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

Jason Villanueva