CPE301 – SPRING 2019

Design Assignment 3A

Student Name: Tanner Tindall

Student #: 8000733733

Student Email: tindat1@unlv.nevada.edu

Primary Github address: <https://github.com/TannerTindall51>

Directory: <https://github.com/TannerTindall51/tindalltannerm_submission/tree/master/Design_Assignments/DA3A>

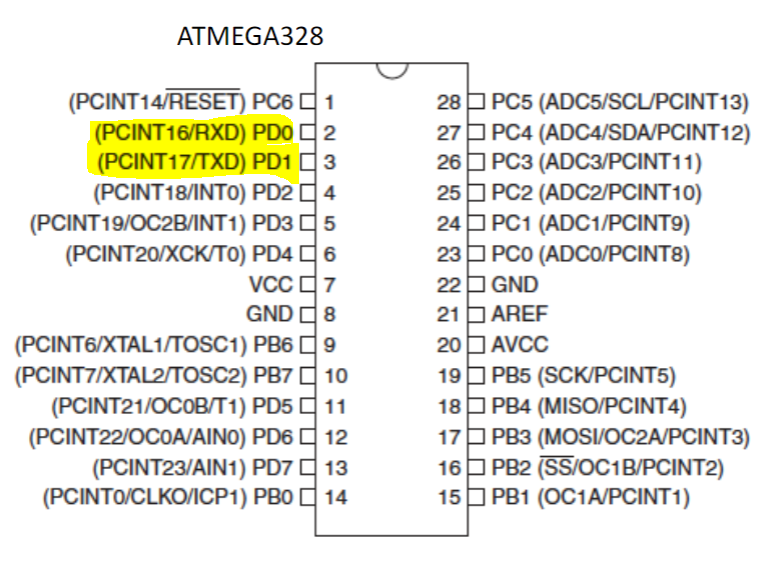
Design Assignment 2A: The goal of the assignment is use GPIO and delays:

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 7.0 (Assembler, Simulator, & Debugger)
* Atmega328PB-Xmini
* Micro USB



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

|  |
| --- |
|  |

Task 1:

//

//Design Assignment 3A - Task 1

//Tanner Tindall

//

#define BAUD 9600

#define *F\_CPU* 16000000UL

#include <avr/io.h>

#include <util/delay.h>

#include <util/setbaud.h>

#include <avr/interrupt.h>

#include <stdlib.h>

#include <stdio.h>

void USART\_init(void) //initialize UART

{ //the baud rate consists of 16 bits therefore a low and high is needed

UBRR0H = *UBRRH\_VALUE*; //high end of baud rate

UBRR0L = *UBRRL\_VALUE*; //low end of baud rate

UCSR0C = \_BV(UCSZ01) | \_BV(UCSZ00); //assigning bit values to UCSR0C

UCSR0B = \_BV(RXEN0) | \_BV(TXEN0); //enabling Reception Complete Interruption & turning on RX/TX

}

void USART\_tx\_string(char\*data) //outputs data to terminal

{

while((\*data!='\0')) //loop until all data is emptied

{

while(!(UCSR0A & (1 << UDRE0))); //wait until data register in emptied

UDR0 = \*data; //once emptied, import into data register UDR0

data++; //increments pointer position for data

}

}

int main(void){

USART\_init(); //initialize USART

int num = 51; //integer value

char array1[34]; //integer array

char array2[34]; //float array

while (1) //primary loop

{

volatile float float\_value = num \* 5; //initialize float value

num = ((*rand*() \* 5) % 100); //randomizing the value

*itoa* (num, array1, 10); //making the integer into a string

*snprintf*(array2,sizeof(array2), "%f\r\n", float\_value); //converting into a character array

USART\_tx\_string("supercalifragilisticexpialidocious"); //print string

USART\_tx\_string(" ");

USART\_tx\_string(array1); //print integer

USART\_tx\_string(" ");

USART\_tx\_string(array2); //print float

USART\_tx\_string("\n");

USART\_tx\_string("\n");

*\_delay\_ms*(1000);

}

}

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

Task 2:

//

//Design Assignment 3A - Task 2

//Tanner Tindall

//

#define BAUD 9600

#define *F\_CPU* 16000000UL

#include <avr/io.h>

#include <util/setbaud.h>

#include <avr/interrupt.h>

#include <stdlib.h>

#include <stdio.h>

volatile *uint8\_t* overFlow; //initializing a global variable to be used as primary overflow for timer

void USART\_init(void) //initialize UART

{ //the baud rate consists of 16 bits therefore a low and high is needed

UBRR0H = *UBRRH\_VALUE*; //high end of baud rate

UBRR0L = *UBRRL\_VALUE*; //low end of baud rate

UCSR0C = \_BV(UCSZ01) | \_BV(UCSZ00); //assigning bit values to UCSR0C

UCSR0B = \_BV(RXEN0) | \_BV(TXEN0); //enabling Reception Complete Interruption & turning on RX/TX

}

void USART\_tx\_string(char\*data) //outputs data to terminal

{

while((\*data!='\0')) //loop until all data is emptied

{

while(!(UCSR0A & (1 << UDRE0))); //wait until data register in emptied

UDR0 = \*data; //once emptied, import into data register UDR0

data++; //increments pointer position for data

}

}

void timer() //timer function

{

TCNT0 = 0; //initializes the 8 bit timer0 register to begin at 0

TCCR0A = 0; //sets the timer to "Normal" Mode by WGM00

TCCR0B |= (1 << CS02) | (1 << CS00);//sets prescaler value to 1024

TIMSK0 = (1<<TOIE0); //enable timer0 overflow interrupt

sei(); //enable interrupts

}

ISR(TIMER0\_OVF\_vect) //timer interrupt function

{

overFlow++; //increment overflow counter

}

int main(void){

USART\_init(); //initialize USART

timer(); //call timer function

int num = 51; //integer value

char array1[34]; //integer array

char array2[34]; //float array

while (1) //primary loop

{

volatile float float\_value = num \*5; //initialize float value

if (overFlow == 61)

{

num = ((*rand*() \* 5) % 100); //randomizing the value

*itoa* (num, array1, 10); //making the integer into a string

*snprintf*(array2,sizeof(array2), "%f\r\n", float\_value); //converting into a character array

USART\_tx\_string("supercalifragilisticexpialidocious"); //print string

USART\_tx\_string(" ");

USART\_tx\_string(array1); //print integer

USART\_tx\_string(" ");

USART\_tx\_string(array2); //print float

USART\_tx\_string("\n");

USART\_tx\_string("\n");

overFlow = 0; //resetting overFlow counter

}

}

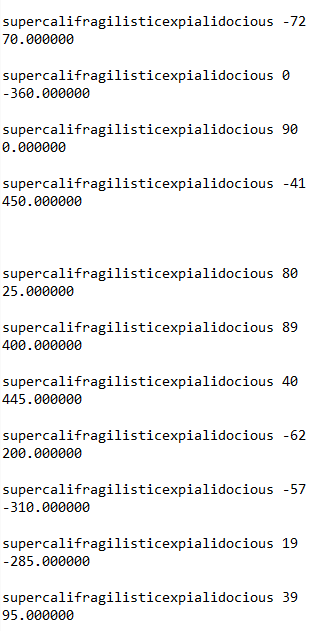
}

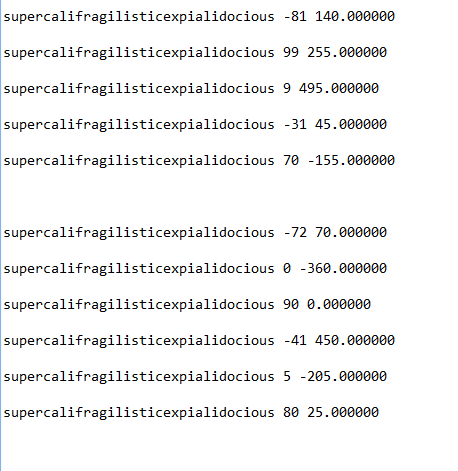
1. **SCHEMATICS**

N/A

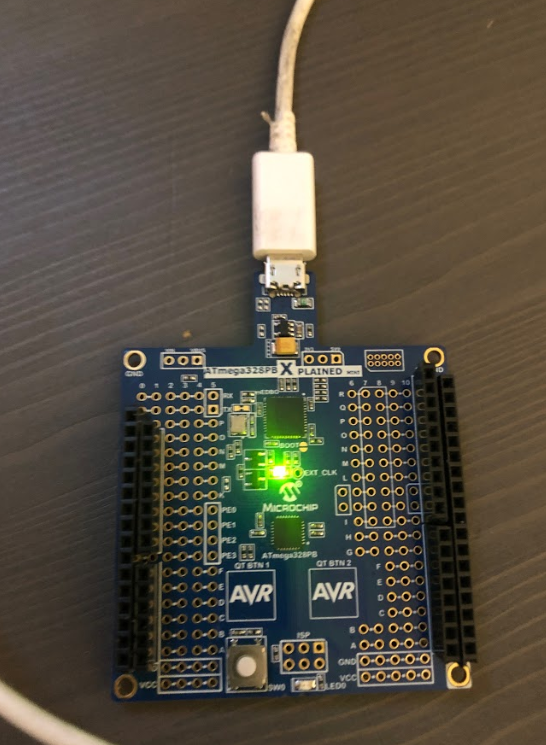
1. **SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)**

The images below display the output of Task 1 (left)and Task 2 (right) seen in the terminal window.





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



1. **VIDEO LINKS OF EACH DEMO**

Task 1 Demo Video: <https://youtu.be/sQbnkWw9bjA>

Task 2 Demo Video: <https://youtu.be/uu_c8tHrCUI>

1. **GITHUB LINK OF THIS DA**

<https://github.com/TannerTindall51/tindalltannerm_submission/tree/master/Design_Assignments/DA3A>

**Student Academic Misconduct Policy**

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

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

Tanner Tindall