CPE301 – SPRING 2019

Design Assignment 3A

Student Name: Cody Hudson

Student #: 8000603642

Student Email: hudsoc1@unlv.nevada.edu

Primary Github address: https://github.com/eed911/class\_proj.git

Directory:

https://github.com/eed911/class\_proj/tree/master/DesignAssignments/DA3A/Project\_3A

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

List of Components used

* ATmega328P Xplained
* RS-232

Block diagram with pins used in the Atmega328P

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

Task1: Write a C AVR program that will display a string, random integer and floating point

values on the serial terminal every 1 sec. Use a timer with interrupt for the 1 sec

delay. Use a FTDI chip for serial to USB conversion

Code:

/\*

\* DA3A\_C.c

\*

\* Created: 3/26/2019 7:21:07 PM

\* Author : Cody Hudson

\*/

#define *F\_CPU* 16000000UL

#include <avr/io.h>

#include <util/delay.h>

#include <avr/interrupt.h>

#include <stdio.h>

#define BAUDRATE 9600

#define BAUD\_PRESCALLER (((*F\_CPU* / (BAUDRATE \* 16UL)))-1)

//Functions

volatile int Count;

void USART\_init( unsigned int ubrr ); //CREATES A VOID FUNCTION "USART\_init"

void USART\_TX\_string(char \*data); //CREATES A VOID FUNCITON "USART\_TX\_string"

char outs[30]; //SETS VARIABLE "OUTS" AS A MATRIX FOR CONVERSIONS

int n; //SETS "n" AS AN INTEGER VALUE

float PI; //SETS "PI" AS A FLOATING VARIABLE

char str[] = "Lets Get It"; //CREATES STRING "LETS GET IT"

char empty[] = " "; //CREATES EMPTY STRING FOR SPACING

int main(void)

{

Count = 0;

TIMSK0 |= (1<<TOIE0); //SETS THE INERRUPT TO ACTIVATE WHEN TIMMER 0 GETS OVERFLOW FLAG

sei (); //SETS GLOBAL INTERRUPT

TCCR0A = 0x00; //SETS NORMAL MODE ON TIMMER

TCCR0B |= (1<<CS02)|(1<<CS00); //SETS PRESCALER TO 1024

USART\_init(BAUD\_PRESCALLER); //SETS THE BAUD PRESCALLER

USART\_TX\_string("Connected!\r\n"); //PRINTS "CONNECTED" TO SERIAL. WE MADE IT!!

while (1); //WAITS HERE FOR TIMMER INTERUPT

}

/\*INT USART (RS-232)\*/

void USART\_init( unsigned int ubrr ){

UBRR0H = (unsigned char)(ubrr>>8); //SIZING THE PRESCALLER

UBRR0L = (unsigned char)ubrr;

UCSR0B = (1 << TXEN0); //Enable Tx interrupt

UCSR0C = (3 << UCSZ00); //asynchronous 8 N 1

}

/\* SEND A STRING TO TEH RS-232\*/

void USART\_TX\_string(char \*data) { //FUNCTION CREATED IN ORDER TO SEND THROUGH SERIAL

while (\*data != '\0') {

while (!(UCSR0A & (1<<UDRE0)));

UDR0 = \*data;

data++;

}

}

ISR (TIMER0\_OVF\_vect){ //TV0V FLAG INTERUPT

while (Count < 60){ //WHILE COUNT IS LESS THAN 76 LOOP

if ((TIFR0 & 0x01) == 1){ //IF TV0V FLAG IS HIGH DO THIS

TIFR0 = 0X01; //CLEAR FLAG

Count++; //INCREMENT COUNT

}

}

if (Count > 59){ //IF COUNT IS > 60 EXICUTE THIS

USART\_TX\_string(str); //PRINT STRING "LETS GET IT"

USART\_TX\_string(empty); //PRINTS A BLANK SPACE

n = *rand*(); //GENERATES A RANDOM NUMBER STORES TO VARIABLE n

PI = 3.1415; //GENERATES PI AND STORES

*snprintf*(outs, sizeof(outs), "%3d\r\n", n);

USART\_TX\_string(outs); //PRINTS THE STING VERSION OF RANDOM NUMBER TO SERIAL PORT

USART\_TX\_string(empty); //PRINTS A BLANK SPACE

*sprintf*(outs, "%f", PI); //CONVERTS THE FLOTING VALUE PI INTO A STRING TO BE SENT THROUGH THE TERMAINL

USART\_TX\_string(outs); //PRINTS THE STRING VERSION PI TO THE SERIAL MONITOR

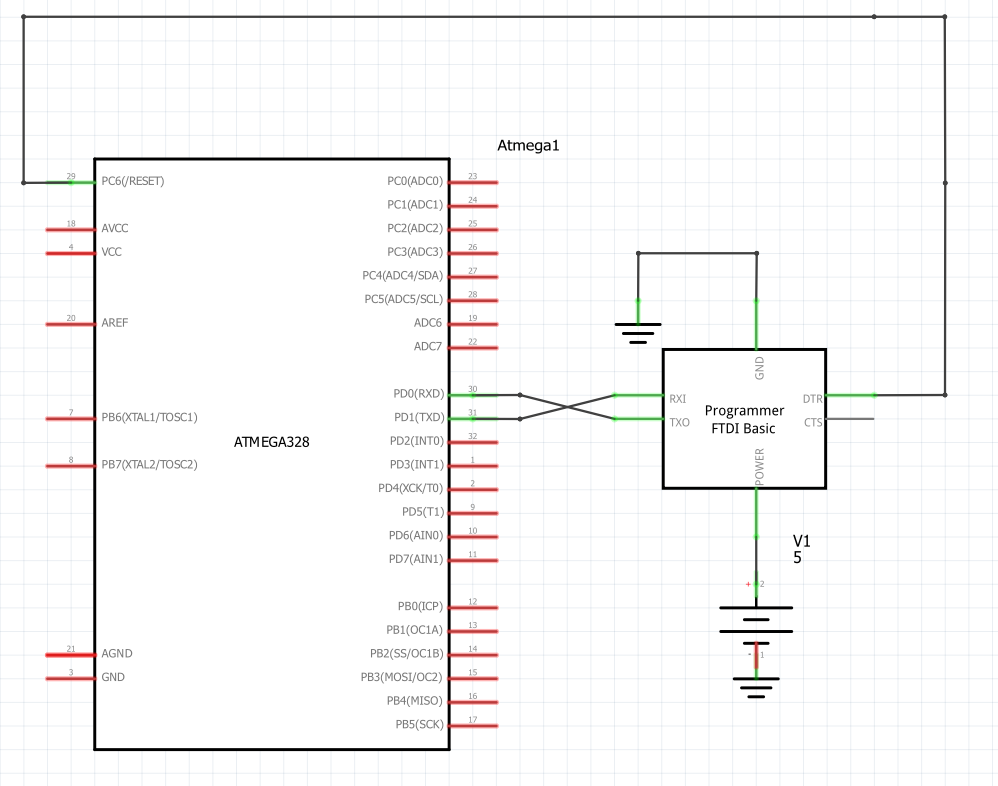
USART\_TX\_string(empty); //PRINTS A BLANK SPACE

Count = 0; //RESETS THE COUNT VARIABLE BACK TO 0

}

}

1. **SCHEMATICS**

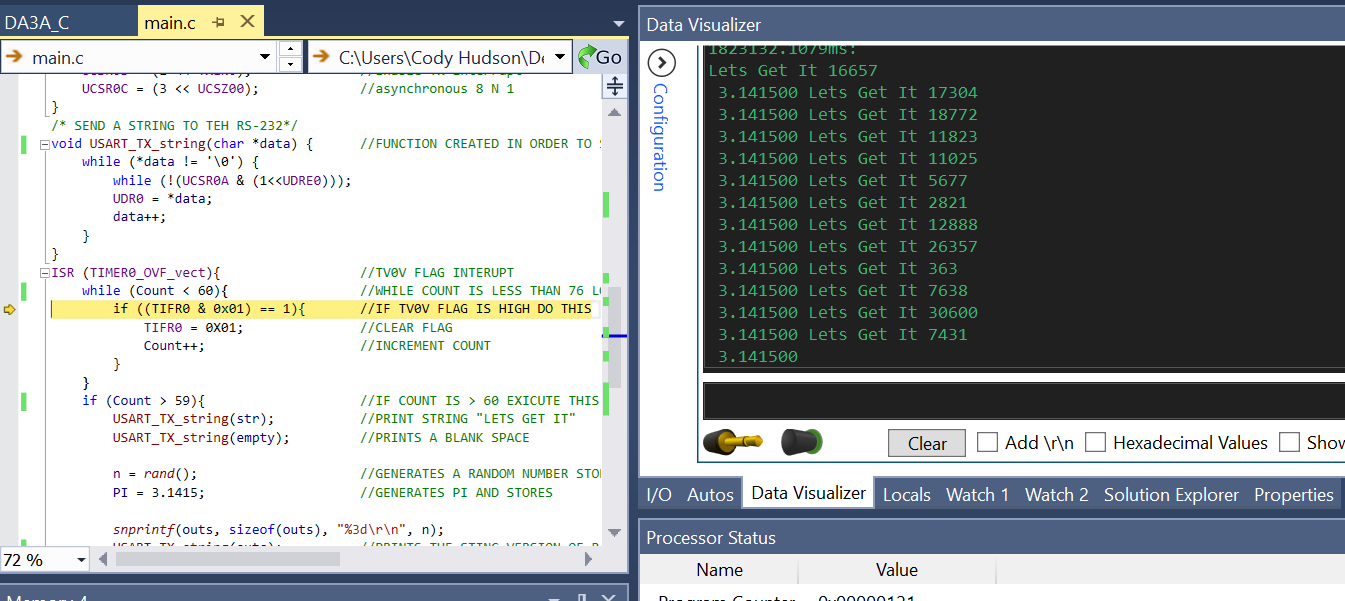


1. **SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)**

N/a all demonstration is done via video on YouTube

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

Screenshot of demo:



1. **VIDEO LINKS OF EACH DEMO**

Demo1:

<https://youtu.be/mN1thIi6GzU>

1. **GITHUB LINK OF THIS DA**

https://github.com/eed911/class\_proj.git

**Student Academic Misconduct Policy**

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

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

NAME OF THE STUDENT