CPE301 – SPRING 2019

Design Assignment 6

Student Name: Mat Tomljenovic

Student #: 1012254710

Student Email: tomljeno@unlv.nevada.edu

Primary Github address: https://github.com/matcroatia/

Directory: https://github.com/matcroatia/submission\_da

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

Atmega328P microcontroller, Xplained mini gyroscope.

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

/\*

\* DA6.c

\*

\* Created: 5/3/2019 5:27:54 PM

\* Author : Mat Tomljenovic

\*/

#define F\_CPU 16000000UL

#include <avr/io.h>

#include <util/delay.h>

#include <inttypes.h>

#include <stdlib.h>

#include <stdio.h>

#include "MPU6050\_res\_define.h"

#include "I2C\_Master\_H\_file.h"

#include "USART\_RS232\_H\_file.h"

float Acc\_x,Acc\_y,Acc\_z,Temperature,Gyro\_x,Gyro\_y,Gyro\_z;

void MPU6050\_Init() /\* Gyro initialization function \*/

{

*\_delay\_ms*(150); /\* Power up time >100ms \*/

I2C\_Start\_Wait(0xD0); /\* Start with device write address \*/

I2C\_Write(SMPLRT\_DIV); /\* Write to sample rate register \*/

I2C\_Write(0x07); /\* 1KHz sample rate \*/

I2C\_Stop();

I2C\_Start\_Wait(0xD0);

I2C\_Write(PWR\_MGMT\_1); /\* Write to power management register \*/

I2C\_Write(0x01); /\* X axis gyroscope reference frequency \*/

I2C\_Stop();

I2C\_Start\_Wait(0xD0);

I2C\_Write(CONFIG); /\* Write to Configuration register \*/

I2C\_Write(0x00); /\* Fs = 8KHz \*/

I2C\_Stop();

I2C\_Start\_Wait(0xD0);

I2C\_Write(GYRO\_CONFIG); /\* Write to Gyro configuration register \*/

I2C\_Write(0x18); /\* Full scale range +/- 2000 degree/C \*/

I2C\_Stop();

I2C\_Start\_Wait(0xD0);

I2C\_Write(INT\_ENABLE); /\* Write to interrupt enable register \*/

I2C\_Write(0x01);

I2C\_Stop();

}

void MPU\_Start\_Loc()

{

I2C\_Start\_Wait(0xD0); /\* I2C start with device write address \*/

I2C\_Write(ACCEL\_XOUT\_H); /\* Write start location address from where to read \*/

I2C\_Repeated\_Start(0xD1); /\* I2C start with device read address \*/

}

void Read\_RawValue()

{

MPU\_Start\_Loc(); /\* Read Gyro values \*/

Acc\_x = (((int)I2C\_Read\_Ack()<<8) | (int)I2C\_Read\_Ack());

Acc\_y = (((int)I2C\_Read\_Ack()<<8) | (int)I2C\_Read\_Ack());

Acc\_z = (((int)I2C\_Read\_Ack()<<8) | (int)I2C\_Read\_Ack());

Temperature = (((int)I2C\_Read\_Ack()<<8) | (int)I2C\_Read\_Ack());

Gyro\_x = (((int)I2C\_Read\_Ack()<<8) | (int)I2C\_Read\_Ack());

Gyro\_y = (((int)I2C\_Read\_Ack()<<8) | (int)I2C\_Read\_Ack());

Gyro\_z = (((int)I2C\_Read\_Ack()<<8) | (int)I2C\_Read\_Nack());

I2C\_Stop();

}

int main()

{

char buffer[20], float\_[10];

float Xa,Ya,Za,t;

float Xg=0,Yg=0,Zg=0;

I2C\_Init(); /\* Initialize I2C \*/

MPU6050\_Init(); /\* Initialize MPU6050 \*/

USART\_Init(9600); /\* Initialize USART with 9600 baud rate \*/

while(1)

{

Read\_RawValue();

Xa = Acc\_x/16384.0; /\* Divide raw value by sensitivity scale factor to get real values \*/

Ya = Acc\_y/16384.0;

Za = Acc\_z/16384.0;

Xg = Gyro\_x/16.4;

Yg = Gyro\_y/16.4;

Zg = Gyro\_z/16.4;

t = (Temperature/340.00)+36.53; /\* Convert temperature in °/c using formula \*/

*dtostrf*( Xa, 3, 2, float\_ ); /\* Take values in buffer to send all parameters over USART \*/

*sprintf*(buffer," Ax = %s g\t",float\_);

USART\_SendString(buffer);

*dtostrf*( Ya, 3, 2, float\_ );

*sprintf*(buffer," Ay = %s g\t",float\_);

USART\_SendString(buffer);

*dtostrf*( Za, 3, 2, float\_ );

*sprintf*(buffer," Az = %s g\t",float\_);

USART\_SendString(buffer);

*dtostrf*( t, 3, 2, float\_ );

*sprintf*(buffer," T = %s%cC\t",float\_,0xF8); /\* 0xF8 Ascii value of degree '°' on serial \*/

USART\_SendString(buffer);

*dtostrf*( Xg, 3, 2, float\_ );

*sprintf*(buffer," Gx = %s%c/s\t",float\_,0xF8);

USART\_SendString(buffer);

*dtostrf*( Yg, 3, 2, float\_ );

*sprintf*(buffer," Gy = %s%c/s\t",float\_,0xF8);

USART\_SendString(buffer);

*dtostrf*( Zg, 3, 2, float\_ );

*sprintf*(buffer," Gz = %s%c/s\r\n",float\_,0xF8);

USART\_SendString(buffer);

}

}

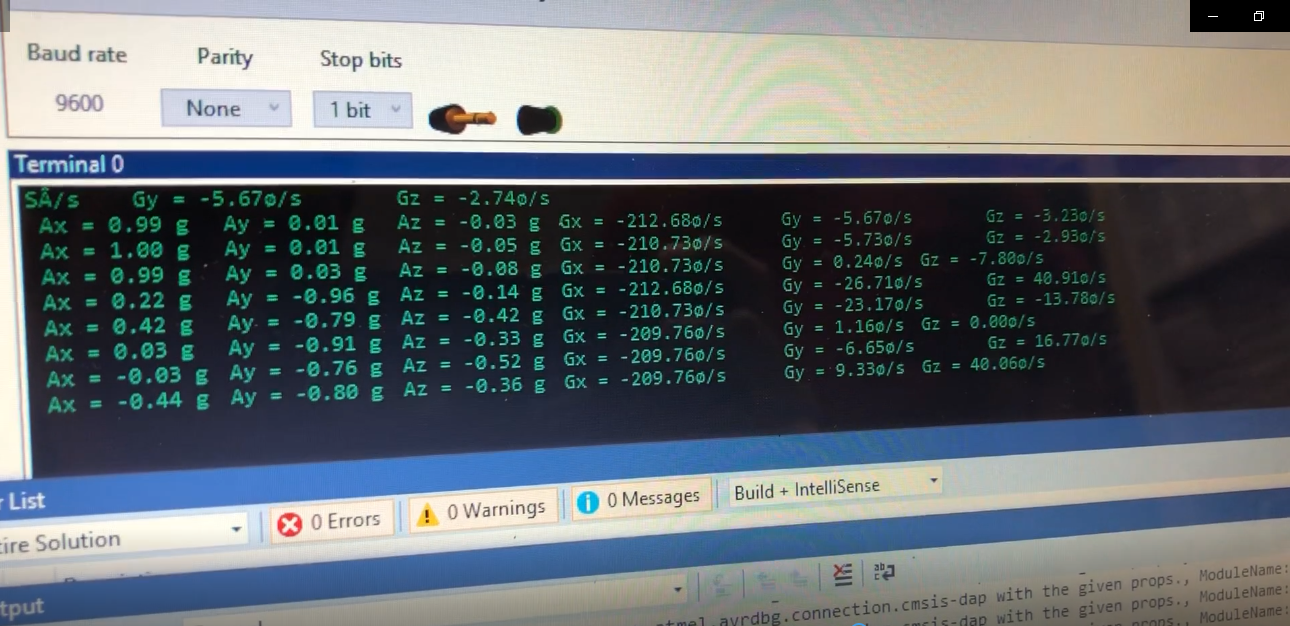
1. **DEVELOPED MODIFIED CODE OF TASK 2/A from TASK 1/A**

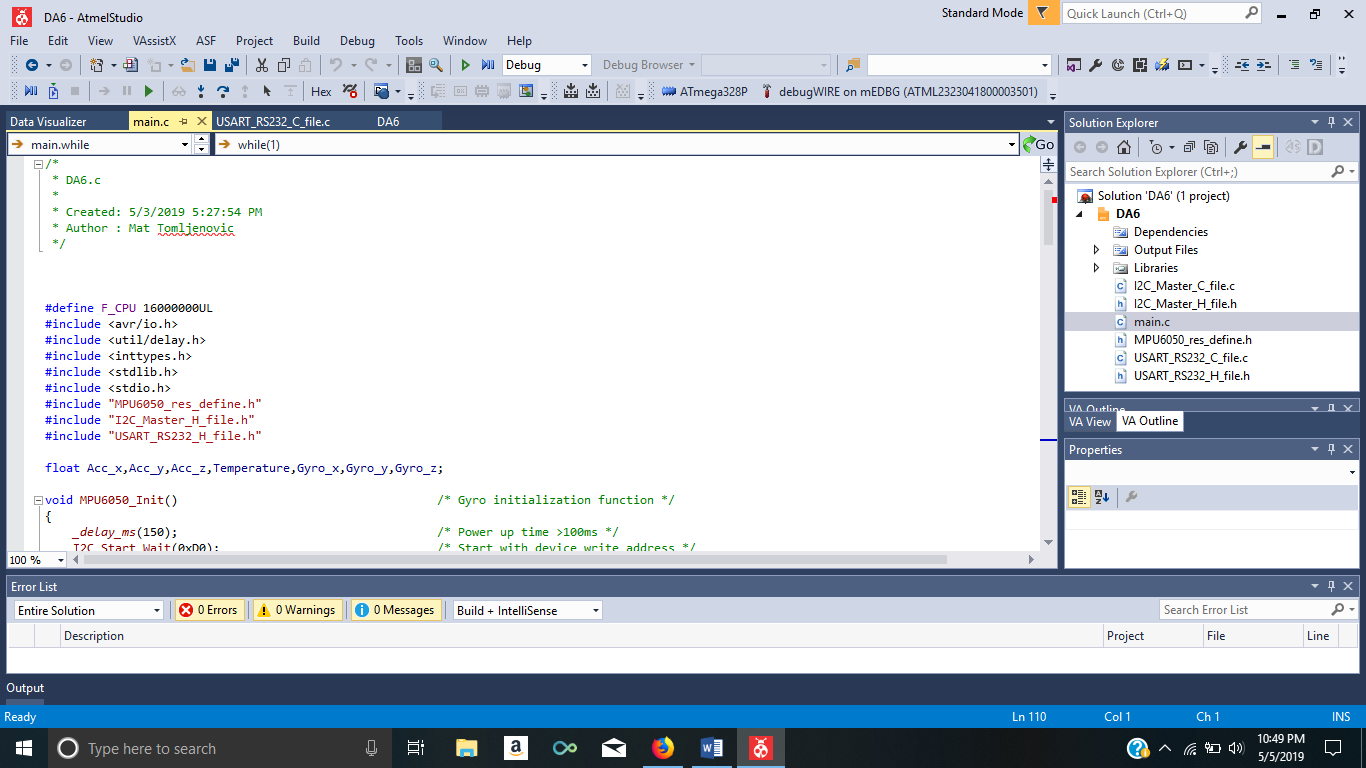
Same as above.

1. **SCHEMATICS**

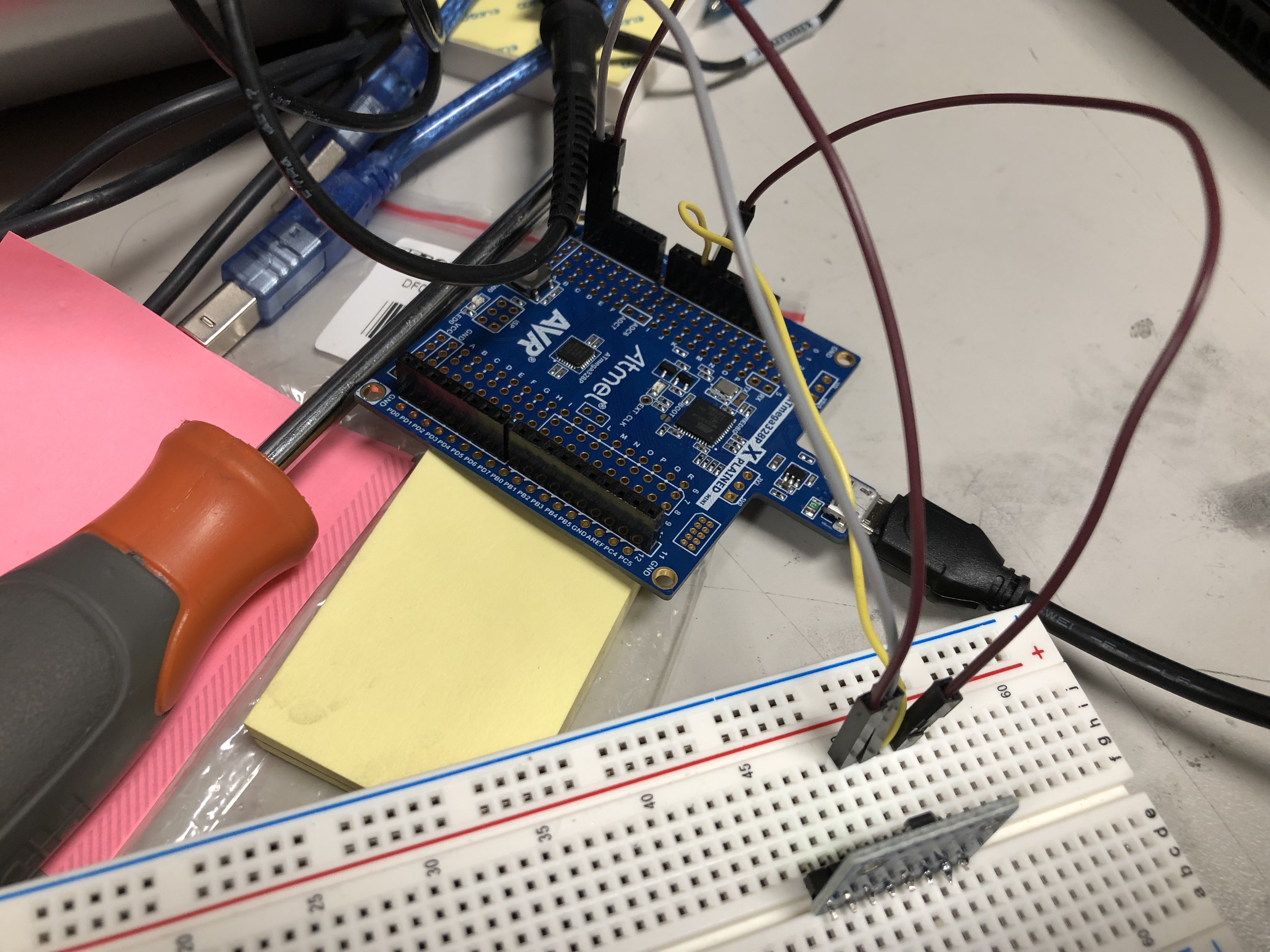
Use fritzing.org

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





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



1. **VIDEO LINKS OF EACH DEMO**

<https://youtu.be/G69euUeNAZM>

1. **GITHUB LINK OF THIS DA**

https://github.com/matcroatia/submission\_da/DA6

**Student Academic Misconduct Policy**

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

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

Matija Tomljenovic