

Design Assignment 6

Student Name: Jacob Patrick Reed

Student #: 1008448895

Student Email: reedj35@unlv.nevada.edu

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

Directory: https://github.com/reedjacobp/submission_da

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

ATmega328PB Xplained Mini
MPU6050

2. DEVELOPED CODE OF TASK 1

```
/*
 * DA6.c
 *
 * Created: 12/5/2019 4:58:33 PM
 * Author : jreed
 */

#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()
{
    _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);
start with device write address
    I2C_Write(ACCEL_XOUT_H);
start location address from where to read */
    I2C_Repeated_Start(0xD1);
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();
    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();
    MPU6050_Init();
    USART_Init(9600);

    while(1)
    {
        Read_RawValue();

        Xa = Acc_x/16384.0;
        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;
    }
}

```

```

        dtostrf( Xa, 3, 2, float_ ); /* Takes the
first parameter and formats it to be 3 characters, 2 decimal points of precision and puts
it in float_ */
        sprintf(buffer," Ax = %s g\t",float_); /* This is where the
value is being put as a string onto the buffer variable */
        USART_SendString(buffer); /* This is
where it is being displayed on the serial terminal */

        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( 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);

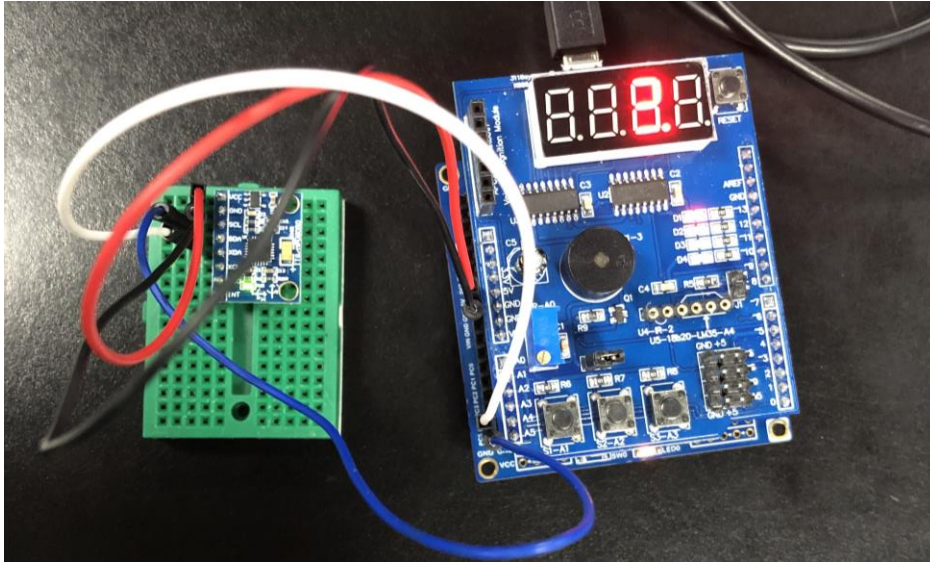
        dtostrf( Zg, 3, 2, float_ );
        sprintf(buffer," Gz = %s%c/s\r\n",float_,0xF8);
        USART_SendString(buffer);
        _delay_ms(1000);
    }
}

```

3. SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)



4. SCREENSHOT OF EACH DEMO (BOARD SETUP)



5. VIDEO LINKS OF EACH DEMO

<https://youtu.be/eIU17CZyB4>

6. GITHUB LINK OF THIS DA

https://github.com/reedjacobp/submission_da/tree/master/DesignAssignments/DA6

Student Academic Misconduct Policy

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

"This assignment submission is my own, original work".

Jacob Reed