

Use M451 I2C to Drive MPU6050

Example Code Introduction for 32-bit NuMicro® Family

Information

Application	This sample code is M451 drives and controls six-axis sensor MPU6050 through I2C interface
BSP Version	M451 Series BSP CMSIS v3.01.003
Hardware	SmartM-M451 Mini

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

*Nuvoton is providing this document only for reference purposes of NuMicro microcontroller based system design.
Nuvoton assumes no responsibility for errors or omissions.*

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

1 Function Description

The sample code uses M451 GPIO to simulate I2C to initialize 6-axis sensor, and reads 6-axis sensor data through I2C. The M451 sends data to PC through UART, so that the position change of 6-axis sensor can be seen intuitively in PC software. The 6-axis sensor used in this code is InvenSense MPU 6050, which communicates with I2C protocol. Initialization and control interface functions can be found in the MPU6050.c function prototype, sensor settings and data acquisition refer to the specifications of MPU6050.

2 Code Description

Initialize MPU6050 and check if the sensor ID is correct :

```
UINT8 MPU_Init(void)
{
    UINT8 res;
    IIC_Init();
    MPU_Write_Byte(MPU_PWR_MGMT1_REG,0X80);
    Delayms(100);
    MPU_Write_Byte(MPU_PWR_MGMT1_REG,0X00);
    MPU_Set_Gyro_Fsr(3);
    MPU_Set_Accel_Fsr(0);
    MPU_Set_Rate(50);
    MPU_Write_Byte(MPU_INT_EN_REG,0X00);
    MPU_Write_Byte(MPU_USER_CTRL_REG,0X00);
    MPU_Write_Byte(MPU_FIFO_EN_REG,0X00);
    MPU_Write_Byte(MPU_INTBP_CFG_REG,0X80);
    res=MPU_Read_Byte(MPU_DEVICE_ID_REG);
    printf("RES=%x\r\n",res);
    if(res==MPU_ADDR)
    {
        MPU_Write_Byte(MPU_PWR_MGMT1_REG,0X01);
        MPU_Write_Byte(MPU_PWR_MGMT2_REG,0X00);
        MPU_Set_Rate(50);
    }else return 1;
    return 0;
}
```

The data acquisition program for MPU 6050 is as follows :

```
u8 mpu_dmp_get_data(float *pitch,float *roll,float *yaw)
{
    float q0=1.0f,q1=0.0f,q2=0.0f,q3=0.0f;
    unsigned long sensor_timestamp;
    short gyro[3], accel[3], sensors;
    unsigned char more;
    long quat[4];
    if(dmp_read_fifo(gyro, accel, quat, &sensor_timestamp, &sensors,&more))return 1;
    if(sensors&INV_WXYZ_QUAT)
    {
```

```
q0 = quat[0] / q30;
q1 = quat[1] / q30;
q2 = quat[2] / q30;
q3 = quat[3] / q30;
//calculator angle of pitch /roll angle / yaw angle
*pitch = asin(-2 * q1 * q3 + 2 * q0* q2)* 57.3; // pitch
*roll = atan2(2 * q2 * q3 + 2 * q0 * q1, -2 * q1 * q1 - 2 * q2* q2 + 1)* 57.3;
// roll
*yaw = atan2(2*(q1*q2 + q0*q3),q0*q0+q1*q1-q2*q2-q3*q3) * 57.3; //yaw
}else return 2;
return 0;
}
```

The code for uploading data to PC through serial port is as follows :

```
void Uart0ReportImu(short aacx,short aacy,short aacz,short gyrox,short gyroy,short
gyroz,short roll,short pitch,short yaw)
{
    u8 tbuf[28];
    u8 i;
    for(i=0;i<28;i++)tbuf[i]=0;
    tbuf[0]=(aacx>>8)&0xFF;
    tbuf[1]=aacx&0xFF;
    tbuf[2]=(aacy>>8)&0xFF;
    tbuf[3]=aacy&0xFF;
    tbuf[4]=(aacz>>8)&0xFF;
    tbuf[5]=aacz&0xFF;
    tbuf[6]=(gyrox>>8)&0xFF;
    tbuf[7]=gyrox&0xFF;
    tbuf[8]=(gyroy>>8)&0xFF;
    tbuf[9]=gyroy&0xFF;
    tbuf[10]=(gyroz>>8)&0xFF;
    tbuf[11]=gyroz&0xFF;
    tbuf[18]=(roll>>8)&0xFF;
    tbuf[19]=roll&0xFF;
    tbuf[20]=(pitch>>8)&0xFF;
    tbuf[21]=pitch&0xFF;
    tbuf[22]=(yaw>>8)&0xFF;
    tbuf[23]=yaw&0xFF;
    Uart0NimingReport(0xAF,tbuf,28);
}
```

```
void Uart0NimingReport(u8 fun,u8*data,u8 len)
{
    u8 send_buf[32];
    u8 i;
    if(len>28) return;
    send_buf[len+3]=0;
    send_buf[0]=0X88;
    send_buf[1]=fun;
    send_buf[2]=len;
    for(i=0;i<len;i++)send_buf[3+i]=data[i];
    for(i=0;i<len+3;i++)send_buf[len+3]+=send_buf[i];
    for(i=0;i<len+4;i++)UART_Write(UART0,&send_buf[i],1);
}
```

3 Software and Hardware Environment

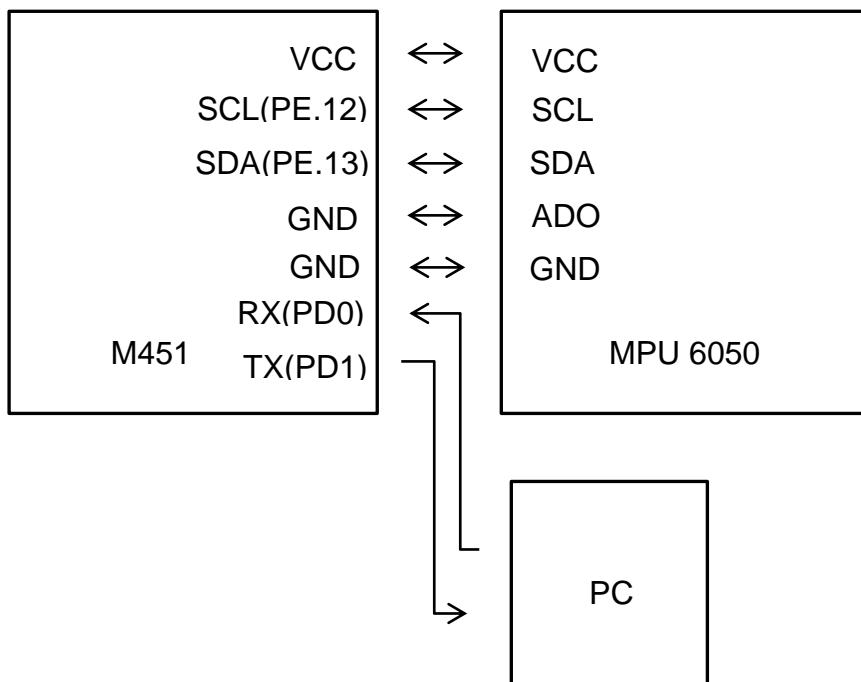
● Software Environment

- BSP Version
 - ◆ M451 Series BSP CMSIS v3.01.003
- IDE Version
 - ◆ Keil uVersion 4.6

● Hardware Environment

- Circuit Components
 - ◆ SmartM-M451 Mini
 - ◆ MPU6050
- Diagram

M451 transmits control commands through I2C to MPU6050 sensor, and UART0 to report data to PC .



4 Directory Information

 EC_M451_Six-axis_Sensor_MP6050_Driver_V1.00	
 Library	Sample code header and source files
 CMSIS	Cortex® Microcontroller Software Interface Standard (CMSIS) by Arm® Corp.
 Device	CMSIS compliant device header file
 StdDriver	All peripheral driver header and source files
 SampleCode	
 ExampleCode	Source file of example code

5 How to Execute Example Code

1. According to Directory Information Entry the KEIL folder in path ExampleCode · double click“M451_Six-axis_Sensor_MPU6050_Driver. uvproj”.
2. Enter the compilation mode interface
 - a. Compilation
 - b. Download code to Flash
 - c. Enter/leave debugging mode
3. Enter the debugging mode interface
 - a. Execution code

6 Revision History

Date	Revision	Description
Sep.25, 2019	1.00	1. Initially issued.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

Please note that all data and specifications are subject to change without notice.
All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.