

ATK-MPU6050 六轴传感器模块 用户手册

高性能三轴加速度+三轴陀螺仪模块

ALIENTEK
广州市星翼电子科技有限公司

修订历史

版本	日期	原因
V1.00	2015/4/18	第一次发布

目 录

1. 特性参数.....	1
2. 使用说明.....	2
2.1 模块引脚说明.....	2
2.2 MPU6050 简介.....	3
2.3 模块重要寄存器简介.....	5
2.4 DMP 使用简介.....	7
3. 结构尺寸.....	10
4. 其他.....	11

1. 特性参数

ATK-MPU6050-V1.1（V1.1 是版本号，下面均以 ATK-MPU6050 表示该产品）是 ALIENTEK 推出的一款高性能三轴加速度+三轴陀螺仪的六轴传感器模块。该模块采用 InvenSense 公司的 MPU6050 芯片作为核心，该芯片内部整合了 3 轴陀螺仪和 3 轴加速度传感器，并可利用自带的数字运动处理器（DMP: Digital Motion Processor）硬件加速引擎，通过主 IIC 接口，向应用端输出姿态解算后的数据。有了 DMP，我们可以使用 InvenSense 公司提供的运动处理资料库，非常方便的实现姿态解算，降低了运动处理运算对操作系统的负荷，同时大大降低了开发难度。

ATK-MPU6050 模块具有：体积小、自带 DMP、自带温度传感器、支持 IIC 从机地址设置和中断、兼容 3.3V/5V 系统、使用方便等特点。

ATK-MPU6050 模块各项参数如表 1.1 和表 1.2 所示。

项目	说明
接口特性	3.3V/5V
通信接口	IIC 接口
通信速率	400Khz (Max)
测量维度	加速度：3 维 陀螺仪：3 维
加速度测量范围	$\pm 2/\pm 4/\pm 8/\pm 16g$
陀螺仪测量范围	$\pm 250/\pm 500/\pm 1000/\pm 2000^{\circ}/\text{秒}$
ADC 位数	16 位
分辨率	加速度：16384LSB/g(Max) 陀螺仪：131LSB/($^{\circ}$ /s)(Max)
输出速率	加速度：1Khz (Max) 陀螺仪：8Khz (Max)
姿态解算输出速率	200Hz (Max)
温度传感器测量范围	-40 $^{\circ}\text{C}$ ~85 $^{\circ}\text{C}$
温度传感器精度	$\pm 1^{\circ}\text{C}$
工作温度	-40 $^{\circ}\text{C}$ ~85 $^{\circ}\text{C}$
模块尺寸	16mm*18mm

表 1.1 ATK-MPU6050 六轴传感器模块基本特性

项目	说明
电源电压	3.3V/5V
IO 口电平 ¹	3.3V LVTTL
功耗	5mA

表 1.2 ATK-MPU6050 六轴传感器模块电气特性

注 1：模块 IO 电压是 3.3V，不过我们做了 5V 兼容性处理（串 120R 电阻），所以也可以直接连接 5V 的 MCU 使用。

2. 使用说明

2.1 模块引脚说明

ATK-MPU6050 六轴传感器模块通过 1*6 的排针（2.54mm 间距）同外部连接，模块可以与 ALIENTEK 战舰 STM32F103 V3、精英 STM32F103、探索者 STM32F407 等开发板直接对接（插 ATK-MODULE 接口），而 ALIENTEK MiniSTM32F103 开发板则可以通过杜邦线连接模块进行测试。所有 ALIENTEK STM32 开发板都提供有相应例程，用户可以直接在这些开发板上，对模块进行测试。

ATK-MPU6050 六轴传感器模块外观如图 2.1.1 所示：



图 2.1.1 ATK-MPU6050 六轴传感器模块实物图

ATK-MPU6050 模块原理图如图 2.1.2 所示：

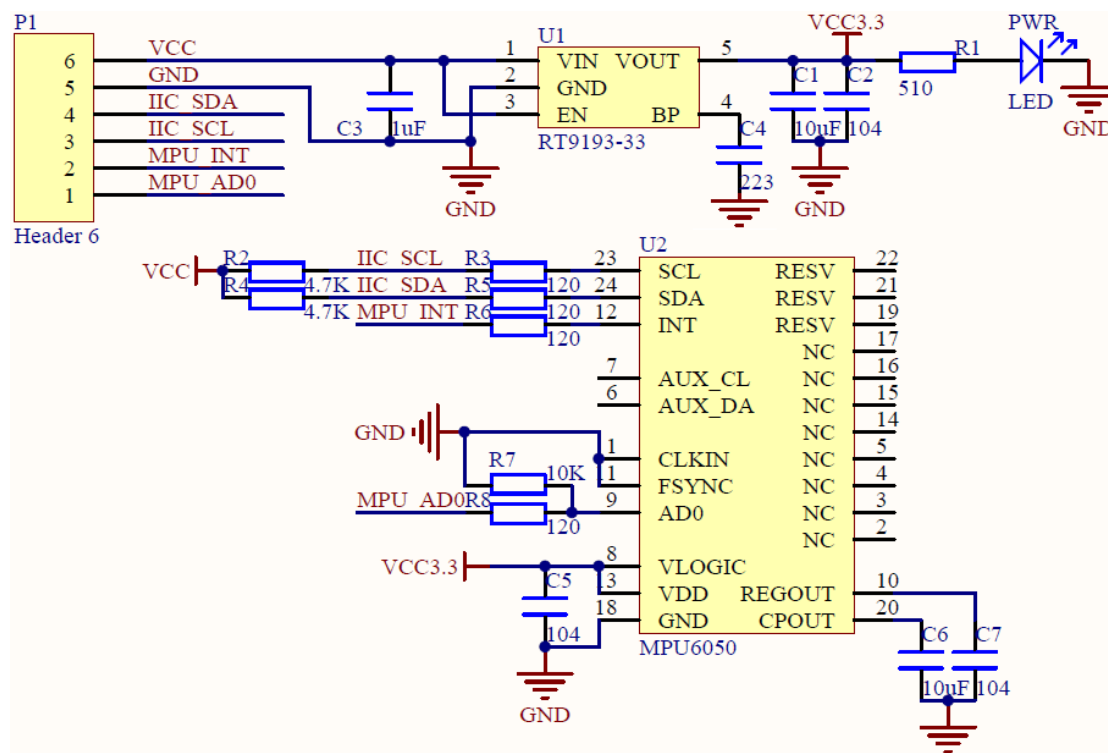


图 2.1.2 ATK-MPU6050 六轴传感器模块原理图

从图 2.1.2 可以看出，模块自带了 3.3V 超低压差稳压芯片，给 MPU6050 供电，因此外部供电可以选择：3.3V / 5V 都可以。模块通过 P1 排针与外部连接，引出了 VCC、GND、IIC_SDA、IIC_SCL、MPU_INT 和 MPU_AD0 等信号，其中，IIC_SDA 和 IIC_SCL 带了 4.7K 上拉电阻，外部可以不用再加上拉电阻了，另外 MPU_AD0 自带了 10K 下拉电阻，当 AD0 悬空时，默认 IIC 地址为 (0X68)。

ATK-MPU6050 六轴传感器模块通过一个 1*6 的排针 (P1) 同外部电路连接，各引脚的详细描述如表 2.1.1 所示：

序号	名称	说明
1	VCC	3.3V/5V 电源输入
2	GND	地线
3	IIC_SDA	IIC 通信数据线
4	IIC_SCL	IIC 通信时钟线
5	MPU_INT	中断输出引脚
6	MPU_AD0	IIC 从机地址设置引脚； ID: 0X68(悬空/接 GND) ID: 0X69(接 VCC)

表 2.1.1 ATK-MPU6050 六轴传感器模块引脚说明

模块仅通过一个 IIC 接口与外部通信，并可以通过 MPU_AD0 设置模块的 IIC 地址，当 MPU_AD0 悬空/接 GND 的时候，模块的 IIC 从机地址为：0X68；当 MPU_AD0 接 VCC 的时候，模块的 IIC 从机地址为：0X69。

2.2 MPU6050 简介

MPU6050 是 InvenSense 公司推出的全球首款整合性 6 轴运动处理组件，相较于多组件方案，免除了组合陀螺仪与加速器时之轴间差的问题，减少了安装空间。

MPU6050 内部整合了 3 轴陀螺仪和 3 轴加速度传感器，并且含有一个第二 IIC 接口 (本模块未引出)，可用于连接外部磁力传感器，并利用自带的数字运动处理器 (DMP: Digital Motion Processor) 硬件加速引擎，通过主 IIC 接口，向应用端输出完整的 9 轴融合演算数据。有了 DMP，我们可以使用 InvenSense 公司提供的运动处理资料库，非常方便的实现姿态解算，降低了运动处理运算对操作系统的负荷，同时大大降低了开发难度。

MPU6050 的特点包括：

- ① 以数字形式输出 6 轴或 9 轴 (需外接磁传感器) 的旋转矩阵、四元数(quaternion)、欧拉角格式(Euler Angle forma)的融合演算数据 (需 DMP 支持)
- ② 具有 131 LSBs/° /sec 敏感度与全格感测范围为±250、±500、±1000 与±2000° /sec 的 3 轴角速度感测器(陀螺仪)
- ③ 集成可程序控制，范围为±2g、±4g、±8g 和±16g 的 3 轴加速度传感器
- ④ 移除加速器与陀螺仪轴间敏感度，降低设定给予的影响与感测器的飘移
- ⑤ 自带数字运动处理(DMP: Digital Motion Processing)引擎可减少 MCU 复杂的融合演算数据、感测器同步化、姿势感应等的负荷
- ⑥ 内建运作时间偏差与磁力感测器校正演算技术，免除了客户须另外进行校正的需求
- ⑦ 自带一个数字温度传感器
- ⑧ 带数字输入同步引脚(Sync pin)支持视频电子影相稳定技术与 GPS
- ⑨ 可程序控制的中断(interrupt)，支持姿势识别、摇摄、画面放大缩小、滚动、快速下降中断、high-G 中断、零动作感应、触击感应、摇动感应功能
- ⑩ VDD 供电电压为 2.5V±5%、3.0V±5%、3.3V±5%；VLOGIC 可低至 1.8V± 5%

- ⑪ 陀螺仪工作电流：5mA，陀螺仪待机电流：5uA；加速器工作电流：500uA，加速器省电模式电流：40uA@10Hz
- ⑫ 自带 1024 字节 FIFO，有助于降低系统功耗
- ⑬ 高达 400Khz 的 IIC 通信接口
- ⑭ 超小封装尺寸：4x4x0.9mm（QFN）

MPU6050 传感器的检测轴如图 2.2.1 所示：

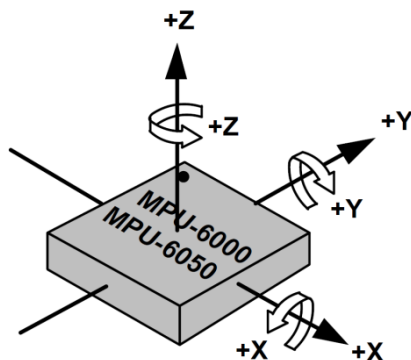


图 2.2.1 MPU6050 检测轴及其方向

MPU6050 的内部框图如图 2.2.2 所示：

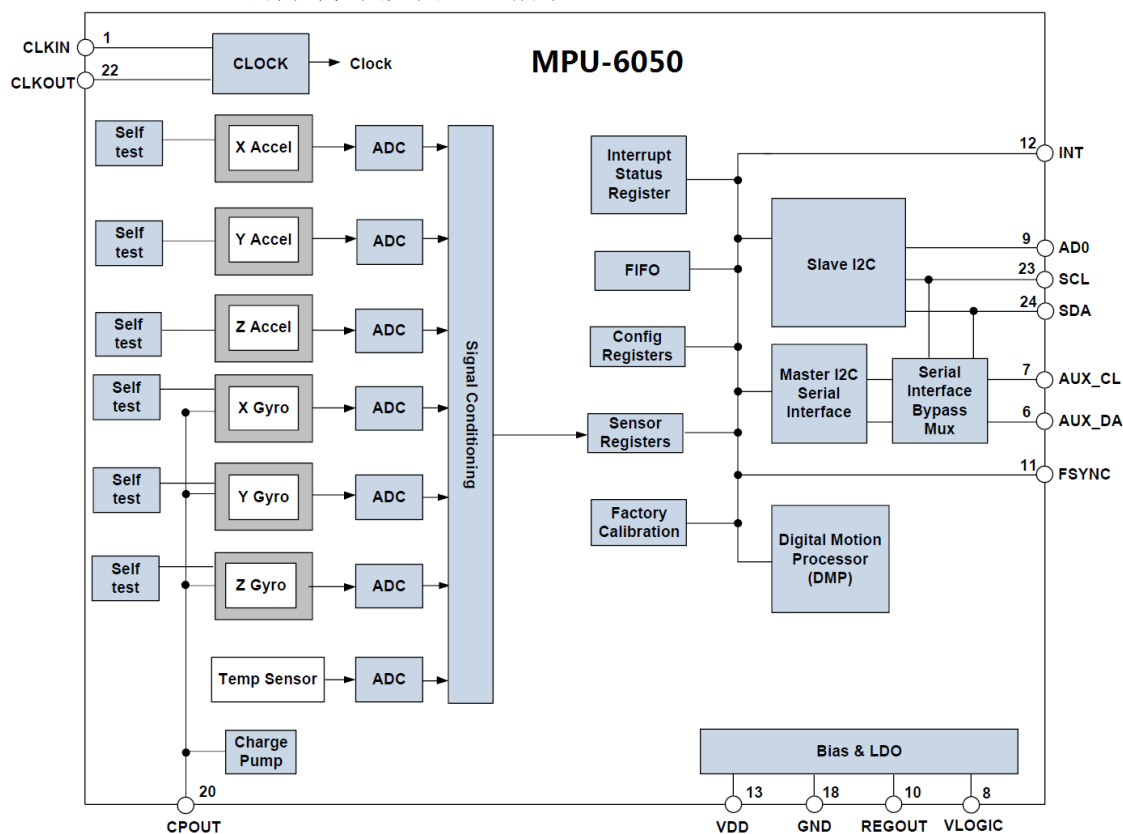


图 2.2.2 MPU6050 框图

其中，SCL 和 SDA 是连接 MCU 的 IIC 接口，MCU 通过这个 IIC 接口来控制 MPU6050，另外还有一个 IIC 接口：AUX_CL 和 AUX_DA，这个接口可用来连接外部从设备，比如磁传感器，这样就可以组成一个九轴传感器。VLOGIC 是 IO 口电压，该引脚最低可以到 1.8V，我们一般直接接 VDD 即可。AD0 是从 IIC 接口（接 MCU）的地址控制引脚，该引脚控制 IIC 地址的最低位。如果接 GND，则 MPU6050 的 IIC 地址是：0X68，如果接 VDD，则是 0X69，注意：这里的地址是不包含数据传输的最低位的（最低位用来表示读写）！！

2.3 模块重要寄存器简介

MPU6050 内部总共有好几十个寄存器，这里，我们仅介绍 MPU6050 使用时的几个重要寄存器，其他的请大家参考：MPU-6000 and MPU-6050 Register Map and Descriptions.pdf 这个文档。

首先，我们介绍电源管理寄存器 1，该寄存器地址为 0X6B，各位描述如图 2.3.1 所示：

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
6B	107	DEVICE_RESET	SLEEP	CYCLE	-	TEMP_DIS	CLKSEL[2:0]		

图 2.3.1 电源管理寄存器 1 各位描述

其中，DEVICE_RESET 位用来控制复位，设置为 1，复位 MPU6050，复位结束后，MPU 硬件自动清零该位。SLEEP 位用于控制 MPU6050 的工作模式，复位后，该位为 1，即进入了睡眠模式（低功耗），所以我们要清零该位，以进入正常工作模式。TEMP_DIS 用于设置是否使能温度传感器，设置为 0，则使能。最后 CLKSEL[2:0]用于选择系统时钟源，选择关系如表 2.3.1 所示：

CLKSEL[2:0]	时钟源
000	内部 8M RC 晶振
001	PLL，使用 X 轴陀螺作为参考
010	PLL，使用 Y 轴陀螺作为参考
011	PLL，使用 Z 轴陀螺作为参考
100	PLL，使用外部 32.768Khz 作为参考
101	PLL，使用外部 19.2Mhz 作为参考
110	保留
111	关闭时钟，保持时序产生电路复位状态

表 2.3.1 CLKSEL 选择列表

默认是使用内部 8M RC 晶振的，精度不高，所以我们一般选择 X/Y/Z 轴陀螺作为参考的 PLL 作为时钟源，一般设置 CLKSEL=001 即可。

接着，我们看陀螺仪配置寄存器，该寄存器地址为：0X1B，各位描述如图 2.3.2 所示：

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1B	27	XG_ST	YG_ST	ZG_ST	FS_SEL[1:0]		-	-	-

图 2.3.2 陀螺仪配置寄存器各位描述

该寄存器我们只关心 FS_SEL[1:0]这两个位，用于设置陀螺仪的满量程范围：0， $\pm 250^{\circ}/S$ ；1， $\pm 500^{\circ}/S$ ；2， $\pm 1000^{\circ}/S$ ；3， $\pm 2000^{\circ}/S$ ；我们一般设置为 3，即 $\pm 2000^{\circ}/S$ ，因为陀螺仪的 ADC 为 16 位分辨率，所以得到灵敏度为： $65536/4000=16.4LSB/(^{\circ}/S)$ 。

接下来，我们看加速度传感器配置寄存器，寄存器地址为：0X1C，各位描述如图 2.3.3 所示：

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1C	28	XA_ST	YA_ST	ZA_ST	AFS_SEL[1:0]		-		

图 2.3.3 加速度传感器配置寄存器各位描述

该寄存器我们只关心 AFS_SEL[1:0]这两个位，用于设置加速度传感器的满量程范围：0， $\pm 2g$ ；1， $\pm 4g$ ；2， $\pm 8g$ ；3， $\pm 16g$ ；我们一般设置为 0，即 $\pm 2g$ ，因为加速度传感器的 ADC 也是 16 位，所以得到灵敏度为： $65536/4=16384LSB/g$ 。

接下来，我看看 FIFO 使能寄存器，寄存器地址为：0X1C，各位描述如图 2.3.4 所示：

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
23	35	TEMP_FIFO_EN	XG_FIFO_EN	YG_FIFO_EN	ZG_FIFO_EN	ACCEL_FIFO_EN	SLV2_FIFO_EN	SLV1_FIFO_EN	SLV0_FIFO_EN

图 2.3.4 FIFO 使能寄存器各位描述

该寄存器用于控制 FIFO 使能，在简单读取传感器数据的时候，可以不用 FIFO，设置对应位为 0 即可禁止 FIFO，设置为 1，则使能 FIFO。注意加速度传感器的 3 个轴，全由 1 个位（ACCEL_FIFO_EN）控制，只要该位置 1，则加速度传感器的三个通道都开启 FIFO 了。

接下来，我们看陀螺仪采样率分频寄存器，寄存器地址为：0X19，各位描述如图 2.3.5 所示：

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
19	25	SMPLRT_DIV[7:0]							

图 2.3.5 陀螺仪采样率分频寄存器各位描述

该寄存器用于设置 MPU6050 的陀螺仪采样频率，计算公式为：

$$\text{采样频率} = \text{陀螺仪输出频率} / (1 + \text{SMPLRT_DIV})$$

这里陀螺仪的输出频率，是 1KHz 或者 8KHz，与数字低通滤波器（DLPF）的设置有关，当 DLPF_CFG=0/7 的时候，频率为 8KHz，其他情况是 1KHz。而且 DLPF 滤波频率一般设置为采样率的一半。采样率，我们假定设置为 50Hz，那么 SMPLRT_DIV=1000/50-1=19。

接下来，我们看配置寄存器，寄存器地址为：0X1A，各位描述如图 2.3.6 所示：

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1A	26	-	-	EXT_SYNC_SET[2:0]			DLPF_CFG[2:0]		

图 2.3.6 配置寄存器各位描述

这里，我们主要关心数字低通滤波器（DLPF）的设置位，即：DLPF_CFG[2:0]，加速度计和陀螺仪，都是根据这三个位的配置进行过滤的。DLPF_CFG 不同配置对应的过滤情况如表 2.3.2 所示：

DLPF_CFG[2:0]	加速度传感器 Fs=1KHz		角速度传感器 (陀螺仪)		
	带宽 (Hz)	延迟 (ms)	带宽 (Hz)	延迟 (ms)	Fs (KHz)
000	260	0	256	0.98	8
001	184	2.0	188	1.9	1
010	94	3.0	98	2.8	1
011	44	4.9	42	4.8	1
100	21	8.5	20	8.3	1
101	10	13.8	10	13.4	1
110	5	19.0	5	18.6	1
111	保留		保留		8

图 2.3.2 DLPF_CFG 配置表

这里的加速度传感器，输出速率（Fs）固定是 1KHz，而角速度传感器的输出速率（Fs），则根据 DLPF_CFG 的配置有所不同。一般我们设置角速度传感器的带宽为其采样率的一半，如前面所说的，如果设置采样率为 50Hz，那么带宽就应该设置为 25Hz，取近似值 20Hz，就应该设置 DLPF_CFG=100。

接下来，我们看电源管理寄存器 2，寄存器地址为：0X6C，各位描述如图 2.3.7 所示：

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
6C	108	LP_WAKE_CTRL[1:0]		STBY_XA	STBY_YA	STBY_ZA	STBY_XG	STBY_YG	STBY_ZG

图 2.3.7 电源管理寄存器 2 各位描述

该寄存器的 LP_WAKE_CTRL 用于控制低功耗时的唤醒频率。剩下的 6 位，分别控制加速度和陀螺仪的 x/y/z 轴是否进入待机模式，这里我们全部都不进入待机模式，所以全部设置为 0 即可。

接下来，我们看看陀螺仪数据输出寄存器，总共有 8 个寄存器组成，地址为：0X43~0X48，通过读取这 8 个寄存器，就可以读到陀螺仪 x/y/z 轴的值，比如 x 轴的数据，可以通过读取 0X43（高 8 位）和 0X44（低 8 位）寄存器得到，其他轴以此类推。

同样，加速度传感器数据输出寄存器，也有 8 个，地址为：0X3B~0X40，通过读取这 8 个寄存器，就可以读到加速度传感器 x/y/z 轴的值，比如读 x 轴的数据，可以通过读取 0X3B（高 8 位）和 0X3C（低 8 位）寄存器得到，其他轴以此类推。

最后，温度传感器的值，可以通过读取 0X41（高 8 位）和 0X42（低 8 位）寄存器得到，温度换算公式为：

$$\text{Temperature} = 36.53 + \text{regval}/340$$

其中，Temperature 为计算得到的温度值，单位为℃，regval 为从 0X41 和 0X42 读到的温度传感器值。

关于 MPU6050 的寄存器简介，我们就介绍到这。MPU6050 的详细资料和相关寄存器介绍，请参考模块资料：4，MPU6050 参考资料→ MPU-6000 and MPU-6050 Product Specification.pdf 和 MPU-6000 and MPU-6050 Register Map and Descriptions.pdf 这两个文档，另外该目录还提供了部分 MPU6050 的中文资料，供大家参考学习。

2.4 DMP 使用简介

经过 2.3 节的介绍，我们可以读出 MPU6050 的加速度传感器和角速度传感器的原始数据。不过这些原始数据，对想搞四轴之类的初学者来说，用处不大，我们期望得到的是姿态数据，也就是欧拉角：航向角（yaw）、横滚角（roll）和俯仰角（pitch）。有了这三个角，我们就可以得到当前四轴的姿态，这才是我们想要的结果。

要得到欧拉角数据，就得利用我们的原始数据，进行姿态融合解算，这个比较复杂，知识点比较多，初学者不易掌握。而 MPU6050 自带了数字运动处理器，即 DMP，并且，InvenSense 提供了一个 MPU6050 的嵌入式运动驱动库，结合 MPU6050 的 DMP，可以将我们的原始数据，直接转换成四元数输出，而得到四元数之后，就可以很方便的计算出欧拉角，从而得到 yaw、roll 和 pitch。

使用内置的 DMP，大大简化了四轴的代码设计，且 MCU 不用进行姿态解算过程，大大降低了 MCU 的负担，从而有更多的时间去处理其他事件，提高系统实时性。

使用 MPU6050 的 DMP 输出的四元数是 q30 格式的，也就是浮点数放大了 2 的 30 次方倍。在换算成欧拉角之前，必须先将其转换为浮点数，也就是除以 2 的 30 次方，然后再进行计算，计算公式为：

```
q0=quat[0] / q30; //q30 格式转换为浮点数
q1=quat[1] / q30;
q2=quat[2] / q30;
q3=quat[3] / q30;
//计算得到俯仰角/横滚角/航向角
```

```
pitch=asin(-2 * q1 * q3 + 2 * q0*q2)* 57.3;           //俯仰角
roll=atan2(2 * q2 * q3 + 2 * q0 * q1, -2 * q1 * q1 - 2 * q2*q2 + 1)* 57.3; //横滚角
yaw=atan2(2*(q1*q2 + q0*q3),q0*q0+q1*q1-q2*q2-q3*q3) * 57.3; //航向角
```

其中 quat[0]~ quat[3]是 MPU6050 的 DMP 解算后的四元数, q30 格式, 所以要除以一个 2 的 30 次方, 其中 q30 是一个常量: 1073741824, 即 2 的 30 次方, 然后带入公式, 计算出欧拉角。上述计算公式的 57.3 是弧度转换为角度, 即 $180/\pi$, 这样得到的结果就是以度($^{\circ}$) 为单位的。关于四元数与欧拉角的公式推导, 这里我们不进行讲解, 感兴趣的朋友, 可以自行查阅相关资料学习。

InvenSense 提供的 MPU6050 运动驱动库是基于 MSP430 的, 我们需要将其移植一下, 才可以用到 STM32F1 上面, 官方原版驱动在模块资料: 4, MPU6050 参考资料→DMP 资料→Embedded_MotionDriver_5.1.rar, 这就是官方原版的驱动, 代码比较多, 不过官方提供了两个资料供大家学习: Embedded Motion Driver V5.1.1 API 说明.pdf 和 Embedded Motion Driver V5.1.1 教程.pdf, 这两个文件都在 DMP 资料文件夹里面, 大家可以阅读这两个文件, 来熟悉官方驱动库的使用。

官方 DMP 驱动库移植起来, 还是比较简单的, 主要是实现这 4 个函数: i2c_write, i2c_read, delay_ms 和 get_ms, 具体细节, 我们就不详细介绍了, 移植后的驱动代码, 可以在 ALIENTEK MPU6050 六轴传感器实验→HARDWARE→MPU6050→eMPL 文件夹内找到, 总共 6 个文件, 如图 2.4.1 所示:

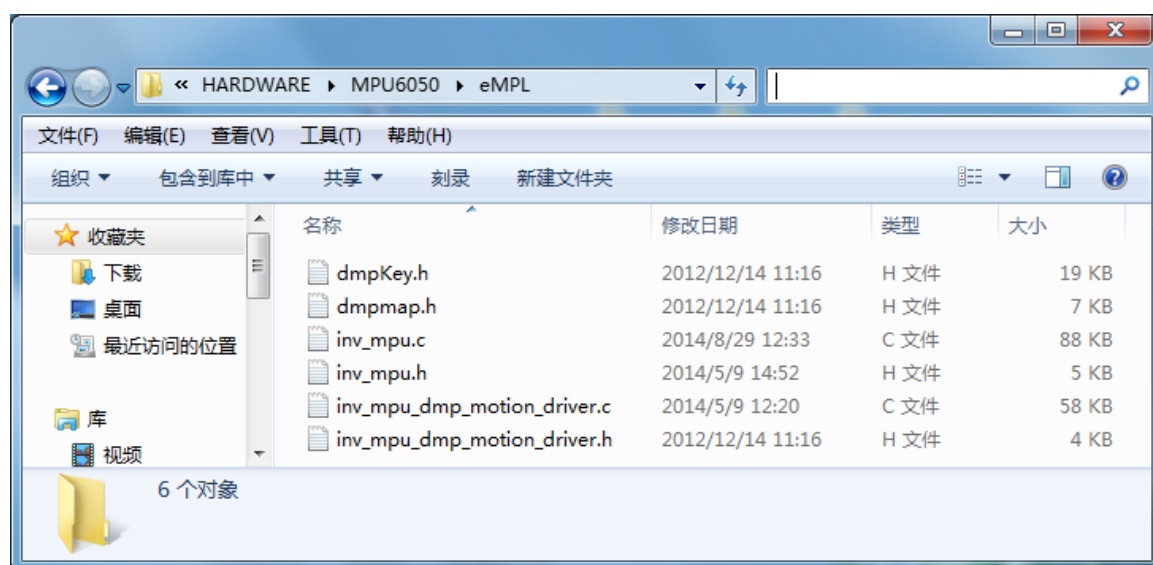


图 2.4.1 移植后的驱动库代码

该驱动库, 重点就是两个 c 文件: inv_mpu.c 和 inv_mpu_dmp_motion_driver.c。其中我们在 inv_mpu.c 添加了几个函数, 方便我们使用, 重点是两个函数: mpu_dmp_init 和 mpu_dmp_get_data 这两个函数, 这里我们简单介绍下这两个函数。

mpu_dmp_init, 是 MPU6050 DMP 初始化函数, 该函数代码如下:

```
//mpu6050,dmp 初始化
//返回值:0,正常
// 其他,失败
u8 mpu_dmp_init(void)
{
    u8 res=0;
    IIC_Init(); //初始化 IIC 总线
```

```
if(mpu_init()==0) //初始化 MPU6050
{
    res=mpu_set_sensors(INV_XYZ_GYRO|INV_XYZ_ACCEL);//设需要的传感器
    if(res)return 1;
    res=mpu_configure_fifo(INV_XYZ_GYRO|INV_XYZ_ACCEL);//设置 FIFO
    if(res)return 2;
    res=mpu_set_sample_rate(DEFAULT_MPU_HZ);    //设置采样率
    if(res)return 3;
    res=dmp_load_motion_driver_firmware();        //加载 dmp 固件
    if(res)return 4;
    res=dmp_set_orientation(inv_orientation_matrix_to_scalar(gyro_orientation));
    //设置陀螺仪方向
    if(res)return 5;
    res=dmp_enable_feature(DMP_FEATURE_6X_LP_QUAT|DMP_FEATURE_TAP
    |DMP_FEATURE_ANDROID_ORIENT|DMP_FEATURE_SEND_RAW_ACCEL
    |DMP_FEATURE_SEND_CAL_GYRO|DMP_FEATURE_GYRO_CAL);
    //设置 dmp 功能
    if(res)return 6;
    res=dmp_set_fifo_rate(DEFAULT_MPU_HZ);//设置 DMP 输出速率(最大 200Hz)
    if(res)return 7;
    res=run_self_test();        //自检
    if(res)return 8;
    res=mpu_set_dmp_state(1); //使能 DMP
    if(res)return 9;
}
return 0;
}
```

此函数首先通过 IIC_Init(需外部提供)初始化与 MPU6050 连接的 IIC 接口，然后调用 mpu_init 函数，初始化 MPU6050，之后就是设置 DMP 所用传感器、FIFO、采样率和加载固件等一系列操作，在所有操作都正常之后，最后通过 mpu_set_dmp_state(1)使能 DMP 功能，在使能成功以后，我们便可以通过 mpu_dmp_get_data 来读取姿态解算后的数据了。

mpu_dmp_get_data 函数代码如下：

```
//得到 dmp 处理后的数据(注意,本函数需要比较多堆栈,局部变量有点多)
//pitch:俯仰角 精度:0.1° 范围:-90.0° <---> +90.0°
//roll:横滚角 精度:0.1° 范围:-180.0° <---> +180.0°
//yaw:航向角 精度:0.1° 范围:-180.0° <---> +180.0°
//返回值:0,正常
// 其他,失败
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; //q30 格式转换为浮点数
    q1 = quat[1] / q30;
    q2 = quat[2] / q30;
    q3 = quat[3] / q30;
    //计算得到俯仰角/横滚角/航向角
    *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;
}

```

此函数用于得到 DMP 姿态解算后的俯仰角、横滚角和航向角。这里就用到了我们前面介绍的四元数转欧拉角公式，将 dmp_read_fifo 函数读到的 q30 格式四元数转换成欧拉角。

利用这两个函数，我们就可以读取到姿态解算后的欧拉角，使用非常方便。DMP 部分，我们就介绍到这。

3. 结构尺寸

ATK-MPU6050 六轴传感器模块的尺寸结构如图 3.1 所示：

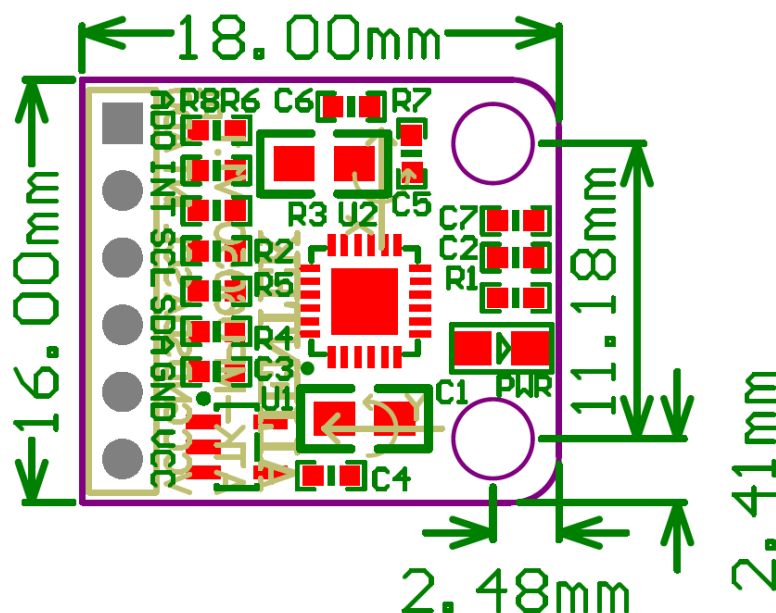


图 3.1 ATK-MPU6050 六轴传感器模块尺寸图

4. 其他

1、购买地址:

官方店铺 1: <http://eboard.taobao.com/>

官方店铺 2: <http://shop62103354.taobao.com>

2、资料下载

ATK-MPU6050 模块资料下载地址: <http://www.openedv.com/posts/list/51289.htm>

3、技术支持

公司网址: www.alientek.com

技术论坛: www.openedv.com

传真: 020-36773971

电话: 020-38271790

