目录

[第九章 MPU6050介绍 2](#_Toc20881)

[9.1 模块了解（特性，功能，作用） 2](#_Toc20808)

[9.1.1 简介： 2](#_Toc31987)

[9.2 加速度和角速度 2](#_Toc5360)

[9.2 模块硬件分析 2](#_Toc27050)

[9.2.1 引脚分布： 2](#_Toc3162)

[9.2.2 器件地址： 2](#_Toc7394)

[9.3 模块软件分析 2](#_Toc12271)

[9.3.1 时序图 2](#_Toc23944)

[9.3.2 步骤 3](#_Toc17005)

# 第九章 MPU6050介绍

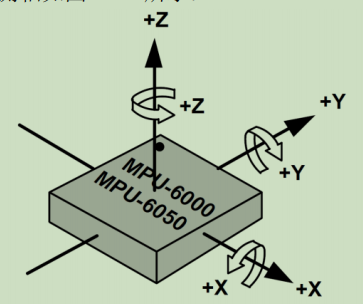
## 9.1 模块了解（特性，功能，作用）

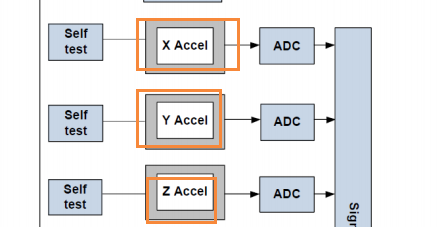
### 9.1.1 简介：



MPU-60X0 是全球首例 9 轴运动处理传感器。它集成了 3 轴 MEMS 陀螺仪，3 轴 MEMS加速度计，以及一个可扩展的数字运动处理器 DMP（ Digital Motion Processor），可用 I2C接口连接一个第三方的数字传感器，比如磁力计。扩展之后就可以通过其 I2C 或 SPI 接口输出一个 9 轴的信号（ SPI 接口仅在 MPU-6000 可用）。 MPU-60X0 也可以通过其 I2C 接口连接非惯性的数字传感器，比如压力传感器。MPU-60X0 对陀螺仪和加速度计分别用了三个 16 位的 ADC，将其测量的模拟量转化为可输出的数字量。为了精确跟踪快速和慢速的运动，传感器的测量范围都是用户可控的，陀螺仪可测范围为±250， ±500， ±1000， ±2000°/秒（ dps），加速度计可测范围为±2， ±4，±8， ±16g。一个片上 1024 字节的 FIFO，有助于降低系统功耗。和所有设备寄存器之间的通信采用 400kHz 的 I2C 接口或 1MHz 的 SPI 接口（ SPI 仅MPU-6000 可用）。 对于需要高速传输的应用， 对寄存器的读取和中断可用 20MHz 的 SPI。

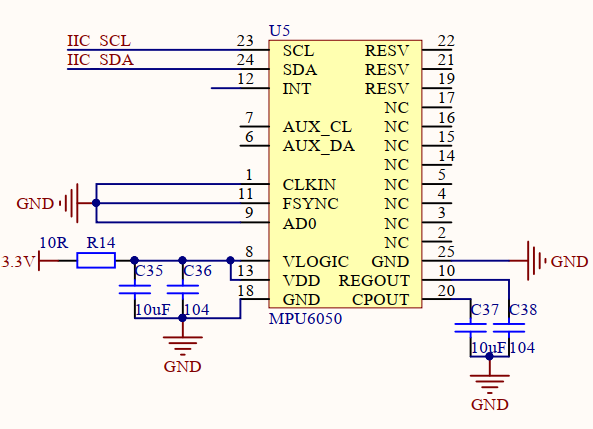
### 9.2 加速度和角速度





## 9.2 模块硬件分析

### 9.2.1 引脚分布：



### 9.2.2 器件地址：

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

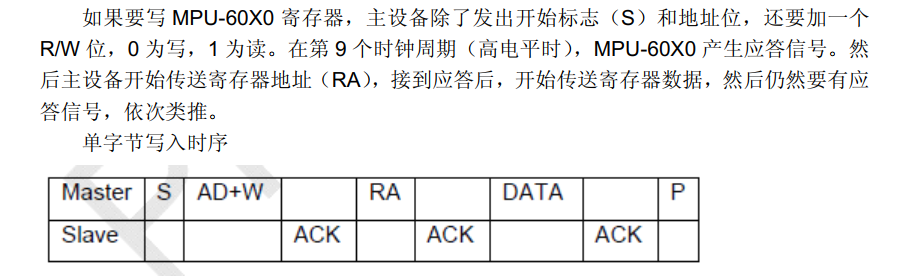
0X69，注意：这里的地址是不包含数据传输的最低位的（最低位用来表示读写）！！

器件地址 + 写方向：0110 1000 << 1 == 1101 000 | 0 – 0Xd0

器件地址 + 读方向：0110 1000 << 1 == 1101 000 | 1 – 0Xd1

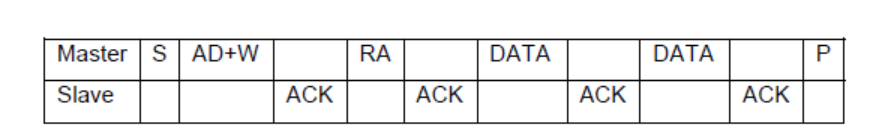
## 9.3 模块软件分析

### 9.3.1 时序图

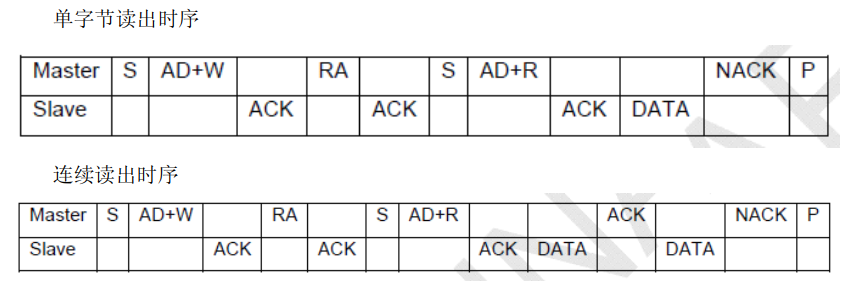


主机发出起始条件 –》 主机发出器件地址 + 写方向 –》 从机发出应答信号 –》 发出寄存器地址—》 从机发出应答信号 –》主机发送写入到的寄存器里的数据 –》 从机发出应答信号 –》 停止条件

连续写入数据



主机发出起始条件 –》 主机发出器件地址 + 写方向 –》 从机发出应答信号 –》 发出寄存器地址—》 从机发出应答信号 –》主机发送写入到的寄存器里的数据（一直写入数据） –》 从机发出应答信号 –》 停止条件



主机发出起始条件 –》 主机发出器件地址 + 写方向 –》 从机发出应答信号 –》 发出寄存器地址—》 从机发出应答信号 –》主机发出起始条件—》主机发出器件地址 + 读方向 –》 从机发出应答信号 –》如果需要一直让从机发送数据则需要主机不断发应答信号，直到不需要发了则主机回非应答—》停止条件

### 9.3.2 步骤

初始化的动作：

1. 找到地址为0x6B的地址的寄存器，往第7位写1，就会复位MPU6050，当复位完成后就会硬件自动把第7位变为0。

2. 找到地址为0x6B的地址的寄存器，往第6位写0，就进入了正常工作模式

3. 找到地址为0x6B的地址的寄存器，往第0-2位写1，就选择了X轴作为参考时钟源

4. 找到地址为0x1B的地址的寄存器，往第3-4位写3，表示陀螺仪量程范围是±2000°/s

5. 找到地址为0x1C的地址的寄存器，往第3-4位写0，表示J加速度计量程范围是±2g

6. 找到地址为0x23的地址的寄存器,往里面写入0，禁止FIFO使能

7. 找到地址为0x19的地址的寄存器,，往里面写入19的值

找到地址为0x1A的地址的寄存器,，往第0-2写入4的值

设置为了50HZ的采样频率

8. 找到地址为0x6c的地址的寄存器,,往里面写入0值

读取回来数据：

读取X轴的加速度：先读取0X3B的地址的寄存器的值（X轴加速度的高8位的值），再读取0x3C的地址的寄存器的值（X轴加速度的低8位的值）

读取Y轴的加速度：先读取0X3D的地址的寄存器的值（Y轴加速度的高8位的值），再读取0x3E的地址的寄存器的值（Y轴加速度的低8位的值）

读取Z轴的加速度：先读取0X3F的地址的寄存器的值（Z轴加速度的高8位的值），再读取0x40的地址的寄存器的值（Z轴加速度的低8位的值）

代码的步骤：

1. 搭建IIC的框架（和SHT20一摸一样）

2. 封装一个单字节写的时序

3. 封装一个连续读的函数

4. 封装一个读写函数（只改变想要改变的位）

往第2和第3位写入2的值

GPIOA->MODER：寄存器本身存储的值

0x3 :代表只需要清零两个位

2：起始的位

GPIOA->MODER &= ~（0x3 << 2）;

GPIOA->MODER |= （0x2 << 2）;

5. 封装一个初始化的函数

6. 封装一个读取加速度值的函数