

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 $\pm 250$ ， $\pm 500$ ， $\pm 1000$ ， $\pm 2000$  °/秒 (dps)，加速度计可测范围为 $\pm 2$ ， $\pm 4$ ， $\pm 8$ ， $\pm 16g$ 。

在网上找了一会，好像 MPU-6050 没有中文的数据手册，由于本人也处于学习阶段，翻译的可能不太准确，只能表达一下简单的意思，以官方数据手册为准。

引脚说明：

引脚编号	MPU-6000	MPU-6050	引脚名称	描述
1	Y	Y	CLKIN	可选的外部时钟输入，如果不用则连到 GND
6	Y	Y	AUX_DA	I2C 主串行数据，用于外接传感器
7	Y	Y	AUX_CL	I2C 主串行时钟，用于外接传感器
8	Y		/CS	SPI 片选 (0=SPI mode)
8		Y	VLOGIC	数字 I/O 供电电压
9	Y		AD0/SDO	I2C Slave 地址 LSB (AD0); SPI 串行数据输出 (SDO)
9		Y	AD0	I2C Slave 地址 LSB (AD0)
10	Y	Y	REGOUT	校准滤波电容连线
11	Y	Y	FSYNC	帧同步数字输入
12	Y	Y	INT	中断数字输出 (推挽或开漏)
13	Y	Y	VDD	电源电压及数字 I/O 供电电压
18	Y	Y	GND	电源地
19, 21, 22	Y	Y	RESV	预留，不接
20	Y	Y	CPOUT	电荷泵电容连线
23	Y		SCL/SCLK	I2C 串行时钟 (SCL); SPI 串行时钟 (SCLK)
23		Y	SCL	I2C 串行时钟 (SCL)
24	Y		SDA/SDI	I2C 串行数据 (SDA); SPI 串行数据输入 (SDI)
24		Y	SDA	I2C 串行数据 (SDA)
2, 3, 4, 5, 14, 15, 16, 17	Y	Y	NC	不接

VDD 供电电压为 2.5V $\pm 5\%$ 、3.0V $\pm 5\%$ 、3.3V $\pm 5\%$ ；VDDIO 为 1.8V $\pm 5\%$

内建振荡器在工作温度范围内仅有 $\pm 1\%$ 频率变化。可选外部时钟输入 32.768kHz 或 19.2MHz

找出几个重要的寄存器：

1) Register 25 – Sample Rate Divider (SMPRT\_DIV)

Type: Read/Write

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

1) SMPLRT\_DIV 8 位无符号值，通过该值将陀螺仪输出分频，得到采样频率

该寄存器指定陀螺仪输出率的分频，用来产生 MPU-60X0 的采样率。  
传感器寄存器的输出、FIFO 输出、DMP 采样和运动检测的都是基于该采样率。  
采样率的计算公式

采样率 = 陀螺仪的输出率 / (1 + SMPLRT\_DIV)

当数字低通滤波器没有使能的时候，陀螺仪的输出平路等于 8KHZ，反之等于 1KHZ。

2) Register 26 – Configuration (CONFIG)

Type: Read/Write

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

1) EXT\_SYNC\_SET 3 位无符号值，配置帧同步引脚的采样

2) DLPF\_CFG 3 位无符号值，配置数字低通滤波器

该寄存器为陀螺仪和加速度计配置外部帧同步（FSYNC）引脚采样和数字低通滤波器（DLPF）。

通过配置 EXT\_SYNC\_SET，可以对连接到 FSYNC 引脚的一个外部信号进行采样。

FSYNC 引脚上的信号变化会被锁存，这样就能捕获到很短的频闪信号。

采样结束后，锁存器将复位到当前的 FSYNC 信号状态。

根据下面的表格定义的值，采集到的数据会替换掉数据寄存器中上次接收到的有效数据

EXT_SYNC_SET	FSYNC Bit Location
0	Input disabled
1	TEMP_OUT_L[0]
2	GYRO_XOUT_L[0]
3	GYRO_YOUT_L[0]
4	GYRO_ZOUT_L[0]
5	ACCEL_XOUT_L[0]
6	ACCEL_YOUT_L[0]
7	ACCEL_ZOUT_L[0]

数字低通滤波器是由 DLPF\_CFG 来配置，根据下表中 DLPF\_CFG 的值对加速度传感器和陀

螺仪滤波

DLPF_CFG	Accelerometer (F <sub>s</sub> = 1kHz)		Gyroscope		
	Bandwidth (Hz)	Delay (ms)	Bandwidth (Hz)	Delay (ms)	Fs (kHz)
0	260	0	256	0.98	8
1	184	2.0	188	1.9	1
2	94	3.0	98	2.8	1
3	44	4.9	42	4.8	1
4	21	8.5	20	8.3	1
5	10	13.8	10	13.4	1
6	5	19.0	5	18.6	1
7	RESERVED		RESERVED		8

3) Register 27 – Gyroscope Configuration (GYRO\_CONFIG)

Type: Read/Write

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

- 1) XG\_ST 设置此位，X 轴陀螺仪进行自我测试。
- 2) YG\_ST 设置此位，Y 轴陀螺仪进行自我测试。
- 3) ZG\_ST 设置此位，Z 轴陀螺仪进行自我测试。
- 4) FS\_SEL 2 位无符号值。选择陀螺仪的量程。

这个寄存器是用来触发陀螺仪自检和配置陀螺仪的满量程范围。

陀螺仪自检允许用户测试陀螺仪的机械和电气部分，通过设置该寄存器的 XG\_ST、YG\_ST 和 ZG\_ST bits 可以激活陀螺仪对应轴的自检。每个轴的检测可以独立进行或同时进行。

自检的响应 = 打开自检功能时的传感器输出 - 未启用自检功能时传感器的输出

在 MPU-6000/MPU-6050 数据手册的电气特性表中已经给出了每个轴的限制范围。当自检的响应值在规定的范围内，就能够通过自检；反之，就不能通过自检。

根据下表，FS\_SEL 选择陀螺仪输出的量程：

FS_SEL	Full Scale Range
0	± 250 °/s
1	± 500 °/s
2	± 1000 °/s
3	± 2000 °/s

4) Register 28 – Accelerometer Configuration (ACCEL\_CONFIG)

- 1) XA\_ST 设置为 1 时，X 轴加速度感应器进行自检。
- 2) YA\_ST 设置为 1 时，Y 轴加速度感应器进行自检。
- 3) ZA\_ST 设置为 1 时，Z 轴加速度感应器进行自检。
- 4) AFS\_SEL 2 位无符号值。选择加速度计的量程。

具体细节和上面陀螺仪的相似。  
根据下表，AFS\_SEL 选择加速度传感器输出的量程。

AFS_SEL	Full Scale Range
0	$\pm 2g$
1	$\pm 4g$
2	$\pm 8g$
3	$\pm 16g$

5) Registers 59 to 64 – Accelerometer Measurements (ACCEL\_XOUT\_H, ACCEL\_XOUT\_L, ACCEL\_YOUT\_H, ACCEL\_YOUT\_L, ACCEL\_ZOUT\_H, and ACCEL\_ZOUT\_L)

Type: Read Only

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
3B	59	ACCEL_XOUT[15:8]							
3C	60	ACCEL_XOUT[7:0]							
3D	61	ACCEL_YOUT[15:8]							
3E	62	ACCEL_YOUT[7:0]							
3F	63	ACCEL_ZOUT[15:8]							
40	64	ACCEL_ZOUT[7:0]							

- 1) ACCEL\_XOUT 16 位 2’s 补码值。  
存储最近的 X 轴加速度感应器的测量值。
- 2) ACCEL\_YOUT 16 位 2’s 补码值。  
存储最近的 Y 轴加速度感应器的测量值。
- 3) ACCEL\_ZOUT 16 位 2’s 补码值。  
存储最近的 Z 轴加速度感应器的测量值。

这些寄存器存储加速感应器最近的测量值。

加速度传感器寄存器，连同温度传感器寄存器、陀螺仪传感器寄存器和外部感应数据寄存器，都由两部分寄存器组成（类似于 STM32F10X 系列中的影子寄存器）：一个内部寄存器，用户不可见。另一个用户可读的寄存器。内部寄存器中数据在采样的时候及时的到更新，仅在串行通信接口不忙碌时，才将内部寄存器中的值复制到用户可读的寄存器中去，避免了直接对感应测量值的突发访问。

在寄存器 28 中定义了每个 16 位的加速度测量值的最大范围, 对于设置的每个最大范围, 都对应一个加速度的灵敏度 ACCEL\_xOUT, 如下面的表中所示:

AFS_SEL	Full Scale Range	LSB Sensitivity
0	$\pm 2g$	16384 LSB/g
1	$\pm 4g$	8192 LSB/g
2	$\pm 8g$	4096 LSB/g
3	$\pm 16g$	2048 LSB/g

#### 6) Registers 65 and 66 – Temperature Measurement (TEMP\_OUT\_H and TEMP\_OUT\_L)

**Type: Read Only**

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
41	65	TEMP_OUT[15:8]							
42	66	TEMP_OUT[7:0]							

1) TEMP\_OUT 16 位有符号值。

存储的最近温度传感器的测量值。

#### 7) Registers 67 to 72 – Gyroscope Measurements (GYRO\_XOUT\_H, GYRO\_XOUT\_L, GYRO\_YOUT\_H, GYRO\_YOUT\_L, GYRO\_ZOUT\_H, and GYRO\_ZOUT\_L)

**Type: Read Only**

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
43	67	GYRO_XOUT[15:8]							
44	68	GYRO_XOUT[7:0]							
45	69	GYRO_YOUT[15:8]							
46	70	GYRO_YOUT[7:0]							
47	71	GYRO_ZOUT[15:8]							
48	72	GYRO_ZOUT[7:0]							

这个和加速度感应器的寄存器相似  
对应的灵敏度:

FS_SEL	Full Scale Range	LSB Sensitivity
0	$\pm 250\text{ }^{\circ}/s$	131 LSB/ $^{\circ}/s$
1	$\pm 500\text{ }^{\circ}/s$	65.5 LSB/ $^{\circ}/s$
2	$\pm 1000\text{ }^{\circ}/s$	32.8 LSB/ $^{\circ}/s$
3	$\pm 2000\text{ }^{\circ}/s$	16.4 LSB/ $^{\circ}/s$

#### 8) Register 107 – Power Management 1 (PWR\_MGMT\_1)

**Type: Read/Write**

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

该寄存器允许用户配置电源模式和时钟源。它还提供了一个复位整个器件的位，和一个关闭温度传感器的位

- 1) DEVICE\_RESET 置 1 后所有的寄存器复位，随后 DEVICE\_RESET 自动置 0.
- 2) SLEEP 置 1 后进入睡眠模式
- 3) CYCLE 当 CYCLE 被设置为 1，且 SLEEP 没有设置，MPU-60X0 进入循环模式，为了从速度传感器中获得采样值，在睡眠模式和正常数据采集模式之间切换，每次获得一个采样数据。在 LP\_WAKE\_CTRL（108）寄存器中，可以设置唤醒后的采样率和被唤醒的频率。
- 4) TEMP\_DIS 置 1 后关闭温度传感器
- 5) CLKSEL 指定设备的时钟源

时钟源的选择：

#### 9) Register 117 – Who Am I (WHO\_AM\_I)

**Type: Read Only**

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
75	117	-	WHO_AM_I[6:1]						-

WHO\_AM\_I 中的内容是 MPU-60X0 的 6 位 I2C 地址

上电复位的第 6 位到第 1 位值为：110100

为了让两个 MPU-6050 能够连接在一个 I2C 总线上，当 AD0 引脚逻辑低电平时，设备的地址是 b1101000，当 AD0 引脚逻辑高电平时，设备的地址是 b1101001

(2013.01.24)

淘宝买的货终于到了，学习用所以没买好的，这个模块只要 18 块钱。





MPU-6000 可以使用 SPI 和 I2C 接口，而 MPU-6050 只能使用 I2C，其中 I2C 的地址由 ADO 引脚决定;寄存器共 117 个，挺多的，下面的是精简常用的，根据具体的要求，适当的添加。

```
#define SMPLRT_DIV 0x19 //采样率分频，典型值：0x07(125Hz) */
#define CONFIG 0x1A // 低通滤波频率，典型值：0x06(5Hz) */
#define GYRO_CONFIG 0x1B // 陀螺仪自检及测量范围，典型值：0x18(不自检，
2000deg/s) */
#define ACCEL_CONFIG 0x1C // 加速计自检、测量范围及高通滤波频率，典型值：0x01(不
自检，2G，5Hz) */

#define ACCEL_XOUT_H 0x3B // 存储最近的 X 轴、Y 轴、Z 轴加速度感应器的测量值 */
#define ACCEL_XOUT_L 0x3C
#define ACCEL_YOUT_H 0x3D
#define ACCEL_YOUT_L 0x3E
#define ACCEL_ZOUT_H 0x3F
#define ACCEL_ZOUT_L 0x40

#define TEMP_OUT_H 0x41 // 存储的最近温度传感器的测量值 */
#define TEMP_OUT_L 0x42

#define GYRO_XOUT_H 0x43 // 存储最近的 X 轴、Y 轴、Z 轴陀螺仪感应器的测量值 */
#define GYRO_XOUT_L 0x44
#define GYRO_YOUT_H 0x45
#define GYRO_YOUT_L 0x46
#define GYRO_ZOUT_H 0x47
#define GYRO_ZOUT_L 0x48

#define PWR_MGMT_1 0x6B // 电源管理，典型值：0x00(正常启用) */
```

```
#define WHO_AM_I 0x75 //IIC 地址寄存器(默认数值 0x68，只读) */
```

编程时用到的关于 I2C 协议规范：

信号	描述
S	开始标志：SCL 为高时 SDA 的下降沿
AD	从设备地址（Slave 地址）
W	写数据位（0）
R	读数据位（1）
ACK	应答信号：在第 9 个时钟周期 SCL 为高时，SDA 为低
NACK	拒绝应答：在第 9 个时钟周期，SDA 一直为高
RA	MPU-60X0 内部寄存器地址
DATA	发送或接受的数据
P	停止标志：SCL 为高时 SDA 的上升沿