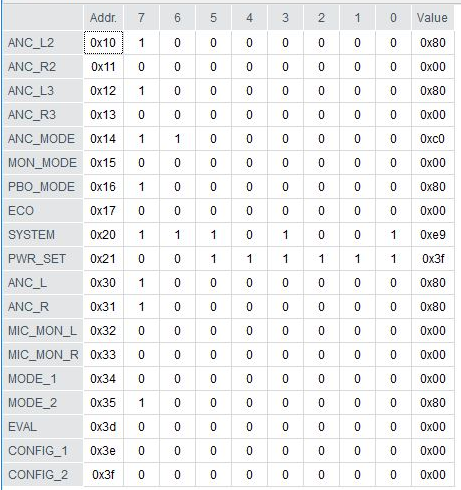
# I2C读写逻辑

* 寄存器地址及对应功能如下表：



* 收发收据格式，总共65个数据，后面的用cb或者00填充：

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 地址 | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 |
| out | 00 | i++ | 08 | 00 | 61:read e1:write | 00 | 03 | 00 | 01：read 00：write | 8e:master 8c:slave | addr | value |
| in | 00 | i++ | 06 | 00 | 61 | 00 | fd:未连接 00:已连接 | 00 | 01 | value | 00 | 03 |

* USB初始化：

OUT 00 07 04 03 01 04 04 cb cb cb cb cb

IN 00 de 03 03 ff 00 00 00 00 00 00 00

OUT 00 01 05 00 67 00 00 00 03 cb cb cb

IN 00 11 08 00 67 00 00 00 03 01 00 03

OUT 00 02 0c 00 e2 00 07 00 00 02 01 02

* 芯片识别：

OUT 00 03 08 00 61 00 03 00 01 8e 20 8f (master)

OUT 00 03 08 00 61 00 03 00 01 8c 20 8d (salve)

IN 00 32 06 00 61 00 fd 00 01 01 00 03

根据in数据判断，fd:未连接，00：已连接

* 芯片第一次连接时响应：

IN 00 61 06 00 61 00 00 00 01 e1 00 03 00 00 00 00

（e1代表版本信息及i2c通电）

* 连接成功后，1ms间隔循环以下动作：

OUT 00 7a 08 00 e1 00 03 00 00 8e 20 09 # 写入版本参数及保持I2C

OUT 00 7b 08 00 e1 00 03 00 00 8e 3f 00 # OPT read模式

OUT 00 94 08 00 61 00 03 00 01 8e addr 8f # 循环询问19组寄存器value

IN 00 4b 06 00 61 00 00 00 01 value 00 03 # 应答前次询问

* 读取寄存器value指令：

Master:

OUT 00 7a 08 00 e1 00 03 00 00 8e 20 09 # 写入版本参数及保持I2C

OUT 00 i++ 08 00 e1 00 03 00 00 8e 3f 00 # OPT read模式

OUT 00 i++ 08 00 61 00 03 00 01 8e addr 8f

IN 00 i++ 06 00 61 00 00 00 01 value 00 03

Slave:

OUT 00 7a 08 00 e1 00 03 00 00 8c 20 09 # 写入版本参数及保持I2C

OUT 00 i++ 08 00 e1 00 03 00 00 8c 3f 00 # OPT read模式

OUT 00 i++ 08 00 61 00 03 00 01 8c addr 8d

IN 00 i++ 06 00 61 00 00 00 01 value 00 03

* 更改寄存器value指令：

Master:

OUT 00 7a 08 00 e1 00 03 00 00 8e 20 09 # 写入版本参数及保持I2C

OUT 00 i++ 08 00 e1 00 03 00 00 8e 3f 02 # OPT write模式

OUT 00 i++ 08 00 e1 00 03 00 00 8e addr value

Slave:

OUT 00 7a 08 00 e1 00 03 00 00 8c 20 09 # 写入版本参数及保持I2C

OUT 00 i++ 08 00 e1 00 03 00 00 8c 3f 02 # OPT write模式

OUT 00 i++ 08 00 e1 00 03 00 00 8c addr value