I2C实验（模拟I2C）

**一、环境：**

开发板型号：NanoPi NEO

系统版本：Ubuntu 16.04.5 LTS 4.14.52

实验内容：通过GPIO模拟I2C协议，通过I2C接口实现24C16模块的读写

实验目的：学习程序逻辑分层、封装函数的方法。

**二、实验步骤：**

1. 引脚连接

VCC-----1

SCL-----7

SDA-----12

GND-----9

2. 编译生成可执行文件并执行

make

./24c16

**三、实验记录：**

1. 程序编写

1.1 主程序

(1) 设置管脚映射wiringPiSetupPhys()，使用wiringPi的库函数；

(2) 利用循环每次读写10个字节。

1.2 驱动程序

(1) 根据i2c的时序图，使用GPIO口模拟i2c协议，编写驱动代码

(2) 根据24c16芯片的手册操作i2c的驱动，完成读写函数的编写

2. 注意问题

(1) wiringPi库延时函数，在延时1us不准，出现波动情况；

(2) 24c16的芯片地址的设置，如图1所示其中p2,p1,p0 为内部集成的页地址，发送地址时先发送高3位p2,p1,p0,后发送低8位地址。

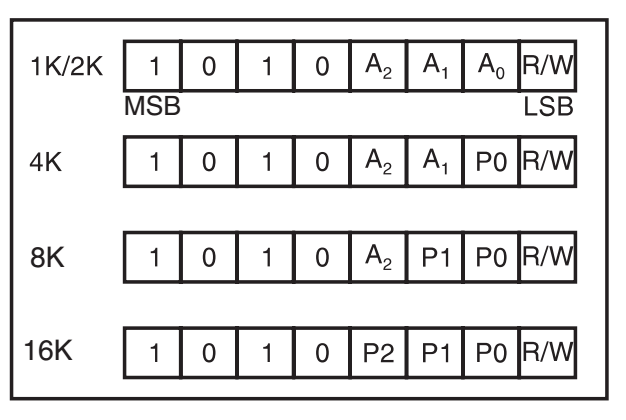


图1

(3) 本程序使用24c16模块的读写程序使用的单字节读写数据，没有使用页写。使用单字节读写数据不考虑分页的情况。

（4）写一个字节需要5ms。 

3. 运行测试

3.1测试读写函数

（1）测试写入字节函数

使用逻辑分析仪分析，写入一个字节0，测试写一个字节函数的准确性。图2可以清晰的看出：SCL为上面波形；SDA为下面波形。

由字节写入函数可知:写入一个字节数据，需要发送3个字节，一个字节发送完，SCL需要拉低等待应答信号，SDA在SCL拉低后发送一个应答信号。

可以根据波形分析出发送的第一个字节是10100000（其中后3位是页地址），第二个字节为00000000，第三个字节为00110000。最后可以得到写入到24c16中的数据是0x30，0x30和写入的字节0相一致。

（2）测试读取字节函数

如图3所示写入字节0后，读出数据，测试读取一个字节函数的准确性。发送三个字节，读取1个字节数据。第一个字节是10100000（其中后3位是页地址），第二个字节为00000000，第三个字节为10100001（切换为读取数据状态），第四个字节为00110000。最后读取到的数据为0x30和写入的数据相同。

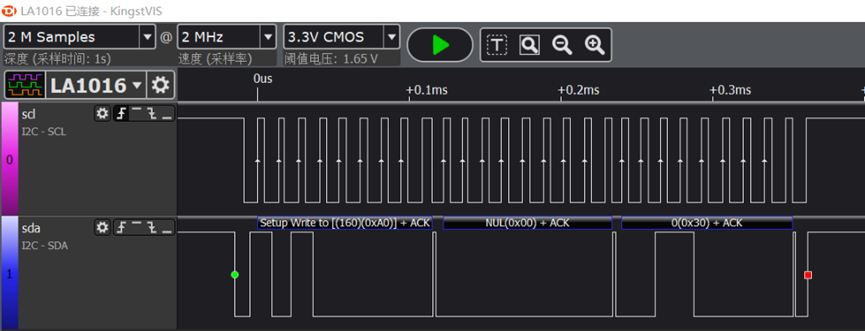


图2

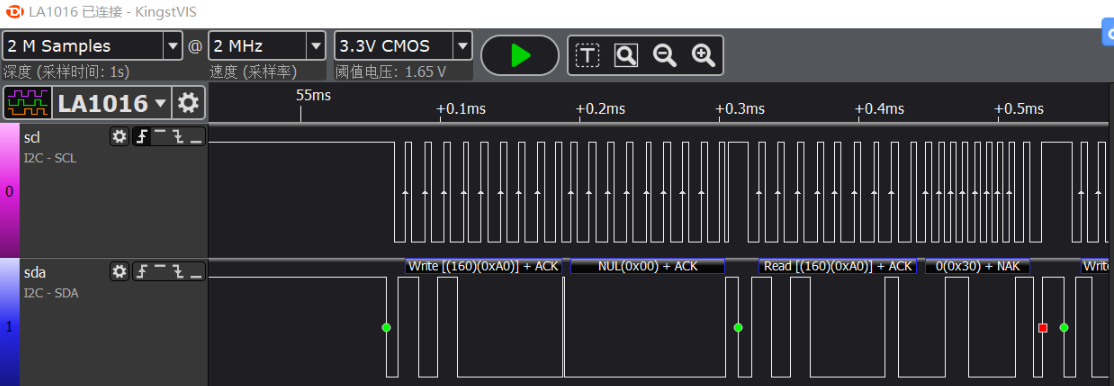


图3

3.2 测试24c16达到的频率

图4测量的sda应答信号的响应时间为4us,可以求出24c16达到最高频率为250khz左右。测量scl一个脉冲的大小，求出利用模拟i2c读写24c16能够达到78khz左右的频率。



图4