**小车测距设计方案**

1. **简介：**

通过树莓派小车完成对操场跑道一圈长度的检测（标准跑道400米）

1. **设计方案：**

1.对树莓派轮子进行测速，采用马达+霍尔传感器的形式。

2.霍尔传感器输出连续脉冲，通过检测脉冲的数量计算小车的距离和前进的速度。

3.根据磁钢和脉冲数量计算车轮转动一周的次数并计算距离

4.根据红外遥控器控制小车行驶方向【出现问题】

{在室外阳光下进行检测时会发生干扰，具体内容看下方思考与总结}

5.在显示器上显示小车行驶过的距离

**三．**  **技术原理：**

**1．霍尔传感器与脉冲信号的原理：**

霍尔传感器时根据霍尔效应制作的一种磁场传感器。霍尔效应是磁场效应的一种。在半导体薄片两端通以控制电流I，并在薄片的垂直方向世家磁感应强度为B的匀强磁场，则在垂直于电流和磁场的方向上，将产生电势差为U的霍尔电压。

当霍尔传感器的磁接触面附近的磁感应强度达到时，传感器输出电压将由低电平转换为高电平，当磁感应强度回落时，输出电压又由高电平转换为低电平。当磁感应强度变化较快时，传感器输出电压就相当于一个脉冲信号。

**2.脉冲信号与测量小车行驶距离的关系：**

测量转速的霍尔传感器和机轴同轴相连，机轴每转一圈，产生一定量的脉冲个数，由霍尔器件电路部分输出。霍尔传感器由霍尔元件和磁钢组成，当霍尔元件和磁钢相对运动时，就会产生脉冲信号，根据磁钢和脉冲数量就可以计算车轮转动一周的次数，再根据车轮的直径或周长算出小车行进的距离。车辆行驶的里程数就是接收到的脉冲数与车轮周长的乘积。

**3.磁钢数量的确认与车轮周长的确认：**

在车轮上只有一处磁性物质的条件下，当车轮没有完整的转动一圈时，传感器就会少产生一个脉冲信号，进而引起误差，此时最大误差为C，车轮上有两处磁性物质时，最大误差为C/2，车轮上有三处磁性物质时，最大误差为C/3，以此类推，可见车轮上磁性物质越多，误差越小。所以在车轮上按一定的角度分布个磁性物质，得到的结果会更精确。

取细线一段放于车轮某一定点处，顺时针旋转一周，取端点测量细线长度，此时细线长度便是小车长度。

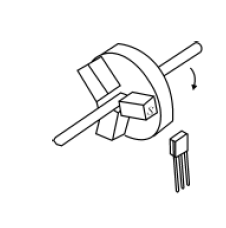
**4.车辆行程测量算法：**

假设车轮长度为L，其上分布了n个磁性物质，在行驶一段距离之后接受到了m个脉冲信号。由于车轮周边分布了n个磁性物质，故转动一圈会产生n个脉冲信号，为求得行驶距离，必须先求出车轮转动的圈数，圈数与车轮周长的乘积就是小车的行驶距离。根据假设条件可知：S（行驶距离）= L**·**m/n

**5.霍尔传感器的安装与实现：**

在小车车轮上安装4-5个磁钢，当车轮转动一圈，将于霍尔传感器发生磁效应，产生电势差，相当于一次脉冲信号，由于没有计数器，选择在程序中计数。

选用的霍尔传感器是：A3144型号（单极） 磁钢：根据车轮大小选取相应大小

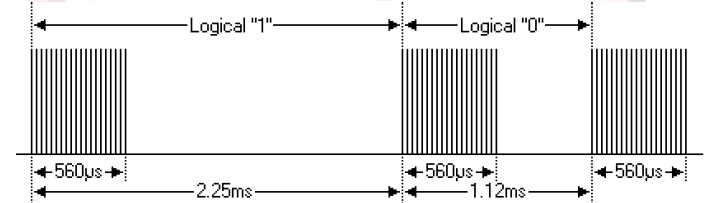
安装位置可选用如图所示：

【注意：A3144型号是单极，所以只有与小磁钢S极才可以产生脉冲信号，且感应距离较小，尽量使其与磁钢直接距离减少。所以在选用器材时也可以选用双极，减少外界干扰，更加精准。】

**6.红外协议之NEC协议**

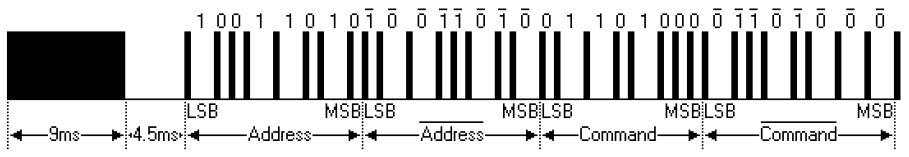
NEC协议载波：38khz

其逻辑1与逻辑0的表示如图所示：



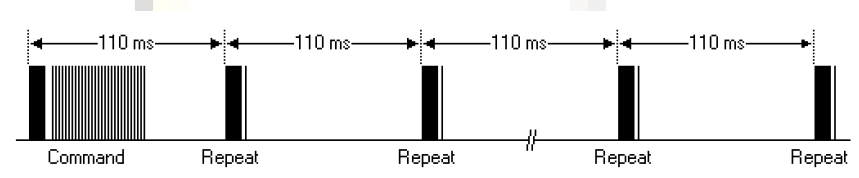
逻辑1为2.25ms，脉冲时间560us；逻辑0为1.12ms，脉冲时间560us。所以我们根据脉冲时间长短来解码。推荐载波占空比为1/3至1/4。

NEC协议格式：

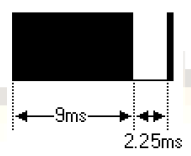


首次发送的是9ms的高电平脉冲，其后是4.5ms的低电平，接下来就是8bit的地址码（从低有效位开始发），而后是8bit的地址码的反码（主要是用于校验是否出错）。然后是8bit 的命令码（也是从低有效位开始发），而后也是8bit 的命令码的反码。

　　以上是一个正常的序列，但可能存在一种情况：你一直按着1个键，这样的话发送的是以110ms为周期的重复码，如下图:



就是说,发了一次命令码之后，不会再发送命令码，而是每隔110ms时间，发送一段重复码。



重复码由9ms高电平和2.25ms的低电平以及560us的高电平组成。

**7.LCD1602显示屏：**

(1).本次使用的是I2c型的LCD1602 液晶显示屏，它背后接的是PCF8574转换器 从16个引脚变为4个（VCC,GND,SCL,SDA）

(2).LCD1602的基本操作分为四种：

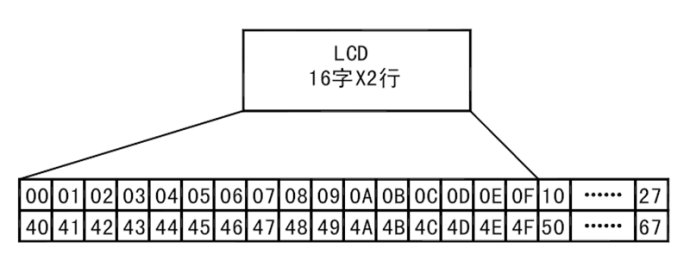
1.    读状态：输入RS=0，RW=1，E=高脉冲。输出：D0—D7为状态字。

2.    读数据：输入RS=1，RW=1，E=高脉冲。输出：D0—D7为数据。

3.    写命令：输入RS=0，RW=0，E=高脉冲。输出：无。

4.    写数据：输入RS=1，RW=0，E=高脉冲。输出：无。

(3).

 左图为显示屏的字节显示位置。

**四．详细技术方案：**

**（1）.树莓派小车的移动**

关于树莓派小车的移动，考虑到小车在操场上移动时，阳光和跑道颜色等因素的不确定性，我们选择使用红外遥控器控制小车的移动，操作简单，人为控制小车的移动轨迹，避免因为循迹失误造成的路程误差。

红外控制模块包含了红外传感器和红外遥控器，使用了LIRC（Linux Infrared remote control），是一个Linux系统下开源的软件包，用于Linux系统接收和发送红外线信号。给遥控器的按键加上对应的事件，比如CH对应小车的前进等等，达到遥控器控制小车移动的目的。

1. 安装lirc sudo apt-get install lirc
2. 修改配置文件/etc/lirc/lirc\_options.conf

#driver          = devinput  
driver = default  
#device = /dev/lirc0  
device          = auto

1. 修改/boot/config.txt

dtoverlay=lirc-rpi,gpio\_in\_pin=19

1. 更改模块配置文件/etc/modules

lirc-dev

lirc-rpi gpio\_in\_pin=19

1. 重启树莓派，让配置生效

sudo reboot

1. 重新开启lirc功能  
   sudo /etc/init.d/lircd restart  
   sudo modprobe lirc\_rpi
2. 关闭红外线接收功能

sudo kill $(pidof lircd)

1. 测试

mode2 -d /dev/lirc0

**（2）.树莓派的测距**

关于树莓派小车的测距，我们选择了使用霍尔元件测距，先测量出小车车轮的周长，根据磁钢和脉冲统计车轮转动的圈数，将圈数与车轮周长相乘，即可得出小车行走的路程。霍尔传感器一般由霍尔元件和磁钢组成，当霍尔元件和磁钢相对运动时就会产生信号脉冲，根据磁钢和脉冲数量就可以计算转速。霍尔传感器具有灵敏，可靠，体积小，无触点，无磨损，使用寿命长，功耗低等特点。这样的测速系统，实现了对脉冲信号的精确、快速测量，硬件成本低，算法简单，稳定性较好。测量速度的霍尔传感器与车轴同轴连接，每当车轴转一圈，产生一定量的脉冲个数，有霍尔器件电路部分输出脉冲，使用计数器统计脉冲的变化次数，即可测量距离。

（3）.系统原理图

霍尔传感器

计算距离

（轮子周长\*次数）

树莓派

计数器

车轮

显示屏显示小车行驶距离的数据

小车前进，传感器感应脉冲信号，程序自动计数

用红外线遥控传感器控制小车行驶

**（3）.小车行驶距离显示**

1. LCD\_1602 初始化指令小结：

0x38 设置16\*2显示（总共可以输出16\*2=32个字符），5\*7点阵（每个字符为5\*7的点阵显示），8位数据接口

0x01 清屏

0x0F 开显示，显示光标，光标闪烁

0x08 只开显示

0x0e 开显示，显示光标，光标不闪烁

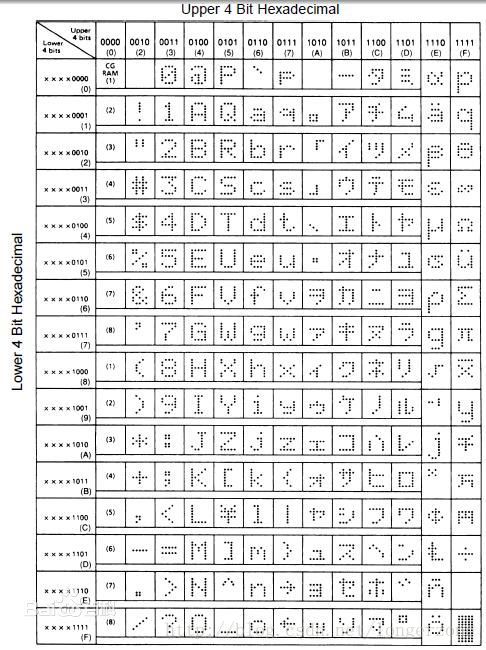
0x0c 开显示，不显示光标

0x06 地址加1，当写入数据的时候光标右移

0x02 地址计数器AC=0;（此时地址为0x80） 光标归原点，但是DDRAM中断内容不变

0x18 光标和显示一起向左移动

2. 1602识别的是ASCII码，下图为它的16位制ASCLL表



3. 树莓派和传感器的连线为：VCC 连接 5V，GND 接地，SCL 连接SCL口，SDA连接SDA口。

**注意：**本来根据资料显示，SCL和SDA口应当连接树莓派的GPIO引脚，可是发现在连接后I2C并不能显示连接设备，只有在连接SCL已经SDA口后才会发现连接设备。

4. 需要准备的工作：

1. 安装 i2c-tools工具与python-smbus

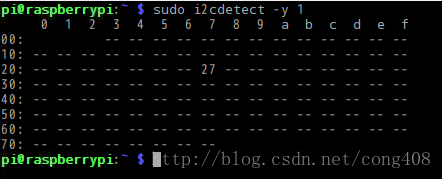
pi@raspberrypi ~ $ sudo apt-get install i2c-tools python-smbus

2. reboot树莓派 重启后千万记得打开i2c接口

3.寻找地址【非常重要！】

pi@raspberrypi ~ $ sudo i2cdetect -y 1

{在测试中错误的原因有一部分就是因为地址}



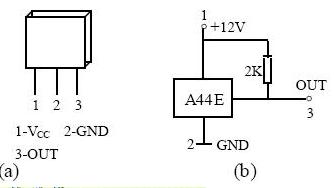
会出现上图所示界面【图源自网络】 一般为27或34，记好自己地址，最后实现代码中常常因为这个出错。

4.测试代码

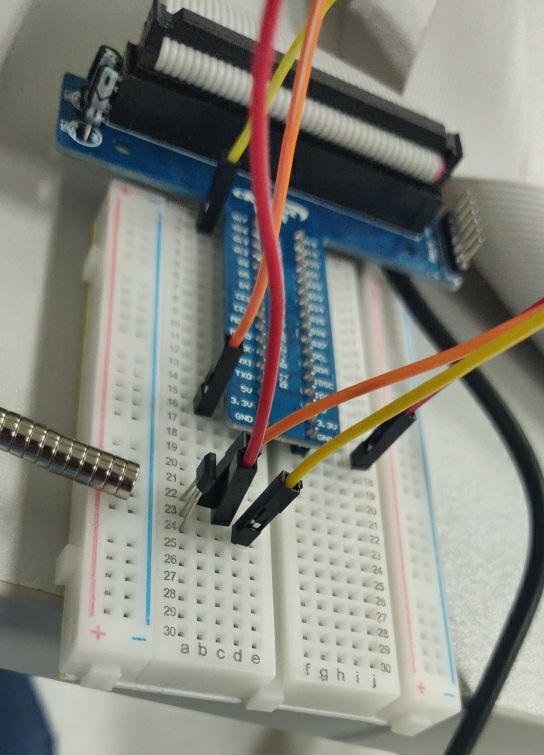
**实验测试：**

1. **霍尔传感器测试：**

**首先我们准备的是A3144 霍尔开关传感器（如下图所示）**

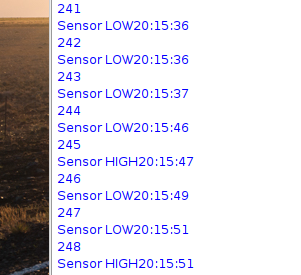
** **

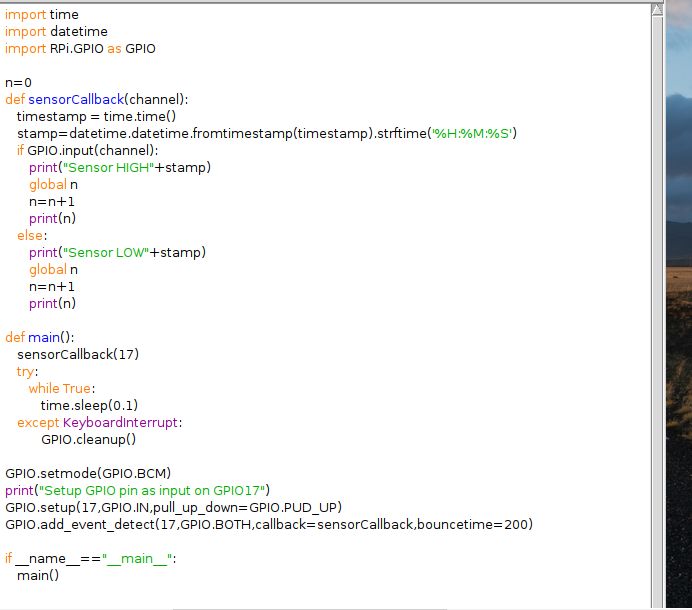
**根据接口（1-vcc连接树莓派3.3V，2-GND 接地，3-cout 接树莓派GPIO口，在测试中我们接的是GPIO 17）**

** **

**左侧是小磁钢，我们选用的是2\*2mm的磁钢，因为车轮较小，恰好在可以感应的区域里放置五个磁钢。**

**测试霍尔传感器的代码如下所示，当磁钢靠近或者经过霍尔传感器时，便会引发高低电压的改变，在程序中放入n来进行计数，算出车轮转动一周的次数，根据检测霍尔传感器与小磁钢的感应距离大约为15mm以内，但是具体还要根据磁钢磁性的不同，以及传感器的质量等问题进行检测。**

** 左侧为测试结果**

** 测试代码**

1. **红外遥控测试**

**下图为红外遥控器和红外传感器。（红外传感器的DO连接树莓派的GPIO１９，VCC接５V，GND接地。）**



测试红外接收功能：mode2 -d /dev/lirc0，

用红外遥控器对着接收器按下任意键，屏幕会打印（配截图）

space 16300

pulse 95

space 28794

pulse 80

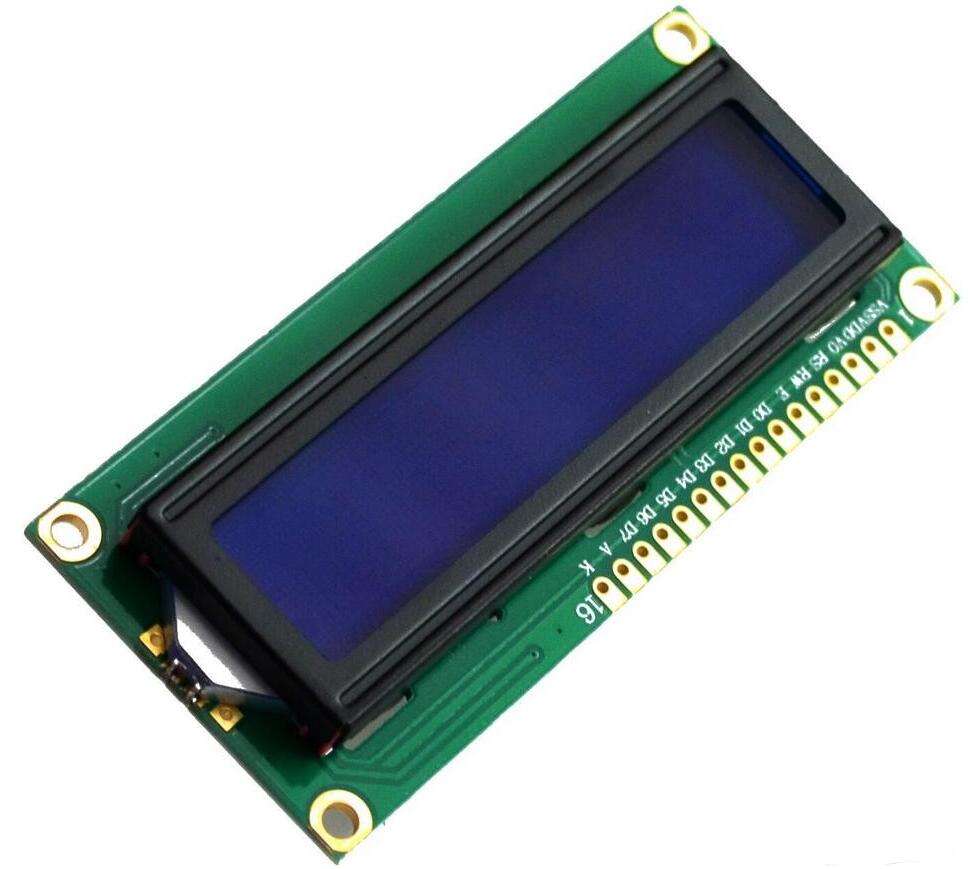
space 19395

证明红外接收功能正常。

测试红外控制功能：（配代码截图）

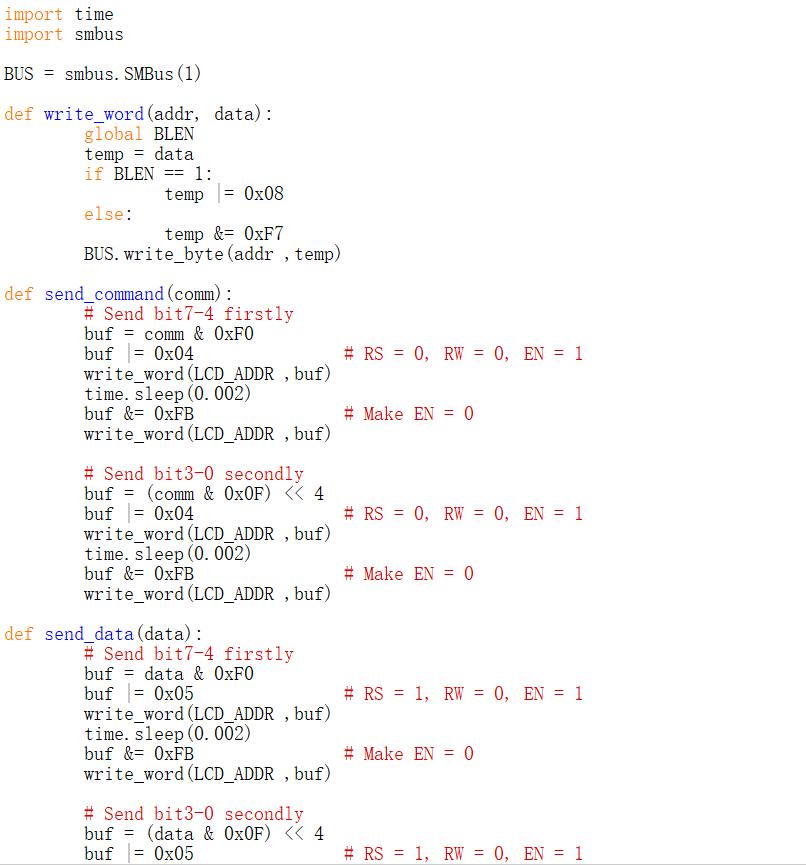
1. **LCD1602显示屏测试**

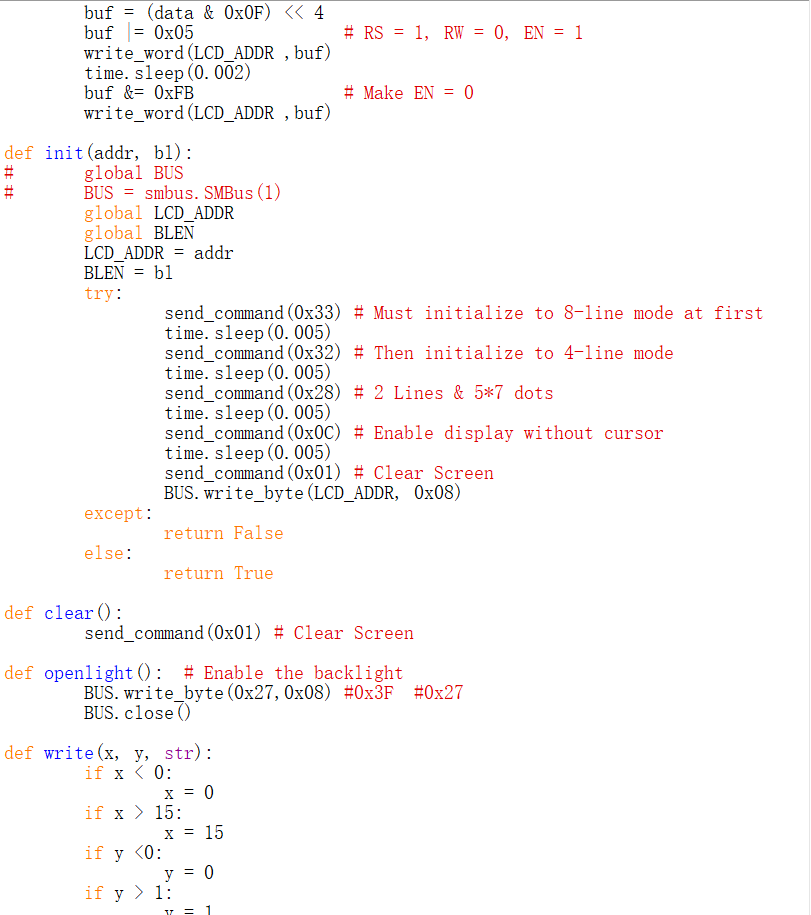
**下图为所用的显示屏和转换器（转换器图片下次去实验室补）：**

****

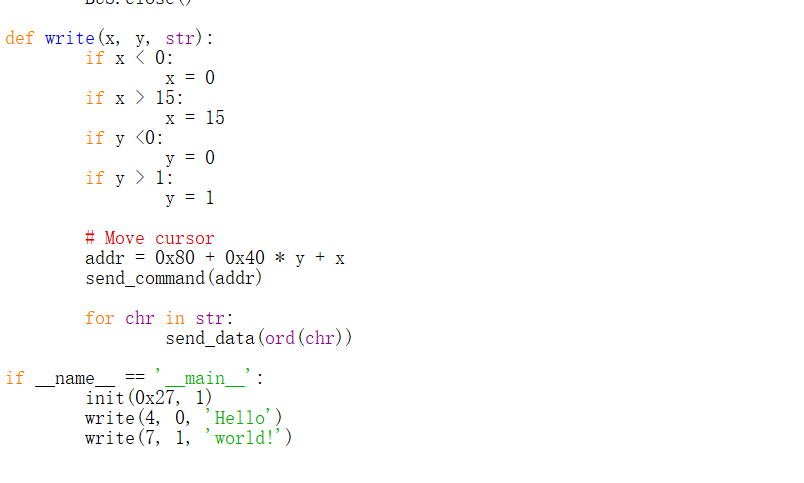
**！注意：显示屏可能会出现亮度过高的情况，在显示屏背后有电阻进行调节，如果大家出现不显示文字，那么可能会是因为屏幕亮度过高导致的，建议调一下电阻，改变对比度显示文字。**

**测试代码：**

****

** 注意 此处！**

**【根据你寻找到的地址 27则为 0x27；34 则为 0x3F】**

****

**测试结果（图片如下所示）：**

**测试结果图片忘记照了，下次去实验室补照**

**五．思考与总结：**

在小车的移动方面，我们做的有些过于简单，暂时还未想到有哪些精确的循迹方法，所以选择了红外遥控器来控制车的移动，免去了复杂的数据处理和计算。在测距方面，目前我们认为通过霍尔传感器测距的方法较为简便和准确，不仅可以测距，也可以通过记录小车移动的时间，来进行测速，后期也可以在小车上安装显示器，来显示已测距离。

通过本次设计，使得我们对于树莓派有了进一步了解，也学习到了霍尔传感器的相关知识和使用方法，为将来的实践工作打下了基础。

问题：

1. 在室外测试中发现外界的阳光会对小车的红外遥控进行干扰，特别是阳光中的红外线会干扰小车的红外线接收器接受小车遥控器传出来的红外数据

解决方案：

1. 在红外遥控接收器附近加上遮光板一类，尽量减少阳光的干扰，但是同时会增加的问题时小车本身红外遥控也会同样受到干扰，接收信号更加困难
2. 更换红外遥控的方法，在寻找中发现了蓝牙模板，可以考虑运用蓝牙模板进行手机控制小车的移动，但是同样也会受到距离的影响
3. 换成最基本的按键控制，太low了，在其他解决方法不成功的情况下才会选用
4. 小车在额外连接多个传感器的情况下，会造成连接错误，在电脑上与小车远程控制时会造成“内部错误”的提醒，在与学长探讨后怀疑是供压不足，树莓派无法连接过多的外部传感器，在寻找中发现小车车身上似乎自带了一个红外接收传感器，准备查文档之后进行改进

解决方法：

在多次测试后找寻报错原因进行修改，在查文档后寻找新的解决方法。