海盗船 --- 红外开关避障功能

在完成基础功能后，我们对机器人来个升级，给它增加避障功能。避障功能，是机器人最基础的功能之一，也是机器人体现自主意识的第一步。避障机器人能够识别前方的障碍物，并避开前方障碍物。这里，我们选用到的是红外数字避障传感器，这是一款相当容易上手的传感器，非常适合初学者使用。该传感器可帮助机器人看到前方的物体。

**硬件材料：**

* 3-80cm红外数字避障传感器 × 3



* 红外接近开关支架 × 3



**组装步骤**

找到硬件材料后，就可以开始组装了，组装不难，更着操作就行。

**STEP 1: 摆放材料**

先找到固定红外开关传感器的3个支架，及需要使用的M3×6MM的螺丝和螺母若干。



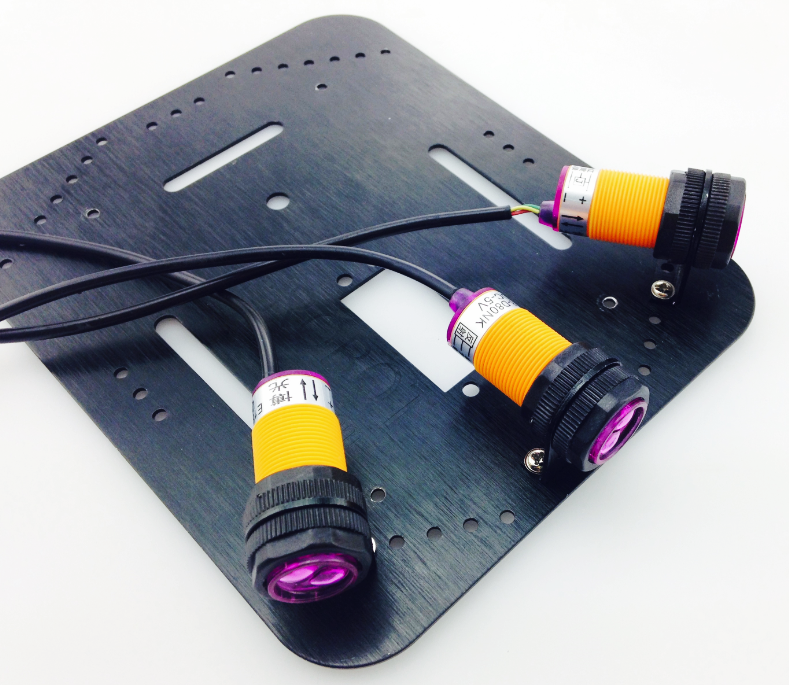
**STEP 2: 固定红外支架**

这里只需将3个红外开关传感器的支架用螺丝拧在上层板上即可。需要注意的一点是，三个支架之间不要间隔太近。否则在检测的时候会造成接收错误信号。



**STEP 3: 安装红外传感器**

先将一个固定环装到传感器上，将传感器从红外支架里往外穿过，并用另一个固定环将其固定住。同样方法安装另外两个。

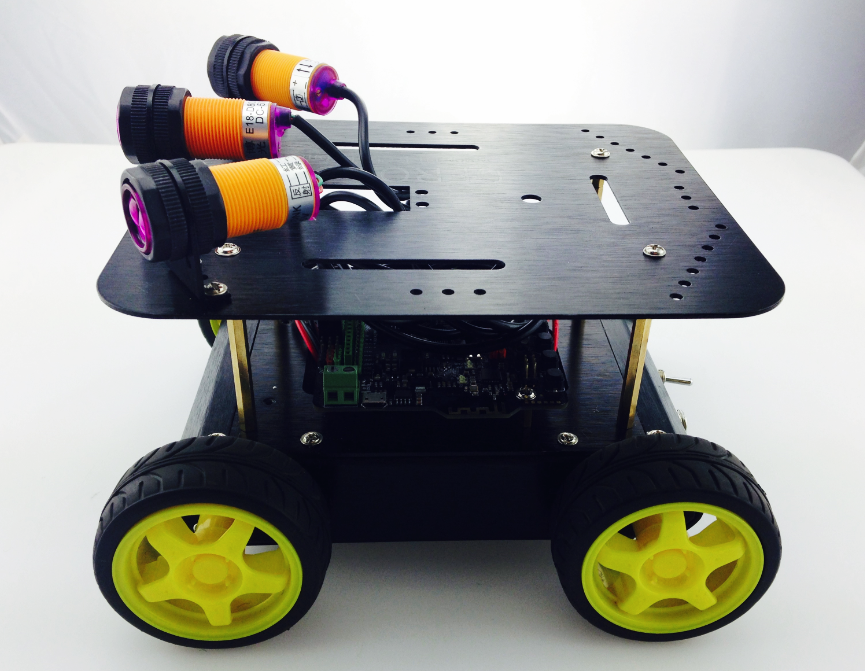


**硬件连接**

传感器安装完成后，先不要急着把上层板接上去，需要将传感器与Romeo BLE控制器连接。左边这张图显示的是三个传感器ABC在小车上对应的位置，分别对应数字口的10，9，8。，连接时只需留意线序是否正确即可。 安装完成后，固定上层板。

**黄色 - 信号线，红色 – VCC, 绿色 – GND。**





**传感器调试**

红外接近开关是一种集发射与接收于一体的光电开关传感器。如果有信号，也就是前方有障碍物，传感器后侧指示灯亮的亮灭，检测距离可随传感器后侧的电位器进行调节，可调范围3-80cm。

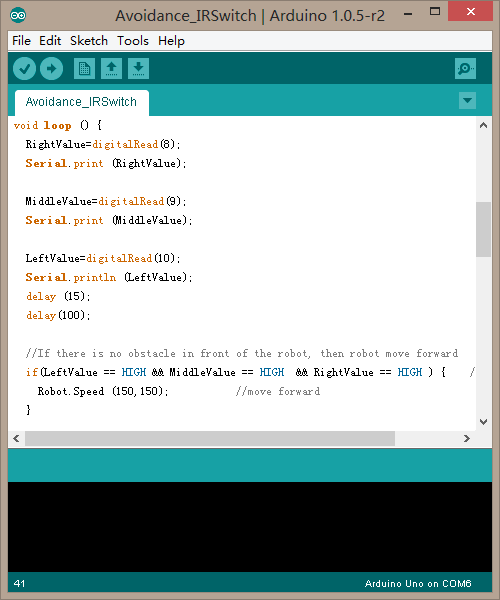
下载代码前，需要对三个传感器的做调整，先插上USB线，给板子供电。下图白色圈出的位置可见有个一字螺丝，这个就是可用来调整检测距离。拿个小盒子(模拟障碍物)，放在传感器探头前，找到传感器的探测点，一旦找到传感器上的指示灯会亮起。此时，可以拧转螺丝，调到一个合适的距离，推荐在15~20cm左右。

同样方法调整另外两个红外传感器，通常建议左右两侧的检测距离略大于中间的检测距离。

c

**输入代码**

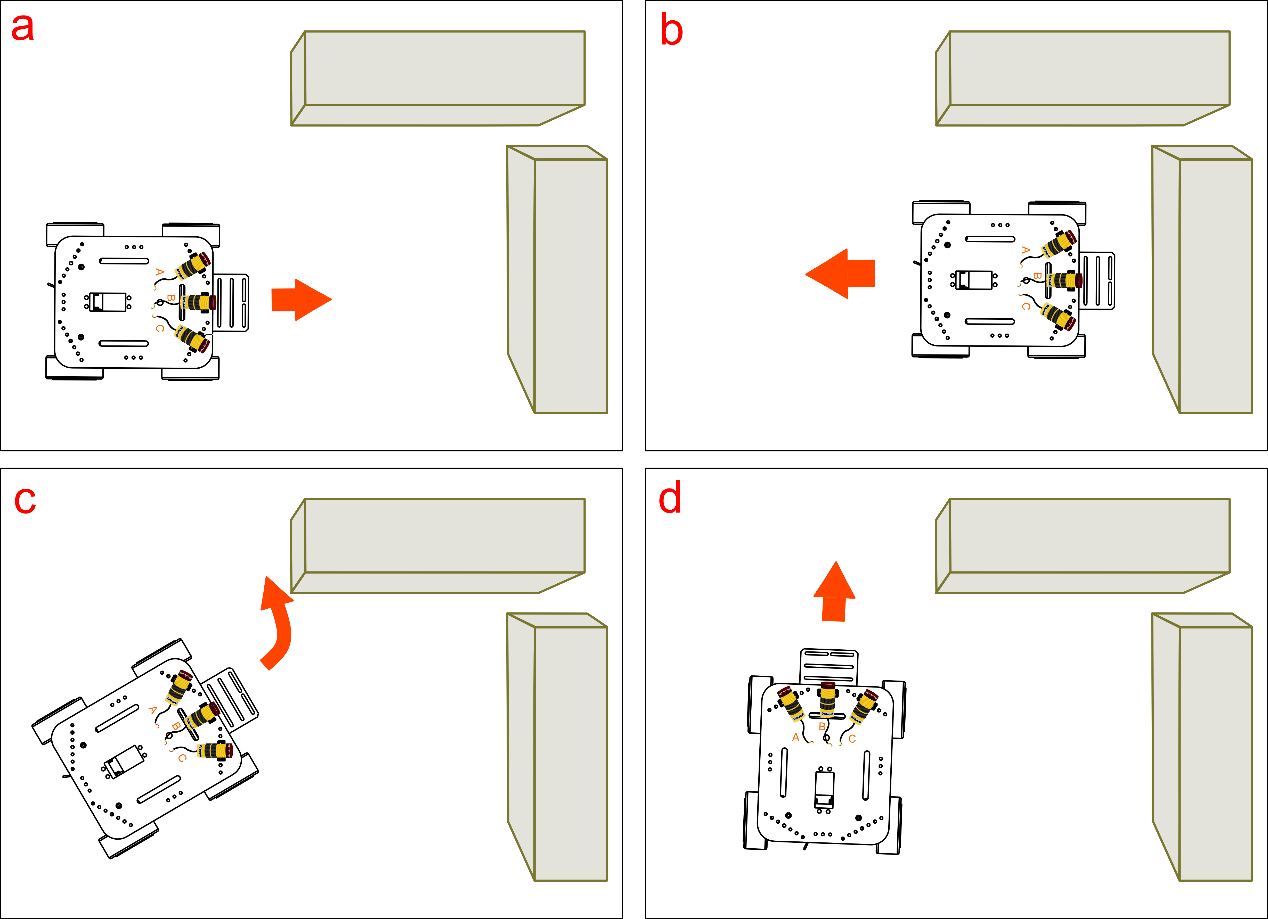
由于代码较长，所以这里就不全部显示出来，你可在软件包中找到Avoidance\_IRSwitch.ino的代码，下载进去后即可。

c

下载完成后，你就可以看到小车的避障效果了。当然，如果你觉得检测距离还不是很满意，也可以做一些微调。直到能正常工作。玩过之后，我们就来简单看下避障原理以及学习下代码，便于你能更好的改造你的机器人！

**避障原理**

通过下面a，b，c，d来简单看下如何通过三个传感器来实现避障。在前方没有障碍物的时候，小车前行(图a)，一旦检测到前方有障碍物，先让小车后退(图b)，然后执行左转(图c)，当然这里你也可以右转。如果小车前方一直存在障碍物，小车将会一直执行后退左转，直到前方没有障碍物，小车保持前进。

****

前面分析的这个场景是小车正前方有障碍物。思考下，如果小车左前方有障碍，或者小车右前方有障碍物，可让小车如何执行？清楚原理之后，我们来看下代码。

当然，这样的避障方式不是唯一的，你也可以有更好的避障方式，可由你自己设定，如果你的逻辑非常紧密，完全可以写出更出色的代码，毫无疑问的可以从你的代码中表现出来。何不试试用自己的思维来写一段控制代码。看完代码回顾，你就能游刃有余的改代码了。

**代码回顾**

基础功能重复部分就不说了，只说下避障部分的。

int RightValue; //Right line tractor sensor on Pin 8

int MiddleValue; //Middle line tractor sensor on Pin 9

int LeftValue; //Left line tractor sensor on Pin 10

//reading 3 pins values of Line Tracking Sensor

RightValue=digitalRead(8);

MiddleValue=digitalRead(9);

LeftValue=digitalRead(10);

用三个变量RightValue，MiddleValue，LeftValue分别记录3个传感器读到的值，digitalRead(pin)函数就是用来读取数字口的值，不明白的可以查看下[语法手册](http://wiki.dfrobot.com.cn/index.php/Arduino%E7%BC%96%E7%A8%8B%E5%8F%82%E8%80%83%E6%89%8B%E5%86%8C%EF%BC%88%E5%A4%9A%E9%A1%B5%E9%9D%A2%E7%89%88%EF%BC%89)。

**红外数字避障传感器数字量传感器，当探头前方有障碍时输出低电平，无障碍物时输出高电平。**

下面这段代码描述的就是避障原理中所说的那张情况 –- 正前方有障碍。三个传感器同时都检测到障碍物了，也就是输出值都为LOW时，小车先执行后退，再执行左转，前进。

if(LeftValue==LOW && MiddleValue==LOW && RightValue==LOW ) {

Robot.Speed (-200,-200); //back off

delay(800);

Robot.Speed (-200,200); //turn left

delay(400);

Robot.Speed (100,100); //move forward

}

如果只有左侧传感器检测到障碍物，那么小车执行，后退，左转。

if(LeftValue == LOW) { //obstacle on the left side

Robot.Speed (-200,-200); //back off

delay(400);

Robot.Speed (200,-200); //turn right

delay(250);

Robot.Speed (100,100); //move forward

}

同样，如果只有右侧传感器检测到障碍物，那么小车执行，后退，右转。

if(RightValue == LOW ){ //obstacle on the right side

Robot.Speed (-200,-200); //back off

delay(400);

Robot.Speed (-200,200); //turn left

delay(250);

Robot.Speed (100,100); //move forward

}

最后，如果前面没有障碍物的话，小车正常前进。

if(LeftValue==HIGH && MiddleValue==HIGH && RightValue==HIGH ) {

Robot.Speed (150,150); //move forward

}

代码中还有这么一段。

if( MiddleValue == LOW){ //obstacle in middle

Robot.Speed (-150,-150); //back off

delay(400);

Robot.Speed (-200,200); //turn left

delay(300);

Robot.Speed (100,100); //move forward

}

同样是前方有障碍物，和我们一开始说的三个传感器同时检测到的区别在于，这种情况下，前方障碍物相对远一点，小车后退的速度慢一点。

Robot.Speed (-150,-150); //back off

这里，我们就用到通过Speed函数来调整速度了。根据代码的说明，能否自己尝试变换一下小车的速度，行动方向等等。

**大改造**

代码就讲到这里，我们这里做的是一个避障机器人。其实，同样的传感器，改造一下就能成为跟踪机器人了。只是方式相反，这里是，检测到物体后避开，跟踪的话，则是检测到物体后，向物体方向靠近。想一下，给自己做个跟踪机器人。