

# AlphaBot

# 用户手册

Nov. 2016

V1.1

# 阅读提示

## 阅读建议

本手册部分内容可能会让用户阅读起来稍感困难，我们建议用户在使用本款小车之前，先稍微学习树莓派的 Python 编程或者 Arduino 编程，以便深入理解示例程序原理。由于篇幅所限，本手册不会涉及太多的程序原理。如果对这方面有更深入的探索，我们还是希望您可以直接查看程序的源代码，因为智能小车开发的实战经验比理论知识更为重要。

树莓派教程：<http://www.waveshare.net/study/portal.php?mod=list&catid=29>

Arduino 教程：<http://www.waveshare.net/study/portal.php?mod=list&catid=7>

由于小车的电气特性所限，相对于其他的开发板，用户应当更加注意安全、正确地使用本产品，以避免小车底板对树莓派或 Arduino 板造成不可逆的电器损坏。因此首次使用本产品的用户，应当尽量按照本手册的操作步骤进行操作。

## 获取产品资料

用户可通过微雪百科获取 AlphaBot 的详细资料，包括用户手册、原理图、示例程序、数据表等资源，以便正确、安全地使用本产品。

😊 资料路径：[www.waveshare.net/wiki/AlphaBot](http://www.waveshare.net/wiki/AlphaBot)



# 目录

AlphaBot 用户手册	1
阅读提示	2
阅读建议	2
获取产品资料	2
目录	3
产品概述	5
简介	5
主要功能	5
功能亮点	5
受众	6
资源简介	7
模块功能	8
主控模块	8
电机驱动模块	9
电源部分	11
AD 采集芯片	11
避障模块	12
测速模块	13
红外遥控	14
超声波测距模块	16

舵机	17
寻迹模块	18
NRF24L01+无线控制	19
蓝牙控制	20
ZigBee 控制	21
WiFi 视频监控	22
示例程序	23
Arduino 示例程序	23
树莓派示例程序	27
Arduino 和树莓派联合控制	35
安全须知	36
注意请保留此信息以供将来参考	36

# 产品概述

## 简介

AlphaBot 智能车开发套件，主要包括 AlphaBot 底板和智能车底盘，支持接入 Arduino 和树莓派。

## 主要功能

Arduino 和树莓派都是目前热门的开发板，机器智能车也是很多电子爱好者的热衷项目，而 AlphaBot 让这三者结合到了一起。通过把 AlphaBot 接入树莓派或 Arduino，配合微雪提供的开源代码和学习教程，便能快速了解和探索智能小车的学习和开发。

AlphaBot 能够感知环境并且做出反应，这其中涉及到的技术要点：

循迹、避障、手机/PC 视频监控、WIFI/蓝牙/ZigBee/红外无线遥控等。

**注：此用户手册不针对某个特定的套餐，可能有些套餐不包含下文介绍到的部分模块，套餐配置以网页介绍中详细配置为准。**

## 功能亮点

- 板载 Arduino 和树莓派接口，可任选其一接入，也可同时接入协同工作。
- 板载 Arduino 和树莓派扩展接口，可接入树莓派或 Arduino 功能扩展板。
- 接口模块化，避障、循迹、测速等模块即接即用，无需繁琐杂乱的杜邦接线。
- 板载 L298P 电机驱动芯片，给车子提供动力，外加二极管保护电路，为车子保驾护航。
- 板载 LM2596 稳压芯片，可给树莓派或 Arduino 提供稳定的 5V 电压。
- 板载 TLC1543AD 采集芯片，方便树莓派接入模拟传感器。

## 受众

许多单片机爱好者都曾经想设计一台属于自己的智能小车。然而当真正动手时却发现并不像想象中简单，可能一知半解甚至毫无头绪。本套件能让您快速了解智能小车的结构框架、基本原理，以及外部模块。

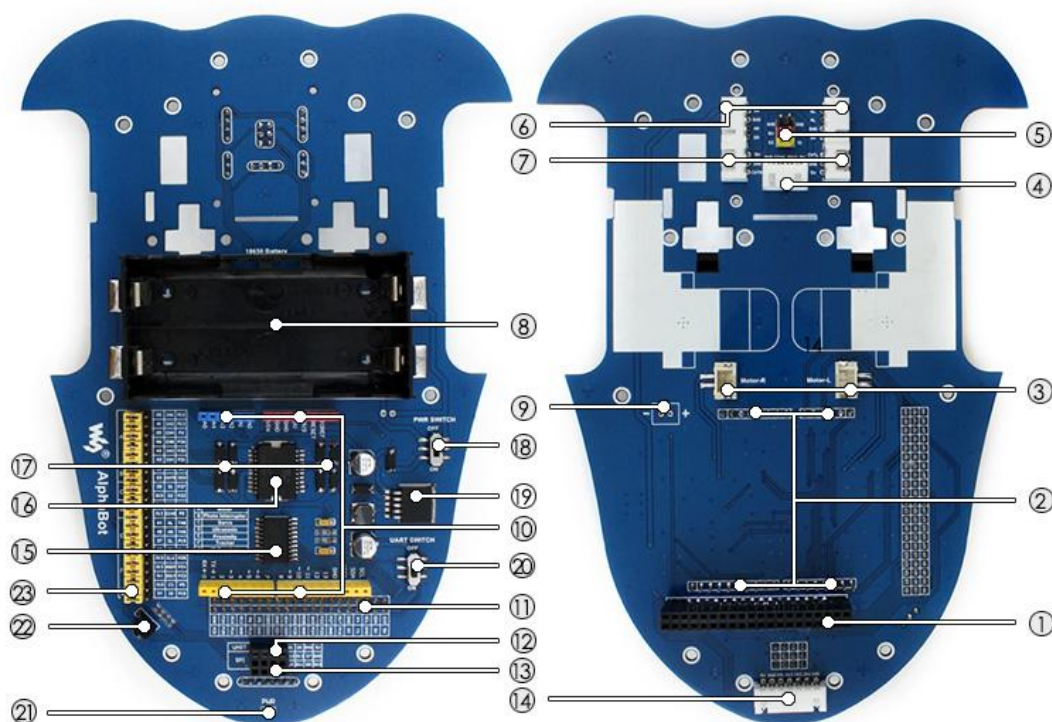
本教程主要通过微雪 AlphaBot 智能小车，讲解智能小车的工作原理和制作过程。示例程序包括循迹，自动壁障，蓝牙遥控，WiFi 控制等。本手册围绕示例程序展开，内容深入浅出，从最小系统到整个小车的主控设计，从简单的小车直线运动到复杂的多功能复合实现。希望单片机初学者可以通过本套件对智能小车有清楚的认识。由于作者水平有限，教程中错误和不足之处在所难免，敬请读者批评、指正。

---

☺ 有任何疑问，都欢迎来到微雪学堂 <http://www.waveshare.net/study> 留言。

---

# 资源简介



## [ 接口简介 ]

### 1. Raspberry Pi 接口

方便接入树莓派

### 2. Arduino 接口

方便接入 Arduino

### 3. 电机接口

### 4. 超声波模块接口

### 5. 舵机模块接口

### 6. 避障模块接口

### 7. 测速模块接口

### 8. 电池座接口

### 9. 预留电源接口（未焊接）

可以接入其它电池给小车供电

### 10. Arduino 扩展接口

方便接入 Arduino 扩展板

### 11. Raspberry Pi 扩展接口

方便接入树莓派扩展板

### 12. UART 接口

方便接入蓝牙模块以实现蓝牙控制小车

### 13. SPI 接口

方便接入 NRF24L01 无线模块

### 14. 循迹模块接口 [ 器件简介 ]

### 15. TLC1543

10 位 AD 采集芯片，方便树莓派接入模拟传感器

### 16. L298P

双 H 桥电机驱动芯片，最大电流可达 2A

### 17. 二极管

防止方向电流烧坏芯片

### 18. 电源开关

### 19. LM2596

5V 电源稳压芯片

### 20. UART 切换开关

拨到 on 一端可实现树莓派和 Arduino 串口通信

### 21. 电源指示灯

### 22. 红外接收器

实现红外遥控小车

## [ 跳线说明 ]

### 23. Arduino 或树莓派选择跳线

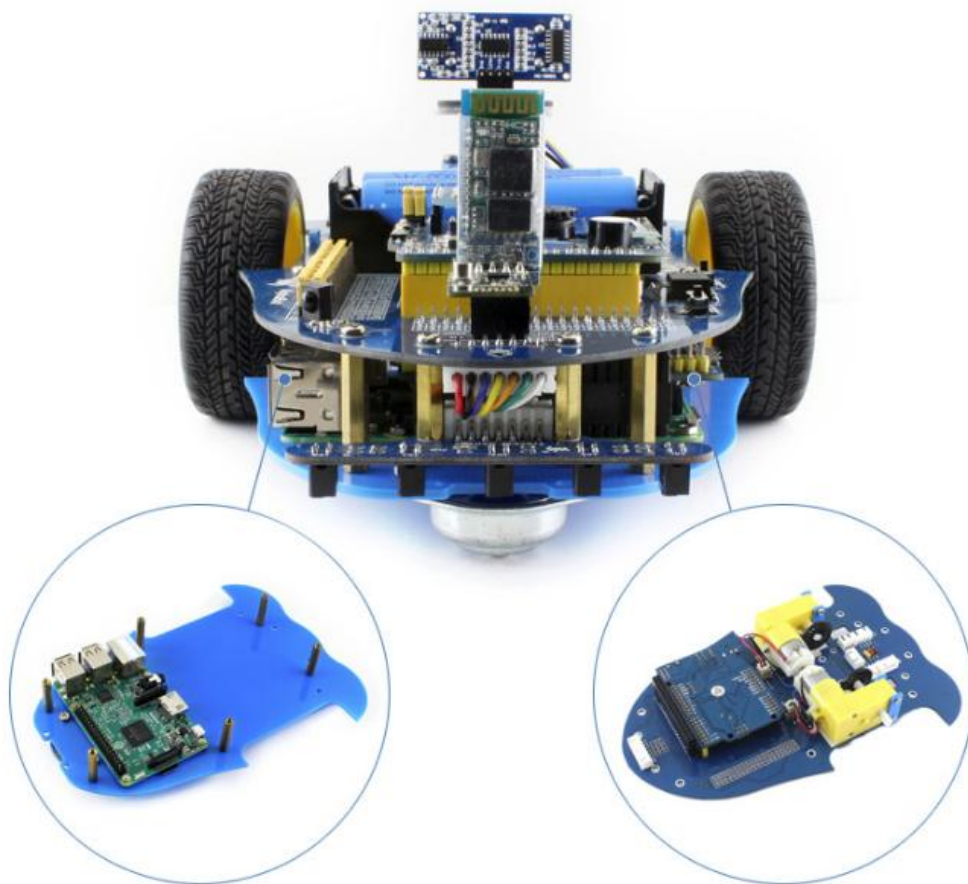
可选择树莓派或者 Arduino 控制小车外设

# 模块功能

## 主控模块

主控模块即整个小车的核心控制板，AlphaBot 留有 Arduino 以及 Raspberry Pi 接口，可单独接入树莓派或 Arduino 板，也可以同时接入，协同工作。充分利用两者的硬件和软件资源，使得小车拥有更广阔的应用场景。

**Raspberry Pi 或 Arduino**  
**可单独接入，也可同时接入协同工作**



---

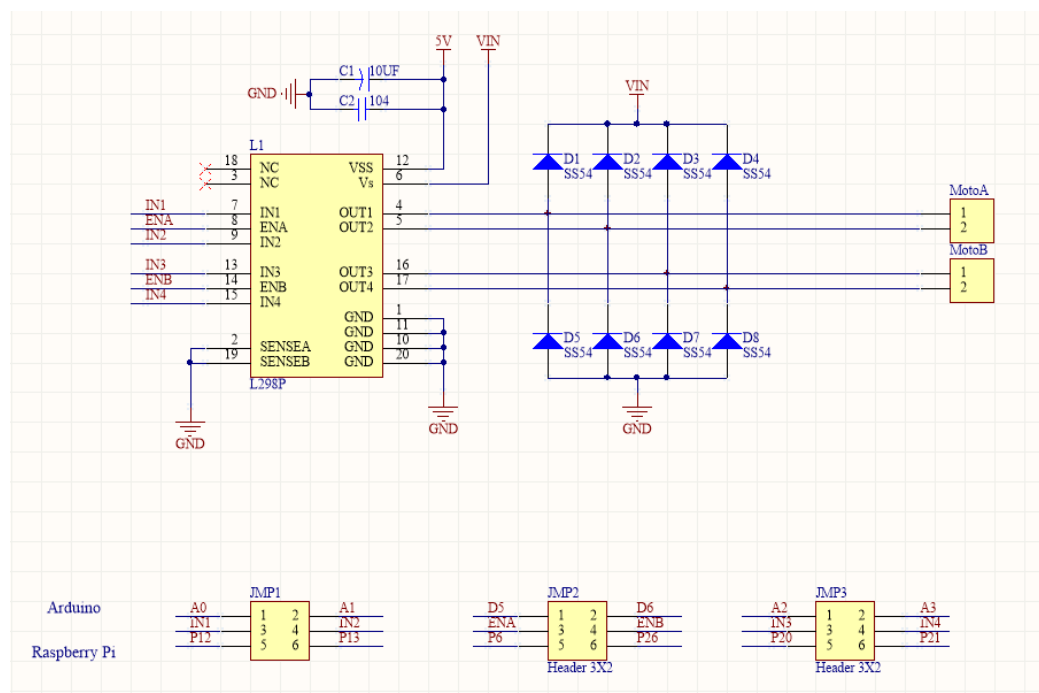
☺ Arduino 和树莓派都可以独立控制外设，通过跳线选择。

---



## 电机驱动模块

电机驱动模块是小车最重要的模块之一，AlphaBot 使用的是 L298P 驱动芯片。L298P 是 ST 公司生产的一种高电压、大电流电机驱动芯片。



## 驱动模块接口定义：

接口	树莓派	Arudino
IN1	P12	A0
IN2	P13	A1
ENA	P6	D5
IN3	P20	A2
IN4	P21	A3
ENB	P26	D6

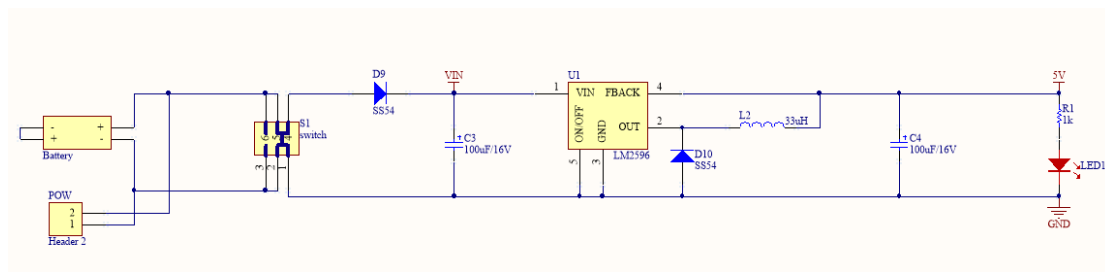
其中 IN1 和 IN2 接小车左电机 ,IN3 和 IN4 接小车右电机。ENA 和 ENB 为输出使能管脚，高电平有效，可以让 IN1、IN2、IN3 和 IN4 输出 PWM 脉冲实现小车调速。

## 控制原理：

IN1	IN2	IN3	IN4	描述
1	0	0	1	左右电机正转，小车前进
0	1	1	0	左右电机反转，小车后退
0	0	0	1	左电机停止，右电机正转，小车的左转
1	0	0	0	右电机停止，左电机正转，小车的右转
0	0	0	0	左右电机停止，小车停止

## 电源部分

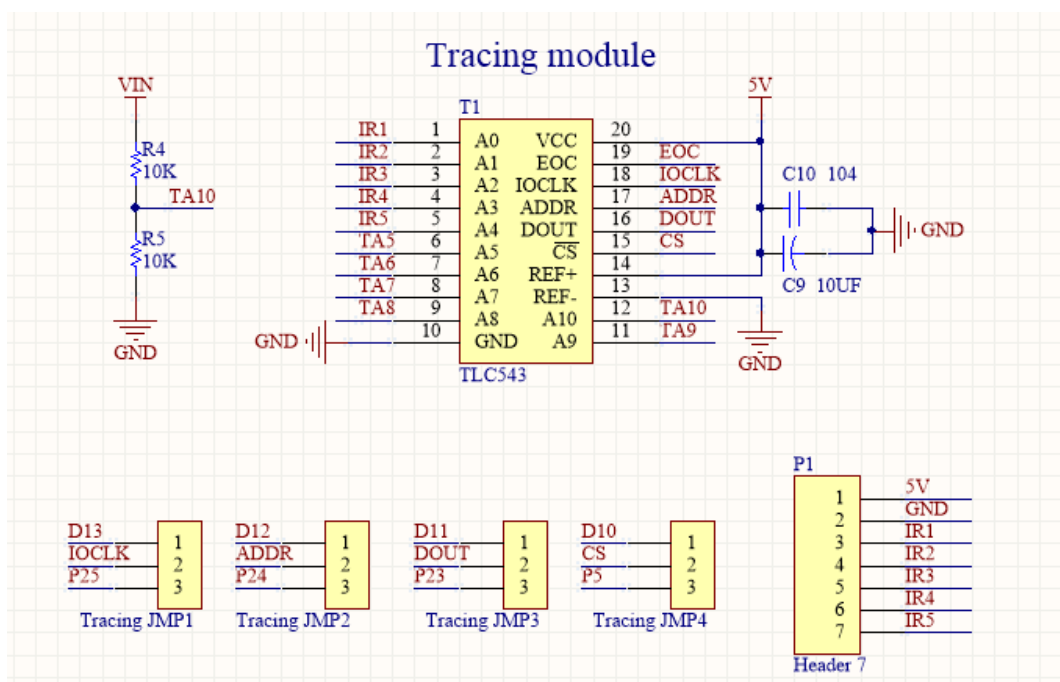
小车采用两节 18650 串联供电，输入电压约为 7.4V（单节电池 3.7V）。经过 LM2596 稳压后为 Arduino、树莓派，传感器供电。



**！** 使用外部供电的时候，电压不应超过 12 V。

## AD 采集芯片

由于树莓派的 GPIO 没有 AD 功能，故 AlphaBot 小车板载一个 10 位 11 通道的 AD 转换芯片。循迹功能会用到。



## 避障模块

避障模块主要由红外收发

器以及比较器组成。红外

发射管是由红外发光二极

管组成发光体，当检测

方向遇到障碍物（反射面）

时，红外线反射回来被接

收管接收，红外接收管是

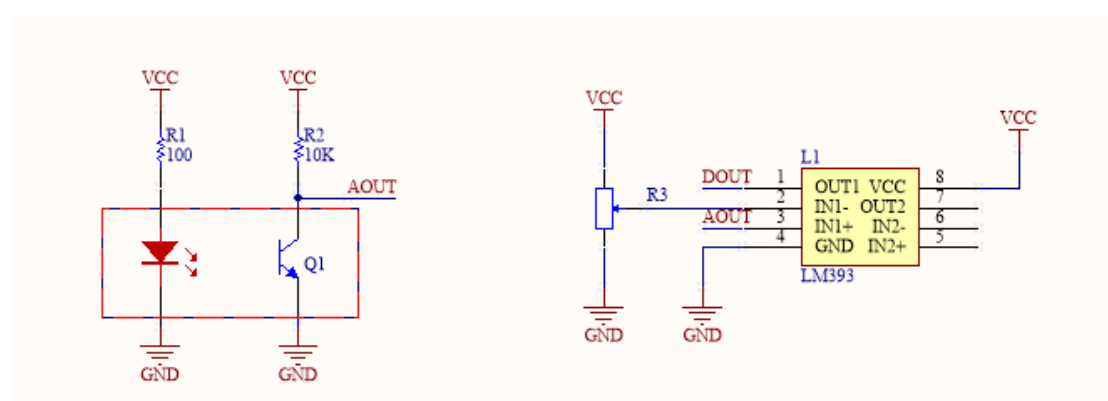
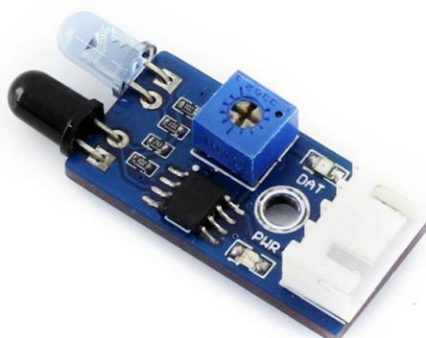
将红外线光信号变成电信

号的半导体器件，它的核心部件是一个特殊材料的 PN 结，随着红外光强度的增加电流也随

之增大输出模拟信号，经过 LM393 比较器处理后输出数字信号，同时绿色指示灯点亮。通

过读 DOUT 的逻辑电平，就可以判定前方是否有障碍物，读 AOUT 的输出电压可以判定距

离。可通过电位器旋钮调节检测距离。



## 测速模块

车速模块主要由码盘结合

WYC-H206 槽型光电传感器

构成。WYC-H206 传感器一

端为红外发射光,另一端为红

外接收,当有遮挡时,红外接

收端接收不到红外,输出高电

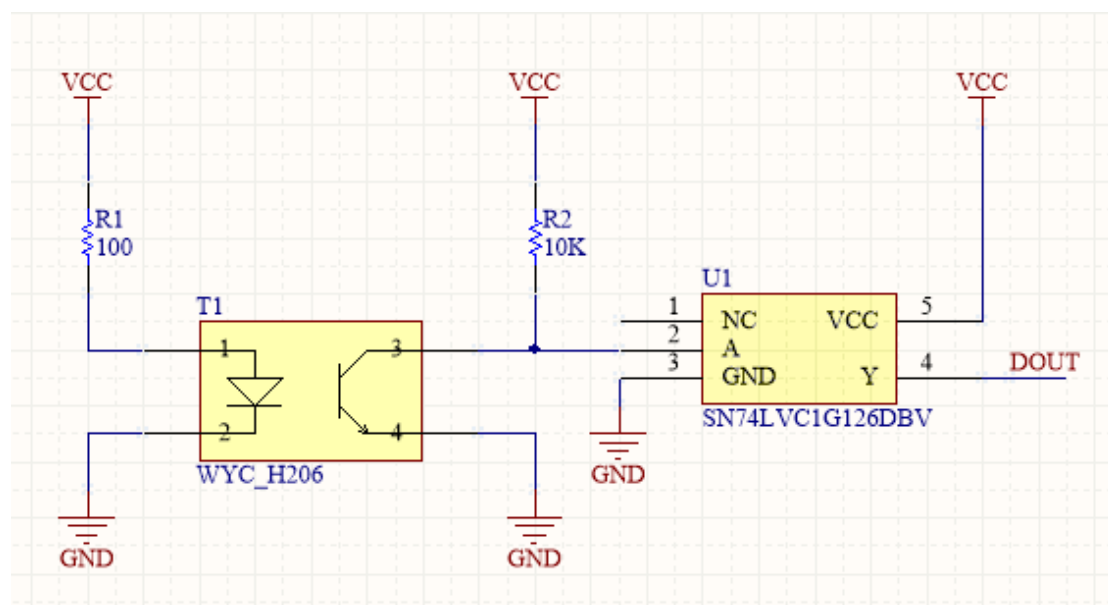
平,经过反相施密特触发器变

为低电平输出,同时信号灯被

点亮。当码盘转动时,DOUT

会输出高低脉冲,测量单位时间内脉冲个数既可算得小车速度。采用施密特触发器使输出信

号更加稳定,波形好,没有抖动现象。

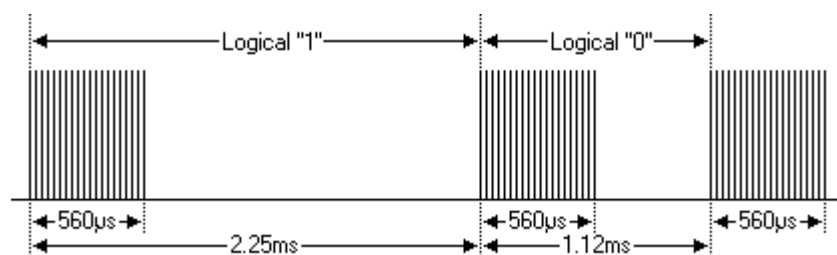


## 红外遥控

小车采用 LFN0038K 接收头,可以接受红外信号。配套的红外遥控器输出的脉冲信号遵循标准的 NEC 编码协议。红外接收器接收到信号之后,会把脉冲输出到主控芯片的接收管脚。主控对接收到的信号解码之后得到按键码,然后控制对应的电机。

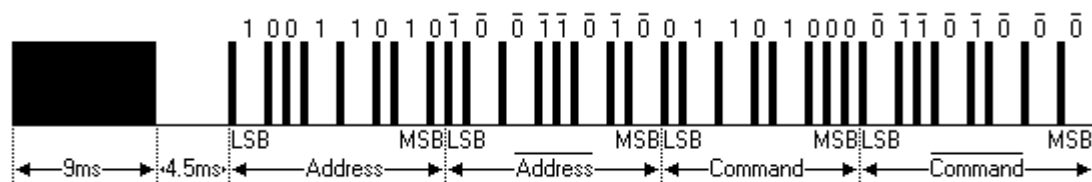


红外 NEC 协议编码采用脉宽调制的串行码,以脉宽为 0.565ms、间隔 0.56ms、周期为 1.125ms 的组合表示二进制的"0";以脉宽为 0.565ms、间隔 1.685ms、周期为 2.25ms 的组合表示二进制的"1"



## 红外控制协议：

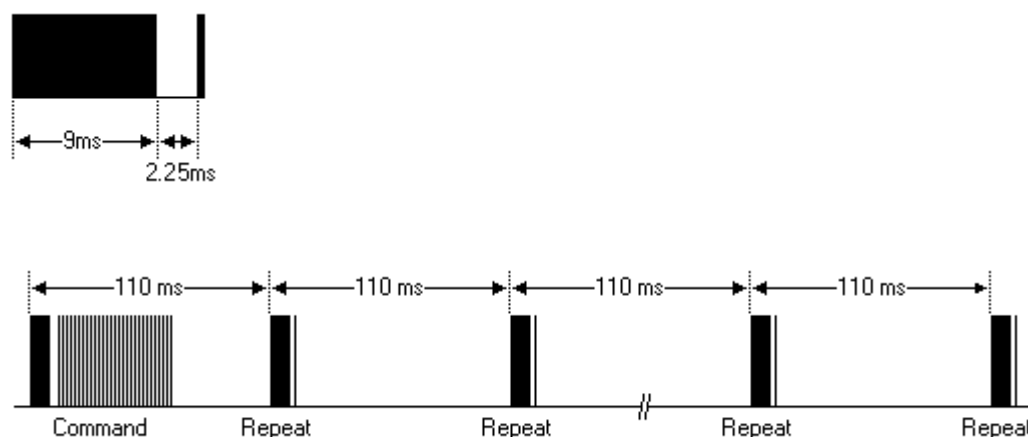
上述“0”和“1”组成的32位二进制码经38kHz的载频进行二次调制以提高发射效率，达到降低电源功耗的目的。然后再通过红外发射二极管产生红外线向空间发射，如下图。



|引导码|用户识别码|用户识别码反码|操作码|操作码反码|

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。

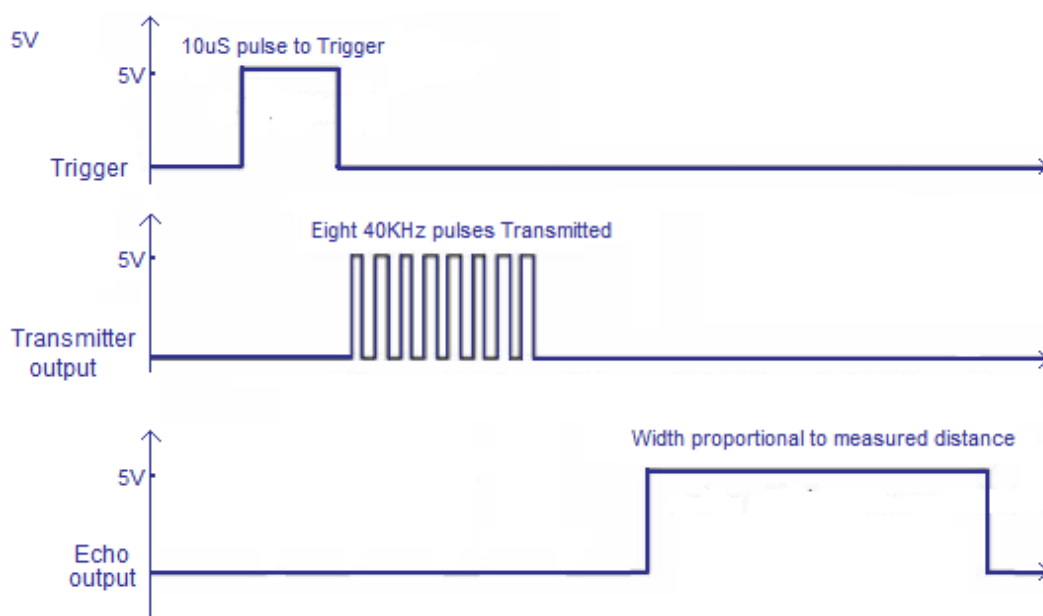
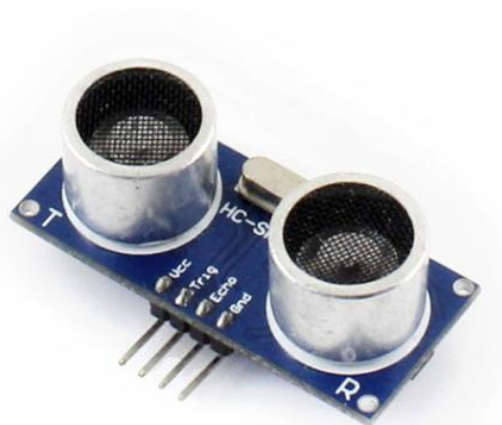
一个命令只发送一次，即使遥控器上的按键一直按着。但是会每110ms发送一次重复码，直到遥控器按键释放。重复码比较简单：一个9ms的AGC脉冲、2.25ms间隔、560uS脉冲。



## 超声波测距模块

超声波测距模块有四个针脚，除了电源(VCC)和地(GND)，还有发射管脚 TRIG 和接收管脚 ECHO。

超声波测距原理是当发送的超声波遇到物体后被反射回来，被接收端接收。通过发送信号到接收到回响信号时间间隔可以算得距离。



以上时序图表明超声波的控制原理。模块工作时 IO 口给至少 10us 的高平信号控制 TRIG 触发测距。模块将自动发射 8 个 40kHz 的方波，并且自动检测是否有信号返回。若有信号返回，通过 ECHO 输出一个高电平，而高电平的持续时间就是超声波发射到返回的时间。根据声音传播的速度和时间我们可以测出距离：测量距离 = ( 测量时间 \* 声音速度 ) / 2。声音的传播速度通常按 340m/s 来计算。



## 舵机



舵机由直流电机、减速齿轮组、传感器和控制电路组成的一套自动控制系统。通过发送信号，指定输出轴的旋转角度。舵机一般由三根线控制，接线方法：红色为电源正（5v），橙色为信号线（IO），褐色为电源地（GND）。

☺ 控制信号由接收机的通道进入信号调制芯片，获得直流偏置电压。

它内部有一个基准电路，产生周期为 20ms，宽度为 1.5ms 的基准信号，将获得的直流偏置电压与电位器的电压比较，获得电压差输出。最后，电压差的正负输出到电机驱动芯片决定电机的正反转。当电机转速一定时，通过级联减速齿轮带动电位器旋转，使得电压差为 0，电机停止转动。

舵机的控制一般需要一个 20ms 左右的时基脉冲，该脉冲的高电平部分一般为 0.5ms-2.5ms。以 180 度角度伺服为例，对应的控制关系是这样的：

脉宽	旋转角度
0.5ms	0 度
1.0ms	45 度
1.5ms	90 度
2.0ms	135 度
2.5ms	180 度

## 循迹模块

循迹模块和红外避障模块

类似，循迹模块采用

ITR20001/T 红外反射传感

器，ITR2001/T 传感器的红

外发射二极管不断发射红

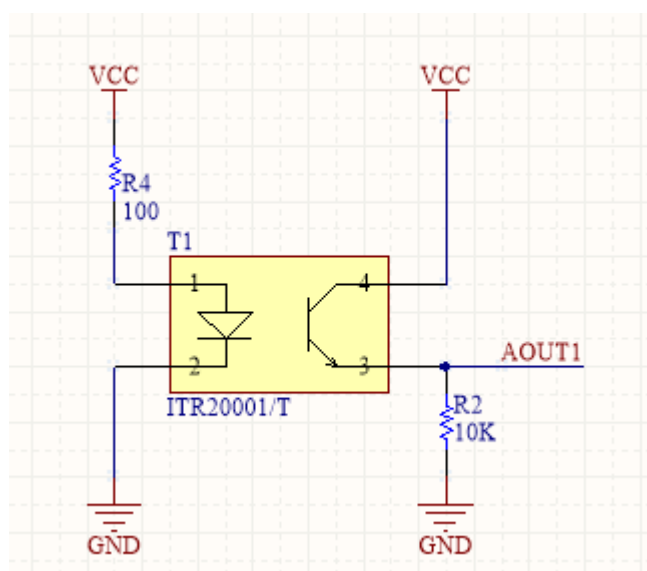
外线，当发射出的红外线被



物体反射时，被红外接收器就收，并输出模拟值。输出模拟值和物体距离以及物体颜色有关。

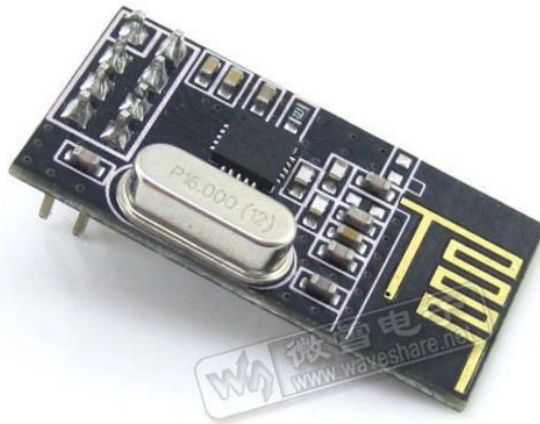
循迹模块采用五路红外传感器，通过测量五路红外传感器，判断黑线的位置，从而控制小车运动。

☺ 具体的控制原理及算法可以查看 Tracker Sensor 用户手册。



## NRF24L01+ 无线控制

AlphaBot 留有 SPI 接口，可以直接接入 NRF24L01 模块。NRF24L01 是一款新型射频收发器件，内置了频率合成器、功率放大器、晶体振荡器、调制器的功能，融合了增强型 ShockBurst 技术，其中地址、输出功率和通信频道都可以通过程序进



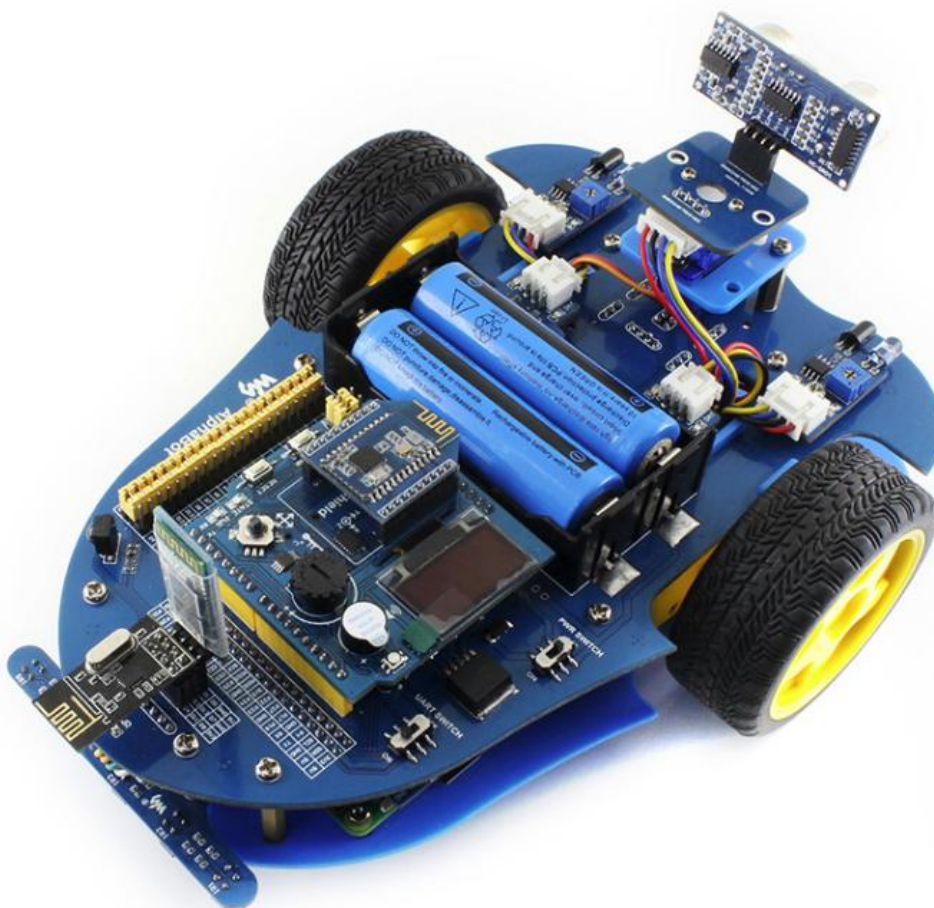
行配置，适用于多机通信。其功耗很低，在以-6dBm 的功率发射时工作电流只有 9mA；而对应接收机的工作电流只有 12.3mA，多种低功耗工作模式（掉电模式、空闲模式）。

通过 SPI 接口配置 NRF24L01 模块，通常情况下将模块配置成 Enhanced ShockBurst™ 收发模式，能够使模块自动处理射频协议。完成配置后，在模块工作过程中只需要改变最低一个字节的内容，实现接收模式和发射模式的切换。

主要参数设置：数据宽度，声明射频数据包中数据占用的位数。使 NRF24L01+能够区分接收数据包中的数据和 CRC 校验码；地址宽度，声明射频数据包中地址占用的位数，能使 NRF24L01+区分地址和数据；接收通道，接收数据的地址，有通道 0 到通道 5 的地址；CRC 校验，使 NRF24L01+能够生成 CRC 校验码和解码；工作频率，可以设置在 2.4-2.524GHz 之间，间隔 1MHz；发射功率，其大小决定了通信距离和整体的功耗；通信速率，1Mbps、2Mbps 通信速率可选择配置。

## 蓝牙控制

AlphaBot 留有 UART 接口，可以直接接入蓝牙 2.0 模块，或者通过 Accessory Shield 模块接入双模蓝牙模块。蓝牙模块是通过串口控制，无须了解任何蓝牙协议。接入蓝牙模块后可以通过手机端控制小车。



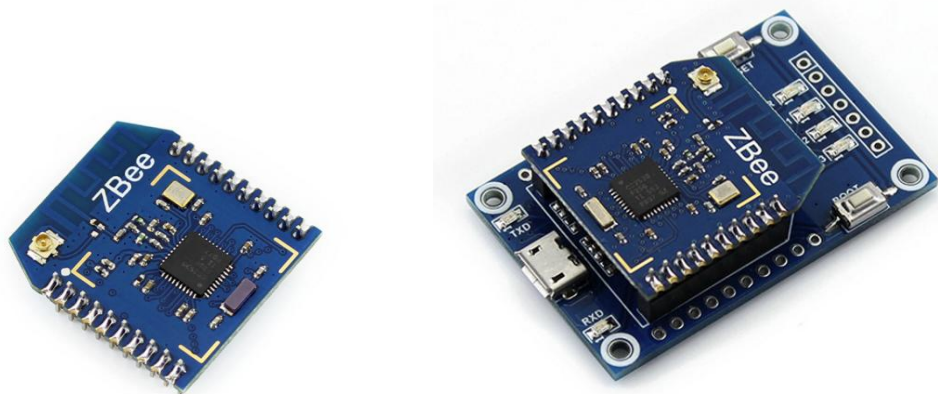
---

**！** 图中接到 Accessory Shield 板上的是双模蓝牙模块，支持蓝牙 2.0 和 4.0 两种模式。我们蓝牙套餐也是配这个。接到 UART 接口上的是蓝牙 2.0 模块，这个我们是没有配的。

---

## ZigBee 控制

AlphaBot 可以通过 Accessory Shield 模块接入 Core2530 (B)可以通过 ZegBee 控制 , 电脑端通过 USB 连接 CC2530 Eval Kit (B)。



通过上位机软件控制小车：





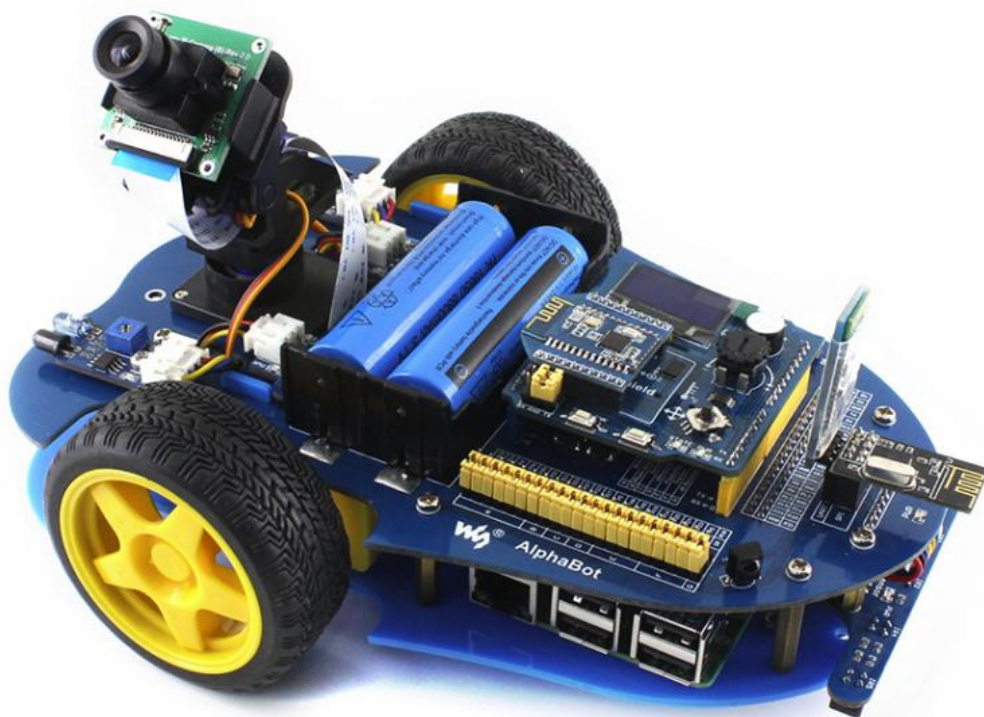
## WiFi 视频监控

AlphaBot 用树莓派作为主控时，可以用手机端或者电脑通过网页实现 WiFi 视频控制功能。

通过 webiopi 实现远程遥控，通过 mjpg-streamer 软件实时显示摄像头拍摄图片。

webiopi 是一个可控制树莓派 GPIO 的 web 框架，该框架面向物联网 IoT 开发。该项目托管于 Google，并可在 sourceforge 上下载到源代码。webiopi 支持 REST Server，CoAP server，并提供 python 库和 javascript 库。

MJPEG-streamer，是用于从 webcam 摄像头采集图像，把他们以流的形式通过基于 ip 的网络传输到拥有浏览器的移动设备。通过 mjpg-streamer 软件可以实现将摄像头中采集到的图像通过网络传输到浏览器网页上显示。



## 示例程序

我们有提供 Arduino 和树莓派的控制程序。可以实现循迹，避障，遥控，视频等功能。

下面的示例程序都可以从微雪百科获取。将 Arduino 目录下的库文件复制到 Arduino IDE 安装目录下 libraries 文件夹中。重启 Arduino IDE，点击 File-> Example，查看是否有 AlphaBot，TRSensors，Servo 等选项，若有说明库导入成功。树莓派程序解压并放在 Raspbian 系统的/home/pi 目录下。

## Arduino 示例程序

### 小车测试程序

点击 File-> Example ->AlphaBot ->E01\_Run\_Test 打开工程。编译并下载到 Arduino 开发板。跳线 A 区 IN1, IN2, ENA, ENB, IN3, IN4 分别接 A0, A1, D5, D6, A2, A3。打开电源开关，小车实现前进运行。若小车的两个轮子不是向前运行，可以更换一下电机接线或者修改 AlphaBot.c 文件管脚设置。

### 红外避障程序

点击 File -> Example ->AlphaBot ->E02\_Infrared\_Obstacle\_Avoidance 打开工程。编译并下载到 Arduino 开发板。小车接上左右 Infrared Proximity Sensor 传感器，电机 A 区跳线接到 Arduino 一端，E 区跳线 DL, DR 分别接 D7, D8。打开电源开关，小车实现红外避障功能。当传感器前方有障碍物时，绿色 LED 灯会亮，调节电位器可以改变感应距离。左边传感器检测到障碍物小车会右转，右边传感器检测到障碍物会左转。

## 红外物体追踪程序

---

点击 File -> Example -> AlphaBot -> E03\_Infrared\_Tracking\_Objects 打开工程。编译并下载到 Arduino 开发板。电机 A 区跳线接到 Arduino 一端, E 区跳线, DL, DR 分别接 D7, D8。打开电源开关, 小车实现物体追踪功能, 若小车前方有物体, 小车会前进。若小车前方没有物体, 小车静止。调节电位器可以改变感应距离。

## 红外遥控程序

---

点击 File -> Example -> AlphaBot -> E04\_Remote\_control 打开工程, 编译并下载到 Arduino 开发板。电机 A 区跳线接到 Arduino 一端, G 区红外接收部分跳线 IR 的接到 Arduino 一端 D4。

预期结果: 按下红外遥控器的数字键控制小车。2, 8, 4, 6, 5 分别代表前进, 后退, 左转, 右转, 停止。按-或+可调节速度。(注: 不同的红外遥控器可能按键编码不一样, 如果不一样需相应修改程序)

## 超声波测距程序

---

点击 File-> Example -> AlphaBot -> E05\_Ultrasonic\_Rangin 打开工程, 编译并下载到 Arduino 开发板。小车接上舵机以及超声波模块, D 区超声波接部分跳线 TRIG, ECHO 的接到 Arduino 一端 D11, D12 管脚。(红外循迹也要这些管脚, 应该去掉相应跳线, 避免干扰)。

打开电源, 串口会显示当前超声波测量的距离。



## 超声波避障程序(不带舵机)

---

点击 File -> Example -> AlphaBot -> E06\_Ultrasonic\_Obstacle\_Avoidance1 打开工程，编译并下载到 Arduino 开发板。小车接上舵机以及超声波模块，D 区超声波接部分跳线 TRIG, ECHO 的接到 Arduino 一端 D11, D12 管脚。

打开电源，小车会实现超声波避障功能呢。当前方有障碍物时，超声波模块会自动转向前进。

## 超声波避障程序(带舵机)

---

点击 File -> Example -> AlphaBot -> E07\_Ultrasonic\_Obstacle\_Avoidance2 打开工程，编译并下载到 Arduino 开发板。小车接上舵机以及超声波模块，D 区超声波接部分跳线 TRIG, ECHO 的接到 Arduino 一端 D11, D12 管脚。C 区舵机部分跳线 S1 接到 Arduino 一端 D9 管脚。(红外循迹也要这些管脚，应该去掉相应跳线，避免干扰)。

打开电源，小车会实现超声波避障功能呢。当前方有障碍物时，超声波模块会测量左边和右边的障碍物的距离，小车会选择障碍物距离远的一向运动。

## 红外循迹程序

---

点击 File -> Example -> AlphaBot -> E08\_Infrared\_Line\_Tracking 打开工程，编译并下载到 Arduino 开发板。小车接上 Tracker Sensor 寻迹模块，电机 A 区跳线接到 Arduino 一端，F 区寻迹部分跳线 CLK, ADDR, DOUT, CS 的分别接到 Arduino 一端 D13, D12, D11, D10。(舵机 C 区，超声波 D 区也是用到这些管脚，应该去掉相应跳线，避免干扰)。

寻迹程序实现小车沿着黑线运行，底色为白色。可以用黑线胶布粘在白色的 KT 板中制作小车循迹场地，黑色轨迹宽度为 15cm。若底色颜色过深会影响循迹效果。

打开电源开关，用手拿着小车在黑线中紧贴地面左右摇晃，此阶段为校准，使小车可以设别

各个传感器的最大值和最小值。校准效果直接影响到循迹效果。直到小车轮子开始转动时开放小车，小车会沿着黑色线运行。

详细的原理以及算法等可以参考 Tracker Sensor 模块用户手册。

## 蓝牙遥控程序

---

点击 File-> Example -> AlphaBot -> E09\_Bluetooth\_Control 打开工程，编译并下载到 Arduino 开发板。将 Accessory Shield 模块接到小车 Arduino 接口上，将蓝牙模块接到 Accessory Shield 模块的 XBee 接口上 另外设置 Accessory Shield 上的跳线帽 :RXD $\leftrightarrow$ TX、TXD $\leftrightarrow$ RX。打开电源，下载蓝牙控制软件并安装到手机中，打开软件并连接上对应蓝牙，按下方向键可以控制小车运动。发送字符串 OLED 会显示相应的字符。

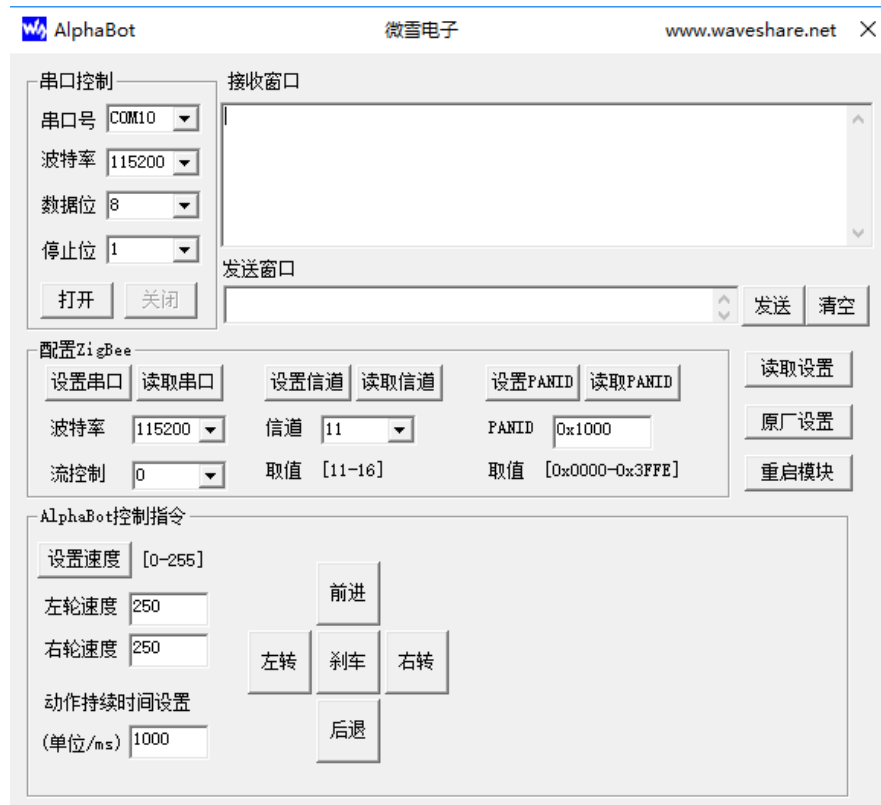
## ZigBee 遥控程序

---

点击 File -> Example -> AlphaBot -> E10\_Zigbee\_Control 打开工程，编译并下载到 Arduino 开发板。将 Accessory Shield 模块接到小车 Arduino 接口上，将 ZigBee 模块 Core2530 (B)接到 Accessory Shield 模块的 XBee 接口上，另外设置 Accessory Shield 上的跳线帽：RXD $\leftrightarrow$ TX、TXD $\leftrightarrow$ RX。。用 USB 线将 CC2530 Eval Kit5 开放套件连接到电脑。

打开小车电源，电脑端打开 AlphaBot.exe 程序，选择对应的串口号。电机按键即可控制小车运动。发送命令字符串 OLED 会显示相应的字符。

通过上位机软件控制小车：



## 树莓派示例程序

将 AlphaBot.tar.gz 解压并放在 raspbian 系统的/home/pi 目录下。

在使用本扩展板之前，我们需要给树莓派安装 python 函数库，树莓派的库函数安装和配置过程请参见为树莓派安装必要的函数库。

如果安装完函数库之后依然无法执行以下的示例程序，那么您可以尝试使用 `chmod +x filename` 命令给程序赋予可执行权限

## 红外避障程序

小车接上左右 Infrared Proximity Sensor 传感器，电机跳线 A 部分 IN1, IN2, ENA, ENB, IN3, IN4 分别接到树莓派一端 P12, P13, P6, P26, P20, P21 管脚。红外感应跳线 E 部分，DL, DR 分别接 P19, P16。

进入 Linux 终端，在终端执行以下命令。

```
cd ~/AlphaBot/python  
sudo python Infrared_Obstacle_Avoidance.py
```

预期结果：小车左边传感器检测到障碍物小车会右转，右边传感器检测到障碍物会左转。

## 红外物体追踪程序

---

小车接上左右 Infrared Proximity Sensor 传感器,电机跳线 A 部分 IN1, IN2, ENA, ENB, IN3, IN4 分别接到树莓派一端 P12, P13, P6, P26, P20, P21 管脚。红外感应跳线 E 部分, DL, DR 分别接 P19, P16.

进入 Linux 终端，在终端执行以下命令。

```
cd ~/AlphaBot/python  
sudo python Infrared_Obstacle_Avoidance.py
```

预期结果：小车实现物体追踪功能，若小车前方有物体，小车会前进。若小车前方没有物体，小车静止。

## 红外遥控程序

---

电机跳线 A 区接到树莓派一端，红外遥控跳线 G 区 IR 管脚接 P18.

进入 Linux 终端，在终端执行以下命令。

```
cd ~/AlphaBot/python  
sudo python IRremote.py
```

预期结果：按下红外遥控器的数字键控制小车。2, 8, 4, 6, 5 分别代表前进，后退，左转，右转，停止。按-或+可调节速度。（注：不同的红外遥控器可能按键编码不一样，如果不一样需修改相应程序）

## 红外循迹程序

---

电机跳线 A 区接到树莓派一端，寻迹跳线 F 区 CLK，ADDR,DOUT,CS 管脚分别接 P25，P24, P23, P5.

进入 Linux 终端，在终端执行以下命令。

```
cd ~/AlphaBot/python
sudo python TRSensors.py
```

预期结果：打开电源开关，用手拿着小车在黑线中紧贴地面左右摇晃，此阶段为校准，使小车可以设别各个传感器的最大值和最小值。校准效果直接影响到循迹效果。直到小车轮子开始转动时开放小车，小车会沿着黑色线运行。

## 超声波避障程序

---

小车接上舵机以及超声波模块，D 超声波跳线 D 区 TRIG,ECHO 的接到树莓派一端 P17,P5 管脚。(红外循迹也要这些管脚，应该去掉相应跳线，避免干扰)

进入 Linux 终端，在终端执行以下命令。

```
cd ~/AlphaBot/python
sudo python Ultrasonic_Obstacle_Avoidance.py
```

预期结果：小车会实现超声波避障功能呢。当前方有障碍物时，超声波模块会测量左边和右边的障碍物的距离，小车会选择障碍物距离远的一向运动。

## WiFi 控制视频程序

---

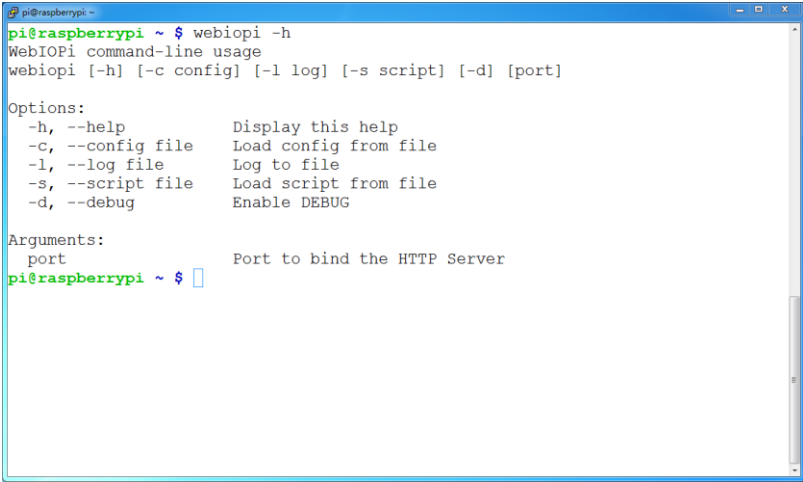
本实验通过 webiopi 实现远程遥控，通过 mjpg-streamer 软件实时显示摄像头拍摄图片。

部分浏览器可能不能正常显示，建议使用谷歌浏览器或火狐浏览器。

### 1. 安装 webiopi

```
cd ~/AlphaBotmjpg-AlphaBot/WebIOPi-0.7.1-raspi2
sudo ./setup.sh
```

验证是否安装成功：执行 `webiopi-h`，如果出现以下界面，说明库安装成功。



```
pi@raspberrypi ~ $ webiopi -h
WebIOpi command-line usage
webiopi [-h] [-c config] [-l log] [-s script] [-d] [port]

Options:
  -h, --help            Display this help
  -c, --config file     Load config from file
  -l, --log file        Log to file
  -s, --script file     Load script from file
  -d, --debug           Enable DEBUG

Arguments:
  port                  Port to bind the HTTP Server
pi@raspberrypi ~ $
```

终端执行：

```
sudo webiopi -d -c /etc/webiopi/config
```

然后在电脑端或者手机端打开网页浏览器，在地址栏内输入树莓派 ip 地址，端口号 8000。

例如：`http://192.168.1.16:8000`（根据实际情况填入）。在登录网站时，需要输入账号和密码，默认的账号和密码分别是：`webiopi` 和 `raspberrypi`。如果进入 WebIOpi Main Menu 页面，说明环境配置成功了：

## WebIOpi Main Menu

### [GPIO Header](#)

Control and Debug the Raspberry Pi GPIO with a display which looks like the physical header.

### [GPIO List](#)

Control and Debug the Raspberry Pi GPIO ordered in a single column.

### [Serial Monitor](#)

Use the browser to play with Serial interfaces configured in WebIOpi.

### [Devices Monitor](#)

Control and Debug devices and circuits wired to your Pi and configured in WebIOpi.

终端按下 **Ctrl + C** 可以结束进程。

## 2. 安装和使用 mjpg-streamer

这部分为小车增加摄像头功能，需树莓派接上摄像头。如不需要则这部分可跳过。

a) 使用摄像头前必须运行 `raspi-config` 命令启用摄像头：

```
sudo raspi-config
```

选择 Enable Camera。

- b) 如果是使用 CSI 接口的摄像头，那么系统找不到/dev/video0 的设备节点。需要在

/etc/modules 文件中添加一行 bcm2835-v4l2：

```
sudo nano /etc/modules
```

加上：

```
bcm2835-v4l2
```

- c) 那么系统启动之后，系统会加载这个文件中的模块名。重启系统后会在/dev 下面发现

video0 设备节点。（bcm2835-v4l2 中的 l 为 L 的小写，不是 1）

- d) 安装依赖库：

```
sudo apt-get install libv4l-dev libjpeg8-dev
```

```
sudo apt-get install subversion
```

- e) 编译程序：

```
cd ~/RPi-Motor-Driver-Board-Demo-Code/mjpg-streamer/
```

```
make USE_LIBV4L2=true clear all
```

如果提示：make: \*\*\* No rule to make target 'clear'. Stop.

请忽略这个警告。

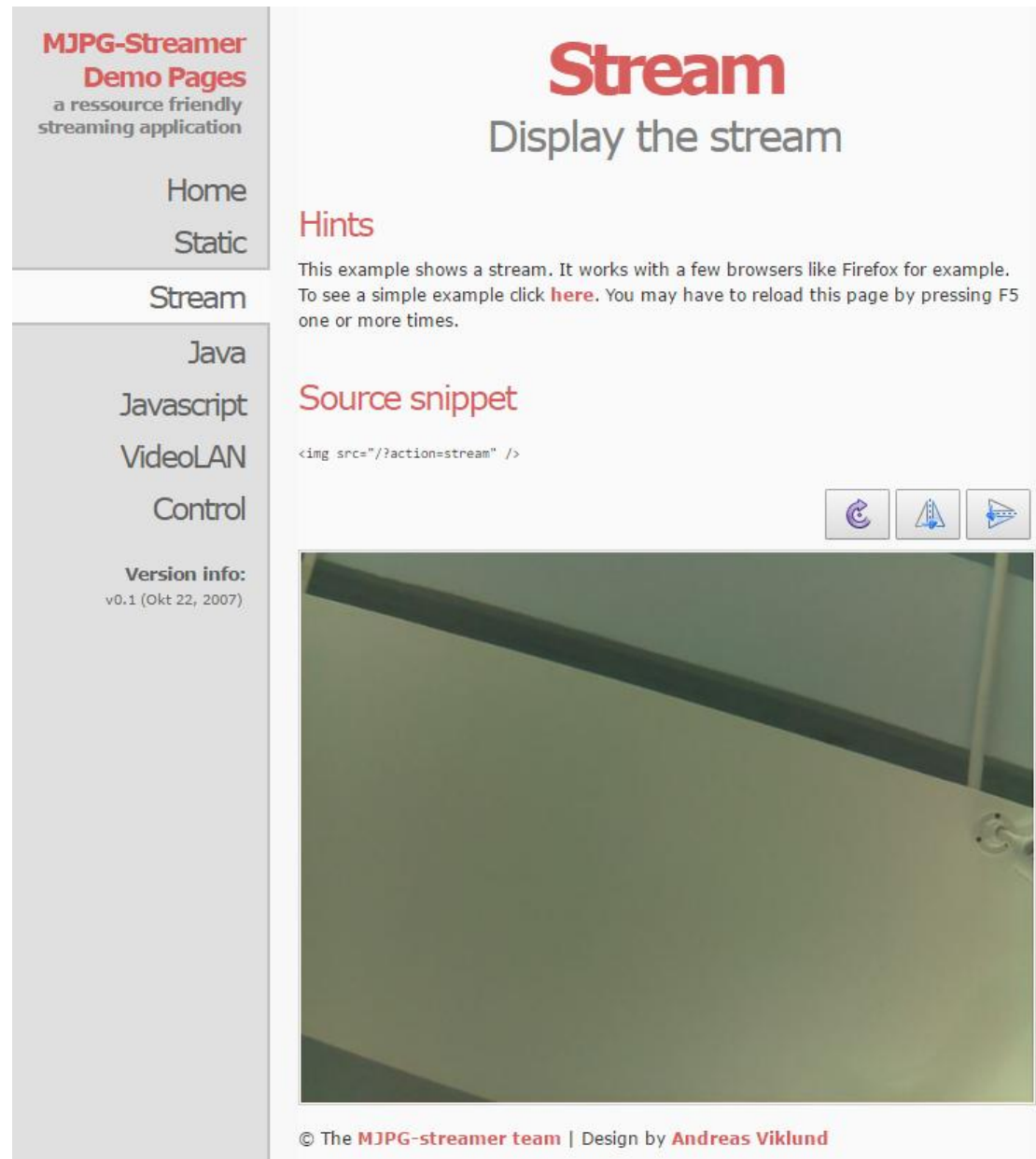
- f) 若编译报错，请尝试修改 mjpg-streamer/plugins/input\_uvc/input\_uvc.c 文件，

将 V4L2\_PIX\_FMT\_MJPEG 改成 V4L2\_PIX\_FMT\_YUYV

- g) 开启 mjpg-streamer：

```
sudo ./start.sh
```

然后在浏览器地址栏内输入树莓派 ip 地址，端口号 8080，注意这里的端口号是 8080，和之前的端口号（8000）是不同的。页面会显示关于 M-JPEG streamer 的信息。点击 Stream 会实时显示摄像头拍摄的内容。



虽然终端按下 **Ctrl + C** 可以结束进程，但是请不要结束这个进程，因为后面还需要用到。

### 3. 运行软件

请不要结束上一步的进程，而是重新开启一个终端。

进入 mjpg-AlphaBot 目录，这个目录同时也是小车控制源码所在的目录。

```
cd ~/mjpg-AlphaBot/mjpg-AlphaBot
```

修改 index.html 文件：

```
sudo nano index.html
```

根据树莓派的实际 ip，修改这个 html 标签：



```

```

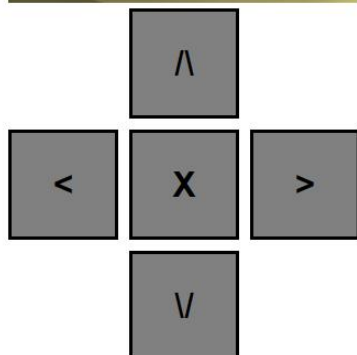
其中，src 后面的地址填入树莓派的 ip 地址（这里是 192.168.1.16），端口号指定为 8080（stream 的端口）。

启动树莓派摄像头小车程序：

```
cd ~/mjpg-AlphaBot/mjpg-AlphaBot
sudo webiopi -c config
```

然后在浏览器地址栏内输入树莓派 ip 地址，端口号 8000（注意这里的端口号不是 8080）。

网页上会显示摄像头捕获的画面，点击图标可以控制小车转向。



！ 如果没有出现这个页面，建议使用 Chrome 或者 Firefox 浏览器。此外还要确保树莓派的当前目录为：

~/mjpg-AlphaBot/mjpg-AlphaBot

这个目录下有一个 index.html 文件，然后再执行：

```
sudowebiopi -c config
```

这样客户端的浏览器才能进入 index.html，否则会进入 WebIOPi Main Menu 页面。

设置开机启动：

替换 config 文件

```
cd ~/mjpg-AlphaBot/mjpg-AlphaBot
sudo cp config /etc/webiopi/config
```

运行如下命令设置开机启动 webiopi.

```
sudo update-rc.dwebiopi defaults
```

用文本编辑器打开/etc/local 文件添加命令。

```
sudo vi /etc/rc.local
```

在注释后面添加如下命令，但是要保证 exit 0 这行代码在最后，然后保存文件退出。

```
cd /home/pi/ mjpg-AlphaBot/mjpg-streamer  
sudo ./start.sh
```

电机跳线 A 部分 IN1, IN2, ENA, ENB, IN3, IN4 分别接到树莓派一端 P12, P13, P6, P26, P20, P21 管脚。舵机跳线 C 部分 S1, S2 分别接 P27, P22.重启树莓派后，用浏览器地址栏内输入树莓派 ip 地址，端口号 8000，即可控制小车。

## Arduino 和树莓派联合控制

以上为仅用树莓派控制小车，舵机，小车调速采用模拟 PWM 控制，会占用 CPU，而且输出的 PWM 也不够稳定，可以通过 Arduino 和树莓派一起控制小车来达到更好的效果。

Arduino 负责控制小车电机，舵机等，树莓派负责视频传输以及网页控制等，Arduino 和树莓派通过串口通信。

1. 关闭树莓派的 shell 串口调试功能，并让串口独立工作。

替换 config 文件。

```
cd ~/mjpg-AlphaBot/ARPI-AlphaBot  
sudo cp config /etc/webiopi/config
```

选择 AdvancedOptions -> Serial -> <No>关闭串口调试功能

打开/boot/config.txt 文件，找到如下配置语句使能串口，如果没有，可添加在文件最后面。

```
enable_uart=1
```

2. Arduino 下载程序。

打开 Arduino IDE，点击 File -> Example -> AlphaBot -> mjpg-AlphaBot 打开工程，编译并下载

到 Arduino 开发板。小车接上 Arduino 和树莓派，电机跳线 A 部分 IN1, IN2, ENA, ENB, IN3, IN4

分别接到 Arduino 一端 A0, A1, D5, D6, A2, A3 管脚。舵机跳线 C 部分，S1, S2 分别接 D9, D10.

串口开关 UART SWITCH 拨到 ON 一端。重启树莓派后，用浏览器地址栏内输入树莓派 ip 地址，端

口号 8000，即可控制小车。

---

**！** 组装的时候应避免让 Arduino 和树莓派发生触碰。

---

# 安全须知

产品名称 智能车扩展开发套件 ,主要包括 AlphaBot 底板和智能车底盘。支持接入 Arduino 和树莓派开发板。

## 注意

### 请保留此信息以供将来参考

#### 警告

**AlphaBot 是精密器件，为避免 AlphaBot 出现故障或者损坏，请遵守以下要求：**

- 本产品通常使用两节 18650 电池供电，工作电压是 7.4 V。与底板与 Arduino 和树莓派搭配使用时的任何外部电源，都应遵守所在地区适用的电源相关法规和标准。
- 本产品应在通风良好的环境下使用，以便电源散热以及保证产品性能。
- 底板拆卸下来之后，应该放置在稳定、平坦、不导电的表面上。不应接触导电物品。
- 底板可以连接树莓派或者 Arduino 两者之一，也可以同时连接 Arduino 和树莓派，在这种情况下，应当避免让 Arduino 和树莓派发生触碰。
- 将不兼容的设备连接到底板可能会导致设备损坏、质保失效。
- 切勿在操作过程中接触水分和湿气，或放置在导电的表面上。
- 切勿让 AlphaBot 接触任何热源，以便该产品在正常室温环境下保持可靠运行。
- 操作过程应避免对印制电路板和其上的电子元器件造成机械或者电器损伤。
- 操作过程应避免触碰已通电的印制电路板，尽可能触碰其边缘，减少静电放电风险。