

## 第八课 红外遥控小车



### 一． 本节要点

学习小车，当然要学会制作遥控小车啦，在一定的空间内能够无线遥控你的小车，是不是很有趣呢，本文教你如何制作遥控小车吧。

- ◆ 学习红外遥控器和红外接收管的使用
- ◆ 了解红外遥控器原理
- ◆ 编程完成红外遥控小车
- ◆ 准备器材：红外遥控器，Micro USB 线缆，miniQ 小车

### 二． 红外遥控原理介绍

#### 1、工作原理

在通用红外遥控系统由发射和接收两大部分组成，发射部分由红外遥控器组成，接收部分由红外接收管组成，红外遥控器发出的信号是一连串的二进制脉冲码 为了使其在无线传输过程中免受其他红外信号的干扰,通常都是先将其调制在特定的载波频率上,然后再经红外发射二极管发射出去,而红外线接收装置则要滤除其他杂波,只接收该特定频率的信号并将其还原成二进制脉冲码,也就是解调.

内置接收管将红外发射管发射出来的光信号转换为微弱的电信号，此信号经由 IC 内部放大器进行放大，然后通过自动增益控制、带通滤波、解调变、波形整形后还原为遥控器发射出的原始编码，经由接收头的信号输出脚输入到电器上的编码识别电路.

### 3.红外遥控协议

我们配套的红外遥控器的编码方式为：NEC 协议。下面就介绍一下 NEC 协议：

•NEC 协议介绍：

特点：（1）8 位地址位，8 位命令位

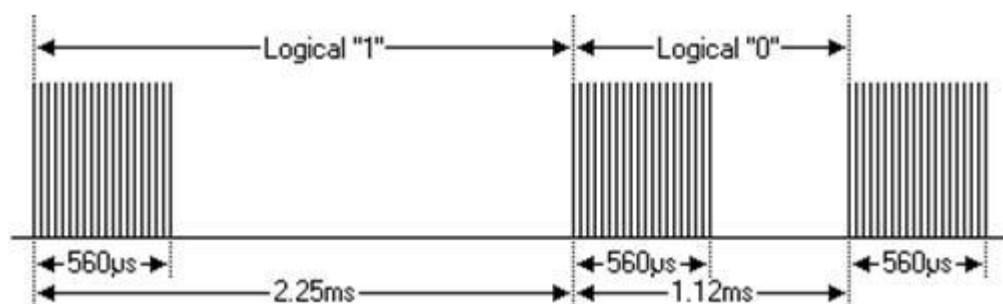
（2）为了可靠性地址位和命令位被传输两次

（3）脉冲位置调制

（4）载波频率 38khz

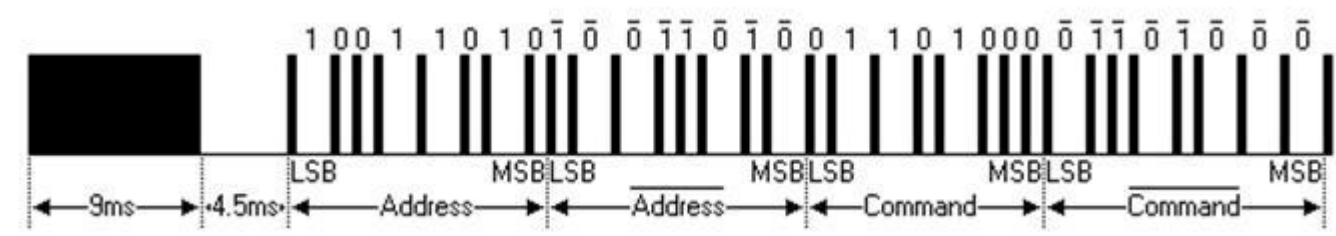
（5）每一位的时间为 1.125ms 或 2.25ms

•逻辑 0 和 1 的定义如下图：



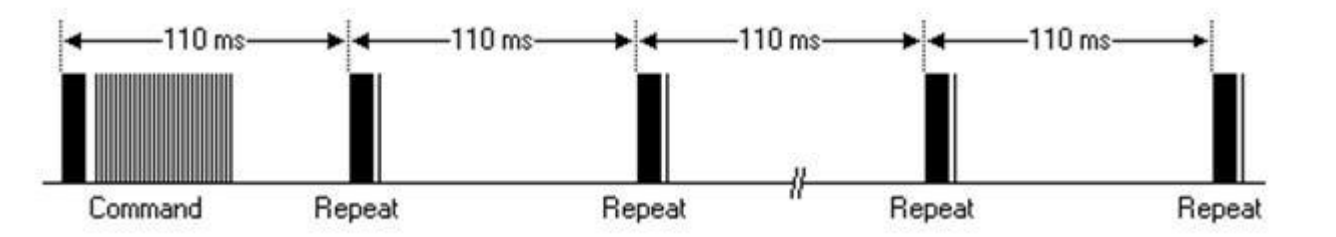
协议如下：

•按键按下立刻松开的发射脉冲：



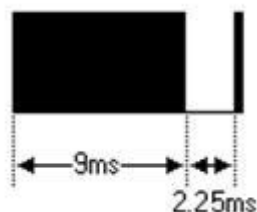
上面的图片显示了 NEC 的协议典型的脉冲序列。注意：这是首先发送 LSB（最低位）的协议。在上面的脉冲传输的地址为 0x59 命令为 0x16。一个消息是由一个 9ms 的高电平开始，随后有一个 4.5ms 的低电平，（这两段电平组成引导码）然后由地址码和命令码。地址和命令传输两次。第二次所有位都取反，可用于对所收到的消息中的确认使用。总传输时间是恒定的，因为每一点与它取反长度重复。如果你不感兴趣，你可以忽略这个可靠性取反，也可以扩大地址和命令，以每 16 位！

•按键按下一段时间才松开的发射脉冲：



一个命令发送一次，即使在遥控器上的按键仍然按下。当按键一直按下时，第一个 110ms 的脉冲与上图一样，之后每 110ms 重复代码传输一次。这个重复代码是由一个 9ms 的高电平脉冲和一个 2.25ms 低电平和 560μs 的高电平组成。

## •重复脉冲



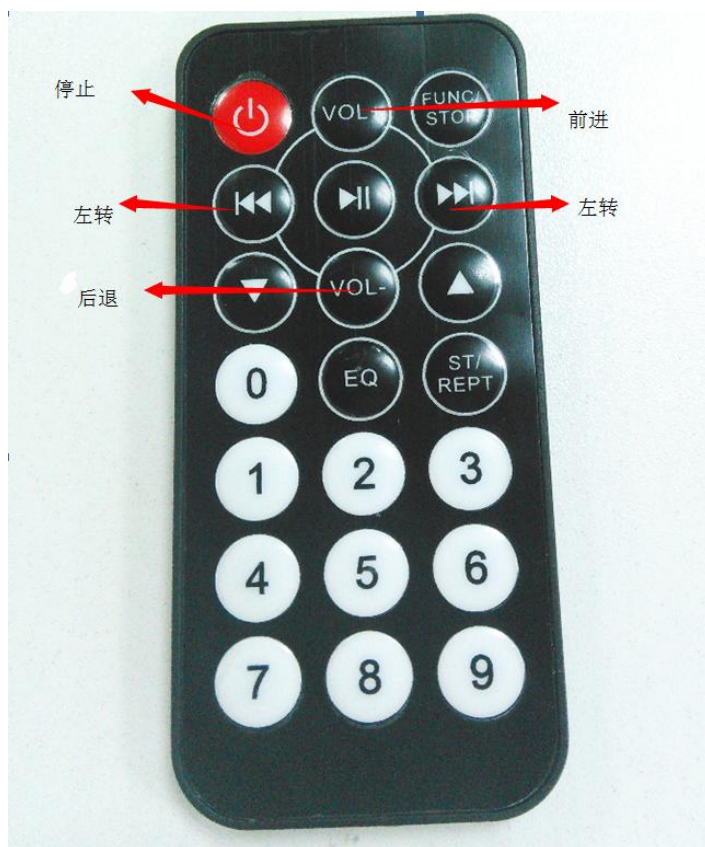
注意：脉冲波形进入一体化接收头以后，因为一体化接收头里要进行解码、信号放大和整形，故要注意在没有红外信号时，其输出端为高电平，有信号时为低电平，故其输出信号电平正好和发射端相反。接收端脉冲大家可以通过示波器看到，结合看到的波形理解程序。

## 2.小车遥控编程思想

根据 NEC 编码的特点和接收端的波形，本实验将接收端的波形分成四部分：引导码（9ms 和 4.5ms 的脉冲）、地址码 16 位（包括 8 位的地址位和 8 位的地址的取反）、命令码 16 位（包括 8 位命令位和 8 位命令位的取反）、重复码（9ms、2.25ms、560us 脉冲组成）。利用定时器对接收到的波形的高电平段和低电平段进行测量，根据测量到的时间来区分：逻辑“0”、逻辑“1”、引导脉冲、重复脉冲。引导码和地址码只要判断是正确的脉冲即可，不用存储，但是命令码必须存储，因为每个按键的命令码都不同，根据命令码来执行相应的动作。

其键值表如下表，以红色按钮为例（以下为 16 进制），00fd 为用户码（同一遥控器所有按键采用同一用户码），00 为命令码，ff 为反码。

在小车试验中，我们只需要控制前进、后退、左转、右转和停止，所以我们只需要使用 5 个按键即可，如图所示。



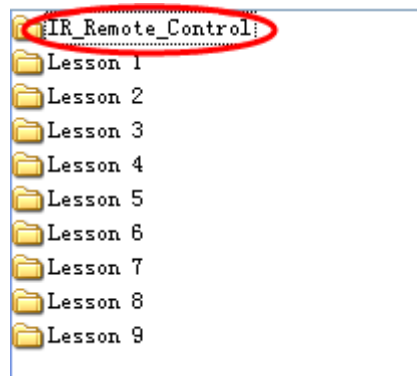
小车所需要的遥控器键值表

遥控器字符	键值
红色按钮	0x00fd00ff
VOL+	0x00fd807f
VOL-	0x00fd906f
左 2 个三角	0x00fd20df
右 2 个三角	0x00fd609f

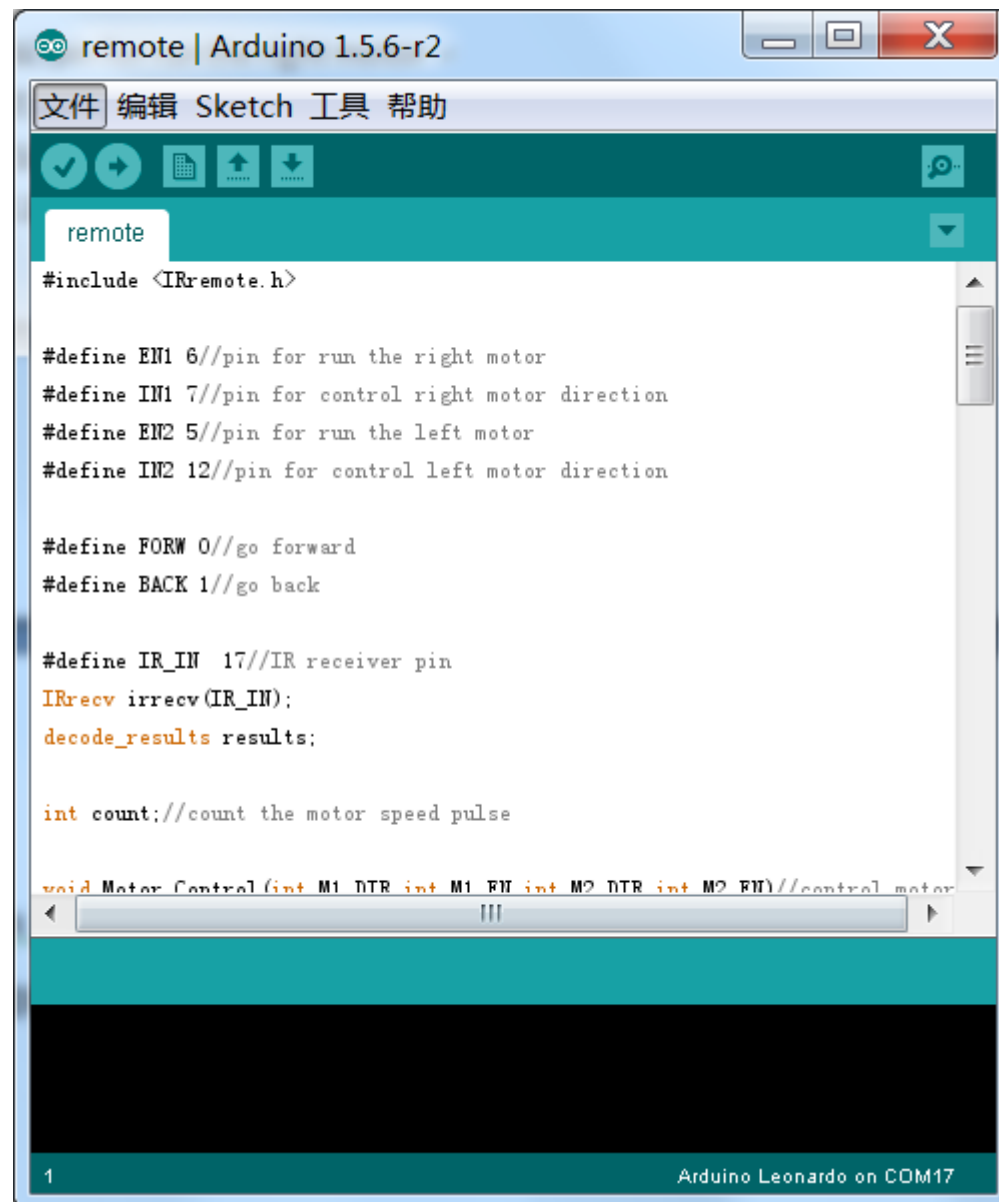
### 三． 程序下载

STEP1： 添加库文件，将 IR\_Remote\_Control 文件夹下的 IRremote 库文件夹复制到

Arduino 中的 library 文件夹中，打开 remote 中的 remote.ino 文件，



STEP2 : 点击下载，用遥控器控制你的小车，你会惊奇的发现，你的小车已经可以按照你的指挥行走啦！



## 四． 软件部分解析

### （1） 定义电机引脚

```
#define EN1 6//pin for run the right motor
#define IN1 7//pin for control right motor direction
#define EN2 5//pin for run the left motor
#define IN2 12//pin for control left motor direction
```

### （2） 定义传感器引脚

```
#define IR_IN 17//IR receiver pin
```

### （3） 根据接收遥控器传来的键值信号判断执行小车动作

```
void dump(decode_results *results)
{
    if(results->value==0x00fd00ff)
    {
        Motor_Control(FORW,0,FORW,0);//停止
    }
    if(results->value==0x00fd807f)
    {
        Motor_Control(FORW,100,FORW,100);//前进
    }
    if(results->value==0x00fd906f)
    {
        Motor_Control(BACK,100,BACK,100);//后退
    }
    if(results->value==0x00fd20df)
    {
        Motor_Control(FORW,100,BACK,100);//左转
    }
    if(results->value==0x00fd609f)
    {
        Motor_Control(BACK,100,FORW,100);//右转
    }
}
```