

红外遥控在单片机系统中应用原理与分析

红外线遥控是目前使用非常广泛的一种通信和遥控手段。由于红外线遥控装置具有体积小、功耗低、功能强、成本低等特点，因而，继彩电、录像机之后，在录音机、音响设备、空调机以及玩具等其它小型电器装置上也纷纷采用红外线遥控。工业设备中，在高压、辐射、有毒气体、粉尘等环境下，采用红外线遥控不仅完全可靠而且能有效地隔离电气干扰。下面，我们对红外遥控的基本原理与在单片机中的应用做一简要分析。

1 红外遥控系统

通用红外遥控系统由发射和接收两大部分组成。应用编/解码专用集成电路芯片来进行控制操作，如图1所示。发射部分包括键盘矩阵、编码调制、LED红外发送器；接收部分包括光、电转换放大器、解调、解码电路。

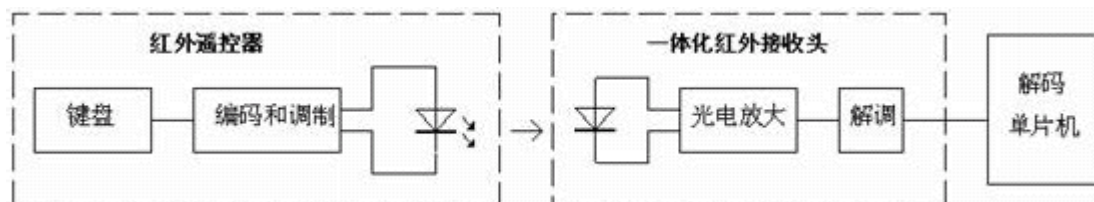


图1 红外线遥控系统框图

2 遥控发射器及其编码

遥控发射器专用芯片很多，根据编码格式可以分成两大类，这里我们以运用比较广泛，解码比较容易的一类来加以说明，现以日本NEC的uPD6121G组成发射电路为例说明编码原理（一般家庭用的DVD、VCD、音响都使用这种编码方式）。当发射器按键按下后，即有遥控码发出，所按的键不同遥控编码也不同。这种遥控码具有以下特征：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”，其波形如图2所示。

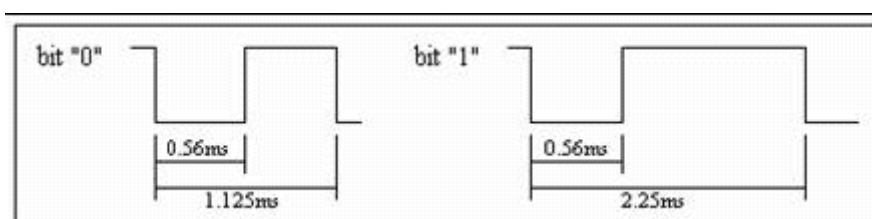


图2 波形图

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，如图3所示

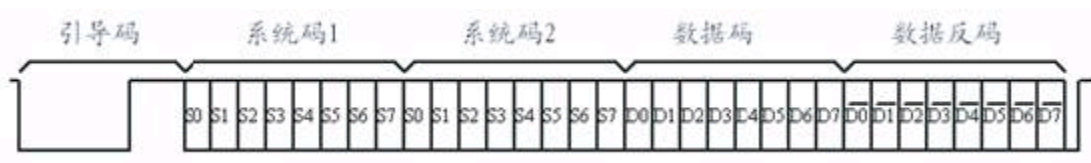


图3 遥控信号编码波形图

UPD6121G产生的遥控编码是连续的32位二进制码组，其中前16位为用户识别码，能区别不同的电器设备，防止不同机种遥控码互相干扰。该芯片的用户识别码固定为十六进制01H；后16位为8位操作码（功

能码)及其反码。UPD6121G最多可以有128种不同组合的编码。遥控器在按键按下后,周期性地发出同一种32位二进制码,周期约为108ms。一组码本身的持续时间随它包含的二进制“0”和“1”的个数不同而不同,大约在45~63ms之间,图4为发射波形图。

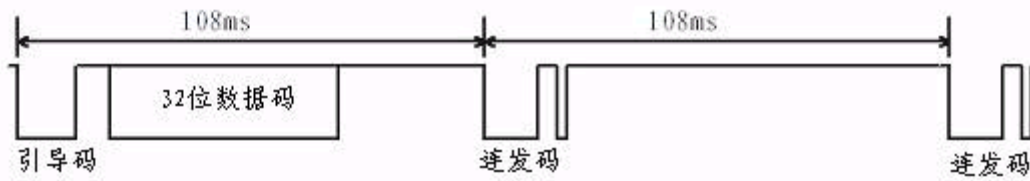


图4 遥控连发信号波形

当一个键按下超过36ms,振荡器使芯片激活,将发射一组108ms的编码脉冲,这108ms发射代码由一个引导码(9ms),一个结果码(4.5ms),低8位地址码(9ms~18ms),高8位地址码(9ms~18ms),8位数据码(9ms~18ms)和这8位数据的反码(9ms~18ms)组成。如果键按下超过108ms仍未松开,接下来发射的代码(连发码)将仅由起始码(9ms)和结束码(2.25ms)组成。



图5 引导码

图6 连发码

3 遥控信号接收

接收电路可以使用一种集红外线接收和放大于一体的一体化红外线接收器,不需要任何外接元件,就能完成从红外线接收到输出与TTL电平信号兼容的所有工作,而体积和普通的塑封三极管大小一样,它适合于各种红外线遥控和红外线数据传输。接收器对外只有3个引脚:Out、GND、Vcc与单片机接口非常方便,如图7所示。

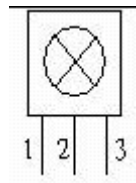


图7 一体化红外接收头

- ① 脉冲信号输出接,直接接单片机的I/O口;
- ② GND接系统的地线(0V);
- ③ Vcc接系统的电源正极(+5V);

4 遥控信号的解码

下面是一个对51实验板配套的红外线遥控器的解码程序,它可以把红外遥控器每一个按键的键值读出来,并且通过实验板上P1口的8个LED显示出来,在解码成功的同时并且能发出“嘀嘀”的提示音。

```

;=====
; 红外遥控接收
;=====
ORG 0000H
MAIN:
JNB P2.2, IR ;遥控扫描
LJMP MAIN ;在正常无遥控信号时,一体化红外接收头输出是高电平,程序一直在循环。
;=====
; 解码程序
    
```

IR:

;以下对遥控信号的9000微秒的初始低电平信号的识别，波形见图5。

MOV R6, #10

IR_SB:

ACALL DELAY882 ;调用882微秒延时子程序

JB P2.2, IR_ERROR ;延时882微秒后判断P2.2脚是否出现高电平如果有就退出解码程序

DJNZ R6, IR_SB ;重复10次，目的是检测在8820微秒内如果出现高电平就退出解码程序

;识别连发码，和跳过4.5ma的高电平。

JNB P2.2, \$;等待高电平避开9毫秒低电平引导脉冲

ACALL DELAY2400

JNB P2.2, IR_Rp ;这里为低电平，认为是连发码信号，见图6。

ACALL DELAY2400 ;延时4.74毫秒避开4.5毫秒的结果码

;以下32数据码的读取，0和1的识别 请看图2

MOV R1, #1AH ;设定1AH为起始RAM区

MOV R2, #4

IR_4BYTE:

MOV R3, #8

IR_8BIT:

JNB P2.2, \$;等待地址码第一位的高电平信号

LCALL DELAY882 ;高电平开始后用882微秒的时间尺去判断信号此时的高低电平状态

MOV C, P2.2 ;将P2.2引脚此时的电平状态0或1存入C中

JNC IR_8BIT_0 ;如果为0就跳转到IR_8BIT_0

LCALL DELAY1000

IR_8BIT_0:

MOV A, @R1 ;将R1中地址的给A

RRC A ;将C中的值0或1移入A中的最低位

MOV @R1, A ;将A中的数暂时存放在R1中

DJNZ R3, IR_8BIT ;接收地址码的高8位

INC R1 ;对R1中的值加1，换下一个RAM

DJNZ R2, IR_4BYTE ;接收完16位地址码和8位数据码和8位数据，

;存放在1AH/1BH/1CH/1DH的RAM中

;解码成功

JMP IR_GOTO

IR_Rp:

;这里为重复码执行处

;按住遥控按键时，每过108ms就到这里来

JMP IR_GOTO

IR_ERROR:

;错语退出

LJMP MAIN ;退出解码子程序

;=====

;遥控执行部份

IR_GOTO:

;这里还要判断1AH和1BH 两个系统码或用户码，用于识别不同的遥控器

```
;MOV A, 1AH
;CJNE A, #xxH, IR_ERROR ;用户码 1 不对则退出
;MOV A, 1BH
;CJNE A, #xxH, IR_ERROR ;用户码 2 不对则退出
;判断两个数据码是否相反
MOV A, 1CH
CPL A
CJNE A, 1DH, IR_ERROR ;两个数据码不相反则退出
;遥控执行部份
;MOV A, 1DH ;判断对应按键
;CJNE A, #xxH, $+6
;LJMP -a跳到对应按键执行处
;CJNE A, #xxH, $+6
;LJMP -a跳到对应按键执行处
;.
MOV P1, 1DH ;将按键的键值通过P1口的8个LED显示出来!
CLR P2. 3 ;蜂鸣器鸣响一嘀嘀嘀-的声音, 表示解码成功
LCALL DELAY2400
LCALL DELAY2400
LCALL DELAY2400
SETB P2. 3 ;蜂鸣器停止
;清除遥控值使连接失效
MOV 1AH, #00H
MOV 1BH, #00H
MOV 1CH, #00H
MOV 1DH, #00H
LJMP MAIN
;=====
; 延时子程序
;=====882
DELAY882: ;1.085x ((202x4)+5)=882
MOV R7, #202
DELAY882_A:
NOP
NOP
DJNZ R7, DELAY882_A
RET
;=====1000
DELAY1000: ;1.085x ((229x4)+5)=999.285
MOV R7, #229
DELAY1000_A:
NOP
NOP
DJNZ R7, DELAY1000_A
```

```
RET
;=====2400
DELAY2400: ;1.085x ((245x9)+5)=2397.85
MOV R7, #245
DELAY2400_A:
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
DJNZ R7, DELAY2400_A
RET
END
```