

## HT48RA0A/HT48RA0-1/HT48RA0-2 PL 型灯管驱动器

文件编码：HA0091s

### 介绍：

PL 型灯管是通过变换器来驱动的。该变换器一般提供较大范围的直流输入电压，并将其转换为高频高压的交流输出来驱动灯管发光。本应用范例中电路使用了低电位的直流电压输入，来驱动 13W 的 PL 型灯管发光。

在本应用范例的说明中，HT48RA0A/HT48RA0-1/HT48RA0-2 的 MCU 被应用作为控制 PL 型灯管的主要核心部件。其基本的运作方式将连同相关电路的控制方法一起进行论述。

### PL 型灯管的工作原理：

PL 型灯管是在玻璃管中填充惰性气体后密封制成的。当灯管连接上高电压源的时候，管内的惰性气体电离产生紫外光，而后激发内部覆盖的磷，发出可见光。

### 范例电路说明：

图 1 是一个最简单的 PL 型灯管驱动器的应用电路。它由一个微控制单元，连接直流电源的两个高速二极管，四个电平转换的 NPN 三极管，一个连接 PL 型灯管灯丝的电容器，和两个连接二极管的电感线圈组成。提供了两个升压电路，一个由 L1 和 Q5 构成，另一个由 L2 和 Q6 构成。图 2 是电路的 PCB 布局。DC-AC 的转换器用于驱动 13 瓦的真空管，4-pin 的 PL 型灯管。

电路的电压来源于一个稳定的 12V 直流电池或者在线的电源，它提供出一个交流电源驱动灯管。这种直接的驱动方式在转换器上提供了明显的有利条件。

HT48RA0A/HT48RA0-1/HT48RA0-2 微处理单元是存在于范例电路核心部分的 8 位高性能 MCU。在电路的 2 个输出引脚上产生一个交变频率信号。这两个输出引脚通过电平转换电路控制升压电路获得一个交变的高频高压信号源来驱动灯管。

范例电路是一个使用三种不同频率信号驱动 PL 型灯管的 DC 转 AC 的转换器。转换器提供了一个瞬间的击打（启动）频率加热灯管的灯丝，持续约几百毫秒。频率在为 250KHz 左右，而后频率变为 9KHz，持续约 0.1 秒，然后频率升至稳定的 22KHz 以使灯管被点亮。

转换器包含一个输入控制端在 MCU 的 PB5 引脚。当该引脚电平为低时，驱动器功能有效，PL 型灯管将被点亮。若该引脚电平为高时，驱动器功能被禁止，PL 型灯管将被熄灭。这种输入信号的控制功能能够被自动照明设备的应用案例所采用。

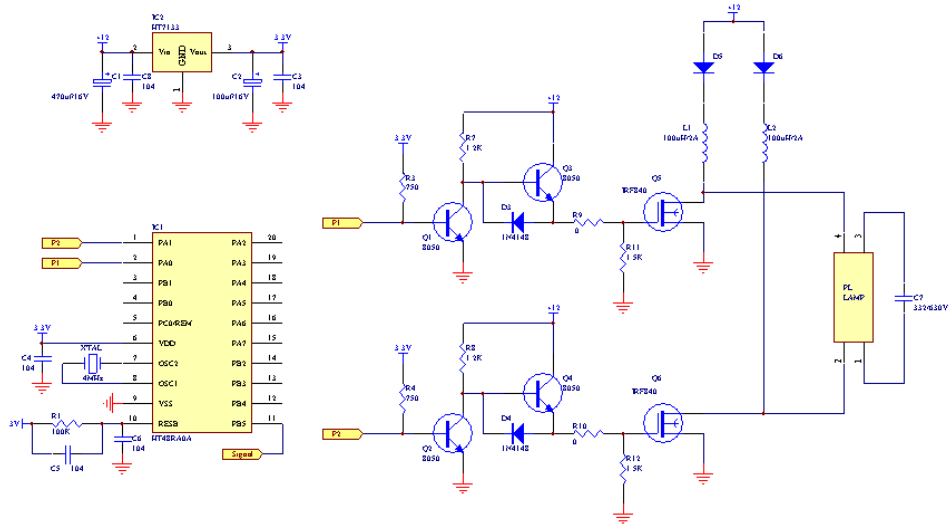


图 1

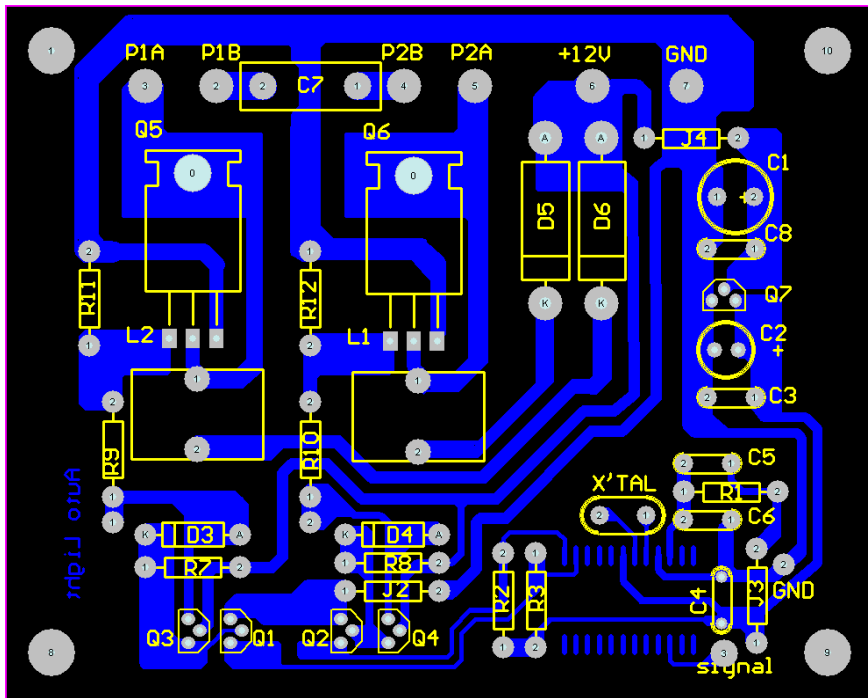


图 2

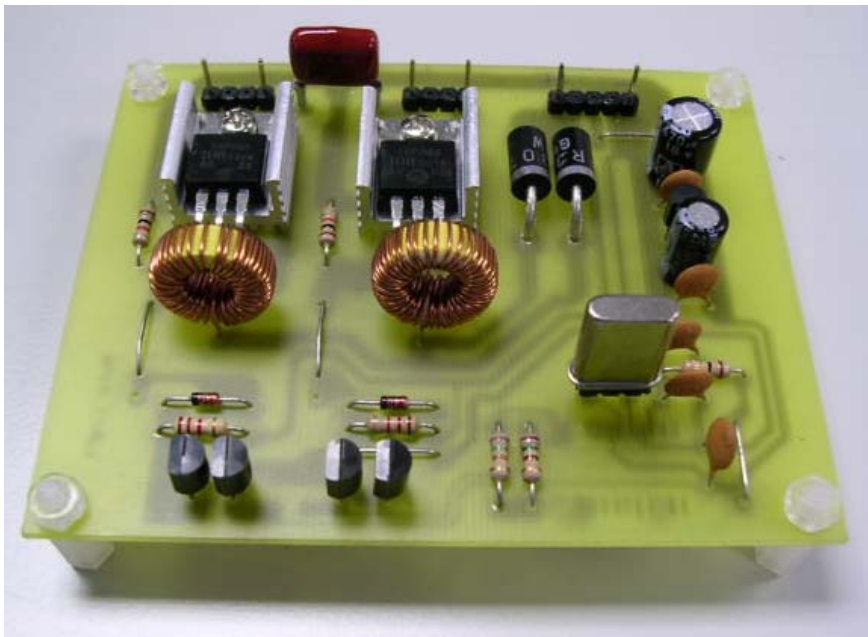


图 3



# HT48RA0A/HT48RA0-1/HT48RA0-2 PL 型灯管驱动器

## 范例电路程序

```
*****
:
:
:   HT48RA0A 自动照明样板
:
:
:*****
INCLUDE HT48RA0A.INC

DATA          .SECTION      AT 20H   'DATA'
;-----
counter0      db           ?
counter1      db           ?
keycount      db           ?
keystatus     db           ?
;-----
CODE          .SECTION
              ORG          00H
              jmp          initial

;-----
              ORG          010H
Initial:
              mov         A,20H
              mov         MP,A
              jmp         clramloop

clramloop:
              clr         R0
              inc         MP

              mov         A,0BFH
              sub         A,MP
              sz          STATUS.0
              jmp         clramloop      ;MP0<3FH 跳转至 clramloop

;-----
```

```

;-----
;
;           -- 主程序 --
;-----
initial_mode:
    clr      keycount

    mov     A,00000011B    ;设置 PA0 和 PA1 输出
                        ;"1"
    mov     PA,A

    mov     A,00000000B    ;
    mov     PC,A

;-----

    mov     A,0E0H;
    mov     counter1,A    ;100×256×32=第一次点火时间
                        ;(4MHz=E0H=0.8 秒)

;.....
;.....
sign_test:
    sz      keystatus.0    ;检测 PB5 引脚电平
    jmp     ignite        ;如果 PB5 电平为低，点亮
                        ;灯管

    calldelay_46
    calldelay_46

    sz      PB5
    jmp     clr_sign
    inc     keycount
    mov     A,0FAH
    sub     A,keycount
    sz      status.0
    jmp     sign_test
    set     keystatus.0    ;
    jmp     sign_test

clr_sign:
    clr     keycount
    mov     A,00000011B    ;设置 PA0 和 PA1 输出
                        ;"1"
    jmp     sign_test

```

```

;.....
ignite:
        clr    keycount
        call   ignite_sub    ;点火模式: 第一次时间
                                ;子程序
;
        mov    A,0FCH        ;4MHz=FCH=0.1 秒
        mov    counter1,A    ;110×256×4=第二次点火
                                ;时间

ignite_se:
                                ;点火模式: 第一次时间
        mov    A,00000001b
        clr    pa
        mov    PA,A
        call   delay_46
        nop
        nop
        nop
        nop
;...
        mov    A,00000010b
        clr    pa
        mov    PA,A
        call   delay_46
        nop
        siz    counter0
        jmp    ignite_se
        siz    counter1
        jmp    ignite_se

;.....
run_mode:
                                ;运作模式: 稳定频率在
                                ;22kHz
        mov    A,00000001b
        clr    pa
        mov    PA,A
        call   delay_run
        snz    keystatus.0    ;
        jmp    initial_mode    ;PB5 为高电平, 熄灭
                                ;灯管
;.....
        mov    A,00000010b
        clr    pa
        mov    PA,A
        call   delay_run
        jmp    run_mode

;-----
;    -- 子程序 --
;-----
;delay_run:
        clr    wdt
        jmp    $+1
        jmp    $+1
        jmp    $+1
        jmp    $+1
        jmp    $+1
        jmp    $+1

```

```

ret
;
delay_run:
    clr    wdt

    snz    PB5    ;检测 PB5 引脚电平
    jmp    delay_run1 ;如果 PB5 电平为高，熄灭
                    ;灯管

    inc    keycount
    mov    A,0FAH
    sub    A,keycount
    sz
    jmp    delay_run2
    clr    keystatus.0
    jmp    delay_run3

delay_run1:
    nop
    jmp    $+1
    jmp    $+1

delay_run2:
    jmp    $+1

delay_run3:
    jmp    $+1

ret

```



# HT48RA0A/HT48RA0-1/HT48RA0-2 PL 型灯管驱动器

delay\_46:

```
clr      wdt  
  
jmp     $+1  
jmp     $+1  
jmp     $+1  
jmp     $+1  
jmp     $+1  
  
jmp     $+1  
jmp     $+1  
jmp     $+1  
jmp     $+1  
jmp     $+1  
  
jmp     $+1  
jmp     $+1  
jmp     $+1  
jmp     $+1  
jmp     $+1  
  
jmp     $+1  
nop  
ret
```

```
-----  
:      -- 子程序 --  
:-----  
ignite_sub:  
  
      mov     A,00000001b  
      mov     PA,A  
;  
      mov     A,00000010b  
      mov     PA,A  
;..  
      mov     A,00000001b  
      mov     PA,A  
;;  
      mov     A,00000010b  
      mov     PA,A  
;..  
      mov     A,00000001b  
      mov     PA,A  
;  
      mov     A,00000010b  
      mov     PA,A  
;..  
      mov     A,00000001b  
      mov     PA,A  
;;  
      mov     A,00000010b  
      mov     PA,A  
;..  
      mov     A,00000001b  
      mov     PA,A  
;.....  
      mov     A,00000001b  
      mov     PA,A  
;;  
      mov     A,00000010b  
      mov     PA,A  
;..  
      mov     A,00000001b  
      mov     PA,A  
;;  
      mov     A,00000010b  
      mov     PA,A
```



```

:.....
mov    A,00000001b
mov    PA,A
;;
mov    A,00000010b
mov    PA,A
:..
mov    A,00000001b
mov    PA,A
;;
mov    A,00000010b
mov    PA,A
:..
mov    A,00000001b
mov    PA,A
;;
mov    A,00000010b
mov    PA,A
:..
mov    A,00000001b
mov    PA,A
;;
mov    A,00000010b
mov    PA,A
:..
mov    A,00000001b
mov    PA,A
;;
mov    A,00000010b
mov    PA,A
:.....
mov    A,00000001b
mov    PA,A
;;
mov    A,00000010b
mov    PA,A
:..
mov    A,00000001b
mov    PA,A
;;
mov    A,00000010b
mov    PA,A
:..
mov    A,00000001b
mov    PA,A
;;
mov    A,00000010b
mov    PA,A

```

```

;...
mov    A,00000001b
mov    PA,A
;;
mov    A,00000010b
mov    PA,A
;...
mov    A,00000001b
mov    PA,A
;;
mov    A,00000010b
mov    PA,A
clr    wdt

siz    counter0
jmp    ignite_sub

siz    counter1
jmp    ignite_sub

clr    pa
;.....
ret
end
```