

LED 显示屏动态显示和远程监控的实现

福星电子网

提供单片机学习板，开发板，最小系统板；超声波测距实验应用板，各类器

件仪表，详情请访问网站 <http://www.fxdzw.com>

摘要: 由于普通 LED 点阵显示屏动态显示通常采用硬件扫描驱动, 这在一些需要特殊显示的场合显得不够灵活。文中提出了一种利用 PC 机和单片机的通讯来实现显示屏灵活的动态显示和远程监控的设计方法, 同时该方法还可以将显示内容在 PC 机上进行预览。

关键词: LED; 动态显示; 远程控制; 显示预览

1 引言

LED 点阵电子显示屏是集微电子技术、计算机技术、信息处理技术于一体的大型显示屏系统。它以其色彩鲜艳, 动态范围广, 亮度高, 寿命长, 工作稳定可靠等优点而成为众多显示媒体以及户外作业显示的理想选择。同时也可广泛应用到军事、车站、宾馆、体育、新闻、金融、证券、广告以及交通运输等许多行业。

目前大多数的 LED 点阵显示系统自带字库。其显示和动态效果(主要是显示内容的滚动)的实现主要依靠硬件扫描驱动, 该方法虽然比较方便, 但显示只能按照预先的设计进行。而实际上经常会遇到一些特殊要求的动态显示, 比如电梯运行中指示箭头的上下移动、某些智能仪表幅值的条形显示、广告中厂家的商标显示等。这时一般的显示系统就很难达到要求。另外, 由于受到存储器本身的局限, 其特殊字符或图案也往往难以显示, 同时显示内容也不能随意更改。本文提出一种利用 PC 机和单片机控制的 LED 显示系统通讯方法。该方法可以对显示内容(包括汉字和特殊图符)进行实时控制, 从而实现诸如闪动、滚动、打字等多种动态显示效果。该方法同时还可以调节动态显示的速度, 同时用户也可以在 PC 机上进行显示效果的预览, 显示内容亦可以即时修改。另外, 通过标准的 RS232 / 485 转换模块还可以实现对显示系统的远程控制。

2 系统硬件设计

本系统主要的硬件设计是下位机单片机的显示控制部分。而上位机(PC机)与单片机显示控制部分的接口为标准 RS232 通讯方式。若需实现远程监控, 只需增加 RS232 / 485 转换模块即可, 该部分已有成熟的电路设计, 故不再详细叙述。

具体的 LED 显示屏控制电路如图 1 所示。整个电路由单片机 89C52、点阵数据存储器 6264、列驱动电路 ULN2803、行驱动电路 TIP122、移位寄存器 4094 及附属电路组成。该电路所设计的电子屏可显示 10 个汉字, 需要 40 个 8×8 LED 点阵模块, 可组成 16×160 的矩形点阵。由于 AT89C52 仅有 8k 存储空间, 而显示的内容由 PC 机控制, 因此不可能预先把需要显示的内容做成点阵存在单片机中, 而只能由 PC 机即时地把所需显示的点阵数据传给单片机并存入缓冲区 6264。

该电路的显示采用逐行扫描方式。工作时, 由单片机从缓冲区取出第一行需要显示的 20 字节点阵数据, 再由列点阵数据输入端 P1. 2 口按位依次串行输入至列移位寄存器, 其数据输入的顺序与显示内容的顺序相反。然后置行点阵选通端 P1. 3 为 1, 即置行移位寄存器的 D 为高电平, STR 使能(所有 4094 的 0E 引脚接 +5V 电平), 从而使列移位寄存器中的数据同时并行输出以选通该行。经延时一段时间后再进行下一行点阵数据的显示。需要注意的是, 每次只能选通一行数据, 即要通过不断的逐行扫描来实现汉字或字符的显示。

3 显示与控制的设计

在笔者设计的 PC 机控制多单片机显示系统中, 用 PC 机实现的主要功能包括单片机显示子系统的选择, 显示方式选择(包括静态、闪动、滚动、打字等), 滚动方向选择(包括上下滚动和左右滚动), 动态显示速度调节(即文字闪动频率、滚动速度、打字显示速度等), 显示内容输入及显示预览等。单片机一般通过 RS232 / 485 串行接收 PC 机发出的显示指令采用定时器中断方式进行行扫描, 每次中断显示一行, 定时中断时间为 1. 25ms, 这样整屏的刷新率为 50Hz, 因而无闪烁感。

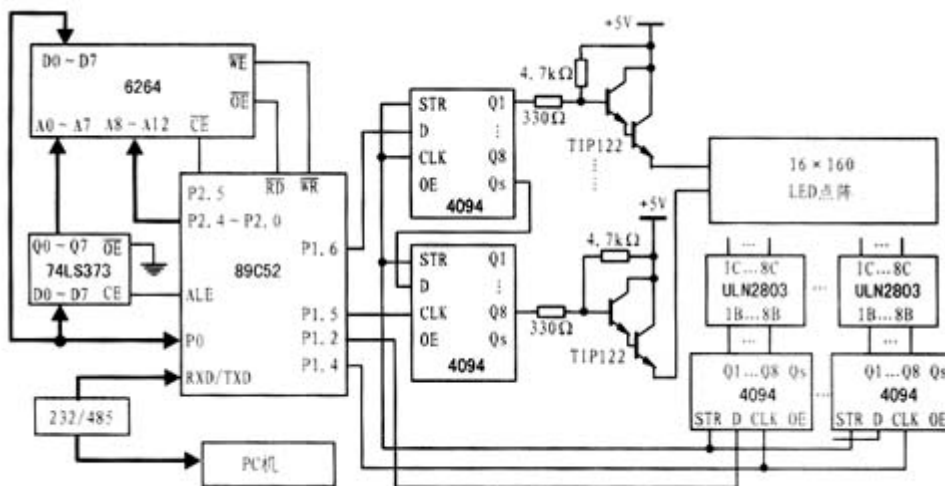


图1 LED显示屏控制电路

实现动态显示速度调节的方法通常是改变定时器的中断时间，但是当显示速度很慢的时候，该方法容易使整屏的刷新率降低，从而使显示内容出现闪烁。因此，本设计采用一种“软定时”方法，即在程序中命名一变量作为“软定时器”，以用来设定两次动态显示的时间间隔。在对定时中断调用计数时，如果调用次数达到设定值，则改变显示内容。为保证能够正常显示，“软定时器”的设定值必须大于整屏显示周期。由于显示屏每行显示1.25ms，整屏显示周期为20ms，考虑到余量的情况，可将软定时器的设定值定在大于30ms。如此循环计数，即可实现动态显示。“软定时器”的设定值可以通过上位机PC机来改变，这样既可实现LED动态显示的速度调节，又可保持显示内容的流畅和无闪烁感。

3.1 单片机动态显示控制

以上提到的静态、闪动、滚动和打字等4种显示方式，实际上是单片机定时中断程序进行行扫描处理的不同方法。下面将分别说明如何实现这4种显示方式。

静态显示只需在定时中断处理程序中从显示缓冲区调入相应的一行显示数据，然后选中该行即可实现该行的显示，如此循环，便可显示整个内容。闪动显示与此类似，不同的是要间隔一个“软定时器”的定时时间，在行扫描时，行移位寄存器的D端打入的全为0，可使得整屏不显示，以确保黑屏时间与显示时间相等，从而实现汉字或图符的闪动显示。

滚动显示要求需要显示的内容每隔一定时间向指定方向（这里以从右向左为例）移动一列，这样显示屏可以显示更多的内容。为此，需要在下次移动显示之前对显示缓冲区的内容进行更改，从而完成相应点阵数据的移位操作。具体操作方法是：

设置一个显示缓冲区（如图2所示），该区应包括两部分：一部分用来保存当前LED显示屏上显示的10个汉字点阵数据；另一部分为点阵数据预装载区，用来保存即将进入LED显示屏的1个汉字的点阵数据。滚动指针始终指向显示屏的最右边原点。当滚动指针移动到需要显示的点阵数据存储区的第1个汉字的首地址时，显示缓冲区LED显示区为空白，而预装载区已保存了第1个待显示汉字的点阵数据。当需要滚动显示时，则可在接下来的扫描周期的每个行扫描中断处理程序中，将对显示缓冲区的相应行点阵数据左移一位，同时更改显示缓冲区的内容。（需要注意的是，要确保该操作能在1.25ms的中断时间内完成。这里89C52采用22MHz晶振，实验证明可以实现该操作）。这样，在一个扫描周期后，整个汉字将左移一列，而显示缓冲区的内容也同时更改。由于预装载区保存了1个汉字点阵数据，即16×16点阵，所以当前显示缓冲区的内容只能移动16列。当下一个滚动到来时，滚动指针将移动到点阵数据存储区的下一个汉字的首地址，并在预装载区存入该汉字的点阵数据。然后重复执行上述操作便可实现滚动显示。特殊字符或图形的显示与此类似，这里不再赘述。

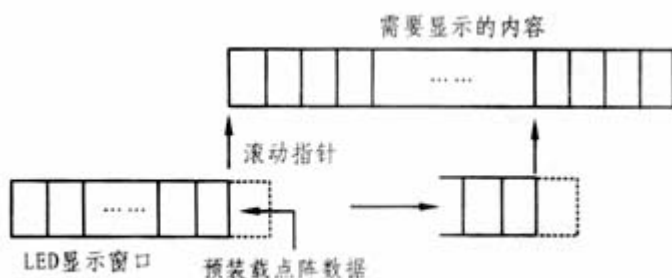


图 2 LED 滚动显示处理示意图

打字显示要求汉字在显示屏上按从左到右的顺序一个个的出现，如同打字的效果。设计时可采用如下方法：首先将 LED 显示屏对应的显示缓冲区全部清零，即 LED 显示空白，然后每间隔一个“软定时器”设定的动态显示时间，显示缓冲区依次加入一个汉字点阵数据并进行扫描显示，这样就可达到打字显示的效果。

3. 2 PC 机控制程序

a. 通讯功能的实现

在 Windows 环境下，实现 PC 与单片机的通讯可利用 Windows 的通讯 API 函数或者利用 VC++（或其它语言）的标准通讯函数 `__inp`、`__outp` 来实现。但上述两种方法比较繁琐，而采用 ActiveX 控件 MSComm32 来实现则非常方便。该控件用事件的方式简化了对串口操作的编程，并可设置串行通信的数据发送和接收，还可对串口状态及串口通信的信息格式和协议进行设置。其初始化程序如下：

```

BOOL CLEDDlg::OnInitDialog()
{
    ...
    m_Mscomm1.SetCommPort(2); //设置串口 2
    if(!m_Mscomm1.GetCommPort())
        m_Mscomm1.SetPortOpen(TRUE);
    m_Mscomm1.SetInputMode(com Input
        Mode Binary); //二进制方式
    m_Mscomm1.Set Settings ("9600,n,8,1")
        ; //设置波特率
    m_Mscomm1.SetRThreshold(1); //事件方式
    m_Mscomm1.SetInBufferCount(0);

    m_Mscomm1.SetOutBufferCount(0);
    ...
}
    
```

一般情况下，PC 要与多个单片机 89C51 系统进行主从式通讯，为了区分各单片机系统，可以使 89C51 采用串口工作方式 3，即 11 位异步接收 / 发送方式，该方式的有效数据为 9 位，其中第 9 位为地址 / 数据信息的标志位，其作用是使从机据此判断发送的数据是否为地址，从而实现多机操作。但现在由于采用的是 MSCOMM 控件来实现 PC 机和单片机之间的通讯，这是一种标准的 10 位串口通信方式，即 8 位标准数据位和该数据的起始位、停止位各 1 位。因此二者格式不相符，故很难利用上述方案。因此可考虑将单片机串口设为工作方式 1，即改为 10 位异步接收 / 发送方式来解决，其通讯流程如下：

首先发通信开始标志，接着发送需要操作的单片机系统地址，然后发送显示工作命令字，该命令包括 2 个字节，前一字节用于设定显示方式和滚动方向，后一字节则用于设定显示速度。再往下是传送显示内容的点阵数据，最后对数据进行校验。该通讯规约非常简便，能够较好的解决上述问题，从而实现 PC 机与多单片机之间的主从式通讯及对显示的控制。

需要注意的是，当显示内容需要改变时，为了避免在单片机串行中断接收数据时，显示屏出现乱码，应使显示屏暂不显示（处于“黑屏”状态），直到数据接收完全，串行中断处理结束时再显示。

汉字字模的提取非常关键，本文的字模数据取自 UC DOS 下的字库文件 HZK16。关于这方面的介绍较多，文献^[2]给出了较为具体的在 VC 下提取汉字字模的方案，这里不再赘述。对于特殊字符或图形点阵数据的提取，简便的方法可以先做一个 BMP 文件，然后用一些取模软件（如字模提取 v2.1）来获得。为了显示方便，点阵数据的格式应为 $n \times (16 \times 8)$ ，不足要求的则应以 0 数据补充。

b. 动态效果模拟显示
为了方便调节 LED 的显示效果，笔者在 PC 机的控制界面上设计了 LED 显示屏的模拟显示，它同实际的显示效果完全一样。用户可以设定显示的模式，并调节显示速度，然后在界面上对显示效果进行预览，同时还可以随时修改和设定参数，因而十分方便简捷。

为此，可先在界面上描绘出虚拟的 LED 显示屏，由于实际的显示屏为 160×16 点阵，故须在界面上设定相同的区域。

实现动态显示效果的方法和以上几种基本类似，这里以滚动显示为例作一说明。对于需要滚动的文字，可以将其设置为位图格式，暂存于内存中，然后利用 VC 提供的位图拷贝函数 `BitBlt` 将位图复制到显示位置。对于特殊字符或图形，则可以直接利用 `BitBlt` 函数调用到显示位置。然后在类 `CLEDDig` 的 `OnTimer` 函数中调用该函数，以实现文字的滚动显示。另外，也可以通过设定不同的响应时间间隔来改变文字的滚动速度。

汉字显示屏广泛应用与汽车报站器，广告屏等。本文介绍一种实用的汉字显示屏的制作，考虑到电路元件的易购性，没有使用 8×8 的点阵发光管模块，而是直接使用了 256 个高亮度发光管，组成了 16 行 16 列的发光点阵。同时为了降低制作难度，仅作了一个字的轮流显示，实际使用时可根据这个原理自行扩充显示的字数。

1 汉字显示的原理：

我们以 UC DOS 中文宋体字库为例，每一个字由 16 行 16 列的点阵组成显示。即国标汉字库中的每一个字均由 256 点阵来表示。我们可以把每一个点理解为一个像素，而把每一个字的字形理解为一幅图像。事实上这个汉字屏不仅可以显示汉字，也可以显示在 256 像素范围内的任何图形。

用 8 位的 AT89C51 单片机控制，由于单片机的总线为 8 位，一个字需要拆分为 2 个部分。

软件打开后输入汉字，点“检取”，十六进制数据的汉字代码即可自动生成，把我们所需要的竖排数据复制到我们的程序中即可。

我们把行列总线接在单片机的 i0 口，然后把上面分析到的扫描代码送入总线，就可以得到显示的汉字了。在这个例子里，由于一共用到 16 行，16 列，如果将其全部接入 89c51 单片机，一共使用 32 条 i0 口，这样造成了 i0 资源的耗尽，系统也再无扩充的余地。实际应用中我们使用 4-16 线译码器 74ls154 来完成列方向的显示。而行方向 16 条线则接在 p0 口和 p2 口。

程序清单：

```
ORG 00H
LOOP: MOV A, #0FFH ; 开机初始化，清除画面
MOV P0, A ; 清除 P0 口
      ANL P2, #00 ; 清除 P2 口
MOV R2, #200
D100MS: MOV R3, #250 ; 延时 100 毫秒
        DJNZ R3, $
        DJNZ R2, D100MS
        MOV 20H, #00H ; 取码指针的初值
```

```

I100:   MOV R1, #100 ; 每个字的停留时间
L16:   MOV R6, #16 ; 每个字 16 个码
        MOV R4, #00H ; 扫描指针清零
        MOV R0, 20H ; 取码指针存入 R0
L3:   MOV A, R4 ; 扫描指针存入 A
        MOV P1, A ; 扫描输出
        INC R4 ; 扫描指针加 1, 扫描下一个
        MOV A, R0 ; 取码指针存入 A
        MOV DPTR, #TABLE ; 取数据表的上半部分的代码
        MOVC A, @A+DPTR
        MOV P0, A ; 输出到 P0
        INC R0 ; 取码指针加 1, 取下一个码。
        MOV A, R0
        MOV DPTR, #TABLE ; 取数据表下半部份的代码
        MOVC A, @A+DPTR
        MOV P2, A ; 输出到 P2 口
        INC R0
MOV R3, #02 ; 扫描 1 毫秒
DELAY2: MOV R5, #248 ;
        DJNZ R5, $
        DJNZ R3, DELAY2
        MOV A, #00H ; 清除屏幕
        MOV P0, A
        ANL P2, #00H
        DJNZ R6, L3 ; 一个字 16 个码是否完成?
        DJNZ R1, L16 ; 每个字的停留时间是否到了?
        MOV 20H, R0 ; 取码指针存入 20H
        CJNE R0, #0FFH, L100 ; 8 个字 256 个码是否完成?
        JMP LOOP ; 反复循环

```

TABLE :

; 汉字“倚”的代码

```
db 01H, 00H, 02H, 00H, 04H, 00H, 1FH, 0FFH
```

```
db 0E2H, 00H, 22H, 00H, 22H, 0FCH, 26H, 88H
```

```
db 2AH, 88H, 0F2H, 88H, 2AH, 0FAH, 26H, 01H
```

```
db 63H, 0FEH, 26H, 00H, 02H, 00H, 00H, 00H
```

; 以下分别输入天，一，出，宝，刀，屠，龙，的代码，略。

end

电路中行方向由 p0 口和 p2 口完成扫描，由于 p0 口没有上拉电阻，因此接一个 4.7k*8 的排阻上拉。如没有排阻，也可用 8 个普通的 4.7k 1/8w 电阻。为提供负载能力，接 16 个 2n5551 的 NPN 三极管驱动。列方向则由 4—16 译码器 74LS154 完成扫描，它由 89C51 的 P1.0---P1.3 控制。同样，驱动部分则是 16 个 2N5401 的三极管完成的。

电路的供电为一片 LM7805 三端稳压器，耗电电流为 100Ma 左右。

采用一块 12*20cm 的万能电路板，应当选用质量好些的发光管，（否则有坏点现象，更换起来较麻烦）首先将 256 个发光管插入电路板，注意插入方向，同时使高度一致，行方向直接焊接起来，列方向则搭桥架空焊接，完成后用万用表测试一下如有不亮的更换掉。

然后找一个电脑硬盘的数据线，截取所需的长度，分别将行，列线引出至电路的相关管脚即可。原理图为了简洁，故只画出了示意图，行列方向只画出了2个三极管，屏幕只画出4个发光管，实际上发光管为256只，三极管行列方向各16只，一共32只。焊接过程认真仔细一天时间即可完成全部制作。将程序编译后烧写入89c51，插入40pin Ic座，即可看到屏幕轮流显示：“倚天一出宝刀屠龙”。

当然，你可将程序的汉字代码部分更换为您所需要的代码即可显示你所需要的汉字

元件清单：

名称	数量	规格
4.7k 1/8w	32	电阻
4.7k*8 排阻	1	
2n5551	16	小功率 NPN 三极管
2n5401	16	小功率 PNP 三极管
led	256	3mm 白发红高亮度
22P	2	瓷片电容
10uf/50v	1	电解电容
100uf/25v	2	电解电容
AT89C51	1	或 AT89S51
40pin Ic 座	1	插 89c51 用
12M	1	晶体
74LS154	1	或 74HC154
LM7805	1	稳压 IC
电源插座	1	
稳压电源	1	

LED 显示屏用电源的设计

林建伟，李震

西安普声电信有限责任公司，陕西西安 710043

1 引言

LED 显示屏是一种迅速发展起来的新型信息显示媒体。随着我国经济的不断发展，已被广泛应用于车站、宾馆、银行、医院等公共场合。显示屏电源是其重要组成部分，主要用来给显示屏发光二极管提供必要的工作电流，保证屏体正常显示。为简单起见，通常采用由一小功率电源带3到4个显示驱动板的供电方案。这样，一个较大面积的显示屏需要配接许多电源模块，例如一个2m×1.5m的屏体，就需要提供24个5V/20A的模块电源。该设计存在以下的缺点。

- 1) 接线复杂每一个电源均需单独地配置交流输入线、直流输出线。
- 2) 电源冗余度差在大多数情况下，屏体显示内容为文字、动画、图片，每个显示驱动板消耗的电流不一样，可能某些电源模块过载，而另一些模块空载。此外，若某一电源失效，会造成屏体的一部分黑屏。
- 3) 电源过载能力差，利用率低屏体在工作时消耗的电流随画面的内容、颜色、亮度而变化，大部分时间电流较小，而大面积高亮度的画面虽消耗电流大，但持续时间短。考虑到LED是恒流驱动的，只要驱动板可正常工作，供电电压可以降低一些。电源最好有下拖形状的限流特性，而不是通常的较陡峭形状的限流特性，以保证有较好的过载能力、较高的利用率。

考虑到以上各点，提出新的供电方案如下：

- 1) 集中供电，采用n+1冗余方案。
- 2) 电源模块设计适当的输出电流，模块可均流。保证屏体装配工艺易实现n+1冗余。
- 3) 电源模块有下拖形状的限流特性以保证有较好的过载能力、较高的利用率。
- 4) 电源模块有扁平的外形，自然散热，易于在屏体上安装，并利用屏体散热。

5) 电源模块带 APFC, 减小对电网的干扰, 适应电网的波动。

2 电路设计

采用集中供电方案可避免分散供电的缺点, 但要求电源的可靠性更高, 否则电源一旦失效会造成整屏的黑屏, 而不是部分黑屏。提高电源可靠性的最积极的办法为提高变换效率, 减少发热量, 同时选用可靠性高的线路与器件。

2. 1AC/DC 电路设计

传统的 AC/DC 全波整流电路采用的是整流+电容滤波电路。这种电路是一种非线性器件和储能元件的组合, 输入交流电压的波形是正弦的, 但输入电流的波形发生了严重的畸变, 呈脉冲状。由此产生的谐波电流对电网有危害作用, 使电源输入功率因素下降。在本设计中整流电路部分采用有源功率因数校正电路 (APFC), 避免了上述缺点。其电路如图 1 所示。

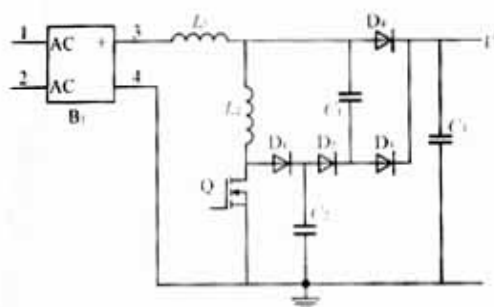


图 1 PFC 无损吸收主电路

与典型 PFC 主电路不同的是此电路选用了无损吸收缓冲网络。该网络降低了开关管的开关损耗, 提高了其稳定性, 增强了其使用寿命。它利用一组无源元件, 使开关管实现了零电流开通和零电压关断, 提高了电源的工作效率, 且相对于其它谐振软开关电路, 降低了生产成本。

下面通过分析 PFC 主开关 Q 的工作过程来说明此无损吸收缓冲网络的工作原理。

1) Q 导通时, 因为电感 L_2 中电流不能突变, 且 C_2 、 C_1 电压不能突变, Q 中的电流从零开始增加, 缓慢上升。通过 D_4 的电流 i_{D4} 渐减。Q 实现零电流开通, 导通的损耗较小。

2) 当电流 i_{D4} 减少为零时, D_4 进入反向恢复状态, 通过电感 L_2 的电流 $i_{L2} = i_{L1} + i_{rD4}$ 。 D_4 反向电流 i_{rD4} 的变化率受到电感 L_2 的控制, 反向恢复损耗降低。

3) 主电感 L_2 中电流缓慢增加, Q 上的电压 u_0 下降。电容 C_2 通过 D_2 、 C_1 、 L_2 、Q 放电, C_2 上的电压 u_{C2} 下降。

4) 当 u_{C2} 下降为零时, C_2 中的能量完全转向 C_1 、 L_2 。 L_2 中的电流饱和不变, u_0 下降变为零, Q 完成零电流通过程。

5) Q 保持开通状态, 与普通 PFC 电路的开关管状态相同。

6) Q 关断时, L_2 中的电流 i_{L2} 通过 D_1 流向 C_2 , C_2 从零开始充电, Q 实现零电压关断, 关断损耗较小。二极管 D_2 、 D_3 使 u_{C2} 最终钳位在输出电压 V_L 。

7) L_2 在导通时存储的能量通过 D_1 、 D_2 流向 C_1 , L_2 逐渐复位。当 L_2 复位后, C_1 中的能量通过 D_3 输出。

8) 当 C_1 两端电压变为零时, D_4 正向导通。Q 完成零电压关断过程。

9) Q 保持关断状态直到开始进入新的开关循环过程。

Q 的开关波形如图 2 所示; Q 的实测导通时间和关断时间如图 3 所示。(电源负载 22A)

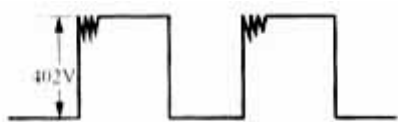


图2 Q的D-S板之间开关波形



图3 Q的导通时间和关断时间

从以上分析可知此无损吸收网络具有以下几个特点。

- 1) Q的最大工作电压等于输出电压 V_L 。
- 2) PFC电路的输出二极管 D_4 的耐压是 V_L 与电感 L_2 的反向电压之和。
- 3) Q中的电流上升率，即Q的开通损耗决定于电感 L_2 两端电压和 L_2 的电感量。
- 4) Q两端的电压上升率，即Q的关断损耗决定于流过电容 C_2 的电流和 C_2 的容量。
- 5) 由于开关动作引起的存储在 L_2 和 C_2 中的能量最终都输出给了负载，保证了转换器的工作效率。

2. 2DC/DC主电路设计

DC/DC主电路采用单端双正激电路。单端双正激电路相对于其它拓扑电路结构，开关管承受电压低，在控制电路设计中不必担心共态导通问题，也不会因电路不对称发生高频变压器单向偏磁，即不存在变压器饱和问题，是一种可靠性较高的电路。考虑到整机的高度不超过60mm，以及变压器工艺、安装、散热的要求，DC/DC变换采用双变压器、双输出电感结构。变压器原边并联，副边各自用一个输出电感，如图4所示。

该电路的无损吸收网络不同于AC/DC部分电路所采用的无损吸收网络。它仅使开关管完成了零电压关断过程。以下以开关 Q_2 为例 (Q_1 与 Q_2 变化状态相同)，简述该网络的工作原理。

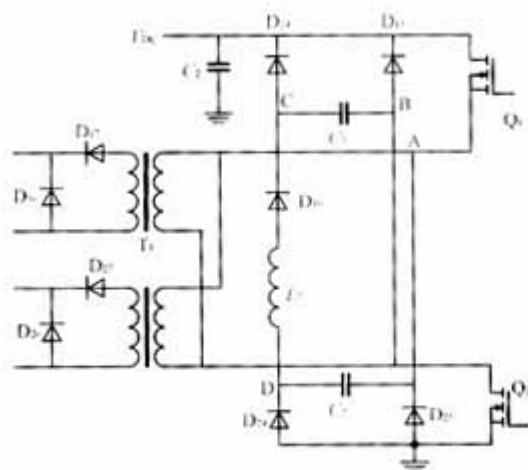


图4 双正激无损吸收主电路

1) 导通过程

Q_1 、 Q_2 开通时，除一路电流通过 Q_1 、 T_1 副边、 Q_2 外，另一路电流流过 Q_1 、 C_2

L_7 、 D_{10} 、 C_7 、 Q_2 形成 LC 振荡回路， C_5 、 C_7 被充电。当 A 与 B 点之间的电压 u_{AB} 等于主电路电压 V_{DC} 时，由于 D_{10} 的单向导电性，振荡结束。电感 L_7 起限制 C_7 、 C_5 中的电流变化的作用。 Q_1 、 Q_2 中流过的电流为从副边折算到原边的负载电流与 C_5 、 C_7 充电电流之和。

2) 关断过程

Q_1 、 Q_2 关断时，由于 B 点对地电压为零， C_7 从零开始充电， Q_2 对地电压 u_{02} 缓慢上升， Q_2 零电压关断。加在 Q_2 上的电压因二极管 D_{15} 的钳位作用，最终为 V_{DC} 。因此，B 点电压升为 V_{DC} 。 Q_2 实现零电压关断过程。

由于变压器励磁电感、漏感及引线寄生电感所引起的感应电势的能量通过 C_7 、 D_{14} 返回电源， Q_2 上的电压维持在 V_{DC} 直到变压器原边磁通复位。此时， Q_1 、 Q_2 上的电压分别为 $V_{DC}/2$ 直到新的工作周期。

Q_2 的开通期间与关断期间的状态与普通开关管同期间的状态相同。

图 5 为实测 Q_2 开关波形。图 6 为实测 Q_2 零电压关断波形。

从以上分析中，可以总结出以下特点。

- 1) 电路中每个开关管的最大工作电压等于电源电压。
- 2) Q_1 、 Q_2 关断的电压上升率分别决定于电容 C_5 、 C_7 的容量。

2.3 控制电路设计

为保证电源安全可靠地工作，电路设计中采用 TOP224Y 制作一反激式开关电源作为辅助源，如图 7 所示。其两路输出分别为 AC/DC 部分和 DC/DC 部分的控制电路供电。

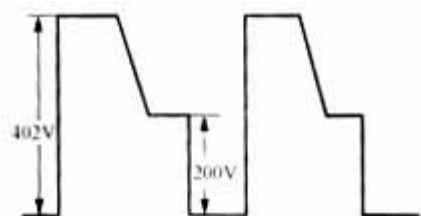


图 5 Q_2 的 D-S 极开关波形

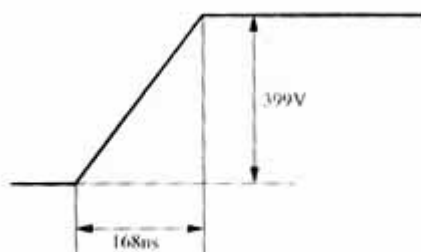


图 6 Q_2 的关断时间

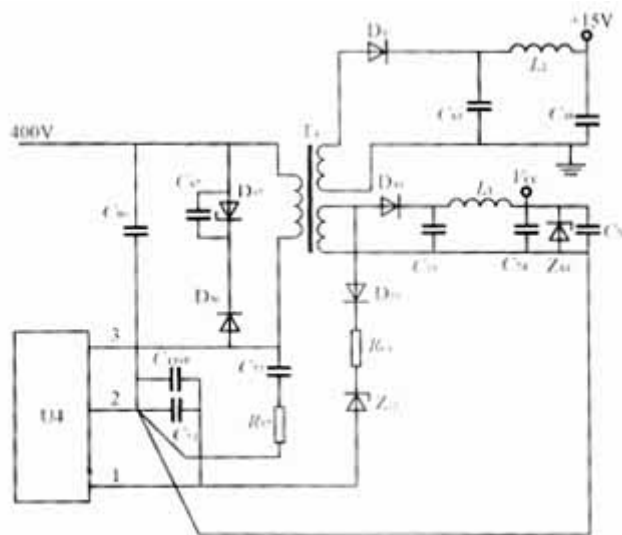


图7 辅助电源电路

AC/DC 控制部分使用 PFC 控制芯片 UC3854B。交流输入过、欠压、PFC 变换直流电压（400V）过、欠压时都关闭 UC3854，使 PFC 部分停止工作。这些故障信号通过隔离光耦传递到 DC/DC 控制电路，以达到在 AC/DC 部分工作不正常时保护主开关管的目的。

DC/DC 控制部分使用了 PWM 控制芯片 UC3846，采用峰值电流型控制模式。峰值电流型控制模式相对于电压控制模式，负载响应速率快，具有逐脉冲限流特性，容易获得下拖形状的限流特性，非常适合在此应用。

n+1 冗余应用时，多模块必须有均流功能。该电源输出电流较大，直接从 DC 输出用分流器取电流信号功率损耗较大，同时装配工艺较复杂。因此，本设计采取了原边电流合成的方法。

用电流传感器取出开关管导通时变压器原边的电流信号。该信号包含了变压器的励磁电流信号与输出电感电流折算到变压器原边的电流信号。因输出电感折算到原边的电流远大于变压器的励磁电流，所以可认为电流传感器取出的即为输出电感的充磁电流。这是输出电感电流的上升部分，只要模拟出输出电感续流时的下降部分，合成后即可得到输出电感的电流信号，也为输出电流信号。取出该合成后的电流信号后就可用于电流保护的 control 与均流控制上了。

如图 8 所示，把电流传感器取出的电流信号经高速单向缓冲后向一电容充电。开关管导通时关闭恒流源，而开关管关断时打开恒流源对电容恒流放电。在选择合适的电路参数后，电容上的电压波形就与输出电感上的电流成比例，放大后即可得到输出电感电流，也即输出电流。

3 实验结果

对样机的测试指标如下

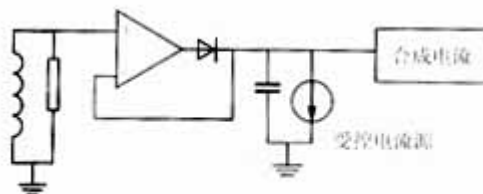


图8 输出电流合成电路

输入电压范围 AC150~270V

输出电压范围 DC4.6~6.3V

输出电流>120A

效率>80%

散热方式自然散热

限流特性下拖

厚度 60mm，可安装在 LED 显示屏体上

用两台样机试验均流如下：

A 机 5.6V B 机 5.3V 不接均流线 B 机不工作

A 机 5.6V B 机 5.3V 接均流线 A 机输出 34A，B 机输出 33A。

4 结语

随着这种电源应用量的不断增加，证明出其具有较高的工作效率和良好的可靠性，是一种性价比较高的产品。