

# T6963C 液晶控制器的应用

## 目 录

一、前 言	1
二、有关 T6963C 的一般介绍	1
1、T6963C 的特点	1
2、T6963C 的引脚说明及其功能	1
三、T6963C 指令集	3
1、指令设置指令：	3
2、控制字设置：	4
3、方式设置：	4
4、显示方式：	5
5、光标形状选择：	5
6、数据自动读 / 写方式设置：	5
7、数据一次读 / 写方式：	5
8、屏读：	5
9、屏拷贝：	5
10、位操作：	5
四、内藏 T6963C 的液晶显示模块与 MPU 的接口方法	6
五、内藏 T6963C 的液晶模块与 MPU 的接口方法	6
1、直接访问方式	6
2、间接控制方式	8
六、应用举例	9
1、清显示 RAM 的子程序	9
2、初始化子程序	10
3、建立 CGRAM	11
4、汉字显示	11
5、文本特征方式应用	13
七、调试步骤	14
附录：ROM 中的字符代码表	14



D-Peak

北京德彼克创新科技有限公司

地址：北京海淀区中关村南大街 48 号 邮编：100081 技术：010-82190029/31 /62175129-805, 806

业务：010-82190024/25/26/27 82190029/31 /62175129/62196724-801 传真：82190028

电子信箱：dpksales@public.bta.net.cn

网址：www.D-Peak.com

## 液晶显示控制器 T6963C 的应用

### 一、前言

T6963C 液晶显示控制器多用于较小规模的液晶显示器件，常被装配在图形液晶显示模块上，以内藏控制器式图形液晶显示模块的形式出现。本文将对 T6963C 作一般性的介绍，然后详细叙述内藏 T6963C 控制器的液晶模块应用。

### 二、有关 T6963C 的一般介绍

#### 1、T6963C 的特点

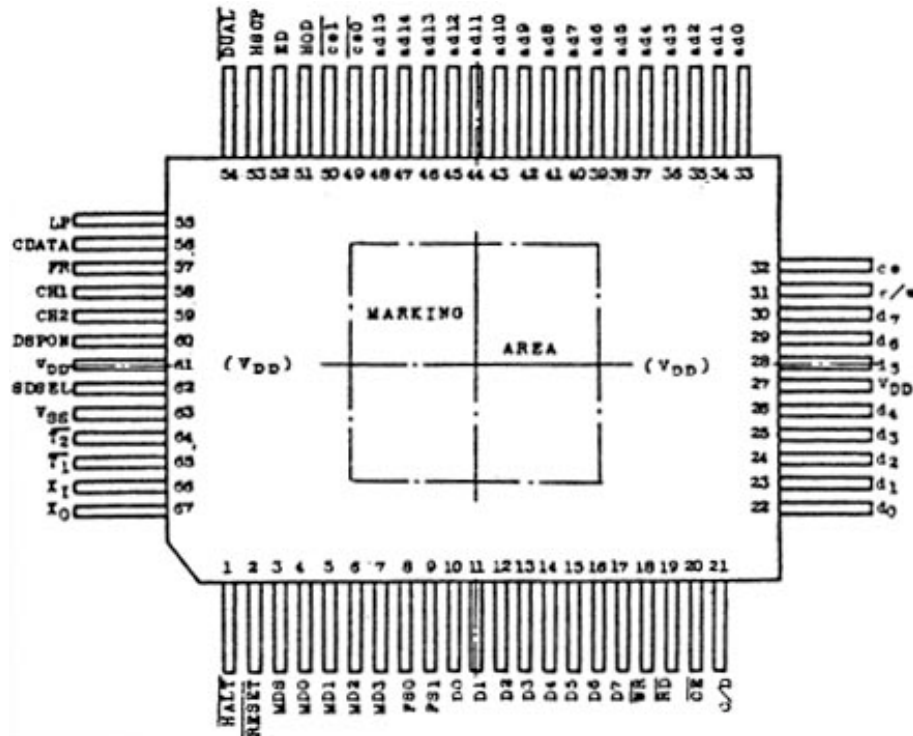
(1) T6963C 是点阵式液晶图形显示控制器，它能与 80 系列的 8 位微处理器直接接口。

(2) T6963C 的字符字体由硬件设置，其字体有 4 种：5×8、6×8、7×8、8×8。

(3) T6963C 的占空比可从 1/16 到 1/128。

(4) T6963C 可以图形方式、字符方式及图形和字符合成方式进行显示，还可以实现字符方式下的特征显示、可象 CAD 那样的屏拷贝操作等等。

(5) T6963C 具有内部字符发生器 CGRAM，共有 128 个字符（见附录），T6963C 可管理 64KRAM，作为显示缓冲区及字符发生器 CGRAM。并可允许 MPU 随时访问显示缓冲区，甚至可以进行位操作。



#### 2、T6963C 的引脚说明及其功能

T6963C 的 QFD 封装共有 67 个引脚，各引脚说明如下：

(1) D0—D7：T6963C 与 MPU 接口的数据总线，三态。

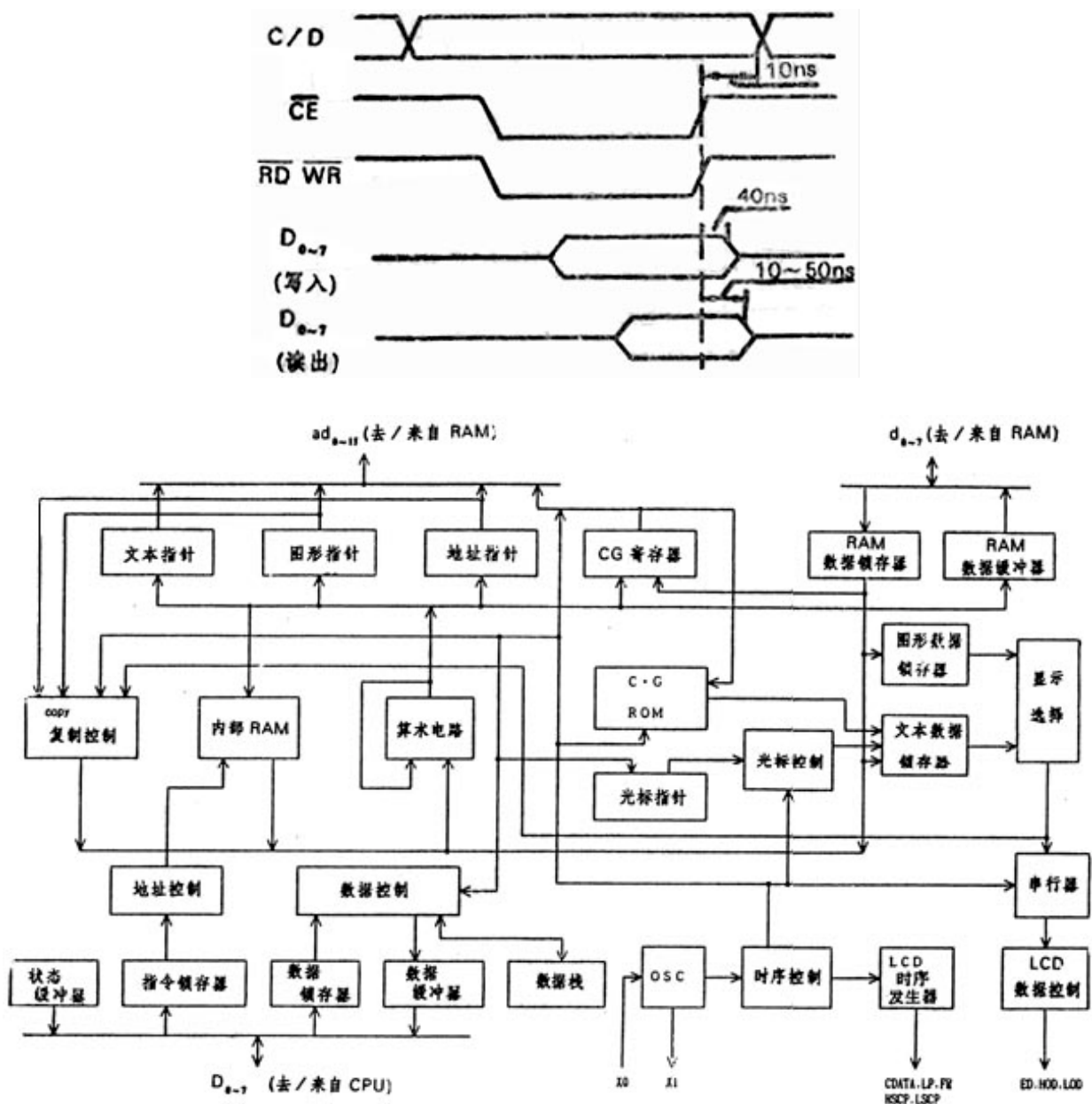
(2) RD, WR：读、写选通信号，低电平有效，输入信号。

(3) CE：T6963C 的片选信号，低电平有效。

(4) C/D：通道选择信号，1 为指令通道，0 为数据通道。

(5) RESET, HALT：RESET 为低电平有效的复位信号，它将行、列计数器和显示寄存器清零，关显示；HALT 具有 RESET 的基本功能，还将中止内部时钟振荡器的工作，以保护液晶显示器件。

以上引脚为 T6963C 与 MPU 接口的引脚，时序关系如下：



T6963C 原理图

(6) DUAL, SDSEL:

DUAL=1 为单屏结构, =0 为双屏结构;

SDSEL =0 为一位串行数据传输方式, =1 为 2 位并行数据传输方式。

(7) MD2, MD3: 设置显示窗口长度, 从而确定了列数据传输个数的最大值, 其组合逻辑关系如下:

MD3	1	1	0	0
MD2	1	0	1	0
每行字符数	32	40	64	80

(8) MDS, MD1, MD0: 设置显示窗口宽度(行), 从而确定 T6963C 的帧扫描信号的时序和显示驱动的占空比系数, 当 DUAL=1 时, 其组合功能如下:

MDS	0	0	0	0	1	1	1	1
MD1	1	1	0	0	1	1	0	0
MD0	1	0	1	0	1	0	1	0
字符行	2	4	6	8	10	12	14	16
总行数	16	32	48	64	80	96	112	128
占空比	1/16	1/32	1/48	1/64	1/80	1/96	1/112	1/128

当 DUAL=0 时，以上设置中的字符行和总行数增至原来的 2 倍，其它都不变，这种情况下的液晶屏结构为双屏结构。

(9) FS1, FS0: 显示字符的字体选择。

FS1	1	1	0	0
FS0	1	0	1	0
字体	5×8	6×8	7×8	8×8

(10) XI, XO: 振荡时钟引脚。

(11) AD0—15: 输出信号，显示缓冲区 16 位地址总线。

(12) D0—7: 三态，显示缓冲区 8 位数据总线。

(13) R/W: 输出，显示缓冲区读 / 写控制信号。

(14) CE: 输出，显示缓冲区片选信号，低电平有效。

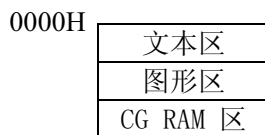
(15) CE0, CE1: 输出，DUAL=1 时的存储器片选信号。

(16) T1, T2, CH, CH2: 用来检测 T6963C 工作使用情况，T1, T2 作为测试信号输入端，CH, CH2 作为输出端。

(17) HOD, HSCP, LOD LSCP(CE1), ED LP, CDATE, FR 为 T6963C 驱动部信号，我们可以不作太多了解。

(18) 内存分配

单屏结构



双屏结构



### 三、T6963C 指令集

T6963C 的初始化设置一般都由硬件作了设置，因此其指令系统将集中于显示功能的设置上。T6963C 的指令可无参数、有一个或两个参数。每条指令的执行都是先送入参数（如果有的话），再送入指令代码。每次操作之前最好先进行状态字检测。T6963C 的状态字如下所示：

STA7	STA6	STA5	STA4	STA3	STA2	STA1
STA0: 指令读写状态			1: 准备好		0: 忙	
STA1: 数据读写状态			1: 准备好		0: 忙	
STA2: 数据自动读状态			1: 准备好		0: 忙	
STA3: 数据自动写状态			1: 准备好		0: 忙	
STA4: 未用						
STA5: 控制器运行检测可能性			1: 可能		0: 不能	
STA6: 屏读 / 拷贝出错状态			1: 出错		0: 正确	
STA7: 闪烁状态检测			1: 正常显示		0: 关显示	

由于状态位作用不一样，因此执行不同指令必须检测不同状态位。在 MPU 一次读/写指令和数据时，STA0 和 STA1 要同时有效——“准备好”状态。当 MPU 读/写数组时，判断 STA2 和 STA3 状态。屏读、屏拷贝指令使用 STA6。STA5 和 STA7 反映 T6963C 内部运行状态。指令系统的说明如下：

#### 1、指令设置指令：

D1,D2

0	0	1	0	0	N2	N1	N0
---	---	---	---	---	----	----	----

D1, D2 为第一和第二个参数，后一个字节为指令代码，根据 N0, N1, N2 的取值，该指令有三种含义(N0, N1, N2 不能有两个同时为 1)

D1	D2	指令代码	功能
水平位置(有效位 7 位)	垂直位置(有效 5 位)	21H(N0=1)	光标指针设置
地址(有效位 5 位)	00H	22H(N1=1)	CGRAM 偏置地址设置
低字节	高字节	24H(N2=1)	地址指针位置

A. 光标指针设置: D1 表示光标在实际液晶屏上离左上角的横向距离(字符数), D2 表示纵向距离(字符行)。

B. CGRAM 偏置地址寄存器设置: 设置了 2K CGRAM 在 64K 显示 RAM 的高 5 位地址(偏置地址), 地址的低 3 位为行计数器值, 中间 8 位则为字符代码。

CGRAM 的实际地址为:

逻辑地址	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	
偏置地址	C4	C3	C2	C1	C0												
字符代码							D7	D6	D5	D4	D3	D2	D1	D0			
行地址指针+)															R2	R1	R0
实际地址	V15	V14	V13	V12	V11	V10	V9	V8	V7	V6	V5	V4	V3	V2	V1	V0	

C. 地址指针设置则设置了一个当前要对其进行操作的显示缓冲区(RAM) 的一个单元, D1、D2 为该单元地址的低位和高位地址。

## 2、控制字设置:

D1,D2	0	1	0	0	0	0	N1	N0
-------	---	---	---	---	---	---	----	----

根据 N1, N0 的不同取值, 该指令有四种指令功能形式:

N1	N0	D1	D2	指令代码	功能
0	0	低字节	高字节	40H	文本区首址
0	1	字节数	00H	41H	文本区宽度(字节数/行)
1	0	低字节	高字节	42H	图形区首址
1	1	字节数	00H	43H	图形区宽度(字节数/行)

文本区和图形区首地址对应显示屏上左上角字符位或字节位, 修改该地址可以产生“卷动”效果。D1, D2 分别为该地址的低位和高位字节。文本区域宽度和图形区域宽度用于调整使用的有效显示窗口宽度, 表示每行可有效显示的字符数或字节数。T6963C 硬件设置的显示窗口宽度是指所允许的最大有效显示窗口宽度。需说明的是当硬件设置 6×8 字体时, 图形显示区单元的低 6 位有效, 对应显示屏上 6×1 显示位。

## 3、方式设置:

无参数	1	0	0	0	N3	N2	N1	N0
-----	---	---	---	---	----	----	----	----

N3: 字符发生器选择位, N3=1 为 CGRAM, 字符代码为 00H—FFH。=0 为 CGROM。由于 CGROM 字符代码为 00H—7FH。因此选用 80H—FFH 字符代码时, 早自动选择 CGRAM。

N2—N0: 合成显示方式控制位, 其组合功能如下表:

N2	N1	N0	合成方式
0	0	0	逻辑"或"合成
0	0	1	逻辑"异或"合成
0	1	1	逻辑"与"合成
1	0	0	文本特征

当设置文本方式和图形方式均打开时, 上述合成显示方式设置才有效。其中的文本特征方式是指将图形区改为文本特征区该区与文本相同, 每个字节作为对应的文本区的每个字符显示的特征, 包括字符显示与不显示、字符闪烁及字符的“负向”显示。通过这种方式, T6963C 可以控制每个字符的文本特征。文本特征区内, 字符的文本特征码由一个字节的低四位组成, 即:

D7	D6	D5	D4	D3	D2	D1	D0
*	*	*	*	d3	d2	d1	d0

d3: 字符闪烁控制位, =1 为闪烁, =0 为不闪烁,

d2—d0 的组合如下:

d2	d1	d0	显示效果
0	0	0	正向显示
1	0	1	负向显示
0	1	1	禁止显示, 空白

启用文本特征方式时可在原有图形区和文本区外另开一图形区作为文本特征区, 以保持原图形区的数

据。显示缓冲区可划分如下：(单屏结构)

SAD1	文本显示区	显示 缓冲区 RAM
SAD2	图形显示区	
SAD2'	文本特性区	
CGRAM(2k)		

#### 4、显示方式：

无参数	1	0	0	1	N3	N2	N1	N0
-----	---	---	---	---	----	----	----	----

- N0: 1/0, 光标闪烁启用 / 禁止
- N1: 1/0, 光标显示启用 / 禁止
- N2: 1/0, 文本显示启用 / 禁止
- N3: 1/0, 图形显示启用 / 禁止

#### 5、光标形状选择：

无参数	1	0	1	0	0	N2	N1	N0
-----	---	---	---	---	---	----	----	----

光标形状为 8 点(列)×N 行, N 的值为 0—7H, 由 N2—N0 确定

#### 6、数据自动读 / 写方式设置：

无参数	1	0	1	1	0	0	N1	N0
-----	---	---	---	---	---	---	----	----

该指令执行后, MPU 可以连续地读 / 写数据(来 / 去显示缓冲区 RAM), 每读 / 写一次地址指针自动增 1。自动读 / 写结束后, 必须写入自动结束命令以使 T6963C 退出自动读 / 写状态, 开始接受其它指令。N2, N0 组合功能如下：

N1	N2	指令代码	功 能
0	0	B0H	自动写设置
0	1	B1H	自动读设置
1	*	B2H/B3H	自动读 / 写结束

#### 7、数据一次读 / 写方式：

D1	1	1	0	0	0	N2	N1	N0
----	---	---	---	---	---	----	----	----

D1	N2	N1	N0	指令代码	功 能
数据	0	0	0	C0H	数据写, 地址加 1
—	0	0	1	C1H	数据读, 地址加 1
数据	0	1	0	C2H	数据写, 地址减 1
—	0	1	1	C3H	数据读, 地址减 1
数据	1	0	0	C4H	数据写, 地址不变
—	1	0	1	C5H	数据读, 地址不变

#### 8、屏读：

无参数	1	1	1	0	0	0	0	0
-----	---	---	---	---	---	---	---	---

该指令将当前由地址指针指向的某一位置上的显示状态(8×1 点阵)作为一个字节的数据送到 T6963C 的数据栈内, 等待 MPU 的读取, 它是文本数据与图形数据在该位置上的逻辑合成值。地址指针应在图形区内设置。

#### 9、屏拷贝：

无参数	1	1	1	0	1	0	0	0
-----	---	---	---	---	---	---	---	---

该指令将当前地址指针(图形区内)指向的位置开始的一行显示状态拷贝到相对应的图形显示区的一组单元内, 该指令不能用于文本特征方式下或双屏结构液晶显示器的应用上。

#### 10、位操作：

无参数	1	1	1	0	N3	N2	N1	N0
-----	---	---	---	---	----	----	----	----

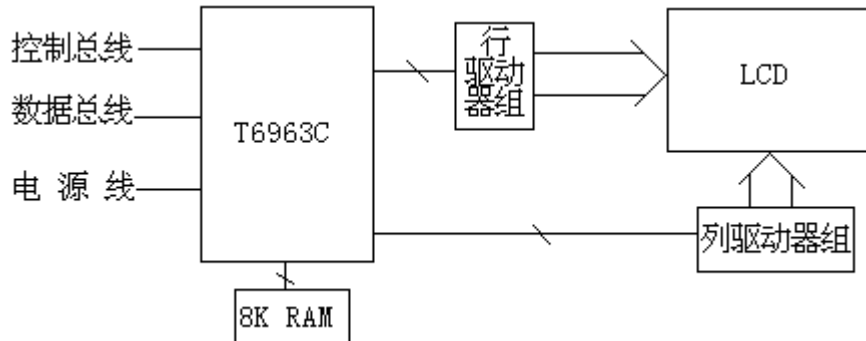
该指令可将显示缓冲区某单元的某一位清零或置 1, 该单元地址由当前地址指针提供。

N3=1 置 1, N3=0 清零。 N2—N0: 操作位对应该单元的 D0—D7 位。

#### 四、内藏 T6963C 的液晶显示模块的外特性

内藏 T6963C 的液晶显示模块上已经实现了 T6963C 与行、列驱动器及显示缓冲区 RAM 的接口，同时也用硬件设置了液晶屏的结构(单 / 双屏)，数据传输方式，显示窗口长度、宽度等等。内藏 T6963C 的单屏结构点阵图形液晶显示模块的方框图如下：

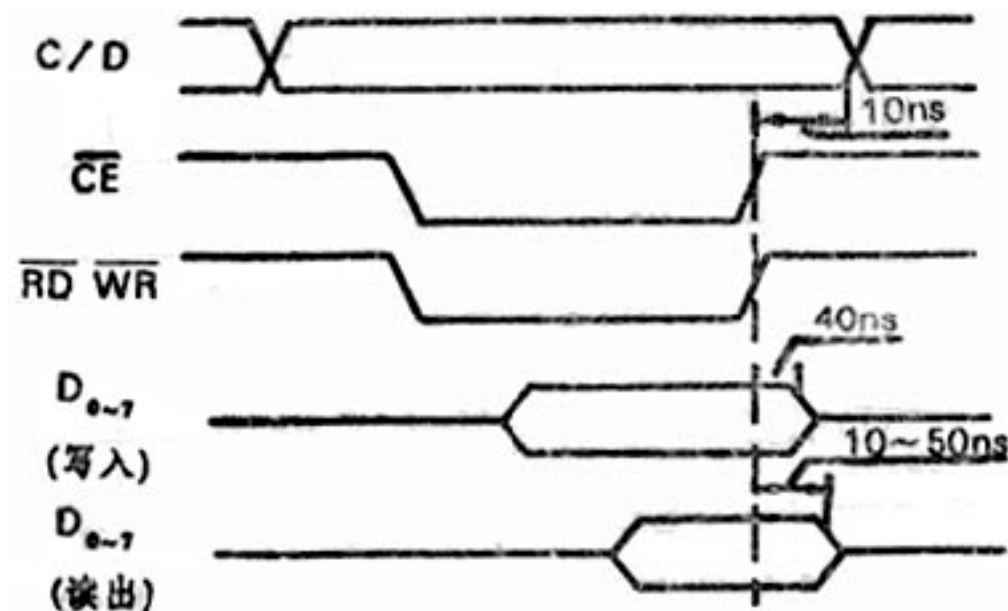
各模块的引脚及说明



1、2、3 引脚分别为 FG、GND、Vcc。

型号 / PIN	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
MGLS12864T	V <sub>0</sub>	WR	RD	CS	C/D	RST	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	FS		
MGLS24064T	V <sub>0</sub>	WR	RD	CS	C/D	NC	RST	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	FS	NC
MGLS240128T	V <sub>0</sub>	WR	RD	CS	C/D	RST	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	FS		
DMF5001	V <sub>adj</sub>	V <sub>ee</sub>	WR	RD	CS	C/D	HALT	RST	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	NC
DMF5002	V <sub>adj</sub>	V <sub>ee</sub>	WR	RD	CS	C/D	HALT	RST	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	NC
DMF5003	(同 DMF5001, 为冷阴极背光)																
DMF5005	V <sub>ee</sub>	WR	RD	CS	C/D	NC	RST	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	FS	NC
DMF5010	(同 DMF5005, 为冷阴极背光)																
DMF50172(双屏)	V <sub>adj</sub>	WR	RD	CS	C/D	NC	RST	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	NC	NC

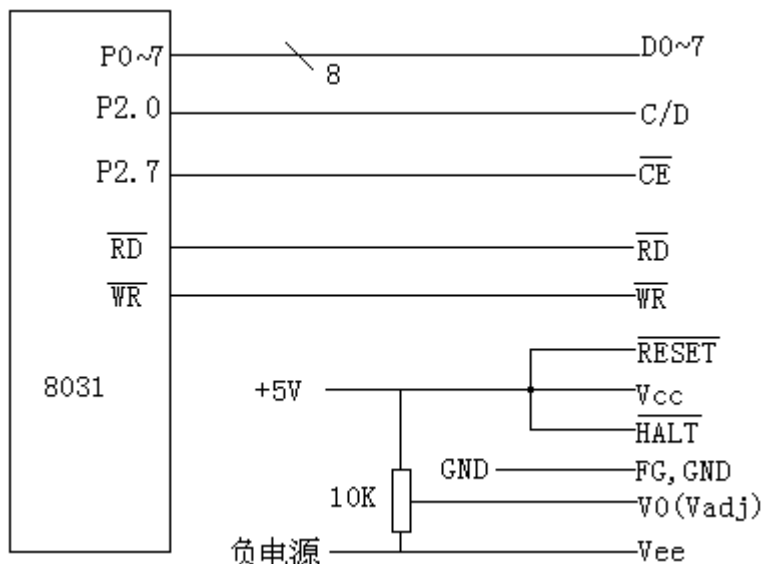
与 MPU 的接口时序



#### 五、内藏 T6963C 的液晶模块与 MPU 的接口方法

##### 1、直接访问方式

MPU 利用数据总线与控制信号直接采用存储器访问形式或 I/O 设备型号控制 T6963C 类液晶显示模块。接口电路如下：



8031 数据口 P0 直接与液晶显示模块的数据口连接，由于 T6963C 接口适用于 8080 系列和 Z80 系列 CPU，所以可以直接用 8031 的  $\overline{RD}$ 、 $\overline{WR}$  作为液晶显示模块的读、写控制信号，液晶显示模块  $\overline{RESET}$ 、 $\overline{HALT}$  挂在+5V 上。 $\overline{CE}$  信号可由地址线译码或与  $\overline{MREQ}$  或  $\overline{IORQ}$  共同产生。例如，取 138 译码器 Y0 输出。C/D 信号由 8031 地址线 A0 提供，A0=1 为指令口地址；A0=0 为数据口地址。各驱动子程序如下：

(1) 读标志字子程序

占用寄存器：DPTR, A; 输出寄存器：A 存储标志字

```
PR0: MOV DPTR, #CADD ;指令口地址
      MOVX A, @DPTR ;读标志字
      RET
```

由此程序派生出判断有关标志位的子程序：

(1) 判断 STA0、STA1 状态位子程序，在写指令的读、写数据之前这两个标志位必须同时为“1”：

```
PR01: ACALL PR0
       JNB ACC.0, PR01 ;判别 STA0
       JNB ACC.1, PR01 ;判别 STA1
       RET
```

(2) 判断 STA2 标志位子程序，该位在数据自动读操作过程中取代 STA0 和 STA1 有效。在连续读过程中每读一次之前都要确认 STA2=1。

```
PR02: ACALL PR0
       JNB ACC.2, PR02 ;判别 STA2
       RET
```

同理，数据自动写标志位 STA3 的判断子程序如下：

```
PR03: ACALL PR0
       JNB ACC.3, PR03 ;判别 STA3
       RET
```

(3) 在屏读或屏拷贝指令执行后，紧接着要判断 STA6 标志位，若 STA6=0，则表示指令条件正确，执行有效。如：

```
PR06: ACALL PR0
       JB ACC.6, ERR ;判别 STA6
       RET
```

ERR: ..... ;出错处理程序

(4) 写指令和数据子程序



有了判断各标志位的子程序，写指令的子程序也随之而来了。如：

占用寄存器：DPTR, R2, R3, R4, A:

输入寄存器：R2 参数第一字节, R3 参数第二字节, R4 指令代码

```

PR1:  ACALL  PR01      ;双字节参数指令入口
      MOV    A, R2
      ACALL  PR14
PR11: ACALL  PR01      ;单字节参数指令入口
      MOV    A, R3
      ACALL  PR14
PR12: ACALL  PR01      ;无参数指令入口
      MOV    A, R4
      SJMP  PR15
PR14: MOV    DPTR, #DADD ;写数据入口, 数据口地址
PR15: MOVX   @DPTR, A
      RET
    
```

此程序是通用程序，当写入单参数指令时，应把参数或数据送入 R3 内，其余程序入口为 PR11。无参数指令写入子程序入口为 PR12。

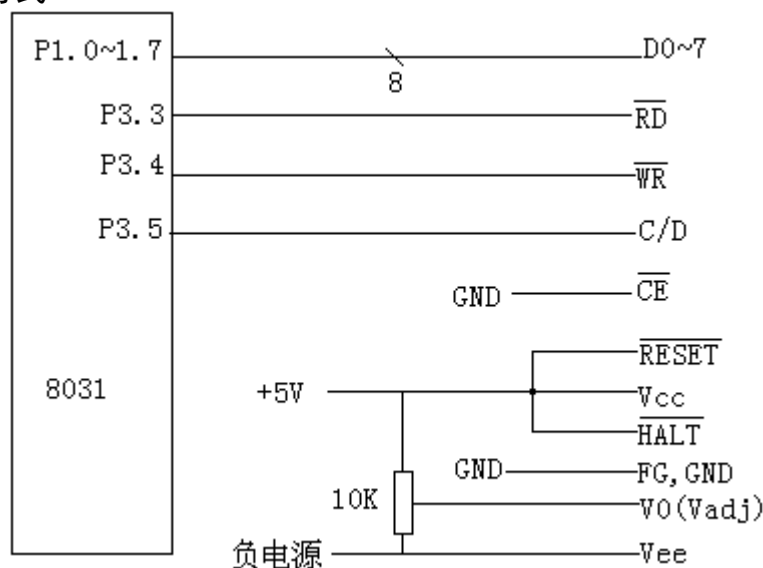
(3) 读数据子程序

占用寄存器：R0, A; 输出寄存器：A 存储数据

```

PR2:  ACALL  PR01
      MOV    DPTR, #DADD
      MOVX   A, @DPTR
      RET
    
```

## 2、间接控制方式



间接控制方式是 MPU 通过并行接口间接实现对液晶显示模块控制。根据液晶显示模块的需要，并行接口需要一个8 位的并行 P3 口中 3 位作为读、写及寄存器选择信号。由于并行接口只用于液晶显示模块，所以 CE 信号接地就行了。MPU 通过并行接口操纵液晶显示模块，要对其时序关系有一个清楚的了解，并应在程序中明确地反应出来。间接控制方式的基本程序如下：

(1) 读标志字子程序

占用寄存器：A; 输出寄存器：A 存储标志字

```

PR0:  ORL    P3, #38H    ;控制口初始化, C/D=1, RD=1, WR=1
    
```

```

ORL    P1, #0FFH    ;P1 口为输入方式
CLR    P3.3         ;RD= 0
MOV    A, P1        ;读标志字
SETB   P3.3         ;RD=1
RET

```

此程序可以用直接访问方式中的 PR01、PR02、PR03 和 PR06 等于程序直接调用。

(2) 写指令和数据子程序

占用寄存器: R2, R3, R4, A:

输入寄存器: R2 参数第一字节, R3 参数第二字节, R4 指令代码

```

PR1:  ACALL  PR01    ;双字节参数指令入口
      MOV    A, R2
      ACALL  PR14
PR11: ACALL  PR01    ;单字节参数指令入口
      MOV    A, R3
      ACALL  PR14
PR12: ACALL  PR01    ;无参数指令入口
      MOV    A, R4
      SJMP  PR15
PR14: CLR    P3.5    ;写数据入口, 数据口地址
PR15: CLR    P3.4
      MOV    P1, A
      SETB  P3.4
      SETB  P3.5
      RET

```

(3) 读数据子程序

占用寄存器: A; 输出寄存器: A 存储数据

```

PR2:  ACALL  PR01
      CLR    P3.5    ;C / D= 0
      ORL   P1, #0FFH ;P1 口输入方式
      CLR    P3.3    ;RD= 0
      MOV    A, P1    ;读数据
      ORL   P3, #38H ;RD=C/D=1
      RET

```

## 六、应用举例

### 1、清显示 RAM 的子程序

```

PR3:  MOV    R2, #00H ;RAM 地址=0000H; 双屏结构液晶, 下半屏起始地址为 8000H
      MOV    R3, #00H
      MOV    R4, #24H ;设置地址指令
      ACALL  PR1
      MOV    R4, #0B0H ;自动写指令
      ACALL  PR12
      MOV    R2, #1FH  ;循环量 8K 字节
PR31: MOV    R3, #0FFH
PR32: ACALL  PR03    ;判别 STA3
      CLR    A

```

```
ACALL PR14 ;写 00H
DJNZ R3, PR32
DJNZ R2, PR31
MOV R4, #0B2H ;自动写结束指令
ACALL PR12
RET
```

## 2、初始化子程序；双屏结构液晶，下半屏特性与上半屏一样，但下半屏起始地址为 8000H

```
PR4: LCALL PR3
MOV R2, #00H ;设置文本区
MOV R3, #00H
MOV R4, #40H
ACALL PR1
MOV R2, #DB ;见*注
MOV R3, #00H
MOV R4, #41H
ACALL PR1
MOV R2, #00H ;设置图形区
MOV R3, #08H
MOV R4, #42H
ACALL PR1
MOV R2, #DB ;见*注
MOV R3, #00H
MOV R4, #43H
ACALL PR1
MOV R4, #80H ;设置显示方式
ACALL PR12
MOV R4, #94H ;设置显示开关，开文本显示
ACALL PR12
RET ;初始化完成
```

下面是初调程序，在屏左上角显示“！”

```
MOV R2, #00H
MOV R3, #00H
MOV R4, #24H
LCALL PR1
MOV R3, #01H
MOV R4, #0C0H
LCALL PR11
RET
```

\*DB: 与 LCM 屏面一行字符数有关 (8×8)，选择如下：

LCM 点阵	DB 值	说 明	LCM 点阵	DB 值	说 明
128×64	10H	16 字节 / 行	160×128	14H	20 字节 / 行
240×64	1EH	30 字节 / 行	128×112	10H	16 字节 / 行

## 3、建立 CGRAM

- (1) 设置偏置寄存器内容
- (2) 建立字符的字模数据及字符代码

以汉字“德”为例，字符代码的选择范围在 80H~FFH 之间，则“德”字的结构字模数及字符代码的

定义如下表所示:

“德”字结构	字 模 数 据								字符代码
左上部	00H	04H	06H	0CH	19H	34H	2FH	0FH	80H
左下部	1BH	39H	39H	6BH	0FH	1EH	1CH	08H	81H
右上部	60H	60H	78H	78H	0ECH	7CH	0FCH	6CH	82H
右下部	78H	0F8H	0EEH	66H	0BCH	0CEH	7EH	00H	83H

(3) 写入 CGRAM

占用寄存器: R2, R3, R4, DPTR

```

PR5: MOV R2, #03H
      MOV R3, #00H
      MOV R4, #22H ;偏置寄存器设置指令
      ACALL PR1
      MOV R2, #00H
      MOV R3, #1CH ;CGRAM 地址
      MOV R4, #24H ;地址指针设置
      ACALL PR1
      MOV R4, #0B0H ;自动写指令
      ACALL PR12
      MOV R2, #00H ;数组量
PR51: ACALL PR03 ;判断 STA3
      MOV DPTR, #TAB1 ;数组首地址
      MOV A, R2
      MOVC A, @A+DPTR ;取数据
      ACALL PR14 ;写入 CGRAM
      INC R2
      CJNE R1, #20H, PR51
      WOV R4, #0B2H ;自动定结束指令
      ACALL PR12
      RET
    
```

“德”字字模数组

```

TAB1: DB 00H 04H 06H 0CH 19H 34H 2FH 0FH
      DB 1BH 39H 39H 6BH 0FH 1EH 1CH 08H
      DB 60H 60H 78H 78H 0ECH 7CH 0FCH 6CH
      DB 78H 0F8H 0EEH 66H 0BCH 0CEH 7EH 00H
    
```

## 4、汉字显示

### 1) 文本方式下汉字显示

地址指针已设置在要写入汉字的左上角位置

占用寄存器: R2, R3, R4, A; 输入寄存器: R3 存储汉字代码

```

PR6: MOV R4, #0C0H ;写数据, 地址加 1 指令
      ACALL PR11 ;R3 在主程序中设定, 如 R3=80H
      INC R3
      INC R3
      ACALL PR11 ;R3=82H
      DEC R3 ;R3=81H
      MOV R2, #(DB—2);DB 值见 PR4 注
      MOV R4, #0C1H ;读数据,地址加 1 指令
    
```

```

PR61: ACALL  PR12
      DJNZ   R2, PR61      ;地址指针移到汉字左下角
      MOV    R4, #0C0H
      ACALL  PR11
      INC   R3
      INC   R3              ;R3=83H
      ACALL  PR11
      RET
    
```

(2) 图形方式下汉字显示

现在从计算机中提取汉字字模已是非常容易的事了。本公司提供从计算机中提取汉字字模的程序，根据计算机的 16 点阵汉字字模数据的排布，如下：

1	17
2	18
3	19
...	...
16	32

PR7 为图形方式下汉字显示的子程序：

占用寄存器：A, B, R2, R3, R4, R5, DPTR；输入寄存器：A 汉字代码

占用位地址：00H, 00H=0 为“正向”显示；00H=1 为“负向”显示

```

PR7:  MOV   R2, #10H      ;循环量
      MOV   DPTR, #CCTAB
      MOV   B, #20H      ;一字模的字节数
      MUL  AB             ;确定所需汉字数组的首地址
      ADD  A, DPL
      PUSH ACC
      MOV  A, B
      ADDC A, DPH
      PUSH ACC
      MOV  R5, #00H      ;变址
PR71: POP  DPH
      POP  DPL
      MOV  A, R5
      MOVC A, @A+DPTR
      PUSH DPL
      PUSH DPH
      JNB  00H, PR72
      CPL  A
PR72: MOV  R3, A
      MOV  R4, #0C0H
      LCALL PR11
      INC  R5
      POP  DPH
      POP  DPL
      MOV  A, R5
      MOVC A, @A+DPTR
      PUSH DPL
      PUSH DPH
    
```

```

        JNB    00H, PR73
        CPL    A
PR73: MOV    R3, A
        LCALL PR11
        MOV    R3, # (DB—2) ;DB 值见 PR4 注
        MOV    R4, #0C1H
PR74: LCALL PR12
        DJNZ   R3, PR74
        INC    R5
        DJNZ   R2, PR71
        RET

```

CCTAB: DB: ... ;汉字字模

### 5、文本特征方式应用

在屏左上角显示“WELCOME!”, “WELCOME”负向闪烁显示, “!”正向显示, 子程序为 PR8, 其中 TAB2 为字符代码, TAB3 为相应的文本特征码。

```

PR8:  MOV    R4, #9CH ;启用文本和图形方式
      LCALL PR12
      MOV    R4, #84H ;设置文本特征方式
      LCALL PR12
      MOV    R2, #00H
      MOV    R3, #00H
      MOV    R4, #24H ;地址指针设置
      LCALL PR1
      MOV    R2, #00H ;写入字符数
PR81: MOV    DPTR, #TAB2 ;字符表
      MOV    A, R2
      MOVC   A, @A+DPTR
      MOV    R4, #0C0H
      LCALL PR11
      INC    R2
      CJNE   R2, #08H, PR81
      MOV    R2, #00H
      MOV    R3, #08H
      MOV    R4, #24H ;地址指针设置
      LCALL PR1
      MOV    R2, #00H ;写入字符数
PR82: MOV    DPTR, #TAB3 ;字符表
      MOV    A, R2
      MOVC   A, @A+DPTR
      MOV    R4, # 0C0H
      LCALL PR11
      INC    R2
      CJNE   R2, #08H, PR82
      RET

```

TAB2: DB 37H 25H 2CH 23H 2FH 2DH 25H 01H

TAB3: DB 0DH 0DH 0DH 0DH 0DH 0DH 0DH 0DH 08H

## 七、调试步骤

- 1、正确接线。注意电源，尤其是负电源接线是否正确；
- 2、通电，调节对比度电位器使屏显示区域呈现比未加电时的底色要深；
- 3、执行程序，调节对比度电位器，观看显示效果。

调试失败原因：

- 1、接线错误
- 2、地址选择错误
- 3、程序输入有误
- 4、仿真器控制信号未借出
- 5、初始化设置有误

## 八、相关液晶块的一些电气参数（参考值）

工作负电压，测试条件：温度 20℃；电源电压：4.9±0.1V

MGLS—12864T	V0=-5.4V	DMF5001N	VADJ=-14.45V； VEE=-15.00V
MGLS—12864T-HT	V0=-10.5V	DMF5002N	VADJ=-13.05V； VEE=-13.75V
MGLS—24064T	V0=-10.5V	DMF5005N	V0=-7.8V
MGLS—240128T	V0=-16V	DMF50172N	VADJ=-14.45V

## 附录：ROM 中的字符代码表

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
3	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
4	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
5	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
6	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
7	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F