

第十三章 内置控制器型图形液晶显示模块的应用

——HD61202U 液晶显示驱动控制器

在小规模点阵液晶显示模块上使用液晶显示驱动控制器组成液晶显示驱动控制系统是非常有益的。这使得液晶显示模块的硬件电路简单化,从而使模块的成本降低。但是这也使软件功能的要求大大提高。也就是说,许多显示功能如光标、字符库、闪烁等都需要由软件编制而成。HD61203U,HD61202U 就是这类液晶显示驱动控制器套件。之所以称它们为套件是因为 HD61203U,HD61202U 必须配套使用。以香港精电公司产品为例,有 MGLS12864 和 MGLS19264 两种规格。本节将以此模块为例,详细叙述内置 HD61202U 图形液晶显示模块的应用。

第一节 内置 HD61202U 图形液晶显示模块的电路特性

内置 HD61202U 图形液晶显示模块的驱动和控制系统是由一片 HD61203U 作为 64 路行驱动器,两片或三片 HD61202U 组成 128 点列或 192 点列的列驱动器组。所以内置 HD61202U 图形液晶显示模块的电路特性实际上是 HD61203U 和 HD61202U 组合的电路特性。

一、HD61203U 的电路特性

HD61203U 是带有振荡器和显示时序发生器的行驱动器。它具有 64 路行驱动输出,但它的特点在于它本身带有振荡器和时序发生器,通过外接振荡电阻电容使其上电后就以其设定的占空比系数 $1/N$ 值开始行扫描工作;它可以自行完成行、列驱动时序的生成及分配,自动进行行驱动的工作;同时向列驱动器输出同步信号及显示数据驱动所需的脉冲时序,控制列驱动器工作。HD61203U 与 HD61202U 组成完整的图形驱动控制系统。

HD61203U 的原理框图和引脚图分别如图 13-1 和图 13-2 所示。

HD61203U 内部的振荡器是 RC 振荡器。它通过 R,CR 和 C 三端外接振荡电阻 R_f 和振荡电容 C_f 就可以自行工作;它也可以直接通过 CR 端外接一个时钟信号工作。振荡器的工作将使时序发生电路工作以产生显示时序信号和与其相配的列驱动器的工作时钟信号。这些输出信号有:

- | | |
|------------------|--|
| CL2 | 数据锁存信号;提供给列驱动器作为数据锁存同步信号。 |
| M | 交流驱动波形信号;提供给列驱动器作为交流驱动波形信号。 |
| FRM | 帧信号;提供给列驱动器作为显示同步信号。 |
| $\phi 1, \phi 2$ | 工作时钟信号;提供给列驱动器作为列驱动器内部逻辑电路工作时钟信号。 |
| DL, DR | 双向移位寄存器输入/输出端;提供串行行数据输出,可以级联多个 HD61203U 一起工作。此时全部作为扫描信号输出。 |

HD61203U 的振荡器频率和显示扫描占空比是可以使用引脚电平选择的。

FS 工作频率选择端;当显示帧频为 70HZ 时:

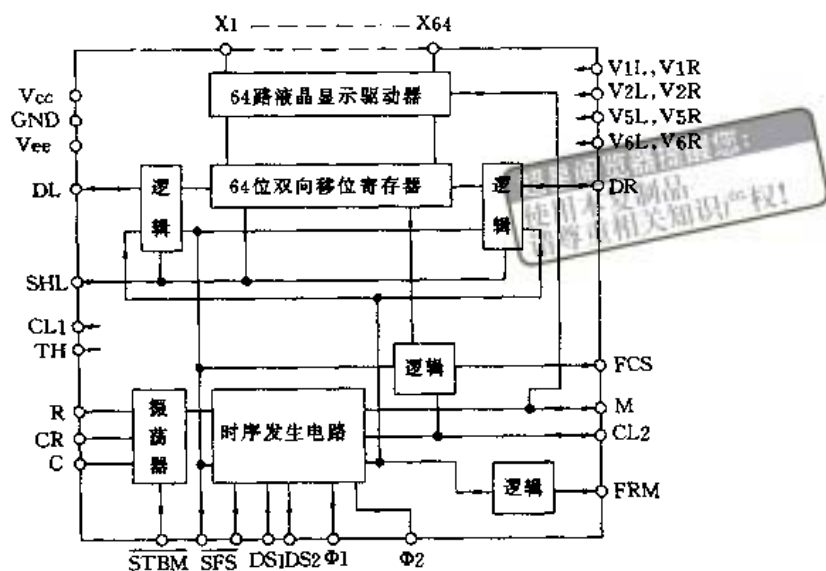


图 13-1 HD61203U 原理框图

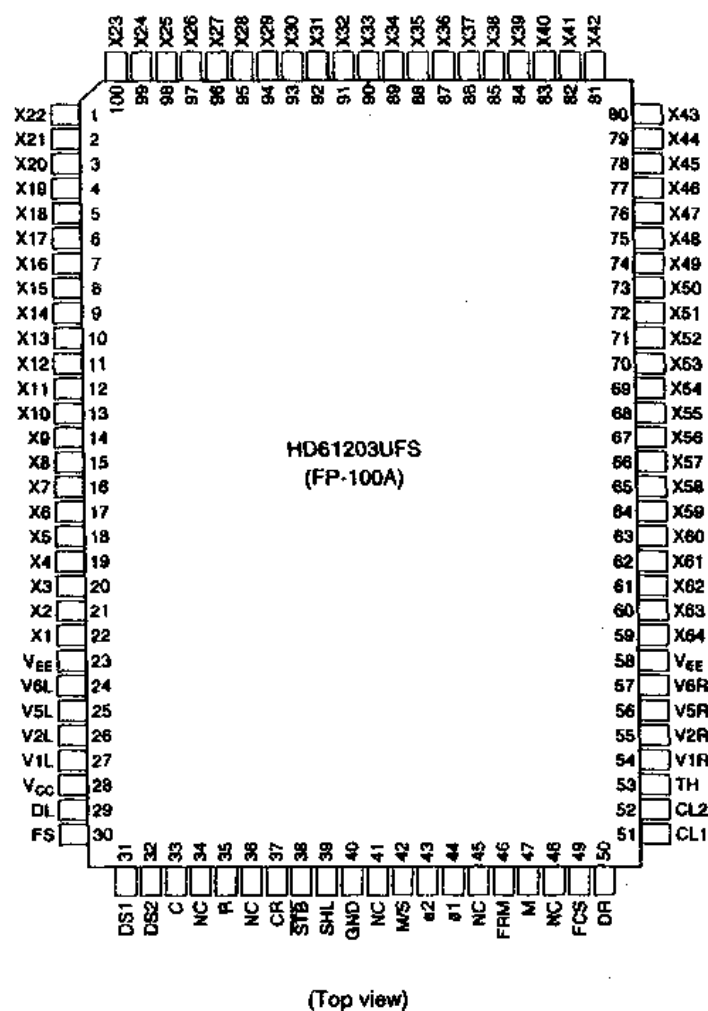


图 13-2 HD61203U 引脚图

$FS = V_{CC}$ $F_{osc} = 430\text{kHz}$

$FS = GND$ $F_{osc} = 215\text{kHz}$

DS1, DS2 显示占空比的选择端；它们的电平组合设置了 HD61203U 四种扫描行数和占空比值(如表 13-1 所示)。

表 13-1

DS1	DS2	占空比值
GND	GND	1/48
GND	V_{CC}	1/64
V_{CC}	GND	1/96
V_{CC}	V_{CC}	1/128

HD61203U 作为行驱动器没有锁存器，移位寄存器直接控制驱动电平；数据移位寄存器是双向传输的，传输方向由 SHL 设置。移位脉冲由 CL2 同步提供，移位脉冲的相位选择由 FCS 端设置，当 $FCS = V_{CC}$ 时，CL2 的上升沿有效； $FCS = GND$ 时，CL2 的下降沿有效；HD61203U 驱动器是低阻抗点阵液晶图形显示行驱动器，64 路 X1~X64 驱动端可以直接驱动较大面积的液晶显示器件的行电极。驱动输出由交流驱动波形 M 和液晶显示驱动电压组合产生(如表 13-2 所示)。

表 13-2

移位寄存器数据	M	输出电平	含 义
0	1	V6L, V6R	非选显示高电平
0	0	V5L, V5R	非选显示低电平
1	1	V2L, V2R	选择显示低电平
1	0	V1L, V1R	选择显示高电平

HD61203U 有两种工作方式，一种为主工作方式，功能如上所述；另一种为从工作方式，此时 HD61203U 振荡器和显示时序发生器停止工作，振荡器的 C 端和 R 端开路，CR 端接至 V_{CC} 。HD61203U 纯粹是行驱动功能，它的时序信号全部由外部器件提供，信号端 CL2 和 M 均为输入方式； $\phi 1, \phi 2$ 和 FRM 作开路态；DS1, DS2 和 FS 接至 V_{CC} ；DL 和 DR 受 SHL 控制选择输入/输出方式，接收或传送帧信号 FLM，如表 13-3 所示。

表 13-3

SHL	DL	DR
V_{CC}	输入	输出
GND	输出	输入

HD61203U 的工作方式是可以设置主/从方式设置端 M/S 的电平来选择。当 M/S 为“1”时，HD61203U 为主工作方式，当 M/S 为“0”时，HD61203U 为从工作方式。

HD61203U 的其他引脚作用如下：

V_{CC} 逻辑电源
 GND 逻辑电源地
 V_{ee} 驱动器逻辑负电源

$\overline{\text{STB}}$, TH, CL1 测试端不用时 $\overline{\text{STB}}$ 接 V_{CC} , TH 和 CL1 接 GND

HD61103U 由 CMOS 工艺制作, 功耗低, 在显示期间功耗最大仅为 5mW。足以用电池支持其工作。HD61203U 直流电特性如表 13-4 所示。

表 13-4

项 目	符号	测试条件	最小值	典型值	最大值	单位
逻辑电源	V_{CC}		2.7		5.5	V
驱动电源	V_{EE}		$V_{\text{CC}}-17$		$V_{\text{CC}}+0.3$	V
振荡器频率	F_{osc}	$R_f=47\text{k}\Omega$ $C_f=20\text{pF}$	315	450	585	kHz
电源电流	I_{CC}	1/128 占空比 R_fC_f 同上	—	—	1.0	mA
电源电流	I_{CE}	1/128 占空比	—	—	100	μA
工作温度	T_{opr}		-20	—	75	$^{\circ}\text{C}$
存储温度	T_{stg}		-55	—	125	$^{\circ}\text{C}$

二、HD61202U 的电特性

HD61202U 是带显示存储器的图形液晶显示列驱动控制器。它的特点是内置 64×64 位的显示存储器, 显示屏上各像素点的显示状态与显示存储器的各位数据一一对应, 显示存储器的数据直接作为图形显示的驱动信号。显示数据为“1”, 相应的像素点显示; 显示数据为“0”, 相应的像素点就不显示。同时 HD61202U 配备了一套显示存储器的管理电路和与计算机的接口电路, 允许计算机直接访问显示存储器, 也就是说 HD61202U 可以直接与计算机的总线连接。HD61202U 原理框图和引脚图分别如图 13-3 和图 13-4 所示。但是 HD61202U 不能独立工作, 因为它本身不能生成显示时序, 所以 HD61202U 需要与相应的带振荡器和显示时序发生器的行驱动器 HD61203U 配套才能形成一个完整的液晶驱动和控制系统。这个系统将省去了计算机与驱动器之间的液晶显示控制器, 由此组成的液晶显示模块的驱动和控制系统必然会大大降低模块的成本。

HD61202U 的主要特性为:

- 拥有 64×64 位(512 字节)的显示存储器, 其数据直接作为显示驱动信号。
- 8 位并行数据接口, 适配 M6800 系列时序。
- 64 路列驱动输出。
- 简单的操作指令 显示开关设置, 显示起始行设置, 地址指针设置和数据读/写等指令。
- 低功耗, 在显示期间功耗最大为 2mW。
- 宽电压工作 $V_{\text{CC}}=2.7\text{V} \sim 5.5\text{V}$
 $V_{\text{EE}}=0\text{V} \sim -10\text{V}$

HD61202U 是内置 HD61202U 图形液晶显示模块的电路特性的核心, 下面我们按照控制器的三分析方法分析 HD61202U 的工作原理。

一、控制部

HD61202U 的控制部是 HD61202U 的核心。HD61202U 工作的逻辑时序信号需要由外部电路提供, 它本身不能生成。 ϕ_1, ϕ_2 端就是作为 HD61202U 内部逻辑电路工作的时钟信号的

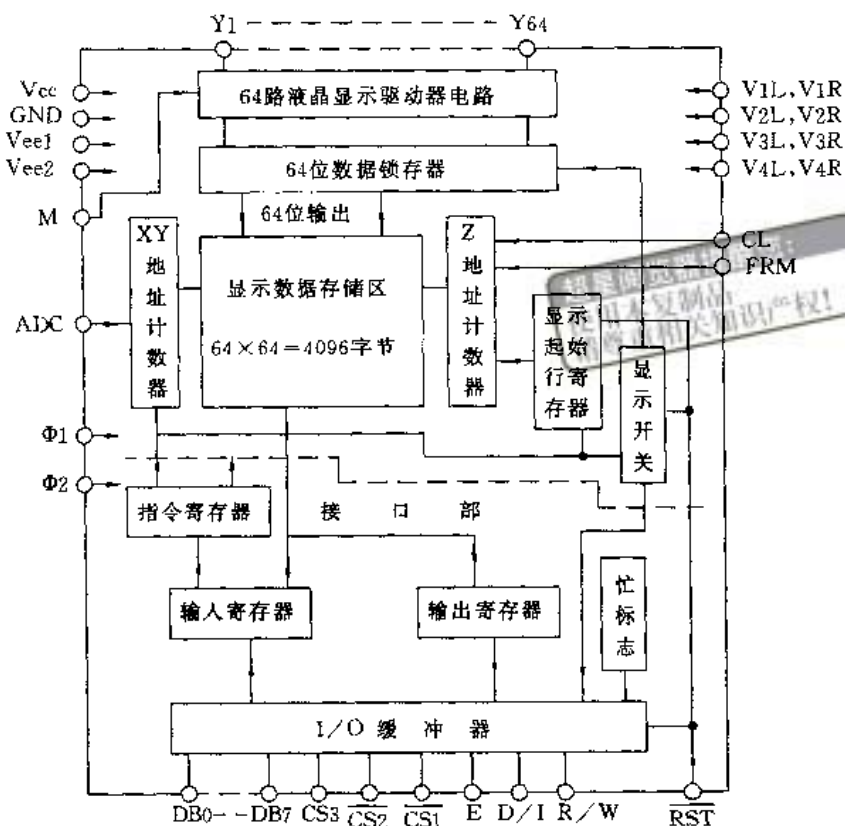


图 13-3 HD61202U 原理框图

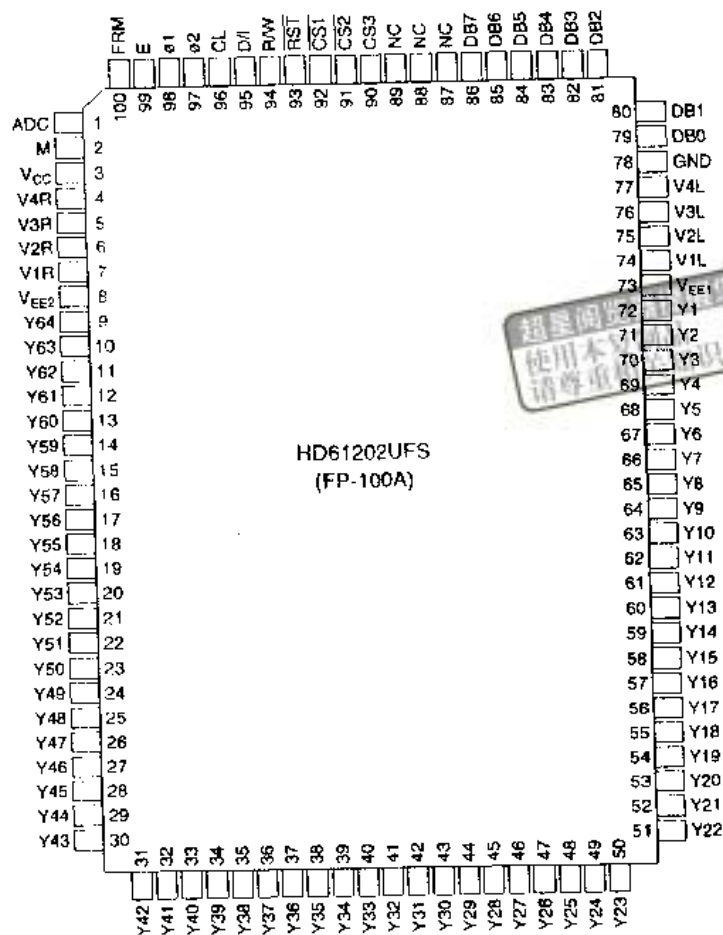
输入端。它们提供了显示存储器操作和指令运行所需的时钟信号。

控制部的主要电路是由显示存储器及其配套的管理电路组成。HD61202U 具有 4096 位显示存储器。其结构是以一个 64×64 位的方阵形式排布的。显示存储器的作用一是存储计算机传送的显示数据，二是作为控制信号源直接控制液晶驱动电路的输出。显示存储器为双端口存储器结构，结构原理示意图如图 13-5 所示。从数据总线侧看有 64 位，按 8 位数据总线长度分成 8 路，称为页面，由 X 地址寄存器控制；每个页面都有 64 个字节，用 Y 地址计数器控制，这一侧是提供给计算机操作的，是双向传输形式。XY 地址计数器选择了计算机所要操作的显示存储器的页面和列地址，从而唯一地确定计算机所要访问的显示存储器单元。从驱动数据传输侧看有 64 位，共 64 行，这一侧是提供给驱动器使用的，仅有输出形式。HD61202U 列驱动器为 64 列驱动输出，正好与显示存储器列向(竖向)单元对应。Z 地址计数器为显示行指针是用来选择当前要传输的数据行。

为了在驱动输出 Y_i 与液晶显示器件列电极连接布线方便，HD61202U 将在显示存储器的 Y 地址计数器设计为双向加一功能。功能可以由 ADC 端电平设置，如图 13-5 所示。当 ADC 接 V_{CC} 时，Y 地址计数器自左向右加一，地址 \$0 对应驱动输出 Y_1 ，\$63 对应驱动输出 Y_{64} ；当 ADC 接 GND 时，Y 地址计数器自右向左加一，地址 \$1 对应驱动输出 Y_{64} ，地址 \$63 对应驱动输出 Y_1 。但是要注意的是尽管地址加一的方向有变化，仅是改变了显示存储器单元地址与驱动输出的对应关系，而显示存储器单元地址与显示屏上点像素的对应关系不变，因此不影响计算机对显示存储器的数据存取的顺序。

在数据总线侧：

X 地址寄存器是一个三位页地址寄存器，其输出控制着显示存储器中 8 个页面的选择，



(Top view)

图 13-4 HD61202U 引脚图

也就是控制着数据传输通道的八选一选择器。X 地址寄存器可以由计算机以指令形式设置。X 地址寄存器没有自动修改功能，所以要想转换页面需要重新设置 X 地址寄存器的内容。

Y 地址计数器是一个 6 位循环加一计数器。它管理某一页面上的 64 个单元。Y 地址计数器可以由计算机以指令形式设置，它和页地址指针结合唯一选通显示存储器的一个单元。Y 地址计数器具有自动加一功能。在显示存储器读/写操作后 Y 地址计数器将自动加一。当计数器加至 3FH 后循环归零再继续加一。

由 XY 地址计数器确定了计算机所需访问的显示存储器单元的地址。计算机在访问显示存储器之前必须要设置 XY 地址计数器。计算机写入或读出显示存储器的数据代表显示屏上某一点列上的垂直 8 点行的数据。D0 代表最上一行的点数据。例如 XY 地址计数器为 0，即 X 地址寄存器为 0，Y 地址计数器为 0，选择显示存储器的 0 页面的第 1 单元。该单元的 8 位数据对应着显示屏上第 1 点列的垂直 8 点(行)，D0 为第一行第一列的点数据，D1 为第二行第一列的点数据，……，D7 为第 8 行第一列的点数据。在此以后，计算机每访问一次显示存储器，Y 地址计数器将自动加一，地址指针将指向下一个单元。

数据输入输出缓冲器是显示存储器显示数据存取的通道，它与接口部的数据总线连接，传输着计算机存取的显示数据。

在驱动数据传输侧：

显示存储器以一行 64 位作为驱动数据总线与显示驱动电路连接的。作为显示驱动信号，

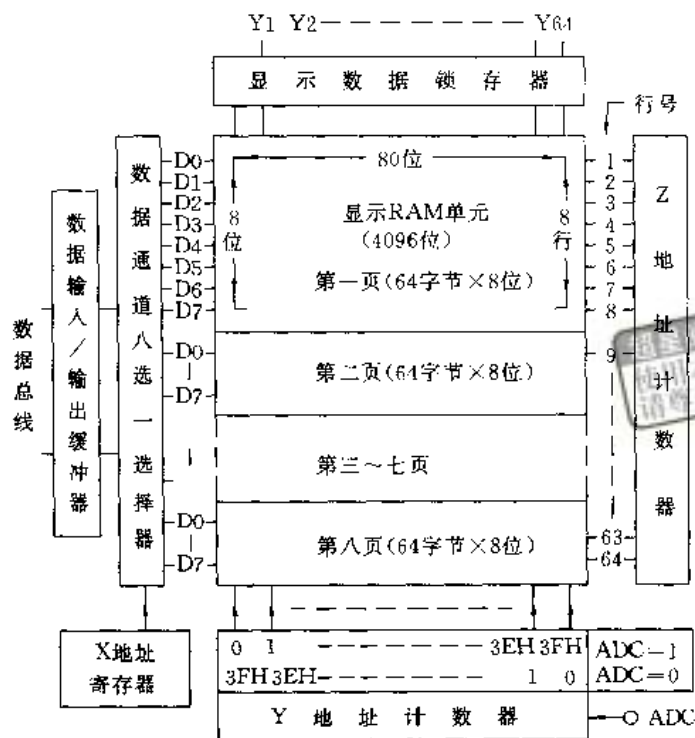


图 13-5 HD61202U 显示存储器的结构原理图

该数据总线上的 64 位数据直接控制驱动电路输出 $Y_1 \sim Y_{64}$ 的输出波形。该路数据总线的管理逻辑电路包括有显示数据锁存器，显示起始行寄存器，Z 地址计数器显示开/关触发器以及外部信号 CL 和 FRM 等。

显示起始行寄存器：6 位寄存器，它规定了显示存储器所对应显示屏上第一行的行号。该行的数据将作为显示屏上第一行显示状态的控制信号。显示起始行寄存器的内容由计算机以指令代码的格式写入。

Z 地址计数器：6 位地址计数器，用于确定当前显示行的扫描地址。Z 地址计数器具有自动加一功能，它与行驱动器的行扫描输出同步，选择相应的列驱动器的数据输出。在行驱动器发来的 CL 信号脉冲的下降沿时加一。在 FRM 信号的高电平时置入显示起始行寄存器的内容，以作为再循环显示的开始。

显示数据锁存器：64 位锁存器。输入来自显示存储器的驱动数据总线，输出接至驱动电路。它是显示驱动数据总线与驱动电路的接口，显示数据锁存器的锁存控制信号为外部信号 CL。在 CL 信号脉冲的上升沿处显示数据锁存器将显示驱动数据总线上的数据锁存至输出端，并保证显示存储器的修改不会影响显示效果。

显示开/关触发器：该触发器的输出一路控制显示数据锁存器的清除端，一路返回到接口控制电路作为状态字中的一位表示当前的显示状态。该触发器的作用就是控制显示驱动输出的电平以控制显示屏的开关。在触发器输出为“关”电平时，显示数据锁存器的输入被封锁并将输出置“0”，从而使显示驱动输出全部为非选择波形，显示屏呈不显示状态。在触发器输出为“开”电平时，显示数据锁存器受 CL 控制，显示驱动输出受显示驱动数据总线上数据控制，显示屏将呈显示状态。显示开/关触发器受逻辑电路控制，计算机可以通过硬件 RST 复位和软件指令“显示开关设置”的写入来设置显示开/关触发器的输出状态。

外部控制信号定义如下：

FRM 输入 显示同步信号(帧信号)。高电平有效。它同步于行扫描信号。在 FRM 脉冲到来时, HD61202U 将 Z 地址计数器复置成显示起始行寄存器的内容。

CL 输入 显示数据的锁存信号。在 CL 上升沿处 HD61202U 将驱动数据总线上内容锁入显示数据锁存器内; 在 CL 下降沿处使 Z 地址计数器加一。

显示存储器驱动数据总线的工作原理: 在外信号 FRM 作用下显示起始行寄存器的内容作为初始值被置入 Z 地址计数器内, 以选通显示存储器中相应的行作为起始行数据并送入驱动数据总线上。在 CL 的作用下, 该数据锁入显示数据锁存器内, 同时 Z 地址计数器加一, 将选通下一行的 64 位数据并送入显示驱动数据总线上, 等待着下一个 CL 的到来。如此循环下去, 直至下一个 FRM 的到来, 重新预置 Z 地址计数器内容, 开始下一帧(周期)扫描的过程。

二、驱动部

液晶显示驱动电路由 64 路列驱动器组成。驱动器的控制信号由显示数据锁存器提供。驱动器的输出为 Y1~Y64; 驱动器的驱动电源由 V_{ee} 和 V1~V4 组成, 控制信号由 M 端提供。它们定义如下:

- Y1~Y64 输出 液晶显示列驱动输出, 与液晶显示器件的列电极连接。
- M 输入 交流驱动波形信号, 由行驱动器 HD61203U 提供。
- V_{ee1}, V_{ee2} 驱动器的驱动电源。
- V1L, V1R 驱动器的驱动电平之一: 选择高电平;
- V3L, V3R 驱动器的驱动电平之一: 非选择高电平;
- V4L, V4R 驱动器的驱动电平之一: 非选择低电平;
- V2L, V2R 驱动器的驱动电平之一: 选择低电平。

驱动器输出 Y1~Y64 的输出电平(V1~V4) 与交流驱动波形信号 M 和显示数据 D 之间的关系如表 13-5 所示。

表 13-5

M	D	Y _i (i=1~64)	含 义
1	1	V1	选择显示高电平
0	1	V2	选择显示低电平
1	0	V3	未选显示高电平
0	0	V4	未选显示低电平

$$V_{cc} \geq V1L = V1R \geq V3L = V3R \geq V4L = V4R \geq V2L = V2R \geq V_{ee1} = V_{ee2}$$

三、接口部

接口部控制电路是 HD61202U 内部寄存器和显示存储器与计算机总线连接的缓冲器电路。它一方面受来自计算机操作信号的控制, 一方面置身于 HD61202U 内部逻辑时序的控制, 完成计算机与 HD61202U 之间的数据传递。接口控制电路由 I/O 缓冲器、输入寄存器、输出寄存器、指令寄存器和状态字寄存器等组成。

I/O 缓冲器为双向三态数据缓冲器。是 HD61202U 内部总线与计算机总线的结合部。其作用是将两个不同时钟下工作的系统连接起来, 实现通讯。I/O 缓冲器在三个片选信号 $\overline{CS1}$, $\overline{CS2}$ 和 CS3 组合有效状态下, I/O 缓冲器开放, 实现 HD61202U 与计算机之间的数据传递。当

片选信号组合为无效状态时，I/O 缓冲器将中断 HD61202U 内部总线与计算机数据总线的联系，对外总线呈高阻状态。因此，HD61202U 不影响计算机的其他数据操作功能。

输入寄存器用于接收在计算机运行速度下传送给 HD61202U 的数据并将其锁存在输入寄存器内，其输出将在 HD61202U 内部工作时钟的运作下将数据写入指令寄存器或显示存储器内。

输出寄存器用于暂存从显示存储器读出的数据，在计算机读操作时，输出寄存器将当前锁存的数据通过 I/O 缓冲器送入计算机数据总线上。

指令寄存器用于接收计算机发来的指令代码，通过译码将指令代码置入相关的寄存器或触发器内。

状态字寄存器是 HD61202U 与计算机通讯时唯一的“握手”信号。状态字寄存器向计算机表示了 HD61202U 当前的工作状态。尤其是状态字中的“忙”标志位是计算机在每次对 HD61202U 访问是必须要读出判别的状态位。当 HD61202U 处于“忙”状态时，I/O 缓冲器被封锁，此时计算机对 HD61202U 的任何操作(除读取状态字操作外)都将是无效的。

接口部设置了一个复位端 $\overline{RST}$ 用于在 HD61202U 上电时或需要时实现硬件电路对 HD61202U 的复位。该复位功能将实现：

- 设置显示状态为关显示状态。 $\overline{RST}$ 从硬件电路上将显示开关触发器置零。
- 显示起始寄存器清零。显示 RAM 第一行对应显示屏上的第一行。
- 在复位期间将状态字中 RESET 位置“1”。

HD61202U 在接口部与计算机连接的信号线有：

DB0~DB7	三态	数据总线。
$\overline{CS1}$ $\overline{CS2}$ $CS3$	输入	片选信号。仅当 $\overline{CS1} = \overline{CS2} = 0$ 且 $CS3 = 1$ 时，HD61202U 才选通。
E	输入	使能信号。
D/I	输入	寄存器选择信号。D/I=0，选择指令寄存器； D/I=1，选择显示存储器。
R/W	输入	读/写选择信号。R/W=1，读操作； R/W=0，写操作。
$\overline{RST}$	输入	复位信号，低有效。复位时，HD61202U 将显示开关和显示起始行寄存器清零。

HD61202U 在接口部设置了三个片选端 $\overline{CS1}$ 、 $\overline{CS2}$ 和 $CS3$ 。这对单片 HD61202U 运行而言是有些多余的了，但在液晶显示模块中多数为多片 HD61202U 组合使用，所以使用三个片选的状态组合来选通其中一片 HD61202U 将使模块的接口简单化。

HD61202U 与计算机的接口时序适配 M6800 系列计算机的时序，控制信号关系如表 13-6 所示。

表 13-6

$\overline{CS1}$	$\overline{CS2}$	$CS3$	D/I	R/W	E	DB7~DB0	功能
X	X	X	X	X	0	高阻	总线释放
0	0	1	0	0	1	输入	写指令代码

(续表)

$\overline{CS1}$	$\overline{CS2}$	CS3	D/I	R/W	E	DB7~DB0	功能
0	0	1	0	1		输出	读状态字
0	0	1	1	0		输入	写显示数据
0	0	1	1	1		输出	读显示数据

计算机对 HD61202U 的操作原理为：计算机要想访问 HD61202U，必须首先读取接口部的状态字寄存器的内容，主要是要判别状态字中的“忙”标志；在“忙”标志表示为“不忙”时，计算机方可访问 HD61202U。在写操作时，HD61202U 在计算机写操作信号的作用下将计算机发来的数据锁存进输入寄存器内，使其转到 HD61202U 内部时钟的控制之下，同时 HD61202U 将 I/O 缓冲器封锁，置“忙”标志位向计算机提供 HD61202U 正在处理计算机发来的数据的信息；HD61202U 根据计算机在写数据时提供的 D/I 状态将输入寄存器的内容送入指令寄存器处理或显示存储器相应的单元，处理完成后，HD61202U 将撤消对 I/O 缓冲器的封锁，同时将“忙”标志位清零，向计算机表示 HD61202U 已准备好接收下一个计算机的操作。在读显示数据时，计算机要有一个操作周期的延时，即“空读”的过程。这是因为在计算机读操作下，HD61202U 向数据总线提供输出寄存器当前的数据，并在读操作结束时将当前地址指针所指的显示存储器单元的数据写入输出寄存器内，同时将列地址计数器加一。也就是说计算机不是直接读取到显示存储器单元，而是读取一个中间寄存器—输出寄存器的数据。而这个数据是上一次读操作后存入到输出寄存器的内容，这个数据可能是上一地址单元的内容，也可能是地址修改前某一单元的内容。因此在计算机设置所要读取的显示存储器地址后，第一次的读操作实际上是要求 HD61202U 将所需的显示存储器单元的数据写入输出寄存器中，供计算机读取。只有从下一次计算机的读操作起，计算机才能读取所需的显示数据。

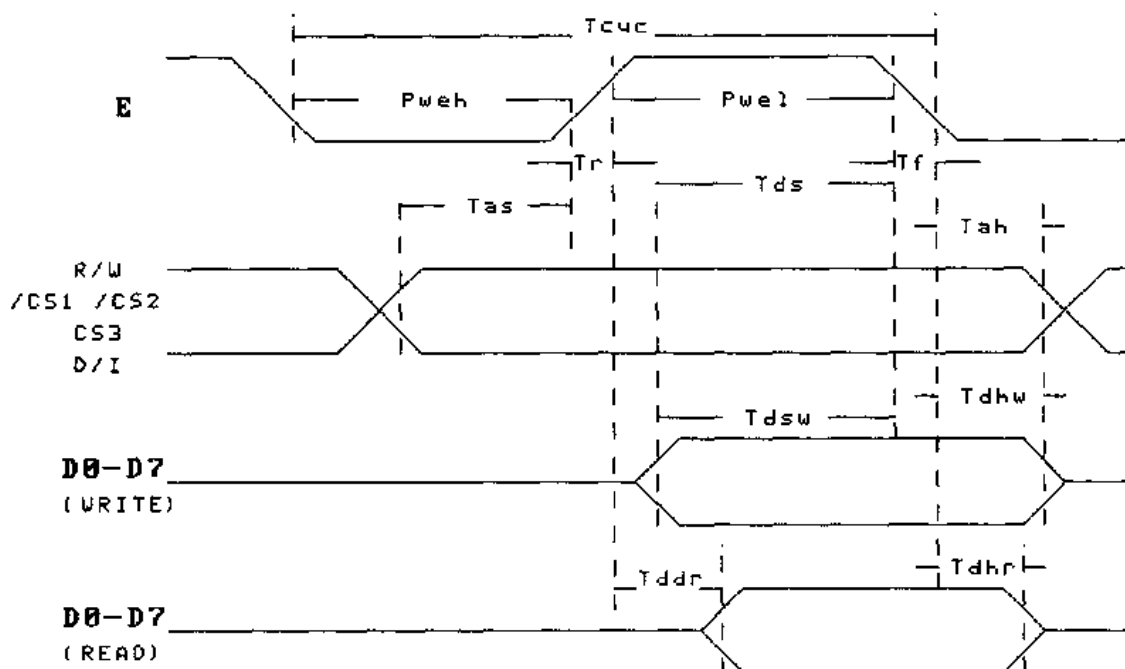


图 13-6 HD61202U 的操作时序图

HD61202U 操作时序图如图 13-6 所示。其电特性如表 13-7 和表 13-8 所示。

表 13-7 $T_a = -20 \sim +75^{\circ}\text{C}$ $\text{GND}=0\text{V}$ $V_{\text{cc}}=2.7 \sim 5.5\text{V}$

项 目	符号	最小值	典型值	最大值	单位
E 周期时间	T_{cyc}	1000	—	—	ns
E 高电平宽度	P_{weh}	450	—	—	ns
E 低电平宽度	P_{wel}	450	—	—	ns
E 上升时间	T_r	—	—	25	ns
E 下降时间	T_f	—	—	25	ns
地址建立时间	T_{as}	140	—	—	ns
地址保持时间	T_{ah}	10	—	—	ns
数据建立时间	T_{dsr}	200	—	—	ns
数据延时时间	T_{ddr}	—	—	320	ns
数据保持时间(写)	T_{dhw}	10	—	—	ns
数据保持时间(读)	T_{dhr}	20	—	—	ns

表 13-8 直流特性 $\text{GND}=0$ $V_{\text{cc}}=2.7 \sim 5.5\text{V}$ $V_{\text{cc}}-V_{\text{ee}}=8 \sim 16\text{V}$ $T_a = -20 \sim +75^{\circ}\text{C}$

项 目	符号	最小值	典型值	最大值	单位	测试条件
输入高电压	V_{ih}	$0.7 \times V_{\text{cc}}$	—	V_{cc}	V	$V_{\text{cc}}=2.7 \sim 4.5\text{V}$
		2.0	—	V_{cc}	V	$V_{\text{cc}}=4.5 \sim 5.5\text{V}$
输入低电压	V_{il}	0.0	—	0.5	V	$V_{\text{cc}}=2.7 \sim 4.5\text{V}$
		0.0	—	0.8	V	$V_{\text{cc}}=4.5 \sim 5.5\text{V}$
输出高电压	V_{oh}	$0.75 \times V_{\text{cc}}$	—	—	V	$V_{\text{cc}}=2.7 \sim 4.5\text{V}$
		2.4	—	—	V	$V_{\text{cc}}=4.5 \sim 5.5\text{V}$
输出低电压	V_{ol}	—	—	$0.2 \times V_{\text{cc}}$	V	$V_{\text{cc}}=2.7 \sim 4.5\text{V}$
		—	—	—	V	$V_{\text{cc}}=4.5 \sim 5.5\text{V}$
输入漏电流	I_{il}	-1.0	—	1.0	μA	V_{i} 从 GND 到 V_{cc}
高阻态输入电流	I_{isl}	-5.0	—	5.0	μA	V_{i} 从 GND 到 V_{cc}
电源电流	I_{cc1}	—	—	100	mA	显示期间
	I_{cc2}	—	—	500	μA	存取周期 1MHz
逻辑电源电压	V_{cc}	-0.3	—	7.0	V	绝对值
驱动电源电压	V_{ee}	$V_{\text{cc}}-17.0$	—	$V_{\text{cc}}+0.3$	V	绝对值
工作温度	T_{opr}	-20	—	+75	$^{\circ}\text{C}$	绝对值
存储温度	T_{stg}	-55	—	+125	$^{\circ}\text{C}$	绝对值
复位时间	T_{rst}	1.0	—	—	μs	

三、模块特性

由 HD61202U 和 HD61203U 组成的液晶显示模块驱动控制系统电路简单, 经济实用。本节以香港精电公司的 MGLS12864 和 MGLS19264 两种液晶显示模块为实例说明这类液晶显示模块的电路特性。

MGLS19264 是 192(列)×64(行)点阵的液晶显示模块,由一片行驱动器 HD61203U 和三片列驱动器 HD61202U 组成驱动控制系统。MGLS19264 的电路原理图如图 13-7 所示。根据前面所叙述的这两种芯片的功能,HD61203U 和 HD61202U 的驱动输出均为 64 路。而且 HD61202U 显示存储器结构为 64×64 方阵,所以在 MGLS19264 中这三个芯片都做到了物尽其用。HD61203U 为主工作方式(M/S=Vcc),通过 RC 振荡电路上电自动工作,扫描占空比为 1/64 (DS1 = GND, DS2 = Vcc);行驱动输出顺序为 X64 到 X1 (SHL = GND);同时向 HD61202U 提供同步时钟信号使其工作;三片 HD61202U 将显示屏 192×64 平均分配,每一片管理 64×64 点阵像素。三片 HD61202U 分别管理左区 64 列(IC1),中区 64 列(IC2)和右区 64 列(IC3)。一个复位电路控制着三片 HD61202U 的上电工作。在片选设置电路中利用了 HD61202U 的三个片选组合出接口上的两个片选端。注意根据电极布线的需要三片 HD61202U 的 ADC 端的设置不同。R1~R5 组成了液晶驱动的偏压电路。

MGLS12864 是 128(列)×64(行)点阵的液晶显示模块,由一片行驱动器 HD61203U 和两片列驱动器 HD61202U 组成驱动控制系统。MGLS12864 的电路原理图同 MGLS19264 在图 13-7 所示电路中,将 IC1 删去即成为 MGLS12864 的电路原理图。

MGLS19264 和 MGLS12864 的接口电路是相同的。接口定义如表 13-9 所示。

表 13-9

序号	符号	状态	功 能
1	$\overline{\text{CSA}}$	输入	片选 A
2	$\overline{\text{CSB}}$	输入	片选 B
3	GND	—	电源地
4	Vcc	—	逻辑电源正
5	V0	—	液晶显示驱动电源
6	D/I	输入	寄存器选择信号
7	R/W	输入	读/写选择信号
8	E	输入	使能信号
9	DB0	三态	数据总线(最低位)
10	DB1	三态	数据总线
11	DB2	三态	数据总线
12	DB3	三态	数据总线
13	DB4	三态	数据总线
14	DB5	三态	数据总线
15	DB6	三态	数据总线
16	DB7	三态	数据总线(最高位)

MGLS19264 和 MGLS12864 接口信号中的两个片选信号的组合定义见表 13-10。

表 13-10

$\overline{\text{CSA}}$	$\overline{\text{CSB}}$	MGLS12864	MGLS19264
0	0	禁止使用	左区
0	1	左区	中区
1	0	右区	右区
1	1	未选	未选

内置 HD61202U 图形液晶显示模块的电特性就是 HD61202U 的电路特性，这里就不再重复了。

第二节 内置 HD61202U 图形液晶显示模块的软件特性

了解内置 HD61202U 图形液晶显示模块的电路特性后，要使用内置 HD61202U 图形液晶显示模块还需要熟悉其软件特性，即 HD61202U 的指令功能，才能很好地应用内置 HD61202U 图形液晶显示模块。HD61202U 的指令功能非常简单，指令一览表如表 13-11 所示。

表 13-11 HD61202U 指令表

指令名称	控制信号		控制代码							
	D/I	R/W	D7	D6	D5	D4	D3	D2	D1	D0
显示开关设置	0	0	0	0	1	1	1	1	1	D
显示起始行设置	0	0	1	1	L5	L4	L3	L2	L1	L0
页面地址设置	0	0	1	0	1	1	1	P2	P1	P0
列地址设置	0	0	0	1	C5	C4	C3	C2	C1	C0
读取状态字	0	1	BUSY	0	ON/OFF	RESET	0	0	0	0
写显示数据	1	0	数 据							
读显示数据	1	1	数 据							

HD61202U 一共有七条指令，从作用上可分为两类。第一条和第二条指令为显示状态设置类；其余指令为数据读/写操作指令。下面详细解释各个指令的功能：

● 读状态字 (Status Read)

格式	BUSY	0	ON/OFF	RESET	0	0	0	0
----	------	---	--------	-------	---	---	---	---

状态字是计算机了解 HD61202U 当前状态，或是 HD61202U 向计算机提供其内部状态的唯一的信息渠道。状态字为一个字节，其中仅有 3 位有效位，它们是：

BUSY 表示当前 HD61202U 接口控制电路运行状态。BUSY=1 表示 HD61202U 正在处理计算机发来的指令或数据。此时接口电路被封锁，不能接受除读状态字以外的任何操作。BUSY=0 表示 HD61202U 接口控制电路已处于“准备好”状态，等待计算机的访问。

ON/OFF 表示当前的显示状态。ON/OFF=1 表示关显示状态，ON/OFF=0 表示开显示状态。

RESET 表示当前 HD61202U 的工作状态，即反映 $\overline{RST}$ 端的电平状态。当 $\overline{RST}$ 为低电平状态时，HD61202U 处于复位工作状态，RESET=1。当 $\overline{RST}$ 为高电平状态时，HD61202U 为正常工作状态，RESET=0。

在指令设置和数据读写时要注意状态字中的 BUSY 标志。只有在 BUSY=0 时，计算机对 HD61202U 的操作才能有效。因此计算机在每次对 HD61202U 操作之前，都要读出状态字判断 BUSY 是否为“0”。若不为“0”，则计算机需要等待，直至 BUSY=0 为止。计算机对 HD61202U 操作的流程图见图 13-8。

● 显示开关设置(Display on/off)

格式	0	0	1	1	1	1	1	D
----	---	---	---	---	---	---	---	---

该指令设置显示开/关触发器的状态,由此控制显示数据锁存器的工作方式,从而控制显示屏上的显示状态。D 位为显示开/关的控制位。当 D=1 为开显示设置,显示数据锁存器正常工作,显示屏上呈现所需的显示效果。此时在状态字中 ON/OFF=0。当 D=0 为关显示设置,显示数据锁存器被置零,显示屏呈不显示状态,但显示存储器并没有被破坏,在状态字中 ON/OFF=1。

● 显示起始行设置(Display Start Line)

格式	1	1	L5	L4	L3	L2	L1	L0
----	---	---	----	----	----	----	----	----

该指令设置了显示起始行寄存器的内容。HD61202U 有 64 行显示的管理能力,该指令中 L5~L0 为显示起始行的地址,取值在 0~3FH(1~64 行)范围内,它规定了显示屏上最顶一行所对应的显示存储器的行地址。如果定时间隔地,等间距地修改(如加一或减一)显示起始行寄存器的内容,则显示屏将呈现显示内容向上或向下平滑滚动的显示效果。

● 页面地址设置[Set Page(X address)]

格式	1	0	1	1	1	P2	P1	P0
----	---	---	---	---	---	----	----	----

该指令设置了页面地址—X 地址寄存器的内容。HD61202U 将显示存储器分成 8 页,指令代码中 P2~P0 就是要确定当前所要选择的页面地址,取值范围为 0~7H,代表第 1~8 页。该指令规定了以后的读/写操作将在哪一个页面上进行。

● 列地址设置(Set Y address)

格式	0	1	C5	C4	C3	C2	C1	C0
----	---	---	----	----	----	----	----	----

该指令设置了 Y 地址计数器的内容,C5~C0=0~3FH(1~64) 代表某一页面上的某一单元地址,随后的一次读或写数据将在这个单元上进行。Y 地址计数器具有自动加一功能,在每一次读/写数据后它将自动加一,所以在连续进行读/写数据时,Y 地址计数器不必每次都设置一次。

页面地址的设置和列地址的设置将显示存储器单元唯一地确定下来,为后来的显示数据的读/写作了地址的选通。

● 写显示数据(Write Display Data)

格式								
			数			据		

该操作将 8 位数据写入先前已确定的显示存储器的单元内。操作完成后列地址计数器自动加一。

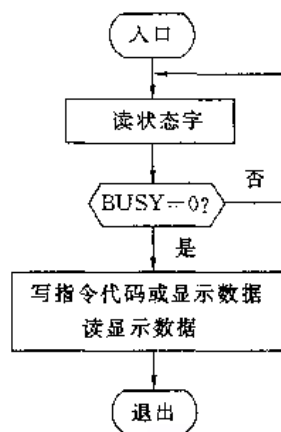


图 13-8 HD61202U 操作流程



● 读显示数据(Read Display Data)

格式		数	据				
----	--	---	---	--	--	--	--

该操作将 HD61202U 接口部的输出寄存器内容读出, 然后列地址计数器自动加一。

第三节 内置 HD61202U 图形液晶显示模块的接口技术

由于 HD61202U 和 HD61203U 组成的驱动控制系统的接口时序适配 M6800 系列的读/写时序, 所以在与 Intel8080 时序的计算机连接时需要有时序的转换。现以国内常用的 8051 系列单片机 8031 为计算机实例, 叙述这类模块的接口技术。

内置 HD61202U 图形液晶显示模块与计算机的连接方式有两种, 一种为直接访问方式, 一种为间接控制方式。下面分别描述这两种方式的实用电路和驱动程序。

一、直接访问方式

直接访问方式就是将液晶显示模块的接口作为存储器或 I/O 设备直接挂在计算机总线上, 计算机以访问存储器或 I/O 设备的方式操作液晶显示模块的工作。直接访问方式的接口实用电路如图 13-9 所示。在图 13-9 中, 计算机 8031 通过高位地址 A11 控制 $\overline{CSB}$; A10 控制 $\overline{CSA}$; 以选通液晶显示屏上各区的控制器 HD61202U; 同时 8031 用地址 A9 作为 R/W 信号控制数据总线的数据流向; 用地址 A8 作为 D/I 信号控制寄存器的选择; E 信号由 8031 的读信号 $\overline{RD}$ 和写信号 $\overline{WR}$ 合成产生。从而实现计算机对内置 HD61202U 图形液晶显示模块的电路连接。电位器用于显示对比度的调节。

以 MGLS19264 为例, 显示屏分左, 中, 右三区, 分别由三片 HD61202U 控制, 在下面提供的驱动子程序中括号内标记的左中右即表示为显示屏三个区的相应的驱动子程序。驱动子程序如下:

COM	EQU	20H	; 指令寄存器
DAT	EQU	21H	; 数据寄存器
CWADD1	EQU	0000H	; 写指令代码地址(左)
CRADD1	EQU	0200H	; 读状态字地址(左)
DWADD1	EQU	0100H	; 写显示数据地址(左)
DRADD1	EQU	0300H	; 读显示数据地址(左)
CWADD2	EQU	0800H	; 写指令代码地址(中)
CRADD2	EQU	0A00H	; 读状态字地址(中)
DWADD2	EQU	0900H	; 写显示数据地址(中)
DRADD2	EQU	0B00H	; 读显示数据地址(中)
CWADD3	EQU	0400H	; 写指令代码地址(右)
CRADD3	EQU	0600H	; 读状态字地址(右)
DWADD3	EQU	0500H	; 写显示数据地址(右)
DRADD3	EQU	0700H	; 读显示数据地址(右)

1. 左区驱动子程序

(1) 写指令代码子程序(左)

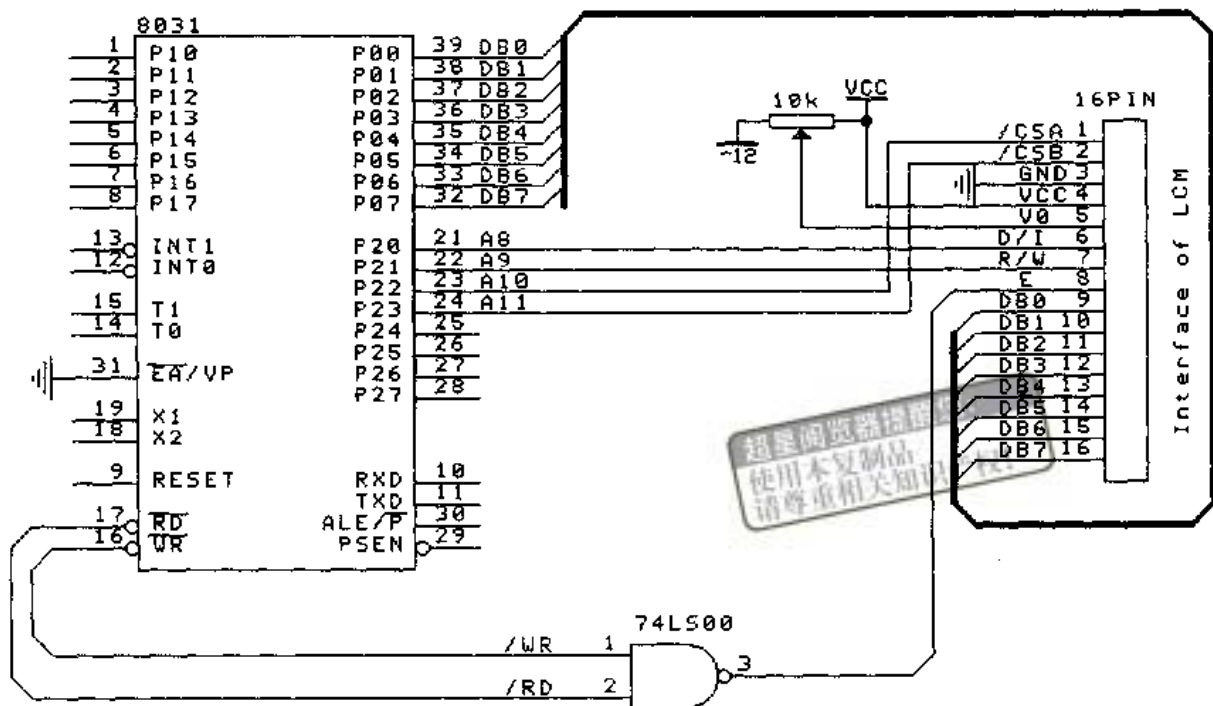


图 13-9 直接访问方式的实用电路

```

PRL0:    PUSH    DPL
          PUSH    DPH
          MOV     DPTR, #CRADD1    ; 设置读状态字地址
PRL01:   MOVX    A, @DPTR          ; 读状态字
          JB      ACC.7, PRL01      ; 判"忙"标志为"0"否, 否再读
          MOV     DPTR, #CWADD1    ; 设置写指令代码地址
          MOV     A, COM            ; 取指令代码
          MOVX    @DPTR, A         ; 写指令代码
          POP     DPH
          POP     DPL
          RET

```

(2) 写显示数据子程序(左)

```

PRL1:    PUSH    DPL
          PUSH    DPH
          MOV     DPTR, #CRADD1    ; 设置读状态字地址
PRL11:   MOVX    A, @DPTR          ; 读状态字
          JB      ACC.7, PRL11      ; 判"忙"标志为"0"否, 否再读
          MOV     DPTR, #DWADD1    ; 设置写显示数据地址
          MOV     A, DAT            ; 取数据
          MOVX    @DPTR, A         ; 写数据
          POP     DPH

```

```

POP      DPL
RET

```

(3) 读显示数据子程序(左)

```

PRL2:      PUSH      DPL
           PUSH      DPH
           MOV       DPTR,#CRADD1    ; 设置读状态字地址
PRL21:     MOVX      A,@DPTR          ; 读状态字
           JB        ACC.7,PRL21     ; 判"忙"标志为"0"否, 否再读
           MOV       DPTR,#DRADD1    ; 设置读显示数据地址
           MOVX      A,@DPTR          ; 读数据
           MOV       DAT,A           ; 存数据
           POP       DPH
           POP       DPL
           RET

```

2. 中区驱动子程序 (MGLS12864 的左区)

(1) 写指令代码子程序(中)

```

PRM0:      PUSH      DPL
           PUSH      DPH
           MOV       DPTR,#CRADD2    ; 设置读状态字地址
PRM01:     MOVX      A,@DPTR          ; 读状态字
           JB        ACC.7,PRM01     ; 判"忙"标志为"0"否, 否再读
           MOV       DPTR,#CWADD2    ; 设置写指令代码地址
           MOV       A,COM            ; 取指令代码
           MOVX      @DPTR,A         ; 写指令代码
           POP       DPH
           POP       DPL
           RET

```

(2) 写显示数据子程序(中)

```

PRM1:      PUSH      DPL
           PUSH      DPH
           MOV       DPTR,#CRADD2    ; 设置读状态字地址
PRM11:     MOVX      A,@DPTR          ; 读状态字
           JB        ACC.7,PRM11     ; 判"忙"标志为"0"否, 否再读
           MOV       DPTR,#DWADD2    ; 设置写显示数据地址
           MOV       A,DAT           ; 取数据
           MOVX      @DPTR,A         ; 写数据
           POP       DPH
           POP       DPL
           RET

```

=====

(3) 读显示数据子程序(中)

```

PRM2:      PUSH    DPL
           PUSH    DPH
           MOV     DPTR, #CRADD2    ; 设置读状态字地址
PRM21:     MOVX    A, @DPTR         ; 读状态字
           JB      ACC. 7, PRM21    ; 判"忙"标志为"0"否, 否再读
           MOV     DPTR, #DRADD2    ; 设置读显示数据地址
           MOVX    A, @DPTR         ; 读数据
           MOV     DAT, A           ; 存数据
           POP     DPH
           POP     DPL
           RET

```



3. 右区驱动子程序 (MGLS12864 的右区)

(1) 写指令代码子程序(右)

```

PRR0:      PUSH    DPL
           PUSH    DPH
           MOV     DPTR, #CRADD3    ; 设置读状态字地址
PRR01:     MOVX    A, @DPTR         ; 读状态字
           JB      ACC. 7, PRR01    ; 判"忙"标志为"0"否, 否再读
           MOV     DPTR, #CWADD3    ; 设置写指令代码地址
           MOV     A, COM            ; 取指令代码
           MOVX    @DPTR, A         ; 写指令代码
           POP     DPH
           POP     DPL
           RET

```

=====

(2) 写显示数据子程序(右)

```

PRR1:      PUSH    DPL
           PUSH    DPH
           MOV     DPTR, #CRADD3    ; 设置读状态字地址
PRR11:     MOVX    A, @DPTR         ; 读状态字
           JB      ACC. 7, PRR11    ; 判"忙"标志为"0"否, 否再读
           MOV     DPTR, #DWADD3    ; 设置写显示数据地址
           MOV     A, DAT            ; 取数据
           MOVX    @DPTR, A         ; 写数据
           POP     DPH
           POP     DPL
           RET

```

=====

(3) 读显示数据子程序(右)


```

PRR2:      PUSH    DPL
           PUSH    DPH
           MOV     DPTR, #CRADD3    ; 设置读状态字地址
PRR21:     MOVX    A, @DPTR         ; 读状态字
           JB      ACC.7, PRR21     ; 判“忙”标志为“0”否, 否再读
           MOV     DPTR, #DRADD3    ; 设置读显示数据地址
           MOVX    A, @DPTR         ; 读数据
           MOV     DAT, A           ; 存数据
           POP     DPH
           POP     DPL
           RET

```

超星阅读器提醒您：
使用本复制品
请尊重相关知识产权！

在 MGLS12864 的应用时, 显示屏被分为左右两个区, 驱动子程序分别为上述的中区和右区的驱动子程序。左区驱动子程序不能使用。

二、间接控制方式

间接控制方式是计算机通过自身的或系统中的并行接口与液晶显示模块连接, 如 8031 的 P1 和 P3 口, 8255 或 Z80--PIO 等并行接口芯片以及像 74LS373 类的锁存器等。计算机通过对这些接口的操作, 以达到对液晶显示模块的控制。这种方式的特点是电路简单, 控制时序由软件实现, 可以实现高速计算机与液晶显示模块的接口。实用电路图如图 13-10 所示。在图中电路中以 8031 的 P1 口作为数据口, P3.0 为 $\overline{\text{CSA}}$, P3.1 为 $\overline{\text{CSB}}$, P3.2 为 D/I, P3.3 为 R/W 和 P3.4 为 E 等信号。电位器用于显示对比度的调节。

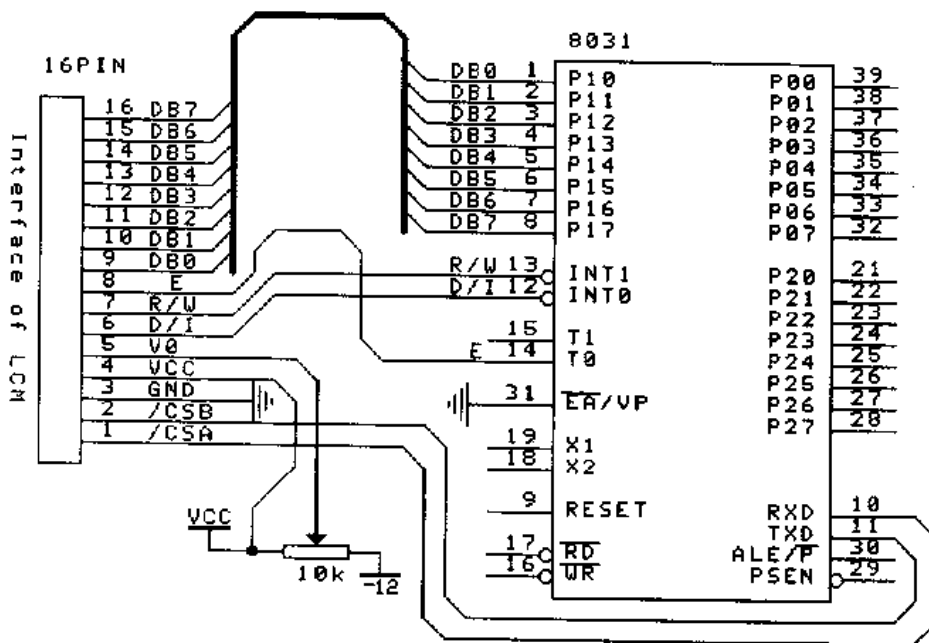


图 13-10 间接控制方式的实用电路

以 MGLS19264 为例, 显示屏分左, 中, 右三区, 分别由三片 HD61202U 控制, 在下面提供的驱动子程序中括号内标记的左中右即表示为显示屏三个区的相应的驱动子程序。驱动子

程序如下:

CSA	EQU	P3.0	; 片选 $\overline{\text{CSA}}$
CSB	EQU	P3.1	; 片选 $\overline{\text{CSB}}$
D/I	EQU	P3.2	; 寄存器选择信号
R/W	EQU	P3.3	; 读/写选择信号
E	EQU	P3.4	; 使能信号

1. 左区驱动子程序

(1) 写指令代码子程序(左)

PRL0:	CLR	CSA	; 片选设置为"00"
	CLR	CSB	;
	CLR	D/I	; D/I=0
	SETB	R/W	; R/W=1
PRL01:	MOV	P1, #0FFH	; P1口置"1"
	SETB	E	; E=1
	MOV	A, P1	; 读状态字
	CLR	E	; E=0
	JB	ACC.7, PRL01	; 判"忙"标志为"0"否, 否再读
	CLR	R/W	; R/W=0
	MOV	P1, COM	; 写指令代码
	SETB	E	; E=1
	CLR	E	; E=0
	RET		

=====

(2) 写显示数据子程序(左)

PRL1:	CLR	CSA	; 片选设置为"00"
	CLR	CSB	;
	CLR	D/I	; D/I=0
	SETB	R/W	; R/W=1
PRL11:	MOV	P1, #0FFH	; P1口置"1"
	SETB	E	; E=1
	MOV	A, P1	; 读状态字
	CLR	E	; E=0
	JB	ACC.7, PRL11	; 判"忙"标志为"0"否, 否再读
	SETB	D/I	; D/I=1
	CLR	R/W	; R/W=0
	MOV	P1, DAT	; 写数据
	SETB	E	; E=1
	CLR	E	; E=0
	RET		

=====

(3) 读显示数据子程序(左)

PRL2:	CLR	CSA	; 片选设置为"00"
	CLR	CSB	;
	CLR	D/I	; D/I=0
	SETB	R/W	; R/W=1
PRL21:	MOV	P1, #0FFH	; P1 口置"1"
	SETB	E	; E=1
	MOV	A, P1	; 读状态字
	CLR	E	; E=0
	JB	ACC. 7, PRL21	; 判"忙"标志为"0"否, 否再读
	SETB	D/I	; D/I=1
	MOV	P1, #0FFH	; P1 口置"1"
	SETB	E	; E=1
	MOV	DAT, P1	; 读数据
	CLR	E	; E=0
	RET		

2. 中区驱动子程序 (MGLS12864 的左区)

(1) 写指令代码子程序(中)

PRM0:	CLR	CSA	; 片选设置为"01"
	SETB	CSB	;
	CLR	D/I	; D/I=0
	SETB	R/W	; R/W=1
PRM01:	MOV	P1, #0FFH	; P1 口置"1"
	SETB	E	; E=1
	MOV	A, P1	; 读状态字
	CLR	E	; E=0
	JB	ACC. 7, PRM01	; 判"忙"标志为"0"否, 否再读
	CLR	R/W	; R/W=0
	MOV	P1, COM	; 写指令代码
	SETB	E	; E=1
	CLR	E	; E=0
	RET		

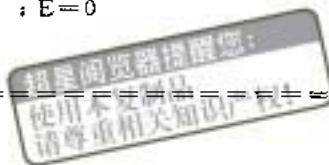
=====

(2) 写显示数据子程序(中)

PRM1:	CLR	CSA	; 片选设置为"01"
	SETB	CSB	;
	CLR	D/I	; D/I=0
	SETB	R/W	; R/W=1
PRM11:	MOV	P1, #0FFH	; P1 口置"1"
	SETB	E	; E=1
	MOV	A, P1	; 读状态字
	CLR	E	; E=0
	JB	ACC. 7, PRM11	; 判"忙"标志为"0"否, 否再读

SETB	D/I	; D/I=1
CLR	R/W	; R/W=0
MOV	P1,DAT	; 写数据
SETB	E	; E=1
CLR	E	; E=0
RET		

=====



(3) 读显示数据子程序(中)

PRM2:	CLR	CSA	; 片选设置为"01"
	SETB	CSB	;
	CLR	D/I	; D/I=0
	SETB	R/W	; R/W=1
PRM21:	MOV	P1,#0FFH	; P1 口置"1"
	SETB	E	; E=1
	MOV	A,P1	; 读状态字
	CLR	E	; E=0
	JB	ACC.7,PRM21	; 判"忙"标志为"0"否, 否再读
	SETB	D/I	; D/I=1
	MOV	P1,#0FFH	; P1 口置"0"
	SETB	E	; E=1
	MOV	DAT,P1	; 读数据
	CLR	E	; E=0
	RET		

3. 右区驱动子程序 (MGLS12864 的右区)

(1) 写指令代码子程序(右)

PRR0:	SETB	CSA	; 片选设置为"10"
	CLR	CSB	;
	CLR	D/I	; D/I=0
	SETB	R/W	; R/W=1
PRR01:	MOV	P1,#0FFH	; P1 口置"1"
	SETB	E	; E=1
	MOV	A,P1	; 读状态字
	CLR	E	; E=0
	JB	ACC.7,PRR01	; 判"忙"标志为"0"否, 否再读
	CLR	R/W	; R/W=0
	MOV	P1,COM	; 写指令代码
	SETB	E	; E=1
	CLR	E	; E=0
	RET		

=====

(2) 写显示数据子程序(右)

```

PRR1:      SETB    CSA           ; 片选设置为"10"
           CLR     CSB           ;
           CLR     D/I          ; D/I=0
           SETB    R/W          ; R/W=1
PRR11:     MOV     P1,#0FFH      ; P1口置"1"
           SETB    E            ; E=1
           MOV     A,P1          ; 读状态字
           CLR     E            ; E=0
           JB      ACC.7,PRR11   ; 判"忙"标志为"0"否, 否再读
           SETB    D/I          ; D/I=1
           CLR     R/W          ; R/W=0
           MOV     P1,DAT        ; 写数据
           SETB    E            ; E=1
           CLR     E            ; E=0
           RET

```

(3) 读显示数据子程序(右)

```

PRR2:      SETB    CSA           ; 片选设置为"10"
           CLR     CSB           ;
           CLR     D/I          ; D/I=0
           SETB    R/W          ; R/W=1
PRR21:     MOV     P1,#0FFH      ; P1口置"1"
           SETB    E            ; E=1
           MOV     A,P1          ; 读状态字
           CLR     E            ; E=0
           JB      ACC.7,PRR21   ; 判"忙"标志为"0"否, 否再读
           SETB    D/I          ; D/I=1
           MOV     P1,#0FFH      ; P1口置"0"
           SETB    E            ; E=1
           MOV     DAT,P1        ; 读数据
           CLR     E            ; E=0
           RET

```

在 MGLS12864 的应用时, 显示屏被分为左右两个区, 驱动子程序分别为上述的中区和右区的驱动子程序。左区驱动子程序不能使用。

第四节 内置 HD61202U 图形液晶显示模块的应用软件

内置 HD61202U 图形液晶显示模块的软件指令非常简单, 所以也就很容易熟悉与应用。本节提供一些实用子程序示例和演示程序, 供使用者参考。

示例程序使用 8031 汇编语言, 接口电路为上一节提供的实用电路, 同时驱动子程序也为上一节所提供。程序是以 MGLS19264 为样机编写, 在 MGLS12864 上使用时将左区的驱动子程序的第一条语句改写为"RET", 将列地址(0~127) 设置在 64~191 范围内就可以应用了。

示例一 初始化子程序

内置 HD61202U 图形液晶显示模块的初始化很简单,就使用两条指令,一条是开显示指令,一条是显示起始行设置指令。程序如下:

```
INT:      MOV      COM,#0C0H      ; 设置显示起始行为第一行
          LCALL    PRL0
          LCALL    PRM0
          LCALL    PRR0
          MOV      COM,#3FH      ; 开显示设置
          LCALL    PRL0
          LCALL    PRM0
          LCALL    PRR0
          RET
```



示例二 清显示 RAM 区(清屏)子程序

```
CLEAR:    MOV      R4,#00H      ; 页面地址暂存器设置
CLEAR1:    MOV      A,R4
          ORL      A,#0B8H      ; "或"页面地址设置代码
          MOV      COM,A        ; 页面地址设置
          LCALL    PRL0
          LCALL    PRM0
          LCALL    PRR0
          MOV      COM,#40H      ; 列地址设置为"0"
          LCALL    PRL0
          LCALL    PRM0
          LCALL    PRR0
          MOV      R3,#40H      ; 一页清 64 个字节
CLEAR2:    MOV      DAT,#00H      ; 显示数据为"0"
          LCALL    PRL1
          LCALL    PRM1
          LCALL    PRR1
          DJNZ     R3,CLEAR2      ; 页内字节清零循环
          INC      R4            ; 页地址暂存器加一
          CJNE     R4,#08H,CLEAR1 ; RAM 区清零循环
          RET
```

示例三 西文字符写入子程序

由于显示屏分三区由三片 HD61202U 控制,所以在程序中要有中间结合部显示字符的处理。程序把显示屏分为 8 个字符行,由页面地址设定;列地址为任意设定。由于 HD61202U 没有字符库,所以在附录 B 第四节中提供了 8×8 点阵的字符字模库。在该字模数据中的第一个字节和第七、八个字节为"0"作为字符间距设置的,该程序可以选择 6×8 点阵字体和 8×8 点阵字体写入。程序的输入寄存器为列地址寄存器,页面地址寄存器和代码寄存器。页面地址寄存器 PAGE 的 D7 位即为字体的选择位。子程序清单如下:

```
COLUMN    EQU      30H          ; 列地址寄存器
```


PAGE	EQU	31H	; 页地址寄存器 D2,D1,D0: 页地址 ; D7: 字符体 D7=0 为 6×8 点阵 ; D7=1 为 8×8 点阵
CODE	EQU	32H	; 字符代码寄存器
COUNT	EQU	33H	; 计数器

CW-PR:	MOV	DPTR, #CTAB	; 确定字符字模块首地址
	MOV	A, CODE	; 取代码
	MOV	B, #08H	; 字模块宽度为 8 个字节
	MUL	AB	; 代码×8
	ADD	A, DPL	; 字符字模块首地址
	MOV	DPL, A	; = 字模块首地址 + 代码×8
	MOV	A, B	
	ADDC	A, DPH	
	MOV	DPH, A	
	MOV	CODE, #00H	; 借用为间址寄存器
	MOV	A, PAGE	; 读页地址寄存器
	JB	ACC. 7, CW-1	; 判字符体
	MOV	COUNT, #06H	; 6×8 点阵
	LJMP	CW-2	
CW-1:	MOV	COUNT, #08H	; 8×8 点阵
CW-2:	ANL	A, #07H	; 取页地址值
	ORL	A, #0B8H	; "或"页地址指令代码
	MOV	COM, A	; 写页地址指针
	LCALL	PRL0	
	LCALL	PRM0	
	LCALL	PRR0	
	MOV	A, COLUMN	; 读列地址寄存器
	CLR	C	
	SUBB	A, #40H	; 列地址-64
	JC	CW-3	; <0 为左屏显示区域
	MOV	COLUMN, A	
	SUBB	A, #40H	; 列地址-64
	JC	CW-21	; <0 为中屏显示区域
	MOV	COLUMN, A	; ≥0 为右屏显示区域
	MOV	A, PAGE	
	SETB	ACC. 5	; 设置区域标志位.
	MOV	PAGE, A	; "00"为左, "01"为中, "10"为右
	LJMP	CW-3	
CW-21:	MOV	A, PAGE	
	SETB	ACC. 4	; 设置区域标志位
	MOV	PAGE, A	
CW-3:	MOV	COM, COLUMN	; 设置列地址值
	ORL	COM, #40H	; "或"列地址指令标志位

	MOV	A,PAGE	; 判区域标志以确定设置哪个控制器
	ANL	A,#30H	
	CJNE	A,#10H,CW_31	; "01"为中区
	LCALL	PRM0	
	LJMP	CW_4	
CW_31:	CJNE	A,#20H,CW_32	; "10"为右区
	LCALL	PRR0	
	LJMP	CW_4	
CW_32:	LCALL	PRL0	; "00"为左区
CW_4:	MOV	A,CODE	; 取间址寄存器值
	MOVC	A,@A+DPTR	; 取字符字模数据
	MOV	DAT,A	; 写数据
	MOV	A,PAGE	; 判区域标志
	ANL	A,#30H	
	CJNE	A,#10H,CW_41	; "01"为中区
	LCALL	PRM1	
	LJMP	CW_5	
CW_41:	CJNE	A,#20H,CW_42	; "10"为右区
	LCALL	PRR1	
	LJMP	CW_5	
CW_42:	LCALL	PRL1	; "00"为左区
CW_5:	INC	CODE	; 间址加一
	INC	COLUMN	; 列地址加一
	MOV	A,COLUMN	; 判列地址是否超出区域范围
	CJNE	A,#40H,CW_6	
CW_6:	JC	CW_9	; 未超出则继续
	MOV	COLUMN,#00H	
	MOV	A,PAGE	; 超出则判在何区域
	JB	ACC.5,CW_9	; 在右区域则退出
	JB	ACC.4,CW_61	; 判在左或中区
	SETB	ACC.4	; 在左区则转中区
	MOV	PAGE,A	
	MOV	COM,#40H	; 设置中区列地址为"0"
	LCALL	PRM0	
	LJMP	CW_9	
CW_61:	SETB	ACC.5	; 在中区则转右区
	CLR	ACC.4	
	MOV	PAGE,A	
	MOV	COM,#40H	; 设置右区列地址为"0"
	LCALL	PRR0	
CW_9:	DJNZ	COUNT,CW_4	; 循环
	RET		

;===== 西文字符库 =====

CTAB; 见附录 B ; 西文字符表

;=====

; 西文显示演示程序段

```
MOV     PAGE, #85H      ; 8×8 点阵字体, 第 6 页
MOV     COLUMN, #30H    ; 起始列为第 48 列
MOV     CODE, #34H      ; 字符代码
LCALL   CW_PR
MOV     PAGE, #85H
MOV     COLUMN, #3CH
MOV     CODE, #45H
LCALL   CW_PR
MOV     PAGE, #85H
MOV     COLUMN, #48H
MOV     CODE, #4CH
LCALL   CW_PR
MOV     PAGE, #85H
MOV     COLUMN, #54H
MOV     CODE, #1AH
LCALL   CW_PR
MOV     R7, #00H
MOV     R6, #60H
LOOP:   MOV     A, R7
        MOV     DPTR, #TAB1
        MOVC    A, @A+DPTR
        MOV     CODE, A
        MOV     PAGE, #05H      ; 6×8 点阵字体, 第 4 页
        MOV     COLUMN, R6
        LCALL   CW_PR
        INC     R7
        MOV     A, #06H
        ADD     A, R6
        MOV     R6, A
        CJNE    R7, #08H, LOOP
        SJMP     $
TAB1:   DB 16H,12H,17H,18H,10H,18H,16H,16H
```



示例四 中文字符写入子程序

汉字显示是国内应用图形液晶显示模块的目的之一。由于 HD61202U 显示存储器的特性,所以不能将计算机内的汉字库提出直接使用,需要将其旋转 90 度后再写入。这里使用了北京清华篷远科贸公司提供的提取汉字软件 PICKHZB.EXE,将所需的汉字从计算机内汉字库提取并旋转 90 度后生成专用的用户字库。其生成字库的格式是前 16 个字节为上半部 16×8 点阵字模数据,后 16 个字节为下半部 16×8 点阵字模数据。该程序仅提供单字节汉字代码寄存器,所以只能建立 256 个汉字库。若要显示更多的汉字就需要使用双字节汉字代码寄存器,这时只需要修改一下程序的前 8 条即可。程序清单如下:

COLUMN	EQU	30H	; 列地址寄存器(0—191)
PAGE	EQU	31H	; 页地址寄存器 D2、D1、D0; 页地址
CODE	EQU	32H	; 字符代码寄存器
COUNT	EQU	33H	; 计数器
CCW_PR:	MOV	DPTR, #CCTAB	; 确定字符字模块首地址
	MOV	A, CODE	; 取代码
	MOV	B, #20H	; 字模块宽度为 32 个字节
	MUL	AB	; 代码×32
	ADD	A, DPL	; 字符字模块首地址
	MOV	DPL, A	; = 字模块首地址 + 代码×32
	MOV	A, B	
	ADDC	A, DPH	
	MOV	DPH, A	
	PUSH	COLUMN	; 列地址入栈
	PUSH	COLUMN	; 列地址入栈
	MOV	CODE, #00H	; 代码寄存器借用为间址寄存器
CCW_1:	MOV	COUNT, #10H	; 计数器设置为 16
	MOV	A, PAGE	; 读页地址寄存器
	ANL	A, #07H	
	ORL	A, #0B8H	; "或"页地址设置代码
	MOV	COM, A	; 写页地址设置指令
	LCALL	PRL0	
	LCALL	PRM0	
	LCALL	PRR0	
	POP	COLUMN	; 取列地址值
	MOV	A, COLUMN	; 读列地址寄存器
	CLR	C	
	SUBB	A, #40H	; 列地址—64
	JC	CCW_2	; <0 为左屏显示区域
	MOV	COLUMN, A	;
	SUBB	A, #40H	; 列地址—64
	JC	CCW_11	; <0 为中屏显示区域
	MOV	COLUMN, A	; ≥0 为右屏显示区域
	MOV	A, PAGE	
	SETB	ACC. 5	; 设置区域标志位.
	MOV	PAGE, A	; "00"为左,"01"为中,"10"为右
	LJMP	CCW_2	
CCW_11:	MOV	A, PAGE	
	SETB	ACC. 4	; 设置区域标志位
	MOV	PAGE, A	
CCW_2:	MOV	COM, COLUMN	; 设置列地址值
	ORL	COM, #40H	; "或"列地址指令标志位
	MOV	A, PAGE	; 判区域标志以确定设置哪个控制器

	ANL	A, #30H	
	CJNE	A, #10H, CCW_31	; "01"为中区
	LCALL	PRM0	;
	LJMP	CCW_4	
CCW_31:	CJNE	A, #20H, CCW_32	; "10"为右区
	LCALL	PRR0	
	LJMP	CCW_4	
CCW_32:	LCALL	PRL0	; "00"为左区
CCW_4:	MOV	A, CODE	; 取间址寄存器值
	MOVC	A, @A+DPTR	; 取汉字字模数据
	MOV	DAT, A	; 写数据
	MOV	A, PAGE	; 判区域标志
	ANL	A, #30H	
	CJNE	A, #10H, CCW_41	; "01"为中区
	LCALL	PRM1	;
	LJMP	CCW_5	
CCW_41:	CJNE	A, #20H, CCW_42	; "10"为右区
	LCALL	PRR1	
	LJMP	CCW_5	
CCW_42:	LCALL	PRL1	; "00"为左区
CCW_5:	INC	CODE	; 间址寄存器加一
	INC	COLUMN	; 列地址寄存器加一
	MOV	A, COLUMN	; 判列地址是否超出区域范围
	CJNE	A, #40H, CCW_6	
CCW_6:	JC	CCW_7	; 未超出则继续
	MOV	COLUMN, #00H	
	MOV	A, PAGE	; 超出则判在何区域
	JB	ACC. 5, CCW_9	; 在右区域则退出
	JB	ACC. 4, CCW_61	; 判在左或中区
	SETB	ACC. 4	; 在左区则转中区
	MOV	PAGE, A	
	MOV	COM, #40H	; 设置中区列地址为"0"
	LCALL	PRM0	
	LJMP	CCW_7	
CCW_61:	SETB	ACC. 5	; 在中区则转右区
	CLR	ACC. 4	
	MOV	PAGE, A	
	MOV	COM, #40H	; 设置右区列地址为"0"
	LCALL	PRR0	
CCW_7:	DJNZ	COUNT, CCW_4	; 当页循环
	MOV	A, PAGE	; 读页地址寄存器
	JB	ACC. 7, CCW_9	; 判完成标志 D7 位, "1"则完成退出
	INC A		; 否则页地址加一
	SETB	ACC. 7	; 置完成位为"1"

本资料为机密文件
 使用本复制品
 请尊重相关知识产权!

```

        ANL      A, #0CFH          ; 清区域标志
        MOV      PAGE, A
        MOV      CODE, #10H       ; 间址寄存器设置为 16
        LJMP     CCW_1            ; 大循环
CCW_9:   RET

; =====
; 中文字符库
CCTAB:   DB 082H,08AH,092H,0B2H,002H,0A7H,092H,05EH; 蓬 90 = 00H
        DB 02AH,0AFH,02AH,05AH,04AH,083H,082H,000H
        DB 000H,080H,040H,03FH,041H,090H,095H,095H
        DB 095H,0FFH,095H,095H,095H,0D0H,040H,000H
        DB 080H,064H,024H,024H,03FH,024H,0A6H,064H; 勃 90 = 01H
        DB 020H,020H,0FFH,020H,020H,0F0H,020H,000H
        DB 000H,008H,049H,089H,07DH,00BH,089H,040H
        DB 030H,00EH,041H,080H,040H,03FH,000H,000H
        DB 040H,042H,044H,0CCH,000H,020H,022H,0E2H; 远 90 = 02H
        DB 022H,022H,0E2H,023H,032H,020H,000H,000H
        DB 000H,040H,020H,01FH,020H,050H,04CH,043H
        DB 040H,040H,047H,048H,048H,06EH,020H,000H
        DB 020H,020H,020H,020H,020H,020H,0A0H,07FH; 大 90 = 03H
        DB 0A0H,020H,020H,020H,020H,030H,020H,000H
        DB 000H,040H,040H,020H,010H,00CH,003H,000H
        DB 001H,006H,008H,010H,020H,060H,020H,000H
; =====
; 中文演示显示程序段
        MOV      PAGE, #02H       ; 第 3 页
        MOV      COLUMN, #35H     ; 第 54 列
        MOV      CODE, #00H      ; 汉字代码
        LCALL    CCW_PR
        MOV      PAGE, #02H       ; 第 3 页
        MOV      COLUMN, #4BH     ; 第 76 列
        MOV      CODE, #01H      ; 汉字代码
        LCALL    CCW_PR
        MOV      PAGE, #02H       ; 第 3 页
        MOV      COLUMN, #63H     ; 第 100 列
        MOV      CODE, #02H      ; 汉字代码
        LCALL    CCW_PR
        MOV      PAGE, #02H       ; 第 3 页
        MOV      COLUMN, #7BH     ; 第 124 列
        MOV      CODE, #03H      ; 汉字代码
        LCALL    CCW_PR
        SJMP     $

```

示例五 滚动演示程序段

定时间隔地有规律地修改显示起始行地址，将产生显示画面平滑的向上或向下滚动。示意程序如下：

```

S_UP:      MOV     R7,#00H           ; 向上滚动程序
S_UP1:     MOV     A,#0C0H           ; 显示起始行设置代码
           ORL     A,R7
           MOV     COM,A
           LCALL   PRL0
           LCALL   PRM0
           LCALL   PRR0
           LCALL   DELAY
           INC     R7
           CJNE    R7,#40H,S_UP1
           MOV     R7,#00H
           LJMP    S_UP1

```

超星阅读器提醒您：
使用本复制品
请尊重相关知识产权！

```

-----
S_DOWN:    MOV     R7,#3FH           ; 向下滚动程序
           MOV     COM,#0C0H         ; 显示起始行设置代码
           LCALL   PRL0
           LCALL   PRM0
           LCALL   PRR0
           LCALL   DELAY
S_DOWN1:   MOV     A,#0C0H           ; 显示起始行设置代码
           ORL     A,R7
           MOV     COM,A
           LCALL   PRL0
           LCALL   PRM0
           LCALL   PRR0
           LCALL   DELAY
           DJNZ    R7,S_DOWN1
           LJMP    S_DOWN

```

```

-----
DELAY:     MOV     R6,#00H           ; 延时子程序
           MOV     R5,#00H
DELAY1:    NOP
           DJNZ    R5,DELAY1
           DJNZ    R6,DELAY1
           RET

```

示例六 绘点子程序

该程序提供了绘图作曲线的基础程序。它可以实现绘点和消点两个功能。在行地址寄存器 LINE 的 D7 位就是用来设置点标志的。D7=1 为绘点即该点被置“1”；D7=0 为消点即该点被清零。在程序中使用了读数据操作，请注意第一个读操作为“空读”。程序清单如下：

```

COLUMN    EQU     30H               ; 列地址寄存器(0—191)

```


LINE	EQU	31H	; 行地址寄存器(0-63) ; D7 为点标志 "1"为置点,"0"为清点

W_DOT:	MOV	A,LINE	; 读行地址寄存器
	ANL	A,#7FH	; 屏蔽点标志位
	MOV	B,#08H	; 8 点行/页
	DIV	AB	; A 为页地址,B 为点位置
	ORL	A,#0B8H	; 设置页面地址
	MOV	COM,A	
	LCALL	PRL0	
	LCALL	PRM0	
	LCALL	PRR0	
	INC	B	; B 作为循环量加一
	SETB	C	; 设置点数据
	CLR	A	; A=0
W_DOT1:	RLC	A	; 数据带进位左移位
	DJNZ	B,W_DOT1	; 循环将点数据移到位
	PUSH	ACC	; 点数据入栈
	MOV	A,COLUMN	; 读列寄存器
	CLR	C	
	SUBB	A,#40H	; 列地址-64
	JC	W_DOT3	; <0 为左屏显示区域
	MOV	COLUMN,A	
	SUBB	A,#40H	; 列地址-64
	JC	W_DOT2	; <0 为中屏显示区域
	MOV	COLUMN,A	; ≥0 为右屏显示区域
	MOV	COM,COLUMN	; 设置列地址值
	ORL	COM,#40H	; "或"列地址指令标志位
	LCALL	PRR0	
	LCALL	PRR2	; "空读"
	LCALL	PRR2	
	MOV	A,LINE	; 取点标志位置入进位位 C
	MOV	C,ACC.7	
	POP	ACC	; 取点数据
	JNC	W_DOT11	; 判点标志为"0"?
	ORL	A,DAT	; 将点数据"1"写入当前显示数据
	LJMP	W_DOT12	
W_DOT11:	CPL	A	; 点数据取反
	ANL	A,DAT	; 将点数据"0"写入当前显示数据
W_DOT12:	MOV	DAT,A	; 存入数据
	MOV	COM,COLUMN	; 设置列地址值
	ORL	COM,#40H	; "或"列地址指令标志位
	LCALL	PRR0	
	LCALL	PRR1	; 写入数据



```

      RET
W_DOT2:  MOV    COM,COLUMN      ; 设置列地址值
        ORL     COM,#40H        ; "或"列地址指令标志位
        LCALL   PRM0
        LCALL   PRM2            ; "空读"
        LCALL   PRM2
        MOV     A,LINE          ; 取点标志位置入进位位 C
        MOV     C,ACC.7
        POP     ACC              ; 取点数据
        JNC     W_DOT21         ; 判点标志为"0"?
        ORL     A,DAT           ; 将点数据"1"写入当前显示数据
        LJMP    W_DOT22
W_DOT21: CPL     A               ; 点数据取反
        ANL     A,DAT           ; 将点数据"0"写入当前显示数据
W_DOT22: MOV     DAT,A           ; 存入数据
        MOV     COM,COLUMN      ; 设置列地址值
        ORL     COM,#40H        ; "或"列地址指令标志位
        LCALL   PRM0
        LCALL   PRM1            ; 写入数据
        RET
W_DOT3:  MOV     COM,COLUMN      ; 设置列地址值
        ORL     COM,#40H        ; "或"列地址指令标志位
        LCALL   PRL0
        LCALL   PRL2            ; "空读"
        LCALL   PRL2
        MOV     A,LINE          ; 取点标志位置入进位位 C
        MOV     C,ACC.7
        POP     ACC              ; 取点数据
        JNC     W_DOT31         ; 判点标志为"0"?
        ORL     A,DAT           ; 将点数据"1"写入当前显示数据
        LJMP    W_DOT32
W_DOT31: CPL     A               ; 点数据取反
        ANL     A,DAT           ; 将点数据"0"写入当前显示数据
W_DOT32: MOV     DAT,A           ; 存入数据
        MOV     COM,COLUMN      ; 设置列地址值
        ORL     COM,#40H        ; "或"列地址指令标志位
        LCALL   PRL0
        LCALL   PRL1            ; 写入数据
        RET
;=====
; 绘点演示程序段
      MOV     R6,#00H           ; 绘点
      MOV     R7,#80H
LOOP1:  MOV     COLUMN,R6        ; 列地址设置

```

```

MOV     LINE,R7           ; 行地址设置
LCALL   W_DOT
INC     R6
INC     R7
CJNE    R7,#0C0H,LOOP1
LOOP2:  DEC     R7
MOV     COLUMN,R6        ; 列地址设置
MOV     LINE,R7          ; 行地址设置
LCALL   W_DOT
INC     R6
CJNE    R7,#80H,LOOP2
LOOP3:  MOV     COLUMN,R6  ; 列地址设置
MOV     LINE,R7          ; 行地址设置
LCALL   W_DOT
INC     R6
INC     R7
CJNE    R7,#0C0H,LOOP3
MOV     R6,#00H          ; 消点
MOV     R7,#00H
LOOP4:  MOV     COLUMN,R6  ; 列地址设置
MOV     LINE,R7          ; 行地址设置
LCALL   W_DOT
INC     R6
INC     R7
CJNE    R7,#040H,LOOP4
LOOP5:  DEC     R7
MOV     COLUMN,R6        ; 列地址设置
MOV     LINE,R7          ; 行地址设置
LCALL   W_DOT
INC     R6
CJNE    R7,#00H,LOOP5
LOOP6:  MOV     COLUMN,R6  ; 列地址设置
MOV     LINE,R7          ; 行地址设置
LCALL   W_DOT
INC     R6
INC     R7
CJNE    R7,#40H,LOOP6
SJMP    $

```

