

51单片机汇编语言教程：9课单片机数据传递类指令

(基于 HJ-1G、HJ-3G 实验板)

单片机数据传递类指令

(1) 以直接地址为目的操作数的指令

MOV direct, A 例: MOV 20H, A

MOV direct, Rn MOV 20H, R1

MOV direct1, direct2 MOV 20H, 30H

MOV direct, @Ri MOV 20H, @R1

MOV direct, #data MOV 20H, #34H

(2) 以间接地址为目的操作数的指令

MOV @Ri, A 例: MOV @R0, A

MOV @Ri, direct MOV @R1, 20H

MOV @Ri, #data MOV @R0, #34H

(3) 十六位数的传递指令

MOV DPTR, #data16

8051是一种8位机，这是唯一的一条16位立即数传递指令，其功能是将一个16位的立即数送入 DPTR 中去。其中高8位送入 DPH，低8位送入 DPL。例：MOV DPTR, #1234H，则执行完了之后 DPH 中的值为12H，DPL 中的值为34H。反之，如果我们分别向 DPH，DPL 送数，则结果也一样。如有下面两条指令：MOV DPH, #35H, MOV DPL, #12H。则就相当于执行了 MOV DPTR, #3512H。

数据传递类指令综合练习：

1 给出每条指令执行后的结果

2 上机练习：

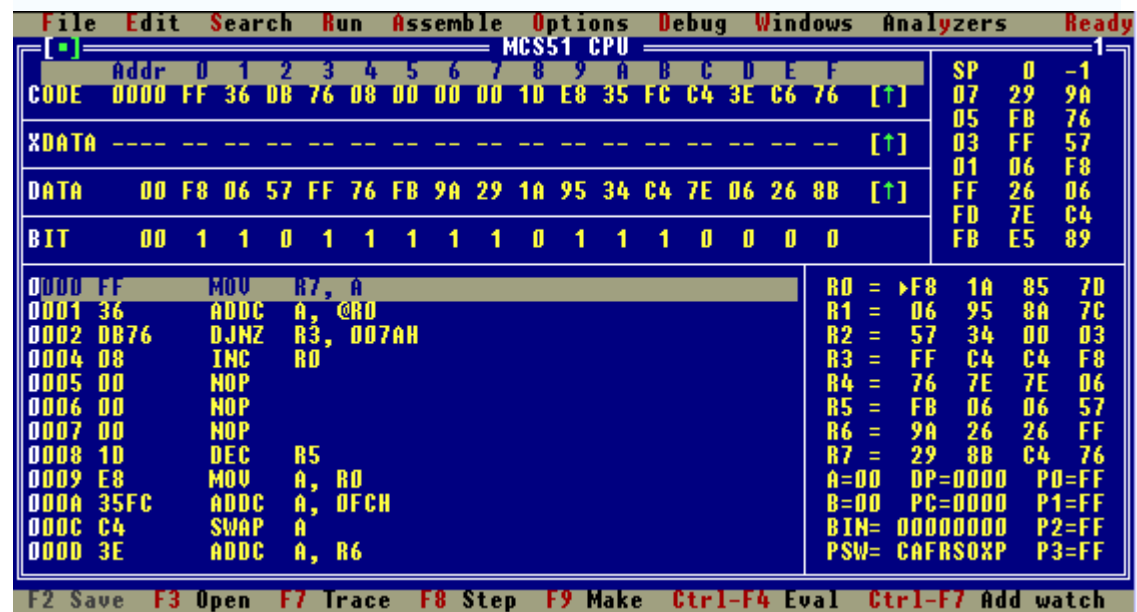
MOV 23H, #30H	(23h)=30h	MOV 45H, 34H	(45H)=34H
MOV 12H, #34H	(12h)=34h	MOV DPTR, #6712H	(DPTR)=6712H
MOV R0, #23H	(R0)=23H	MOV 12H, DPH	(12H)=67H
MOV R7, #22H	(R7)=22H	MOV R0, DPL	(R0)=12H
MOV R1, 12H	(R1)=12H	MOV A, @R0	(A)=67H
MOV A, @R0	(A)=30H		
MOV 34H, @R1	(34H)=34H		

51 单片机汇编语言教程-慧净电子会员收集整理（全部 28 课）

说明：用括号括起来代表内容，如（23H）则代表内部 RAM23H 单元中的值，（A）则代表累加器 A 单元中的值。

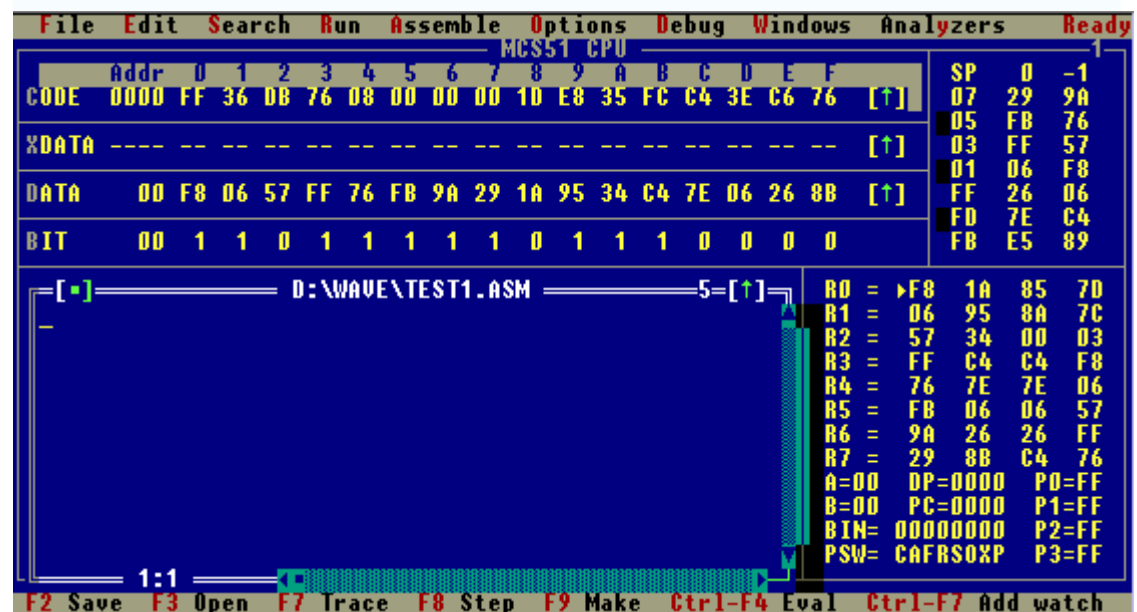
进入 DOS 状态，进入 WAVE 所在的目录，例 D:\WAVE

键入 MCS51，出现如下画面



〈单片机数据传递指令〉图1

按 File→Open，出现对话框后，在 Name 处输入一个文件名（见图2），如果是下面列表中已存在的，则打开这个文件，如果不存在这个文件，则新建一个文件（见图3）



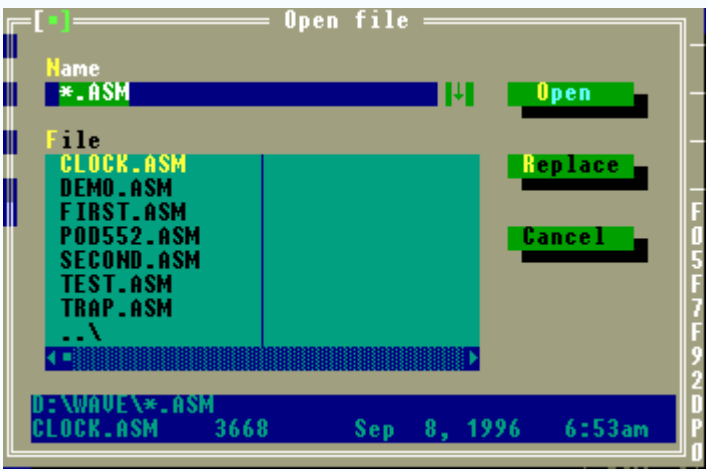


图2

在空白处将上面的程序输入。见图4。用 ALT+A 汇编通过。用 F8 即可单步执行，在执行过程中注意观察屏幕左边的寄存器及 A 累加器中的值的变化。

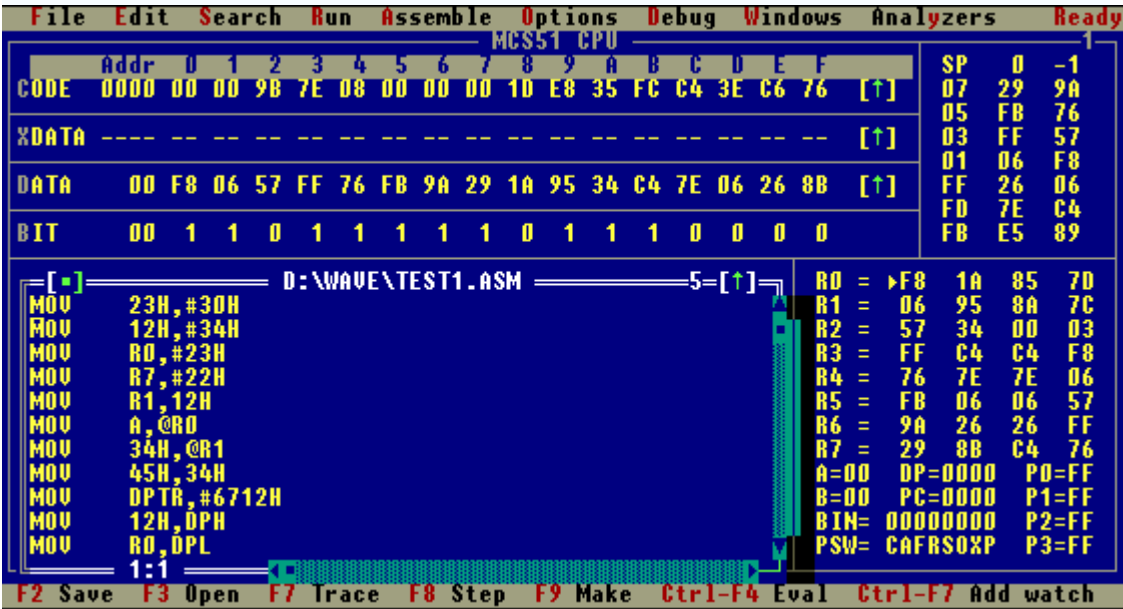


图4

内存中值的变化在此是看不到的，可以用如下方法观察（看图5）：将鼠标移到 DATA，双击，则光标进入此行，此时可以键盘上的上下光标键上下翻动来观察内存值的变化。本行的最前面 DATA 后面的数据代表的是“一段”的开始地址，如现在为20H，再看屏幕的最上方，数字从0到F，显示两者相加就等于真正的地址值，如现在图上所示的内存20H、21H、22H、23H中的值分别是FBH、0EH、E8H、30H。

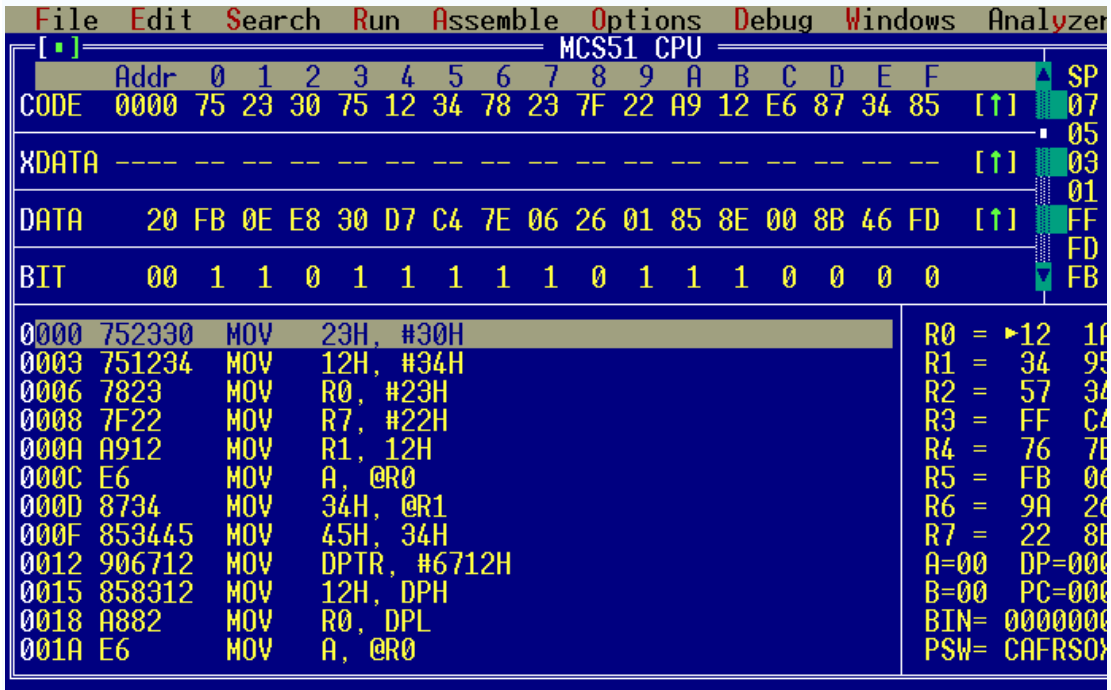


图5

当运行完程序后，即进入它的反汇编区，不是我们想要的东西。为了再从头开始，可以用 CTRL+F2 功能键复位 PC 值。注意此时不会看到原来的窗口，为看到原来的窗口，请用 ALT+4 或 ALT+5 等来切换。当然以上操作也可以菜单进行。CTRL+F2 是程序复位，用 RUN 菜单。窗口用 WINDOWS 菜单。

此次大家就用用熟这个软件吧，说实话，我并不很喜欢它，操作起来不方便，但给我的机器只能上这个，没办法，下次再给网友单独介绍一个好一点的吧。

总结：上面用到的 DOS 软件，各位也不用找了在哪里有下载了，请你用学习板配套光碟里的 KEIL 软件吧，有专门的使用方法，请自己在光碟中查看，当大家学 C 语言时，我们有详细介绍。这一课，大家只要能认识单片机数据传递类指令是什么就可以了。

51 实验板推荐(点击下面的图片可以进入下载资料链接)

