

上海大学 计算机学院

《计算机组成原理实验》报告十二

姓名 赵文宇 学号 23121769
时间 周三 9-11 机位 8 指导教师 刘跃军

实验名称: 建立汇编指令系统

一、实验目的

- 1、建立一个含中文助记符的汇编指令系统。
- 2、用建立的指令系统编制一段程序，并运行之。

二、实验原理

1、**编制汇编指令**:在实验三的“举例”中我们编制了一条微指令，它实现“ $A+w$ 的结果右移一位后的值输出到 OUT”，本质上它是编制了这个微指令对应的 24 个控制信号的电平；实验五的任务 2 中，我们修改机器指令 E8，使其成为“输出 $A+w$ 的结果左移一位后的值到 OUT”指令，它本质上是编制了，EM 中从 E8 开始的连续 4 个地址中的 24bit 值即连续的四条微指令；实验六的背景知识 2 使我们知道：“汇编指令是表达机器指令功能的指令助记符，二者的对应关系由编制的汇编指令表确定”。按步完成这三个编制过程，就定义好一条全新的汇编指令，进一步也可以定义一个汇编指令系统——指令集。

汇编环境 CP226 考虑到了教学上定义汇编指令系统的需求，提供了完成这三个编制任务的集成环境，只要按规定的格式送入编制的符号，系统就会生成相应的汇编指令或汇编指令系统。

2、**汇编表文件**:这个文件的后缀为.DAT，它是一个二维表格式文件，其每一行对应一条指令，这个表共有 3 列，如图 1。第一列是指令的汇编助记符，宽度为 20 个半角字符。第二列是指令的 16 进制编码形式(机器指令)，在实验箱系统就是指令的微程序在，EM 中的起始地址，宽度为 8 个半角字符。第 3 列是这条指令的字节数，宽度为 1 个半角字符，这是本表的重要汇编信息，也是设立本表的原因之一。

这个文件的主要作用是：当编译(汇编)源程序时，查此表把汇编指令翻译成机器指令。即这就是汇编表。

构造这个表文件时也不能带标题行。利用已有.DAT 文件做为模板来构建新指令系统比较方便。具体操作见实验提示。

	20	8	1
ADD A, R0	10	1	1
OUTA	14	1	1

图 1. 汇编表文件格式

3、微程序型指令文件:这个文件的后缀为.MIC, 它也是一个二维表格式的文件, 其每一行对应一条微指令, 这个表共有 11 列(字段), 每列都定义好了属性和宽度, 例如:图 2 是指令集 insfile1.MIC 的格式这个指令集的全部内容见指导书 103 页到 110 页

这个表的主要作用是:当系统调用此文件时把其第 4 列“微程序”的内容送入其第 3 列“微地址”指定的 μEM(微程序存储器)单元。即初始化 μEM。表的第一列为指令的汇编助记符, 内容与表 1 的第 1 列一致。5 到 11 列是对本行微指令的说明, 内容可以省略。

构造这个表文件时不能带标题行。利用已有.MIC 文件做为模板来构建新指令系统比较方便。具体操作见实验提示。

12个英文字符宽	3	3	7	14	19	9	12	10	4	4
ADD A,R?	T2	10	FFF7EF	R?	A		A输出	1		
	T1	11	FFFE90				加运算	1		

图 2. 微程序型指令文件格式

4、指令的机器码文件:这个文件的后缀为.MAC, 也是一个二维表格式文件, 每一行对应一条指令, 表共有 5 列, 如图 3。第 1 列是汇编助记符, 宽度 14, 与表 1 的第 1 列一致。第 2 列是机器码 1, 它是指令的微程序在 EM 中起始地址的二进制表示, 其最后两位是对 R0~R3 的选择, 所以与表 2 的第 3 列一致, 宽度为 15。第 3 列是机器码 2, 是指令带的立即数或存储器地址。第 4 列是机器码 3, 是指令带的第二个存储器地址, 宽度 2。第 5 列是注释, 宽度 100, 用于对指令进行说明。实验箱默认的指令系统 insfile1 没有机器码 3, 其此表的具体内容见指导书 101 页和 102 页。

这个文件的主要作用是:解释汇编表的机器码细节, 所以当编译源程序中的多字节指令时, 可能要查此表。

构造这个表文件时也不能带标题行。利用已有.MAC 文件做为模板来构建新指令系统比较方便。具体操作见实验提示,

14	15	4	2	100
ADD A,R?	000100××10-13			

图 3. 机器码文件格式

三、实验内容

1. 实验任务一 (1、编制一个汇编指令系统，包含下列助记符 2. 用所编制的指令系统，写出源程序，完成 OUT 寄存器交替显示 55, 22, 和 55-22 的值。交替频率为可以清晰辨识即可) (由于两个实验一起验收，所以写在一起了)

指令助记符	指令意义描述
LD A,#*	将立即数打入累加器A
A-W A,#*	累加器A减立即数
跳到 *	无条件跳转指令
OUTA	累加器A输出到OUT
延时	延长显示时间

(1) 实验步骤

先对任务进行分析：

可以翻看系统中已经编好的.mic、.mac、.dat 文件，只需要更改 MOV、JMP、OUT、SUB R?, A 几个指令中的助记符即可，十分的简单。

至此，即可完成该操作。

下面是操作过程：

操作如下：

.DAT 文件修改如下：

LD A,#*	04	2
A-W A,#*	08	2
跳到*	0C	2
OUTA	10	1
延时	14	1

图 4. .DAT 文件修改后

.MAC 文件修改如下：

FATCH_	000000xx
LD A,#*	000001xx
A-W A,#*	000010xx
跳到*	000011xx
OUTA	000100xx
延时	000101xx

图 5. .MAC 文件修改后

.MIC 文件修改如下：

```
[FATCH_ T0 00 CBFFFF
      01 FFFFFFF
      02 FFFFFFF
      03 FFFFFFF
LD A,#*   T1 04 C7FFF7
      T0 05 CBFFFF
      06 FFFFFFF
      07 FFFFFFF
A-W A,#*   T2 08 C7FFEF
      T1 09 FFFE91
      T0 0A CBFFFF
      0B FFFFFFF
跳到*   T1 0C C6FFFF
      T0 0D CBFFFF
      0E FFFFFFF
      0F FFFFFFF
OUTA    T1 10 FFDF9F
      T0 11 CBFFFF
      12 FFFFFFF
      13 FFFFFFF
延时    T3 14 FFFFFFF
      T2 15 FFFFFFF
      T1 16 FFFFFFF
      T0 17 CBFFFF
```

图 6. .MIC 文件修改后

而交替显示 55、22、55-22 的程序也很好编写，只需要将 55 打入 A 寄存器，送到 OUT，延时，将 22 打入 A 寄存器，送到 OUT，延时，用 55-22 打入 A 寄存器，送到 OUT，延时，跳转到开头，整体是一个死循环，即可实现这个程序。具体代码如下所示：

图 7、8. . 详细代码

(2) 实验现象

OUT 寄存器交替显示 55, 22, 和 55-22 的值。

本实验无数据分析与处理，数据记录同实验现象。

(3) 实验结论

可以认为，本次实验正确地完成了编制一个汇编指令系统，包含下列助记符并用所编制的指令系统，写出源程序，完成 OUT 寄存器交替显示 55, 22, 和 55-22 的值。

四、建议和体会

- 1、实验开始前，要先检查一下导线是否功能正常。
- 2、通过自己的尝试与摸索，自己学会了如何进行 CP226 软件上编写汇编语言与实验箱操作的结合。
- 3、依然还是老样子，提前预习的重要性，即在进行实验课前，需要对该节课实验的内容对应的理论知识（虽然好像与机组理论课关系不大）进行温故与复习并且对实验步骤进行大框架上的构想与思考，最好能够提前学会要用到的知识点。

（如果不复习、提前构想的话，仅靠课堂上的那两个小时可能不足以完成所有任务，或者说进展的比较慢，没有什么头绪）。

五、思考题

- 1、为什么汇编指令中可以用“中文符号”？

答：因为中文符号在系统种也是用两个半角符号构成的一个全角符号，和英文符号都可以作为助记符。