

图 2 模型机的数据存储器结构示意图

3. 模型机的内存储器结构

在模型机中，程序和数据共用一个物理存储器（分区使用）。所以，MAROE 和 PCOE 不能同时有效，否则 ABUS 上出现 MAR 和 PC 两个输出的“与”值。其结构如图 3 所示。

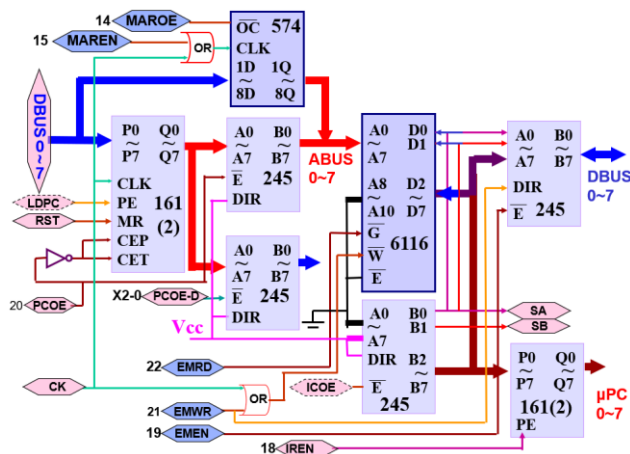


图 3 模型机的内存储器结构示意图

4. 实验箱集成开发环境

人工编译汇编源程序到目标程序的过程即通过查表把汇编指令一对一地翻译成机器指令的简单过程。显然，只要把指令对照表存放在 PC 中，计算机做这种简单工作比人更快、更准确。鉴于 PC 机的强大功能，围绕着程序编译工作，PC 还提供了一系列辅助功能。所有这些功能被打包成一个软件，称为汇编语言程序集成开发环境。由于机器语言是针对 CPU 的，所以这个开发环境也是针对 CPU 的。

本实验箱的汇编语言程序集成开发环境是 CP226 软件。

5. 伪指令

伪指令是不在目标程序中生成机器指令的汇编助记符（汇编命令）。其作用是通知汇编软件如何进行“汇编操作”。不同的汇编系统伪指令不一样。本实验系统有伪指令：END 和 ORG ××。

三、实验内容

1. 实验任务一（手动方式把立即数 33H 写入内存 D1H 单元）

(1) 实验步骤

- ① 打开实验箱电源，检验实验箱的基本功能是否正常。确认无误后，打开实验箱，用导线将 MAROE, MAREN, EMEN, EMRD, EMWR, STEN, X2, X1, X0 接口与开关相连。按三下 TV-ME 按键，将实验箱调整到手动模式。
- ② 设置 K23~K16 为 11010001，并按照下表给出对应的信号：

MAREN	MAROE	EMEN	EMRD	EMWR	STEN	X2	X1	X0
K9	K8	K7	K6	K5	K4	K3	K2	K1
0	1	1	1	1	1	0	0	0

- ③ 按下 STEP 键，D1H 从 IN 寄存器进入 MAR。
- ④ 设置 K23~K16 为 00110011，并按照下表给出对应的信号：

MAREN	MAROE	EMEN	EMRD	EMWR	STEN	X2	X1	X0
K9	K8	K7	K6	K5	K4	K3	K2	K1
1	0	0	1	0	1	0	0	0

- ⑤ 按下 STEP 键，33H 从 IN 寄存器进入 EM。
- ⑥ 实验结束，立即数 33H 已被写入内存 D1H 单元。

(2) 实验现象

D1H 首先由 IN 寄存器进入 MAR，接着 33H 从 IN 寄存器进入 EM，使得 33H 被写入内存 D1H 单元。

(3) 数据记录、分析与处理

MAR 显示 D1H，EM 显示 33H。

(4) 实验结论

成功在手动模式下，把立即数 33H 写入内存 D1H 单元。
完成实验任务一的目的。

2. 实验任务二（手动方式把 D1H 单元的内容读出，再送入 E1H 单元）

(1) 实验步骤

- ① 打开实验箱电源，检验实验箱基本功能。确认无误后，将实验箱调整到手动模式，按照下表连线，并给出对应的信号：

MAREN	MAROE	EMEN	EMRD	EMWR	STEN	X2	X1	X0
K9	K8	K7	K6	K5	K4	K3	K2	K1
1	0	0	0	1	0	1	1	1

- ② 按下 STEP 键，33H 从 EM 进入 ST 堆栈寄存器。

- ③ 设置 K23~K16 为 11100001（选择内存 D1H 单元，D1 的二进制为 11100001），按照下表连线，并给出对应的信号：

MAREN	MAROE	EMEN	EMRD	EMWR	STEN	X2	X1	X0
K9	K8	K7	K6	K5	K4	K3	K2	K1
0	1	1	1	1	1	0	0	0

- ④ 按下 STEP 键，E1H 从 IN 寄存器进入 MAR。

- ⑤ 按照下表连线，并给出对应的信号：

MAREN	MAROE	EMEN	EMRD	EMWR	STEN	X2	X1	X0
K9	K8	K7	K6	K5	K4	K3	K2	K1
1	0	0	1	0	0	0	1	0

- ⑥ 按下 STEP 键，33H 从 ST 堆栈寄存器进入 EM。

- ⑦ 实验结束，D1H 单元的内容已被写入内存 E1H 单元，EM 显示 33H。

(2) 实验现象

33H 首先由 EM 进入 ST 堆栈寄存器，接着 E1H 从 IN 寄存器进入 MAR，最后 33H 从 ST 寄存器进入 EM 中，使得 D1H 单元的内容已被写入内存 E1H 单元。

(3) 数据记录、分析与处理

读取到的数据先暂存在 ST 堆栈寄存器中，然后送入地址，最后写入存储器，分三步进行。

MAR 显示 E1H，EM 显示 33H

(4) 实验结论

成功在手动模式下，把 D1H 单元的内容写入内存 E1H 单元。

完成实验任务二的目的。

3. 实验任务三（在 CP226 汇编语言程序集成开发环境下编写程序，调试和单微步（跟踪）运行，完成下述任务，观察数据走向及寄存器的输入输出状态。将初始值 05H 赋予累加器 A，每次减 1，到 0 时，OUT 寄存器输出 FFH）

(1) 实验步骤

在 CP226 汇编语言程序集成开发环境下编写程序，J1 与 J2 连接，自动状态液晶屏在内部寄存器 μ PC 模式下，编写如下汇编程序：

```
MOV A, #05H
LOOP: SUB A, #01H
      JZ  LOOP1
      JMP LOOP
LOOP1: MOV A, #FFH
      OUT
      END
```

保存文件，文件扩展名为.asm（这一步一定要添加，软件不会自己带上后缀）。
选择端口 COM，编译并下载源程序至实验箱，调试并单步运行（单微指令运行）程序（或者使用键盘上的 F7 按钮），观察并记录实验结果。

(2) 实验现象

随着每一步运行，A 寄存器中的数值由 05H 开始依次减 01H，最后直至 00H。接着发生跳转，A 中显示 FFH，OUT 显示 FFH，程序结束。

(3) 数据记录、分析与处理

汇编程序的具体分析如下所示

MOV A, #05H	将立即数 05H 送至寄存器 A
LOOP: SUB A, #01H	寄存器 A 中内容减去 01H
JZ LOOP1	当遇见零值，就跳转 LOOP1
JMP LOOP	无条件跳转 LOOP
LOOP1: MOV A, #FFH	将 FFH 送至累加器 A
OUT	将寄存器 A 中内容送至 OUT
END	汇编程序终止
	（如果不加程序不会停止）

(4) 实验结论

成功利用 CP226 软件编写汇编程序完成实验任务三，观察了周期下数值的变化类型。

完成实验任务三的目的。

四、建议和体会

体会

通过本次实验，我对计算机内存系统有了更加深入的了解。在前两个实验中，我们是在手动模式下完成任务的。通过连线，选择对应的信号，设置对应的数值，到最后按 STEP 键执行，这样完成任务的方式将一个看似很简单的操作拆分成一个又一个的细节，落实到具体每步选择哪个寄存器，对选中的寄存器该给出什么信号等等，使我们对内存系统的运作模式有了非常清晰地认识。至于第三个实验，我们是通过编写汇编程序实现的。平时经常接触高级语言的我们，通过使用汇编语言，可以深入理解计算机的工作、组成方式，使得程序中的变量在我们眼里不再只是他们在高级语言中对应的变量名，而是在寄存器中被实实在在存储的一个数据。编写汇编程序首先需要熟悉模型机对应的指令集，然后即可进行程序设计。

由于疫情的影响，本次实验是居家完成。通过观看线上的实验讲解视频与学习实验内容 PPT，并未能亲身操作实验箱和 CP226 软件来进行实验，这会一定程度上影响学习效果，因此需要课前高质量的预习与课后的总结以加深理解。并且如果有机会，在返回学校进入实验室后，应该重新实验以验证自己实验报告中的内容。

五、思考题

既然有 **ORG** 伪指令，为什么集成开发环境下载到实验箱的目标程序的第一条执行语句最好存放在 **0** 号地址？

答：由计算机组成的原理可知，程序是按照地址顺序自动执行的，因此一条指令执行时必然有可能会受到来自地址位置在其之前的指令的运行结果所带来的影响。虽然 **ORG** 伪指令可以实现规定程序的起始地址，但是相比之下，若是将第一条执行语句存放在 **0** 号地址，其之前没有任何指令，不存在受任何其之前的指令影响的风险。所以在条件允许的情况下，第一条执行语句最好存放在 **0** 号地址，以避免其他因素影响而产生错误。