

# 上海大学 计算机学院

## 《计算机组成原理实验》报告三

姓名 赵文宇 学号 23121769  
时间 周二 9-11 机位 14 指导教师 刘跃军

---

实验名称: 微指令系统实验

### 一、实验目的

- 1、读出系统已有的微指令，并理解其含义。
- 2、设计并实现微指令系统。

### 二、实验原理

#### ①译码器

前两次实验的每一项操作都是通过人工设置电键  $k_i$  的编码，然后给出一个 CK 脉冲来完成。如果有一个器件能给出对应与每个操作的  $k_i$  编码，这个器件就可以代替我们来为操作译码，从而产生控制信号，该器件就是指令译码器，也叫控制器。

常见的控制器有两种：用组合逻辑电路实现控制信号的逻辑电路结构和用存储器实现控制信号的微程序结构。前者将在《在系统编程技术及应用》课程中学 习。这里学习后者。

#### ②微指令

把一个操作的控制总线的编码放在一个存储单元中，同时给出调用这个单元的方法（例如：这个单元的地址），则对使用者而言，这个调用方法等价于控制总线的编码本身，二者都称为微指令。

例如：在实验二的举例中有控制总线具体连线为下表所示：

表 1. 控制总线连线

控制电键	k9	k8	k7	k6	k4	k3	k2	k1	k0
被控对象	OUTEN	X2	X1	X0	S2	S1	S0	WEN	AEN

#### ③操作序列的形式化表述：

根据以上叙述：我们同样可以将操作“BDH 送入 W 寄存器”的  $k_i$  编码 1111 11101 存入随意选取的 FDH 地址单元；将操作“A-W”的  $k_i$  编码 1111 00111 存入随意选取的 B2H 地址单元；将操作“直通门 D 的内容送 OUT 寄存器”的编码 0100 111111 存入 DCH 地址单元。这样，我们在实验二举例中的操作序列可以用微指令的形式符号表述为：



图 1. 操作序列形式符号化

#### ④实验箱的微指令系统

实验箱生产商的工程师根据这个实验箱的功能、部件数量、必须的基本操作等要求，给它安排了 24 条控制线——控制总线宽度为 24。由于每条微指令有 24 位，相应地存放微指令的微程序存储器的每个地址单元也必须是 24 位。于是，生产商把 3 片 8 位存储器的对应地址并接在一起，构成一个 24 位的存储器。具体连接见下图 2。于是，每选中一个地址，就有 24 位控制信号送上控制总线。

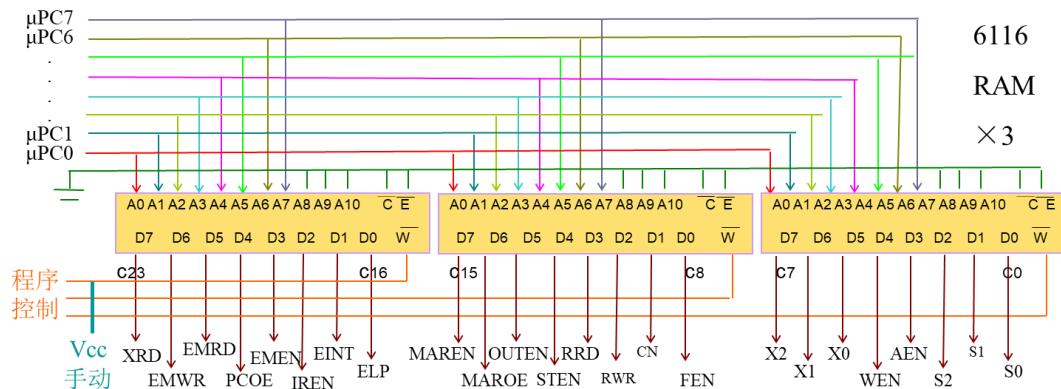


图 2. 用于存储微指令的 24 位存储器

#### ⑤ 在微程序存储器中的微指令

表 2. 在微程序存储器中的微指令

序号	微指令	微地址	序号	微指令	微地址	序号	微指令	微地址
1	7fbff7	91	13	ef7f7f	bc	25	ffffcd7	d4
2	b7bf9f	89	14	fcfff5f	ec	26	ffffe90	11
3	c6ffff	a4	15	feff3f	b9	27	ffffe91	31
4	c77fff	94	16	feff5f	cc	28	ffffe92	66
5	c7fbff	8c	17	ff77ff	84	29	ffffe93	5d
6	c7ffef	2c	18	ff9f9f	95	30	ffffe94	21
7	c7ffff7	7c	19	ffd9f9f	c4	31	ffffe95	4a
8	cbffff	0	20	ffef7f	bd	32	ffffe96	e4
9	cbffff	3e	21	fff7ef	60	33	ffffeb7	d8
10	d6bfff	be	22	fff7f7	70	34	ffffed7	dc
11	d7bfef	65	23	ffffb9f	80	35	fffff17	c0
12	d7bff7	75	24	ffffcb7	d0	36	fffffff	1

#### ⑥ 实验箱微程序存储器的微地址控制

实验箱微程序存储器的微地址由两片可预置初值的四位二进制计数器构成，称作微指令计数器  $\mu$  PC，连线如图 2。其工作方式为：

当 CEP 和 CET=00 时，计数器保持不变。

PE(IREN)=0 时，在 CK 的上升沿，把 P0~P3 引脚的值打入计数器。由于上边 161 的 P0 和 P1 接地，所以打入的初始微地址最后两位总是 00。

当 PE =1 时，在 CK 上升沿，上边 161 进行加一计数。当它加到 1111 值时，其 TC 为 1，使下片 161 的 CEP 和 CET 为 1，下片进入计数有效状态。计数为其他值时 TC 为 0。

当上边的 161 为 1111 值时，下一个 CK 脉冲，使上片的计数值变为 0000，下片加 1，然后上片的 TC=0。IBUS 的值由程序存储器输出，进而由 PC 值决定，不能用电键输入

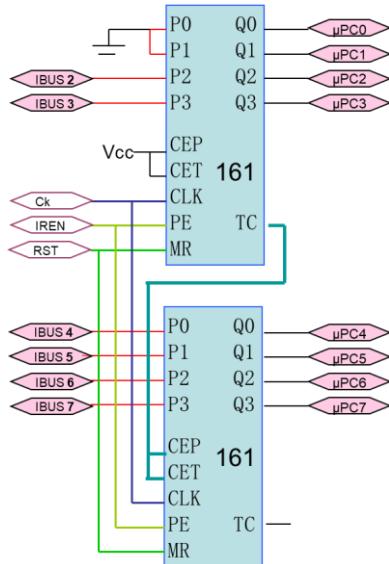


图 3. 四位二进制计数器

## ⑦ 实验箱中小键盘的使用

四个主菜单，用 TV/ME 键切换

1. 观察和修改内部寄存器；
2. 观察和修改程序存储器；
3. 观察和修改微程序存储器；
4. 手动状态。

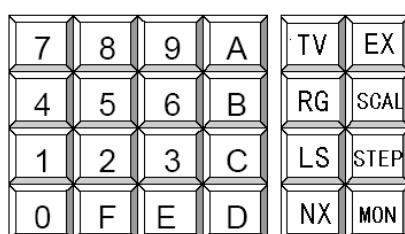


图 4. 实验箱小键盘

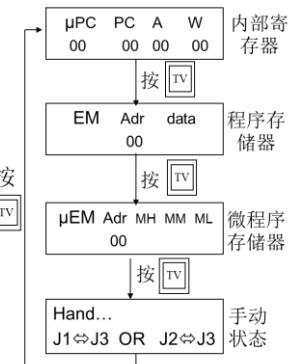


图 5. 切换示意

## ⑧ $\mu$ EM（微程序存储器）状态的操作

进入  $\mu$  EM 状态后，液晶屏显示如下图，闪动光标在 Adr 下，通过实验箱上

的小键盘送入  $\mu$ EM 的微地址后，闪动光标跳到 MH 下，输入微指令的高字节。然后，闪动光标跳到 MM 下，输入微指令的中字节。接着，闪动光标跳到 ML 下，输入微指令的低字节。至此，一条微指令就被输入到了微程序存储器的指定微地址单元。

若在送入  $\mu$ EM 的微地址后，按 NX 键，则液晶屏显示该微地址单元中已存贮的微指令值，但这时并不把这个值送上控制总线。即控制总线无改变。再按 NX 键就显示下一个微地址单元的微指令值。同样，按 LS 键显示上一个微地址单元的微指令值。

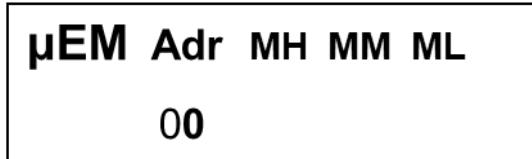


图 6.  $\mu$ EM 状态示意

### 三、实验内容

#### 1. 实验任务一

(观察微指令存储器地址为 31H 单元的内容；分析其控制功能；验证该功能是否实现 编制一条微指令实现“A 非”运算后右移一位的值送 OUT；把这条微指令放入微程序存储器的 32H 单元；验证它的功能是否实现。(假设 A=33H, W=11H, 两题连起来做，故写在一个报告的任务中)

##### (1) 实验步骤

因为本次实验不需要用到导线，所以不需要对导线能否正常工作正常检验。

对于实验一的部分，先按两下 TV/ME 键进入微程序存储器模式，可以看到目前显示屏上“Adr”一栏数值为 00 并有光标跳动，在小键盘上输入 31，可以看到 MH、MM、ML 对应变成了 EF、FE、91，与表 2（在微程序存储器中的微指令）进行比对，发现完全一致，因为在后续按 STEP 键进行的操作中，第一次按 STEP，会对 Adr:31 的前一个微程序进行执行，所以为了避免 Adr:30 的影响，按下 F1/LS 键，使 Adr 栏显示 30，按 FF FF FF 使得 MH、MM、ML 栏上显示数字都变为 FF FF FF，这样以来对 Adr:30 进行执行操作也不会影响最终结果。再按一次 F2/NX 键，检查 Adr:31 的 MH、MM、ML 栏是否正确设置；再按一次 F1/LS 键，检查 Adr:30 的 MH、MM、ML 栏是否正确设置。至此，实验一的部分结束。

在对实验二进行操作前，需要先自己推出“将 A 非运算后移一位的值送 OUT”操作对应的六位十六进制微指令(如下所示)

- (1) “A 非”运算对应的控制总线编码为：c<sub>2</sub>c<sub>1</sub>c<sub>0</sub>=110；
- (2) 标志位默认设置为 0，故 c<sub>8</sub>=0；
- (3) “右移一位的值送 OUT”对应的控制总线编码为：c<sub>7</sub>c<sub>6</sub>c<sub>5</sub>=101；

(4) “数据总线值送 OUT 寄存器”对应的控制总线编码为: c13=0。  
这四个操作的微指令编码: c23~c0=1111 1111 1101 1110 1011  
 $1110=(FFDEBE)_{\text{H}}$

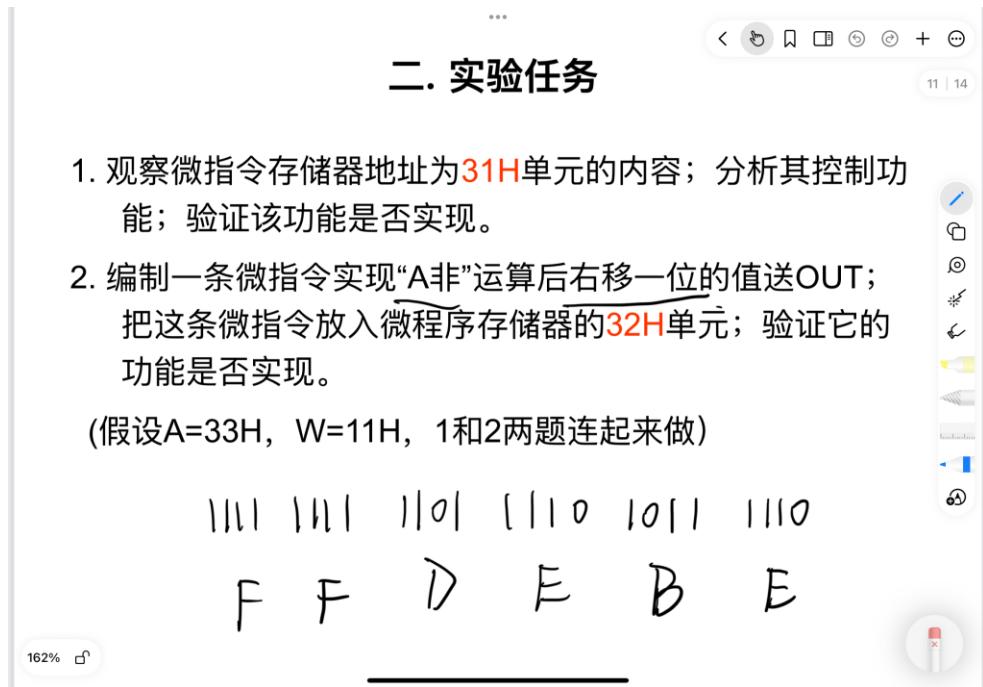


图 7. 微指令推理

推理得到六位十六进制微指令:  $(FFDEBE)_{\text{H}}$  后, 开始进行实际操作, 因为此时还留在微程序存储器模式, 所以可以直接在 Adr 一栏输入 32, 再用小键盘输入 FF DE BE, 观察显示屏上对应数值是否正确, 确认无误后即可按两次 TV/ME 键, 进入  $\mu$  PC 内部寄存器。

通过小键盘分别输入 31 00 33 11, 对应着  $\mu$  PC、A、W 的值。连续按下四次 STEP 键, 观察 ALU 上各寄存器及 OUT 显示屏上的数字。

### (2) 实验现象

在按下第一次 STEP 键后, ALU 上各寄存器及 OUT 显示屏上数字没有任何变化, 此时 A 寄存器为 33, W 寄存器为 11;

在按下第二次 STEP 键后, D 寄存器上数字变为 22, A 寄存器上数字变为 22, W 寄存器上数字不变;

在按下第三次 STEP 键后, D 寄存器上数字变为 DD, R 寄存器上数字变为 6E, R 寄存器旁边红灯亮了 (选中了 R 寄存器);

在按下第四次 STEP 键后, OUT 显示屏上数字变为了 6E。

### (3) 数据记录、分析与处理



图 8. Adr 为 31 时，内置微程序

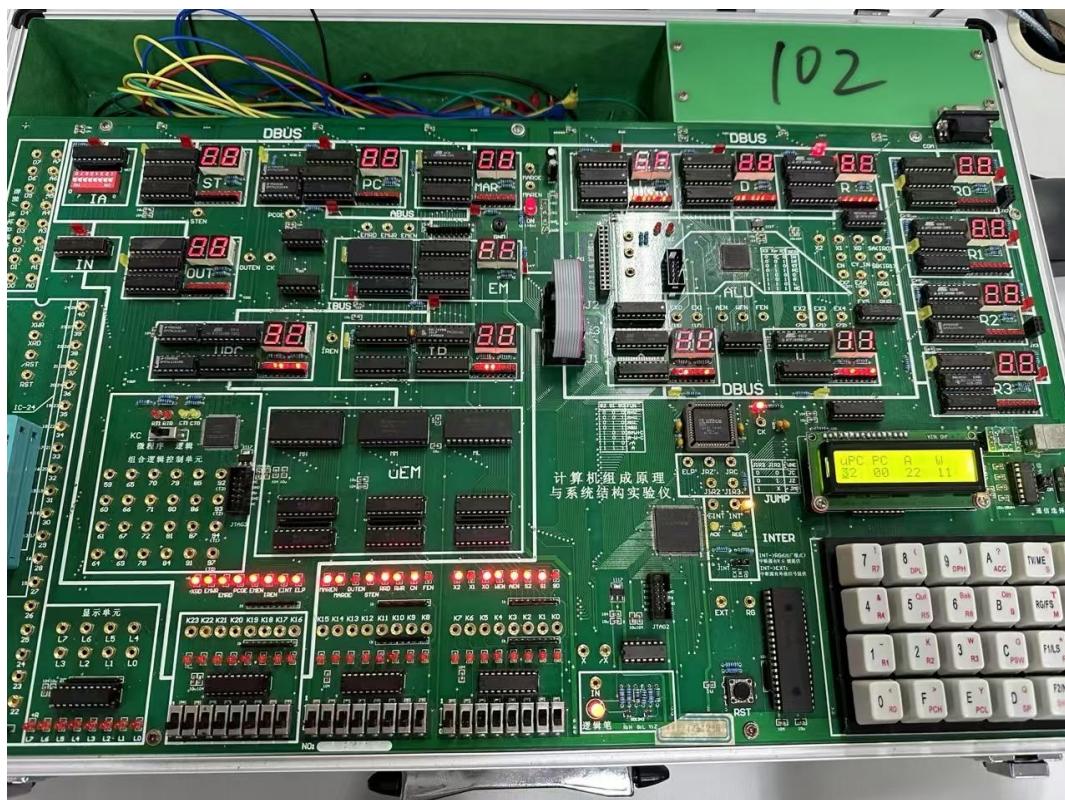


图 9. 按下两次 STEP 键后实验箱情况

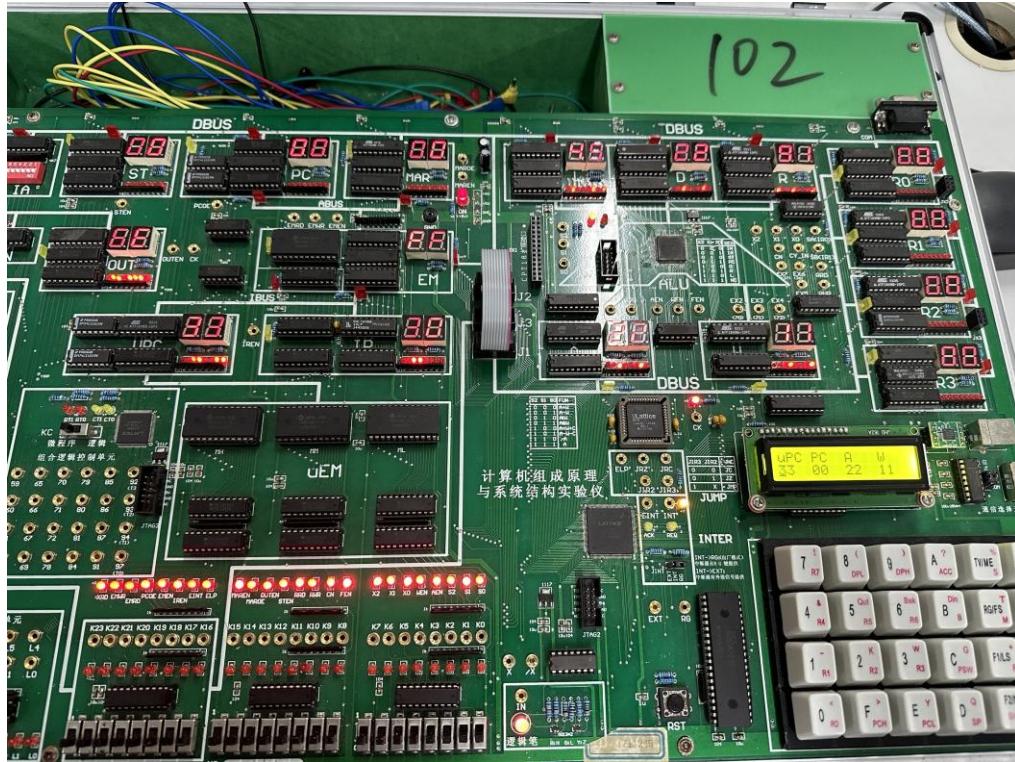


图 10. 按下四次 STEP 键后实验箱情况

#### (4) 实验结论

微指令存储器地址为 31H 单元的控制功能为“计算 A-W，并将结束送入 A 寄存器中”；

正确推出了用于实现用微指令实现“ $A$  非运算后右移一位的值送  $OUT$ ”的操作， $(FFDEBE)_H$ ，并最终计算得出了 6E 的结果。

#### 四、建议和体会

1、用小键盘操作感觉要比连线方便不少，至少不会因为某根导线无法正常工作或者某一根导线错连而导致需要重新来过=-=。

2、吸取了前一次的教训，提前预习，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)。

这次我先来了实验室一会儿，自己研究了下如何进行实验的操作，果然正式做实验时进展都非常迅速且顺利。

3、TV/ME 真的很重要，到目前三节课，每节课都要频繁的用到！

## 五、思考题

Q: 如何给  $\mu$  PC 置初值?

**A:** 按下 RST 键进行初始化/关闭开启电源后，利用小键盘直接对 μPC 进行置初值操作即可。

(实验报告上传： 超星教学平台作业, 请注意上传时间截止日期, 一般为一个星期。请及时上传, 晚交会影响报告成绩。)