# 实验六 微程序控制器实验

## 一、实验目的

1. 掌握微程序控制器的组成原理和工作过程。
2. 理解微指令和微程序的概念，理解微指令与指令的区别与联系。
3. 掌握指令操作码与控制存储器中微程序的对应方法，熟悉根据指令操作码从控制存储器中读出微程序的过程。

## 预习要求

1. 读懂实验电路图；
2. 预习实验电路图，熟悉实验元器件的功能特性和使用方法。

## 三、实验要求

1. 做好实验预习，读懂实验电路图，熟悉实验元器件的功能特性和使用方法；
2. 按照实验内容与步骤的要求进行实验，遇到问题冷静、独立思考，认真仔细地完成实验；
3. 完成实验报告。

## 四、实验内容

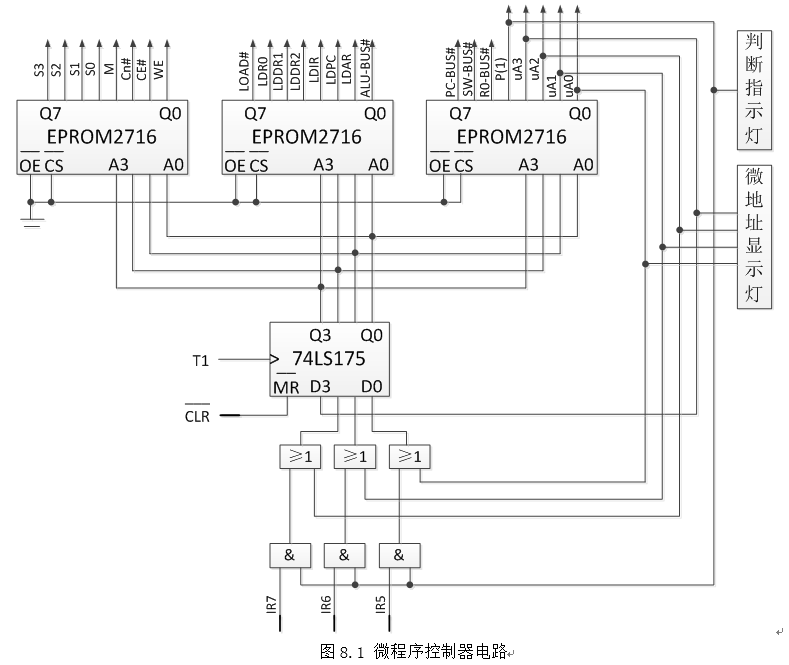
### 实验电路图

本实验使用的主要元器件有：4位数据锁存器74LS175，2K×8 EPROM2716，时序发生器，或门、与门、开关、LED等。芯片详细说明请见附录。

图 8.1为本实验总框图，其中3片EPROM2716构成控制存储器，1片74LS175为微地址寄存器，与74LS175数据输入引脚相连的输入信号线及6个门电路构成了地址转移逻辑。注意，2716输出信号中带后缀“#”的信号为低电平有效信号，不带后缀“#”的信号为高电平有效信号。为简化电路结构，本实验没有使用微命令寄存器，并且在虚拟实验系统中，将3片EPROM2716组合为一个虚拟EPROM组件。本实验使用的EPROM和时序发生器一样，均为虚拟实验系统提供的虚拟组件。

实验电路中涉及的主要控制信号如下：

1. /CE：2716的片选信号。为0时2716正常工作，实验中将其接地，恒置为0；
2. /OE：2716读信号。为0时2716正常工作，/CE = 0，/OE = 0时为读操作，实验中将其接地，恒置为0；
3. /CLR：芯片74LS175的清零信号，低电平有效。
4. TI：微地址加载信号，在T1的上升沿将微地址锁存到74LS175。
5. IR5~IR7：指令操作码的输入信号，这几条信号线本应与指令寄存器的输入引脚相连，但在本实验中，与数据开关相连，指令操作码通过数据开关手动设置。

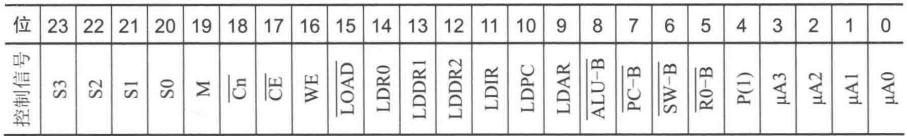
图8.1 微程序控制器电路

### 实验原理

在存储逻辑型计算机中，一条机器指令对应了一个微程序，不同的机器指令对应了不同的微程序，执行一条指令其实就是运行其对应的一个微程序，微程序由微指令组成，是微指令的有序集合。微程序是在设计一台计算机时就预先设计好并且固化在只读存储器中的，以后每当要执行某条指令时，只需找到并运行其对应的微程序。

控制存储器专门用于存放微程序，在本实验中，控制存储器是EPROM2716×3（2K×24位），其中地址输入引脚为A10~A0，实验中仅用到A3~A0，高7位地址线接地，实际存储容量为16×3字节。Q0~Q23这24个输出引脚与24的微指令相对应。

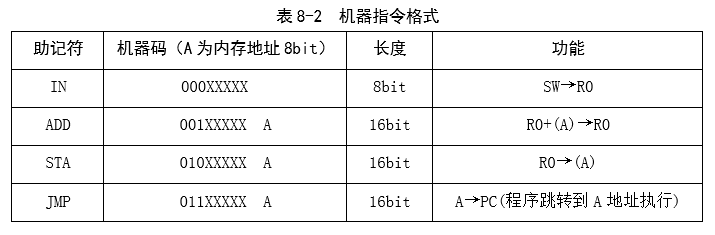
微指令格式如表8-1所示，采用全水平型，字长24位，其中操作控制字段19位，全部采用直接表示法，不使用译码器，每一位表示一个微命令，用于发出全机的操作控制信号；顺序控制字段5位，包括后续微地址μA3~μA0和判别位P1，用于决定下一条微指令的地址。

表8-1 微指令格式

地址转移逻辑电路用于产生下一条微指令的地址，主要由两级与门、或门构成。地址转移逻辑需要用到的数据信号有：后续微地址μA3~μA0、判别位P1、指令操作码IR5~IR7。当P1=0时，下一条微指令的地址即为后续微地址μA3~μA0；当P1=1时，下一条微指令的地址由指令操作码IR5~IR7决定，一般是将操作码进行简单变换，把变换后的值作为下一条微指令的地址，此地址就是该操作码对应的微程序的入口地址。

微地址寄存器74LS175为控制存储器提供微指令地址。当/CLR=0时，微地址寄存器清零，从控制存储器00H地址开始执行微程序，地址转移逻辑生成下一条微指令的地址。每当T1上升沿到来时，新的微指令地址会打入微地址寄存器74LS175，控制存储器随机输出这条微指令，地址转移逻辑继而生成下一条微指令的地址。如果时序信号连续发生，微指令也会按一定的顺序接连输出。本实验用到四条机器指令，指令格式如表8-2所示。

表8-2 机器指令格式

上述四条指令的微程序流程设计如图8.2所示，其中一个方框对应一条微指令，方框右上角的数字为八进制表示的微地址，其中，每条微指令都按照表8-1的格式编写了二进制代码，并预存在控制寄存器芯片EPROM2716×3中。一个方框也表示一个CPU周期，执行一条微指令需要一个CPU周期。四条指令对应四个微程序，每个微程序包括N条微指令，需要执行N个CPU周期。其中部分微指令二进制代码如表8-3所示。

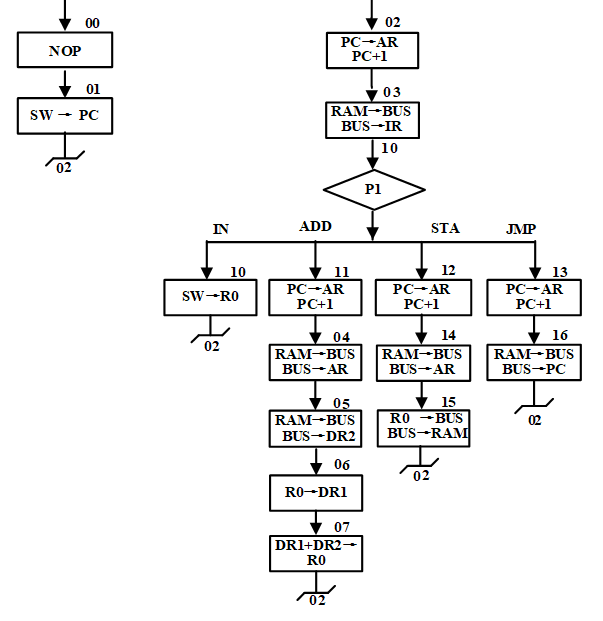
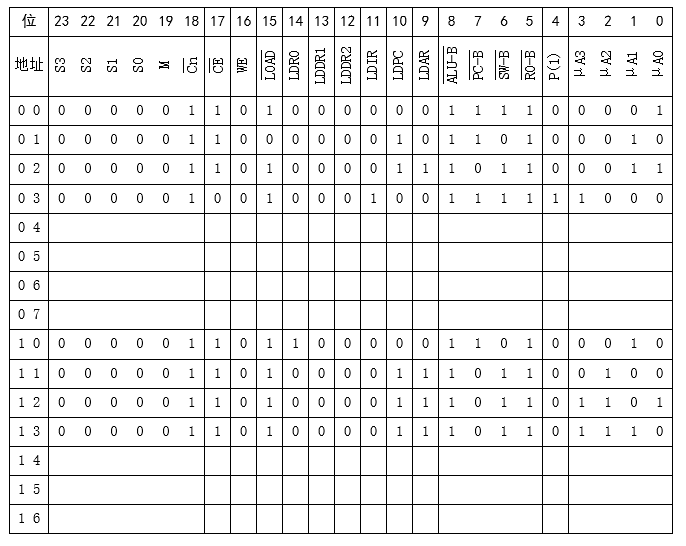


图8.2微程序流程图

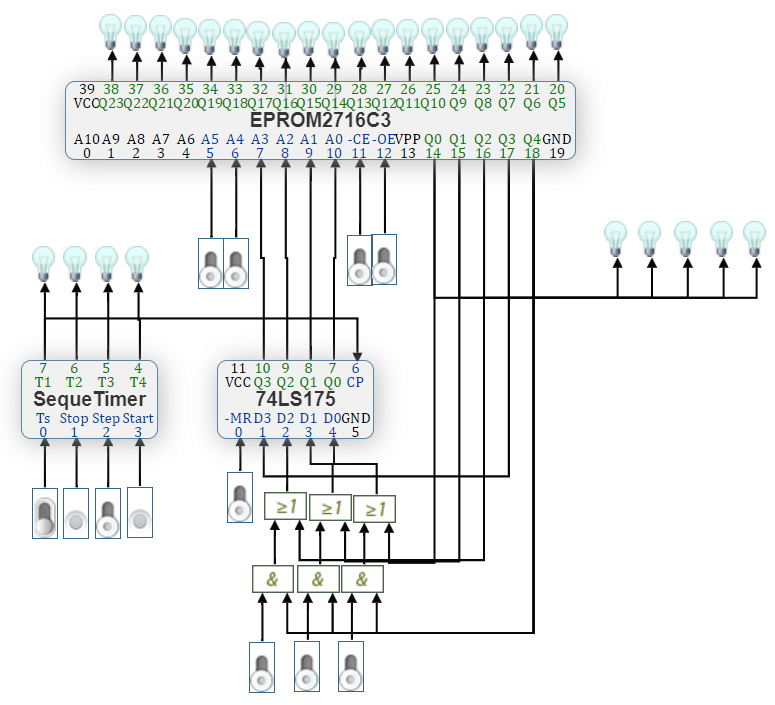
表8-3 微程序二进制代码表



### 实验内容与步骤

1. 运行虚拟实验系统，按照图8.1绘制实验电路图：

（请在此处粘贴电路截图）



1. 电路预设值：将/CE、/OE、A4、A5置0，/CLR 置0，Step置1.
2. 打开电源。此时由于/CLR=0，微地址寄存器清零，给出微程序入口地址00H，控制寄存器随之输出第00号微指令。
3. 将/CLR 置1。
4. 将IR7~IR5均设置为0，思考并回答：若此时连续不断地发出时序信号，微程序真的执行流程是怎样的？请按顺序写出前10条微指令的地址：

00->01->02->03->10->02->03->10->02->03, 顺序循环执行。

前10条微指令的地址 00 01 02 03 10 02 03 10 02 03

1. 连续点击Start，观察微指令的输出顺序，检验控制寄存器输出的微指令是否与表8-3中一致，验证上一步预测的顺序是否正确。
2. 将IR7~IR5设置为不同组合，用单步方式分别独处ADD、STA和JMP三条指令的微程序，用后续微地址和判别指示灯跟踪微程序执行及转移情况，将表8-3中缺少的微程序代码补充完整。

|  |  |
| --- | --- |
| 地址 | 位：23~~0 |
| 04 | 000001 001000001 1111 0 0101 |
| 05 | 000001 001001000 1111 0 0110 |
| 06 | 000001 101010000 1110 0 0111 |
| 07 | 100101 101100000 0111 0 0010 |
| 14 | 000001 001000001 1111 0 1101 |
| 15 | 000001 011000000 1110 0 0010 |
| 16 | 000001 000000010 1111 0 0010 |

1. 思考并回答：若不改变控制器实验电路，IN、ADD、STA和JMP四条指令的微程序在控制存储器中的存放位置是否可以随机安排？有什么限制？为什么？

不可以随机安排；控制存储器是只读存储器，在不改变控制器实验电路的情况下，微程序在控制存储器中存放的位置是固定的，不能随机安排；因为微程序是微指令的有序集合，所以四条指令的微程序在控制存储器中的位置必须是顺序结果，也就是说其一系列微指令有序集合在控制存储器中是有序存储不能随机安排。

### 思考与分析

1. 微程序控制器主要由哪些部件组成？各部件的功能是什么？

（1）控制存储器：用于存放各指令对应的微程序，由只读存储器构成

（2）微指令寄存器：用于存放取出的微指令，它的位数和微指令字长相等

（3）微地址形成部件：用于产生初始微地址和后继微地址，以保证微指令的连续执行

（4）微地址寄存器：接收微地址形成部件送来的微地址，为在读取微指令做准备。

1. 本实验中，地址转移逻辑电路是怎么样利用判别测试字段（P字段）实现微程序分支的？

通过判别测试字段P和指令操作码IR5~IR7决定，进而修改微地址寄存器的内容。只有当P为1的时候指令操作码才能够通过线路影响到下一个指令的地址，否则指令操作码IR5~IR7经过与门后都会变为0，无法起作用。当P1=0时，下一条微指令的地址即为后续微地址μA3~μA0；当P1=1时，下一条微指令的地址由指令操作码IR5~IR7决定。

1. 如果把微程序控制器看作一个黑盒子，那么它的输入信号有哪些？这些信号是哪些部件提供给它的？它的输出信号有哪些？这些信号是发送给哪些部件的？

输入信号：微程序的地址；信号由IR和PSW提供的。

输出信号：微命令；信号发给存储器和运算器。