北京邮电大学
网络空间安全学院

《计算机组成与系统结构》实验报告四
微程序控制器实验

姓　　名 郭航江

学　　号 _____

班　　级 _____

任课教师 _____

2023 年 12 月

一、 实验目的

(1) 掌握微程序控制器的原理

(2) 掌握 TEC-8 模型计算机中微程序控制器的实现方法，尤其是微地址转移逻辑的实现方法。

(3) 理解条件转移对计算机的重要性。

二、 实验内容

1. 微指令格式

根据机器指令功能、格式和数据通路所需的控制信号，TEC-8 采用如图 2.5 所示的微指令格式。微指令字长 40 位，顺序字段 11 位(判别字段 P4~P0，后继微地址 NμA5~NμA0)，控 制字段 29 位，微命令直接控制。

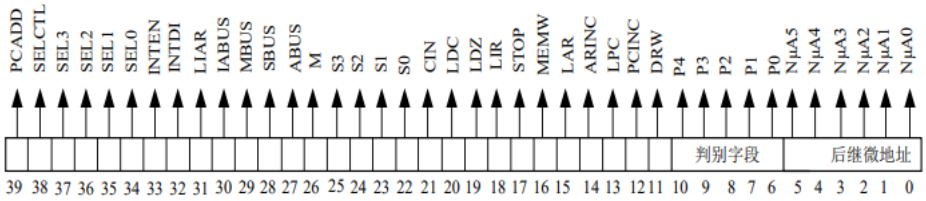


图 2.5 微指令格式

前面的 3 个实验已经介绍了主要的微命令(控制信号)，介绍过的微命令不再重述，这里 介绍后继微地址、判别字段和其它的微命令。

NμA5~NμA0	下址，在微指令顺序执行的情况下，它是下一条微指令的地址
P0	=时，根据后继微地址 NμA5~NμA0 和模式开关 SWC、SWB、SWA 确定下一条微指令的地址。见图 2.6 微程序流程图
P1	=1 时，根据后继微地址 NμA5~NμA0 和指令操作码 IR7~IR4 确定下一条微指令的地址。见图 2.6 微程序流程图。
P2	=1 时，根据后继微地址 NμA5~NμA0 和进位 C 确定下一条微指令的地址。见图 2.6 微程序流程图。
P3	=1 时，根据后继微地址 NμA5~NμA0 和结果为 0 标志 Z 确定下一条微指令的地址。见图 2.6 微程序流程图。
P4	=1 时，根据后继微地址 NμA5~NμA0 和中断信号 INT 确定下一条微指令的地址。模型计算机中，中断信号 INT 由时序发生器在接到中断请求信号后产生。
STOP	=1 时，在 T3 结束后时序发生器停止输出节拍脉冲 T1、T2、T3。
LIAR	=1 时，在 T3 的上升沿，将 PC7~PC0 写入中断地址寄存器 IAR。
INTDI	=1 时，置允许中断标志(在时序发生器中)为 0，禁止 TEC-8 模型计算机响应中断请求
INTEN	=1 时，置允许中断标志(在时序发生器中)为 1，允许 TEC-8 模型计算机响应中断请求
IABUS	=1 时，将中断地址寄存器中的地址送数据总线 DBUS。
PCADD	=1 时，将当前的 PC 值加上相对转移量，生成新的 PC。

2. 微程序流程图

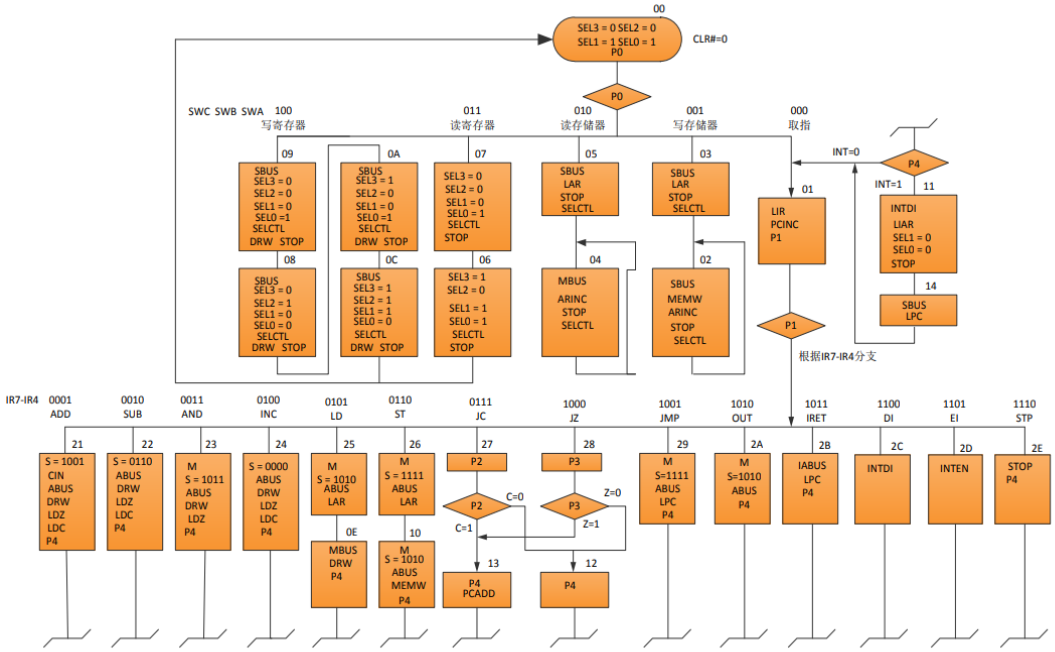


图 2.6 TEC-8 模型计算机微程序流程图

3. 控制电路

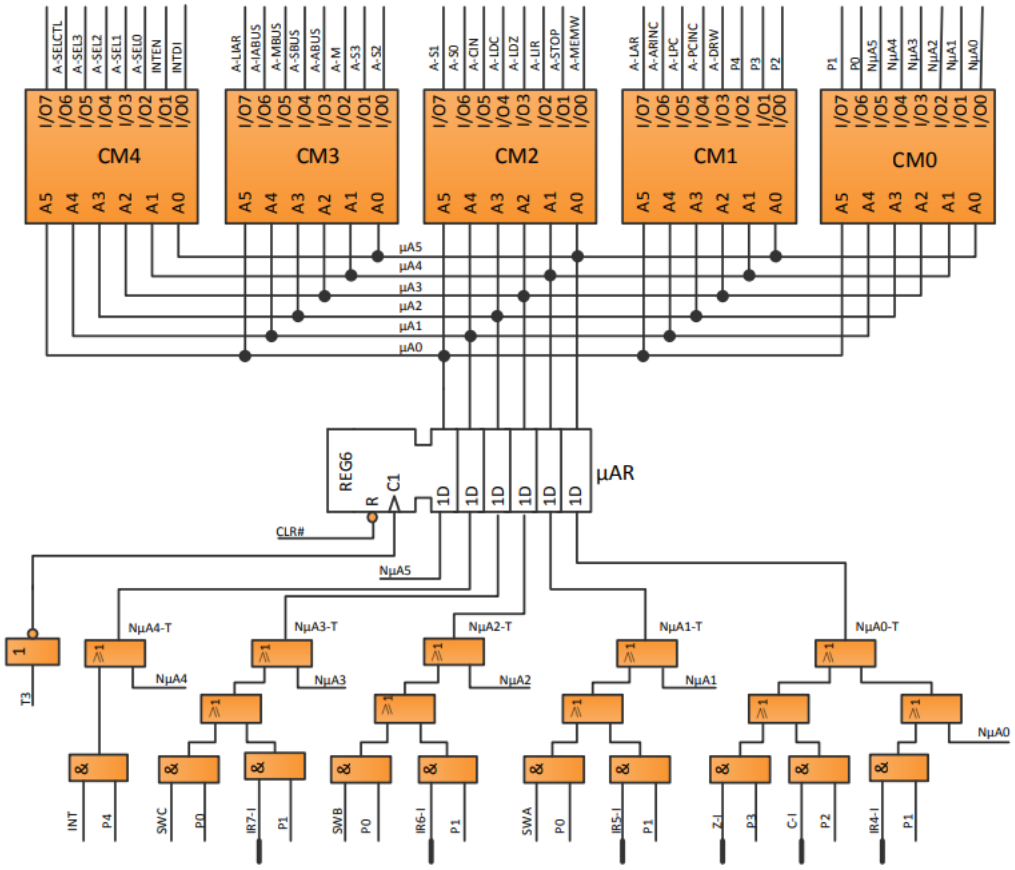


图 2.7 微程序控制器电路图

图 2.7 中，以短粗线标志的信号都有接线孔。信号 IR4-I、IR5-I、IR6-I、IR7-I、C-I

和 Z-I 的实际意义分别等同于 IR4、IR5、IR6、IR7、C 和 Z。INT 信号是时序发生器接收到 中断请求脉冲 PULSE(高电平有效)后产生的中断信号。

(1) 控制存储器

控制存储器由 5 片 58C65 组成，在图 2.6 中表示为 CM4~CM0。其中 CM0 存储微指令最低 的 8 位微代码，CM4 存储微指令最高的 8 位微代码。控制存储器的微代码必须与微指令格式 一致。58C65 是一种 8K×8 位的 E 2 PROM 器件，地址位为 A12~A0。由于 TEC-8 模型计算机只 使用其中 64 个字节作为控制存储器，因此将 A12~A6 接地，A5~A0 接微地址 μA5~μA0。在正常工作方式下，5 片 E 2 PROM 处于只读状态；在修改控制存储器内容时，5 片 E 2 PROM 处于读、 写状态。

(2) 微地址寄存器

微地址寄存器 μAR 由 1 片 74LS174LS 组成，74LS174 是一个 6D 触发器。当按下复位按 钮 CLR 时，产生的信号 CLR#(负脉冲)使微地址寄存器复位，μA5~μA0 为 00H，供读出第一条 微指令使用。在一条微指令结束时，用 T3 的下降沿将微地址转移逻辑产生的下条微指令地 址 NμA5、NμA4-T~NμA0-T 写入微地址寄存器。

(3) 微地址转移逻辑

微地址转移逻辑由若干与门和或门组成，实现“与~或”逻辑。深入理解微地址转移逻辑，对于理解计算机的本质有很重要的作用。计算机现在的功能很强大，但是它是建立在两个很重要的基础之上，一个是最基本的加法和减法功能，一个是条件转移功能。设想一下， 如果没有条件转移指令，实现 10000 个数相加，至少需要 20000 条指令，还不如用算盘计算速度快。可是有了条件转移指令后，一万个数相加，不超过 20 条指令就能实现。因此可以 说，最基本的加法和减法功能和条件转移功能给计算机后来的强大功能打下了基础。本实验中微地址转移逻辑的实现方法是一个很简单的例子，但对于理解条件转移的实现方法大有益处。

4. 实验任务

(1) 正确设置模式开关 SWC、SWB、SWC，用单微指令方式(单拍开关 DP 设置为 1)跟踪控制台 操作读寄存器、写寄存器、读存储器、写存储器的执行过程，记录下每一步的微地址 μA5~μA0、判别位 P4~P0 和有关控制信号的值，写出这 4 种控制台操作的作用和使用方法。

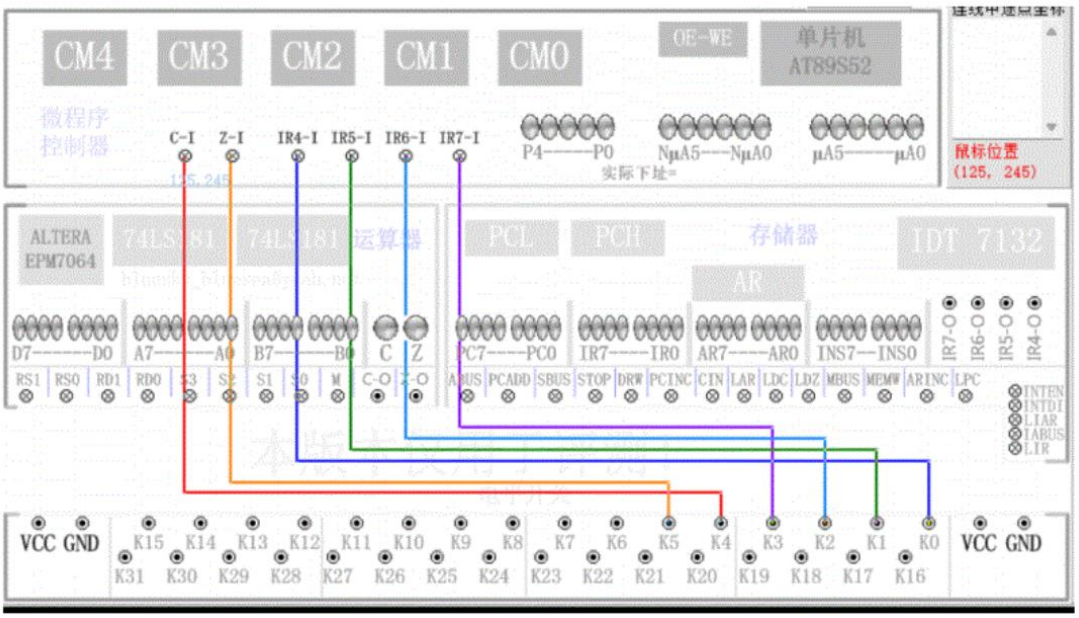
(2) 正确设置指令操作码 IR7~IR4，用单微指令方式跟踪除停机指令 STP 之外的所有指令的 执行过程。记录下每一步的微地址 μA5~μA0、判别位 P4~P0 和有关控制信号的

值。对于 JZ 指令，跟踪 Z=1、Z=0 两种情况；对于 JZ 指令，跟踪 C=1、C=0 两种情况。

三、 实验过程

1.实验准备：将控制器转换开关拨到微程序位置,微程序灯亮，将编程开关设置为正常位置，将单拍开关设置为 1(朝上)。在单拍开关 DP 为 1 时，每按一次 QD 按钮，只执行一条微指令。将信号 IR4-I、IR5-I、IR6-I、IR7-I、C-I、Z-I 依次通过接线孔与电平 K0~K5 连接。通过拨动开关 K0~K5，可以对上述信号设置希望的值。打开电源。按图接线。

K5	K4	K3	K2	K1	K0
Z-I	C-I	IR7-I	IR6-I	IR5-I	IR4-I



2. 读写寄存器、读写存储器，并将 75H、32H、28H、ABH 分别写入 R0、R1、R2、R3 寄存器：按复位按钮 CLR 后，拨动操作模式开关 SWC、SWB、SWA 到希望的位置，按一次 QD 按钮，则进入希望的控制台操作模式。控制台模式开关和控制台操作的对应关系如下:

操作模式	功能选择	备注
000	启动程序运行	
001	写存储器	
010	读存储器	
011	读寄存器	
100	写寄存器	

按一次复位按钮 CLR 按钮，能够结束本次跟踪操作，开始下一次跟踪操作。

(1) 写入寄存器：打开 DP 单排模式，连接电源。调节 SWC、SWB、SWA 为 100，

则现在在地址为 00H 的指令位置，点击 QD，进入地址为 08H 的部分，可以看出，SBUS 总线开关被打开，DRW 为 1，寄存器处于可写入的状态，RD1、RD0 分别是 0 和 1，对应 LR1 可向 R1 寄存器写入数据。在数据开关中设置数据，在此按下 QD，可以看到 R1 中存入了数据，R2 处于被写入的状态，微指令地址跳转到了 0AH 处。再进行两次吸入操作，微指令地址重新回到 00H。

(2) 读取寄存器：按下 CLR 归位，将 SWC、SWB、SWA 设置为 001，读寄存器模式。最初 P0 为 1，处于根据 NuA5-NuA0 和模式开关 SWC、SWB、SWA 确定下一条微指令的地址 SWC、SWB、SWA 为 011。最后要跳转到地址为 07H，按下 QD 观察结果。R1 和 R3 寄存器的值被读出，此时的 P4-P0，NuA5-NuA0 的值全为 0，下一次将跳转回 00H。

(3) 写入存储器：按下 CLR 复位，将 SWC、SWB、SWA 设置为 110 的存储器实验模式，此时 P0 为 1，NuA0 为 1。输入将跳转到地址为 00H 的指令，点击 QD，可以观察到 SBUS 被打开，MEMW 为 1 左端口写入功能打开，ARINC 为 1 地址自增开启在数据开关中输入数据，观察下一步，地址仍会跳转到 02H，仍然进行写入存储器操做，可以看到存储器中被写入了数据。

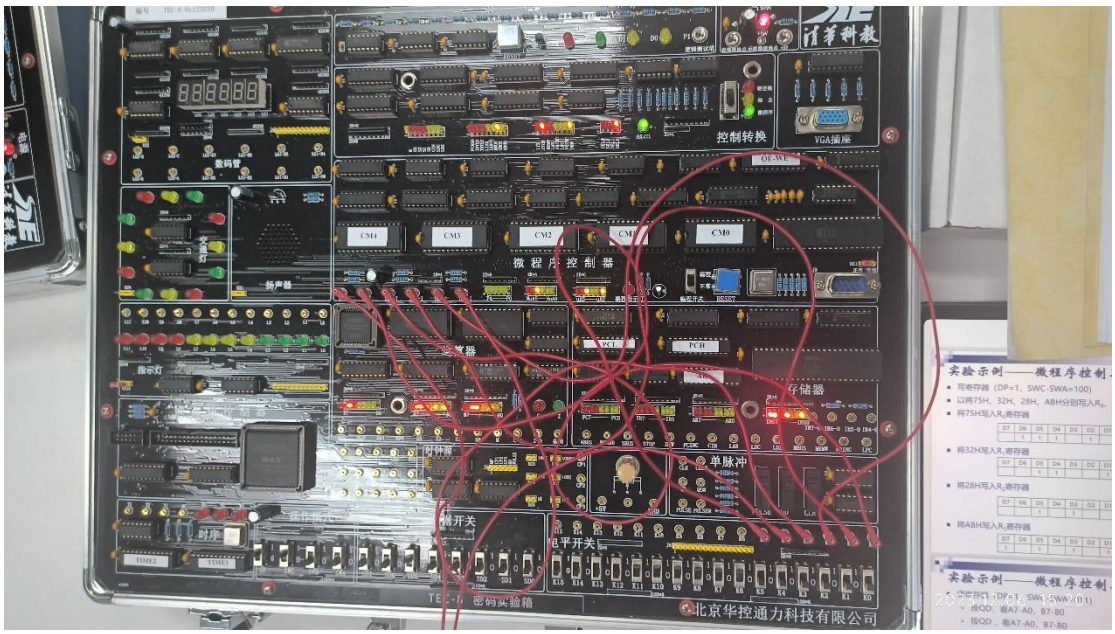
(4) 读取存储器：点击 clr 复位，设置模式开关为合适位置，之后进行指令跳转，便会跳转到 05H 观察存储器，右端口读入被打开，可以看到向左端口写入数据的通道被打通，随后跳转到地址为 32H 处，此时微地址的指令仍跳转到 04H，可知一直在进行都存储器模式，分析一下读存储器模式的逻辑，先读取存储器中第一个地址中的内容，作为地址进行跳转，随后不断输出，跳转后存储器中的内容。

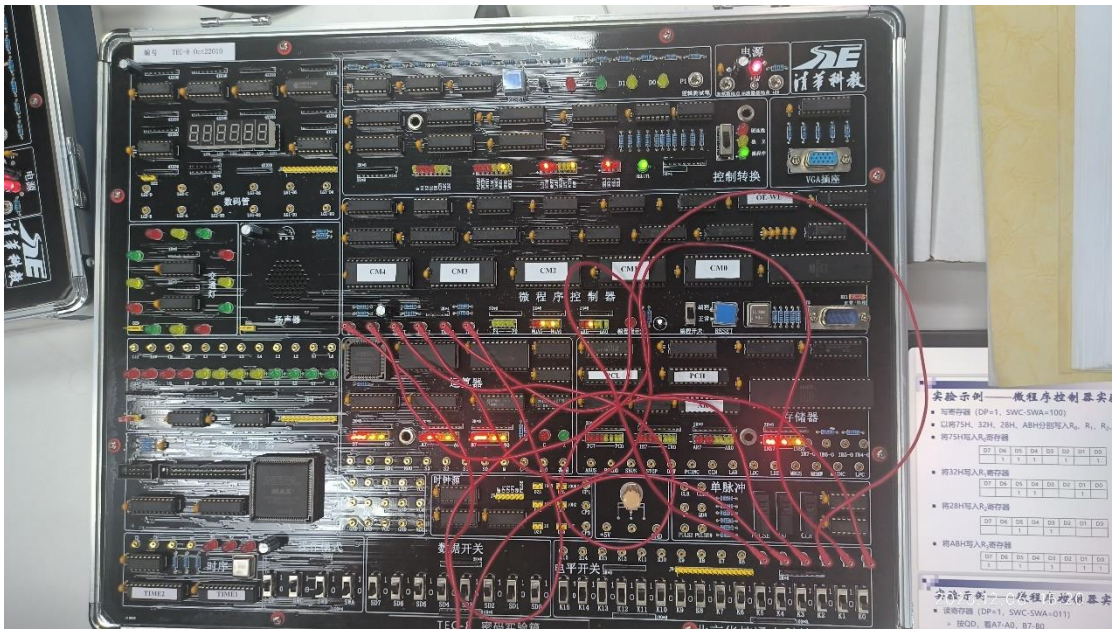
3. 接着上一个步骤，寄存器 R0、R1、R2、R3 分别被写入值 75H、32H、28H、ABH，设置操作模式 000，启动程序运行：

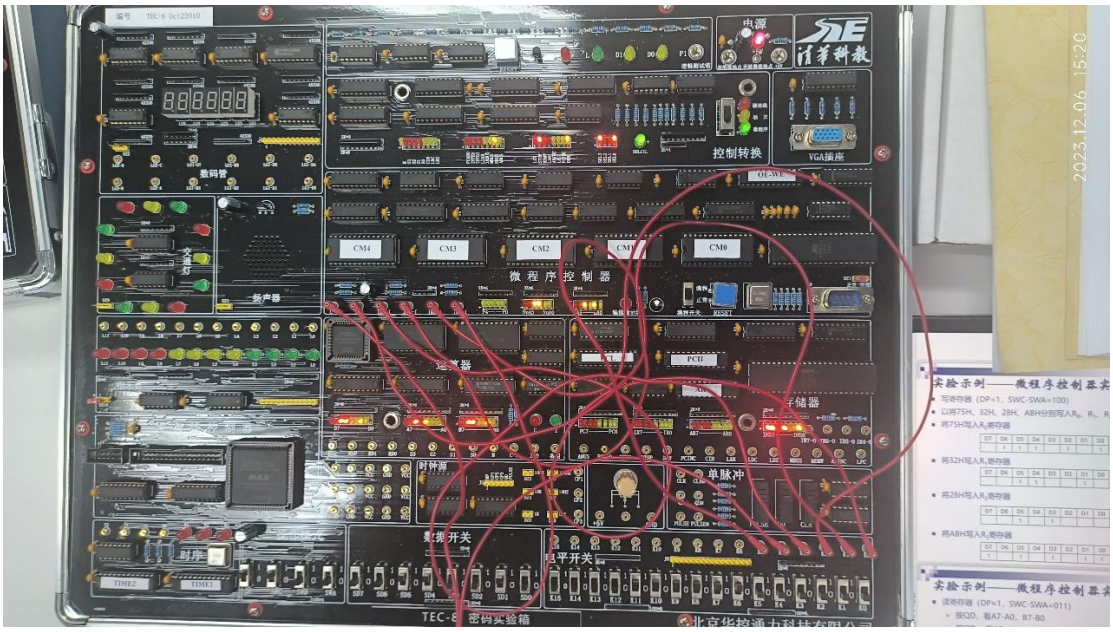
(1) ADD 指令：拨 K0-K3 为 0001，按 QD，进入到 ADD 指令阶段。读取操作数： 从寄存器 R1 中读取操作数 1，从寄存器 R2 中读取操作数 2。这涉及到将 R1 和 R2 的地址设置为相应的值，使得相应的寄存器内容能够通过数据总线(DBUS)送到算术逻辑单元(ALU)的 A 和 B 端口。执行加法操作： 将 ALU 的 A 和 B 端口的内容相加。这需要设置 ALU 的运算类型和控制端口。将结果写回寄存器： 将 ALU 的运算结果写回到寄存器 R3 中。这需要设置写寄存器的控制信号。设置条件码： 根据 ALU 的运算结果设置条件码，如零标志(Z)、进位标志(C)等。更新微地址： 根据指令执行的结果，确定下一条微指令的地址。在 ADD 指令执行完毕后，可能需要跳转到下一条指令或者根据条件跳转。

(2) 条件跳转指令：执行 JC 指令，令 K3（IR7）=0，K2（IR6）=1，K1（IR5）=1，K0（IR4）=1，相当于 JC 指令的操作码。K4(C)=0 情况：按一次 QD 按钮，微地址变为 01H，判别位 P4P0 为 00010。按一次 QD 按钮，微地址变为 27H，判别位 P4P0 为 00100。按一次 QD 按钮，微地址变为 12H，判别位 P4P0 为 10000。按一次 QD 按钮，微地址回到 01H，判别位 P4P0 为 00010。K4(C)=1 情况：按一次 QD 按钮，微地址变为 01H，判别位 P4P0 为 00010。按一次 QD 按钮，微地址变为 27H，判别位 P4P0 为 00100。按一次 QD 按钮，微地址变为 13H，判别位 P4P0 为 10000。按一次 QD 按钮，微地址回到 01H，判别位 P4P0 为 00010。执行 JZ 指令，令 K3（IR7）=1，K2（IR6）=0，K1（IR5）=0，K0（IR4）=0，相当于 JZ 指令的操作码。K5(Z)=0 情况：按一次 QD 按钮，微地址变为 01H，判别位 P4P0 为 00010。按一次 QD 按钮，微地址变为 28H，判别位 P4P0 为 01000。按一次 QD 按钮，微地址变为 12H，判别位 P4P0 为 10000。按一次 QD 按钮，微地址回到 01H，判别位 P4P0 为 00010。K5(Z)=1 情况：按一次 QD 按钮，微地址变为 01H，判别位 P4P0 为 00010。按一次 QD 按钮，微地址变为 28H，判别位 P4P0 为 01000。按一次 QD 按钮，微地址变为 13H，判别位 P4P0 为 10000。按一次 QD 按钮，微地址回到 01H，判别位 P4P0 为 00010。

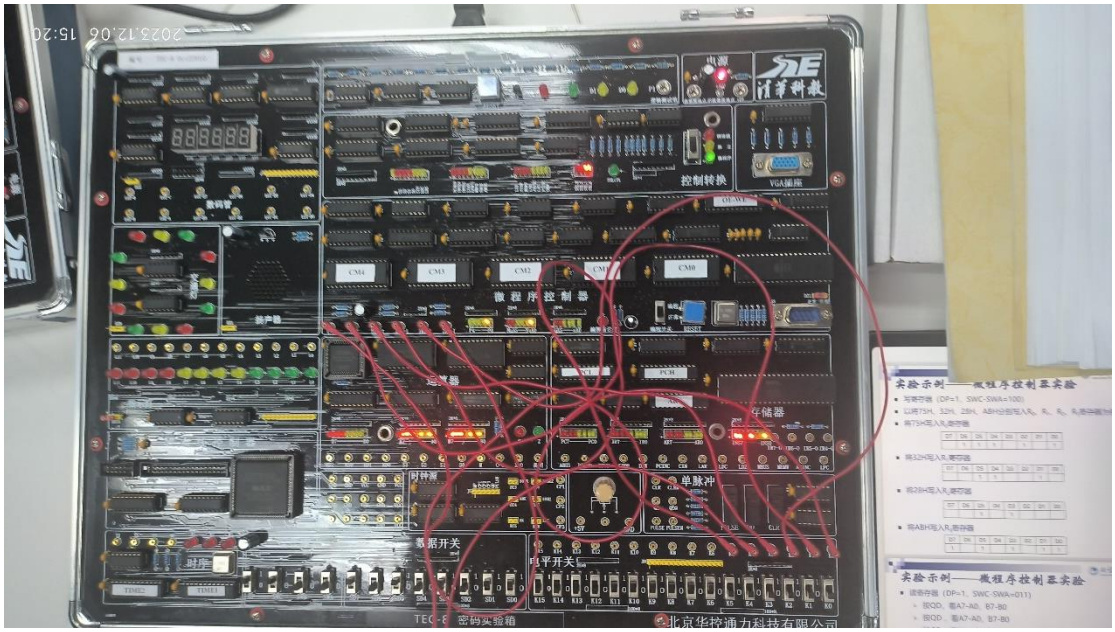
4.实验结果：

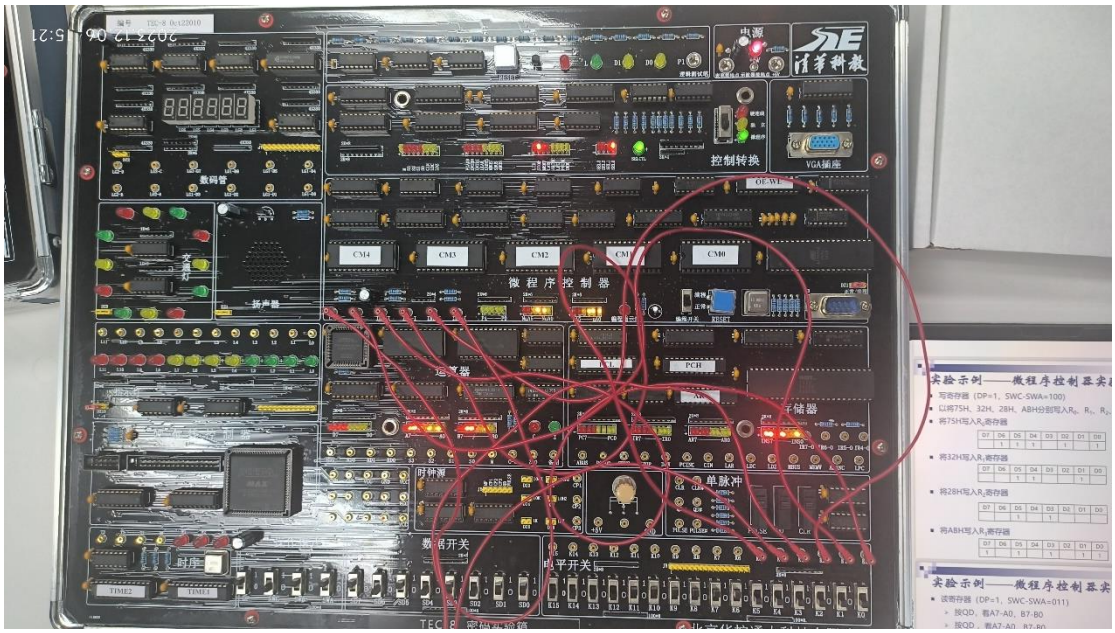

控制转换
VGA插座
微程序控制器
单脉冲
北京华控通力科技有限公司

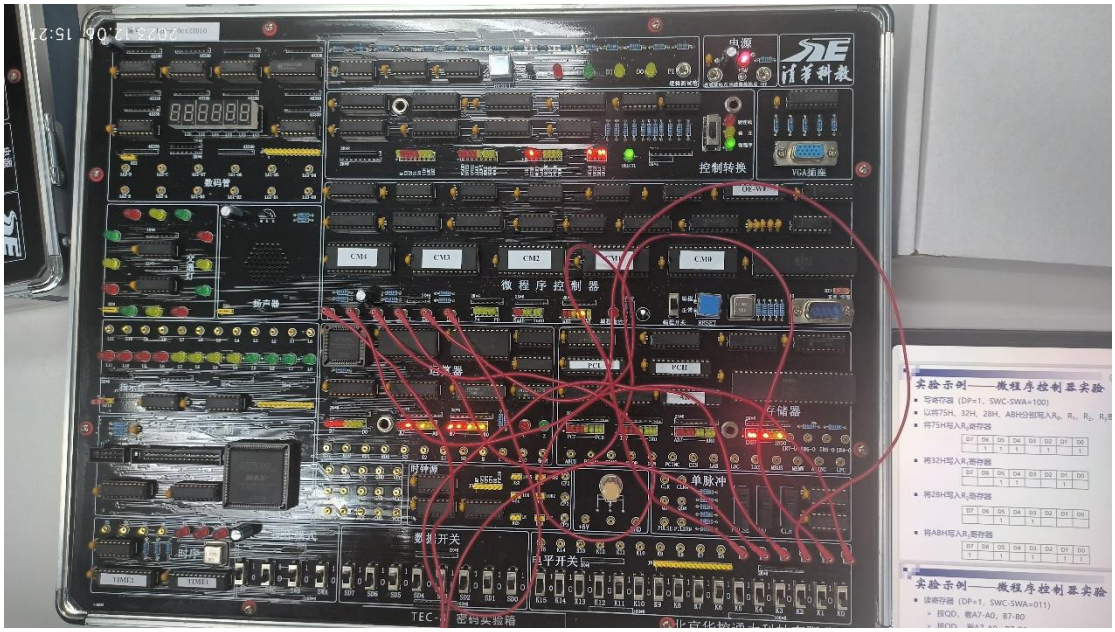

电源
控制转换
VGA插座
微程序控制器
单脉冲

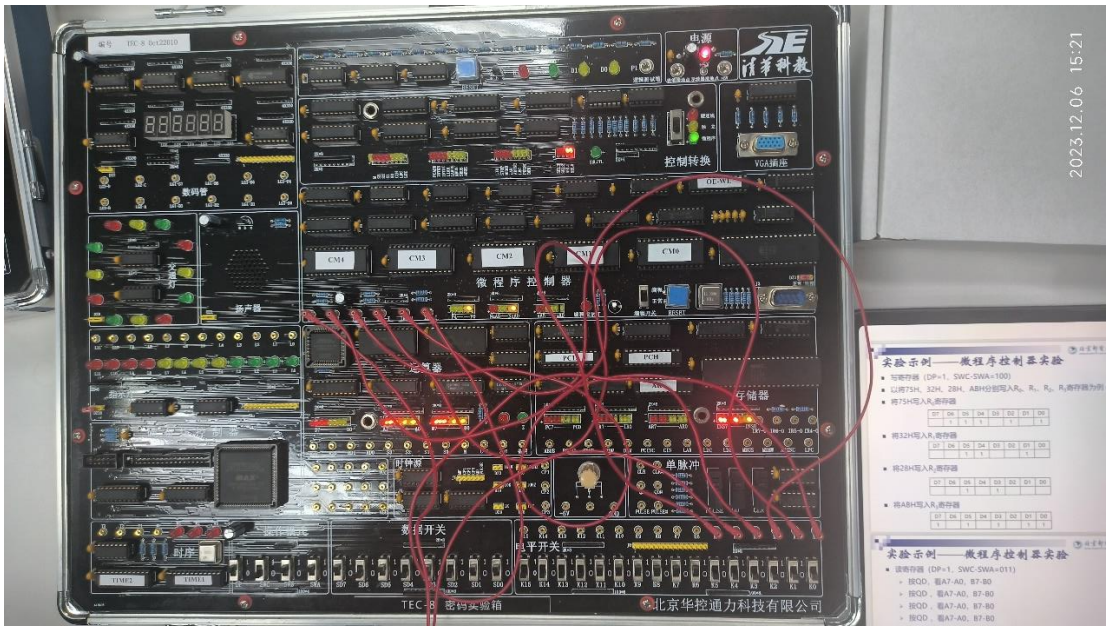

电源
控制转换
VGA插座
微程序控制器
单脉冲
TEC-8 密码实验箱
北京华控通力科技有限公司
2023.12.06 15:20

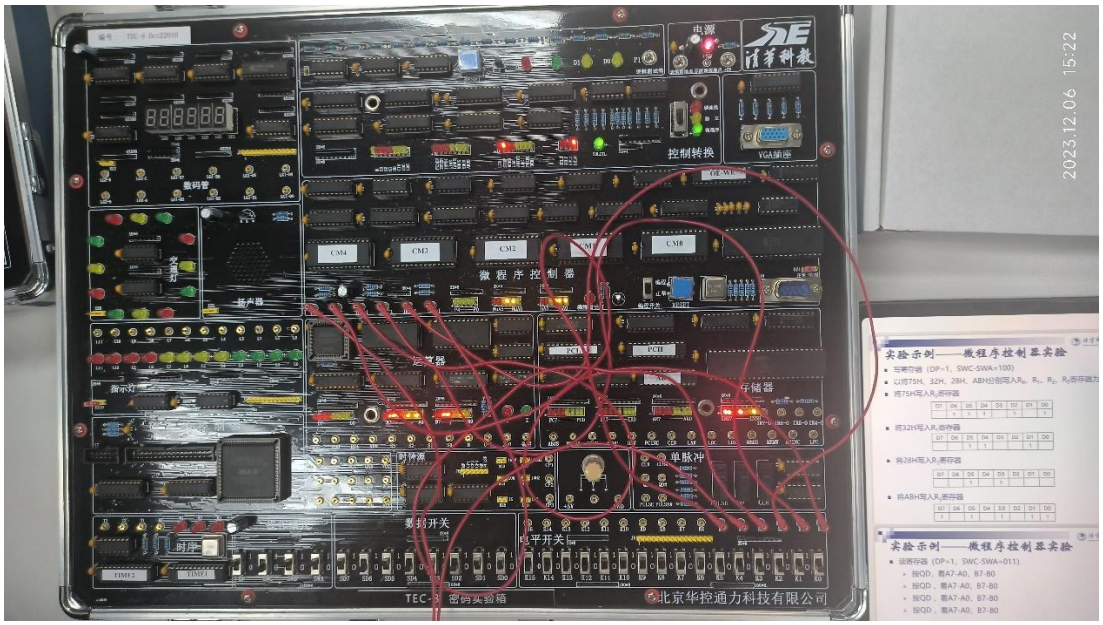

电源
控制转换
VGA插座
微程序控制器
单脉冲
TEC-8 密码实验箱
北京华控通力科技有限公司
15:20

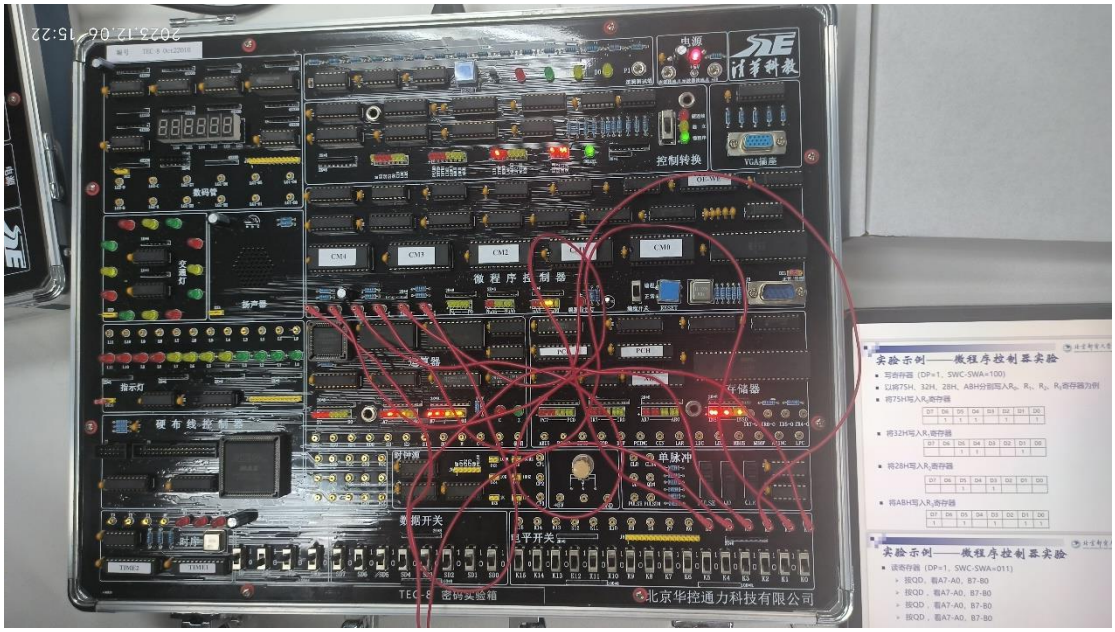

2023.12.06 15:20
控制转换
VGA插座
微程序控制器
单脉冲
数据开关
电平开关
北京华控通力科技有限公司


2023.12.06 15:21
电源
控制转换
VGA插座
微程序控制器
单脉冲
数据开关
电平开关


电源
控制转换
VGA插座
微程序控制器
运算器
存储器
时钟源
单脉冲
数据开关
电平开关
TEC-8 密码实验箱
实验示例——微程序控制器实验


电源
控制转换
VGA插座
微程序控制器
运算器
存储器
时钟源
单脉冲
数据开关
电平开关
TEC-8 密码实验箱
北京华控通力科技有限公司
2023.12.06 15:21
实验示例——微程序控制器实验


2023.12.06 15:22
电源
控制转换
VGA插座
微程序控制器
存储器
单脉冲
数据开关
电平开关
时序
TEC-3 密码实验箱
北京华控通力科技有限公司
实验示例——微程序控制器实验

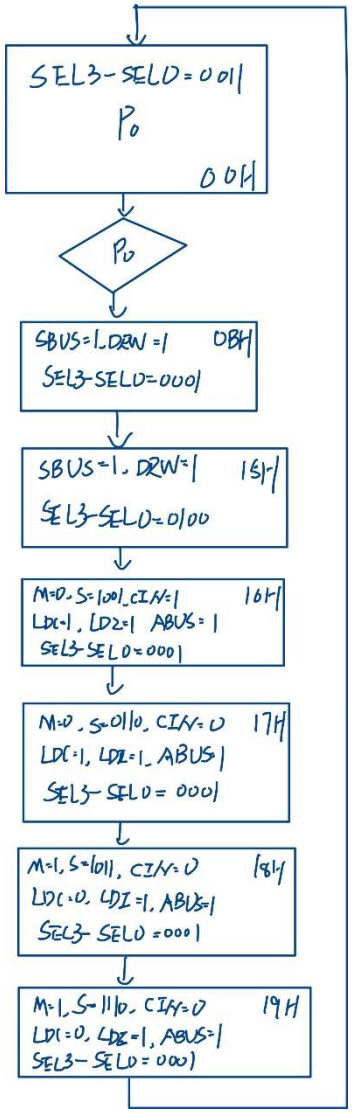

2023.12.06 15:22
电源
控制转换
VGA插座
微程序控制器
存储器
硬布线控制器
单脉冲
数据开关
电平开关
时序
TEC-8 密码实验箱
北京华控通力科技有限公司
实验示例——微程序控制器实验

功能选择	P4~P0	μA	NμA	实际下址	D7-D0	AR	A	B
写寄存器	00001	00H	01H	09H	00H		00H	00H
	00000	09H	08H	08H	F0H		00H	F0H
	00000	08H	0AH	0AH	E0H		00H	E0H
	00000	0AH	0CH	0CH	D0H		00H	D0H
	00000	0CH	00H	00H	00H		F0H	F0H
读寄存器	00001	00H	01H	07H	00H		F0H	E0H
	00000	07H	06H	06H	00H		D0H	C0H
	00000	06H	00H	00H			F0H	F0H
写存储器	00001	00H	01H	03H	00H	00H		
	00000	03H	02H	02H	80H	80H		
	00000	02H	02H	02H	90H	81H		
	00000	02H	02H	02H	91H	82H		
读存储器	00001	00H	01H	05H	00H	00H		
	00000	05H	04H	04H	90H	80H		
	00000	04H	04H	04H	91H	81H		

C 条件转移	0111	000001->100111->010010 上为 C=0，下为 C=1 000001->100111->010011	00010->00100->10000 上为 C=0，下为 C=1 00010->00100->10000	100000->010010->000001 上为 C=0，下为 C=1 100000->010010->000001
Z 条件转移	1000	000001->100011->010010 上为 Z=0,下为 Z=1 000001->101000->010011	00010->01000->10000 上为 Z=0，下为 Z=1 00010->01000->10000	100000->010010->000001 上为 Z=0，下为 Z=1 100000->010010->000001

四、 实验思考与心得

对于问题试根据运算器组成实验、双端口存储器实验和数据通路实验的实验过程。流程图如下。



对于压缩微指令格式长度，去掉下址字段，采用 μPC = μPC + 1 的方式生成微指令地址，每一条指令都有一个下址字段，对控存的浪费是巨大的，增加了一个运算器，减少了下址字段，节约了控存空间。或者可以将多个微操作中相同的或相似的字段合并为一个共享字段。这可以通过使用字段位域或编码来实现。共享字段的使用可以减小每个微指令的长度。

通过微程序控制器实验，我深入了解了微指令格式、微程序流程图、控制电路以及微地址转移逻辑的原理和实现方法。微指令格式在计算机控制单元中扮演着关键的角色，通过对机器指令功能、格式和数据通路所需的控制信号进行设计，实现了对计算机各种操作的灵活控制。在实验中，我对微指令格式的各字段进行了深入的理解，包括微指令字长、顺序字段、控制字段等，这为我更好地理解微程序控制器的运作提供了基础。微程序流程图成为了我理解微程序控制器工作原理的有效工具。通过分析微指令的流程，我能够清晰地了解每个微操作的执行过程，包括对寄存器的读写、存储器的读写以及各种条件转移操作。控制电路的设计与实现也成为了我的关注点，我深入研究了控制存储器、微地址寄存器和微地址转移逻辑

的连接和功能。特别是微地址转移逻辑的实现方法，对于理解条件转移的本质有着重要的作用。通过这个实验，我更加深刻地认识到条件转移对计算机的重要性，它使计算机能够根据运算结果灵活选择执行路径，从而实现更为复杂的控制流程。在实际操作方面，通过正确设置模式开关、指令操作码以及进行单微指令方式的跟踪，我培养了对实际操作的熟练技能。这种实际操作的经验对于将理论知识应用到实际中具有重要的帮助。同时，在实验思考中，对微指令格式的优化进行的思考，如压缩微指令长度、去掉下址字段等，让我思考了计算机体系结构的演进和优化过程，为设计更高效的微程序控制器提供了启示。