

第 1 章 TEC-8 计算机硬件综合实验系统	1
1.1 TEC-8 实验系统的用途	1
1.2 TEC-8 实验系统技术特点	1
1.3 TEC-8 实验系统组成	1
1.4 逻辑测试笔.....	2
1.5 TEC-8 实验系统结构和操作	3
1.6 模型计算机指令系统	6
1.7 开关、按钮、指示灯.....	7
1.9 E2PROM 中微代码的修改	9
第 2 章 计算机组织与体系结构基本实验.....	16
2.1 运算器组成实验	16
2.2 双端口存储器实验	25
2.3 数据通路实验.....	31
2.4 微程序控制器实验.....	39
2.5 CPU 组成与机器指令的执行	46
2.6 中断原理实验.....	52

第 1 章 TEC-8 计算机硬件综合实验系统

1.1 TEC-8 实验系统的用途

TEC-8 计算机硬件综合实验系统，以下简称 TEC-8 实验系统，是清华大学科教仪器厂生产的一个**专利产品**，专利号 ZL 2007 2 0149391.9。它用于**数字逻辑与数字系统、计算机组成原理、计算机体系结构**三门课程的实验教学，也可用于数字系统的研究开发，为提高学生的动手能力、培养学生的创新精神提供了一个良好的舞台

1.2 TEC-8 实验系统技术特点

(1) 模型计算机采用 8 位字长、简单而实用，有利于学生掌握模型计算机整机的工作原理。通过 8 位数据开关用手动方式输入二进制测试程序，有利于学生从最底层开始了解计算机工作原理。

(2) 指令系统采用 4 位操作码，可容纳 16 条指令。已实现加、减、与、加 1、存数、取数、条件转移、无条件转移、输出、中断返回、开中断、关中断和停机等 14 条指令，指令功能非常典型。

(3) 采用双端口存储器作为主存，实现数据总线和指令总线双总线体制，实现指令流水功能，体现出现代 CPU 设计思想。

(4) 控制器采用微程序控制器、硬连线控制器和独立 3 种类型，体现了当代计算机控制器技术的完备性。

(5) 微程序控制器、硬连线控制器和独立之间的转换采用独创的一次全切换方式，切换不用关掉电源，切换简单、安全可靠。

(6) 控制存储器中的微代码可用 PC 计算机下载，省去了 E²PROM 器件的专用编程器和对器件的插、拔。

(7) 运算器中 ALU 采用 2 片 74LS181 实现，4 个 8 位寄存器组用 1 片 EPM7064 实现，设计新颖。

(8) 每一条机器指令的时序采用不定长机器周期方式，符合现代计算机设计思想。

(9) 扩展小板的通用区提供了 15 个双列直插的器件插座，用于《数字逻辑和数字系统》课程的基本实验。

(10) 1 片在系统可编程器件 CPLD 可用于作为硬连线控制器使用，又可用于《数字逻辑与数字系统》课程的大型设计实验。为了安排大型设计实验，提供了用发光二极管代表的按东、西、南、北方向的安排的 12 个交通灯，6 个数码管，1 个喇叭和 1 个 VGA 接口。

(11) 设计《计算机组织与体系结构》课程实验考虑了与前导课程《数字逻辑与数字系统》实验的衔接。由于在《数字逻辑与数字系统》实验中已经进行了大量的接、插线实践，因此在 TEC-8 上进行《计算机组成与体系结构》课程实验接线较少，让学生把精力集中在实验现象的观察、思考和实验原理的理解上。

1.3 TEC-8 实验系统组成

整个 TEC-8 计算机硬件综合实验系统由下列部分构成：电源、试验台、下载

电缆、通讯线。下面分别对各个组成部分予以介绍：

(1) 电源

安装在实验箱的下部，输出+5V，最大电流为 3A。220V 交流电源开关安装在实验箱的右侧。220V 交流电源插座安装在实验箱的背面。实验台上有一个+5V 电源指示灯。

(2) 实验台

实验台安装在实验箱的上部，由一块印制电路板构成。TEC-8 模型计算机安装在这块印制电路板上。学生在实验台上进行实验。

(3) 下载电缆

用于将新设计的硬连线控制器或者其他电路下载到 CPLD 器件中。下载前必须将下载电缆的一端和 PC 机的并行口连接，另一端和实验台上的下载插座连接。

(4) 通讯线

通讯线分为两种一种是 COM 转 COM 的，另外一种 USB 转 COM 的。两种通讯线 COM 转 COM 的和 USB 转 COM 通讯线都是用于在 PC 机上在线修改控制存储器中的微代码。COM 转 COM 两端，一端直接连 PC 机器的 COM 口，另一端连试验台上的 COM 口。USB 通讯线一端接 PC 机的 USB 口，另一端接实验台上的 COM 口。(两种转换线在配件中二选一。原则上标配 COM 转 COM)。

1.4 逻辑测试笔

在数字电路实验中，对信号的测量是一个重要问题。常用的测试工具有示波器、万用表和逻辑测试笔。示波器的好处是直观、准确，用波形显示信号的状态，常用于对连续的周期波形进行测量，数字示波器对非周期信号的测量也很有效，缺点是造价较高。万用表价格便宜，使用方便，对信号电压能进行精确测量，缺点是不能测量脉冲信号。逻辑测试笔常用于测量信号的电平，判断一个较窄的脉冲是否发生以及发生了几个脉冲，缺点是无法对信号的电压作精确测量。数字电路实验中，关心的不是信号的具体电压而是信号的电平，逻辑测试笔作为一种方便、直观的测试工具，得到了广泛应用。TEC-8 实验台上许多信号都连接发光二极管作为指示灯，指示信号的电平，同时配备了逻辑测试笔。TEC-8 实验系统上配置的逻辑测试笔在测试信号的电平时，红灯亮表示高电平，绿灯亮表示低电平，红灯和绿灯都不亮表示高阻态。在测试脉冲个数时，首先按一次 Reset 按钮，使 2 个黄灯 D1、D0 灭，处于测试初始状态。TEC-8 实验台上的逻辑测试笔最多能够测试 3 个连续脉冲。测试信号的状态显示如表 1.1 所示。

表 1.1 指示灯对应得信号的状态

红绿	测试结果	D	测试结果
00	高阻态	00	没有脉冲
10	高电平	01	1 个脉冲
01	低脉冲	10	2 个脉冲
		11	3 个脉冲

数字电路的测试大体上分为静态测试和动态测试两部分。静态测试指的是给定数字电路若干静态输入值，测量输出是否正确。在静态测试基础上，给数字电路输入端加脉冲信号，用示波器或者逻辑测试笔测试数字电路输出是否正确。一般地说，时序电路应当进行动态测试。

1.5 TEC-8 实验系统结构和操作

1.5.1 模型计算机时序信号

TEC-8 模型计算机主时钟 MF 的频率为 1MHz，执行一条微指令需要 3 个节拍脉冲 T1、T2、T3。TEC-8 模型计算机时序采用不定长机器周期，绝大多数指令采用 2 个机器周期 W1、W2，少数指令采用一个机器周期 W1 或者 3 个机器周期 W1、W2、W3。

图 1.1 是 3 个机器周期的时序图。

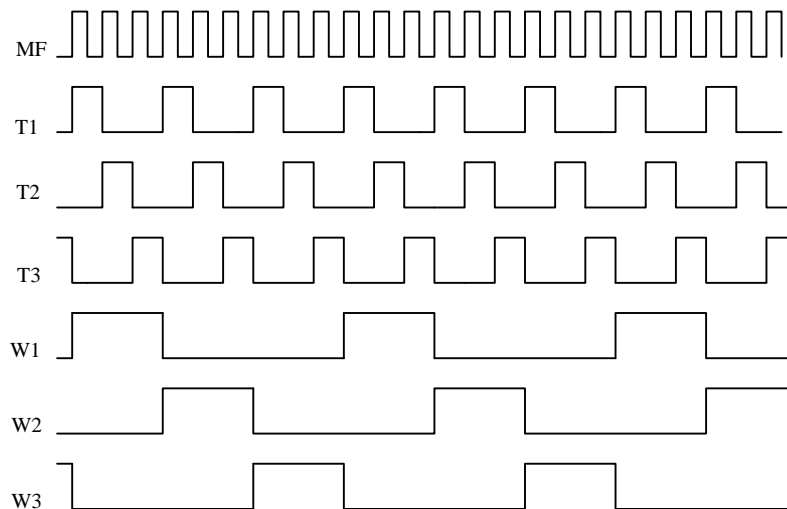


图 1.1 TEC-8 模型计算机时序图

1.5.2 模型计算机组成

图 1.2 是 TEC-8 模型计算机电路框图。下面介绍主要组成模块：时序发生器、算逻运算单元 ALU、双端口寄存器组 7064、数据开关 SD、双端口存储器 7132、组合逻辑控制器、微程序控制器、若干寄存器和若干选择器等；下面分别予以详细介绍：

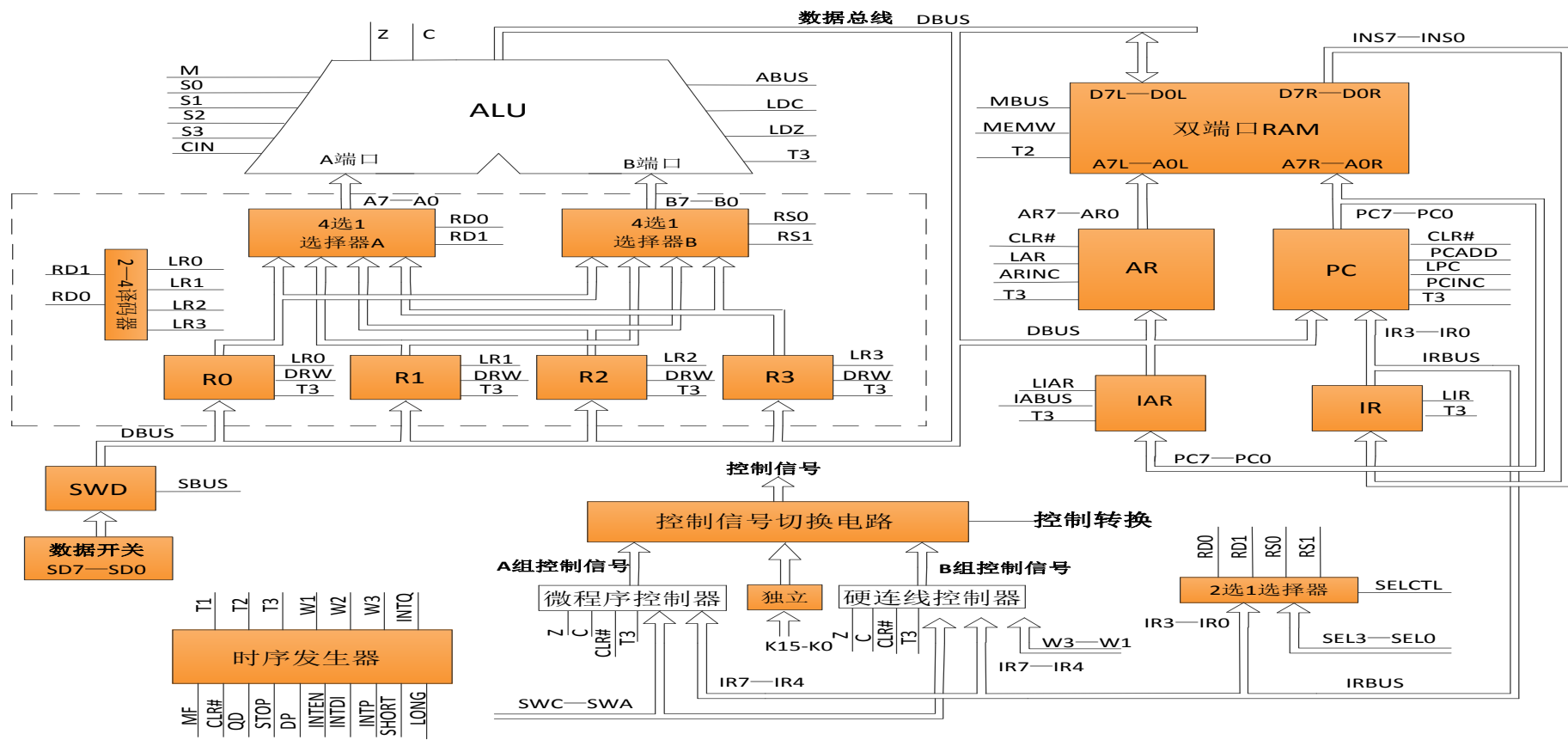


图 1.2 TEC-8 模型计算机框图

1. 时序发生器

它由 2 片 GAL22V10 组成，产生节拍脉冲 T1、T2、T3，节拍电位 W1、W2、W3，以及中断请求信号 ITNQ。主时钟 MF 采用石英晶体振荡器产生的 1MHz 时钟信号。T1、T2、T3 的脉宽为 1 微妙。一个机器周期包含一组 T1、T2、T3。

2. 算术逻辑单元 ALU

算术逻辑单元由 2 片 74LS181 加 1 片 74LS74、1 片 74LS244、1 片 74LS245、1 片 74LS30 组成，进行算术逻辑运算。74LS181 是一个 4 位的算术逻辑器件，2 个 74LS181 级联构成一个 8 位的算术逻辑单元。在 TEC-8 模型计算机中，算术逻辑单元 ALU 对 A 端口的 8 位数和 B 端口的 8 位数进行加、减、与、或和数据传送 5 种运算，产生 8 位数据结果、进位标志 C 和结果为 0 标志 Z。当信号 SBUS 为 1 时，将运算的数据结果送数据总线 DBUS。

3. 双端口寄存器组

双端口寄存器组由 ALTERA 公司的 1 片可编程器件 EPM7064 组成，向 ALU 提供两个运算操作数 A 和 B，保存运算结果。EPM7064 里面包含 4 个 8 位寄存器 R0、R1、R2、R3，4 选 1 选择器 A，4 选 1 选择器 B，2-4 译码器。在图 1.2 中，用虚线围起来的部分全部放在一个 EPM7064 中。4 个寄存器通过 4 选 1 选择器向 ALU 的 A 端口提供 A 操作数，通过 4 选 1 选择器 B 向 ALU 的 B 端口提供 B 操作数，2-4 译码器产生信号 LR0、LR1、LR2 和 LR3，选择保存运算数据结果的寄存器。

4. 数据开关 SD7~SD0

8 位数据开关 SD7~SD0 是双位开关，拨到朝上位置时表示“1”，拨到朝下位置时表示“0”。用于编制程序并把程序放入存储器，设置寄存器 R3~R0 的值。通过拨动数据开关 SD7~SD0 得到的程序或者数据通过 SWD 送往数据总线 DBUS。SWD 是 1 片 74LS244。

5. 双端口 RAM

双端口 RAM 由 1 片 IDT7132 及少许附加电路组成，存放程序和数据。双端口 RAM 是一种 2 个端口可同时进行读、写的存储器，2 个端口各有独立的存储器地址、数据总线和读、写控制信号。在 TEC-8 中，双端口存储器的左端口是个真正的读、写端口，用于程序的初始装入操作，从存储器中取数到数据总线 DBUS，将数据总线 DBUS 上的数写入存储器；右端口设置成只读方式，从右端口读出的指令 INS7~INS0 被送往指令寄存器 IR。

6. 程序计数器 PC、地址寄存器 AR 和中断地址寄存器 IAR

程序计数器 PC 由 2 片 GAL22V10 和 1 片 74LS244 组成向双端口 RAM 的左端口提供存储器地址 PC7~PC0，程序计数器 PC 具有 PC 复位功能，从数据总线 DBUS 上装入初始 PC 功能，PC 加 1 功能，PC 和转移偏量相加功能。

地址寄存器 AR 由 1 片 GAL22V10 组成，向双端口 RAM 的左端口提供存储器地址 AR7~AR0。它具有从数据总线 DBUS 上装入初始 AR 功能和 AR 加 1 功能。

中断地址寄存器 IAR 是 1 片 74LS374，它保存中断时的程序地址 PC。

7. 指令寄存器 IR

指令寄存器是 1 片 74LS273，用于保存从双端口 RAM 中读出的指令。它的输出 IR7~IR4 送往硬连线控制器、微程序控制器，IR3~IR0 送往 2 选 1 选择器。

8. 微程序控制器

微程序控制器产生 TEC-8 模型计算机所需的各种控制信号。它由 5 片 HN58C65、1 片 74LS174、3 片 74LS32 和 3 片 74LS06 组成。5 片 HN58C65 组成控制存储器，存放微程序代码；1 片 74LS174 是微地址寄存器。3 片 74LS32 和 3 片

74LS08 组成微地址转移逻辑。

9. 硬连线控制器

硬连线控制器由 1 片可编程器件 CPLD 组成，产生 TEC-8 模型计算机所需的各种控制信号。

10. 控制信号切换电路

控制信号切换器由 7 片 74LS244 和 1 个三刀转换开关组成。拨动一次转换开关，就能够实现一次控制信号的切换。当转换开关拨到朝上位置时，硬连线控制器灯亮，TEC-8 模型计算机使用硬连线控制器产生的控制信号；当转换开关拨到中间位置时，TEC-8 模型计算机各个部件独立，控制信号需要通过开关来控制；当转换开关拨到朝下位置时，TEC-8 模型计算机使用微程序控制器产生的控制信号。

11. 2 选 1 选择器

2 选 1 选择器由 1 片 74LS244 组成，用于在指令中的操作数 IR3~IR0 和控制信号 SEL3~SEL0 之间进行选择，产生目的寄存器编码 RD1、RD0，产生源寄存器编码 RS1、RS0。

1.6 模型计算机指令系统

TEC-8 模型计算机是个 8 位机，字长是 8 位。多数指令是单字指令，少数指令是双字指令。指令使用 4 位操作码，最多容纳 16 条指令。

已实现加法、减法、逻辑与、加 1、存数、取数、Z 条件转移、C 条件转移、无条件转移、输出、中断返回、开中断、关中断和停机 14 条指令。指令系统如表 12.2 所示。

表 1.2 TEC-8 模型计算机指令系统

名称	助记符	功 能	指令格式		
			IR(7-4)	IR(3-2)	IR(1-0)
加法	ADD Rd, Rs	$Rd \leftarrow Rd + Rs$	0001	Rd	Rs
减法	SUB Rd, Rs	$Rd \leftarrow Rd - Rs$	0010	Rd	Rs
逻辑与	AND Rd, Rs	$Rd \leftarrow Rd \text{ and } Rs$	0011	Rd	Rs
加 1	INC Rd	$Rd \leftarrow Rd + 1$	0100	Rd	XX
取数	LD Rd, [Rs]	$Rd \leftarrow [Rs]$	0101	Rd	Rs
存数	ST Rs, [Rd]	$Rs \rightarrow [Rd]$	0110	Rd	Rs
C 条件转移	JC addr	C=1, 则 $PC \leftarrow @ + \text{offset}$	0111	offset	
Z 条件转移	JZ addr	Z=1, 则 $PC \leftarrow @ + \text{offset}$	1000	offset	
无条件转移	JMP [Rd]	$PC \leftarrow Rd$	1001	Rd	XX
输出	OUT Rs	$DBUS \leftarrow Rs$	1010	XX	Rs
中断返回	IRET	返回断点	1011	XX	XX
关中断	DI	禁止中断	1100	XX	XX
开中断	EI	允许中断	1101	XX	XX
停机	STP	暂停运行	1110	XX	XX

1.2 中，XX 代表随意值。Rs 代表源寄存器号，Rd 代表目的寄存器号。在条件转移指令中，@代表当前 PC 的值，offset 是一个 4 位的有符号数，第 3 位是

符号位，0 代表正数，1 代表负数。注意：@不是当前指令的 PC 值，是当前指令的 PC 值加 1。

指令系统中，指令操作码 0000B 没有对应的指令，实际上指令操作码 0000B 对应着一条 nop 指令，即什么也不做的指令。当复位信号为 0 时，对指令寄存器 IR 复位，使 IR 的值为 00000000B，对应一条 nop 指令。这样设计的目的是适应指令流水的初始状态要求。

1.7 开关、按钮、指示灯

为了在实验过程中观察各种数据，TEC-8 实验系统设置了大量的指示灯。

1、与运算器有关的指示灯

信号名称	信号灯符号	备 注
数据总线指示灯	D7~D0	
运算器 A 端口指示灯	A7~A0	
运算器 B 端口指示灯	B7~B0	
进位信号指示灯	C	
结果为 0 信号指示灯	Z	

2、与存储器有关的指示灯

信号名称	信号灯符号	备 注
程序计数器指示灯	PC7~PC0	
地址指示灯	AR7~AR0	
存储器右端口数据指示灯	INS7~INS0	
指令寄存器指示灯	IR7~IR0	
双端口存储器右端口数据指示灯	INS7~INS0	

3、与微程序控制器有关的信号指示灯

在使用微程序控制器时，控制信号指示灯指示微程序控制器产生的控制信号以及后继微地址 $N\mu A5 \sim N\mu A0$ 和判别位 $P4 \sim P0$ ，微地址指示灯指示当前的微地址 $\mu A5 \sim \mu A0$ ；微地址指示灯 $\mu A5 \sim \mu A0$ 、后继微地址 $N\mu A4 \sim N\mu A0$ 和判别位指示灯 $P4 \sim P0$ 只在微程序控制器方式下有实际意义，在硬布线控制器和独立方式下没有实际意义。

信号名称	信号灯符号	备 注
微地址指示灯	$\mu A5 \sim \mu A0$	
后继微地址指示灯	$N\mu A5 \sim N\mu A0$	
判别位指示灯	$P4 \sim P0$	

4、节拍脉冲信号和节拍电位信号指示灯

按下启动按钮 QD 后，至少产生一组节拍脉冲 T1、T2、T3，无法用指示灯显示 T1、T2、T3 的状态，因此设置了 T1、T2、T3 观测插孔，使用 TEC-8 实验台上提供的逻辑测试笔能够观测 T1、T2、T3 是否产生。

硬连线控制器产生的节拍电位信号 W1、W2 和 W3 有对应的指示灯。

5、其它指示灯

信号名称	信号灯符号
控制台操作指示灯	=1 时，表明进行控制台操作；
	=0 时，表明运行测试程序；
硬连线控制器指示灯	=1 时，表明使用硬连线控制器；
	=0 时，表明使用微程序控制器或者独立状态；
+5V 指示灯	指示+5V 电源的状态

TEC-8 实验平台上有下列按钮：

1、启动按钮 QD

按一次启动按钮 QD，则产生 2 个脉冲 QD 和 QD#。QD 为正脉冲，QD#为负脉冲，脉冲的宽度与按下 QD 按钮的时间相同。正脉冲 QD 启动节拍脉冲信号 T1、T2 和 T3。

2、复位按钮 CLR

按一次复位按钮 CLR，则产生 2 个脉冲 CLR 和 CLR#。CLR 为正脉冲，CLR#为负脉冲，脉冲的宽度与按下 CLR 按钮的时间相同。负脉冲 CLR#使 TEC-6 模型计算机复位，处于初始状态。

3. 中断按钮 PULSE

按一次中断按钮 PULSE，则产生 2 个脉冲 PULSE 和 PULSE#。PULSE 为正脉冲，PULSE#为负脉冲，脉冲的宽度与按下 PULSE 按钮的时间相同。正脉冲 PULSE 向 TEC-8 模型计算机发出中断请求。

TEC-8 实验平台上有下列开关：

1、数据开关 SD7~SD0

这 8 个双位开关用于向寄存器中写入数据、向存储器中写入程序或者用于设置存储器初始地址。当开关拨到朝上位置时为 1，拨到向下位置时为 0。

2、电平开关 S15~S0

这 16 个双位开关用于在实验时设置信号的电平。每个开关上方都有对应的接插孔，供接线使用。开关拨到朝上位置时为 1，拨到向下位置时为 0。

3、单微指令开关 DP

单微指令开关控制节拍脉冲信号 T1、T2、T3 的数目。当单微指令开关 DP 朝上时，处于单微指令运行方式，每按一次 QD 按钮，只产生一组 T1、T2、T3；当单微指令开关 DP 朝下时，处于连续运行方式，每按一次 QD 按钮，开始连续产生 T1、T2、T3，直到按一次 CLR 按钮或者控制器产生 STOP 信号为止。

4. 控制器转换开关

当控制器转换开关朝上时，使用硬连线控制器；当控制器转换开关朝下时，使用微程序控制器。

5. 编程开关

当编程开关朝下时，TEC-8 模型计算机处于正常工作状态；当编程开关朝上时，处于编程状态。在编程状态下，修改控制存储器中的微代码状态，。

6、操作模式开关 SWC、SWB、SWA

操作模式 SWC、SWB、SWA 确定的 TEC-8 模型计算机操作模式如下：

操作模式	实验功能	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	
101	运算器组成实验	
110	双端口存储器实验	
111	数据通路实验	

1.9 E²PROM 中微代码的修改

1. E²PROM 的两种工作方式

TEC-8 模型计算机中的 5 片 E²PROM(CM4~CM0) 有两种工作方式，一种叫“**正常**”工作方式，作为控制存储器使用；一种叫“**编程**”工作方式并且 DZ11 的短路子放置在“编程”位置，可用于修改 E²PROM 的微代码。当编程开关拨到“正常”位置并且 DZ11 放置在“正常”位置，则 TEC-8 可以正常做实验，CM4~CM0 作为控制存储器使用，里面的微代码正常读出，供数据通路使用。当编程开关拨到“编程”位置时，CM4~CM0 只受 TEC-8 实验系统中的单片机和 DZ11 的控制，用来对 5 片 E²PROM 编程。在编程状态下，不进行正常实验。**特别提示：正常实验时编程开关的位置必须拨到“正常”位置、DZ11 短路子必须放置在左边【“正常”位置】，否则可能破坏 E²PROM 原先的内容。**

2. 安装 Prolific usb-to-serical comm prot 驱动程序

如果通讯线不是 COM 对 COM，而是 COM 对 USB，则需要安装驱动。PC 计算机通过 RS232 串行通讯方式和 TEC-8 实验系统中的单片机 89S52 通讯，从而达到修改控制存储器 E²PROM 的目的。如果 TEC-8 实验系统上的编程线采用 USB 对 COM 通讯线，因此需要一个驱动程序，将 USB 通讯方式转换为 RS232 通讯方式，这个驱动程序就是 Prolific usb-to-serical comm prot。出厂时提供的光盘上有这个驱动程序，在出厂资料盘/7 驱动文件夹内。

当第一次用出厂时提供的编程电缆将 PC 机的一个 USB 口和 TEC-8 实验系统上的 USB 口连接时，PC 机自动检测出安装了新硬件，并自动启动“安装新硬件驱动程序”服务，在 PC 机屏幕上弹出“找到新的硬件向导”第一个对话框，如图 1.3 所示。

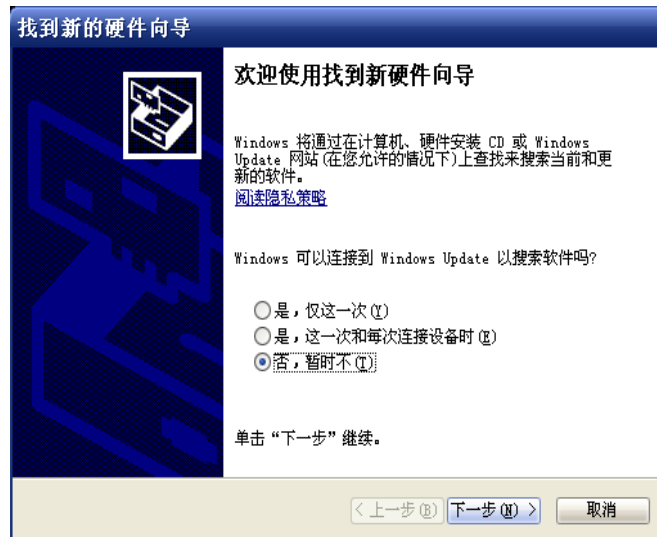


图 1.3 找到新硬件向导对话框 1

直接点击找到新的硬件向导右下角的取消框，采用其他方式安装。打开出厂资料盘/7 驱动目录，直接安装驱动即可。安装完毕后如下所示：



图 1.4 初安装完后串口状态

现在串口的端口号为 COM5，如果需要调整，则如下操作所示：

1. 在 COM5 上右击属性如下所示：



图 1.5 端口属性

2. 点击端口属性如下所示：



1.6 端口设置

3. 更改端口号如下所示：

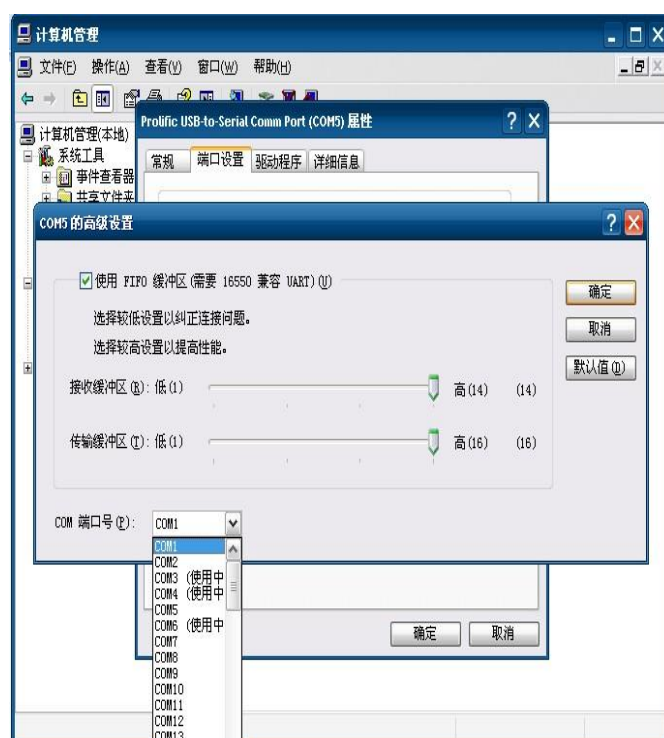


图 1.7 更改端口号

4. 点击确定后在计算机管理中刷新即可得到需要的端口号，如下所示：



图 1.8 设置完成的串口

3. 串口调试助手 2.2 介绍

顾名思义，串口调试助手是一个调试 PC 机串口的程序，在 TEC-8 实验系统中，首先在 PC 机上通过串口调试程序将新的 E²PROM 数据下载到单片机中，由单片机完成对 E2PROM 的编程。

串口调试助手使用极其简单。通过双击出厂时提供的该软件的图标，PC 机屏幕上出现如图 1.8 所示的该软件对话框。



图 1.9 串口调试助手对话框

(1) 选择串口号

选择和 TEC-8 通讯使用的串口号，在 COM1~COM4 中选择一个。串口的设置要与 CP2102 USB to UART Bridge Controller 驱动程序将 USB 转换的 R232 串口号一致。该串口号可用下列方式得到。

在用编程电缆将 PC 机一个 USB 口和 TEC-8 实验系统连接的情况下，用鼠标右键点击 PC 机桌面上的“我的电脑”图标，弹出一个菜单，如图 1.9 所示。

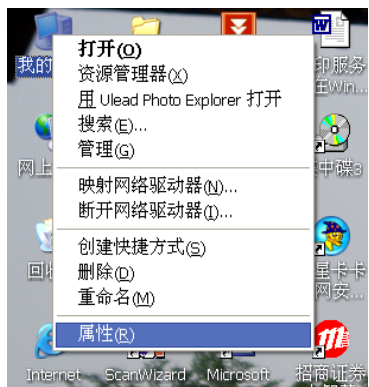


图 1.10 “我的电脑”操作菜单

如图 1.9 那样，点击“属性(R)”菜单项，弹出系统属性对话框，如图 1.10。



图 1.11 系统属性对话框

选中“硬件”菜单项后，点击“设备管理器”按钮，弹出设备管理器窗口，如图 1.12 所示。



图 1.12 设备管理器窗口

在设备管理器窗口中可以找到该 USB 口代替的串口号。图 1.11 中是 COM1。具体的串口号根据 PC 机和当时 PC 机的具体环境而定。

(2) 设置波特率等参数

由于串口调试助手需要和 TEC-8 实验系统上的单片机通讯，因此它设置的串口参数需要和单片机内设置的参数一致，即波特率为 **2400** 波特，数据位 8 位，无校验位，停止位 1 位。这些参数设置不正确将无法通讯。

(3) 窗口下部空白区为 PC 数据发送窗口，其上面较大的空白区为 PC 数据接收窗口。

4. 修改 CM4~CM0 的步骤(备注：原则上不允许修改以免设备不能正常工作，如果需要修改请联系技术支持部门即可，当然 也可以自行修改，有其他疑问可再联系技术支持部门。如果想批量的修改 ROM 中的内容，可以查看随机光盘，内有 TEC8DOWNT0 软件，不用安装下在即可。跳线方式可参考 E²PROM 的两种工作方式中的特别提示。)

(1) 编写二进制格式的微代码文件

微代码文件的格式是二进制。TEC-8 实验系统上使用的 E²PROM 的器件型号是 HN58C65。虽然 1 片 HN58C65 的容量是 2048 个字节，但是在 TEC-8 实验系统中作为控制存储器使用时，每片 HN58C65 都只使用了 64 个字节。因此在改写控制存储器内容时，首先需要生成 5 个二进制文件，每个文件包含 64 个字节。

(2) 连接编程电缆

在 TEC-8 关闭电源的情况下，用出厂时提供的编程电缆将 PC 机的一个 USB 口和 TEC-8 实验系统上的 USB 口相连。

(3) 将编程开关拨到“编程”位置, DZ11 短路子放置“编程”位置。

(4) 将串口调试助手程序打开，设置好串口号和参数。

(5) 打开电源，按一下单片机复位键。

(6) 发送微代码。

串口调试助手的接收区此时会显示信息‘WAITING FOR COMMAND ...’，提示等待命令。这个等待命令的提示信息是 TEC-8 实验系统发送给串口调试助手的，表示 TEC-8 实验系统已准备好接受命令。

一共有 5 个命令，分别是 0、1、2、3 和 4，分别对应被编程的 CM0、CM1、CM2、CM3 和 CM4。

如果准备修改 CM0，则在数据发送区写入‘0’，按‘手动发送’按钮，将命令‘0’发送给 TEC-8 实验系统，通知它要写 CM0 文件了。

数据接收区会出现‘PLEASE CHOOSE A CM FILE’。通过点击“选择发送文件”按钮选择要写入 CM0 的二进制文件。然后点击“发送文件”按钮将文件发往 TEC-8 实验系统。

TEC-8 实验系统接收数据并对 CM0 编程，然后它读出 CM0 的数据和从 PC 机接收到数据比较，不管正确与否，TEC-8 实验系统都向串口调试助手发回结果信息，在数据接收窗口显示出来。

对一个 E²PROM 编程完成后，根据需要可再对其它 E²PROM 编程，全部完成后，按一次 TEC-8 实验系统上的“单片机复位”按钮结束编程。最后将编程开关拨到“正常”位置，并且将 DZ11 放置在左端。

注意：对 CM0、CM1、CM2、CM3 和 CM4 的编程顺序无规定，只要在发出器件号后紧接着发送该器件的编程数据(文件)即可。编程也可以只对一个或者几个

E²PROM 编程，不一定对 5 个 E²PROM 全部编程。

第 2 章 计算机组织与体系结构基本实验

2.1 运算器组成实验

2.1.1 运算器组成实验：为程序控制器方式

一、实验类型：原理性+分析性

二、实验目的

- (1)熟悉逻辑测试笔的使用方法。
- (2)熟悉 TEC-8 模型计算机的节拍脉冲 T1、T2、T3；
- (3)熟悉双端口通用寄存器组的读写操作；
- (4)熟悉运算器的数据传送通路；
- (5)验证 74LS181 的加、减、与、或功能；
- (6)按给定的数据，完成几种指定的算术、逻辑运算运算。

三、实验设备

序列号	名 称	数 量	备 注
1	TEC-8 实验系统	1 台	
2	双踪示波器	1 台	
3	直流万用表	1 块	
4	逻辑测试笔	1 支	在实验箱上方

四、实验电路

为了进行本实验，首先需要了解 TEC-8 模型计算机的基本时序。在 TEC-8 中，执行一条微指令(或者在硬连线控制器中完成 1 个机器周期)需要连续的 3 个节拍脉冲 T1、T2 和 T3。它们的时序关系如图 2.1 所示：

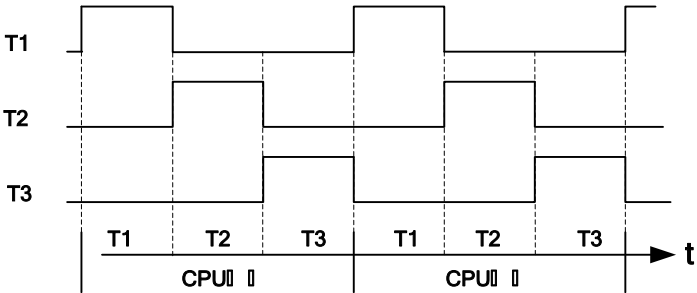


图 2.1 机器周期与 T1、T2、T3 时序关系图

对于运算器操作来说，在 T1 期间，产生 2 个 8 位参与运算的数 A 和 B，A 是被加数，B 是加数；产生控制运算类型的信号 M、S3、S2、S1、S0 和 CIN；产生控制写入 Z 标志寄存器的信号 LDZ 和控制写入 C 标志寄存器的信号 LDC，产生将运算的数据结果送往数据总线 DBUS 的控制信号 ABUS。这些控制信号保持到 T3 结束；在 T2 期间，根据控制信号，完成某种运算功能；在 T3 的上升沿，保存运算的数据结果到一个 8 位寄存器中，同时保存进位标志 C 和结果为 0 标志 Z。

图 2.2 是运算器组成实验的电路图。

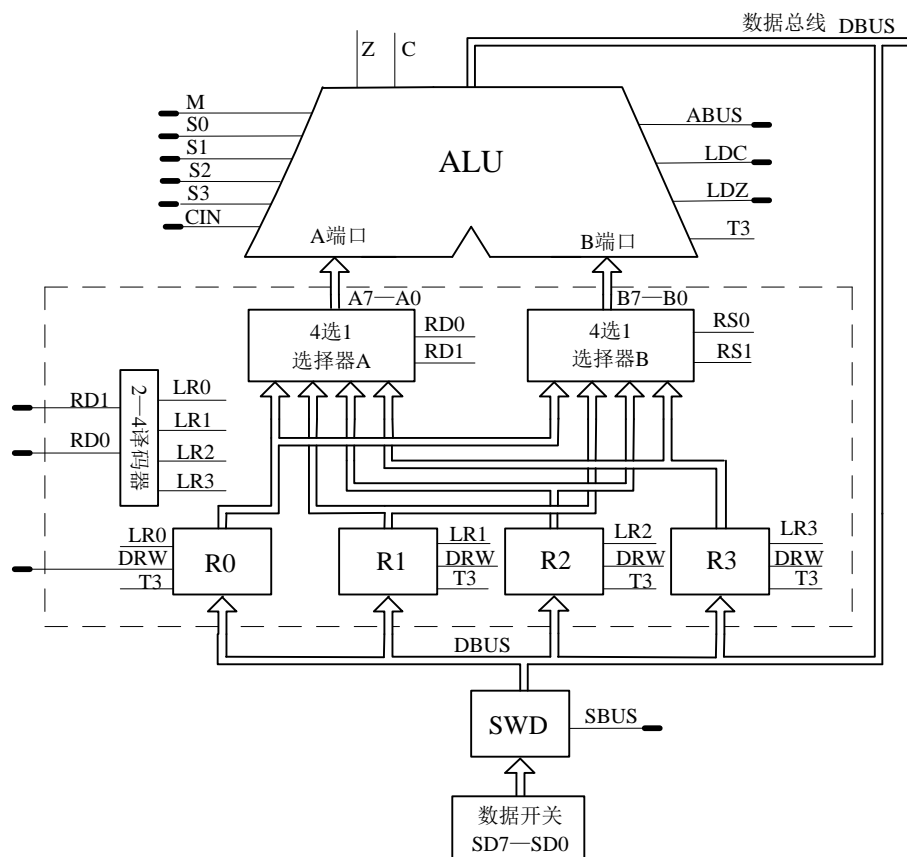


图 2.2 运算器组成实验电路图

双端口寄存器组由 1 片 EPM7064 (U40) (图 2.2 中用虚线围起来的部分) 组成, 内部包含 4 个 8 位寄存器 R0、R1、R2、R3, 4 选 1 选择器 A, 4 选 1 选择器 B 和 1 个 2-4 译码器。根据信号 RD1、RD0 的值, 4 选 1 选择器 A 从 4 个寄存器中选择 1 个寄存器送往 ALU 的 A 端口。根据信号 RS1、RS0 的值, 4 选 1 选择器 B 从 4 个寄存器中选择 1 个寄存器送往 ALU 的 B 端口。2-4 译码器对信号 RD1、RD0 进行译码, 产生信号 LR0、LR2、LR3、LR4, 任何时刻这 4 个信号中只有一个为 1, 其它信号为 0。LR3~LR0 指示出被写的寄存器。当 DRW 信号为 1 时, 如果 LR0 为 1, 则在 T3 的上升沿, 将数据总线 DBUS 上的数写入 R0 寄存器, 余类推。

数据开关 SD7~SD0 是 8 个双位开关。用手拨动这些开关, 能够生成需要的 SD7~SD0 的值。数据开关驱动器 SWD 是 1 片 74 LS 244 (U50)。在信号 SBUS 为 1 时, SD7~SD0 通过 SWD 送往数据总线 DBUS。在本实验中, 使用数据开关 SD7~SD0 设置寄存器 R0、R1、R2 和 R3 的值。

ALU 由 2 片 74LS181 (U41 和 U42)、1 片 74LS74、1 片 74 LS 244、1 片 74 LS 245 和 1 片 74LS30 构成。74LS181 完成算术逻辑运算, 74 LS 245 和 74 LS 30 产生 Z 标志, 74 LS 74 保存标志 C 和标志 Z。ALU 对 A7~A0 和 B7~B0 上的 2 个 8 位数据进行算术逻辑运算, 运算后的数据结果在信号 ABUS 为 1 时送数据总线 DBUS (D7~D0), 运算后的标志结果在 T3 的上升沿保存进位标志位 C 和结果为 0 标志位 Z。加法和减法同时影响 C 标志和 Z 标志, 与操作和或操作只影响 Z 标志。

应当指出, 74LS181 只是许多种能做做算术逻辑运算器件中的一种器件, 这里它仅作为一个例子使用。

74LS181 能够进行 4 位的算术逻辑运算, 2 片 74 LS 181 级连在一起能够 8 位运算, 3 片 74LS181 级连在一起能够进行 12 位运算, 余类推。所谓级联方式,

就是将低 4 位 74LS181 的进位输出引脚 C_{n+4} 与高 4 位 74LS181 的进位输入引脚

C_n 连接。在 TEC-8 模型计算机中，U42 完成低 4 位运算，U41 完成高 4 位运算，二者级连在一起，完成 8 位运算。在 ABUS 为 1 时，运算得到的数据结果送往数据总线 DBUS。数据总线 DBUS 有 4 个信号来源：运算器、存储器、数据开关和中断地址寄存器，在每一时刻只允许其中一个信号源送数据总线。

本实验中用到的信号归纳如下：

序列号	信号名	功能说明
1	S3、S2、S1、S0	控制 74LS181 的运算类型
2	CIN	低位 74LS181 的进位输入
3	SEL3、SEL2 (RD1、RD0)	选择送 ALU 的 A 端口的寄存器
4	SEL1、SEL0 (RS1、RS0)	选择送 ALU 的 B 端口的寄存器
5	DRW	=1 时，在 T3 上升沿对 RD1、RD0 选中的寄存器进行写操作，将数据总线 DBUS 上的数 D7~D0 写入选定的寄存器
6	SETCTL	=1 时，实验系统处于实验台状态。 =0 时，实验系统处于运行程序状态
7	SBUS	=1 时，将运算结果送数据总线 DBUS =0 时，禁止运算结果送数据总线 DBUS
8	ABUS	=1 时，将运算结果送数据总线 DBUS
9		=0 时，禁止运算结果送数据总线 DBUS
10	M	运算模式：M=0 为算术运算；M=1 逻辑运算；
11	LDZ	=1 时，如果运算结果为 0，在 T3 的上升沿，将 1 写入到 Z 标志寄存器；如果运算结果不为 0，将 0 保存到 Z 标志寄存器。
12	LDC	=1 时，在 T3 的上升沿将运算得到的进位保存到 C 标志寄存器。
13	A7~A0	送往 ALU 的 A 端口的数
14	B7~B0	送往 ALU 的 B 端口的数
15	D7~D0	数据总线 DBUS 上的 8 位数
16	C	进位标志
17	Z	结果为 0 标志

上述信号都有对应的指示灯。当指示灯灯亮时，表示对应的信号为 1；当指示灯不亮时，对应的信号为 0。实验过程中，对每一个实验步骤，都要记录上述信号（可以不记录 SETCTL）的值。另外 $\mu A5 \sim \mu A0$ 指示灯指示当前微地址。

应当指出，74LS181 对减法运算采用的是补码运算方式，即先求得[-减数]的补码，然后和被减数的补码相加的方式完成。因此一个较大的数减去一个较小的数，或者 2 个相等的数相减时产生进位。

五、实验任务

1. 用双踪示波器和逻辑测试笔测试节拍脉冲信号 T1、T2、T3。
2. 对下述 7 组数据进行加、减、与、或运算。

- | | |
|-------------------|--------------------|
| (1) A=0F0H, B=10H | (5) A=0FFH, B=0AAH |
| (2) A=10H, B=0F0H | (6) A=55H, B=0AAH |
| (3) A=03H, B=05H | (7) A=0C5H, B=61H |
| (4) A=0AH, B=0AH | |

六、实验步骤

1. 实验准备

将控制器转换开关拨到微程序位置，将编程开关设置为正常位置，将开关 DP 拨到向上位置。打开电源。

2. 用逻辑测试笔测试节拍脉冲信号 T1、T2、T3

(1) 将逻辑测试笔的一端插入 TEC-8 实验台上的“逻辑测试笔”上面的插孔中，另一端插入“T1”上方的插孔中。

(2) 按复位按钮 CLR，使时序信号发生器复位。

(3) 按一次逻辑测试笔框内的 Reset 按钮，使逻辑测试笔上的脉冲计数器复位，2 个黄灯 D1、D0 均灭。

(4) 按一次启动按钮 QD，这时指示灯 D1、D0 的状态应为 01B，指示产生了一个 T1 脉冲；如果再按一次 QD 按钮，则指示灯 D1、D0 的状态应当为 10B，表示又产生了一个 T1 脉冲；继续按 QD 按钮，可以看到在单周期运行方式下，每按一次 QD 按钮，就产生一个 T1 脉冲。

(5) 用同样的方法测试 T2、T3。

3. 进行加、减、与、或实验

(1) 设置加、减、与、或实验模式

按复位按钮 CLR，使 TEC-8 实验系统复位。指示灯 $\mu A5 \sim \mu A0$ 显示 00H。将操作模式开关设置为 SWC=1、SWB=0、SWA=1，准备进入加、减、与、或实验。

按一次 QD 按钮，产生一组节拍脉冲信号 T1、T2、T3，进入加、减、与、或实验。

(2) 设置数 A

指示灯 $\mu A5 \sim \mu A0$ 显示 0BH。在数据开关 SD7~SD0 上设置数 A。在数据总线 DBUS 指示灯 D7~D0 上可以看到数据设置的正确不正确，发现错误需及时改正。设置数据正确后，按一次 QD 按钮，将 SD7~SD0 上的数据写入 R0，进入下一步。

(3) 设置数 B

指示灯 $\mu A5 \sim \mu A0$ 显示 15H。这时 R0 已经写入，在指示灯 B7~B0 上可以观察到 R0 的值。在数据开关 SD7~SD0 上设置数 B。设置数据正确后，按一次 QD 按钮，将 SD7~SD0 上的数据写入 R1，进入下一步。

(4) 进行加法运算

指示灯 $\mu A5 \sim \mu A0$ 显示 16H。指示灯 A7~A0 显示被加数 A(R0)，指示灯 B7~B0 显示加数 B(R1)，D7~D0 指示灯显示运算结果 A+B。按一次 QD 按钮，进入下一步。

(5) 进行减法运算

指示灯 $\mu A5 \sim \mu A0$ 显示 17H。这时指示灯 C(红色)显示加法运算得到的进位 C，指示灯 Z(绿色)显示加法运算得到的结果为 0 信号。指示灯 A7~A0 显示被减数 A(R0)，指示灯 B7~B0 显示减数 B(R1)，指示灯 D7~D0 显示运算结果 A-B。按一次 QD 按钮，进入下一步。

(6) 进行与运算

指示灯 $\mu A5 \sim \mu A0$ 显示 18H。这时指示灯 C(红色)显示减法运算得到的进位 C，指示灯 Z(绿色)显示减法运算得到的结果为 0 信号。

指示灯 A7~A0 显示数 A(R0)，指示灯 B7~B0 显示数 B(R1)，指示灯 D7~D0 显示运算结果 A and B。按一次 QD 按钮，进入下一步。

(7)进行或运算

指示灯 $\mu A5 \sim \mu A0$ 显示 19H。这时指示灯 Z(绿色)显示与运算得到的结果为 0 信号。指示灯 C 保持不变。指示灯 A7~A0 显示数 A(R0)，指示灯 B7~B0 显示数 B(R1)，指示灯 D7~D0 显示运算结果 A or B。按一次 QD 按钮，进入下一步。

(8)结束运算

指示灯 $\mu A5 \sim \mu A0$ 显示 00H。这时指示灯 Z(绿色)显示或运算得到的结果为 0 信号。指示灯 C 保持不变。

按照上述步骤，对要求的 7 组数据进行运算。

七、实验要求

1. 做好实验预习，掌握运算器的数据传输通路及其功能特性。

2. 写出实验报告，内容是：

(1)实验目的。

(2)根据实验结果填写表 2.1。

表 2.1 运算器组成实验结果数据表

实验数据		实验结果									
数 A	数 B	加			减			与		或	
		数据结果	C	Z	数据结果	C	Z	数据结果	Z	数据结果	Z

(3)结合实验现象，每一实验步骤中，对下述信号在所起的作用进行解释：M、S0、S1、S2、S3、CIN、ABUS、LDC、LDZ、SEL3、SEL2、SEL1、SEL0、DRW、SBUS。并说明在该步骤中，哪些信号是必需的，哪些信号不是必需的，哪些信号必需采用实验中使用的值，哪些信号可以不采用实验中使用的值。

八、可探索和研究的问题

1. ALU 具有记忆功能吗？如果有，如何设计？

2. 为什么在 ALU 的 A 端口和 B 端口的数据确定后，在数据总线 DBUS 上能够直接观测运算的数据结果，而标志结果却在下一步才能观测到？

2.1.2 运算器组成实验：独立方式

一、实验类型：原理性+分析性

二、实验目的

- (1)熟悉逻辑测试笔的使用方法。
- (2)熟悉 TEC-8 模型计算机的节拍脉冲 T1、T2、T3；
- (3)熟悉双端口通用寄存器组的读写操作；
- (4)熟悉运算器的数据传送通路；
- (5)验证 74LS181 的加、减、与、或等功能；
- (6)按照表中提供的功能自行验证其中几种即可。

三、实验设备

序列号	名 称	数 量	备 注
1	TEC-8 实验系统	1 台	
2	双踪示波器	1 台	
3	直流万用表	1 块	
4	逻辑测试笔	1 支	在实验箱上方

四、实验电路和实验任务

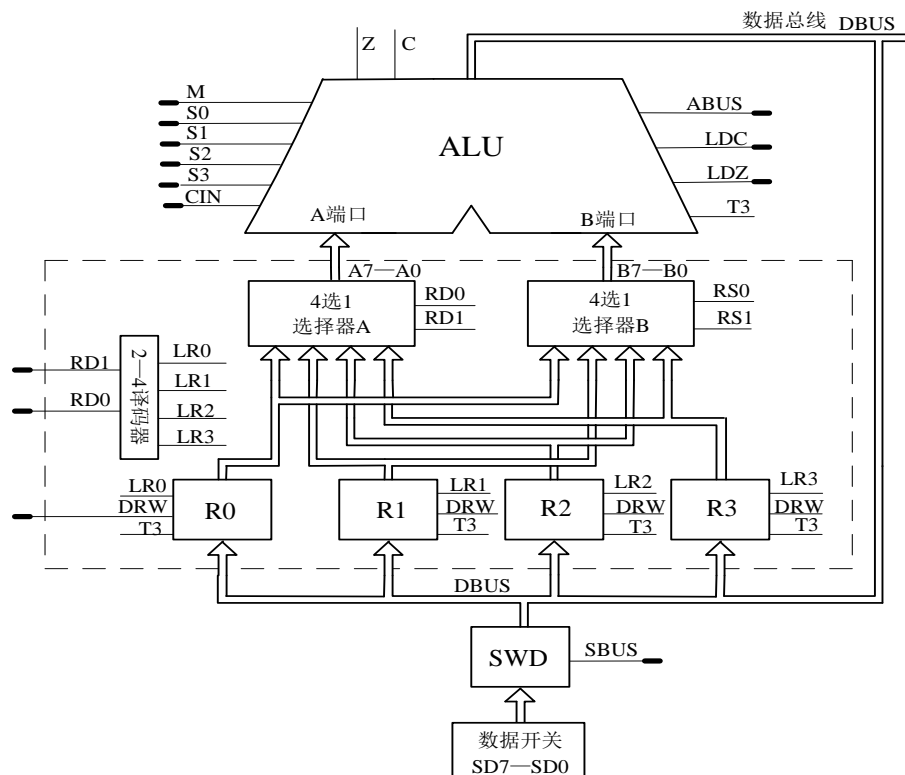


图 2.3 运算器组成实验电路图

(1) 实验电路

上图 2.3 标识出了本实验所用的运算器数据通路图。参与运算的数据首先通过试验台操作板上的 8 个二进制数据开关 SD7-SD0 来设置，然后输入到双端口通用寄存器堆 RF 中。

双端口寄存器堆 RF 由 1 个 ALTERA EPM 7064 实现，功能相当于 4 个八位通用寄存器，用于保存参与运算的数据，运算后的结果也要送到双端口通用寄存器堆 RF 中保存。双端口寄存器堆模块 RF 的控制信号中 RD1、RD0 用于选择送 ALU 的 A 端口(左端口)的通用寄存器。RS1、RS0 用于选择送 ALU 的 B 端口(右端口)的通用寄存器。

(2) 实验任务

1. 按图所示，将运算器模块与实验台操作板上的线路进行连接。由于运算器模块内部的连线已经由印制电路板连接好，故接线任务仅仅是完成数据开关、控制信号模拟开关、与运算模块的外部连线。特别注意：为了建立清楚的整机概念，培养严谨的科研能力，手工连线时绝对有必要的。
2. 用开关 K15-K0 向通用寄存器堆 RF 内的 R3-R0 寄存器置数据。然后读出 R3-R0 的数据，在数据总线 DBUS 上显示出来。
3. 验证 ALU 的正逻辑算术、逻辑运算功能。

注意：进位信号 C 是运算器 ALU 最高进 C_{n+4} 的反，既有进位为 1，无进位为 0。

选择方式				M=1 逻辑运算	M=0 算术运算
S3	S2	S1	S0	逻辑运算	CN=1 【有进位】
0	0	0	0	$F=/A$	$F=A$
0	0	0	1	$F=/(A+B)$	$F=(A+B)$
0	0	1	0	$F=(/A)B$	$F=A+/B$
0	0	1	1	$F=0$	$F=-1$ (补码形式)
0	1	0	0	$F=/(AB)$	$F=A$ 加 $A(/B)$
0	1	0	1	$F=/B$	$F=(A+B)$ 加 A/B
0	1	1	0		$F=A$ 减 B 减 1
0	1	1	1	$F=A/B$	$F=(AB/)$ 减 1
1	0	0	0	$F=/A+B$	$F=A$ 加 AB
1	0	0	1		$F=A$ 加 B
1	0	1	0	$F=B$	$F=(A+/B)$ 加 AB
1	0	1	1	$F=AB$	$F=AB$ 减 1
1	1	0	0	$F=1$	$F=A$ 加 A
1	1	0	1	$F=A+/B$	$F=(A+B)$ 加 A
1	1	1	0	$F=A+B$	$F=(A+/B)$ 加 A
1	1	1	1	$F=A$	$F=A$ 减 1

4. 具体实验步骤：将“控制转换”开关拨到最中间位置既“独立”灯亮。

第一步：测试寄存器写入和读出；【操作模式：1100】

名称	电平控制信号开关	K6	K5	K4	K3	K2	K1	K0
功能	信号名称	RD1	RD0	DRW	SBUS	RS1	RS0	MBUS
置 R0	操作模式			1	1			
置 R1	写 REG 操作模式：1100		1	1	1			
置 R2		1		1	1			
置 R3		1	1	1	1			

备注：写寄存器完成后可以直接在写寄存器操作模式下，通过 K6、K5 拨动开关查看写入寄存器中的数据，对应的数据灯：A7~A0。通过 K2、K1 拨动开关也可以查看写入寄存器中的数据，对应的数据灯 B7~B0。

第二步：运算器实验【操作模式：1101】

名称	K15	K14	K13	K12	K11	K10	K9	K8
序号	M	S0	S1	S2	S3	CIN	ABUS	LDC
	运算器组成操作模式：1101							
	送两个数到 REG，K6K5、K2K1 分别选择加与被加							
		1			1	1	1	1

名称	K7	K6	K5	K4	K3	K2	K1	K0
序号	LDZ	RD1	RD0	DRW	SBUS	RS1	RS0	MBUS
				1	1			
			1	1	1			
	1							

备注：运算器实验答案只提供了加法运算的控制信号，其他运算功能请参考上页中 ALU 表的运算逻辑功能即可。

2.1.3 实验报告记录:

2.2 双端口存储器实验

2.2.1 双端口存储器实验：微程序控制器方式

一、实验类别 原理性+分析性

二、实验目的

- (1)了解双端口静态存储器 IDT7132 的工作特性及其使用方法；
- (2)了解半导体存储器怎样存储和读取数据；
- (3)了解双端口存储器怎样并行读写；
- (4)熟悉 TEC-8 模型计算机中存储器部分的数据通路。

三、实验设备

序列号	名 称	数 量	备 注
1	实验系统	1 台	
2	双踪示波器	1 台	
3	直流万用表	1 块	
4	逻辑测试笔	1 支	在试验箱上方

四、实验电路

图 2.3 是双端口存储器实验的电路图。

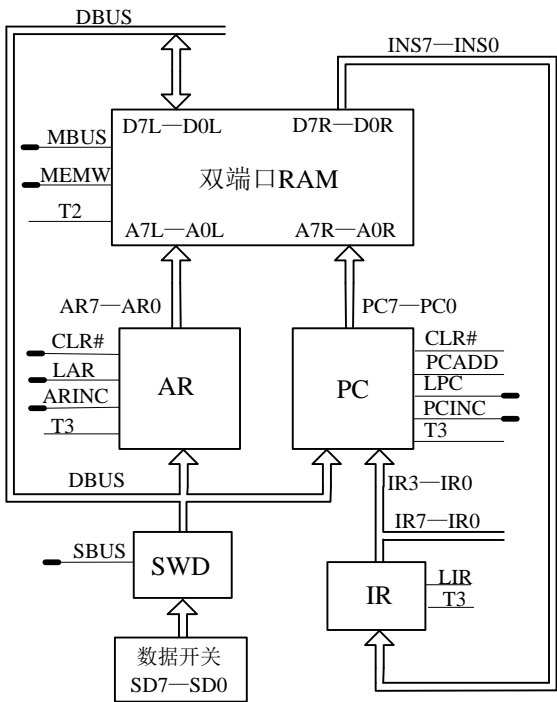


图 2.3 双端口存储器实验电路图

双端口 RAM 电路由 1 片 IDT7132 及少许附加电路组成，存放程序和数据。IDT7132 有 2 个端口，一个称为左端口，一个称为右端口。2 个端口各有独立的存储器地址线、数据线和 3 个读、写控制信号：CE#、R/W#和 OE#，可以同时器件内部的同一存储体同时进行读、写。IDT7132 容量为 2048 字节，TEC-8 实验系统只使用 64 字节。

在 TEC-8 实验系统中，左端口配置成读、写端口，用于程序的初始装入操作，

从存储器中取数到数据总线 DBUS，将数据总线 DBUS 上的数写入存储器。当信号 MEMW 为 1 时，在 T2 为 1 时，将数据总线 DBUS 上的数 D7~D0 写入 AR7~AR0 指定的存储单元；当 MBUS 信号为 1 时，AR7~AR0 指定的存储单元的数送数据总线 DBUS。右端口设置成只读方式，从 PC7~PC0 指定的存储单元读出指令 INS7~INS0，送往指令寄存器 IR。

程序计数器 PC 由 2 片 GAL22V10 (U53 和 U54) 组成。向双端口 RAM 的右端口提供存储器地址。当复位信号 CLR# 为 0 时，程序计数器复位，PC7~PC0 为 00H。当信号 LPC 为 1 时，在 T3 的上升沿，将数据总线 DBUS 上的数 D7~D0 写入 PC。当信号 PCINC 为 1 时，在 T3 的上升沿，完成 PC 加 1。当 PCADD 信号为 1 时，PC 和 IR 中的转移偏量 (IR3~IR0) 相加，在 T3 的上升沿，将相加得到的和写入 PC 程序计数器。

地址寄存器 AR 由 1 片 GAL22V10 (U58) 组成，向双端口 RAM 的左端口提供存储器地址 AR7~AR0。当复位信号 CLR# 为 0 时，地址寄存器复位，AR7~AR0 为 00H。当信号 LAR 为 1 时，在 T3 的上升沿，将数据总线 DBUS 上的数 D7~D0 写入 AR。当信号 ARINC 为 1 时，在 T3 的上升沿，完成 AR 加 1。

指令寄存器 IR 是 1 片 74LS273 (U47)，用于保存指令。当信号 LIR 为 1 时，在 T3 的上升沿，将从双端口 RAM 右端口读出的指令 INS7~INS0 写入指令寄存器 IR。

数据开关 SD7~SD0 用于设置双端口 RAM 的地址和数据。当信号 SBUS 为 1 时，数 SD7~SD0 送往数据总线 DBUS。本实验中用到的信号归纳如下：

名 称	功 能 说 明
MBUS	=1 时，将双端口 RAM 的左端口数据送到数据总线 DBUS。
MEMW	=1 时，在 T2 为 1 期间将数据总线 DBUS 上的 D7~D0 写入双端口 RAM 写入的存储器单元由 AR7~AR0 指定。
LIR	=1 时，在 T3 的上升沿将从双端口 RAM 的右端口读出的指令 INS7~INS0 写入指令寄存器 IR。读出的存储器单元由 PC7~PC0 指定。
LPC	当它为 1 时，在 T3 的上升沿，将数据总线 DBUS 上的 D7~D0 写入程序计数器 PC。
PCINC	=1 时，在 T3 的上升沿 PC 加 1。
LAR	=1 时，在 T3 的上升沿，将数据总线 DBUS 上的 D7~D0 写入地址寄存器 AR。
ARINC	=1 时，在 T3 的上升沿，AR 加 1。
SBUS	=1 时，数据开关 SD7~SD0 的数送数据总线 DBUS。
AR7~AR0	双端口 RAM 左端口存储器地址。
PC7~PC0	双端口 RAM 右端口存储器地址。
INS7~INS0	从双端口 RAM 右端口读出的指令，本实验中作为数据使用。
D7~D0	数据总线 DBUS 上的数。

上述信号都有对应的指示灯。当指示灯灯亮时，表示对应的信号为 1；当指示灯不亮时，对应的信号为 0。实验过程中，对每一个实验步骤，都要记录上述信号 (可以不纪录 SETCTL) 的值。另外 $\mu A5 \sim \mu A0$ 指示灯指示当前微地址。

五、实验任务

1. 从存储器地址 10H 开始，通过左端口连续向双端口 RAM 中写入 3 个数：85H，

- 60H, 38H。在写的过程中, 在右端口检测写的数据是否正确。
2. 从存储器地址 10H 开始, 连续从双端口 RAM 的左端口和右端口同时读出存储器的内容。

六、实验步骤

1. 实验准备

将控制器转换开关拨到微程序位置, 将编程开关设置为正常位置。打开电源。

2. 进行存储器读、写实验

(1) 设置存储器读、写实验模式

按复位按钮 CLR, 使 TEC-8 实验系统复位。指示灯 $\mu A5 \sim \mu A0$ 显示 00H。将操作模式开关设置为 SWC=1、SWB=1、SWA=0, 准备进入双端口存储器实验。

按一次 QD 按钮, 进入存储器读、写实验。

(2) 设置存储器地址

指示灯 $\mu A5 \sim \mu A0$ 显示 0DH。在数据开关 SD7~SD0 上设置地址 10H。在数据总线 DBUS 指示灯 D7~D0 上可以看到地址设置的正确不正确, 发现错误需及时改正。设置地址正确后, 按一次 QD 按钮, 将 SD7~SD0 上的地址写入地址寄存器 AR(左端口存储器地址)和程序计数器 PC(右端口存储器地址), 进入下一步。

(3) 写入第 1 个数

指示灯 $\mu A5 \sim \mu A0$ 显示 1AH。指示灯 AR7~AR0(左端口地址)显示 10H, 指示灯 PC7~PC0(右端口地址)显示 10H。在数据开关 SD7~SD0 上设置写入存储器的第 1 个数 85H。按一次 QD 按钮, 将数 85H 通过左端口写入由 AR7~AR0 指定的存储器单元 10H。

(4) 写入第 2 个数

指示灯 $\mu A5 \sim \mu A0$ 显示 1BH。指示灯 AR7~AR0(左端口地址)显示 11H, 指示灯 PC7~PC0(右端口地址)显示 10H。观测指示灯 INS7~INS0 的值, 它是通过右端口读出的由右地址 PC7~PC0 指定的存储器单元 10H 的值。比较和通过左端口写入的数是否相同。在数据开关 SD7~SD0 上设置写入存储器的第 2 个数 60H。按一次 QD 按钮, 将第 2 个数通过左端口写入由 AR7~AR0 指定的存储器单元 11H。

(5) 写入第 3 个数

指示灯 $\mu A5 \sim \mu A0$ 显示 1CH。指示灯 AR7~AR0(左端口地址)显示 12H, 指示灯 PC7~PC0(右端口地址)显示 11H。观测指示灯 INS7~INS0 的值, 它是通过右端口读出的由右地址 PC7~PC0 指定的存储器单元 11H 的值。比较和通过左端口写入的数是否相同。在数据开关 SD7~SD0 上设置写入存储器的第 3 个数 38H。按一次 QD 按钮, 将第 3 个数通过左端口写入由 AR7~AR0 指定的存储器单元 12H。

(6) 重新设置存储器地址

指示灯 $\mu A5 \sim \mu A0$ 显示 1DH。指示灯 AR7~AR0(左端口地址)显示 13H, 指示灯 PC7~PC0(右端口地址)显示 12H。观测指示灯 INS7~INS0 的值, 它是通过右端口读出的由右地址 PC7~PC0 指定的存储器单元 12H 的值。比较和通过左端口写入的数是否相同。在数据开关 SD7~SD0 重新设置存储器地址 10H。按一次 QD 按钮, 将 SD7~SD0 上的地址写入地址寄存器 AR(左端口存储器地址)和程序计数器 PC(右端口存储器地址), 进入下一步。

(7) 左、右两 2 个端口同时显示同一个存储器单元的内容。

指示灯 $\mu A5 \sim \mu A0$ 显示 1FH。指示灯 AR7~AR0(左端口地址)显示 10H, 指示灯 PC7~PC0(右端口地址)显示 10H。观测指示灯 INS7~INS0 的值, 它是通过右端口

读出的由右地址 PC7~PC0 指定的存储器单元 10H 的值。观测指示灯 D7~D0 的值，它是从左端口读出的由 AR7~AR0 指定的存储器单元 10H 的值。

按一次 QD 按钮，地址寄存器 AR 加 1，程序计数器 PC 加 1，在指示灯 D7~D0 和指示灯 INS7~INS0 上观测存储器的内容。继续按 QD 按钮，直到存储器地址 AR7~AR0 为 12H 为止。

七、实验要求

1. 做好实验预习，掌握双端口存储器的使用方法和 TEC-8 模型计算机存储器部分的数据通路。
2. 写出实验报告，内容是：
 - (1)实验目的。
 - (2)根据实验结果填写表 2.1。

表 2.1 双端口存储器实验结果表

实验数据		实验结果					
左端口存储器地址	通过左端口写入的数据	第一次从右端口读出的数		同时读出时的读出结果			
		右端口存储器地址	读出的数	左端口存储器地址	读出的数	右端口存储器地址	读出的数

(3)结合实验现象，在每一实验步骤中，对下述信号所起的作用进行解释：SBUS、MBUS、LPC、PCINC、LAR、ARINC、MEMW。并说明在该步骤中，哪些信号是必需的，哪些信号不是必需的，哪些信号必需采用实验中使用的值，哪些信号可以不采用实验中使用的值。

八、可研究和探索的问题

1. 在通过左端口向双端口 RAM 写数时，在右端口可以同时观测到左端口写入的数吗？为什么？

2.2.2 双端口存储器实验：独立方式

首先将“控制转换”开关拨到最中间位置既“独立”灯亮。

双端口存储器实验：【操作模式：1110】

名称 序号	操作 模式	K15 SBUS	K14 ARINC	K13 LAR	K12 MEMW	K11 MBUS	K10 PCINC	K9 LPC
1	1110	1		1				1
2		1	1		1			
3		1	1		1			
4		1	1		1			
5		1		1				1
6			1		1		1	
7			1		1		1	
8			1		1		1	

名称 序号	K8 ABUS	SD	实验现象	备注
1	0	10	置 AR、PC/ AR=PC=10	
2		85	写第 1 个数 85/AR=11,PC=10,INS= IR =85	
3		60	写第 2 个数 60/AR=12,PC=10,INS= IR =60	
4		38	写第 3 个数 38/AR=13,PC=10,INS= IR =38	
5		10	重置 AR、PC/ AR=PC=10	
6			AR=PC=10,INS=85	
7			AR=PC=11,INS=60	
8			AR=PC=12,INS=38	

2.2.3 实验报告记录:

2.3 数据通路实验

2.3.1 数据通路实验：微程序控制器方式

一、实验类型 原理性+分析性

二、实验目的

- (1)进一步熟悉 TEC-8 模型计算机的数据通路的结构；
- (2)进一步掌握数据通路中各个控制信号的作用和用法；
- (3)掌握数据通路中数据流动的路径。

三、实验设备

序列号	名 称	数 量	备 注
1	实验系统	1 台	
2	双踪示波器	1 台	
3	直流万用表	1 块	
4	逻辑测试笔	1 支	在试验箱上

四、实验原理

数据通路实验电路图如图 2.4 所示。它由运算器部分、双端口存储器部分加上数据开关 SD7~SD0 连接在一起构成。

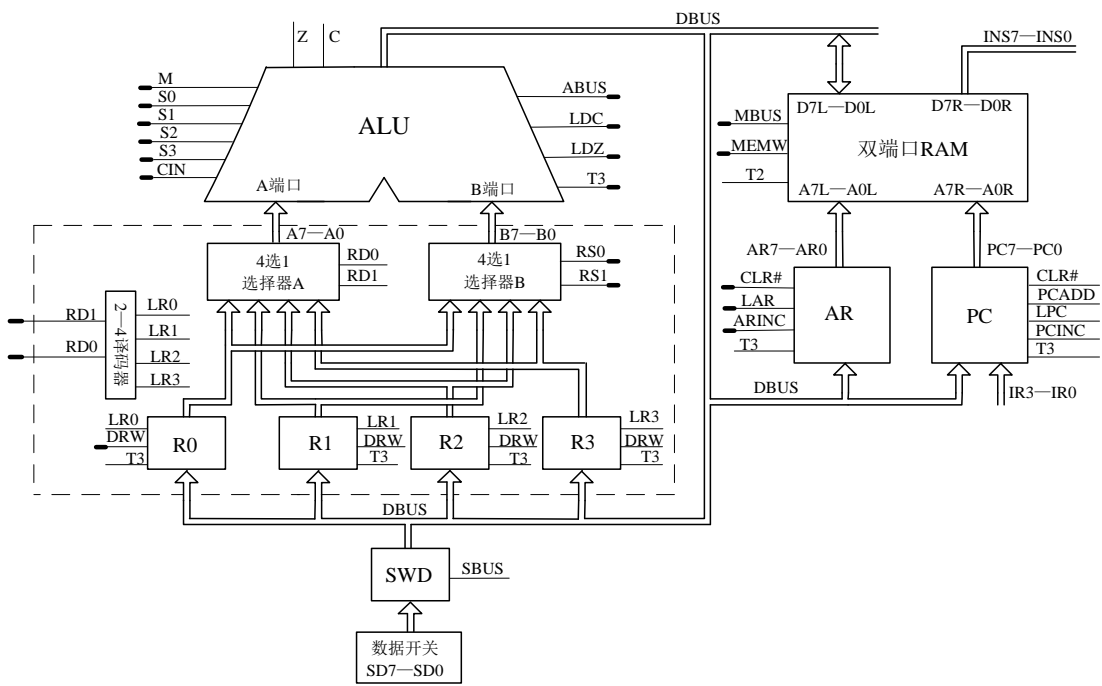


图 2.4 数据通路实验电路图

数据通路中各个部分的作用和工作原理在 2.1 节和 2.2 节已经做过详细说明，不再重述。这里主要说明 TEC-8 模型计算机的数据流动路径和方式。

在进行数据运算操作时，由 RD1、RD0 选中的寄存器通过 4 选 1 选择器 A 送往 ALU 的 A 端口，由 RS1、RS0 选中的寄存器通过 4 选 1 选择器 B 送往 ALU 的 B 端口；信号 M、S3、S2、S1、S1 和 S0 决定 ALU 的运算类型，ALU 对 A 端口和 B 端口的两个数连同 CIN 的值进行算数逻辑运算，得到的数据运算结果在信号 ABUS 为 1 时送往数据总线 DBUS；在 T3 的上升沿，数据总线 DBUS 上的数据结果写入由 RD1、RD0 选中的寄存器。

在寄存器之间进行数据传送操作时，由 RS1、RS0 选中的寄存器通过 4 选 1 选择器 B 送往 ALU 的 B 端口；ALU 将 B 端口的数在信号 ABUS 为 1 时送往数据总线 DBUS；在 T3 的上升沿将数据总线上的数写入由 RD1、RD0 选中的寄存器。ALU 进行数据传送操作由一组特定的 M、S3、S2、S1、S0、CIN 的值确定。

在进行运算操作时，由 RS1、RS0 选中的寄存器通过 4 选 1 选择器 B 送往 ALU 的 B 端口；由 RD1、RD0 选中的寄存器通过 4 选 1 选择器 A 送往 ALU 的 A 端口；ALU 对数 A 和 B 进行运算，运算的数据结果在信号 ABUS 为 1 时送往数据总线 DBUS；在 T3 的上升沿将数据总线上的数写入由 RD1、RD0 选中的寄存器。ALU 进行何种运算操作由 M、S3、S2、S1、S0、CIN 的值确定。

在从存储器中取数操作中，由地址 AR7~AR0 指定的存储器单元中的数在信号 MEMW 为 0 时被读出；在 MBUS 为 1 时送数据总线 DBUS；在 T3 的上升沿写入由 RD1、RD0 选中的寄存器。

在写存储器操作中，由 RS1、RS0 选中的寄存器过 4 选 1 选择器 B 送 ALU 的 B 端口；ALU 将 B 端口的数在信号 ABUS 为 1 时送往数据总线 DBUS；在 MEMW 为 1 且 MBUS 为 0 时，通过左端口将数据总线 DBUS 上的数在 T2 为 1 期间写入由 AR7~AR0 指定的存储器单元。

在读指令操作时，通过存储器右端口读出由 PC7~PC0 指定的存储器单元的内容送 INS7~INS0，当信号 LIR 为 1 时，在 T3 的上升沿写入指令寄存器 IR。

数据开关 SD7~SD0 上的数在 SBUS 为 1 时送到数据总线 DBUS 上，用于给寄存器 R0、R1、R2 和 R3，地址寄存器 AR，程序计数器 PC 设置初值，用于通过存储器左端口向存储器写入测试程序。

数据通路实验中涉及到的信号如下：

M、S3、S2、S1、S0	控制 74LS181 的算术逻辑运算类型。
CIN	低位 74LS181 的进位输入。
SEL3 (RD1)	选择送 ALU 的 A 端口的寄存器和被写入的寄存器。
SEL2 (RD0)	
SEL1 (RS1)	
SEL0 (RS0)	选择送往 ALU 的 B 端口的寄存器。
DRW	=1 时，在 T3 上升沿对 RD1、RD0 选中的寄存器进行写操作，将数据总线 DBUS 上的数 D7~D0 写入选定的寄存器。
ABUS	=1 时，将运算结果送数据总线 DBUS， =0 时，禁止运算结果送数据总线 DBUS。
SBUS	=1 时，将运算结果送数据总线 DBUS， =0 时，禁止运算结果送数据总线 DBUS。
A7~A0	送往 ALU 的 A 端口的数。
B7~B0	送往 ALU 的 B 端口的数。
D7~D0	数据总线 DBUS 上的 8 位数。
MBUS	=1 时，将双端口 RAM 的左端口数据送到数据总线 DBUS。
MEMW	=1 时，在 T2 为 1 期间将数据总线 DBUS 上的 D7~D0 写入双端口 RAM，写入的存储器单元由 AR7~AR0 指定。
LPC	=1 时，在 T3 的上升沿，将数据总线 DBUS 上的 D7~D0 写入程序计数器 PC。
PCINC	=1 时，在 T3 的上升沿 PC 加 1。
LAR	=1 时，在 T3 的上升沿，将数据总线 DBUS 上的 D7~D0 写入地址寄存器 AR。
ARINC	=1 时，在 T3 的上升沿，AR 加 1。
SBUS	=1 时，数据开关 SD7~SD0 的数送数据总线 DBUS。
AR7~AR0	双端口 RAM 左端口存储器地址。
PC7~PC0	双端口 RAM 右端口存储器地址。
INS7~INS0	从双端口 RAM 右端口读出的指令，本实验中作为数据使用。
SETCTL	=1 时，实验系统处于实验台状态。 =0 时，实验系统处于运行程序状态。

上述信号都有对应的指示灯。当指示灯灯亮时，表示对应的信号为 1；当指示灯不亮时，对应的信号为 0。实验过程中，对每一个实验步骤，都要记录上述信号的值。另外 $\mu A5 \sim \mu A0$ 指示灯指示当前微地址。

五、实验任务

1. 将数 75H 写到寄存器 R0，数 28H 写道寄存器 R1，数 89H 写到寄存器 R2，数 32H 写到寄存器 R3。
2. 将寄存器 R0 中的数写入存储器 20H 单元，将寄存器 R1 中的数写入存储器 21H 单元，将寄存器 R2 中的数写入存储器 22H 单元，将寄存器 R3 中的数写入存储器 23H 单元。
3. 从存储器 20H 单元读出数到存储器 R3，从存储器 21H 单元读出数到存储器 R2，从存储器 21H 单元读出数到存储器 R1，从存储器 23H 单元读出数到存储器

R0。

4. 显示 4 个寄存器 R0、R1、R2、R3 的值，检查数据传送是否正确。

六、实验步骤

1. 实验准备

将控制器转换开关拨到微程序位置，将编程开关设置为正常位置。打开电源。

2. 进行数据通路实验

(1) 设置数据通路实验模式

首先将“控制转换”开关拨到最下方位置既“微程序”灯亮。按复位按钮 CLR，使 TEC-8 实验系统复位。指示灯 $\mu A5 \sim \mu A0$ 显示 00H。将操作模式开关设置为 SWC=1、SWB=1、SWA=1，准备进入数据通路实验。

按一次 QD 按钮，进入数据通路实验。

(2) 将数 75H 写到寄存器 R0、数 28H 写到 R1、数 89H 写到 R2、数 32H 写到 R3。指示灯 $\mu A5 \sim \mu A0$ 显示 0FH。在数据开关 SD7~SD0 上设置数 75H。在数据总线 DBUS 指示灯 D7~D0 上可以看到数设置得正确不正确，发现错误需及时改正。数设置正确后，按一次 QD 按钮，将 SD7~SD0 上的数写入寄存器 R0，进入下一步。

依照写 R0 的方式，在指示灯 $\mu A5 \sim \mu A0$ 显示 32H 时，在指示灯 B7~B0 观测寄存器 R0 的值，将数 28H 写入 R1；在指示灯 $\mu A5 \sim \mu A0$ 显示 33H 时，在指示灯 B7~B0 上观测 R1 的值，将数 89H 写入 R2；在指示灯 $\mu A5 \sim \mu A0$ 显示 34H 时，在指示灯 B7~B0 上观测 R2 的值，将数 32H 写入 R3。

(3) 设置存储器地址 AR 和程序计数器 PC

指示灯 $\mu A5 \sim \mu A0$ 显示 35H。此时指示灯 B7~B0 显示寄存器 R3 的值。在数据开关 SD7~SD0 上设置地址 20H。在数据总线 DBUS 指示灯 D7~D0 上可以看到地址设置得正确不正确。地址设置正确后，按一次 QD 按钮，将 SD7~SD0 上的地址写入地址寄存器 AR 和程序计数器 PC，进入下一步。

(4) 将寄存器 R0、R1、R2、R3 中的数依次写入存储器 20H、21H、22H 和 23H 单元。

指示灯 $\mu A5 \sim \mu A0$ 显示 36H。此时指示灯 AR7~AR0 和 PC7~PC0 分别显示出存储器左、右两个端口的存储器地址。指示灯 A7~A0、B7~B0 和 D7~D0 都显示寄存器 R0 的值。按一次 QD 按钮，将 R0 中的数写入存储器 20H 单元，进入下一步。

依照此法，在指示灯 $\mu A5 \sim \mu A0$ 显示 37H 时，在 INS7~INS0 上观测存储器 20H 单元的值，将 R1 中的数写入存储器 21H 单元；在指示灯 $\mu A5 \sim \mu A0$ 显示 38H 时，在 INS7~INS0 上观测存储器 21H 单元的值，将 R2 中的数写入存储器 22H 单元；在指示灯 $\mu A5 \sim \mu A0$ 显示 39H 时，在 INS7~INS0 上观测存储器 22H 单元的值，将 R3 中的数写入存储器 23H 单元。

(5) 重新设置存储器地址 AR 和程序计数器 PC

指示灯 $\mu A5 \sim \mu A0$ 显示 3AH。此时指示灯 PC7~PC0 显示 23H，INS7~INS0 显示存储器 23H 单元中的数。在数据开关 SD7~SD0 上设置地址 20H。按一次 QD 按钮，将地址 20H 写入地址寄存器 AR 和程序计数器 PC，进入下一步。

(6) 将存储器 20H、21H、22H 和 23H 单元中的数依次写入寄存器 R3、R2、R1 和 R0。

指示灯 $\mu A5 \sim \mu A0$ 显示 3BH。此时指示灯 AR7~AR0 和 PC7~PC0 显示 20H，指示灯 D7~D0 和 INS7~INS0 同时显示存储器 20H 中的数，按一次 QD 按钮，将存储器 20H 单元中的数写入寄存器 R3，进入下一步。

依照此法，在指示灯 $\mu A5 \sim \mu A0$ 显示 3CH 时，在指示灯 B7~B0 上观测 R3 的值，

将存储器 21H 单元中的数写入寄存器 R2；在指示灯 $\mu A5 \sim \mu A0$ 显示 3DH 时，在指示灯 B7~B0 上观测 R2 的值，将存储器 22H 单元中的数写入寄存器 R1；在指示灯 $\mu A5 \sim \mu A0$ 显示 3EH 时，在指示灯 B7~B0 上观测 R1 的值，将存储器 23H 单元中的数写入寄存器 R0。

(7)观测 R0 的值

指示灯 $\mu A5 \sim \mu A0$ 显示 00H。此时指示灯 A7~A0 显示 R0 的值，指示灯 B7~B0 显示 R3 的值。

七、实验要求

1. 做好实验预习，掌握 TEC-8 模型计算机的数据通路及各种操作情况下的数据流动路径和流动方向。

2. 写出实验报告，内容是：

(1)实验目的。

(2)根据实验结果填写表 2.3。

表 2.3 数据通路实验结果表

$\mu A5 \sim \mu A0$	A7~A0	B7~B0	D7~D0	AR	PC	INS7	R0	R1	R2	R3
0FH										
32H										
33H										
34H										
35H										
36H										
37H										
38H										
39H										
3AH										
3BH										
3CH										
3DH										
3EH										
00H										

(3)结合实验现象，在每一实验步骤中，对下述信号所起的作用进行解释：SBUS、MBUS、LPC、PCINC、LAR、ARINC、MEMW、M、S0、S1、S2、S3、CIN、ABUS、SEL3、SEL2、SEL1、SEL0、DRW、SBUS。并说明在该步骤中，哪些信号是必需的，哪些信号不是必需的，哪些信号必需采用实验中使用的值，哪些信号可以不采用实验中使用的值。

(4)写出下列操作时，数据的流动路径和流动方向：给寄存器置初值、设置存储器地址、将寄存器中的数写到存储器中，从存储器中读数到寄存器。

八、可探索和研究的问题

1. 如果用 I-cache 和 D-cache 来代替双端口存储器，请提出一种数据通路方案。

2.3.2 实验报告记录:

2.3.3 数据通路实验：独立方式：

首先将“控制转换”开关拨到最中间位置既“独立”灯亮。【操作模式：1111】

名称	K15	K14	K13	K12	K11	K10	K9	K8	备 注
信号名	SBUS	DRW	RD1	RD0	RS1	RS0	MBUS	M	
序列号									
01	1	1							
02	1	1		1					
03	1	1	1						
04	1	1	1	1					
05	1								
06								1	
07						1		1	
08					1			1	
09					1	1		1	
10	1				1	1			
11		1	1	1			1		
12		1	1		1	1	1		
13		1		1	1		1		
14		1				1	1		
15									

名称	K7	K6	K5	K4	K3	K2	K1	K0	SD	备注
信号名	S3	S1	ABUS	MEMW	LAR	ARINC	LPC	PCINC		
序列号										
01									75	
02									28	
03									89	
04									32	
05					1		1		20	
06	1	1	1	1		1				
07	1	1	1	1		1		1		
08	1	1	1	1		1		1		
09			1	1		1		1		
10					1					
11						1				
12						1				
13						1				
14						1				
15										

2.3.4 实验报告记录:

2.4 微程序控制器实验

2.4.1 微程序控制器实验部分

一、实验类型 原理性+设计性+分析性

二、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。

三、实验设备

序列号	名 称	数 量	备 注
1	实验系统	1 台	
2	双踪示波器	1 台	
3	直流万用表	1 块	
4	逻辑测试笔	1 支	在试验箱上

四、实验电路

微程序控制器与硬连线控制器相比，由于其规整性、易于设计以及需要的时序发生器相对简单，在上世纪七、八十年代得到广泛应用。本实验通过一个具体微程序控制器的实现使学生从实践上掌握微程序控制器的一般实现方法，理解控制器在计算机中的作用。

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。

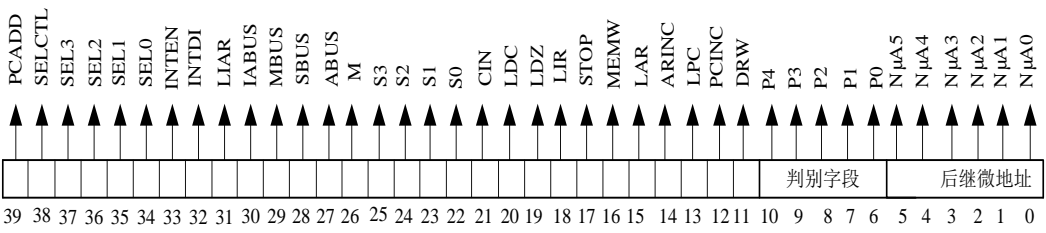


图 2.5 微指令格式

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。

N μ A5~N μ A0	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	=1 时，根据后继微地址 N μ A5~N μ A0 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 N μ A5~N μ A0 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 N μ A5~N μ A0 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 N μ A5~N μ A0 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 N μ A5~N μ A0 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。

由于 TEC-8 模型计算机有微程序控制器和硬连线控制器 2 个控制器，因此微程序控制器产生的控制信号以前缀“A-”标示，以便和硬连线控制器产生的控制信号区分。硬连线控制器产生的控制信号以前缀“B-”标示。

2. 微程序流程图

根据指令系统和控制台功能和数据通路，TEC-8 模型计算机的微程序流程图如图 2.6 所示。图 2.6 中，为了简洁，将许多以“A-”为前缀的信号，省略了前缀。

需要说明的是，图 2.6 中没有包括运算器组成实验、双端口存储器实验和数据通路三部分。这三部分的微程序很简单，微程序都是顺序执行的，根据这三个实验很容易画出。

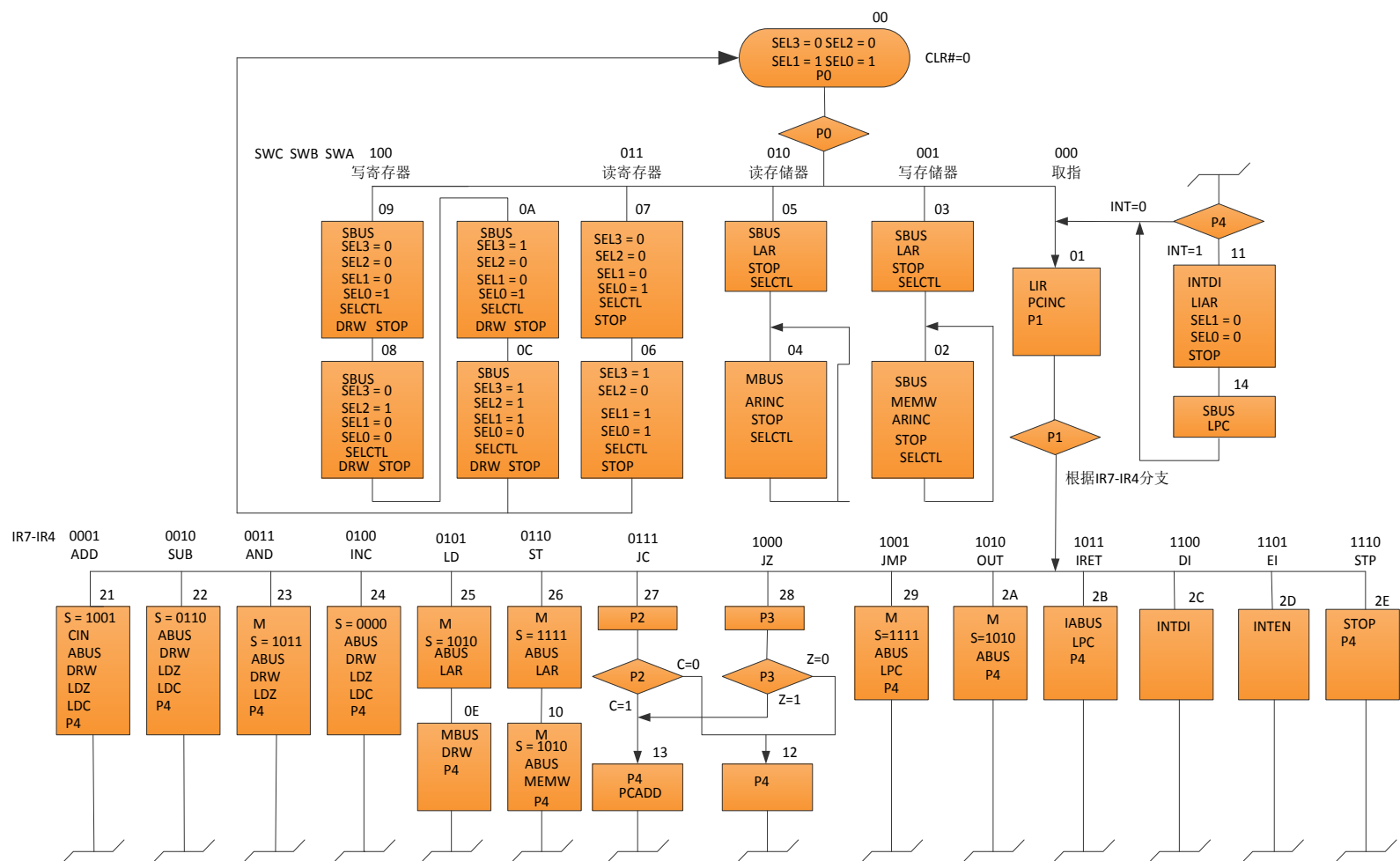


图 2.6 TEC-8 模型计算机微程序流程图

(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例子,但对于理解条件转移的实现方法大有益处。

下面分析根据后继微地址 $N\mu A5 \sim N\mu A0$ 、判别位 $P1$ 和指令操作码如何实现微程序分枝的。

微地址 $N\mu A5 \sim N\mu A0$ 中的微指令是一条功能为取指令的微指令,在 $T3$ 的上升沿,从双端口存储器中取出的指令写入指令寄存器 IR 。在这条微指令中,后继微地址为 $20H$,判别位 $P1$ 为 1、其他判别位均为 0。因此根据微地址转移逻辑,很容易就知道,下一条微指令的微地址是:

$$N\mu A5-T = N\mu A5$$

$$N\mu A4-T = N\mu A4$$

$$N\mu A3-T = N\mu A3 \text{ or } P1 \text{ and } IR7$$

$$N\mu A2-T = N\mu A2 \text{ or } P1 \text{ and } IR6$$

$$N\mu A1-T = N\mu A1 \text{ or } P1 \text{ and } IR5$$

$$N\mu A0-T = N\mu A3 \text{ or } P1 \text{ and } IR4$$

新产生的微地址 $N\mu A5-T \sim N\mu A0-T$ 在 $T3$ 的下降沿写入微地址寄存器 μAR ,实现了图 2.6 微程序流程图图所要求的根据指令操作码进行微程序分枝。

五、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 $\mu A5 \sim \mu A0$ 、判别位 $P4 \sim P0$ 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 $IR7 \sim IR4$,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 $\mu A5 \sim \mu A0$ 、判别位 $P4 \sim P0$ 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。

六、实验步骤

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 $K0 \sim K5$ 连接。通过拨动开关 $K0 \sim K5$,可以对上述信号设置希望的值。

打开电源。

2. 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。

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。

3. 跟踪指令的执行

按复位按钮 CLR 后，设置操作模式开关 SWC=0、SWB=0、SWA=0，按一次 QD 按钮，则进入启动程序运行模式。设置电平开关 K3~K0，使其代表希望的指令操作码 IR7~IR4，按 QD 按钮，跟踪指令的执行。

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。

七、实验要求

1. 认真做好实验的预习，掌握 TEC-8 模型计算机微程序控制器的工作原理。
2. 写出实验报告，内容是：
 - (1)实验目的
 - (2)控制台操作的跟踪过程。写出每一步的微地址 $\mu A5 \sim \mu A0$ 、判别位 P4~P0 和有关控制信号的值。
 - (3)写出这 4 种控制台操作的作用和使用方法。
 - (4)指令的跟踪过程。写出每一步的微地址 $\mu A5 \sim \mu A0$ 、判别位 P4~P0 和有关控制信号的值。
 - (5)写出 TEC-8 模型计算机中的微地址转移逻辑的逻辑表达式。分析它和各种微程序分枝的对应关系。

八、可探索和研究的问题

1. 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，画出这部分的微程序流程图。
2. 你能将图 2.5 中的微指令格式重新设计压缩长度吗？

Case No.	Case Name	Case Type	Case Status	Case Date	Case Time	Case Location	Case Description	Case Details	Case Notes
1	John Doe	Case 1	Open	2023-01-01	10:00	New York	Case 1 Description	Case 1 Details	Case 1 Notes
2	Jane Smith	Case 2	Open	2023-01-02	11:00	Los Angeles	Case 2 Description	Case 2 Details	Case 2 Notes
3	Bob Johnson	Case 3	Open	2023-01-03	12:00	Chicago	Case 3 Description	Case 3 Details	Case 3 Notes
4	Alice Brown	Case 4	Open	2023-01-04	13:00	San Francisco	Case 4 Description	Case 4 Details	Case 4 Notes
5	Charlie Davis	Case 5	Open	2023-01-05	14:00	Seattle	Case 5 Description	Case 5 Details	Case 5 Notes
6	Diana Prince	Case 6	Open	2023-01-06	15:00	Portland	Case 6 Description	Case 6 Details	Case 6 Notes
7	Frank Miller	Case 7	Open	2023-01-07	16:00	San Diego	Case 7 Description	Case 7 Details	Case 7 Notes
8	Grace Lee	Case 8	Open	2023-01-08	17:00	San Jose	Case 8 Description	Case 8 Details	Case 8 Notes
9	Harry Wilson	Case 9	Open	2023-01-09	18:00	San Antonio	Case 9 Description	Case 9 Details	Case 9 Notes
10	Ivy Green	Case 10	Open	2023-01-10	19:00	San Jose	Case 10 Description	Case 10 Details	Case 10 Notes
11	Jack White	Case 11	Open	2023-01-11	20:00	San Jose	Case 11 Description	Case 11 Details	Case 11 Notes
12	Karen Black	Case 12	Open	2023-01-12	21:00	San Jose	Case 12 Description	Case 12 Details	Case 12 Notes
13	Liam Grey	Case 13	Open	2023-01-13	22:00	San Jose	Case 13 Description	Case 13 Details	Case 13 Notes
14	Mia Blue	Case 14	Open	2023-01-14	23:00	San Jose	Case 14 Description	Case 14 Details	Case 14 Notes
15	Noah Red	Case 15	Open	2023-01-15	00:00	San Jose	Case 15 Description	Case 15 Details	Case 15 Notes
16	Olivia Yellow	Case 16	Open	2023-01-16	01:00	San Jose	Case 16 Description	Case 16 Details	Case 16 Notes
17	Peter Purple	Case 17	Open	2023-01-17	02:00	San Jose	Case 17 Description	Case 17 Details	Case 17 Notes
18	Quinn Pink	Case 18	Open	2023-01-18	03:00	San Jose	Case 18 Description	Case 18 Details	Case 18 Notes
19	Rachel Green	Case 19	Open	2023-01-19	04:00	San Jose	Case 19 Description	Case 19 Details	Case 19 Notes
20	Sam Brown	Case 20	Open	2023-01-20	05:00	San Jose	Case 20 Description	Case 20 Details	Case 20 Notes
21	Tina White	Case 21	Open	2023-01-21	06:00	San Jose	Case 21 Description	Case 21 Details	Case 21 Notes
22	Uma Black	Case 22	Open	2023-01-22	07:00	San Jose	Case 22 Description	Case 22 Details	Case 22 Notes
23	Victor Grey	Case 23	Open	2023-01-23	08:00	San Jose	Case 23 Description	Case 23 Details	Case 23 Notes
24	Wendy Blue	Case 24	Open	2023-01-24	09:00	San Jose	Case 24 Description	Case 24 Details	Case 24 Notes
25	Xavier Yellow	Case 25	Open	2023-01-25	10:00	San Jose	Case 25 Description	Case 25 Details	Case 25 Notes
26	Yara Purple	Case 26	Open	2023-01-26	11:00	San Jose	Case 26 Description	Case 26 Details	Case 26 Notes
27	Zoe Pink	Case 27	Open	2023-01-27	12:00	San Jose	Case 27 Description	Case 27 Details	Case 27 Notes
28	Adam Green	Case 28	Open	2023-01-28	13:00	San Jose	Case 28 Description	Case 28 Details	Case 28 Notes
29	Bella White	Case 29	Open	2023-01-29	14:00	San Jose	Case 29 Description	Case 29 Details	Case 29 Notes
30	Carl Black	Case 30	Open	2023-01-30	15:00	San Jose	Case 30 Description	Case 30 Details	Case 30 Notes
31	Dora Grey	Case 31	Open	2023-01-31	16:00	San Jose	Case 31 Description	Case 31 Details	Case 31 Notes
32	Ethan Blue	Case 32	Open	2023-02-01	17:00	San Jose	Case 32 Description	Case 32 Details	Case 32 Notes
33	Fiona Yellow	Case 33	Open	2023-02-02	18:00	San Jose	Case 33 Description	Case 33 Details	Case 33 Notes
34	Gavin Purple	Case 34	Open	2023-02-03	19:00	San Jose	Case 34 Description	Case 34 Details	Case 34 Notes
35	Hannah Pink	Case 35	Open	2023-02-04	20:00	San Jose	Case 35 Description	Case 35 Details	Case 35 Notes
36	Ian Green	Case 36	Open	2023-02-05	21:00	San Jose	Case 36 Description	Case 36 Details	Case 36 Notes
37	Jessica White	Case 37	Open	2023-02-06	22:00	San Jose	Case 37 Description	Case 37 Details	Case 37 Notes
38	Kyle Black	Case 38	Open	2023-02-07	23:00	San Jose	Case 38 Description	Case 38 Details	Case 38 Notes
39	Laura Grey	Case 39	Open	2023-02-08	00:00	San Jose	Case 39 Description	Case 39 Details	Case 39 Notes
40	Mason Blue	Case 40	Open	2023-02-09	01:00	San Jose	Case 40 Description	Case 40 Details	Case 40 Notes
41	Natalie Yellow	Case 41	Open	2023-02-10	02:00	San Jose	Case 41 Description	Case 41 Details	Case 41 Notes
42	Oscar Purple	Case 42	Open	2023-02-11	03:00	San Jose	Case 42 Description	Case 42 Details	Case 42 Notes
43	Pamela Pink	Case 43	Open	20					

2.5 CPU 组成与机器指令的执行

2.5.1 CPU 组成与机器指令的执行：微程序控制器方式

一、实验类型 原理性+分析性+设计性

二、实验目的

(1)用微程序控制器控制数据通路，将相应的信号线连接，构成一台能运行测试程序的 CPU。

(2)执行一个简单的程序，掌握机器指令与微指令的关系。

(3)理解计算机如何取出指令、如何执行指令、如何在一条指令执行结束后自动取出下一条指令并执行，牢固建立的计算机整机概念。

三、实验设备

序列号	名 称	数 量	备 注
1	实验系统	1 台	
2	双踪示波器	1 台	
3	直流万用表	1 块	
4	逻辑测试笔	1 支	在试验箱上

四、实验电路

本实验将前面几个实验中的所有电路，包括时序发生器、通用寄存器组、算术逻辑运算部件、存储器、微程序控制器等模块组合在一起，构成一台能够运行程序的简单处理机。数据通路的控制由微程序控制器完成，由微程序解释指令的执行过程，从存储器取出一条指令到执行指令结束的一个指令周期，是由微程序完成的，即一条机器指令对应一个微程序序列。

在本实验中，程序装入到存储器中和给寄存器置初值是在控制台方式下手工完成的，程序执行的结果也需要用控制台操作来检查。TEC-8 模型计算机的控制台操作如下：【控制转换开关放置在最下方，微程序灯亮】

(1)写存储器

写存储器操作用于向存储器中写测试程序和数据。

按复位按钮 CLR，设置 SWC=0、SWB=0、SWA=1。按 QD 按钮一次，控制台指示灯亮，指示灯 $\mu A5 \sim \mu A0$ 显示 03H，进入写存储器操作。在数据开关 SD7~SD0 上设置存储器地址，通过数据总线指示灯 D7~D0 可以检查地址是否正确。按 QD 按钮一次，将存储器地址写入地址寄存器 AR，指示灯 $\mu A5 \sim \mu A0$ 显示 02H，指示灯 AR7~AR0 显示当前存储器地址。在数据开关上设置被写的指令。按 QD 按钮一次，将指令写入存储器。写入指令后，从指示灯 AR7~AR0 上可以看到地址寄存器自动加 1。在数据开关上设置下一条指令，按 QD 按钮一次，将第 2 条指令写入存储器。这样一直继续下去，直到将测试程序全部写入存储器。

(2)读存储器

读存储器操作用于检查程序的执行结果和检查程序是否正确写入到存储器中。

按复位按钮 CLR，设置 SWC=0、SWB=1、SWA=0。按 QD 按钮一次，控制台指示灯亮，指示灯 $\mu A5 \sim \mu A0$ 显示 05H，进入读存储器操作。在数据开关 SD7~SD0 上设置存储器地址，通过指示灯 D7~D0 可以检查地址是否正确。按 QD 按钮一次，指

示灯 AR7~AR0 上显示出当前存储器地址，在指示灯 D7~D0 上显示出指令或数据。再按一次 QD 按钮，则在指示灯 AR7~AR0 上显示出下一个存储器地址，在指示灯 D7~D0 上显示出下一条指令。一直操作下去，直到程序和数据全部检查完毕。

(3)写寄存器

写寄存器操作用于给各通用寄存器置初值。

按复位按钮 CLR，设置 SWC=1、SWB=0、SWA=0。按 QD 按钮一次，控制台指示灯亮，指示灯 $\mu A5 \sim \mu A0$ 显示 09H，进入写寄存器操作。在数据开关 SD7~SD0 上设置 R0 的值，通过指示灯 D7~D0 可以检查地址是否正确，按 QD 按钮，将设置的数写入 R0。指示灯 $\mu A5 \sim \mu A0$ 显示 08H，指示灯 B7~B0 显示 R0 的值，在在数据开关 SD7~SD0 上设置 R1 的值，按 QD 按钮，将设置的数写入 R1。指示灯 $\mu A5 \sim \mu A0$ 显示 0AH，指示灯 B7~B0 显示 R1 的值，在在数据开关 SD7~SD0 上设置 R2 的值，按 QD 按钮，将设置的数写入 R2。指示灯 $\mu A5 \sim \mu A0$ 显示 0CH，指示灯 B7~B0 显示 R2 的值，在在数据开关 SD7~SD0 上设置 R3 的值，按 QD 按钮，将设置的数写入 R3。指示灯 $\mu A5 \sim \mu A0$ 显示 00H，指示灯 A7~A0 显示 R0 的值，指示灯 B7~B0 显示 R3 的值。

(4)读寄存器

读寄存器用于检查程序执行的结果。

按复位按钮 CLR，设置 SWC=0、SWB=1、SWA=1。按 QD 按钮一次，控制台指示灯亮，指示灯 $\mu A5 \sim \mu A0$ 显示 07H，进入读寄存器操作。指示灯 A7~A0 显示 R0 的值，指示灯 B7~B0 显示 R1 的值。按一次 QD 按钮，指示灯 $\mu A5 \sim \mu A0$ 显示 06H，指示灯 A7~A0 显示 R2 的值，指示灯 B7~B0 显示 R3 的值。

(5)启动程序运行

当程序已经写入存储器后，按复位按钮 CLR，使 TEC-6 模型计算机复位，设置 SWC=0、SWB=0、SWA=0，按一次启动按钮 QD，则启动测试程序从地址 00H 运行。如果单拍开关 DP=1，那么每按一次 QD 按钮，执行一条微指令；连续按 QD 按钮，直到测试程序结束。如果单拍开关 DP=0，那么按一次 QD 按钮后，程序一直运行到停机指令 STP 为止。如果程序不以停机指令 STP 结束，则程序将无限运行下去，结果不可预知。

五、实验任务

1. 将下面的程序手工汇编成二进制机器代码并装入存储器。

表 2.4 在预习时完成。表中地址 0FH、10H、11H 中存放的不是指令，而是数。此程序运行前要 R2 的值为 12H，R3 的值为 0FH。

表 2.4 预习时要求完成的手工汇编

地址	指令	机器 16 进制代码
00H	LD R0, [R3]	
01H	INC R3	
02H	LD R1, [R3]	
03H	SUB R0, R1	
04H	JZ 0BH	
05H	ST R0, [R2]	
06H	INC R3	
07H	LD R0, [R3]	
08H	ADD R0, R1	
09H	JC 0CH	
0AH	INC R2	
0BH	ST R2, [R2]	
0CH	AND R0, R1	
0DH	OUT R2	
0EH	STP	
0FH	85H	
10H	23H	
11H	0EFH	
12H	00H	

2. 通过简单的连线构成能够运行程序的 TEC-8 模型计算机。
TEC-8 模型计算机所需的连线很少，只需连接 6 条线，具体连线见实验步骤。
3. 将程序写入寄存器，并且给 R2、R3 置初值，跟踪执行程序，用单拍方式运行一遍，用连续方式运行一遍。用实验台操作检查程序运行结果。

六、实验步骤

1. 实验准备

将控制器转换开关拨到微程序位置，将编程开关设置为正常位置。

将信号 IR4-I、IR5-I、IR6-I、IR7-I、C-I、Z-I 依次通过接线孔与信号 IR4-O、IR5-O、IR6-O、IR7-O、C-O、Z-O 连接。使 TEC-8 模型计算机能够运行程序的整机系统。

打开电源。

2. 在单拍方式下跟踪程序的执行

- (1)通过写存储器操作将程序写入存储器。
- (2)通过读操作将程序逐条读出，检查程序是否正确写入了存储器。
- (3)通过写寄存器操作设置寄存器 R2 为 12H、R3 为 0FH。
- (4)通过读寄存器操作检查设置是否正确。
- (5)将单拍开关 DP 设置为 1，使程序在单微指令下运行。
- (6)按复位按钮 CLR，复位程序计数器 PC 为 00H。将模式开关设置为 SWC=0、SWB=0、SWA=0，准备进入程序运行模式。

(7)按一次 QD 按钮，进入程序运行。每按一次 QD 按钮，执行一条微指令，直到程序结束。在程序执行过程中，记录下列信号的值：PC7~PC0、AR7~AR0、 μ A5~ μ A0、IR7~IR0、A7~A0、B7~B0 和 D7~D0。

(8)通过读寄存器操作检查 4 个寄存器的值并记录。

(9)通过读存储器操作检查存储单元 12H、13H 的值并记录。

3. 在连续方式下运行程序

由于单拍方式下运行程序并没有改变存储器中的程序。因此只要重新设置 R2 为 12H、R3 为 0FH。然后将单拍开关 DP 设置为 0，按复位按钮 CLR 后，将模式开关设置为 SWC=0、SWB=0、SWA=0，准备进入程序运行模式。按一次 QD 按钮，程序自动运行到 STP 指令。通过读寄存器操作检查 4 个寄存器的值并记录。通过读存储器操作检查存储单元 12H、13H 的值并记录。

七、实验要求

1. 认真做好实验的预习，在预习时将程序汇编成机器十六进制代码。

2. 写出实验报告，内容是：

(1)实验目的。

(2)填写表 2.4。

(3)填写表 2.5。

表 2.5 单拍方式下指令执行结果指令执行跟踪结果

指令	μ A	PC	AR	IR	A7~A0	B7~B0	D7~D0

(4)单拍方式和连续方式程序执行后 4 个寄存器的值、寄存器 12、13 单元的值。

(5)对表 2.5 中数据的分析、体会。

(6)结合第 1 条和第 2 条指令的执行，说明计算机中程序的执行过程。

(7)结合程序中条件转移指令的执行过程说明计算机中如何实现条件转移功能。

八、可探索和研究的问题

1. 如果需要全面测试 TEC-8 模型计算机的功能，需要什么样的测试程序？请写出测试程序，并利用测试程序对 TEC-8 模型计算机进行测试。

2.5.2 实验报告记录:

2.6 中断原理实验

2.6.1 中断原理实验部分：微程序控制器方式

一、实验类型 原理性+分析性

二、实验目的

- (1)从硬件、软件结合的角度，模拟单级中断和中断返回的过程；
- (2)通过简单的中断系统，掌握中断控制器、中断向量、中断屏蔽等概念；
- (3)了解微程序控制器与中断控制器协调的基本原理；
- (4)掌握中断子程序和一般子程序的本质区别，掌握中断的突发性和随机性。

三、实验设备

序号	名称	数量	备注
1	实验系统	1 台	
2	双踪示波器	1 台	
3	直流万用表	1 块	
4	逻辑测试笔	1 支	在试验箱上方

四、实验原理

1. TEC-8 模型计算机中的中断机构

TEC-8 模型计算机中有一个简单的单级中断系统，只支持单级中断、单个中断请求，有中断屏蔽功能，旨在说明最基本的工作原理。

TEC-8 模型计算机中有 2 条指令用于允许和屏蔽中断。DI 指令称作关中断指令。此条指令执行后，即使发生中断请求，TEC-8 也不响应中断请求。EI 指令称作开中断指令，此条指令执行后，TEC-8 响应中断。在时序发生器中，设置了一个允许中断触发器 EN_INT，当它为 1 时，允许中断，当它为 0 时，禁止中断发生。复位脉冲 CLR#使 EN_INT 复位为 0。使用 VHDL 语言描述的中断控制器如下：

```
INT_EN_P: process (CLR#, MF, INTEN, INTDI, PULSE, EN_INT)
begin
    if CLR# = '0' then
        EN_INT <= '0';
    elsif MF'event and MF = '1' then
        EN_INT <= INTEN or (EN_INT and (not INTDI));
    end if;
    INT <= EN_INT and PULSE;
end process;
```

在上面的描述中，CLR#是按下复位按钮 CLR 后产生的低电平有效的复位脉冲，MF 是 TEC-8 的主时钟信号，INTEN 是执行 EI 指令产生的允许中断信号，INTDI 是执行 DI 指令产生的禁止中断信号，PULSE 是按下 PULSE 按钮产生的高电平有效的中断请求脉冲信号，INT 是时序发生电路向微程序控制器输出的中断程序执行信号。

为保存中断断点的地址，以便程序被中断后能够返回到原来的地址继续执

行，设置了一个中断地址寄存器 IAR，参看第 1 章中的图 1.2。中断地址寄存器 IAR 是 1 片 74LS374(U44)。当信号 LIAR 为 1 时，在 T3 的上升沿，将 PC 保存在 IAR 中。当信号 IABUS 为 1 时，IABUS 中保存的 PC 送数据总线 DBUS，指示灯显示出中断地址。由于本实验系统只有一个断点寄存器而无堆栈，因此仅支持一级中断而不支持多级中断。

中断向量即中断服务程序的入口地址，本实验系统中由数据开关 SD7~SD0 提供。

2. 中断的检测、执行和返回过程

一条指令的执行由若干条微指令构成。TEC-8 模型计算机中，除指令 EI、DI 外，每条指令执行过程的最后一条微指令都包含判断位 P4，用于判断有无中断发生，参看本章图 2.6。因此在每一条指令执行之后，下一条指令执行之前都要根据中断信号 INT 是否为 1 决定微程序分支。如果信号 INT 为 1，则转微地址 11H，进入中断处理；如果信号 INT 为 0，则转微地址 01H，继续取下一条指令然后执行。

检测到中断信号 INT 后，转到微地址 11H。该微指令产生 INTDI 信号，禁止新的中断发生，产生 LIAR 信号，将程序计数器 PC 的当前值保存在中断地址寄存器(断点寄存器)中，产生 STOP 信号，等待手动设置中断向量。在数据开关 SD7~SD0 上设置好中断地址后，机器将中断向量读到 PC 后，转到中服务程序继续执行。

执行一条指令 IRET，从中断地址返回。该条指令产生 IABUS 信号，将断点地址送数据总线 DBUS，产生信号 LPC，将断点从数据总线装入 PC，恢复被中断的程序。

发生中断时，关中断由硬件负责。而中断现场(包括 4 个寄存器、进位标志 C 和结果为 0 标志 Z)的保存和恢复由中断服务程序完成。中断服务程序的最后两条指令一般是开中断指令 EI 和中断返回指令 IRET。为了保证从中断服务程序能够返回到主程序，EI 指令执行后，不允许立即被中断。因此，EI 指令执行过程中的最后一条微指令中不包含 P4 判别位。

五、实验任务

1. 了解中断每个信号的意义和变化条件，并将表 2.5 中的主程序和表 2.6 中的中断服务程序手工汇编成十六进制机器代码。此项任务在预习中完成。

表 2.5 主程序的机器代码

地址	指令	机器代码
00H	EI	
01H	INC R0	
02H	INC R0	
03H	INC R0	
04H	INC R0	
05H	INC R0	
06H	INC R0	
07H	INC R0	
08H	INC R0	
09H	JMP [R1]	

表 2.6 中断服务程序的机器代码

地址	指令	机器代码
45H	ADD R0, R0	
46H	EI	
46H	IRET	

为了保证此程序能够循环执行，应当将 R1 预先设置为 01H。R0 的初值设置为 0。

2. 将 TEC-8 连接成一个完整的模型计算机。
3. 将主程序和中断服务程序装入存储器，执行 3 遍主程序和中断服务程序。列表记录中断有关信号的变化情况。特别记录好断点和 R0 的值。
4. 将存储器 00H 中的 EI 指令改为 DI，重新运行程序，记录发生的现象。

六、实验步骤

1. 实验准备

将控制器转换开关拨到微程序位置，将编程开关设置为正常位置。

将信号 IR4-I、IR5-I、IR6-I、IR7-I、C-I、Z-I 依次通过接线孔与信号 IR4-O、IR5-O、IR6-O、IR7-O、C-O、Z-O 连接。使 TEC-8 模型计算机能够运行程序的整机系统。

打开电源。

2. 通过控制台写存储器操作，将主程序和中断服务程序写入存储器。

3. 执行 3 遍主程序和中断子程序

(1)通过控制台写寄存器操作将 R0 设置为 00H，将 R1 设置为 01H。

(2)将单拍开关 DP 设置为连续运行方式(DP=0)，按复位按钮 CLR，使 TEC-8 模型计算机复位。按 QD 按钮，启动程序从 00H 开始执行。

(3)按一次 PULSE 按钮，产生一个中断请求信号 PULSE，中断主程序的运行。记录下这时的断点 PC、R0(指示灯 A7~A0 上显示)的值。

(4)将单拍开关 DP 设置为单拍方式(DP=1)，在数据开关上设置中断服务程序的入口地址 45H。按 QD 按钮，一步步执行中断服务程序，直到返回到断点为止。

(5)按照步骤(1)~(4)，再重复做 2 遍。

4. 将存储器 00H 的指令改为 DI，按照步骤 3，重做一遍，记录发生的现象。

七、实验要求

1. 认真做好实验的预习，在预习时将程序汇编成机器十六进制代码。
2. 写出实验报告，内容是：
 - (1)实验目的。
 - (2)填写表 2.5。
 - (3)填写表 2.6。
 - (4)填写表 2.7。

表 2.7 中断原理实验结果

执行程序顺序	PC 断点值	中断时的 R0
第 1 遍		
第 2 遍		
第 3 遍		
第 4 遍		

- (4)分析实验结果，得到什么结论？
- (5)简述 TEC-8 模型计算机的中断机制。

八、可研究和探索的问题

1. 在 TEC-8 模型计算机中，采用的是信号 PULSE 高电平产生中断。如果改为信号 PULSE 的上升沿产生中断，怎么设计时序发生器中的中断机制？提出设计方案。

