

# 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

姓 名 王何佳  
学 号 2023211603  
班 级 2023211804  
邮 箱 624772990@qq.com  
任课教师 苑 洁

2024 年 11 月

(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~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 $\mu A5 \sim \mu A0$ 、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。

## 2. 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~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。

### (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

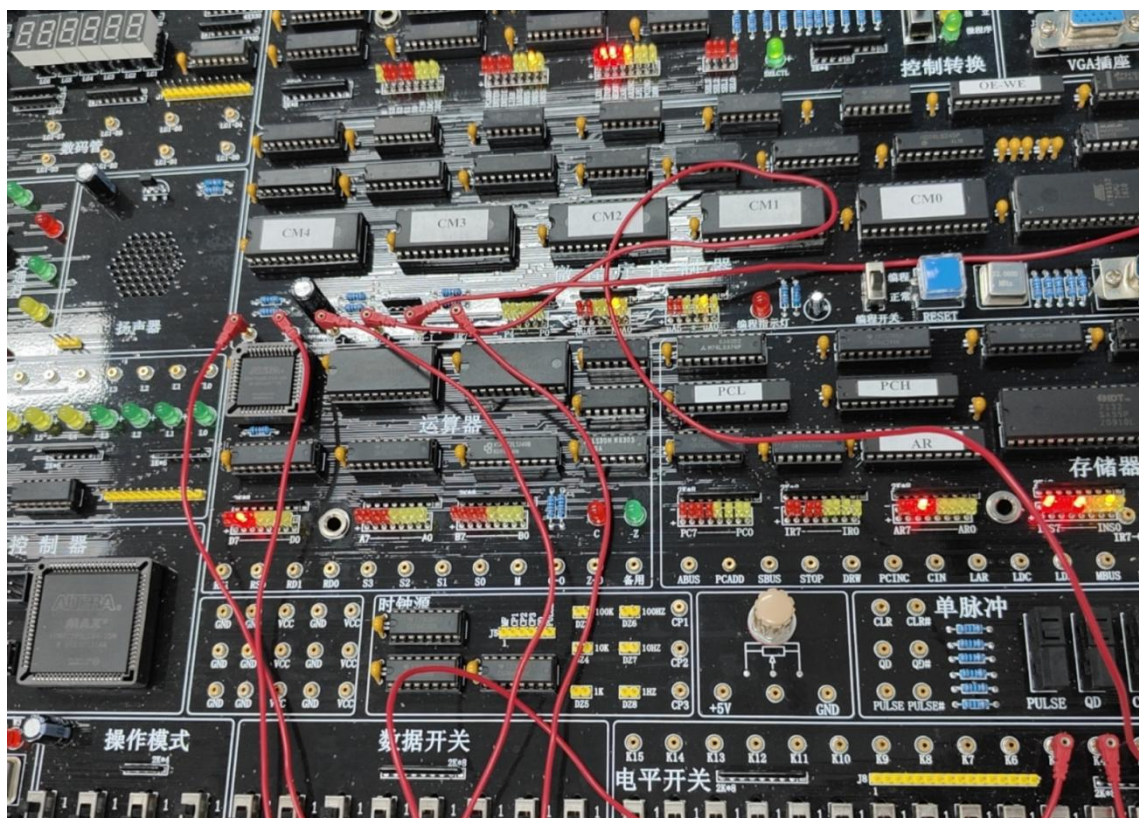
按复位按钮 CLR 后，设置操作模式开关 SWC=0、SWB=0、SWA=0，按一次 QD 按钮，则进入启动程序运行模式。设置电平开关 K3~K0，使其代表希望的指令操作码 IR7~IR4，按 QD 按钮，跟踪指令的执行。按一次复位按钮 CLR 按钮，能够结束本次跟踪操作，开始下一次跟踪操作。

## 三、实验过程

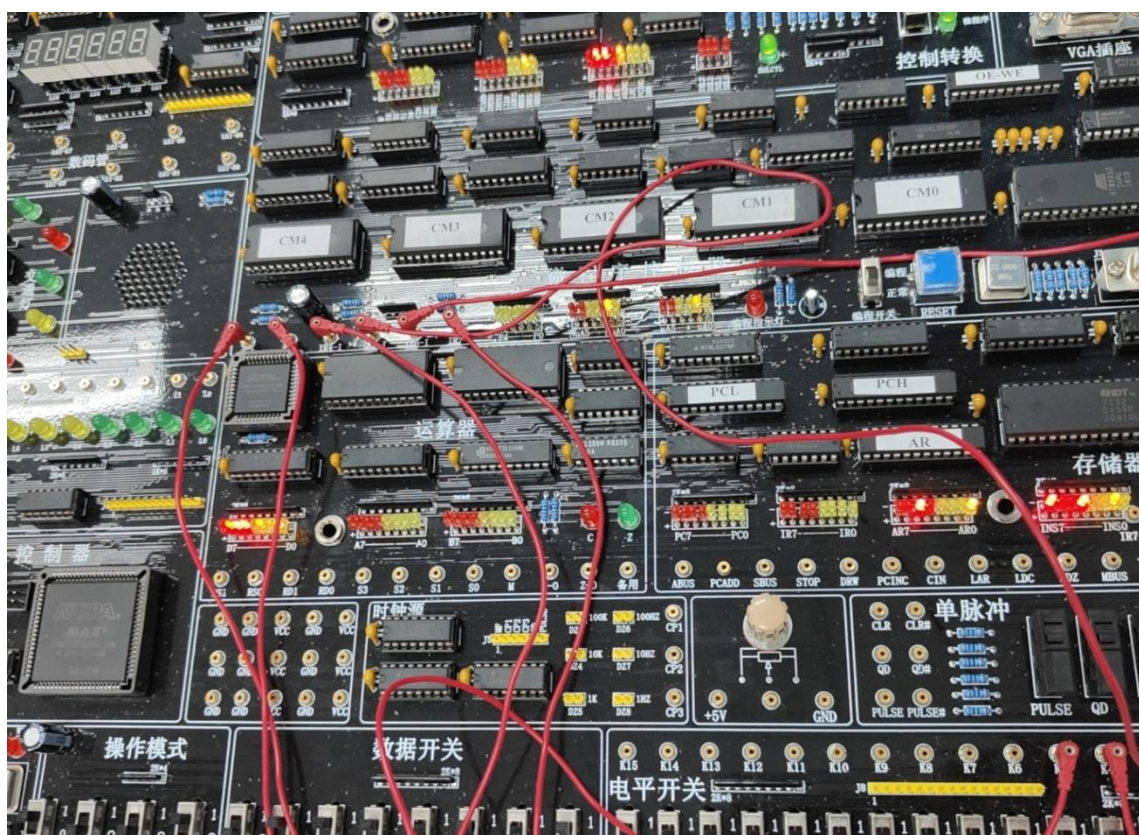
### 1. 接线







存储 75H:



存储其他三个数操作类似。

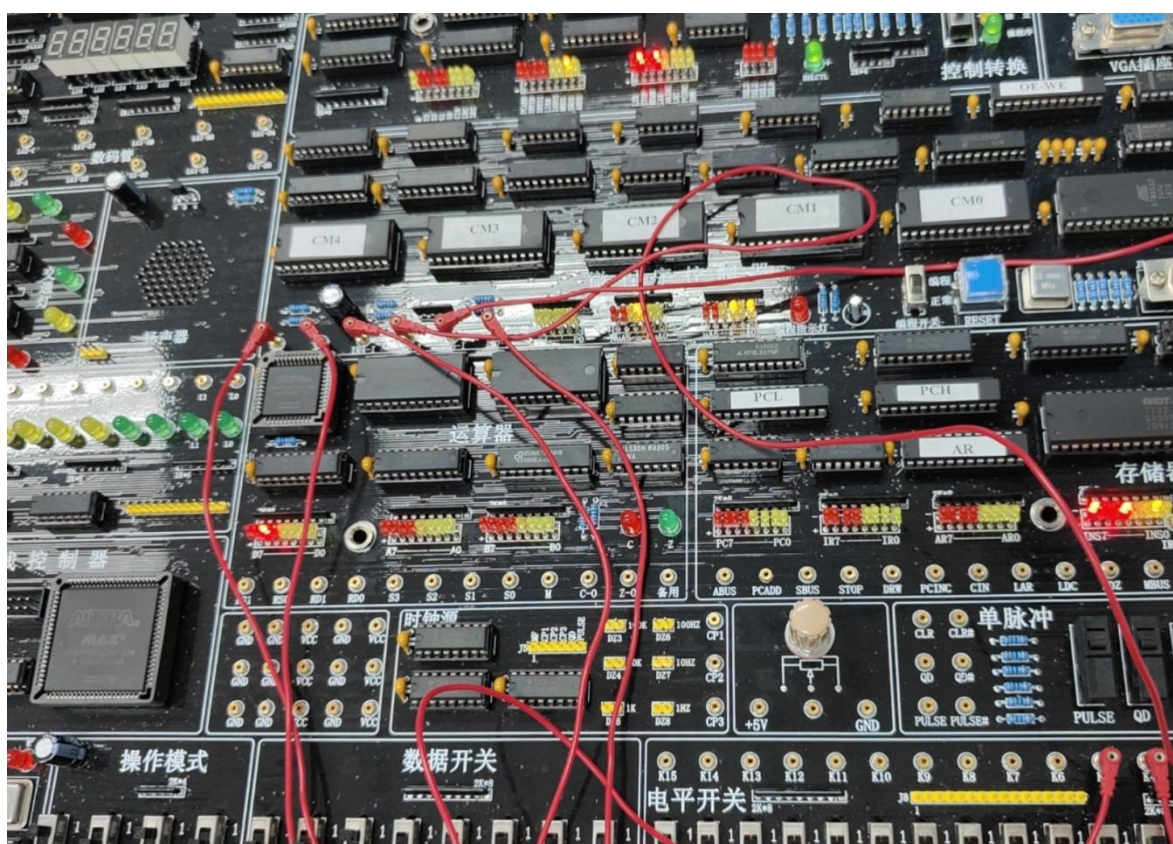


## （2）读存储器

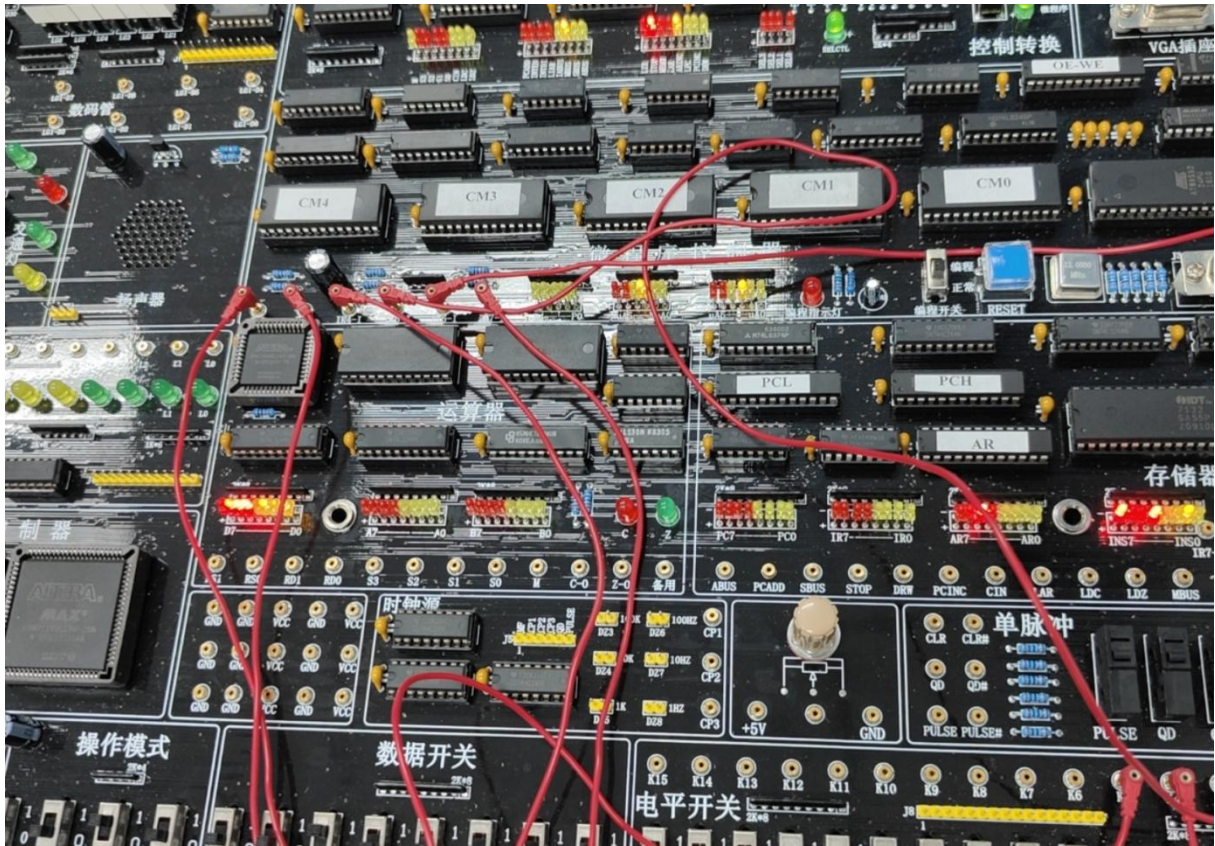
数据记录如下：

操作模式 (SWC-SWA)	功能选择	$\mu A5 \sim \mu A0$	SBUS	STOP	SELCTL	ARINC	LAR	AR	操作
010	读存储器	05H	1	1	1	0	1	00H	设置地址20H
		04H	1	1	1	1	0	20H	读出75H
		04H	1	1	1	1	0	21H	读出32H
		04H	1	1	1	1	0	22H	读出28H
		04H	1	1	1	1	0	23H	读出ABH

设置地址 20H:



按 QD 读出 20H 位置的 75H:



读出其他三个值类似。

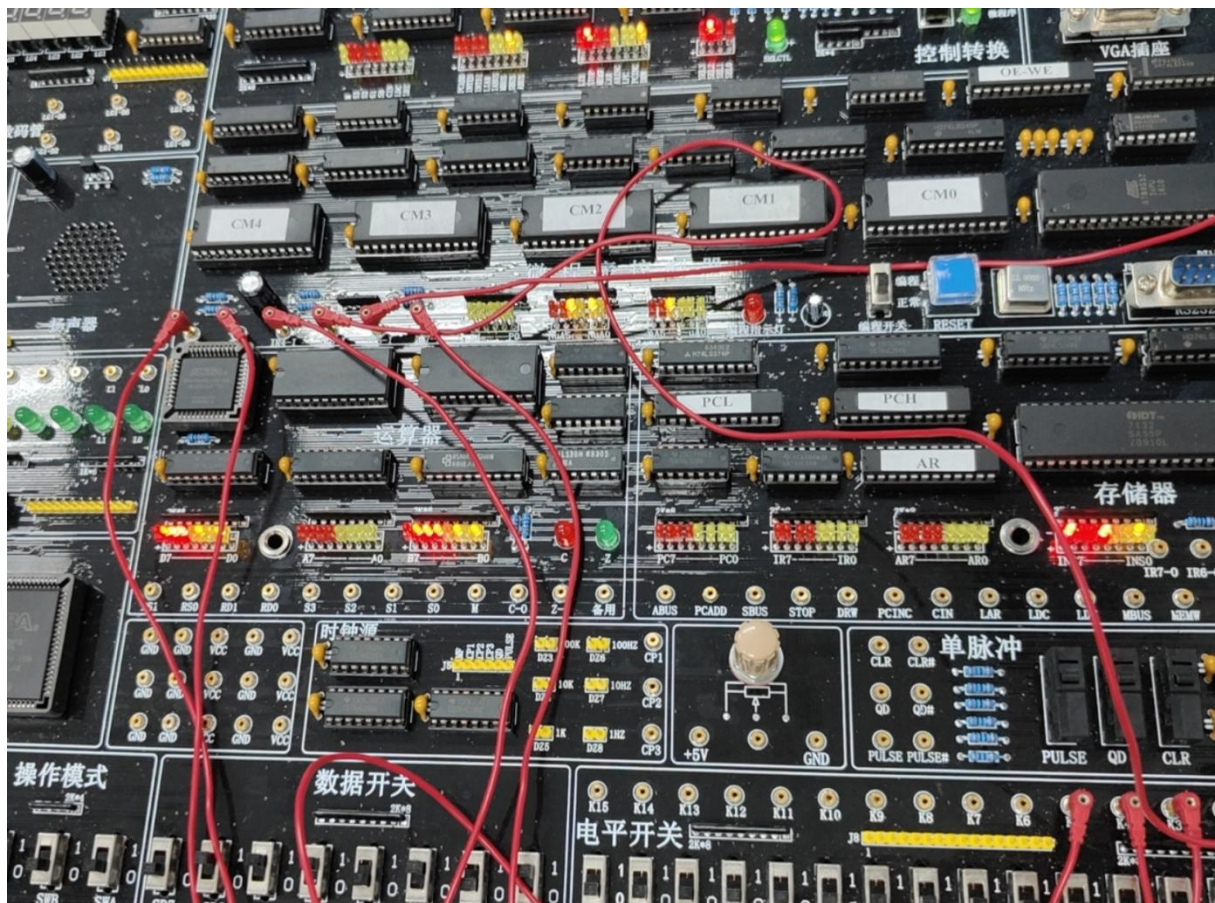
### (3) 写寄存器

数据记录如下：

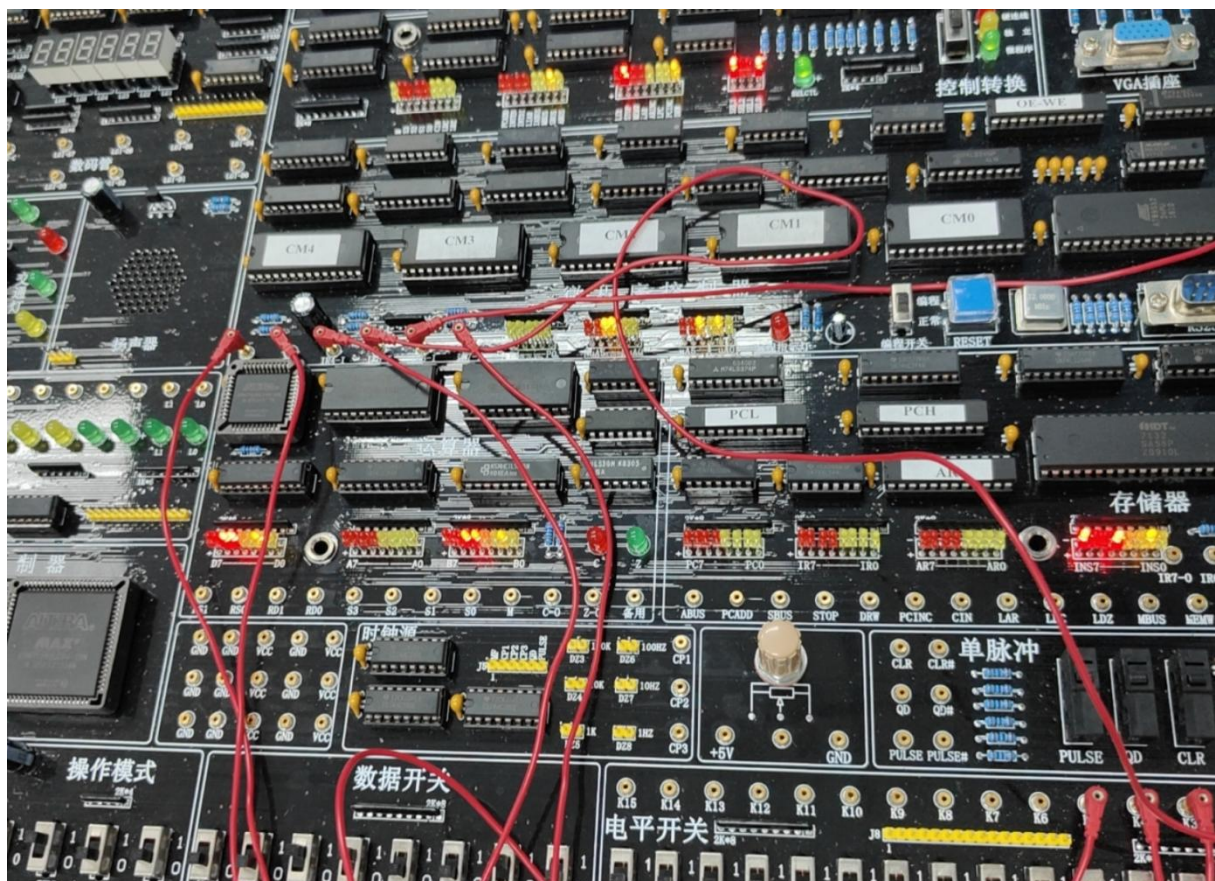
操作模式 (SWC-SWA)	功能选择	$\mu A5 \sim \mu A0$	SBUS	STOP	DRW	R0	R1	R2	R3	操作
100	写寄存器	08H	1	1	1	75H	00H	00H	00H	75H写入R0
		0AH	1	1	1	75H	32H	00H	00H	32H写入R1
		0CH	1	1	1	75H	32H	28H	00H	28H写入R2
		00H	1	1	1	75H	32H	28H	ABH	ABH写入R3

75H 写入 R0:





32H 写入 R1:





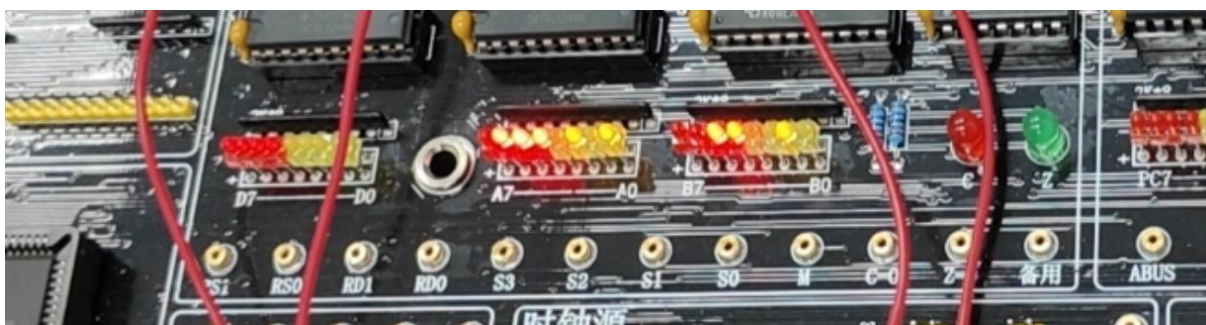
其他两个类似。

#### (4) 读寄存器

数据记录如下：

操作模式 (SWC-SWA)	功能选择	A7~A0	B7~B0	R0	R1	R2	R3	操作
011	读寄存器	75H	32H	75H	32H	28H	ABH	读R0和R1
		28H	ABH	75H	32H	28H	ABH	读R3和R3

在 A7-A0 读出 R0=75H，在 B7-B0 读出 R1=32H。



按 QD，在 A7-A0 读出 R2=28H，在 B7-B0 读出 R3=ABH。



#### (5) 控制台操作的作用和使用方法

根据实验，将控制台操作的作用和使用方法总结如下：

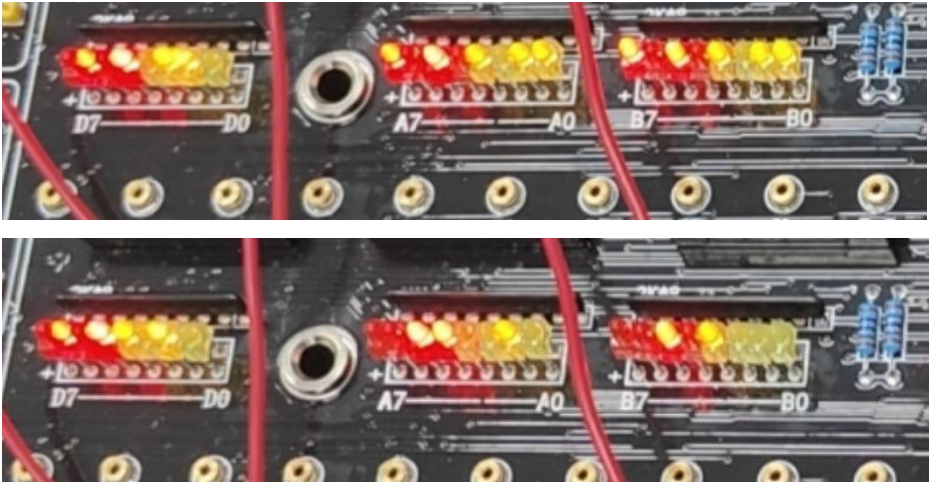
操作模式 (SWC-SWA)	功能选择	操作的作用	使用方法
000	启动程序运行	将程序启动到指定的地址指令处进行运行	控制K5~K0, 将指令指向要运行的存储单元。
001	写存储器	指向指定存储地址，随后存入数据到存储器	先调节D7~D0取想要开始存储的地址，随后存入数据。
010	读存储器	指向指定存储地址，随后读取数据到总线	先调节D7~D0取想要开始读取的地址，随后按地址依次读取数据。
011	读寄存器	按寄存器顺序，读取数据	按QD进行读取
100	写寄存器	按寄存器顺序，将数据存入寄存器	调节D7~D0取数据，按顺序存入寄存器当中

### 3. 指令的跟踪过程

#### 加法 (ADD) 实现

将 IR (K3-K0) 设为 0001，进行加法操作。

操作模式 (SWC-SWA)	功能选择	K5-K0	P4~P0	Rd (A7~A0)	Rs (B7~B0)	D7~D0
000	加法实现	000001B	10000B	ABH	ABH	56H
			10000B	32H	28H	5AH



再按一次 QD，相加后的结果存至 A7~A0。

其他运算可按下表设置 IR（K3-K0），并设置进位（K4）和零标志（K5）的值，执行。

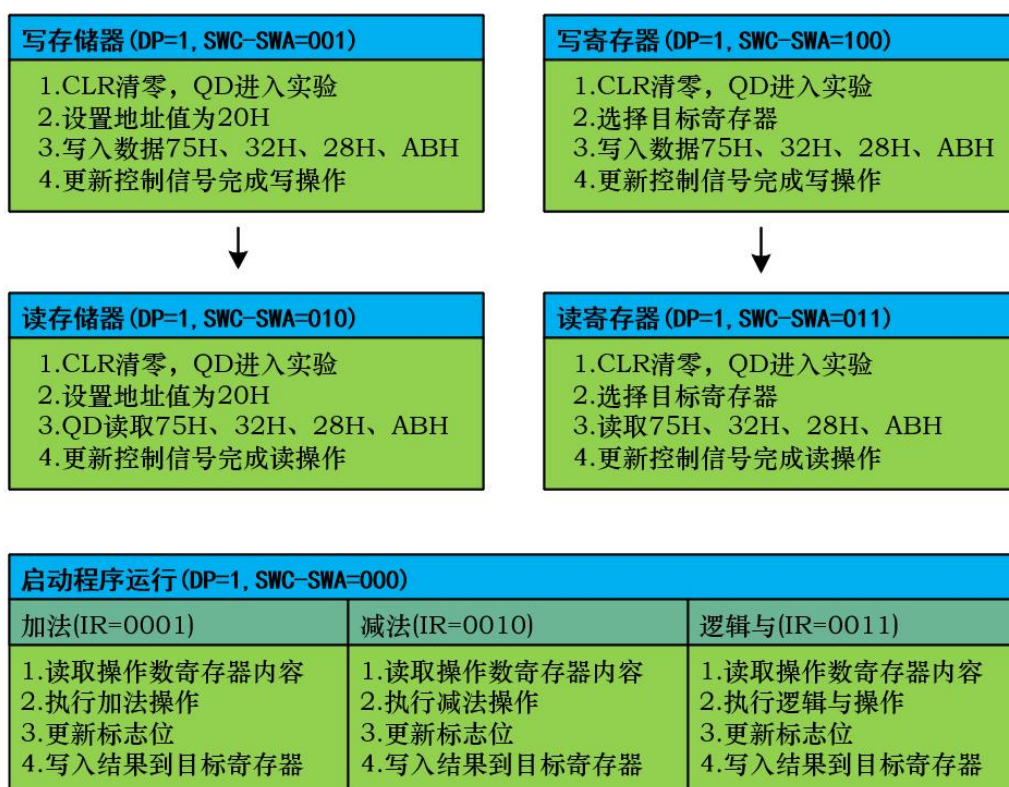
名称	助记符	功 能	指令格式		
			IR(7-4)	IR(3-2)	IR(1-0)
加法	ADD Rd, Rs	$Rd \leftarrow Rd + Rs$	0001	Rd	Rs
减法	SUB Rd, Rs	$Rd \leftarrow Rd - Rs$	0010	Rd	Rs
逻辑与	AND Rd, Rs	$Rd \leftarrow Rd \text{ and } Rs$	0011	Rd	Rs
加 1	INC Rd	$Rd \leftarrow Rd + 1$	0100	Rd	XX
取数	LD Rd, [Rs]	$Rd \leftarrow [Rs]$	0101	Rd	Rs
存数	ST 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



## 四、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。

### 微程序控制器实验流程图



2. 你能将图 2.5 中的微指令格式重新设计压缩长度吗？

通过对微指令格式进行优化，可以显著减少字长。顺序字段原占用 11 位，包括 5 位判别字段和 6 位后继微地址，通过精简分支条件将判别字段压缩至 3 位，并将后继微地址压缩至 5 位，总长度可减至 8 位。控制字段原占用 29 位，通过合并常见微命令组合、信号分组与多路复用，以及动态分配方式，将其压缩至 20~24 位。经过这些优化，微指令总长度由 40 位缩减至 28 位，既节省了存储资源，又提升了效率。

### 重新设计的微指令格式

字段名称	位数	描述
判别字段 $P_2 \sim P_0$	3	表示分支条件，共支持 8 种条件
后继微地址 $N_{\mu}A_4 \sim N_{\mu}A_0$	5	表示下一条微指令地址，共支持 32 个地址
控制字段	20	控制微命令，用于控制数据通路和操作，采用分组与多路复用优化

## 五、实验思考与心得

在实验中，我通过跟踪控制台操作来执行读寄存器、写寄存器、读存储器和写存储器的操作。这一过程让我对计算机的控制台操作有了更直观的认识，也加深了我对计算机硬件操作流程的理解。通过拨动操作模式开关 **SWC**、**SWB**、**SWA** 到特定位置，并按 **QD** 按钮，我能够进入不同的控制台操作模式。这种操作模式的切换，让我对控制台模式开关与控制台操作之间的对应关系有了更清晰的认识。通过设置操作模式开关和电平开关，跟踪指令的执行。这一步骤是实验的核心，它不仅让我理解了指令执行的过程，还让我体会到了微程序控制的灵活性和高效性。通过按 **QD** 按钮，我能够跟踪每一条指令的执行，这种逐条跟踪的方式极大地增强了我对计算机指令执行过程的理解。整个实验过程中，复位按钮 **CLR** 的使用让我能够在每次操作后迅速结束当前跟踪，开始下一次跟踪操作。这不仅提高了实验的效率，也让我更好的掌握实验流程。