

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

实验 3. 数据通路实验

一、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。

二、实验内容

1. 实验原理

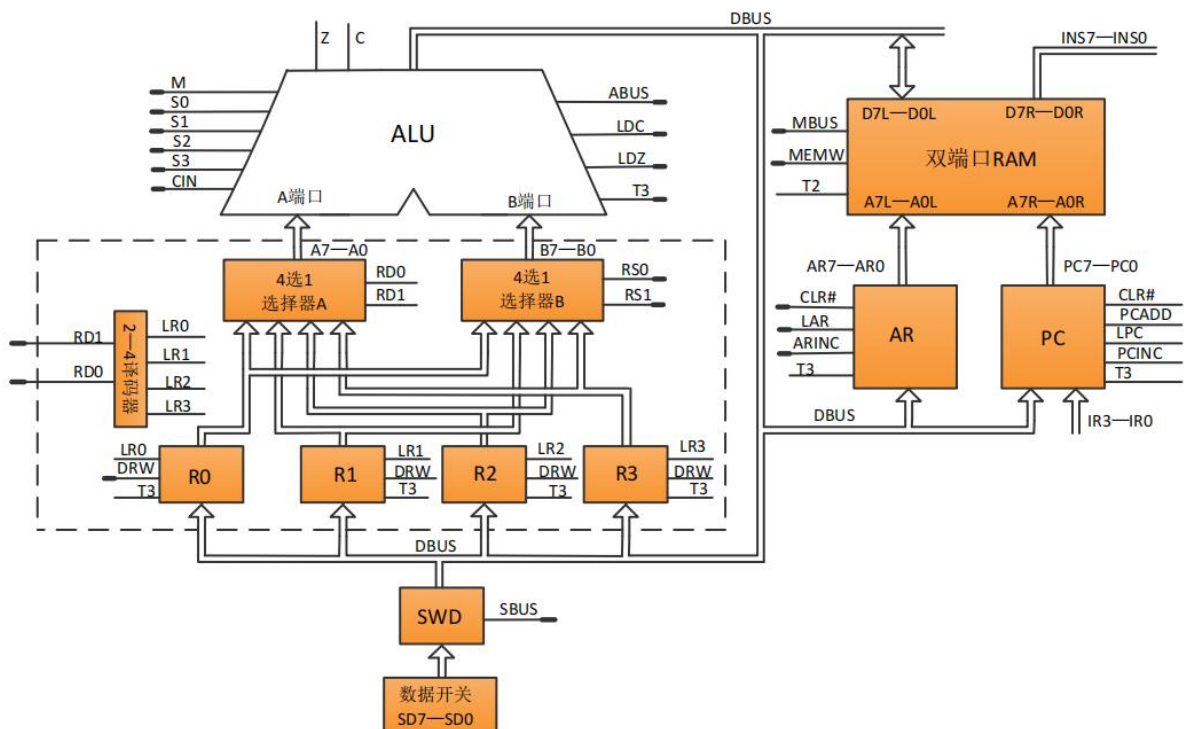


图 2.4 数据通路实验电路图

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、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 的值进行算术逻辑运算，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。

2. 实验任务

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。

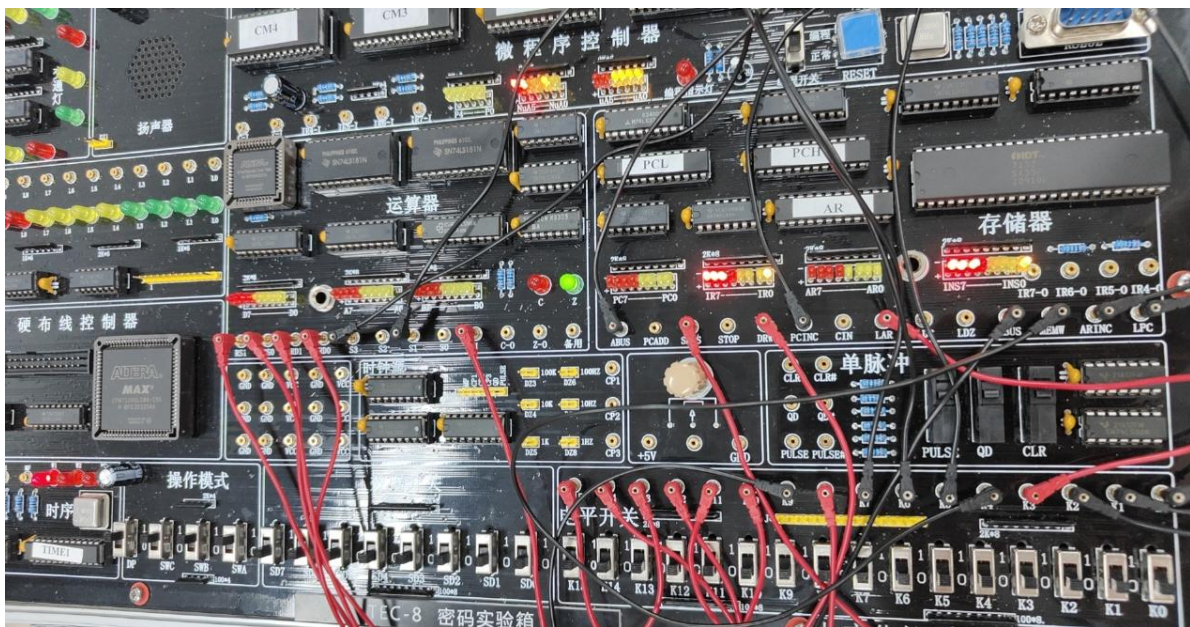
1. 将数 75H 写到寄存器 R0，数 28H 写到寄存器 R1，数 89H 写到寄存器 R2，数 32H 写到寄存器 R3。
2. 将寄存器 R0 中的数写入存储器 20H 单元，将寄存器 R1 中的数写入存储器 21H 单元，将寄存器 R2 中的数写入存储器 22H 单元，将寄存器 R3 中的数写入存储器 23H 单元。
3. 从存储器 20H 单元读出数到寄存器 R3，从存储器 21H 单元读出数到寄存器 R2，从存储器 22H 单元读出数到寄存器 R1，从存储器 23H 单元读出数到寄存器 R0。
4. 显示 4 个寄存器 R0、R1、R2、R3 的值，检查数据传送是否正确。

三、实验过程（独立方式）

1. 按下图接线

K15	K14	K13	K12	K11	K10	K9	K8
SBUS	DRW	RD1	RD0	RS1	RS0	MBUS	M

K7	K6	K5	K4	K3	K2	K1	K0
S3	S1	ABUS	MEMW	LAR	ARINC	LPC	PCINC

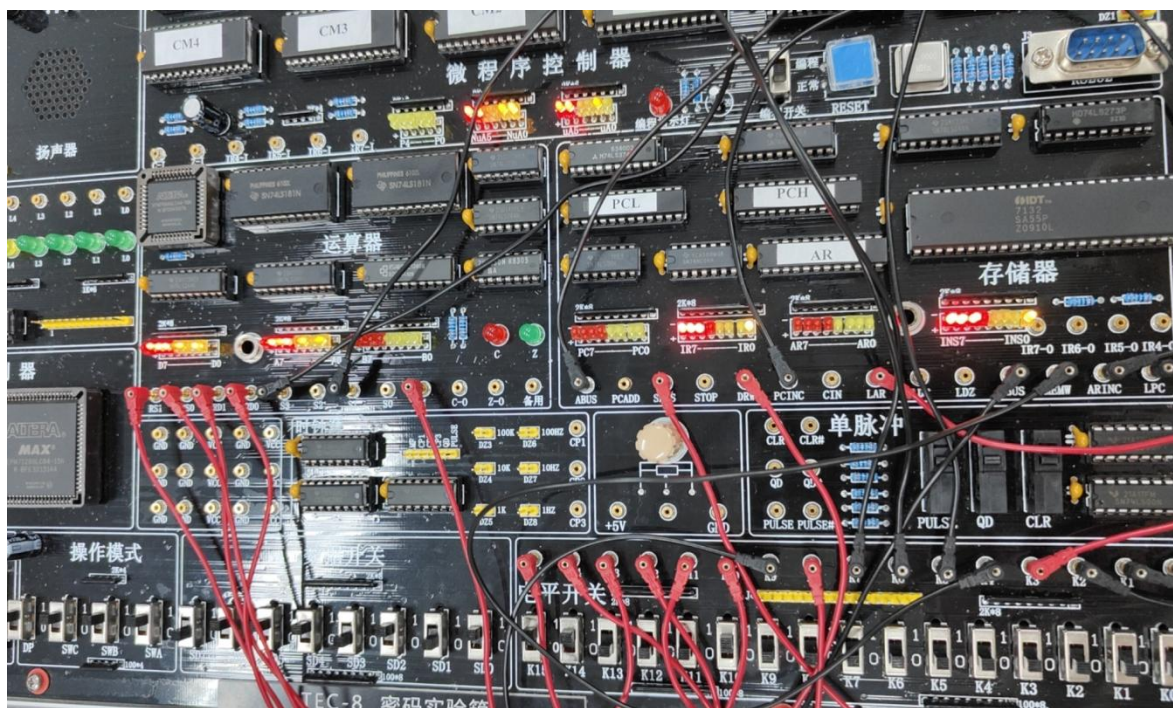


2. 写数据到寄存器

将数 75H 写到寄存器 R0。

K15	K14	K13	K12	K11	K10	K9	K8	K7	K6	K5	K4	K3	K2	K1	K0
1	1				1										

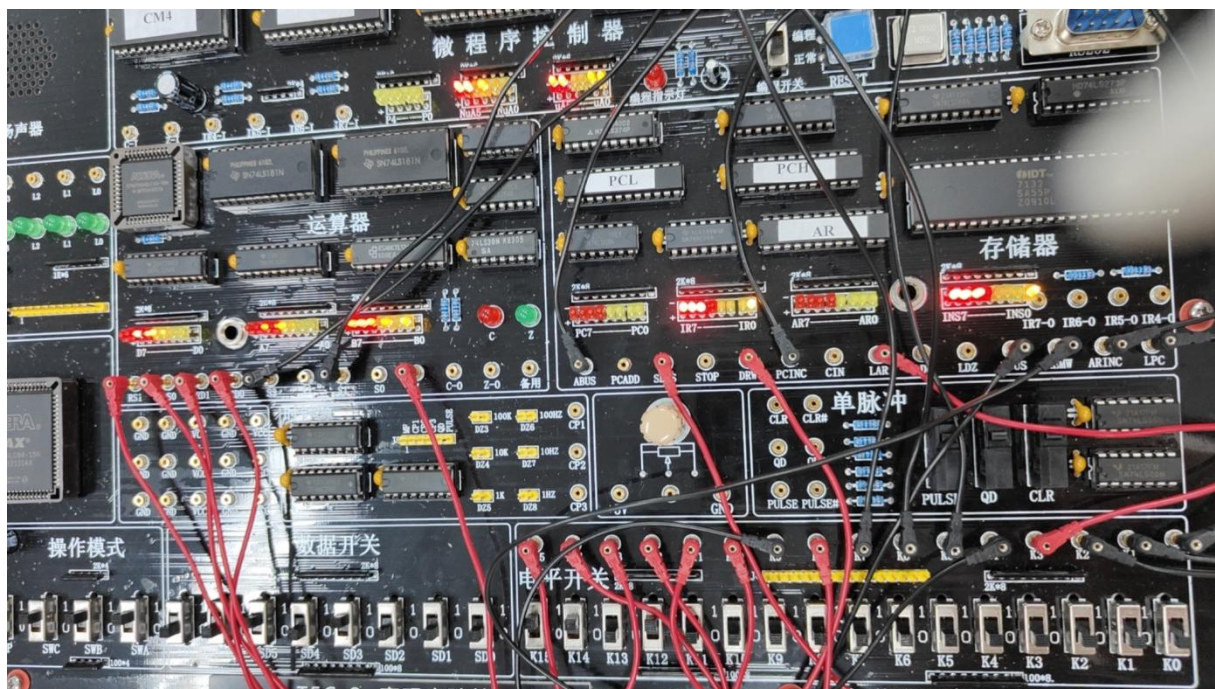
D7	D6	D5	D4	D3	D2	D1	D0
	1	1	1		1		1



将数 28H 写到寄存器 R1。

[illegible]

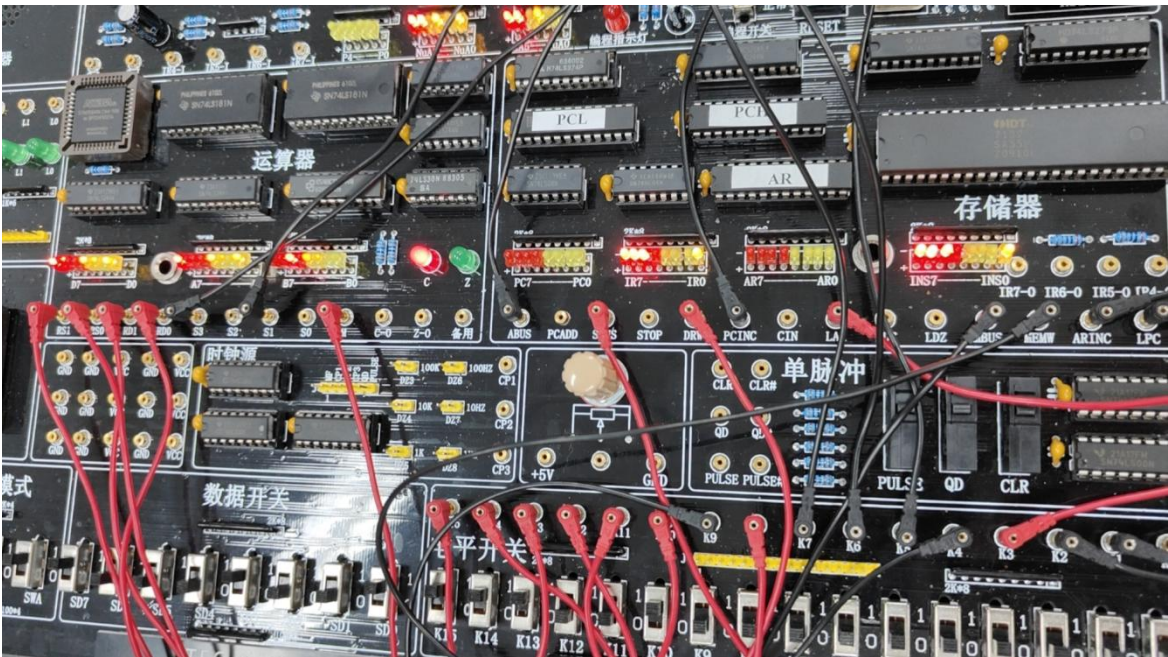
D7	D6	D5	D4	D3	D2	D1	D0
		1		1			



将数 89H 写到寄存器 R2。

[illegible]

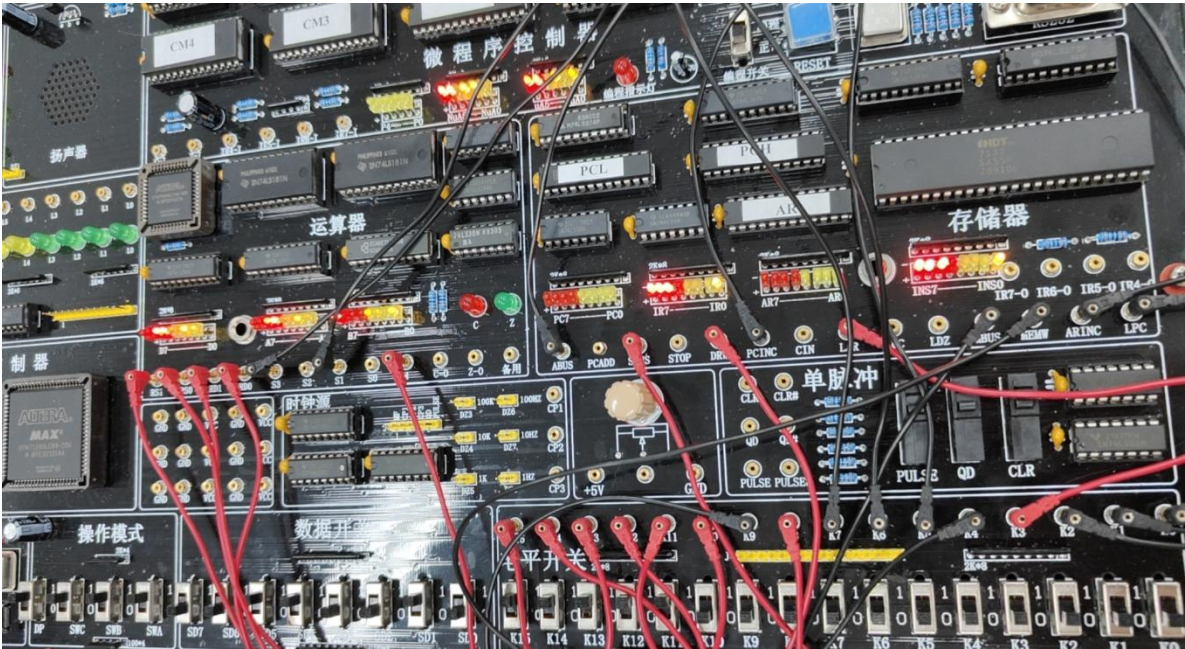
D7	D6	D5	D4	D3	D2	D1	D0
1				1			1



将数 32H 写到寄存器 R3。

K15	K14	K13	K12	K11	K10	K9	K8	K7	K6	K5	K4	K3	K2	K1	K0
1	1	1	1	1											

D7	D6	D5	D4	D3	D2	D1	D0
		1	1			1	

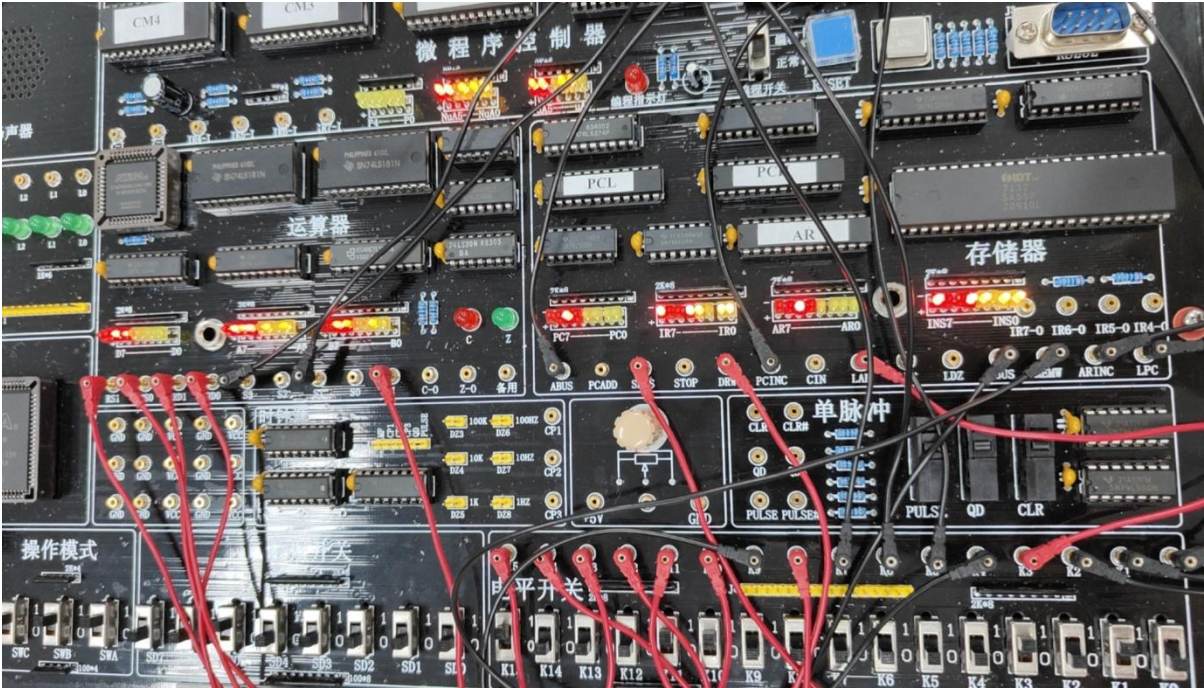


3. 寄存器中的数据写入存储器

设置地址为 20H。

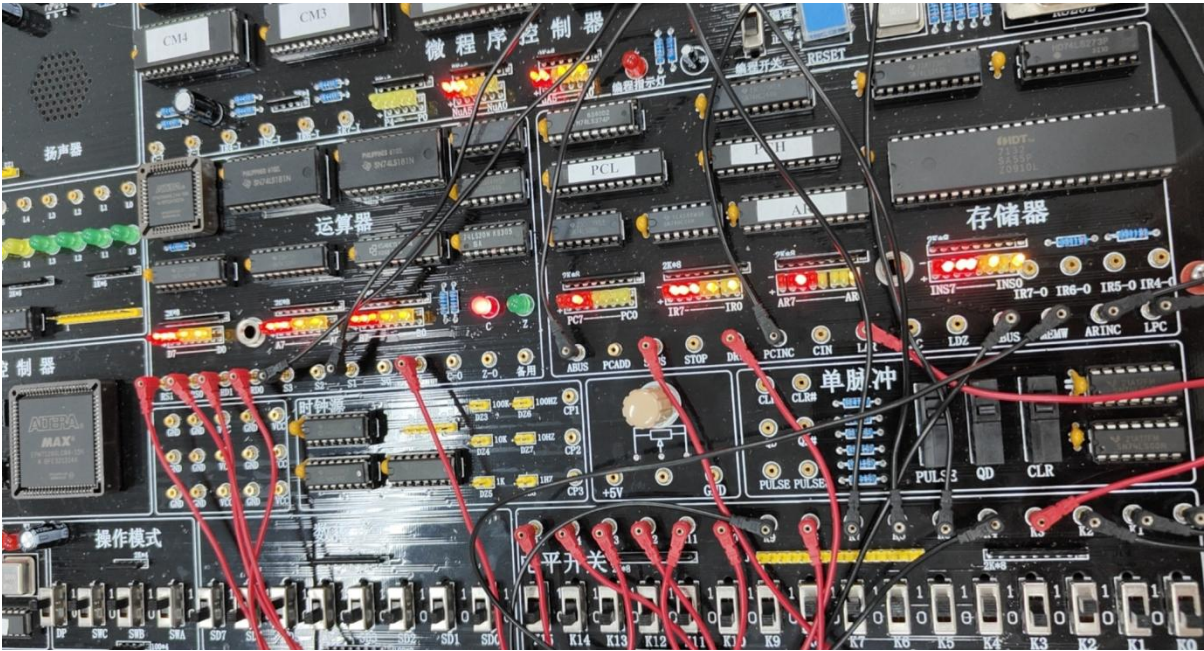
K15	K14	K13	K12	K11	K10	K9	K8	K7	K6	K5	K4	K3	K2	K1	K0
1				1	1							1		1	

D7	D6	D5	D4	D3	D2	D1	D0
		1					



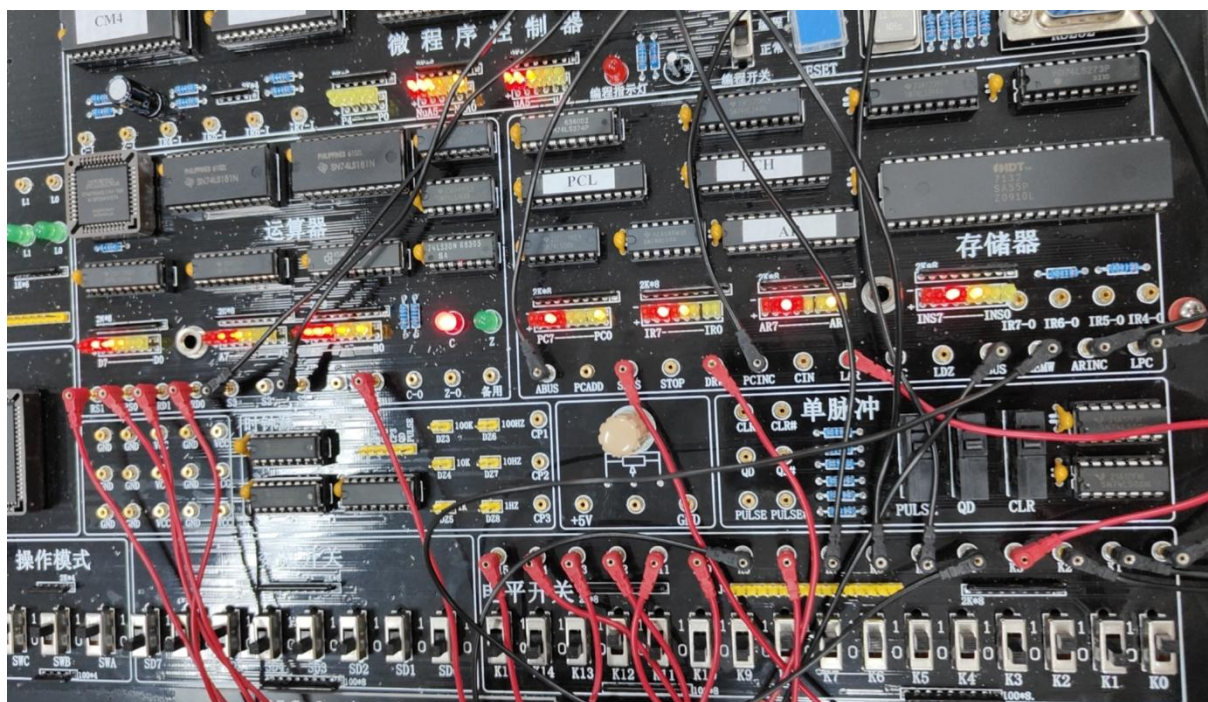
将 R0 中的数写入存储器 20H 单元。

K15	K14	K13	K12	K11	K10	K9	K8	K7	K6	K5	K4	K3	K2	K1	K0
							1	1	1	1	1		1		



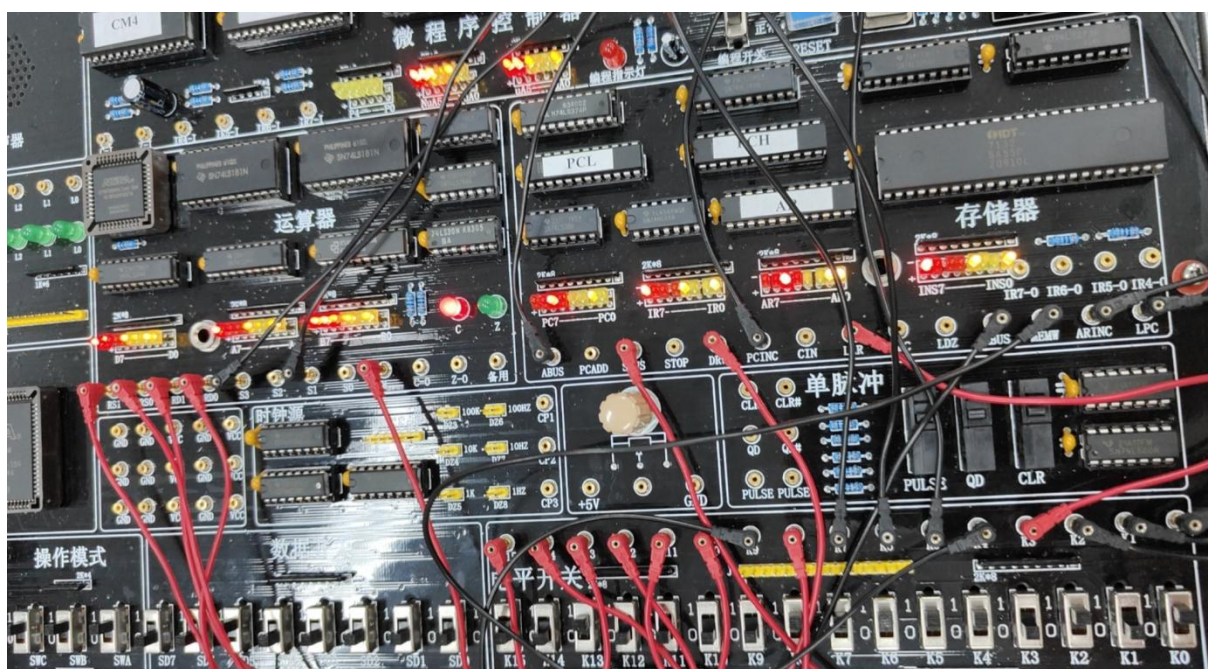
将 R1 中的数写入存储器 21H 单元。

K15	K14	K13	K12	K11	K10	K9	K8	K7	K6	K5	K4	K3	K2	K1	K0
			1				1	1	1	1	1		1		1



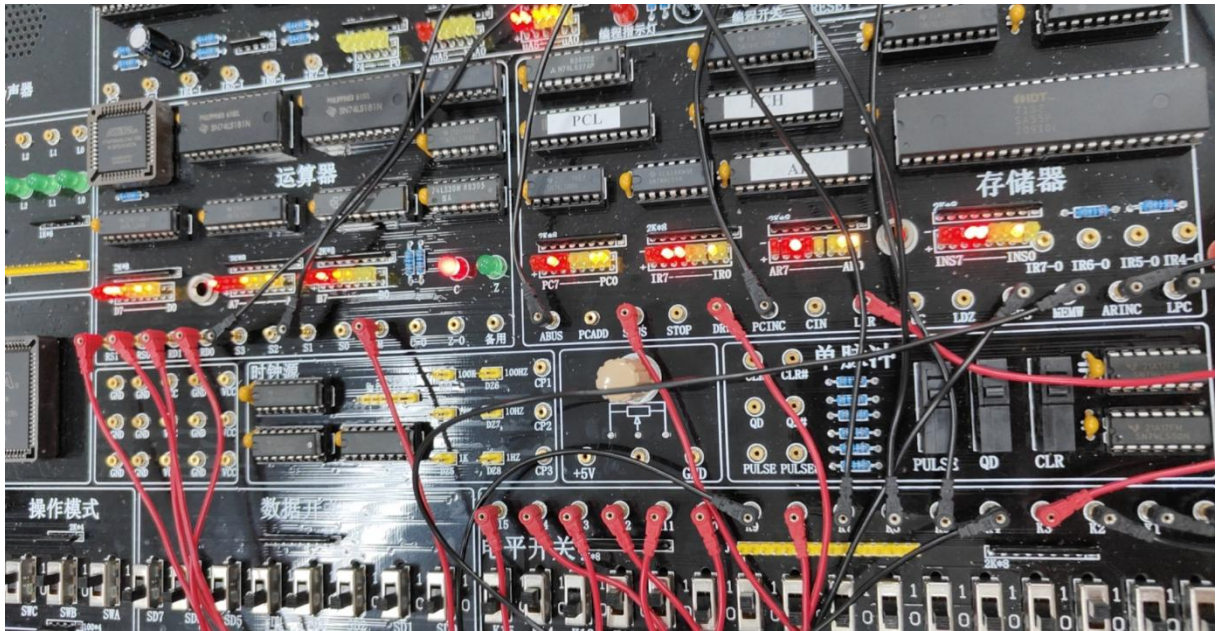
将 R2 中的数写入存储器 22H 单元。

K15	K14	K13	K12	K11	K10	K9	K8	K7	K6	K5	K4	K3	K2	K1	K0
		1					1	1	1	1	1		1		1



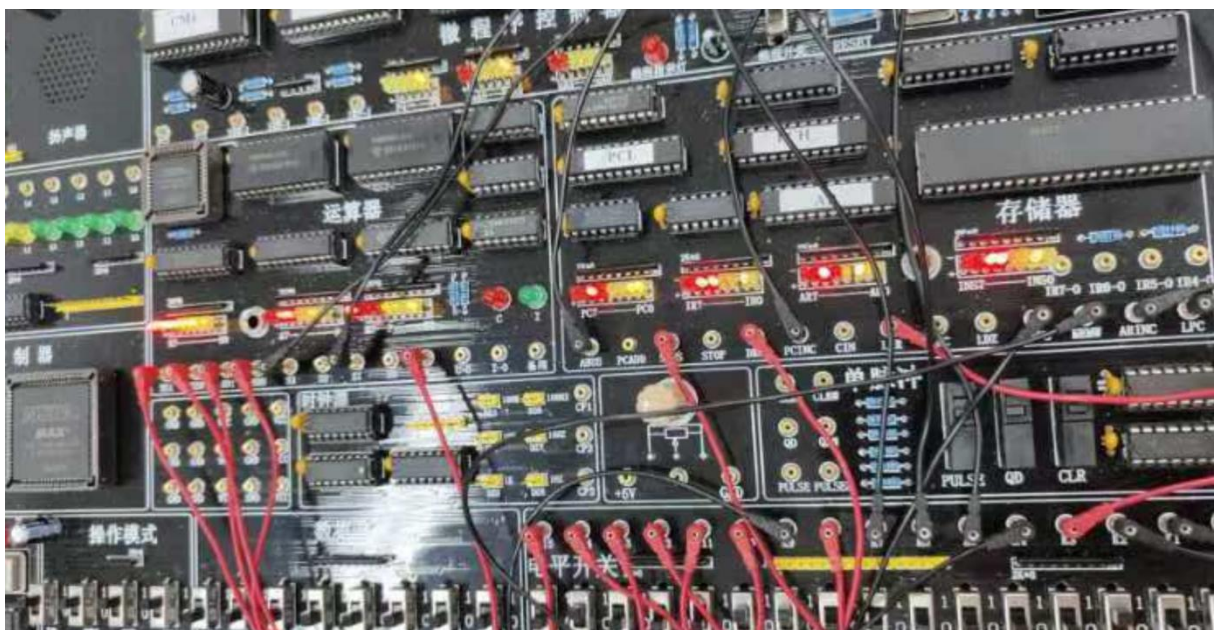
将 R3 中的数写入存储器 23H 单元。

K15	K14	K13	K12	K11	K10	K9	K8	K7	K6	K5	K4	K3	K2	K1	K0
		1	1				1	1	1	1	1		1		1



从存储器 23H 单元读出数到寄存器 R0。

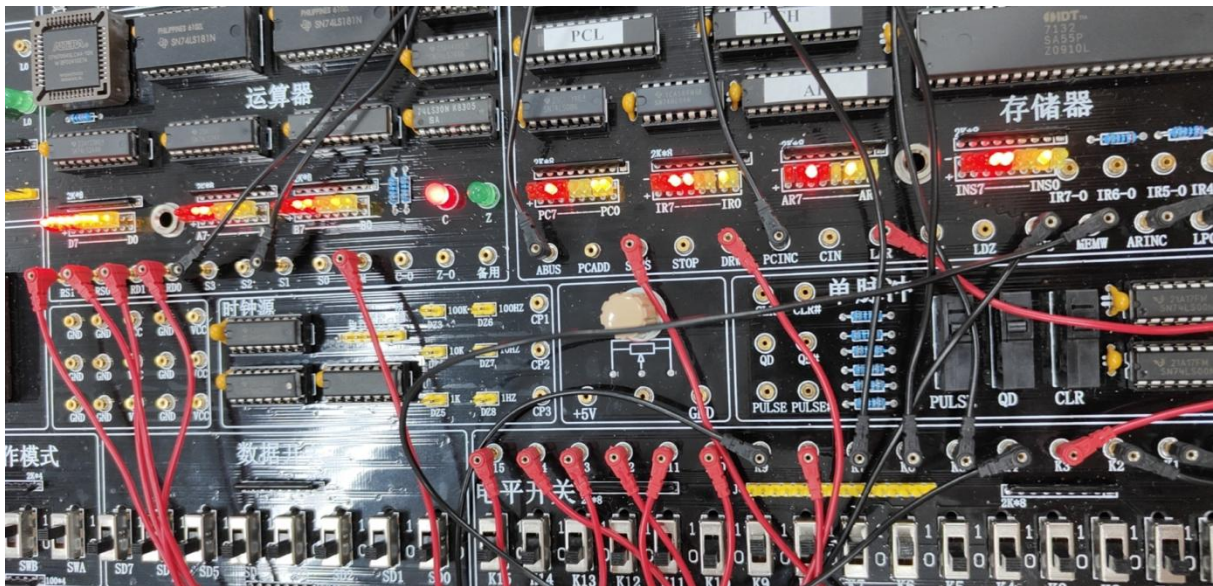
K15	K14	K13	K12	K11	K10	K9	K8	K7	K6	K5	K4	K3	K2	K1	K0
	1				1	1							1		



5. 读出寄存器的值

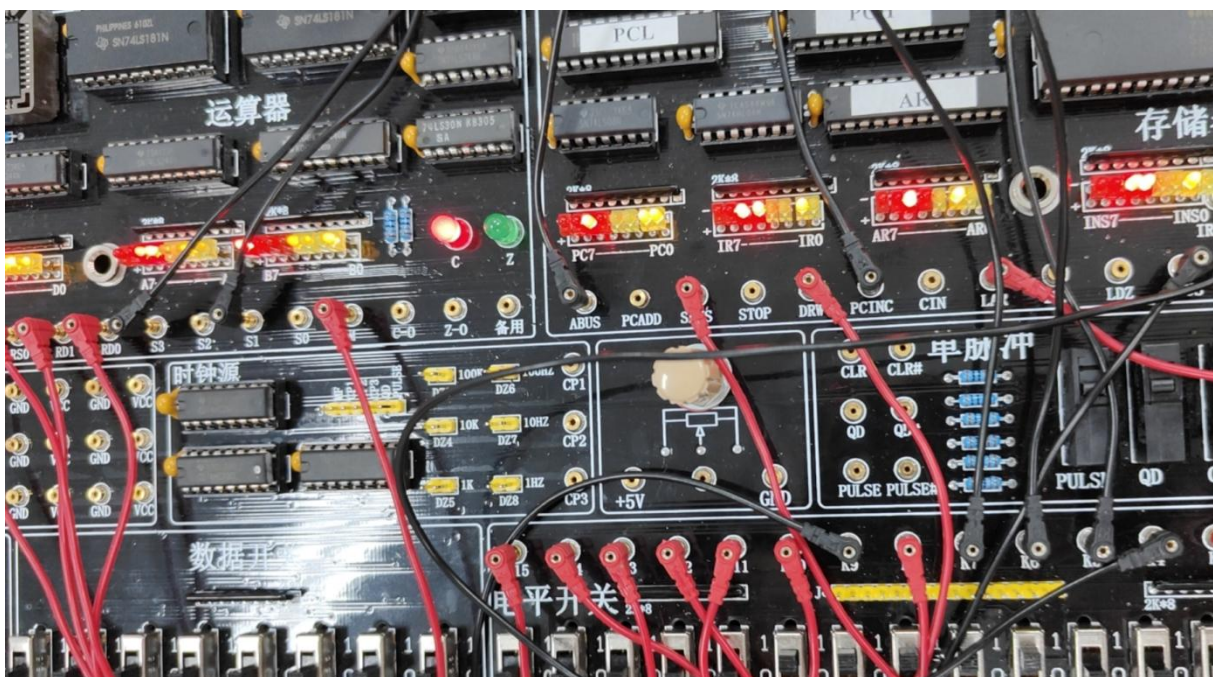
读出 R0=32H (B7~B0)。

K15	K14	K13	K12	K11	K10	K9	K8	K7	K6	K5	K4	K3	K2	K1	K0
						1	1	1	1						



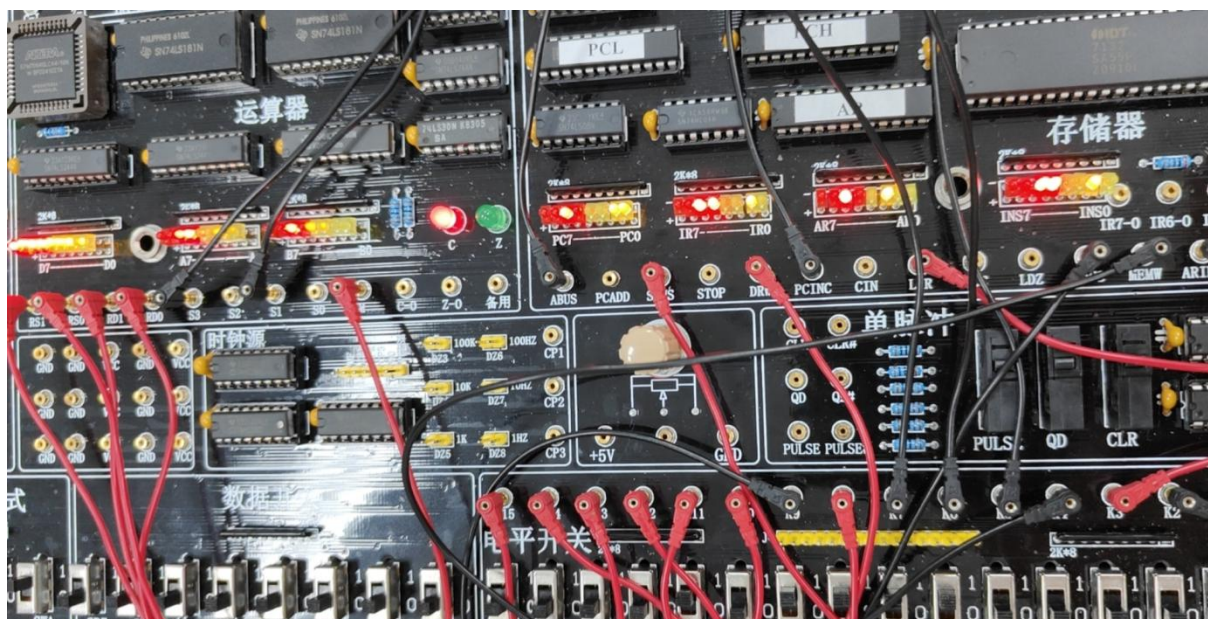
读出 R1=89H。

K15	K14	K13	K12	K11	K10	K9	K8	K7	K6	K5	K4	K3	K2	K1	K0
					1	1	1	1	1						



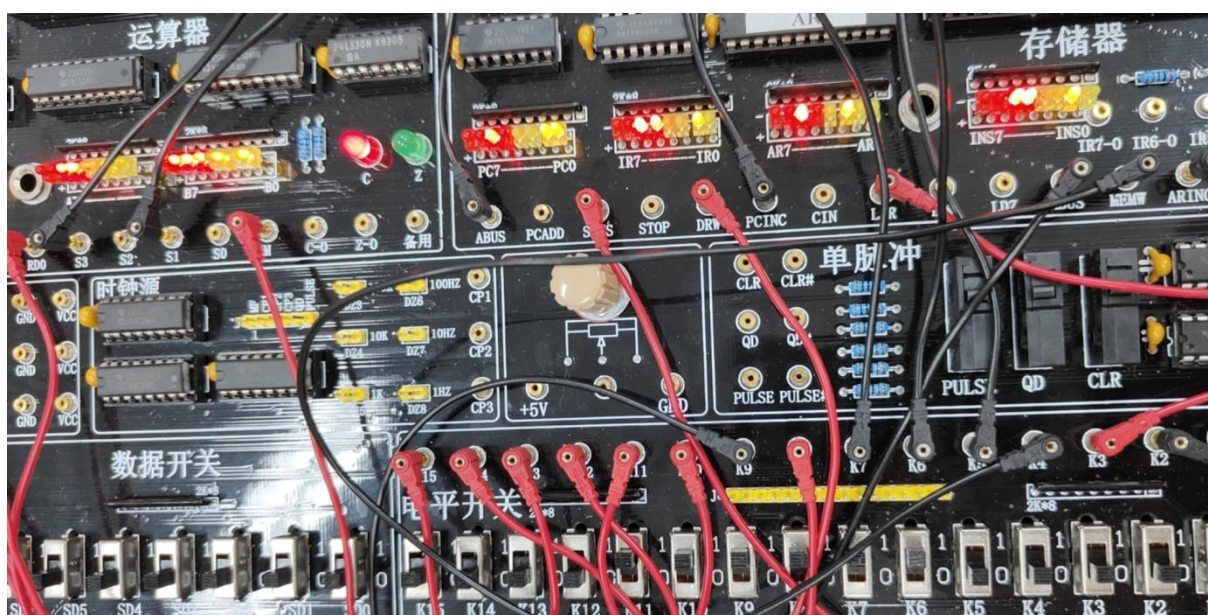
读出 R2=28H。

K15	K14	K13	K12	K11	K10	K9	K8	K7	K6	K5	K4	K3	K2	K1	K0
				1		1	1	1	1						



读出 R3=75H。

K15	K14	K13	K12	K11	K10	K9	K8	K7	K6	K5	K4	K3	K2	K1	K0
				1	1	1	1	1	1						



6. 实验结果

整理实验数据如下表：

$\mu A5 \sim \mu A0$	A7~A0	B7~B0	D7~D0	AR	PC	INS7	R0	R1	R2	R3
0FH	00H	00H	00H	00H	00H	E1H	00H	00H	00H	00H
32H	75H	00H	75H	00H	00H	E1H	75H	00H	00H	00H
33H	28H	75H	28H	00H	00H	E1H	75H	28H	00H	00H
34H	89H	28H	89H	00H	00H	E1H	75H	28H	89H	00H
35H	32H	89H	32H	00H	00H	E1H	75H	28H	89H	32H
36H	75H	32H	20H	20H	20H	9BH	75H	28H	89H	32H
37H	75H	75H	75H	21H	20H	75H	75H	28H	89H	32H
38H	28H	75H	28H	22H	21H	28H	75H	28H	89H	32H
39H	89H	75H	89H	23H	22H	89H	75H	28H	89H	32H
3AH	32H	75H	32H	24H	23H	32H	75H	28H	89H	32H
3BH	75H	75H	20H	20H	23H	32H	75H	28H	89H	32H
3CH	75H	75H	28H	21H	23H	32H	75H	28H	89H	75H
3DH	28H	75H	89H	22H	23H	32H	75H	28H	28H	75H
3EH	89H	28H	32H	23H	23H	32H	75H	89H	28H	75H
00H	32H	89H	FEH	24H	23H	32H	32H	89H	28H	75H

四、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。

通过 I-cache 来获取地址，D-cache 进行获取数据，进行存储。

五、实验思考与心得

完成这个实验后，我对微处理器的寄存器和存储器之间的数据传输有了更深入的理解。实验的第一步是将特定的数值写入寄存器 R0、R1、R2 和 R3。这个步骤涉及到对十六进制数的理解和寄存器操作的熟悉程度。通过将 75H、28H、89H 和 32H 分别写入对应的寄存器，我加深了对寄存器地址和数据操作的认识。第二步是将寄存器中的数值写入到存储器的指定单元。这个过程让我理解了 CPU 如何与存储器交互，以及如何通过地址来定位存储器中的数据。将 R0、R1、R2 和 R3 的内容分别写入 20H、21H、22H 和 23H 单元，是对存储器地址和数据传输控制的实践。第三步是从存储器的特定单元读取数据到寄存器。这个步骤中，我注意到了地址的准确性对于数据传输的重要性。特别是从 21H 单元读取数据到 R1 和 R2，这让我意识到在实际操作中，地址的精确性和操作的顺序对于实验结果的影响。最后一步是显示寄存器 R0、R1、R2 和 R3 的值，以验证数据传输的正确性。这一步是检验前面步骤是否正确执行的关键。通过检查寄存器的值，我可以确认数据是否正确地在寄存器和存储器之间传输。

通过这个实验，我学习到了微处理器中数据传输的基本原理和操作流程。我认识到了精确的地址管理和数据操作对于整个系统正常工作的重要性。