



实验五 微程序控制器(1)

了解微指令是如何工作的

微程序控制器

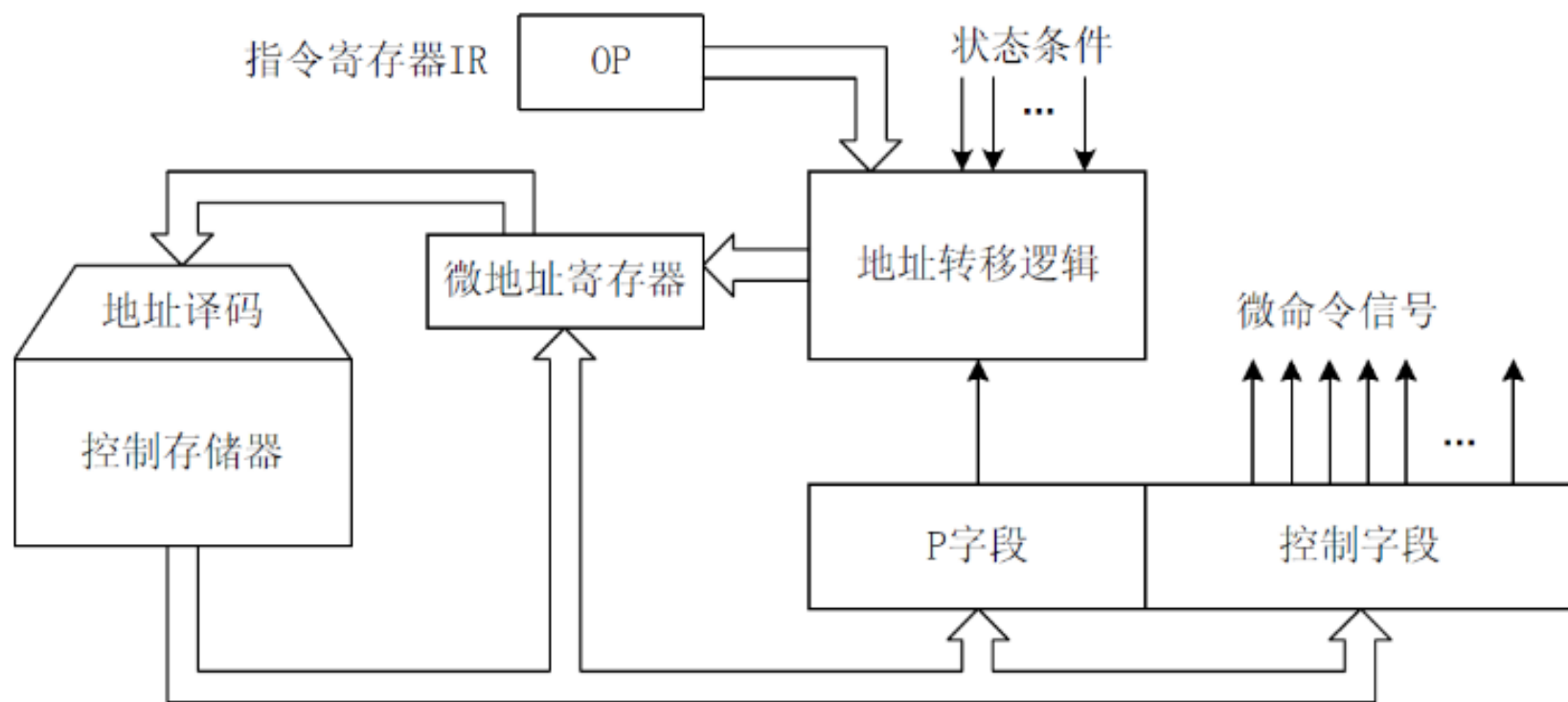


图 3-2-1 微程序控制器组成原理框图

本次实验的主要内容

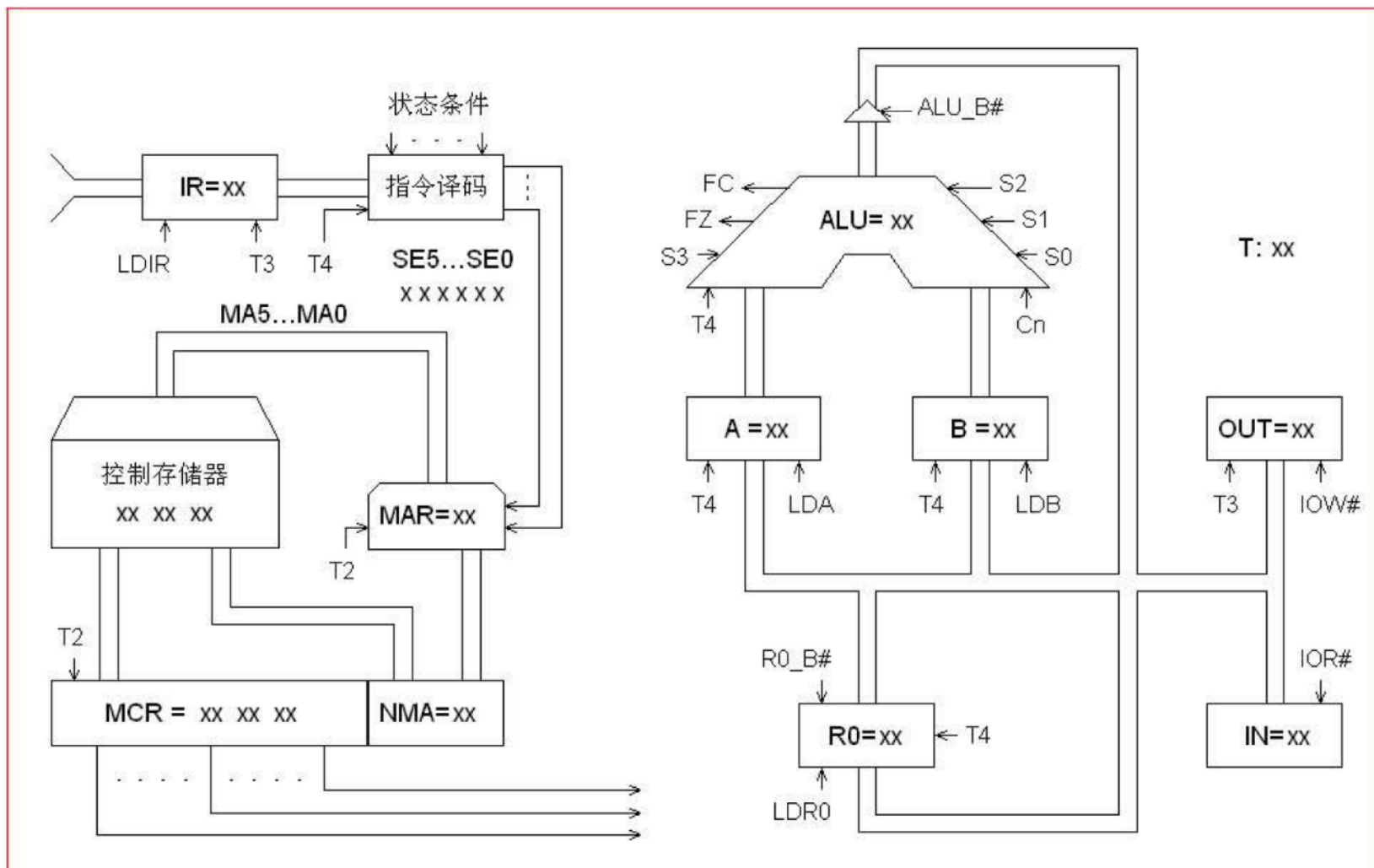
分解执行4条机器指令

助记符	机器指令码	说明
IN	0010 0000	$IN \rightarrow R0$
ADD	0000 0000	$R0 + R0 \rightarrow R0$
OUT	0011 0000	$R0 \rightarrow OUT$
HLT	0101 0000	停机

机器指令与微指令的关系

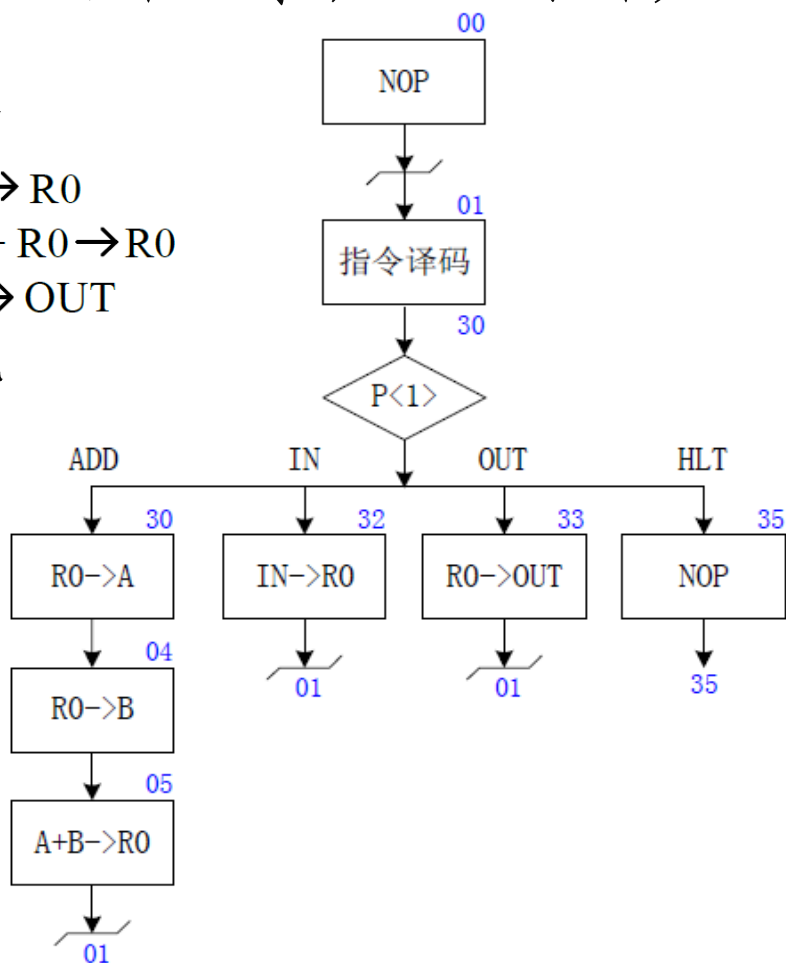
- 1、一条机器指令对应一个微程序；一条机器指令所完成的操作划分为若干条微指令完成；
- 2、指令存放在存储器中；微指令存放在控制存储器中
- 3、一个CPU周期对应一条微指令，一个指令周期至少包含两个CPU周期

数据通路



4条机器指令所对应的微程序

助记符	机器指令码	说明
IN	0010 0000	$IN \rightarrow R0$
ADD	0000 0000	$R0 + R0 \rightarrow R0$
OUT	0011 0000	$R0 \rightarrow OUT$
HLT	0101 0000	停机



微地址以16进制表示

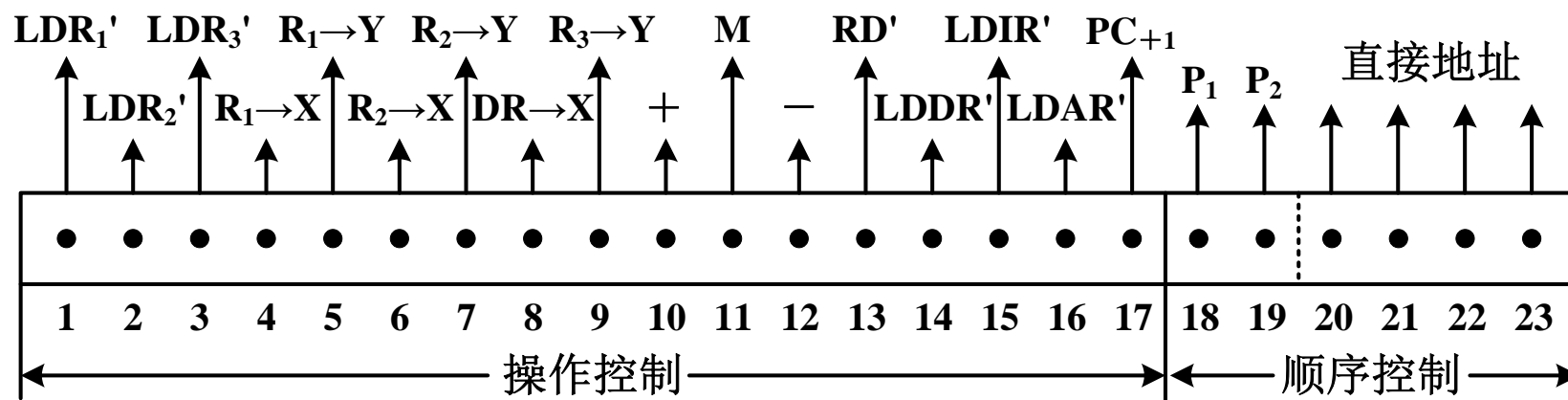
图 3-2-9 微程序流程图

设计8条微指令

存放在控制存储器中

每条微指令存放的位置（地址）

每条微指令的内容



需要微命令控制的微操作有.....

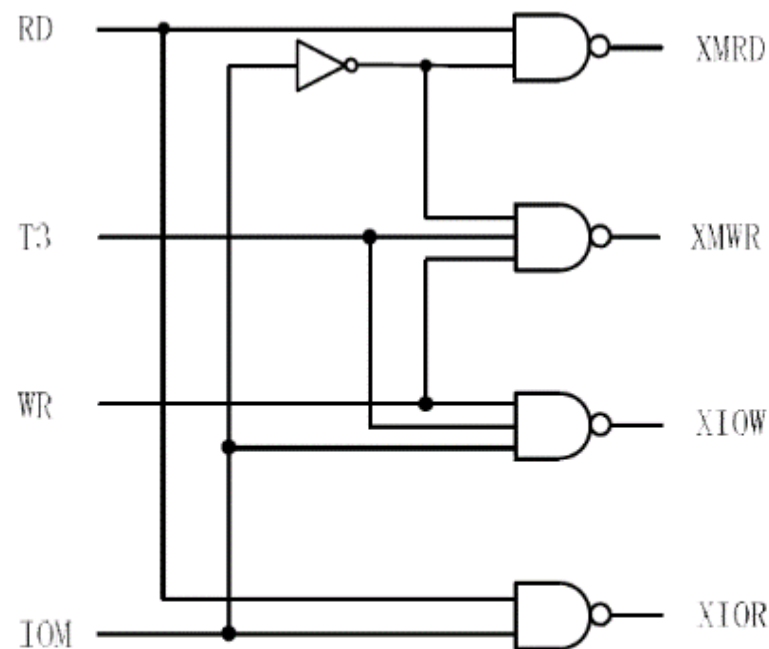
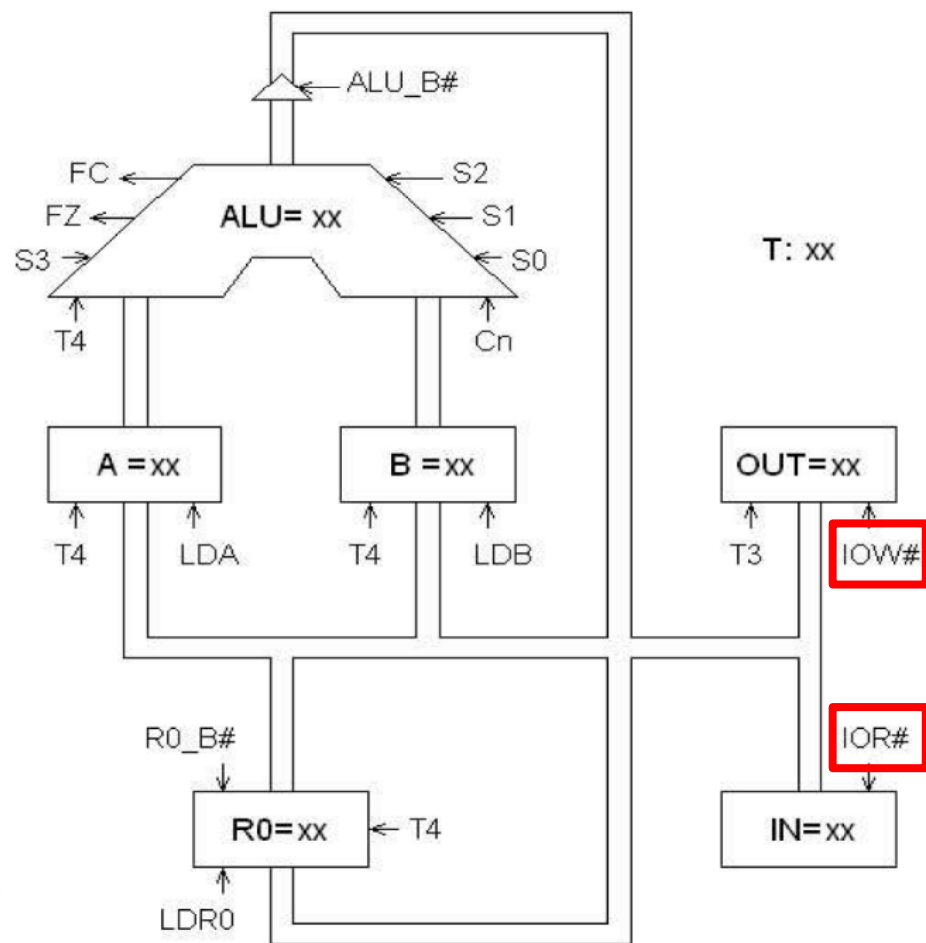
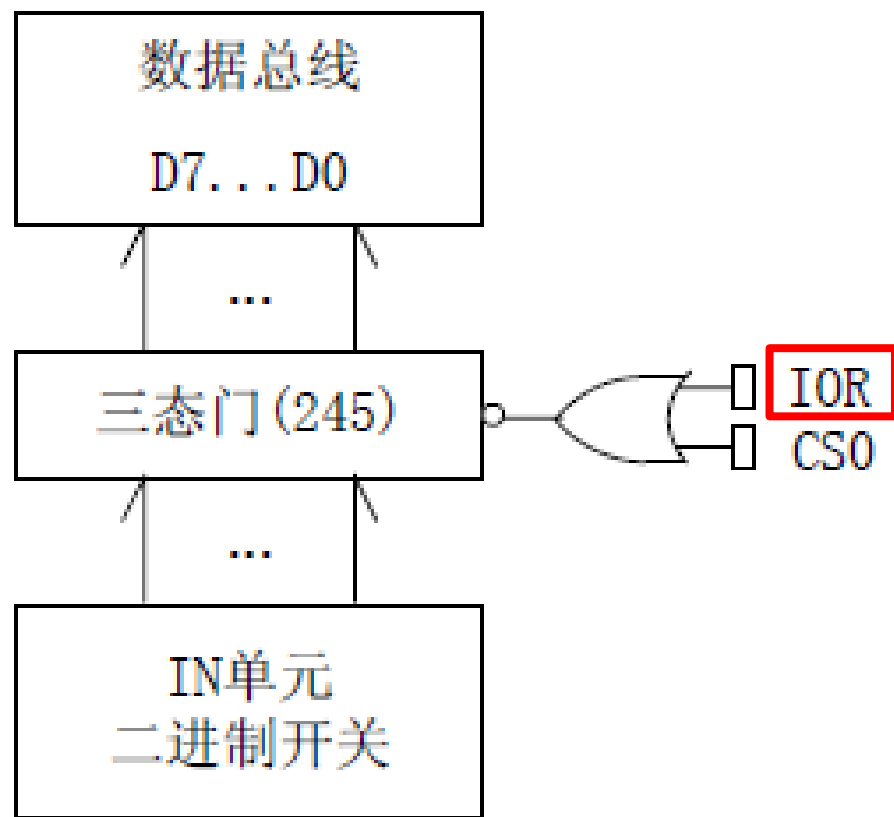


图 3-2-4 读写控制逻辑

IOM=0表示对**MEM**操作

IOM=1表示对外设操作(输入、输出)

需要微命令控制的微操作有.....



IN单元原理图

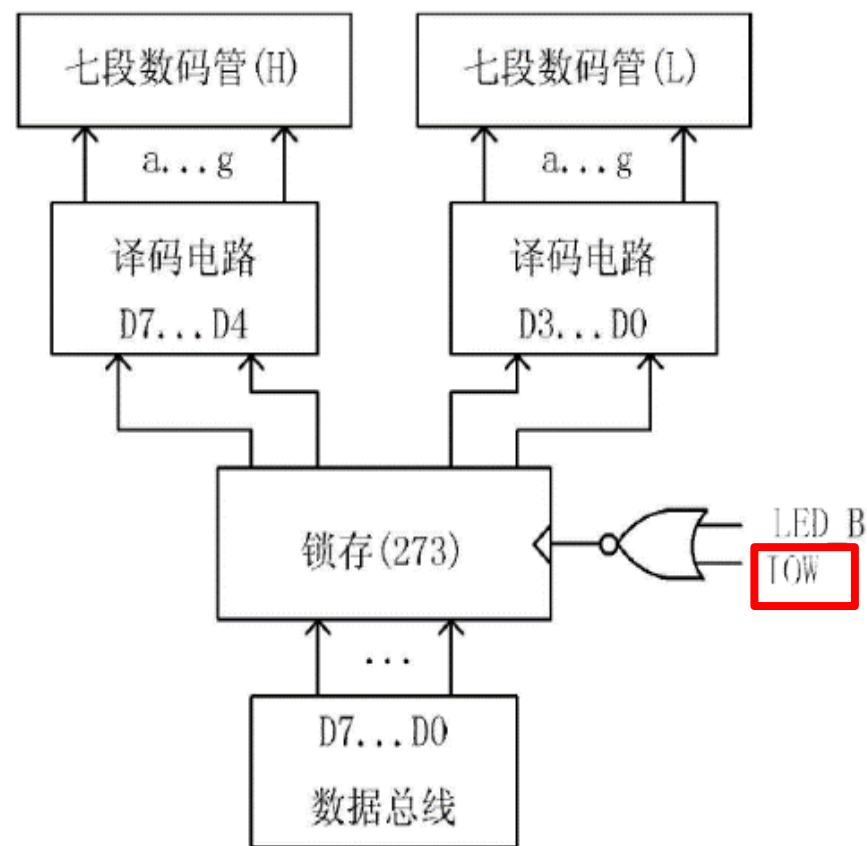


图 3-2-6 OUT 单元原理图

需要微命令控制的微操作有.....

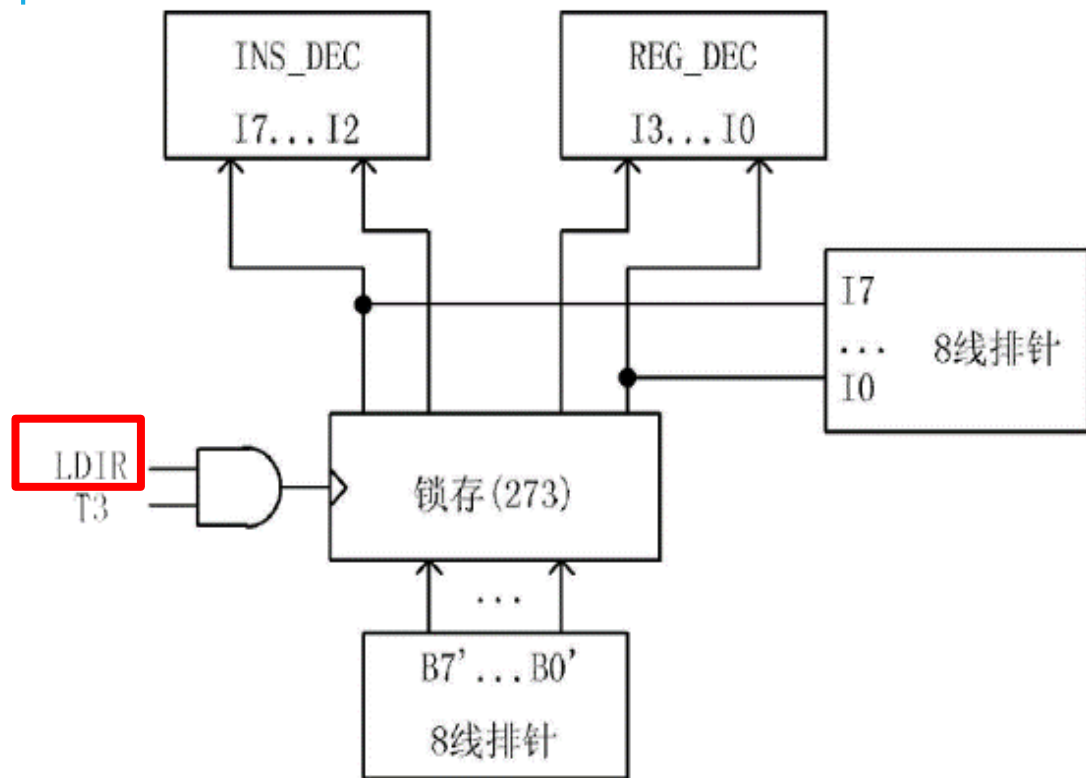


图 3-2-5 IR 单元原理图

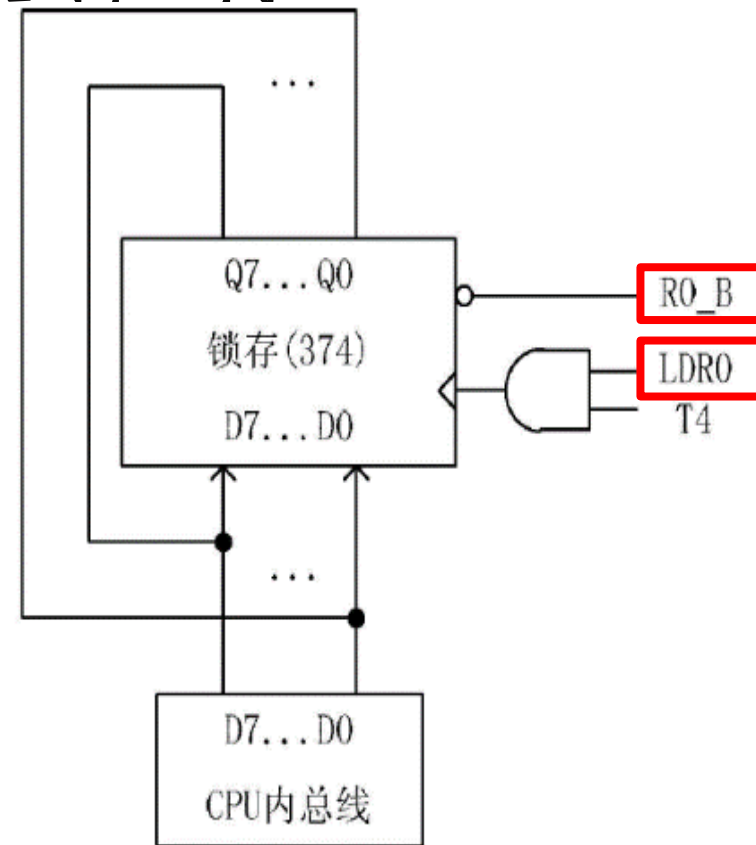


图 3-2-7 R0 原理图

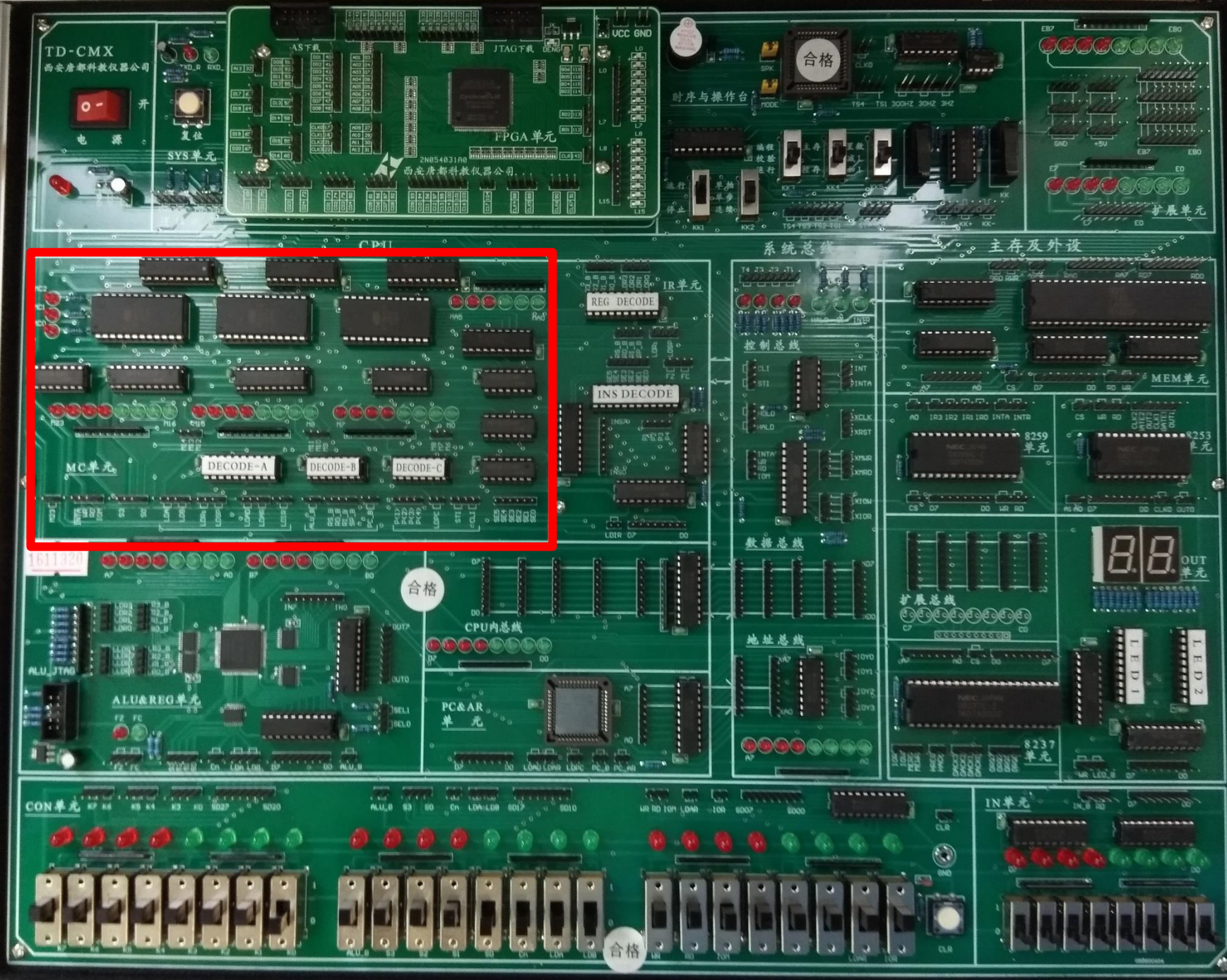


表 3-2-1 微指令格式

23	22	21	20	19	18-15	14-12	11-9	8-6	5-0
M23	M22	WR	RD	IOM	S3-S0	A字段	B字段	C字段	MA5-MA0

A字段

14	13	12	选择
0	0	0	NOP
0	0	1	LDA
0	1	0	LDB
0	1	1	LDR0
1	0	0	保留
1	0	1	保留
1	1	0	保留
1	1	1	LDIR

B字段

11	10	9	选择
0	0	0	NOP
0	0	1	ALU_B
0	1	0	RO_B
0	1	1	保留
1	0	0	保留
1	0	1	保留
1	1	0	保留
1	1	1	保留

C字段

8	7	6	选择
0	0	0	NOP
0	0	1	P<1>
0	1	0	保留
0	1	1	保留
1	0	0	保留
1	0	1	保留
1	1	0	保留
1	1	1	保留

译码器的输出都为低电平

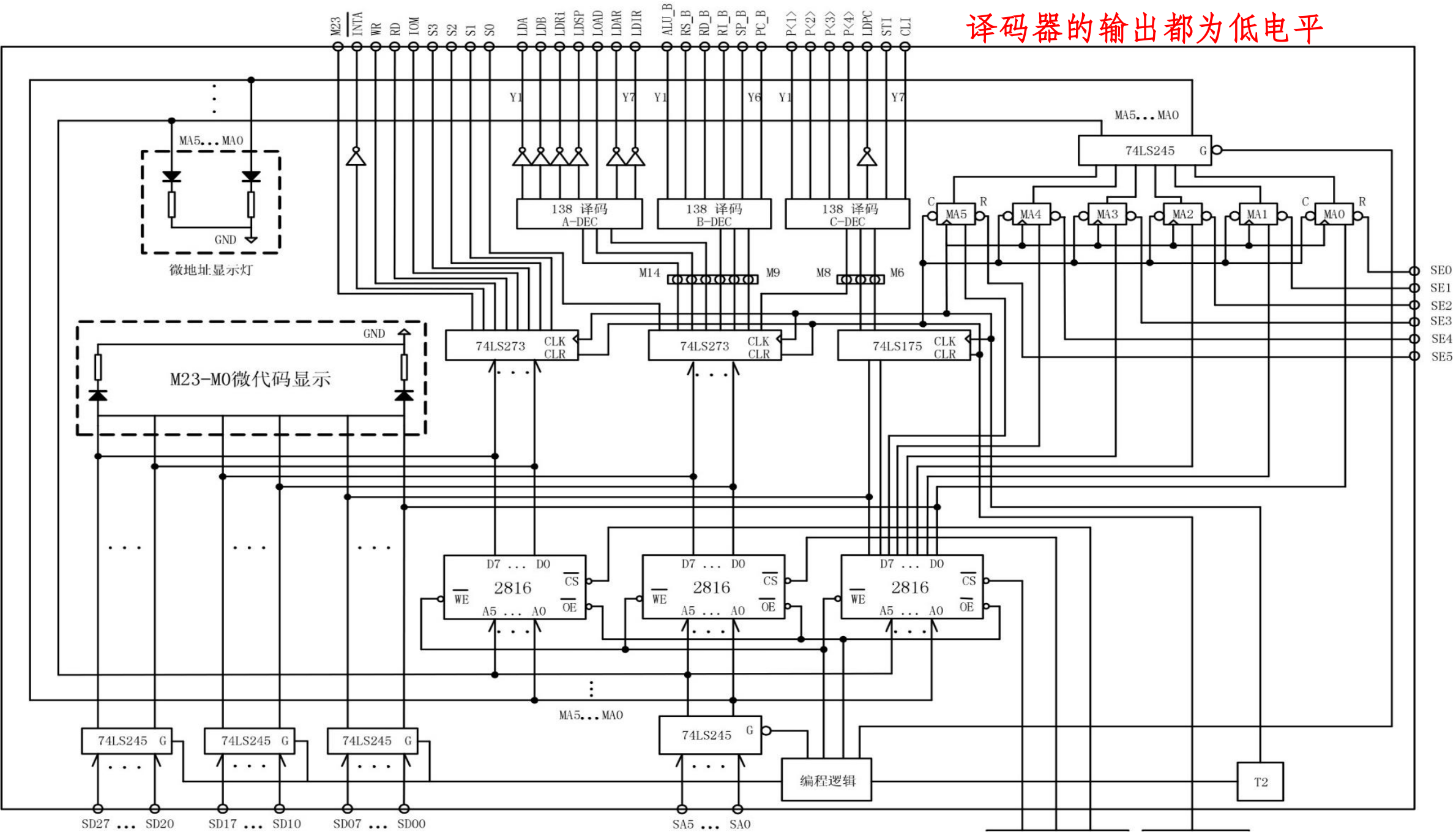
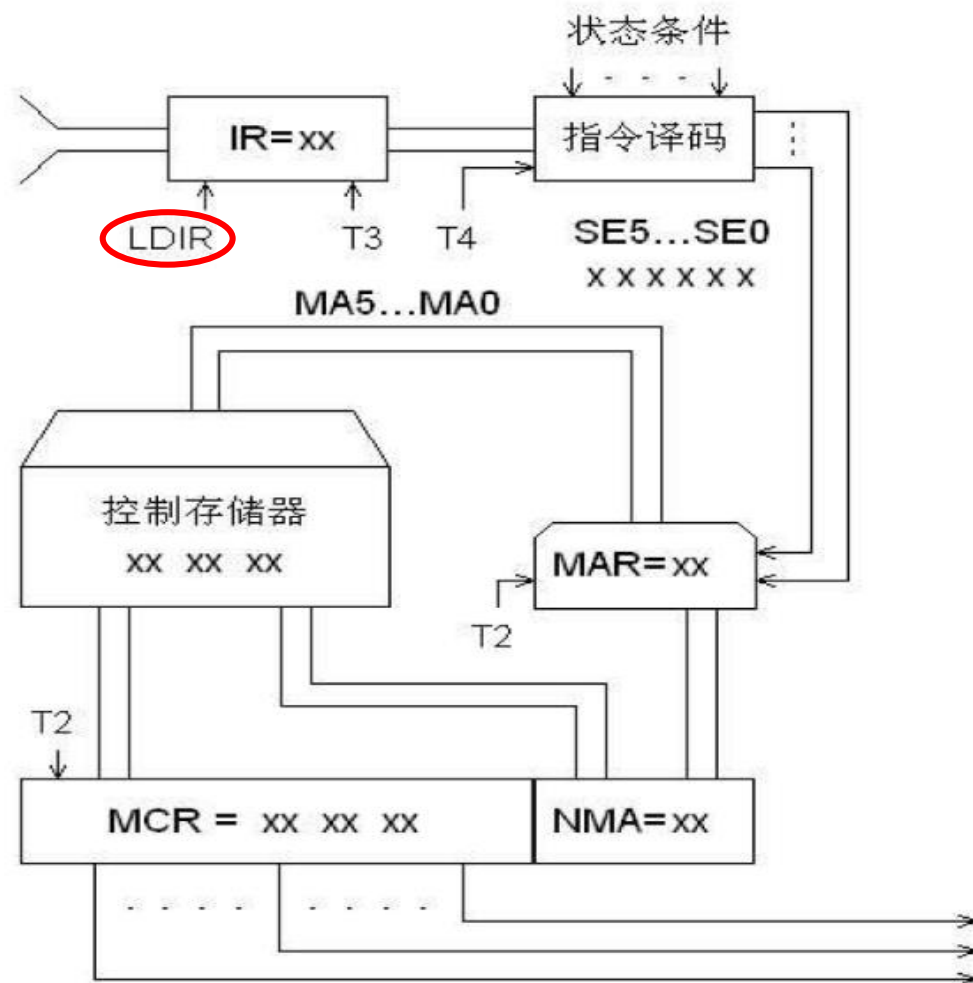


图 3-2-2 微程序控制器原理图

P<1>测试的地址转移逻辑



当 $I_6=I_7=1$ 时， I_7 、 I_6 、 I_3 、 I_2 修改微地址最后四位
否则， I_7 、 I_6 、 I_5 、 I_4 修改微地址最后四位

指令译码

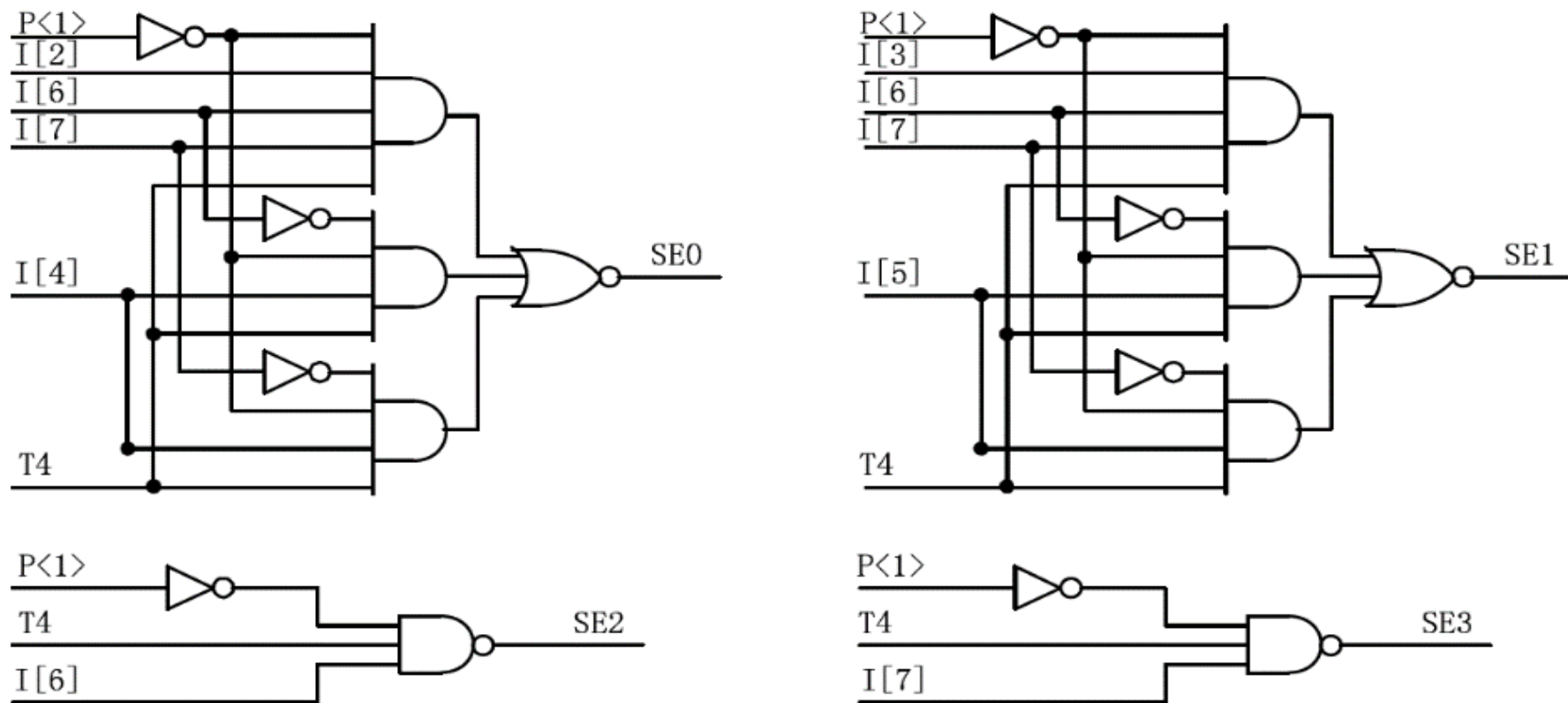


图 3-2-3 指令译码原理图

[illegible]

请参照此页的格式和地址设定完成上页表格

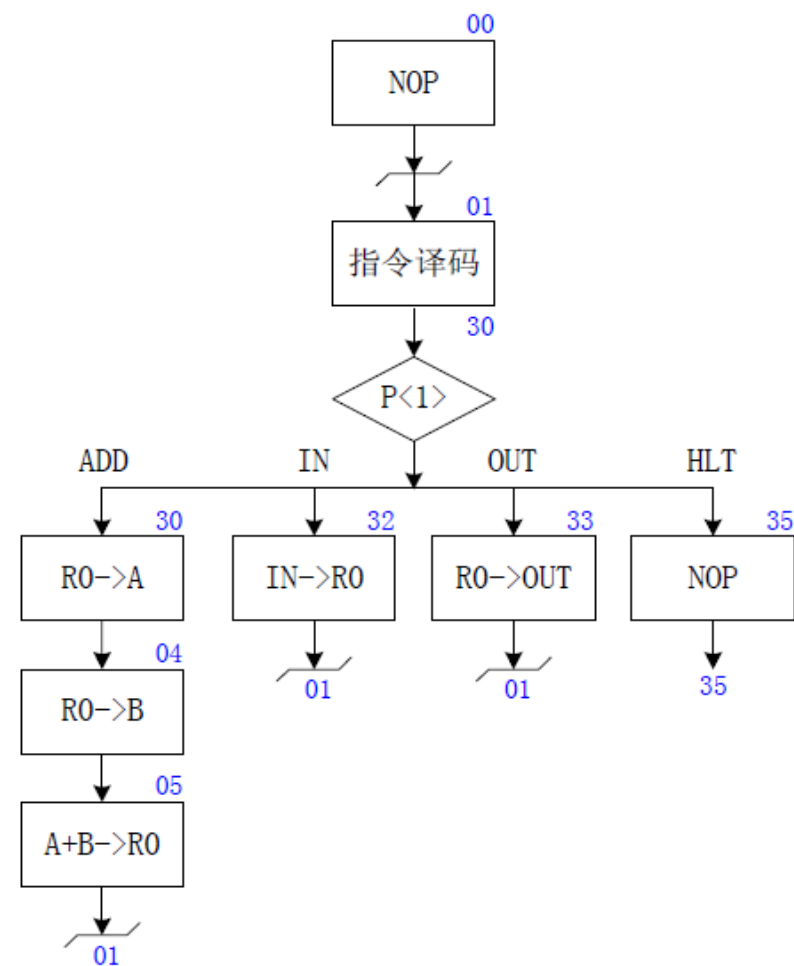


表 3-2-1 微指令格式

23	22	21	20	19	18-15	14-12	11-9	8-6	5-0
M23	M22	WR	RD	IOM	S3-S0	A字段	B字段	C字段	MA5-MA0

A字段

14	13	12	选择
0	0	0	NOP
0	0	1	LDA
0	1	0	LDB
0	1	1	LDRO
1	0	0	保留
1	0	1	保留
1	1	0	保留
1	1	1	LDIR

B字段

11	10	9	选择
0	0	0	NOP
0	0	1	ALU_B
0	1	0	RO_B
0	1	1	保留
1	0	0	保留
1	0	1	保留
1	1	0	保留
1	1	1	保留

C字段

8	7	6	选择
0	0	0	NOP
0	0	1	P<1>
0	1	0	保留
0	1	1	保留
1	0	0	保留
1	0	1	保留
1	1	0	保留
1	1	1	保留

图 3-2-9 微程序流程图