



《计算机组成原理》

实 验 指 导 书

计算机与通信学院

目录

实验一	运算器实验.....	1
实验二	存储器实验.....	4
实验三	微控制器实验.....	6
实验四	基本模型机的设计与实现.....	10
实验五	复杂模型机的设计与实现.....	15

实验一 运算器实验

一、实验目的

- 1、掌握简单运算器的数据传送通路组成原理。
- 2、验证算术逻辑运算功能发生器 74LS181 的组合功能。

二、实验设备

现代计算机组成原理实验系统、计算机。

三、实验注意事项

不要随手把金属物品如钥匙等放在实验面板上，以免烧坏芯片。不要用手触摸芯片。
先接线，后打开电源，养成不带电接线的习惯以免烧坏芯片。

四、实验内容

1、实验原理

算术逻辑单元 ALU 的数据通路如图1-1 所示。其中运算器ALU181 根据74LS181 的功能用VHDL 硬件描述语言编辑而成，构成8 位字长的ALU。参加运算的两个8 位数据分别为A[7..0]和B[7..0]，运算模式由S[3..0]的16 种组合决定，而S[3..0]的值由4 位2 进制计数器LPM_COUNTER 产生，计数时钟是Sc1k（图1-1）；此外，设M=0，选择算术运算，M=1 为逻辑运算，CN 为低位的进位位；F[7..0]为输出结果，CO 为运算后的输出进位位。两个8 位数据由总线IN[7..0]分别通过两个电平锁存器74373 锁入。

2、实验步骤

（1）使用下载器连接电脑与实验箱，检查无误后给实验箱上电。

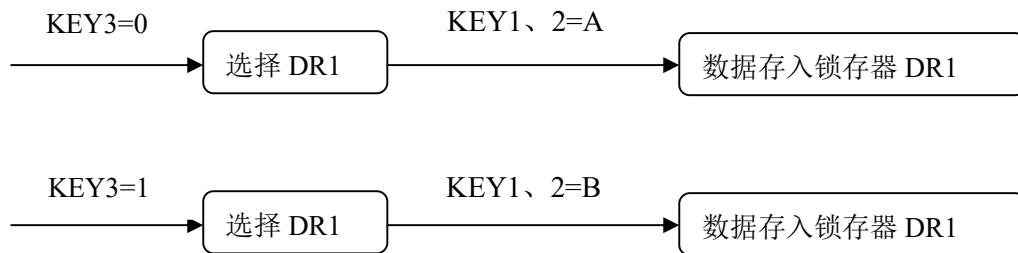
（2）在QuartusII 环境下，打开运算器工程文件alu.bdf，然后将alu.sof文件下载到实验台。了解此运算器设计示例，分析数据总线、控制总线。也可直接调用内置元件74181以原理图输入方式完成电路设计。具体方法见《实验参考资料—软硬件操作流程》。

（3）选择实验系统的电路模式：NO.0，即数码管“模式指示”显示0。

（4）用键1、2向DR1和DR2寄存器置数，根据数据选择开关（键3 控制）的高/低电平选择总线通道上的8位数据进入对应的74373 中。方法如下：

首先将键3置低电平，用键2、键1分别向A[7..0] 置数（实验批次），这时在数码管2/1 上显示输

入的数据A；然后键3 置高电平，再用键2、键1 分别向B[7..0]置数（序号），数码管4/3显示输入的数据B。至此图1-1 中的两个74373 锁存器中分别被锁入了加数A和被加数B。



（5）控制键6置S[3..0]，键7设置CN值，键8置M值，数码管6/5显示运算结果。按表1-1，验证ALU的算术、逻辑运算功能，并记录实验数据。

表 1-1 ALU 实验数据表

S3 S2 S1 S0	A[7..0]	B[7..0]	算术运算 M=0		逻辑运算 (M=1)
			Cn=0 (无进位)	cn=1 (有进位)	
0000			F= ()	F= ()	F= ()
0001			F= ()	F= ()	F= ()
0010			F= ()	F= ()	F= ()
0011			F= ()	F= ()	F= ()
0100			F= ()	F= ()	F= ()
1001			F= ()	F= ()	F= ()
1010			F= ()	F= ()	F= ()
1100			F= ()	F= ()	F= ()
1101			F= ()	F= ()	F= ()
1110			F= ()	F= ()	F= ()
1111			F= ()	F= ()	F= ()

五、实验要求

1、做好实验预习，掌握运算器的数据传送通路和ALU 的功能特性，并熟悉本实验中所用的控制台开

实验二 存储器实验

一、实验目的

掌握静态随机存取存储器 RAM 工作特性及数据的读写方法。

二、实验设备

现代计算机组成原理实验系统、计算机。

三、实验注意事项

不要随手把金属物品如钥匙等放在实验面板上，以免烧坏芯片。不要用手触摸芯片。

先接线，后打开电源，养成不带电接线的习惯以免烧坏芯片。

四、实验内容

1、实验原理

在 FPGA 中利用嵌入式阵列块EAB 可以构成存储器，lpm_ram_dq 的结构如图2-1。数据从ram_dp0的左边D[7..0]输入，从右边Q[7..0]输出，R/W——为读/写控制信号端。数据的写入：当输入数据和地址准备好以后，在inclock 是地址锁存时钟，当信号上升沿到来时，地址被锁存，数据写入存储单元。数据的读出：从 A[7..0]输入存储单元地址，在CLK 信号上升沿到来时，该单元数据从Q[7..0]输出。

R/W——读/写控制端，低电平时进行读操作，高电平时进行写操作。

CLK——读/写时钟脉冲； DATA[7..0]——RAM_dq0 的8 位数据输入端。

A[7..0]——RAM 的读出和写入地址； Q[7..0]——RAM_dq0 的8 位数据输出端。

2、实验步骤

(1) 在 QuartusII 环境下，打开存储器工程文件 RAM_DP1.bdf，然后将 RAM_DP1.sof 文件下载到实验台。

(2) 选择实验电路模式 1，通过键 1、键 2 输入 RAM 的 8 位数据，键 3、键 4 输入存储器的 8 位地址。键 8 控制读/写允许，低电平时读允许，高电平时写允许；键 7 (CLK0) 产生读/写时钟脉冲，即生成写地址锁存脉冲，对 lpm_ram_dq 进行写/读操作。

(3) 利用 QuartusII 的 In-System Memory Content Editor 工具对 RAM 中的数据进行读、写、修改操作，可以将系统读写工具窗口的数据与实验箱上数码管上显示的数据对照起来看。

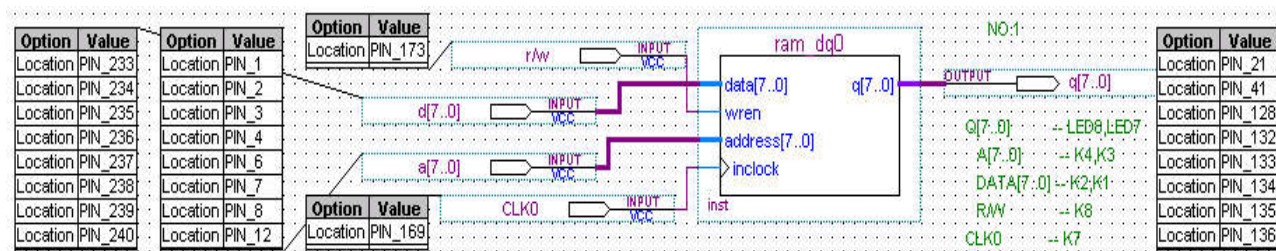


图 2-1 lpm_ram_dp 实验电路图

表 2—1 典型 RAM 器件 6264 功能表

工作 方式	I/O		输入		
	DI	DO	/OE	/WE	/CS
非选择	X	HIGH-Z	X	X	H
读出	HIGH-Z	DO	L	H	L
写入	DI	HIGH-Z	H	L	L
写入	DI	HIGH-Z	L	L	L
选择	X	HIGH-Z	H	H	L

五、实验要求

1. 做好实验预习，掌握随机存储器 RAM 的工作特性和读写方法。
2. 利用 QuartusII 的 In-System Memory Content Editor 工具对 RAM 中的数据进行读、写、修改操作，并在实验报告上详细记录实验过程。
3. 利用实验平台按键，实现在“批次”地址写入数据“序号”，并读出，在实验报告上详述实验过程。

实验三 微控制器实验

一、实验目的

- 1、掌握时序信号发生电路组成原理。
- 2、掌握微程序控制器的设计思想和组成原理。
- 3、掌握微程序的编制、写入，观察微程序的运行。

二、实验设备

现代计算机组成原理实验系统、计算机。

三、实验注意事项

不要随手把金属物品如钥匙等放在实验面板上，以免烧坏芯片。不要用手触摸芯片。

先接线，后打开电源，养成不带电接线的习惯以免烧坏芯片。

四、实验内容

1、实验原理

连续节拍发生电路设计（图 3-1），由 4 个 D 触发器组成，可产生 4 个等间隔的时序信号 $T_1 \sim T_4$ 。其中 CLK1 为时钟信号。当 RST1 为低电平时，T1 输出为“1”，而 T2、T3、T4 输出为“0”；当 RST1 由低电平变为高电平后，T1~T4 将在 CLK1 的输入脉冲作用下，周期性地轮流输出正脉冲，机器进入连续运行状态。

微程序控制器的组成如图3-2。其中控制存储器由FPGA 中的LPM_ROM 构成，输出24 位控制信号。在 24 位控制信号中，微命令信号18 位，微地址信号6 位。在不判别测试的情况下，在T2 时刻将打入微地址寄存器uA 的内容，即为下一条微指令地址。当T4 时刻进行测试判别时，转移逻辑满足条件后输出的负脉冲通，过强制端将某一触发器置为“1”状态，完成地址修改。

微程序控制器中的微控制代码可以通过对FPGA 中LPM_ROM 的配置进行输入，通过编辑LPM_ROM.mif文件修改微控制代码。详细情况可参考实验二中FPGA 中LPM_RAM 的配置方法。

2、微指令

实验模型机 CPU 的微指令共 24 位，由操作控制字段和下地址字段组成。编码时将微操作控制字段划分为若干个小字段，每个小字段独立译码，每个码点表示一个微命令。其微指令结构如表 3-1 所示。微指令的功能及表 3-2 中 A、B、C 各字段功能说明如下：

- uA5—uA0：微程序控制器的微地址输出信号，是下一条要执行的微指令的微地址。
- S3、S2、S1、S0：由微程序控制器输出的 ALU 操作选择信号，以控制执行 16 种算术操作或 16 种逻辑操作中的某一种操作。
- M：微程序控制输出的 ALU 操作方式选择信号。M=0 执行算术操作；M=1 执行逻辑操作。
- Cn：微程序控制器输出的进位标志信号，Cn=0 表示 ALU 运算时最低位有进位，Cn=1 则表示无进位。
- WE：微程序控制器输出的 RAM 控制信号。当 /CE=0 时，如 WE=0，为存储器读；如 WE=1，为存储器写。
- A9、A8：译码后产生 CS0、CS1、CS2 信号，分别作为 SW_B、RAM、LED 的选通控制信号。
- A 字段（15、14、13）：译码后产生与总线相连接的各单元的输入选通信号（表 3-2）。
- B 字段（12、11、10）：译码后产生与总线相连接的各单元的输出选通信号。
- C 字段（9、8、7）：译码后产生分支判断测试信号 P(1)~P(4) 和 LDPC 信号。

表 3-1 24 位微代码定义

24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
S3	S2	S1	S0	M	Cn	WE	A9	A8	A			B			C			<u>uA5</u>	<u>uA4</u>	<u>uA3</u>	<u>uA2</u>	<u>uA1</u>	<u>uA0</u>
操作控制信号									译码器			译码器			译码器			下地址字段					

表 3-2 A、B、C 各字段功能说明

A 字段				B 字段				C 字段			
15	14	13	选择	12	11	10	选择	9	8	7	选择
0	0	0		0	0	0		0	0	0	
0	0	1	LDRi	0	0	1	RS-B	0	0	1	P (1)
0	1	0	LDDR1	0	1	0	RD_B	0	1	0	P (2)
0	1	1	LDDR2	0	1	1	RJ_B	0	1	1	P (3)
1	0	0	LDIR	1	0	0	SFT_B	1	0	0	P (4)
1	0	1	LOAD	1	0	1	ALU-B	1	0	1	LDAR
1	1	0	LDAR	1	1	0	PC-B	1	1	0	LDPC

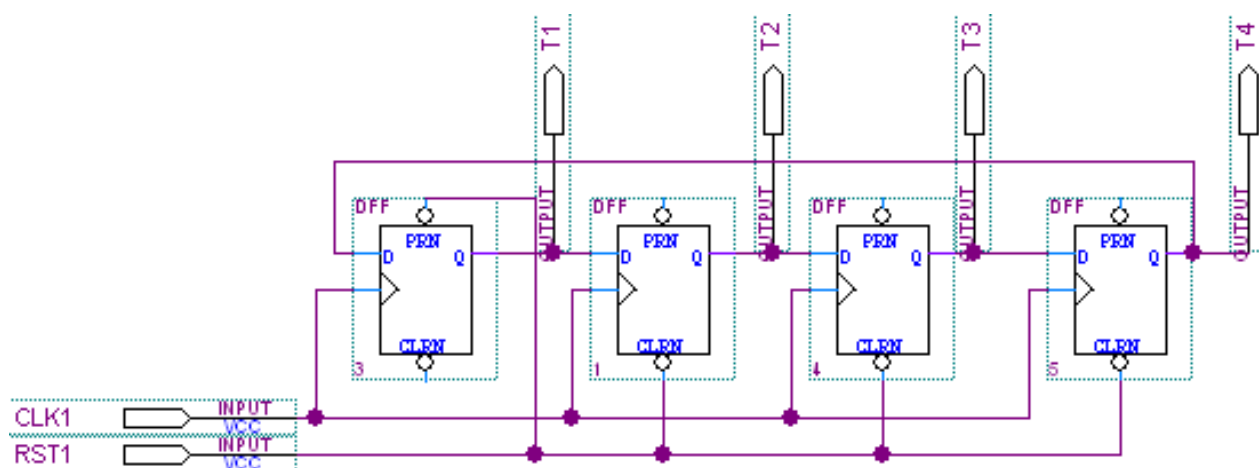


图3-1 节拍脉冲发生器的工作原理

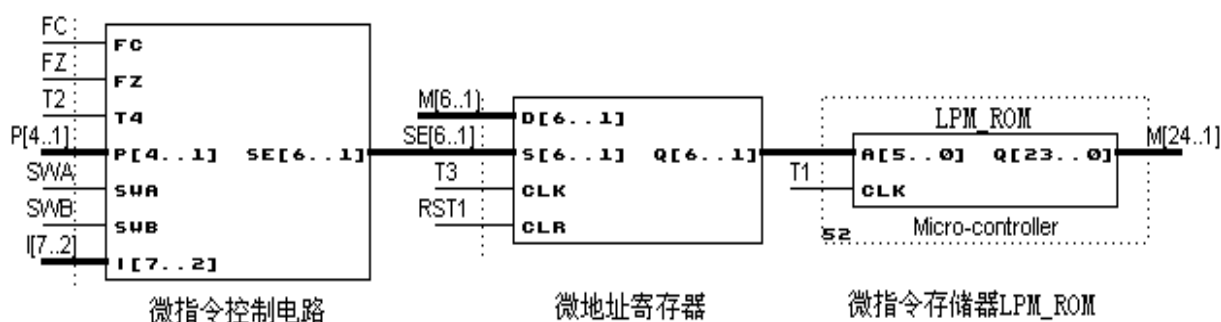


图 3-2 微程序控制电路

2、实验步骤

1) 节拍脉冲发生器时序电路实验。

下载T4.SOF 文件，或输入节拍电路，先仿真，得到节拍脉冲发生器工作波形。再选择实验模式1，Clock0接4Hz，键8 控制RST1，高电平时可以看到，发光管1、2、3、4 分别显示T1、T2、T3、T4 的输出电平。

2) 微指令控制电路实验。

下载 se5_1.sof 到实验台，或输入微指令控制电路，并锁定引脚。编译、下载到实验系统中，选择实验台工作模式No. 1。键盘/显示定义如下：

- (1) 键1、键2 输入6 位微指令数据I[7..2]，键2 中的高两位还作为标志位FC、FZ；
- (2) 键3 输入分支控制信号P[4..1]；3) 键4 输入控制台的控制信号SWA、SWB；
- (3) 键8 输入节拍信号T4；4) 数码 5、数码6 显示微地址控制信号SE[6..1]。

根据微程序控制器的内部结构，记录当FC、FZ 变化时，微指令I[7..2]的变化，对输出微地址控制信号SE[6..1]的影响；观察、记录当微指令 I[7..2]的值变化时，SE[6..1]的变化情况；观察、记录分支信

号 $P[4..1]$ 有效时，微指令 $I[7..2]$ 的变化对输出微地址控制信号 $SE[6..1]$ 的影响；观察、记录 SWA、SWB 对输出微地址控制信号 $SE[6..1]$ 的影响。

五. 实验要求

- 1、做好实验预习。
- 2、将仿真的节拍脉冲发生器工作波形记录在实验报告上，注意时间轴的选择。
- 3、将微指令控制电路测得的值记录在实验报告上。

实验四 基本模型机的设计与实现

一、实验目的

- 1、在掌握部件单元电路实验的基础上，进一步将其组成系统以构造一台基本模型实验计算机。
- 2、设计五条机器指令，并编写相应的微程序，具体上机调试，掌握整机软硬件组成概念。

二、实验设备

现代计算机组成原理实验系统、计算机。

三、实验注意事项

不要随手把金属物品如钥匙等放在实验面板上，以免烧坏芯片。不要用手触摸芯片。

先接线，后打开电源，养成不带电接线的习惯以免烧坏芯片。

四、实验内容

1、实验原理

1) 8 位模型计算机的数据通路框图如图 4-1 所示。

2) 指令格式

一条指令就是机器语言的一个语句，它是一组有意义的二进制代码。指令的基本格式如表 4-1 所示。其中操作码 OP-CODE 指明了指令的操作性质及功能，地址码则给出了操作数或操作数的地址，采用寄存器直接寻址方式。

表 4-1 指令的基本格式

位	7	6	5	4	32	10
功能	OP-CODE				rs	rd

模型机指令系统中包括有五条基本指令，分为算术运算指令、存 / 取指令和控制转移指令等三种类型。其功能如表 4-3 所示。五条机器指令分别是：IN（输入）、ADD（二进制加法）、STA（存数）、OUT（输出）、JMP（无条件转移），指令格式如表 3-1 所示（高 4 位二进制数为操作码）。其中，OP-CODE 为操作码，rs 为源寄存器，rd 为目的寄存器，并规定了寄存器操作数的格式（表

4-2)。其中 IN 为单字长（8 位二进制），其余为双字长指令，XX H 为 addr 对应的十六进制地址码。

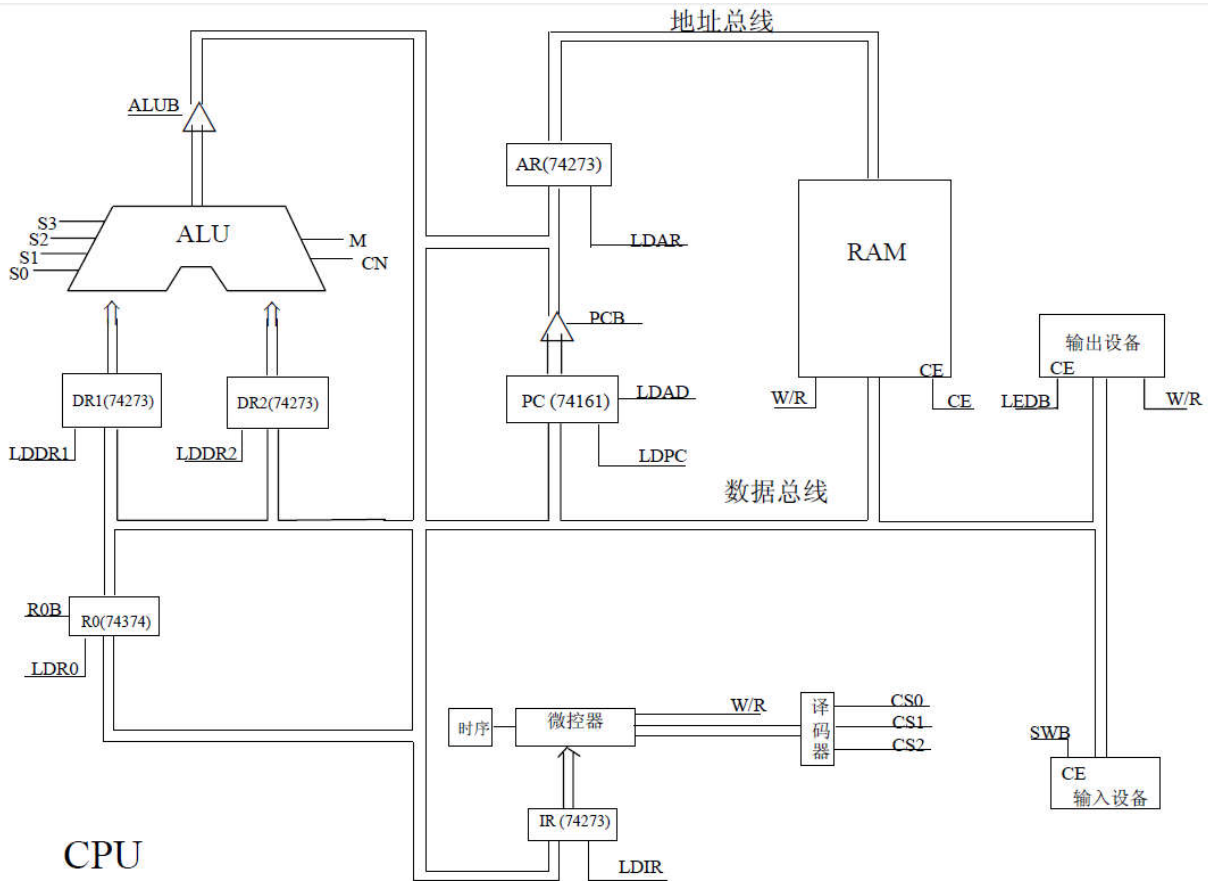


图 4-1 基本模型机数据通路框图

表 4-2 寄存器操作数

rs 或 rd	选定的寄存器
00	R0
01	R1
10	R2

表 4-3 模型机指令系统，及其指令编码形式

助记符	机器指令码	Addr 地址码	功能说明
IN	0 0H		“INPUT”中的数据→R0
ADD addr	1 0H	XX H	R0+[addr] →R0
STA addr	2 0H	XX H	R0 → [addr]
OUT addr	3 0H	XX H	[addr] → BUS
JMP addr	4 0H	XX H	addr → PC

3) 实验程序

地址	内容	助记符	说明
00	00	IN	“INPUT” → R0, 键盘输入数据
01	10	ADD [0AH]	[R0]+[0AH] → R0
02	0A		
03	20	STA [0BH]	[R0] → [0BH]
04	0B		
05	30	OUT [0BH]	[0BH] “OUTPUT”, 显示输出数据
06	0B		
07	40	<u>JMP [08H]</u>	[08H] → PC
08	00		
09	00		
0A	34	34H	被加数
0B	XX		求和结果

4) 基本模型机实验微程序流程如图 4-2。

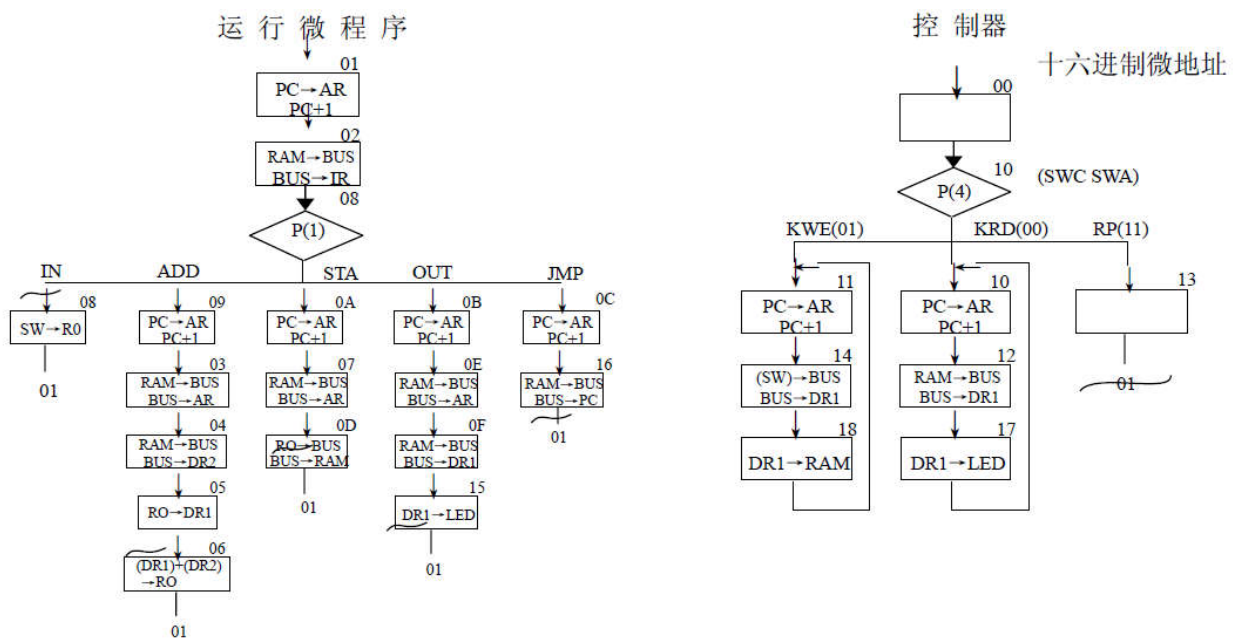


图 4-2 微程序流程图

2、实验步骤

- 1) 在 QuartusII 环境下，打开基本模型机工程文件 CPU5A.bdf，将 CPU5A.sof 文件下载到实验台。了解此 CPU 设计示例，包括引脚锁定与实验系统控制键的关系等。
- 2) 利用 QuartusII 的 In-System Memory Content Editor 工具观察 ROM、RAM 中预置的数据。
- 3) “模式选择”置 0，按一次“系统复位”按钮。
- 4) 控制开关 SWB、SWA（键 4、键 3）设置为“11”。再通过键 2、键 1 输入运算数据，指导老师没有特殊要求时，运算数据用学生的序号。
- 5) 按动键 7（注意每按 2 次产生一个单步脉冲），使 CPU 单步（微指令单步）运行。即每按 2 次键 7 产生一个 STEP 单步脉冲，执行一个微操作。通过液晶显示屏显示数据观察微程序的执行过程。
- 6) 用自己的运算数据完成表4-5的“STEP”、“后续uA 微地址”、“MC 微指令”、“PC”、“IR 指令”、“执行结果”这6列数据。“完成功能”这一列写实验报告时完成。

表 4-4 LCD 液晶显示屏功能说明

名称	作用	名称	作用
IN	输入单元 INPUT	DR1	暂寄存器 DR1
OUT	输出单元 OUTPUT	DR2	暂寄存器 DR2
ALU	算术逻辑单元	PC	程序计数器
BUS	内部数据总线	AR	地址寄存器
R0	寄存器 R0	RAM	程序/数据存储器
R1	寄存器 R1	IR	指令寄存器
R2	寄存器 R2	MC	微程序控制器

现代计算机组成原理实验					
IN	00	OUT	00	ALU	00
R0	00	R1	00	R2	00
DR1	00	DR2	00	BUS	00
PC	00	AR	00	RAM	00
IR	00	uA	00	MC	018110

图 4-3 LCD 液晶显示屏

五、实验要求

1. 实验之前应认真准备，熟悉实验步骤和具体实验内容。
2. 用In-System Memory Content Editor观察ROM、RAM中的数据。
3. 通过液晶屏，观察各相关寄存器、ALU、DR1、PC、IR、AR、BUS、MC等内容的变化情况，根据微程序控制流程，单步跟踪微程序的执行情况。
4. 通过INPUT（键2、键1）输入运算数据（序号），跟踪程序的执行情况，并详细记录每条微指令执行后，相关单元输出数据的变化情况，依次执行机器指令，记录实验数据（表4-5）。
5. 完成实验报告时，按自己的理解填写表4-5“完成功能”这一列。

表 4-5 微指令执行情况

STEP	后续 uA	PC	IR	执行结果	完成功能
1	00	00	00	控制台操作	控制台（读/写/运行）功能判断
2	23				SWB、WSA=（11）转 RP，分支转移
3	01				转程序执行方式
4	02	01	00	PC→AR=00H, PC+1=01H	程序计数器给地址寄存器传值00，并自加一
5	10			RAM(00H)=00→BUS→IR=00H	RAM将00地址中的值00(指令)传递给IR
6	01	03	00	R0=56H	执行微指令 SW→R0

实验五 复杂模型机的设计与实现

一、实验目的

综合运用所学计算机原理知识，设计并实现较为完整的计算机。

二、实验设备

现代计算机组成原理实验系统、计算机。

三、实验注意事项

不要随手把金属物品如钥匙等放在实验面板上，以免烧坏芯片。不要用手触摸芯片。

先接线，后打开电源，养成不带电接线的习惯以免烧坏芯片。

四、数据格式及指令系统

1、数据格式

模型机规定采用定点补码表示法表示数据，且字长为 8 位，其格式如下：

7	6 5 4 3 2 1 0
符 号	尾 数

其中第 7 位为符号位，数值表示范围是： $-1 \leq X < 1$ 。

2、指令格式

模型机设计四大类指令共十六条，其中包括算术逻辑指令、I/O 指令、存数指令、取数指令、转移指令和停机指令。

(1) 算术逻辑指令

设计 9 条算术逻辑指令并用单字节表示，寻址方式采用寄存器直接寻址，其格式如下：

7 6 5 4	3 2	1 0
OP-CODE	RS	RD

其中，OP—CODE 为操作码，RS 为源寄存器，RD 为目的寄存器，并规定：

RS 或 RD	选定的寄存器
00	R0
01	R1

9 条算术逻辑指令的名称、功能具体见表 5—1。

(2) 访问指令及转移指令

模型机设计 2 条访问指令：即存数 STA、取数 LDA；2 条转移指令：即无条件转移 JMP、有进位转移指令 BZC。指令格式为：

7 6	5 4	3 2	1 0
0 0	M	OP-CODE	RD
D			

其中，OP-CODE 为操作码，RD 为目的寄存器地址（LDA、STA 指令使用）。D 为位移量（正负均可），M 为寻址模式，其定义如下：

寻址模式 M	有效地址 E	说 明
00	$E=D$	直接寻址
01	$E=(D)$	间接寻址
10	$E=(RI)+D$	RI 变址寻址

本模型机规定变址 RI 指定为寄存器 R2。

(3) I / O 指令

输入 IN 和输出 OUT 指令采用单字节指令，其格式如下：

7 6 5 4	3 2	1 0
OP-CODE	addr	RD

其中，addr=01 时选中“INPUT DEVICE”中的键盘输入设备，addr=10 时，选中“OUTPUT DEVICE”中的 LCD 点阵液晶屏作为输出设备。

(4) 停机指令

指令格式如下：

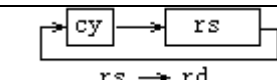
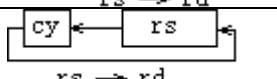
7 6 5 4	3 2	1 0
OP-CODE	0 0	0 0

HALT 指令，用于实现停机操作。

3、指令系统

本模型机共有 16 条基本指令，其中算术逻辑指令 7 条，访问内存指令和程序控制指令 4 条，输入输出指令 2 条，其它指令 1 条。表 5—1 列出了各条指令的格式、汇编符号、指令功能。

表 5-1 指令系统表

助记符号	指令格式	功能								
CLR rd	<table><tr><td>0111</td><td>00</td><td>rd</td></tr></table>	0111	00	rd	$0 \rightarrow rd$					
0111	00	rd								
MOV rs, rd	<table><tr><td>1000</td><td>rs</td><td>rd</td></tr></table>	1000	rs	rd	$rs \rightarrow rd$					
1000	rs	rd								
ADC rs, rd	<table><tr><td>1001</td><td>rs</td><td>rd</td></tr></table>	1001	rs	rd	$rs + rd + cy \rightarrow rd$					
1001	rs	rd								
SBC rs, rd	<table><tr><td>1010</td><td>rs</td><td>rd</td></tr></table>	1010	rs	rd	$rs - rd - cy \rightarrow rd$					
1010	rs	rd								
INC rd	<table><tr><td>1011</td><td></td><td>rd</td></tr></table>	1011		rd	$rd + 1 \rightarrow rd$					
1011		rd								
AND rs, rd	<table><tr><td>1100</td><td></td><td>rd</td></tr></table>	1100		rd	$rs \wedge rd \rightarrow rd$					
1100		rd								
COM rd	<table><tr><td>1101</td><td></td><td>rd</td></tr></table>	1101		rd	$\overline{rd} \rightarrow rd$					
1101		rd								
RRC rs, rd	<table><tr><td>1110</td><td></td><td>rd</td></tr></table>	1110		rd						
1110		rd								
RLC rs, rd	<table><tr><td>1111</td><td></td><td>rd</td></tr></table>	1111		rd						
1111		rd								
LDA M, D, rd	<table><tr><td>00</td><td>M</td><td>00</td><td>rd</td></tr><tr><td colspan="4">D</td></tr></table>	00	M	00	rd	D				$E \rightarrow rs$
00	M	00	rd							
D										
STA M, D, rd	<table><tr><td>00</td><td>M</td><td>01</td><td>rd</td></tr><tr><td colspan="4">D</td></tr></table>	00	M	01	rd	D				$rd \rightarrow E$
00	M	01	rd							
D										
JMP M, D	<table><tr><td>00</td><td>M</td><td>10</td><td>rd</td></tr><tr><td colspan="4">D</td></tr></table>	00	M	10	rd	D				$E \rightarrow PC$
00	M	10	rd							
D										
BZC M, D	<table><tr><td>00</td><td>M</td><td>11</td><td>rd</td></tr><tr><td colspan="4">D</td></tr></table>	00	M	11	rd	D				当 CY=1 或 Z=1 时, $E \rightarrow PC$
00	M	11	rd							
D										
IN addr, rd	<table><tr><td>0100</td><td>01</td><td>rd</td></tr></table>	0100	01	rd	$addr \rightarrow rd$					
0100	01	rd								
OUT addr, rd	<table><tr><td>0101</td><td>10</td><td>rd</td></tr></table>	0101	10	rd	$rd \rightarrow addr$					
0101	10	rd								
HALT	<table><tr><td>0110</td><td>00</td><td>00</td></tr></table>	0110	00	00	停机					
0110	00	00								

五、总体设计

复杂模型机的数据通路框图如图 5—1 所示。根据复杂模型机的硬件电路设计监控软件(机器指令)，再根据机器指令要求，设计微程序流程图及微程序，最后形成 16 进制文件。

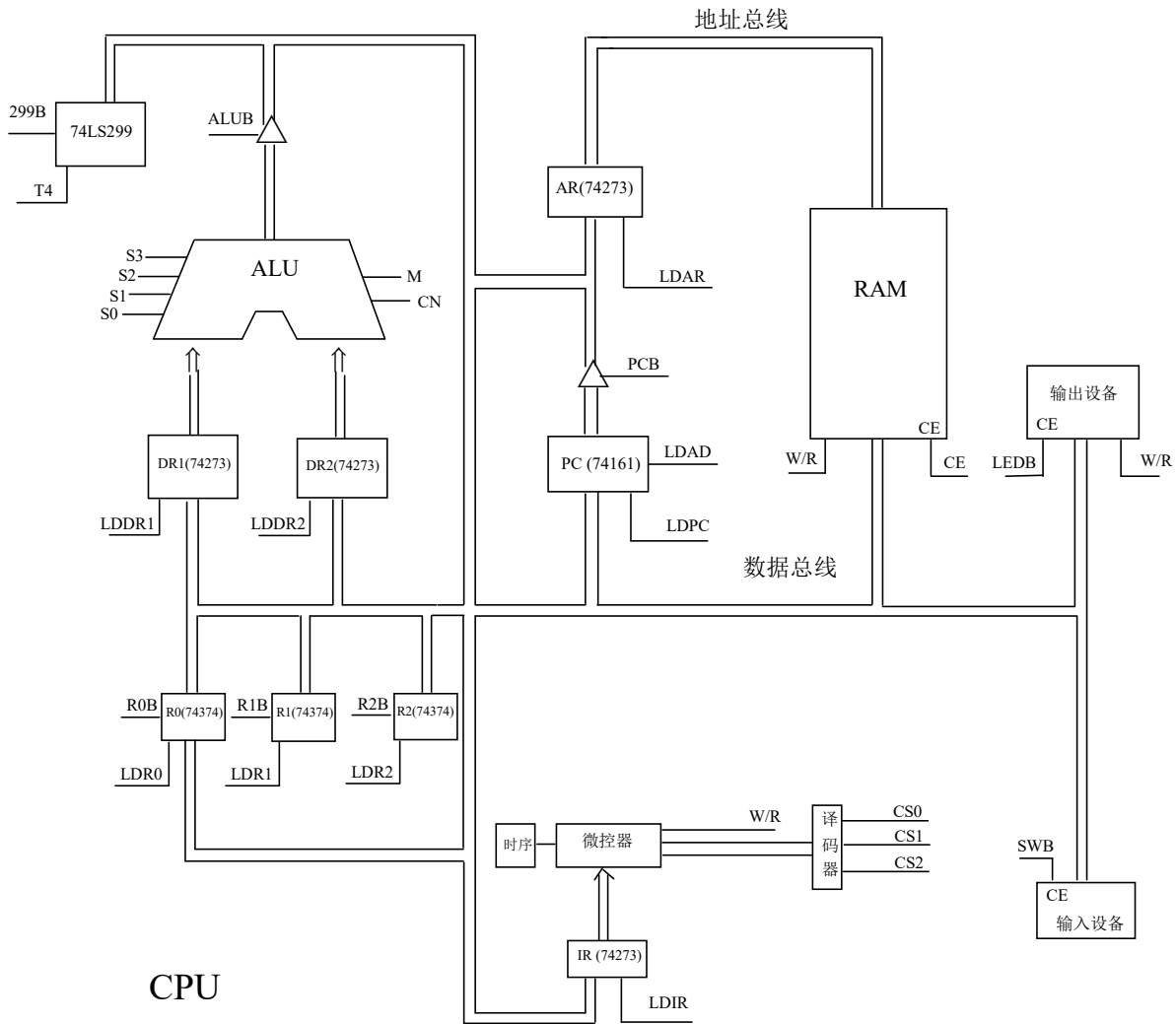


图 5-1 数据通路框图

六、实验步骤

1. 设计复杂模型机的监控软件，详细如下：

RAM 地址 (16 进制)	内容 (16 进制)	助记符	说 明
00	00	IN	“INPUT DEVICE” → R0
01	10	ADD [0DH]	R0+[0DH] → R0
02	0D		
03	80	RLC	
04	00	IN	“INPUT DEVICE” → R0
05	60	RRC	
06	70	RL	
07	20	STA [0EH]	R0 → [0EH]
08	0E		
09	30	OUT [0EH]	[0EH] → OUTPUT
0A	0E		

0B	40	JMP [addr]	00→ PC
0C	00		自定
0D	45		存数单元
0E			

2. 在 QuartusII 环境下，打开基本模型机工程文件 bdf，将 sof 文件下载到实验台。了解此 CPU 设计示例，包括引脚锁定与实验系统控制键的关系等。
3. 利用 QuartusII 的 In-System Memory Content Editor 工具观察 ROM、RAM 中预置的数据。
4. “模式选择”置 1，按一次“系统复位”按钮。
5. 通过键 2、键 1 输入运算数据，第一个数用批次号，第二个数用学生的序号。
6. 按动键 7（注意每按 2 次产生一个单步脉冲），使 CPU 单步（微指令单步）运行。即每按 2 次键 7 产生一个 STEP 单步脉冲，执行一个微操作。通过液晶显示屏显示数据观察微程序的执行过程。
7. 用自己的运算数据完成表3-5的“STEP”、“后续uA 微地址”、“MC 微指令”、“PC”、“IR指令”、“执行结果”这6列数据。“完成功能”这一列写实验报告时完成。

七、实验要求

1. 实验之前应认真准备，熟悉实验步骤和具体实验内容。
2. 用In-System Memory Content Editor观察ROM、RAM中的数据。
3. 通过液晶屏，观察各相关寄存器、ALU、DR1、PC、IR、AR、BUS、MC等内容的变化情况，根据微程序控制流程，单步跟踪微程序的执行情况。
4. 通过INPUT（键2、键1）输入运算数据（序号），跟踪程序的执行情况，并详细记录每条微指令执行后，相关单元输出数据的变化情况，依次执行机器指令，记录实验数据（表4-5）。
5. 完成实验报告时，按自己的理解填写表4-5“完成功能”这一列。

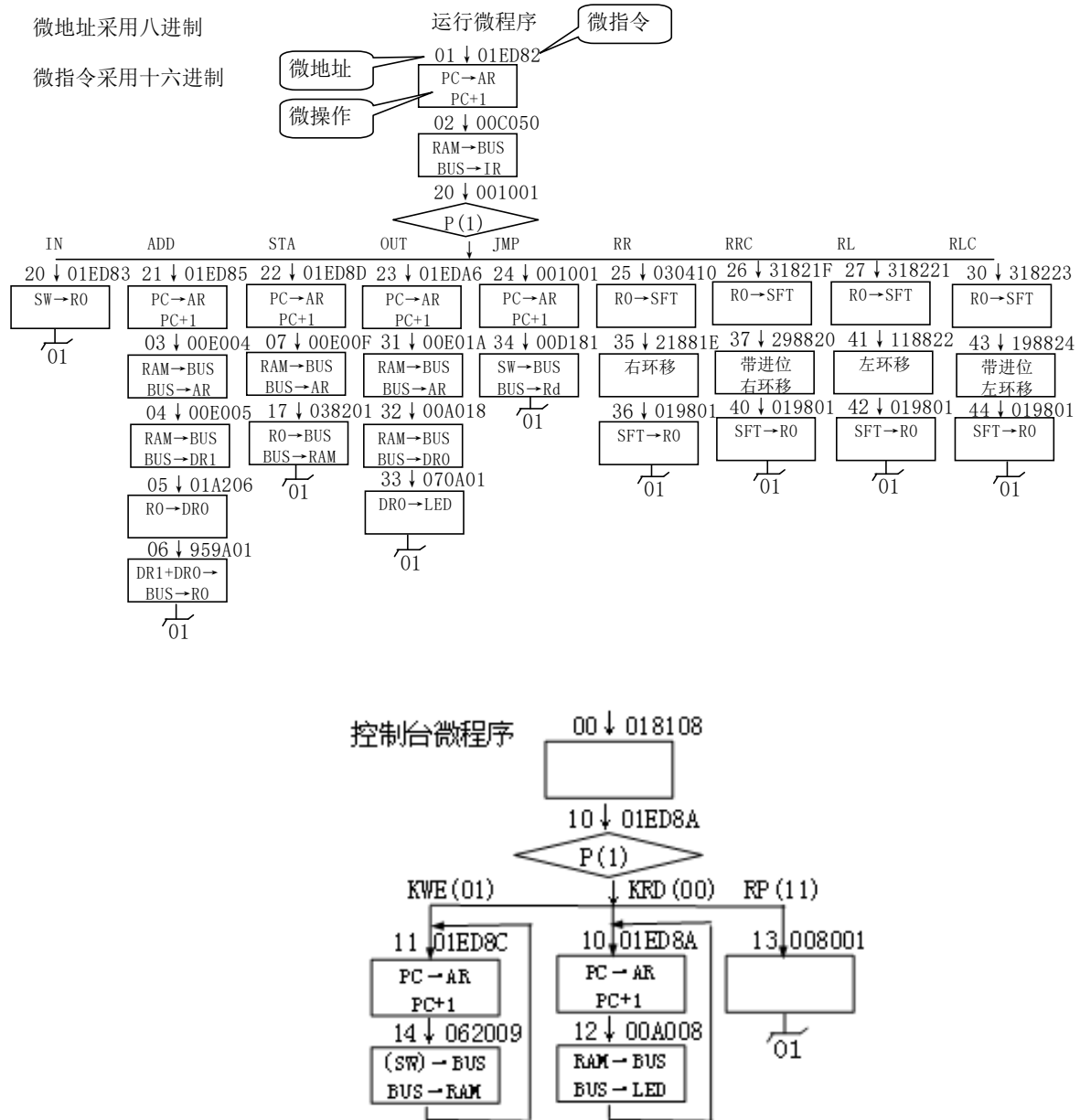


图 5-3 微程序流程图

附录 A 实验电路结构图

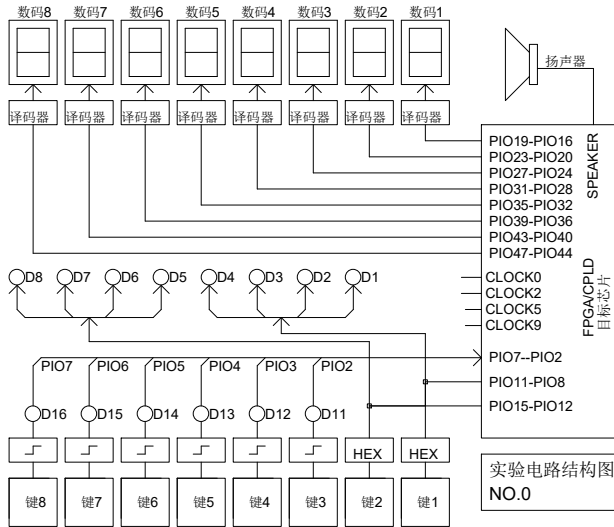


图 3 实验电路结构图 NO.0

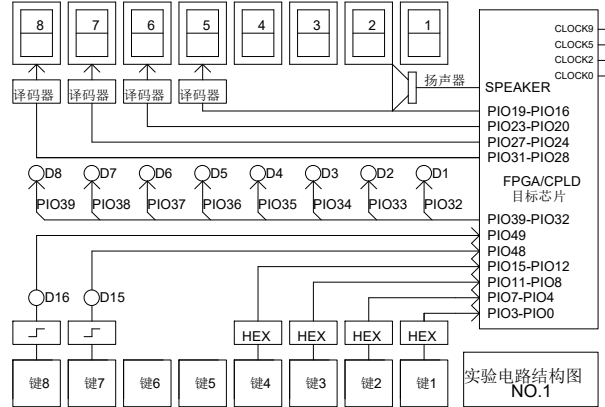


图 4 实验电路结构图 NO.1

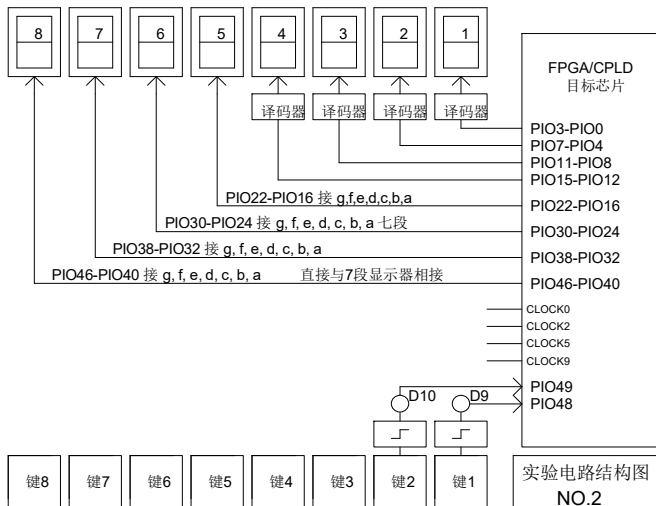


图 5 实验电路结构图 NO.2

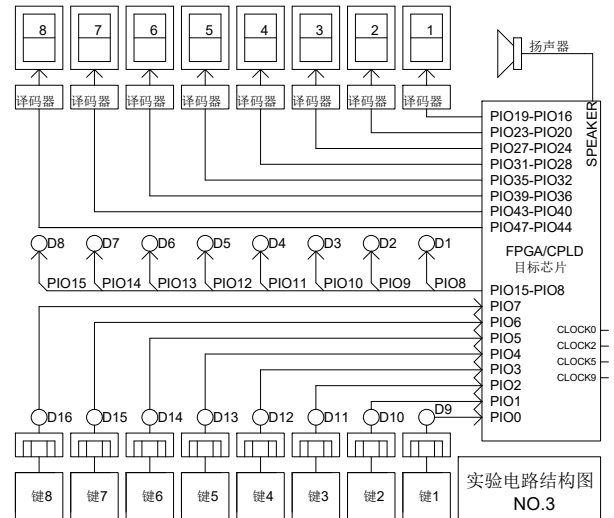
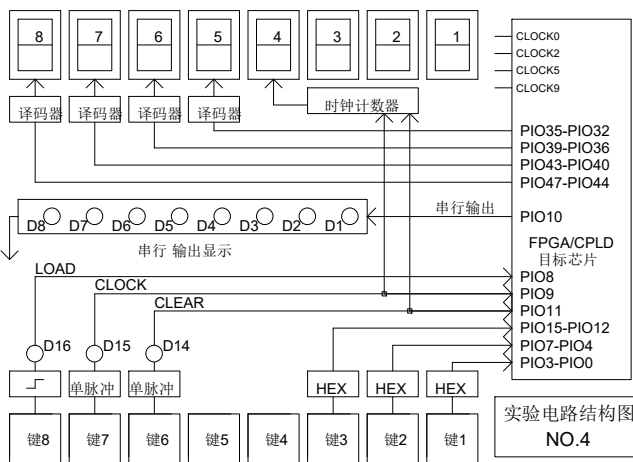
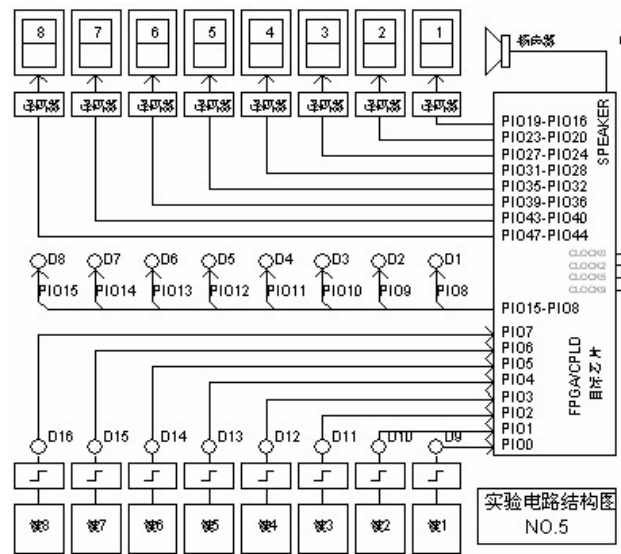


图 6 实验电路结构图 NO.3



实验电路结构图 NO.4



实验电路结构图 NO.5

图 7 实验电路结构图 NO. 4

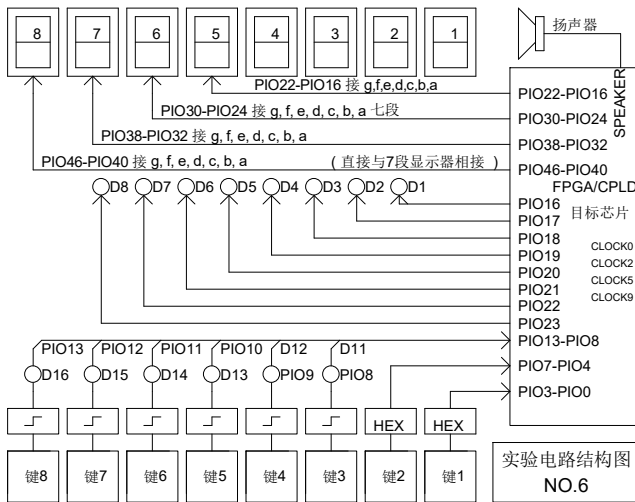


图 9 实验电路结构图 NO. 6

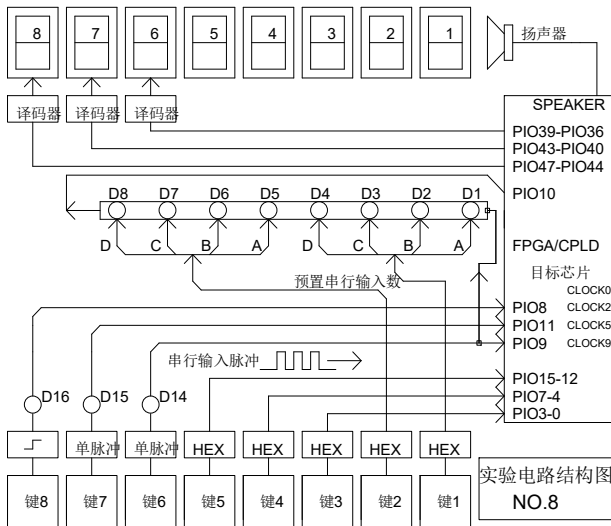


图 11 实验电路结构图 NO. 8

图 8 实验电路结构图 NO.5

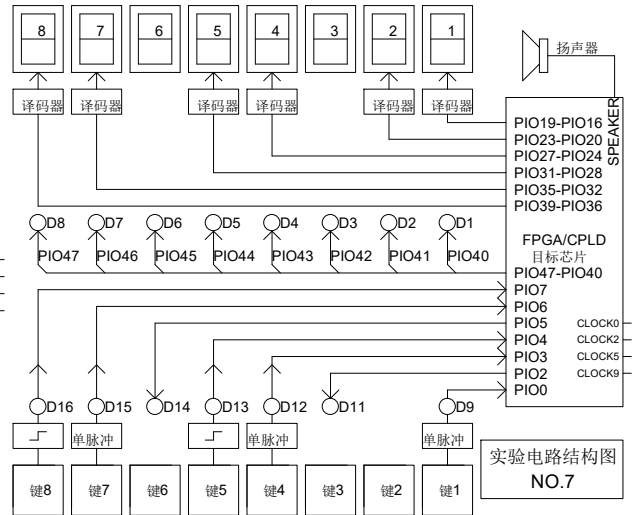


图 10 实验电路结构图 NO. 7

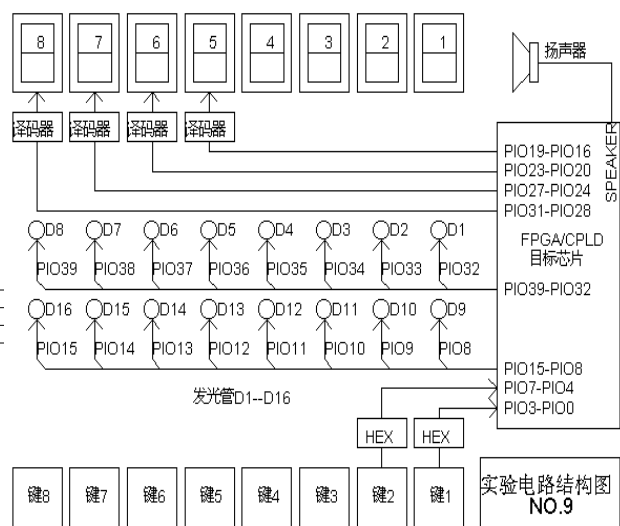


图 12 实验电路结构图 NO. 9

附录 B 实验电路结构图说明

1. 实验电路信号资源符号图说明

结合附图 2-1，以下对实验电路结构图中出现的信号资源符号功能作出一些说明：

(1) 附图 2-1a 是 16 进制 7 段全译码器，它有 7 位输出，分别接 7 段数码管的 7 个显示输入端：a、b、c、d、e、f 和 g；它的输入端为 D、C、B、A，D 为最高位，A 为最低位。例如，若所标输入的口

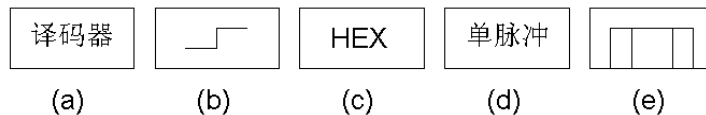


图 2-1 实验电路信号资源符号图

线为 PI019~16，表示 PI019 接 D、18 接 C、17 接 B、16 接 A。

(2) 附图 2-1b 是高低电平发生器，每按键一次，输出电平由高到低、或由低到高变化一次，且输出为高电平时，所按键对应的发光管变亮，反之不亮。

(3) 附图 2-1c 是 16 进制码 (8421 码) 发生器，由对应的键控制输出 4 位 2 进制构成的 1 位 16 进制码，数的范围是 0000~1111，即[^]H0 至[^]HF。每按键一次，输出递增 1，输出进入目标芯片的 4 位 2 进制数将显示在该键对应的数码管上。

(4) 附图 2-1d 是单次脉冲发生器。每按一次键，输出一个脉冲，与此键对应的发光管也会闪亮一次，时间 20ms。

(5) 附图 2-e 是琴键式信号发生器，当按下键时，输出为高电平，对应的发光管发亮；当松开键时，输出为高电平，此键的功能可用于手动控制脉冲的宽度。具有琴键式信号发生器的实验结构图是 NO. 3。

2. 各实验电路结构图特点与适用范围简述

(1) **结构图 NO. 0:** 目标芯片的 PI016 至 PI047 共 8 组 4 位 2 进制码输出，经外部的 7 段译码器可显示于实验系统上的 8 个数码管。键 1 和键 2 可分别输出 2 个四位 2 进制码。一方面这四位码输入目标芯片的 PI011~PI08 和 PI015~PI012，另一方面，可以观察发光管 D1 至 D8 来了解输入二进制的数值。例如，当键 1 控制输入 PI011~PI08 的数为“B”时，则发光管 D4 和 D2 亮，D3 和 D1 灭。电路的键 8 至键 3 分别控制一个高低电平信号发生器向目标芯片的 PI07 至 PI02 输入高电平或低电平，扬声器接在“SPEAKER”上，具体接在哪一引脚要看目标芯片的类型，这需要查第 3 节的引脚对照表。如目标芯片为 EPEC6/12，则扬声器接在“174”引脚上。目标芯片的时钟输入未在图上标出，也需查阅第 3 节的引脚对照表。例如，目标芯片为 EP1C6，则输入此芯片的时钟信号有 CLOCK0 或 CLOCK9，共 4 个可选的输入端，对应的引脚为 28 或 29。具体的输入频率，可参考主板频率选择模块。此电路可用于设计频率计，周期计，计数器等等。

(2) **结构图 NO. 1:** 适用于作加法器、减法器、比较器或乘法器等。例如，加法器设计，可利用键 4 和键 3 输入 8 位加数；键 2 和键 1 输入 8 位被加数，输入的加数和被加数将显示于键对应的数码管 4-1，相

加的和显示于数码管 6 和 5；可令键 8/7 控制此加法器的最低位进位。

(3) **结构图 NO. 2:** (4) 直接与 7 段数码管相连的连接方式的设置是为了便于对 7 段显示译码器的设计学习。以图 NO. 2 为例，如图所标“PI046-PI040 接 g、f、e、d、c、b、a”表示 PI046、PI045..PI040 分别与数码管的 7 段输入 g、f、e、d、c、b、a 相接。

可用于作 VGA 视频接口逻辑设计，或使用数码管 8 至数码管 5 共 4 个数码管作 7 段显示译码方面的实验；而数码管 4 至数码管 1，4 个数码管可作译码后显示，键 1 和键 2 可输入高低电平。

(4) **结构图 NO. 3:** 特点是有 8 个琴键式键控发生器，可用于设计八音琴等电路系统。也可以产生时间长度可控的单次脉冲。该电路结构同结构图 NO. 0 一样，有 8 个译码输出显示的数码管，以显示目标芯片的 32 位输出信号，且 8 个发光管也能显示目标器件的 8 位输出信号。

(5) **结构图 NO. 4:** 适合于设计移位寄存器、环形计数器等。电路特点是，当在所设计的逻辑中有串行 2 进制数从 PI010 输出时，若利用键 7 作为串行输出时钟信号，则 PI010 的串行输出数码可以在发光管 D8 至 D1 上逐位显示出来，这能很直观地看到串出的数值。

(6) **结构图 NO. 5:** 8 键输入高低电平功能，目标芯片的 PI019 至 PI044 共 8 组 4 位 2 进制码输出，经外部的 7 段译码器可显示于实验系统上的 8 个数码管。

(7) **结构图 NO. 6:** 此电路与 NO. 2 相似，但增加了两个 4 位 2 进制数发生器，数值分别输入目标芯片的 PI07~PI04 和 PI03~PI00。例如，当按键 2 时，输入 PI07~PI04 的数值将显示于对应的数码管 2，以便了解输入的数值。

(8) **结构图 NO. 7:** 此电路适合于设计时钟、定时器、秒表等。因为可利用键 8 和键 5 分别控制时钟的清零和设置时间的使能；利用键 7、5 和 1 进行时、分、秒的设置。

(9) **结构图 NO. 8:** 此电路适用于作并进/串出或串进/并出等工作方式的寄存器、序列检测器、密码锁等逻辑设计。它的特点是利用键 2、键 1 能序置 8 位 2 进制数，而键 6 能发出串行输入脉冲，每按键一次，即发一个单脉冲，则此 8 位序置数的高位在前，向 PI010 串行输入一位，同时能从 D8 至 D1 的发光管上看到串形左移的数据，十分形象直观。

(10) **结构图 NO. 9:** 若欲验证交通灯控制等类似的逻辑电路，可选此电路结构。

(11) 当系统上的“模式指示”数码管显示“A”时，系统将变成一台频率计，数码管 8 将显示“F”，“数码 6”至“数码 1”显示频率值，最低位单位是 Hz。测频输入端为系统板右下侧的插座。

附录 C 结构图信号/与芯片引脚对照表

结构图上的 的信号名	GW2C35 EP2C35FBGA4 84C8 CycloneII	GW3C40 EP3C40Q24 0C8N CycloneII I	GWXS200 XC3S200 SPARTAN
	引脚号	引脚号	引脚号
PI00	AB15	18	21
PI01	AB14	21	22
PI02	AB13	22	24
PI03	AB12	37	26
PI04	AA20	38	27
PI05	AA19	39	28
PI06	AA18	41	29
PI07	L19	43	31
PI08	J14	44	33
PI09	H15	45	34
PI010	H14	46	15
PI011	G16	49	16
PI012	F15	50	35
PI013	F14	51	36
PI014	F13	52	37
PI015	L18	55	39
PI016	L17	56	40
PI017	K22	57	42
PI018	K21	63	43
PI019	K18	68	44
PI020	K17	69	45
PI021	J22	70	46
PI022	J21	73	48
PI023	J20	76	50
PI024	J19	78	51
PI025	J18	80	52
PI026	E11	112	113
PI027	E9	113	114
PI028	E8	114	115
PI029	E7	117	116
PI030	D11	118	117
PI031	D9	126	119
PI032	D8	127	120
PI033	D7	128	122
PI034	C9	131	123

PI035	H7	132	123
PI036	Y7	133	125
PI037	Y13	134	126
PI038	U20	135	128
PI039	K20	137	130
PI040	C13	139	131
PI041	C7	142	132
PI042	H3	143	133
PI043	U3	144	135
PI044	P3	145	137
PI045	F4	146	138
PI046	C10	159	139
PI047	C16	160	140
PI048	G20	161	141
PI049	R20	162	143
PI060	AB16	226	2
PI061	AB17	230	3
PI062	AB18	231	4
PI063	AB19	232	5
PI064	AB20	235	7
PI065	AB7	236	9
PI066	AB8	239	10
PI067	AB11	240	11
PI068	A10	186	161
PI069	A9	185	156
PI070	A8	184	155
PI071	A7	183	154
PI072	A6	177	152
PI073	A5	176	150
PI074	A4	173	149
PI075	A3	171	148
PI076	AB9	6	12
PI077	AB10	9	13
PI078	B5	169	147
PI079	Y10	166	146
SPEAKER	Y16	164	144
CLOCK0	L1	152	184
CLOCK2	M1	149	203
CLOCK5	M22	150	204
CLOCK9	B12	151	205