

# 《计算机组成与设计》

## 实验指导书

2018.2

山东大学

## 目录

实验 1 基本逻辑门逻辑实验.....	3
实验 2 逻辑运算电路.....	4
实验 3 补码加法器.....	5
实验 4 移位器.....	7
实验 5 四位补码运算器.....	9
实验 6 二进制补码加法器实验.....	10
实验 7 节拍脉冲发生器时序电路实验.....	12
实验 8 时序系统实验.....	14
实验 9 七段译码设计.....	15
实验 10 LPM_ROM 实验 .....	16
实验 11 LPM_RAM 实验.....	18
实验 12 RAM 扩展实验.....	20
实验 13 控制器实验.....	21
实验 14 综合实验.....	24
附录 常用实验器件引脚图.....	26

# 实验 1 基本逻辑门逻辑实验

## 一、实验目的

- 1. 掌握 TTL 常用逻辑门输入与输出之间的逻辑关系。
- 2. 熟悉 TTL 中、小规模集成电路的外型、管脚和使用方法。

## 二、实验所用 QuartII 库内器件

- 1. 二输入四异或门 74LS86
- 2. 三态门 TRI
- 3. 四位二进制计数器 74LS161
- 4. 3-8 译码器 74LS138
- 5. 双向移位寄存器 74LS194
- 6. 8 位寄存器 74LS74

## 三、实验内容

测试 74LS86、tri、138、161、74LS194、74LS74 集成电路模块，分析其输入和输出之间的逻辑关系。

## 四、实验提示

- 1. 在 quartii 环境下，建立工程文件，调入被测器件，将其输入管脚绑定在相应的开关上，输出管脚绑定在相应的指示灯上，建立源文件。
- 2. 将该文件编译下载到实验平台上的 FPGA 内，按照所测器件的功能表逐条验证。
- 3. 用实验台的电平开关输出作为被测器件的输入。拨动开关，则改变器件的输入电平。
- 4. 将被测器件的输出引脚与实验台上的电平指示灯连接。指示灯亮表示输出电平为 1，指示灯灭表示输出电平为 0。

## 五、所测器件的功能表和管脚见附录。

例：实验接线图及实验结果（真值表及语言描述）

（1）74LS86 中包含 4 个异或门，可测试一个异或门即可，如图 1 所示。

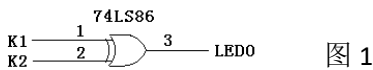


图 1

（2）可使平台工作于模式 5，图中的 K1、K2 绑定在 FPGA 的 PIO0（52）、PIO1（55）两个管脚上，LED0 绑定在 FPGA 的 PIO8（60）管脚上。

（3）测试 74LS86 逻辑关系接线图及测试结果

输 入		输 出
引脚 1	引脚 2	引脚 3
L	L	L
L	H	H
H	L	H
H	H	L

## 实验 2 逻辑运算电路

### 1、实验内容及说明

本实验要求设计一个能实现 1 位逻辑乘  $ab$ 、逻辑或  $a+b$ 、半加( $a\oplus b$ )的逻辑运算电路。图 2 为实现上述逻辑功能的电路原理图,其中参与运算的两个 1 位二进制数为  $a$  和  $b$ ,  $and$ 、 $xor$  和  $or$  分别为与运算、异或运算和或运算控制输入端。

### 2、实验步骤

(1) 原理图输入: 根据图 2 所示电路,完成逻辑运算的电路原理图设计。

(2) 管脚锁定: 完成原理图中输入、输出的管脚锁定。

可使平台工作于模式 5,将  $a$  操作数锁定在键 1 上;将  $b$  操作数锁定在键 2 上;将输出  $e$  锁定在 D1 上;将与运算控制输入  $and$  端锁定在键 3 上;将或运算控制输入  $or$  端锁定在键 4 上;将异或运算控制输入  $xor$  端锁定在键 5 上。

(3) 原理图编译、适配和下载: 在 Quartus II 环境中选择 EP4CE6/10 器件,进行原理图的编译和适配,无误后完成下载。

(4) 功能测试: 利用输入开关及发光二极管 LD 测试逻辑运算部件的功能并记录测试结果。

(5) 生成元件符号。

### 3、选做

利用一位逻辑运算的结果实现两个 4 位二进制数  $A(a_3a_2a_1a_0)$  和  $B(b_3b_2b_1b_0)$  的逻辑运算并生成元件符号。使平台工作于模式 5,当按键开关不足时,可使用平台上红色的拨码开关。

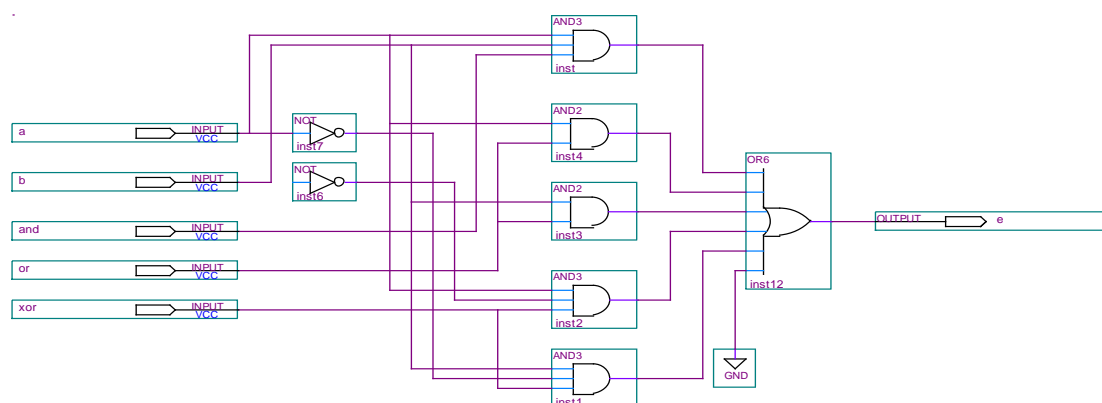


图 2 逻辑运算电路原理图

## 实验 3 补码加法器

### 1、实验内容及说明

本实验要求采用一位全加器和异或门设计一个四位二进制补码加法器，操作数和运算结果均采用补码表示，该加法器要求可做补码加法，也可做补码减法运算。

在补码运算中：

$$(X)_{\text{补}} + (Y)_{\text{补}} = (X + Y)_{\text{补}}$$

$$(X)_{\text{补}} - (Y)_{\text{补}} = (X)_{\text{补}} + (-Y)_{\text{补}}$$

图 3 是四位补码加法器的原理图，图中 SUM3-SUM0 表示四位和数，C4 为向高位的进位，A3-A0 为 A 操作数，B3-B0 为 B 操作数，C0 为低位来的进位，K 为控制端，当 K 为高电平时，执行减法运算，K 为低电平时，执行加法运算。

### 2. 实验步骤

(1) 原理图输入：从元件库中选取相应器件实现四位补码加法器电路设计。

(2) 管脚锁定：平台工作于模式 5，将 SUM3-SUM0、C4 依次定义在 LED 指示灯 D5-D1 上，将 K、C0、A3-A0、B3-B0 依次绑定在红色拨码开关 dout1、dout2、键 8-键 5、键 4-键 1 上。

(3) 原理图编译、适配和下载：在 Quartus II 环境中选择 **EP4CE6/10E 器件**，进行原理图的编译和适配，无误后完成下载。

(4) 功能测试：利用输入开关键改变 K、C0 和 A、B 操作数的值，看 LED 指示灯显示的结果是否正确并记录结果。

(5) 生成元件符号。

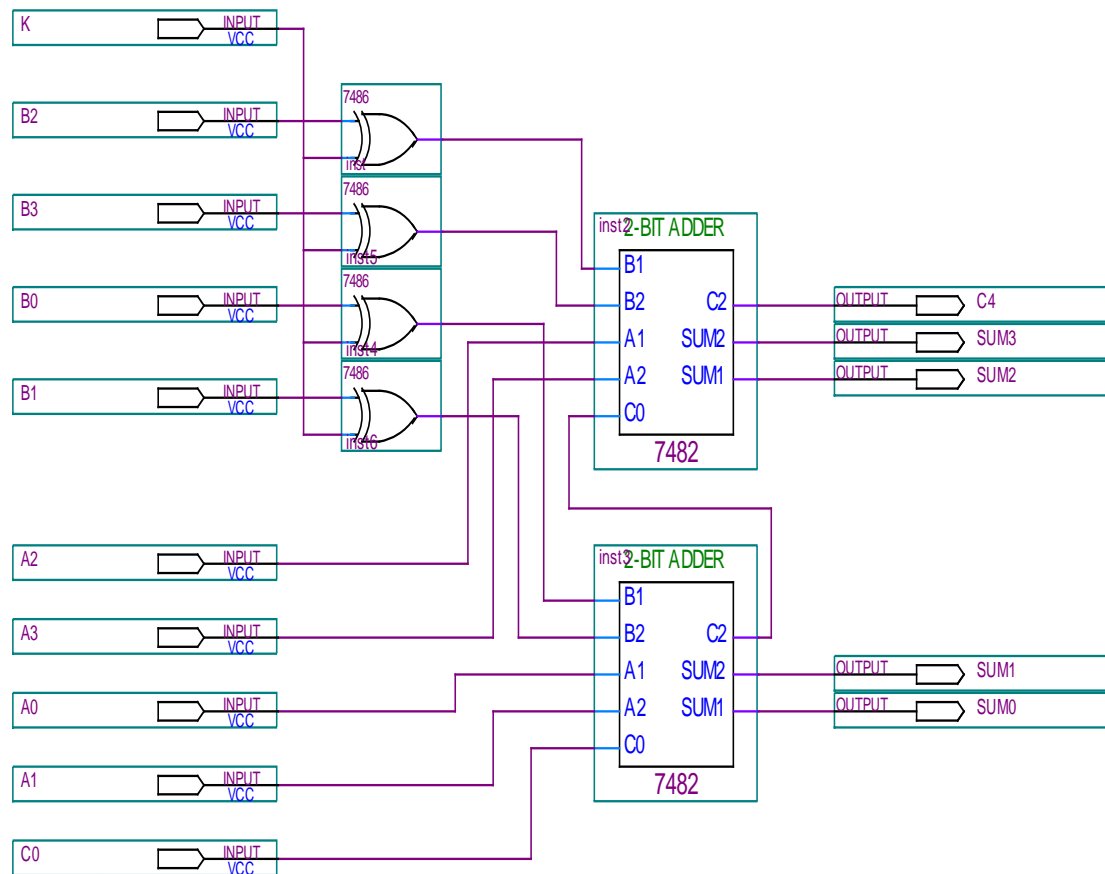


图 3 四位补码加法器原理图

## 实验 4 移位器

### 1、实验内容及说明

本实验要求采用传送方式实现二进制数的移位电路。图 4.4 给出了可对四位二进制数实现左移 1 位 ( $\times 2$ )，右移 1 位 ( $\div 2$ ) 和直接传送功能的移位线路，这也是运算器的主要功能。

在 LM（左移）的控制下可实现左移 1 位，空位补 0。

在 RM（右移）的控制下可实现右移 1 位，空位补 0。

在 DM（直送）的控制下可实现直接传送。

### 2、实验步骤

（1）用图形输入法完成图 4.3 逻辑电路输入。

（2）管脚锁定：平台工作于模式 5，将四位二进制数 a3-a0 定义在键 4—键 1 上；将 4 位输出 q3-q0 定义在 D4—D1 上；将 LM 定义在键 5 上，高电位有效；将 DM 定义在键 6 上，高电位有效；将 RM 定义在键 7 上，高电位有效，完毕后下载。

（3）设置键 4—键 1 为任意 4 位数，在 LM、DM、RM 的作用下分别观察 D3—D0 的显示，并分析其正确性。

（4）生成元件符号。

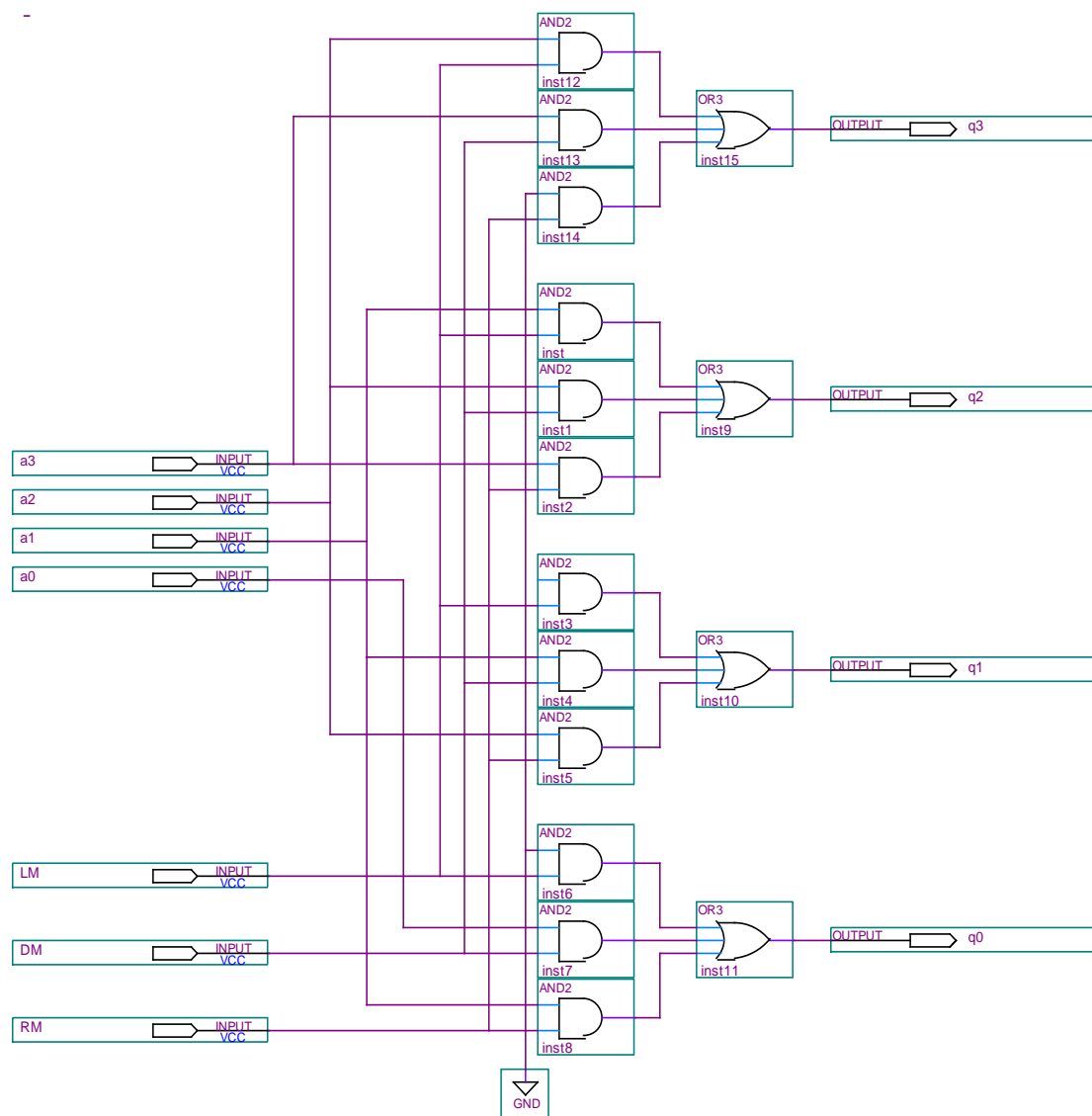


图 4 移位器电路原理图



# 实验 5 四位补码运算器

## 1、实验内容及说明

本实验要求设计一个能够实现补码加法、减法、加 1、左移、右移、直接传送等功能的四位补码运算器，图 5 为四位补码运算器的电路框图,图中，移位器具有左移、右移和直接传送功能。

实验时，图 5 中的元件可直接调用，其中 R0、R1、R2 四位寄存器可直接调用元件库中的 4D 寄存器 74173（设计原理图时 74173 的 G1N、G2N、MN、NN 和 CLR 管脚接地），移位器可调用本章实验 4 中设计的移位器元件，四位加法器可调用本章实验 3 中设计的补码加法器元件。

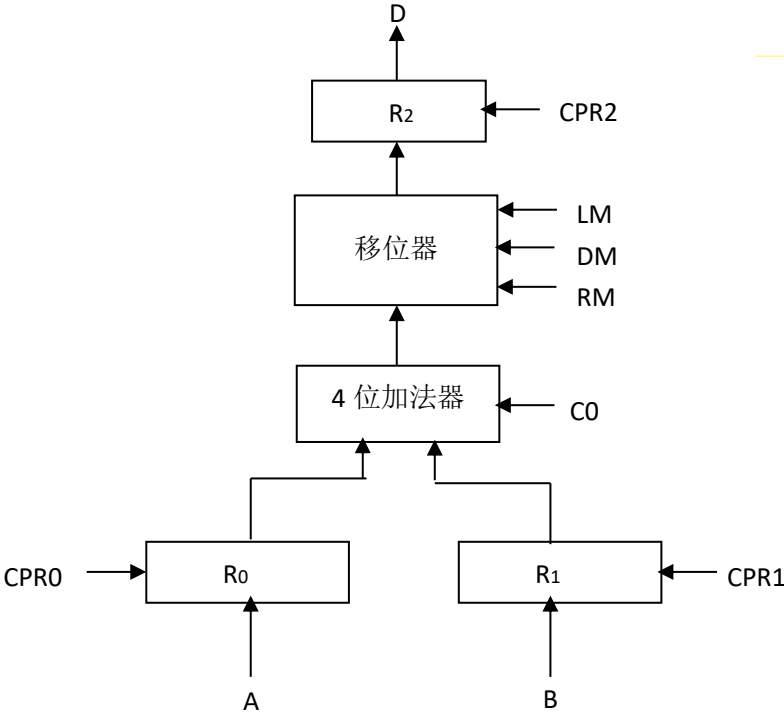


图 5 四位补码运算器电路框图

## 2、实验步骤

（1）原理图输入：调用寄存器、移位器、补码加法器等元件根据图 5 完成四位补码运算器电路设计、调试并生成元件符号。

（2）管脚锁定：平台工作于模式 5，将 4 位操作数 A（a3a2a1a0）锁定在键 8-键 5 上、将 4 位操作数 B(b3b2b1b0)锁定在键 4-键 1 上、将打入脉冲 CPR0 锁定在拨码开关 Dout1 上、

将打入脉冲 CPR1 锁定在拨码开关 Dout2 上、将打入脉冲 CPR2 锁定在拨码开关 Dout3 上、将 LM 锁定在拨码开关 Dout4 上、将 DM 锁定在拨码开关 Dout5 上、将 RM 锁定在拨码开关 Dout6 上、将 C0（低位的进位）锁定在拨码开关 Dout7 上。

拨码开关 Dout 由低电平拨向高电平再拨向低电平，相当于一个脉冲。

将存放结果的寄存器 R2 的输出端 D(d3d2d1d0)分别锁定在 D4—D1 上。

（3）管脚锁定完毕后在 Quartus II 中选择 EP4CE6/10E 器件，进行原理图的编译和适配，无误后完成下载。

（4）功能测试：利用开关与指示灯测试：加法、减法、加 1、左移、右移、直传等功能，并分析结果正确性。

## 实验 6 二进制补码加法器实验

### 一、实验目的：

根据补码加法器的模型，理解数据流及其时序关系。

掌握加法器实现补码加、减运算的基本原理。

### 二、实验方案：

本实验运算器模型，可分为数据运算以及符号位的产生两部分。

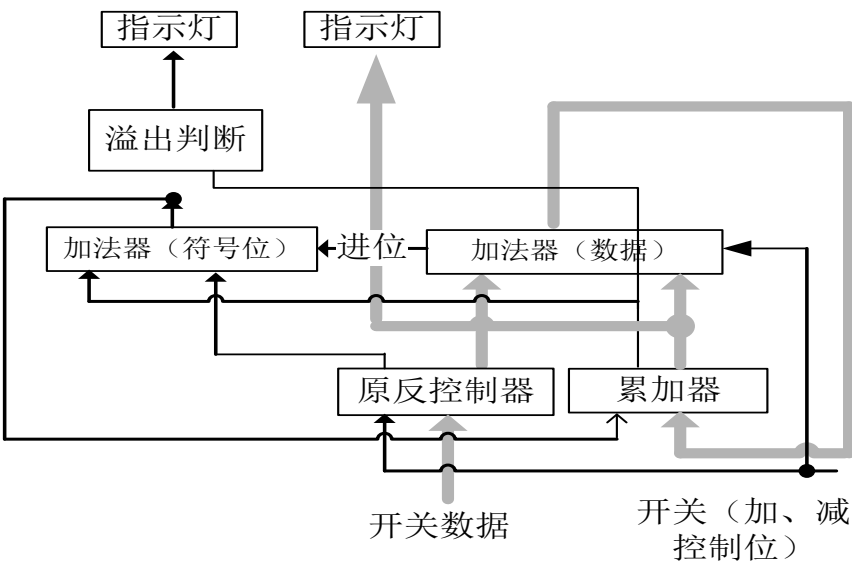


图 10 补码加、减运算器结构图

### 三、实验要求：

- ◆ 数据宽度为 4 位，设计出实验线路图。
- ◆ 设计试验步骤。
- ◆ 使用开关进行数据加载，完成补码加、减运算。
- ◆ 符号位运算采用双符号位，累加器应有清零控制。
- ◆ 通过指示灯观察运算结果，记录实验现象，写出实验报告。

#### 四、参考器件：

累加器选用一片 74LS273；加法器用两片 74 LS283；原、反码控制器用一片 74LS86.

溢出判断用一片 74LS86。

## 实验 7 节拍脉冲发生器时序电路实验

1、实验目的：掌握节拍脉冲发生器的设计方法，理解节拍脉冲发生器的工作原理。

2、实验原理：连续节拍发生电路可由 4 个 D 触发器组成（见图 11-1），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，由实验台右边的方波信号源 clock0 提供，clock0 具有 1Hz~50MHz 的多种方波信号频率。实验者可根据实验自行选择信号频率。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（EXEC）。

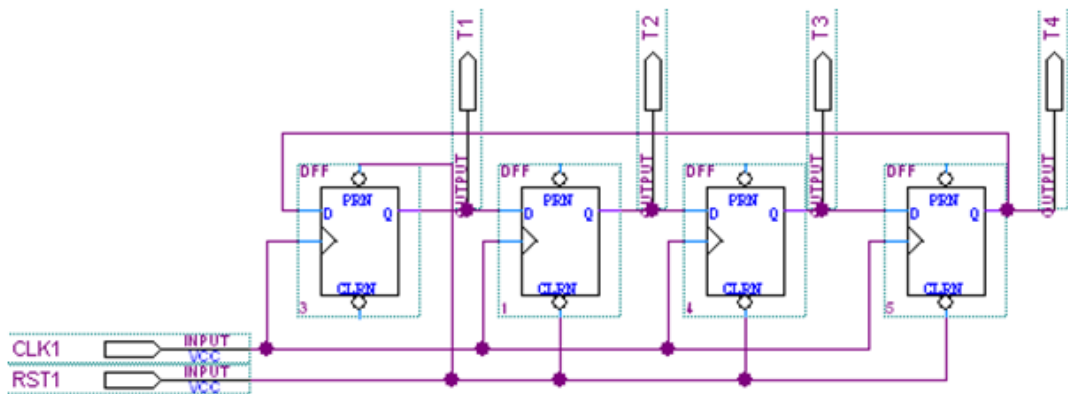


图 11-1 节拍脉冲发生器的工作原理

T1~T4 以及 CLK1、RST1 的工作波形如图 11-2 所示。

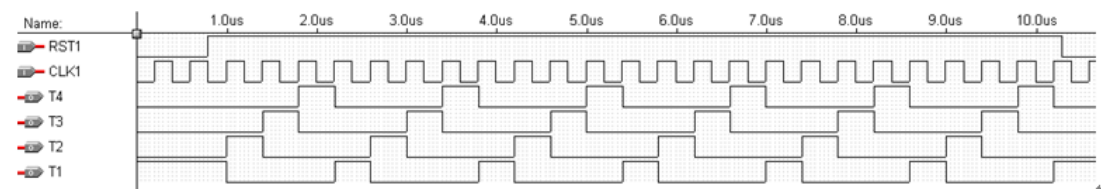


图 11-2 节拍脉冲发生器工作波形

3、实验任务：

### （1）连续节拍发生电路设计

设计工程文件，硬件电路如图 11-1 所示。使实验平台工作于模式 5，主系统时钟源接 4Hz，键 8 控制 RST1，高电平时可以看到，发光管 D1、D2、D3、D4 分别显示 T1、T2、T3、T4 的输出电平，锁定引脚并硬件下载测试。引脚锁定后进行编译、下载和硬件测试实验。将实验过程和实验结果写进实验报告。

### （2）单步节拍发生电路设计

用单步节拍发生电路可以对微程序进行单步运行调试，电路如图 11-3 所示。该电路每

当 RST1 出现一个负脉冲后，仅输出一组 T1、T2、T3、T4 节拍信号，直到 RST1 出现下一个负脉冲，波形如图 11-4 所示。

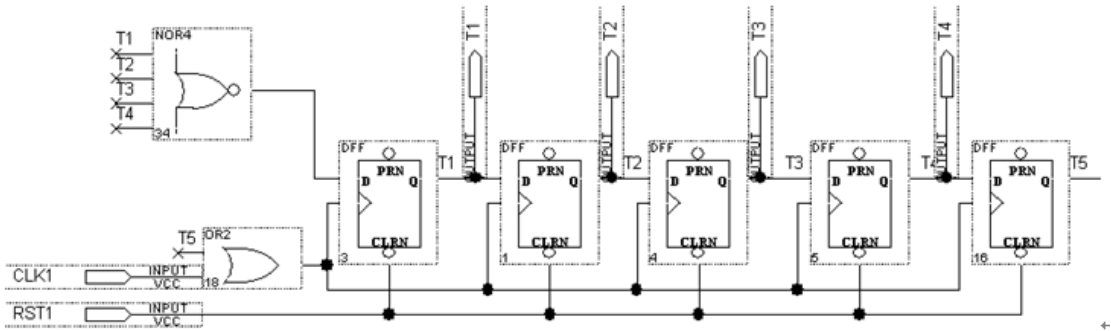


图 11-3 单步运行电路工作原理

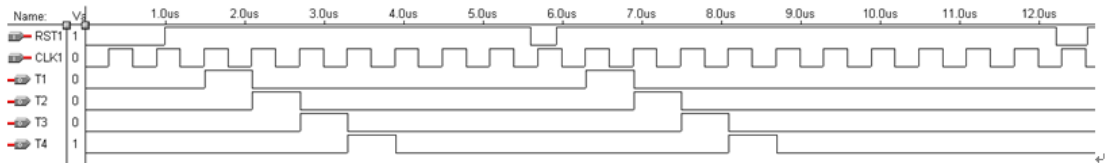


图 11-4 单步运行电路工作波形

设计工程文件，硬件电路如图 11-3 所示。使实验平台工作于模式 5，主系统时钟源接 4Hz，键 8 控制 RST1，高电平时可以看到，发光管 D1、D2、D3、D4 分别显示 T1、T2、T3、T4 的输出电平，锁定引脚并硬件下载测试。引脚锁定后进行编译、下载和硬件测试实验。将实验过程和实验结果写进实验报告。

# 实验 8 时序系统实验

## 1、实验目的：

掌握计算机实验中时序系统的设计方法。设计一个基本时序系统，该系统具有 4 个节拍电平及四相工作脉冲，其时序关系参阅下图中的 M0—M3，T0—T3。

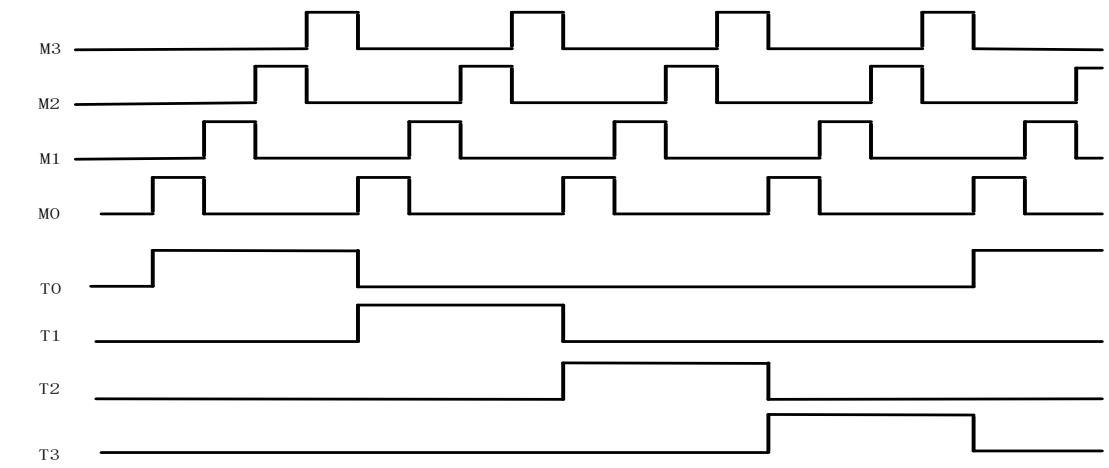


图 12-1 时序图

## 2、实验方案：

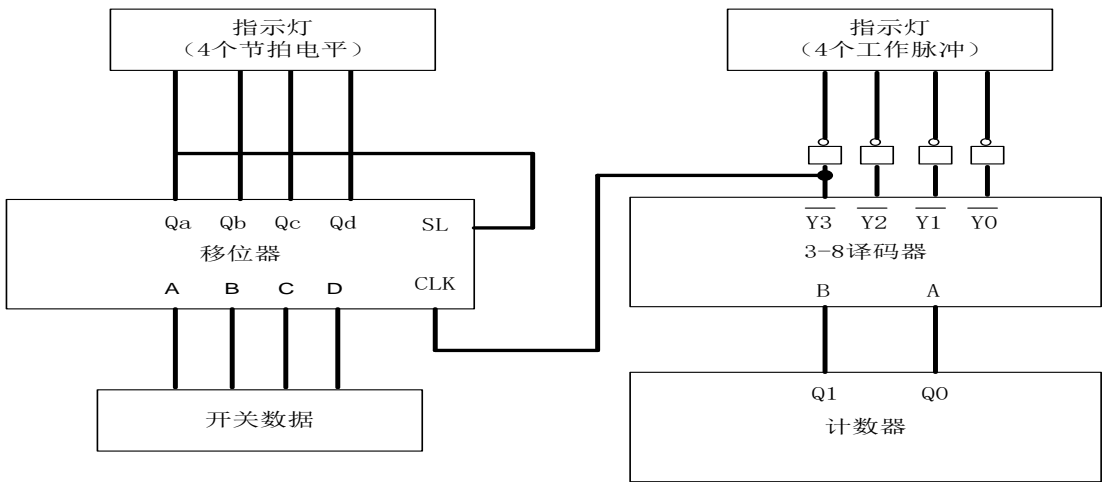


图 12-2 时序试验结构图

## 3、实验要求：

- ◆ 开关数据为移位器预置 0001。
- ◆ 选用适当方案，设计出实验线路图。
- ◆ 设计试验步骤。
- ◆ 利用指示灯观察实验现象，写出实验报告。

## 4、参考器件：

计数器可以采用 74LS161；译码器采用 74LS138；移位寄存器采用 74LS194；反相器采用 74LS04。

## 实验 9 七段译码设计

1、**实验目的：**熟悉 Quartus II 的设计流程全过程，学习计数器的设计和硬件测试。掌握原理图的设计方法。

2、**实验原理：**4 位计数器连接 7 段译码，多数码管进行显示控制。实验框图如图 6 所示。

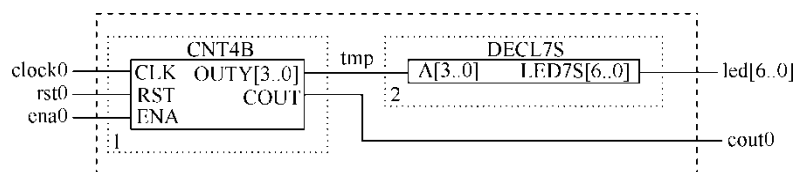


图 6 原理图示意图

其中，CNT4B 采用 74161 计数器芯片实现，DECL7S 采用 7448（共阳）设计。

### 3、实验内容：

（1）设计工程文件，使实验平台工作于模式 6，锁定引脚并硬件下载测试，输入引脚 clock0 绑定于键 8，输入引脚 rst0 绑定于键 7，清零引脚绑定于键 6，输出引脚 led[6..0] 绑定于数码 8。引脚锁定后进行编译、下载和硬件测试实验。将实验过程和实验结果写进实验报告。

（2）实验报告：将实验原理、设计过程、硬件测试结果写进实验报告。

# 实验 10 LPM\_ROM 实验

参考实验示例和实验课件：/COMT\_DEMO/CHPT4/ DEMO\_44\_ROM/ 和 实验 4\_4.ppt 。

参考《[程序存储器数据存储器参考资料](#)》

## 1 实验目的：

- (1) 掌握 FPGA 中 lpm\_rom 的设置，作为只读存储器 ROM 的工作特性和配置方法；
- (2) 用文本编辑器编辑 mif 文件配置 ROM，学习以 mif 格式文件加载于 lpm\_rom 中；
- (3) 在初始化存储器编辑窗口编辑 mif 文件配置 ROM；
- (4) 验证 FPGA 中 LPM\_ROM 的功能。

## 2 实验内容：

实验中主要掌握三方面的内容：1、LPM\_ROM 的参数设置；2、LPM\_ROM 中数据的写入，即初始化文件的编写；3、LPM\_ROM 的实际应用，在实验台上的调试方法。

## 3 实验步骤：

参考《[程序存储器数据存储器参考资料](#)》中的 LPM\_ROM 的设计过程。下载示例工程文件（图 7）至实验台上的 FPGA，选择实验台模式为 0，24 位数据输出由数码 8 至数码 3 显示，6 位地址由键 2、键 1 输入，键 1 负责低 4 位，地址锁存时钟 CLK 由键 8 控制，每一次上升沿，将地址锁入，数码管 8/7/6/5/4/3 将显示 ROM 中输出的数据。发光管 6 至 1 显示输入的 6 位地址值。

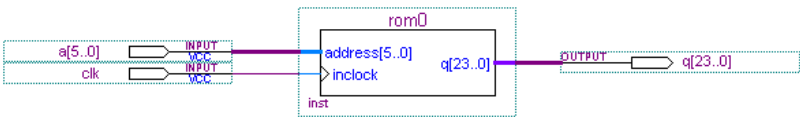


图 7-1 LPM\_ROM 的结构

Addr	+0	+1	+2	+3	+4	+5	+6	+7
00	018108	00ED82	00C050	00E004	00B005	01A206	959A01	00E00F
08	00ED8A	00ED8C	00A008	008001	062009	062009	070A08	038201
10	001001	00ED83	00ED87	00ED99	00ED9C	31821D	31821F	318221
18	318223	00E01A	00A01B	070A01	00D181	21881E	019801	298820
20	019801	118822	019801	198824	019801	018110	000002	000003
28	000004	000005	000006	000007	000008	000009	00000A	00000B
30	00000C	00000D	00000E	00000F	000010	000011	000012	000013
38	000014	000015	000016	000017	000018	000019	00001A	00001C

图 7-2 ROM 初始化文件 ROM\_A.mif 的内容

(4) 实验任务：实验前认真复习 LPM-ROM 存储器部分的有关内容；用图形编辑设计 lpm\_rom。

- 1、要求用 LPM 元件库设计 LPM\_ROM，地址总线宽度 address[]和数据总线宽度 q[]分别



为 6 位和 24 位。2、建立相应的工程文件，设置 `lpm_rom` 数据参数，`lpm_ROM` 配置文件的路径 (`ROM_A.mif`)，并设置在系统 ROM/RAM 读写允许，以便能对 FPGA 中的 ROM 在系统读写。

3、锁定输入输出引脚。4、完成全程编译。5、下载 SOF 文件至 FPGA，改变 `lpm_ROM` 的地址 `a[5..0]`，外加读脉冲，通过实验台上的数码管比较读出的数据是否与初始化数据(`rom4.mif` 中的数据)一致。6、打开 QuartusII 的在系统存储模块读写工具，了解 FPGA 中 ROM 中的数据，并对其进行在系统写操作。(3) 记录实验数据，写出实验报告。

## 实验 11 LPM\_RAM 实验

参考实验示例和实验课件：/COMT\_DEMO/CHPT4t/ DEMO\_45\_RAM/ 和 实验 4\_5.ppt 。

参考《[程序存储器数据存储器参考资料](#)》。

(1) 实验目的：1、了解 FPGA 中 RAMlpm\_ram\_dq 的功能；2、掌握 lpm\_ram\_dq 的参数设置和使用方法；3、掌握 lpm\_ram\_dq 作为随机存储器 RAM 的仿真测试方法，工作特性和读写方法。

(2) 实验内容：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 8-1。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。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。数据的读出控制是从 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 是读/写控制端，低电平时进行读操作，高电平时进行写操作；CLK 是读/写时钟脉冲信号；DATA[7..0]是 RAM\_dq0 的 8 位数据输入端；A[7..0]是 RAM 的读出和写入地址；Q[7..0]是 RAM\_dq0 的 8 位数据输出端。

(3) 实验步骤：1、按图 7-1 输入电路图，进行编译、引脚锁定、向 FPGA 配置下载；2、通过键 1、键 2 输入 RAM 的 8 位数据（选择实验台工作模式 1），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、lpm\_ram\_dq 也能加入初始化文件（这里是 5\_ram.mif 是后面将要用到的模型 CPU 执行微程序文件），注意此文件加入的路径表达和文件表达方式（图 8-2）：5\_ram.mif（注意后缀 mif 要小写）；选择 RAM 的 ID 名取为：ram1。

实验中选择实验电路模式为 NO.1，按以上方式进行验证实验。首先控制读出初始化数据，与载入的初始化文件 ram\_dp1.mif 中的数据进行比较，然后控制写入一些数据，再读出比较。使用在系统读写 RAM 的工具对其中的数据进行读写操作，设置成连续读模式，将在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进行对比（图 8-3）。

(4) 实验要求：1、设计数据宽度和地址宽度均为 8 位 lpm\_ram\_dq；2、设计对 lpm\_ram\_dq 进行测试的波形文件，完成对 lpm\_ram\_dq 时序仿真和硬件测试；3、利用系统读写 RAM 的工具对其中的数据进行读、写、修改、加载新的数据文件操作；4、写出实验报告，包括工作原理、调试和测试结果。

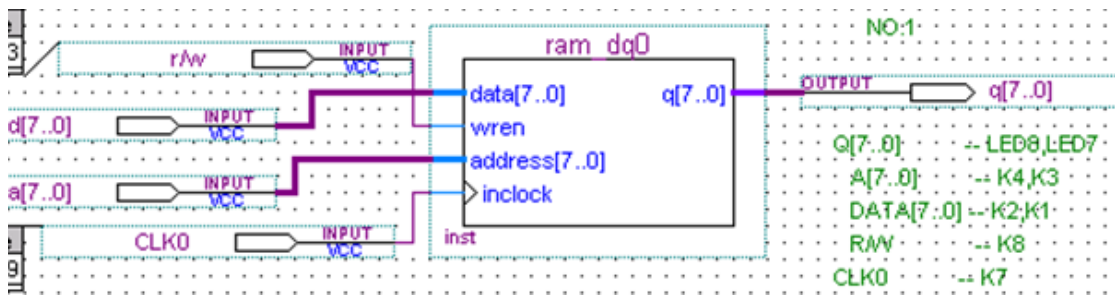


图 8-1 LPM\_RAM 的结构

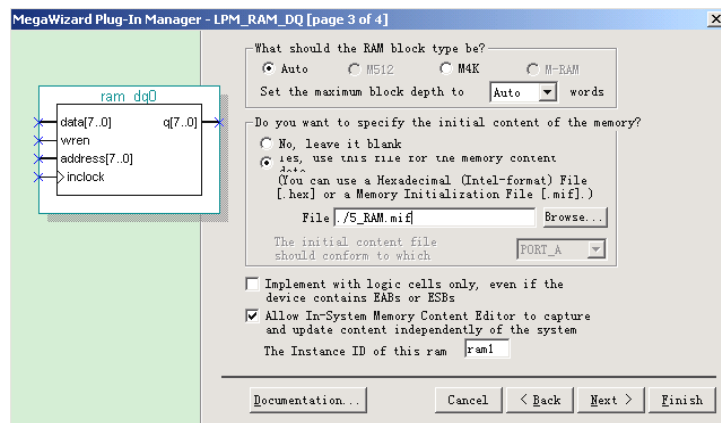


图 8-2 lpm\_ram\_dq 加入初始化文件和选择在系统读写 RAM 功能

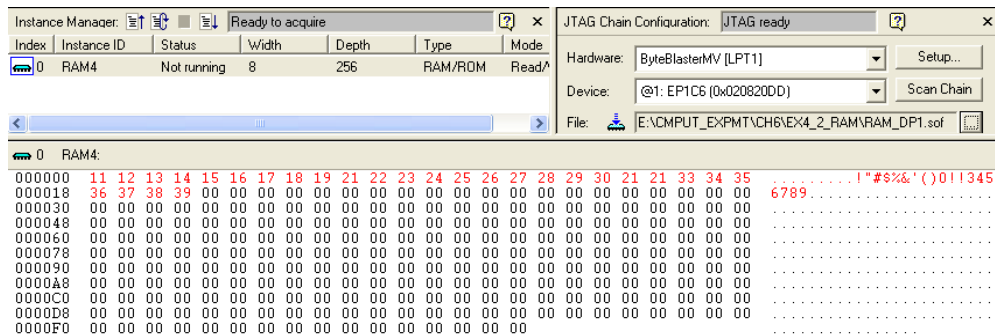


图 8-3 使用在系统读写 RAM 的工具对 lpm\_ram 中的数据进行读写操作

## 实验 12 RAM 扩展实验

### 一、实验目的：

了解半导体静态随机读写存储器 RAM 的工作原理及其使用方法。

掌握半导体存储器的字、位扩展技术。

### 二、实验方案：

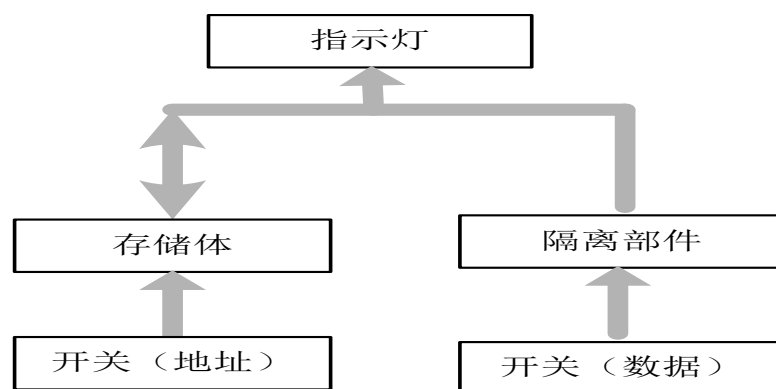


图 9 RAM 实验结构图

### 三、实验要求：

◆采用 1K x 4 的 LPM\_RAM 的结构（参考实验 8 生成器件），构成 1K x 8 的存储器。

◆选择五个不连续的存贮单元地址，分别存入不同内容，作单个存贮器单元的读/写操作实验。

◆采用 1K x 4 的 LPM\_RAM 的结构，构成 2K x 4 的存储器。

◆必须使用译码器进行扩展（三输入都用，接开关）。

◆选择五个不连续的存贮单元地址，分别存入不同内容，作单个存贮器单元的读/写操作实验。

◆选用适当芯片，根据各种控制信号的极性和时序要求，设计出实验线路图。

◆分别设计试验步骤。

◆使用开关进行数据加载，通过指示灯显示实验结果，记录试验现象，写出实验报告。给出字扩展试验中每片 RAM 芯片的地址范围。

### 四、参考器件：

隔离部件采用三态门 TRI。

译码器采用 74LS138

备注：为简化试验，地址可只用低 4 位（其余地址可接地）。

## 实验 13 控制器实验

### 1、实验内容及说明

目前控制器设计大都采用微程序设计方法，又称存储逻辑控制器。微程序控制器电路结构如图 13-1 所示。它由控制存储器 CROM、微程序  $\mu PC$  计数器和微指令寄存器  $\mu IR$  构成。

其中，微程序计数  $\mu PC$  向控制存储器提供 8 位微地址，在控存读信号  $\overline{\mu RD}$  的作用下，读出一条长 24 位的微指令代码，并在打入命令  $CP\mu IR$  的作用下，送入  $\mu IR$ 。

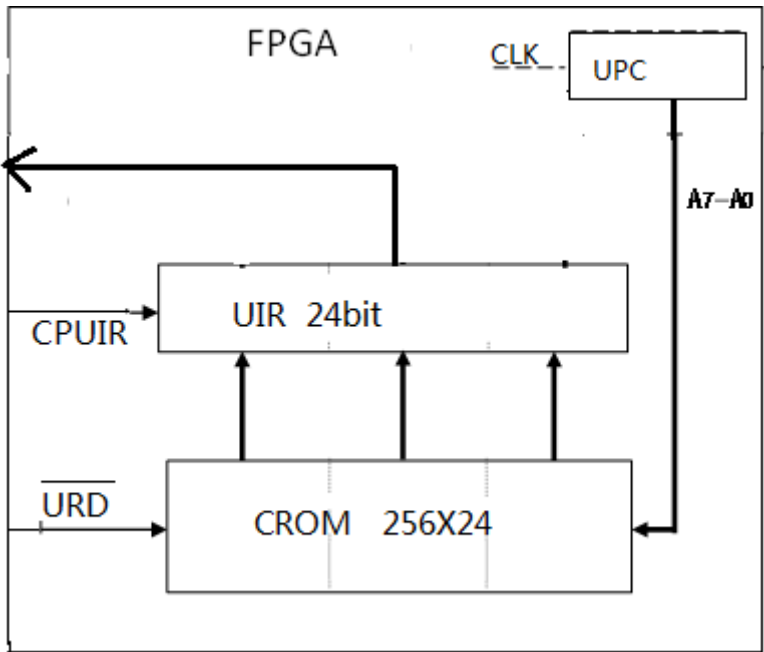


图 13-1 微程序控制器框图

每当按一次脉冲键便产生一个负脉冲，该脉冲的作用是：

- 作为读控存的命令  $\overline{\mu RD}$ 。
- 负脉冲当作  $CP\mu IR$  将读出的微指令打入微指令寄存器  $\mu IR$ 。
- 负脉冲的上升沿使  $\mu PC+1$  形成下一条微指令的地址。
- 负脉冲反相后的上升沿作为寄存器打入脉冲。

微程序时序如图 13-2。

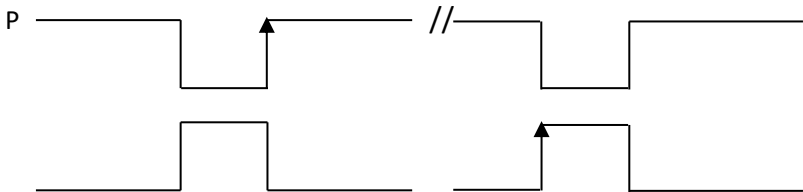


图 13-2 微程序时序

## 2、实验步骤

(1) 微程序计数器  $\mu$  PC 的设计，完成 8 位具有加 1 功能和清除功能的计数器设计并封装，如图 13-3。

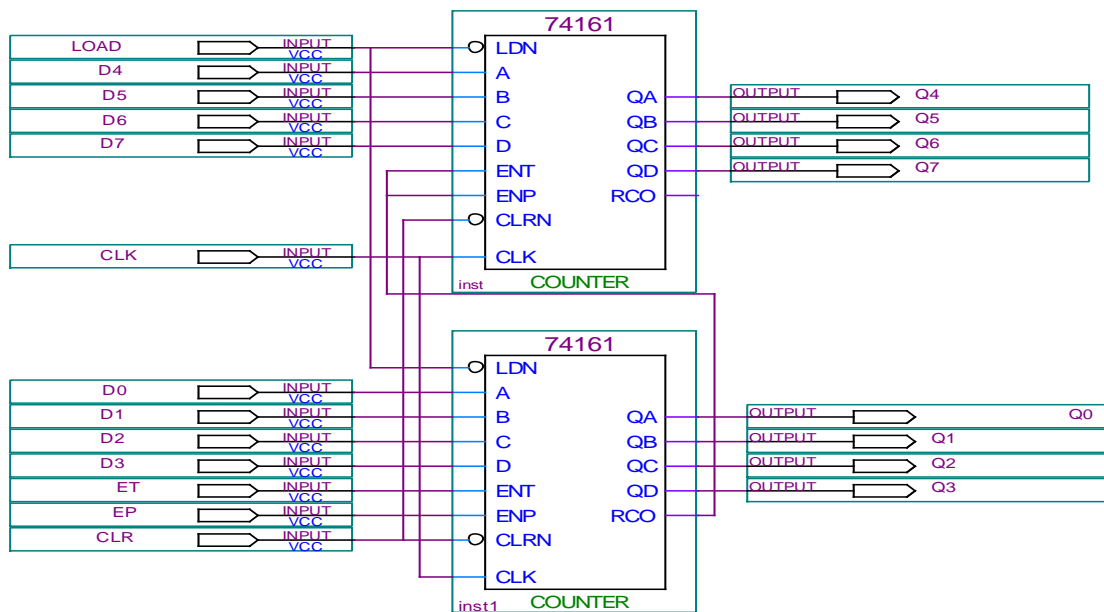


图 13-3 UPC 的设计图

其中，CLR：清零端，，低电平有效；CLR=0 时，Q7Q6Q5Q4Q3Q2Q1Q0=00000000；

LOAD：置数端，低电平有效；LOAD=0 时，在 CLK 的上升沿，

Q7Q6Q5Q4Q3Q2Q1Q0=D7D6D5D4D3D2D1D0；

当 CLR=1，LOAD=1，ET=1，EP=1 时，对 CLK 进行增 1 计数。

注意：本实验使用时，只有 clk、CLR 两引脚引出，其它引脚，ET、EP、LOAD 接高电平。

(2) 按图 13-1 完成微程序控制器的连线及引脚锁定。

- \* 用单脉冲驱动  $\mu$  PC 的计数脉冲 CPPC、 $\mu$  RD、CP  $\mu$  IR。
- \* 将  $\mu$  PC 的 8 位输出锁定在 A7—A0 上。
- \* 实验平台工作于模式 5，将单脉冲锁定于键 8，将  $\mu$  PC 的复位端 CLR 锁定在键 7，输出 UIR 的 24 位接数显 3—数显 8。

(3) 设计并初始化控制存储器。

从 ROM 的 0 号单元开始写入、写入的微程序代码随意确定。

(4) 读出微指令

◆按 CPU 复位按键，清除微指令计数器。

◆按单脉冲键

读出 0 号单元中的微指令代码送  $\mu\text{IR}23$ - $\mu\text{IR}0$  同时显示在数显上；

将微指令计数器  $\mu\text{PC} + 1$  为读下条微指令做准备。

◆连续按单脉冲键

微指令连续从 ROM 中读出并显示。

注意：在读出过程中要和原先写入的代码比对看是否正确

# 实验 14 综合实验

## 1、实验内容及说明

CPU 综合实验电路包括运算器电路和控制器电路。图 14-1 给出了 CPU 综合实验结构框图。

运算器由三个寄存器 R0、R1、R2、移位器、加法器等构成，并组装在一起构成 ALU 算术逻辑运算部件，参照四位补码运算器电路框图所示。

## 2、实验步骤

- (1) 调用 ALU 模块、 $\mu$ PC 模块及门电路按 CPU 综合实验结构框图完成连线。
- (2) 管脚定义：实验平台工作于模式 5，ALU 的输入数据 a3-a0 依次锁定在 $\mu$ IR23- $\mu$ IR420 上，CPR0、CPR1、CPR2 依次锁定在 $\mu$ IR7- $\mu$ IR5 上，LM、DM、RM、C0 依次锁定在 $\mu$ IR4- $\mu$ IR1 上，P 锁定在键 8 上。Q3-Q0 依次锁定在 D4-D1 上。

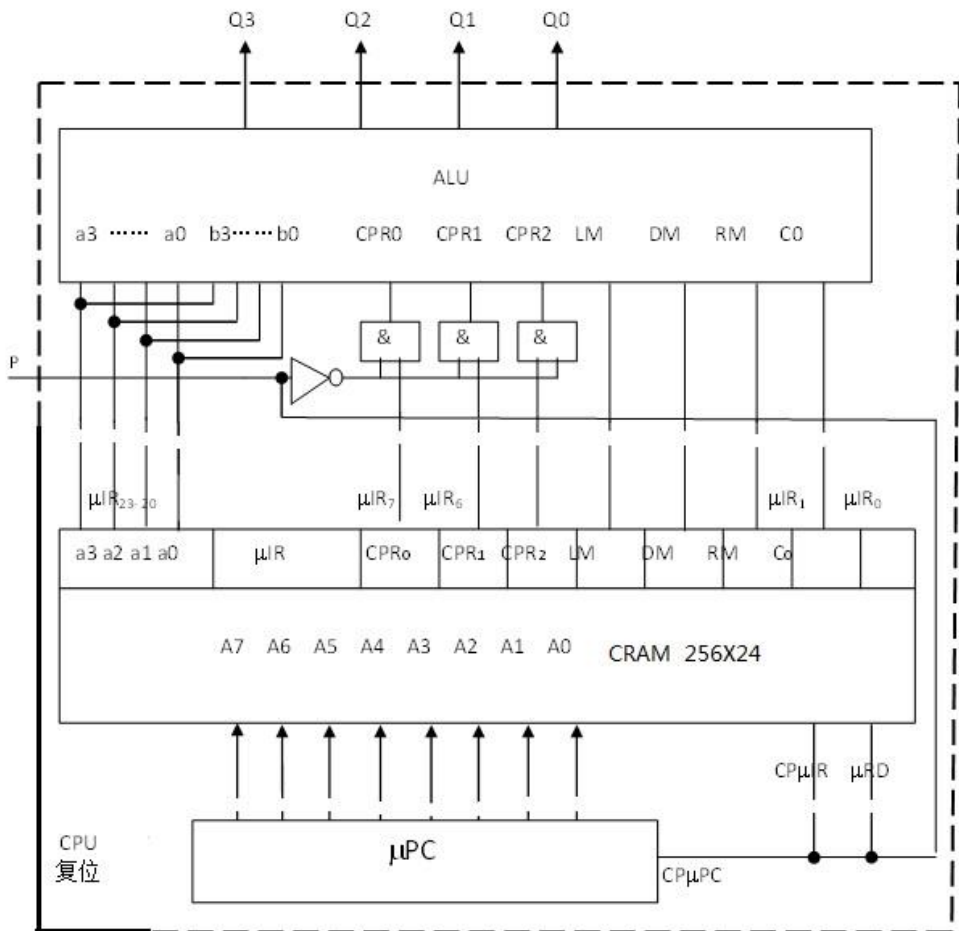


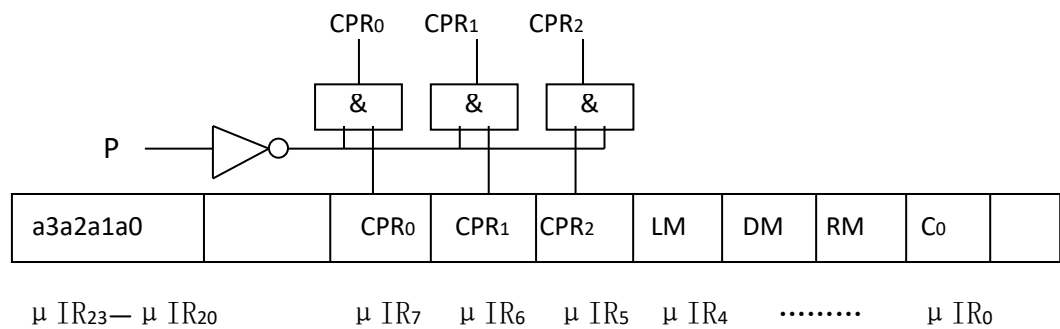
图 14-1 CPU 综合实验结构框图

- (3) 适配、下载



(4) 编制微程序

微指令可确定如下格式：



将微指令格式分为两部分：前面部分 $\mu IR_{23} \sim \mu IR_{20}$  可设置数据，后面部分 $\mu IR_7 \sim \mu IR_0$

可确定微命令，例：需要 CPR0 脉冲，该位为 1，否则为 0；备用位填 0。

例题：编写一个  $0110 + 1000$  的微程序。

寄存器分配：0110 送 R0、1000 送 R1、结果送 R2。

操作步骤	微指令	说明
0 1 1 0→R0;	60 00 80H	存入控制存储器 ROM 的 0 单元。
↓		
1 0 0 0→R1;	80 00 40H	存入控制存储器 ROM 的 1 单元。
↓		
R0+R1→R2;	00 00 08H	存入控制存储器 ROM 的 2 单元。

(5) 功能检查

按 CPU 复位键清  $\mu PC$ ，使之指向控制存储器的 0 号单元。

每按一次单脉冲键，便执行一条微指令。

按第 3 次单脉冲键，微程序执行完毕。

运算结果应存放在 R2 中，并用 LD3—0 指示。

同学可编制多种多样的微程序来完成各种操作。

例  $2 \times (R0 + R1) \Rightarrow R2;$

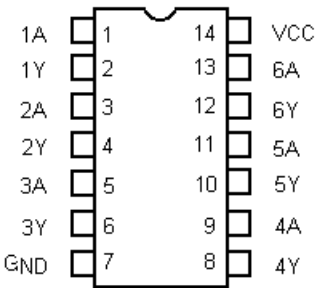
$1/2 (R0 + R1) \Rightarrow R2;$

.....

# 附录 常用实验器件引脚图

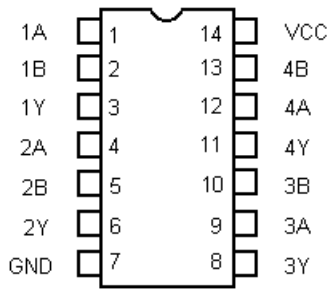
## 1、六反相器 74LS04

$Y = \overline{A}$



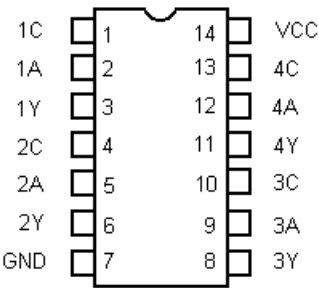
## 2、四 2 输入异或门 74LS86

$Y=A \oplus B = A\overline{B} + \overline{A}B$

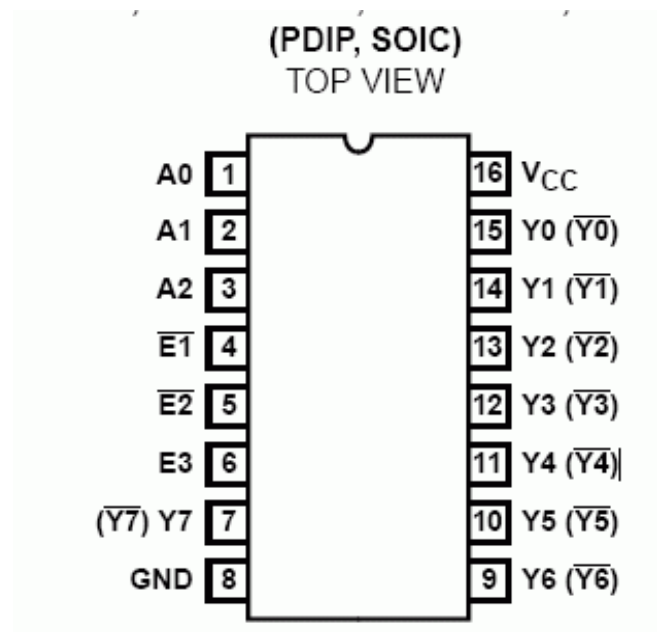


## 3、三态输出的四总线缓冲门 74LS125

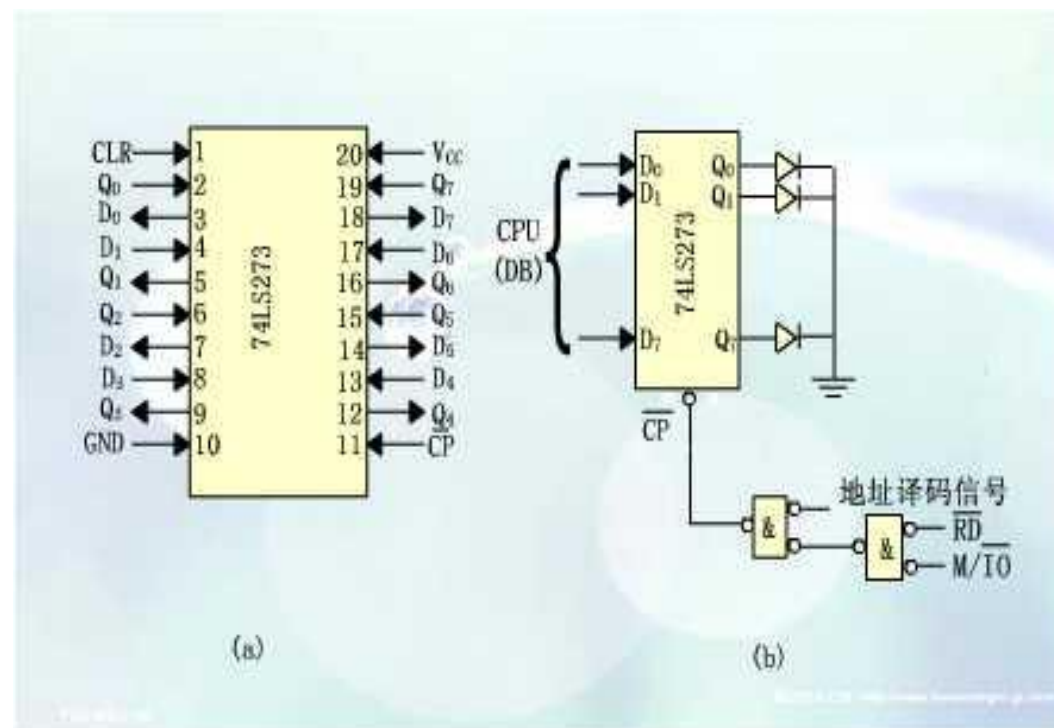
正逻辑  
 $Y = A$   
C为高时输出截止



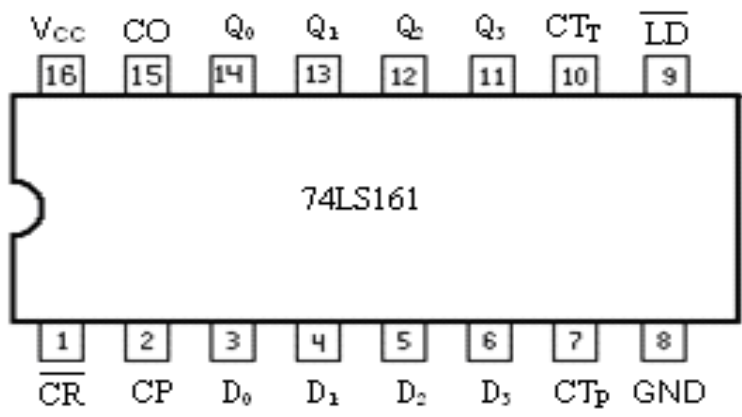
#### 4、3-8译码器 74LS138



#### 5、8D触发器74LS273



6、74LS161（四位同步二进制加法计数器）

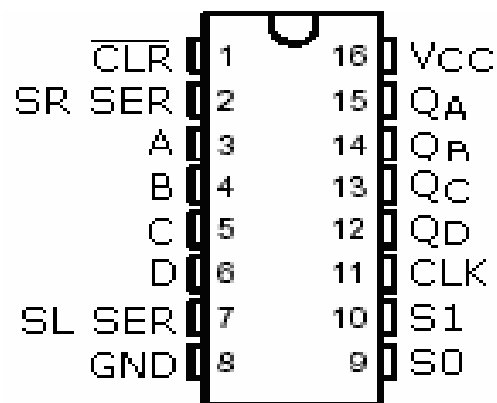


CO 为进位输出端。

功能表

输 入									输 出					说 明
$\overline{CR}$	$\overline{LD}$	CT <sub>P</sub>	CT <sub>T</sub>	CP	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>	CO	
0	x	x	x	x	x	x	x	x	0	0	0	0	0	异步清零
1	0	x	x	↑	d <sub>3</sub>	d <sub>2</sub>	d <sub>1</sub>	d <sub>0</sub>	d <sub>3</sub>	d <sub>2</sub>	d <sub>1</sub>	d <sub>0</sub>		CO=CT <sub>T</sub> Q <sub>3</sub> Q <sub>2</sub> Q <sub>1</sub> Q <sub>0</sub>
1	1	1	1	↑	x	x	x	x	计数 保持 保持					CO=Q <sub>3</sub> Q <sub>2</sub> Q <sub>1</sub> Q <sub>0</sub>
1	1	0	x	x	x	x	x	x						CO=CT <sub>T</sub> Q <sub>3</sub> Q <sub>2</sub> Q <sub>1</sub> Q <sub>0</sub>
1	1	x	0	x	x	x	x	x					0	

7、74LS194（双向移位寄存器）



功能表：

输				入						输 出			
清零	模式		时钟	串行 SER		并 行				Q <sub>A</sub>	Q <sub>B</sub>	Q <sub>C</sub>	Q <sub>D</sub>
$\overline{\text{CLR}}$	S <sub>1</sub>	S <sub>0</sub>		CLK	左 SL	右 SR	A	B	C				
L	×	×	×	×	×	×	×	×	×	L	L	L	L
H	×	×	L	×	×	×	×	×	×	Q <sub>AO</sub>	Q <sub>BO</sub>	Q <sub>CO</sub>	Q <sub>DO</sub>
H	H	H	↑	×	×	a	b	c	d	a	b	c	d
H	L	H	↑	×	H	×	×	×	×	H	Q <sub>An</sub>	Q <sub>Bn</sub>	Q <sub>Cn</sub>
H	L	H	↑	×	L	×	×	×	×	L	Q <sub>An</sub>	Q <sub>Bn</sub>	Q <sub>Cn</sub>
H	H	L	↑	H	×	×	×	×	×	Q <sub>Bn</sub>	Q <sub>Cn</sub>	Q <sub>Dn</sub>	H
H	H	L	↑	L	×	×	×	×	×	Q <sub>Bn</sub>	Q <sub>Cn</sub>	Q <sub>Dn</sub>	L
H	L	L	×	×	×	×	×	×	×	Q <sub>AO</sub>	Q <sub>BO</sub>	Q <sub>CO</sub>	Q <sub>DO</sub>

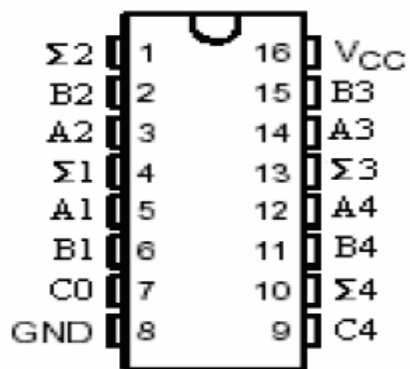
a、b、c、d=分别为A、B、C 或D 输入端上稳定状态输入的电平。

QAO、QBO、QCO、QDO=在已建立稳定状态输入条件之前QA、QB、QC、QD 相应的电平。

QAn、QBn、QCn、QDn=在时钟最新 ↑ 跃变之前的QA、QB、QC、QD 的电平。

H=高电平 L=低电平 ×=不定 ↑=从低电平转换到高电平

### 8、74LS 283（快速进位四位二进制全加器）



## 功能表

输 入				输 出					
				C <sub>0</sub> =L			C <sub>0</sub> =H		
				C <sub>2</sub> =L			C <sub>2</sub> =H		
A <sub>1</sub> A <sub>3</sub>	B <sub>1</sub> B <sub>3</sub>	A <sub>2</sub> A <sub>4</sub>	B <sub>2</sub> B <sub>4</sub>	Σ <sub>1</sub> Σ <sub>3</sub>	Σ <sub>2</sub> Σ <sub>4</sub>	C <sub>2</sub> C <sub>4</sub>	Σ <sub>1</sub> Σ <sub>3</sub>	Σ <sub>2</sub> Σ <sub>4</sub>	C <sub>2</sub> C <sub>4</sub>
L	L	L	L	L	L	L	H	L	L
H	L	L	L	H	L	L	L	H	L
L	H	L	L	H	L	L	L	H	L
H	H	L	L	L	H	L	H	H	L
L	L	H	L	L	H	L	H	H	L
H	L	H	L	H	H	L	L	L	H
L	H	H	L	H	H	L	L	L	H
H	H	H	L	L	L	H	H	L	H
L	L	L	H	L	H	L	H	H	L
H	L	L	H	H	H	L	L	L	H
L	H	L	H	H	H	L	L	L	H
H	H	L	H	L	L	H	H	L	H
L	L	H	H	L	L	H	H	L	H
H	L	H	H	H	L	H	L	H	H
L	H	H	H	H	L	H	L	H	H
H	H	H	H	L	H	H	H	H	H

H=高电平

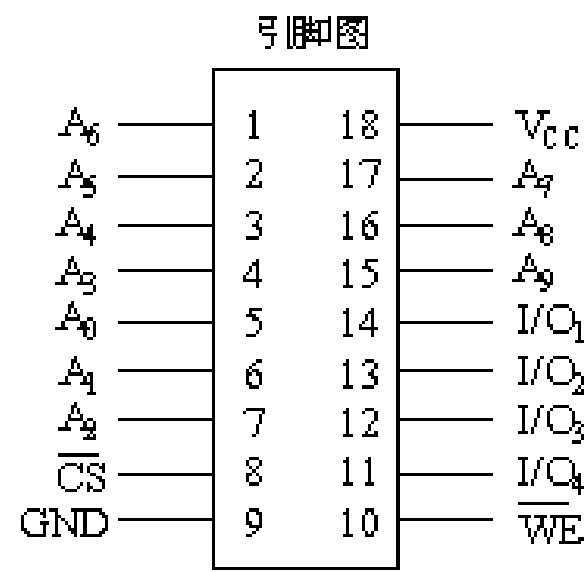
L=低电平

说明:

$A_1$ 、 $B_1$ 、 $A_2$ 、 $B_2$ 和 $C_0$ 是用于确定输出 $\Sigma_1$ 、 $\Sigma_2$ 和内部进位 $C_2$ 值的。

$A_3$ 、 $B_3$ 、 $A_4$ 、 $B_4$ 和 $C_2$ 是用于确定输出 $\Sigma_3$ 、 $\Sigma_4$ 和 $C_4$ 值的。

9、INTEL2114 静态存储器



Intel2114RAM 存储器芯片为双列直插式集成电路芯片，共有 18 个引脚，引脚图如图 4-3 所示，各引脚的功能如下：

$A_0 - A_9$ ： 10 根地址信号输入引脚。

$\overline{WE}$ ： 读 / 写控制信号输入引脚，当  $\overline{WE}$  为低电平时，使输入三态门导通，信息由数据总线通过输入数据控制电路写入被选中的存储单元；反之从所选中的存储单元读出信息送到数据总线。

I/O 1 ~ I/O 4： 4 根数据输入 / 输出信号引脚。

$\overline{CS}$ ： 片选信号，低电平有效，通常接地址译码器的输出端。