

# 第4章

## 计算机功能模块的原理与设计

# 4.1 CPU基本功能与结构

## 1. CPU的功能



# 4.1 CPU基本功能与结构

## 2. CPU结构

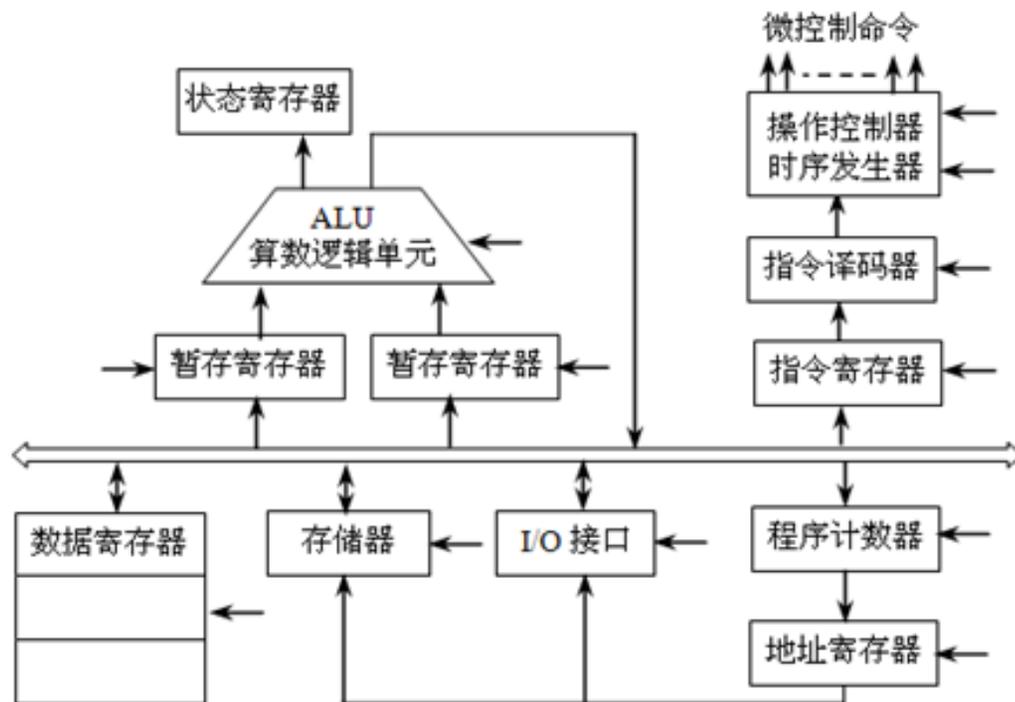


图 4-1 CPU 组成部件的功能模块图

# 4.2 计算机中的基本部件

## 4.2.1. 算术逻辑单元

表 4-1 ALU 的运算功能

选择端 S3 S2 S1 S0	M=H 逻辑操作	M=L 算术操作	
	逻辑功能	Cn=L (无进位)	Cn=H (有进位)
0 0 0 0	$F = \overline{A}$	$F = A$	$F = A$ 加1
0 0 0 1	$F = \overline{A + B}$	$F = A + B$	$F = (A + B)$ 加1
0 0 1 0	$F = \overline{AB}$	$F = A + \overline{B}$	$F = A + \overline{B} + 1$
0 0 1 1	$F = 0$	$F = \text{减} 1$ (2的补码)	$F = 0$
0 1 0 0	$F = \overline{AB}$	$F = A$ 加 $\overline{A}\overline{B}$	$F = A$ 加 $\overline{A}\overline{B}$ 加1
0 1 0 1	$F = \overline{B}$	$F = (A + B)$ 加 $\overline{A}\overline{B}$	$F = (A + B)$ 加 $\overline{A}\overline{B} + 1$
0 1 1 0	$F = A \oplus B$	$F = A$ 减 $B$	$F = A$ 减 $B$ 减1
0 1 1 1	$F = \overline{AB}$	$F = A + \overline{B}$	$F = (A + \overline{B})$ 减1
1 0 0 0	$F = \overline{A + B}$	$F = A$ 加 $AB$	$F = A$ 加 $AB$ 加1
1 0 0 1	$F = \overline{A \oplus B}$	$F = A$ 加 $B$	$F = A$ 加 $B$ 加1
1 0 1 0	$F = B$	$F = (A + \overline{B})$ 加 $AB$	$F = (A + \overline{B})$ 加 $AB$ 加1
1 0 1 1	$F = AB$	$F = AB$	$F = AB$ 减1
1 1 0 0	$F = 1$	$F = A$ 加 $A^*$	$F = A$ 加 $A$ 加1
1 1 0 1	$F = A + \overline{B}$	$F = (A + B)$ 加 $A$	$F = (A + B)$ 加 $A$ 加1
1 1 1 0	$F = A + B$	$F = (A + \overline{B})$ 加 $A$	$F = (A + \overline{B})$ 加 $A$ 加1
1 1 1 1	$F = A$	$F = A$	$F = A$ 减1

注1、\* 表示每一位都移至下一更高有效位，“+”是逻辑或，“加”是算术加。  
 注2、在借位减法表达上，表4-1与TTL器件74LS181的真值表略有不同。

**【例4-1】**

```
module ALU181A (S, A, B, F, M, CN, CO, FZ);
  input[3:0] S; input[7:0] A,B; input M, CN;
  output[7:0] F; output CO, FZ;
  wire[7:0] F; wire CO;wire[8:0] A9, B9;
  reg FZ; reg[8:0] F9;
  assign A9={1'b0,A} ; assign B9={1'b0,B};
  always @(M or CN or A9 or B9 or S) begin
    case (S)
      4'b0000 : if (M==0) F9<=A9+CN ;           else F9<=~A9;
      4'b0001 : if (M==0) F9<=(A9|B9) + CN ;    else F9<=~(A9|B9);
      4'b0010 : if (M==0) F9<=(A9|(~B9))+ CN;   else F9<=(~A9)&B9;
      4'b0011 : if (M==0) F9<=9'b000000000-CN; else F9<=9'b000000000;
      4'b0100 : if (M==0) F9<=A9+(A9 & ~B9)+CN; else F9<=~(A9 & B9) ;
      4'b0101 : if (M==0) F9<=(A9|B9)+(A9& ~B9)+CN; else F9<= ~B9 ;
      4'b0110 : if (M==0) F9<= A9-B9-CN ;       else F9<=A9^B9 ;
      4'b0111 : if (M==0) F9<=(A9 & (~B9))-CN;  else F9<=A9&(~B9) ;
      4'b1000 : if (M==0) F9<=A9+(A9 & B9)+CN;  else F9<=(~A9)|B9 ;
      4'b1001 : if (M==0) F9<=A9+B9+CN;         else F9<=~(A9^B9);
      4'b1010 : if (M==0) F9<=(A9|(~B9))+A9&B9+CN; else F9<=B9 ;
      4'b1011 : if (M==0) F9<=(A9 & B9)-CN ;    else F9<=A9&B9 ;
      4'b1100 : if (M==0) F9<=A9+A9+CN;        else F9<=9'b000000001 ;
      4'b1101 : if (M==0) F9<=(A9|B9)+A9+CN;    else F9<=A9|(~B9) ;
      4'b1110 : if (M==0) F9<=(A9|(~B9))+A9+CN; else F9<=A9|B9 ;
      4'b1111 : if (M==0) F9<=A9-CN ;          else F9<=A9 ;
      default : F9<=9'b000000000;
    endcase
    if (A9==B9) FZ<=1'b0; else FZ<=1'b1 ;
  end
  assign F=F9[7:0] ;
  assign CO=F9[8] ;
endmodule
```

# 4.2 计算机中的基本部件

## 4.2.1. 算术逻辑单元

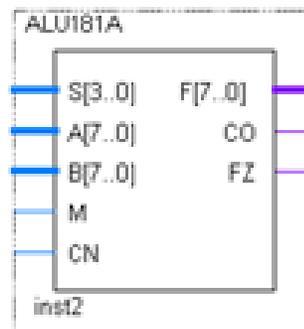


图 4-2 ALU 逻辑图

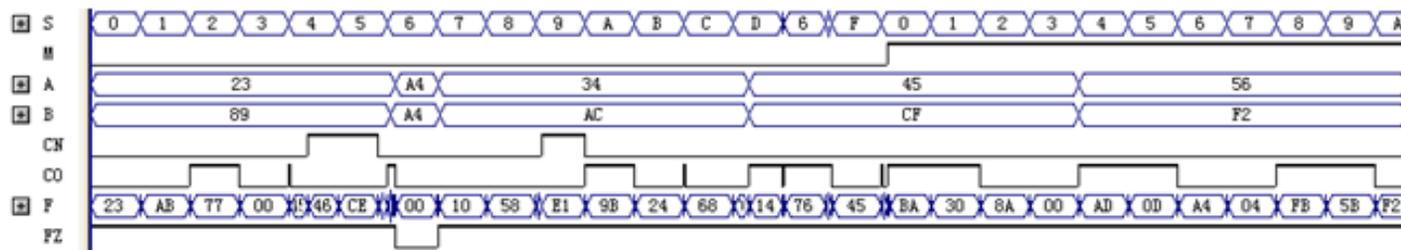


图 4-3 例 4-1 仿真波形

# 4.2 计算机中的基本部件

## 4.2.2 数据缓冲寄存器

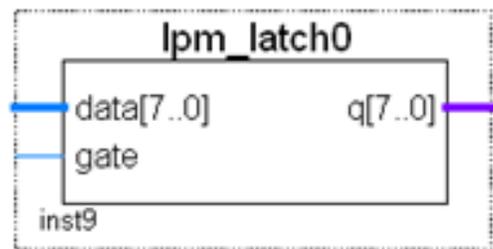


图 4-4 数据缓冲寄存器

# 4.2 计算机中的基本部件

## 4.2.3 移位运算器

表 4-2 移位运算器的功能

M	S1	S0	功 能
0	0	任意	保持
0	1	0	带进位循环左移
0	1	1	循环左移
1	0	0	循环右移
1	0	1	带进位循环右移
1	1	1	加载待移位数据
1	1	0	加载待移位数据

## 【例 4-2】

```
module SFT8 (CLK, M, C0, S, D, QB, CN);
    input CLK, M, C0; input[1:0] S; input[7:0] D;
    output[7:0] QB; output CN;
    wire[7:0] QB; wire CN; wire[2:0] ABC;
    reg[7:0] REG; reg CY;
    always @(posedge CLK)
        case (ABC)
            3'b011 : begin REG[0]<=C0; REG[7:1]<=REG[6:0];
                        CY<=REG[7]; end
            3'b010 : begin REG[0]<=REG[7]; REG[7:1]<=REG[6:0]; end
            3'b100 : begin REG[7]<=REG[0]; REG[6:0]<=REG[7:1]; end
            3'b101 : begin REG[7]<=C0; REG[6:0]<=REG[7:1]; CY<=REG[0]; end
            3'b110 : begin REG[7:0]<=D[7:0]; end
            3'b111 : begin REG[7:0]<=D[7:0]; end
            default : begin REG <= REG ; CY<=CY;end
        endcase
    assign ABC={M,S};    assign CN=CY;    assign QB[7:0]=REG[7:0];
endmodule
```

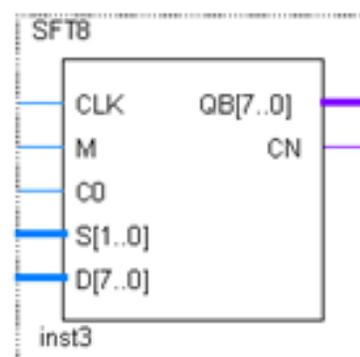


图 4-5 移位运算器

# 4.2 计算机中的基本部件

## 4.2.3 移位运算器

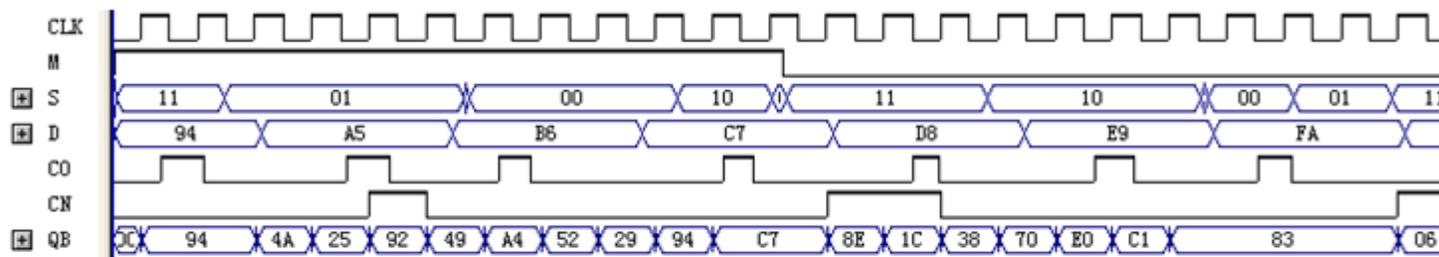


图 4-6 移位运算器 SFT8 的仿真波形

## 4.2.4 程序存储器与数据存储器

# 4.2 计算机中的基本部件

---

## 4.2.5 程序计数器与地址寄存器

### 1. 程序计数器

### 2. 地址寄存器

# 4.2 计算机中的基本部件

## 4.2.5 程序计数器与地址寄存器

### 3. 地址信号产生电路

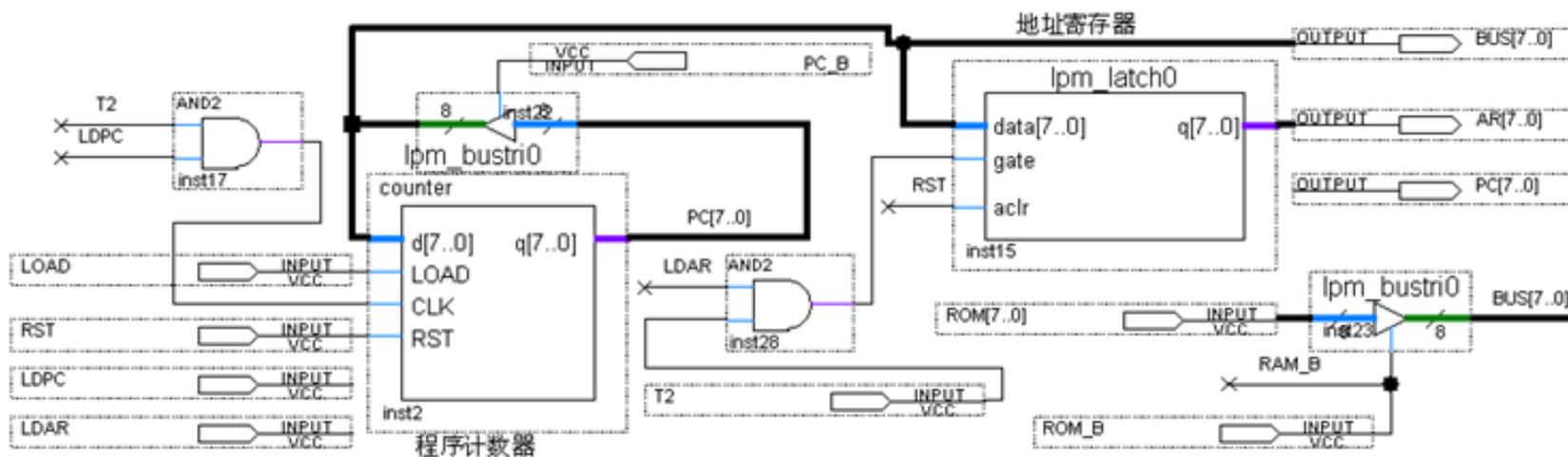


图 4-7 地址信号产生电路

# 4.2 计算机中的基本部件

## 4.2.5 程序计数器与地址寄存器

### 3. 地址信号产生电路

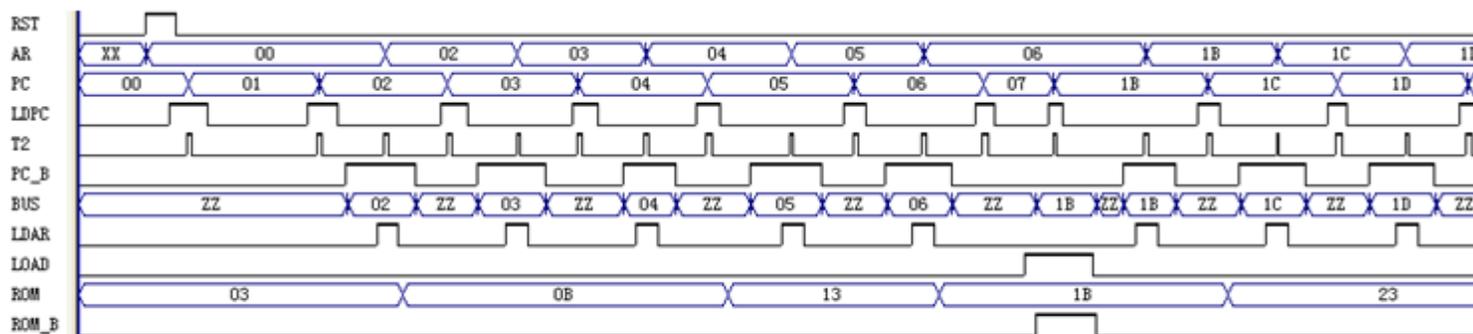


图 4-8 地址信号产生电路图 4-7 的仿真波形

## 4.2.6 指令寄存器

# 4.2 计算机中的基本部件

## 4.2.7 微程序控制器

### 1. 微程序基本名词术语

微命令

微操作

微程序

下址字段

相容性微操作和相斥性微操作

# 4.2 计算机中的基本部件

## 4.2.7 微程序控制器

### 2. 微程序控制的基本原理

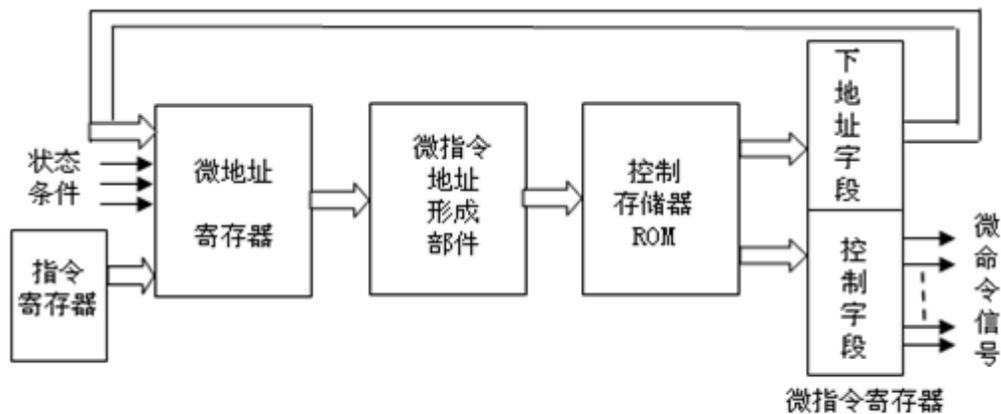


图 4-9 微程序控制的基本原理图

# 4.2 计算机中的基本部件

---

## 4.2.7 微程序控制器

### 3. 微程序执行过程

### 4. 指令译码器

# 4.2 计算机中的基本部件

## 4.2.8 微程序控制器电路结构

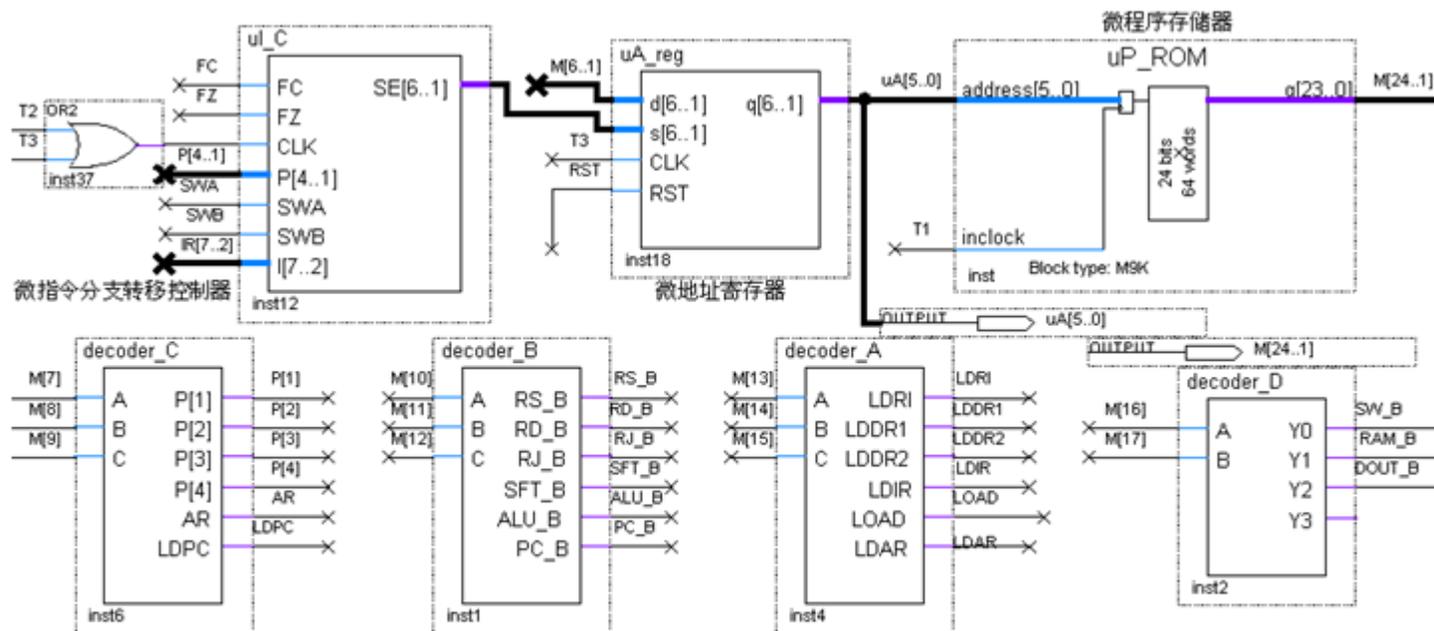


图 4-10 8 位模型计算机的微程序控制器 uPC 内部的电路结构

# 4.2 计算机中的基本部件

## 4.2.8 微程序控制器电路结构

### 1. 微指令译码电路

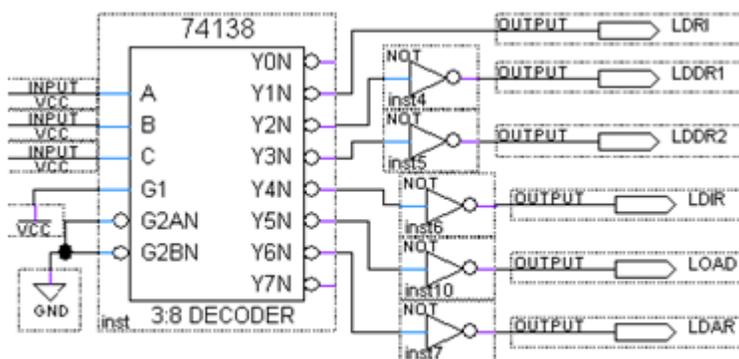


图 4-11 decoder\_A 译码器电路

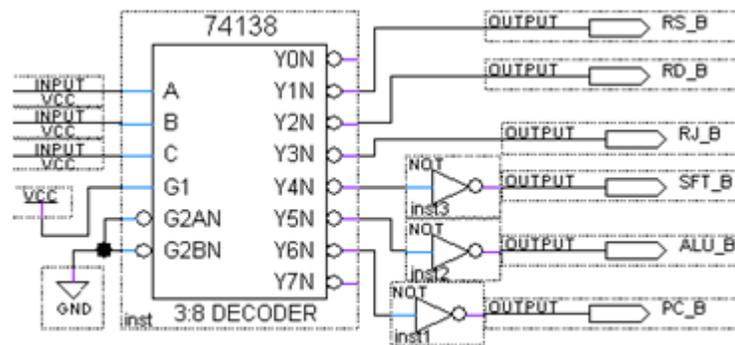


图 4-12 decoder\_B 译码器电路

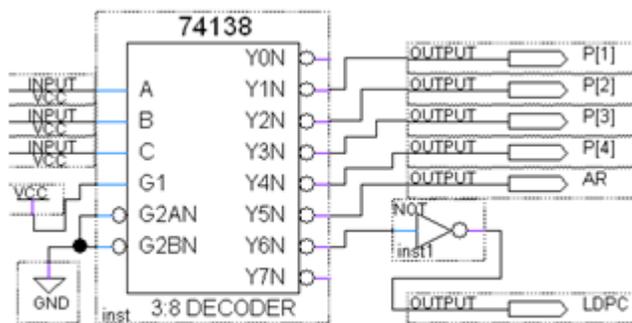


图 4-13 decoder\_C 译码器电路

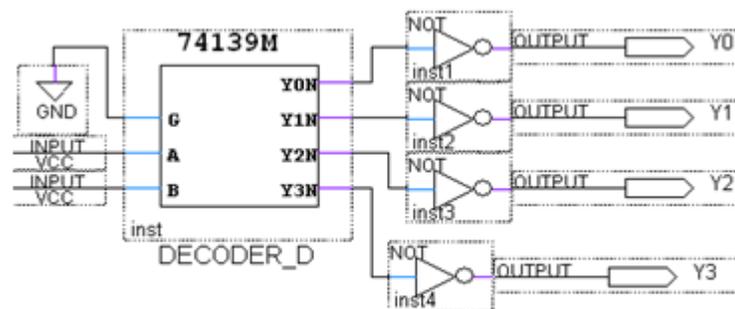


图 4-14 decoder\_D 译码器电路

# 4.2 计算机中的基本部件

## 4.2.8 微程序控制器电路结构

### 2. 微地址分支转移控制电路

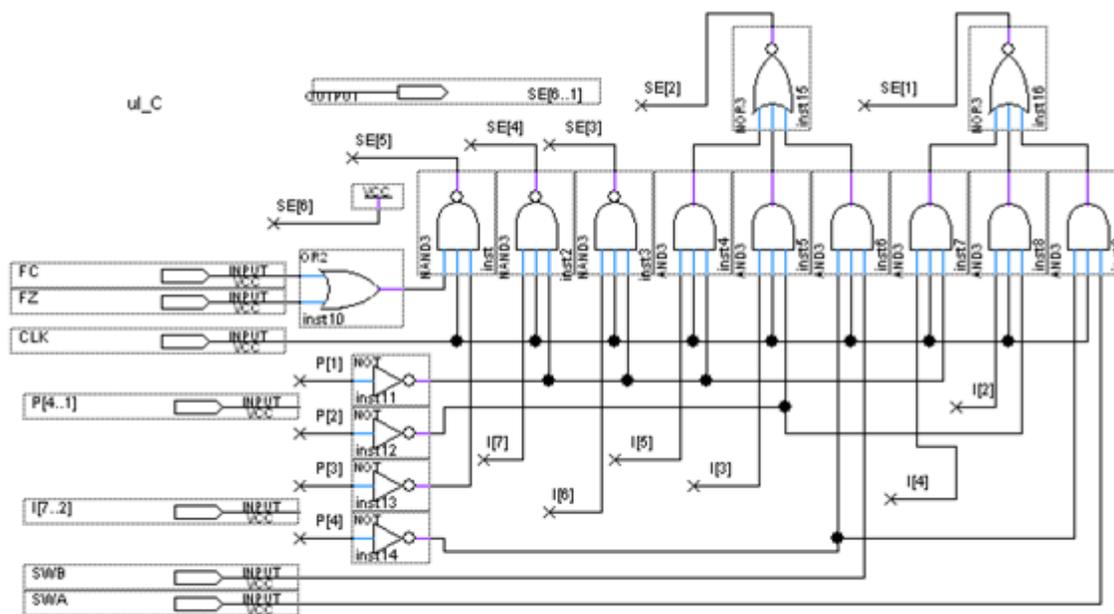


图 4-15 分支转移控制电路原理图

# 4.2 计算机中的基本部件

## 4.2.8 微程序控制器电路结构

### 3. 微地址寄存器

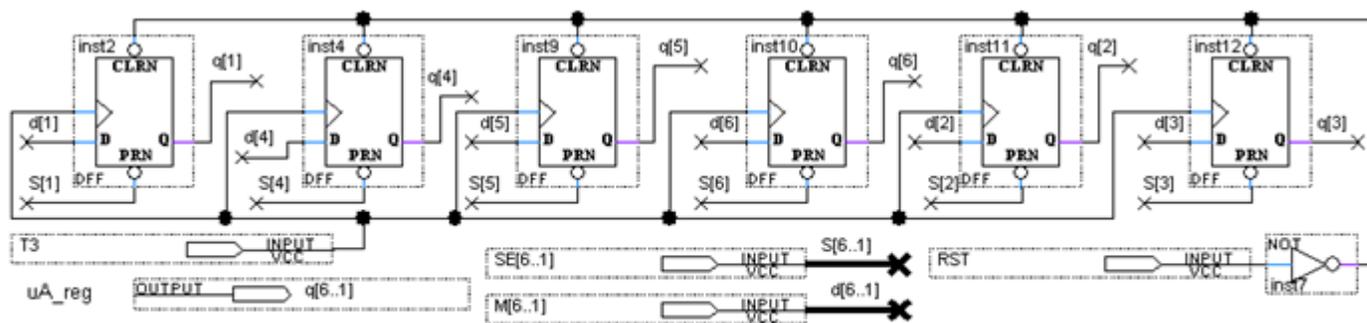


图 4-16 微地址寄存器电路原理图

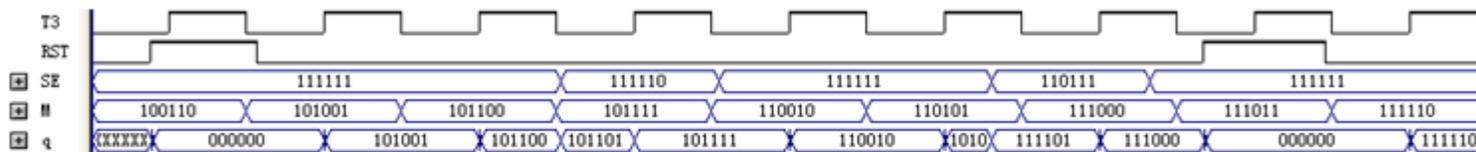


图 4-17 微地址寄存器仿真波形图

# 4.2 计算机中的基本部件

## 4.2.8 微程序控制器电路结构

### 4. 数据寄存器存取控制逻辑

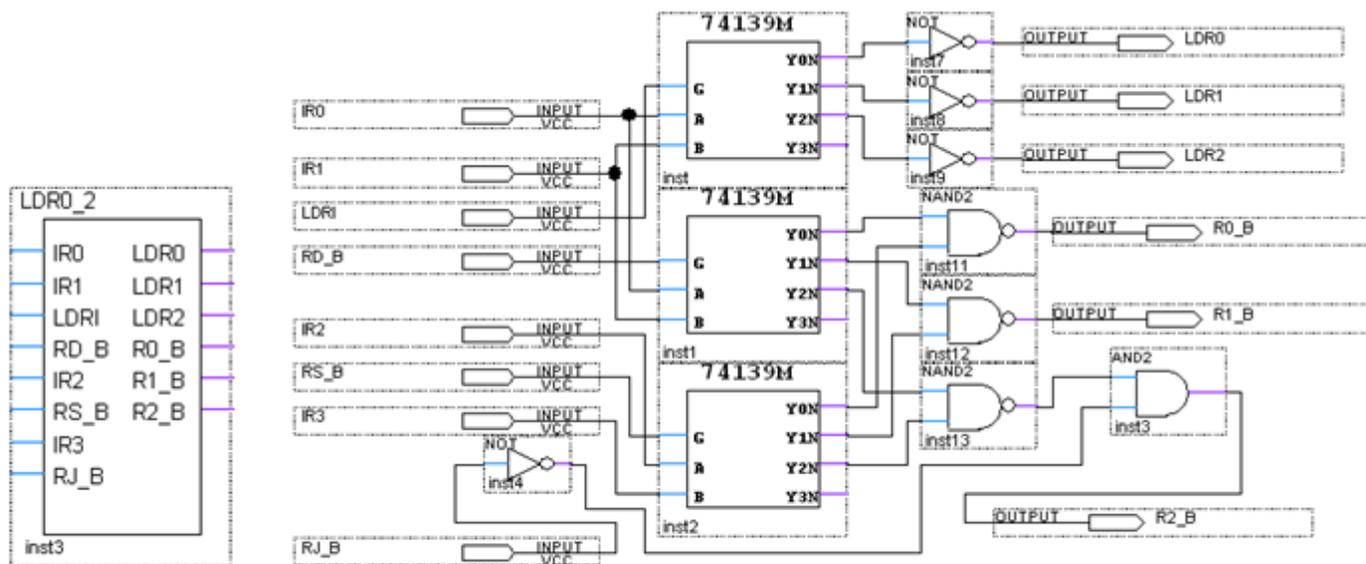


图 4-18 数据寄存器控制逻辑

# 4.2 计算机中的基本部件

## 4.2.9 时序发生器

### 1. 连续节拍发生电路

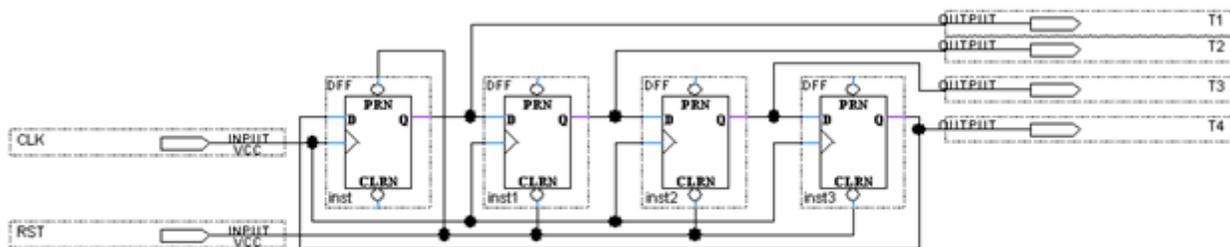


图 4-19 节拍脉冲发生器电路

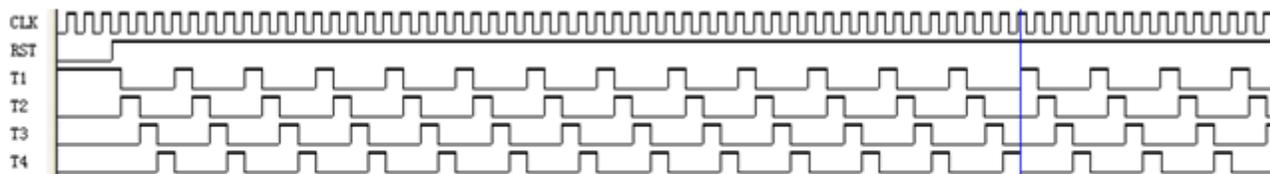


图 4-20 节拍脉冲发生器工作波形

# 4.2 计算机中的基本部件

## 4.2.9 时序发生器

### 2. 单步节拍发生电路

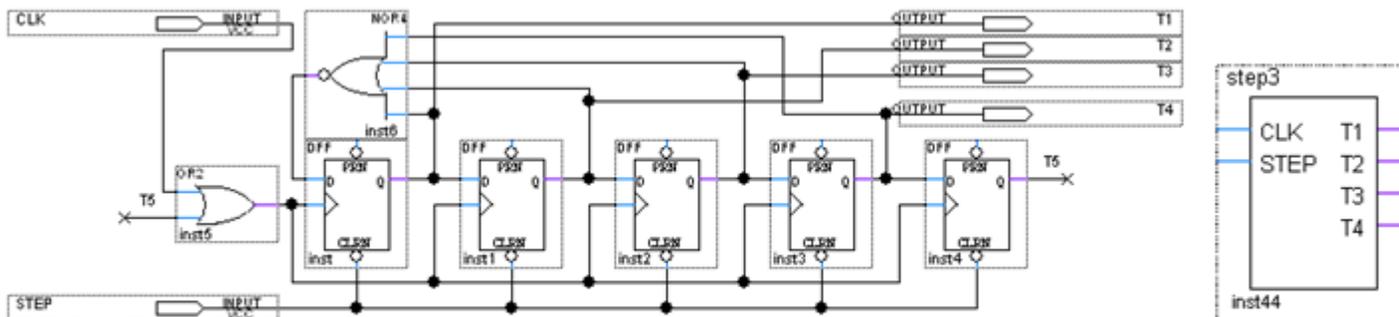


图 4-21 单步运行电路 step3 中的逻辑电路

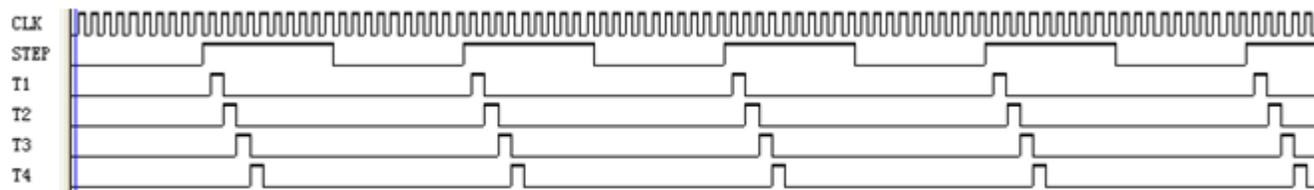


图 4-22 图 4-21 电路的单步运行电路工作波形

# 4.3 数据通路设计

## 4.3.1 模型计算机的数据通路

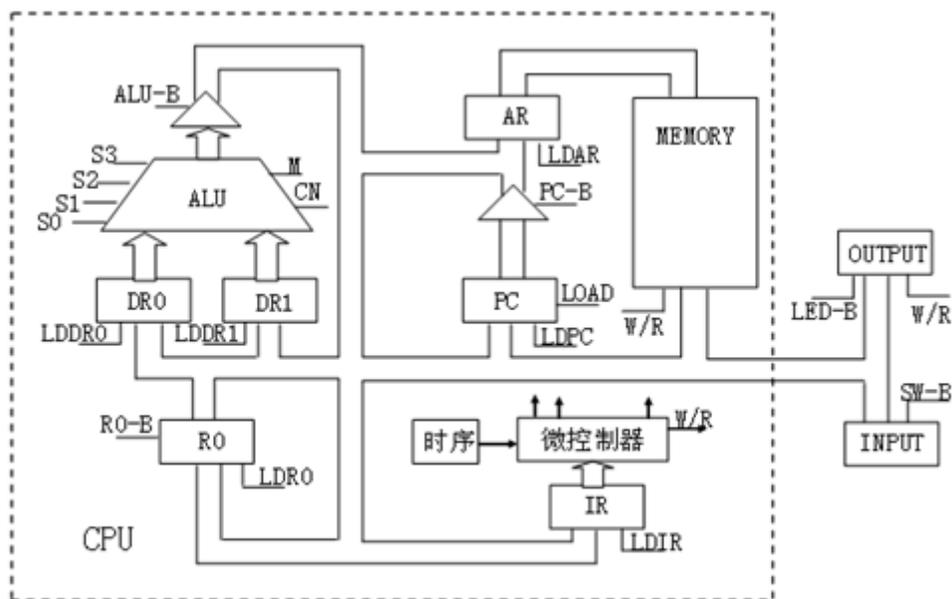


图 4-23 单总线结构的模型机 CPU 的内部数据通路

# 4.3 数据通路设计

## 4.3.2 模型机的电路结构

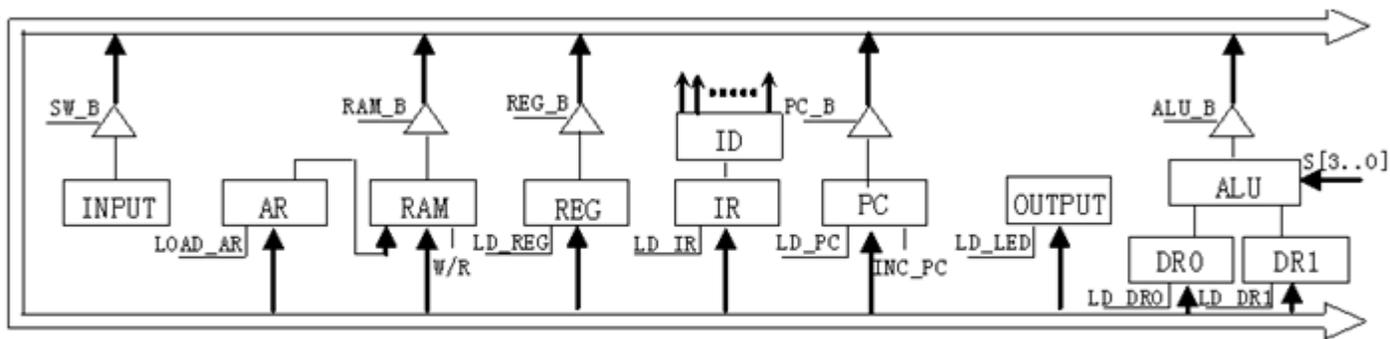


图 4-24 简单模型 CPU 的数据通路

# 实验与设计

## 4-1. 算术逻辑运算单元ALU设计实验

## 4-2. 带进位算术逻辑运算单元ALU设计实验

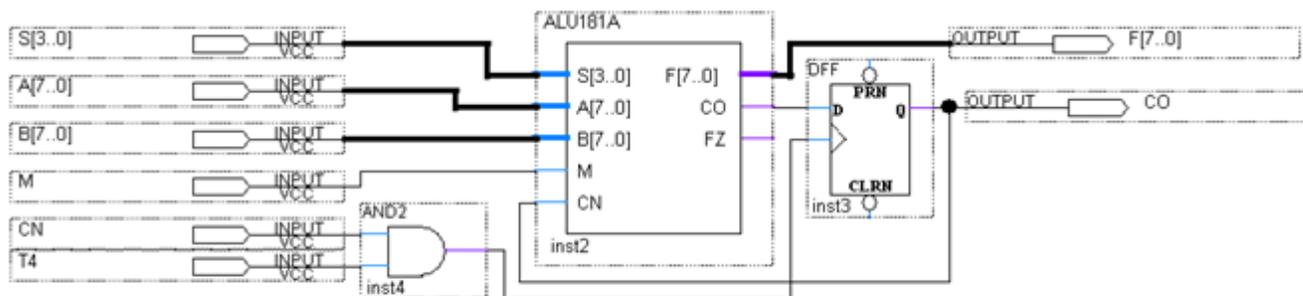


图 4-25 带进位控制的 ALU

# 实验与设计

## 4-2. 带进位算术逻辑运算单元ALU设计实验

表 4-3

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)	
0101	F F	0 1	F= ( )	F= ( )	F= ( )
0110	F F	0 1	F= ( )	F= ( )	F= ( )
0111	F F	0 1	F= ( )	F= ( )	F= ( )
1000	F F	F F	F= ( )	F= ( )	F= ( )
1001	F F	F F	F= ( )	F= ( )	F= ( )
1010	F F	F F	F= ( )	F= ( )	F= ( )

## 4-3. 移位运算器设计实验

# 实验与设计

## 4-4. LPM\_ROM实验

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 4-26 ROM 初始化文件 ROM\_A.mif 的内容

## 4-5. LPM\_RAM实验

## 4-6. 微控制器实验1：节拍脉冲发生器时序电路实验

# 实验与设计

## 4-7. 微控制器实验2：程序计数器PC与地址寄存器AR实验

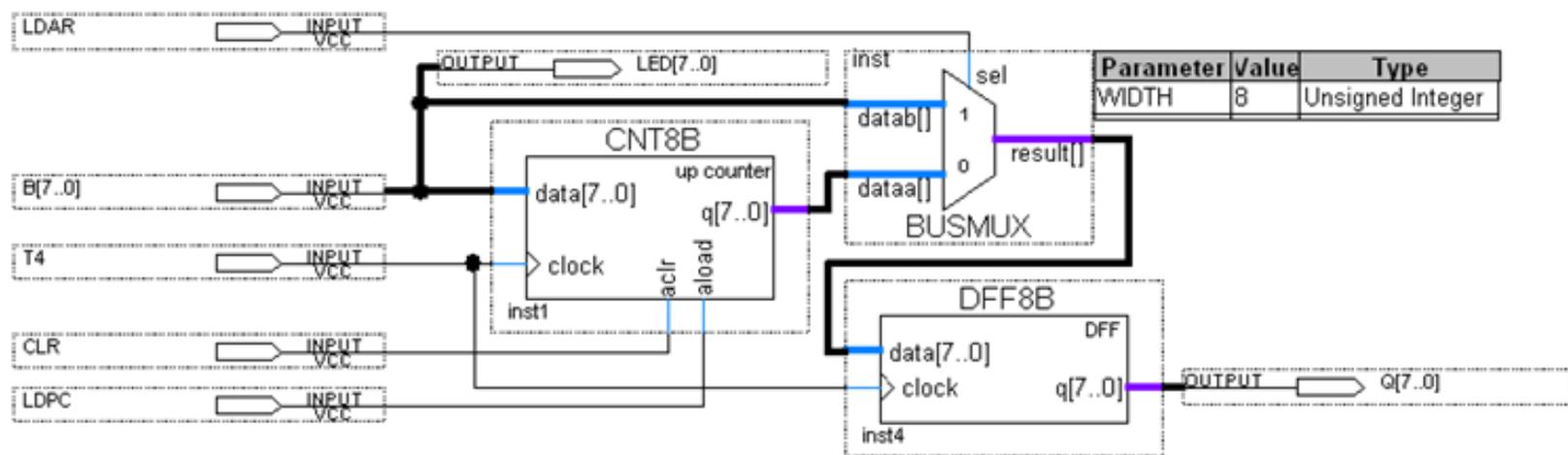


图 4-27 程序计数器实验电路原理图

# 实验与设计

## 4-7. 微控制器实验2：程序计数器PC与地址寄存器AR实验

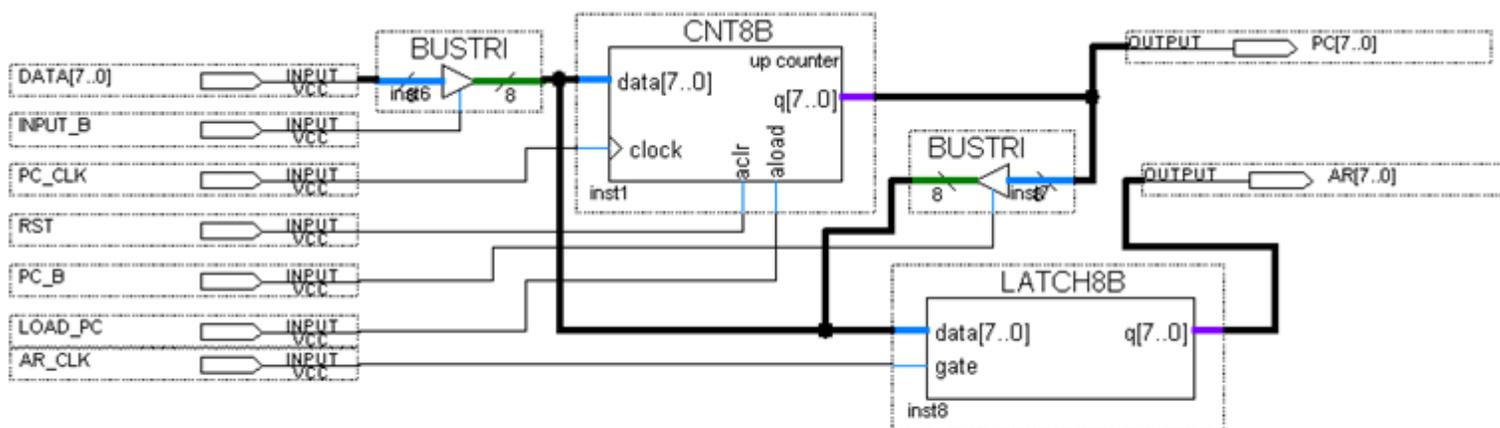


图 4-29 程序计数器与地址寄存器实验电路

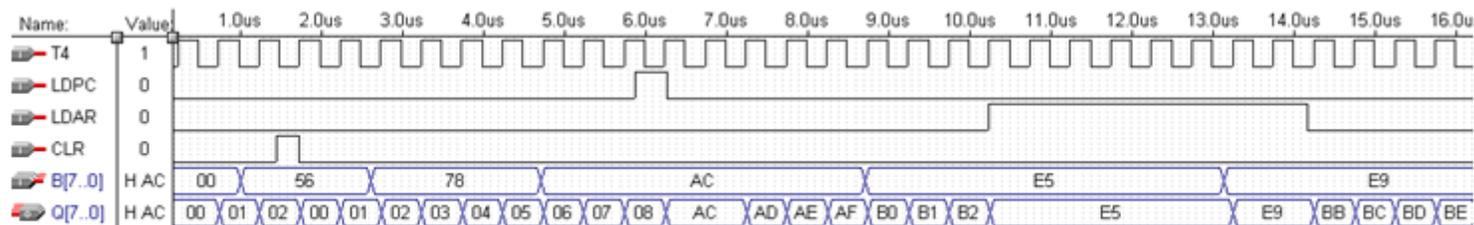


图 4-28 程序计数器工作波形

# 实验与设计

## 4-8. 微控制器实验3：微控制器组成实验

Addr	+0	+1	+2	+3	+4	+5	+6	+7
00	018108	00ED82	00C050	00A004	00E0A0	00E006	00A007	00E0A0
08	00ED8A	00ED8C	00A008	008001	062009	00A00E	01B60F	95EA25
10	00ED83	00ED85	00ED8D	00EDA6	001001	030401	018016	3D9A01
18	019201	01A22A	03B22C	01A232	01A233	01A236	318237	318239
20	009001	028401	05DB81	0180E4	018001	95AAA0	00A027	01BC28
28	95EA29	95AAA0	01B42B	959B41	01A42D	65AB6E	0D9A01	01AA30
30	0D8171	959B41	019A01	01B435	05DB81	B99B41	0D9A01	298838
38	019801	19883A	019801	070A08	062009	000000	000000	000000

图 4-30 存储器初始化文件的内容