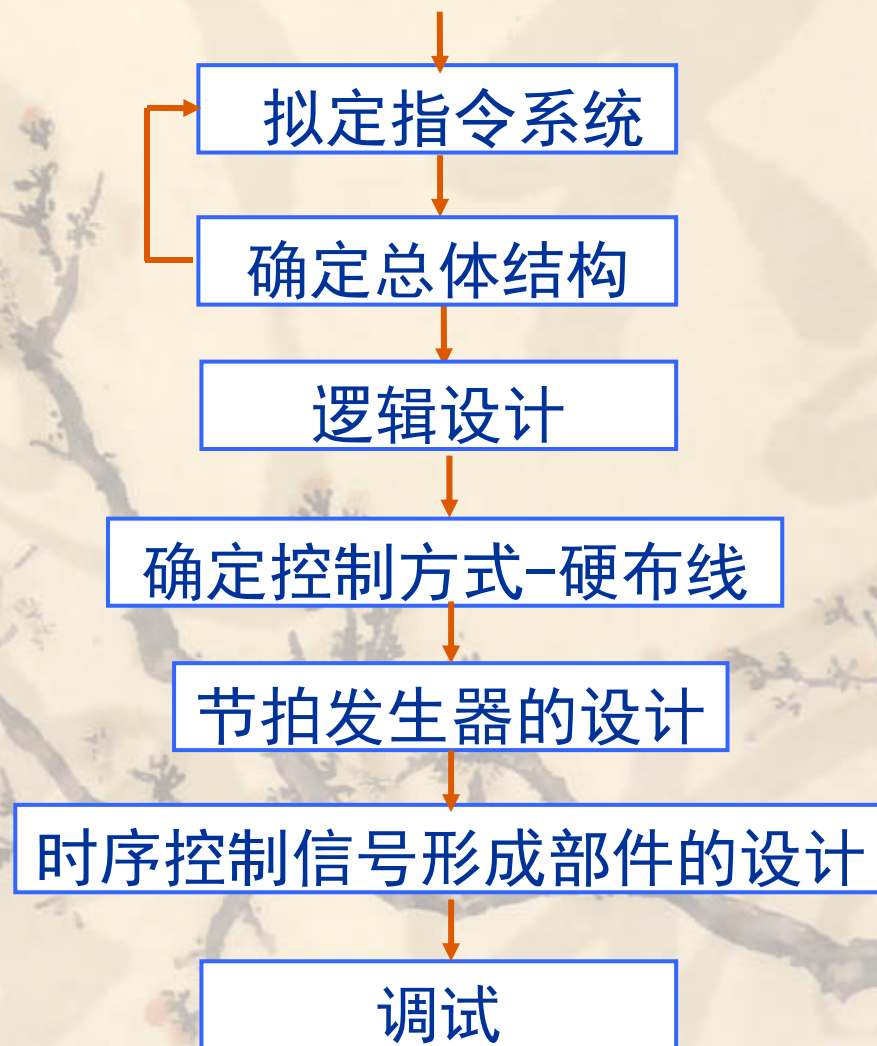


第4讲—简单模型机的设计 (硬布线实现)

计算机学院

张瑞华

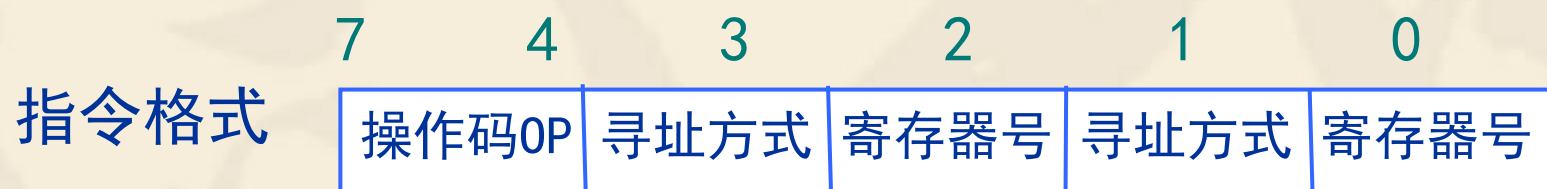
模型机设计步骤



1、拟定指令系统

- ❖ 拟定指令系统将涉及到基本字长、指令格式、指令种类、寻址方式等内容。这些内容的确定又和总体结构密切相关。
- ❖ 基本字长
 - ∞ 存储器容量为 256×8 ，基本字长定为8位
- ❖ 指令格式
 - ∞ 指令格式可有单字长指令和双字长指令两种
 - ∞ 在双字长格式中，第二字节一般定义为操作数或操作数地址。

基本字长 8位



源操作数

目的操作数

❖ 指令类型

- ❧ 模型机有单操作数指令、双操作数指令和无操作数指令。
- ❧ 操作码OP共4位，最多可定义16条指令。

❖ 寻址方式

- ❧ 当寻址方式位为0，是寄存器寻址，操作数在指定的寄存器中，相应的寄存器号位为0是R0，为1是寄存器 R1；
- ❧ 当寻址方式位为1时，寻址方式位和寄存器号位组合，
 - ❖ 10：是立即数寻址，操作数在指令的下一个单元；
 - ❖ 11：是直接寻址，操作数地址在指令的下一个单元。

2、确定总体结构

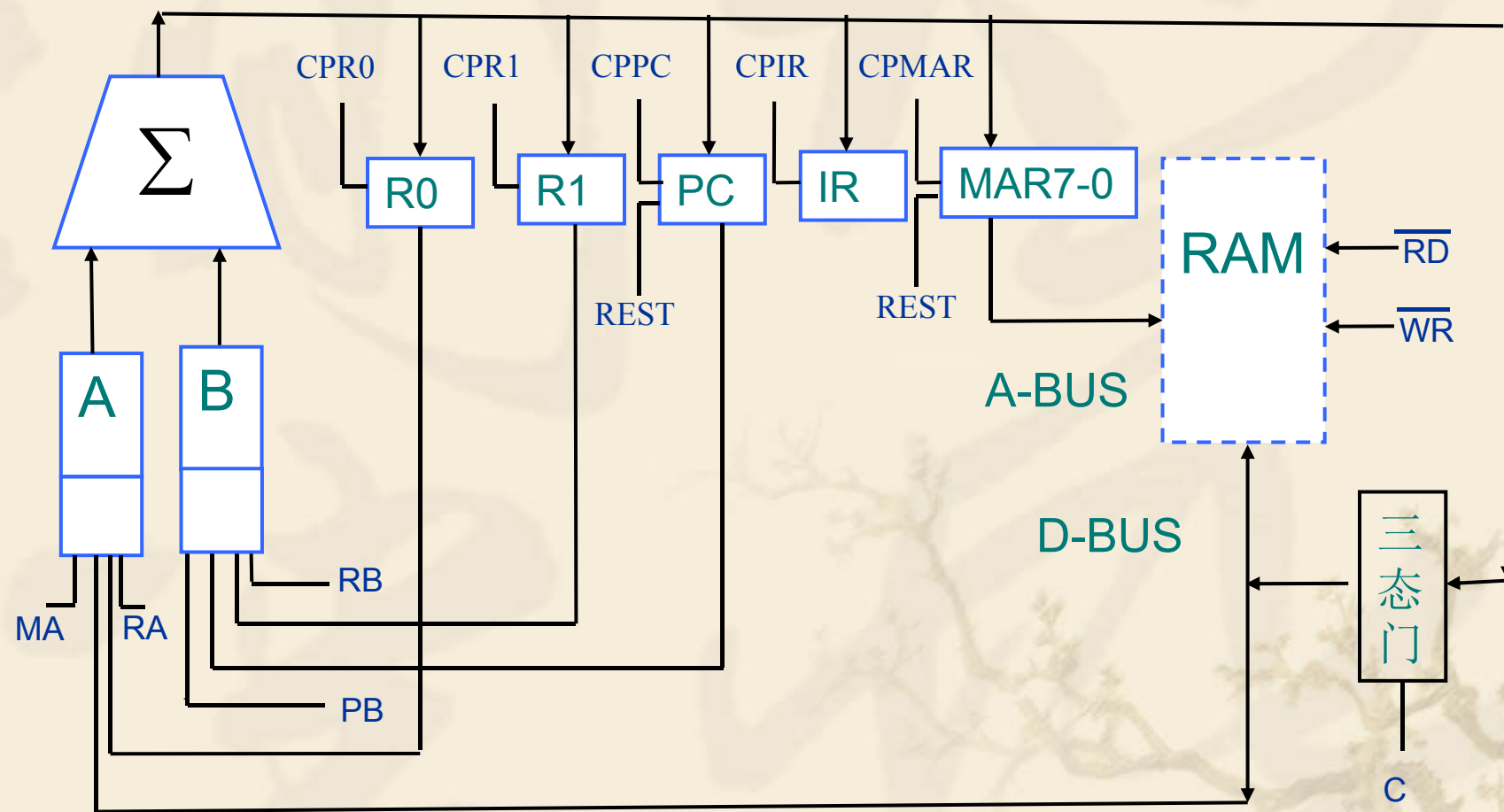


图1

❖ 寄存器组的设置

- ❧ R_0 、 R_1 为通用寄存器，8位。
- ❧ IR为指令寄存器，8位。
- ❧ PC程序计数器，8位。
- ❧ MAR为地址寄存器，8位。

❖ 加法器ALU的设置

- ❧ 采用74181、74182实现

❖ 选择器的设置

- ❧ 连入A选择器的数据来源是RAM的读出数据和 R_0 寄存器的数据。
- ❧ 连入B选择器的数据来源是PC的数据和 R_1 的数据。

❖ 数据通路

❧ 模型机的数据通路是以总线为基础,以CPU为核心构成的。

❖ 取指令:

❧ MA A直传 CPIR

❧ $RAM \rightarrow \text{选择器A} \rightarrow \Sigma \rightarrow \text{Bus} \rightarrow \text{IR}$

❖ 送指令地址

❧ PB B直传 CPMAR

❧ $PC \rightarrow \text{选择器B} \rightarrow \Sigma \rightarrow \text{Bus} \rightarrow \text{MAR}$

❖ 指令计数器+1

❧ PB A加B加1 (A为0) cppc

❧ $PC \rightarrow \text{选择器B} \rightarrow \Sigma \rightarrow \text{Bus} \rightarrow \text{PC}$

❖ $R_0 \rightarrow R_1$

∞

RA

A直传

CPR1

∞ $R_0 \rightarrow \text{选择器A} \rightarrow \Sigma \rightarrow \text{Bus} \rightarrow R_1$

❖ $R_1 \rightarrow \text{RAM}$

∞

RB

B直传

C

WR

∞ $R_1 \rightarrow \text{选择器B} \rightarrow \Sigma \rightarrow \text{Bus} \rightarrow \text{RAM}$

3、 逻辑设计

- ❖ 总体结构中，虚线框内的RAM是FPGA之外预先配置好的。
- ❖ （1）ALU的逻辑设计
 - ∞ 由74181和74182组成。

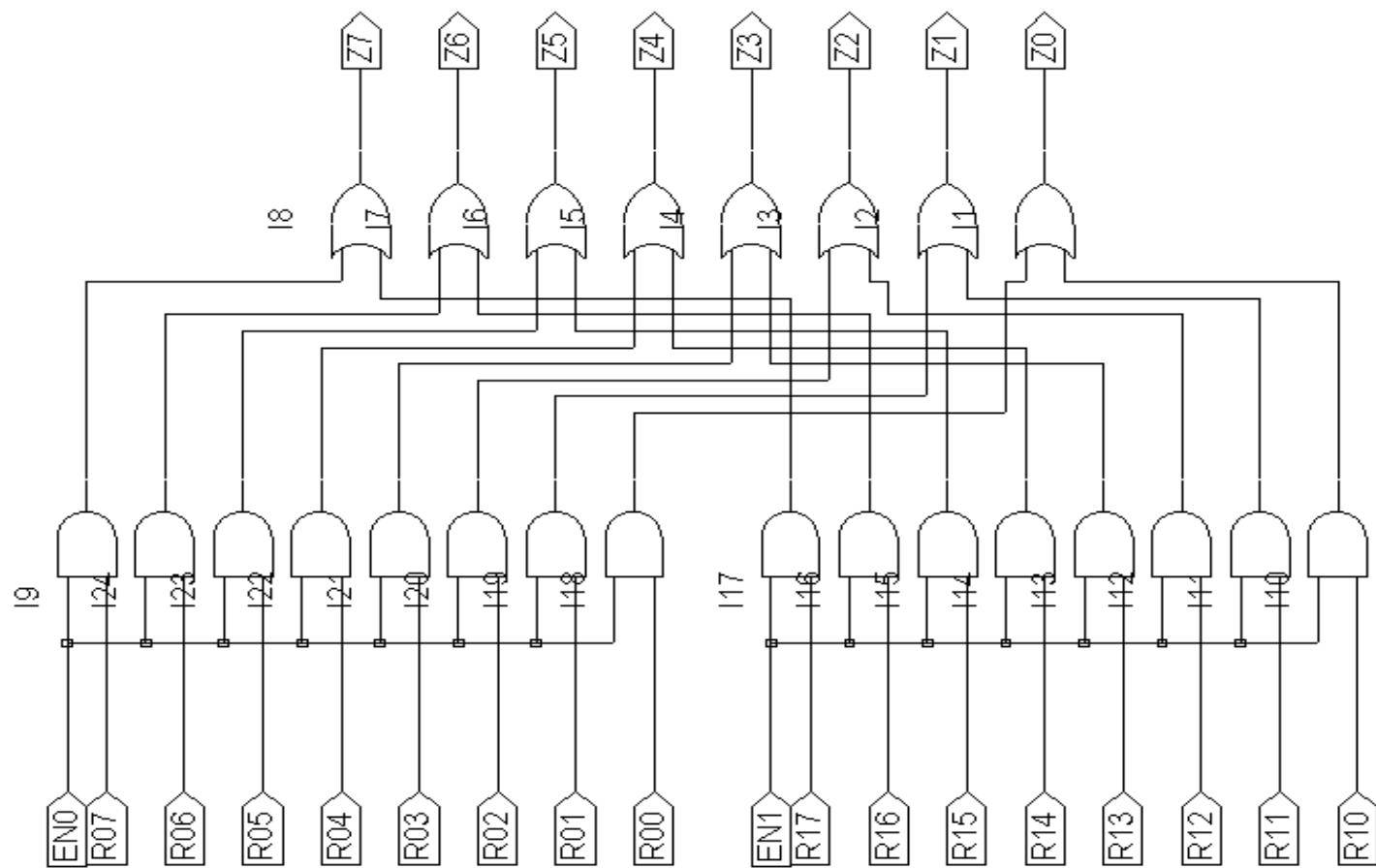


图2 选择器设计

❖ (2) 寄存器的设计

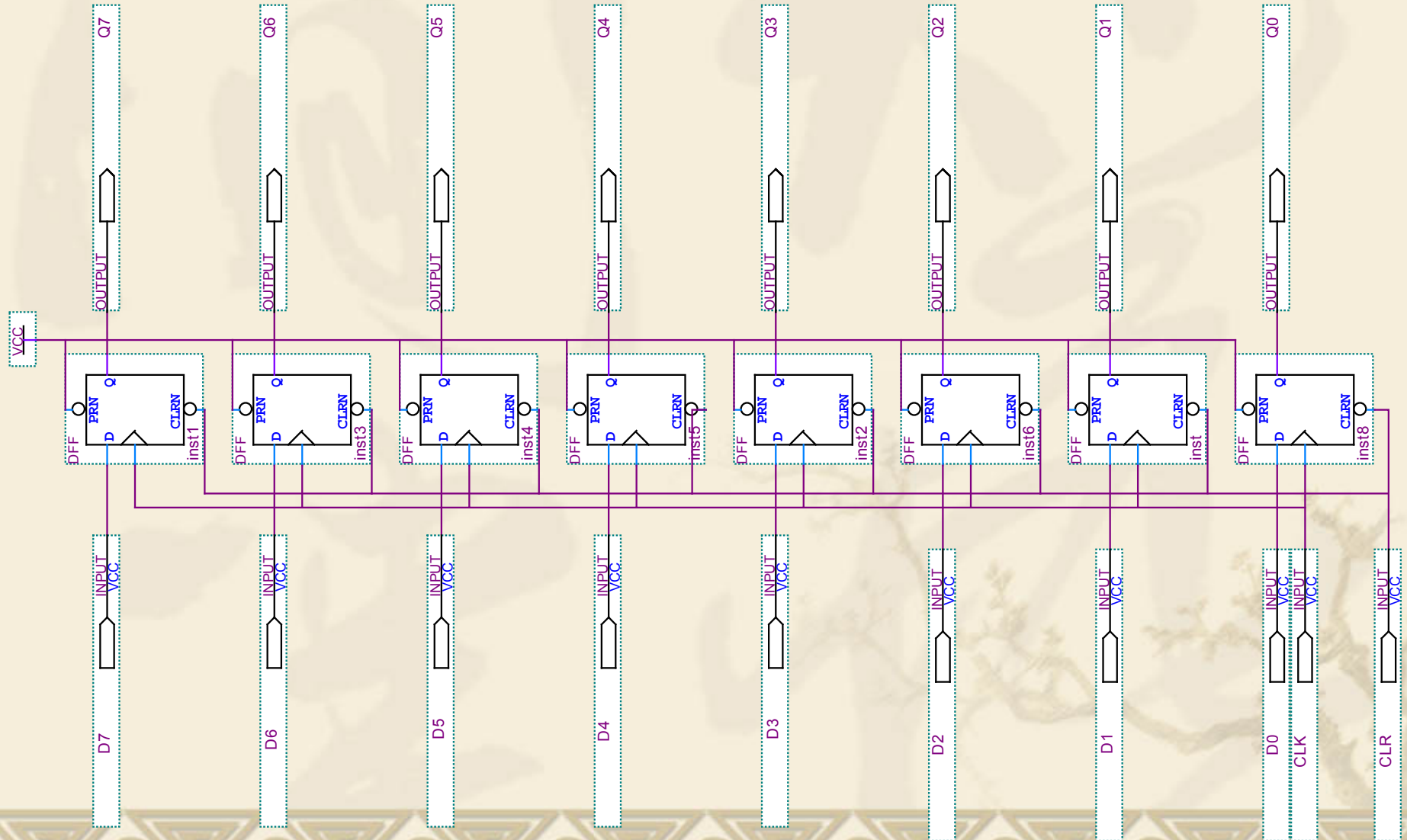
❖ 不带复位的寄存器

- ❧ 结构中 R_0 、 R_1 通用寄存器，可存放操作数或结果、中间结果，每个寄存器均由8个D触发器构成。
- ❧ 在 CPR_i 的作用下接收总线的数据送入寄存器，输出连入选择器。
- ❧ 指令寄存器IR其结构同通用寄存器。

❖ 带复位的寄存器

- ❧ 结构中MAR地址寄存器是一个带复位的寄存器，带复位是指当有复位信号时，MAR清零。
- ❧ 逻辑图如图3所示。

图3 带复位的八位寄存器逻辑图



❖ 程序计数器PC的设计

- ❧ 程序计数器结构如上图3所示，是有复位信号的8位寄存器。PC加1是通过加法器实现的。
- ❧ 复位信号RET 的作用是有复位信号时，计数器PC清零。

❖ (3) 三态门的设计

- ❧ 直接利用库器件tri实现：
- ❧ C=L时，三态；C=H时，Y=A

❖ (4) 部件之间的连接

- ❧ 由系统结构图（图1）可看出，部件之间的连接是采用以**CPU**为中心的总线连接方式。
- ❧ 加法器的输出通过总线**BUS**连接到所有寄存器和存储器的输入端，除指令寄存器**IR**和地址寄存器**MAR**的输出端外，其它部件的输出端分别送入选择器**A**和选择器**B**。
- ❧ 连线图如图4所示。

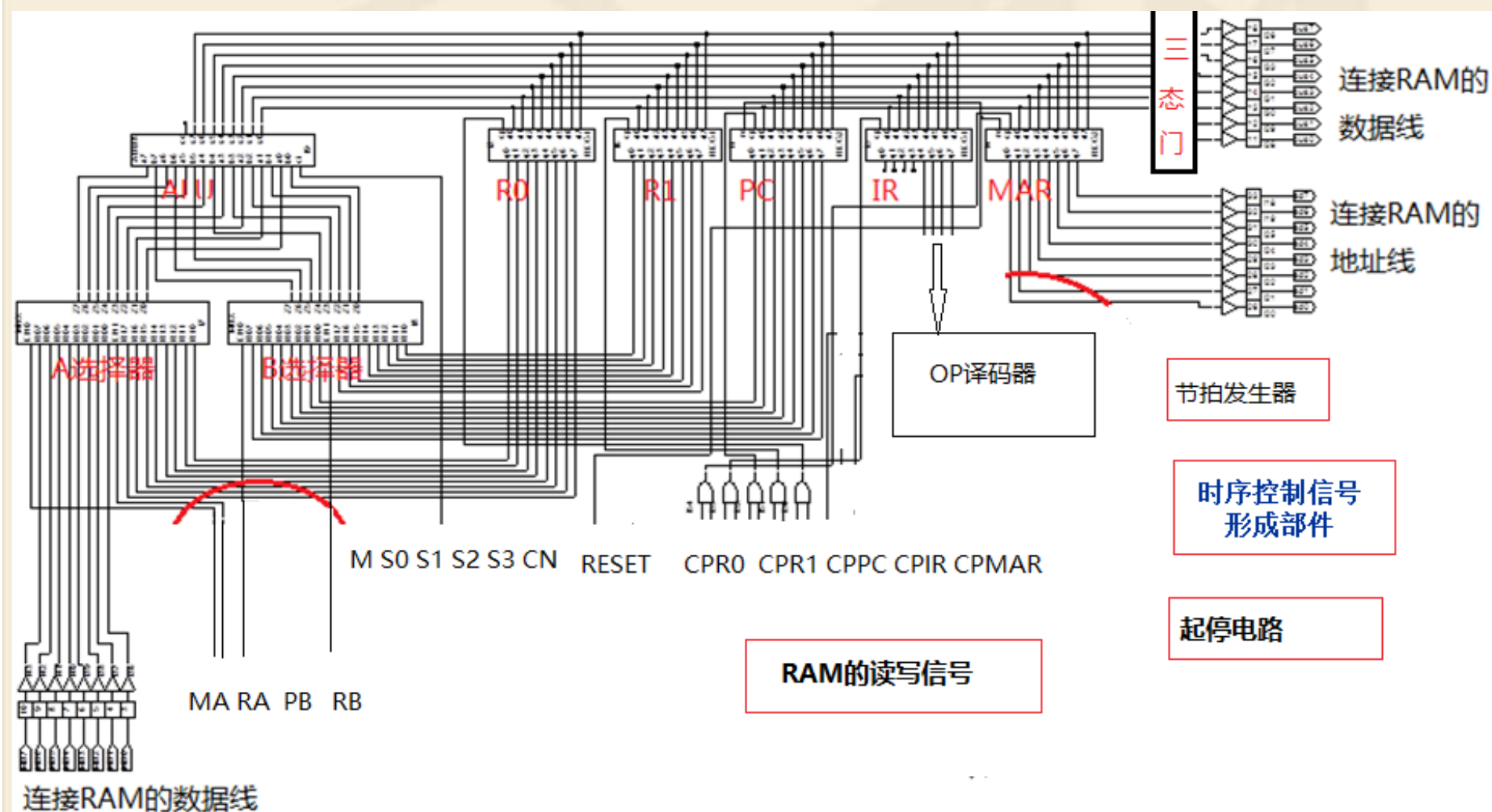


图4

4、确定控制方式

- ❖ 控制命令是确定信息的流向，不同的数据通路需要不同的控制命令。
- ❖ 架构图图1中，涉及到了许多控制命令例如 CPR_0 、CPMAR、MA、RB等等，这些命令如何产生？
- ❖ 通常有两种方式，即硬布线逻辑电路方式和微程序方式。
- ❖ 本讲模型机采用硬布线逻辑电路方式。

确定总体结构

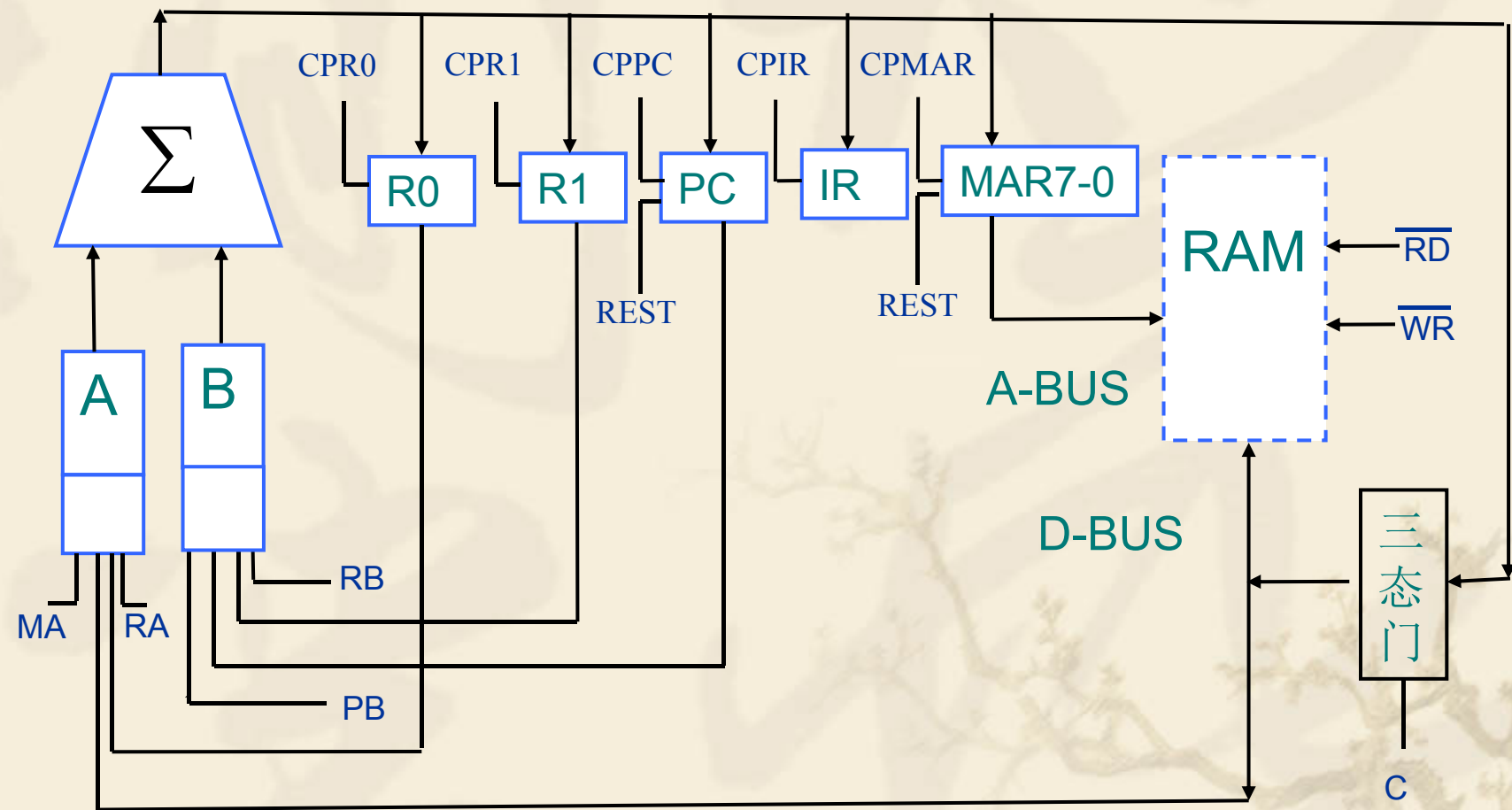
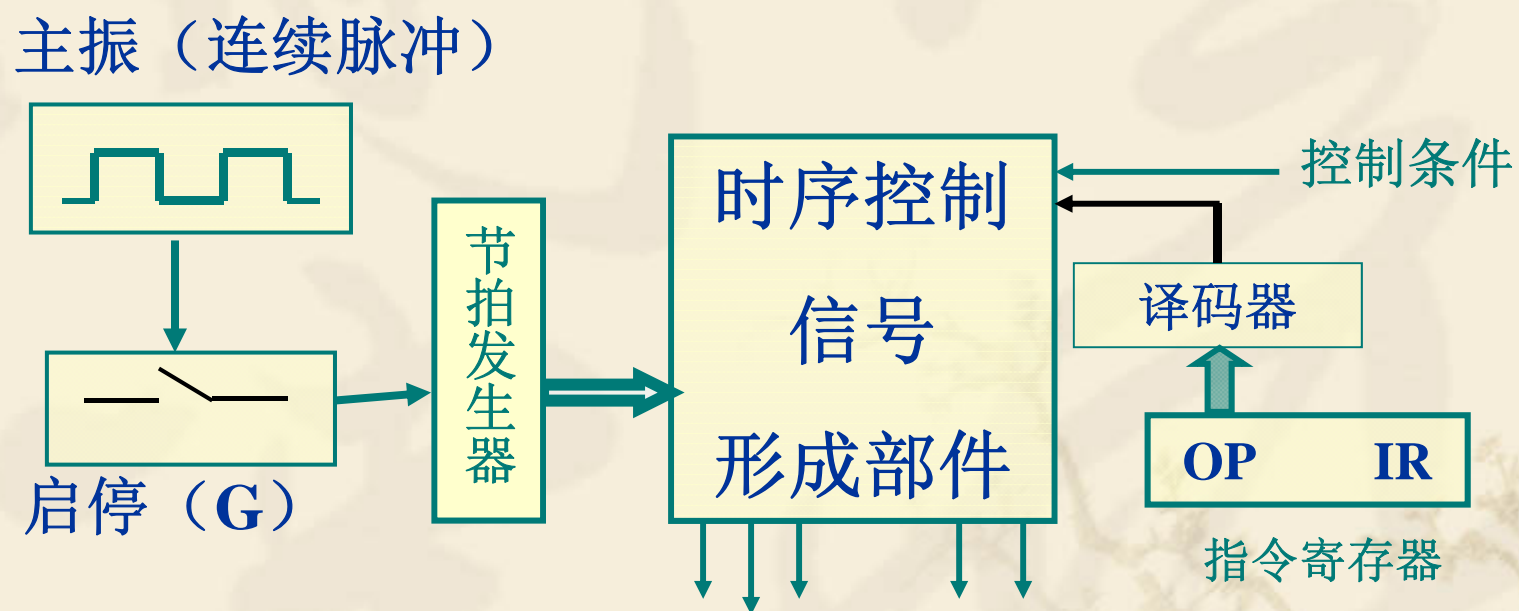


图1

❖ (1) 硬布线逻辑电路控制器的结构图:



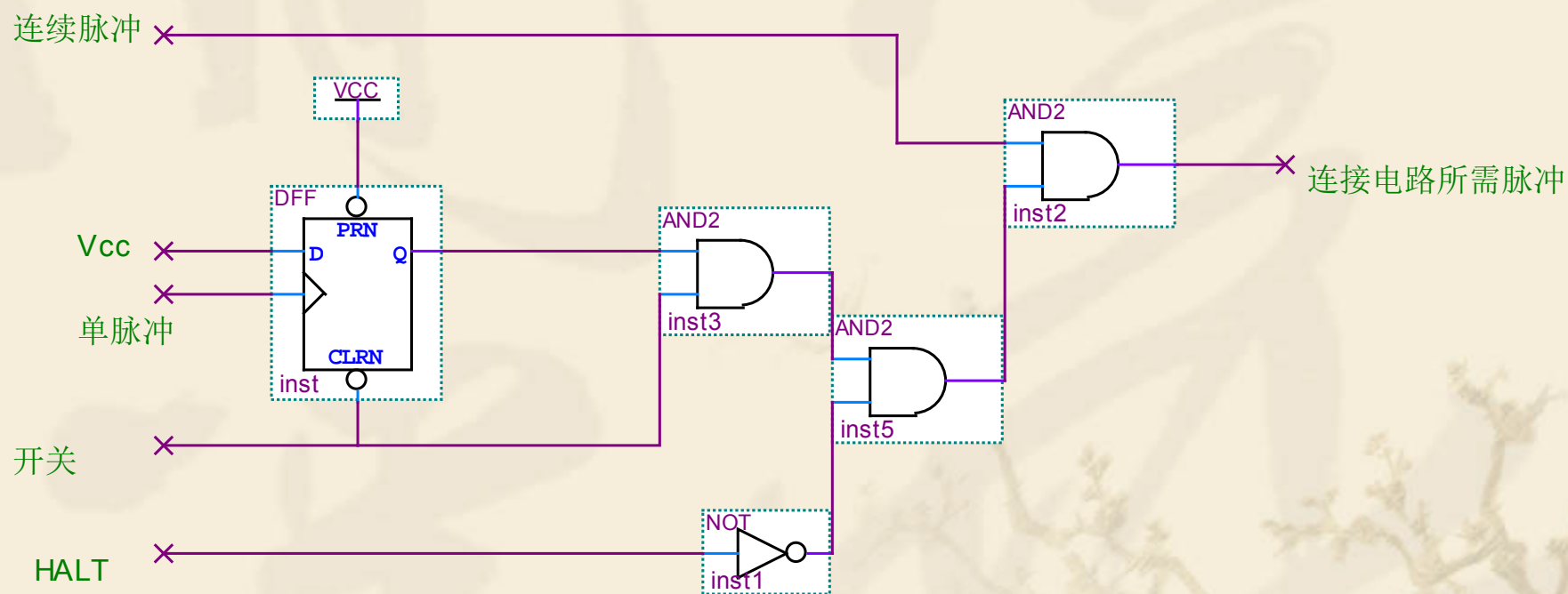
- ❖ 两种实现方式的区别：
- ❖ 两种控制器实质性的差别，表现在处理指令各执行步骤的接续关系的方案和给出时序控制信号的方法完全不同，从而造成控制器的具体组成和运行原理、运行性能上的一些差异。
- ❖ 两种控制器组成的主要差异：微程序控制器中的控存变成这里的**时序信号产生部件**，还取消了微指令寄存器；原来的下地址形成部件变成了这里的**节拍发生器**；原来的微地址映射部件变成这里的**操作码译码器**；一些信号连接关系也有某些变化。

组合逻辑控制器的组成与运行原理

- ❖ 组合逻辑控制器用**节拍发生器**（**Timing**，几个触发器构成的时序逻辑电路）不同的状态组合来区分一条指令不同的执行步骤，指令执行步骤的接续是通过变换节拍发生器的状态组合完成的，不同于微程序控制器中通过下地址部件给出不同的微指令地址来实现。这里用节拍发生器取代了原来的下地址形成部件。
- ❖ 组合逻辑控制器是通过由“**与—或**”两级逻辑关系构成的时序控制信号产生部件来直接给出全部的时序控制信号。送到第一级各“**与门**”的输入信号是指令操作码和节拍发生器的节拍状态(可能还有控制条件)，每个与门产生一个与项输出，相关的与项输出信号送到第二级的“**或门**”，**每个或门输出的就是一个时序控制信号**。全部的时序控制信号由许多个“与—或”逻辑门给出。与用控制存储器存放全部控制信号的微程序控制方案不同，这里用时序控制信号产生部件取代了原来的控制存储器，还取消了那里的微指令寄存器线路，把控制信号直接送到被控制的部件。

- ❖ (2) 硬布线控制器各部件的设计
- ❖ 主振和启停电路的设计
- ❖ 译码器的设计
- ❖ 其它部件的设计:
 - ❧ 所有指令执行步骤划分和功能确定;
 - ❧ 节拍发生器 (TIMING) 设计与实现;
 - ❧ 时序控制信号产生部件的设计与实现;

- ❖ 主振和启停电路的设计
- ❖ 主振可使用连续脉冲，程序执行后，查看结果。
- ❖ 电路如下：



❖ 操作码**OP**译码器的设计

❖ 利用指令的操作码，得出逻辑表达式：

$$MOV1 = \overline{IR_7} \overline{IR_6} \overline{IR_5} IR_4$$

$$MOV2 = \overline{IR_7} \overline{IR_6} IR_5 \overline{IR_4}$$

$$ADD = \overline{IR_7} \overline{IR_6} IR_5 IR_4$$

$$MOV3 = \overline{IR_7} IR_6 \overline{IR_5} \overline{IR_4}$$

$$HALT = \overline{IR_7} IR_6 \overline{IR_5} IR_4$$

❖ 使用上述逻辑表达式，利用与门、非门设计电路。

所有指令执行步骤划分和功能确定

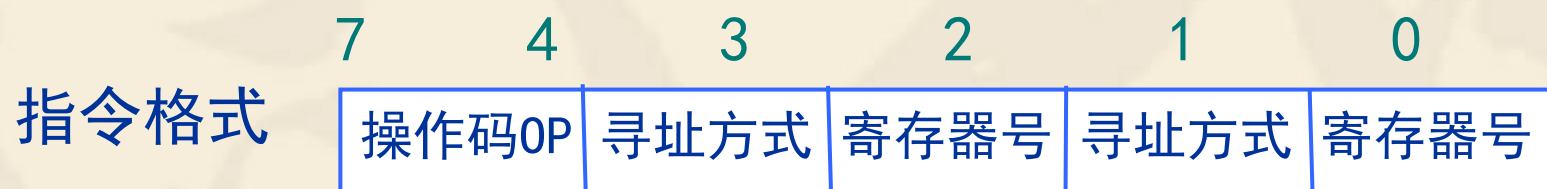
❖ 编写程序

- ❧ MOV1 05#, R₀
- ❧ MOV2 01#, R₁
- ❧ ADD R₀, R₁
- ❧ MOV3 R₁, (FA#)
- ❧ HALT

❖ 操作码二进制代码

- ❧ MOV1: 0001
- ❧ MOV2: 0010
- ❧ ADD : 0011
- ❧ MOV3: 0100
- ❧ HALT: 0101

基本字长 8位



源操作数

目的操作数

❖ 指令类型

- ❧ 模型机有单操作数指令、双操作数指令和无操作数指令。
- ❧ 操作码OP共4位，最多可定义16条指令。

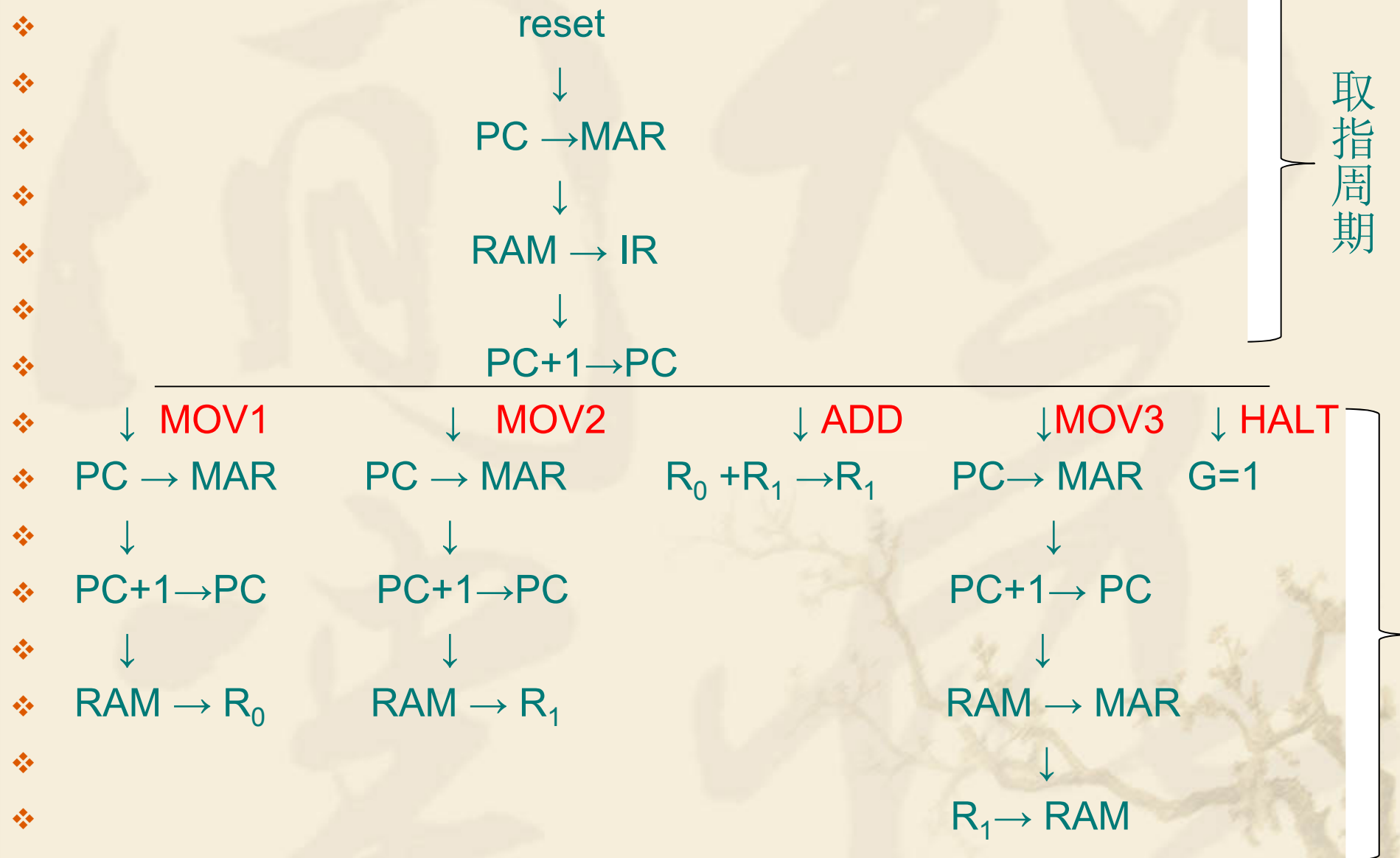
❖ 寻址方式

- ❧ 当寻址方式位为0，是寄存器寻址，操作数在指定的寄存器中，相应的寄存器号位为0是R0，为1是寄存器 R1；
- ❧ 当寻址方式位为1时，寻址方式位和寄存器号位组合，
 - ❖ 10：是立即数寻址，操作数在指令的下一个单元；
 - ❖ 11：是直接寻址，操作数地址在指令的下一个单元。

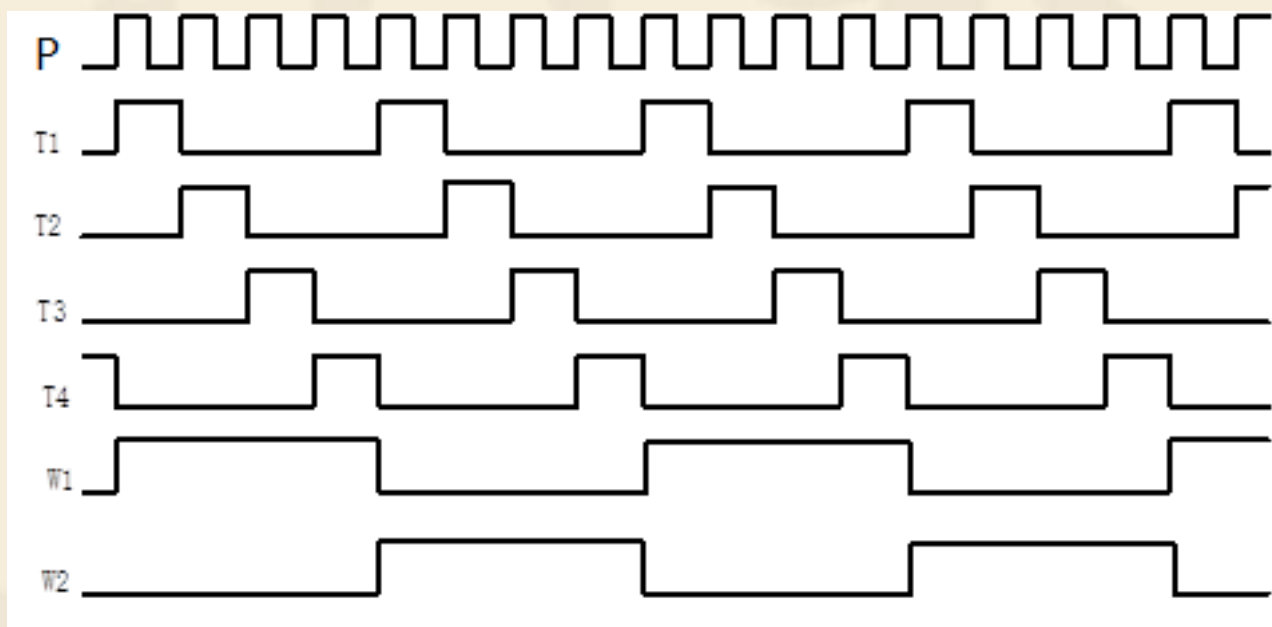
❖ 程序代码:

地址	代码	地址	内容
0	0001 10 00	1	0000 0101
2	0010 10 01	3	0000 0001
4	0011 00 01		
5	0100 01 11	6	1111 0101
7	0101 00 00		

❖ 所有指令执行流程

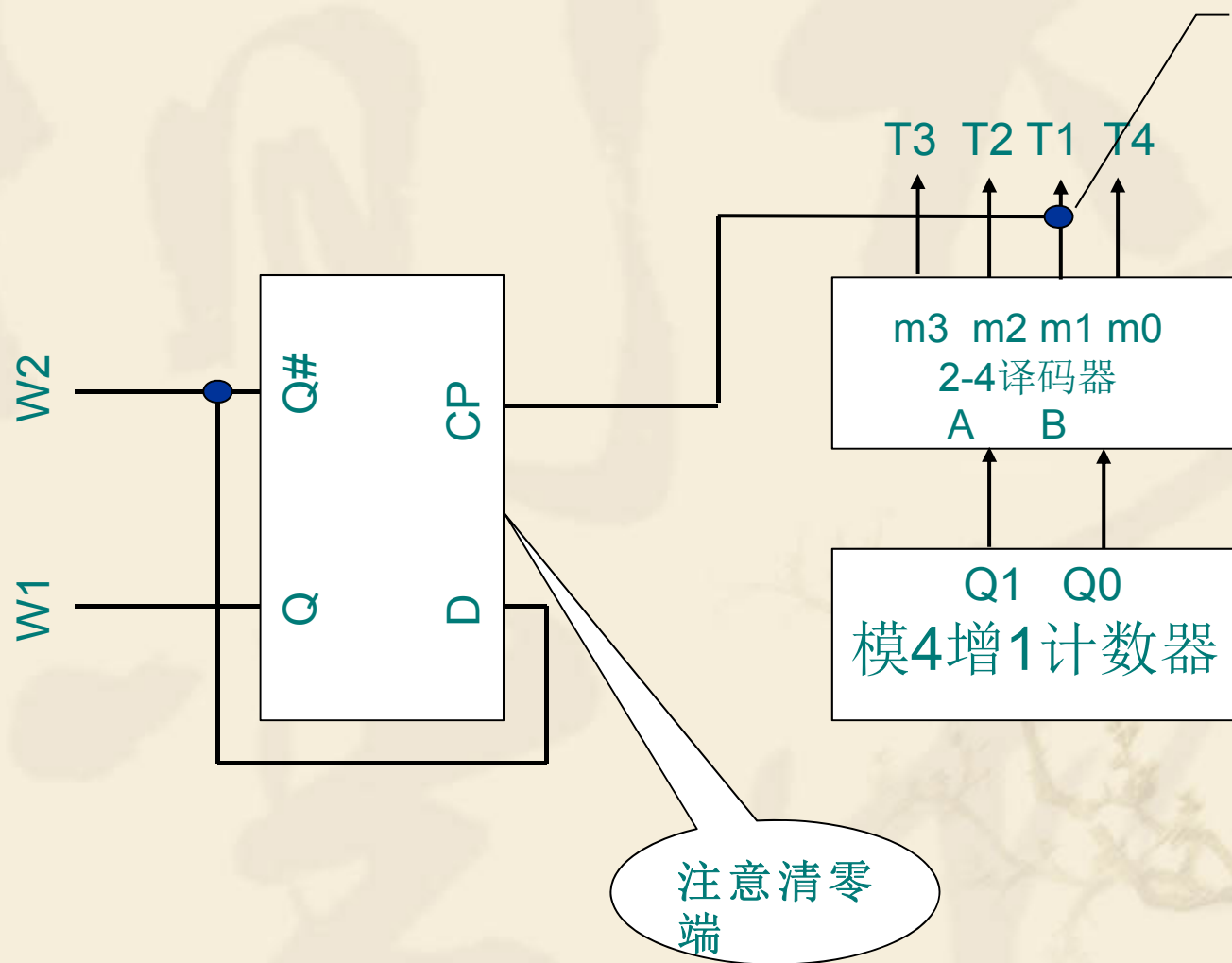


- ❖ 根据指令执行流程，设计时序为：分取指周期和执行周期，每个周期为4节拍，波形图如下：



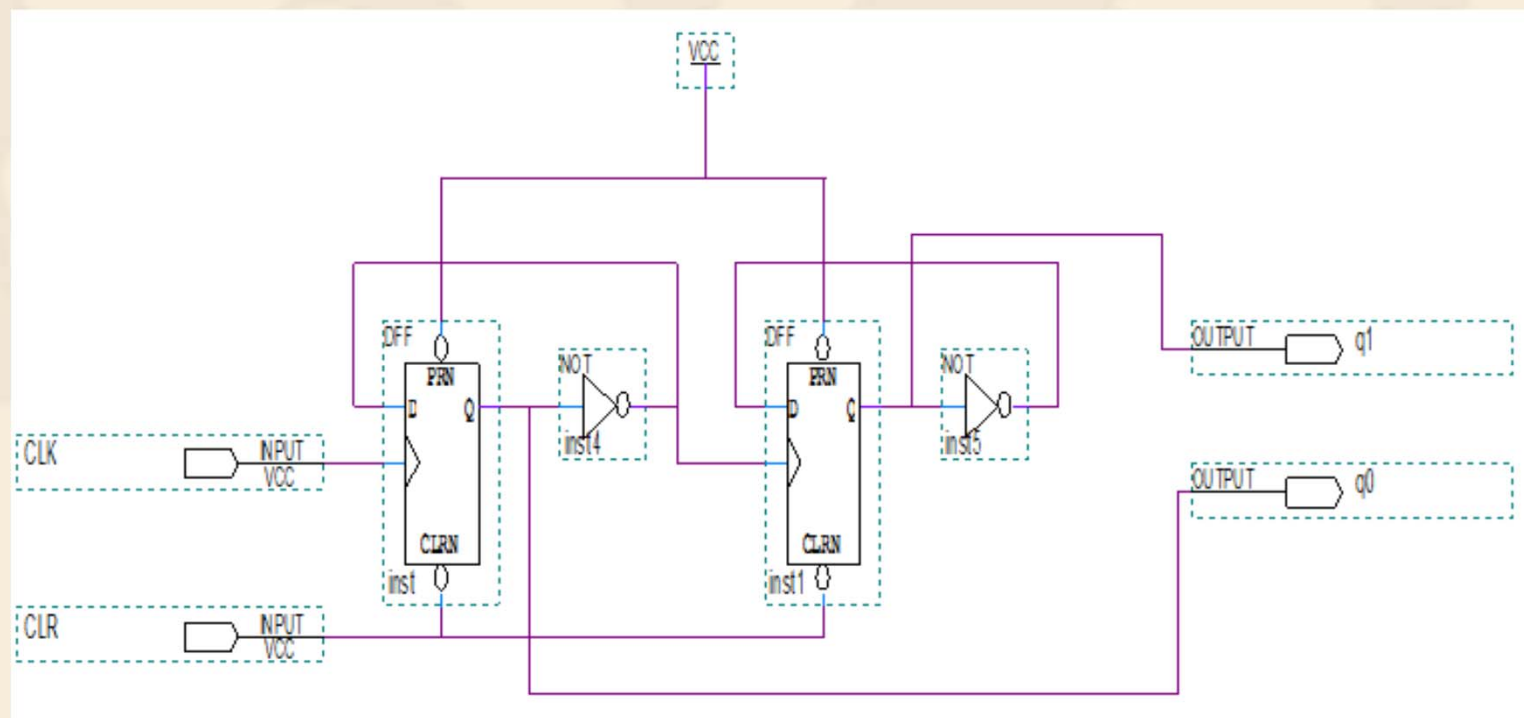
- ❖ W1高电平时是取指周期的4节拍，W2高电平时是执行周期的4节拍。该电路设计类似计组的课内实验—时序电路。

❖ 节拍发生器实现电路框图如下：

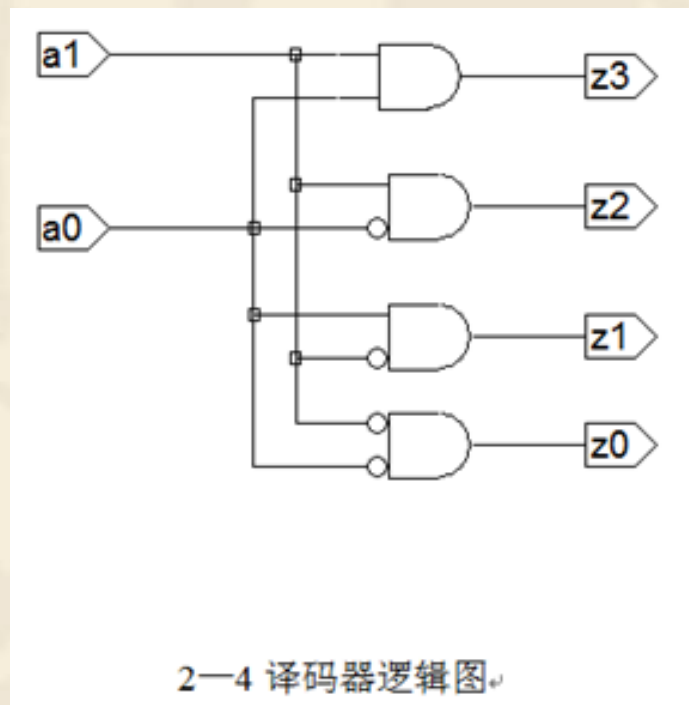


Q1Q0变换00、01、10、11，从01、10的变化有中间状态：00；同理，从11、00的变化有中间状态：10，所以，m0、m2不能做触发脉冲。顺序不变。

❖ 模4增1计数器



❖ 2-4译码器



❖ 时序控制信号产生部件的设计

❖ 根据指令执行流程和数据通路，各控制信号的列表如下：

状态	W1（取指周期）	W2（执行周期）			
指令	所有指令	MOV1	MOV2	ADD	MOV3
控制信号					
MA	T2	T3	T3		T3
RA				T1	
PB	T1、T3	T1、T2	T1、T2		T1、T2
RB				T1	T4
CPR0		T3			
CPR1			T3	T1	
CPPC	T3	T2	T2		T2
CPIR	T2				
CPMAR	T1	T1	T1		T1、T3
RD	T2	T3	T3		T3
WR					T4
驱动C					T4
G					
M	1 (T1)、1 (T2)、0 (T3)	1 (T1)、0 (T2)、1 (T3)	1 (T1)、0 (T2)、1 (T3)	0 (T1)	1 (T1)、0 (T2)、1 (T3)、1 (T4)
S3	1 (T1)、1 (T2)、1 (T3)	1 (T1)、1 (T2)、1 (T3)	1 (T1)、1 (T2)、1 (T3)	1 (T1)	1 (T1)、1 (T2)、1 (T3)、1 (T4)
S2	0 (T1)、1 (T2)、0 (T3)	0 (T1)、0 (T2)、1 (T3)	0 (T1)、0 (T2)、1 (T3)	0 (T1)	0 (T1)、0 (T2)、1 (T3)、0 (T4)
S1	1 (T1)、1 (T2)、0 (T3)	1 (T1)、0 (T2)、1 (T3)	1 (T1)、0 (T2)、1 (T3)	0 (T1)	1 (T1)、0 (T2)、1 (T3)、1 (T4)
S0	0 (T1)、1 (T2)、1 (T3)	0 (T1)、1 (T2)、1 (T3)	0 (T1)、1 (T2)、1 (T3)	1 (T1)	0 (T1)、1 (T2)、1 (T3)、0 (T4)
CN	0 (T3)	0 (T2)	0 (T2)	1 (T1)	0 (T2)

表中单元格内，填写的是该控制信号发生的条件

❖ 根据列表，各控制信号的逻辑表达式为：

$$MA = W_1T_2 + W_2T_3(MOV1 + MOV2 + MOV3)$$

$$RA = W_2ADD \bullet T_1$$

$$PB = W_1(T_1 + T_3) + W_2(T_1 + T_2)(MOV1 + MOV2 + MOV3)$$

$$RB = W_2(ADD \bullet T_1 + MOV3 \bullet T_4)$$

$$CPR_0 = W_2 \bullet MOV1 \bullet T_3 \bullet P$$

$$CPR_1 = W_2(ADD \bullet T_1 + MOV2 \bullet T_3)P$$

$$CPPC = (W_1T_3 + W_2T_2(MOV1 + MOV2 + MOV3))P$$

$$CPIR = W_1T_2P$$

$$CPMAR = (W_1T_1 + W_2(MOV1 \bullet T_1 + MOV2 \bullet T_1 + MOV3 \bullet (T_1 + T_3)))P$$

$$\overline{RD} = \overline{W_1T_2 + W_2T_3(MOV1 + MOV2 + MOV3)}$$

$$\overline{WR} = \overline{W_2MOV3 \bullet T_4}$$

$$C = W_2MOV3 \bullet T_4$$

$$G = W_2HALT \bullet T_1$$

$$\begin{aligned}
M &= W_1(T_1 + T_2 + \overline{T_3}) + W_2 \bullet MOV1 \bullet (T_1 + \overline{T_2} + T_3) + W_2 \bullet MOV2 \bullet (T_1 + \overline{T_2} + T_3) \\
&\quad + W_2 \bullet ADD \bullet \overline{T_1} + W_2 \bullet MOV3 \bullet (T_1 + \overline{T_2} + T_3 + T_4) \\
&= W_1\overline{T_3} + W_2 \bullet MOV1 \bullet \overline{T_2} + W_2 \bullet MOV2 \bullet \overline{T_2} + W_2 \bullet ADD \bullet \overline{T_1} + W_2 \bullet MOV3 \bullet \overline{T_2}
\end{aligned}$$

$$S_3 = 1$$

$$\begin{aligned}
S_2 &= W_1(\overline{T_1} + T_3 + T_2) + W_2 \bullet MOV1 \bullet (\overline{T_1} + T_2 + T_3) + W_2 \bullet MOV2 \bullet (\overline{T_1} + T_2 + T_3) \\
&\quad + W_2 \bullet ADD \bullet \overline{T_1} + W_2 \bullet MOV3 \bullet (\overline{T_1} + T_2 + T_4 + T_3) \\
&= W_1\overline{T_1}T_3 + W_2 \bullet MOV1 \bullet \overline{T_1}T_2 + W_2 \bullet MOV2 \bullet \overline{T_1}T_2 + W_2 \bullet ADD \bullet \overline{T_1} + W_2 \bullet MOV3 \bullet \overline{T_1}T_2T_4
\end{aligned}$$

$$\begin{aligned}
S_1 &= W_1(T_1 + T_2 + \overline{T_3}) + W_2 \bullet MOV1 \bullet (T_1 + \overline{T_2} + T_3) + W_2 \bullet MOV2 \bullet (T_1 + \overline{T_2} + T_3) \\
&\quad + W_2 \bullet ADD \bullet \overline{T_1} + W_2 \bullet MOV3 \bullet (T_1 + \overline{T_2} + T_3 + T_4) \\
&= W_1\overline{T_3} + W_2 \bullet MOV1 \bullet \overline{T_2} + W_2 \bullet MOV2 \bullet \overline{T_2} + W_2 \bullet ADD \bullet \overline{T_1} + W_2 \bullet MOV3 \bullet \overline{T_2}
\end{aligned}$$

$$\begin{aligned}
S_0 &= W_1(\overline{T_1} + T_2 + T_3) + W_2 \bullet MOV1 \bullet (\overline{T_1} + T_2 + T_3) + W_2 \bullet MOV2 \bullet (\overline{T_1} + T_2 + T_3) \\
&\quad + W_2 \bullet ADD \bullet T_1 + W_2 \bullet MOV3 \bullet (\overline{T_1} + T_4 + T_2 + T_3) \\
&= W_1\overline{T_1} + W_2 \bullet MOV1 \bullet \overline{T_1} + W_2 \bullet MOV2 \bullet \overline{T_1} + W_2 \bullet ADD \bullet T_1 + W_2 \bullet MOV3 \bullet \overline{T_1} \bullet \overline{T_4}
\end{aligned}$$

$$CN = W_1\overline{T_3} + W_2 \bullet MOV1 \bullet \overline{T_2} + W_2 \bullet MOV2 \bullet \overline{T_2} + W_2 \bullet ADD \bullet T_1 + W_2 \bullet MOV3 \bullet \overline{T_2}$$

❧ 根据逻辑表达式，利用基本的与门、或门、非门实现控制信号即可。

❧ 该控制信号的实现，结合内核的数据通路，设计完整的电路图，下载到FPGA芯片中；

❧ 汇编程序以十六进制代码从0单元开始写入RAM的相应单元中。

❖ 统调

❖ 具体步骤如下：

❧ 1、按复位键RET

❖ 使MAR清0、指令计数器PC清0，保证从存储器0号单元取指令。

❧ 2、按单脉冲键，启动程序执行。

❖ 要求：实现更多的指令，如减法、逻辑运算等。