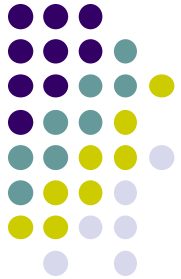


计算机组成与系统结构

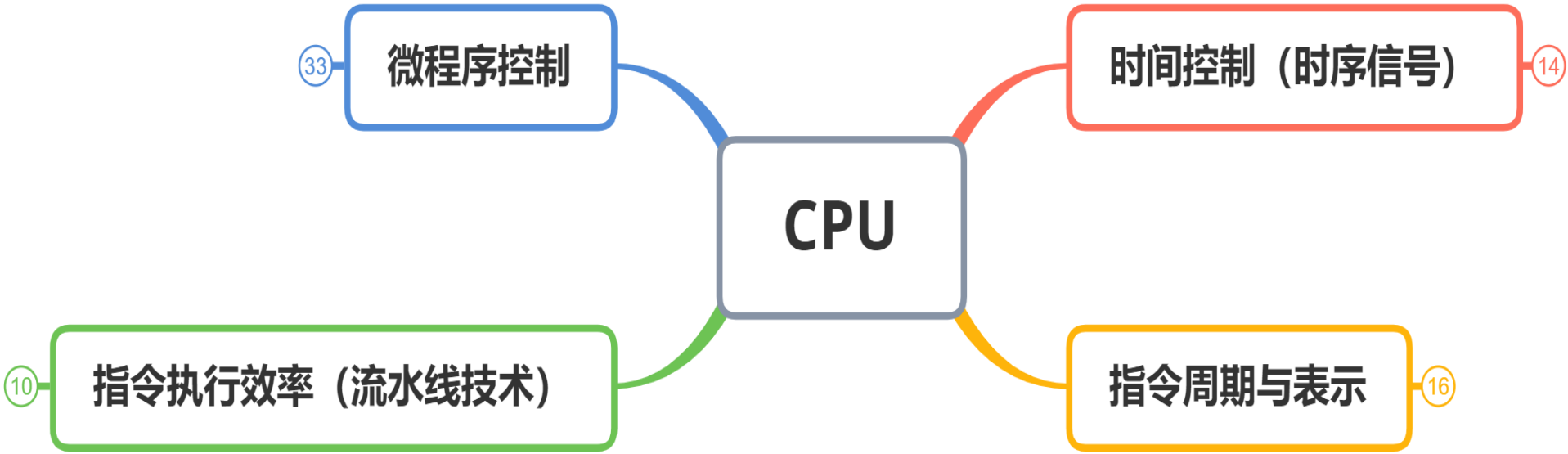
第五章 中央处理机 (3)

吕昕晨

lvxinchen@bupt.edu.cn

网络空间安全学院

中央处理机



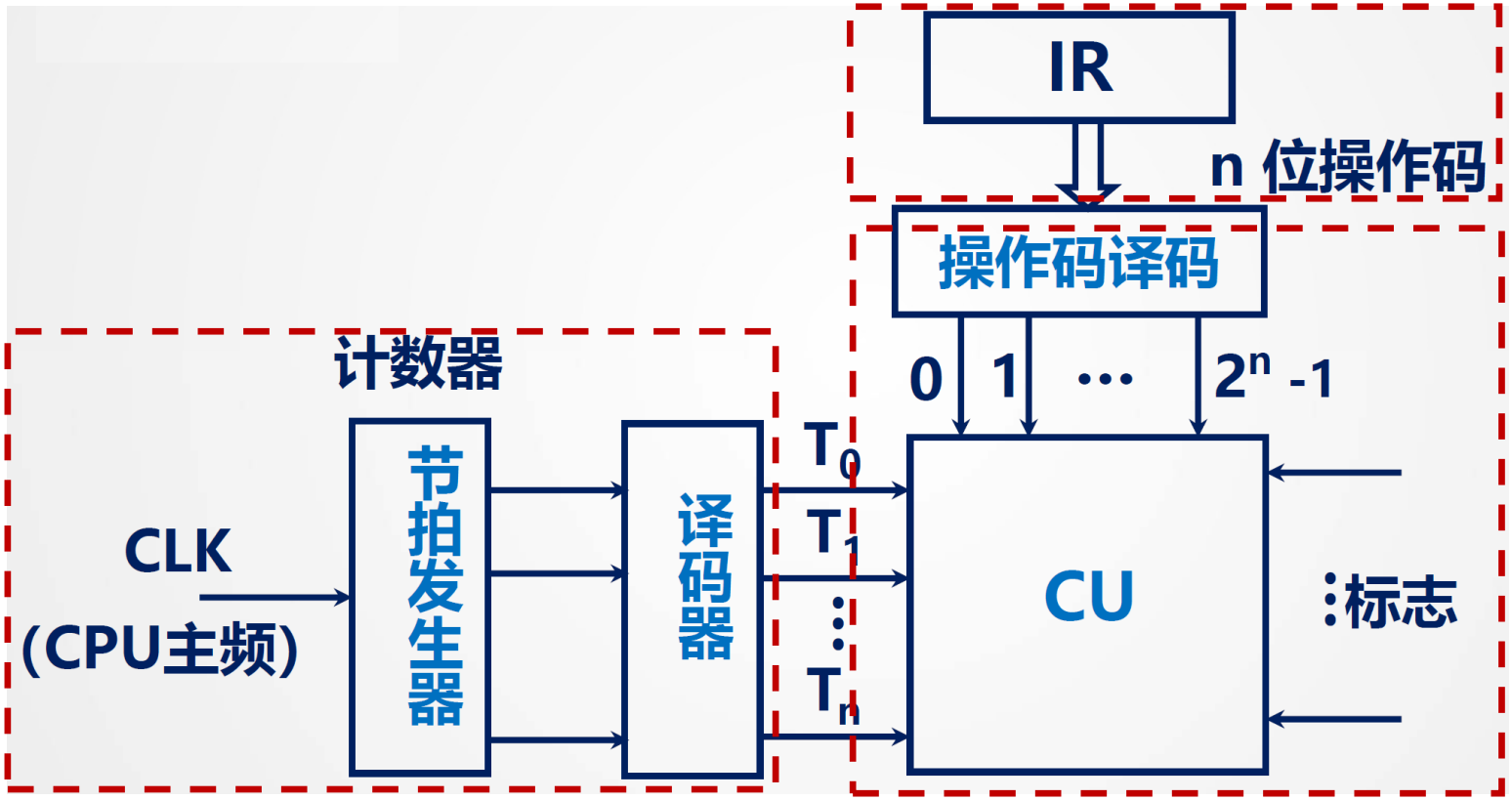
回顾：CPU的功能总结

- CPU：中央处理器
  - 根据编写的程序，自动从存储器中取出指令，并完成指令操作
- 指令操作——复杂的控制过程
  - 核心：多个操作信号在指定时间完成对应控制
  - 关键：描述时间、事件
  - **操作控制**（一条指令有若干操作信号实现）
  - **时间控制**（指令各个操作实施时间的定时）

如何控制

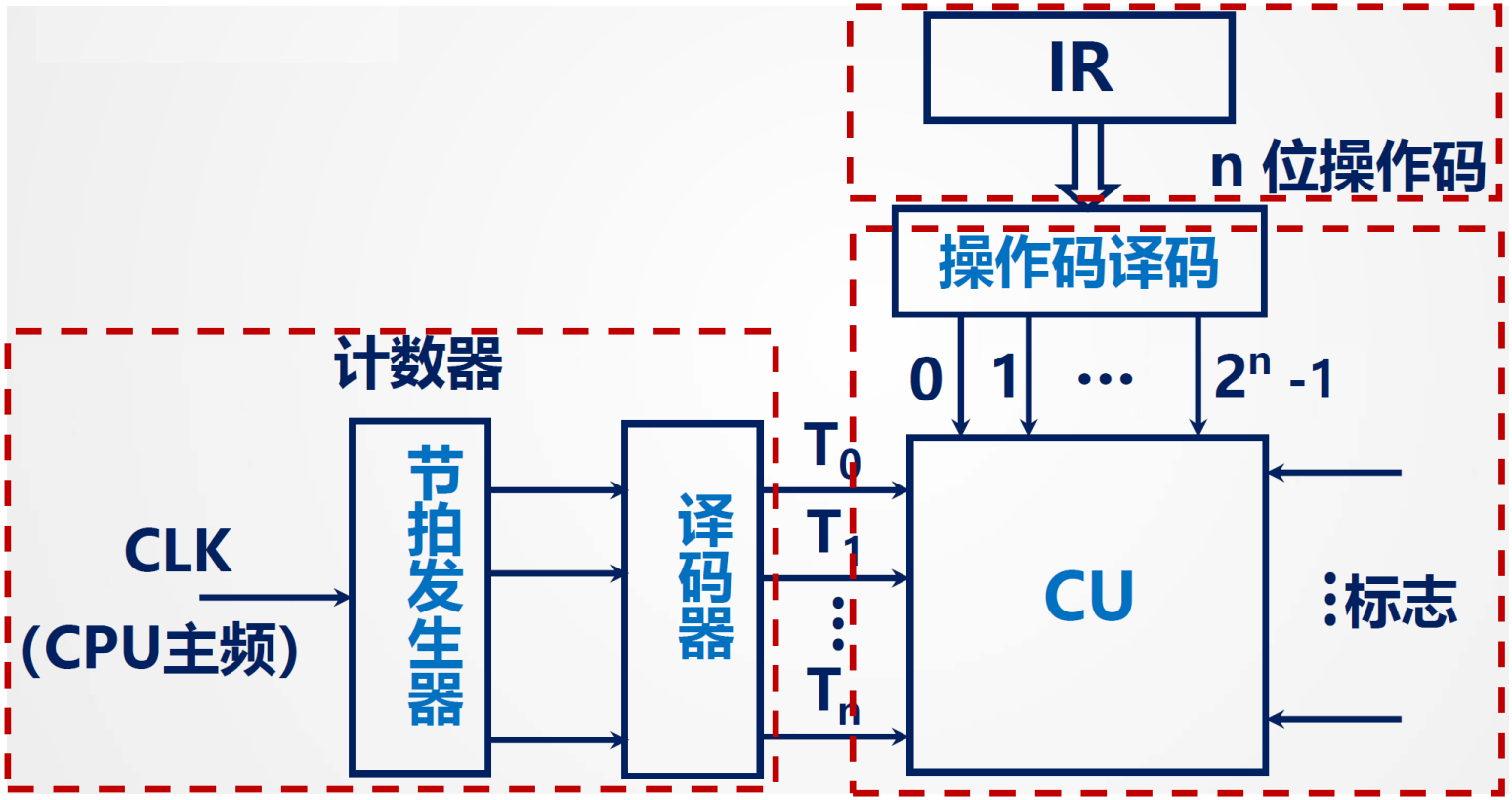
取指令 —操作控制、时间控制→ 执行指令

控制器：时间控制+操作控制



Chp5-1：时间控制　　Chp5-2：操作控制（方框图）

控制器：时间控制+操作控制



Chp5-1：时间控制

T0 – Tn（脉冲）

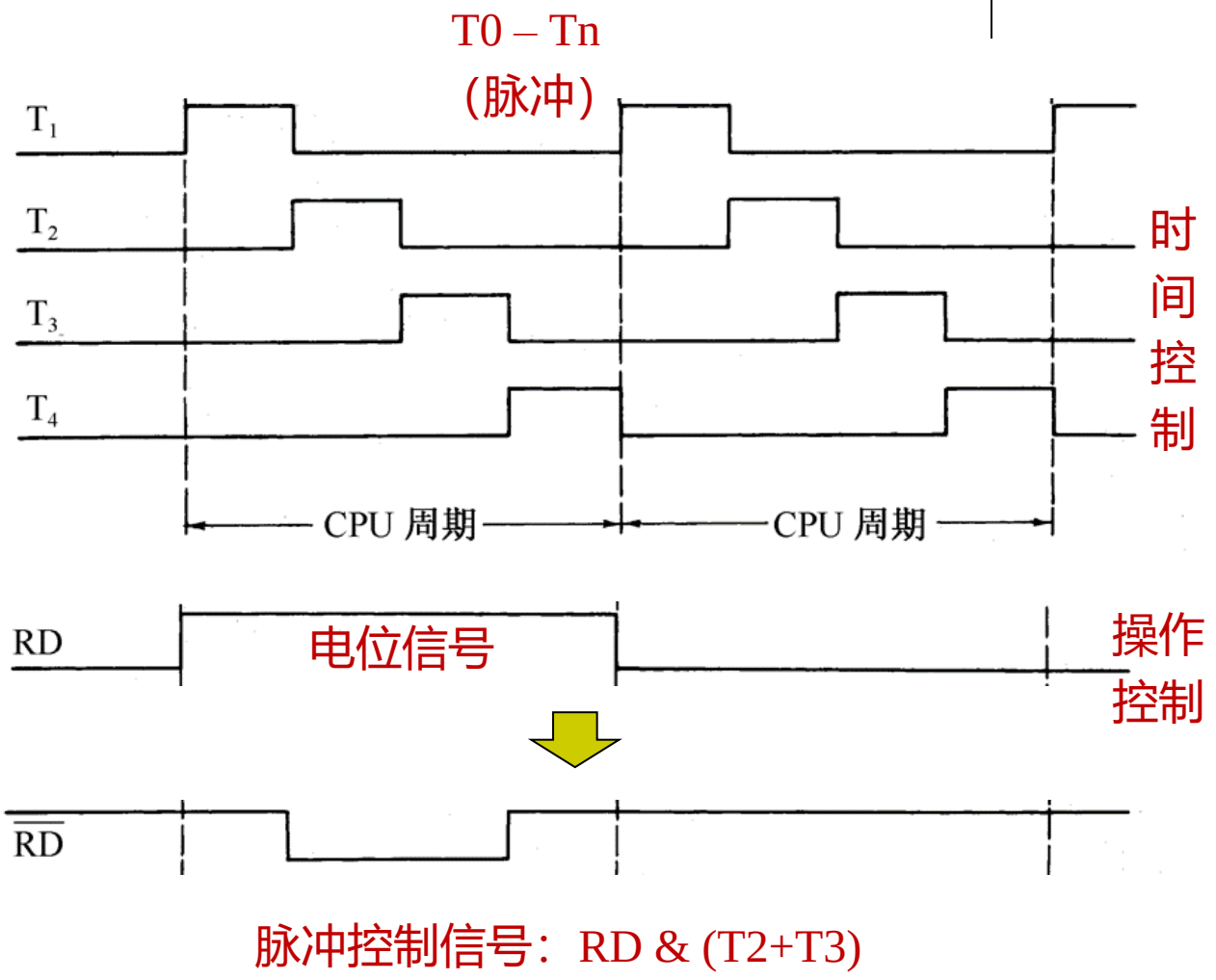
Chp5-2：操作控制（方框图）

电位信号（CPU周期）

控制器：时间控制+操作控制

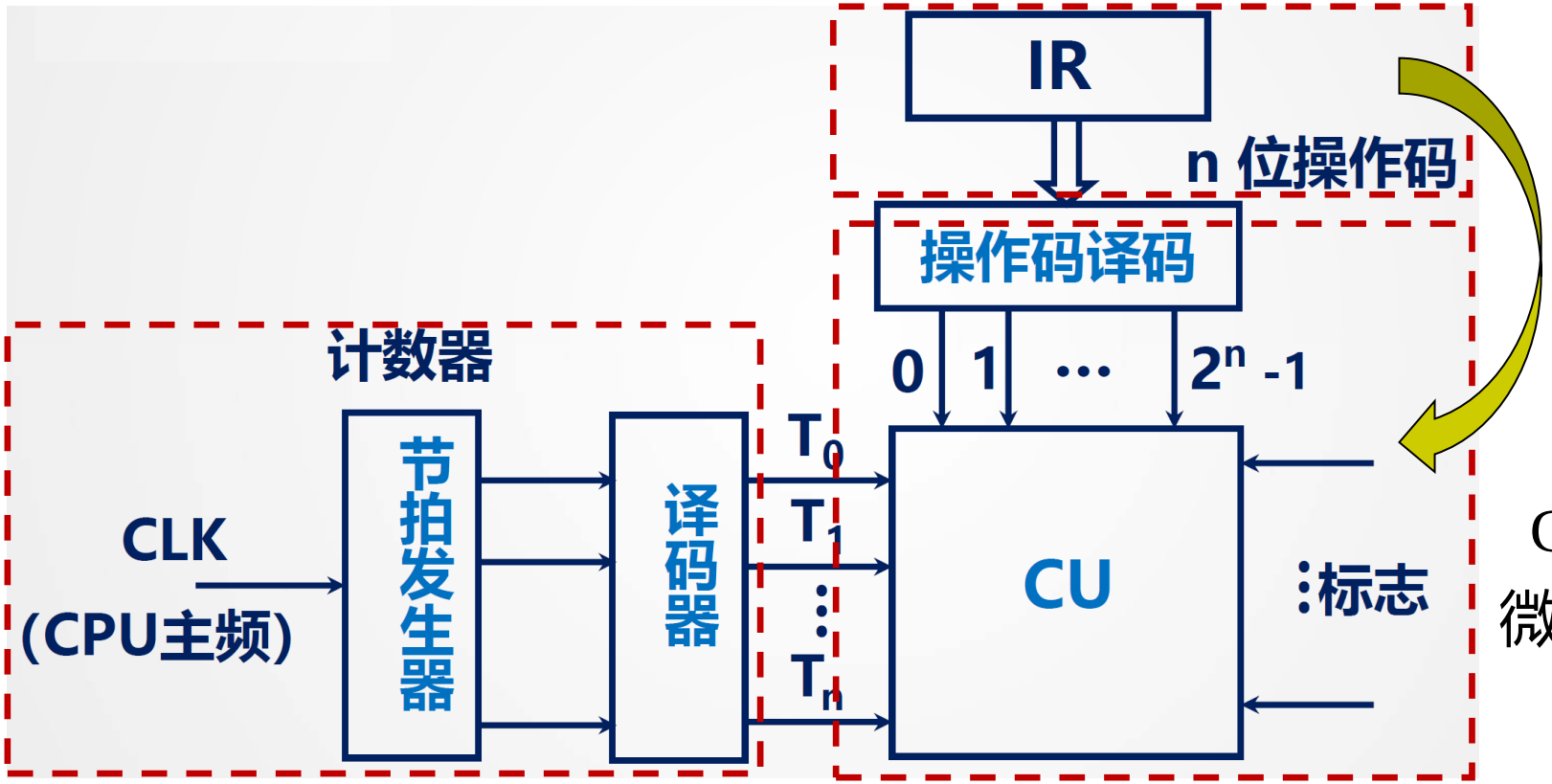
- 节拍脉冲
  - CNT：0~3
  - T1 — T4

- 读写控制信号
  - 读写控制功能
  - RD（电位信号）
  - RD非（生成脉冲控制）



脉冲控制信号：RD & (T2+T3)

微程序控制



如何将机器指令翻译成微操作?

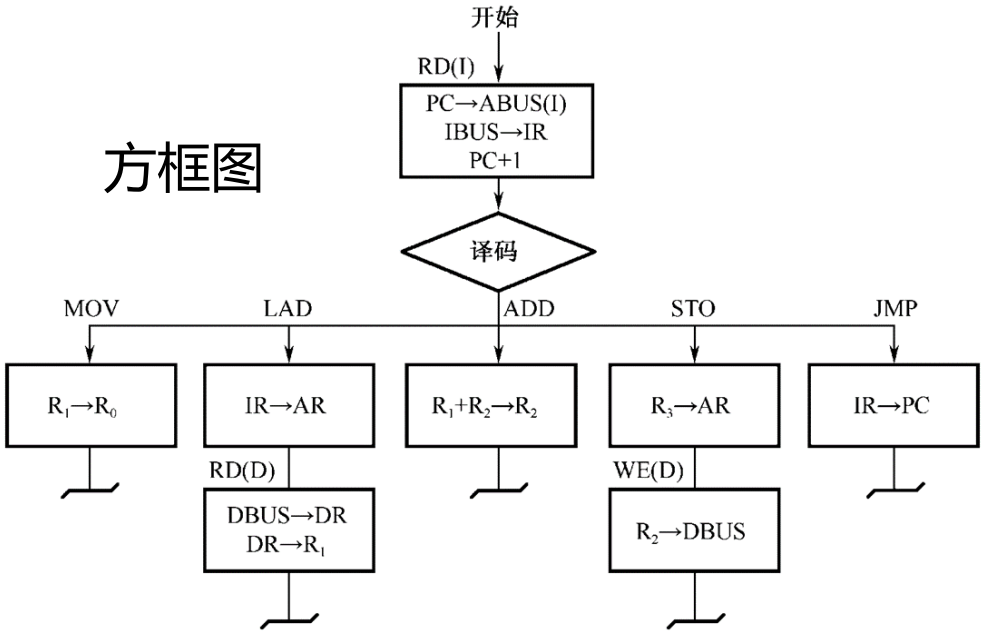
Chp5-3：微程序控制

Chp5-1：时间控制

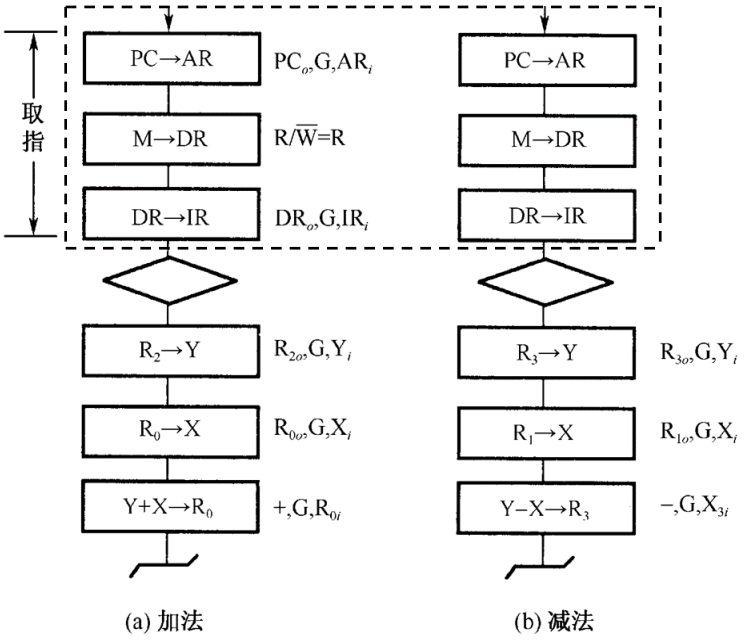
Chp5-2：操作控制（方框图）

微程序控制：思路

方框图

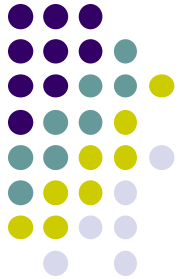


指令周期流程图



- 问题：如何存储/翻译方框图内容
  - 表示：指令→多个CPU周期内容（微程序/微指令）
  - 存储：依照指令的方式，用（微/控制）存储器存储微指令
  - 翻译：微指令译码（结合T周期），生成控制信号

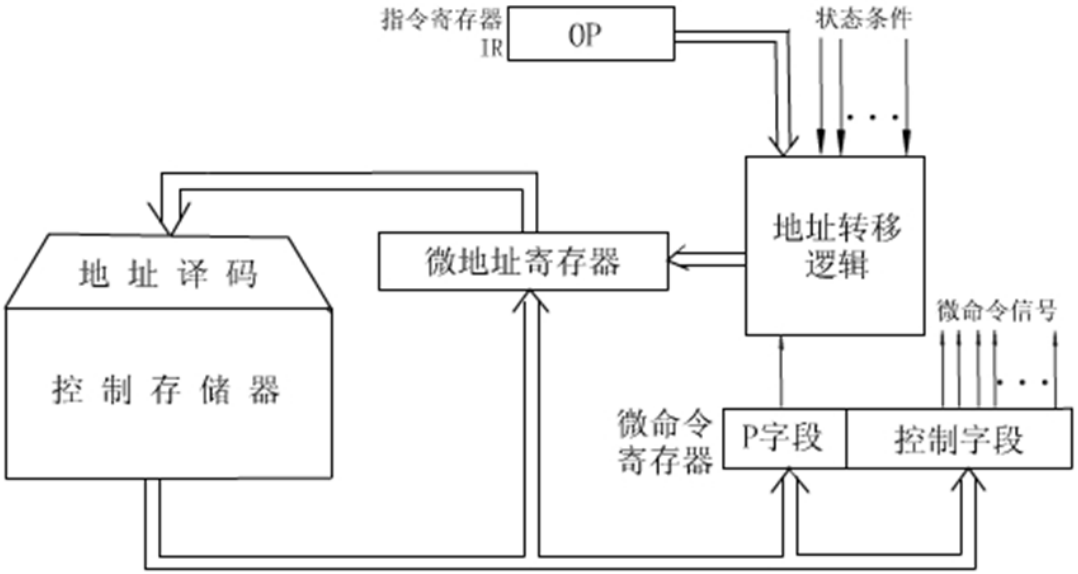
第五章 中央处理机



- 微程序/硬件布线控制器简介
- 微程序控制原理
- 微程序控制技术
- 微程序控制设计

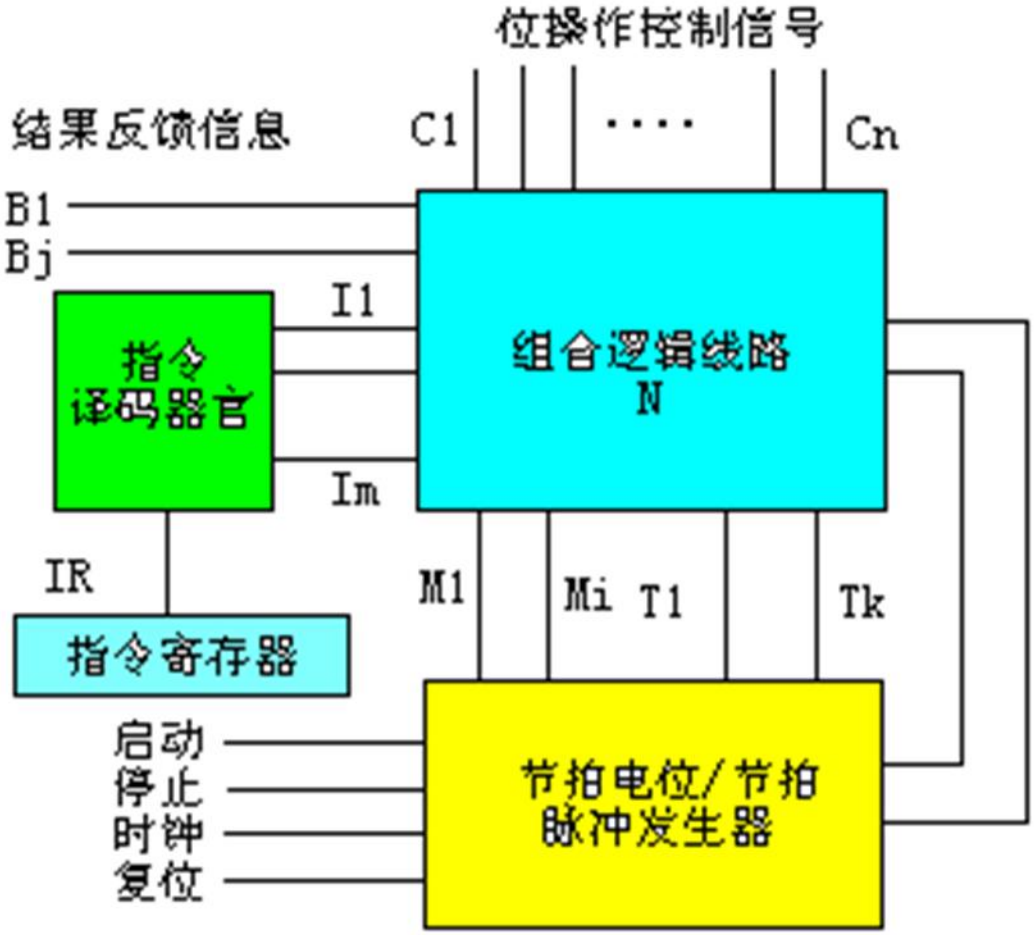
微程序控制器

- 基本思想与实现方法
  - 把操作控制信号编制成微指令，存放到控制存储器里，运行时，从控存中取出微指令，产生指令运行所需的操作控制信号
  - 微程序设计技术是用软件方法来设计硬件的技术



硬件布线控制器

- 基本思想
  - 通过逻辑电路直接连线而产生的，又称为组合逻辑控制方式
- 设计目标
  - 使用最少元件（复杂的树形网络）
  - 速度最高
  - FPGA实验



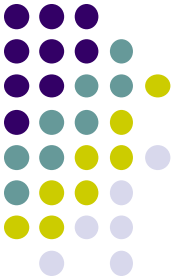
微操作控制信号产生

- 微程序控制器
  - 微操作控制信号由微指令产生，并且可以重复使用
  - 优点：灵活、设计方便
- 硬件布线控制器
  - 某一微操作控制信号由布尔代数表达式描述的输出函数产生
  - 树形网络结构复杂，设计困难，但适合VLSI实现
  - 设计微操作控制信号流程
    - 根据所有机器指令流程图，寻找出产生同一个微操作信号的所有条件，并与适当的节拍电位和节拍脉冲组合
    - 从而写出其布尔代数表达式并进行简化
    - 用门电路或可编程器件来实现

硬布线与微程序控制方式对比

类别 / 对比项目	微程序控制器	硬布线控制器
工作原理	微操作控制信号以微程序的形式存放在控制存储器中，执行指令时读出即可	微操作控制信号由组合逻辑电路根据当前的指令码、状态和时序，即时产生
执行速度	慢	快
规整性	较规整	烦琐、不规整
应用场合	CISC CPU	RISC CPU
易扩充性	易扩充修改	困难

第五章 中央处理机



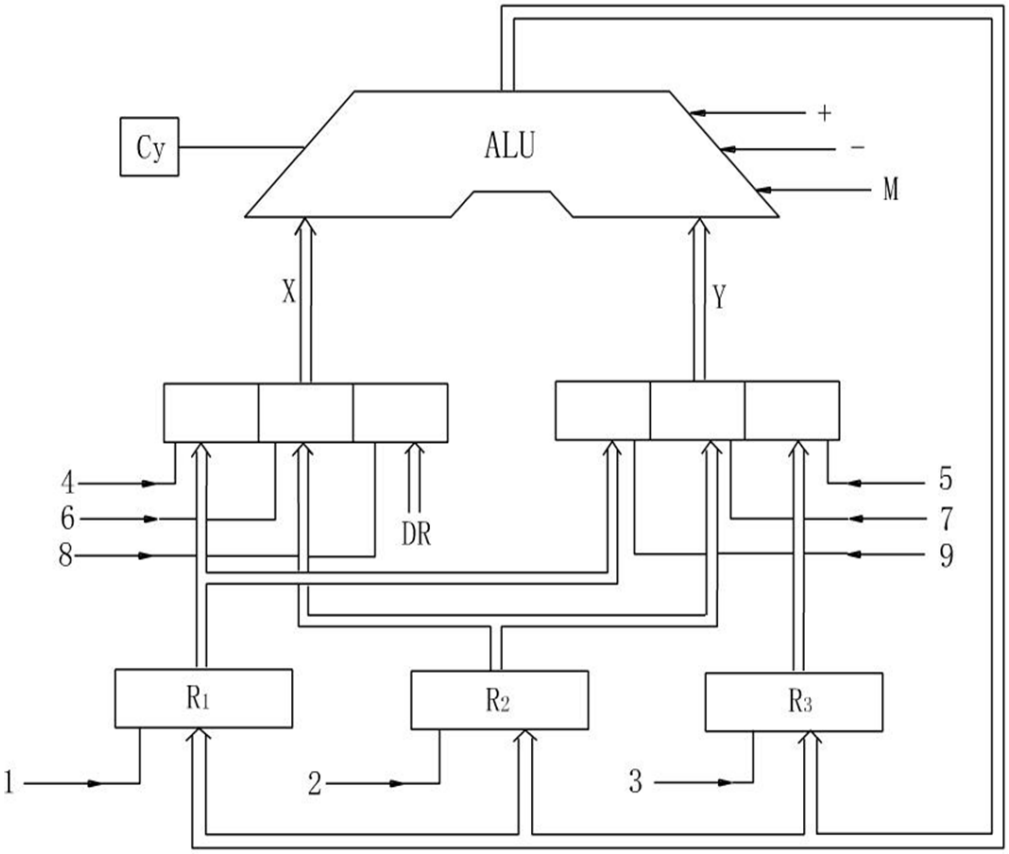
- 微程序/硬件布线控制器简介
- 微程序控制原理
  - 微命令/微操作/微指令/微程序
  - 微程序控制原理与示例
- 微程序控制技术
- 微程序控制设计

微命令和微操作

- **微命令**
  - 控制部件向执行部件发出的各种控制命令
  - 构成控制序列的最小单位
  - 例如：打开或关闭某个控制门的电位信号、某个寄存器的打入脉冲等
- **微操作**：微命令的操作过程
  - 微命令和微操作是一一对应的
  - 关系：微命令是微操作的控制信号，微操作是微命令的操作过程
  - 微操作是执行部件中最基本的操作

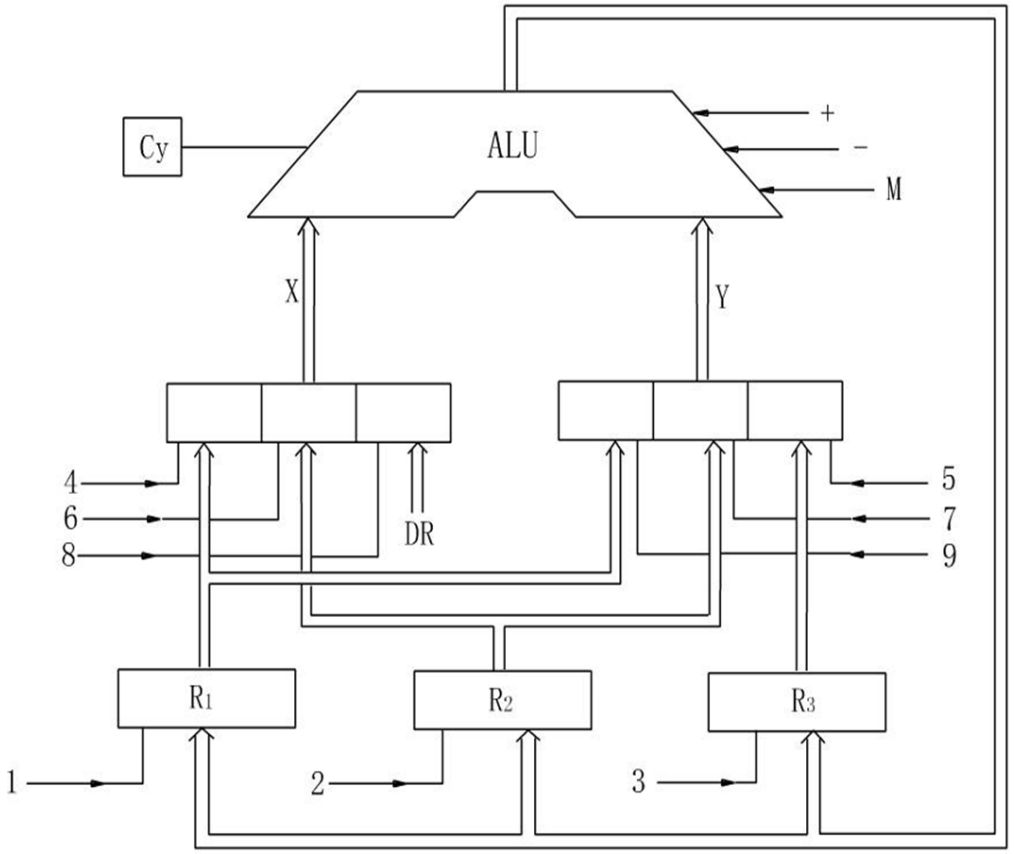
微操作相容、互斥性质

- 由于数据通路/操作对象的结构关系，微操作可分为：
  - **互斥**微操作
    - **不能同时**或不能在同一个CPU周期内并行执行的微操作
  - **相容**微操作
    - 是指**能够同时**或在同一个CPU周期内并行执行的微操作



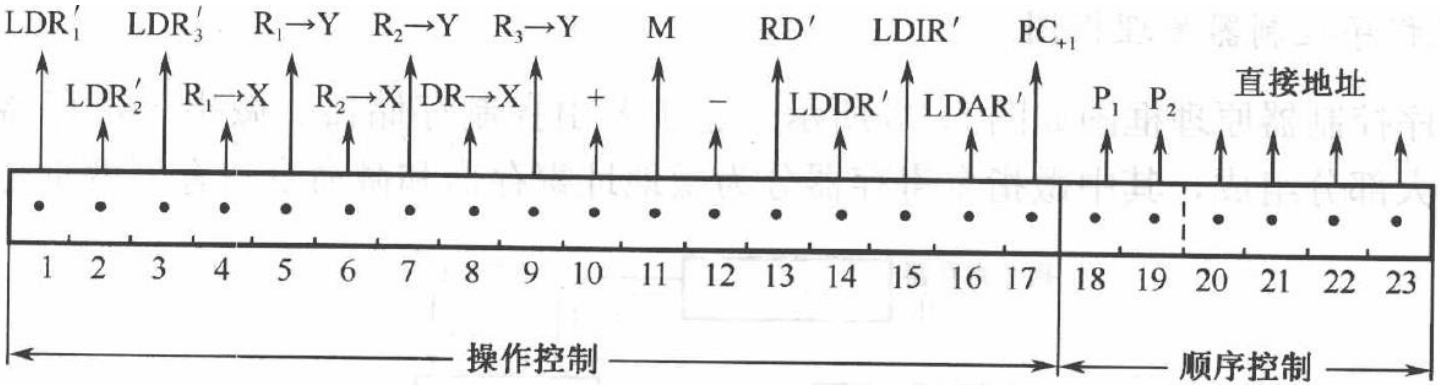
微操作相容、互斥性质

- 互斥微操作
  - ALU控制信号
    - +、-、M（传送）
  - ALU的X/Y输入端
    - X：4、6、8
- 相容微操作
  - 寄存器R1~R3
    - 1、2、3
  - X端与Y端
    - 4、7（任意组合）



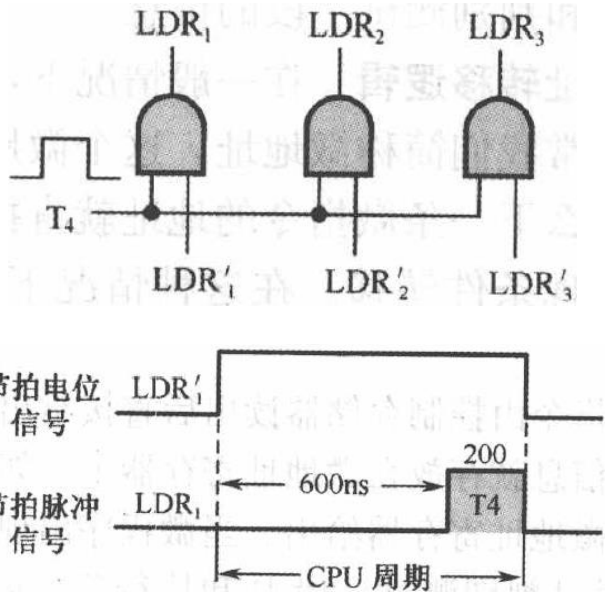
微指令结构

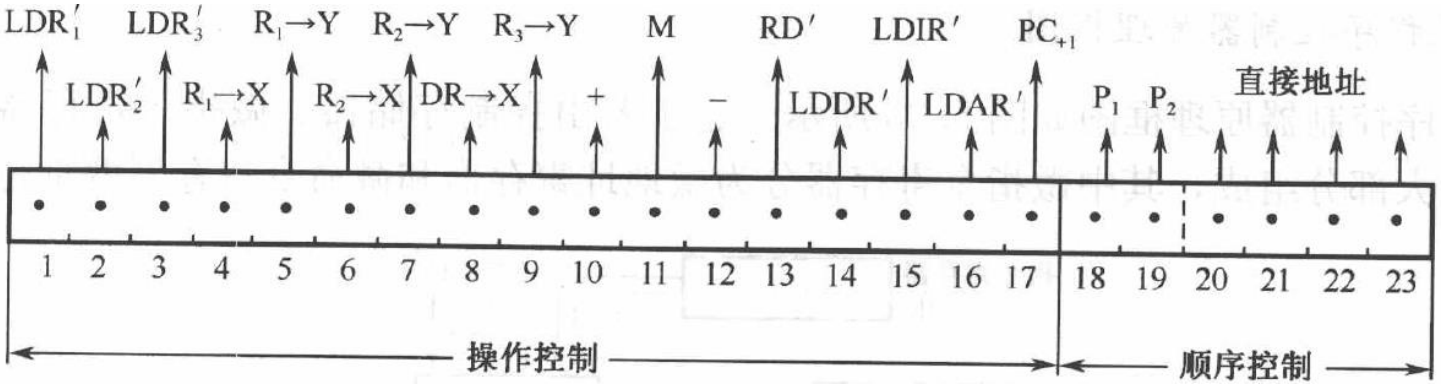
- **微指令**
  - 一个CPU周期可执行多条微操作（相容微操作）
  - 这些微操作控制信息，存储在控制存储器里，就是微指令
  - 微命令的组合，微指令存储在控制器的控制存储器
- 组成
  - 操作控制字段（1~17位）
  - 顺序控制字段（18~23位）



微指令——操作/顺序控制字段

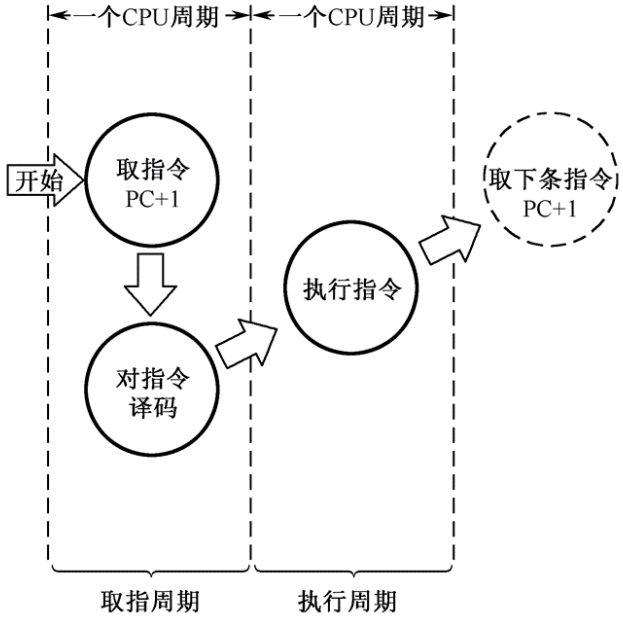
- **操作控制字段**，又称微操作码字段
  - 产生各个微操作控制信号
    - 某位为1，表明发微指令
    - 微指令发出的控制信号都是节拍电位信号，持续时间为一个CPU周期
    - 引入节拍脉冲信号作时间控制
- **顺序控制字段**，又称微地址码字段
  - 控制产生下一条要执行的微指令地址





微程序

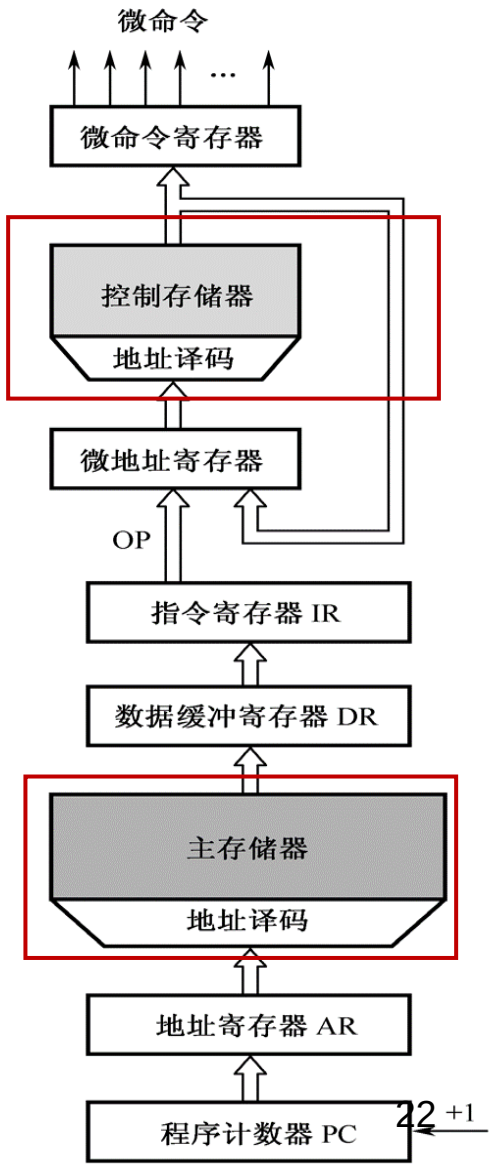
- 机器指令包含多个CPU周期
  - 一个CPU周期微操作对应一条微指令
  - 一条机器指令包含多条微指令
- **微程序**
  - 一系列微指令的有序集合
  - 一段微程序对应一条机器指令



机器指令与微指令的关系

机器指令与微指令的关系

- **机器指令对应微程序**
  - 微程序由一系列微指令组成
  - 例如，BCD加法
- 存储位置与**地址**
  - 机器指令存储在**主存储器**，对应PC
  - 微指令存储在**控制存储器**，对应微地址
- 作用
  - 机器指令明确指令功能
  - 微指令明确微操作信号



相关概念总结（重要）

- 指令系统：所有机器指令的集合（第四章）
- 1条机器指令（第四章）：1个微程序（第五章）
- 微程序：若干条微指令的组合（指令周期）
- 微指令：一系列微命令（CPU周期）
- 微命令与微操作（一一对应、T周期）
- 程序执行流程涉及：
  - 程序→机器指令→微程序
    →微指令→微命令（微操作）

单选题 1分

设置

下列描述正确的是

A 若干条微指令实现一条机器指令，一条微指令包含多个微命令

B 若干条机器指令实现一条微指令，一条微命令包含多个机器指令。

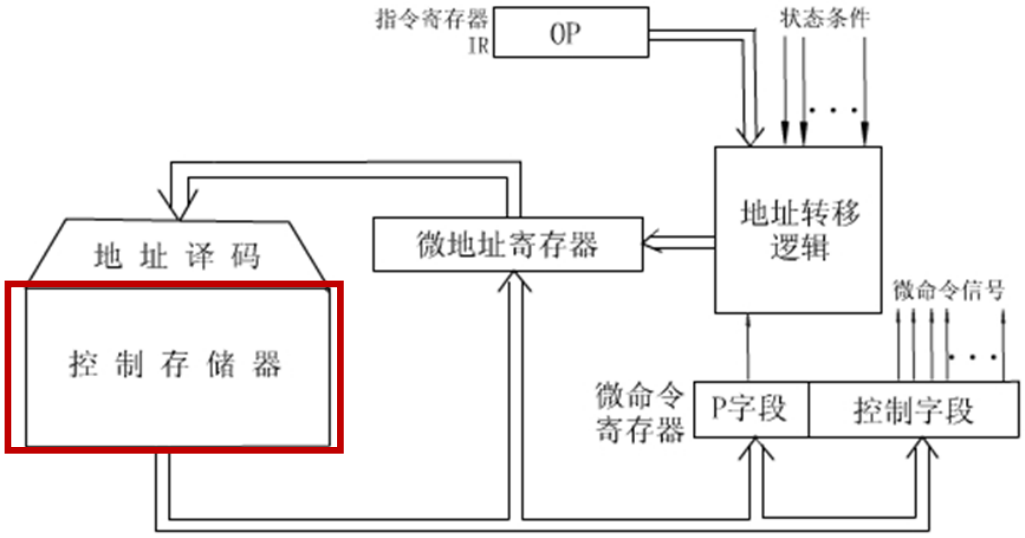
C 一条微指令实现一条机器指令，一条微指令包含多个微命令。

D 若干个微命令实现一条机器指令，一个微命令包含多条微指令。

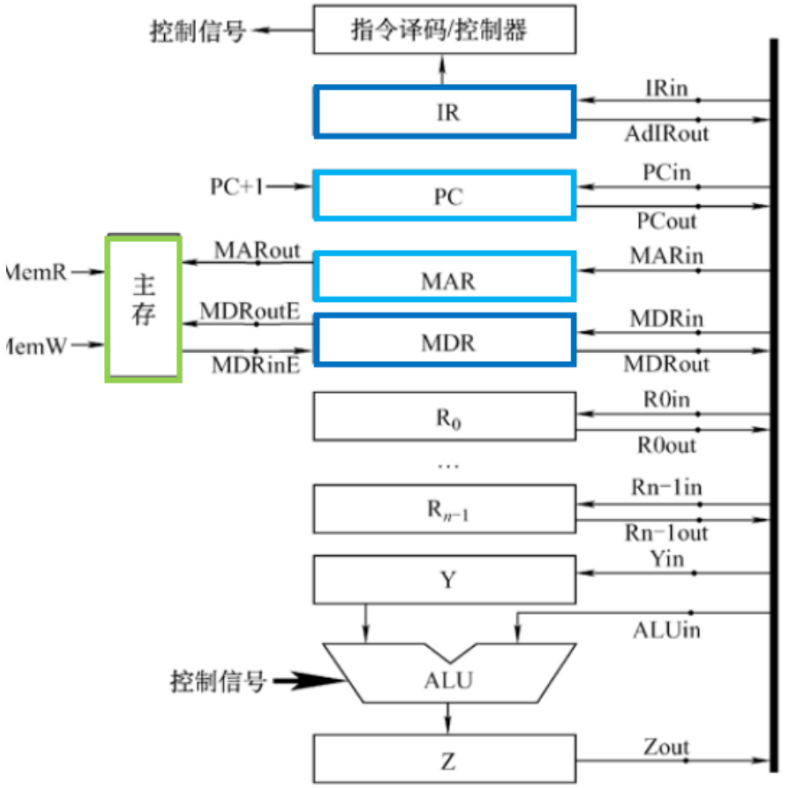
第五章 中央处理机

- 微程序/硬件布线控制器简介
- 微程序控制原理
  - 微命令/微操作/微指令/微程序
  - 微程序控制原理与示例
- 微程序控制技术
- 微程序控制设计

微程序控制原理对比



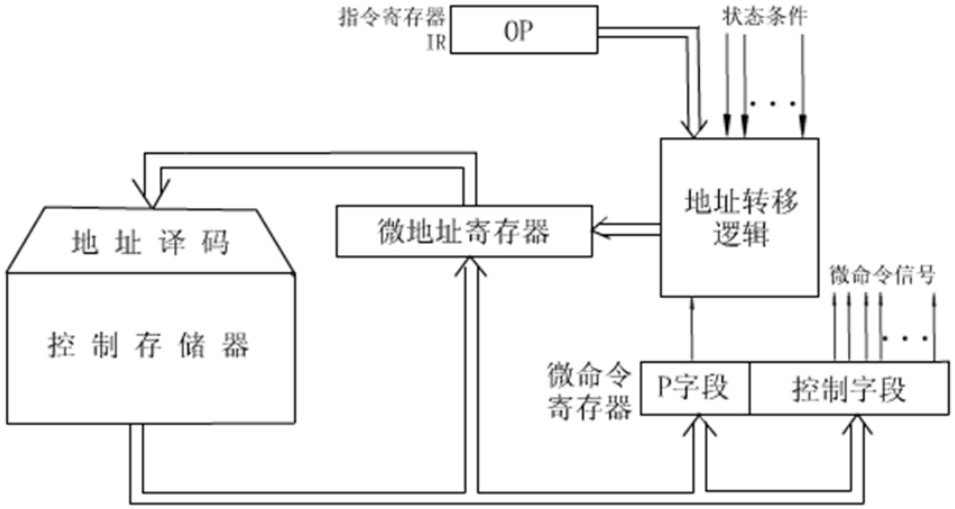
微程序控制原理框图



数据通路图

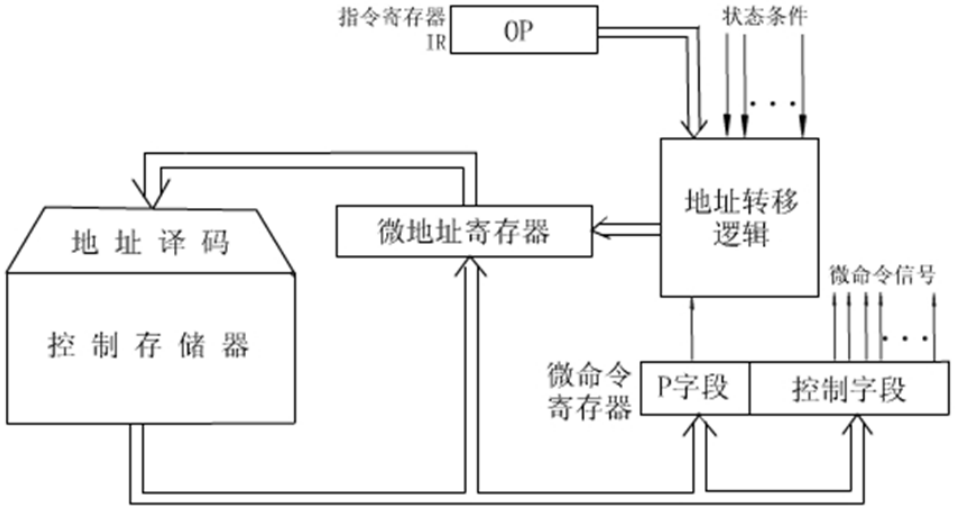
微程序控制原理框图

- **控制存储器(μCM)**
  - 微程序控制器的**核心部件**，用来存放微程序
  - 性能(容量、速度、可靠性等)与计算机性能密切相关
- 微指令寄存器(μIR)
  - 用来存放从μCM取出的正在执行的微指令
  - 位数同微指令字长相等

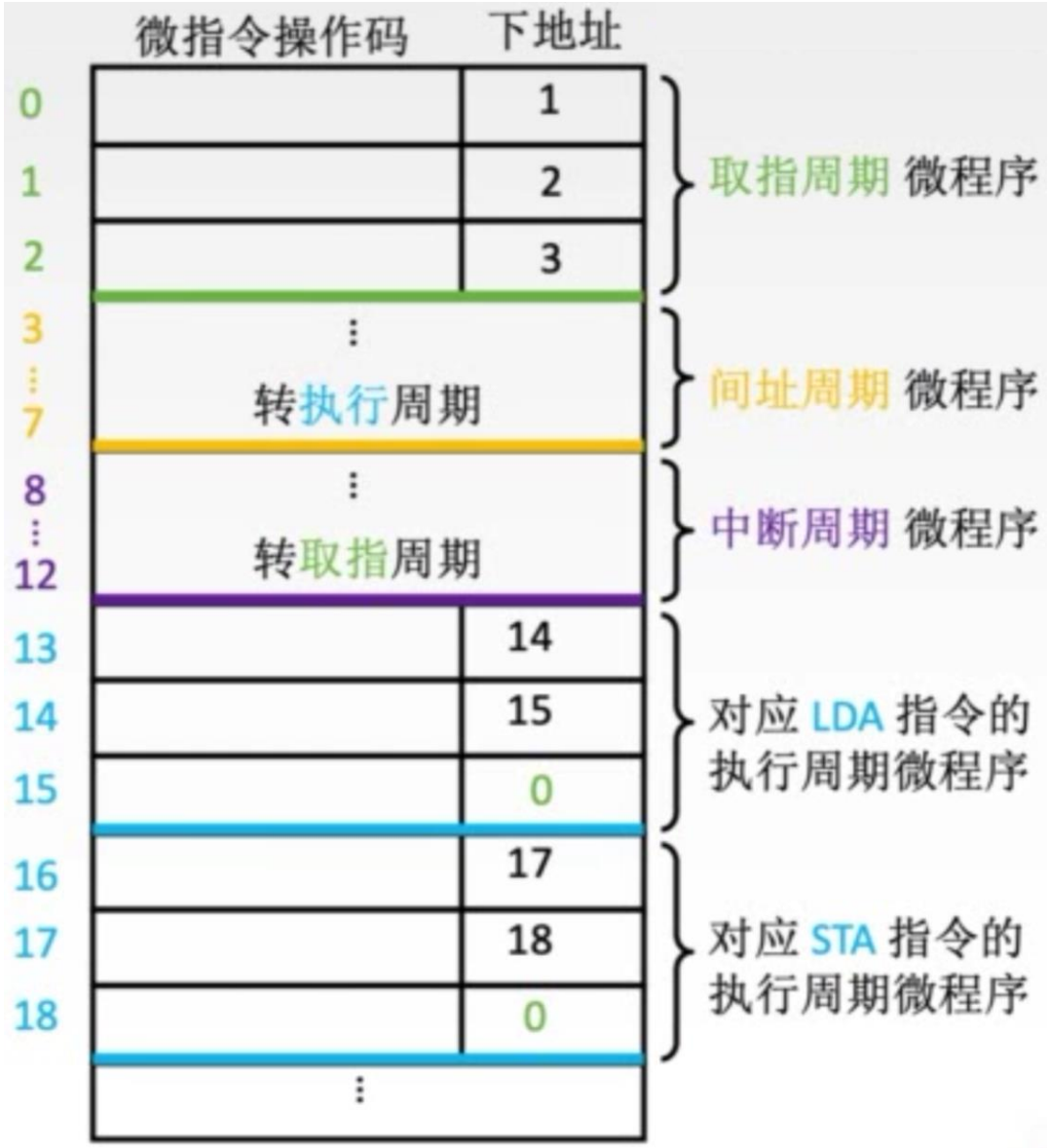


微程序控制原理框图

- 微地址形成部件
  - 用来产生初始微地址和后继微地址
  - 保证微指令的连续执行
- 微地址寄存器(μMAR/ μAR/μPC)
  - 接受微地址形成部件送来的微地址
  - 为下一步从μCM中读取微指令作准备



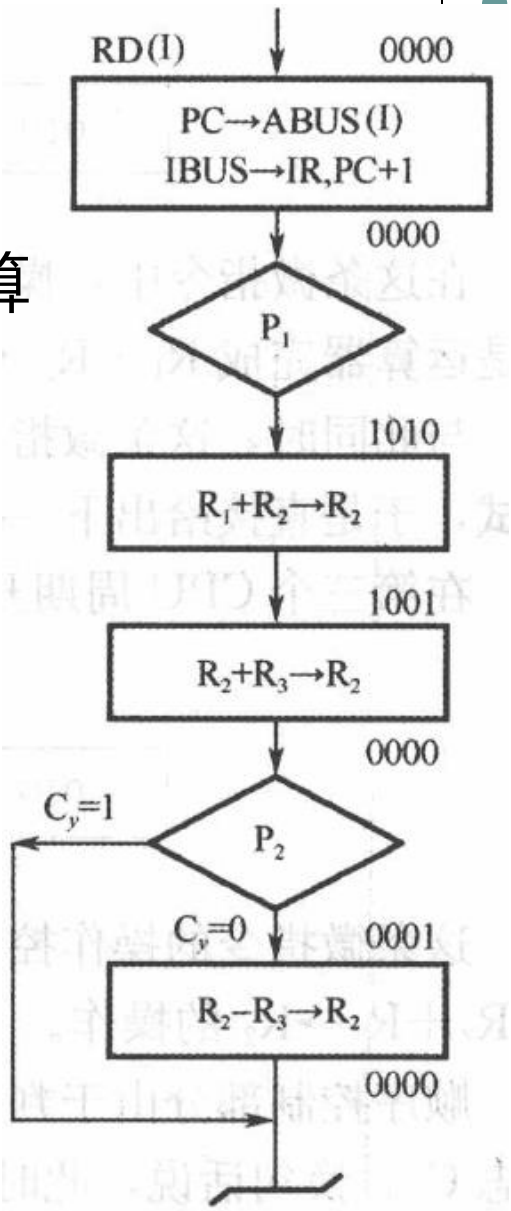
微程序控制——uCM组织



- 公共部分
  - 取指周期
- 执行部分
  - 对应指令功
  - 末尾转下地址0
  - 循环执行

微程序控制示例

- 十进制加法指令（一条指令）
  - 功能：用BCD码完成十进制加法运算
  - 流程框图（R3=6）
    - 取指、译码、正常加法
    - 加6修正
    - 若产生进位
      - 保持不变
    - 若未产生进位
      - 退回，减6

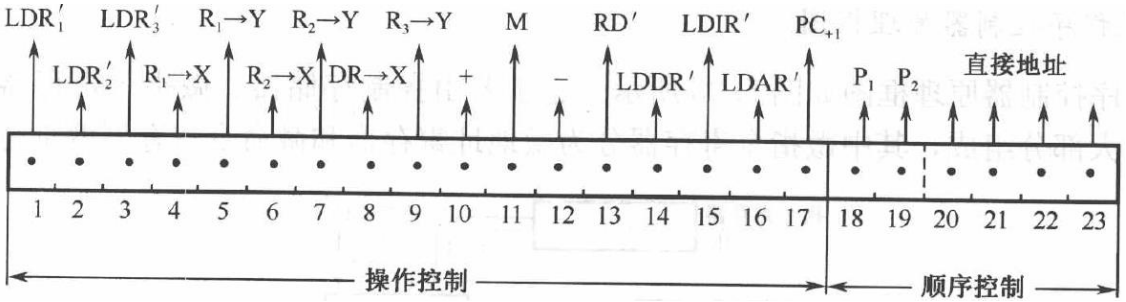


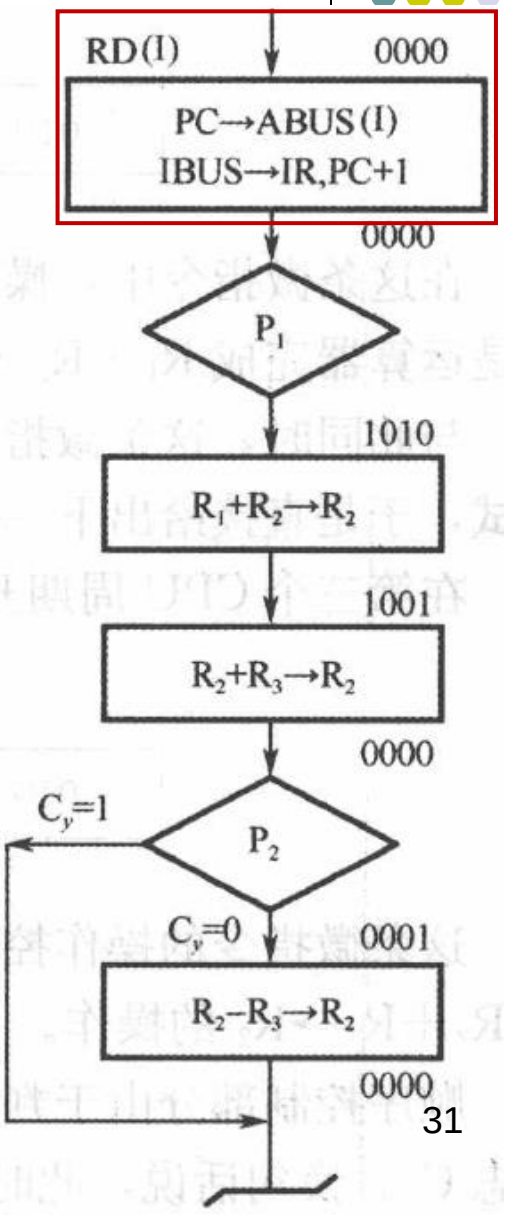
微程序控制示例

- 取指令——微指令编码

000	000	000	000	10111	10	0000

- 第13位：RD（I），I-Cache读操作
- 第15位：LDIR，指令放入IR
- 第16位：LDAR，PC送I-Cache
- 第17位：PC+1，自增
- 第18位，P1测试，用IR寄存器OP段作为地址





微程序控制示例

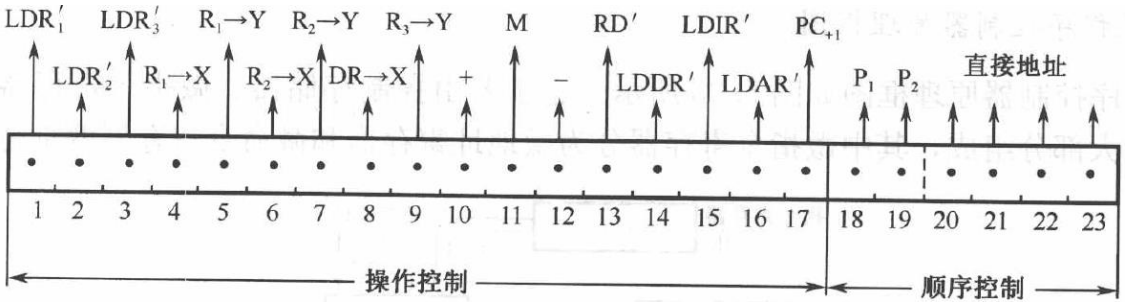
- 加法指令——微指令编码

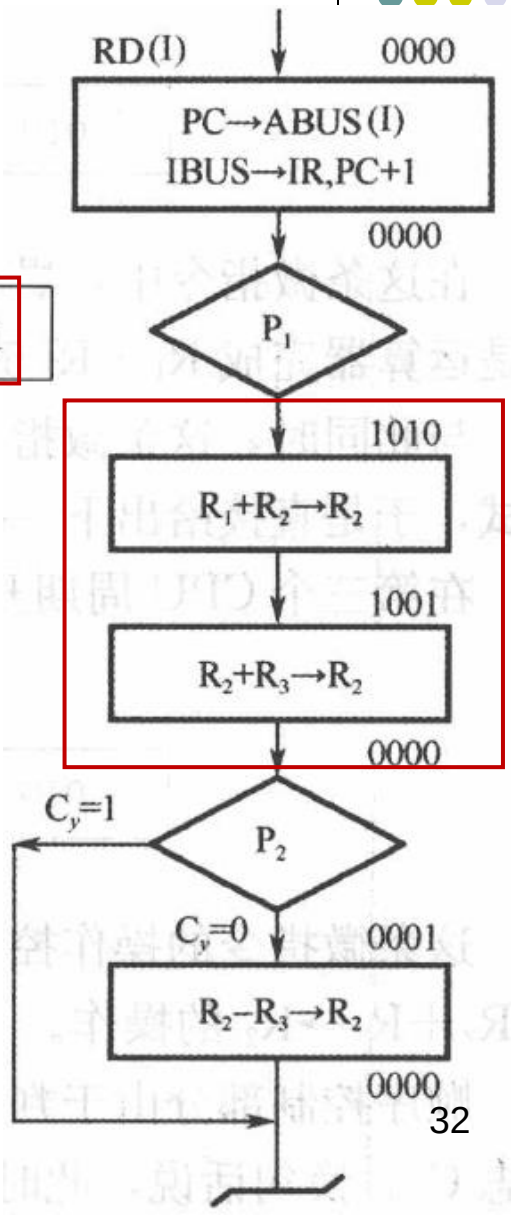
010	100	100	100	00000	00	1001

- R1+R2→R2，下一条微地址1001

010	001	001	100	00000	01	0000

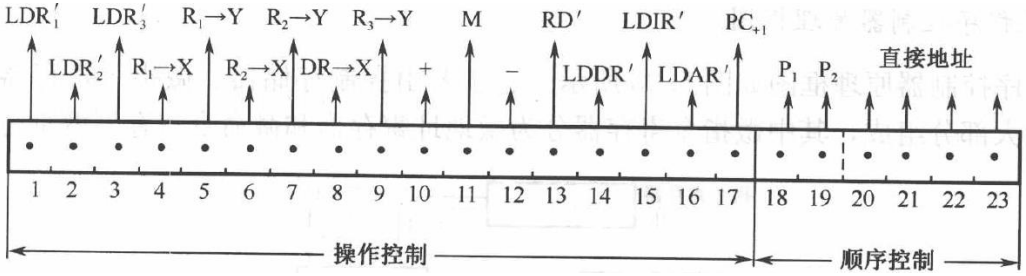
- R2+R3→R2
- LDR2、R2→X、R3→Y、+
- 执行P2测试，测试进位标志Cy





第五章 中央处理机

- 微程序/硬件布线控制器简介
- 微程序控制原理
- 微程序控制技术
  - 微指令结构（操作控制字段）
  - 微地址形成（顺序控制字段）
  - 微指令格式
- 微程序控制设计

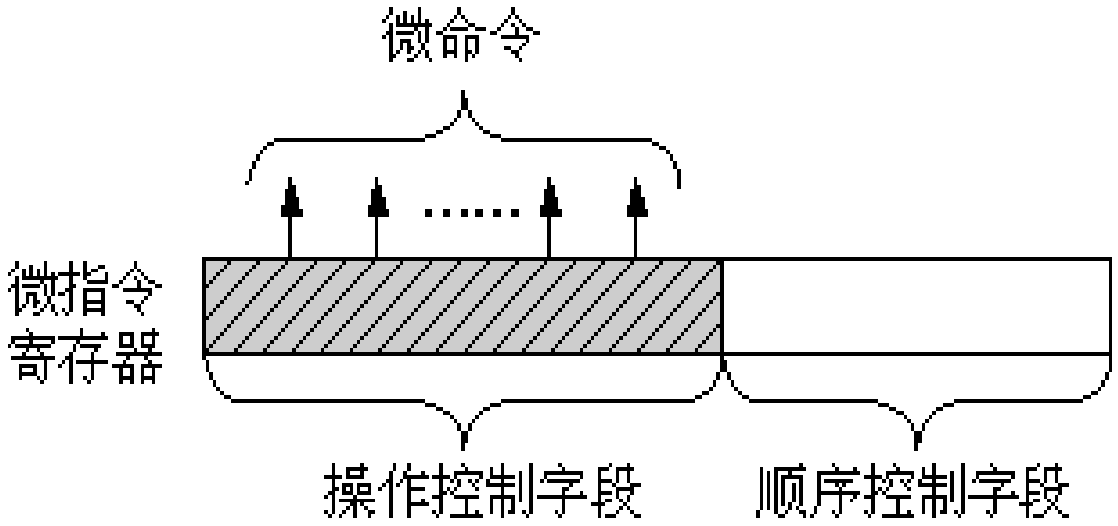


微程序设计——微指令结构

- 设计微指令应当追求的目标
  - 有利于缩短微指令的**长度**
  - 有利于缩小**控制存储器的容量**
  - 有利于提高微程序的执行**速度**
  - 有利于提高微程序设计的**灵活性**
- 微命令编码
  - 直接表示法
  - 编码表示法
  - 混合表示法

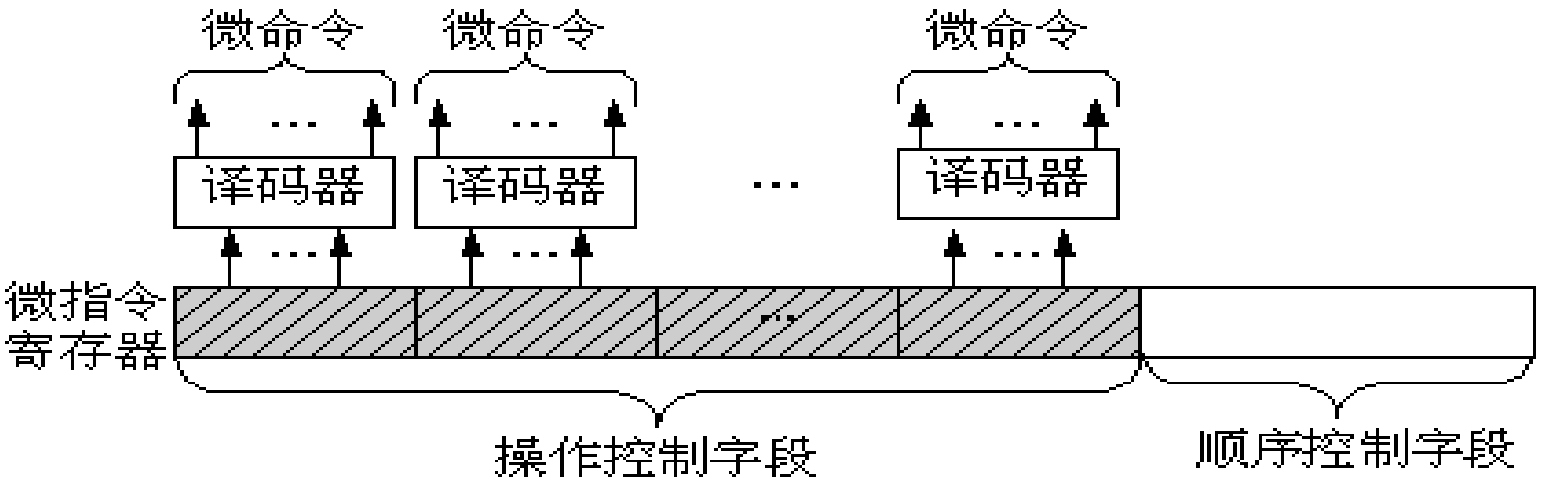
微命令编码——直接表示法

- **直接表示法**
  - 操作控制字段中的各位分别可以直接控制计算机，不需要进行译码
  - 优点：简单直观，输出直接进行控制
  - 缺点：微指令字较长，控制存储器容量大



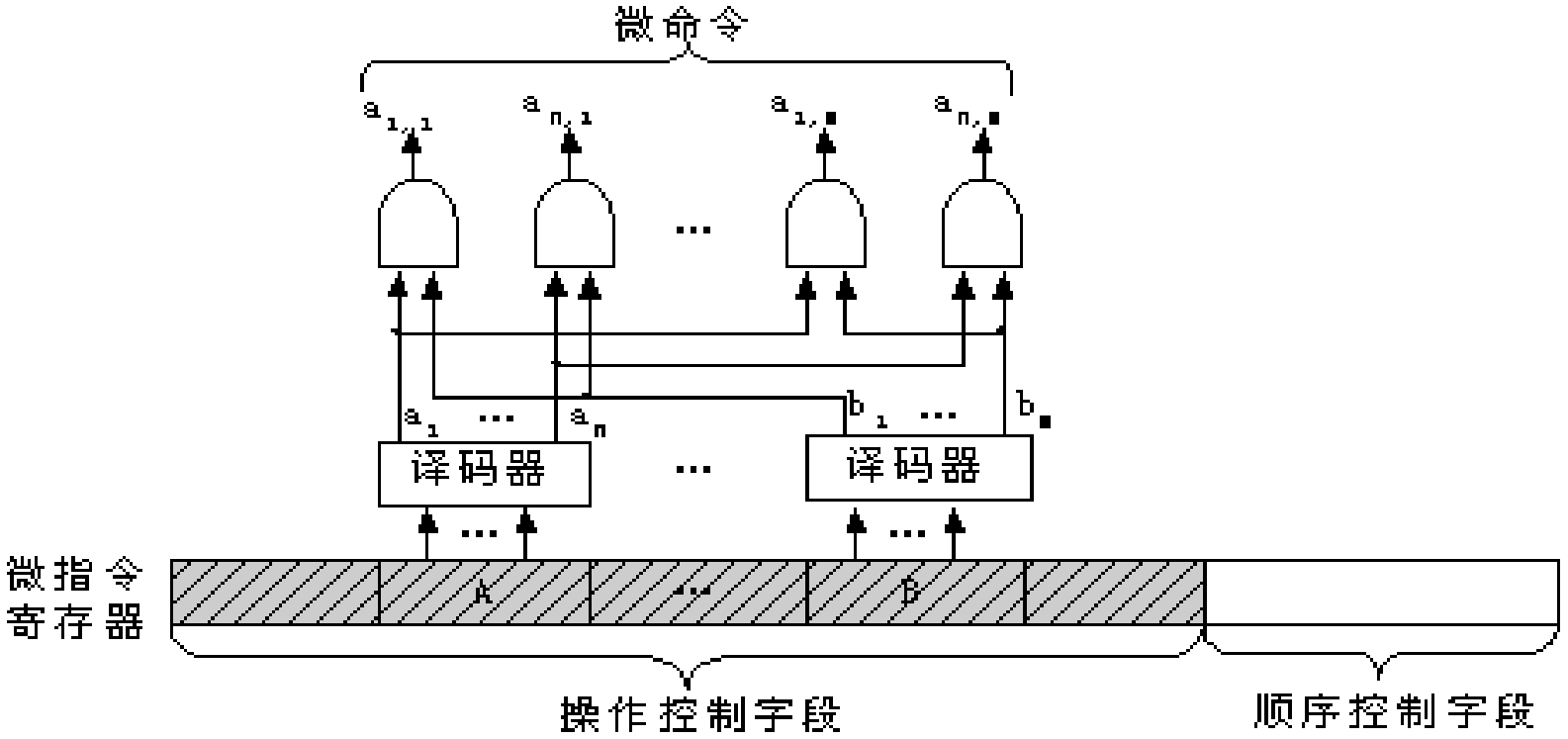
微命令编码——编码表示法

- **编码表示法**
  - 将操作控制字段分为若干个小段（段内互斥），每段内采用最短编码法，段与段之间采用直接控制法
  - 优点：利用互斥性，缩短微指令字长
  - 缺点：额外译码电路，执行速度较慢



微命令编码——混合表示法

- **混合表示法**
  - 结合直接表示法与编码表示法
  - 综合指令字长、灵活性、执行速度等方面要求



单选题 1分

设置

某微程序控制器采用直接表示/编码表示法进行编码，已知共有33个微命令，构成5个互斥类，分别包含7、3、12、5、6个微命令，则控制字段位数为?

A 12、5

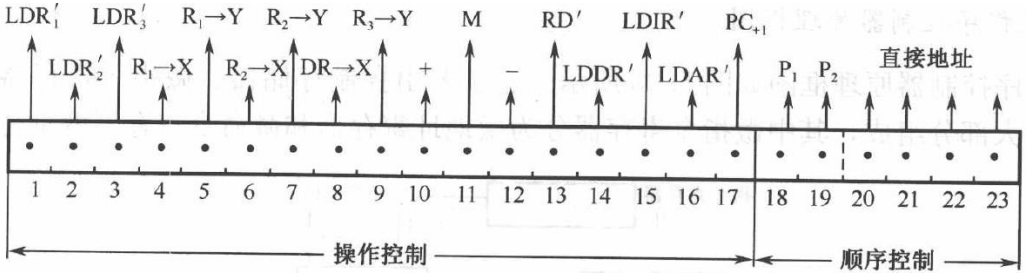
B 33、12

C 33、15

D 33、5

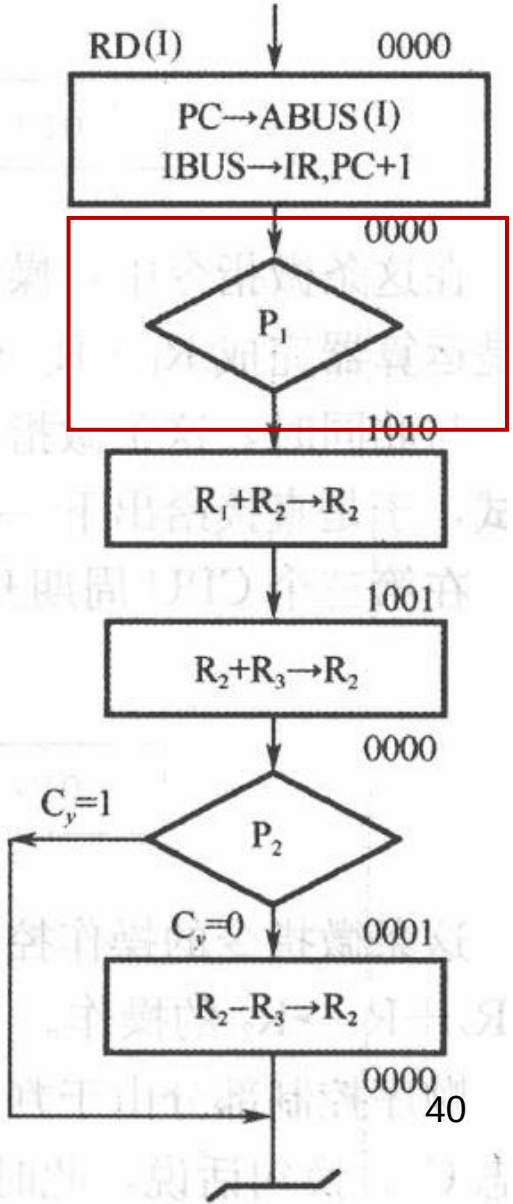
第五章 中央处理机

- 微程序/硬件布线控制器简介
- 微程序控制原理
- 微程序控制技术
  - 微指令结构（操作控制字段）
  - 微地址形成（顺序控制字段）
  - 微指令格式
- 微程序控制设计



微地址形成方法

- 入口地址
  - 每条机器指令对应一段微程序，当公用的取指微程序从主存中取出机器指令
  - 通过P1测试，机器指令的操作码字段指出各段微程序的入口地址，这是一种多分支(或多路转移)的情况
- 微地址形成方式
  - 计数器方式
  - 多路转移方式

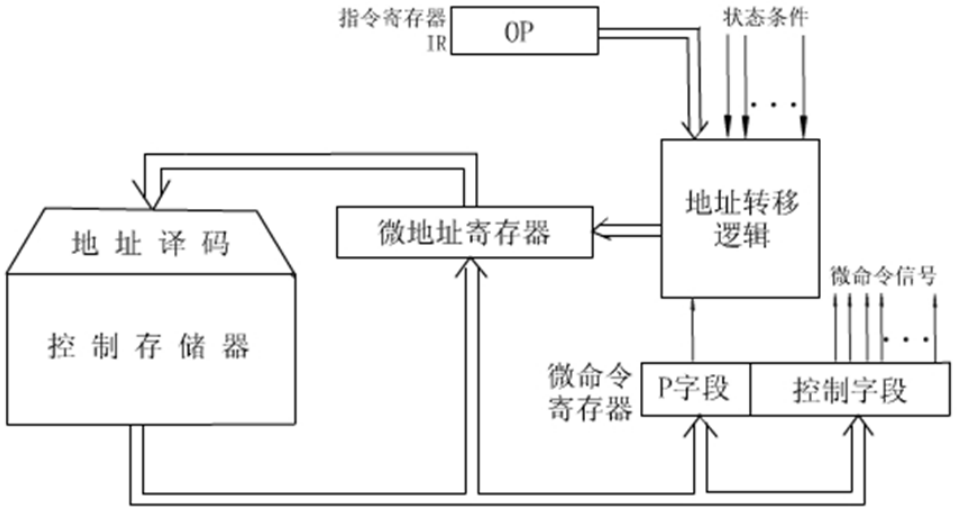


计数器方式

- 计数器方式
  - 微程序顺序执行时，其后继微地址就是现行微地址加上一个增量（通常为1）
  - 当微程序遇到转移或转子程序时，由微指令的转移地址段来形成转移微地址
  - 在微程序控制器中也有一个微程序计数器μPC，一般情况下都是将微地址寄存器μMAR作为μPC
- 性能分析
  - 优点：简单、易于掌握，编制微程序容易
  - 缺点：这种方式不能实现两路以上的并行微程序转移，不利于提高微程序的执行速度

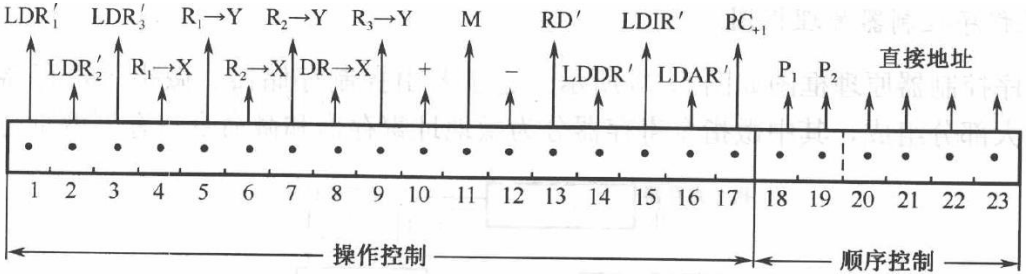
多路转移方式

- 多路转移方式
  - 基本原理
    - 若不产生分支，顺序执行
    - 产生分支，根据状态条件，产生“备选”地址（IR OP段）
    - N位状态条件标志→2^N路转移
  - 性能分析：灵活性好、并行转移；需设计地址转移逻辑



第五章 中央处理机

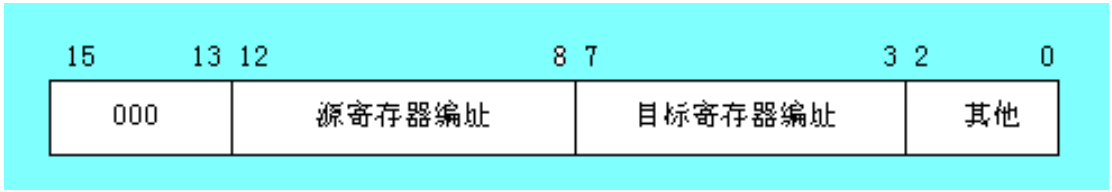
- 微程序/硬件布线控制器简介
- 微程序控制原理
- 微程序控制技术
  - 微指令结构
  - 微地址形成
  - 微指令格式（常见水平型，其他表示方法？）
- 微程序控制设计



微指令格式分类

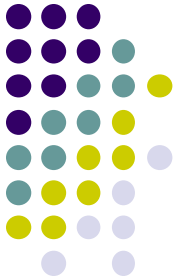
- **水平型微指令**
  - 指一次能定义并能并行执行多个微命令的微指令
  - 分别为：直接表示、编码、混合
- **垂直型微指令**
  - 类似机器指令操作码方式，由微操作码字段给出微命令
  - 缺点：一次给出的微命令数目较少

水平型



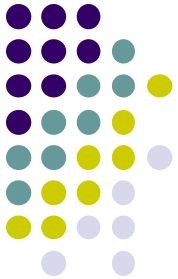
垂直型

水平型/垂直型微指令比较



- 水平型微指令和垂直型微指令的比较
  - 并行能力
    - 水平型微指令并行操作能力强，效率高，灵活性强
    - 垂直型微指令则较差
  - 执行时间
    - 水平型微指令执行一条指令的时间短
    - 垂直型微指令执行时间长
  - 微指令字长
    - 水平型微指令：微指令字较长而微程序短的特点
    - 垂直型微指令则相反

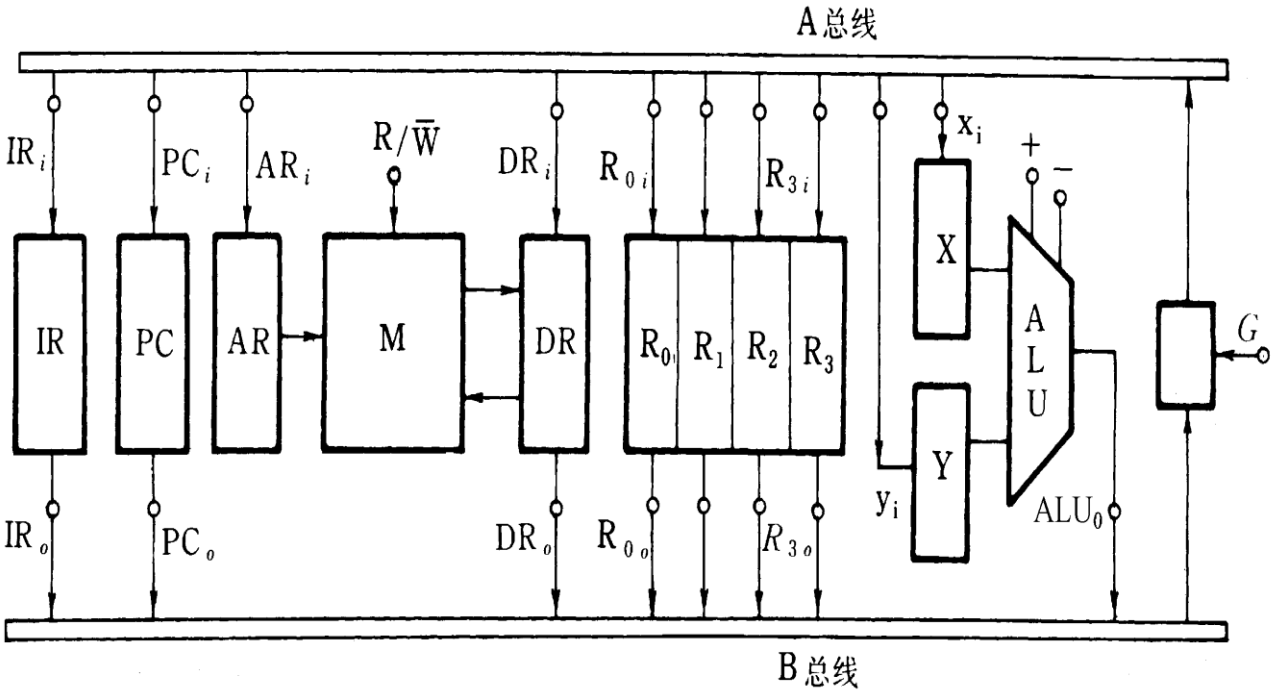
第五章 中央处理机



- 微程序/硬件布线控制器简介
- 微程序控制原理
- 微程序控制技术
- 微程序控制设计（填uCM，填微指令）

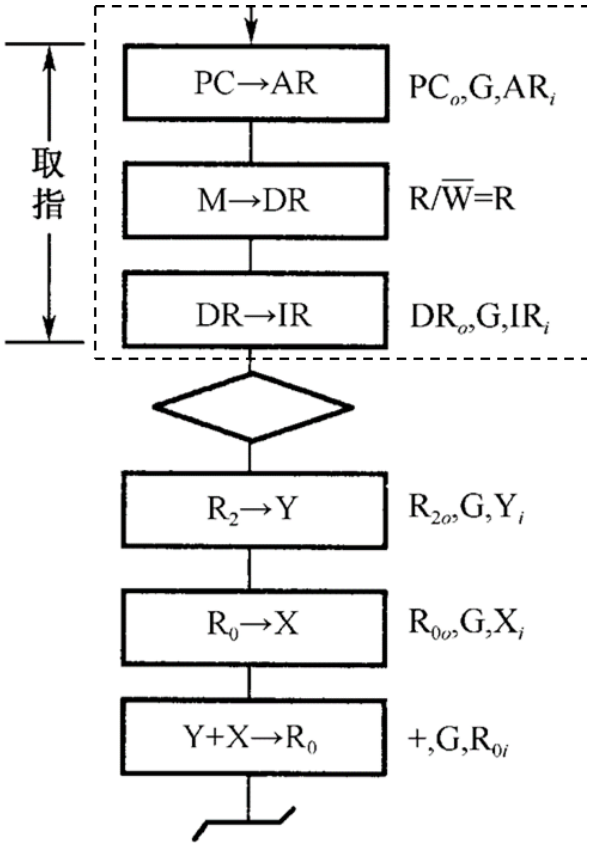
指令周期流程图与微程序控制

- 指令周期流程图与微程序控制对比与转换
  - 共同点：对机器指令的微观（微操作）表征
  - 示例，ADD R2，R0 → （功能）R2+R0→R0

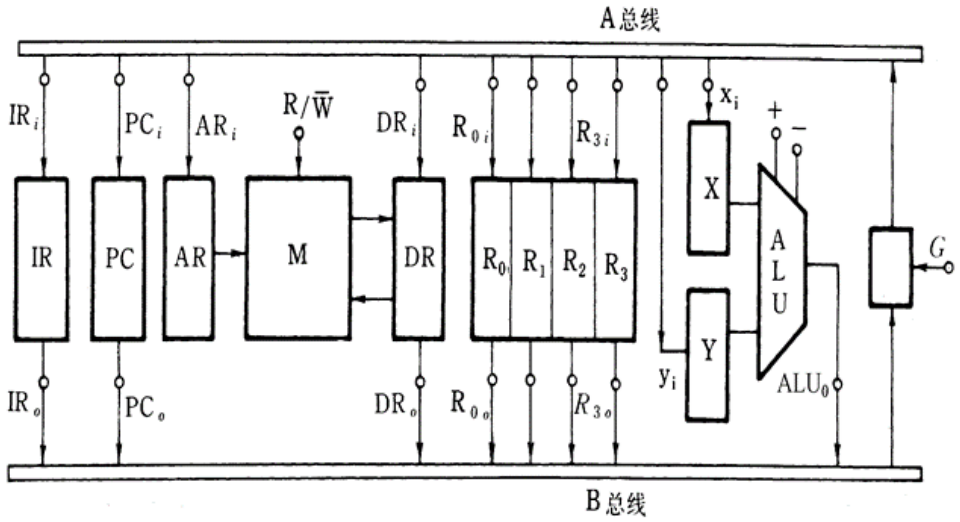


指周期流程图（图形化表示）

- 示例，ADD R2，R0 → （功能）R2+R0→R0



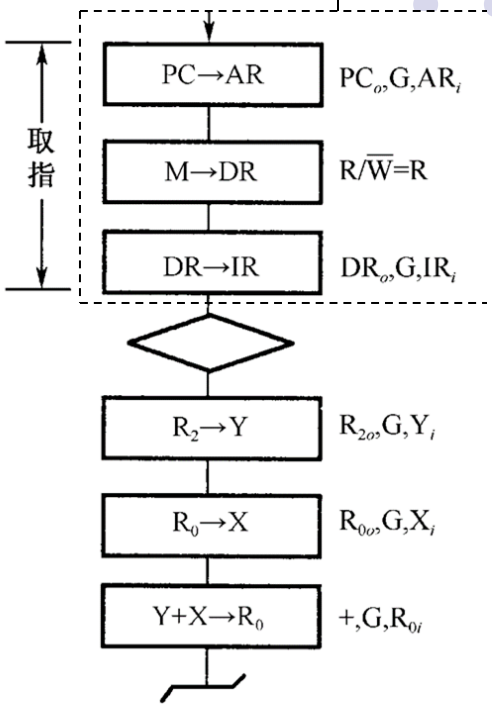
(a) 加法



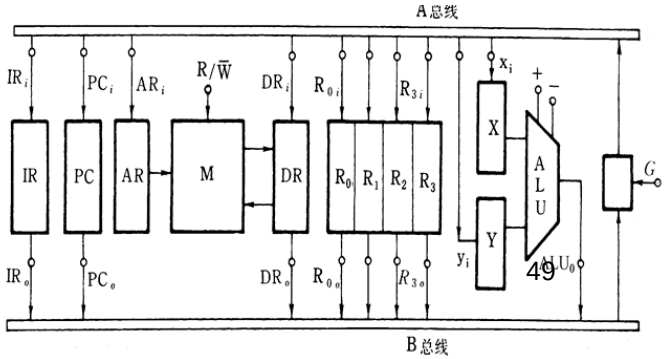
微程序控制（程序化表示）

- 根据表格情况，调整微指令内容

微地址	微操作命令	功能
		取指阶段 uAR=OP;微地址由操作码决定
……	……	……
		ADD R2，R0： R2+R0→R0 注：ADD语句 OP码为0101

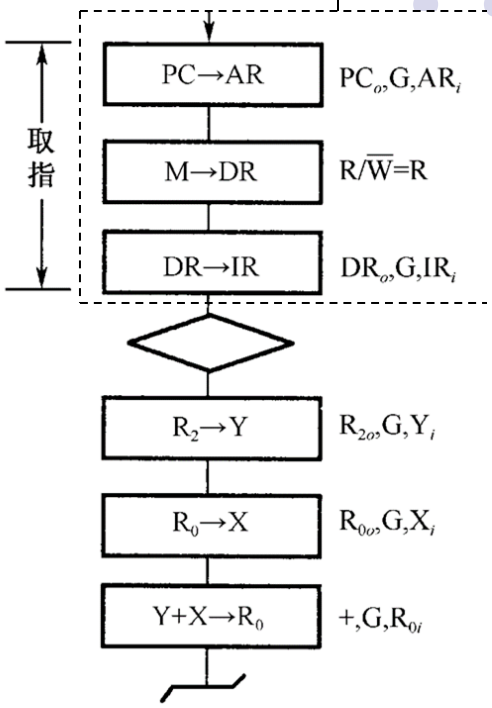


(a) 加法

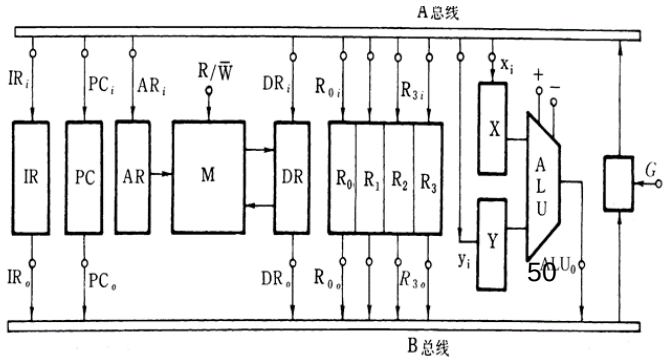


微程序控制（程序化表示）

微地址	微操作命令	功能
0	PC_o,G,AR_i ,R/W=R uAR=uAR+1	取指阶段 uAR=OP;微地址由操作码决定
1	DR_o,G,IR_i uAR=OP	
……	……	……
5	R_{2o},G,Y_i uAR=uAR+1	ADD R2，R0： R2+R0→R0 注：ADD语句 OP码为0101
6	R_{0o},G,X_i uAR=uAR+1	
7	+,G,R_{0i},uAR=0	



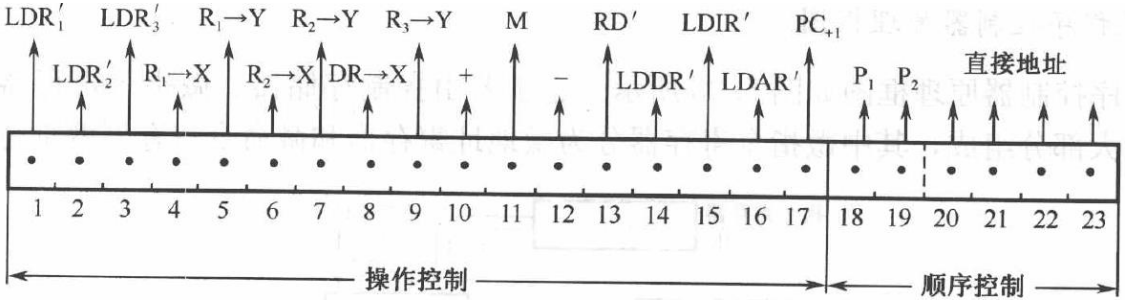
(a) 加法

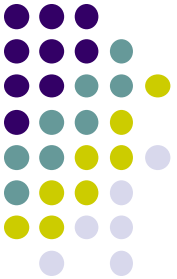


微程序控制（微指令编码）

- 按指令格式编码
  - P测试
  - 下地址
- 根据实验流程加强知识巩固

微地址	微操作命令	功能
0	PC_o,G,AR_i ,R/W=R uAR=uAR+1	取指阶段 uAR=OP;微地址由操作码决定
1	DR_o,G,IR_i uAR=OP	
……	……	……
5	R_{2o},G,Y_i uAR=uAR+1	ADD R2，R0：R2+R0→R0 注：ADD语句OP码为0101
6	R_{0o},G,X_i uAR=uAR+1	
7	+,G,R_{0i},uAR=0	





- 微程序控制
  - 微程序控制原理
    - 优点：灵活、设计方便
    - 基本概念
      - 微命令-微操作
        - 相容
        - 互斥
      - 微指令
        - 操作控制字段
        - 顺序控制字段
      - 微程序-机器指令
    - 逻辑结构
      - 控制存储器（μCM）
      - 微指令寄存器（μIR）
      - 微地址形成部件
      - 微地址寄存器（μMAR、μAR）
  - 微程序控制技术
    - 微指令结构
      - 直接表示法：速度快、控制字长
      - 编码表示法：互斥指令编码、缩短字长
      - 混合表示法
    - 微地址形成
      - 入口地址（0000）
      - 顺序判别字段（跳转或分支）
      - 微地址形成方式
        - 计数器方式
        - 多路转移方式
    - 微指令格式
      - 水平型微指令：常见方法、指令字长、微程序短
      - 垂直性微指令：参考汇编、指令字短、微程序长
  - 微程序控制示例
    - 指令周期的微程序表示
      - 对照指令周期流程图
      - 注意：微地址μAR