

计算机组成原理

PRINCIPLES OF COMPUTER ORGANIZATION

第26次课：微程序控制计算机的基本工作原理

杜国栋

信息科学与工程学院计算机科学与工程系

gddu@ysu.edu.cn



燕山大学
YANSHAN UNIVERSITY



课程目标

- 掌握实现微程序控制的基本原理；
- 熟悉微程序控制的基本概念；
- 了解微程序控制器。





产生操作控制信号——微程序控制

微程序控制器发展：

(1) 微程序设计的概念和原理最早是由英国剑桥大学的 M.V .Wilkes 教授于 1951 年提出来的。

(2) 1964 年, IBM 公司在 IBM360 系列机上成功地采用了微程序设计技术,解决了指令系统的兼容问题。

(3) 70 年代以来, 由于 VLSI 技术的发展, 推动了微程序设计技术的发展和应用, 目前, 大多数计算机都采用微程序设计技术。





产生操作控制信号——微程序控制

基本思想

- ◆ 若干条微命令编制成一条微指令，控制实现一步操作
- ◆ 若干条微指令组成一段微程序，解释执行一条机器指令
- ◆ 微程序事先放在控制存储器中，执行机器指令时再取出

CPU的构成 { 引入了程序技术，使设计规整；
 引入了存储逻辑，使功能易于扩展。



微操作 —

一条指令可以分解成很多最基本的操作，这种最基本的不可再分割的操作称为微操作。不同的机器指令具有不同的微操作序列。

微命令 —

在微程序控制的计算机中，将控制部件向执行部件发出的各种控制命令叫做微命令，它是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。因此，微命令是控制计算机各部件完成某个基本微操作的命令。

微指令 —

在微程序控制的计算机中，将由同时发出的控制信号所执行的一组微操作称为微指令。将一条指令分成若干条微指令，按次序执行这些微指令，就可以实现指令的功能。

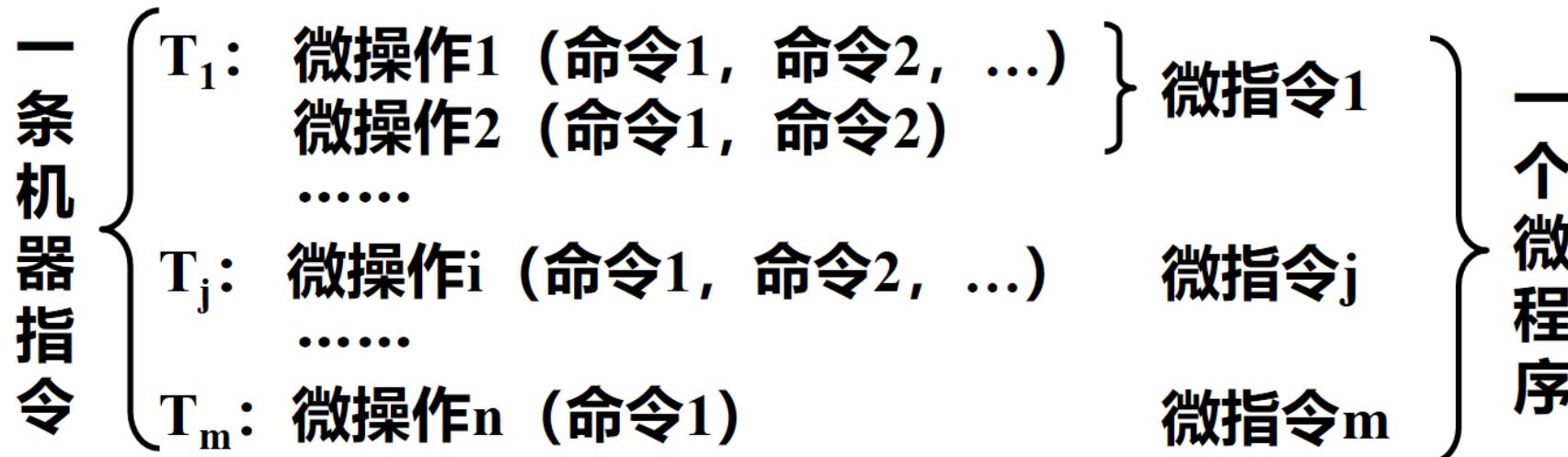
微程序 —

计算机的程序由指令序列构成，而计算机每条指令的功能均由微指令序列解释完成，这些微指令序列的集合就叫做微程序。



产生操作控制信号—微程序控制

指令、微程序、微指令的关系：





产生操作控制信号——微程序控制

程序和微程序的区别？

- (1) 层次不同，一个处于程序层次，一个处于微程序层次；(**软件层次 vs. 硬件层次**)
- (2) 微程序是由微指令组成的，它用于描述机器指令，实际上是机器指令的实时解释器，它是由计算机的设计者事先编制好并存放在控制存储器中的。对于程序员来说，计算机系统中微程序一级的结构和功能是透明的；
- (3) 程序则最终由机器指令组成，它是由软件设计人员事先编制好并存放在主存或辅存中的。





产生操作控制信号—微程序控制

机器指令和微指令的区别？

- (1) 机器指令和微指令所处的级别不同，机器指令在程序层次，微指令在微程序层次；
- (2) 一系列微指令的有序集合构成一段微程序，完成一条机器指令的功能实现。
- (3) 机器指令和微指令结构不同，机器指令由操作码和操作数地址构成，微指令由控制信号（或微操作或微命令）和下址构成。





产生操作控制信号——微程序控制

控制存储器：

存储微程序的存储器，称为控制存储器；

由于微程序是对机器指令的解释，且指令系统的机器指令固定，所以微程序固定，控制存储器只需要ROM即可；

由于微程序由微指令序列组成，而微指令包括控制字段和下一条微指令地址，故微指令字长要远大于指令字长，继而需要较大容量的控制存储器；

执行一条机器指令就相当于执行控制存储器中的一段微程序。

一条机器指令的功能一般对应于（ ）。

- A 一段微程序
- B 一条微指令
- C 一条微命令
- D 一个微操作

提交

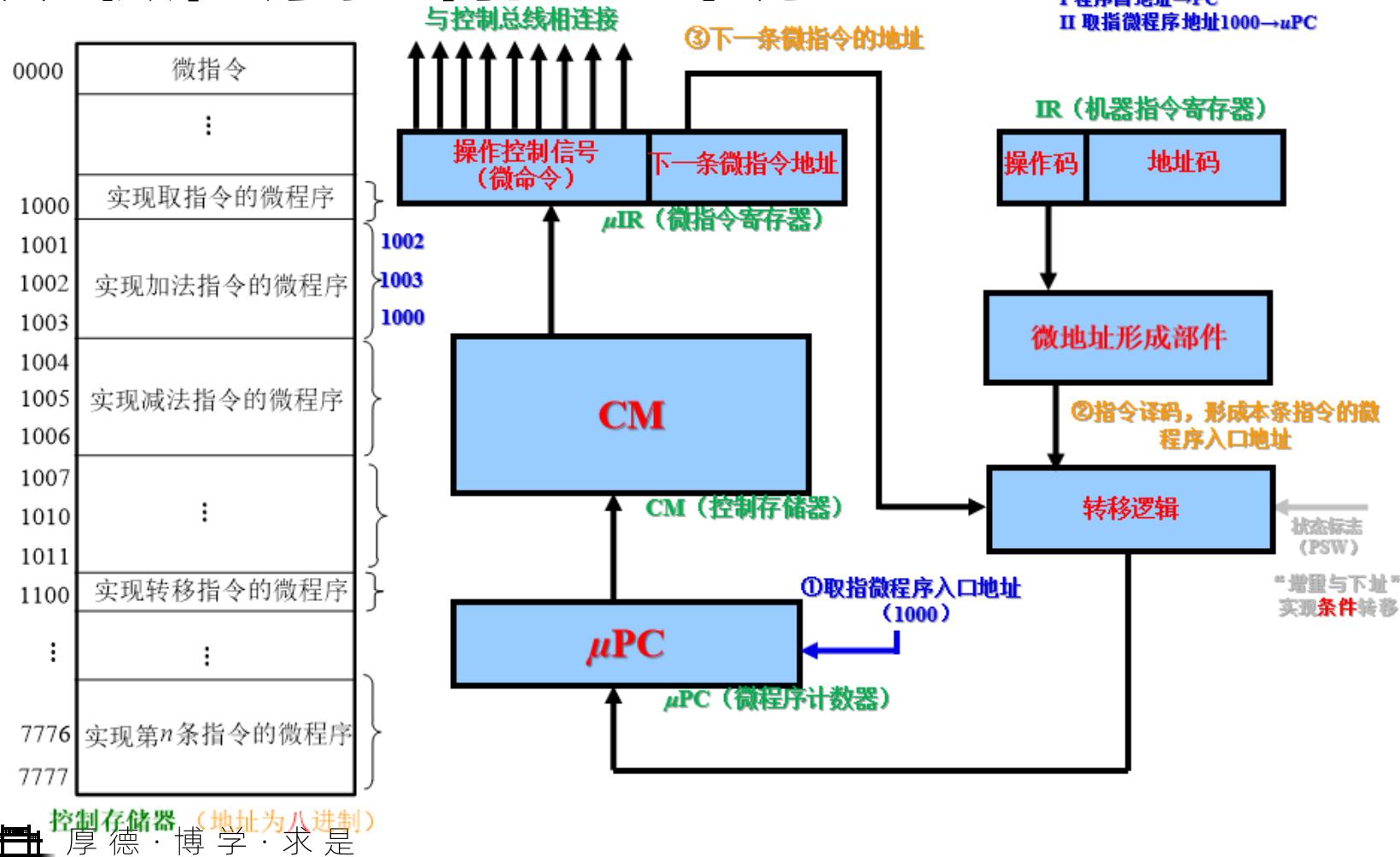
一条机器指令的功能一般对应于（ ）。

- A 一段微程序
- B 一条微指令
- C 一条微命令
- D 一个微操作

提交

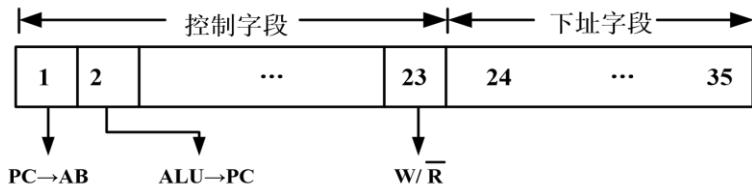


实现微程序控制的基本原理





实现微程序控制的基本原理

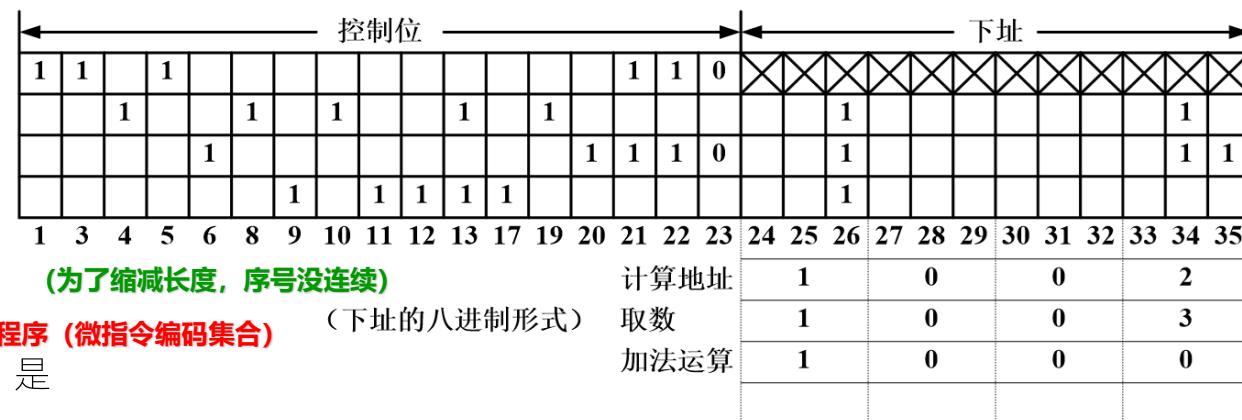


(1) 教材中模型机共有23个控制信号，所以总共有23个控制位，即控制字段为23位；

(2) 如控制存储器为4K字(寻址范围为 2^{12})，则每条微指令还需要12位来表示下址。

序号	控制信号	功 能	序号	控制信号	功 能
1	PC→AB	指令地址送地址总线	13	+	ALU 进行加法运算
2	ALU→PC	转移地址送 PC	14	-	ALU 进行减法运算
3	PC+1	程序计数器加 1	15	A	ALU 进行逻辑乘运算
4	Imm(Disp) →ALU	立即数或位移量送 ALU	16	V	ALU 进行逻辑加运算
5	DB→IR	取指到指令寄存器	17	ALU→GR	ALU 运算结果送通用寄存器
6	DB→DR	数据总线上的数据送数据寄存器	18	ALU→DR	ALU 运算结果送数据寄存器
7	DR→DB	数据寄存器中的数据送数据总线	19	ALU→AR	ALU 计算得到的有效地址送地址寄存器
8	rsl→GR	寄存器地址送通用寄存器	20	AR→AB	地址寄存器内容送地址总线
9	rs,rd→GR	寄存器地址送通用寄存器	21	ADS	地址总线上地址有效
10	(rs1)→ALU	寄存器内容送 ALU	22	M/ \overline{IO}	访问存储器或 I/O
11	(rs)→ALU	寄存器内容送 ALU	23	W/R	写或读
12	DR→ALU	数据寄存器内容送 ALU			

微指令格式 方块内为空表示不发出信号





微程序的设计技术

在实际进行微程序设计时，要考虑下面三个问题：

- (1) 如何缩短微指令字长 (缩短操作控制字段)；
- (2) 如何减少微程序长度 (增加条件转移—循环)；
- (3) 如何提高微程序设计的执行速度 (微指令格式设计=垂直型→水平型)。





微指令控制字段的编译法（缩短为指令字长）

微指令的编码，就是对微指令的控制字段采用的表示方法。

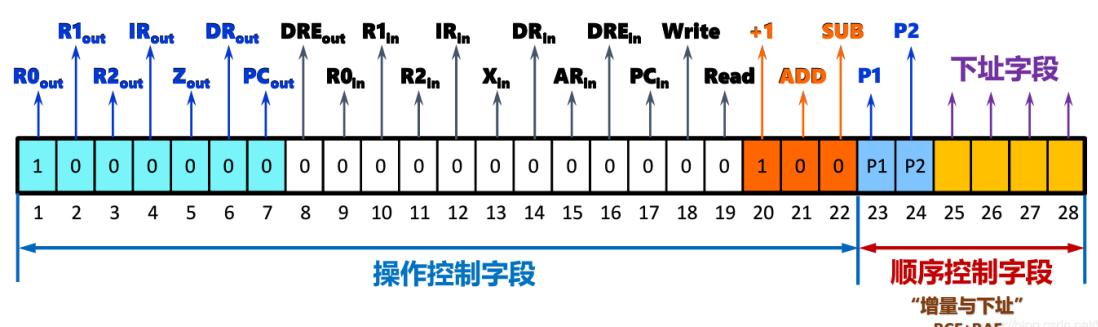
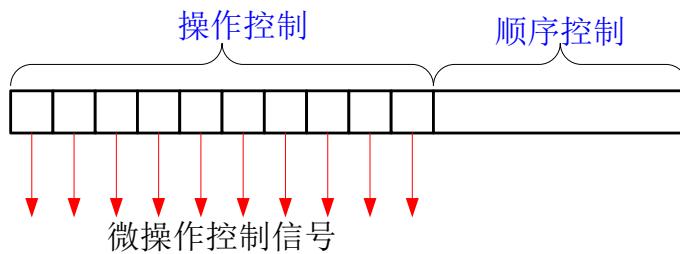
(1) 直接控制法(直接编码法、不译码法)

操作控制字段中的每一位代表一个微命令。

“1” - 有效，“0” - 无效。

优点：速度最快、简单直观，输出直接用于控制

缺点：微指令字较长，因而使控制存储器容量较大。





微指令控制字段的编译法（缩短为指令字长）

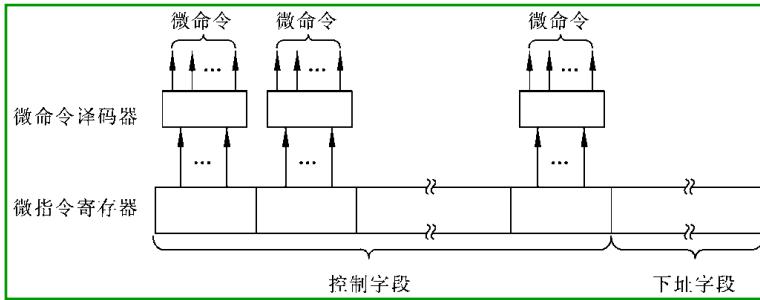
(2) 字段直接编译法（使用较普遍）

微周期（机器周期）：是一条微指令所需的执行时间。

如果有若干个（一组）微命令，在每次选择使用它们的机器周期内（基本操作），只有一个微命令起作用，那么这若干个微命令是互斥的。

选择互斥的微命令入同组分组编码，用微命令译码器译码控制。

如：PC-G、ALU-G、R0-G、R1-G、R2-G，其中G指通用寄存器



分段的原则：

- 互斥性的微命令分在同一段内，兼容性的微命令分在不同段内；
- 每个小段中包含的信息位不能太多，否则将增加译码线路的复杂性和译码时间；
- 每个小段要留出一个状态，表示本字段不发出任何命令，通常用00...0表示

例：字段长度为3位时，最多只能表示7个互斥的微命令，通常代码000表示不发微命令。字段长度n与所能表示的微命令数m的关系： $m=2^n-1$

优点：缩短了微指令长度。

代价：在微指令寄存器的输出端，为该字段增加一个译码器，该译码器的输出即为原来的微命令。

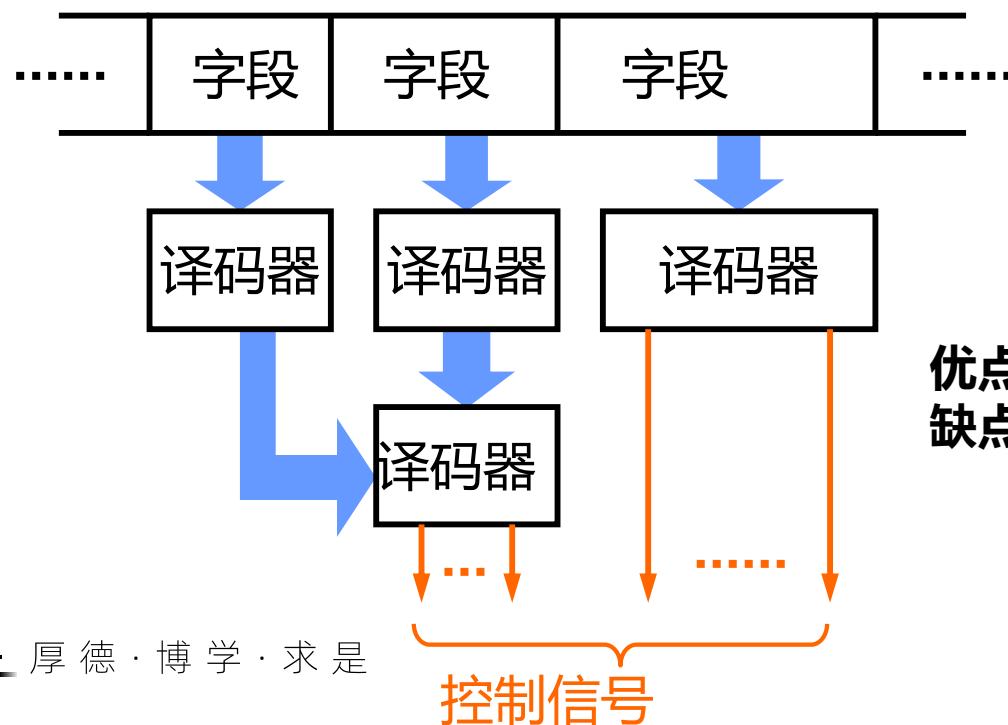




微指令控制字段的编译法（缩短为指令字长）

(3) 字段间接编译法

如果在字段直接编译法中，还规定一个字段的某些微命令，要兼由另一字段中的某些微命令来解释，称为字段间接编译法。



优点：进一步减少了指令长度。
缺点：削弱微指令的并行控制能力。





微程序流的控制

-增加程序流灵活性，即条件转移

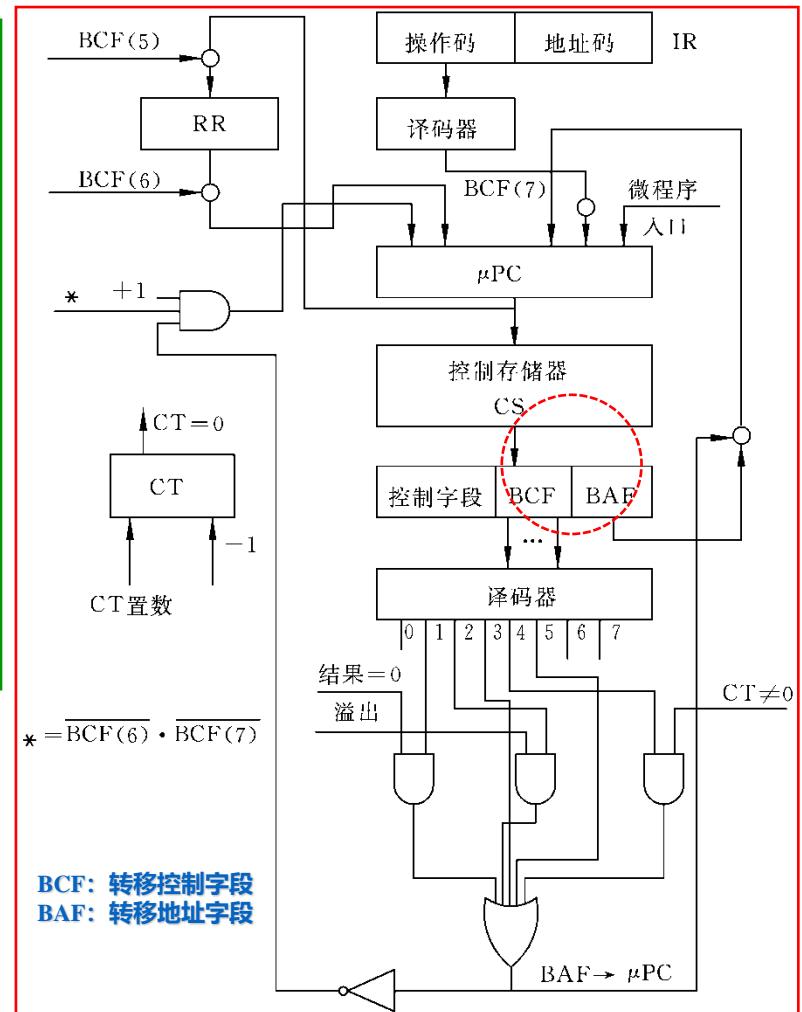
表 6.2 产生后继微地址的微命令

BCF 字段		硬件条件	计数器 CT		返回寄存器 RR 输入	后继微地址
编码	微命令名称		操作前	操作		
0	顺序执行	×	×	×	×	$\mu\text{PC}+1$
1	结果为 0 转移	结果为 0	×	×	×	BAF
						$\mu\text{PC}+1$
2	结果溢出转移	溢出	×	×	×	BAF
		不溢出				$\mu\text{PC}+1$
3	无条件转移	×	×	×	×	BAF
4	测试循环	×	为 0	CT-1	×	$\mu\text{PC}+1$
			不为 0			BAF
5	转微子程序	×	×	×	$\mu\text{PC}+1$	BAF
6	返回	×	×	×	×	RR
7	操作码形成微地址	×	×	×	×	由操作码形成

注：×表示“任意”或“无影响”。

机器启动过程（通电）

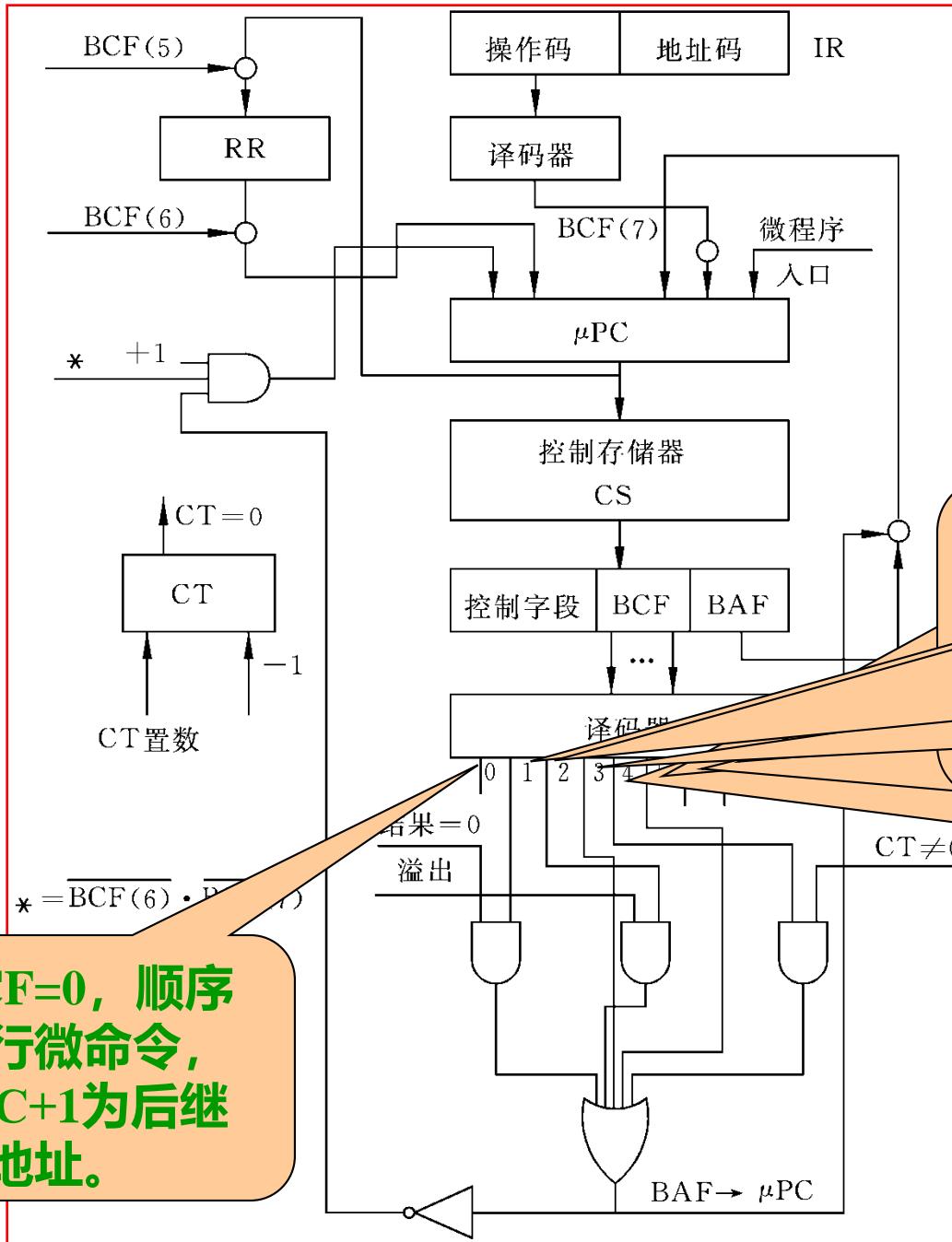
- (1) 由 reset 信号在 PC 内置入开机后执行的第一条指令的地址； **程序首地址 → PC**
- (2) 同时在微指令寄存器内置入一条“取指”微指令； **取指微程序地址 1000 → μPC**
- (3) 并将其他一些有关的状态位或寄存器置于初始状态。



使微程序具有程序的流程控制功能：

- 1、反复执行一条微指令（提高效率）；
- 2、有条件 (PSW) 转移 (动态性)；
- 3、异常 (溢出) 的处理 (动态性)；

“增量与下址字段”方式原理图



BCF=0，顺序执行微命令，
μPC+1为后继微地址。

指令。

微地址从BAF达μPC。
假如CT=0，表示循环结束，后继微地址为μPC+1。本条微命令同时完成CT-1操作。

BCF=5，转微子程序
微命令 增量与下址字段

BCF=2，条件转移微命令，
当运算结果溢出时，将BAF送μPC，否则
μPC+1→μPC



微指令格式 (提高微程序设计的执行速度，微指令垂直型→水平型)

(1) 水平型微指令

从编码方式看，直接编码、字段直接编码和字段间接编码都属于水平型微指令。

特点：在一条微指令中定义并执行多个并行操作微命令。

水平型微指令 一次能定义并执行多个并行操作。

基本格式

如何表示一系列控制信号？



优点：微程序短，执行速度快；

缺点：微指令长，编写微程序较麻烦。

(2) 垂直型微指令

特点：不强调实现微指令的并行控制功能，通常一条微指令只要求能控制实现一二种操作。这种微指令格式与指令相似；每条指令有一个操作码；每条微指令有一个微操作码。

基本格式

优点：微指令短、简单、规整，便于编写微程序；

缺点：微程序长，执行速度慢，工作效率低。



(3) 混合型微指令

在垂直型的基础上增加一些不太复杂的并行操作。微指令较短，仍便于编写；微程序也不长，执行速度加快。



厚德·博学·求是



微指令格式 (提高微程序设计的执行速度, 微指令垂直型→水平型)

水平型微指令与垂直型微指令的比较

- (1) **水平型**微指令并行操作能力强, 效率高, 灵活性强, **垂直型**微指令则差。
- (2) **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。
- (3) 由**水平型**微指令解释指令的微程序, 具有微指令字比较长, 但微程序短的特点。**垂直型**微指令则相反, 微指令字比较短而微程序长。
- (4) **水平型**微指令用户难以掌握, 而**垂直型**微指令与指令比较相似, 相对来说, 比较容易掌握。



例题：某机共有 52 个微操作控制信号，构成 5 个相斥类的微命令组，各组分别包含 5、8、2、15、22 个微命令。已知可判定的外部条件有两个，微指令字长 28 位。

- (1) 若微指令采用水平型并且后续微指令地址的形成采用下址字段法，试设计符合题目要求的微指令格式。
- (2) 指出该机控制存储器的容量是多少位 (bit)。

 作答



产生操作控制信号—微程序控制

例题：某机共有 52 个微操作控制信号，构成 5 个相斥类的微命令组，各组分别包含 5、8、2、15、22 个微命令。已知可判定的外部条件有两个，微指令字长 28 位。

- (1) 若微指令采用水平型并且后续微指令地址的形成采用下址字段法，试设计符合题目要求的微指令格式。
- (2) 指出该机控制存储器的容量是多少位 (bit)。

解答：

- (1) 微指令格式如下：



微命令字段（控制字段）：
 $3+4+2+4+5=18$ 位

2位

$28-18-2=8$ 位

- (2) 该机的控存容量为：

$$2^8 \times 28 = 7168 \text{ 位}$$



厚德·博学·求是



例题：某机采用微程序控制方式，微指令字长 24 位，采用水平型编码控制的微指令格式，共有微命令 28 个，构成 4 个互斥类，各包含 5 个、 7 个、 14 个和 2 个微命令。控制下址的外部条件共 3 个（直接控制）。

- (1) 控制存储器容量应该为？
- (2) 设计出微指令的具体格式。

 作答



产生操作控制信号—微程序控制

例题：某机采用微程序控制方式，微指令字长 24 位，采用水平型编码控制的微指令格式，共有微命令 28 个，构成 4 个互斥类，各包含 5 个、 7 个、 14 个和 2 个微命令。控制下址的外部条件共 3 个（直接控制）。

(1) 控制存储器容量应该为？

(2) 设计出微指令的具体格式。

(1) 4 个相斥类的控制位分别为 3 位， 3 位， 4 位， 2 位，共 12 位；控制下址的外部条件 3 位（直接控制），余下 9 位作为下址，所以控制存储器的容量最多为 $2^9 \times 24\text{位} = 512 \times 24\text{位}$

(2) 微指令格式如图所示：

3bit	3bit	4bit	2bit	外部条件 3bit	下址 9bit
------	------	------	------	-----------	---------





产生操作控制信号——硬布线控制

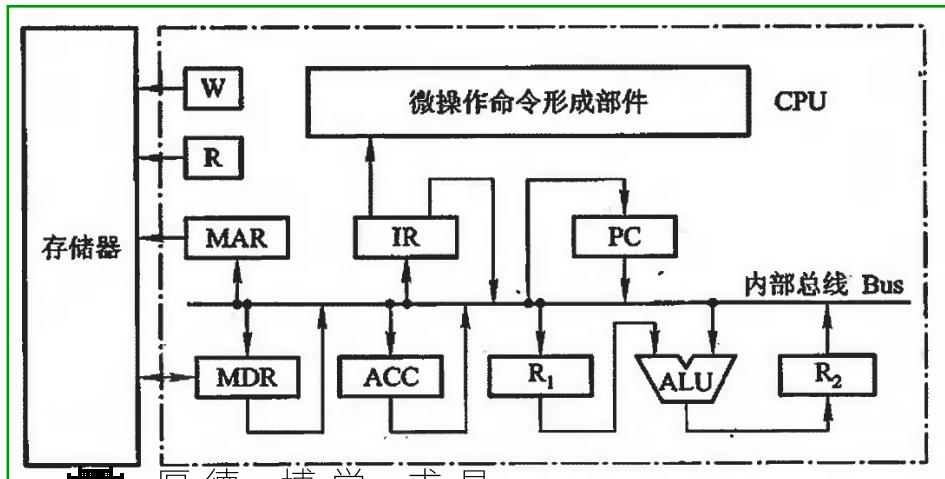
硬布线控制：微操作产生部件通过逻辑电路直接连线而产生控制信号（组合逻辑设计思想），以布尔代数（数字电路课程）为工具设计。

输入：来自指令译码器的输出、时序发生器的时序信号

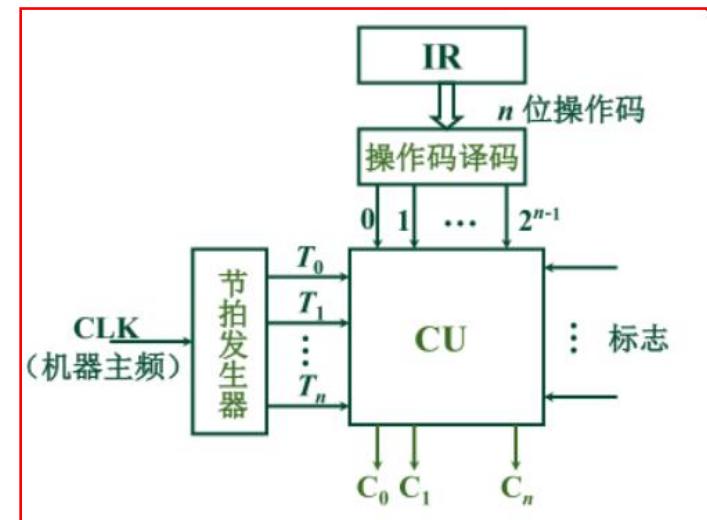
输出：一组带有时间标志的微操作控制信号

每个微操作控制信号是指令、时序等的逻辑函数，可表示为：

微操作=周期·节拍·脉冲·指令码·其他条件



计算机操作的最短时间单位是时钟周期。



一个微命令，可能在不同指令码、不同机器周期、不同时钟周期（节拍）、不同外部条件下产生。具体表达式可将指令系统中每条指令的指令周期进行细化并列表分析得出。



产生操作控制信号——硬布线控制

硬布线控制与微程序控制的比较

1. 实现

微程序控制器的控制功能是在存放微程序的控制存储器和存放当前正在执行的微指令的寄存器直接控制下实现的，而硬布线控制则由逻辑门组合实现。

设计简单，方便修改与扩展 \leftrightarrow 结构复杂，设计麻烦，检查调试困难

2. 性能

微程序控制的速度 < 硬布线控制的速度

原因：执行每条微指令都要从控存中读取一次，影响了速度，而硬布线逻辑主要取决于电路延迟





产生操作控制信号—硬布线控制

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

RISC的速度快,所以普遍采用硬布线控制器



相对于微程序控制器，硬布线控制器的特点是（ ）。

- A 指令执行速度慢，指令功能的修改和扩展容易
- B 指令执行速度慢，指令功能的修改和扩展难
- C 指令执行速度快，指令功能的修改和扩展容易
- D 指令执行速度快，指令功能的修改和扩展难

 提交

相对于微程序控制器，硬布线控制器的特点是（ ）。

- A 指令执行速度慢，指令功能的修改和扩展容易
- B 指令执行速度慢，指令功能的修改和扩展难
- C 指令执行速度快，指令功能的修改和扩展容易
- D 指令执行速度快，指令功能的修改和扩展难

 提交

相对于硬布线控制器，微程序控制器的优点在于（ ）。

- A 速度较快
- B 结构比较规整
- C 复杂性和非标准化程度较低
- D 增加或修改指令较为容易

 提交

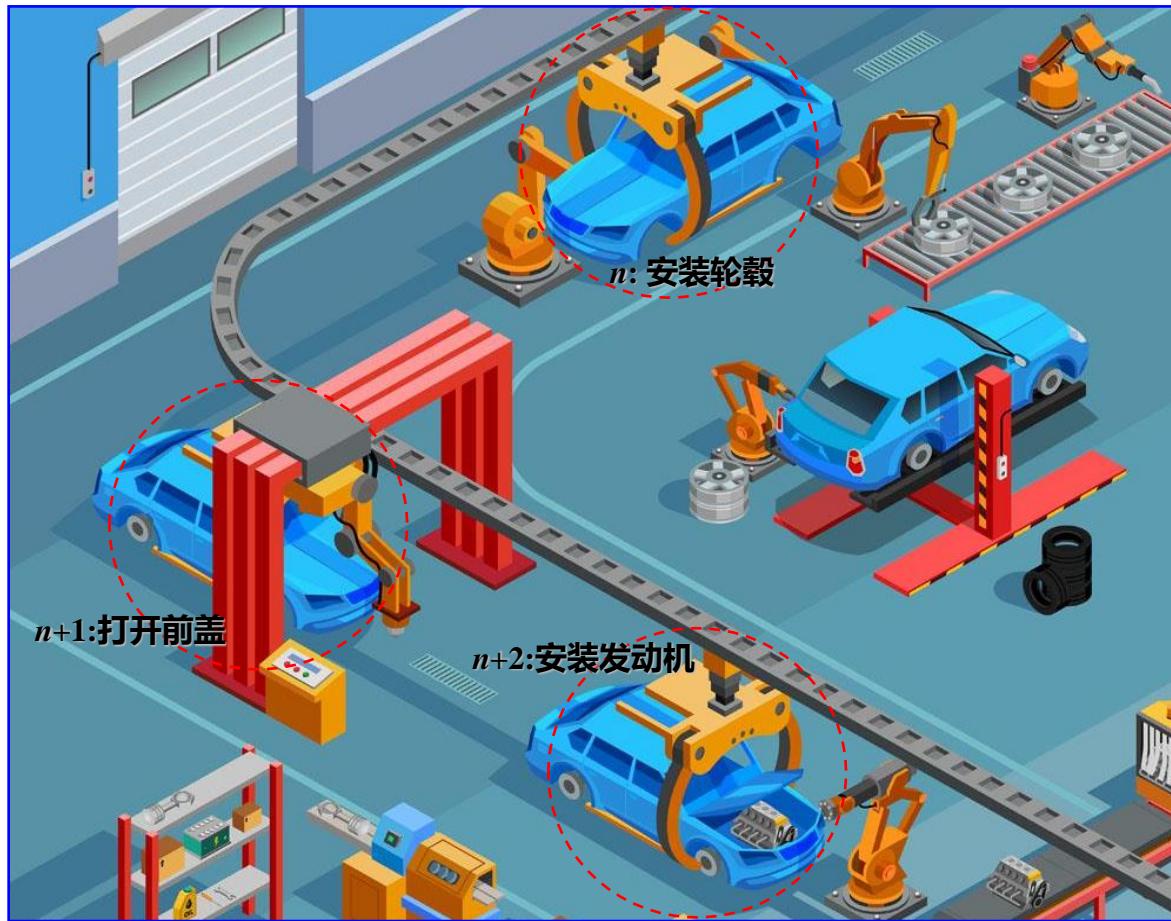
相对于硬布线控制器，微程序控制器的优点在于（ ）。

- A 速度较快
- B 结构比较规整
- C 复杂性和非标准化程度较低
- D 增加或修改指令较为容易

 提交



流水线技术

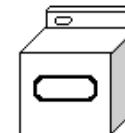




流水线技术

洗衣房

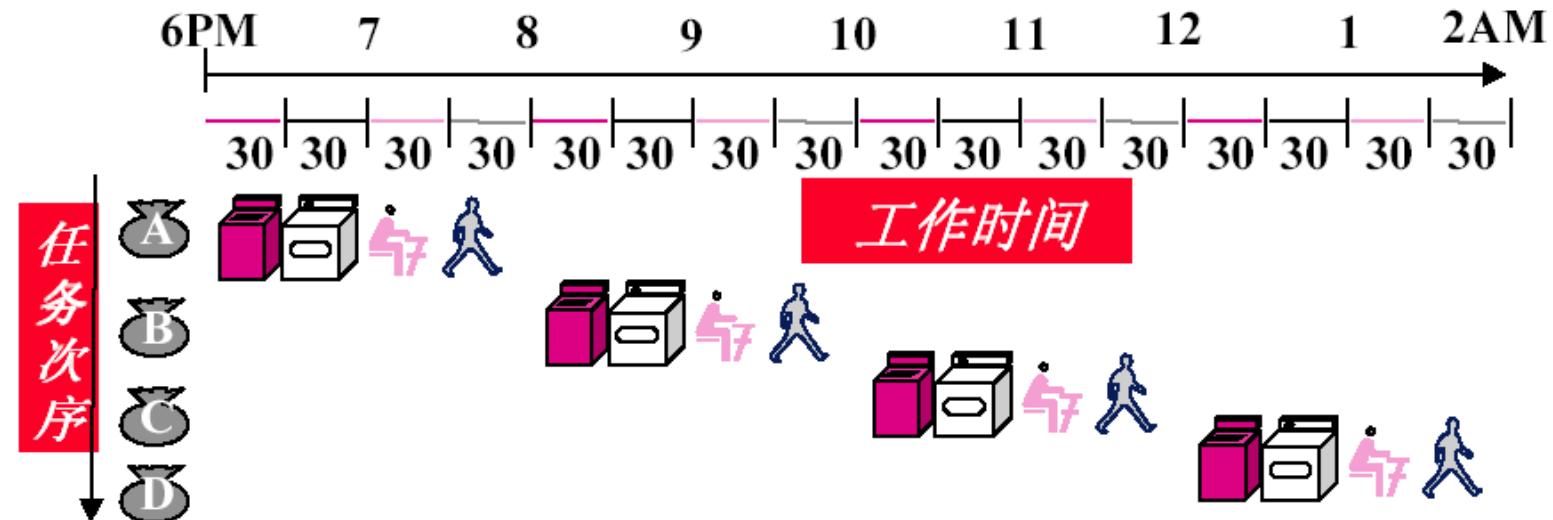
- 张三、李四、王五、赵六每人有一包衣服需要洗涤、烘干、熨整
- 洗衣机需要**30分钟**
- 烘干机需要**30分钟**
- 熨斗需要**30分钟**
- 洗衣工需要**30分钟**将衣物放到抽屉里





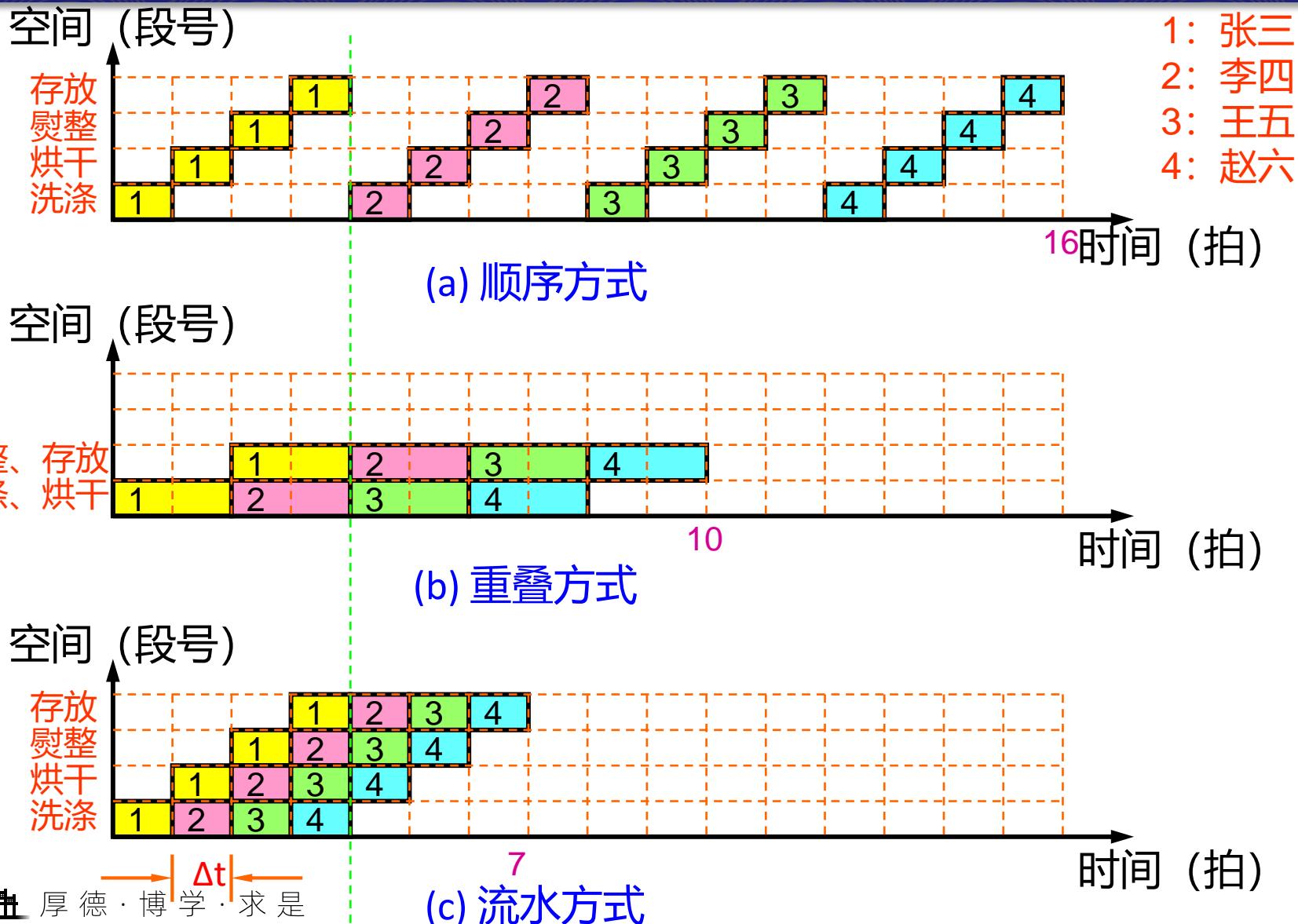
流水线技术

串行洗衣店



- 串行洗衣店需要8个小时完成4个工作量
- 如果他们了解流水技术，那么需要多长时间完成上述工作呢？







流水线技术

取指1	计算地址1	取操作数1	计算存结果1	取指2	计算地址2	...
-----	-------	-------	--------	-----	-------	-----

$$T = mn \Delta t$$

串行微程序控制



指令的4级 ($\rightarrow m$ 级) 流水控制
(4条 $\rightarrow n$ 条)

$$T = m \Delta t + (n-1) \Delta t$$

流水线 —

流水线是将一个较复杂的处理过程分成 m 个复杂程度相当、处理时间大致相等的子过程，每个子过程由一个独立的功能部件来完成，处理对象在各子过程连成的线路上连续流动。在同一时间 Δt ， m 个部件同时进行不同的操作，完成对不同子过程的处理。



厚德·博学·求是

一条机器指令是由若干条（ ）组成的序列来实现的。

- A 微操作
- B 微指令
- C 微程序
- D 指令

提交

一条机器指令是由若干条（ ）组成的序列来实现的。

- A 微操作
- B 微指令
- C 微程序
- D 指令

提交

以下关于机器指令与微指令的叙述中，正确的是（ ）。

- A 计算机运行时，机器指令与微指令都存放在主存中
- B 待执行的微指令的地址存放在程序计数器（PC）中
- C 一条微指令的功能可用多条机器指令组成的程序实现
- D 一条机器指令的功能可用多条微指令组成的微程序实现

 提交

以下关于机器指令与微指令的叙述中，正确的是（ ）。

- A 计算机运行时，机器指令与微指令都存放在主存中
- B 待执行的微指令的地址存放在程序计数器（PC）中
- C 一条微指令的功能可用多条机器指令组成的程序实现
- D 一条机器指令的功能可用多条微指令组成的微程序实现

 提交

关于微程序的叙述，以下正确的是（ ）。

- A 微程序控制器属于硬连线控制器
- B 微程序预先存储在主存的ROM中
- C 每条微指令与一段机器程序相对应
- D 每条机器指令与一段微程序相对应

提交

关于微程序的叙述，以下正确的是（ ）。

- A 微程序控制器属于硬连线控制器
- B 微程序预先存储在主存的ROM中
- C 每条微指令与一段机器程序相对应
- D 每条机器指令与一段微程序相对应

 提交



有问题欢迎随时跟我讨论

办公地点：西校区信息馆423

邮 箱：gddu@ysu.edu.cn

