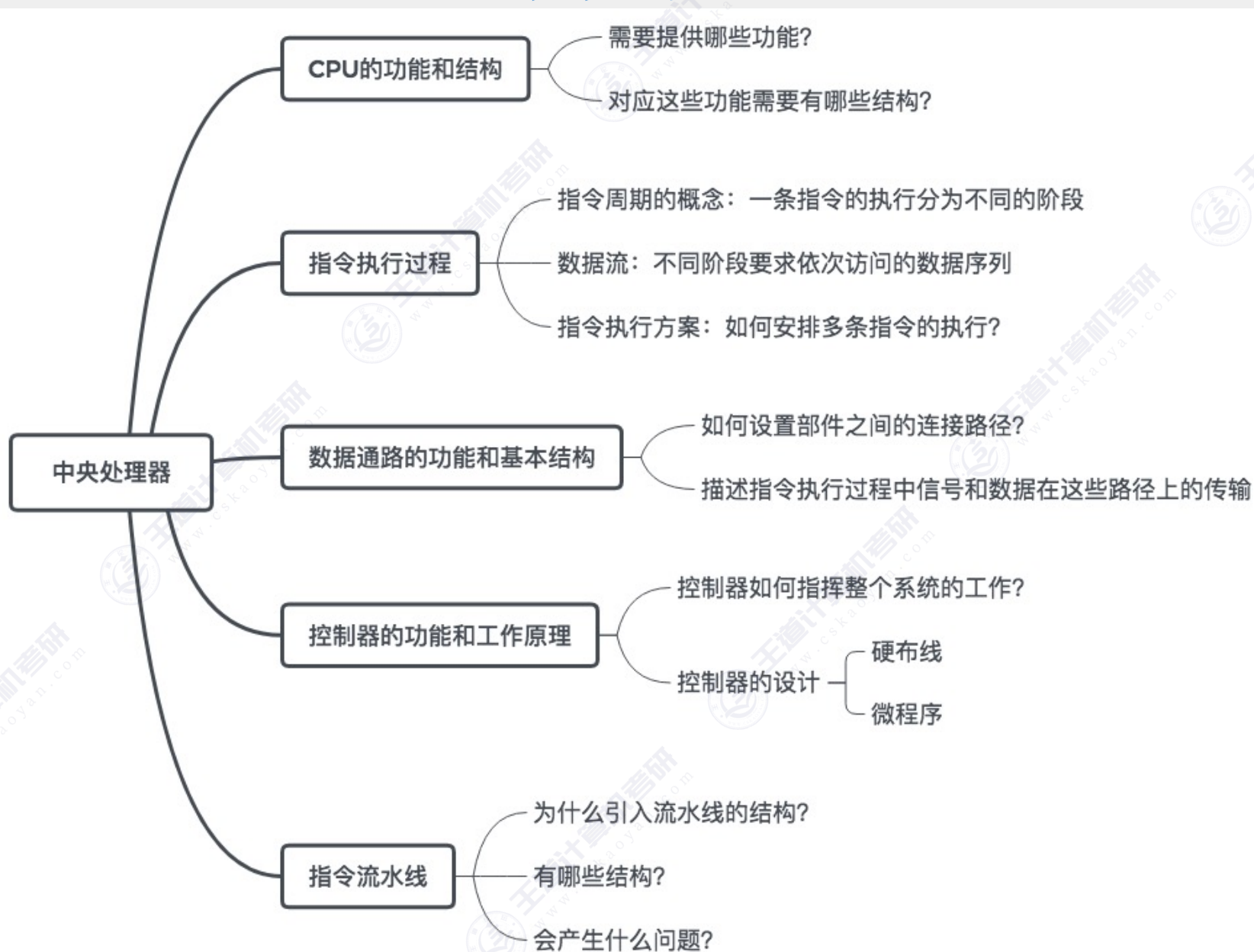


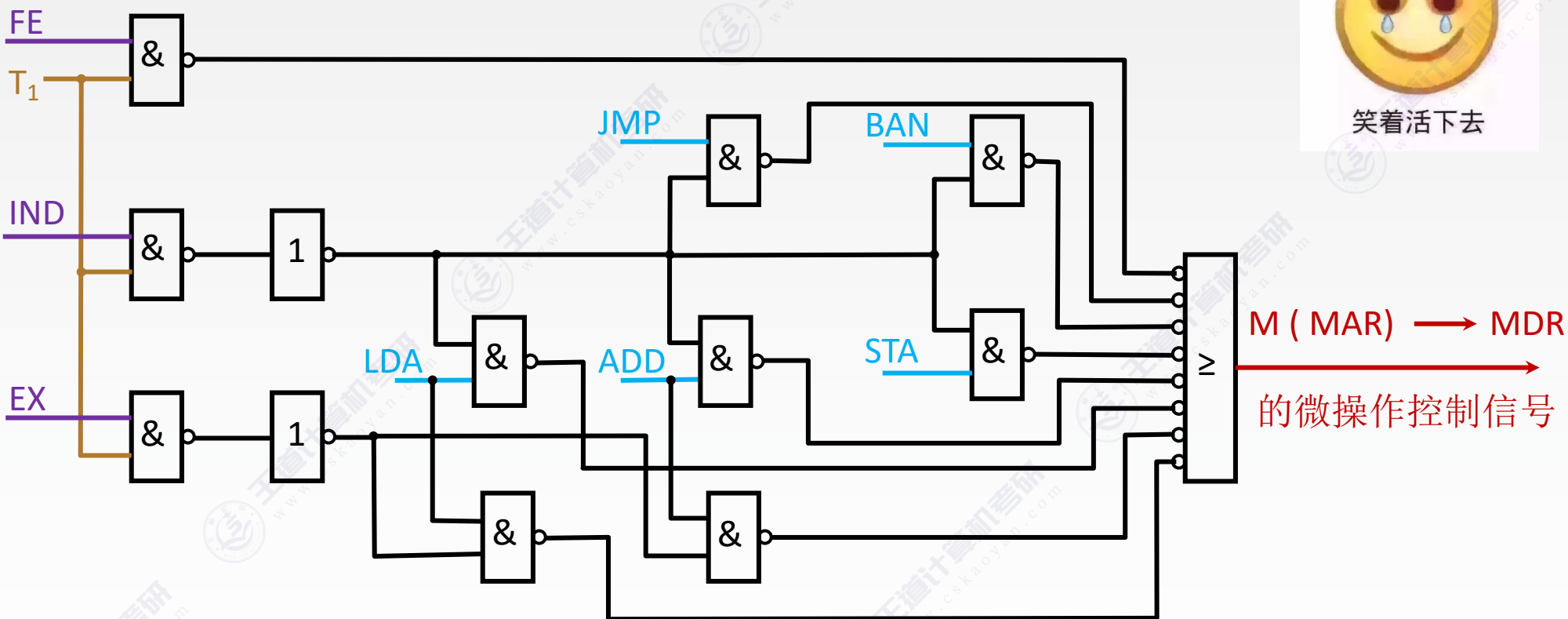
本节内容

微程序控制器 的基本原理

本章总览



硬布线控制器的设计思路



笑着活下去

硬布线控制器：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

时序信息包含机
器周期、节拍

微程序控制器的设计思路

采用“存储程序”的思想，CPU 出厂前将所有指令的“微程序”存入“控制器存储器”中

高级语言代码
高级语言代码



禁止套娃

指令1

指令2

指令3

指令4

指令5

指令6

取指周期
(FE=1)

间指周期
(IND=1)

执行周期
(EX=1)

中断周期
(INT=1)

T_0 : 微操作1、微操作2
 T_1 : 微操作3
 T_2 : 微操作4

T_0 : 微操作5、微操作2
 T_1 : 微操作6
 T_2 : 微操作7

T_0 : 微操作7
 T_1 : 微操作8
 T_2 : 微操作9、微操作6

T_0 : 微操作5、微操作2
 T_1 : 微操作10
 T_2 : 微操作11

微指令a: 完成微操作1、2
微指令b: 完成微操作3
微指令c: 完成微操作4

微指令d: 完成微操作5、2
微指令e: 完成微操作6
微指令f: 完成微操作7

微指令f: 完成微操作7
微指令g: 完成微操作8
微指令h: 完成微操作9、6

微指令d: 完成微操作5、2
微指令i: 完成微操作10
微指令j: 完成微操作11

微程序

程序：由指令序列组成

微程序：由微指令序列组成，每一种指令对应一个微程序

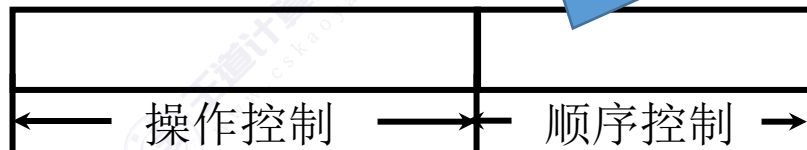
指令是对程序执行步骤的描述

微指令是对指令执行步骤的描述

指令是对微指令功能的“封装”

微指令基本格式

指明下一条微指令的地址



微命令与微操作一一对应
微指令中可能包含多个微命令

地址

P
K
指令1的微程序
指令2的微程序
微指令序列

微程序控制器的基本结构

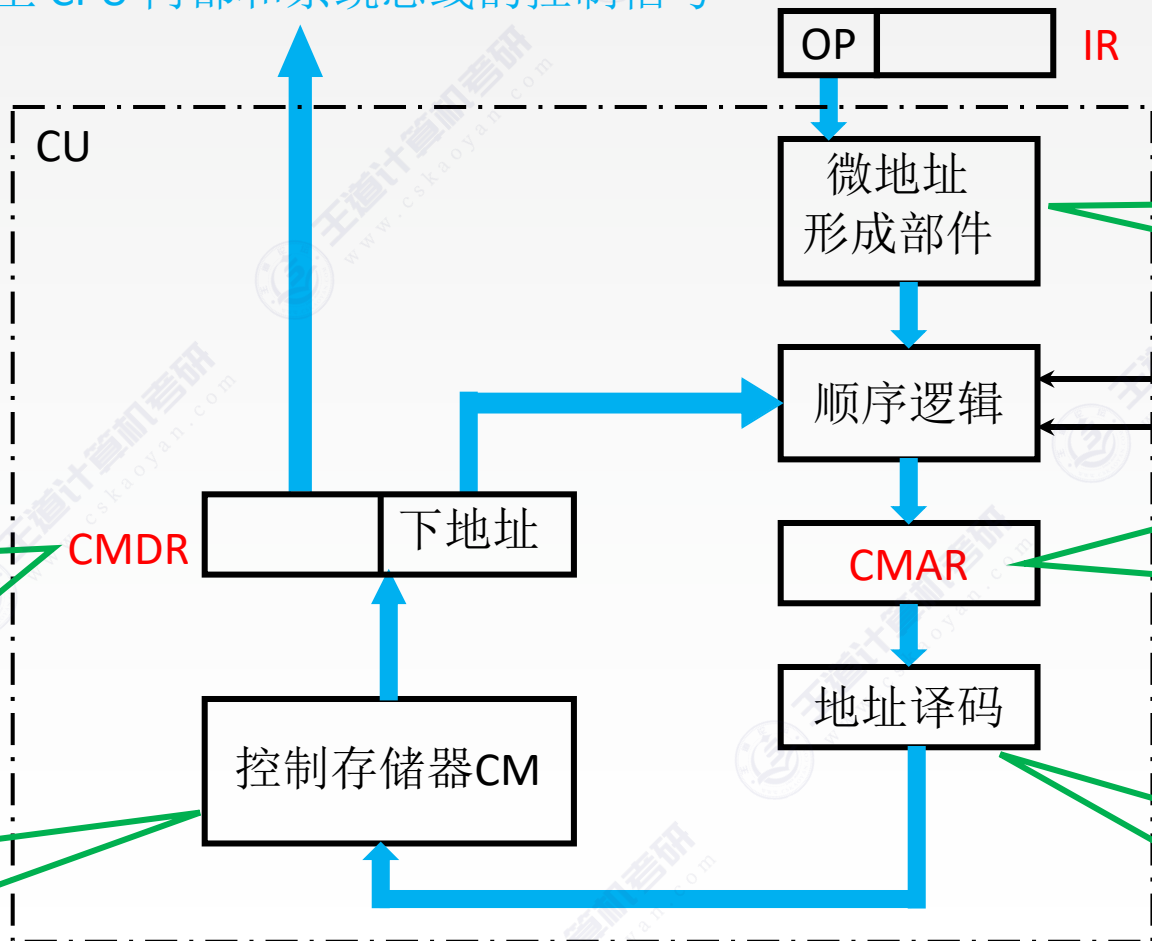
思考：所有指令的取指周期、间址周期、中断周期所对应的微指令序列都一样，是否可以共享使用？

至 CPU 内部和系统总线的控制信号

CM（按地址寻访）

别名：μIR，用于存放从CM中取出的微指令，它的位数同微指令字长相等。

用于存放各指令对应的微程序，控制存储器可用只读存储器ROM构成。



产生初始微地址和后继微地址，以保证微指令的连续执行。

标志
CLK

别名：μPC，微地址寄存器，接收微地址形成部件送来的微地址，为在CM中读取微指令作准备。

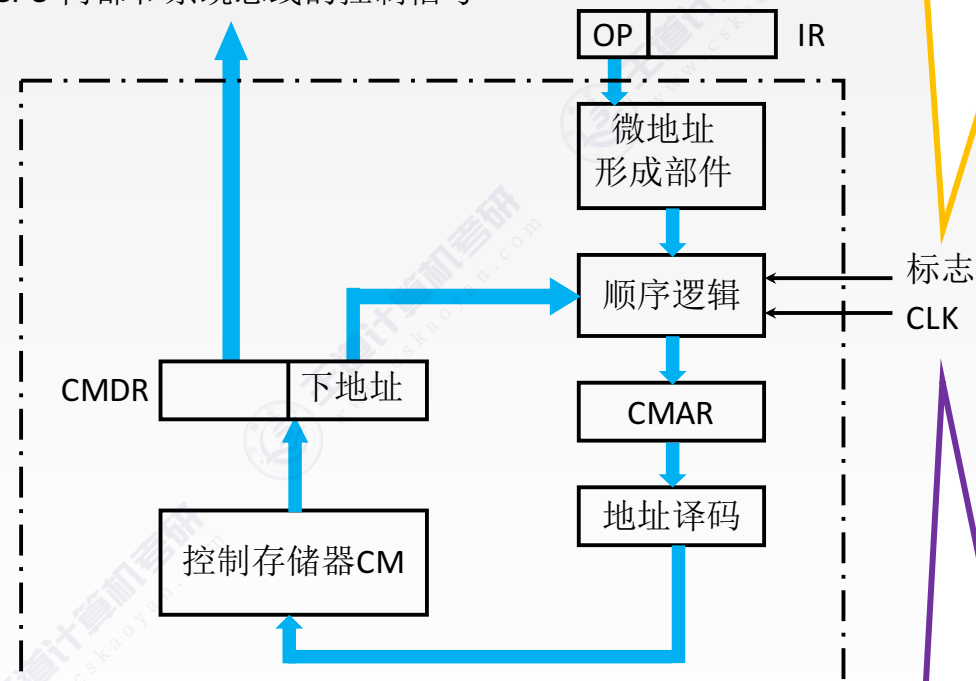
将地址码转化为存储单元控制信号。

微指令
存放地
址

微程序控制器的工作原理

微指令操作码	下地址	
	1	取指周期 微程序
	2	
	3	
⋮		间址周期 微程序
转执行周期		
⋮		
转取指周期		
⋮		中断周期 微程序
转取指周期		
⋮		
⋮		
	14	对应 LDA 指令的 执行周期微程序
	15	
	0	
	17	对应 STA 指令的 执行周期微程序
	18	
	0	
⋮		

至 CPU 内部和系统总线的控制信号



根据指令地址
码的寻址特征
位判断是否要
跳过间址周期

取数指令 LDA X

取指周期: #0、#1、#2

间址周期: #3、#4 #7

执行周期: #13、#14、#15

中断周期: #8、#9 #12

根据中断信号
判断是否进入
中断周期

微指令
存放地
址

微程序控制器的工作原理

微指令操作码	下地址
	1
	2
	3
⋮	
转执行周期	
⋮	
转取指周期	
	14
	15
	0
	17
	18
	0
⋮	

取指周期 微程序

间址周期 微程序

中断周期 微程序

对应 LDA 指令的
执行周期微程序

对应 STA 指令的
执行周期微程序

通常是公用的，故如果某指令系统中有 n 条机器指令，则CM中微程序的个数至少是 $n+1$ 个

一些早期的CPU、物联网设备的CPU可以不提供间接寻址和中断功能，因此这类CPU可以不包含间址周期、中断周期的微程序段

Tips: 物理上，取指周期、执行周期看起来像是两个微程序，但逻辑上应该把它们看作一个整体。因此，“一条指令对应一个微程序”的说法是正确的

微程序控制器

CU 的结构

- 微地址形成部件
 - 微地址即微指令在 CM 中的存放地址
 - 通过指令操作码形成对应微程序的第一条微指令的存放地址
- 顺序逻辑
 - 根据某些机器标志和时序信息确定下一条微指令的存放地址
- CMAR (μPC)
 - 指明接下来要执行的微指令的存放地址
- 地址译码器
 - 将 CMAR 内的地址信息译码为电信号, 控制 CM 读出微指令
- 控制存储器 CM
 - 存放所有机器指令对应的微程序 (微指令序列)
 - 用 ROM 实现, 按地址寻访。通常在 CPU 出厂时就把所有微程序写入
- CMDR (μIR)
 - 微指令寄存器, 用于存放当前要执行的微指令。CM(μPC) $\rightarrow \mu IR$

工作原理

- 指令周期 = 取值周期 \rightarrow 间址周期 \rightarrow 执行周期 \rightarrow 中断周期。其中间址、中断周期可有可无
- 处理取指周期、间址周期、中断周期的微指令序列通常是公用的。执行周期的微指令序列各不相同
- 取指周期的微指令序列固定从 #0 开始存放。执行周期的微指令序列的存放根据指令操作码确定

概念对比

- 程序 vs 微程序; 指令 vs 微指令; 主存储器 vs 控制器存储器 (CM); MAR vs CMAR; MDR vs CMDR; PC vs μPC ; IR vs μIR
- 微命令、微操作、微指令、微程序 之间的关系
- 指令周期: 从主存取出并执行一条机器指令所需的时间
- 微周期 (微指令周期): 从控制器存储器取出一条微指令并执行相应微操作所需的时间



公众号：王道在线



b站：王道计算机教育



抖音：王道计算机考研