

# 第四章 控制器原理与 CPU组织

本章主要讨论：

**CPU组成**

运算器

控制器

数据通路结构

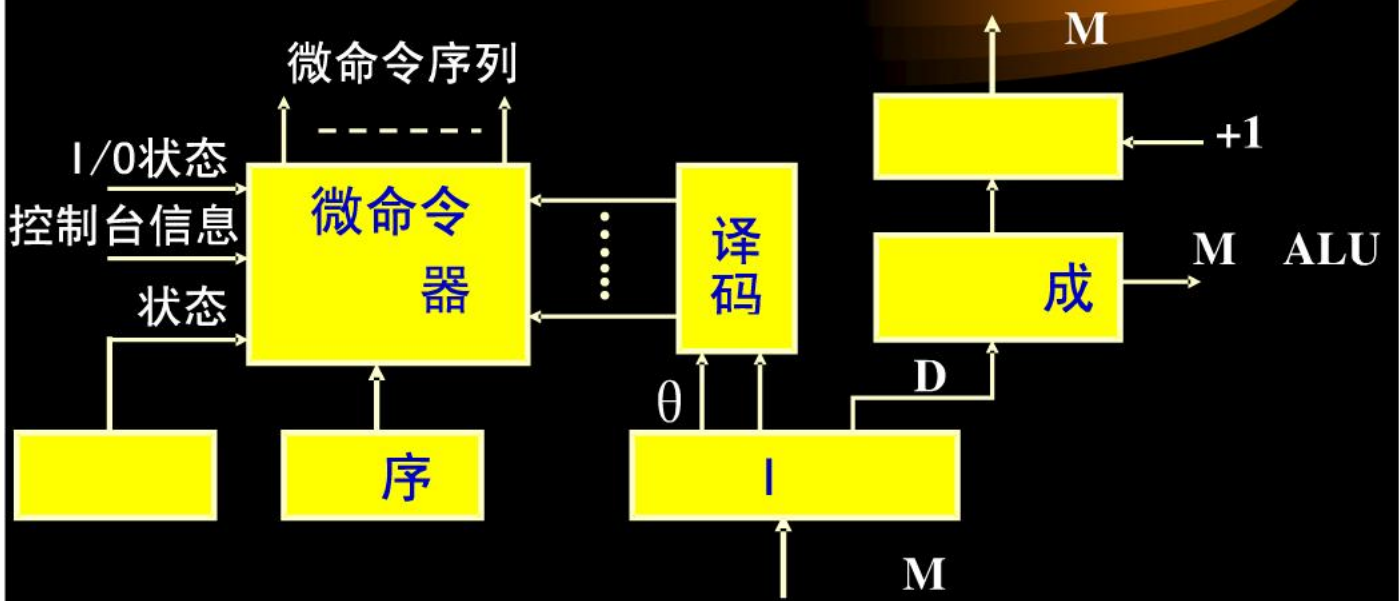
与外部的连接

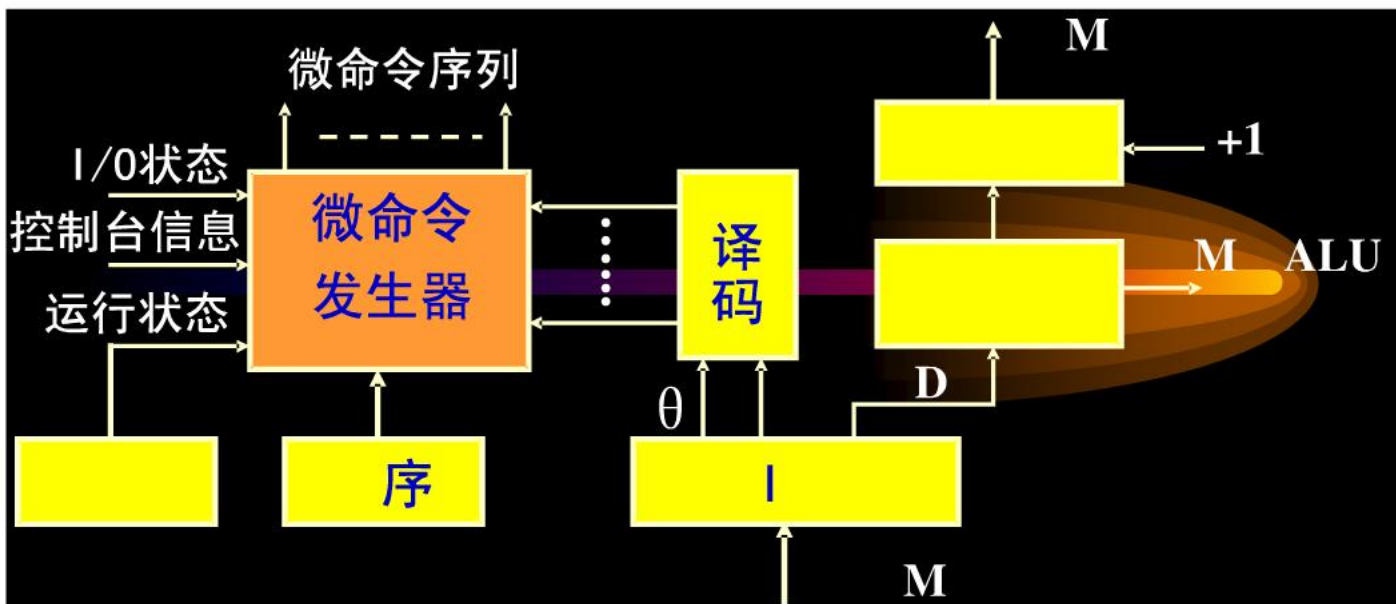
**CPU工作原理**

—— 指令的执行过程

# 第一节 组合逻辑控制器原理

## 4.1.1 控制器组成





微命令发生器  
生

控制

微命令发生器

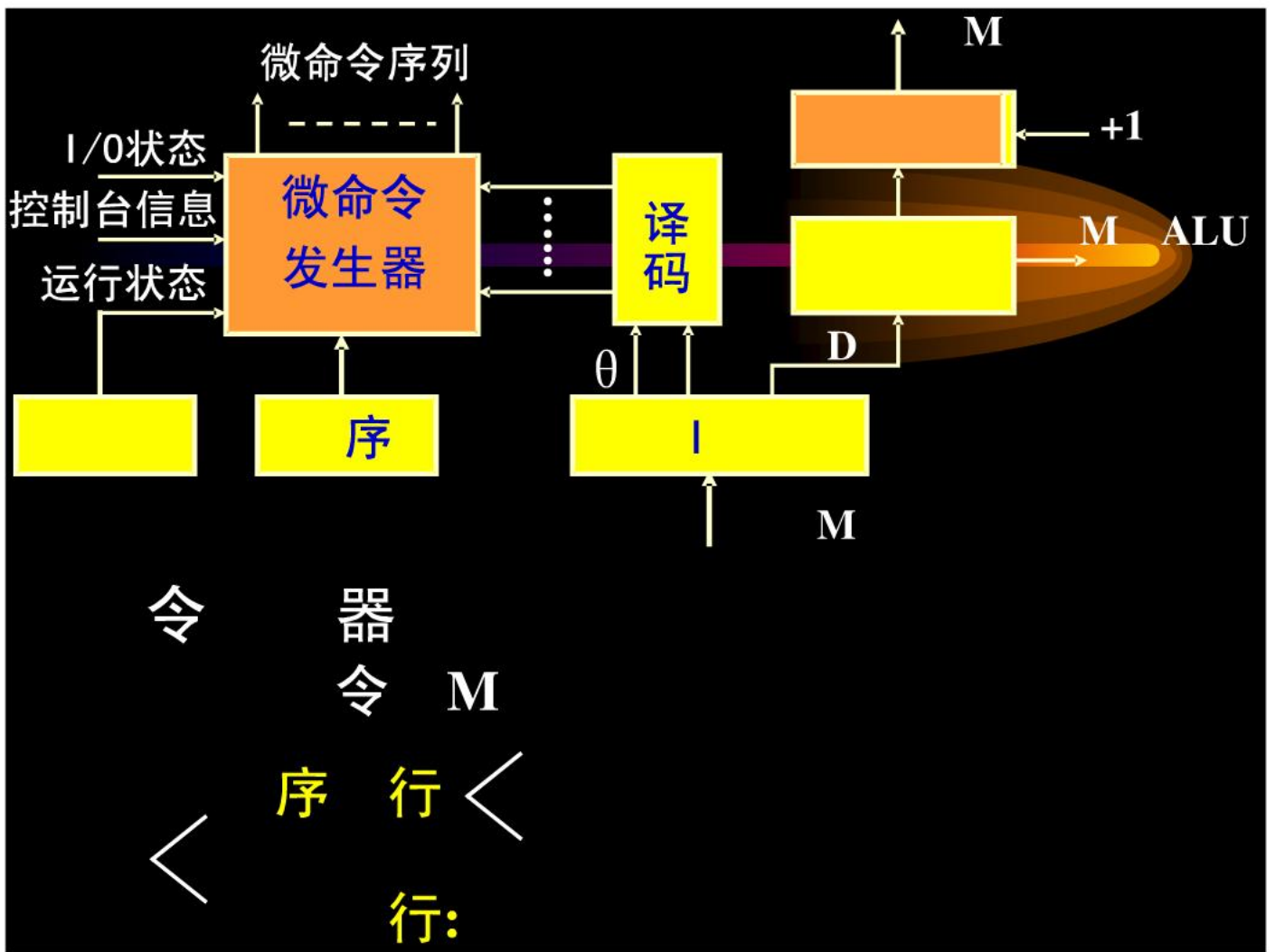
微命令

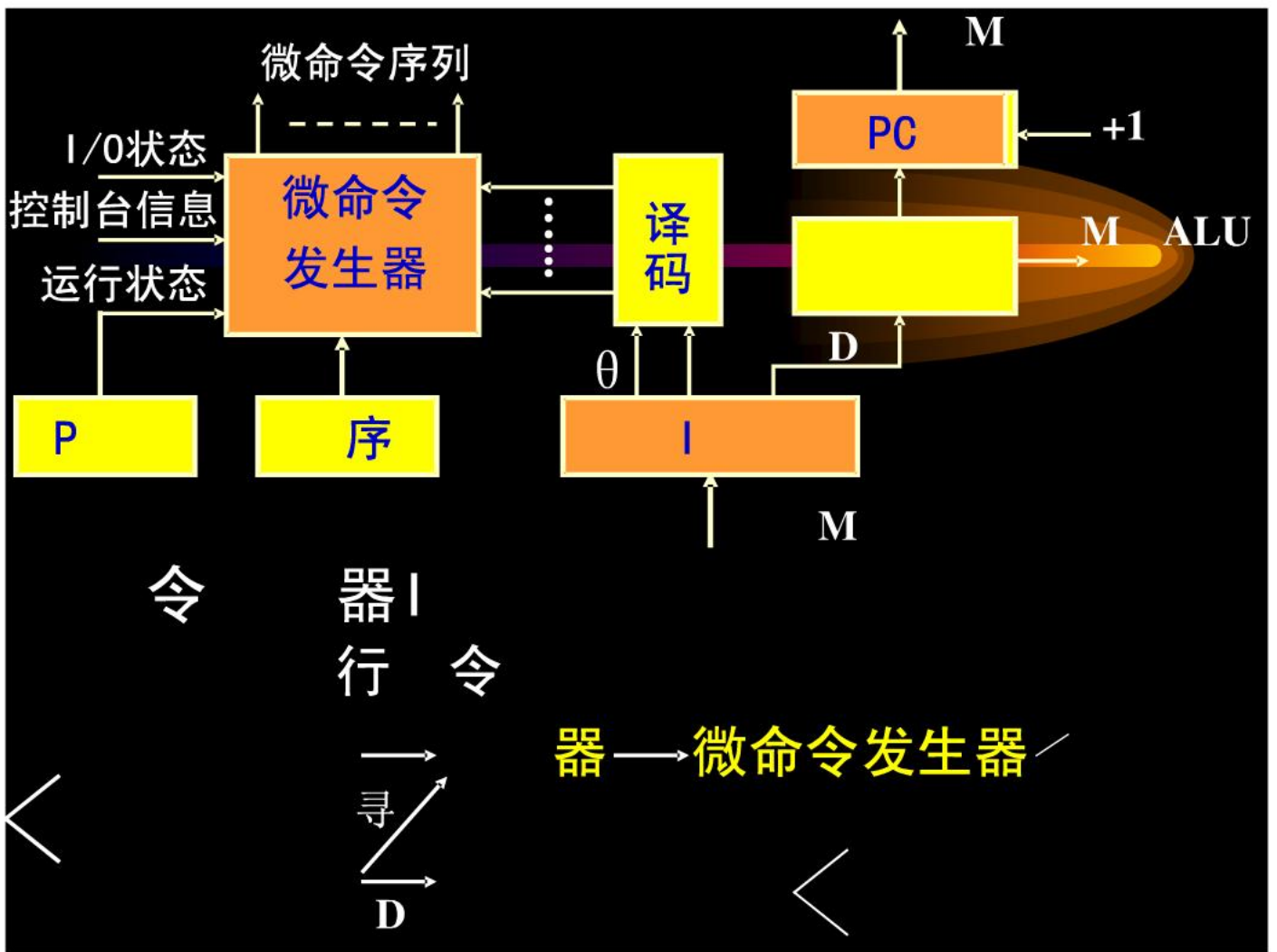
微

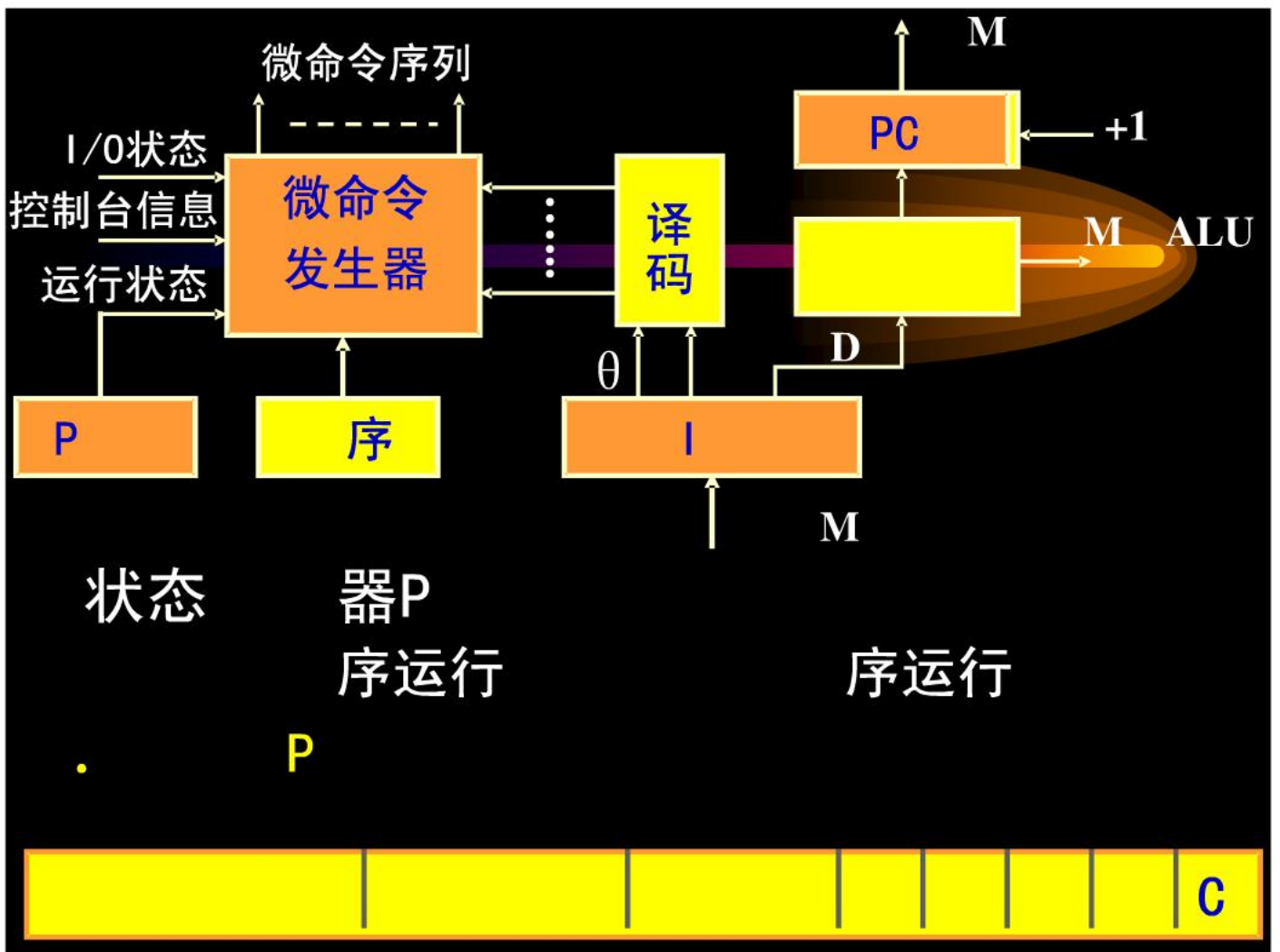
命令

生微命令









15 12 11 8 7 6 5 4 3 2 1 0

工作方式

优先级

T

N

Z

V

C

(1) 条件码

反映程序运行结果

(2) 跟踪标志

C=1 进位

V=1 溢出

Z=1 结果为0

N=1 结果为负

为程序查错设置的断点标志T。

T=1, 执行跟踪程序

程序

初始化置T为1

⋮

测试T

跟踪程序

⋮

⋮

⋮

⋮



### (3) 优先级

为现行程序赋予优先级别，以决定是否响应外部中断请求。

- { 程序优先级高于外部优先级，不响应
- { 程序优先级低于外部优先级，可响应

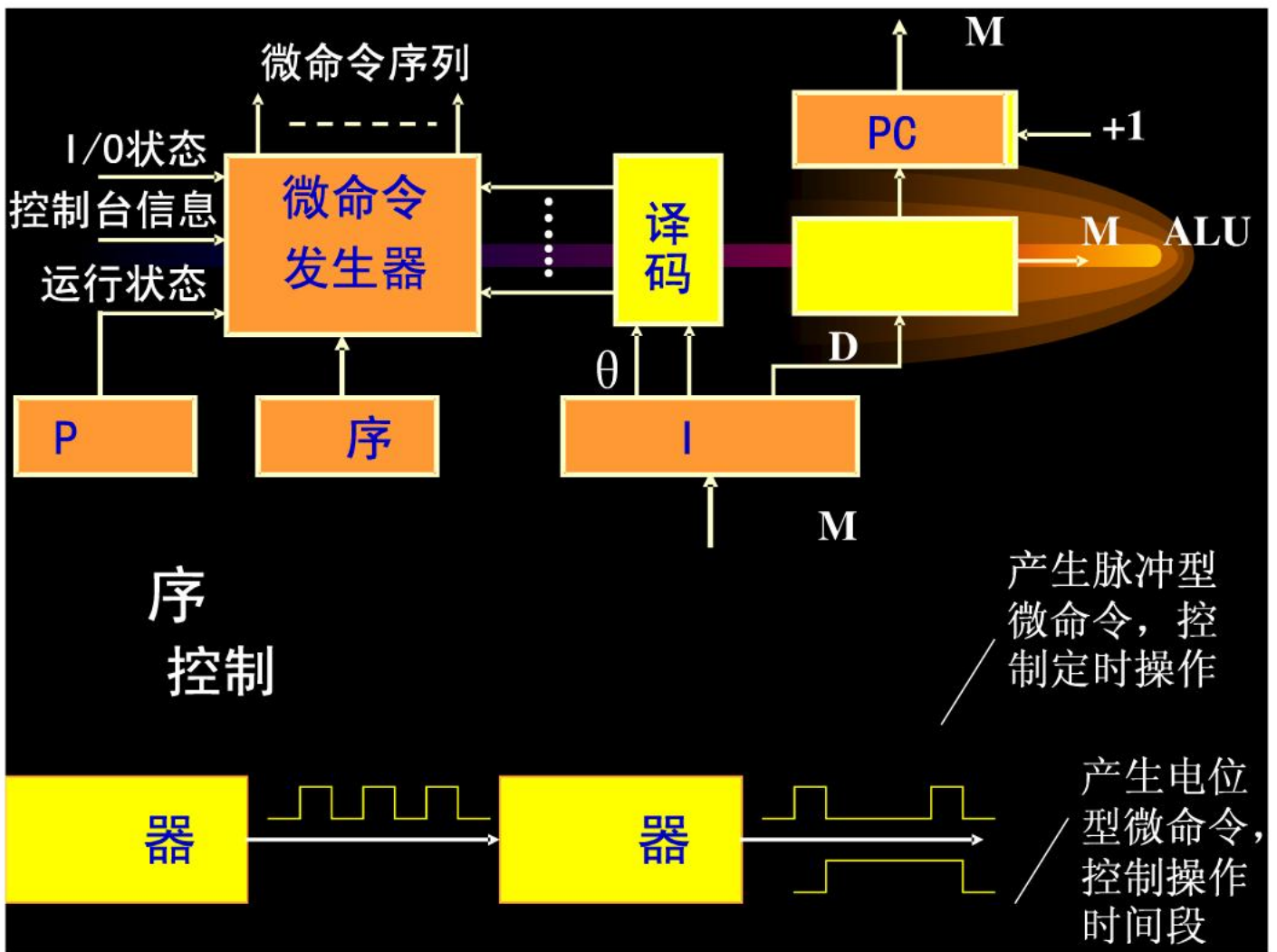
### (4) 工作方式

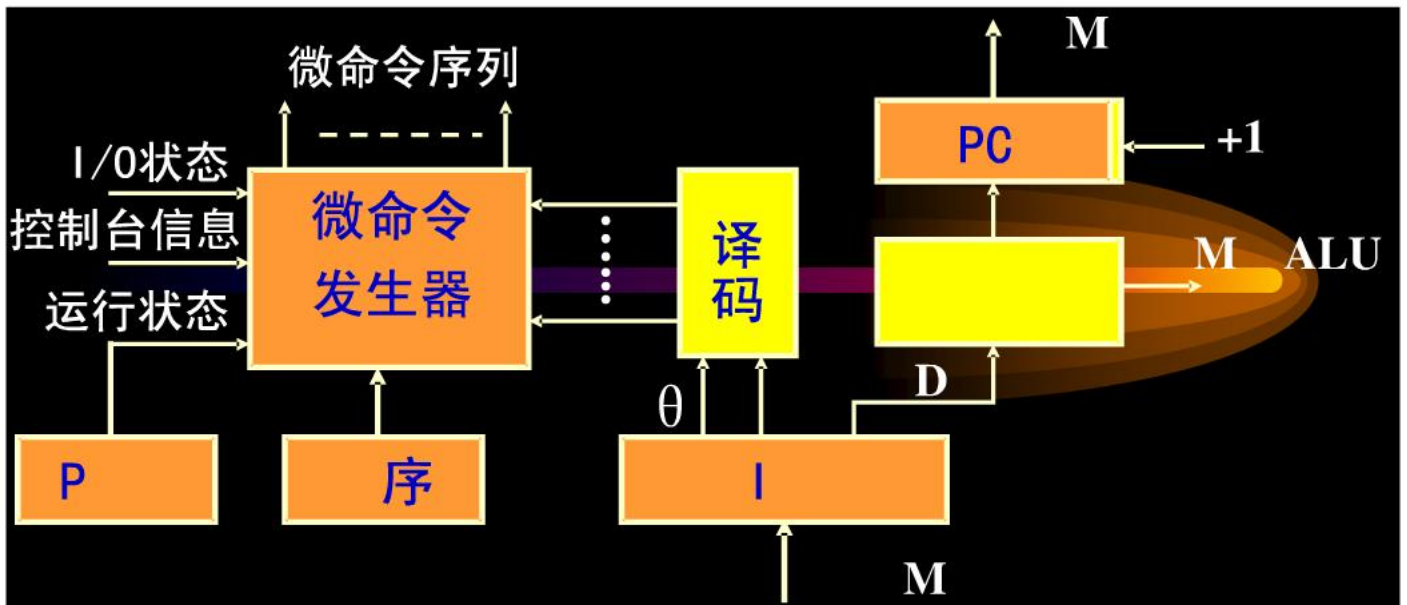
规定程序的特权级。

- { 用户方式：禁止程序执行某些指令
- { 核心方式：允许程序执行所有指令

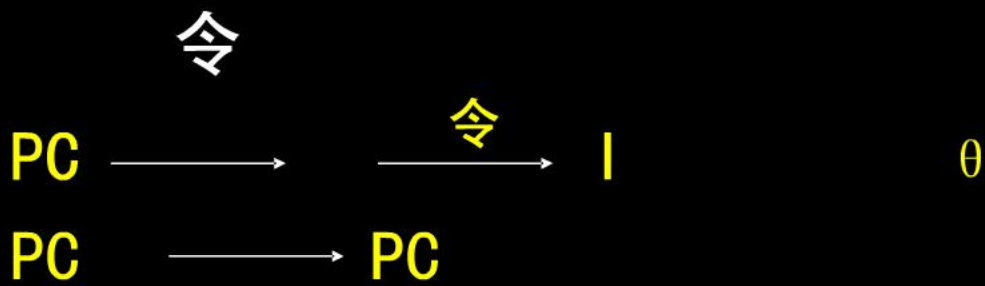
PSW在CPU中，反映程序运行状态；控制/状态字在接口中，反映CPU命令、设备状态。

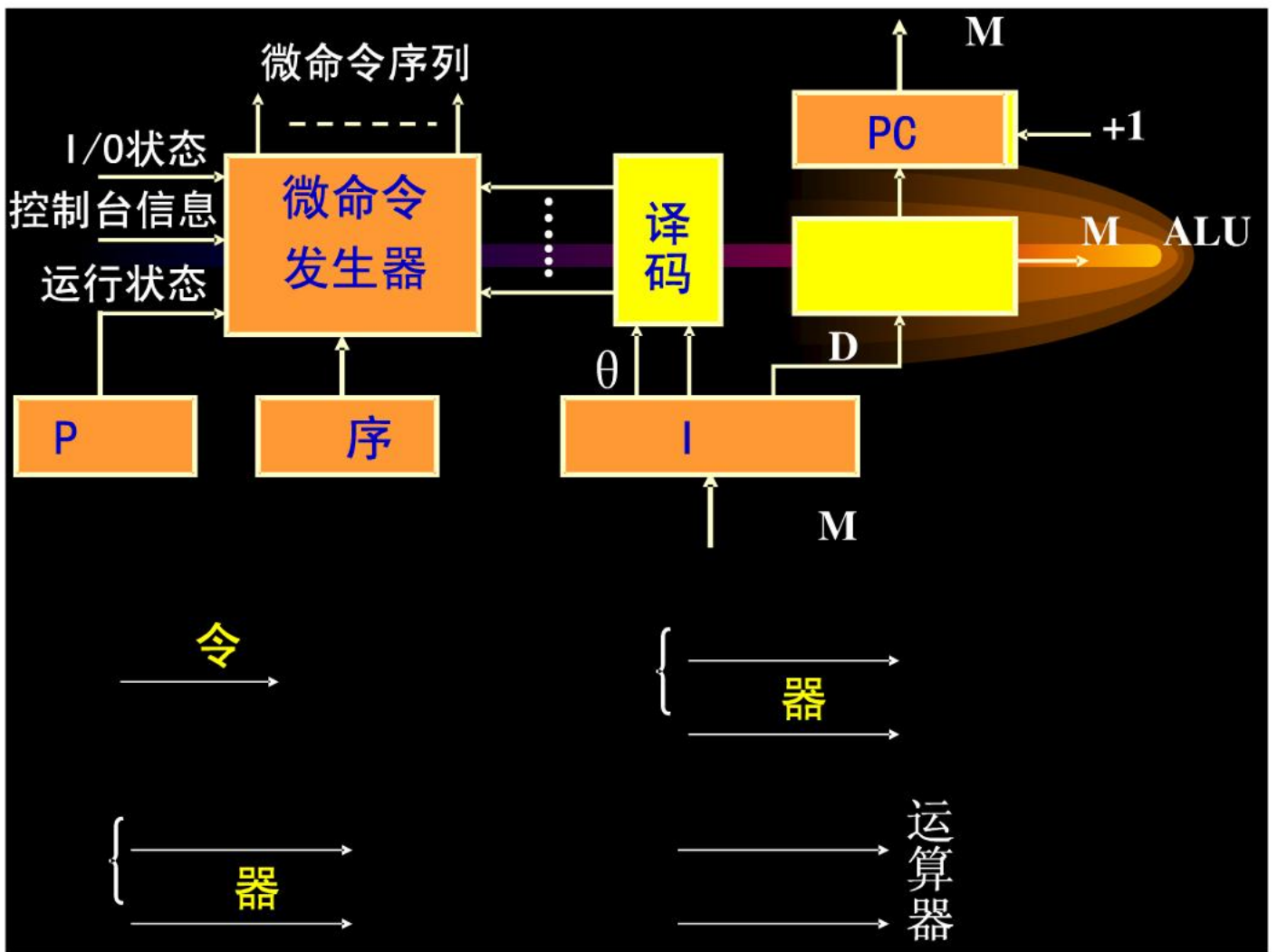


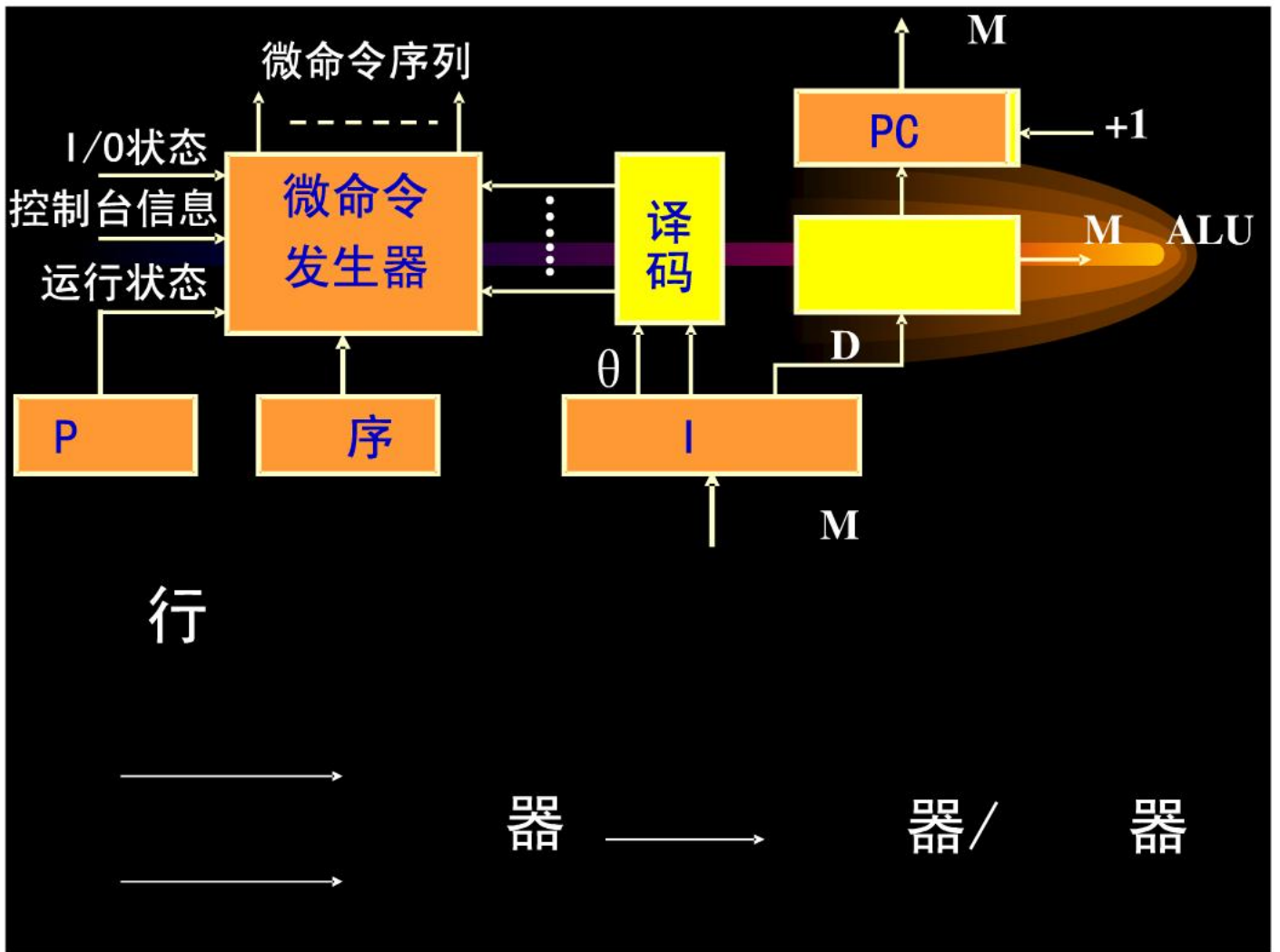




## 4.1.2 控制器





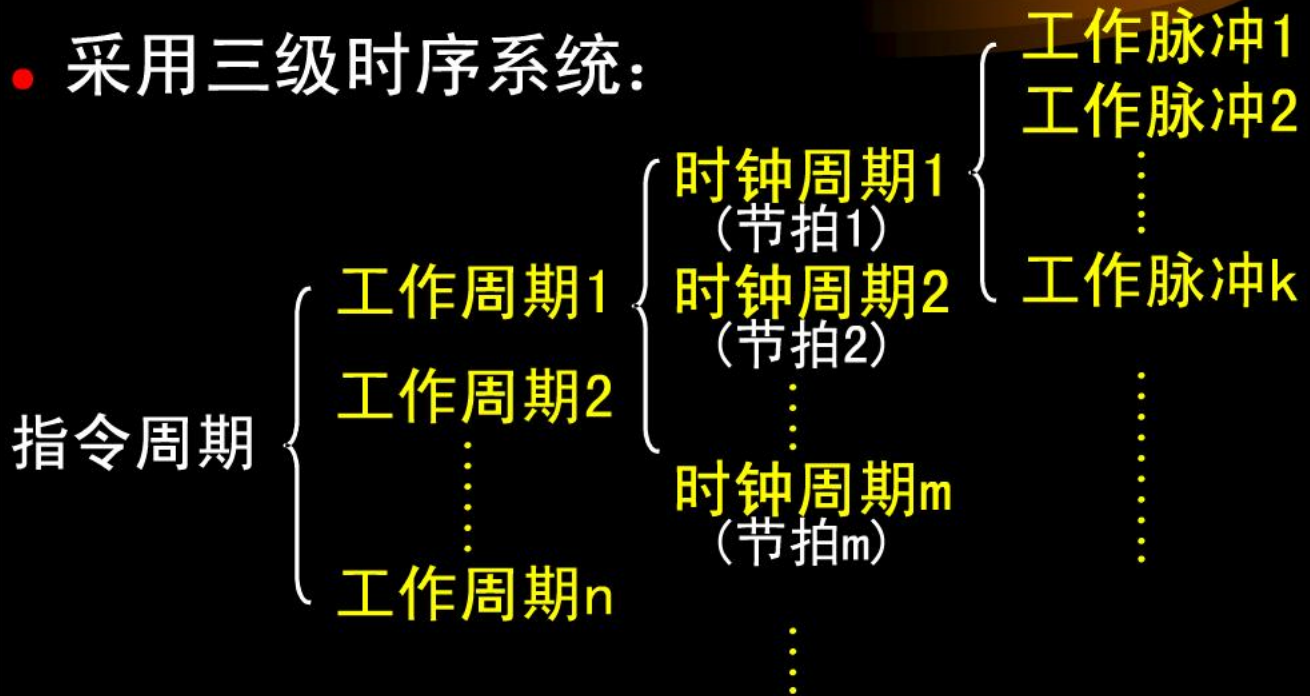


### 4.1.3 时序控制方式

#### 即时序信号与操作的关系

#### 1. 组合逻辑控制器的时序划分

- 采用三级时序系统:



# • 时序关系:

晶振输出

工作脉冲P — 对微操作定时

打入IR 打入PC

时钟T1 — 控制分步操作时间

取出指令

时钟T2

修改PC

工作周期1 — 控制不同阶段操作时间

取指

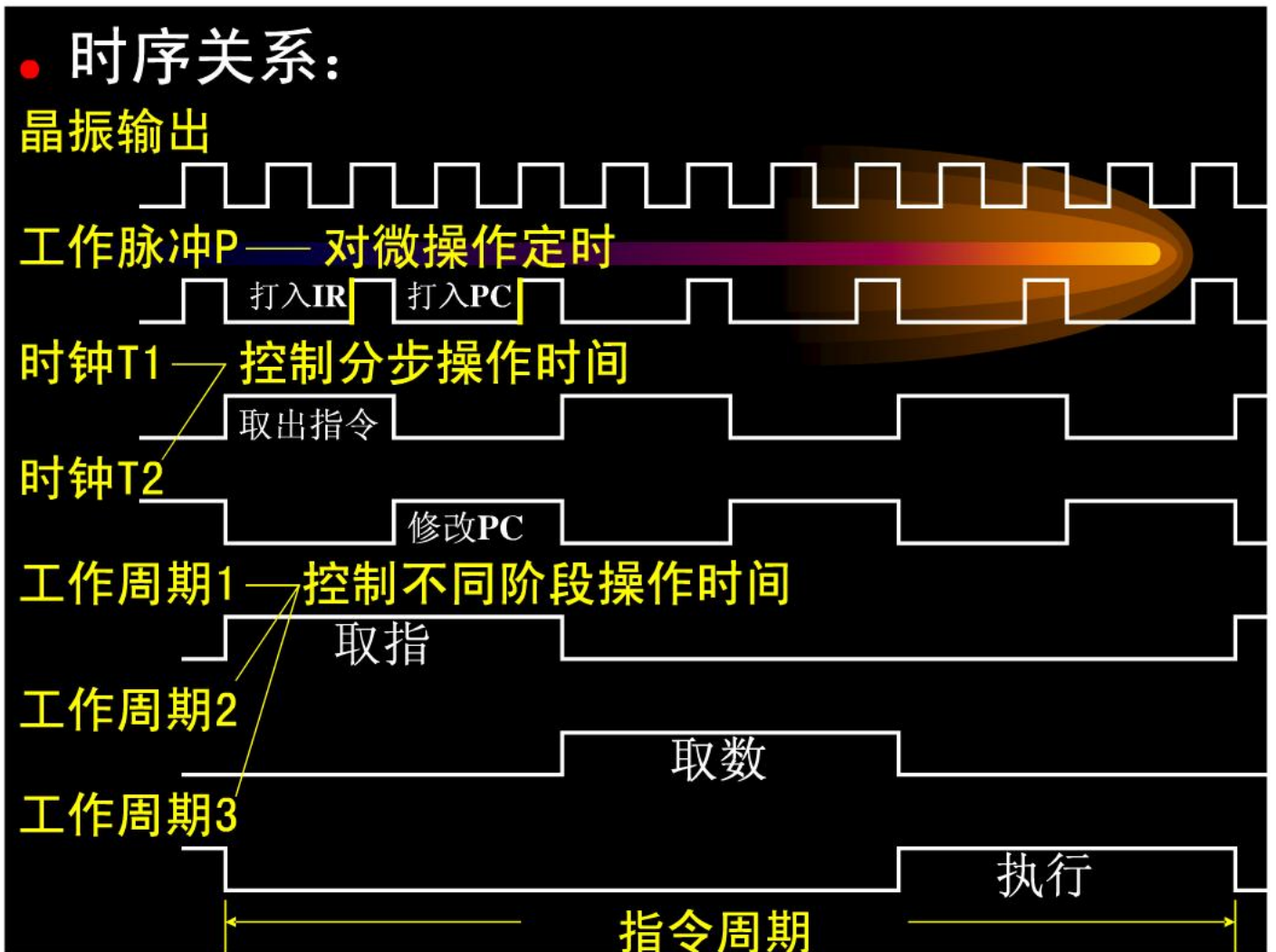
工作周期2

取数

工作周期3

执行

指令周期



## 2. 时序控制方式及其变化

### (1) 同步控制

由CPU或其他设备提供

①定义：各项操作受统一时序控制。

②特点：有明显时序时间划分，时钟周期时间固定，各步操作的衔接、各部件之间的数据传送受严格同步定时控制。

③优缺点：时序关系简单，时序划分规整，控制不复杂；控制逻辑易于集中，便于管理。  
时间安排不合理。

④应用场合：用于CPU内部、设备内部、系统总线操作（各挂接部件速度相近，传送时间确定，传送距离较近）。

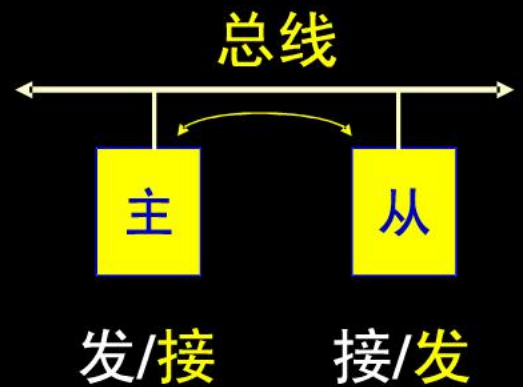
## (2) 异步控制

①定义：各项操作按不同需要安排时间，不受统一时序控制。

②特点：无统一时钟周期划分，各操作间的衔接和各部件之间的信息交换采用异步应答方式。

例. 异步传送操作

- 主设备：  
申请并掌握总线权的设备。
- 从设备：  
响应主设备请求的设备。





● 操作流程:

主设备输出端与总线连接

主设备获得总线控制权

主设备询问从设备

从设备准备好?

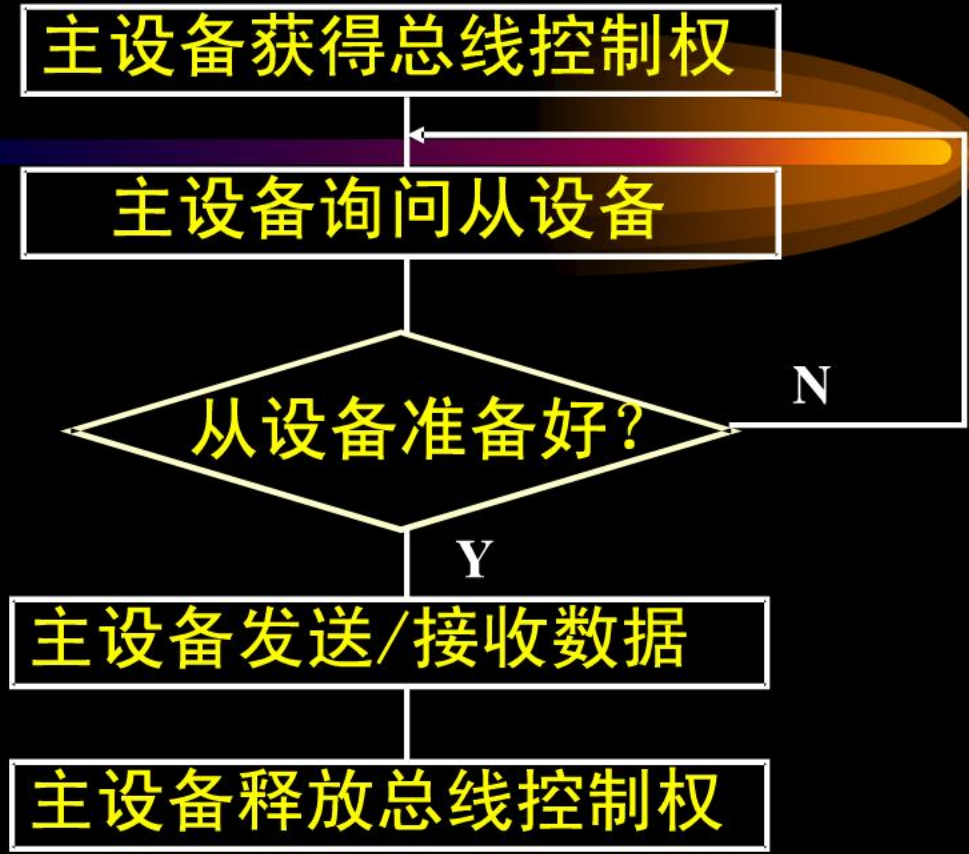
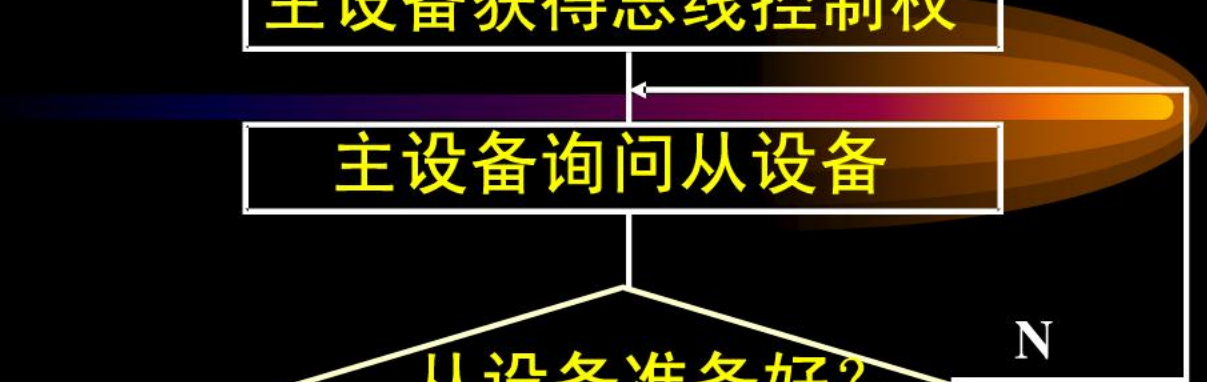
N

Y

主设备发送/接收数据

主设备释放总线控制权

主设备输出端与总线断开



③优缺点： 时间安排紧凑、合理；  
控制复杂。

④应用场合： 用于异步总线操作（各挂接部件速度差异大，传送时间不确定，传送距离较远）。

### （3）同步方式的变化

①不同指令安排不同时钟周期数  
指令周期长度可变，时钟周期长度不变。  
可用计数器指示时钟周期数的变化。

②总线周期中插入延长周期

经总线传送一次数据所用的时间（送地址、读/写）

总线周期长度可变，时钟周期长度不变。

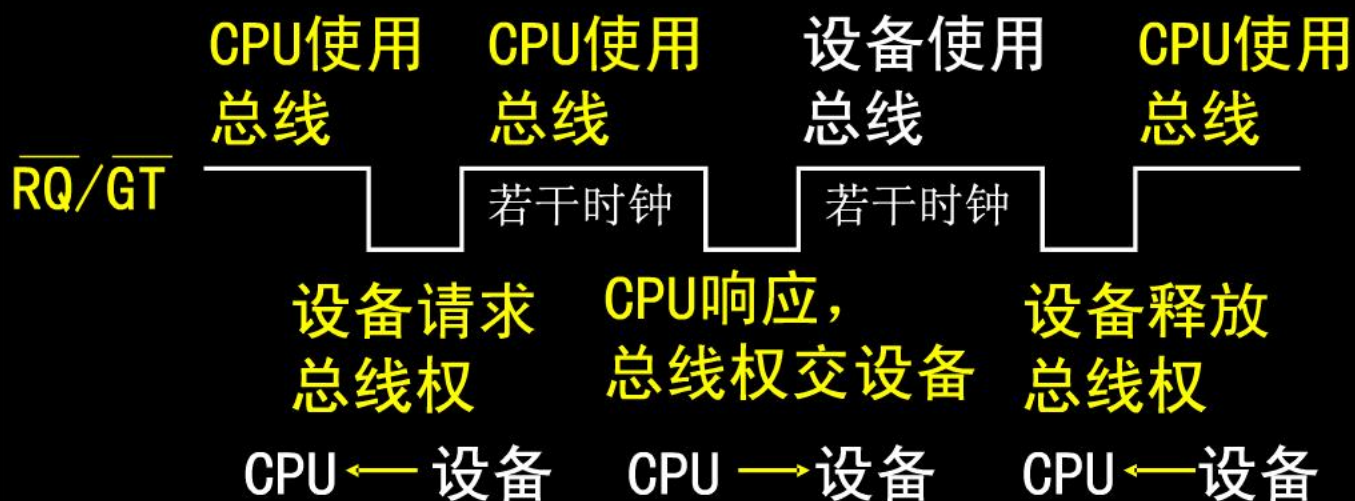
例. 一个总线周期包含4个时钟周期



### ③同步方式引入异步应答

以固定时钟周期作为时序基础，引入应答思想。

例. 8088最大模式，用一根总线请求/应答线实现总线权的转移。



## 4.1.4 组合逻辑控制方式的优缺点及应用

### 1. 组合逻辑控制方式

综合化简产生微命令的条件，形成逻辑式，用组合逻辑电路实现；

执行指令时，由组合逻辑电路(微命令发生器)在相应时间发出所需微命令，控制有关操作。

### 2. 优缺点

- 产生微命令的速度较快。
- 设计不规整，设计效率较低；  
控制器核心结构零乱，不便于检查和调试。

## 2. 优缺点

- 产生微命令的速度较快。
- 设计不规整，设计效率较低；  
控制器核心结构零乱，不便于检查和调试。
- 不易修改、扩展指令系统功能。

## 3. 应用场合

用于高速计算机，或小规模计算机。