



主要内容

1

概述

2

数字程控交换机的构成

3

呼叫处理原理

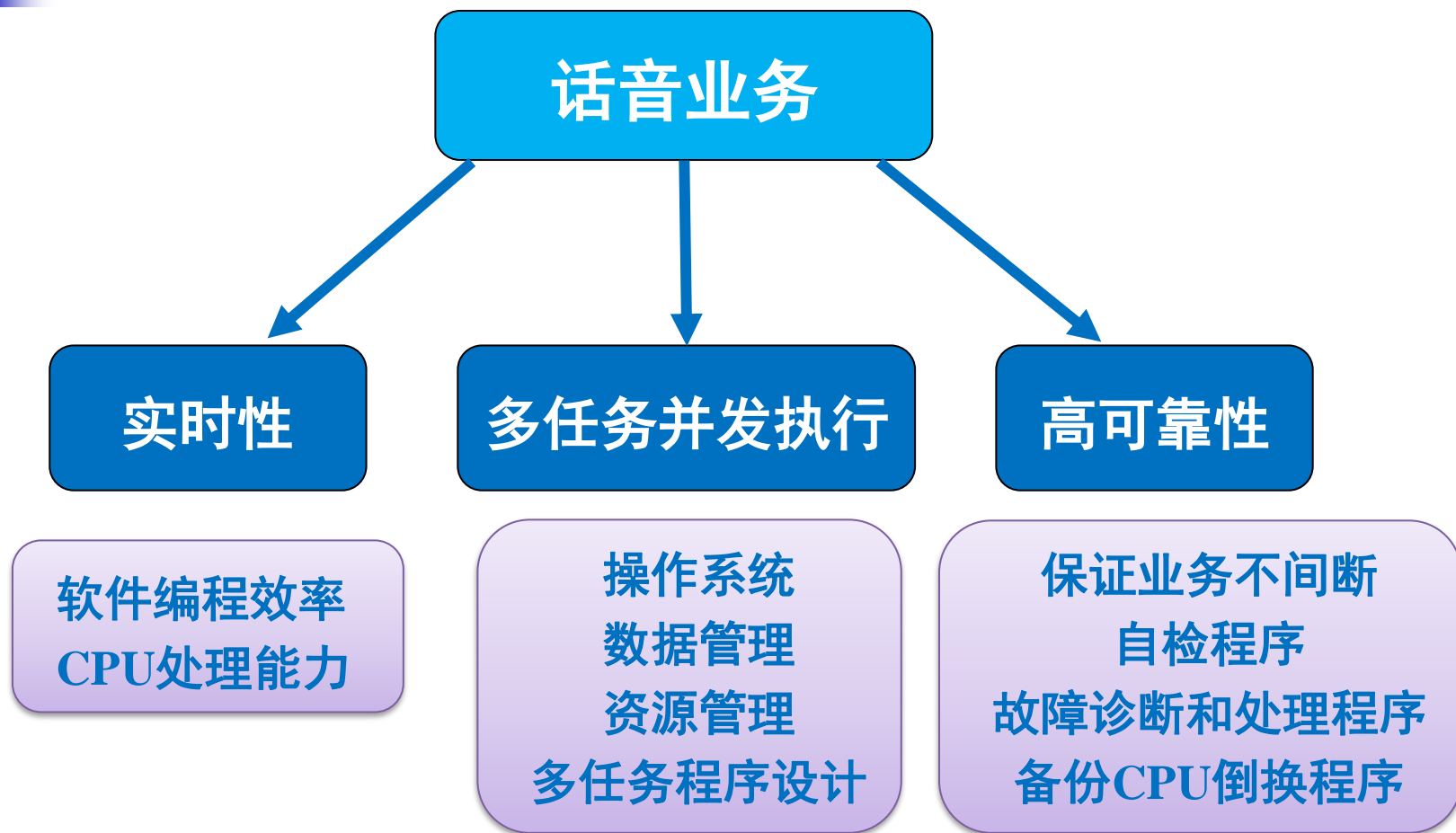


呼叫处理原理

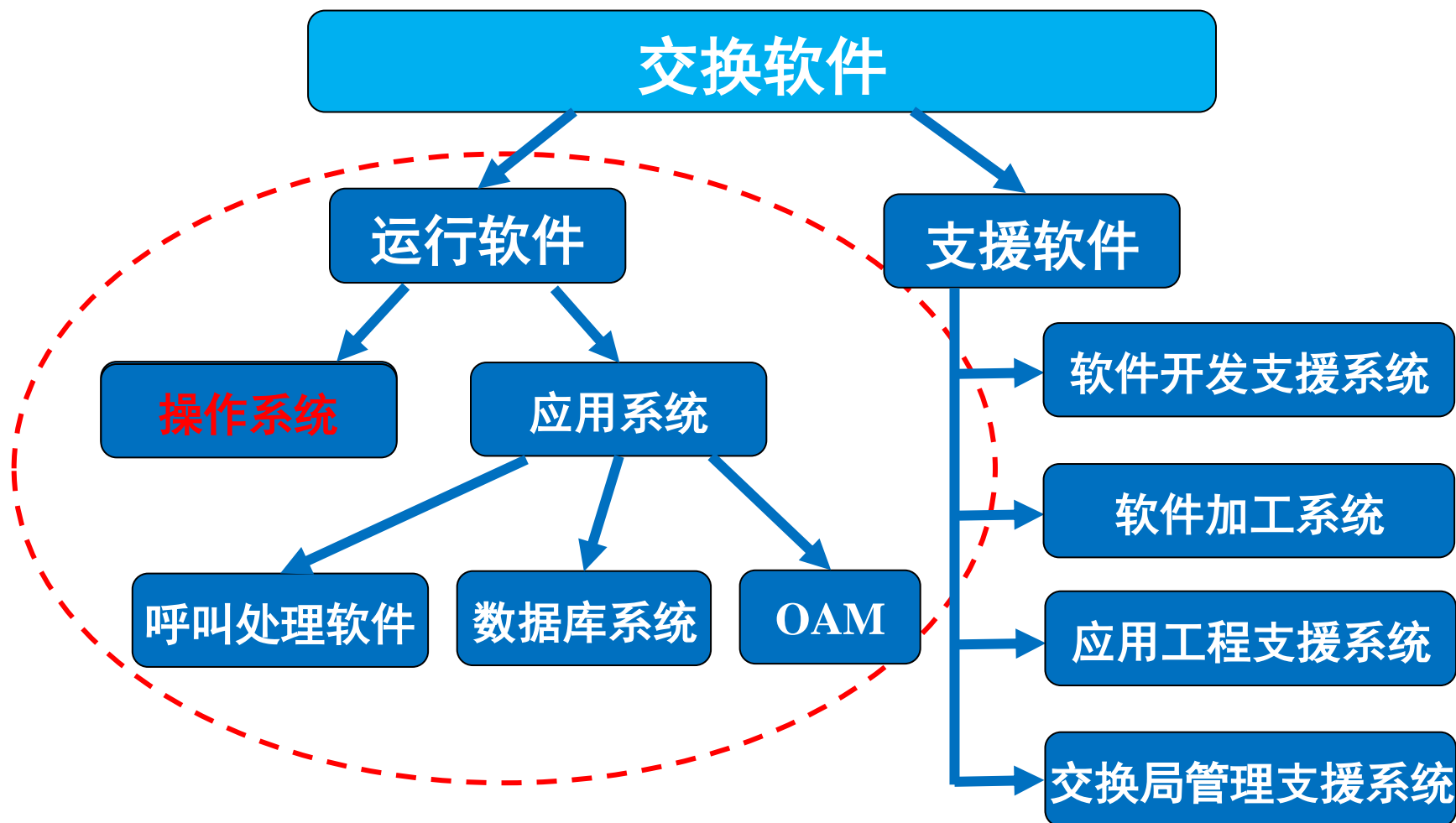
- 程控交换软件概述
- 扩展有限状态机&SDL
- 呼叫处理原理
- 交换软件设计技术

程控交换

交换软件特点



交换软件系统组成





程控交换机的操作系统

交换机的操作系统？

你知道哪些操作系统？

实时多任务操作系统

实时性

多任务性

实时多任务操作系统支持多任务（Task）并发处理，由于多任务的并发性因而必然会引起任务的同步、互斥、通信以及资源共享等问题。

任务的分级

故障级任务

负责故障识别和紧急处理等功能，具有最高优先级

故障

周期级任务

由时钟中断按周期性启动的任务。如每隔10ms周期性启动的拨号脉冲识别程序，启动周期为100ms的用户群扫描程序等。周期级任务的优先级较故障级任务低、比基本级任务高。

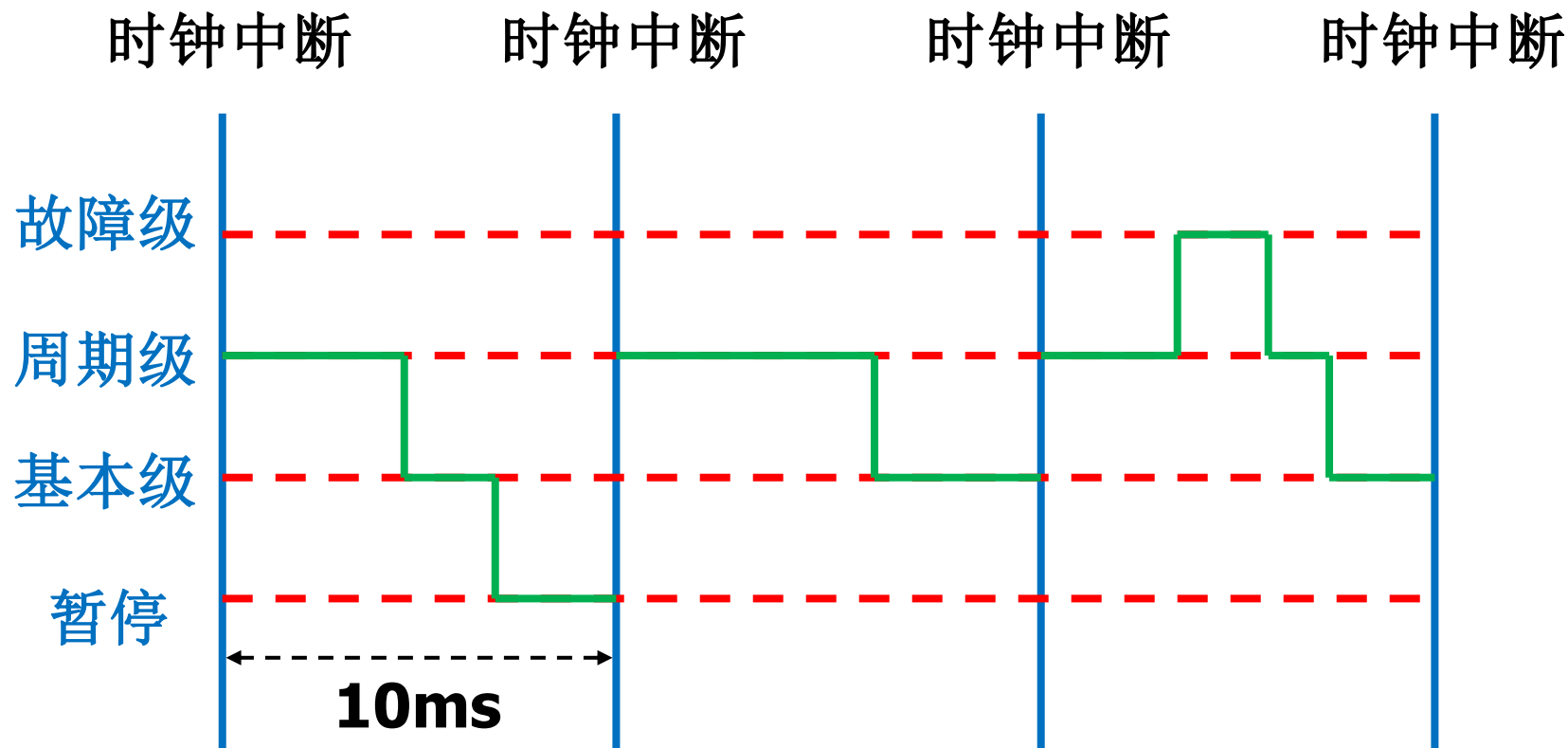
时钟中断

基本级任务

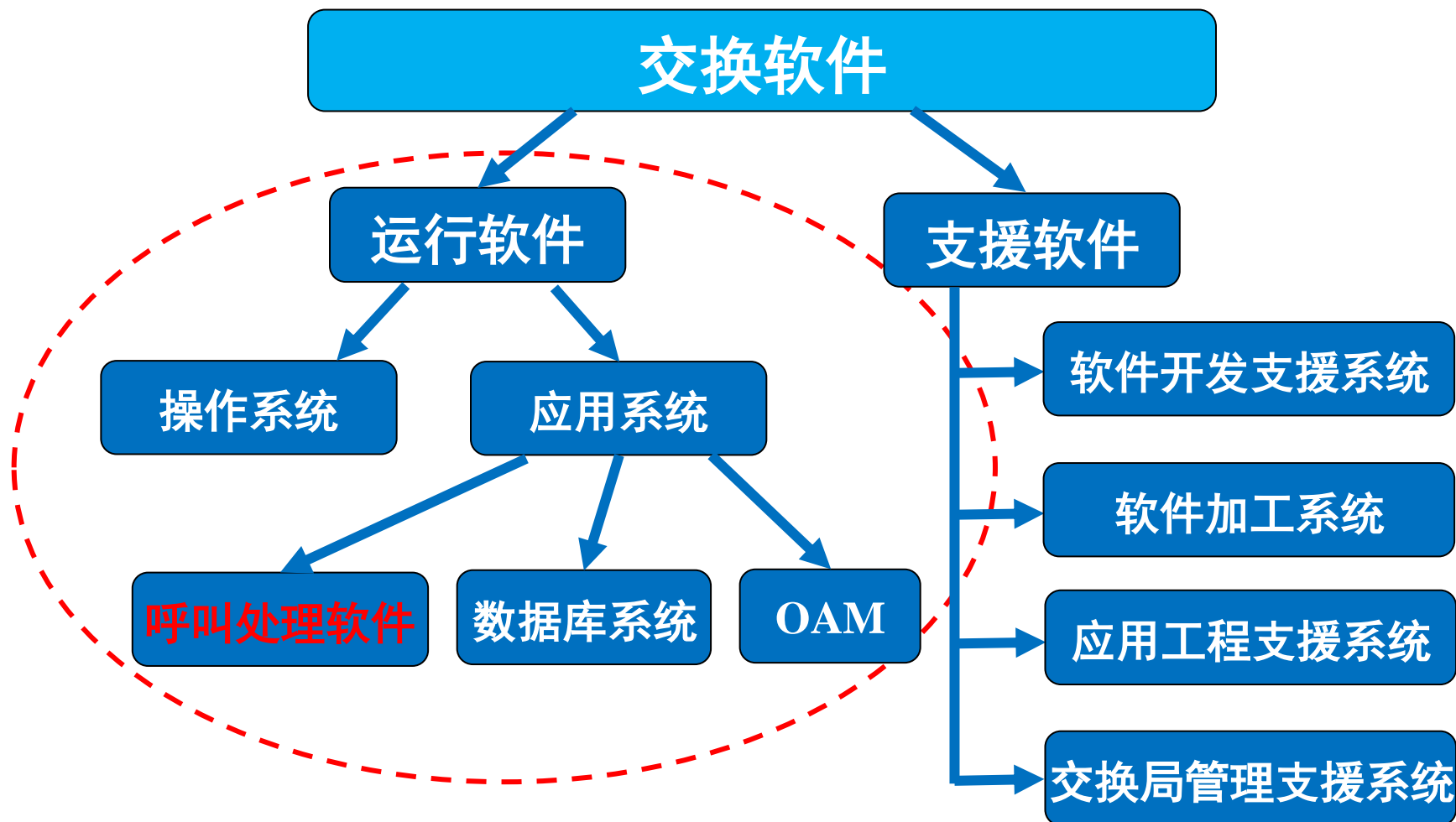
由事件启动的实时性要求不高、可以适当延迟执行的任务，其优先级最低。

事件

不同优先级的任务（举例）



交换软件系统组成



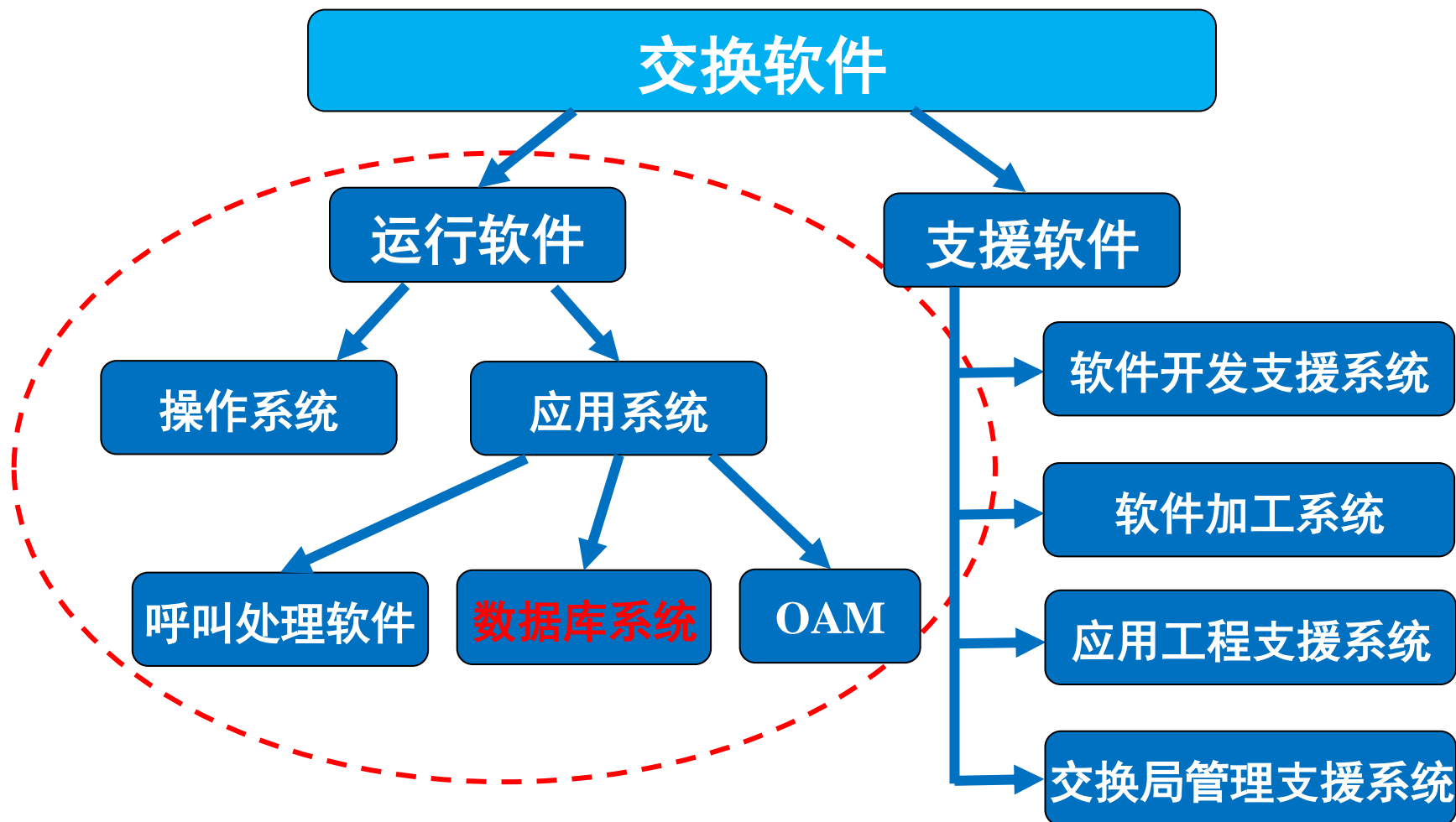


呼叫处理软件

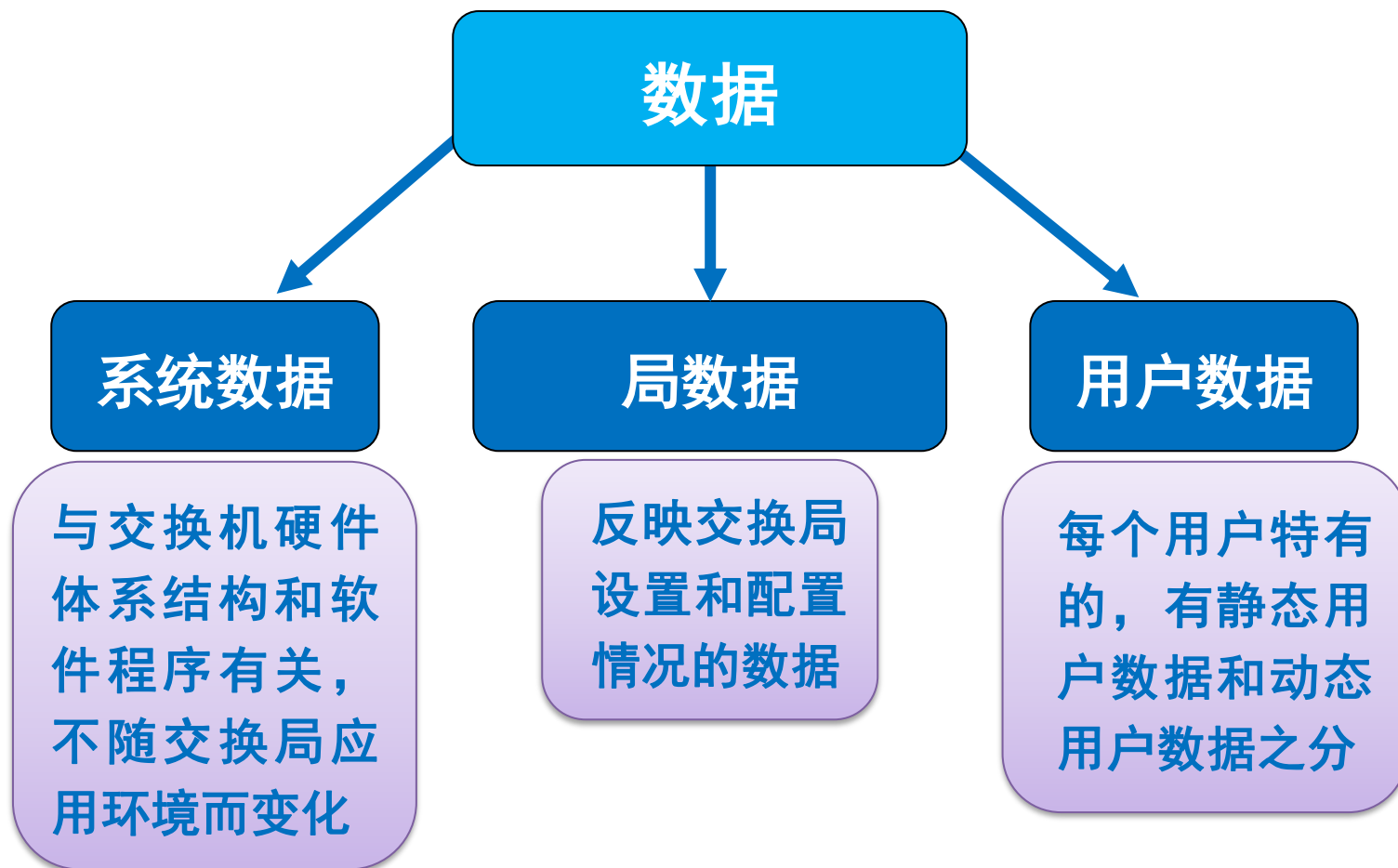
具体完成的**功能**如下：

- 用户线和中继线上输入信号（呼叫信号、地址信号）检测识别
- 呼叫资源管理（时隙、中继电路、DTMF收号器）
- 用户数据、呼叫状态以及号码等的分析
- 路由选择
- 控制呼叫状态迁移
- 控制计时、送音和交换网络的连接；
- 信令协议的处理等

交换软件系统组成



数据库系统





用户数据

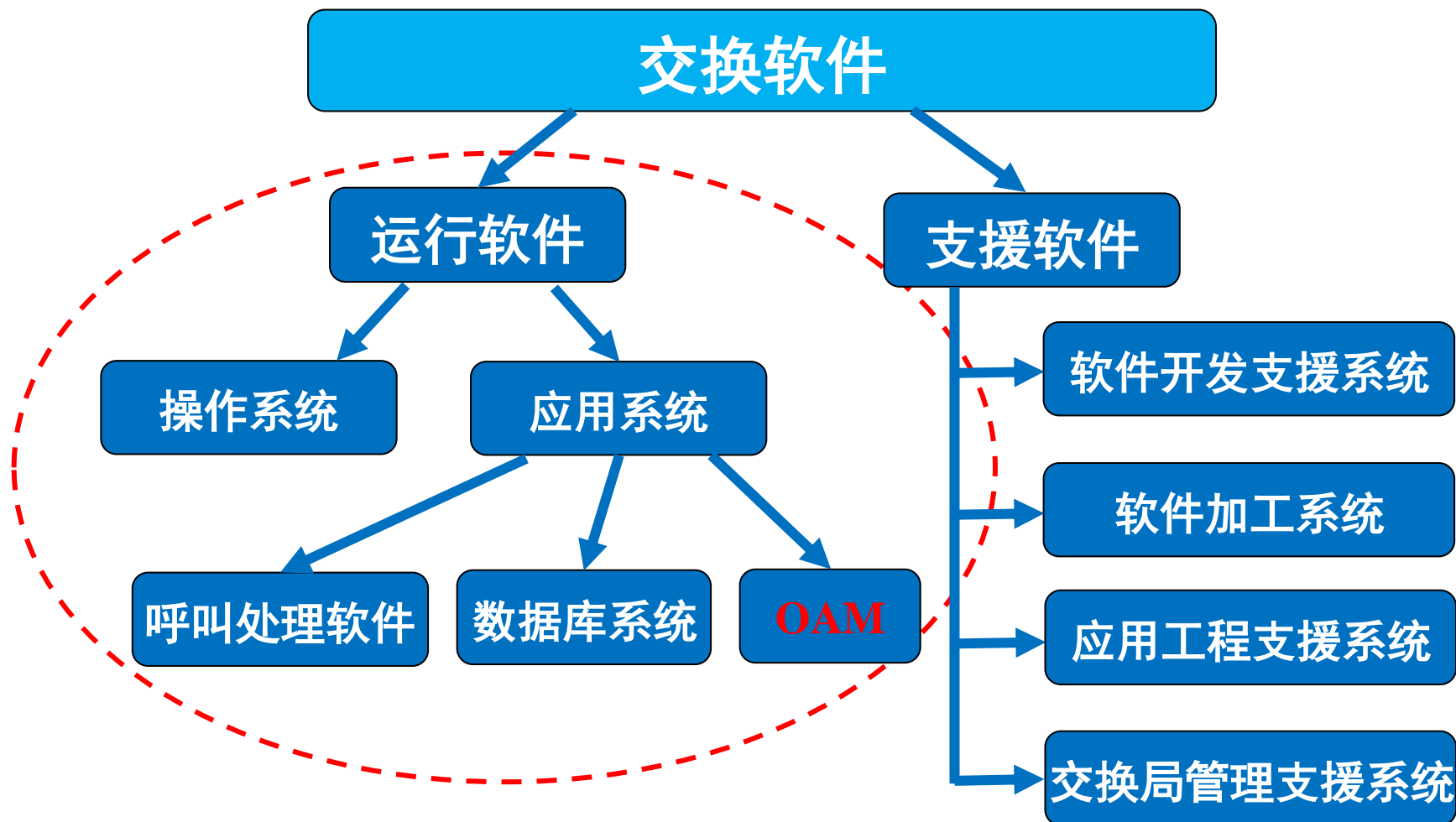
- 用户类别：住宅用户、公用电话用户、PABX用户、传真用户等
- 话机类别：PULSE话机、DTMF话机
- 用户状态：空闲、忙、测试、阻塞等
- 限制情况：呼出限制、呼入限制等
- 呼叫权限：本局呼叫、本地呼叫、国内长途、国际长途等
- 计费类别：定期、立即、免费等
- 优先级：普通用户、优先用户
- 新业务权限：呼叫转移、会议电话、三方通话、呼叫等待等
- 新业务登记的数据：转移号码、热线号码等
- 用户号码：用户电话薄号码、用户设备号等
- 呼叫动态数据：呼叫状态、时隙、收号器号、计数值等



局数据

- **硬件配置：**用户端口数、出/入中继线数、DTMF收号器数等
- **各种号码：**本地网编号及号长、局号、应收号码等
- **路由数据：**局向、路由数
- **计费数据：**呼叫详细话单（CDR）等
- **统计数据：**话务量、呼损、呼叫情况等
- **交换机类别：**C1-C5，C5又分为市话端局、长市合一等
- **复原方式：**主叫控制、被叫控制、互不控制

交换软件系统组成





OAM（操作维护管理）软件

用以保证系统高效、灵活、可靠地运行，具体完成的功能如下：

- 用户数据和局数据管理
- 测试
- 告警
- 故障诊断与处理
- 动态监视
- 话务统计
- 计费
- 过负荷控制等



呼叫处理原理

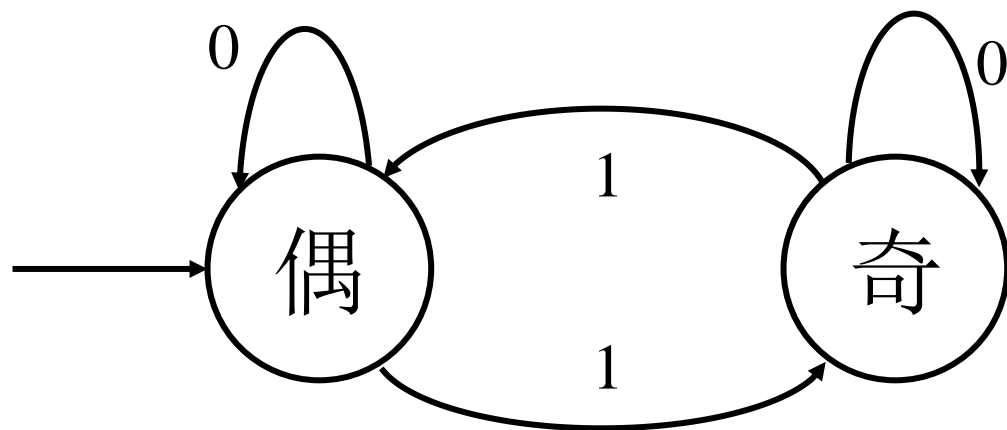
- 程控交换软件概述
- 扩展有限状态机&SDL
- 呼叫处理原理
- 交换软件设计技术

扩展有限状态机 & SDL

有限状态自动机: 可定义为一个四元系统 $\langle S, i, E, T \rangle$

其中:

- S : 系统状态集, 状态数有限
- i : 系统初始状态, $i \in S$
- E : 输入字母集
- T : 转移函数集, 是从 $S \times E$ 到 S 的映射



单比特奇偶检验器的
有限状态自动机



扩展有限状态机 & SDL

扩展有限状态自动机： 定义为一个四元系统 $\langle S, i, E, T \rangle$
其中：

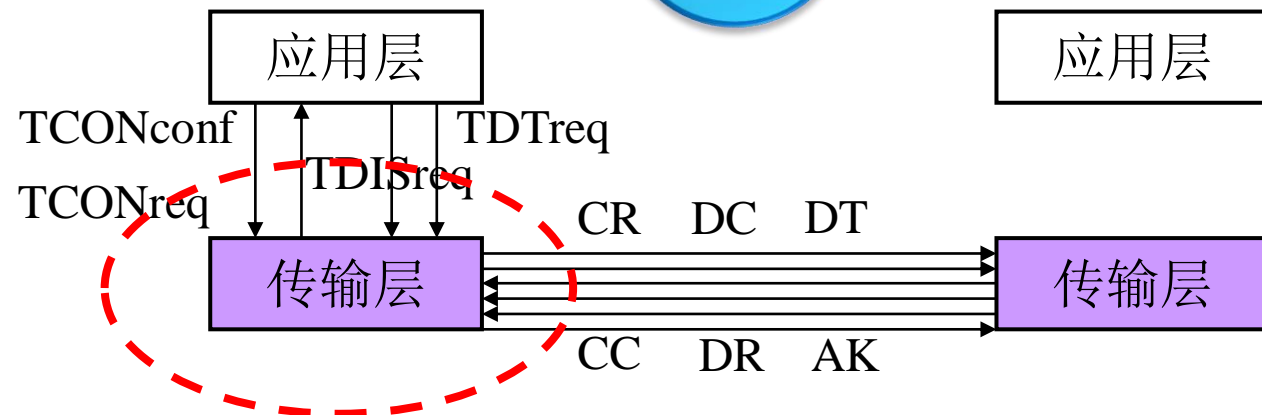
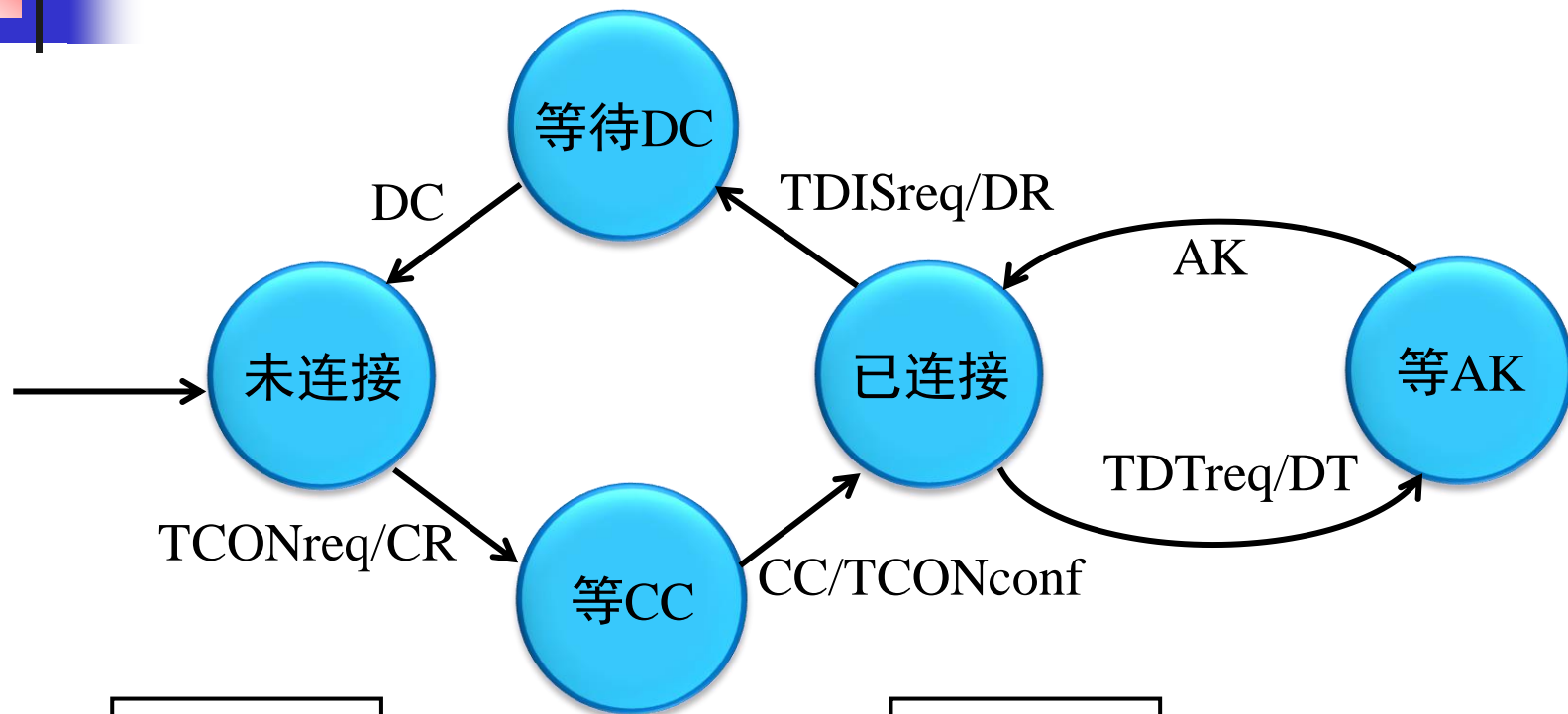
- S ：系统状态集，状态数有限
- i ：系统初始状态， $i \in S$
- E ：输入事件集
- T ：转移函数，是从 $S \times E$ 到 S 的映射

$$S_i \xrightarrow{t} S_j \in T, \quad S_i, S_j \in S$$

- $t = e / p: a$; 或
- $t = e / a$; 或
- $t = e$

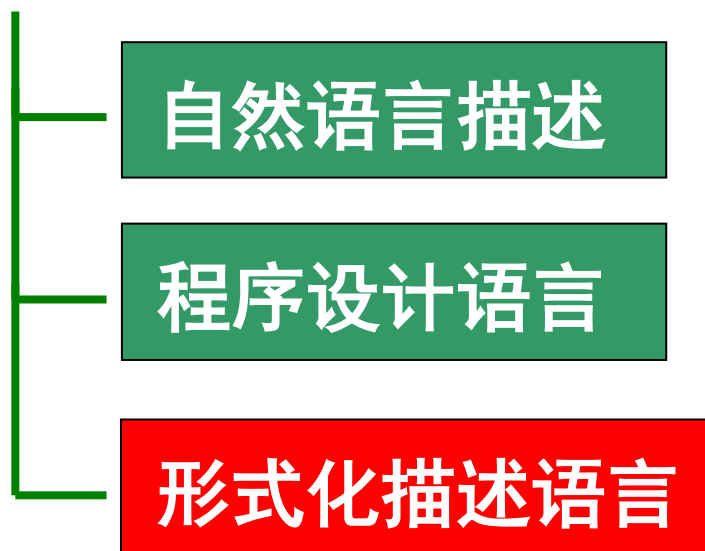
其中： $e \in E$ ， p 为谓词，表示条件； a 表示动作

传输层 (TCP) 协议机



SDL（Specification and Description Language）是**ITU-T**提出的一种形式化描述语言，由**Z.100**建议定义。

系统行为：复杂性





形式化方法（描述语言）特点

- 形式化的语法
- 形式化的语义
- 清楚的概念模型
- 界面统一表示
- 强大的表达和描述功能
- 有助于系统的实现和完善

分析、设计

模拟、验证、测试

交流、维护、再开发

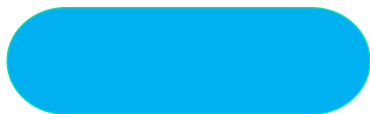


SDL的特点

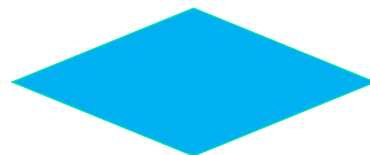
- 可用于需求分析到具体实现的整个开发过程
- 适应于实时系统
- 基于扩展的有限状态机
- 图形化表示方式，可视性强
- 具有面向对象的特征

SDL图形化表示常用符号

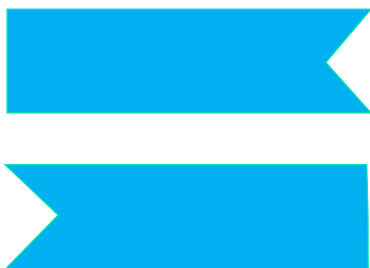
状态



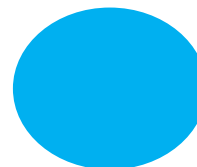
判断



输入



连接



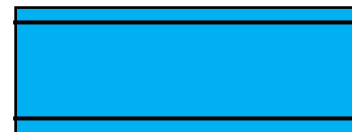
输出



任务



进程创建





呼叫处理原理

- 程控交换软件概述
- 扩展有限状态机&SDL
- 呼叫处理原理
- 交换软件设计技术

分析本局呼叫的处理过程

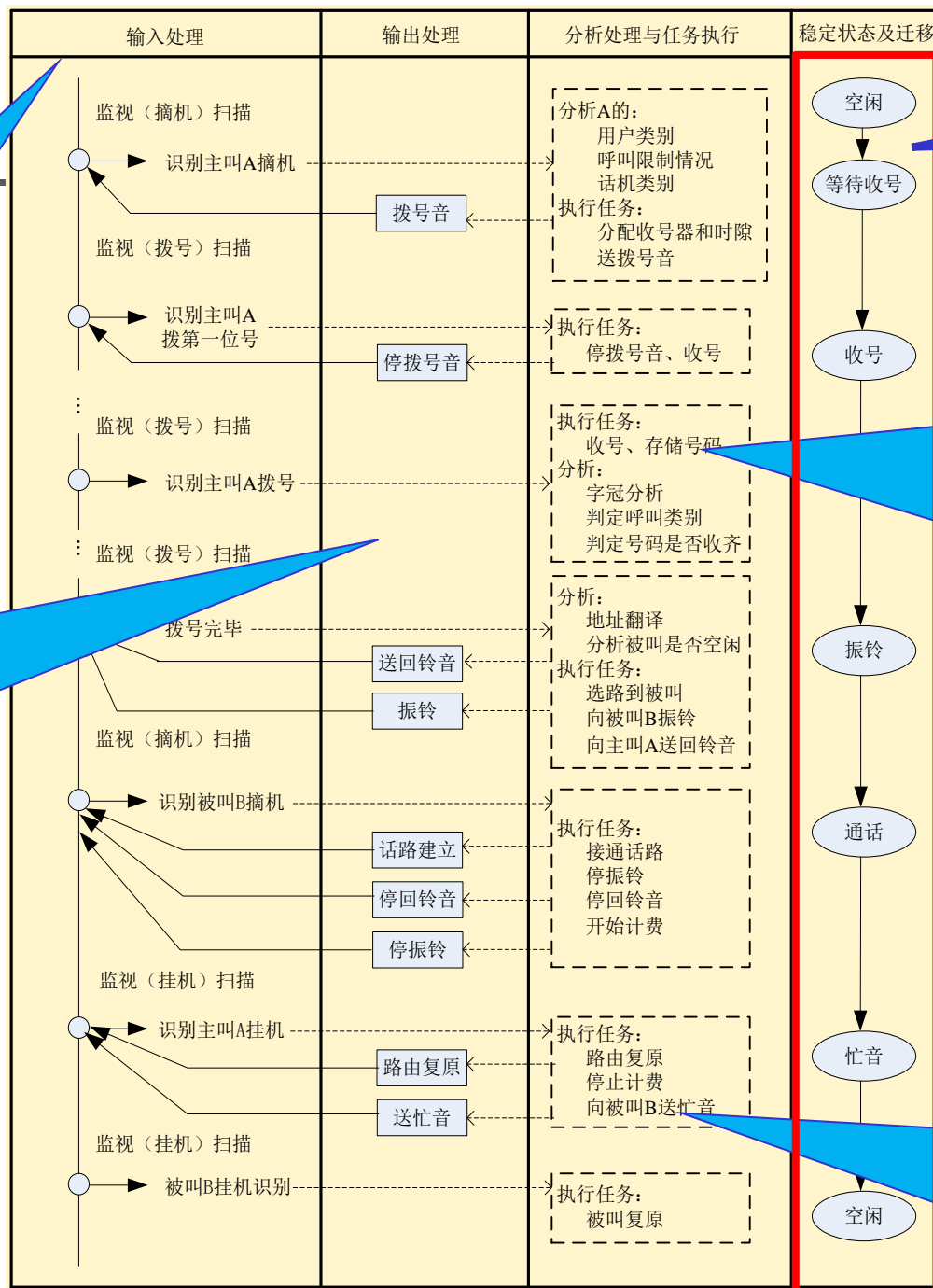
呼叫进展状况	交换机处理动作、状态变化
主叫A 摘机呼叫	<ol style="list-style-type: none">1、交换机检测到用户A的摘机信号2、交换机检查用户A的类别，识别是普通电话、公用电话等3、交换机检查用户呼叫限制情况4、交换机检查话机类别，以确定是PULSE还是DTMF收号方式
向A送拨号音 准备收号	<ol style="list-style-type: none">1、交换机选择一个空闲收号器和空闲的时隙（路由）2、交换机向主叫A送拨号音3、监视主叫A所在用户线的输入信号（拨号），准备收号
收号与 号码分析	<ol style="list-style-type: none">1、交换机收到第一位号码后停拨号音2、交换机按位存储收到的号码3、交换机对号首进行分析，判定呼叫类别（本局、出局、长途、特服等），并确定应收号长4、交换机对“已收号长”进行计数，并与“应收号长”比较5、号码收齐后，本局呼叫进行号码翻译，确定被叫6、交换机检查被叫用户是否空闲，若空闲则选定该被叫

分析本局呼叫的处理过程

呼叫进展状况	交换机处理动作、状态变化
建立连接 向 B 振铃 向 A 送回铃音	<ol style="list-style-type: none">1、交换机将路由接至被叫B2、向被叫B振铃3、向主叫A送回铃音4、主、被叫通话路由建立完毕5、监视主、被叫用户状态
被叫应答 进入通话	<ol style="list-style-type: none">1、被叫摘机应答，交换机检测到后，停振铃和停回铃音2、A、B通话3、开始计费4、监视主、被叫用户状态
一方用户挂机 向另一方送忙音	<ol style="list-style-type: none">1、如果主叫A先挂机，交换机检测到后，复原路由，停止计费，向被叫B送忙音2、如果被叫B先挂机，交换机检测到后，复原路由，停止计费，向主叫A送忙音
通话结束	被催挂的用户挂机，释放占用的所有资源，通话结束。

指识别和接收输入信号的过程，由输入处理程序

任务执行中，要输出一些信令、消息或动作命令，该过程为输出处理



状态迁移

是对输入处理的结果（输入信号）、当前状态以及各种数据进行分析，以决定下一步执行什么任务的过程

任务执行是指在迁移到下一个稳定状态之前，根据分析处理结果，完成相关任务

- ❑ 整个呼叫处理过程可分为若干个阶段，每个阶段可以用一个稳定的状态来表示；
- ❑ 整个呼叫处理的过程就是在一个稳定状态下，处理机监视、识别输入信号，进行分析处理，执行任务和输出命令，然后跃迁到下一个稳定状态的循环过程；
- ❑ 两个稳定的状态之间要执行各种处理；
- ❑ 在一个稳定状态下，若没有输入信号，状态不会迁移；
- ❑ 相同的输入信号在不同的状态下会有不同的处理，并迁移到不同的状态；
- ❑ 在同一状态下，对不同输入信号的处理是不同的；
- ❑ 在同一状态下，输入同样信号，也可能因不同情况得出不同结果。



呼叫处理的特点

□ 总结

- ✓ 基于扩展的有限状态机
- ✓ 输入处理、分析处理、任务执行和输出处理

通信软件：输入事件、状态！！



呼叫处理原理——输入处理

输入处理程序需完成的功能主要有：

- 用户线扫描监视
- 中继线扫描监视（接收公共信道信令）
- 接收拨号脉冲、DTMF信号
- 接收操作台（OAM）的各种命令

对用户线、中继线、信令设备进行监视和信号识别，生成相应事件放入队列，供其它程序取用，大多属于**周期级程序**。



用户线扫描分析

用户线上各种不同的状态具有的共同的特点：

□ 形成直流回路（**续**）、断开直流回路（**断**）

摘机—“续”，挂机—“断”

送脉冲—“断”，脉冲间隔—“续”

□ 周期性监视

用户摘挂机扫描周期——100~200ms

拨号脉冲识别周期——8~10ms



摘挂机识别原理

设：用户在挂机状态时扫描输出为“**1**”（断）

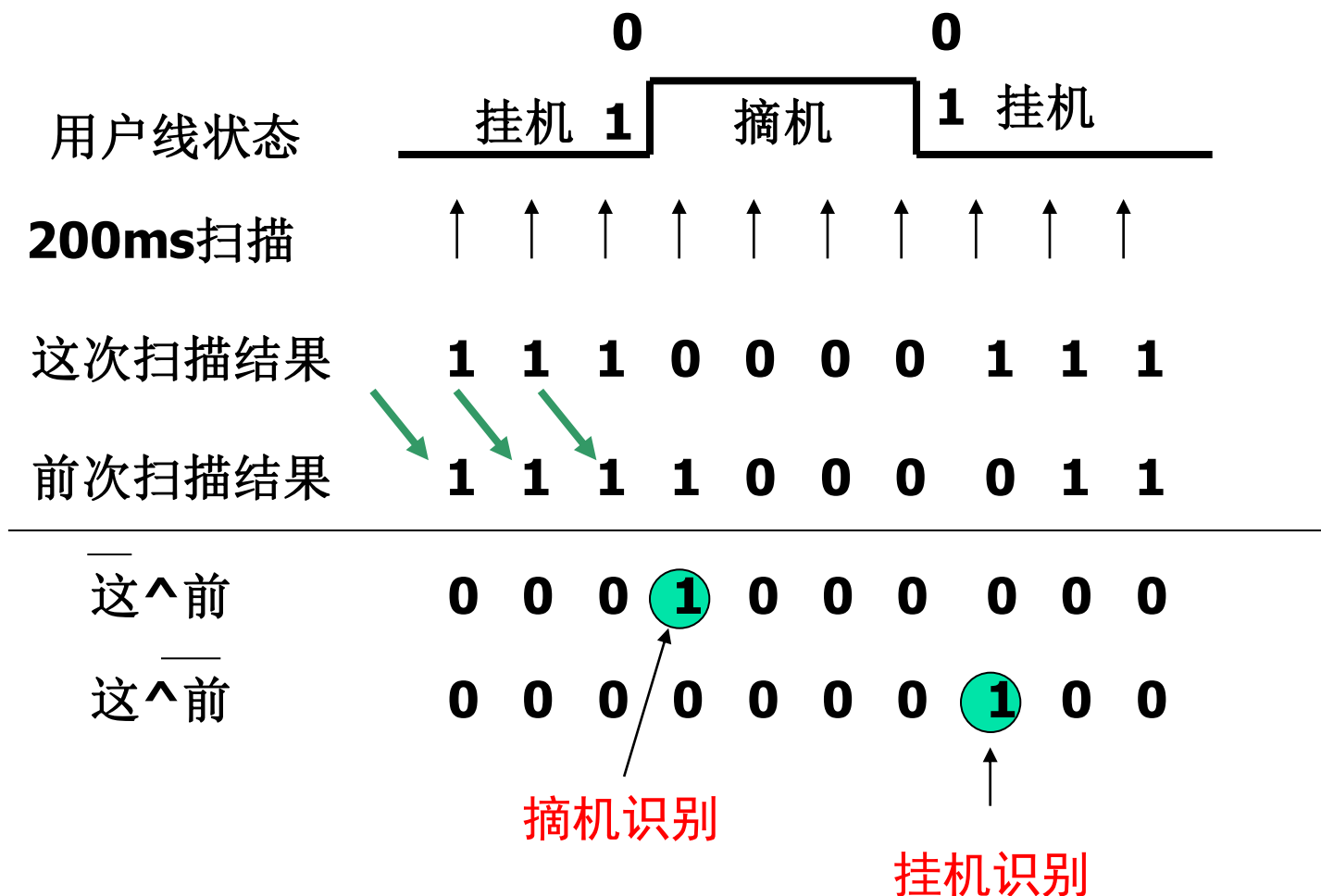
用户在摘机状态时扫描输出为“**0**”（续）

摘挂机扫描程序的执行周期为**200ms**

摘机识别：就是找到从“**1**”到“**0**”的**变化点**

挂机识别：就是找到从“**0**”到“**1**”的**变化点**

摘挂机识别原理





呼叫处理原理——分析处理

输入处理得到的各种事件，交分析处理程序分析，以决定下一步的工作。它没有固定周期，属于**基本级程序**。

- 去话分析
- 号码分析
- 来话分析
- 状态分析

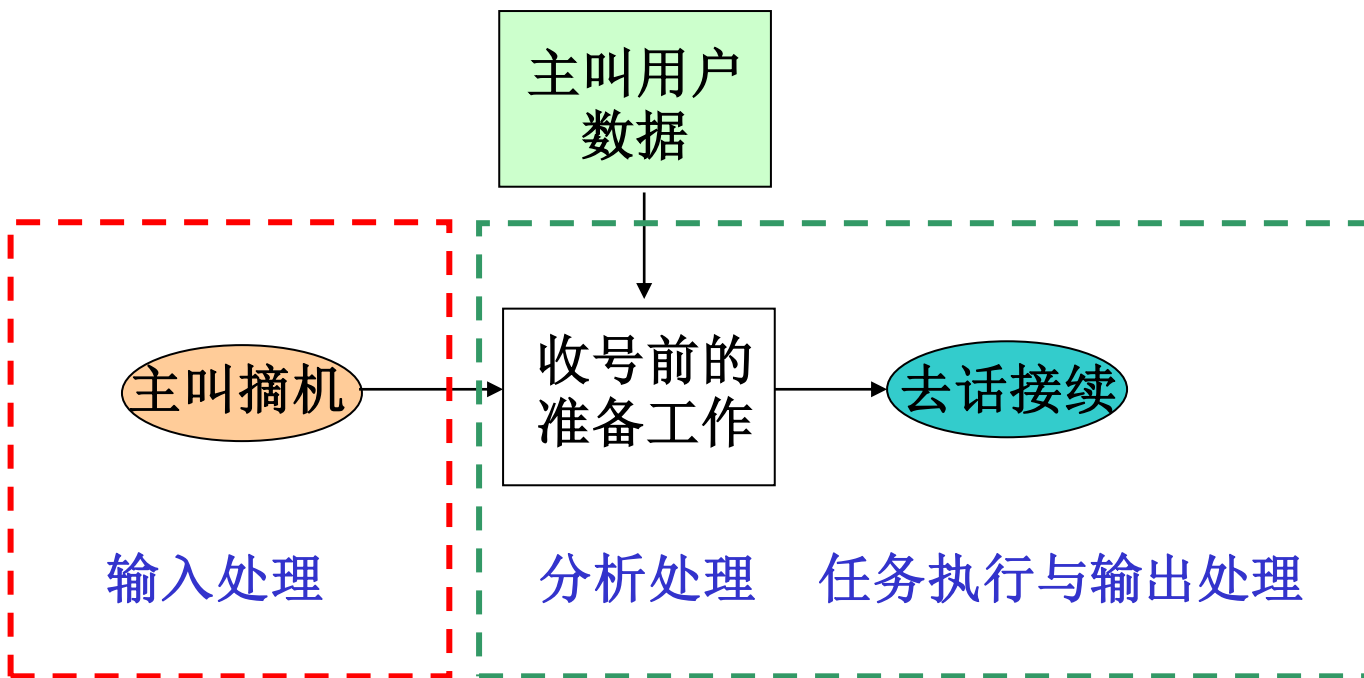


基本级程序

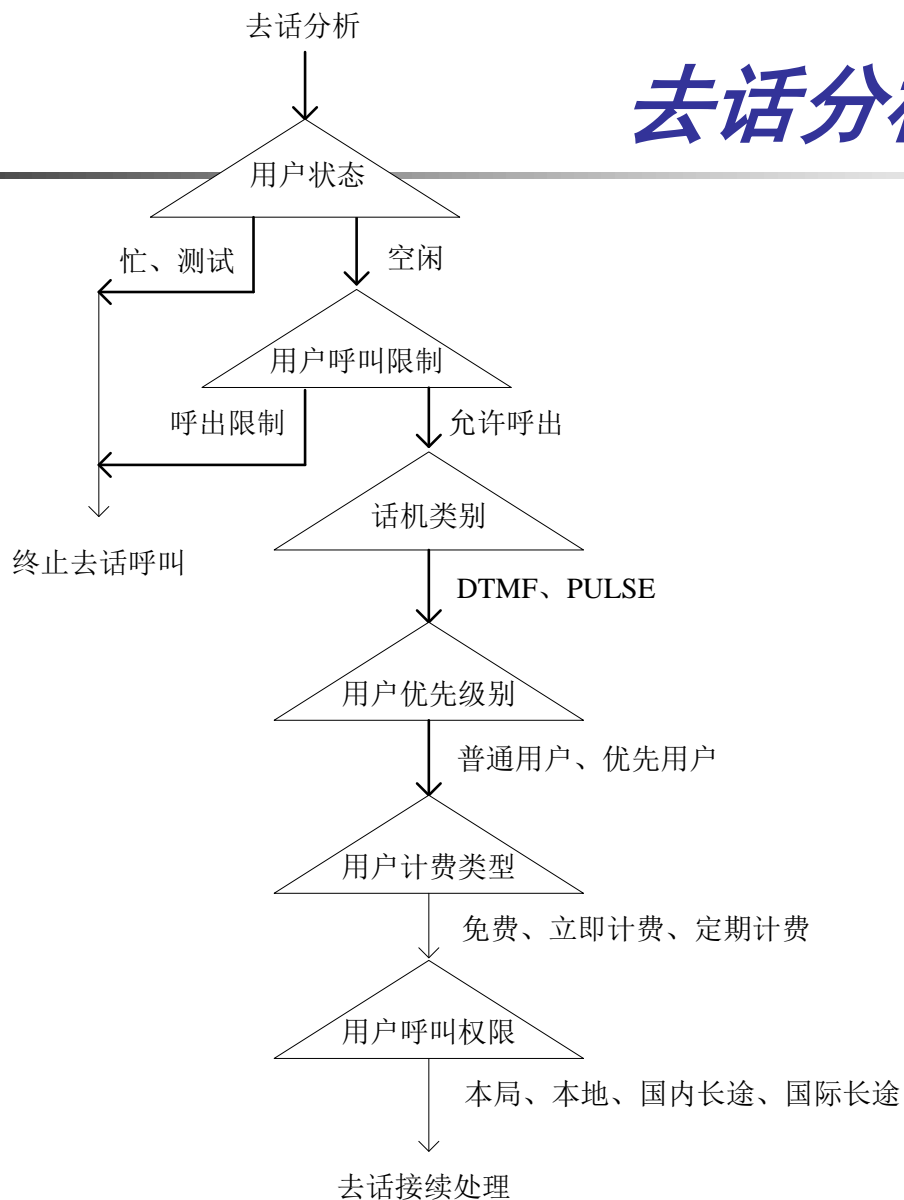
去话分析

去话分析：主叫用户摘机发起呼叫时进行的分析

分析基于：主叫用户数据



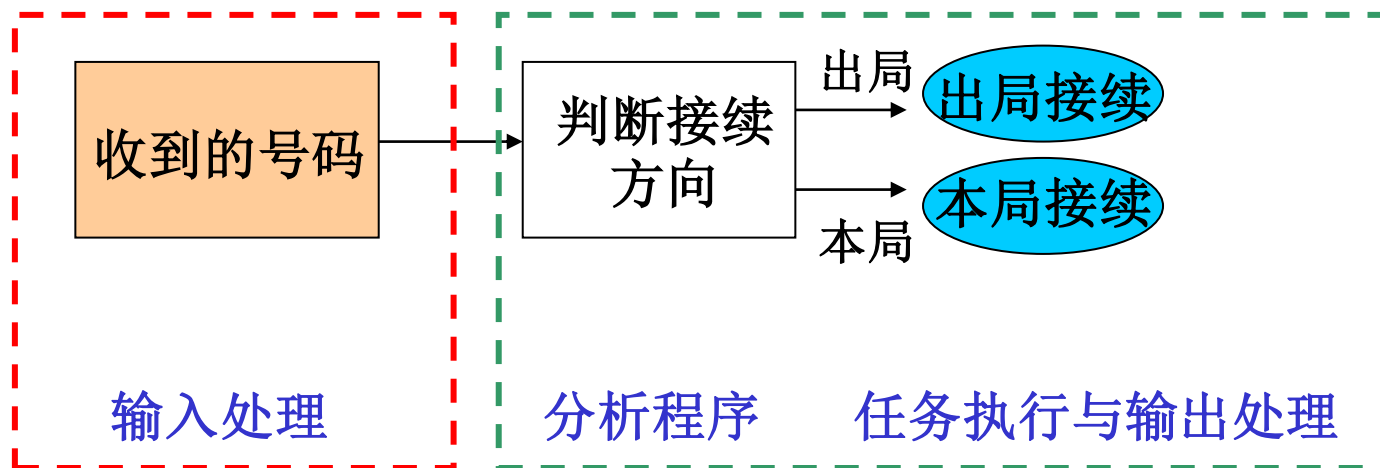
去话分析的一般流程



号码分析

号码分析：在收到用户的拨号号码时进行的分析

分析基于：用户拨号号码（用户线、中继线）





号码分析

号码分析：号首分析、号码翻译

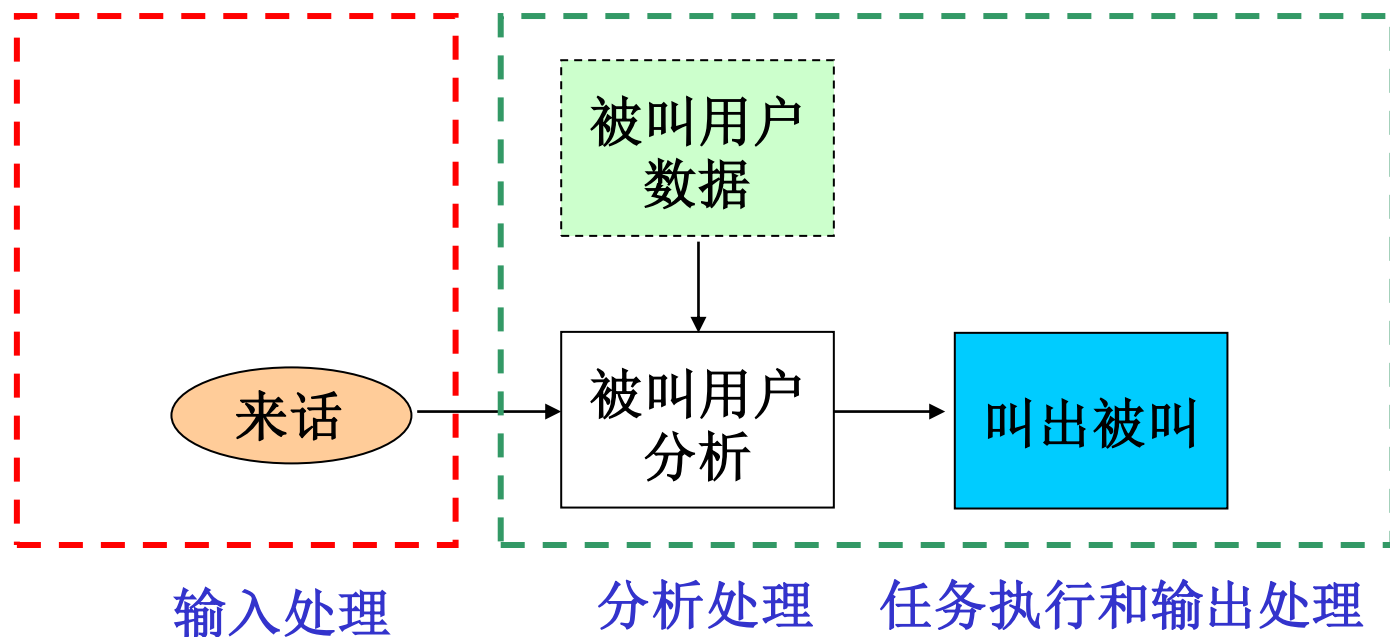
号首分析：收到的前几位号码的分析，一般为1~3位，以判定呼叫的接续类型，获取应收号长和路由等信息。

号码翻译：对被叫号码进行分析处理，以获得被叫用户（用户设备号）。

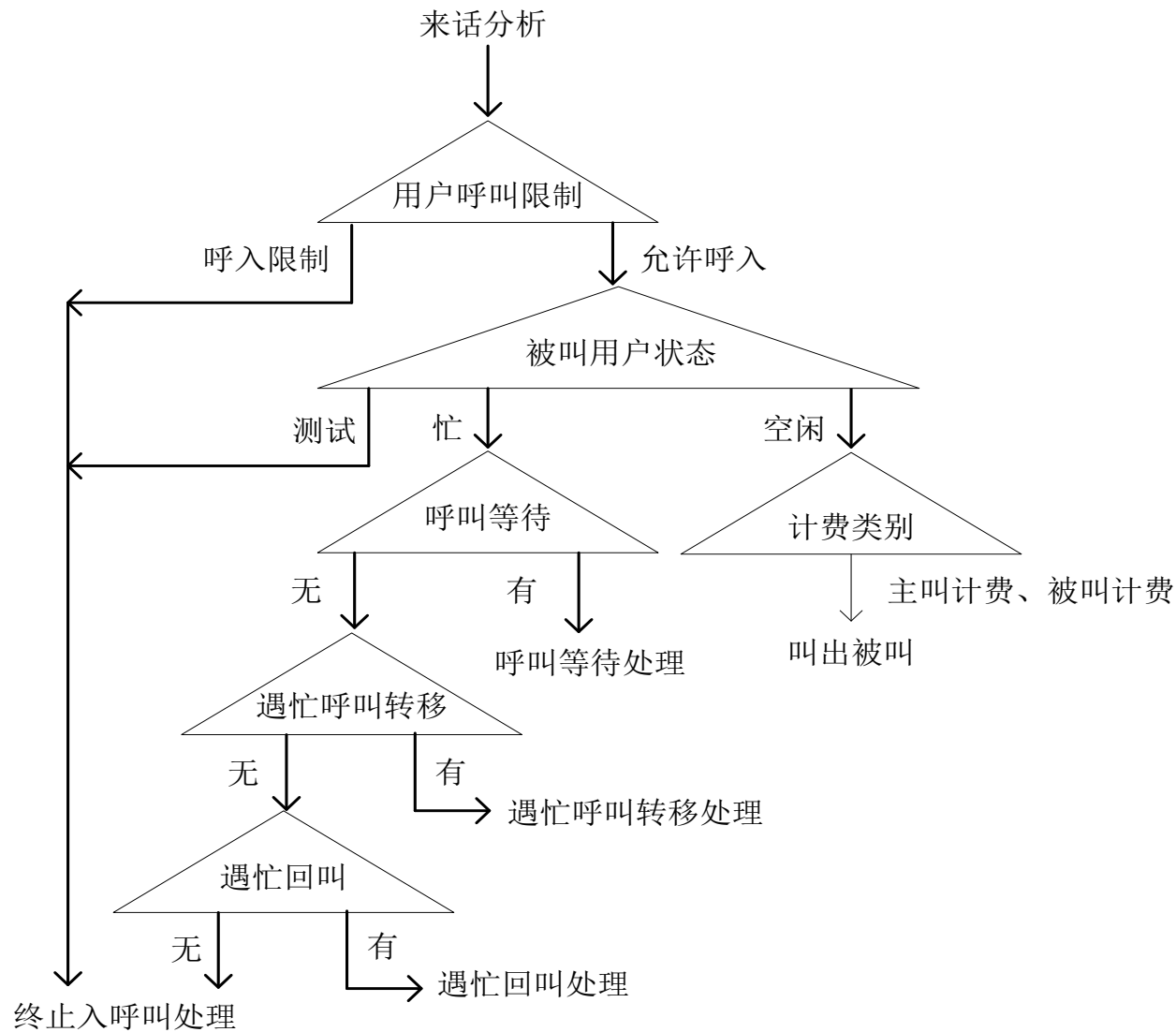
来话分析

来话分析：在收到它局呼入信号且被叫在本局时进行的分析

分析基于：被叫用户数据



来话分析的一般流程

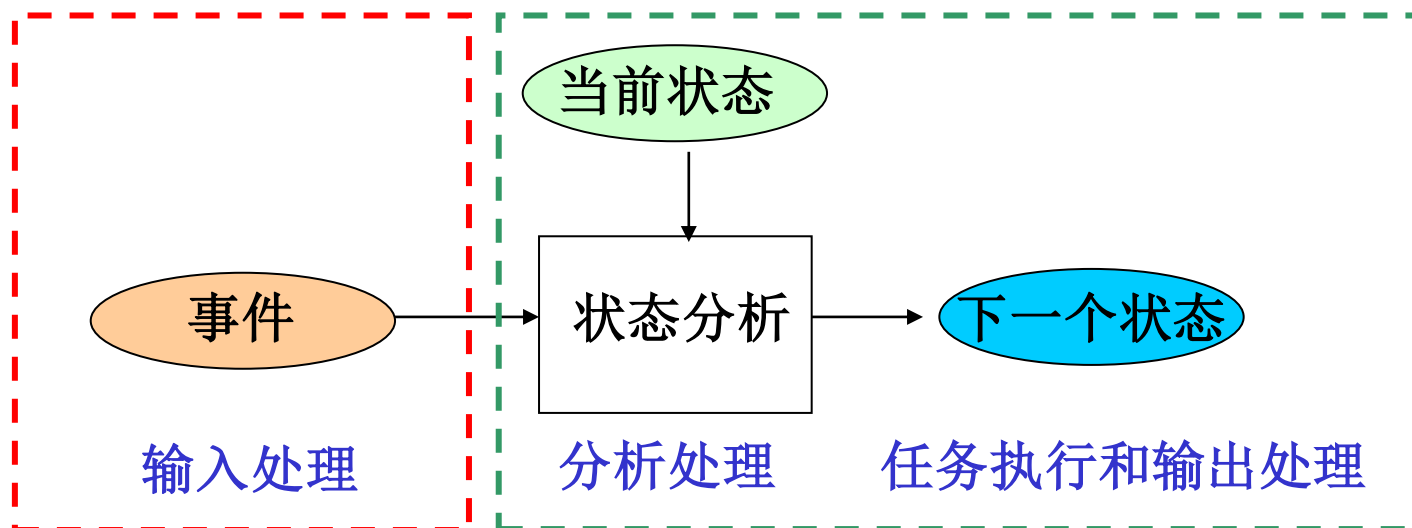


状态分析

状态分析的信息源是状态和输入事件。

呼叫状态：空闲、等待收号、收号、振铃、通话、忙音
空号音、催挂音、挂起等

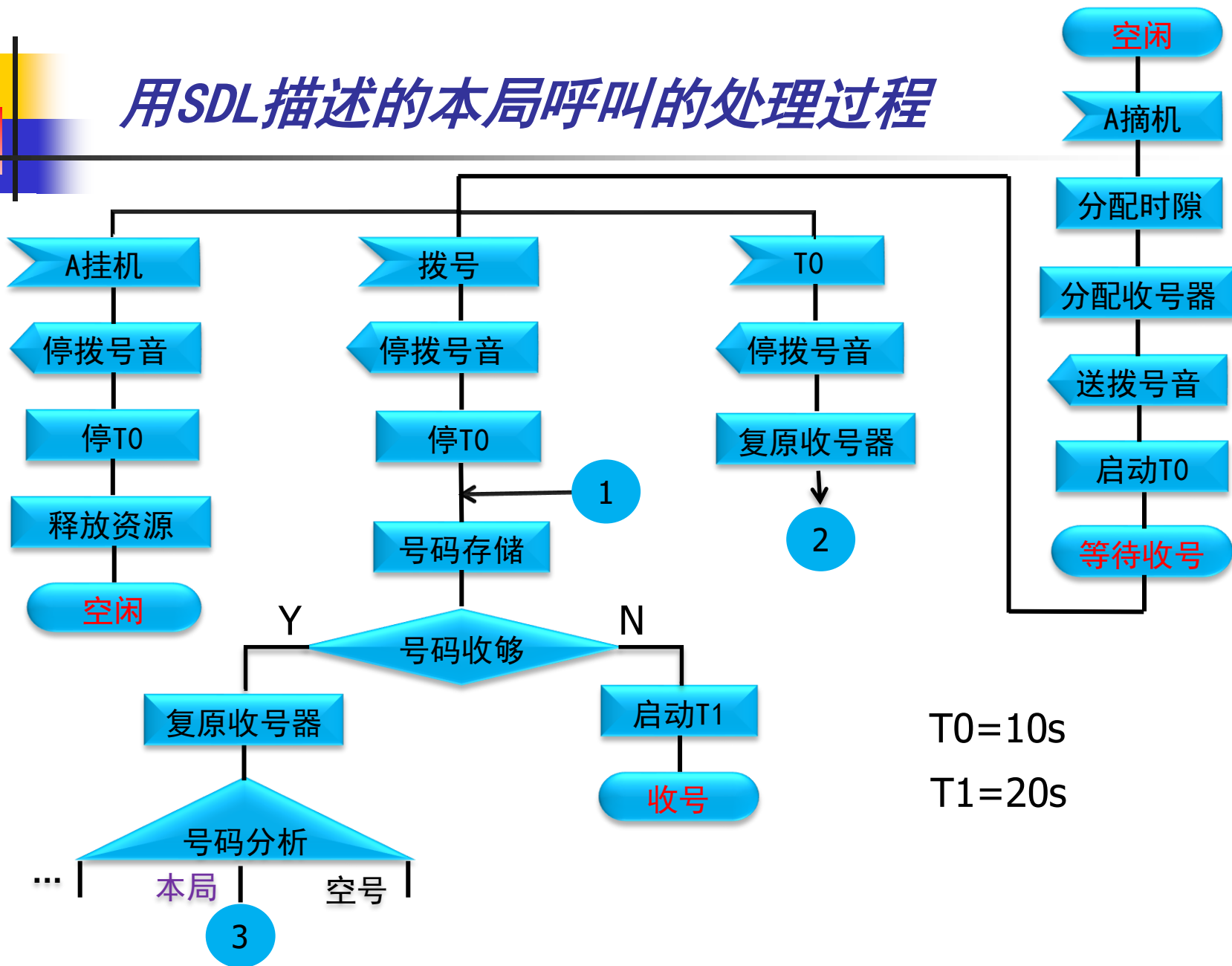
事件：摘机、挂机、超时、号码、空错号（分析结果产生）等

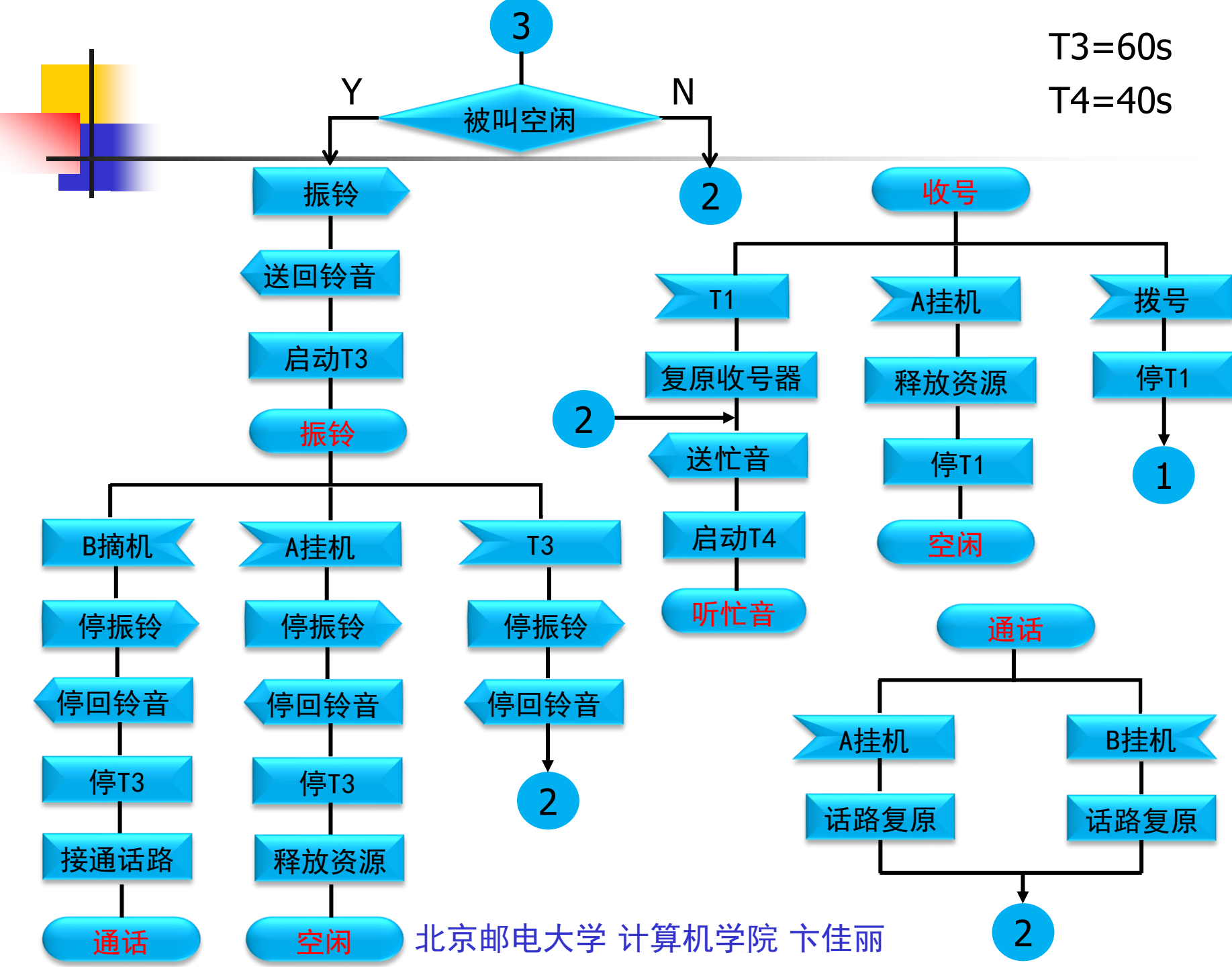


呼叫处理原理——任务执行和输出处理

- 启动和停止各种**计时器**
 - 分配和释放各种**资源**（DTMF收号器、时隙）
 - 形成信令、处理机间通信消息和驱动硬件的控制命令
 - 开始和停止计费、计算操作、存储各种号码
 - 对用户数据、局数据的读写操作
-
- 驱动交换网络**建立或拆除话路**
 - 送/停各种信号音、振铃和停振铃
 - 连接DTMF收号器
 - 发送公共信道信令、发送处理机间通信信息等

用SDL描述的本局呼叫的处理过程







呼叫处理原理

- 程控交换软件概述
- 扩展有限状态机&SDL
- 呼叫处理原理
- 交换软件设计技术



(1) 群处理

群处理:

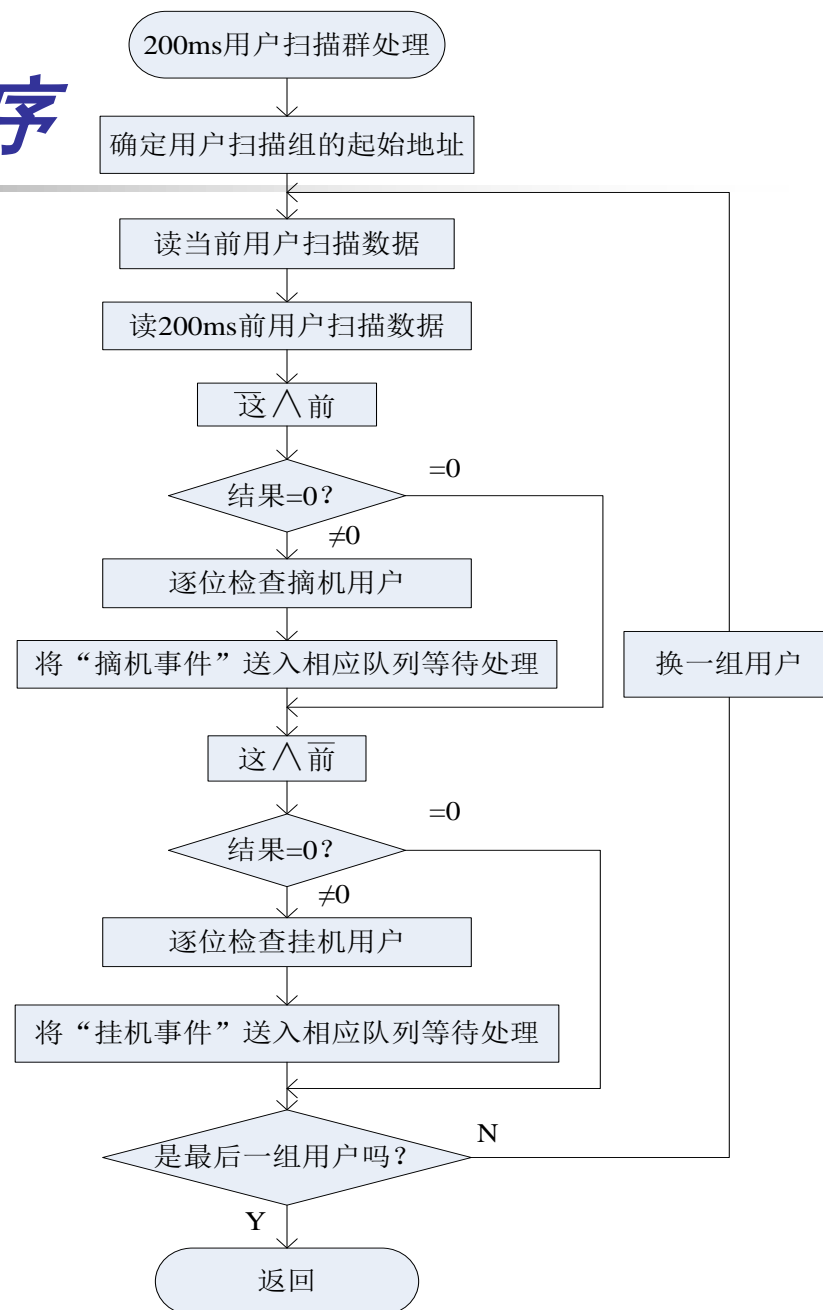
为提高效率，在软件设计中尽可能对一群对象同时进行逻辑运算和处理。

群处理——摘挂机识别（检测）

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
这次扫描结果	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1
前次扫描结果	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
— 这	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
— 前	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
— 这八前（摘机）	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
— 这八前（挂机）	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

用户群扫描处理程序

不仅对用户线扫描可采用群处理技术，对中继线状态扫描、按键号码标志信号扫描等都可采用群处理技术，群处理技术在程控交换软件设计中应用广泛。

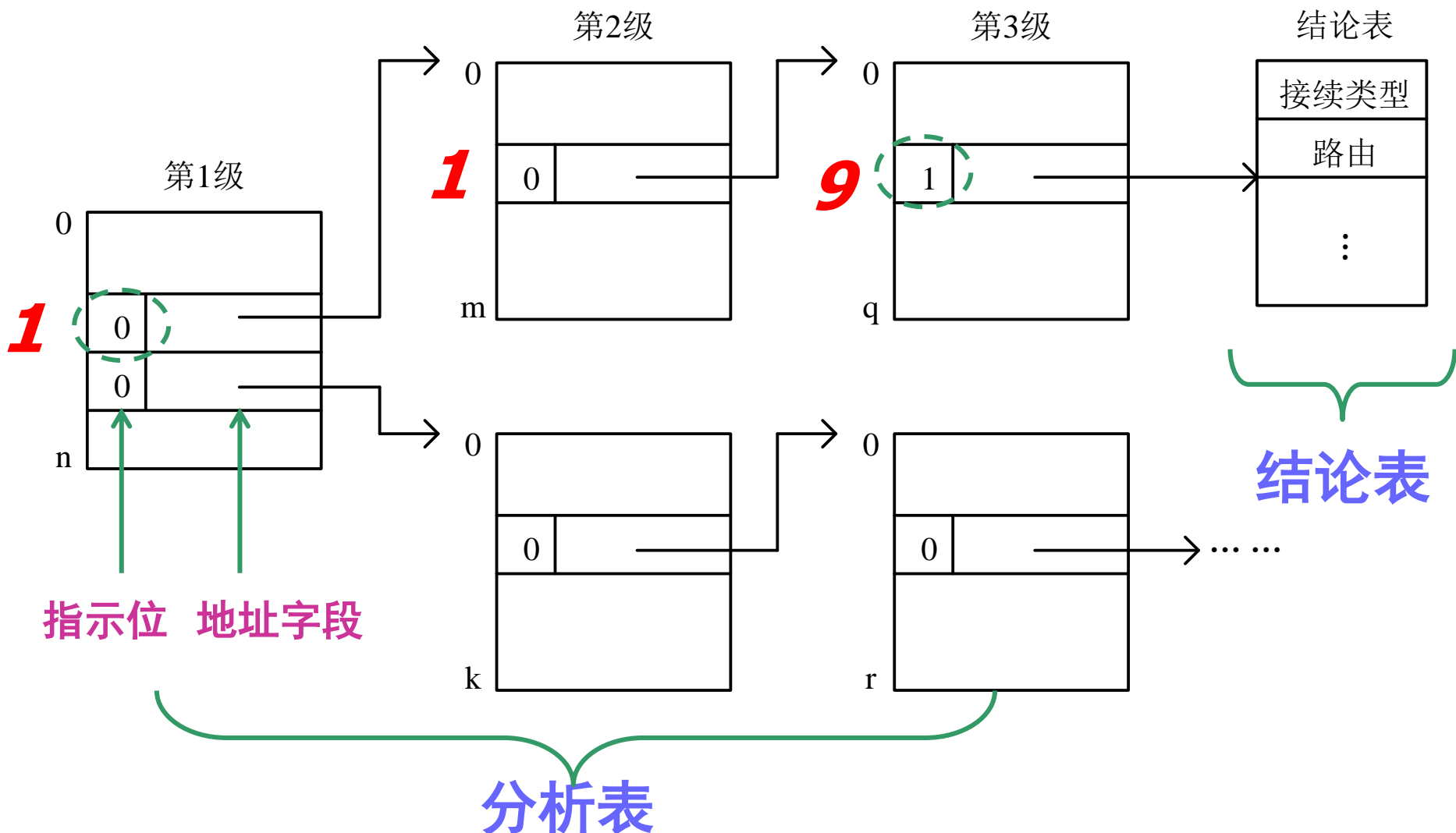




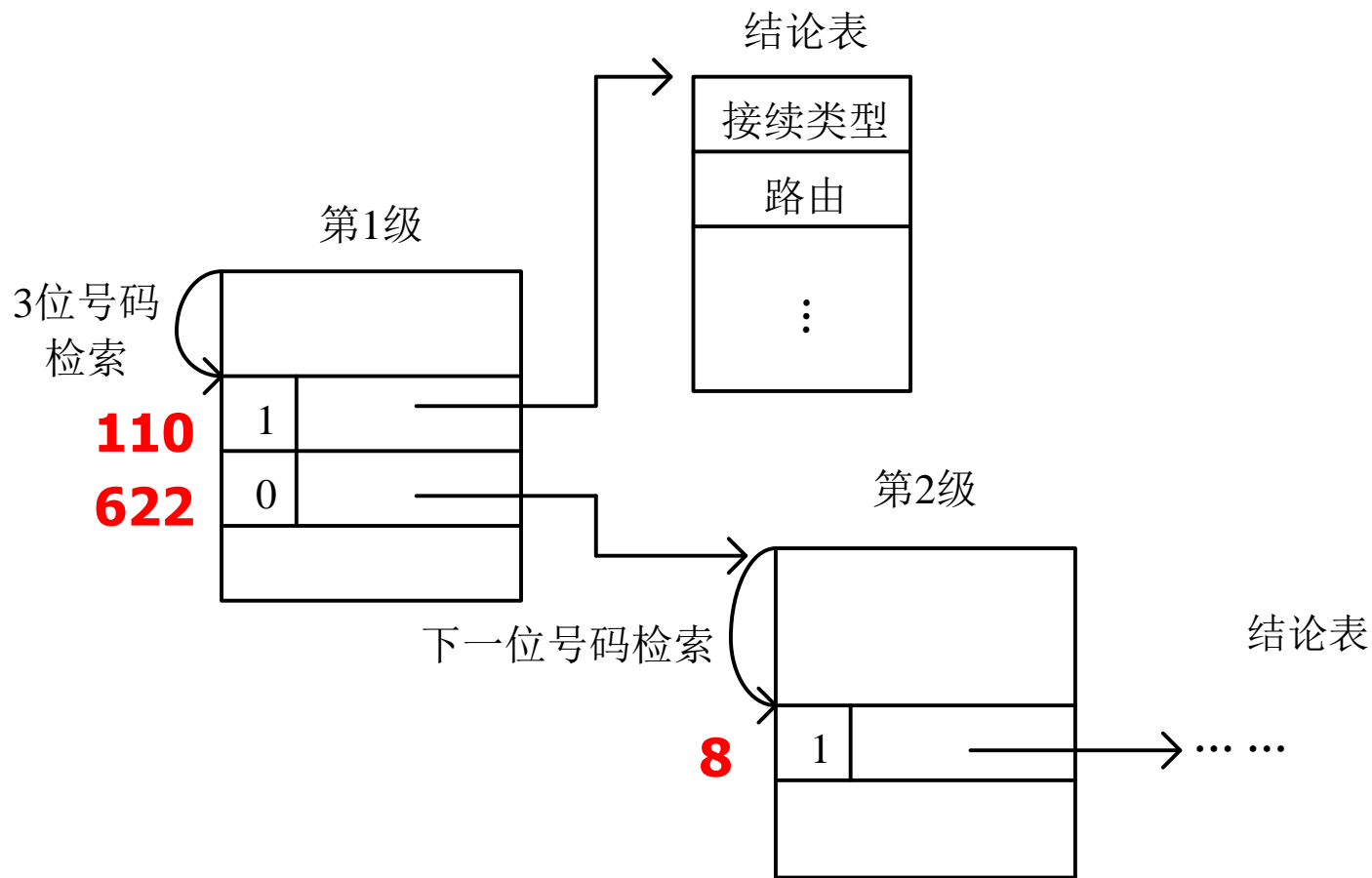
(2) 逐次展开法

逐次展开法基于逐次展开的分析表，
该表为多级检索表，呈树型结构。

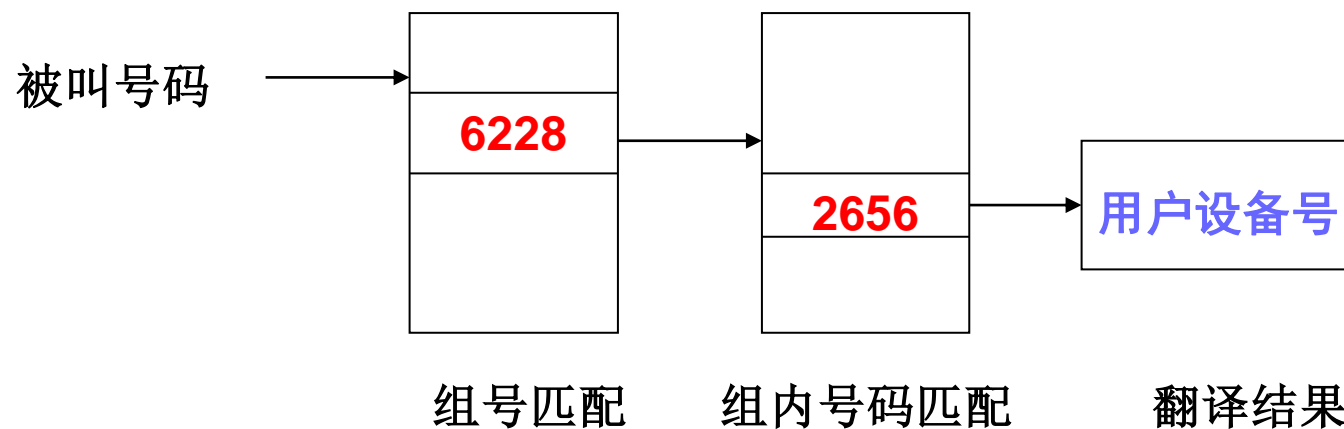
(2) 逐次展开法——号首分析



(2) 逐次展开法——号首分析



(2) 逐次展开法——号码翻译





(2) 逐次展开法——周期级程序调度

举例:

周期级程序调度: 时钟中断

应用需求: 周期10ms (k个)

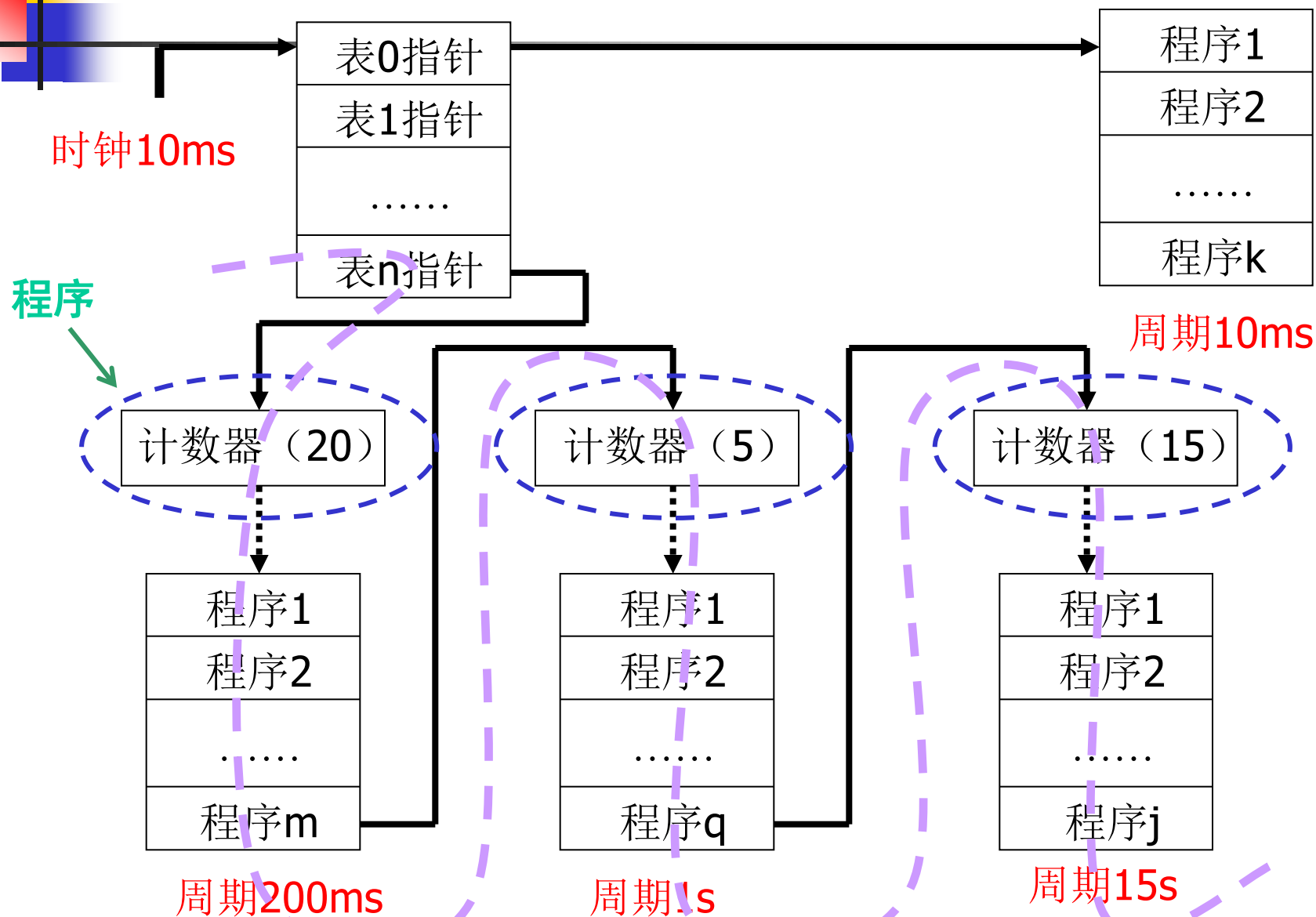
周期200ms (m个)

周期1s (q个)

周期15s (j个)

时钟: 10ms

(2) 逐次展开法——周期级程序调度





(3) 表格驱动

表格驱动是根据所给参数查表来启动程序执行的方法，可灵活地实现程序的调用执行。

表格驱动技术包括两部分内容：

- 驱动表格
- 调度管理程序



形成时间表

行地址

屏蔽表

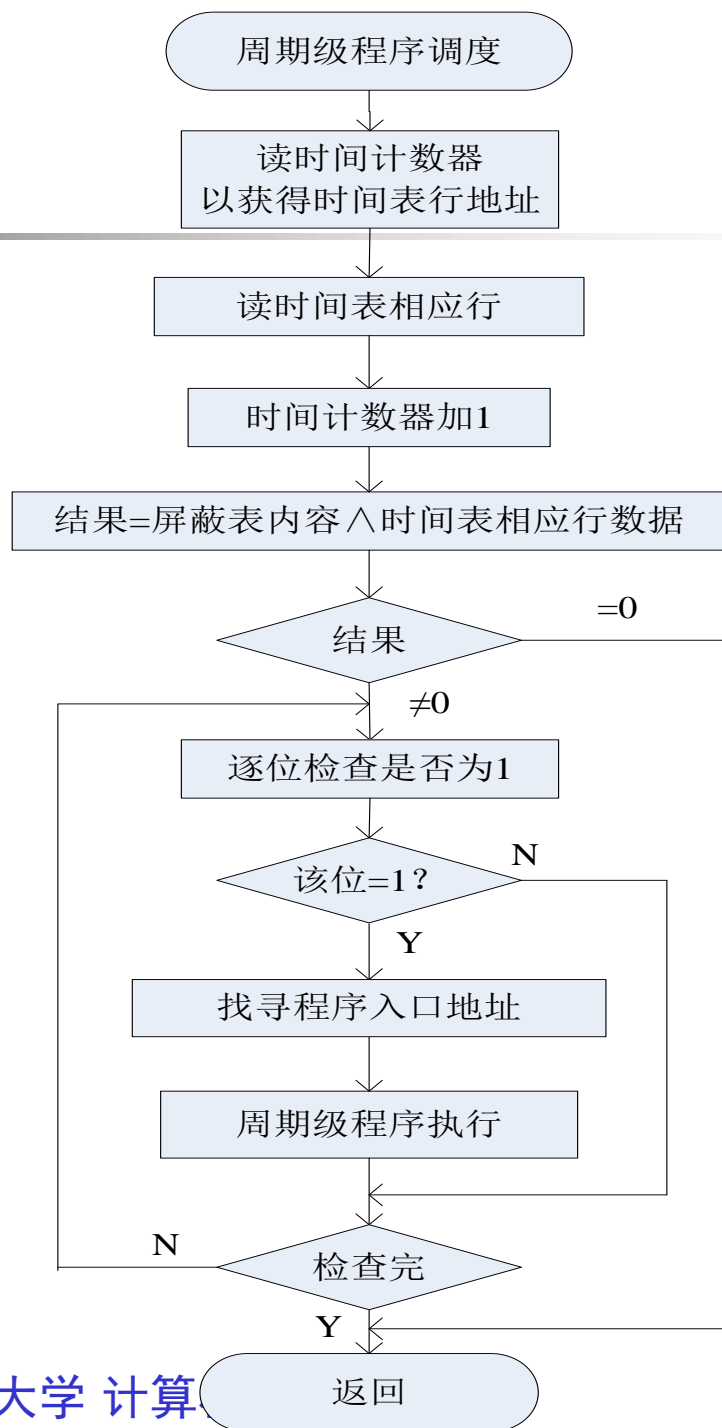
一种灵活控制程序调用的机制

时间表

程序地址表

驱动周期级程序调度执行的表格结构

基于表格驱动的调度管理程序流程图





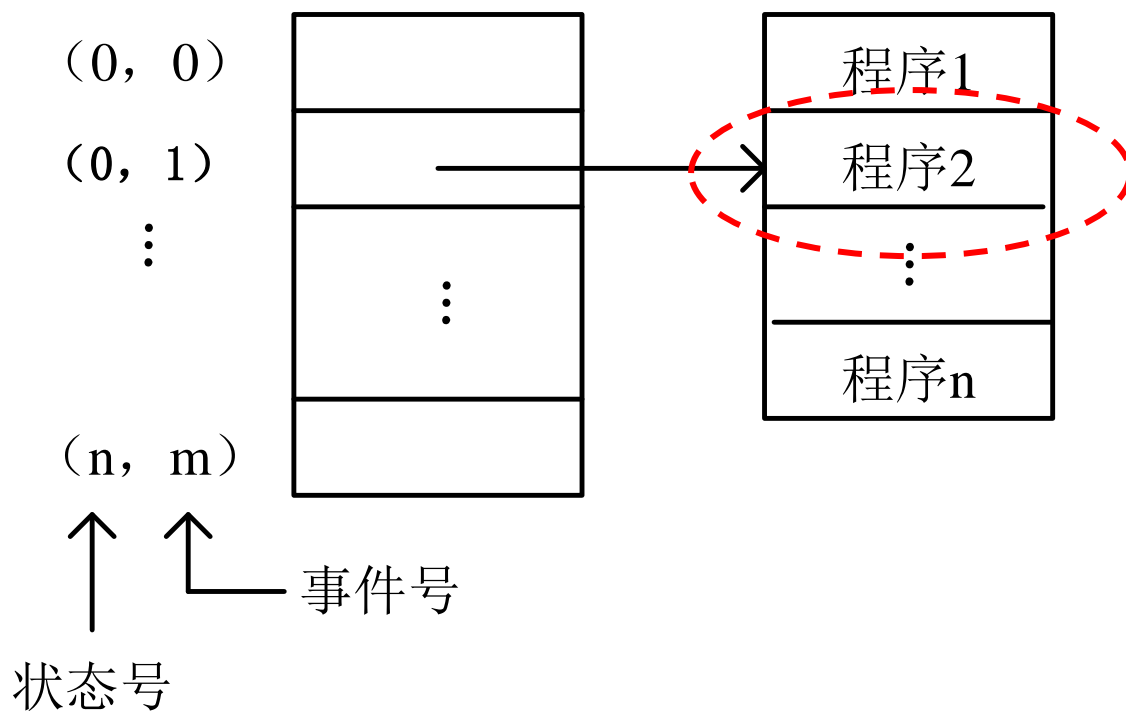
(4) 有限状态机的实现

呼叫处理过程可以用扩展的有限状态机来描述，因而呼叫处理程序的实现，就是实现呼叫处理的有限状态机。

设计实现有限状态机的方法有很多，常用的两种实现方法有：二维数组法、多级表法。

基于二维数组的有限状态机的实现

任务执行程序入口地址





状态:

- 0 (空闲)
- 1 (准备收号)
- 2 (收号)
- 3 (听忙音)
- 4 (振铃)
- 5 (通话)
-

事件:

- 0 (主叫摘机)
- 1 (拨号)
- 2 (主叫挂机)
- 3 (被叫摘机)
- 4 (被叫挂机)
- 5 (T0超时)
- 6 (T1超时)
- 7 (T3超时)
- 8 (T4超时)

.....

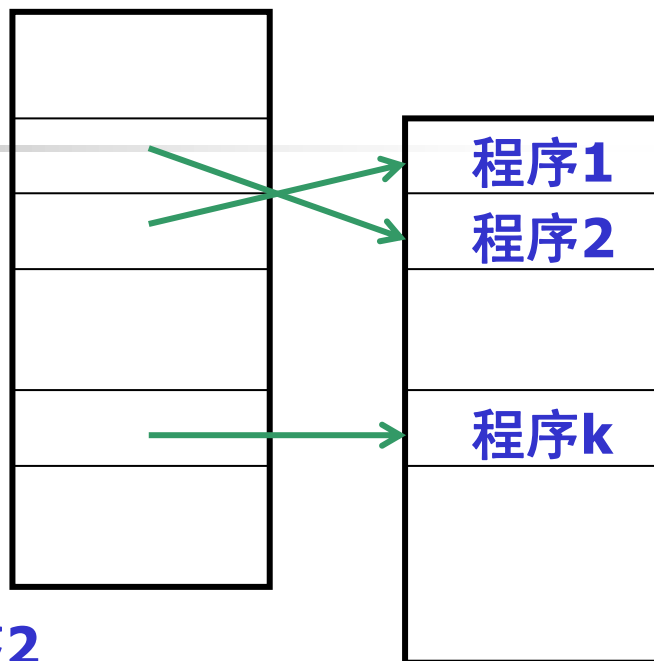
举例



(4,2)

(4,3)

(4,7)

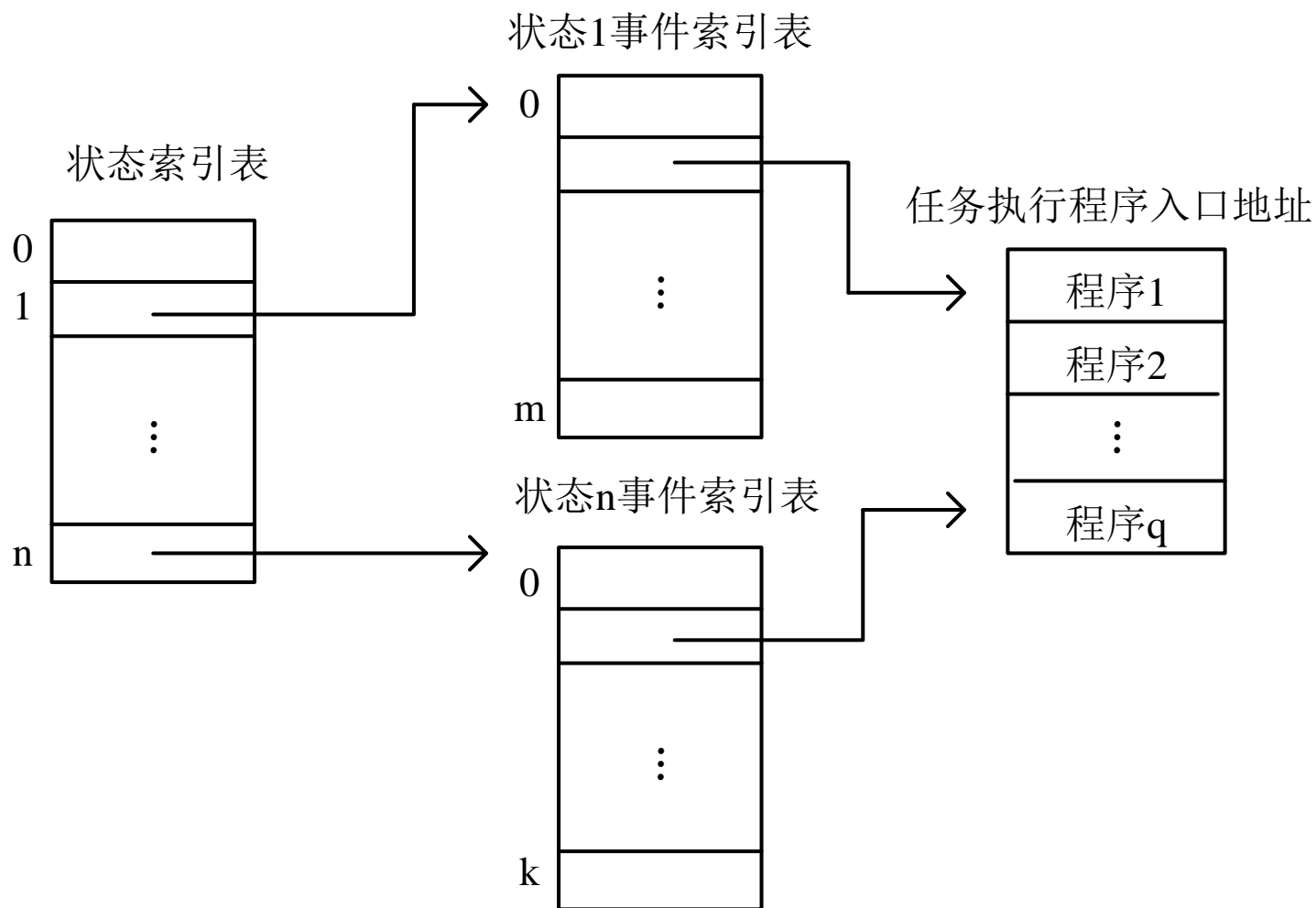


程序2

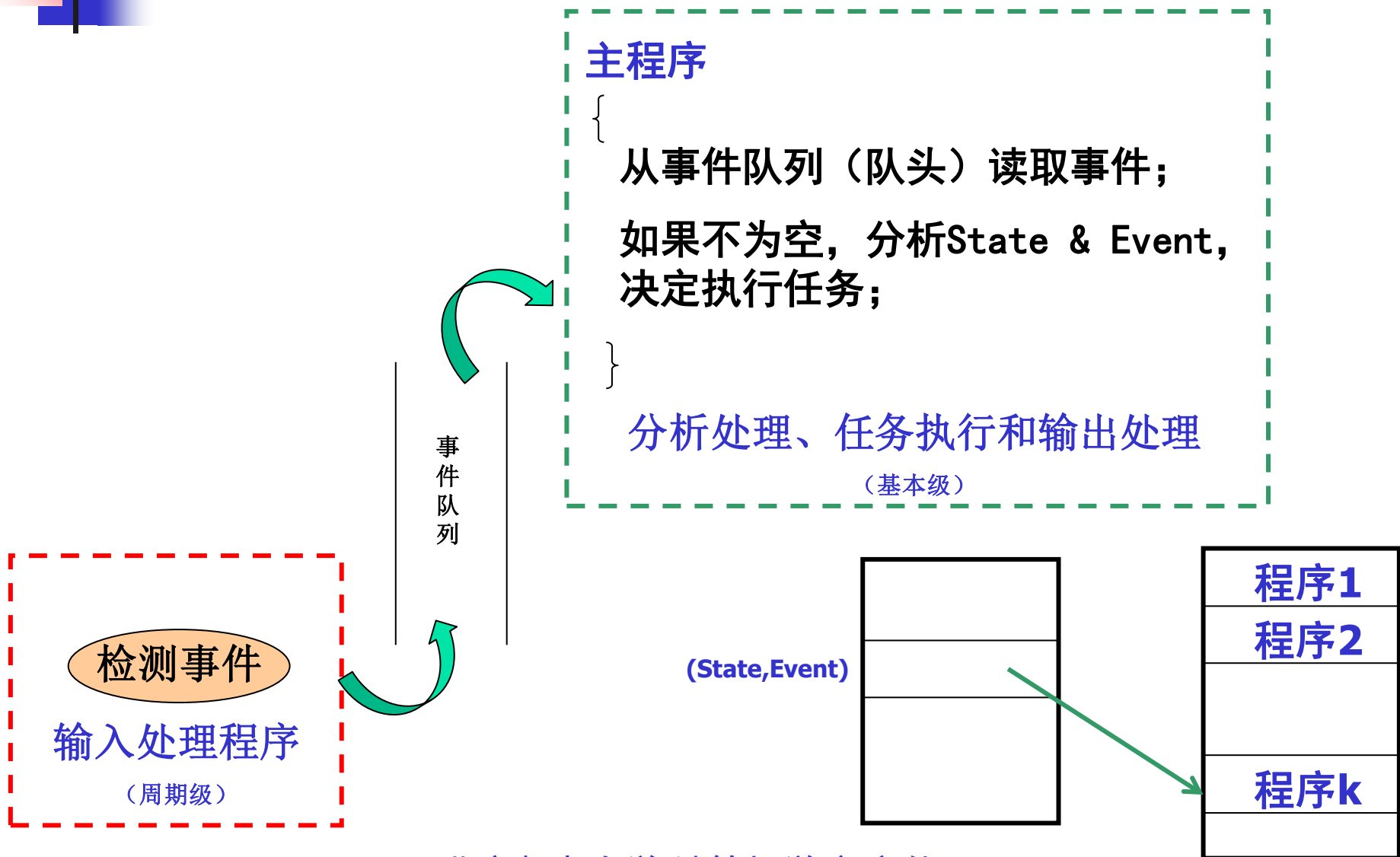
```

{
  Stop Ringing;
  Stop Return tone;
  Stop T3;
  Release Resource;
  State:=0;
}
    
```

基于多级表的有限状态机的实现



呼叫处理软件设计与实现





学习要求：

- 结合课堂所讲内容，认真阅读教材第三章，体会本章“小结”；
- 深刻理解下列重点内容：
 - 程控交换系统的体系结构
 - 用户电路、数字中继电路的工作原理
 - 控制系统多处理机的工作方式
 - 程控交换软件系统的特点
 - 呼叫处理的基本原理
- 完成作业。



作业

- 1、完成图3.36 “听忙音” 状态直到呼叫释放的呼叫处理的SDL图描述。
- 2、给出图3.36整个本局呼叫处理过程的程序架构，包括主程序、子程序和主要的数据结构。
- 3、某程控交换机装有32个模块，已知每 8个模块合用一台处理机， 每台处理机完成一次呼叫平均需要执行 36000条指令， 每条指令平均执行时间为1微秒， 固定开销 $a=0.15$ ，最大占用率 $t=0.95$ ， 试求该交换机总呼叫处理能力 N 值是多少？