



北京邮电大学  
Beijing University of Posts and Telecommunications

# Computer Architecture

## 计算机系统结构

(互联网)

北京邮电大学  
邢坚 2020年5月



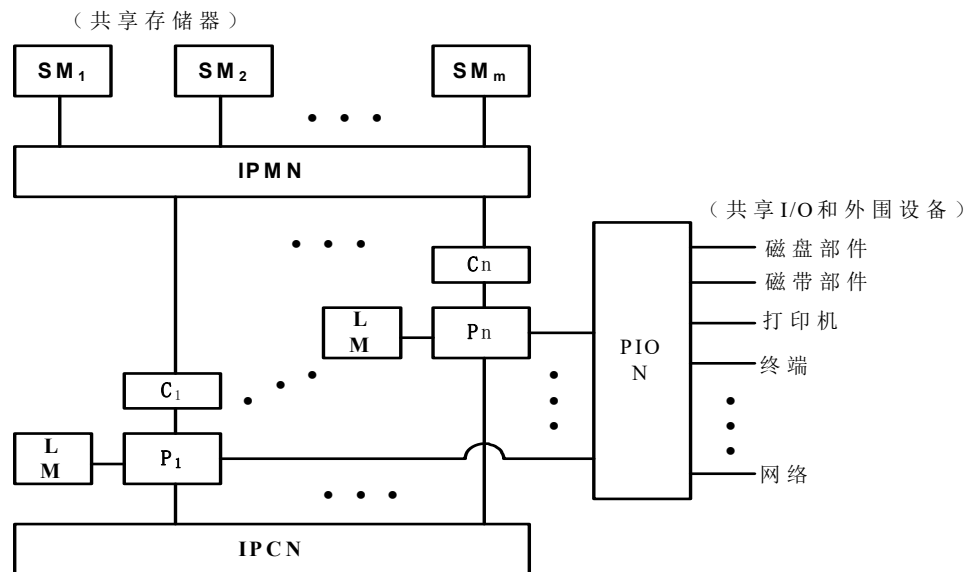
北京邮电大学  
Beijing University of Posts and Telecommunications

# Networks

Computer power increases by the square of the number of nodes on the network.

Robert Metcalf

# Networks



图注：IPMN(处理机-存储器网络)  
PION(处理机-I/O网络)  
IPCN(处理机之间通信网络)  
P (处理机)  
C (高速缓冲存储器)  
SM (共享存储器)  
LM (本地存储器)



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# Networks

## ■ 互联类型

- WAN -
- LAN -
- SAN – Storage or system area network

## ■ 互连网络的作用

- 互连网络是一种由开关元件按照一定的拓扑结构和控制方式构成的网络，用来实现计算机系统中节点之间的相互连接。
  - 互连网络已成为并行处理系统的核心组成部分，对整个计算机系统的性价比有着决定性的影响
  - 3大要素：互连结构，开关元件，控制方式



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# 互连网络的表示方法

## ■ 输入结点与输出结点之间的对应关系

- 互连函数表示法 - 互连函数反映了网络输入数组和输出数组之间对应的置换关系或排列关系，有时也称为置换函数或排列函数

$$f(x_{n-1} \dots x_1 x_0) = x_0 x_{n-2} \dots x_1 x_{n-1}$$

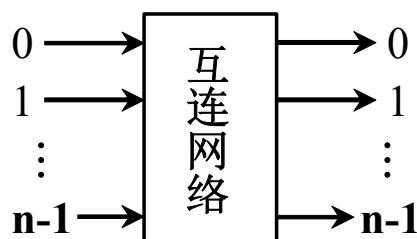
自变量和函数可以用二进制（十进制）表示

- 图形表示法

- 输入输出对应表示法

例如：恒等函数：

$$I(x_{n-1} x_{n-2} \dots x_1 x_0) = x_{n-1} x_{n-2} \dots x_1 x_0$$



输入：0 1 2 3 4 5 6 7

输出：1 0 3 2 5 4 7 6

图形表示法

输入输出对应表示法

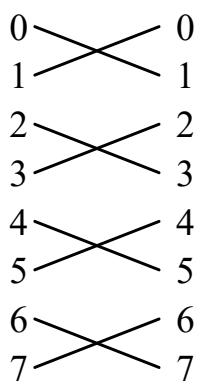


邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

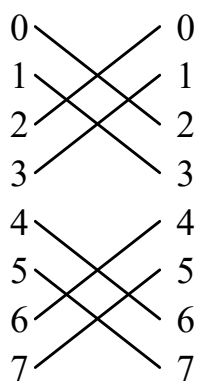
# 交换函数 (Exchange)

$$E_k(x_{n-1} \dots x_{k+1} x_k x_{k-1} \dots x_1 x_0) = x_{n-1} \dots x_{k+1} \bar{x}_k x_{k-1} \dots x_1 x_0$$

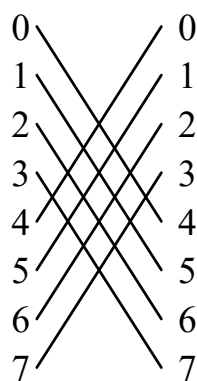
- 当n=3时，共有3种函数，每种函数能够表示8个结点(N=8)之间的连接关系。



E<sub>0</sub> 交换函数



E<sub>1</sub> 交换函数



E<sub>2</sub> 交换函数

$$E_0(x_2 x_1 x_0) = x_2 x_1 \bar{x}_0$$

$$E_1(x_2 x_1 x_0) = x_2 \bar{x}_1 x_0$$

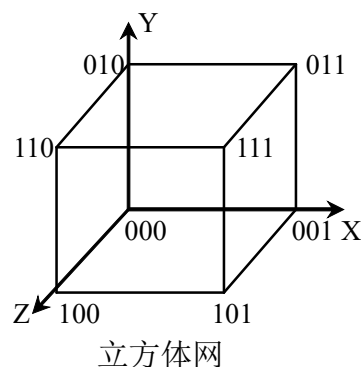
$$E_2(x_2 x_1 x_0) = \bar{x}_2 x_1 x_0$$



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# 交换函数 (Exchange)

- 由于交换函数主要用于超立方体互连网中，因此也称为超立方体函数



- 用Cube表示，如：Cube0、Cube1、Cube2等。
- 用交换函数构成的互连网络的结点度为n，网络直径也为n。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# 全混洗函数(Perfect shuffle)-均匀洗牌

$$S(x_{n-1}x_{n-2} \cdots x_1x_0) = x_{n-2}x_{n-3} \cdots x_1x_0x_{n-1} \quad (\text{循环左移})$$

- 子混洗(subshuffle)  $S_{(k)}$  最低k位循环左移
- 超混洗(supershuffle)  $S^{(k)}$  最高k位循环左移

$$S_{(k)}(x_{n-1}x_{n-2} \cdots x_kx_{k-1}x_{k-2} \cdots x_1x_0) = x_{n-1}x_{n-2} \cdots x_k \underline{x_{k-2} \cdots x_1x_0x_{k-1}}$$

$$S^{(k)}(x_{n-1}x_{n-2} \cdots x_{n-k}x_{n-k-1} \cdots x_1x_0) = \underline{x_{n-2} \cdots x_{n-k}x_{n-1}}x_{n-k-1} \cdots x_1x_0$$

- 逆混洗 (S的逆函数)

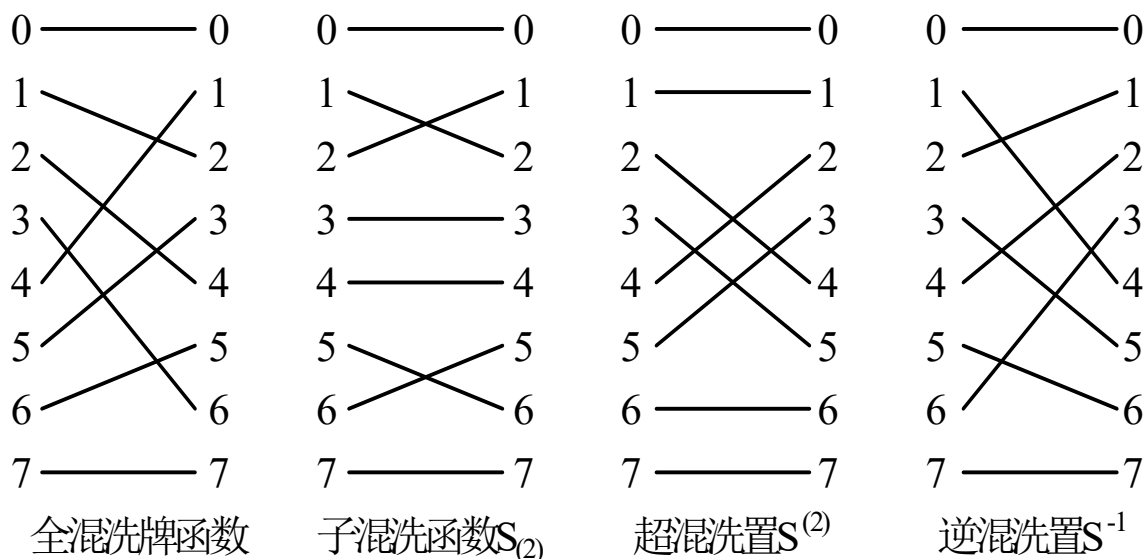
$$S^{-1}(x_{n-1}x_{n-2} \cdots x_1x_0) = x_0x_{n-1}x_{n-2} \cdots x_1$$



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心



# 全混洗函数(Perfect shuffle)



只用全混洗函数不能实现任意结点之间的互连，  
通常用全混洗函数与其他函数，如交换函数一起  
构成互连网络



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# 蝶式函数(Butterfly)

$$B(x_{n-1}x_{n-2} \cdots x_1x_0) = x_0x_{n-2} \cdots x_1x_{n-1} \quad (\text{最高位和最低位互换})$$

- 子蝶式(subbutterfly)  $B_{(k)}$  最低k位的最高位与最低位互换
- 超蝶式(superbutterfly)  $B^{(k)}$  最高k位的最高位与最低位互换

$$B_{(k)}(x_{n-1}x_{n-2} \cdots x_kx_{k-1}x_{k-2} \cdots x_1x_0) = x_{n-1}x_{n-2} \cdots x_k \underline{x_0x_{k-2} \cdots x_1x_{k-1}}$$

$$B^{(k)}(x_{n-1}x_{n-2} \cdots x_{n-k} \cdots x_1x_0) = \underline{x_{n-k}x_{n-2} \cdots x_{n-k-1}x_{n-1}x_{n-k-2} \cdots x_1x_0}$$



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# 反位序函数(Bit Reversal)

$$R(x_{n-1}x_{n-2} \cdots x_1x_0) = x_0x_1 \cdots x_{n-2}x_{n-1} \quad (\text{位序颠倒})$$

- 子反位序函数:  $R_{(k)}$  最低k位的位序颠倒
- 超反位序函数:  $R^{(k)}$  最高k位的位序颠倒

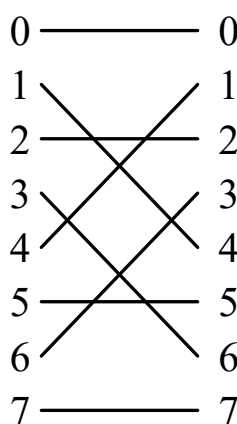
$$R_{(k)}(x_{n-1}x_{n-2} \cdots x_kx_{k-1}x_{k-2} \cdots x_1x_0) = x_{n-1}x_{n-2} \cdots x_k \underline{x_0x_1 \cdots x_{k-2}x_{k-1}}$$

$$R^{(k)}(x_{n-1}x_{n-2} \cdots x_{n-k}x_{n-k+1} \cdots x_1x_0) = \underline{x_{n-k}x_{n-k+1} \cdots x_{n-2}x_{n-1}x_{n-k-1} \cdots x_1x_0}$$

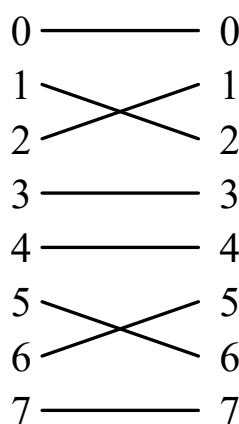


邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

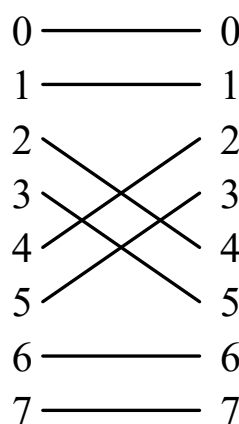
## 蝶式 & 反位序



B=R函数



B<sub>(2)</sub>=R<sub>(2)</sub>函数



B<sup>(2)</sup>=R<sup>(2)</sup>函数

与全混洗函数类似，只用蝶式函数也不能实现任意结点之间的互连。

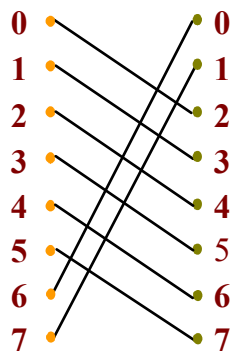


邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

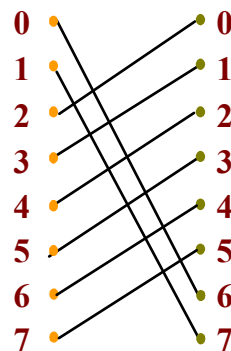
# 移数函数

## ■ 函数式:

- $a(x) = (x \pm k) \bmod N$   $1 \leq x \leq N-1$ ,  $1 \leq k \leq N-1$
- 将各输入端都错开一定的位置（模N）后连到输出端。



(a) 左移移数函数  $k=2$



(b) 右移移数函数  $k=2$



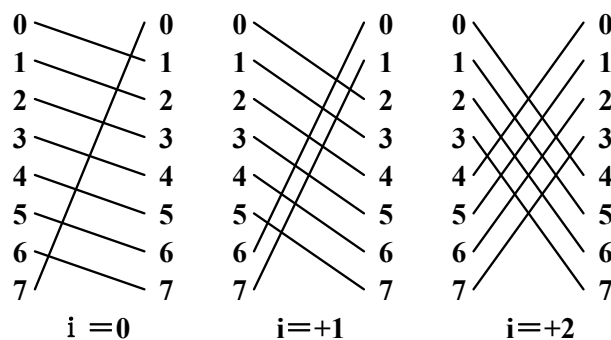
邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# PM2I函数 – 特殊的移数函数

$$PM2_{+i}(x) = (x + 2^i) \bmod N$$

$$PM2_{-i}(x) = (x - 2^i) \bmod N, 0 \leq x \leq N-1, 0 \leq i \leq n-1, n = \log_2 N$$

- N为节点数。N=8时



- 采用移数函数可以构成环型网（包括单向环网、双向环网、弦环网）、方格网、移数网等。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

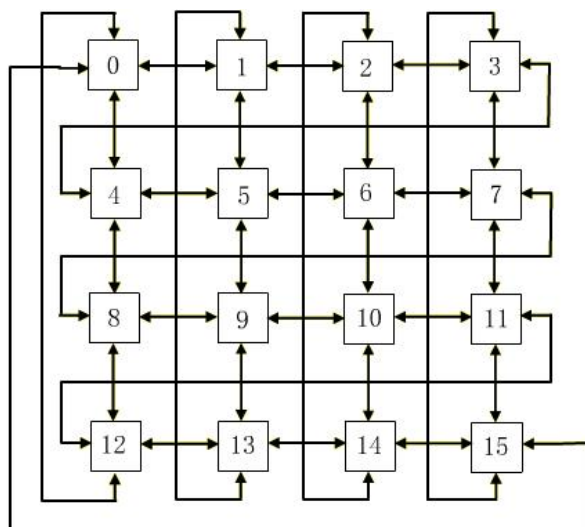
# PM2I函数

- PM2I互连网络共有 $2n$ 个互连函数。
- 当 $N=8$ 时( $n=3$ )，有6个PM2I函数：
  - $PM2_{+0}$ : ( 0 1 2 3 4 5 6 7 )
  - $PM2_{-0}$ : ( 7 6 5 4 3 2 1 0 )
  - $PM2_{+1}$ : ( 0 2 4 6 ) ( 1 3 5 7 )
  - $PM2_{-1}$ : ( 6 4 2 0 ) ( 7 5 3 1 )
  - $PM2_{+2}$ : ( 0 4 ) ( 1 5 ) ( 2 6 ) ( 3 7 )
  - $PM2_{-2}$ : ( 4 0 ) ( 5 1 ) ( 6 2 ) ( 7 3 )



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# PM2I函数



## 网络函数

- $PM2_{+0}$ :  
(0 1 2 ... 15)
- $PM2_{-0}$ :  
(15 14 13 ... 0)
- $PM2_{\pm 2}$ :  
(0 4)(1 5)(2 6)(3 7)  
(4 8)(5 9)(6 10)(7 11)  
(8 12)(9 13)(10 14)(11 15)  
(12 0)(13 1)(14 2)(15 3)



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心



## 互连函数(举例)

有16个处理器，编号分别为0, 1, ..., 15，用一个N=16的互连网络互连。当该互连网络实现的互连函数分别为：

- (1)  $Cube_3$
- (2)  $PM2_{+3}$
- (3)  $PM2_{-0}$
- (4) S
- (5) S(S)

分别给出与第13号处理器所连接的处理器号。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

## 互连函数(举例)

解：(1) 由  $Cube_3(x_3x_2x_1x_0) = \bar{x}_3x_2x_1x_0$

得  $Cube_3(1101) = 0101$ ，即处理器13连接到处理器5。

令  $Cube_3(x_3x_2x_1x_0) = 1101$ ，得  $x_3x_2x_1x_0 = 0101$ ，故与处理器13相连的是处理器5。

所以处理器13与处理器5双向互连。

(2) 由  $PM2_{+3} = (j+2^3) \bmod 16$ ，得  $PM2_{+3}(13) = (13+2^3) \bmod 16 = 5$ ，即处理器13连接到处理器5。

$PM2_{+3}(j) = (j+2^3) \bmod 16 = 13$ ，得  $j = 5$ ，故与处理器13相连的是处理器5。

所以处理器13与处理器5双向互连。

(3) 由  $PM2_{-0}(j) = (j-2^0) \bmod 16$ ，得  $PM2_{-0}(13) = 13-2^0 = 12$ ，即处理器13连接到处理器12。

令  $PM2_{-0}(j) = (j-2^0) \bmod 16 = 13$ ，得  $j = 14$ ，故与处理器13相连的是处理器14。

所以处理器13连至处理器12，而处理器14连至处理器13。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

## 互连函数(举例)

(4) 由 $S(x_3x_2x_1x_0) = x_2x_1x_0x_3$ , 得 $S(1101) = 1011$ , 即处理器13连接到处理器11。

令 $S(x_3x_2x_1x_0) = 1101$ , 得 $x_3x_2x_1x_0 = 1110$ , 故与处理器13相连的是处理器14。

所以处理器13连至处理器11, 而处理器14连至处理器13。

(5) 由 $S(S(x_3x_2x_1x_0)) = x_1x_0x_3x_2$ , 得 $S(S(1101)) = 0111$ , 即处理器13连接到处理器7。

令 $S(S(x_3x_2x_1x_0)) = 1101$ , 得 $x_3x_2x_1x_0 = 0111$ , 故与处理器13相连的是处理器7。

所以处理器13与处理器7双向互连。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

## 互连网络的结构参数

- 网络通常是用有向边或无向边连接有限个节点的图来表示
  - **网络规模N**: 网络中结点的个数(网络所能连接的部件的数量)。
  - **结点度d**: 与结点相连接的边数(通道数), 包括入度和出度。进入结点的边数叫入度, 从结点出来的边数叫出度。
  - **节点距离**: 网络中的任意两个节点, 从一个节点出发到另一个节点终止所需要跨越的边数的最小值。
  - **网络直径D**: 网络中任意两个结点之间距离的最大值。
  - **结点间的线长**: 两个结点间连线的长度。
  - **对称性**: 从任何结点看到拓扑结构都是一样的网络称为对称网络。
  - **等分宽度b**: 当某一网络被切成相等的两半时, 沿切口的最小边数(通道)称为通道等分宽度, 用b表示。而线等分宽度就是 $B = b \times w$ 。其中w为通道宽度(用位表示)。该参数主要反映了网络最大流量。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# 互连网络的性能指标

## ■ 评估互连网络性能的两个基本指标：时延和带宽

### ■ **通信时延** - 指从源节点到目的节点传送一条消息所需的总时间，它由以下4部分构成：

- **软件开销**：在源节点和目的节点用于收发消息的软件所需的执行时间。主要取决于两端端节点处理消息的软件内核。
- **通道时延**：通过通道传送消息所花的时间。
  - 通路时延 = 消息长度/通道带宽
  - 通常由瓶颈链路的通道带宽决定
- **选路时延**：消息在传送路径上所需的一系列选路决策所需的时间开销。与传送路径上的节点数成正比。
- **竞争时延**：多个消息同时在网络中传送时，会发生争用网络资源的冲突。为避免或解决争用冲突所需的时间就是竞争时延。很难预测，它取决于网络的传输状态。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# 互连网络的性能指标

### ■ **网络时延** - 通道时延与选路时延的和。

- 它是由网络硬件特征决定的，与程序行为和网络传输状态无关。

### ■ **端口带宽** - 对于互连网络中的任意一个端口来说，其端口带宽是指单位时间内从该端口传送到其他端口的最大信息量。

- 在对称网络中，端口带宽与端口位置无关。网络的端口带宽与各端口的端口带宽相同。
- 非对称网络的端口带宽则是指所有端口带宽的最小值。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

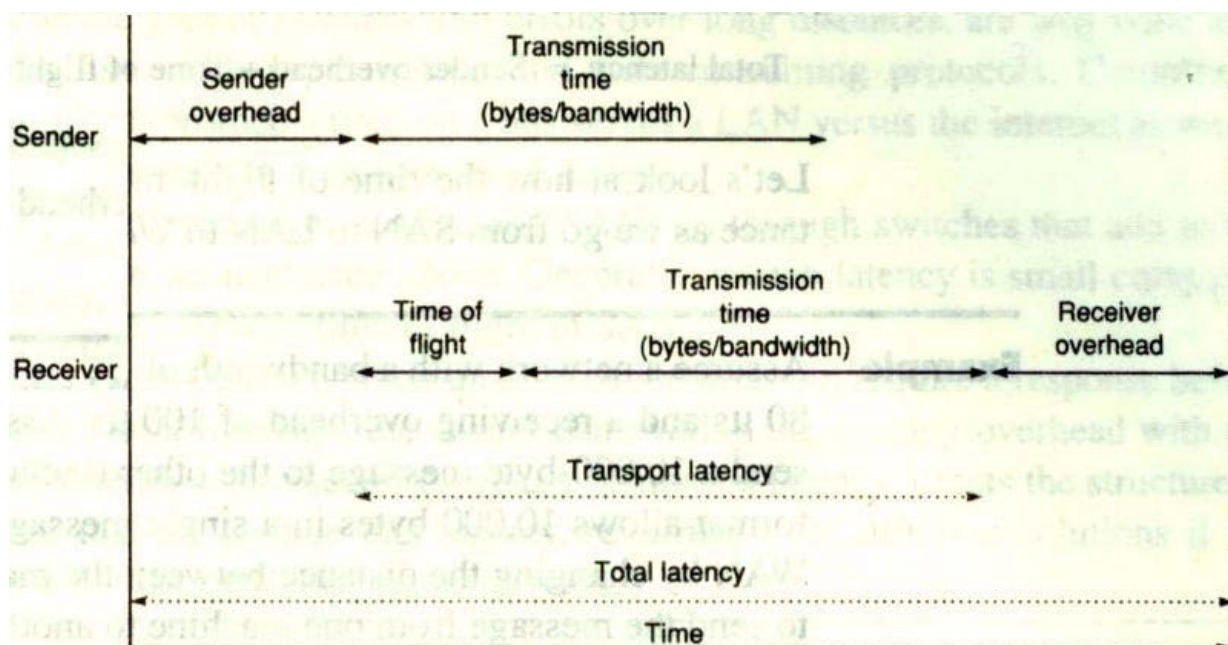
# 互连网络的性能指标

- **聚集带宽** - 网络从一半节点到另一半节点，单位时间内能够传送的最大信息量。
  - 例如，HPS是一种对称网络。网络规模N的上限：512，端口带宽：40MB/s，则：
    - HPS的聚集带宽： $(40\text{MB/s} \times 512) / 2 = 10.24\text{GB/s}$
- **等分带宽** - 与等分宽度对应的切平面中，所有边合起来单位时间所能传送的最大信息量。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

## 互连网络的性能参数（点对点）



总时延 = 发送方开销 + 飞行时间 + 消息长度/带宽 + 接收方开销



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心



# 互连网络分类

- 静态互连网络(Static Networks)
  - 各节点之间有固定的连接通路、且在运行中不能改变的网络。
- 动态互连网络(Dynamic Networks)
  - 由交换开关构成、可按运行程序的要求动态地改变连接状态的网络。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

## Static Networks

- 每一个开关元件固定地与一个结点相连，建立该结点与邻近结点之间的连接通路，直接实现两结点之间的通信。这种网络比较适合于构造通信模式可预测或可用静态连接实现的计算机。
  - 一维的有线性阵列结构；
  - 二维的有环形、星形、树形、网格形等；
  - 三维的有立方体等；
  - 三维以上的有超立方体等。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

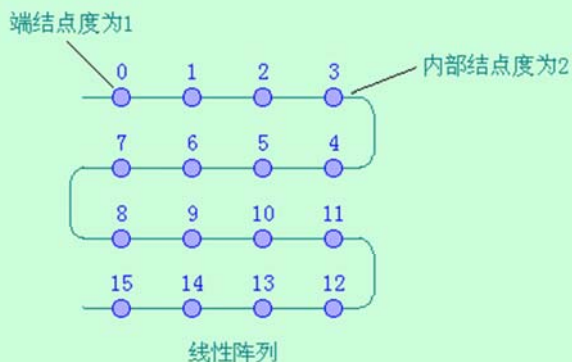


# Static Networks

- 线性阵列 一种一维的线性网络，其中 $N$ 个节点用 $N-1$ 个链路连成一行。

- 端节点的度：1
- 其余节点的度：2
- 直径： $N-1$
- 等分宽度 $b=1$

## 线性阵列



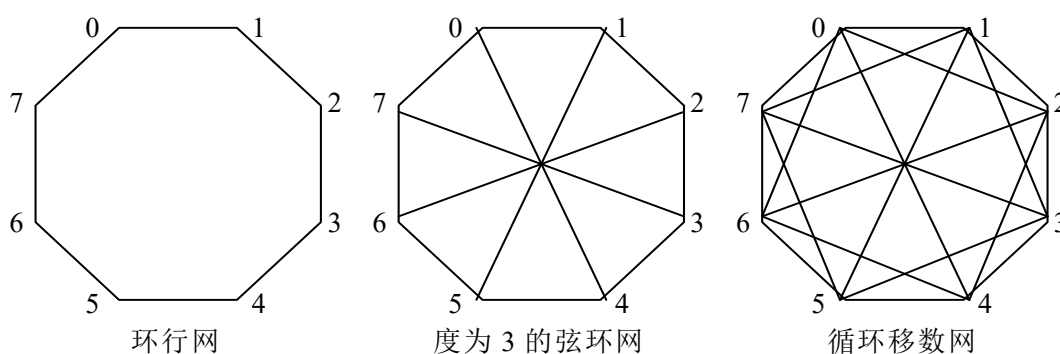
### 线性阵列与总线的区别：

总线是通过切换与其连接的许多结点来实现时分特性的  
线性阵列允许不同的源结点和目的结点对并行地使用其不同的部分



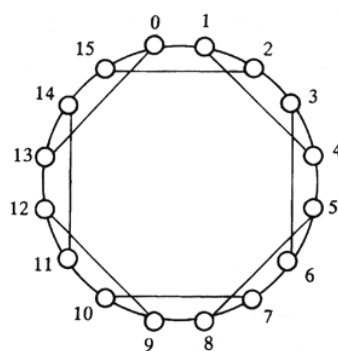
邱坚 北京邮电大学 计算机学院/软件学院 物

# Static Networks

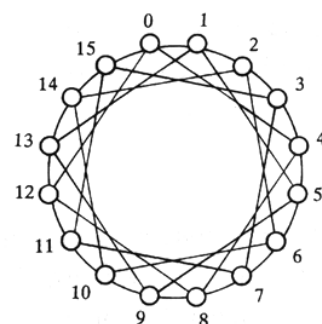


- 以上网络的

- 维数？
- 网络函数？
- 结点度？
- 网络直径？
- 对称性？



(c)度为3的带弦环

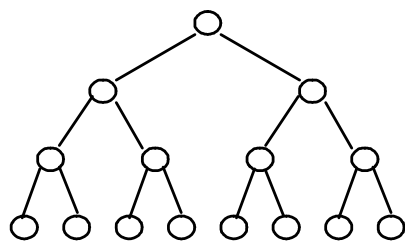


(d)度为4的带弦环(与 Illiac 网相同)

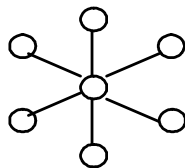


邱坚 北京邮电大学 计算机学院/软件学院

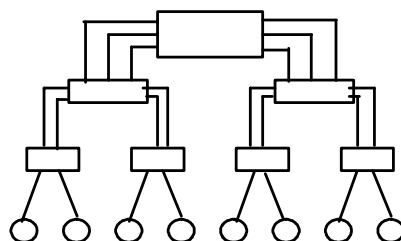
# Static Networks



二叉树网



星形网



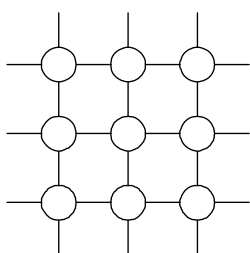
二叉胖树网

- 一棵 $k$ 层二叉树有 $N=2^k-1$ 个结点，结点度是3，直径是 $2(k-1)$ ；
- 星形是一种特殊的2层树，结点度很高，为 $d=N-1$ ，直径是2；
- 二叉胖树的结点度从叶子结点往根结点逐渐增加；胖树缓解了一般二叉树根结点通信速度高的矛盾；

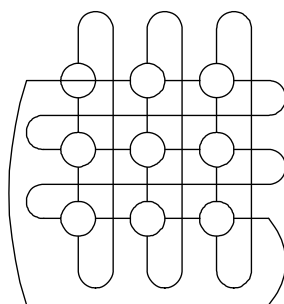


邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

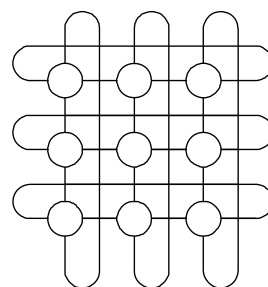
# Static Networks



网格形



Illiac网

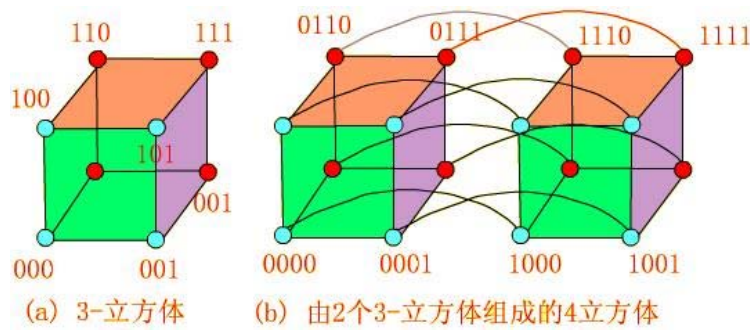


环形网格网



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# Static Networks

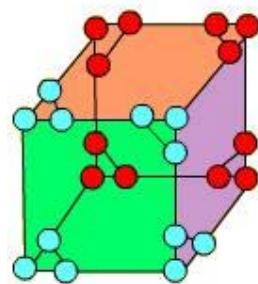


- $n$ 维立方体由 $N=2^n$ 个结点构成，分布在 $n$ 维上，每维有两个结点；超立方体网采用交换函数，**结点度为 $n$** ，**直径也为 $n$** ；
- 相邻结点的二进制编号仅差一位，而两个结点间的**距离正好等于这两个结点二进制编号间不同的位数**；
- 在两个结点之间总存在 **$n$ 条不同的路径可走**，因而容错性能好。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# Static Networks



## 带环的3维立方体

带环3-立方体

- 带环3维立方体(3-CCC)。它把超立方体的每个结点换成一个有 $n$ 个结点的环。一个3维立方体可改成带环3维立方体(3-CCC)。这样一来，**不管 $n$ 维立方体的 $n$ 有多大，所有的结点的度数仍都是3**

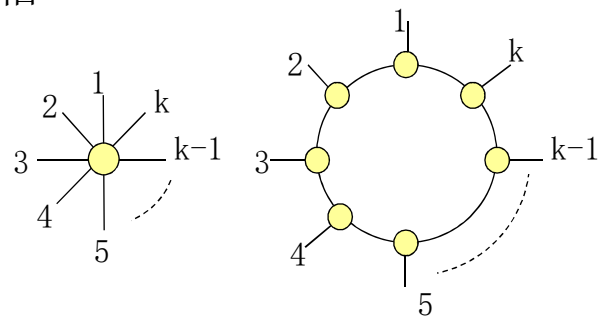


邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# Static Networks

## ■ 带环k-立方体（简称k-CCC）

- k-立方体的变形，它是通过用k个节点构成的环取代k-立方体中的每个节点而形成的。
- 网络规模为 $N=k \times 2k$
- 网络直径为 $D=2k-1 + \lfloor k/2 \rfloor$ 
  - 比k-立方体的直径大一倍
- 等分宽度为 $b=N/(2k)$

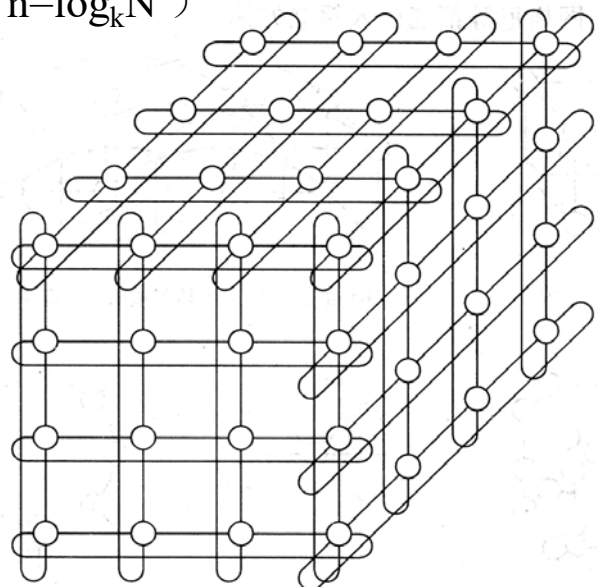


邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# Static Networks

## ■ k元n-立方体网络，其中参数n是立方体的维数，k是基数，即每一维上的节点个数。

- $N=k^n$  , (  $k=\sqrt[n]{N}$  ,  $n=\log_k N$  )
- 4元3-立方体网络:



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统



# Static Networks

## ■ k元n-立方体网络

- k元n-立方体的节点可以用基数为k的n位地址  $A = a_1a_2...a_n$  来表示。
  - 其中  $a_i$  表示该节点在第i维上的位置
- 通常把低维k元n-立方体称为环网，而把高维k元n-立方体称为超立方体。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

## 静态互连网络特征一览表

网络类型	节点度d	网络直径D	链路数l	等分宽度B	对称性	网络规格说明
线线阵列	2	$N-1$	$N-1$	1	非	N个节点
环形	2	$\lceil N/2 \rceil$	N	2	是	N个节点
全连接	$N-1$	1	$N(N-1)/2$	$(N/2)^2$	是	N个节点
二叉树	3	$2(h-1)$	$N-1$	1	非	树高 $h = \lceil \log_2 N \rceil$
星形	$N-1$	2	$N-1$	$\lceil N/2 \rceil$	非	N个节点
2D网格	4	$2(r-1)$	$2N-2r$	r	非	$r \times r$ 网格, $r = \sqrt{N}$
Illiac网	4	$r-1$	$2N$	$2r$	非	与 $r = \sqrt{N}$ 的带弦环等效
2D环网	4	$2\lceil r/2 \rceil$	$2N$	$2r$	是	$r \times r$ 环网, $r = \sqrt{N}$
超立方体	n	n	$nN/2$	$N/2$	是	N个节点, $n = \lceil \log_2 N \rceil$ (维数)
CCC	3	$2k-1 + \lceil k/2 \rceil$	$3N/2$	$N/(2k)$	是	$N = k \times 2^k$ 节点 环长 $k \geq 3$
k元n-立方体	$2n$	$n\lceil k/2 \rceil$	$nN$	$2k^{n-1}$	是	$N = k^n$ 个节点



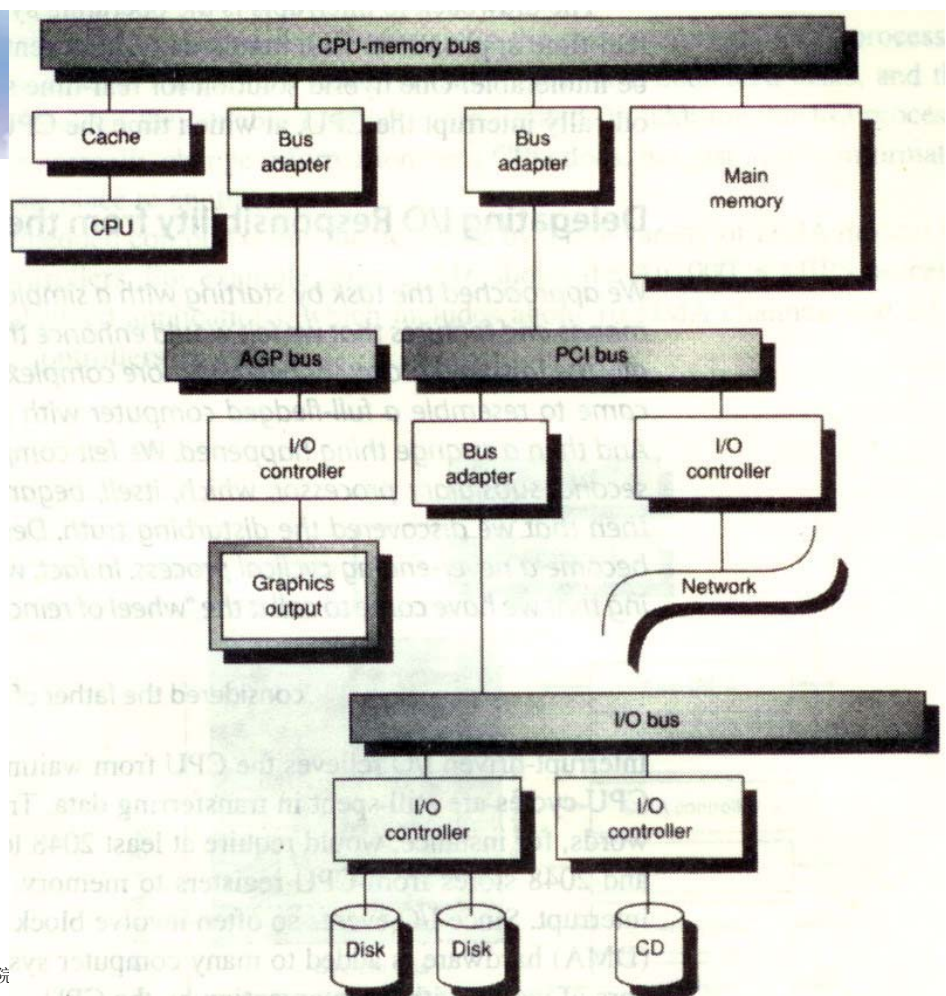
# Dynamic networks

- 总线网络(BUS)
- 交叉开关网络(crossbar)
- 多级互连网络(MIN – Multistage Interconnection Network)



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

## ■ 总线系统



邱坚 北京邮电大学 计算机学院

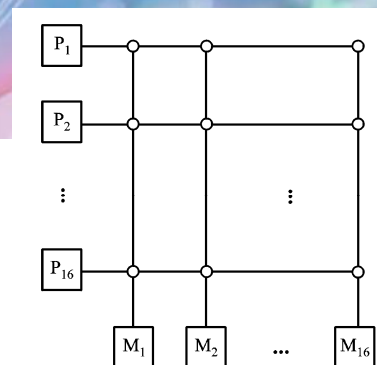
# Dynamic networks

- 总线系统实际上是一组导线和插座用于处理与总线相连的处理机、存储模块和外围设备间的数据业务
- 在多个请求情况下，总线仲裁逻辑必须每次能将总线服务分配或重新分配给一个请求，按分时工作原理进行工作 - 争用总线/时分总线
- 总线系统与其他两种动态连接网络相比，其结构简单、实现成本低、带宽较窄



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# Dynamic networks



- 交叉开关网络
  - 交叉点开关能在对偶（源、目的）之间形成动态连接，同时实现多个对偶之间的无阻塞连接。
  - 带宽和互连特性最好。
  - 一个 $n \times n$ 的交叉开关网络，可以无阻塞地实现 $n!$ 种置换。
  - 对一个 $n \times n$ 的交叉开关网络来说，需要 $n^2$ 套交叉点开关以及大量的连线。
    - 当 $n$ 很大时，交叉开关网络所需要的硬件数量非常巨大。

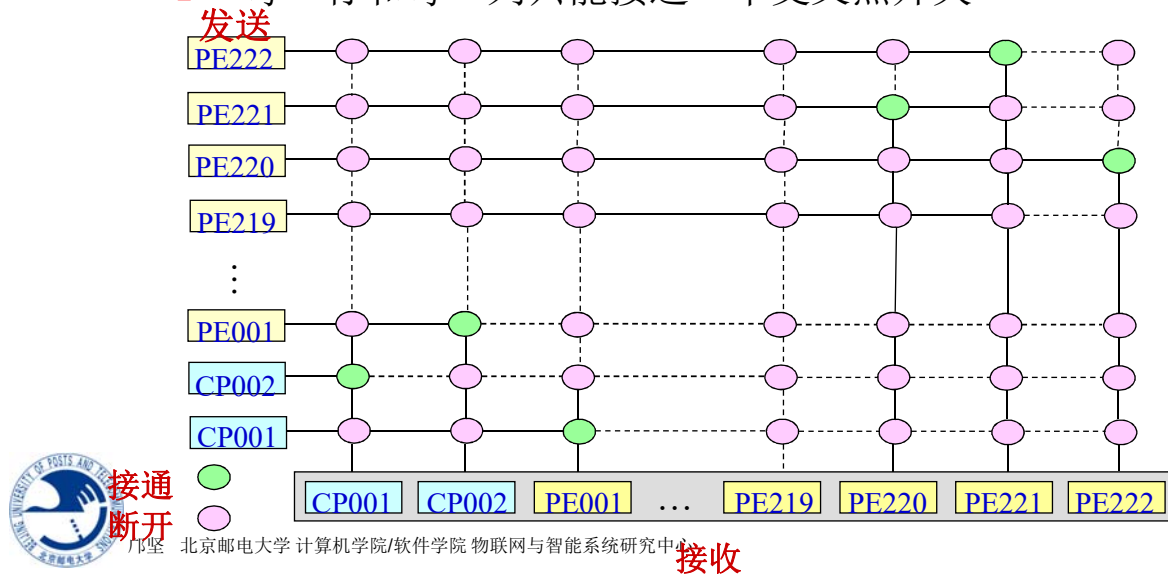


邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# Dynamic networks

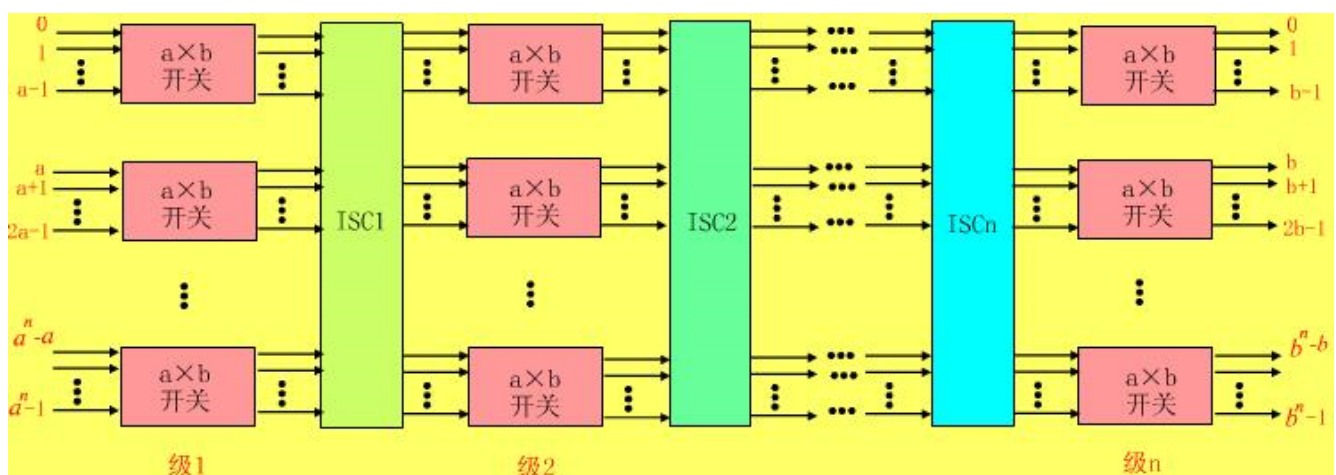
- Fujitsu公司制造的向量并行处理机VPP500所采用的大型交换开关网络 ( $224 \times 224$ )

- PE: 带存储器的处理机
- CP: 控制处理机
- 每一行和每一列只能接通一个交叉点开关



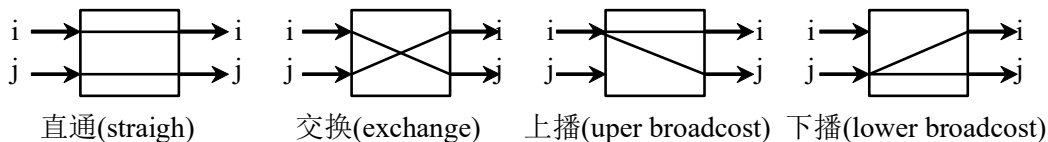
# Dynamic networks

- 多级互连网络MIN
- 多级互连网络采用的关键技术:
  - 交换开关;
  - 交换开关之间的拓扑连接;
  - 对交换开关的不同控制方式。



# Dynamic networks

- **交换开关** - 一个 $a \times b$ 交换开关有 $a$ 个输入和 $b$ 个输出。每个输入可与一个或多个输出相连，但是在输出端必须避免发生冲突。
- 一对一、一对多映射是允许的；但**不允许有多对一映射**。
- 只容许一对一映射时称为置换连接，称这种开关为 $n \times n$ 交叉开关。最常用的二元开关： $a=b=2$ 。



模块大小	合法状态	置换连接
$2 \times 2$	4	2
$4 \times 4$	256	24
$8 \times 8$	16 777 216	40 320
$n \times n$	$n^n$	$n!$



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# Dynamic networks

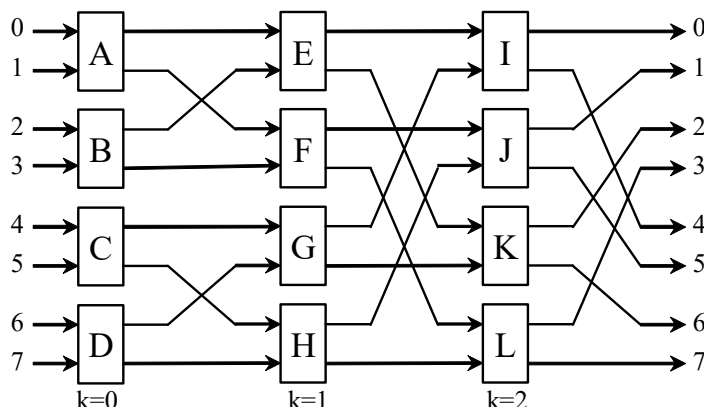
- **控制方式** - 在多级互连网络中，有多级交换开关，每一级又有多个交换开关
  - 通常有三种控制方式
    - 级控制：同一级交换开关使用同一个控制信号控制。
    - 单元级控制：每个交换开关分别控制。
    - 部分级控制：例如，第 $i$ 级使用 $i+1$ 个控制信号控制 ( $0 \leq i \leq n-1$ )
- **拓扑结构** - 前一级交换开关的输出端与后一级交换开关的输入端之间的连接模式
  - ISC通常采用前面介绍的互连函数实现拓扑结构



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心



# Dynamic networks



## ■ 多级立方体网

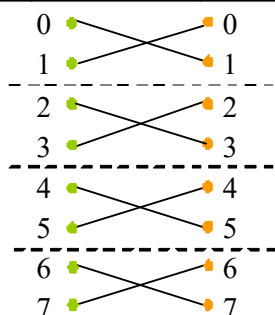
- 当所有开关都直通时，实现恒等变换；
- 当A、B、C、D四个开关交换，其余直通时实现C0互连函数；
- 当E、F、G、H四个开关交换，其余直通时实现C1互连函数；
- 当I、J、K、L四个开关交换，其余直通时实现C2互连函数。
- 也可以通过选择不同的控制方式，构成不同的互连网络：
  - 采用级控制，可以构成交换网；
  - 采用部分级控制，可以构成移数网；
  - 采用单元控制，可以构成二进制n方体网。



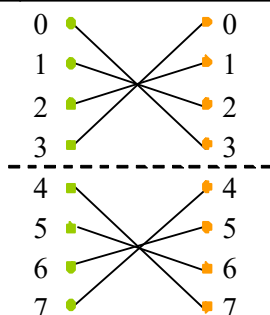
邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# Dynamic networks

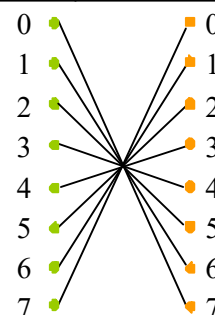
IJKL	EFGH	ABCD	互连关系（交换）	互连函数
0	0	0	(0)(1)(2)(3)(4)(5)(6)(7)	$\equiv$
0	0	1	(0,1) (2,3) (4,5) (6,7)	$E_0$
0	1	0	(0,2) (1,3) (4,6) (5,7)	$E_1$
0	1	1	(0,3) (1,2) (4,7) (5,6)	$E_0+E_1$
1	0	0	(0,4) (1,5) (2,6) (3,7)	$E_2$
1	0	1	(0,5) (1,4) (2,7) (3,6)	$E_0+E_2$
1	1	0	(0,6) (1,7) (2,4) (3,5)	$E_1+E_2$
1	1	1	(0,7) (1,6) (2,5) (3,4)	$E_0+E_1+E_2$



(a) 4 组 2 元交换



(b) 2 组 4 元交换



(c) 1 组 8 元交换



邱坚



# Dynamic networks

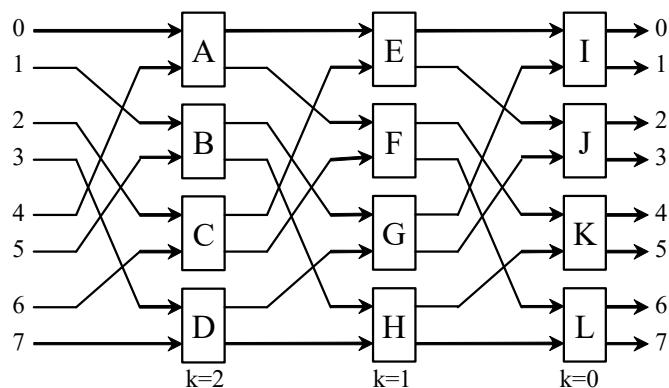
ABCD	EG	FH	I	J	KL	互连关系（右循环）	互连函数	功能说明
1	0	0	1	0	0	(0,1,2,3,4,5,6,7)	$PM_{2+0}$	右循环1位
0	0	1	1	1	0	(0,2,4,6)(1,3,5,7)	$PM_{2+1}$	右循环2位
0	1	0	1	1	1	(0,4)(1,5)(2,6)(3,7)	$PM_{2+2}$	右循环4位
1	1	0	0	0	0	(0,1,2,3)(4,5,6,7)	$PM_{1+0(2)}$	2组右环2位
0	0	1	0	0	0	(0,2)(1,3)(4,6)(5,7)	$PM_{2+1(2)}$	2组右环1位
1	0	0	0	0	0	(0,1)(2,3)(4,5)(6,7)	$PM_{2+0(1)}$	4组右环1位

- 第*i*级用*i*+1个控制信号，因此共有6个控制信号，有64种不同的控制。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# Dynamic networks



- Omega网**
  - 8入8出，ISC是对8个对象的均匀洗牌模式
  - 开关模块采用单元控制方式
  - 能够实现任意一个输入端到任意一个输出端的连接。
  - 不能同时实现多个输入端到多个输出端的连接。例如：同时实现0→6和4→7有冲突
  - 能够实现从任意一个输入端到所有输出端的广播



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# Dynamic networks

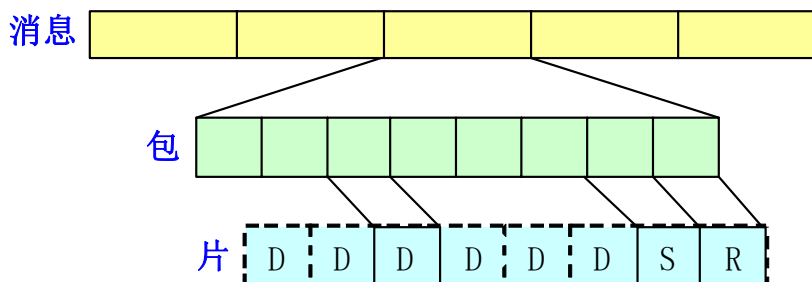
网络特性	总线系统	多级网络	交叉开关
单位数据传送的最小时延	恒定	$O(\log_k n)$	恒定
每台处理机的带宽	$O(w/n)$ 至 $O(w)$	$O(w)$ 至 $O(nw)$	$O(w)$ 至 $O(nw)$
连线复杂性	$O(w)$	$O(nw \log_k n)$	$O(n^2 w)$
开关复杂性	$O(n)$	$O(n \log_k n)$	$O(n^2)$
连接特性和寻径性能	一次只能一对一	只要网络不阻塞，可实现某些置换和广播	全置换，一次一个
典型计算机	Symmetry S1, Encore Multimax	BBNTC-2000 IBM RP3	Cray Y-MP/816 Fujitsu VPP 500
说明	总线上假定有 $n$ 台处理机；总线宽度为 $w$ 位	$n \times n$ MIN采用 $k \times k$ 开关，其线宽为 $w$ 位	假定 $n \times n$ 交叉开关的线宽为 $w$ 位



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

## 消息传递机制

- 当源节点和目的节点之间没有直接的连接时，消息需要经过中间的节点进行传递。  
**寻径**就是用来实现这种传递的通信方法和算法。有的称之为**路由**。
- 消息(节点之间进行通信的逻辑单位)的格式



R: 寻径信息

S: 顺序号

D: 数据片



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# 消息传递机制 - 寻径方式

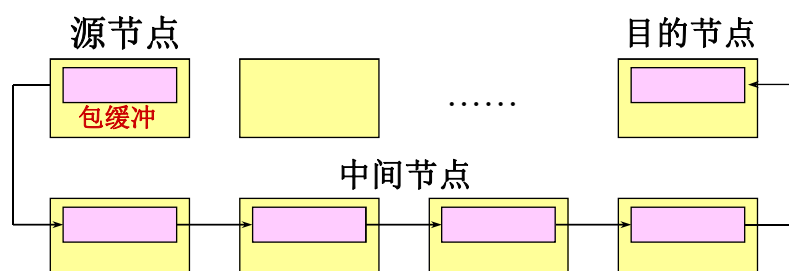
- **线路交换(Circuit switching)**: 在传递一个信息之前, 需要先建立一条从源节点到目的节点的物理通路, 然后再传递信息。
  - 优点: 传输带宽较大, 平均传输时延较小, 而且使用的缓冲区小。
    - 适合于具有动态和突发性的大规模并行处理数据的传送。
  - 缺点: 需要频繁地建立源节点到目的节点的物理通路, 时间开销会很大。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# 消息传递机制 - 寻径方式

- **存储转发(Store-and-forward)**: 最简单的分组交换方式。
  - 包是信息传递的基本单位。包从源节点经过一系列中间节点到达目的节点。
  - 要求: 所经过的每个中间节点都要设置一个包缓冲器, 用于保存所传递的包。当一个包到达某个中间节点时, 该节点先把这个包全部存储起来, 然后在出口链路可用、而且下一个节点的包缓冲器也可用的情况下, 传递给下一个节点。
  - 缺点
    - 包缓冲区大, 不利于VLSI实现;
    - 网络时延大, 与节点距离成正比。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# 消息传递机制 - 寻径方式

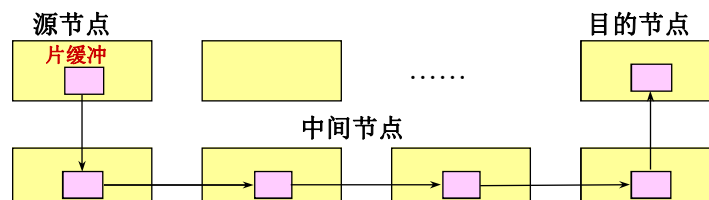
- **虚拟直通(Virtual cut-through)**: 对存储转发方式的一种改进, 减少了网络时延。
  - 没有必要等到信息包全部放入缓冲器后再作路由选择, 只要接收到用作寻径的包头, 就可作出判断。
  - 如果节点的输出链路空闲, 信息包可以不必存储在该节点的缓冲器中, 而是立即传送到下一个节点。
  - 如果整条链路都空闲, 包就可以立即直达目的节点。
  - 当出现寻径阻塞时, 虚拟直通方式需要将整个信息包全部存储在寻径节点中, 要求每个节点都有足够大的缓冲区。
    - 不利于VLSI的实现



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

# 消息传递机制 - 寻径方式

- **虫蚀方式(Wormhole)**: 把信息包“切割”成更小的单位 - “片”, 而且使信息包中各片的传送按流水方式进行。
  - 可以减少节点中缓冲器的容量, 缩短传送延迟时间
  - 处理的最小信息单位是“片”。当一个节点把头片送到下一个节点后, 那么接下来就可以把后面的各个片也依次送出
  - 一个节点一旦开始传送一个包中的头片后, 这个节点就必须等待这个包的所有片都送出去后, 才能传送其他包。不同包的片不能混合在一起传送。
  - 优点
    - 每个节点的缓冲器较小, 易于VLSI实现; 有较小的网络传输延迟; 通道共享性好, 利用率高; 易于实现选播和广播通信模式。
  - 缺点
    - 当消息的一片被阻塞时, 整个消息的所有片都将被阻塞在所在节点, 占用了节点资源。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

## 消息传递机制

- **确定性寻径**：通信路径完全由源节点地址和目的地址来决定，也就是说，寻径路径是预先唯一地确定好了的，而与网络的情况无关。
- **自适应寻径**：通信的通路每一次都要根据资源或者网络的情况来选择。
  - 可以避开拥挤的或者有故障的节点，使网络的利用率得到改进。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心

## 消息传递机制

### ■ 超立方体寻径

考虑一个由 $N=2^n$ 个节点构成的 $n$ 方体，每个节点的编号是形为 $b=b_{n-1}b_{n-2}\dots b_1b_0$ 的二进制编码。

设：源节点 $s=s_{n-1}s_{n-2}\dots s_1s_0$

目的节点 $d=d_{n-1}d_{n-2}\dots d_1d_0$

现在要确定一条从 $s$ 到 $d$ 的步数最少的路径。



邱坚 北京邮电大学 计算机学院/软件学院 物联网与智能系统研究中心



# 消息传递机制

将这个 $n$ 方体的各维表示成 $i=1, 2, \dots, n$ ，其中第 $i$ 维对应于节点地址中的第 $i-1$ 位。

设 $v=v_{n-1}v_{n-2}\dots v_1v_0$ 是路径中的任一节点。路径可以根据以下算法唯一地确定：

① 计算方向位 $r_i = s_{i-1} \oplus d_{i-1}$ ，其中 $i=1, 2, \dots, n$ 。

令 $v=s$ ， $i=1$ ，反复执行以下步骤：

② 如果 $r_i=1$ ，则从当前节点 $v$ 寻径到下一节点；否则，就跳过这一步。

③  $i \leftarrow i+1$ 。如果 $i \leq n$ ，则转第②步，否则退出。

