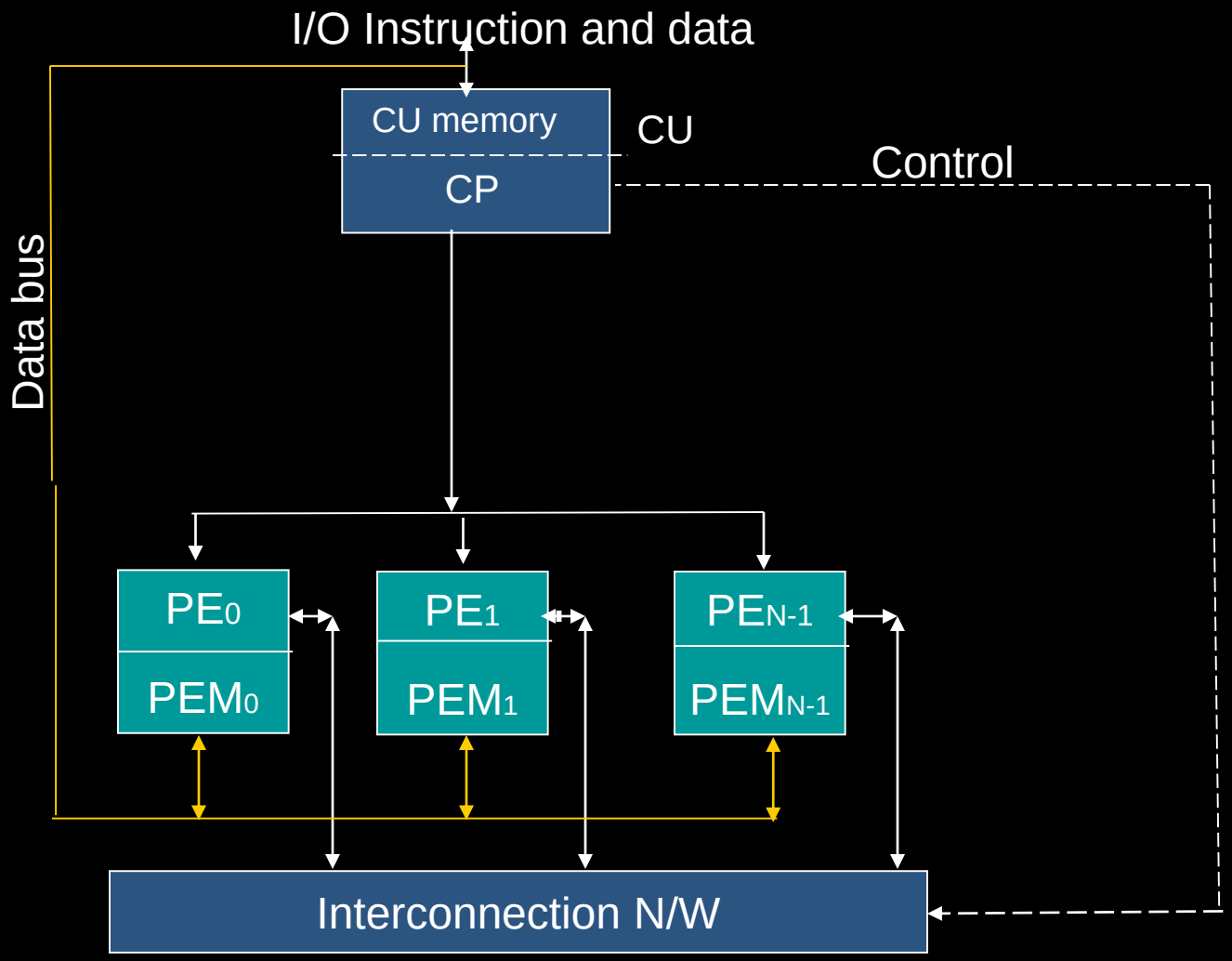
Synchronous Parallel Processing

SIMD Computer Organization

A synchronous array of parallel processors is called an array processor, which consists of PEs under the supervision of one control unit (CU).

The function of CU is to decode all the instrns and to determine whether and where the decoded instruction should be executed.

SIMD Computer Organization:

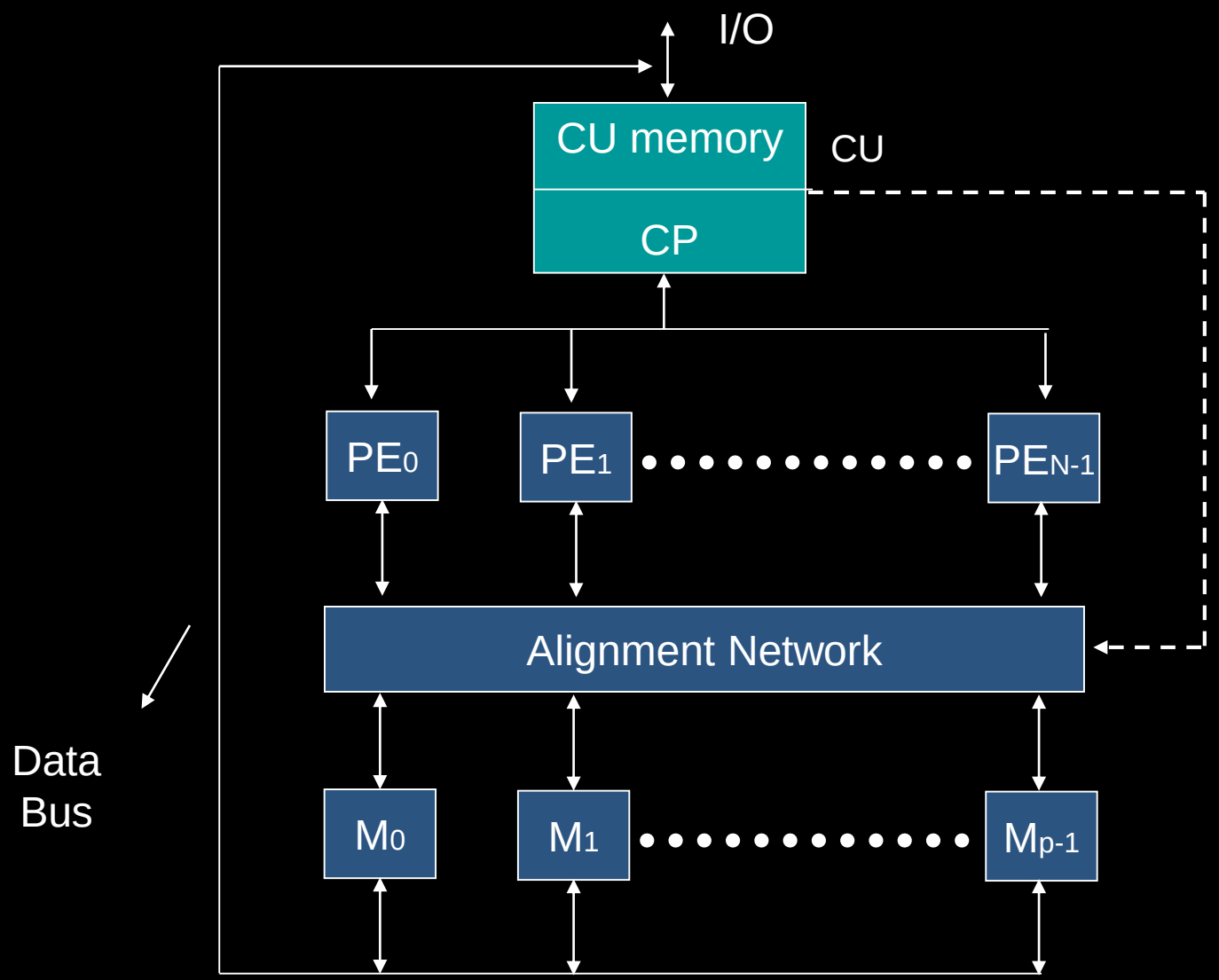


CONFIG I

Masking schemes : to control the status of each PE.

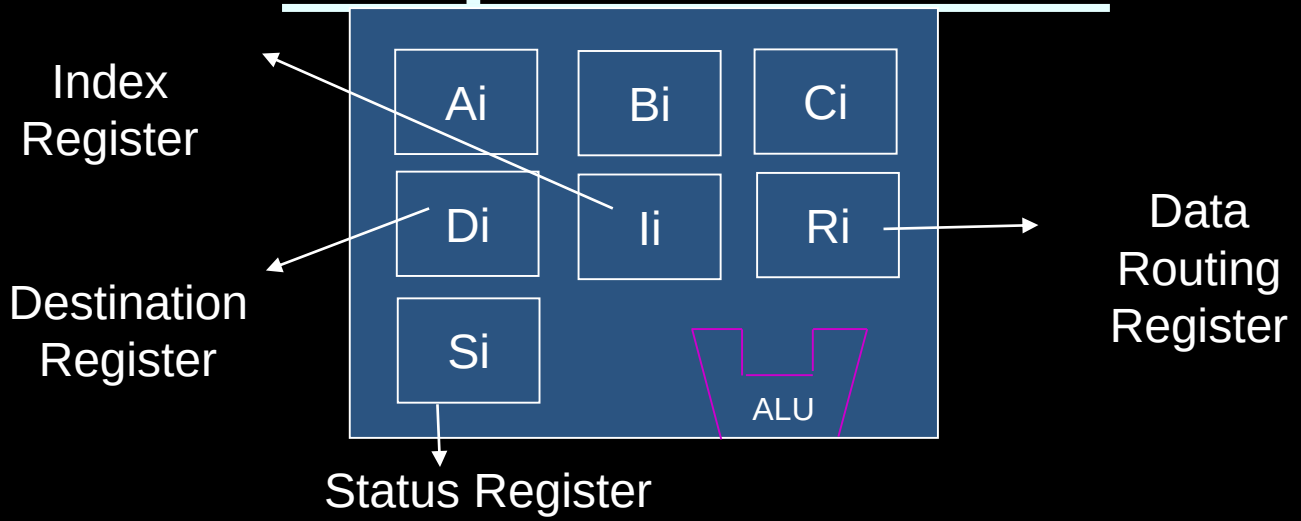
PE can be active/disabled.

Masking vector


I/O
CU memory
CU
CP
PE0
PE1
PEN-1
Alignment Network
Data
Bus
M0
M1
Mp-1

Config II:

Components in a PE:



A_i, B_i and C_i are general purpose registers.

- Only the contents of Ri are transferred to other PE's during data transfer.
- If $N = 2^m$ (m = no. of bits required to identify a PE) PE's are there, then D_i will hold "m" bit address of the destined PE.
- Each PE_i is either active or inactive during instruction cycle:

$S_i = 1$ "Active" $S_i = 0$ "Inactive"

Necessity of Data Routing:

- Consider an Array:

$$A = (A_0, A_1, \ldots\ldots\ldots, A_{n-1})$$

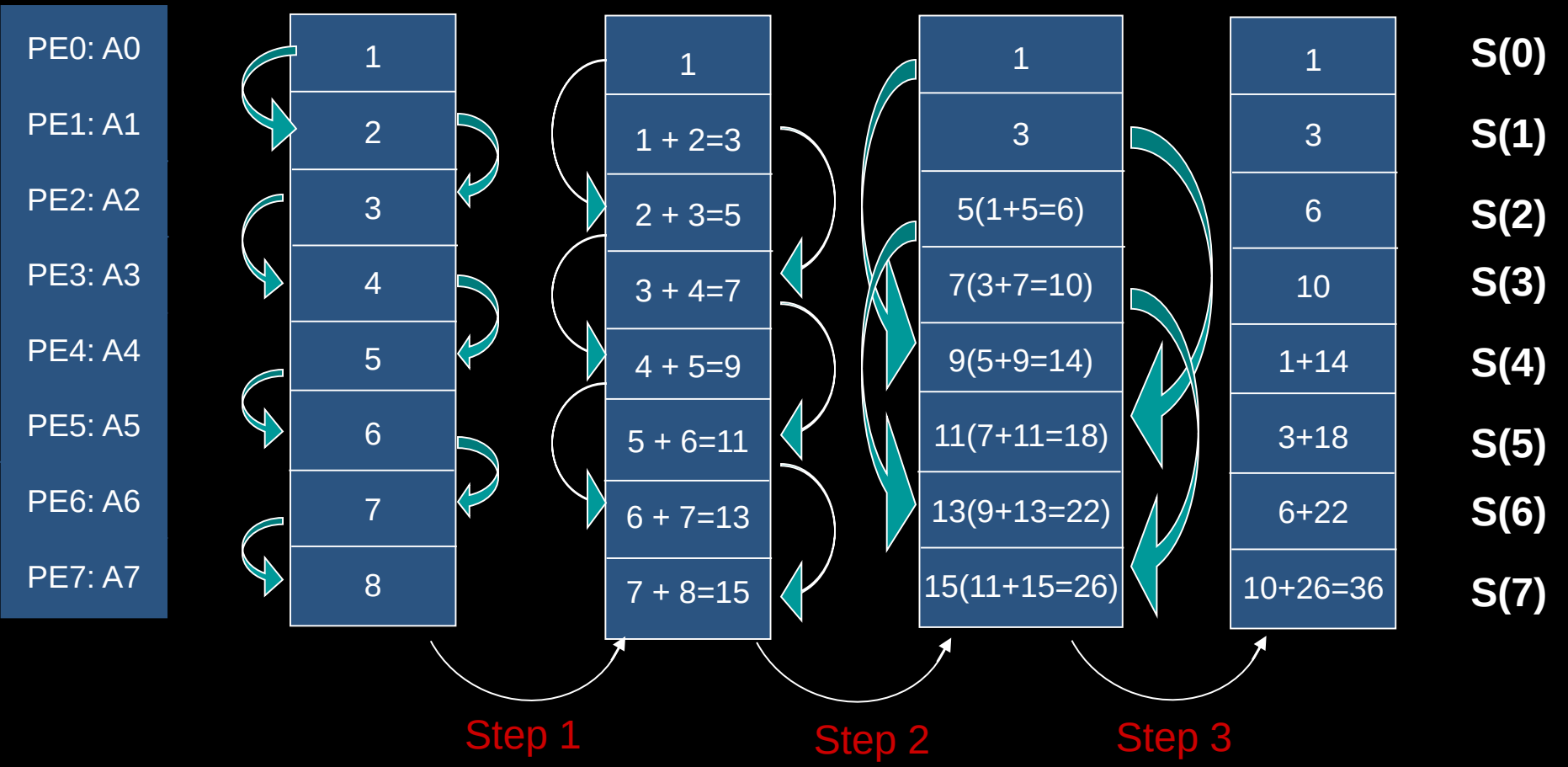
Now, for computing:

$$\mathbf{S(n) = \sum_{i=0}^{n-1} Ai}$$

In SISD, the same thing would take 8 steps/loops by formula:

sum = sum + A[i]

SIMD:


PE0: A0
PE1: A1
PE2: A2
PE3: A3
PE4: A4
PE5: A5
PE6: A6
PE7: A7
1
2
3
4
5
6
7
8
1
1 + 2=3
2 + 3=5
3 + 4=7
4 + 5=9
5 + 6=11
6 + 7=13
7 + 8=15
1
3
5(1+5=6)
7(3+7=10)
9(5+9=14)
11(7+11=18)
13(9+13=22)
15(11+15=26)
1
3
6
10
1+14
3+18
6+22
10+26=36
S(0)
S(1)
S(2)
S(3)
S(4)
S(5)
S(6)
S(7)
Step 1
Step 2
Step 3

Algorithm:

Step # 1: Ai would transfer data in Ri i = 0-6

$A_i \longrightarrow R_i$

$R_i \longrightarrow R_{i+1}$ i = 0-6

$A_i + R_i \longrightarrow A_i$ i = 1-7

Step # 2:

$A_i \longrightarrow R_i$ i = 0-5

$R_i \longrightarrow R_{i+2}$ i = 0-5

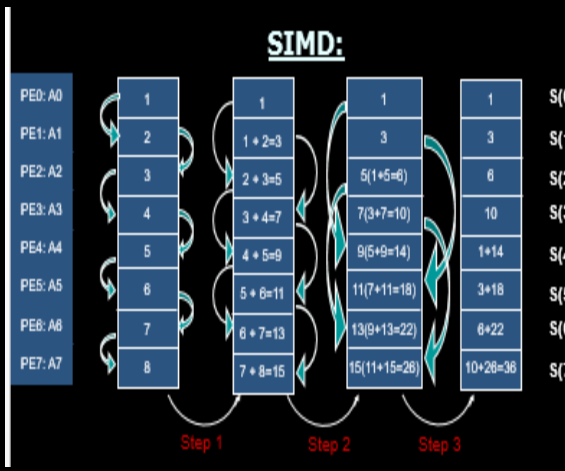
$A_i + R_i \longrightarrow A_i$ i = 2-7

Step # 3:

$A_i \longrightarrow R_i$ i = 0-3

$R_i \longrightarrow R_{i+4}$ i = 0-3

$A_i + R_i \longrightarrow A_i$ i = 4-7



Masking Scheme:

During Data Routing:

Step # 1: PE_7 is disabled.

Step # 2: PE6 and PE7 are disabled.

Step # 3: PE4 – PE7 are disabled.

During Addition:

Step # 1: PE0 is not involved.

Step # 2: PE0, PE1 are not involved.

Step # 3: PE0-PE3 are not involved.

SIMD INTERCONNECTION NETWORK

Interconnection networks are needed to route data—

- from processors to memories (concurrent access to a shared memory structure), or
- from one PE (processor + memory) to another (to provide a message-passing facility).

Measures of interconnection performance:
(Metrics for Static Networks)

Node degree (d): Number of edges (links or channels) incident on a node.

Node degree tells no. of I/O ports associated with a node,

and should ideally be small and constant.

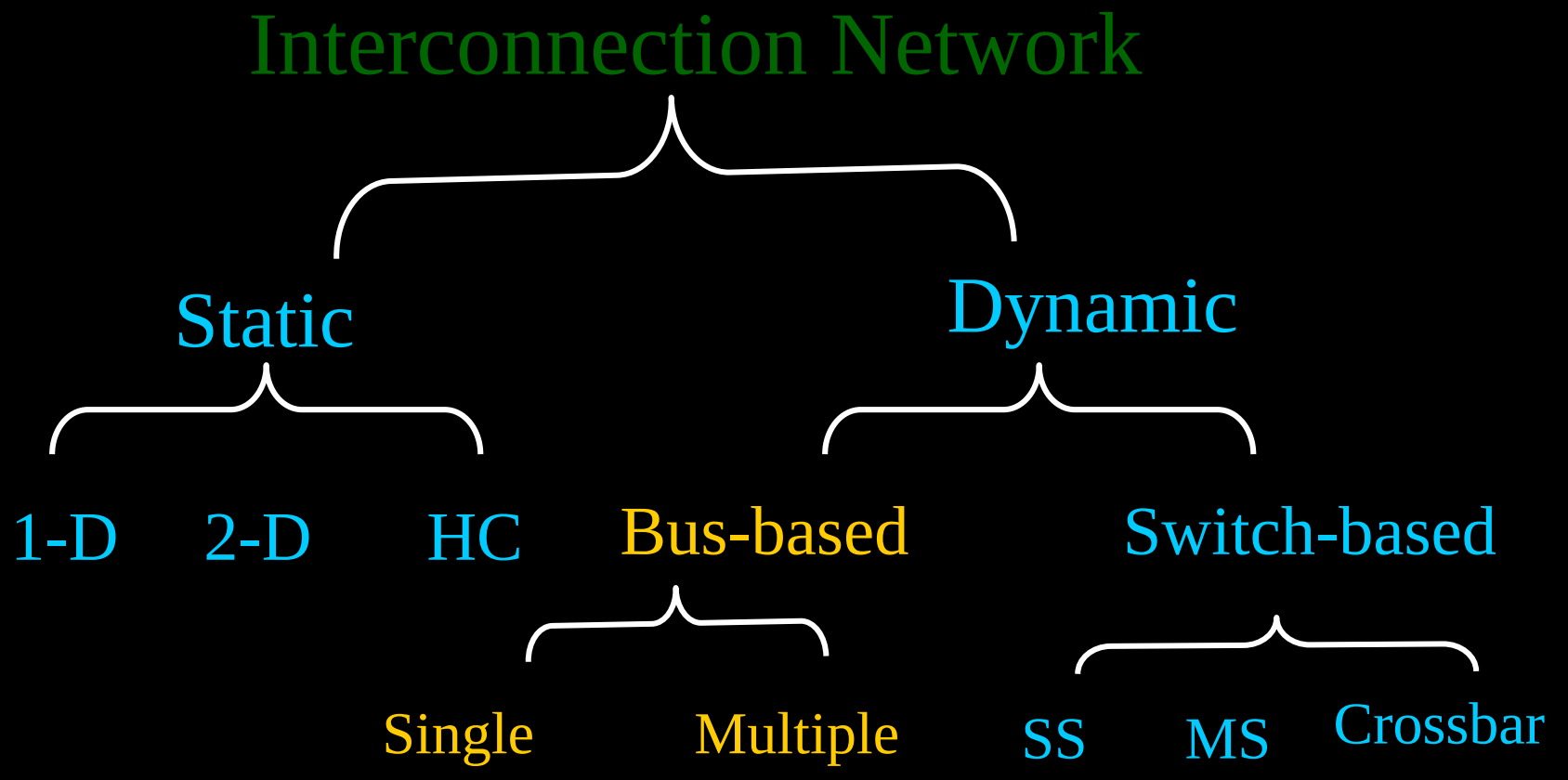
•*Diameter (D):* of a n/w is the max shortest path b/n any two nodes.

The *D* of the n/w indicates the max number of distinct hops b/n any two nodes.

Measured by the no. of links traversed; this should be as small as possible (from a communication point of view).

*Channel bisection width (b): M*inimum number of edges cut to split a network into two parts each having the same number of nodes.

Interconnection Network Taxonomy



Static Networks – Linear Array

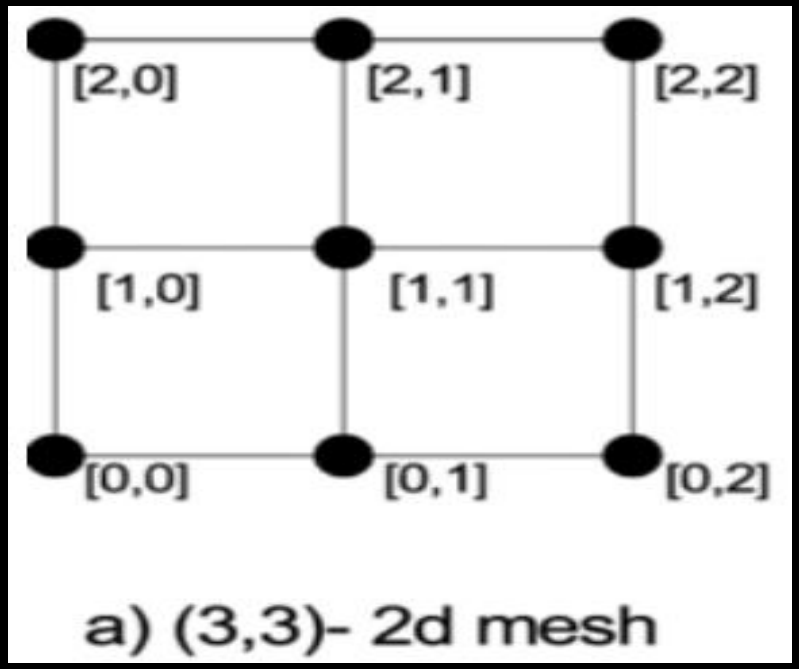
- n nodes connected by n-1 links; segments between different pairs of nodes can be used in parallel.
- Internal nodes have degree 2; end nodes have degree 1.
- Diameter = n-1
- Bisection width = 1

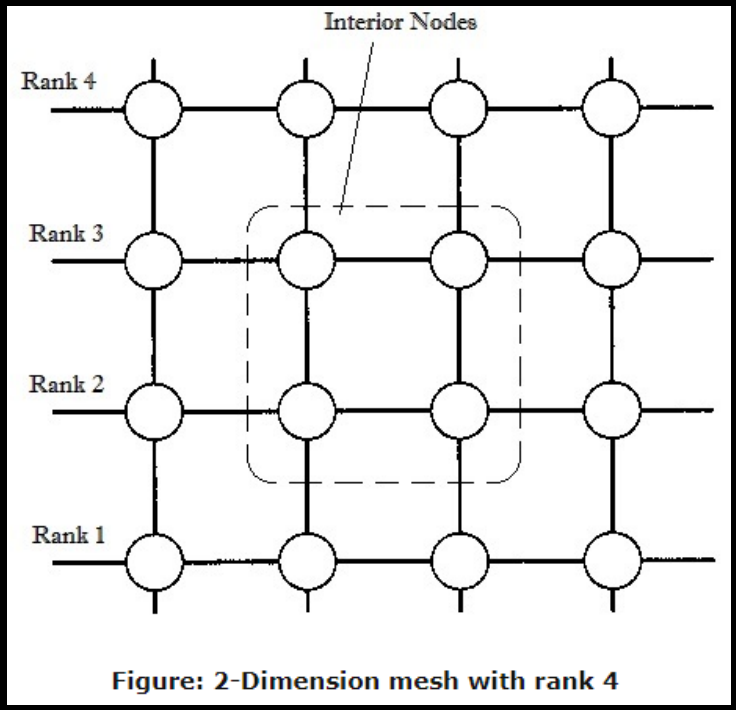
SDL:

- Static Networks: Ring, Chordal ring, Star, Completely connected networks

Static Interconnection Networks (2D)

- A *n-dimensional mesh* is an extension of the linear array [ring]. Degree: 2-4

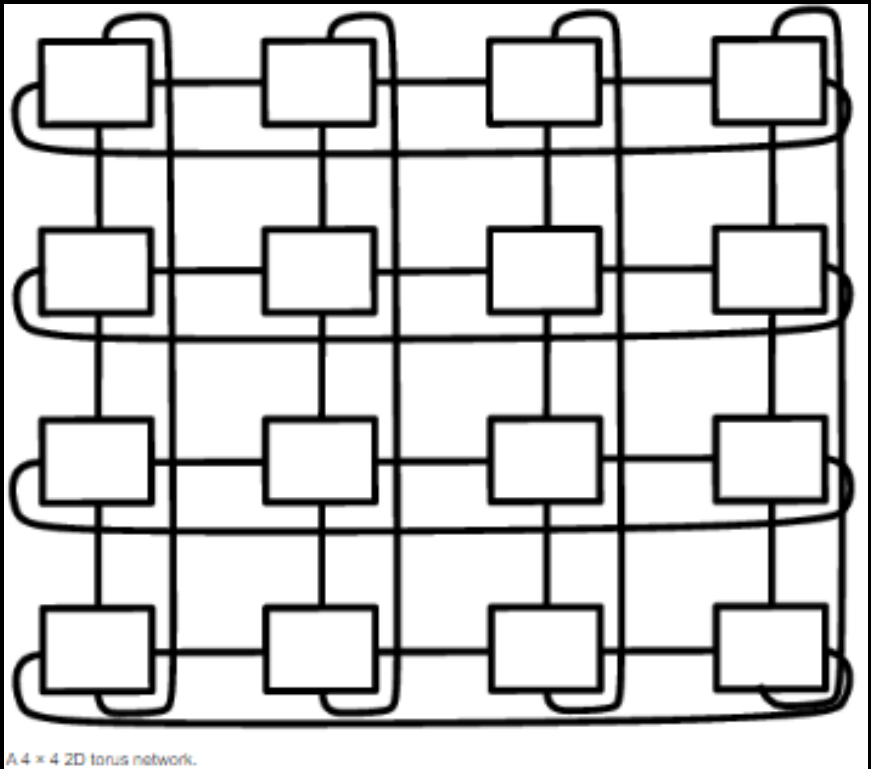




A *k*-dimensional mesh with $N = n^k$ (n is the rank mesh) nodes has an interior node degree of 2k and the network diameter is $k(n-1)$.

Static Interconnection Networks (2D)

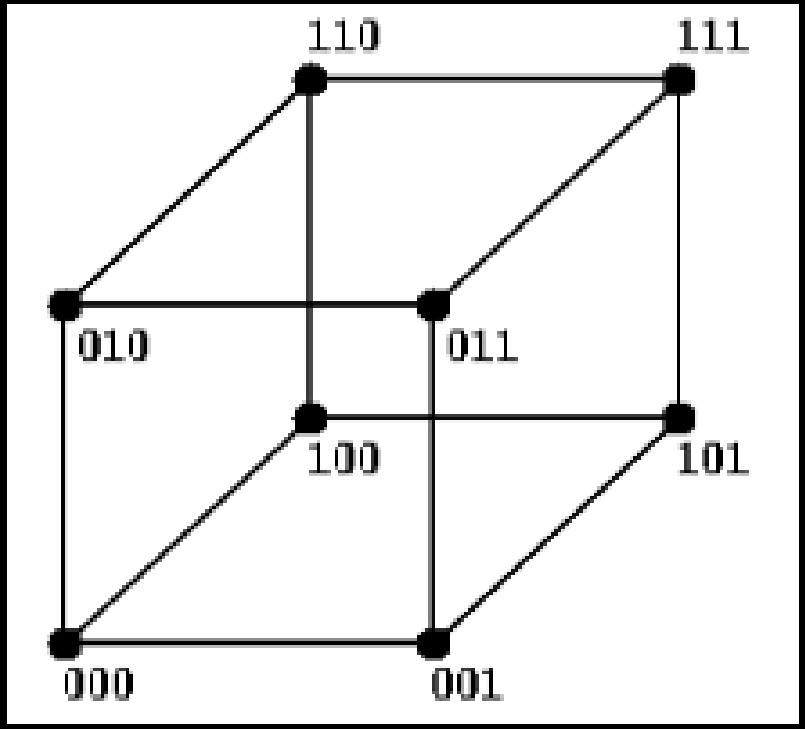
- A *n-dimensional mesh* [*torus* or *wraparound mesh*] is an extension of the linear array [ring]. Degree: 4



A 4 × 4 2D torus network.

Static Interconnection Networks 3D: Hypercubes

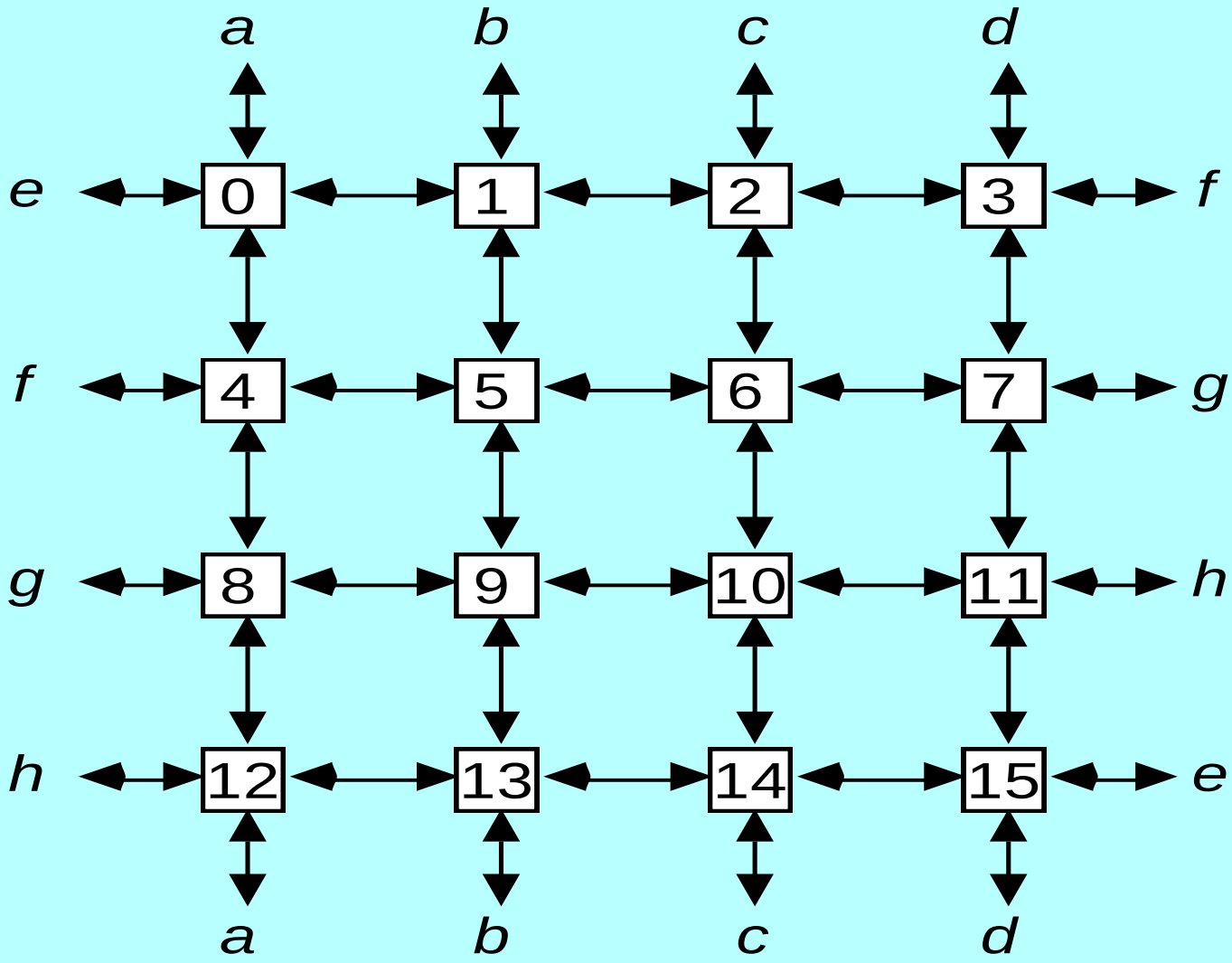
An hypercube is a multi-dimensional mesh with exactly two processors in each dimension. A *n*-cube consists of $N = 2^n$ nodes. The node degree of *n* cube equals *n* and so does the network diameter.



SDL:

- Static Networks: Ring, Chordal ring, Star, Completely connected networks
- K-ary n-cube network
- 3-cube connected cyclic (CCC) network
- 4-cube, 5-cube network

Design of
Mesh interconnection network (ILLIAC IV N/W)



In the Illiac IV, each processor i was connected to processors:

$\{i+1,\ i-1,\ i+4,$ and $i-4\}$ (mod 16).

Here are the routing functions:

$R_{+1}(i) \equiv (i + 1) \bmod N$

$R_{-1}(i) \equiv (i - 1) \bmod N$

$R_{+r}(i) \equiv (i + r) \bmod N$

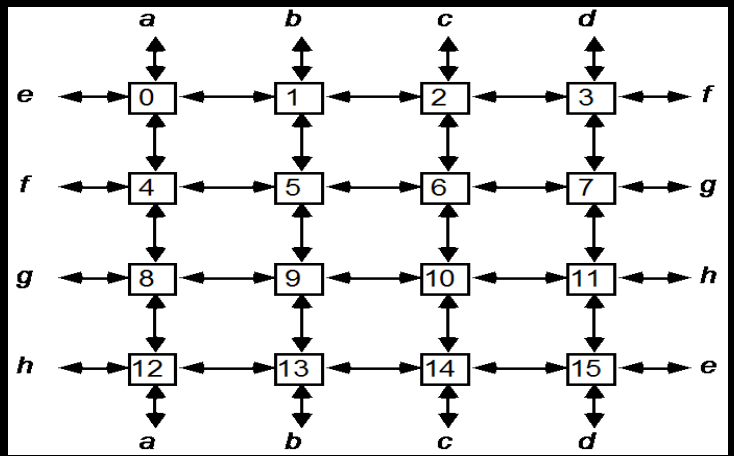
$R_{-r}(i) \equiv (i - r) \bmod N$ where $r = \sqrt{N}$

Where $0 \le i \le N-1$.

The permutation cycles:

$R_{+1} \equiv (0\ \ 1\ \ 2\ \ \ldots\ \ N-1)$

$R_{-1} \equiv (N-1\ \ \ldots\ \ 2\ \ 1\ \ 0)$



R_{+4}= (0 4 8 12) (1 5 9 13) (2 6 10 14) (3 7 11 15)
R_{-4}= (15 11 7 3) (14 10 6 2) (13 9 5 1) (12 8 4 0)

The diameter of an Illiac-IV mesh is $\sqrt{N} - 1$. For example, in a 16-node mesh structure, it takes a maximum of 3 steps.

Chordal Ring N/W is ILLIAC-IV n/w. Also called as partially connected n/w. (Diag.)

Barrel shifting network:

A barrel shifter is sometimes called a *plus-minus-* 2^i network.

Routing functions:

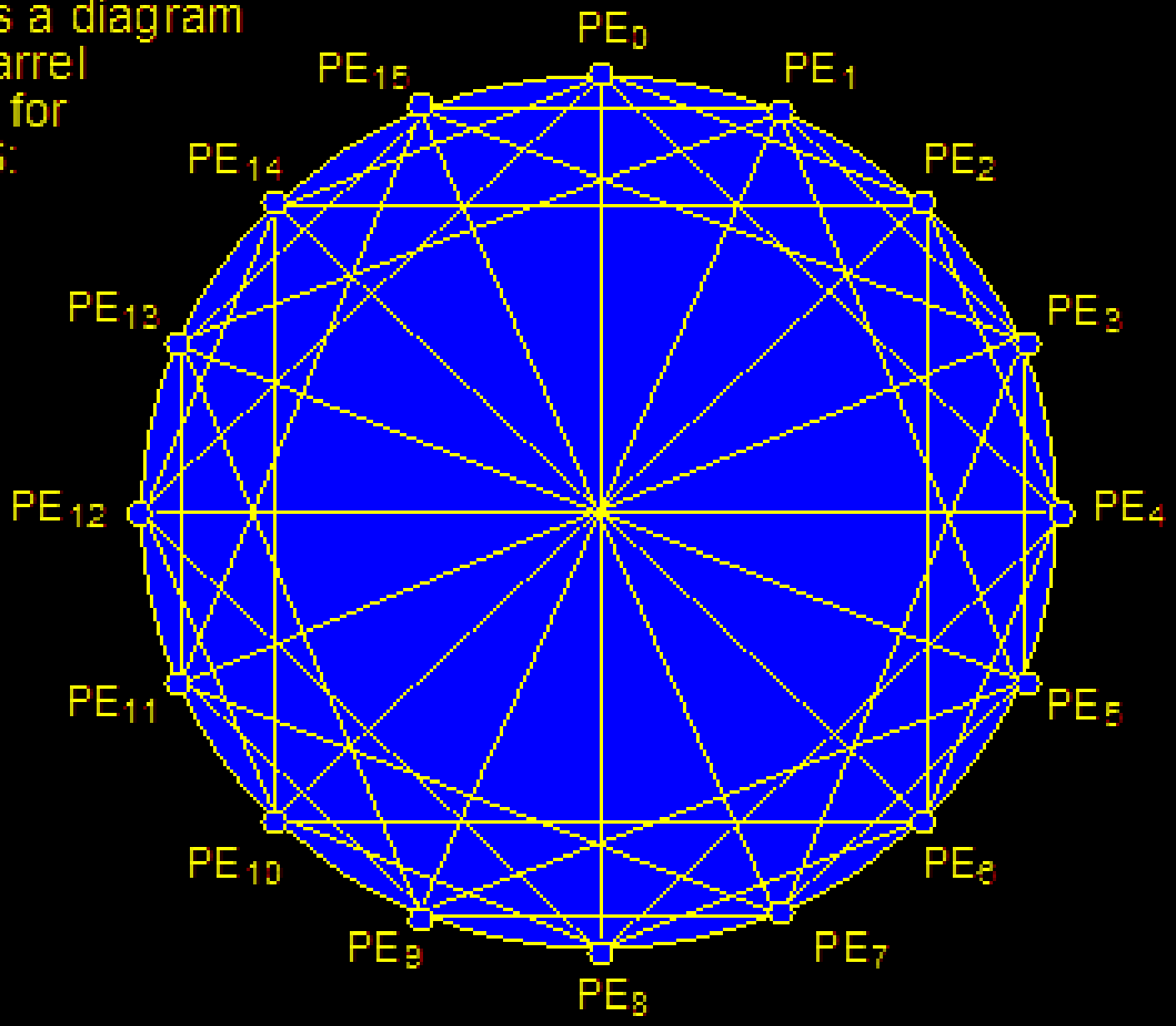
$$B_{+i}(j) = (j + 2^i) \bmod N$$
$$B_{-i}(j) = (j - 2^i) \bmod N$$

where $0 \le j \le N - 1$ and $0 \le i < \log_2 N$.

B_{+0}, B_{-0}, B_{+1}, B_{-1}, B_{+2}, B_{-2}, B_{+3} and B_{-3}

In general, the diameter of a barrel shifter is $(\log_2 M)/2$.

Here is a diagram of a barrel shifter for $N = 16$:



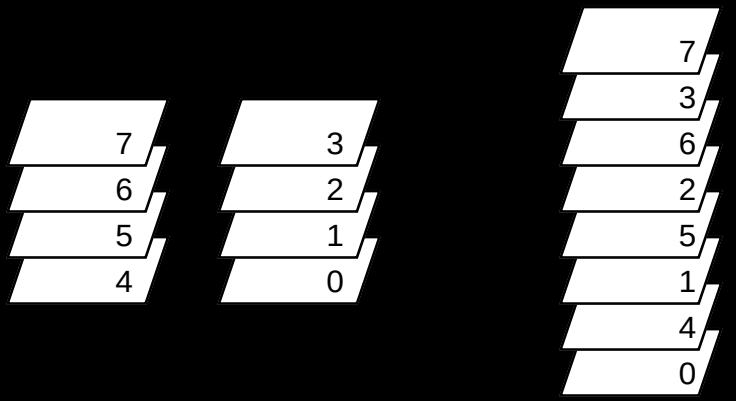
SDL:

- Static Networks: Ring, Chordal ring, Star, Completely connected networks
- K-ary n-cube network
- 3-cube connected cyclic (CCC) network
- 4-cube, 5-cube network
- Barrel Shifting network
- Comparison of Mesh with Barrel shifting

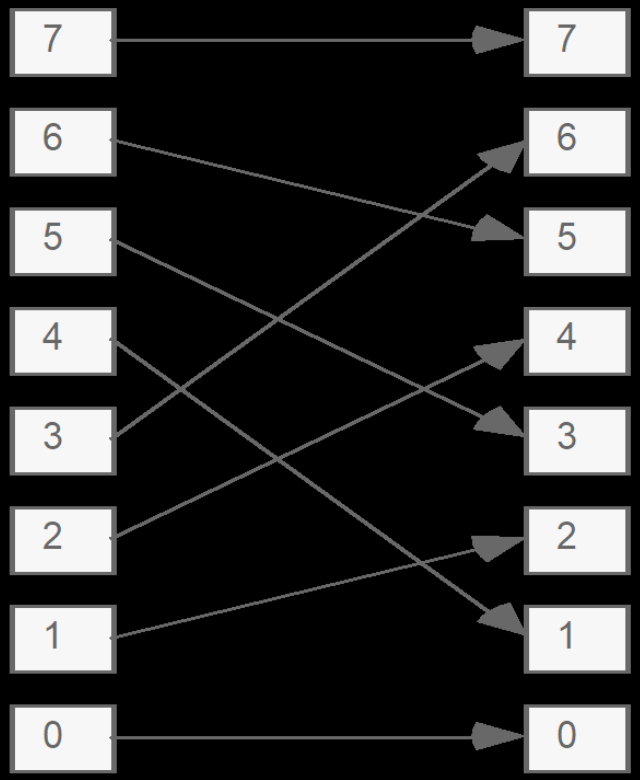
Perfect-shuffle interconnection:

This interconnection network is defined by the routing function

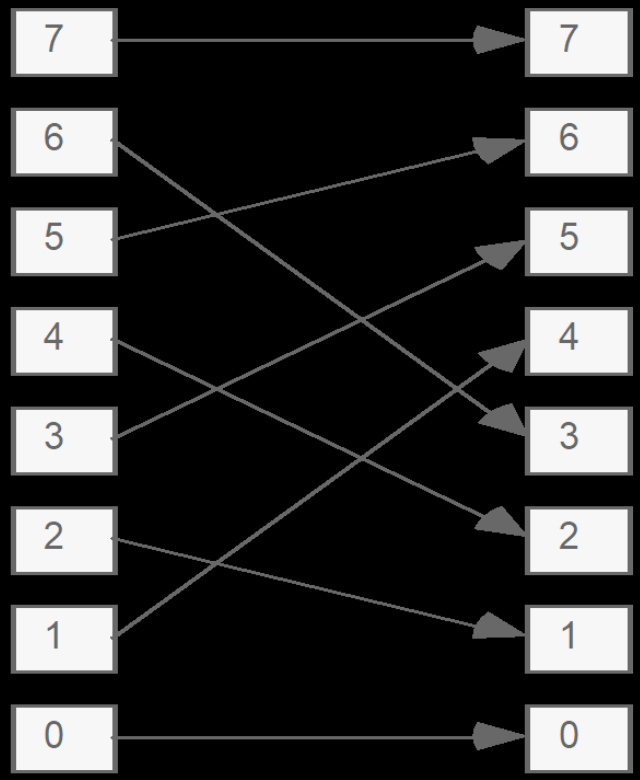
$$S\,(a_{n-1}\ \ldots\ a_1 a_0)_2 = (a_{n-2}\ \ldots\ a_1 a_0\, a_{n-1})_2$$



It describes what happens when we divide a card deck of, e.g., 8 cards into two halves and shuffle them "perfectly."



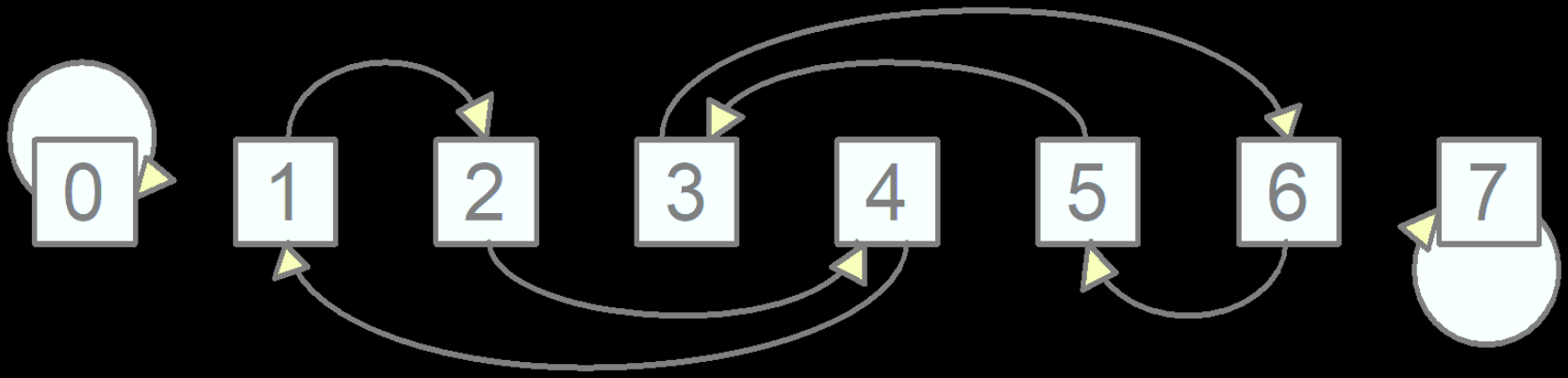
Perfect Shuffle



Inverse Shuffle

Hence permutation can be expressed as
S=(0) (1 2 4) (3 6 5) (7)
We can draw the processor interconnections required to obtain this transformation.

a shuffle network is not a complete interconnection network. This can be seen by looking at what happens as data is *recirculated* through the network.



An *exchange* permutation can be added to a shuffle network to make it into a complete interconnection structure.

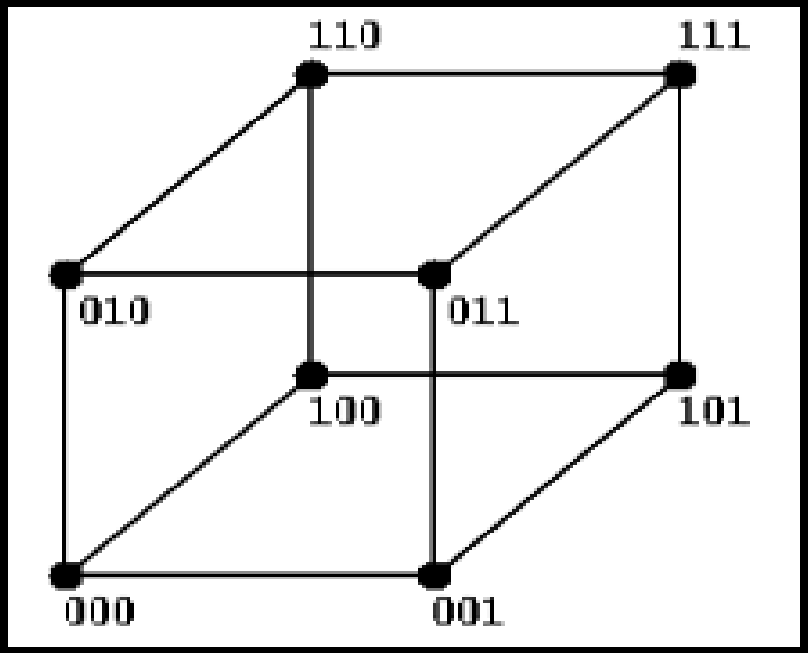
$$E\,(a_{n-1}\ \ldots\ a_1\,a_0)_2 = a_{n-1}\ \ldots\ a_1\,\bar{a}_0$$

Hence permutation can be expressed as

E= (0 1) (2 3) (4 5) (6 7)

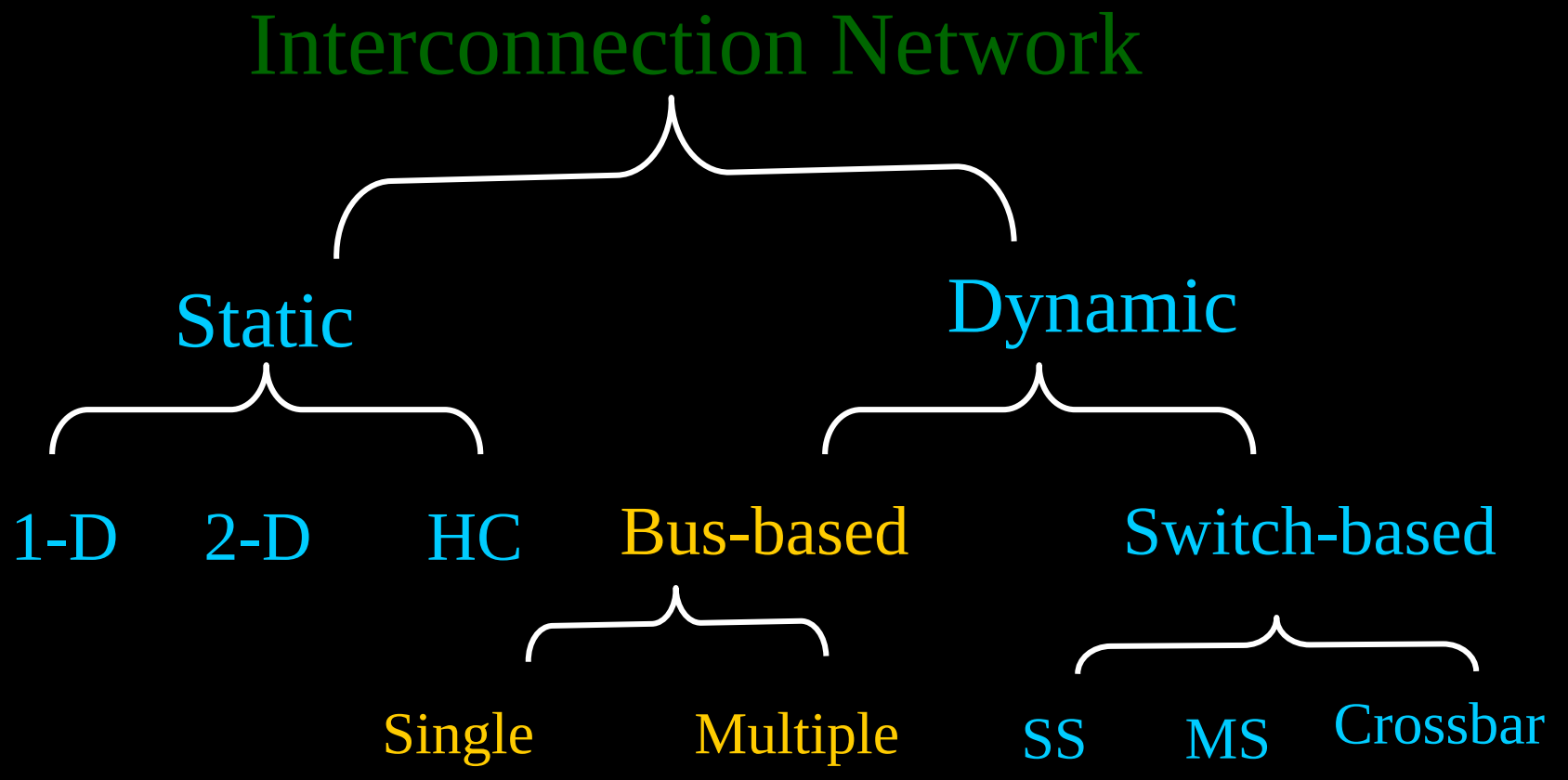
with a shuffle-exchange network, arbitrary cyclic shifts of an N-element array can be performed in log N steps.
Here is a diagram of a multistage omega network for $N = 8$.

Hypercube



A hypercube is a generalized cube. In a hypercube, there are 2^n nodes, for some n. Each node is connected to all other nodes whose numbers differ from it in only one bit position.

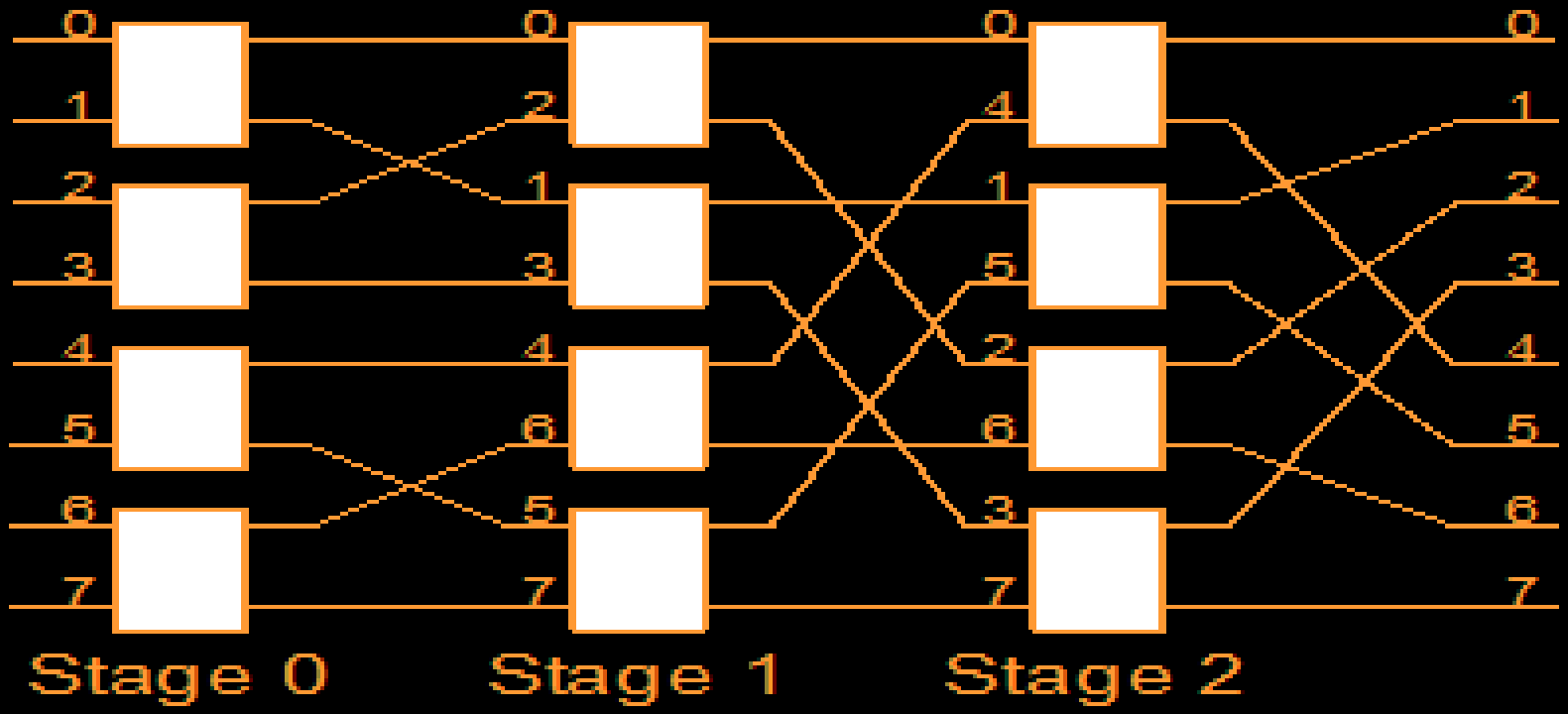
Interconnection Network Taxonomy



The routing functions have this form:

$$C_i(a_{n-1} \ldots a_{i+1}\, a_i\ \ a_{i-1} \ldots a_0)_2$$
$$= \ (a_{n-1} \ldots a_{i+1}\, \bar{a}_i\ \ a_{i-1} \ldots a_0)_2$$

For a multistage cube network, we can diagram the paths from one cell to another like this:



Multistage Interconnection Network

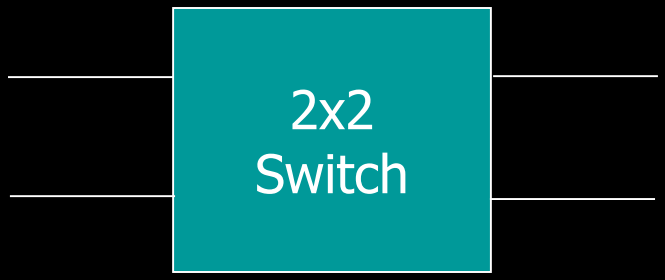
Switch Modules

- A x B switch module
- A inputs and B outputs
- In practice, A = B = power of 2
- Each input is connected to one or more outputs (conflicts must be avoided)
- One-to-one (permutation) and one-to-many are allowed

Multistage Interconnection Network
Switch Modules

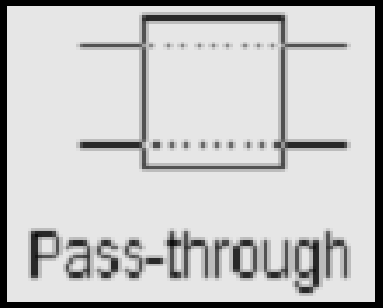
- A 2×2 switch can be configured for
  - Straight-through
  - Crossover
  - Upper broadcast (upper input to both outputs)
  - Lower broadcast (lower input to both outputs)

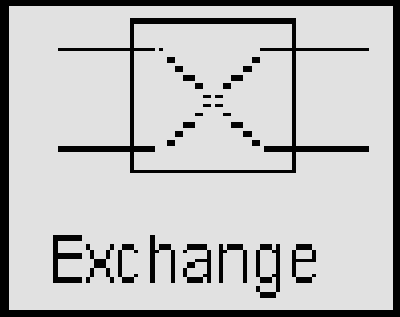
Binary Switch

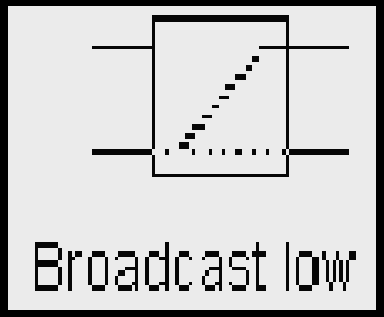


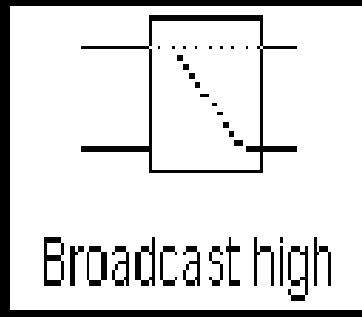
Legitimate States = 4

Permutation Connections = 2





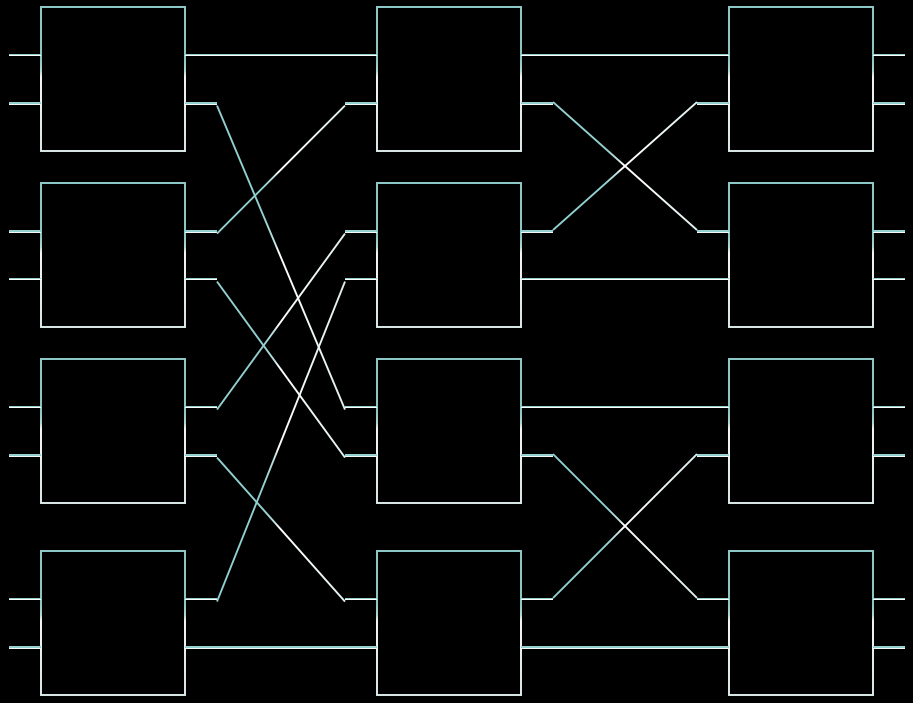




Baseline Network

- A baseline network has a simple recursive generation procedure.
- The first stage contains one N x N block and second stage two (N/2) x (N/2) subblocks.
- The construction process is recursively applied to the subblocks until the N/2 subblock is of size 2 x 2 are reached. The subblocks are straight and crossover between the two inputs and two outputs.

8×8 Baseline Network



Question: Construct a 16 x 16 Baseline network.