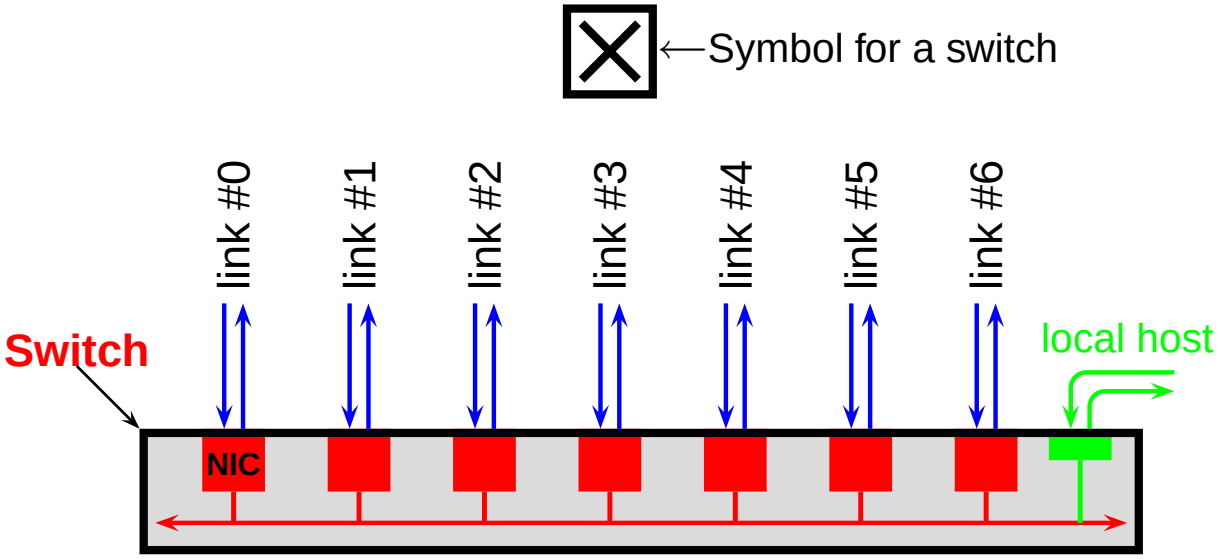
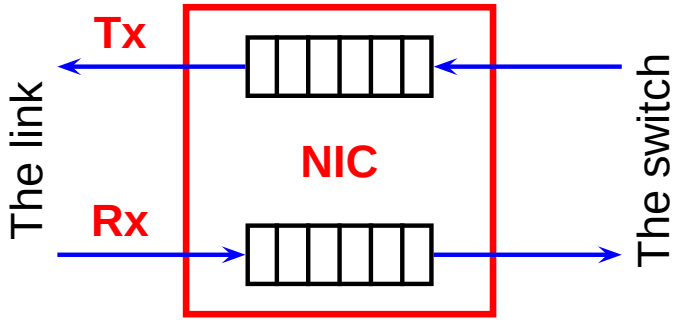
Switch

A **DLL** switch is similar to a router but it operates within the Data Link Layer (no notion of IP addresses).



Several bi–directional links (“interfaces”) are connected to a switch; each link is handled a device that resembles a NIC card in a computer.

The switch has an input and an output buffer for each link (like a router).



This makes access to a switch collision–free but introduces the possibility of frames being dropped inside a switch due to congestion (contention for output buffers or for internal interconnection).

Inside a switch

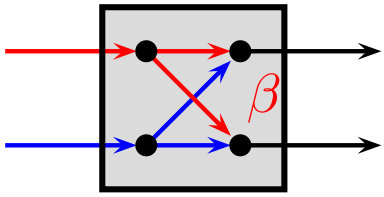
There are several ways to interconnect the NICs inside a switch. The standard options are:

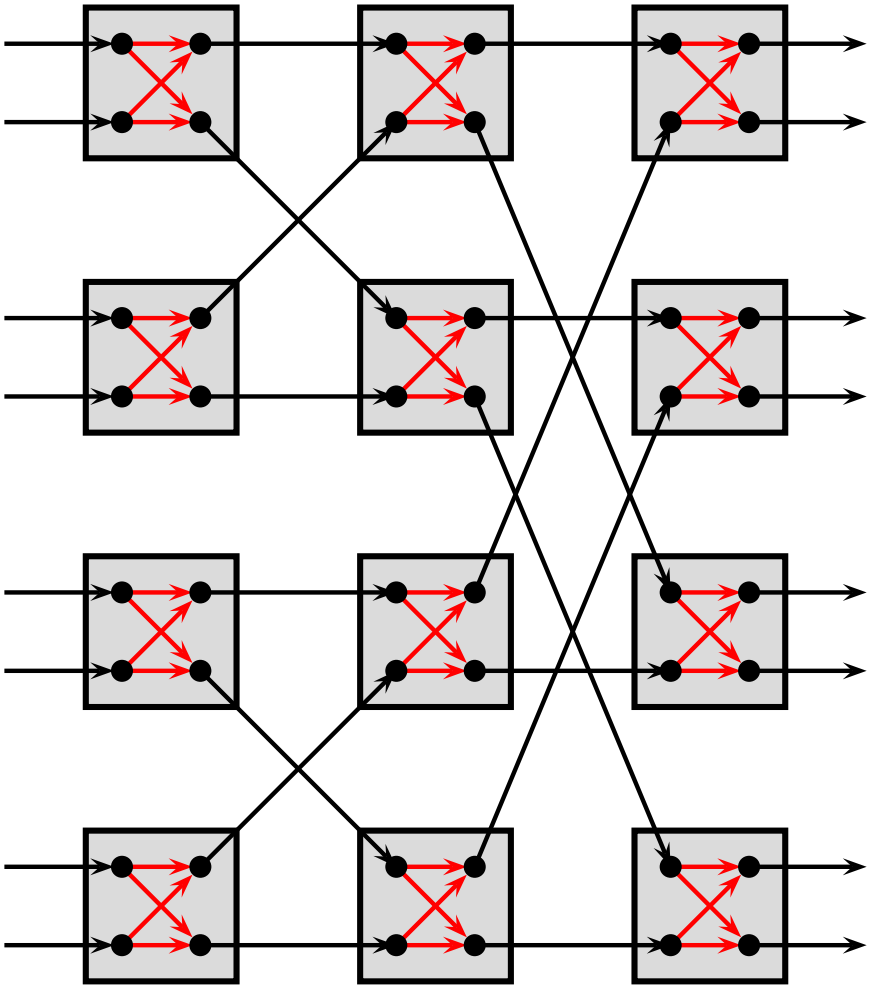
Broadcast bus: a single multiple–access bus connects all the inputs (the input buffers of the NICs) to all the outputs (output buffers of the NICs).
A MAC protocol arbtrates the use of the bus.

Banyan switch: A cascade of β elements (2×2 switches) connectes all the inputs to all the outputs. Collisions may occur as a result of contention for one output port in a β element.

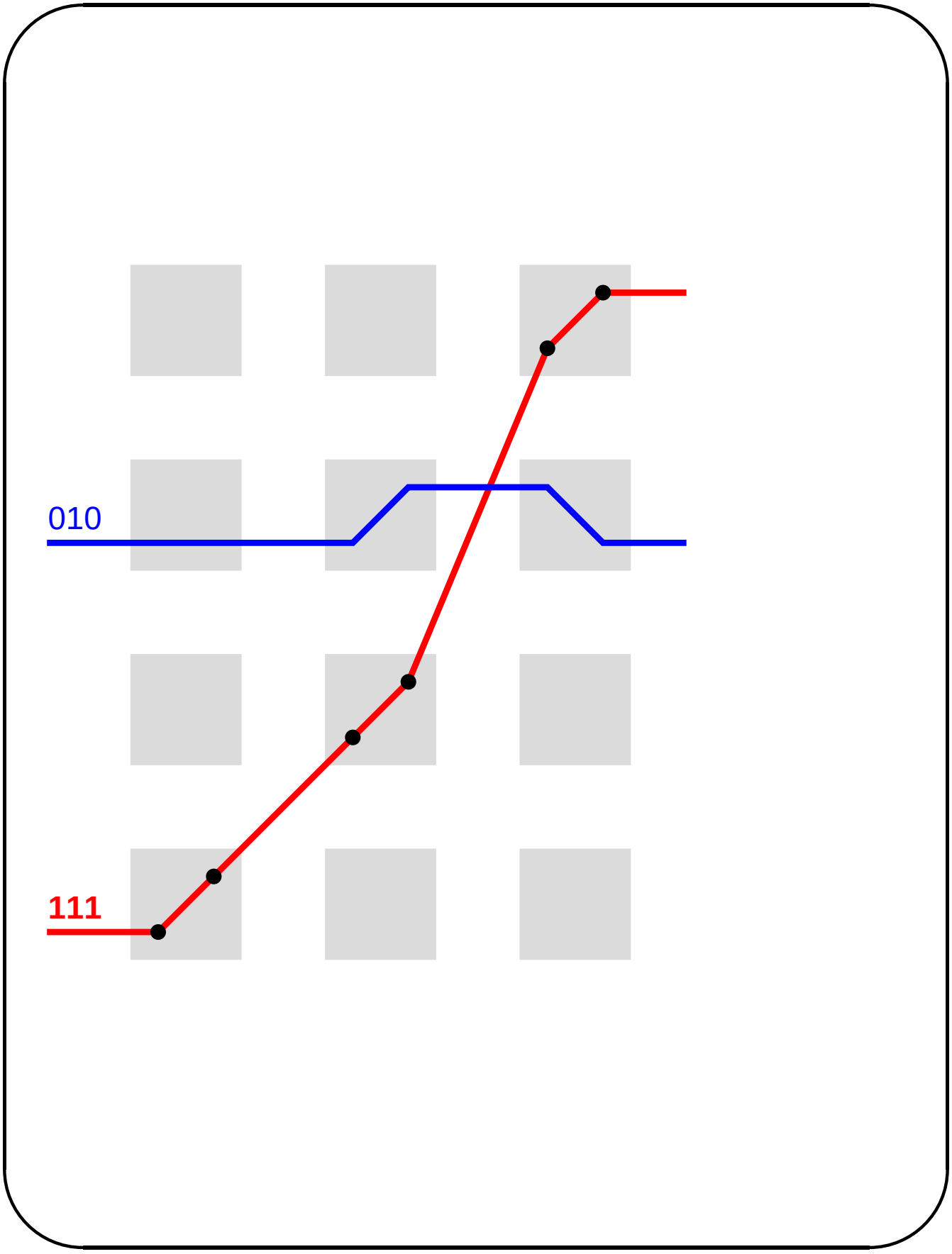
Crossbar switch: Each input buffer has its own output bus line and each output buffer has its own input bus. Each input bus has a tap connector to every output bus and receives from it all the traffic sent to it. There are no collisions but an input bus may be busy when a frame sent to it appears at the connector (the frame is lost).

Each β element is a simple 2×2 switch. A collison occurs when both inputs need the same output line.

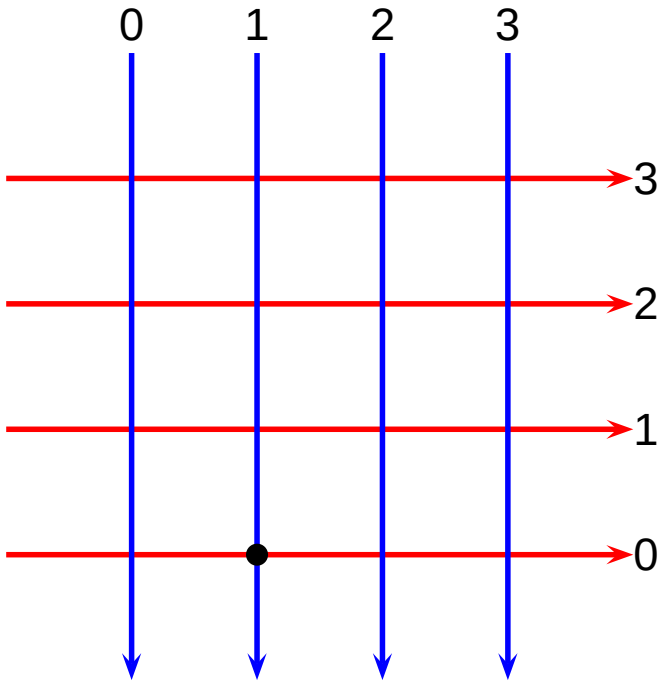




shuffle: $\pm 2^0$ $\pm 2^1$ $\pm 2^2$

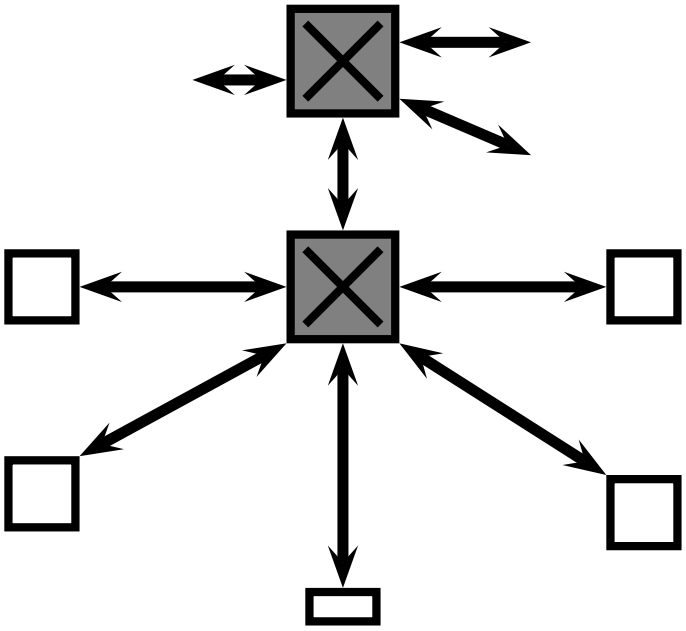

010
111

Crossbar switch



If (1) sends a frame to (0), input bus 0 will pick the frame from output bus 1.

Switches to switches



Switches can be connected to other switches, creating a WAN with Data Link Layer “routers” (switches). The main difference (besides the obvious absence of IP) is that switches do not know whether a link leads to a switch or to a host. Therefore they do not exchange information. Likewise, switches do not choose the path for a frame–it is automatically given by their tables.

Switched Ethernet

A switch performs several functions:

- Maintains and updates a **switch table** which contains all the known nodes.
- Drops unwanted frames.
- Forwards frames to a single link if the destination is known.
- Broadcasts to all links if the destination is unknown.

Switch table

A switch is not a router: it assumes that the destination is at the other end of a link. It maintains a table of MAC addresses that are assumed to be attached to each link (possibly a broadcast link).

An entry in the switch table is made of 3 parts:

MAC address	Interface	Last used
A1-B2-C3-D4-E5-F6	4	10:35

The “interface” is the index of the link where the node with the given address is.[a]

The “last used” is the time when this entry was last updated. It is used for table management.

[a] The node may be further away–past yet another switch–but this switch does not know it.

Table management

The switch starts with an empty table. Whenever the switch receives a frame, it looks in the table for the **source address** of the frame. If the address is not there, a new entry is created.

This way the switch “knows” what addresses live up teach link; this information is used in forwarding frames in the future.

Eventually, the table may become full. Then, the least recently used entry is purged. Additionally, there is a time–to–live for entries. An entry is purged if it was not used for the last TTL amount of time even if there is space in the table.

Operation of a switch

When a switch receives on interface x a frame with source address $\mathcal{A}$, it performs the following action:

- There is no entry for $\mathcal{A}$. No entry is created and the frame is forwarded to the output buffers of all the interfaces except x.
- There is an entry for $\mathcal{A}$ showing that $\mathcal{A}$ is on interface y. The entry for $\mathcal{A}$ is updated ("Last used") and the frame is placed in the output buffer of y.
- There is an entry for $\mathcal{A}$ showing that $\mathcal{A}$ is on interface x. The frame is discarded. The table is not updated.

Operation of a switch

When a switch receives on interface x a frame with destination address $\mathcal{A}$, it performs the following action:

- There is no entry for $\mathcal{A}$. An entry is created saying that $\mathcal{A}$ is reachable through interface x.
- There is an entry for $\mathcal{A}$ showing that $\mathcal{A}$ is on interface y. The entry for $\mathcal{A}$ is updated ("Interface"): x replaces y.
- There is an entry for $\mathcal{A}$ showing that $\mathcal{A}$ is on interface x. The table is updated ("Last used").