Network on Chips
NoC

Bus pros (☺) and cons (☹)

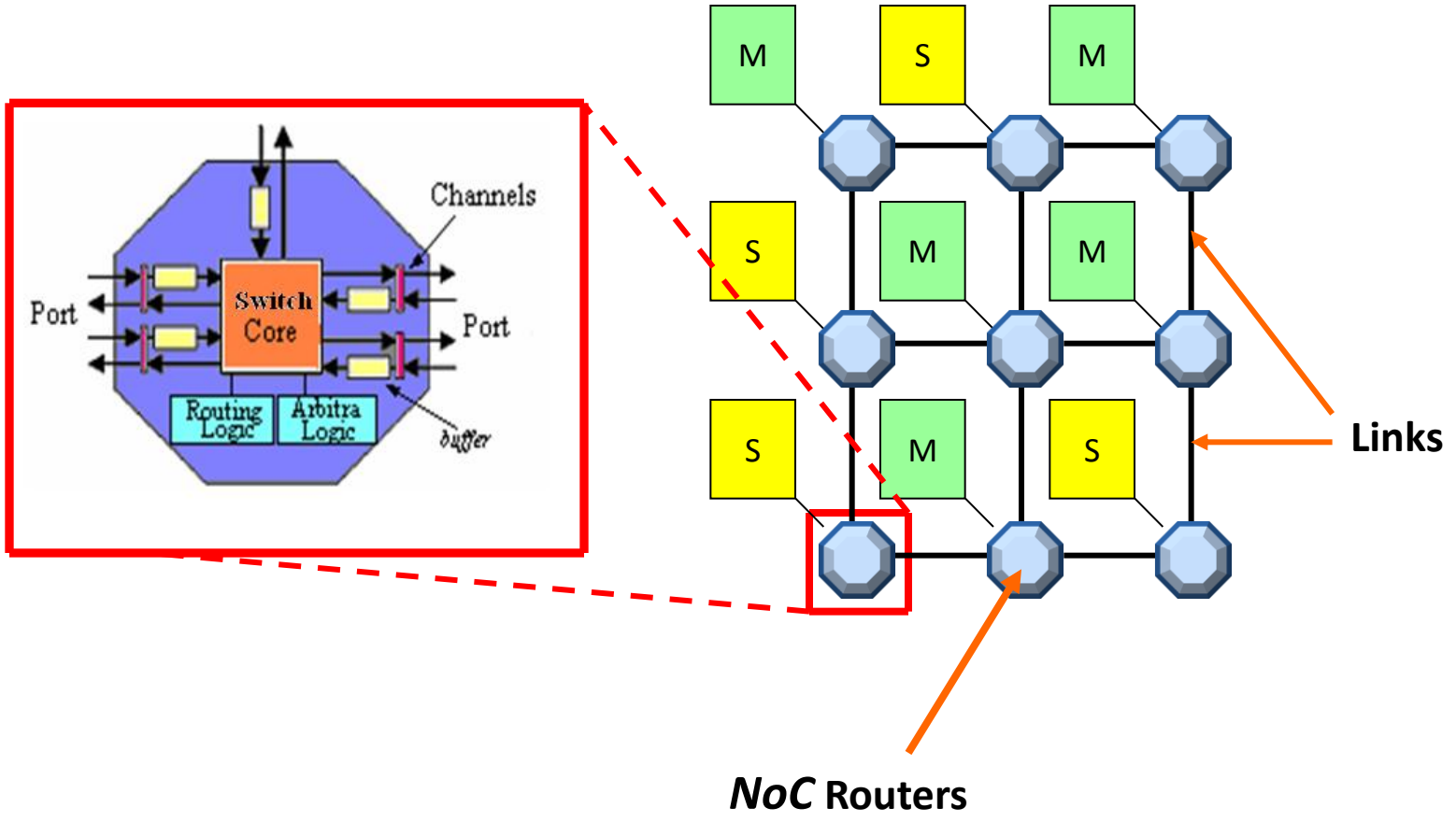
- ☺ Bus latency is zero once arbiter has granted control.
- ☺ The silicon cost of a bus is near zero.
- ☺ Any bus is almost directly compatible with most available IPs, including software running on CPUs.
- ☺ The concepts are simple and well understood.
- ☹ Every unit attached adds parasitic capacitance, therefore electrical performance degrades with growth.
- ☹ Bus timing is difficult in a deep submicron process.
- ☹ Bus arbiter delay grows with the number of masters. The arbiter is also instance-specific.
- ☹ Bandwidth is limited and shared by all units attached.

Communication Structure:

Network-on-Chip

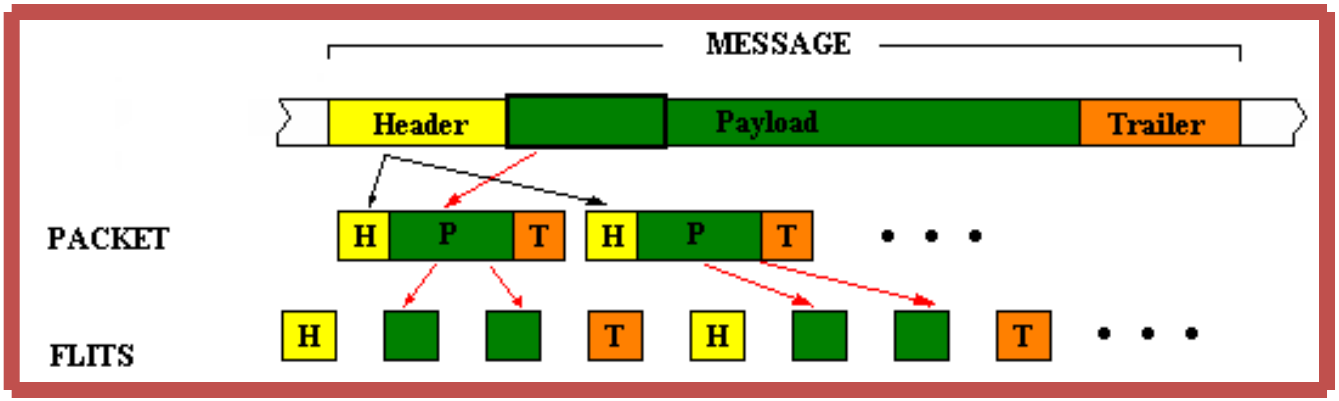
Network-on-Chip: Structural Characteristics and Configuration Parameters

NoC *(Network-on-Chip)*

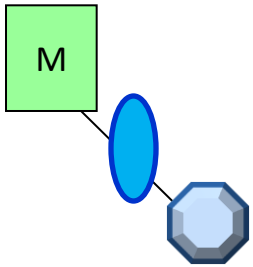


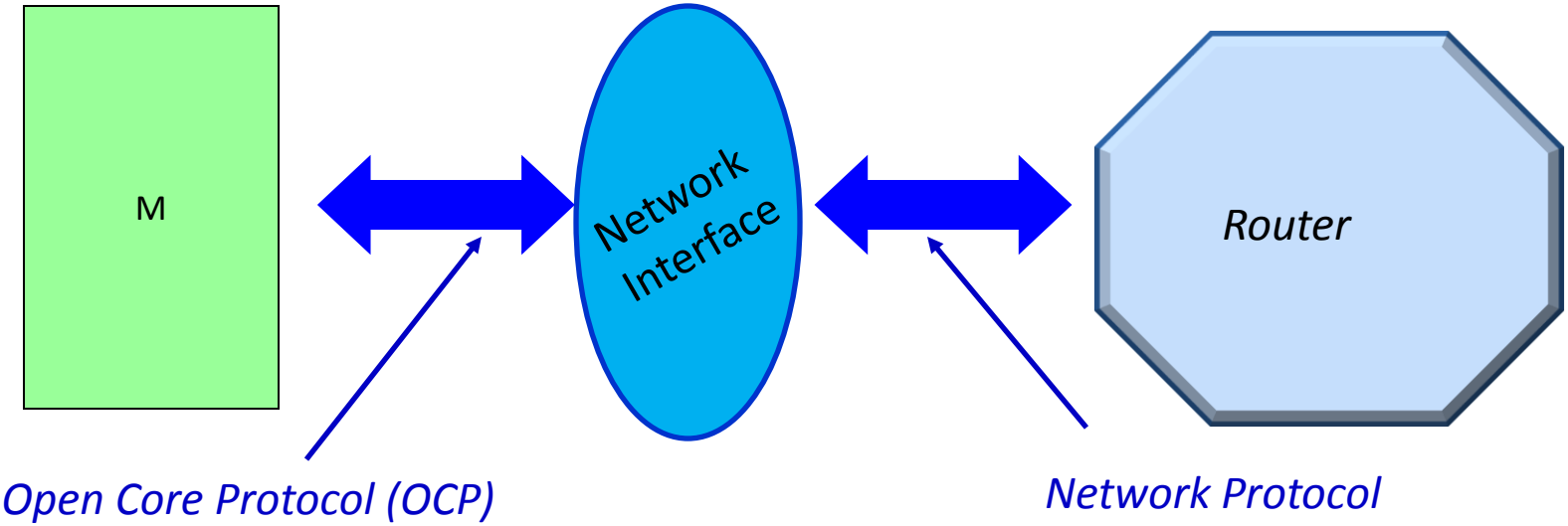
NoC *(Network-on-Chip) ??tail trailer*

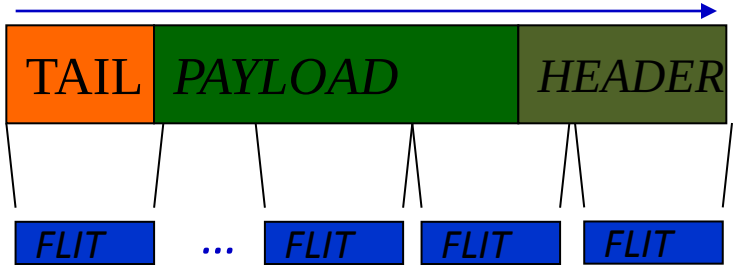
Communication:



NoC *(Network-on-Chip)*

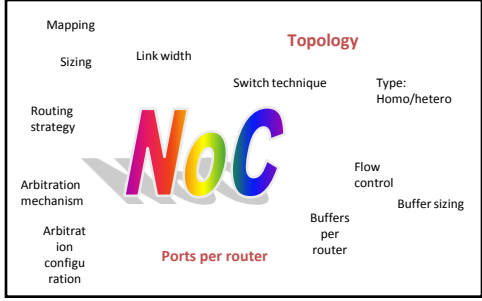




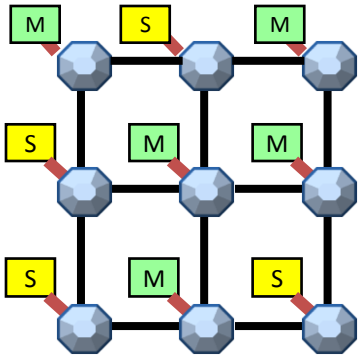


- Transmit
  - Access routing tables
  - Assemble packets
  - Split into flits
- Receive
  - Synchronize
  - Drop routing information

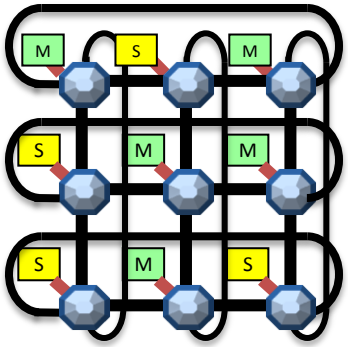
NoC parameters: **Topology** (1)



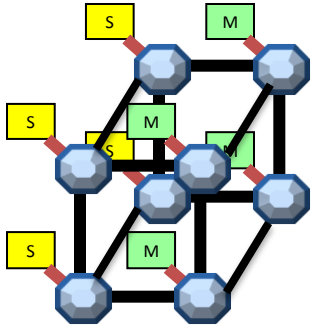
Direct NoC



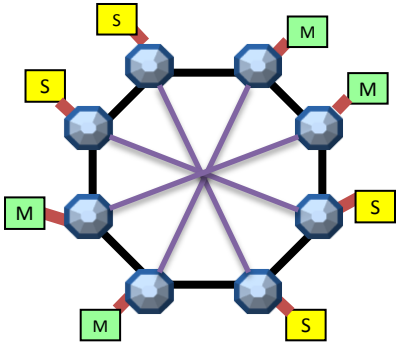
Mesh



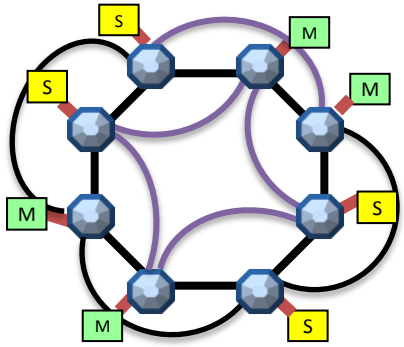
Torus



Hipercube

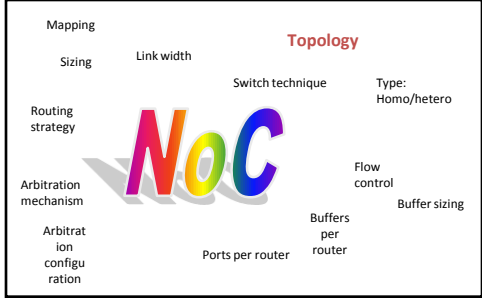


Octagon

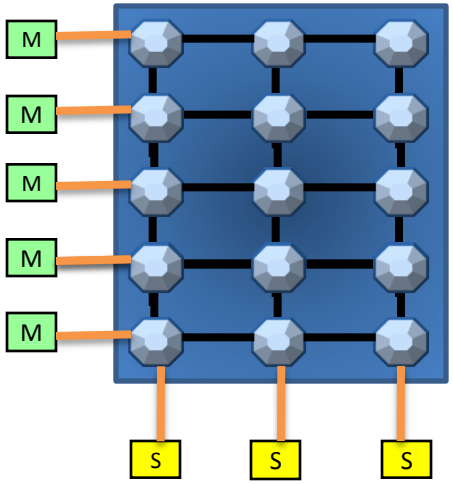


Nested Ring

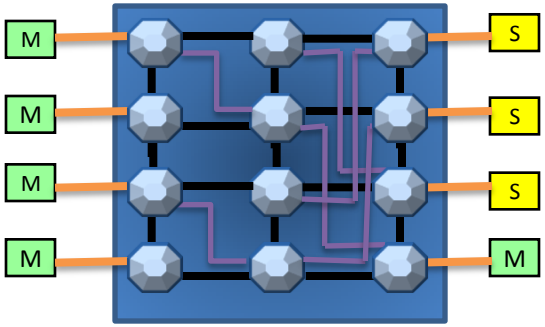
NoC parameters: **Topology** (2)



Indirect NoC

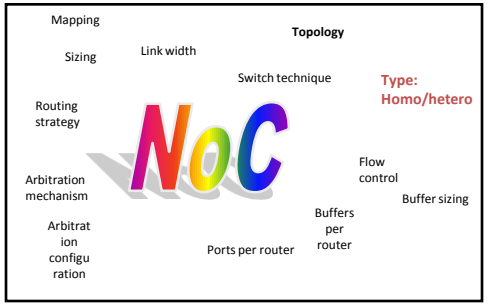


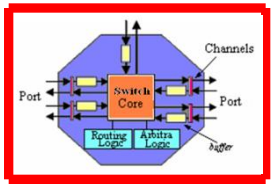
Crossbar

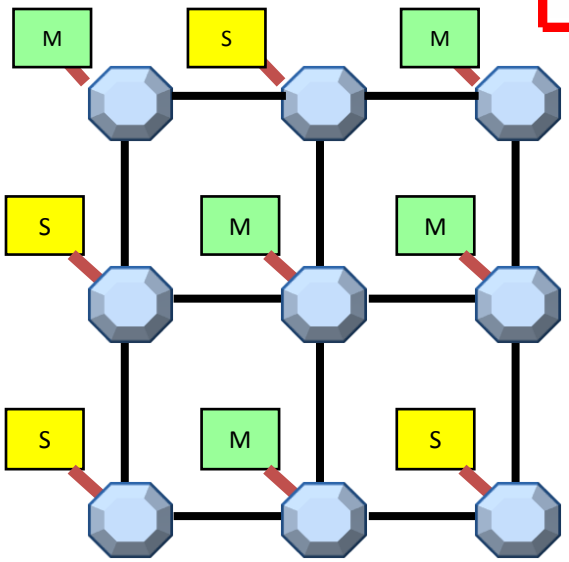


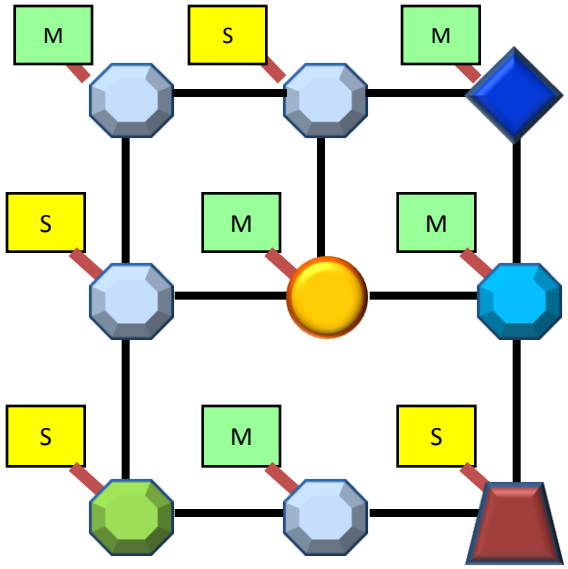
Multi-stage

NoC parameters: **Network Type**





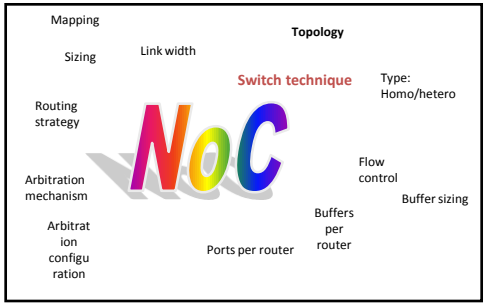


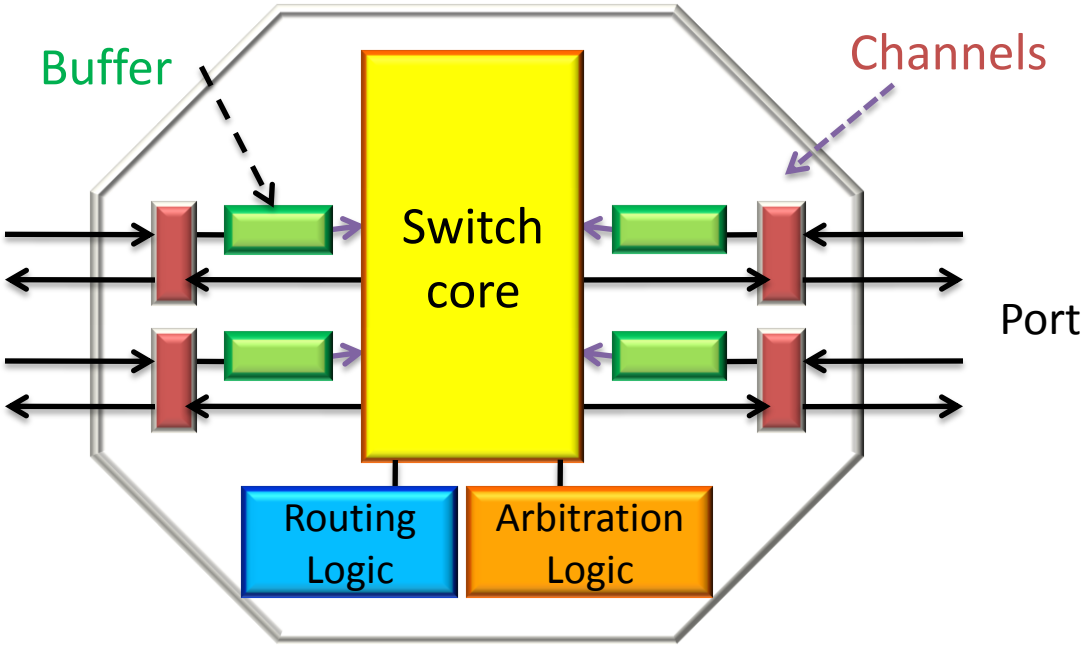


Homogeneous *NoC*

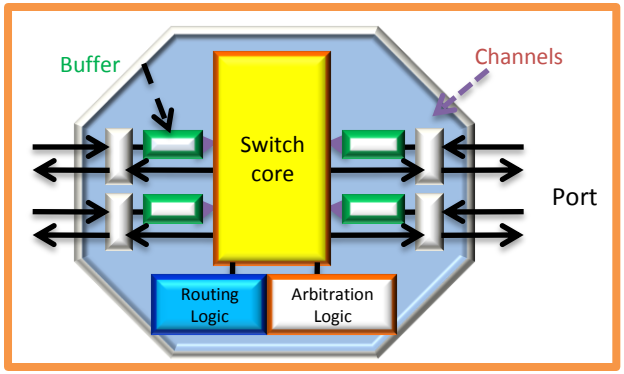
Heterogeneous *NoC*

NoC parameters: **Switch technique** (1)

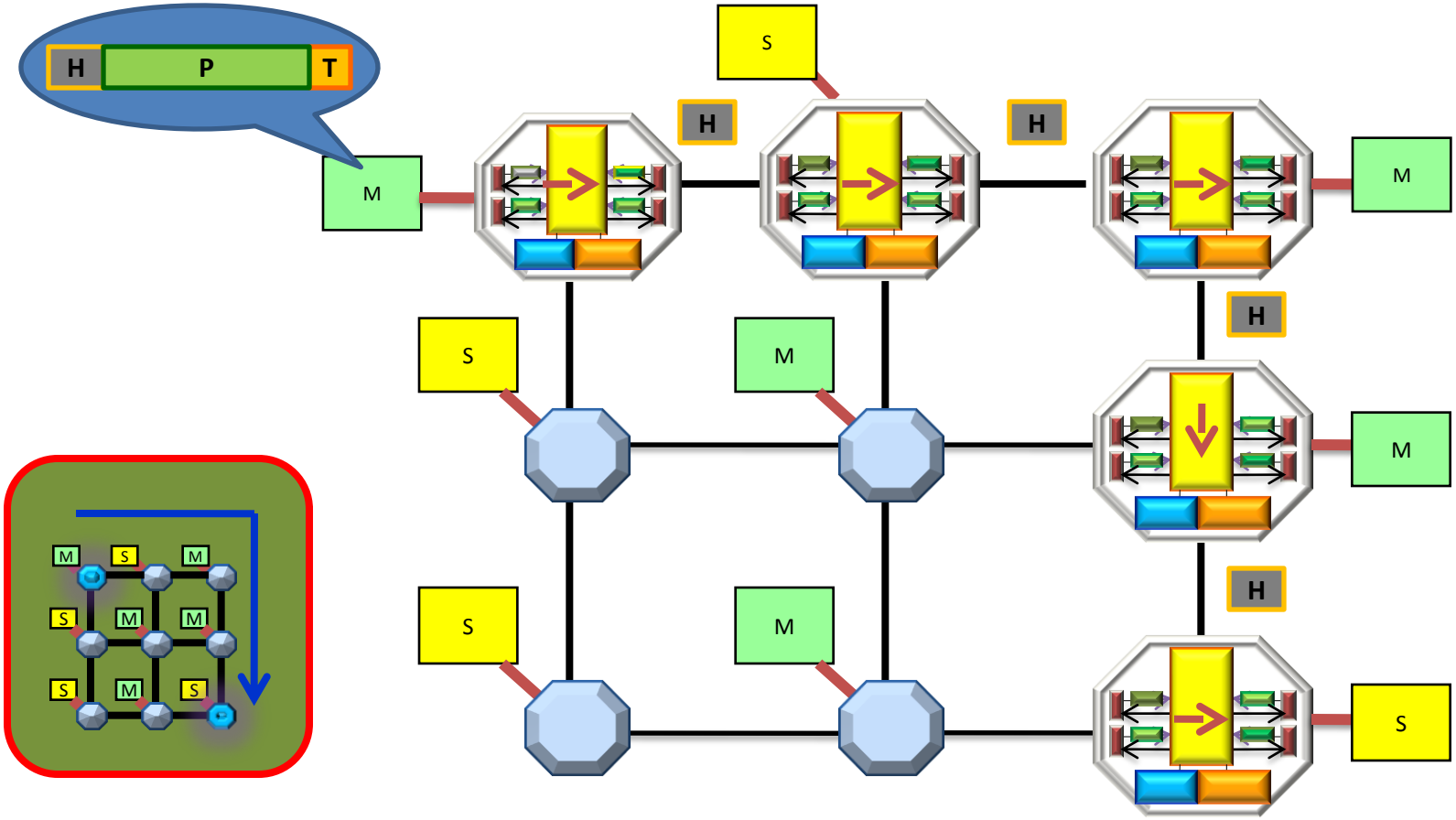




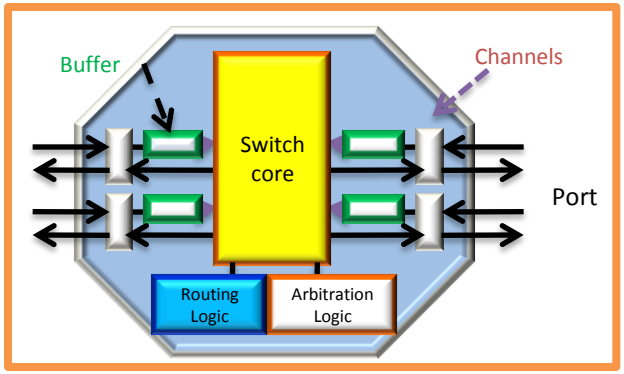
NoC parameters: **Switch technique** (2)



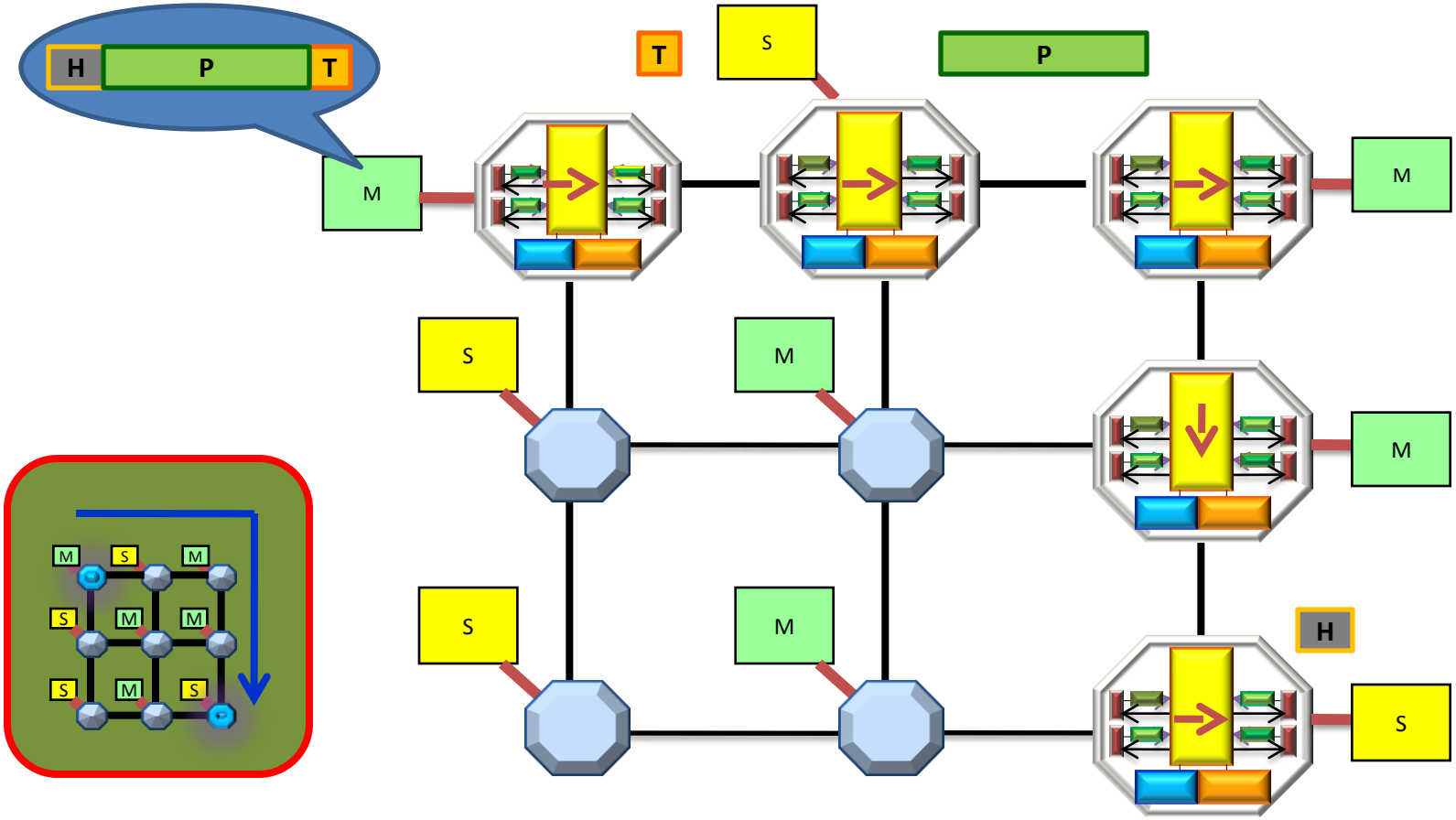
1. Circuit switch



NoC parameters: **Switch technique** (3)

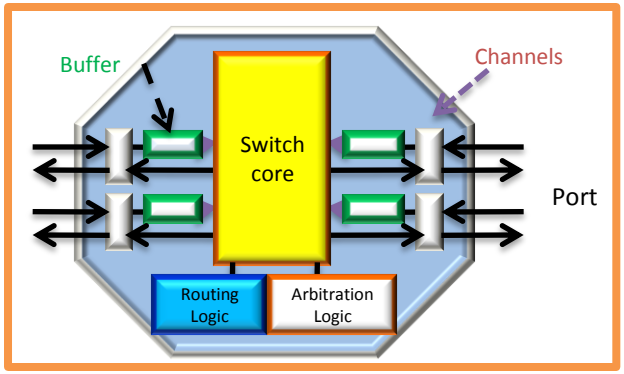


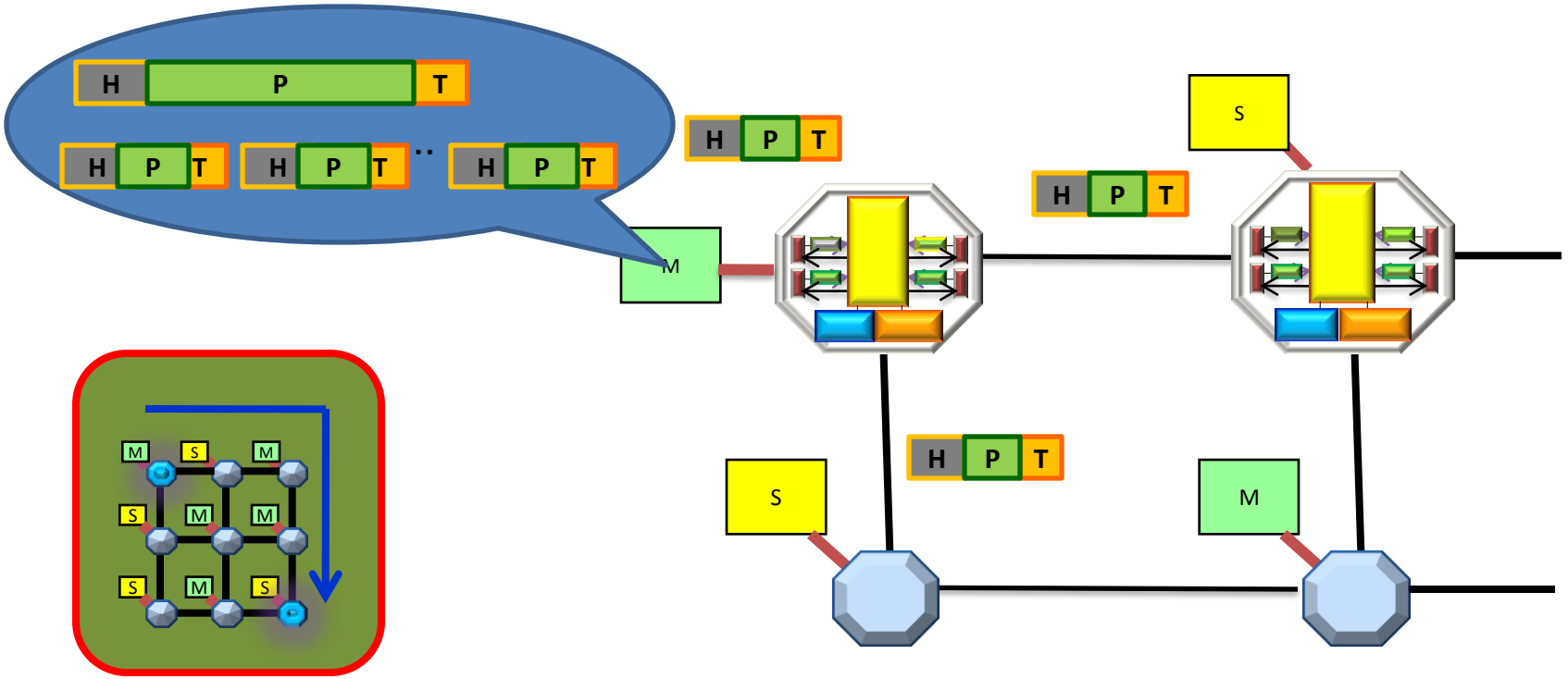
1. Circuit switch



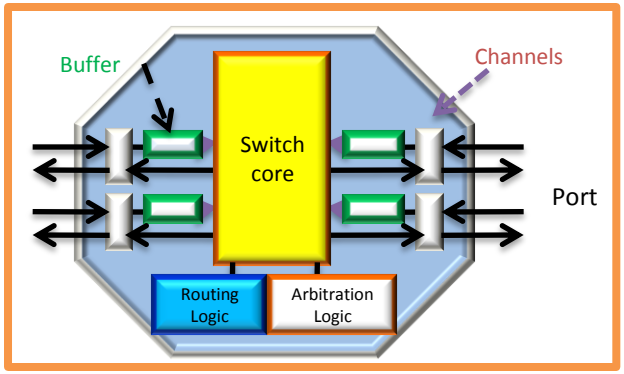
NoC parameters: **Switch technique** (4)

2. Packet switch

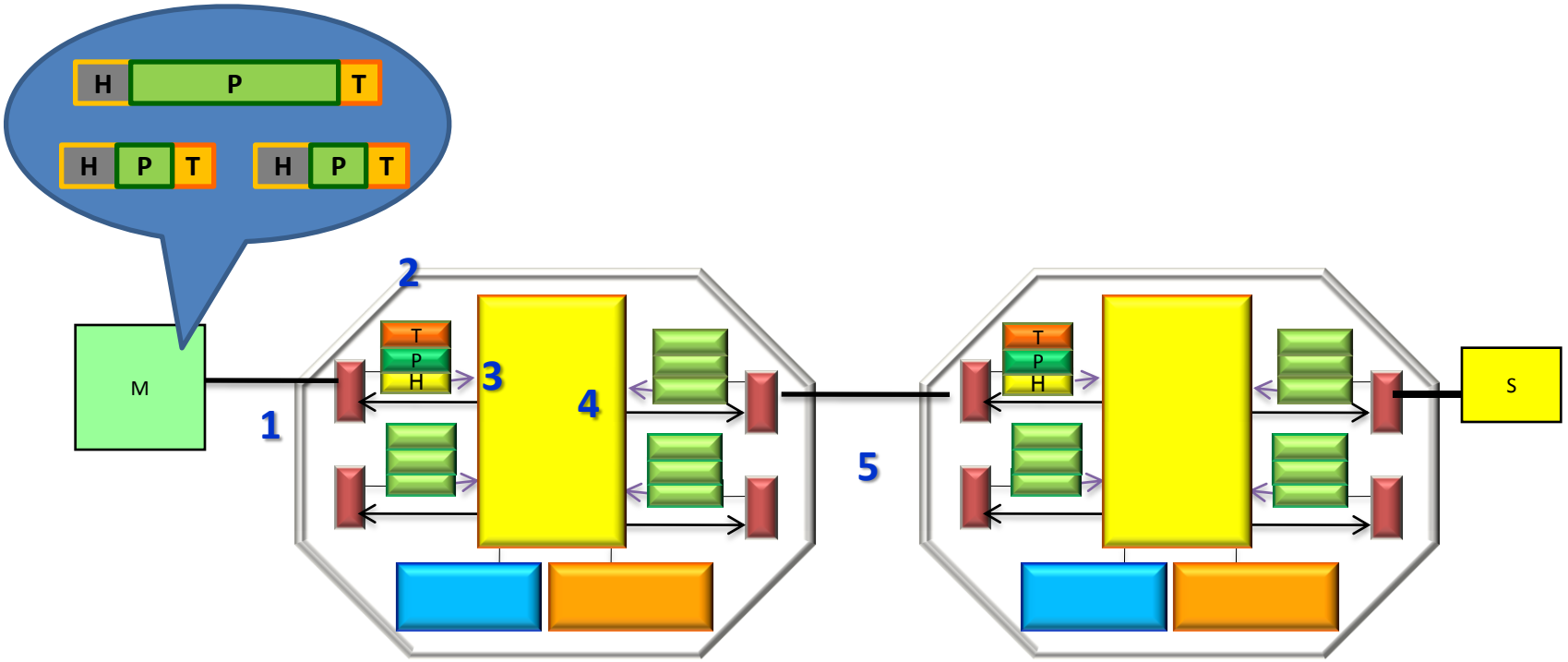




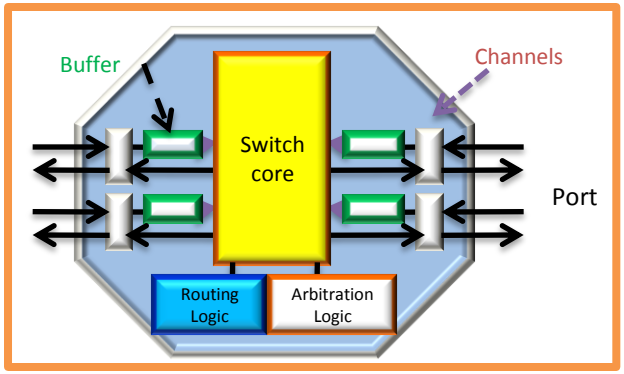
NoC parameters: **Switch technique** (5)



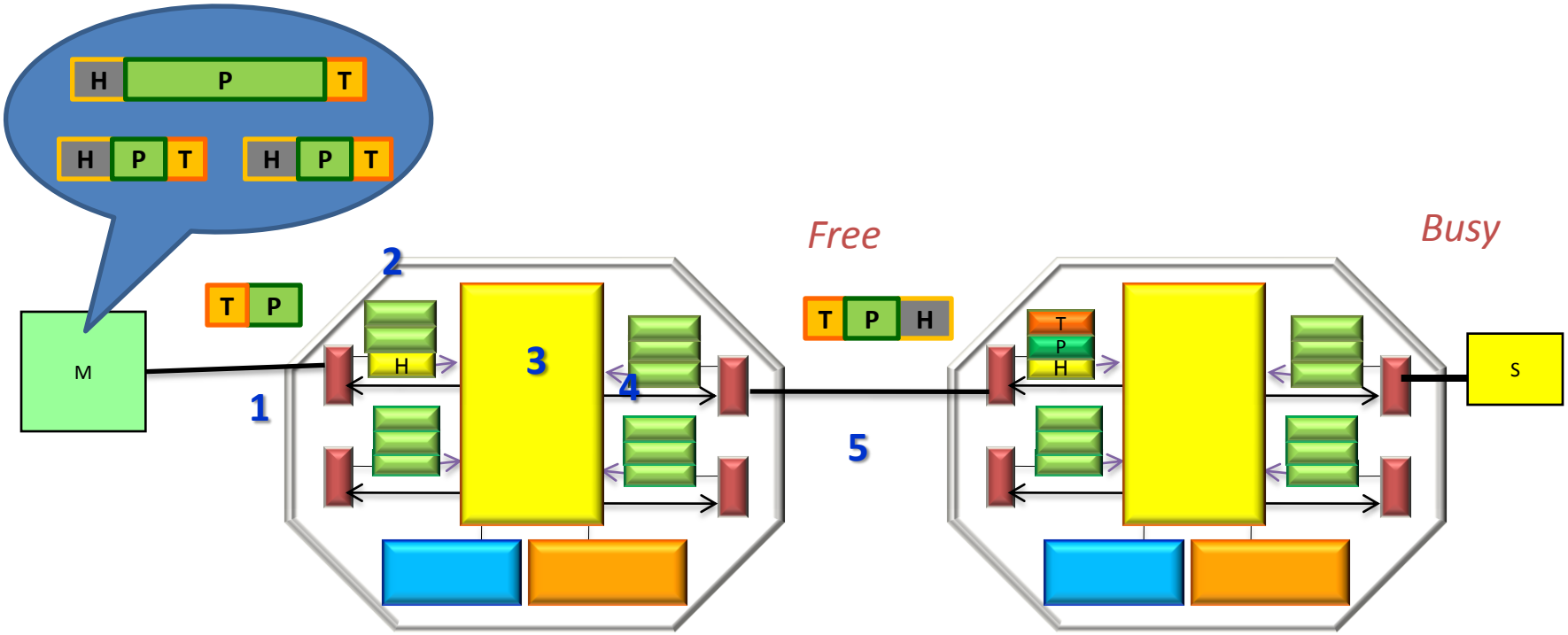
2. 1 Store-and-forward



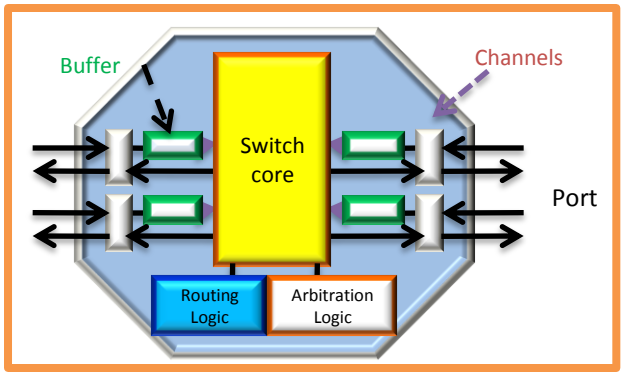
NoC parameters: **Switch technique** (6)



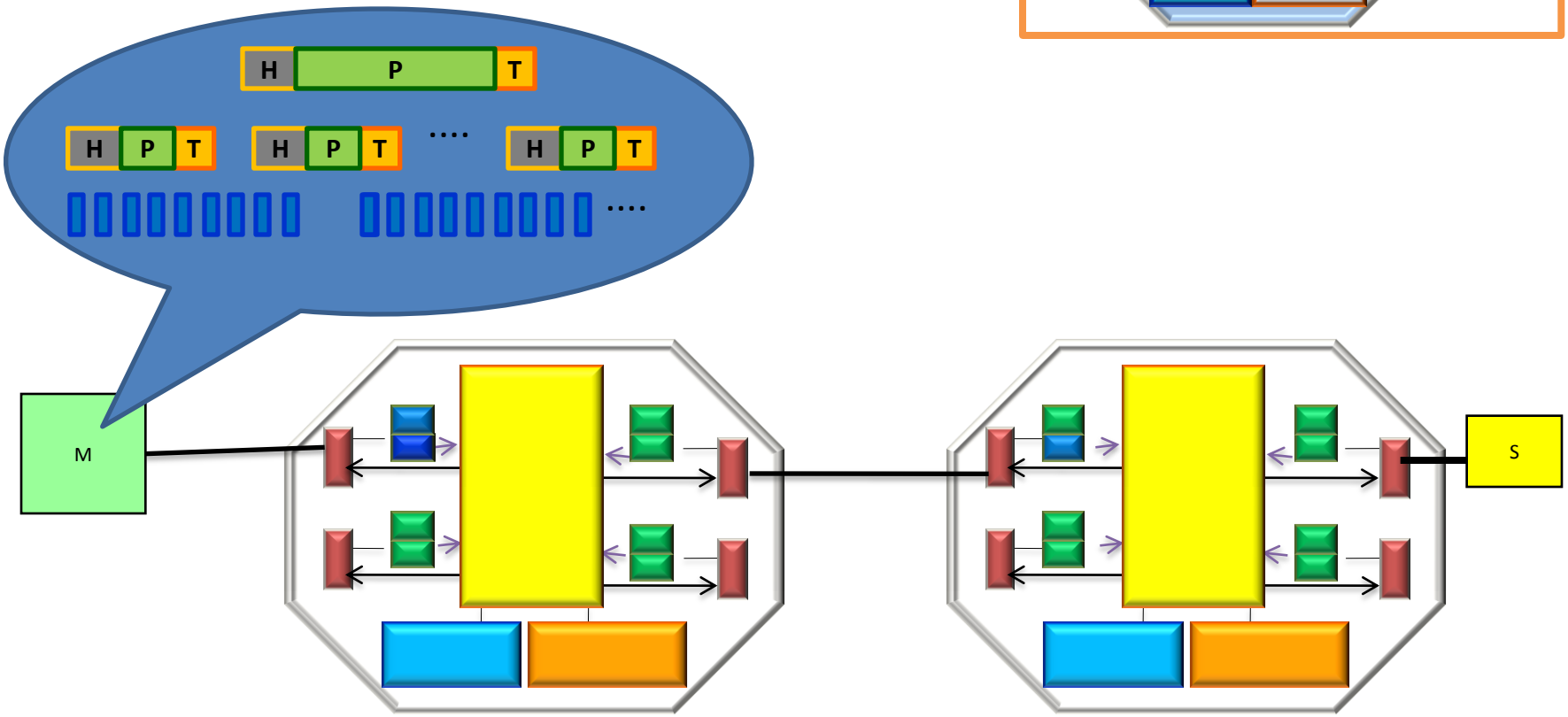
2. 2 Virtual cut-through



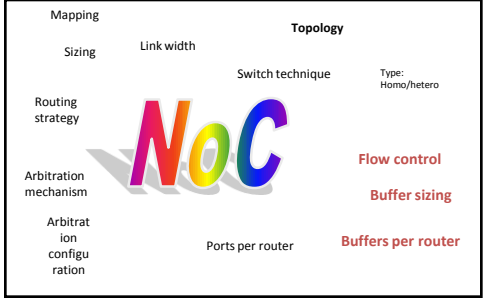
NoC parameters: **Switch technique** (7)



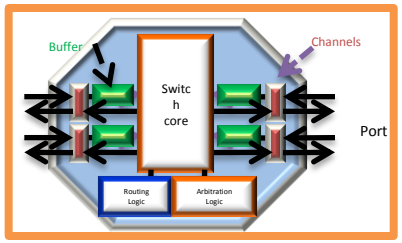
2.3 Wormhole



NoC parameters: **Flow Control** (1)

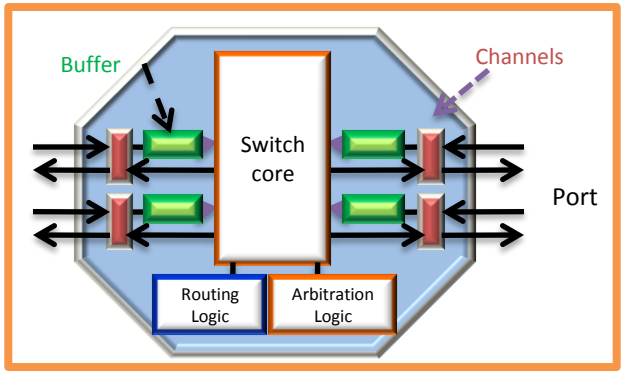


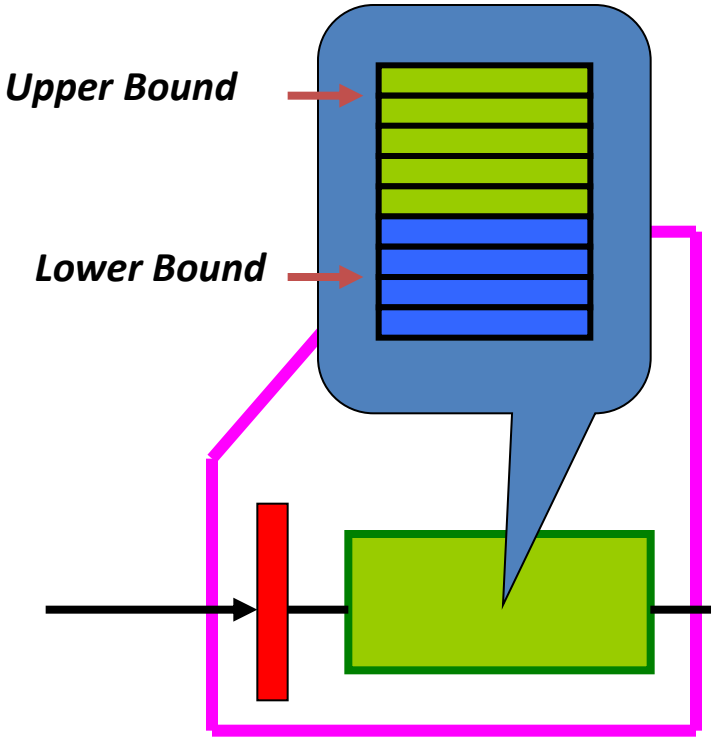
- Resource competition (collisions).
- Decisions about packet management:
  - Discard.
  - Block (receive and store).
  - Deviate.

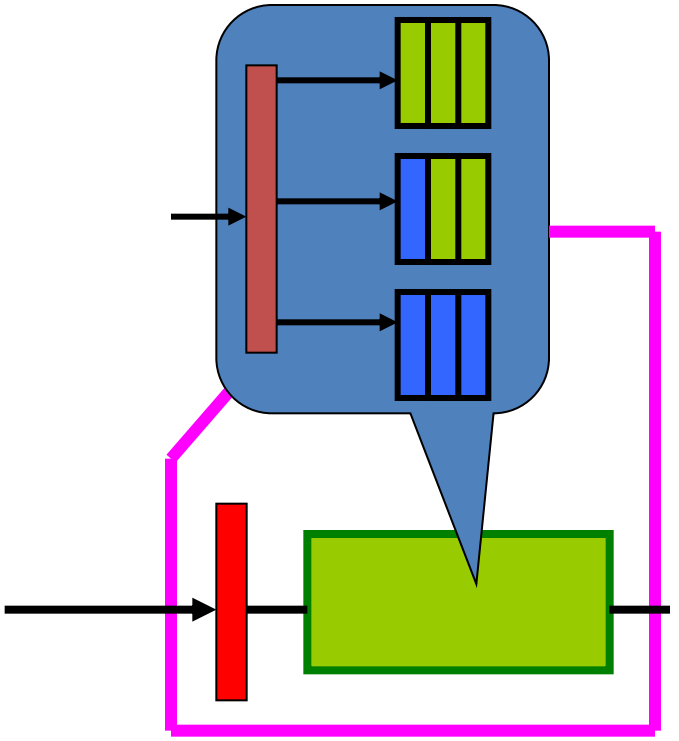


Buffer and channel allocation

NoC parameters: **Flow Control** (2)



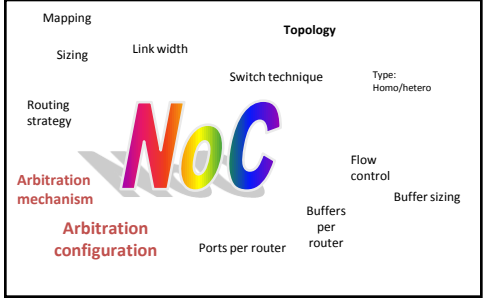


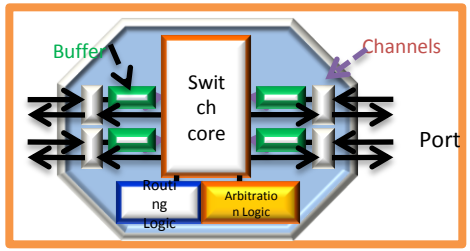


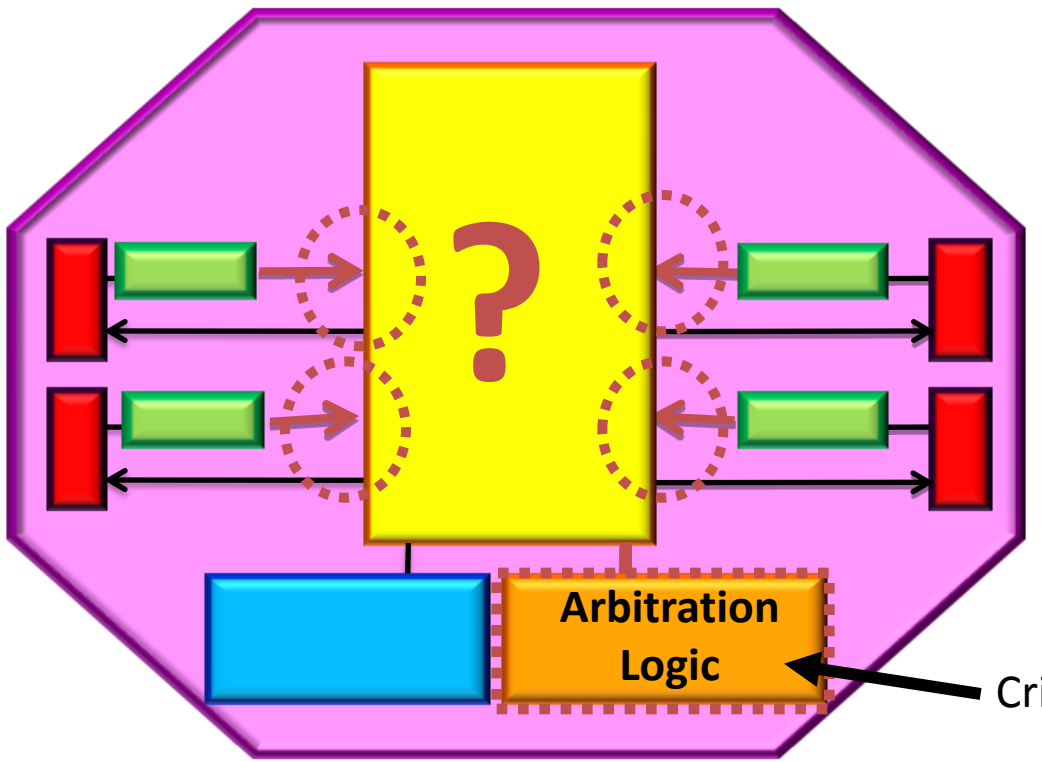
Slack Buffer

Virtual Channel

NoC parameters: **Arbitration Logic**



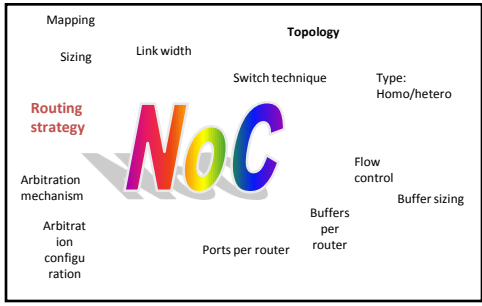


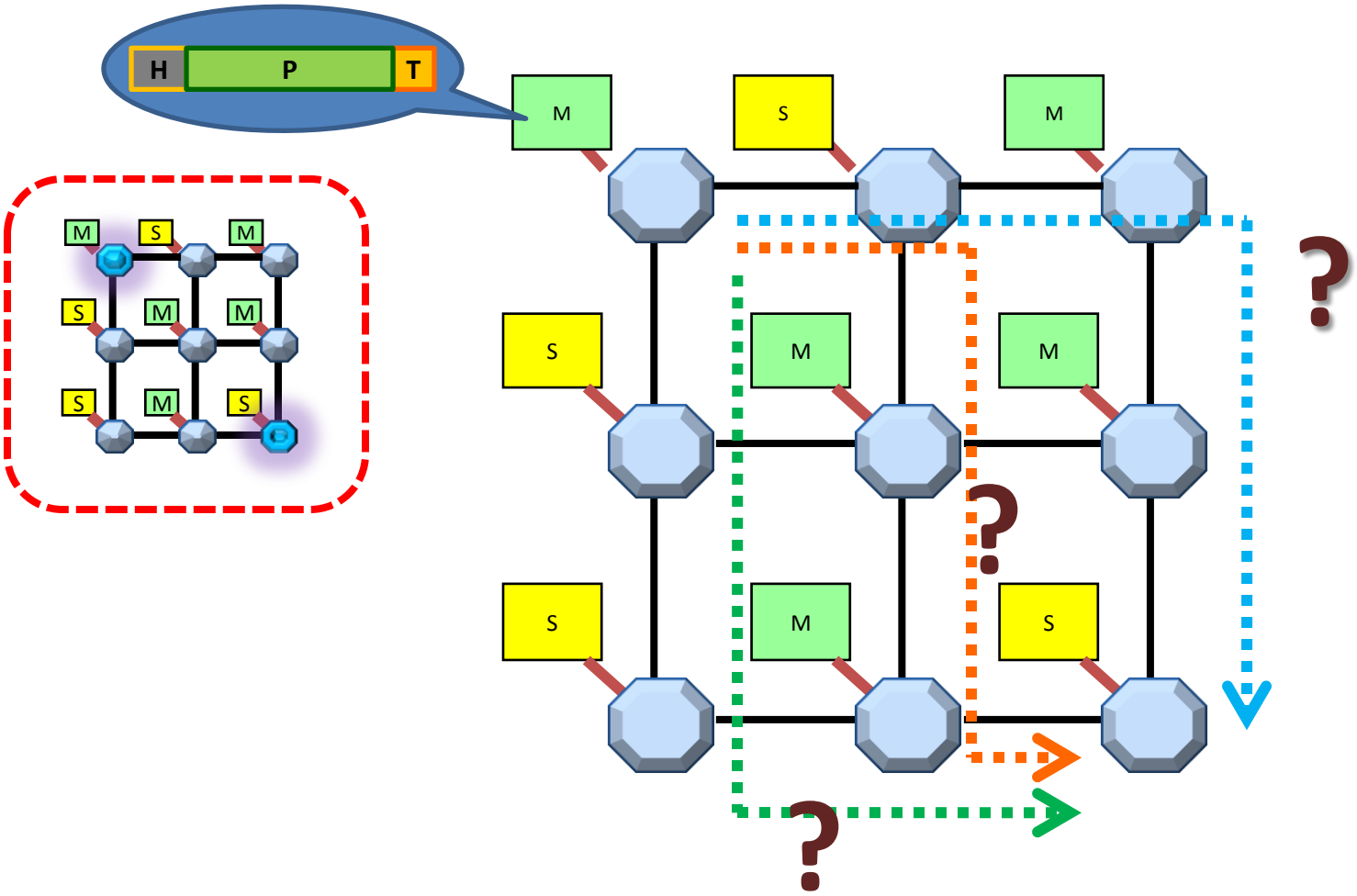


- Static priority
- Dynamic priority
- Deadline

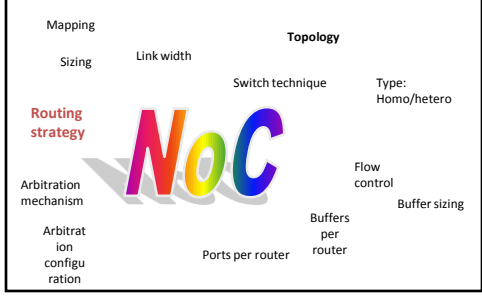
Criteria

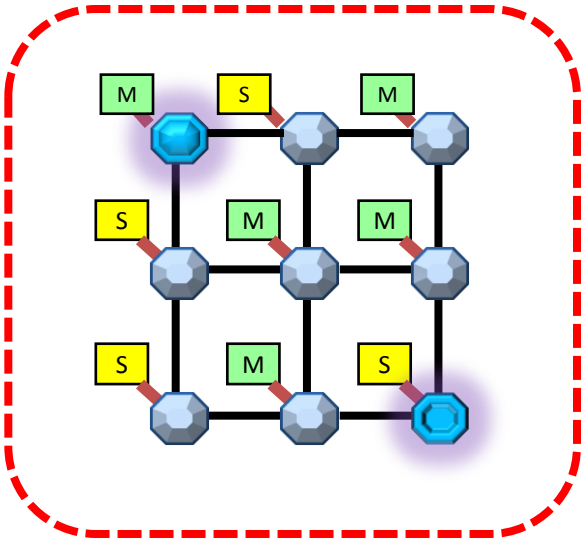
NoC parameters: **Routing strategy** (1)





NoC parameters: **Routing strategy** (2)





1. Implementation

- Table
- FSM

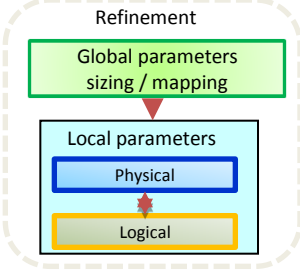
2. Routing moment

- Static (compilation)
- Dynamic (execution)
  - Progressive
  - Minimal

3. Routing unit

- Central
- Source
- Distributed

NoC
Configuration Parameters



Mapping

Sizing

Topology

Link width

Switch technique

Type:
Homo/hetero

Routing strategy

Flow control

Arbitration mechanism

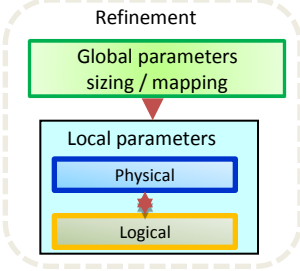
Buffer sizing

Buffers per
router

Arbitration
configuration

Ports per router

Huge search space!



NoC local parameters

- Logical parameters
  - Routing strategy (*XY, adaptive, west-first, north-last…)*
  - Arbitration mechanism and configuration
    - *Static: fixed priority (port priority), round-robin (port priority)*
    - *Dynamic*
    - *Deadline (arrival time)*
  - Switch technique (*circuit switch, packet switch*)
- Physical parameters
  - Ports per router (*2, 3, 4, 5, 6)*
  - Buffers per router (*1, 2….(n x # ports*))
  - Buffers size *(# bits, # flits, #words, #packets*)
  - Flow control (*single queue, slack buffer, virtual channel*)
  - Type (*homogeneous, heterogeneous*)
  - Topology (*direct, indirect*)
  - Link width *(# flits, # bits*)

NoC: Good news

- ☺ Only point-to-point one-way wires are used, for all network sizes.
- ☺ Aggregated bandwidth scales with the network size.
- ☺ Routing decisions are distributed and the same router is re-instanciated, for all network sizes.
- ☺ NoCs increase the wires utilization (as opposed to ad-hoc p2p wires)

BUT…

☹ Internal network contention causes (often unpredictable) latency.

☹ The network has a significant silicon area.

☹ Bus-oriented IPs need smart wrappers.

☹ Software needs clean synchronization in multiprocessor systems.

☹ System designers need reeducation for new concepts.