

# Δίκτυα-σε-Chip

4-May-2023

1

## Από τα Δίκτυα Διασύνδεσης σε Δίκτυα-σε-Chip

Box-to-Box



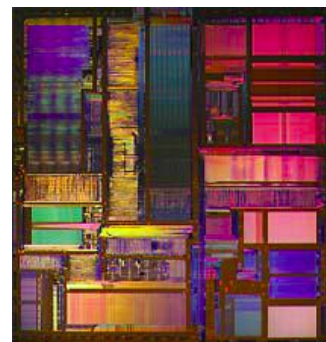
Board-to-board



Chip-to-Chip



On-Chip



Multi-Chip: Supercomputers, Data Centers, Internet Routers, Servers

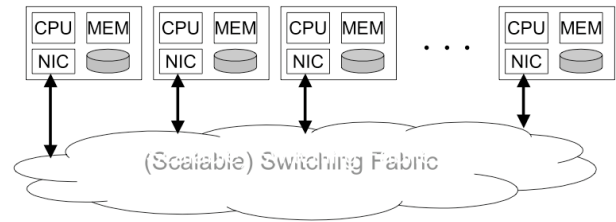
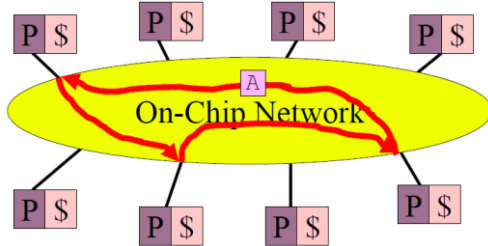
On-Chip: Servers, Laptops, Phones, HDTVs, Access routers, Smartphones

2

## Τι είναι ένα Δίκτυο-σε-Chip ?

Παράδειγμα: ένας πολυ-επεξεργαστής

Load R, Address

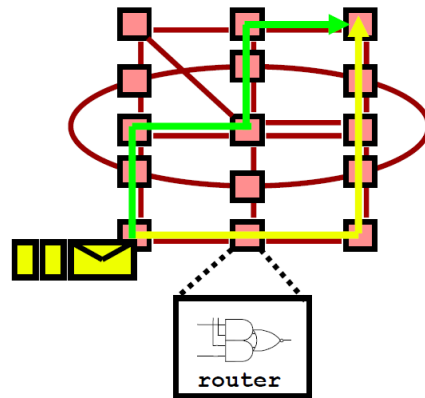


- Πρέπει να μεταφερθούν μηνύματα συγχρονισμού και γραμμές/block της cache μεταξύ των πυρήνων στο σύστημα

3

## Σχεδίαση ενός Δικτύου-σε-Chip

- Τοπολογία (καθυστέρηση, επεκτασιμότητα)
- Δρομολόγηση
- Έλεγχος ροής
- Αρχιτεκτονική ενός router ?



4

## Αρχιτεκτονική Δικτύου Διασύνδεσης

- Τοπολογία, πως συνδέονται οι κόμβοι (επεξεργαστές, μνήμες, κάρτες, κ.α.)
- Δρομολόγηση, ποιο μονοπάτι πρέπει να ακολουθήσει ένα μήνυμα
- Έλεγχος ροής, πως προχωράει το μήνυμα σε κάθε ενδιαμέσο σύνδεσμο από την πηγή έως τον προορισμό
- Αρχιτεκτονική Router: πως σχεδιάζεται ένας router
- Αρχιτεκτονική πρωτοκόλλου διασύνδεσης: πως σχεδιάζονται οι σύνδεσμοι

5

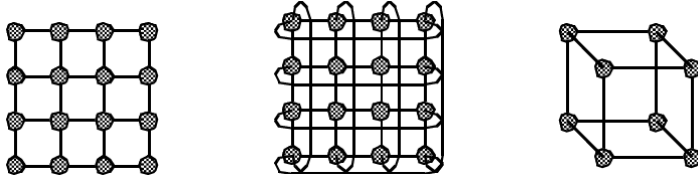
## Βασικές Τοπολογίες



- Δρομολόγηση από το A στο B βάσει διεύθυνσης
- Γραμμική τοπολογία, torus, κ.α
- Διάμετρος, μέση απόσταση, μέγιστη απόσταση,...
- Παραδείγματα torus: FDDI, SCI, Fiberchannel, Intel Xeon

6

## Mesh και Torus

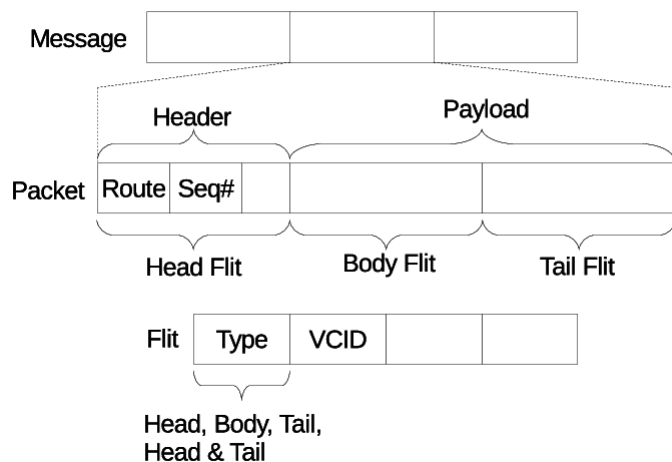


- Τοπολογία  $d$ -διαστάσης,  $N$  κόμβων βαθμού  $k$  (αριθμός συνδέσμων/κόμβο)
- Mesh, torus πολυ-διάστατα

7

## Μυνήματα – Πακέτα - Flit

- Πακέτο: η βασική μονάδα δεδομένων για δρομολόγηση (routing: επιλογή μονοπατιού)
- Flit: η βασική μονάδα σταθερού μεγέθους, όλα ακολουθούν το ίδιο μονοπάτι (flow control: δέσμευση buffer, έλεγχος ροής)



8

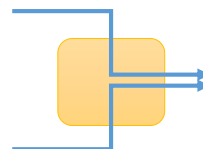
## Circuit ή Packet Switching?

- **Circuit switching:** ρυθμίζει την πλήρη διαδρομή και στέλνει δεδομένα (κανείς άλλος δεν μπορεί να χρησιμοποιήσει αυτούς τους συνδέσμους)
  - Ταχύτερη μεταφορά δεδομένων, καλοί ρυθμοί μετάδοσης
  - Χρονοβόρο στον καθορισμό της διαδρομής και στην αποδέσμευση
- **Packet switching** δρομολόγηση ανά πακέτο: δρομολόγηση καθενός πακέτου ανεξάρτητα από το προηγούμενο
  - Πρέπει να γίνεται δυναμικά σε κάθε πακέτο
  - Δεν υπάρχει κόστος αρχικό και τελικό καθορισμού μονοπατιού

9

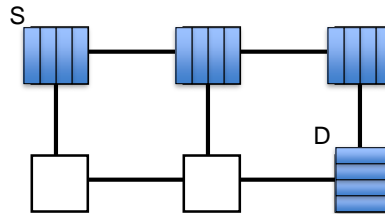
## Επιδόσεις Δικτύου, Καθυστέρηση

- Καθυστέρηση δρομολόγησης
  - Συνάρτηση της απόστασης και της μεταγωγής ανά κόμβο
  - Συνιστώσες: topology, routing algorithm, communicating nodes, switching strategy
- Καθυστέρηση λόγω συγκρούσεων
  - Ένας σύνδεσμος επικοινωνίας μπορεί να είναι κατηλλημένος από ένα μόνο μήνυμα
  - Εξαρτάται από την τοπολογία, την στρατηγική μεταγωγής και τον αλγόριθμο δρομολόγησης
- Τι γίνεται όταν 2 πακέτα θέλουν ταυτόχρονα να φύγουν από το ίδιο link?
  - Αποθήκευση του ενός
  - Απόρριψη του ενός
  - Ανακατεύθυνση σε άλλο link (deflection)



10

## Έλεγχος Ροής

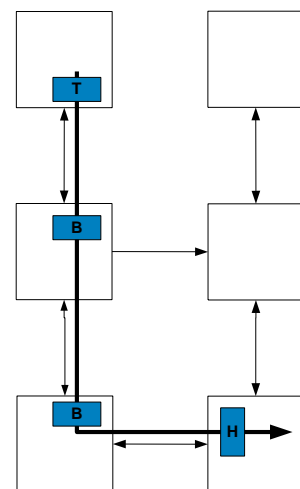


- Το πακέτο μεταφέρεται από router σε router: store-n-forward
  - Έλεγχος ροής σε κάθε πακέτο
  - Μεγάλη καθυστέρηση ανά πακέτο (για μεγάλα πακέτα)
- Έλεγχος ροής **cut-through**
  - Προώθηση μόλις φτάσει η επικεφαλίδα του πακέτου
  - Τι γίνεται αν η έξοδος είναι μπλοκαρισμένη? (μεγάλοι buffers !)

11

## Έλεγχος Ροής Wormhole

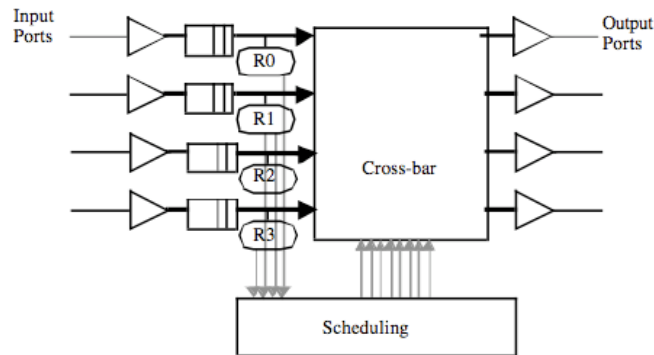
- Κάθε πακέτο τεμαχίζεται σε μικρότερες μονάδες, flits ίδιου μεγέθους
  - Body ακολουθεί την επικεφαλίδα, tail ακολουθεί το body
  - Pipelined
  - Αν η επικεφαλίδα μπλοκάρει, τότε το υπόλοιπο πακέτο σταματά
  - πληροφορία δρομολόγησης (src/dest) υπάρχει μόνο στην επικεφαλίδα
- Πως γνωρίζουν τα flit του πακέτου (body) που να πάνε?



12

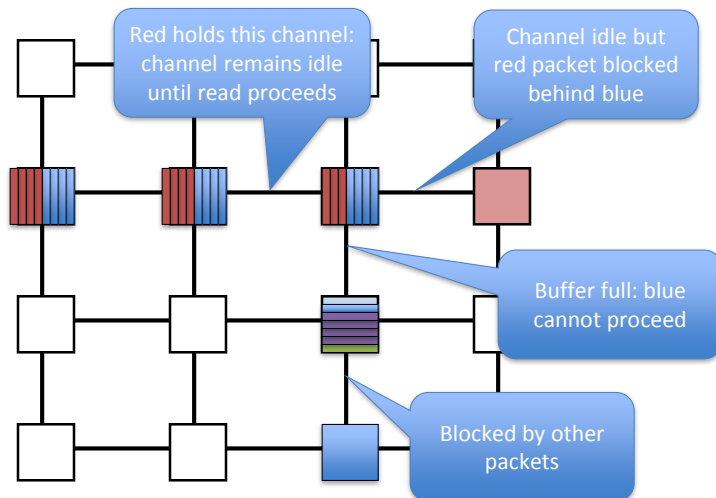
## Head of Line Blocking

- Λόγω της πολιτικής FIFO στις εισόδους, ένα worm δεν έχει τη δυνατότητα να προπεράσει ένα που βρίσκεται μπροστά του και έχει άλλο προορισμό (πόρτα εξόδου)



13

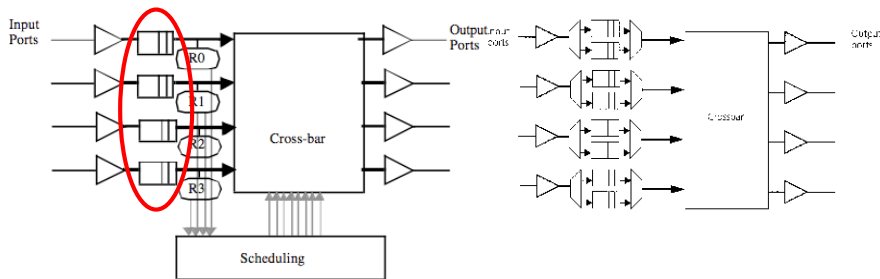
## Head of Line Blocking



14

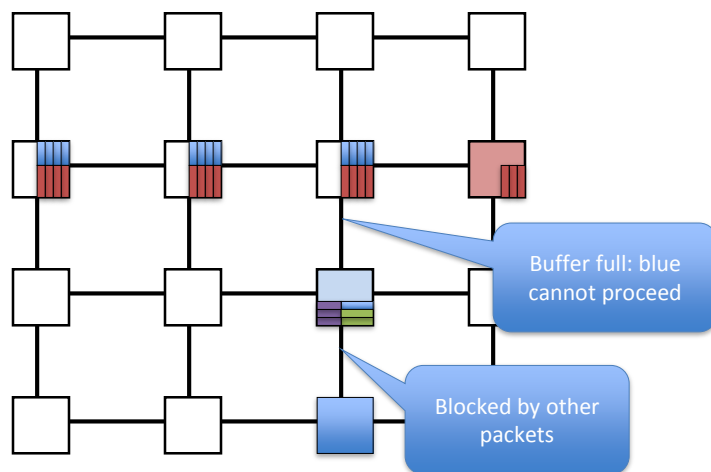
## Έλεγχος Ροής με Virtual Channel

- Βασική ιδέα: πολύπλεξη πολλών καναλιών πάνω από ένα φυσικό κανάλι
- Τεμαχισμός του buffer εισόδου σε πολλαπλούς buffers που μοιράζονται ένα φυσικό κανάλι
- Dally, “Virtual Channel Flow Control,” ISCA 1990.



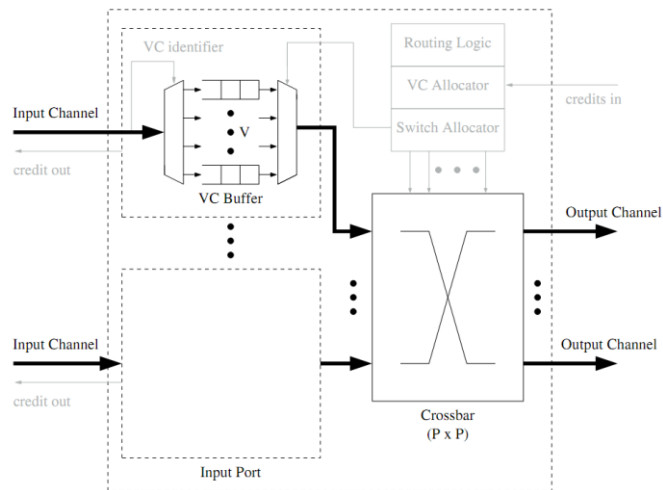
15

## Virtual Channel Flow Control



16

## Ένας Μοντέρνος Router με Virtual Channel

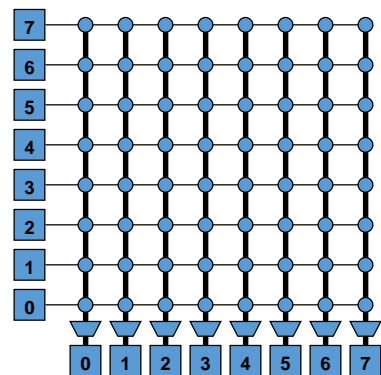


17

## Crossbar

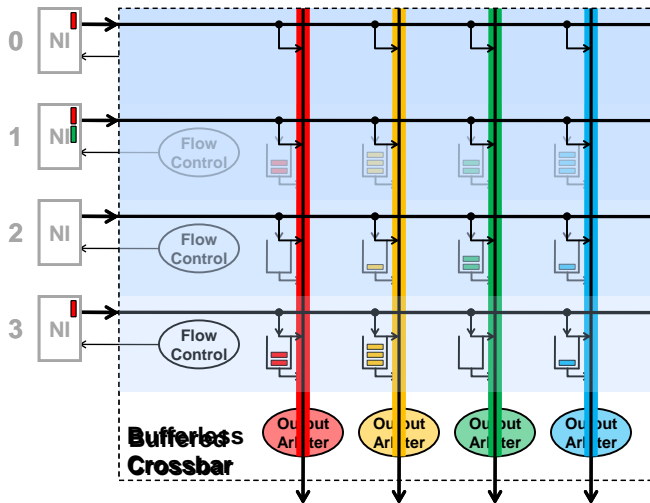
- Κάθε κόμβος συνδέεται με κάθε άλλον
- Καλό για μικρό αριθμό κόμβων
  - Μικρή καθυστέρηση, μεγάλο throughput
  - - ακριβό
  - - μη επεκτάσιμο ( κόστος:  $O(n^2)$  )

- IBM POWER5
- Sun Niagara I/II



18

## Buffered Crossbar



+ απλούστερο scheduling

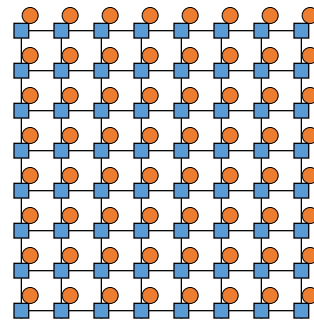
+ αποδοτικό και για πακέτα μεταβλητού μεγέθους

- Απαιτεί  $N^2$  buffers

19

## Mesh

- Κόστος  $O(N)$
- Μέση καθυστέρηση:  $O(\sqrt{N})$
- Εύκολη σχεδίαση on-chip: κανονική δομή και σύνδεσμοι ίσου μήκους
- Ποικιλία μονοπατιών πολλοί δρόμοι να φτάσεις από τον ένα κόμβο στον άλλο
- Κατασκευασμένο στον Tileria 100-core



20

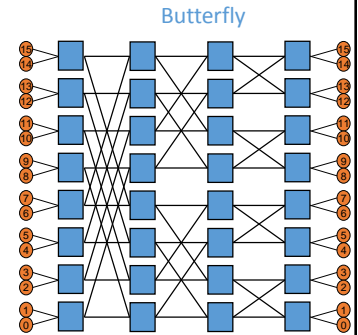
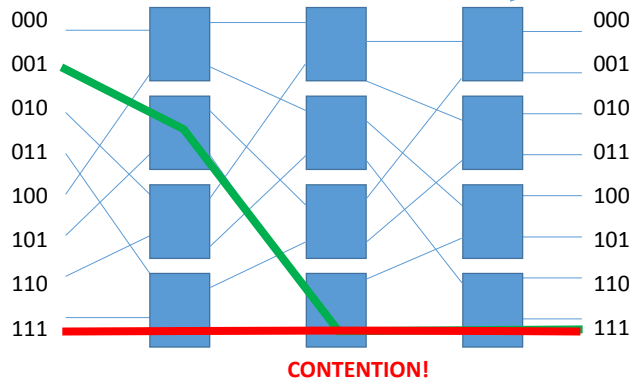
## Δίκτυα Πολλαπλών Σταδίων

Βασική ιδέα:  $\log N$  switches μεταξύ κόμβων

+ Cost  $O(N \log N)$

Πολλές παραλλαγές διάταξης (Omega, Butterfly, Benes, Banyan, ...)

Each switch is a 2x2 crossbar



21

## Παραδείγματα

System	Topology	Routing	Switching	Flow ctrl
MIT RAW	2D mesh (32bit)	XY DOR	WH, no VC	Credit
UPMC SPIN	Fat Tree (32bit)	Up*/down*	WH, no VC	Credit
QuickSilver ACM	H-Tree (32bit)	Up*/down*	1-flit, no VC	Credit
UMass Amherst aSOC	2D mesh	Shortest-path	Pipelined CS, no VC	Timeslot
Sun T1	Crossbar (128bit)	-	-	ACK/NACK
Cell BE EIB	Ring (128bit)	Shortest-path	Pipelined CS, no VC	Credit
TRIPS (operand)	2D mesh (109bit)	YX DOR	1-flit, no VC	On/off
TRIPS (on-chip)	2D mesh (128bit)	YX DOR	WH, 4 VCs	Credit
Intel SCC	2D torus (32bit)	XY,YX DOR, odd-even TM	WH, no VC	On/off
TILE64 iMesh	2D mesh (32bit)	XY DOR	WH, no VC	Credit
Intel 80-core NoC	2-D mesh (32bit)	Source routing	WH, 2 lanes	On/off

22