

# ΚΥΚΛΩΜΑΤΑ VLSI

Πανεπιστήμιο Ιωαννίνων



Τμήμα Μηχανικών Η/Υ και Πληροφορικής



Κεφάλαιο 14°

Γ. Τσατούχας



1

# ΚΥΚΛΩΜΑΤΑ VLSI

## Διάρθρωση



VLSI Systems  
and Computer Architecture Lab

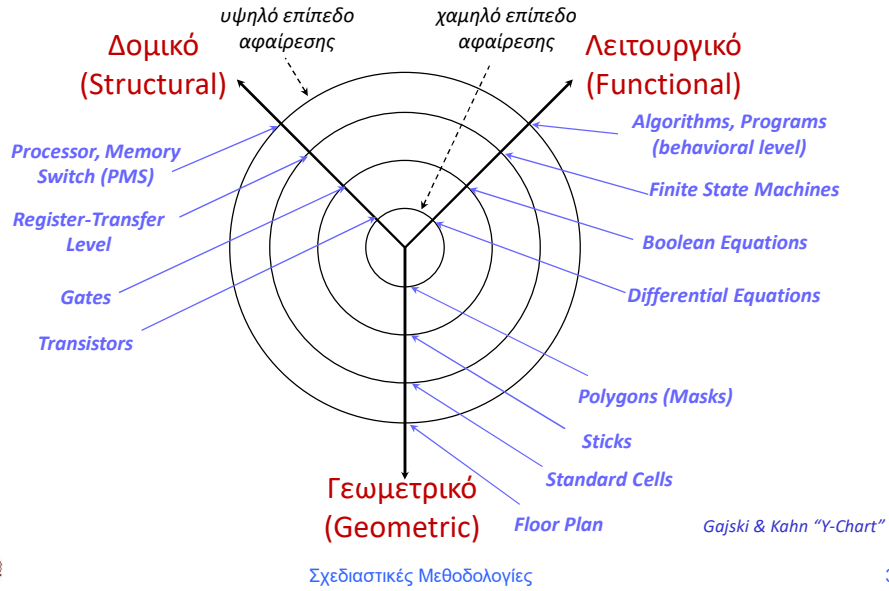
1. Σχεδιαστικά επίπεδα αφαίρεσης
2. Σχεδίαση σε επίπεδο τρανζίστορ
3. Σχεδίαση με δομικά κύτταρα
4. Σχεδίαση με διατάξεις
5. Προγραμματιζόμενες διατάξεις
6. Σχεδίαση με μακροκύτταρα
7. Συστήματα σε ένα ολοκληρωμένο (SoC)

Σχεδιαστικές Μεθοδολογίες

2

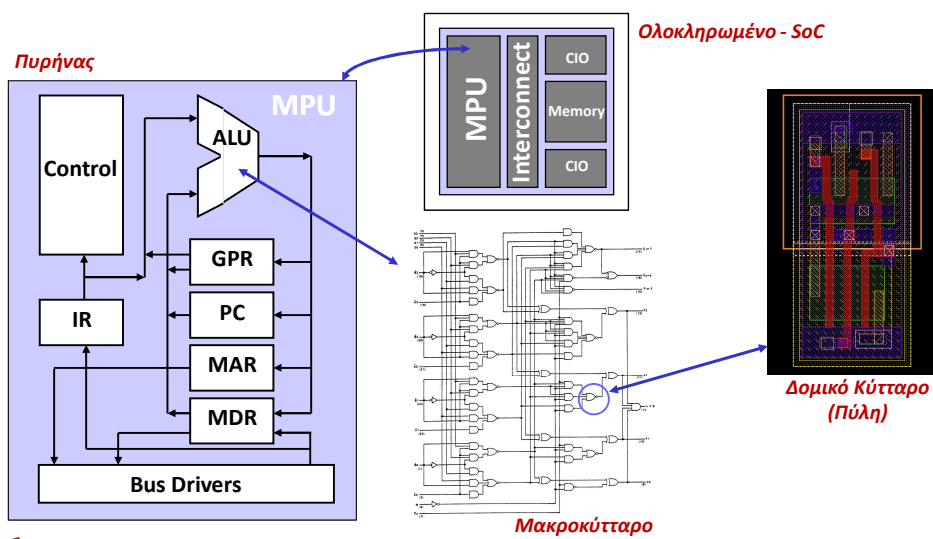
2

## Σχεδιαστικά Επίπεδα Αφαίρεσης



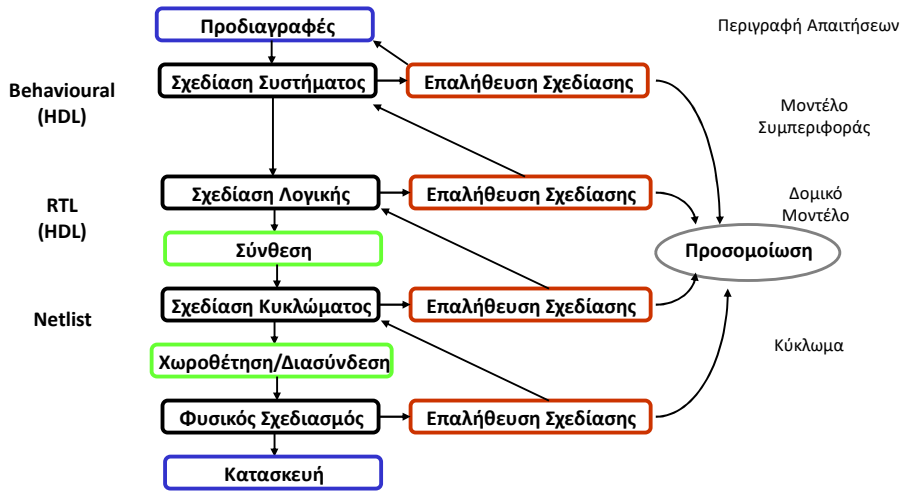
3

## Επίπεδα Σχεδίασης



4

## Σχεδιαστική Μεθοδολογία

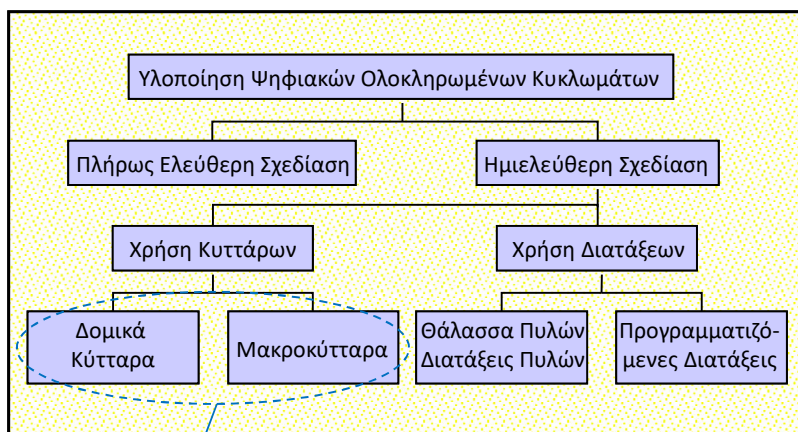


Σχεδιαστικές Μεθοδολογίες

5

5

## Σχεδιαστικές Επιλογές



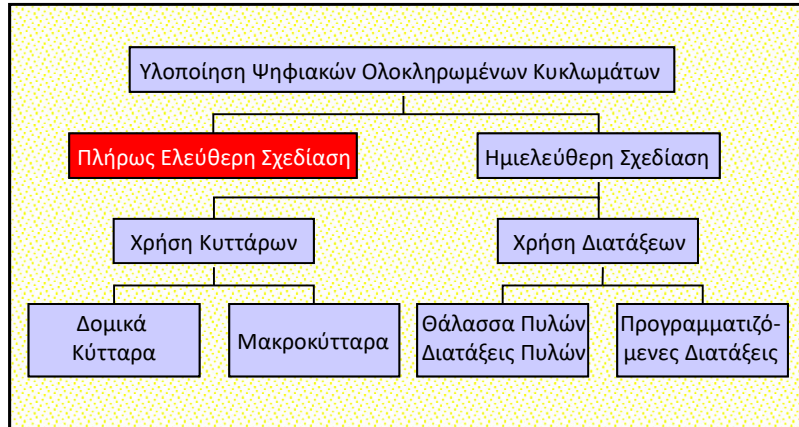
Σχεδίαση Ολοκληρωμένων Κυκλωμάτων Ειδικών Εφαρμογών  
Application Specific Integrated Circuits (ASICs)

Σχεδιαστικές Μεθοδολογίες

6

6

## Σχεδίαση στο Επίπεδο των Τρανζίστορ

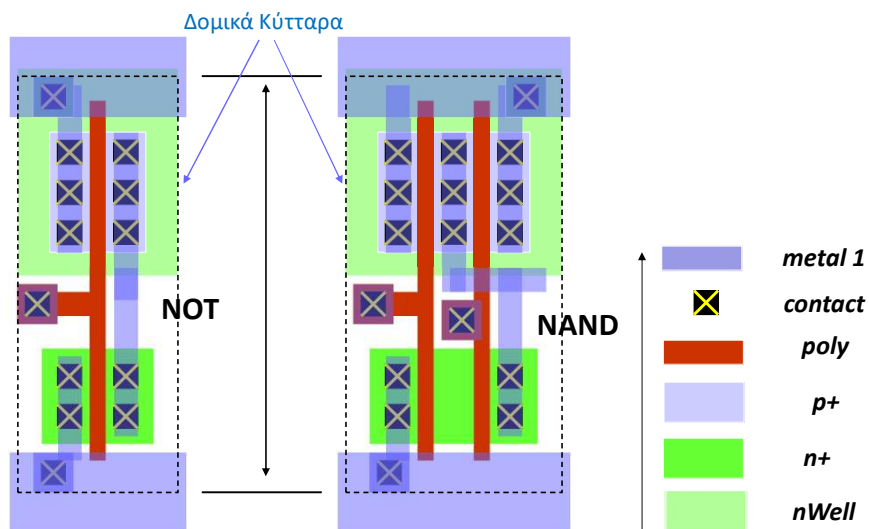


Σχεδιαστικές Μεθοδολογίες

7

7

## Σχεδίαση Κυττάρων

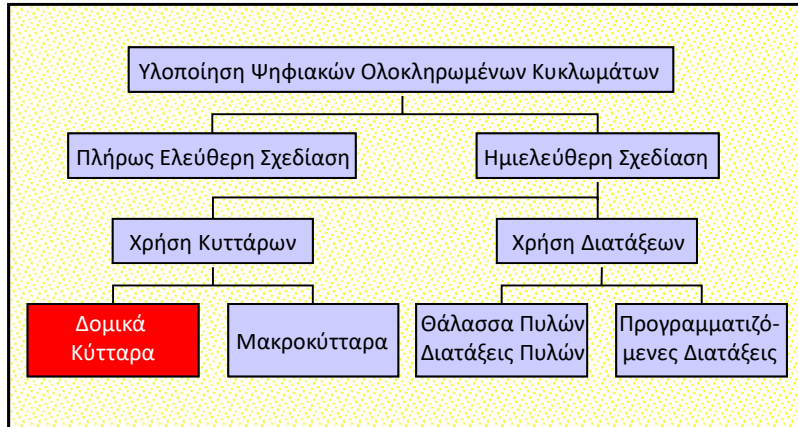


Σχεδιαστικές Μεθοδολογίες

8

8

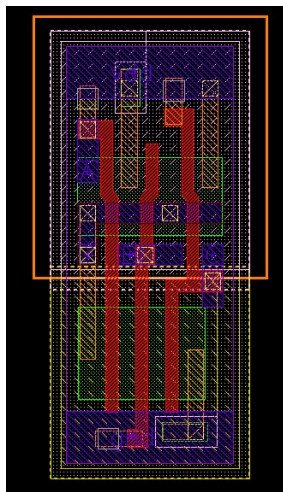
## Σχεδίαση με Δομικά Κύτταρα



## Δομικά Κύτταρα

Standard Cells

NAND 3-εισόδων



ST-Microelectronics

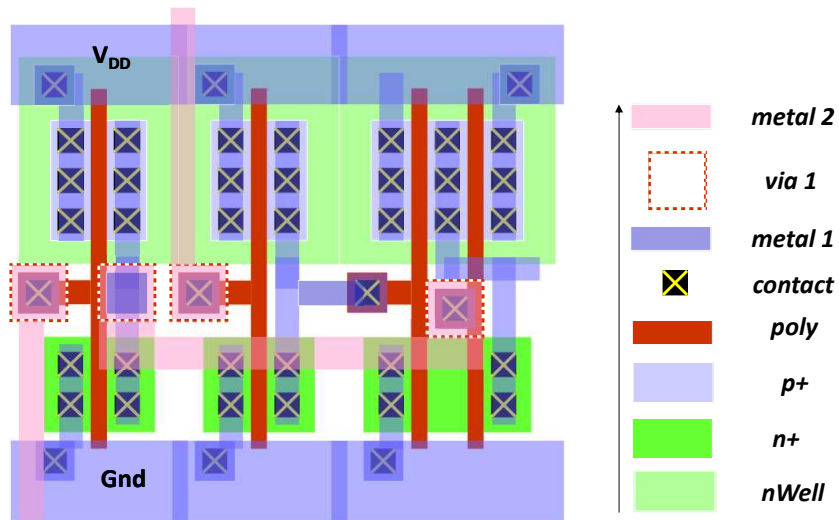
Χαρακτηριστικά Λειτουργίας

Path	1.2V - 125°C	1.6V - 40°C
$In1-t_{pLH}$	$0.073+7.98C+0.317T$	$0.020+2.73C+0.253T$
$In1-t_{pHL}$	$0.069+8.43C+0.364T$	$0.018+2.14C+0.292T$
$In2-t_{pLH}$	$0.101+7.97C+0.318T$	$0.026+2.38C+0.255T$
$In2-t_{pHL}$	$0.097+8.42C+0.325T$	$0.023+2.14C+0.269T$
$In3-t_{pLH}$	$0.120+8.00C+0.318T$	$0.031+2.37C+0.258T$
$In3-t_{pHL}$	$0.110+8.41C+0.280T$	$0.027+2.15C+0.223T$

$C$  = χωρητικότητα στην έξοδο  
 $T$  = χρόνος ανόδου / καθόδου σήματος εισόδου



## Σχεδίαση με Δομικά Κύτταρα (I)

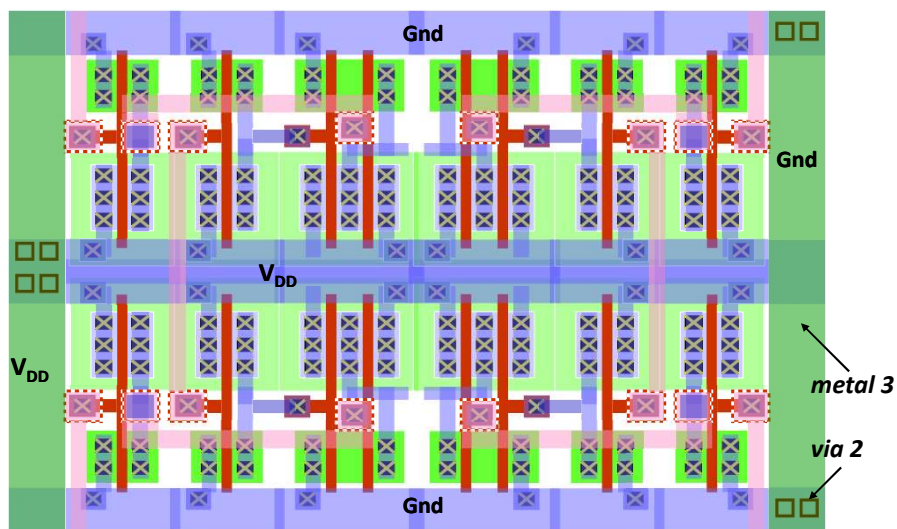


Σχεδιαστικές Μεθοδολογίες

11

11

## Σχεδίαση με Δομικά Κύτταρα (II)

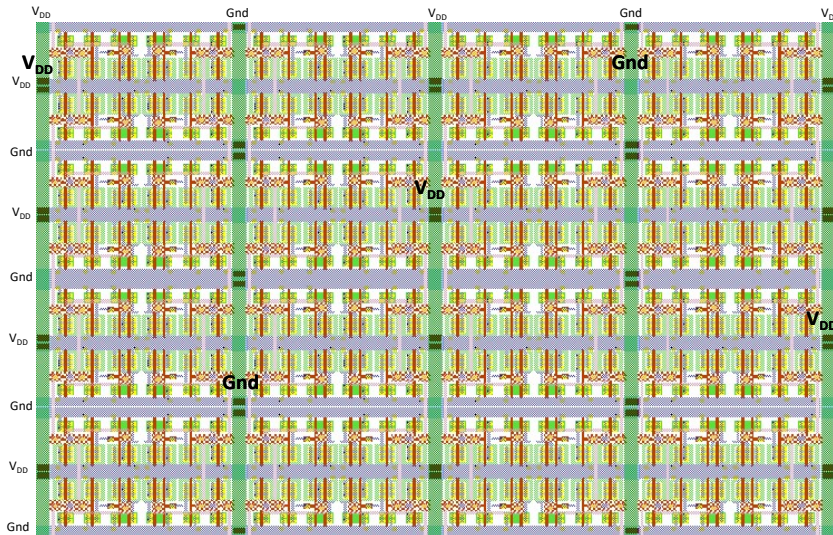


Σχεδιαστικές Μεθοδολογίες

12

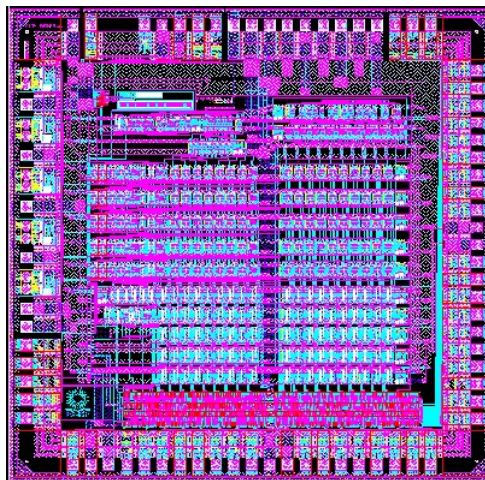
12

## Σχεδίαση με Δομικά Κύτταρα (IV)



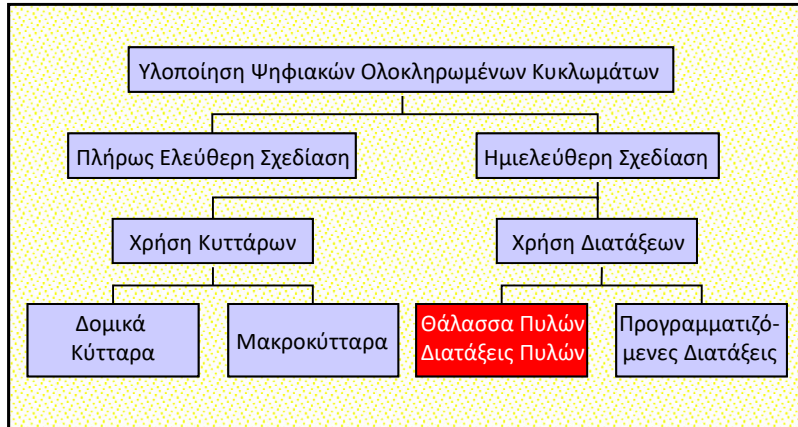
13

## Πλήρης Εικόνα Ολοκληρωμένου



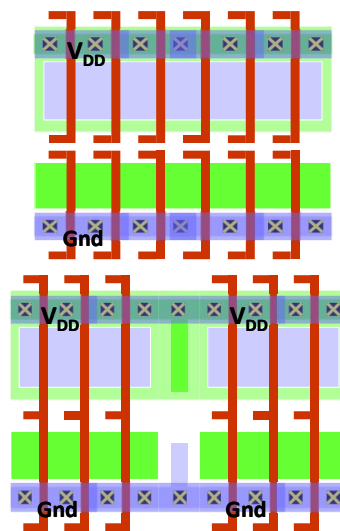
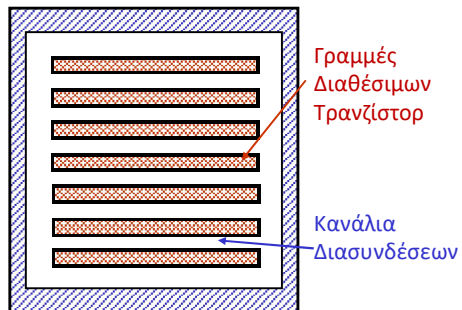
14

## Σχεδίαση με Διατάξεις Πυλών



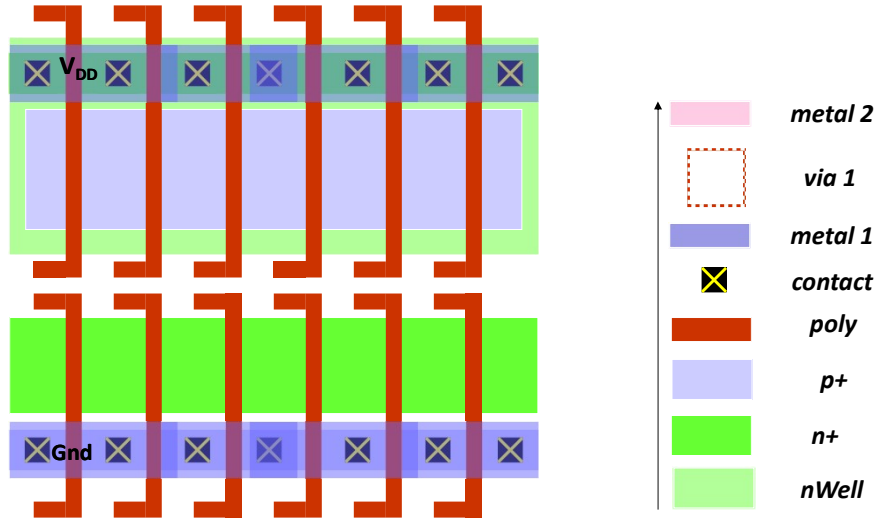
## Διατάξεις Πυλών - Θάλασσα Πυλών

Ολοκληρωμένο  
Διάταξης Πυλών - Θάλασσας Πυλών





## Κύτταρο Διάταξης με Μονωμένη Πύλη

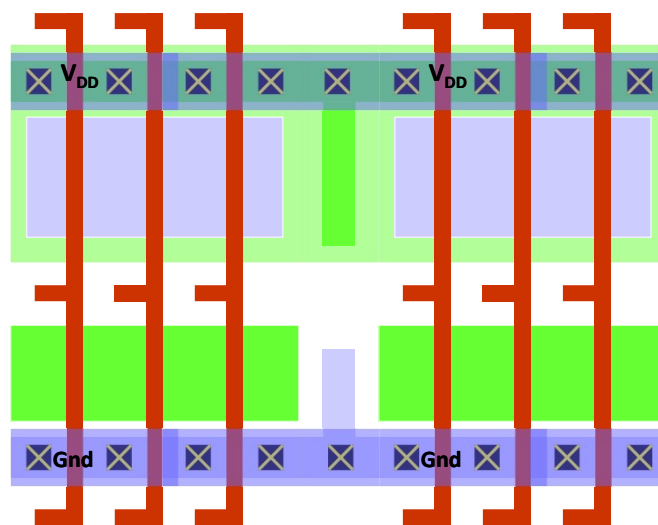


Σχεδιαστικές Μεθοδολογίες

17

17

## Κύτταρο Διάταξης με Μόνωση Οξειδίου

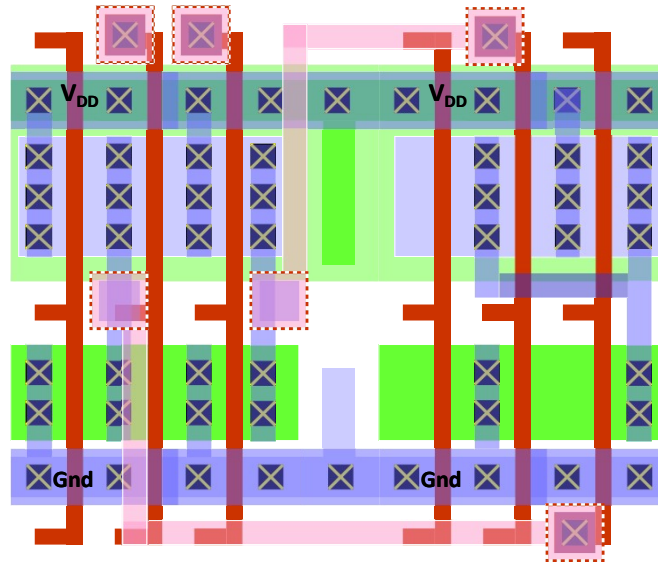


Σχεδιαστικές Μεθοδολογίες

18

18

## Διασυνδέσεις Κυττάρων

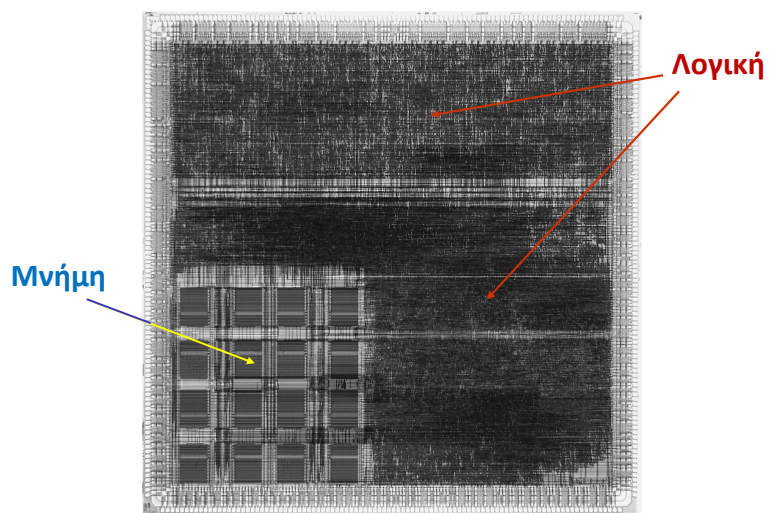


Σχεδιαστικές Μεθοδολογίες

19

19

## Ολοκληρωμένο Διάταξης Πυλών



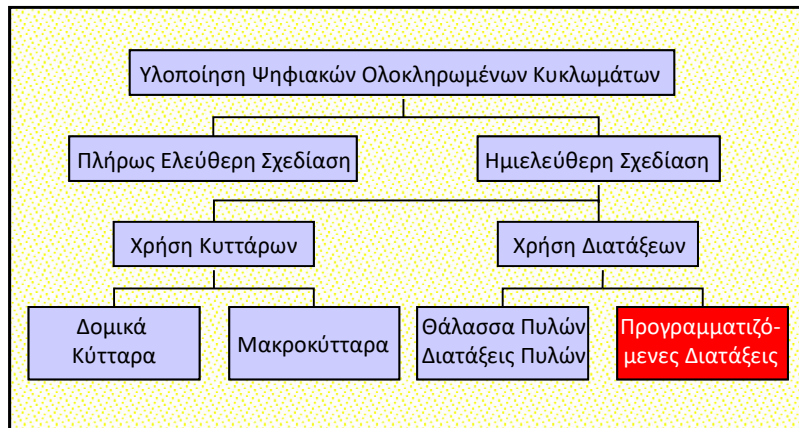
LSI Logic (0.6 μm CMOS)

Σχεδιαστικές Μεθοδολογίες

20

20

## Σχεδίαση με Προγραμματιζόμενες Διατάξεις



21

## Προγραμματιζόμενες Διατάξεις

### Κατηγοριοποιήσεις Προγραμματιζόμενων Διατάξεων:

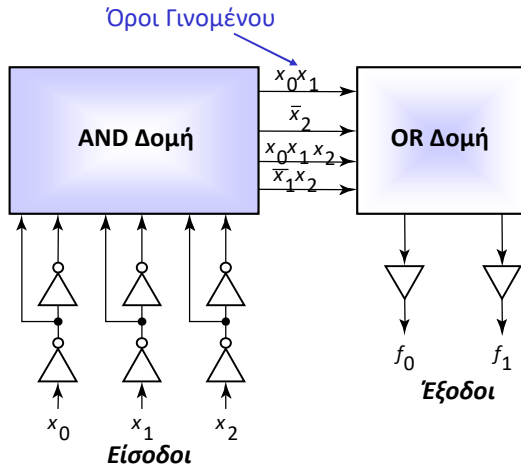
- Με βάση την τεχνική προγραμματισμού
  - Χρήση αντιασφαλειών (antifuses) – Μόνιμος προγραμματισμός
  - Χρήση μνημών EPROM
  - Χρήση μνημών RAM (και εξωτερικών EPROM)
- Με βάση τον τύπο της προγραμματιζόμενης λογικής
  - Χρήση διατάξεων λογικής (logic arrays)
  - Χρήση πινάκων αναφοράς (look-up tables)
- Με βάση τον τύπο των προγραμματιζόμενων διασυνδέσεων
  - Χρήση καναλιών διασύνδεσης (channel-routing)
  - Χρήση πλεγματικών δικτυωμάτων (mesh networks)



22

# Προγραμματιζόμενες Λογικές Διατάξεις

Programmable Logic Arrays (PLAs)



Κάθε λογική συνάρτηση μπορεί να εκφραστεί ως άθροισμα γινόμενων (AND-OR) δηλ. ως άθροισμα ελαχιστόρων.

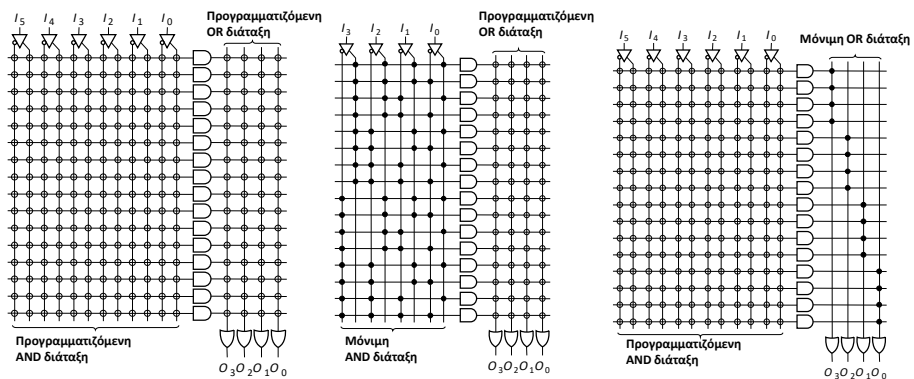
$$f_0 = x_0x_1 + \overline{x_2}$$

$$f_1 = x_0x_1x_2 + \overline{x_2} + \overline{x_1x_2}$$

Τα PLAs προγραμματίζονται ως προς τα γινόμενα (ελαχιστόρους) και/ή ως προς τα αθροίσματα που θα παράγουν.



# Προγραμματιζόμενες Λογικές Διατάξεις



PLA

PROM

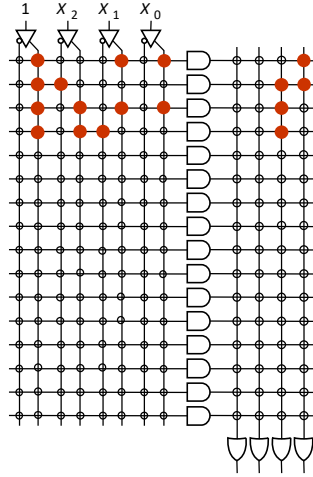
PAL

⊕ Προσδιορίζει προγραμματιζόμενη σύνδεση

● Προσδιορίζει μόνιμη σύνδεση



## Υλοποίηση Συναρτήσεων σε PLA



$$f_0 = x_0 x_1 + \overline{x_2}$$

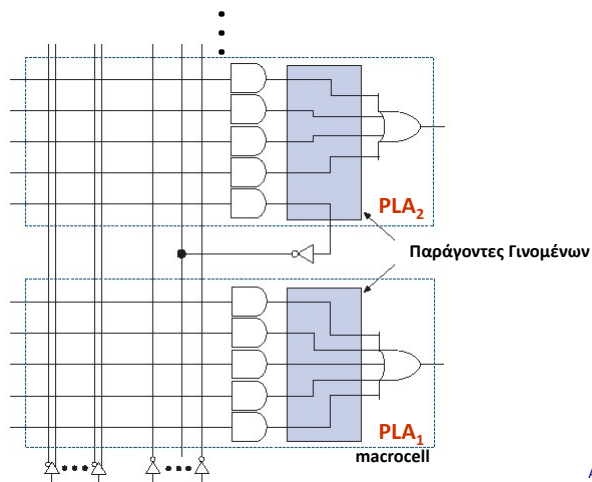
$$f_1 = x_0 x_1 x_2 + \overline{x_2} + \overline{x_1} x_2$$

● : προγραμματισμένος κόμβος NA NA  $f_1$   $f_0$

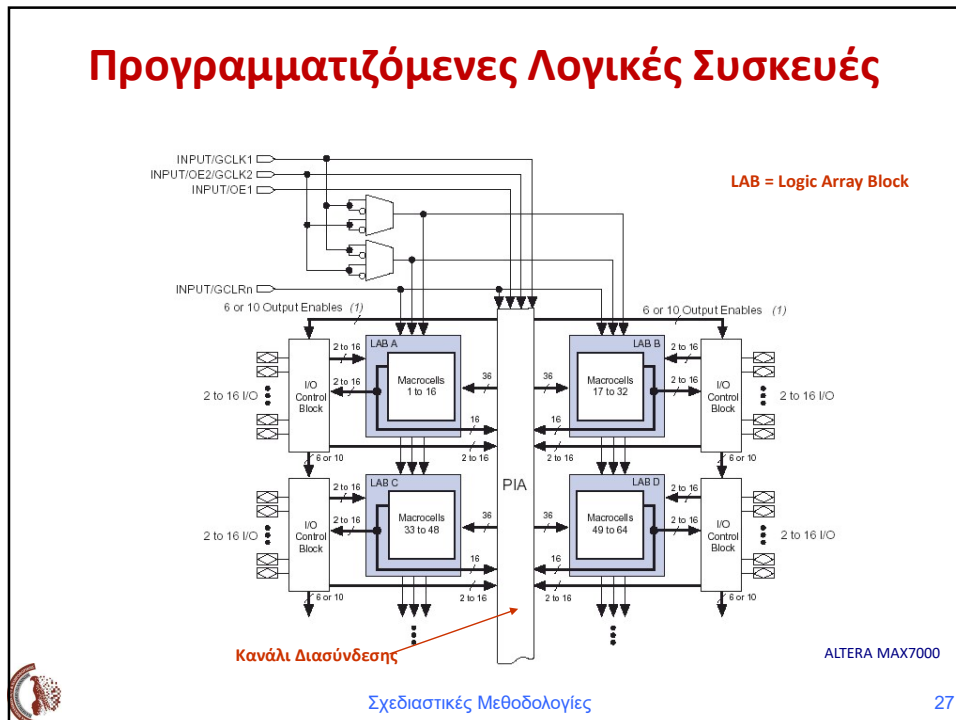


## Προγραμματιζόμενες Λογικές Συσκευές

Programmable Logic Devices (PLDs)

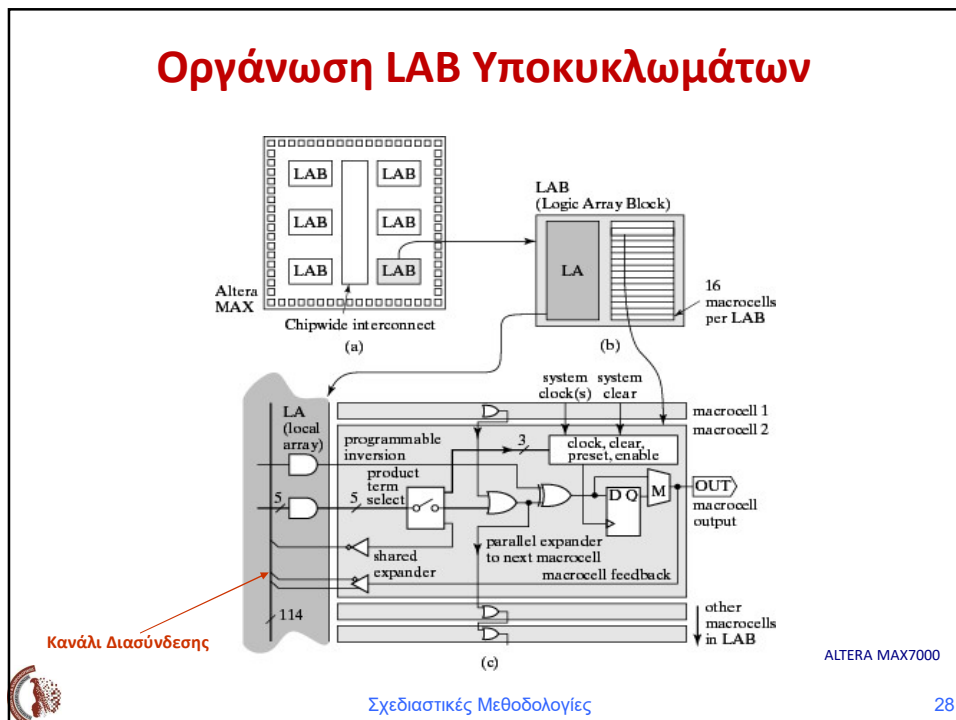


## Προγραμματιζόμενες Λογικές Συσκευές



27

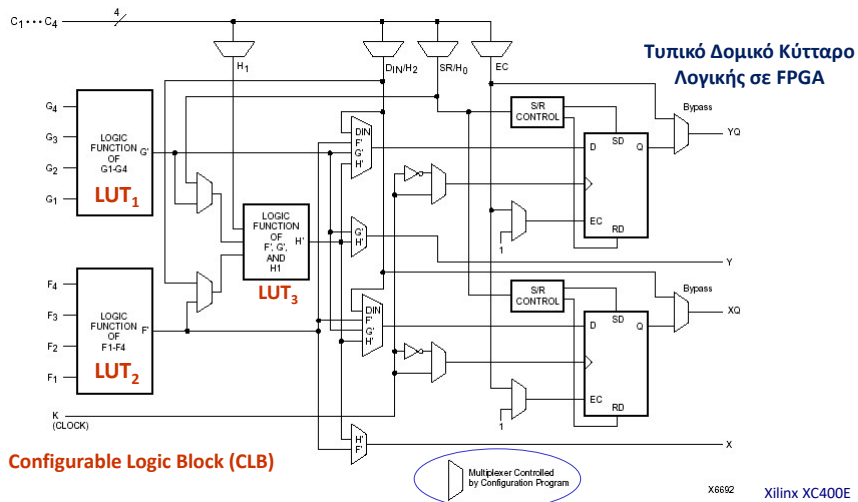
## Οργάνωση LAB Υποκυκλωμάτων



28

# Προγραμματιζόμενες Διατάξεις Λογικής

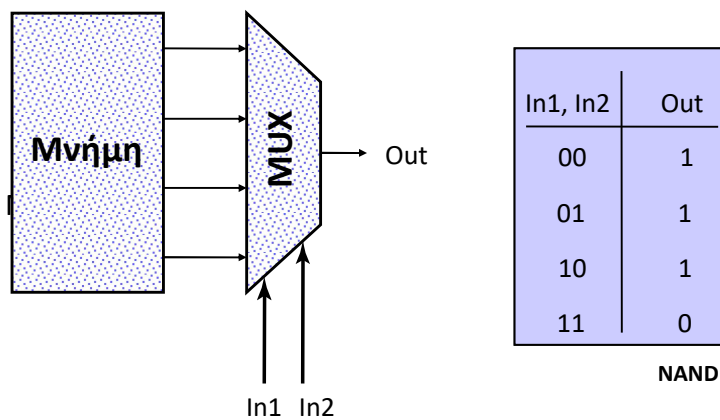
Field Programmable Gate Arrays (FPGAs)



29

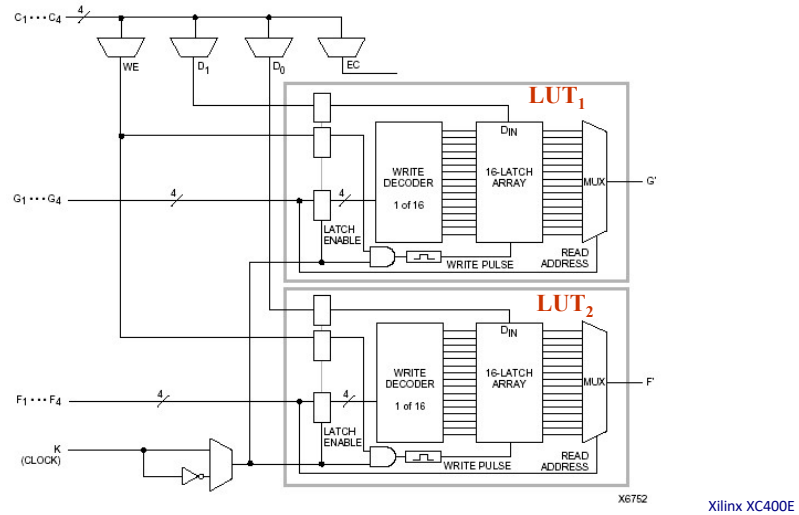
# Λογική με Χρήση Πινάκων Αναφοράς

Look-Up Table (LUT) Logic Cell



30

## Οργάνωση LUT Υποκυκλωμάτων

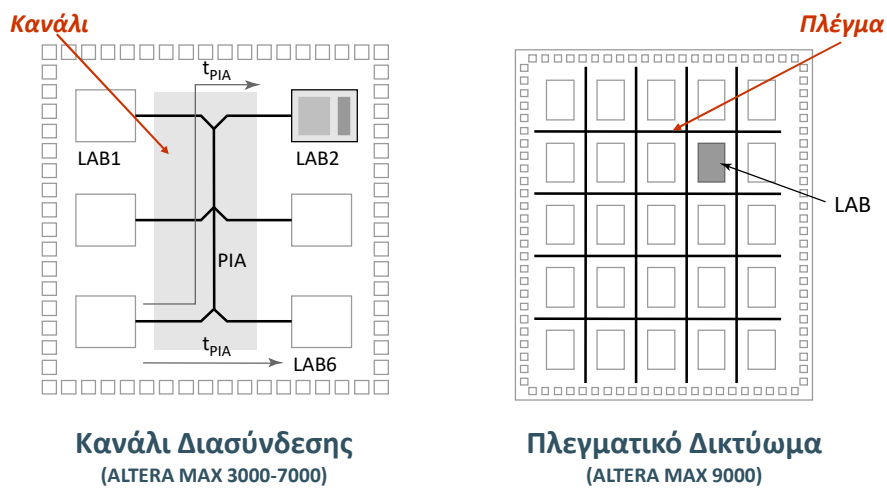


Σχεδιαστικές Μεθοδολογίες

31

31

## Δικτύωματα Διασύνδεσης



Κανάλι Διασύνδεσης  
(ALTERA MAX 3000-7000)

Πλεγματικό Δικτύωμα  
(ALTERA MAX 9000)

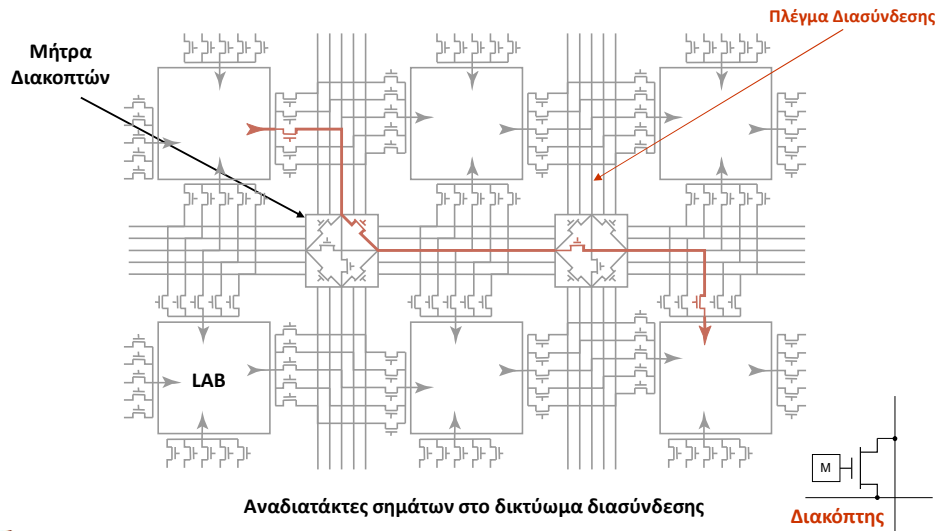
Σχεδιαστικές Μεθοδολογίες

32

32



## Πλεγματοκό Δικτύωμα

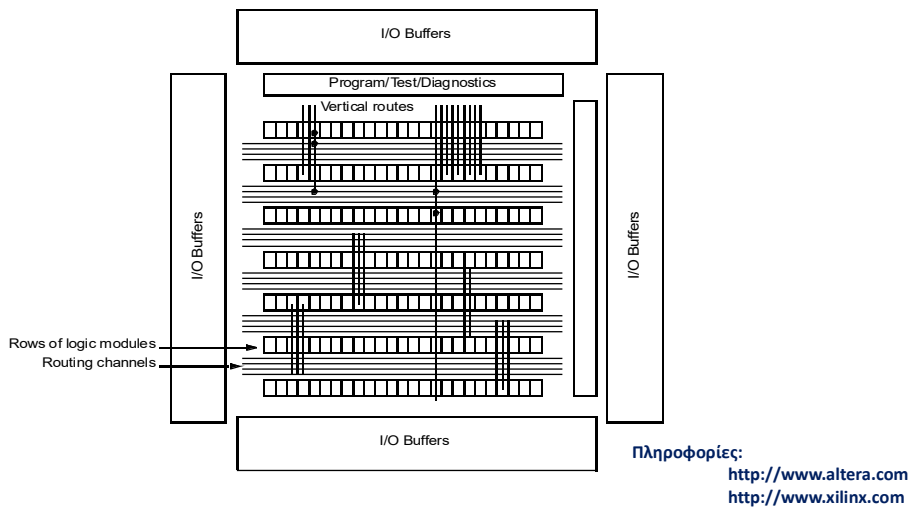


Σχεδιαστικές Μεθοδολογίες

33

33

## Γενική Χωροθέτηση FPGAs

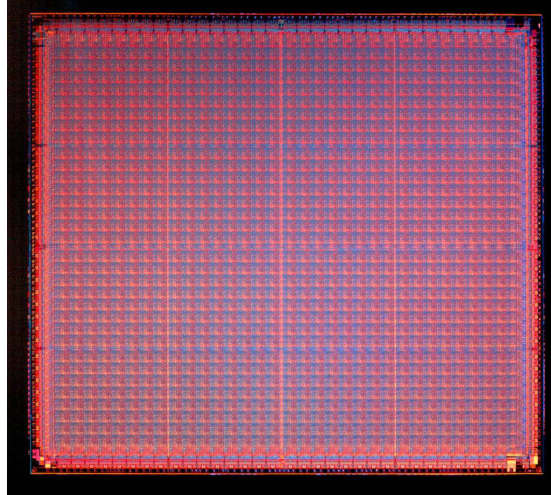


Σχεδιαστικές Μεθοδολογίες

34

34

## Ολοκληρωμένο FPGA



Xilinx XC400E

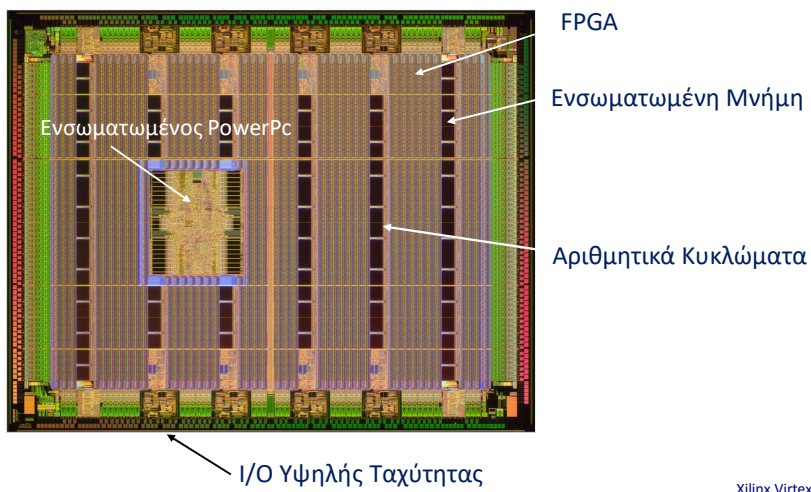


Σχεδιαστικές Μεθοδολογίες

35

35

## Ετερογενείς Προγραμματιζόμενες Πλατφόρμες



Xilinx Virtex-II Pro

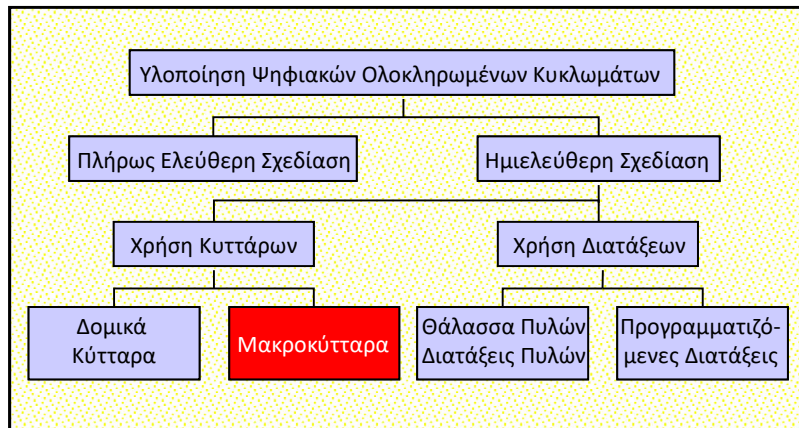


Σχεδιαστικές Μεθοδολογίες

36

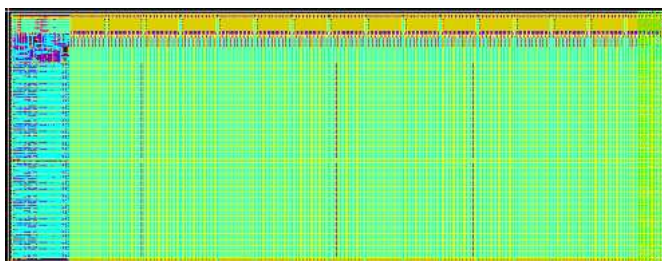
36

## Σχεδίαση με Μακροκύτταρα



37

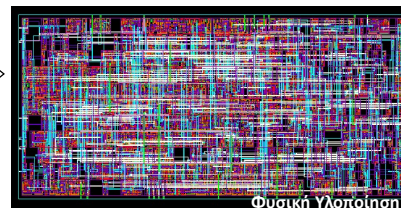
## Μακροκύτταρα



Άκαμπτο (Hard)  
Μακροκύτταρο  
Μνήμης

```
string mat = "booth";  
directive (multtype = mat);  
output signed [16] Z = A * B;  
HDL περιγραφή
```

Εύκαμπτο (Soft)  
Μακροκύτταρο Πολλαπλασιαστή



38

## Γλώσσες Περιγραφής Κυκλωμάτων

Hardware Description Languages (HDLs)

Οι γλώσσες HDL (VHDL, Verilog) χρησιμοποιούνται για την περιγραφή ηλεκτρονικών συστημάτων. Προσφέρουν τη δυνατότητα:

- καταγραφής προδιαγραφών
- τεκμηρίωσης
- επαλήθευσης με τη χρήση προσομοίωσης
- τυπικής επαλήθευσης
- σύνθεσης

Στόχος:

- αξιόπιστη διαδικασία σχεδίασης, με ταυτόχρονη ελαχιστοποίηση του κόστους και του απαιτούμενου χρόνου
- αποφυγή σχεδιαστικών λαθών

```
entity half_adder is
  port (in0, in1 : in bit;
        sum, cout : out bit);
end entity half_adder;

architecture rtl of half_adder is
begin
  sum <= in0 xor in1;
  cout <= in0 and in1;
end architecture rtl;

architecture str of half_adder is
begin
  ...
end architecture str;
```

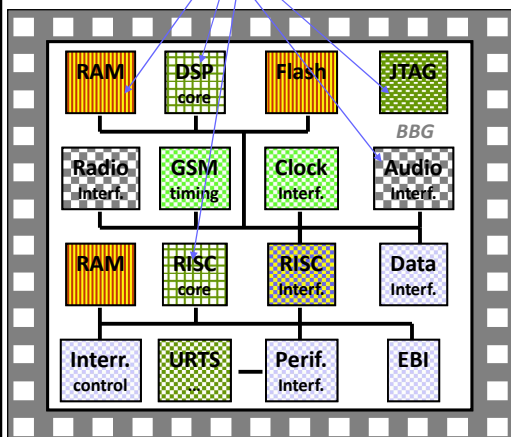
Παράδειγμα περιγραφής σε VHDL



## Συστήματα-σε-ένα-Ολοκληρωμένο

Systems-on-a-Chip (SoC)

Μακροκύτταρα ή πυρήνες (cores)

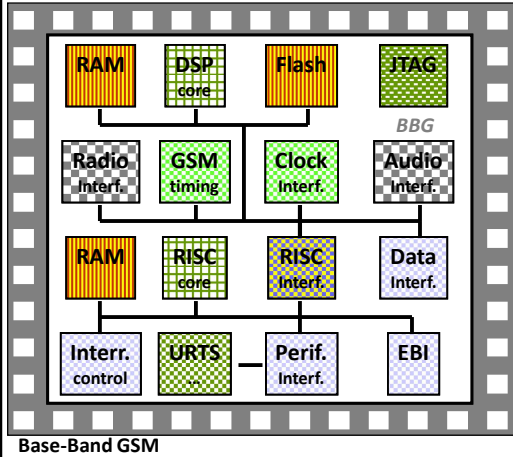


- Χρήση εμφωλευμένων κυκλωματικών πυρήνων όπου το κόστος, η απόδοση, η κατανάλωση και η αξιοπιστία έχουν τον πρώτο λόγο.
- Ανάγκη DSP κυκλωμάτων.
- Ύπαρξη μεικτών σχεδιασμών.
- Συνδυασμός προγραμματιζόμενων τεχνικών και τεχνικών σχεδίασης κυκλωμάτων ειδικών εφαρμογών.
- Το λογισμικό είναι αδιαχώριστος κρίκος.



Οι πυρήνες είναι συνήθως μονάδες πνευματικής ιδιοκτησίας (intellectual property – IP blocks)

## Θέματα Σχεδίασης SoC

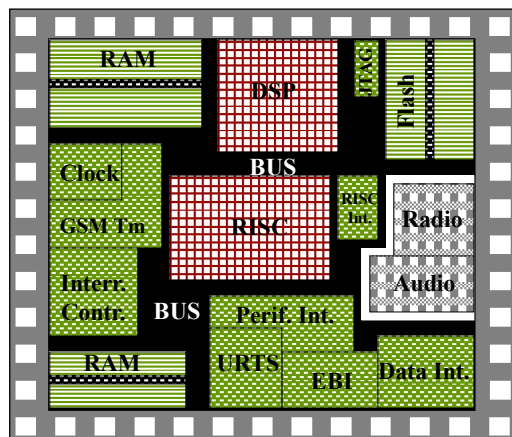


Base-Band GSM

- Διαχωρισμός υλικού/λογισμικού.
- Προδιαγραφές μερών.
- Σύγχρονη/ασύγχρονη σχεδίαση.
- Hard/Soft IPs.
- Χωροθέτηση (floorplanning).
- Διασύνδεση / αρτηρίες δεδομένων.
- Διαμοιρασμός ρολογιού και τροφοδοσιών.
- Τεχνικές ανάδειξης σχεδιαστικών λαθών (debug strategies).
- Τεχνικές επαλήθευσης σχεδιασμού (verification).
- Τεχνικές ελέγχου ορθής λειτουργίας (design for testability).

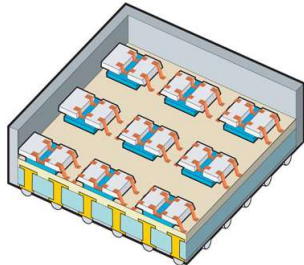


## Χωροθέτηση - Διασύνδεση

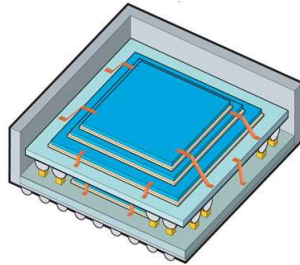


# Πακετάρισμα !

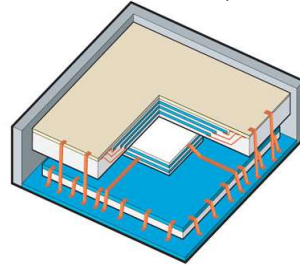
## System-in-a-Package (SiP)



Single Package SiP

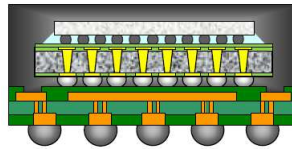


Package-on-Package (PoP)



Package-in-Package (PiP)

IEEE Spectrum 3/11



3D Chip Stacking

Σχεδιαστικές Μεθοδολογίες

43