

HY220

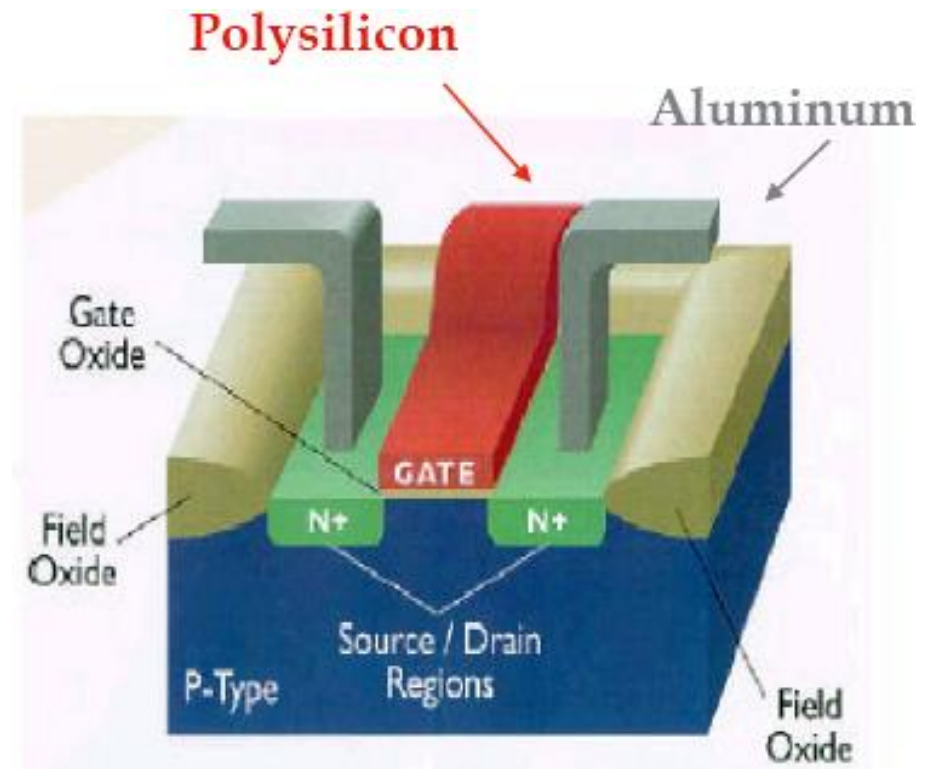
Εργαστήριο Ψηφιακών Κυκλωμάτων

Εαρινό Εξάμηνο
2023

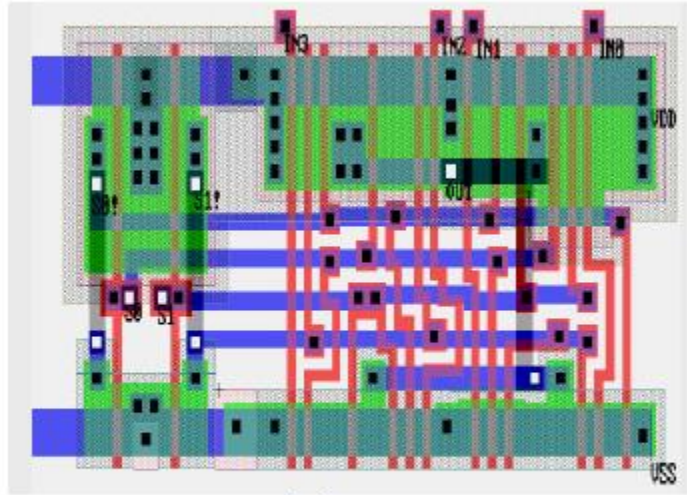
Ροή Σχεδίασης Κυκλωμάτων
και Εργαλεία CAD

Transistor: Δομική μονάδα κυκλωμάτων

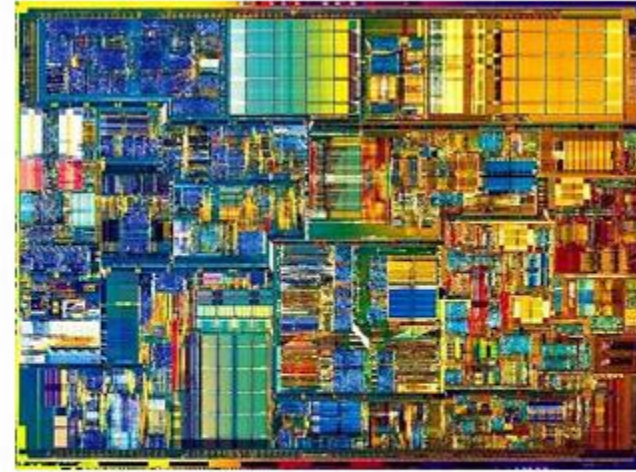
- Τα ολοκληρωμένα κυκλώματα υλοποιούνται κυρίως σε τεχνολογία CMOS (Complementary MOS)
 - Βάση της τεχνολογίας τα transistors τύπου MOSFET
 - metal oxide semiconductor field effect transistors
 - transistor επίδρασης πεδίου τύπου μέταλλο – οξειδίο – ημιαγωγός)
 - συμπεριφέρονται σαν διακόπτες



Chip = Γράφος Transistors



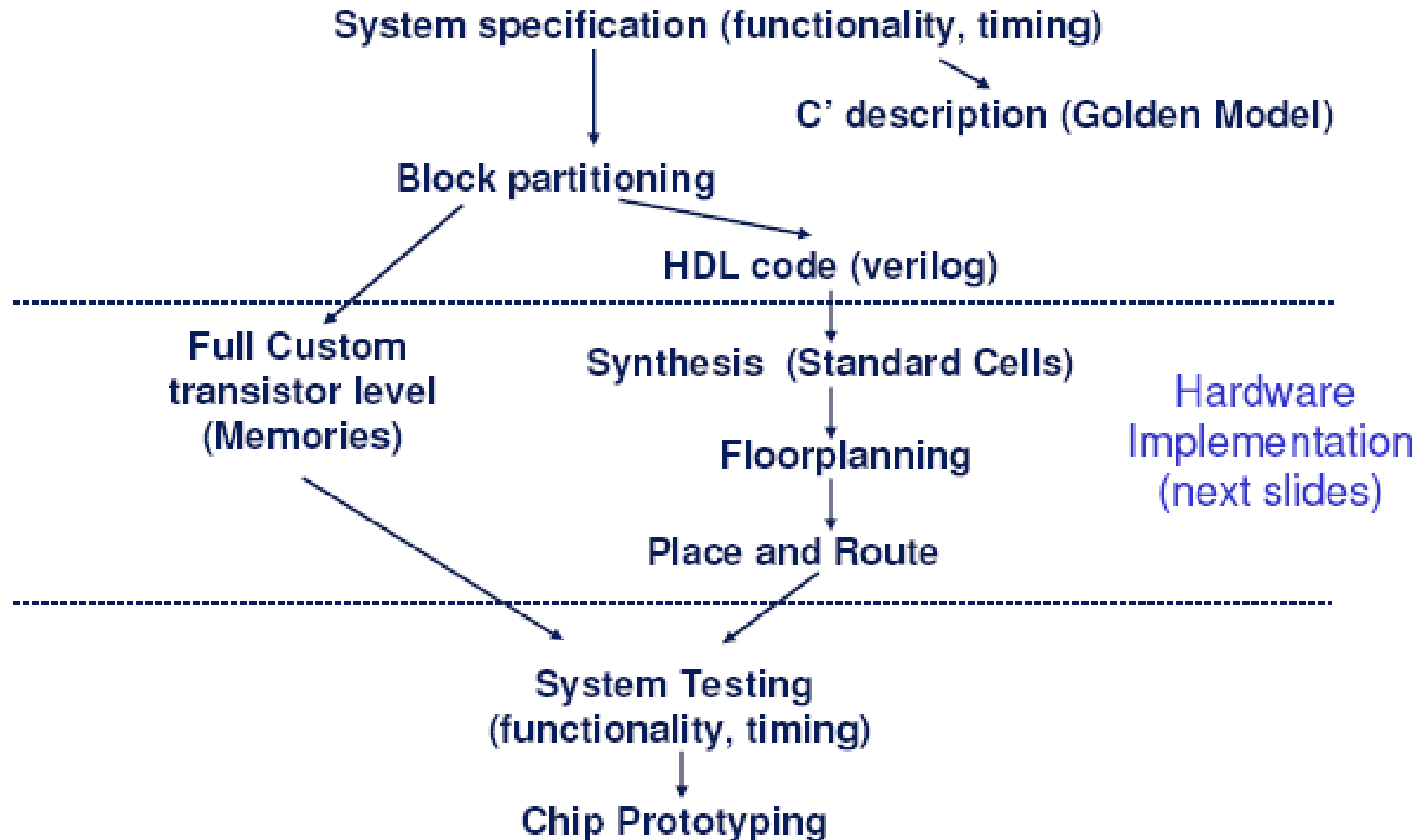
4:1 mux
10 πύλες



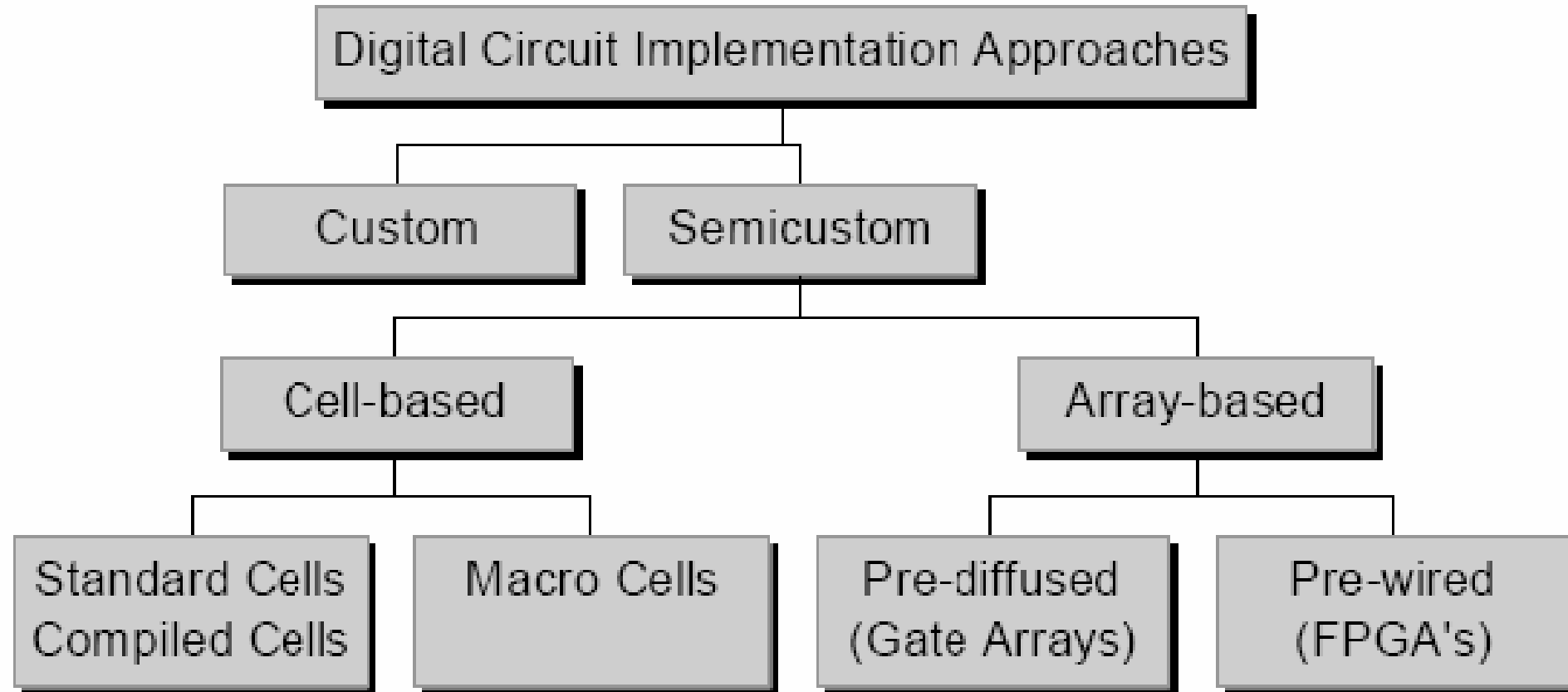
Intel processor
10 000 000 πύλες

- Η τεχνολογία (process) που χρησιμοποιούμε για την κατασκευή (fabrication) καθορίζει τις παραμέτρους των transistors και το μέγεθός τους
- Καθώς βελτιώνεται η τεχνολογία:
 - Μικραίνει το μέγεθος των transistors (περισσότερα transistors στον ίδιο χώρο)
 - Αυξάνεται η ταχύτητά τους
 - Ελαττώνεται η κατανάλωση ενέργειάς τους

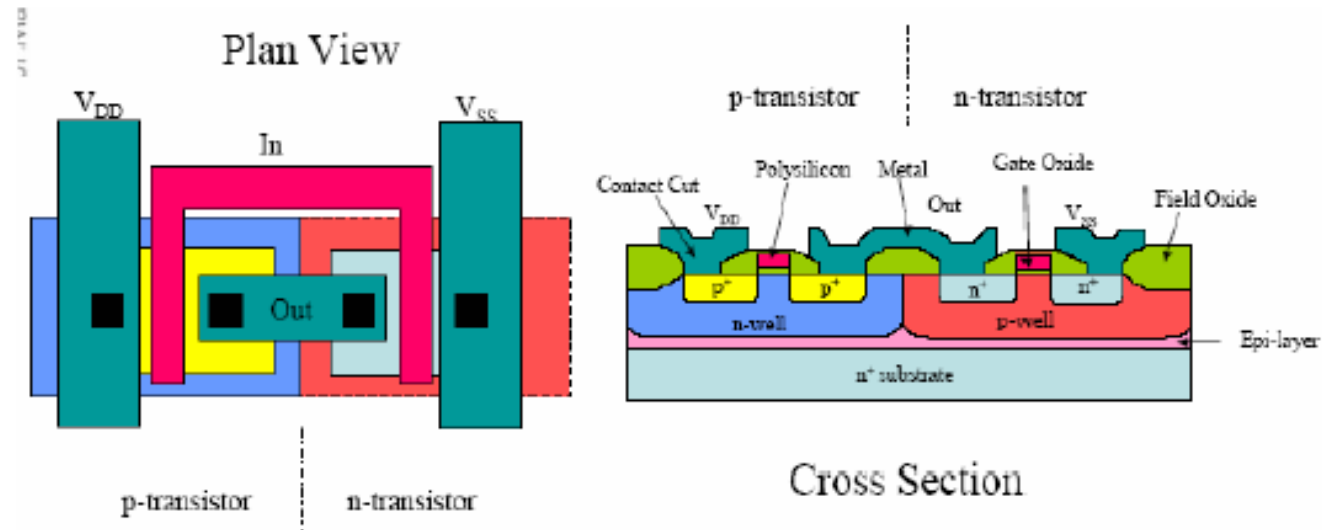
Ροή Σχεδίασης – Design Flow



Μέθοδοι Σχεδίασης



Full Custom

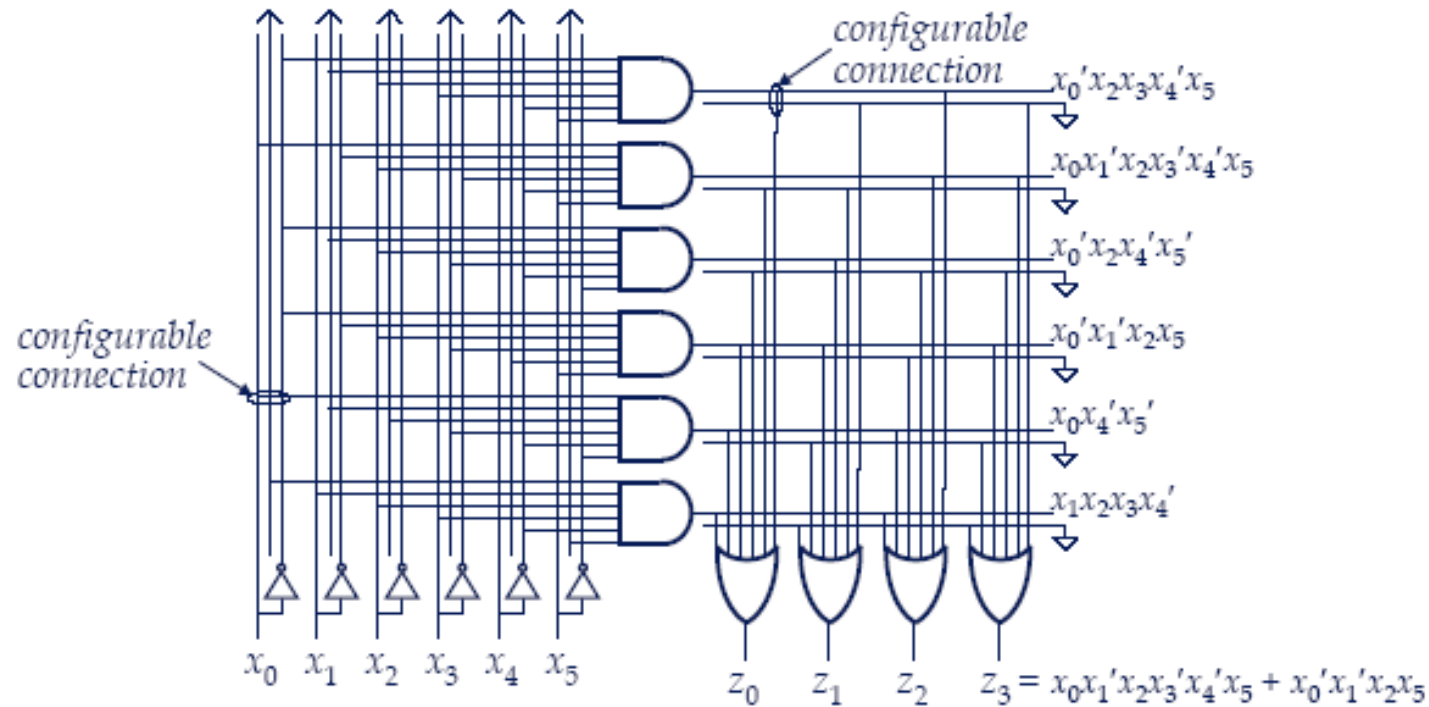


- Η διάταξη (layout) των transistors είναι χειροποίητη χρησιμοποιώντας VLSI editors. Χρήσιμο κυρίως σε μικρά designs λόγω απαιτήσεων σε χρόνο
 - Μέγιστη ελευθερία
 - Μπλοκ υψηλών επιδόσεων
 - Αργή διαδικασία

Semi-Custom: Array-Based (Gate Array)

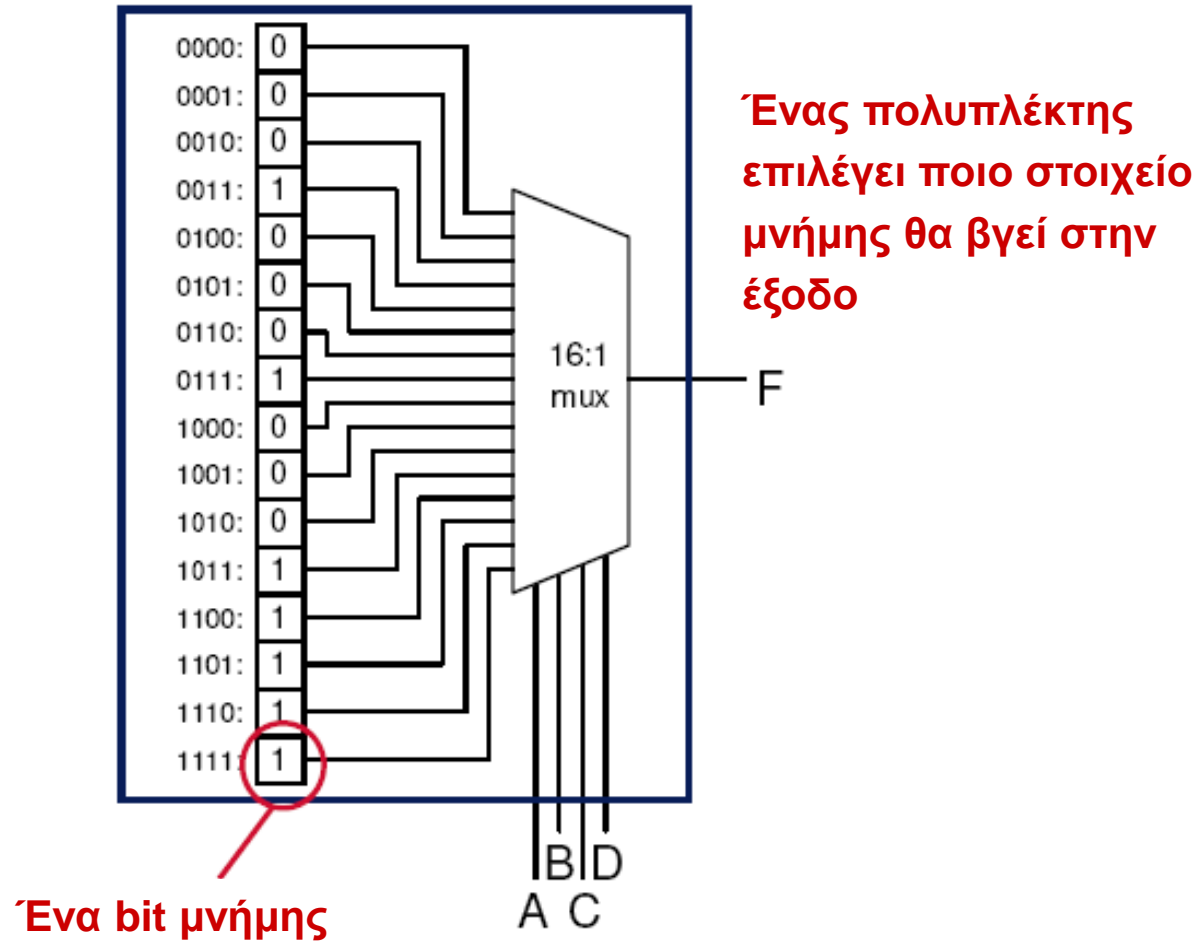
- Μεγάλοι πίνακες από transistors δίνονται από τους κατασκευαστές των chips.
- Διασυνδέοντας αυτά τα transistor με τον κατάλληλο τρόπο προκύπτει η επιθυμητή λογική

Semi-custom: Programmable Logic Array (PLA)

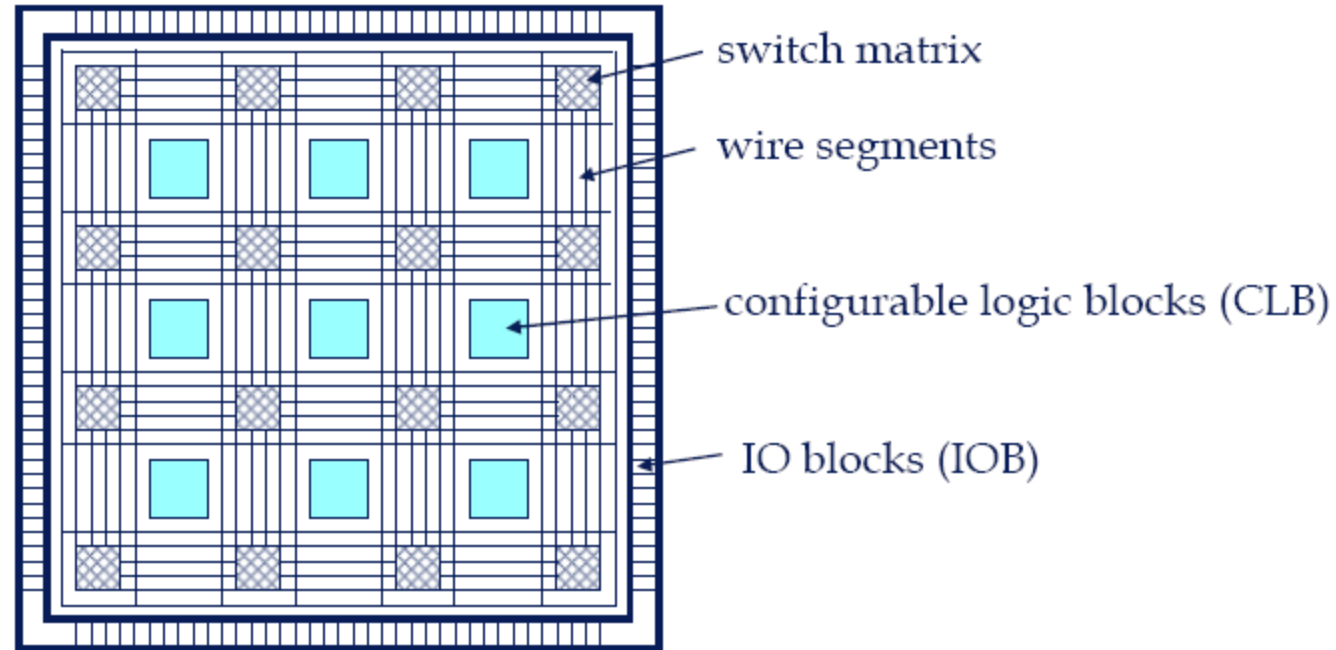


- Οι PLAs έχουν προγραμματιζόμενα AND και OR επίπεδα (planes).
- Μπορούν να υλοποιήσουν οποιαδήποτε 2-επίπεδη λογική AND-OR
- Αποδοτική φυσική υλοποίηση σε τεχνολογία CMOS.

Προγραμματιζόμενη Λογική: LUTs (Lookup Tables)

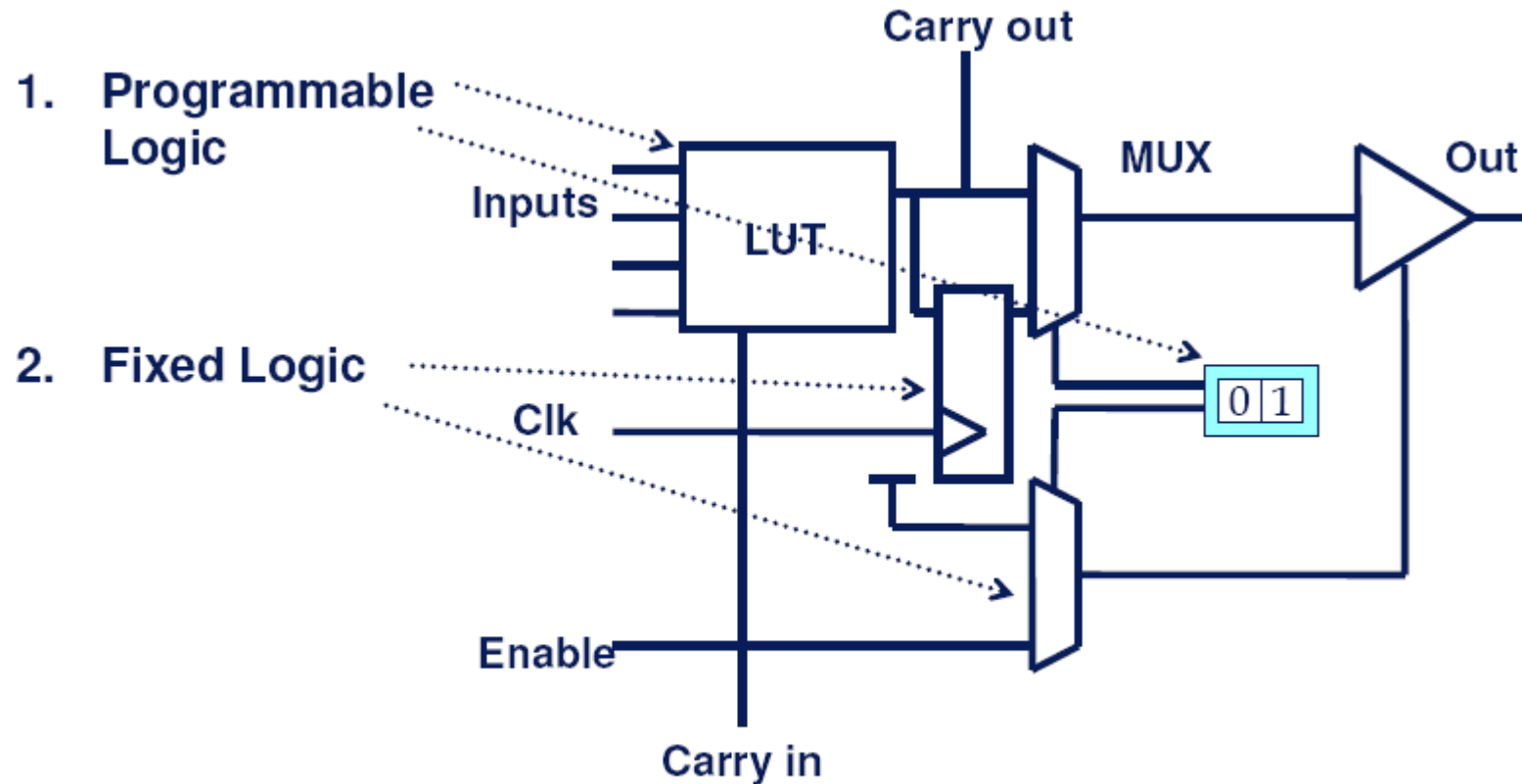


FPGA: Field Programmable Gate Array



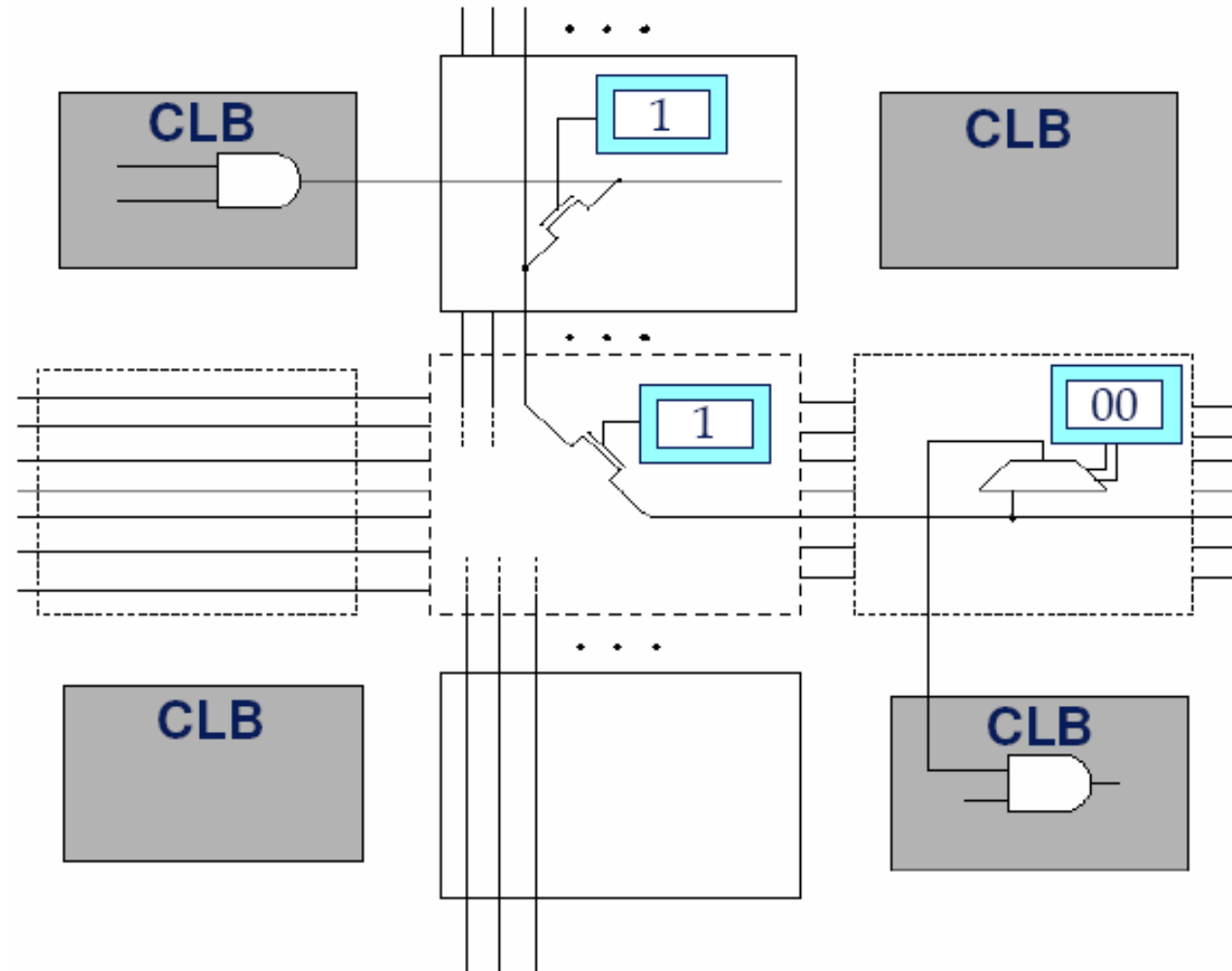
- Τα CLBs συνδέονται στα κοντινά καλώδια
- Τα καλώδια συνδέονται μεταξύ τους μέσω του switch matrix
- Υπάρχουν και μακρινά καλώδια για να διασυνδέουν απομακρυσμένα CLBs
- Οι πληροφορίες για την προγραμματιζόμενη λογική είναι αποθηκευμένες σε bit μνήμης τα οποία φορτώνονται κατά τον αρχικό προγραμματισμό της.

Παράδειγμα ενός CLB (Logic Element)

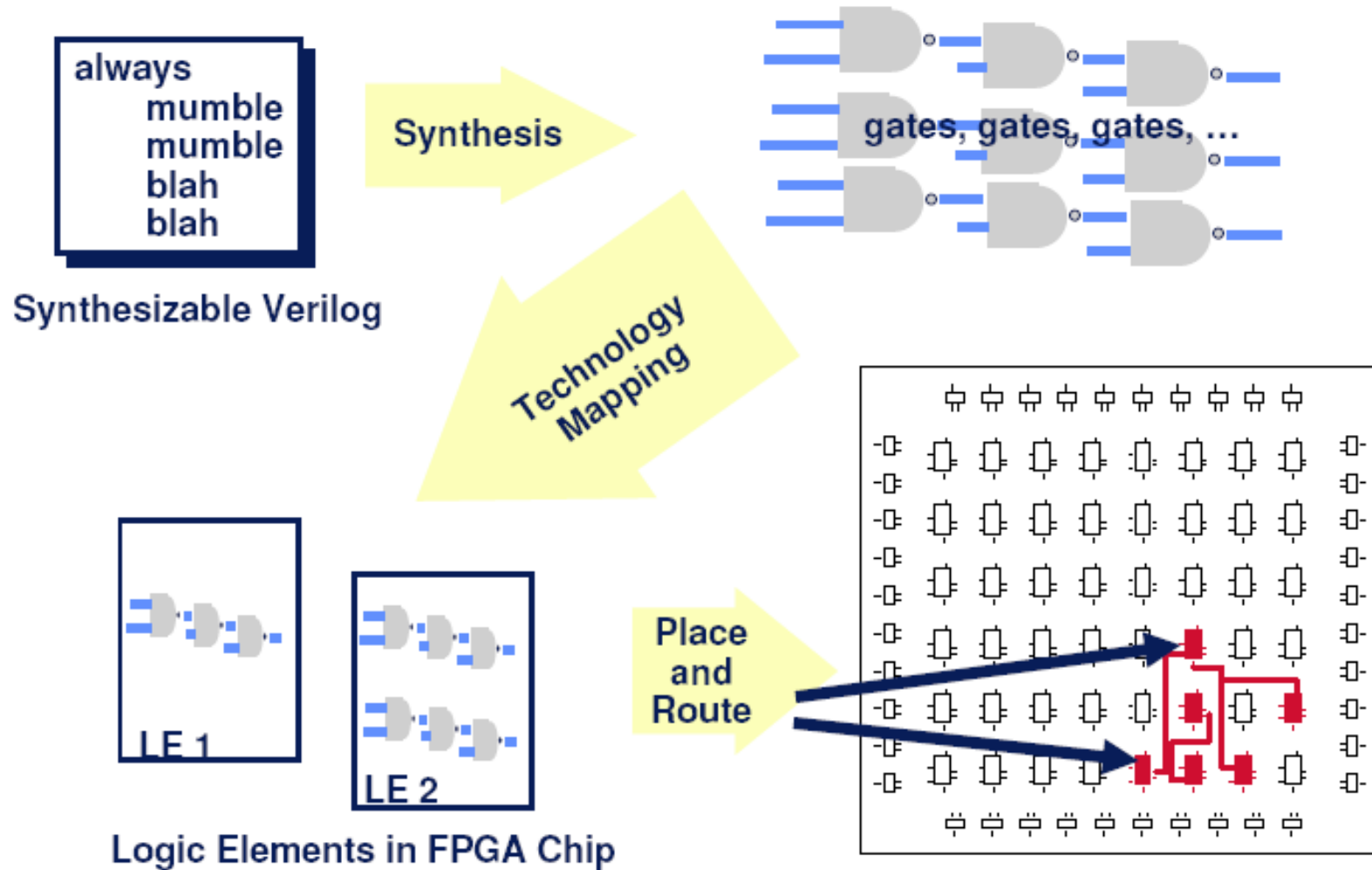


LE example

Η διασύνδεση στην FPGA

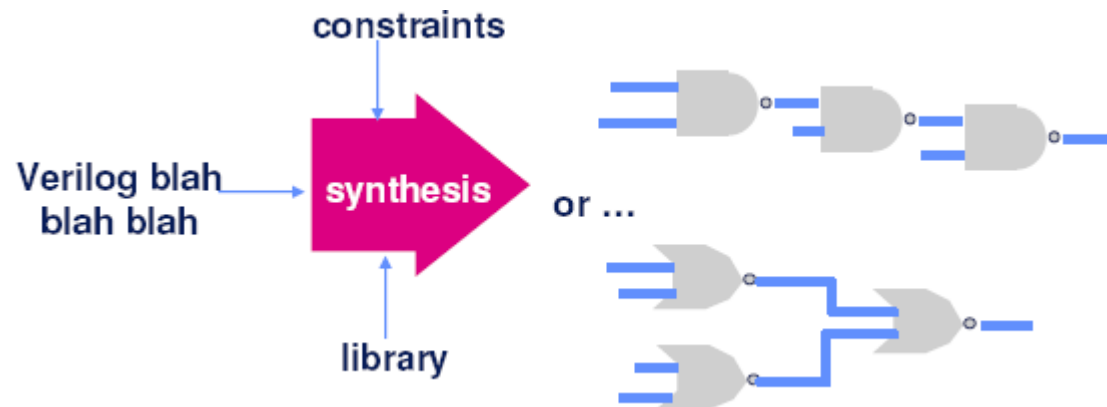


Η ροή σχεδίασης για FPGA



Η σύνθεση (synthesis)

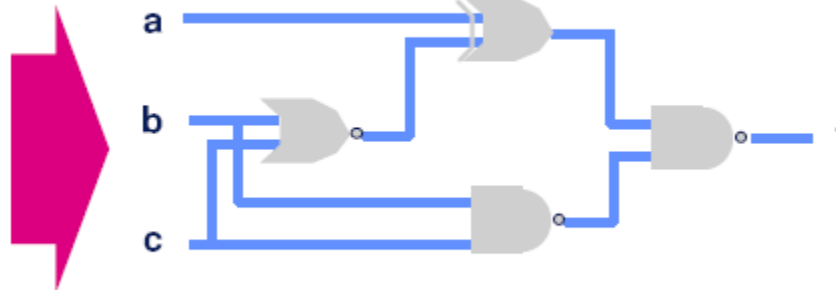
- Λογική σύνθεση (logic synthesis)
 - Ένα εργαλείο/πρόγραμμα σχεδιάζει κυκλώματα από «αφηρημένες» περιγραφές της λογικής
 - Δέχεται περιορισμούς (constraints) για το μέγεθος, την ταχύτητα κ.α
 - Χρησιμοποιεί βιβλιοθήκες (π.χ. 3-input gates)
- Πως;
 - Περιγράφουμε σε Verilog «αφηρημένα» τη λογική
 - Το εργαλείο μας παράγει εναλλακτικές υλοποιήσεις



Απλό παράδειγμα σύνθεσης

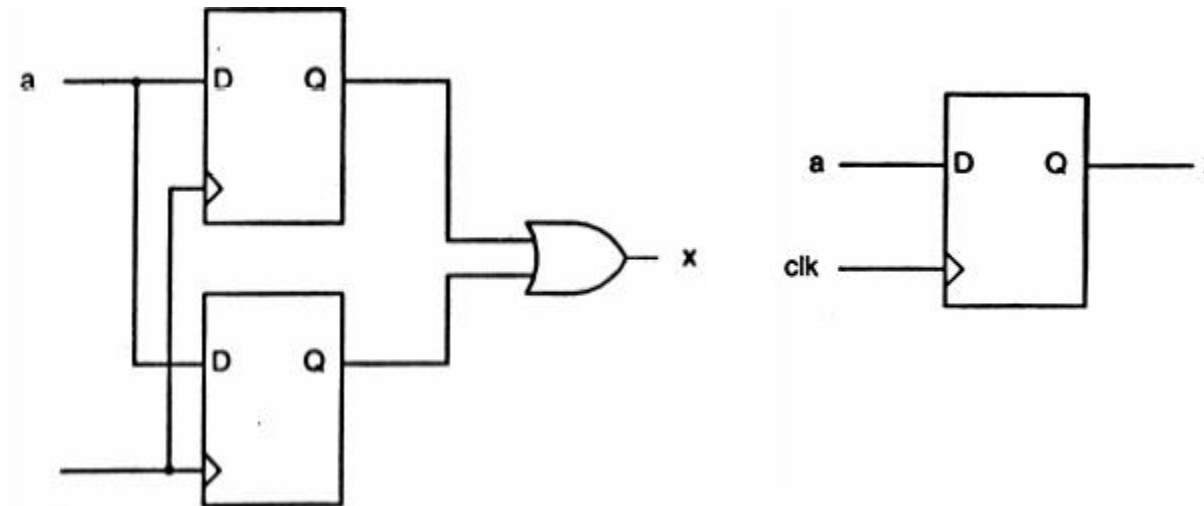
- Τι συμβαίνει;
 - Γράφουμε τον κώδικα και η σύνθεση μας βγάζει τις πύλες
 - Μπορεί να χρησιμοποιεί διαφορετικές βιβλιοθήκες από τον σχεδιαστή (εδώ μόνο πύλες 2 εισόδων)
 - Μια περιγραφή είναι αρκετή για να μας παραχθούν αρκετές διαφορετικές υλοποιήσεις!!!
 - ... αλλά αυτό προϋποθέτει ότι ξέρουμε την υλοποίηση σε πύλες - το οποίο καταλήγει να μην είναι «αφηρημένη» περιγραφή βέβαια!!!

```
module gate (f, a, b, c);  
  output f;  
  input a, b, c;  
  
  and A (a1, a, b, c),  
       B (a2, a, ~b, ~c),  
       C (a3, ~a, o1);  
  or D (o1, b, c),  
     E (f, a1, a2, a3);  
endmodule
```



Αυτοματοποιημένη Λογική Σύνθεση

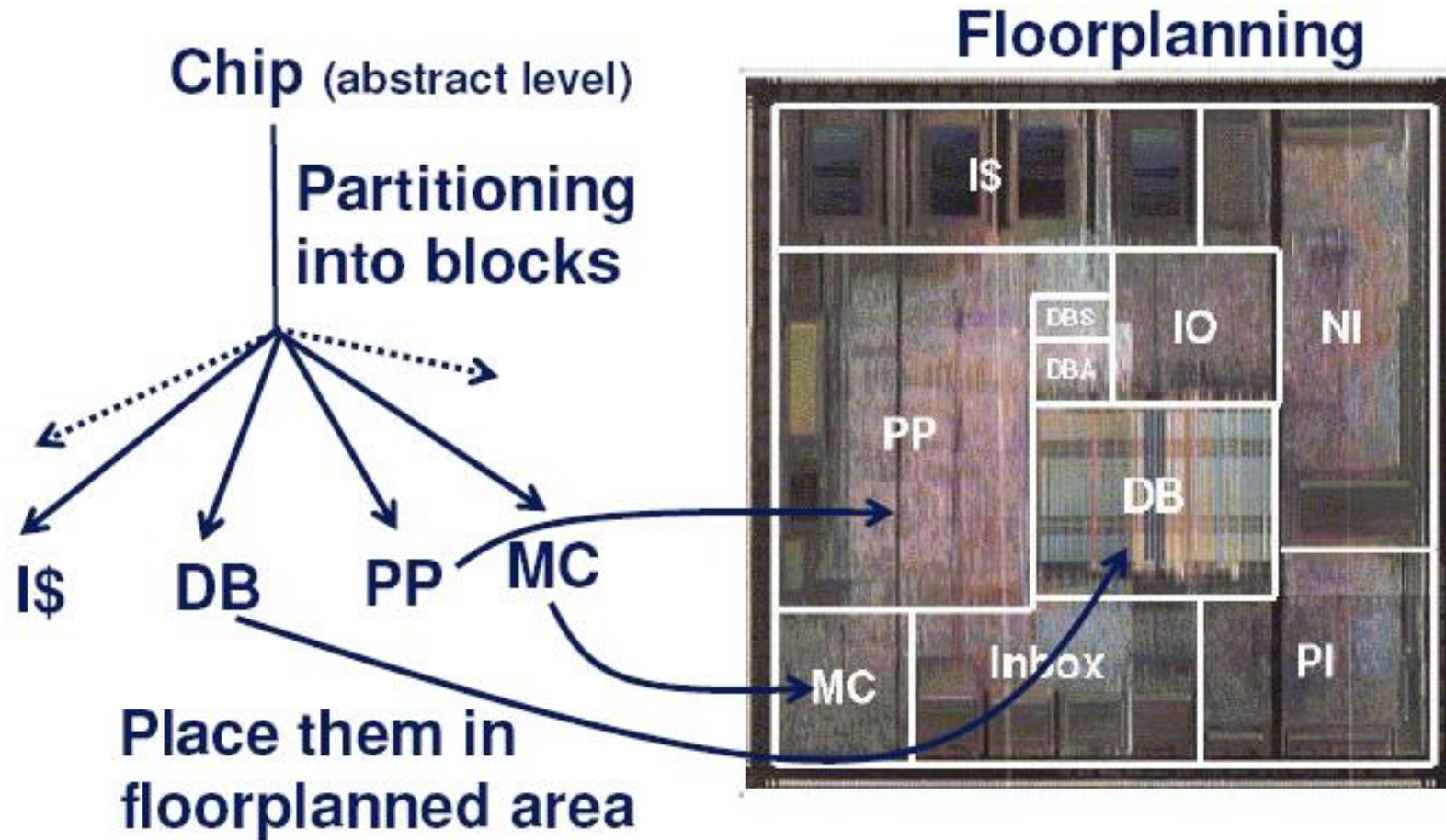
- Η σύνθεση πολύ συχνά ερμηνεύει τον κώδικα **διαφορετικά από την προσομοίωση !!!**
- Η περιττή λογική μπορεί να μην ανιχνεύεται πάντα!
 - Τα παρακάτω κυκλώματα είναι λειτουργικά ισοδύναμα!



Mapping – Place & Route

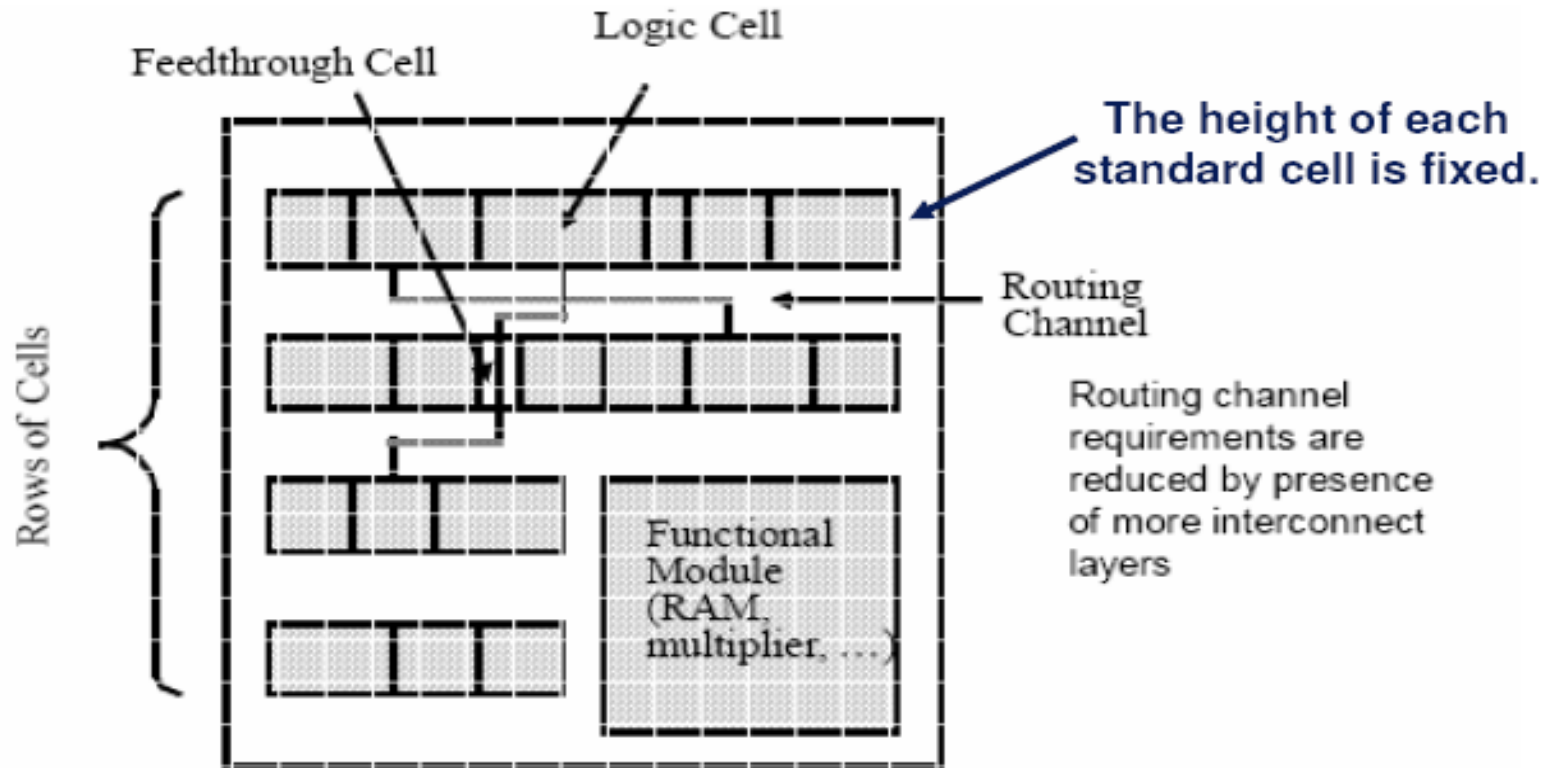
- **Mapping:** Μεταφορά της λογικής που παράγεται από την σύνθεση στα λογικά στοιχεία (LE-CLBs-Cells) που παρέχει φυσικά η εκάστοτε τεχνολογία (FPGA-ASIC)
 - Κατά το mapping η λογική μετατρέπεται κατάλληλα σε αυτήν που υποστηρίζει η τεχνολογία.
- **Place & Route:** Χωροθέτηση και Διασύνδεση των κυκλωμάτων στην φυσική τοπολογία.
 - Τοποθετεί την λογική στα CLBs της FPGA και τα διασυνδέει με τέτοιο τρόπο ώστε να καλύπτονται οι χρονικοί περιορισμοί
 - Η καθυστέρηση των καλωδίων (wiring delay), για την διασύνδεση μεταξύ των blocks της λογικής, πρέπει να είναι αποδεκτή
 - Τοποθετεί τα κρίσιμα κομμάτια του κυκλώματος κοντά για να μειώσει όσο το δυνατόν την καθυστέρηση από τα καλώδια
 - Η καθυστέρηση των σημάτων στο κύκλωμα εξαρτάται σημαντικά από την καθυστέρηση της διασύνδεσης (routing delay)

Partitioning and Floorplanning Κατάτμηση και Κάτοψη Σχεδίου



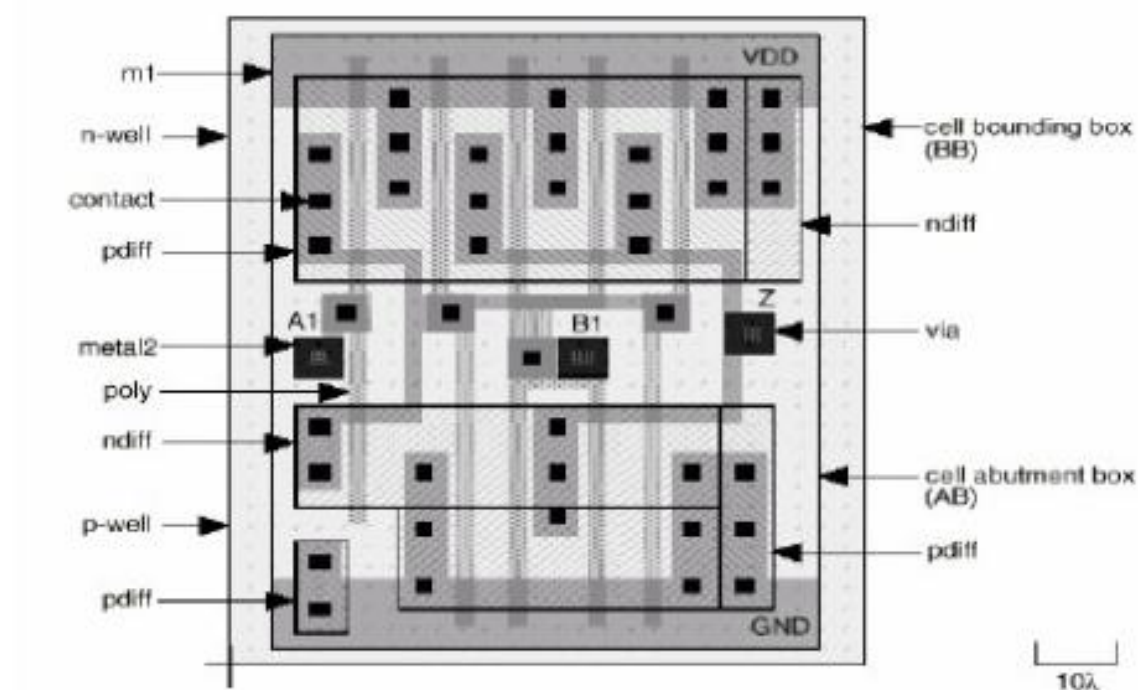
Cell-Based Design

- Semi-custom: λύση βασισμένη σε εργαλεία που χρησιμοποιούν standard cells που παρέχει η τεχνολογία και είναι ομοιόμορφα

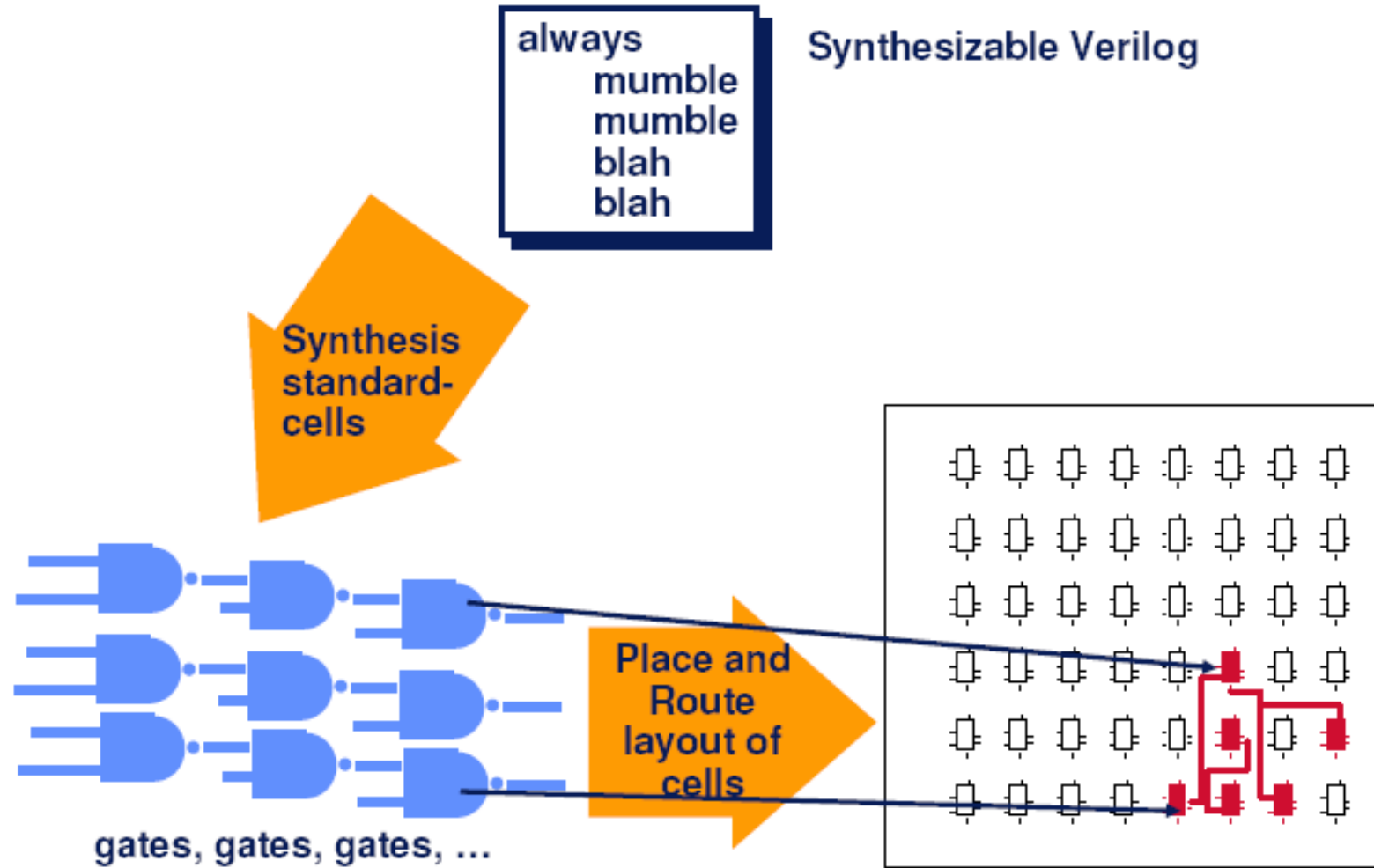


Διάταξη ενός standard cell

- Η διάταξη (layout) ενός standard cell από μια βιβλιοθήκη standard cell. Είναι full-custom!

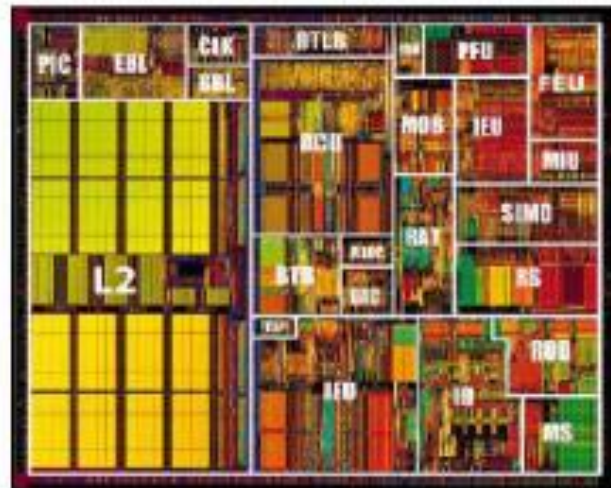


Cell-based Flow

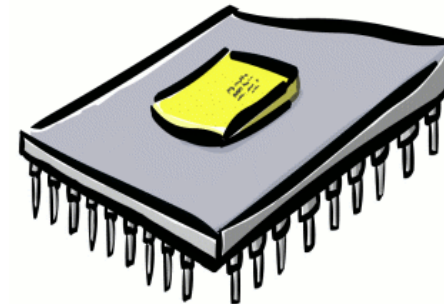


ASIC: Application Specific Integrated Circuit

- Chip για μια συγκεκριμένη εφαρμογή. Δεν μπορεί να ξαναπρογραμματιστεί η λογική του για να καλύψει άλλες λύσεις
- Δίνουμε στον κατασκευαστή την διάταξη με full-custom ή semi-custom blocks.



Fabricate



Κόστη Υλοποίησης

- Κόστη:

- Unit cost (κόστος μονάδας): το χρηματικό κόστος για την κατασκευή κάθε αντιγράφου του συστήματος
- NRE cost (Non-Recurring Engineering cost – Μη επαναλαμβανόμενο κόστος σχεδίασης): Το εφάπαξ κόστος για την σχεδίαση του συστήματος
- Total-Cost = NRE-Cost + Unit-Cost * #Units
- Per-Product-Cost = Total-Cost / # Units = (NRE-Cost / # Units) + Unit-Cost

- Παράδειγμα:

- NRE-cost = \$2000, Unit-cost = \$100
- Για 10 μονάδες:
 - Total = 2000 + 10*100 = \$3000
 - Per-Product-Cost = (2000/ 10) + 100 = \$300
 - Για να καλυφθεί το NRE κόστος **επιβαρύνθηκε το κόστος της κάθε μονάδας με \$200!!!**

FPGA vs. ASIC (Pros and Cons)

- FPGA (gate-array)
 - Χαμηλό αρχικό κόστος
 - Χαμηλό χρηματικό ρίσκο
 - Γρήγορες κατασκευαστικές αλλαγές (reprogram)
 - Εύκολες αλλαγές στη σχεδίαση
 - Επαναπρογραμματιζόμενη
 - Αργό Ρολόι
 - Μικρή χωρητικότητα σε πύλες
- ASIC (full-custom, semi-custom)
 - Γρήγορο ρολόι
 - Μεγάλη χωρητικότητα σε πύλες
 - «Πυκνό» σχέδιο
 - Υψηλό κόστος
 - Αργές κατασκευαστικές αλλαγές
 - Μεγάλος χρόνος κατασκευής

Εργαλεία CAD: Computer Aided Design

