

# Διάχυτα και Ενσωματωμένα Συστήματα

Πρόγραμμα Μεταπτυχιακών Σπουδών  
“Επιστήμη και Τεχνολογία  
της Πληροφορικής και των Υπολογιστών”

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών  
Πανεπιστήμιο Δυτικής Αττικής

Ακαδημαϊκό έτος 2018-2019  
Γιάννης Βογιατζής - Παναγιώτης Καρκαζής

Παρουσίαση 1<sup>η</sup>: Εισαγωγή

# Επικοινωνία

Γιάννης Βογιατζής

[voyageri@teiath.gr](mailto:voyageri@teiath.gr)

Γραφείο 207 (2<sup>ος</sup> όροφος)

Παναγιώτης Καρκαζής

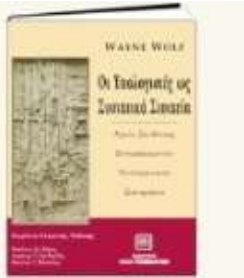
[pkarkazis@gmail.com](mailto:pkarkazis@gmail.com)

Γραφείο 203 (2<sup>ος</sup> όροφος)

# Επισκόπηση

- Εισαγωγή στα ενσωματωμένα συστήματα
  - Τεχνολογίες υλοποίησης (ASIC-layout, FPGA κλπ)
- Επεξεργαστές ειδικού σκοπού
  - Θέματα αρχιτεκτονικής
- Γλώσσες περιγραφής υλικού
  - VHDL, Verilog
- Υλοποίηση σε FPGA
- Επεξεργαστές γενικού σκοπού
- Μικροελεγκτές PIC

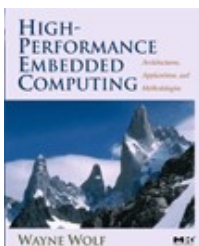
# Βιβλιογραφία



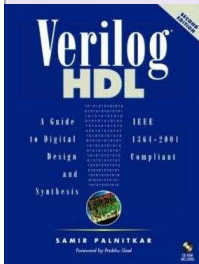
**Οι Υπολογιστές ως Συστατικά Στοιχεία,**  
Wayne Wolf



**Σχεδιασμός Ψηφιακών Συστημάτων σε FPGAs,**  
Wayne Wolf



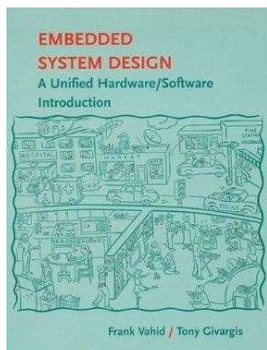
**High Performance Embedded Computing**  
Wayne Wolf



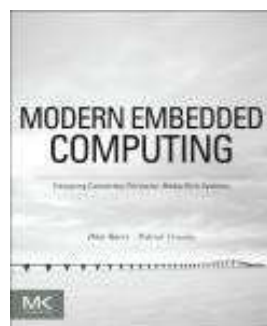
**Verilog HDL,** Samir Palnitkar, Second Edition  
(2003), Prentice Hall, ISBN 978-  
0132599702

Σημειώσεις / διαφάνειες στο eclass

# Βιβλιογραφία



**Embedded System Design: A Unified Hardware/Software Introduction**, Frank Vahid, Tony D. Givargis, Wiley (2001), ISBN 978-0471386780



**Modern Embedded Computing**, Peter Barry and Patrick Crowley, (2012), Morgan Kaufmann, ISBN 978-0123914903



**Embedded Systems Design with Platform FPGAs**, Ronald Sass and Andrew Schmidt, (2010), Morgan Kaufmann, ISBN 978-0123743336

# Ώρες Μαθήματος

Πέμπτη 19:00 – 21:15

# Τελική Αξιολόγηση - Βαθμολόγηση

## Τελικός Βαθμός

Βαθμός στην τελική εξέταση (100%)

Δυνατότητα εκπόνησης βοηθητικής εργασίας

# Ενσωματωμένα Συστήματα

Συνήθως με τον όρο Υπολογιστικό Σύστημα αναφερόμαστε σε

- PC
- Laptop
- Server
- Mainframe

Στην πραγματικότητα υπάρχει ένας άλλος τύπος υπολογιστικών συστημάτων, πολύ πιο διαδεδομένος . . .

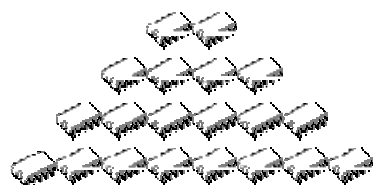
# Ενσωματωμένα Συστήματα

Υπολογιστές υπάρχουν  
εδώ...



Και εδώ ...

Ακόμη και εδώ...



## Ενσωματωμένα Συστήματα

Υπολογιστικά συστήματα ενσωματωμένα  
σε ηλεκτρονικές συσκευές

Δύσκολο να ορισθούν επακριβώς.

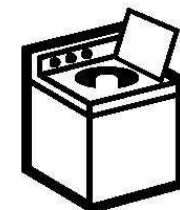
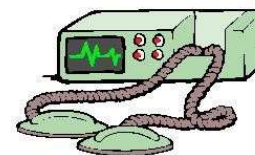
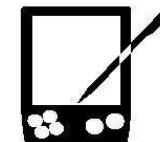
Δυνητικά κάθε υπολογιστικό σύστημα  
εκτός του “κλασικού” υπολογιστή

Δισεκατομμύρια μονάδων παράγονται  
ετησίως σε σύγκριση με εκατομμύρια  
μονάδες desktop υπολογιστών

# Ενδεικτική λίστα...

Anti-lock brakes  
Auto-focus cameras  
Automatic teller machines  
Automatic toll systems  
Automatic transmission  
Avionic systems  
Battery chargers  
Camcorders  
Cell phones  
Cell-phone base stations  
Cordless phones  
Cruise control  
Curbside check-in systems  
Digital cameras  
Disk drives  
Electronic card readers  
Electronic instruments  
Electronic toys/games  
Factory control  
Fax machines  
Fingerprint identifiers  
Home security systems  
Life-support systems  
Medical testing systems

Modems  
MPEG decoders  
Network cards  
Network switches/routers  
On-board navigation  
Pagers  
Photocopiers  
Point-of-sale systems  
Portable video games  
Printers  
Satellite phones  
Scanners  
Smart ovens/dishwashers  
Speech recognizers  
Stereo systems  
Teleconferencing systems  
Televisions  
Temperature controllers  
Theft tracking systems  
TV set-top boxes  
VCR's, DVD players  
Video game consoles  
Video phones  
Washers and dryers

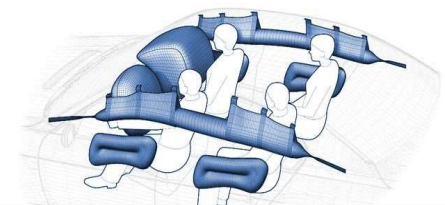


# Ενσωματωμένοι υπολογιστές



# Ενσωματωμένοι υπολογιστές - αυτοκινητοβιομηχανία

- Περισσότεροι από 70 ενσωματωμένοι επεξεργαστές ελέγχουν:
  - Κινητήρα (υψηλή απόδοση, χαμηλή κατανάλωση)
  - Συστήματα άνεσης και ευκολίας
  - Συστήματα ασφαλείας
    - Anti-locking Braking System (ABS)
    - Dynamic Stability Control (DSC) system
    - Anti-Slip Control (ASC) system
    - Ανάπτυξη των αερόσακων
    - Περιορισμός των ζωνών ασφαλείας
    - Αισθητήρες πρόσκρουσης
    - Αισθητήρες πίεσης των ελαστικών
  - Συστήματα επικοινωνιών



# Ενσωματωμένοι υπολογιστές στην αυτοκινητοβιομηχανία

IEEE Spectrum 2009:

- The cost of electronics as a percent of vehicle costs climbed from around **5% in the late 1970s to 15% by 2005** (excluding final assembly costs).
- For **hybrids**, where the amount of software needed for engine control alone is nearly twice as great as that for a standard car, the cost of electronics as a percent of vehicle costs is closer to 45%.
- Within 10 years, some experts predict that **the cost of electronics are expected to rise to 50%** for conventional vehicles and **80% for hybrids**.
- For today's premium cars, the cost of **software and electronics** can reach 35 - 40% of the cost of a car, with **software development** contributing about 13 - 15% of that cost.
- If it costs US \$10 a line for developed software (which is low) for a premium car, its software alone represents about a billion dollars' worth of investment.

# ΕΣ και IoT

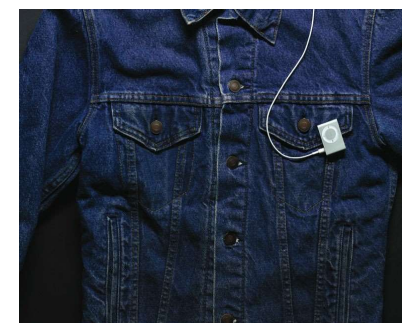
- Embedded systems is a subset of IoT.
- IoT is simply "A network of Internet connected objects able to collect and exchange data."
- The word "Internet of Things" has two main parts;
  - Internet being the backbone of connectivity, and
  - Things meaning objects / devices .
- Simply put, you have "things"
  - (things are nothing but your embedded system devices)
  - that sense and collect data
  - and send it to the internet.
  - This data can be accessible by other "things" too.
- Finally,
$$\text{IoT} = \text{ES} + \text{NT} + \text{IT}$$
- At the simplest level, IoT is a combination of Embedded Systems (ES), Network Technology (NT) and Information Technology (IT).

# Ενσωματωμένοι υπολογιστές

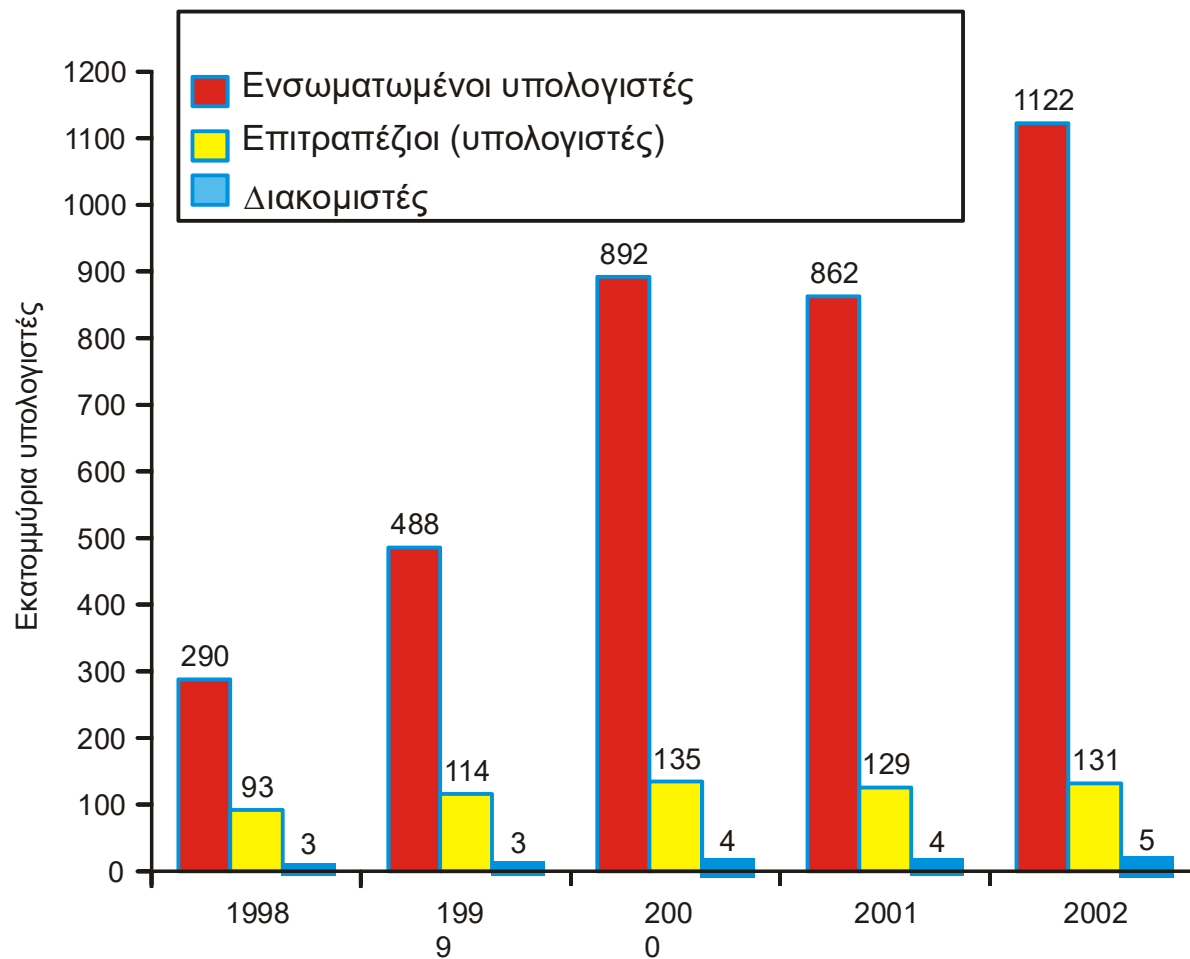
- Το πιο ραγδαία αναπτυσσόμενο κομμάτι της αγοράς των υπολογιστών
- Μέρος ενός μεγαλύτερου συστήματος
  - Σε προϊόντα ευρείας κατανάλωσης
  - Τηλεπικοινωνίες,
  - Αυτοκινητοβιομηχανία,
  - Αεροδιαστημική, Ιατρικές εφαρμογές
- Περιλαμβάνει:
  - Ενσωματωμένους επεξεργαστές /Μικροελεγκτές
  - Μνήμη (Flash) / Σκληρούς δίσκους
  - Συσκευές εισόδου-εξόδου ειδικού σκοπού
  - Αναλογικά συστήματα, αισθητήρες, κλπ.
- Μεγάλο εύρος επεξεργαστικής ισχύος/κόστους
  - 8-bit, 16-bit, 32-bit CPUs
  - CPUs για
  - video games,
  - network switches κλπ.

Embedded  
computers

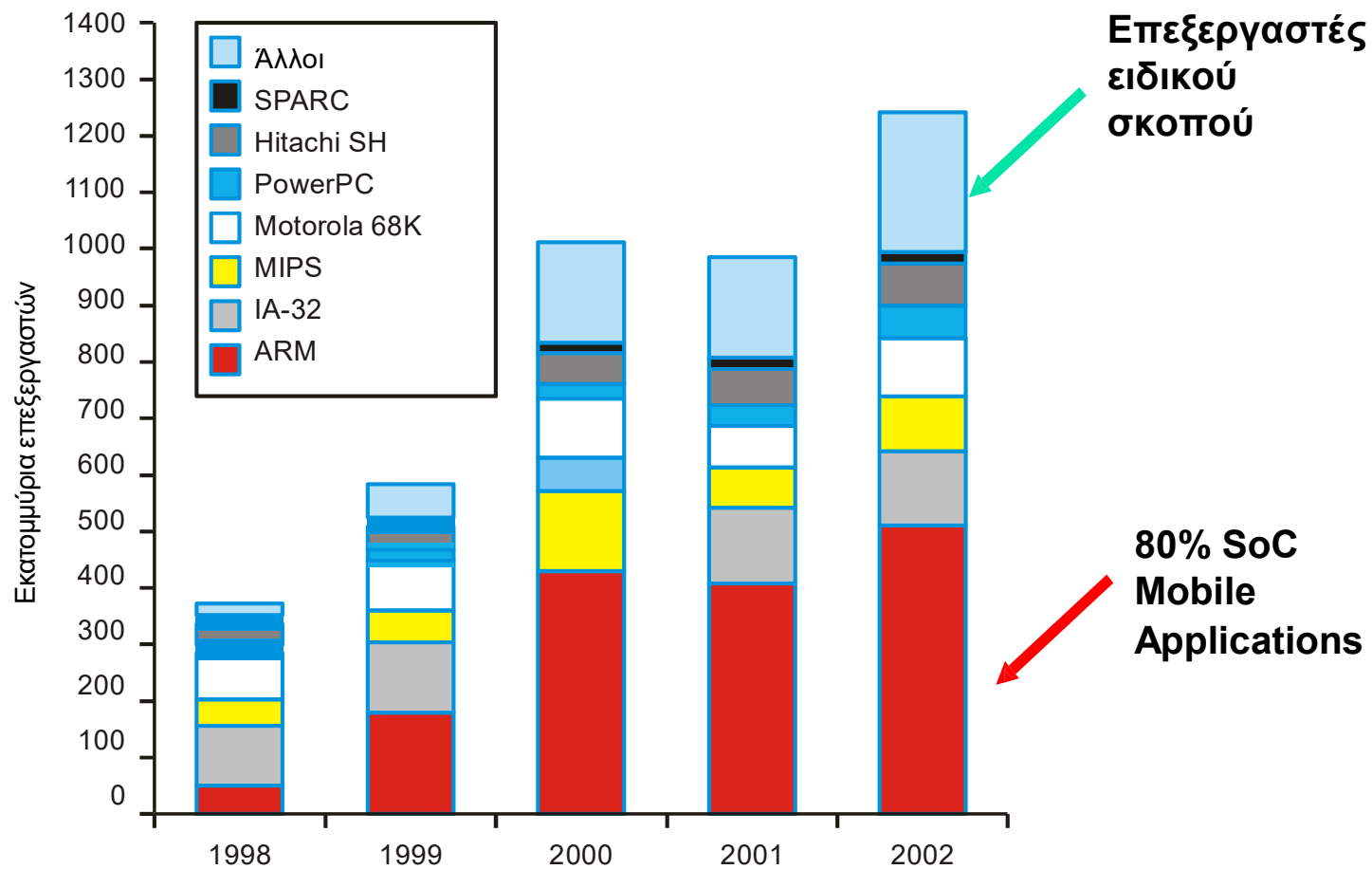
**ΒΡΙΣΚΟΝΤΑΙ ΠΑΝΤΟΥ**



# Αγορά υπολογιστών



# Πωλήσεις επεξεργαστών



# Ενσωματωμένα Συστήματα

Επισκόπηση

Τί είναι;

Σχεδιαστικές προκλήσεις - βελτιστοποίηση

Τεχνολογίες σχεδιασμού

Επεξεργαστή

Ολοκληρωμένων κυκλωμάτων

# Τι είναι Ενσωματωμένο Σύστημα

- Ένα υπολογιστικό σύστημα με περιορισμούς
  - συγκεκριμένη εφαρμογή
  - φυσική μορφή
  - ισχύς
  - πόροι και χαρακτηριστικά συστήματος
  - συμπεριφορά του χρήστη

## Εφαρμογές και φυσική μορφή

- Τα ΕΣ στοχεύουν σε συγκεκριμένη εφαρμογή (ή κατηγορία εφαρμογών)
- Το μέγεθος και η μορφή συχνά καθορίζονται από το είδος της εφαρμογής
  - Συσκευές ιατρικής παρακολούθησης που φοριούνται από ασθενείς, πρέπει να έχουν μορφή και βάρος που να μην παρεμποδίζουν την κίνηση του ασθενούς
  - Ένα κινητό τηλέφωνο πρέπει να έχει μέγεθος που να του επιτρέπει να χωράει στην τσέπη

## Ενσωματωμένοι υπολογιστές

- Πρωταρχικός στόχος: Επίτευξη της ζητούμενης απόδοσης με το μικρότερο δυνατό κόστος
- Απόδοση σύμφωνα με τις προδιαγραφές
  - π.χ. video frame processing
- Άλλα θέματα:
- Ελαχιστοποίηση απαιτούμενης μνήμης
  - Σημαντικό μέρος του συνολικού κόστους
- Ελαχιστοποίηση κατανάλωσης ισχύος
  - Χρήση μπαταρίας
  - packaging χαμηλότερου κόστους
  - απουσία ψύξης
- Αξιοπιστία, ασφάλεια
  - Για κρίσιμες εφαρμογές (safety critical applications)

# Ισχύς

- Η κατανάλωση ισχύος είναι καθοριστικός περιορισμός
- Στα ΕΣ η ισχύς κυμαίνεται από microWatt μέχρι μερικά Watt
  - συνήθως λειτουργούν σε μικρούς χώρους, με μπαταρία
  - η απαγωγή θερμότητας είναι πιο δύσκολη
- Αναγκαίοι μηχανισμοί διαχείρισης της ισχύος
  - Διαφορετικά προφίλ
    - χαμηλής κατανάλωσης
    - υψηλών επιδόσεων

## Συμπεριφορά χρήστη

- Οι **χρήστες υπολογιστών** πρέπει να μπορούν να αντιμετωπίσουν προβλήματα και αστοχίες της συσκευής
  - εφαρμογές που “κολλάνε”
  - ανάγκη για επανεγκατάσταση εφαρμογών
  - συνεχή update
  - έλλειψη συμβατότητας
  - Γενικότερα, ο χρήστης πρέπει να “εξειδικευτεί” στην αντιμετώπιση προβλημάτων
- Αντιθέτως, ένα ΕΣ αφήνει ελάχιστα περιθώρια για το χρήστη
  - Η λειτουργία πρέπει να είναι αξιόπιστη και ντετερμινιστική
    - Ο χρήστης περιμένει από ένα τηλέφωνο ή ένα media player να έχει άμεση απόκριση
    - Βιομηχανικά ΕΣ πρέπει να ανταποκρίνονται βάσει Service-Level Agreement

## Σχεδιαστικές Προκλήσεις

- Στόχος: Κατασκευή υλοποίησης με την επιθυμητή λειτουργικότητα
- Πρόκληση: ταυτόχρονη ικανοποίηση διαφορετικών σχεδιαστικών **μετρικών**
- Μετρική: Ένα μετρήσιμο χαρακτηριστικό μιας υλοποίησης

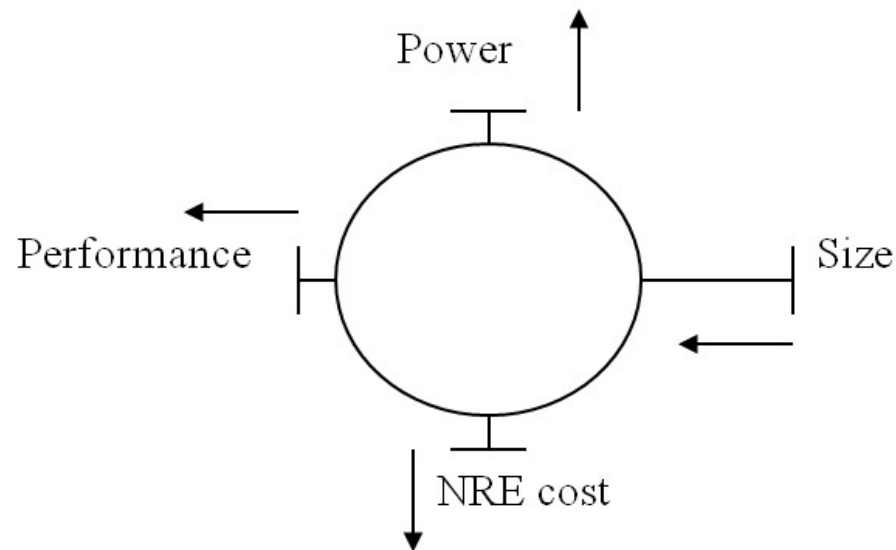
## Συνηθισμένες μετρικές I

- **NRE** (Non-Recurring Engineering cost): Το εφάπαξ χρηματικό κόστος σχεδιασμού του συστήματος
- **Κόστος μονάδας**: Το χρηματικό κόστος κατασκευής κάθε αντιγράφου του συστήματος, εξαιρώντας το NRE
- **Μέγεθος**: Ο φυσικός χώρος που απαιτείται από το σύστημα
- **Απόδοση**: Ο χρόνος εκτέλεσης ή το throughput του συστήματος
- **Ισχύς**: Ενέργεια που καταναλώνεται από το σύστημα
- **Flexibility**: η ικανότητα να αλλάξει η λειτουργικότητα του συστήματος χωρίς υψηλό κόστος NRE

## Συνηθισμένες μετρικές II

- **Time-to-prototype:** Ο απαιτούμενος χρόνος κατασκευής μιας λειτουργικής έκδοσης του συστήματος
- **Time-to-market:** Ο απαιτούμενος χρόνος για την ανάπτυξη ενός συστήματος που να μπορεί να πωληθεί
- **Συντηρησιμότητα:** Η δυνατότητα τροποποίησης του συστήματος μετά την αρχική κυκλοφορία του
- **Ορθότητα, ασφάλεια**

## Οι μετρικές είναι αντικρουόμενες...



- Για να μπορέσουμε να βελτιστοποιήσουμε τις μετρικές χρειάζεται εμπειρία τόσο με το λογισμικό όσο και με το υλικό
- Ο σχεδιαστής πρέπει να γνωρίζει διάφορες τεχνολογίες για να επιλέξει το καλύτερο με βάση
  - τη συγκεκριμένη εφαρμογή και
  - τους περιορισμούς

# NRE και κόστος μονάδας I

**Κόστος μονάδας:** Το χρηματικό κόστος κατασκευής κάθε αντιγράφου του συστήματος, εξαιρώντας το NRE

**NRE** (Non-Recurring Engineering cost): Το εφάπαξ χρηματικό κόστος σχεδιασμού του συστήματος

Συνολικό κόστος = NRE + (Κόστος μονάδας) \* (Πλήθος μονάδων)

Κόστος ανά προϊόν =

(Συνολικό κόστος) / (Πλήθος μονάδων) =

NRE / (Πλήθος μονάδων) + Κόστος μονάδας

Παράδειγμα: NRE = € 2000

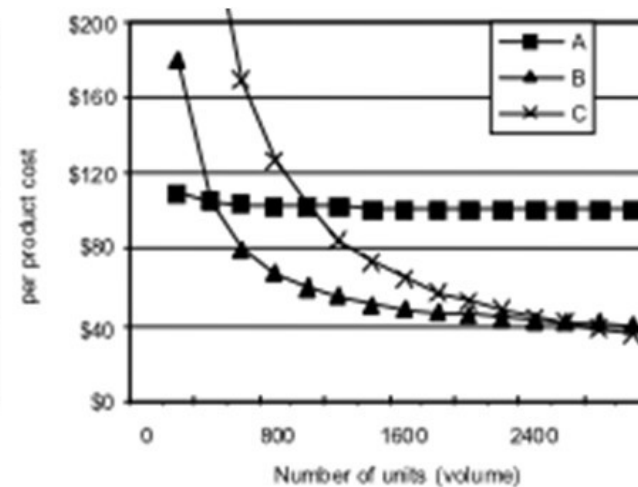
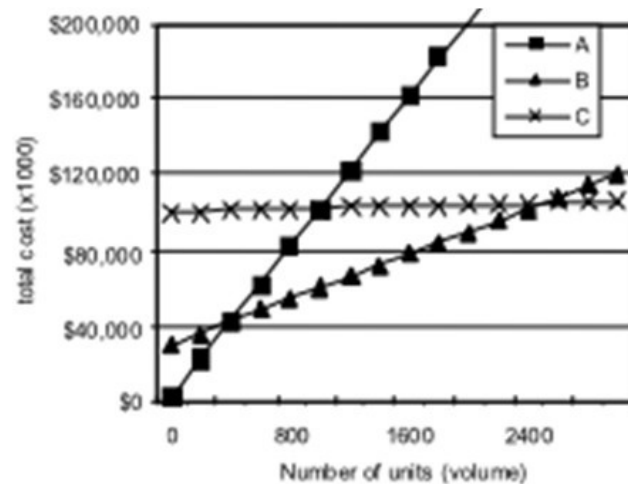
Κόστος μονάδας = € 100

Για 10 μονάδες:

Συνολικό κόστος = € 2000 + 10 \* € 100 = € 3000

Κόστος ανά προϊόν =  $\frac{€ 2000}{10} + € 100 = € 300$

## NRE και κόστος μονάδας II

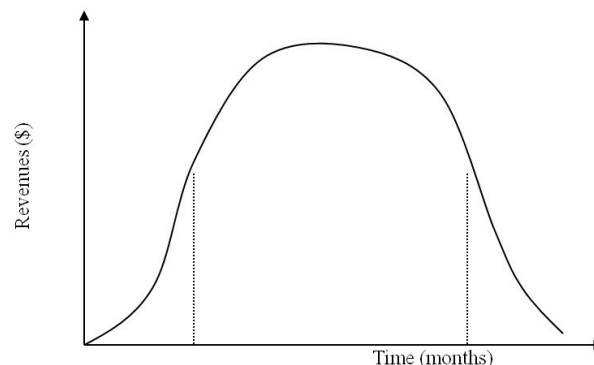


Τεχνολογία A: NRE = € 2.000, Κόστος μονάδας = € 100

Τεχνολογία B: NRE = € 30.000, Κόστος μονάδας = € 30

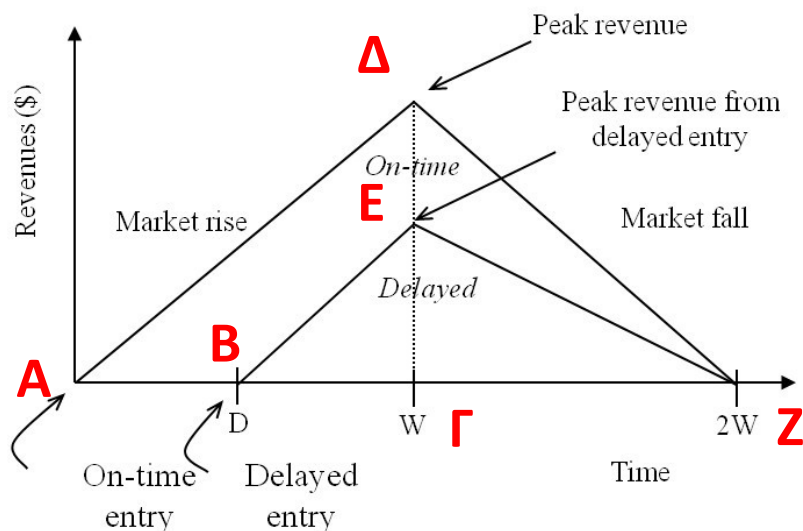
Τεχνολογία C: NRE = € 100.000, Κόστος μονάδας = € 2

# Time-to-market



- Ο χρόνος που απαιτείται για την ανάπτυξη ενός προϊόντος στο σημείο που να μπορεί να πωληθεί σε πελάτες
- Market window: Περίοδος κατά την οποία το προϊόν θα έχει τις υψηλότερες πωλήσεις
- Οι καθυστερήσεις εισάγουν «χασούρα»

# Time-to-market: Απλοποιημένη προσέγγιση



Απλοποιήσεις:

1. Τα τρίγωνα ΑΓΔ, ΑΔΖ θεωρούνται ισοσκελή
2. Οι ΑΔ και ΒΕ είναι παράλληλες

Η διάρκεια ζωής του προϊόντος είναι  $2W$

Δημιουργείται ένα τρίγωνο βάσει των χρόνων

Το εμβαδόν του τριγώνου δίνει τις πωλήσεις

Η διαφορά μεταξύ των 2 τριγώνων δίνει τις απώλειες λόγω καθυστέρησης

# Παράδειγμα

- Ο κύκλος ζωής ενός προϊόντος είναι 24 μήνες ( $2W = 24$ ).
- Το συνολικό πλήθος συσκευών που αναμένεται να πουλήσει το προϊόν είναι 100.000.
- Δύο ανταγωνίστριες εταιρείες ξεκινούν τη διαδικασία σχεδιασμού του προϊόντος ταυτόχρονα.
- Η Εταιρεία Α επιλέγει να επενδύσει NRE 500.000 ευρώ και ολοκληρώνει το σχεδιασμό του προϊόντος σε 4 μήνες.
- Το κόστος κατασκευής της κάθε συσκευής για την Α είναι 40 ευρώ.
- Η Εταιρεία Β επιλέγει να επενδύσει μικρότερο NRE (100.000) και να ολοκληρώσει το σχεδιασμό σε 8 μήνες με το ίδιο κόστος κατασκευής.
- Οι δύο εταιρείες πωλούν το προϊόν στην ίδια τιμή, 80 ευρώ.
- Θεωρήστε ότι οι πωλήσεις ακολουθούν την κατανομή τριγώνου και ότι δεν υπάρχει άλλη ανταγωνίστρια εταιρεία στο συγκεκριμένο προϊόν.
- Ποιο είναι το συνολικό πλήθος συσκευών που πούλησε κάθε εταιρεία και ποιο το κέρδος κάθε εταιρείας;
- Ποιο το συνολικό πλήθος συσκευών για το οποίο οι δύο εταιρείες έχουν το ίδιο κέρδος;

**ΜΕΧΡΙ ΕΔΩ 17/10/2018**

# Απόδοση

- Ευρέως χρησιμοποιούμενη μετρική
- Δύσκολα μετρήσιμη
- **Συχνότητα ρολογιού, εντολές ανά δευτερόλεπτο**
  - Είναι καλό μέτρο;
- Ψηφιακή φωτογραφική –
  - Ο χρήστης ενδιαφέρεται για την ταχύτητα επεξεργασίας των εικόνων,
  - όχι για την ταχύτητα του ρολογιού ή τις εντολές ανά δευτερόλεπτο
- Καθυστέρηση (χρόνος απόκρισης)
  - Χρόνος μεταξύ έναρξης και λήξης μιας εργασίας
  - Μια κάμερα A επεξεργάζεται εικόνες σε 0,25 sec
  - Άρα σε 1 sec επεξεργάζεται 4 εικόνες

# Απόδοση (2)

## Throughput

- Εργασίες ανά δευτερόλεπτο
- Η κάμερα A επεξεργάζεται 4 εικόνες/sec
- Το throughput μπορεί να είναι μεγαλύτερο από όσο επιβάλλει ο χρόνος καθυστέρησης
- Μια κάμερα B μπορεί να έχει χρόνο επεξεργασίας 0,25 sec, αλλά να επεξεργάζεται 8 εικόνες/sec
- Λήψη νέας εικόνας ενώ η προηγούμενη αποθηκεύεται

# Τεχνολογίες ΕΣ

## Τεχνολογία - Ορισμός:

Ο τρόπος επίτευξης μιας εργασίας, με τη χρήση τεχνικών διαδικασιών, μεθόδων ή γνώσης

Στα ΕΣ εξετάζουμε τεχνολογίες:

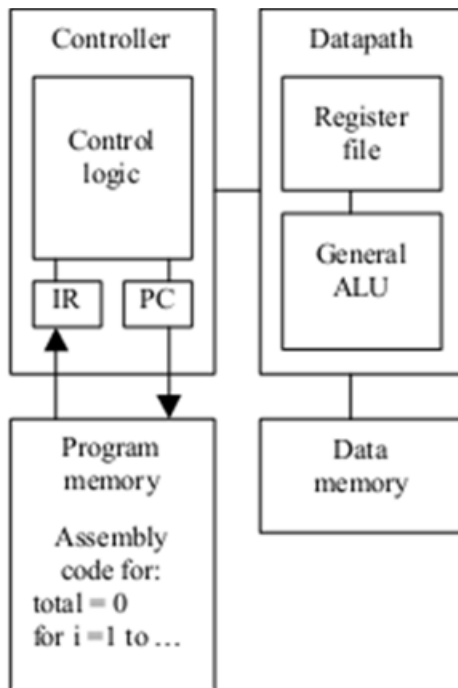
Επεξεργαστή

Ολοκληρωμένων Κυκλωμάτων

# Τεχνολογία Επεξεργαστή I

Η αρχιτεκτονική της μηχανής υπολογισμού που χρησιμοποιείται για την υλοποίηση της επιθυμητής λειτουργικότητας ενός συστήματος

"Επεξεργαστής" δε σημαίνει πάντα επεξεργαστής γενικού σκοπού



Επεξεργαστής γενικού σκοπού

## Τεχνολογία Επεξεργαστή II

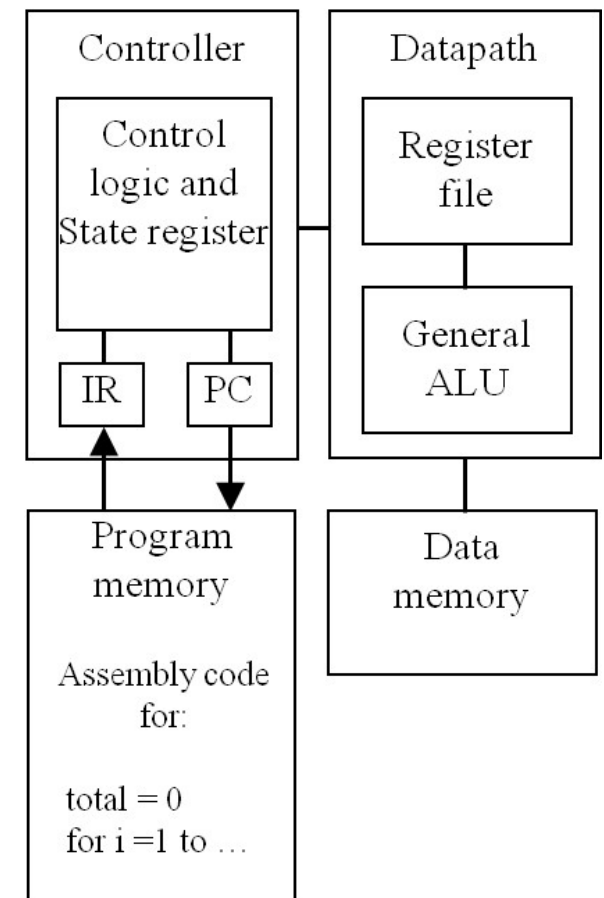
Οι επεξεργαστές διαφέρουν στην προσαρμογή τους ανάλογα με το πρόβλημα



```
total = 0  
for i = 1 to N loop  
  total += M[i]  
end loop
```

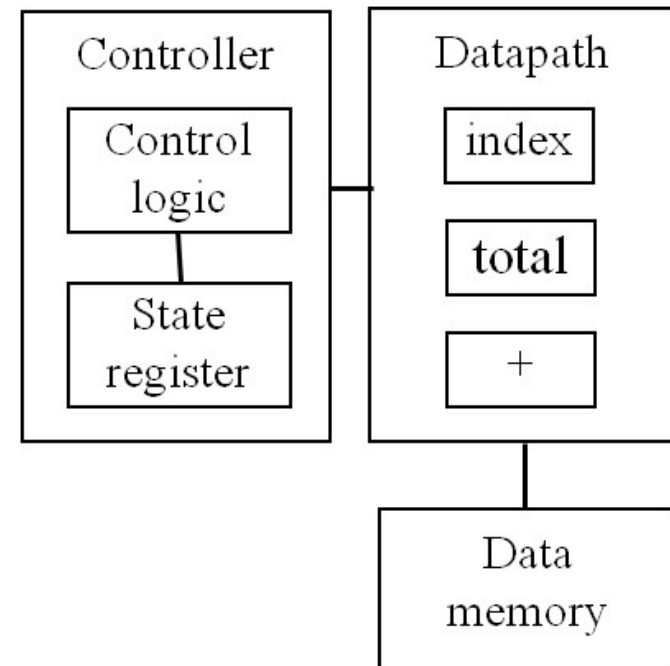
# Επεξεργαστής γενικού σκοπού

- Προγραμματιζόμενη συσκευή
- Χρησιμοποιείται σε πολλές εφαρμογές
- Πιο γνωστοί οι x86, i3, i5, i7
- Χαρακτηριστικά
  - Μνήμη προγράμματος
  - Γενικό datapath
  - Μεγάλο αρχείο καταχωρητών
  - Γενική ALU
- Οφέλη χρήσης
  - Μικρό NRE κόστος
  - Μικρό time-to-market
  - Μεγάλη ευελιξία



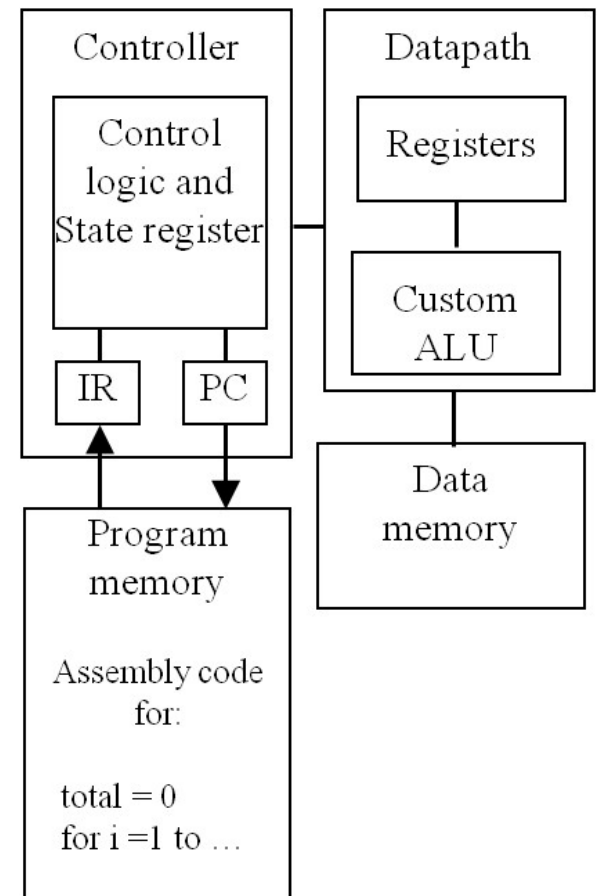
# Επεξεργαστής μοναδικού σκοπού

- Ψηφιακό κύκλωμα ειδικά σχεδιασμένο για να εκτελεί ακριβώς ένα πρόγραμμα
  - Συνεπεξεργαστής, επιταχυντής ή περιφερειακό
- Χαρακτηριστικά
  - Περιέχει μόνο τα στοιχεία που απαιτούνται για να εκτελέσει το συγκεκριμένο πρόγραμμα
  - Δεν υπάρχει μνήμη προγράμματος
- Οφέλη χρήσης
  - Γρήγορος
  - Χαμηλή κατανάλωση
  - Μικρο μέγεθος



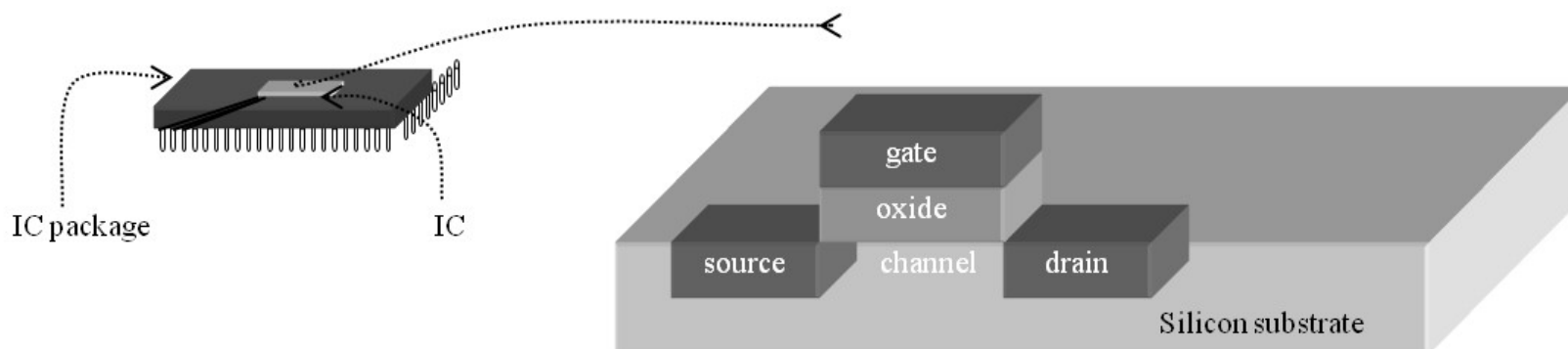
# Επεξεργαστής για εξειδικευμένες εφαρμογές

- Προγραμματιζόμενος επεξεργαστής
- Βελτιστοποιημένος για μια συγκεκριμένη κατηγορία εφαρμογών
- Συμβιβασμός μεταξύ επεξεργαστή γενικής χρήσης και μοναδικού σκοπού
- Χαρακτηριστικά
  - Μνήμη προγράμματος
  - Βελτιστοποιημένο datapath
  - Ειδικές δομικές μονάδες
- Οφέλη χρήσης
  - Αποτελεί συμβιβασμό ανάμεσα στα προηγούμενα δύο είδη
  - Μερική ευελιξία - Καλή απόδοση
  - Σχετικά χαμηλή κατανάλωση και μέγεθος



# Τεχνολογία Ολοκλήρωσης

- Η μέθοδος με την οποία μια ψηφιακή υλοποίηση απεικονίζεται σε ένα IC
  - IC: Integrated circuit ή “chip”
  - Τα IC μπορεί να αποτελούνται από πολλαπλά επίπεδα



# Τρανζίστορ

- Κατασκευάστηκε το 1948 στα Bell Labs
- In acknowledgement of this accomplishment, Shockley, Bardeen, and Brattain were jointly awarded the 1956 Nobel Prize in Physics "for their researches on semiconductors and their discovery of the transistor effect"

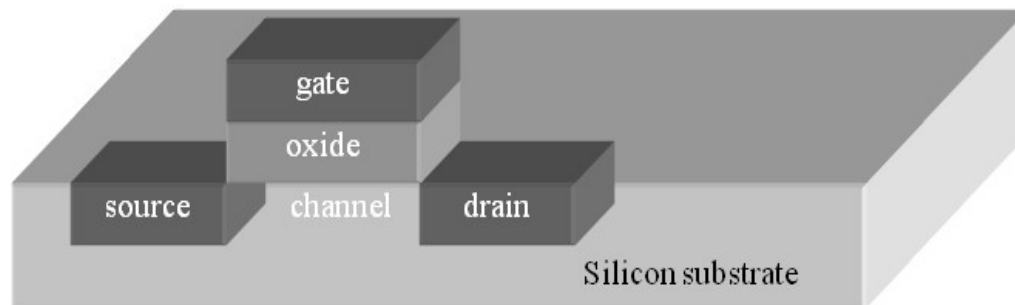


John Bardeen, William Shockley and Walter Brattain at Bell Labs, 1948.

# Λειτουργία Τρανζίστορ

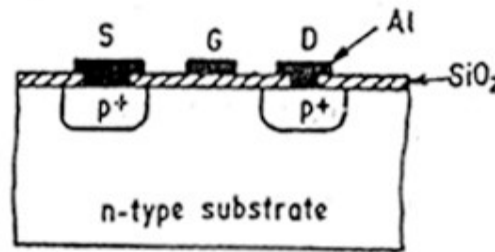
---

- Σαν διακόπτης
- Επιτρέπει (ή όχι) τη διέλευση ρεύματος από το source στο drain ανάλογα με το ρεύμα στο gate

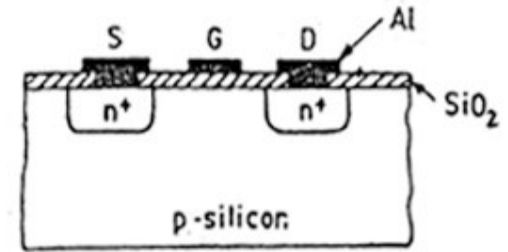


# Λειτουργία Τρανζίστορ

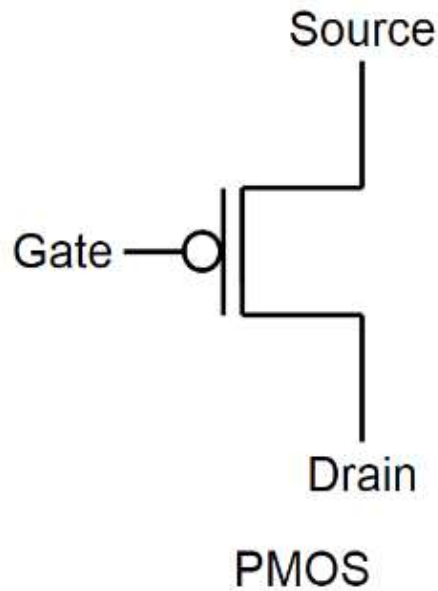
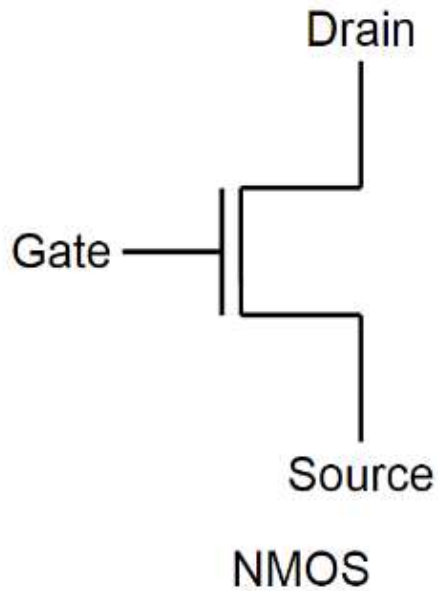
- Δύο τύποι: nMOS και pMOS



Cross-section of PMOS structure.



Cross-section of NMOS structure.

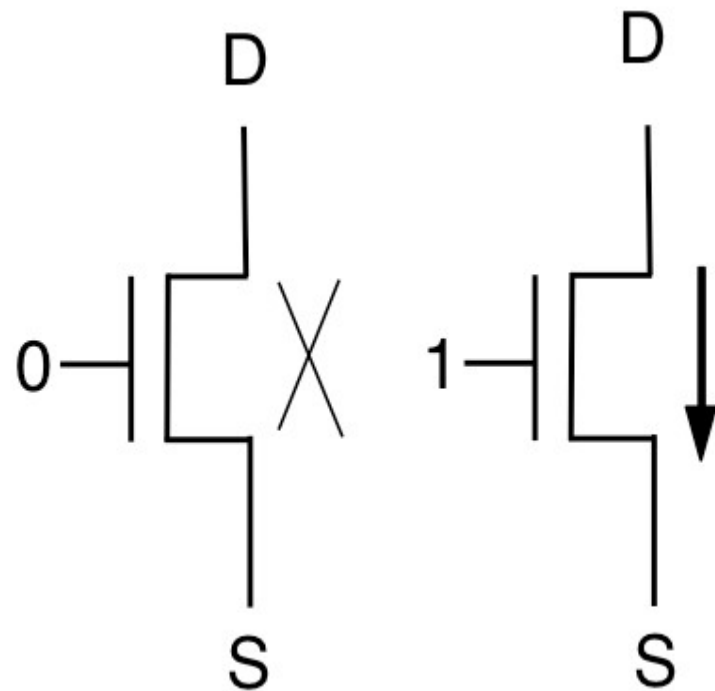


# Λειτουργία Τρανζίστορ

---

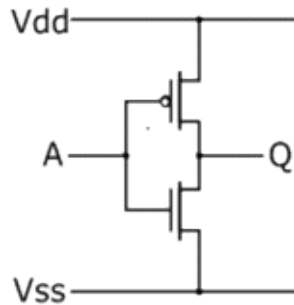
## nMOS και pMOS

- nMOS: με 1 άγει, με 0 δεν άγει
- pMOS: με 0 άγει, με 1 δεν άγει



# Τρανζίστορ και πύλες

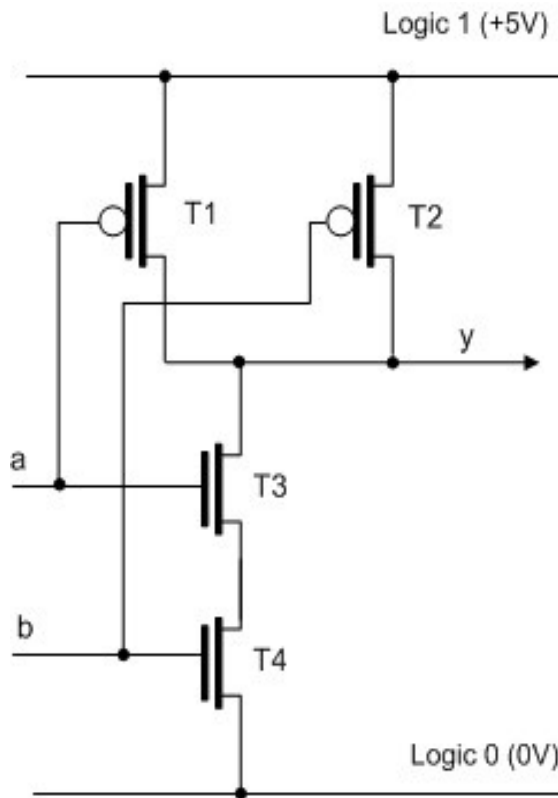
Συνδέοντας transistor κατάλληλα μπορούμε να φτιάξουμε πύλες ...



Πύλη NOT με transistor

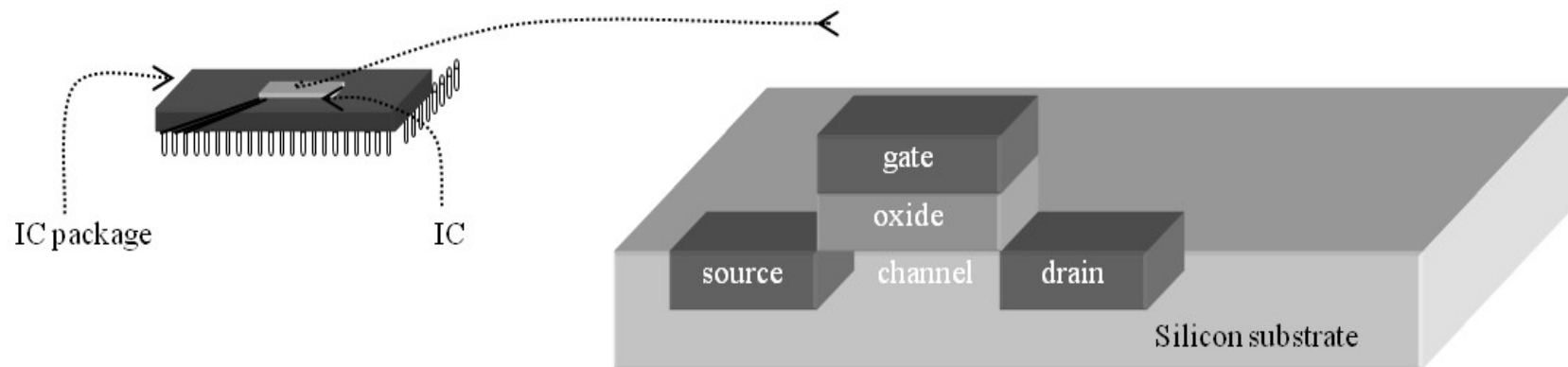
nMOS=> 1: A, 0: ΔA

pMOS=> 0: A, 1: ΔA



# Τρανζίστορ και ολοκληρωμένο κύκλωμα

...και τελικά ολοκληρωμένα κυκλώματα



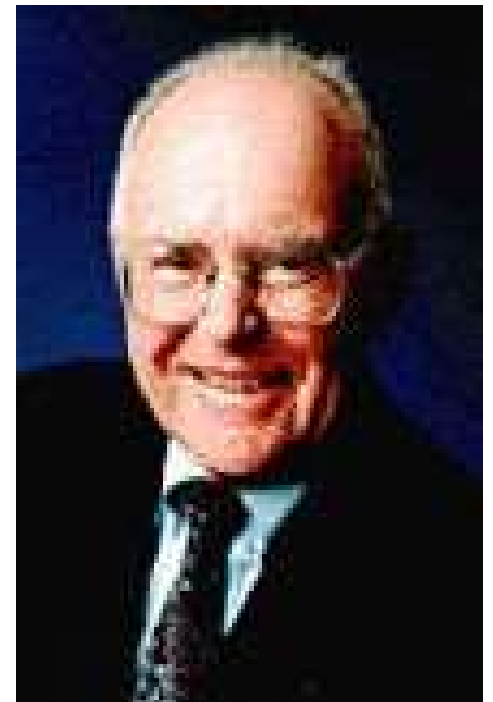
Τεχνολογία Κατασκευής (65 nm, 13 nm etc)

# Κάποιες προβλέψεις αποδείχθηκαν σωστές...

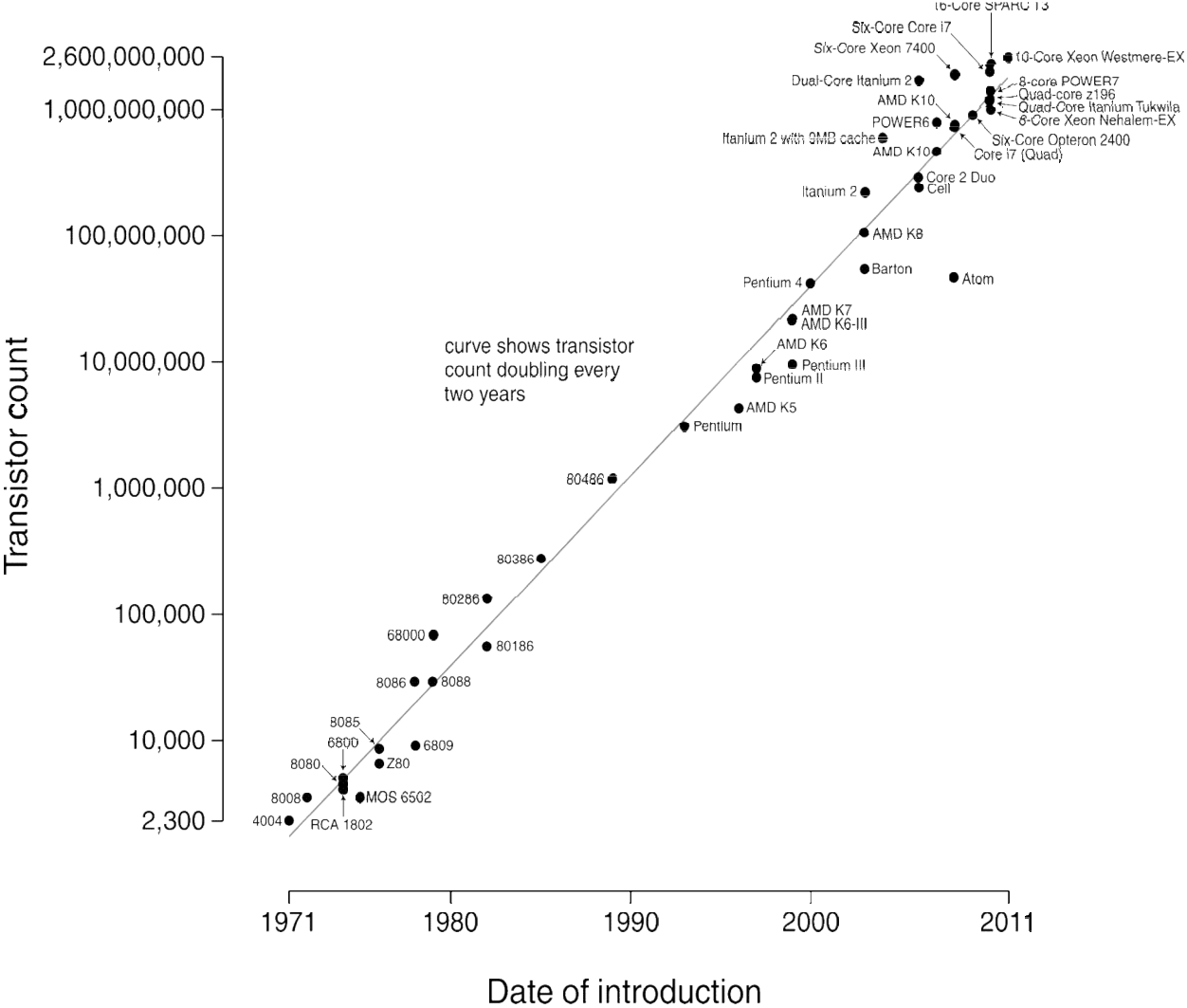
---

- Νόμος του Moore
  - Ο Gordon Moore προέβλεψε (1965, 1975) την συρρίκνωση του μεγέθους των transistors
  - Διπλασιασμός της πυκνότητας ολοκλήρωσης (# transistors)
    - Κάθε 12 μήνες (1965)
    - Κάθε 24 μήνες (1975)

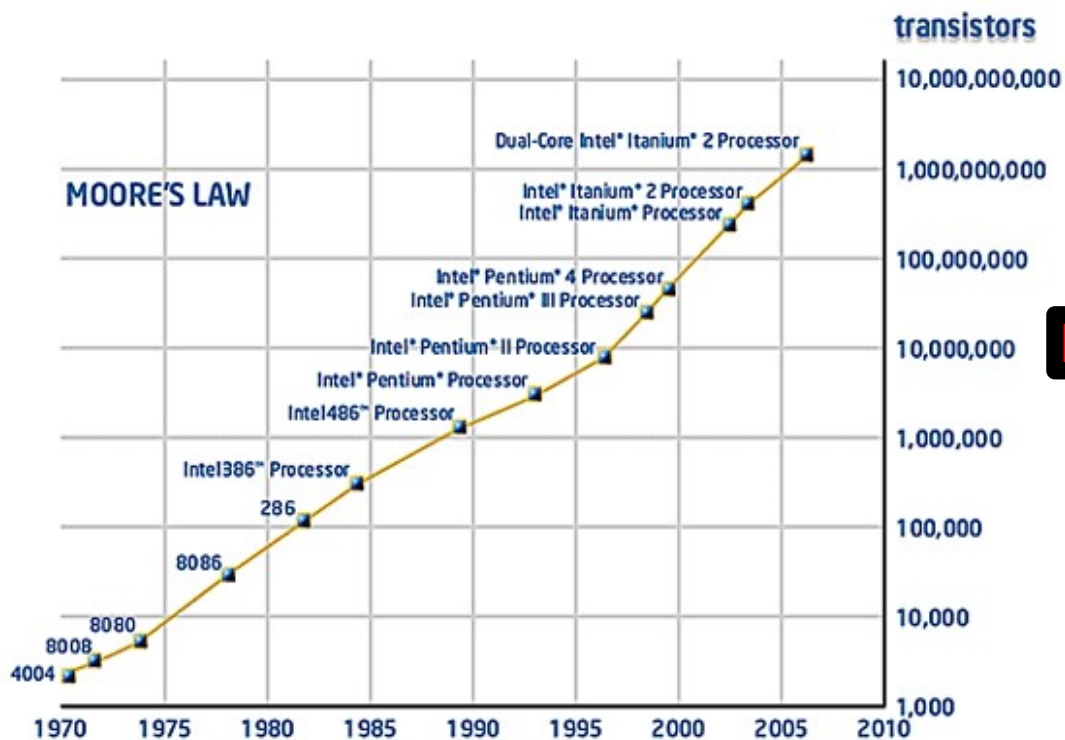
Gordon Moore Συνιδρυτής της Intel



# Νόμος του Moore



# Μια αναλογία ...



Το 1978 μια πτήση Νέα Υόρκη-Παρίσι κόστιζε **900 ευρώ** και διαρκούσε **7 ώρες**

Αν οι αεροπορικές μεταφορές είχαν την ίδια εξέλιξη με αυτή του νόμου του Moore, το ίδιο ταξίδι θα κόστιζε ένα **λεπτό του ευρώ** και θα διαρκούσε **1 δευτερόλεπτο!**

# Αποτελέσματα της εξέλιξης

---

- Αύξηση της απόδοσης
- Μείωση του κόστους
- Μικρότερα ολοκληρωμένα κυκλώματα (O.K.)
- Περισσότερες λειτουργίες

# Κάποιες προβλέψεις αποδείχθηκαν λαθασμένες...

---

**“I think there is a world market for  
maybe five computers.”**

Thomas Watson,  
Chairman of IBM, 1943

**“There is no reason for any  
individual to have a computer in  
their home.”**

Ken Olson,  
President, Chairman and Founder of  
Digital Equipment Corp., 1977

**“640K ought to be enough for  
anybody.”**

Bill Gates, Microsoft founder, 1981  
(though today he denies he said it)

# Κατηγορίες Υπολογιστών

---



Επιτραπέζιοι  
(Desktop)

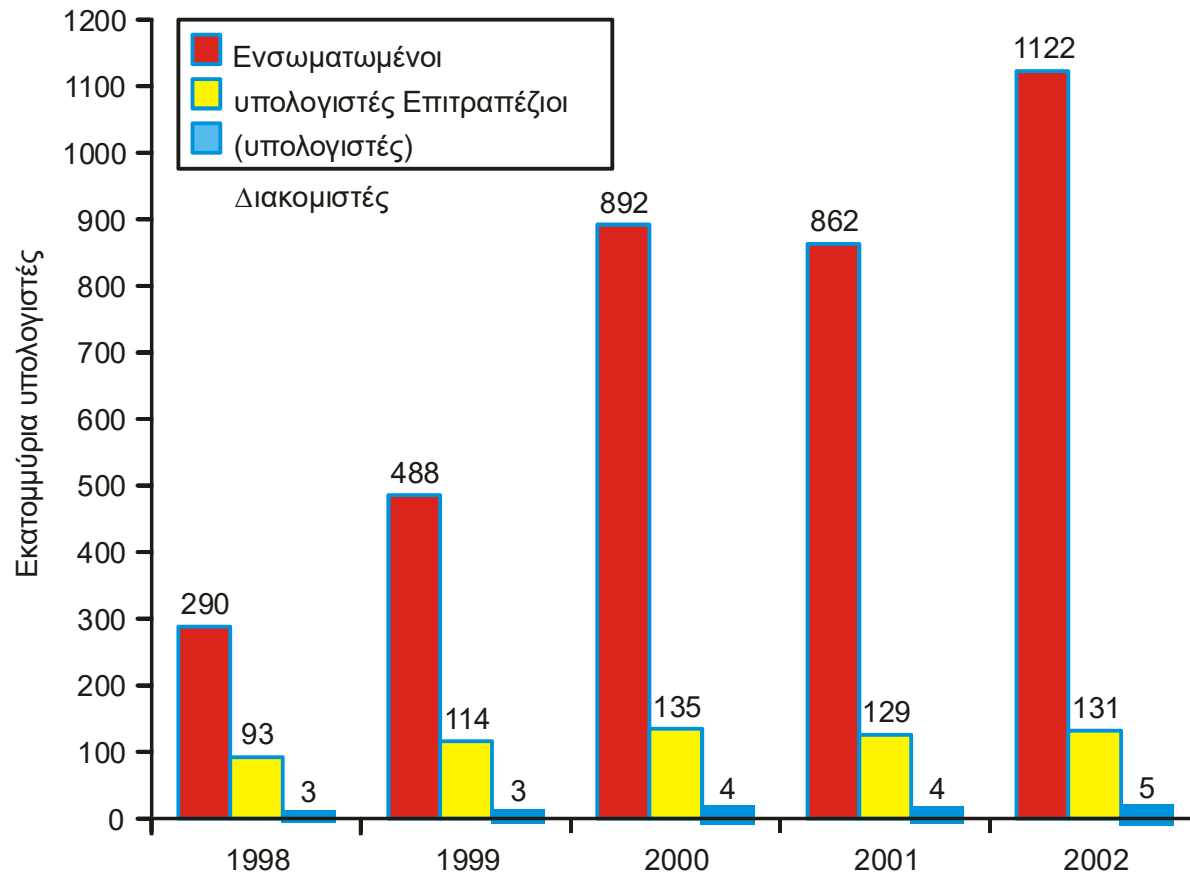


Διακομιστές  
(Servers)

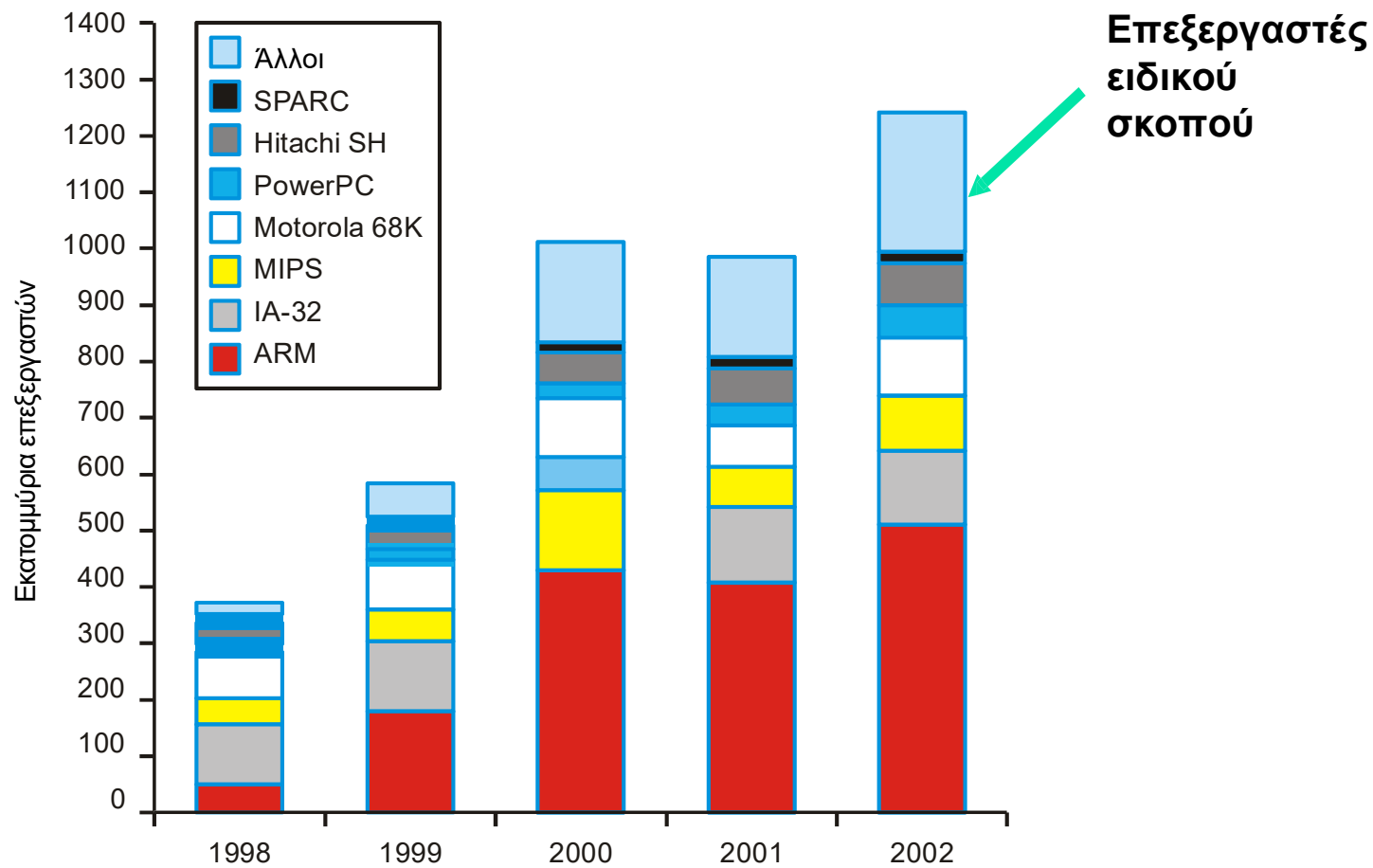


Ενσωματωμένοι  
(Embedded)

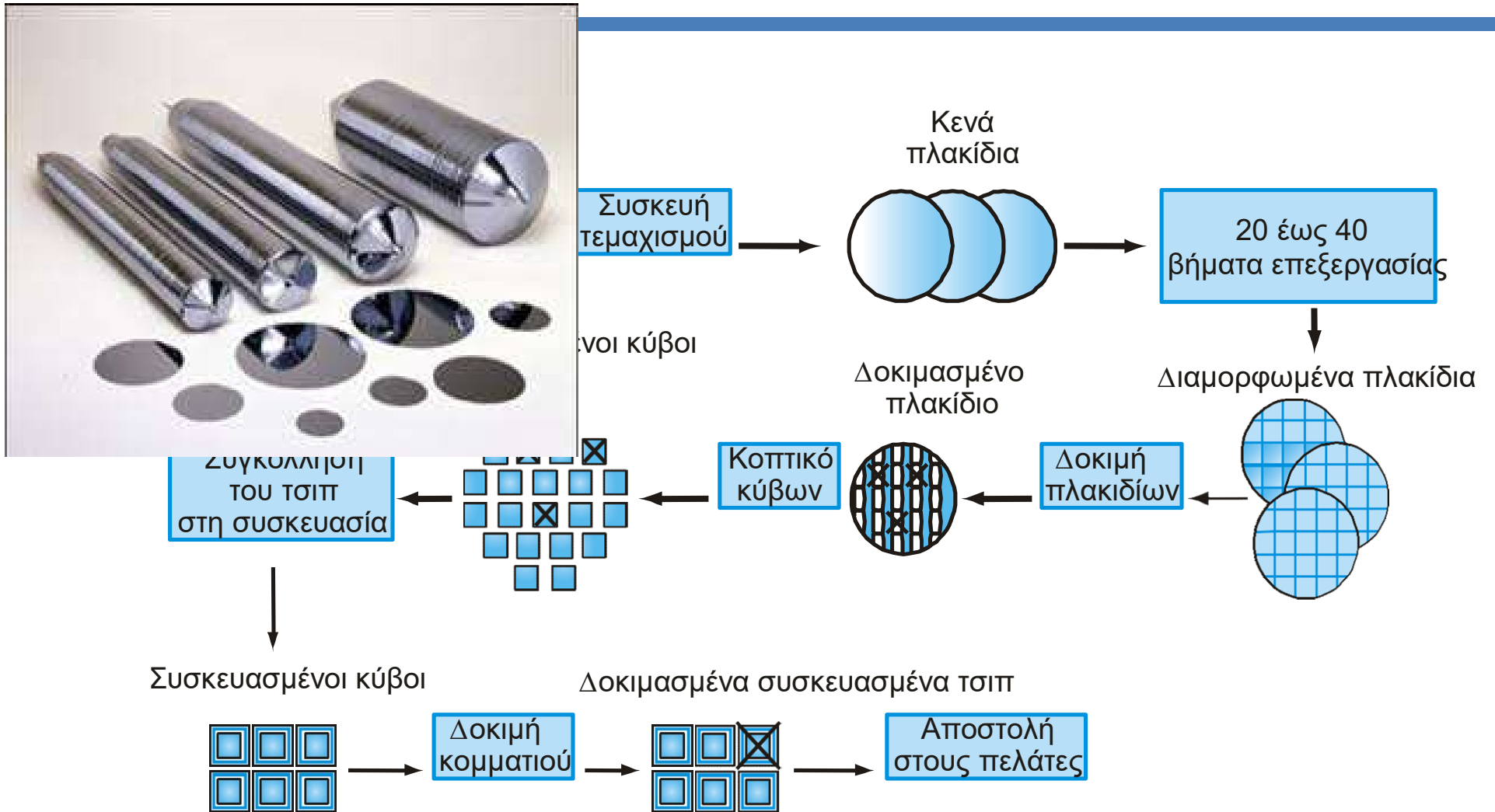
# Αγορά υπολογιστών



# Αγορά επεξεργαστών

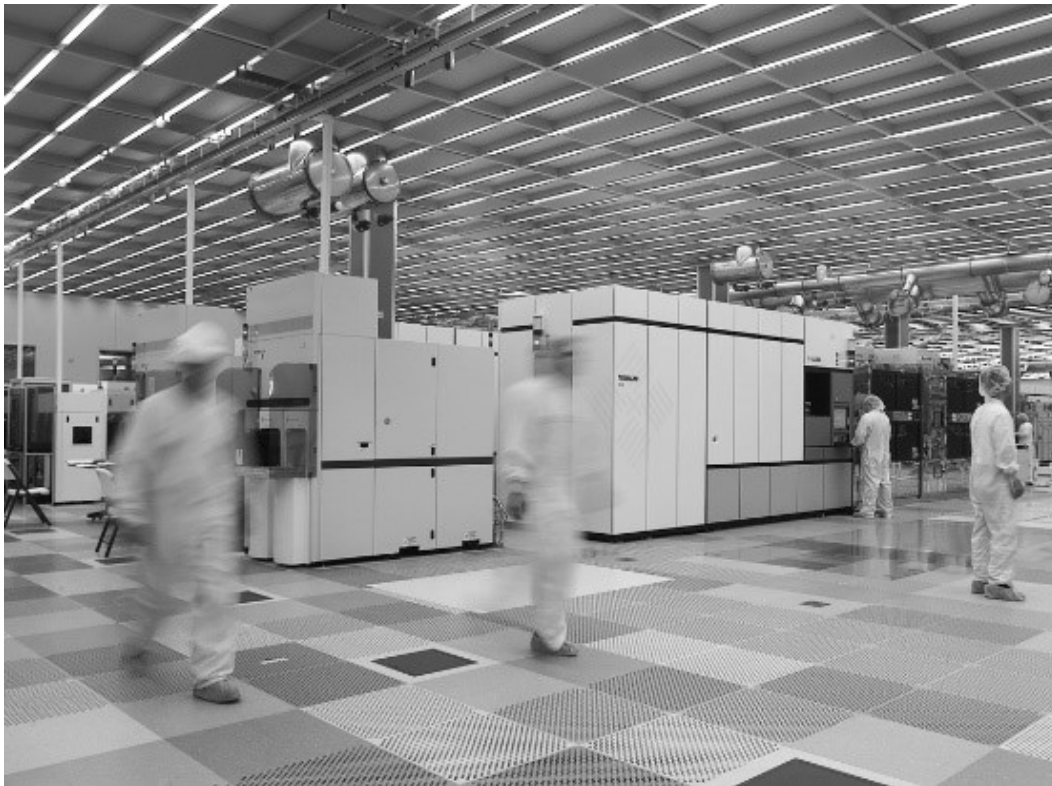


# Διαδικασία κατασκευής ΟΚ



# Fabrication

- ❑ Chips are built in huge factories called fabs
- ❑ Contain clean rooms as large as football fields



Courtesy of International  
Business Machines Corporation.  
Unauthorized use not permitted.

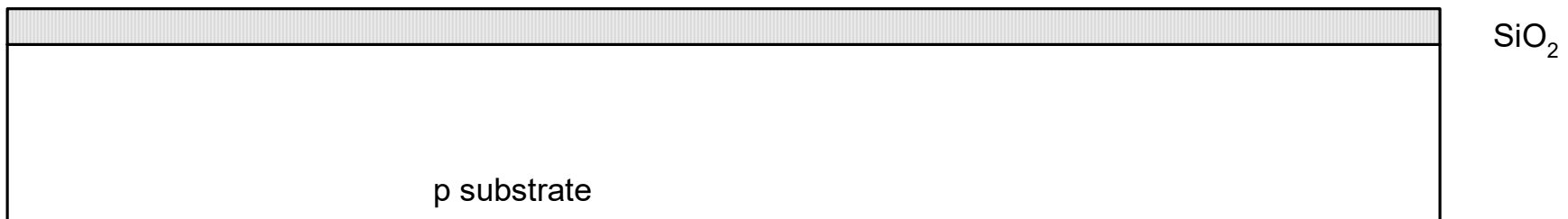
# Fabrication Steps

- ❑ Start with blank wafer
- ❑ Build inverter from the bottom up
- ❑ First step will be to form the n-well
  - Cover wafer with protective layer of  $\text{SiO}_2$  (oxide)
  - Remove layer where n-well should be built
  - Implant or diffuse n dopants into exposed wafer
  - Strip off  $\text{SiO}_2$

p substrate

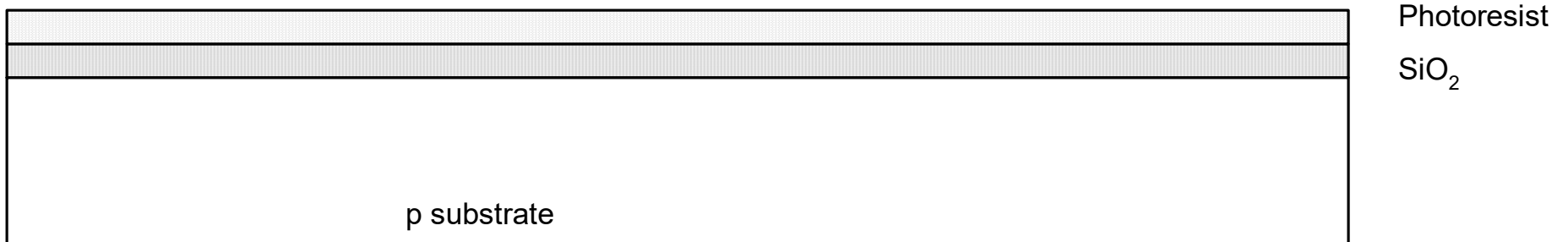
# Oxidation

- Grow  $\text{SiO}_2$  on top of Si wafer
  - 900 – 1200 C with  $\text{H}_2\text{O}$  or  $\text{O}_2$  in oxidation furnace



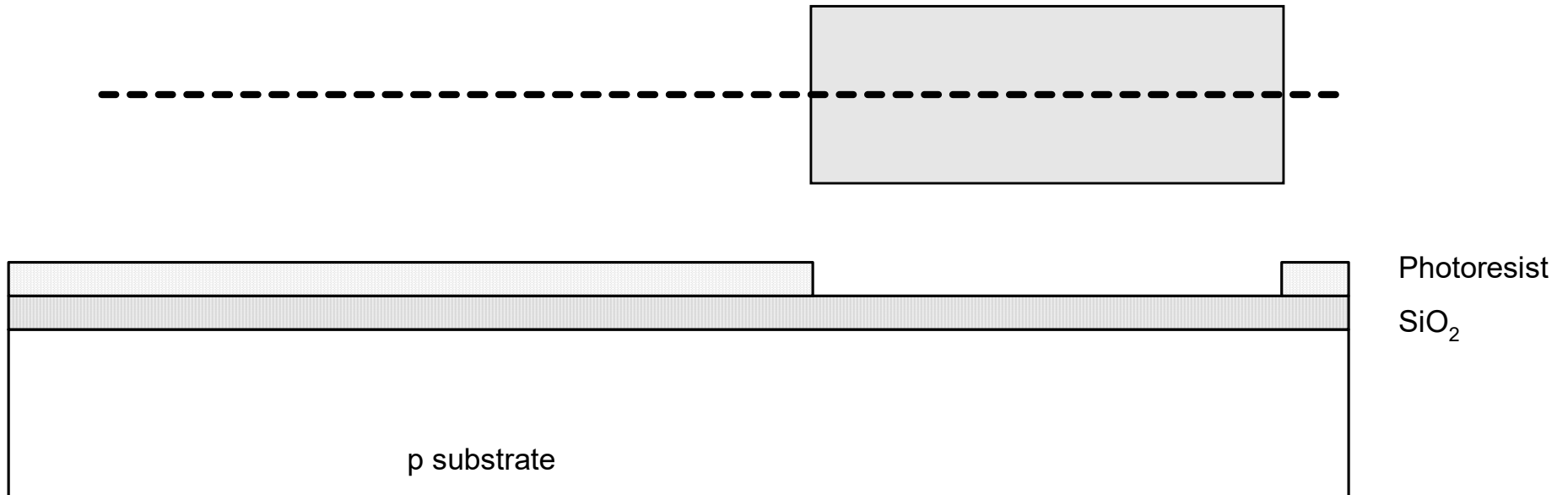
# Photoresist

- Spin on photoresist
  - Photoresist is a light-sensitive organic polymer
  - Softens where exposed to light



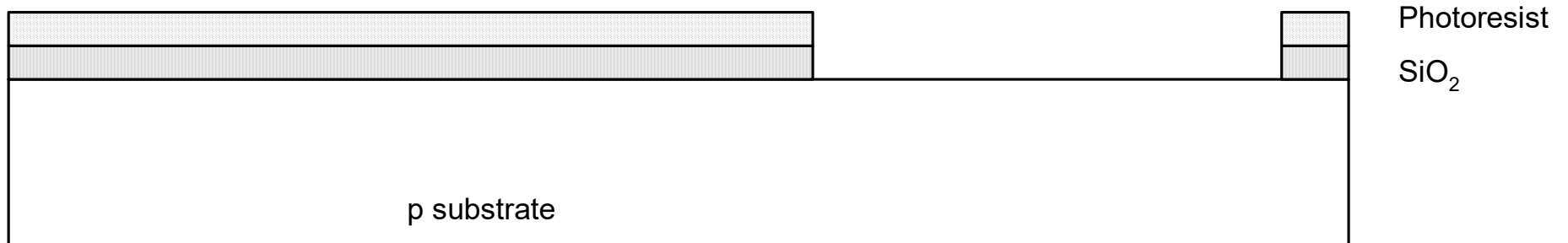
# Lithography

- Expose photoresist through n-well mask
- Strip off exposed photoresist



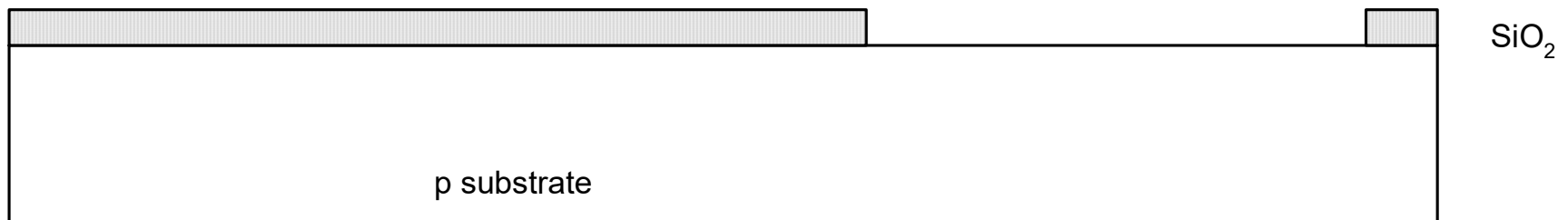
# Etch

- ❑ Etch oxide with hydrofluoric acid (HF)
  - Seeps through skin and eats bone; nasty stuff!!!
- ❑ Only attacks oxide where resist has been exposed



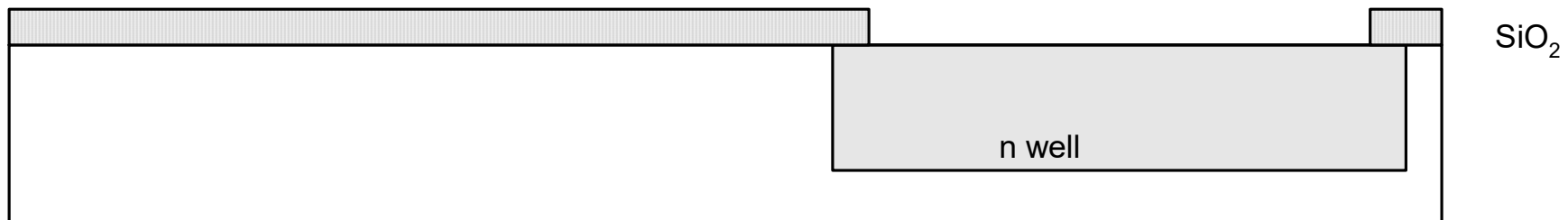
# Strip Photoresist

- ❑ Strip off remaining photoresist
  - Use mixture of acids called piranha etch
- ❑ Necessary so resist doesn't melt in next step



# n-well

- ❑ n-well is formed with diffusion or ion implantation
- ❑ Diffusion
  - Place wafer in furnace with arsenic gas
  - Heat until As atoms diffuse into exposed Si
- ❑ Ion Implantation
  - Blast wafer with beam of As ions
  - Ions blocked by  $\text{SiO}_2$ , only enter exposed Si



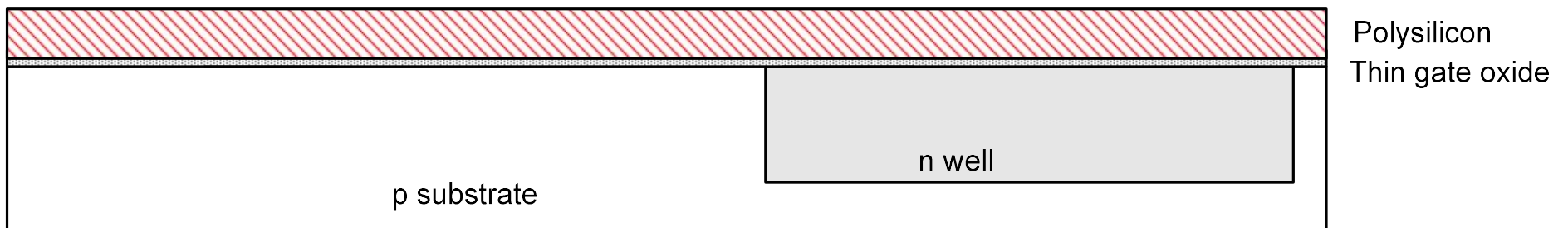
# Strip Oxide

- ❑ Strip off the remaining oxide using HF
- ❑ Back to bare wafer with n-well
- ❑ Subsequent steps involve similar series of steps



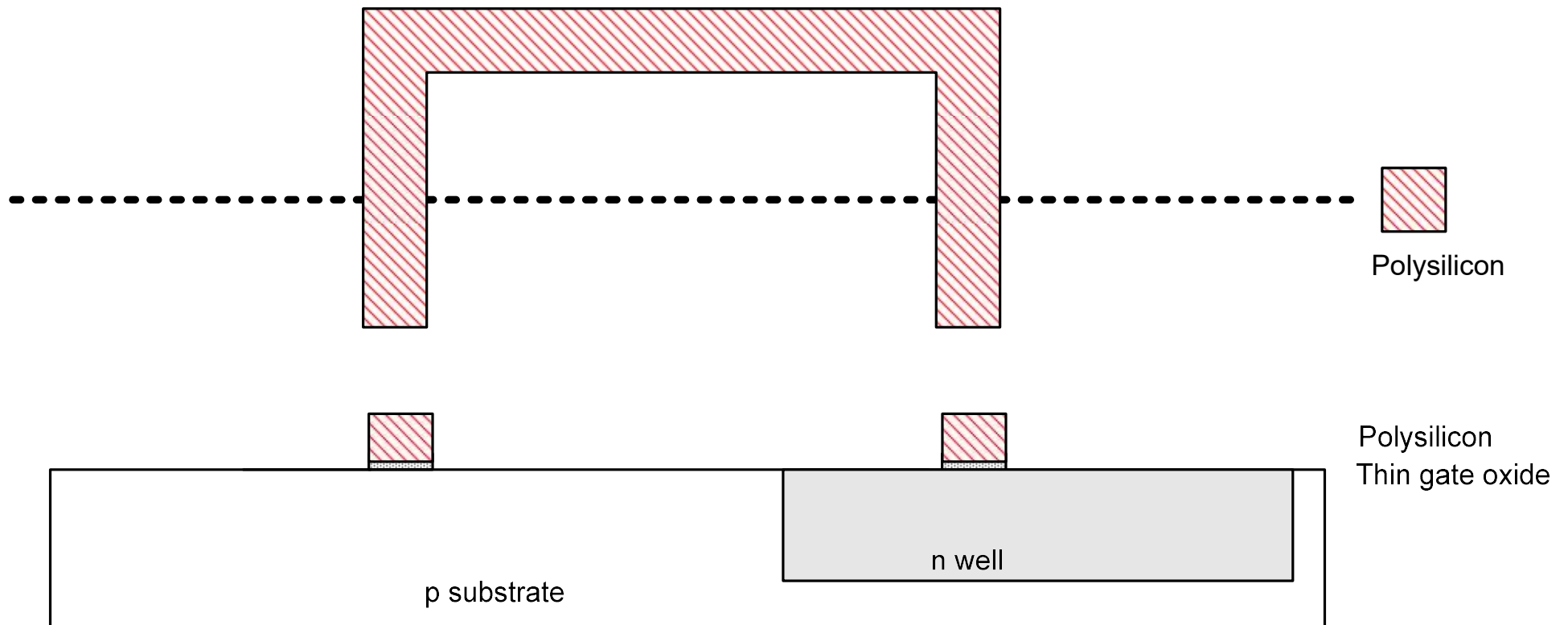
# Polysilicon

- ❑ Deposit very thin layer of gate oxide
  - $< 20 \text{ \AA}$  (6-7 atomic layers)
- ❑ Chemical Vapor Deposition (CVD) of silicon layer
  - Place wafer in furnace with Silane gas ( $\text{SiH}_4$ )
  - Forms many small crystals called polysilicon
  - Heavily doped to be good conductor



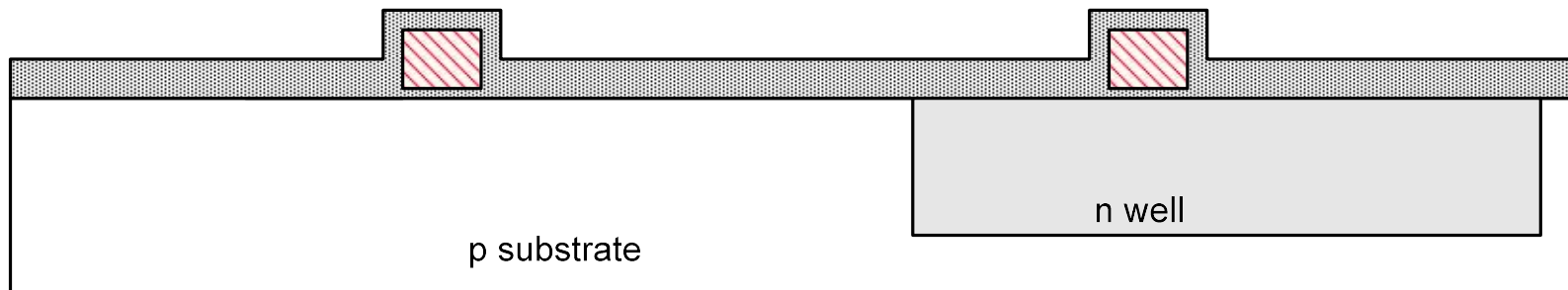
# Polysilicon Patterning

- ❑ Use same lithography process to pattern polysilicon



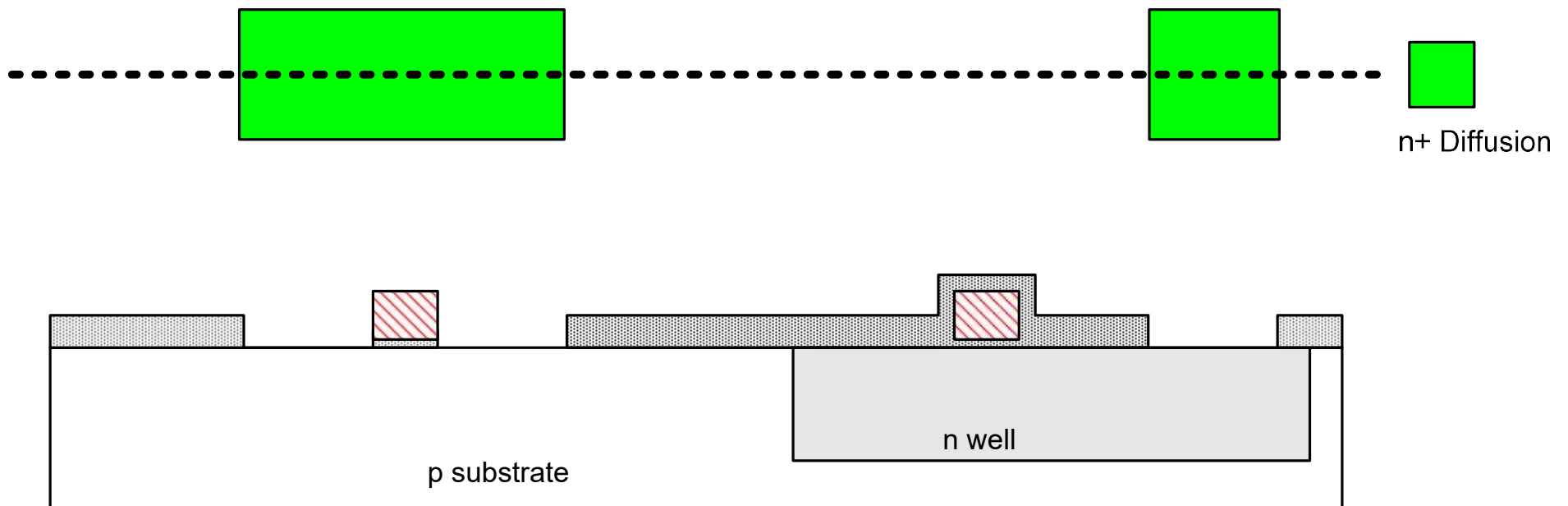
# Self-Aligned Process

- ❑ Use oxide and masking to expose where n+ dopants should be diffused or implanted
- ❑ N-diffusion forms nMOS source, drain, and n-well contact



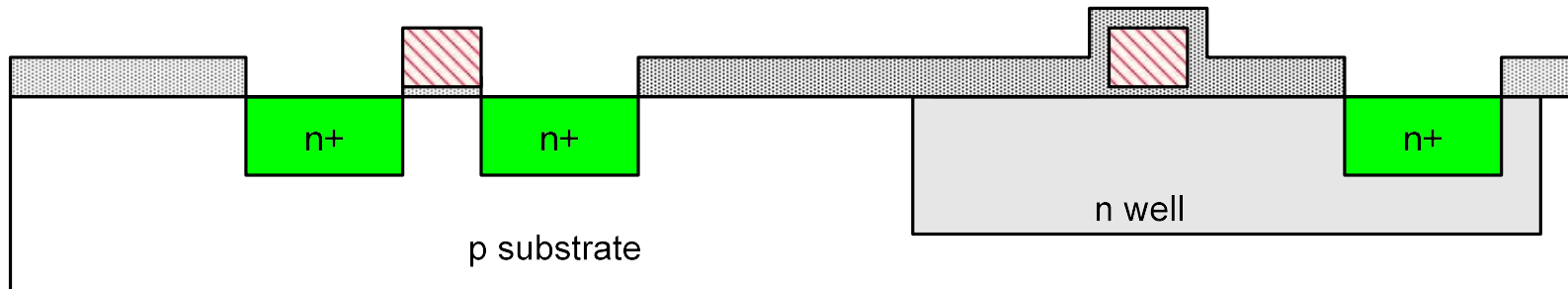
# N-diffusion

- ❑ Pattern oxide and form n+ regions
- ❑ *Self-aligned process* where gate blocks diffusion
- ❑ Polysilicon is better than metal for self-aligned gates because it doesn't melt during later processing



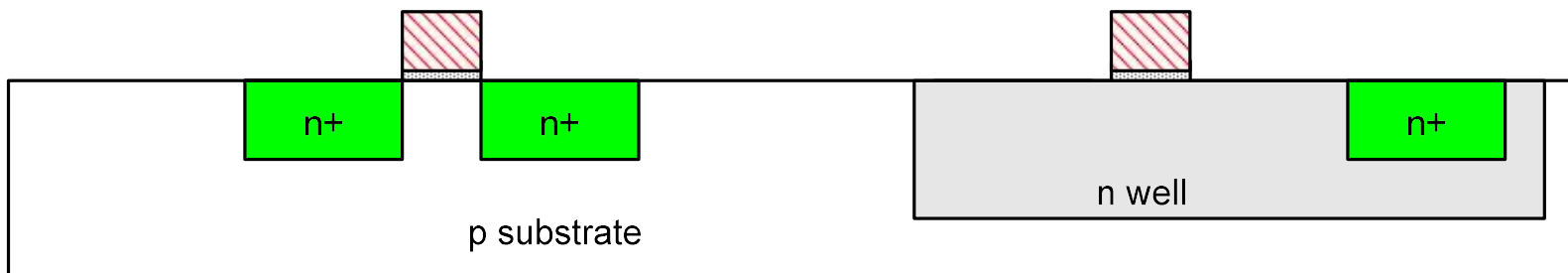
# N-diffusion cont.

- ❑ Historically dopants were diffused
- ❑ Usually ion implantation today
- ❑ But regions are still called diffusion



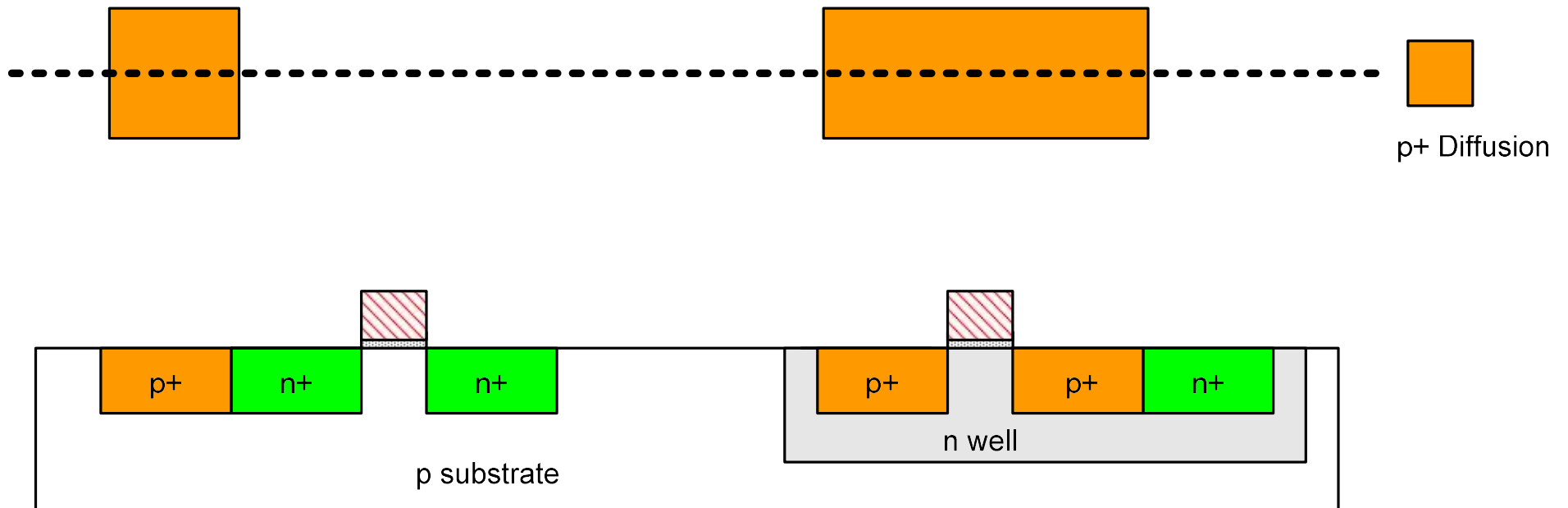
# N-diffusion cont.

- Strip off oxide to complete patterning step



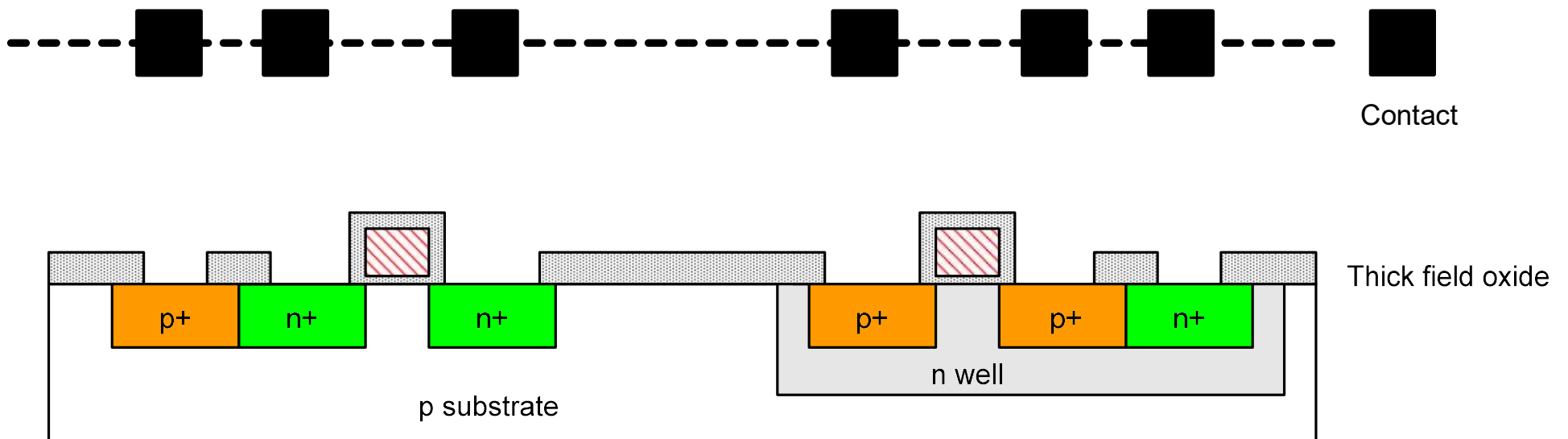
# P-Diffusion

- Similar set of steps form p+ diffusion regions for pMOS source and drain and substrate contact



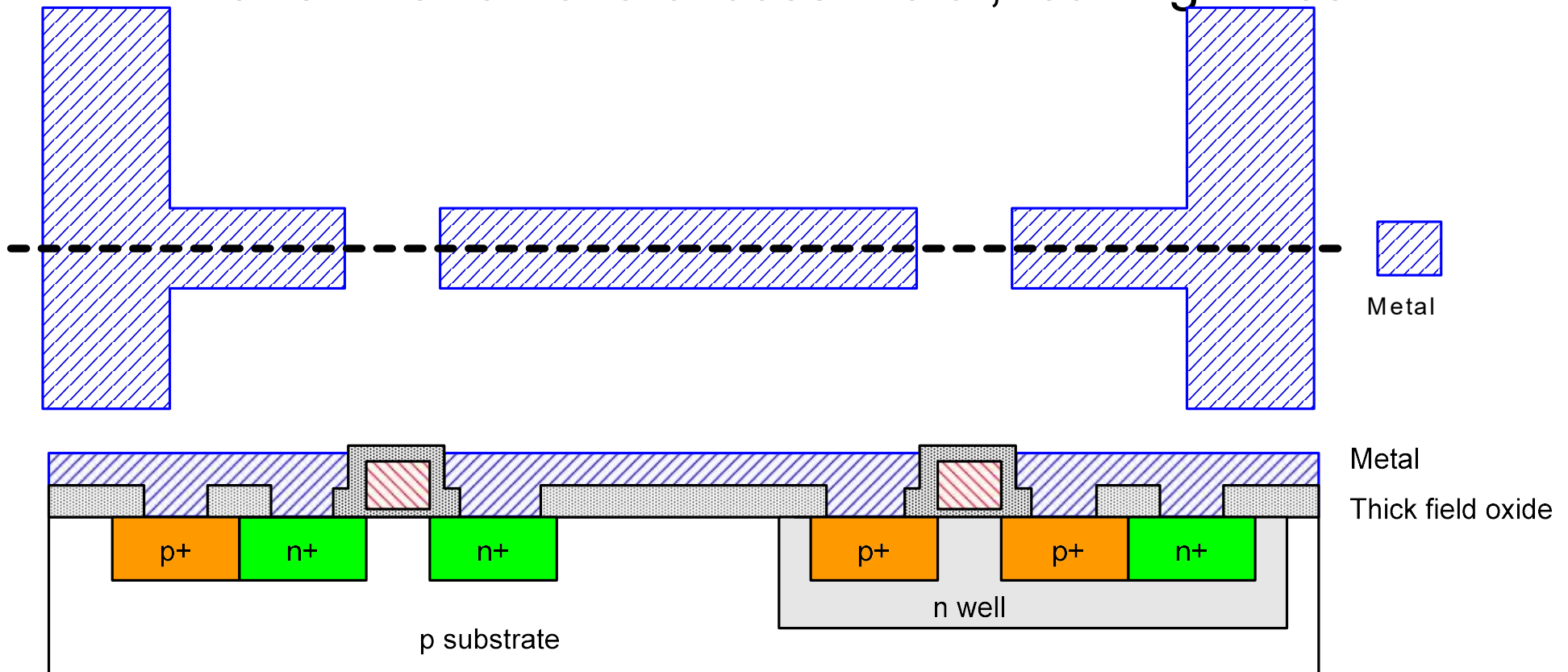
# Contacts

- ❑ Now we need to wire together the devices
- ❑ Cover chip with thick field oxide
- ❑ Etch oxide where contact cuts are needed

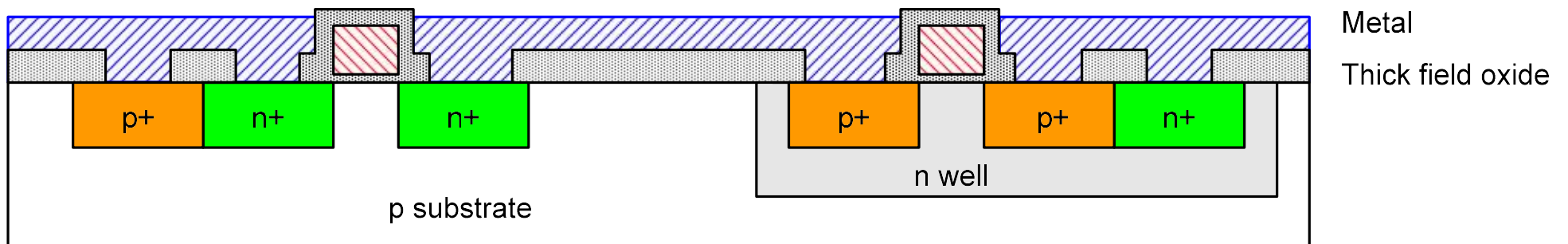


# Metalization

- ❑ Sputter on aluminum over whole wafer
- ❑ Pattern to remove excess metal, leaving wires



# Αντιστροφήας (κάτοψη)



# Βασική λογική σχεδίαση και άλγεβρα Boole

## Μαθηματική Λογική – Μαθηματική Συμπερασματολογία

Μαθηματικές προτάσεις – Λογικοί τελεστές

Οι βασικοί λογικοί (ή Boolean) τελεστές που χρησιμοποιούνται είναι οι: **NOT, AND, OR**.

**Πύλες** = Βασικά ψηφιακά λογικά στοιχεία που αντιστοιχούν στις βασικές λογικές συναρτήσεις.

**AND:** Λογικός πολ/σμός (σύζευξη)  
 **$A \cdot B$**

Boolean Έκφραση:

**OR:** Λογική πρόσθεση (διάζευξη)  
 **$A + B$**

Boolean Έκφραση:

**NOT:** Λογική αντιστροφή  
 **$\overline{A}$  ή  $\sim A$  ή  $A'$**

Boolean Έκφραση:

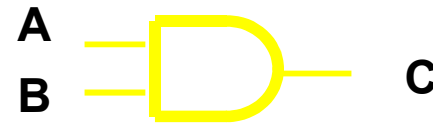
# Λογικός πολλαπλασιασμός (AND)

Έχει αποτέλεσμα = 1, μόνο εάν και οι δύο είσοδοι είναι 1.

TRUE=1, FALSE=0

$$C = A \cdot B \text{ ----- } C = A \text{ AND } B$$

Λογική Πύλη



Πίνακας αληθείας

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

Q: Ποια είσοδος (0/1) καθορίζει την έξοδο;

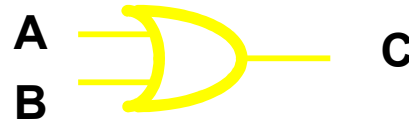
# Λογική πρόσθεση (OR)

Έχει αποτέλεσμα = 1, εάν κάποια από τις εισόδους ισούται με 1.

TRUE=1, FALSE=0

$$C = A + B \text{ ----- } C = A \text{ OR } B$$

Λογική Πύλη



Πίνακας αληθείας

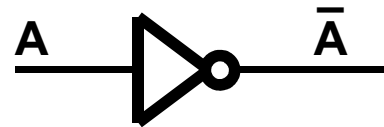
A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

Q: Ποια είσοδος (0/1) καθορίζει την έξοδο;

# Λογική αντιστροφή (NOT)

Έξοδος αντίστροφη της εισόδου

Λογική Πύλη



Πίνακας αληθείας

A	$\sim A$
0	1
1	0

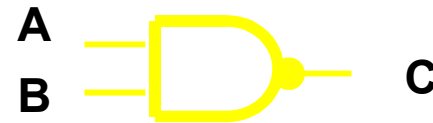
# NAND

Έχει αποτέλεσμα = 1, μόνο εάν και οι δύο είσοδοι είναι 1.

TRUE=1, FALSE=0

$$C = A \cdot B \text{ ----- } C = A \text{ AND } B$$

Λογική Πύλη



Πίνακας αληθείας

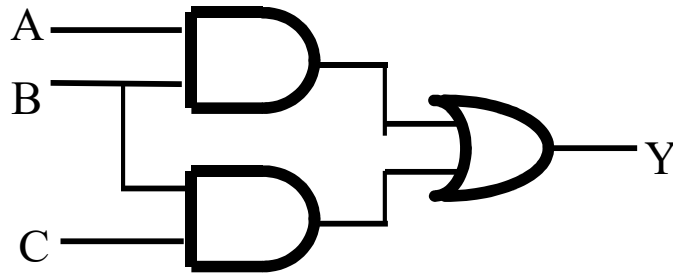
A	B	C
0	0	1
0	1	1
1	0	1
1	1	0

Q: Ποια είσοδος (0/1) καθορίζει την έξοδο;

Q: Ποια είσοδος (0/1) αντιστρέφει την άλλη είσοδο;

# Συνδυάζοντας Λογικές Πύλες

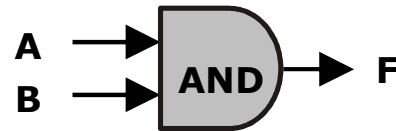
**Η έξοδος ενός συνδυαστικού κυκλώματος εξαρτάται μόνο από τις εκάστοτε εισόδους.**



**AND - OR Λογικό Κύκλωμα**

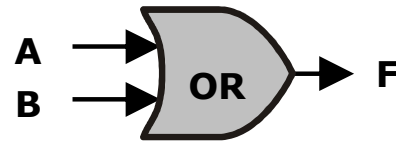
# Σύνοψη: Λογικά κυκλώματα - Πύλες

□ AND:  $F = A \cdot B$



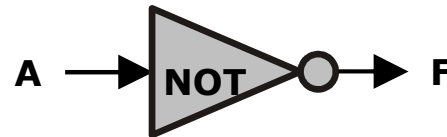
Η έξοδος F είναι 1 όταν **όλες** οι εισόδους (A, B) είναι 1

□ OR:  $F = A + B$



Η έξοδος F είναι 1 όταν **τουλάχιστον** μία είσοδος είναι 1

□ NOT:  $F = A'$



Η έξοδος είναι το αντίθετο της εισόδου

□ XOR:  $F = A' \cdot B + A \cdot B'$



Η έξοδος F είναι 1 όταν **μία και μόνο** μία είσοδος είναι 1

□ NAND:  $F = (A \cdot B)'$

□ NOR:  $F = (A + B)'$

□ XNOR:  $F = A \cdot B + A' \cdot B'$

# Πύλες: Άσκηση

Έστω η λογική συνάρτηση:

$$F = A \cdot B' + A' \cdot B \cdot C$$

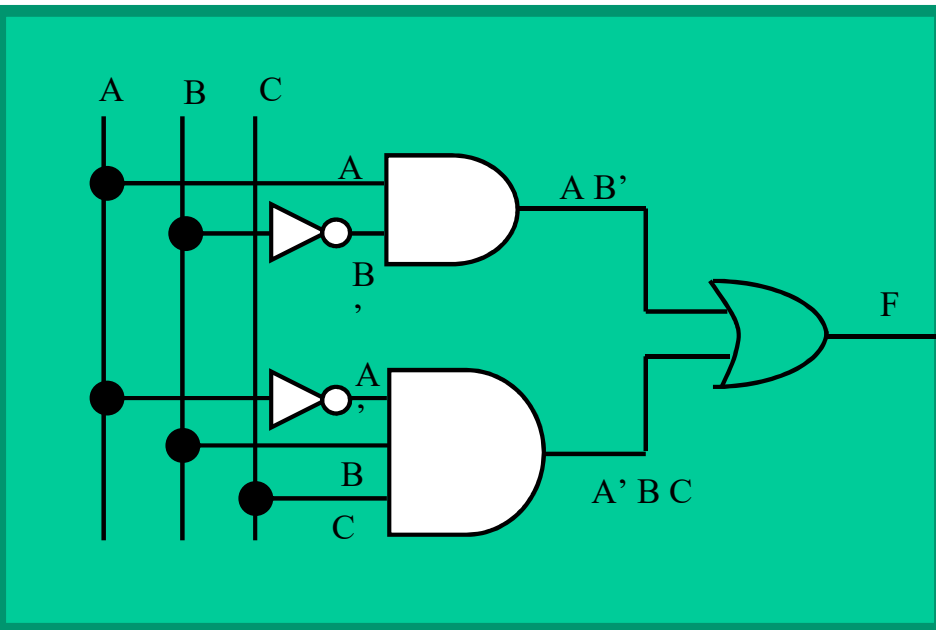
- 1) Γράψτε τη συνάρτηση, χρησιμοποιώντας τα ονόματα των πυλών (AND, OR, NOT)
- 2) Σχεδιάστε το αντίστοιχο λογικό κύκλωμα
- 3) Σχηματίστε τον πίνακα αληθείας της F

# Πύλες: Άσκηση

$$F = A \cdot B' + A' \cdot B \cdot C$$

$$F = (A \text{ AND } (\text{NOT } B)) \text{ OR } ((\text{NOT } A) \text{ AND } B \text{ AND } C)$$

Λογικό κύκλωμα



Πίνακας αληθείας

A	B	C	A'	B'	K = AB'	L = A'BC	F = K+L
0	0	0	1	1	0	0	0
0	0	1	1	1	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	0	0	1	1
1	0	0	0	1	1	0	1
1	0	1	0	1	1	0	1
1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0

# IC Packaging

## IC Packaging

- Ηλεκτρικές συνδέσεις με το chip για τη μεταφορά σήματος και ισχύος
- Προστασία από την υγρασία, τη σκόνη, τα αέρια και άλλες πιθανές προσμείξεις
- Απαγωγή θερμότητας

# Τεχνολογία IC

Τρεις τύποι τεχνολογίας IC:

Πλήρως παραμετροποιήσιμη/VLSI

Ημι-παραμετροποιήσιμη ASIC (Application-specific integrated circuit)

PLD (Programmable Logic Device) / FPGA

# Πλήρως παραμετροποιήσιμη/VLSI

- Όλα τα επίπεδα είναι βελτιστοποιημένα για ένα συγκεκριμένο ΕΣ
  - Τοποθέτηση και μέγεθος transistor
  - Καλωδίωση
- Οφέλη
  - Εξαιρετική απόδοση,
  - μικρό μέγεθος,
  - χαμηλή ισχύς
- Μειονεκτήματα
  - Υψηλό κόστος NRE (π.χ. \$ 300k),
  - Μεγάλο time-to-market

## Πλήρως παραμετροποιήσιμη/VLSI (συν.)

- Ο σχεδιαστής πρέπει να διαχειριστεί τα επιμέρους γεωμετρικά σχήματα τα οποία αντιπροσωπεύουν τα χαρακτηριστικά του κάθε τρανζίστορ για το τσιπ
- Μια απλή σχεδίαση με 3000 πύλες μπορεί να απαιτεί τη διαχείριση 300.000 ορθογωνίων ανά chip
- Η μέση ημερήσια παραγωγή ενός μηχανικού είναι λίγες δεκάδες τρανζίστορ

# Ημι-παραμετροποιήσιμη ASIC

- Τα χαμηλότερα στρώματα είναι πλήρως ή μερικώς υλοποιημένα
- Οι σχεδιαστές αναλαμβάνουν τη δρομολόγηση των καλωδίων
- Οφέλη
  - Καλές επιδόσεις,
  - καλό μέγεθος,
  - μικρότερο NRE από ό,τι μια πλήρως παραμετροποιήσιμη εφαρμογή (\$ 10k - \$ 100k)
- Μειονεκτήματα
  - Εξακολουθούν να απαιτούνται εβδομάδες έως μήνες για την ανάπτυξη

# PLD/FPGA

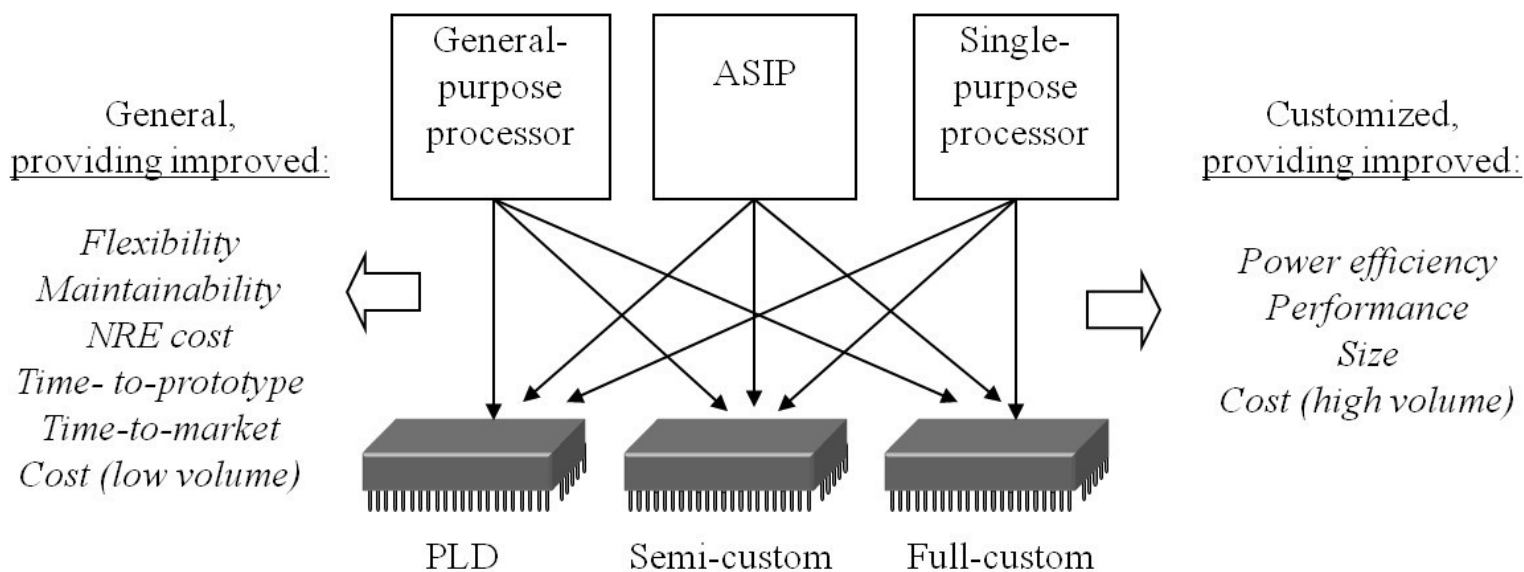
- Όλα τα στρώματα είναι ήδη υλοποιημένα
  - Οι σχεδιαστές αγοράζουν ένα IC
  - Οι διασυνδέσεις των IC πρέπει είτε να δημιουργηθούν είτε να καταστραφούν (αν ήδη υπάρχουν)
- Οφέλη
  - Χαμηλό NRE κόστος και
  - πρακτικά άμεση διαθεσιμότητα των IC
- Μειονεκτήματα
  - Μεγαλύτερο μέγεθος,
  - ακριβότερα (\$ 30/μονάδα),
  - μεγαλύτερη κατανάλωση και
  - πιο αργά

## PLD/FPGA (wikipedia)

- Historically, FPGAs have been slower, less energy efficient and generally achieved less functionality than their fixed ASIC counterparts.
- An older study had shown that designs implemented on FPGAs need on average 40 times as much area, draw 12 times as much dynamic power, and run at one third the speed of corresponding ASIC implementations.
- More recently, FPGAs such as the Xilinx Virtex-7 or the Altera Stratix 5 have come to rival corresponding ASIC and ASSP solutions by providing significantly reduced power, increased speed, lower materials cost, minimal implementation real-estate, and increased possibilities for re-configuration 'on-the-fly'.
- Where previously a design may have included 6 to 10 ASICs, the same design can now be achieved using only one FPGA.

# Επεξεργαστής και τεχνολογία IC

- Βασικοί συμβιβασμοί
  - Γενικό έναντι ειδικού
  - Όσον αφορά στην τεχνολογία επεξεργαστών ή την τεχνολογία IC
- Οι δύο επιλογές είναι ανεξάρτητες



# Χρονικοί περιορισμοί

## real-time συστήματα

Ένα σύστημα είναι πραγματικού χρόνου, εάν η συνολική ορθότητα μιας πράξης εξαρτάται όχι μόνο από τη λογική ορθότητα της, αλλά και από το χρόνο στον οποίο εκτελείται

**Hard:** Η απώλεια μιας προθεσμίας οδηγεί σε συνολική αποτυχία του συστήματος

**Soft:** Η χρησιμότητα ενός αποτελέσματος υποβαθμίζεται με το πέρας της προθεσμίας, υποβαθμίζοντας έτσι την ποιότητα των παρεχόμενων υπηρεσιών

## Reactive

Αντιδρά στις αλλαγές στο περιβάλλον του συστήματος και υπολογίζει αποτελέσματα σε πραγματικό χρόνο

# Παραδείγματα real-time ΕΣ

## Cruise controller οχήματος

- παρακολουθεί τους αισθητήρες ταχύτητας και αντιδρά με το φρένο
- υπολογίζει την επιτάχυνση ή επιβράδυνση μέσα σε περιορισμένο χρονικό διάστημα
- ένας καθυστερημένος υπολογισμός μπορεί να οδηγήσει σε αποτυχία διατήρησης του ελέγχου του οχήματος

## Έλεγχος αερόσακων:

- Τι κάνει;
- Πότε ανοίγουν οι αερόσακοι;
- Με ποια σειρά;
- Με ποια ταχύτητα;
- Πόσο χρόνο έχει για να κάνει τους υπολογισμούς;

# Λειτουργικά Συστήματα ΕΣ

Ανάλογα με τις ανάγκες της εφαρμογής το λογισμικό μπορεί να αλληλεπιδρά με το υλικό

## **απευθείας**

- υψηλότερη απόδοση,
- αλλά απαιτείται
  - γνώση της αρχιτεκτονικής που χρησιμοποιείται και
  - των περιφερειακών

## **μέσω Λειτουργικού Συστήματος**

παρέχει λειτουργίες για την πολυ-επεξεργασία και επιτρέπει στο σχεδιαστή να αναπτύξει εφαρμογές ως επί το πλείστον ανεξάρτητα από την υποκείμενη αρχιτεκτονική

## Βιβλιοθήκες/IP (Intellectual Property)

- Βιβλιοθήκες που επιτρέπουν την επαναχρησιμοποίηση υπαρχουσών υλοποιήσεων
- Αυξάνεται η παραγωγικότητα
- Σε επίπεδο λογικής μπορεί μια βιβλιοθήκη να αποτελείται από πύλες
- Σε επίπεδο RT (Register-transfer level) μπορεί να αποτελείται από καταχωρητές, πολυπλέκτες, αποκωδικοποιητές και άλλες δομικές μονάδες
- Σε επίπεδο περιγραφής συμπεριφοράς (behavioral-level) μπορεί να αποτελείται από bus interfaces, ελεγκτές οθόνης ακόμη και επεξεργαστές γενικής χρήσης
- Σε επίπεδο συστήματος (system-level) από ολοκληρωμένα συστήματα επίλυσης συγκεκριμένων προβλημάτων

# Επαλήθευση

Κάνει όντως ότι περιμένουμε;

Κάτω από οποιεσδήποτε συνθήκες;

Automotive electronics

Για μικρά σχέδια ή για επαλήθευση μόνο ορισμένων βασικών χαρακτηριστικών, η επαλήθευση είναι εύκολη

Για μεγαλύτερα ή περίπλοκα συστήματα είναι πολύ δύσκολη ή εκτινάσσει το κόστος

Συνήθως τα συστήματα ελέγχονται με προσομοίωση αντί για επαλήθευση

Οι προσομοιώσεις παρέχουν εκτεταμένο έλεγχο του σχεδιασμού και επιτρέπουν στο σχεδιαστή να παρατηρεί τις διεργασίες που συνήθως εκτελούνται μέσα σε ένα ΕΣ

Οι προσομοιώσεις μπορούν να χρησιμοποιηθούν για να αυξηθεί η εμπιστοσύνη στην ορθότητα και την πληρότητα του σχεδιασμού, **αλλά δεν μπορούν να την αποδείξουν**

# Συσχεδίαση Υλικού-Λογισμικού

- Κατά το παρελθόν:
  - Τεχνολογίες σχεδίασης υλικού και λογισμικού ήταν πολύ διαφορετικές
  - Πρόσφατη ωρίμανση της σύνθεσης επιτρέπει την ενιαία θεώρηση του υλικού και λογισμικού
- Συσχεδίαση Υλικού-Λογισμικού
- Τι προβλήματα δημιουργούνται όταν σχεδιάζω ταυτόχρονα υλικό και λογισμικό;
  - Τι θα συμβεί αν στο τέλος δε συναργάζονται;

# Συμπεράσματα

- Τα ΕΣ είναι παντού
  - η χρήση τους αυξάνει συνεχώς
- Πρόκληση η βελτιστοποίηση των σχεδιαστικών μετρικών
  - Οι μετρικές «ανταγωνίζονται» μεταξύ τους
- Ενιαία θεώρηση του υλικού και λογισμικού είναι αναγκαία για τη βελτίωση της παραγωγικότητας
- Βασικές τεχνολογίες
  - **Επεξεργαστής**: γενικής χρήσης, για συγκεκριμένες εφαρμογές και ειδικού σκοπού
  - **IC**: Full-custom, semi-custom, PLD

Ερωτήσεις;

