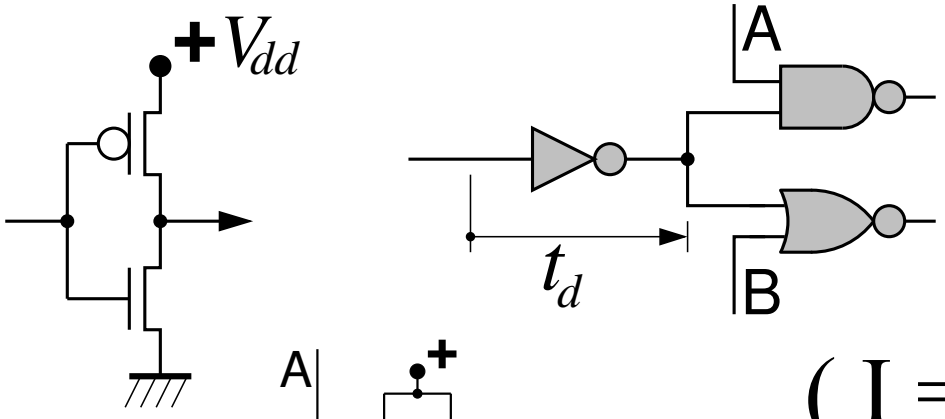
Ταχύτητα και Κατανάλωση Ενέργειας των Κυκλωμάτων CMOS

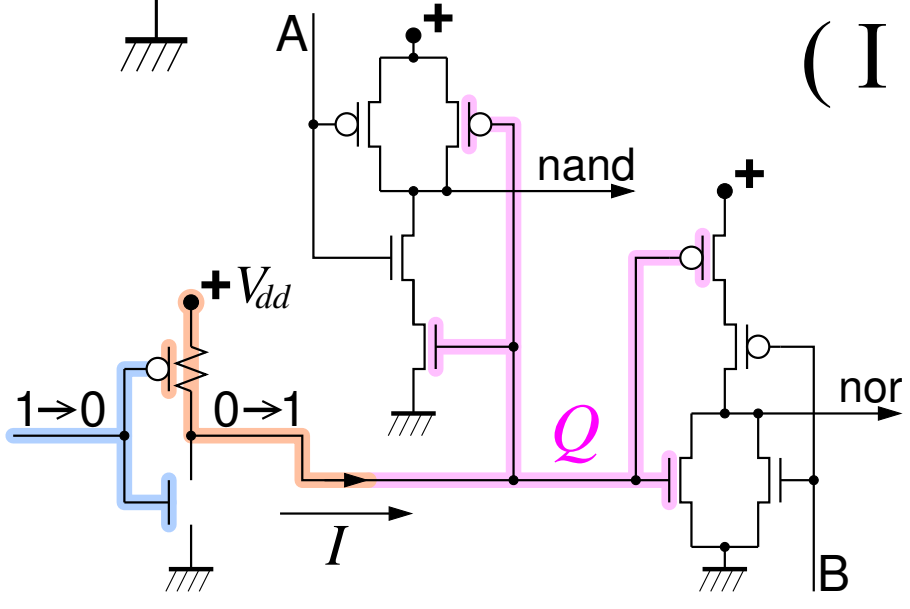
12b (§12.8) – 14-18 Δεκ. 2020 – Μανόλης Κατεβαίνης

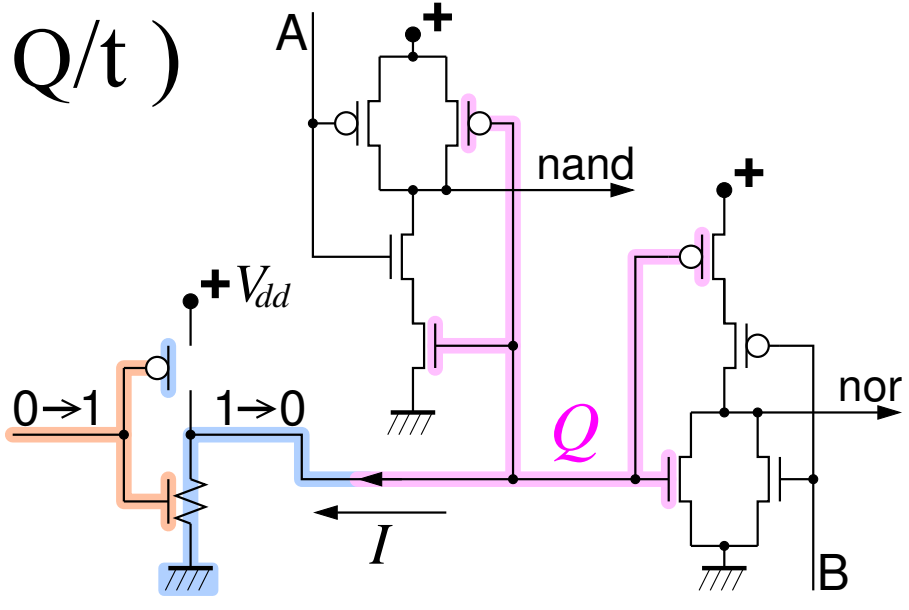
Καθυστέρηση: παροχή/απορρόφηση Φορτίου



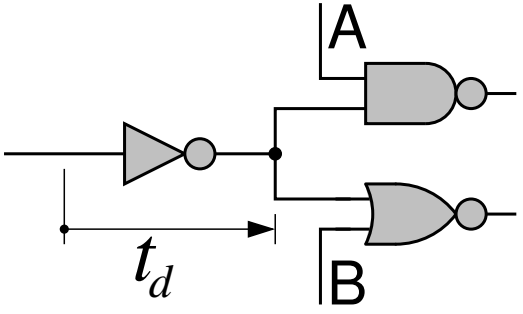
$$t_d \approx Q / I$$

(I = Q/t)

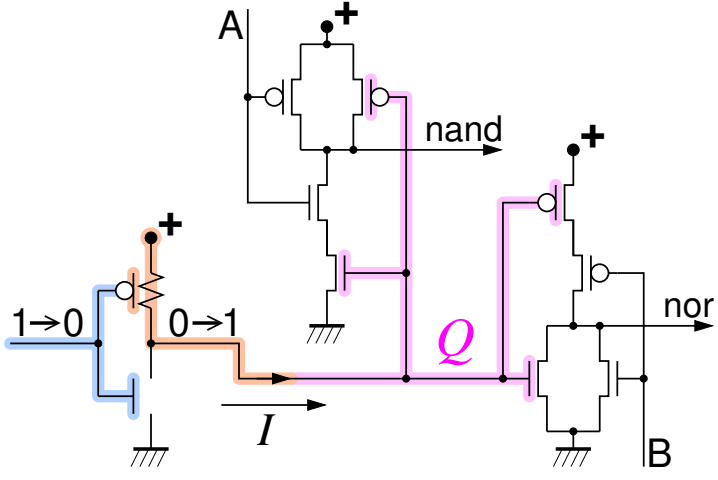




Καθυστερήσεις σε CMOS: πόσο Φορτίο, με τι Ρεύμα



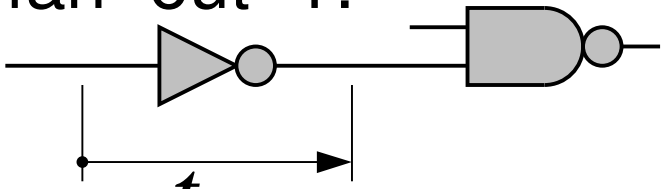
$$t_d \approx Q / I$$



- Οι μεγαλύτερες καθυστερήσεις σε CMOS οφείλονται στο χρόνο παροχής/απορρόφησης φορτίου
  - στα transistors των ακροατών
  - συνήθως 2 transistors ανά ακροατή
- Το περισσότερο φορτίο απαιτείται στις πύλες των transistors
  - λεπτό οξείδιο γιά έλεγχο καναλιού
  - ⇒ μεγάλη χωρητικότητα (C = Q/V)
- Ρεύμα περνά μόνο μέχρι να φορτιστούν/εκφορτ. οι ακροατές

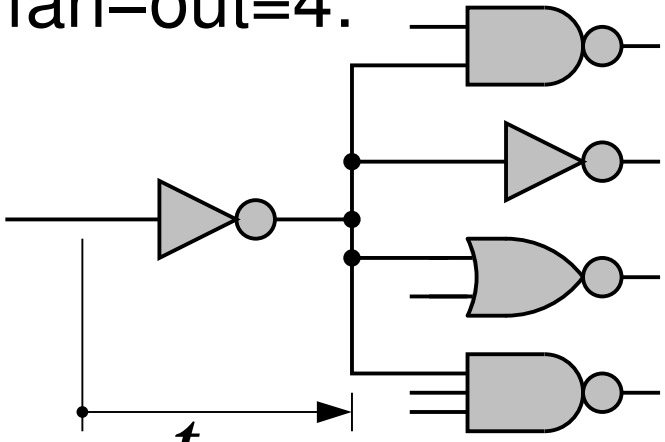
Καθυστέρηση ~ Fan-Out (Πλήθος Ακροατών)

fan–out=1:



t_{d1}

fan–out=4:



t_{d4}

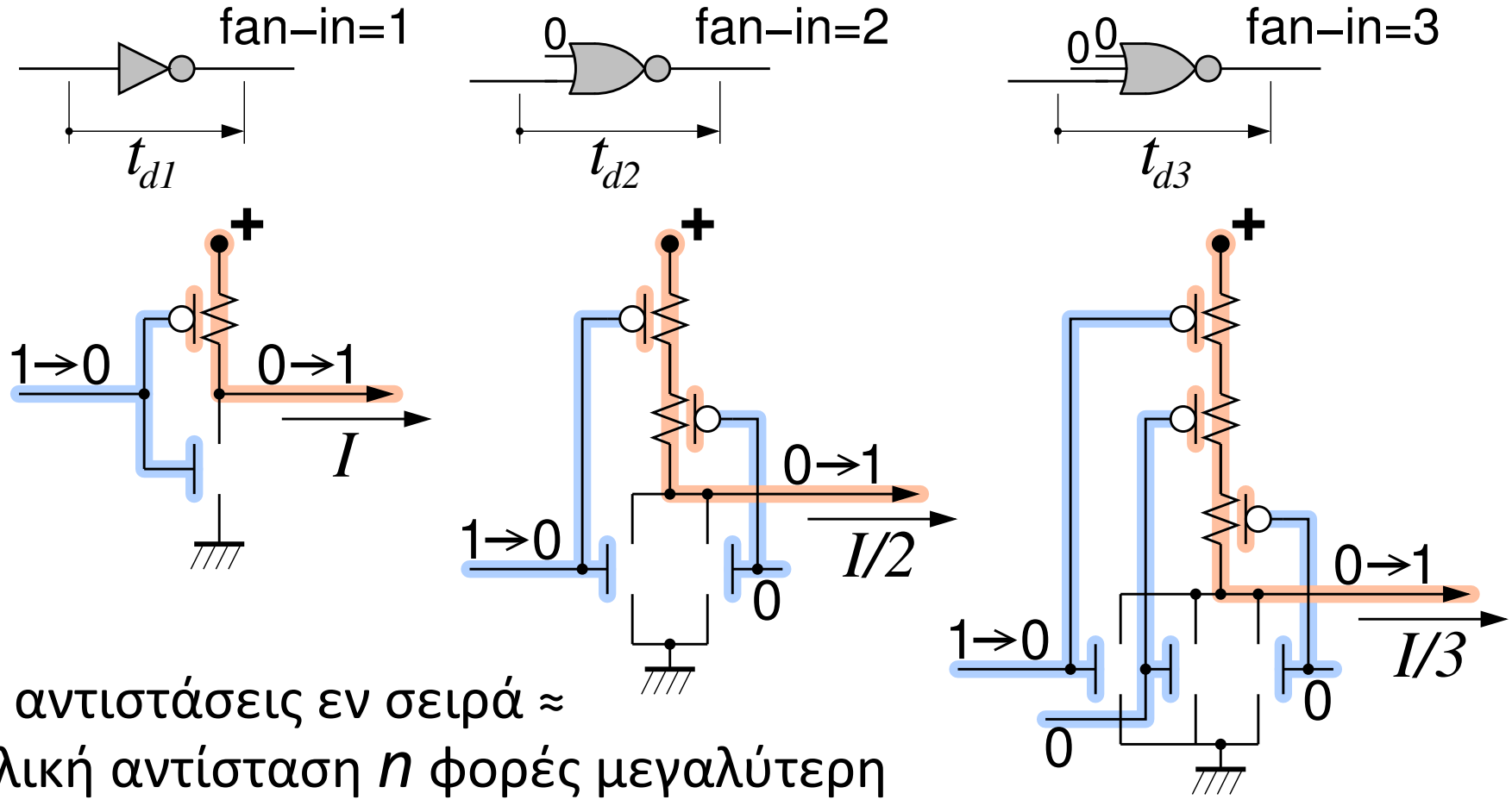
- Η κάθε είσοδος ακροατή:
  - συνήθως 2 transistors, ανεξαρτή-των πλήθους εισόδων ακροατή
- Τα δύο transistors κάθε ακροατή χρειάζονται το φορτίο τους
- Ολικό φορτίο ~ πλήθος ακροατών
- Καθυστέρηση ~ ολικό φορτίο

⇒ Καθυστέρηση ~ *Fan-out*

- fan-out = πλήθος ακροατών

$$\Rightarrow t_{d4} \approx 4 \times t_{d1}$$

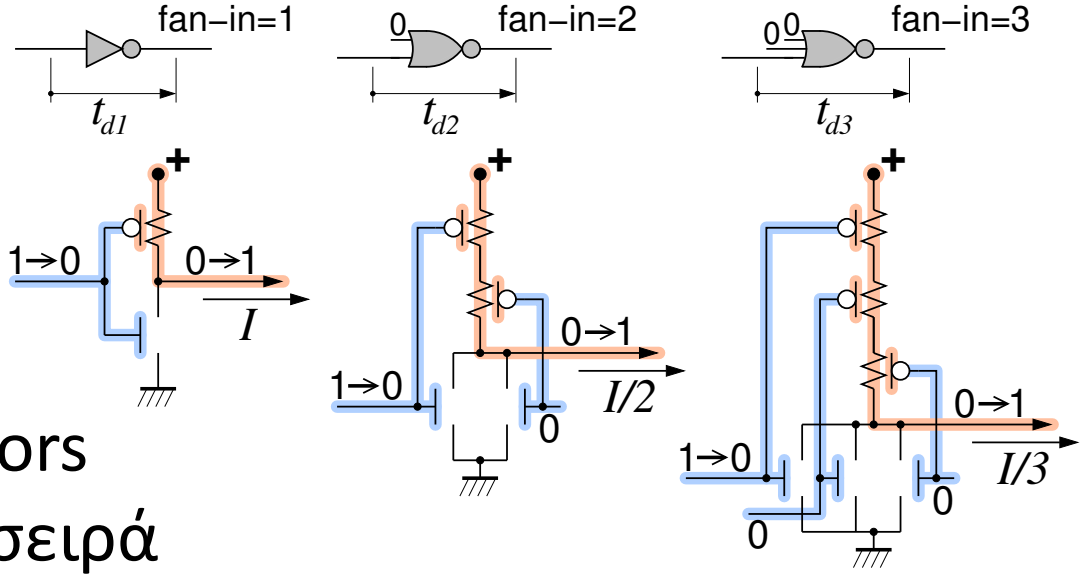
Καθυστέρηση ~ Fan-In (Πλήθος Εισόδων)



n αντιστάσεις εν σειρά ≈
ολική αντίσταση *n* φορές μεγαλύτερη

Καθυστέρηση ~ Fan-In (Πλήθος Εισόδων)

- Σε πρώτη προσέγγιση, το ρεύμα φόρτισης/ εκφόρτισης είναι αντίστροφα ανάλογο προς το πλήθος transistors οδήγησης που είναι εν σειρά



- $t_d \approx Q / I$

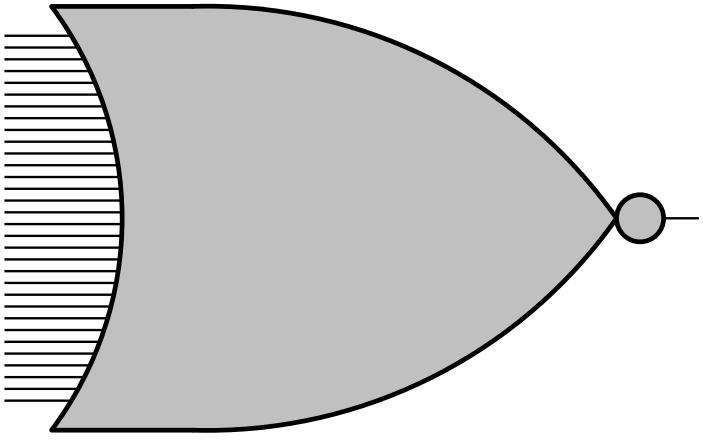
$\Rightarrow$ μικρότερο ρεύμα $\Rightarrow$ μεγαλύτερη καθυστέρηση

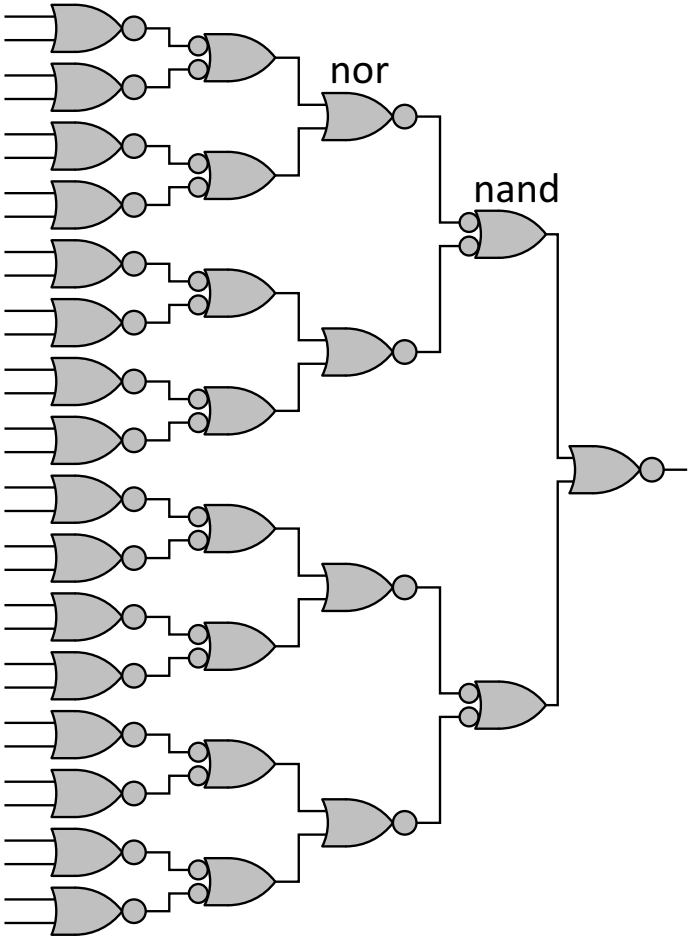
$\Rightarrow$ Καθυστέρηση (περίπου) ανάλογη (και) προς fan-in

Δένδρα γιά λογαριθμική αντί γραμμικής καθυστέρησης

- Π.χ. ανίχνευση μηδενός γιά 32-μπιτη λέξη
- Πύλη NOR με fan-in 32 θα είχε καθυστέρηση 32 φορές την «βασική» καθυστέρηση
- Ισοδύναμο δένδρο πυλών με fan-in=2 θα έχει καθυστέρηση 5 επίπεδα επί 2 φορές την «βασική» καθυστ. καθένα

⇒ Καθυστέρηση ~ λογάριθμος fan-in, αντί γραμμικά γιά μονολιθική πύλη

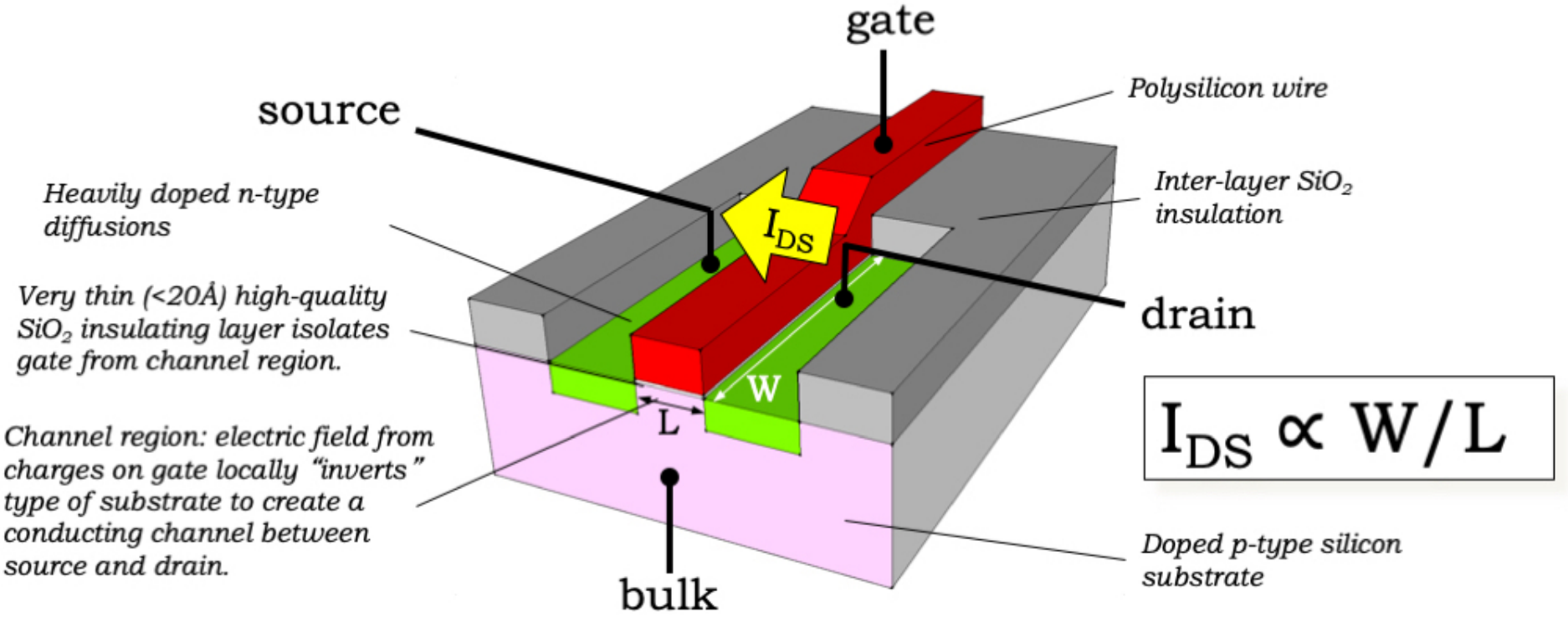




Όσο απλούστερα & μικρότερα, τόσο πιό γρήγορα!

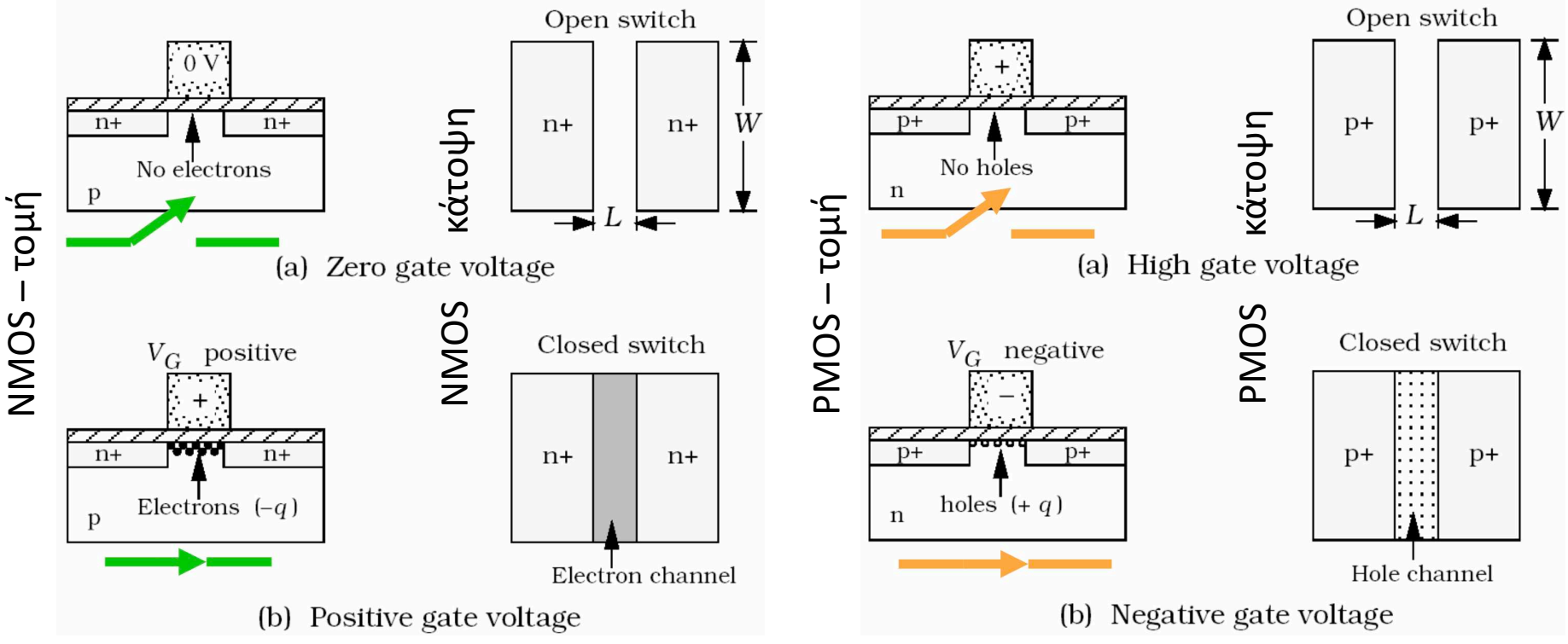
- Συμπέρασμα: *Απλά και μικρά κυκλώματα*, γιά ταχύτητα!
- Λιγότεροι ακροατές (fan-out) $\Rightarrow$ απλούστερο
- Λιγότερες είσοδοι (fan-in) $\Rightarrow$ απλούστερο
- Λιγότερες πύλες, ακροατές, είσοδοι $\Rightarrow$ μικρότερο
- Τη δεκαετία του ’80, αυτή η παρατήρηση υπήρξε η απαρχή του «κινήματος» *RISC*: Reduced Instruction Set Computer – Υπολογιστές ελαττωμένου (απλούστερου) Ρεπερτορίου Εντολών, που σήμερα είναι σημαντικά δημοφιλείς (και θα είναι το παράδειγμά μας στο HY-225)

Φαρδύτερα transistors οδηγούν μεγαλύτερο ρεύμα



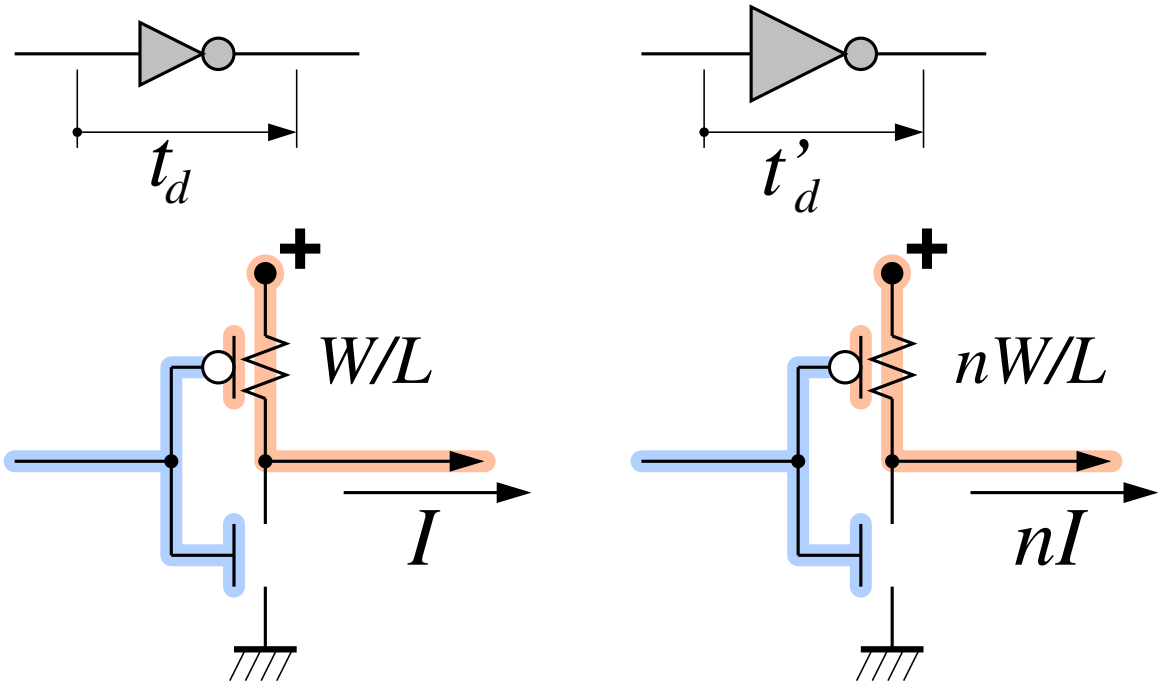
- Μήκος καναλιού, L: το ελάχιστο που επιτρέπει η τεχνολογιά κατασκ.
- Πλάτος καναλιού, W: το αυξάνουμε όταν θέλουμε μεγαλύτερο I

Φαρδύτερα transistors οδηγούν μεγαλύτερο ρεύμα



- Μήκος καναλιού, L: το ελάχιστο που επιτρέπει η τεχνολογιά κατασκ.
- Πλάτος καναλιού, W: το αυξάνουμε όταν θέλουμε μεγαλύτερο I

Φαρδύτερα xtors οδηγούν ταχύτερα τον επόμενο

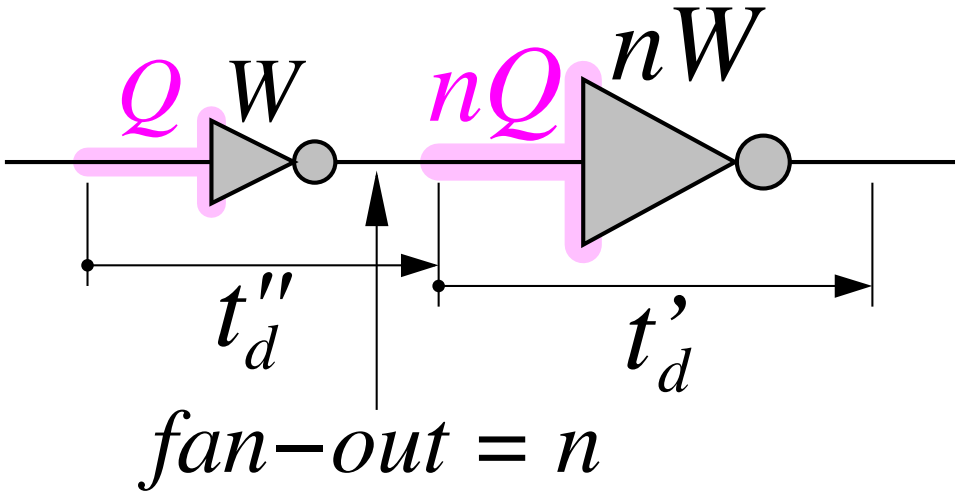


$$t_d \approx Q / I$$

- Αυξάνοντας το W αυξάνουμε αναλογικά το I, άρα μειώνουμε την καθυστέρηση…
- Μόνο που… (βλ. επόμενη διαφάνεια)

- Μήκος καναλιού, L: το ελάχιστο που επιτρέπει η τεχνολογιά κατασκ.
- Πλάτος καναλιού, W: το αυξάνουμε όταν θέλουμε μεγαλύτερο I

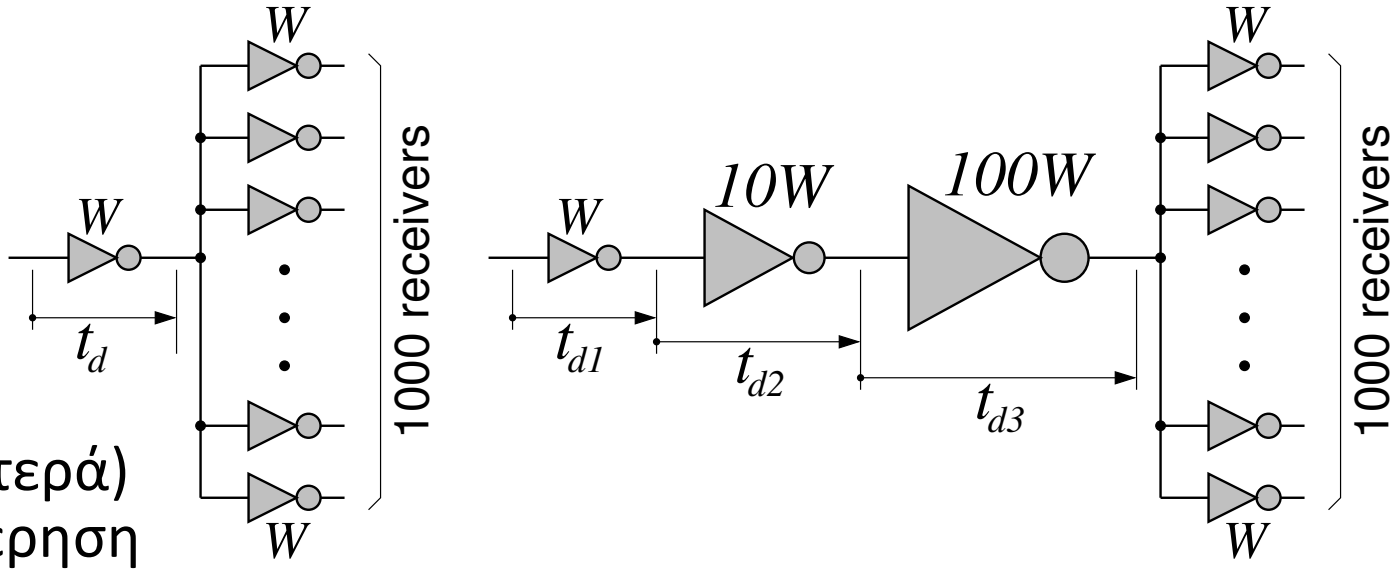
Φαρδύτερα xtors επιβραδύνουν τον προηγούμενο



- Το L όλων των transistors είναι ίδιο: το ελάχιστο δυνατό
- Αυξάνοντας το W, γιά να αυξήσουμε το ρεύμα, αυξάνουμε και το εμβαδόν της πύλης του transistor, άρα και το φορτίο που αυτή χρειάζεται
- Κάνοντας μιά πύλη ισχυρότερη, μέσω αυξημένου W, γιά να γίνει αυτή πιό γρήγορη, την κάνουμε και «βαρύτερη» στην οδήγηση από την προηγούμενη πύλη, δηλαδή η προηγούμενη είναι τώρα σαν να αποκτά fan-out ισότιμο προς n «βασικές» πύλες

Αλυσίδα Οδηγητών γιά Βαριά Φορτία

- Π.χ. το σήμα του ρολογιού (clock) οδηγεί πάρα πολλούς ακροατές
- Ένας, απλός οδηγητής (αριστερά) θα είχε καθυστέρηση $t_d \approx$ 1000 «βασικές» καθυστερήσεις, διότι έχει fan-out = 1000



Αλυσίδα οδηγητών (δεξιά) με αύξοντα μεγέθη transistors:

- $t_{d1} \approx$ 10 «βασικές» καθυστερήσεις: ρεύμα I, φορτίο $10 \cdot W$
- $t_{d2} \approx$ 10 «βασικές» καθυστερήσεις: ρεύμα $10 \cdot I$, φορτίο $100 \cdot W$
- $t_{d3} \approx$ 10, ομοίως ($1000 \cdot W$ / $100 \cdot I$) $\Rightarrow$ Συνολική καθ. 30 αντί 1000 (!)

Buses: καθυστέρηση ανάλογη πλήθους οδηγητών

- n οδηγητές, καθένας με transistors πλάτους W
- Συνολικό φορτίο στο bus ανάλογο του n (πλήθος xtors)
- Ένας οδηγητής αναμένος $\Rightarrow$ ένα transistor μεγέθους W οδηγεί φορτίο ανάλογο του n $\Rightarrow$ καθυστ. ανάλογη του n
- Αύξηση πλάτους των transistors δεν οφελεί:
  - εάν ξέραμε ποιό απ' όλα είναι (πάντα) ο οδηγητής, θα φαρδαίναμε αυτό και μόνον, και άρα θα επιταχύνονταν η λεωφόρος
  - επειδή οι οδηγητές εναλλάσονται, θα έπρεπε να τους φαρδύνουμε όλους, αλλά τότε αυξάνει κατ' αναλογία και το συνολικό φορτίο πάνω σε όλα τα transistors, άρα δώρον άδωρο

Μνήμες: όσο μεγαλύτερη, τόσο πιό αργή

- Μνήμη (εσωτερικά) = μεγάλος πολυπλέκτης μέσω bus

$\Rightarrow$ καθυστέρηση αυξάνει με το πλήθος flip-flops στο bus

- Ευτυχώς, λιγότερο από γραμμικά με το πλήθος των bits:
  1. Διάσταση ενός μπλόκ μνήμης ανάλογη προς την ρίζα του πλήθους των bits στο μπλόκ (περίπου τετράγωνο σχήμα)
  2. Κάθε chip μνήμης περιέχει πολλά, μικρότερα μπλοκς
  - Κάθε μικρότερο μπλόκ έχει καθυστέρηση ανάλογα μικρότερη
  - Ανάγνωση από τη συνολική μνήμη = πολυπλέκτης από τα μπλοκς: αυτός ο πολυπλέκτης είναι σχετικά μικρότερος και με σχετικά γρηγορότερες πύλες
- Δένδρο πολυπλεκτών $\Rightarrow$ λογαριθμική καθυστ. (όπως sl.7)

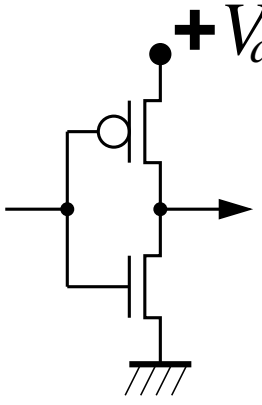
Πρόοδος Τεχνολογίας: μείωση του L

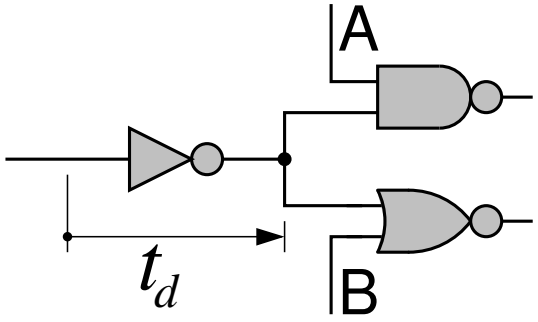
- "Technology Node": η ελάχιστη διάσταση ή απόσταση που μπορεί να κατασκευαστεί πάνω στο chip
  - 10μm (1971), 1μm (1984), 130nm (2001), 14 ή 10 ή 7nm τώρα
- Γιά μήκος καναλιού, L, θέλουμε το ελάχιστο δυνατό
- Μειώνοντας τις διαστάσεις, L (και W), κερδίζουμε σε όλα:
  - Περισσότερα transistors / chip (με το τετράγωνο της διάστασης)
  - Εμβαδόν πύλης transistor = $W \times L$ μικραίνει με το τετράγ. διαστ.
  - Μικρότερο εμβαδόν πύλης $\Rightarrow$ μικρότερο φορτίο στην πύλη
  - Μικρότερο φορτίο με ίδιο ρεύμα δίνει μεγαλύτερη ταχύτητα (το ρεύμα είναι ανάλογο προς το W/L)

Κατανάλωση Ενέργειας: ο νέος περιοριστικός παραγ.

- Τον 20ο αιώνα, περιοριστικός παράγοντας γιά την ταχύτητα των chips ήταν η τεχνολογία κατασκευής
- Τον 21ο αιώνα, περιοριστικός παράγοντας ταχύτητας έγινε η μη υπερθέρμανση των chips!
  - η σύγχρονη τεχνολογία επιτρέπει ρολόγια των 10 ή 20 GHz, αλλά ένας επεξεργαστής με τέτοιο ρολόϊ θα έλιωνε λόγω υπερθέρμανσης από τα μεγάλα ρεύματα και τάσεις που απαιτούνται γιά να δουλέψει σε τόσο ψηλές ταχύτητες
- Τα μεγάλα Κέντρα Δεδομένων (& Υπερυπολογιστές) καταναλώνουν πάρα πολλή ενέργεια: περιβαλλοντικό πρόβλημα

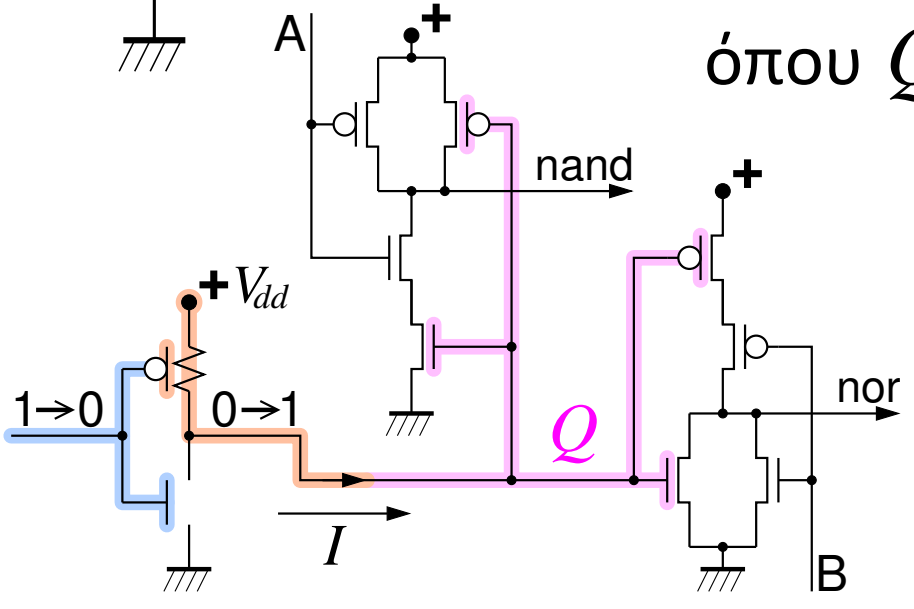
Ενέργεια = κίνηση Φορτίου × Διαφορά Δυναμικού

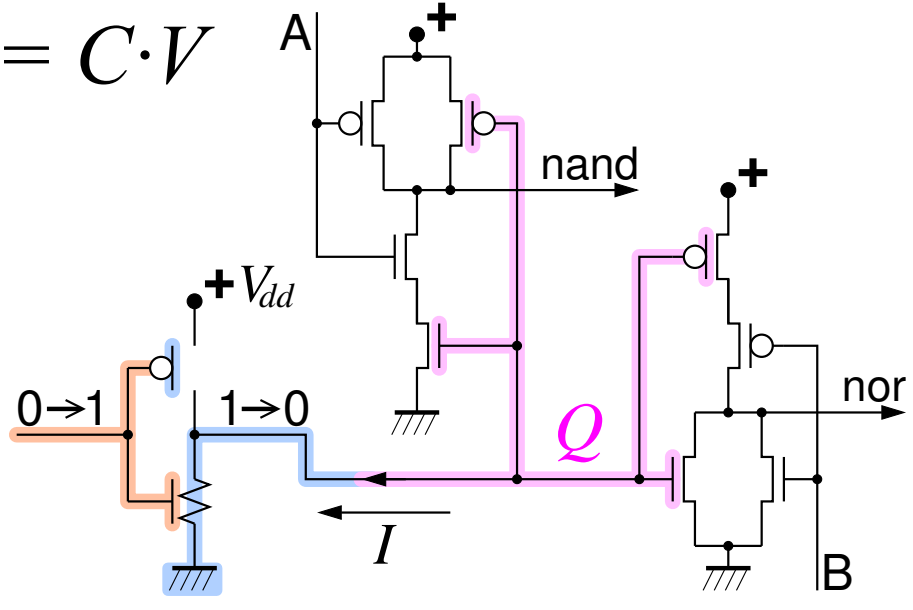




$$E = Q \cdot V_{dd}$$

όπου $Q = C \cdot V$



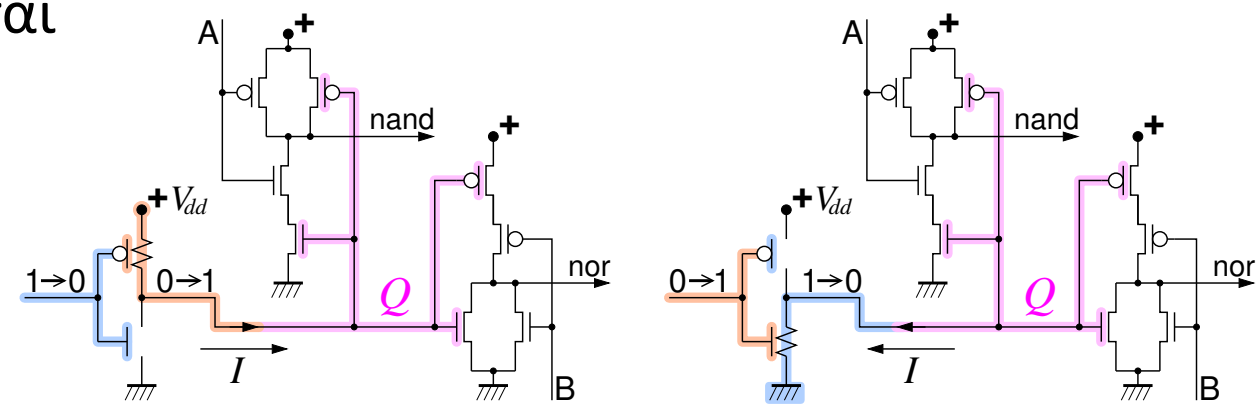


Κάθε ανεβοκατέβασμα 0→1→0: Ενέργεια = $C \cdot V_{dd}^2$

- Ενέργεια = Δύναμη × Απόσταση = Φορτίο × Ένταση Πεδίου × Απόσταση = Φορτίο × Διαφορά Δυναμικού
- Φορτίο = Χωρητικότητα × Τάση
  - Χωρητικότητα = διηλεκ.σταθ. × Εμβαδόν Πύλης / πάχος διηλεκτρ.

⇒ Ενέργεια = Χωρητικότητα × Τάση Τροφ. *στο Τετράγωνο*

- Σε κάθε άναμα και σβήσιμο ενός κόμβου μέσα στο chip, τόση ενέργεια μετατρέπεται σε θερμότητα στις αντιστάσεις των transistors ανέλκυσης & καθέλκυσης



Ολική Ενέργεια = $\Sigma_{κόμβοι}\ C \cdot V_{dd}^2 \cdot Πλήθος\ Ανεβοκατεβ.$

- Ολική Ενέργεια απαιτούμενη γιά έναν υπολογισμό:
- Πόσοι ηλεκτρικοί κόμβοι θα ανεβοκατέβουν;
  - μόνον όταν αλλάζει η τιμή (τάση) ενός κόμβου καταναλ. ενέργεια
  - πολλοί κόμβοι δεν αλλάζουν τιμή σε κάθε κύκλο ρολογιού
- Πόσο πολλές φορές θα ανεβοκατέβουν αυτοί;
  - πόση «εργασία» χρειάζεται ο υπολογισμός που θα γίνει
- Πόση ηλ. χωρητικότητα έχουν αυτοί οι κόμβοι;
  - συνάρτηση fan-out, πλάτους transistors, ελαχ. διαστ. τεχνολογίας
- Μειώστε την Τάση Τροφοδοσίας!!
  - Ταχύτητα ανάλογη V_{dd}, ενέργεια ανάλογη V_{dd}^2

Ενεργειακό κόστος– ταχύτητα – παραλληλισμός

- Μειώστε την Τάση Τροφοδοσίας!!
  - Ταχύτητα ανάλογη V_{dd}, ενέργεια ανάλογη V_{dd}^2
- Εάν η απάντηση δεν με επείγει: μείωση V_{dd} γιά να μειωθεί το ενεργειακό κόστος του επιθυμητού υπολογισμού
- Εάν η απάντηση με επείγει:
- Εάν μπορώ να παραλληλοποιήσω τον υπολογισμό, τρέχω τα παράλληλα κομμάτια σε πολλαπλούς επεξεργαστές
- Εάν δεν μπορώ να παραλληλοποιήσω τον υπολογισμό: ανεβάζω το V_{dd} και θα μου κοστίσει παραπάνω ενέργεια!