

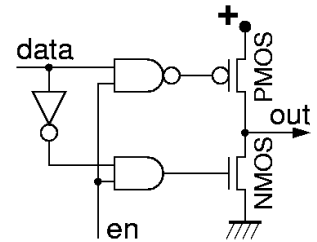
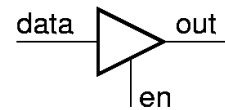
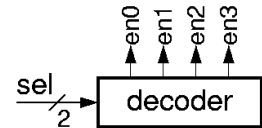
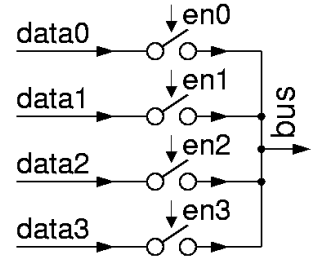
## Εργαστήριο 9: Τρικατάστατοι Οδηγητές, Λεωφόροι, Μνήμες SRAM

29 Νοεμβρίου - 3 Δεκεμβρίου 2010

[Βιβλία: **πρέπει** να διαβάσετε τις παραγράφους **10.3.1, 10.3.2** (σελ. 1021-1025) του βιβλίου του Wakerly, και τις σελίδες 371-375 της **2ης έκδοσης** του βιβλίου του Mano. Προαιρετικά μπορείτε επίσης να διαβάσετε: Wakerly: τα υπόλοιπα θέματα από § 10.2 έως και 10.3.4 (σελ. 1020-1030)· Mano (4η έκδοση): § 7.2, 7.3 (σελ. 285-294, εκτός 288-289)].

### 9.1 Πολύπλεξη μέσω Τρικατάστατων Οδηγητών:

Στην §2.6 είδαμε ότι οι πολυπλέκτες αποτελούν βασικότατο δομικό λίθο των ψηφιακών συστημάτων. Στην §4.9 είδαμε πώς φτιάχνεται ένας πολυπλέκτης από πύλες, ενώ στο πείραμα 2.7 είχαμε δει έναν πολυπλέκτη φτιαγμένο με ηλεκτρονόμους. Στο σχήμα δεξιά φαίνεται μία άλλη υλοποίηση πολυπλέκτη με διακόπτες: η διαφορά έχει ως εξής. Για έναν πολυπλέκτη  $n$ -σε-1, το κύκλωμα του πειράματος 2.1 δέχεται  $\log n$  εισόδους επιλογής, τις οποίες και αποκωδικοποιεί στους  $n$  συνδυασμούς τους προκειμένου να αποκαταστήσει έναν αγωγίμο δρόμο μεταξύ της εξόδου και μίας και μόνο εισόδου. Για έναν ίδιο πολυπλέκτη  $n$ -σε-1, το κύκλωμα δεξιά δέχεται  $n$  εισόδους επιλογής,  $en_0, \dots, en(n-1)$ , οι οποίες πρέπει να είναι **ήδη** αποκωδικοποιημένες, δηλαδή ακριβώς μία από αυτές πρέπει να είναι ενεργή (1) και οι υπόλοιπες αδρανείς (0).



Όταν οι πηγές πληροφοριών,  $data_0, data_1$ , κλπ, βρίσκονται σχετικά μακριά ή μία από την άλλη, ή όταν είναι επιθυμητή η προσθαφαίρεση πηγών κατά τη λειτουργία του συστήματος (π.χ. προσθήκη νέας κάρτας σε υπολογιστή), τότε η υλοποίηση του πολυπλέκτη μέσω διακοπών είναι προτιμότερη της υλοποίησης μέσω πυλών. Η πρώτη αιτία είναι το πλήθος και το κόστος των απαιτούμενων συρμάτων: το κύκλωμα με πύλη OR απαιτεί να συλλεγούν οι πληροφορίες από όλες τις πηγές σε ένα κεντρικό σημείο, πράγμα που χρειάζεται τόσα σύρματα όσες και οι πηγές· αντίθετα, το κύκλωμα με διακόπτες λειτουργεί με ένα μόνο σύρμα που διατρέχει το σύστημα απ' άκρη σ' άκρη μεταφέροντας κάθε φορά την πληροφορία της όποιας πηγής μας ενδιαφέρει τη στιγμή εκείνη. (Ο αποκωδικοποιητής και τα σύρματα επιλογής είναι ένα επιπλέον κόστος, αλλά όταν υπάρχουν πολλαπλά σύρματα (bits) δεδομένων το κόστος της επιλογής αποσβέννεται μεταξύ όλων αυτών· επίσης, υπάρχει πλήθος εφαρμογών όπου η επιλογή γίνεται με άλλους, καταναμημένους τρόπους). Το δεύτερο πλεονέκτημα της πολύπλεξης μέσω διακοπών είναι η ευκολία προσθήκης νέων πηγών πληροφορίας: εάν κάθε πηγή έχει ενσωματωμένο και το δικό της διακόπτη εξόδου, τότε η προσθήκη αυτή γίνεται μέσω απλής σύνδεσης συρμάτων --π.χ. με την εισαγωγή μίας κάρτας σε μάν υποδοχή-πρίζα.

Η παραπάνω λογική συμπεριφορά της πηγής πληροφορίας με τον διακόπτη στην έξοδο υλοποιείται στην τεχνολογία CMOS με το κύκλωμα που φαίνεται στο κάτω μέρος του σχήματος. Το κύκλωμα αυτό μπορεί να φέρει την έξοδο  $out$  σε μία από τρεις διαφορετικές καταστάσεις: μηδέν (0), ένα (1), ή "**υψηλή αντίσταση**" (**HiZ** - high impedance - υψηλή εμπέδηση). Λόγω των τριών καταστάσεων της εξόδου του, το κύκλωμα αυτό ονομάζεται **τρικατάστατος οδηγητής (tri-state driver)**. Η κατάσταση 0 αντιστοιχεί σε ενεργοποιημένο διακόπτη ( $en=1$ ) και σε πληροφορία 0, οπότε η έξοδος

πρέπει να οδηγηθεί σε χαμηλή τάση, ανάβοντας το transistor NMOS. Η κατάσταση 1 αντιστοιχεί σε ενεργοποιημένο διακόπτη ( $en=1$ ) και σε πληροφορία 1, οπότε η έξοδος πρέπει να οδηγηθεί σε υψηλή τάση, ανάβοντας το transistor PMOS. Η κατάσταση HiZ αντιστοιχεί σε αδρανή (σβηστό) διακόπτη ( $en=0$ ), οπότε η έξοδος πρέπει να μείνει ασύνδετη --να μην τοποθετηθεί καμία πληροφορία σε αυτήν, αφήνοντας άλλες πηγές να την οδηγήσουν· σε αυτή την περίπτωση πρέπει και τα δύο transistors, NMOS και PMOS, να παραμείνουν σβηστά. Οι πύλες NAND, NOT, και AND που φαίνονται στο σχήμα προφανώς επιτυγχάνουν ακριβώς τη λειτουργία αυτή· επίσης εξασφαλίζουν ότι σε καμία περίπτωση δεν ανάβουν ταυτόχρονα και τα δύο transistors, NMOS και PMOS, αφού κάτι τέτοιο και απροσδιόριστη τάση εξόδου θα έδινε, και υπερβολικό ρεύμα τροφοδοσίας θα ξόδευε, και υπερθέρμανση των transistors θα προκαλούσε.

Ένα σύρμα στο οποίο συνδέονται πολλαπλοί τρικατάστατοι οδηγητές προκειμένου να υλοποιηθεί πολύπλεξη των αντίστοιχων δεδομένων εισόδου ονομάζεται "**λεωφόρος**" (**bus**), ή "αρτηρία" ή "διάδρομος" (ή καμιά φορά "δίαυλος"). Ο κανόνας λειτουργίας μιάς λεωφόρου είναι ότι, ανά πάσα στιγμή, **το πολύ ένας** από τους οδηγητές που συνδέονται σε αυτήν επιτρέπεται να είναι ενεργοποιημένος· ο οδηγητής αυτός προσδιορίζει την τάση (λογική τιμή) της λεωφόρου. Όποτε δεν υπάρχει κανείς ενεργοποιημένος οδηγητής, η λογική τιμή της λεωφόρου είναι *απροσδιόριστη*, δηλαδή η πληροφορία πάνω της είναι "σκουπίδια" ή "θόρυβος" (σε ειδικές περιπτώσεις τεχνολογιών μπορεί να είναι προβλέψιμη η τιμή αυτή, όπως συμβαίνει με την "δυναμική αποθήκευση πληροφορίας" (δυναμική μνήμη) μέσα σε chips τεχνολογίας CMOS, όμως αυτά δεν αφορούν το παρόν μάθημα). Μία υδραυλική αναλογία της λεωφόρου είναι ένας σωλήνας που διατρέχει πολλά διαμερίσματα, και τον οποίο μπορούν να τροφοδοτούν βρύσες από κάθε διαμέρισμα. Ένας τρικατάστατος οδηγητής είναι μιά από αυτές τις βρύσες, αλλά με χωριστό ρομπινέτο ζεστού και κρύου νερού. Για να μεταδώσουμε πληροφορία "0" ανοίγουμε το κρύο ρομπινέτο· για να μεταδώσουμε "1" ανοίγουμε το ζεστό· για να μη μεταδώσουμε τίποτα, αφήνοντας κάποιον άλλον να μεταδώσει, κλείνουμε εντελώς και τα δύο ρομπινέτα μας. Εάν μεταδίδω ταυτόχρονα κι εγώ (π.χ. "1") και κάποιος άλλος (π.χ. "0"), τότε ο σωλήνας θα έχει "χλιαρό" νερό, που δεν είναι ούτε 0 ούτε 1· οπότε δεν μεταδίδει κανείς πληροφορία, η θερμοκρασία του σωλήνα είναι τυχαία, εξαρτώμενη από άλλους παράγοντες (π.χ. χειμώνας/καλοκαίρι).

Η "αρχιτεκτονική λεωφόρου" (bus) αποτελεί την απλούστερη μορφή δικτύου επικοινωνίας (communication network), και χρησιμοποιείται ευρύτατα όποτε η επικοινωνία γίνεται με ένα "κοινόχρηστο" μέσο (shared medium). Τέτοιο κοινόχρηστο μέσο μπορεί να είναι ένα σύρμα (όπως στην "κλασσική" μορφή του δικτύου "Ethernet" --όχι "switched ethernet"), μία ραδιοσυχνότητα σε δεδομένη περιοχή (π.χ. τα ραδιοταξί μιάς πόλης ή συνοικίας, ή μία από τις ραδιοσυχνότητες της κινητής τηλεφωνίας), ή παραδοσιακότερα μέσα όπως τα ηχητικά κύματα σε μιά αίθουσα συνεδρίασης. Σε όλες αυτές τις περιπτώσεις υπάρχουν ειδικοί κανόνες ("πρωτόκολο επικοινωνίας") για το ποιός και πότε μπορεί να "πάρει το λόγο", δηλαδή με ποιόν τρόπο θα εξασφαλιστεί η τήρηση του κανόνα ότι το πολύ ένας οδηγητής (ομιλητής) μπορεί να είναι ενεργός κάθε φορά.

## 9.2 Πολύμπιτες Τρικατάστατες Λεωφόροι (Buses):

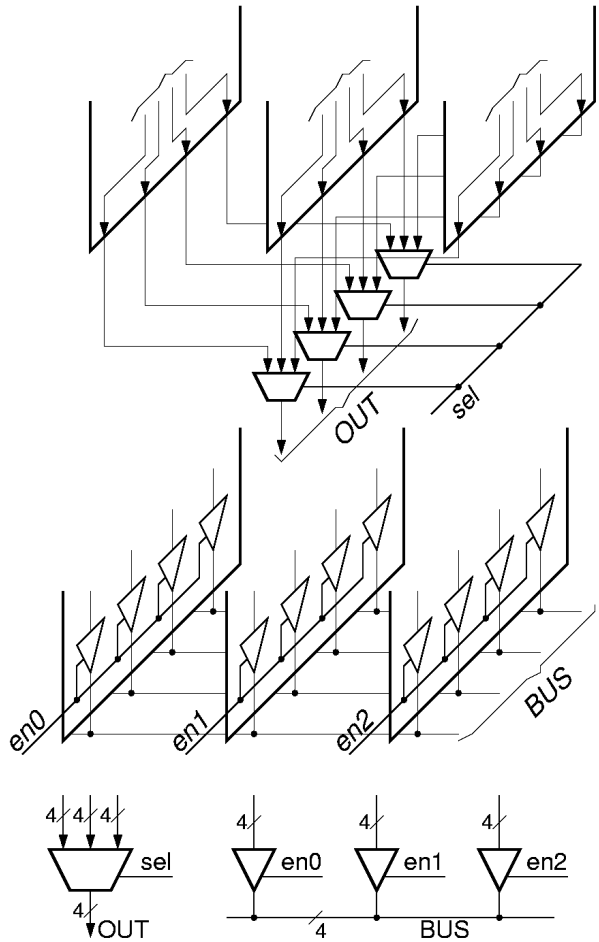
Λεωφόροι (buses) του ενός bit (ένα σύρμα) είναι κοινές όταν η απόσταση επικοινωνίας είναι σημαντική. Για μικρότερες αποστάσεις, π.χ. μέσα σε ένα κουτί ή πάνω σε μία πλακέτα ή μέσα σε ένα chip, συνήθως χρησιμοποιούμε λεωφόρους πολλαπλών bits, δηλαδή λεωφόρους που μεταφέρουν, κάθε φορά, μιά ολόκληρη λέξη πληροφορίας αποτελούμενη από πολλά bits· όλα τα bits της λέξης αυτής προέρχονται από την ίδια πηγή. Το σχήμα (φανταστείτε το σαν τριδιάστατο), δείχνει ένα παράδειγμα τριών πηγών (τρεις πλακέτες), καθεμία από τις οποίες έχει μιά τετράμπιτη λέξη πληροφορίας να στείλει. Στο επάνω μέρος του σχήματος, η λέξη εξόδου, OUT, προκύπτει από έναν τετράμπιτο πολυπλέκτη 3-σε-1· αυτός αποτελείται, όπως είδαμε στην §8.1, από τέσσερις βασικούς (μονόμπιτους) πολυπλέκτες 3-σε-1, όπου όλοι τους έχουν κοινά σήματα επιλογής, sel. Στη μέση του σχήματος φαίνεται η υλοποίηση του ίδιου συστήματος με τρικατάστατους οδηγητές· η λέξη εξόδου εμφανίζεται πάνω στην τετράμπιτη λεωφόρο BUS. Όλοι οι τρικατάστατοι οδηγητές της κάθε μιάς πηγής ελέγχονται (ενεργοποιούνται) από ένα κοινό σήμα ελέγχου, en0, en1, en2. Στο κάτω μέρος του σχήματος φαίνονται τα συνεπτυγμένα σύμβολα των δύο υλοποιήσεων: κάθε τετράδα συρμάτων που μεταφέρει

τα 4 bits της ίδιας λέξης παριστάνεται με ένα σύρμα και μία πλάγια γραμμή που υποδεικνύει το πλάτος της λέξης· το "πλάτος" του πολυπλέκτη ή των τρικατάστατων οδηγητών προκύπτει εμμέσως από το πλάτος των λέξεων που αυτοί χειρίζονται.

### 9.3 Εσωτερική Οργάνωση Μνημών SRAM και Μνήμες Πολλαπλών Chips:

Παρακαλείσθε να διαβάσετε για το θέμα αυτό τις παραγράφους **10.3.1, 10.3.2** (σελ. 1021-1025) του βιβλίου του Wakerly, και τις σελίδες 371-375 της **2ης έκδοσης** του βιβλίου του Mano· όλες αυτές περιλαμβάνονται στην εξεταστέα ύλη του μαθήματος. Παρακαλώ, λάβετε υπ' όψη τις εξής σημειώσεις:

- Σχήμα 10-20 (σελ. 1023) Wakerly: το σχήμα δείχνει μόνο την ισοδύναμη λογική συμπεριφορά του κυττάρου μνήμης. Το πραγματικό κύκλωμα είναι διαφορετικό, και λειτουργεί ισοδύναμα μεν αλλά με διαφορετικό τρόπο· σε τεχνολογία CMOS, το πραγματικό κύκλωμα του κυττάρου "στατικής" μνήμης αποτελείται από μόνο 6 transistors.
- Σελίδες από το βιβλίο Mano: η παράγραφος αυτή είναι γραμμένη για αναγνώστες οι οποίοι δεν ξέρουν για τρικατάστατους οδηγητές. Έτσι, για το σχήμα 7-29, γράφει ότι οι 4x8 έξοδοι δεδομένων των τεσσάρων chips πρέπει να περάσουν από 8 πύλες Η για να δώσουν τις 8 τελικές γραμμές εξόδου· οι πύλες αυτές δεν φαίνονται στο σχήμα. Στην πραγματικότητα, εμείς ξέρουμε ότι τα περισσότερα chips μνήμης προσφέρουν τρικατάστατες εξόδους, οπότε οι πύλες Η περιττεύουν· έτσι, το σχήμα 7-29 δείχνει την πραγματικότητα: η έξοδος είναι μία οκτάμπιτη τρικατάστατη λεωφόρος (bus).



### 9.4 SRAM με Αμφίδρομα Pins Δεδομένων (Τρικατάστατη Λεωφόρος):

Τα σύρματα που οδηγούνται από κλασικές (όχι τρικατάστατες) πύλες έχουν μονόδρομη φορά ροής της πληροφορίας (unidirectional wire), από την έξοδο της οδηγητριας πύλης προς τις εισόδους που συνδέονται στο σύρμα (§3.3). Αντίθετα, οι τρικατάστατες λεωφόροι επιτρέπουν **αμφίδρομη** (bidirectional) μεταφορά πληροφορίας: η κατεύθυνση μπορεί να αλλάξει κάθε φορά που αλλάζει ο ενεργός οδηγητής. Την ιδιότητα αυτή την εκμεταλλεύονται οι συσκευές που επιθυμούν να επικοινωνούν αμφίδρομα (άλλοτε αποστολή και άλλοτε λήψη πληροφοριών) με άλλες συσκευές μέσω πολυπλέκτη-λεωφόρου. Εάν η επικοινωνία γίνει με κλασικές πύλες, χρειάζονται χωριστά σύρματα εισόδου και χωριστά σύρματα εξόδου, όπως στη μνήμη του σχήματος 7-27 του βιβλίου (σελ. 370)· αν όμως η επικοινωνία γίνει μέσω τρικατάστατης λεωφόρου, τότε αρκεί η συσκευή να επικοινωνεί με τον έξω κόσμο μέσω του ενός, μοναδικού σύρματος ανά bit της λεωφόρου. Ως αμφίδρομα, τα σύρματα αυτά συχνά ονομάζονται σύρματα "**εισόδου/εξόδου**" (input/output - I/O).

Ας χρησιμοποιήσουμε σαν παράδειγμα το chip στατικής RAM (SRAM) του παρόντος εργαστηρίου: πρόκειται για το chip "PD43256BCZ" (της εταιρείας NEC) που είναι μία

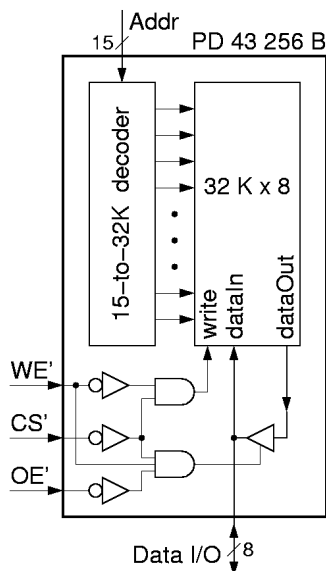
SRAM μεγέθους 32Kx8· ένα λογικό διάγραμμα υψηλού επιπέδου του chip φαίνεται στο σχήμα, και περισσότερες πληροφορίες γι' αυτό μπορείτε να βρείτε στο: <http://www.downloads.reactivemicro.com/Public/Electronics/Static%20RAM/NEC%20uPD43256B%2032k%20x%208bit%20SRAM%20Datasheet.pdf> . Το chip αυτό έχει μία μονόδρομη, δεκαπεντάμπτη είσοδο διεύθυνσης, τρεις μονόδρομες εισόδους ελέγχου, και μία αμφίδρομη, οκτάμπτη είσοδο/έξοδο δεδομένων. Τα τελευταία αυτά 8 σύρματα προορίζονται να συνδεθούν σε μία λεωφόρο δεδομένων. Όταν θέλουμε να γράψουμε στη μνήμη, πρέπει κάποια άλλη συσκευή να οδηγήσει τη λεωφόρο, βάζοντας εκεί τα δεδομένα που θέλουμε να εγγραφούν· η μνήμη διαβάζει τα δεδομένα από τα σύρματα αυτά, και τα αποθηκεύει στην επιθυμητή λέξη. Όταν θέλουμε να διαβάσουμε από τη μνήμη, ενεργοποιούμε τους τρικατάστατους οδηγητές που έχει μέσα του το chip, οπότε αυτό γίνεται ο οδηγητής της λεωφόρου και τοποθετεί εκεί τη λέξη που του ζητήσαμε να διαβάσει· από εκεί, τη λέξη αυτή θα την πάρουν οι άλλοι "ακροατές" που συνδέονται πάνω στη λεωφόρο.

Από άποψη χρονισμού, η λειτουργία του PD43256B έχει ως εξής. Η ανάγνωση από το chip μπορεί να θεωρηθεί σαν καθαρά συνδυαστική πράξη: μπορούμε να φανταστούμε ότι ένας μεγάλος, οκτάμπιτος πολυπλέκτης 32K-σε-1 φέρνει στον εσωτερικό κόμβο "dataOut" το περιεχόμενο της λέξης εκείνης την οποία επιλέγουν τα 15 bits της διεύθυνσης· μόλις αλλάξει η διεύθυνση, μετά από λίγο (καθυστερήση ανάγνωσης) αλλάζουν και τα dataOut και γίνονται ίσα με τα περιεχόμενα της νέας θέσης μνήμης (η πραγματική εσωτερική λειτουργία δεν είναι έτσι, αλλά αυτό δεν μας αφορά εδώ). Η εγγραφή, από την άλλη μεριά, πρέπει να γίνει προσεκτικότερα: πρώτα πρέπει να τοποθετηθεί η επιθυμητή διεύθυνση εγγραφής στην είσοδο διεύθυνσης, και **μετά** από αυτό και μόνο πρέπει να ανάψει το εσωτερικό σήμα "write". Όση ώρα είναι αναμένο το σήμα write πρέπει η διεύθυνση να παραμένει σταθερή· καθ' όλη αυτή την ώρα, η επιλεγόμενη λέξη μνήμης λειτουργεί σαν μανταλωτής: ό,τι δεδομένα υπάρχουν πάνω στις εισόδους "dataIn" εγγράφονται στη λέξη --προφανώς, τα τελευταία τέτοια δεδομένα πριν σβήσει το σήμα write θα παραμείνουν στην επιλεγμένη λέξη μνήμης. Η είσοδος διεύθυνσης επιτρέπεται να αλλάξει μόνον αφού σβήσει το σήμα write.

Η εξωτερική συμπεριφορά του chip προσδιορίζεται από τους παραπάνω κανόνες και από τα τρία σήματα ελέγχου, WE', CS', και OE', σύμφωνα με το λογικό διάγραμμα που φαίνεται στο σχήμα. Και τα τρία σήματα ελέγχου έχουν αρνητική πολικότητα: ενεργοποιούν τη λειτουργία τους, το καθένα, όταν είναι μηδέν (0). Το σήμα CS' (αρνητικό Chip Select) ενεργοποιεί ή αδρανοποιεί ολόκληρο το chip· προορίζεται για χρήση όταν φτιάχνουμε μία μεγάλη μνήμη από πολλά chips, για να επιλέγουμε σε ποίο chip απευθυνόμαστε κάθε φορά. Όταν CS'=0 (ενεργό chip), το σήμα WE' (αρνητικό Write Enable) ενεργοποιεί την εγγραφή, δηλαδή ανάβει το write, οπότε γίνεται μηδέν (0). Τέλος, όταν CS'=0 (ενεργό chip), το σήμα OE' (αρνητικό Output Enable) ανάβει τους τρικατάστατους οδηγητές της λεωφόρου εξόδου, όταν WE'=1 δηλαδή όταν δεν κάνουμε εγγραφή (άρα κάνουμε ανάγνωση), με άλλα λόγια όταν θέλουμε η λέξη που διαβάζουμε να βγει έξω από το chip, στη λεωφόρο δεδομένων.

### **Πείραμα 9.5: Τρικατάστατη Λεωφόρος με Μνήμες SRAM**

Κατασκευάστε και ελέγξτε το κύκλωμα που φαίνεται στο σχήμα. Είναι οργανωμένο γύρω από μία οκτάμπτη λεωφόρο, BUS. Σ' αυτήν συνδέονται τρεις πηγές πληροφορίας: δύο (οκτάμπιτα) SRAM chips (μέσον) σαν αυτά που μελετήσαμε παραπάνω, και μία πηγή οκτάμπιτων αριθμών (επάνω) που προορίζονται σαν δεδομένα εγγραφής στις μνήμες. Κάθε chip μνήμης έχει 32K (οκτάμπιτες) λέξεις, άρα 15 bits διεύθυνσης· όμως, λόγω περιορισμών στα σύρματα και στην πηγή διευθύνσεων, εμείς θα χρησιμοποιήσουμε μόνο τις πρώτες 256 λέξεις στο κάθε chip, μεταβάλλοντας μόνο τα 8 (LS) bits από τα 15 bits διεύθυνσης, και κρατώντας τα υπόλοιπα 7 bits σταθερά στο 0. Τροφοδοτούμε την ίδια διεύθυνση και στα δύο chips. Σαν πηγή τόσο διευθύνσεων όσο και δεδομένων θα

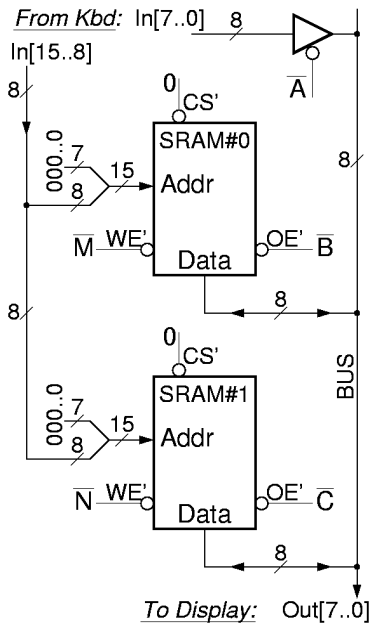


χρησιμοποιήσετε το πληκτρολόγιο (§8.3): τα μεν 8 MS bits του (In[15..8]) θα χρησιμοποιήσετε σαν πηγή διεύθυνσης, τα δε 8 LS bits του (In[7..0]) θα χρησιμοποιήσετε σαν πηγή δεδομένων. Όταν πληκτρολογείτε ένα νέο τετραψήφιο δεκαεξαδικό αριθμό, τα δύο αριστερά ψηφία του θα προσδιορίζουν τη νέα διεύθυνση, και τα δύο δεξιά ψηφία του θα προσδιορίζουν τα νέα δεδομένα (για να αλλάξετε μόνο το ένα από τα δύο -- δεδομένα ή διεύθυνση-- πρέπει δυστυχώς να ξαναπληκτρολογήσετε και το άλλο, αμετάβλητο).

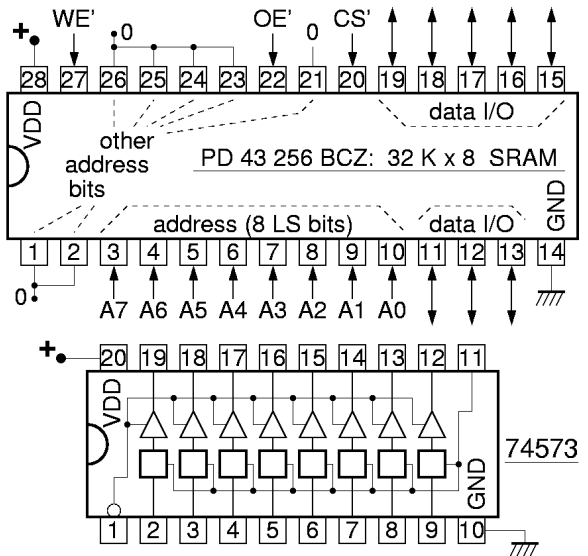
Στη λεωφόρο συνδέεται επίσης και ένας ακροατής, η οθόνη υγρών κρυστάλλων (LCD) (για την ακρίβεια μόνο τα 8 LS bits της), για να παρακολουθούμε τη λογική τιμή της λεωφόρου. Λάβετε υπ' όψη σας ότι το κύκλωμα εισόδου της οθόνης έχει ασθενείς αντιστάσεις καθέλκυσης: όταν ένα bit που του συνδέουμε είναι "αιωρούμενο" (floating - HiZ - ασύνδετο - ανοικτό κύκλωμα), τότε αυτή η αντίσταση καθέλκυσης το φέρνει σε χαμηλή τάση, κι έτσι η οθόνη δείχνει 0· όταν κάποιος (τρικατάστατος) οδηγητής οδηγήσει αυτό το bit με 0 ή 1, τότε ο οδηγητής αυτός υπερισχύει της (ασθενούς) αντίστασης καθέλκυσης, και φέρνει το bit στην τάση που αυτός επιθυμεί. Άρα, όποτε η λεωφόρος μένει ανοδηγήτη (δηλαδή όλοι οι τρικατάστατοι οδηγητές της είναι σβηστοί), στην οθόνη θα βλέπετε "00" (δεκαεξαδικό) στα δύο δεξιά ψηφία της. (Τα δύο αριστερά ψηφία της οθόνης αντιστοιχούν στα bits Out[15..8], που είναι ασύνδετα, άρα, για τον ίδιο παραπάνω λόγο, αυτά θα δείχνουν πάντα "00").

Η λειτουργία του κυκλώματος έχει ως εξής. Επιλέγουμε μία διεύθυνση και μία λέξη δεδομένων μέσω του πληκτρολογίου. Στη συνέχεια, πατάμε το διακόπτη A για να ενεργοποιήσουμε τον επάνω τρικατάστατο οδηγητή· η λέξη δεδομένων πρέπει να εμφανιστεί πάνω στη λεωφόρο και την δούμε στην οθόνη. Με πατημένο το διακόπτη A, ανάβουμε και στη συνέχεια σβήνουμε έναν από τους διακόπτες M ή N, ώστε να ενεργοποιήσουμε την εγγραφή σε μία από τις δύο μνήμες· εάν το κύκλωμα λειτουργεί σωστά, τα δεδομένα από τη λεωφόρο θα πρέπει τώρα να έχουν γραφτεί στην επιλεγμένη διεύθυνση της μνήμης αυτής (γράψτε σ' ένα χαρτί τη διεύθυνση και τα δεδομένα για επαλήθευση αργότερα). Αλλάζουμε τα δεδομένα και ενεργοποιούμε το WE' της άλλης μνήμης· τα νέα δεδομένα θα πρέπει να γράφτηκαν στην επιλεγμένη διεύθυνση εκείνης της μνήμης. Αλλάζουμε και τη διεύθυνση και τα δεδομένα, και επαναλαμβάνουμε τη διαδικασία ούτως ώστε να γράψουμε προκαθορισμένες (και διαφορετικές!) λέξεις σε κάμποσες από τις 256 προσπελάσιμες λέξεις καθεμιάς από τις δύο μνήμες. Προσοχή: τη στιγμή της αλλαγής της διεύθυνσης να μην είναι ενεργός κανένας από τους διακόπτες M ή N. Στη συνέχεια, αφού "γεμίσαμε" (αρχικοποιήσαμε) τις μνήμες, διαβάζουμε τα περιεχόμενά τους για να διαπιστώσουμε αν είναι τα σωστά: επιλέγουμε (μέσω του πληκτρολογίου) τις διεθύνσεις στις οποίες γράψαμε (τα δεδομένα από το πληκτρολόγιο τώρα δεν θα πρέπει να έχουν επίδραση), και για κάθε διεύθυνση πατάμε πρώτα το διακόπτη B για να διαβάσουμε από την πρώτη μνήμη, και μετά το διακόπτη C για να διαβάσουμε από τη δεύτερη μνήμη· κάθε φορά, πρέπει να εμφανίζονται στην οθόνη τα δεδομένα που είχαμε γράψει παλαιότερα σε εκείνη τη διεύθυνση εκείνης της μνήμης (που είναι σκόπιμο να είναι διαφορετικά μεταξύ τους, διαφορετικά από τα δεδομένα άλλων διεθύνσεων, και διαφορετικά από τα τωρινά δεδομένα του πληκτρολογίου). Φυσικά, ποτέ δεν πρέπει να είναι πατημένοι ταυτόχρονα περισσότεροι από ένας από τους διακόπτες A, B, C, οι οποίοι ενεργοποιούν τους τρεις διαφορετικούς οδηγητές της λεωφόρου.

Τα chips μνήμης είναι ακριβά, και γι' αυτό δεν σας έχουν δοθεί χωριστά σε καθέναν, στο "σακουλάκι" της §3.6· για τον ίδιο ακριβώς λόγο, να είστε **πολύ προσεκτικοί** να μην τα κάψετε/χαλάσετε! Θα τα βρείτε έτοιμα (αλλά ασύνδετα) πάνω στην πλακέτα συνδέσεων (breadboard), και όταν τελειώσετε αφήστε τα επίσης πάνω στην πλακέτα, χωρίς να τα βγάλετε, για να μην φθείρονται (στραβώνουν) τα ποδαράκια τους (βγάλτε μόνο τα



σύρματά σας, και το δικό σας chip, 74573). Τα pins της SRAM φαίνονται στο σχήμα. Τα pins διεύθυνσης που θα χρησιμοποιήσουμε εμείς είναι τα 3 έως 10· τα υπόλοιπα 7 pins διεύθυνσης, που δεν τα χρησιμοποιούμε, πρέπει να συνδεθούν σε κάποια σταθερή τάση (π.χ. να γειωθούν (λογικό 0)), διότι αλλιώς, αν μείνουν ασύνδετα (αιωρούμενα - floating) μπορεί να παίρνουν τυχαίες (και μεταβαλλόμενες!) τιμές. Τα pins δεδομένων (εισόδου/εξόδου) είναι τα 11, 12, 13, και 15 έως και 19. Τέλος, οι 3 εισόδοι ελέγχου βρίσκονται στα pins 20, 22, και 27. Σαν **τρικατάστατους οδηγητές** για τα δεδομένα εγγραφής, χρησιμοποιήστε τα 8 bits ενός chip μανταλωτών "74573" όπως αυτά που χρησιμοποιήσατε στα 2 τελευταία εργαστήρια. Όπως είχαμε πεί, τα chips αυτά έχουν τρικατάστατες εξόδους. Για να χρησιμοποιήσουμε μόνο τους οδηγητές, χωρίς τους μανταλωτές, αρκεί να συνδέσουμε μόνιμως στο 1 το σήμα φόρτωσης (pin 11): τότε, ό,τι υπάρχει στις εισόδους "εγγράφεται" συνεχώς και αμέσως και εμφανίζεται στις εξόδους. Η είσοδος ενεργοποίησης των τρικατάστατων οδηγητών είναι το pin 1, και είναι **αρνητικής** λογικής: ενεργοποιεί τους οδηγητές με 0, και τους αδρανοποιεί με 1.



**Πριν** φτάσετε στο εργαστήριο κάντε ένα πλήρες σχεδιάγραμμα συνδεσμολογίας, δείχνοντας ακριβώς ποιο pin τίνος chip πρέπει να συνδεθεί πού, όπως έχουμε ήδη πεί πολλές φορές. Επίσης, γράψτε σ' ένα χαρτί το "πρόγραμμα ελέγχων" σας (test plan): σε ποιές διευθύνσεις θα γράψετε, και τι (διαφορετικά) δεδομένα σε καθεμία, στο κάθε chip. **Στο εργαστήριο**, κατασκευάστε το κύκλωμα και ελέγξτε τη λειτουργία του. Όταν σβήνετε την τάση τροφοδοσίας, τι γίνονται τα περιεχόμενα της μνήμης;

### Άσκηση 9.6: Μνήμη Διπλασίου Ύψους (Λέξεων)

Το κύκλωμα του προηγουμένου εργαστηρίου μπορεί, με ελάχιστες αλλαγές στα σήματα ελέγχου, να χρησιμοποιηθεί σαν μία μεγάλη μνήμη με διπλάσιο αριθμό λέξεων απ' όσες λέξεις έχει το κάθε chip (ή χρησιμοποιούμε εμείς στο κάθε chip), δηλαδή μία μνήμη διπλασίου "ύψους" (θεωρώντας ότι το "ύψος" αντιστοιχεί στις λέξεις της μνήμης και το "πλάτος" στα bits της κάθε λέξης). Ας παραμείνουμε προς στιγμήν στις 256 μόνο λέξεις ανά chip του προηγουμένου πειράματος (μόνο 8 bits διεύθυνσης στα chips): τότε η "συνολική" μνήμη θα έχει 512 λέξεις, άρα 9 bits διεύθυνσης. Θεωρήστε ότι το έννατο bit διεύθυνσης είναι ο διακόπτης M· χρησιμοποιήστε τον για να ελέγξετε κατάλληλα τα σήματα CS' των chips, διότι έτσι πετυχαίνετε "μ' ένα σμπάρο δυό τρυγόνια": αδρανοποιήτε ταυτόχρονα και τις εγγραφές και τις αναγνώσεις στο εκάστοτε μη επιλεγμένο chip. Όταν το κάνετε σωστά αυτό, μπορείτε στη συνέχεια να οδηγήστε από μίαν ίδια, κοινή είσοδο τα δύο σήματα WE' των δύο chips, και από μίαν άλλη, επίσης κοινή είσοδο τα δύο σήματα OE' και των δύο chips. Κάντε το σχεδιάγραμμα του κυκλώματος στην αναφορά του εργαστηρίου σας, και εξηγήστε εν συντομία το γιατί. Αν έχετε χρόνο, δοκιμάστε το και στο εργαστήριο (π.χ. οδηγήστε τα δύο WE' από τον διακόπτη B', και τα δύο OE' από τον διακόπτη C').

Εάν στη λεωφόρο δεν υπάρχει κανένας άλλος οδηγητής, τότε μπορούμε να αποφασίσουμε όποτε δεν κάνουμε εγγραφή να κάνουμε πάντοτε ανάγνωση. Πώς θα χρησιμοποιούσατε τότε τον διακόπτη A και μόνο (με τις δύο συμπληρωματικές του εξόδους, A και A') για να ελέγξετε κατάλληλα και τον επάνω τρικατάστατο οδηγητή και τα WE' των δύο chips, και τα OE' των δύο chips, όλα μόνο από τον A; (Ας πούμε ότι "A ελεύθερος" σημαίνει "ανάγνωση", και "A πατημένος" σημαίνει "εγγραφή"). Κάντε το σχεδιάγραμμα του κυκλώματος στην αναφορά του εργαστηρίου σας, και εξηγήστε εν συντομία πώς και γιατί δουλεύει σωστά.

Τέλος, ας αφήσουμε κατά μέρος τον περιορισμό για 8 μόνο bits διεύθυνσης, δηλαδή 256 μόνο λέξεις ανά chip. Θεωρήστε το πλήρες "ύψος" (βάθος) του κάθε chip, των 32K λέξεων. Τότε, η "συνολική" μας μνήμη είναι 64K λέξεις, και χρειάζεται 16 bits διεύθυνσης. Δείτε σ' ένα σχεδιάγραμμα μία δεκαεξάμπιτη διεύθυνση να μπαίνει, και πώς αυτή θα τροφοδοτήσει το κύκλωμά μας για να συμπεριφέρεται αυτό σαν μία μεγάλη μνήμη 64K x 8.

### Άσκηση 9.7: Η Θέση των bits Δεδομένων και Διευθύνσεων

Στο παραπάνω πείραμα 9.5 δεν δώσαμε σημασία στο ποιο ποδαράκι του chip ήταν το LS bit των δεδομένων, ποιο το επόμενο, κ.ο.κ. Σύμφωνα με το εγχειρίδιο, το pin 11 του chip φαίνεται να είναι το LS bit των δεδομένων, και το pin 12 το επόμενο.

- i. Σχεδιάστε τα 8 flip-flops (8 κουτάκια) μίας λέξης της μνήμης (π.χ. της 13), και θεωρήστε ότι γράφουμε εκεί ως δεδομένα τον αριθμό F9, έχοντας συνδέσει σωστά το LS bit *BUS[0]* στο ποδαράκι 11 του chip, το επόμενο bit *BUS[1]* στο ποδαράκι 12, κ.ο.κ. Γράψτε μέσα σε κάθε flip-flop (κουτάκι) το σωστό bit, 1 ή 0.
- ii. Θεωρήστε ότι από το παραπάνω κύκλωμα (i) διαβάζουμε την παραπάνω λέξη (ας πούμε την 13). Ποιός αριθμός θα εμφανιστεί στο BUS;
- iii. Έστω τώρα ότι στο κύκλωμα (i) συνδέσαμε ανάποδα το LS bit *BUS[0]* στο ποδαράκι 12 και το bit *BUS[1]* στο ποδαράκι 11, ενώ τα υπόλοιπα bits παραμένουν όπως πριν. Έστω ότι πάλι γράφουμε στη λέξη 13, έχοντας τον αριθμό F9 πάνω στο BUS. Τώρα, τι bits θα μπουν μέσα σε κάθε κουτάκι της λέξης στη μνήμη; -- σχεδιάστε όπως στο (i). Ποιός αριθμός θα νόμιζε κανείς ότι είναι αυτός, εάν μπορούσε να "δεί" μέσα στο chip;
- iv. Έστω ότι από το νέο ("ανάποδο") κύκλωμα (iii) διαβάζουμε την παραπάνω λέξη, 13. Ποιός αριθμός θα εμφανιστεί στο BUS; Είναι αυτός ίδιος με τον αριθμό που θα νόμιζε κανείς αν μπορούσε να "δεί" μέσα στο chip, ή ίδιος με τον αριθμό που "γράψαμε" από το BUS; Γιατί; Τι συμπέρασμα βγάξετε;

Μία αντίστοιχη με την παραπάνω περίπτωση συμβαίνει και με τα bits διεύθυνσης! Θεωρήστε ότι γράφουμε στη λέξη 0 τον αριθμό 10, στην 1 τον 11, στην 2 τον 12, και στην 3 τον 13. Σχεδιάστε αυτές τις 4 πρώτες λέξεις της μνήμης, την μία κάτω από την άλλη, και γράψτε μέσα σε καθεμία τα περιεχόμενά της. Στη συνέχεια θεωρήστε ότι, κατά λάθος, κάποιος συνέδεσε το LS bit *Addr[0]* της διεύθυνσης στο ποδαράκι 9 του chip που είναι το bit "A1", και το επόμενο bit *Addr[1]* της διεύθυνσης στο ποδαράκι 10 του chip που είναι το bit "A0" (ενώ όλα τα υπόλοιπα bits διεύθυνσης είναι σωστά συνδεδεμένα). Ξαναγράψουμε στη διεύθυνση 0 τον αριθμό 10, στην 1 τον 11, στην 2 τον 12, και στην 3 τον 13· οι διευθύνσεις αυτές αντιστοιχούν στο τι "σωστά" δίνει το *Addr*, ενώ το chip τα βλέπει αλλιώς στα pins A1, A0. Σχεδιάστε τώρα τις 4 πρώτες λέξεις της μνήμης, και γράψτε μέσα σε καθεμία τα περιεχόμενά της· είναι "ανακατεμένα"; Στο ίδιο αυτό κύκλωμα, με τα ανάποδα ποδαράκια 9 και 10, διαβάζουμε από τις διευθύνσεις 0, 1, 2, και 3· και πάλι, οι διευθύνσεις αυτές αντιστοιχούν στο τι δίνει το *Addr*, ενώ το chip τα βλέπει αλλιώς στα pins A1, A0. Τι δεδομένα διαβάσαμε από τη διεύθυνση 0, τι από την 1, τι από την 2, τι από την 3; Μοιάζουν λάθος ή σωστά; Γιατί; Τι συμπέρασμα βγάξετε;

[Up to the Home Page of CS-120](#)

© copyright University of Crete, Greece.  
last updated: 22 Nov. 2010, by [M. Katevenis](#).