



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

Ψηφιακή Σχεδίαση

Εργαστήριο 2:
Διευθύνσεις, Αποκωδικοποιητές, Κώδικες,
Ηλεκτρονόμοι

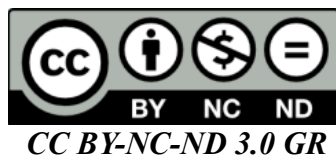
Μανόλης Γ.Η. Κατεβαίνης

Τμήμα Επιστήμης Υπολογιστών

Άδειες Χρήσης

• Το παρόν εκπαιδευτικό υλικό υπόκειται στην άδεια χρήσης **Creative Commons** και ειδικότερα

*Αναφορά – Μη εμπορική Χρήση – Όχι Παράγωγο Έργο 3.0 Ελλάδα
(Attribution – Non Commercial – Non-derivatives 3.0 Greece)*



• Εξαιρείται από την ως άνω άδεια υλικό που περιλαμβάνεται στις διαφάνειες του μαθήματος, και υπόκειται σε άλλου τύπου άδεια χρήσης. Η άδεια χρήσης στην οποία υπόκειται το υλικό αυτό αναφέρεται ρητώς.

Χρηματοδότηση

• Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.

• Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Κρήτης**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.

• Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Ψηφιακή Σχεδίαση

Εργαστήριο 2:

Διευθύνσεις, Αποκωδικοποιητές, Κώδικες, Ηλεκτρονόμοι

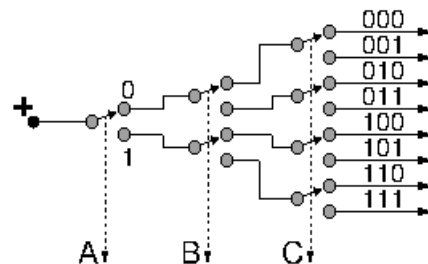
[Βιβλία: προαιρετικά μπορείτε να διαβάσετε: Wakerly: § 1.1 - 1.4 (σελ. 1-10)· ή Mano (4η έκδοση): § 1.1 (σελ. 1-3) και § 1.9 (σελ. 28-31)]

2.1 Δένδρα Αποφάσεων

Η μνήμη που είδαμε στο προηγούμενο εργαστήριο (§1.7) είχε πολλές εισόδους επιλογής (οκτώ), που πρέπει ο χρήστης να είναι πολύ προσεκτικός *μία και μόνο μία* από αυτές να είναι επιλεγμένη (ενεργή, αναμμένη). Στους σημερινούς υπολογιστές, που οι μνήμες έχουν εκατομμύρια (ή μάλλον δισεκατομμύρια!) λέξεις, δεν θα ήταν καθόλου πρακτικό ο χρήστης της μνήμης να πρέπει να ελέγχει (να αναβοσβήνει) εκατομμύρια (ή και δισεκατομμύρια) συρματάκια προκειμένου να μπορέσει να προσδιορίσει ποιάν από τις τόσες πολλές λέξεις θέλει να διαβάσει. Χρειαζόμαστε πραγματικά τόσο πολλές "μονάδες πληροφορίας" (bits) για να επιλέξουμε την επιθυμητή λέξη μέσα σε μία μνήμη; Η ισοδύναμη: χρειαζόμαστε τόσο πολλά bits για να πούμε σ' έναν πολυπλέκτη ποιάν από τις εισόδους τους να επιλέξει αυτός; Η απάντηση είναι πως όχι, χρησιμοποιώντας την έννοια της κωδικοποίησης και τα κυκλώματα αποκωδικοποιητών που θα δούμε σε αυτό το εργαστήριο.

Ας θυμηθούμε από το πείραμα 1.3 ότι όταν ένα κύκλωμα έχει 3 διακόπτες (3 bits) σαν εισόδους του, τότε ο πίνακας αληθείας του έχει 8 γραμμές: οι 3 διακόπτες μπορούν να βρεθούν σε 8 διαφορετικούς συνδυασμούς καταστάσεων. Εάν αντιστοιχίσουμε σε καθέναν από αυτούς τους 8 συνδυασμούς μία από τις 8 λέξεις της μνήμης μας, τότε θα έχουμε καταφέρει με 3 μόνο διακόπτες (3 bits) να ελέγχουμε τη μνήμη μας και να επιλέγουμε οιαδήποτε από τις 8 λέξεις της θέλουμε! Αυτήν ακριβώς τη δουλειά κάνουν τα κυκλώματα αποκωδικοποίησης που θα μελετήσουμε σε αυτό το εργαστήριο, και ο καλύτερος τρόπος να τα καταλάβουμε είναι τα *δένδρα αποφάσεων* που θα δούμε αμέσως τώρα.

Το κύκλωμα δεξιά χρησιμοποιεί τρεις διακόπτες, A, B, και C. Ο διακόπτης A είναι απλός, όπως και αυτοί που έχουμε χρησιμοποιήσει μέχρι τώρα σε προηγούμενα εργαστήρια. Ας συμβολίζουμε με 0 (μηδέν) όταν αυτός είναι επάνω (ελεύθερος) και με 1 (ένα) όταν αυτός είναι κάτω. Ο διακόπτης B είναι *διπλός*, αποτελούμενος από δύο *συνδεδεμένους* απλούς διακόπτες: ο διακόπτης στο κάτω μισό του βρίσκεται *πάντα* στην ίδια κατάσταση όπως και ο διακόπτης στο επάνω μισό - ή και οι δύο επάνω (0), ή και οι δύο κάτω (1). Ο διακόπτης C είναι *τετραπλός*: όταν είναι ελεύθερος (0) είναι και τα 4 (συνδεδεμένα) κομμάτια του επάνω, ενώ όταν είναι πατημένος (1) είναι και τα 4 (συνδεδεμένα) κομμάτια του κάτω.



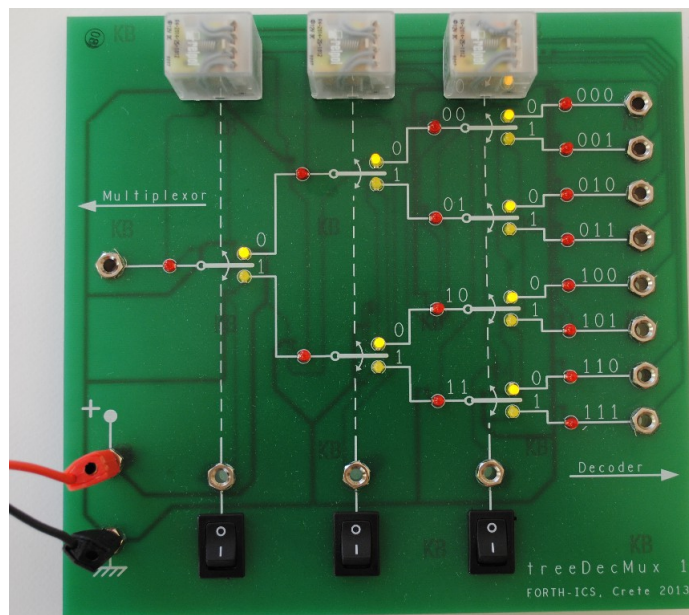
Το σχήμα αυτού του κυκλώματος θυμίζει ένα πλαγιαστό δένδρο, με την ρίζα και τον κορμό του αριστερά, και τα φύλλα δεξιά. Το λέμε *δένδρο αποφάσεων* διότι σε κάθε βαθμίδα του (σε κάθε διακλάδωσή του) ένας από τους τρεις διακόπτες αποφασίζει

προς τα πού θα στείλει το ηλεκτρικό ρεύμα που έρχεται από αριστερά, από τον θετικό πόλο της πηγής μας: εάν ο διακόπτης είναι 0 (ελεύθερος), στέλνει το ρεύμα επάνω· εάν είναι 1 (πατημένος) το στέλνει κάτω. Το δένδρο αποφάσεων λοιπόν είναι σαν την ανάλυση όλων των περιπτώσεων: ανάλογα με το συνδυασμό αποφάσεων των τριών διακοπών, ο ρεύμα που έρχεται από την πηγή θα καταλήξει σε μία (και μόνο μία) από τις οκτώ εξόδους. Στο σχήμα, η κάθε έξοδος σημαδεύεται από 3 bits, τα οποία δείχνουν τη θέση που πρέπει να έχουν οι τρεις διακόπτες προκειμένου το ρεύμα από την πηγή να φτάσει σε αυτή την έξοδο. Π.χ., προκειμένου το ρεύμα να φτάσει στη δεύτερη από επάνω έξοδο, πρέπει οι διακόπτες A-B-C να βρίσκονται στις θέσεις 0-0-1, δηλαδή επάνω-επάνω-κάτω.

Το κύκλωμα αυτό λέγεται *αποκωδικοποιητής 3-σε-8* (3-to-8 decoder), επειδή παίρνει έναν *κώδικα* τριών bits (τις εισόδους A-B-C) και τον αναλύει σε μία έξοδο για καθέναν από τους 8 υπάρχοντες συνδυασμούς. Έχει τη βασική ιδιότητα ότι *μία και μόνο μία* έξοδος του είναι αναμμένη κάθε φορά. Παρατηρήστε τώρα τον "δρόμο" που οδηγεί σε κάθε έξοδο του αποκωδικοποιητή. Κάθε τέτοιος δρόμος περνάει από ακριβώς τρεις διακόπτες --έναν A, έναν B, κι έναν C-- και οι τρεις τους εν σειρά. Άρα, κάθε έξοδος του αποκωδικοποιητή υλοποιεί μία λογική ΚΑΙ. Ο δρόμος που οδηγεί στην επάνω έξοδο περνά από όλους τους διακόπτες στην επάνω κατάσταση τους ("ΟΧΙ πατημένοι"), άρα η έξοδος αυτή ανάβει όταν και μόνον όταν: (ΟΧΙ A) ΚΑΙ (ΟΧΙ B) ΚΑΙ (ΟΧΙ C). Αυτό ακριβώς δείχνει και η ονομασία "000": η έξοδος αυτή "αποκωδικοποιεί" (ανιχνεύει) τον "κώδικα" (συνδυασμό) 0-0-0, δηλαδή την περίπτωση όπου και A=0 (A ψευδές, ή όχι A) και B=0 (όχι B) και C=0 (όχι C). Η δεύτερη έξοδος, η "001", βρίσκεται στην άκρη του δρόμου A ελεύθερος και B ελεύθερος και C πατημένος, δηλαδή (ΟΧΙ A) ΚΑΙ (ΟΧΙ B) ΚΑΙ (C), ή αλλιώς A=0 (όχι A) και B=0 (όχι B) και C=1 (ναι C). Και ούτω καθ' εξής, μέχρι την κάτω-κάτω έξοδο, την "111", που βρίσκεται στην άκρη του δρόμου A πατημένος και B πατημένος και C πατημένος, δηλαδή (A) ΚΑΙ (B) ΚΑΙ (C), ή αλλιώς A=1 (ναι A) και B=1 (ναι B) και C=1 (ναι C). Τελικά λοιπόν, οι 8 εξοδοί του αποκωδικοποιητή 3-σε-8 είναι όλες οι (οκτώ) συναρτήσεις ΚΑΙ των τριών εισόδων του, A, B, C, όπου η κάθε είσοδος E εμφανίζεται είτε σαν "ΟΧΙ E" είτε σαν "(ναι) E" --στους 8 συνδυασμούς που υπάρχουν κάνοντας όλες αυτές τις εναλλαγές "πολικότητας" (αρνητική πολικότητα το "όχι", θετική πολικότητα το "ναι").

Πείραμα 2.2: Η Πλακέτα Αποκωδικοποιητή 3-σε-8

Στο εργαστήριο θα βρείτε την εκπαιδευτική πλακέτα που φαίνεται στη φωτογραφία δεξιά, και η οποία υλοποιεί ακριβώς το παραπάνω δένδρο αποφάσεων με τα τρία επίπεδα (τρεις διακόπτες) και τις 8 εξόδους. Οι διακόπτες είναι όλοι τετραπλοί --απλώς, το πρώτο επίπεδο

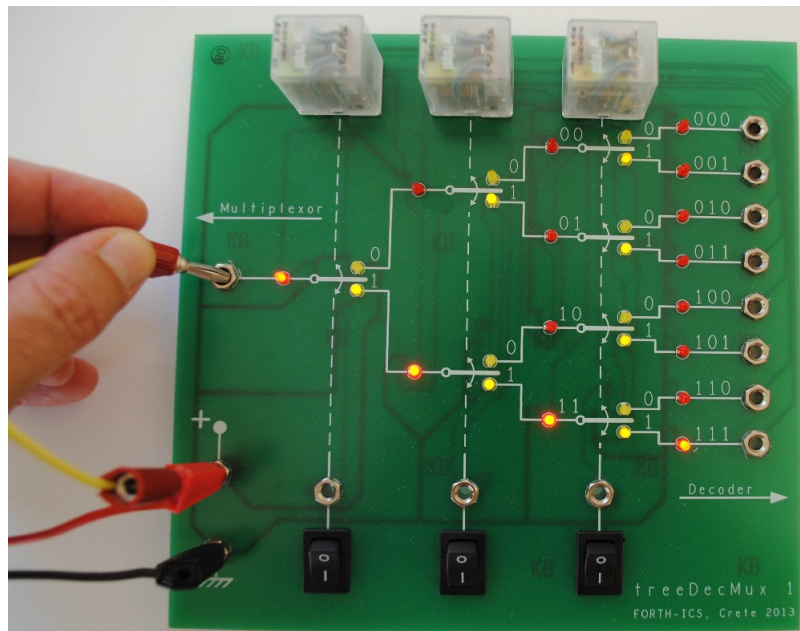


χρησιμοποιεί μόνο τον ένα διακόπτη από την τετράδα, και το δεύτερο επίπεδο χρησιμοποιεί μόνο τους 2 από τους 4 διακόπτες του. Οι τρεις αυτοί τετραπλοί διακόπτες βρίσκονται στην επάνω πλευρά της πλακέτας, και είναι σε μορφή

ηλεκτρονόμων, σαν αυτούς που θα δούμε παρακάτω στην §2.12, μόνο που εδώ είναι τετραπλοί (και δουλεύουν με 12 Volt) ενώ εκεί θα είναι διπλοί (και θα δουλεύουν στα 6 ή και 5 Volt). Οι ηλεκτρονόμοι (relays) είναι διακόπτες που κινούνται από έναν ηλεκτρομαγνήτη, και όχι με το χέρι. Εάν κοιτάξετε προσεκτικά μέσα από το διαφανές τους περίβλημα, θα διακρίνετε τους 4 χωριστούς διακόπτες, και θα δείτε ότι κινούνται ακριβώς οι 4 μαζί, όλοι δεξιά ή όλοι αριστερά. Οι τρεις ηλεκτρονόμοι που υπάρχουν πάνω στην πλακέτα αντιστοιχούν στους 3 διακόπτες A, B, C της προηγούμενης παραγράφου, και ελέγχονται (κινούνται), αντίστοιχα, από τους 3 μαύρους (χειροκίνητους) διακόπτες στην κάτω πλευρά της πλακέτας.

Τα κίτρινα LED δείχνουν σε ποιά θέση βρίσκεται ο κάθε διακόπτης. Αναβοσβήστε τους 3 μαύρους διακόπτες, και παρατηρήστε πώς αυτοί επηρεάζουν καθέναν τον ηλεκτρονόμο του, καθώς και την ή τις LED που βρίσκονται στη δική του βαθμίδα του δένδρου. Φτιάξτε, έναν-έναν τους 8 συνδυασμούς στους οποίους μπορούν να βρεθούν οι 3 διακόπτες.

Στη συνέχεια, όπως δείχνει η επόμενη φωτογραφία δεξιά, συνδέστε ένα σύρμα στη θετική τροφοδοσία (κόκκινο), και με την άλλη άκρη του ακουμπήστε την ρίζα του δένδρου, αριστερά στη μέση. Αμέσως ανάβουν μερικές από τις κόκκινες LED, δείχνοντάς σας σε ποιά σημεία φτάνει το ρεύμα από την θετική τροφοδοσία.



Παρατηρήστε ότι οι κίτρινες LED σας δείχνουν τη θέση του κάθε διακόπτη, είτε φτάνει σε αυτόν ρεύμα από την τροφοδοσία είτε όχι· αντίθετα, οι κόκκινες LED σας δείχνουν *μόνον* τα μέρη εκείνα του κυκλώματος όπου φτάνει ρεύμα από την τροφοδοσία. Στη φωτογραφία, όλοι οι διακόπτες είναι "πατημένοι" (1-1-1), οπότε το ρεύμα πηγαίνει στην κάτω έξοδο (111), όπως δείχνουν και οι κάτω κόκκινες LED που ανάβουν. Φτιάξτε, και πάλι τους 8 συνδυασμούς στους οποίους μπορούν να βρεθούν οι 3 διακόπτες, και μελετήστε προσεκτικά, γιά καθέναν τους, το ποιές κίτρινες και ποιές κόκκινες LED ανάβουν. Δείτε το "όνομα" της εξόδου όπου φτάνει το "κόκκινο", και την θέση των μαύρων διακοπτών, A, B, C, που οδηγούν σε αυτό.

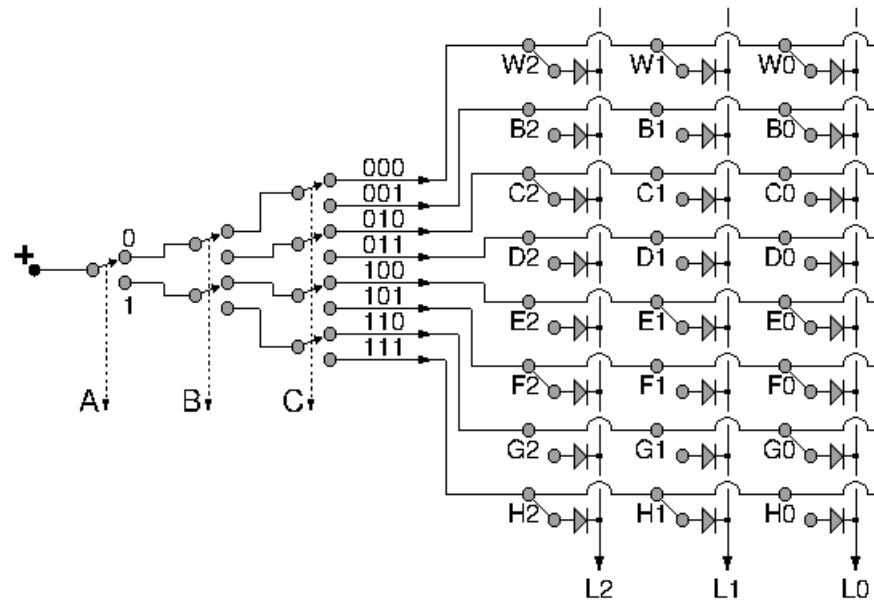
2.3: Διευθύνσεις Μνήμης και Αποκωδικοποιητής Διευθύνσεων

Στην §1.7, είδαμε τη βασική οργάνωση της Μνήμης σε λέξεις (words), σχεδιαζόμενες οριζόντια σαν γραμμές, αποτελούμενες η κάθεμία από bits, σχεδιαζόμενα κατακόρυφα, σε στήλες. Εκεί, και στο αντίστοιχο πείραμα 1.8, η επιλογή της λέξης που θέλαμε εκάστοτε να διαβάσουμε (δηλαδή οι τιμές των bits της να εμφανιστούν στις εξόδους δεδομένων, κάτω) γίνονταν τροφοδοτώντας ηλεκτρικό ρεύμα στο οριζόντιο σύρμα (γραμμή) που αντιστοιχούσε στην επιθυμητή λέξη. Όμως, όπως είπαμε παραπάνω, σε μία μεγάλη μνήμη, με N ως πούμε λέξεις, η επιλογή της λέξης που θέλουμε να διαβάσουμε μέσω του "ανάματος" του αντίστοιχου ενός από τα N

σύρματα γραμμής δεν βολεύει, διότι (α) το N είναι μεγάλος αριθμός, άρα πρέπει να έχουμε και να ελέγχουμε πολλά σύρματα εισόδου (ελέγχου), και (β) διότι πρέπει όποιος τροφοδοτεί (ελέγχει) αυτά τα σύρματα να είναι προσεκτικός ώστε πάντοτε ένα και *μόνον* ένα από αυτά να είναι "αναμμένο".

Τη λύση έρχονται να δώσουν οι *αποκωδικοποιητές*, που είδαμε παραπάνω. Έτσι, με $n = \log_2(N)$ bits επιλογής ($\log_2()$ εννοούμε τον δυαδικό λογάριθμο - το λογάριθμο με βάση 2) μπορούμε να επιλέγουμε μία μεταξύ $N = 2^n$ γραμμών δεδομένων εισόδου. Παραδείγματος χάριν, η πλακέτα του προηγούμενου πειράματος ήταν ένας αποκωδικοποιητής 3-σε-8: $n=3$ bits επιλογής αποκωδικοποιούνται στους $N=8$ συνδυασμούς τους, κι έτσι επιλέγουν την επιθυμητή είσοδο δεδομένων μεταξύ των $N=8$ παραχουσών τέτοιων.

Αυτός ακριβώς είναι ο τρόπος με τον οποίο οργανώνονται οι μνήμες των υπολογιστών και γενικότερα των ψηφιακών συστημάτων, όπως φαίνεται και στο σχήμα δεξιά. Τα τρία bits επιλογής, A, B, C,

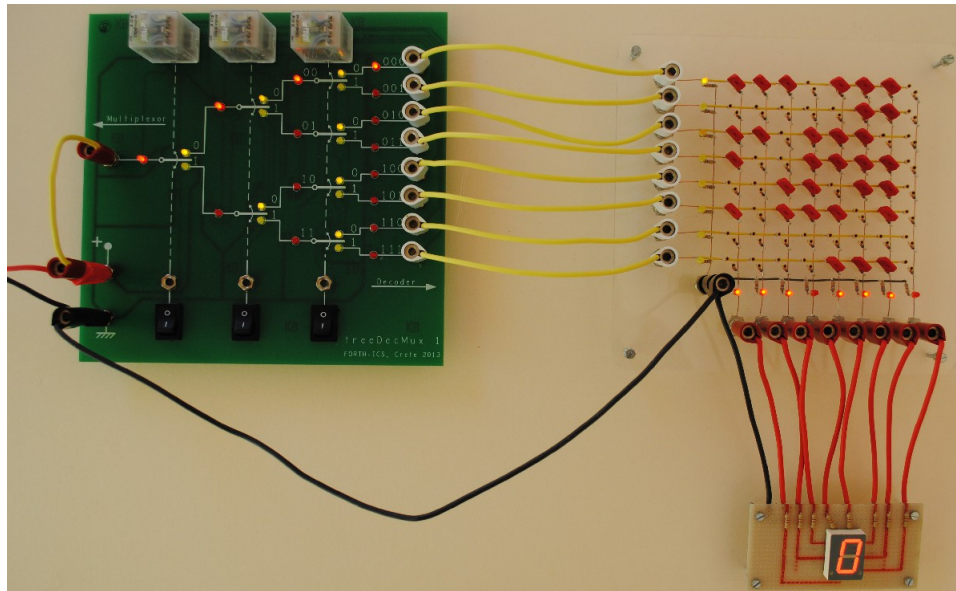


αποκωδικοποιούνται στο αριστερό μέρος του κυκλώματος στους οκτώ συνδυασμούς τους, 000, 001, 010,... έως 111. Ο ένας και *μόνον* ένας συνδυασμός που ισχύει ανά πάσα στιγμή (ο επιλεγμένος συνδυασμός) τροφοδοτεί με ρεύμα (επιλέγει) το αντίστοιχο οριζόντιο σύρμα (γραμμή) και την αντίστοιχη λέξη της μνήμης, και προκαλεί την *ανάγνωση* της λέξης αυτής, δηλαδή την εμφάνιση των bits της στα (εδώ τρία) κατακόρυφα σύρματα (εξόδους) ανάγνωσης δεδομένων.

Τα bits επιλογής της επιθυμητής λέξης, εδώ τα A, B, C, λέγονται **Διεύθυνση Μνήμης (memory address)**, και γι' αυτό το αριστερό μέρος του κυκλώματος το λέμε *αποκωδικοποιητή διευθύνσεων* (address decoder). Πρέπει να φανταζόμαστε τη μνήμη σαν ένα μακρύ δρόμο με N σπιτάκια, ένα για κάθε λέξη (εμείς τον δρόμο τον σχεδιάζουμε κατακόρυφα, με τα σπιτάκια το ένα κάτω από το άλλο, ίσως σαν ορόφους μιάς μεγάλης πολυκατοικίας ή ενός ουρανοξύστη), και τη Διεύθυνση Μνήμης να την φανταζόμαστε σαν την ταχυδρομική διεύθυνση του κάθε σπιτιού (της κάθε λέξης) --βασικά, ο αριθμός του κάθε σπιτιού πάνω σε αυτό το δρόμο. Στο δεκαδικό σύστημα αρίθμησης ταχυδρομικών διευθύνσεων, με τριψήφιους ($n=3$) αριθμούς σπιτιών μπορούμε να έχουμε $N = 10^3 = 1000$ σπιτάκια σε αυτό το δρόμο, με αριθμούς από το 000 μέχρι το 999. Στις ψηφιακές μνήμες, που λειτουργούν με δυαδικά bits, με διευθύνσεις των $n=3$ bits μπορούμε να έχουμε $N = 2^3 = 8$ λέξεις-σπιτάκια σε αυτή τη μνήμη-δρόμο.

Πείραμα 2.4: Η Μνήμη ROM με τον Αποκωδικοποιητή Διευθύνσεων

Στο Εργαστήριο, ενώστε τώρα την πλακέτα αποκωδικοποιητή 3-σε-8 του παραπάνω πειράματος 2.2 με τη μνήμη ROM (χειροκίνητης εγγραφής) του πειράματος 1.8, όπως δείχνει η φωτογραφία. Κάθε έξοδος του αποκωδικοποιητή συνδέεται και επιλέγει μία λέξη της μνήμης. Ο αποκωδικοποιητής έχει τη βασική ιδιότητα ότι *μία και μόνο μία* έξοδος του είναι αναμμένη ανά πάσα στιγμή. Αυτό ακριβώς χρειάζεται και η μνήμη στις εισόδους επιλογής της. Το κύκλωμα αυτό είναι ίδιο με το σχήμα της προηγούμενης παραγράφου 2.3, με μόνη τη διαφορά ότι εδώ η μνήμη έχει πλάτος 8 bits, αντί των τριών bits στην §2.3. Πρέπει να συνδέσετε τη γείωση της πλακέτας ROM στη γείωση της τροφοδοσίας (μαύρο καλώδιο στη φωτογραφία), αλλιώς το ρεύμα από την ROM και το 7-segment display δεν έχει πώς να γυρίσει στο τροφοδοτικό. Από την άλλη, η πλακέτα ROM δεν χρειάζεται καμιά ειδική θετική τροφοδοσία, διότι αυτή έρχεται μέσω του δέντρου του αποκωδικοποιητή, και από την εκάστοτε επιλεγόμενη έξοδό του περνάει στην επιλεγόμενη γραμμή της μνήμης, και από εκεί, από όσα "πριζάκια" (βραχυκυκλωτήρες) βρίσκει, στις αντίστοιχες στήλες της μνήμης και LED's του 7-segment display.



Πειραματιστείτε γράφοντας (χειροκίνητα) στη μνήμη διάφορα περιεχόμενα, για να εμφανίζονται στον ενδείκτη 7-τμημάτων άλλοτε τα ψηφία 0 έως 7, και άλλοτε άλλα γράμματα, σύμβολα, ή αριθμοί. Μετά από κάθε σύνολο τέτοιων εγγραφών, "διαβάστε" τα περιεχόμενα της μνήμης, με τη σειρά ή ανακατωμένα, εμφανίζοντάς τα στον ενδείκτη 7-τμημάτων. Φροντίστε να μάθετε καλά ποιός συνδυασμός των bits διεύθυνσης (address), A-B-C, επιλέγει ποιά λέξη (γραμμή) της μνήμης.

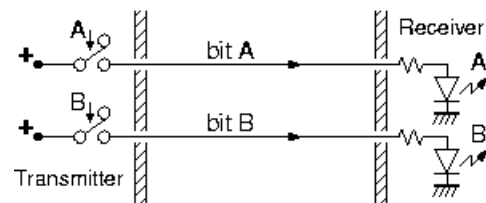
2.5 ROM και Συνδυαστικά Κυκλώματα από Πύλες AND-OR

Παρατηρήστε ότι στο παραπάνω κύκλωμα η μνήμη αποθηκεύει, για κάθε bit εξόδου (για κάθε λαμπίτσα του ενδείκτη 7-τμημάτων), σε ποιές περιπτώσεις (συνδυασμούς) των bits διεύθυνσης πρέπει αυτή η έξοδος να ανάβει (τιμή της συνάρτησης =1 ή αληθής), και σε ποιές περιπτώσεις να σβήνει. Αυτό, είναι ακριβώς το ίδιο με το τι δείχνει ο πίνακας αληθείας της συνάρτησης --μιλώντας για την κάθε έξοδο (λαμπίτσα) σαν συνάρτηση των (τριών) bits εισόδου (διεύθυνσης της μνήμης). Ουσιαστικά, είναι σαν η μνήμη να είναι ο πίνακας αληθείας των 8 συναρτήσεων των 8 εξόδων (LED's)· κάθε θέση της μνήμης που έχει "πριζάκι" αντιστοιχεί σε "1" στον πίνακα αληθείας, και αυτές που δεν έχουν αντιστοιχούν σε "0".

Αυτό μπορούμε να το εκφράσουμε και "επίσημα", με συναρτήσεις AND και OR. Ας πάρουμε σαν παράδειγμα την (αριστερή) έξοδο L2 στο σχήμα της προηγούμενης ενότητας. Αυτή "ανάβει" (χάρη στα πριζάκια W2, C2, F2, H2) στις περιπτώσεις: 000, 010, 101, και 111, δηλαδή στις περιπτώσεις: "000": (NOT A) AND (NOT B) AND (NOT C), "010": (NOT A) AND (B) AND (NOT C), "101": (A) AND (NOT B) AND (C), και "111": (A) AND (B) AND (C). Οι τέσσερις έξοδοι του αποκωδικοποιητή που τροφοδοτούν αυτές τις τέσσερις γραμμές υπολογίζουν ακριβώς αυτές τις τέσσερις συναρτήσεις AND. Ο πολυπλέκτης L2 της μνήμης, από την άλλη, υπολογίζει το OR αυτών των τεσσάρων περιπτώσεων, αφού ανάβει όποτε ανάβει οιαδήποτε από αυτές τις γραμμές. Άρα, συνολικά, η L2 ισούται με το OR των παραπάνω τεσσάρων συναρτήσεων AND. Στο Εργαστήριο 4 θα δούμε έναν μεθοδικό τρόπο να εξάγουμε και να απλοποιούμε τέτοιες συναρτήσεις.

2.6 Τα Bits και η Κωδικοποίηση Μηνυμάτων

Κυκλώματα σαν αυτά που είδαμε στα προηγούμενα εργαστήρια --ψηφιακά, όπως θα τα πούμε παρακάτω-- χρησιμοποιούνται για την αποθήκευση, επεξεργασία, και αποστολή κωδικοποιημένων μηνυμάτων και πληροφοριών. Στο σχήμα φαίνεται ένα απλό σύστημα μετάδοσης πληροφοριών από έναν πομπό (transmitter) σε έναν δέκτη (receiver): έχει δύο σύρματα, καθένα από τα οποία μπορεί να διαρρέεται ή όχι από ρεύμα.



Πληροφορία, εδώ, είναι το αν υπάρχει ή δεν υπάρχει ρεύμα στο κάθε σύρμα, πράγμα που ο πομπός το ρυθμίζει με τη θέση του κάθε διακόπτη, και ο δέκτης το καταλαβαίνει από το αν ανάβει ή όχι η κάθε LED. Κάθε σύρμα μεταδίδει μία ποσότητα πληροφορίας ικανή να επιλέξει ένα από δύο πράγματα, καταστάσεις, ή μηνύματα. Για το λόγο αυτό, την ποσότητα αυτή πληροφορίας την ονομάζουμε *δυναδικό ψηφίο (binary digit)* ή **δυφίο (bit)** --όνομα που αποτελεί τον πιό γνωστό όρο της Επιστήμης Υπολογιστών! Το σύστημα του σχήματος μεταδίδει δύο bits πληροφορίας από τον πομπό στο δέκτη, το bit A και το bit B.

Κάθε bit μεταδιδόμενης πληροφορίας μπορεί να έχει δύο μόνο διαφορετικές τιμές: από ηλεκτρική άποψη αυτές είναι απουσία ή παρουσία ρεύματος για τα μέχρι στιγμής κυκλώματά μας, ή χαμηλή ή υψηλή ηλεκτρική τάση για τα συνηθισμένα chips τύπου CMOS των σημερινών μικροηλεκτρονικών συστημάτων. Η **ερμηνεία** όμως που δίδεται στις δύο αυτές διαφορετικές τιμές είναι θέμα *σύμβασης* (συμφωνίας) ανάμεσα στον πομπό και το δέκτη, και είναι εντελώς *αυθαίρετη*, καθώς και *ανεξάρτητη* από τις δύο ηλεκτρικές τιμές του bit: πολλές φορές μάλιστα, η ερμηνεία αυτή είναι σκόπιμα κρυφή και μη αυτονόητη ώστε να αποφευχθούν υποκλοπές του μηνύματος, όπως π.χ. όταν χρησιμοποιούνται κώδικες κρυπτογραφίας. Προφανώς, για να υπάρξει επιτυχής επικοινωνία, πρέπει ο πομπός και ο δέκτης να έχουν προσυμφωνήσει στην ίδια ερμηνεία των ηλεκτρικών τιμών: πέραν των συμμετεχόντων στην επικοινωνία, όμως, δεν είναι ανάγκη κανείς άλλος να ξέρει ή να έχει συμφωνήσει με αυτή την ερμηνεία. Ο επόμενος πίνακας δίνει, σε δύο ζευγάρια στηλών, μερικά δημοφιλή ζευγάρια ερμηνείας των δύο τιμών ενός bit: τα δύο πρώτα ζευγάρια είναι οι συνηθισμένες ηλεκτρικές αναπαραστάσεις που είπαμε παραπάνω, και τα υπόλοιπα είναι μερικές δημοφιλείς ερμηνείες τους.

ρεύμα	όχι ρεύμα	ναι	όχι
ψηλή τάση	χαμηλή τάση	αληθές	ψευδές
πατημένος	ελεύθερος	1	0

ON	OFF	ενεργός	αδρανής
αναμμένος	σβηστός	θετικός	αρνητικός
πάνω	κάτω	αρνητικός	θετικός

Απ' όλα αυτά τα ζευγάρια, το πίο σύντομο στη γραφή είναι το "1, 0", γι' αυτό πολύ συχνά υιοθετούμε αυτά τα σύμβολα για τις δύο δυνατές τιμές ενός bit πληροφορίας. Ένα άλλο σχόλιο αρμόζει στα δύο τελευταία ζευγάρια: μερικές φορές μπορεί το ON ή το 1 να το ερμηνεύουμε σαν "θετικός", και μερικές σαν "αρνητικός", και αντίστροφα για το OFF ή το 0. Όταν βλέπουμε την έννοια "θετικός" σαν συναφή με τις έννοιες "ενεργός" ή "αληθής", έρχεται φυσικό να τις συμβολίσουμε όλες αυτές με την ίδια τιμή του bit (την τιμή "1"). Από την άλλη, όπως θα δούμε στο εργαστήριο 6, όταν παριστάνουμε προσημασμένους ακεραίους αριθμούς με τον "κώδικα συμπληρώματος-ως-προς-2", τότε το "αριστερό" bit της αναπαράστασης είναι 1 για τους αρνητικούς αριθμούς και 0 για τους μεγαλύτερους ή ίσους του μηδενός. Αυτές οι δύο, αντίθετες μεταξύ τους ερμηνείες της τιμής ενός bit δείχνουν καθαρά το πόσο σχετική και αυθαίρετη είναι η ερμηνεία αυτή, και πόσο αυτή είναι θέμα απλής σύμβασης μεταξύ αυτών που χρησιμοποιούν το bit.

Με ένα μόνο bit, πολύ μικρή ποσότητα πληροφορίας μπορούμε να μεταφέρουμε --μπορούμε να επιλέξουμε ανάμεσα σε ίσα-ίσα δύο μόνο προκαθορισμένα μηνύματα· για περισσότερες επιλογές (περισσότερη πληροφορία) χρειαζόμαστε και περισσότερα bits. Στο σύστημα του σχήματος στην αρχή της παραγράφου, ο πομπός μετέδιδε στο δέκτη δύο (2) bits πληροφορίας· ας δούμε τώρα πώς μπορούμε να τα εκμεταλλευτούμε αυτά. Ο πρώτος τρόπος εκμετάλλευσης πολλαπλών bits πληροφορίας είναι το **κάθε bit να έχει τη δική του, χωριστή ερμηνεία**· ας δούμε δύο παραδείγματα. Σαν πρώτο παράδειγμα, ας πούμε ότι ο πομπός του παραπάνω σχήματος βρίσκεται σε ένα δωμάτιο ξενοδοχείου, ο δέκτης βρίσκεται στην κουζίνα του ξενοδοχείου, το bit A σημαίνει "πεινάω - παρακαλώ φέρτε μου το πιάτο της ημέρας", και το bit B σημαίνει "δινάω - παρακαλώ φέρτε μου το ποτό της ημέρας". Τότε, οι 4 δυνατοί συνδυασμοί καταστάσεων των δύο διακοπών A και B μεταφέρουν ένα από τα εξής τέσσερα δυνατά μηνύματα στην κουζίνα:

- A=0, B=0: είμαι μιά χαρά - δεν θέλω τίποτα·
- A=0, B=1: παρακαλώ πολύ φέρτε μου ένα ποτό της ημέρας (χωρίς φαγητό)·
- A=1, B=0: παρακαλώ πολύ φέρτε μου ένα πιάτο της ημέρας (χωρίς ποτό)·
- A=1, B=1: παρακαλώ πολύ φέρτε μου ένα πιάτο και ένα ποτό της ημέρας.

Σαν δεύτερο παράδειγμα, ας πούμε ότι η επικοινωνία αυτή τη φορά είναι *οπτική* αντί ηλεκτρική όπως πριν: πομπός είναι τα πίσω φώτα φρένων και όπισθεν ενός αυτοκινήτου, και δέκτης είναι ο οδηγός του από πίσω αυτοκινήτου που βλέπει τα φώτα. Ας ονομάσουμε bit A την κατάσταση των φρένων (1=πατημένα, 0=ελεύθερα), και bit B την κατάσταση του λεβιέ ταχυτήτων (1=όπισθεν, 0=άλλη θέση). Τότε, οι 4 δυνατοί συνδυασμοί καταστάσεων των δύο bits πληροφορίας που μεταφέρουν τα κόκκινα και άσπρα πίσω φώτα δίνουν στον πίσω οδηγό ένα από τα εξής τέσσερα δυνατά μηνύματα:

- A=0, B=0: προχωρώ μπροστά, κανονικά·
- A=0, B=1: κάνω όπισθεν·
- A=1, B=0: προχωρώ μπροστά αλλά φρενάρω·
- A=1, B=1: φρενάρω και προτίθεμαι να κάνω όπισθεν·

Ο δεύτερος τρόπος εκμετάλλευσης πολλαπλών bits πληροφορίας είναι η κάθε ερμηνεία να αντιστοιχεί σε **όλα τα bits μαζί, σαν ομάδα**· ας δούμε το ανάλογο των δύο προηγουμένων παραδειγμάτων σε αυτό το στυλ. Στην περίπτωση του ξενοδοχείου, μπορεί τα δύο bits να χρησιμοποιούνται για την παραγγελία ενός φαγητού από ένα μικρό μενού· ελλείψει περισσοτέρων bits, δεν υπάρχει δυνατότητα

μεγαλύτερου μενού ή συνδυασμού πολλαπλών παραγγελιών (φαγητό, σαλάτα, ποτό, κλπ):

- $A=0, B=0$: είμαι μιά χαρά - δεν θέλω τίποτα.
- $A=0, B=1$: παρακαλώ πολύ φέρτε μου ένα σάντουιτς.
- $A=1, B=0$: παρακαλώ πολύ φέρτε μου μία μακαρονάδα.
- $A=1, B=1$: παρακαλώ πολύ φέρτε μου μία πριζόλα.

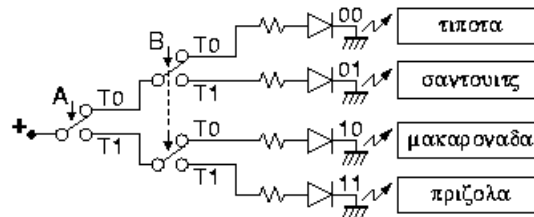
Όπως βλέπουμε εδώ, δεν υπάρχει ερμηνεία για το καθένα bit χωριστά από το άλλο: δεν μπορεί να ερμηνευτεί το bit A σαν μακαρονάδα και το bit B σαν σάντουιτς, διότι η πριζόλα δεν αποτελεί συνδυασμό... σάντουιτς με μακαρονάδα. Στο δεύτερο παράδειγμα, ας πούμε ότι τώρα το bit A είναι το πίσω αριστερό πορτοκαλί φώς (φλας) του αυτοκινήτου, και το bit B είναι το πίσω δεξί φλας. Σε αυτή την περίπτωση, η πληροφορία προς τον πίσω οδηγό είναι η παρακάτω (εδώ, "0" σημαίνει σβηστό φώς, και "1" σημαίνει ότι το φώς αναβοσβήνει). Όπως και στο τελευταίο παράδειγμα, το bit A δεν μπορεί να ερμηνευτεί πάντα σαν "στρίβω αριστερά" και το bit B σαν "στρίβω δεξιά", διότι ο συνδυασμός $A=1, B=1$ δεν σημαίνει "στρίβω αριστερά και δεξιά".

- $A=0, B=0$: προχωρώ ίσια.
- $A=0, B=1$: θα στρίψω δεξιά.
- $A=1, B=0$: θα στρίψω αριστερά.
- $A=1, B=1$: πρόσεχε - προσπαθώ να βρώ πού θα παρκάρω.

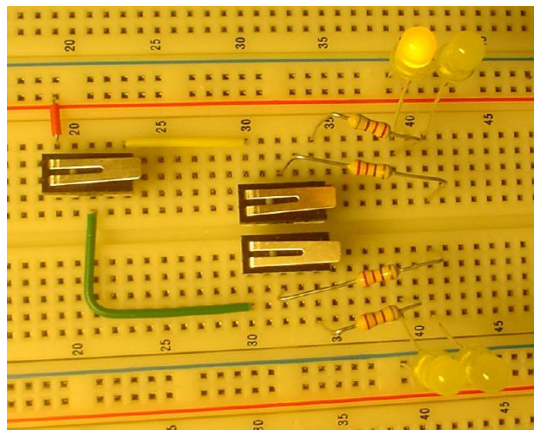
Τέλος, ας κάνουμε κι ένα πίο σύνθετο παράδειγμα: το ξενοδοχείο αποφασίζει να αναβαθμίσει τις υπηρεσίες εστιατορίου του και εγκαθιστά σε κάθε δωμάτιο 5 διακόπτες και 5 σύρματα προς την κουζίνα, μεταδίδοντας έτσι 5 bits πληροφορίας, A, B, C, D, και E. Τα 3 πρώτα bits, A, B, C χρησιμοποιούνται για την παραγγελία φαγητού, και τα 2 υπόλοιπα για την παραγγελία ποτού. Οι 8 συνδυασμοί τιμών των bits ABC αποφασίζεται να σημαίνουν: (000) δεν θέλω φαγητό, (001) σάντουιτς, (010) μακαρονάδα, (011) γεμιστά, (100) μουσακά, (101) ψάρι, (110) μπιφτέκια, (111) μπριζόλα. Οι 4 συνδυασμοί τιμών των bits DE αποφασίζεται να σημαίνουν: (00) δεν θέλω ποτό, (01) μπύρα, (10) κρασί, (11) ούζο. Έτσι, όταν η κουζίνα βλέπει ABCDE = 00011 στέλνει ένα σκέτο ουζάκι, ενώ όταν βλέπει 10110 στέλνει ένα ψάρι με κρασί. Συνολικά, υπάρχουν 32 ($=8 \times 4$) δυνατά μηνύματα: 1 μήνυμα ότι ο πελάτης δεν θέλει τίποτα (00000), 7 μηνύματα φαγητού χωρίς ποτό, 3 μηνύματα ποτού χωρίς φαγητό, και 21 ($=7 \times 3$) συνδυασμοί κάποιου φαγητού με κάποιο ποτό.

Πείραμα 2.7: Αποκωδικοποιητής 2-σε-4 με Διακόπτες

Ο μάγειρας του προηγούμενου ξενοδοχείου δεν μπορούσε ποτέ να θυμηθεί ποιός από τους κώδικες 01, 10, και 11 αντιστοιχούσε στη μακαρονάδα, ποιός στην μπιρζόλα, και ποιός στο σάντουιτς (γιά να περιοριστούμε απλώς στα προ της αναβάθμισης). Γιά να τον βοηθήσετε, φτιάξτε το κύκλωμα αποκωδικοποιητή που φαίνεται δεξιά. Το κύκλωμα αυτό έχει δύο εισόδους, τα bits A και B· το bit A ελέγχει το διακόπτη A. Το bit B ελέγχει τους δύο δεξιούς διακόπτες που φαίνονται στο σχήμα: αυτοί πρέπει πάντα να αναβοσβήνουν και οι δύο μαζί. Παρ' ότι υπάρχουν και διπλοί διακόπτες (DPDT - double pole double throw), εμείς στο εργαστήριο δεν έχουμε τέτοιους, γι' αυτό θα χρησιμοποιήσετε δύο απλούς SPDT βαλμένους δίπλα-δίπλα όπως φαίνεται στη φωτογραφία ώστε να πατιούνται κι οι δύο μαζί μ' ένα δάκτυλο.



Ο συγκεκριμένος αποκωδικοποιητής εδώ είναι μεγέθους **2-σε-4**, δηλαδή αποκωδικοποιεί 2 εισόδους στους 4 συνδυασμούς τους, άρα έχει 4 εξόδους· το κύκλωμα εδώ έχει σαν εξόδους 4 LED's. Εάν πάνω από κάθε LED βάλουμε ένα ημιδιαφανές πλαστικό με γραμμένο επάνω το αντίστοιχο μήνυμα, όπως στο σχήμα δεξιά, θα έχουμε προσφέρει την επιθυμητή βοήθεια στο μάγειρα του ξενοδοχείου.



Φτιάξτε τον πίνακα αληθείας γιά καθεμιά από τις 4 εξόδους του κυκλώματος. Επίσης, γιά την κάθε έξοδο παρατηρήστε το κύκλωμα που την τροφοδοτεί (δύο διακόπτες εν σειρά) και εκφράστε την με μιά "εξίσωση" (π.χ. (A)ΚΑΙ(ΟΧΙ(B))), βάσει των όσων είπαμε στην §1.2. Φτιάξτε και ελέγξτε το κύκλωμα.

2.8 Πλήθος Συνδυασμών (Μηνυμάτων) των n bits

Σε πόσους διαφορετικούς συνδυασμούς τιμών μπορεί να βρεθεί μιά ομάδα από n το πλήθος bits; Με άλλα λόγια, πόσα διαφορετικά μηνύματα μπορούμε να κωδικοποιήσουμε αν έχουμε στη διάθεσή μας n bits; Ή, έχοντας μιά ομάδα n bits, ανάμεσα σε πόσα πολλά διαφορετικά πράγματα μπορούμε να επιλέξουμε (υποδείξουμε) ένα; Ξέρουμε ήδη ότι ένα bit έχει δύο δυνατές τιμές. Επίσης ξέρουμε ότι δύο bits μπορούν να βρίσκονται σε έναν από 4 διαφορετικούς συνδυασμούς τιμών: γιά την κάθε μιά από τις 2 τιμές του πρώτου υπάρχουν 2 διαφορετικοί συνδυασμοί με τις 2 διαφορετικές τιμές του δεύτερου.

Γενικότερα, κάθε φορά που προσθέτουμε **άλλο ένα bit** στην ομάδα, **διπλασιάζεται** το πλήθος των συνδυασμών: γιά την τιμή 0 του νέου bit έχουμε τους συνδυασμούς - κώδικες - μηνύματα που είχαμε και πριν βάσει των υπολοίπων bits, και γιά την τιμή 1 του νέου bit έχουμε άλλους τόσους νέους συνδυασμούς, πάλι βάσει των υπολοίπων bits. Έτσι προκύπτει ότι τα n bits μπορούν να βρίσκονται σε 2^n διαφορετικούς συνδυασμούς τιμών, ή με n bits μπορούμε να διαλέξουμε ένα ανάμεσα σε 2^n πράγματα, ή να κωδικοποιήσουμε ένα ρεπερτόριο 2^n διαφορετικών επιτρεπών μηνυμάτων. Γι' αυτό, οι δυνάμεις του 2 παίζουν κεφαλαιώδη ρόλο στους υπολογιστές,

και θα τις βρίσκουμε μπροστά μας συνεχώς:

- **1 bit** μπορεί να επιλέξει ένα ανάμεσα σε $2^1 = 2$ διαφορετικά πράγματα/συνδυασμούς/μηνύματα,
- **2 bits** μπορούν να επιλέξουν ένα ανάμεσα σε $2^2 = 4$ διαφορετικά πράγματα/συνδυασμούς,
- **3 bits** μπορούν να επιλέξουν ένα ανάμεσα σε $2^3 = 8$ διαφορετικά πράγματα/συνδυασμούς,
- **4 bits** επιλέγουν ανάμεσα σε $2^4 = 16$ διαφορετικά πράγματα/συνδυασμούς/μηνύματα,
- **5 bits** επιλέγουν ανάμεσα σε $2^5 = 32$ διαφορετικά πράγματα/συνδυασμούς/μηνύματα,
- **6 bits** επιλέγουν ανάμεσα σε $2^6 = 64$ διαφορετικά πράγματα/συνδυασμούς/μηνύματα,
- **7 bits** επιλέγουν ανάμεσα σε $2^7 = 128$ διαφορετικά πράγματα/συνδυασμούς/μηνύματα,
- **8 bits** επιλέγουν ανάμεσα σε $2^8 = 256$ διαφορετικά πράγματα/συνδυασμούς/μηνύματα,
- **9 bits** επιλέγουν ανάμεσα σε $2^9 = 512$ διαφορετικά πράγματα/συνδυασμούς/μηνύματα,
- **10 bits** επιλέγουν ανάμεσα σε $2^{10} = 1024 = 1 \text{ K}$ (Kilo) διαφορετικά πράγματα,
- **11 bits** επιλέγουν ανάμεσα σε $2^{11} = 2048 = 2 \text{ K}$ διαφορετικά πράγματα,
- **12 bits** επιλέγουν ανάμεσα σε $2^{12} = 4096 = 4 \text{ K}$ διαφορετικά πράγματα,
- **13 bits** επιλέγουν ανάμεσα σε $2^{13} = 8192 = 8 \text{ K}$ διαφορετικά πράγματα,
- **14 bits** επιλέγουν ανάμεσα σε $2^{14} = 16,384 = 16 \text{ K}$ διαφορετικά πράγματα,
- **15 bits** επιλέγουν ανάμεσα σε $2^{15} = 32,768 = 32 \text{ K}$ διαφορετικά πράγματα,
- **16 bits** επιλέγουν ανάμεσα σε $2^{16} = 65,536 = 64 \text{ K}$ διαφορετικά πράγματα, ...
- **20 bits** επιλέγουν ανάμεσα σε $2^{20} = 1,048,576 = 1 \text{ M}$ (Mega) διαφορετικά πράγματα, ...
- **30 bits** επιλέγουν ανάμεσα σε $2^{30} = 1,073,741,824 = 1 \text{ G}$ (Giga) διαφορετικά πράγματα, ...
- **40 bits** επιλέγουν ανάμεσα σε $2^{40} = 1,099,511,627,776 = 1 \text{ T}$ (Tera) διαφορετικά πράγματα, κ.ο.κ.

Τα bits πληροφορίας, εκτός από το να τα μεταδίδουμε ή επεξεργαζόμαστε, τα αποθηκεύουμε επίσης, σε **μνήμες** (memories), για τις οποίες θα μιλήσουμε αργότερα. Όταν μιλάμε για τις μνήμες και τη χωρητικότητά τους, δεν πρέπει να συγχέουμε το πόσα bits χωράνε με το πόσους συνδυασμούς μπορεί να παραστήσει ένα πλήθος από bits. Για παράδειγμα, έστω ότι το πληροφοριακό σύστημα ενός Πανεπιστημίου κωδικοποιεί τον αριθμό μητρώου του κάθε φοιτητή με 24 bits. Αυτό σημαίνει ότι στο πληροφοριακό αυτό σύστημα δεν μπορούν να χωρέσουν πάνω από περίπου 16 εκατομμύρια (16 M) φοιτητές (παρελθόντων και παρόντων ετών) (16,777,216 φοιτητές, για την ακρίβεια). Αν τώρα ένας υπολογιστής στη γραμματεία αυτού του Πανεπιστημίου έχει μία (μάλλον μικρή) μνήμη 16 Mbits, αυτό σημαίνει ότι στη μνήμη αυτή χωράνε να αποθηκευτούν μέχρι περίπου 16 εκατομμύρια διαφορετικά bits (16,777,216 bits για την ακρίβεια). Τα bits αυτά, αν τα βλέπαμε σαν μία μόνο ομάδα, μπορούν να βρεθούν σε ένα πλήθος συνδυασμών τόσο τεράστιο που ούτε καν να το φανταστούμε μπορούμε (κάπου γύρω στο 1.6-εκατομμυριάκις εκατομμύριο...). Όμως, τα bits αυτά δεν τα κοιτάμε ποτέ σαν μία μόνο ομάδα, αλλά σαν πολλές: για παράδειγμα, αν σε αυτά αποθηκεύσουμε αριθμούς μητρώου φοιτητών, "κολλητά" τον έναν με τον άλλον, τότε θα χωρέσουν γύρω στις 700 χιλιάδες τέτοιοι αριθμοί

μητρώου (για την ακρίβεια, $16777216 / 24 = 699050.67$ αριθμοί μητρώου).

Ένα σκέτο bit μπορεί να μεταφέρει πολύ μικρή ποσότητα πληροφορίας, γι' αυτό συχνά τα bits τα χρησιμοποιούμε κατά ομάδες, όπως είπαμε παραπάνω. Μεταξύ των διαφόρων δυνατών μεγεθών ομάδων, η πιο συνηθισμένη, σε όλους ανεξαιρέτα τους σημερινούς υπολογιστές, είναι τα **οκτώ (8) bits** που ονομάζονται ένα **Byte**. Έτσι, η κωδικοποίηση του κάθε αριθμού μητρώου στο προηγούμενο παράδειγμα ήταν σε 3 Bytes (= $3 \times 8 = 24$ bits). Σαν σύντμηση, το "b" μικρό συμβολίζει το bit, και το "B" κεφαλαίο συμβολίζει το Byte.

Ας θεωρήσουμε τώρα έναν υπολογιστή που έχει μνήμη 256 MBytes· αυτό σημαίνει $256 \times 8 = 2048$ Mbits = 2 Gbits. Από τις μνήμες σαν αυτήν, θέλουμε να ζητάμε να διαβάσουμε ορισμένα κομμάτια τους που μας ενδιαφέρουν κατ' επιλογή· συνήθως, το κομμάτι που μας ενδιαφέρουν είναι ένα Byte, και όχι μεμονωμένα bits διότι αυτά θεωρούνται πολύ μικρά. Κάθε φορά που θέλουμε να διαβάσουμε λοιπόν από τη μνήμη μας των 2 Gb = 256 MB, πρέπει να της διευκρινίσουμε ποιο από τα 256 εκατομμύρια Bytes που αυτή περιέχει εμείς θέλουμε να διαβάσουμε. Πόσα bits χρειαζόμαστε για να επιλέξουμε ένα ανάμεσα σε 256 M πράγματα; Σύμφωνα με τα παραπάνω, χρειαζόμαστε 28 bits. Τα 28 αυτά bits που πρέπει να δώσουμε στη μνήμη τα λέμε **διεύθυνση** του Byte της το οποίο επιθυμούμε να επιλέξουμε: κάθε Byte της μνήμης έχει τη δική του διεύθυνση, σαν να είναι ένα σπιτάκι σε έναν πολύ μακρύ δρόμο.

Στις μονάδες της Φυσικής, ο πολλαπλασιαστής "k" (μικρό) σημαίνει 1000· π.χ. 1 kg = 1000 g, 1 km = 1000 m, 1 kHz = 1000 Hz. Όταν μιλάμε για bits ή για Bytes, ο πολλαπλασιαστής "K" (κεφαλαίο) σημαίνει 1024, όπως τον ορίσαμε παραπάνω· π.χ. 1 Kb = 1024 bits, 1 KB = 1024 Bytes. Εκεί που τα πράγματα είναι διαφορετικά, είναι με τους πολλαπλασιαστές "M" και "G" (κεφαλαία): αυτοί, άλλοτε σημαίνουν 1,000,000 και 1,000,000,000 αντίστοιχα, μπροστά από τις παραδοσιακές φυσικές μονάδες, και άλλοτε σημαίνουν 1,048,576 και 1,073,741,824 αντίστοιχα, μπροστά από τα bits και τα Bytes που αφορούν χωρητικότητα μνημών. Έτσι, 1 MHz = 10^6 Hz ενώ 1 Mb = 2^{20} bits, και 1 GHz = 10^9 Hz ενώ 1 GB = 2^{30} Bytes. (Τα πράγματα χειροτερεύουν όταν μιλάμε για ταχύτητες δικτύων υπολογιστών: συνήθως, 1 Mb/s = 10^6 bits/second και 1 Gb/s = 10^9 b/s, επειδή οι ταχύτητες αυτές πηγάζουν από ρολόγια του 1 MHz ή 1 GHz!...).

2.9 Αναλογικά και Ψηφιακά Ηλεκτρονικά Συστήματα

Ας σκεφτούμε το εξής παράδειγμα. Μία εταιρεία εμπορίας κατεψυγμένων ειδών θέλει να παρακολουθεί εξ' αποστάσεως τη θερμοκρασία του ψυγείου της, προκειμένου να εντοπίζει γρήγορα τυχόν βλάβες. Το ηλεκτρονικό θερμόμετρο που υπάρχει μέσα στο ψυγείο έχει περιοχή λειτουργίας από -25°C έως $+25^{\circ}\text{C}$, και ακρίβεια $\pm 0.1^{\circ}\text{C}$. επομένως, οι μετρήσεις του που έχει νόημα να μεταφέρονται είναι -25.0°C , -24.8°C , -24.6°C , ..., $+24.8^{\circ}\text{C}$, $+25.0^{\circ}\text{C}$. Η μέτρηση θα μεταφέρεται από το ψυγείο ως το γραφείο του φύλακα μέσω ηλεκτρικών καλωδίων. Υπάρχουν δύο (τουλάχιστο) τρόποι μετάδοσης αυτής της μέτρησης:

Η **αναλογική** (analog) μετάδοση λειτουργεί περίπου ως εξής: χρησιμοποιούμε ένα ηλεκτρικό σύρμα (συν την αναγκαία γείωση για να κλείνει κύκλωμα), και πάνω σε αυτό βάζουμε μίαν ηλεκτρική τάση ίση με τη θερμοκρασία επί έναν συντελεστή αναλογίας π.χ. $0.1 \text{ V}/^{\circ}\text{C}$. επειδή η τάση που μεταδίδουμε είναι ανάλογη προς την θερμοκρασία, η μετάδοση λέγεται "αναλογική". Π.χ. αν η μέτρηση του θερμόμετρου είναι -18.4°C τότε στο σύρμα θα βάλουμε μία τάση -1.84 Volt . Αφού το θερμόμετρο έχει περιοχή λειτουργίας από -25°C έως $+25^{\circ}\text{C}$, τα ηλεκτρονικά κυκλώματα που οδηγούν το σύρμα θα πρέπει να μπορούν να ρυθμίζουν την τάση του από -2.5 V έως

+ 2.5 V. Προκειμένου να μην υπάρχει απώλεια ακρίβειας στη μετάδοση της μέτρησης, θα πρέπει το **συνολικό σφάλμα** των ηλεκτρονικών μετάδοσης να μην υπερβαίνει τα ± 0.01 Volt (δηλ. ± 10 mV). Παραδείγματος χάριν, έστω ότι η μέτρηση είναι -18.4°C , και θα έπρεπε να μεταδώσουμε -1.840 V, αλλά ο πομπός έχει σφάλμα $+3$ mV κι έτσι στην πραγματικότητα μεταδίδει -1.837 V· σε αυτό προστίθεται ηλεκτρικός θόρυβος $+8$ mV από παρεμβολές κατά μήκος του σύρματος μετάδοσης, κι έτσι στον δέκτη φτάνει τάση -1.829 V· ο δέκτης έχει σφάλμα $+2$ mV, κι έτσι νομίζει ότι βλέπει -1.827 V· ξέροντας ότι οι μετρήσεις είναι ακέραια πολλαπλάσια του 0.2°C , ο δέκτης ερμηνεύει την τάση που (νομίζει ότι) βλέπει σαν -18.2°C αντί του σωστού -18.4°C . Το λάθος συνέβη επειδή το συνολικό σφάλμα του συστήματος μετάδοσης είναι $3+8+2 = 13$ mV που ξεπερνά την επιτρεπτή ανοχή του ± 0.01 V που αντιστοιχεί στην ανοχή της μέτρησης θερμοκρασίας των $\pm 0.1^{\circ}\text{C}$.

Η **ψηφιακή** (digital) μετάδοση λειτουργεί περίπου ως εξής: χρησιμοποιούμε οκτώ ηλεκτρικά σύρματα (συν την αναγκαία γείωση για να κλείνει κύκλωμα), και πάνω σε αυτά τα σύρματα βάζουμε 8 bits πληροφορίας που αποτελούν την κωδικοποίηση της θερμοκρασίας σύμφωνα με έναν κώδικα που εμείς αποφασίσαμε· επειδή ο κώδικας αποτελείται από "ψηφία" (digits), η μετάδοση λέγεται "ψηφιακή". Υπάρχουν 251 διαφορετικές δυνατές μετρήσεις θερμοκρασίας από το -25°C έως το $+25^{\circ}\text{C}$ ανά 0.2°C ($50^{\circ}\text{C} / 0.2^{\circ}\text{C} = 250$), επομένως ξέρουμε ότι 8 bits αρκούν για την κωδικοποίηση ενός τέτοιου ρεπερτορίου μηνυμάτων, αφού 8 bits έχουν 256 διαφορετικούς συνδυασμούς. Για τη μετάδοση του καθενός bit πληροφορίας, ως χρησιμοποιήσουμε μιά ηλεκτρική τάση 0.0 V για την τιμή OFF και $+5.0$ V για την τιμή ON· η περιοχή αυτή τάσεων είναι η συνηθισμένη στα κυκλώματα του εργαστηρίου μας, και το εύρος της είναι 5 Volt, όσο δηλαδή και το εύρος λειτουργίας των αναλογικών ηλεκτρονικών στο προηγούμενο παράδειγμα (από -2.5 V έως $+2.5$ V). Προκειμένου να μην υπάρξει λάθος στη μετάδοση, τώρα, πρέπει το **συνολικό σφάλμα** των ηλεκτρονικών μετάδοσης να μην υπερβαίνει τα ± 2.50 Volt (δηλ. ± 2500 mV). Παραδείγματος χάριν, έστω ότι το bit που θέλουμε να μεταδώσουμε είναι OFF, άρα πρέπει να μεταδώσουμε 0.0 V, αλλά ο πομπός έχει σφάλμα $+400$ mV κι έτσι στην πραγματικότητα μεταδίδει $+0.4$ V· σε αυτό προστίθεται ηλεκτρικός θόρυβος $+1500$ mV από παρεμβολές κατά μήκος του σύρματος μετάδοσης, κι έτσι στον δέκτη φτάνει τάση $+1.9$ V· ο δέκτης έχει σφάλμα $+200$ mV, κι έτσι νομίζει ότι βλέπει $+2.1$ V. Ξέροντας όμως ότι οι αναμενόμενες τιμές του bit είναι είτε 0.0 V είτε 5.0 V, και δεδομένου ότι τα $+2.1$ V που (νομίζει ότι) βλέπει είναι πιά κοντά στο 0 απ' ό,τι στο 5, ερμηνεύει σωστά το bit που λαμβάνει σαν OFF και όχι σαν ON.

Βλέπουμε ότι σε αυτό το παράδειγμα το ψηφιακό σύστημα στέλνει οκτώ (8) ηλεκτρικές τάσεις σαν πληροφορία, αντί της μίας (1) μόνο ηλεκτρικής τάσης που στέλνει το αναλογικό, αλλά το ψηφιακό σύστημα ανέχεται διακόσιες πενήντα (250) φορές περισσότερο θόρυβο και έλλειψη ακρίβειας στις τάσεις των ηλεκτρονικών κυκλωμάτων λειτουργίας του απ' όσο το αναλογικό (± 2500 mV αντί ± 10 mV). Η τεράστια εξάπλωση των ψηφιακών ηλεκτρονικών συστημάτων, σήμερα, οφείλεται στα εξής πλεονεκτήματά τους:

- **Ανοχή στο θόρυβο:** όπως εξηγήσαμε με το παραπάνω παράδειγμα, τα ψηφιακά συστήματα μπορούν να ανεχθούν τόσο πολύ θόρυβο και ανακρίβειες στη λειτουργία τους ώστε μπορούν να καταστούν σχεδόν "αλάνθαστα", σε αντίθεση με τα αναλογικά όπου όλο και κάποιος θόρυβος τελικά παρεισφρέει.
- **Χαμηλό κόστος** των ψηφιακών ηλεκτρονικών κυκλωμάτων σε σύγκριση με αναλογικά που θα έκαναν αντίστοιχη επεξεργασία σήματος: αναλογικά ηλεκτρονικά κυκλώματα με χαμηλό θόρυβο και υψηλή γραμμικότητα (χαμηλή παραμόρφωση) είναι πολύ δυσκολότερο να κατασκευαστούν απ' ό,τι ψηφιακά κυκλώματα που κάνουν την αντίστοιχη επεξεργασία μέσω αριθμητικών

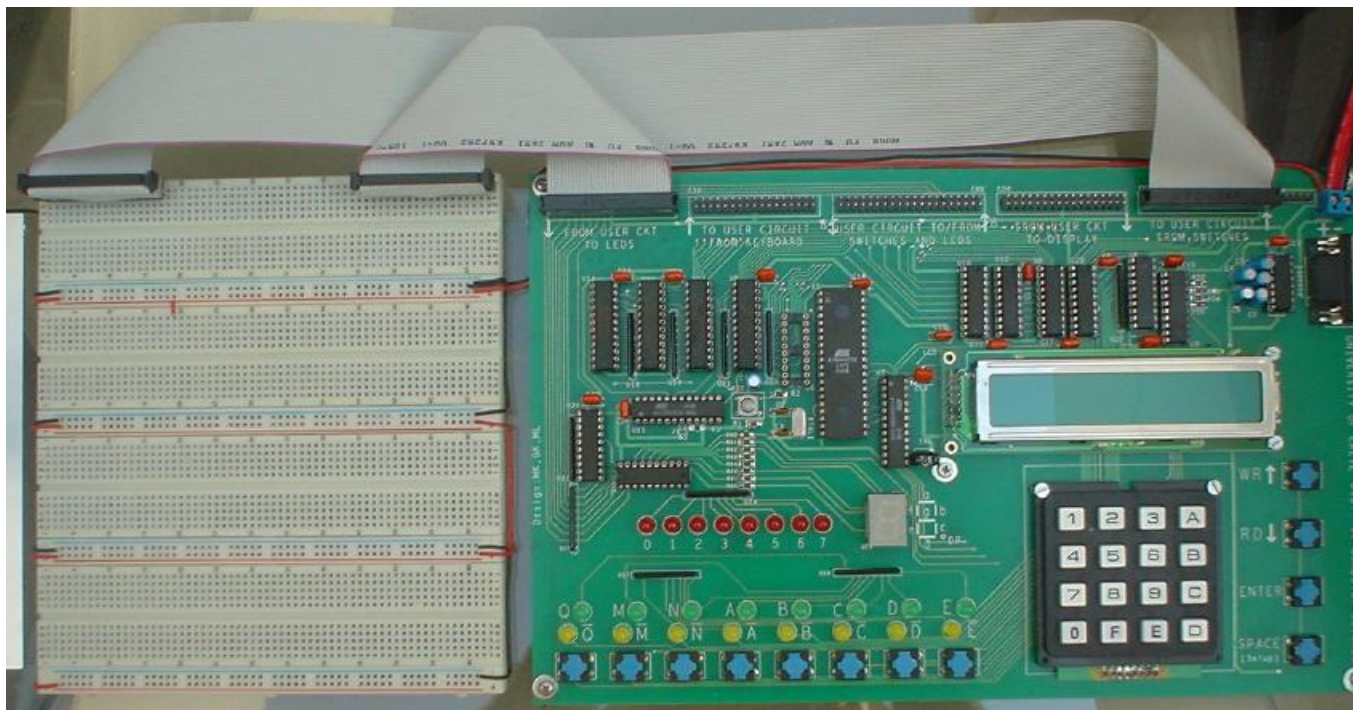
πράξεων πάνω σε κατάλληλα κωδικοποιημένες πληροφορίες.

- **Αναπαράσταση εγγενώς διακριτών πληροφοριών:** πέραν των εγγενώς αναλογικών πληροφοριών (π.χ. ήχος και εικόνα), υπάρχουν και εγγενώς διακριτές πληροφορίες όπως π.χ. το κείμενο. Θα ήταν εξαιρετικά δύσκολο και αφύσικο να παριστάνουμε κάθε γράμμα της αλφαβήτου μέσω μίας διαφορετικής ηλεκτρικής τάσης, η δε λογική επεξεργασία του κειμένου ή των πληροφοριών που αυτό παριστά θα ήταν αδύνατη με αναλογικό τρόπο.

Transistors και Διακόπτες: Τα ψηφιακά συστήματα κατασκευάζονται σήμερα σε μορφή μικροηλεκτρονικών chips (IC - integrated circuit - ολοκληρωμένο κύκλωμα) που περιέχουν το καθένα χιλιάδες ή εκατομμύρια transistors. Τα transistors αυτά, όταν λειτουργούν ψηφιακά, συμπεριφέρονται σαν διακόπτες, που άλλοτε κάνουν επαφή (ανάβουν) και άλλοτε την διακόπτουν (σβήνουν). Φυσικά, δεν υπάρχει κανένα μαγικό χέρι που να αναβοσβήνει αυτούς τους διακόπτες --αυτοί ανοιγοκλείνουν υπο την επίδραση (ψηφιακών) ηλεκτρικών τάσεων. Για το λόγο αυτό, ξεκινήσαμε τη μελέτη των ψηφιακών συστημάτων μελετώντας απλούς, καθημερινούς διακόπτες.

2.10 Συνδέσεις με την Πλακέτα Εισόδων/Εξόδων

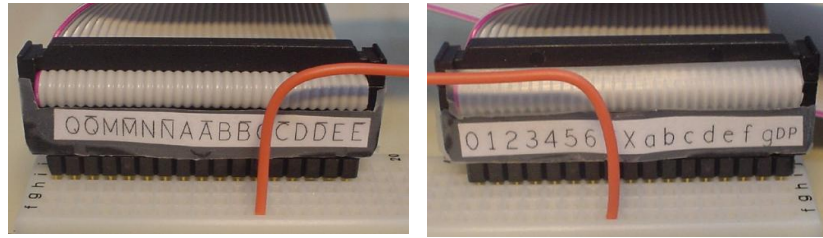
Στα περισσότερα πειράματα από εδώ και πέρα θα χρησιμοποιήσουμε εισόδους από τους διακόπτες και εξόδους προς τις ενδεικτικές λυχνίες της πλακέτας εισόδων - εξόδων, μέσω των καλωδιοταινιών που φαίνονται στο επάνω μέρος της φωτογραφίας, εδώ.



Η πρώτη (μακρύτερη) καλωδιοταινία, που καταλήγει στην πάνω αριστερή γωνία της πλακέτας συνδέσεων, φέρνει εισόδους προς το κύκλωμα του χρήστη από τους διακόπτες Q, M, N, A, B, C, D, και E της πλακέτας εισόδων/εξόδων. Η δεύτερη (κοντή) καλωδιοταινία, που καταλήγει στην πάνω δεξιά γωνία της πλακέτας συνδέσεων, παραλαμβάνει εξόδους από το κύκλωμα του χρήστη και τις οδηγεί στις ενδεικτικές λυχνίες 0 έως 7 και a έως g και DP της πλακέτας εισόδων/εξόδων. Οι καλωδιοταινίες καταλήγουν σε ακροδέκτες τοποθετημένους στις υποδοχές της πλακέτας συνδέσεων. Κάθε ζευγάρι ακροδεκτών που βρίσκονται στην ίδια πεντάδα υποδοχών μεταφέρουν το ίδιο σήμα (είναι βραχυκυκλωμένοι μεταξύ τους): ζευγάρια σε διαφορετικές πεντάδες μεταφέρουν διαφορετικά σήματα. Για να κάνουμε

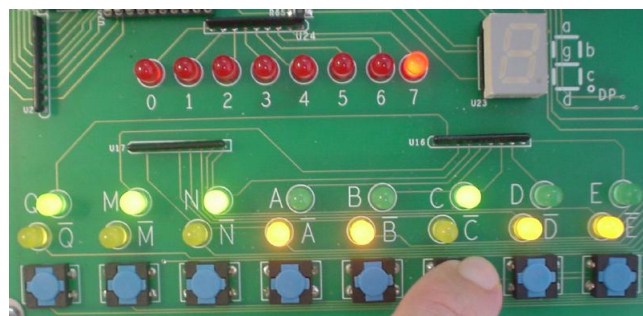
συνδέσεις στους διακόπτες και τις LED's μέσω των καλωδιοταινιών, χρησιμοποιούμε τις υπόλοιπες τρεις υποδοχές της κάθε πεντάδας, στην πλακέτα συνδέσεων. **Μην πειράζετε** τις καλωδιοταινίες και τους ακροδέκτες τους: φθείρονται εύκολα, λυγίζουν και σπάνε.

Στις άκρες των καλωδιοταινιών υπάρχουν ετικέτες που δείχνουν σε ποίο σήμα αντιστοιχεί ο κάθε ακροδέκτης,



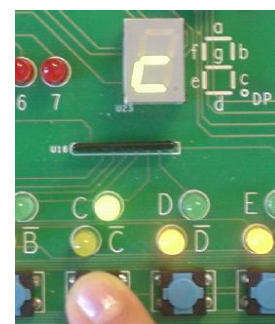
περίπου όπως στις φωτογραφίες δεξιά. Στις φωτογραφίες αυτές έχουμε βάλει ένα σύρμα που συνδέει την είσοδο από τον διακόπτη C στην έξοδο προς την LED υπ' αριθμό 7. Στην καλωδιοταινία των διακοπών υπάρχει μία αχρησιμοποίητη θέση, αριστερά: στην ταινία των LED η αχρησιμοποίητη θέση είναι στη μέση, σημειωμένη με "X" στη φωτογραφία. Ο κάθε διακόπτης τροφοδοτεί **δύο σήματα**, σε δύο πεντάδες: ένα με το ίδιο το σήμα του διακόπτη, π.χ. Q, και ένα με το αντίστροφό του, **OXI(Q)**, που σημειώνεται με το όνομα του διακόπτη με μία παύλα από πάνω· αυτά τα αντίστροφα των σημάτων εισόδου θα διευκολύνουν τα κυκλώματά μας σε επόμενα εργαστήρια.

Η επόμενη φωτογραφία, δεξιά, δείχνει την περιοχή διακοπών και ενδεικτικών λυχνιών στο εμπρός αριστερά μέρος της πλακέτας εισόδων/εξόδων που θα χρησιμοποιήσουμε σε αυτό και τα επόμενα μερικά εργαστήρια· δεν θα χρησιμοποιήσουμε ακόμα το μικρό πληκτρολόγιο 4x4 και την



οθόνη υγρών κρυστάλλων στο δεξιό μέρος της πλακέτας. Οι κόκκινες LED οδηγούνται από τα ομώνυμά τους σήματα στην άκρη της καλωδιοταινίας που είδαμε παραπάνω: εάν υπάρχει το σύρμα από τον διακόπτη C στην LED 7 του προηγούμενου ζεύγους φωτογραφιών, τότε κάθε φορά που πατάμε τον διακόπτη C, όπως στην εδώ φωτογραφία, ανάβει η κόκκινη LED υπ' αριθμόν 7. Όσες κόκκινες LED έχουν το σήμα τους ασύνδετο παραμένουν σβηστές. Αμέσως πάνω από κάθε διακόπτη υπάρχει μία κίτρινη και μία πράσινη LED. Πάντα ανάβει μία και μόνο μία από αυτές, υποδεικνύοντας την τρέχουσα τιμή του αντίστοιχου bit: όταν ανάβει η κίτρινη LED το bit είναι 0, και όταν ανάβει η πράσινη το bit είναι 1. Στη φωτογραφία, Q=M=N=C=1 και A=B=D=E=0· αντίστροφα, το σήμα π.χ. **OXI(C)**, συμβολισμένο σαν C-παύλα, είναι 0 στη φωτογραφία, ενώ το **OXI(D)** είναι 1. Οι τρεις πρώτοι διακόπτες, Q, M, και N, είναι διαφορετικοί από τους υπόλοιπους πέντε (A, B, C, D, και E): οι τρεις αριστεροί έχουν **μνήμη**, ενώ οι πέντε υπόλοιποι **δεν** έχουν. Όποτε πατάμε έναν από τους διακόπτες με μνήμη, το αντίστοιχο bit αλλάζει τιμή, και **παραμένει** στην νέα του τιμή (την θυμάται) μέχρι να ξαναπατηθεί ο διακόπτης. Οι υπόλοιποι διακόπτες, που δεν έχουν μνήμη, δίνουν στο bit τους την τιμή 1 όση ώρα πατιούνται, και επιστρέφουν στην τιμή 0 μόλις αφεθούν ελεύθεροι.

Δεξιά από τις κόκκινες LED υπάρχει ένα αριθμητικό "ψηφίο", σαν αυτά των γνωστών μας αριθμομηχανών τσέπης ή ψηφιακών ρολογιών. Αυτό λέγεται **ένδειξη 7 τμημάτων (7-**



segment display), διότι έχει 7 γραμμικές φωτοεκπομπούς διόδους και μία LED-κουκίδα (δεκαδική τελεία). Οι LED αυτές, που ονομάζονται a, b, c, d, e, f, g, και DP (decimal point), όπως δείχνει το μικρό σχηματάκι δεξιά τους, τροφοδοτούνται από τους ομώνυμους ακροδέκτες της καλωδιοταινίας που είδαμε νωρίτερα. Πειραματιστείτε συνδέοντας διάφορα καλώδια από διακόπτες προς LED's· για παράδειγμα, συνδέστε τον ακροδέκτη C, που είχαμε και πριν, αυτή τη φορά στην LED "d", και στη συνέχεια από την LED "d" επίσης στην LED "e" και από 'κει επίσης στην LED "g". Τώρα, όταν πατάμε τον διακόπτη C ανάβουν οι τρεις μικρές γραμμικές LED "d-e-g", εμφανίζοντας το σχήμα της φωτογραφίας δεξιά. Φυσικά, **απαγορεύεται να βραχυκυκλώνετε** δύο διακόπτες μεταξύ τους, είτε άμεσα, είτε έμμεσα --συνδέοντάς τους δηλαδή στην ίδια LED ή σε πολλαπλές LED που όμως είναι μεταξύ τους συνδεδεμένες: αν κάνετε ένα τέτοιο βραχυκύκλωμα και οι δύο διακόπτες είναι σε διαφορετική κατάσταση καθέννας, τότε το ένα κύκλωμα οδήγησης προσπαθεί να δώσει στο κοινό σημείο υψηλή τάση και το άλλο κύκλωμα οδήγησης προσπαθεί να του δώσει χαμηλή τάση, με αποτέλεσμα το βραχυκύκλωμα να διαρρέεται από μεγάλο ρεύμα, οι οδηγητές να ζεσταίνονται, και η τάση του κοινού σημείου να είναι απροσδιόριστη!

2.11 Ηλεκτρικά Ελεγχόμενοι Διακόπτες

Μέχρι τώρα, όλες οι εισοδοί των ψηφιακών κυκλωμάτων που φτιάξαμε ήταν διακόπτες που τους πατούσε ένα ανθρώπινο χέρι. Προφανώς, δεν μπορεί κανείς να συνεχίσει έτσι για πολύ, εξ αιτίας ενός πλήθους δυσκολιών: (i) έχουμε λίγα (σχετικά) δάκτυλα· (ii) είμαστε αργοί σε σχέση με τα ηλεκτρονικά κυκλώματα· (iii) πολλά κυκλώματα χρειάζονται την ίδια είσοδο σε πολλά μέρη, όπως π.χ. στους δύο διακόπτες B του αποκωδικοποιητή 2-σε-4 της §2.7, και υπάρχει ένα προφανές μηχανικό όριο στο πόσους διακόπτες μπορεί να κινήσει ένα δάκτυλο· και το σημαντικότερο: (iv) δεν μπορούμε να πάρουμε την έξοδο ενός κυκλώματος - υποσυστήματος και να την χρησιμοποιήσουμε σαν είσοδο σε ένα άλλο κύκλωμα - υποσύστημα, και κατά συνέπεια (v) δεν μπορούμε ούτε να πάρουμε μία πληροφορία αποθηκευμένη σε μία ηλεκτρονική μνήμη και να την χρησιμοποιήσουμε σαν είσοδο ενός κυκλώματος επεξεργασίας της.

Αυτό που χρειάζεται λοιπόν για να προχωρήσουμε είναι *ηλεκτρικά ελεγχόμενοι διακόπτες*, δηλαδή διακόπτες που να μπορεί να τους ανοιγοκλείνει ένα ηλεκτρικό σήμα, χωρίς την παρέμβαση ανθρώπινου χεριού· για την ακρίβεια, θέλουμε η έξοδος ενός διακόπτη να μπορεί να ανάψει ή να σβήσει έναν ή περισσότερους άλλους παρόμοιους διακόπτες. Στη σημερινή μικροηλεκτρονική τεχνολογία, αυτό επιτυγχάνεται χρησιμοποιώντας **transistors** σαν τέτοιους ηλεκτρικά ελεγχόμενους διακόπτες. Τα transistors είναι συνδυασμός ημιαγωγών με κατάλληλες προσμίξεις άλλων στοιχείων --συνήθως πυρίτιο (silicon - τετρασθενές) με προσμίξεις τρισθενών και πεντασθενών στοιχείων-- που του προσδίδουν ιδιότητες ενισχυτή και διακόπτη όταν χρησιμοποιηθούν υπό κατάλληλες τάσεις σε κατάλληλο κύκλωμα. Ο τρόπος κατασκευής και λειτουργίας του transistor είναι κυρίως θέμα Φυσικής της στερεάς κατάστασης, και ξεφεύγει από τα πλαίσια του παρόντος μαθήματος.

Οι υπολογιστές κατασκευάζονται με transistors από περίπου τη δεκαετία του '60 και πέρα. Οι πρώτοι υπολογιστές, τις δεκαετίες '40 και '50, ήταν κατασκευασμένοι με ηλεκτρονικές λυχνίες κενού (vacuum tubes), και ένας από αυτούς, ο Mark-II (στο Harvard την εποχή του πολέμου), με ηλεκτρομηχανικούς διακόπτες σαν αυτούς που θα περιγράψουμε παρακάτω. [Στις 9/9/1945, που ο Mark-II δεν δούλευε καλά, ψάχνοντας οι τεχνικοί βρήκαν την αιτία: ένα ζουζούνι (bug) --μία πεταλουδίτσα για την ακρίβεια-- είχε μπει στο διακόπτη F.70, και δεν τον άφηνε να κάνει καλά επαφή· έβγαλαν το ζουζούνι, και μαθεύτηκε σύντομα ότι είχαν κάνει "debugging"

(απεντόμωση) της μηχανής --από τότε, ο όρος αυτός χρησιμοποιείται ευρύτατα στους υπολογιστές... (για περισσότερες πληροφορίες, δείτε [εδώ](#))]. Εν τω μεταξύ, τα τηλεφωνικά κέντρα χρησιμοποιούσαν ηλεκτρομηχανική τεχνολογία ήδη από το πρώτο μισό του 20ου αιώνα· τέτοια "αναλογικά" τηλεφωνικά κέντρα επιβίωσαν μέχρι πριν λίγα χρόνια σε μερικές χώρες όπως και η δική μας, και τα αναγνώριζε κανείς από το χαρακτηριστικό ήχο "κρου-κρου-κρου-κρου" που έκαναν μέχρι να σε συνδέσουν.

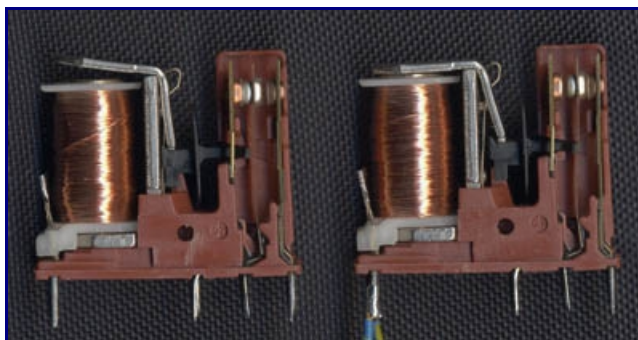
2.12 Ηλεκτρονόμοι (Relays)

Οι ηλεκτρομηχανικοί διακόπτες, γνωστοί σαν *ηλεκτρονόμοι* (*relays*, και στην αργκό των ηλεκτρολόγων: *ρελέδες*), εκτός από τα παλαιά τηλεφωνικά κέντρα και τον υπολογιστή Mark-II της δεκαετίας του '40, χρησιμοποιούνται ευρύτατα και σήμερα σε όλες τις ηλεκτρικές και βιομηχανικές εγκαταστάσεις, λόγω της αντοχής τους στις υψηλές τάσεις (εκατοντάδες ή χιλιάδες Volt), σε αντίθεση με τους ημιαγωγούς που καίγονται εύκολα από υπερτάσεις. Εμείς, εδώ, θα τους χρησιμοποιήσουμε σε μερικά εργαστήριά μας εξ αιτίας της εξαιρετικής εποπτείας που προσφέρουν: βλέπεις και ακούς το διακόπτη να ανοιγοκλείνει μπροστά σου, κι έτσι νιώθεις τη λειτουργία του ψηφιακού κυκλώματος πολύ καλύτερα απ' όσο με τα transistors, όπου δεν μπορεί κανείς να δει τι συμβαίνει με τα ηλεκτρόνια μέσα τους.

Η αρχή λειτουργίας του ηλεκτρονόμου είναι απλή: ένας *ηλεκτρομαγνήτης*, δηλαδή ένας σιδηρούς πυρήνας με ένα ηλεκτρικό πηνίο τυλιγμένο γύρω του, έλκει τον *οπλισμό* του, δηλαδή ένα κινητό σιδερένιο μοχλό, όταν περνάει ηλεκτρικό ρεύμα μέσα από το πηνίο. Μόλις διακοπεί το ηλεκτρικό ρεύμα, ο μαγνήτης παύει να έλκει τον οπλισμό του, και ένα ελατήριο επαναφέρει τον τελευταίο στην αρχική του θέση, μακριά από τον σιδερένιο πυρήνα. Συνδέουμε στον οπλισμό έναν ή περισσότερους διακόπτες, και καθώς ο οπλισμός κινείται υπό την επίδραση του ηλεκτρομαγνήτη αυτός κάνει τους διακόπτες να ανοιγοκλείνουν.

Ηλεκτρονόμοι υπάρχουν σε όλα τα μεγέθη, από μικροί, κατάλληλοι για να τους ελέγχει μιά χαμηλή τάση (5 - 12 Volt), π.χ. από ένα ηλεκτρονικό όργανο, και αυτοί με τη σειρά τους να αναβοσβήνουν μερικά φώτα (250 V, 5 A) ή ένα μικρό κινητήρα (π.χ. αντλία νερού), μέχρι πολύ μεγάλοι, κατάλληλοι για να ανοιγοκλείνουν τις μηχανές ολόκληρων εργοστασίων ή υποσταθμών της ΔΕΗ. Εμείς, φυσικά, θα χρησιμοποιήσουμε ένα τύπο ηλεκτρονόμου από τους μικρότερους, με ποδαράκια (ηλεκτρικές επαφές) κατάλληλα για την πλακέτα συνδέσεων (σε αντίθεση με άλλους που είναι κατάλληλοι για να βιδώνονται σε ηλεκτρικούς πίνακες).

Στην φωτογραφία δεξιά φαίνεται ένας ηλεκτρονόμος παρόμοιος με αυτούς του εργαστηρίου μας, στις δύο καταστάσεις του: αδρανής (αριστερά), και ενεργός (δεξιά). Το ηλεκτρικό πηνίο διέγερσης (εκατοντάδες σπείρες λεπτού, ξανθού, μονωμένου σύρματος, τυλιγμένες γύρω από τον πυρήνα) βρίσκεται στο αριστερό μέρος του ηλεκτρονόμου. Οι εξωτερικοί ακροδέκτες του πηνίου είναι δύο ποδαράκια στην αριστερή άκρη του ηλεκτρονόμου· στη φωτογραφία φαίνεται μόνο το μπροστινό --το άλλο είναι ακριβώς πίσω του. Ο πυρήνας του ηλεκτρομαγνήτη περνάει μέσα από το πηνίο, βγαίνει από κάτω, και ανεβαίνει προς τα επάνω ακριβώς δίπλα και δεξιά από το πηνίο (γκρί γυαλιστερό σίδηρο). Το μαγνητικό κύκλωμα συνεχίζει με το επάνω ήμισυ του οπλισμού, που βρίσκεται πάνω από τον



ηλεκτρομαγνήτη. Ο οπλισμός είναι το σίδηρο σε σχήμα "Γ" με ελαφρώς αμβλεία γωνία που βρίσκεται πάνω και δεξιά από το πηνίο. Στην αριστερή φωτογραφία, δεν περνάει ρεύμα από το πηνίο και ο οπλισμός βρίσκεται ψηλά, όπου τον συγκρατεί το κατακόρυφο ελατήριο που ίσα-ίσα φαίνεται. Στη δεξιά φωτογραφία, περνάει ρεύμα από το πηνίο και ο οπλισμός έχει χαμηλώσει και έχει κολλήσει στον πυρήνα, ελκόμενος από τον ενεργό ηλεκτρομαγνήτη. Με την κίνηση αυτή, το κάτω-δεξί άκρο του οπλισμού έχει κινηθεί δεξιά, και έχει παρασύρει ένα μαύρο, οριζόντιο, πλαστικό έμβολο, το οποίο με τη σειρά του έχει παρασύρει τη μεσαία επαφή του διακόπτη που φαίνεται στη φωτογραφία.

Στη δεξιά πλευρά του ηλεκτρονόμου υπάρχουν δύο διακόπτες, ένας εμπρός κι ένας πίσω· στη φωτογραφία φαίνεται μόνο ο μπροστινός. Ο διακόπτης που φαίνεται στη φωτογραφία έχει έναν πόλο --το μεσαίο, χάλκινο, κατακόρυφο, λεπτότερο έλασμα-- και δύο επαφές --τα δύο παχύτερα κατακόρυφα χάλκινα ελάσματα, ένα αριστερά και ένα δεξιά. Άρα, πρόκειται για έναν διακόπτη SPDT, σαν αυτούς της §0.8. Οι εξωτερικοί ακροδέκτες του διακόπτη είναι τα τρία ποδαράκια που φαίνονται κάτω από αυτόν, με την ίδια σειρά με την οποία βρίσκονται και τα αντίστοιχα ελάσματα μέσα στο διακόπτη. Από την πίσω πλευρά, που δεν φαίνεται στη φωτογραφία, υπάρχει άλλος ένας διακόπτης SPDT, ηλεκτρικά μονωμένος από τον πρώτο, που όμως κινείται από το ίδιο οριζόντιο μαύρο έμβολο, άρα ανοιγοκλείνει ταυτόχρονα με τον πρώτο. Οι 3 ακροδέκτες του πίσω διακόπτη είναι ακριβώς πίσω από τους ακροδέκτες του μπροστινού, και κρύβονται από αυτούς στη φωτογραφία.

Παρατηρήστε ότι ο διακόπτης είναι ηλεκτρικά πλήρως μονωμένος από τον ηλεκτρομαγνήτη και το πηνίο του: επικοινωνούν μόνο μηχανικά, μέσω του οριζόντιου μαύρου πλαστικού εμβόλου. Έτσι επιτυγχάνουμε τα δύο ηλεκτρικά κυκλώματα, το ελέγχον (πηνίο) και το ελεγχόμενο (διακόπτης), να είναι εντελώς ανεξάρτητα, με δυνατότητα να υπάρχει μεγάλη (και μεταβαλλόμενη) διαφορά δυναμικού μεταξύ των δύο· π.χ. το ένα μπορεί να ανήκει σε ένα ευαίσθητο ηλεκτρονικό όργανο, ενώ το άλλο μπορεί να συνδέεται σε ένα θορυβώδη ηλεκτρικό κινητήρα, τροφοδοτούμενο από άλλη φάση τριφασικής παροχής. Αυτό είναι ένα πλεονέκτημα των ηλεκτρονόμων που δεν το έχουν οι ηλεκτρονικοί διακόπτες (transistors).

Από την άλλη μεριά, φυσικά, για να αλλάξει κατάσταση ο ηλεκτρονόμος πρέπει να κινηθούν μηχανικά τμήματα, άρα απαιτείται χρόνος πολλών χιλιοστών του δευτερολέπτου (millisecond - ms), τη στιγμή που οι ηλεκτρονικοί διακόπτες (transistors) αναβοσβήνουν σε χρόνο σημαντικά κάτω του δισεκατομμυριοστού του δευτερολέπτου (nanosecond - ns), σήμερα, δηλαδή είναι δεκάδες εκατομμύρια φορές γρηγορότεροι. Επίσης, για να κολλήσει ο οπλισμός ενός ηλεκτρονόμου του εργαστηρίου μας απαιτείται ρεύμα της τάξης των 100 mA, άρα ισχύς γύρω στα 500 mW (στα 5 V που δουλεύουμε). Σε αντίθεση με αυτό, ένας σημερινός ηλεκτρονικός διακόπτης (transistor) μπορεί να ανοιγοκλείνει σε πλήρη ταχύτητα καταναλώνοντας κάτω από 50 μ W, και αναλογικά λιγότερο σε χαμηλότερες ταχύτητες, δηλαδή είναι δεκάδες χιλιάδες φορές οικονομικότερος στη μπαταρία (και στην παραγόμενη θερμότητα). Τέλος, ένα σημερινό transistor με τα γύρω κυκλώματά του καταλαμβάνει εμβαδόν γύρω στο ένα τετραγωνικό μ m (μικρόμετρο), συγκρινόμενο προς τα 300 τετραγωνικά mm (χιλιοστόμετρα) του ηλεκτρονόμου μας (χωρίς να παίρνουμε υπ' όψη και τη διαφορά ύψους), δηλαδή ο ηλεκτρονικός διακόπτης είναι γύρω στο ένα δισεκατομμύριο φορές μικρότερος!

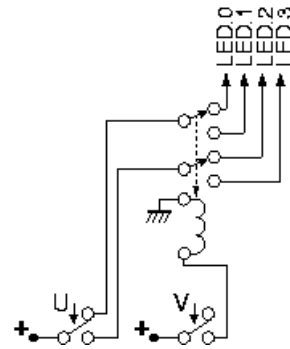
Σαν σύμβολο του ηλεκτρονόμου στα κυκλώματά μας θα χρησιμοποιούμε αυτό που φαίνεται δεξιά. Στο επάνω μέρος είναι οι δύο ηλεκτρικά ελεγχόμενοι διακόπτες. Στο κάτω μέρος υπάρχει ένα σύμβολο πηνίου, που συμβολίζει τον ηλεκτρομαγνήτη. Τα δύο μέρη ενώνονται με μία διακεκομμένη γραμμή, που συμβολίζει ότι το πηνίο ελέγχει το διακόπτη. Η



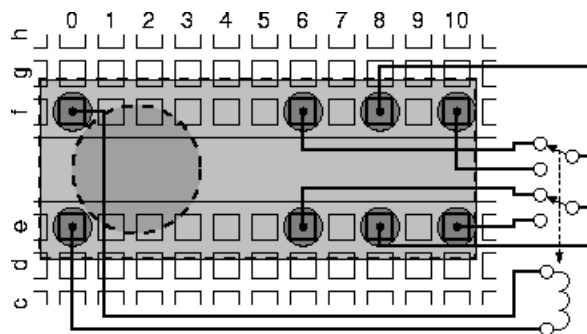
διακεκομμένη γραμμή έχει ένα βέλος προς το πηνίο, που συμβολίζει τη σύμβαση πολικότητάς μας: όταν το πηνίο είναι αδρανές (δεν διαρρέεται από ρεύμα), οι διακόπτες είναι στην επάνω κατάσταση τους· όταν το πηνίο ενεργοποιείται (διαρρέεται από ρεύμα), έλκει τους διακόπτες προς την φορά του βέλους, δηλαδή τους φέρνει στην κάτω κατάσταση τους. Παρατηρήστε ότι το πηνίο δεν έχει συγκεκριμένη πολικότητα, δηλαδή ενεργοποιείται με ρεύμα είτε της μίας φοράς είτε της άλλης. Επίσης παρατηρήστε ότι το πηνίο είναι ηλεκτρικά μονωμένο από τους διακόπτες, άρα δεν μπορεί να περάσει ρεύμα ανάμεσα στους ακροδέκτες του πηνίου και αυτούς των διακοπών, ούτε η διαφορά τάσης μεταξύ τους έχει καμία επίδραση στον ηλεκτρονόμο.

Πείραμα 2.13: Σύνδεση και Λειτουργία του Ηλεκτρονόμου

Θα δοκιμάσουμε πρώτα τη σύνδεση και τη λειτουργία του ηλεκτρονόμου χρησιμοποιώντας τον για να αντικαταστήσει τους δύο διακόπτες B που έπρεπε να κινούνται μαζί στον αποκωδικοποιητή 2-σε-4 του πειράματος 2.7. Στο νέο κύκλωμα, δεξιά, αντικαθιστούμε τους δύο εκείνους διακόπτες με τους διακόπτες του ηλεκτρονόμου, οι οποίοι ούτως ή άλλως κινούνται πάντα μαζί. Για να κινηθεί ο ηλεκτρονόμος σύμφωνα με τις επιθυμίες του χρήστη, προσθέτουμε το νέο διακόπτη "V", κάτω. Όταν πατιέται ο V, τροφοδοτεί με ρεύμα το πηνίο του ηλεκτρονόμου, κι αυτό έλκει τον οπλισμό του και κατά συνέπεια "πατάει" τους δύο διακόπτες του, επάνω. Επομένως, τελικά, οι δύο διακόπτες του ηλεκτρονόμου θα βρίσκονται πάντα στην ακριβώς ίδια κατάσταση με τον διακόπτη V, δηλαδή πετύχαμε να "αντιγράψουμε" την κατάσταση του V στους δύο νέους διακόπτες του ηλεκτρονόμου.

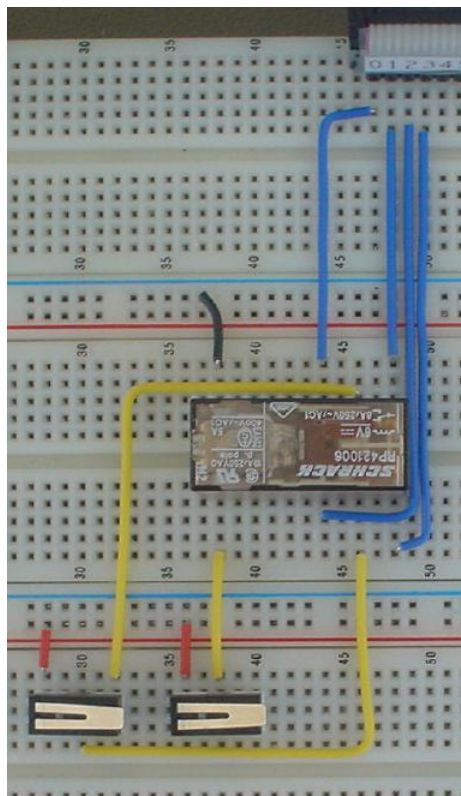


Στις πλακέτες συνδέσεων, οι ηλεκτρονόμοι του εργαστηρίου μας τοποθετούνται όπως φαίνεται στο σχεδιάγραμμα εδώ, στις σειρές e και f, δηλαδή τις πλησιέστερες στο κεντρικό "αυλάκι". Προτιμάμε να τους τοποθετούμε με το πηνίο αριστερά. Τότε, οι ακροδέκτες ενεργοποίησης του πηνίου βρίσκονται στις δύο **αριστερότερες υποδοχές που καλύπτει πλήρως** ο ηλεκτρονόμος. Οι ακροδέκτες των διακοπών βρίσκονται στη δεξιά άκρη του ηλεκτρονόμου, στήλη παρά στήλη. Στην επάνω σειρά (f) είναι ο ένας διακόπτης, και στην κάτω σειρά (e) είναι ο άλλος. Ο πόλος του κάθε διακόπτη είναι στο μεσαίο από τους 3 ακροδέκτες. Η επαφή αδράνειας είναι κοντά στο πηνίο, ενώ η επαφή ενεργοποίησης είναι στη δεξιά άκρη του ηλεκτρονόμου, στην **δεξιότερη υποδοχή που καλύπτει** ο ηλεκτρονόμος. Όταν ξέρουμε την (οριζόντια) συντεταγμένη των ακροδεκτών του πηνίου, οι συντεταγμένες των ακροδεκτών των διακοπών προκύπτουν προσθέτοντας τους αριθμούς 6 (επαφή αδράνειας), 8 (πόλος), και 10 (επαφή ενεργοποίησης).



Οι ακροδέκτες των διακοπών βρίσκονται στη δεξιά άκρη του ηλεκτρονόμου, στήλη παρά στήλη. Στην επάνω σειρά (f) είναι ο ένας διακόπτης, και στην κάτω σειρά (e) είναι ο άλλος. Ο πόλος του κάθε διακόπτη είναι στο μεσαίο από τους 3 ακροδέκτες. Η επαφή αδράνειας είναι κοντά στο πηνίο, ενώ η επαφή ενεργοποίησης είναι στη δεξιά άκρη του ηλεκτρονόμου, στην **δεξιότερη υποδοχή που καλύπτει** ο ηλεκτρονόμος. Όταν ξέρουμε την (οριζόντια) συντεταγμένη των ακροδεκτών του πηνίου, οι συντεταγμένες των ακροδεκτών των διακοπών προκύπτουν προσθέτοντας τους αριθμούς 6 (επαφή αδράνειας), 8 (πόλος), και 10 (επαφή ενεργοποίησης).

Κατασκευάστε στο εργαστήριο το παραπάνω κύκλωμα, και χρησιμοποιήστε το για να διαπιστώσετε πειραματικά τον τρόπο λειτουργίας του ηλεκτρονόμου. Αντί να συνδέουμε LED's μέσω αντιστάσεων, χρησιμοποιήστε τις LED 0, 1, 2, και 3 της πλακέτας εισόδων/εξόδων μέσω της καλωδιωταινίας, όπως στη φωτογραφία: εκμεταλλευόμαστε την ιδιότητα αυτών των LED's να παραμένουν σβηστές όσο δεν τους παρέχεται ρεύμα (ανοικτό κύκλωμα), και να ανάβουν όταν συνδεθούν σε πηγή υψηλής τάσης (π.χ. 5 V εδώ) οπότε και τους παρέχεται (θετικό) ρεύμα. Για τα πηνία των ηλεκτρονόμων, όμως, **δεν** μπορούμε να χρησιμοποιήσουμε τις εισόδους από τους διακόπτες της πλακέτας εισόδων/εξόδων, διότι οι εκεί ηλεκτρονικοί οδηγητές είναι φτιαγμένοι για να τροφοδοτούν μόνο chips, τα οποία χρειάζονται μικρό ρεύμα, και δεν μπορούν να δώσουν το μεγάλο ρεύμα που ζητούν οι ηλεκτρονόμοι για να κινηθούν. Στο εργαστήριο, παρατηρήστε προσεκτικά τη μηχανική κίνηση που κάνει ο οπλισμός του ηλεκτρομαγνήτη και οι δύο διακόπτες μέσα στον ηλεκτρονόμο όταν ο διακόπτης V τροφοδοτεί με ρεύμα το πηνίο. Δείτε ποιές φωτοдиодοι ανάβουν κάθε φορά και επιβεβαιώστε τη λειτουργία του κυκλώματος σαν αποκωδικοποιητή. Όταν τελειώσετε, **μην** χαλάσετε το κύκλωμά σας, διότι θα αποτελέσει τη βάση για το επόμενο πείραμα.



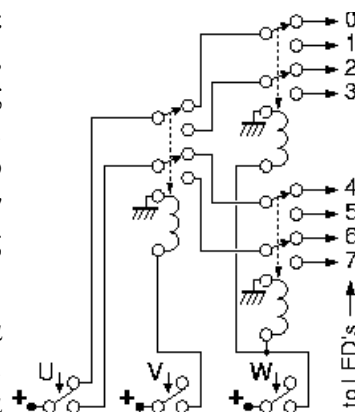
Πείραμα 2.14: Αποκωδικοποιητής 3-σε-8

Ο αποκωδικοποιητής του προηγούμενου πειράματος επεκτείνεται κατά προφανή τρόπο σε περισσότερες εξόδους, προσθέτοντας περισσότερα bits εισόδου και περισσότερα επίπεδα στο δέντρο του· καθώς το κάνουμε αυτό, πληθαίνουν και οι διακόπτες που πρέπει να ελέγχονται κατά όμοιο τρόπο από ορισμένα από τα bits εισόδου. Επεκτείνετε τον αποκωδικοποιητή σε τρία bits εισόδου, U, V, W, και 8 εξόδους, όπως στο σχήμα.

Στο τρίτο επίπεδο του δέντρου χρειαζόμαστε 4 διακόπτες SPDT, ελεγχόμενους όλους από το τρίτο bit εισόδου, το W. Επειδή δεν έχουμε τετραπλό διακόπτη ή ηλεκτρονόμο στα 5 Volt, χρησιμοποιήστε στη θέση του 2 ηλεκτρονόμους, τους οποίους θα ελέγχετε από το ίδιο σημείο, τον διακόπτη W. Επομένως, οι 4 αυτοί διακόπτες των ηλεκτρονόμων στο 3ο επίπεδο του δέντρου αποκωδικοποίησης θα αποτελούν πάντα "αντίτυπο" της εισόδου W, όπως και οι 2 διακόπτες στο 2ο επίπεδο αποτελούν αντίγραφο του bit εισόδου V.

Πριν φτάσετε στο εργαστήριο, φτιάξτε τον πίνακα αληθείας για καθεμιά από τις 8 εξόδους του κυκλώματος.

Επίσης, για την κάθε έξοδο παρατηρήστε το κύκλωμα που την τροφοδοτεί (τρεις διακόπτες εν σειρά) και εκφράστε την με μίαν "εξίσωση" (π.χ. (U)ΚΑΙ(OXI(V))ΚΑΙ(W)), βάσει των όσων



εκφράστε την με μίαν "εξίσωση" (π.χ. (U)ΚΑΙ(OXI(V))ΚΑΙ(W)), βάσει των όσων

είπαμε στην [§1.2](#). Στο εργαστήριο, φτιάξτε και ελέγξτε το κύκλωμα, παρατηρώντας ποιός συνδυασμός διακοπών ανάβει ποιά λάμπα, και δείξτε το στον βοηθό σας. Ένα μέρος του κυκλώματος το έχετε έτοιμο από το προηγούμενο πείραμα.