



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ  
ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

---

Ψηφιακή Σχεδίαση

**Εργαστήριο 8:**  
**Μετρητής μέσω Αθροιστή & Καταχωρητή,**  
**Ακμοπυροδότηση**

Μανόλης Γ.Η. Κατεβαίνης

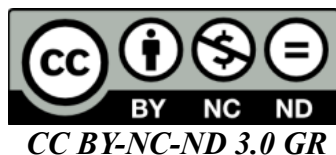
Τμήμα Επιστήμης Υπολογιστών

---

## 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



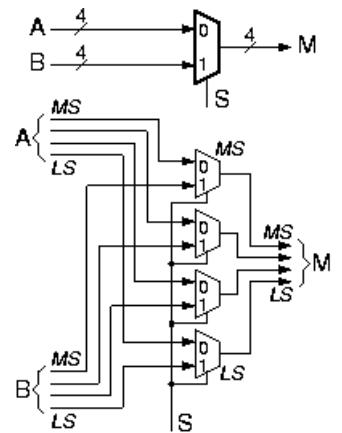
# Ψηφιακή Σχεδίαση

## Εργαστήριο 8: Μετρητής μέσω Αθροιστή & Καταχωρητή, Ακμοπυροδότηση

[Βιβλία: προαιρετικά μπορείτε να διαβάσετε: Wakerly: § 7.2.5 (σελ. 639-641) (αν θέλετε, ρίξτε μία ματιά και στις § 8.1.4 - 8.2.3 (σελ. 789-798))· Mano (4η έκδοση): μισή § 5.4 (σελ. 188-191)].

### 8.1 Πολύμπιτοι Πολυπλέκτες

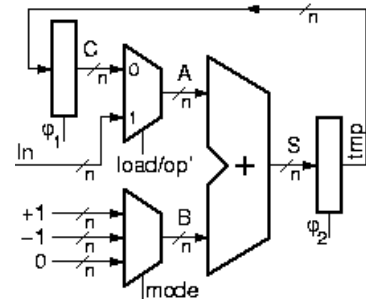
Στην §1.5 είδαμε ότι οι πολυπλέκτες είναι κυκλώματα με δύο ή περισσότερες εισόδους δεδομένων, μία έξοδο δεδομένων, και μία είσοδο επιλογής (είσοδο ελέγχου, με ένα ή περισσότερα bits)· ανάλογα με την τιμή της εισόδου ελέγχου, επιλέγεται μία από τις εισόδους δεδομένων και η λογική τιμή της αντιγράφεται στην έξοδο. Στην §4.9 είδαμε πώς κατασκευάζεται ένας πολυπλέκτης από πύλες, και πώς αυτός λειτουργεί εσωτερικά. Σε όλα αυτά τα παραδείγματα, οι εισόδοι δεδομένων ήταν του ενός bit καθεμία. Πολύ συχνά όμως, στις εφαρμογές, θέλουμε να επιλέξουμε μεταξύ εισόδων που έχουν πολλαπλά bits καθεμία· αυτό μπορούμε να το κάνουμε χρησιμοποιώντας τόσους πολυπλέκτες (του ενός bit) όσα και τα bits των λέξεων εισόδου, όπως φαίνεται στο σχήμα δίπλα.



Κανονικά, όλες οι εισόδοι δεδομένων πρέπει να έχουν το ίδιο πλήθος bits καθεμία (4 bits στο παράδειγμά μας). Το πλήθος των bits της εξόδου πρέπει να επαρκεί για να μεταφέρει την πληροφορία της οιασδήποτε επιλεγόμενης εισόδου, άρα πρέπει να ισούται με αυτό το πλήθος bits των εισόδων (εάν οι εισόδοι δεν είχαν όλες το ίδιο πλήθος bits, η έξοδος θα έπρεπε να έχει το μεγαλύτερο από τα πλήθη αυτά, για να "χωράει" την φαρδύτερη είσοδο, αλλά τότε θα έπρεπε να μας προδιαγράψουν και τι θα περιέχουν τα επιπλέον bits της εξόδου όποτε επιλέγεται είσοδος με λιγότερα bits). Ο ρόλος του **πολύμπιτου (multibit)** πολυπλέκτη, λοιπόν, είναι να τροφοδοτήσει το κάθε bit της εξόδου από το κατάλληλο bit της επιλεγόμενης εισόδου, όπως δείχνει το σχήμα. Για παράδειγμα, σαν MS bit της εξόδου πρέπει να μπορεί να επιλεγεί το MS bit μίας οιασδήποτε από τις εισόδους δεδομένων, μέσω του "MS" (περισσότερο σημαντικού) πολυπλέκτη· και ο "LS" πολυπλέκτης πρέπει να επιλέγει το LS bit της εξόδου μεταξύ των LS bits όλων των εισόδων, κ.ο.κ. Επομένως, χρειαζόμαστε από έναν (μονόμπιτο) πολυπλέκτη για το κάθε bit της εξόδου· κάθε τέτοιος (μονόμπιτος) πολυπλέκτης πρέπει να έχει τόσες εισόδους δεδομένων όσες και οι λέξεις δεδομένων εισόδου που θέλουμε να πολυπλέξουμε. Συνολικά, για να φτιάξουμε έναν (πολύμπιτο) πολυπλέκτη *m-σε-1* των *n* bits (*n-bit m-to-1 multiplexer*), χρειαζόμαστε *n* μονόμπιτους πολυπλέκτες μεγέθους *m-σε-1* ο καθένας. Προφανώς, σε δεδομένη στιγμή, οι πολυπλέκτες πρέπει να επιλέγουν καθένας το κατάλληλο bit, της *ίδιας* εισόδου· άρα, η είσοδος ελέγχου (επιλογής) πρέπει να έχει την ίδια, κοινή τιμή για όλους τους πολυπλέκτες. Για τον πολύμπιτο αυτό πολυπλέκτη χρησιμοποιούμε ένα συνοπτικό σύμβολο, όπως φαίνεται στο επάνω μέρος του σχήματος.

## 8.2 Μετρητής από Αθροιστές, Καταχωρητές, και Πολυπλέκτες

Το ακολουθιακό κύκλωμα που θα μελετήσουμε σε αυτό το εργαστήριο λέγεται "**μετρητής**" (counter), διότι μετράει, ξεκινώντας από δοσμένο αριθμό, δηλαδή θυμάται έναν αριθμό και τον αυξάνει (ή τον μειώνει) κατά ένα κάθε φορά που το επιθυμεί ο χρήστης. Ψηφιακοί μετρητές μπορούν να κατασκευαστούν με flip-flops διαφόρων τύπων (και το βιβλίο Wakerly έχει πολλά τέτοια παραδείγματα στην παράγραφο 8.4), αλλά εμείς θα ακολουθήσουμε έναν διαφορετικό τρόπο, χρησιμοποιώντας μανταλωτές τύπου D και αθροιστές. Ο τρόπος που θα ακολουθήσουμε εμείς φαίνεται στο σχήμα δίπλα, είναι απλούστερος, και είναι και κοντύτερα στον τρόπο λειτουργίας των αριθμητικών μονάδων στους σημερινούς υπολογιστές.



Όπως δείχνει το σχήμα, ένα ζευγάρι μανταλωτών, tmp και C (ελεγχόμενοι από διαφασικό ρολοί), είναι υπεύθυνοι για τη διατήρηση της τιμής του μετρητή. Κατά τη βασική λειτουργία, ξεκινώντας από την τιμή αυτή, C, ο αθροιστής υπολογίζει το άθροισμα  $S=C+1$ : το άθροισμα S αποθηκεύεται στην προσωρινή θέση tmp, και στη συνέχεια μεταφέρεται στον μανταλωτή C σαν η νέα τιμή του μετρητή. Ο μετρητής αποκτά ευελιξία με την προσθήκη δύο (πολύμπιτων) πολυπλεκτών στις δύο εισόδους του αθροιστή. Ο πρώτος πολυπλέκτης μας επιτρέπει να επιλέγουμε αν η βάση A της πρόσθεσης θα είναι η αποθηκευμένη τιμή του μετρητή, C, ή νέα, εξωτερική τιμή που έρχεται από την είσοδο In. Ο δεύτερος πολυπλέκτης μας επιτρέπει να επιλέγουμε τι θα προστεθεί στην βάση A: η τιμή  $B=+1$ , ή  $B=-1$ , ή  $B=0$ . Με τη συνδυασμένη χρήση των δύο πολυπλεκτών προκύπτουν οι τέσσερις χρήσιμες λειτουργίες του κυκλώματος, καθώς και δύο λιγότερο ενδιαφέρουσες πράξεις:

- **Count Up** (μέτρηση προς τα επάνω --αύξηση του C κατά 1):  $S = C+1$  ( $A=C$ ,  $B=+1$ ),
- **Count Down** (μέτρηση προς τα κάτω --ελάττωση του C κατά 1):  $S = C-1$  ( $A=C$ ,  $B=-1$ ),
- **Load** (φόρτωση αρχικής τιμής στον C από την εξωτερική είσοδο):  $S = In$  ( $A=In$ ,  $B=0$ ),
- **Noop** (καμία πράξη --διατήρηση αμετάβλητης της τιμής του C):  $S = C$  ( $A=C$ ,  $B=0$ ), και
- οι δύο λιγότερο ενδιαφέρουσες λειτουργίες:  $S = In+1$ , και  $S = In-1$ .

## 8.3 Το Πληκτρολόγιο και η Οθόνη LCD

Σε αυτό και τα επόμενα εργαστήρια, η επικοινωνία της πλακέτας συνδέσεων με την πλακέτα εισόδων/εξόδων θα γίνεται μέσω τριών καλωδιοταινιών, όπως φαίνεται δεξιά, διαφορετικών από τις δύο που είχαμε μέχρι τώρα (§2.10).





Η μεσαία από τις τρεις καλωδιωταινίες, της οποίας η απόληξη φαίνεται σε μεγέθυνση στη δεύτερη φωτογραφία, φέρνει στο κύκλωμα του χρήστη εισόδους από 6 από τους 8 γνωστούς μας διακόπτες --τους M, N, A, B, C, και D (θετική και αρνητική πολικότητα)-- και στέλνει εξόδους από το κύκλωμα του χρήστη στις 8 γνωστές μας κόκκινες λυχνίες LED, 0 έως και 7. Δεν παρέχεται σύνδεση στους διακόπτες Q και E και στον ενδείκτη 7 τμημάτων (7-segment display), και αυτοί δεν μπορούν πλέον να χρησιμοποιηθούν.



Οι δύο άλλες καλωδιωταινίες προσφέρουν νέες δυνατότητες εισόδου/εξόδου, τις οποίες θα εξηγήσουμε εδώ. Οι απολήξεις τους φαίνονται σε μεγέθυνση στις επόμενες φωτογραφίες. Η αριστερή καλωδιωταινία φέρνει στο κύκλωμα

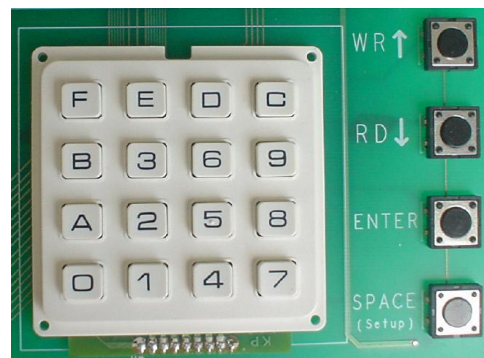
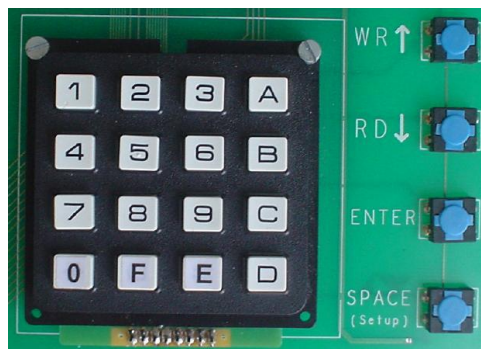


του χρήστη 16 bits εισόδου από το πληκτρολόγιο της πλακέτας εισόδων/εξόδων. Ο χρήστης μπορεί να την χρησιμοποιήσει είτε σαν μία δεκαεξάμπιτη (16-bit) είσοδο, ή σαν δύο οκτάμπιτες (8-bit) εισόδους, ή σαν τέσσερις τετράμπιτες (4-bit) εισόδους, ή σαν άλλα μίγματα λέξεων διαφόρων πλατών. Τα 16 bits αριθμούνται από το 0 (λιγότερο σημαντικό - LS) δεξιά μέχρι το 15 (περισσότερο σημαντικό - MS) αριστερά.

Η δεξιά καλωδιωταινία στέλνει 16 bits εξόδου από το κύκλωμα του χρήστη στην οθόνη υγρών κρυστάλλων (LCD - Liquid Crystal Display) της πλακέτας εισόδων/εξόδων. Και πάλι, η σύνδεση αυτή μπορεί να χρησιμοποιηθεί είτε σαν μία δεκαεξάμπιτη (16-bit) έξοδος, ή σαν δύο οκτάμπιτες (8-bit), ή τέσσερις τετράμπιτες (4-bit), κλπ. Η αρίθμηση των bits είναι ομοίως από το 0 (LS) δεξιά μέχρι το 15 (MS) αριστερά.



Στην δεξιά πλευρά της



πλακέτας εισόδων/εξόδων, υπάρχει ένα πληκτρολόγιο όπως ένα από τα δύο στις φωτογραφίες δεξιά (τα άσπρα πληκτρολόγια χρησιμοποιήθηκαν σε μερικές πλακέτες λόγω έλλειψης μαύρων στην αγορά, και έχουν δυστυχώς τα πλήκτρα σε ανορθόδοξες θέσεις). Επάνω από το πληκτρολόγιο υπάρχει μία οθόνη LCD με δύο γραμμές κειμένου: η πρώτη γραμμή γράφει "FROM USER CKT: ..." ("ckt" είναι συντομογραφία για "circuit" - κύκλωμα), και προορίζεται να μας δείχνει τιμές που έρχονται από το κύκλωμα του χρήστη, και η δεύτερη γραμμή γράφει "TO USER CKT: ...", και προορίζεται να στέλνει τιμές στο κύκλωμα του χρήστη.

Όταν η πλακέτα εισόδων/εξόδων αρχικοποιείται (δηλ. μετά το άναμα της τροφοδοσίας), η οθόνη LCD εμφανίζεται όπως δεξιά, με μόνη την επάνω γραμμή. Ο τετραψήφιος δεκαεξαδικός αριθμός που εμφανίζεται ισούται με τις τιμές που το κύκλωμα του χρήστη στέλνει στην οθόνη αυτή. Στη φωτογραφία αυτή, το κύκλωμα χρήστη δίνει "1" στα bits 11, 9, και 6, και "0" στα υπόλοιπα, δηλαδή δίνει τον δεκαεξάμπιτο αριθμό "0000.1010.0100.0000", ο οποίος εμφανίζεται σαν "0A40" στο δεκαεξαδικό. Όταν η οθόνη είναι σε αυτή την κατάσταση, με τον cursor στην επάνω γραμμή, τότε η ένδειξη "from user circuit" αντανακλά συνεχώς την παρούσα τιμή των εξόδων του χρήστη, δηλαδή αλλάζει μόνη της αμέσως μόλις αλλάξουν ένα ή περισσότερα bits που έρχονται από το κύκλωμα του χρήστη.



Στη συνέχεια, όταν πατηθεί το πλήκτρο ENTER, η οθόνη μπαίνει σε μία νέα κατάσταση, όπως δεξιά, όπου **παύει να παρακολουθεί το κύκλωμα του χρήστη** και δέχεται νέες τιμές από το πληκτρολόγιο για να τις στείλει στο κύκλωμα του χρήστη. Η κατάσταση αυτή φαίνεται από το ότι ο cursor βρίσκεται στην κάτω γραμμή.



Με τον cursor στην κάτω γραμμή, πληκτρολογούμε έναν τετραψήφιο δεκαεξαδικό αριθμό, π.χ. τον αριθμό "FF01" στη φωτογραφία δεξιά. Όσο πληκτρολογούμε και όσο ο cursor βρίσκεται στην κάτω γραμμή, οι παλαιές τιμές που δίδονταν από πριν στο κύκλωμα του χρήστη **δεν αλλάζουν ακόμα**.



Μόλις πατήσουμε ENTER, ο αριθμός που πληκτρολογήσαμε, εδώ "FF01", αποστέλλεται στο κύκλωμα του χρήστη, αλλάζοντας την παλαιά τιμή, εδώ σε "1111.1111.0000.0001". Η τιμή αυτή θα παραμείνει σταθερή μέχρι το επόμενο ανάλογο πάτημα του ENTER. Επίσης, τώρα που ο cursor είναι στην επάνω γραμμή, η επάνω γραμμή **παρακολουθεί ξανά, συνεχώς** τις αλλαγές τιμών από το κύκλωμα του χρήστη: εδώ, οι τιμές αυτές άλλαξαν σε "0000.1110.0000.0000", και γι' αυτό η οθόνη δείχνει ότι λαμβάνει "0E00" από το κύκλωμα χρήστη.



Εάν πατηθεί ξανά το ENTER, όπως δεξιά, η οθόνη παύει να παρακολουθεί το κύκλωμα χρήστη, και δέχεται είσοδο από το πληκτρολόγιο. Στη φωτογραφία δεξιά, μετά το ENTER, πατήθηκαν τα πλήκτρα 7 και 8.



Στην επόμενη και τελευταία φωτογραφία βλέπουμε την οθόνη όταν, μετά τα πλήκτρα 7 και 8 που είχαν πατηθεί από πριν, πατήθηκαν ακόμα τα 9, A, και ENTER: μόλις πατήθηκε το ENTER, η νέα είσοδος "789A"



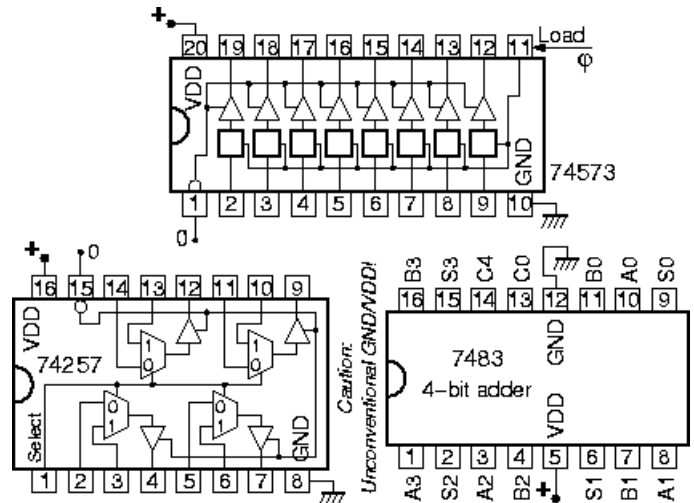
στάλθηκε στο κύκλωμα χρήστη, αλλάζοντας την προηγούμενη "FF01", και η οθόνη άρχισε πάλι να παρακολουθεί τις εξόδους από το κύκλωμα χρήστη, οι οποίες εδώ

φαίνονται να άλλαξαν σε "8600".

### Πείραμα 8.4: ο Μετρητής Up/Down/Load/Noop, με Διφασικό Ρολοί

Κατασκευάστε και δοκιμάστε τον παραπάνω μετρητή της §8.2 στο εργαστήριο, για  $n=4$  (τετράμπιτος μετρητής). Φτιάξτε καθέναν από τους δύο καταχωρητές με ένα chip 74573 (βλ. §3.7), σαν αυτό που χρησιμοποιήσαμε και στο πείραμα 7.8· παρ' ότι χρησιμοποιούμε μόνο τους μισούς μανταλωτές του κάθε τέτοιου chip, εν τούτοις χρειαζόμαστε δύο chips διότι οι δύο καταχωρητές έχουν διαφορετικό σήμα φόρτωσης.

Τα pins του chip επαναλαμβάνονται στο σχήμα εδώ· μην ξεχάστε να συνδέσετε το pin 0 στο λογικό 0 προκειμένου να παραμένουν ενεργοποιημένες οι τρικατάστατοι έξοδοι του chip (θα μάθουμε αργότερα τι είναι "τρικατάστατοι" έξοδοι). Τροφοδοτήστε τα δύο σήματα ελέγχου/χρονισμού  $\phi_1$  και  $\phi_2$  από τους διακόπτες A και B. Σαν τετράμπιτο αθροιστή χρησιμοποιήστε ένα chip 7483 (βλ. §3.7), σαν εκείνο του πειράματος 6.8· για διευκόλυνσή σας, τα pins του αθροιστή δίδονται ξανά στο σχήμα εδώ. **Προσοχή:** το chip αυτό έχει τα pins τροφοδοσίας σε ασυνήθιστες θέσεις!



Γιά την πολύπλεξη των τιμών C και  $I_n$  προς δημιουργία της εισόδου A του αθροιστή, χρησιμοποιήστε ένα chip 74257 (τετράμπιτος πολυπλέκτης 2-σε-1), του οποίου τα pins φαίνονται στο σχήμα δίπλα. Η είσοδος επιλογής είναι το pin 1: όταν είναι 0 επιλέγονται οι αριστερές εισόδου, και όταν είναι 1 επιλέγονται οι δεξιές εισόδου· οδηγήστε το σήμα αυτό από τον διακόπτη M. Οι έξοδοι και αυτού του chip είναι "τρικατάστατοι" (θα μάθουμε αργότερα τι είναι αυτό)· για να λειτουργούν σωστά για τους εδώ σκοπούς μας, συνδέστε το pin 15 (ενεργοποίηση εξόδων, αρνητικής πολικότητας) στο λογικό 0. Περισσότερες πληροφορίες για αυτά τα chips μπορείτε να βρείτε στην §3.7 ή στην ιστοσελίδα <http://focus.ti.com/lit/ds/symlink/sn74ls257b.pdf>.

Σαν εξωτερικές εισόδους  $I_n$  χρησιμοποιήστε 4 από τα bits της καλωδιοταινίας που έρχεται από το πληκτρολόγιο. Επίσης, για την τροφοδότηση της σταθεράς στην είσοδο B του αθροιστή, αντί πολυπλέκτη, χρησιμοποιήστε κατευθείαν 4 άλλα από τα bits της ίδιας καλωδιοταινίας. Φροντίστε οι θέσεις των bits να είναι τέτοιες ώστε η κάθε τετράδα να αντιστοιχεί ακριβώς σε ένα από τα ψηφία του τετραψηφίου δεκαεξαδικού αριθμού. Κάθε φορά που θα θέλετε να αλλάξετε ή την είσοδο  $I_n$  ή την είσοδο B, θα χρειάζεται να πληκτρολογήσετε και τις δύο, καθώς και 2 άλλα, αχρησιμοποίητα δεκαεξαδικά ψηφία.

Συνδέστε τους 4 τετράμπιτους αριθμούς C, A, S, και tmp στις 4 τετράδες bits της καλωδιοταινίας που πηγαίνει προς την οθόνη LCD· με τον τρόπο αυτό, κάθε ένα από τα 4 ψηφία της γραμμής "from user ckt" θα σας δείχνει την παρούσα κατάσταση καθενός από τους 4 αυτούς αριθμούς (όποτε ο cursor είναι στην επάνω γραμμή).

Κάντε ένα πλήρες σχεδιάγραμμα συνδεσμολογίας, όπως στην §4.10. Κατασκευάστε το κύκλωμα και ελέγξτε τη σωστή λειτουργία του. Αρχίστε επιλέγοντας την εξωτερική είσοδο  $I_n$  και τη σταθερά 0, και ενεργοποιώντας τη φάση  $\phi_2$  και μετά την  $\phi_1$ · παρακολουθήστε πώς αλλάζουν διαδοχικά τα σημεία A, S, tmp, και C.

Στη συνέχεια, αλλάξτε μέσω του διακόπτη M την επιλογή του πολυπλέκτη, ώστε



$A=C$ , και δώστε από το πληκτρολόγιο τη σταθερά +1 στο B. Ενεργοποιείτε εναλλάξ τις δύο φάσεις του ρολογιού, και παρακολουθείτε τους τετράμπιτους αριθμούς καθώς κινούνται γύρω από το κύκλωμα και αυξάνονται κατά 1 κάθε φορά. Συνεχίστε μέχρις ότου η τιμή του μετρητή συστραφεί (wrap around) και αρχίσει να ξαναπαίρνει τις ίδιες τιμές όπως στην αρχή.

Μετά, πληκτρολογήστε τη σταθερά -1 στο B, και ενεργοποιείτε ξανά εναλλάξ τις δύο φάσεις του ρολογιού· παρακολουθείτε τους τετράμπιτους αριθμούς να αλλάζουν με την αντίθετη σειρά και να συστρέφονται πάλι μέχρι την αρχική τιμή και πέραν αυτής.

Ενεργοποιήστε ταυτόχρονα τις 2 φάσεις ρολογιού (παράνομη κατάσταση) και παρατηρήστε την οθόνη LCD. Όταν σβήσουν οι 2 φάσεις, ποιός αριθμός μένει στο μετρητή; Υπάρχει τρόπος να προβλεφθεί αυτός; Επαναλάβετε πολλές φορές για να δείτε αν είναι προβλέψιμος ο αριθμός που προκύπτει, και προσπαθήστε να εξηγήσετε τις παρατηρήσεις σας.

Τέλος, ξαναφορτώστε έναν προκαθορισμένο αριθμό μέσω της εισόδου In (διαφορετικόν από τον πρώτο, και διαφορετικόν από τον αριθμό που είχε μείνει στο μετρητή), και ξαναμετρήστε λίγο πάνω και λίγο κάτω.

Ερώτηση κρίσεως: Αν ερμηνεύσουμε την τετράμπιτη τιμή του μετρητή σαν **προσημασμένο αριθμό συμπληρώματος-2**, είναι η μέτρηση προς τα πάνω πάντα σωστή; Όταν "τελειώνουν" τα bits και ο μετρητής συστρέφεται, από ποιόν αριθμό "πηδάει" σε ποιόν; Η μέτρησή προς τα κάτω είναι πάντα σωστή; Κατά τη συστροφή, από ποιόν αριθμό "πηδάει" σε ποιόν; Αν τώρα ερμηνεύσουμε την τετράμπιτη τιμή του μετρητή σαν **μη προσημασμένο αριθμό**, τότε είναι η μέτρηση προς τα πάνω πάντα σωστή; Σε ποιο σημείο συμβαίνει το "πήδημα" των αριθμών κατά τη συστροφή; Η μέτρησή προς τα κάτω είναι πάντα σωστή; Κατά τη συστροφή, από ποιόν αριθμό "πηδάει" σε ποιόν; Πώς δικαιολογείται η ορθότητα της λειτουργίας του μετρητή τόσο για προσημασμένους αριθμούς όσο και για μη προσημασμένους; Θυμηθείτε τη συνεστραμμένη αναπαράσταση των αριθμών που είδαμε στην §6.2.

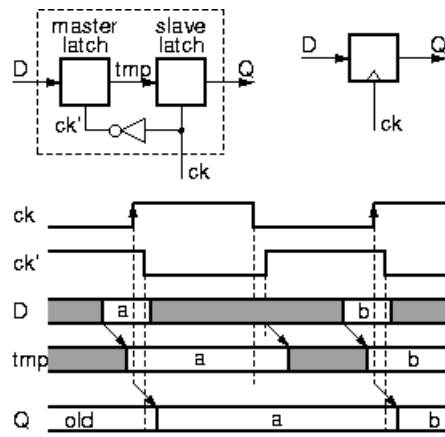
## 8.5 Καταχωρητές Αφέντη-Σκλάβου, Απλά (μονοφασικά) Ρολόγια, Ακμοπυροδότηση

Συνήθως, στα ακολουθιακά συστήματα, οι τιμές που εγγράφονται στα στοιχεία μνήμης είναι συνδυαστικές συναρτήσεις των τιμών που προϋπήρχαν αποθηκευμένες στα ίδια αυτά στοιχεία μνήμης· για παράδειγμα, στο μετρητή που είδαμε, η νέα τιμή του,  $C(\text{new})$ , είναι συνάρτηση της προηγούμενης τιμής του,  $C(\text{prev})$ :  $C(\text{new}) = C(\text{prev})+1$  ή  $C(\text{prev})-1$ . Όπως είδαμε, για να γίνει με ασφάλεια μιά τέτοια εγγραφή, δεν αρκεί ένας μανταλωτής ανά bit αποθηκευόμενης πληροφορίας, διότι όταν ανάβει το σήμα εγγραφής καταστρέφεται η αποθηκευμένη τιμή, με αποτέλεσμα να καταστρέφεται και η νέα τιμή --που είναι συνάρτησή της-- προτού προλάβει να αποθηκευτεί ασφαλώς.

Η λύση, όπως είδαμε, είναι η χρήση **δύο** μανταλωτών ανά bit αποθηκευόμενης πληροφορίας: πρώτα εγγράφεται η νέα τιμή στο ένα από αυτά,  $\text{tmp} = f(C)$ , και μετά αντιγράφεται αυτή στο άλλο,  $C = \text{tmp}$ , καταστρέφοντας την παλαιά τιμή, ούτως ώστε να είναι έτοιμη για την επόμενη πράξη. Αφού λοιπόν οι μανταλωτές χρησιμοποιούνται τόσο συχνά κατά ζευγάρια, τα ζευγάρια αυτά θα τα θεωρούμε συνήθως σαν ένα νέο δομικό λίθο. Κανονικά, οι μεμονωμένοι μανταλωτές χρειάζονται διφασικά ρολόγια για να δουλέουν, με την ιδιότητα ότι ποτέ δεν είναι αναμένες ταυτόχρονα και οι δύο φάσεις του ρολογιού. Όταν όμως φτιάχνουμε ζευγάρια μανταλωτών σαν ενιαίο κύκλωμα, απλοποιούμε αυτή την απαίτηση, ζητώντας ένα μοναδικό ρολόϊ (μοναδική φάση) από τον εξωτερικό χρήστη, και

φτιάχνοντας εσωτερικά και προσεγγιστικά τη δεύτερη φάση.

Το σχήμα δεξιά δείχνει τον απλούστερο τρόπο κατασκευής ενός τέτοιου ζευγαριού μανταλωτών. Τις δύο φάσεις του ρολογιού τις προσεγγίζουμε με το εξωτερικό σήμα ρολογιού,  $ck$  (**clock**), και το συμπλήρωμά του,  $ck'$ . Η προσέγγιση αυτή είναι ατελής σε ένα σημείο: λόγω της μη μηδενικής καθυστέρησης του αντιστροφέα, όταν το σήμα  $ck$  ανάβει, και μέχρι να προλάβει να σβήσει το  $ck'$ , βρίσκονται για λίγο αναμένα τα σήματα φόρτωσης και των δύο μανταλωτών, όπως φαίνεται στο διάγραμμα χρονισμού στο κάτω μέρος του σχήματος. Η



ατέλεια αυτή αντιμετωπίζεται από τον κατασκευαστή με το να φροντίσει η καθυστέρηση αυτού του αντιστροφέα να είναι πάντα σημαντικά μικρότερη από την καθυστέρηση του δεύτερου μανταλωτή, όπως θα αναλυθεί παρακάτω.

Επειδή ο δεύτερος μανταλωτής ακολουθεί πάντα πιστά τον πρώτο, παίρνοντας την τιμή του κάθε φορά που ανάβει το ρολόι, γι' αυτό ο πρώτος λέγεται "αφέντης" και ο δεύτερος "σκλάβος". Έτσι προέκυψε το όνομα αυτού του κυκλώματος: "**flip-flop αφέντη-σκλάβου**" (master-slave flip-flop). Όπως θα δούμε, η συμπεριφορά αυτού του flip-flop καθορίζεται από την **ακμή** του ρολογιού (εδώ, την θετική ακμή), δηλαδή από τη στιγμή που η τιμή του ρολογιού αλλάζει καθ' ορισμένη κατεύθυνση (εδώ, από 0 σε 1), και όχι από το χρονικό διάστημα που το ρολόι έχει ορισμένη τιμή, π.χ. τιμή 1.

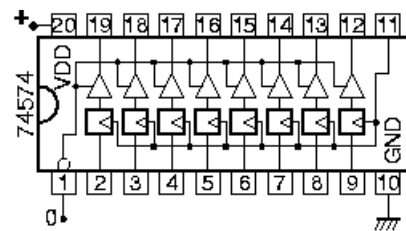
Υπάρχουν και άλλοι τρόποι να φτιάξει κανείς τέτοια flip-flops που η συμπεριφορά τους να καθορίζεται από μία ακμή του ρολογιού, τους οποίους εμείς δεν θα δούμε γιατί είναι πίο πολύπλοκοι. Τα άλλα εκείνα κυκλώματα τα ονομάζουμε "**ακμοπυροδότητα flip-flops**" (edge-triggered flip-flops), διότι ενεργοποιούνται (πυροδοτούνται) από την ακμή του ρολογιού. Δεδομένου ότι η εξωτερική συμπεριφορά των flip-flop αφέντη-σκλάβου και των ακμοπυροδότητων flip-flops είναι η ίδια, ο κόσμος συχνά δεν κάνει διάκριση μεταξύ τους, και το ίδιο θα κάνουμε κι εμείς --θα χρησιμοποιούμε τους δύο όρους σαν ισοδύναμους. Όταν γίνεται διάκριση μεταξύ "μανταλωτή" (latch) και "flip-flop", συνήθως η διάκριση αυτή είναι ότι το flip-flop είναι ακμοπυροδότητο (ή αφέντη-σκλάβου), ενώ ο μανταλωτής είναι όπως τον είδαμε μέχρι τώρα --ενεργοποιείται καθ' όλη τη χρονική διάρκεια που το σήμα φόρτωσης είναι αναμένο (1). Τέλος, από άποψη γραφικού συμβόλου, η διάκριση γίνεται μέσω ενός μικρού τριγώνου στην είσοδο ρολογιού που τοποθετείται στα flip-flops αφέντη-σκλάβου ή ακμοπυροδότητα, όπως φαίνεται στο σχήμα δεξιά, ενώ δεν τοποθετείται στους απλούς μανταλωτές.

Η λειτουργία του κυκλώματος φαίνεται με λεπτομέρεια στο διάγραμμα χρονισμού στο κάτω μέρος του σχήματος. Η είσοδος D επιτρέπεται να αλλάζει τιμή συνεχώς, εκτός από ένα μικρό χρονικό "παράθυρο εγκυρότητας" γύρω από την "ενεργή ακμή" του ρολογιού· εδώ, η ενεργή ακμή είναι η θετική (ανερχόμενη), ενώ σε άλλα κυκλώματα μπορεί να είναι η αρνητική (κατερχόμενη). Κατά την ημιπερίοδο πριν την ενεργή ακμή, όταν  $ck'=1$ , ο αφέντης μανταλωτής είναι ενεργοποιημένος, κι έτσι η τιμή της εισόδου D αντιγράφεται συνεχώς στον εσωτερικό κόμβο tmp· παρ' όλα αυτά, ο σκλάβος είναι αδρανής, κι έτσι η έξοδος Q παραμένει σταθερά ίση με την παλαιά της τιμή "old". Μόλις έλθει η ενεργή ακμή του ρολογιού, ο αφέντης κλείνει, κρατώντας μέσα του την τελευταία τιμή της εισόδου D που είδε, a. Ταυτόχρονα, ο σκλάβος ανοίγει, φέρνοντας την τιμή αυτή, a, στην έξοδο Q. Καθ' όλη τη διάρκεια της υπόλοιπης περιόδου, η έξοδος Q δεν μπορεί να ξαναλλάξει, αρχικά διότι ο κόμβος tmp είναι σταθερός, και αργότερα διότι ο σκλάβος μανταλωτής είναι σβηστός.

Το συνολικό αυτό κύκλωμα έχει την επιθυμητή ιδιότητα ότι η εισόδος του D επιτρέπεται να είναι συνδυαστική συνάρτηση της εξόδου του Q, διότι όταν η έξοδος αλλάζει το κύκλωμα είναι ήδη αναισθητο σε αλλαγές της εισόδου του, δηλαδή αυτές δεν επηρεάζουν την ήδη αποθηκευμένη τιμή του, για τον εξής λόγο. Η έξοδος αλλάζει τόση ώρα μετά την ενεργή ακμή του ρολογιού όση η καθυστέρηση του σκλάβου. Όμως, ο αφέντης καθίσταται αναισθητος σε αλλαγές της εισόδου D μόλις σβήσει το ck'. Επειδή ο αντιστροφείας κατασκευάζεται με καθυστέρηση πάντα μικρότερη από εκείνη του σκάλβου, το δεύτερο γεγονός (αναισθησία εισόδου) συμβαίνει πάντα πριν από το πρώτο (αλλαγή εξόδου).

Συμπερασματικά, η λειτουργία ενός ακμοπυροδότητου flip-flop (αφέντη-σκλάβου) έχει ως εξής. Η τιμή που υπήρχε στην εισοδό του λίγο πριν την ενεργή ακμή του ρολογιού αποθηκεύεται μέσα στο flip-flop κατά την ενεργή ακμή του ρολογιού, εμφανίζεται στην έξοδο του flip-flop λίγο μετά την ενεργή ακμή, και παραμένει εκεί σταθερή μέχρι λίγο μετά την επόμενη ενεργή ακμή του ρολογιού. Το flip-flop που είδαμε είναι flip-flop "θετικής ακμοπυροδότησης": η ενεργή ακμή του ρολογιού του είναι η θετική ακμή (μετάβαση από το 0 στο 1). Άλλα flip-flops είναι αρνητικής ακμοπυροδότησης": η ενεργή ακμή τους είναι η μετάβαση του ρολογιού από το 1 στο 0. Παρατηρώντας εξωτερικά ένα flip-flop, κατά την άλλη ακμή του ρολογιού (την μη ενεργή), τίποτα το ιδιαίτερο δεν συμβαίνει --με άλλα λόγια, μόνο η μία από τις δύο ακμές σε κάθε περίοδο ρολογιού είναι η ενεργή ακμή, κατά την οποία συμβαίνει η "εγγραφή" στο flip-flop.

### **Πείραμα 8.6: ο Μετρητής με Ακμοπυροδότηση**



Στον μετρητή του πειράματος 8.4, αντικαταστήστε τους δύο καταχωρητές τύπου μανταλωτή με έναν ακμοπυροδότητο καταχωρητή. Χρησιμοποιήστε ένα chip 74574, το οποίο, όπως φαίνεται δίπλα, έχει πανομοιότυπη θέση pins με τους μανταλωτές 74573, και διαφέρει μόνο κατά το ότι τα flip-flops του είναι ακμοπυροδότητα. Περισσότερες πληροφορίες για αυτά τα chips μπορείτε να βρείτε στην §3.7 ή στην ιστοσελίδα <http://focus.ti.com/lit/ds/symlink/sn74als574b.pdf>. Τροφοδοτήστε το pin 11 (ρολόι) από το διακόπτη A. **Πριν** το εργαστήριο κάντε ένα πλήρες σχεδιάγραμμα συνδεσμολογίας. **Στο εργαστήριο**, δοκιμάστε το κύκλωμα όπως και το προηγούμενο. Παρατηρήστε τη λειτουργία της ακμοπυροδότησης: μόλις πατάτε το διακόπτη (ενεργή ακμή ρολογιού), οι έξοδοι του καταχωρητή αλλάζουν, και αμέσως μετά (δεν φαίνεται η καθυστέρηση) αλλάζουν και οι έξοδοι του αθροιστή· όταν αφήνετε το διακόπτη (μη ενεργή ακμή ρολογιού) δεν γίνεται τίποτα.

### **Πείραμα 8.7: Καταχωρητής Ολίσθησης με Ακμοπυροδότηση**

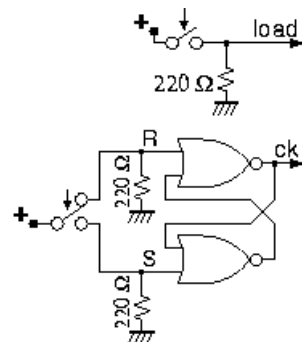
Υλοποιήστε την "ολίσθηση" των bits του πειράματος 7.8 μέσω ενός ακμοπυροδότητου καταχωρητή 74574, όπως στο προηγούμενο πείραμα, αντί μέσω των δύο καταχωρητών-μανταλωτών 74573 της §7.8. Τροφοδοτήστε το ρολοί από το διακόπτη A. Παρατηρήστε και τα 8 bits του καταχωρητή στις LED's. Αν έχετε χρόνο, παίξτε με το πού συνδέετε τον αντιστροφεία ανάδρασης και τι σχήματα αυτό προκαλεί στις LED's, σε συνδυασμό με τις (τυχαίες) αρχικές τιμές των bits του καταχωρητή.

## **8.8 Καθαρισμός Παλμών Διακόπτη (Switch Debouncing)**

[Η παράγραφος αυτή είναι εκτός ύλης μαθήματος, και προαιρετική στην ανάγνωση].

Όταν δεν έχουμε κυκλώματα όπως στην πλακέτα εισόδων/εξόδων που κάνουν "καθαρισμό παλμών" για μας, και προσπαθούμε να τροφοδοτήσουμε σήματα ρολογιού κατευθείαν από "γυμνούς" διακόπτες, πρέπει πρώτα να λύσουμε ένα πρακτικό θέμα που έχει να κάνει με τη δημιουργία "καθαρών" παλμών ρολογιού. Γενικά, γεννούμε παλμούς φόρτωσης μανταλωτών, load, με το κύκλωμα που φαίνεται στο επάνω μέρος του σχήματος. Το κύκλωμα αυτό δίνει 1 όποτε ο διακόπτης κάνει επαφή και 0 όποτε δεν κάνει. Το κύκλωμα αυτό έχει την εξής ιδιομορφία, η οποία στους μανταλωτές δεν μας πειράζει: κάθε μηχανικός διακόπτης, όταν πέφτει με φόρα πάνω στην επαφή του, αναπηδά μερικές φορές μέσα σε σύντομο χρονικό διάστημα (περίπου 10 ms) μέχρι να ισορροπήσει στην θέση επαφής. Η συνέπεια είναι ότι τη στιγμή που πατάμε το διακόπτη, από 0 που ήταν η έξοδος, αυτή γίνεται μερικές φορές εναλλάξ 1, 0, 1, 0, 1, 0, και στο τέλος ισορροπεί σταθερά στο 1. Στους μανταλωτές, αυτό δεν μας πειράζει, διότι αν σ' ένα κύκλωμα με διαφασικά ρολόγια ανάψει κάμποσες φορές η μία φάση **χωρίς** ενδιάμεσα να ανάψει η άλλη, το αποτέλεσμα είναι το ίδιο με το να είχε ανάψει η φάση μόνο μία φορά. Αν όμως το σήμα αυτό το χρησιμοποιήσουμε σαν ρολοί σ' έναν ακμοπυροδότητο καταχωρητή, αυτός θα δει τόσες ενεργές ακμές όσα και τα χοροπηδητά του διακόπτη --αριθμός σχετικά μικρός μεν, πλην όμως απρόβλεπτος.

Η λύση του προβλήματος είναι να χρησιμοποιηθεί ένας διακόπτης SPDT και ένας μανταλωτής τύπου RS, όπως φαίνεται στο κάτω μέρος του σχήματος. Η βασική ιδιότητα αυτού του κυκλώματος είναι η εξής: όταν το κινητό μέρος του διακόπτη βρίσκεται καθ' οδόν από τη μία επαφή προς την άλλη, τότε και οι δύο εισοδοί R και S του μανταλωτή είναι 0 (καμία από τις δύο δεν κάνει επαφή με την τροφοδοσία αριστερά), κι έτσι η έξοδος ck παραμένει *σταθερή* στην προηγούμενη τιμή της. Όταν ο διακόπτης δεν είναι πατημένος,  $R=1$  και  $S=0$ , άρα  $ck=0$ . Όταν ο διακόπτης πατηθεί, και ενώ βρίσκεται καθ' οδόν, πρώτα γίνεται  $R=0$ , οπότε η έξοδος παραμένει σταθερή εκεί που ήταν, στο 0. Μόλις ο διακόπτης ακουμπήσει για *πρώτη φορά* την κάτω επαφή, γίνεται  $S=1$  ενώ  $R=0$ , άρα η έξοδος αλλάζει τιμή σε  $ck=1$ . Όταν ο διακόπτης αναπηδήσει, γίνεται  $S=0$  ενώ  $R=0$ , οπότε η έξοδος παραμένει σταθερή *εκεί που ήταν*, δηλαδή στο 1. Παρατηρήστε ότι οι αναπηδήσεις δεν είναι τόσο μεγάλες ώστε ο διακόπτης να ξανακάνει επαφή επάνω --απλώς χοροπηδάει για λίγο κοντά στην κάτω επαφή. Τελικά, όταν ο διακόπτης ισορροπήσει, γίνεται  $S=1$  ενώ  $R=0$ , άρα η έξοδος παραμένει στο 1. Συνολικά, βλέπουμε ότι η έξοδος έκανε μόνο μία μετάβαση από το 0 στο 1, τη στιγμή της *πρώτης* επαφής του διακόπτη κάτω. Τα ανάλογα συμβαίνουν και όταν αφήνουμε το διακόπτη να επανέλθει επάνω.



Στο μετρητή του πειράματος 8.6, αν τροφοδοτήσουμε το ρολοί από έναν απλό διακόπτη χωρίς καθαρισμό παλμών, θα μπορούσαμε να μετρήσουμε το πλήθος των αναπηδήσεων του διακόπτη, υποθέτοντας ότι αυτό είναι μικρότερο του 16. Αν κάνετε το πείραμα, ελέγξτε αν αυτό το πλήθος είναι σταθερό.