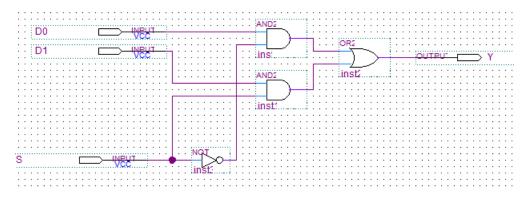
Εργαστηριακή Άσκηση 3

ΧΡΗΣΤΟΣ ΚΟΥΤΟΥΛΗΣ ΑΜ:5064 ΜΑΡΙΑ ΚΑΤΩΛΗ, ΑΜ:5083 ΟΜΑΔΑ 18

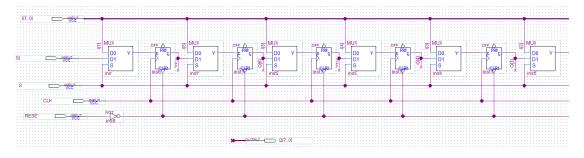
REGISTER

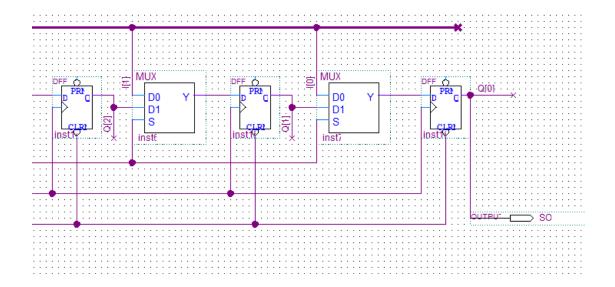
Φτιάξαμε έναν καταχωρητή 8bitou αριθμού, ο οποίος αποτελείται από 8 πολυπλέκτες δύο λειτουργιών και 8 DFFs.

Ο πολυπλέκτης:

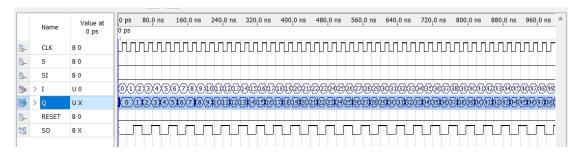


Ο καταχωρητής (σε δύο screenshots):



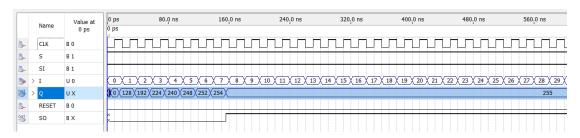


Η εξομοίωση για την παράλληλη φόρτωση:



Τα δεδομένα φορτώνονται και βγαίνουν ως έξοδος.

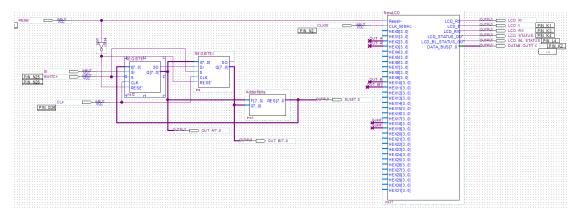
Η εξομοίωση για την δεξιά ολίσθηση:



Έχουμε έναν αριθμό με 8 δυαδικά ψηφία και αρχίζουμε με το 00000000 (=0), 10000000 (=128), 11000000 (=192), κ.ο.κ.

ΚΥΚΛΩΜΑ FIBONACCI

Το σχηματικό για το κύκλωμα που υπολογίζει την ακολουθία Fibonacci:



Το RESET αντιστρέφεται πριν μπει στους καταχωρητές A,B, ενώ στο LCD δεν αντιστρέφεται, γιατί το reset στο LCD ενεργοποιείται στο 0, ενώ στους καταχωρητές στο 1.

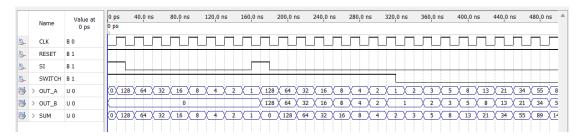
Οι καταχωρητές λειτουργούν με δεξιά ολίσθηση και παράλληλη φόρτωση ανάλογα με την είσοδο επιλογής μας SWITCH. Όταν SWITCH=0 τότε γίνεται η παράλληλη φόρτωση, ενώ όταν SWITCH=1 τότε γίνεται η δεξιά ολίσθηση.

Συνδέουμε τα αποτελέσματα των δύο καταχωρητών με τον αθροιστή ώστε να προσθέτει κάθε φορά τους δύο τελευταίους όρους της ακολουθίας Fibonacci και στη συνέχεια να αποθηκεύει το αποτέλεσμα στον καταχωρητή Α. Επιλέξαμε τον καταχωρητή Α για την αποθήκευση του αποτελέσματος της πρόσθεσης, γιατί αυτός -πριν γίνει η αποθήκευση του αποτελέσματος της πρόσθεσης- έχει μεταφέρει το δικό του αποτέλεσμα -που είναι ο ένας από τους δύο αριθμούς της πρόσθεσης- στον καταχωρητή Β. Αυτό επιτυγχάνεται λόγω της καθυστέρησης του αθροιστή να μεταφέρει το αποτέλεσμα στον Α. Ο Α βρίσκει χρόνο να μεταφέρει τα δεδομένα του πριν τα αντικαταστήσει.

Παρακάτω φαίνεται ο πίνακας που φτιάξαμε για το LCD display. Προσέχουμε ότι έχουμε δώσει τις θέσεις 2,3,10,11,18,19 για να προβάλλονται τα δυναμικά στοιχεία (που είναι τα αποτελέσματα του A, του B και του αθροιστή αντίστοιχα). Γι' αυτό έχουμε συνδέσει τα $OUT_A[7..4]$ με το HEX2[3..0], $OUT_A[3..0]$ με το HEX3[3..0], $OUT_B[7..4]$ με το HEX10[3..0], $OUT_B[3..0]$ με το HEX11[3..0], SUM[7..4] με το HEX18[3..0] και το SUM[3..0] με το HEX19[3..0].

Addr	+0	+1	+2	+3	+4	+5	+6	+7	ASCII
0	41	20	00	00	20	20	20	20	Α
8	42	20	00	00	20	20	20	20	В
16	46	20	00	00	20	20	20	20	F
24	20	20	20	20	20	20	20	20	

Η εξομοίσωση:



Αρχικά βάζουμε τον πρώτο άσσο στο σύστημα, περιμένουμε 8 ακμές του ρολογιού για να μεταφερθεί στο τέλος του καταχωρητή Α (μέσω δεξιάς ολίσθησης, SWITCH=1) και τότε προσθέτουμε και άλλον έναν άσσο, περιμένουμε άλλες 8 ακμές για να μεταφερθεί ο πρώτος στο τέλος του καταχωρητή Β και το τελευταίος στο τέλος του καταχωρητή Α. Έτσι οι καταχωρητές μας έχουν και οι δύο αποθηκευμένοι τον αριθμό 1. Τότε φορτώνουμε (SWITCH=0) τα δεδομένα στον αθροιστή και σε κάθε ακμή του ρολογιού γίνεται η πρόσθεση. Το αποτέλεσμα του αθροιστή γυρίζει στον καταχωρητή Α ένω το αποτέλεσμα του Α πηγαίνει στον καταχωρητή Β, όπως εξηγείται παραπάνω.