

ΨΗΦΙΑΚΑ ΣΥΣΤΗΜΑΤΑ HARDWARE ΣΕ ΧΑΜΗΛΑ ΕΠΙΠΕΔΑ ΛΟΓΙΚΗΣ 1

ΟΘΟΝΗ 7 ΤΜΗΜΑΤΩΝ – 4 ΨΗΦΙΩΝ

ΜΕΡΟΣ Α

ΦΙΛΙΠΠΟΣ ΓΕΡΜΑΝΟΠΟΥΛΟΣ ΑΕΜ:10005

ΣΙΤΑΡΙΔΗΣ ΠΑΝΑΓΙΩΤΗΣ ΑΕΜ:10249

ΣΥΝΟΨΗ ΤΟΥ ΜΕΡΟΥΣ Α ΤΗΣ ΕΡΓΑΣΙΑΣ

Το μέρος Α της εργασίας αφορά την υλοποίηση ενός κωδικοποιητή 4 bit σε 7, για την εμφάνιση ψηφίων σε οθόνη LED, τον σχεδιασμό του FSM για την οδήγηση των 4 ανόδων και την προβολή των αποτελεσμάτων σε προσομοίωση.

ΠΕΡΙΓΡΑΦΗ

Αρχικά έχουμε ένα ρολόι συχνότητας 50MHz, ένα σήμα reset και 4 σήματα των 4^{ων} bit.

Για να καθορίσουμε ποια από τα σήματα των ανόδων θα είναι ενεργά ανά πάσα στιγμή δημιουργήσαμε ένα module «Clock_Divider» στο οποίο με την χρήση ενός μετρητή ορίσαμε ως την διάρκεια του σήματος των ανόδων στα 0,32μs, όπως ζητείται. Ειδικότερα, αφού το ρολόι μας έχει περίοδο 20ns χρειάζεται ο μετρητής να μετρήσει 16 παλμούς του ρολογιού για να φτάσει τα 0,32μs. Στο ίδιο module επίσης γνωρίζοντας ότι με 4 σήματα ανόδων μπορούν να προκύψουν 16 καταστάσεις, ορίσαμε ένα σήμα “anode”, των 4bit το οποίο παίρνει τιμές ανάλογα με την κάθε κατάσταση και στέλνεται ως έξοδος. Στην ουσία υπάρχουν 4

σήματα που αντιπροσωπεύουν την ενεργοποίηση μίας ανόδου και σε κάθε ένα από αυτά αντιστοιχούν 4 καταστάσεις. Αναλυτικότερα η μία κατάσταση οδηγεί τα σήματα των ενδείξεων, δηλαδή των χαρακτήρων που θέλουμε να προβάλλουμε σε κάθε οθόνη και οι υπόλοιπες 3 μεσολαβούν ως περιθώριο ασφαλείας ούτως ώστε να μην υπάρχουν χρονικές επικαλύψεις στα σήματα και να μην αλλοιωθούν. Επιπλέον καθώς η χωρητικότητα των ενδείξεων είναι αρκετά μεγάλη θα πρέπει να χρησιμοποιηθούν τιμές του μετρητή ώστε να υπάρχει χρόνος προετοιμασίας μεταξύ των ανόδων και των δεδομένων των ενδείξεων. Αυτό το περιθώριο αλλαγής διαρκεί 2 περιόδους. Έτσι δημιουργήσαμε μια μεταβλητή «anode2» η οποία παίρνει τις ίδιες τιμές με την μεταβλητή anode, αλλά 2 κύκλους νωρίτερα.

Για την τοποθέτηση των ψηφίων σε χρονική σειρά δημιουργήθηκε ένα module, το «BCD_Control», το οποίο δέχεται ως είσοδο τα 4 ζητούμενα σύμβολα και το προαναφερθέν σήμα anode2 από το module Clock_Divider. Ως έξοδο δίνει το ψηφίο που έχει σειρά την δεδομένη στιγμή.

Εφόσον επιλέχθηκε το κατάλληλο σύμβολο και καθορίστηκε η κατάλληλη άνοδος για να εμφανιστεί αυτό στην αντίστοιχη οθόνη, μένει να αλλάξουν κατάλληλα τα σήματα των 7 ενδείξεων του κυκλώματος LED. Αυτό επιτυγχάνεται με το module «Decoder.sv», το οποίο δέχεται ως όρισμα το σήμα των 4bit που προορίζεται για προβολή και έχει ως έξοδο ένα σήμα

των 7 bit που σηματοδοτεί που σηματοδοτεί ποια τμήματα LED πρέπει να ανάψουν στην οθόνη για την ορθή εμφάνιση του συμβόλου.

Τέλος στο module testbench.sv δημιουργήθηκε το ρολόι, το σήμα reset και κλήθηκε το TOP με τα κατάλληλα ορίσματα.

ΣΧΟΛΙΑΣΜΟΣ ΛΥΣΗΣ ΚΑΙ ΠΡΟΒΛΗΜΑΤΩΝ

Κυριότερο πρόβλημα κατά την διάρκεια της εργασίας υπήρξε η ορθή οδήγηση των ανόδων μέσω του περιστροφικού μετρητή. Αυτό γιατί θεωρήθηκε πως η εξασφάλιση ενός περιθωρίου ασφαλείας από την επιστροφή ενός σήματος ανόδου στο λογικό 1, μέχρι την πτώση του επομένου ήταν αρκετή ώστε να μην υπάρξουν αλληλεπικαλύψεις και αλλοιώσεις των ενδείξεων. Ωστόσο έγινε αντιληπτή η σημασία του περιθωρίου αλλαγής και έτσι εντάχθηκε η μεταβλητή anode2 για την διαφοροποίηση αυτή και την επίλυση του προβλήματος.

Επίσης ένα ακόμα πρόβλημα που παρουσιάστηκε ήταν η σειρά των $4^{\omega\upsilon}$ συμβόλων που προοριζόταν προς απεικόνιση κάθε φορά. Παραδείγματος χάρη, στην προσπάθειά μας να απεικονίσουμε την λέξη $4^{\omega\upsilon}$ συμβόλων «-194» μετατρέπαμε με σειρά το κάθε

σύμβολο σε μία μεταβλητή των 4bit και τα αποστέλλουμε στο κύριο module TOP του design.sv. Ωστόσο, το module BCD_Control το οποίο ήταν υπεύθυνο για την εξαγωγή του κατάλληλου συμβόλου προς απεικόνιση επέλεγε τα σύμβολα με αντίθετη σειρά, με αποτέλεσμα να εμφανιζόταν στο τέλος η λέξη «491-». Αποφασίστηκε έτσι, να μεταβιβάσουμε τα ορίσματα στο module TOP στην testbench.sv με ανάποδη σειρά.

Δυσκολία αποτέλεσε ακόμη η διαίρεση της εργασίας σε υποπροβλήματα και επιμέρους FSMs, όπως και η απόδοση ρόλων σε αυτά. Λόγου χάρη, κατά τη διάρκεια της ανάλυσής μας, είχε δημιουργηθεί ένα επιπρόσθετο module που τελικά δυσκόλεψε τον συλλογισμό στην δημιουργία του 7 segmentdisplaydriver και τελικά αφαιρέθηκε και ένα μικρό κομμάτι του που κρίθηκε απαραίτητο συμπεριλήφθηκε στο module Clock_Divider. Συνειδητοποιήσαμε δηλαδή πόσο απαραίτητος κρίνεται ο αναλυτικός σχεδιασμός της λύσης και η κατανόηση του προβλήματος σε βάθος.