

# HY220: Εργαστήριο Ψηφιακών Κυκλωμάτων

Τμήμα Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης  
Χειμερινό Εξάμηνο 2013

## Εργαστήριο 3: Υλοποίηση Συναγερμού

4 Νοεμβρίου έως 8 Νοεμβρίου 2013 (4<sup>η</sup> εβδομάδα)

### 1.1 Σκοπός της Εργαστηριακής Άσκησης

Σε αυτήν την Άσκηση θα επιχειρήσουμε να δημιουργήσουμε το κύκλωμα ενός συστήματος συναγερμού για έλεγχο παραβίασης σε έναν εποπτευόμενο χώρο. Με χρήση πλήκτρων, διακοπών και LEDs θα πρέπει να επιδείξετε την λειτουργία του συστήματος αυτού.

### 1.2 Σχεδιασμός Συστήματος Συναγερμού

Πρώτο βήμα για την υλοποίηση της εργασίας είναι ο σχεδιασμός, στο χαρτί ή και σε ηλεκτρονική μορφή, το κύκλωμα που υλοποιεί την λογική ελέγχου, σύμφωνα με τους παρακάτω κανόνες. Το σύστημα κάνει χρήση 4 πλήκτρων (ENTER-CODE, ARM/DISARM, VIOLATION, RESET), που αντιστοιχούν σε διακριτές καταστάσεις λειτουργίας του:

1. **ENTER-CODE:** Αντιστοιχεί στην κατάσταση λειτουργίας όπου το σύστημα περιμένει την εισαγωγή κωδικού (8-bit ακέραιος). Στην κατάσταση αυτή, το σύστημα διαβάζει τον κωδικό από τους διαθέσιμους διακόπτες, και τον αποθηκεύει όταν πατηθεί το αντίστοιχο πλήκτρο ("ENTER CODE"). Το σύστημα παραμένει ανενεργό όσον αφορά την παρακολούθηση του χώρου για διάγνωση παραβίασης. Όσο το σύστημα δεν έχει αποθηκεύσει κωδικό, θα πρέπει να ανάβει ένα LED. Όταν αποθηκευθεί κωδικός, το LED πρέπει να σβήσει. Έχοντας αποθηκεύσει κωδικό, το σύστημα δεν εξετάζει ακόμα τον χώρο για έλεγχο παραβίασης, μέχρι να ενεργοποιηθεί πλήρως.
2. **ARM/DISARM:** Ενεργοποίηση/απενεργοποίηση του συναγερμού. Στην κατάσταση αυτή, το σύστημα διαβάζει κωδικό (8-bit ακέραιο) από τους διαθέσιμους διακόπτες, και συγκρίνει με τον αποθηκευμένο κωδικό, όταν πατηθεί το αντίστοιχο πλήκτρο ("ARM/DISARM"). Εάν ο κωδικός από την είσοδο είναι ο ίδιος με τον αποθηκευμένο, τότε το σύστημα «οπλίζει». Εάν το σύστημα είναι ήδη «οπλισμένο», τότε απενεργοποιείται. Όσο το σύστημα είναι «οπλισμένο» θα πρέπει να ανάβει ένα LED. Στην κατάσταση αυτή, το σύστημα εξετάζει τον χώρο για έλεγχο παραβίασης.
3. **VIOLATION:** Αντιστοιχεί την κατάσταση λειτουργίας όπου το σύστημα έχει διαγνώσει παραβίαση. Για να εξομοιωθεί παραβίαση, θα χρησιμοποιήσετε πλήκτρο. Όταν πατηθεί το πλήκτρο αυτό, και εφόσον το σύστημα είναι «οπλισμένο», θα πρέπει να ανάβει ένα LED. Στην κατάσταση αυτή, το σύστημα περιμένει να εισαχθεί κωδικός (από τους 8 διακόπτες). Εάν ο κωδικός από την είσοδο είναι ο ίδιος με τον αποθηκευμένο, τότε το σύστημα θεωρεί ότι δεν υφίσταται πλέον παραβίαση του χώρου, συνεπώς σβήνει το αντίστοιχο LED, αλλά παραμένει «οπλισμένο».

4. **RESET**: Με το πάτημα αυτού του πλήκτρου, το σύστημα επανέρχεται στην αρχική του κατάσταση, με άλλα λόγια «ξεχνά» τον τυχόν αποθηκευμένο κωδικό και δεν εξετάζει πλέον τον χώρο για έλεγχο παραβίασης. Μετά από το πάτημα αυτού του πλήκτρου, η μόνη αποδεκτή ενέργεια είναι η εισαγωγή και αποθήκευση κωδικού.

Η λογική του ζητούμενου συστήματος συναγερωμού απαιτεί την σχεδίαση και υλοποίηση μηχανής πεπερασμένων καταστάσεων (Finite State Machine - FSM).

### 1.3 Συγγραφή του Κώδικα Verilog και Προσομοίωση

Αφού ετοιμάσατε το σχέδιο του κυκλώματος, θα πρέπει να το μετατρέψετε σε κώδικα Verilog. Αμέσως μετά πρέπει να δημιουργήσετε ένα testbench module με το οποίο θα δοκιμάσετε να προσομοιώσετε το σύστημα.

### 1.4 Δημιουργία UCF και Τοποθέτηση

Αφού ολοκληρώσετε τα προηγούμενα βήματα, πρέπει να δημιουργήσετε το Αρχείο Περιορισμών Χρήστη (User Constraints File – UCF) δηλώνοντας τις εισόδους και τις εξόδους που θα χρησιμοποιήσετε στην πλακέτα και την τάση που θα τους ασκήσετε, όπως δείχνει το ucf αρχείο που λάβατε στο εργαστήριο 0 το οποίο είναι κομμάτι του master ucf που έχει δοθεί από τους κατασκευαστές και υπάρχει στο Documentation της πλακέτας στη σελίδα: <http://zedboard.org/documentation/1521>

### 1.5 Παράδοση

Θα παραδώσετε στους βοηθούς του μαθήματος αναφορά που θα περιλαμβάνει το σχέδιο του κυκλώματος, και τον κώδικα για την προσομοίωση.