



Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών
Υπολογιστών
Έτος: 2024-2025

Εργαστήριο Ψηφιακών Συστημάτων

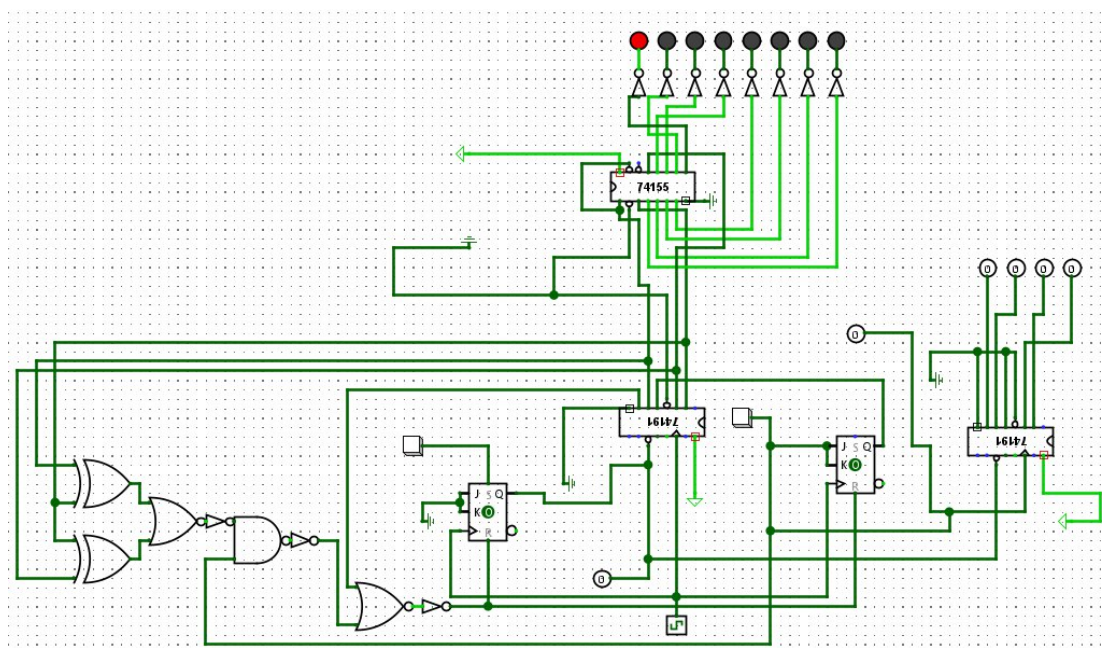
Ομάδα 10

Παπαδόπουλος Χαράλαμπος - 03120199

Στρουμπάκου Ειρήνη - 03121183

Εργαστήριο #5: Μοντελοποίηση και προσομοίωση του ελέγχου του Pong σε Logisim Evolution

Στο εργαστήριο αυτό καλούμαστε να ολοκληρώσουμε την προσομοίωση που είχαμε ξεκινήσει στο προηγούμενο εργαστήριο προσθέτοντας την λειτουργία του reset σε περίπτωση όπου ο χρήστης “χάσει”. Πιο συγκεκριμένα, ενώ στο προηγούμενο εργαστήριο είχαμε υλοποιήσει την λειτουργία του κυκλώματος με τα leds και με 2 buttons, πλέον ελέγχουμε εάν το κουμπί της απόκρουσης πατιέται την σωστή στιγμή. Ακολουθεί η εικόνα της υλοποίησής μας στο Logisim Evolution:



Προκειμένου να εξηγήσουμε την υλοποίησή μας, αρχικά πρέπει να αναφέρουμε τις δύο περιπτώσεις στις οποίες θεωρούμε ότι ο χρήστης χάνει και το σύστημα κάνει reset, δηλαδή σταματάει η κίνηση των leds.

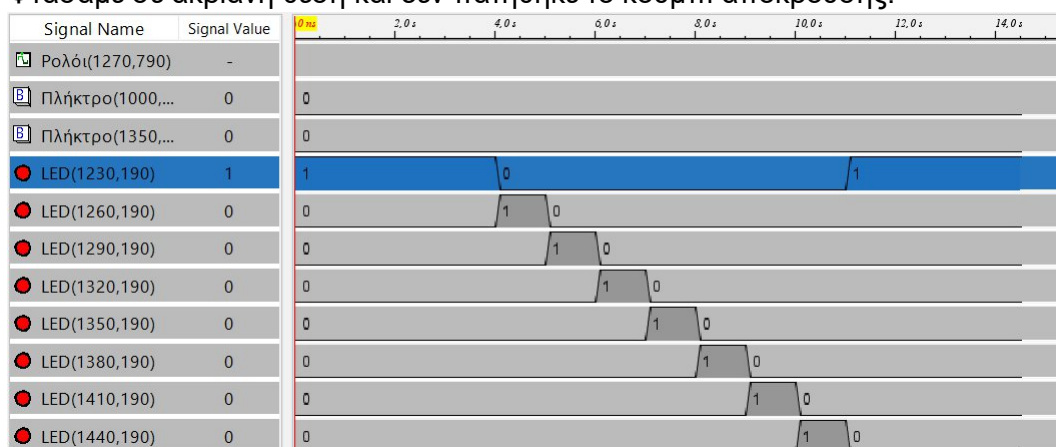
1. Εάν η μπάλα φτάσει σε ακριανή θέση και ο χρήστης δεν πατήσει το κουμπί της απόκρουσης, δηλαδή το δεξιό κουμπί του σχήματος, τότε ο χρήστης έχασε, γίνεται reset και σταματάει η κίνηση.
2. Εάν ο χρήστης πατήσει το κουμπί απόκρουσης σε μη ακριανή θέση, τότε πάλι θεωρούμε ότι έχασε, γίνεται reset και σταματάει η κίνηση.

Για την 1^η περίπτωση, παρατηρούμε ότι, όταν δεν πατηθεί το κουμπί όσο βρισκόμαστε σε ακριανή θέση, το Q_D του counter γίνεται 1. Από την άλλη στην 2^η περίπτωση θέλουμε να ελέγξουμε αν το κουμπί πατήθηκε σε μη ακριανή θέση. Οι ακριανές θέσεις έχουν τιμές 000 και 111, δηλαδή όλα τα bits είναι ίσα. Επομένως, θα κάνουμε reset εάν το κουμπί πατήθηκε ή εάν δεν έχω ίσα όλα τα bits. Έχοντας αυτά ως δεδομένα το reset θα συνδέεται σε μία OR με τις περιπτώσεις 1 και 2. Πιο αναλυτικά, το 1 άκρο της OR είναι το Q_D , το οποίο όταν είναι ίσο με 1 σηματοδοτεί την ακριανή θέση χωρίς πάτημα κουμπιού, και το άλλο άκρο είναι ο έλεγχος ισοδυναμίας των bits σε συνδυασμό με το πάτημα κουμπιού. Χρησιμοποιούμε λοιπόν για την περίπτωση 2 μία πύλη AND με μία είσοδο να γίνεται 1 μόνο όταν πατηθεί το κουμπί και την άλλη να γίνεται 1 όταν ικανοποιείται η ακόλουθη σχέση $(Q_a \text{ XOR } Q_b) \text{ OR } (Q_b \text{ XOR } Q_c)$. Τελικά, συνοψίζοντας η σχέση που υλοποιούμε είναι $\text{RESET} = (Q_d) \text{ OR } (\text{parity_Check AND Button})$ με $\text{parity_check} = (Q_a \text{ XOR } Q_b) \text{ OR } (Q_b \text{ XOR } Q_c)$.

Στο κύκλωμά μας φαίνεται επιπλέον δεξιά και ένας counter, ο οποίος έχει την ευθύνη του score, το οποίο αυξάνεται κάθε φορά που πατάμε σωστά το κουμπί απόκρουσης. Ειδικότερα, το ρολόι αυτού του counter το χειριζόμαστε με το πάτημα του κουμπιού απόκρουσης, οπότε όταν πατήσουμε το κουμπί αυτό αυξάνεται το count κατά 1. Αυτό απεικονίζεται στα 4 αρχικά ίσα με 0 pins που τοποθετήσαμε στα Q_A, Q_B, Q_C, Q_D. Στον counter αυτό γίνεται reset, όταν γίνει και στο υπόλοιπο κύκλωμα, αφού θέλουμε να σταματήσει το score και να ξαναξεκινήσει από την αρχή. Πρέπει, τέλος, να αναφέρουμε ότι η εκφώνηση της άσκησης μας επέτρεπε την χρήση μόνο hand, nor και αντιστροφών, ενώ εμείς χρησιμοποιήσαμε και xor. Πριν την υλοποίηση ρωτήσαμε ωστόσο τον κύριο Παναγόπουλο και μας είπε ότι δεν υπάρχει πρόβλημα όσον αφορά τις πύλες που θα χρησιμοποιήσουμε για την τελική μας υλοποίηση. Οπότε επιλέξαμε να χρησιμοποιήσουμε και xor για βέλτιστη υλοποίηση.

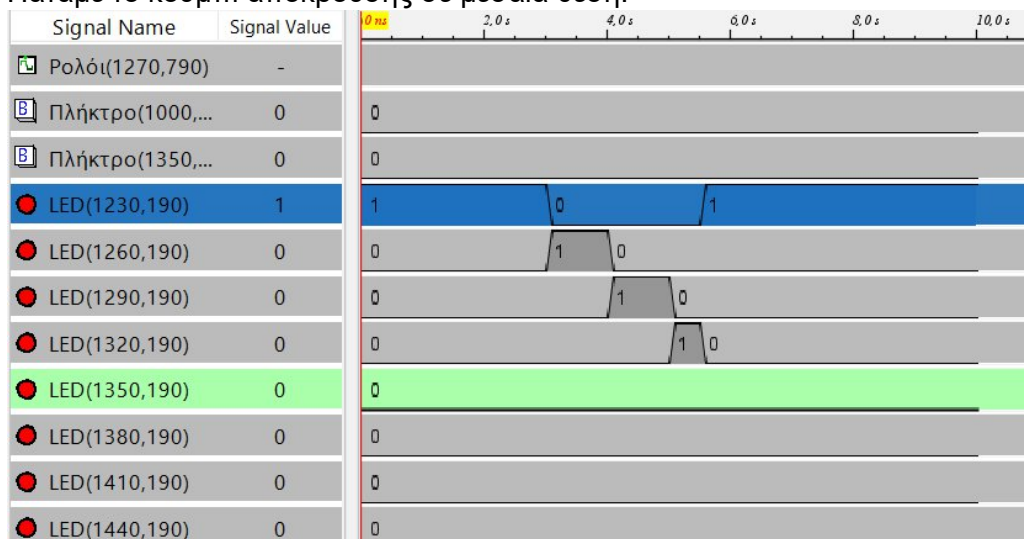
Ακολουθούν οι κυματομορφές από την υλοποίηση του κυκλώματος:

1. Φτάσαμε σε ακριανή θέση και δεν πατήθηκε το κουμπί απόκρουσης:



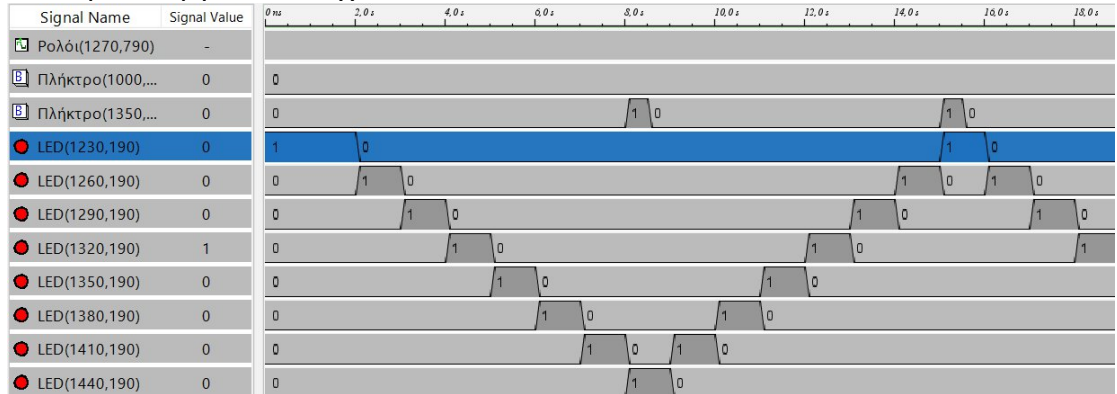
Παρατηρούμε ότι αρχικά, πριν πατήσουμε το κουμπί εκκίνησης, βρισκόμαστε σε κατάσταση αδράνειας και είναι αναμμένο το led 1 χωρίς να αλλάζει κάτι. Έπειτα, στα 4s περίπου, όπου πατάμε το κουμπί, ξεκινάει η κίνηση και βλέπουμε ότι πράγματι ανάβουν διαδοχικά τα επόμενα leds. Φτάνουμε στο τελευταίο λίγο μετά τα 10s και δεν πατάμε το κουμπί απόκρουσης, όπως θα έπρεπε. Επομένως, ο χρήστης χάνει και επιστρέφουμε στην αρχική κατάσταση αδράνειας, όπου είναι αναμμένο μόνο το led 1 και δεν υπάρχει κίνηση.

2. Πατάμε το κουμπί απόκρουσης σε μεσαία θέση:



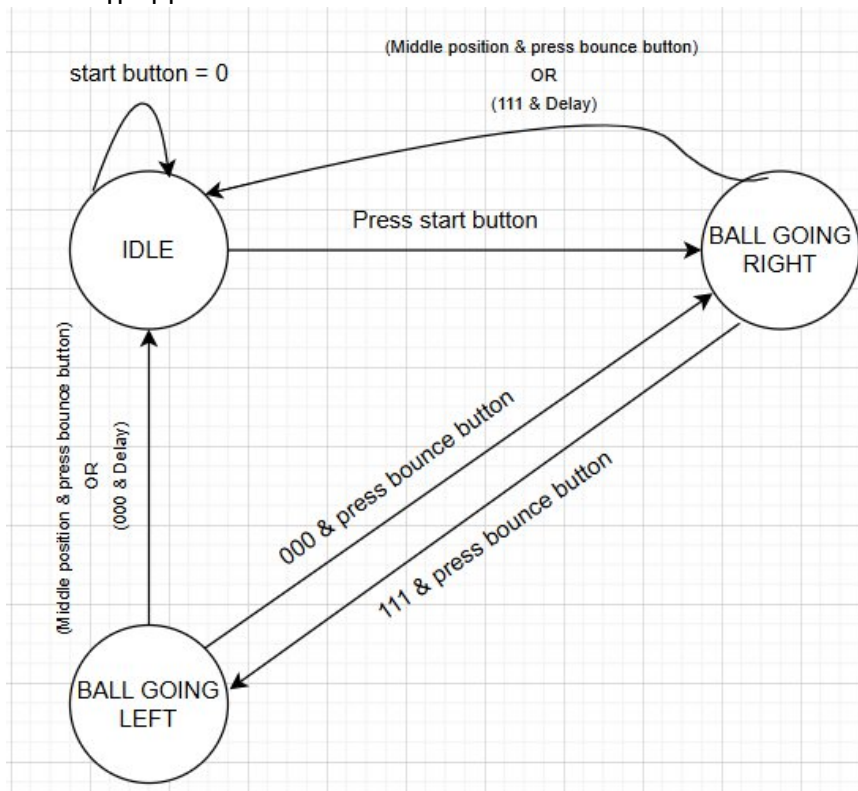
Στην συγκεκριμένη περίπτωση επιλέξαμε να πατήσουμε το κουμπί απόκρουσης, όσο βρισκόμασταν στο 4^ο led. Παρατηρούμε και εδώ όπως και στην 1^η περίπτωση την αρχική κατάσταση αδράνειας μέχρι το πάτημα του κουμπιού περίπου στα 3s. Έπειτα, ξεκινάει πάλι η κίνηση προς τα δεξιά, μέχρι που πατιέται στα 5,5s το κουμπί απόκρουσης. Καθώς, δεν βρισκόμαστε σε ακριανή θέση, ο παίκτης χάνει και επιστρέφουμε στην αρχική κατάσταση αδράνειας και βλέπουμε σταθερή τιμή 1 στο led 1.

3. Σωστή λειτουργία του παιχνιδιού:



Παρατηρούμε εδώ ότι το παιχνίδι λειτουργεί σωστά, αφού αρχικά με το πάτημα του κουμπιού εκκίνησης ξεκινάει η κίνηση προς τα δεξιά. Μόλις φτάσω στην δεξιά ακριανή θέση και πατήσω το κουμπί απόκρουσης, πράγματι ξεκινάει η κίνηση προς τα αριστερά. Το παιχνίδι συνεχίζεται κανονικά, αφού δεν έχει κάνει λάθος ο χρήστης.

Ακολουθεί το διάγραμμα καταστάσεων:



Οι καταστάσεις που θεωρήσαμε είναι 3: κατάσταση αδράνειας(idle), κίνηση προς τα δεξιά και κίνηση προς τα αριστερά. Στην σωστή λειτουργία του παιχνιδιού πηγαίνω από την κατάσταση idle στο ball going right με το κουμπί εκκίνησης. Έπειτα, υπάρχει κίνηση μεταξύ του ball going left και ball going right, όταν πληρούνται οι προϋποθέσεις. Όπως, δηλαδή φαίνεται και στο διάγραμμα, πάω από ball going left στο ball going right μόνο όταν είμαι σε τιμή 000 και πατάω κουμπί απόκρουσης και αντίστοιχα από ball going right στο ball going left με τιμή 111. Τέλος, στις 2 περιπτώσεις που έχουμε ήδη αναφέρει και χάνει ο χρήστης επιστρέφω στην κατάσταση idle, όπου και παραμένω όσο start button = 0.