

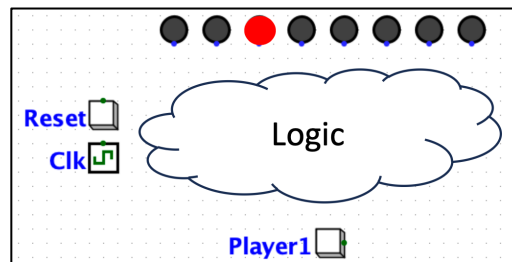


Νικόλαος Γιαννόπουλος – 03122086
Κωνσταντίνος Καργάκος – 03122166

Εργαστήριο Ψηφιακών Συστημάτων

Εργαστηριακή Αναφορά #5 – Μοντελοποίηση και προσομοίωση του ελέγχου του Pong σε Logisim Evolution

Στο εργαστήριο αυτό θα μοντελοποιήσουμε και θα προσομοιώσουμε τη λογική του ελέγχου του Pong στο Logisim Evolution.



Η λειτουργία του συστήματος έχει ως εξής:

Αρχικά, ο Counter και ο Decoder είναι Disabled και τα Led είναι κλειστά. Αφού πατηθεί και αφεθεί το κουμπί Reset αρχίζει το παιχνίδι, δηλαδή δίνουμε σήμα ENABLE στον Counter και στον Decoder. Ταυτόχρονα θέτουμε RESET στον Counter και σήμα UP, ώστε η μπάλα να ξεκινήσει από το 1^ο Led και να μετακινείται προς το 8^ο Led σε κάθε παλμό του ρολογιού.

Μετά την έναρξη του παιχνιδιού, ο παίκτης καλείται να πατάει το κουμπί Play όταν η μπάλα βρεθεί σε ακριανό Led. Τότε, δίνουμε αντίθετο σήμα DOWNUP στον Counter, ώστε να “αποκρουστεί” η μπάλα προς την αντίθετη κατεύθυνση. Αυτό επαναλαμβάνεται για όσο ο παίκτης πατάει το κουμπί Play, όταν η μπάλα βρίσκεται σε ακριανή θέση.

Το παιχνίδι τελειώνει (Game Over) και κάνουμε Disable τον Counter και τον Decoder σε μία από τις δύο περιπτώσεις:

1. Το κουμπί Play πατιέται όταν η μπάλα βρίσκεται σε ενδιάμεση θέση.
2. Ο παίκτης δεν πατήσει έγκαιρα το κουμπί Play όταν η μπάλα βρίσκεται σε ακριανή θέση.

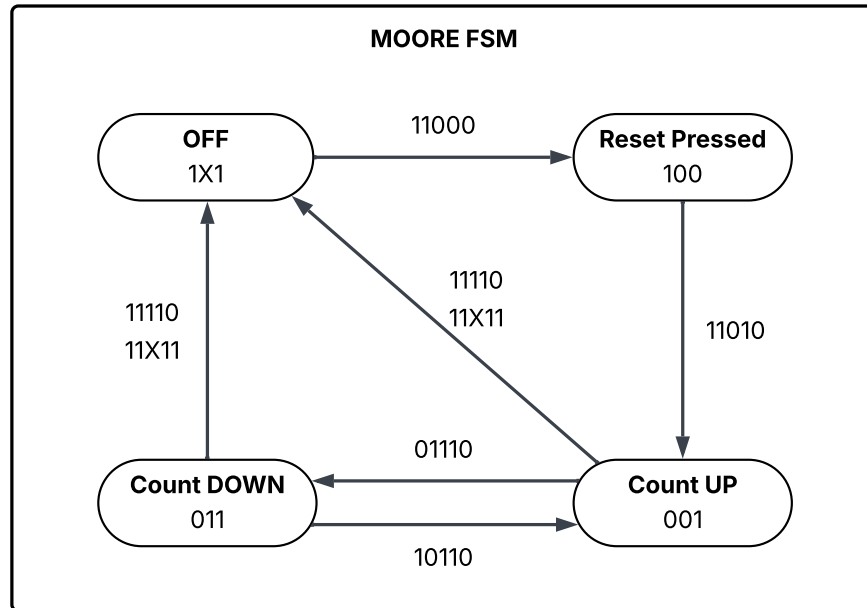
Επίσης, αν πατηθεί το κουμπί Reset, ενώ το παιχνίδι έχει αρχίσει, τότε σταματάει, κάνοντας Disable τον Counter και τον Decoder.

Διάγραμμα Καταστάσεων

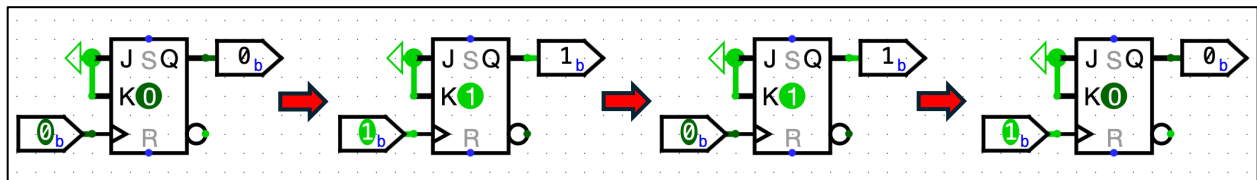
Inputs: EdgeR EdgeL Play Reset QD

Outputs: ENABLE DOWNUP LOAD

Signal	0	1
EdgeR, Edge L	LED ON	LED OFF
Play	Released	Pressed
Reset	Pressed	Released
ENABLE	Enabled	Disabled
DOWNUP	Up	Down
LOAD	Load Value	Nothing



Υλοποιούμε την παραπάνω λογική με τη χρήση JK Flip-Flop:



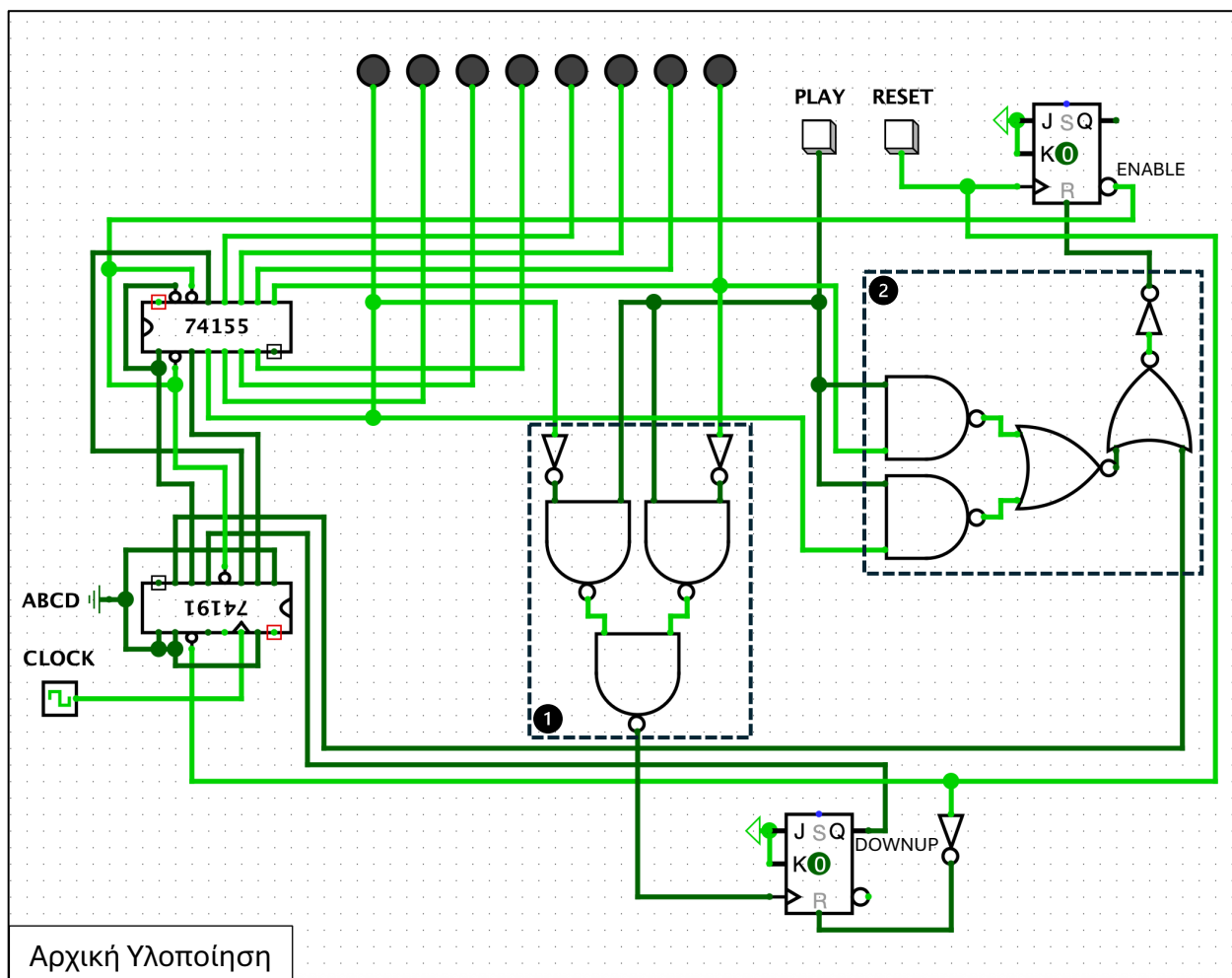
Θέτουμε τις εισόδους J και K του JK Flip-Flop στο λογικό 1. Κατά την εναλλαγή του CLK από 0 σε 1 η έξοδος του FF αντιστρέφεται κάθε φορά. Επίσης, εάν θέσουμε σε οποιαδήποτε στιγμή λογικό 1 στην είσοδο CLR του FF, τότε η έξοδος γίνεται 0.

Αυτή η διάταξη του JK Flip-Flop είναι χρήσιμη στις καταστάσεις που θέλουμε η έξοδος να παραμένει ίδια για ένα χρονικό διάστημα, και να αλλάζει με το πάτημα των κουμπιών ή άλλων σημάτων. Χρησιμοποιούμε δύο τέτοιες διατάξεις JK Flip-Flop:

Το πρώτο JK Flip-Flop χρησιμοποιούμε για να θέσουμε το σήμα DOWNUP του Counter.

Το σήμα αυτό θέλουμε να εναλλάσσεται μόνο σε συγκεκριμένες περιπτώσεις (ακριανή θέση + πάτημα Play), ενώ θέλουμε να μηδενίζεται (CLR) κατά την εκκίνηση του παιχνιδιού.

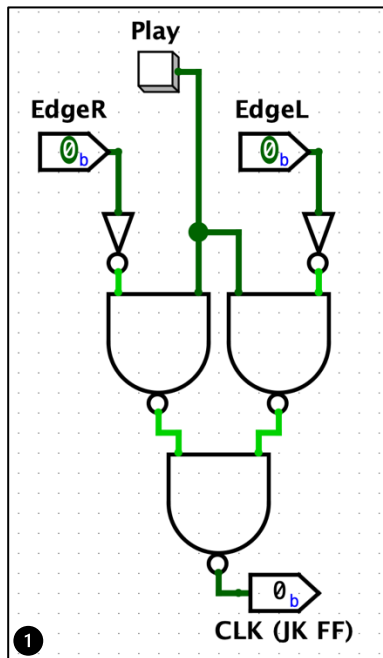
Το δεύτερο JK Flip-Flop χρησιμοποιούμε για να θέσουμε το σήμα ENABLE του Counter και του Decoder. Το σήμα αυτό θέλουμε να εναλλάσσεται μόνο κατά το πάτημα του Reset (αρχή/τέλος παιχνιδιού), ενώ θέλουμε να μηδενίζεται (CLR) σε περίπτωση Game Over. Επειδή το σήμα ENABLE είναι αντίστροφης λογικής χρησιμοποιούμε την έξοδο Q' του FF.



Στο ολοκληρωμένο κύκλωμα φτιάξαμε δύο κυκλώματα λογικής που ελέγχουν για τις σημαντικές καταστάσεις και ελέγχουν τις εισόδους των δύο JK Flip-Flop:

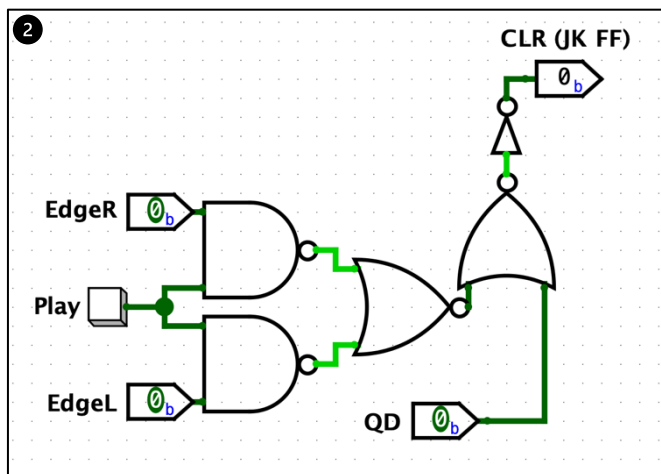
Το πρώτο λογικό κύκλωμα έχει έξοδο το CLK του FF που ελέγχει το σήμα DOWNUP. Δέχεται ως είσοδο το σήμα από τα δύο ακριανά Led (EdgeR και EdgeL), καθώς και από το κουμπί Play. Η έξοδος γίνεται από 0 σε 1 μόνο όταν το κουμπί Play είναι πατημένο και ταυτόχρονα είναι αναμμένο ένα από τα ακριανά Led.

Στη συγκεκριμένη υλοποίηση δεν ελέγχουμε αν η μπάλα είναι δεξιά ή αριστερά, αλλά απλώς εναλλάσσουμε το σήμα DOWNUP όταν αυτή φτάσει σε οποιαδήποτε άκρη και πατηθεί το κουμπί. Αυτό το κάνουμε γιατί, αρχικά και πατώντας το κουμπί Reset, θέτουμε Reset στον Counter, ώστε η μπάλα να αρχίσει από το Led 0 και κάνουμε CLR το FF, ώστε η αρχική φορά να είναι UP. Από εκεί και πέρα όταν η μπάλα φτάσει στην απέναντι θέση απλά εναλλάσσουμε το DOWNUP.



Το δεύτερο λογικό κύκλωμα έχει έξοδο το CLR του FF που ελέγχει το σήμα ENABLE. Δέχεται ως είσοδο το σήμα από τα δύο ακριανά Led (EdgeR και EdgeL), από το κουμπί Play αλλά και από το σήμα QD του Counter.

Η σκέψη μας είναι το κύκλωμα αυτό να δίνει έξοδο 1 μόνο σε περίπτωση Game Over. Δηλαδή στις δύο παρακάτω περιπτώσεις:



1. Όταν και τα δύο ακριανά Led δεν είναι αναμμένα, δηλαδή η μπάλα βρίσκεται στη μέση, και ταυτόχρονα πατηθεί το κουμπί Play.
2. Όταν η μπάλα περάσει το ακριανό Led και το κουμπί δεν πατηθεί. Για να ελέγξουμε αυτήν την περίπτωση παίρνουμε το σήμα QD του Counter, δηλαδή το τέταρτο bit (Most Significant Bit) του αριθμού που παράγει ο Counter. Όταν το QD γίνει 1 η μέτρηση του Counter έχει ξεπεράσει την τιμή 7 (0111) ή την τιμή 0 (0000) για Count UP ή DOWN αντίστοιχα, οπότε και καταλαβαίνουμε ότι το κουμπί δεν πατήθηκε εγκαίρως.

Κόστος Αρχικής Υλοποίησης σε τεχνολογία TTL:

- 2x JK Flip-Flop (1x IC)
- 5x NAND (2x IC)
- 4x NOT (1x IC)
- 2x NOR (1x IC)

Σύνολο: 5x IC

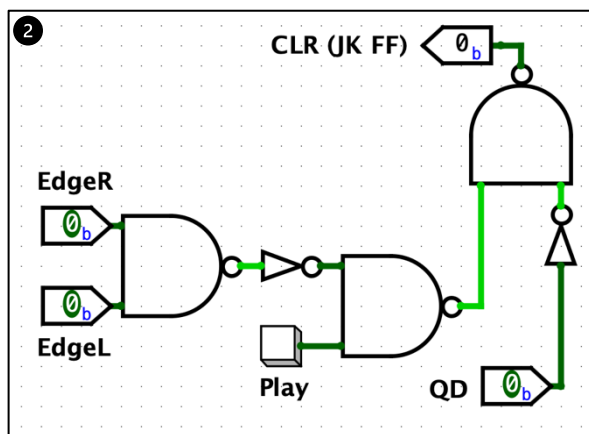
Το Button Play είναι κανονικής λογικής (1 στο πάτημα), ενώ το Button Reset είναι αντίστροφης λογικής (0 στο πάτημα), μιας και το δεύτερο θέλουμε να δίνει θετικό παλμό στο CLK του FF αφού πατηθεί και αφεθεί. Η υλοποίηση της βέλτιστης λογικής των Button κυκλωματικά βοηθάει στη μειωμένη χρήση λογικών πυλών NOT.

Απλοποίηση/Βελτιστοποίηση Υλοποίησης

Μπορούμε να βελτιστοποιήσουμε τα λογικά κυκλώματα, ώστε να μειώσουμε το κόστος υλοποίησης με τις παρακάτω δύο απλοποιήσεις:

Το λογικό κύκλωμα ❷ είναι δυνατόν να απλοποιηθεί στο διπλανό κύκλωμα κρατώντας την λειτουργία του ίδια. Το νέο κύκλωμα έχει μόνο 3 πύλες NAND (αντί για 4), 2 πύλες NOT (αντί για 1) και καθόλου πύλες NOR (αντί για 2).

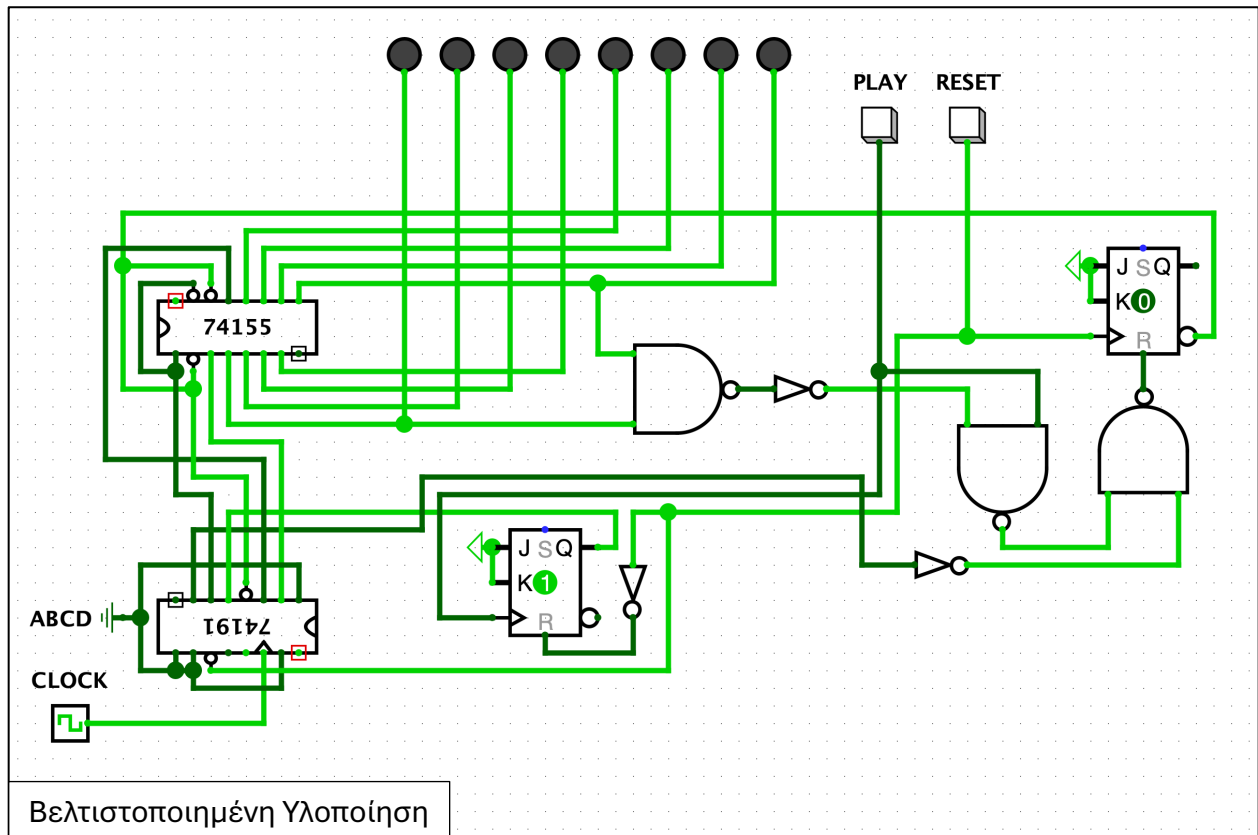
Η απλοποίηση έγινε με τη χρήση του πίνακα αληθείας μέσα από το Logisim Evolution.



Το λογικό κύκλωμα ❶ που είναι υπεύθυνο για την είσοδο του Flip-Flop που ελέγχει το σήμα DOWNUP είναι δυνατόν να παραληφθεί τελείως και την είσοδο του Flip-Flop να ελέγχει κατευθείαν το κουμπί Play. Η λειτουργικότητα του συστήματος παραμένει η ίδια. Αυτό συμβαίνει διότι, παρόλο που, χωρίς αυτό το λογικό κύκλωμα, η εναλλαγή του DOWNUP γίνεται με το πάτημα του Play ανεξαρτήτως της θέσης της μπάλας, εάν υπάρξει κατάσταση που προβλέπει Game Over αυτή θα ανιχνευτεί από το λογικό κύκλωμα ❷ και το παιχνίδι θα τερματίσει.

Ουσιαστικά, το σύστημα πλέον κάνει εναλλαγή από UP σε DOWN και αντίστροφα με κάθε πάτημα του Play χωρίς περιορισμό και μόνο ελέγχει για καταστάσεις που οδηγούν σε Game Over στις οποίες και τερματίζει αμέσως.

Τελικά, το κύκλωμα διαμορφώνεται ως εξής:



Κόστος Βελτιστοποιημένης Υλοποίησης σε τεχνολογία TTL:

- 2x JK Flip-Flop (1x IC)
- 3x NAND (1x IC)
- 3x NOT (1x IC)

Σύνολο: 3x IC

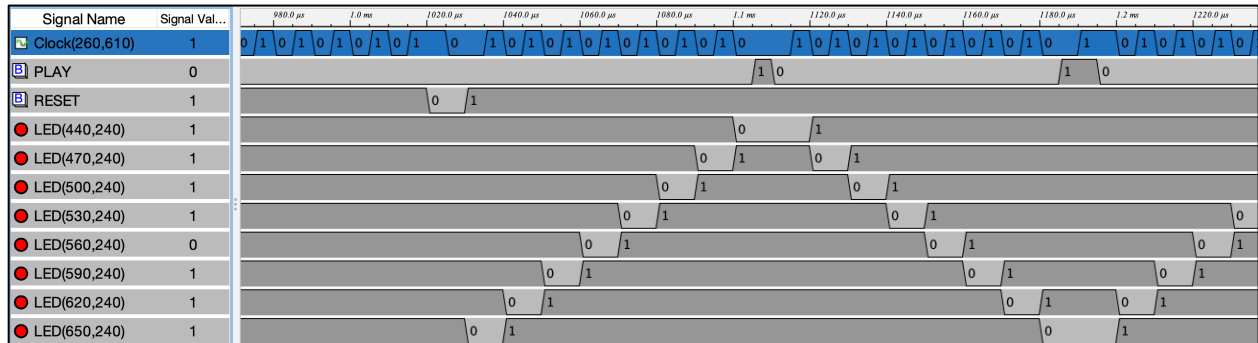
Βλέπουμε σημαντική βελτίωση στο κόστος υλοποίησης σε σχέση με την αρχική υλοποίηση.

Από το κύκλωμα που έχουμε φτιάξει στο Breadboard έως τώρα έχουμε αφαιρέσει τον διακόπτη ON/OFF, που είχαμε βάλει, αφού πλέον το σήμα *ENABLE* του Counter και του Decoder ελέγχεται από το λογικό κύκλωμα.

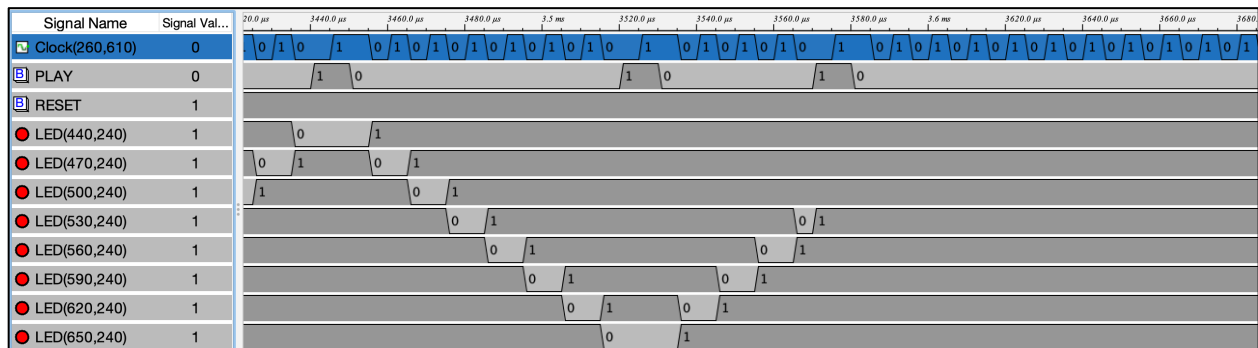
Αρχείο Υλοποίησης: Στο αρχείο lab05.circ περιλαμβάνονται 2 κυκλώματα. Το **pong_initial** (Αρχική Υλοποίηση) και το **pong_optimized** (Βελτιστοποιημένη Υλοποίηση).

Κυματομορφές Λειτουργίας

Επιτυχείς αποκρούσεις: όταν πατήσουμε και αφήσουμε το Reset, τότε το παιχνίδι ξεκινά και ανάβει το 1^ο Led. Η μπάλα κινείται και φτάνει στο 8^ο Led, όπου πατάμε το Play και αποκρούεται. Έπειτα, η κατεύθυνση της μπάλας αλλάζει και αυτό επαναλαμβάνεται κάθε φορά και εφόσον το Play πατηθεί όταν η μπάλα βρίσκεται σε ακριανή θέση.



Πρώιμη απόκρουση: εάν η μπάλα βρίσκεται σε κάποια ενδιάμεση θέση και πατηθεί το Play, τότε το παιχνίδι σταματάει και τα Led σβήνουν. Για να αρχίσει ξανά το παιχνίδι πρέπει να πατηθεί και να αφεθεί το Reset.



Καθυστερημένη απόκρουση: εάν η μπάλα φτάσει σε ακριανό Led και δεν πατηθεί έγκαιρα το Play, τότε το παιχνίδι σταματάει και τα Led σβήνουν. Για να αρχίσει ξανά το παιχνίδι πρέπει να πατηθεί και να αφεθεί το Reset.

