

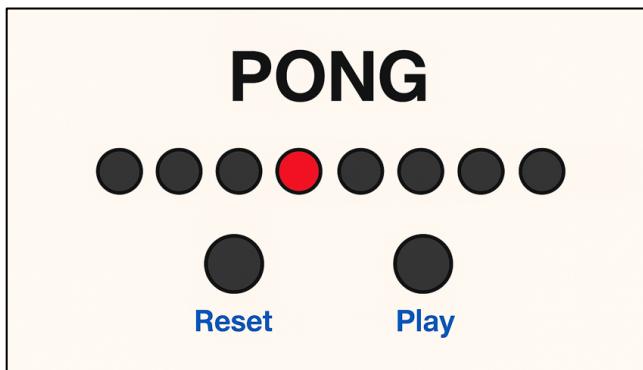


Νικόλαος Γιαννόπουλος – 03122086
Κωνσταντίνος Καργάκος – 03122166

Εργαστήριο Ψηφιακών Συστημάτων

Εργαστηριακή Αναφορά #6 – Λογική Ελέγχου και Ολοκλήρωση του Pong

Στο εργαστήριο αυτό θα προσθέσουμε τον έλεγχο στο κύκλωμα και θα ολοκληρώσουμε την υλοποίηση ενός λειτουργικού παιχνιδιού Pong.



Η λειτουργία του Pong έχει ως εξής:

Αρχικά, τα LED είναι σβηστά και το παιχνίδι ξεκινά όταν ο παίκτης πατήσει και αφήσει το κουμπί Reset. Η οθόνη ενεργοποιείται και η μπάλα αρχίζει να κινείται από τη μία άκρη προς την άλλη. Κάθε φορά που φτάνει σε ακριανό LED και ο παίκτης πατήσει το κουμπί Play η μπάλα “αποκρούεται” προς την αντίθετη κατεύθυνση. Αυτό επαναλαμβάνεται για όσο ο παίκτης πατάει το κουμπί Play, όταν μόνο η μπάλα βρίσκεται σε ακριανή θέση.

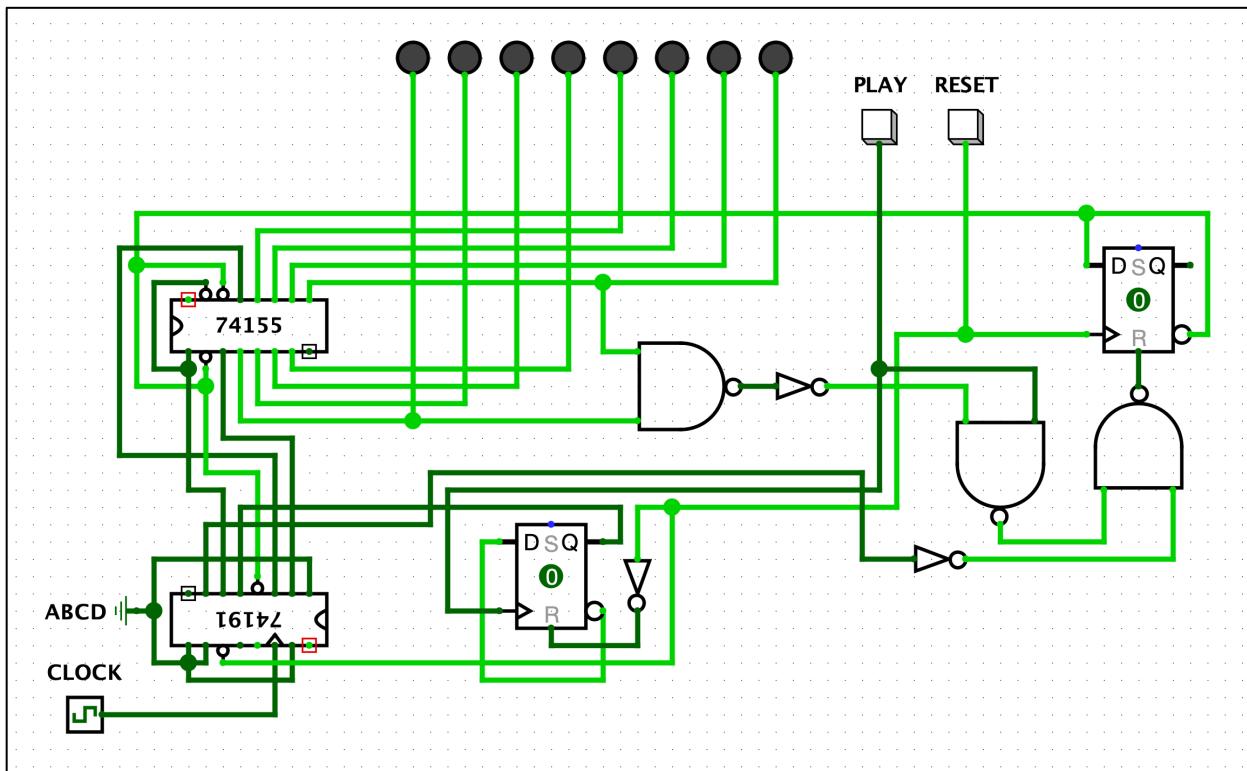
Τα LED σβήνουν, το παιχνίδι σταματάει και επανέρχεται στην αρχική του κατάσταση σε μία από τις εξής τρεις περιπτώσεις:

1. Ο παίκτης πατά το κουμπί Play, ενώ η μπάλα βρίσκεται σε ενδιάμεση θέση.
2. Ο παίκτης δεν πατά έγκαιρα το κουμπί Play, όταν η μπάλα βρίσκεται σε ακριανή θέση.
3. Ο παίκτης πατά το κουμπί Reset, οποτεδήποτε θέλει να σταματήσει το παιχνίδι.

Ο παίκτης μπορεί να ξεκινήσει ξανά το παιχνίδι πατώντας και αφήνοντας το κουμπί Reset.

Λογική σχεδίαση κυκλώματος:

Η τελική υλοποίηση του λογικού κυκλώματος είναι παρόμοια με αυτή που κάναμε στο προηγούμενο εργαστήριο. Λόγω έλλειψης των ολοκληρωμένων JK-FF αντικαταστήσαμε τα JK-FF με D-FF. Τα σήματα εξόδων των FF παραμένουν ίδια, ενώ για να πετύχουμε συμπεριφορά ίδια με αυτή των JK-FF που είχαμε υλοποιήσει, συνδέσαμε μεταξύ τους την είσοδο D με την έξοδο Q' σε κάθε D-FF. Το κύκλωμα, λοιπόν, διαμορφώθηκε ως εξής:



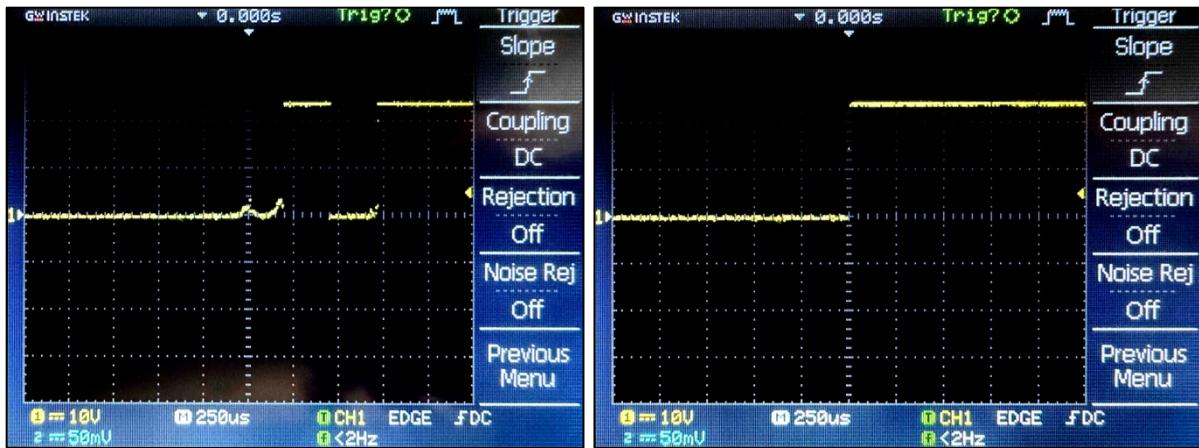
Φτιάχνοντας το κύκλωμα στο Breadboard διαπιστώσαμε τα εξής δύο προβλήματα:

1. Υπάρχει **Bounce** στα **Button Play** και **Reset**. Αυτό προκαλεί ανεπιθύμητες στιγμιαίες μεταβολές των σημάτων **Play** και **Reset**, τα οποία αλλάζουν τις εξόδους των δύο **Flip Flop** και κατ' επέκταση τη συμπεριφορά του κυκλώματος. Το πρόβλημα αυτό λύνουμε με την προσθήκη **RC** κυκλώματος ($\tau=R \cdot C = 1\text{ms}$) σε κάθε κουμπί, το οποίο καθυστερεί και μεταβάλει σταδιακά την τάση εξόδου του κουμπιού. Για να ψηφιοποιήσουμε την έξοδο του **RC** κυκλώματος χρησιμοποιούμε το ολοκληρωμένο **7414**, το οποίο είναι ένας λογικός **inverter** με καθυστέρηση (Schmitt – Trigger).

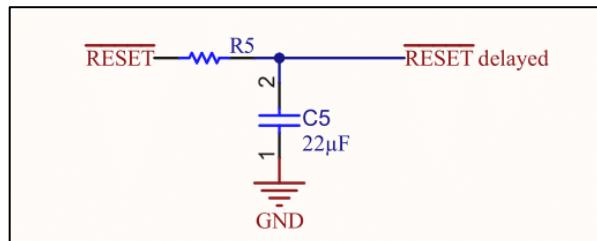
Ουσιαστικά αυτό το ολοκληρωμένο αλλάζει την κατάσταση της εξόδου του (LOW, HIGH) όταν η τάση εισόδου περάσει μια συγκεκριμένη τιμή. Το **RC** και το **7414** μαζί υλοποιούν το κύκλωμα **Debounce**, το οποίο παίρνει ένα σήμα με απρόβλεπτη συμπεριφορά (Bouncing, όπως φαίνεται και στην φωτογραφία) και το μετατρέπει σε ένα καθαρό

Ψηφιακό σήμα, το οποίο μπορούμε να χρησιμοποιήσουμε με ασφάλεια για τις εισόδους του λογικού μας κυκλώματος.

Στο **αριστερό γράφημα** φαίνεται το bounce του button, ενώ στο **δεξί γράφημα** φαίνεται η έξοδος του button με την προσθήκη του debounce κυκλώματος:

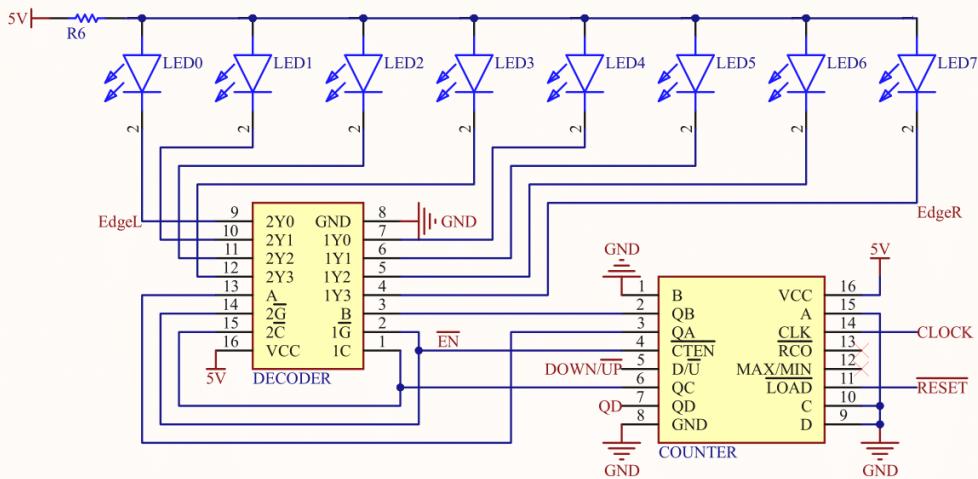


- Κατά το πάτημα του κουμπιού **Reset**, ενεργοποιείται το σήμα **CLR** του ενός **Flip Flop** (αυτού που είναι υπεύθυνο για την εναλλαγή του **DOWNTUP**), το οποίο μένει ενεργοποιημένο μέχρι να αφεθεί το κουμπί **Reset**. Την στιγμή, όμως, που αφήνεται το κουμπί Reset, ταυτόχρονα αλλάζει και η κατάσταση του άλλου **Flip Flop** (αυτού που ελέγχει το **Enable**). Πολλές φορές παρατηρούμε, με το άφημα του κουμπιού **Reset**, να αλλάζει πρώτα η κατάσταση του **Flip Flop** και ύστερα να απενεργοποιείται το σήμα **CLR**, οπότε ενεργοποιείται κατάσταση Game Over από το λογικό κύκλωμα και η κατάσταση του **Flip Flop**, που ελέγχει το **Enable**, επανέρχεται κατευθείαν στην αρχική του. Εμείς θέλουμε με το άφημα του **Reset** να απενεργοποιείται πρώτα το σήμα **CLR** και μετά να αλλάζει κατάσταση το άλλο **Flip Flop** και όχι το αντίθετο. Μιας και στη λογική του κυκλώματος οι δύο αυτές αλλαγές γίνονται ταυτόχρονα από το ίδιο κουμπί, προσθέτουμε ένα **RC** κύκλωμα ($\tau=R*C=22ms$), ώστε να καθυστερήσει η αλλαγή της κατάστασης του **Flip Flop**, για να εξασφαλίσουμε ότι θα έχει πρώτα απενεργοποιηθεί το σήμα **CLR**.

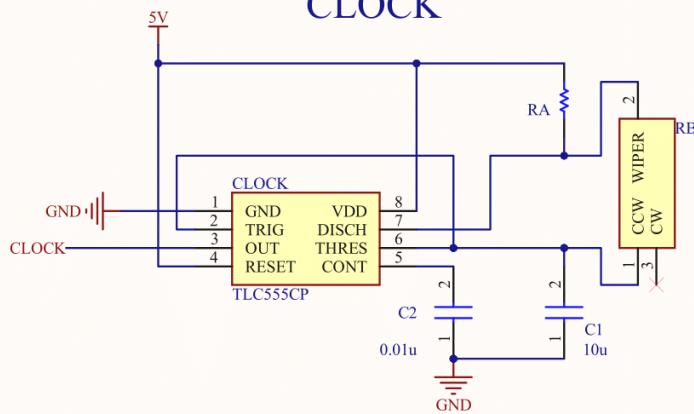


Συνολική Σχεδίαση Κυκλώματος

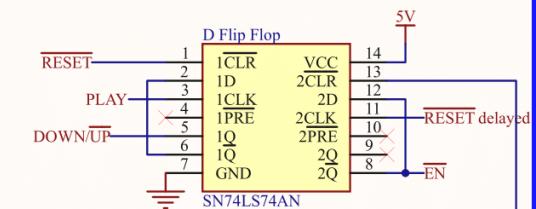
DISPLAY



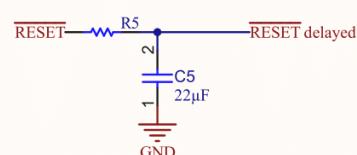
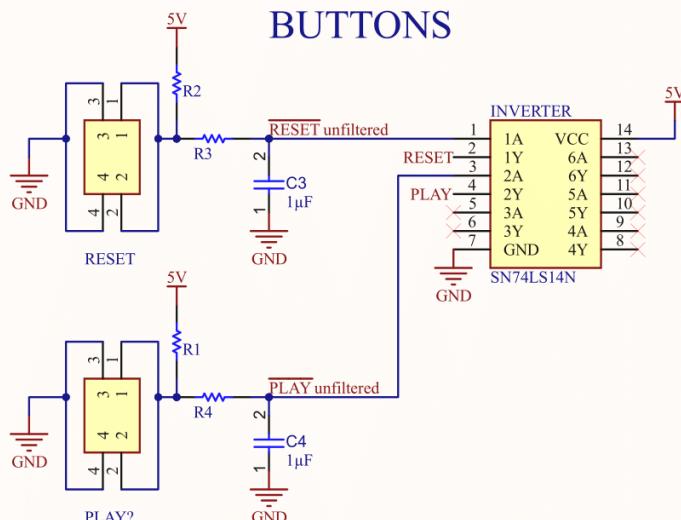
CLOCK



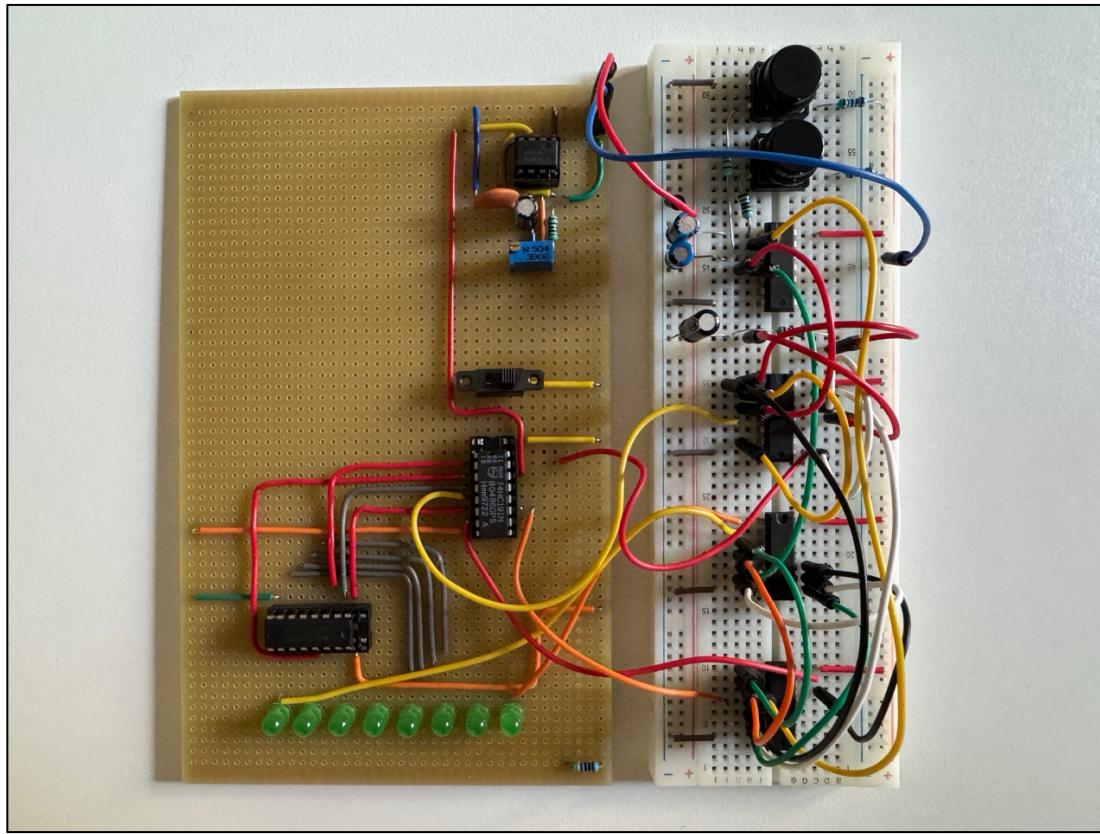
LOGIC CIRCUIT



BUTTONS

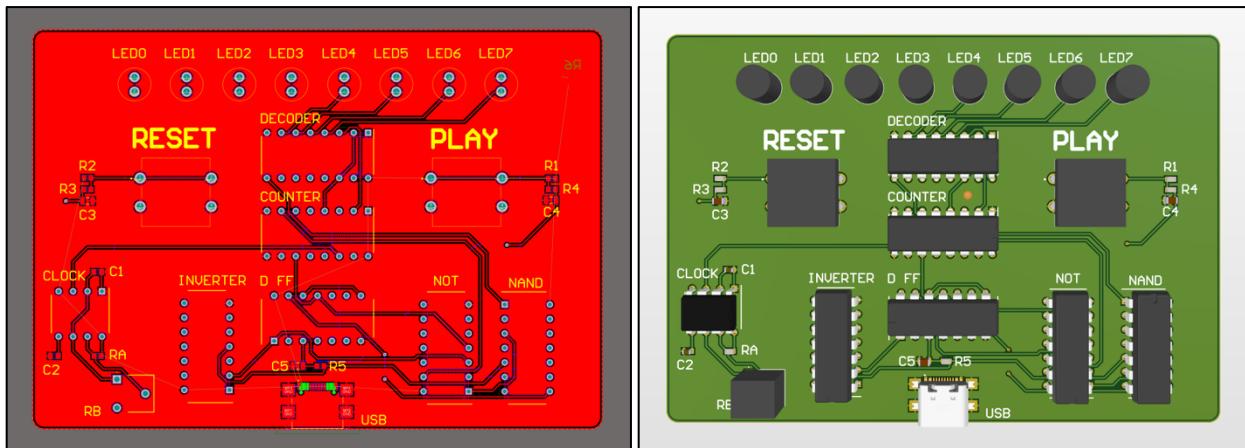


Κατασκευή Κυκλώματος στο Breadboard – Διάτρητη Πλακέτα



Σχεδίαση Κυκλώματος σε PCB

Σχεδιάσαμε το τελικό κύκλωμα σε πλακέτα PCB με στόχο να φτιάξουμε το παιχνίδι Pong σε μια συμπαγής και καθαρή πλακέτα, η οποία θα έχει μικρότερο μέγεθος, καλύτερη χωροταξία, μειωμένη πολυπλοκότητα και μεγαλύτερη αξιοπιστία.



Λόγω περιορισμένου χρόνου, δεν προλάβαμε να προμηθευτούμε την πλακέτα PCB και να υλοποιήσουμε το κύκλωμα σε αυτή μέχρι την παράδοση της αναφοράς. Σκοπεύουμε να ολοκληρώσουμε το παιχνίδι στη πλακέτα PCB μόλις την παραλάβουμε.