Σταθουδάκης Σταύρος, ΑΕΜ:03512 17/11/2023 Lab1

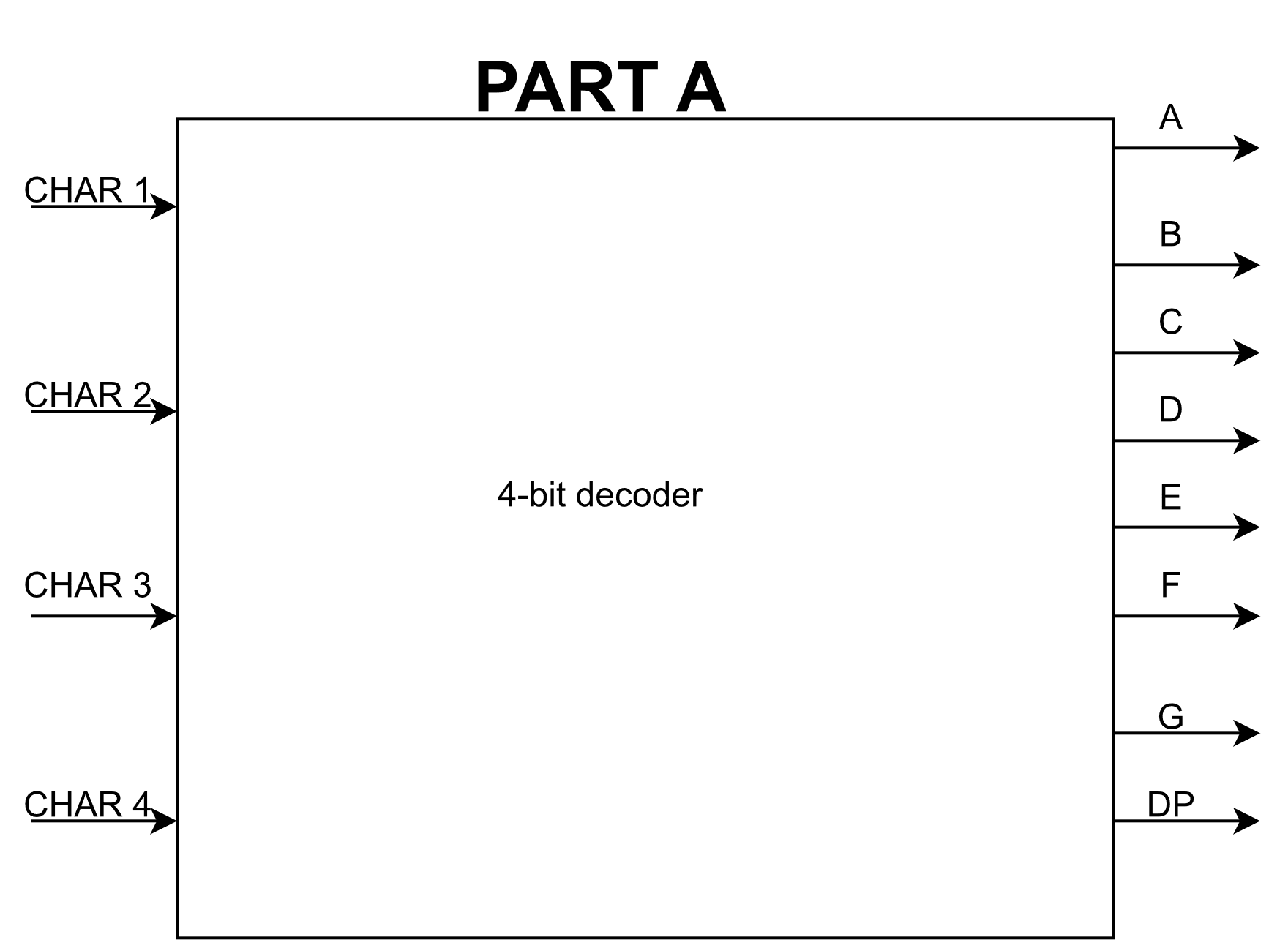
# Περίληψη

# Εισαγωγή

Οι στόχοι της εργασίας ήταν να δημιουργήσουμε ένα περιστρογικό μύνημα σε ένα LED display 4 χαρακτήρων. Η εργασία στην ουσία μας βοήθησε να δημιουργήσουμε ένα ένα τα modules τα οποία θα χρησιμοποιούσαμε για να καταφέρουμε να φτάσουμε στον τελικό μας στόχο. Ενδιάμεσα μας έβαλε να δημιουργήσουμε κάποιες πιο εύκολες μορφές αυτού του τελικού στόχου έτσι ώστε να προλάβουμε να εξοικειωθούμε. Πχ για να καταφέρουμε να έχουμε περιστροφικό μύνημα, έπρεπε πρώτα να καταφέρουμε να δείχουμε το μύνημα στην οθόνη. Αυτός ήταν ο σκοπός του PartB (το πιο δύσκολο για εμένα). Έπειτα έπρεπε να περιστρέφουμε το μύνημα με το πάτημα ενός κουμπιού (Partc) και τέλος να περιστρέφεται μόνιμα από μόνο του μετα από ένα συγκεκριμένο χρονικό διάστημα. Επιτεύχθηκαν όλοι αυτοί οι στόχοι

# Μέρος Α - Υλοποίηση Αποκωδικοποίησης 7-τμημάτων

## Υλοποίηση

* Το κύκλωμα π δημιουργείτε σε αυτό το μέρος είναι ένας 4-bit αποκωδικοποιητής ο οποίος έχει 8 outputs, τα οποίο στην ουσία επιλέγουν ποία τμήματα του LED display θα είναι αναμένα, για να δημιουργηθεί το κατάλληλο σχήμα.
* Ο κώδικας περιέχει μία ασύχρονη always ή οποία έχει ως έισοδο 4 bit και έξοδο 8 bit.
* Μέσα στην always έχει χρησιμοποιθεί μία case εντολή για να κάνει τον κώδικα πιο ευανάγνωστο. Τα assignments μέσα στην Verilog είναι blocking.

## Επαλήθευση

* Στο testbench δοκιμάζονται όλες οι πιθανές τιμές που υπάρχουν ως input (16 τιμες αφού ειναι 4 bit) και έπειτα επαληθεύονται εξετάζοντας τις κυματομορφές των σημάτων εξόδου.
* Το ποσοστό κάλυψης του κώδικα είναι 100% καθώς όπως ανέφερα δοκιμάζονται όλες οι πιθανές τιμές που υπάρχουν ως input.
* Δεν υπήρχαν προβήματα

# Μέρος Β - Οδήγηση Τεσσάρων Ψηφίων

## Υλοποίηση

* Στον κώδικα εμπεριέχονται:
  + τα modules για την μείωση του clock speed , από αυτό που μας δίνει η fpga σε αυτό που επιθυμούμε (20ns -> 200ns).
  + Μία συγχρονισμένη always (δλδ ενα flip flop) το οποίο αυξάνει τον counter με κάθε αλλαγή του ρολογιού.
  + Ένα module (anodes) το οποίο βάση του counter ανοίγει τις κατάλληλες ανόδους και αλλάζει το output char.
  + Ενα module, το οποίο καλεί το partA και βάση του char επιλέγει ποιο μύνημα θα φαίνεται στο LED display
* 