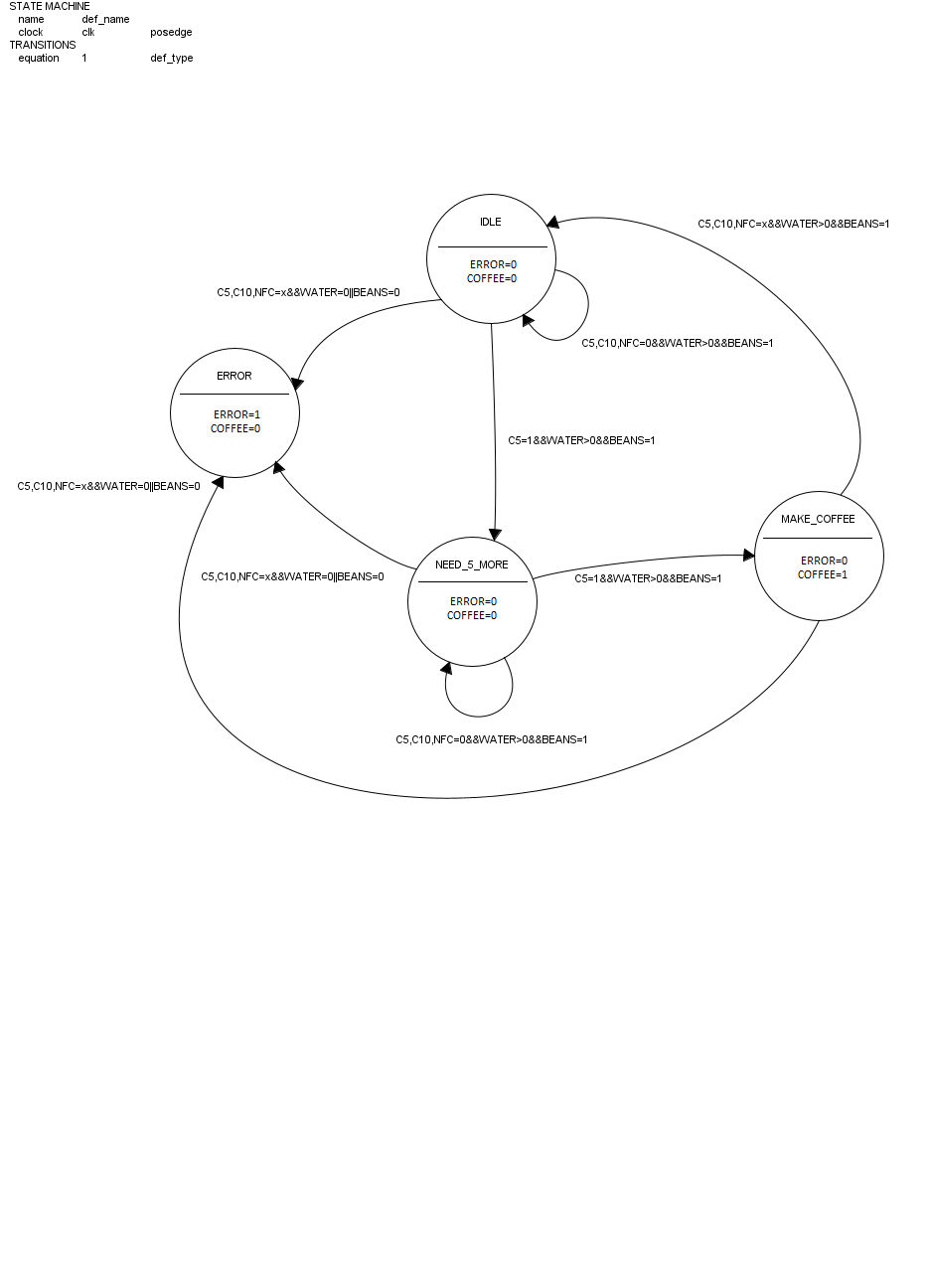
# 3η ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ

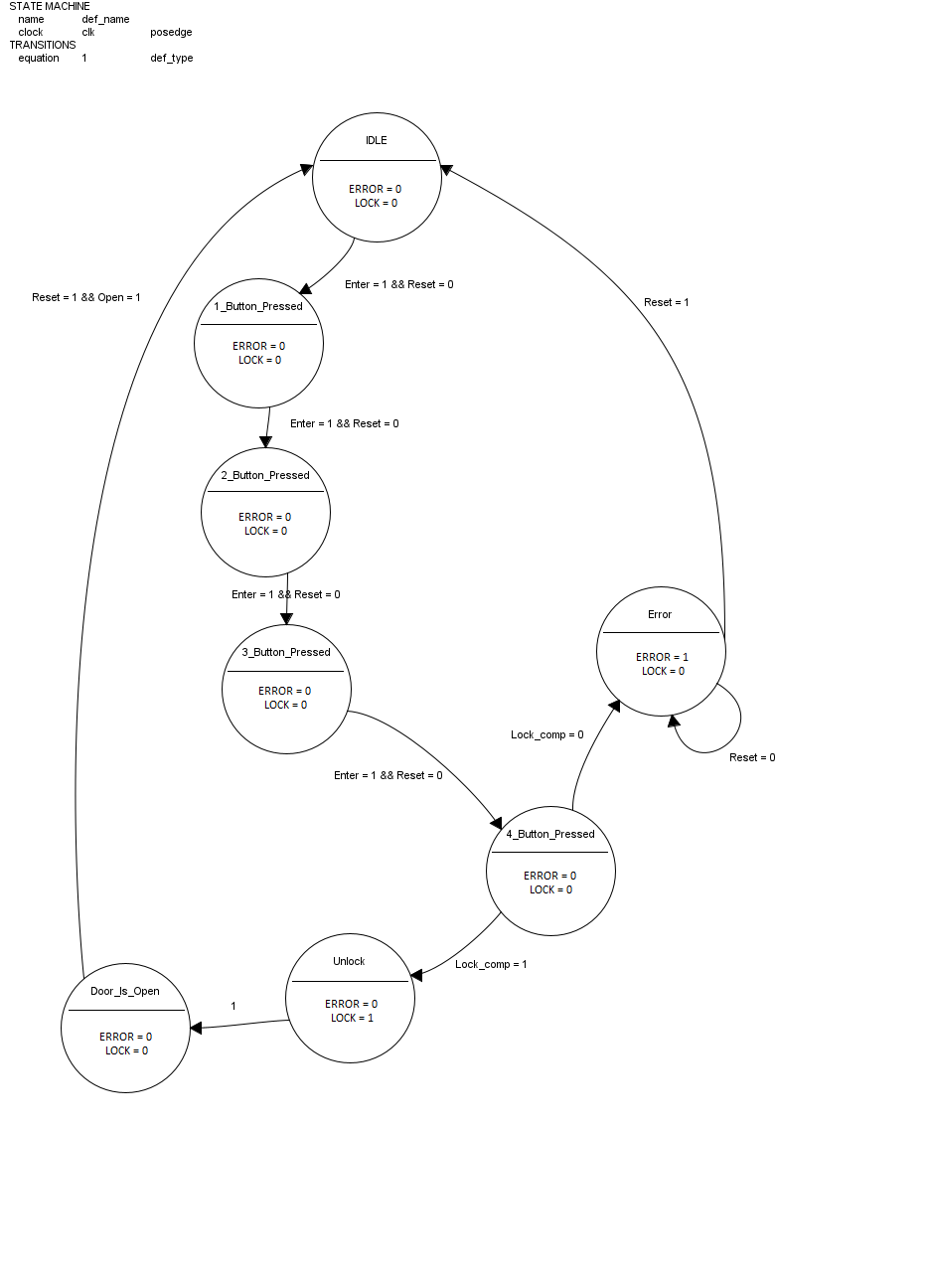
# “ΣΧΕΔΙΑΣΜΟΣ ΣΥΣΤΗΜΑΤΩΝ VLSI”

Ζητούμενα:

* 1. Για το πρώτο ζητούμενο μας ζητήθηκε η υλοποίηση του FSM της λειτουργίας μιας μηχανής καφέ. Ακολουθεί μια απλοποιημένη έκδοση του FSM το οποίο χρησιμοποιήσαμε για να γράψουμε τον κώδικα του ερωτήματος 1.2.



* 1. Για το ερώτημα 2 του πρώτου ζητούμενου μας ζητήθηκε να γράψουμε τον κώδικα που αντιστοιχεί στο παραπάνω FSM καθώς και ένα Testbench για το τεστάρισμα του κυκλώματος. Οι κώδικες επισυνάπτονται στα αρχεία my\_vending.v και my\_vending\_tb.v στον φάκελο SRC. Για την υλοποίηση του κυκλώματος υλοποιήσαμε 3 always blocks, ένα για το ρολόι του κυκλώματος, το σύγχρονο Reset και τον συγχρονισμό των αλλαγών καταστάσεων και των αλλαγών στις εξόδους, ένα για τις αλλαγές καταστάσεων και ένα για τις αλλαγές στις εξόδους με βάση την παρούσα κατάσταση στην οποία βρίσκεται το σύστημα την κάθε στιγμή. Για τις αλλαγές καταστάσεων χρησιμοποιήσαμε την case της Verilog ενώ ως default κατάσταση θεωρήσαμε την κατάσταση IDLE στην οποία το κύκλωμα βρίσκεται σε αδράνεια. Για το Testbench του κυκλώματος χρησιμοποιήσαμε τη λογική του να θέσουμε τιμές εισόδου, ώστε να ελέγξουμε αν πραγματοποιούνται σωστά οι μεταβάσεις στις αντίστοιχες καταστάσεις, ενώ εμείς περιμένουμε τα σήματα να πάρουν τις κατάλληλες τιμές.
  2. Για το πρώτο ερώτημα του 2ου ζητούμενου υλοποιήσαμε έναν parallel input parallel output register (pipo\_register module) στον οποίο αποθηκεύεται ο αριθμός με τον οποίο θα συγκρίνεται η τελική είσοδος του τελικού κυκλώματος, ενώ θα ελέγχεται από τα σήματα enable (είσοδος επίτρεψης αλλαγής τιμής), clr (καθαρισμός καταχωρητή) και set (θέτει τον καταχωρητή στην τιμή 0xFFFF). Στην συνέχεια υλοποιήσαμε έναν συγκριτή (comparator module) ο οποίος παίρνει στην είσοδό του 2 αριθμούς και έχει σαν έξοδο το σήμα equal το οποίο γίνεται 1 όταν οι 2 αριθμοί είναι ίσοι. Τέλος υλοποιήσαμε το κύκλωμα lockcomp το οποίο χρησιμοποιεί Instances των παραπάνω κυκλωμάτων κατάλληλα συνδεδεμένα. Για το Testbench του παραπάνω κυκλώματος θέσαμε μια τιμή στον καταχωρητή του lockcomp και εισάγαμε όλες τις τιμές στην είσοδο pressed\_code μέχρι να πάρουμε το σήμα equal που μας επιβεβαιώνει ότι βρέθηκε ο κωδικός που εισάγαμε πιο πριν (τεχνική παρόμοια με το brute force). Οι κώδικες επισυνάπτονται στα αρχεία lockcomp.v και lockcomp \_tb.v στον φάκελο SRC.
  3. Για το 2ο ερώτημα μας ζητήθηκε η υλοποίηση του FSM της λειτουργίας μιας κλειδαριάς. Ακολουθεί μια απλοποιημένη έκδοση του FSM το οποίο χρησιμοποιήσαμε για να γράψουμε τον κώδικα του ερωτήματος 2.3.



* 1. Για τον κώδικα του 3ου ερωτήματος ακολουθήσαμε την ίδια τακτική με το ερώτημα 1.2. και υλοποιήσαμε το αντίστοιχο FSM. Οι κώδικες επισυνάπτονται στα αρχεία dlock.v.

Σημείωση: Δεν υλοποιήθηκε η λειτουργία αλλαγής κωδικού και το Testbench του ερωτήματος