



1^η ΟΜΑΔΑ ΑΣΚΗΣΕΩΝ
ΓΙΑ ΤΟ ΜΑΘΗΜΑ "Συστήματα Μικροϋπολογιστών"
(παράδοση μέχρι 27 Απριλίου 2020)

Ασκήσεις προσομοίωσης

Να γίνει έλεγχος των προγραμμάτων με τη χρήση του προσομοιωτή του εκπαιδευτικού συστήματος μLAB. Περιγραφή του συστήματος, το πρόγραμμα προσομοίωσης, τις οδηγίες χρήσης και εγκατάστασης θα βρείτε στο site του μαθήματος.

(Οι 3 ασκήσεις που ακολουθούν είναι όλες ασκήσεις προσομοίωσης - να υλοποιηθούν και να δοκιμαστούν στο πρόγραμμα προσομοίωσης του εκπαιδευτικού συστήματος μLAB)

1^η ΑΣΚΗΣΗ: Σε ένα μΥ-Σ 8085 να γραφεί σε assembly το παρακάτω πρόγραμμα που δίνεται σε γλώσσα μηχανής και να εξηγηθεί η λειτουργία του:

06 01 3A 00 20 FE 00 CA 13 08 1F DA 12 08 04 C2 0A 08 78 2F 32 00 30 CF

Το πρόγραμμα υποθέτουμε ότι είναι φορτωμένο στη μνήμη με αρχή τη διεύθυνση 0800 και δίνεται για διευκόλυνσή σας ότι οι bold κωδικοί είναι εντολές.

Η διαδικασία της αποκωδικοποίησης (disassembly) θα διευκολυνθεί με τη χρήση του πίνακα 2 του παραρτήματος 2 των σημειώσεων *Εισαγωγή στο Εκπαιδευτικό Σύστημα mLAB*. Να δοθεί το πρόγραμμα σε assembly (και με συμβολικές διευθύνσεις). Επίσης να γίνει και το διάγραμμα ροής του προγράμματος. Τι αλλαγές πρέπει να γίνουν για να έχουμε σε συνεχόμενη μορφή τη λειτουργία του παραπάνω προγράμματος, δηλαδή να επαναλαμβάνεται χωρίς τέλος;

Υπόδειξη: Μπορείτε να ακολουθήσετε τη διαδικασία της εφαρμογής 1 της 1^{ης} εργαστηριακής άσκησης των σημειώσεων - *Εισαγωγή στο Εκπαιδευτικό Σύστημα mLAB*.

2^η ΑΣΚΗΣΗ: Να γραφεί σε assembly πρόγραμμα που να απεικονίζει, στη θύρα εξόδου 3000 Hex, ένα αναμμένο led το οποίο να κινείται αριστερά (από το LSB προς το MSB) και όταν φτάνει στο όγδοο (MSB) να κινείται δεξιά (προς το LSB) κ.ο.κ. (θέσεις led 1234567876543212... κ.λπ.) όταν το LSB της θύρας των dip switch (θύρα εισόδου 2000 Hex) είναι ON. Αλλιώς, οποτεδήποτε είτε στην αρχή είτε ενδιάμεσα, γίνεται OFF το LSB των dip switch, το led να κάνει κυκλική κίνηση (θέσεις led 1234567812... κ.λπ.). Τέλος, το 2^ο LSB της θύρας των dip switch όταν γίνεται ON να ανάβει χωρίς να κινείται το led της θέσης 0. Στη συνέχεια, όταν ξαναγίνει OFF να συνεχίζεται η κίνησή του από το σημείο που είχε μείνει, σύμφωνα με το LSB των dip switch.

Να γίνει χρήση της θύρας εισόδου dip switch (θέση μνήμης 2000 Hex) και της θύρας εξόδου των LED (που αντιστοιχεί στη θέση μνήμης 3000 Hex – προσοχή στην αντίστροφη λογική απεικόνισης). (Διάρκεια ανάμματος ~½ sec).

Μπορείτε να εισάγετε χρονοκαθυστερήση (μέσω της ρουτίνα DELB (προκαλεί μεταβλητή καθυστέρηση ίση με την τιμή του ζεύγους BC επί 1 ms) που υπάρχει στο *Παράρτημα 1* (σελ. 91: των σημειώσεων - *Εισαγωγή στο Εκπαιδευτικό Σύστημα mLAB*). Για να κάνετε έλεγχο ορθότητας υπάρχουν 2 τρόποι: α/ βηματική εκτέλεση του προγράμματός σας (προσοχή να έχετε αντικαταστήσει τη ρουτίνα DELB με 3 εντολές NOP), β/ για να ελέγξετε σε ένα επιλεγμένο σημείο του προγράμματος την ορθότητα λειτουργίας μέχρι εκεί, εισάγετε την εντολή CF (RST 1, σελ. 60: των σημειώσεων - *Εισαγωγή στο Εκπαιδευτικό Σύστημα mLAB*) που προκαλεί επιστροφή στο monitor πρόγραμμα όπου μπορείτε να εξετάσετε τιμές καταχωρητών και θέσεων μνήμης.

Να σημειωθεί ότι χρειάζεται να δίνεται στην αρχή ενός προγράμματος η εντολή IN 10H που αίρει την προστασία της μνήμης του εκπαιδευτικού συστήματος μLAB επιτρέποντας έτσι πρόσβαση για αποθήκευση μεταβλητών και δεδομένων οπουδήποτε στην διαθέσιμη μνήμη RAM του συστήματος (0800 – 0BFF Hex) βλ. χάρτη μνήμης μLAB - σελ. 7 των σημειώσεων - *Εισαγωγή στο Εκπαιδευτικό Σύστημα mLAB*. Αλλιώς επιτρέπεται πρόσβαση μόνο στην περιοχή των διευθύνσεων 0B00 - 0BAF Hex (χώρος δεδομένων χρήστη).

3^η ΑΣΚΗΣΗ: Να επεκταθεί το 4ο παράδειγμα που αφορά στη μετατροπή δυαδικού αριθμού των 8 bits σε δεκαδική μορφή 2 ψηφίων (σελ. 84 του βιβλίου) χωρίς τον περιορισμό να είναι μικρότεροι του 100_{10} . Τα 8 bit του δυαδικού αριθμού x υποθέτουμε ότι δίνονται από τα dip switches της πόρτας εισόδου (θέση μνήμης 2000 Hex). Το αποτέλεσμα να εμφανισθεί στην πόρτα εξόδου των LED (που αντιστοιχεί στη θέση 3000 Hex) ως εξής: οι μονάδες στα 4 LSB και οι δεκάδες 4 MSB. Στην περίπτωση που ο αριθμός είναι μεγαλύτερος του 99, το αποτέλεσμα να εμφανίζεται σε modulo 100 δηλαδή να εμφανίζεται ο αριθμός $x-100$ ή $x-200$. Το πρόγραμμα να είναι συνεχούς λειτουργίας.

*Παρατήρηση: Τα προγράμματα να συνοδεύονται υποχρεωτικά στα κυριότερα σημεία τους από **πολύ σύντομα** σχόλια.*

Θεωρητικές Ασκήσεις

4^η ΑΣΚΗΣΗ: Να μελετηθεί από τεχνικο-οικονομική άποψη η κατασκευή μιας φορητής ηλεκτρονικής συσκευής με τη χρήση 3^{ων} διαφορετικών τεχνολογιών:

1. Χρήση διακριτών στοιχείων και ολοκληρωμένων μονάδων (I.C.) όπως μικροελεγκτών, περιφερειακών, μνημών κ.λπ. τα οποία συναρμολογούνται σε μια σε μια σχετικά μεγάλη πλακέτα. Το αρχικό κόστος σχεδίασης του συστήματος υποθέτουμε ότι είναι 20.000€. Το κόστος των I.C. ανά πλακέτα υποθέτουμε ότι είναι 15€ και η κατασκευή της πλακέτας με την συναρμολόγησή της επίσης 15€ ανά πλακέτα.
2. Χρήση FPGAs και μικρού αριθμού περιφερειακών τοποθετημένα σε μια μικρή πλακέτα. Αρχικό κόστος σχεδίασης του συστήματος: 10.000€, κόστος ανά πλακέτα των I.C.: 60€, κόστος κατασκευής της πλακέτας και συναρμολόγησης ανά τεμάχιο: 10€.
3. Σχεδίαση ειδικού SoC με μια πολύ μικρή πλακέτα. Αρχικό κόστος σχεδίασης: 300.000€, κόστος παραγωγής ανά τεμάχιο των I.C.: 1€, κόστος κατασκευής πλακέτας και συναρμολόγησης ανά τεμάχιο: 1€.

Να σχεδιαστούν οι αντίστοιχες καμπύλες κόστους ανά τεμάχιο συσκευής για τις 3 τεχνολογίες. Να υποδειχθούν για την κάθε μία τεχνολογία οι τρεις περιοχές αριθμού τεμαχίων που είναι συμφερότερες (δηλ. έχουν χαμηλότερο κόστος). Επίσης διερευνήστε για ποια τιμή των I.C. στην τεχνολογία των FPGAs θα μπορούσε να εξαφανιστεί η επιλογή της 1^{ης} τεχνολογίας; Σχολιάστε τα αποτελέσματα.

Ασκήσεις στην Γλώσσα περιγραφής υλικού Verilog

Να επιλυθούν τα παρακάτω προβλήματα από το βιβλίο: “Ψηφιακή Σχεδίαση” του Morris Mano & Michael Ciletti (5η έκδοση):

5^η ΑΣΚΗΣΗ: (i) Να δοθεί σε επίπεδο πυλών (δομική) περιγραφή των εξής λογικών συναρτήσεων: $F1=A(CD + B) + BC'D'$, $F2(A, B, C, D)=\Sigma (0, 2, 3, 5, 7, 8, 10, 11, 14, 15)$, $F3=ABC + (A + B)CD + (B + CD)E$, $F4=A(BC + D + E) + CDE$.

(ii) Να δοθεί επίσης ο κώδικας Verilog για τις ίδιες συναρτήσεις σε Μοντελοποίηση ροής δεδομένων (με εντολές συνεχούς ανάθεσης).

6^η ΑΣΚΗΣΗ: (i) Σχεδιάστε το λογικό διάγραμμα των ψηφιακών κυκλωμάτων που ορίζονται από τις παρακάτω περιγραφές Verilog:

(a) **module** Circuit_A (A, B, C, D, F);
 input A, B, C, D;
 output F;
 wire w, x, y, z, a, d;
 or (x, B, C, d);
 and (y, a, C);
 and (w, z, B);

```

and (z, y, A);
or (F, x, w);
not (a, A);
not (d, D);
endmodule

```

```

(b) module Circuit_B (F1, F2, F3, A0, A1, B0, B1);
    output F1, F2, F3;
    input A0, A1, B0, B1;
    nor (F1, F2, F3);
    or (F2, w1, w2, w3);
    and (F3, w4, w5);
    and (w1, w6, B1);
    or (w2, w6, w7, B0);
    and (w3, w7, B0, B1);
    not (w6, A1);
    not (w7, A0);
    xor (w4, A1, B1);
    xnor (w5, A0, B0);
endmodule

```

```

(c) module Circuit_C (y1, y2, y3, a, b);
    output y1, y2, y3;
    input a, b;
    assign y1 = a || b;
    and (y2, a, b);
    assign y3 = a && b;
endmodule

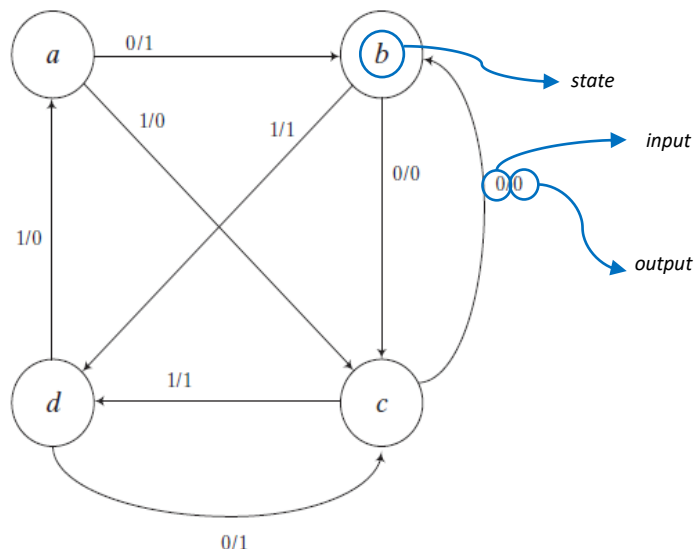
```

(ii) Δώστε την ιεραρχική περιγραφή HDL σε επίπεδο πύλης για έναν αθροιστή-αφαιρέτη τεσσάρων bit για μη προσημασμένους δυαδικούς αριθμούς.

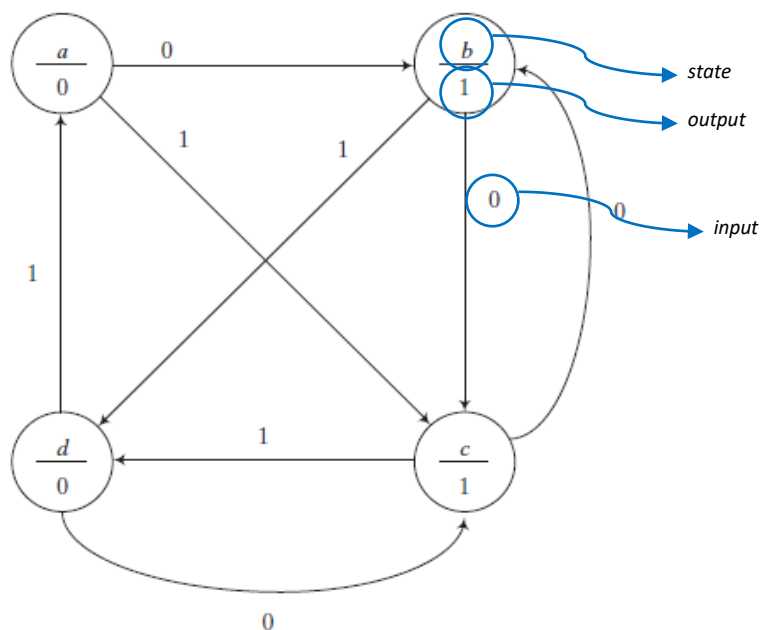
(iii) Χρησιμοποιώντας τον τελεστή υπό συνθήκη (? :), γράψτε μια περιγραφή ροής δεδομένων HDL ενός αθροιστή-αφαιρέτη τεσσάρων bits μη προσημασμένων αριθμών.

7η ΑΣΚΗΣΗ: Οι παρακάτω ζητούμενες περιγραφές συμπεριφοράς, να δοθούν χωρίς τα προγράμματα δοκιμαστικής εισόδου για επαλήθευση ορθότητας).

(i) Γράψτε ένα μοντέλο Verilog της Mealy FSM (η έξοδος είναι συνάρτηση της παρούσας εισόδου και κατάστασης) που περιγράφεται από το διάγραμμα κατάστασης του παρακάτω σχήματος:



(ii) Γράψτε ένα μοντέλο Verilog της Moore FSM (η έξοδος είναι συνάρτηση μόνο της παρούσας κατάστασης) που περιγράφεται από το διάγραμμα κατάστασης του παρακάτω σχήματος:



(iii) Γράψτε μια περιγραφή συμπεριφοράς Verilog ενός up-down counter των 4 bit που διαθέτει 2 εισόδους (Up/Down και Clear). Να γίνει χρήση D-flip flop. Να θεωρήσετε τις εισόδους σύγχρονες (με το ρολόι).

(ii) Παρατηρήσεις

Οι αναφορές κάθε ομάδας ασκήσεων θα παραδίδονται ηλεκτρονικά (θα γίνονται upload στο site του μαθήματος).

Η παρούσα 1^η ομάδα ασκήσεων θα πρέπει να παραδοθεί μέχρι την **Δευτέρα 27 Απριλίου 2020**. Οι ασκήσεις παραδίδονται κατά ομάδες των δυο (2) ή του ενός (1) ατόμου χωρίς αυτό να έχει επίπτωση στο βαθμό. Στην 1^η περίπτωση θα πρέπει να γίνεται upload **και από τους 2 σπουδαστές** η ίδια αναφορά και με τα δυο ονόματα στην 1^η σελίδα της αναφοράς. Αλλιώς θα θεωρείται ότι δεν έχει παραδοθεί. Σε ξεχωριστά αρχεία να είναι οι κώδικες των προσομοιώσεων (αυτό δεν αφορά στα αρχεία της Verilog). Τα αρχεία με τους κώδικες και η αναφορά να **συμπιεστούν σε ένα αρχείο** που αυτό θα γίνει upload. Οι αναφορές σας να είναι σύντομες και περιεκτικές. Τα προγράμματα να έχουν πολύ σύντομα σχόλια όπου κρίνετε ότι χρειάζονται για να κατανοηθεί η λειτουργία τους.

Ο βαθμός από τις Ομάδες Ασκήσεων, που συνολικά θα δοθούν, θα ληφθεί υπόψη κατά 20%.

Η Εργαστηριακή Άσκηση (αν πραγματοποιηθεί φέτος) θα ληφθεί υπόψη κατά 10% και ο βαθμός της γραπτής εξέτασης κατά 70% στην διαμόρφωση του τελικού βαθμού.

Διευκρινίζεται ότι όσοι περσινοί σπουδαστές παρέδωσαν τις Ομάδες των Ασκήσεων δεν χρειάζεται να τις παραδώσουν φέτος (αυτές ισχύουν για 1 έτος).

ΠΑΡΑΤΗΡΗΣΗ: Αν θέλετε να εγκαταστήσετε το mLab σε σύστημα linux μπορείτε να κάνετε χρήση της Virtual Machine στο Oracle Virtualbox.