



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
ΕΡΓΑΣΤΗΡΙΟ ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΩΝ & ΥΛΙΚΟΥ
ΕΡΓΑΣΤΗΡΙΑΚΕΣ ΑΣΚΗΣΕΙΣ ΓΙΑ ΤΟ ΜΑΘΗΜΑ: ΗΡΥ 312
ΟΡΓΑΝΩΣΗ ΥΠΟΛΟΓΙΣΤΩΝ

ΕΑΡΙΝΟ ΕΞΑΜΗΝΟ 2017-18
Εργαστήριο 4

**Υλοποίηση σύνθετων εντολών σε έναν επεξεργαστή
πολλαπλών κύκλων**

Σκοπός του Εργαστηρίου

1. Ο ορισμός της αρχιτεκτονικής ενός συνόλου σύνθετων εντολών.
2. Η σχεδίαση και ολοκλήρωση ενός επεξεργαστή πολλαπλών κύκλων που υλοποιεί σύνθετες εντολές και θα βασίζεται σε προηγούμενο παραδοτέο.
3. Η βαθύτερη κατανόηση λειτουργίας ενός επεξεργαστή πολλαπλών κύκλων.

Αρχιτεκτονική Σύνθετων Εντολών

Καλείστε να υλοποιήσετε έναν non-pipelined επεξεργαστή βασισμένο σε ένα extended υποσύνολο της αρχιτεκτονικής συνόλου εντολών C-CHARIS (Custom-CHAnia Risc Instruction Set, Έκδοση 2), ο οποίος **εκτός από τις εντολές που υλοποιούσε στα προηγούμενα εργαστήρια**, θα υλοποιεί και το παρακάτω σύνολο εντολών:

Custom εντολές:

- addi_MMX_byte
- poly2
- rfld
- rfst

Οι παραπάνω εντολές ανάλογα με την λειτουργία τους θα μπορούν να ανήκουν σε κάποιο από τα δύο format εντολών που σας δόθηκε στο 2^ο εργαστήριο.

Η διευθυνσιοδότηση της μνήμης (RAM) του νέου σας datapath θα γίνεται με διευθύνσεις bytes, ενώ οι λέξεις στη μνήμη θα πρέπει να είναι ευθυγραμμισμένες σε πολλαπλάσια των 4 Bytes.

Η κωδικοποίηση των εντολών καθώς και η λειτουργία τους θα γίνεται σύμφωνα με τον ακόλουθο πίνακα:

Opcode	FUNC	ΕΝΤΟΛΗ	ΠΡΑΞΗ	Περιγραφή
110001	-	MMX_addi_byte	$RF[rd] \leftarrow RF[rs](31 \text{ downto } 24) + \text{immed}(7 \text{ downto } 0) \& \\ RF[rs](23 \text{ downto } 16) + \text{immed}(7 \text{ downto } 0) \& \\ RF[rs](15 \text{ downto } 8) + \text{immed}(7 \text{ downto } 0) \& \\ RF[rs](7 \text{ downto } 0) + \text{immed}(7 \text{ downto } 0)$	“Βλέπει” τους καταχωρητές σαν 4 ανεξάρτητα bytes. Προσθέτει τα αντίστοιχα bytes του rs με το τελευταίο byte του immed και τα αποτελέσματα τα ενώνει δημιουργώντας μία λέξη 32 bits, η οποία αποθηκεύεται στον καταχωρητή rd.
100000	010000	poly2	$RF[rd] \leftarrow RF[rt] * RF[rt] * MEM[RF[rs]] + RF[rt] * \\ MEM[RF[rs]+4] + MEM[RF[rs]+8]$	Σε αυτή την πράξη, γίνεται ο υπολογισμός του πολυωνύμου 2 ^{ου} βαθμού $\text{poly2}(x) = ax^2 + bx + c$, όπου το x είναι ακέραιος αποθηκευμένος στον καταχωρητή rt, και τα a, b, c είναι επίσης ακέραιοι αποθηκευμένοι σε διαδοχικές θέσεις μνήμης. Η διεύθυνση του a περιγράφεται από τον καταχωρητή rs, ενώ οι τιμές των b και c, βρίσκονται στις επόμενες θέσεις μνήμης. Ο πολλαπλασιασμός μπορεί να υλοποιηθεί στην ALU μέσω του διαθέσιμου τελεστή της VHDL.
011100	-	rfl d	$\text{base_addr} = RF[rs] + \text{SignExtend}(\text{Imm}) \\ \text{for}(i=1; i<32; i++) \\ RF[i] \leftarrow MEM[\text{base_addr} + 4*i]$	Φορτώνει τις τιμές όλων των καταχωρητών της RF από 31 διαδοχικές θέσεις μνήμης.
011110	-	rfst	$\text{base_addr} = RF[rs] + \text{SignExtend}(\text{Imm}) \\ \text{for}(i=1; i<32; i++) \\ MEM[\text{base_addr} + 4*i] \leftarrow RF[i]$	Αντίθετη της rfl d, αποθηκεύει τις τιμές όλων των καταχωρητών της RF σε 31 διαδοχικές θέσεις μνήμης.

Διεξαγωγή

A. Μελετήστε την κωδικοποίηση των εξειδικευμένων εντολών

Μελετήστε την κωδικοποίηση των εντολών και ορίστε για κάθε μία από αυτές το format στο οποίο ανήκει.

B. Επεκτείνετε την λειτουργικότητα του CHARIS

B1. Υλοποίηση του datapath του επεξεργαστή ECHARIS

Μελετήστε το datapath που υλοποιήσατε στο προηγούμενο εργαστήριο. Βρείτε και υλοποιήστε τις αλλαγές που απαιτούνται στο datapath, πχ. πολυπλεκτών, καταχωρητών, μετρητών, για την υλοποίηση των παραπάνω εντολών αλλά και όλων των εντολών του προηγούμενου εργαστηρίου.

ΠΡΟΣΟΧΗ!!! Για όλες τις εντολές θα χρησιμοποιήσετε τα βασικά modules που υλοποιήσατε στα προηγούμενα εργαστήρια.

B2. Μελέτη και υλοποίηση του control του επεξεργαστή ECHARIS

Παρατηρήστε ξανά όλο το Control που κατασκευάσατε από το προηγούμενο εργαστήριο και κάντε τις αλλαγές-προσθήκες που χρειάζονται για να ελέγξετε τη ροή εκτέλεσης της κάθε νέας εντολής.

ΠΡΟΣΟΧΗ!!! Μετά τις αλλαγές στο datapath του ECHARIS ίσως χρειαστεί να γίνουν ξανά αλλαγές στο control του επεξεργαστή σας για την σωστή υλοποίηση των εντολών του προηγούμενου εργαστηρίου.

Τέλος, ενώστε το datapath του ECHARIS με το control που υλοποιήσατε και προσομοιώστε την σωστή λειτουργία όλων των εντολών.

Εκτέλεση

1. **ΠΡΟΣΟΧΗ!!!! Πριν από οποιαδήποτε αλλαγή στον κώδικα του προηγούμενου εργαστηρίου κρατήστε ένα backup του κώδικα του Εργαστηρίου 3 διότι θα τον χρειαστούμε σε επόμενο εργαστήριο.**
2. Για τον έλεγχο της ορθής λειτουργίας του συγκεκριμένου εργαστηρίου θα πρέπει να λειτουργούν ορθά τουλάχιστον οι εντολές li, lui και sw από το προηγούμενο εργαστήριο.
3. Αλλάξτε τον κώδικα VHDL που υλοποιεί το datapath του επεξεργαστή CHARIS ώστε να μπορεί να υλοποιεί τις νέες εντολές.
4. Προσθέστε ό,τι επιπλέον λογική χρειαστείτε στο datapath (καταχωρητές, μετρητές, πολυπλέκτες κτλ.).
5. Αλλάξτε τον κώδικα VHDL που υλοποιεί την μονάδα ελέγχου του επεξεργαστή CHARIS σύμφωνα με τις νέες ανάγκες και ονομάστε το αρχείο σας: Extended_control.vhd

6. Συνδέστε το datapath με το CONTROL ώστε να υλοποιήσετε την πλήρη λειτουργία ενός επεξεργαστή πολλαπλών κύκλων. Ονομάστε το αρχείο σας: ECHARIS.vhd
7. Επιβεβαιώστε την ορθή λειτουργία του επεξεργαστή δίνοντας πραγματικές εντολές σαν είσοδο και ελέγχοντας τις τιμές των σημάτων εξόδου σε κάθε κατάσταση της FSM.
8. Προσομοιώστε και επιβεβαιώσετε την λειτουργία του επεξεργαστή εκτελώντας εντολές μια-μια (με το χέρι).
9. Προσομοιώστε και επιβεβαιώσετε την λειτουργία του επεξεργαστή **εκτελώντας το πρόγραμμα που σας δόθηκε και που χρησιμοποιεί το extended instruction set του CHARIS (test.data)** με το οποίο θα αρχικοποιήσετε την IMEM.

Παραδοτέα

1. Αρχεία κώδικα VHDL (πηγαίος) την ημέρα πριν το εργαστήριο.
2. Σχηματικό διάγραμμα της μηχανής πεπερασμένων καταστάσεων του control **σημειώνοντας τις αλλαγές** που κάνατε σε σχέση με το προηγούμενο εργαστήριο.
3. Σχηματικό διάγραμμα του datapath **σημειώνοντας τις αλλαγές** που υλοποιήσατε σε σχέση με το προηγούμενο εργαστήριο.
4. Κυματομορφές προσομοίωσης (καλύψτε όλες τις περιπτώσεις).
5. Σύντομη αναφορά στη διαδικασία σχεδίασης και υλοποίησης (μαζί με σχόλια και ενδεχόμενα προβλήματα που παρατηρήθηκαν για μελλοντική του βελτίωση του εργαστηρίου).

ΚΑΛΗ ΕΠΙΤΥΧΙΑ