

Αναφορά Εργαστηριακής άσκησης 2

Ομάδα LAB31239665

Καλογεράκης Στέφανος AM:2015030064

Πίσκοπος Διονύσης AM:2015030115

Σκοπός εργαστηριακής άσκησης

Σκοπός της δεύτερης εργαστηριακής άσκησης ήταν η υλοποίηση της βαθμίδας ανάκλησης εντολών (**IFSTAGE**), της βαθμίδας αποκωδικοποίησης εντολών (**DECSTAGE**), της βαθμίδας εκτέλεσης εντολών (**EXSTAGE**) αλλά και την βαθμίδα πρόσβασης μνήμης (**MEMSTAGE**) η σύνδεση των οποίων οδηγεί στο πλήρες datapath ενός απλού επεξεργαστή.

Προεργασία

Σαν παραδοτέα/προεργασία του εργαστηρίου ζητήθηκε τόσο ο πηγαίος κώδικας όλων των components που υλοποιήθηκαν όσο και οι κυματομορφές προσομοίωσης. Επιπρόσθετα, ζητήθηκε προαιρετικά η υλοποίηση ενός σχηματικού διαγράμματος του ολοκληρωμένου datapath με όλες τις συνδέσεις των επιμέρους συνδέσεις components το οποίο παραθέτουμε στο παράρτημα παρακάτω.

Περιγραφή

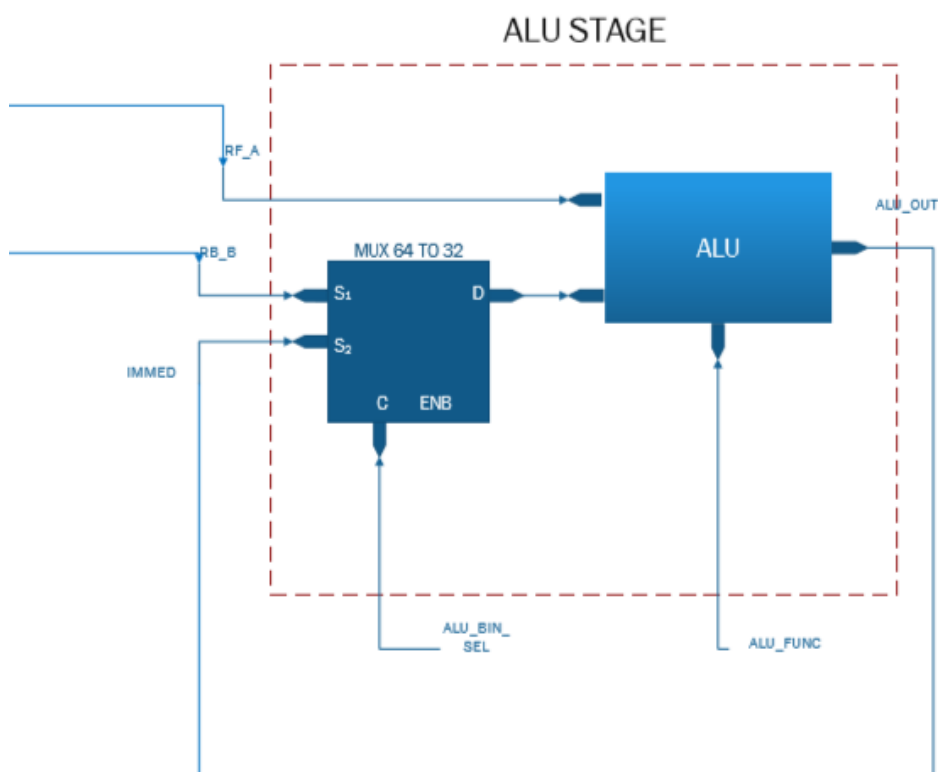
Η δεύτερη άσκηση χωρίστηκε σε τέσσερα μέρη υλοποίησης όλων των διαφορετικών component. Κατά σειρά, υλοποιήθηκαν η IFSTAGE, η DECSTAGE, η EXSTAGE, και MEMSTAGE. Στην συνέχεια, παρουσιάζονται αναλυτικά οι προηγούμενες υλοποιήσεις.

Βαθμίδα ανάκλησης εντολών (IFSTAGE)

Στο πρώτο κομμάτι της εργαστηριακής άσκησης υλοποιήσαμε την βαθμίδα ανάκλησης εντολών. Τα συστατικά της φαίνονται και στο παραπάνω σχήμα. Σημαντικό ρόλο παίζει το σήμα PC_sel το οποίο επιλέγει μέσω ενός καταχωρητή κατά πόσο ο PC θα αυξηθεί κατά 4 ή $PC + 4 + PC_Immed * 4$. Το component αυτό είναι ιδιαίτερα σημαντικό καθώς επιτρέπει την εκτέλεση όλων των εντολών τύπου branch. Στο τέλος συνδέσαμε την έξοδο PC με την Memory που είχε δοθεί από την εκκώνηση. Τα σήματα που χρησιμοποιήθηκαν από την Memory ήταν τα **inst_addr** και **inst_dout** χωρίς να απαιτηθεί κάποιο increment στην διεύθυνση.

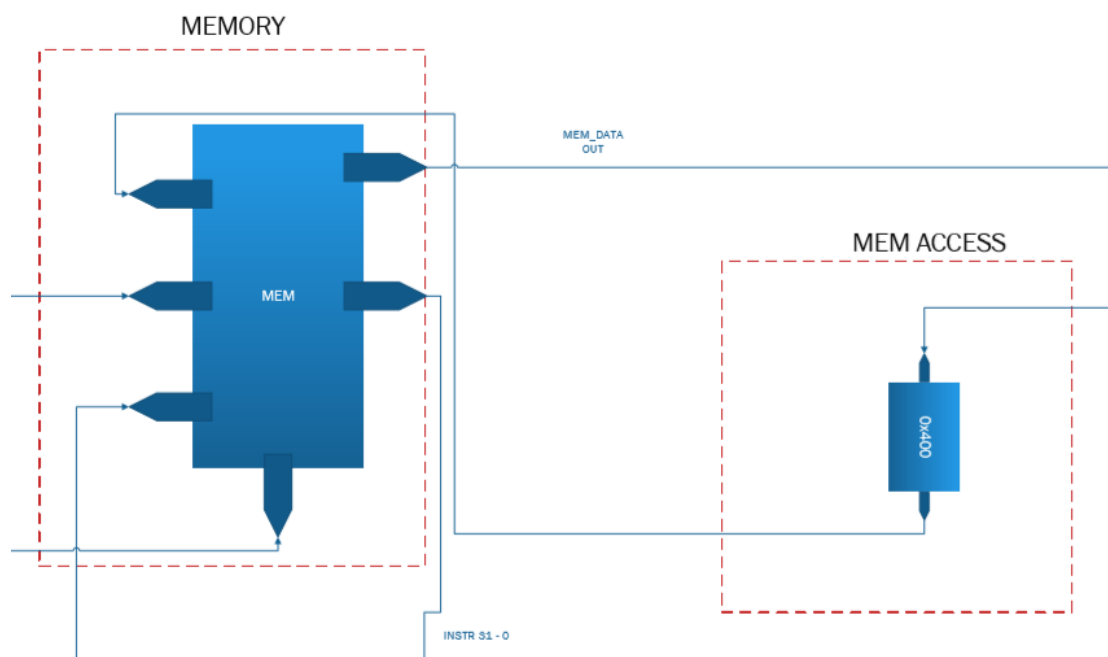
- 1 Multiplexer 10 to 5 με εισόδους τα bits 20-16 ή 15-11 με σκοπό να επιλέγουμε το φορμάτ της εντολής που δίνεται από το instruction set, ανάλογα την εντολή.
- 1 component που υλοποιεί την μετατροπή του 16-bit σταθερού αριθμού του instruction set (bits 15-0) σε 32 bit με επιλογή για sign extend, zero filling , εξαρτώμενο από το σήμα που λαμβάνει το κουτί από το control immedExt.
- 1 component shift 2 bits, που λαμβάνει την έξοδο του προηγούμενου, και ανάλογα από το σήμα που θα ληφθεί από το control, επιλέγει αν θα γίνει αριστερή ολίσθηση του 32 bit immediate κατά 2 ή όχι.
- 1 RF component, αντιγραφή του προηγούμενου εργαστηρίου οπού θα ενωθούν τα προαναφερθέν κομμάτια για να αποτελέσουν το decodestage.

Βαθμίδα εκτέλεσης εντολών (EXSTAGE)



Κατά την υλοποίηση της EXSTAGE χρησιμοποιήσαμε την ALU που είχαμε δημιουργήσει στην πρώτη εργαστηριακή άσκηση. Σαν πρώτο όρισμα, η ALU δέχεται σε κάθε περίπτωση τον καταχωρητή rs του Register File (RF[rs]) ενώ το δεύτερο όρισμα επιλέγεται από έναν πολυπλέκτη με εναλλακτικές είτε τους καταχωρητές rt, rd του Register File (RF[rt], (RF[rd])) είτε ένα σήμα Immediate. Ανάλογα την κάθε περίπτωση λοιπόν, η ALU εκτελεί όλες τις πράξεις που υλοποιήθηκαν απο το προηγούμενο εργαστήριο ανάμεσα σε δύο καταχωρητές ή ανάμεσα σε έναν καταχωρητή και ένα Immediate.

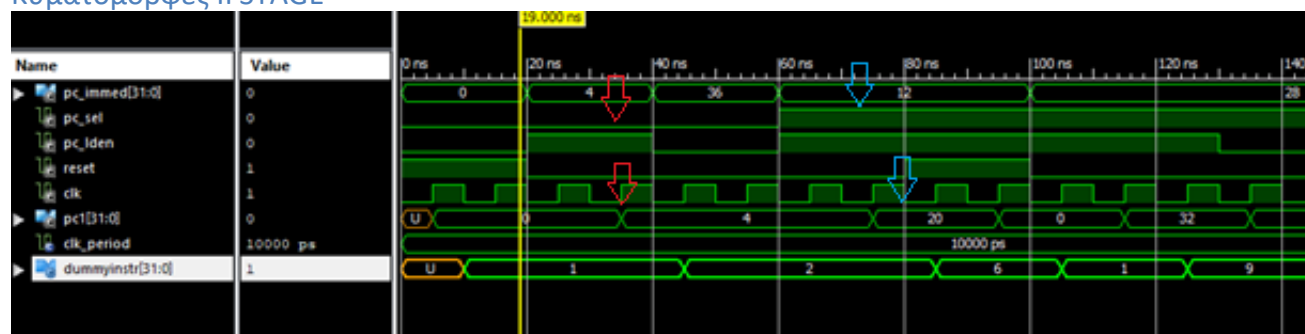
Βαθμίδα πρόσβασης μνήμης (MEMSTAGE)



Κατά το τελευταίο κομμάτι της εργαστηριακής άσκησης χρησιμοποιήθηκε και πάλι η Memory που μας είχε δοθεί από την εκφώνηση. Σε αυτήν την περίπτωση οι είσοδοι/έξοδοι της μνήμης που μας ενδιέφεραν ήταν **clk**, **data_we**, **data_addr**, **data_din** και **data_dout**. Η λειτουργία του συγκεκριμένου component είναι ιδιαίτερα απλή ένα σήμα MEM_WrEn να δίνει την δυνατότητα για γράψιμο στην μνήμη μόλις ενεργοποιείται. Το σήμα MEM_DataIn περιέχει τα δεδομένα του καταχωρητή για εγγραφή στην μνήμη ενώ το MEM_DataOut είναι η έξοδος της μνήμης σε περίπτωση φόρτωσης τα οποία γράφονται στη συνέχεια στην Register File. Τέλος, το σήμα ALU_MEM_Addr μας δίνει σε ποια διεύθυνση μνήμης θα γράψουμε αφού την αυξήσουμε κατά 1024(0x400) αφού έχουμε κοινή μνήμη

Κυματομορφές-Προσομοίωση

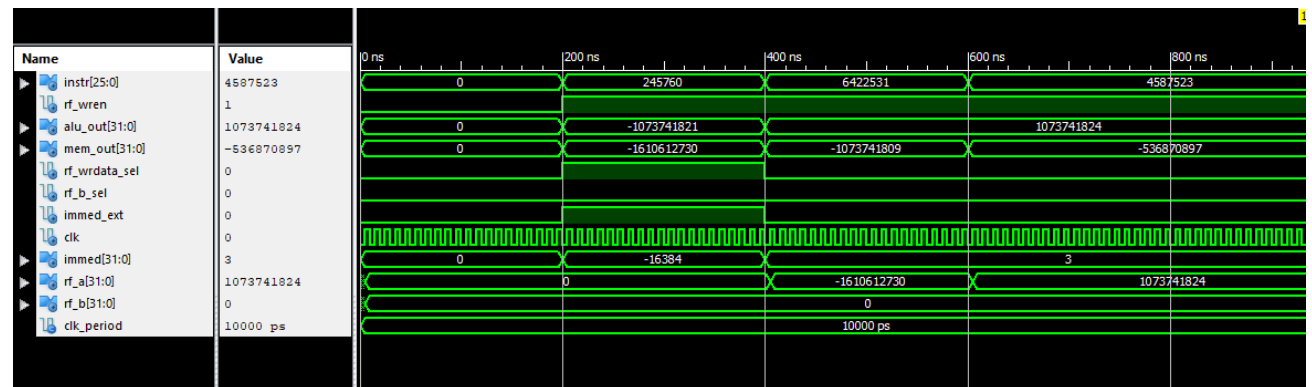
Κυματομορφές IFSTAGE



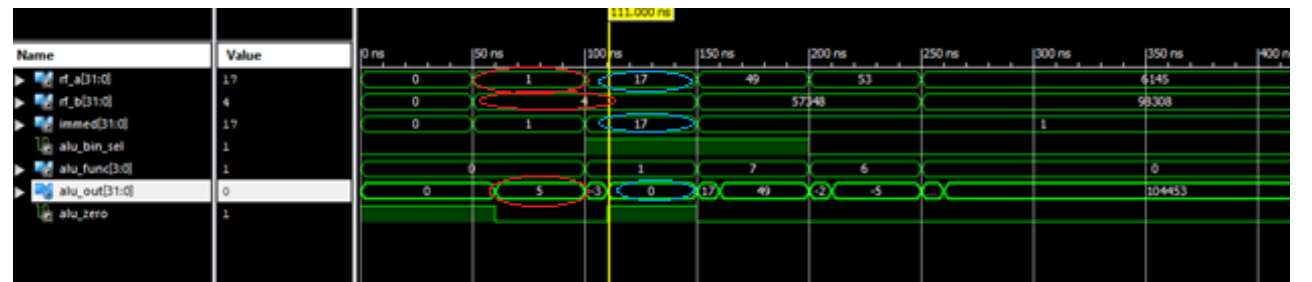
Παρατηρούμε ότι στις περιπτώσεις που ενεργοποιείται το PC_LdEn πραγματοποιείται είτε αύξηση στην τιμή του PC κατά 4(κόκκινο χρώμα) είτε κατά 4 + Immed επαληθεύοντας σε κάθε περίπτωση το αρχικό σχεδιάγραμμα. Το τελευταίο

σήμα (dummyInstr) είναι το Dout της μνήμης το οποίο επαληθεύει με την σειρά του την ορθότητα των κυματομορφών μας.

Κυματομορφές DECSTAGE

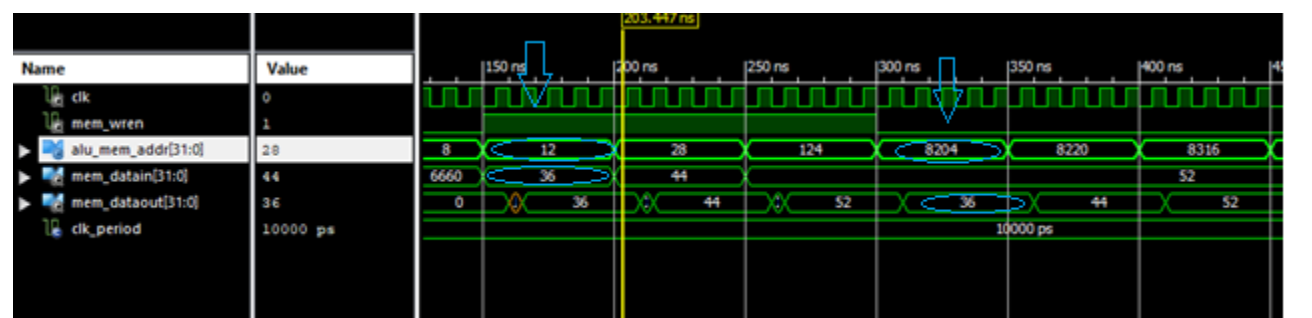


Κυματομορφές EXSTAGE



Χαρακτηριστικά παραθέτουμε με κόκκινο χρώμα μια πρόσθεση(κωδικός ALU 0) με RF_A και RF_B που ενεργοποιείται όταν το ALU_Bin_Sel είναι μηδέν, ενώ με μπλε χρώμα βλέπουμε την πράξη της αφαίρεσης μεταξύ RF_A και Immediate όταν ενεργοποιηθεί το ALU_Bin_Sel

Κυματομορφές MEMSTAGE



Βλέπουμε από την παραπάνω κυματομορφή ότι αρχικά στην περίπτωση που ενεργοποιείται το MEM_WeN πάμε σε συγκεκριμένες θέσεις μνήμης και κάνουμε ανάθεση τιμών, ενώ στην συνέχεια πηγαίνοντας στις αντίστοιχες θέσεις μνήμης με ένα increment 0x400 όπως όριζε η εκφώνηση διαβάζουμε τις ίδιες τιμές πράγμα που επιβεβαιώνει την ορθότητα του κώδικα μας.

Επισυνάπτεται παρακάτω το σχηματικό διάγραμμα συνδεσμολογίας του ολοκληρωμένου datapath όπως υλοποιήθηκε

