

4ή Εργαστηριακή Άσκηση

ΥΛΟΠΟΙΗΣΗ ΣΥΝΘΕΤΩΝ ΕΝΤΟΛΩΝ ΣΕ ΕΠΕΞΕΡΓΑΣΤΗ ΠΟΛΛΑΠΛΩΝ ΚΥΚΛΩΝ

Ομάδα LAB31235515

ΧΡΗΣΤΟΣ ΖΗΣΚΑΣ 2014030191

ΑΝΤΩΝΗΣ ΑΝΤΩΝΑΚΑΚΗΣ 2014030160

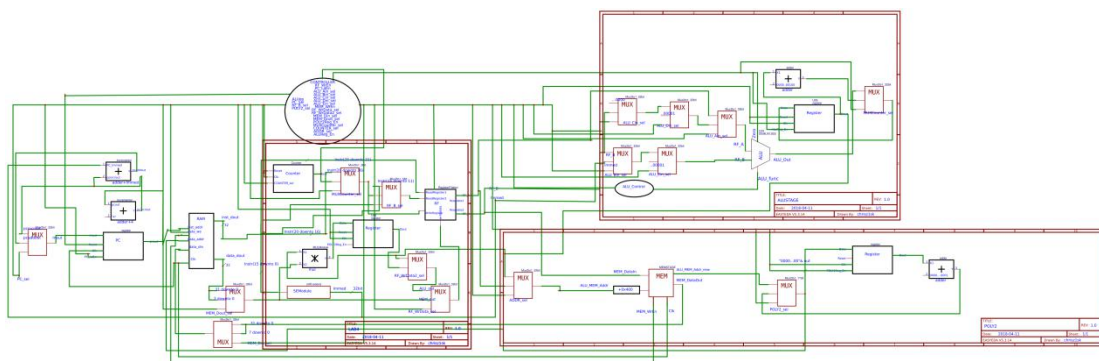
Σκοπός εργαστηριακής άσκησης

Η επέκταση της λειτουργικότητας του επεξεργαστή πολλαπλών κύκλων η οποία τελειοποιεί τον επεξεργαστή προσδίδοντας του περαιτέρω δυναμική. Εμπλουτίζεται εκ νέου η αρχιτεκτονική εντολών, το οποίο συνεπάγεται με επιπρόσθετη διαχείριση του control για τον σκοπό αυτό. Παράλληλα, η ενίσχυση της σχεδίασης για την υλοποίηση των νέων εντολών καθώς και για τη διατήρηση της λειτουργικότητας του επεξεργαστή όσον αφορά την εκτέλεση της ήδη υπάρχουσας αρχιτεκτονικής.

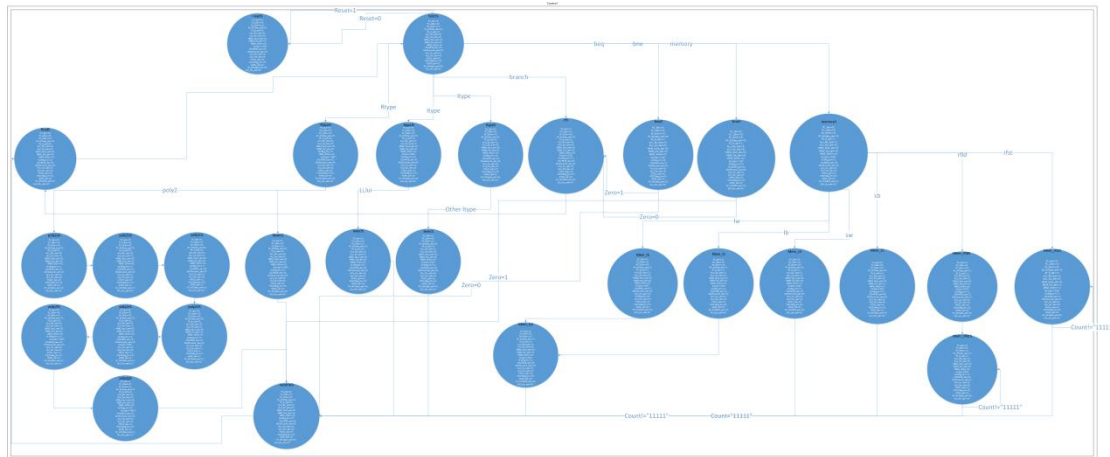
Προεργασία

Για την συμπλήρωση της σχεδίασης του επεξεργαστή πολλών κύκλων αναγκαία είναι η διόρθωση και εξέλιξη του control ώστε να διευθετούνται το νέο σύνολο εντολών καθώς ζητούνται νέα σήματα ελέγχου ώστε να κατευθύνουν το datapath. Φυσικά για το νέο control χρειάζεται και νεότερη εκδοχή της διαδρομής δεδομένων που συνάδει με διευθέτηση των νέων εντολών. Επιπλέον, υπάρχει μια μικρή τροποποίηση στην εσωτερική μονάδα ελέγχου ώστε να προστεθούν οι νέοι υπολογισμοί που προωθούνται από την alu. Το νέο datapath εντολών αφορά την είσοδο πολυπλεκτών είτε για την εισροή των δεδομένων στις κατάλληλες διευθύνσεις είτε ως ορθό εξαγόμενο καθώς και την είσοδο counter που συνεισφέρουν στην εκτέλεση της εντολής rfst, rfid. Χωρίς να εισάγεται δεύτερη μονάδα alu δημιουργήθηκε μονάδα πολλαπλασιασμού -MULModule που συντρέχει στην εκτέλεση της εντολής poly2. Οι επιπλέον προσθήκες συνοδεύονται από την παρουσία καταχωρητών. Η σχεδίαση της διαδρομής εντολών καθώς της fsm συστήνονται παρακάτω:

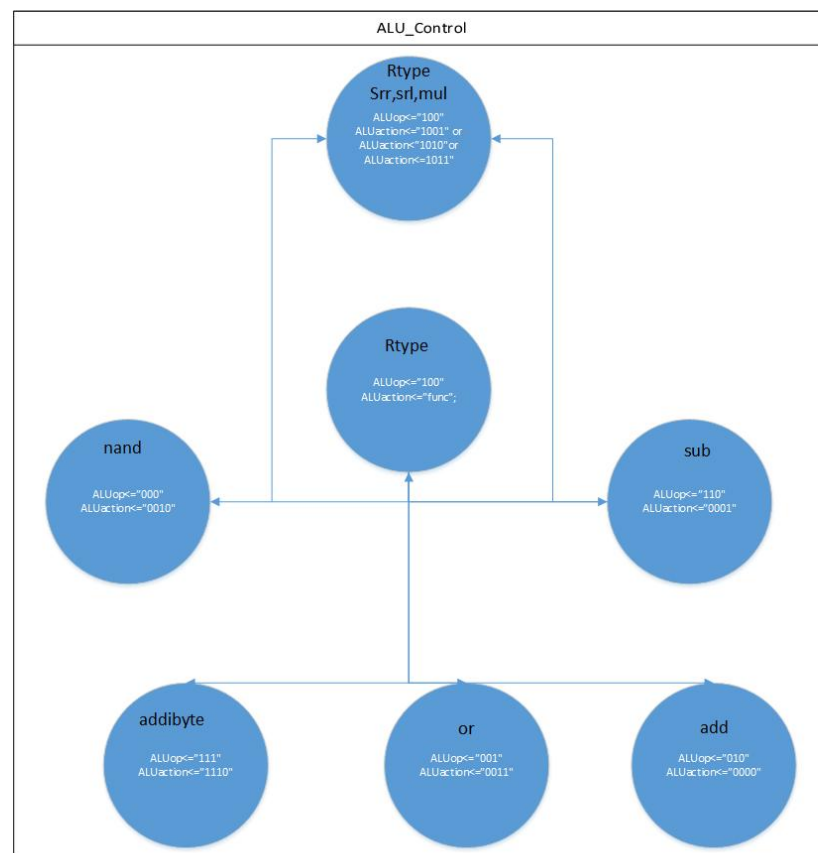
Βαθμίδα Datapath



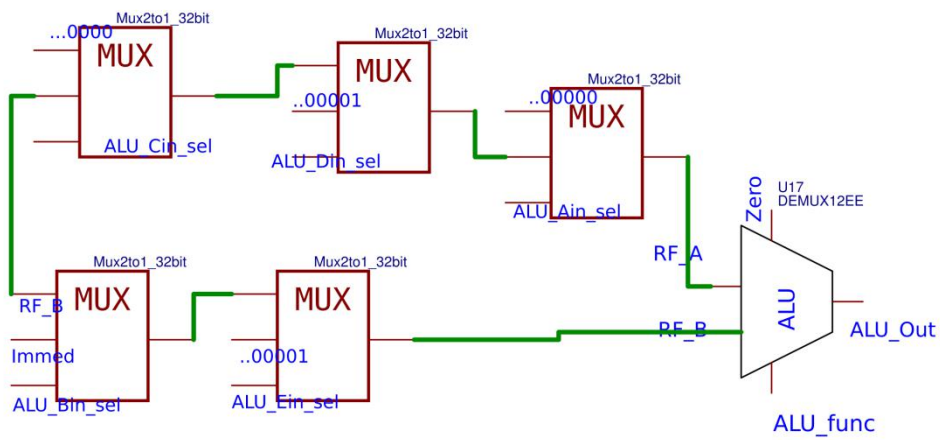
Μονάδα Ελέγχου (Control)



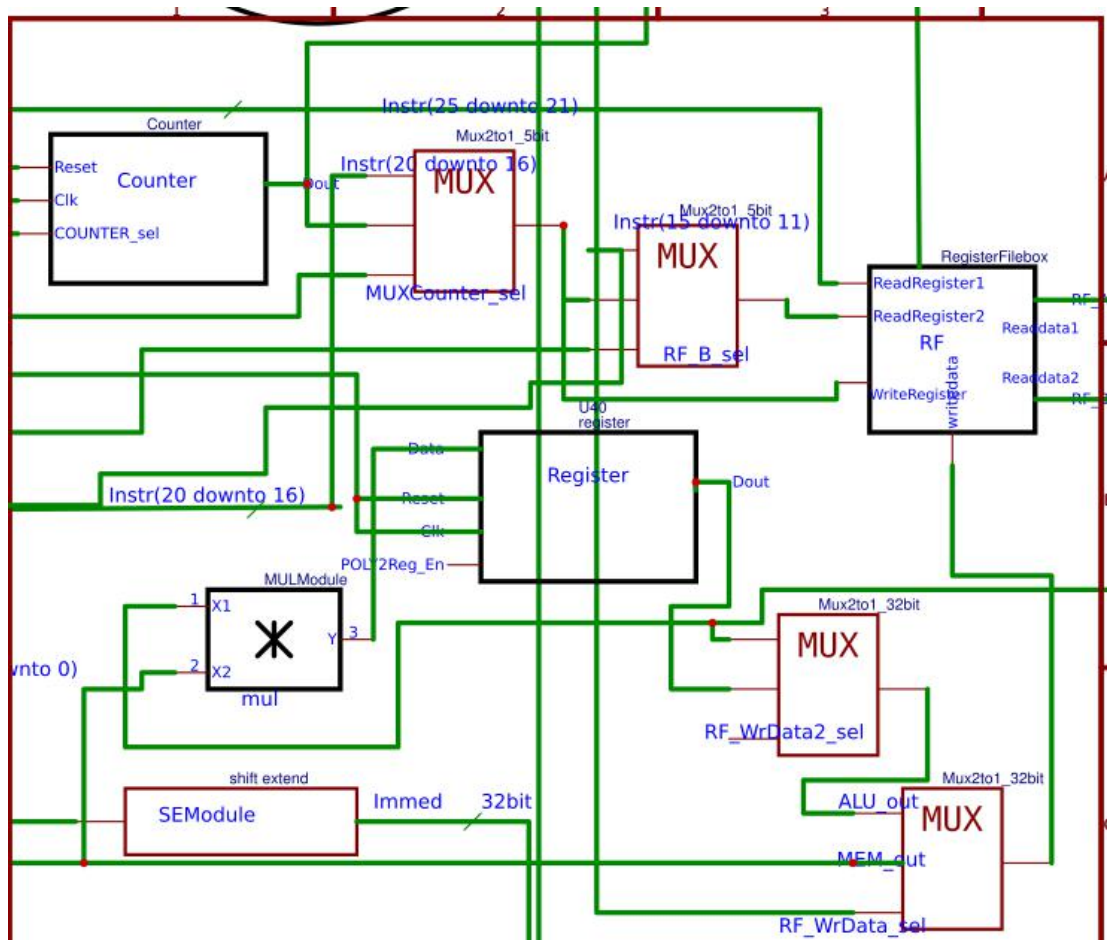
Μονάδα Ελέγχου ALU (ALU_Control)



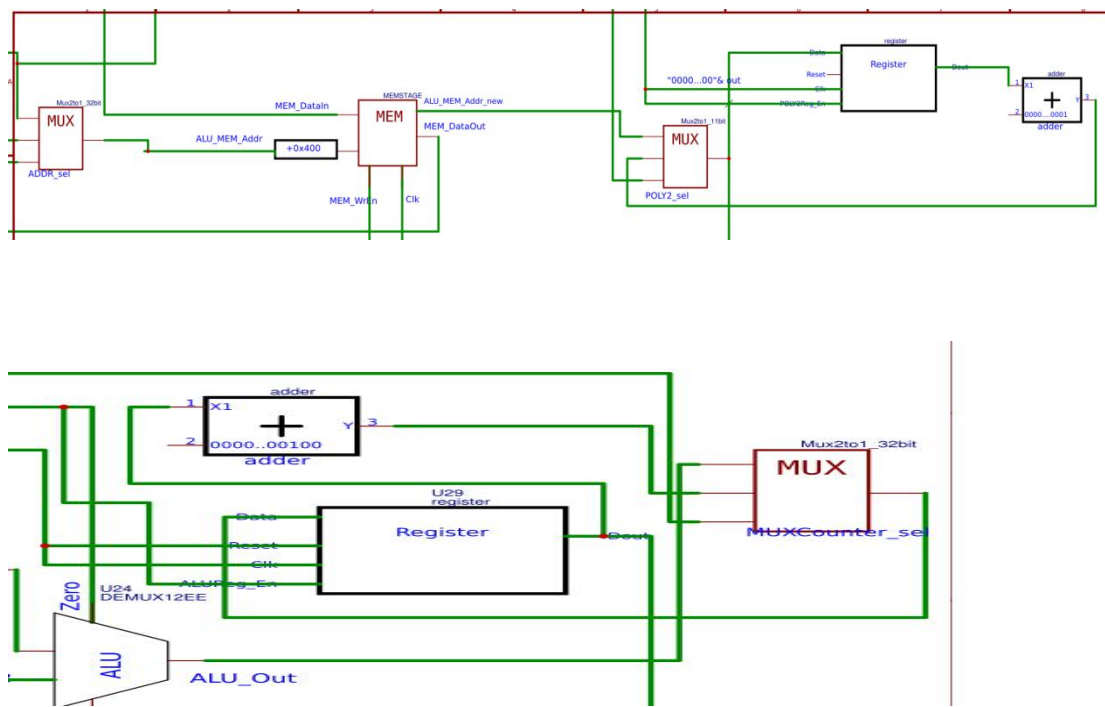
Μονάδα Υπολογισμού (ALUstage)



Μονάδα Αποκωδικοποίησης (DECstage)



Τροποποιήσεις



Περιγραφή

Η λειτουργικότητα του datapath και του control σχετικά με παλαιότερη αρχιτεκτονική δεν διαφέρει στην νέα έκδοση. Υπάρχουν μετατροπές εξαιτίας των νέων στοιχείων που εκτελούνται από τον επεξεργαστή.

Η βαθμίδα αποκωδικοποίησης (decstage) αναβαθμίζεται με την εισαγωγή ενός counter καθώς και επιπρόσθετων πολυπλεκτών που συνεισφέρουν στην εκτέλεση των εντολών rfld,rfst. Ο μετρητής δείχνει σε ποία θέση γίνεται επεξεργασία στο αρχείο καταχωρητών. Όσο δεν έχουν λάβει τιμή από την εντολή όλες οι θέσεις του rf ο counter αυξάνεται μέχρι να φτάσει στην τελική θέση. Στη θύρα εγγραφής συνδέεται πολυπλέκτης, ο οποίος δέχεται στην είσοδο του τον counter, ώστε κατά την εκτέλεση rfld να δέχονται επεξεργασία καθεμία από τις θέσεις στο αρχείο καταχωρητών. Σε άλλη περίπτωση ο πολυπλέκτης εξάγει την διεύθυνση ανάγνωση που αφορά το destination[rd]. Η έξοδος του πολυπλέκτη οδηγείται στον πολυπλέκτη που δρομολογεί την δεύτερη θύρα ανάγνωση. Το load του rf επεκτείνεται σε δυο κύκλους καθώς ακολουθείται όμοια λογική όπως και με την εντολή lb,lw-απαιτείται διάρκεια για να γίνει αναζήτηση στο ζητούμενο address. Στο δεύτερο κύκλο απενεργοποιείται το enable του μετρητή ώστε να μην γίνει υπέρβαση των αναμενόμενων θέσεων. Το στοιχείο επέκτασης εξελίσσεται με περισσότερη λειτουργικότητα εξαιτίας της εισαγωγής των νέων εντολών.

Η βαθμίδα υπολογισμών (alustage) δέχεται μετατροπές που συνδράμουν στην εκτέλεση της εντολής poly2. Τοποθετούνται πολυπλέκτες ώστε να εισέρχονται στην alu τα κατάλληλα δεδομένα. Σε πρώτη φάση πρέπει να βρεθεί το μονώνυμο δευτέρου βαθμού και ακολουθούν τα υπόλοιπα. Οπότε και οι πολυπλέκτες χρησιμοποιούνται για αυτό το σκοπό.

Για το $RF[rt] * RF[rt]$ αρκεί η τοποθέτηση στην είσοδο της alu πολυπλέκτες που να δέχονται ως είσοδο την δεύτερη τιμή εξόδου του rf ενώ θα έχουν ως δεύτερη είσοδο είτε 0 είτε 1 εφόσον είναι αναγκαίο το μονώνυμο πρώτης ή μηδενικής τάξης. Για το μονώνυμο μηδενικής τάξης τοποθετείται και πολυπλέκτης στην δεύτερη είσοδο της alu. Προσθέτοντας η μονάδα alu συνθέτει και την λειτουργία του πολλαπλασιασμού για τις απαιτήσεις της σχεδίασης. Αναμένεται λοιπόν ρύθμιση και στην alu μονάδα ελέγχου για το σκοπό αυτό.

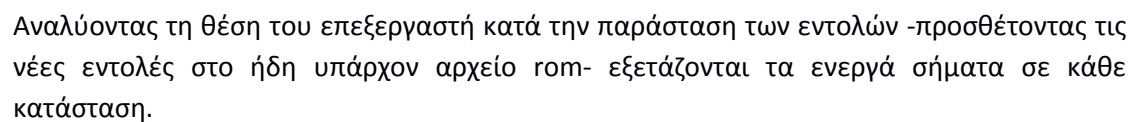
Το datapath με τη σειρά του , έχει δεχθεί διαφοροποιήσεις . Στην έξοδο της alu έχουν τοποθετηθεί στα στοιχεία ενός καταχωρητή που κρατάει το εξαγόμενο της, έναν αθροιστή ώστε να καταλήγουν σε επεξεργασία οι επιθυμητές διευθύνσεις κατά την εκτέλεση rfid,rfst. Ο αθροιστής παίρνει το αποτέλεσμα του καταχωρητή και το αυξάνει κατά 4- αποτελεί και διεύθυνση και ως εκ τούτου γίνεται πρόοδος στο επόμενο στοιχείο. Σε οποιαδήποτε άλλη εντολή το αποτέλεσμα περνάει απευθείας στον πολυπλέκτη και συνεχίζει τη διαδρομή του.

Όσον αφορά την διεύθυνση για τη μνήμη , έχει τοποθετηθεί πολυπλέκτης που έχει ως είσοδο την πρώτη τιμή εξαγωγής από το αρχείο καταχωρητών ώστε να γίνει εισροή στην μνήμη στην σωστή θέση $MEM[RF[rs]]$ -αφορά την εντολή poly2-. Ομοίως τοποθετούνται ένας αθροιστής ,καταχωρητής και πολυπλέκτης ώστε να υπάρχει μετάβαση στο επόμενο στοιχείο στη μνήμη από το δεδομένο rs . Με την εκσκαφή των αποτελεσμάτων από την μνήμη μεταχειρίζονται ένας πολλαπλασιαστής και ένας καταχωρητής. Πραγματοποιείται ο πολλαπλασιασμός του στοιχείο που αφορά το αρχείο καταχωρητών με το αποτέλεσμα που έχει αποκαλυφθεί από τη μνήμη . Το αποτέλεσμα παραμένει σε καταχωρητή . Με την νέα εισαγωγή δεδομένων ο καταχωρητής διατηρώντας την τιμή του ανανεώνεται προσθέτοντας την νέα τιμή προς καταχώρηση.Για αυτό το λόγο το enable του καταχωρητή παραμένει ενεργό μόνο για την στιγμή που καταχώρησης αλλιώς θα υπήρχε αλλοίωση στα δεδομένα. Οπότε εδώ παραμένουν τα αθροίσματα των μονωνύμων και το τελικό αποτέλεσμα αποθηκεύεται στο αρχείο καταχωρητών. Σε αυτό το σημείο ο καταχωρητής συνδέεται με πολυπλέκτη και η έξοδος του συνδέεται με τον πολυπλέκτη εγγραφής της rf.

Το control εμπλουτίζεται με περισσότερες καταστάσεις ώστε να ανταποκριθούν τα σήματα στις εντολές του συστήματος. Η εντολή MMX_addi_byte επηρεάζεται μόνο από το αντικείμενο επέκτασης καθώς δεν υπήρχαν διαφοροποιήσεις λόγω αυτής της εντολής στο datapath. Το poly2 εκτελείται σε 3 φάσεις των 2 κύκλων οπότε συνολικά 6 κύκλους για την αναζήτηση και την καταχώρηση των ορθών αποτελεσμάτων και στον 7 κύκλο εκπληρώνεται η εγγραφή.Το rfst εκτελείται όσον η τιμή του μετρητή δεν έχει φτάσει στο 11111. Οπότε 31 επαναλήψεις . Σε κάθε επανάληψη γράφεται ένα νέο εξαγόμενο στη ram.Το rfid εκτελείται με την ίδια λογική μόνο που χρειάζεται και ένας κύκλος ώστε να αναζητηθούν τα στοιχεία από την μνήμη και με κατάλληλη προσπέλαση να βρεθούν οι 31 διαδοχικές θέσεις . Σε εκείνο το κύκλο ο counter απενεργοποιείται και η ενέργεια τίθεται προς καταχώρηση στην rf.

Ουσιαστικά τα νέα σήματα που συμμετέχουν στη σχεδίαση πράττουν μια και μόνο λειτουργία για την επιτυχία του κυκλώματος αλλά αποτελούν το ίδιο σημαντικά με τα υπόλοιπα.

Παρουσιάζονται οι κυματομορφές του επεξεργαστή σύνθετων εντολών πολλών κύκλων.

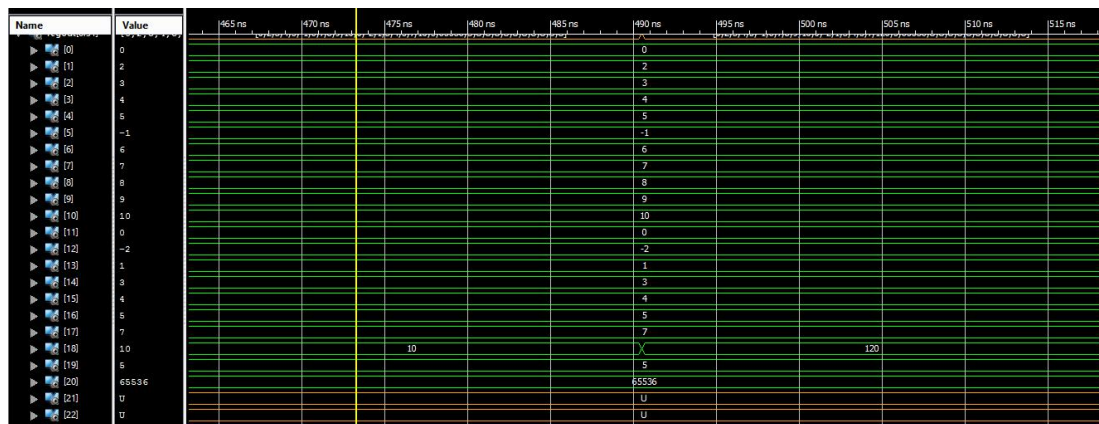


Για τον υπολογισμό $RF[rt]*RF[rt]$ ενεργοποιείται το alu_cin_sel & alu_aio_sel ώστε η έξοδος δεύτερη έξοδος της rf να περάσει στην πρώτη και την δεύτερη είσοδο στην alu.

Έπειτα εκτελείται ο πολλαπλασιασμός τους και αποθηκεύεται στον καταχωρητή . Αφού βρεθεί η σωστή διεύθυνση της ram γίνεται ο πολλαπλασιασμός $RF[rt] * RF[rt] * MEM[RF[rs]]$ και το αποτέλεσμα αποθηκεύεται σε νέο καταχωρητή-σηκώνεται το σήμα poly2regen. Για να βρεθεί η θέση $RF[rs]$ στη ram εγκαθίσταται πολυπλέκτης με εισόδους την πρώτη έξοδο της rf και την έξοδο του καταχωρητή στην έξοδο της alu. Σε δεύτερο στάδιο η εντολή πρέπει να εκτελέσει $RF[rt] * 1$ οπότε σηκώνεται το alu_din_sel & alu_ain_sel και έπειτα προχωρώντας στην επόμενη θέση στη $MEM[RF[rs] + 4]$ γίνεται η πράξη $RF[rt] * MEM[RF[rs] + 4]$ και ακολούθως αποθηκεύεται στο καταχωρητή. Στο τρίτο στάδιο διαφοροποιείται η συνάρτηση της alu σε πρόσθεση οπότε οι είσοδοι της alu γίνονται μηδέν και ένα αντίστοιχα-ενεργοποιείται το alu_ein_sel. Ο πολλαπλασιασμός με το $MEM[RF[rs] + 8]$ αποθηκεύεται στο καταχωρητή . Στον επόμενο κύκλο συνθέτει καταχώρηση στην rf.

rt = 00011(3)	rs = 01010(10)
$RF[rt] = 4$	$RF[rs] = 10 \rightarrow 10 / 4 = 2$
$RF[rt] * RF[rt] = 16$	$MEM[1024 + 2 = 1026] = 5$
$RF[rt] * 1 = 4$	$MEM[1027] = 10$
$0 + 1 = 1$	$MEM[1028] = 0$

Το αποτέλεσμα: $5 * 16 + 4 * 10 + 1 * 0 = 120$



Εξετάζοντας το στοιχείο πολλαπλασιασμού -θέσεις 1026 με καταχώρηση 5 και 1027 με 10 αποθηκεύεται το στοιχείο 120 στη θέση 18.

Name	Value
clk	0
reset	0
inst[31:0]	01110000011001010000000000000000
currents	mem_rfds
regout[0:31]	{0, 2, 3, 4, 5, 6, 7, 8, 9, 10, ..., 120, 5, 5}
ram[2047:0]	{0, 0, 0, 0, 0, ..., 0, 0, 0, 0, 0, ..., 0, 0, 0, 0, 0, ...}
aluaction[3:0]	0000
muxcounter_sel	1
counter_sel	0
rf_wrdstsel	0
alu_cn_sel	0
alu_ein_sel	0
alu_din_sel	0
poly2_sel	0
poly2reg_en	0
addr_sel	0
alu_bin_sel	1
alu_ain_sel	0
rf_b_sel	1
a[15:0]	386
b[15:0]	0
outm[31:0]	0
alureg_en	0
mem_wren	0
rf_wren	1

[illegible]

$$MEM[1028] = 0$$

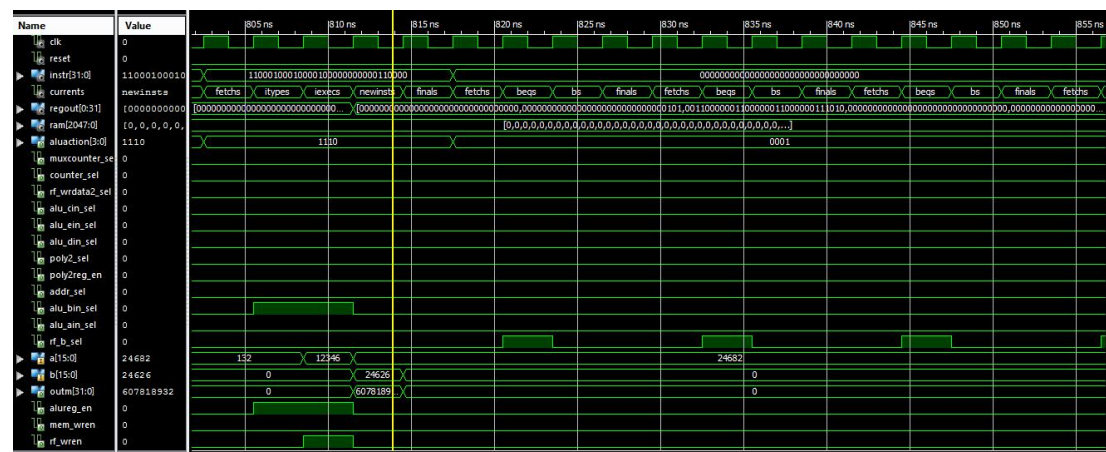
Εντολή mmx addi byte -- 110001000100001000000000000000000000110000

Η εντολή δεν διαφέρει από τις εντολές Immediate . Η τροποποίηση της σχεδίασης για την εντολή αυτή καλύπτεται από τη μονάδα επέκτασης και το αποτέλεσμα τίθεται προς αποθήκευση. Στο αντικείμενο επέκτασης εξάγεται τμηματικά το τελευταίο byte του

immediat σε κάθε από τα τέσσερα bytes . Διαφέρει από τις εντολές li,lui - είναι ενεργοποιημένο το alu_bin_sel. Η alu καταφέρει με ξεχωριστό κωδικό να αποσπάσει το εξαγόμενο πρόσθεσης των εκάστοτε byte μεταξύ του immediat και της RF[rs]. Η εγγραφή γίνεται στον θέση 2 της RF και προς ανάγνωση τίθεται η ίδια θέση.

	RF[rs]	Immediat	Result
7 downto 0	00001010	00110000	00111010
15 downto 8	00000000	00110000	00110000
23 downto 16	00000000	00110000	00110000
31 downto 24	00000000	00110000	00110000

Οπότε : 00110000001100000011000000111010 που είναι και το επιθυμητό



Συμπεράσματα/Προβλήματα

Η εργαστηριακή άσκηση καταλήγει σε σπουδαία συμπεράσματα αφού ενοποιείται η σχεδίαση ενός πολύπλοκου επεξεργαστή πολλαπλών κύκλων με περίτεχνη λειτουργία. Το συμπέρασμα πάνω σε αυτού του είδους τη σχεδίαση συνοψίζεται στην πολυπλοκότητα των λειτουργιών που μπορεί να καταφέρει ένας επεξεργαστής και επομένως να τεθούν σε ισχύ σύμφωνα με νέες εντολές ,νέες λειτουργίες μέσα από την υπάρχουσα λογική .