

ΟΡΓΑΝΩΣΗ ΥΠΟΛΟΓΙΣΤΩΝ

Εργασία 1: Επεξεργαστής ενός κύκλου

Ομάδα εργασίας: LAB31243955

Θεόδωρος Κουτσαβδής: 2017030145

Ιωάννης Οικονομόπουλος: 2017030048

1η φάση εργασίας

A) Μονάδα αριθμητικών και λογικών πράξεων (ALU).

Στο πρώτο μέρος της 1ης φάσης εργασίας ζητήθηκε η δημιουργία μίας μονάδας αριθμητικών και λογικών πράξεων (**ALU**), η οποία πρόκειται για ένα συνδυαστικό κύκλωμα που δεν έχει ρολόι. Η ALU παίρνει ως είσοδο ένα 4-bit αριθμό, ο οποίος δείχνει τον κωδικό της πράξης που υλοποιείται στους τελεστέους A και B, οι οποίοι είναι 32-bit. Το αποτέλεσμα της πράξης αποθηκεύεται στην έξοδο **out1**, η οποία είναι επίσης 32-bit. Επιπλέον, όταν το αποτέλεσμα της πράξης είναι ίσο με το μηδέν ενεργοποιείται η έξοδος **Zero** (ZeroFlag, 1-bit). Επίσης, υπάρχουν οι έξοδοι **Cout** και **Ovf** (1-bit το καθ' ένα), τα οποία είναι ίσα με μηδέν σε όλες τις περιπτώσεις εκτός από την περίπτωση της πρόσθεσης και της αφαίρεσης. Και στις δύο περιπτώσεις το Cout γίνεται ίσο με '1', όταν το αποτέλεσμα της πράξης είναι μεγαλύτερο των 32-bit (ο συγκεκριμένος έλεγχος γίνεται με ένα βοηθητικό σήμα 33-bit). Το Ovf στην περίπτωση της πρόσθεσης γίνεται ίσο με ένα είτε όταν προσθέτουμε δύο θετικούς αριθμούς και το αποτέλεσμα είναι αρνητικό είτε όταν προσθέτουμε δύο αρνητικούς και το αποτέλεσμα είναι θετικό (έλεγχος 32ου bit). Τέλος, στην περίπτωση της αφαίρεσης το Ovf γίνεται ίσο με ένα είτε όταν αφαιρούμε από ένα θετικό, έναν αρνητικό αριθμό και το αποτέλεσμα είναι αρνητικός αριθμός, είτε όταν αφαιρούμε από έναν αρνητικό, έναν θετικό αριθμό και το αποτέλεσμα είναι θετικός αριθμός (έλεγχος 32ου bit).

B) Αρχείο καταχωρητών (Register File).

B1. Δημιουργία καταχωρητή.

Στο δεύτερο μέρος της 1ης φάσης εργασίας ζητήθηκε η δημιουργία ενός καταχωρητή 32-bit. Η είσοδος του καταχωρητή (**DataIn**) βγαίνει στην έξοδο (**DataOut**) μόνο όταν το σήμα **WriteEnable** είναι ίσο με '1' και έρχεται ο θετικός παλμός ρολογιού (συγκεκριμένα θα βγει 10 ns αργότερα από τον θετικό παλμό λόγω καθυστέρησης). Το σήμα **reset** αρχικοποιεί την έξοδο ίση με μηδέν πάλι σύμφωνα με τη θετική ακμή ρολογιού. Σύμφωνα με τα παραπάνω συμπεραίνεται ότι τόσο το reset όσο και το WriteEnable είναι σύγχρονα σήματα.

B2. Δημιουργία αρχείου καταχωρητών RF

Το αρχείο καταχωρητών αποτελείται από 3 τμήματα:

- 1) 32 καταχωρητές 32-bit.
- 2) Έναν αποκωδικοποιητή
- 3) δυο πολυπλέκτες 32 προς 1

Το αρχείο καταχωρητών του επεξεργαστή δέχεται εισόδους τα σήματα Awr(5 bits), Din τα οποία εκφράζουν το που θα γίνει η εγγραφή μέσα στο αρχείο και το τι θα εγγραφεί στον καταχωρητή ο οποίος είναι ο destination. Η επιλογή του καταχωρητή στον οποίο θα εγγραφεί το Din επιτυγχάνεται με έναν αποκωδικοποιητή 5-32 του οποίου κάθε μια από τις 32 εξόδους συνδέεται στο WrEn του κάθε καταχωρητή, αν και το Write enable του αρχείου καταχωρητών είναι ενεργό, τότε το Din περνάει στο Datain του καταχωρητή που επιλέχθηκε. Το αρχείο καταχωρητών έχει επίσης εισόδους άλλα 2 σήματα των 5 bit τα οποία όπως και το Awr αναπαριστούν διεύθυνση. Τα 2 σήματα αυτά επιλέγουν μέσω δύο πολυπλεκτών 1-32, τα Dataout (για την επίτευξη των 32 από 32 bit εισόδων του κάθε πολυπλέκτη δημιουργήθηκε package το οποίο όριζε τον τύπο array of std_logic_vector(31 downto 0) και εμπεριέχεται στα αρχεία του Register file) των καταχωρητών τα οποία θα βγούνε στις δύο εξόδους Dout1 και Dout2 του αρχείου καταχωρητών.

2η φάση εργασίας

Βαθμίδα ανάκλησης εντολών IFSTAGE

Η βαθμίδα ανάκλησης εντολών **IFSTAGE** αποτελείται από 2 τμήματα:

- 1) Την βαθμίδα **IF**.
- 2) Την μνήμη **RAM**.

Η βαθμίδα ανάκλησης εντολών **IF** αποτελείται από 4 επιμέρους τμήματα (των οποίων η λειτουργία τους έχει αναλυθεί παραπάνω):

- 1) Έναν καταχωρητή **PC** πλάτους 32-bit.
- 2) Έναν αθροιστή που υπολογίζει το **PC+4**.
- 3) Έναν αθροιστή που υπολογίζει το **PC+4 + SignExt(Immed)*4**.
- 4) Έναν πολυπλέκτη (**MUX**) 2 προς 1 που περνά στον PC ή την τιμή PC+4 ή την τιμή PC+4 + SignExt(Immed)*4.

Η λειτουργία της βαθμίδας IFSTAGE είναι να δίνεται σωστή τιμή στον καταχωρητή PC ,έτσι ώστε να διαβάζονται σωστά οι εντολές που έχουν αρχικοποιηθεί στη μνήμη (αρχείο rom.data). Για το λόγο αυτό είναι απαραίτητη η σύνδεση της βαθμίδας IF και της μνήμης RAM. Συγκεκριμένα, η έξοδος του PC (με την ενεργοποίηση του **PC_LdEn**) μπαίνει ως είσοδος τόσο στην μνήμη όσο και στον PC+4 αθροιστή. Η μνήμη “διαβάζει” από το 0 έως το 1024 (text segment) και έτσι εξέρχεται το instruction, όπου είναι η εντολή που διαβάστηκε από το rom.data αρχείο. Τα 16 τελευταία bits του instruction μπαίνουν ως είσοδος στον $PC+4 + \text{SignExt(Immed)} * 4$ αθροιστή μαζί με το αποτέλεσμα του αθροιστή PC+4. Έτσι ο πολυπλέκτης με το σήμα **PC_sel** επιλέγει αν ο PC θα προχωρήσει +4 ή αν θα προχωρήσει σύμφωνα με το Immed αν υπάρχει εντολή διακλάδωσης (b, beq , bne) .

Βαθμίδα αποκωδικοποίησης εντολών DECODE

Για την βαθμίδα DECODE χρειάστηκε το port map του αρχείου καταχωρητών διότι έχει σαν εισόδους κομμάτια της εντολής η οποία αποκωδικοποιείται, επίσης αποτελείται από δύο ακόμα πολεπλέκτες 2 προς 1. Ο ένας ανάλογα με τύπου εντολή πραγματοποιούμε επιλέγει αν θα διαβαστεί ο rd ή ο rt. Ο άλλος επιλέγει εάν θα μπει στο DataIn του αρχείου καταχωρητών το αποτέλεσμα της ALU ή η έξοδος της μνήμης σε περίπτωση που έχουμε εντολή load. Τα 16 least significant bits της εντολής που αποκωδικοποιείται βάσει του CHARIS θα πάθουν ή sign extend ή zerofill ή sign extend και ολίσθηση ή zero fill και ολίσθηση. Ανάλογα με 2-bits τα οποία επιλέγονται αυθαίρετα, ελέγχονται αυτές τις 4 περιπτώσεις και το Immed της εντολής κάθε φορά παθαίνει το ανάλογο για να χρησιμοποιηθεί ως είσοδος σε επόμενες βαθμίδες.

Βαθμίδα εκτέλεσης εντολών EXSTAGE

Η βαθμίδα εκτέλεσης εντολών EXSTAGE αποτελείται από δύο τμήματα:

A) Την ALU.

B) Έναν πολυπλέκτη που επιλέγει ποιος είναι ο δεύτερος τελεστής της ALU.

Η πρώτη είσοδος της ALU είναι πάντα η είσοδος **RF_A** (ο καταχωρητής rs) , ενώ η δεύτερη είσοδος επιλέγεται ανάμεσα **RF_B** (ο καταχωρητής rt ή rd) και το **Immed** , με τον πολυπλέκτη που έχει ως είσοδο το **ALU_Bin_Sel**. Το RF_A και το RF_B είναι οι έξοδοι που έρχονται από το RegisteFile. Με τον τρόπο αυτό και σύμφωνα με το κατάλληλο **ALU_func** παίρνουμε το αποτέλεσμα της πράξης **ALU_out**.

Βαθμίδα πρόσβασης στην Μνήμη (MEM)

Η βαθμίδα πρόσβασης στη μνήμη αποτελείται από:

A) Τη βαθμίδα εκτέλεσης εντολών EXSTAGE.

B) Τη μνήμη RAM.

Γ) Τη βαθμίδα MEMSTAGE.

Η βαθμίδα **EXSTAGE** αναλύθηκε παραπάνω όπως επίσης και η RAM, μόνο που στη συγκεκριμένη βαθμίδα χρησιμοποιούμε το data segment μέρος της μνήμης, όπου γίνεται αποθήκευση και φόρτωση εντολών.

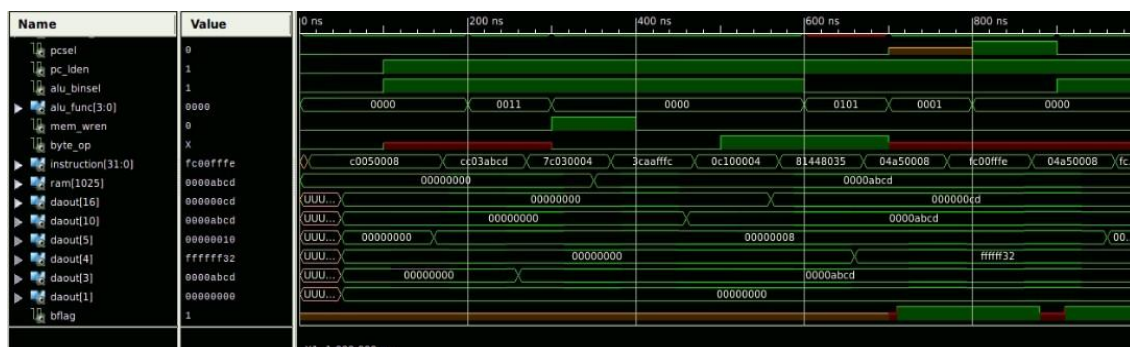
Στη βαθμίδα **MEMSTAGE** σύμφωνα με τα σήματα που δίνονται στις εισόδους **ByteOp** και **Mem_WrEn** γίνεται η επιλογή μεταξύ των εντολών lw, sw, lb, sb. Με τον τρόπο αυτό περνά σωστή διεύθυνση στη μνήμη (για εγγραφή ή ανάγνωση) από το σήμα της **ALU_MEM_Addr**. Τέλος, σύμφωνα με το αν υπάρχει εγγραφή ή όχι ανάβει το σήμα **MM_WrEn**.

Στη βαθμίδα **MEM**, γίνεται η κατάλληλη συνδεσμολογία ανάμεσα στις βαθμίδες EXSTAGE, MEMSTAGE. Αξιοσημείωτο, είναι το γεγονός ότι από το αποτέλεσμα της ALU (έξοδος EXSTAGE) κρατάμε τα ψηφία 12 μέχρι 2 και τότε τους προσθέτουμε 0x400 (+1024), ώστε να μπει σωστή διεύθυνση εγγραφής ή ανάγνωσης στη μνήμη.

3η φάση εργασίας

Μέρος Datapath

Στη συγκεκριμένη φάση της εργασίας ήταν απαραίτητη η συνδεσμολογία όλων των βαθμίδων που υλοποιήθηκαν και αναλύθηκαν στη δεύτερη φάση της εργασίας. Για την επιβεβαίωση της σωστής λειτουργίας του κυκλώματος φορτώθηκε το rom.data αρχείο που περιλαμβάνει το πρόγραμμα αναφοράς 1 και 2 (έχουν φορτωθεί και τα δύο ταυτόχρονα). Στην παρακάτω φωτογραφία φαίνεται το testbench του datapath, όπου φαίνονται και τα σήματα των καταχωρητών και της μνήμης.



Μέρος Control

Στο συγκεκριμένο μέρος της εργασίας, η μόνη είσοδος ,εκτός από το clock και το reset , είναι το instruction. Η λειτουργία του είναι η παραγωγή σωστών σημάτων ελέγχου για την κάθε εντολή ξεχωριστά, σύμφωνα με το με το opcode και το func της κάθε εντολής.

Μέρος PROC_SC

Το μέρος αυτό περιλαμβάνει την τελική λειτουργία του κυκλώματος, αφού ενώνονται τα μέρη **DataPath** και **Control**. Το συγκεκριμένο μέρος έχει απλά ως εισόδους το clock, το reset και το PC_LdEn. Στο παρακάτω testbench φαίνονται όλα τα σήματα που επιβεβαιώνουν τη σωστή λειτουργία του τελικού κυκλώματος. (dout--> καταχωρητές, ram(1025--> μνήμη).

