

Αναφορά Εργαστηρίου 2

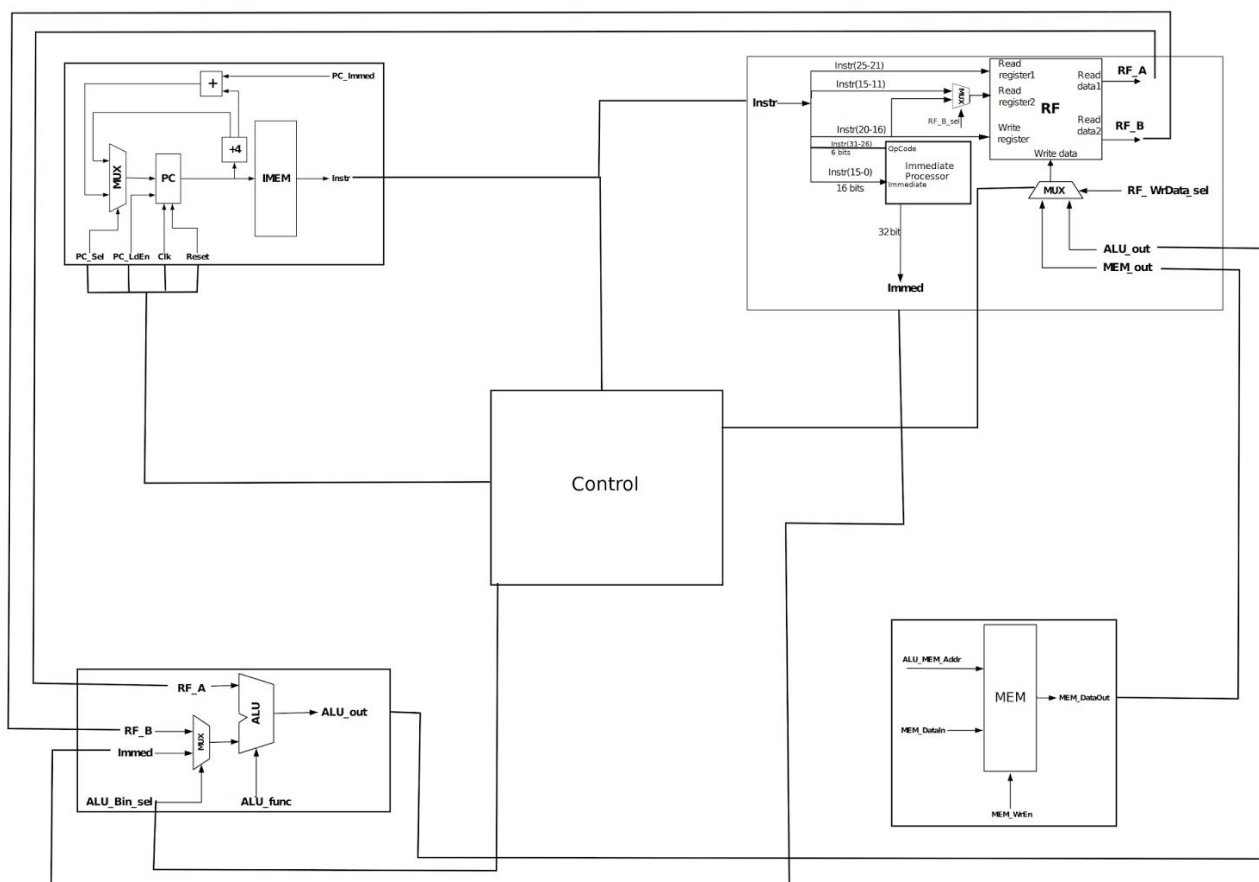
Οργάνωση Υπολογιστών

Ομάδα Εργασίας: Κριθαράκης Εμμανουήλ, Φωτάκης Τζανής

Κωδικός Ομάδας: LAB31231483

Προεργασία

Ως προεργασία του συγκεκριμένου εργαστηρίου είχε ζητηθεί όπως είναι φυσικό ο κώδικας των διαφόρων modules που περιγράφονται παρακάτω αλλά και ένα σχεδιάγραμμα που παρουσιάζει την συνδεσμολογία μεταξύ των διαφόρων ζητούμενων σταδίων που κληθήκαμε να αναπτύξουμε. Παρόλα αυτά δεν ζητήθηκε η υλοποίηση του σε επίπεδο κώδικα. Παρακάτω παραθέτεται το σχετικό σχεδιάγραμμα.



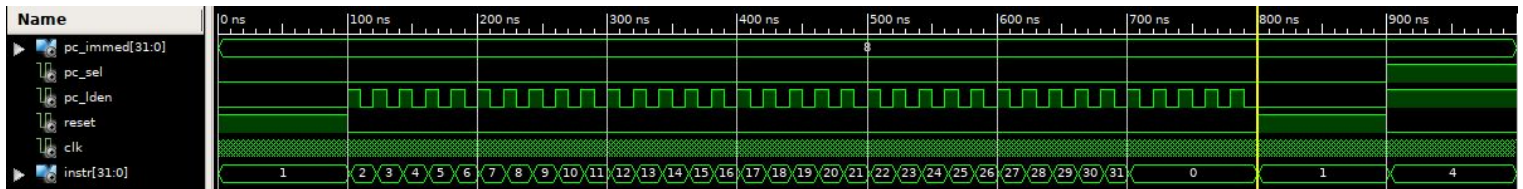
Περιγραφή της Άσκησης

Το εργαστήριο αυτό χωρίζεται σε τέσσερα σκέλη όπου το καθένα υλοποιεί από ένα στάδιο του επεξεργαστή με το δοσμένο instruction set CHARIS-4.

IFSTAGE

Όπως ζητήθηκε δημιουργήσαμε ένα module με όνομα IF_MEM σύμφωνα με τον δοσμένο κώδικα στην εκφώνηση της άσκησης. Για την υλοποίηση του IFSTAGE έγινε η χρήση ενός instance του IF_MEM όπως επίσης και ενός 32-bit ακμοπυροδότητου register, ενός BusMux2 δηλαδή ενός πολυπλέκτη 2 εισόδων των 32 bit και μίας εξόδου 32 bit και τέλος δύο αθροιστών, ενός που αναλαμβάνει την πρόσθεση του PC με το 4 και ενός που αναλαμβάνει την πρόσθεση του PC με το Immediate για την υλοποίηση εντολών branch. Σημαντικό να αναφερθεί είναι το γεγονός ότι κατά την σύνδεση του PC register στην είσοδο addr του IF_MEM χρησιμοποιούνται τα bit εξόδου του register από το 2 έως το 11 για να γίνεται η επιλογή της εντολής εξόδου του IF_MEM κατά εντολή (4 byte) και όχι ανά byte (1 byte). Αυτό γίνεται γιατί η μνήμη μας είναι word-addressable και όχι byte-addressable.

Προσομοίωση

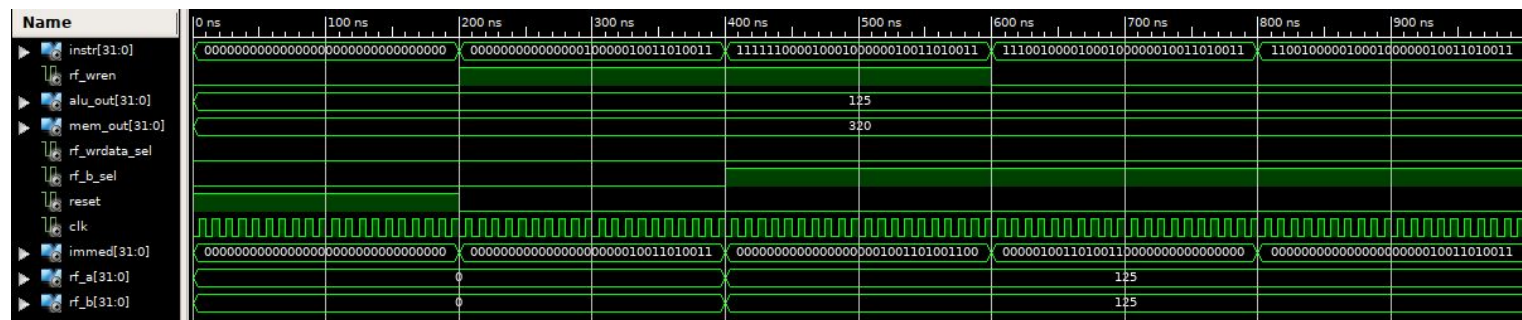


- 0-100: Reset του κυκλώματος
- 100-800: Διάβασμα μέχρι και την 36^η εντολή (από 32^η έως 36^η οι εντολές περιέχουν την τιμή 0).
- 800-900: Reset του κυκλώματος
- 900-1000: Εκτέλεση εντολής διακλάδωσης κατά 8 byte, δηλαδή συνολικά κατά 12 byte (8 byte λόγω του immediate και 4 byte λόγω της σταθερής αύξησης του pc)

DECSTAGE

Για την υλοποίηση του συγκεκριμένου σταδίου έγινε χρήση του Register File module που δημιουργήσαμε στο προηγούμενο εργαστήριο όπως επίσης και ενός πολυπλέκτη 2 εισόδων των 5 bit και μίας εξόδου των 5 bit για την επιλογή της δευτέρας διεύθυνσης ανάγνωσης από το RF. Ακόμα, ένας πολυπλέκτης των 2 εισόδων των 32 bit και μίας εξόδου 32 bit για την επιλογή των δεδομένων εγγραφής στο RF μεταξύ των ALU και MEM. Τέλος, έγινε η χρήση ενός επιπλέον module που δημιουργήσαμε, το ImmediateProcessor, το οποίο όπως το λέει και το όνομά του επεξεργάζεται το Immediate με βάση το instruction που έχει δοθεί στο DECSTAGE.

Προσομοίωση

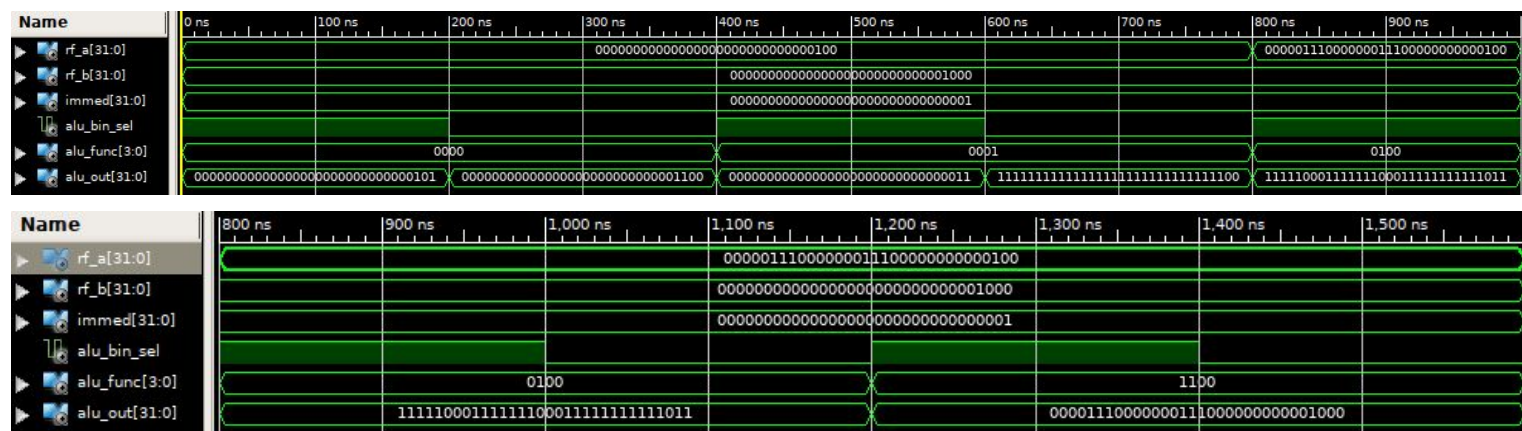


- 0-200: Reset του κυκλώματος.
- 200-400: Εγγραφή του πρώτου register και sign extension του immediate.
- 400-600: Εγγραφή του δεύτερου register και sign extension μαζί με shift left κατά 2 του immediate.
- 600-800: Διάβασμα του πρώτου και του δεύτερου register και zero fill μαζί με shift left κατά 16 του immediate.
- 800-1000: Zero fill του immediate.

ALUSTAGE

Το στάδιο αυτό αποτελείται από την ALU που αναπτύχθηκε στο προηγούμενο εργαστήριο όπως επίσης και από έναν πολυπλέκτη 2 εισόδων των 32 bit και μίας εξόδου των 32 bit ο οποίος κάνει την επιλογή των δεδομένων της δεύτερης εισόδου της ALU μεταξύ της δεύτερης εξόδου του RF και του Immediate.

Προσομοίωση



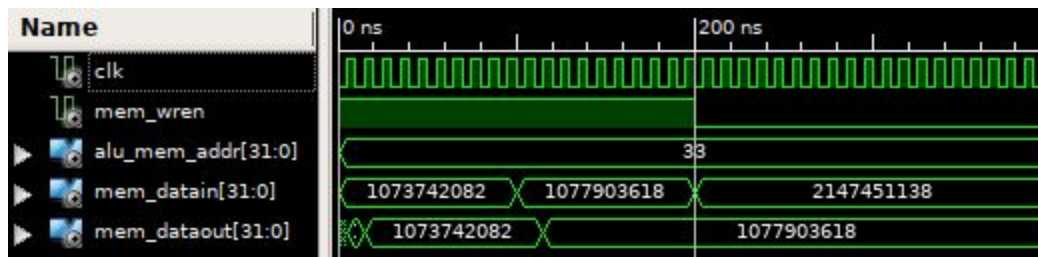
- 0-200: RF_A + Immediate
- 200-400: RF_A + RF_B

- 400-600: RF_A - Immediate
- 600-800: RF_A - RF_B
- 800-1200: Complimentary(RF_A)
- 1200-1600: Left_Cycle_Rotation(RF_A)

MEMSTAGE

Το συγκεκριμένο στάδιο αποτελείται από ένα Data_MEM module που ζητήθηκε να δημιουργηθεί σύμφωνα με την εκφώνηση.

Προσομοίωση



- 0-100: Δίνεται ως είσοδος η τιμή 1.073.742.082 και ενεργοποιείται η εγγραφή οπότε μετά από δύο κύκλους ρολογιού παρατηρείται στην έξοδο το ίδιο νούμερο.
- 100-200: Αντίστοιχα με το παραπάνω δίνεται είσοδος η τιμή 1.077.903.618 και κρατείται ενεργή η εγγραφή οπότε παρατηρείται στην έξοδο το ίδιον νούμερο.
- 200-400: Απενεργοποιείται η εγγραφή και παράλληλα αλλάζει η είσοδος παρόλα αυτά δεν αλλάζει η έξοδος.

Συμπεράσματα-Παρατηρήσεις

- Μία σημαντική παρατήρηση για την διεξαγωγή του εργαστηρίου (όπως αναφέρθηκε και την ώρα του εργαστηρίου) θα ήταν ότι τα specification του instruction set του πρώτου εργαστηρίου με το συγκεκριμένο διαφέρουν στην υλοποίηση της ALU στις εντολές shr, shl, sla. Πιθανολογούμε ότι η αλλαγή έγινε για να είναι πιο εύκολη η κωδικοποίηση των εντολών για τα σήματα του control unit.