

Αρχιτεκτονική Υπολογιστών

Ασκήσεις

Άσκηση 1.1: Απόδοση, ρολόι και CPI

- Θεωρήστε 3 επεξεργαστές P1, P2, P3 που εκτελούν το ίδιο σύνολο εντολών

| Processor | Clock frequency | CPI |
|-----------|-----------------|-----|
| P1 | 2 GHz | 1.5 |
| P2 | 1.5 GHz | 1.0 |
| P3 | 3 GHz | 2.5 |

- Ποιος επεξεργαστής έχει την καλύτερη απόδοση;
- Αν κάθε επεξεργαστής εκτελεί ένα πρόγραμμα σε 10 sec, βρείτε το πλήθος των κύκλων και το πλήθος των εντολών

Άσκηση 1.2: Απόδοση, ρολόι, κατηγορίες εντολών και CPI

- 2 υλοποιήσεις του ίδιου ISA: P1, P2
 - 4 κατηγορίες εντολών: A, B, C, D

| | Clock frequency | CPI A | CPI B | CPI C | CPI D |
|----|-----------------|-------|-------|-------|-------|
| P1 | 1.5 GHz | 1 | 2 | 3 | 4 |
| P2 | 2.0 GHz | 2 | 2 | 2 | 2 |

- Πρόγραμμα με 10^6 εντολές: 10% κατηγορία A, 20% B, 50% C και 20% D. Ποια υλοποίηση είναι ταχύτερη;
- Συνολικό CPI για κάθε υλοποίηση;
- Βρείτε τους κύκλους ρολογιού και το χρόνο εκτέλεσης του προγράμματος για τις 2 υλοποιήσεις

3

Άσκηση 1.3: Απόδοση, εντολές, CPI και χρόνος εκτέλεσης

- Πλήθος των εντολών ενός προγράμματος και CPI των εντολών

| Arithmetic | Store | Load | Branch |
|------------|-------|------|--------|
| 500 | 50 | 100 | 50 |
| 1 | 5 | 5 | 3 |

- Ποιος είναι ο χρόνος εκτέλεσης σε έναν επεξεργαστή των 2 GHz;
- Βρείτε το CPI του προγράμματος

4

Άσκηση 1.4: Μεταγλωττιστές και απόδοση

- Για το ίδιο πρόγραμμα χρησιμοποιούνται 2 μεταγλωττιστές

| Μεταγλωττιστής Α | | Μεταγλωττιστής Β | |
|-------------------|------------------|-------------------|------------------|
| Αριθμός εντολών | Χρόνος εκτέλεσης | Αριθμός εντολών | Χρόνος εκτέλεσης |
| 1.0×10^9 | 1.0 sec | 1.2×10^9 | 1.4 sec |

- Βρείτε το CPI για κάθε περίπτωση (θεωρήστε χρόνο κύκλου ρολογιού 1 ns)
- Ένας νέος μεταγλωττιστής Γ χρησιμοποιεί 600 εκατομμύρια εντολές και έχει CPI 1.1. Ποια είναι η επιτάχυνση που επιτυγχάνει έναντι των μεταγλωττιστών Α και Β;

5

Άσκηση 1.5: Απόδοση και νόμος Amdahl

- Θεωρήστε έναν επεξεργαστή και τους χρόνους CPU ενός προγράμματος

| FP instructions | Integer instructions | Load/ Store | Branch instructions | Total time |
|-----------------|----------------------|-------------|---------------------|------------|
| 40 s | 80 s | 50 s | 30 s | 200 s |

- Πόσο θα μειωθεί ο συνολικός χρόνος εάν ο χρόνος των εντολών FP μειωθεί κατά 20%;
- Για να μειωθεί ο συνολικός χρόνος κατά 20% πόσο πρέπει να μειωθεί ο χρόνος των εντολών ακεραίων;
- Μπορεί να μειωθεί ο συνολικός χρόνος κατά 20% εάν μειωθεί μόνο ο χρόνος των εντολών διακλάδωσης;

6

Άσκηση 2-1

- Υλοποιήστε σε συμβολική γλώσσα MIPS τον παρακάτω βρόχο:
 - base address A = \$a1, base address B = \$a2

```
int A[100], B[100];
for (i=1; i < 100; i++) {
    A[i] = A[i-1] + B[i];
}
```

```

li    $t0, 1           # Starting index of i
li    $t5, 100        # Loop bound
loop:
lw    $t1, 0($a1)     # Load A[i-1]
lw    $t2, 4($a2)     # Load B[i]
add   $t3, $t1, $t2   # A[i-1] + B[i]
sw    $t3, 4($a1)     # A[i] = A[i-1] + B[i]
addi  $a1, $a1, 4     # Go to i+1
addi  $a2, $a2, 4     # Go to i+1
addi  $t0, $t0, 1     # Increment index variable
bne   $t0, $t5, loop # Compare with Loop Bound
```

7

Άσκηση 2-2

- Έστω ότι ο MIPS δεν υλοποιούσε την εντολή αριστερής λογικής ολίσθησης. Γράψτε την ακολουθία εντολών assembly που θα υλοποιούσε μια ψευδοεντολή logical-shift-left (LSL)

- LSL \$s1, \$s2, n

```

li    $t1, n
addi  $s1, $s2, 0
loop:
add   $s1, $s1, $s1
addi  $t1, $t1, -1
bne   $t1, $0, loop
```

8

Άσκηση 2-3

- Έστω μια νέα εντολή MIPS `bcp` (block copy) που αντιγράφει ένα μπλοκ λέξεων από μια διεύθυνση σε μια άλλη.
 - address of the source block = `$t1`, address of the destination block = `$t2`, number of words = `$t3` (> 0)
- Γράψτε κώδικα MIPS που υλοποιεί την αντιγραφή μπλοκ χωρίς την εντολή `bcp`
 - Συγκρίνετε τον χρόνο εκτέλεσης των 2 υλοποιήσεων για την αντιγραφή ενός μπλοκ 100 λέξεων

```
loop:
    lw  $t4, 0($t1)
    sw  $t4, 0($t2)
    addi $t1, $t1, 4
    addi $t2, $t2, 4
    addi $t3, $t3, -1
    bne $t3, $zero, loop
```

9

Άσκηση 2-4

- Εάν γίνουν οι παρακάτω αλλαγές στη μορφή των εντολών του MIPS:
 - 8 καταχωρητές
 - Άμεσες σταθερές των 10-bit
- Ποιο το μέγεθος των πεδίων μιας εντολής R-format;
- Ποιο το μέγεθος των πεδίων μιας εντολής I-format;
- Σχολιάστε πώς οι παραπάνω αλλαγές θα μπορούσαν να μειώσουν ή να αυξήσουν το μέγεθος ενός προγράμματος του MIPS

10

Άσκηση 2-5

- Οι παρακάτω λειτουργίες δεν περιλαμβάνονται στο σύνολο εντολών του MIPS:
 - `andn $t1,$t2,$t3` `#$t2) and (not $t3)`
 - `xnor $t1,$t2,$t3` `#not ($t2 xor $t3)`
- Υλοποιήστε τις παραπάνω εντολές χρησιμοποιώντας τον ελάχιστο αριθμό εντολών του MIPS

11

Άσκηση 2-6

- Έστω ότι ο \$t0 περιέχει την τιμή:
 - a. 0110 1101 0001 0000 0000 0000 0000 0010
 - b. 1111 1111 1111 1111 1111 1111 1111 1111
- και ο \$t1 περιέχει την τιμή:
 - 0011 1111 1111 1000 0000 0000 0000 0000
- Ποια είναι η τιμή του \$t2 μετά τις επόμενες εντολές:

```
slt    $t2, $t0, $t1
beq    $t2, $zero, ELSE
j      DONE
ELSE:  addi  $t2, $zero, 2
DONE:
```
- Ποια είναι η τιμή του \$t2 εάν αντικαταστήσουμε την `slt` με την `sltu`:

12

Άσκηση 2-7

- Θεωρήστε τον παρακάτω βρόχο σε συμβολική γλώσσα MIPS:

```
LOOP: slt  $t2, $0, $t1
      bne  $t2, $0, ELSE
      j    DONE
ELSE:  addi $s2, $s2, 2
      addi $t1, $t1, -1
      j    LOOP
```

DONE:

- Υποθέστε αρχικές τιμές \$t1=10 και \$s2=0. Ποια είναι η τιμή του \$s2 μετά την εκτέλεση του βρόχου;
- Γράψτε την ισοδύναμη ρουτίνα σε C. (Υποθέστε ότι οι καταχωρητές \$s2 και \$t1 είναι οι ακέραιοι A και i.)
- Πόσες εντολές του MIPS εκτελούνται;

13

Άσκηση 2-8

- Γράψτε κώδικα MIPS που αποθηκεύει τις παρακάτω σταθερές στον \$t1

- a. 0010 1101 0001 0000 0000 0000 0000 0100
- b. 1111 1111 1111 1111 1111 1111 1111 1100

- Εάν η τρέχουσα τιμή του PC είναι 0x00000000 μπορείτε με μια εντολή άλματος (jump) να μεταφέρετε την εκτέλεση στις παραπάνω διευθύνσεις;
- Εάν η τρέχουσα τιμή του PC είναι 0x00400600 μπορείτε με μια εντολή διακλάδωσης (branch) να μεταφέρετε την εκτέλεση στις παραπάνω διευθύνσεις;
- Εάν η τρέχουσα τιμή του PC είναι 0x00000600 μπορείτε με μια εντολή διακλάδωσης (branch) να μεταφέρετε την εκτέλεση στις παραπάνω διευθύνσεις;

14

Άσκηση 3-1

- Γράψτε τη δυαδική κωδικοποίηση του δεκαδικού αριθμού -0.6875 στην αναπαράσταση IEEE 754 (α) απλής ακρίβειας και (β) διπλής ακρίβειας

- $0.6875 = 0.1011$
κανονικοποίηση: 1.011×2^{-1}

- (α) $sign = 1$, $exp = 127 + (-1) = 126$

1 01111110 01100000000000000000000000000000

- (β) $sign = 1$, $exp = 1023 + (-1) = 1022$

1 011111111110 01100000000000000000000000000000...0

15

Άσκηση 3-2

- Ποιος δεκαδικός αριθμός αναπαριστάται από τον αριθμό κινητής υποδιαστολής IEEE 754 απλής ακρίβειας

- 1 10000001 10100000000000000000000000000000

- $(-1)^{\text{πρόσημο}} \times (1+\text{κλάσμα}) \times 2^{(\text{εκθέτης}-\text{πόλωση})} =$

$$(-1)^1 \times (1+0.101) \times 2^{(129-127)} =$$

$$-1.625 \times 4 = 6.5$$

16

Άσκηση 3-3

- Ποιος είναι ο μικρότερος (α) κανονικοποιημένος και (β) μη-κανονικοποιημένος θετικός αριθμός (εκτός του +0) που μπορεί να αναπαρασταθεί σε IEEE 754 απλής ακρίβειας;
 - Δείξτε την δυαδική κωδικοποίηση και την δεκαδική τιμή
- (α) $0\ 00000001\ 000000000000000000000000$
 $(+1) * (2^{(1-127)}) * (1.000000000000000000000000)$
 $= 1.1754943508222875079687365372222\ e-38$
- (β) $0\ 00000000\ 0000000000000000000000001$
 $(+1) * (2^{(0-127)}) * (0.000000000000000000000001)$
 $= 7.0064923216240853546186479164496\ e-46$

17

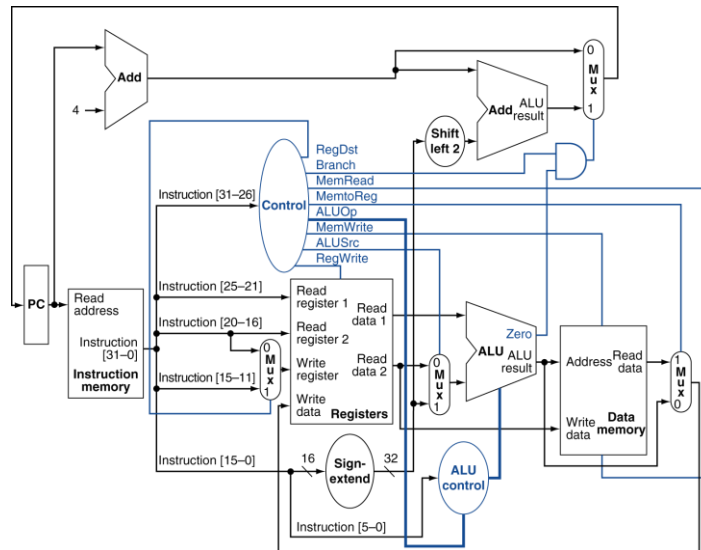
Άσκηση 4-1

- `add $4, $2, $3`
- `lw $3, 200($2)`
- `beq $2, $3, Label`

- Ποιες είναι οι τιμές των σημάτων Read register 1 και Read Register 2 και Write register;
- Ποια είναι η τιμή των σημάτων Read data 1, Read data 2, ALU result & Write data (υποθέστε ότι οι αρχικές τιμές των καταχωρητών είναι \$2=200, \$3=100 και της θέσης μνήμης `mem(400) = 50`);

18

Άσκηση 4-1



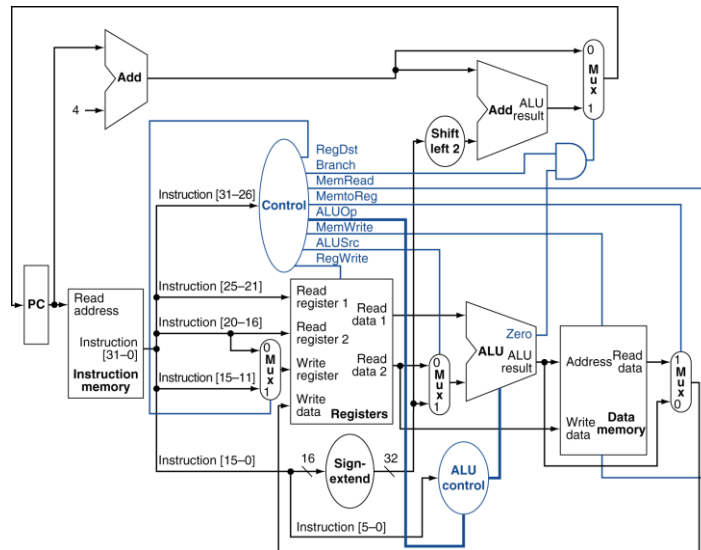
19

Άσκηση 4-2

- Μπορείτε να αφαιρέσετε το σήμα ελέγχου MemtoReg και στον αντίστοιχο πολυπλέκτη να χρησιμοποιήσετε το σήμα ελέγχου ALUSrc ή MemRead;
- Μπορεί ένα από τα δύο σήματα (MemRead και ALUSrc) να αντικαταστήσει το άλλο;

20

Άσκηση 4-2



21

Άσκηση 4-3

- Προσθέστε στην υλοποίηση του MIPS μία παραλλαγή της lw (load word) που προσθέτει δύο καταχωρητές για να υπολογίσει τη διεύθυνση των δεδομένων (είναι τύπου R). Εάν χρειάζεται προσθέστε λογική ή σήματα στη διαδρομή δεδομένων ή στη λογική ελέγχου.

22

Άσκηση 4-3

