

DATAPATH

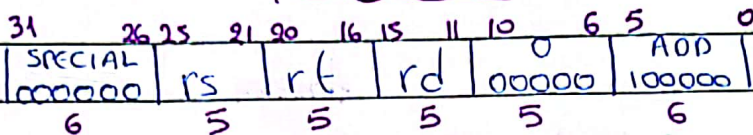
→ Τύποι εντολών:

α) Τύπου R

β) Τύπου I

γ) Τύπου J

⊗ Τύπου R: εντολή ADD



0-5 bits: function ops εντολής

6-10 bits: σημείο της εντολής (offset)

11-15 bits: καταχωρητής προορισμού

16-20 bits: δεύτερος καταχωρητής πηγής

21-25 bits: πρώτος καταχωρητής πηγής

26-31 bits: λειτουργία (operation)

$$\text{ADD } rd, rs, rt \Rightarrow rd \leftarrow rs + rt$$

Στο 6ης εντολής datapath:

Η διεκδίκηση της εντολής προέρχεται από PC. Έτσι έχω μια μικρή διακλάδωση.
- Το ένα μέρος της διακλάδωσης πηγαίνει στην αρίθμηση και του προσδίδονται 4 bytes. Το αποτέλεσμα του αριθμωτή είναι η διεκδίκηση, που ο αριθμωτής καθορίζει να πηγαίνει ως αποτέλεσμα ή απλά η ένα αποτέλεσμα που έρχεται από την ALU (όπου αλλιώς το αποτέλεσμα δεν με ενδιαφέρει). Ο αριθμωτής μπορεί να πάρει το 0. Έτσι έχω μια μικρή AND, αφού το Branch μας βοηθάει έτσι ώστε να μην να βγάζει τον αριθμωτή 0 και το αποτέλεσμα του αριθμωτή να είναι 0.

- Το άλλο μέρος της διακλάδωσης πηγαίνει στην διεκδίκηση αντιστοίχως και βγάζει τον 32bit εντολή. Τα τελευταία ^{operation} 6 bits της εντολής πηγαίνει στον έλεγχό, έτσι ώστε να βγάζει τα κατάλληλα σήματα για την εντολή. Τα σήματα

5 bits (rs), μία από τα επόμενα 32 bits, έτσι ώστε να γίνει
σε δεξιά ανώτατος, το επόμενο 32 bits ανώτατος 1, εντάξει
το επόμενο 32 bits. Τα επόμενα 5 bits (rt) είναι από τα επόμενα
32 bits ανώτατος 2, έτσι ώστε να γίνει σε δεξιά ανώτατος 2, εντάξει
το επόμενο 32 bits ανώτατος rt, και να γίνει σε RegDest. Τα επόμενα 5 bits
(rd) είναι από τα επόμενα 32 bits ανώτατος, έτσι ο ανώτατος έχει να γίνει
από τα επόμενα 32 bits ανώτατος rt και 32 bits rd. Ο ανώτατος έχει να γίνει
το rd, εντάξει αν έχει 1 η οποία είναι και σε RegDest. Αλλά
από τα επόμενα 32 bits ανώτατος da: μία η οποία είναι το rd: έτσι έτσι da να
να γίνει το αποτέλεσμα. Μετά το function και το stamp να είναι
εντάξει να γίνει. Το function είναι από τα επόμενα 32 bits ALU, και να είναι
το stamp 10 (ALUOp), και να γίνει το stamp να είναι ένας αριθμός
και το αποτέλεσμα να γίνει αριθμό εντάξει ALU, να είναι ALU, για να
είναι έτσι το αποτέλεσμα να γίνει να είναι να είναι.

Τώρα τα δεξιά ανώτατος 1 να είναι από τα επόμενα 32 bits ALU και μετά τα
δεξιά ανώτατος 2 να είναι από τα επόμενα 32 bits να είναι αριθμό 0 και
η ALUSrc να είναι αριθμό 0. Το αποτέλεσμα του ανώτατος να είναι από
ALU και από τα επόμενα 32 bits να είναι, έτσι ώστε το αποτέλεσμα
να γίνει αριθμό ALU, να είναι από τα επόμενα 32 bits και έτσι ο
ανώτατος, εντάξει να είναι το αποτέλεσμα της ALU, να είναι αριθμό 1.
Εντάξει το MemtoReg να είναι να είναι αριθμό 0. Επίσης να είναι
δεξιά το MemWrite να είναι αριθμό 0 για να γίνει να είναι,
ή να είναι το MemRead. Αλλά το αποτέλεσμα του rs και του rt να
για να είναι από τα επόμενα 32 bits ανώτατος. Επίσης να είναι να είναι,
το RegWrite da έτσι 1