

Laboration 4

HENRIK BÄCK
850611-6253

Karlstads Universitet
Första inlämning 2005-02-21
Ändrad p.g.a retur 2005-03-14

Handledare: Hans Hedbom
Nils Dåverhög

Deluppgift 1.

IF-steg: Här läser vi intruktionen från minnet.
Från 80020000 och detta ger 012a4020 som instruktion.

ID-steg: PC ökar med 4 för tala om var nästa instruktion finns

Vi sätter nu Read register 1 till 09
och därefter Read register 2 till 0a
Här finns de värden

När dessa har satts så hämtas dessa värden på Read data 1 och 2. Där får vi 15 och 18

Eftersom vi inte ska hoppa så spelar det ingen roll vad som kommer till
SignExtend

EX-steg: De tidigare värdena från ID-steg sättes till 0 eftersom efterföljande
instruktion är NOP.

Efter som vi inte skall hoppa så spelar det inte heller här någon roll vad som sker med
skiftningen.

Vi fick ut 15 och 18 ur ID/EXE dessa läggs in i ALU och där utförs nu add-instruktionen

Ut ur ALU får vi 2d efter addering.

08 på Instruktion och det är det register där vi skall skriva vårt resultat.

MEM-steg: De tidigare värdena från ID-steget sättes till noll.

Vi skall inte göra något i med minnet, och därför kommer detta steg bara
sätta utgången till WB till det värdet vi hade innan.

WB-steg: Vi får in 2d som sätts på det som skall skrivas till register

08 sätts på det register som skall skrivas till.

Det tar 5 klockcyklar innan värdet finns.

MemRead används inte. Därför det inte finns något att läsa från minnet

Deluppgift 2.

IF-steg: Läs in instruktionen 8d280000

ID-steg: Läs in register 08 och 09.

Vi får ut 80020020 från register 09 och 0 från register 08.

Vi måste läsa in båda för vi vet fortfarande inte vilken instruktion vi har.

EX-steg: Här utförs en addering med 0 eftersom vi inte har någon offset. Vi får kvar 80020020.

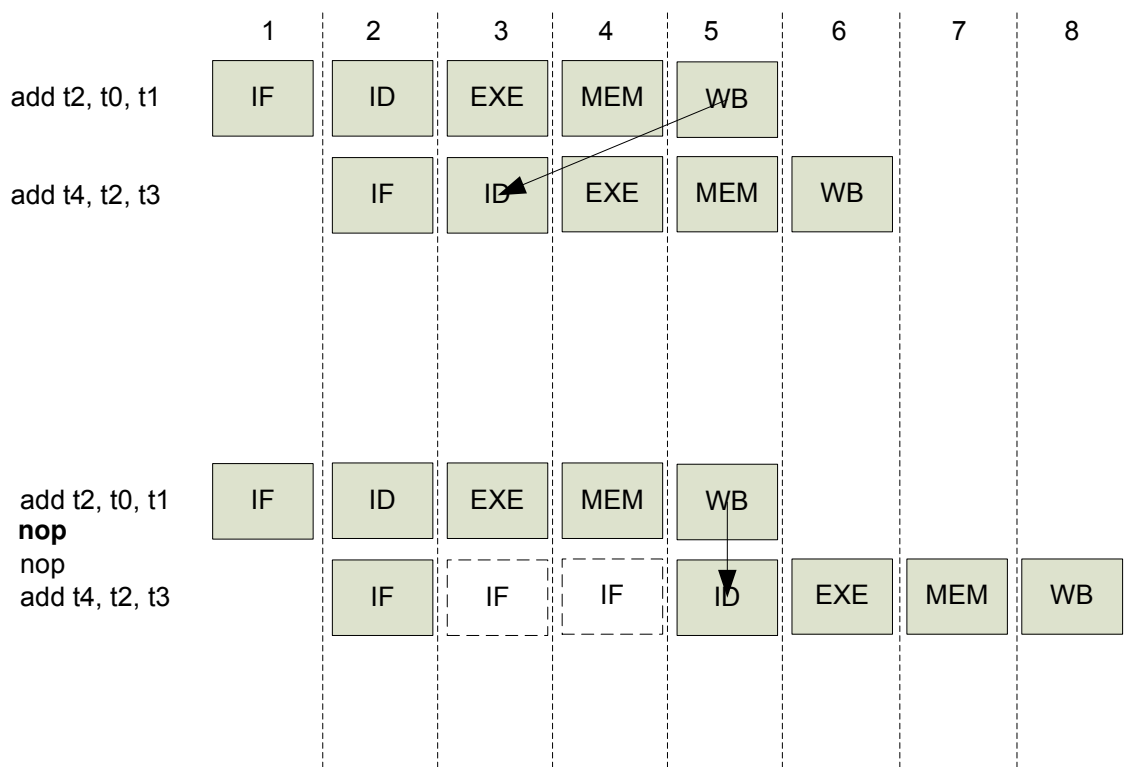
MEM-steg: Läser in nytt värde från minnet, alltså det värdet som ligger på 80020020

WB-steg: Här skrivs värdet från 80020020 till register 08.

Deluppgift 3.

Resultatet blir 0. Troligen därför beror det på att värdet från LUI inte är tillgängligt när lw skall spara den till t0. Därav finns det inget att spara och det blir "fel".

Deluppgift 4.



Deluppgift 5.

Det verkar fungera även utan NOP instruktioner. Därför att det finns en styrning som vet att värdet användes i föregående operation. Den har kvar värdet direkt efter additionen och kan därifrån ta värdet till den efterföljande operationen.