PROIECT MIPS 32

Documentatie

Proiectul a fost testat si incarcat pe placa si este complet functional. Am incercat sa fac o simulare in absenta unei placute de testare, dar am constatat ca era prea dificil sa urmaresc semnalele si/sau sa scriu codul astfel incat sa vizualizez tot ce voiam sa verific.

Partea cea mai dificila a fost la scrierea programului in sine. Fiind multe instructiuni, incurcam instructiunile, registrele intre ele sau locul lor in instructiune. Am rescris de cateva ori toate liniile in binar. A urmat trasarea executiei, unde am reusit cu usurinta sa imi dau seama ca legasem niste semnale gresit. Dupa ce am corectat, restul verificarii a fost usoara.

Cele 4 instructiuni alese:

**SRA**

Descriere: Deplasare aritmetică la dreapta pentru un registru, rezultatul este memorat în altul. Se repetă valoarea bitului de semn

RTL: $d 🡨 $t >> h; PC 🡨PC + 4;

ASM: sra $d, $t, h

Binar: 000000 00000 ttttt ddddd hhhhh 000011

**XOR**

Descriere: SAU-Exclusiv logic între două registre, memorează rezultatul în alt registru

RTL: $d 🡨 $s ^ $t; PC 🡨 PC + 4;

ASM: xor $d, $s, $t

Binar: 000000 sssss ttttt ddddd 00000 100110

**ORI**

Descriere: SAU logic între un registru și o valoare imediată, memorează rezultatul în alt registru

RTL: $t 🡨 $s | ZE(imm); PC 🡨 PC + 4;

ASM: ori $t, $s, imm

Binar: 001101 sssss ttttt iiiiiiiiiiiiiiii

**BNE**

Descriere: Salt condiționat dacă două registre sunt diferite

RTL: if $s ≠ $t then PC 🡨 (PC + 4) + (SE(offset) << 2) else PC 🡨 PC + 4;

ASM: bne $s, $t, offset

Binar: 000101 sssss ttttt oooooooooooooooo

Semnalele acestor 4 instructiuni sunt deja in fisierul cu semnale pentru toate instructiunile.