# Documentație MIPS32

Nume și prenume: Ungureanu George Octavian

Grupa: 30224

1. Componente MIPS

Pentru implementarea procesorului MIPS pe 32 de biți am folosit următoarele componente:

* MPG.VHD - generează un puls la fiecare apăsare a butonului
* IFetch.vhd – in această componentă sunt stocate toate instrucțiunile programului, oferind adresa instrucțiunii ce trebuie executată
* ID.vhd – decodifică instrucțiunea primită și generează coduri pentru function, sa, Ext\_Imm, Rd1 și Rd2, de asemenea pe intrările sale primește și adresa WD
* EX.vhd – această unitate efectuează operațiile aritmetice necesare, in funcție de intrările Rd1, Rd2, Ext\_Imm, Alu\_Src, sa, func, PC si ALURes
* MEM.vhd – in această componentă sunt păstrate valori pe 32 de biți, de asemenea scrierea este sincronă, iar citirea asincronă
* UC.vhd – generează semnalele pentru fiecare operație
* SSD.vhd – afișează pe un set de 8 afișoare SSD( seven segment display) informații utile precum: instrucțiunea curentă, adresa PC+4, Rd1, Rd2, WD, Ext\_Imm, MemData, AluRes
* test\_env.vhd – Aceasta este componenta ce ilustrează arhitectura întregului procesor

Întreg proiectul este complet funcțional, iar testarea a fost efectuată pe o plăcuță NEXYS A7-50T.

În timpul testării si creării proiectului au fost întâmpinate anumite probleme precum: scrierea greșită a Instrucțiunii-ului, scrierea greșită a generării adreselor pentru branch si jump și calcularea greșită a valorii branch.

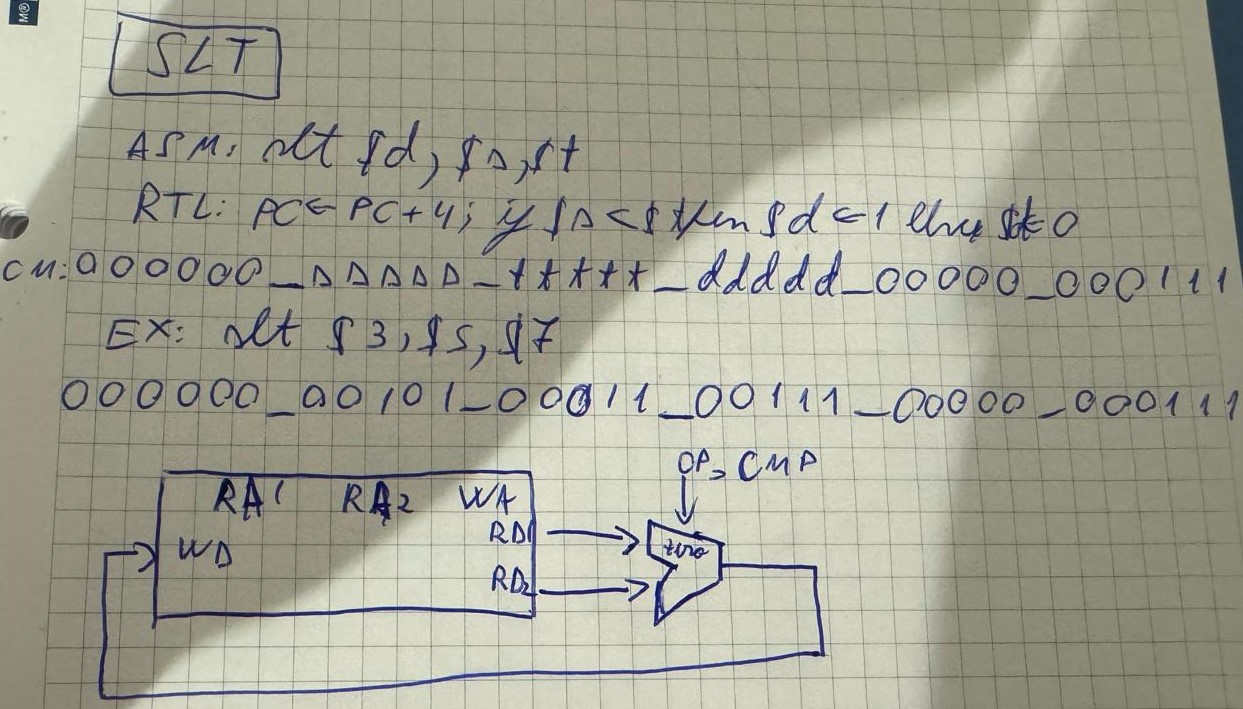
1. Instrucțiuni MIPS

O imagine care conține text, număr, captură de ecran, Paralel

Descriere generată automat

Instrucțiunile alese sunt BNE,ORI,SLT și XOR

SLT:



Instrucțiunea setează 1 pe registrul &d daca &s<&t

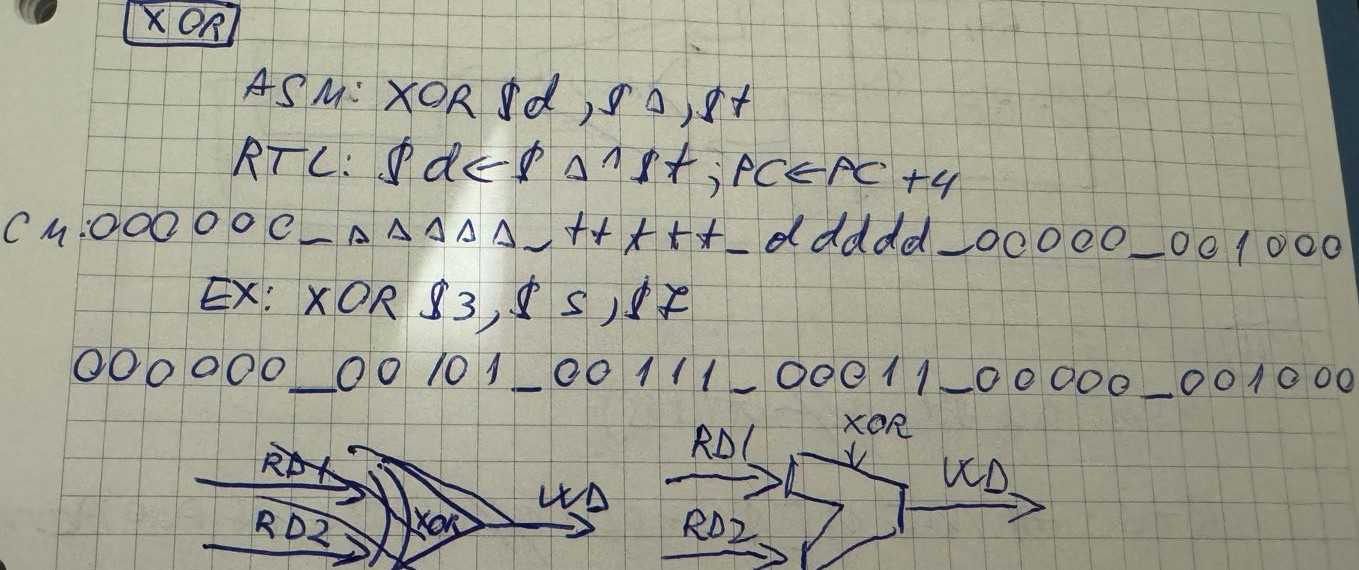
ORI:
O imagine care conține text, Font, scris de mână, linie

O imagine care conține text, scris de mână, linie, cerneală

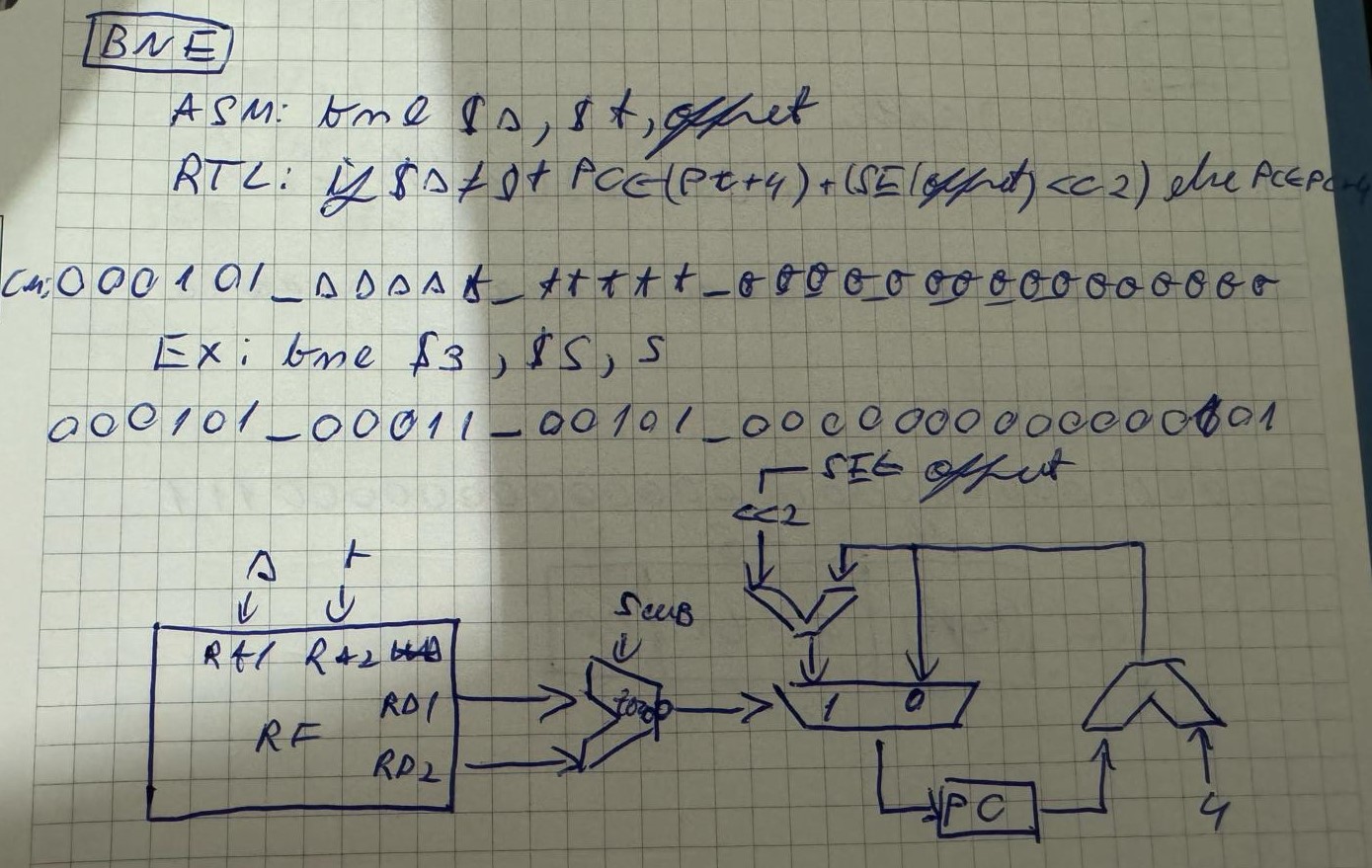
Descriere generată automat

Instrucțiunea execută OR immediate între două numere

XOR:



Instrucțiunea execută XOR între două numere

BNE:

Instrucțiunea verifică dacă două numere nu sunt egale, dacă este adevărat sare la adresa precizată

1. Execuția programului

Să se determine numărul de valori pozitive și impare dintr-un șir de N elemente

stocat în memorie începând cu adresa 8. N se citește din memorie de la adresa 4.

Numărul de valori determinate se va scrie în memorie la adresa 0

**Trasarea execuției programului de test pentru MIPS32**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Pas** | **SW(7:5)** | "000" | "001" | "010" | "011" | "100" | "101" | "110" | "111" | **De completat numai pentru instrucțiuni de salt** | |
| **Instr** (*în asamblare*) | **Instr** (*hexa*) | **PC+4** | **RD1** | **RD2** | **Ext\_Imm** | **ALURes** | **MemData** | **WD** | **BranchAddr** | **JumpAddr** |
| 0 | addi &10,&0,1 | 40A0001 | X"00000004" | X"00000000" | X"00000001" | X"00000001" | X"00000001" | X"00000000" | X"00000001" | X"" | X"" |
| 1 | addi &9,&0,4 | 4090004 | X"00000008" | X"00000000" | X"00000004" | X"00000004" | X"00000004" | X"0000000A" | X"00000004" |  |  |
| 2 | addi &8,&0,0 | 4080000 | X"0000000C" | X"00000000" | X"00000000" | X"00000000" | X"00000000" | X"00000000" | X"00000000" |  |  |
| 3 | addi $2,$0,0 | 4020000 | X"00000010" | X"00000000" | X"00000000" | X"00000000" | X"00000000" | X"00000000" | X"00000000" |  |  |
| 4 | addi $1,$0,0 | 4010000 | X"00000014" | X"00000000" | X"00000000" | X"00000000" | X"00000000" | X"00000000" | X"00000000" |  |  |
| 5 | lw &4,4($0) | 8040004 | X"00000018" | X"00000000" | X"0000000A" | X"00000004" | X"00000004" | X"0000000A" | X"0000000A" |  |  |
| 6 | lw &3,8($2) | 8430008 | X"0000001C" | X"00000000" | X"FFFFFFFFF" | X"00000008" | X"00000008" | X"FFFFFFFFF" | X"FFFFFFFFF" |  |  |
| 7 | beq&4,&8,7 | 10880007 | X"00000020" | X"0000000A" | X"00000000" | X"00000007” | X"0000000A" | X"FFFFFFFFF" | X"0000000A" | X”0000003C” |  |
| 8 | and&5,&3,&10 | 6A2804 | X"00000024" | X"FFFFFFFFF" | X"00000001" | X"00002804" | X"00000001" | X"00000000" | X"00000001" |  |  |
| 9 | slt &6,&0,&3 | 33007 | X"00000028" | X"00000000" | X"FFFFFFFFF" | X"00003007" | X"00000000" | X"00000000" | X"00000000" |  |  |
| 10 | bne&6,&5,1 | 14C50001 | X"0000002C" | X"00000000" | X"00000001" | X"00000001" | X"FFFFFFFFF" | X"00000000" | X"FFFFFFFFF" | X”00000030” |  |
| 11 | add&1,&1,&10 | 2A0800 | X"00000030" | X"00000000" | X"00000001" | X"00000800" | X"00000003" | X"00000000" | X"00000001" |  |  |
| 12 | add&8,&8,&10 | 10A4000 | X"00000034" | X"00000000" | X"00000004" | X"00004000" | X"00000001" | X"00000000" | X"00000001" |  |  |
| 13 | add&2,&2,&9 | 491000 | X"00000038" | X"00000000" | X"00000000" | X"00001000" | X"00000004" | X"0000000A" | X"00000004" |  |  |
| 14 | J 6 | 1C000006 | X"0000003C" | X"00000000" | X"00000000" | X"00000006" | X"00000000" | X"00000000" | X"00000000" |  | X”00000018” |
| 15 | add&2,&2,&10 | 491000 | X"00000040" | X"00000000" | X"0000000B" | X"00001000" | X"00000011" | X"0000000A" | X"00000026" |  |  |
| 16 | sw&1,8(&2) | C410008 | X"00000044" | X"00000000" | X"00000005" | X"00000008" | X"0000002E" | X"00000005" | X"00000005" |  |  |
| 17 |  |  |  |  |  |  |  |  |  |  |  |
| 18 |  |  |  |  |  |  |  |  |  |  |  |
| 19 |  |  |  |  |  |  |  |  |  |  |  |
| 20 |  |  |  |  |  |  |  |  |  |  |  |
| 21 |  |  |  |  |  |  |  |  |  |  |  |
| 22 |  |  |  |  |  |  |  |  |  |  |  |
| 24 |  |  |  |  |  |  |  |  |  |  |  |

1. Schema RTL

O imagine care conține diagramă, Plan, linie, Desen tehnic

Descriere generată automat