**Trasarea execuției programului de test pentru MIPS32**

Valorile se completează în hexazecimal așa cum trebuie să apară pe SSD. Succesiunea pașilor reprezintă ordinea de execuție în timp la apăsarea butonului ENable. **Pasul 0 corespunde stării inițiale a circuitului (PC = 0), iar** **pasul *N* caracterizează starea după apăsarea de *N* ori a butonului ENable**. Inițial registrele vor avea valoarea 0 (care se atribuie automat în lipsa unei inițializări explicite a RF), iar memoria de date RAM poate fi inițializată cu valori dorite. Tabelul se completează pentru tot programul sau, dacă are buclă, până la finalul primei iterații. *Buclă = revenirea execuției la o instrucțiune care a mai fost executată anterior.*

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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 | lw $1, 0($zero) | 8C010000 | 00000004 | 00000000 | 00000000 | 00000000 | 00000000 | 00000008 | 00000008 |  |  |
| 1 | lw $2, 4($zero) | 8C020004 | 00000008 | 00000000 | 00000000 | 00000004 | 00000004 | 00000005 | 00000005 |  |  |
| 2 | addi $3, $zero, 1 | 20030001 | 0000000C | 00000000 | 00000000 | 00000001 | 00000001 | 00000008 | 00000001 |  |  |
| 3 | addi $4, $zero, 1 | 20040001 | 00000010 | 00000000 | 00000000 | 00000001 | 00000001 | 00000008 | 00000001 |  |  |
| 4 | sll $3, $3, 3 | 000318C0 | 00000014 | 00000000 | 00000001 | 000018C0 | 00000008 | 00000000 | 00000008 |  |  |
| 5 | sll $4, $4, 3 | 000420C0 | 00000018 | 00000000 | 00000001 | 000020C0 | 00000008 | 00000000 | 00000008 |  |  |
| 6 | sw $3, 0($1) | AC230000 | 0000001C | 00000008 | 00000008 | 00000000 | 00000008 | 00000000 | 00000008 |  |  |
| 7 | addi $1, $1, 4 | 20210004 | 00000020 | 00000008 | 00000008 | 00000004 | 0000000C | 00000000 | 0000000C |  |  |
| 8 | sw $4, 0($1) | AC240000 | 00000024 | 0000000C | 00000008 | 00000000 | 0000000C | 00000000 | 0000000C |  |  |
| 9 | addi $1, $1, 4 | 20210004 | 00000028 | 0000000C | 0000000C | 00000004 | 00000010 | 00000000 | 00000010 |  |  |
| 10 | addi $5, $zero, 3 | 20050003 | 0000002C | 00000000 | 00000000 | 00000003 | 00000003 | 00000008 | 00000003 |  |  |
| 11 | addi $2, $2, 1 | 20420001 | 00000030 | 00000005 | 00000005 | 00000001 | 00000006 | 00000005 | 00000006 |  |  |
| 12 | (loop) beq $5, $2, 13(end) | 10A2000D | 00000034 | 00000003 | 00000006 | 0000000D | 00000003 | 00000008 | 00000003 |  | 0000001A = linia 26 din registrul de instructiuni |
| 13 | addi $1, $1, -8 | 2021FFF8 | 00000038 | 00000010 | 00000010 | FFFFFFFF8(-8) | 00000008 | 00000008 | 00000008 |  |  |
| 14 | lw $3, 0($1) | 8C230000 | 0000003C | 00000008 | 00000008 | 00000000 | 00000008 | 00000008 | 00000008 |  |  |
| 15 | srl $3, $3, 3 | 000318C2 | 00000040 | 00000000 | 00000008 | 000018C2 | 00000001 | 00000008 | 00000001 |  |  |
| 16 | addi $1, $1, 4 | 20210004 | 00000044 | 00000008 | 00000008 | 00000004 | 0000000C | 00000008 | 0000000C |  |  |
| 17 | lw $4, 0($1) | 8C240000 | 00000048 | 0000000C | 00000008 | 00000000 | 0000000C | 00000008 | 00000008 |  |  |
| 18 | srl $4, $4, 3 | 000420C2 | 0000004C | 00000000 | 00000008 | 000020C2 | 00000001 | 00000008 | 00000001 |  |  |
| 19 | addi $1, $1, 4 | 20210004 | 00000050 | 0000000C | 0000000C | 00000004 | 00000010 | 00000000 | 00000010 |  |  |
| 20 | add $6, $3, $4 | 00643020 | 00000054 | 00000001 | 00000001 | 00003020 | 00000002 | 00000008 | 00000002 |  |  |
| 21 | sll $6, $6, 3 | 000630C0 | 00000058 | 00000000 | 00000002 | 000030C0 | 00000010 | 00000000 | 00000010 |  |  |
| 22 | sw $6, 0($1) | AC260000 | 0000005C | 00000010 | 00000010 | 00000000 | 00000010 | 00000000(read first, then write) | 00000010 |  |  |
| 23 | addi $1, $1, 4 | 20210004 | 00000060 | 00000010 | 00000010 | 00000004 | 00000014 | 00000000 | 00000014 |  |  |
| 24 | addi $5, $5, 1 | 20A50001 | 00000064 | 00000003 | 00000003 | 00000001 | 00000004 | 00000005 | 00000004 |  |  |
| 25 | j 12 (fib\_loop) | 0800000C | 00000068 | 00000000 | 00000000 | 0000000C | 00000000 | 00000008 | 00000000 |  | 0000000C |
| 26=12 | (loop) beq $5, $2, 13(end) | 10A2000D | 00000034 | 00000004 | 00000006 | 0000000D | 00000002 | 00000008 | 00000002 |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |