**Quiz-5/Marks = 5/ Time= 20 mins**

A superscalar processor, shown below, is running a program given below. Show the execution sequence using “in-order-issue-in-order-completion” policy. Also calculate the run time. Calculate the uses 1GHz clock.

Fetch

Unit-1

Fetch

Unit-2

Decode

Unit-1

Decode

Unit-2

Integer ALU

Floating Point

Unit

Write-back

Unit-1

Write-back

Unit-2

Inst-1: floating point operation and requires 3 clock cycles

Inst-2: integer operation and requires one clock cycle

Inst-3: floating point operation, requires two clock cycles and depends on output of Inst-2.

Inst-4: Integer operation, requires two clock cycles

Inst-5: Integer operation, requires one clock cycle but depends on output of Inst-4

Inst-5 and Inst-6 are integer operations, require one clock cycle each but conflict for a register within CPU

Please note that Integer operations may use any ALU but floating point operations require floating point ALU only.

|  |  |  |
| --- | --- | --- |
|  | CLOCK | comments |
| FU1  Inst-1  FU2  Inst-2  DU1  DU2  I-ALU  FP-ALU  WBU1  WBU2 | 0 |  |
| FU1  Inst-3  FU2  Inst-4  DU1  Inst-1  DU2  Inst-2  I-ALU  FP-ALU  WBU1  WBU2 | 1 |  |
| FU1  Inst-5  FU2  Inst-6  DU1  Inst-3  DU2  Inst-4  I-ALU  Inst-2  FP-ALU  Inst-1  WBU1  WBU2 | 2 |  |
| FU1  Inst-5  FU2  Inst-6  DU1  Inst-3  DU2  Inst-4  I-ALU  FP-ALU  Inst-1  WBU1  WBU2 | 3 |  |
| FU1  Inst-5  FU2  Inst-6  DU1  Inst-3  DU2  Inst-4  I-ALU  FP-ALU  Inst-1  WBU1  WBU2 | 4 |  |
| FU1  Inst-5  FU2  Inst-6  DU1  Inst-3  DU2  Inst-4  I-ALU  FP-ALU  WBU1  Inst-2  WBU2  Inst-1 | 5 | Inst-1 takes 3 clock cycles  Since In-order-issue and In-order-completion is used, result of Inst-2 can’t be saved ahead of Inst-1.  Moreover, Inst-3 depends on Inst-2, so save the result of Inst-2 and issue of Inst-3 can’t be performed at the same clock |
| FU1  FU2  DU1  Inst-5  DU2  Inst-6  I-ALU  Inst-4  FP-ALU  Inst-3  WBU1  WBU2 | 6 |  |
| FU1  FU2  DU1  Inst-5  DU2  Inst-6  I-ALU  Inst-4  FP-ALU  Inst-3  WBU1  WBU2 | 7 | Both Inst-3 and Inst-4 require two clock cycles each |
| FU1  FU2  DU1  Inst-5  DU2  Inst-6  I-ALU  FP-ALU  WBU1  Inst-4  WBU2  Inst-3 | 8 | Inst-5 depends on Inst-4, so save the result of Inst-4 and Issue of Inst-5 can’t be performed at the same clock |
| FU1  FU2  DU1  DU2  Inst-6  I-ALU  Inst-5  FP-ALU  WBU1  WBU2 | 9 | Inst-5 and Inst-6 have conflict for a register within CPU so these two instructions can’t be issued at the same clock. |
| FU1  FU2  DU1  DU2  I-ALU  Inst-6  FP-ALU  WBU1  WBU2 | 10 | Due to recourse conflict, completion of Inst-5 is delayed. |
| FU1  FU2  DU1  DU2  I-ALU  FP-ALU  WBU1  Inst-5  WBU2  Inst-6 | 11 |  |

Alternate solution:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| |  | | --- | | CLOCK | | 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | 8 | | 9 | | 10 | | 11 | | |  |  | | --- | --- | | FU1 | FU2 | | I-1 | I-2 | | I-3 | I-4 | | I-5 | I-6 | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | | |  |  | | --- | --- | | DU1 | DU2 | |  |  | | I-1 | I-2 | | I-3 | I-4 | | I-5 | I-6 | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | | |  |  | | --- | --- | | I-ALU | FP-ALU | |  |  | |  |  | | I-2 | I-1 | |  | I-1 | |  | I-1 | |  |  | | I-4 | I-3 | | I-4 | I-3 | |  |  | | I-5 |  | | I-6 |  | |  |  | | |  |  | | --- | --- | | WBU1 | WBU2 | |  |  | |  |  | |  |  | |  |  | |  |  | | I-1 | I-2 | |  |  | |  |  | | I-3 | I-4 | |  |  | |  |  | | I-6 | I-5 | |

**Run time: 11x10-9 sec**

**Quiz-6/Marks = 5/ Time= 10 mins**

1. What do you understand by Register Renaming? 1
2. What types of data dependencies could be removed by this? Give examples. 3
3. Also comment what type of instruction issuing policy prefer this technique. 1



