



Final assessment project document

Team 10(semester) Members:

Khalid Amgad Hamed, Sec: 1, BN: 16

Kareem Omar Mostafa, Sec: 2, BN: 6

Marc Nagui Nabeel, Sec: 2, BN: 7

Sherif Essam Ahmad, Sec: 1, BN: 30



Updated Design:

Attached to this document an image file detailing the updated design will be present with the name: **Design (no hazards).png**.

The codes of the non-hazard CPU will be included in the no hazards folder in the **CPU (no hazards folder)**.

Attached too an image with the forwarding unit , hazard detection unit and branch prediction unit with the name **Design (hazards).png**

The codes of the full CPU after adding extra units will be included in the with **hazards folder in the CPU** .

Updated tables of codes:

A pdf containing tables of opcodes/ ALU operation codes / instruction control signal will be in the deliverables with the name: **design (no hazards) tables.pdf**

Note: assembling the codes converts them from the given form into a non-standard form that we used in our design.

What is not implemented in the processor:

- 1- The interrupt signal and the corresponding RTI instruction.
- 2- The hazard detection for the instruction (swap)
- 3- The hazard detection of (pop) not work correctly
- 4- A very extreme case in jz prediction(if 2 consecutive jmps and the second one is predicted wrong it will flush 2 instructions so the first jmp is cancelled too
- 5- cache
- 6- memory is working on one cycle
- 7- we read the ram from the output of assembler manually



Testing:

The testing of the project has been separated into two parts:

- 1- Testing without hazards(2 nops if needed): the CPU was tested from a functionality perspective to assure that all implemented instructions were working correctly. the tests were made by the team members as the testing occurred before the test cases from the T.A.s were published. The mentioned testing will be included in the folder **test cases** from the no hazards folder in the deliverables.
- 2- Testing with hazards: the CPU was test after integration with the hazard detection and handling modules using the provided test cases by the T.A.s The results of said tests will be included in **FinalTestCases** Folder.

- Some test cases we put 2 nops after pop instruction
- The branch.asm case I modified LDM R0,0 to ADD R0 ,R6,R6 while R6 =0 Because in our project LDM take 2 clock cycles and when I jump wrongly to it and correct it get stuck into the middle of the 2 cycles of the instructions
- We didn't run Branchpredcection.asm because it have interrupt which we haven't implement and the extreme case of 2 jmps and the second is wrong