BIRZEIT UNIVERSITY

**Faculty of Engineering & Technology**
**Department of Electrical and Computer Engineering**
**Summer Semester 2020/2021**

***ENC3310– Advanced Digital Design***
***Course Project Report***

***Instructor: Dr. Abdellatif Abu-Issa***

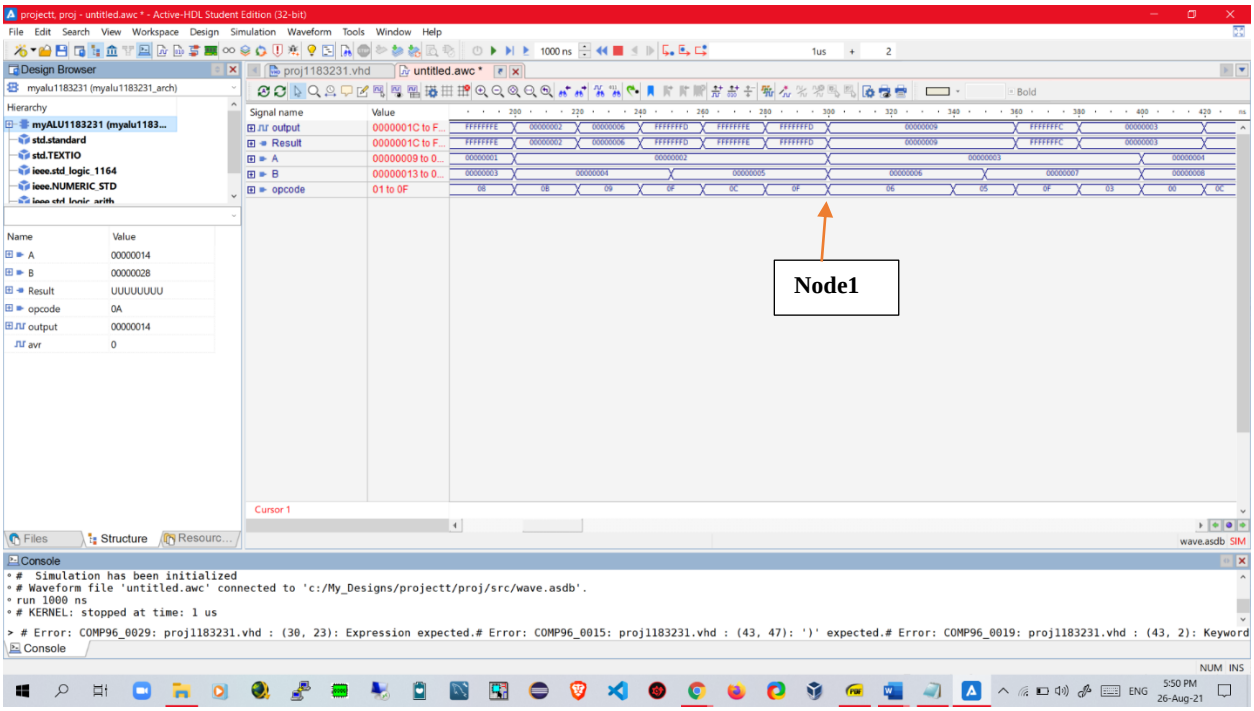
Student Name : Ahmad Barhoum
Student ID Number :1183231

ALU : Arithmetic logic unit , it is obvious from the name that this device can do arithmetic and logic operations on the input ,it has an opcode which by using it we can manipulate which operation to do on the input , as it has 2 input pins each from 32 bits , also one output from 32 bits

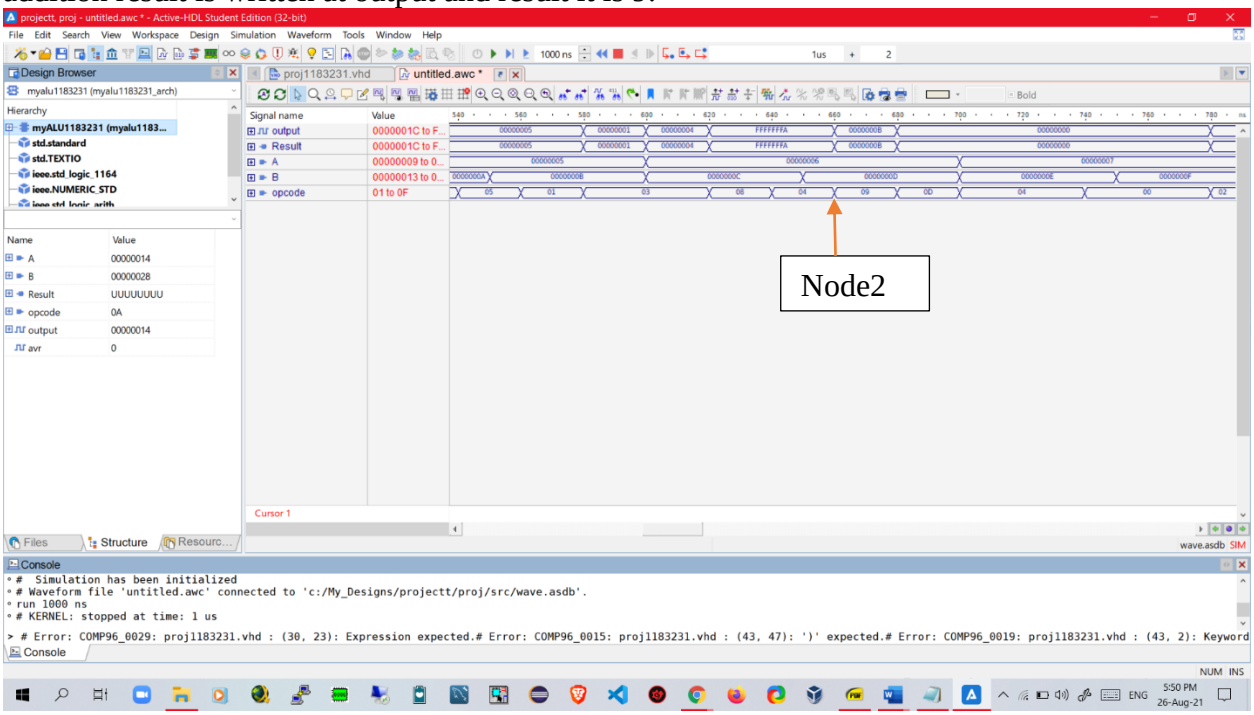
As my number 1183231 ends with 1 I used this table for the opcode operations

| operation | **Opcode Number** |
|---|---|
| a + b | **6** |
| a – b | **8** |
| \|a\| | **10** |
| -a | **12** |
| max(a,b) | **14** |
| min (a,b) | **11** |
| avg(a,b) | **13** |
| not a | **15** |
| a or b | **2** |
| a and b | **3** |
| a xor b | **9** |

And here is two of screenshots of the results after simulation:




I did not build a test bench as there is nothing in the project shows that I must do it , I entered the data in the waveform , as it is obvious in the screenshots it shows that it is working for example if we look at node1 the opcode value is (6) , so addition must be done , A = 6 and B=3 , the addition result is written at output and result it is 9.




At node2 the opcode is 9 so xor operation must be done , B=(D) =1101 in binary
A=(6)= 0110 in binary , after A xor B ,the result must be 1011 which equals to B in hex.
As this value is in result so ALU is working correctly.

Then after that the register file was made in other entity , it has  3 addresses as input each is 5 bits long  , also the main input 32 bits , also 2 outputs each is 32 bits this is used as memory unit which can deals with two demands at the time .
The initial values that should be added as my number 1183231 so this table was used:

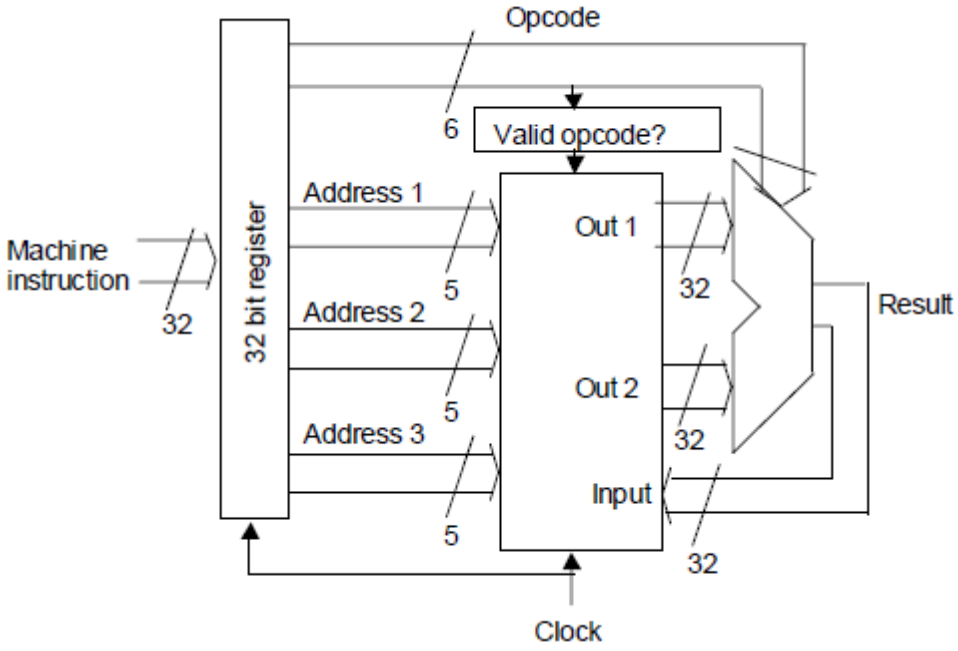
| | 3 |
|---|---|
| 0 | **0** |
| 1 | **12996** |
| 2 | **11490** |
| 3 | **7070** |
| 4 | **6026** |
| 5 | **3322** |
| 6 | **10344** |
| 7 | **6734** |
| 8 | **15834** |
| 9 | **15314** |
| 10 | **6000** |
| 11 | **12196** |
| 12 | **11290** |
| 13 | **13350** |
| 14 | **2086** |
| 15 | **6734** |
| 16 | **7430** |
| 17 | **14102** |
| 18 | **13200** |
| 19 | **3264** |
| 20 | **2368** |
| 21 | **15846** |
| 22 | **11710** |
| 23 | **14736** |
| 24 | **5338** |
| 25 | **5544** |
| 26 | **1852** |
| 27 | **3898** |
| 28 | **16252** |
| 29 | **1048** |
| 30 | **5642** |
| 31 | **0** |

In this part of code an error appeared , I tried many ways to solve it , I have used other ways to enter data , I have searched over the web , I saw that the way now in the code for entering the data is the best and most used also it is not complicated but it did not work with my code , there was no enough time to solve it , it took from me a lot of time, so I just pass to the next points as the project has many other details and thins that must be done, the main issue seems to be in the conversion .
So testing was not achieved due to the error but , the compile show that is no other errors in the entity.

Then a clock and enable pin was added as needed in part2 , in order for the system to be synchronous , so the issues of the combinational circuits will not occur (as overwrite).

Also there are more details in the comments in the code.

Using all of what is build before to create a simple small microprocessor
By using the work library as previous entities were compiled then nodes were connected by portmap , so no need to recode and initialize new entities that do and behave the same way as the previous ones.




Some other details such as the enable signal are added and commented in code.
For the micro pressor , it must first check the opcode if it is valid or not , (valid here means that equals one of the opcodes for the ALU as in the table shown above in ALU section) , so if the opcode is valid enable will be equal to 0 and the micro processor will do his work , if not it will not work.
Then work library for the ALU and register entity were used another time to get result and not to write new code while previous ones is used for same reason.

Then Last as my number is 1183231 so the 3rd number is 2
In this case , a code to find the minimum number stored in the register from 1 to 30 must be done.
As this one has no new nodes , just like a testbench so the entity has no lines in it, for architecture section the clock was generated by using the not statement as in the slides ,
I have not finish it as there is no enough time and the project not small and It has a lot of ideas , a lot of code but the idea that a loop must go from 0 to 30 , then it must compare using the ALU by using find minimum opcode between the first bit ( at 0) and the next one , then the min one must

be saved in a parameter , then this one saved in the parameter , must then be compared with the next one till it reaches the end of the loop ( the $30^{th}$ ).
Then the min number will be in the parameter.
I could not afford a full test for the system built as there are small errors .

References :


1- Slides for this subject(I merged them into one , so searching is easier).
2- The manual for the digital lab.
3- https://vhdlwhiz.com/initialize-ram-from-file/
4- https://www.xilinx.com/support/answers/4328.html
5- https://www.eng.auburn.edu/~nelson/courses/elec4200/Slides/VHDL%205%20Memory%20Models.pdf

6- https://www.fpga4student.com/2017/06/vhdl-code-for-arithmetic-logic-unit-alu.html