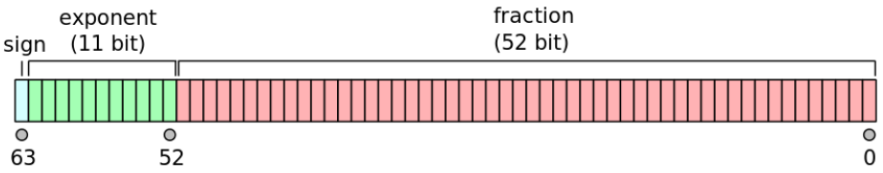
Pipelined Double Precision Floating Point Adder Design

(Self Project submission for resume verification)

Name: Aditya Ajay Wani **Roll No: 193079020**

Theory:

A floating point is used to represent fractional values, or when a wider range is needed. Double precision may be chosen when the range or precision of single-precision would be insufficient. The Format of Double Precision IEEE754 is representation below:

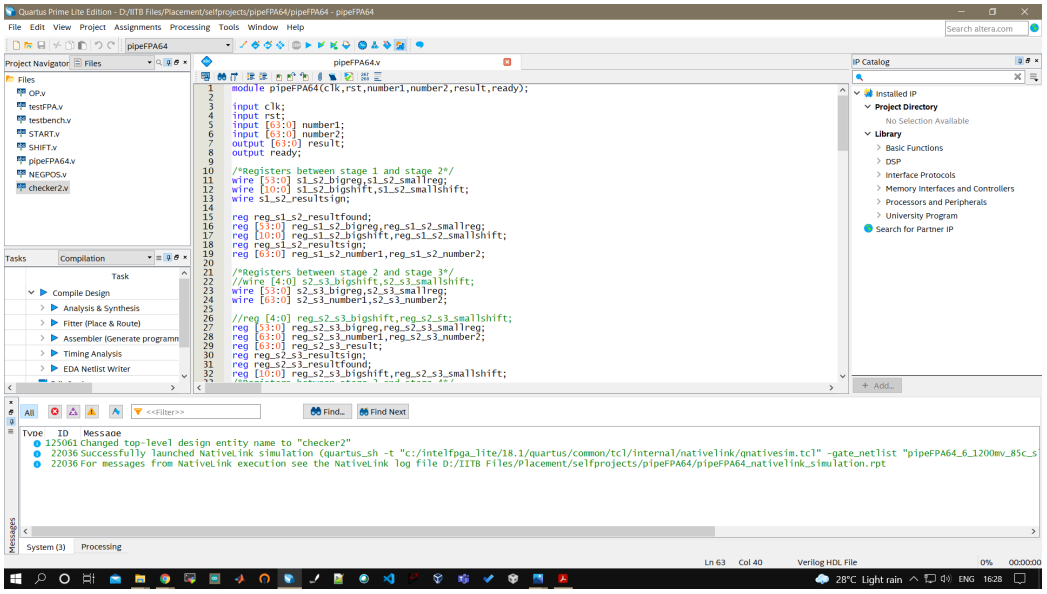


There is an assumed 1 before the mantissa part. To do addition of the two numbers we need to match the decimal point before addition. If the exponents are different then we need to do shifting of one of the numbers(preferable to shift smaller number left) and add/ subtract them later depending on the sign of the two numbers. The result obtained is to be normalized to match the assumed 1 to be one portion of the mantissa.

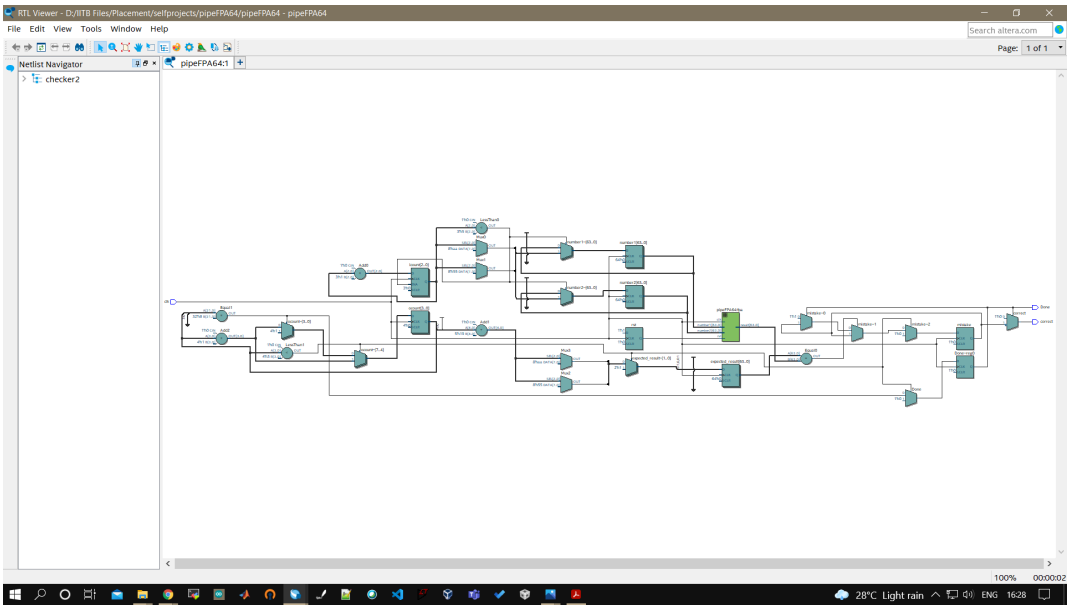
Implementation:

The module START identifies the bigger and the smaller numbers.
The module NEGPOS shifts the smaller number right by the necessary amount.
The module OP is used to add/subtract the to numbers after the necessary shifts.
The module SHIFT is used to normalize the result.

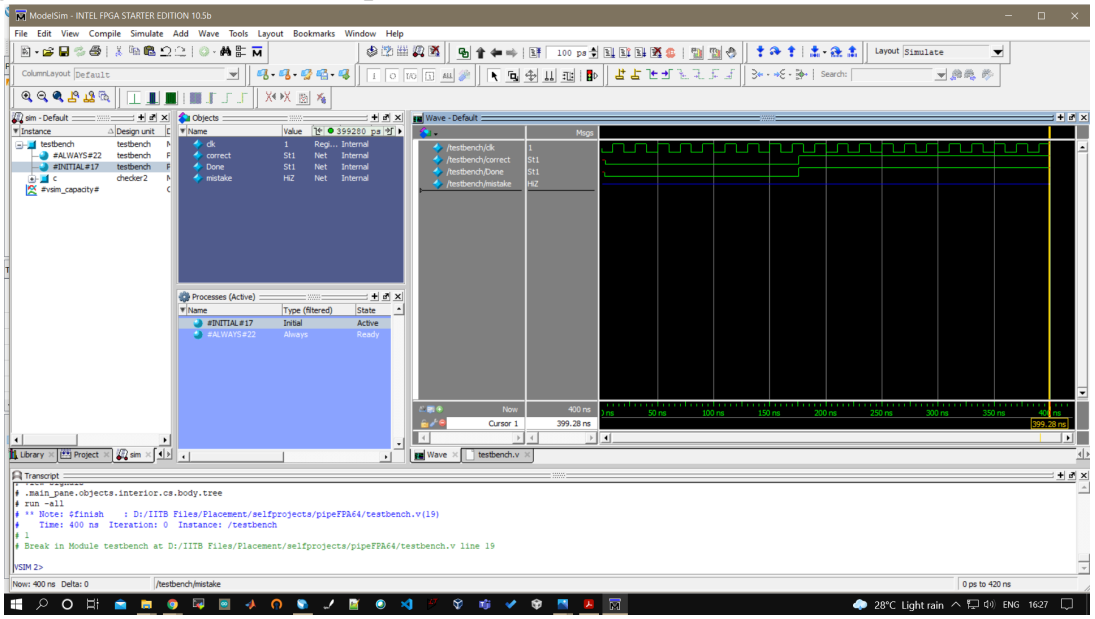
The top module is checker2.v which instantiates the pipeFPA64 module.
The checker2 inputs 4 different inputs to the adder and checks for the correct results at the output after 4 cycles as the results come out from the pipeline.
Testbench.v file is used as a testbench to instantiate the checker2 module.
The quartus window is shown below:



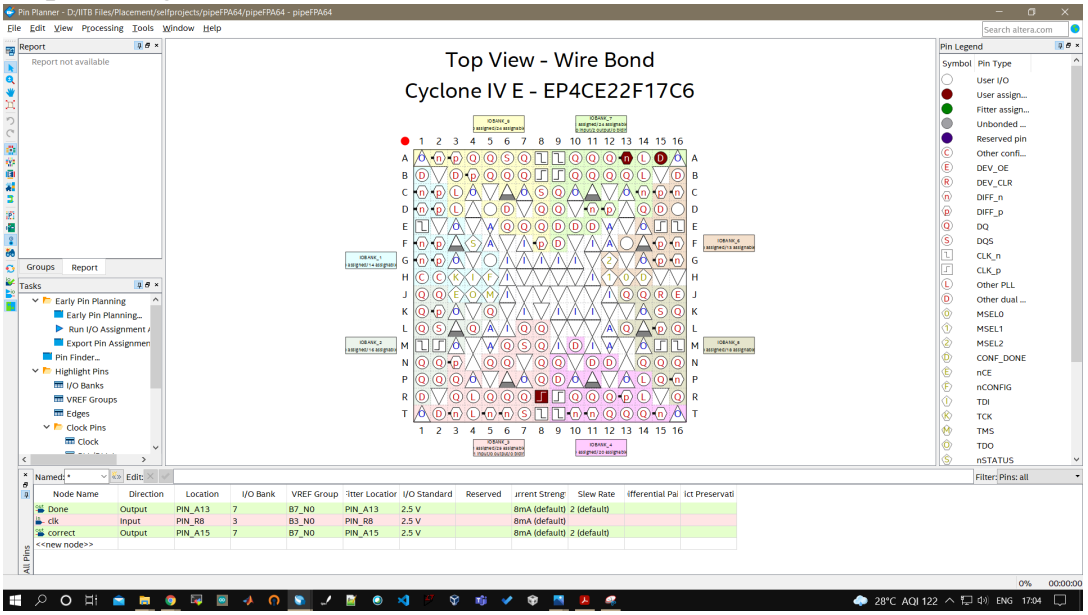
The netlist View is shown below:



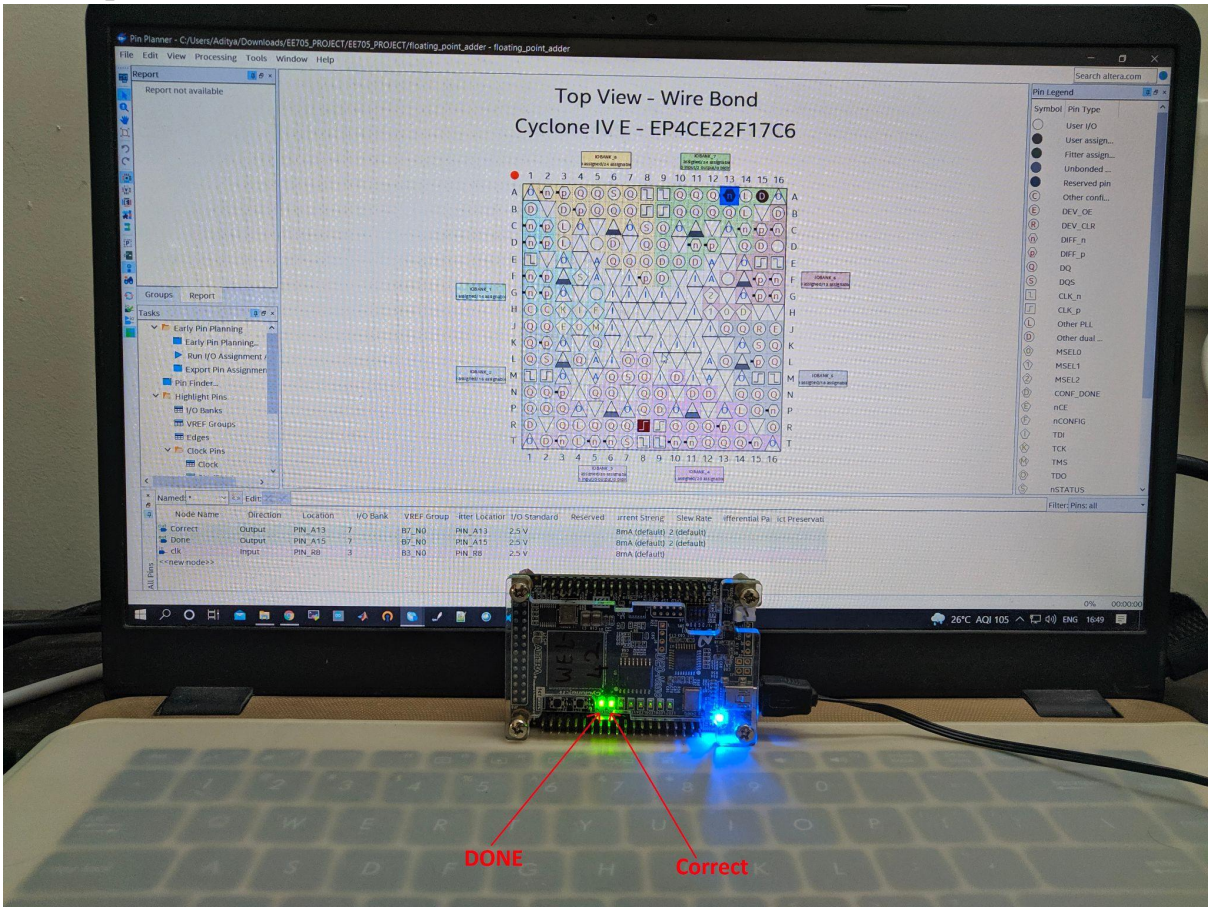
The gate-level simulation output is shown below:



The pin planning window for De0 nano FPGA is shown below:



The implementation on De0 Nano is shown below:



References:
[1] J. Thompson, N. Karra and M. J. Schulte, "A 64-bit decimal floating-point adder," IEEE Computer Society Annual Symposium on VLSI, 2004, pp. 297-298, doi: 10.1109/ISVLSI.2004.1339563.
[2] https://en.wikipedia.org/wiki/Double-precision_floating-point_format