

**Integrated Systems Architecture**

Lab 2

digital arithmetic

**Prof. Guido Masera - Prof. Maurizio Martina**

Group: ISA25

Payamreza Pourreza – S299859

Ali Fakour Razeghi – S299860

Hooman Khedersolh Sedeh

19 December 2022

**1 Digital arithmetic and logic synthesizers**

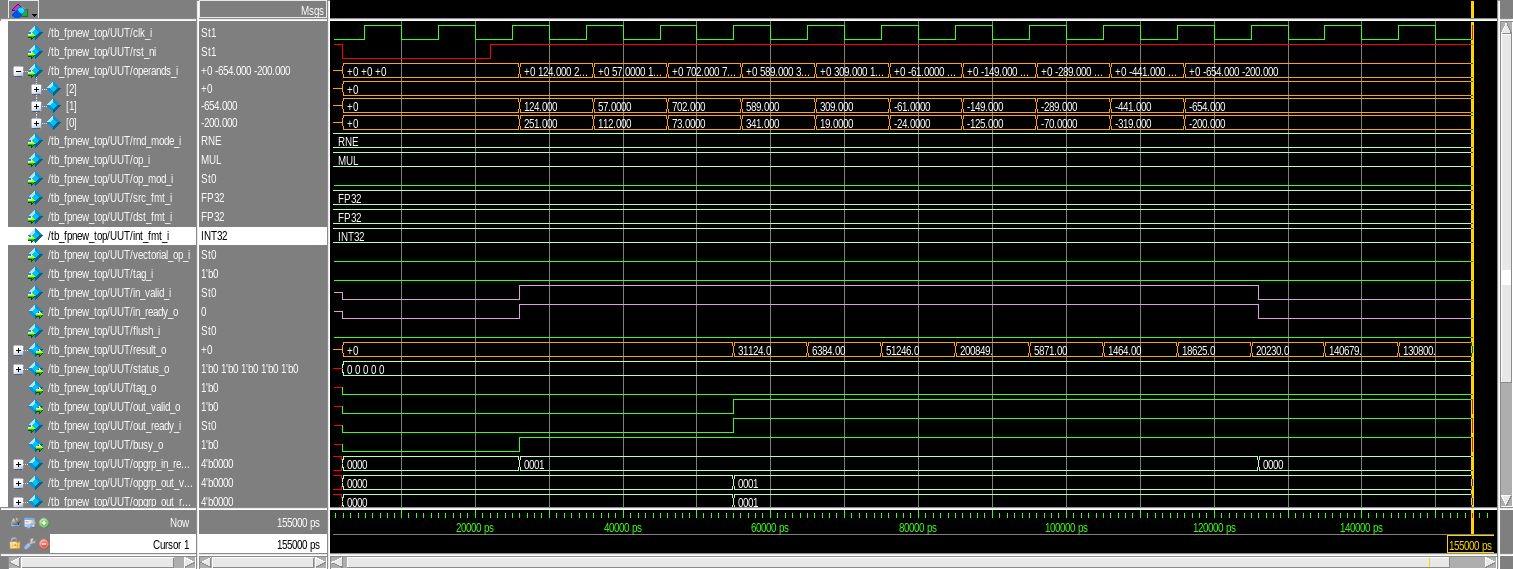
* 1. **Introduction**

**In this lab session, we are going to do floating point multiplication. To achieve the aims of this lab, we start with implementing the multiplication using the design ware components that can be used as CSA or PPARCH, then by R8-MBE (Radix 8 Modified Booth’s Encoder) and implement Dadda-like adder plane we can obtain a faster multiplier. Dadda tree uses a selection of full and half adders to sum the partial products in stages until two numbers are left. The design is similar to the Wallace multiplier, but the different reduction tree reduces the required number of gates and makes it slightly faster (for all operand sizes).**

Simulations and Logic Synthesis

1. **. Synthesis Strategies**

**Simulation and verification of the test bench:**

****

**2.1 Synthesize with compile. Find the maximum frequency and the area.**

**2.2 Repeat the previous step issuing the optimize registers command after compile. Find the maximum frequency and the area. Verify the netlist behavior via simulation.**

**2.3 Repeat the previous step issuing only the compile ultra command. Find the maximum frequency and the area. Verify the netlist behavior via simulation.**

**2.4 Force Design Compiler to flatten the hierarchy and to implement the Significands multiplier (Mantissa multiplier in fpnew\_fma.sv (3)) as a CSA multiplier. Find the maximum frequency and the area with the commands compile and optimize registers. Verify the netlist behavior via simulation.**

**2.5 Repeat the previous step by forcing the Design Compiler to implement the Significands multiplier as a PPARCH multiplier. Find the maximum frequency and the area with the commands compile and optimize registers. Verify the netlist behavior via simulation.**