LAB 3 REPORT

Shiyi Pang and Yang Liu

Introduction

In this lab, a sequential divider is designed to do the division calculation among unsigned numbers. This lab includes the main design component divider, one comparator, a constant package, decoder and a top level entity display. This lab is simulated on Modelsim and synthesized on Quartus. The final function of the division was implemented into DE2-115 board.

Function

In this lab, the main function was designed to do a calculation of long division with a finite statement effected by a clock signal, whenever the rising edge of the clock signal happens, the calculation starts immediately with using a comparator unit in this clock cycle, the same comparator unit was being used repeatedly during each clock cycle time.

When start button was pressed, it passes dividend and divisor value to a signal and being used for future calculation. Then it check the overflow case which the divisor is ‘0’. It return a ‘1’ value to overflow signal once overflow case happens.

Every cycle, the output isGreatEq signal is passed to the quotient vector from the highest to lowest digits sequentially. The output of DOUT of comparator is passed to a tem value and being combined with the next digits of dividend and make it the input signal for our next clock cycle for the comparator.

Problem & Hint

In this lab, the benefit of the sequential divider is that we could check each clock cycle’s comparator output and input which gives us all the values happens when calculating the long division.

The simulation is not work at first since we using a wrong divider_const value which is supposed to be 8 for divided and 4 for divisor.

Simulation

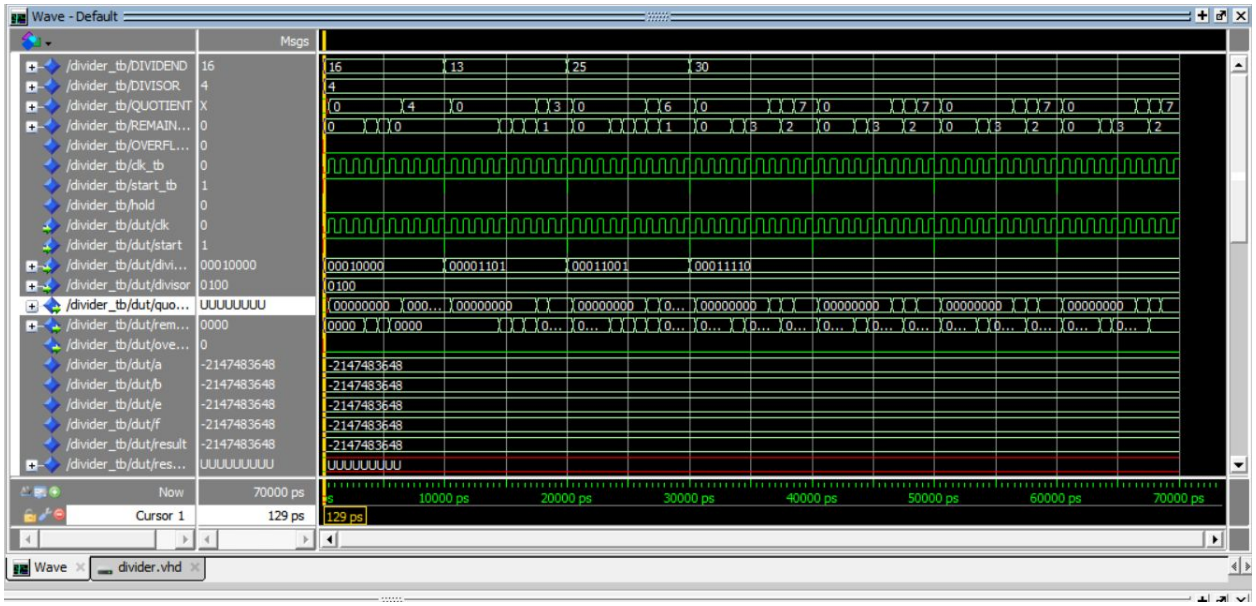

Figure 1 full division cycle (dividend width = 8 and divisor width =4)

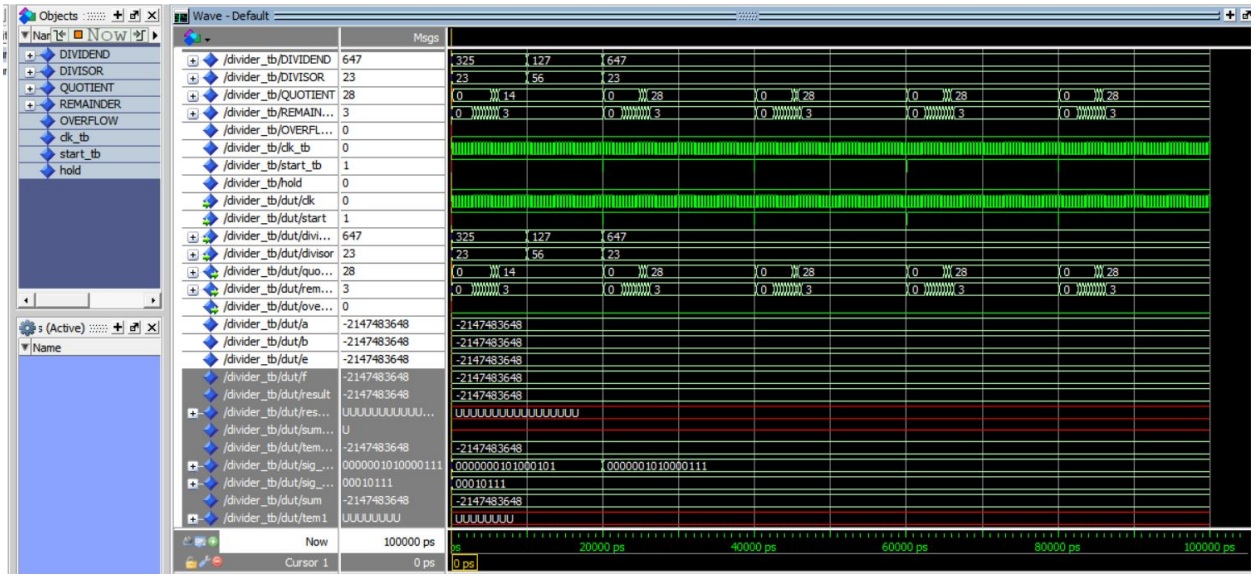

Figure 2 full division cycle (dividend width = 16 and divisor width = 8)

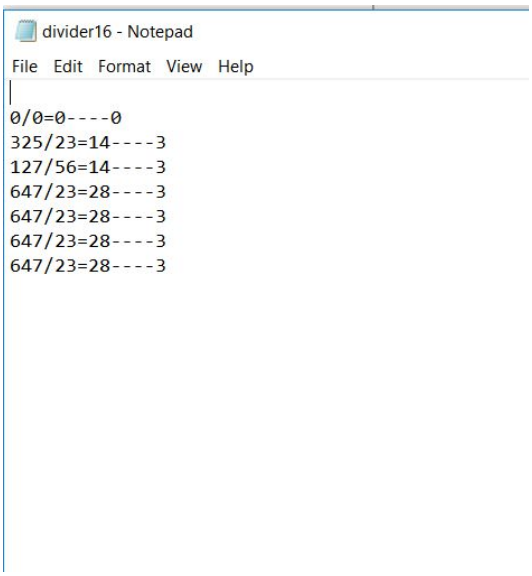

Figure 3 Simulation output (dividend width = 16 and divisor width = 8)

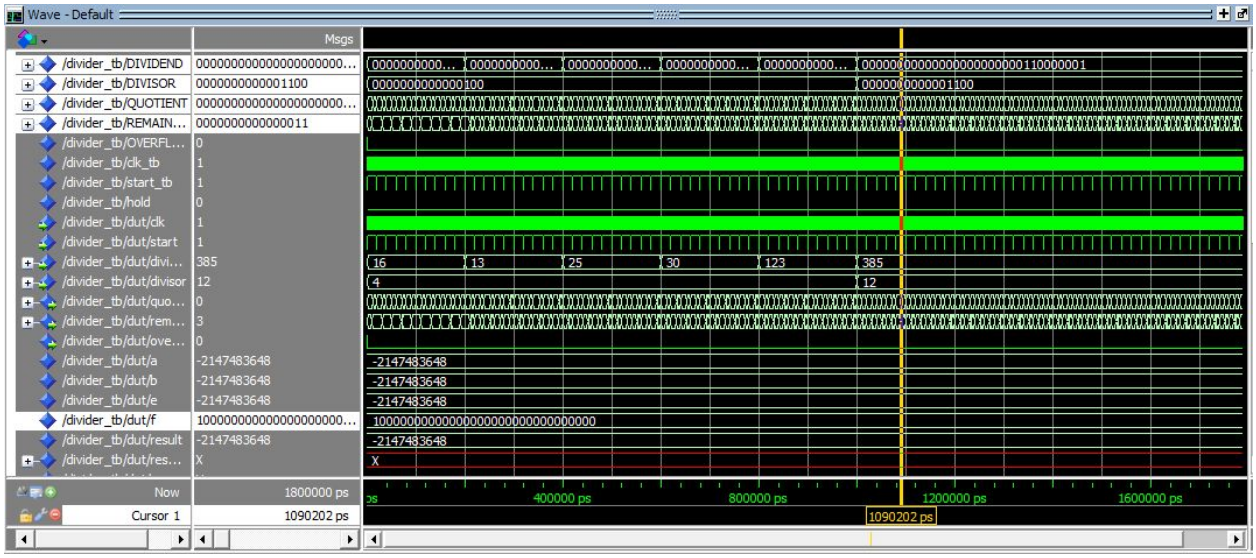

Figure 4 full division cycle (dividend width = 32 and divisor width = 16)

Figure 5 Simulation output (dividend width = 32 and divisor width = 16)