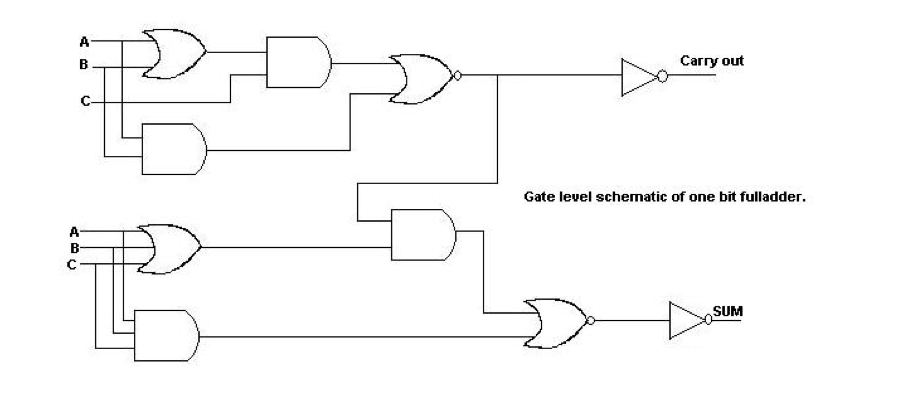
**Description of Circuit:**

The implemented circuit for this project is a one-bit full adder consisting of three half adders. The circuit has three inputs, and two outputs, one of which is a sum and the other which is carry. The circuit was implemented using a structural method, by creating entities for each gate and then calling upon the different components and mapping them together. This method lead to a very organized, straightforward, easy to understand program.

**Structural Diagram:**



**Code:**

library ieee;

use ieee.std\_logic\_1164.all;

entity threeor is port

 (A,B,C : in bit; sum,cout : out bit);

 end threeor;

 entity andgate is port

 (in1, in2 : in bit; out1 : out bit);

 end andgate;

 entity norgate is port

 (in1,in2 : in bit; out1: out bit);

 end norgate;

 entity orgate is port

 (in1,in2 : in bit; out1: out bit);

 end orgate;

 entity notgate is port

 (in1: in bit; out1 : out bit);

 end notgate;

 architecture behavior of andgate is

   begin

     out1 <= in1 and in2;

 end architecture;

 architecture behavior of norgate is

   begin

     out1 <= in1 nor in2;

 end architecture;

 architecture behavior of orgate is

   begin

     out1 <= in1 or in2;

 end architecture;

 architecture behavior of notgate is

   begin

     out1 <= not in1;

 end architecture;

 architecture fulladder of threeor is

 component andgate is port

   (in1, in2 : in bit; out1 : out bit);

 end component;

 component orgate is port

   (in1, in2 : in bit; out1 : out bit);

 end component;

 component norgate is port

   (in1, in2 : in bit; out1 : out bit);

 end component;

 component notgate is port

   (in1 : in bit; out1 : out bit);

 end component;

   signal d,e,f,g,h,i,j,k:bit;

   begin

     or1 : orgate port map(A,B,d);

     and1: andgate port map(A,B,e);

     and2: andgate port map(C,d,f);

     nor1: norgate port map(e,f,g);

     not1: notgate port map(g,cout);

     or2: orgate port map (C,d,h);

     and3:andgate port map(C,e,i);

     and4: andgate port map(g,h,j);

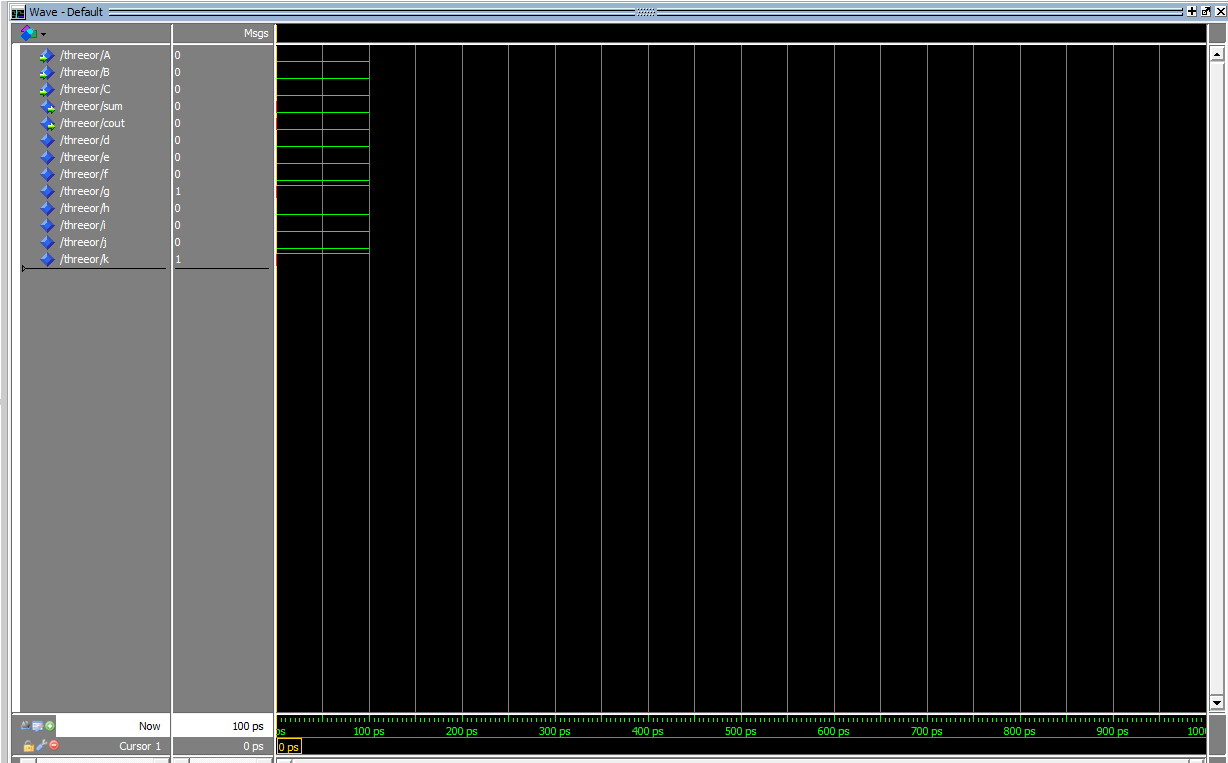
     nor2: norgate port map(i,j,k);

     not2: notgate port map(k,sum);

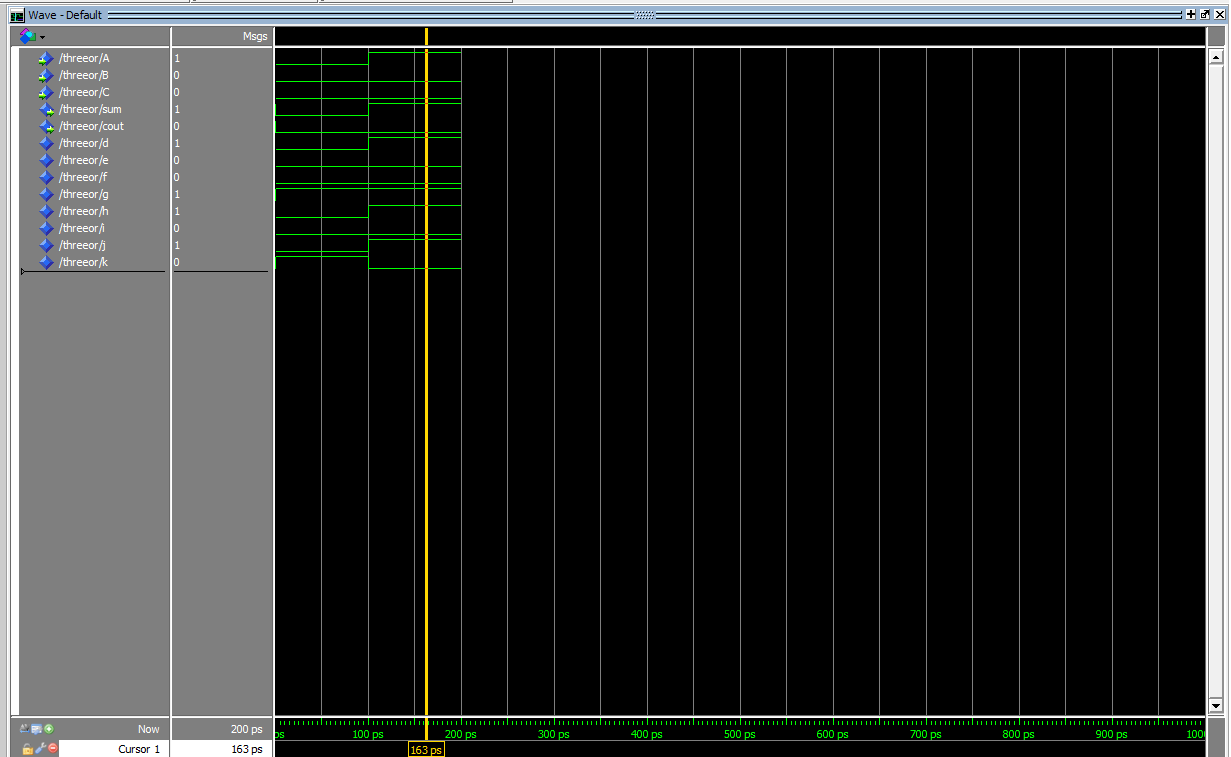
 end fulladder;

**Simulation:**

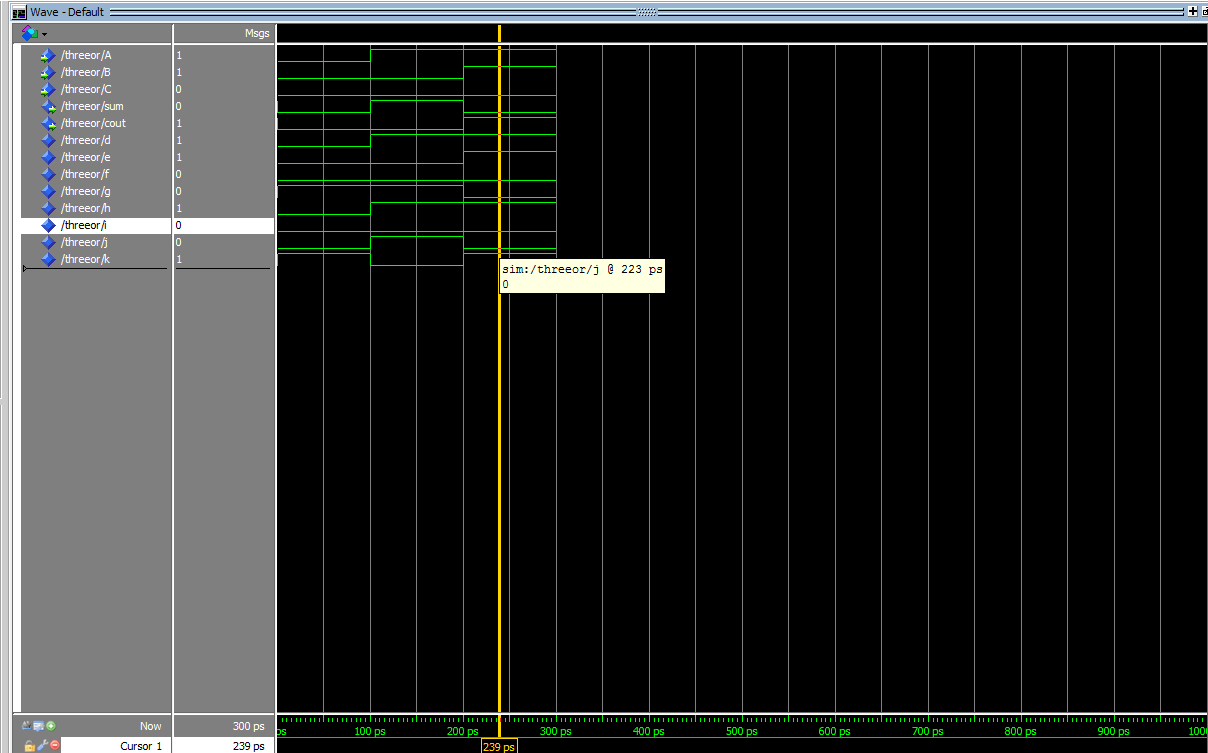
0+0+0

****

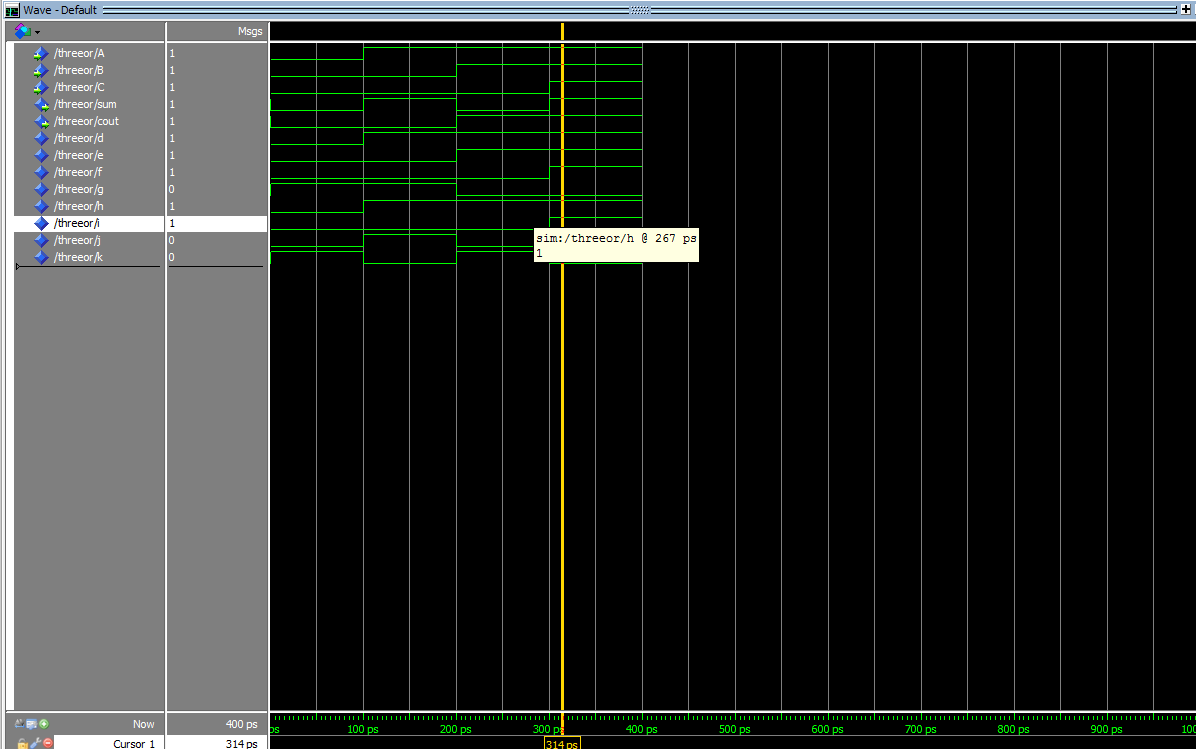
1+0+0



1+1+0



1+1+1



**Results:**

The inputs 0+0+0, 1+0+0, 1+1+0, and 1+1+1 were tested. 0+0+0 returned 0 and 0 for both sum and carry out. 1+0+0 returned 1 for sum and 0 for carry. 1+1+0 returned 0 for sum and 1 for carry. 1+1+1 returned 1 for both sum and carry.