Program of half subtractor:

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_ARITH.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

entity half\_sub is

port( A, B : in std\_logic;

DIFF, Borrow : out std\_logic);

end entity;

architecture dataflow of half\_sub is

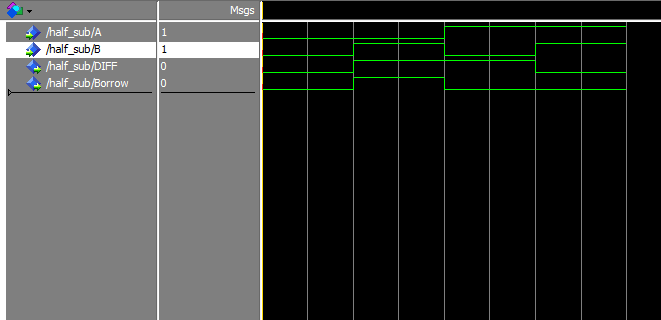
begin

DIFF <= A xor B;

Borrow <= (not A) and B;

end architecture;

OUTPUT:



Program of full subtractor:

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_ARITH.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

entity full\_sub is

port( A, B, C: in std\_logic;

DIFF, Borrow : out std\_logic);

end entity;

architecture dataflow of full\_sub is

begin

DIFF <= (A xor B) xor C;

Borrow <= ((not A) and (B or C)) or (B and C);

end dataflow;

OUTPUT:

