**VHDL Code of Half Adder:**

library IEEE;

entity HA is

Port ( a : in STD\_LOGIC;

b : in STD\_LOGIC;

sum : out STD\_LOGIC;

carry : out STD\_LOGIC);

end HA;

architecture Behavioral of HA is

begin

sum<= a xor b;

carry<= a and b;

end Behavioral;

**VHDL Code of Full Adder:**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity fa is

Port ( a : in STD\_LOGIC;

b : in STD\_LOGIC;

cin : in STD\_LOGIC;

s : out STD\_LOGIC;

cot : out STD\_LOGIC);

end fa;

architecture Behavioral of fa is

component HA

Port ( a : in STD\_LOGIC;

b : in STD\_LOGIC;

sum : out STD\_LOGIC;

carry : out STD\_LOGIC);

end component;

signal s1, c1, c2: std\_logic

begin

HA1: HA port map( a,b, s1, c1);

HA2: HA port map(s1, cin, s, c2);

cot<= c1 or c2;

end Behavioral;

**Testbench:**

ENTITY bbbb IS

END bbbb;

ARCHITECTURE behavior OF bbbb IS

COMPONENT fa111

PORT(

a : IN std\_logic;

b : IN std\_logic;

cin : IN std\_logic;

s : OUT std\_logic;

cout : OUT std\_logic

);

END COMPONENT;

--Inputs

signal a : std\_logic := '0';

signal b : std\_logic := '0';

signal cin : std\_logic := '0';

--Outputs

signal s : std\_logic;

signal cout : std\_logic;

BEGIN

uut: fa111 PORT MAP (

a => a,

b => b,

cin => cin,

s => s,

cout => cout

);

stim\_proc: process

begin

a<='0';

b<='0';

cin<='0';

wait for 100 ns;

a<='0';

b<='0';

cin<='1';

wait for 100 ns;

a<='0';

b<='1';

cin<='0';

wait for 100 ns;

a<='0';

b<='1';

cin<='1';

wait for 100 ns;

a<='1';

b<='0';

cin<='0';

wait for 100 ns;

a<='1';

b<='0';

cin<='1';

wait for 100 ns;

a<='1';

b<='1';

cin<='0';

wait for 100 ns;

a<='1';

b<='1';

cin<='1';

wait for 100 ns;

a<='0';

b<='0';

cin<='0';

wait for 100 ns;

wait;

end process;

END;

**Simulation:**

A screenshot of a computer

Description automatically generated

**UCF:**

net a loc = p87;

net b loc= p86;

net cin loc= p85;

net s loc = p162;

net cot loc = p165;

A screenshot of a computer

Description automatically generated