# VHDL Code of a Two – bit Multiplier

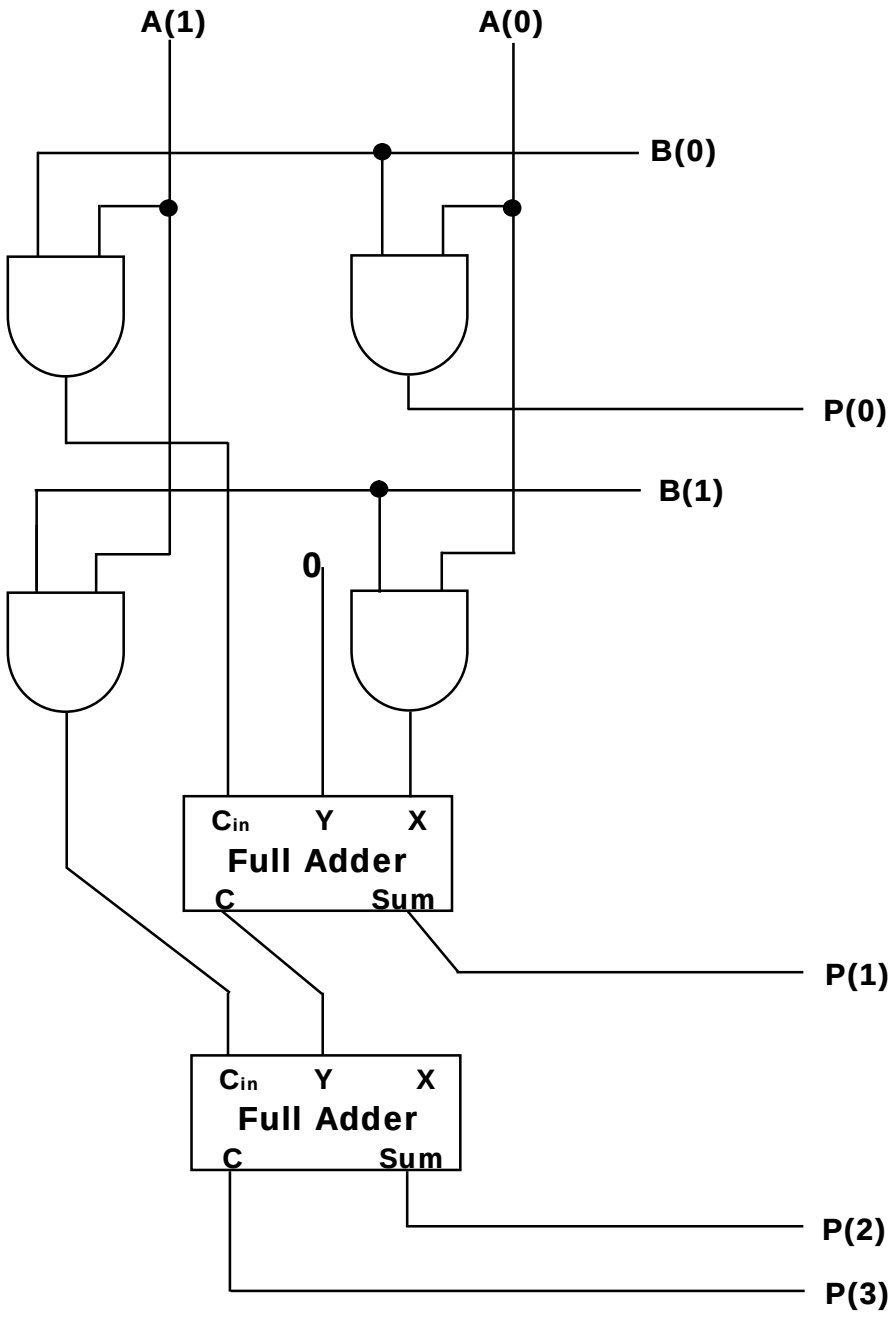


**Figure: A 2-Bit Multiplier**

**The VHDL code may be written as:**

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity AND2 is
            Port ( in1 : in std_logic; in2 : in std_logic;   out1 : out std_logic);
end AND2;

architecture AND2 of AND2 is

begin
        out1 <= in1 and in2;
end AND2;


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity FullAdd is
              Port(X: in STD_LOGIC; Y: in STD_LOGIC; Cin: in STD_LOGIC;
                   SUM: out STD_LOGIC; C: out STD_LOGIC);
end FullAdd;

architecture Fadd of FullAdd is

begin
         SUM <= (X xor Y xor Cin);
         C   <= ((X and Y) or ((X xor Y) and Cin));
end Fadd;

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Mul is
   Port ( A : in std_logic_vector(1 downto 0);
          B : in std_logic_vector(1 downto 0);
          P : out std_logic_vector(3 downto 0));
end Mul;

architecture Mul of Mul is

        component AND2  Port ( in1 : in std_logic; in2 : in std_logic;
                               out1 : out std_logic);
        end component;

        component FullAdd  Port(X: in STD_LOGIC; Y: in STD_LOGIC;
                              Cin: in STD_LOGIC;  SUM: out STD_LOGIC;
                               C: out STD_LOGIC);
        end component;

        signal temp: STD_LOGIC_VECTOR(1 to 4);
```