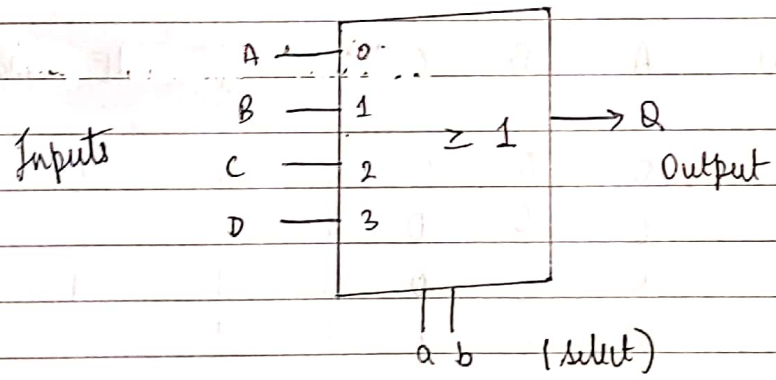
LAB 4: Given any 4 variable logic expression, simplify using Entered Variable Map and realize the simplified logic expression using 8:1 multiplexer IC. And implement the same in HDL.

Multiplexer

→ A multiplexer, also known as a data selector, is a device that selects between analog or digital input signals and forwards it to a single output line.

→ A multiplexer of 2^n inputs has select lines, which are used to select which input line to send to the output.



Map Entered Variable (MEV)

A Map Entered Variable is a Karnaugh map in which the size of the map is reduced by removing one or more of the variables from the specification of map cell locations.

→ One of the Input variable is placed inside Karnaugh map.

→ This reduces the Karnaugh map size by one degree.

i.e; a 3 variables problem that requires

2^3 = 8 locations in Karnaugh map will require $(2)^{3-1}$ = 4 locations in entered variable map.

→ This technique is particularly useful for mapping problems with more than 4 input variables.

Example:

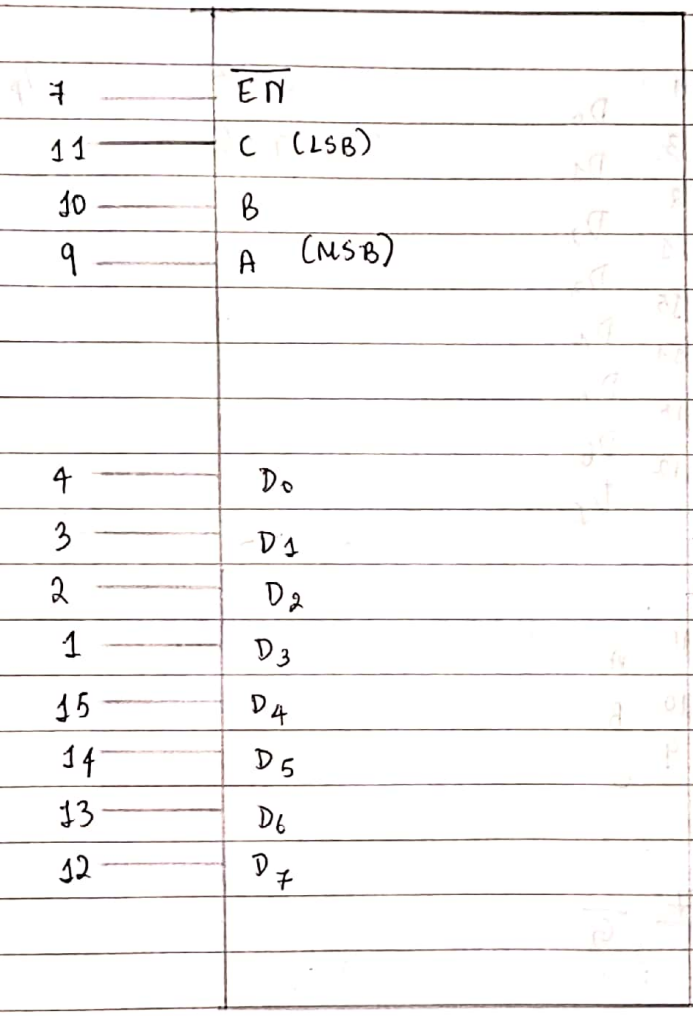
$f(A,B,C,D) = \Sigma m(2,3,4,5,13,15) + dc(8,9,10,11)$

Decimal	A	B	C	D	f	MEV	map entry
0	0	0	0	0	0	0	D_0
1	0	0	0	1	0		
2	0	0	1	0	1	1	D_1
3	0	0	1	1	1		
4	0	1	0	0	1	1	D_2
5	0	1	0	1	1		
6	0	1	1	0	0	0	D_3
7	0	1	1	1	0		
8	1	0	0	0	X	X	D_4
9	1	0	0	1	X		
10	1	0	1	0	X	X	D_5
11	1	0	1	1	X		
12	1	1	0	0	0	D	D_6
13	1	1	0	1	1		
14	1	1	1	0	0	D	D_7
15	1	1	1	1	1		

8 : 1 multiplexer IC

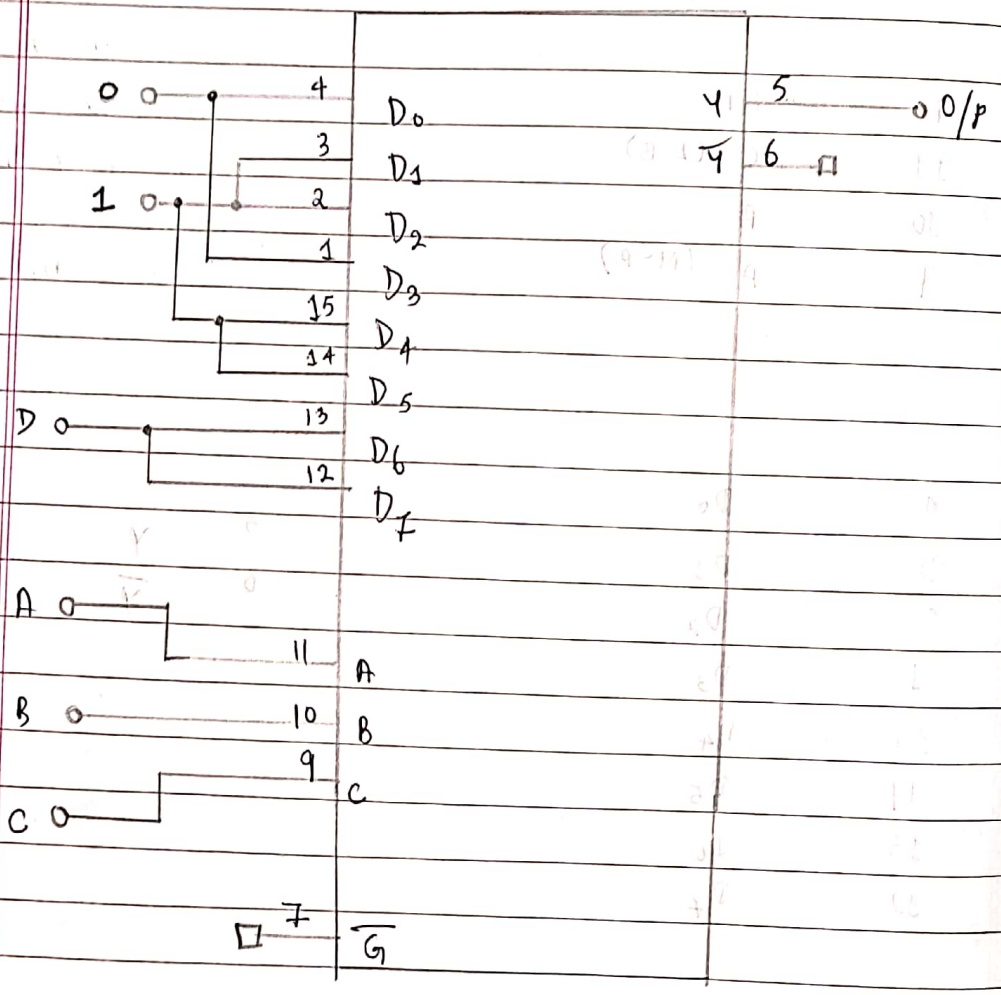
Pin details - IC 74151.

74 x x 151

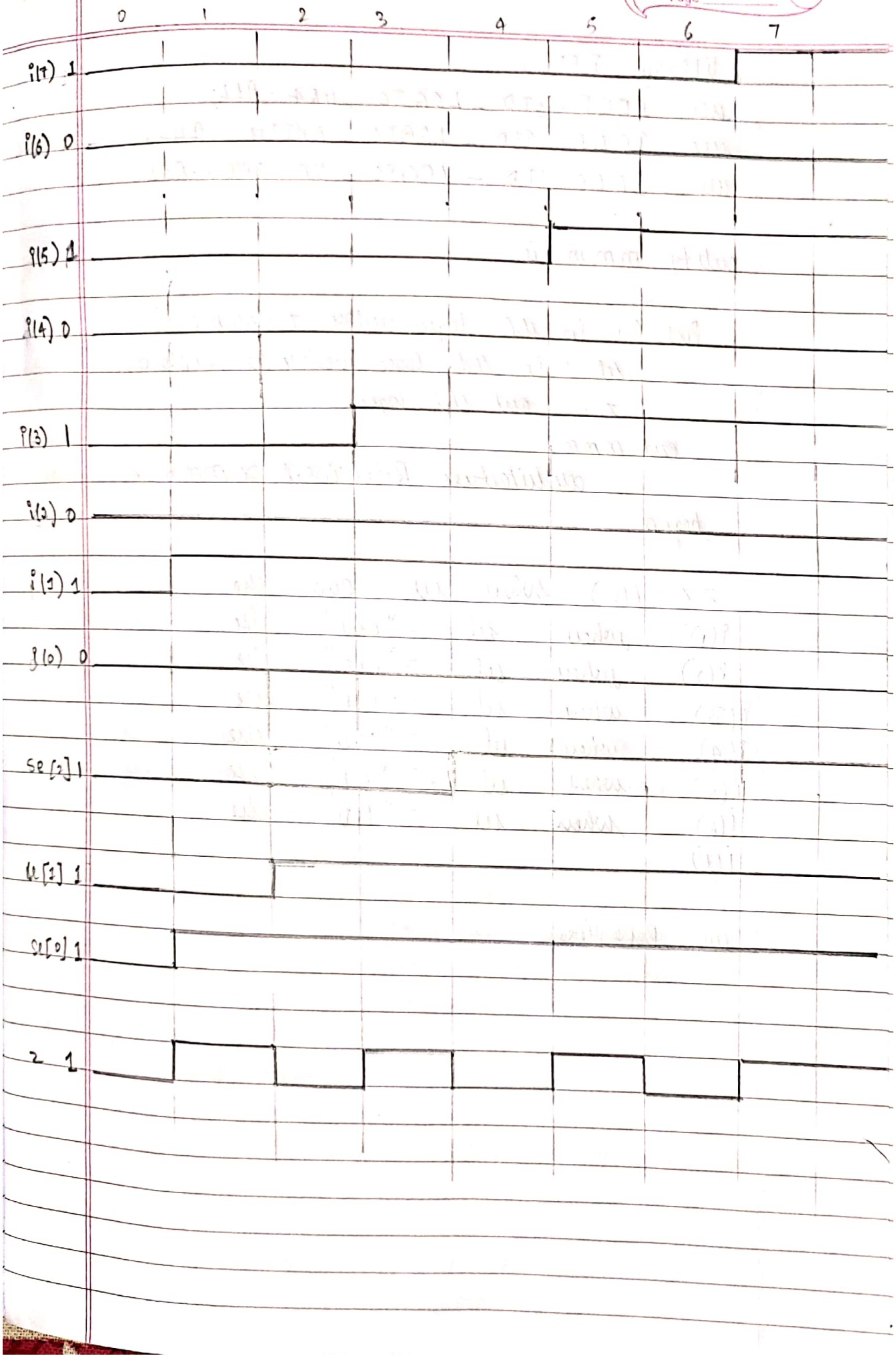


8:1 Multiplexer.

Circuit Diagram:



Multiplexer.



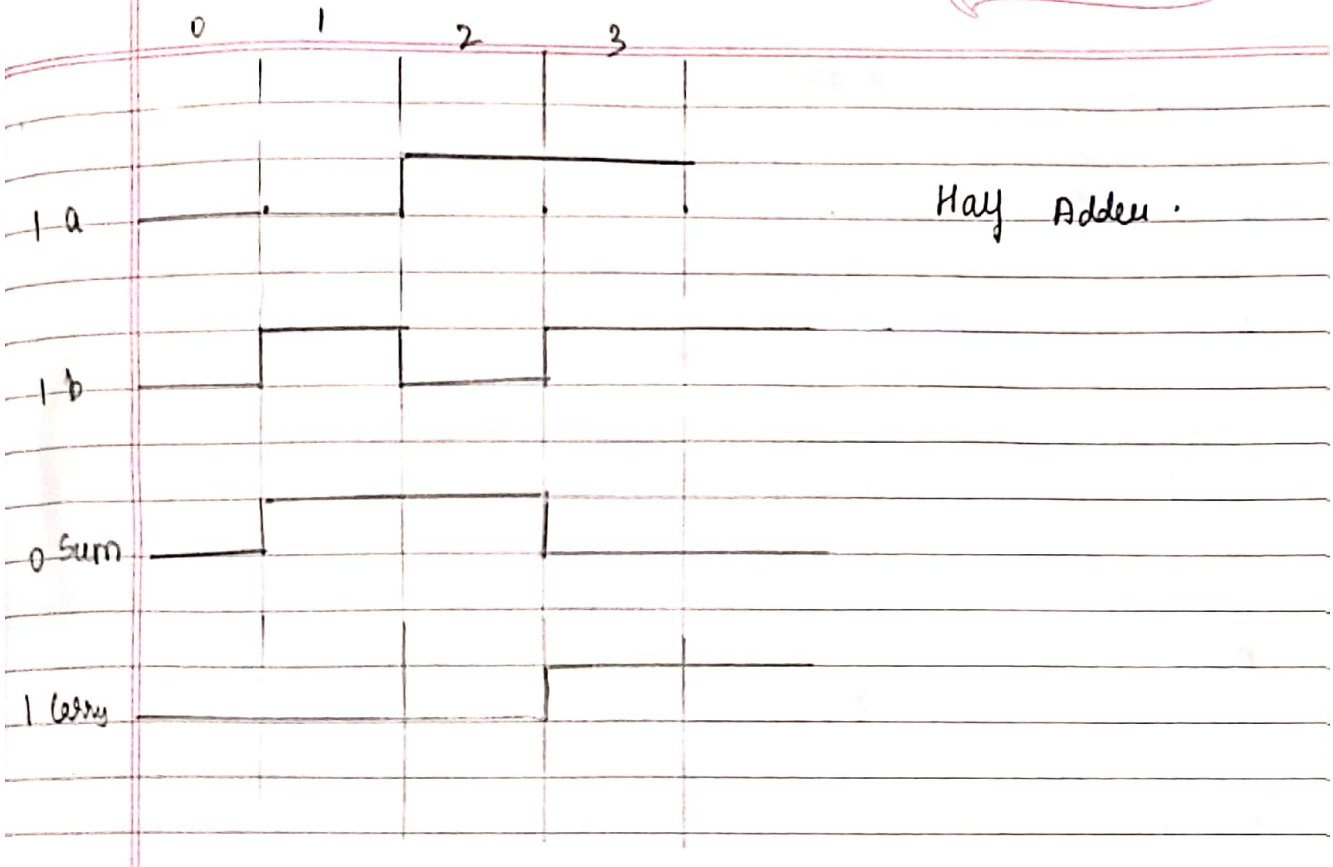
```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity mmm is

    Port ( i : in std_logic_vector (7 downto 0);
           sel : in std_logic_vector (2 downto 0);
           z : out std_logic);
end mmm;
        architecture Behavioral of mmm is
begin

z <= i(0) when sel = "000" else
i(1) when sel = "001" else
i(2) when sel = "010" else
i(3) when sel = "011" else
i(4) when sel = "100" else
i(5) when sel = "101" else
i(6) when sel = "110" else
i(7)

end Behavioral;
```



Half Adder.

Code.

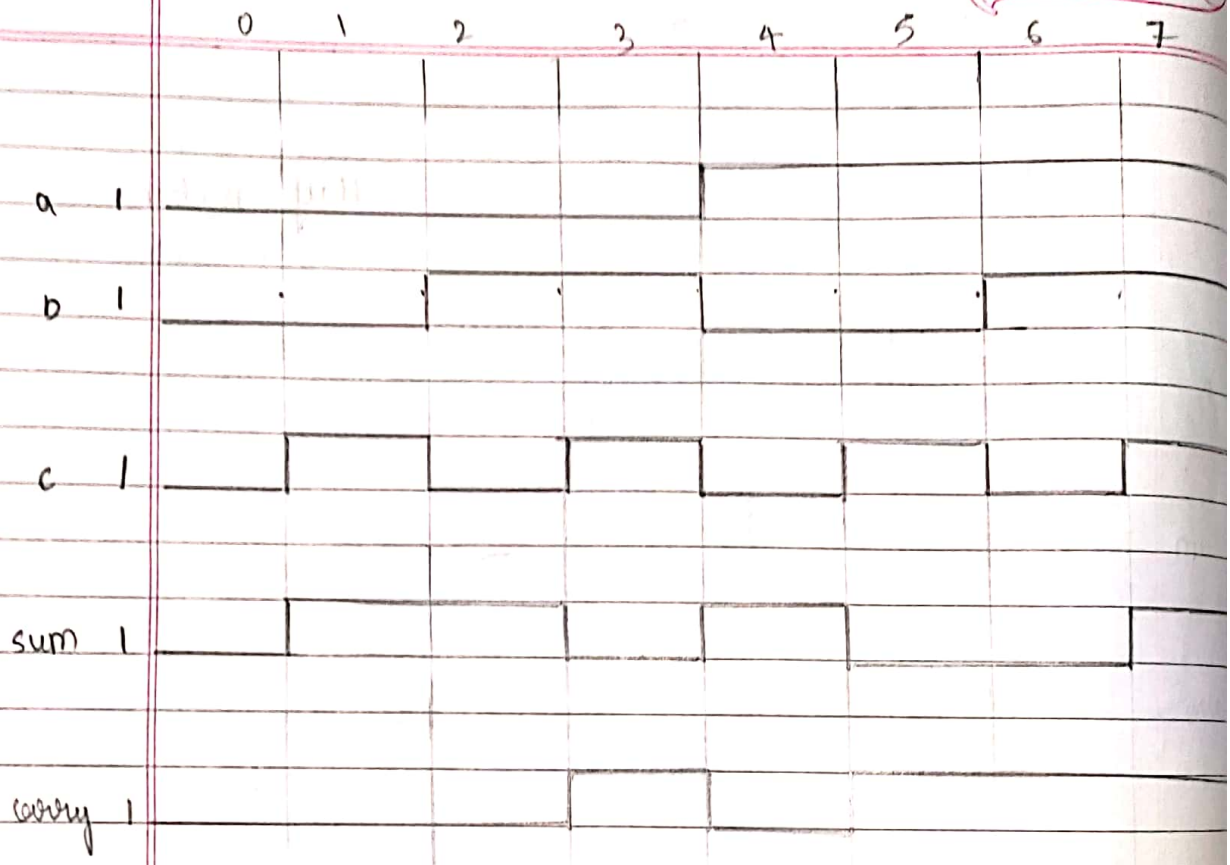
```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity adder is
    Port ( A : in std_logic;
           B : in std_logic;
           SUM : out std_logic;
           CARRY : out std_logic);

end adder;

architecture Behavioral of adder is
begin
  SUM <= A XOR B;
  CARRY <= A AND B;
      end Behavioral.
```

Full Adder



Code:

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

entity FADD is
    Port ( A : in std_logic;
           B : in std_logic;
           C : in std_logic;
           SUM : out std_logic;
           CARRY : out std_logic);
end FADD;
architecture Behavioral of FADD is
begin
    SUM <= A XOR B XOR C;
    CARRY <= (A AND B) OR (C AND (A XOR B
end Behavioral;
```