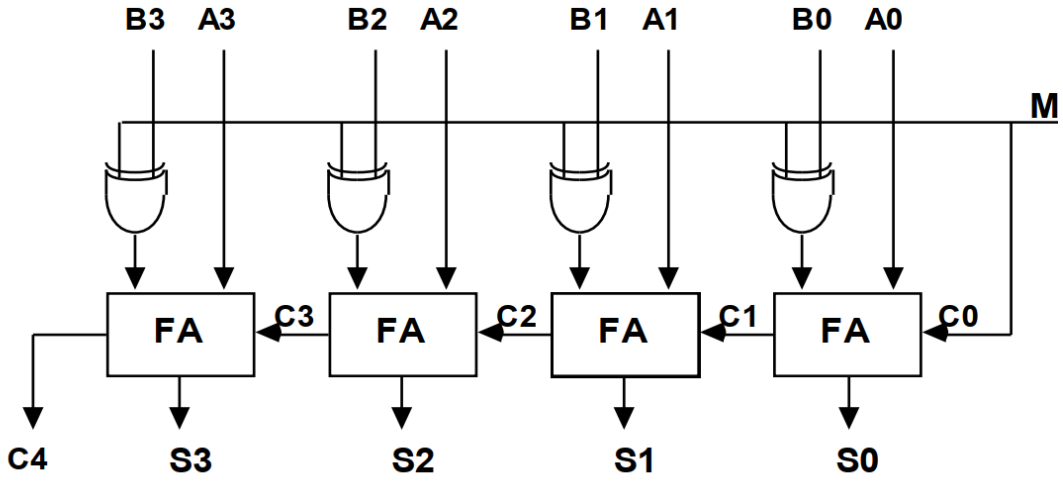
EXPERIMENT No-10

Aim: 2's Complement Adder-Subtractor using VHDL

Description:

- **4 -bit Adder-Subtractor:**



- For M=1, subtract, the 2's complement of B is formed by using XORs to form the 1's complement and adding the 1 applied to C0
- For M=0, add, B is passed through unchanged

VHDL program:

```
library ieee;
use ieee.std_logic_1164.all;
entity addsub is
  port( M: in std_logic;
        A,B  : in std_logic_vector(3 downto 0);
        S  : out std_logic_vector(3 downto 0);
        Cout, OVERFLOW : out std_logic);
end addsub;
```

```
architecture struct of addsub is
component Full_Adder is
  port( X, Y, Cin : in std_logic;
        sum, Cout : out std_logic);
end component;
signal C1, C2, C3, C4: std_logic;
signal TMP: std_logic_vector(3 downto 0);

begin

TMP(0)<= M xor B(0);
TMP(1)<= M xor B(1);
TMP(2)<= M xor B(2);
TMP(3)<= M xor B(3);
FA0:Full_Adder port map(A(0),TMP(0),M, S(0),C1);
FA1:Full_Adder port map(A(1),TMP(1),C1, S(1),C2);
FA2:Full_Adder port map(A(2),TMP(2),C2, S(2),C3);
FA3:Full_Adder port map(A(3),TMP(3),C3, S(3),C4);
OVERFLOW <= C3 XOR C4;
Cout <= C4;
end struct;
```

Waveforms/Outputs: