

Batch – 05

Day 5 TASK – 2

4*1 MUX VHDL Stimulation:

```
use IEEE.STD_LOGIC_1164.all;
```

```
entity mux_4to1 is
```

```
port(
```

```
    A,B,C,D : in STD_LOGIC;
```

```
    S0,S1: in STD_LOGIC;
```

```
    Z: out STD_LOGIC
```

```
);
```

```
end mux_4to1;
```

```
architecture bhv of mux_4to1 is
```

```
begin
```

```
process (A,B,C,D,S0,S1) is
```

```
begin
```

```
    if (S0 ='0' and S1 = '0') then
```

```
        Z <= A;
```

```
    elsif (S0 ='1' and S1 = '0') then
```

```
        Z <= B;
```

```
    elsif (S0 ='0' and S1 = '1') then
```

```
        Z <= C;
```

```
    else
```

```
        Z <= D;
```

```
    end if;
```

```
end process;
```

```
end bhv;
```

TestBench:

```
LIBRARY ieee;
```

```
USE ieee.std_logic_1164.ALL;
```

```
ENTITY tb_mux IS
```

```
END tb_mux;
```

```
ARCHITECTURE behavior OF tb_mux IS
```

```
-- Component Declaration for the Unit Under Test (UUT)
```

```
COMPONENT mux_4to1
```

```
PORT(
```

```
    A : IN std_logic;
```

```
    B : IN std_logic;
```

```
    C : IN std_logic;
```

```
    D : IN std_logic;
```

```
    S0 : IN std_logic;
```

```
    S1 : IN std_logic;
```

```
    Z : OUT std_logic
```

```
);
```

```
END COMPONENT;
```

```
--Inputs
```

```
signal A : std_logic := '0';  
signal B : std_logic := '0';  
signal C : std_logic := '0';  
signal D : std_logic := '0';  
signal S0 : std_logic := '0';  
signal S1 : std_logic := '0';
```

```
--Outputs
```

```
signal Z : std_logic;
```

```
BEGIN
```

```
-- Instantiate the Unit Under Test (UUT)
```

```
uut: mux_4to1 PORT MAP (
```

```
    A => A,
```

```
    B => B,
```

```
    C => C,
```

```
    D => D,
```

```
    S0 => S0,
```

```
    S1 => S1,
```

```
    Z => Z
```

```
);
```

```
-- Stimulus process
```

```
stim_proc: process
```

```
begin
```

```
    -- hold reset state for 100 ns.
```

```
    wait for 100 ns;
```

A <= '1';

B <= '0';

C <= '1';

D <= '0';

S0 <= '0'; S1 <= '0';

wait for 100 ns;

S0 <= '1'; S1 <= '0';

wait for 100 ns;

S0 <= '0'; S1 <= '1';

wait for 100 ns;

S0 <= '0'; S1 <= '1';

wait for 100 ns;

end process;

END;

