



16x4 Encoder

VLSI Systems

Assignment-2 Annexure-I

PREPARED BY

Priyank Lohariwal

BCSE-IV

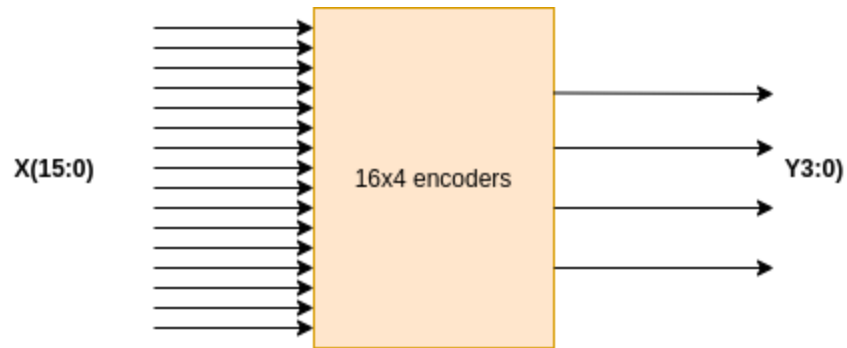
001710501055

Jadavpur University

Description

Designing 16x4 encoder using 4x2 encoders with generate statements.

Block Diagram

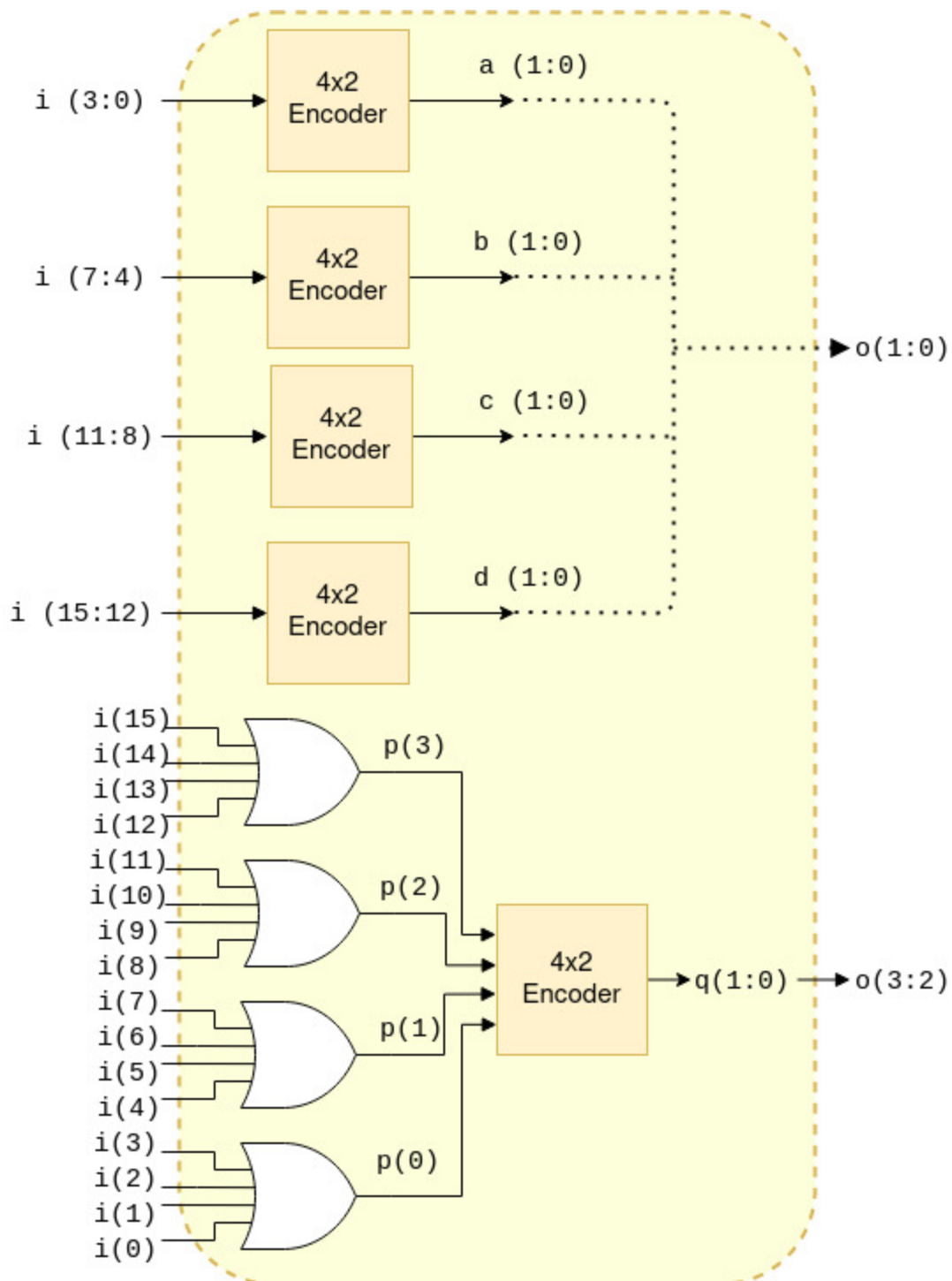


16x4 Encoder

Truth Table

X(15:0)																Y(3:0)			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	z	z	z	z
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	1
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	1
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1

Circuit Diagram



Code

```
entity ass2_annex1 is
    Port ( X : in  STD_LOGIC_VECTOR (15 downto 0);
           Y : out STD_LOGIC_VECTOR (3 downto 0));
end ass2_annex1;

architecture Behavioral of ass2_annex1 is
    component ass2_2ba is
        port( X: in STD_LOGIC_VECTOR (3 downto 0);
              Y: out STD_LOGIC_VECTOR(1 downto 0));
    end component;
    signal a: STD_LOGIC_VECTOR(7 downto 0);
    signal q: STD_LOGIC_VECTOR(1 downto 0);
    signal p: STD_LOGIC_VECTOR(3 downto 0);
begin
    c5: ass2_2ba port map(p, q);

    p(3) <= X(15) or X(14) or X(13) or X(12);
    p(2) <= X(11) or X(10) or X(9) or X(8);
    p(1) <= X(7) or X(6) or X(5) or X(4);
    p(0) <= X(3) or X(2) or X(1) or X(0);

    gen: for k in 0 to 3 generate
        c1: ass2_2ba port map(X(4*k+3 downto 4*k), a(2*k+1 downto 2*k));
    end generate;

    p1: process(X, a, q, p)
    begin
        if q = "ZZ" then
            Y <= "ZZZZ";
        else
            for k in 0 to 3 loop
                if a(2*k+1 downto 2*k) /= "ZZ" then
                    Y <= q & a(2*k+1 downto 2*k);
                end if;
            end loop;
        end if;
    end process;

end Behavioral;
```

Test Bench

```
ARCHITECTURE behavior OF ass2_annex1_test_bench IS
    -- Component Declaration for the Unit Under Test (UUT)
    COMPONENT ass2_annex1
    PORT (
        X : IN  std_logic_vector(15 downto 0);
        Y : OUT std_logic_vector(3  downto 0)
    );
    END COMPONENT;

    --Inputs
    signal X : std_logic_vector(15 downto 0) := (others => '0');

    --Outputs
    signal Y : std_logic_vector(3  downto 0);
BEGIN
    -- Instantiate the Unit Under Test (UUT)
    uut: ass2_annex1 PORT MAP (
        X => X,
        Y => Y
    );

    -- Stimulus process
    stim_proc: process
    begin
        X <= "0000000000000000";
        wait for 1 ps;
        for i in 0 to 15 loop
            X(i) <= '1';
            wait for 1ps;
            X(i) <= '0';
        end loop;
    end process;

END;
```

Timing Diagram

