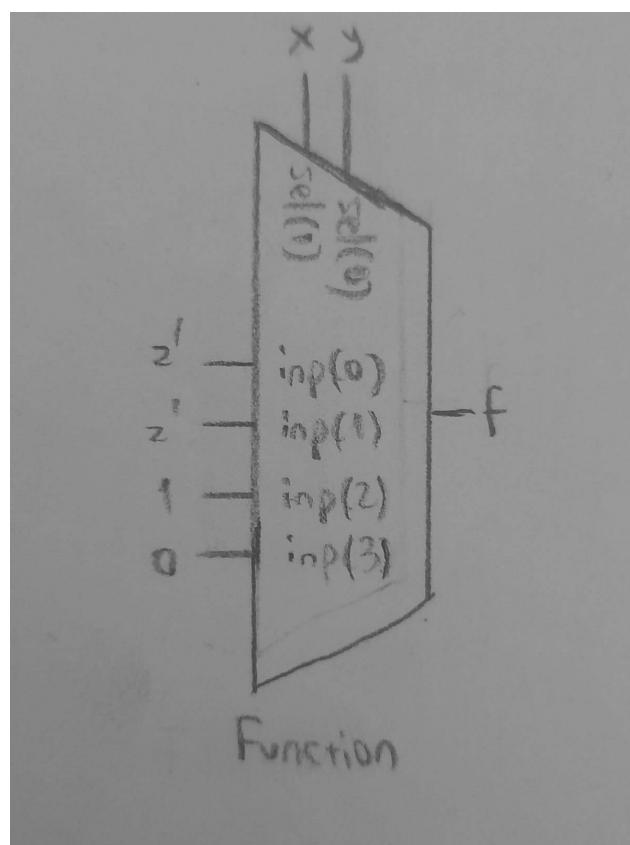
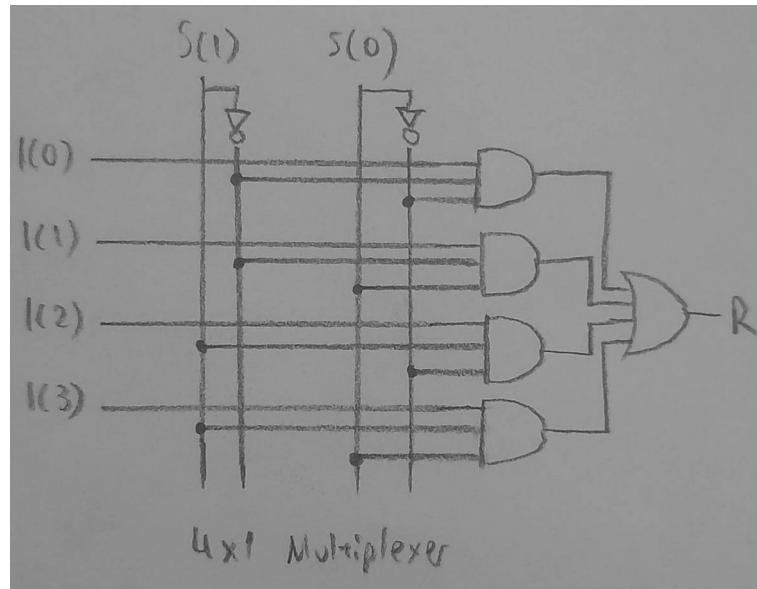
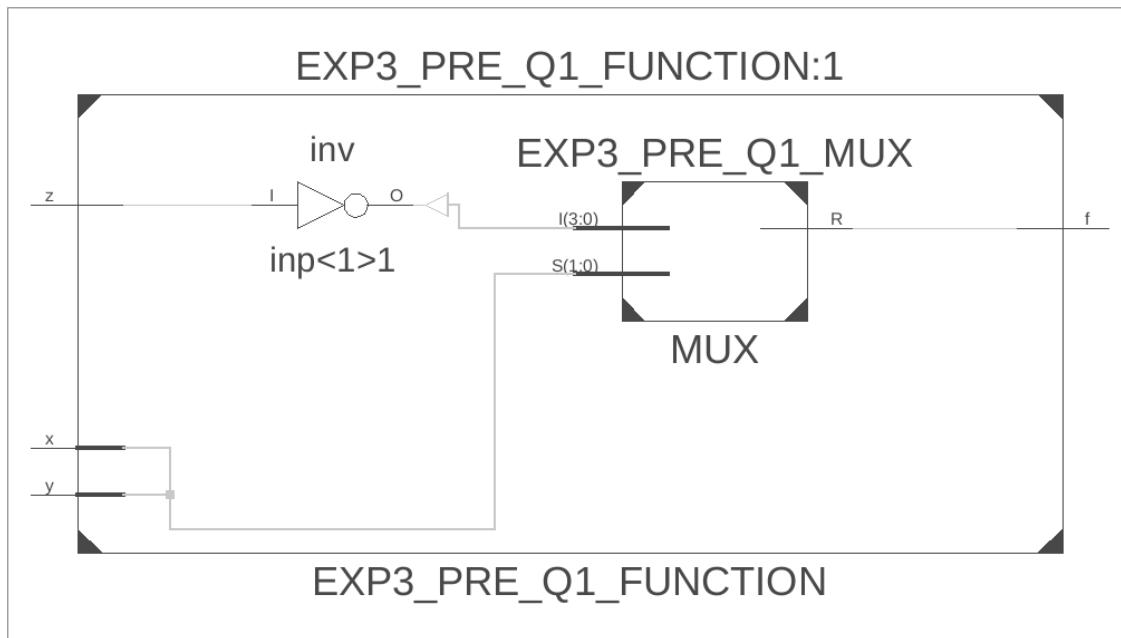
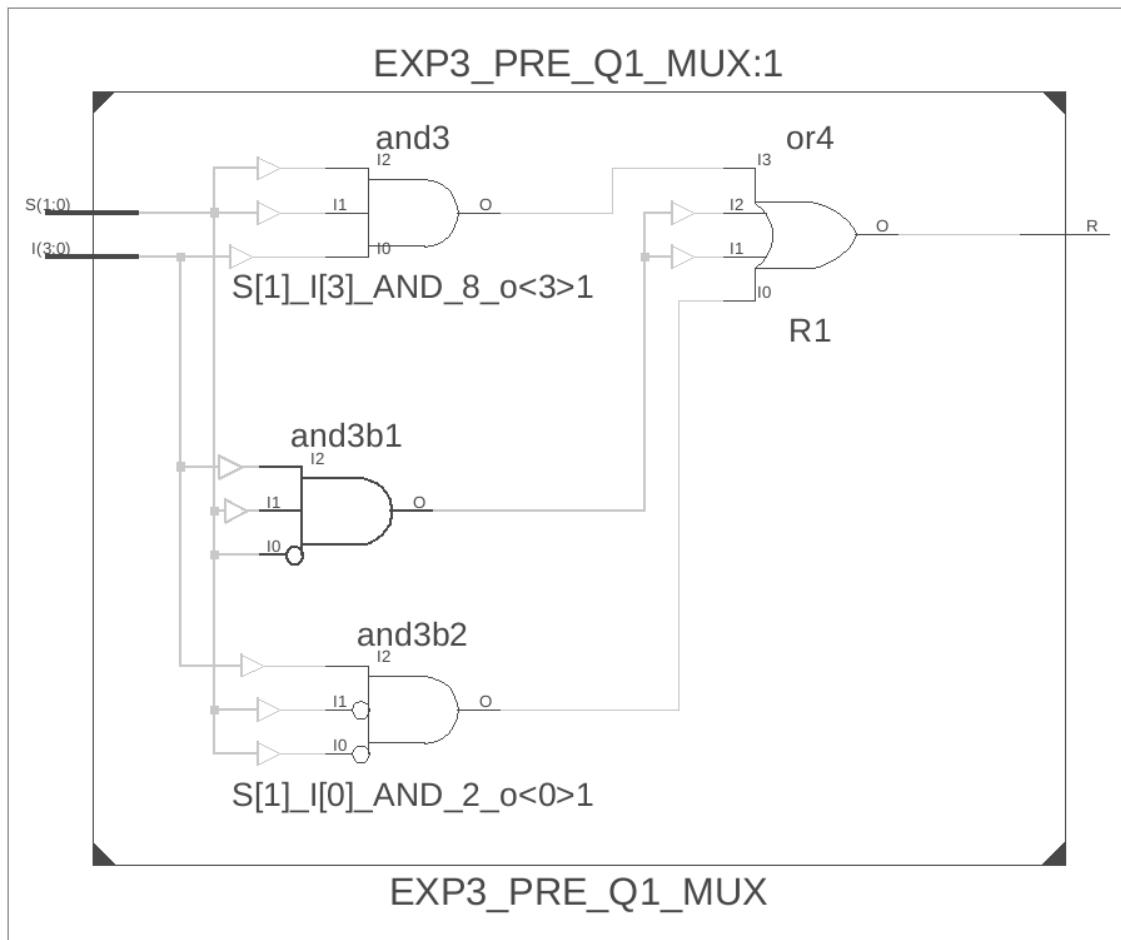


EXPERIMENT 3 PRELIMINARY WORK

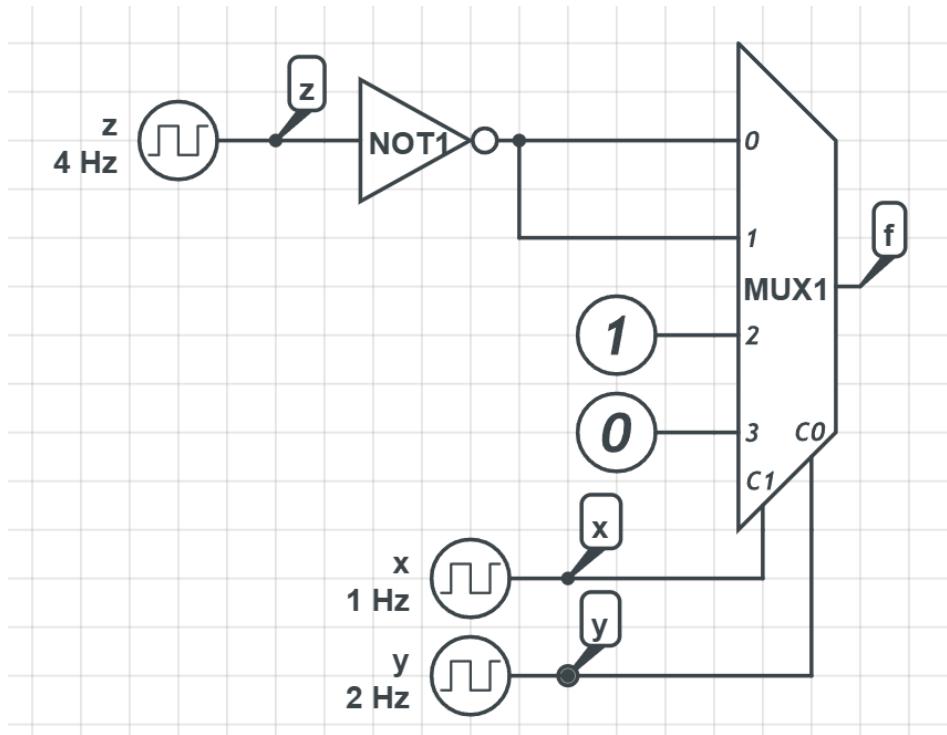
Q1: 1. Analytical Solution

Hand-drawn circuits and RTL schematics

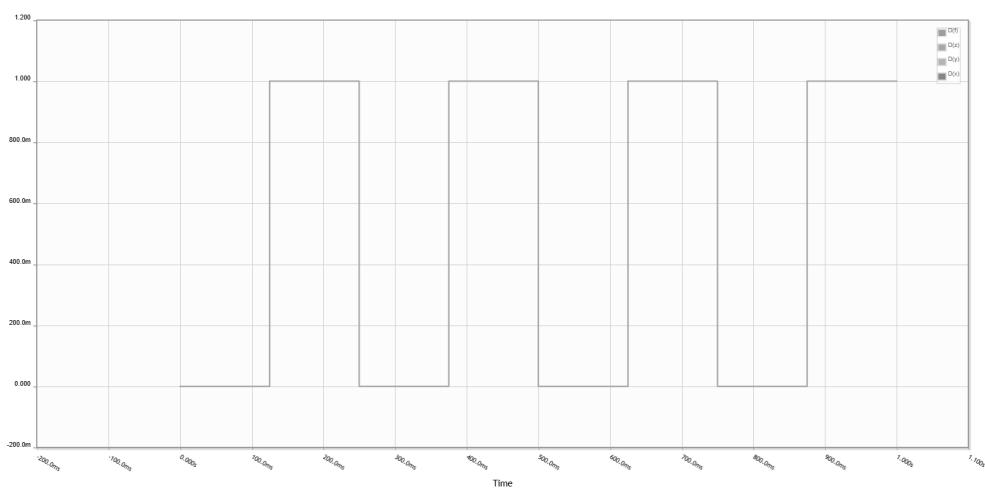




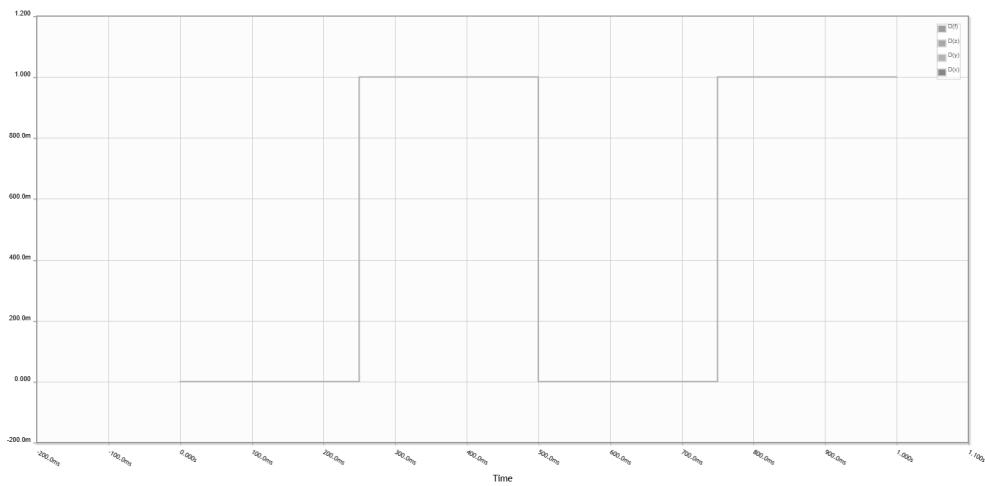
CircuitLab



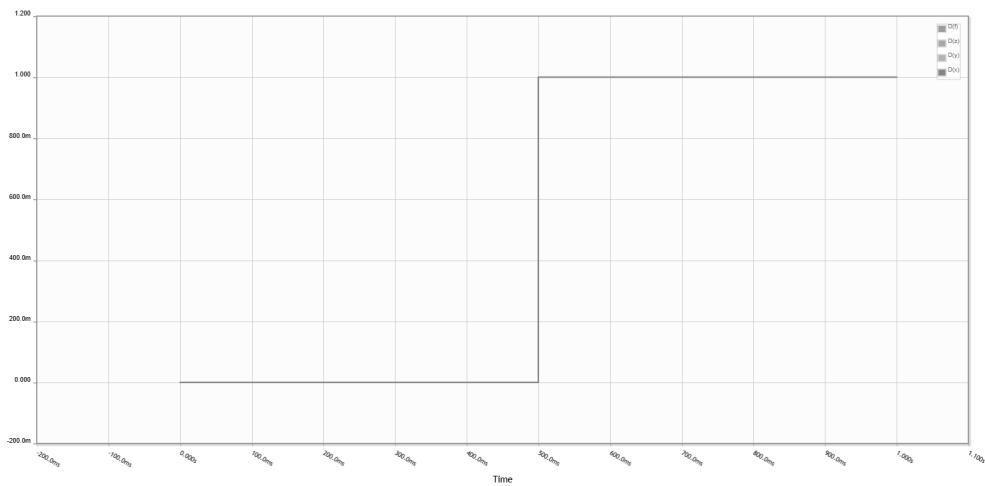
Graph of Input x



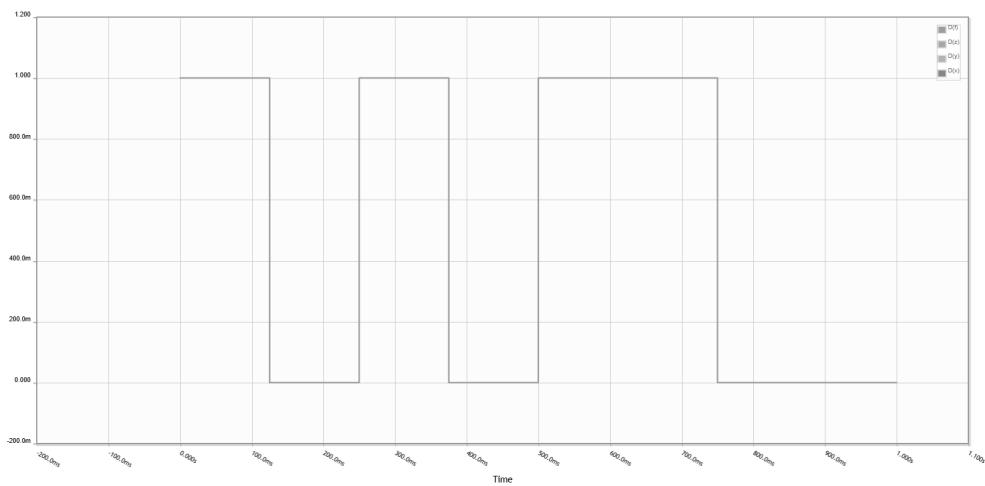
Graph of Input y



Graph of Input z



Graph of Output f



2. Codes

VHDL - MUX

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity EXP3_PRE_Q1_MUX is
    Port ( I : in STD_LOGIC_VECTOR (3 downto 0);
           S : in STD_LOGIC_VECTOR (1 downto 0);
           R : out STD_LOGIC);
end EXP3_PRE_Q1_MUX;

architecture Behavioral of EXP3_PRE_Q1_MUX is

begin
    R <= (not S(1) and not S(0) and I(0)) or
          (not S(1) and S(0) and I(1)) or
          (S(1) and not S(0) and I(2)) or
          (S(1) and S(0) and I(3));
end Behavioral;

```

VHDL - Function

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity EXP3_PRE_Q1_FUNCTION is
    Port ( x, y, z : in STD_LOGIC;
           f : out STD_LOGIC);
end EXP3_PRE_Q1_FUNCTION;

architecture Behavioral of EXP3_PRE_Q1_FUNCTION is

signal inp : STD_LOGIC_VECTOR (3 downto 0);
signal sel : STD_LOGIC_VECTOR (1 downto 0);

component EXP3_PRE_Q1_MUX
    Port ( I : in STD_LOGIC_VECTOR (3 downto 0);
           S : in STD_LOGIC_VECTOR (1 downto 0);
           R : out STD_LOGIC);
end component;

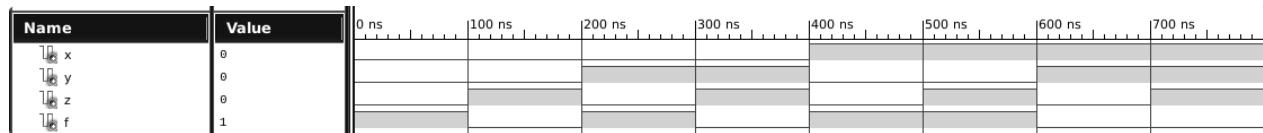
begin
    sel(0) <= y; sel(1) <= x;
    inp(0) <= not z; inp(1) <= not z; inp(2) <= '1'; inp(3) <= '0';
    MUX : EXP3_PRE_Q1_MUX port map (inp, sel, f);
end Behavioral;

```

3. Results

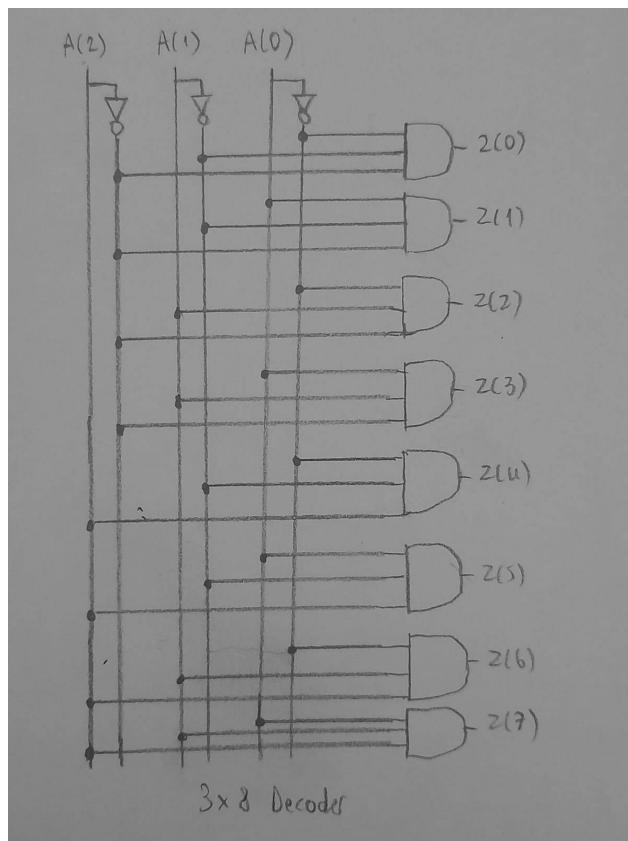
Test bench

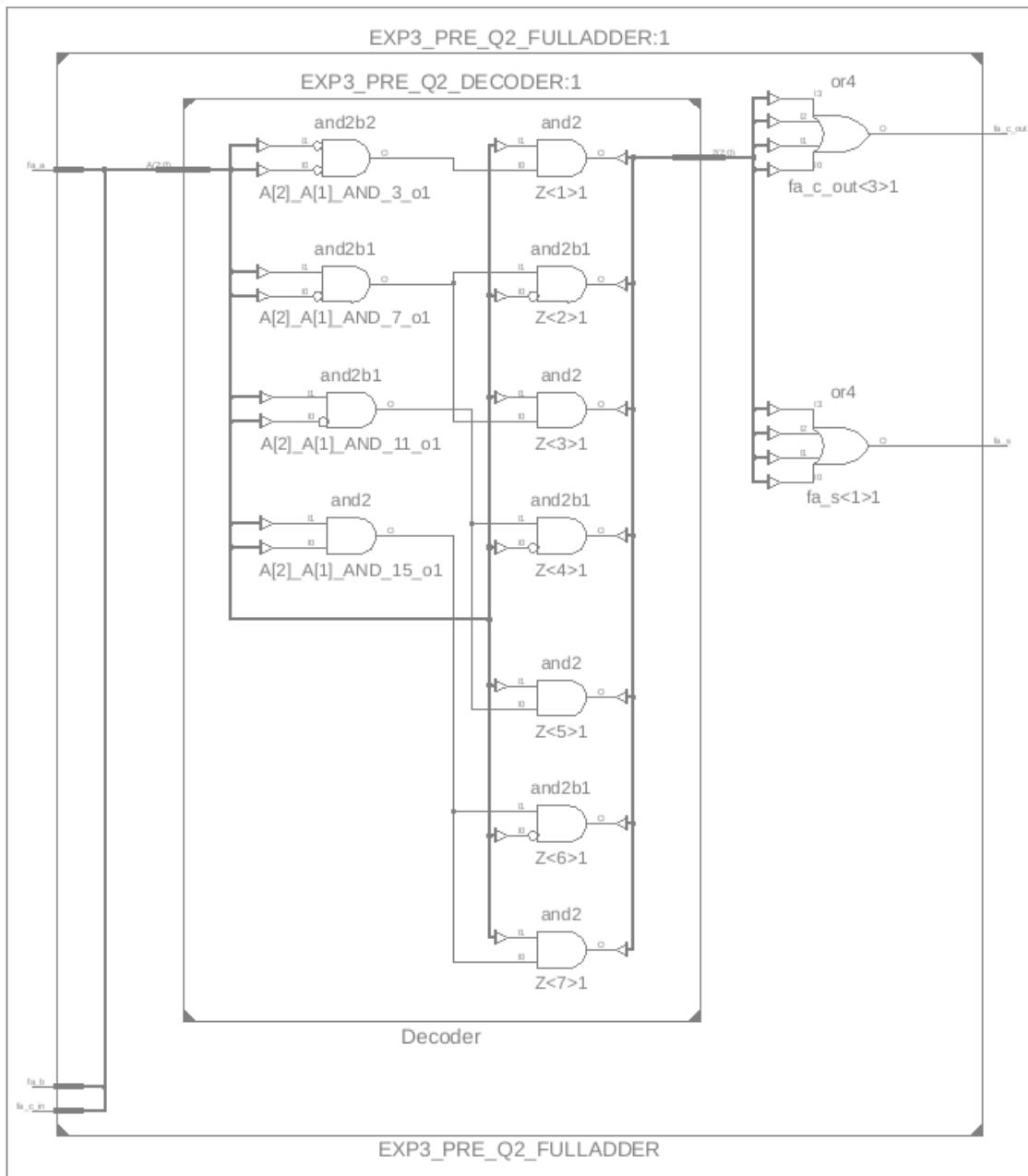
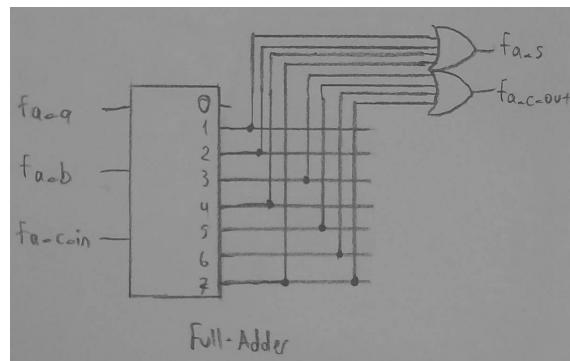
```
stim_proc: process
begin
    x <= '0'; y <= '0'; z <= '0'; wait for 100 ns;
    x <= '0'; y <= '0'; z <= '1'; wait for 100 ns;
    x <= '0'; y <= '1'; z <= '0'; wait for 100 ns;
    x <= '0'; y <= '1'; z <= '1'; wait for 100 ns;
    x <= '1'; y <= '0'; z <= '0'; wait for 100 ns;
    x <= '1'; y <= '0'; z <= '1'; wait for 100 ns;
    x <= '1'; y <= '1'; z <= '0'; wait for 100 ns;
    x <= '1'; y <= '1'; z <= '1'; wait for 100 ns;
end process;
```



Q2: 1. Analytical Solution

Hand-drawn circuits and RTL schematic





2. Codes

VHDL - Decoder

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity EXP3_PRE_Q2_DECODER is
    Port ( A : in STD_LOGIC_VECTOR (2 downto 0);
           Z : out STD_LOGIC_VECTOR (7 downto 0));
end EXP3_PRE_Q2_DECODER;

architecture Behavioral of EXP3_PRE_Q2_DECODER is
begin
    Z(0) <= not A(2) and not A(1) and not A(0);
    Z(1) <= not A(2) and not A(1) and A(0);
    Z(2) <= not A(2) and A(1) and not A(0);
    Z(3) <= not A(2) and A(1) and A(0);
    Z(4) <= A(2) and not A(1) and not A(0);
    Z(5) <= A(2) and not A(1) and A(0);
    Z(6) <= A(2) and A(1) and not A(0);
    Z(7) <= A(2) and A(1) and A(0);
end Behavioral;

```

VHDL - Full Adder

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity EXP3_PRE_Q2_FULLADDER is
    Port (fa_a, fa_b, fa_c_in : in STD_LOGIC;
          fa_s, fa_c_out : out STD_LOGIC);
end EXP3_PRE_Q2_FULLADDER;

architecture Behavioral of EXP3_PRE_Q2_FULLADDER is
signal res : STD_LOGIC_VECTOR (7 downto 0);
signal inp : STD_LOGIC_VECTOR (2 downto 0);
component EXP3_PRE_Q2_DECODER is
    Port ( A : in STD_LOGIC_VECTOR (2 downto 0);
           Z : out STD_LOGIC_VECTOR (7 downto 0));
end component;
begin
    inp <= fa_a & fa_b & fa_c_in;
    Decoder : EXP3_PRE_Q2_DECODER port map(A => inp, Z => res);

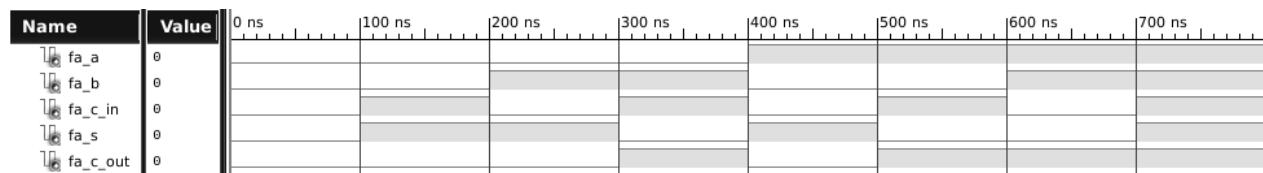
    fa_s <= res(1) or res(2) or res(4) or res(7);
    fa_c_out <= res(3) or res(5) or res(6) or res(7);
end Behavioral;

```

3. Results

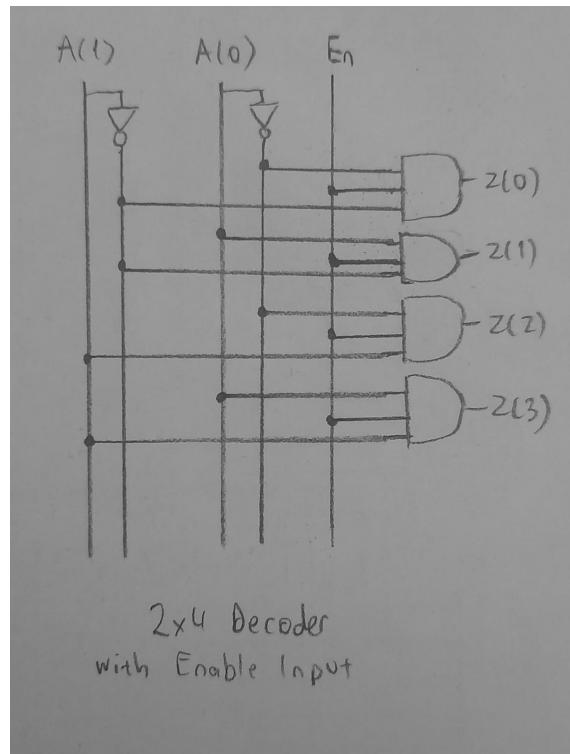
Test bench - Full Adder

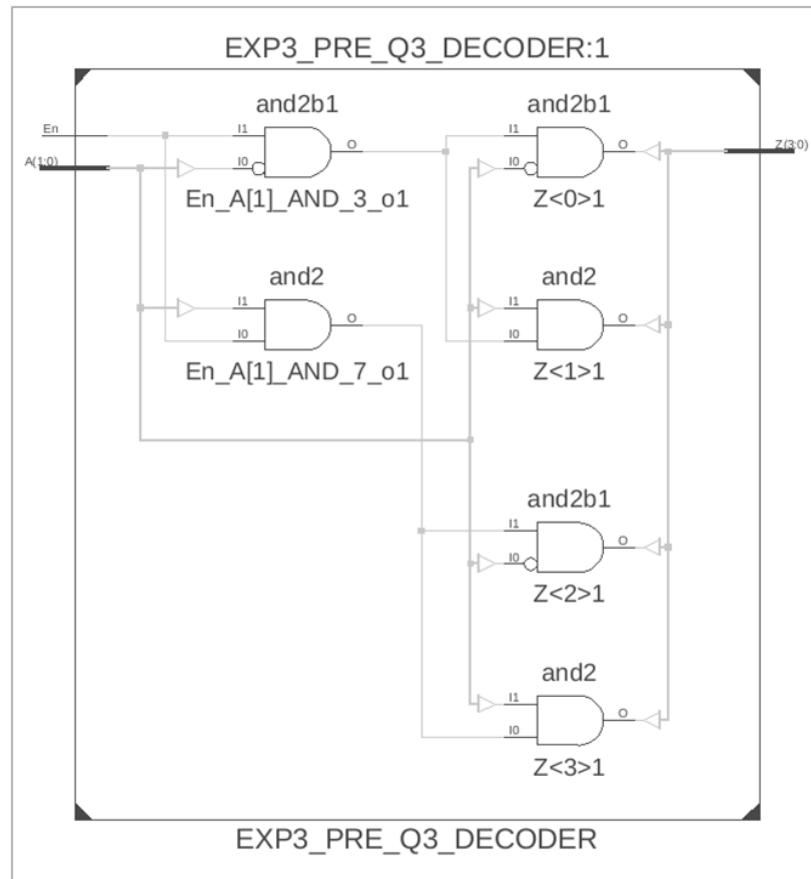
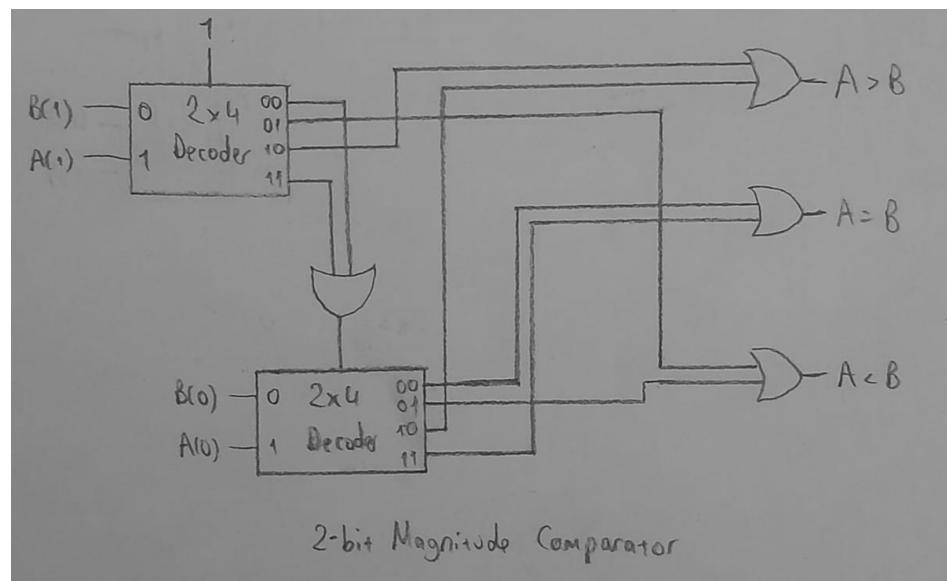
```
stim_proc: process
begin
    fa_a <= '0'; fa_b <= '0'; fa_c_in <= '0'; wait for 100 ns;
    fa_a <= '0'; fa_b <= '0'; fa_c_in <= '1'; wait for 100 ns;
    fa_a <= '0'; fa_b <= '1'; fa_c_in <= '0'; wait for 100 ns;
    fa_a <= '0'; fa_b <= '1'; fa_c_in <= '1'; wait for 100 ns;
    fa_a <= '1'; fa_b <= '0'; fa_c_in <= '0'; wait for 100 ns;
    fa_a <= '1'; fa_b <= '0'; fa_c_in <= '1'; wait for 100 ns;
    fa_a <= '1'; fa_b <= '1'; fa_c_in <= '0'; wait for 100 ns;
    fa_a <= '1'; fa_b <= '1'; fa_c_in <= '1';
    wait;
end process;
```

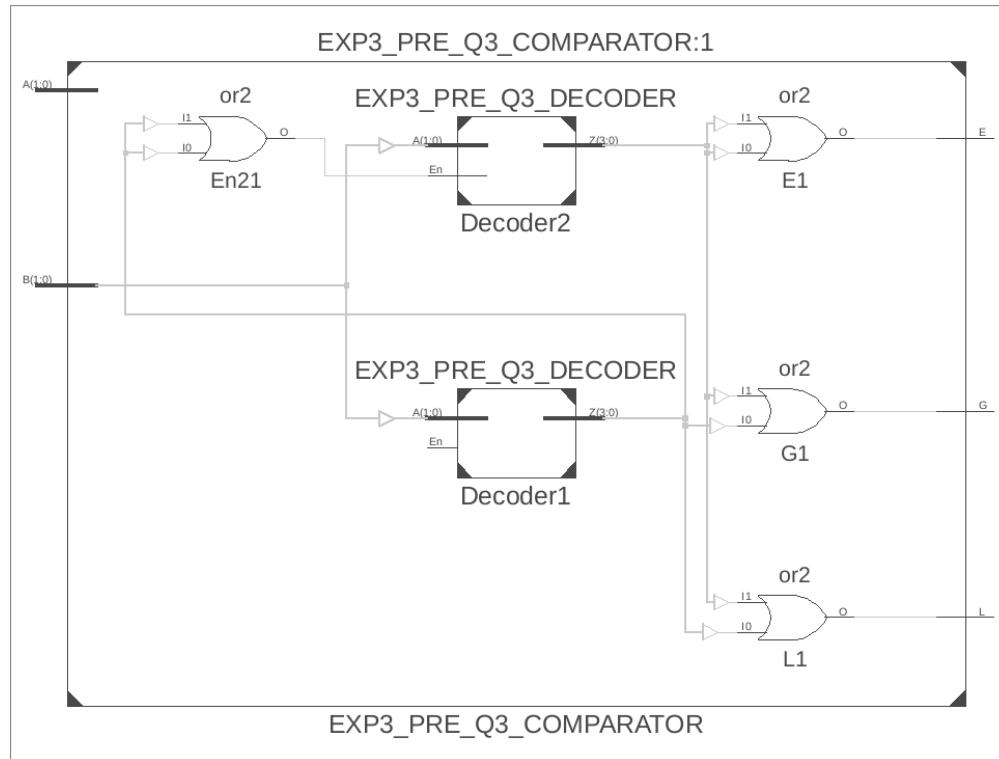


Q3: 1. Analytical Solution

Hand-drawn circuits and RTL schematics







2. Codes

VHDL - Decoder

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity EXP3_PRE_Q3_DECODER is
    Port ( A : in STD_LOGIC_VECTOR (1 downto 0); En : in STD_LOGIC;
           Z : out STD_LOGIC_VECTOR (3 downto 0));
end EXP3_PRE_Q3_DECODER;
architecture Behavioral of EXP3_PRE_Q3_DECODER is
begin
    Z(0) <= En and not A(1) and not A(0);
    Z(1) <= En and not A(1) and A(0);
    Z(2) <= En and A(1) and not A(0);
    Z(3) <= En and A(1) and A(0);
end Behavioral;

```

VHDL - Comparator

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity EXP3_PRE_Q3_COMPARATOR is
    Port ( A, B : in STD_LOGIC_VECTOR (1 downto 0);
           G, E, L : out STD_LOGIC);
end EXP3_PRE_Q3_COMPARATOR;

```

```

architecture Behavioral of EXP3_PRE_Q3_COMPARATOR is
component EXP3_PRE_Q3_DECODER
    Port ( A : in STD_LOGIC_VECTOR (1 downto 0); En : in STD_LOGIC;
            Z : out STD_LOGIC_VECTOR (3 downto 0));
end component;
signal D1_in, D2_in : STD_LOGIC_VECTOR(1 downto 0);
signal D1_out, D2_out : STD_LOGIC_VECTOR (3 downto 0);
signal En2 : STD_LOGIC;
begin
    D1_in <= A(1) & B(1); D2_in <= A(0) & B(0);
    Decoder1 : EXP3_PRE_Q3_DECODER port map(D1_in, '1', D1_out);
    En2 <= D1_out(0) or D1_out(3);
    Decoder2 : EXP3_PRE_Q3_DECODER port map(D2_in, En2, D2_out);
    G <= D1_out(2) or D2_out(2);
    E <= D2_out(0) or D2_out(3);
    L <= D1_out(1) or D2_out(1);
end Behavioral;

```

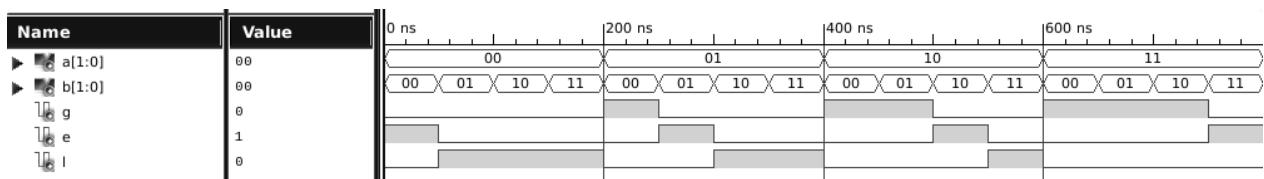
3. Results

Test bench

```

stim_proc: process
begin
    a <= "00"; b <= "00"; wait for 50 ns;
    a <= "00"; b <= "01"; wait for 50 ns;
    a <= "00"; b <= "10"; wait for 50 ns;
    a <= "00"; b <= "11"; wait for 50 ns;
    a <= "01"; b <= "00"; wait for 50 ns;
    a <= "01"; b <= "01"; wait for 50 ns;
    a <= "01"; b <= "10"; wait for 50 ns;
    a <= "01"; b <= "11"; wait for 50 ns;
    a <= "10"; b <= "00"; wait for 50 ns;
    a <= "10"; b <= "01"; wait for 50 ns;
    a <= "10"; b <= "10"; wait for 50 ns;
    a <= "10"; b <= "11"; wait for 50 ns;
    a <= "11"; b <= "00"; wait for 50 ns;
    a <= "11"; b <= "01"; wait for 50 ns;
    a <= "11"; b <= "10"; wait for 50 ns;
    a <= "11"; b <= "11"; wait for 50 ns;
end process;

```



Q4: 1. Analytical Solution

By looking at the truth table, we could directly say that $B(7) = 0$, $B(0) = A(0)$.

Set up 4 k-maps for $B(1)$.

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	1	1
	01	0	0	1	1
11	1	1	0	0	
10	0	0	0	0	

$$\begin{array}{l} A(5)=0, A(4)=0 \\ (\overline{A(2)} + A(2))(\overline{A(3)}A(1)) + A(3)A(2)\overline{A(1)} \end{array}$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	1	1	0	0
	01	0	0	1	1
11	0	0	0	0	
10	0	0	1	1	

$$\begin{array}{l} A(5)=0, A(4)=1 \\ \overline{A(3)}\overline{A(2)}\overline{A(1)} + A(3)A(2)A(1) + A(3)\overline{A(2)}A(1) \end{array}$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	1	1	0	0
	01	1	1	0	0
11	0	0	1	1	
10	0	0	1	1	

$$\begin{array}{l} A(5)=1, A(4)=0 \\ (\overline{A(2)} + A(2))\overline{A(3)}\overline{A(1)} + (\overline{A(2)} + A(2))A(3)A(1) \end{array}$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	0	0
	01	1	1	0	0
11	0	0	1	1	
10	1	1	0	0	

$$\begin{array}{l} A(5)=1, A(4)=1 \\ \overline{A(3)}\overline{A(2)}\overline{A(1)} + A(3)A(2)A(1) + A(3)\overline{A(2)}\overline{A(1)} \end{array}$$

The minimized expression for $B(1)$ is

$$\begin{aligned} B(1) = & (A(5) \oplus A(4))\overline{A(3)}\overline{A(2)}\overline{A(1)} + \overline{A(5)}\overline{A(4)}\overline{A(3)}\overline{A(2)}A(1) + A(5)\overline{A(3)}A(2)\overline{A(1)} \\ & + \overline{A(5)}\overline{A(3)}A(2)A(1) + \overline{A(5)}\overline{A(4)}A(3)A(2)\overline{A(1)} + A(5)A(3)A(2)A(1) \\ & + A(5)A(4)A(3)\overline{A(2)}\overline{A(1)} + (A(5) \oplus A(4))A(3)\overline{A(2)}A(1) \end{aligned}$$

Set up 4 k-maps for $B(2)$.

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	0	0
	01	1	1	1	1
11	0	0	1	1	
10	0	0	0	0	

$$\begin{array}{l} A(5)=0, A(4)=0 \\ A(2)(A(1) \oplus A(3)) + A(3)A(2)A(1) \end{array}$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	1	1	0	0
	01	0	0	0	0
11	0	0	0	0	
10	1	1	1	1	

$$\begin{array}{l} A(5)=0, A(4)=1 \\ \overline{A(3)}\overline{A(2)}\overline{A(1)} + A(3)A(2)A(1) + A(3)\overline{A(2)}A(1) \end{array}$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	1	1
	01	1	1	0	0
11	1	1	1	1	
10	0	0	0	0	

$$\overline{A(2) \overline{A(3)} A(1)} + A(3)A(2)A(1) + A(2)(\overline{A(1)} \oplus A(3))$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	0	0
	01	0	0	1	1
11	0	0	0	0	
10	1	1	0	0	

$$\overline{A(3)A(2) A(1)} + A(3)A(2)A(1)$$

The minimized expression for B(2) is

$$\begin{aligned} B(2) = & \overline{A(5)A(4)\overline{A(3)}\overline{A(2)}\overline{A(1)}} + A(5)\overline{A(4)}\overline{A(3)}\overline{A(2)}A(1) + (\overline{A(5)} \oplus A(4))\overline{A(3)}A(2)A(1) \\ & + A(5)\overline{A(4)}A(3)A(2)\overline{A(1)} + \overline{A(4)}(\overline{A(1)} \oplus A(3))A(2) + A(4)A(3)\overline{A(2)}\overline{A(1)} \\ & + \overline{A(5)}A(4)A(3)\overline{A(2)}A(1) \end{aligned}$$

Set up 4 k-maps for B(3).

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	0	0
	01	0	0	0	0
11	0	0	0	0	
10	1	1	0	0	

$$\overline{A(3)A(2) A(1)}$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	1	1
	01	0	0	0	0
11	1	1	0	0	
10	0	0	0	0	

$$\overline{A(3) \overline{A(2)} A(1)} + A(3)A(2)\overline{A(1)}$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	0	0
	01	0	0	1	1
11	0	0	0	0	
10	0	0	0	0	

$$\overline{A(3)A(2)A(1)}$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	1	1	0	0
	01	0	0	0	0
11	0	0	0	0	
10	0	0	1	1	

$$\overline{A(3) \overline{A(2)} A(1)} + A(3)A(2)\overline{A(1)}$$

The minimized expression for B(3) is

$$\begin{aligned} B(3) = & \overline{A(5)} \overline{A(4)} A(3) \overline{A(2)} \overline{A(1)} + \overline{A(5)} A(4) (\overline{A(3)} \overline{A(2)} A(1) + A(3) A(2) \overline{A(1)}) \\ & + A(5) \overline{A(4)} \overline{A(3)} A(2) A(1) + A(5) A(4) (\overline{A(3)} \oplus A(1)) \overline{A(2)} \end{aligned}$$

Set up 4 k-maps for B(4).

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	0	0
	01	0	0	0	0
11	1	1	1	1	
10	0	0	1	1	

$$\overline{A(3)} \overline{A(2)} \overline{A(1)} + A(3) A(2) A(1) + A(3) \overline{A(2)} A(1)$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	1	1	1	1
	01	0	0	0	0
11	0	0	1	1	
10	0	0	0	0	

$$\overline{A(3)} \overline{A(2)} \overline{A(1)} + \overline{A(2)} A(3) A(1) + A(3) A(2) A(1)$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	1	1	1	1
	01	1	1	1	1
11	0	0	0	0	
10	0	0	0	0	

$$\overline{A(3)} \overline{A(2)} \overline{A(1)} + \overline{A(2)} A(3) A(1) + \overline{A(3)} A(2)$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	1	1
	01	1	1	1	1
11	0	0	0	0	
10	1	1	1	1	

$$\overline{A(2)} \overline{A(3)} A(1) + \overline{A(3)} A(2) + A(3) \overline{A(2)} \overline{A(1)} + A(3) \overline{A(2)} A(1)$$

The minimized expression for B(4) is

$$\begin{aligned} B(4) = & (A(5) \oplus A(4)) (\overline{A(3)} \overline{A(2)} \overline{A(1)}) + A(5) \overline{A(3)} A(2) + \overline{A(5)} \overline{A(4)} A(3) A(2) \overline{A(1)} \\ & + \overline{A(5)} A(3) A(2) A(1) + A(5) A(4) A(3) \overline{A(2)} \overline{A(1)} + (\overline{A(5)} \oplus A(4)) A(3) \overline{A(2)} A(1) \\ & + (A(5) + A(4)) \overline{A(3)} \overline{A(2)} A(1) \end{aligned}$$

Set up 4 k-maps for B(5).

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	0	0
	01	0	0	0	0
	11	0	0	0	0
	10	0	0	0	0

$$\begin{array}{c} A(5)=0, A(4)=0 \\ 0 \end{array}$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	0	0
	01	1	1	1	1
	11	1	1	1	1
	10	1	1	1	1

$$\begin{array}{c} A(5)=0, A(4)=1 \\ \overline{A(3)}A(2) + A(3)A(2) + \overline{A(3)}A(2) \end{array}$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	1	1	1	1
	01	1	1	1	1
	11	0	0	0	0
	10	0	0	0	0

$$\begin{array}{c} A(5)=1, A(4)=0 \\ \overline{A(3)}A(2) + A(3)A(2) \end{array}$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	0	0
	01	0	0	0	0
	11	1	1	1	1
	10	0	0	0	0

$$\begin{array}{c} A(5)=1, A(4)=1 \\ A(3)A(2) \end{array}$$

The minimized expression for B(5) is

$$B(5) = A(5)\overline{A(4)}\overline{A(3)}\overline{A(2)} + (A(5) \oplus A(4))\overline{A(3)}A(2) + A(4)A(3)A(2) + \overline{A(5)}A(4)A(3)\overline{A(2)}$$

Set up 4 k-maps for B(6).

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	0	0
	01	0	0	0	0
	11	0	0	0	0
	10	0	0	0	0

$$\begin{array}{c} A(5)=0, A(4)=0 \\ 0 \end{array}$$

		A(1)A(0)			
		00	01	11	10
A(3)A(2)	00	0	0	0	0
	01	0	0	0	0
	11	0	0	0	0
	10	0	0	0	0

$$\begin{array}{c} A(5)=0, A(4)=1 \\ 0 \end{array}$$

		A(1)A(0)	00	01	11	10	
		A(3)A(2)	00	0	0	0	0
		01	1	1	1	1	
		11	1	1	1	1	
		10	1	1	1	1	

$$\begin{aligned} A(5) &= 1, A(4) = 0 \\ A(3) \end{aligned}$$

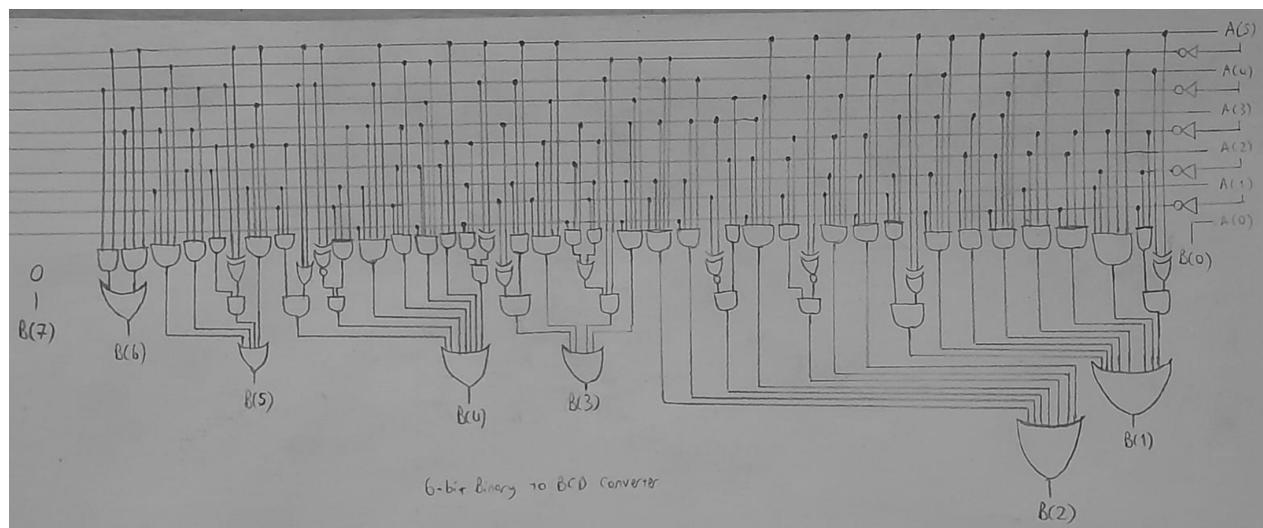
		A(1)A(0)	00	01	11	10	
		A(3)A(2)	00	1	1	1	1
		01	1	1	1	1	
		11	1	1	1	1	
		10	1	1	1	1	

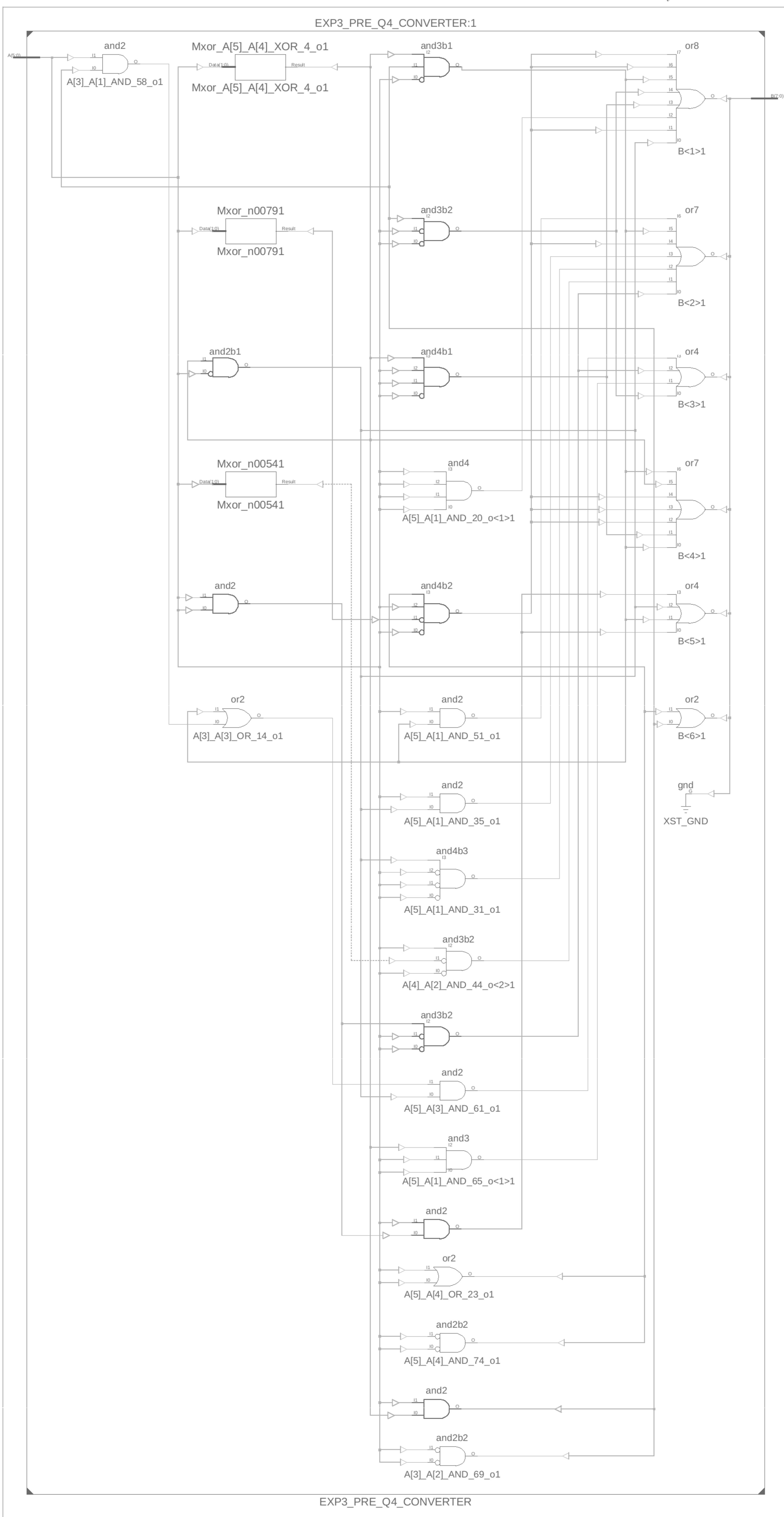
$$\begin{aligned} A(5) &= 1, A(4) = 1 \\ A(3) \end{aligned}$$

The minimized expression for B(6) is

$$B(6) = A(5)\overline{A(4)}A(3) + A(5)A(4)$$

Hand-drawn circuit and RTL schematic





EXP3_PRE_Q4_CONVERTER

2. Codes

VHDL

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity EXP3_PRE_Q4_CONVERTER is
    Port ( A : in STD_LOGIC_VECTOR (5 downto 0);
            B : out STD_LOGIC_VECTOR (7 downto 0));
end EXP3_PRE_Q4_CONVERTER;
architecture Behavioral of EXP3_PRE_Q4_CONVERTER is
begin
    B(0) <= A(0);
    B(1) <= ((A(5) xor A(4)) and not A(3) and not A(2) and not A(1))
              or (not A(5) and not A(4) and not A(3) and not A(2) and A(1))
              or (A(5) and not A(3) and A(2) and not A(1))
              or (not A(5) and not A(3) and A(2) and A(1))
              or (not A(5) and not A(4) and A(3) and A(2) and not A(1))
              or (A(5) and A(3) and A(2) and A(1))
              or (A(5) and A(4) and A(3) and not A(2) and not A(1))
              or ((A(5) xor A(4)) and A(3) and not A(2) and A(1));
    B(2) <= (not A(5) and A(4) and not A(3) and not A(2) and not A(1))
              or (A(5) and not A(4) and not A(3) and not A(2) and A(1))
              or ((A(5) xnor A(4)) and not A(3) and A(2) and A(1))
              or (A(5) and not A(4) and A(3) and A(2) and not A(1))
              or (not A(4) and (A(3) xnor A(1)) and A(2))
              or (A(4) and A(3) and not A(2) and not A(1))
              or (not A(5) and A(4) and A(3) and not A(2) and A(1));
    B(3) <= (not A(5) and not A(4) and A(3) and not A(2) and not A(1))
              or (not A(5) and A(4) and ((not A(3) and not A(2) and A(1))
                                         or (A(3) and A(2) and not A(1))))
              or (A(5) and not A(4) and not A(3) and A(2) and A(1))
              or (A(5) and A(4) and (A(3) xnor A(1)) and not A(2));
    B(4) <= ((A(5) xor A(4)) and not A(3) and not A(2) and not A(1))
              or (A(5) and not A(3) and A(2))
              or (not A(5) and not A(4) and A(3) and A(2) and not A(1))
              or (not A(5) and A(3) and A(2) and A(1))
              or (A(5) and A(4) and A(3) and not A(2) and not A(1))
              or ((A(5) xnor A(4)) and A(3) and not A(2) and A(1))
              or ((A(5) or A(4)) and not A(3) and not A(2) and A(1));
    B(5) <= (A(5) and not A(4) and not A(3) and not A(2))
              or ((A(5) xor A(4)) and not A(3) and A(2))
              or (A(4) and A(3) and A(2))
              or (not A(5) and A(4) and A(3) and not A(2));
    B(6) <= (A(5) and not A(4) and A(3)) or (A(5) and A(4));
    B(7) <= '0';
end Behavioral;

```

3. Results

Test bench

```
stim_proc: process
begin
    A <= "000110"; wait for 100 ns; A <= "000111"; wait for 100 ns;
    A <= "011010"; wait for 100 ns; A <= "011011"; wait for 100 ns;
    A <= "111000"; wait for 100 ns; A <= "111001"; wait for 100 ns;
    A <= "111110"; wait for 100 ns; A <= "111111"; wait for 100 ns;
end process;
```

Name	Value	0 ns	100 ns	200 ns	300 ns	400 ns	500 ns	600 ns	700 ns
► a[5:0]	000110	000110	000111	011010	011011	111000	111001	111110	111111
► b[7:0]	00000110	00000110	00000111	00100110	00100111	01010110	01010111	01100010	01100011

Q5: 1. Analytical Solution

ABCD	a	b	c	d	e	f	g
0000	1	1	1	1	1	1	0
0001	0	1	1	0	0	0	0
0010	1	1	0	1	1	0	1
0011	1	1	1	1	0	0	1
0100	0	1	1	0	0	1	1
0101	1	0	1	1	0	1	1
0110	1	0	1	1	1	1	1
0111	1	1	1	0	0	0	0
1000	1	1	1	1	1	1	1
1001	1	1	1	1	0	1	1
1010	0	0	0	0	0	0	0
1011	0	0	0	0	0	0	0
1100	0	0	0	0	0	0	0
1101	0	0	0	0	0	0	0
1110	0	0	0	0	0	0	0
1111	0	0	0	0	0	0	0

AB	CD	00	01	11	10
00	00	1	0	1	1
01	01	0	1	1	1
11	11	0	0	0	0
10	10	1	1	0	0

AB	CD	00	01	11	10
00	00	1	1	1	1
01	01	1	0	1	0
11	11	0	0	0	0
10	10	1	1	0	0

$$b = A'C'D' + B'C' + A'B' + A'CD$$

AB	CD	00	01	11	10
00	00	1	1	1	0
01	01	1	1	1	1
11	11	0	0	0	0
10	10	1	1	0	0

$$c = B'C' + A'B + A'D$$

$$a = B'C'D' + AB'C' + A'BD + A'C$$

	CD	00	01	11	10
AB	00	1	0	1	1
	01	0	1	0	1
	11	0	0	0	0
	10	1	1	0	0

$$d = B'C'D' + AB'C' \\ + A'BC'D + A'B'C + A'CD'$$

	CD	00	01	11	10
AB	00	1	0	0	1
	01	0	0	0	1
	11	0	0	0	0
	10	1	0	0	0

$$e = B'C'D' + A'CD'$$

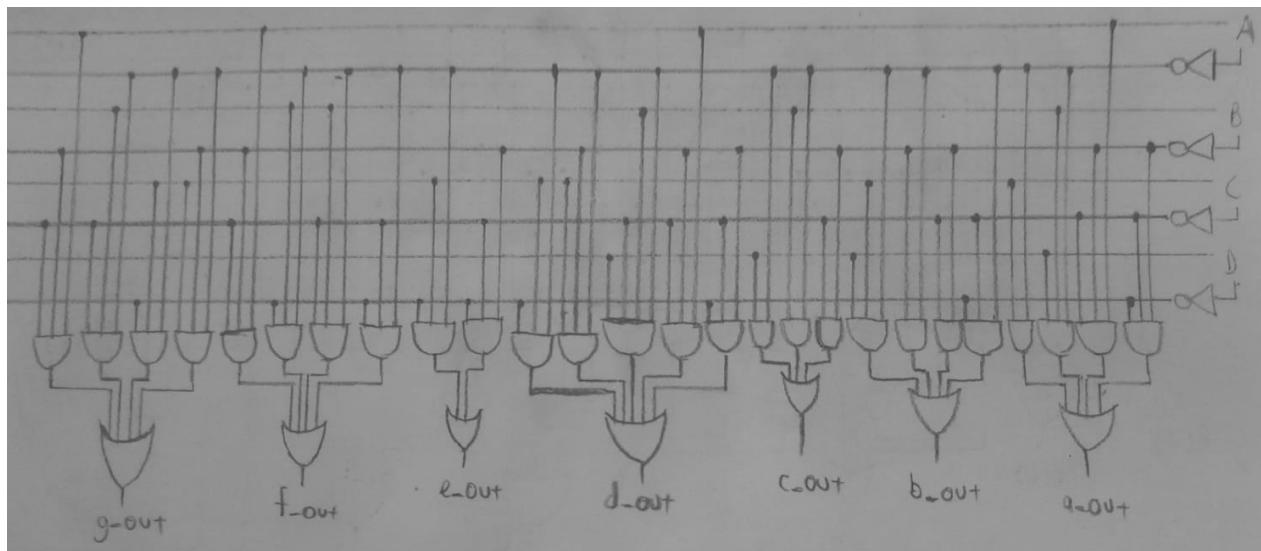
	CD	00	01	11	10
AB	00	1	0	0	0
	01	1	1	0	1
	11	0	0	0	0
	10	1	1	0	0

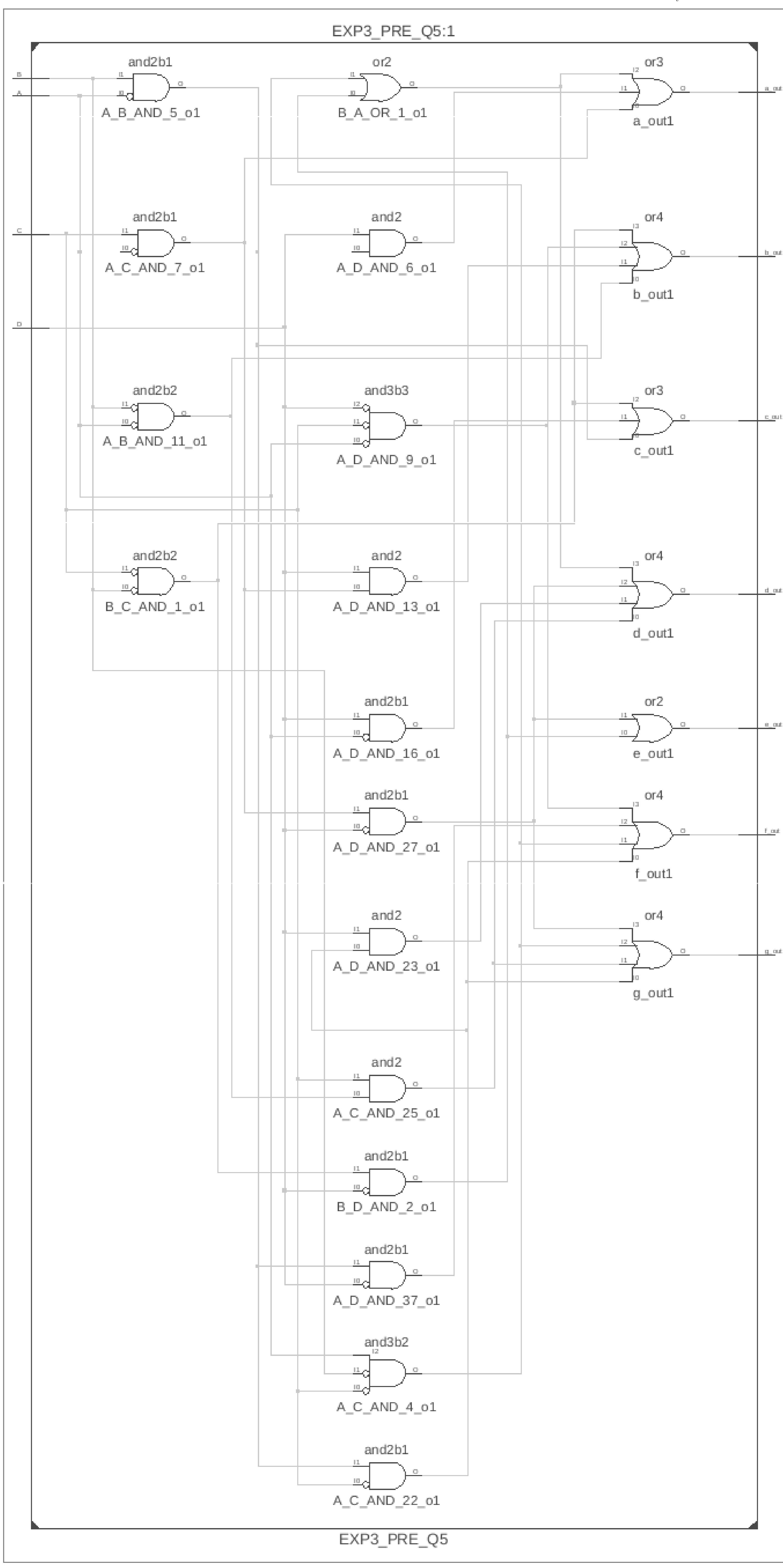
$$f = A'C'D' + A'BC' + A'BD' + ABC'$$

	CD	00	01	11	10
AB	00	0	0	1	1
	01	1	1	0	1
	11	0	0	0	0
	10	1	1	0	0

$$g = A'B'C + A'CD' + A'BC' + AB'C'$$

Hand-drawn circuit and RTL schematic





2. Codes

VHDL

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity EXP3_PRE_Q5 is
    Port ( A, B, C, D : in STD_LOGIC;
            a_out, b_out, c_out, d_out,
            e_out, f_out, g_out : out STD_LOGIC);
end EXP3_PRE_Q5;

architecture Behavioral of EXP3_PRE_Q5 is

begin
    a_out <= (not B and not C and not D) or (A and not B and not C)
              or (not A and B and D) or (not A and C);
    b_out <= (not A and not C and not D) or (not B and not C)
              or (not A and not B) or (not A and C and D);
    c_out <= (not B and not C) or (not A and B) or (not A and D);
    d_out <= (not B and not C and not D) or (A and not B and not C)
              or (not A and B and not C and D) or (not A and not B and C)
              or (not A and C and not D);
    e_out <= (not B and not C and not D) or (not A and C and not D);
    f_out <= (not A and not C and not D) or (not A and B and not C)
              or (not A and B and not D) or (A and not B and not C);
    g_out <= (not A and not B and C) or (not A and C and not D)
              or (not A and B and not C) or (A and not B and not C);
end Behavioral;

```

3. Results

Test bench

```

stim_proc: process
begin
    A <= '0'; B <= '0'; C <= '0'; D <= '0'; wait for 100 ns;
    A <= '0'; B <= '0'; C <= '0'; D <= '1'; wait for 100 ns;
    A <= '0'; B <= '0'; C <= '1'; D <= '0'; wait for 100 ns;
    A <= '0'; B <= '0'; C <= '1'; D <= '1'; wait for 100 ns;
    A <= '0'; B <= '1'; C <= '0'; D <= '0'; wait for 100 ns;
    A <= '0'; B <= '1'; C <= '0'; D <= '1'; wait for 100 ns;
    A <= '0'; B <= '1'; C <= '1'; D <= '0'; wait for 100 ns;
    A <= '0'; B <= '1'; C <= '1'; D <= '1'; wait for 100 ns;
    A <= '1'; B <= '0'; C <= '0'; D <= '0'; wait for 100 ns;
    A <= '1'; B <= '0'; C <= '0'; D <= '1'; wait for 100 ns;
    A <= '1'; B <= '0'; C <= '1'; D <= '0'; wait for 100 ns;
    A <= '1'; B <= '0'; C <= '1'; D <= '1'; wait for 100 ns;

```

```

A <= '1'; B <= '1'; C <= '0'; D <= '0'; wait for 100 ns;
A <= '1'; B <= '1'; C <= '0'; D <= '1'; wait for 100 ns;
A <= '1'; B <= '1'; C <= '1'; D <= '0'; wait for 100 ns;
A <= '1'; B <= '1'; C <= '1'; D <= '1'; wait for 100 ns;
end process;

```

