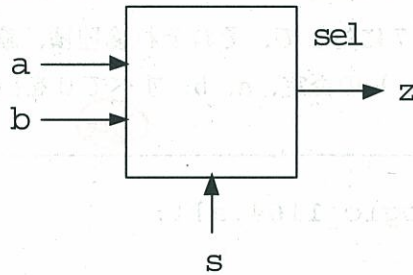


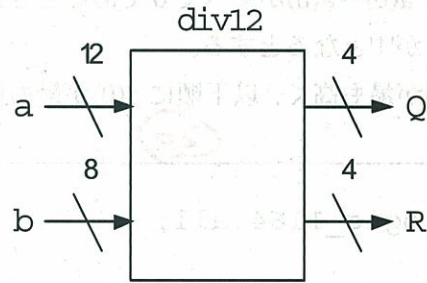
番号	氏名 秋田 純一
----	----------

1. 次のような入出力を持つ回路を VHDL で記述する際のエンティティ記述を完成させよ。なお回路の名称はブロック図の上に記されている。(10点×2) ※到達目標1・2・3

(1)



(2)



⑩×2
* 2(2)=7. (-1)~(-5)

```
entity sel is
  port ( a, b, s: in std_logic;
         z: out std_logic );
end sel;
```

```
entity div12 is
  port ( a: in std_logic_vector
         (11 downto 0);
         b: in std_logic_vector
         (7 downto 0);
         Q, R: out std_logic_vector
         (3 downto 0) );
end div12;
```

2. 2つの入力 a, b から、s で指定したものを出力 x とするセレクタを 2通りの方法によって VHDL で記述した。ただし一方の動作は正しくない。両者にそれぞれの下に示したグラフのような入力を与えた場合に予想されるそれぞれの出力を、それぞれ下のグラフに示せ。ただし変化のタイミングが同時である箇所を縦の点線で明示すること。(10×2点) ※到達目標4・5

```
library ieee;
use ieee.std_logic_1164.all;

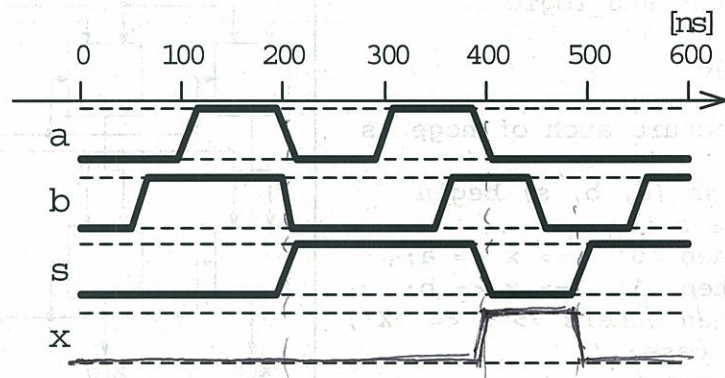
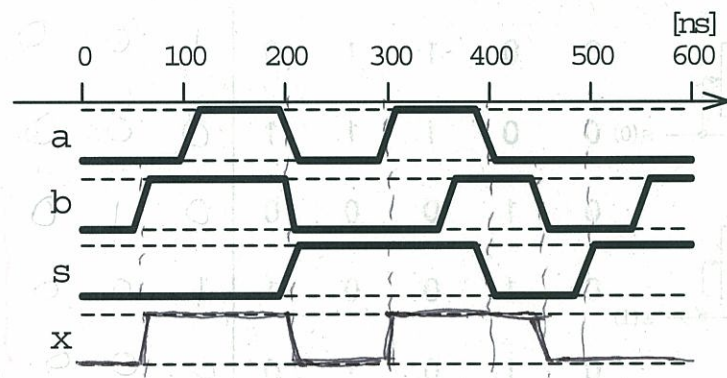
entity sel1 is
  port (
    a, b: in std_logic;
    s: in std_logic;
    x: out std_logic
  );
end sel1;

architecture arch of sel1 is
begin
  process (a, b, s) begin
    case s is
      when '1' => x <= a;
      when '0' => x <= b;
      when others => x <= 'X';
    end case;
  end process;
end arch;
```

```
library ieee;
use ieee.std_logic_1164.all;

entity sel2 is
  port (
    a, b: in std_logic;
    s: in std_logic;
    x: out std_logic
  );
end sel2;

architecture arch of sel2 is
begin
  process (s) begin
    case s is
      when '1' => x <= a;
      when '0' => x <= b;
      when others => x <= 'X';
    end case;
  end process;
end arch;
```



⑩×2
* タイミング指定なし
→ (-1)~(-5)

3. 教科書や授業で扱った回路の VHDL 記述を参考に、指定された機能をもつ回路の VHDL 記述を、欠けている箇所を補って完成させよ。(20点×2) ※到達目標 3・4

(1) 4ビットプライオリティエンコーダ enc4

※a(0)~a(3)のいずれかが'1'であるときには、出力 x はその添え字 0~3 となり、a(0)~a(3)がすべて'0'であるときには、出力 x は全ビットが'1'となるとする。
 ※入力の優先度は a(3)が最も高く、以下順に a(0)が最も低い。

```

library ieee;
use ieee.std_logic_1164.all;

entity enc4 is
    port (
        a: in std_logic_vector(3 downto 0);
        x: out std_logic_vector(2 downto 0)
    );
end enc4;

architecture arch of enc4 is
begin
    process (a) begin
        if (a(3) = '1')
            then x <= "011";
        elsif (a(2) = '1')
            then x <= "010";
        elsif (a(1) = '1')
            then x <= "001";
        elsif (a(0) = '1')
            then x <= "000";
        else
            x <= "111";
        end if;
    end process;
end arch;
    
```

※逆 → △
 ※ 011 → 111 等
 他に異なる → △

(2) 8ビットALU alu8

※入力 a, b に対して、f で指定する論理演算結果を x として出力する。
 ※f=0, 1, 2, 3, 4, 5, 6, 7 に対して、それぞれ論理積、論理和、排他的論理和、a の否定、b の否定、a、b、すべて 0 を出力する。

```

library ieee;
use ieee.std_logic_1164.all;

entity alu8 is
    port (
        a, b: in std_logic_vector(7 downto 0);
        f: in std_logic_vector(2 downto 0);
        x: out std_logic_vector(7 downto 0)
    );
end alu8;

architecture arch of alu8 is
begin
    process (a, b, f) begin
        case f is
            when "000" => x <= a and b;
            when "001" => x <= a or b;
            when "010" => x <= a xor b;
            when "011" => x <= not a;
            when "100" => x <= not b;
            when "101" => x <= a;
            when "110" => x <= b;
            when "111" => x <= "00000000";
            when others => x <= "XXXXXXXX";
        end case;
    end process;
end arch;
    
```

※ f a+b 等 → △
 ※ a → +2
 ※ 文法ミス → △

4. 以下のような VHDL 記述による回路 hoge を図 4-1 のように接続した回路に対して、表 4-1 のような入力を与えた場合に得られる出力を表 1 中に示せ。(20点) ※到達目標 5

```

library ieee;
use ieee.std_logic_1164.all;

entity hoge is
    port (
        a, b, s, in std_logic;
        x: out std_logic
    );
end hoge;

architecture arch of hoge is
begin
    process (a, b, s) begin
        case s is
            when '0' => x <= a;
            when '1' => x <= b;
            when others => x <= 'X';
        end case;
    end process;
end arch;
    
```

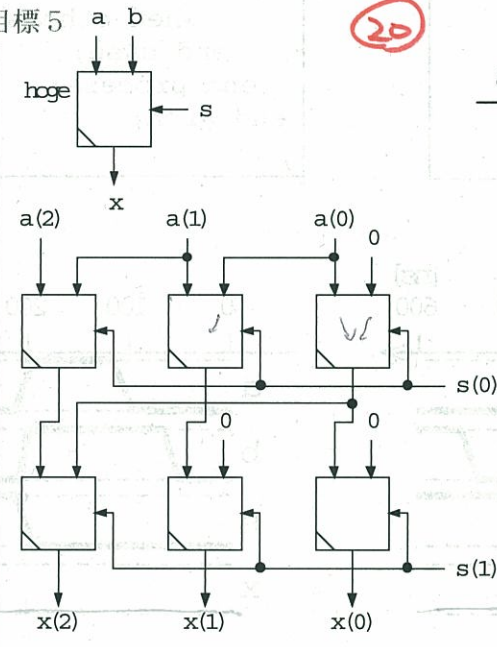


図 4-1

表 4-1

a(2)	a(1)	a(0)	s(1)	s(0)	x(2)	x(1)	x(0)
0	0	1	0	0	0	0	1
0	0	1	0	1	0	1	0
0	0	1	1	0	1	0	0
0	0	1	1	1	0	0	0
0	1	0	0	0	0	1	0
0	1	0	0	1	1	0	0
0	1	0	1	0	0	0	0
0	1	0	1	1	0	0	0

※
 ※
 ※