

解答あり

番号	氏名 秋田 氏
----	---------

1. 次のような2種類の回路 hoge1, hoge2 に図1のような信号を与えたときの出力 q1, q2 を図示せよ。ただし q1, q2 の初期値は'0'とする。また q1, q2 の変化のタイミングを、対応する信号のタイミングと結ぶ縦の点線で明示すること。(10点×2)

```
entity hoge1 is
  port (clk, rst, d: in std_logic;
        q1: out std_logic);
end hoge1;

architecture arch of hoge1 is
begin
  process (clk, rst) begin
    if (clk'event and clk = '1') then
      if (rst = '1') then q1 <= '0';
      else q1 <= d;
      end if;
    end if;
  end process;
end arch;
```

```
entity hoge2 is
  port (clk, rst, d: in std_logic;
        q2: out std_logic);
end hoge2;

architecture arch of hoge2 is
begin
  process (clk, rst) begin
    if (rst = '1') then q2 <= '0';
    elsif (clk'event and clk = '1') then
      q2 <= d;
    end if;
  end process;
end arch;
```

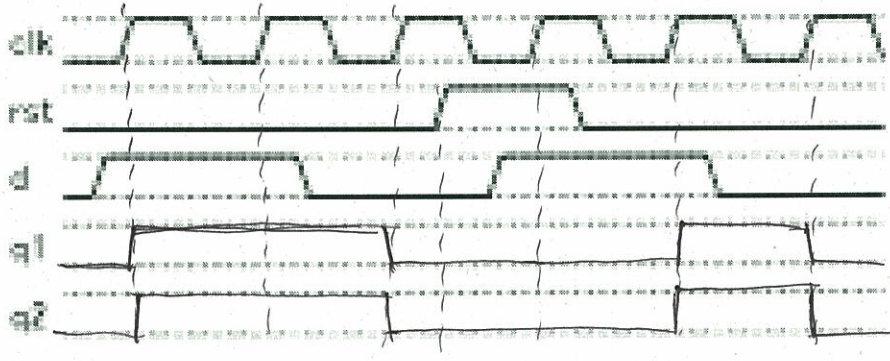


図1

点線は →

2. 次のような仕様のデータバス hoge3 を VHDL (アーキテクチャ記述のみでよい) で記述せよ。ただし授業で扱った「構造を意識する記述」「構造を意識しない記述」のどちらでもよい。(20点)

```
architecture arch of hoge3 is
  process (clk, rst) begin
    if (rst = '1') then q <= "00000000";
    elsif (clk'event && clk = '1') then
      case s is
        when "00" => q <= q + 1;
        when "01" => q <= q + a;
        when "10" => q <= q - 1;
        when "11" => q <= q - a;
      end case;
    end if;
  end process;
```

- 入力s (機能選択: 2ビット)、a (データ: 8ビット)。
- 出力は q (結果: 8ビット)。
- s="00"のときは、現在の q の値に 1 を加えて q に保持。
- s="01"のときは、現在の q の値に a を加えて q に保持。
- s="10"のときは、現在の q の値から 1 を減じて q に保持。
- s="11"のときは、現在の q の値から a を減じて q に保持。

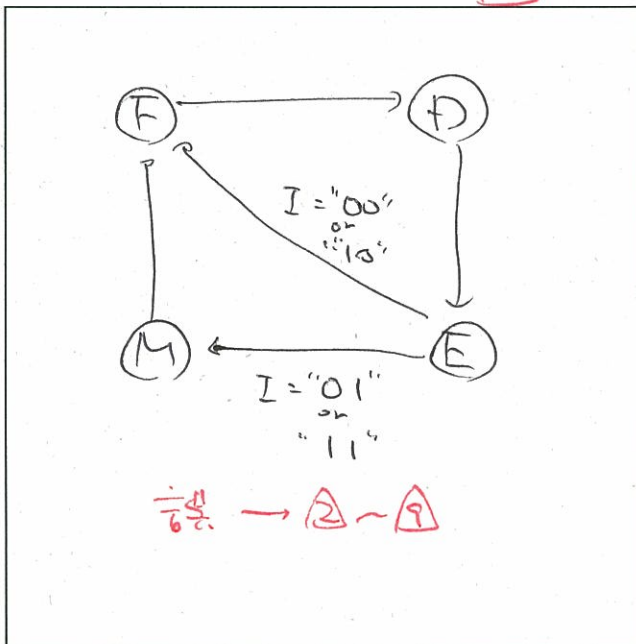
動作が正しくなるように  
 記述  
 文法ミス: A - A  
 構造がわかるように → A - A  
 process a → A  
 clk  
 clk a  
 同期性 (プロセス) 2つ → A  
 end arch;

3. メモリ内の命令を順に実行するプロセッサの動作を制御するステートマシン hoge4 を考える。この hoge4 は、4 つの状態 F, D, E, M をもち、それぞれ命令取得 (フェッチ)、命令解釈 (デコード)、命令実行、結果のメモリへの書き戻し、の動作に対応し、基本的にはこの順に遷移が起こる。hoge4 は入力 I (2 ビット) をもち、これは表 1 のような意味を持つ、デコードされた命令の内容であるとする。ステートマシンの状態遷移図における状態 M は、入力 I の値に応じて表 1 のように飛ばされることある (例えば何も実行しない NOP 命令では、結果のメモリへの書き戻しは必要ないため、状態遷移は F→D→E→次の F、となる)。このステートマシン hoge4 の状態遷移図を図示し、VHDL 記述を完成させよ。なお F\_ST, D\_ST, E\_ST, M\_ST は、それぞれ状態 F, D, E, M を表す状態コードとし、リセット時の初期状態は F とする。(50 点)

表 1

I	命令の記号	命令の意味	状態 M の経由
00	NOP	何もしない	×
01	ADD	加算	○
10	JMP	ジャンプ	×
11	LD	代入	○

状態遷移図



```

process (clk, rst) begin
  if (rst = '1' then state <= F_ST;
  elsif (clk'event and clk = '1') then
    case state is
      when F_ST => state <= D_ST;
      when D_ST => state <= E_ST;
      when E_ST =>
        if (I = "00" or I = "10") then
          state <= F_ST;
        else state <= M_ST;
        end if;
      when M_ST => state <= F_ST;
    end case;
  end if;
end process;
  
```

基本動作: ④  
 文法: ① ~ ⑤  
 構造: ⑤ ~ ⑨

4. この授業を通して学んだことが、あなたの普段の生活や今後の進路にどのような関係・影響があったか(またはありそうか)、他の授業やこれまでの授業で学んだこととの関連やあなた自身の感想などを交えて、自由に考えを述べてください。(記述の内容は点数に反映させませんので、思うままに自由に述べてください) (10点)

基本的 12 ⑩ + d