

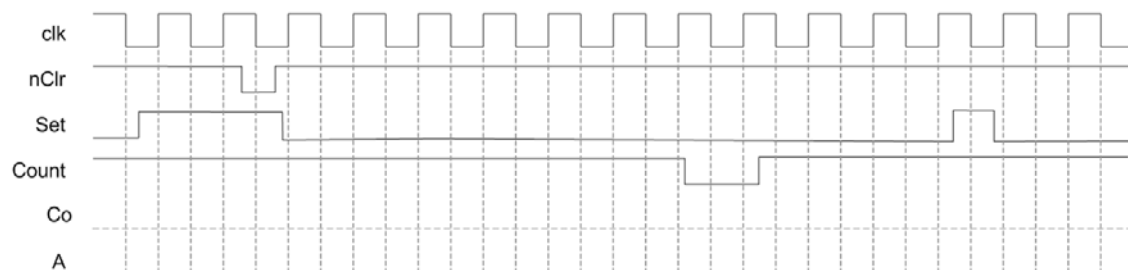
디지털회로설계 및 언어 Final Test

Jun. 12, 2019

배점 1(H18), 2(3H24), 3c(H20), 그외 각 H6.

1. 다음 설계에서 IN4=3 으로 가정하여 답하라.
 - a. HDL code 에서 잘못된 것을 찾아 고쳐라.
 - b. timing diagram 에서 빈 곳을 채워라.

```
module Exam_code (Count, Set, IN4, clk, nClr, A, Co);  
    input Count, Set, clk, nClr;  
    input [3:0] IN4;  
    output Co;  
    output [3:0] A;  
    reg [3:0] A;  
    Co <= ~Set & (A == 4'b1001);  
    always @ (posedge clk, negedge nClear)  
        if (nClear) A <= 4'd0;  
        else if (Set) A <= IN4;  
        else if (Count) if (Co) A <= 4'd0; else A <= A+1'b1;  
        else A <= A;  
endmodule
```



2. 다음 문제에 대해 답하라.
- $+22.625_{10}$ 을 2진수로 바꾸되 소수점 이하 5자리 까지 계산하라. (자리버림 이용)
 - a의 답에 대한 2의 보수를 구하라.
 - b의 답을 16진수로 바꿔라.
 - -22.625_{10} 을 2진수로 바꿔라.
 - a와 d에서 주어진 수에 -16을 곱한 값을 이진수로 바꿔라.
 - $A=12'h9cb, B=2'b01, C=2'h3$ 일때 $D=\{C, 2\{B\}, 1'b0\}$ 를 구하라.
3. 아래 그림의 chart를 참고하여 아래 질문에 답하라. Verilog HDL code에 대한 질문은 “Verilog HDL code” 부분을 참고하여 답하되 code내의 comments를 잘 읽고 이를 반영하라. 그리고 positive-edge triggered circuit을 가정하라
- chart에 적합한 이름을 원어와 함께 말하라. 단, 수업에서 사용한 용어이어야 한다.
 - HDL code의 “part TOP” 부분에서 오류를 찾아 수정하라.
 - HDL code에서 빠진 부분, 즉 code A, code B, code C, code D 부분을 올바른 코드로 채워넣어라.
- code A
code B
code C
code D
- 아래 chart를 ASM chart로 바꿔 그려라.
 - chart를 RTL 표현(RTL description)으로 바꿔라.
 - 시스템이 정상적으로 동작하고 있는 동안 St 입력이 '0'으로 바뀌었다. 이 조건에서 state가 S_0로 진입한 시점부터 처음 3클럭동안 시스템 내 모든 레지스터의 값을 말하라.
 - 문제 3f에서 언급된 시점 이후 St 입력이 '1'로 다시 바뀌었다. 처음 3주기 동안 시스템 내 모든 레지스터의 값을 말하라.
 - 문제 3g에서 일정시간이 경과하여 S_2로 진입한 직후 A, k, E, F의 값을 말하라.

```

module Test_RTL (St,clk,nClr,E,F,A67,A,B,x); // define i/o
    input St,clk,nClr;
    output E,F,A67,x;
    output [7:0] A,B;
// Specify system registers
    reg [7:0] A, B, k; // registers
    reg E, F, A67;
    reg [1:0] ps, ns; //control register
//Specify nets

```

```

    wire k_eq_2;

// Encode the states
    parameter S_0 = 2'b00, S_1 = 2'b01, S_2 = 2'b11;
    parameter Ain = 8'd223, Bin=8'd7, kin = 8'h18;

//Specify signals
    A67 <= A{6} | A{7};

// Generate internal signals
    always @(k)
        if (k==8'd2) k_eq_2 <= 1'b1; else k_eq_2 <= 1'b0;

// State transition for control logic
    always @(posedge CLK, negedge nClr)
        if (nClr) ps <= S_0; else ps <= ns;

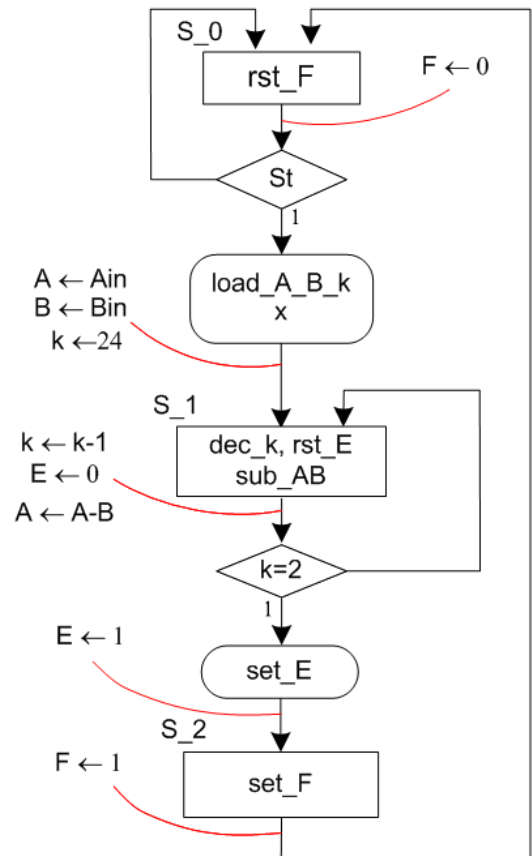
// Bottom of part TOP
// Code for next state logic
    always @ (St, k_eq_2, ps)
        case (ps)
            (code A)
            endcase

// Code for output logic from ASMD chart
    always @ (code B)
        (code C)

//Code for register transfer operations
        (code D)

endmodule

```



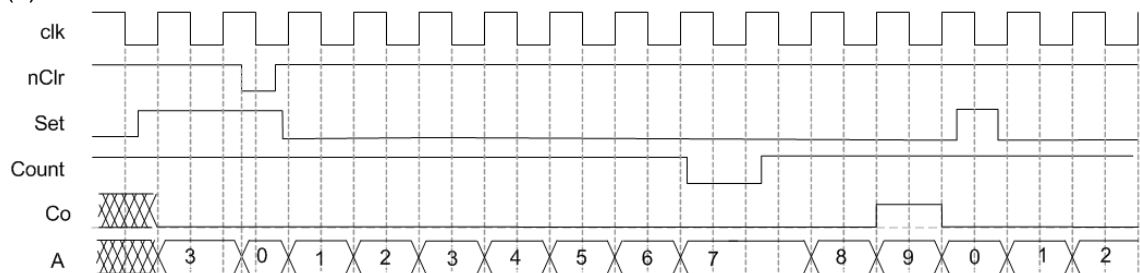
모범답안

1. Co 가 net type 인 경우

(a)

```
module Exam_code (Count, Set, IN4, clk, nClr, A, Co);  
    input Count, Set, clk, nClr;  
    input [3:0] IN4;  
    output Co;  
    output [3:0] A;  
    reg [3:0] A;  
    assign Co = ~Set & (A == 4'b1001);  
    always @ (posedge clk, negedge nClr)  
        if (~nClr) A <= 4'd0;  
        else if (Set) A <= IN4;  
        else if (Count) if (Co) A <= 4'd0; else A <= A+1'b1;  
        else A <= A;  
endmodule
```

(b)



1. Co 가 reg type 인 경우

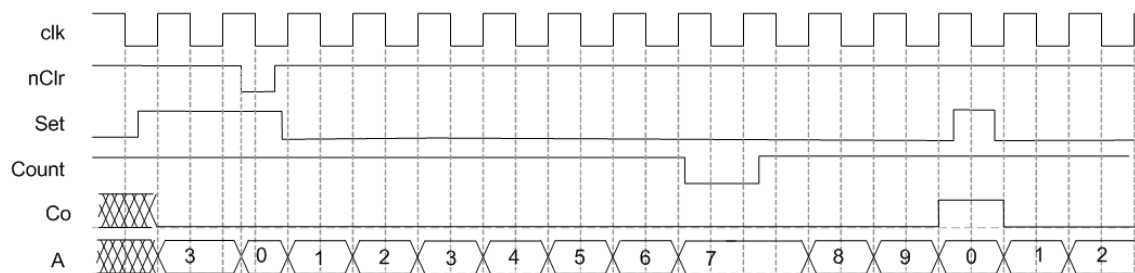
(a)

```

module Exam_code (Count, Set, IN4, clk, nClr, A, Co);
    input Count, Set, clk, nClr;
    input [3:0] IN4;
    output Co;
    output [3:0] A;
    reg [3:0] A;
    reg Co;
    Co <= ~Set & (A == 4'b1001);
    always @ (posedge clk, negedge nClr)
    begin
        Co <= ~Set & (A == 4'b1001);
        if (~nClr) A <= 4'd0;
        else if (Set) A <= IN4;
        else if (Count) if (Co) A <= 4'd0; else A <= A+1'b1;
        else A <= A;
    end
endmodule

```

(b)



2.

a. $+22.625_{10}$

$$22_{10} = 10110_2, \quad 0.625 \times 2^5 \div 2^3 = 20_{10}$$

$$0.625 \times 2^5 \div 2^3 = 5_{10} \div 2^3 = 10_2 \div 2^3 = 0.10_2$$

$$\rightarrow +22.625_{10} = 010110.1000_2$$

b. $\rightarrow 101001.01100_2$

c. $E9.60_{16}$

d. $-22.625_{10} = +22.625_{10} \times (-1)$

Therefore, $-22.625_{10} = 101001.011_2$

e. a에 -16 곱하고 2진수로

a의 2진수 표현 $\rightarrow 010110.1000_2$

$a \times 16 \rightarrow 010110100.0_2$

$a \times (-16) \rightarrow 1010010110.0_2$

(d의 2진수 $= -22.625_{10}) \times (-16)$

$= (a의 2진수) \times 16$

$= 010110100.0_2$

f. $D = 110110_2$

3.

a. ASMD(Algorithmic State Machine and Datapath) chart

b.

```
module Test_RTL (St,clk,nClr,E,F,A67,A,x); // define i/o
    input St,clk,nClr;
    output E,F,A67,x;
    output [7:0] A,B;
// Specify system registers and net
    reg [7:0] A, B, k; // registers
    reg E, F, A67, k_eq_2; //E and F flip-flops
    reg [1:0] ps, ns; //control register
    reg rst_F,load_A_B_k,x,dec_k,rst_E,sub_AB,set_E,set_F;
//Specify nets
    wire k_eq_2;
// Encode the states
    parameter S_0 = 2'b00, S_1 = 2'b01, S_2 = 2'b11;
    parameter Ain = 8'd223, Bin=8'd7, kin = 8'h18;
//Specify signals
    A67 <= A{6} | A{7}; <--- assign A67 = A[6] | A[7];
// Generate internal signals
    always @(k)
        if (k==8'd2) k_eq_2 <= 1'b1; else k_eq_2 <= 1'b0;
// State transition for control logic
    always @(posedge CLKclk, negedge nClr)
        if (nClr~nClr) ps <= S_0; else ps <= ns;
```

c.

Code A =====

Code B =====

```
always @ (St, k_eq_2, ps)
    case (ps)
        S_0: if(St) ns <= S_1; else ns <= S_0;
        S_1: if(k_eq_2) ns <= S_2; else ns <= S_1;
        S_2: ns <= S_0;
    endcase
```

```
always @(St, k_eq_2, ps)
```

Code C =====

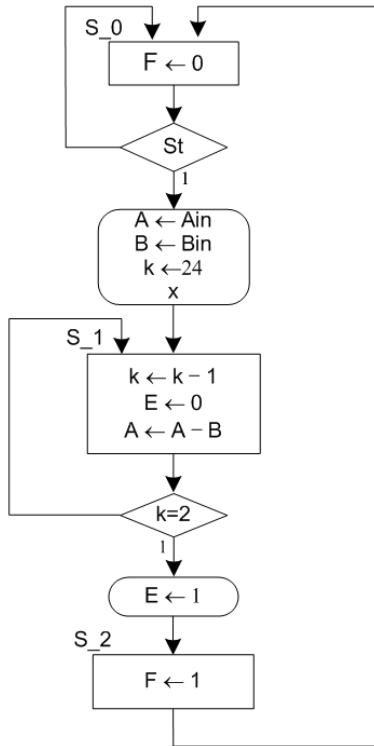
```
begin
  rst_F <= 0; load_A_B_k <= 0; x <= 0;
  dec_k <= 0; rst_E <= 0; sub_AB <= 0; set_E <= 0;
  set_F <= 0;

  case (ps)
    S_0: begin
      rst_F <=1;
      if(St) begin load_A_B_k <= 1; x <=1; end
      end
    S_1: begin
      dec_k <=1;
      rst_E <= 1; sub_AB <= 1;
      if(k_eq_2) set_E <= 1; end
    S_2: set_F <= 1;
  endcase
end
```

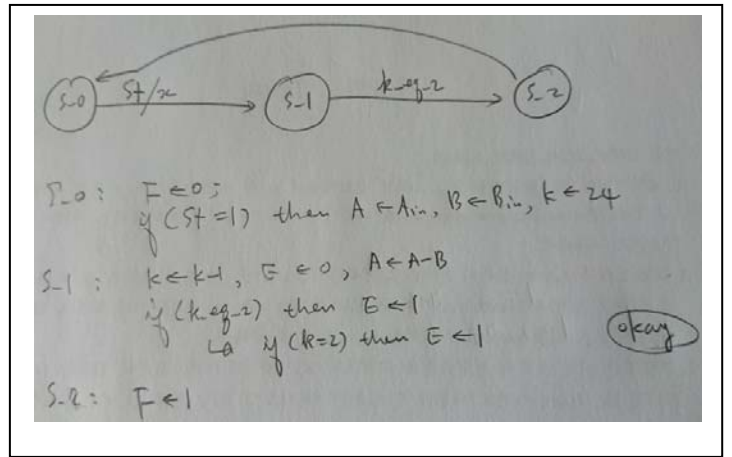
Code D =====

```
always @(posedge clk) begin
  if(rst_F)      F <= 0;
  if(load_A_B_k) begin A <= Ain; B <= Bin; k <= kin; end
  if(dec_k)      k <= k - 1;
  if(rst_E)      E <= 0;
  if(sub_AB)     A <= A-B;
  if(set_E)      E <= 1;
  if(set_F)      F <= 1;
end
```


d.



e.



f. System registers: A, B, k, E, F

S_0 진입 직후: $A=223-23 \times 7=62, B=7, k=1, E=1, F=1$

1st clock 후: $A=62, B=7, k=1, E=1, F=0$

2nd clock 후: $A=62, B=7, k=1, E=1, F=0$

g. System registers: A, B, k, E, F

1st clock 후: $A=223, B=7, k=24, E=1, F=0$

2nd clock 후: $A=216, B=7, k=23, E=0, F=0$

3rd clock 후: $A=209, B=7, k=22, E=0, F=0$

h. System registers: A, k, E, F

$A=62, k=1, E=1, F=0$