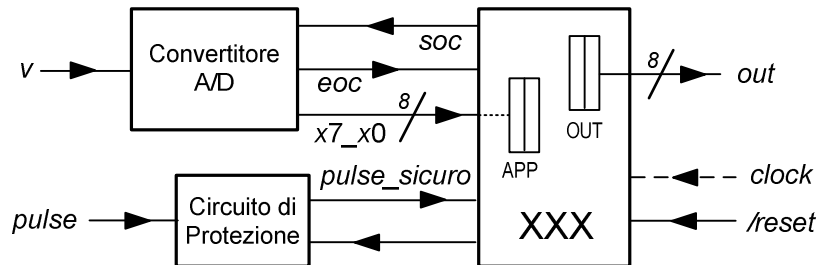


Esercizio 1

Descrivere e sintetizzare come rete sequenziale sincronizzata di Moore un *riconoscitore di sequenze*, che ha in ingresso stati di ingresso a due bit e riconosce la sequenza {11}, {00}, {01}, dove ciascuno degli stati di ingresso tra parentesi graffe può durare per un numero arbitrario di clock (≥ 1). L'uscita z deve stare ad uno per un solo clock dopo aver riconosciuto la sequenza corretta. Per la sintesi si adotti un modello con flip-flop JK come elementi di marcatura dello stato interno e reti combinatorie in forma SP.

Esercizio 2

Premessa 1: Due impulsi consecutivi su *pulse* sono così distanziati nel tempo *da non dare alcun problema di nessun tipo, però* ciascun impulso ha una durata molto più breve di un periodo del clock ma deve comunque essere visto da XXX. Per questo motivo occorre preliminarmente **specificare la struttura del circuito di protezione** evidenziato in figura.

Premessa 2: La tensione bipolare v è tale che il Convertitore, che opera in binario bipolare, non debba mai emettere il byte 'H00. **Specificare la caratteristica** cui deve soddisfare la tensione v .

Risolto quanto indicato nelle premesse, **descrivere** l'Unità **XXX** che si evolve ciclicamente in accordo alle seguenti specifiche:

- 1 Nel primo ciclo dopo il reset iniziale, si evolve come se l'ultimo campione prelevato dal Convertitore fosse il byte anomalo 'H00, cioè come se il registro APP contenesse 'H00.
- 2 Ogni ciclo (primo compreso) ha la durata di 100 periodi di clock e si compone di due fasi:
 - a) Nella prima fase XXX manda sull'uscita *out* l'ultimo campione prelevato dal Convertitore A/D e ne preleva uno nuovo (il Convertitore risponde in tempi non noti ma molto minori di 100 periodi del clock).
 - b) Nella seconda fase, che termina con la naturale fine del conteggio dei 100 periodi di clock, XXX modifica o lascia inalterato il valore di *out* a seconda che rilevi o meno l'arrivo di un impulso tramite *pulse_sicuro*. L'eventuale modifica consiste nel sostituire il valore attuale di *out* con il byte anomalo 'H00.

Individuare infine le equazioni algebriche delle variabili di condizionamento e lasciare in Verilog la conseguente descrizione della Parte Controllo di XXX.

Soluzione esercizio 1

La tabella di flusso è nella figura accanto. Scegliendo la codifica S0=00, S1=01, S2=10, S3=11, si ottiene:

$$z = y_1 \cdot y_0.$$

Utilizzando un modello strutturale che prevede flip-flop JK come meccanismi di marcatura, le tabelle delle transizioni per la rete combinatoria che produce lo stato interno sono le seguenti:

x_1x_0	00	01	11	10	z
S0	S0	S0	S1	S0	0
S1	S2	S0	S1	S0	0
S2	S2	S3	S1	S0	0
S3	S0	S0	S1	S0	1

y_1y_0	x_1x_0	00	01	11	10	
S0	00	00	00	01	00	a_1a_0
S1	01	10	00	01	00	
S3	11	00	00	01	00	
S2	10	10	11	01	00	

y_1y_0	x_1x_0	00	01	11	10	
S0	00	0-	0-	0-	0-	j_1k_1
S1	01	1-	0-	0-	0-	
S3	11	-1	-1	-1	-1	
S2	10	-0	-0	-1	-1	

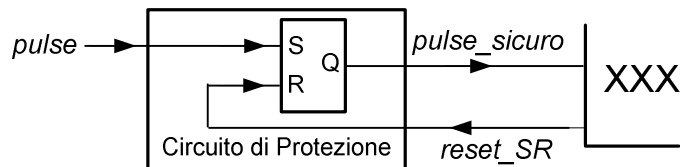
y_1y_0	x_1x_0	00	01	11	10	
S0	00	0-	0-	1-	0-	j_0k_0
S1	01	-1	-1	-0	-1	
S3	11	-1	-1	-0	-1	
S2	10	0-	1-	1-	0-	

Da cui si ricava la seguente sintesi SP:

$$j_1 = \overline{x_1} \cdot \overline{x_0} \cdot y_0, \quad j_0 = x_1 \cdot x_0 + y_1 \cdot x_0$$

$$k_1 = y_0 + x_1, \quad k_0 = \overline{x_1} + \overline{x_0}$$

Esercizio 2 - Una possibile soluzione



```

module XXX(x7_x0,soc,eoc,out,pulse_sicuro,reset_SR,reset_);
input      clock,reset_;
input[7:0] x7_x0;
output     soc;
input      eoc;
output[7:0] out;
input      pulse_sicuro;
output     reset_SR;
reg        soc;          assign soc=soc;
reg        RESET_SR;     assign reset_SR=RESET_SR;
reg[7:0] APP,OUT;        assign out=OUT;
reg[6:0] COUNT;          parameter Num_Periodi=10;
reg[1:0] STAR;           parameter S0=0,S1=1,S2=2,S3=3;
always @(reset_==0) begin COUNT<=Num_Periodi; RESET_SR<=1; soc<=0;
                        APP<='H00; STAR=S0; end
always @(posedge clock) if (reset_==1) #1
    casex(STAR)
        S0: begin COUNT<=COUNT-1; OUT<=APP; RESET_SR<=0; soc<=1;
STAR<=(eoc==1)?S0:S1;end
        S1: begin COUNT<=COUNT-1; soc<=0; APP<=x7_x0; STAR<=(eoc==0)?S1:S2; end
        S2: begin OUT<=(pulse_sicuro==1)?'H00:OUT;
RESET_SR<=(pulse_sicuro==1)?1:0;
COUNT<=(COUNT==1)?Num_Periodi:(COUNT-1);
STAR<=(COUNT==1)?S0:S2; end
    endcase
endmodule

```