Esercizio 1

Sia data una rete combinatoria che: i) riceve in ingresso tre variabili x_2, x_1, x_0 che esprimono un numero naturale X ad una cifra in base 5 (in codifica 421) ed una *variabile di comando b*, e ii) produce in uscita tre variabili y_2, y_1, y_0 che esprimono un numero naturale Y ad una cifra in base 5 ed una variabile *c* secondo la seguente legge.

Il numero naturale Y è legato al numero naturale X dalla relazione

$$Y = \begin{cases} |X+1|_5 & b=0 \\ |X-1|_5 & b=1 \end{cases}$$

La variabile *c* vale 1 se il risultato dell'operazione scritta tra $|\bullet|$ non è rappresentabile su una cifra in base 5 e 0 altrimenti.

Attenzione: c'e' scritto "il risultato della operazione scritta tra $|\bullet|$"

1) Descrivere la rete *nella sua completezza*, riempiendo le mappe
2) Sintetizzare la sottorete che genera y_0 a costo minimo a porte NOR. Trovare tutte le liste di copertura non ridondanti.

Soluzione

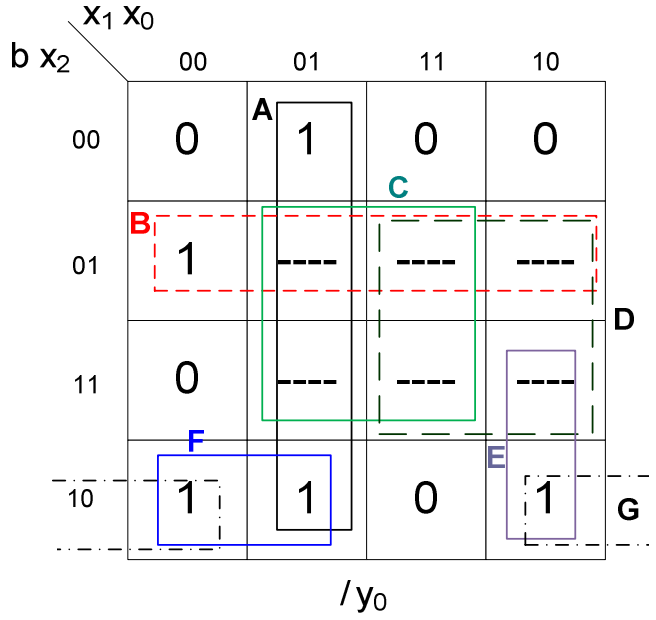
b x_2 \ x_1 x_0	00	01	11	10
00	0	0	0	0
01	1	----	----	----
11	0	----	----	----
10	1	0	0	0

c

b x_2 \ x_1 x_0	00	01	11	10
00	001	010	100	011
01	000	----	----	----
11	011	----	----	----
10	100	000	010	001

$y_2\, y_1\, y_0$

Dalla mappa sopra scritta si ricava immediatamente la sintesi PS:



Gli implicanti A, B sono essenziali, C, D sono assolutamente eliminabili, e F,G,H sono semplicemente eliminabili. Le liste di copertura irridondanti sono: {A,B,G} e {A,B,E,F}. La prima delle due è a costo minimo.

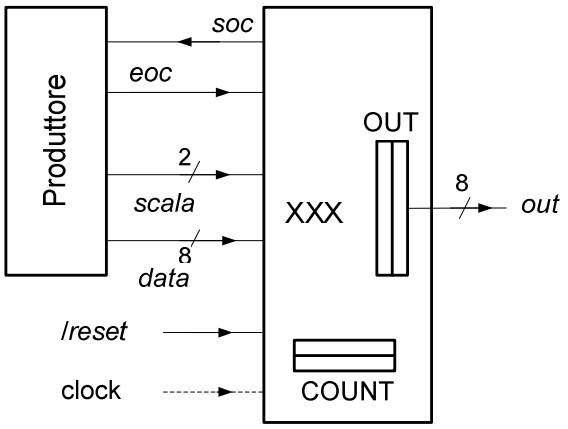
La sintesi PS è pertanto: $\overline{y_0} = \overline{x_1} \cdot x_0 + \overline{b} \cdot x_2 + b \cdot \overline{x_2} \cdot \overline{x_1}$, da cui si ricava immediatamente quella a porte NOR:

$$y_0 = \overline{\left(\overline{x_1 + \overline{x_0}}\right) + \left(\overline{b + \overline{x_2}}\right) + \left(\overline{\overline{b} + x_2 + x_1}\right)}$$

Es. 2

Il Produttore invia, su richiesta dell'Unità **XXX**, una informazione da 10 bit di cui un numero (a 2 bit) tramite le variabile *scala* e un byte tramite le variabili *data*. Descrivere e sintetizzare l'Unità *XXX* che, partendo da uno stato interno S0 ripeta all'infinito un ciclo S0, S1, …, S0, S1, … in cui:

1) Emette tramite *out* l'ultimo byte ricevuto tramite *data* e lo mantiene per un numero di periodi di clock pari a N = (***scala* · 8**).
2) **Mentre mantiene** fermo lo stato di *out*, richiede al Produttore una nuova informazione



NOTE

a) Si usi un registro COUNT per effettuare il conteggio e si ponga, al reset, COUNT<=8, OUT<= 'HAA e STAR<=S0.
b) Si supponga che il valore *scala* sia un numero sempre maggiore di 0 e che il Produttore sia sufficientemente veloce da produrre un nuova informazione utile in un tempo inferiore a N periodi di clock.

Fare, per **XXX**, un diagramma temporale che, partendo dal reset, includa **tutti** (ripeto: tutti) i seguenti stati interni: **S0,** S1, S2,….., **S0, S1** e che il Produttore, quando risponde alla prima richiesta, invii tramite *scala* , il numero 2 e tramite *data* il byte 'H55.

Descrivere e DISEGNARE la parte operativa limitatamente al registro COUNT

Una possibile soluzione

```
module XXX(soc, eoc, scala,data, out, clock,reset_);
input  clock, reset_;
output soc;
input  eoc;
input  [1:0] scala;
input  [7:0] data;
output [7:0] out;
reg        SOC;  assign soc=SOC;
reg [7:0]  OUT;  assign out=OUT;
reg [4:0]  COUNT;
reg [1:0]  STAR; parameter S0=0, S1=1, S2=2;
always @(reset_==0) begin SOC<=0; COUNT<=8; OUT<='HAA; STAR<=S0; end
always @(posedge clock) if (reset_==1) #4
  casex(STAR)
   S0:  begin COUNT<=COUNT-1; SOC<=1; STAR<=(eoc==1)?S0:S1; end
   S1:  begin COUNT<=COUNT-1; SOC<=0; STAR<=(eoc==0)?S1:S2; end
   S2:  begin OUT<=(COUNT==1)?data:OUT; COUNT<=(COUNT==1)?{scala,3'B0}:(COUNT-
1);
              STAR<=(COUNT==1)?S0:S2; end
  endcase
endmodule
```