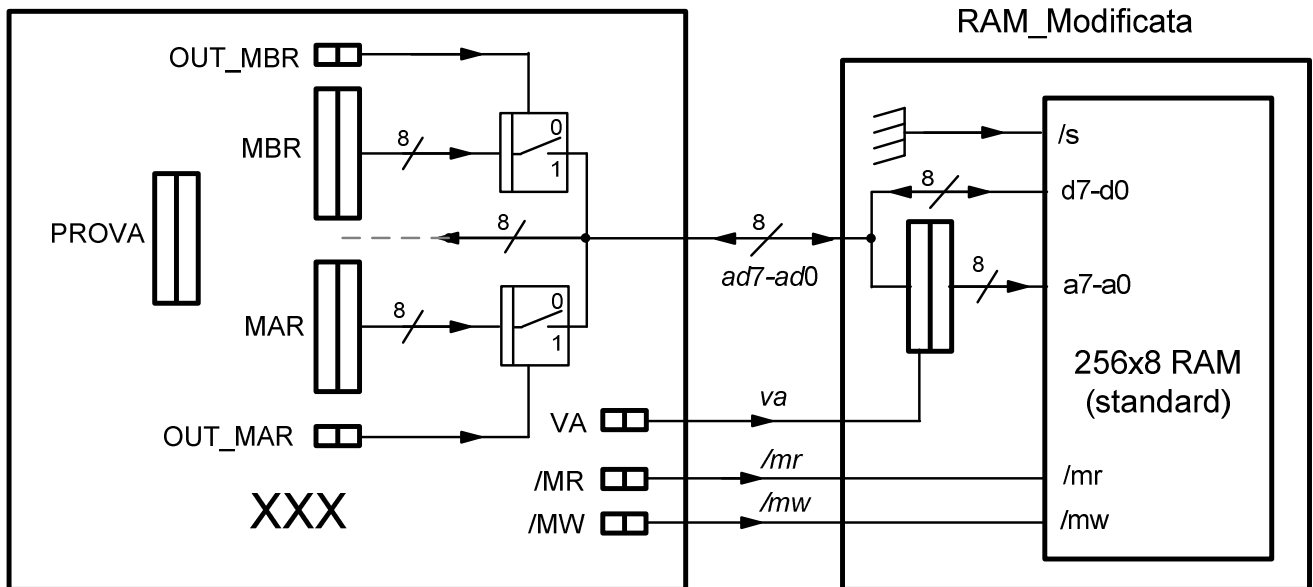


Esercizio 1

Sintetizzare una rete che ha in ingresso un numero naturale in base 10 su quattro cifre, codificate BCD (o 8421), e tre uscite d2, d5, d3, che valgono 1 se il numero in ingresso è divisibile per 2, 5 e 3 rispettivamente. Descrivere esplicitamente qualunque rete non svolta a lezione.

Data la assoluta semplicità degli algoritmi richiesti, verrà valutata la qualità della soluzione.

Esercizio 2



L'Unità XXX usa, per accedere alla RAM_Modificata, un unico bus ad_7-ad_0 per gli indirizzi e per i dati. Durante un ciclo di accesso alla RAM_Modificata, l'Unità XXX prima immette sul bus gli indirizzi e poi utilizza il bus per i dati: la presenza di indirizzi validi è segnalata alla RAM_Modificata, dalla variabile va , che viene messa ad 1 per un ciclo di clock e poi riportata a 0. La RAM (standard) non necessita di stati di wait.

Scrivere in Verilog:

- 1) un microsottoprogramma che legga la locazione di memoria il cui indirizzo sia stato preventivamente depositato in MAR e che lasci nel registro MBR il contenuto della locazione letta.

```
read0: begin ... ; end
...
...
readF: begin ..; STAR<=MJR; end
```

- 2) un microsottoprogramma che scriva il contenuto di MBR nella locazione di memoria il cui indirizzo sia stato preventivamente depositato in MAR.

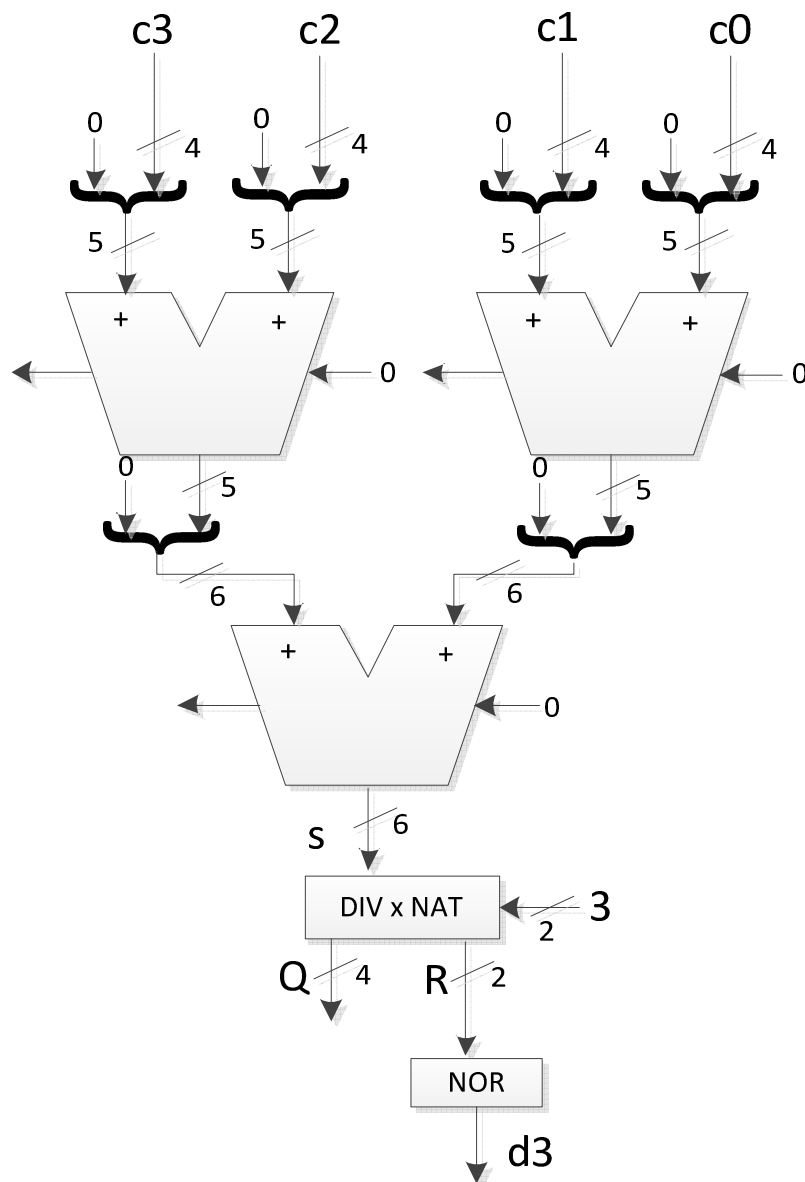
```
write0: begin ... ; end
...
...
writeF: begin ..; STAR<=MJR; end
```

- 3) Descrivere l'Unità XXX che, possedendo i micro sottoprogrammi di cui sopra, immetta il byte 'HAA nella locazione di indirizzo 231 e quindi legga questa locazione e ne immetta il contenuto nel registro PROVA.

Soluzione Esercizio 1

- Un numero in base 10 è divisibile per 2 se la cifra meno significativa è 0, 2, 4, 6, 8. Pertanto, detto $c_{x,y}$ il bit y della cifra decimale x ($0 \leq x \leq 3, 0 \leq y \leq 3$) si ha $d2 = \overline{c_{0,0}}$.
- Un numero in base 10 è divisibile per 5 se la cifra meno significativa è 0 o 5. Pertanto,

$$d5 = \overline{c_{0,0}} \cdot \overline{c_{0,1}} \cdot \overline{c_{0,2}} \cdot \overline{c_{0,3}} + \overline{c_{0,0}} \cdot \overline{c_{0,1}} \cdot \overline{c_{0,2}} \cdot \overline{c_{0,3}}$$
- Un numero in base 10 è divisibile per 3 se $s = c_0 + c_1 + c_2 + c_3$ è divisibile per tre. s è un numero (binario) che sta su 6 bit. Pertanto d3 può essere prodotto dal resto della divisione di s per 3, con la rete sintetizzata sotto:



Esercizio 2 - Una possibile soluzione

```
module XXX(ad7_ad0, va, mr_, mw_, p, reset_);
  input  p, reset_;
  output va,mr_, mw_;
  inout[7:0]  ad7_ad0;

  reg      SEND_MBR, SEND_MAR, VA, MR_, MW_;
  reg[7:0] MBR, MAR, PROVA;
  assign va=VA, mr_=MR_ , mw_=MW_;
  assign ad7_ad0=( {SEND_MBR,SEND_MAR}==2'B10)?MBR:
                  ( {SEND_MBR,SEND_MAR}==2'B01)?MAR:8'BZ;

  reg[3:0]  STAR, MJR;
  parameter S0=0,S1=1,S2=2,
             read0=3,read1=4,read2=5,readF=6,
             write0=7,writel=8,write2=9,write3=10,
             write4=11,writeF=12;

  always @(posedge p or negedge reset_)
    if (reset_==0) begin VA=0; MR_=1; MW_=1; STAR=S0; end
    else #7
      casex(STAR)
        S0: begin MBR='HAA; MAR=231; MJR<=S1; STAR<=write0; end
        S1: begin MJR<=S2; STAR<=read0; end
        S2: begin PROVA<=MBR; STAR<=S2; end

// Microsottoprogrammi

    read0:    begin SEND_MBR<=0; SEND_MAR<=1; STAR<=read1; end
    read1:    begin VA<=1; STAR<=read2; end
    read2:    begin VA<=0; SEND_MAR<=0; MR_<=0; STAR<=readF; end
    readF:    begin MBR<=ad7_ad0; MR_<=1; STAR<=MJR; end

    write0:   begin SEND_MBR<=0; SEND_MAR<=1; STAR<=writel; end
    writel:   begin VA<=1; STAR<=write2; end
    write2:   begin VA<=0; SEND_MBR<=1; SEND_MAR<=0;
               STAR<=write3; end
    write3:   begin MW_<=0; STAR<=write4; end
    write4:   begin MW_<=1; STAR<=writeF; end
    writeF:   begin SEND_MBR<=0; STAR<=MJR; end

    endcase
endmodule
```