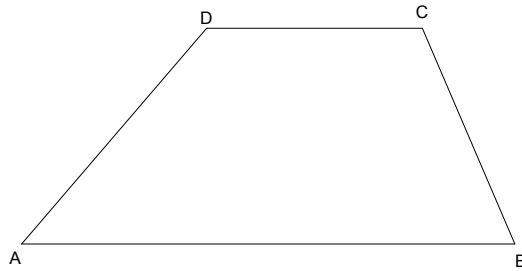


Esercizio 1

Sia dato un piano cartesiano a coordinate intere, sul quale insistono quattro punti A,B,C,D, che individuano i vertici di un trapezio, disposti come in figura. Sintetizzare la rete che prende in ingresso le coordinate dei punti, rappresentate su n bit in complemento alla radice, e produce in uscita:

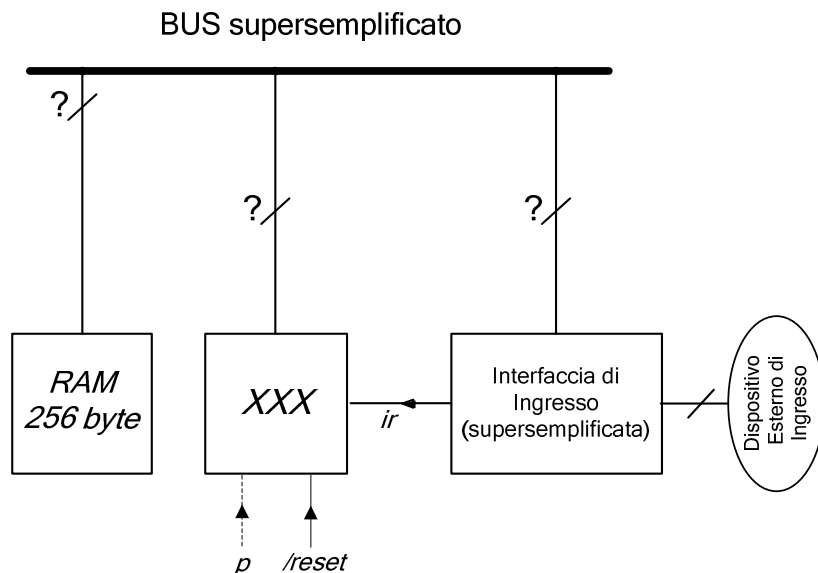
- l'area del trapezio, sotto le ipotesi che il medesimo abbia le basi AB e CD parallele all'asse delle ascisse;
- un flag *error* che vale 1 quando l'ipotesi di sopra non è verificata, e 0 altrimenti

Sintetizzare qualunque rete non descritta esplicitamente a lezione.



Esercizio 2

L'Unità XXX, che gestisce una interfaccia di ingresso e una RAM da 256 byte, si evolve come segue: i) effettua due prelievi di un byte ciascuno dall'interfaccia di ingresso; ii) interpreta il primo byte prelevato come un indirizzo ed il secondo come un dato e scrive il secondo byte nella RAM all'indirizzo specificato dal primo byte; iii) ripete questi prelievi e queste scritture per 1024 volte, poi si ferma.



L'interfaccia di ingresso è una versione estremamente semplificata dell'interfaccia con handshake vista a lezione, in quanto è gestibile esclusivamente ad interruzione di programma ed è sempre abilitata sia ad essere letta sia ad interrompere. Il tempo di risposta delle reti interne all'interfaccia è molto molto molto minore di un ciclo del clock.

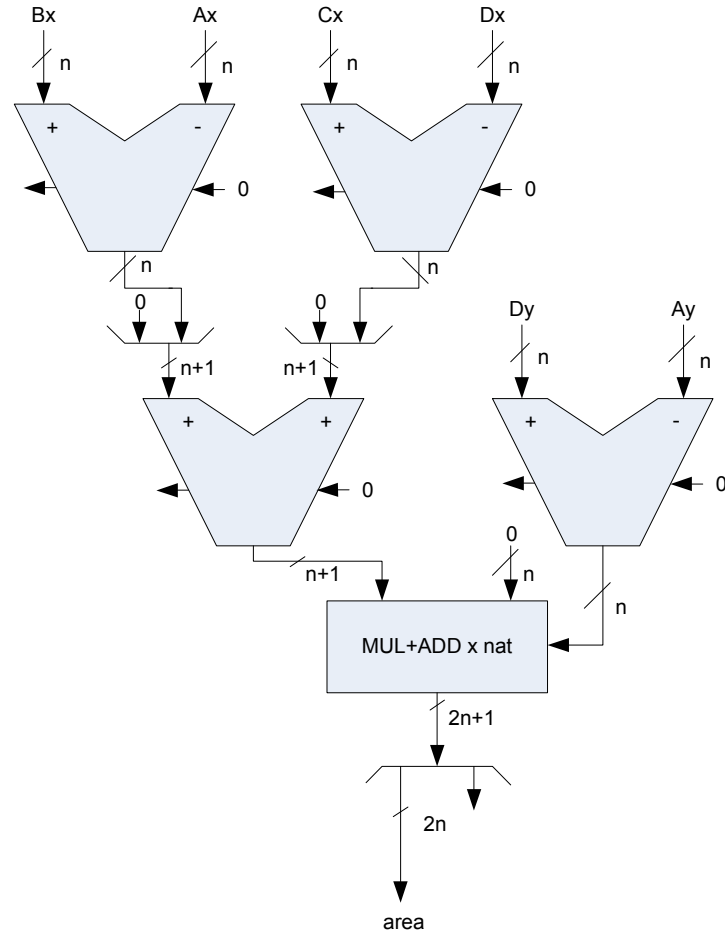
L'Unità XXX non è un processore e quindi non ha spazi di memoria e/o spazi di I/O

Ciò premesso,

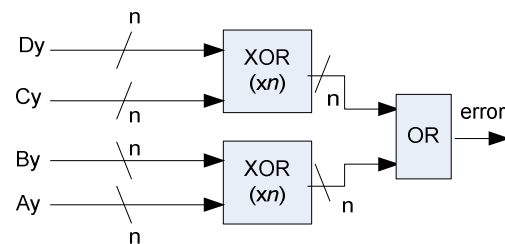
- Si disegni l'interfaccia semplificata, e si specifichino le (poche) connessioni con l'Unità XXX
- Si descriva e si sintetizzi l'Unità XXX.

Soluzione Es 1

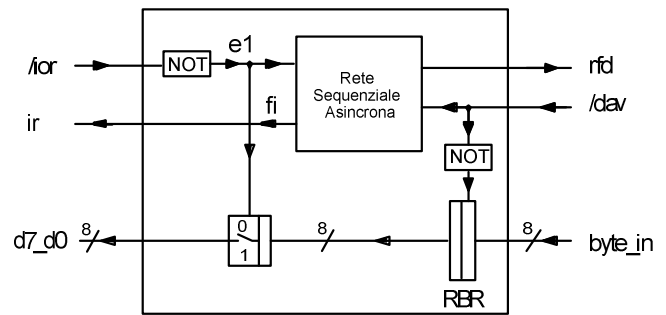
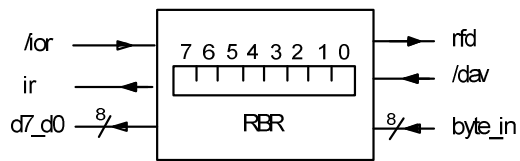
Siano P_x, P_y le rappresentazioni delle coordinate del punto P . La formula dell'area del trapezio è $area = (\overline{AB} + \overline{CD}) \cdot h / 2$, dove $\overline{AB} = B_x - A_x$, $\overline{CD} = C_x - D_x$, $h = D_y - A_y$. Date le ipotesi, $\overline{AB}, \overline{CD}, h$ sono naturali su n bit. La rete che calcola l'area è quindi la seguente:



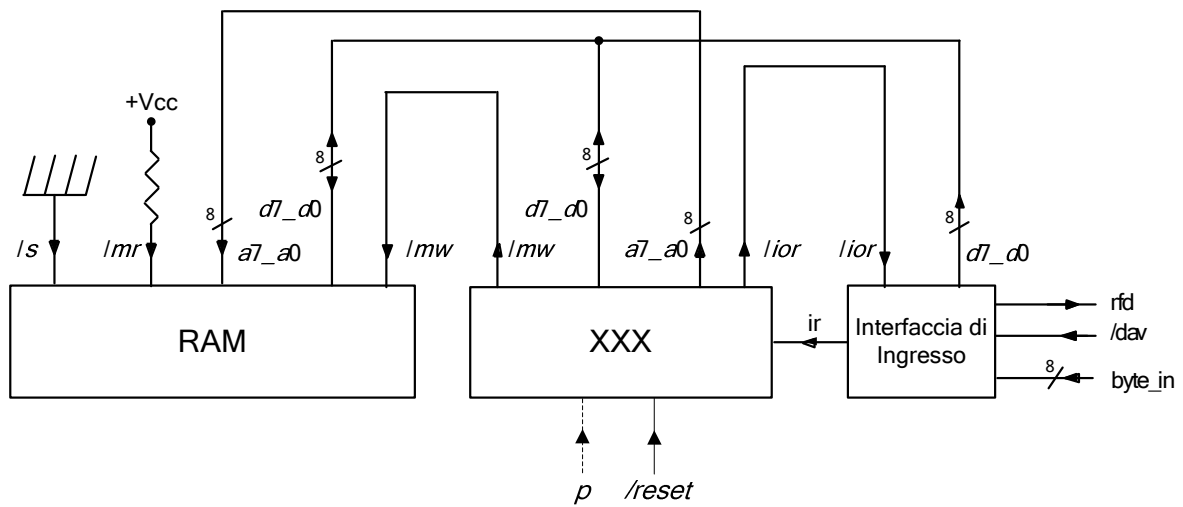
Il flag *error* vale 1 se $C_y \neq D_y$ OR $A_y \neq B_y$. Pertanto, la rete che produce il flag è la seguente:



Una possibile soluzione Es 2



Interfaccia semplificata di ingresso



Connessioni all'Unità XXX

```

//-----
module XXX(a7_a0,d7_d0,ior_,mw_,ir, p,reset_);
input      p, reset_;
output [7:0] a7_a0;
inout  [7:0] d7_d0;
output      ior_, mw_;
input      ir;

reg      IOR_, MW_;    assign ior_=IOR_; assign mw_=MW_;
reg [7:0] MAR;          assign a7_a0=MAR;
reg      DIR;
reg [7:0] MBR;          assign d7_d0=(DIR==1)?MBR:8'BZZ; //FORCHETTA

reg [9:0] COUNT;
reg [3:0] STAR;
parameter S0=0,S1=1,S2=2,S3=3,S4=4,S5=5,S6=6,S7=7,S8=8,S9=9;

always @(posedge p or negedge reset_)
if (reset_==0) begin DIR=0; IOR_=1; MW_=1; COUNT=1023; STAR=S0; end
else #3
casex(STAR)

//Prelievo primo byte
S0: begin DIR<=0; STAR<=(ir==0)?S0:S1; end
S1: begin IOR_<=0; STAR<=S2; end
S2: begin STAR<= S3; end
S3: begin MAR<=d7_d0; IOR_<=1; STAR<=S4; end

//Prelievo secondo byte
S4: begin STAR<=(ir==0)?S4:S5; end
S5: begin IOR_<=0; STAR<=S6; end
S6: begin STAR<= S7; end
S7: begin MBR<=d7_d0; IOR_<=1; DIR<=1; STAR<=S8; end

//Scrittura in RAM
S8: begin COUNT<=COUNT-1; MW_<=0; STAR<=S9; end
S9: begin MW_<=1; STAR<=(COUNT==0)?S9:S0; end
endcase
endmodule

```