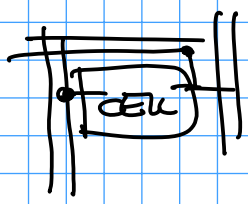
Verilog (VHDL)

RTL
Register transfer language

FPGA 20 anni 8M celle



* 3-5 ingressi
  ↳ 1 bit di uscita
* 1 bit di memoria
* Router

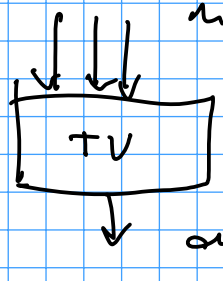
primitive nome (output, input, ...) ;

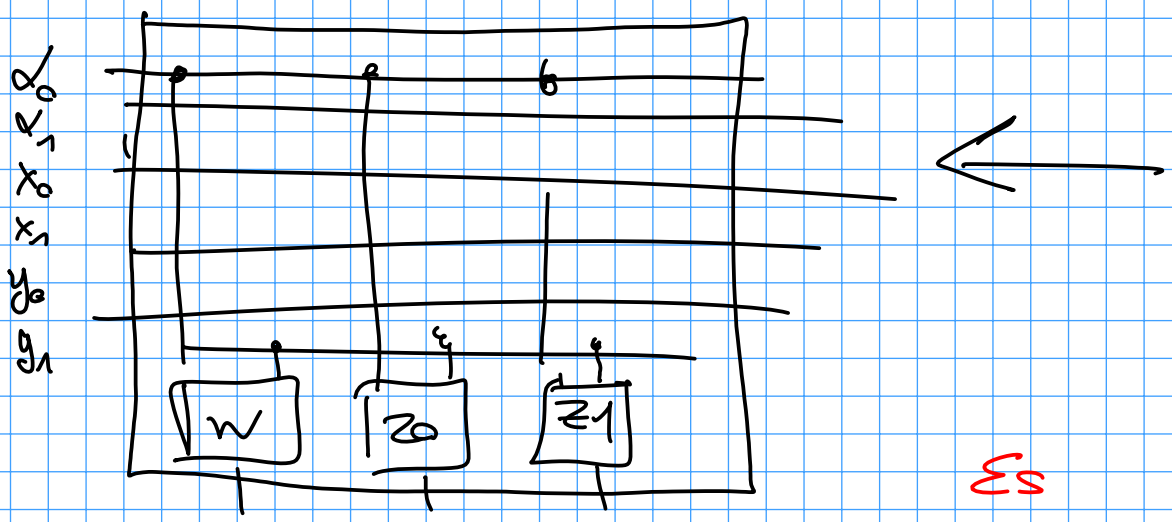
table

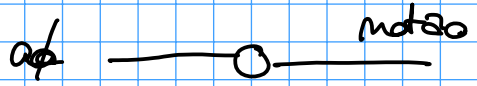
0 ... ; 1

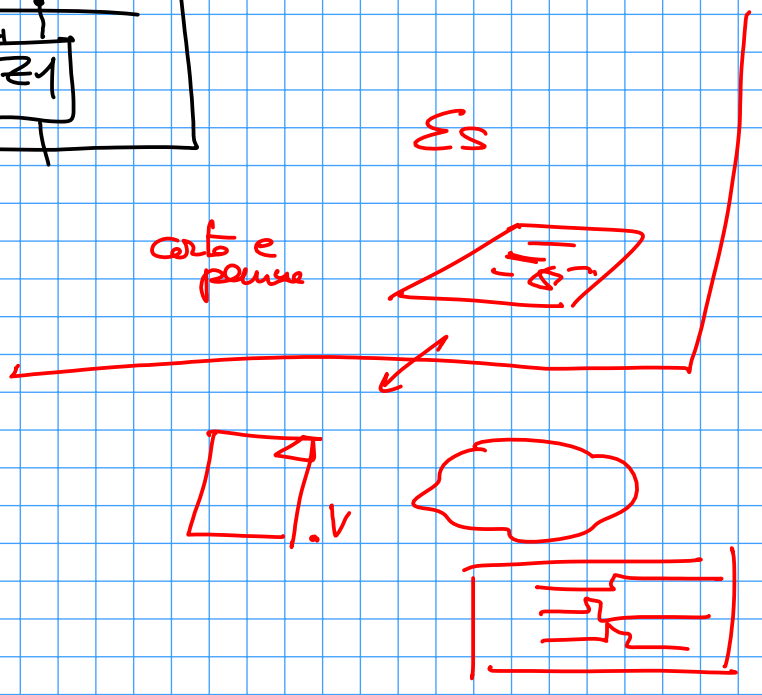
end table

end primitive









ritardi

n°fp

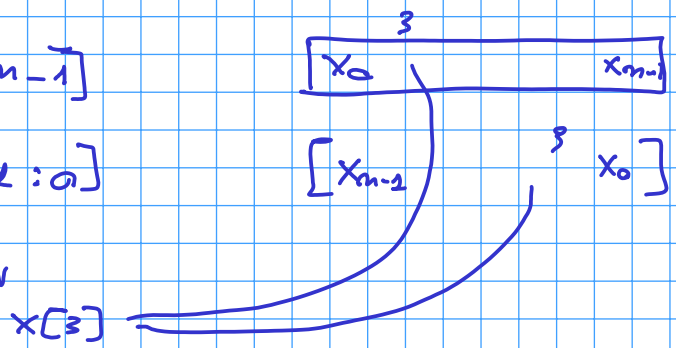
tip. di dato

wire x;

1 bit

x[0:m-1]

x[m-1:0]



[0 :m-1]

1 2 3 base 10

wire

reg

#cifre ' base numero
bit

8 4 2 1

4'b 0110

4' h f 4'b 1111

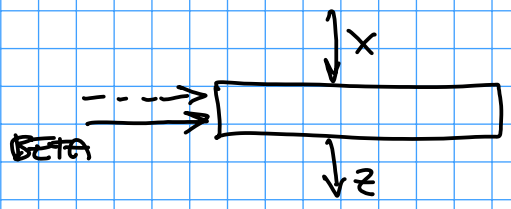
clock

reg c;

initial
begin
c=0;
end

always
begin
#5 c=1;
#1 c=0;
end

while true




```
module reg(z, x, c, B);
  output z;
  input x, c, B;

  reg R;
  initial
    begin
      R = 0;
    end

  always @(negedge c)
    begin
      if(B == 1)
        R = x;
    end
endmodule
```

```
module name (output z, input x[0:N-1]);
  parameter N = 16;

  assign z = ......
endmodule
```

! ~
&& &
|| |

```
test ( );
```

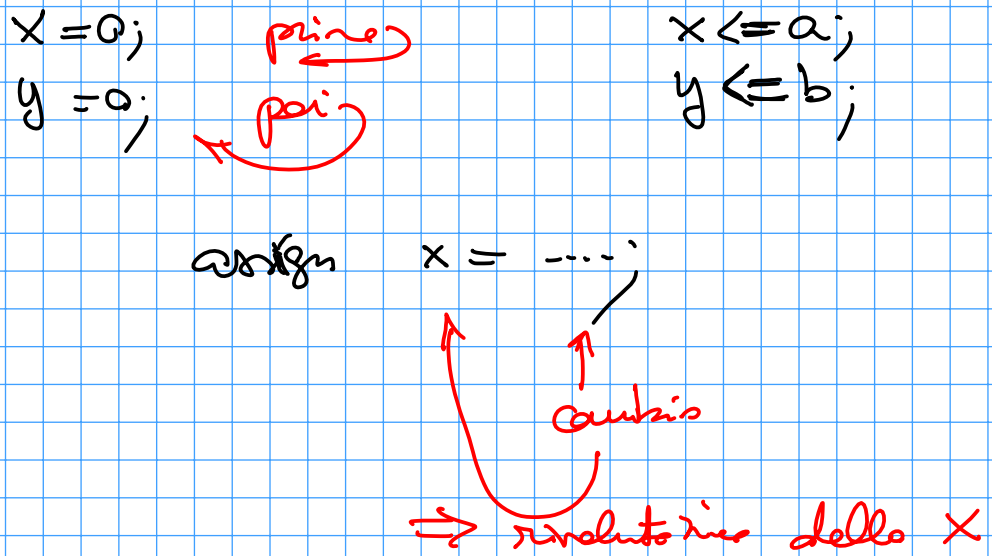
```
wire z;
reg x[0:7];

assign z = &x;
```

name #(8) name_name(_,_)
name #(8) B(_,_)

8
x[8:15]
8
x[0:7]

reg x[0:15]


x = a;
y = a;
prima
poi
x <= a;
y <= b;
assign x = ...;
cambio
⇒ variabile della x

```
if (x==0)
    r= 1;
else
    r=0;
```

```
r = (x==0 ? 1 : 0)
```

```
case ( r )
    val : ——
    val : ——
    default : ——
endcase
```

Xilinx

Vivado (WEB)

iverilog

Altera

Quartus WEB

apt-get install iverilog

gtkwave