

## Esercizio 2: Verilog

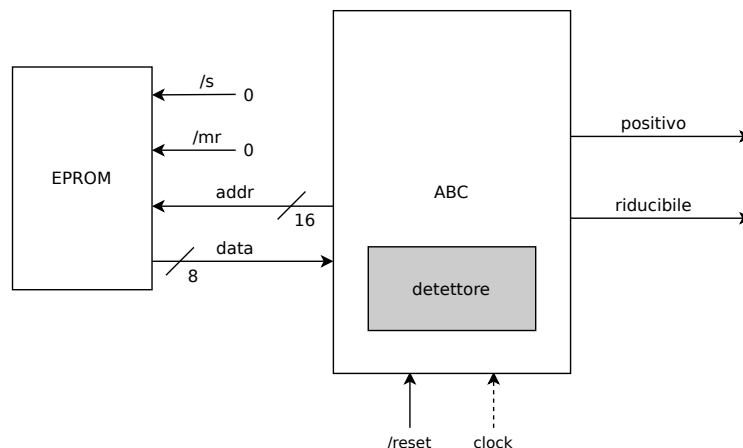


Figura 1: Schema del sistema

L'unità ABC, in Figura 1, si comporta con un periodo di 10 cicli di clock come segue:

1. Legge una locazione da una memoria EPROM 16Kx8, a partire dall'indirizzo 'h0000;
2. Interpreta gli 8 bit come la codifica di due cifre in base 10 (MSD, LSD), ciascuna costituita dai 4 bit alti/bassi del byte letto;
3. Detto  $A$  il numero naturale letto al punto precedente, lo interpreta come rappresentazione del numero intero  $a$  su due cifre in base 10;
4. Imposta le due uscite `positivo` e `riducibile` sulla base del fatto che  $A$  sia o non sia la rappresentazione di un numero intero  $a$  positivo e riducibile;
5. Torna al punto 1 accedendo alla locazione successiva.

- Descrivere in Verilog l'unità ABC e sintetizzarla in accordo al modello con parte operativa e parte controllo.  
Si supponga che la parte controllo sia implementata secondo un modello basato su microindirizzi e si scriva la struttura della ROM come commento alla parte controllo medesima.
- Sintetizzare come modulo a parte la rete combinatoria `detettore` utilizzata per il calcolo del risultato. Se lo si ritiene, si utilizzino le reti combinatorie fornite in `reti_standard.v`.

Si supponga che la memoria sia sufficientemente veloce da non necessitare di stati di wait. Il valore delle uscite al reset deve essere non determinato (1'bX).

Note sulla sintesi della rete combinatoria:

- Non sono sintesi di reti combinatorie, e non saranno quindi ritenute valide, descrizioni contenenti operatori aritmetici e/o di relazione predefiniti del Verilog, e.g. `+`, `-`, `*`, `/`, `<`, `>` etc.
- Sono invece validi a questo scopo gli operatori di concatenamento, (dis)uguaglianza, assegnamenti a più vie, utilizzo di altre reti che o a) rispettano questi criteri o b) provengono da `reti_standard.v`
- È sufficiente che la rete combinatoria sia così sintetizzata in almeno uno dei file consegnati