

Esercizio 2: Verilog

Il metodo della bisezione è un algoritmo numerico iterativo per trovare le radici di una funzione $f(x)$. Siano dati $f(x)$ e gli estremi di un intervallo $[l_0, r_0]$ tali che

- $f(x)$ è continua ed ha una sola radice $x_0 \in \mathbb{N}$ e $\in [l_0, r_0]$ tale che $f(x_0) = 0$,
- $f(l_0) < 0$ e $f(r_0) \geq 0$.

L'algoritmo consiste quindi nel ridurre l'intervallo iterativamente fino a trovare tale radice. Sia $m = \lfloor \frac{l_i + r_i}{2} \rfloor$, ossia il punto intermedio dell'intervallo corrente $[l_i, r_i]$. Se $f(m) < 0$ l'intervallo viene aggiornato come $[l_{i+1}, r_{i+1}] = [m, r_i]$, mentre se $f(m) \geq 0$ l'intervallo viene aggiornato come $[l_{i+1}, r_{i+1}] = [l_i, m]$.

Ai fini di questo esercizio, si considerino polinomi di secondo grado $f(x) = x^2 - b \cdot x - c$ di cui sono forniti i parametri naturali b e c , tale che $x_0 \in \mathbb{N}$, come da esempio in Figura 1. Si noti che sotto queste ipotesi e seguendo questo algoritmo, se $l_i = r_i - 1$ allora $x_0 = r_i$.

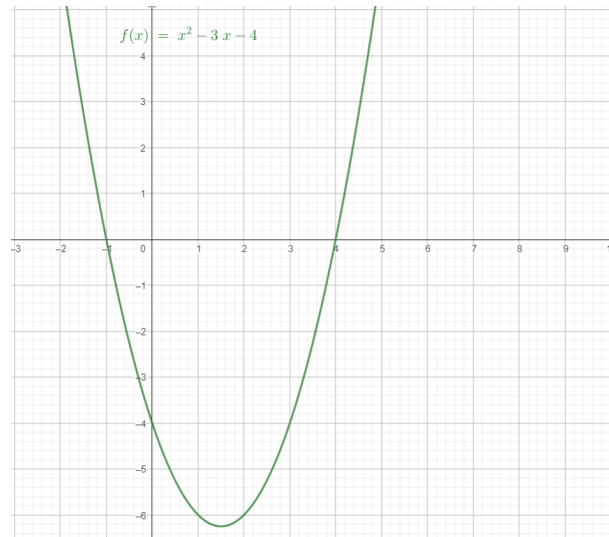


Figura 1: Piano cartesiano, con $f(x) = x^2 - 3x - 4$. Realizzato con www.geogebra.org

L'Unità ABC in Figura 2 calcola la radice x_0 per la funzione $f(x)$ e l'intervallo di ricerca forniti dalla rete consumatore. La rete consumatore fornisce i parametri del calcolo prima e durante tutto il tempo in cui imposta soc a 1, come indicato dalla temporizzazione in Figura 3, mentre la rete ABC risponde con la radice x_0 secondo le solite procedure dell'handshake soc/eoc. Si assuma che i parametri forniti in ingresso siano numeri naturali e sempre conformi alle ipotesi di cui sopra.

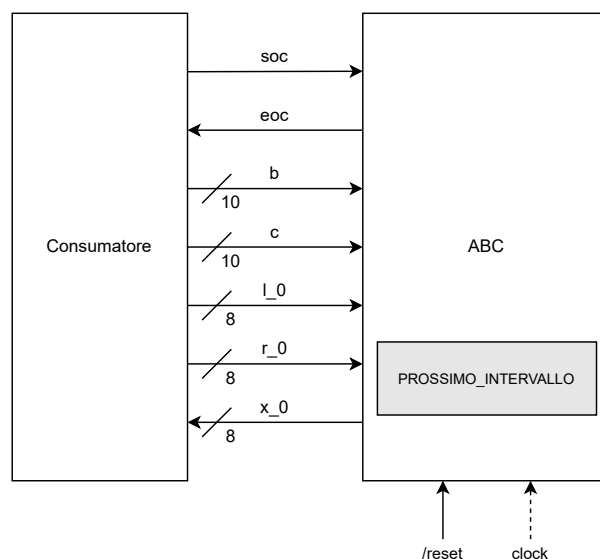


Figura 2: Schema del sistema

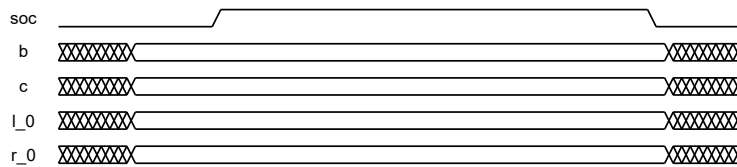


Figura 3: Diagramma di temporizzazione relativo ai parametri in ingresso.

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 PROSSIMO_INTERVALLO utilizzata per calcolare $[l_{i+1}, r_{i+1}]$ a partire da $[l_i, r_i]$. Se lo si ritiene, si utilizzino le reti combinatorie fornite nel file reti_standard.v.

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.

I file testbench.v e reti_standard.v sono al link: <http://tinyurl.com/4upfyxmc>