

Compito di linguaggi di descrizione dell'hardware

Esercizio 1

Si realizzi un modello comportamentale in VHDL di un componente che può lavorare sia da latch che da flip-flop. In particolare, sia d l'ingresso con il dato da campionare e clk l'ingresso con il segnale di clock. Un ulteriore ingresso m seleziona il modo di funzionamento, se $m = 0$ il componente funziona da D-latch trasparente e se $m = 1$ funziona da FF. Il latch è in sample nella fase alta del clock e il FF campiona sul fronte di salita. L'uscita sia q . Il FF ha anche come parametri generici il tempo di setup τ_{DC} e il tempo di risposta τ_{CQ} (pt. 5.0).

Soluzione

https://www.edaplayground.com/x/hNT_

Esercizio 2

Si descriva al livello comportamentale in VHDL un componente combinatorio che riceve in ingresso una parola (di tipo `std_logic`) b di 8 bit che rappresenta un numero intero senza segno B . Compito della rete è produrre in uscita il valore $B/3$ senza usare le operazioni di moltiplicazione e divisione. A questo riguardo si noti che $B/3$ può essere approssimato come $B * 43/128$ che può essere espresso come $(B * 32 + B * 8 + B * 2 + B)/128$ e descritto in VHDL tramite opportune operazioni di shift e di somma. Si dimensionino le variabili temporanee in modo da evitare perdite di informazione (a parte chiaramente il troncamento dello shift corrispondente alla divisione per 128) (pt. 5.0).

Soluzione

<https://www.edaplayground.com/x/LhYe>

Esercizio 3

Si consideri il seguente algoritmo:

0. $u0 := a + b;$
1. $u1 := c + u0;$
2. $u2 := u1 * d;$
3. $u3 := e + f;$
4. $u4 := g + h;$
5. $u5 := u2 * i;$
6. $u6 := u3 * k;$
7. $u7 := u5 * u6;$
8. $u8 := u4 + u7;$

Si tracci il DFG dell'algoritmo. Utilizzando l'ipotesi di ciclo singolo, si esplori lo spazio dei possibili scheduling a latenza minima. Se ne determinino due che minimizzano il numero di risorse complessive, ma che presentano costi, come

occupazione di area, diversi. Si commentino brevemente tali soluzioni individuando quella di costo minore (pt. 5.0).

Soluzione

Si hanno due possibili scheduling a latenza minima che hanno lo stesso numero totale di risorse. Quello con 2 adder e 1 moltiplicatore é preferibile in quanto i moltiplicatori, a parit  della dimensione delle parole in ingresso, hanno una dimensione maggiore degli adder.

