Modelarea în Verilog utilizând blocuri `always` și `initial`

Probleme propuse

Oprițoiu Flavius
flavius.opritoiu@cs.upt.ro

October 9, 2024

Problema 1

Construiți un modul Verilog numit *msd*, utilizând blocuri always, având o intrare *i* pe 5 biți și o ieșire *o* pe 4 biți. Ieșirea *o* are valoarea celei mai semnificative cifre zecimale din reprezentarea numărului fără semn de la intrarea *i*. Exemplu: $i = 23 \rightarrow o = 2$; $i = 9 \rightarrow o = 9$.

Notă: Pentru implementarea problemei descărcați fișierul sursă *msd.v* și, respectiv, fișierul script *run_msd.txt*, modificând doar fișierul sursă Verilog.

Problema 2

Proiectați un modul numit *div3*, având o intrare *i*, pe 4 biți si o ieșire *o*, pe numărul minim de biți necesar, astfel încât la ieșire să fie furnizat câtul împărțirii numărului fără semn de la intrare la 3. Nu se va folosi operatorul de împărțire "/" iar soluția va folosi blocuri always. Exemplu: $i = 10 \rightarrow o = 3$; $i = 2 \rightarrow o = 0$.

Notă: Pentru implementarea problemei descărcați fișierul sursă ▸ *div3.v* și, respectiv, fișierul script ▸ *run_div3.txt*, modificând doar fișierul sursă Verilog.

Problema 3

Proiectați un dispozitiv de numărare a biților de 1 din reprezentarea numărului conectat la intrarea *i*. Modulul se va numi *cnt1s*, și are o intrare *i*, pe 6 biți și o ieșire *o* pe numărul minim necesar de biți. Construiți acest modul utilizând blocuri always.

Notă: Pentru implementarea problemei descărcați fișierul sursă *cnt1s.v* și, respectiv, fișierul script *run_cnt1s.txt*, modificând doar fișierul sursă Verilog.

Problema 4

Construiți un dispozitiv numit *seq3b*, având o intrare i pe 4 biți și o ieșire o pe 1 bit. Ieșirea va fi activă daca în numărul binar de la intrare exista o secvență de 3 biți consecutivi având aceeași valoare. Exemplu: $i = 14 \rightarrow o = 1$; $i = 9 \rightarrow o = 0$.

Notă: Pentru implementarea problemei descărcați fișierul sursă *seq3b.v* și, respectiv, fișierul script *run_seq3b.txt*, modificând doar fișierul sursă Verilog.

Problema 5

Proiectați un modul *mul5bcd* având o intrare i pe 4 biți și două ieșiri d și u ambele pe 4 biți. La intrarea i se primește o cifră BCD pe 4 biți iar modulul va furniza la ieșiri rezultatul înmulțirii cifrei i cu cifra 5 în BCD: ieșirea d va reprezenta cifra zecilor pentru rezultat iar ieșirea u va reprezenta cifra unităților pentru rezultat.
Exemplu: $i = 3(0011) \rightarrow d = 1(0001)$, $u = 5(0101)$;
$i = 9 \rightarrow d = 4(0100)$, $u = 5(0101)$.

Notă: Pentru implementarea problemei descărcați fișierul sursă ▸ *mul5bcd.v* și, respectiv, fișierul script ▸ *run_mul5bcd.txt*, modificând doar fișierul sursă Verilog.

Problema 6

Proiectati unitatea *text2nibble*, având intrarea *i* pe 8 biți și ieșirea *o* pe 4 biți. Intrarea *i* primește un caracter ASCII. Dacă caracterul este cifră zecimală ('0' la '9'), furnizează la ieșire valoarea cifrei, altfel furnizează valoarea 15.

Caracter ASCII	Valoarea caracterului ASCII (baza 16)	Iesire o
'0'	30 (00110000)	0
'1'	31 (00110001)	1
'2'	32	2
'3'	33	3
'4'	34	4
'5'	35	5
'6'	36	6
'7'	37	7
'8'	38	8
'9'	39 (00111001)	9

Problema 6 (contin.)

Notă: Pentru implementarea problemei descărcați fișierul sursă *text2nibble.v* și, respectiv, fișierul script *run_text2nibble.txt*, modificând doar fișierul sursă Verilog.

Problema 7

Proiectati un registru pe 4 biți, numit *r4b*. Registrul are facilități de încarcare paralelă a conținutului de la intrarea *d* pe 4 biți și de deplasare la dreapta a conținutului cu 1 bit, caz în care valoarea bitului mai semnificativ este primită de la intrarea pe un bit *sh_in*. Registrul are intrările de comandă, sincrone, *ld* - care activează încarcarea paralelă și, repsectiv, *sh* - care declanșază deplasarea la dreapta. Registrul va avea ieșirea *q*, pe 4 biți, reprezentând conținutul registrului. Pentru implementare modificați fișierul sursă ▸ *r4b.v*.

Notă: Dispozitivele secvențiale sincrone, cum este și registrul din problemă, vor avea implicit intrările de tact *clk* și o linie de inițializare, asincronă, activă la 0, *rst_b*, excepție cazul în care sunt specificate alte semnale.