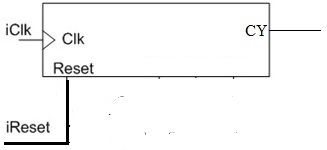
**Laborator 5**

**Modelare/simulare divizoare de frecvenţă**

* 1. **Modelare unui divizor de frecvenţă modulo 1000**

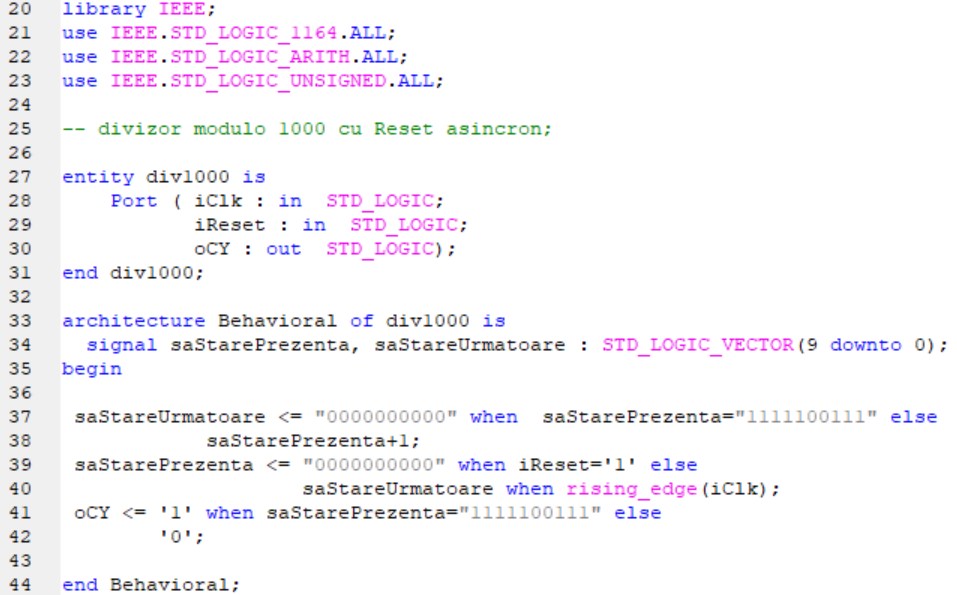
Spre deosebire de numărător, la divizorul de frecvenţă contează numai lungimea ciclului de numărare ṣi eventual factorul de umplere al semnalului la ieṣire dar nu are nici o importanţă modul în care se face asignarea (codificarea) stărilor. Schema bloc este cea din Fig. 5.1



**Fig. 5.1**

Divizorul va avea 1000 de stări de la 0 la 999. Pentru simplitate vom considera o codificare binară naturală pe 10 biţi. Ultima stare va avea asignată combinaţia binară **1111100111**.

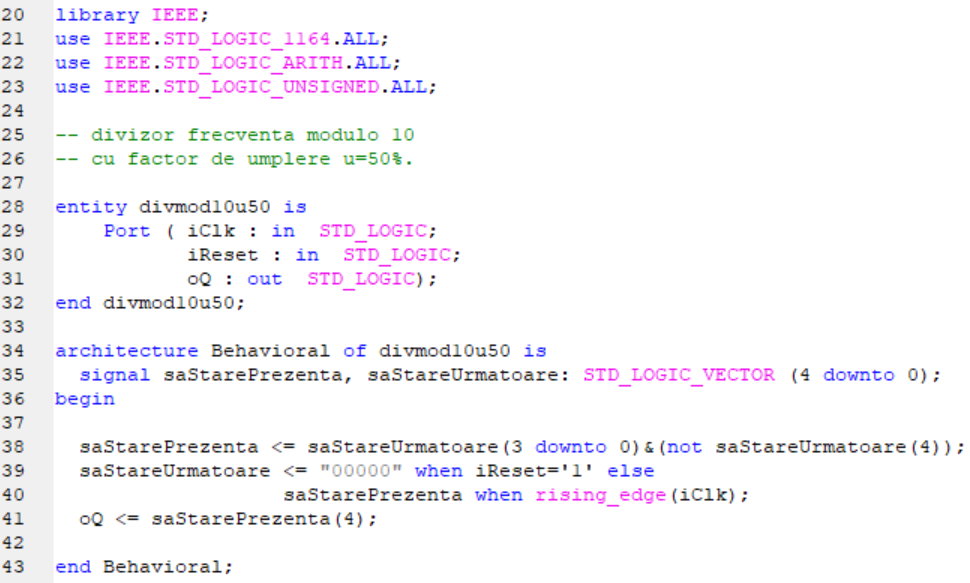
Programul VHDL din Fig. 5.2 permite modelarea ṣi verificarea funcţională a divizorului.



**Fig. 5.2**

* 1. **Modelarea unui divizor de frecvenţă modulo 10 cu factor de umplere 50%**

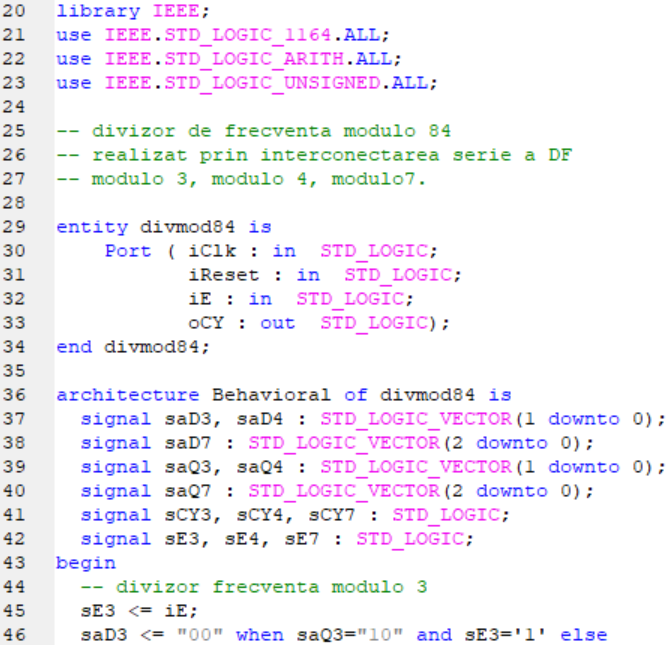
Dacă se utilizează metoda de proiectare dela punctul precedent ṣi utilizăm o asignare binară naturală a stărilor, nu vom putea obţine factorul de umplere dorit (ce factor de umplere putem obţine?). Pentru a obţine factorul de umplere 50% , putem folosi o asignare Johnson pe 5 biţi care are avantajul că este simetrică pentru bitul cel mai semnificativ, deci putem culege semnalul de pe bitul Q(4). Implementarea se va face cu un registru de deplasare pe 5 biţi folosind un program ca cel din Fig. 5.3.

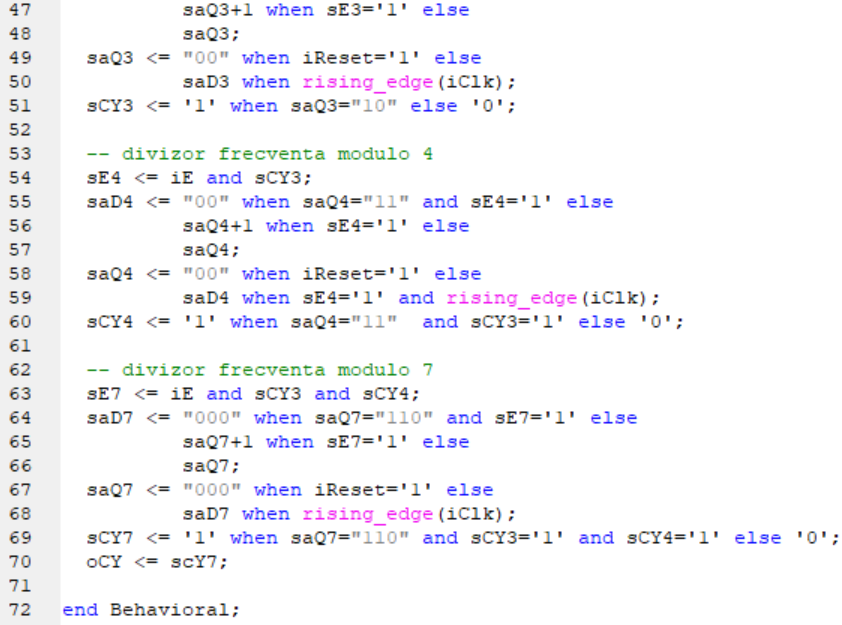
****

**Fig. 5.3**

* 1. **Interconectarea divizoarelor de frecvenţă**

Pentru exemplificare vom considera 3 divizoare de frecvenţă (modulo 3, modolo 4, modulo 7). Prin interconectarea serie a acestor divizoare se obţine un divizor de frecvenţă modulo 3x4x7=84. Interconectarea se va face conform schemei din Fig. 5.4. O varianta de program VHDL este prezentată în Fig. 5.5.





**Fig. 5.5**

**5.4 Desfăṣurarea lucrării**

1. **Se vor sintetiza schemele prezentate mai sus.**
2. **Se va simula funcţionarea fiecărei scheme ṣi se vor formula concluziile corespunzătoare.**
3. **Se va sintetiza un divizor de frecvenţă modulo 10 cu u=10%, 20%, 30%, 40% 60%.**