**4.Componentele utilizate si semnalele lor**

**MASTER**

Este componenta principala a proiectului care leaga toate celelaltecomponente intre ele si are practic rol de organigrama

**Semnificatia semnalelor de intrare :**

-sw este un vector de 4 biti primeste semnalele a 4 switch-uri ale placii FPGA

-ok1,back1,exi1 sunt3 biti care primesc semnalele a trei butoane ale placii FPGA

-clk este semnalul de clock al placii FPGA

**Semnificatia semnalelor de iesire :**

-afisor este un vector de 7 biti care transmite semnalul de afisare pe un afisor cu 7 segmente

-segments un vector de 8 biti care allege afisorul pe care afisor se va afisa informatia de la semnalul “afisor”(functioneaza ca un decodificator)

**Clock-uri**

-CLK02s este un clock careare o perioada de 0,4 secunde

-Clk1khz este un clock careare o perioada de 0,2 milisecundesecunde

-CLk100khz este un clock care are o frecventade50 de khz

-2 memorii RAM: una pentru numarul de bancnote din bancomat si una pentru PIn-urile si soldurile din conturile bancare, respective pentru optiunea admin

-un registru care salveaza ultima valoare primita mai mica decat 10000

-un algoritm greedy de impartire a sumei primite in bancnote. Daca se poate imparti suma primita in bancnote, atunci se vor modifica memoriile RAM, iar daca nu se poate, se afiseaza mesaj de eroare

-un algoritm de introducere bancnote, care intai te va intreba cate bancnote de 500 de lei vrei sa introduci, apoi de 200 si tot asa pana la 1, iar apoi iti va spune daca suma din bancomat nu depaseste 10000. Daca nu depaseste va afisa un mesaj de ok, iar daca da, va afisa un mesaj de eroare.

-un algoritm de interogare bancnote, care va afisa pe rand la ce tip de bancnota e curent in faza de afisare si cate bucati se afla in bancomat. Cand se apasa ok va trece la urmatorul tip de bancnota si numarul de bucati.

-3 clock-uri diferite fata de cel al placii: unul de 0,4 secunde, unul de 500Hz si unul de 50Khz

-un debouncer de butoane care lasa butonul pornit doar o perioada foarte mica

-un algoritm "read\_integer" care primeste semnalele de la patru switch-uri si construieste un numar intreg mai mic decat 10000

-un algoritm "number\_to\_digits" care primeste un intreg mai mic decat 1000 si il imparte in 4 integeri care reprezinta cifra miilor,sutelor,zecilor si unitatilor numarului primit

-un algoritm "Digit\_to\_BCD" care primeste o cifra si o afiseaza pe afisor

-un program "Display\_number" care primeste un numar mai mic decat 10000, il transforma in 4 cifre de tip digit si apeleaza "Digit\_to\_BCD" de 4 ori

-un program "master\_display" care apeleaza de 8 ori "Digit\_to\_BCD" si conecteaza semnalele primite la afisoare