

Aplicatie cod cifru pentru securizarea dulapurilor

Samoila Denis-Valentin

Silaghi Andrei

Grupa: 30213

Indrumator: Dr. Ing. Dragos Florin Lisan

APLICATIE COD CIFRU PENTRU SECURIZAREA DULAPURILOR

1. Specificatii

Sa se implementeze o aplicatie care permite utilizatorului adaugarea unui cifru din 3 caractere distincte pentru securizarea unui dulap (asemanator dulapurilor folosite la vestiarele de sport, mall, etc)

Cerinte functionale:

1. Un led LIBER_OCUPAT va avea functia de a semnala faptul ca dulapul este liber(led stins) sau ocupat(led aprins)
2. Utilizatorul va apasa un buton ADAUGA_CIFRU pentru a semnala inceperea introducerii codului. Un led INTRODUC_CARACTERE se va aprinde pentru a marca starea
3. Utilizatorul va adauga pe rand 3 caractere cu ajutorul butoanelor UP si DOWN
4. Caracterele sunt curprinse in intervalul 0-1-...-8-9-A-B-...-F
5. Caracterul curent este afisat pe SSD
6. Pentru trecerea la urmatorul caracter utilizatorul va apasa butonul ADAUGA_CIFRU
7. Caracterul anterior introdus ramane afisat
8. Urmatorul caracter este vizibil pe afisaj pe pozitia urmatoare

9. După introducerea celui de al treilea caracter, la apăsarea butonului ADAUGA_CIFRU, afișajul SSD se va stinge iar cifru va fi în starea blocat prin aprinderea ledului LIBER_OCUPAT.

10. Ledul INTRODUC_CARACTERE se va stinge

11. Existența unui buton/switch RESET în timpul introducerii cifrului pentru revenire în starea inițială (ledul LIBER_OCUPAT se va stinge, afișajul SSD este gol, ledul INTRODUC_CARACTERE se va stinge)

12. Utilizatorul va apăsa butonul/switch ADAUGA_CIFRU pentru a începe introducerea codului pentru deblocarea cifrului

13. Se vor relua pașii 2-8

14. La introducerea ultimului caracter, la apăsarea butonului ADAUGA_CIFRU se va face verificarea, dacă codul introdus corespunde cu codul anterior.

15. În cazul de egalitate, ledul LIBER_OCUPAT se va stringe, ledul INTRODUC_CARACTERE se va stinge, afișajul SSD se golește

16. În cazul de inegalitate, ledul LIBER_OCUPAT va rămâne aprins, ledul INTRODUC_CARACTERE se va stinge, afișajul SSD se golește.

Cerinte non-functionale:

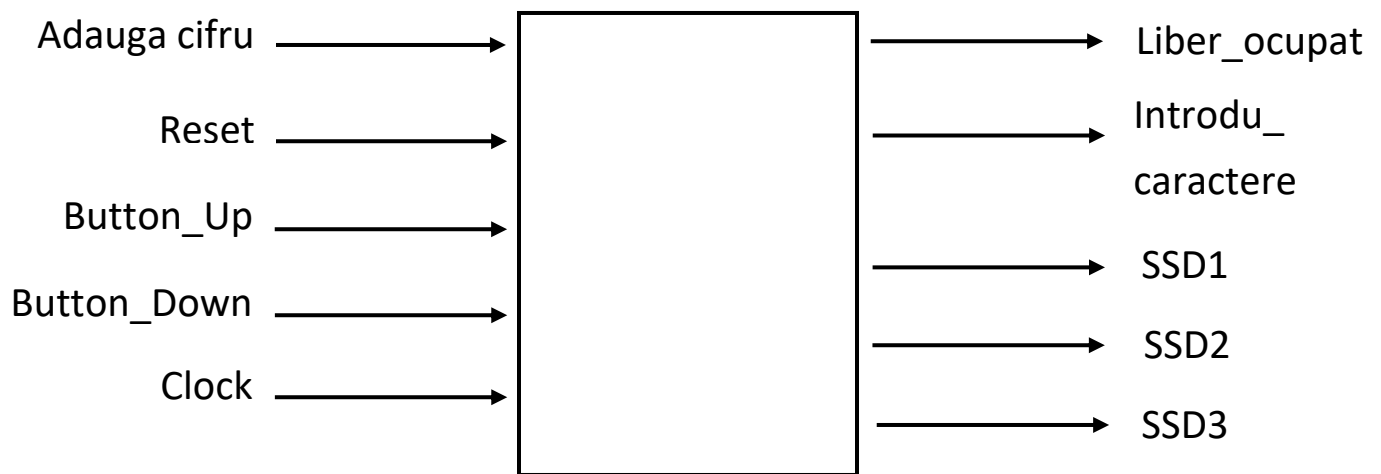
- Implementare pe placuta
- Utilizare SSD
- Utilizare switch, led, butoane

Exemplu use case:

Utilizatorul alege un dulap cu ledul Liber_ocupat stins. Apasă pe butonul ADAUGA_CIFRU pentru a introduce caracterele. Caracterul "0" este vizibil pe SSD. Introduce primul caracter "2" prin apăsarea de 2 ori a butonului DOWN. Pe SSD se modifică afișajul o dată cu apăsarea butonului și anume: 0->1->2. Utilizatorul apasă din nou pe ADAUGA_CIFRU pentru a introduce al doilea caracter "1". Utilizatorul apasă din nou pe ADAUGA_CIFRU pentru a introduce al doilea caracter "3". Utilizatorul apasă din nou ADAUGA_CIFRU, codul este salvat, conținutul SSD este gol, ledul LIBER_OCUPAT este aprins, ledul INTRODUC_CARACTERE se va stinge.

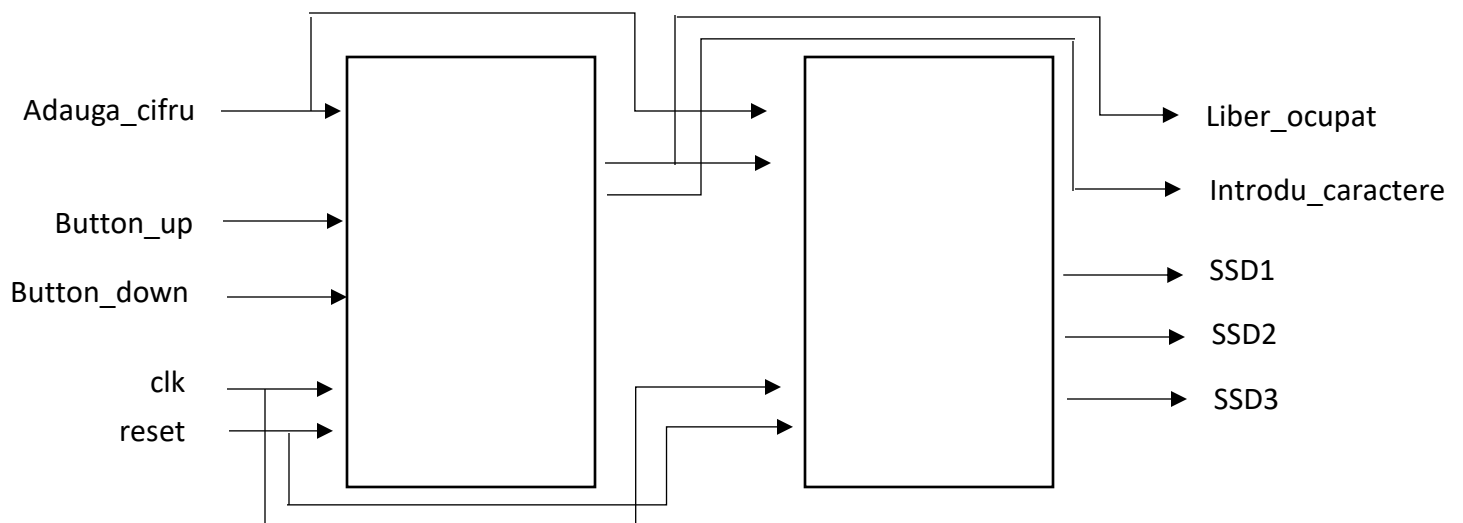
2. Proiectare

2.1 Schema bloc



2.2 Unitatea de control si Unitatea de executie

2.2.1 Maparea intrarilor si iesirilor cutiei mari pe cele doua componente UC si UE



Intrari de date: button_up, button_down

Intrari de control: reset, clock, adauga cifru

Iesiri de date: ssd1, ssd2, ssd3

Iesiri de control: adauga_cifru, liber_ocupat

2.2.2 Determinarea resurselor

Resurse

1. Debouncer

Acesta inlatura semnalele suplimentare care apar la apasarea unui buton si care pot afecta functionarea sistemului



Avem un semnal de intrare btn generat prin apasarea unui buton de catre utilizator, semnal de intrare R de reset generat de asemenea de catre utilizator, semnal de intrare clk reprezentand semnalul de clock al sistemului; semnalul de iesire btndb trimite semnalul mai departe spre UE

2. Numarator_button_up/down

Prin intermediul acestui numarator se selecteaza cifra care trebuie introdusa, parcurgand crescator sau descrescator cifre de la 0 la F

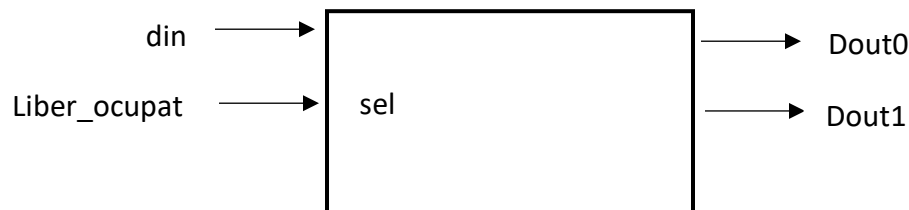


Semnalele de intrare Btn_up si Btn_down, ce provin de la butoane apasate de utilizator, selecteaza cifra prin parcurgerea crescatoare sau descrescatoare; semnalul de intrare reset ce provine de la UE reseteaza numaratorul cu valoarea 0

Semnalul de iesire pe 4 biti dout transmite catre UE cifra care trebuie memorata

3. Demultiplexor 1:2 pe 4 biti

Cu ajutorul acestuia se selecteaza setul de registri in care se vor stoca cifrele introduse.

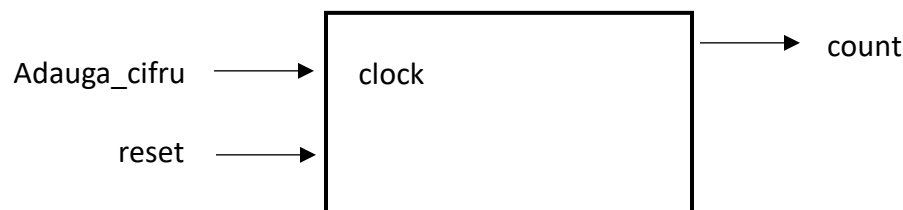


Intrarea din primește un semnal pe 4 biti provenit de la numaratorul button up/down, intrarea sel primește semnalul liber_ocupat generat de UC, cu ajutorul caruia se selecteaza iesirile corespunzatoare(sel=0->dout0, sel=1-> dout1).

Iesirile dout0 si dout1 sunt pe 4 biti si trimit semnale mai departe in UE

4. Numarator modulo 0-3

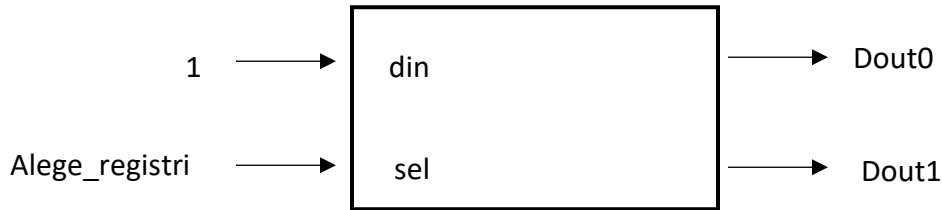
Acest numarator selecteaza cifra din cod asupra careia efectuam modificari la momentul respective: 0->dulap liber/ocupat, nu s-a apasat adauga_cifru, iar introdu_caractere este stins; 1,2,3->se introduc prima cifra, a doua cifra respective a treia cifra din cod



Intrarea clock primește semnalul generat prin apasarea butonului adauga_cifru; intrarea reset este conectata la butonul de reset care poate fi apasat de utilizator; iesirea count pe 4 biti trimite un semnal mai departe in UE pentru selectarea cifrei corespunzatoare din cod

5. Demultiplexor 1:2 pe 1 bit

Acest demultiplexor genereaza semnale care selecteaza registri ce urmeza a fi resetati



Intrarea din se afla mereu pe 1 logic, intrarea sel primeste semnalul provenit de la iesirea alege_registri a UC, iar iesirile dout0 si dout1 trimit semnale in UE spre setul de registri corespunzatori spre a fi resetati

6. Demultiplexor 1:4 pe 4 biti

Prin intermediul celor 2 dmux-uri existente se selecteaza pentru fiecare cod cifra corespunzatoare iesirilor din multiplexor



Intrarea din pe 4 biti primeste semnalul venit de la dmux 1:2 pe 4 biti, mai exact cifra ce trebuie introdusa; intrarea sel pe 2 biti primeste semnalul de iesire venit de la numaratorul modulo, selectandu-se astfel registrul in care se va stoca numarul; iesirile dout pe 4 biti se conecteaza la registrii corespunzatori

7. Demultiplexor 1:4 pe 1 bit

Sunt tot in numar de 2 si se ocupa de resetarea registrilor



Intrarea din primește semnalul venit de la dmux 1:2 pe 1 bit; intrarea sel este pe 2 biti și asemenea dmux-ului precedent, selectează registrul corespunzător, care de data aceasta va fi resetat; ieșirile dout trimit semnale spre registrii, semnale care vor indica resetarea

8. Registri de memorare pe 4 biti

În număr de 6, 3 pentru codul dinaintea blocării dulapului și 3 pentru codul de după blocare; aceștia memorează cifrele din cod



Întrearea clk primește semnalul venit de la clockul sistemului; intrarea R reprezintă comanda pentru reset a registrului și primește semnalul provenit în urma operației „or” între semnalul de reset manual și semnalul reset_reg1, care reșetează primul set de registre; întrearea L reprezintă comanda de load pentru memorarea cifrei introduse, iar d_in primește cifra care trebuie memorată venită de la dmux 1:4 pe 4 biti.

Ieșirea d_out pe 4 biti trimite mai departe în UE un semnal către multiplexoare și către UC pentru comparare

9. Multiplexor 2:1 pe 4 biti

3 multiplexoare: câte unul pentru fiecare cifră din coduri (ex: primul mux face selecția între prima cifră a primului cod și prima cifră din al doilea cod)



Intrearea din0 corespunde cifrei primului cod, iar din1 corespunde cifrei celui de-al doilea cod; intrearea sel face selectia intre din0 si din1 si trimite semnalul corespunzator pe iesirea dout, care va merge mai departe catre afisoarele ssd

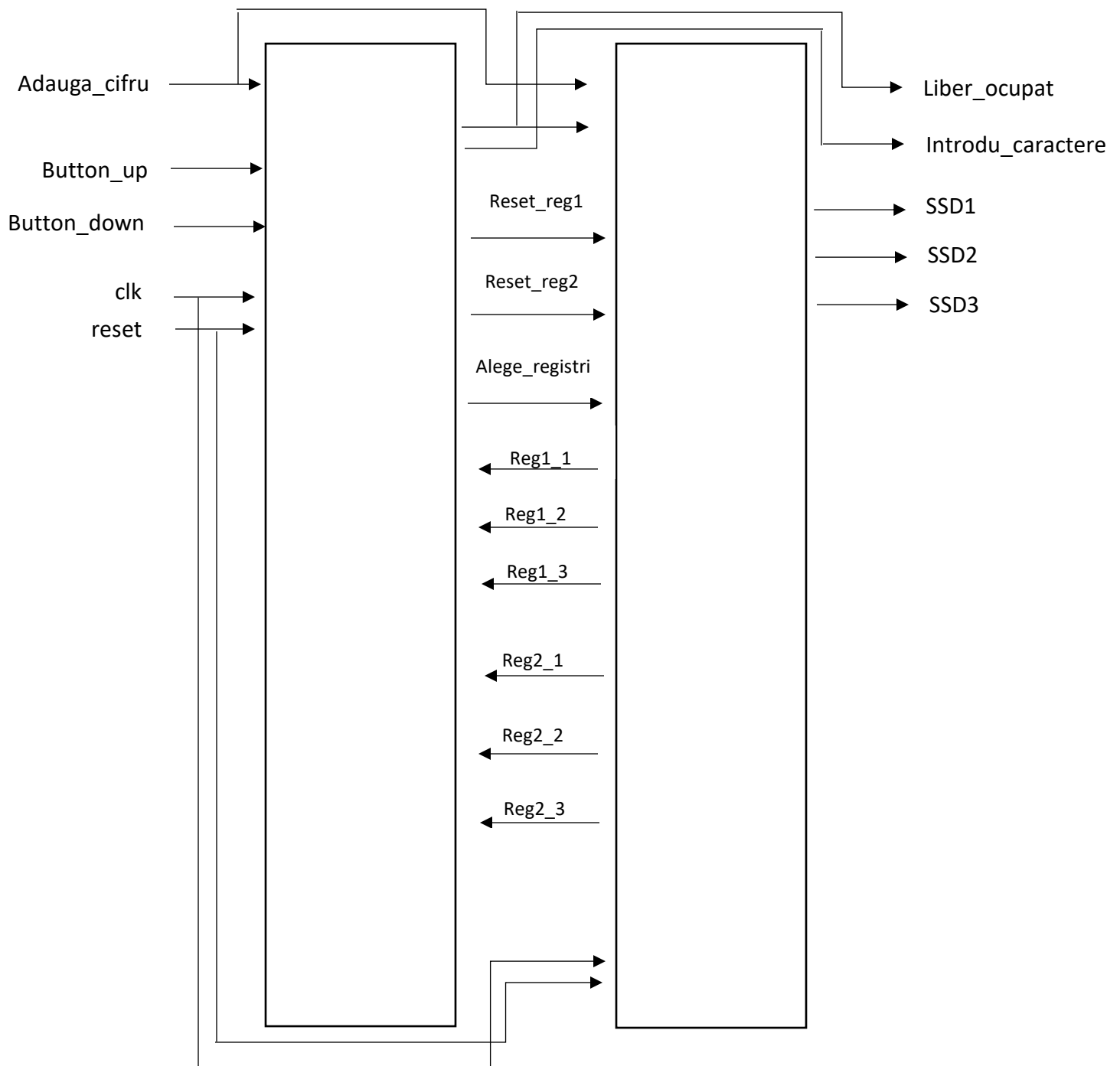
10.BCD decoder

Transforma informatia primita in hexazecimal in informatie ce poate fi afisata pe SSD.

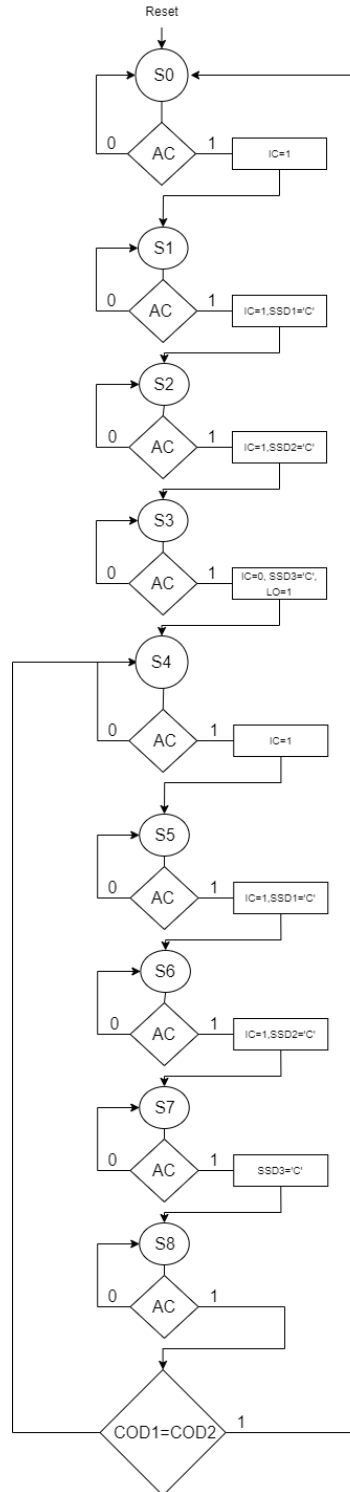


Intrarea hexa primeste numarul provenit din iesirea multiplexorului 2:1 iar iesirea decod trimite informatia modificata catre afisoarele SSD.

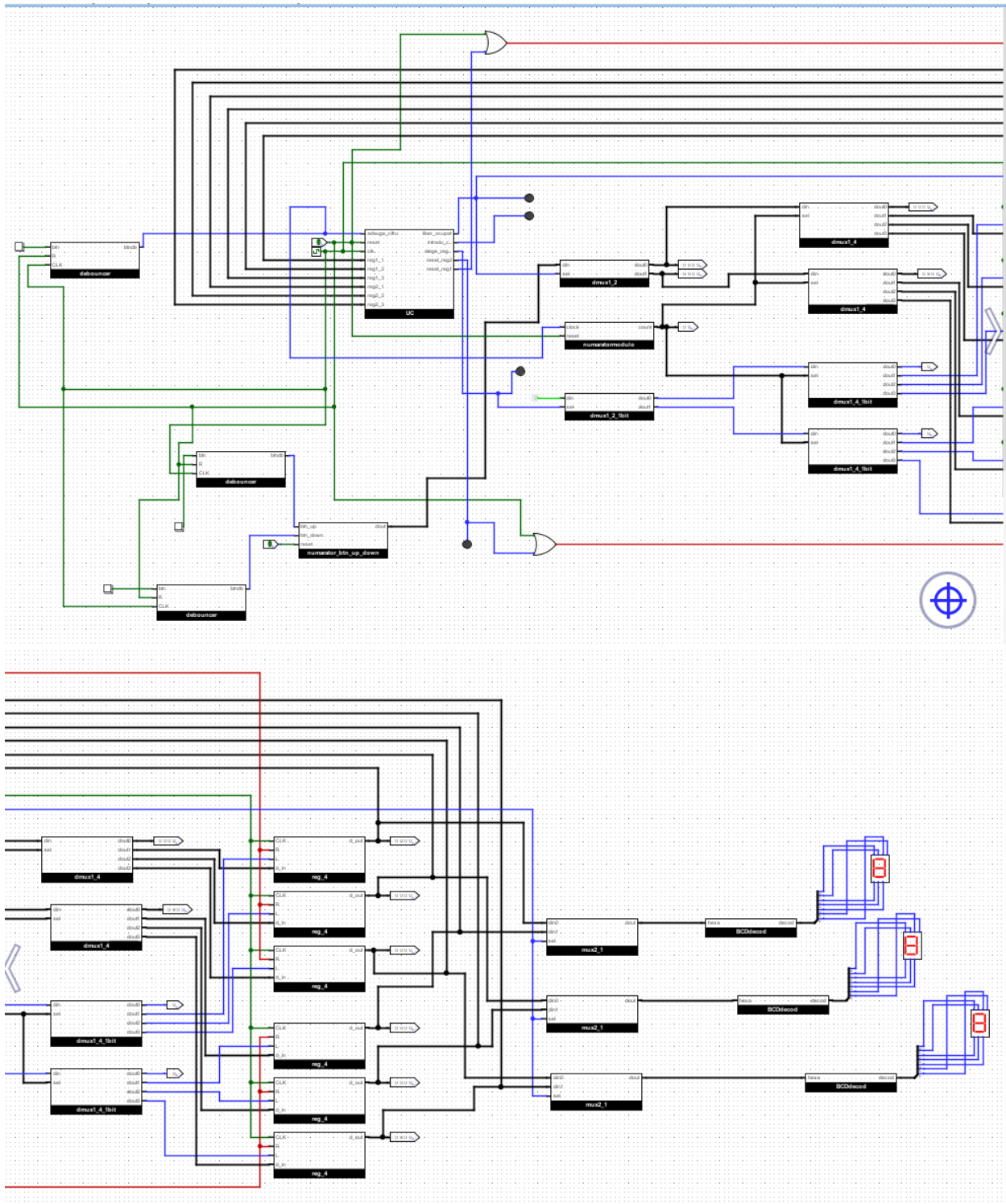
2.2.3 Schema bloc a primei descompunerii



2.2.4 Reprezentarea UC prin diagrama de stari (organigrama).



2.2.5 Schema de detaliu a proiectului



3. Manual de utilizare

Pentru a initializa introducerea cifrului se apasa butonul “Adauga cifru”. La prima apasare a acestuia se va aprinde ledul “Introdu caractere”, care semnalizeaza ca putem incepe sa introducem codul. Cu ajutorul butoanelor up si down se selecteaza un numar de la 0-9, A-F care va fi afisat, iar prin apasarea butonului “Adauga cifru” se memoreaza numarul introdus. Cand se ajunge la al treilea numar, prin apasarea butonului “Adauga cifru”, dulapul se blocheaza, ledul “Introdu caractere” se stinge iar ledul “Liber/ocupat” se va aprinde.

Pentru deblocare, se repeta pasii anteriori de introducere a codului. In cazul in care codul introdus este identic cu cel memorat anterior, dulapul se deblocheaza, iar ledul “Liber/ocupat” se stinge. In caz contrar, dulapul ramane blocat, iar ledul “Liber/ocupat” ramane aprins.

4. Justificarea solutiei alese

Am ales aceasta rezolvare deoarece o consideram logica si usor de inteles. Resursele folosite sunt cele studiate la laborator, lucru care a facilitat implementarea acestora in proiect. Astfel, informatia studiata in cadrul laboratoarelor a fost mai mult decat suficienta. Tranzitia starilor in UC se realizeaza fluent, elementele decizionale fiind putine si exacte.

5. Posibilitati de dezvoltari ulterioare

Registrii de memorare se pot inlocui cu o memorie RAM pentru a reduce numarul de resurse si costul de productie. In locul butoanelor up/down se poate folosi o tastatura pentru a usura introducerea cifrului.

6. Bibliografie

Indrumator laborator

Curs de Proiectarea Sistemelor Numerice

Cuprins

CUPRINS

1.Specificatii	2
2.Proiectare	4
2.1 Schema bloc	4
2.2 Unitatea de comanda si unitatea de executie	4
2.2.1 Maparea intrarilor si iesirilor cutiei mari pe cele doua componente UC si UE	4
2.2.2 Determinarea resurselor	5
2.2.3 Schema bloc a primei descompuneri	10
2.2.4 Reprezentarea UC prin diagrama de stari	11
2.2.5 Schema de detaliu a proiectului	12
3. Manual de utilizare	13
4. Justificarea solutiei alese	13
5. Posibilitati de dezvoltari ulterioare	13
6. Bibliografie	13
Cuprins	14