**Documentatie proiect lift P+12**

**Facultatea:Automatică și Calculatoare**

**Specializare:Calculatoare si tehnologia informatiei**

**Materia:Proiectarea sistemelor numerice**

**Elevi**: Pojar Andrei-Gabriel

Morar Iuliu-Mihai

**Îndrumator laborator**: Lisman Dragos-Florin

**Cuprins**

1.Specificația proiectului

2.Descrierea schemei proiectului

3.Listă de intrări și ieșiri

4.Listă de componente

5.Legarea componentelor

6.Simulare în Active-HDL

7.Justificarea soluției alese

8.Posibilități de dezvoltare

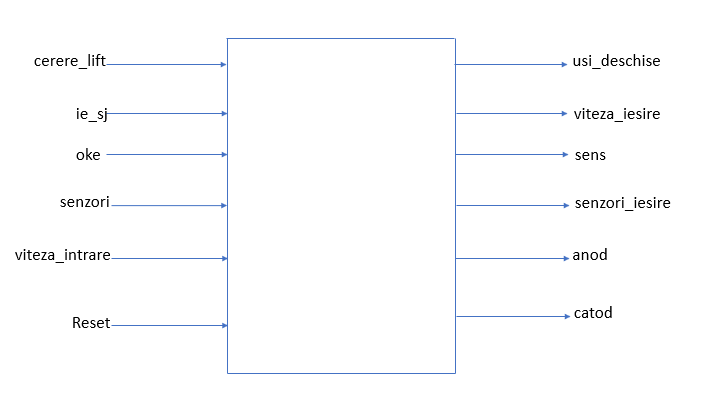
**1.Specificația proiectului**

Să se proiecteze un sistem care controlează un lift într-un hotel cu P+12 etaje. Liftul trebuie să îndeplinească cerințele oamenilor din interior și cererile exterior(sus,jos),care apar pe parcurs de la ușile aflate la fiecare nivel. Se onorează cererile în ordinea etajelor,indiferent de unde provin ele(lift sau exterior). Liftul are un orificiu de intrare care detectează depășirea greutații maxime admise și nu pornește în acest caz. Plecarea nu are loc dacă ușile nu sunt închise. Ușile trebuie să stea deschise un interval de timp programabil. Ușile nu se inchid dacă există o persoană în ușă. Viteza liftului va fi selectabilă între două valori: 1 sau 3 secunde/etaj. Se consideră că la inceput liftul se gasește la parter,cu ușile deschise.

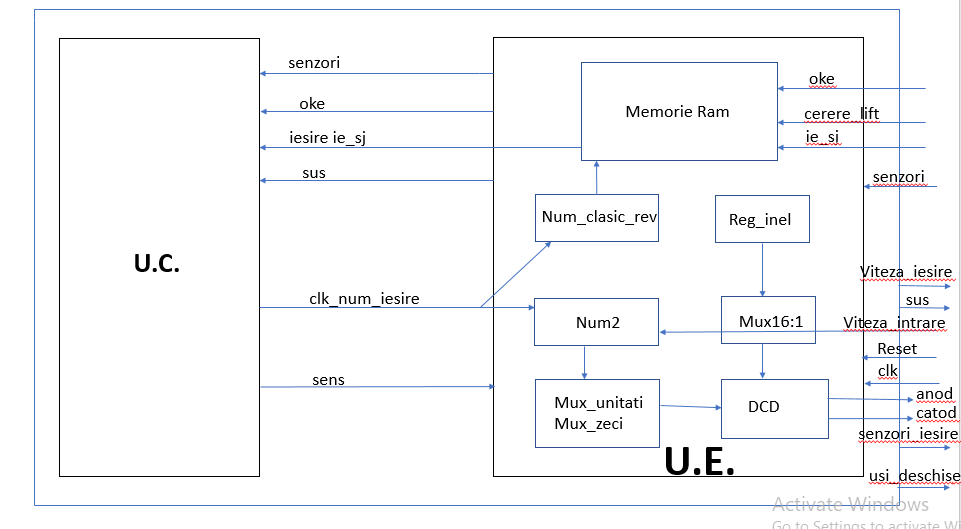
**2.Descrierea schemei proiectului**

**1)Schemă bloc**

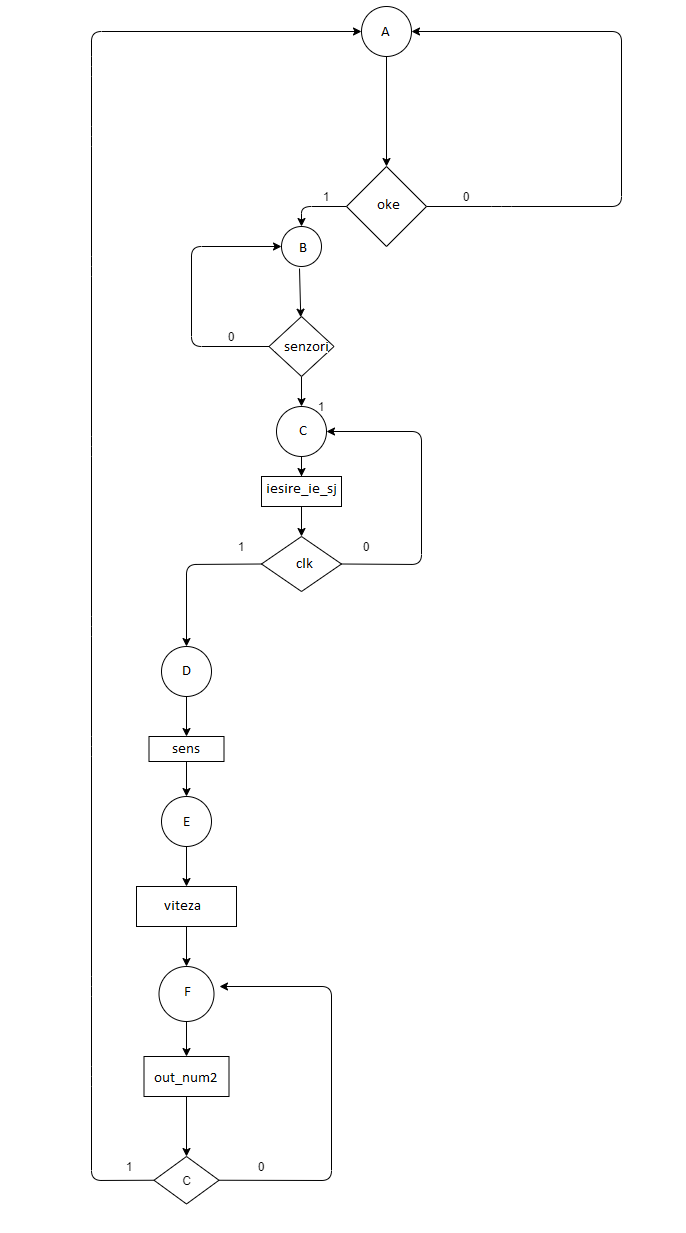
Automatul va fi prezentat sub forma unei cutii negre.



**2)Schema(UC+UE)**



**3)Organigrama unității de control**

****

**3.Listă de intrări și ieșiri**

**1)Intrări:**

-**cerere\_lift**=intrare pe 4 biți-switch-uri

Utilizatorul va introduce etajul la care vrea să ajungă,adică cerere din interior sau etajul la care utilizatorul se află în momentul în care cheamă liftul,adică cerere din exterior.

-**ie\_sj**=intrare pe 4 biți-switch-uri

Cu ajutorul acestei intrări se vor introduce date cu privire la poziția utilizatorului(interior sau exterior) si la sensul dorit de acesta în cererea lui din exterior(up sau down).

-**oke=**intrare pe un bit-buton

Cu ajutorul acestei intrări se va salva etajul și va arăta că utilizatorul a terminat de introdus cererea sa(exterior sau interior).

-**senzori=**intrare pe 2 biti-2 butoane

Senzorii se refera la senzorii de greutata maxima si de persoana in usa.

**-viteză\_intrare=**intrare pe 1 bit-switch

Se va alege viteza de deplasare a liftului (valoarea 0 pentru 1 secundă/etaj, respectiv valoarea 1 pentru 3 secunde/etaj)

**-Reset**=intrare pe 1bit-buton

Va duce liftul in starea inițială,la parter cu ușile deschise.

**2)Ieșiri**

**-usi\_deschise**-led

Se aprinde atunci când ușile sunt deschise.

-**viteza\_ieșire**-led

Se aprinde atunci când viteza este 3s/etaj și se stinge când e 1s/etaj.

**-sens**-led

Se aprinde atunci când liftul urcă(sens=1).

**-senzori\_ieșire**-led

Se aprinde atunci când greutatea maximă admisă este depășită sau există o persoană în ușa liftului.

**-anod și catod-**afișor

Se folosesc pentru a afișa etajul curent la care se află liftul.

**4.Listă de componente**

**-Memoria Ram**

Memoria RAM are rolul de a stoca cererile introduse de utilizator.

Se stochează în memorie datele din ie\_sj.

Are drept intrări două seturi de adrese:

1)cererea care vine de la utilizator pe 4 biți-se scrie în memorie(ie\_sj)

2)ieșirea de la numărătorul clasic(4 biți)-se citește din memorie

Memoria RAM mai are ca intrări pe oke cu rol de we.Are ca ieșire pe ie\_sj\_ieșire care are datele citite de la adresa dată de ieșirea numărătorului clasic,iar ieșirea este trimisă in UC.

**-Numărător clasic etaj sens**

Este un numărător reversibil și are rolul de a parcurge toate etajele liftului până în momentul în care primește un semnal că s-a găsit un etaj la care liftul trebuie să oprească(etaj curent=etaj destinație). Când acesta are valoarea “0000” sensul este de urcare(sens=1),iar când ajunge la valoarea “1100” sensul este de coborâre(sens=0). Ieșirea acestui numărător va merge la memoria RAM pentru a verifica dacă există o cerere la acea adresă.

**-Numărătorul numărul 2**

Este un numărător reversibil și va număra în momentul în care se găsește un etaj la care liftul trebuie să ajungă. Fiecare valoare a numărătorului va fi afișată pe afișorul BCD-7 segmente.

**-Unitatea de decizie**

i=cerere din interior

e=cerere din exterior

s=urcare

j=coborâre

sens=sensul curent al liftului

Ieșirea unității de decizie va avea valoarea 1 în următoarele cazuri:

clk\_num reprezintă ieșirea unității.

i=0,e=0,s=0,j=0,sens=0 =>clk\_num=1

i=1,e=0,s=0,j=0,sens=1=>clk\_num=1

i=0,e=1,s=1,j=0,sens=0=>clk\_num=1

i=0,e=1,s=0,j=1,sens=0=>clk\_num=1

În toate celelalte cazuri,iesirea clk\_num este 0.

**-Registru inel(de deplasare circular)**

Registrul inel este folosit pentru afișorul pe 7 segmente. Are ca intrări un clk,un reset,iar ca ieșire un semnal pe 4 biți. Dupa primul semnal de clock registrul inel va avea valorea “1110”,iar apoi va devein “1101”,”1011” etc. Prin această deplasare continuă la stânga a bitului 0,poziția zeroului din semnalul de ieșire va indica anodul care se va activa.

**-Divizor de frecvență**

Practic,acest divizor de frecvență este un numărător pe 26 de biți.Rolul lui este de a încetini pe placa timpul de execuție pentru a putea urmări datele afișate pe afișorul cu 7 segmente.

**-Decodificator**

Rolul decodificatorului este de a decodifica informația provenită de la multiplexor. El are ca și intrare ieșirea multiplexorului și ca ieșire un semnal catod pe 7 biți.

**-Multiplexor**

Multiplexorul 16 la 1 va avea ca și selecție ieșirea registrului inel și va avea 16 intrări dintre care se vor folosi urmatoarele:

-7=0111

-11=1011

-13=1101

-14=1110

Fiecare din aceste intrări va corespunde unui afișor.

**-Numărător pentru viteză**

Pentru acesta vom folosi și o componentă Mux 2 la 1 și o poartă and. Acest numărător de viteză numără în intervalul 0-1 sau 3. Când acest numărător își atinge limita maximă el va activa un semnal care va intra într-un Mux 2 la 1 ca intrare. Acest Mux va folosi ca selectie pe viteza\_intrare(care e o intrare a liftului) și astfel vor iesi impulsuri de clock în funcție de viteza selectată(v\_selectata).

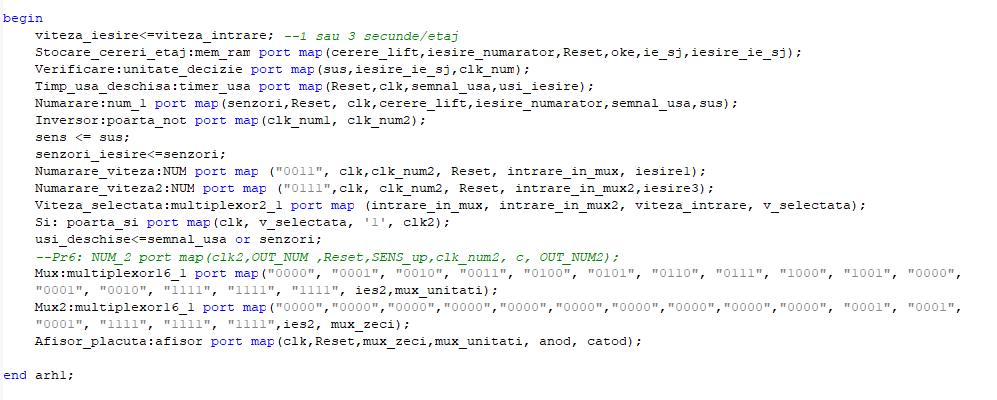
**-Cronometru(timer) pentru ușă**

Aceasta componentă va avea ca intrare un semnal de clock,un reset și un semnal\_ușă. Ca ieșire va avea uși\_ieșire care va fi folosită la ieșirea liftului uși\_deschise.

**-Afișor BCD 7 segmente**

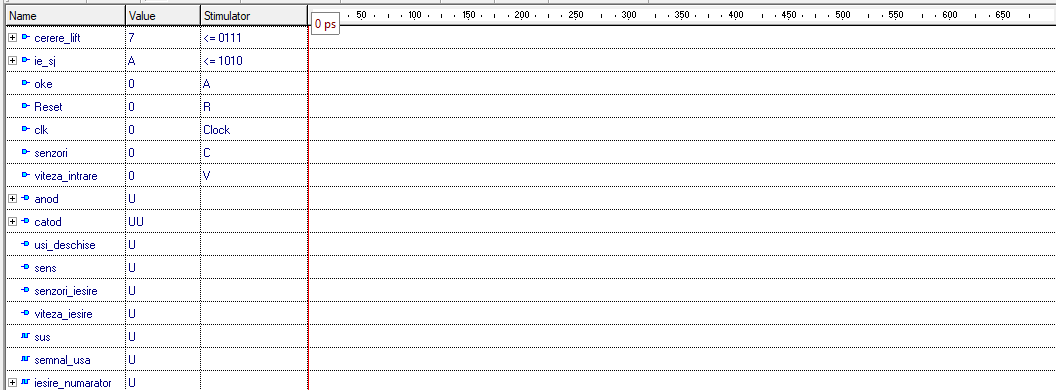
Pe acest afișor vom activa pe rând unul câte unul,fiecare anod,iar pe restul îi dezactivăm. Vom avea nevoie pentru a implementa afișorul de divizorul de frecvență,de decodificator,respectiv de multiplexorul 16 la 1 și de registrul inel,a cărui ieșire va fi selecția multiplexorului.

**5.Legarea componentelor**

****

**6.Simulare în Active-HDL**

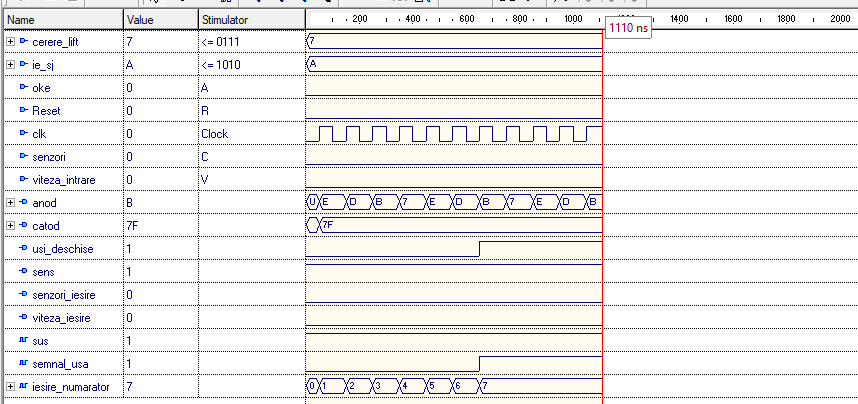
După ce deschidem proiectul în Active-HDL va trebui să compilăm tot(Compile All),iar apoi să simulăm. După ce am compilat și ne-am asigurat că totul este în regulă,vom inițializa o simulare(Initialize Simulation) si vom crea un nou waveform. Aici,vom adăga semnalele necesare simulării și vizionării rezultatului. Se vor da valori semnalelor utilizând opțiunea Stimulators.Apoi vom alege ce stimulator se potriveste intrării respective (clock,hotkey,force value etc).

****

După ce am urmat acești pași,vom da un Run For și vom urmări cum se modifică rezultatele.

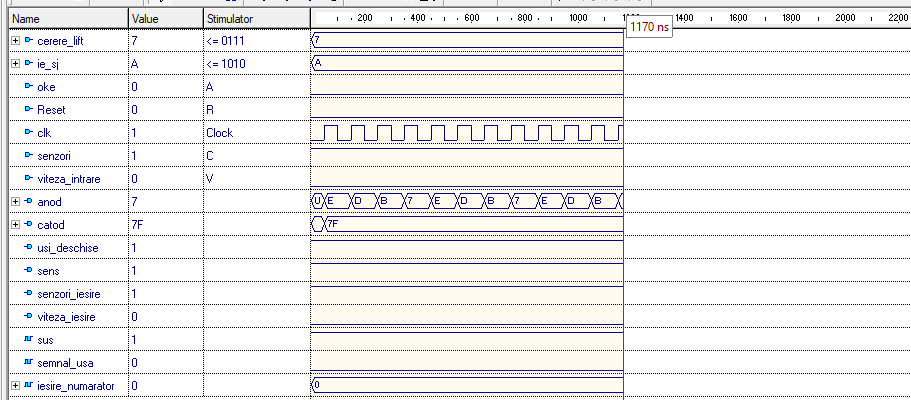
Vom da câteva exemple de teste:

1. Când avem o cerere de lift la un anumit etaj,iar senzorii de greutate maximă și de persoană în ușă sunt pe 0,reset pe 0 si viteza de intrare pe 0.



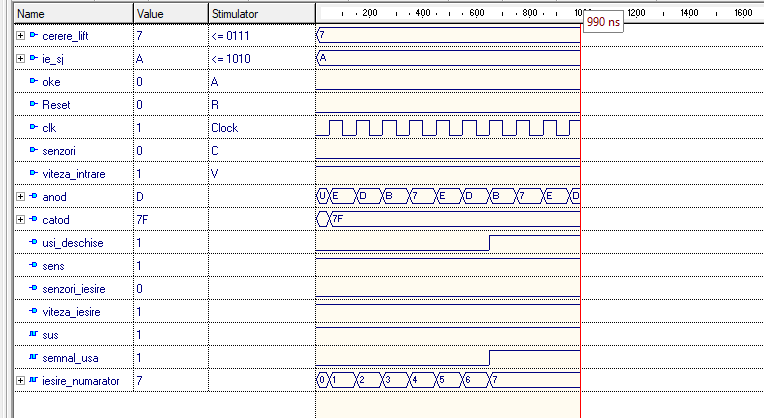
De exemplu daca cerere\_lift=7,atunci usi\_deschise va fi 0 pe tot parcursul urcarii,iar când ajunge la etaju 7 se vor deschide adică va fi 1. Apoi sensul este 1,adică de urcare. Iesire\_numărător va număra până la etajul dat,iar când ajunge la acel etaj se va opri din numărat.

2)Când avem o cerere de lift la un anumit etaj,iar senzorii de greutate maximă și de persoană în ușă sunt pe 1,reset pe 0 si viteza de intrare pe 0.



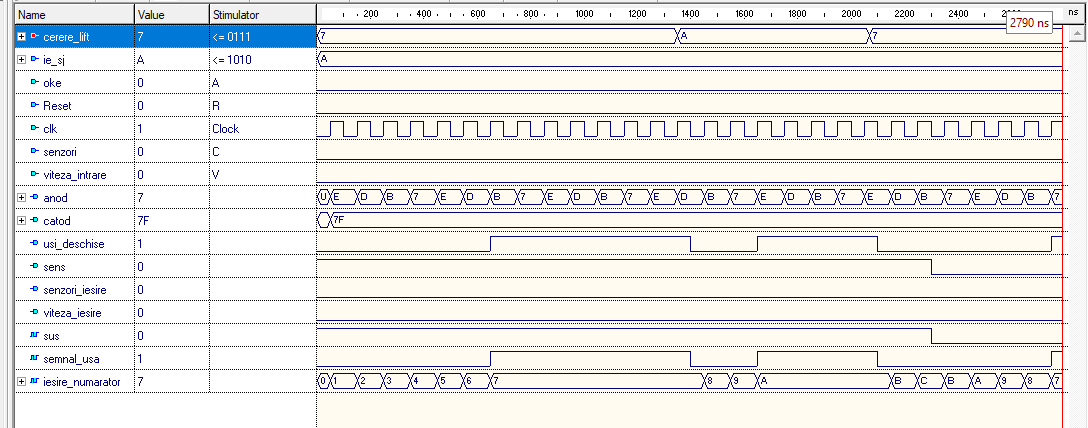
Observăm că dacă senzorii de greutate maximă și de persoană în ușă sunt pe 1,adică activi atunci indifferent cât este cerere\_lift ,liftul nu va porni ,adica numărătorul nu va număra,iar ușile vor rămâne deschise.

3)Când avem o cerere de lift la un anumit etaj, iar senzorii de greutate maximă și de persoană în ușă sunt pe 0,reset pe 0 si viteza de intrare pe 1.



Observăm că viteză\_ieșire va fi 1 cât timp viteză\_intrare e 1.Restul functioneaza la fel(numărătorul numără până la etajul la care are de mers etc.)

4) Când avem o cerere de lift la un anumit etaj,iar liftul trebuie(exemplu:suntem la etajul 10 și avem cerere la 7), iar senzorii de greutate maximă și de persoană în ușă sunt pe 0,reset pe 0 si viteza de intrare pe 1.



Observăm că atunci când coborâm sens se face 0(adica liftul coboară).

**7.Justificarea soluției alese**

Pentru implementarea liftului și pentru realizarea proiectului, am utilizat cea mai utilizabilă metodă ,adică împărțirea pe componente.

Am folosit variabile cu nume suggestive pentru a nu se crea confuzii.

Apoi am ales să simulez în Active-HDL deoarece este mai ușor de înțeles pentru toți cunoscătorii de VHDL.

Resursele software neceseare pentru realizarea acestui proiect sunt minime. Este nevoie de un PC pe care Active-HDL să ruleze fără probleme.

**8.Posibilități de dezvoltare**

O posibilă dezvoltare ar putea fi reprezentată de mărirea numărului de etaje ca să fie compatibil și cu clădirile mai înalte.

O altă posibilă dezvoltare ar putea fi reprezentată de introducerea unei comenzi vocale care să te atenționeze vocal când ai ajuns la etajul dorit,când ai depășit greutatea maximă admisă sau când este persoană în ușă și multe alte oportunități care se pot face cu o comandă vocală.