

## Structura For Loop

- **Structura For Loop**

Se gasesc in grupul Programming-->Structures-->For Loop

Aplicatiile in care la un moment dat trebuie sa repetam anumite parti ale aplicatiilor de un anumit numar de ori, sunt realizate utilizand structura "For Loop"

- **Utilizarea structurii For Loop**

Structuria "For Loop" este similara instructiunii "for" din C++. Este folosita pentru a repeta o secventa de n ori unde n este precizat inainte de inceperea repetitiei.

Cea mai populara aplicatie utilizand instructiunea for este afisarea a n naturale la un anumit interval de timp for v0 v0. Vom relaiza deci in LabVIEW, utilizand structura For Loop o astfel de aplicatie. Pentru a se putea urmari afisarea celor n de numere va trebui introdusa o temporizare de aproximativ 300 ms.

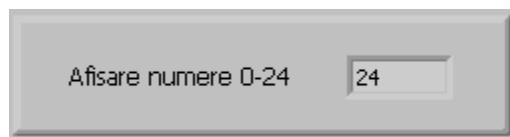
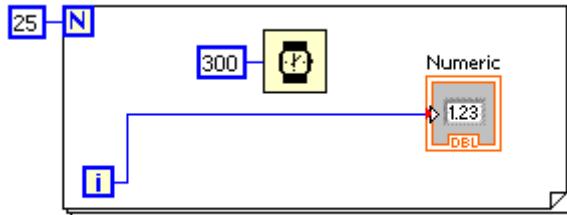


Diagrama bloc va contine structura "For Loop" in interiorul careia se include afisarea numerelor si temporizarea afisarii, este prezentata in figura de jos:



Aplicatia poate fi imbunatatita for v0 v1 in sensul ca vom pune un control pentru a stabili intarzirea dintre doua afisari, iar afisarea se va face si analogic folosind un control de tip meter.

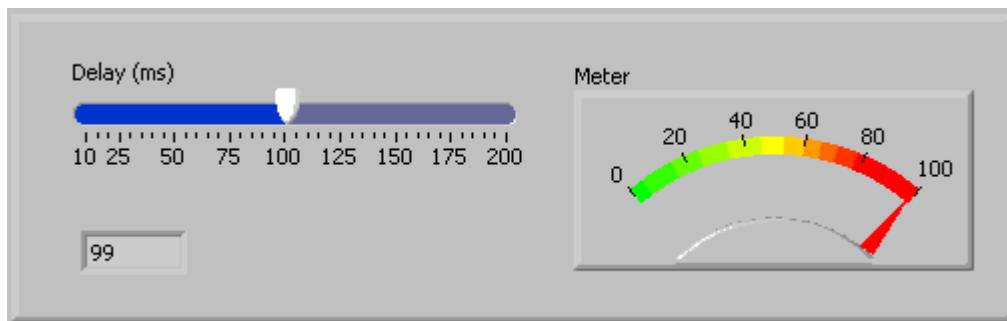
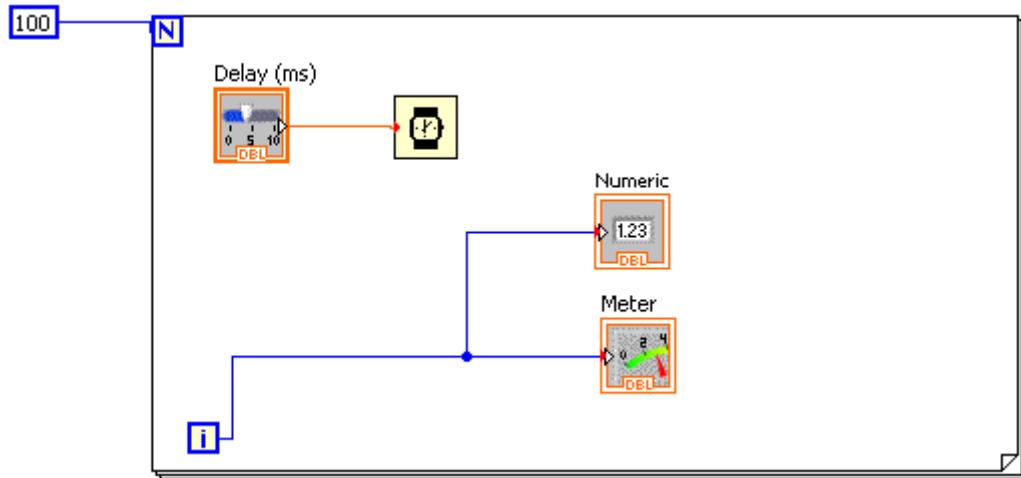


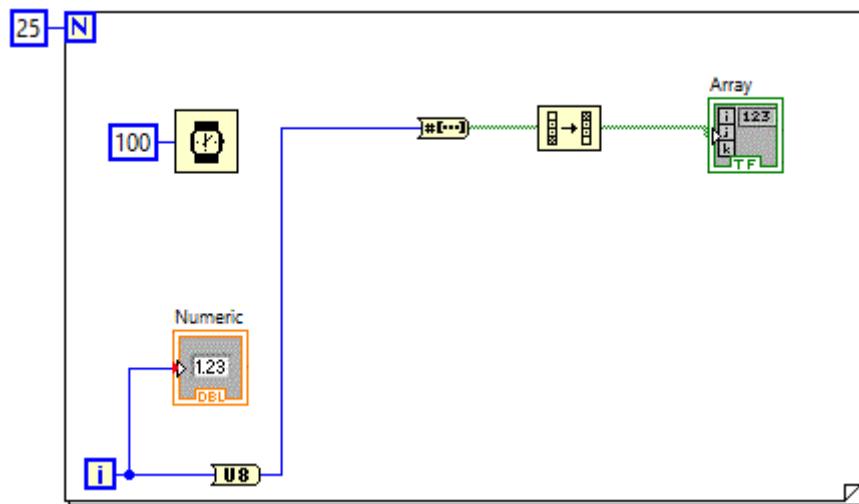
Diagrama bloc va contine structura "For Loop" in interiorul careia se include afisarea numerelor in cele doua controale cat si temporizarea stabilita de la un control separat.



Vom relua aplicatiile cu led-uri for v0 v2 si vom afisa repetitiv n numere atat in zecimal cat si in binar utilizand led-uri.



Diagrama bloc va trebui deci sa contina o structura "For Loop" inclusa intr-o structura "While Loop"



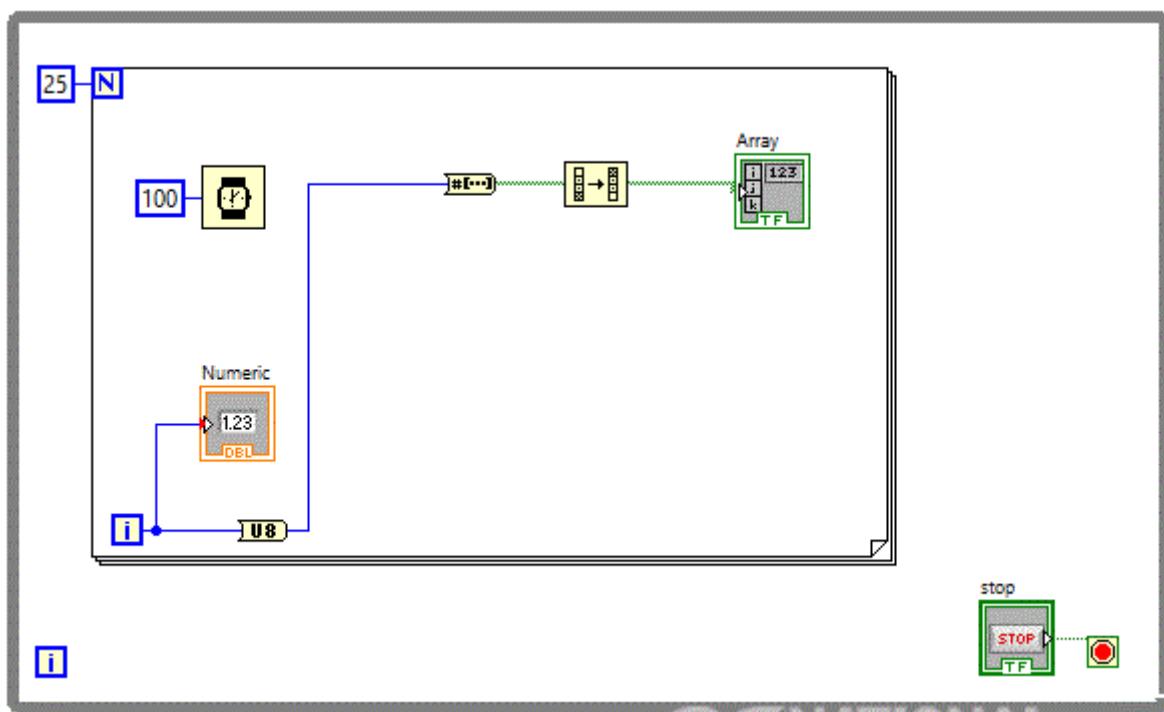
- **Structura For Loop impreuna cu structura While Loop**

Aplicatia anterioara se opreste dupa efectuarea celor 25 de pasi precizati in structura "For Loop"

Pentru a rula in mod continuu, in aplicatia anterioara trebuie sa mai plasam o structura de programare de tipul "While Loop" obtinand astfel aplicatia [for v0 v3](#).



In vederea rularii repetate, pana la apasarea butonului "Stop" se va utiliza deci inca o structura repetitiva de tipul "While Loop".



Aplicatia anterioara ruleaza repetitiv pana la apasarea butonului "Stop" dar nu mai repede de terminarea cyclului "For" in curs de executie. Cu alte cuvinte efectul comenzi "Stop" nu este imediat. Pentru a se observa mai bine acest fenomen, s-a realizat aplicatia [for v0 v4](#) in care se poate stabili numarul de iteratii.

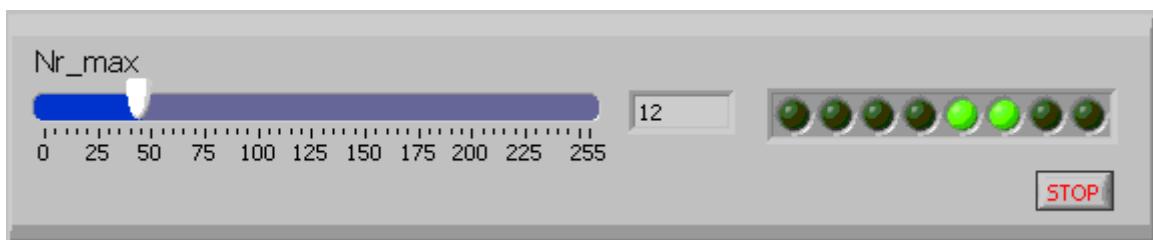
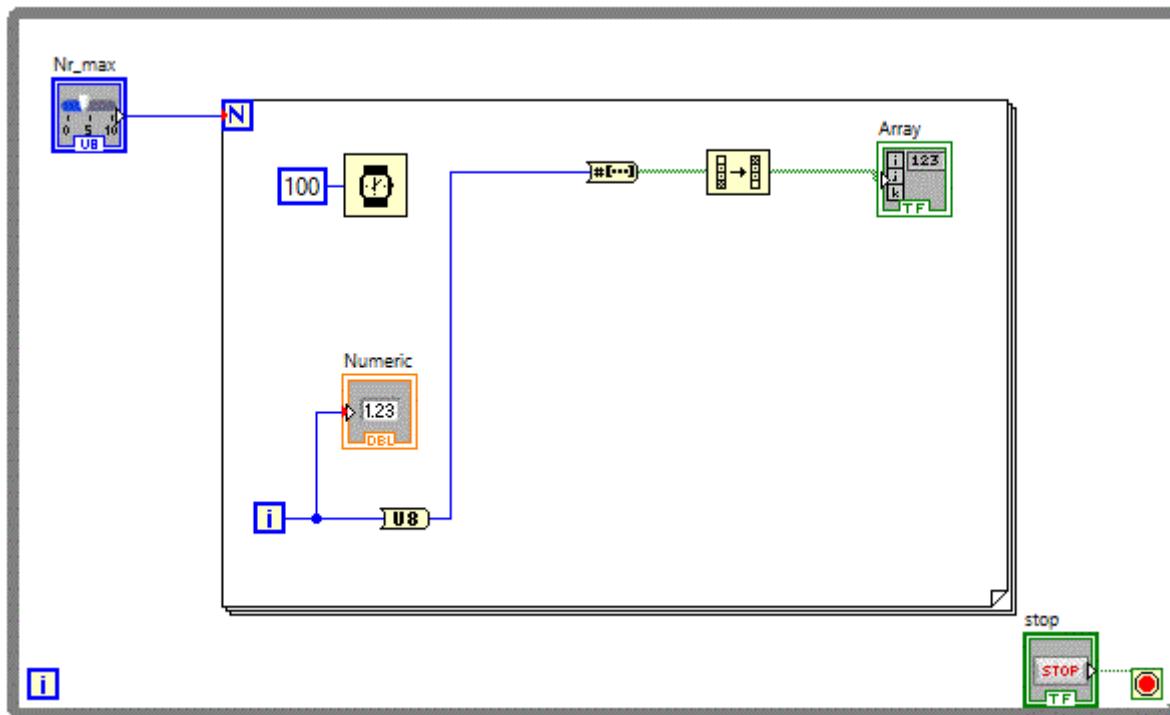
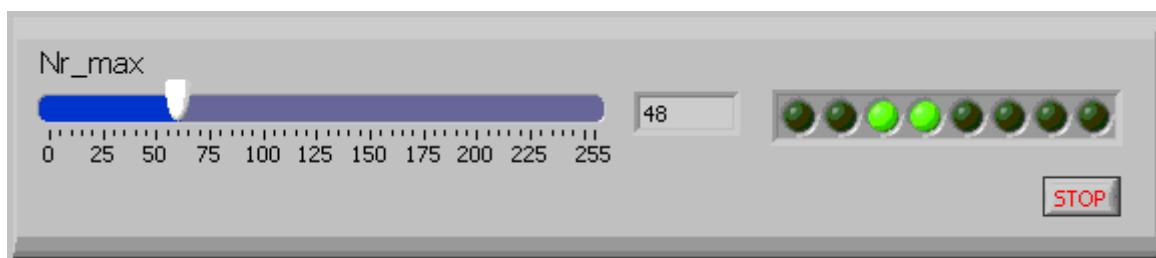


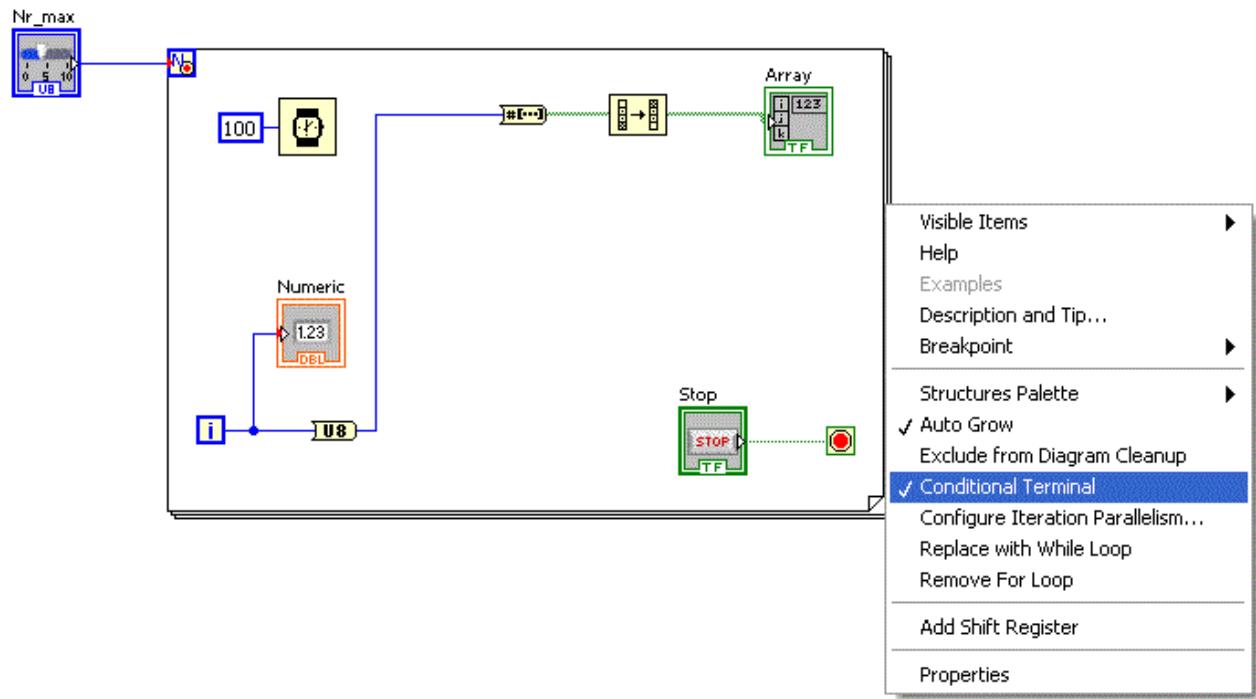
Diagrama bloc va contine legatura de la controlul care furnizeaza numarul de iteratii spre structura "For Loop".



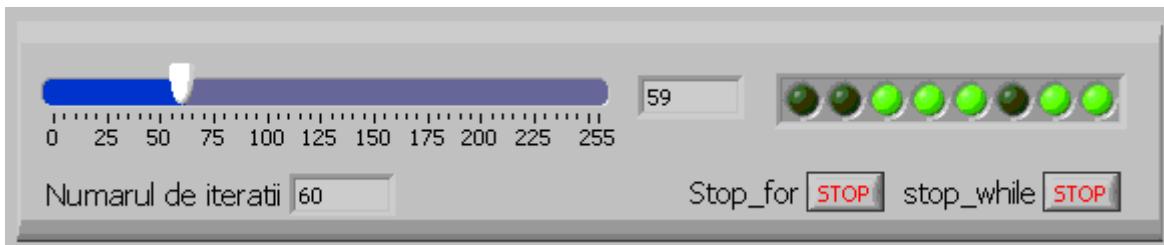
Pentru a corecta neajunsurile amintite mai sus, am putea folosi in urmatoarea aplicatie, [for v0 v5](#) o structura "For Loop" in care este validata optiunea "Conditional Terminal".



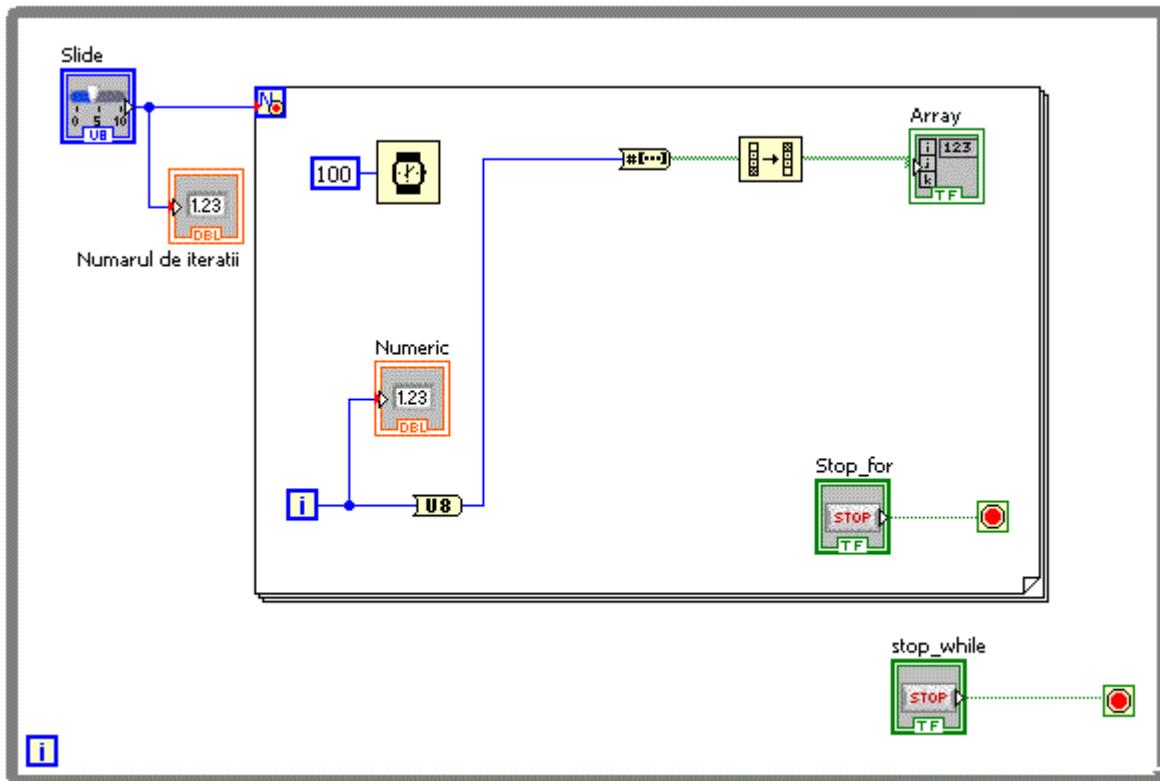
Validarea optiunii "Conditional Terminal" se face folosind click-dreapta pe structura "While Loop".



Dezavantajul folosirii acestei metode constă în faptul că la apasarea butonului "Stop" se ieșe din aplicatie, deci nu avem posibilitatea să reluam rularea aplicatiei decat după relansarea ei. Trebuie să folosim deci în continuare și structura "While Loop" la fel ca în aplicatia [for v0 v6](#):

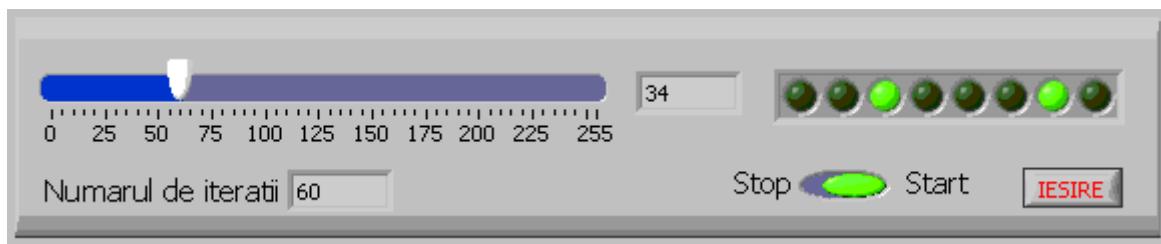


Avem deci două butoane, unul pentru a stopa structura "For Loop" și unul pentru a stopa structura "While Loop".

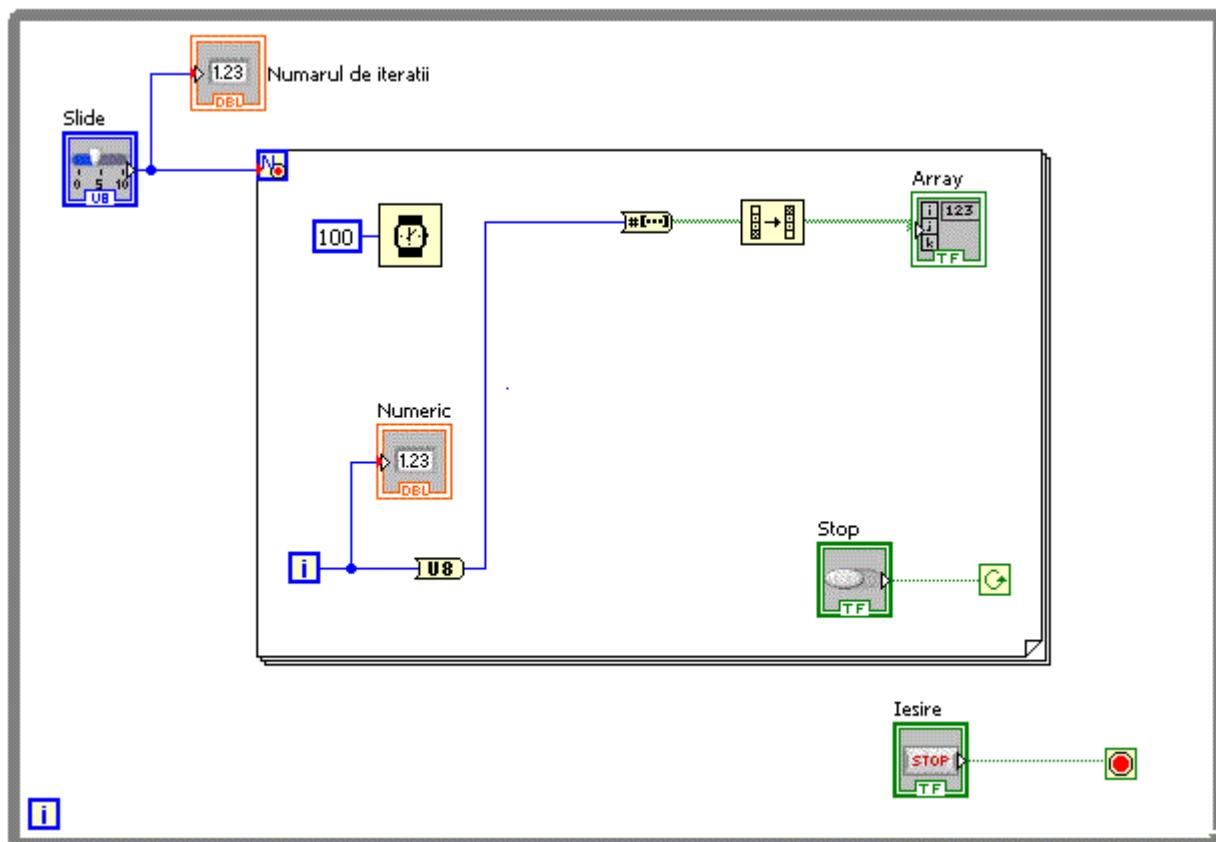


Dupa apasarea butonului "Stop\_for" aplicatia se reinitializeaza dar porneste imediat un nou ciclu "For Loop".

Daca dorim sa se opreasca pana la o noua actionare a butonului "Stop\_for" va trebui sa modificam comportamentul acestui buton si sa-l transformam intr-un comutator [for\\_v0\\_v7](#).



Butonul "Stop\_for" se transforma intr-un comutator "Start-Stop" iar butonul "Stop" are acum semnificatia de "Iesire" din aplicatie.



Toate aplicatiile pe care le vom realiza in viitor vor fi cuprinse intr-o structura "While Loop" si vor avea cele doua butoane "Start-Stop" si "Iesire"

Bazandu-ne pe aceste considerente, aplicatia pentru afisarea a n numere devine: [for v0\\_v8](#)

In aceasta aplicatie avem posibilitatea opririi si pornirii acesteia, iar in cazul in care se doreste iesirea din aplicatie, in cazul in care aplicatia este oprită, iesirea din aplicatie se face imediat, ne mai fiind necesara asteptarea terminarii repetitiilor din cadrul structurii "For Loop"

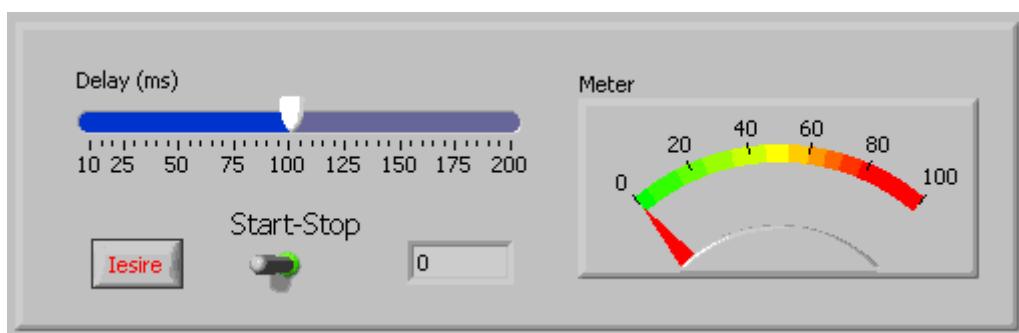
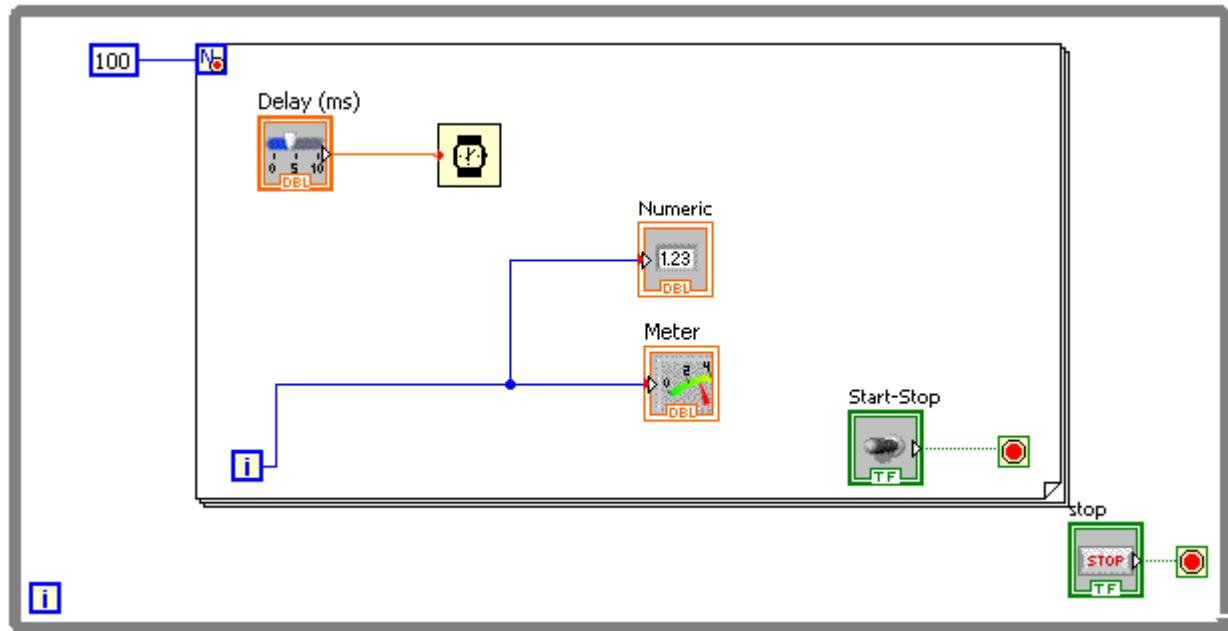


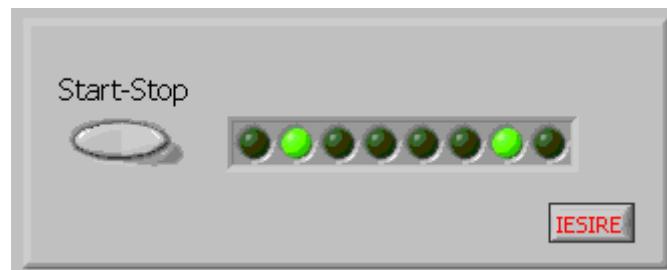
Diagrama bloc fiind:



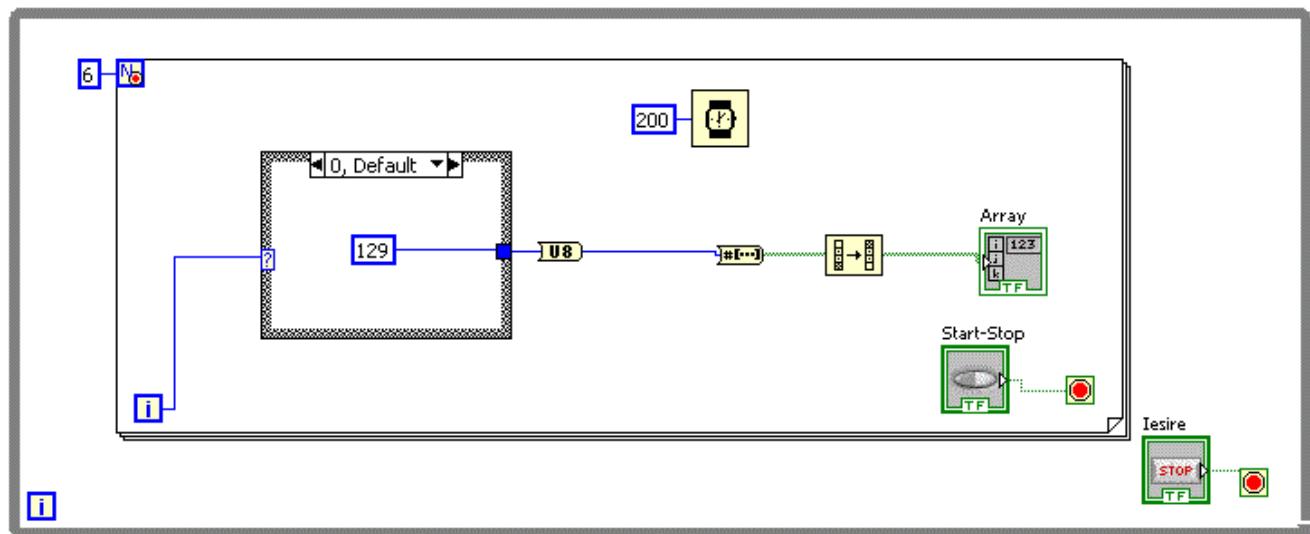
- Utilizarea structurii For Loop impreuna cu structurile decizionale

Structuria "For Loop" poate contine la randul ei diverse structuri de programare printre care si structurile de tip decizional.

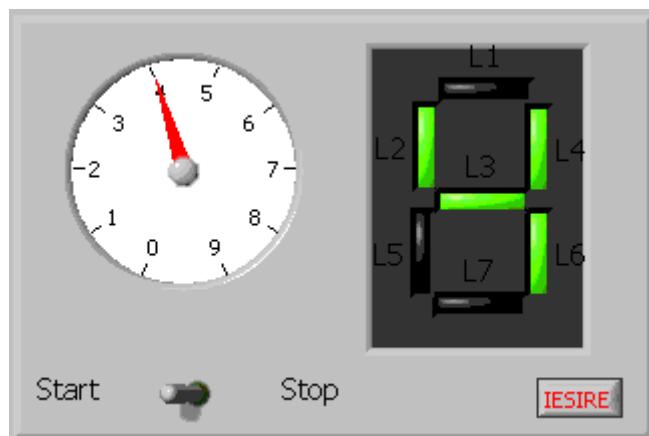
Vom relua aplicatia in care s-a realizat un joc de lumini pe led-uri dar de data aceasta seventele vor fi afisate automat fata sa apasam diverse controale [for v0 v10](#).



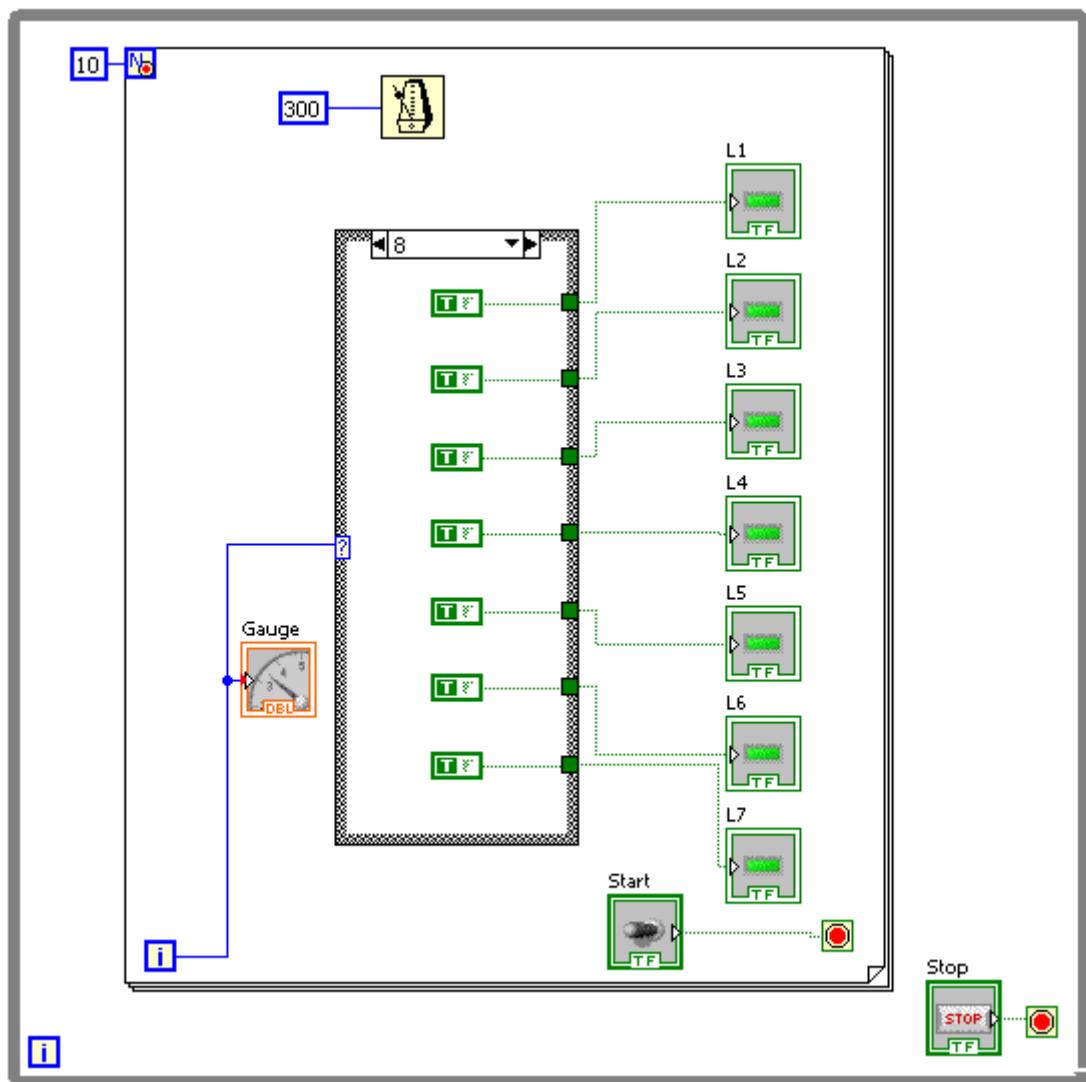
In diagrama bloc, se observa includerea vechii aplicatii intr-o structura "For Loop".



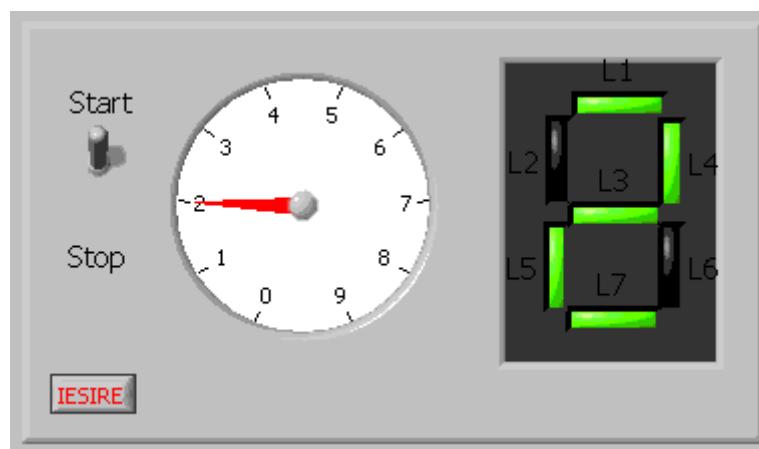
Vom simula acum un afisor pe 7 segmente realizat din led-uri [for\\_v0\\_v11](#)



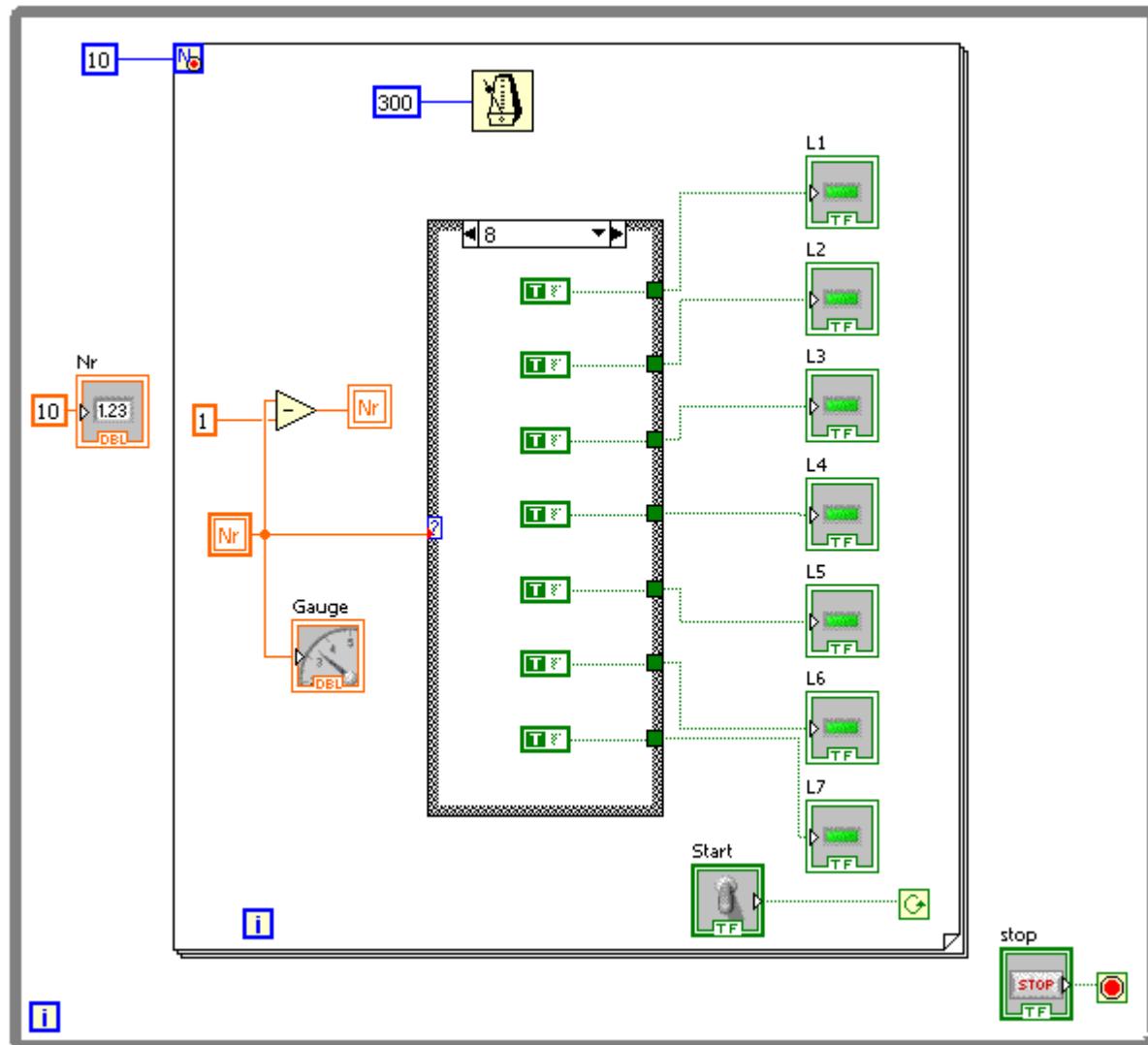
Pentru a comanda corespunzator cele 7 led-uri vom folosi structruri "Case"



Daca dorim sa realizam un numarator cu numarare inversa (de la 9 la 0) for v0 v12 de forma:



Nu vom mai putea folosi variabila *i*, va trebui sa introducem o variabila locala.

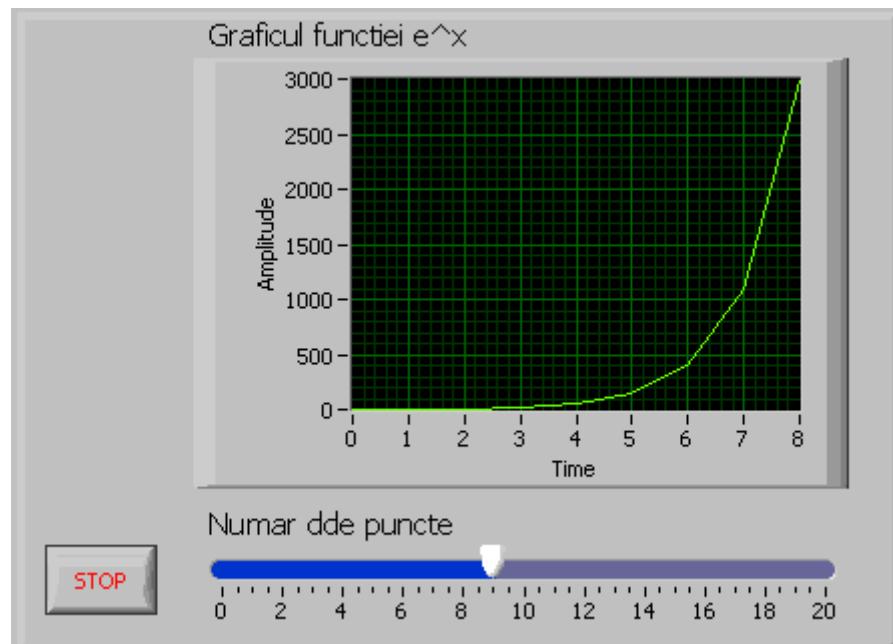


Variabila locala este "Nr", variabila initializata cu 10 la intrarea in structura "For Loop" si este decrementata dupa fiecare iteratie in cadrul structurii "For Loop".

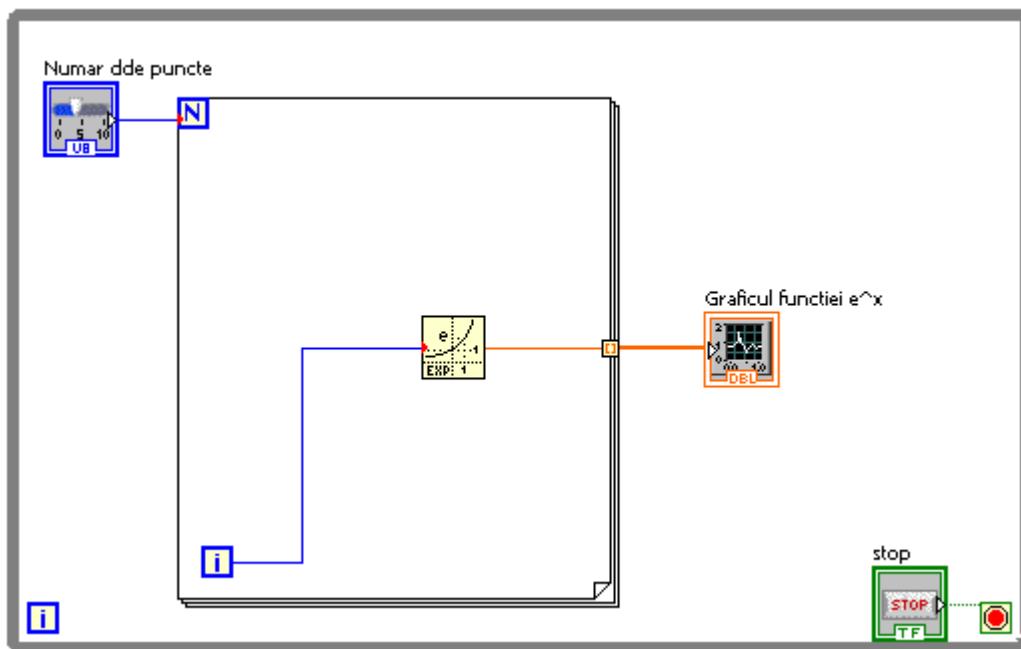
- **Utilizarea structurii For Loop pentru realizarea si afisarea graficelor**

Pe baza structurilor repetitive vom realiza aplicatii pentru trasarea graficului diverselor functii matematice.

Pentru inceput sa trasam graficul functiei  $e^x$ . Sa realizam aplicatia [for\\_v0\\_v00](#)



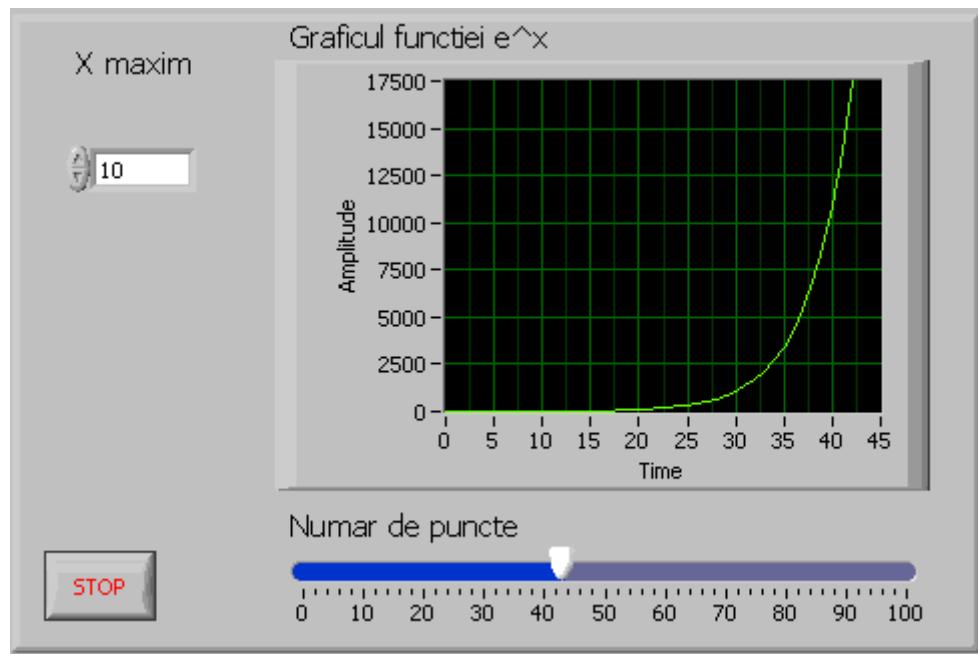
Vom folosi o structura "For Loop" si vom utiliza variabila i care reprezinta iteratia curenta in structura repetitiva, pe post de variabila x.



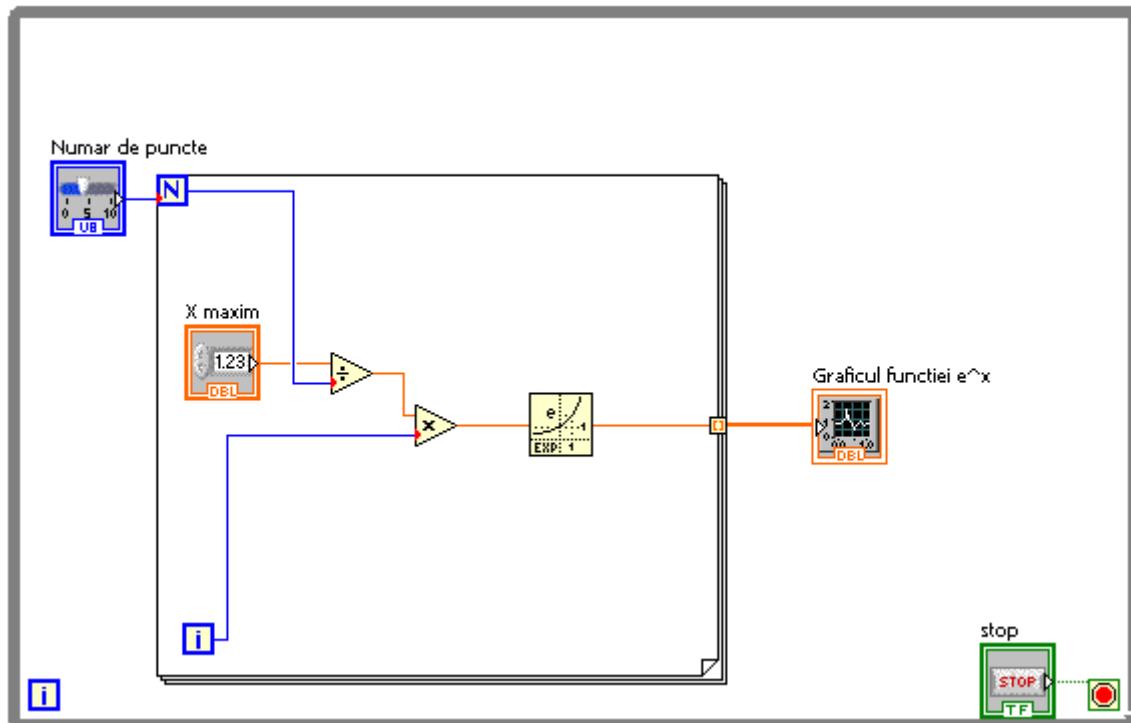
Daca dorim ca afisarea sa se faca dupa ce s-au terminat repetitiile in structura "For Loop" vom plasa un control de tip Modern-->Graph-->Waveform Graph. Se observa ca legatura intre structura "For Loop" si Waveform Graph este de tip 1D Array.

Dezavantajul folosirii variabilei i pe post de variabila x are dezavantajul ca x nu ia decat valori de tip "Integer". Pentru a trasa graficul functiei  $e^x$  variabila x trebuie sa fie de tip real. Va reiebui atunci sa precizam domeniul valorilor de intrare si numarul de pasi. Numarul de

pasi il putem seta setand N, dar pentru domeniu va trebui sa introducem un control suplimentar. [for v0 v21](#)



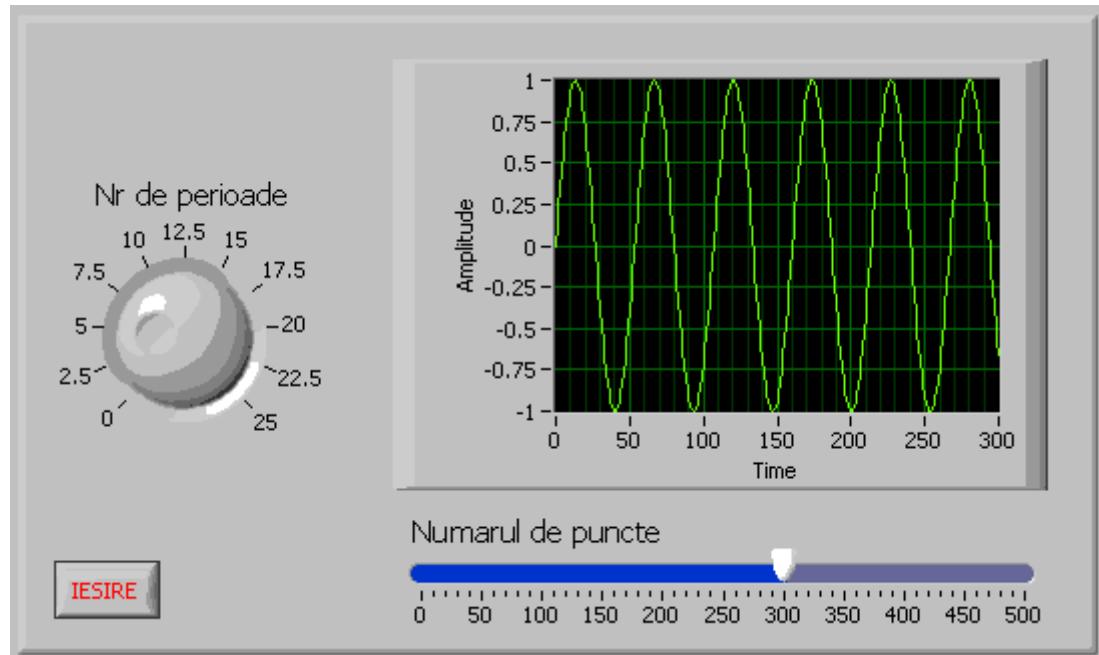
Valoare furnizata de controlul "X maxim" va fi utilizata pentru calculul latimii unui pas in functie de numarul de pasi.



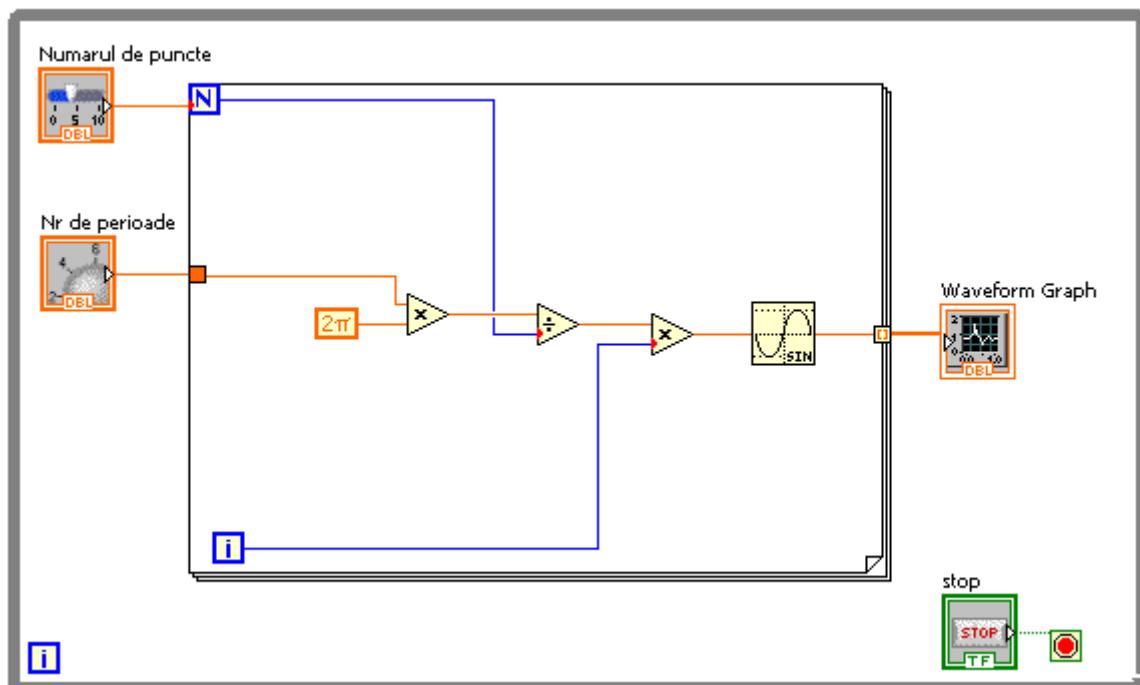
Latimea unui pas este deci :  $X \text{ maxim}/\text{Numar de puncte}$ , iar valoarea curenta pentru care se calculeaza valoarea functiei este  $i * X \text{ maxim}/\text{Numar de puncte}$ , dupa cum se poate vedea in

diagrama logica de sus.

Sa realizam acum graficul functiei "sin(x)". Dorim sa afisam dinamic graficul acestei functii, in sensul ca vrem sa modificam perioada, respectiv frecventa acestei functii, si afisarea sa tina cont de aceste modificarri si sa reafiseze corespunzator graficul acestieei functii. Ne propunem deci mai degraba sa simulam un generator de semnal sinusoidal **for v0 v22**.

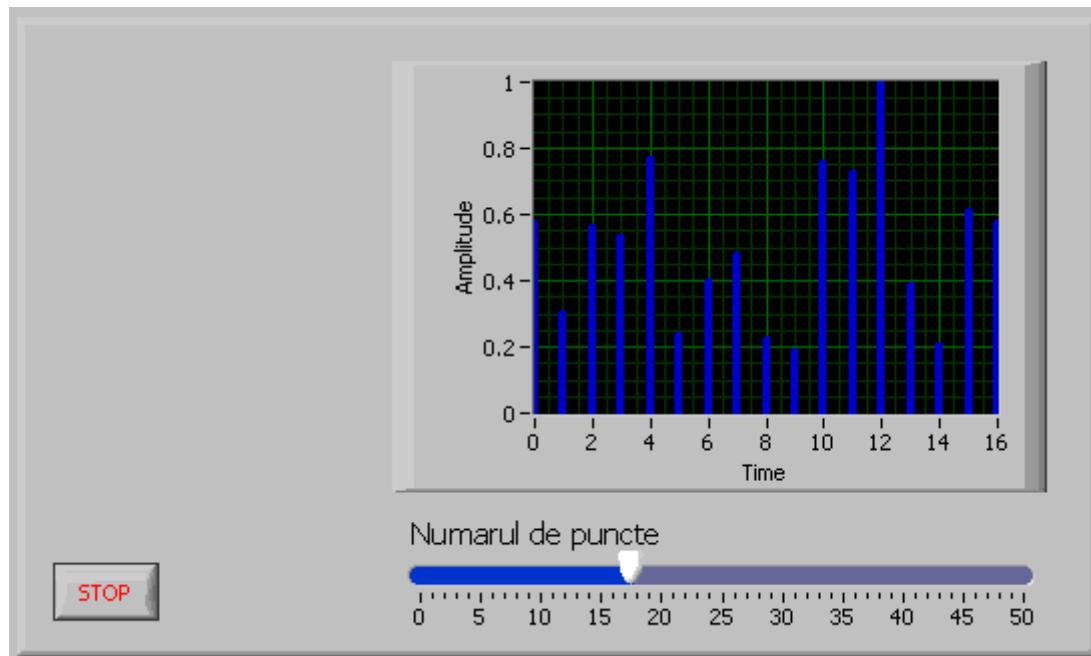


Aplicatia este similara cu aplicatia anterioara cu deosebirea ca nu mai folosim functia  $e^x$  ci  $\sin(x)$ .

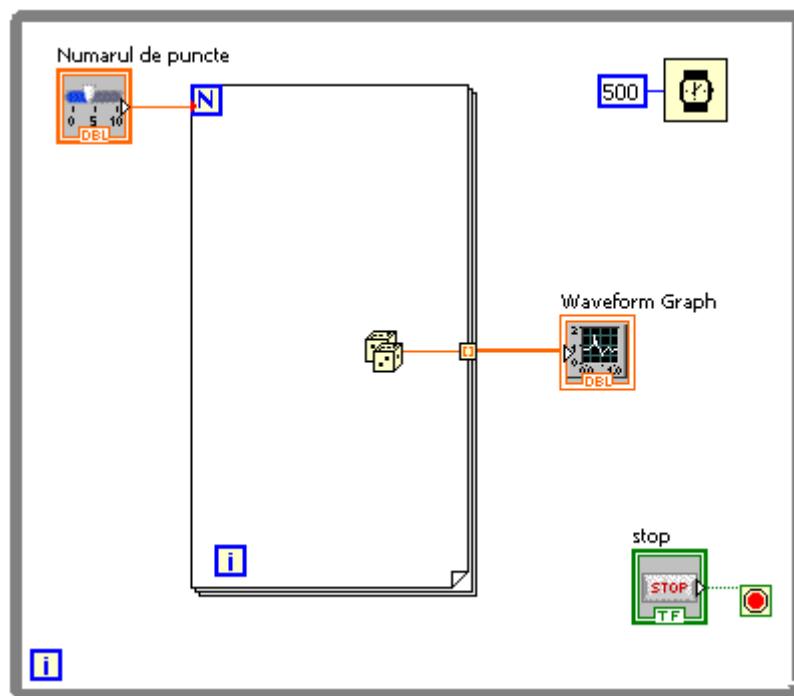


Dupa cum se observa valoarea curenta  $x=(2*\pi*Nr\ de\ perioade/Numarul\ de\ puncte)*i$ .

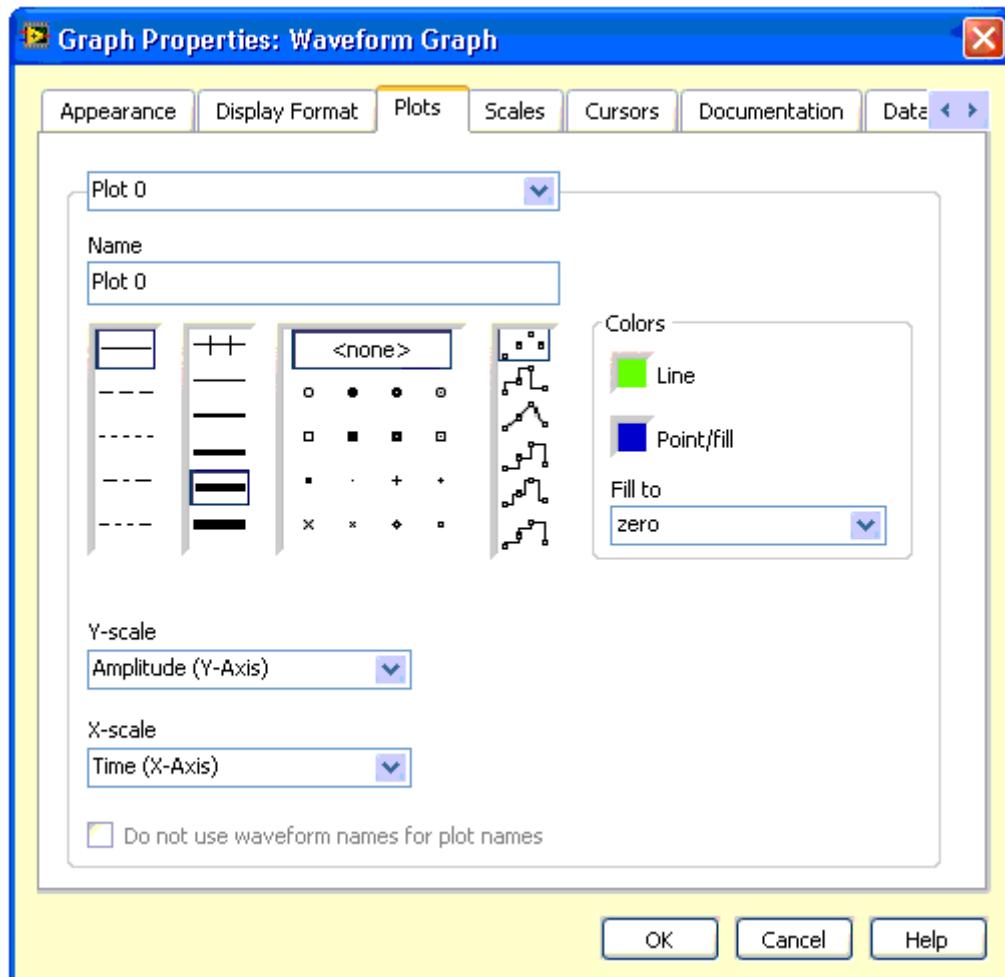
Vom folosi in continuare controlul Waveform Graph pentru a trasa grafice sub forma de histograma [for v0 v23](#).



Vom folosi un generator de numere aleatoare si vom afisa valorile generate intr-un control de tipul Waveform Graph

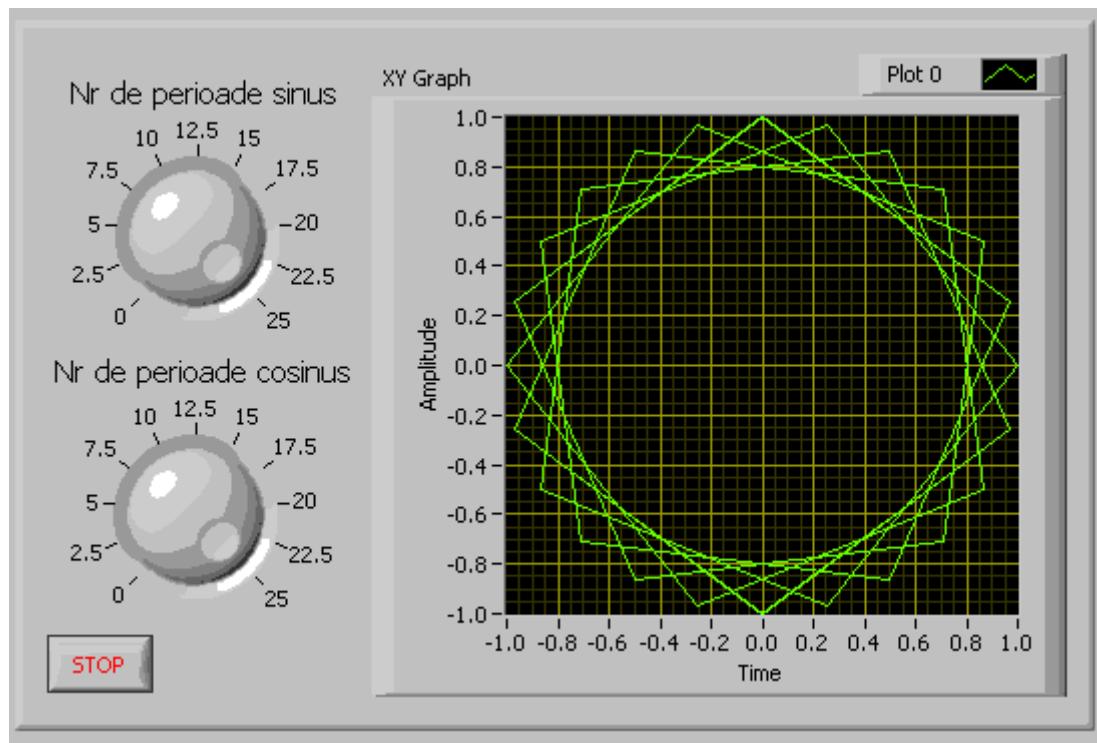


Pentru a obtine afisarea sub forma de histograma, a fost setat corespunzator control de tipul Waveform Graph

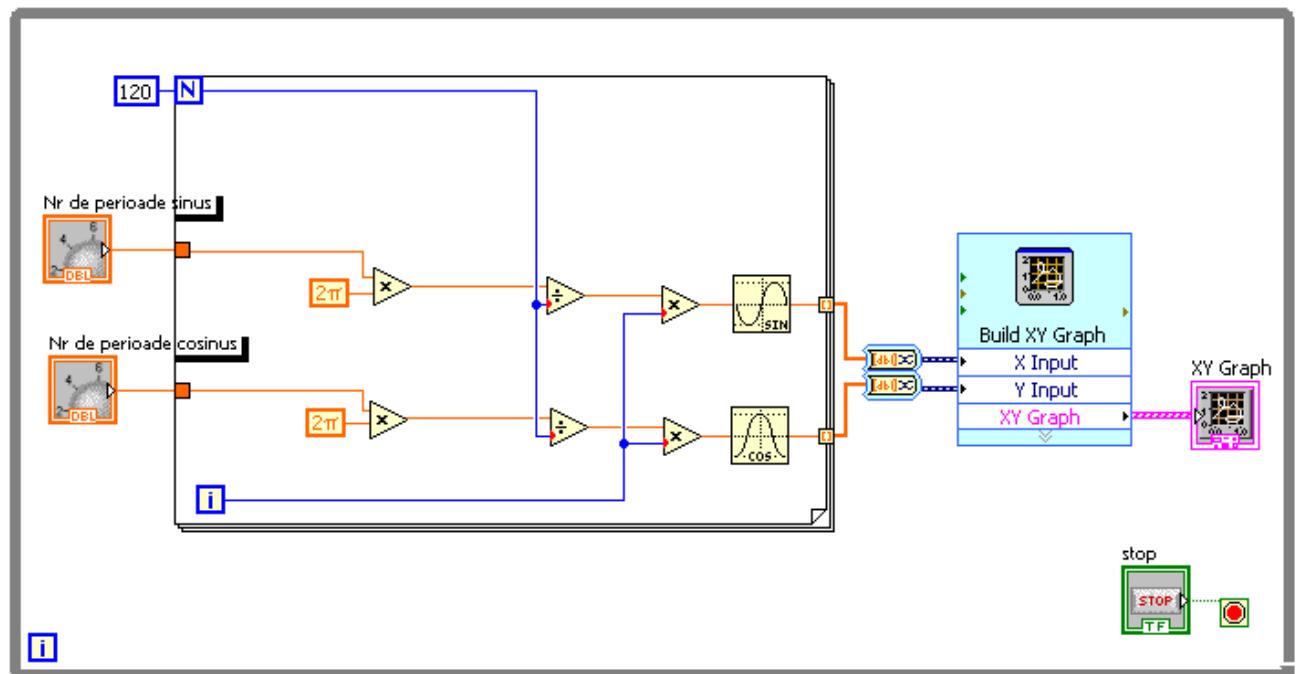


Am folosit pana acum controlul Waveform Graph pentru a trasa evolutia in timp a diverselor functii. Vom folosi in continuare controlul XY Graph pentru a trasa grafice in coordonatele X si Y.

Vom realiza aplicatia **for v0 v24** in care valoarea coordonatei X va fi data de functia  $\sin(x)$  iar valoarea coordonatei Y va fi data de functia  $\cos(y)$



Pe diagrama bloc vom conecta la intrarea controlul XY Graph cele două valori:  $\sin(x)$  și  $\cos(y)$

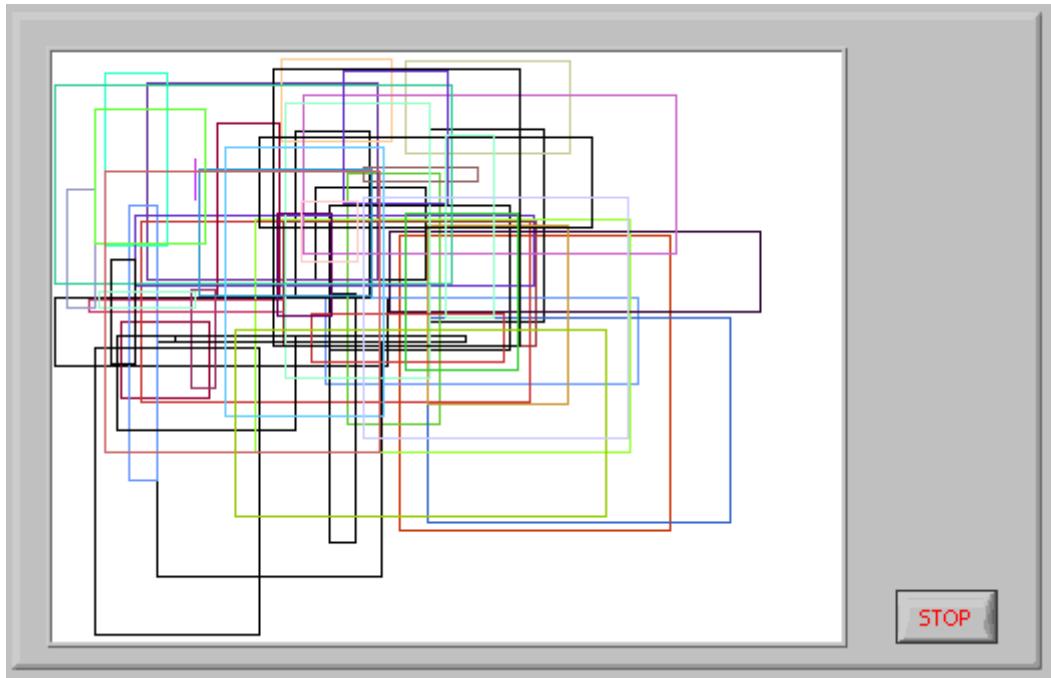


Iesirile din structura "For Loop" sunt de tipul Array saa ca o sa revenim asupra lor in cardul sectiunii "Utilizarea tablourilor".

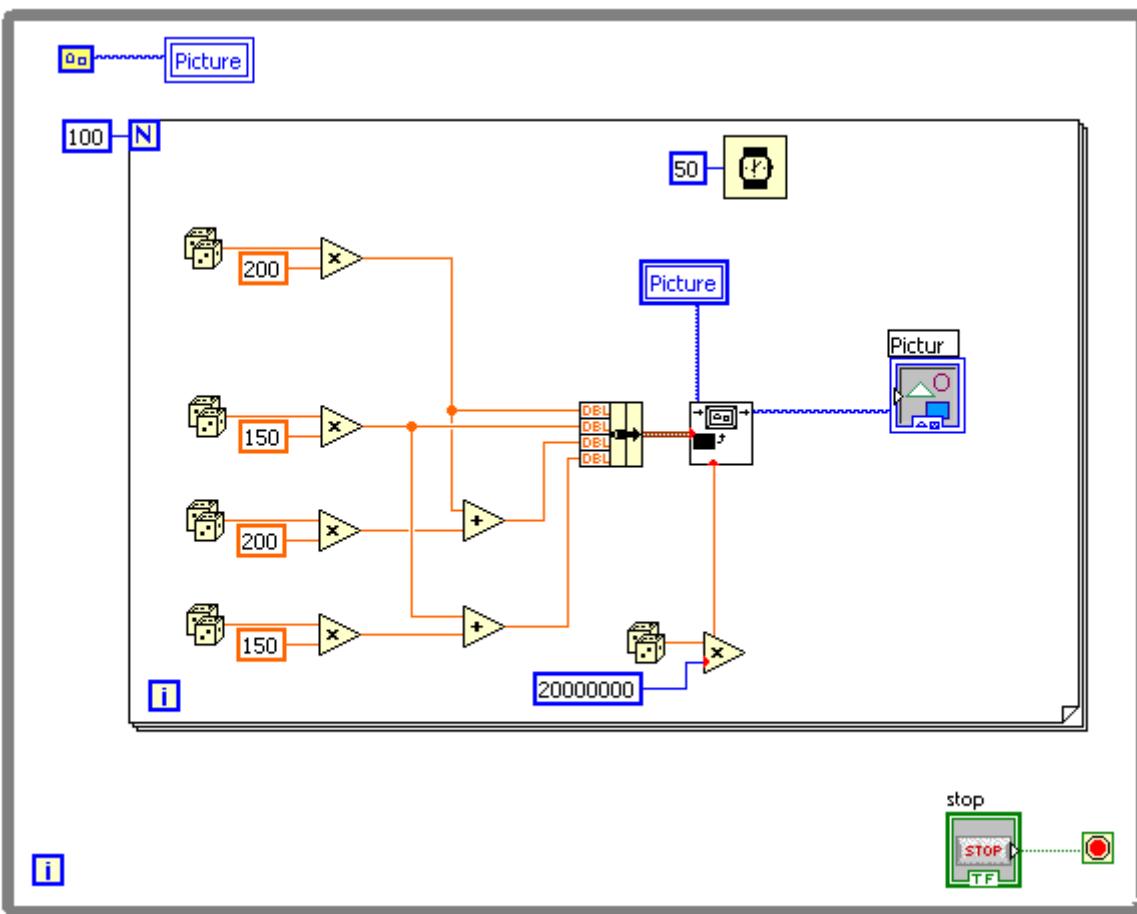
- Utilizarea structurii "For Loop" pentru grafica si figuri geometricice

Am utilizat facilitatile de trasare a figurilor geometrice si in cadrul sectiunii "Structura While Loop". continuam sa folosim si in cadrul acestiei sectiuni facilitatile grafice oferite in cadrul mediului de dezvoltare "LabView"

Vom realiza o aplicatie for v0 v30 in care vom trasa aleator dreptunghiuri intr-un control de tip "Picture"

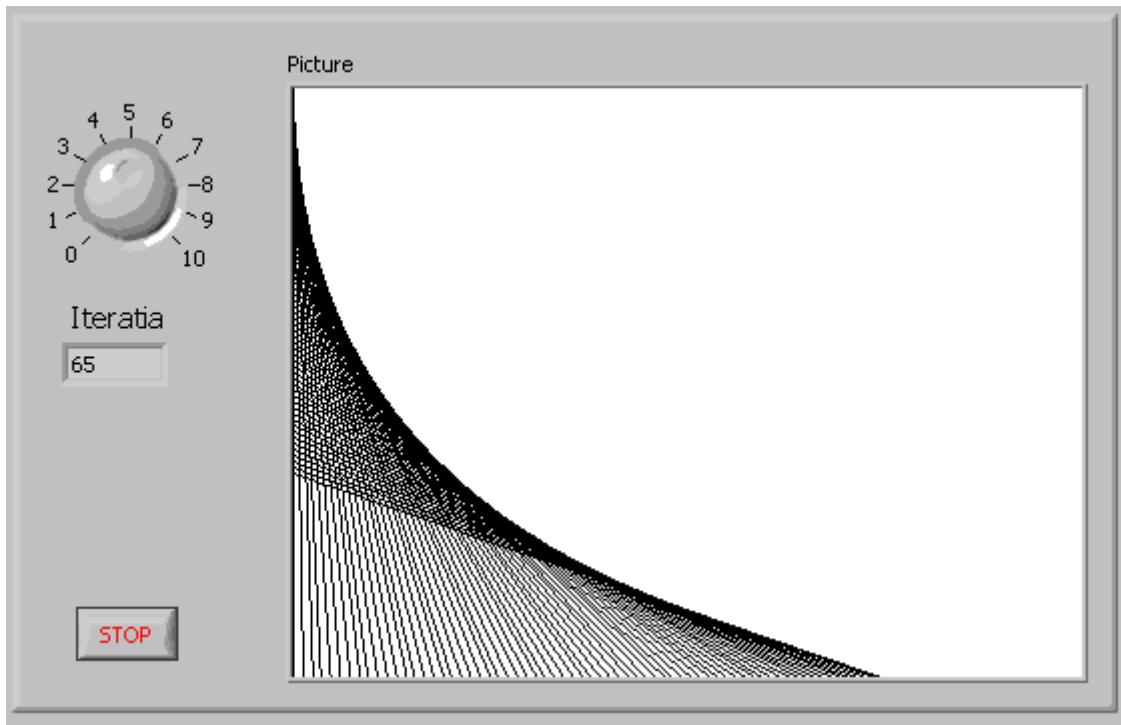


Se va folosi o structura "For Loop" pentru a trasa dreptunghiuri de dimensiuni si culori aleatoare.

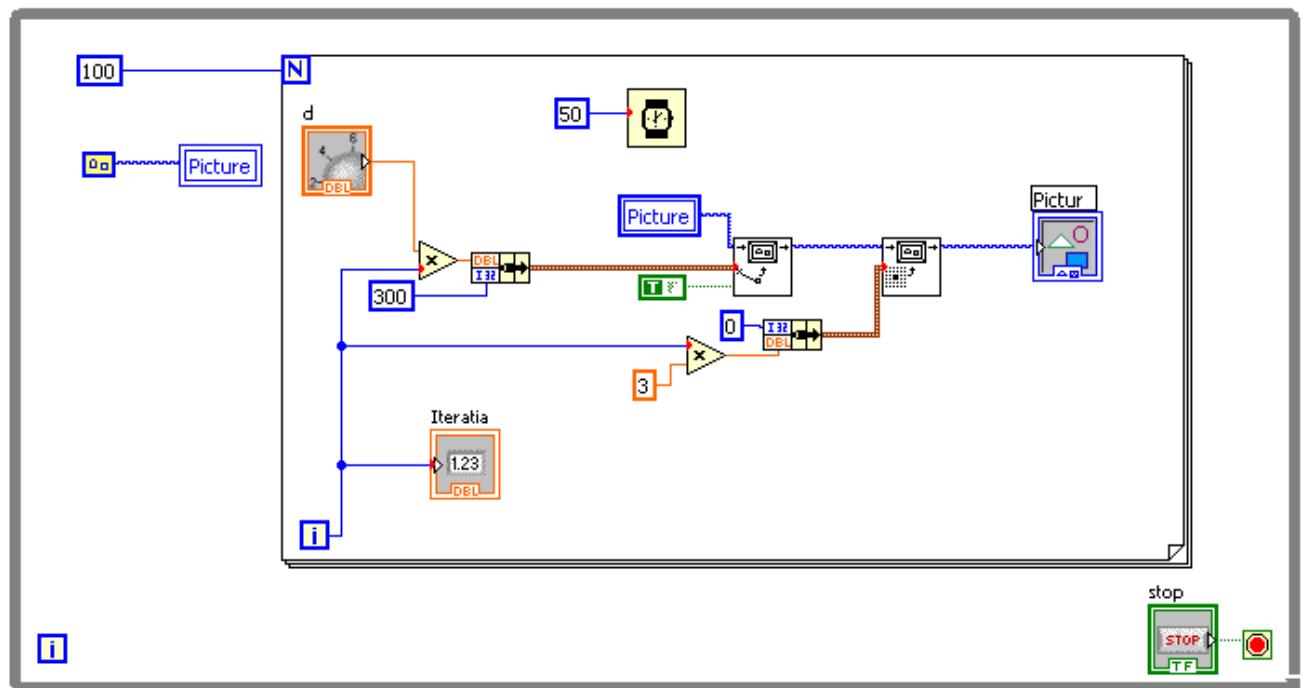


S-a folosit functia "Draw Rectangle" in interiorul unei structuri "For Loop" cu iteratia =100.

Urmatoarea aplicatie [for v0\\_v31](#) , traseaza 100 de linii a caror coordonate sunt incrementate continuu, factorul de incrementare este dat de utilizator prin intermediul unui control.



Se va folosi deci functia "Draw Point" si "draw Line" intr-o structura "For Loop".



Se observa ca toate liniile au coordonata de inceput  $x_0=0$   $y_0=\text{varabil}$  si coordonata de sfarsit  $x=\text{variabil}$  si  $y=300$

In urmatoarea aplicatie [for\\_v0\\_v32](#)  $x_0$  si  $y$  si numarul de iteratii pot fi setate de utilizator prin intermediul controalelor din imaginea de jos:

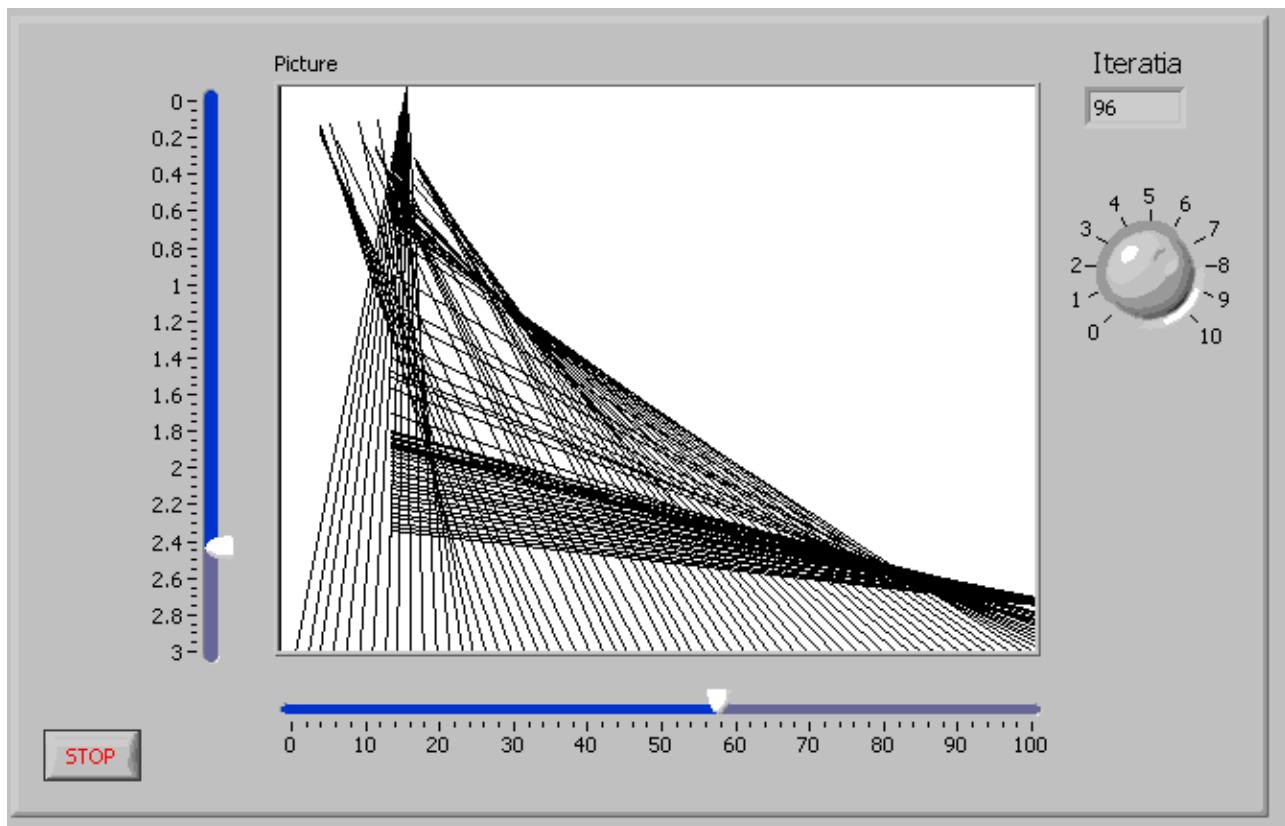


Diagrama logica, reflecta utilizarea acestor controale.

