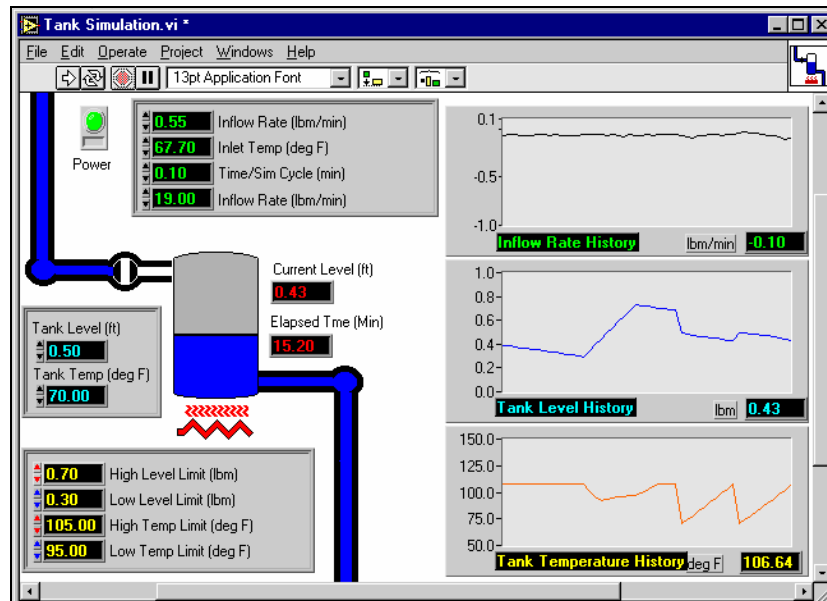


Carmen Bujoreanu



LABVIEW, PRIETENUL NOSTRU

IAȘI, 2015

CUPRINS

1. Introducere.....	
1.1. Prezentarea opțiunilor meniului afișat la lansarea LabVIEW.....	
1.2. Consultarea exemplelor instalate.....	
1.3. Rolul și componența barei cu unelte din fereastra panoului frontal.....	
1.4. Prezentarea elementelor din bara cu unelte din fereastra diagramei bloc.....	
1.5. Prezentarea și utilizarea meniurilor contextuale.....	
1.6. Prezentarea barei orizontale cu meniuri.....	
1.7. Prezentarea casetei cu unelte generale.....	
1.8. Metode de asistare în LabVIEW.....	
2. Controale și indicatoare.....	
2.1. Considerații teoretice.....	
2.2. Tipuri de controale și indicatoare.....	
2.3. Controale și indicatoare de tip numeric.....	
2.4. Controale și indicatoare de tip boolean.....	
2.5. Controale și indicatoare tip șir de caractere.....	
2.6. Controale și indicatoare tip tablou și grupare de date.....	
2.7. Controale și indicatoare pentru reprezentări grafice. Exemple....	
2.8. Aplicații personale.....	
3. Meniuri proprii.....	
3.1. Meniuri proprii specifice elementelor numerice. Exerciții.....	
3.2. Meniuri proprii specifice elementelor cu scală. Exerciții.....	
3.3. Meniuri proprii specifice elementelor booleene. Exerciții.....	
3.4. Meniul propriu al elementelor de tip Cluster. Exerciții.....	
3.5. Ordonarea componentelor. Exerciții.....	
3.6. Meniuri pentru elemente ListBox, Table și Ring. Exerciții.....	
3.7. Variabile locale. Exerciții.....	
3.8. Noduri de proprietăți. Exerciții. Tema.....	
4. Funcții.....	
4.1. Considerații teoretice. Exemple.....	
4.2. Funcții pentru valori numerice. Exerciții.....	
4.3. Funcții pentru valori boolene. Exerciții.....	
4.4. Funcții pentru valori alfanumerice (string). Exerciții.....	
4.5. Inserarea simbolurilor de funcții în diagrama. Exerciții.....	
4.6. Realizarea legăturilor în diagramă. Exerciții.....	

4.7. Fluxul de date. Exerciții.....	
4.8. Salvarea unei aplicații. Exerciții.....	
4.9. Funcții pentru valori scalare. Exerciții. Tema.....	
4.10. Funcții pentru valori vectoriale (Array). Exerciții. Tema.....	
4.11 Funcții pentru elemente de tip Cluster. Exerciții.....	
4.12. Elemente și funcții de tip Picture. Exerciții. Tema.....	
5. Structuri de programare.....	
5.1. Inserarea structurilor în diagramă. Exerciții.....	
5.2. Structura secvențială (Sequence). Exerciții.....	
5.3. Structura cauzală (Case). Exerciții.....	
5.4. Structura repetitivă cu număr fix de iterații (bucă For). Exerciții	
5.5. Structura repetitivă cu condiție de terminare (bucă While).	
Exerciții.....	
5.6. Registri de transfer în structuri repetitive. Exerciții. Tema.....	
6. Elementul Waveform Chart. Exercițiu.....	
7. Elementul Waveform Graph. Exercițiu.....	
8. Elementul XY Graph. Exercițiu. Tema.....	
Aplicații-1.Simularea unui subsistem de control al unui manipulator	
industrial.....	
Aplicații-2.Simularea sistemului de automatizare al unui cuptor de	
tratament termic.....	
Aplicații-3.Achiziția și înregistrarea datelor pentru măsurarea	
momentelor de frecare și de frânare în rulmenții testați la gripare.....	
Aplicații-4.Determinarea experimentală a constantei elastice a unui	
resort. Medoda echilibrului.....	
9. Bibliografie	

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 1

Instrumentatia virtuala reprezinta ceea ce era acum un deceniu lantul de masurare, la care s-a inlocuit partea de instrumente fizice cu instrumente virtuale. Un instrument virtual este compus dintr-o parte hardware (in principal un convertor analogic digital) si o parte software care permite configurarea instrumentului dupa dorinta utilizatorului.

Cu putin timp in urma, utilizatorul isi configura instrumentul fizic cu ajutorul butoanelor si potentiometrelor, facea conexiunile dintre instrument si aparatele de inregistrare pe hartie sau electronice si trebuia sa le aiba pe toate in acelasi loc pentru a le supraveghea si a le manevra.

Acum, utilizatorul introduce o placa de achizitie date in computer si cu ajutorul soft-ului de programare grafica isi configureaza instrumentul de masurare si in plus poate crea si cate inregistratoare grafice doreste. Toata partea fizica a lantului de masurare aflandu-se pe o placa controlata de microprocesorul computerului, utilizatorului nu-i ramane decat sa se conecteze la traductoarele specifice marimii de masurat si sa stie sa manevreze un mouse. Reprezentarea instrumentelor fizice se face pe monitorul computerului cu ajutorul elementelor grafice existente in biblioteca limbajului de programare grafica.

Exista **doua posibilitati** de a lucra cu instrumentatie virtuala:

- doresti sa-ti creezi singur instrumentele necesare si atunci trebuie sa cunosti limbajul de programare grafica si sa dispui de acesta (necesarul de timp minim pentru invatare ar fi de 30 de ore);
- doresti sa fii numai utilizator de instrumentatie virtuala, si atunci trebuie sa-ti cumperi de la un dezvoltator de aplicatii un program executabil ce va face numai ceea ce iti doresti.

Avantajele instrumentatiei virtuale se deduc si din cele spuse mai sus:

- ocupa un spatiu mic (practic un computer si un monitor);
- poate fi cu elemente distribuite (pot masura in mai multe locuri odata);
- datele se pot transmite prin internet (laboratorul de masurare se poate afla intr-un anume loc iar analiza rezultatelor se poate face in cu totul alta parte);
- instrumentele nu mai ocupa un spatiu fizic (o magazie) ci sunt stocate in memoria computerului;
- flexibilitate maxima privind configurarea instrumentelor (oricand se poate sterge un instrument din memorie si se poate face altul, se pot adauga elemente de comanda sau indicatoare, canale sau memorie);
- dispar practic problemele legate de murdarirea comutatoarelor sau imperfectiunea conexiunilor;
- dispar sau se micsoareaza mult erorile de offset sau de calibrare;
- costurile privind achizitia de aparate si intretinerea lor se reduc foarte mult, tinand cont ca o singura placa multifunctionala de achizitie date impreuna cu softul aferent poate inlocui o multime de alte instrumente fizice dedicate;
- interfata grafica foarte prietenoasa cu utilizatorul;
- timpul relativ mic de invatare a limbajului de programare grafica;
- multimea de instrumente virtuale gata construite pentru a masura, a face analiza semnalului, a-l prelucra si a-l transmite oriunde doreste utilizatorul;

Un **sistem de achizitie date** (sau de masurare) este format, in principal, din traductoare, conditionatoare de semnal, placi de achizitie date, software si computer.

Laboratorul urmareste prezentarea conceptului de instrumentatie virtuala, caracteristicile mediului **LabVIEW** si structura unui instrument virtual. Dupa realizarea panoului frontal al IV, trebuie implementata functionalitatea programului; se construiesc diagrama bloc, care reprezinta codul sursa al instrumentului (arata **CUM ?** se rezolva problema). Pentru realizarea diagramei bloc, se utilizeaza limbajul grafic **G**. Sunt prezentate elemente de baza care trebuie cunoscute pentru ca studentii sa realizeze si sa utilizeze instrumente virtuale.

1.1 Prezentarea opțiunilor meniului afișat la lansarea LabVIEW

Mediul LabVIEW se lansează făcând dublu-clic cu butonul din stânga al mouse-lui pe pictograma asociată programului (aflată pe desktop-ul sistemului Windows) sau alegând cu mouse-ul meniul

“Start/ Programs/ National Instruments LabVIEW / LabVIEW”;

Pe ecranul calculatorului apare meniul prezentat în figura 1.1

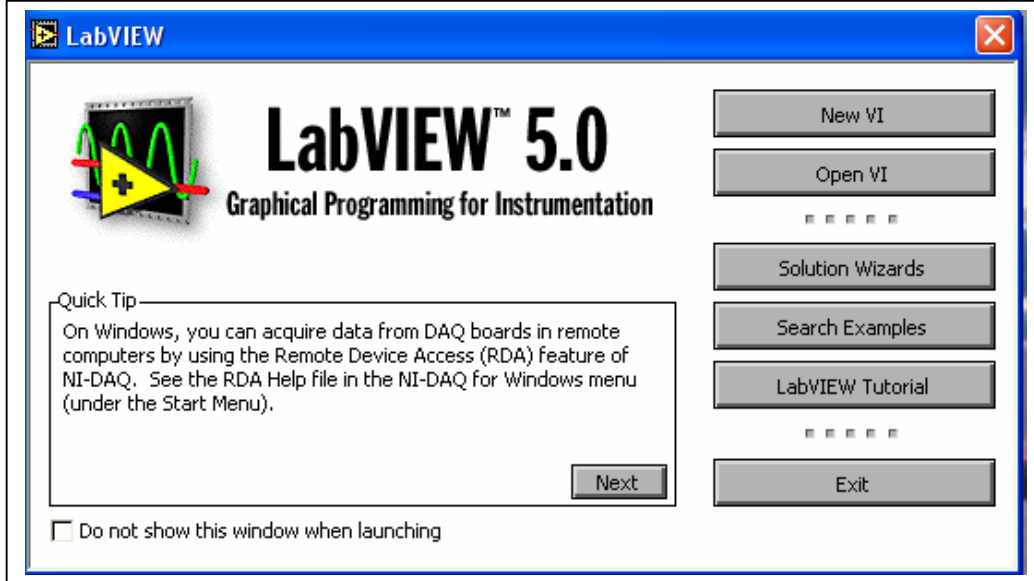


Fig. 1.1. Meniul afișat după lansarea mediului LabVIEW

Semnificația butoanelor din fereastra ilustrată în figura 1.1 se prezintă în continuare:

- “*IV Nou* “ (**New VI**):
Se creează un *IV* nou.
- “*Deschide un IV*” (**Open VI**) :
Se deschide un *IV* existent.
- ” *Vrăjitorul de realizare a unei soluții* “ (**Solution Wizard**):
Se lansează un utilitar, care permite realizarea în mod interactiv a unei aplicații personalizate pentru achiziția de date sau instrumentale.
- “ *Căutare exemple* ”(**Search Examples**):
Se lansează un utilitar, care permite consultarea de exemple distribuite de firma NI odată cu mediul **LabVIEW**.
- ” *Îndrumător LabVIEW* ” (**LabVIEW Tutorial**):
Se lansează un program demonstrativ de învățare (tutorial) despre LabVIEW , care folosește opțional placa de sunet. Programul demonstrativ trebuie să fie instalat anterior (ocupă în jur de 40 MB) sau să existe CD-ul original poziționat în unitate.
- ” *Ieșire* ” (**Exit**):
Se părăsește mediul **LabVIEW**.

Acțiuni care trebuie realizate:

- Se indentifică acțiunile realizate de butoanele din dreapta ferestrei (fig. 1.1), afișate la lansarea mediului **LabVIEW**.
- Se alege opțiunea “**New VI**” pentru creerea unui *IV* nou. Se observă că pentru noul *IV* au fost deschise două ferestre, suprapuse una peste cealaltă, asemenea a două file de carte. Prima fereastră este *Fereastra Panoului Frontal al IV*; sub această fereastră se găsește *Fereastra Diagramei Bloc a IV*.
- Se trece din *Fereastra Panoului Frontal* în *Fereastra Diagramei Bloc*, selectând cu mouse-ul din meniul orizontal “ **Windows / Show Diagram**”.
- Din *Fereastra Diagramei Bloc* se va reveni înapoi în *Fereastra Panoului Frontal* alegând din meniul orizontal “ **Windows / Show Panel**”.
- Fiind activată *Fereastra Panoului Frontal*, se alege “ **File/ Close**” pentru închiderea *IV*.

1.2 Consultarea exemplelor instalate

În continuare se prezintă modalitatea de consultare a exemplelor instalate odată cu mediul **LabVIEW**.

Consultarea exemplelor constituie principala sursă de învățare a diferitelor tehnici de programare în **LabVIEW** și de utilizare a diferitelor tipuri de obiecte și funcții:

- După lansarea mediului **LabVIEW**, se obține pe ecranul calculatorului meniul din figura 1.1. Se alege opțiunea “*Caută exemple*” (**Search Examples**).
- Se prezintă o fereastră meniu, în care opțiunile (de tip text) sunt grupate pe diferite subiecte. Se vor nota în caiete principalele grupe de subiecte disponibile.
- Din secțiunea “*Demonstrații*” (**Demonstrations**) se alege opțiunea “*Controlul Procesului*” (**Process Control**). Exemplele disponibile apar grupate pe verticală. În coloana din stânga este afișată denumirea *IV*, iar în partea dreaptă o scurtă descriere a exemplului. Prin selectarea cu mouse-ul a denumirii unui exemplu (afișat în prima coloană), se va declanșa deschiderea *IV* pentru consultare.
- Înainte să se selecteze cu mouse-ul având denumirea “ *Controlul Procesului de Amestecare* ” (**Control Mixer Process**), se va nota în caiete explicația din coloana dreaptă.
- După consultarea *Ferestrei Panoului Frontal* se trece în *Fereastra Diagramei Bloc* alegând “ **Windows/ Show Diagram**”. Se revine în *Fereastra Panoului Frontal* prin “ **Windows/ Show Panel**”.
- Se închide exemplul, alegând din meniul orizontal ” **File/ Close**”.
- După revenirea în meniul afișat de **LabVIEW** (fig.1.1), se poate alege din nou opțiunea “*Caută exemple*” (**Search Examples**), pentru a deschide alte exemple.

Observație:

Nu se vor salva modificările (voite / involuntare) realizate în timpul consultării exemplelor oferite de biblioteca **LabVIEW**. Exemplele trebuie păstrate în varianta originală, oferite de firma NI.

1.3 Rolul și componența barei cu unelte din fereastra panoului frontal

Pe marginea unui exemplu de *IV* se va prezenta în continuare rolul și componența barei cu unelte din fereastra panoului frontal.

Se citesc și se rețin elementele din secțiunea care urmează și care prezintă elemente din bara cu unelte ale ferestrei panoului frontal.

Fereastra Panoului Frontal și ale *Diagramei Bloc* din *IV* posedă fiecare câte o bară orizontală cu unelte. Aceste bare cu unelte sunt implementare prin intermediul unor butoane, liste derulante sau indicatoare de stare, utilizate prin editarea, trasarea și execuția *IV*. O parte din opțiunile celor două bare cu unelte sunt comune și anume cele dispuse în partea dreaptă a barei.

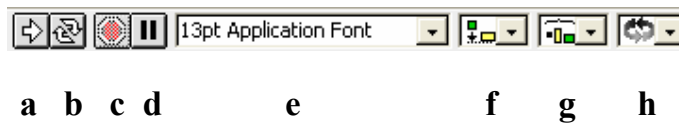


Fig. 1.2. Bara cu unelte din fereastra panoului frontal

Prin butonul “*Rulare IV*” (fig.1.2 a) se lansează în execuție (comanda **Run**) programul curent.

În figura 1.2 se prezintă cele două variante sub care poate apărea butonul **Run** în timpul execuției *IV*: cazul 1.2 a reflectă situația în care *IV* este modulul principal al aplicației (apelează alte *IV*), iar cazul 1.2b corespunde situației în care *IV* este executat de către *IV* apelant. Elementul din figura 1.2a se modifică în cel redat de figura 1.2 c în momentul apariției unor erori (cel mai adesea prezente în diagrama bloc), datorită cărora *IV* nu poate fi compilat. Pentru consultarea surselor de eroare, se apasă cu butonul stâng al mouse-ului pe butonul din figura 1.2c (simbolizând o “*săgeată frântă*”); se afișează o listă verticală conținând sursele de eroare, care se pot inspecta și localiza în diagrama bloc.

Pentru rularea continuă (se reia la infinit execuția programului după terminare) a *IV* se va acționa asupra elementului “*Rulare Continuă a IV*” din figura 1.2 b.

Oprirea execuției (indiferent de starea de execuție curentă) unui *IV* se face prin butonul de “*Oprire a Rulării IV*” (fig. 1.2 c). Este o metodă practică de utilizator pentru oprirea necondiționată a rulării unui *IV* (care se execută cu afișarea panoului frontal).

Observație:

Dacă se poziționează cursorul mouse-ului deasupra oricărui element din bara cu unelte, se afișează în dreptul cursorului mouse-ului un mesaj (tool tip text), care informează despre denumirea uneltei.

Ținând cont de observația anterioară, să se rețină în caiete utilizarea recomandată, obținută pentru restul elementelor din bara orizontală cu unelte a ferestrei panoului frontal (fig. 1.2 d-h).

Se lansează mediul **LabVIEW** și din meniul prezentat în figura 1.4 se alege opțiunea “*Caută Exemple*” (**Search Examples**).

Se localizează *IV* denumit “*Rădăcina Pătrată*” (**Square Root**), din secțiunea “**Fundamentals/Structures**”; se notează explicația din partea dreaptă a numelui exemplului și se selectează cu mouse-ul pentru deschidere .

Exemplul selectat este deschis în **LabVIEW** pentru consultare. Se va observa pentru diferite date de intrare ale controlului cu identificator “*Număr*” (**Number**) rezultatul afișat de indicator “*Valoarea Rădăcinii Pătrate*” (**Square Root Value**).

1.4 Prezentarea elementelor din bara cu unelte din fereastra diagramei bloc

Se pornește mediul **LabVIEW** și se deschide *IV* denumit “*Rădăcină Pătrată*” (**Square Root**) din secțiunea “**Fundamentals/ Structures**”.

Se reține că atunci când un *IV* existent este prezentată automat doar fereastra care reține realizarea panoului frontal; utilizatorul va trebui să solicite explicit afișarea diagramei bloc, pentru consultarea implementării funcționalității *IV*. În situația creării unui *IV* nou, sunt afișate automat ambele ferestre: panoul frontal și diagrama bloc.

Se trece în fereastra care reține implementarea funcționalității instrumentului: se selectează din bara cu meniuri opțiunea “*Ferestre/ Afișează Diagrama Bloc*” (**Windows/Show Diagram**).

Se rețin elementele care apar în bara cu unelte din fereastra diagramei bloc, prezentată în figura 1.7.



Fig. 1.3. Bara cu unelte din fereastra diagramei bloc

Acționând asupra butonului din figura 1.3, e, se activează/inactivează regimul de rulare a *IV* cu afișarea fluxului datelor. Dacă butonul afișează imaginea unui “*bec emițând lumină*”, atunci pe parcursul rulării *IV* se poate urmări ordinea de executare a componentelor diagramei bloc (este inhibată și facilitarea de execuție paralelă a unor sub*IV*).

Trasarea interactivă (programul așteaptă ca utilizatorul să indice execuția următoarei instrucțiuni) a execuției *IV* se face utilizând elementele din figura 1.3, f-h.

Utilizând butonul din figura 1.3, f se execută următoarea instrucțiune a *IV* , după care se așteaptă o nouă interacțiune din partea utilizatorului; în cazul în care instrucțiunea curentă corespunde apelului unui sub*IV* (procedură), atunci se va executa prima instrucțiune din *IV* apelat. Pentru o trasare a execuției unui *IV* pe blocuri de instrucțiuni (toate instrucțiunile aparținând unui bloc vor fi executate simultan), se va utiliza butonul din figura 1.3, g; dacă instrucțiunea următoare este un apel de sub*IV* , atunci se va executa întreg programul apelat. Revenirea din blocul apelat în cel apelat se face la apăsarea butonului ilustrat în figura 1.3, h.

În timpul rulării unui *IV* (rulare cu afișarea fluxului datelor), se poate declanșa de către utilizator o întrerupere temporară (nelimitată ca perioadă) a execuției programului prin butonul de “*Generare Pauză*” (fig. 1.3, d); continuarea execuției *IV* se face apăsând din nou.

În fereastra diagramei bloc a *IV* “Rădăcină Pătrată” se urmărește fluxul datelor. Se apasă butonul din figura 1.3, e pentru a urmări fluxul datelor (imaginea unui “bec emițând lumină”). Se utilizează butonul de rulare a *IV* (fig. 1.3, a) și se observă modul în care se execută “instrucțiunile” care implementează funcționalitatea *IV*. Se repetă rularea *IV*.

În timpul rulării *IV*, fiind activat modul de afișare a fluxului datelor, se exersează utilizarea Butonului de întrerupere temporară și de reluare a execuției programului. Se inactivează modul de afișare a fluxului datelor de rulare. Se închide *IV* “Rădăcina Pătrată”.

1.5 Prezentarea și utilizarea meniurilor contextuale

În meniul de programare **LabVIEW**, aproape fiecare obiect folosit la realizarea unui *IV* are asociat un meniu conținând opțiuni, prin care se pot realiza operații de actualizare a unor proprietăți (vizuale sau comportamentale) asociate; acestea sunt meniurile *pop-up*, numite și *meniuri contextuale*.

Opțiunile care apar într-un meniu contextual sau dependente de tipul obiectului:

- panoul frontal;
- diagrama bloc;
- obiecte din panoul frontal (controale, indicatoare, etichete libere)
- funcții din diagrama bloc;
- apeluri de *IV*.

Se selectează și păstrează cursorul mouse-ului deasupra unui obiect (prezent în panoul frontal sau diagrama bloc); afișarea meniului contextual asociat se face apăsând butonul din dreapta mouse-ului. Prin dotarea editoarelor cu meniuri contextuale (*pop-up*), într-un meniu grafic de programare (**Windows**), se oferă o metodă rapidă de acces la comenzile obișnuite.

Se pornește mediul **LabVIEW** și se alege deschiderea unui *IV* nou.

În fereastra *Panoului Frontal* al noului *IV* se poziționează cursorul mouse-ului în interiorul acestuia. Apăsând butonul din dreapta al mouse-ului se afișează meniul contextual asociat panoului frontal; se reține că prin această metodă se afișează la poziția cursorului mouse-ului caseta cu colecția disponibilă de controale și indicatoare, din care se alege obiectul de interfață dorit, pentru a fi plasat în panoul frontal.

Observație:

Nu este necesar să se permită afișarea în permanență, pe ecran a casetei cu obiecte pentru realizarea panoului frontal, din motive de economisire a spațiului util, de lucru; când este nevoie, se apelează meniul contextual asociat ferestrei panoului frontal, conținând caseta cu controale și indicatoare.

Laborator 1

Se selectează printr-un clic cu butonul stâng al mouse-ului pictograma care simbolizează un control numeric digital din caseta de controale și indicatoare, meniul “ **Numeric/ Digital Control**”. Obiectul de tip control va fi plasat pe suprafața panoului frontal și la poziția cursorului mouse-ului printr-un nou clic cu butonul stânga.

După ce controlul numeric cu afișare digitală a fost plasat pe suprafața panoului frontal, se observă că în partea superioară a chenarului controlului este afișat un mic dreptunghi înnegrit, de dimensiunea unei litere. Se introduce de la tastatură textul “**ControlNr1**”, reprezentând identificatorul asociat obiectului (eticheta proprie sau owned label), după care se apasă pe butonul afișat în bara cu unelte din fereastra panoului frontal.

Se afișează meniul contextual asociat controlului “**ControlNr1**” și se reține denumirea opțiunilor posibile (fig. 1.4), descrise în Laboratorul 2.

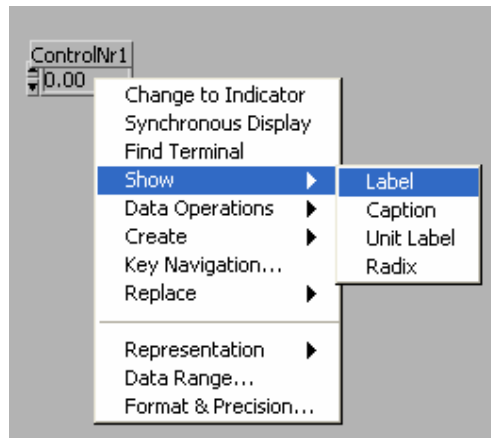


Fig. 1.4. Meniul contextual asociat unui control numeric

1.6 Prezentarea barei orizontale cu meniuri

Având mediul **LabVIEW** deschis, se concentrează atenția asupra meniului orizontal, afișat în partea superioară a ferestrei panoului frontal (fig. 1.5).

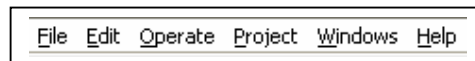


Fig. 1.5. Bara cu meniuri din LabVIEW

Bara orizontală cu meniuri conține opțiuni pentru realizarea unor acțiuni obișnuite (gestionare fișiere, editare), implementate în majoritatea aplicațiilor **Windows**, dar și opțiuni specifice mediului **LabVIEW**. În continuare se prezintă cele mai importante opțiuni, utilizatorii fiind invitați să identifice locul opțiunilor în bara cu meniuri și să se familiarizeze cu acestea pe parcursul exercițiilor.

În momentul selectării, prin butonul stânga al mouse-ului, a unei opțiuni din bara orizontală cu meniuri (fig. 1.5), apare un subdomeniu derulant, conținând mai multe opțiuni dispuse pe verticală.

Laborator 1

Prin selectarea opțiunii “Fișier” (File) se permite utilizatorului realizarea unor acțiuni (încărcare, salvare, compilare) de gestionare a fișierelor, care găzduiesc *IV*-le. Submeniul derulant, afișat în momentul selectării acestei opțiuni, este evidențiat în fig. 1.6.

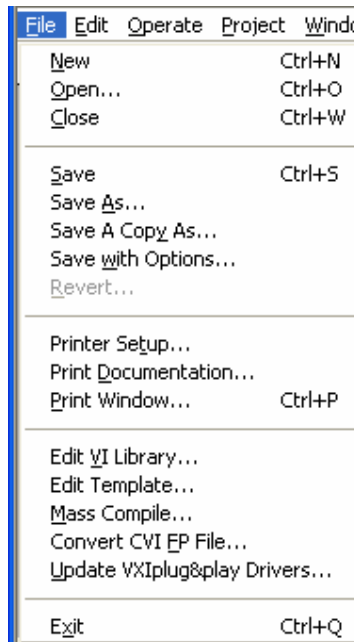


Fig.1.6 Submeniul derulant „Fișier”

Se prezintă în continuare semnificația opțiunilor din submeniul “Fișier” (File):

- Se declanșează operația de creare a unui *IV* nou (fig. 1.6, a).
- Se deschide un *IV* existent (fig. 1.6, b).
- Se închide fereastra curentă (fig.1.6, c). Dacă acțiunea se declanșează din fereastra panoului frontal, atunci se închide fișierul, care reține implementarea *IV*.
- Se realizează salvarea modificărilor realizate asupra *IV* (fig. 1.6,d).
- Se permite salvarea *IV*(deschis în editare) sub un alt nume de fișier și eventual în alt director și pe o altă unitate de disc (fig.1.6, e).
- Se realizează o dublură a *IV* și permite salvarea într-un fișier sub un alt nume (fig.1.6, f).
- Se oferă opțiuni pentru tipărirea programului (fig.1.6, g): panou frontal, diagrama bloc, pictograma asociată, *IV*-le apelate ș.a. Tipărirea programului se poate face la imprimantă sau într-un fișier (format **HTML** sau **RTF**).
- Se apelează un editor specializat (fig. 1.6, h), care permite realizarea de operații de ștergere a *IV* dintr-un fișier bibliotecă (**LLB**) și indicarea modulului principal din bibliotecă (se afișează la începutul listei numerelor de *IV* din bibliotecă).
- Se realizează conversia tuturor *IV* dintr-un fișier bibliotecă la versiunea curentă a mediului **LabVIEW** (fig. 1.6, i).

Realizarea tipăririi *IV* într-un fișier reprezintă o modalitate foarte utilă de arhivare, independentă de versiunea mediului **LabVIEW** utilizat. Parcurgând fișierul documentație al *IV* se poate reface programul, efortul de reintroducere fiind neglijabil în comparație cu siguranța păstrării codului sursă al aplicației.

Laborator 1

Opțiunea de “Editare” (**Edit**) din meniul bară orizontal pune la dispoziția dezvoltatorului programului câteva operații, care pot fi folosite în crearea sau modificarea panoului frontal și a diagramei bloc. Fig. 1.7 prezintă opțiunile subdomeniului derulant “Editare”.

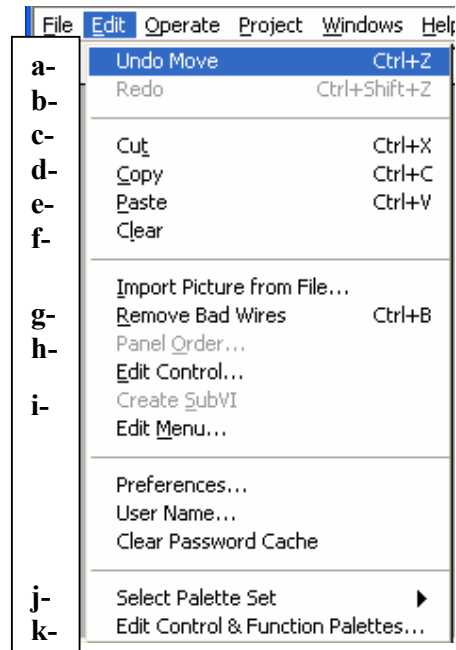


Fig. 1.7 Submeniul derulant „Editare”

Semnificația celor mai importante opțiuni din subdomeniul derulant “Editare” se prezintă în continuare:

- Se anulează efectele ultimei operații de editare (fig. 1.7, a);
- Se reface ultima operație de editare (fig. 1.7, b);
- Se copiază obiectele selectate în memoria temporară clipboard și apoi se elimină din instrumentul virtual (fig. 1.7, c);
- Se copiază obiectele selectate în memoria temporară clipboard (fig. 1.7, d);
- Conținutul memoriei temporare clipboard se inserează în fereastra curentă (fig. 1.7, e);
- Se șterg obiectele selectate (se obține același efect, dacă se apasă tasta <Delete>, fig. 1.7, f);
- Se elimină toate firele care realizează conexiuni eronate (fig. 1.7, g);
- Se permite schimbarea de către utilizator a ordinii de parcurgere (la rulare) a obiectelor din panoul frontal (fig. 1.7, h);
- Se generează pentru obiectele selectate din diagrama bloc un nou IV : pictograma asociată noului IV înlocuiește obiectele selectate (fig. 1.8, i);
- Se selectează o aranjare personalizată a elementelor, care apar în casetele cu controale și indicatoare, respectiv cu funcții și IV (fig. 1.7, j);
- Se personalizează casetele cu controale și indicatoare, respectiv cu funcții și IV (fig. 1.7, k).

În **LabVIEW** versiunea 5.0.1, se oferă implicit posibilitatea anulării efectelor ultimelor 8 operații de editare, realizate asupra oricăror IV deschise. Numărul de

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 1

anulări a efectelor acțiunilor de modificare poate varia între 0 și 99; prin stabilirea acestui număr la o valoare prea mare, se reduce dimensiunea memoriei disponibile, de lucru, firma **National Instruments** recomandând evitarea utilizării valorii 0, care poate determina blocarea sistemului. Accesul la această valoare se face prin meniul **“Edit / Preferences”**, opțiunea **“Block Diagram”**.

Mediul **LabVIEW** permite utilizatorilor crearea unor aranjări personalizate ale elementelor componente din casetele cu controale, respectiv funcții; odată personalizată, dispunerea se salvează sub un nume sugestiv (atribuit de utilizator) și se selectează ulterior pentru utilizare (**Edit / Select Palette Set**). În mod implicit sunt definite un număr de patru aranjamente:

1. elemente de bază (*basic*);
2. implicit toate (*default*);
3. elemente care vizează achiziția de date (*daq_view*);
4. elemente pentru realizarea măsurărilor (*t&m_view*).

Pentru o anumită configurație a casetei cu controale, respectiv cu funcții, se poate opta între afișarea elementelor componente sub forma unor pictograme-grafice sau text (**Edit / Select Palette Set/ Display Style**).

Lansarea editorului, care permite personalizarea casetelor cu controale și funcții, se face selectând din meniul bară opțiunea **“Edit / Edit Control & Function Palettes...”**. Pentru aranjarea existentă sau pentru crearea uneia noi (opțiunea **“new setup...”** din meniul derulant), se permite rearanjarea conținutului, ștergerea sau inserarea de noi elemente, după care se optează pentru salvarea modificărilor realizate (**Save Changes**). Accesul la operațiile de editare este dat de opțiunile meniurilor contextuale, apelate la apăsarea butonului dreapta al mouse-ului, poziționat deasupra obiectelor sau casetei.

Observație:

Dacă se utilizează **LabVIEW** versiunea 5.0.1, se va verifica dacă în caseta cu funcții sunt disponibile intrările pentru instrumentele virtuale: **“Resize Panel.vi”**, **„Release Instrument.vi”**, **„Open Panel.vi”**, **„Run Instrument.vi”**, ... și **“Write File+ (string).vi”**, **„Read File+ (string).vi”**, ... În caz contrar, se realizează două intrări noi în caseta cu funcții, pentru accesarea directă a instrumentelor virtuale destinate controlului *IV* și pentru operații cu fișiere. Se poate întâmpla ca, în urma instalării programului **LabVIEW**, fișierele bibliotecă cu *IV* să existe pe harddisk, iar caseta cu funcții să nu conțină intrări pentru toate *IV*.

După ce se pornește editorul pentru personalizarea casetei cu funcții (**“Edit / Edit Control & Function Palettes...”**), se inserează succesiv două submeniuri (din meniul contextual se alege **“Insert / Submenu...”** și se selectează **“Link to a library”**) astfel:

- Se introduce numele submeniului **“VI Control”** și se indică sursa:
...\\labview\\vi.lib\\utility\\victl.llb
- Se introduce numele submeniului **“File”** și se indică sursa:
...\\labview\\vi.lib\\utility\\file.llb
- Meniul **LabVIEW** se instalează implicit în directorul
“c:\\Program Files\\National Instruments”.

În final, se salvează modificările; se verifică utilizarea aranjamentului casetei cu funcții, care conține cele două submeniuri adăugate anterior (**Edit / Select Palette Set / Display Style**).

Atenție!

Studentilor le este permisă personalizarea casetelor cu controale și funcții doar cu acordul cadrului didactic supraveghetor!

Obținerea de informații suplimentare, referitoare la *IV* și sub*IV* apelate, se face prin opțiunea “Proiect”(Project) din meniul bară prezentat în figura 1.8.

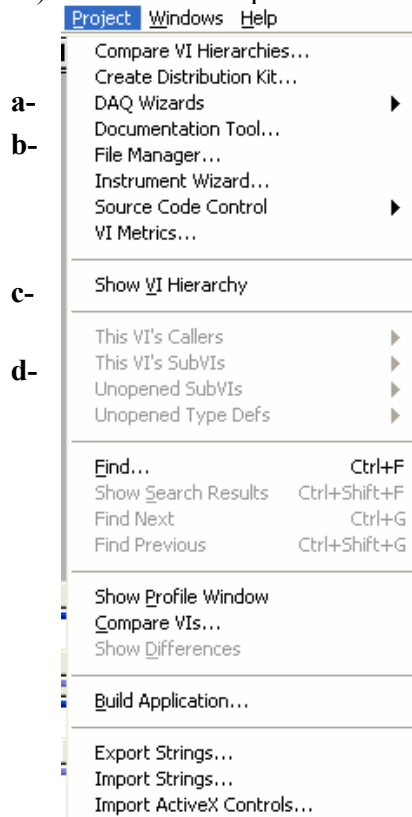


Fig. 1.8. Submeniul derulant „Proiect” cu opțiuni adiționale pentru *IV*

Semnificația opțiunilor submeniului “*Proiect*” se prezintă în continuare:

- Se lansează utilitare pentru consultarea canalelor sau selectarea unei soluții privind achiziția de date (**DAQ**, fig. 1.8, a).
- Se lansează un utilitar pentru gestionarea *IV* și a bibliotecilor **.LLB** de instrumente virtuale (fig. 1.8, b). Se oferă posibilitatea realizării unor operații de copiere, redenumire, ștergere a unui *IV* sau bibliotecă **.LLB**, creare bibliotecă **.LLB** sau director nou, expandarea într-un subdirector a conținutului bibliotecii **LLB** în fișiere fizice de instrumente virtuale, conversia conținutului unui subdirector (conținând fișiere *IV*) într-un fișier bibliotecă **LLB**.
- Se afișează arborele de apel cu *IV* pentru modulul curent (exemplul din fig. 1.10), văzut ca modul principal (fig. 1.8, c).
- Se afișează pictogramele *IV* apelate din modulul curent (fig. 1.8, d).

1.6 Prezentarea barei orizontale cu meniuri

Pentru accesarea ferestrelor deschise și a casetelor, utilizatorului i se pune la dispoziție meniul “*Ferestre*” (**Windows**), cu opțiunile prezentate în figura 1.9.

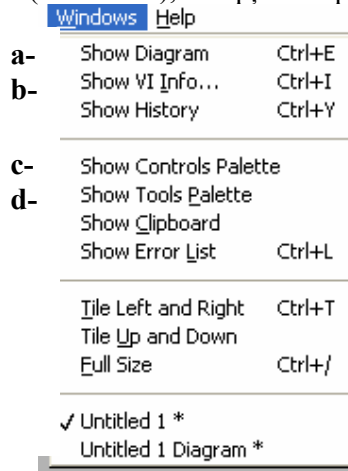


Fig. 1.9. Submeniul derulant “Ferestre “ pentru accesarea ferestrelor

Semnificația opțiunilor din submeniul derulant “*Ferestre*” se prezintă în cele ce urmează:

- Se realizează trecerea între fereastra *Panoului Frontal* și fereastra *Diagramei Bloc* (fig. 1.9, a).
- Se afișează o fereastră, în care se introduce descrierea *IV* curent (fig. 1.9, b).
- Se afișează /ascunde caseta cu controale și indicatoare din fereastra *Panoului Frontal* sau caseta cu funcții și *IV* din fereastra *Diagramei Bloc* (fig. 1.9, c).
- Se afișează / ascunde caseta cu unelte generale (fig. 1.9, d).

Obținerea de informații descriptive despre obiectele panoului frontal sau diagramei bloc a *IV*, lansarea utilitarului de asistare (**help**) sau consultarea versiunii mediului **LabVIEW** utilizat se face prin meniul de “*Asistare*” (**Help**) din bara cu meniuri.

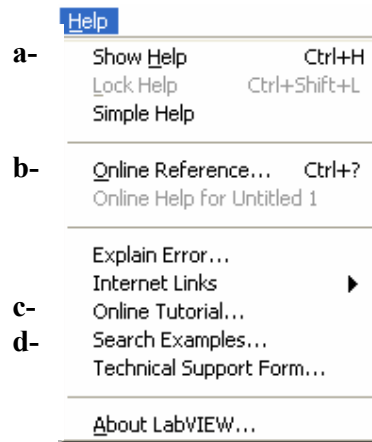


Fig. 1.10. Opțiunile de asistare ale submeniului “Asistare”

Prezentarea opțiunilor submeniului “Asistare” se prezintă în continuare :

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 2

- Se afișează/ ascunde fereastra de asistență (fig. 1.10, a)
- Se deschide utilitarul de asistență (fig. 1.10, b)
- Se lansează programul de prezentare al mediului **LabVIEW** (tutorial), care trebuie să fie instalat pe hard disc sau să existe în unitatea de CD-ROM discul original (fig. 1.10, c)
- Se lansează un utilitar, care permite utilizatorului consultarea interactivă a exemplurilor din mediul **LabVIEW**, grupate pe 5 secțiuni (fig. 1.10, d).

Studentii vor localiza în meniul bară opțiunile prezentate anterior, recitind din laborator semnificația acestora.

1.7 Prezentarea casetei cu unelte generale

Caseta cu unelte generale cuprinde instrumentele folosite de utilizator la crearea, editarea sau trasarea execuției instrumentelor virtuale. Afișarea / ascunderea casetei cu unelte generale se face prin opțiunea “ *Ferestre/ Afișează Caseta cu Unelte*” (**Windows /Show Tools Palette**) din bara cu meniuri. Caseta cu unelte generale se prezintă în fig. 1.11, în care s-a definit un sistem de identificare matricială a uneltelor.

O altă metodă de afișare a casetei cu unelte generale se bazează pe comportarea contextuală : se poziționează cursorul mouse-lui în interiorul ferestrei *Panou Frontal* sau *Diagramă Bloc* și se apasă simultan tasta *Shift* și butonul din dreapta mouse-lui.

Se exersează cele două metode de afișare a casetei cu unelte generale prezentate (din bara cu meniuri și acceleratori).

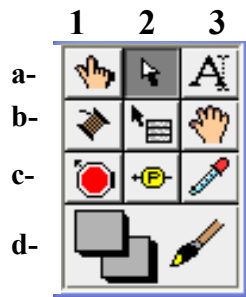


Fig. 1.11 Caseta cu unelte generale

Semnificația elementelor casetei cu unelte generale este următoarea:

- *Unealta pentru operare* (fig. 1.11, a1)
Prin intermediul uneltei se manevrează mai ales valorile elementelor panoului frontal; Se folosește mai rar în diagrama bloc, la stabilirea valorii constantelor (exemplu : constanta universală true- false, tablou de constante).
- *Unealta de editare* (fig. 1.11, a2).
Se realizează: selectare, mutare, redimensionare obiect.
- *Unealta de etichetare* (fig. 1.11, a3).
Permite introducerea de la tastatură a textului.
- *Unealta de interconectare* (fig. 1.11, b1)
Se folosește în fereastra diagramei bloc, pentru a realiza legăturile între noduri (elemente de execuție). Firele definesc fluxul datelor. În panoul

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 2

frontal se realizează corespondențele între controale / indicatoare și terminalele conectorului.

- *Unealta de defilare* (fig. 1.11, b2)
Se realizează defilarea conținutului ecranului, fără a folosi barele de defilare ale ferestrei.
- *Unealta prin care se inserează* în diagrama bloc un punct de întrerupere a execuției *IV*, pe unul dintre fire (pe un fir se acceptă un singur punct de întrerupere) sau noduri.
Se folosește în scopul depanării programului (fig. 1.11, c1).
- *Unealta sondă* (fig. 1.11, c2)
Se definește o fereastră în care se afișează valoarea transmisă pe un fir de legătură (în faza de execuție a *IV*). Se utilizează diagrama bloc, mai ales pentru depanarea programului.
- Se preia culoarea obiectului asupra căruia se execută clic cu butonul stânga al mouse-ului (fig. 1.11, c3).
- *Unealta de colorare* (fig. 1.11, d)
Se folosește mai ales la realizarea panoului frontal, permițând stabilirea culorii pentru prim planul și de fundalul obiectelor.

Se creează un *IV* nou prin opțiunea “**File/ New**” din bara cu meniuri. În continuare se rămâne în fereastra panou frontal.

Se afișează caseta cu controale și indicatoare, dacă nu este vizibilă.

Metoda 1 : Din meniul bară orizontal se alege “ **Windows/ Show Controls Palette**”.

Metoda 2 : Prin meniuri contextuale, cursorul mouse-ului poziționat în interiorul *Ferestrei Panou Frontal*, se apasă butonul stâng al mouse-ului.

Se selectează un control numeric digital (**Numeric/Digital Control**) și printr-un clic cu butonul din stânga al mouse-ului acesta va fi poziționat pe suprafața panou frontal al noului *IV*. Se tastează “*Număr 1*” pentru eticheta proprie a controlului.

Se selectează unealta de editare. Se poziționează cursorul mouse-ului (de exemplu) deasupra colțului dreapta-jos a căsuței valoare a controlului numeric digital “*Număr 1*”; se observă cum cursorul mouse-ului va lua forma unui colțar, indicând posibilitatea redimensionării obiectului. Pentru redimensionare, se ține apăsat butonul stâng al mouse-ului și se realizează mișcarea mouse-ului în direcția înainte sau înapoi, funcție de redimensionarea dorită (micșorare sau mărire); se va elibera butonul stâng al mouse-ului în momentul obținerii efectului de redimensionare dorit. În general, redimensionarea poate să se facă combinat: pe verticală și orizontală.

Se reține în fereastra panoului frontal, la fiecare apăsare a tastei <Spacebar>, se realizează comutarea între unealta de editare și cea de operare. Se exersează de câteva ori schimbarea uneltei curente prin apăsarea tastei <Spacebar>.

Se trece în fereastra diagramei bloc. Se reține că, la fiecare apăsare a tastei <Spacebar>, se realizează comutarea între unealta de editare și cea de interconectare. Se exersează de câteva ori schimbarea uneltei curente, prin apăsarea tastei <Spacebar>. Se revine în fereastra panoului frontal.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 2

La fiecare apăsare a tastei <Tab>, se parcurge circular lista uneltelor: operare-editare-etichetare-colorare. Să se tasteze această posibilitate.

Se poziționează unealta de operare deasupra căsuței valoare a controlului. Se reține modificarea aspectului cursorului mouse-ului într-o bară verticală de dimensiunea unei litere, indicând că pentru acel obiect se poate declanșa operația de editare a valorii reținute. Pentru declanșarea acțiunii de editare, se face clic stânga cu mouse-ul în interiorul căsuței valoare; cu tastele <Delete> și <Backspace>, se pot șterge cifrele afișate în control și se poate introduce de la tastatură o valoare numerică reală. Apăsând tasta <Enter> sau butonul din bara cu meniuri, pe care este afișat textul "Enter", noua valoare va fi atribuită controlului.

Se exersează utilizarea uneltelor disponibile de colorare. Se reține că prin realizarea unei interfețe încărcate de culoare se deranjează și distrage atenția operatorului uman; în același timp, se îngreunează activitatea plăcii video la fiecare reîmprospătare a ecranului.

1.8 Metode de asistare în LabVIEW

Mediul LabVIEW oferă dezvoltatorilor de programe două metode de asistență, care pot fi utilizate pe parcursul dezvoltării unui *IV* : fereastra (**Help Window**) și utilitarul (**Online Help**) de asistență.

Afișarea /ascunderea **ferestrei de asistență** se poate face prin două metode: se alege opțiunea "**Help/Show Help** " din bara cu meniuri sau apăsând concomitent *Ctrl. /H*.

Fereastra de asistență se utilizează în cazurile următoare:

- La afișarea descrierii pentru obiectele din panoul frontal și diagrama bloc ;
- La consultarea casetei cu funcții (în fereastra diagramei bloc), se prezintă semnătura și o descriere de utilizare (eventual și a parametrilor formali) a componentei, deasupra căreia se află cursorul mouse-lui ;
- În diagrama bloc, terminalul conectorului *IV*, asupra căruia se oprește unealta de interconectare, este evidențiat prin afișare intermitentă în fereastra de asistență; metoda de asistare este utilizată frecvent în realizarea conexiunilor la nodurile (funcții, sub*IV*) dotate cu mai multe terminale, pentru selectarea exactă a parametrilor așteptați;
- În diagrama bloc se permite afișarea structurii de dată, folosită pentru valoarea care se transferă între două noduri, pe o legătură existentă: se oprește unealta de interconectare deasupra legăturii; se reamintește faptul, că nodul reprezintă elementul de execuție din limbajul **G**.

Se începe un nou *IV* și se trece în diagrama bloc asociată. Se activează afișarea casetei cu funcții (**Windows / Show Functions Palette**) din bara cu meniuri și fereastra de asistență (se exersează ambele metode prezentate).

Se consultă și se rețin descrierile (afișate în fereastra de asistență) operatorilor din caseta cu funcții, grupați pe următoarele tipuri:

- *numeric* (**Numeric**);
- *boolean* (**Boolean**);
- *tablou* (**Array**);

- *relationali (Comparison).*

Utilitarul de asistență prezintă o descriere detaliată a obiectelor disponibile pentru realizarea diagramei bloc; se lansează din bara cu meniuri, dacă se selectează "**Help/Online Reference...**" sau din fereastra de asistență (fig. 1.17, c). Utilitarul de asistență din **LabVIEW** este asemănător cu a altor aplicații realizate pentru platforma **Windows**; acest lucru îl face ușor de întrebuințat.

Se lansează utilitarul de asistență prin ambele variante prezentate. Studentii se familiarizează cu posibilitățile de regăsire a informației disponibile.

2. CONTROALE ȘI INDICATOARE

2.1. Scopul lucrării

Se prezintă elementele disponibile pentru realizarea panoului frontal *IV*.

2.2. Considerații teoretice

Un program realizat în **LabVIEW** are trei părți componente:

- panoul frontal;
- diagrama bloc;
- semnătura (pictograma și conectorul).

Panoul frontal al unui *IV* reprezintă interfața cu utilizatorul. Operatorul indică de la tastatură sau mouse datele de intrare ale programului prin interacțiune asupra elementelor grafice de interfață, numite *controale*. În urma prelucrărilor datelor de intrare, rezultatele sunt afișate spre consultare prin intermediul *indicatoarelor*.

Un *IV* comunică cu alte *IV* prin intermediul controalelor (parametrii formali de intrare pentru modul) și indicatoarelor (parametrii formali de ieșire în acel moment *IV* este folosit ca subrutină (sub*IV*) .

2.2.1. Tipuri de controale și indicatoare

Sunt disponibile controale și indicatoare pentru majoritatea tipurilor de date: numeric, șir de caractere, boolean, tabel, tablou, grupare de date, grafic, ș.a. Pentru un anumit tip de date sunt disponibile mai multe variante de controale și indicatoare, utilizatorul putând să opteze pentru cea mai sugestivă realizare.

În etapa construirii panoului frontal, utilizatorul alege pe rând elemente de interfață necesare din **caseta cu controale și indicatoare** și le așează în interiorul ferestrei panou frontal. Afișarea casetei cu controale (dacă nu este vizibilă) se poate face prin două metode:

1. din bara cu meniuri se alege "*Ferestre/Afișează Caseta cu Controale*"(**Windows/ Show Controls Palette**)
sau

2. se poziționează cursorul mouse-ului în interiorul ferestrei panoului frontal și se apelează meniul contextual.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 2

Recomandare:

De cele mai multe ori se lucrează sub mediul **Windows** , spațiul de lucru pe ecran este redus (mai ales în cazul folosirii unui monitor cu diagonala de 14 inch) datorită prezenței ferestrelor auxiliare cu unelte /componente de interfață, care însoțesc majoritatea aplicațiilor cu caracter grafic. Pentru a avea o viziune cât mai cuprinzătoare asupra panoului frontal al *IV* , se recomandă inhibarea afișării ferestrelor suplimentare; majoritatea casetelor auxiliare se pot închide putând fi apelate prin meniuri contextuale (afișare temporară prin selectarea unei componente de interfață sau opțiuni asociate unei acțiuni).

Dacă cursorul mouse-ului este "*purtat*" peste caseta cu controale, se afișează automat denumirea grupului de componente în titlul ferestrei gazdă. Afișarea conținutului unui anumit grup este declanșată la realizarea unui clic stânga al mouse-lui pe pictograma corespunzătoare. După identificarea grupului, se alege varianta dorită a elementului de interfață (control / indicator); preluarea componentei din caseta cu controale se face prin clic stânga pe pictograma asociată, după care se eliberează butonul mouse-ului. Cursorul mouse-ului va lua forma unei mâini, care reține componenta selectată; plasarea componentei pe suprafața panoului frontal se va face la poziția curentă a cursorului mouse-ului, în urma unui clic stânga.

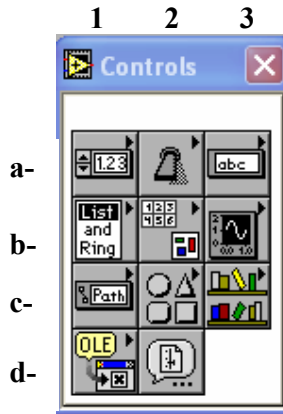


Fig.2.1 Caseta cu elemente de control si indicatoare

Semnificația elementelor din caseta cu controale și indicatoare se prezintă în continuare în legătură cu fig. 2.1, unde s-a adoptat un sistem de identificare matricial:

- componente numerice (fig. 2.1, a1);
- componente booleene (fig. 2.1, a2);
- componente șir de caractere și tabel (fig. 2.1, a3);
- liste cu opțiuni (fig. 2.1, b1): listă circulară / inel (**Text Ring**), meniuri derulante (**Menu Ring**, **Dialog Ring**), cutie listă (**Listbox**);
- tablou și grupare de date (fig. 2.1, b2);
- reprezentări grafice (fig. 2.1, b3);
- cale de fișier (fig. 2.1, c1);
- elemente decorative (fig. 2.1, c2);
- componente utilizator (fig. 2.1,c3);
- componente pentru suport **ActiveX** (fig. 2.1, d1);
- opțiuni pentru selectarea unui control stocat într-un fișier (se permite consultarea structurii de directoare, (fig.2.1, d2).

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 2

Imediat după plasarea controlului /indicatorului pe suprafața panoului frontal se recomandă indicarea unei etichete proprii (owned label), care reprezintă identificatorul obiectului în codul sursă al programului (diagrama bloc a *IV*).

Stabilirea valorilor pentru opțiunile adiționale elementului de interfață, care se pot referi la afișarea unor elemente descriptive suplimentare, selectarea comportării mecanice (controale booleene), tipul de reprezentare folosit, domeniul valorilor posibile (minim, maxim, valoare implicită) și numărul de zecimale semnificative, se face prin intermediul meniului contextual asociat controlului sau indicatorului, ilustrat în fig. 2.2.

Opțiunile aceluiași meniu contextual permit și conversia controlului în identificator sau invers, localizarea terminalului asociat în diagrama bloc (codul sursă al programului), crearea în diagrama bloc de elemente (noduri) pentru controlul proprietăților asociate, stabilirea unui comentariu care să descrie semnificația și rolul variabilei, asocierea elementului de interfață a unei combinații de taste utilizate pe post de accelerator (key navigation), folosită la rularea *IV*.

Prin intermediul nodurilor proprietate, asociate controalelor bară, indicatoarelor, programatorul modifică (acțiune transparentă utilizatorului aplicației) din codul sursă al programului (diagrama bloc) atributele fizice ale obiectelor de interfață din panoul frontal: vizibilitatea (*visible*), dezactivarea (*disabled*), poziția (x,y) în panoul frontal, culoarea ș.a.

Majoritatea opțiunilor, ilustrate în meniul contextual (fig. 2.2), se regăsesc și la celelalte tipuri de controale și indicatoare.

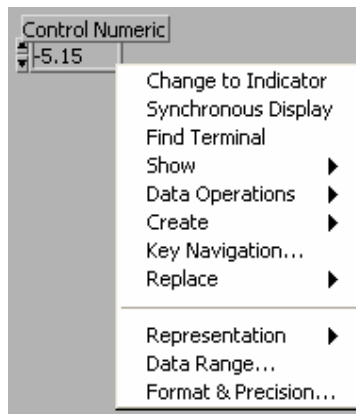


Fig.2.2 Meniu contextual asociat unui control numeric

Programatorii pot specifica pentru controalele din panoul frontal taste acceleratori, prin care acestea să fie selectate la execuția programului: se folosește opțiunea "**Key Navigation...**" din meniul contextual asociat elementului de interfață. Controlul pentru care s-a apăsât combinația de taste (la rularea *IV*) devine obiect curent de editare.

Într-un panou frontal nu pot exista două controale, care să utilizeze aceeași combinație de taste acceleratori. Pentru indicatoare nu se pot asocia acceleratori; aceștia nu sunt obiecte de interfață destinate actualizării valorilor reținute.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 2

Asocierea unei taste accelerator pentru selectarea controlului (la execuția *IV*) este utilizată mai ales pentru butoanele de comandă, elemente de interfață cu două stări (true sau false, corespunzător apăsat sau neapăsat). Prin utilizarea acceleratorului unui buton de comandă, se produce același efect obținut la apăsarea cu mouse-ul pe acel control.

Se recomandă utilizarea acceleratorilor pentru controale tip buton mai ales în aplicațiile utilizate pentru supravegherea unor fenomene, care se desfășoară rapid în timp, trebuind achiziționate valori la intervale scurte de timp. În acest caz, este important să nu apară un eveniment, a cărui tratare din partea unității centrale de prelucrare (UCP) a calculatorului (pe care se realizează monitorizarea) să ducă, de exemplu, la pierderea unor valori critice ale fenomenului analizat sau la apariția unor probleme neașteptate în funcționarea *IV*.

Un *eveniment* posibil (apărut la rularea aplicației) ar putea fi cauzat și de mișcarea mouse-ului pe suprafața panoului frontal al *IV* , ducând la ocuparea UCP cu tratarea acestei întreruperi, care ar putea fi fatală pentru analiza procesului. De aceea, pentru aplicațiile în care timpul UCR este critic, se recomandă deconectarea fizică a mouse-ului din sistemul calculatorului; interacțiunea utilizatorului cu aplicația se va face prin acceleratorii (tastatura calculatorului) asociați controalelor prezente în panourile frontale.

Fiecare control și indicator plasat în panoul frontal primește automat un număr de ordine (ordinea din panoul frontal). Primul element are numărul de ordine 0, al doilea are asociată valoarea 1 ș.a.m.d. Numărul de ordine din panoul frontal al controalelor se poate modifica interactiv prin opțiunea **"Edit/Panel Order..."** din bara cu meniuri, determinând afișarea valorii numerice curente în stânga fiecărui element.

Parcursarea controalelor (la execuția *IV*) în ordinea crescătoare a numărului de ordine din panoul frontal se face prin apăsarea tastei <Tab>; revenirea pe un control anterior se face prin apăsarea simultană a tastelor <Shift> și <Tab>.

2.2.2. Controale și indicatoare de tip numeric

Transmiterea valorilor numerice spre program sau afișarea rezultatelor numerice se face prin intermediul controalelor și indicatoarelor numerice. Există mai multe realizări ale componentelor de interfață, corespunzătoare tipului de dată numeric. Fiind activă fereastra panoului frontal, pentru consultarea colecției de componente numerice se va afișa caseta cu controale (dacă nu este vizibilă): din bara cu meniuri se alege **"Ferestre /Afișează Caseta cu Controale"** (**Windows/Show Controls Palette**), sau se apelează meniul contextual pentru obiectul fereastra panou frontal (clic dreapta); din caseta cu controale se selectează grupul **"Numeric"**, care va afișa colecția cu variantele de realizare a elementelor de interfață corespunzătoare tipului numeric, aspect redat în fig. 2.3.

Colecția de controale și indicatoare numerice reține următoarele variante de realizare:

- digital;
- cursor orientat vertical (*vertical slide*) sau orizontal (*horizontal slide*);
- rezervor (*tank*);
- termometru (*thermometer*);
- buton rotitor (*knob*);

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 2

- buton disc de telefon (*dial*);
- cadran aparat de măsură (*meter*);
- kilometraj (*gauge*).

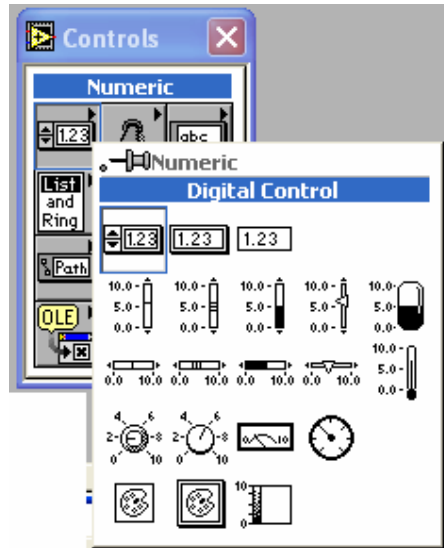


Fig. 2.3 Variante ale componentelor de interfata pentru tipul numeric de date

Controalele numerice digitale sunt preferate pentru simplitatea realizării grafice, prevenind posibile confuzii de folosire a aplicației de către operatorul uman. În general, interfețele se recomandă să fie cât mai simple și în realizarea lor să se folosească cele mai sugestive componente vizuale. Controalele și indicatoarele numerice digitale afișează rezultatele unor prelucrări matematice. Controalele și indicatoarele numerice digitale se recomandă a fi utilizate la implementarea interfeței *IV* (parametrii de intrare și de ieșire), care vor fi folosite ca și subrutine auxiliare, utilizatorul ne-interacționând cu panoul frontal al acestora.

Introducerea valorilor într-un control numeric digital se face cu unealta de operare și acționând prin una din metodele următoare:

1. Se face clic stânga deasupra chenarului controlului, pentru a plasa cursorul în câmpul de editare și se introduce noua valoare de la tastatură sau se folosesc săgețile de pe claviatură.
2. Se folosesc butoanele (dispuse pe verticală în partea stângă a câmpului de editare) de incrementare sau decrementare. În interiorul câmpului de editare se pot folosi tastele *<Delete>* și *<Backspace>*, pentru ștergerea caracterului din dreapta, respectiv stânga, cursorului.

Încheierea operației de editare se face apăsând tasta *<Enter>*, sau butonul etichetat "Enter" (afișat în stânga barei cu unelte), sau realizând un clic stânga cu mouse-ul în exteriorul controlului.

Pentru un control numeric, plasat în panoul frontal, se pot stabili suplimentar unele proprietăți prin meniul contextual asociat:

- valoarea curentă afișată este stabilită implicit (**Data Operations/ Make Current Value Default**);

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 2

- asocierea unui accelerator (**Key Navigation**);
- formatul de reprezentare (**Representation**): se poate alege unul din cele 12 tipuri disponibile, redată în fig. 2.5;
- domeniul valorilor posibile, incrementul (**Data Range...**) folosit;
- stabilirea formatului de afișare (numeric, dată calendaristică) și a numărului de zecimale utilizate în afișare (**Format & Precision...**).

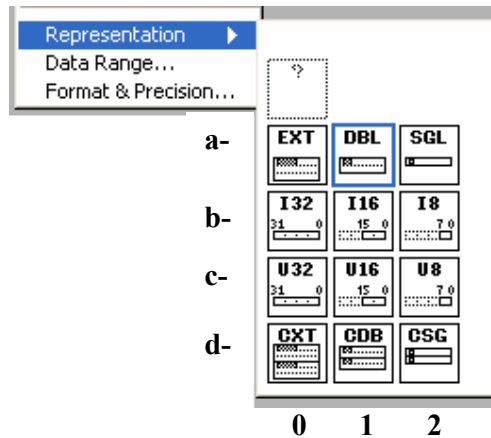


Fig. 2.4. Formate de reprezentare disponibile din meniul contextual

Semnificația formatelor de reprezentare din figura 2.4 se prezintă în continuare :

- real, precizie extinsă (96 biți, fig. 2.4, a0);
- real, dublă precizie (64 biți, fig. 2.4, a1)
- real, simplă precizie (32 biți, fig. 2.4, a2);
- întreg cu semn, dublu cuvânt (32 biți, fig. 2.4, b0);
- întreg cu semn, cuvânt (16 biți, fig. 2.4, b1);
- întreg cu semn, octet (8 biți, fig. 2.4, b2) ;
- întreg fără semn, dublu cuvânt (32 biți, fig. 2.4, c0);
- întreg fără semn, cuvânt (16 biți, fig. 2.4, c1);
- întreg fără semn, octet (8 biți, fig. 2.4, c2);
- precizie extinsă numere complexe (96 biți, fig. 2.4, d0);
- dublă precizie (64 biți) numere complexe (fig. 2.4, d1);
- simplă precizie (32 biți), numere complexe (fig. 2.4, d2).

Numărul de zecimale stabilite pentru un control numeric este folosit la afișarea valorii numerice și nu influențează precizia cu care se operează; precizia depinde de formatul de reprezentare selectat.

În continuare se prezintă componentele din colecția tipului numeric cu cadran grafic.

Componentele de interfață din colecția tipului de dată numeric posedă, pe lângă afișajul numeric digital și un cadran grafic, specific variantei :

- cursor orientat vertical (*vertical slide*) sau orizontal (*horizontal slide*);
- rezervor (*tank*);
- termometru (*thermometer*);
- buton rotitor (*knob*);
- buton disc de telefon (*dial*);

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 2

- cadran aparat de măsură (*meter*);
- kilometraj (*gauge*).

Cadranul grafic este în permanență vizibil pentru o componentă, vizibilitatea afișajului numeric digital fiind opțională. Prezentare valorii reținute de componenta de interfață prin afișajul numeric digital se face prin opțiunea "**Show / Digital Display**", din meniul contextual asociat.

Componenta de interfață poate fi schimbată din control în indicator și invers, prin opțiunea "*Change to Indicator*", respectiv "**Change to Control**", din meniul contextual.

Introducerea (la editarea sau rularea *IV*) valorii numerice pentru un control se face cu unealta de operare prin:

- afișajul numeric digital
- sau
- prin mișcarea acului indicator (cu mouse-ul).

Pentru o componentă cu cadran grafic se pot stabili, ca și la cele digitale, următoarele: accelerator, format reprezentare, domeniul valorilor permise, numărul de zecimale. Prin meniul contextual se pot stabili în plus opțiuni referitoare la formatul scalei (**Scale**) și înlocuirea valorilor numerice de pe scală cu etichete sugestive (**Text Scale**), ca în fig. 2.5.

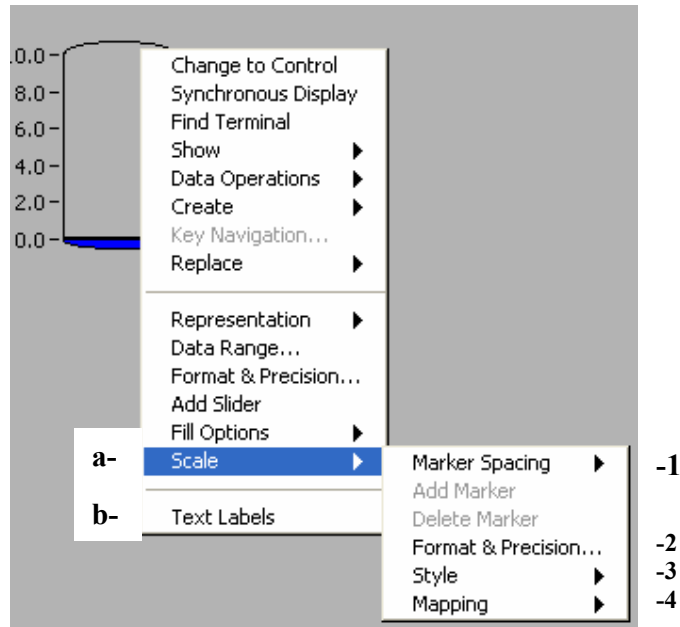


Fig. 2.5. Opțiuni pentru formatul scalei și etichete la controale/ indicatoare cu cadran grafic

Semnificațiile opțiunilor din fig. 2.5 sunt următoarele:

- opțiuni pentru distribuția (*Uniformă/Arbitrară*) marcajelor (fig. 2.5, a1);
- stabilirea numărului de zecimale și reprezentarea (fig. 2.5, a2);
- opțiuni pentru afișarea marcajelor și valorilor (fig. 2.5, a3);
- tipul de scală (*Linear/Logaritmă*, fig. 2.5, a4);

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 2

- valorile de pe scală sunt înlocuite prin etichete sugestive (fig. 2.5, b).

Pe scala unui control /indicator cu cadran grafic se pot defini (în faza de editare a *IV*) etichete sugestive (fig. 2.6) prin selectarea opțiunii "*Etichete Text*" (**Text Labels**), a meniului contextual din fig. 2.6, b. O etichetă este un șir de caractere (indicat de utilizator), căreia îi este asociată automat (de mediul **LabVIEW**) o valoare întreagă pozitivă. După selectarea acestei opțiuni sunt afișate automat două etichete, indicând o valoare minimă (*min*) și maximă (*max*), iar cursorul mouse-ului este poziționat în caseta de editare; după introducerea de la tastatură a unei etichete se apasă concomitent <Shift> și <Enter>, pentru a trece la următoarea. Încheierea procesului de introducere a etichetelor se face:

- apăsând tasta <Enter> din partea dreaptă a tastaturii (zona tastelor numerice)

sau

- selectând cu mouse-ul butonul de comandă <Enter> din bara cu unelte (afișat în partea stângă).

Prima etichetă de pe scală are asociată valoarea 0; ultimei etichete îi corespunde o valoare dată de numărul de etichete minus valoare 1.

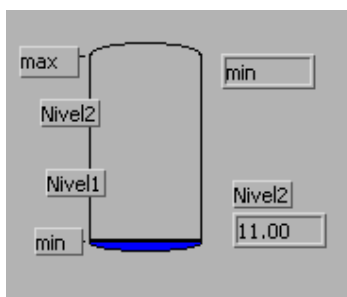


Fig. 2.6. Un control numeric tip rezervor pentru care s-au definit etichete pe scală

2.2.3 Controale și indicatoare de tip boolean

Componentele de interfață ale panoului frontal, aparținând tipului boolean au două valori posibile, corespunzând stării de adevărat (**True**) sau fals (**False**). Valoarea logică fals este valoarea inițială: componenta booleană a fost preluată din caseta de controale/indicatoare și plasată pe suprafața panoului frontal.

Există mai multe realizări grafice de controale de tip boolean: butoane, care pot fi apăstate, comutatoare, întrerupătoare, butoane radio. Indicatoarele booleene sunt reprezentări vizuale ale valorii logice și simulează leduri, realizate sub diferite forme (rotunde, pătrate).

Colecția componentelor de interfață, aparținând tipului boolean sunt disponibile din caseta de controale, grupate sub titlul **Boolean**.(fig. 2.7).

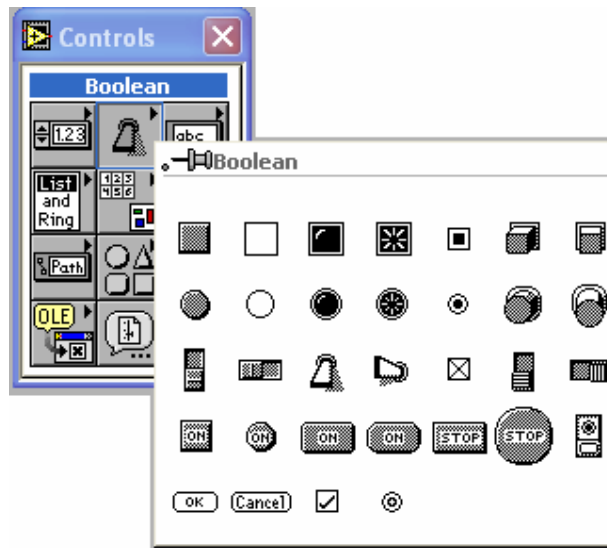


Fig. 2.7. Controale/indicatoare aparținând tipului boolean

În etapa construirii panoului frontal, se poate verifica comportarea unui control sau indicator boolean, dacă se selectează unealta și se acționează cu butonul stânga al mouse-ului; la fiecare apăsare a mouse-ului, componenta de interfață de tip boolean va trece în stare complementară.

Prin intermediul opțiunilor meniului contextual, asociat obiectului de tip boolean, se stabilesc valorile unor proprietăți suplimentare sau se realizează acțiuni specifice:

- schimbarea în control/indicator;
- afișarea/ascunderea etichetei proprii (*owned label*) și a etichetei corespunzătoare stării logice (*boolean text*) pe suprafața componentei;
- stabilirea valorii implicite;
- crearea unui nod proprietate în diagrama bloc;
- stabilirea comportării mecanice (doar pentru controale).

Opțiunile meniului contextual sunt ilustrate în figura 2.8.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 3

Fiecare control/indicator boolean poate fi redimensionat prin unealta de editare și colorat prin unealta de colorare.

Pentru fiecare control sau indicator din panoul frontal, se recomandă indicarea unei etichete proprii, care este identificatorul (numele) asociat variabilei și folosit în diagrama bloc (codul programului). De cele mai multe ori, nu este necesară afișarea acestuia în panoul frontal (putând deruta utilizatorul final al aplicației), ascunderea făcându-se prin meniul contextual, opțiunea „Show/Label” (neselectată).

Majoritatea controalelor/indicatoarelor se găsesc în starea corespunzătoare valorii logice fals, după ce au fost preluate din casta controale și așezate pe suprafața panoului frontal.

Unele componente de interfață afișează unul din cuvintele următoare:

- *închide (off)*, *anulează (cancel)* sau *oprește (stop)*, pentru starea logică fals;
- *deschide (on)* sau *în-regulă (ok)*, pentru starea logică adevărat, pe suprafața grafică, evidențiind valoarea logică corespunzătoare stării curente.

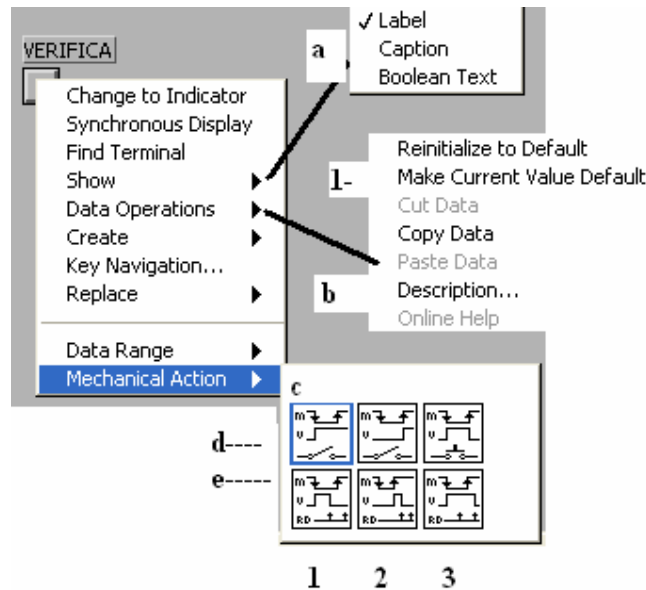


Fig. 2.8. Meniul contextual asociat unei componente booleene

Eticheta corespunzătoare stării poate fi modificată prin unealta etichetare; pentru stabilirea etichetei celeilalte stări, se trece mai întâi componenta în starea logică complementară (cu unealta de operare).

O metodă de a verifica starea curentă a unui control/indicator boolean, care nu afișează implicit eticheta stării este de a o vizualiza prin opțiunea „Show/ Boolean Text” din meniul contextual (fig. 2.8.a).

Semnificația opțiunilor din meniul contextual asociat unei componente booleene (fig. 2.8) se prezintă în continuare:

- se vizualizează informații suplimentare (eticheta proprie și eticheta stării, fig.2.8.a);
- submeniul „Acțiuni asupra datei” (**Data Operations**, fig. 2.8.b);
- se stabilește valoarea implicită (fig.2.8.b1);

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 3

- submeniul pentru stabilirea comportamentului mecanic al controlului boolean (fig. 2.8.c);
- comportare mecanică „*comută la apăsare*” (**switch when pressed**, fig.2.8.d1);
- comportare mecanică „*comută la eliberare*” (**switch when released**, fig.2.8.d2);
- comportare mecanică „*comută înainte de eliberare*” (**switch until released**, fig.2.8.d3);
- comportare mecanică „*zăvorește la apăsare*” (**latch when pressed**, fig.2.8.e1);
- comportare mecanică „*zăvorește la eliberare*” (**latch when released**, fig.2.8.e2);
- comportare mecanică „*zăvorește înainte de eliberare*” (**latch until released**, fig.2.8.e3).

În momentul selectării opțiunii „ Stabilește Valoarea Curentă Implicită”(**Make Current Value Default**, fig. 2.8.b1) se va stabili pentru componenta selectată starea/valoarea-logică curentă ca stare/valoare implicită.

Pictograma asociată comportării mecanice evidențiază grafic, în timp, succesiunea operațiilor: acțiunea utilizatorului cu mouse-ul asupra controlului (litera **M**), noua valoare a controlului (litera **V**), momentul preluării valorii controlului de către diagrama bloc (literele **RD**).

Semnificația comportării mecanice pentru controalele booleene este după cum urmează:

- „*Comută la apăsare*” (fig.2.8.d1)
La fiecare apăsare a butonului mouse-ului se modifică starea/valoarea controlului(comportare similară întrerupătorului de lumină). Valoarea controlului nu este afectată de preluarea valorii în diagrama bloc.
- „*Comută la eliberare*”(fig.2.8.d2)
Valoarea controlului se modifică după eliberarea butonului mouse-ului. Controlul rămâne în noua stare până la următoarea interacțiune a utilizatorului. Valoarea controlului nu este afectată de preluarea valorii de diagrama bloc.
- „*Comută înainte de eliberare*”(fig.2.8.d3)
Valoarea controlului se modifică la apăsare și se reține până la eliberarea butonului mouse-ului; la eliberare, controlul revine la valoarea/starea anterioară interacțiunii. Valoarea controlului nu este afectată de preluarea valorii de diagrama bloc.
- „*Zăvorește la apăsare*”(fig.2.8.e1)
La apăsare, se modifică valoarea; noua valoare este reținută până când este preluată de diagrama bloc. După preluarea valorii, controlul revine la valoarea anterioară interacțiunii.
- „*Zăvorește la eliberare*”(fig.2.8.e2)

Valoarea controlului se modifică după eliberarea butonului mouse-ului. După preluarea valorii de diagrama bloc, controlul revine la valoarea anterioară a interacțiunii.

- „Zăvorește înainte de eliberare”(fig.2.8.e3)
Valoarea controlului se modifică la apăsare. Controlul revine la valoarea anterioară, dacă survine unul din elementele:
 - diagrama bloc a preluat valoarea
 - sau
 - s-a eliberat butonul mouse-ului.

Se recomandă comportare mecanică „zăvorește la apăsare” controalelor booleene, care să declanșeze o singură dată o operație, la fiecare apăsare (exemplu: tipărirea unor date, trecerea în alt ecran, părăsirea ecranului curent).

Pentru un control boolean, pentru care s-a indicat o comportare mecanică din clasa „zăvorean”, nu se mai pot defini variabile locale în diagrama bloc. Prin variabilă locală se poate înțelege utilizarea sub alt nume a variabilei, cu care este pusă în corespondență; în limbajul Pascal sau C, ar echivala cu o variabilă de tip pointer, asociate unei variabile. Fiecărui element de interfață din panoul frontal îi corespunde o variabilă terminal în diagrama bloc; prin variabila locală se adaugă în diagrama bloc încă o variabilă-terminal, prin care programatorul are acces la valoarea controlului/indicatorului părinte, pentru inspectare sau actualizare.

2.2.4. Controale și indicatoare tip șir de caractere

Componentele de interfață tip șir de caractere pentru citirea/afișarea mărimilor aparținând acestui tip de dată.

În etapa construirii panoului frontal, se introduce o valoare pentru un control/indicator șir de caractere prin unealta de etichetare sau de operare; valoarea actualizată este asociată componentei în urma apăsării butonului <Enter>, afișat în bara orizontală cu meniuri sau a tastei <Enter> din partea dreaptă a tastaturii (zona tastelor numerice). Dacă se apasă tasta <Enter> atunci se va forța trecerea la un rând nou; valoarea (textul) afișat într-o componentă șir de caractere se poate întinde pe mai multe linii (lățimea cadrului). Când se editează conținutul componentei și cursorul ajunge în dreptul limitei dreapta a spațiului de editare a controlului/indicatorului, se trece automat la o linie nouă; la un moment dat este vizibilă o singură linie alfanumerică.

Un control/indicator șir de caractere poate fi redimensionat prin unealta de editare. Dacă valoarea unui control/indicator se întinde pe mai multe linii, se poate afișa bara de defilare (scroll bar) a conținutului, prin opțiunea meniului contextual „Afișează / Bara derulare” (**Show / Scrollbar**). Opțiunea este disponibilă atât pentru controale cât și pentru indicatoare; pentru a avea succes la această opțiune va trebuie anterior, mărită suficient dimensiunea verticală (înălțimea finală să fie minim triplul înălțimii unei singure linii de text) a componentei (cu unealta de editare) pentru a asigura și afișarea elementului de control bară de derulare. Dacă nu se mărește suficient înălțimea componentei, atunci opțiunea din meniul contextual nu va fi disponibilă.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 3

Valoarea componentei se poate declara valoare-implicită dacă se selectează din meniul contextual opțiunea „Acțiuni asupra datei / Stabilește Valoarea Curentă Implicită” (**Data Operations / Make Current Value Default**).

Componenta selectată poate fi colorată prin unealta de colorare. Pentru caracterele utilizate la afișare se poate indica suplimentar: utilizarea unui anumit tip de caractere (font); dimensiunea și colorarea caracterelor, încadrarea orizontală a conținutului alfanumeric (stânga, centrat, dreapta), opțiuni disponibile prin intermediul barei cu unelte.

Meniul contextual, asociat unui control tip șir de caractere (etichetat „Parola”), este prezentat în figura 2.9, prin care utilizatorului aplicației i se cere introducerea unei parole de acces; verificarea corectitudinii parolei introduse se face în urma apăsării butonului de control, etichetat „VERIFICA” pentru care s-a definit tasta accelerator <F7>

Caracterele introduse în controlul „Parola” sunt protejate, fiind afișate prin caractere „*”. Conținutul controlului este restricționat la o singură linie, care însă poate depăși lățimea vizibilă componentei; în figura 2.10, b opțiunea „Limitează la o singură linie” este selectată – utilizatorul neputând introduce caracterul linie nouă prin apăsarea tastei <Enter> în conținutul șirului de caractere.

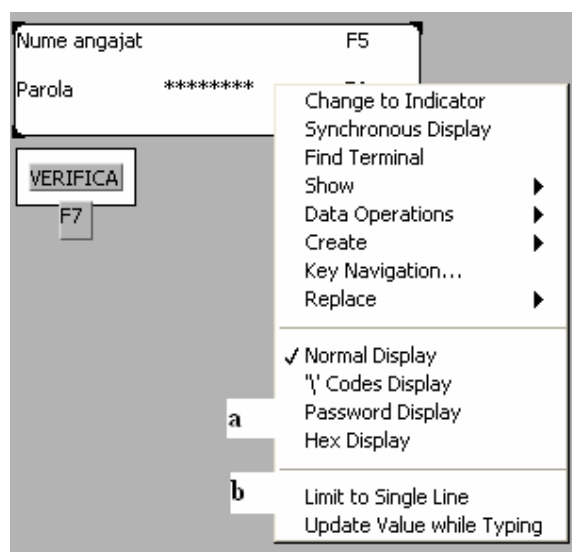


Fig.2.10. Meniul contextual al unui control șir de caractere:
a – opțiunea pentru protejarea afișării șirului de caractere;
b – opțiunea selectată interzice introducerea caracterului linie nouă la apăsare tastei <Enter>

O altă componentă de interfață care facilitează introducerea/afișarea datelor tip șir de caractere este tabelul, redat în figura 2.10.

Ca structură de date, un tabel este un tablou bidimensional (matrice), cu elemente (celule) șir de caractere. Elementele sunt grupate pe linii, putând exista mai multe

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 3

linii. O anumită celulă a tabelului se va găsi la intersecția dintre linia și coloana corespunzătoare.

Un tabel poate fi control (permite actualizarea datelor conținute în celule) sau indicator (afișarea valorilor nu permite actualizarea datelor).

Pentru o componentă de interfață tabel, se permite accesul prin intermediul meniului contextual, asociat la eticheta proprie, afișarea indexului liniei și coloanei, afișarea barei verticală/orizontală de derulare, afișarea titlului pentru fiecare linie (Row Headers) și a capului de tabel (Column Headers) și definirea unei taste accelerator. Redimensionarea tabelului și a celulelor se face prin unealta de editare.

Numărul de ordine pentru linii/coloane începe de la valoarea zero.

Introducerea valorilor în etapa construirii panoului frontal, într-o componentă tabel (pentru capul de tabel, titlul de linie, celule), se face prin unealta de operare sau de etichetare.

Pentru a se ajunge la o anumită celulă (la execuția IV), se indică numărul liniei și coloanei prin controalele din figura 2.10, a și b. Componentele index de coloană și linie indică poziția în tabel pentru celula afișată curent în colțul stânga sus a zonei, în care este prezentat conținutul tabelului.

Fig.2.10. Componenta de interfață tip tabel:

a-index coloană; b-index linie; c-titlul liniilor; d-capul de tabel; e-celulele tabelului.

2.2.5. Controale și indicatoare tip tablou și grupare de date

Tabloul (*Array*) este o structură de date compusă, cu elemente aparținând aceluiași tip: numeric, boolean, șir de caractere, cale de fișier (*path*), identificator unic (*refnum*), grupare de date (*cluster*). Se spune că tabloul este o structură omogenă de date. Elementele dețin în cadrul structurii compuse o poziție bine determinată; accesul la o anumită celulă se face prin indexare.

Pentru un tablou cu **N**-elemente, primul element se găsește la poziția 0, iar ultimul element din tablou la poziția **N-1**. Un tablou poate avea mai multe dimensiuni; numărul maxim de elemente/dimensiune este $2^{31}-1$. Dimensiunea unui tablou este supusă dimensiunii memoriei disponibile. Pentru fiecare dimensiune a unui tablou se folosește un index.

În limbajul **G**, elementele unui tablou nu pot fi la rândul lor tablouri.;se pot defini tablouri cu mai multe dimensiuni. Elementele unui tablou pot fi de tip grupare de date, care să conțină elemente tip tablou, cu elemente de tip grupare de date ș.a.

Programarea în mediul **LabVIEW** încurajează folosirea structurii de date tablou, prin puternice funcții puse la dispoziția dezvoltatorului de IV.

Pentru crearea în panoul frontal a unui control/indicator de tip tablou, se selectează din caseta cu controale, grupul „*Tablou & Grupare de Date*” (**Array & Cluster**), un obiect „*Tablou*” (**Array**) așa cum se indică în figura 2.11.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 3

După plasarea obiectului de interfață tip tablou la locul dorit, pe suprafața panoului frontal, se va indica eticheta proprie a variabilei (identificatorul variabilei tip tablou); acest lucru este semnalizat utilizatorului prin prezența unui dreptunghi înnegrit, de dimensiunile unui caracter, plasat în partea superioară, stângă a chenarului componentei.

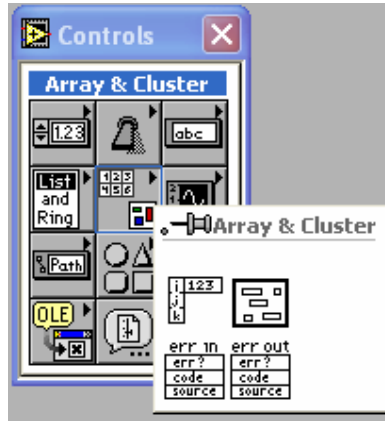


Fig. 2.11. Localizarea variabilei tip Tablou în caseta cu controale

S-a definit o variabilă tip tablou, pentru care urmează să se indice tipul elementelor. Numai variabila tip tablou singură nu poate fi folosită, neavând asociată vreo semnificație de control sau indicator; se spune despre o variabilă tip tablou că este indicator sau control după tipul elementelor. Este logic să se deducă concluzia: elementele unui tablou sunt toate controale sau toate indicatoare.

Dintre tipurile de date permise pentru elementele tabloului se alege o componentă din caseta cu controale și indicatoare; componenta selectată se poziționează cu mouse-ul deasupra variabilei tip Tabou . Interiorul chenarului variabilei tip Tablou va fi afișat hașurat, indicând faptul că pentru variabila tablou se va indica tipul elementelor; printr-un clic stânga cu mouse-ul, se încheie definirea tipului elementelor tabloului.

Pentru componenta tablou sunt afișate două căsuțe: **căsuța indicelui**, din partea stângă și **căsuța valoare**, în partea dreaptă; într-o căsuță valoare se afișează valoarea elementului tabloului de pe poziția prezentată în căsuța indice. Cu unealta de editare se poate mări dimensiunea căsuței valoare: se trage de colțul din dreapta-jos al căsuței pe direcția orizontală, înspre dreapta sau pe direcția verticală, în jos. Dacă se mărește dimensiunea căsuței valoare, se afișează simultan mai multe elemente ale tabloului: dispunerea este pe orizontală sau pe verticală. Pentru modificarea orientării elementelor tabloului trebuie să se reducă numărul de celule afișate la una singură, după care să se redimensioneze căsuța valoare în direcția dorită. Celulele elementelor, al căror conținut este umbrit, indică faptul că nu sunt definite.

Variabila tablou definită anterior are o singură dimensiune. Stabilirea dimensiunilor suplimentare pentru un tablou se face astfel:

- se mărește (pe direcția verticală în jos) cu unealta de editare căsuța indicelui

sau

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 3

- se selectează opțiunea „Adaugă o dimensiune” (**Add Dimension**) din meniul contextual, asociat variabilei tablou.

Reducerea dimensiunilor unui tablou se realizează:

- micșorând (pe direcția verticală în sus), cu unealta de editare, căsuța indicelui

sau

- selectând opțiunea „Elimină o Dimensiune” (**Remove Dimension**) din meniul contextual asociat variabilei tablou

Opțiunea „Elimină o Dimensiune” din meniul contextual asociat variabilei tablou este accesibilă doar dacă tabloul are minim 2 dimensiuni.

Pentru un tablou cu mai multe dimensiuni, căsuțele indice sunt dispuse pe verticală.

Pentru un tablou cu mai multe dimensiuni (matrice), se poate mări zona valorilor atât pe orizontală, cât și pe verticală: se obține o afișare matricială a elementelor. Elementul afișat în celula stânga sus a matricei de vizualizare a valorilor se găsește în tablou, la poziția indicată de valorile afișate ale indicilor.

Redimensionarea căsuței valoare a componentei elementului de tablou este imposibilă, după ce a fost plasată în interiorul chenarului variabilei tablou; pentru redimensionare (vizualizarea mai multor cifre) se procedează astfel: componenta se scoate în exteriorul variabilei tablou, se aduce cu unealta de editare (**Positioning Tool**) la dimensiunea dorită, iar în final se plasează în interiorul chenarului componentei tablou.

O grupare de date (Cluster) este o structură compusă de date. Nu este obligatoriu ca elementele componente să aparțină aceluiași tip, spre deosebire de limitarea apartenenței unice a tipului la elementele unui tablou (**Array**). Se spune că *gruparea de date* este o structură eterogenă. Structura *grupare de date* din limbajul **G** este implementată în limbajul **C** prin tipul structură (struct), iar în Pascal prin tipul înregistrare (**record**).

Elementele componente ale unei grupări de date pot aparține oricărui tip valid de dată, din limbajul grafic **G**.

2.2.6. Controale și indicatoare pentru reprezentări grafice

Prin intermediul acestor componente de interfață, plasate în panoul frontal, se permite vizualizarea reprezentărilor grafice.

Indicatoarele primesc valorile numerice, necesare realizării reprezentării grafice, în urma prelucrărilor din diagrama bloc.

Controalele se utilizează în locul indicatoarelor, dacă valorile necesare realizării reprezentării grafice sunt primite ca parametri de intrare pentru IV curent, folosit drept subrutină (subIV). Nu se permite utilizatorului să realizeze o reprezentare grafică interacționând direct asupra unui control.

Componentele de interfață, dedicate reprezentărilor grafice, se împart în doua categorii generale:

- diagramă (**chart**);
- grafic (**graph**).

Diagramele sunt reprezentări grafice ale unei mărimi, care se modifică pe o perioadă de timp.

Exemplu:Diagrama evoluției temperaturii pentru o durată specificată de timp.

Graficele reprezintă variația a două mărimi: y în funcție de x .

Diagrama, spre deosebire de Grafic, păstrează punctele reprezentate anterior, permițând vizualizarea variației în timp a procesului; noile puncte sunt afișate în continuarea celor vechi. Reprezentarea punctelor curente în Grafic se face după ștergerea reprezentării anterioare.

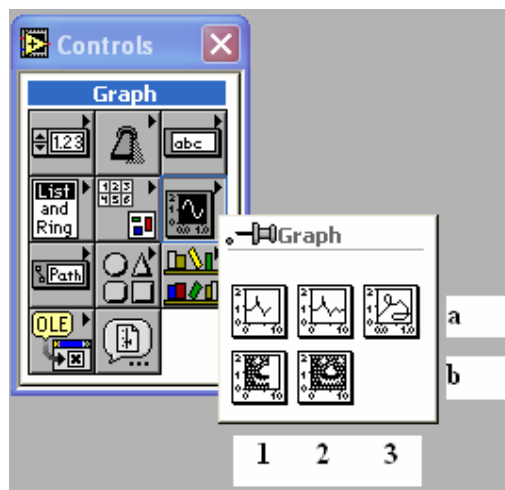


Fig. 2.12. Componente de interfață pentru reprezentări grafice:
a1–diagramă undă; a2-grafic undă; a3-grafic XY;
b1-diagramă intensitate; b2-grafic intensitate.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 4

Sunt disponibile 5 tipuri de componente de interfață pentru vizualizarea reprezentării grafice:

1. diagrame undă (**waveform charts**);
2. grafice undă (**waveform graphs**);
3. grafice XY (**XY graphs**);
4. diagrame intensitate (**intensity charts**);
5. grafice intensitate (**intensity graphs**).

Componentele dedicate vizualizării reprezentărilor grafice se găsesc în casta cu controale și indicatoare, grupul „*Grafice*” (**Graphs**), conform figurii 2.12.

1. Diagrama undă

Este o componentă de interfață, dedicată afișării uneia sau mai multor reprezentări grafice simultan, pentru care se urmărește variația în timp.

În figura 2.13 se indică elementele suplimentare de interfață, asociate unei componente tip diagramă undă, în care se reprezintă grafic variația unei mărimi în timp. Se permite redimensionarea componentei prin unealta de editare.

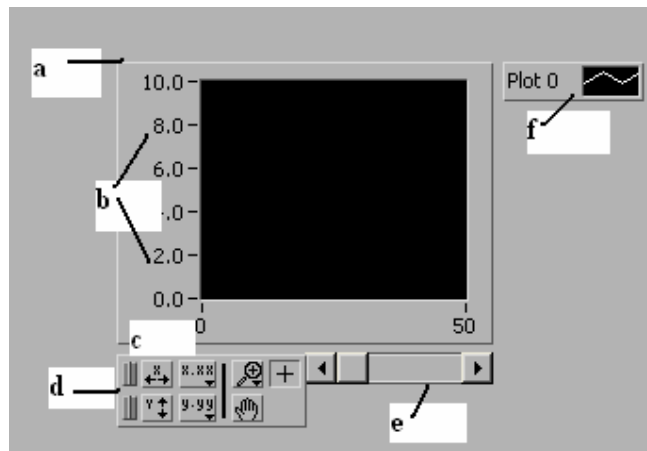


Fig.2.13. Elementele componenteii Diagramă undă:
a-identificatorul; b-axa ordonatelor (mărimia); c-axa absciselor (numărul de puncte);d-caseta cu unelte grafice; e-bara de defilare; f-legenda.

Caseta cu unelte grafice reține elementele, care se pot folosi de către utilizator în faza de execuție a IV, prin care se realizează operații de scalare și de stabilire a unor opțiuni asociate reprezentării. Bara de defilare permite inspectarea reprezentării punctelor afișate anterior. Legenda permite descrierea fiecărei reprezentări, stabilirea culorii, indicarea punctelor pe reprezentare, stilul de marcare al punctelor, tipul liniei utilizate ș.a. Afișajul numeric indică ultima valoare reprezentată grafic.

Afișarea/ascunderea elementelor adiționale de interfață asociate componenteii se face prin meniul contextual asociat, fapt redat în figura 2.14.

Pot fi reprezentate mai multe variații în timp, ca în figura 2.15, folosind în comun axa ordonatelor. Prin selectarea opțiunii „*Reprezentări stivuite*” (**Stack Plots**, fig. 2.15, h), reprezentările sunt afișate pe verticală, una sub alta (fig. 2.15, b); fiecare

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 4

reprezentare posedă propria axă a ordonatelor, utilizatorul putând stabili domenii diferite. Se revine selectând opțiunea „*Reprezentări Suprapuse*”(Overlay Plots).

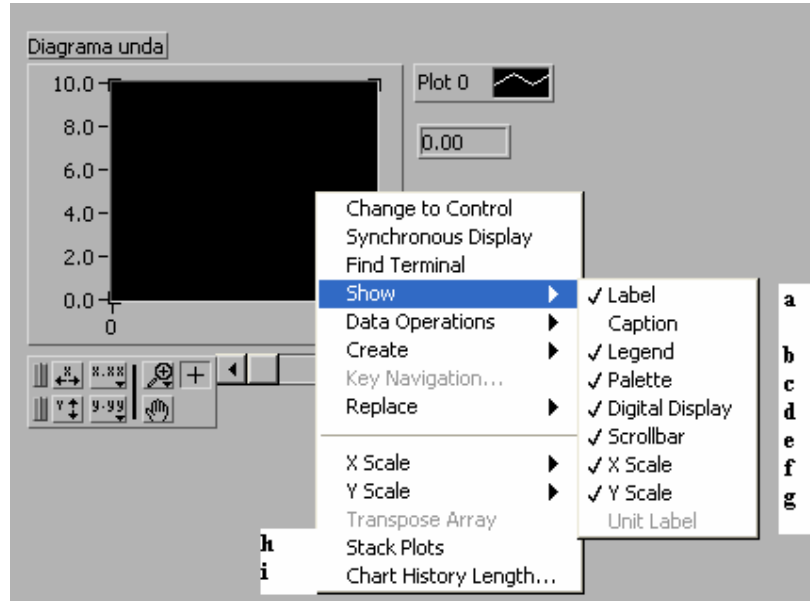


Fig.2.14. Meniul contextual asociat componentei tip Diagramă undă:
a-identificatorul asociat; b-legendă; c-caseta cu unelte grafice; d-afișaj numeric;
e-bara de defilare; f-abscisa; g-ordonata; h-afișarea una sub alta, pe verticală, a
reprezentărilor; i-stabilirea numărului de puncte de reținut

Pentru stabilirea limitelor domeniului pentru una din abscise (în cazul reprezentării Diagramei undă este vorba doar de ordonate), se selectează unealta de etichetare; se face clic stânga cu mouse-ul deasupra etichetei numerice, de pe scală, care se dorește a se modifica. După vizualizarea cursorului de editare, se editează cu tastatura valoarea afișată și se finalizează apăsând tasta <Enter> sau selectând cu mouse-ul butonul din bara orizontală cu unelte, pe care este afișat textul „Enter”.

Punctele curente (noi) de afișat sunt reprezentate în continuarea celor existente; numărul punctelor reținute și reprezentate anterior este stabilit prin opțiunea „*Dimensiunea Istorie a Diagramei*” (**Chart History Length**, fig. 2.14, i). Valorile punctelor reprezentate anterior sunt reținute într-o memorie temporară (*buffer*), a cărei dimensiune este stabilită prin această opțiune; în momentul atingerii limitei maxime, se renunță la vechile puncte reținute. Inspectarea punctelor anterioare, reținute în memoria temporară, se face prin bara de defilare orizontală („se merge” înapoi la axa timpului).

În cazul reprezentărilor multiple (în aceeași componentă), se pot stabili pentru fiecare reprezentare în parte eticheta descriptivă, caracteristicile stilului liniei, culoarea, tipul punctelor ș.a. Accesul la aceste opțiuni se face prin meniul contextual, asociat fiecărei etichete indicate în *Legendă*, ca în figura 2.16.

Dacă elementul *Legendă* nu este vizibil, acesta se va selecta pentru afișarea opțiunii, din meniul contextual asociat componentei tip Diagramă undă (fig. 2.14. b). Implicit, *Legendă* afișează doar intrarea pentru o singură reprezentare; pentru prezentarea descrierii tuturor reprezentărilor, se trage cu unealta de editare (pe direcția în jos)

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 4

colțul inferior (stâng sau drept) al casetei *Legendei*. Cu unealta de etichetare se poate modifica descrierea asociată fiecărei reprezentări. Se poziționează unealta de editare deasupra descrierii unei reprezentări și se apelează meniul contextual, prin care se stabilesc opțiunile folosite la realizarea fiecărei reprezentări în parte. (fig. 2.16).

O parte din elementele și opțiunile prezentate și asociate reprezentării Diagramă undă se regăsesc și la celelalte 4 tipuri de componente.

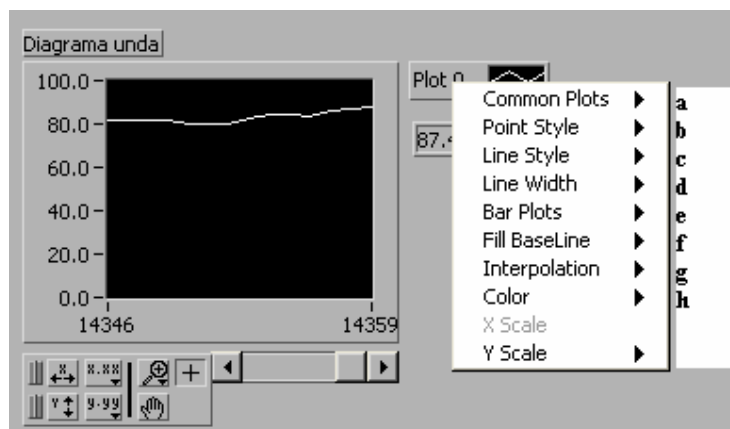


Fig. 2.16. meniul contextual asociat Legendei unei reprezentări:
a-selectare tip reprezentare (6 tipuri); b-tipuri de marcarea puncte; c-tipuri de linii; d-grosimea liniei; e-utilizare bare; f-tipuri de umplere a zonelor față de axe; g-tip trasare linie între puncte; h-culoare reprezentare

Pentru o componentă tip Diagramă undă, se poate stabili suplimentar modalitatea de înprospătare a datelor: opțiunea „Mod de Actualizare” (**Update Mode**), submeniul „Acțiuni asupra datei” (**Data Operations**), din meniul contextual asociat, redat în figura 2.17.

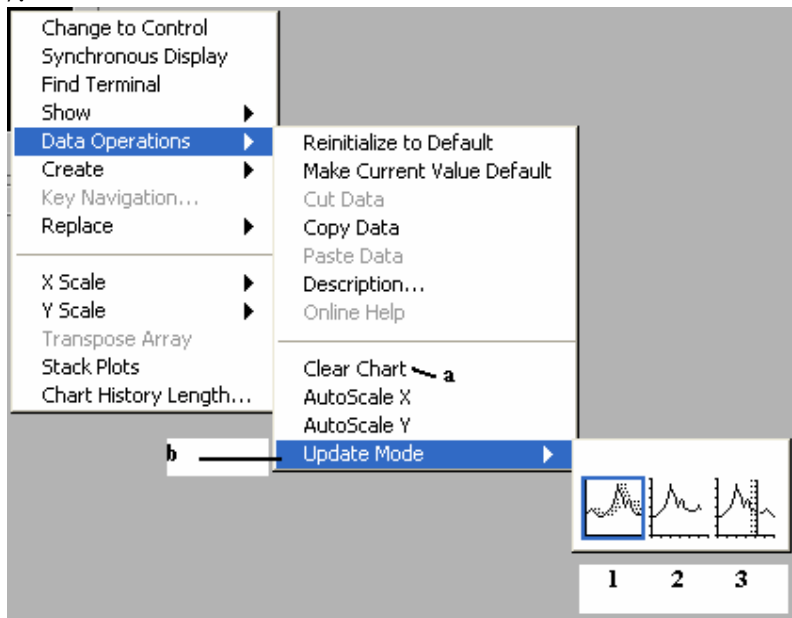


Fig.2.17. Meniul contextual cu opțiuni asupra datelor:

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 4

a-ștergerea reprezentărilor; b1-diagramă tip panglică;
b2-diagramă tip domeniu; b3-diagramă tip baleiere

Comportarea celor trei moduri de actualizare a noilor puncte într-o reprezentare diagramă undă este după cum urmează:

- *Diagrama tip panglică (strip chart, fig. 2.17, b1):*
Noile puncte se afișează în partea dreaptă a celor existente, în timp ce vechile puncte sunt retrase spre stânga.
- *Diagrama tip domeniu (scope chart, fig. 2.17, b2):*
Noile puncte se afișează în partea dreaptă a celor existente. În momentul în care reprezentarea atinge marginea din dreapta a zonei de vizualizare, reprezentarea atinge marginea din dreapta a zonei de vizualizare, reprezentarea este ștearsă și reîncepută din marginea din stânga.
- *Diagrama tip baleiere (sweep chart, fig. 2.17, b3):*
Noile puncte se afișează în partea dreaptă a celor existente. La atingerea marginii din dreapta a zonei de vizualizare, reprezentarea nu este ștearsă; noile puncte sunt afișate începând din partea stângă, peste cele vechi (existente). Punctele noi, afișate sunt indicate de o linie verticală, care baleiază zona de vizualizare.

2. Graficul undă

Se utilizează pentru reprezentarea funcțiilor de o variabilă, având punctele distribuite uniform pe axa absciselor. Acesta se aseamănă cu *Diagrama undă*; se permite realizarea uneia sau mai multor reprezentări simultan.

3. Graficul XY

Este componenta cea mai generală, care permite realizarea reprezentărilor grafice ale funcțiilor de două variabile x și y.

În figura 2.18 se arată un exemplu de utilizare a componentei *Grafic XY*. Într-o componentă *Grafic XY* se pot realiza una sau mai multe reprezentări simultan.

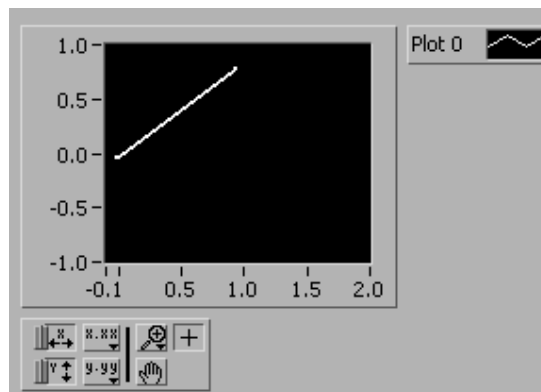


Fig. 2.18. Reprezentări utilizând o componentă Grafic XY

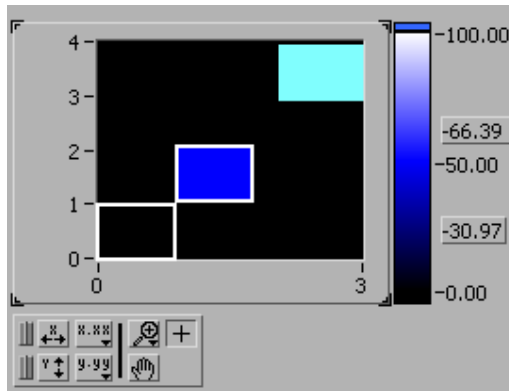
ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 4

4. Diagrama intensitate

Permite reprezentarea grafică a valorilor numerice prin intermediul unor dreptunghiuri pline, colorate. Valorile numerice (pot fi reale) sunt reținute într-o structură tip matrice (tablou ce două dimensiuni); elementele matricei reprezintă codurile culorilor de umplere a zonelor dreptunghiulare. Codurile culorilor sunt preluate dintr-o structură, care reține corespondența *cod – culoare*.

În figura 2.19 este redat un exemplu de reprezentare grafică, care folosește *Diagrama intensitate*.



10.10	0	0	0
0	50.5	0	0
0	0	0	98.08

Fig. 2.20. Matricea de valori

Fig. 2.19. Reprezentarea prin Diagrama intensitate:
a-valoarea reprezentată grafic; b-scala cu codurile culorilor

Valoarea n de la indicii de linie i și de coloană j din matrice se reprezintă grafic printr-un dreptunghi de culoarea având codul asociat n , la coordonatele: abscisă i și ordonată j . toate dreptunghiurile afișate la o valoare x a abscisei (dispuse pe aceeași verticală) corespund valorilor din matrice de pe linia i , unde $i = x$.

Conținutul matricei cu valori reprezentate este dat în figura 2.20.

Stabilirea culorilor și a codurilor culorilor se poate face interactiv sau prin program (se utilizează proprietățile asociate obiectelor). Stabilirea codificării cod-culoare în mod interactiv se face prin modificarea culorii etichetei numerice de pe scala verticală cu codurile culorilor (care este codul numeric al culorii): după selectarea scalei, se alege din meniul contextual asociat opțiunea „Adăugare a unui cod” (**Add Marker**); se selectează eticheta pentru care se dorește modificarea culorii și din meniul contextual asociat se alege opțiunea „Culoarea codului” (**Marker Color**). Se pot utiliza maxim 256 culori; această valoare maximă este influențată și de tipul plăcii video, precum și de numărul culorilor folosite curent în Windows.

Observațiile de la reprezentarea prin Diagramă undă sunt valabile și aici.

5. Graficul intensitate

Are o comportare asemănătoare „Diagrama intensitate”; deosebirea constă în faptul că la „Graficul intensitate”, în momentul afișării noilor puncte, se șterg prima dată reprezentările anterioare.

APLICATII PERSONALE

1. Explorarea conținutului casetei cu componente de interfață

Algoritmul este următorul:

1. Se pornește mediul **LabVIEW** și se alege opțiunea de creare a unui *IV* nou.
2. Se observă că sunt deschise simultan două ferestre (afișate suprapuse) corespunzătoare (*Fereastra Panoului Frontal* și *Fereastra Diagramei Bloc*) noului *IV*; fereastra panoului frontal este întotdeauna complet vizibilă.
3. Dacă caseta cu controale este vizibilă/ascunsă atunci se va exersa modalitatea de ascundere/afișare a acesteia selectând opțiunea „*Afișează Caseta cu Controale*” (**Show Controls Palette**) din bara orizontală cu meniuri „*Ferestre*” (Windows). În continuare, se va afișa caseta cu controale.
4. Se va reține și exersa posibilitatea plasării casetei cu controale oriunde pe suprafața de lucru: se poziționează cursorul mouse-ului deasupra titlului casetei cu controale (bara orizontală în care este afișat textul „**Controls**”), se ține apăsat butonul din stânga al mouse-ului în momentul atingerii poziției dorite.
5. Caseta cu controale fiind afișată, se va reține posibilitatea de ascundere a casetei, dacă se selectează cu butonul stâng al mouse-ului butonul de închidere a ferestrei (colțul dreapta sus), asociate casetei. Se reamintește că în mediul **Windows**, fiecărei aplicații îi este asociată o fereastră; în acea fereastră se execută aplicația; dacă se închide fereastra asociată aplicației, atunci este închisă și aplicația. Se ascunde caseta cu componente de interfață, închizând fereastra în care este afișată caseta.
6. Se exersează afișarea temporară a casetei cu controale la poziția curentă a cursorului mouse-ului: se apelează meniul contextual asociat suprafeței panoului frontal.
7. Se afișează caseta cu controale. Se poartă cursorul mouse-ului deasupra grupelor de componente existente; se identifică și se rețin în caiete denumirile grupelor disponibile. Pentru fiecare grupă se inspectează cu atenție (fără grabă) principalele variante de realizare ale elementelor de interfață.
8. Se închide *IV* fără a-l salva: în fereastra panou frontal se apasă simultan tastele *<Ctrl>* și *<W>*.

2. Consultarea grupului de componente tip numeric

Algoritmul este următorul:

1. Se pornește mediul **LabVIEW**. Se alege opțiunea de creare a unui *IV* nou.
2. În fereastra panoului frontal se afișează caseta cu controale.
3. Pentru grupul de componente numerice se localizează și se rețin variantele disponibile: se plimbă pe rând cursorul mouse-ului peste pictogramele asociate componentelor numerice și se reține descrierea afișată.
4. Se închide *IV* fără a-l salva: din fereastra panou frontal se alege opțiunea „*Închide*” din submeniul „*Fișier*” din bara cu meniuri (**File/Close**).

3. Consultarea grupului de componente tip boolean

Algoritmul este următorul:

1. Se pornește mediul **LabVIEW**. Se alege opțiunea de creare a unui *IV* nou.
2. În fereastra panoului frontal se afișează caseta cu controale.
3. Pentru grupul de componente booleene se localizează și se rețin variantele disponibile: se plimbă pe rând cursorul mouse-ului peste pictogramele asociate componentelor numerice și se reține descrierea afișată.
4. Se alege pe rând diferite variante de realizare și se plasează în panoul frontal; se redimensionează și se colorează obiectele. Pentru fiecare componentă de interfață aparținând tipului boolean se definește identificatorul și se asociază taste acceleratori (pentru controale). Se alege rularea continuă a *IV* (butonul „*Rulare Continuă a IV*” – fig. 1.5, b); se interacționează asupra controalelor booleene atât cu mouse-ul cât și prin tastele accelerator definite. Se oprește execuția programului prin butonul „*Oprirea Rulării*” (fig.1.5,c).
5. Se închide *IV* fără a-l salva: din fereastra panou frontal se alege opțiunea „*Închide*” din submeniul „*Fișier*”, din bara cu meniuri (**File/Close**).

4. Consultarea grupului de componente tip șir de caractere

Algoritmul este următorul:

1. Se localizează grupul de componente tip șir de caractere în caseta cu controale.
2. Se reține descrierea elementelor componente.

5. Tehnici de personalizare a componentei tabel

Să se rețină modalitățile, prin care se pot redimensiona obiecte de interfață de tip tabel, liniile și coloanele, precum și acțiunile prin care se pot selecta valorile reținute în celule.

Pentru a ajunge la pagina de asistență, care tratează aceste subiecte, se parcurg pașii următori:

1. Se pornește utilitarul de asistență (din bara cu meniuri „**Help/Online Reference**”).
2. Din secțiunea „*Privire generală*” (**Overview**) se alege prima opțiune „*Șiruri și Fișiere de Intrare/Ieșire*”(**Strings and File I/O**).
3. Se alege penultima opțiune „*Controale și Indicatoare tip Șir*” (**String Controls and Indicators**).
4. Se alege opțiunea „*Tabele*” (**Tables**).

6. Consultarea grupului de componente tip tablou și grupare de date

Algoritmul este următorul:

1. Se localizează grupul de componente tip tablou și grupare de date în caseta cu controale.
2. Se reține descrierea elementelor componente.

7. Deschiderea unor aplicații la tipul tablou

Algoritmul este următorul:

1. Se exersează tehnicile de editare (redimensionare) asupra *IV* referitoare la tablouri din biblioteca de exemple **ARRAYS.LLB**, care se găsește la următoarea cale pe hard disk:

C:\Program Files\National Instruments\labVIEW\examples\general\arrays.llb

(dacă **LabVIEW** a fost instalat pe discul C:, în directorul implicit)

Nu se salvează eventualele modificări pentru exemplele consultate, pentru păstrarea versiunii originale!

2. Se alege opțiunea de deschidere a unui *IV* existent „*Deschidere Fișier*” (**Open**) din submeniul derulant „*Fișier*” (**File**), din bara orizontală cu meniuri. Se afișează o fereastră de dialog, în care utilizatorul trebuie să indice fișierul pentru reținerea *IV* căutat (fig.2.27). Fereastra de dialog prezintă implicit conținutul directorului, în care este instalat mediul **LabVIEW**:

C:\Program Files\National Instruments\labVIEW

3. Se deschide succesiv subdirectorul „*Exemple*” (**Examples**), „*Generale*” (**General**): se selectează numele subdirectorului și
 - Se face dublu clic cu butonul stâng al mouse-uluisau
 - Se alege butonul „*Deschide*” (**Open**) aflat în partea inferioară dreaptă a ferestrei dialog.
4. Se selectează cu mouse-ul denumirea fișierului bibliotecă (conține mai multe *IV*) **ARRAYS.LLB**. Pentru afișarea conținutului bibliotecii,
 - Se face dublu clic cu butonul stâng al mouse-uluisau
 - Se alege butonul „*Deschide*” (**Open**) aflat în partea inferioară dreaptă a ferestrei dialog.
5. Se selectează cu mouse-ul *IV* „**Building Arrays.vi**” (se observă extensia „**.vi**”) și se deschide exemplul în **LabVIEW**:
 - Se face dublu clic cu butonul stâng al mouse-uluisau
 - Se alege butonul „*De acord*” (**OK**) aflat în partea inferioară, dreaptă a ferestrei dialog.
6. Se utilizează redimensionări, mutări și colorări ale elementelor panoului frontal. Se execută exemplul deschis.
7. Se deschid și alte exemple din biblioteca „**ARRAYS.LLB**”.

8. Consultarea grupului de componente pentru reprezentări grafice

Algoritmul este următorul:

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE
Laborator 5

1. Se localizează grupul de componente pentru reprezentări grafice în caseta cu controale.
2. Se reține descrierea elementelor componente.

9. Deschiderea unor IV existente, referitoare la reprezentări grafice

Exemplul 1:

1. Să se deschidă prin opțiunea „Fișier/Deschide”(File/Open) IV:
... \examples\general\graphs\charts.llb\CHARTS.vi
2. Să se lanseze în execuție; să se observe realizarea reprezentărilor. Să se oprească rularea IV prin butonul cu eticheta „STOP”. Se închide exemplul.

Exemplul 2:

1. Să se deschidă IV:
... \examples\general\graphs\gengraphs.llb\GRAPH WAVEFORM
ARRAY.vi
2. Să se lanseze în execuție. Se închide exemplul.

Exemplul 3:

1. Să se deschidă IV:
... \examples\general\graphs\gengraphs.llb\XY GRAPH.vi
2. Să se lanseze în execuție. Se închide exemplul.

Exemplul 4:

1. Să se deschidă IV:
... \examples\general\graphs\gengraphs.llb\BOUNCING CUBE.vi
2. Să se lanseze în execuție; să se modifice pe rând valoarea unor parametri și să se observe influența asupra balansării cubului în spațiu. Se închide exemplul.

10. Deschiderea unor IV existente, care reflectă utilizarea componentelor tip boolean

Exemplul 1:

- 1) Să se deschidă prin opțiunea „Fișier/Deschide”(File/Open) IV:
... \examples\general\controls\booleans.llb\Mechanical Action of Booleans.vi
- 2) Să se lanseze în execuție. Acest exemplu urmărește clarificarea comportării mecanice a butoanelor. Să se acționeze pe rând asupra butoanelor și să se rețină rezultatele obținute în funcție de comportarea asociată fiecărui control. Se închide exemplul.

Exemplul 2:

1. Să se deschidă IV aflat la calea:
... \examples\general\controls\booleans.llb\Using Buttons for
Options.vi
2. Să se lanseze în execuție. Să se apese pe rând pe cele trei butoane, care vor să simuleze un meniu; se va observa valoarea afișată de indicatorul „Acțiune” (Action). Se închide exemplul.

3. Functii

După realizarea panoului frontal al *IV*, trebuie implementată funcționalitatea programului: se construiește diagrama bloc, care reprezintă codul sursă al instrumentului (arata "CUM ?" se rezolvă problema). Pentru realizarea diagramei bloc, se utilizează limbajul grafic G, limbaj de generația a cincea; programatorul (persoana care realizează diagrama bloc) selectează și assemblează componente grafice de execuție, care definesc funcționalitatea *IV*.

Sunt prezentate elementele de bază ale limbajului G, care trebuiesc cunoscute înainte de a începe realizarea unui *IV*.

3.1. Considerații teoretice

Pentru realizarea codului sursă al *IV* nu se scriu linii de text reprezentând instrucțiuni (respectând o sintaxă rigidă și de multe ori anevoioasă); se aleg elemnte de execuție, care se plasează în diagrama bloc și se realizează legături între acestea, pentru definirea fluxului datelor.

Elementele utilizate la realizarea diagramei bloc sunt clasificate în trei grupe generale:

1. noduri;
2. terminale;
3. fire.

1. Nodurile

Sunt elementele de execuție din limbajul G, ale unui *IV*. Nodurile din limbajul G corespund operatorilor, funcțiile predefinite, instrucțiunilor, subrutinelor realizate de utilizator – din limbajele covenționale de programare strcturată.

Există șapte tipuri de noduri:

- fučții predefinite;
- subrutine (sub*IV* oferite de firma NI sau utilizator);
- instrucțiuni pentru controlul execuției programelor (numite structuri);
- formule de calcul;
- noduri proprietate;
- variabile locale și globale;
- componente care permit apelul procedurilor scrise în limbajul C sau **Pascal** (numite „Code Interface Nodes”-CINs).

Un nod se execută doar în momentul, în care există valori disponibile pentru toți parametrii de intrare; datele-rezultat sunt furnizate simultan (pe toate ieșirile) în exterior, doar după ce toate operațiile din corpul nodului s-au realizat.

Observație :

Pentru un nod se respectă, în general, următoarea convenție:

- parametrii de intrare se definesc în partea stângă a chenarului nodului;
- parametrii de ieșire se regăsesc în partea dreaptă.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 6

Respectarea convenției plasării parametrilor unui nod facilitează „citirea” diagramei bloc, conform căreia fluxul datelor este din partea stângă spre partea dreaptă.

Nodurile sunt disponibile (mai puțin nodurile proprietate asociate obiectelor din panoul frontal) prin **caseta cu funcții și IV**, prezentată în figura 3.1. Afășarea casetei cu funcții și IV(dacă nu este vizibilă) se poate face prin doua metode:

- din bara cu meniuri se alege „**Windows/Show Functions Palette**”(Ferestre/Afișează Caseta cu Funcții);
- având cursorul mouse-ului poziționat în interiorul ferestrei diagramei bloc se apelează meniul contextual.

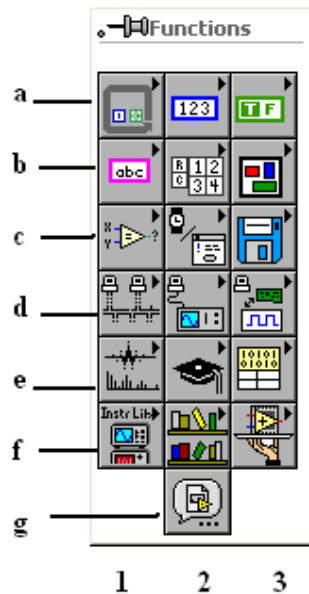


Fig. 3.1 Caseta cu funcții și IV

Dacă se poartă cursorul mouse-ului peste caseta cu funcții și IV,se afășează automat denumirea grupului curent de componente.Afișarea conținutului unui anumit grup este declanșată la realizarea unui clic la stânga al mouse-ului pe pictograma corespunzătoare.După identificarea grupului,se alege nodul dorit și se plasează la locul preconizat,în diagrama bloc.

Elementele casei cu funcții și IV (fig.3.1) se prezintă în continuare:

- instrucțiuni pentru controlul execuției programelor,formula de calcul,variabila locală și globală (fig.3.1,a1);
- funcții aritmetice,trigonometrice,logaritmice,coplexe,de conversie,constanță numerică cu valoare stabilită de utilizator,constante numerice universale (fig.3.1,a2);
- funcții logice și constanta universală adevărat/fals (fig.3.1,a3)
- funcții și constante pentru tipul șir de caractere (fig.3.1,b1);
- funcțiile tipului tablou și constanta tablou (fig.3.1,b2);
- funcțiile tipului grupare de date (cluster) și constanta grupare de date (fig.3.1,b3);
- funcții de comparare (fig.3.1,c1);

- funcții pentru dată calendaristică și timp, funcții pentru afișarea mesajelor utilizator, IV pentru tratarea erorilor (fig.3.1,c2);
- funcții și IV pentru gestionarea fișierelor (fig.3.1,c3);
- IV pentru achiziția de date (fig.3.1,d3);
- IV pentru analiza datelor (3.1,e1);
- IV utilizate în exemplele de simulare (fig.3.1,e2);
- funcție pentru apelul funcțiilor **DLL**, funcție pentru apelul procedurilor scrise în alte limbaje ©, funcții și IV pentru manevrarea datelor (fig.3.1,e3);
- IV realizate de utilizator și salvate în directorul „...\\LabVIEW\\USER.LIB”(fig.3.1,f2);
- funcții și IV pentru controlul aplicațiilor (fig.3.1,f3);
- permite inserarea apelului unui IV existent în diagrama bloc a IV apelant (fig.3.1,g2);
- IV pentru pachetul („**G Math**”) adițional de matematică (fig.3.1,g3);

Funcțiile predefinite realizează operații elementare în limbajul **G**: calcule, obținerea unor informații despre timp/dată/fișiere, conversia datelor. Funcțiile nu sunt IV, nu posedă panou frontal și diagramă bloc, astfel:

- se afișează fereastra de asistență (**Help/Help Window**);
- se afișează caseta cu funcții și IV (dacă era ascunsă);
- se deschide grupul care reține componenta căutată și se poziționează cursorul mouse-ului deasupra pictogramei asociate.

Fereastra de asistență va afișa utilizatorului numele funcției, semnătura (pictograma și tipul parametrilor de intrare/ieșire) și descrierea funcției.

Unele funcții au **numărul parametrilor** de intrare/ieșire **variabil**: pentru utilizarea funcției, trebuie să se indice explicit numărul necesar de parametri (după ce pictograma funcției a fost plasată în diagrama bloc) se face în felul următor:

- se redimensionează pictograma asociată funcției până în momentul, în care se obține numărul dorit de intrări/ieșiri (cu unealta de editare)

sau

- se selectează din meniul contextual asociat intrărilor/ieșirilor pictogramei funcției: opțiunea „*Adaugă Intrare*”(Add Input)

Exemple:

- funcția „*Combinare Aritmetică*” (**Compound Arithmetic**), din grupul „*Numeric*”;
- funcția „*Concatenare Șiruri de Caractere*” (**Concatenate Strings**), din grupul „*String*”;
- funcția „*Grupare*” (**Bundle**), din grupul „*Cluster*”;
- funcția „*Întreșere Tablouri 1D*” (**Interleave 1D Arrays**), din grupul „*Array*” sau
- opțiunea „*Adaugă Ieșire*” (**Add Output**).

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 6

Exemple:

- funcția „*Degrupare*” (**Unbundle**), din grupul „**Cluster**”
- funcția „*Împărțire Elemente Tablou 1D*” (inversia funcției de întrețesere, **Decimate 1D Array**), din grupul „**Array**”.

3.2 Functii pentru valori numerice

Funcțiile pentru valori numerice sunt grupate în subpaleta **Numeric** a paletei de funcții.

Se remarcă o primă categorie, dedicată funcțiilor aritmetice simple: adunare (**Add**), scădere (**Subtract**), înmulțire (**Multiply**), împărțire (**Divide**), incrementare (**Increment**), decrementare (**Decrement**), valoare absolută (**Absolute Value**), rotunjire la cel mai apropiat întreg (**Round To Nearest**), rotunjire la întregul inferior (**Round To -Infinity**), rotunjire la întregul superior (**Round To +Infinity**), rădăcina pătrată (**Square Root**), schimbare de semn (**Negate**), produsul dintre o valoare x și o putere a lui 2 (**Scale By Power Of 2**), inversare (**Reciprocal**).

În subpaleta **Numeric** se găsește de asemenea elementul **Numeric Constant** (constanta numerică).

Atunci când un astfel de element este dispus în diagramă, el capătă automat valoarea 0 și intră în modul de editare, așteptând ca utilizatorul să introducă valoarea numerică dorită.

Dacă, imediat după dispunerea în diagramă, în loc de a introduce o valoare în element, utilizatorul efectuează altă operație, o altă valoare constantă poate fi specificată utilizând unealta de operare sau pe cea de editare a textelor.

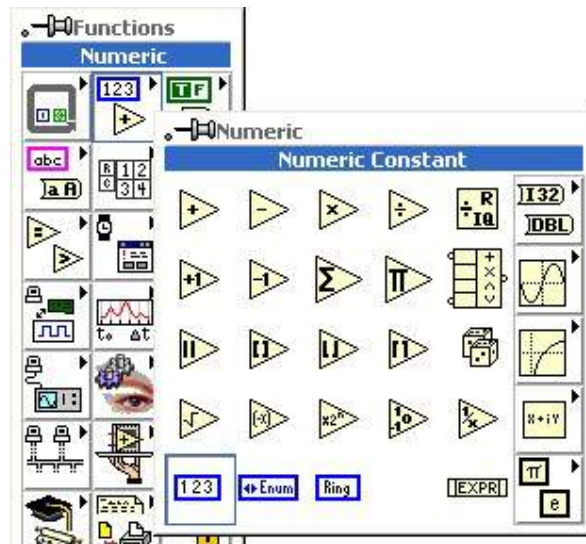


Fig. 3.2 Functii pentru valori numerice

EXERCITIU

In subpaleta **Numeric** a paletei de functii, identificati functiile aritmetice simple enumerate mai sus. Pe parcursul navigarii in paleta de functii, urmariti descrierea functiilor in fereastra **Help**.

3.3 Functii pentru valori boolene

Functiile pentru valori boolene sunt grupate in subpaleta **Boolean** a paletei de functii.

Se remarca functiile logice simple SI (**And**), SAU (**Or**), SAU EXCLUSIV (**Exclusive Or**), NEGARE (**Not**), SI NEGAT (**Not And**), SAU NEGAT (**Not Or**), SAU NEGAT EXCLUSIV (**Not Exclusive Or**) si IMPLICA (**Implies**).

Subpaleta contine de asemenea, in partea inferioara, cele doua constante logice: ADEVARAT (**True**) si FALS (**False**).

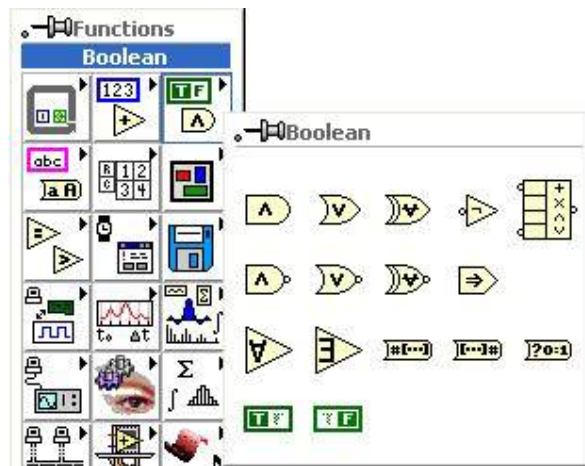


Fig. 3.3 Functii pentru valori boolene

EXERCITIU

In subpaleta **Boolean** a paletei de functii, identificati functiile logice simple si constantele logice enumerate mai sus. Pe parcursul navigarii in paleta de functii, urmariti descrierea functiilor in fereastra **Help**

3.4 Functii pentru valori alfanumerice (string)

Functiile pentru valori alfanumerice sunt grupate in subpaleta **String** a paletei de functii.

Printre cele mai simple se numara cele pentru transformarea caracterelor in majuscule (**To Upper Case**) si pentru transformarea caracterelor in minuscule (**To Lower Case**).

In partea sa inferioara, subpaleta contine un element **String Constant** pentru introducerea in diagrama de valori alfanumerice constante.

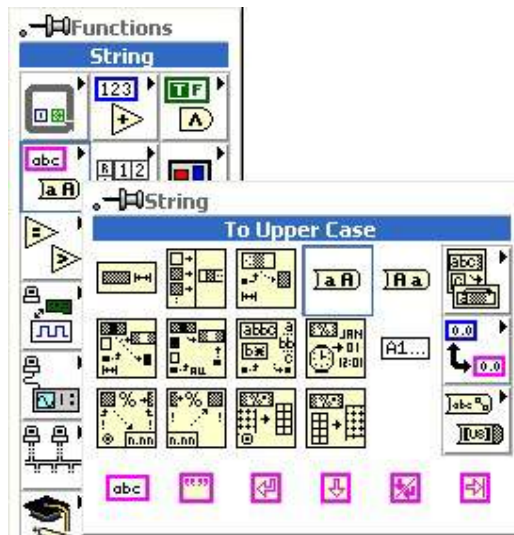


Fig. 3.4 Functii pentru valori alfanumerice

EXERCITIU

In subpaleta **String** a paletei de functii, identificati functiile si constanta alfanumerica enumerate mai sus. Pe parcursul navigarii in paleta de functii, urmariti descrierea functiilor in fereastra **Help**

3.5 Inserarea simbolurilor de functii in diagrama

1. Dispunerea unei functii in diagrama incepe cu selectarea simbolului corespunzator din **paleta de functii**.
2. Dupa selectarea simbolului functiei, se deplaseaza cursorul mouse-ului pana in pozitia din diagrama in care se doreste dispunerea.
Atat timp cat cursorul mouse-ului este deplasat, simbolul functiei va fi reprezentat impreuna cu terminalele corespunzatoare datelor proprii de intrare si de iesire.
3. Daca, in timpul deplasarii, simbolul functiei este trecut suficient de aproape de un terminal sau de o zona a fluxului de date la care se poate efectua o legatura, acea legatura este stabilita automat.
Daca simbolul functiei este indepartat, in cursul deplasarii, de zona sau terminalul respectiv, legatura stabilita automat dispare.
Daca nu se doreste efectuarea automata a unei legaturi, se apasa tasta **Space** in timp ce butonul mouse-ului este apasat.
4. Cand simbolul functiei ajunge in pozitia dorita, se face click cu unul dintre butoanele mouse-ului.
Daca in acel moment exista o legatura stabilita automat, legatura respectiva se pastreaza.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 6

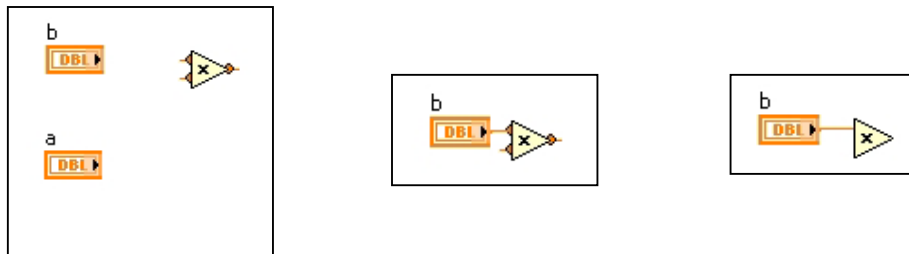


Fig. 3.5

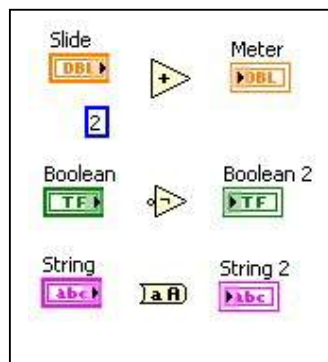
EXERCITIU

Din subpaleta **Numeric** a paletii de functii, introduceti in diagrama o functie **Add** si pozitionati-o ca in figura, intre terminalele celor doua elemente numerice. Introduceti de asemenea in diagrama un element de tipul **Numeric Constant** si stabiliti-i acestuia valoarea **2**.

Din subpaleta **Boolean**, introduceti in diagrama o functie **Not** si pozitionati-o intre terminalele celor doua elemente booleene.

Din subpaleta **String**, introduceti in diagrama o functie **To Upper Case** si pozitionati-o intre terminalele celor doua elemente alfanumerice.

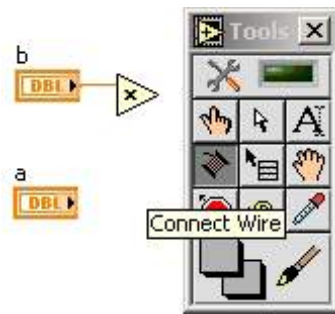
Dupa pozitionarea simbolurilor de functii in diagrama, deschideti fereastra **Help** si afisati in aceasta informatiile referitoare la fiecare functie prin deplasarea mouse-ului deasupra simbolului



3.6 Realizarea legaturilor in diagrama

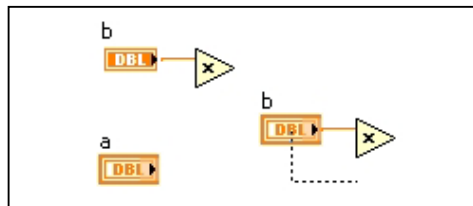
Realizarea unei legaturi intre doua componente ale diagramei se efectueaza cu ajutorul uneltei de conectare.

Ordinea in care se selecteaza cele doua componente (sursa si destinatia) nu este importanta. Legatura poate fi realizata in orice sens, deoarece mediul de programare va identifica automat sursa si destinatia pentru a stabili sensul de circulatie al datelor.

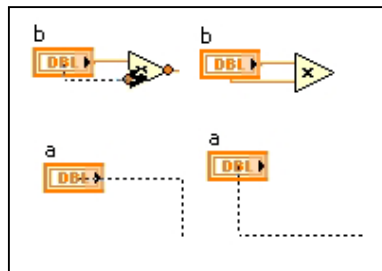


1. Se deplaseaza cursorul mouse-ului (unealta de conectare) deasupra primei componente (aceasta va incepe sa clipeasca) si se face click. Actiunea este echivalenta cu prinderea unui capat al firului de componenta respectiva.

Din acest moment, daca mouse-ul este deplasat, pe ecran va apare imaginea unui fir cu un capat fixat de prima componenta si cu un capat legat de cursorul mouse-ului.



2. Se deplaseaza cursorul mouse-ului deasupra celei de-a doua componente (si aceasta va incepe sa clipeasca) si se face click, fixand astfel al doilea capat al firului de aceasta componenta.



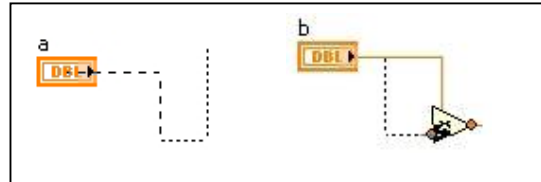
Obs. 1: Daca, la un moment dat, firul care urmeaza mouse-ul are un sens (orar sau trigonometric) neconvenabil, sensul poate fi schimbat prin apasarea tastei blanc (spatiu).

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 7

Obs. 2: Daca se doreste ca traseul unei legaturi sa fie format din mai multe portiuni in unghi drept, un click intr-un punct din diagrama va fi echivalent cu fixarea firului in acel punct.

Obs. 3: Daca o componenta a diagramei are rol de sursa pentru un traseu ramificat, se pot efectua legaturi care sa aiba drept punct de pornire un segment al fluxului de date.



Obs. 4: Daca o legatura a fost efectuata corect, ea va fi afisata in culoarea corespunzatoare tipului de date ce vor circula prin zona respectiva a fluxului de date. Daca legatura este incorecta (conecteaza doua componente de tipuri diferite), aceasta va fi afisata cu culoarea neagra si intrerupta.

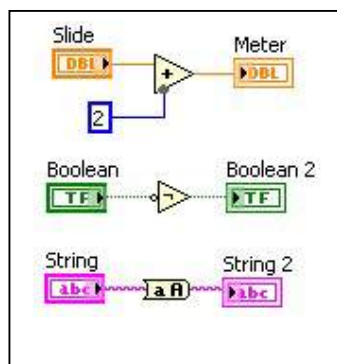


EXERCITIU

Utilizand unealta de conectare, realizati legaturile din figura de mai jos. Observati ca, similar terminalelor, legaturile din diagrama au culori ce semnificatia tipul datelor scalare corespunzatoare (portocaliu pentru numere reale, albastru pentru numere intregi, verde pentru valori booleene, roz pentru valori alfanumerice).

Astfel, atunci cand programul va rula, indicatorul **Meter** va afisa suma dintre valoarea controlului **Slide** si constanta **2**, indicatorul **Boolean 2** va afisa valoarea logica negata a controlului **Boolean**, iar indicatorul **String 2** va afisa textul din controlul **String** transformat in majuscule.

Deschideti fereastra **Help**, selectati unealta de conectare si pozitionati cursorul mouse-ului deasupra unei legaturi astfel incat aceasta sa "clipecte". Observati ca, in aceasta situatie, in fereastra **Help** sunt afisate informatii referitoare la tipul de valori ce circula prin legatura respectiva.



3.7 Fluxul de date

Fluxul de date este format din totalitatea legaturilor ce conecteaza diversele terminale de elemente, simboluri de functii sau proceduri si structuri de programare din cadrul **diagramei**.

Rolul fluxului de date este acela de **reprezentare grafica a algoritmului** dupa care aplicatia va prelucra **datele de intrare** pentru a calcula **valorile de iesire**. Traseele ce formeaza fluxul de date pot fi simple, cu o sursa si o destinatie, sau ramificate, cu o sursa si mai multe destinatii.

Datele de intrare sunt acele valori (informatii) pe care un program le primeste dintr-o sursa oarecare.

Datele de intrare pot fi:

- introduse de catre utilizator prin intermediul tastaturii sau al mouse-ului;
- extrase chiar de catre program din fisiere de date aflate pe diverse suporturi de memorie (hard-disk, discheta, CD etc.);
- primite prin intermediul diverselor echipamente (placa de sunet, camera video, placa de retea, placa de achizitie de date etc.);
- extrase de catre program din memorie, daca au fost stocate in aceasta de catre alte programe

Datele de iesire reprezinta valorile pe care programul le genereaza ca urmare a executarii instructiunilor sale si le transmite utilizatorului.

Transmiterea datelor de iesire se poate efectua:

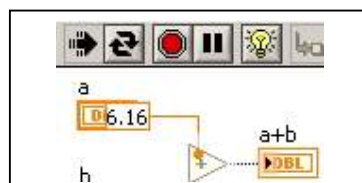
- prin afisarea acestora pe monitor;
- prin salvarea in fisiere de date ce vor fi ulterior accesate de catre utilizator sau de catre alte programe;
- prin stocarea in memorie pentru a fi utilizate de alte programe;
- prin intermediul altor echipamente

Traseele ce formeaza fluxul de date pot fi simple, cu o sursa si o destinatie, sau ramificate, cu o sursa si mai multe destinatii. Indiferent de modul sau sensul in care traseele sunt dispuse pe diagrama, "circulatia" datelor in cadrul fluxului se efectueaza doar **de la surse catre destinatii**.

Prin **sursa** se intelege, de exemplu, terminalul unui element de control, simbolul unei constante sau zona de iesire a unei functii sau proceduri.

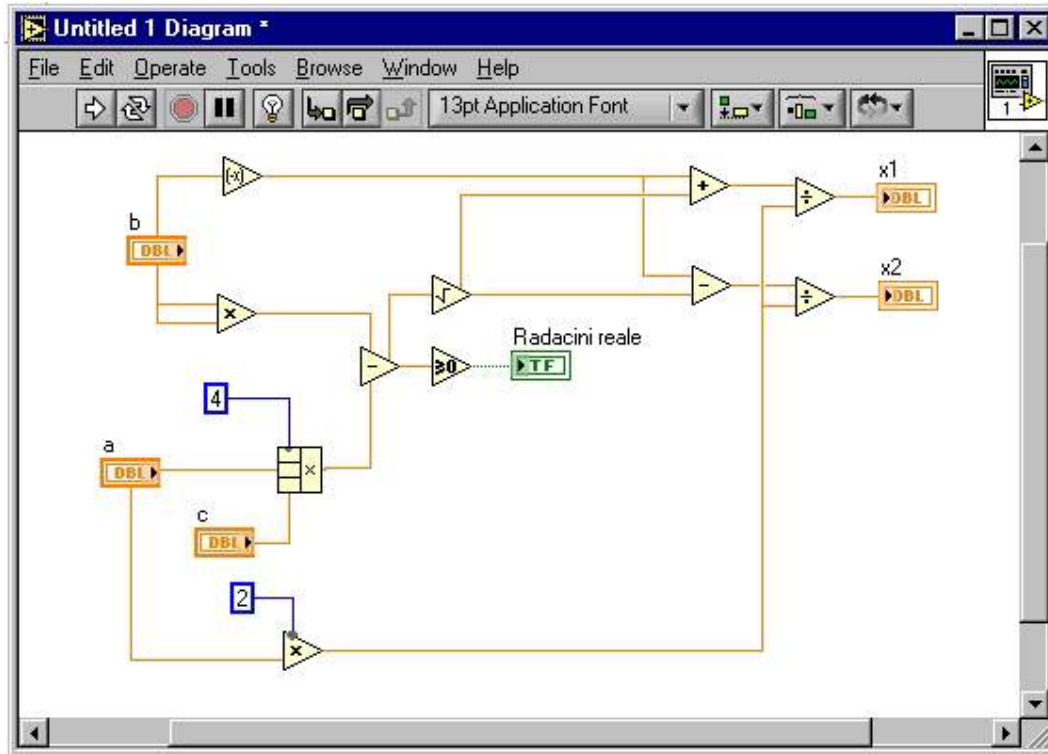
Prin **destinatie** se intelege, de exemplu, terminalul unui element indicator sau zona de intrare a unei functii sau proceduri.

In orice mod de rulare (**Run, Run Continously**), prin apasarea butonului **Highlight Execution**, utilizatorul are posibilitatea de a urmari o animatie ce sugereaza deplasarea valorilor prin fluxul de date. Atunci cand o valoare paraseste o componenta a diagramei ce este o sursa a unei legaturi, valoarea respectiva este afisata in diagrama.



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 7



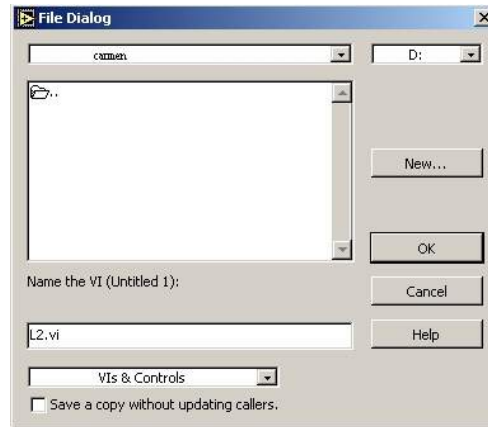
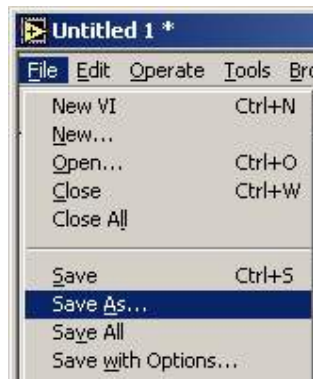
EXERCITII

1. In panoul aplicatiei pe care ati realizat-o, cu **unealta de operare**, modificati valoarea elementului de control **Slide**, schimbati pozitia controlului **Boolean** si introduceti un text format din minuscule in controlul **String**. Apasati sageata **Run** pentru a rula programul. Schimbati din nou pozitia controlului **Boolean** si apasati din nou sageata **Run**.
2. Apasati butonul **Run Continuously** si, in timp ce programul ruleaza in modul continuu, modificati valoarea elementului de control **Slide**, schimbati pozitia controlului **Boolean** si modificati textul din controlul **String**.
3. Apasati butonul **Pause**. Rularea programului se va suspenda temporar, aplicatia va trece automat in diagrama si va indica prin aprindere intermitenta locul din diagrama in care programul s-a oprit.
4. Apasati butonul **Highlight Execution**, apoi butonul **Step Over**. Observati afisarea in diagrama a valorilor de iesire din functii.
- 5."Stingeti" butonul **Continue** (cel care avea initial eticheta **Pause**). Observati afisarea in diagrama a modului in care valorile circula pe fluxul de date al programului.
- 6."Stingeti" butonul **Highlight Execution** si intrerupeti rularea continua prin apasarea butonului **Abort execution**.

3.8 Salvarea unei aplicatii

Atat timp cat inca nu a fost salvat intr-un fisier, un program are numele **Untitled** urmat de un numar de ordine. Pentru a specifica folder-ul in care programul sa fie salvat, precum si numele fisierului, se alege din meniul **File** optiunea **Save As...** In urma selectarii comenzii **Save As...**, programul deschide o fereastra de dialog in care utilizatorul poate selecta folder-ul in care sa se efectueze salvarea si numele fisierului.

Programele realizate in mediul de programare grafica LabVIEW se salveaza in fisiere cu extensia **VI !**



Din momentul in care a fost salvat prima data, programului i se afiseaza numele atat in bara superioara a panoului cat si in cea a diagramei. Daca, dupa salvare, programului i se aduc in continuare modificari, langa numele acestuia apare un asterisc (*), semnalizand astfel faptul ca versiunea ce a fost salvata nu este cea mai recenta.

Comanda **Save As...** se poate folosi daca se doreste salvarea unei noi versiuni a fisierului intr-o noua locatie sau cu un nou nume

EXERCITII

1. Pe discul **D** al computerului, in directorul **Utilizatori**, subdirectorul **CCD**, creati un subdirector cu numele vostru.
2. Salvati aplicatia pe care ati realizat-o in subdirectorul propriu.

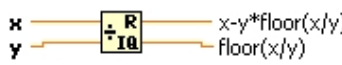
TEMA

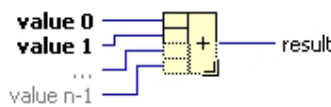
1. Realizati un program pentru calculul ariei si perimetrului unui cerc, in functie de valoarea razei introdusa de catre utilizator
2. Realizati un program pentru calculul ariei si perimetrului unei elipse, in functie de valorile introduse de catre utilizator pentru cele doua semiaxe
3. Realizati un program care sa exemplifice utilizarea diverselor functii booleene
4. Realizati un program pentru rezolvarea ecuatiei de gradul al doilea

4. Functii pentru valori scalare



Functii pentru valori numerice

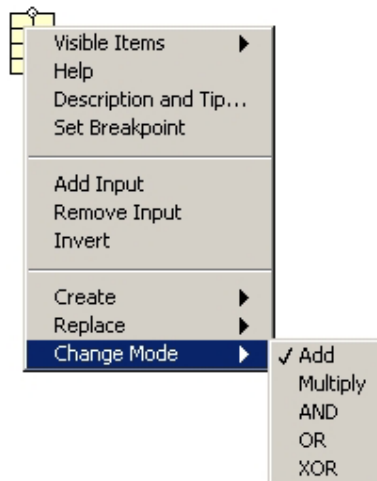
Pe langa functiile aritmetice simple descrise in lectia anterioara, subpaleta **Numeric** a paletei de functii contine, in zona sa principala, si urmatoarele functii:

 **Quotient & Remainder** calculeaza catul si restul impartirii valorii conectate la intrarea X la valoarea conectata la intrarea Y.

 **Compound Arithmetic** (disponibila si in subpaleta **Boolean** a paletei de functii) permite efectuarea operatiilor de adunare (**Add**) sau inmultire (**Multiply**), sau a operatiilor logice **SI (AND)**, **SAU (OR)** sau **SAU EXCLUSIV (XOR)**, asupra unui numar variabil de valori de intrare.


Numarul de intrari este stabilit de catre utilizator prin "dimensionarea" simbolului functiei astfel:

- se plaseaza cursorul mouse-ului la mijlocul uneia din laturile orizontale ale simbolului, astfel incat pe laturile de sus si de jos sa apara cate un cerculet 
- se apasa butonul stang al mouse-ului si se deplaseaza mouse-ul tinand butonul apasat
- pe masura ce mouse-ul este deplasat, simbolul functiei isi modifica dimensiunile, modificandu-se astfel numarul de intrari 
- cand numarul de intrari in functie este cel dorit, se elibereaza butonul mouse-ului. Pentru a selecta operatia efectuata de catre functie (adunare, inmultire etc).
- se apasa butonul drept al mouse-ului pentru a deschide meniul propriu al functiei
- se deschide submeniul **Change Mode**
- se selecteaza operatia dorita



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 7

 number (0 to 1) **Random Number (0-1)** este o functie ce nu poseda intrari, rolul sau fiind de a genera o valoare reala aleatoare in intervalul (0, 1).

Subpaleta **Numeric** a paletei de functii contine de asemenea o serie de subpalete dedicate unor categorii particulare de functii:

>>>subpaleta **Conversion**, continand printre altele:

- 12 functii pentru schimbarea modului de reprezentare a unei valori numerice.
- functia **Boolean To (0,1)** (disponibila si in subpaleta **Boolean** a paletei de functii), care transforma o valoare logica (**True** sau **False**) intr-o valoare numerica (**1** respectiv **0**).

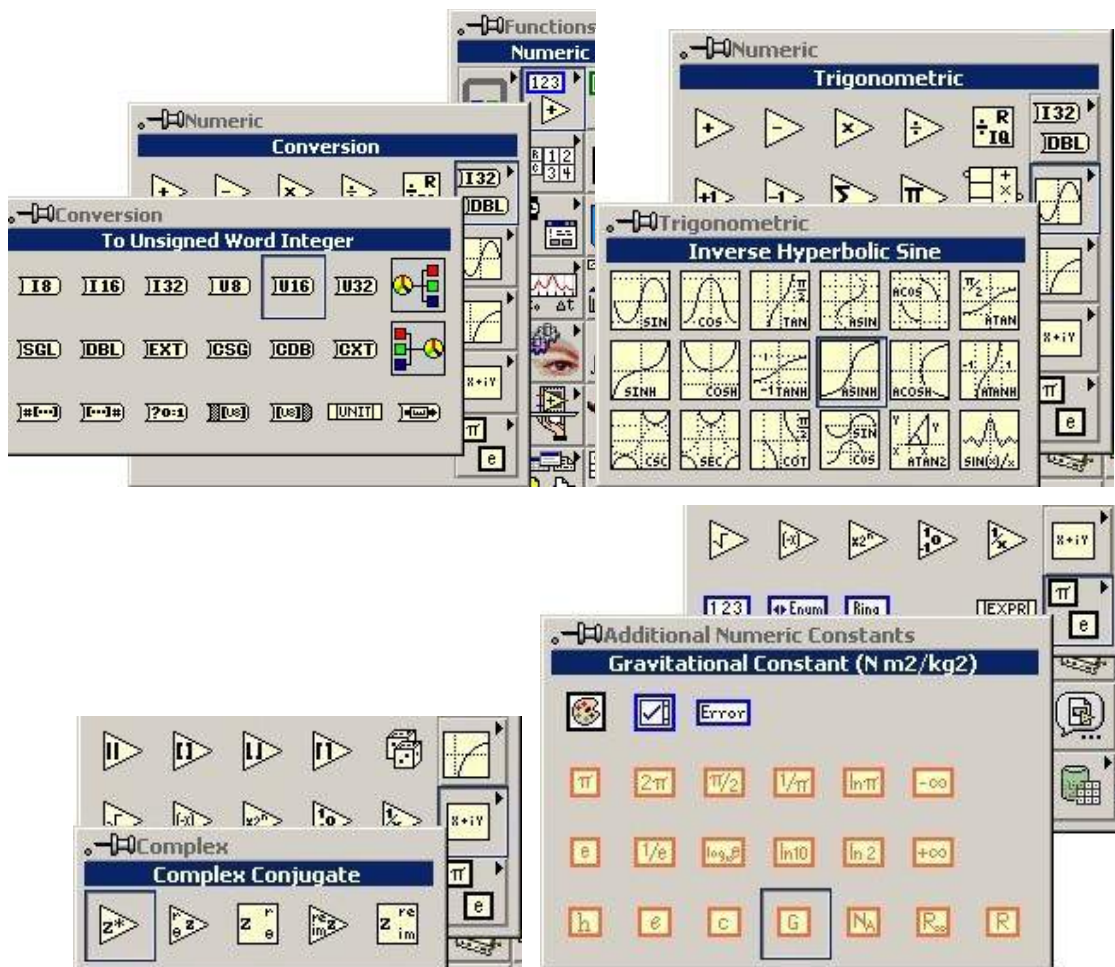
Boolean **?0:1** — 0, 1

>>> subpaleta **Trigonometric**, continand majoritatea functiilor trigonometrice

>>> subpaleta **Logarithmic**, continand functii exponentiale si logaritmice

>>> subpaleta **Complex**, continand functii asociate numerelor complexe

Subpaleta **Additional Numeric Constants** a paletii de functii contine valori constante des utilizate in trigonometrie si in lucrul cu functii exponentiale sau logaritmice, precum si constante utilizate in fizica.



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE


Laborator 7


EXERCITIU

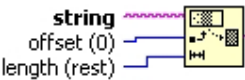
In zona principala a paletii de functii si in subpaletile dedicate ale acesteia, identificati functiile si constantele enumerate mai sus.

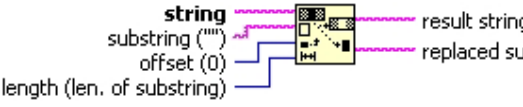
Construiti un program care sa afiseze catul si restul impartirii a doua numere naturale introduse de catre utilizator.

Funcții uzuale pentru valori alfanumerice

 >>> Funcția **String Length** determina numărul de caractere dintr-o valoare alfanumerică (text). Sunt luate în considerare și caracterele neafisabile (spatiu, Tab, sfarsit de linie etc). Numarul de caractere de la iesirea functiei este exprimat ca un intreg pe patru octeti.

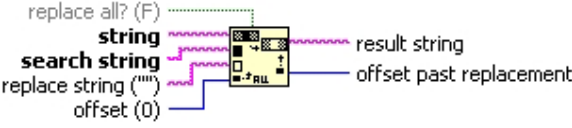
 >>> Funcția **Concatenate Strings** posedă un număr de intrări ce poate fi modificat de către utilizator prin dimensionarea simbolului functiei. La iesirea sa, functia genereaza o valoare alfanumerica obtinuta prin concatenarea textelor de la intrarile sale, in ordinea de sus in jos.

 >>> Funcția **String Subset** extrage, din textul conectat la intrarea **string**, o porțiune de **length** caractere începând de la caracterul aflat la poziția **offset** inclusiv (pentru primul caracter se considera **offset** = 0). Dacă la intrarea **offset** a functiei nu este introdusa nici o valoare, functia considera ca extragerea se va efectua incepand cu primul caracter (valoarea implicita 0 a intrarii **offset**). Dacă la intrarea **length** nu este introdusa nici o valoare, functia considera ca extragerea se va efectua pana la ultimul caracter inclusiv.

 >>> Funcția **Replace Substring** șterge o porțiune din textul conectat la intrarea **string** și introduce în locul respectiv textul conectat la intrarea **substring**.

Porțiunea înlocuită începe de la caracterul aflat pe poziția **offset** inclusiv și conține **length** caractere.

Intrarea **offset** are valoarea implicită 0, iar intrarea **length** are o valoare implicită egală cu numărul de caractere legat la intrarea **substring**. Dacă la intrarea **substring** nu se leagă nici o valoare (textul de inserat este vid), functia efectuează de fapt o ștergere a unei porțiuni, iar dacă la intrarea **length** se leagă valoarea 0 (textul de șters este vid) functia efectuează de fapt o inserare a unei porțiuni.

 >>> Funcția **Search and Replace String** caută textul **search string** în textul conectat la intrarea **string**, iar atunci când îl găsește îl înlocuiește cu textul conectat la intrarea **replace**

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 7

string. Cautarea are loc incepand de la pozitia **offset** a textului string.
Daca la intrarea **offset** nu se conecteaza nici o valoare, cautarea are loc incepand cu prima pozitie.

Daca la intrarea **replace string** nu se conecteaza nici o valoare, atunci functia efectueaza de fapt o stergere (inlocuire cu text vid).

Daca la intrarea **replace all?** nu se conecteaza nici o valoare, atunci functia se opreste dupa prima aparitie a textului **search string**. Daca la aceasta intrare se conecteaza o valoare logica **True**, atunci functia efectueaza inlocuiri pentru toate aparitiile textului **search string**.

Functia genereaza textul rezultat in urma inlocuirii la iesirea **result string**, iar la iesirea **offset past replacement** genereaza o valoare indicand pozitia caracterului aflat imediat dupa ultimul caracter inlocuit.

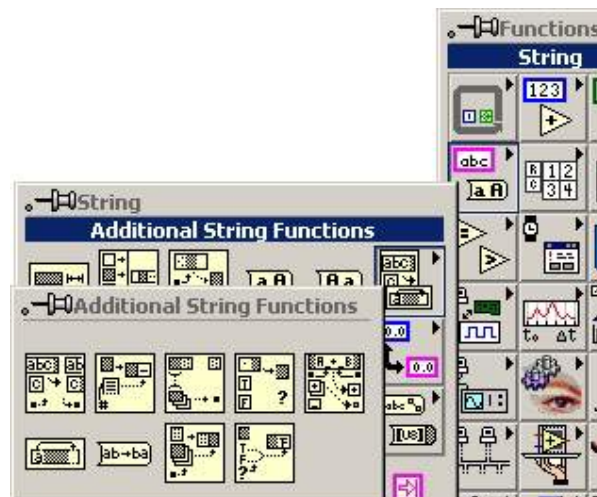
EXERCITIU

Pentru fiecare dintre functiile prezentate mai sus, construiti cate un program care sa exemplifice modul de utilizare a functiei.

De exemplu, pentru functia **String Length**, panoul programului va contine un element de control pentru valori alfanumerice, in care utilizatorul va introduce un text, si un element indicator pentru valori numerice, in care programul va afisa numarul de caractere ale textului introdus.

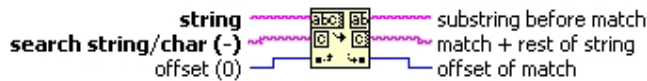
Funcții adiționale pentru valori alfanumerice

Subpaleta **Additional String Functions** a paletelor **String** contine o serie de functii uzuale pentru lucrul cu valori alfanumerice:



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 7



>>> Functia **Search/Split String** cauta textul conectat la intrarea **search string/char** in textul conectat la intrarea

string.

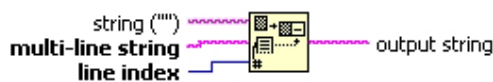
Cautarea se efectueaza incapand cu pozitia **offset**.

Daca textul cautat este gasit, functia divide textul initial in doua parti:

- partea dinainte de textul cautat (**substring before match**)
- textul cautat urmat de restul textului in care s-a facut cautarea (**match + rest of string**).

Functia genereaza la iesirea **offset of match** pozitia in care a fost gasit textul cautat.

Daca textul cautat nu a fost gasit, iesirea **substring before match** contine tot textul legat la intrarea **string**, iar iesirea **offset of match** are valoarea -1.



>>> Functia **Pick Line** selecteaza din textul **multi-line string**, ce contine mai multe linii, pe cea cu indicele **line index** si adauga linia

respectiva la textul conectat la intrarea **string**.

Daca la intrarea **string** nu a fost conectata nici o valoare, rezultatul este chiar linia cu indicele specificat.



>>> Functia **Match True/False String** verifica daca unul din textele **true string** sau **false string** coincide cu inceputul textului conectat la intrarea

string.

In caz afirmativ, functia genereaza la iesirea **output string** restul textului de la intrarea **string**, iar la iesirea **selection** o valoare booleana corespunzatoare intrarii care a coincis (**True** pentru **true string** si **False** pentru **false string**).

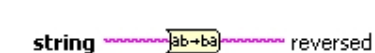
Daca ambele texte coincid, functia considera textul **true string**.

Daca nici un text nu coincide, iesirea **selection** are valoarea **False** iar iesirea **output string** contine intreg textul **string**.



>>> Functia **Rotate String** efectueaza o permutare circulara a caracterelor textului conectat la intrarea

string, mutand primul caracter pe ultima pozitie.



>>> Functia **Reverse String** inverseaza ordinea caracterelor textului conectat la intrarea **string**



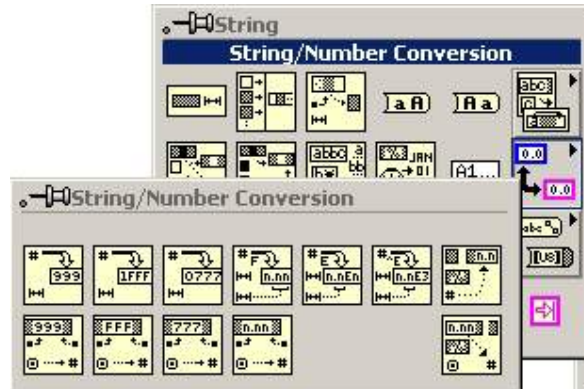
>>> Functia **Append True/False String** concateneaza la textul conectat la intrarea **string** unul din textele conectate la intrarile **true string** sau **false string**, in functie de valoarea booleana

conectata la intrarea **selector**.


Subpaleta **String/Number Conversion** a paletii **String** contine functii pentru conversia valorilor alfanumerice in valori numerice sau pentru efectuarea operatiei inverse.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE


Laborator 7



Dintre acestea, cele mai des utilizate sunt:

number —  **width (-)** — decimal integer string

>>> Functia Number To Decimal String
transforma numarul intreg de la intrarea **number** intr-un text de cel putin **width** caractere. Daca numarul de la intrarea functiei nu este intreg, el este trunchiat. Daca intrarea **width** nu este legata, numarul de caractere ale textului va fi egal cu numarul de cifre ale intregului.

string offset default (OL)  offset past number number

>>> Functia **Decimal String To Number** transforma intr-o valoare numerica intreaga portiunea din textul **string** incepand de la pozitia **offset** inclusiv (implicit de la pozitia 0).

Modul de reprezentare al valorii de la iesirea **number** este identic cu cel al valorii numerice intregi conectate la intrarea **default** (implicit intreg pe 32 de biti). Iesirea **offset past number** are semnificatia pozitiei primului caracter din **string** dupa portiunea care a fost transformata in valoare numerica.

Subpaleta **String/Number Conversion** contine de asemenea functii pentru transformarea textelor din si in numere in baza de numeratie 16 sau 8 sau in formatul exponential, fractional sau ingineresc.

EXERCITIU

Pentru fiecare dintre functiile prezentate mai sus, construiti cate un program care sa exemplifice modul de utilizare a functiei.

De exemplu, pentru functia **Reverse String**, panoul programului va contine un element de control pentru valori alfanumerice in care utilizatorul va introduce un text si un element indicator, de asemenea pentru valori alfanumerice, la care programul va afisa textul inversat.

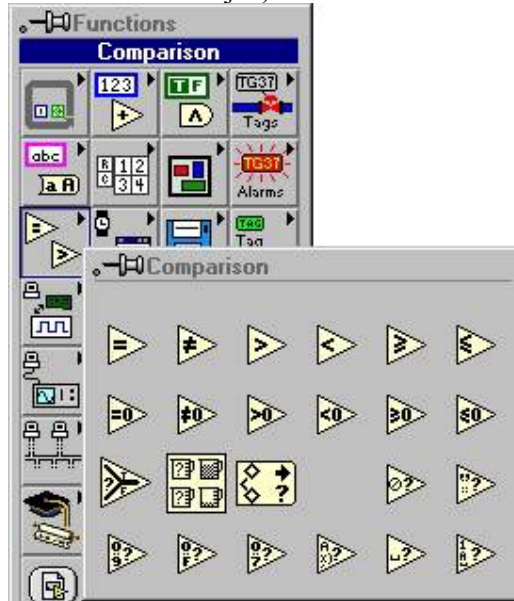
ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 8

Functii pentru comparatii

Functiile pentru efectuarea comparatiilor sunt grupate in subpaleta **Comparison** a paletei de functii. Caracteristic acestor functii este faptul ca, daca datele de intrare pot fi de diverse tipuri (numerice, booleene, alfanumerice etc), datele de iesire sunt la majoritatea acestor functii valori logice (booleene).


Conteaza de asemenea ordinea in care datele de intrare sunt legate la terminalul functiei (la intrarea de sus sau la cea de jos).

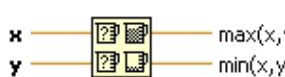


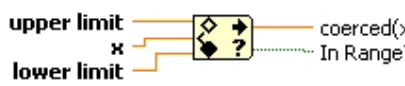
In cazul functiei **Greater?** de exemplu, iesirea va avea valoarea **True** daca valoarea de la intrarea de sus este mai mare decat valoarea de la intrarea de jos si va avea valoarea **False** in caz contrar.



Pe langa functiile de comparare a doua marimi sau de comparare a unei marimi cu valoarea nula, subpaleta **Comparison** contine urmatoarele:


 >>> Functia **Select** genereaza la iesire valoarea de la intrarea **t** (**True**) daca valoarea logica de la intrarea **s** este **True**, respectiv valoarea de la intrarea **f(False)** daca valoarea logica de la intrarea **s** este **False**. Valorile de la intrarile **t** si **f** pot fi atat numerice cat si alfanumerice.


 >>> Functia **Max & Min** genereaza la iesirea **max** cea mai mare dintre valorile **x** si **y**, respectiv la iesirea **min** pe cea mai mica dintre acestea.


 >>> Functia **In Range and Coerce** verifica daca valoarea de la intrarea **x** este cuprinsa intre valorile de la intrarile **lower limit** (limita inferioara) si **upper**


ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 8

limit (limita superioara).

Valoarea logica reprezentand rezultatul verificarii este generata la iesirea **In Range?** In mod implicit, intervalul de incadrare este inchis la stanga si deschis la dreapta. Inchiderea sau deschiderea intervalului la extremitatile sale poate fi configurata de catre utilizator din meniul propriu al functiei, activand sau dezactivand optiunile **Include upper limit** si **Include lower limit**.

Rezultatul "fortarii" valorii **x** in intervalul prescris este generat la iesirea **coerced(x)**.

char  **digit?** >>> Functia **Decimal Digit?** genereaza o valoare logica **True** in situatia in care caracterul de la intrarea sa este o cifra intre 0 si 9 si genereaza o valoare **False** in caz contrar.

Subpaleta **Comparison** contine de asemenea functii pentru a verifica daca un caracter este cifra in baza de numeratie 16, cifra in baza de numeratie 8, caracter tiparibil sau caracter netiparibil.

EXERCITIU

Pentru fiecare dintre functiile prezentate mai sus, construiti cate un program care sa exemplifice modul de utilizare a functiei.

De exemplu, pentru functia **Select**, panoul programului va contine un element de control boolean, doua elemente de control alfanumerice si un indicator alfanumeric. Programul va afisa la elementul indicator una din valorile de la cele doua controale alfanumerice, in functie de valoarea **True** sau **False** a elementului boolean

Functii pentru lucrul cu valori data - timp si de dialog

Functiile pentru lucrul cu valori data - timp si functiile de dialog se gasesc in meniul **Time & Dialog** al paletei de functii.



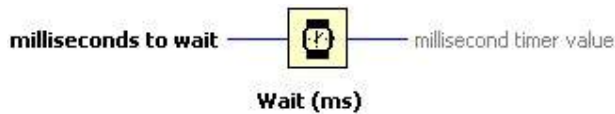
ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 8

Printre cele mai utilizate functii din aceasta categorie se numara:



>>> **Tick Count (ms):** determina numarul de milisecunde indicat de ceasul intern al computerului (considerat de la momentul la care calculatorul a fost pornit)

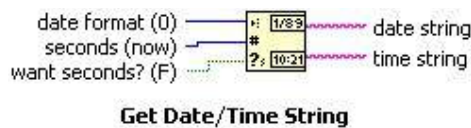


>>> **Wait (ms):** intrerupe executia programului pentru intervalul de timp specificat



>>> **Format Date/Time String:** transforma o valoare numerica reprezentand un numar de secunde intr-un text.

Numarul de secunde este considerat a fi scurs incepand cu data de 1 ianuarie 1904, ora 12.00. La intrarea **time format string** a functiei se poate specifica formatul in care textul rezultat sa fie construit.



>>> **Get Date/Time String:** transforma o valoare numerica reprezentand un numar de secunde in doua texte, unul reprezentand data si celalalt timpul. La intrarea **date format** se poate lega una din valorile **0**, **1** sau **2**, reprezentand

formatele **scurt**, **lung** sau **abreviat** pentru data. Daca numarul de secunde nu este conectat, se considera numarul de secunde al ceasului intern al computerului. Iesirea text ce reprezinta timpul poate contine sau nu secunde, in functie de valoarea intrarii booleene **want seconds?**



>>> **Get Date/Time In Seconds:** determina numarul de secunde scurs incepand cu data de 1 ianuarie 1904, ora 12.00, pana la data si timpul curente ale calculatorului.



>>> **One Button Dialog:** determina aparitia unei ferestre de dialog, continand un text specificat prin program, fereastra ce asteapta apasarea de catre utilizator a unui buton (numele butonului poate fi si el configurat).



>>> **Two Button Dialog:** determina aparitia unei ferestre de dialog, continand un text specificat prin program, fereastra ce asteapta apasarea de catre utilizator a unuia din

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 8

cele doua butoane disponibile (numele butoanelor pot fi de asemenea configurate). Functia genereaza o valoare booleana **True** daca utilizatorul a apasat butonul din stanga sau **False** daca a fost apasat butonul din dreapta.

EXERCITII

Pentru fiecare dintre functiile prezentate mai sus, construiti cate un program care sa exemplifice modul de utilizare a functiei.

De exemplu, pentru functia **Tick Count (ms)**, realizati un program care sa afiseze numarul de secunde scurse de la pornirea calculatorului.

TEMA

1. Construiti un program care sa genereze un numar aleator intr-un interval ale carui limite minima si maxima sunt specificate de catre utilizator
2. Realizati un program in care utilizatorul sa dispuna de patru elemente de control booleene sub forma de LED-uri. Considerand ca cele patru LED-uri reprezinta cifrele binare ale unui numar in baza de numeratie 2, programul sa afiseze numarul respectiv in baza 10.
3. Construiti un program in care utilizatorul sa introduca lungimea unui vector si unghiul acestuia cu orizontala (in grade), iar programul sa afiseze lungimile proiectiilor vectorului pe cele doua axe de coordonate.
4. Construiti un program in care utilizatorul sa introduca lungimile si unghiurile cu orizontala (in grade) pentru doi vectori, iar programul sa afiseze lungimea vectorului rezultat, unghiul facut de acesta cu orizontala si lungimile proiectiilor sale pe cele doua axe.
5. Realizati un program pentru adunarea, scaderea, inmultirea si impartirea a doua numere complexe.
6. Intr-un text, sa se inlocuiasca toate spatiile (blanc) cu liniuta de subliniere (underscore)
7. Fiind dat un cuvnt, sa se transforme acesta astfel incat prima litera sa fie majuscula si restul literelor sa fie minuscule.
8. Fiind dat un text, sa se verifice faptul ca toate caracterele din cadrul acestuia sunt minuscule.
9. Sa se compare doua valori numerice a si b si sa se aprinda cate un LED pentru fiecare din situatiile $a < b$, $a = b$ si $a > b$.
10. Sa se verifice daca doua texte contin aceleasi caractere, in aceeasi ordine, indiferent de tipul de litere (minuscule sau majuscule).

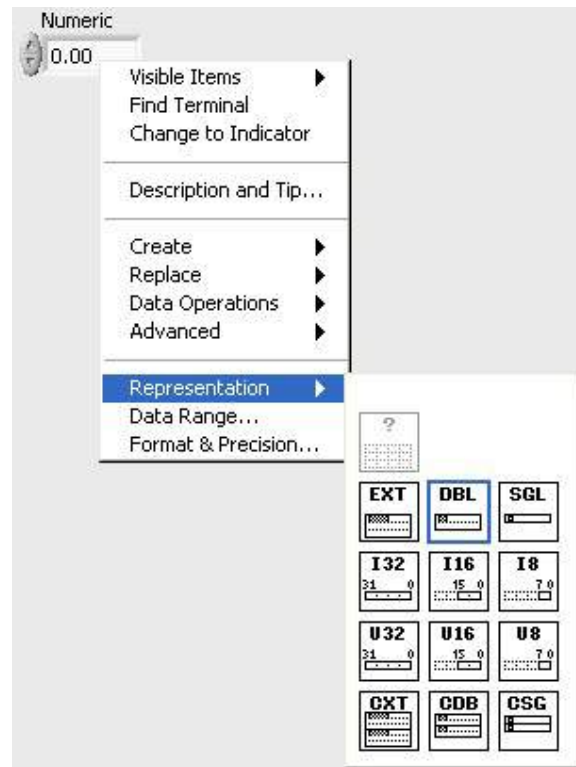
ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 8

11. Fiind dat un text ce reprezinta un nume si un prenume separate printr-un spatiu, sa se obtina din acesta un text continand cele doua initiale, fiecare urmata de cate un punct.
12. Fiind dat un text, sa se determine numarul de caractere din acesta diferite de spatiu (blanc).
13. Fiind dat un text, sa se verifice faptul ca acesta nu contine spatii libere
14. Fiind dat un text ce reprezinta un nume si un prenume separate printr-un spatiu, sa se separe in doua texte, unul reprezentand numele si unul prenumele
15. Fiind dat un text ce nu contine spatii, sa se verifice daca acesta contine intre sase si opt caractere.
16. Sa se realizeze un program care sa aprinda un LED atunci cand utilizatorul introduce o valoare mai mare decat 5.
17. Sa se realizeze un program care, in functie de pozitia unui comutator boolean, sa calculeze fie suma fie produsul a doua valori numerice introduse de catre utilizator.
18. Fiind date doua valori numerice reale a si b, sa se calculeze suma dintre dublul valorii mai mari si triplul valorii mai mici.
19. Fiind dat un text ce reprezinta o propozitie, sa se verifice faptul ca punctul (.) apare o singura data, la sfarsitul acesteia.
20. Sa se afiseze, in elemente indicatoare distincte, numerele de ore, minute si secunde scurse de la pornirea calculatorului.
21. Sa se afiseze, in elemente indicatoare distincte, numerele de zile, ore, minute si secunde scurse de la 1 ianuarie 1904, ora 12.00.
22. Sa se afiseze, in diverse formate, un text reprezentand data si timpul curente ale calculatorului.
23. Sa se realizeze un program in care utilizatorul sa introduca trei valori numerice (a, b si c) ordonate crescator.
In functie de cum valorile respective au fost introduse corect (ordonate crescator) sau nu, programul sa afiseze o fereastra de dialog cu textul **RASPUNS CORECT** sau **RASPUNS INCORECT**.
24. Sa se realizeze un program care sa aprinda sau nu un indicator **Round LED**, in functie de raspunsul **DA** sau **NU** pe care utilizatorul il da la aparitia unei ferestre de dialog cu intrebarea **Aprindem becul ?**

5. Meniuri proprii

Meniurile proprii ale elementelor de control sau indicatoare numerice contin o optiune specifica, **Representation**, cu ajutorul careia se specifica modul in care valoarea numerica respectiva este stocata in memorie.



In submeniul **Representation**, utilizatorul are la dispozitie 12 optiuni:
>>> pe prima linie: optiuni pentru numere **reale**
>>> pe a doua linie: optiuni pentru numere **intregi**
>>> pe a treia linie: optiuni pentru numere **naturale**
>>> pe a patra linie: optiuni pentru numere **complexe**

Intre diversele optiuni de pe o linie, diferenta consta in **numarul de octeti (bytes)** alocati in memorie pentru stocarea valorii respective:

EXT (extended) = numar real cu precizie extinsa (16 octeti)

DBL (double) = numar real cu precizie dubla (8 octeti)

SGL (single) = numar real cu precizie simpla (4 octeti)

I32 (long) = numar intreg pe 4 octeti

I16 (word) = numar intreg pe 2 octeti

I8 (byte) = numar intreg pe 1 octet

U32 (unsigned long) = numar natural pe 4 octeti

U16 (unsigned word) = numar natural pe 2 octeti

U8 (unsigned byte) = numar natural pe 1 octet

CXT (complex extended) = numar complex cu precizie extinsa (32 octeti)

CDB (complex double) = numar complex cu precizie dubla (16 octeti)

CSG (complex single) = numar complex cu precizie simpla (8 octeti)

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

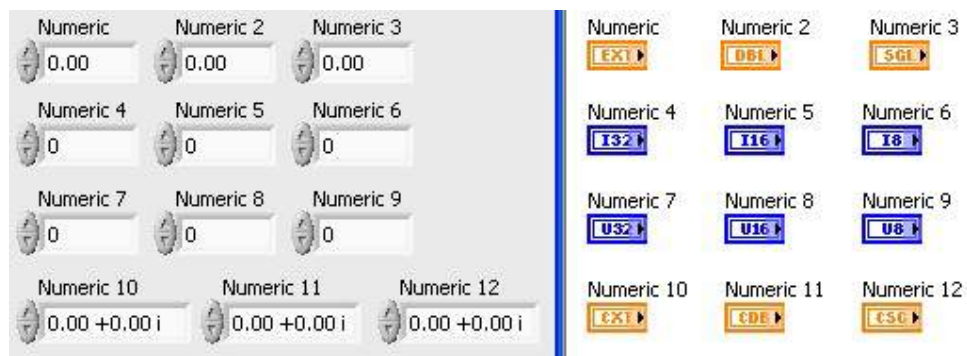
Laborator 8

Numarul de octeti alocati pentru stocare se reflecta in **intervalul de valori** in care poate exista valoarea respectiva sau in **precizia** acesteia:

Reprezentare	Interval de variatie	Nr. de zecimale
EXT si CXT	(-1,19e+4932; -6,48e-4966) si (6,48e-4966; 1,19e+4932)	15 ... 33, functie de sistemul de operare
DBL si CDB	(-1,79e+308; -4,94e-324) si (4,94e-324; 1,79e+308)	15
SGL si CSG	(-3,40e+38; -1,40e-45) si (1,40e-45; 3,40e+38)	6
I32	-2.147.483.648 ... 2.147.483.647	-
I16	-32.768 ... 32.767	-
I8	-128 ... 127	-
U32	0 ... 4.294.967.295	-
U16	0 ... 65.535	-
U8	0 ... 255	-

Textul din interiorul terminalului unui element de control sau indicator numeric se modifica pentru a reflecta modul de reprezentare ales.

In cazul in care valoarea numerica este una intreaga sau naturala, terminalul isi modifica de asemenea culoarea, din **portocaliu** in **albastru**



EXERCITII

Exersati modificarea modului de reprezentare a unui element de control numeric. Observati in panou aparitia sau disparitia zecimalelor iar in diagrama modificarea culorii si textului din interiorul terminalului.

Observati ce se intampla in situatia in care, avand un element de control numeric cu valoare reala negativa, se schimba modul de reprezentare al acestuia in numar natural.

5.1 Meniuri proprii specifice elementelor numerice

Elementele de control sau indicatoare numerice dispun in meniul propriu de doua optiuni specifice:

- **Data Range...** pentru stabilirea unui interval de variatie particular.
- **Format & Precision...** pentru specificarea unui mod particular de afisare a valorii.

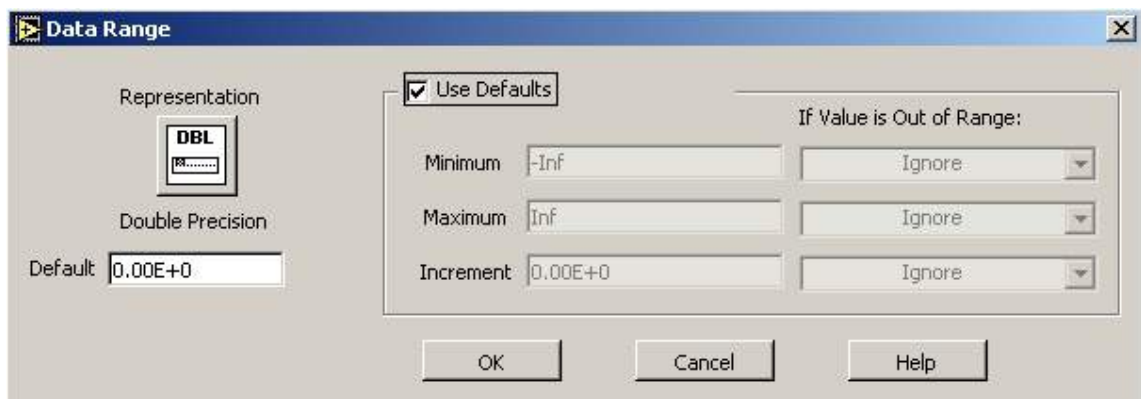
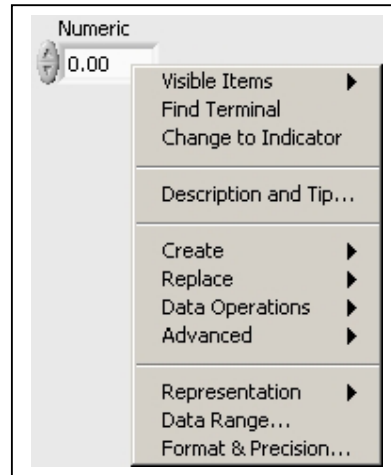
Selectarea optiunii **Data Range...** conduce la deschiderea unei ferestre suplimentare. Daca se dezactiveaza optiunea **Use Defaults**, utilizatorul poate specifica valorile minima si maxima pe care le poate lua elementul respectiv, precum si incrementul intre doua valori succesive.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 8

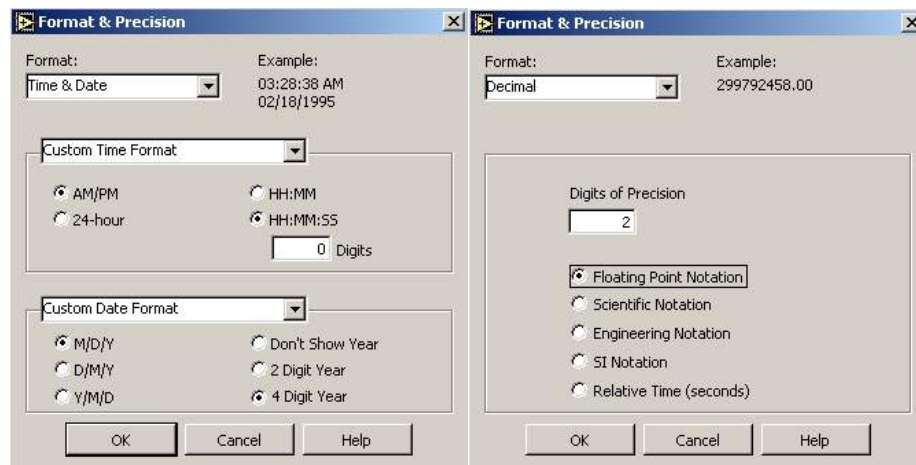
De asemenea, prin optiunea **If Value is Out of Range**, utilizatorul poate dicta modul de comportament in cazul in care se incearca introducerea unei valori in afara intervalului stabilit: daca in locul optiunii **Ignore** se selecteaza optiunea **Coerce**, elementul va forta ramanerea valorii intre limitele impuse.

Fereastra suplimentara **Data Range** permite modificarea si din acest loc a modului de reprezentare a valorii numerice (**Representation**).



Selectarea din meniul propriu a optiunii **Format & Precision** deschide o fereastră din care utilizatorul poate dicta afisarea valorii numerice in format:

- **zecimal**, caz in care se pot specifica numarul de zecimale si notatia (cu virgula mobila, stiintifica, ingineriasca etc.)
- **timp - data**, situatie in care se pot selecta diverse formate de afisare. In cazul in care se selecteaza formatul **timp - data**, valoarea elementului va reprezenta numarul de secunde scurse de la data de 1 ianuarie 1904, ora 3:00 AM.



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 8

In situatiile in care elementul de control sau indicator numeric are un mod de reprezentare de tipul numar intreg sau natural, optiunea **Format & Precision** permite suplimentar si afisarea valorilor in sistem de numeratie hexazecimal, octal sau binar.

EXERCITIUL

Disponeti pe panou un element de control numeric de tipul **Horizontal Pointer Slide** si modificati valorile **Minimum**, **Maximum** si **Increment** din optiunea **Data Range...** a meniului propriu.

Incercati apoi sa modificati valoarea elementului de control in afara limitelor impuse si observati comportamentul elementului.

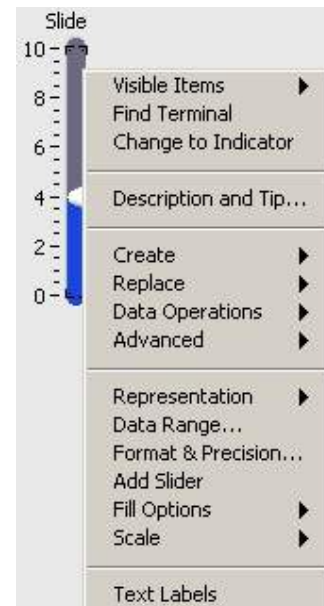
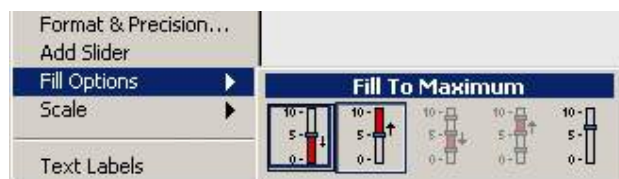
Disponeti pe panou un element de control numeric de tipul **Digital Control** si selectati optiunea **Format & Precision** din meniul propriu al acestuia. Pentru formatul **zecimal**, modificati numarul de zecimale si notatia (stiintifica, ingineriasca etc) si observati consecintele.

Selectati apoi formatul **timp - data** cu diverse formate de afisare si observati consecintele.

5.2 Meniuri proprii specifice elementelor cu scala

Elementele de control sau indicatoare numerice cu scala (potentiometre liniare sau circulare, indicatoare cu ac etc) dispun in meniul propriu de optiuni suplimentare referitoare la modul de afisare:

- optiunea **Fill Options** permite alegerea modului in care sa se coloreze portiunile elementului aflate de o parte si de alta a cursorului (colorat deasupra, colorat dedesubt, necolorat etc)



- optiunea **Scale** dispune de o serie de submeniuri proprii pentru pozitionarea uniforma sau arbitrara a gradatiilor de pe scala (**Marker Spacing 4**), pentru stabilirea formatului de afisare a valorilor de pe scala (**Format & Precision...**), pentru alegerea variantei de densitate a gradatiilor (**Style**) si pentru selectarea unei scale liniare sau logaritmice (**Mapping**)

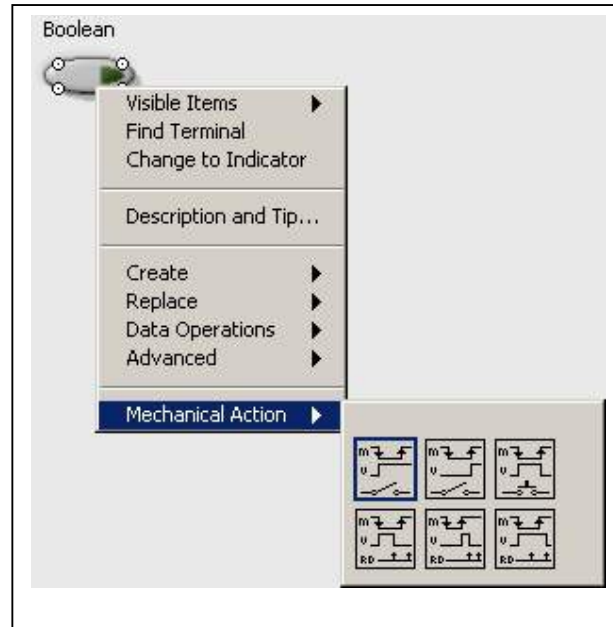
EXERCITII

Disponeti pe panou un element de tipul **Vertical Pointer Slide**, selectati din optiunea **Fill Options** a meniului sau propriu diverse moduri de umplere si observati consecintele.

Pentru acelasi element, afisati scala cu gradatii arbitrare (**Marker Spacing**), afisati valorile de pe scala cu o zecimala (**Format & Precision**), modificati densitatea gradatiilor de pe scala (**Scale**), apoi transformati tipul scalei in logaritmic (**Mapping**).

5.3 Meniuri proprii specifice elementelor booleene

Elementele de control booleene dispun in meniul propriu de optiunea suplimentara **Mechanical Action**, aceasta permitand stabilirea modului in care elementul se comporta la apasarea butonului mouse-ului de catre utilizator.



- **Switch When Pressed** modifica valoarea logica a elementului atunci cand se apasa butonul mouse-ului, iar valoarea ramane modificata si dupa eliberarea butonului
- **Switch When Released** modifica valoarea logica a elementului atunci cand se elibereaza butonul mouse-ului (dupa ce in prealabil a fost apasat), iar valoarea ramane modificata si dupa eliberare
- **Switch Until Released** modifica valoarea logica a elementului atunci cand se apasa butonul mouse-ului, iar dupa eliberarea butonului valoarea elementului redevine cea dinaintea apasarii
- **Latch When Pressed** modifica valoarea logica a elementului atunci cand se apasa butonul mouse-ului, dar aceasta modificare dureaza **doar pana cand valoarea logica este transmisa in diagrama pe fluxul de date**. Imediat ce valoarea elementului a fost transmisa, acesta revine la valoarea dinaintea apasarii, chiar daca utilizatorul continua sa tina butonul mouse-ului apasat.
- **Latch When Released** modifica valoarea logica a elementului atunci cand se elibereaza butonul mouse-ului (dupa ce in prealabil a fost apasat), dar aceasta modificare dureaza doar pana cand valoarea logica este transmisa in diagrama pe fluxul de date. Dupa ce valoarea a fost transmisa, elementul revine la valoarea initiala.
- **Latch Until Released** modifica valoarea logica a elementului atunci cand se apasa butonul mouse-ului, iar modificarea se pastreaza atat timp cat utilizatorul tine butonul apasat. Dupa ce utilizatorul elibereaza butonul, elementul revine la valoarea initiala doar in momentul in care se realizeaza o transmitere a valorii pe fluxul de date.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 8

EXERCITII

Dispuneti pe panou un element boolean de control de tipul **Vertical Toggle Switch** si un element indicator de tipul **Round LED**. In diagrama, conectati intre ele terminalele celor doua elemente.

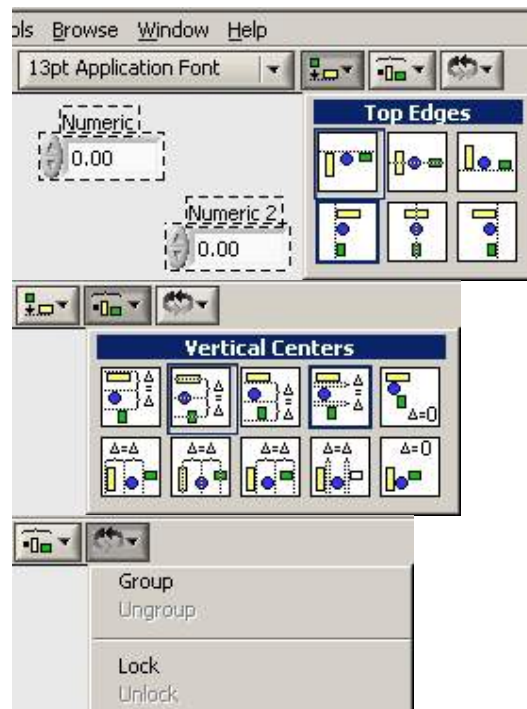
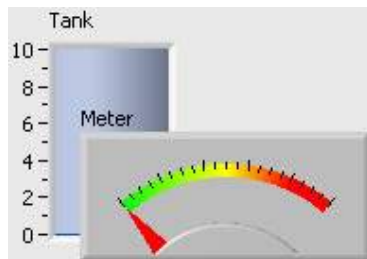
Pentru fiecare optiune **Mechanical Option** a elementului de control, efectuati o rulare continua (butonul **Run Continuously**) si observati modul in care se comporta comutatorul si modul in care valoarea sa logica este transmisa elementului **Round LED**.

Dati exemple de situatii in care fiecare optiune ar fi utila (de exemplu, optiunea **Latch When Pressed** este utila atunci cand, in urma apasarii butonului de catre utilizator, se deruleaza o actiune cu un timp foarte scurt si nu se doreste ca aceasta actiune sa fie repetata chiar daca utilizatorul intarzie sa ridice degetul de pe butonul mouse-ului).

5.4 Ordonarea componentelor

In situatia in care se doreste alinierea mai multor elemente de pe panou sau a mai multor componente ale diagramei, se selecteaza obiectele respective apoi se deschide **meniul de aliniere** si se alege varianta dorita:

- aliniere dupa o orizontala a limitelor superioare, centrelor sau limitelor inferioare
- aliniere dupa o verticala a limitelor din stanga, centrelor sau limitelor din dreapta



Atunci cand mai multe elemente sunt suprapuse, alegerea din **meniul de ordonare** a unei optiuni are drept urmare schimbarea pozitiei "in adancime" a elementelor selectate: **Move Forward** = un pas inainte, **Move Backward** = un pas inapoi, **Move To Front** = trecere in planul cel mai apropiat, **Move To Back** = trecere in planul cel mai indepartat.

Selectarea mai multor componente si alegerea optiunii **Group** are drept efect selectarea intregului grup atunci cand una dintre componente este selectata ("desfiintarea" grupului se realizeaza cu optiunea **Ungroup**). Selectarea uneia sau mai

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 8

multor componente si alegerea optiunii **Lock** nu mai permite deplasarea sau modificarea dimensiunilor componentelor respective (anularea efectului se realizeaza cu optiunea **Unlock**)

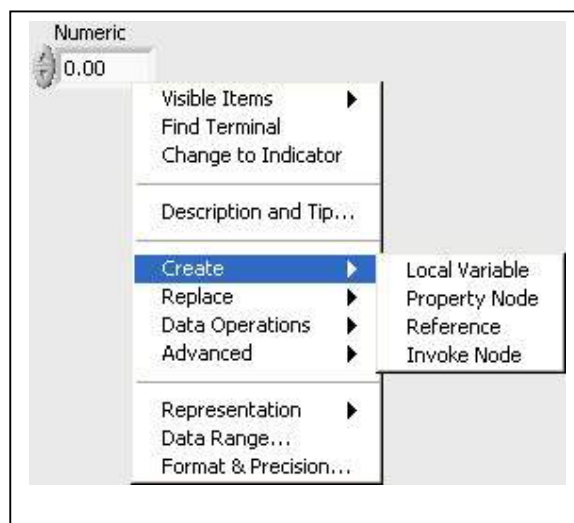
EXERCITIU

Disponeti elemente de control si indicatoare pe panoul programului si exersati diverse variante de aliniere si distribuire.

Suprapuneti unele dintre elemente si exersati optiunile de schimbare a pozitiei "in adancime"

Exersati optiunile de grupare a mai multor elemente si de blocare a mutarii si dimensionarii.

5.5. Variabile locale



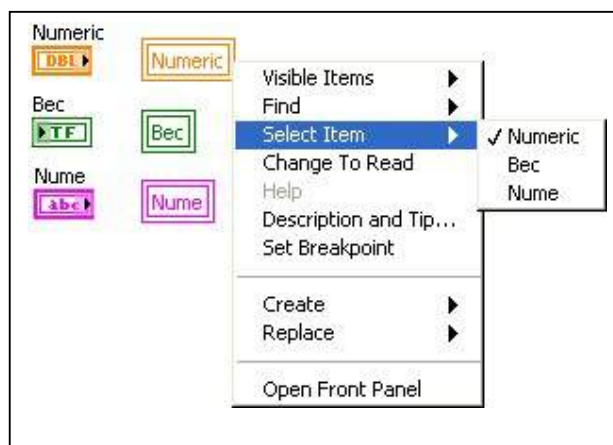
Selectarea, din meniul **Create** al unui element de control sau indicator, a optiunii **Local Variable**, va conduce la introducerea in diagrama a unei entitati suplimentare, numita **variabila locala**.

O variabila locala reprezinta o **instanta** a elementului corespunzator: o **dublura** a acelui element.

Daca, la un moment dat, un element isi modifica valoarea, toate variabilele sale locale, indiferent de locul in care sunt dispuse in diagrama,

capata valoarea corespunzatoare.

Variabilele locale sunt utilizate in situatia in care este complicat (sau imposibil) sa se efectueze o legatura direct la terminalul unui element (de exemplu, atunci cand legatura ar trebui efectuata de-a lungul unei zone intinse din diagrama, zone in care sunt deja multe componente). Un element poate avea mai multe variabile locale.



Initial o variabila locala este o dublura a elementului din al carui meniu propriu a fost creata, numele acelui element fiind inscris in simbolul variabilei. Terminalul variabilei locale are culoarea corespunzatoare tipului de date pe care le manipuleaza. Elementul pe care variabila locala il reprezinta poate fi schimbat.

Pentru aceasta, din meniul propriu al variabilei locale,

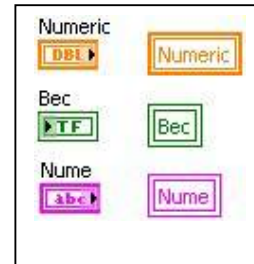
optiunea **Select Item**, se selecteaza numele noului element pe care variabila locala il va reprezenta.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 8

In mod implicit, atunci cand este creata, indiferent ca elementul pe care il reprezinta este unul de control sau indicator, o variabila locala este un **element indicator** (contur subtire).

In aceasta situatie, valoarea care ajunge la variabila locala este automat capatata de elementul pe care il reprezinta (aceasta este si o modalitate de a schimba prin program valorile elementelor de control).



Daca, din meniul propriu al variabilei locale, se selecteaza optiunea **Change To Read**, variabila locala devine **element de control** (contur ingrosat). In aceasta situatie, variabila locala va genera valoarea elementului pe care il reprezinta.

EXERCITIU

In panoul unei aplicatii, dispuneti un element boolean de control de tipul **Vertical Toggle Switch**, unul de tipul **Slide Switch** si un indicator boolean **Round LED**.

Creati o variabila locala a elementului **Vertical Toggle Switch**. In diagrama, conectati terminalul elementului **Slide Switch** atat la terminalul elementului **Round LED** cat si la variabila locala a elementului **Vertical Toggle Switch**.

Rulati programul cu optiunea **Run Continuously**, actionati comutatorul **Slide Switch**, observati si explicati comportamentul elementului **Vertical Toggle Switch**.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

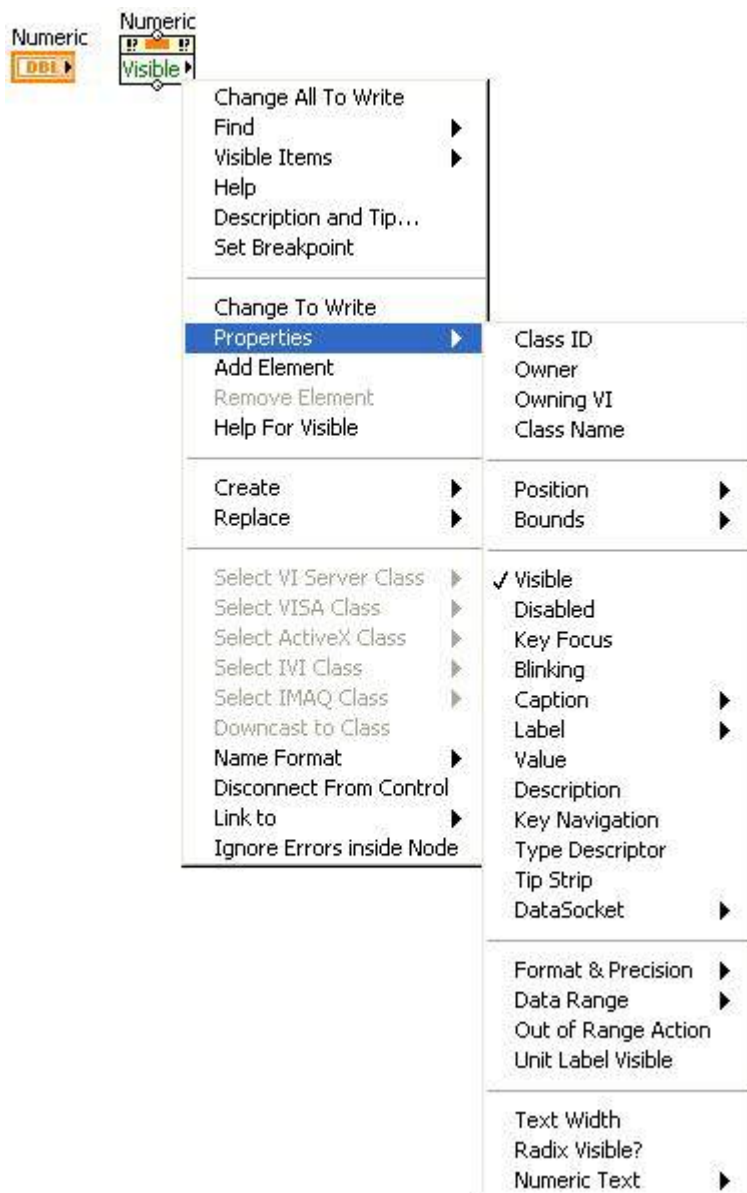
Laborator 9

5.6 Noduri de proprietati

Selectarea, din meniul **Create** al unui element de control sau indicator, a optiunii **Property Node**, va conduce la introducerea in diagrama a unei entitati suplimentare, numita **nod de proprietati**.

Nodul de proprietati al unui element reprezinta o componenta prin intermediul careia, in timpul executiei programului, pot fi determinate (in sens de "citire") sau modificate automat (scrise) anumite proprietati ale elementului respectiv.

In mod implicit, atunci cand este creat, un nod de proprietati **citeste** starea logica (afisat sau ascuns) a **vizibilitatii** elementului pe care il reprezinta. Deschizand meniul propriu al nodului de proprietati, optiunea **Properties**, se poate selecta proprietatea pe care nodul o reprezinta.



Printre cele mai des utilizate proprietati se numara:

- **Position:** pozitia elementului pe panou, exprimata in pixeli pe orizontala si pe verticala
- **Bounds:** dimensiunile elementului, exprimate in pixeli
- **Visible:** daca aceasta proprietate are valoarea False, elementul nu mai este afisat in panou (desi continua sa existe si are terminal in diagrama)
- **Disabled:** valoarea 0 semnifica faptul ca utilizatorul poate actiona acel element, valoarea 1 faptul ca elementul nu poate fi actionat (este dezactivat), valoarea 2 faptul ca elementul este dezactivat si afisat in culori "sterse"

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 9

- **Blinking**: atunci cand aceasta proprietate are valoarea True, elementul este afisat "clipind"
- **Label**: proprietati de formatare a textului din eticheta

Un nod de proprietati poate fi trecut din starea implicita de **citire** in starea de **scriere**, prin selectarea optiunii **Change To Write** din meniul sau propriu. In starea de scriere, un nod de proprietati poate primi valori care vor modifica proprietatile elementului corespunzator.

Un nod de proprietati poate fi dimensionat, pentru a gestiona mai multe proprietati ale elementului pe care il reprezinta. Atunci cand contine mai multe proprietati, acestea pot fi in mod independent de scriere sau de citire. Elementul reprezentat de catre un nod de proprietati poate fi schimbat prin selectarea din meniul propriu al nodului a optiunii **Link To**, similar optiunii **Select Item** din cazul variabilelor locale. In diagrama pot exista mai multe noduri de proprietati ale aceluiasi element.



EXERCITII

Disponeti in panoul unei noi aplicatii un element de control boolean de tipul **Vertical Toggle Switch** si un element de control numeric. Creati un nod de proprietati al elementului numeric, treceti nodul respectiv in starea de scriere si conectati la el terminalul elementului boolean. Rulati programul cu optiunea **Run Continuously**, actionati comutatorul boolean, observati si explicati efectul.

In aceeaasi aplicatie, dispuneti pe panou un element de tipul **String Control**, creati un nod de proprietati pentru acesta, treceti nodul respectiv in starea de scriere, selectati pentru el proprietatea **Disabled** si legati la nod terminalul elementului de control numeric. Rulati programul cu optiunea **Run Continuously** si actionati comutatorul boolean astfel incat elementul de control numeric sa fie vizibil. Introduceti in elementul numeric valoarea **0** si introduceti un text in elementul **String**.

Introduceti in elementul numeric valoarea **1** si incercati sa modificati textul din elementul **String**. Introduceti in elementul numeric valoarea **2**, observati si explicati efectul.

TEMA

1. Construiti un program pentru determinarea solutiilor unei ecuatii de gradul al doilea. Pentru afisarea radacinilor, programul va dispune de doua perechi de indicatoare numerice separate: una cu mod de reprezentare real, cealalta cu mod de reprezentare complex.

Fiecare din cele doua perechi de elemente indicatoare va fi vizibila sau nu, in functie de natura radacinilor ecuatiei.

2. Construiti un program in care utilizatorul sa poata alege, prin intermediul unui comutator boolean, tipul de ecuatie pe care doreste sa o rezolve: de gradul I sau de gradul al II-lea.

Elementul de control numeric de la care se va introduce coeficientul lui x^2 va fi vizibil doar in cazul ecuatiei de gradul al II-lea. In cazul ecuatiei de gradul I, elementul respectiv va deveni invizibil si va capata valoarea 0.

3. In aceleasi conditii ca la tema nr. 2, atunci cand se opteaza pentru rezolvarea unei ecuatii de gradul I, elementul de control numeric pentru coeficientul lui x^2 sa capete valoarea 0, sa fie dezactivat (utilizatorul sa nu mai poata modifica valoarea) si sa fie afisat in culori sterse.

4. Sa se realizeze un program pentru determinarea solutiilor unei ecuatii de gradul al doilea in care elementele de control pentru introducerea coeficientilor ecuatiei sa aiba valoarea implicita NaN. In cazul in care utilizatorul nu introduce o valoare numerica in unul din elementele respective, elementul sa clipeasca pentru a semnaliza eroarea.

5. Sa se realizeze un program in care, prin intermediul a doua elemente de control numerice (un **Vertical Pointer Slide** si un **Horizontal Pointer Slide**), utilizatorul sa poata specifica pozitia in panou a unui element indicator **Round LED**.

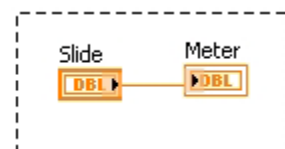
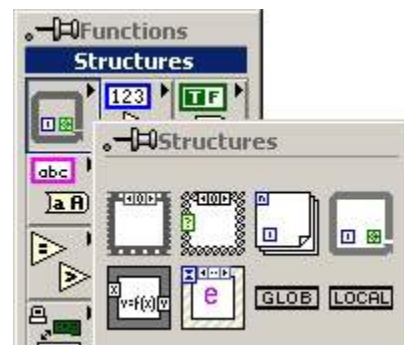
6. Structuri de programare

6.1 Inserarea structurilor in diagrama

Structurile de programare sunt componente ale diagramei ale caror simboluri se gasesc in subpaleta **Structures** a paletei de functii.

In diagrama, o structura de programare urmeaza sa contina o serie de terminale de elemente, simboluri de functii si valori constante ce vor forma un flux de date in interiorul structurii.

Daca, in momentul in care structura este aleasa din subpaleta **Structures**, o parte din elementele pe care urmeaza sa le contina se afla deja dispuse in diagrama, chenarul structurii poate fi trasat in jurul componentelor respective (dupa selectarea din subpaleta, se apasa butonul stang al mouse-ului in unul din colturile viitorului chenar si se deplaseaza mouse-ul tinand butonul apasat: pe masura ce mouse-ul este deplasat, se traseaza automat un contur cu linie intrerupta, contur ce va fi inlocuit de catre structura in momentul eliberarii butonului mouse-ului).



Dupa dispunerea in diagrama, conturul structurii poate fi deplasat sau dimensionat. Atunci cand o structura este deplasata, toate elementele din interiorul sau se deplaseaza odata cu ea.

Elemente din afara structurii pot fi mutate in interiorul acesteia, iar elemente din interior pot fi mutate in afara structurii.

Daca, prin dimensionare sau deplasare, o structura ajunge sa se suprapuna cu alte elemente din diagrama, **acestea nu vor fi incluse** in structura

EXERCITIU

Exersati inserarea unei structurii oarecare (de exemplu **Sequence**) in diagrama, trasand de la inceput chenarul sau in jurul unui grup de elemente.

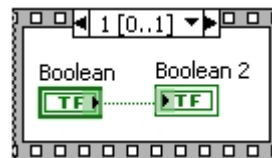
Mutati elemente din structura in afara sa si aduceti elemente din exterior in interiorul structurii.

Deplasati si dimensionati structura.

6.2 Structura secventiala (Sequence)

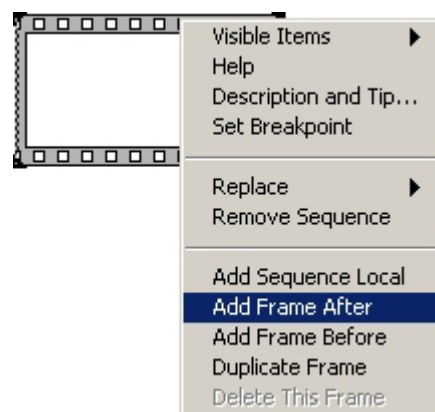
Atunci cand dispune de mai multe ferestre, fiecare dintre acestea cu propriul flux de date, o structura **Sequence** executa in ordine ferestrele respective.

De exemplu, in figurile de mai jos sunt reprezentate doua ferestre ale unei structurii **Sequence**. Structura va executa intai fereastra cu indicele **0**, trimittandu-se valoarea de la elementul de control **Slide** la elementul indicator **Meter**, apoi va executa fereastra **1** ce are ca efect transmiterea valorii booleene intre cele doua elemente.



Atunci cand este dispusa in diagrama, o structura **Sequence** contine o singura fereastra, aceasta fiind totodata si fereastra curenta (vizibila).

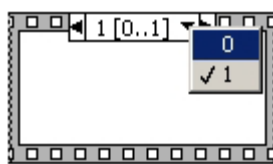
Deschizand meniul propriu al structurii si alegand una din optiunile **Add Frame After** sau **Add Frame Before**, se poate adauga o fereastra dupa sau inaintea celei curente.



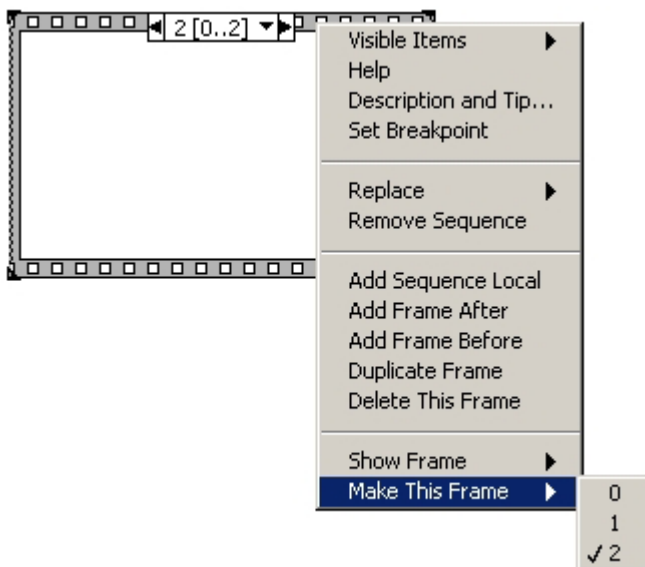
Daca o structura **Sequence** contine mai mult de o fereastra, atunci ferestrele primesc **indici** (numere de ordine) incepand cu **0**, iar pe latura superioara a conturului structurii apare un **selector** prin intermediul caruia se poate trece de la o fereastra la alta.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 9



Trecerea la o alta fereastră se poate realiza fie prin intermediul celor doua sageti de la capetele selectorului, fie apasand pe sageata din partea dreapta (indreptata in jos) si selectand indicele ferestrei dorite din lista care se desfasoara. La un moment dat, este vizibil doar fluxul din fereastră curenta a unei structuri **Sequence**.



Se poate crea o dublura a ferestrei curente, selectand din meniul propriu al structurii optiunea **Duplicate Frame**.

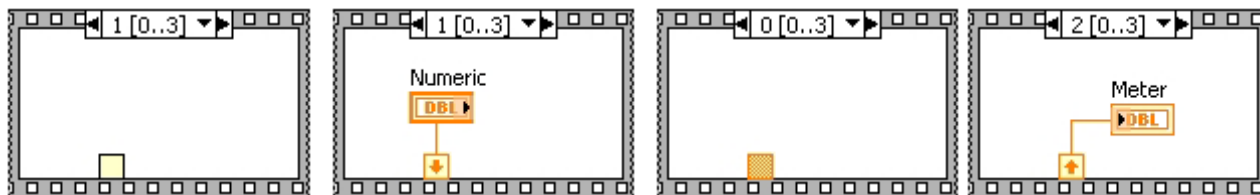
Fereastră curenta poate fi stearsa, cu optiunea **Delete This Frame**.

Atunci cand o structura **Sequence** contine cel putin doua ferestre, ordinea acestora poate fi schimbata: se modifica indicele ferestrei curente, selectand optiunea **Make This Frame**, apoi noul indice.

Atunci cand se doreste transferul unei valori intre doua ferestre ale unei structuri **Sequence**, prin

selectarea optiunii **Add Sequence Local** se dispune pe conturul structurii o **variabila locala** a acesteia (sub forma unui patrat galben).

Daca, in una din ferestrele structurii, se conecteaza o valoare la variabila locala, fereastră respectiva devine **fereastră sursa** pentru acea variabila. In acea fereastră, simbolul variabilei locale contine o fereastră indreptata spre exteriorul structurii. In ferestrele anterioare ferestrei sursa, simbolul variabilei locale devine hasurat, semn ca valoarea variabilei locale nu este accesibila in acele ferestre.



In ferestrele de dupa fereastră sursa, simbolul variabilei locale va contine o sageata indreptata spre interiorul structurii. In aceste ferestre, valoarea transferata prin intermediul variabilei locale poate fi utilizata in fluxul de date. O structura **Sequence** poate dispune de mai multe variabile locale.

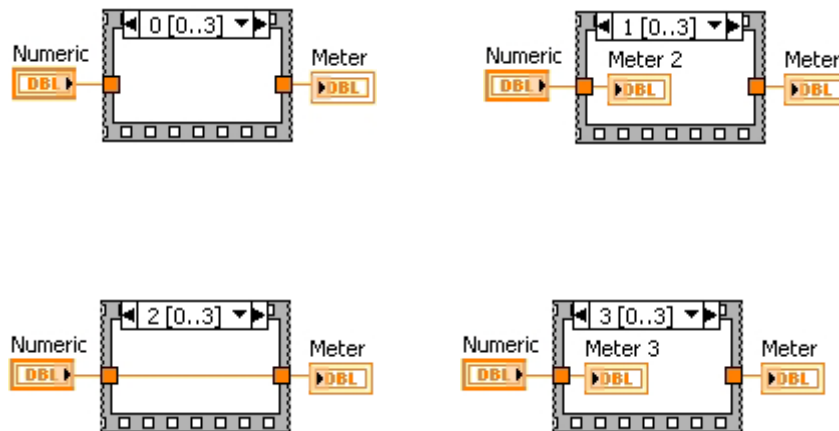
ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 9

Intr-o structura **Sequence** pot fi utilizate valori provenite din fluxul de date exterior. O valoare introdusa din exterior intr-o structura **Sequence** va putea fi utilizata in oricare dintre ferestrele structurii.

Daca, intr-o anumita fereasta, se face o legatura din structura **Sequence** in exteriorul acesteia, la iesirea respectiva (patratul de pe contur) nu va mai putea fi legata o alta valoare din alta fereasta (ar exista o incertitudine privind valoarea care iese din structura).

Indiferent de indicele ferestrei in care s-a facut o legatura spre exterior, valoarea va parasii iesirea de pe conturul structurii **Sequence** doar dupa executarea ultimei ferestre.



EXERCITIU

Introduceti in diagrama o structura **Sequence** cu patru ferestre. Pornind de la un element de control numeric al carui terminal este exterior structurii, calculati in ferestre succesive dublul, patratal, radacina patrata si inversa ($1/x$) valorii numerice respective.

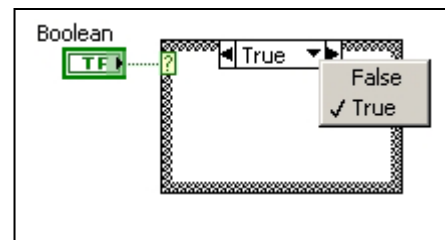
Valoarea calculata in fiecare fereasta a structurii va fi afisata intr-un element indicator separat.

Adaugati structurii o a cincea fereasta si, utilizand variabile locale, calculati si afisati in aceasta ultima fereasta suma valorilor calculate in ferestrele anterioare.

6.3 Structura cauzala (Case)

Atunci cand este dispusa in diagrama, o structura **Case** contine doua ferestre (**True** si **False**), fiecare dintre acestea cu propriul flux de date. Pe conturul structurii se afla dispus un **terminal selector** (de culoare verde, continand semnul intrebarii).

Structura **Case** va executa doar una din cele doua ferestre, in functie de valoarea booleana (**True** sau



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 9

False) conectata la terminalul sau selector.

Daca la terminalul selector al unei structuri **Case**, in locul unei valori booleene, se conecteaza o valoare numerica (numar intreg sau natural), numele ferestrelor se vor modifica automat din **True** si **False** in **0** si **1**. Structura va executa fereastra al carei nume va coincide cu valoarea numerica ce ajunge la terminalul sau selector.

In aceasta situatie, are sens ca structura **Case** sa contina mai mult de doua ferestre. Deoarece structura **Case** nu va putea contine atatea ferestre cate valori numerice pot ajunge la terminalul sau selector, una dintre ferestre trebuie declarata drept **fereastră implicita (Default)**.

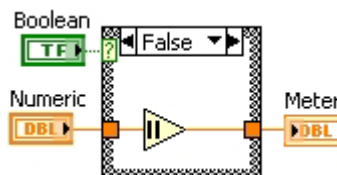
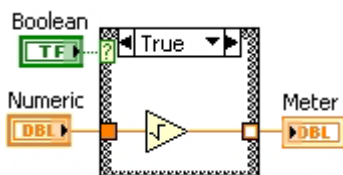
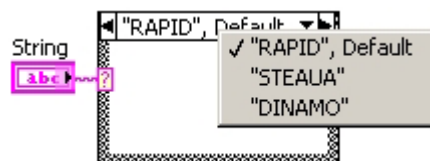
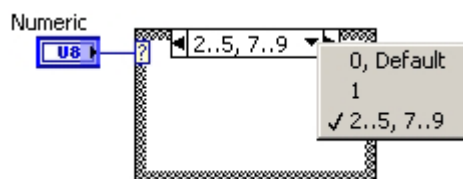
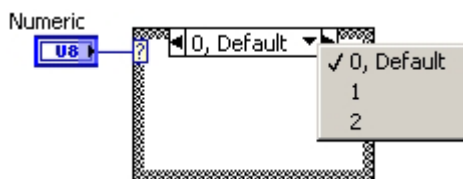
Fereastră implicita va fi executata atunci cand valorii ajunse la terminalul selector nu ii corespunde nici un nume de fereastră. Un nume de fereastră poate contine si descrierea unei multimi de valori, de genul 2..5, 7..9, intelegand prin aceasta ca fereastră respectiva va fi executata daca la terminalul selector ajunge una din valorile 2, 3, 4, 5, 7, 8 sau 9.

O valoare nu poate exista in numele mai multor ferestre (ar exista o incertitudine privind fereastră ce trebuie executata).

Daca la terminalul selector al unei structuri **Case** se conecteaza o valoare alfanumerica, atunci numele ferestrelor vor trebui sa fie de tipul **string** (se modifica utilizand unealta de editare texte). Si in aceasta situatie trebuie sa existe o fereastră implicita.

Se pot de asemenea defini multi de valori (de genul "A".. "F"), cu conditia evitarii dublurilor.

O valoare introdusa din exterior intr-o structura **Case** este accesibila in orice fereastră a acesteia.



Daca dintr-o fereastră a unei structuri **Case** se scoate o valoare in exterior, pe conturul structurii apare un **terminal de iesire** de culoare alba, semn ca diagrama contine o eroare. Atunci cand o structura **Case** are pe contur un terminal de iesire, la acesta trebuie definite legaturi **din toate ferestrele structurii**. In caz contrar, se spune ca iesirea din structura **nu este complet definita**.

Atunci cand iesirea este complet definita, terminalul de iesire se umple cu culoarea reprezentand tipul de date conectate (pentru a fi corect definita, la o iesire trebuie sa ajunga **acelasi tip de date** din toate ferestrele structurii).

EXERCITI

1. Introdueți în diagrama o structură **Case** la al carei terminal selector conectați terminalul unui element de control boolean. În funcție de valoarea elementului de control boolean, la terminalul unui element indicator de tip **String** din afara structurii, trimiteți textul **ADEVARAT** sau **FALS**.

2. Introdueți în diagrama o structură **Case** la al carei terminal selector conectați terminalul unui element de control numeric pentru valori naturale. În funcție de valoarea elementului de control numeric, la terminalul unui element indicator de tip **String** din afara structurii, trimiteți textul **CIFRA PARA**, **CIFRA IMPARA** sau **NUMAR**.

3. Introdueți în diagrama o structură **Case** la al carei terminal selector conectați terminalul unui element de control alfanumeric.

În funcție de primul caracter din elementul de control, la terminalul unui element indicator de tip **String** din afara structurii, trimiteți textul **PRIMA PARTE** (pentru primul caracter de la A la M), **A DOUA PARTE** (pentru primul caracter de la N la Z) sau **NU INCEPE CU LITERA**.

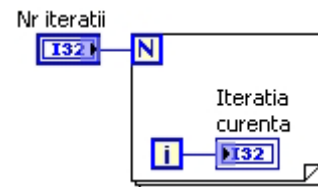
6.4 Structura repetitiva cu numar fix de iteratii (bucla For)

O buclă **For** execută de un număr fixat de ori porțiunea de diagramă pe care o conține. O execuție a porțiunii interne de diagramă poartă numele de **iteratie**.

O buclă **For** dispune în colțul stnga - sus de un terminal (notat cu **N**), la care trebuie legată o valoare care să specifice numărul de iterații pe care buclă urmează să le execute.

În interiorul buclei se află un **terminal numărător** (notat cu **i**) care generează o valoare naturală reprezentând **indicele iterației curente** (aflate în curs de execuție).

Indicii iterațiilor sunt numerotați începând cu valoarea **0**, astfel încât ultima iterație executată de către buclă **For** va avea indicele **N-1**.



EXERCITI

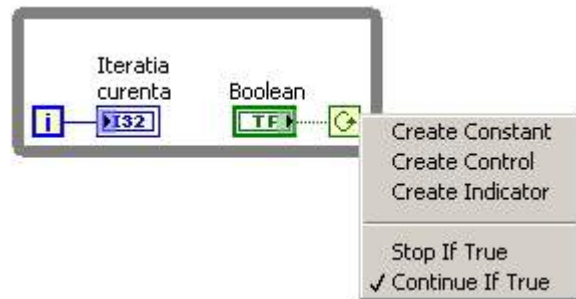
Introdueți în diagramă o buclă **For**.

Conectați terminalul unui element de control numeric la terminalul **N** al buclei și terminalul unui element indicator numeric la terminalul numărător al buclei. Introdueți în buclă o funcție de așteptare, astfel încât fiecare iterație a buclei să se efectueze în interval de două secunde.

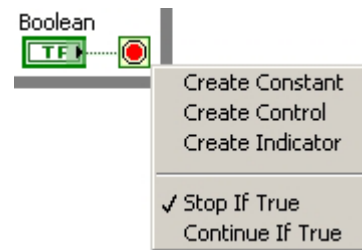
Rulați programul și urmăriți în panou modificarea indicelui iterației curente.

6.5 Structura repetitiva cu conditie de terminare (buclo While)

O bucla **While** executa portiunea de diagrama pe care o contine **pana cand la terminalul de continuare** (aflat in coltul dreapta-jos) apare o anumita valoare booleana.



Atunci cand o bucla **While** este dispusa in diagrama, terminalul sau de continuare se afla in starea implicita **Continue If True**. Buclo va continua sa execute iteratii succesive atat timp cat la terminalul sau de continuare ajunge o valoare logica **True**. Daca la terminalul de continuare ajunge o valoare logica **False**, bucla **While** nu mai trece la urmatoarea iteratie ci isi **incheie executia**.



Starea implicita a terminalului de continuare poate fi modificata. Terminalul poate fi trecut in starea **Stop If True** fie cu unealta de operare, fie apasand butonul drept al mouse-ului si selectand starea din meniul propriu.

In starea **Stop If True** bucla **While** isi va incheia executia atunci cand la terminalul de continuare ajunge valoarea logica **True**.

In coltul stanga - jos, bucla **While** dispune de un **terminal numarator** (notat cu **i**) care genereaza o valoare naturala reprezentand **indicele iteratiei curente** (aflate in curs de executie). Indicii iteratiilor sunt numerotati incepand cu valoarea **0**.

EXERCITIU

Introduceti in diagrama o bucla **While**.

Conectati terminalul unui element de control boolean de tipul **Stop Button** la terminalul de continuare al buclei si terminalul unui element indicator numeric la terminalul numarator al buclei.

Treceti terminalul de continuare al buclei in starea **Stop If True**. Introduceti in bucla o functie de asteptare, astfel incat fiecare iteratie a buclei sa se efectueze in interval de o secunda.

Rulati programul si urmariti in panou modificarea indicelui iteratiei curente. Apasati butonul **STOP** pentru a incheia executia buclei **While**.

6.6 Registri de transfer in structuri repetitive

Registrii de transfer sunt componente ale structurilor repetitive ce pot fi utilizate pentru **transferul unor valori între doua sau mai multe iteratii succesive** ale structurii. Pentru a adauga un registru de transfer unei structuri repetitive (buclo **For** sau **While**), se selecteaza optiunea **Add Shift Register** din meniul propriu al structurii.

Un registru de transfer contine doua **terminale** (stang si drept) dispuse pe conturul structurii. O valoare conectata la terminalul drept al registrului va "apare" in terminalul stang la iteratia urmatoare.

Cu alte cuvinte, terminalul stang introduce in iteratia curenta valoarea ce a fost transmisa la terminalul drept in iteratia anterioara.

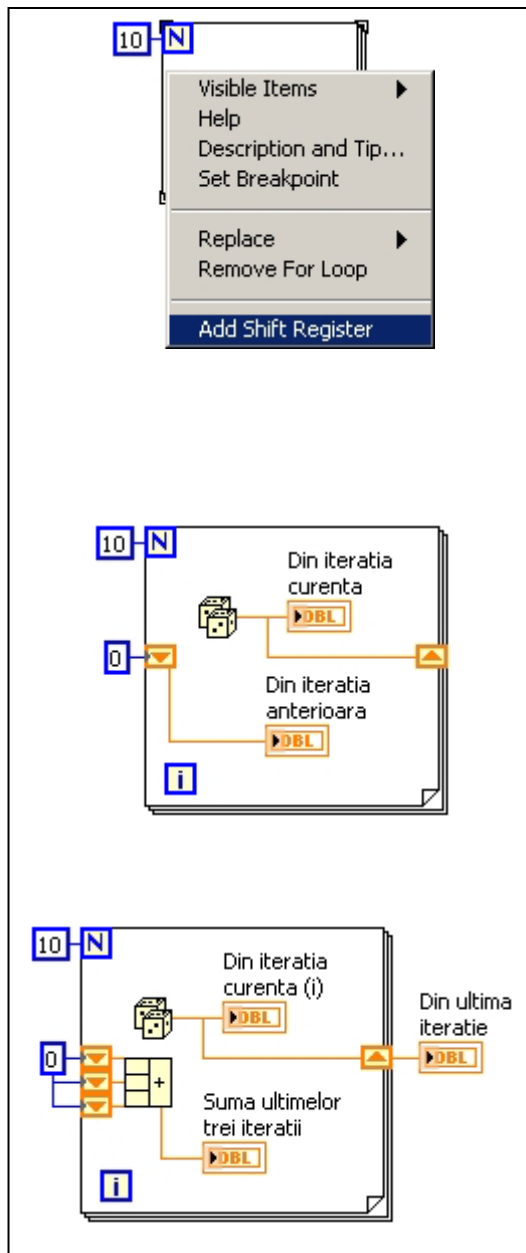
La executarea primei iteratii a unei bucle, nu exista inca nici un transfer efectuat de catre registru între cele doua terminale ale sale. Terminalul stang al registrului nu a fost inca "alimentat" cu o valoare din iteratia precedenta. Pentru a se putea efectua extragerea unei valori din terminalul stang la prima iteratie, se conecteaza la acest terminal o valoare din exteriorul buclei. Se spune ca registrul de transfer se **initializeaza**.

Terminalul stang al unui registrului de transfer poate fi **dimensionat**, capatand astfel mai multe **componente**.

Intr-o astfel de situatie, valoarea transferata din iteratia anterioara se afla in componenta superioara.

De sus in jos, fiecare componenta contine cate o valoare provenita dintr-o iteratie mai indepartata. Dupa executarea ultimei iteratii a buclei, din terminalul drept al registrului se poate extrage ultima valoare transferata acestuia.

O structura repetitiva poate contine mai multi registri de transfer, fiecare cu una sau mai multe componente ale terminalului stang si fiecare transferand între iteratii valori de diverse tipuri.



EXERCITIU

Introduceti in diagrama o bucla **While** si adaugati-i acesteia doi registri de transfer. Prin primul dintre registri transferati o valoare numerica aleatoare **X**, iar prin cel de-al doilea o valoare booleana, obtinuta ca rezultat al evaluarii expresiei **X > 0,5**. Initializati cei doi registri de transfer cu valori corespunzatoare.

Disponeti pe panou elemente indicatoare pentru afisarea valorilor din iteratia curenta si din cea precedenta, precum si un buton la a carui apasare executia buclei **While** sa se incheie. Introduceti in bucla **While** o functie care sa determine o asteptare de trei secunde la fiecare iteratie.

Apasati butonul **Run**, apoi treceti in diagrama, apasati butonul **Highlight Execution** si urmariti modul in care valorile sunt transferate între iteratii.

TEMA

1. Sa se realizeze un program care sa aprinda un **LED** pentru trei secunde, sa-l stinga pentru o secunda, apoi sa repete ciclul pâna la apasarea de catre utilizator a unui buton **STOP**.
2. Construiti un program care sa simuleze functionarea unui semafor, aprinzand succesiv cate un **LED** rosu, galben sau verde.
3. **LED**-urile rosu si verde vor sta aprinse cate 10 secunde, **LED**-ul galben va sta aprins doua secunde. Programul va rula pana la apasarea de catre utilizator a unui buton **STOP**.
4. Sa se realizeze un program care sa numere descrescator de la 20 la zero, la interval de o secunda.
5. Sa se realizeze un program care sa numere crescator, din unu în unu, la interval de o secunda, pâna când utilizatorul apasa un buton **STOP**.
6. Sa se realizeze un program care, în momentul în care utilizatorul introduce o valoare numerica mai mare decât cinci, sa aprinda un **LED** si sa se opreasca.
7. Sa se realizeze un program care sa genereze continuu numere aleatoare, la interval de 0,5 secunde, pâna când valoarea generata x îndeplineste conditia $x > 0,97$.
8. Sa se realizeze un program care sa afiseze, in milisecunde, timpul scurs intre apasarile de catre utilizator a doua butoane.
9. Sa se realizeze un program care sa permita introducerea de catre utilizator a unor valori numerice reale a si b , sa astepte apasarea unui buton **OK** si apoi sa aprinda un **LED** daca este îndeplinita conditia $a < b$.

10. Realizati un program care sa efectueze urmatoarele:

- sa afiseze doua numere naturale, generate aleator intre 1 si 10;
- sa astepte pana in momentul in care utilizatorul, dupa ce a introdus intr-un element de control valoarea produsului celor doua numere, apasa un buton **OK**;
- sa compare valoarea introdusa de catre utilizator cu produsul calculat al celor doua numere aleatoare si sa aprinda un **LED** daca raspunsul a fost corect;
- -daca raspunsul a fost incorect, sa afiseze valoarea corecta;
- sa afiseze timpul necesar utilizatorului pentru a da raspunsul;
- sa afiseze numarul total de intrebari, procentajul raspunsurilor corecte si timpul mediu de raspuns;
- -sa ruleze pana la apasarea de catre utilizator a unui buton **STOP**.

7. Functii pentru valori vectoriale (Array)

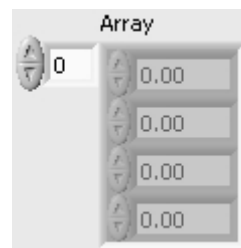
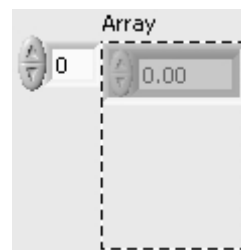
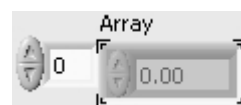
7.1 Definirea valorilor si a numarului de valori vizibile

Atunci cand este dispus in panou si ii este definit tipul de date, un element de tip **Array** afiseaza la un moment dat o singura valoare.

Pentru a modifica numarul de valori afisate, se pozitioneaza intai cursorul mouse-ului pe un colt al elementului **Array**, astfel incat pe colturile acestuia sa apara patru simboluri "echer" . La aparitia celor patru simboluri "echer", se apasa butonul stang al mouse-ului si, tinandu-l apasat, se deplaseaza mouse-ul astfel incat sa se redimensioneze conturul elementului **Array**. Pe masura ce mouse-ul este deplasat, conturul redimensionat al elementului **Array** este marcat cu linie intrerupta. Atunci cand conturul a fost redimensionat astfel incat sa poata cuprinde numarul dorit de valori, se elibereaza butonul mouse-ului si elementul **Array** este afisat redimensionat. Redimensionarea conturului se poate efectua atat pe verticala cat si pe orizontala.

In partea sa stanga, un element de tip **Array** dispune de un **index**.

Indexul unui element **Array** indica **numarul de ordine** al valorii afisate in pozitia superioara (**indicele** valorii). Indicele primei valori dintr-un **Array** este 0.



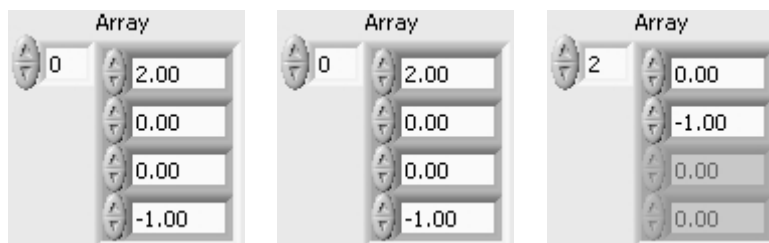
Imediat dupa ce i se defineste tipul de date, un **Array** nu contine nici o valoare (este **vid**). Faptul este semnalizat prin afisarea "opaca" a elementelor pe care **Array**-ul le contine.

Atunci cand se defineste o valoare din **Array**, aceasta este afisata normal. Daca, la un moment dat, utilizatorul defineste o valoare si exista valori cu indici mai mici inca nedefinite, acestea vor capata automat o valoare implicita, in functie de tip.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 9

Cand se modifica valoarea indexului unui **Array**, valorile acestuia se deplaseaza astfel incat in pozitia superioara sa fie afisata valoarea cu indexul selectat. Daca, in urma acestei deplasari, unele valori nu mai sunt vizibile, faptul nu echivaleaza cu stergerea acestora: valorile continua sa existe in cadrul **Array**-ului si pot fi vizualizate prin modificarea corespunzatoare a indexului.



EXERCITIU

Modificati numerele de valori vizibile ale celor trei **Array**-uri create in exercitiul anterior.

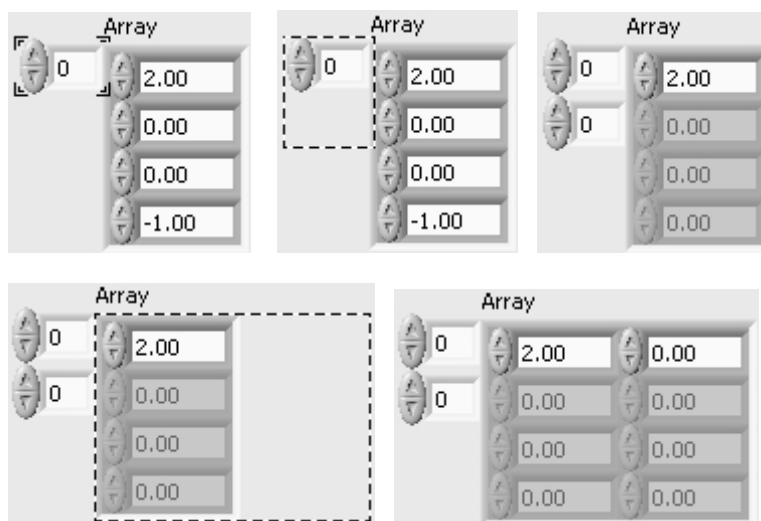
Transformati **Array**-urile in elemente de control (daca este cazul) si definiti valori in cadrul acestora.

Modificati indexul fiecarui **Array** si observati modul in care valorile se deplaseaza.

7.2 Definirea numarului de dimensiuni

In mod implicit, atunci cand este creat, un **Array** are o singura dimensiune, putand fi descris drept o matrice coloana.

Numarul de dimensiuni ale unui **Array** poate fi modificat prin dimensionarea indexului acestuia: se pozitioneaza cursorul mouse-ului pe un colt al indexului, astfel incat pe colturile acestuia sa apara patru simboluri "echer", se apasa butonul mouse-ului si se deplaseaza mouse-ul tinand butonul apasat, pana cand indexul capata dimensiunea dorita. Daca un **Array** are mai mult de o dimensiune, se poate stabili afisarea mai multor linii si mai multor coloane.

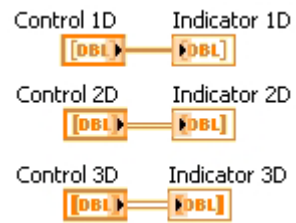


ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 9

Terminalul unui element **Array** se caracterizeaza prin prezenta unor paranteze drepte a caror grosime este proportionala cu numarul de dimensiuni. Atunci cand se realizeaza legaturi in diagrama, conexiunile pe care circula un **Array** cu o dimensiune sunt mai groase decat cele pe care circula valori scalare. Pentru **Array**-urile cu doua sau mai multe dimensiuni, conexiunile sunt reprezentate cu linie dubla.

Spatiul intermediar al liniei duble este cu atat mai mare cu cat numarul de dimensiuni ale **Array**-ului este mai mare.



EXERCITIU

Mariti numerele de dimensiuni ale **Array**-urilor create in exercitiile anterioare si afisati mai multe linii si mai multe coloane.

Observati ca, atunci cand un **Array** cu o dimensiune este transformat in unul cu doua dimensiuni (matrice), valorile existente initial in **Array** sunt distribuite pe prima linie a matricei.

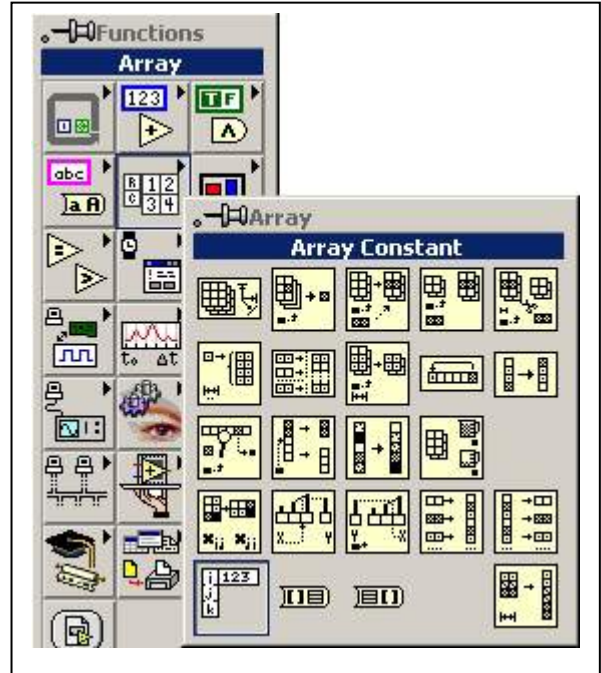
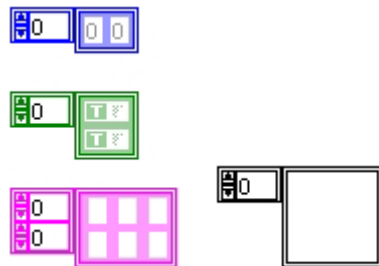
Observati comportamentul unei matrici la introducerea unei valori anterior nedefinite. Modificati indexurile unei matrici si observati modul de deplasare a valorilor

7.3 Construirea Array-urilor in diagrama

Atunci cand este necesara existenta in diagrama a unei constante **Array**, aceasta poate fi selectata din meniul cu functii pentru valori vectoriale (**Array**) al paletii de functii.

La dispunerea sa in diagrama, similar elementelor **Array** de pe panou, o astfel de constanta nu are definit tipul de date pe care le va contine.

Definirea tipului de date se realizeaza prin introducerea in chenarul constantei **Array** a unei constante de tipul dorit (numeric, boolean sau alfanumeric).



O constanta **Array** permite aceleasi operatii de definire a numarului de valori vizibile sau de modificare a numarului de dimensiuni ca si elementele **Array** din panou.

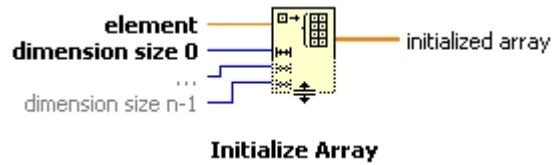
Pentru a obtine in mod dinamic (programatic), in diagrama, structuri de valori de tipul **Array**, pot fi utilizate o serie de functii specifice.

Funcția **Initialize Array** genereaza un **Array** cu una sau mai multe dimensiuni, continand aceeasi valoare (conectata la intrarea **element**) in toate pozitiile. Numarul de valori de-a lungul unei dimensiuni se specifica la intrarea **dimension size** a functiei.

Laborator 10

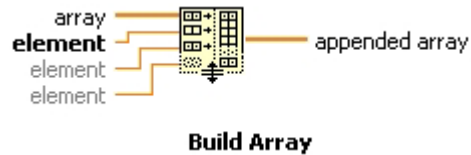
Atunci cand este dispusa in diagrama, functia are o singura intrare **dimension size**, generand astfel un **Array** cu o singura dimensiune. Daca se doreste generarea unui **Array** cu mai multe dimensiuni, simbolul functiei poate fi dimensionat.

Daca la una dintre intrarile **dimension size** se conecteaza valoarea **0**, sau daca toate intrarile respective sunt neconectate, functia va genera un **Array** vid.



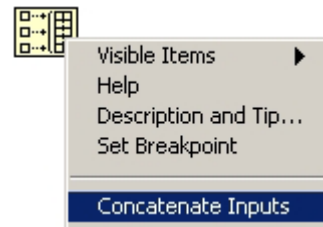
Functia **Build Array** concateneaza la un **Array** una sau mai multe elemente cu un numar de dimensiuni imediat inferior (adauga valori scalare la un **Array** cu o dimensiune, adauga linii la matrici s.a.m.d.).

Atunci cand este dispusa in diagrama, functia are o singura intrare. Numarul dorit de intrari se obtine prin dimensionarea simbolului functiei.



Daca la toate intrarile functiei sunt legate elemente cu acelasi numar de dimensiuni, functia poate opera in doua moduri, dupa cum optiunea **Concatenate Inputs** din meniul sau propriu este sau nu activata:

- daca optiunea este activata, functia va concatena intrarile si va obtine un **Array** cu un numar de dimensiuni egal cu al intrarilor (optiunea nu este valabila daca toate intrarile sunt scalare);
- daca optiunea nu este activata, functia va obtine un **Array** cu un numar de dimensiuni imediat superior celui al intrarilor (**Array** cu o dimensiune din valori scalare, matrice din linii s.a.m.d.).



Functia **Interleave 1D Array** construiesc un **Array** cu o singura dimensiune prin **intreteserea** mai multor **Array**-uri avand de asemenea o singura dimensiune. **Array**-ul rezultat este format prin preluarea alternativa a cate unei valori din fiecare **Array** de intrare.

La dispunerea in diagrama, functia contine doar doua intrari, dar numarul acestora poate fi modificat prin dimensionarea simbolului.

EXERCITIU

Inserati in diagrama o constanta **Array** de valori numerice, cu o dimensiune, continand primele cinci numere naturale.

Initializati un **Array** cu o dimensiune, continand de zece ori valoarea 7. Construiti o matrice care sa contina pe prima linie rezultatul concatenarii celor doua **Array**-uri de mai sus, iar pe a doua linie rezultatul intreteserii valorilor acestora. Explicati de ce ultimele cinci valori de pe a doua linie a matricei sunt nule.

7.4 Elementele Array si structurile repetitive

Atunci cand, in diagrama, un **Array** este conectat la o structura repetitiva **For**, bucla respectiva va prelua, la fiecare iteratie a sa, doar cate o valoare din **Array**. Valoarea preluata la o iteratie este cea cu indicele din **Array** egal cu indicele iteratiei buclei.

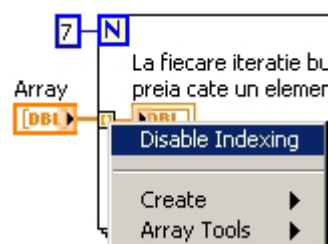
In aceasta situatie, nu mai este necesara conectarea unei valori la terminalul **N** al buclei. Aceasta va executa atatea iteratii cate valori sunt in **Array**-ul de intrare.



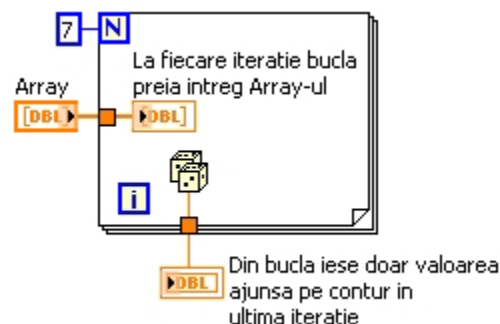
Daca se conecteaza totusi o valoare la terminalul **N**, bucla **For** va executa un numar de iteratii egal cu minimul dintre acea valoare si numarul de valori din **Array**.

Daca, din interiorul unei bucle **For**, o valoare scalara este conectata la conturul acesteia, atunci cand bucla isi incheie ultima iteratie, valorile scalare ajunse pe contur in toate iteratiile vor parasi bucla grupate intr-un **Array**.

Comportamentul buclei la intrarea unui **Array** poate fi modificat: daca, din meniul propriu al componentei de pe conturul buclei, se alege optiunea **Disable Indexing**, la fiecare iteratie a buclei va fi preluat intreg **Array**-ul de intrare. In acest caz, este necesar ca o valoare sa fie conectata la terminalul **N** al buclei.



Daca se alege optiunea **Disable Indexing** pentru o componenta de pe contur la care sunt conectate valori din interiorul buclei, din acea componenta, dupa incheierea ultimei iteratii, va parasi bucla doar valoarea din ultima iteratie. Valorile ce au ajuns pe contur in toate celelalte iteratii vor fi pierdute.



Considerentele de mai sus sunt valabile pentru **Array**-uri de orice dimensiune:

- dintr-o matrice, se poate prelua la fiecare iteratie a buclei fie cate o linie, fie intreaga matrice;
- daca un **Array** unidimensional din interiorul buclei este conectat pe conturul acesteia, la incheierea ultimei iteratii componenta buclei va genera fie o matrice, fie doar **Array**-ul primit in ultima iteratie.

Modurile in care un **Array** se poate comporta la intrarea sau la iesirea dintr-o structura repetitiva sunt valabile si in cazul buclelor **While**, cu deosebirea ca, in cazul acestora, comportamentul implicit este inversat:

- in mod implicit, o bucla **While** va prelua la fiecare iteratie intreg **Array**-ul de intrare. Doar in cazul in care componentei de pe contur i se selecteaza, din meniul propriu, optiunea **Enable Indexing**, bucla **While** va prelua cate o valoare la fiecare iteratie.
- in mod implicit, daca o valoare din interiorul unei bucle **While** se conecteaza la conturul acesteia, componenta de pe contur va genera, dupa incheierea executiei buclei, doar valoarea primita in ultima iteratie.

Doar daca se selecteaza optiunea **Enable Indexing** a componentei de pe contur, aceasta va genera, dupa incheierea ultimei iteratii, un **Array** cuprinzand valorile primite in cadrul tuturor iteratiilor.

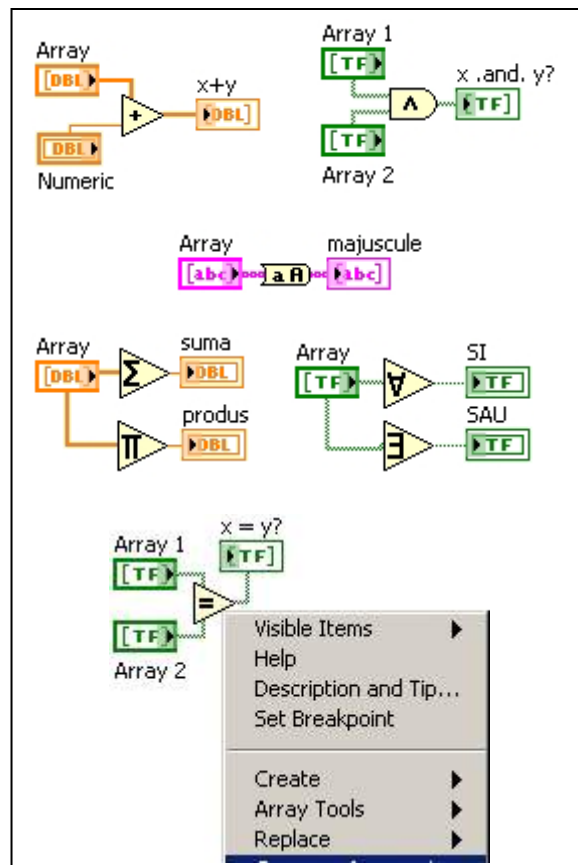
7.5 Alte functii pentru elemente Array

Funcțiile aritmetice si logice pentru valori scalare, precum si cele pentru valori alfanumerice, permit conectarea de **Array**-uri continand valori de tipurile corespunzatoare.

Se pot astfel realiza:

- operatii aritmetice asupra valorilor dintr-un **Array** (radacina patrata, functii trigonometrice etc.), intre un **Array** si o valoare scalara sau intre doua **Array**-uri;
- operatii logice intre un **Array** si o valoare booleana sau intre doua **Array**-uri cu valori booleene;
- operatii asupra **Array**-urilor de valori alfanumerice.

Funcțiile **Add Array Elements** si **Multiply Array Elements** determina suma, respectiv produsul valorilor numerice dintr-un **Array**. Funcțiile **And Array Elements** si **Or Array Elements** efectueaza operatiile logice **SI**, respectiv **SAU**, asupra tuturor valorilor booleene dintr-un **Array**, obtinand o valoare



booleana scalara.

Atunci cand au drept operanzi un **Array** si o valoare scalara sau doua **Array**-uri, functiile de comparatie genereaza un **Array** de valori booleene, compararea fiind efectuata valoare cu valoare.

Daca, din meniul propriu al functiei de comparatie, se alege optiunea **Compare Aggregates**, functia va genera o valoare booleana scalara. De exemplu, daca se compara doua **Array**-uri de valori numerice cu functia **Equal?** in starea **Compare Aggregates**, functia va genera valoarea **True** doar daca este indeplinita conditia de egalitate intre toate perechile de valori aflate pe aceleasi pozitii in cele doua **Array**-uri.

Functia **Number To Boolean Array**, aflata in meniul pentru functii booleene, transforma un numar natural intr-un **Array** de valori booleene, efectuand de fapt transformarea din baza 10 in baza 2.

Bitul cel mai putin semnificativ este reprezentat in **Array** de valoarea cu indicele 0. De exemplu, valoarea $6=0+2^1+2^2$ este transformata intr-un **Array** cu trei valori booleene: (**False**, **True**, **True**).

number ———— [] Boolean array

Number To Boolean Array

Boolean array [] ———— number

Boolean Array To Number

Functia **Boolean Array To Number** realizeaza transformarea inversa, dintr-un **Array** de valori booleene intr-un numar natural.

TEMA

1. Sa se calculeze diferenta dintre valorile maxima si minima dintr-un **Array** de valori numerice.
2. Sa se verifice prin aprinderea unui **LED** daca valoarea minima dintr-un **Array** de valori numerice apare înaintea valorii maxime.
3. Sa se ordoneze descrescator un **Array** de valori numerice.
4. Sa se calculeze suma dintre primul si ultimul element ale unui **Array** de valori numerice.
5. Sa se calculeze media elementelor ramase dupa eliminarea valorilor minima si maxima dintr-un **Array** de valori numerice.
6. Sa se realizeze un program pentru generarea unui **Array** de 50 de numere aleatoare cuprinse între **N1** si **N2** (**N1** si **N2** specificate de catre utilizator).

7. Sa se realizeze un program pentru generarea unui **Array** de 50 de valori booleene aleatoare, astfel încât probabilitatea de aparitie a valorii logice **True** sa fie 0,7.

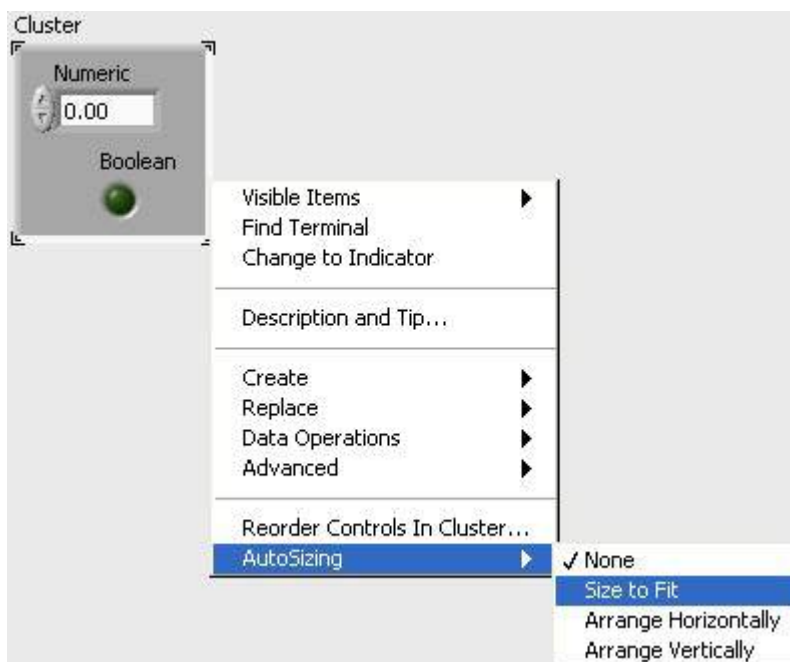
8. Meniul propriu al elementelor de tip Cluster

Submeniul **Autosizing** din meniul propriu al unui element de tipul **Cluster** permite:

- modificarea dimensiunilor chenarului, astfel incat acesta sa se "muleze" pe multimea de elemente din interior (optiunea **Size to Fit**)
- alinierea pe orizontala sau pe verticala a elementelor din interiorul chenarului

Elementele de diverse tipuri din interiorul unui **Cluster** sunt o multime ordonata. Ordinea elementelor este cea in care acestea au fost introduse in elementul **Cluster**. Optiunea **Reorder Controls In Cluster...** permite modificarea indicilor elementelor (numerelor de ordine).

La selectarea acestei optiuni, panoul aplicatiei devine negru iar programul asteapta ca utilizatorul sa indice succesiv, prin click-uri cu mouse-ul, elementele cu indicii 0, 1, 2 s.a.m.d.



Indicele pe care il va capata elementul pe care se va efectua urmatorul click este mentionat in dreptul textului **Click to set to** (initial 0). Langa fiecare element sunt afisati cate doi indici.

Indicele avut de element inainte de inceperea modificarilor este scris cu negru pe fond alb.

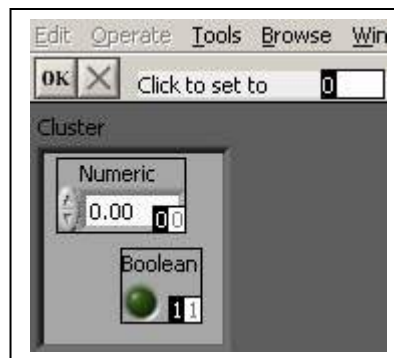
Indicele modificat este scris cu alb pe fond negru.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 10

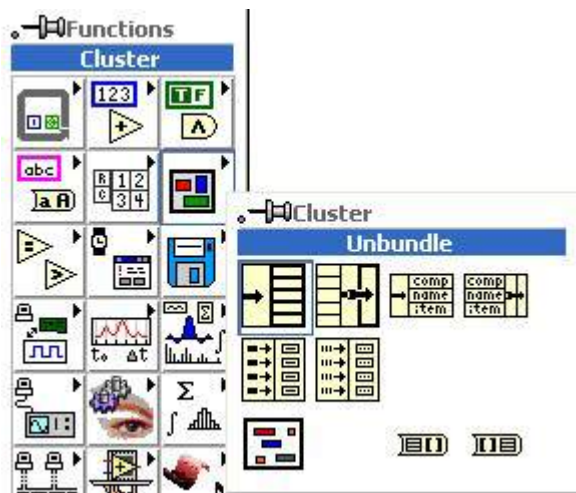
Validarea modificarilor se efectueaza prin apasarea butonului **OK**

Anularea modificarilor indicilor si revenirea la situatia anterioara se efectueaza prin apasarea butonului marcat cu **X**.



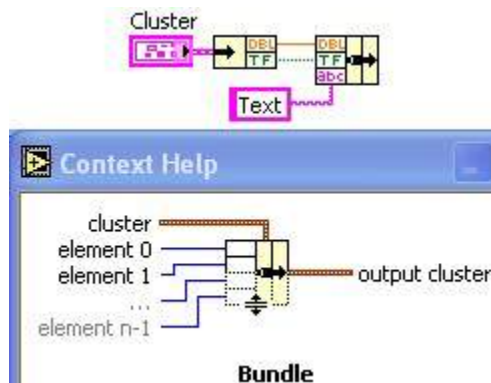
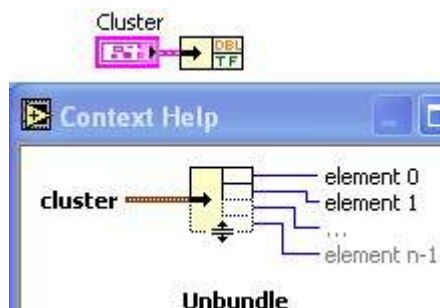
Functii pentru elemente de tip Cluster

Functiile specifice elementelor de tipul **Cluster** sunt grupate in meniul cu acelasi nume al paletii de functii.



Functia **Unbundle** permite separarea elementelor unui **Cluster**. Atunci cand o functie **Unbundle** este dispusa in diagrama, ea poseda doua iesiri si poate fi dimensionata.

Cand se realizeaza o legatura intre terminalul unui **Cluster** si o functie **Unbundle**, functia se dimensioneaza automat, corespunzator numarului de elemente din **Cluster**. Iesirile functiei capata culori si inscriptii corespunzatoare elementelor componente. Iesirea din pozitia superioara corespunde elementului cu indicele 0, cea imediat de sub ea elementului cu indicele 1 s.a.m.d.

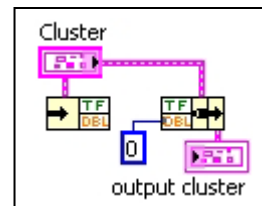


Funcția **Bundle** grupează mai multe valori (elemente) într-o structură de tip **Cluster**. În mod implicit funcția posedă două intrări dar poate fi dimensionată.

Dacă la intrarea **cluster** a funcției **Bundle** nu se conectează nimic, atunci la toate intrările sale trebuie să existe conexiuni, pentru a defini complet structura elementului **Cluster** rezultat.

Dacă la intrarea **cluster** se conectează o structură de date, simbolul funcției **Bundle** se dimensionează automat, iar intrările sale capătă culori și inscripții conform structurii de date conectate.

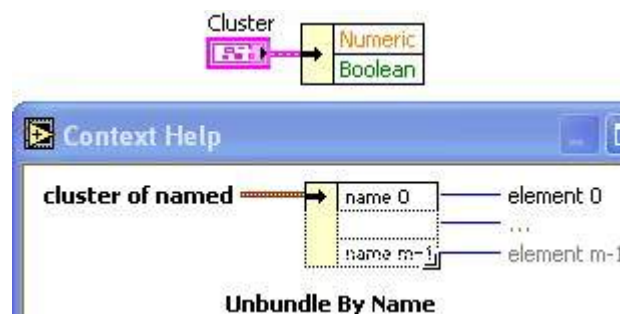
În această situație, nu mai este necesar ca la toate intrările funcției să fie conectate valori.



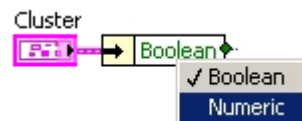
Funcția nu mai creează o nouă structură ci înlocuiește, în structura legată la intrarea **cluster**, eventualele valori conectate la unele din intrările sale.

Dacă elementele din interiorul unui **Cluster** posedă etichete (**label**), separarea acestora se poate efectua și cu ajutorul funcției **Unbundle By Name**.

Atunci când la intrarea sa se conectează un **Cluster**, ieșirile funcției **Unbundle By Name** capătă nume ce corespund etichetelor componentelor elementului **Cluster**.



Dacă funcția are mai puține intrări decât numărul elementelor din **Cluster**, numele sunt atribuite în ordinea elementelor.



Simbolul funcției **Unbundle By Name** poate fi dimensionat.

Numele atribuit unei ieșiri (și implicit elementul din **Cluster** ce se obține la ieșirea respectivă) poate fi modificat:

- făcând click cu unealta de operare pe ieșirea respectivă, apoi selectând numele dorit din meniul care se deschide;
- deschizând meniul propriu al funcției (click cu butonul drept al mouse-ului) și selectând numele dorit din submeniul **Select Item**.



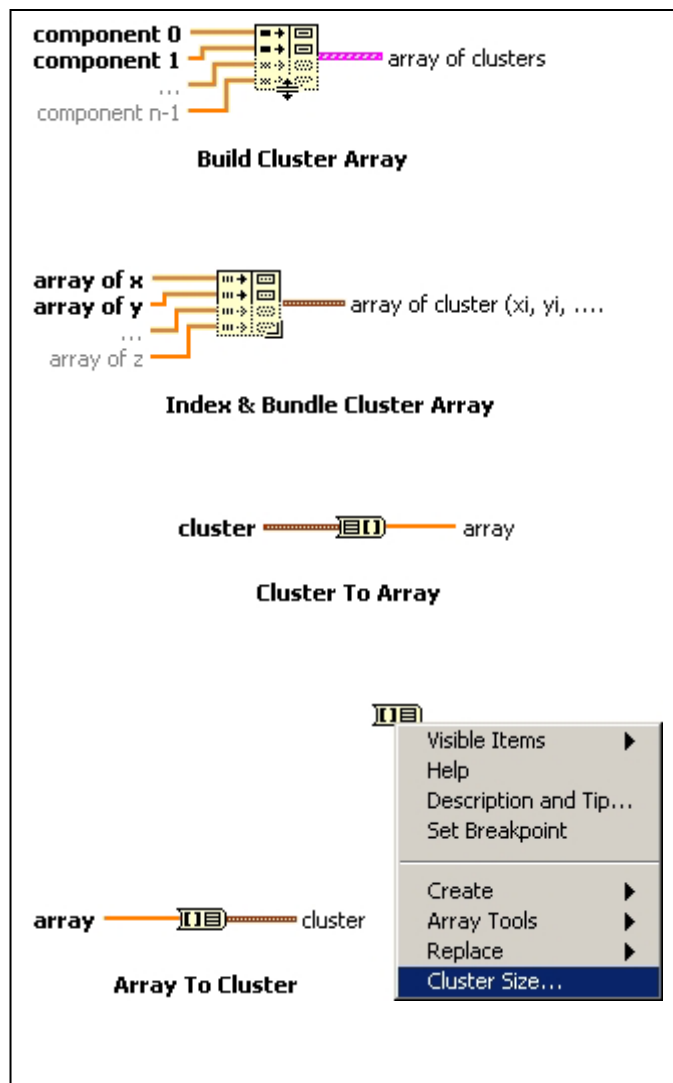
Laborator 10

Funcția **Build Cluster Array** preia un număr de elemente de tip **Array** (simbolul funcției poate fi dimensionat), introduce fiecare **Array** într-un **Cluster** și combina toate aceste elemente **Cluster** într-un element **Array** rezultat. Toate elementele **Array** de la intrare trebuie să fie de același tip.

Funcția **Index & Bundle Cluster Array** creează un **Array** de elemente **Cluster**. Elementul **Cluster** cu indicele **i** conține valorile cu indicii **i** din elementele **Array** de la intrare. Elementele **Array** de la intrare pot fi de tipuri diferite.

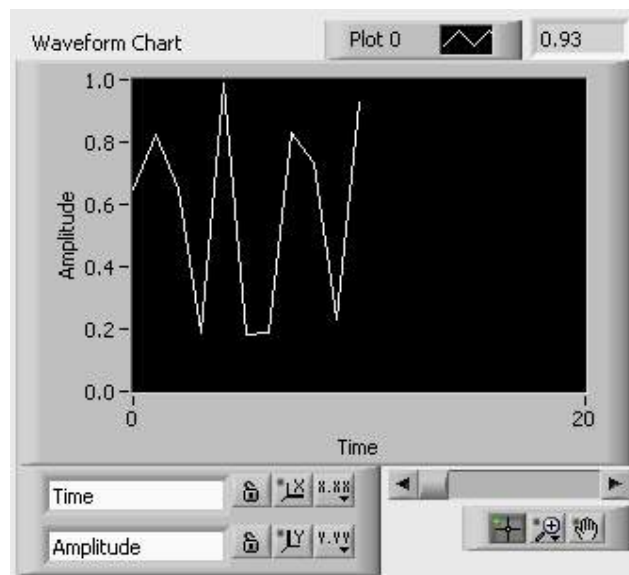
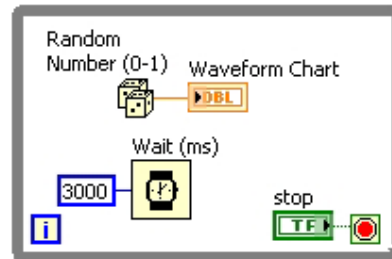
Funcția **Cluster To Array** dispune elementele dintr-un **Cluster** într-un **Array**. Elementele din **Cluster** trebuie să fie de același tip.

Funcția **Array To Cluster** dispune valori dintr-un **Array** într-un **Cluster**. Numărul de elemente din **Cluster** se stabilește selectând, din meniul propriu al funcției, opțiunea **Cluster Size...** (implicit 9). Dacă în elementul **Array** se afla mai multe valori decât numărul de elemente din **Cluster**, atunci vor fi preluate doar o parte dintre acestea și anume primele.



9. Elementul Waveform Chart

Unui element de tip **Chart** i se pot trimite in mod succesiv, punct cu punct, valorile ordonate ale punctelor pe care sa le traseze grafic. In figura alaturata, bucla **While** executa cate o iteratie la fiecare trei secunde (datorita functiei **Wait (ms)**). La fiecare iteratie, functia **Random Number (0-1)** trimite cate o valoare aleatoare la elementul **Chart**.

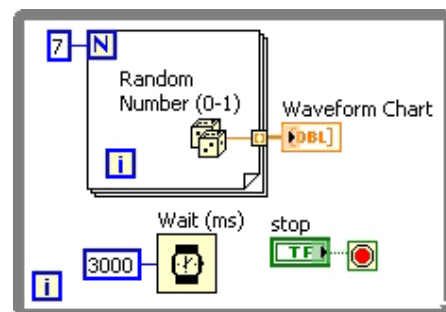


Un element **Chart** accepta nu numai valori trimise punct cu punct (scalare) ci si siruri de valori (**Array**).

Atunci cand primeste un sir de valori (considerate de asemenea tot ordonate ale punctelor), un element **Chart** adauga la sfarsitul graficului deja existent nu un singur punct ci un numar de puncte egal cu numarul de valori din sirul primit. In figura de mai jos, la fiecare iteratie a buclei **While** (la fiecare trei secunde), bucla **For** genereaza un sir (**Array**) de sapte valori aleatoare, sir ce este trimis elementului **Chart**.

La fiecare trei secunde, la graficul din elementul **Chart** sunt adaugate astfel inca sapte puncte.

Daca un element **Chart** primeste o matrice de valori numerice (**Array** cu doua dimensiuni), atunci el va trasa simultan un numar de grafice egal cu numarul de linii ale matricei.



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 11

EXERCITIU

Construiti diagramele din figurile de mai sus si verificati modul de functionare.

10. Elementul Waveform Graph

Un element de tip **Graph** nu accepta valori individuale (scalare) ci numai siruri (**Array**) de valori.

Spre deosebire de elementele **Chart**, atunci cand primeste un sir de valori, elementul **Graph** sterge graficul pe care il afisase anterior si afiseaza doar graficul format din noile puncte primite. Graficul din figura alaturata se va schimba la fiecare trei secunde. Valorile afisate anterior se vor pierde.

Un element de tip **Graph** considera de asemenea valorile succesive pe care le primeste drept ordonate ale punctelor de pe grafic.

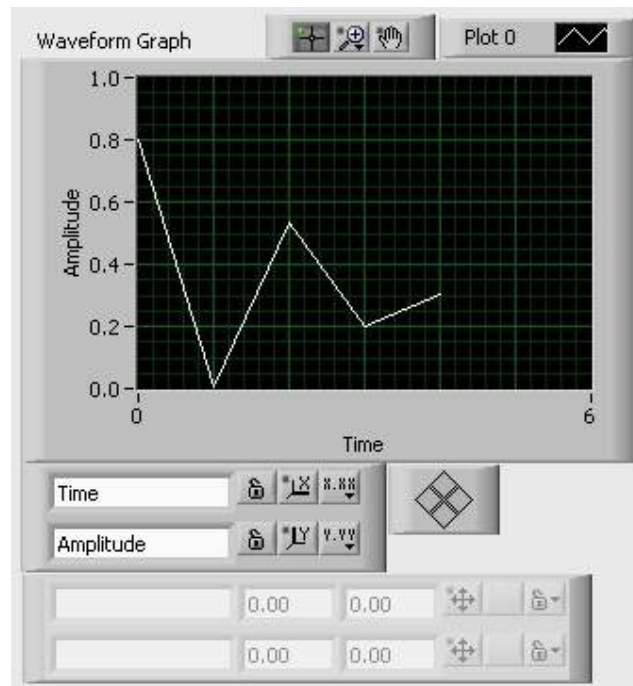
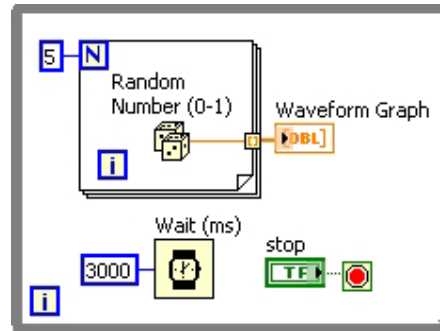
In modul implicit, abscisele punctelor sunt considerate automat crescatoare din unitate in unitate (0 pentru prima valoare primita, 1 pentru a doua s.a.m.d.).

Elementele de tip **Graph** permit insa, suplimentar, definirea abscisei primului punct si a distantei pe orizontala dintre doua puncte succesive.

In aceasta situatie, valorile x_0 si Δx se introduc (printr-o functie **Bundle**) intr-un **Cluster** impreuna cu sirul de valori ce reprezinta ordonatele punctelor, iar la terminalul elementului **Graph** se conecteaza iesirea functiei **Bundle** (figura de mai jos).

Daca un element **Graph** primeste o matrice de valori numerice (**Array** cu doua dimensiuni), atunci el va trasa simultan un numar de grafice egal cu numarul de linii ale matricei.

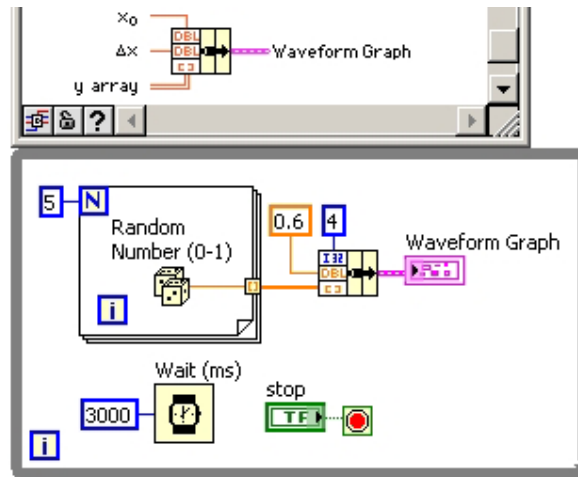
In situatia in care se traseaza mai multe grafice si se doresc definirea abscisei x_0 a



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 11

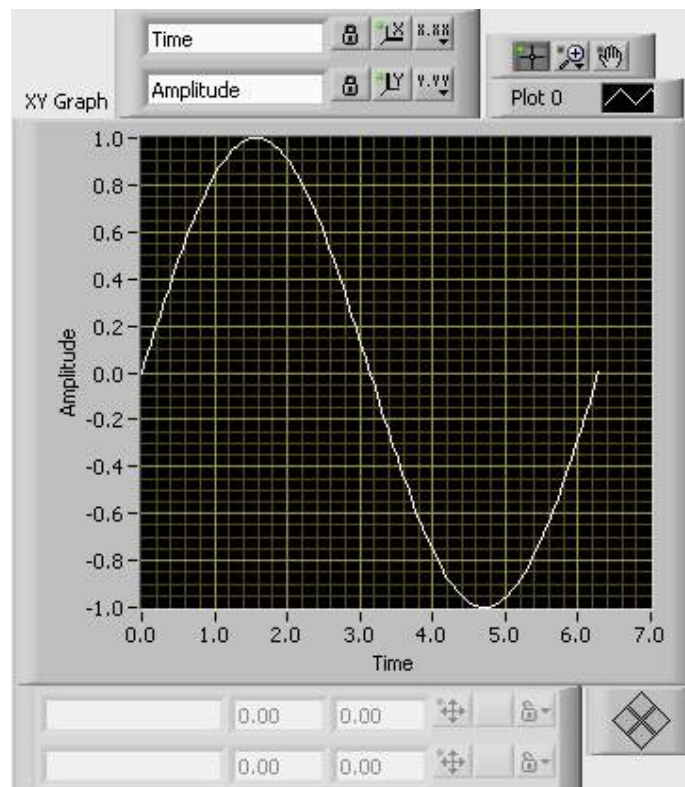
primului punct si a distantei **Dx** pe orizontala dintre doua puncte succesive, marimile respective vor fi aceleasi pentru toate graficele. In functia **Bundle**, la cea de a treia intrare se conecteaza matricea de valori.



EXERCITIU

Construiti diagramele din figurile de mai sus si verificati modul de functionare

11. Elementul XY Graph



Un element de tip **XY Graph** accepta la intrare un **Cluster** format din doua siruri (**Array**) de valori.

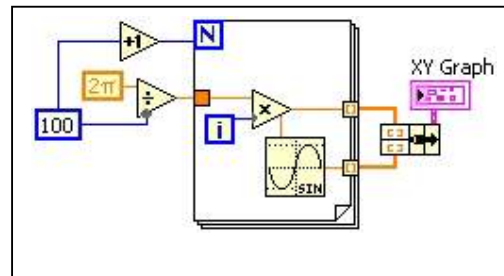
ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 11

Primul sir reprezinta abscisele iar cel de-al doilea ordonatele punctelor de pe grafic. Evident, cele doua siruri trebuie sa contina acelasi numar de valori. Atunci cand se primeste o astfel de structura de date, graficul anterior este sters.

In figura alaturata este prezentat un program pentru trasarea graficului functiei sinus intre 0 si 2π prin 101 puncte.

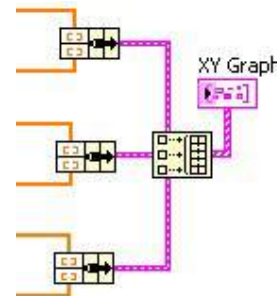
Stabilind ca punctele vor fi la distante egale pe orizontala, se determina intai distanta Δx pe abscisa dintre doua puncte succesive, impartind la 100 intervalul de trasare, de lungime 2π .



Intr-o bucla **For** cu numarul de iteratii egal cu numarul de puncte se determina la fiecare iteratie abscisa unui punct cu relatia $x_i = i \cdot \Delta x$ si se calculeaza ordonata $y_i = \sin(x_i)$. La iesirea din bucla **For**, sirurile de valori x_i si y_i sunt grupate intr-un cluster (cu functia **Bundle**), iar acesta este trimis la terminalul elementului **XY Graph**.

Daca se doreste trasarea simultana a mai multor grafice:

- se construiesc pentru fiecare grafic cate un **Cluster** format din doua **Array**-uri (unul pentru abscisele si altul pentru ordonatele punctelor);
- iesirile functiilor **Bundle** se conecteaza la o functie **Build Array**;
- iesirea functiei **Build Array** se conecteaza la terminalul elementului **XY Graph**.

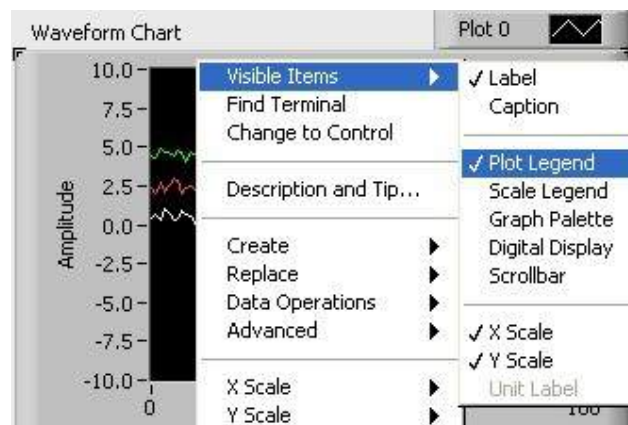


EXERCITIU

Construiti diagramele din figurile de mai sus si verificati modul de functionare

Optiuni specifice elementelor pentru reprezentari grafice

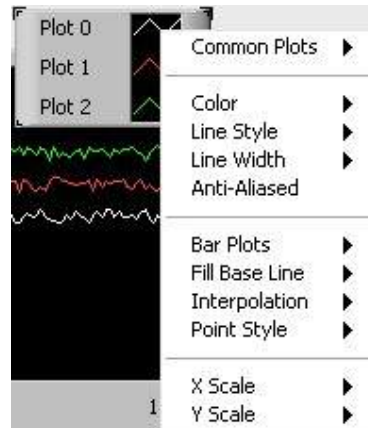
In meniul propriu al elementelor pentru reprezentari grafice, submeniul **Visible Items** contine o serie de optiuni pentru afisarea unor componente specifice.



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

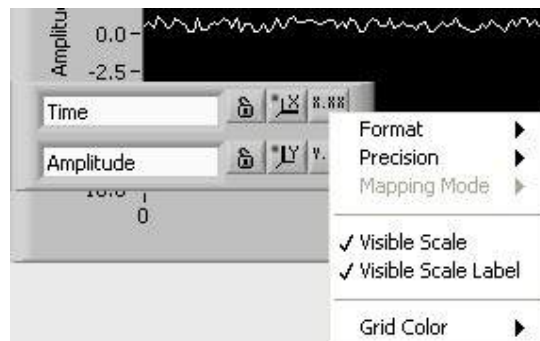
Laborator 11

Optiunea **Plot Legend** afiseaza o legenda ce poate fi pozitionata independent sau dimensionata pentru a avea un numar de pozitii egal cu numarul de grafice reprezentate. Numele graficelor (initial **Plot 0**, **Plot 1** s.a.m.d.) pot fi modificate cu ajutorul uneltei de editare a textelor.



Apasand butonul drept al mouse-ului atunci cand cursorul este pozitionat deasupra unui grafic din legenda, se deschide un meniu propriu din care pot fi configurate optiuni de afisare ale graficului respectiv: culoare, tip si grosime de linie etc.

Optiunea **Scale Legend** afiseaza o componenta ce permite configurarea modului in care sunt afisate scalele elementului de reprezentare grafica: eticheta scalei, format si precizie de reprezentare, vizibilitatea scalei sau a etichetei acesteia, culoarea caroiajului.



Pentru fiecare scala sunt disponibile:

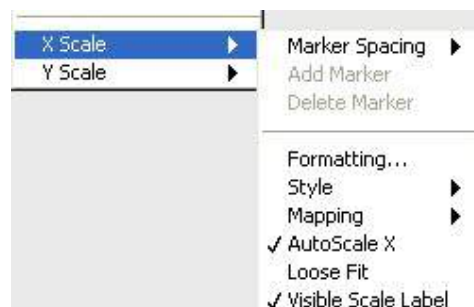
- un buton (notat cu **X** sau **Y**) la a carui apasare se realizeaza o autoscalare pe directia respective
- un buton (marcat cu un lacat) care, atunci cand este apasat, mentine continuu autoscalarea pe directia corespunzatoare.

Optiunea **Graph Palette** afiseaza o componenta cu unelte ce permit:

- deplasarea graficelor in interiorul elementului (butonul cu manuta)
- deschiderea unui submeniu cu unelte pentru marire sau micsoare statica pe diverse directii sau dinamica.



Optiunile **X Scale** si **Y Scale** din meniul propriu permit configurarea unor parametri ai modului de afisare a scalelor.



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 11

EXERCITII

1. Realizati un program in care, intr-un element de tip **Chart**, sa fie trasate doua grafice cu valori aleatoare (unul intre 0 si 1, celalalt intre 2 si 3).
2. Afisati componentele **Plot Legend**, **Scale Legend** si **Graph Palette** ale elementului de tip **Chart**.
3. Modificati modurile in care sunt afisate cele doua grafice (culoare, tip si grosime de linie, stilul punctelor).
4. Afisati graficele sub forma de histograme.
5. Modificati modul de interpolare al graficelor.
6. Umpleti zona de sub un grafic sau zona dintre cele doua grafice.
7. Modificati etichetele scalelor.
8. Autoscalati graficul pe axa **Y**.
9. Modificati formatul si precizia de reprezentare a valorilor de pe cele doua scale.
10. Afisati un caroiaj rosu.
11. Exersati lucrul cu unelte de marire sau micorare a imaginii din componenta **Graph Palette**.
12. Exersati configurarea unor parametri ai modului de afisare a scalelor (optiunile **X Scale** si **Y Scale** din meniul propriu al elementului de tip **Chart**).

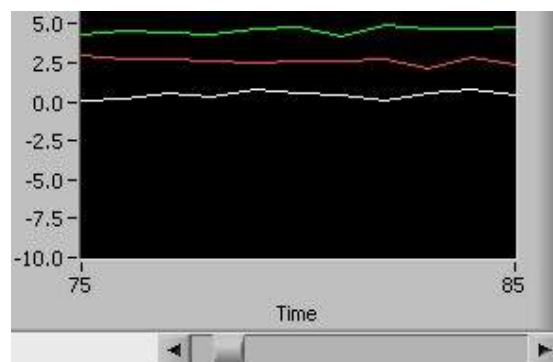
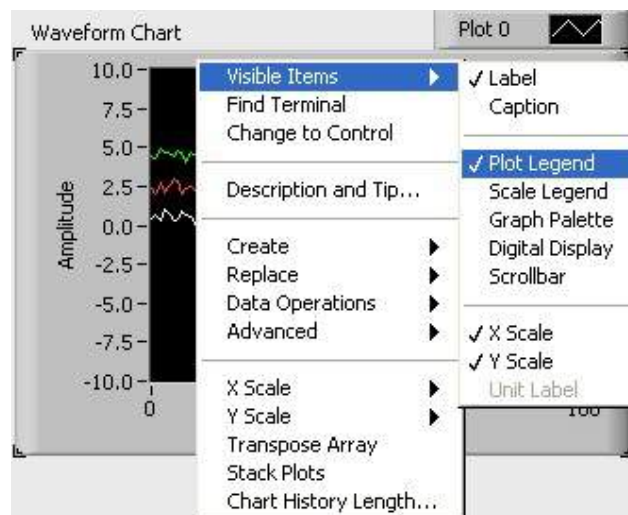
Optiuni specifice elementelor de tip Chart

In submeniul **Visible Items** al elementelor de tip **Chart** exista cateva optiuni specifice acestor elemente.

Optiunea **Digital Display** conduce la afisarea unui numar de elemente indicatoare pentru valori numerice egal cu numarul de pozitii ale legendei. Valoarea afisata de catre fiecare element indicator este cea a ultimului punct de pe graficul corespunzator.



Optiunea **Scrollbar** conduce la afisarea unei bare de defilare ce permite deplasarea pe directie orizontala a graficelor afisate.



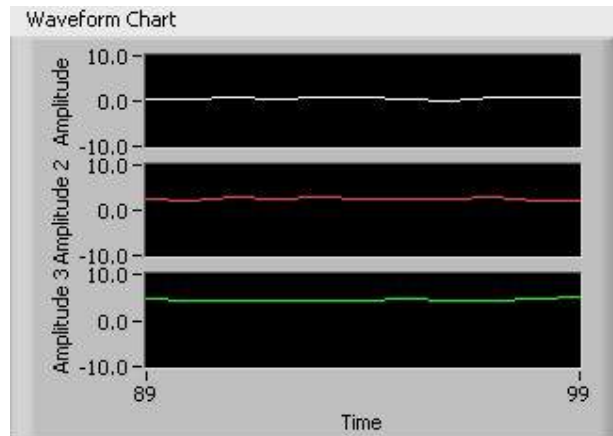
ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 11

Selectarea din meniul propriu al elementului **Chart** a optiunii **Transpose Array** realizeaza transpunerea matricei de valori, astfel incat nu se mai traseaza cate un grafic pentru fiecare linie ci cate unul pentru fiecare coloana.

In cazul afisarii mai multor grafice, optiunea **Stack Plots** imparte zona de afisare in mai multe portiuni distincte, cate una pentru fiecare grafic, cu scale separate pe axa Y.

Optiunea **Chart History Length...** permite sa se stabileasca numarul de valori ce pot fi stocate in "memoria interna" a elementului **Chart** (implicit 1024).



Daca unui element **Chart** i se trimit mai multe valori decat acest numar, valorile cele mai vechi se pierd si nu mai sunt vizibile la o eventuala "defilare" a graficului.

EXERCITII

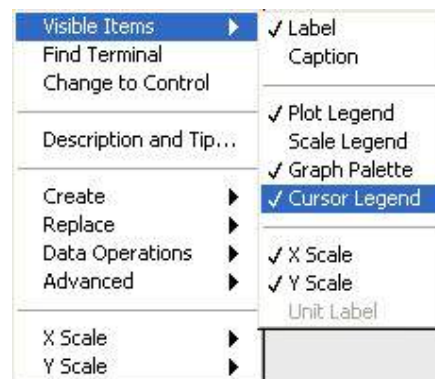
Pentru elementul de tip **Chart** din exercitiul anterior:

- afisati display-urile digitale si bara de defilare;
- defilati graficul cu ajutorul barei;
- impartiti zona de afisare a elementului **Chart** in doua portiuni distincte

Optiuni specifice elementelor de tip Graph si XY Graph

In submeniul **Visible Items** al elementelor de tip **Waveform Graph** sau **XY Graph**, optiunea **Cursor Legend** determina afisarea unei componente prin intermediul careia utilizatorul poate afisa si manipula unul sau mai multe cursoare.

Componenta **Cursor Legend** este formata dintr-o lista de cursoare si un element pentru comanda deplasarii cursorului.



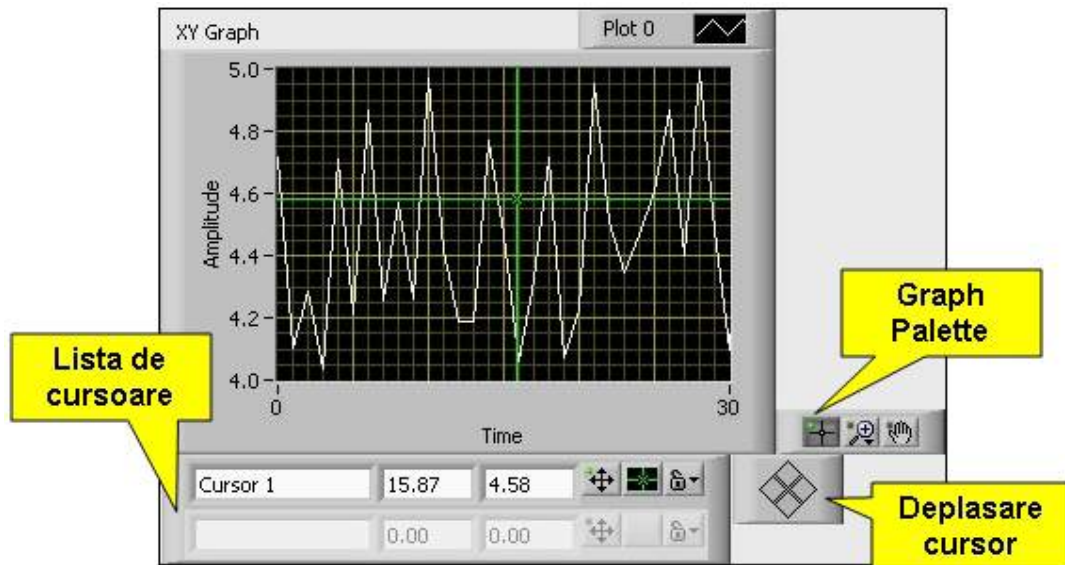
In lista de cursoare (initial cu doua linii vide), fiecarei linii ii corespunde un cursor. Lista poate fi dimensionata.

Pentru a adauga un nou cursor, este suficient sa se introduca un nume al acestuia sau sa se apese oricare din cele trei butoane de pe linia corespunzatoare. Pentru a sterge un cursor sau a insera un nou cursor intr-o anumita pozitie, din meniul

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 11

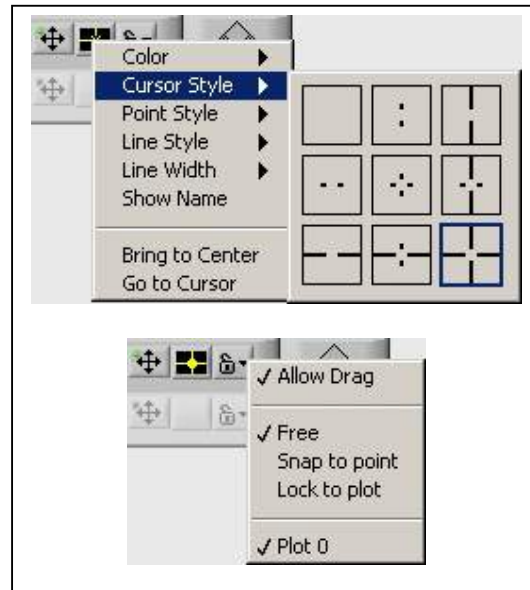
propriu al paletii de cursoare, submeniul **Data Operations**, se aleg optiunile **Delete Element** sau **Insert Element Before**.



O linie corespunzatoare unui cursor contine, in partea sa stanga, un element pentru numele cursorului si doua elemente pentru coordonatele X si Y ale acestuia.

In partea sa dreapta, fiecare linie corespunzatoare unui cursor contine trei butoane:

- un buton pentru activarea elementului de comanda a deplasarii: daca acest buton are LED-ul propriu aprins, atunci cursorul poate fi deplasat apasand pe unul din cele patru butoane romboidale ale elementului pentru comanda deplasarii cursorului. Daca LED-ul butonului este stins, elementul de comanda nu este activ.
- un buton pentru configurarea caracteristicilor cursorului (culoare, stil, forma punctului central, stilul si grosimea firelor reticulare s.a.m.d.)
- un buton prin care se activeaza sau nu posibilitatea de deplasare a cursorului cu unealta de operare (**Allow Drag**), se "leaga" cursorul de un anumit grafic (**Lock to plot**) sau de punctele acestuia (**Snap to point**) sau se lasa cursorul liber (**Free**).



Daca un cursor are optiunea **Allow Drag** activata, el va putea fi deplasat cu unealta de operare daca butonul din stanga al componentei **Graph Palette** este activat (are LED-ul aprins). Deplasarea se poate efectua fie "tragand" de punctul central al cursorului, fie de unul dintre firele reticulare.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 11

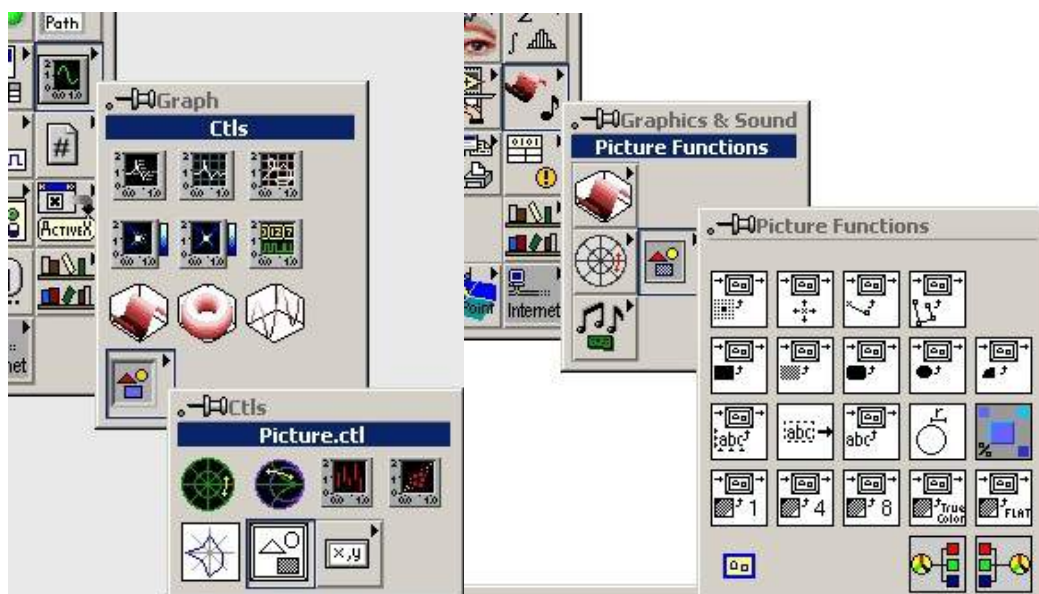
Elementele pentru nume si coordonate, precum si butoanele, pot fi dimensionate. Elementele pentru nume si coordonate poseda meniuri proprii individuale.

EXERCITII

1. Realizati un program in care, intr-un element de tip **Waveform Graph**, sa fie trasate doua grafice cu valori aleatoare (unul intre 0 si 1, celalalt intre 2 si 3).
2. Afisati lista de cursoare si definiti un cursor.
3. Modificati caracteristicile cursorului (culoare, stil, forma punctului central, stilul si grosimea firelor reticulare s.a.m.d.)
4. Deplasati cursorul cu ajutorul elementului de comanda a deplasarii, intai in starea liber, apoi "legat" de unul dintre grafice.
5. Deplasati cursorul cu unealta de operare, "tragand" de intersectia firelor reticulare, intai in starea liber, apoi "legat" de unul dintre grafice.
6. Afisati coordonatele X si Y ale cursorului cu o singura zecimala

12. Elemente si functii de tip Picture

Pentru a dispune pe panoul aplicatiei un element de tip **Picture.ctl**, acesta poate fi selectat din meniul de controale, submeniul **Graph**, submeniul **Ctls**.



Funcțiile pentru lucrul cu elemente de tip **Picture.ctl** sunt dispuse în meniul de funcții, submeniul Un element de tip **Picture** reprezintă o matrice de pixeli, fiecare pixel fiind definit prin coordonatele sale X și Y.

Originea sistemului de coordonate se afla în colțul stanga-sus al figurii. Axa X are sensul pozitiv spre dreapta, iar axa Y are sensul pozitiv în jos. **Graphics & Sound**, submeniul **Picture Functions**.

Utilizarea functiilor de tip Picture

Functiile pentru lucrul cu elemente de tip **Picture.ctl** efectueaza operatii asupra unei figuri in general in sensul de adaugare a unui element geometric in figura.

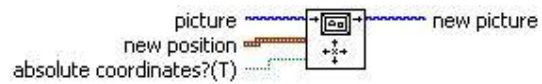
In imaginea de mai jos se pleaca de la o constanta **Empty Picture** (figura vida).

In prima etapa, utilizand functia **Move Pen.vi**, se deplaseaza un creion imaginar in punctul de coordonate (50, 50) al figurii.

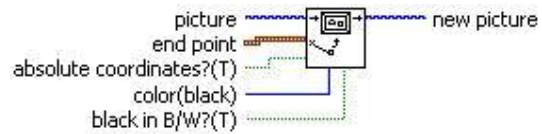
Functia **Move Pen.vi** nu adauga nimic la figura.

Cu ajutorul functiei **Draw Line.vi**, in figura preluata de la functia **Move Pen.vi** este trasat un segment de culoare rosie. Functie **Draw Line.vi** considera ca punctul de plecare al segmentului este punctul in care se afla creionul imaginar, fiind necesar ca functiei sa i se specifice doar punctul final al segmentului, in acest caz punctul de coordonate (120, 140).

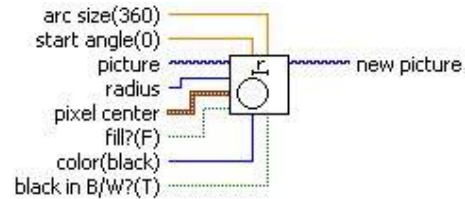
In figura preluata de la iesirea functiei **Draw Line.vi**, cu ajutorul functiei **Draw Circle by Radius.vi**, este trasat un cerc plin (vezi constanta **True**) de culoare albastra, cu centrul in punctul de coordonate (200, 200) si de raza 20. Figura obtinuta la iesirea functiei **Draw Circle by Radius.vi** este transferata elementului de tip **Picture.ctl**, rezultatul fiind afisarea figurii.



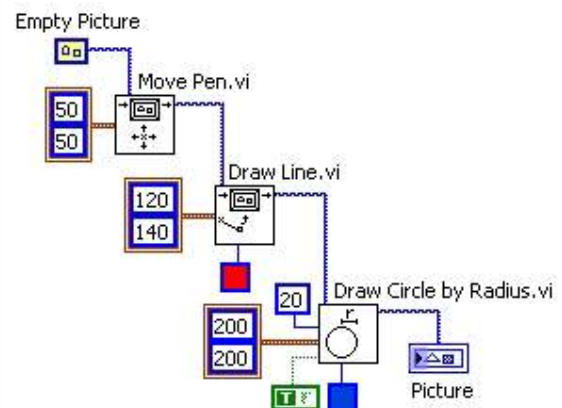
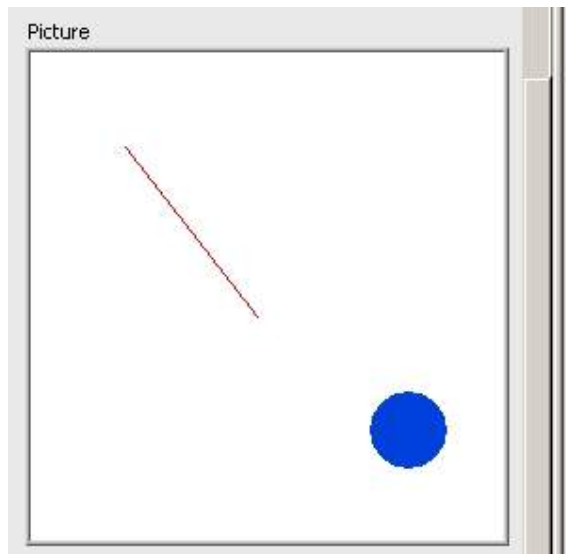
Move Pen.vi



Draw Line.vi



Draw Circle by Radius.vi



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE
Laborator 11

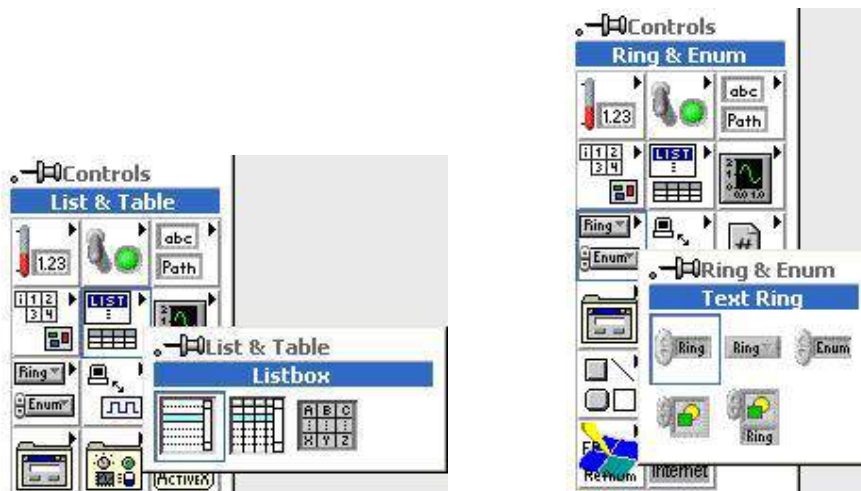
EXERCITIU

1. Construiti diagrama din figura de mai sus si verificati modul de functionare
2. Sa se realizeze un program pentru trasarea graficului unei parabole.
3. Sa se realizeze un program pentru trasarea graficului unui cerc
4. Sa se realizeze un program pentru trasarea graficului unei elipse
5. Sa se realizeze un program pentru simularea comportamentului unui resort elastic
6. Sa se realizeze un program pentru simularea comportamentului unui resort elastic, utilizand elemente **Picture**

12. Meniuri pentru elemente ListBox, Table si Ring

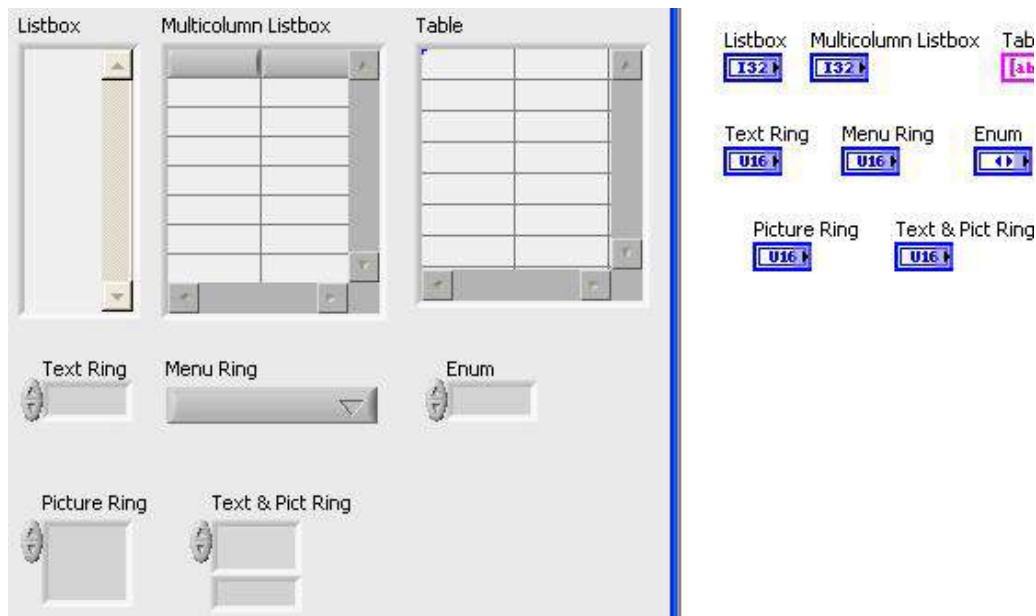
Meniul **List & Table** al paletei de controale contine elemente de tipurile **ListBox**, **Multicolumn ListBox** si **Table**.

Meniul **Ring & Enum** contine elemente de tipurile **Text Ring**, **Menu Ring**, **Enum**, **Picture Ring** si **Text & Pict Ring**.



Tipul de date

Caracteristica principala a elementelor de tip **List** sau **Ring** este aceea ca, desi in panou utilizatorul are la dispozitie o lista de elemente text, valorile corespunzatoare din diagrama sunt numerice.

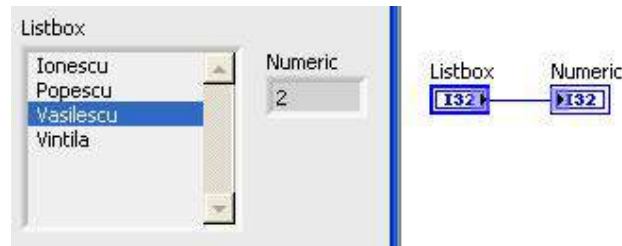


ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 12

Aceasta se observa in primul rand din faptul ca toate elementele (cu exceptia celor de tipurile **Table** sau **Enum**) au terminale ce semnifica numere intregi (**I32**) sau naturale (**U16**).

De exemplu, **valoarea numerica** a unui element de tipul **Listbox** este reprezentata de **indicele (numarul de ordine)** textului selectat de catre utilizator (prima linie de text are indicele 0).



Definirea itemilor unui element ListBox

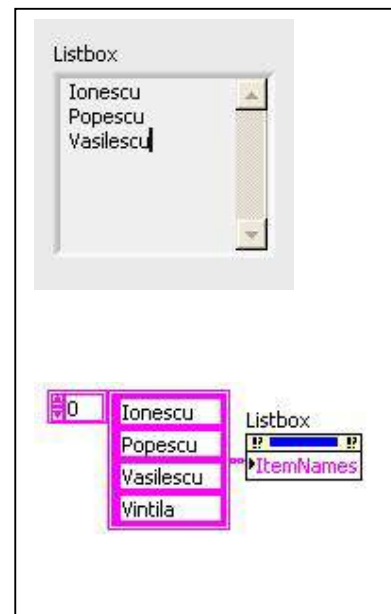
Itemii unui element **Listbox** reprezinta liniile de text din interiorul acestuia.

Modalitatea cea mai simpla de a defini acesti itemi, **valabila insa doar atunci cand programul nu ruleaza**, este de a-i introduce cu ajutorul unelei de editare a textelor.

Fiecare item trebuie introdus pe o linie separata, modalitatea de introducere fiind similara celei de la elementele alfanumerice:

- daca se apasa tasta **Enter** "mare" (din apropierea tastelor pentru litere), se trece la linia urmatoare
- daca se apasa tasta **Enter** "mica" (din zona cu taste numerice aflata in partea dreapta a tastaturii) sau daca se face click cu mouse-ul in afara elementului, se considera ca s-a incheiat introducerea itemilor.

Textul format de itemii unui element **Listbox** poate fi editat ulterior pentru modificare, stergere sau inserare.



Daca se doreste editarea itemilor unui **Listbox in timpul rularii programului**, se procedeaza astfel:

- se creeaza un **nod de proprietati** al elementului **Listbox**
- se selecteaza pentru nodul respectiv proprietatea **ItemNames**
- din meniul propriu al nodului se selecteaza optiunea **Change To Write**
- se conecteaza la nod un **Array** de valori **String** ce vor reprezenta itemii elementului **Listbox**

EXERCITII

1. Dispuneti in panou un element de tipul **Listbox** si introduceti, cu unealta de editare texte, cativa itemi ai acestuia.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 12

2. Realizati un program in care itemii unui element **Listbox** sa fie generati in timpul rularii.

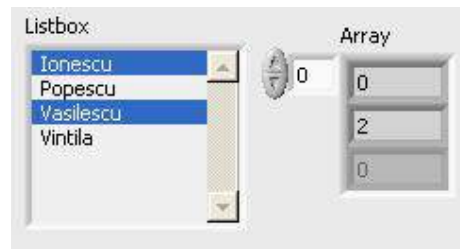
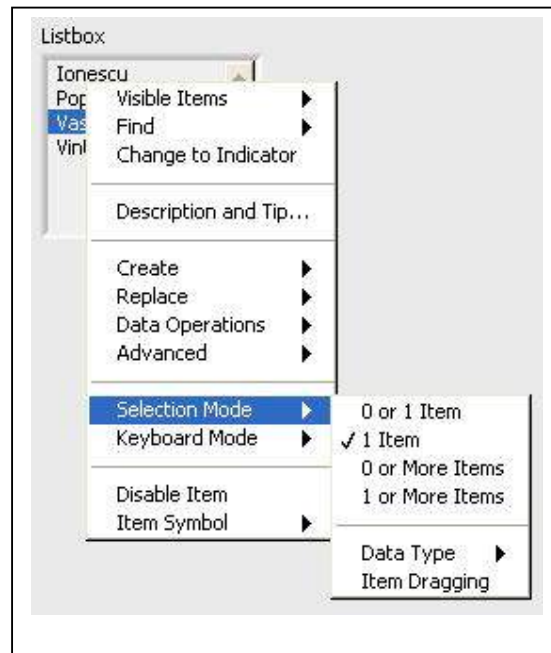
Selectii multiple

Din meniul propriu al unui element de tipul **Listbox**, optiunea **Selection Mode**, poate fi specificat numarul de itemi ce pot fi selectati simultan.

Atunci cand exista mai multe variante de selectie (nici unul, unul sau mai multi itemi), **pentru a selecta sau deselecta un item se face click pe acesta tinand apasata tasta Shift**.

Daca un element **Listbox** este in modul de selectie **0 or 1 Item** si nu are nici un item selectat, valoarea corespunzatoare a elementului este **-1**.

Daca sunt selectati mai multi itemi ai unui **Listbox**, atunci acesta nu mai este reprezentat de o valoare scalara ci de un **Array** ce contine indicii itemilor selectati. Daca un element **Listbox** este in modul de selectie **0 or More Items** si nu are nici un item selectat, atunci el este reprezentat de un **Array vid**.



EXERCITIU

1. Dispuneti un element **Listbox** pe panou si stabiliti modul de selectare **0 or More Items**.

2. Dispuneti de asemenea pe panou un element indicator **Array** in care sa fie afisati indicii itemilor selectati.

3. Exersati diverse variante de selectare: nici un item, un item, mai multi itemi.

SIMULARI

1. Simularea unui subsistem de control al unui manipulator industrial

Manipulatorul real este echipat cu un traductor pentru masurarea în coordonate absolute a pozitiei sale pe directie orizontala.

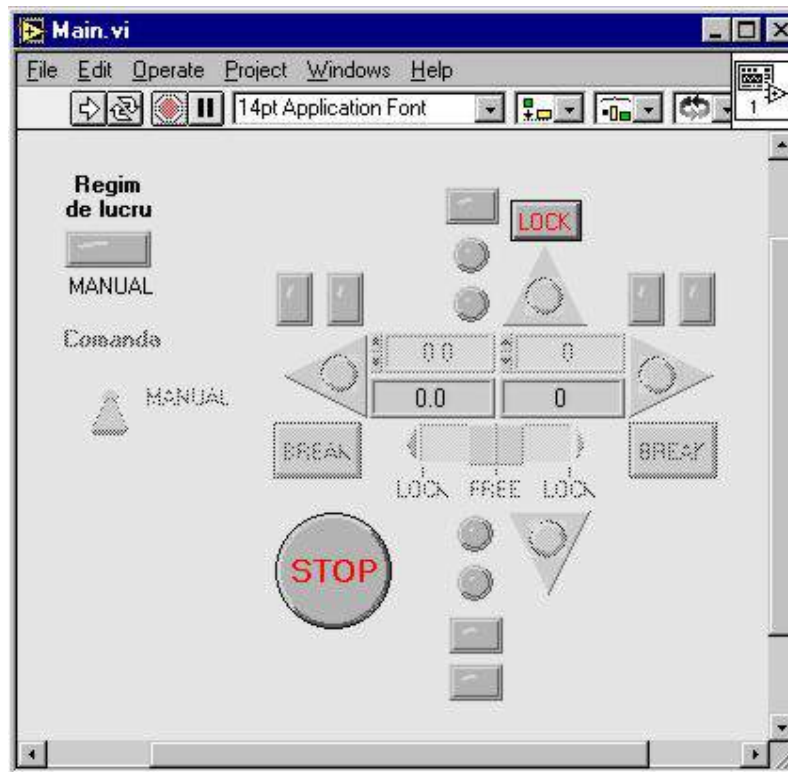
În deplasarea pe directie verticala, un manipulator trece pe lângă patru senzori de pozitie, dintre care doi sunt dispusi la capetele cursei respective.

Fiecare manipulator dispune de asemenea de doi senzori de coliziune montati pe extremitatile sale si de un senzor pentru sesizarea faptului ca în baia de tratare deasupra careia se afla manipulatorul se gaseste deja o sarja.

Aplicatia consta din **doua instrumente virtuale**, ruland pe computere diferite, fiecare echipat cu cate o placa multifunctionala pentru achizitie de date.

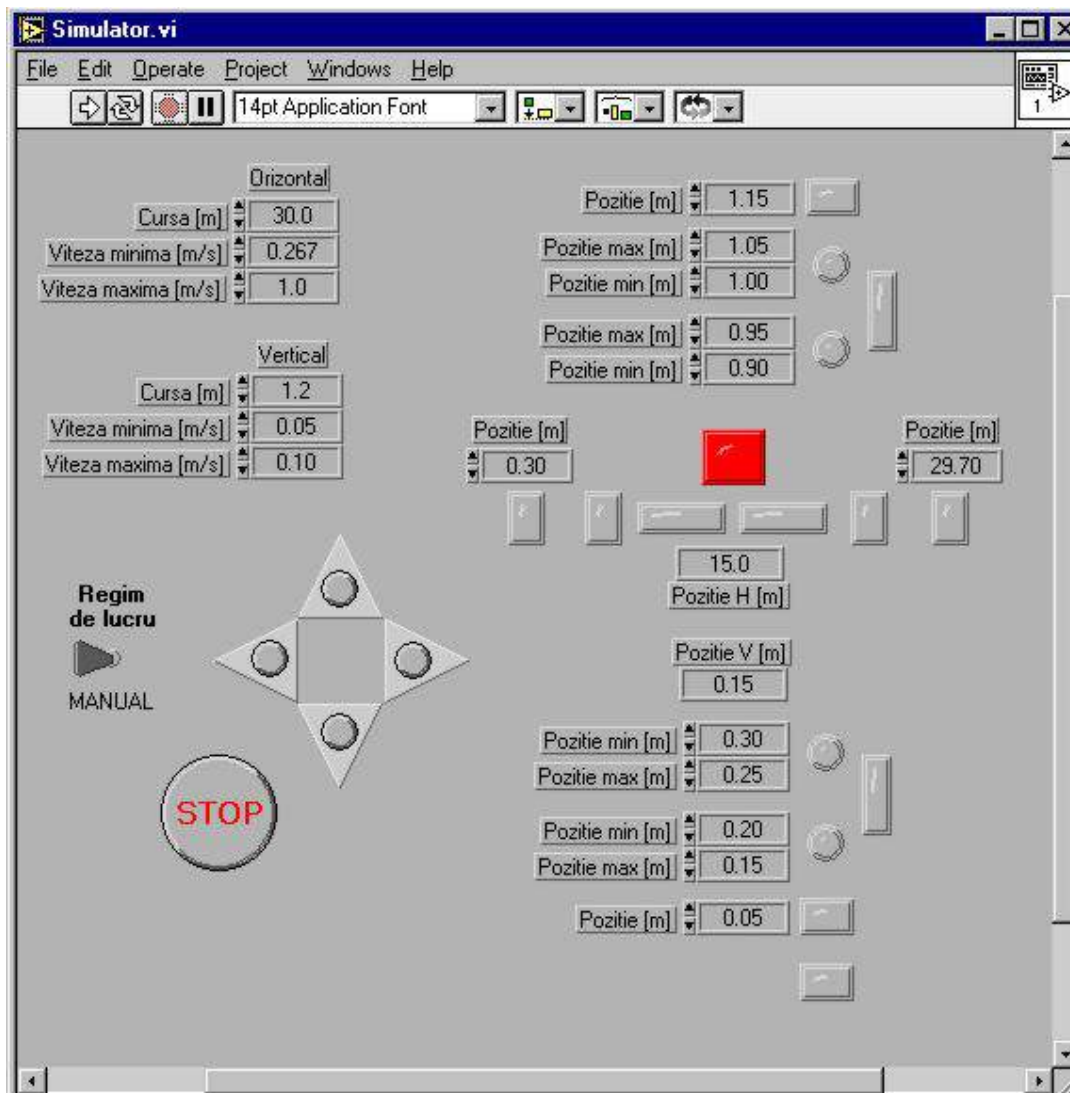
Main.vi contine panoul de comanda prin software a manipulatorului (accesibil utilizatorului).

Simulator.vi reprezinta panoul de comanda hardware si afiseaza parametrii de stare si de pozitie ai manipulatorului simulat.



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 12



2. Simularea sistemului de automatizare al unui cuptor de tratament termic

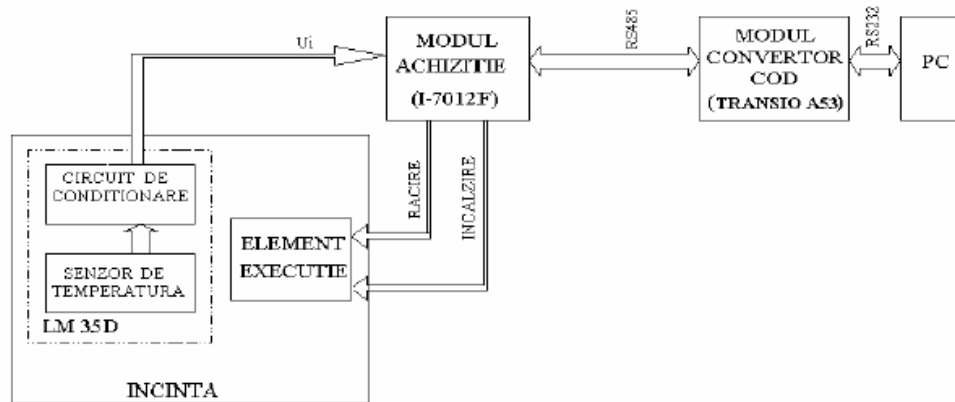
Sistemul prezentat este un sistem de monitorizare si control automat al temperaturii realizat cu PC (CNIV 2004- Ioan Lita, Ion Bogdan Cioc, Rodica Mihaela Teodorescu, Universitatea din Pitesti, Catedra de Electronica si Calculatoare, Email: lita@upit.ro)

Reglarea automata a temperaturii este realizata de calculator prin intermediul aplicatiei software, care îndeplineste si functia de regulator automat tripozitional.

Schema bloc a sistemului este prezentata în figura de mai jos.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 12



Pentru masurarea temperaturii în puncte situate la distanta fata de calculatorul de proces, se foloseste un modul de achizitie de date independent I-7012F, situat aproape de locul în care se face masurarea.

Modulul I-7012F contine un convertor analog-numeric pe 16 biti, are 1 intrare analogical diferentiala, o intrare digitala si doua iesiri digitale, iar comunicatia se realizeaza pe o interfata RS-485 ce asigura o viteza de comunicare de pâna la 115,2 Kbps.

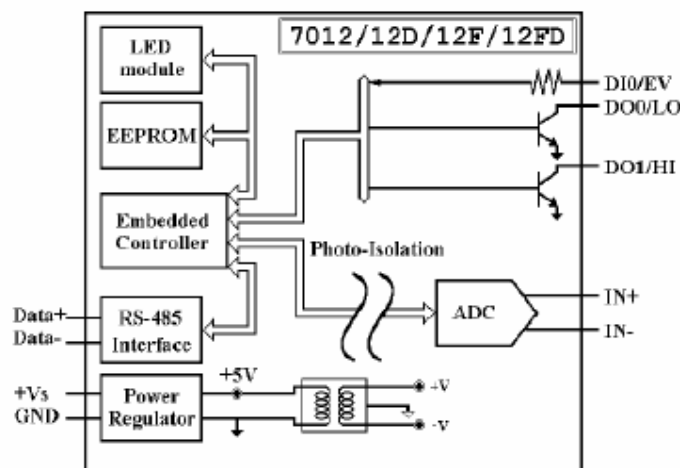


Fig. 2 – Schema bloc a modului I-7012F

Partea soft a aplicatiei este realizata în LabVIEW 6.1 si permite monitorizarea temperaturii într-o incinta prin masurarea temperaturii curente, reprezentarea grafica a evolutiei temperaturii din incinta respective si salvarea datelor într-un fisier pe harddisk .

Valoarea temperaturii masurate este comparata continuu cu doua limite prestabilite, iar în cazul depasirii acestor limite se comanda pornirea unor instalatii de racire sau de încalzire.

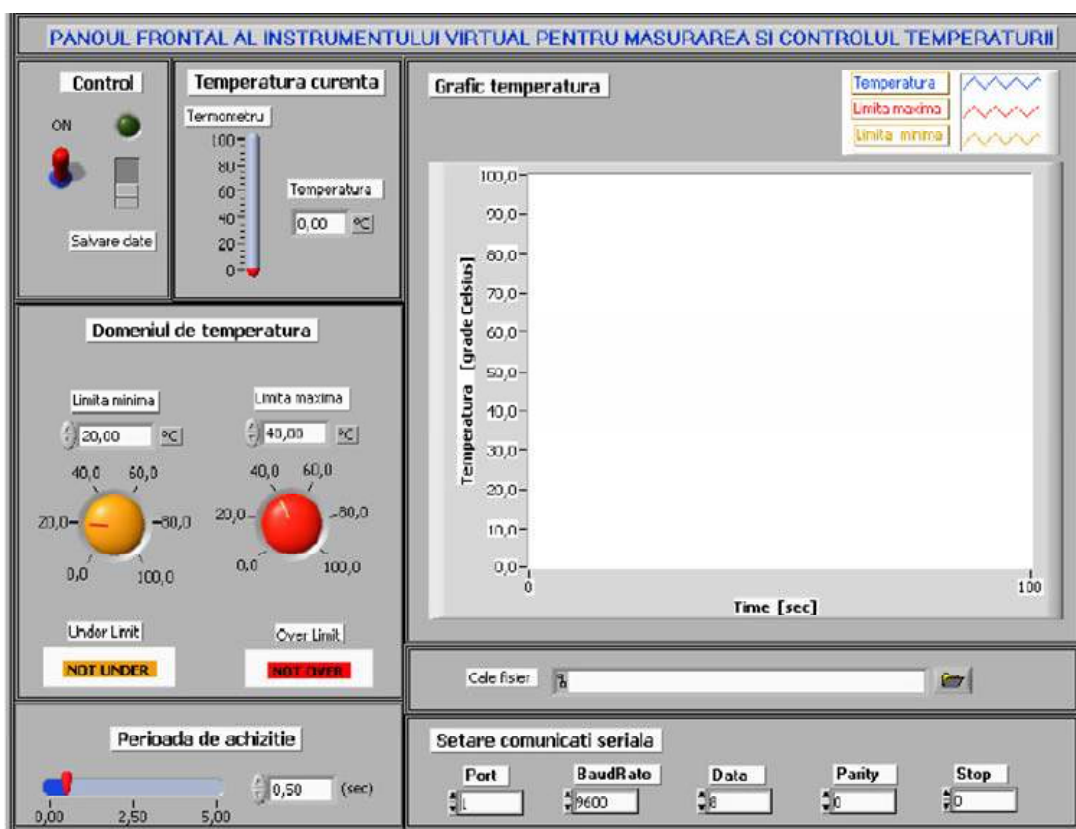
Limitele respective sunt stabilite de catre operator prin intermediul interfetei aplicatiei si se pot modifica oricând în timpul executiei programului. Daca temperatura se încadreaza între limitele stabilite, cele doua semnale de comanda sunt inactice, ceea ce înseamna ca nici una din instalatii nu este pornita .

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

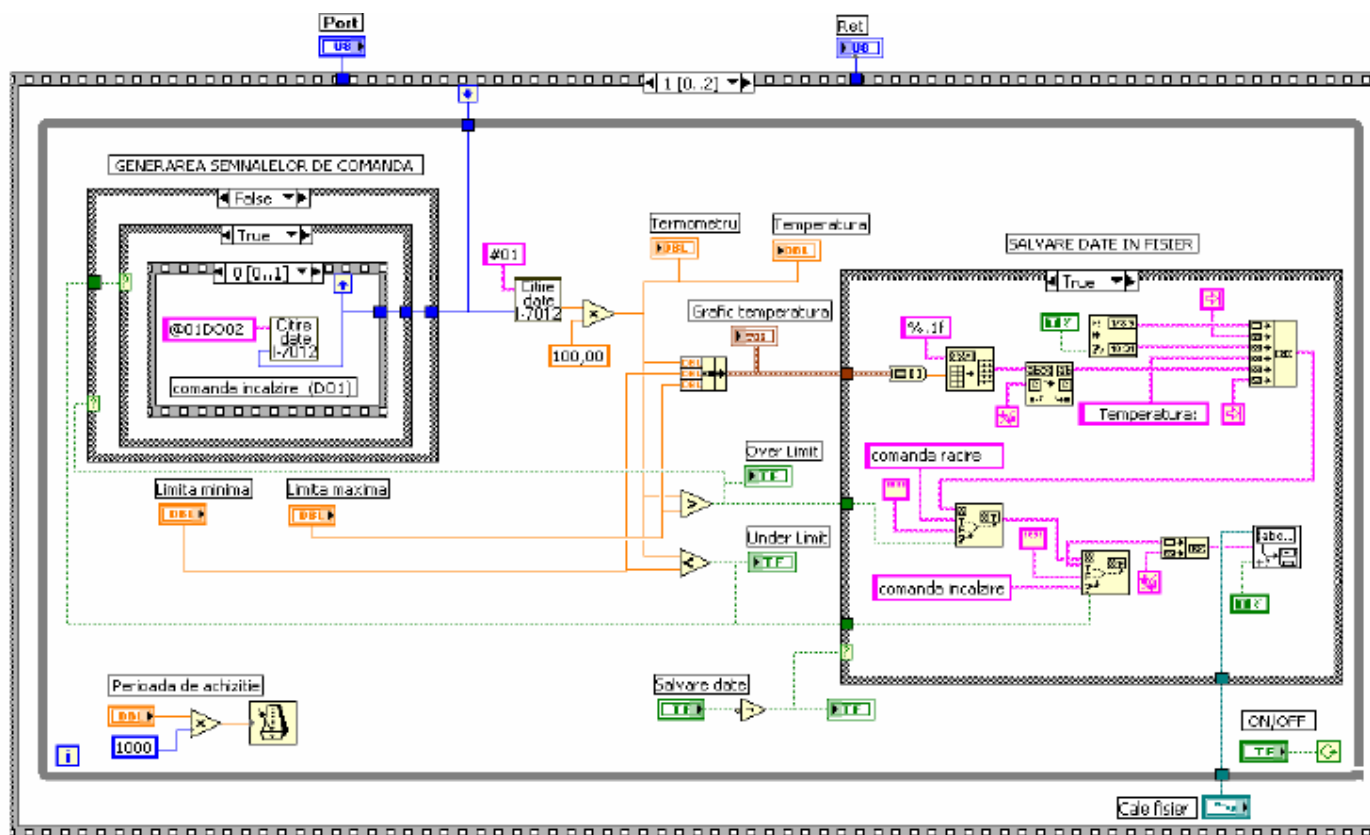
Laborator 12

În momentul în care nivelul temperaturii nu se mai încadrează în limitele impuse, se activează una din cele două iesiri digitale ale plăcii.

Controlul iesirilor digitale precum și a citirii valorii aplicate la intrarea analogică a plăcii se face prin trimiterea unor cuvinte de comandă în format sir de caractere. Comunicarea și controlul modului I-7012F se face cu ajutorul subinstrumentului virtual I-7012, care este prezentat mai jos. Aceste cuvinte de comandă (sir de caractere) încep cu un caracter special ce identifică tipul de comandă, care este urmat de adresa modului și de comandă propriu-zisă.



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE Laborator 12



Măsurarea semnalelor în LabVIEW

5.1. Scopul lucrării

- Deprinderea noțiunilor teoretice fundamentale despre măsurarea semnalelor în LabVIEW;
- Crearea unor instrumente virtuale foarte simple cu ajutorul cărora vom măsura diferiți parametri ai semnalelor;

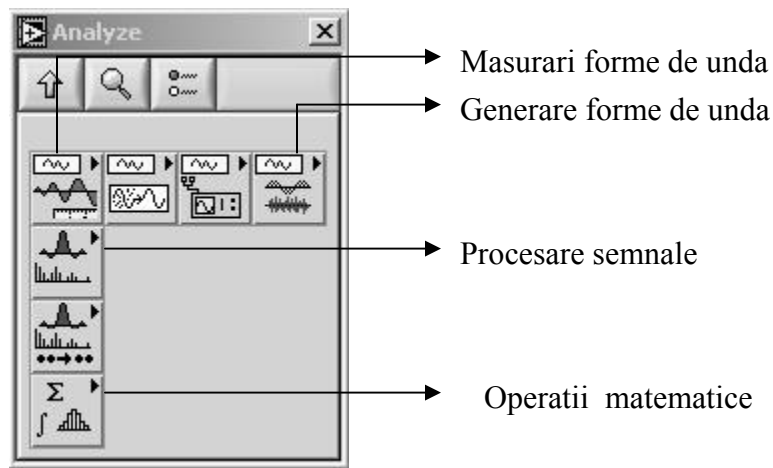
5.2. Aparatura necesară

- stații de lucru care au instalat LabVIEW

5.3.Considerații teoretice

5.3.1 Noțiuni generale

Vom analiza și vom lucra cu funcțiile din paleta de funcții *Analyze*. Pentru exemplificare a se vedea figura 1.



În prima parte a laboratorului vom lucra în principal cu funcțiile din paleta de funcții *Waveforms Measurements*. Această paletă de funcții este prezentată în figura 2. Această paletă de funcții ne oferă o serie de funcții pentru a efectua măsurători ale semnalelor atât în domeniul timp cât și în domeniul frecvență. Ca de exemplu se pot efectua măsurătorile:

- măsurarea valorii medii ale semnalelor (DC-Direct Current);
- măsurări ale valorii efective (RMS-Root Mean Square);
- măsurarea nivelelor semnalelor și a amplitudinii acestora;
- spectru FFT (ne returnează faza și amplitudinea);
- spectru de putere FFT, timpii de creștere și cădere, rata de creștere;

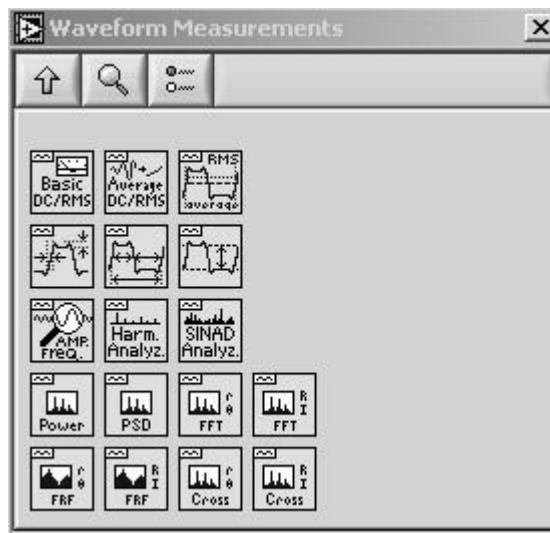


Figura 2 – Paleta de funcții Waveform Measurements

În cele ce urmează vom parcurge exemple pentru exemplificarea funcțiilor din paleta prezentată mai sus.

Exemplul 1

Vom măsura în acest prim exemplu amplitudinea și nivelele de high și low pentru două semnale obținute prin sumarea, respective înmulțirea a două forme de undă. Se vor parcurge pașii:

1. În cadrul diagramei bloc vom plasa din cadrul paletii de funcții Waveform>>Waveform Generation, funcția Sine Waveform, cu ajutorul căreia vom genera o formă de undă sinusoidală (se vor folosi două funcții Sine Waveform). A se vedea figura 3.
2. În cadrul panoului frontal vom plasa din cadrul paletii cu funcții numerice butoane rotative, iar din cadrul paletii Graph vom plasa funcția Waveform Graph. (se vor folosi patru funcții Waveform Graph și 4 butoane rotative).
3. Se realizează rutarea așa cum se vede și în figura 3, după cum urmează: la terminalele frequency și amplitude vom conecta cele două terminale de ieșire ale celor două butoane rotative pe care le-am plasat în cadrul diagramei bloc, iar la terminalul signal out al funcției Sine Waveform vom conecta intrarea graficului unde vom reprezenta forma de undă.
4. Din cadrul meniului Tools vom alege Edit Text și vom schimba intervalele pentru frecvență și amplitudine după cum urmează: pentru frecvență până la 50, iar la amplitudine până la 10.
5. Tot cu ajutorul la Edit Text vom schimba denumirile butoanelor rotative, respectiv la grafice.
6. Din cadrul paletii de funcții Waveforms Operations vom plasa funcțiile Add Waveforms, respectiv Multiply Waveforms. Se va realiza rutarea așa cum se vede în figura 3 de mai jos.
7. Din cadrul paletii Waveforms Measurements vom plasa funcția Amplitude and Levels (vom avea nevoie de două astfel de funcții). La terminalul signal in vom aduce ieșirea din sumatorul de semnale, respectiv din multiplicator. Vom crea indicatoare pentru terminalele high state level, low state level și amplitudine.

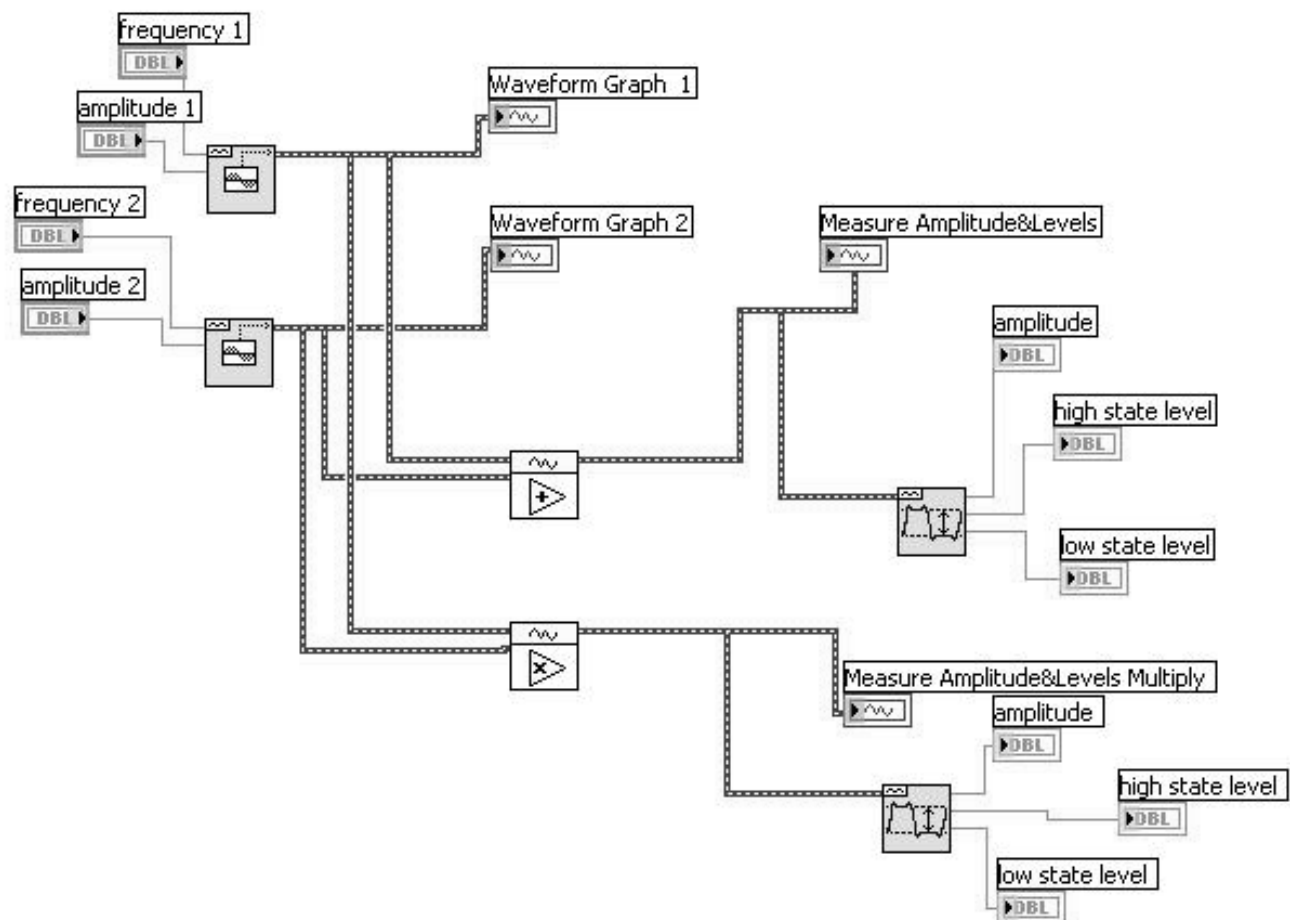
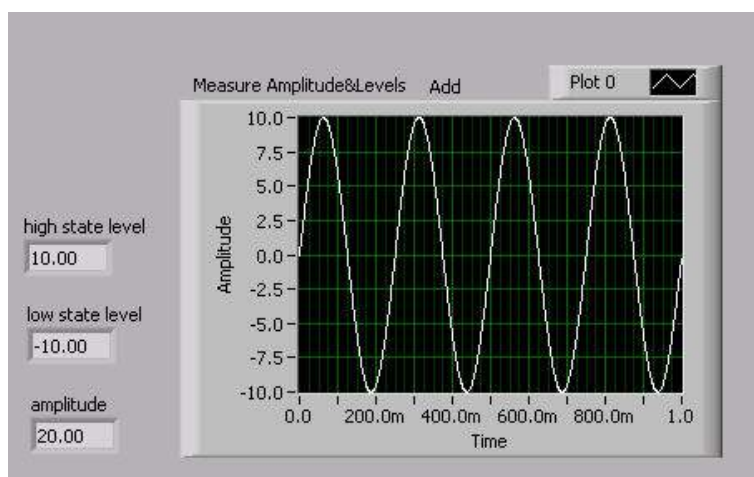


Figura 3-VI pentru măsurarea amplitudinii și a nivelelor unui semnal



Panou frontal masurare nivele si amplitudine

Exemplul 2

În aceste exemple vom prezenta două dintre cele mai comune măsurători care le efectuăm asupra semnalelor și anume nivelele DC și RMS despre care am amintit și la începutul laboratorului.

Ce este nivelul DC a unui semnal?

Putem folosi nivelul DC pentru a defini valoarea unui semnal static sau care variază lent. Măsurătorile pentru DC pot avea și valori pozitive și negative. Nivelul DC a unui semnal continuu $v(t)$ pe intervalul de timp de la t_1 la t_2 este dat de ecuația:

$$V_{DC} = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} V(t) dt$$

, unde $t_2 - t_1$ reprezintă timpul de integrare sau timpul de măsurare. Deci nivelul DC este de fapt valoarea medie a unui semnal calculată pe un interval de timp.

A. Vom parcurge următorii pași pentru a realiza acest instrument virtual:

1. În cadrul diagramei bloc vom plasa din cadrul paletii de funcții Waveform>>Waveform Generation, funcția Square Waveform, cu ajutorul căreia vom genera o formă de undă dreptunghiulară. A se vedea figura 4. Vom crea elemente de control pentru terminalele amplitude, frequency, offset și duty cycle.
2. Din cadrul paletii de funcții Waveforms Measurements vom plasa în cadrul diagramei bloc funcția Basic Averaged DC-RMS. Semnalul de la terminalul de ieșire a funcției Square Waveform îl vom aduce la terminalul de intrare a funcției pentru calculul nivelelor DC-RMS.
3. Se vor crea elemente de indicare pentru terminalele DC value, RMS value și un element de control pentru tipul ferestrei folosite la integrare.
4. Se va realiza rutarea așa cum se vede în figura 4. Se va lucra cu un offset de 2V. Ce observați dacă offsetul e 0? Ce se întâmplă cu nivelul DC modificând factorul de umplere a semnalului?

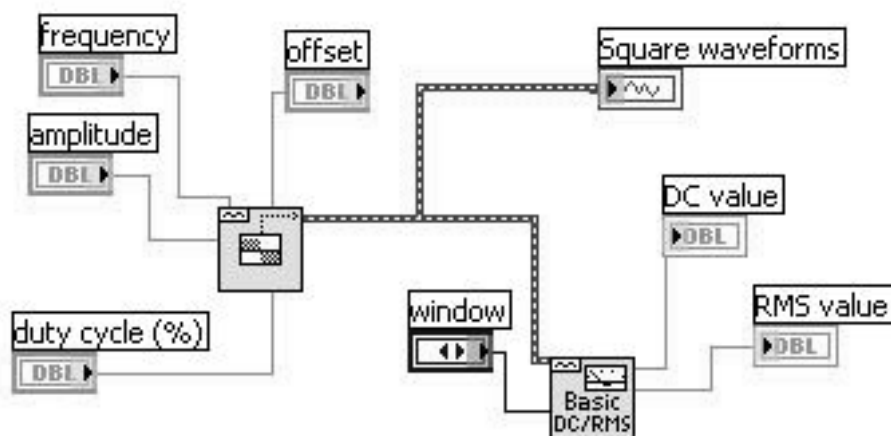
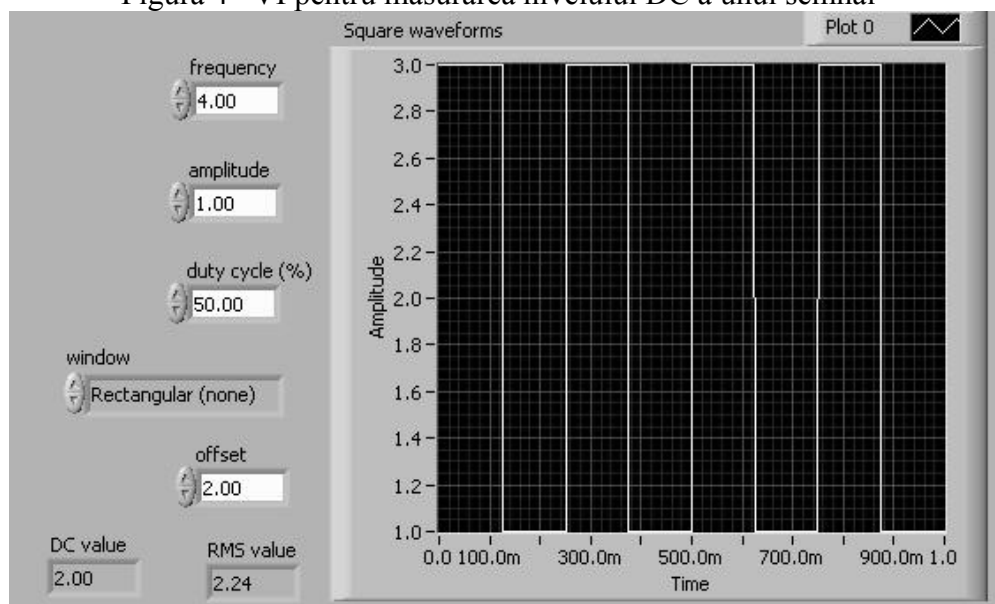


Figura 4 –VI pentru măsurarea nivelului DC a unui semnal



Panou frontal pentru masurarea nivelului DC a unui semnal

Ce este nivelul RMS a unui semnal?

Măsurarea nivelului RMS este folosită atunci când este nevoie de o reprezentare a energiei. Valoarea sa este întotdeauna pozitivă. Nivelul RMS a unui semnal continuu pe un interval de timp (t_1, t_2) este dat de ecuația:

$$V_{RMS} = \sqrt{\frac{1}{t_2 - t_1} \int_{t_1}^{t_2} V^2(t) dt}$$

,unde t_2-t_1 reprezintă nivelul de integrare sau timpul de măsurare. Deci nivelul RMS este valoarea efectivă a unui semnal măsurată pe un interval de timp.

B. Vom parcurge următorii pași pentru a realiza acest instrument virtual:

1. În cadrul diagramei bloc vom plasa din cadrul paletelor de funcții Waveform>>Waveform Generation, funcția Sine Waveform, cu ajutorul căreia vom genera o formă de undă sinusoidală .A se vedea figura 5.Vom crea elemente de control pentru terminalele amplitude, frequency, offset.
2. Din cadrul paletelor de funcții Waveforms Measurements vom plasa în cadrul diagramei bloc funcția Basic Averaged DC-RMS .Semnalul de la terminalul de ieșire a funcției Sine Waveform îl vom aduce la terminalul de intrare a funcției pentru calculul nivelelor DC-RMS.
3. Se vor crea elemente de indicare pentru terminalul RMS value și un element de control pentru tipul ferestrei folosite la integrare.
4. Se va realiza rutarea așa cum se vede în figura 5.Se vor da valorile 50Hz și 310V la frecvența semnalului sinusoidal și respectiv amplitudinea acestuia.
5. În cadrul panoului frontal vom da click dreapta pe indicatorul Waveform Graph și din meniul derulant vom alege Visible Items-Graph Palette.Va apare în cadrul panoului frontal un set de unelte cu care putem mări și micșora o porțiune din grafic dorită.

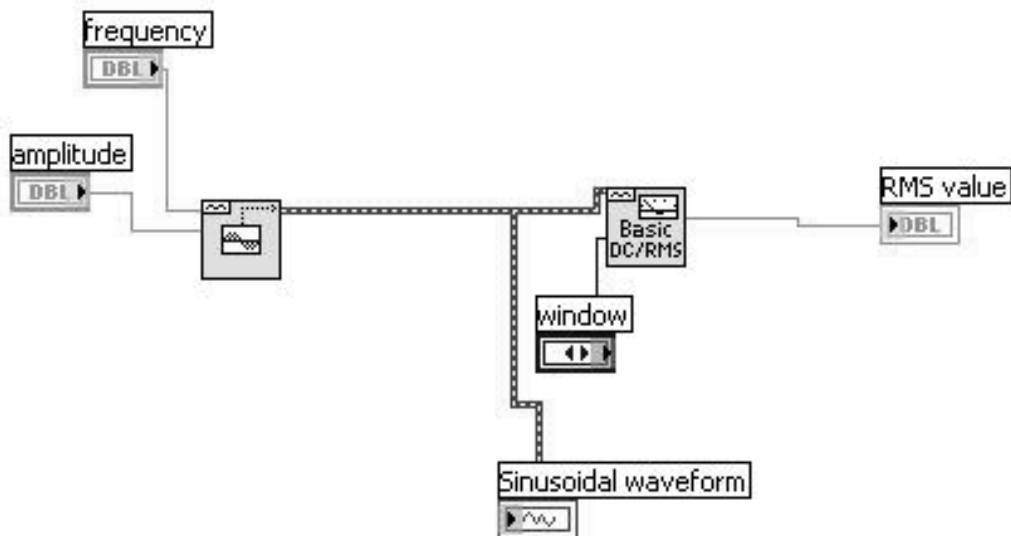


Figura 5-VI pentru măsurarea nivelului RMS a unui semnal

În continuare vom efectua măsurări în domeniul frecvenței.Vom măsura amplitudinea și faza unui spectru Fourier.

Noțiuni teoretice despre Transformata Fourier

Semnalul periodic $x(t)$ satisface relația:

$$x(t) = x(t + T) \quad (1)$$

În care T reprezintă perioada. Din punct de vedere matematic relația trebuie satisfăcută pentru orice valoare a timpului $t \in (-\infty, +\infty)$. Evident, semnalele reale au durate finite și, de aceea, relația (1) este satisfăcută într-un interval de timp finit. Se va spune că semnalul este periodic, dacă durata semnalului este mult mai mare decât perioada T și dacă nu interesează regimurile tranzitorii din circuite, determinate de apariția sau dispariția semnalului. În aceste condiții se poate accepta că durata semnalului este practic infinită, că semnalul este aplicat la intrarea circuitului începând de la un moment și sfârșind la un alt moment, indefinit îndepărtate față de intervalul de timp în care se studiază semnalul sau răspunsul circuitului la semnalul considerat.

Perioadei T îi corespunde frecvența $F = \frac{1}{T}$ și frecvența unghiulară sau pulsația

$$\Omega = 2\pi \cdot F = \frac{2\pi}{T}.$$

S-a văzut că analiza Fourier a semnalului periodic constă în scrierea

$$x(t) = C_0 + \sum_{n=1}^{\infty} C_n \cos n \cdot \Omega \cdot t + \sum_{n=1}^{\infty} S_n \sin n \cdot \Omega \cdot t \quad (2)$$

care reprezintă *forma trigonometrică* a seriei Fourier.

Observând că

$$C_n \cos n \cdot \Omega \cdot t + S_n \sin n \cdot \Omega \cdot t = A_n \cos(n \cdot \Omega \cdot t + \varphi_n)$$

Cu notațiile

$$A_n^2 = C_n^2 + S_n^2, \quad \varphi_n = -\arctg \frac{S_n}{C_n} \quad (3)$$

se găsește *forma armonică* a seriei Fourier

$$x(t) = A_0 + \sum_{n=1}^{\infty} A_n \cos(n \cdot \Omega \cdot t + \varphi_n) \quad (4)$$

Exemplul 4

Vom crea un instrument virtual pentru măsurarea fazei și amplitudinii componentelor spectrale a unui semnal căruia i-am aplicat transformata Fourier. În acest scop vom parcurge următorii pași:

1. În cadrul diagramei bloc vom plasa din cadrul paletii de funcții Waveform>>Waveform Generation, funcția Sine Waveform, cu ajutorul căreia vom genera o formă de undă sinusoidală (se vor folosi două funcții Sine Waveform). A se vedea figura 6.
2. În cadrul panoului frontal vom plasa din cadrul paletii cu funcții numerice butoane rotative, iar din cadrul paletii Graph vom plasa funcția Waveform Graph. (se vor folosi patru funcții Waveform Graph și 4 butoane rotative).
3. Se realizează rutarea așa cum se vede și în figura 6, după cum urmează: la terminalele frequency și amplitude vom conecta cele două terminale de ieșire ale celor două

- butoane rotative pe care le-am plasat în cadrul diagramei bloc, iar la terminalul signal out al funcției Sine Waveform vom conecta intrarea graficului unde vom reprezenta forma de undă.
4. Din cadrul meniului Tools vom alege Edit Text și vom schimba intervalele pentru frecvență și amplitudine. Tot cu ajutorul la Edit Text vom schimba denumirile butoanelor rotative, respectiv la grafice.
 5. Din cadrul paletelor de funcții Waveforms Operations vom plasa funcția Multiply Waveforms. Se va realiza rutarea așa cum se vede în figura 6.
 6. Din cadrul paletelor Waveforms Measurements vom plasa funcția FFT Spectrum (Mag-Phase). La terminalul view vom crea un element de control; la fel vom proceda și la terminalul window. La terminalele magnitude și phase vom crea elemente indicatoare.
 7. Din cadrul paletelor Signal Processing, vom alege paleta de funcții Frequency Domain din care vom plasa funcția Real FFT. La unul din cele 4 grafice vom lega ieșirea FFT, iar la altul rezultatul după multiplicarea celor două forme de undă.

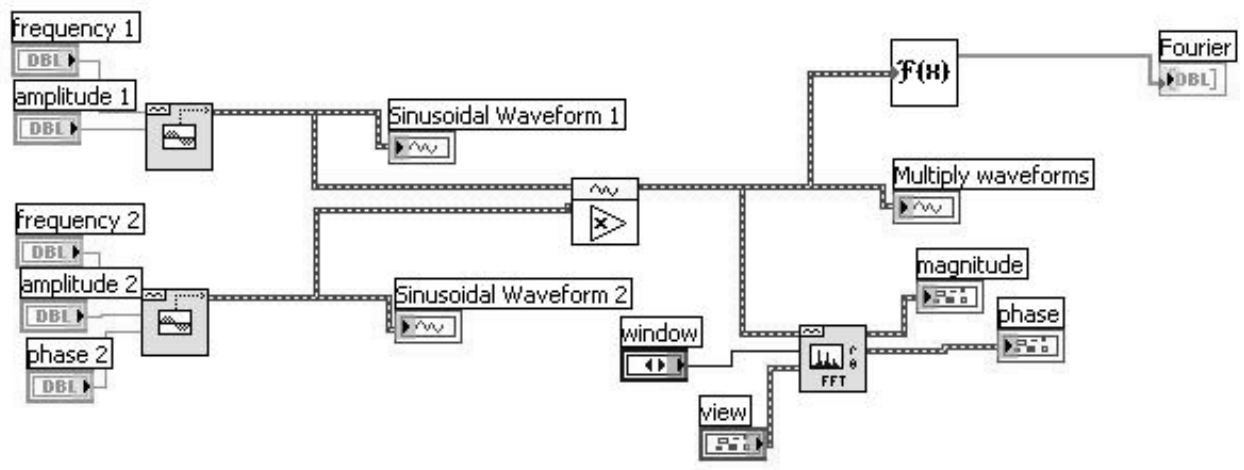
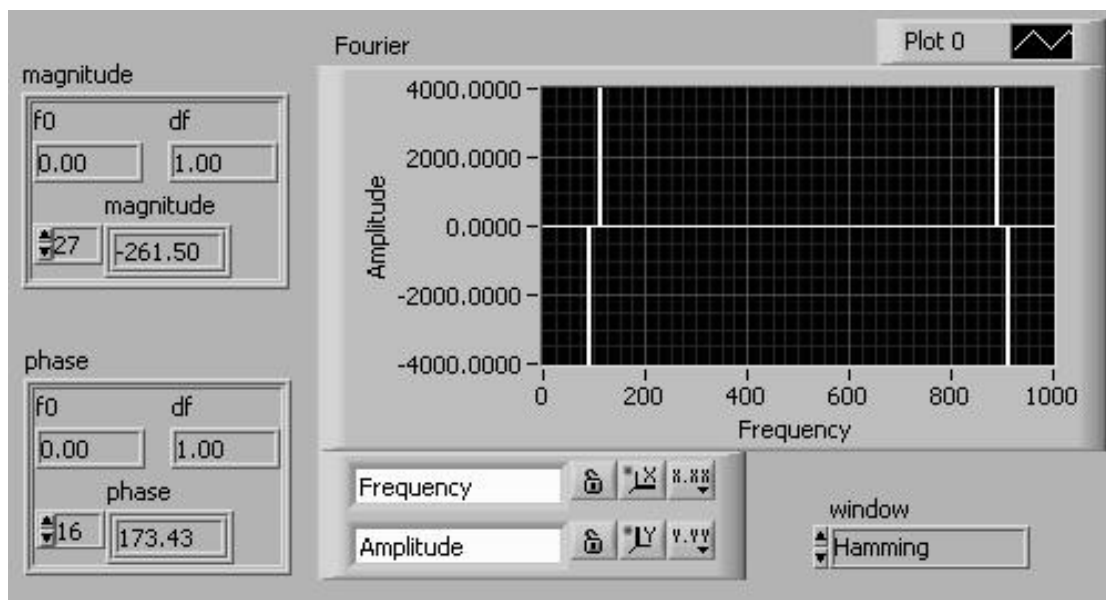


Figura 6-VI pentru aflarea amplitudinii și fazei unui spectru Fourier



Panou frontal pentru faza si amplitudine a unui spectru Fourier

Exemplul 5

Vom crea un instrument virtual pentru a vedea spectrul de putere FFT. Mai jos am amintit noțiunea de spectru de putere FFT.

Spectrul de putere $S_{xx}(f)$ a unei funcții $x(t)$ este definit ca:

$$S_{xx}(f) = |X(f)|^2, \text{ unde } X(f) = F\{x(t)\}.$$

Pentru a crea acest instrument virtual vom parcurge următorii pași:

1. Din paleta de funcții Waveform Generation vom plasa funcțiile Sine Waveform și Uniform White Noise Waveform. Vom crea elemente de control pentru amplitudine și frecvență în cazul funcției Sine Waveform. Pentru funcția de generare a zgomotului vom crea un control pentru amplitudinea zgomotului și o constantă pentru terminalul sampling info (vom introduce valorile 25600, 2048). Vom suma cele două semnale generate de funcțiile prezentate mai sus.
2. Din paleta de funcții Waveform Measurement vom plasa funcția FFT Power Spectrum. Vom crea elemente de control pentru terminalele averaging parameters, window, restart averaging, dB On; elemente de indicare vom avea în cazul terminalelor averaging done, averaging completed. Această funcție calculează media spectrului de putere a semnalului de intrare.
3. Din cadrul panoului frontal vom plasa funcția Waveform Graph la care vom lega ieșirea funcției FFT Power Spectrum.
4. Vom plasa o structură While pentru execuție continuă. Și vom crea un buton de stop (condiția pentru while este stop if true).
5. Se va realiza rutarea după cum se vede în figura 7.

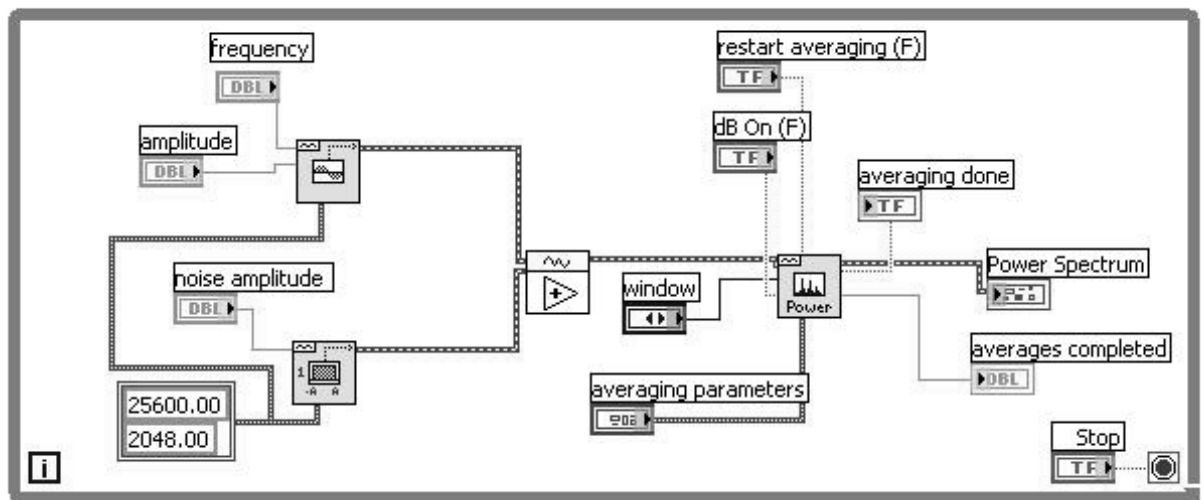
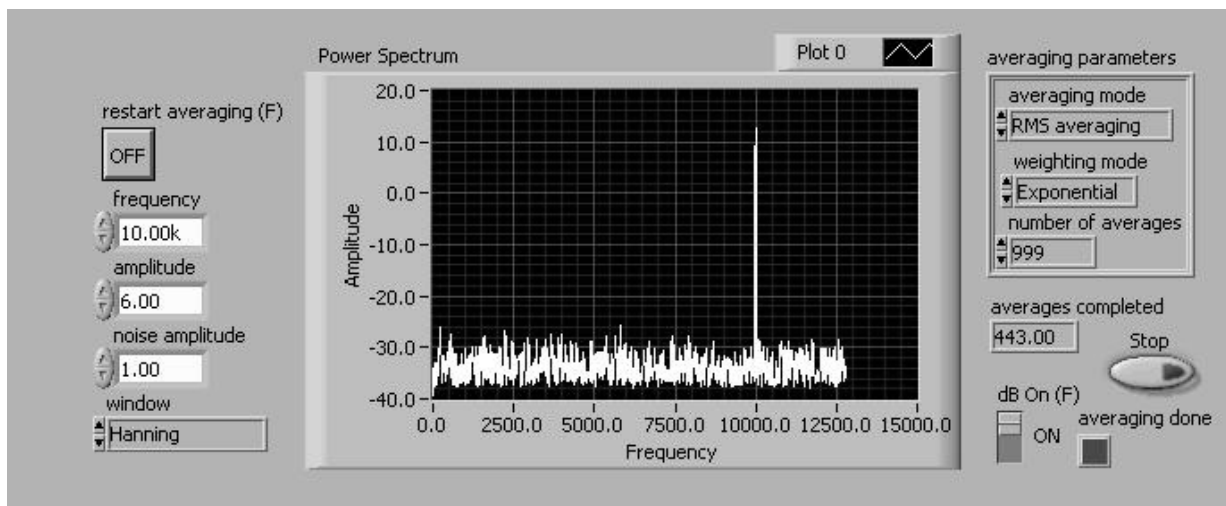


Figura 7-VI pentru afișarea spectrului de putere a unui semnal



Panou frontal pentru afisarea spectrului de putere a unui semnal

Exemplul 6

Cum folosim diagrama Smith ?

Folosim diagrama Smith pentru a studia comportarea unei linii de transmisie. Linia de transmisie este un mediu prin care energia și semnalele sunt transmise. O linie de transmisie poate fi un fir sau atmosferă prin care un semnal este transmis. Fiecare din aceste linii de transmisie are un efect asupra semnalului care va fi transmis. Acest efect numit impedanța liniei de transmisie poate modifica faza sau poate atenua un semnal AC. O linie de transmisie are asociată o impedanță care este o măsură a rezistenței și a reactanței liniei de

transmisie. Putem scrie impedanța ca $z=r+jx$, unde z este un număr complex ce conține ambele componente r -rezistența, x -reactanța.

Putem folosi diagrama Smith pentru a reprezenta impedanțele liniei de transmisie. Reprezentarea constă din trasarea de cercuri de rezistență și reactanță constantă. Putem reprezenta o impedanță dată localizând intersecția a două cercuri apropiate de reactanță și rezistență. O dată ce a fost trasată diagrama Smith putem face potrivire de impedanță, dar în același timp putem calcula și coeficientul de reflexie a unei linii de transmisie, calculând modulul și faza acestuia.

Pentru a realiza acest instrument virtual vom parcurge următorii pași:

1. Din cadrul paletelor de funcții Graphics&Sound –Picture Plots vom plasa funcția Smith Plot. Vom crea următoarele elemente de control corespunzătoare terminalelor: smith plot, dimension, smith plot cosmetics, smith plot grid.
2. Se va crea tot pentru funcția prezentată mai sus, pentru terminalul pict out un element indicator.
3. Din cadrul paletelor de funcții Comparison vom plasa funcția Select (returnează valoarea de la unul din terminale în funcție de valoarea unei variabile booleene). Vom crea elemente de control pentru autoscalare și afisarea diagramei Smith la scara normală. În interiorul acestor elemente de control plasăm elemente de control din cadrul paletelor de funcții Numeric din cadrul panoului frontal. Se vor plasa pentru fiecare element de control (autoscalare și afisare Smith) patru elemente de control (Xmin, Xmax, Ymin, Ymax - a se vedea figura 9.)
4. Se va face rutarea așa cum se vede în figura 8.

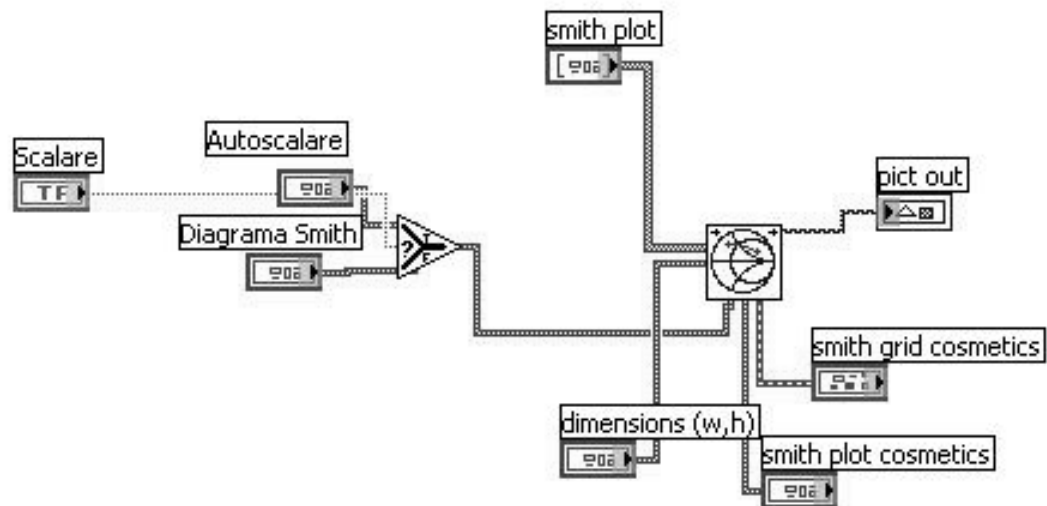
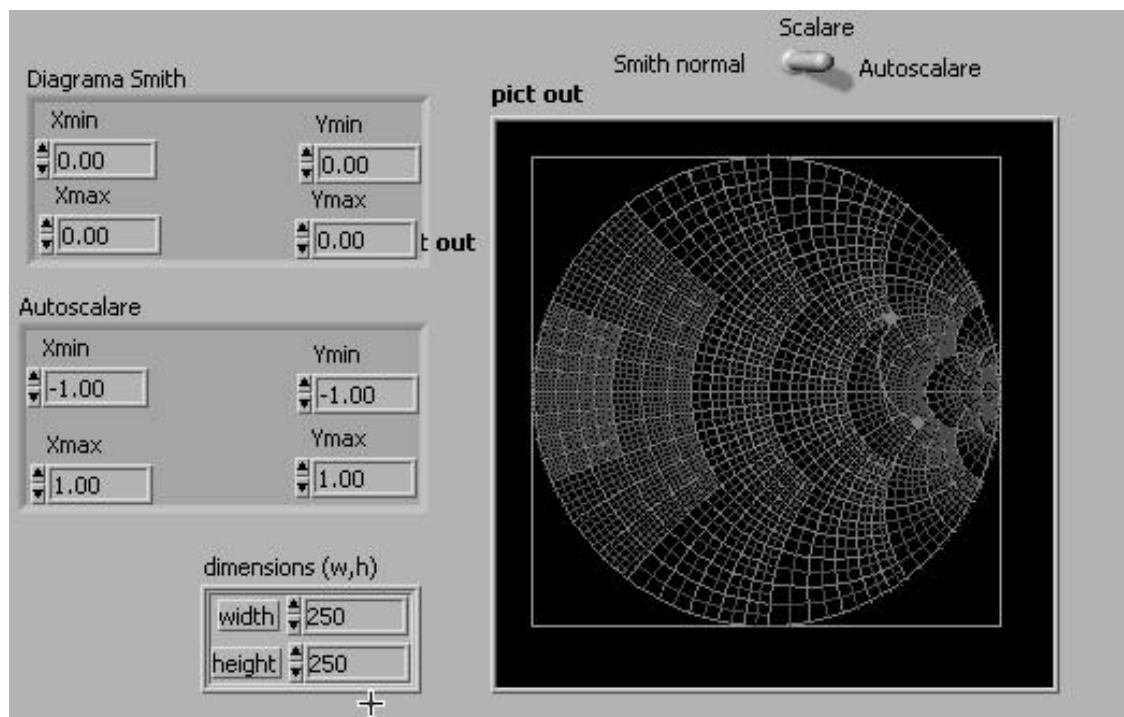


Figura 8-VI pentru reprezentarea de impedante pe diagrama Smith



Panou frontal pentru diagrama Smith

5.3.2. Modul de lucru

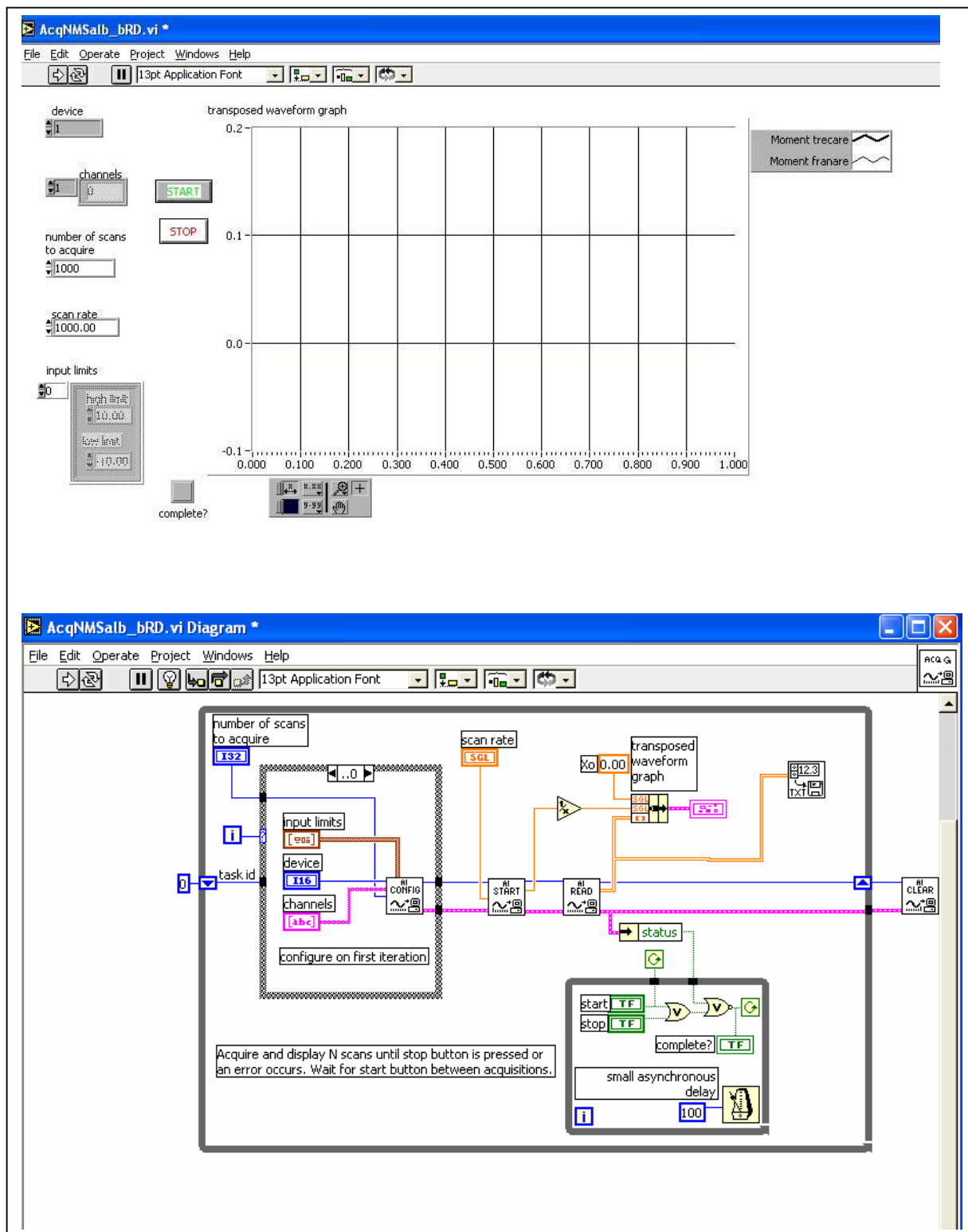
- a. Se deschide LabVIEW;
- b. Familiarizarea cu meniurile amintite în cadrul lucrării de laborator;

5.3.3. Prelucrarea rezultatelor

Realizați cele șase exemple prezentate mai sus.

ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE
Laborator 13

3. Achizitia si salvarea datelor pentru masurarea momentelor de frecare si de franare in rulmentii testati la gripare



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

Laborator 13

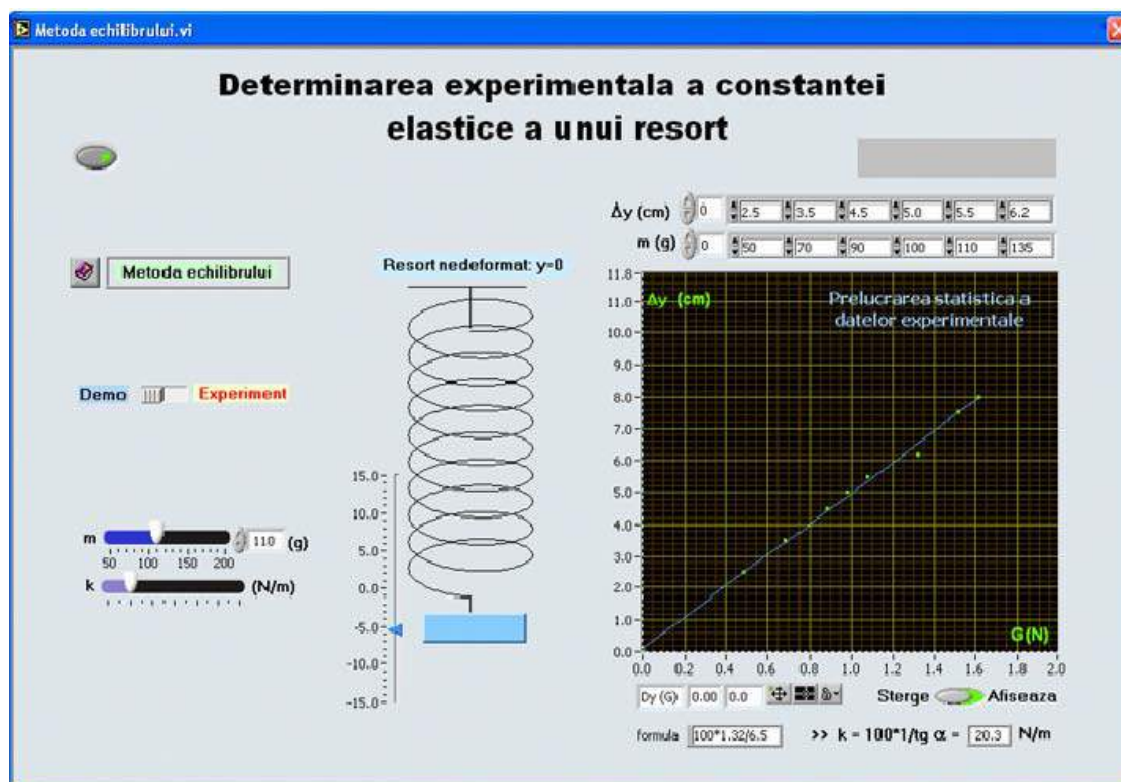
4. Determinarea experimentală a constantei elastice a unui resort. Metoda echilibrului

Pentru determinarea constantei elastice a unui resort au fost proiectate aplicații ce prezintă metode experimentale diferite.

1. Această aplicație propune **metoda echilibrului**, utilizatorul măsurând deformarea resortului pentru corpuri de mase diferite.

Valorile obținute se introduc în tablourile datelor experimentale, graficul aplicației afișând atât punctele experimentale, cât și curba ce descrie dependența deformării resortului de valoarea forței deformatoare (prelucrarea apelează metoda celor mai mici pătrate).

Utilizatorul va folosi cursorul diagramei pentru a determina panta dreptei, apoi va introduce formula de calcul prin care se determină valoarea constantei elastice a resortului. Secțiunea Demo prezintă curba teoretică, precum și soluția problemei propuse.



ACHIZITIA SI PRELUCRAREA DATELOR EXPERIMENTALE

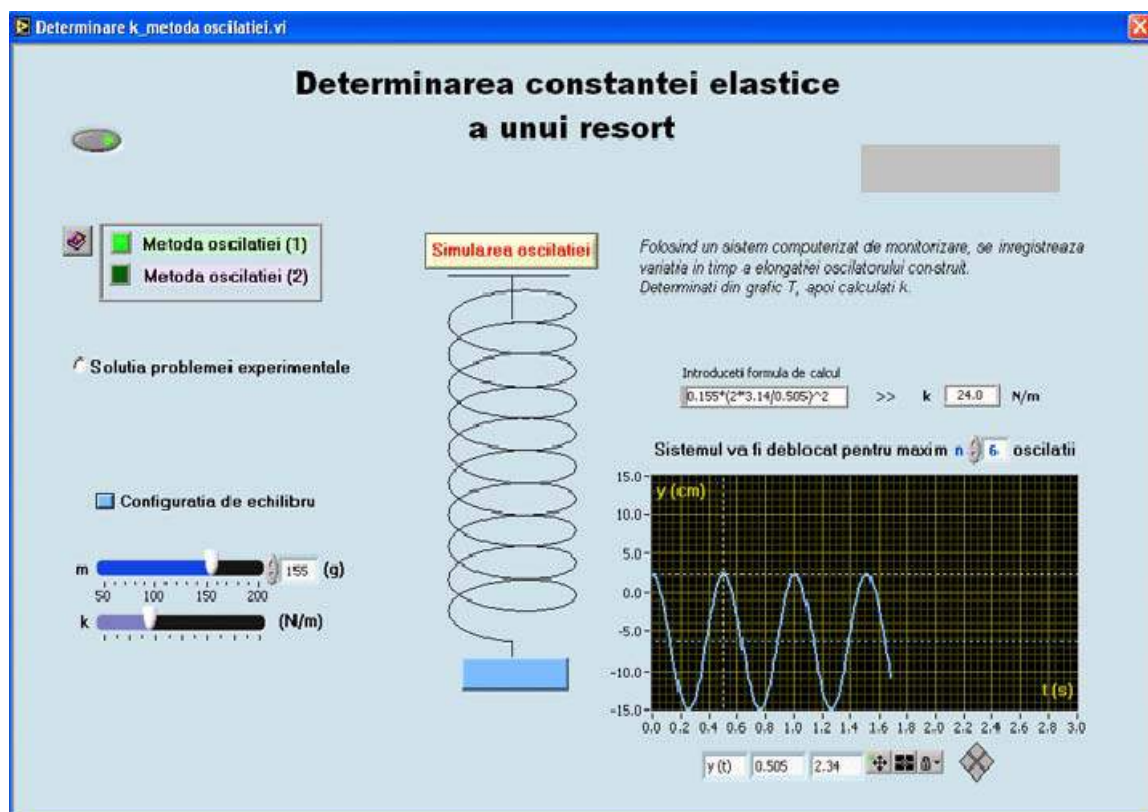
Laborator 13

2. Aplicatia prezinta doua metode experimentale pentru determinarea constantei elastice a unui resort.

Utilizatorul va scoate sistemul din configuratia de echilibru si va determina experimental perioada oscilatiilor :

- pentru prima metoda propusa aceasta valoare va fi determinata din graficul $y(t)$
- pentru cea de a doua metoda se va masura timpul necesar efectuării unui anumit numar de oscilatii).

Folosind apoi expresia perioadei oscilatorului armonic se va calcula valoarea constantei elastice a resortului.



BIBLIOGRAFIE

1. **Bujoreanu, C., Drăgan, B.,** *Monitoring and diagnostic aspects concerning bearings scuffing failure*, Bul. IPI, tom LII(LVI), fasc. 6A, seria Constructii de Mașini, p.103-108, ISSN 1011-2855, 2006
2. **Bujoreanu, C., Cretu, Sp., Drăgan, B.,** *Data acquisition and signal processing in scuffing failure of ball bearings*, Revista de Mecatronica, nr.4, Proc. of COMEFIM 7, p.205-210, 2004
3. **Chen, M. H. ș.a.** *Active control of gear vibration using specially configured sensors and actuators* , Mater. Struct. 9, p.342–350, 2012
4. **Cottet, F., Ciobanu, O.,** *Bazele Programarii in LabVIEW*, Ed. Matrix Rom Bucuresti, 1998
5. **Drăgan, B.,** *Achizitia și procesarea semnalului vibroacustic*, Ed. Politehniun Iasi, 2004
6. **Maier, V., Maier, C.D.,** *LabVIEW in Calitatea Energiei Electrice*, Editura Albastra Cluj-Napoca, 2000
7. **Stefanescu, C., Cupcea, N.,** *Sisteme inteligente de masura și control*, Editura Albastra Cluj-Napoca, 2002
8. *****LabVIEW-** *Data Acquisition/Course Manual/Users Guide*, vol.I-IV, april 1994 Edition.
9. *****LabVIEW**, *The Measurement and Automation*, Catalog, 2004
10. *****LabVIEW** - *Graphical Programming for Instrumentation*, National Instruments, 1996.
11. www.ni.com