

Aplicația II.1

INTRODUCERE ÎN LABVIEW

1.1. Ce este LabVIEW

LabVIEW este un limbaj grafic de programare care permite dezvoltarea de aplicații utilizând pictograme. Spre deosebire de limbajele de programare textuale, în cadrul cărora instrucțiunile sunt cele care determină execuția programului, **LabVIEW**-ul folosește, în locul acestora, fluxul de date evidențiat printr-o prezentare grafică adecvată.

Pe de altă parte, în mod asemănător cu celelalte limbaje de programare, **LabVIEW**-ul conține biblioteci extinse de funcții și subrutine care pot fi utilizate în numeroase aplicații, precum achiziția, prelucrarea, analiza, prezentarea și stocarea datelor. Cu ajutorul unor echipamente de achiziție de semnale de la diverse tipuri de traductoare, **LabVIEW**-ul permite utilizarea performantă a calculatorului pentru măsurarea diverselor mărimi fizice, precum și controlul anumitor procese.

Numele utilitarului provine din faptul că **National Instruments**, corporația care a introdus programele **LabVIEW**, a denumit generic aplicațiile **Instrumente Virtuale**, prescurtat **VI** (din limba engleză **Virtual Instruments**). **LabVIEW**-ul a avut în vedere, în primele sale versiuni, realizarea de programe care să substituie prin intermediul calculatorului o serie de instrumente și aparate electronice, adesea imitând imaginile și modurile de operare ale acestora. De altfel, **LabVIEW** reprezintă prescurtarea denumirii din limba engleză **Laboratory Virtual Instrument Engineering Workbench**.

În **LabVIEW** un **VI** este constituit cu ajutorul următoarelor trei componente:

- **Panoul frontal**, care servește ca interfață cu utilizatorul;
- **Diagrama bloc**, care conține sub formă grafică codurile sursă care realizează funcționarea **VI**-ului;
- **Pictograma** (icon) și **panoul conector**, prin intermediul cărora este identificat **VI**-ul și se asigură posibilitatea ierahizării, adică a interconectării în cadrul altui **VI**, constituind un **subVI** (asemănător cu o subrutină).

LabVIEW dispune de numeroase instrumente care asigură o configurare ușoară a unui anumit tip de **VI**. De asemenea, include sute de exemple de **VI**-uri corespunzătoare unor variate domenii de aplicații, pe care utilizatorul poate să le folosească ca atare și să le încorporeze în **VI**-uri de mai mare complexitate potrivit scopului urmărit, sau le poate modifica pentru a le adapta particularităților aplicației.

Utilizarea completă și eficientă a multiplelor resurse și facilități pe care le oferă acest mediu de programare grafică necesită studierea atentă a documentațiilor

cuprinse în pachetul **LabVIEW** și care pot fi accesate din meniul **Help**. Dintre acestea, pentru început, cele mai importante sunt: **LabVIEW Tutorial** și **LabVIEW User Manual**.

1.2. Deschiderea unei aplicații LabVIEW

Accesând **LabVIEW** din lista de programe, sau executând click pe pictograma aferentă aflată în **Desktop**-ul calculatorului, pe ecranul monitorului apare caseta de dialog introductivă din fig.1.1, în care, apăsând pe butonul **New VI**, se vor deschide principalele două ferestre - **Panoul frontal** și **Diagrama bloc** - care vor permite dezvoltarea unei noi aplicații. În cazul în care se dorește deschiderea unui **VI** realizat anterior se apasă butonul **Open VI** și se caută fișierul în care a fost salvat respectivul **VI**.

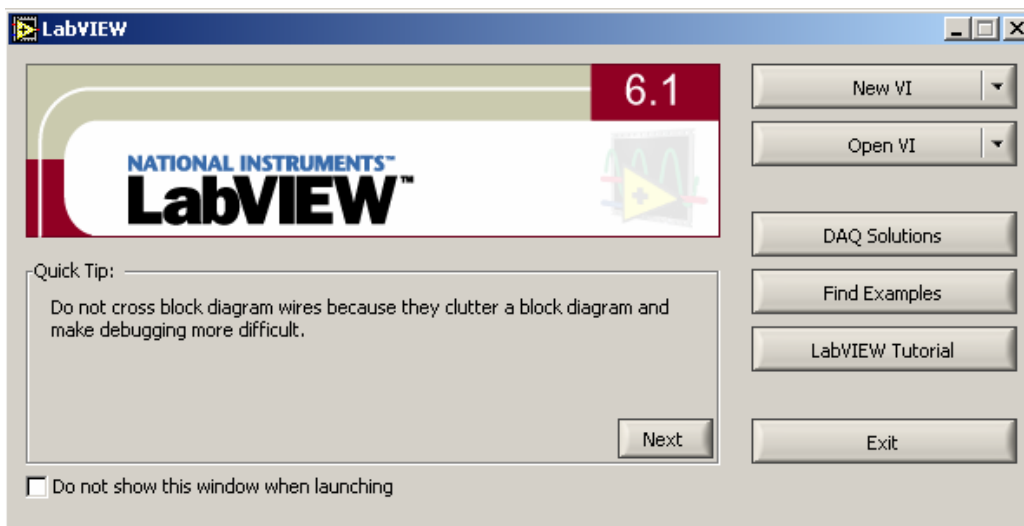


Fig.1.1. Caseta introductivă de lansare a programului **LabVIEW 6.1**

Panoul frontal

În fig.1.2.a se poate vedea fereastra care afișează **Panoul frontal** al unui **VI** care, așa cum se va arăta ulterior, constă din dispozitivul de pornire/oprire al unui generator de numere aleatoare și un display pe care se vizualizează sub o formă scalată numerele generate.

După cum se poate observa pe fig.1.2.a structura ferestrei panoului frontal cuprinde următoarele componente:

- *Bara de nume*, aflată la partea superioară, este destinată să afișeze numele atribuit **VI**-ului, de obicei la salvarea acestuia. Inițial programul stabilește denumirea **Untitled 1**;

- *Bara cu meniurile principale*, situată sub bara de nume, conține meniurile principale ale panoului frontal, cu denumiri în mare parte similare ca și la alte medii de programare;

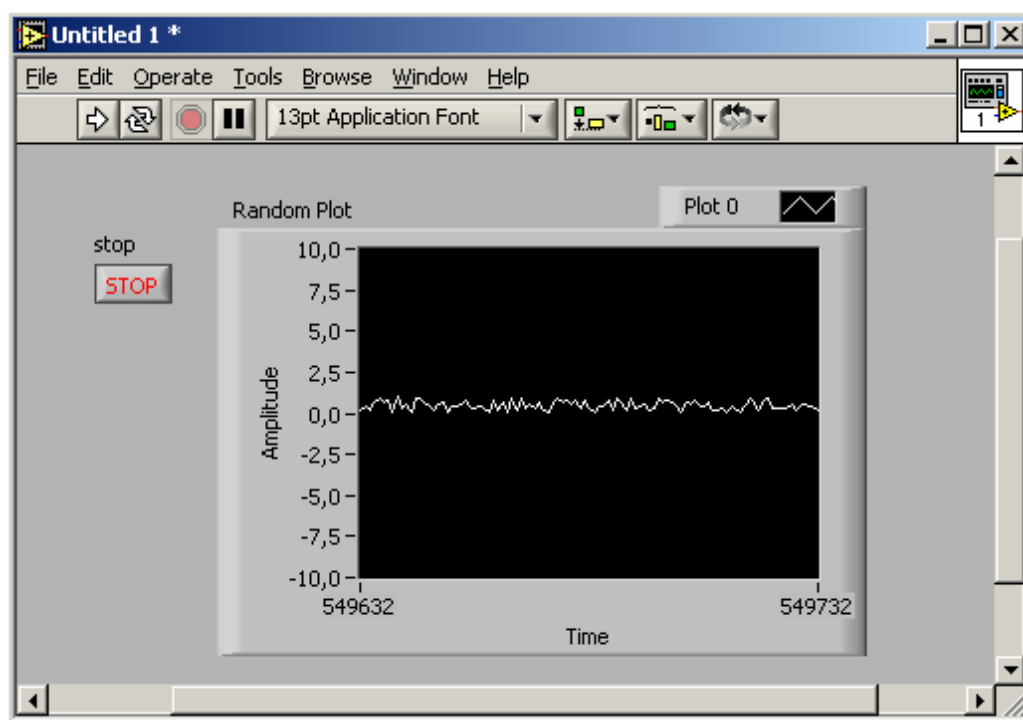


Fig.1.2.a. Panoul frontal al unui VI

- *Bara de comenzi* conține butoanele de comandă a execuției programului VI-ului și de aranjare a obiectelor din panoul frontal;
- *Spațiul de lucru*, situat sub bara de comenzi, ocupă cea mai mare parte a panoului frontal întrucât în acest cadru vor fi plasate obiectele de lucru ale VI-ului;
- *Barele cu cursoare*, dispuse în partea de jos, respectiv în partea dreaptă a spațiului de lucru, servesc pentru navigarea în acest spațiu.

Pe spațiul de lucru al **panoului frontal** se pot dispune dispozitive de control (controale) - de exemplu butoane, taste, cursoare etc - și dispozitive de afișare (indicatoare) de diverse tipuri - LED-uri, scale de instrumente de măsurat, display-uri grafice. Prin intermediul acestora se asigură interfața interactivă cu utilizatorul, controalele simulează intrările aplicate VI-ului și introduc date în diagrama bloc, iar indicatoarele simulează ieșirile VI-ului și afișează datele furnizate de către diagrama bloc rezultate din execuția programului aferent. Atât controalele cât și indicatoarele imită, de regulă, forma și modul de acționare corespunzătoare unor instrumente reale, diferența constând în aceea că ele sunt manipulate cu ajutorul mouse-ului sau a tastaturii. În cazul panoului frontal din fig.1.2.a se poate vedea butonul de **Stop**, care reprezintă controlul, și display-ul grafic **Random Plot** care reprezintă indicatorul.

Diagrama bloc

Fereastra care afișează **diagrama bloc** corespunzătoare VI-ului, al cărui panou frontal a fost redat anterior, este reprezentată în fig.1.2.b.

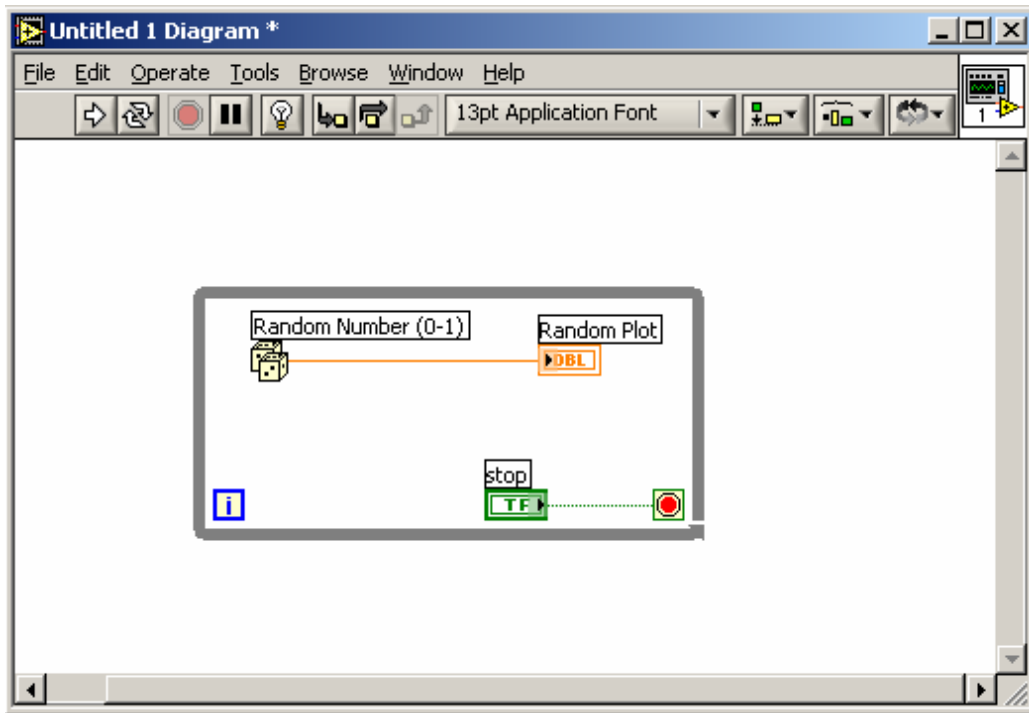


Fig.1.2.b. Diagrama bloc a unui VI

Diagrama bloc cuprinde, în spațiul de lucru, reprezentarea grafică a funcțiilor corespunzătoare obiectelor de pe panoul frontal și ilustrează modul în care circulă fluxul de date și operațiile de prelucrare ale acestora. Astfel, diagrama bloc conține codurile sursă ale programului VI-ului. În cadrul diagramei bloc obiectele de pe panoul frontal apar sub formă de terminale, care constituie porturi de intrare și de ieșire prin care se efectuează schimburile de informații cu panoul frontal. Obiectele din diagrama bloc, care evidențiază operațiile efectuate de către programul VI-ului, se numesc **noduri** (sunt asemănătoare cu operatorii/subrutinele din programele textuale). Transferul datelor între obiectele din diagrama bloc este asigurat prin legături de conexiune între acestea (**wires**) având culori, stiluri și grosimi diferite, în funcție de tipurile de date pe care le vehiculează. Orice legătură pornește de la o singură sursă de date, dar poate fi conectată la unul sau mai mulți receptori. În diagrama bloc se mai întâlnesc și așa-numitele structuri (**structures**), care sunt reprezentări grafice ale buclelor și instrucțiunilor de control de la limbajele de programe textuale.

În fig.1.2.b **generatorul de numere aleatoare** este simbolizat sub forma unei perechi de zaruri având eticheta **Random Number**. Butonului de stop din panoul frontal îi corespunde terminalul de intrare **TF** având eticheta **Stop** și este de natura unei variabile booleene care pornește sau oprește generatorul de numere aleatoare. Display-ului de pe panoul frontal îi corespunde terminalul de ieșire **DBL** având eticheta **Random Plot**. Generatorul de numere aleatoare, care constituie sursa de

generare a datelor, este conectat cu destinația reprezentată de terminalul de ieșire prin care datele se transmit la indicatorul de pe panoul frontal.

În fig.1.2.a și fig.1.2.b panoul frontal și diagrama bloc au fost prezentate prin două imagini separate din considerente de evidențiere a obiectelor aflate în cele două spații de lucru. În fapt, la deschiderea unei aplicații, cele două ferestre apar parțial suprapuse, în prim plan panoul frontal și în plan secund diagrama bloc. Pozițiile celor două ferestre se pot schimba, de exemplu în situația precedentă executând click pe suprafața vizibilă a diagramei bloc acesta trece în primul plan și panoul frontal în cel secund.

Pictograma (icon-ul) este o reprezentare grafică semnificativă a **VI**-ului, constând din imagini sau text, ori combinații ale ambelor. Fiecare **VI** se identifică astfel printr-o asemenea pictogramă de mici dimensiuni, care este afișată într-o casetă dispusă în colțul din dreapta sus în ferestrele panoului frontal și diagramei bloc. Această pictogramă servește și pentru reprezentarea sintetică ca **subVI** în diagrama bloc a altui **VI**. În același scop, al utilizării ca **subVI**, servește și **panoul conector**, care conține un set de terminale prin care se definesc intrările și ieșirile care se interconectează, de regulă câte un terminal pentru fiecare control și indicator aflat pe panoul frontal.

1.3. Prezentarea principalelor mijloace pentru construirea unei aplicații LabVIEW

Principalele mijloace pentru construirea unei aplicații **LabVIEW** sunt: meniurile, butoanele și paletele.

Meniurile

Pe bara de meniuri sunt listate șapte tipuri, aceleași pentru ambele ferestre, având atât opțiuni comune tuturor aplicațiilor din alte medii de programare, cât și opțiuni specifice **LabVIEW**. Denumirile și conținutul meniurilor sunt descrise mai jos.

File - este meniul care servește pentru deschiderea, închiderea, salvarea unei noi aplicații sau a uneia existente, setarea imprimantei și lansarea comenzii de tipărire, descrierea proprietăților **VI**, listarea fișierelor recent deschise, lansarea comenzii de ieșire.

Edit - este meniul destinat efectuării operațiilor curente de editare, precum: anulează/reface ultima comandă, șterge/copiază obiectul selectat în clipboard, plasează conținutul clipboard-ului în fereastră, șterge definitiv obiectul selectat, lansează comanda, găsește (un anumit **VI**) și afișează rezultatele, particularizează controalele, scalează dimensiunile obiectului cu cele ale panoului, importă figuri dintr-un anumit fișier, șterge conexiunile întrerupte, crează **subVI**-uri, permite modificarea timpului de execuție.

Operate - este meniul prin care se exercită acțiuni asupra derulării programului cum sunt execuția/oprirea rulării **VI**-ului curent, întreruperea rulării când este apelat **VI**-ul, tipărirea panoului în completare, trecerea datelor într-un fișier în

aceeași situație, afișarea opțiunilor de trecere a datelor în fișier, transformarea valorilor curente în valori prestabilite, aducerea tuturor controalelor și indicatoarelor la valorile prestabilite, comutarea între regimurile de execuție și editare, descărcarea aplicației, motorul de timp real, comutarea execuției obiectivului, conectarea la panoul de acces la distanță.

Tools - este meniul prin care se pot utiliza diverse facilități ale **LabVIEW** precum: browser-ul instrumentelor National Instruments, accesarea driver-elor respectivelor instrumente, afișarea canalelor și expertului (wizard) pentru achiziția datelor, înregistrarea datelor și gestionarea supervizării, compararea atributelor și ferestrelor, controlul codurilor sursă, istoria **VI**-ului, numele utilizatorului, construcția aplicației/partajarea bibliotecilor, gestionarea și editarea bibliotecii **VI**-ului, controlul și gestionarea conexiunilor, instrumente **WEB**, operații avansate (șiruri de date, erori etc), opțiuni privind rutele de stocare și obiectivele de timp real.

Browse - este meniul de căutare pentru ierahizarea **VI**-urilor, apelarea **subVI**-urilor și punctele de întrerupere.

Window - este meniul care permite schimbarea ferestrelor, adică vizualizarea ferestrei diagramei bloc atunci când se lucrează cu panoul frontal și invers, afișarea paletei controalelor (când se lucrează cu fereastra panou frontal), respectiv a paletei funcțiilor (când se lucrează cu fereastra diagramă bloc), afișarea conținutului memoriei clipboard și a listei erorilor, dispunerea ferestrelor alăturat sau una deasupra celeilalte, ocuparea întregului ecran, prezentarea ferestrelor **VI**-ului deschis în sesiunea curentă.

Help - este un meniu cu rol de ajutor ca și în cazul altor medii de programare, care conține multiple opțiuni precum: afișarea/blocarea **Help**-ului contextual care oferă ajutor cu privire la obiectele și operațiile de lucru din ferestre, informații privind pachetul de documentații și exemple **LabVIEW**, afișarea adreselor de **WEB** din domeniu, explicarea erorilor de programare a **VI**-ului curent, informații despre înregistrarea datelor și controlul supervisor, **Help** pentru **VI**-uri cu mobilitate flexibilă, informații cu privire la procesarea imaginilor (**IMAQ vision**) cu **VI**-uri și exemple, ajutor privind driver-ele pentru anumite instrumente avansate (**NI-switches** și **NI-IMAQ**), informații cu privire la versiunea **LabVIEW**.








Butoanele

Pe bara de butoane din fereastra panoului frontal se pot vedea 8 butoane și casete ale căror denumiri se vizualizează poziționând mouse-ul deasupra lor. Semnificațiile acestora, în ordinea de la stânga la dreapta, sunt prezentate în cele ce urmează:

1. Butonul **Run (Execută)** determină execuția o singură dată a programului de funcționare a **VI**-ului. Atunci când rulează programul butonul își schimbă forma.

2. Butonul **Run Continuously (Execuție continuă)** face ca programul să ruleze în mod continuu, fără oprire. Pe parcursul acestui mod de rulare își schimbă forma atât acest buton cât și precedentul.

3. Butonul **Abort Execution (Oprirea execuției)** servește pentru oprirea programului și este activ numai pe perioada rulării programului când își schimbă culoarea.

-  conectare, utilizată în special în diagramă;
-  afișare a meniurilor proprii;
-  deplasare a imaginii într-o fereastră (scroll);
-  inserarea de puncte de oprire a rulării (breakpoint);
-  inserarea de puncte de probă;
-  selectarea de culori;
-  colorare.

După cum rezultă din prezentarea de mai sus, uneltele permit executarea de operații necesare pentru crearea, modificarea și corectarea VI-urilor. Este de observat că în momentul în care se execută click pe una dintre uneltele pointerul mouse-ului ia forma acesteia.

Paleta Controls (Controale) se poate afișa numai atât timp cât este deschisă fereastra panoului frontal (este aferentă numai acesteia). În fig.1.4 este reprezentată această paletă într-o formă puțin diferită de aceea sub care apare pe ecran, în sensul că

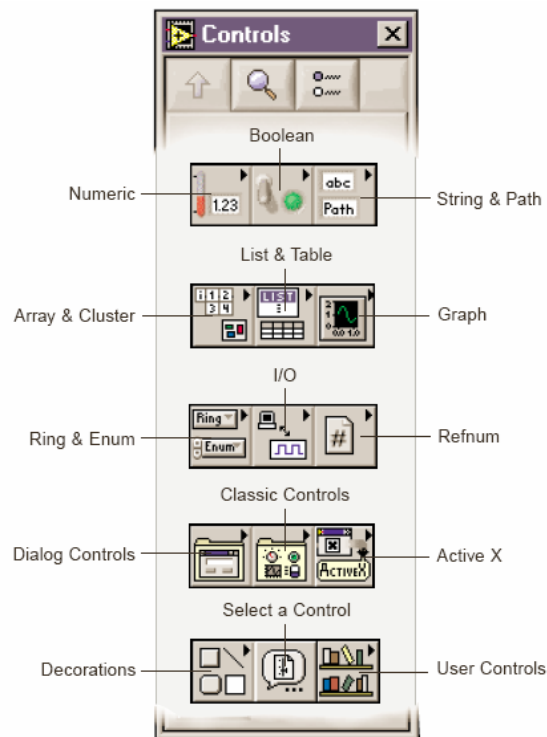


Fig.1.4. Paleta de controale (**Controls**)

liniile cu butoane sunt distanțate pentru a li se putea tipări titlurile în limba engleză.

După cum se poate vedea, această paletă conține un număr de 15 butoane principale, care de fapt reprezintă subpalete în cadrul cărora se asigură accesul la obiectele din categoria respectivă, ce pot fi de două feluri: controale și indicatoare. Subpaletele se accesează executând click pe butonul corespunzător categoriei dorite și la partea superioară a paletei apare numele subpaletii deasupra căreia s-a aflat pointerul mouse-ului.

Paleta **Controls** poate fi accesată și executând click cu tasta dreapta a mouse-ului pe suprafața de lucru a panoului. În acest caz subpaletele se deschid prin poziționarea pointerului pe butonul dorit.

La partea de sus, sub denumirea paletii, se află 3 butoane care servesc pentru navigare în cadrul paletii.

Pentru a exemplifica modul în care se prezintă o subpaletă atunci când este accesată, în figurile următoare sunt redată elementele de control și indicatoarele pentru valori scalare conținute în subpaletele **Boolean** și **Numeric**.



Fig.1.5. Subpaleta **Boolean**

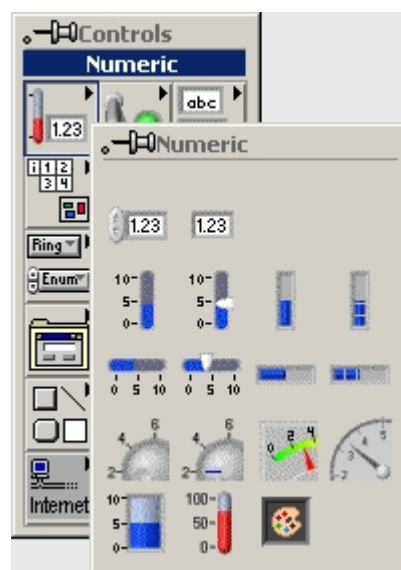


Fig.1.6. Subpaleta **Numeric**

Pe fig.1.5 și fig.1.6 se pot observa obiectele de tip control, respectiv de tip indicator, care sunt utilizate pentru crearea de noi **VI**-uri, sau pentru modificarea unora existente. Fiecare subpaletă conține astfel de elemente adecvate categoriei pe care o reprezintă. Unele subpalete au în partea dreapta sus butoane sub formă de vârf de săgeată prin intermediul cărora se pot deschide, la rândul lor, alte subpalete secundare din care se pot alege alte tipuri de controale și indicatoare. Obiectele necesare se selectează din subpalete cu ajutorul mouse-ului și se trag în spațiul de lucru cu tasta stângă apăsată până în poziția dorită.

Paleta Functions (Funcții) se poate afișa numai atât timp cât este deschisă fereastra diagramei bloc (este aferentă numai acesteia). În fig.1.7 este reprezentată această paletă într-o formă puțin diferită de aceea sub care apare pe ecran, în sensul că liniile cu butoane sunt distanțate pentru a li se putea tipări titlurile în limba engleză.

După cum se poate vedea, această paletă conține un număr de 23 de butoane, care de fapt reprezintă subpalete în cadrul cărora se asigură accesul la funcțiile și aplicațiile din categoria respectivă și 3 butoane de navigație. Subpaletele se accesează executând click pe butonul corespunzător categoriei dorite, și la partea superioară a paletei, apare numele subpaletii deasupra căreia s-a aflat pointerul mouse-ului.

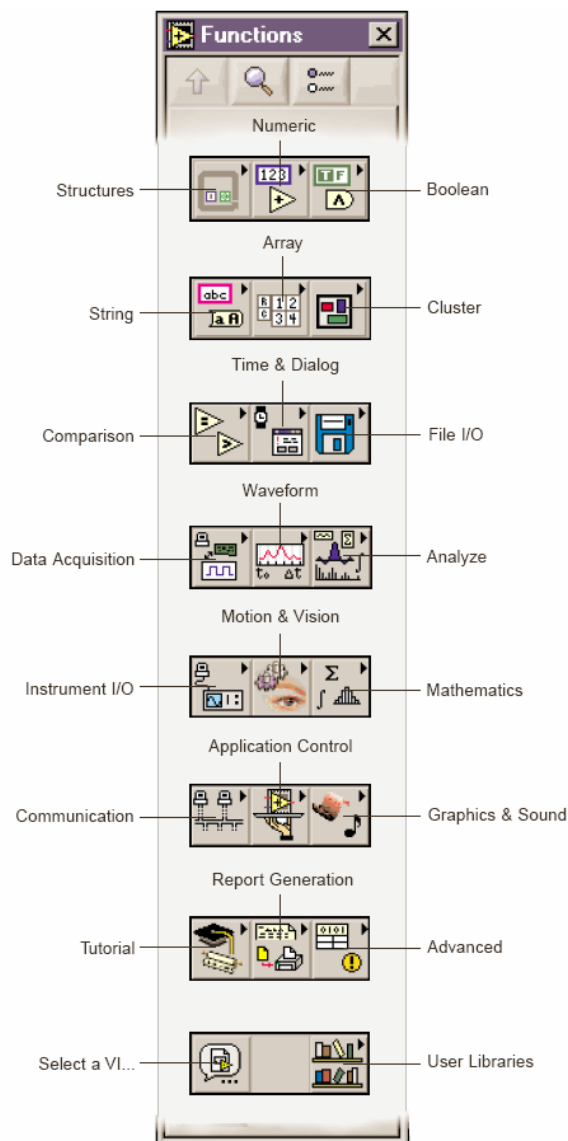


Fig.1.7. Paleta de funcții (**Functions**)

Paleta **Functions** poate fi accesată și executând click cu tasta dreaptă a mouse-ului pe suprafața de lucru a panoului. În acest caz subpaletele se deschid prin poziționarea pointerului pe butonul dorit.

Pentru a exemplifica modul în care se prezintă o subpaletă atunci când este accesată, în fig.1.8 și fig.1.9 sunt redată funcțiile pentru valori scalare conținute în subpaletele **Numeric Constant** și **Boolean**.

Pe fig.1.8 și fig.1.9 se pot observa funcțiile numerice, respectiv cele booleene, care sunt utilizate pentru crearea de noi **VI-uri**, sau pentru modificarea unora existente. Fiecare subpaletă conține asemenea elemente adecvate unei game largi de funcții și

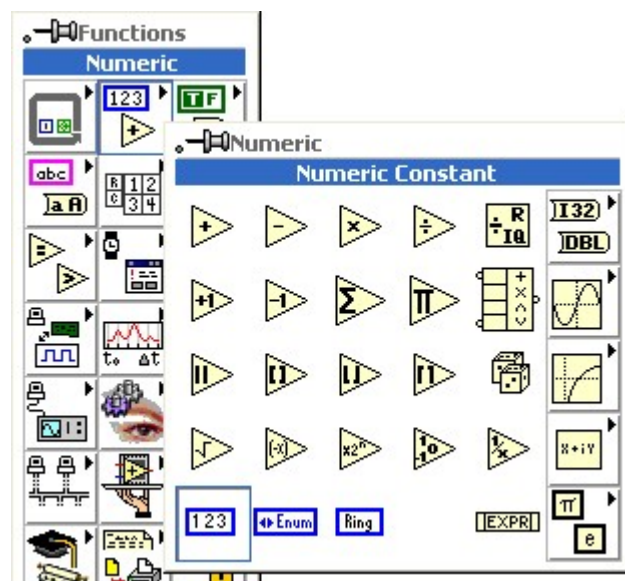


Fig.1.8. Subpaleta **Numeric Constant**

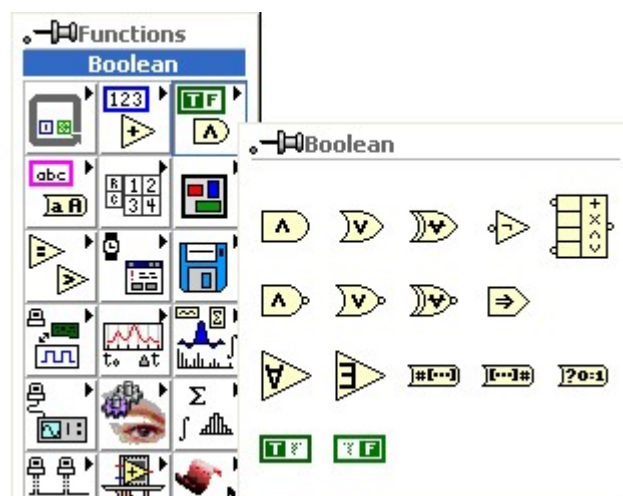


Fig.1.9. Subpaleta **Boolean**

aplicații, de la operații aritmetice simple până la deschiderea/închiderea de fișiere, organizarea coloanelor etc. Unele subpalete au în partea dreaptă sus butoane sub formă de vârf de săgeată, prin intermediul cărora se pot deschide, la rândul lor subpalete secundare, din care se pot alege alte funcții și aplicații. Funcțiile necesare se selectează din subpalete cu ajutorul mouse-ului și se trag în spațiul de lucru al diagramei cu tasta stângă apăsată până în poziția dorită (la fel ca în cazul controalelor).

1.4. Configurarea obiectelor din panoul frontal

Structurarea panoului frontal constă în selectarea din paleta **Controls** și dispunerea convenabilă în spațiul de lucru a controalelor și indicatoarelor necesare ca elemente de interfață pentru introducerea datelor de intrare, respectiv pentru citirea datelor de ieșire rezultate din funcționarea VI-ului proiectat. În vederea unei mai bune utilizări a spațiului de lucru, precum și pentru o creștere a posibilităților de manipulare și de observare a obiectelor, de regulă apare necesitatea configurării acestora, care constă în operații de reordonare, de modificare a dimensiunilor - manual sau automat - în raport de dimensiunile ferestrei, de colorare a obiectelor, de plasare a unor etichete și valori ale datelor de intrare, de introducere a navigației de la tastatură etc. Toate aceste operații de configurare pot fi efectuate folosind **meniurile obiect (meniuri proprii)**, care se deschid atunci când se execută click cu tasta dreaptă a mouse-ului, pointerul acestuia fiind plasat deasupra obiectului.

Pentru a ilustra modul în care meniurile proprii pot fi utilizate pentru configurare, în fig.1.10 sunt prezentate unele opțiuni comune pentru controalele cu valori numerice, booleene sau alfanumerice, iar în continuare sunt explicate semnificațiile acestora..

Visible Items ► Label - afișează sau ascunde eticheta obiectului. Conținutul etichetei poate fi modificat cu unealta de editare a textelor;

Visible Items ► Caption - afișează sau ascunde eticheta suplimentară a elementului. Conținutul etichetei poate fi, de asemenea, modificat cu unealta de editare a textelor;

Find Terminal - face trecerea în diagramă și indică terminalul elementului prin

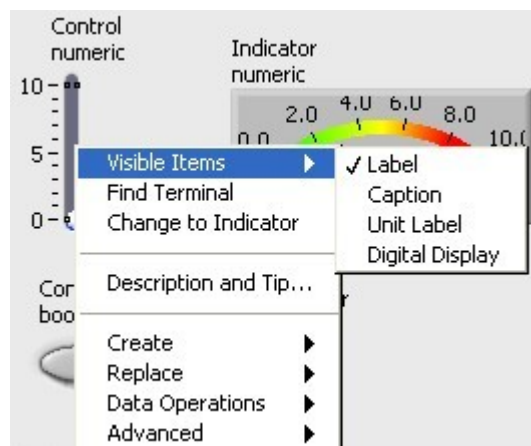


Fig.1.10. Meniuri proprii de configurare

încadrarea acestuia într-un chenar cu linie întreruptă animată;

Change to Indicator (sau **Change to Control**) - schimbă tipul elementului din control în indicator sau invers;

Description and Tip... - permite introducerea unei scurte descrieri referitoare la element și a unui text ajutător (**Tip**), care apare atunci când cursorul mouse-ului este poziționat deasupra elementului;

Create► - creează în diagramă entități suplimentare atașate elementului de pe panou (vor fi discutate ulterior);

Replace► - afișează paleta de controale, astfel încât utilizatorul poate selecta un element cu care să îl înlocuiască pe cel curent;

Data Operations► - permite stabilirea valorii implicite a elementului (**Make Current Value Default**), reinițializarea unui element la valoarea sa implicită (**Reinitialize to Default Value**) sau operații de copiere a valorii elementului (**Cut, Copy, Paste**);

Advanced► - modalități de configurare avansată a elementelor (vor fi discutate în lucrări ulterioare).

Meniurile proprii cuprind și opțiuni specifice obiectelor prin intermediul cărora se oferă facilități de configurare impuse de particularitățile acestora. Ca exemple în acest sens se pot menționa pentru controalele numerice **Data Range** - care permite stabilirea unui anumit interval de variație și **Format & Precision** - pentru specificarea unui anumit mod de afișare a valorii sau, pentru controalele booleene, **Mechanical Action** - care oferă mai multe opțiuni privind modul în care se comportă controlul respectiv la apăsarea tastei mouse-ului.

Un alt aspect al configurării îl constituie modificarea dimensiunilor, gruparea și fixarea obiectelor de pe panou într-un anumit cadru potrivit unor criterii avantajoase pentru utilizator. Modificarea dimensiunilor (mărimii) obiectelor se poate efectua observând că la deplasarea unelei de poziționare peste obiectul selectat, dacă acesta este de formă rectangulară, în colțurile dreptunghiului de încadrare apar forme triunghiulare de redimensionare asupra cărora se poate acționa cu mouse-ul. Gruparea și fixarea obiectelor se poate face cu butonul **Reorder** selectând **Group** și **lock** din meniul acestuia. De asemenea, se pot utiliza casetele de aliniere din bara de butoane cu ajutorul cărora se pot executa alinieri pe verticală sau pe orizontală.

O scalare automată a obiectelor în raport cu mărimea panoului frontal se poate obține și folosind opțiunea **Scale Object With Panel** din meniul **Edit**.

Date fiind multitudinea de controale și de indicatoare precum și cerințele de configurare foarte variate, există numeroase proceduri care se vor însuși pe parcursul executării diverselor aplicații.

1.5. Obiectele din diagrama bloc și relația cu cele din panoul frontal

Obiectele utilizate în diagrama bloc sunt denumite terminale, noduri sau funcții, legate între ele prin conexiuni (wires).































Terminalele

Obiectele din panoul frontal apar în diagrama bloc sub formă de terminale. Astfel, terminalele reprezintă porturi de intrare și de ieșire, care permit schimbul de

informații între panoul frontal și diagrama bloc. Datele care se introduc prin controalele dispuse pe panoul frontal sunt transmise în diagrama bloc prin terminale de intrare, iar rezultatele funcționării ies din diagrama bloc și sunt transmise indicatoarelor din panoul frontal prin terminalele de ieșire.

Este de observat că există mai multe tipuri de terminale. În general, un terminal este orice loc în care se poate atașa o conexiune. Există astfel terminale ale controalelor și indicatoarelor, terminale ale nodurilor, terminale de tip constante și terminale specializate ale structurilor. Pe de altă parte, terminalele care furnizează date - cum sunt terminalele controalelor, terminalele de ieșire ale nodurilor (output) și constantele - sunt denumite terminale sursă, iar terminalele care recepționează date - cum sunt cele ale indicatoarelor, cele de intrare ale nodurilor (input) - sunt terminale destinație. În diagrama bloc terminalele sunt reprezentate prin dreptunghiuri cu laturile duble având în interior un cod și un vârf de săgeată.

În tabelul care urmează sunt listate tipurile de terminale pentru date numerice, booleene și șiruri:

Control	Indicator	Data Type	Color
		Single-precision floating-point numeric	Orange
		Double-precision floating-point numeric	Orange
		Extended-precision floating-point numeric	Orange
		Complex single-precision floating-point numeric	Orange
		Complex double-precision floating-point numeric	Orange
		Complex extended-precision floating-point numeric	Orange
		Signed 8-bit integer numeric	Blue
		Signed 16-bit integer numeric	Blue
		Signed 32-bit integer numeric	Blue
		Unsigned 8-bit integer numeric	Blue
		Unsigned 16-bit integer numeric	Blue
		Unsigned 32-bit integer numeric	Blue
		Enumerated type	Blue
		Boolean	Green
		String	Pink

După cum se poate observa, terminalele controalelor se pot deosebi prin aceea că au linia exterioară a dreptunghiurilor mai groasă decât la cele ale indicatoarelor, iar vârfurile de săgeată sunt plasate pe latura din dreapta și orientate către exterior, pe când la indicatoare sunt plasate pe latura din stânga și orientate către interior. Simbolurile înscrise în dreptunghiuri și culorile sunt diferite în funcție de tipurile de date și de caracteristicile de reprezentare ale acestora.

Terminalele de tip constante sunt acelea care introduc date cu valori fixe direct în diagrama bloc și pot fi de două tipuri: constante universale și constante definite de utilizator.

Ca exemple de constante universale cu valori numerice fixe sunt cele întâlnite în mod curent în matematică și fizică (∞ , e , π , ...). Ele se pot găsi în paleta de funcții executând **Functions >> Numeric >> Additional Numeric Constants**.

Constantele definite de utilizator sunt constante stabilite în timpul editării și a căror valoare nu poate fi modificată în timpul execuției VI-ului. Paleta **Functions** conține constante organizate în diverse subpalete după tipuri de date, de exemplu numerice, booleene, șiruri etc, dar și după alte criterii, de tip enumerare, cutie colorată, listă de nume și altele. O constantă definită de utilizator se creează executând click pe un terminal cu tasta dreaptă a mouse-ului și selectând **Create Constant** din meniul propriu. Modificarea constantei se face utilizând unealta de operare cu care se face click pe constanta respectivă. Se poate crea și o etichetă utilizând opțiunea **Label** din meniul propriu.

Nodurile

Nodurile sunt obiecte în diagrama bloc care au intrări și /sau ieșiri și care execută anumite operații în funcționarea VI-ului. Ele sunt asemănătoare declarațiilor, operatorilor, funcțiilor, subrutinelor de la limbajele de programare convenționale. **LabVIEW** conține următoarele 7 tipuri de noduri:

- **Functions (funcții)** - elemente de execuție incluse, comparabile cu un operator, o funcție sau o declarație;
- **SubVIs (subVI-uri)** – VI-uri utilizate în diagrama bloc a unui VI, similare cu subrutinele;
- **Structures (structuri)** – elemente de control al transferului de date în diagrama bloc și stabilesc ordinea și repetarea execuției programului VI-ului;
- **Formula Nodes (noduri cu formule)** – structuri redimensionabile care servesc pentru introducerea de formule și ecuații în diagrama bloc;
- **Property nodes (noduri cu proprietăți)** - noduri care permit să se prescrie sau să se găsească proprietățile unei categorii de obiecte;
- **Invoke Nodes (noduri de apelare)** – noduri care execută proceduri specifice unei categorii;
- **Code Interface Nodes (noduri –coduri de interfațare)** – noduri care permit apelarea de coduri ale unor limbaje de programare textuale.

În cele ce urmează se vor prezenta câteva exemple de noduri din categoriile funcții și structuri.

Noduri de funcții 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**), rădăcină 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.

Noduri de funcții pentru valori booleene

Funcțiile pentru valori booleene sunt grupate în subpaleta **Boolean** a paletelor de funcții. Se remarcă funcțiile 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)** și **IMPLICA (Implies)**. Subpaleta conține, de asemenea, în partea inferioară, cele două constante logice: **ADEVARAT (True)** și **FALS (False)**.

În fig.1.8 și fig.1.9 referitoare la paleta **Functions** sunt redată subpaletelor **Numeric Constants** și **Boolean** în care se pot vedea formele în care apar în diagrama bloc nodurile de funcții pentru valori numerice, respectiv pentru valori booleene.

Noduri din categoria structuri

Ca și celelalte tipuri de noduri, structurile au terminale care le permit să se conecteze cu alte noduri, programul rulând automat, adică din momentul în care apar datele la intrare sunt furnizate rezultatele la terminalele de ieșire. Fiecare structură are o frontieră redimensionabilă, care include o porțiune din diagrama bloc, denumită subdiagramă, în care rularea programului se execută după regulile impuse de structură. Terminalele care transferă datele înspre și dinspre subdiagramă se numesc **tunele** și sunt puncte de conectare situate pe frontieră. În subpaleta **Structures** a paletelor **Functions** se găsesc 6 tipuri de structuri :

Structura **For Loop** - buclă repetitivă cu număr fix de iterații;

Structura **While Loop** - buclă repetitivă cu condiție de terminare;

Structura **Case** – structură de caz ce poate conține mai multe subdiagrame dintre care una singură se execută în funcție de mărimea aplicată la intrare;

Structura **Sequence** – structură secvențială care conține mai multe subdiagrame ce se execută în ordine succesivă;

Structura **Formula Node** – structură care efectuează operații matematice pentru intrări numerice;

Structura **Event** – structură care poate conține una sau mai multe subdiagrame a căror execuție depinde de modul în care utilizatorul intervine asupra **VI**-ului

Meniurile obiect în diagrama bloc

Atunci când se deschide **meniul obiect** al unui terminal din diagramă, acesta conține aproape aceleași opțiuni ca și meniul elementului corespunzător de pe panou, cu următoarele deosebiri:

- opțiunea **Find Terminal** a elementului de pe panou este înlocuită cu opțiunea **Find Control** sau **Find Indicator**. Apare opțiunea suplimentară **Hide Control** (sau **Hide Indicator**). Efectul acesteia este **ascunderea** elementului corespunzător de pe panou. Pe lângă opțiunea **Change to Indicator** (sau **Change to Control**) este adăugată și opțiunea **Change to Constant**, efectul acesteia fiind de înlocuire a elementului de control sau indicator cu o **constantă** (constantele pot exista doar în diagramă, astfel încât elementul de control sau indicator dispăre de pe panou);
- în submeniul **Create** apar opțiunile suplimentare **Constant**, **Control** și **Indicator**, efectul acestora fiind de adăugare (în diagramă sau în panou, după caz) a unei constante sau a unui element de control sau indicator de același tip cu elementul al cărui meniu propriu a fost deschis.

Realizarea conexiunilor în diagrama bloc

Realizarea unei conexiuni între două componente ale diagramei se efectuează cu ajutorul unelei de conectare selectată din paleta **Tools**. Ordinea în care se selectează cele două componente - sursa și destinația - nu este importantă. Legătura poate fi realizată în orice sens, deoarece programul identifică automat sursa și destinația pentru a stabili sensul de circulație a datelor. În fig.1.11 este ilustrată procedura de realizare a conexiunilor, care va fi explicată pas cu pas în cele ce urmează:

- se deplasează, cu ajutorul mouse-ului, unealta de conectare deasupra primei componente care va începe să pâlpâie și se execută click (poz.1). Această acțiune este echivalentă cu prinderea unui capăt al conexiunii de obiectul respectiv. Din acest moment, la deplasarea mouse-ului pe spațiul diagramei va apărea imaginea conexiunii cu un capăt fixat la prima componentă și celălalt capăt legat la unealta de conectare (poz.2);

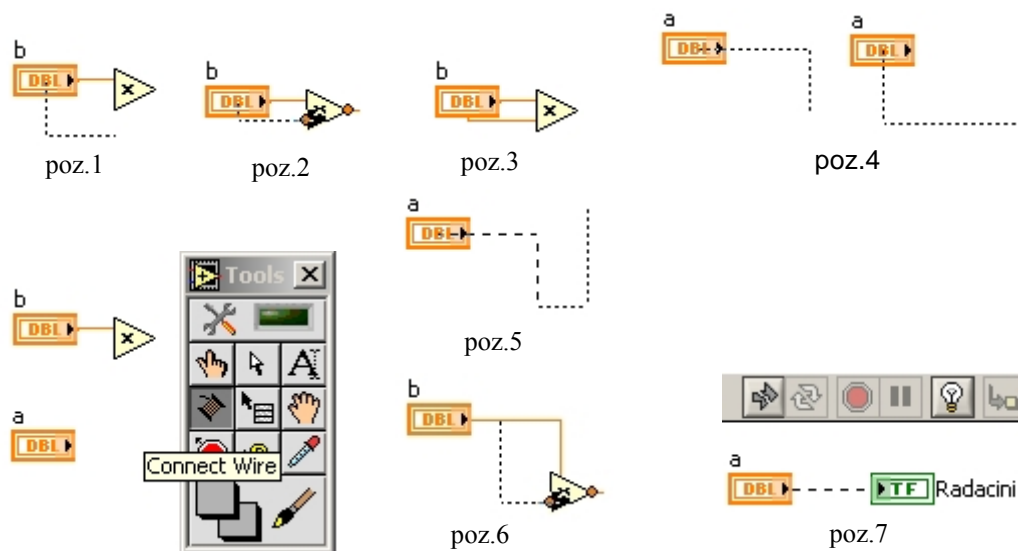


Fig.1.11. Realizarea conexiunilor în diagrama bloc

- deplasând în continuare unealta de conectare deasupra celei de a doua componente (care începe să pâlpâie) se execută click și astfel cel de al doilea capăt al conexiunii se fixează de această componentă (poz.3);
- dacă, la un moment dat, conexiunea capătă un sens neconvenabil acesta poate fi schimbat apăsând tasta spațiu (poz.4);
- în cazul în care se dorește ca traseul unei conexiuni să fie format din segmente dispuse așa încât să facă unghiuri drepte, un click în punctul de schimbare de direcție este echivalent cu fixarea conexiunii în acel punct (poz.5);
- dacă o componentă a diagramei are rol de sursă pentru un traseu ramificat, atunci se pot realiza conexiuni care să aibă ca punct de pornire capătul unui segment al traseului (poz.6);
- atunci când o conexiune este executată corect, ea este afișată în culoarea corespunzătoare tipului de date care circulă în zona respectivă. Dacă legătura este incorectă ea va fi afișată în culoarea neagră întreruptă (poz.7).

În cazul structurilor se poate conecta direct un terminal din exterior cu unul din interior printr-un **tunel**. Tunelul este deplasabil și deci se formează acolo unde conexiunea trebuie să treacă peste frontiera structurii.

Atunci când două conexiuni se intersectează, primul fir apare întrerupt ca și cum s-ar afla sub cel de al doilea. Dacă obiectele conectate sunt deplasate, conexiunile se deplasează și ele în mod automat.

Selectarea unei conexiuni în vederea ștergerii sau modificării se face în modul următor: executând click o singură dată cu unealta de poziționare se selectează segmentul (orizontal sau vertical) deasupra căruia se află unealta respectivă, dacă se face click de două ori se selectează ramura din care face parte segmentul (între două puncte de joncțiune), iar dacă se face click de trei ori se selectează toată conexiunea (între toate terminalele conectate).

Ștergerea unui segment se face prin selectarea lui, după care se apasă tasta **Delete** sau **Backspace**, ori se selectează opțiunea **Edit >> Clear** (sau **Cut**). La mutarea unui obiect conectat se poate să apară un segment în plus care să nu aibă o poziție convenabilă. Pentru a-l aduce în poziția dorită se selectează cu unealta de poziționare, după care, ținând tasta mouse-ului apăsată, se deplasează în noua locație.

În cazul unei conexiuni greșite, pe lângă faptul că aceasta apare ca o linie întreruptă, butonul **Run** se schimbă în **BrokenRun**. Dacă nu se știe locul unde se află conexiunea greșită se selectează opțiunea **Edit >> Remove Bad Wires** și în acest caz toate conexiunile greșite vor fi șterse. Executând click cu tasta dreaptă pe conexiunea greșită, se afișează meniul propriu din care alegând **List Errors** va apare o casetă în care se pot citi erorile posibile și cauzele lor.

1.6. Salvarea unei aplicații; VI nou creat sau modificat

Atât timp cât încă nu a fost salvat într-un fișier, un program are numele **Untitled** urmat de un număr de ordine. Atunci când se salvează este necesar să i se atribuie un nume.

Pentru a specifica folder-ul în care programul să fie salvat, precum și numele fișierului, se alege din meniul **File** opțiunea **Save As...**. În urma selectării comenzii **Save As...**, programul deschide o fereastră de dialog în care utilizatorul poate selecta folder-ul în care să se efectueze salvarea și numele fișierului.

Programele realizate în mediul de programare grafică **LabVIEW** se salvează în fișiere cu extensia **.vi**.

Din momentul în care a fost salvat prima dată, programului i se afișează numele atât în bara superioară a panoului cât și în cea a diagramei.

Dacă, după salvare, programului i se aduc în continuare modificări, lângă numele acestuia apare un asterisc (*), semnalizând astfel faptul că versiunea ce a fost salvată nu este cea mai recentă.

Pentru a salva un program care are deja o locație și un nume stabilite, este suficient să se apese combinația de taste **Ctrl** și **S**, sau să se selecteze comanda **Save** din meniul **File**.

Comanda **Save As...** se poate folosi dacă se dorește salvarea unei noi versiuni a fișierului într-o nouă locație, sau cu un nou nume.

În cazurile în care trebuie salvate mai multe **VI**-uri și se dorește o grupare a lor, **LabVIEW** permite crearea unei "biblioteci" de **VI**-uri. Crearea unei astfel de biblioteci se obține selectând **Save As** din meniul **File**, ceea ce are ca efect deschiderea unei ferestre **Name the VI (Untitled 1)** unde, apăsând pe butonul **New VI Library**, va apare o casetă în care se introduce numele bibliotecii urmat de extensia "**.lib**", după care se afișează o nouă casetă de dialog în care se înscrie numele **VI**-ului din biblioteca nou creată.

1.7. Căutarea și vizualizarea unor exemple de **VI**-uri create de **LabVIEW**

LabVIEW-ul conține sute de exemple de **VI**-uri create pentru a demonstra facilitățile acestui mediu de programare grafică pentru diverse domenii ale instrumentației, de la achiziția datelor, analiza și procesarea semnalelor, măsurarea unei game largi de mărimi fizice, până la controlul și automatizarea proceselor. Studiarea acestor exemple este deosebit de utilă pentru demonstrarea bazelor programării grafice, pentru aprofundarea posibilităților de construire de instrumente virtuale, de observare a rulării acestora și chiar pentru importarea directă sau cu modificări a unora dintre ele ca **subVI**-uri.

Există două modalități de a accede la exemplele oferite de **LabVIEW** și anume:

- a) În caseta de dialog inițială a **LabVIEW** (fig.1.1) se apasă butonul **Find Examples**, ceea ce conduce la deschiderea casetei de dialog denumită tot **Find Examples**, în care este afișată lista de domenii din care se pot extrage exemple și două opțiuni **Browse** și **Search**;
- b) În cazul în care a fost deschisă o aplicație din meniul **Help**, se selectează opțiunea **Find Examples** după care se procedează la fel ca în cazul precedent.

În continuare, folosind opțiunea **Browse**, se selectează una dintre categoriile de exemple afișate. Presupunând că s-a selectat **Industry Applications >> Analysis** se execută apoi dublu click pe **Temperature System Demo** și se deschide un exemplu de **VI** care permite să se simuleze și să se afișeze variații de temperatură, să se avertizeze la depășirea unor limite impuse, să se calculeze valoarea medie și dispersia pe un anumit interval și să se prezinte histograma valorilor de temperatură.

Dacă se utilizează opțiunea **Search** atunci căutarea se va face după cuvinte cheie. Pentru găsirea exemplului precedent se introduce “temperatures” și, executand click în caseta de dialog, apar mai multe opțiuni de **VI**-uri, printre care și cel care interesează - **Temperature System Demo.vi**.

În afara celor descrise mai sus, **LabVIEW** mai conține și unele modele predefinite de **VI**-uri (**templates**), care pot fi utilizate pentru crearea de **VI**-uri uzuale. Accesul la aceste modele predefinite se face din meniul **File >> New** selectând **Start from Template**.

1.8. Documentațiile LabVIEW

LabVIEW are atașată o amplă și variată documentație concepută astfel încât să fie accesibilă începătorilor, dar folositoare și utilizatorilor experimentați. Accesul la documentația **LabVIEW** se poate face din meniul **Help** pe două căi:

a) Selectând opțiunea **Search the LabVIEW Bookshelf** care deschide o casetă de dialog în care sunt afișate documentele introductive, manualele de utilizare și notele de aplicații;

b) Selectând opțiunea **VI, Function & How-To Help >> Related Documentation** se deschide o fereastră conținând lista cu documentațiile **LabVIEW** însoțită de scurte comentarii privind cuprinsul și cui se adresează.

Indiferent de opțiunea adoptată, principalele documentații **LabVIEW** - care trebuie accesate și consultate în mod progresiv - sunt următoarele:

- **Getting Started with LabVIEW** - este un manual introductiv care permite utilizatorului să ia contact cu mediul de programare grafică și cu principalele caracteristici necesare pentru crearea de aplicații pentru achiziții de date, instrumentație și control;

- **LabVIEW Tutorial** - conține principiile esențiale ale **LabVIEW** și, folosind prezentarea unor “activități” de bază, familiarizează cu etapele de creare a unui **VI**;

- **LabVIEW User Manual** - este un manual de nivel mai ridicat, care tratează conceptele și tehnicile programării în **LabVIEW**, elementele și funcțiile care se folosesc pentru crearea de **VI**-uri pentru testare, măsurare, achiziție și stocare de date, prelucrare și analiză;

- **LabVIEW Measurements Manual** - este un manual destinat însușirii de cunoștințe avansate cu privire la realizarea unor variate tipuri de aplicații de instrumentație și de analiză pentru măsurări industriale și de laborator;

- **LabVIEW Development Guidelines** - descrie și analizează modul în care trebuie conceput un **VI**, tehnici de proiectare și de documentare. Presupune însușirea cunoștințelor din manualele anterioare.

În **Related documentation** se pot găsi și alte titluri și link-uri de acces la diverse documentații despre **LabVIEW**.

1.9. Chestiuni de studiat

1. Se va deschide o aplicație nouă și se vor vizualiza cele două ferestre, panoul frontal și diagrama bloc, recunoscându-se elementele componente ale acestora.

2. Se vor deschide succesiv meniurile și se vor observa opțiunile acestora; se vor studia semnificațiile butoanelor și operațiile care pot fi executate prin acționarea lor.

3. Selectând meniul și opțiunile adecvate, se va afișa și se va vizualiza documentația **LabVIEW Tutorial**.

4. Se vor afișa paletele **Controls** și **Tools** și se vor studia semnificațiile subpaletelor pe care le conțin. Folosind aceste palete se va dispune în panoul frontal un obiect de tipul **Vertical Pointer Slide** și altul de tipul **Meter**. Se va preciza căror categorii aparțin cele două obiecte. Se va observa ce se întâmplă în diagrama bloc în urma operațiilor anterioare și se va explica semnificația obiectelor apărute. Idem pentru un obiect de tipul **Vertical Toggle Switch** și unul de tipul **Round Led**. Se vor deplasa obiectele în spațiul de lucru.

5. Folosind unealta de selectare din paleta **Tools** se vor deplasa în spațiul de lucru obiectele de la punctul 4 și se vor modifica dimensiunile acestora. Se va căuta modalitatea de ștergere a elementelor din panou. Pentru fiecare din aceste operații, se va observa și comenta ce se întâmplă în diagrama bloc.

6. Se va afișa paleta **Functions**, se vor deschide subpaletele componente și se vor observa semnificațiile lor. Se vor dispune în diagrama bloc funcția sumă (**Add**) și constanta numerică (**Numeric Constant**) cu valoarea numerică **10**. Idem pentru funcția **AND** și constanta logică **True Constant**. Se vor modifica pozițiile elementelor din diagramă folosind mai multe proceduri.

7. Folosind unealta de conectare, se va realiza o conexiune între terminalul elementului **Slide** și cel al elementului **Meter**. Se vor explica semnificațiile aspectelor, culorilor terminalelor și conexiunii și pe cele ale simbolurilor din casetele terminalelor. Se va șterge apoi conexiunea.

8. Disponând în panoul frontal controlul **Vertical Toggle Switch** și indicatorul **Round Led** de la punctul 4, se va executa conexiunea dintre acestea în diagrama bloc. Deschizând meniul propriu al elementului de control, se va accesa opțiunea **Mechanical Action** și se va efectua câte o rulare continuă pentru fiecare dintre variantele afișate. Se va observa și se va comenta comportarea obiectelor respective.

9. În panoul frontal se va plasa un control numeric și un indicator numeric, iar în diagrama bloc funcția sumă (**Add**) și o constantă numerică (**Numeric Constant**). Se vor executa conexiunile între obiectele care apar în diagrama bloc. Apoi, dând diverse valori controlului numeric și rulând programul (**Run** și **Run Continuously**), se va observa care sunt valorile afișate de indicatorul numeric.

10. Se va găsi și se va deschide exemplul **Tank Simulation.vi**. Se va rula programul de execuție și se vor observa mărimile afișate și fluxul de date din diagrama bloc în regimuri de funcționare continuă și intermitentă. Se vor acționa elementele de execuție și se vor observa efectele acestora.

11. Se va construi o aplicație simplă pentru generarea și afișarea unor numere aleatoare; pașii necesari în construirea aplicației sunt descriși în **Getting Started with LabVIEW**, document pdf aflat în setul de documentații aferent programului.

1.10. Modul de lucru și prezentarea rezultatelor

Punctele 1 ... 9 de la chestiuni de studiat se vor efectua în conformitate cu descrierile detaliate ale limbajului **LabVIEW** prezentate în paragrafele 1.2 ... 1.5.

Punctul 10 al chestiunilor de studiat se va face în conformitate cu precizările de la paragraful 1.7.

Punctul 11 al chestiunilor de studiat se va efectua – pas cu pas – conform precizărilor din documentația **Getting Started with LabVIEW**. După verificarea funcționării corecte, se va salva aplicația sub denumirea **Generator de numere aleatoare_[nume/prenume student]_[grupa].vi** în directorul aferent grupei/seriei respective.

În acest scop, fiecare student, urmărind punctele de la chestiunile de studiat, va înscrie răspunsurile, rezultatele, comentariile și concluziile proprii, iar acolo unde este cazul, se vor salva unele dintre imaginile obținute ca rezultat al activităților desfășurate, folosind nume sugestive pentru fiecare, urmate de extensia obligatorie (nume, grupă), așa cum s-a precizat anterior.