

Aplicația II.2

CREAREA DE INSTRUMENTE VIRTUALE (VI)

Denumirea de instrumente virtuale, prescurtat **VI**, provine din faptul că în primele versiuni programele **LabVIEW** realizau, prin intermediul calculatorului, funcții ale unor instrumente și aparate electronice, adesea imitând imaginile și modurile de operare ale acestora.

Mediul de dezvoltare **LabVIEW** a evoluat rapid și în prezent noțiunea de **VI** s-a extins, astfel că acoperă o gamă largă de aplicații inginerești, de la efectuarea de calcule matematice, achiziția și prelucrarea datelor, generarea de diverse tipuri de semnale, până la simularea sau controlul unor procese industriale complexe.

În cadrul acestei lucrări se vor prezenta aplicații simple, ce au ca scop familiarizarea cu elementele de bază ale **LabVIEW**-ului și care permit crearea de **VI**-uri.

2.1. VI pentru calculul sumei și produsului

Primul exemplu de realizare a unui **VI** simplu este o aplicație prin care se efectuează suma și produsul a două numere, a căror valoare poate fi modificată de utilizator.

În conformitate cu cele arătate în **Aplicația II.1**, pentru deschiderea unei noi aplicații, în caseta de dialog se execută click pe butonul **New VI** și se afișează cele două ferestre: panoul frontal și diagrama bloc. Operațiile care urmează constau din:

- disponerea de elemente în panoul frontal;
- disponerea de elemente în diagrama bloc;
- realizarea conexiunilor între elementele dispuse în diagrama bloc.

Panoul Frontal

Elementele din panoul frontal sunt de două tipuri: de control și de afișare (indicatoare). În cadrul diagramei bloc regăsim “motorul” aplicației, adică modulele funcțiilor și interconexiunile între acestea, precum și elemente corespunzătoare celor din panoul frontal.

În cazul acestui exemplu, în panoul frontal trebuie să se dispună două controale, care să reprezinte cele două numere și unul sau mai multe indicatoare. Pentru controale se selectează din paleta **Controls** butonul **Numeric** și, din meniul nou apărut, **Vertical Pointer Slide** (cursor vertical). Forma cursorului se modifică, iar când este poziționat pe panou apare conturul elementului selectat. Efectuând click în locul final, se va afișa controlul cu ajutorul căruia se va putea alege un număr într-un anumit interval. După ce s-a plasat controlul, se selectează din paleta **Tools** săgeata

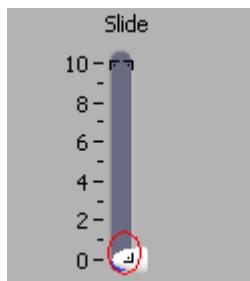


Fig. 2.1. Marginea de redimensionare a slide-ului

(care permite re poziționare, redimensionare sau selecție). Dacă se poziționează pointer-ul mouse-ului deasupra controlului introdus, se observă că în anumite puncte forma acestuia se schimbă, permițând modificarea mărimii slide-ului sau a unor componente ale sale. Slide-ul nou introdus trebuie mărit pe verticală. Pentru aceasta se folosesc marginile de redimensionare așa cum se arată în fig.2.1. Atenție însă, există posibilitatea de mărire a butonului de variație, ce va conduce la o operație greșită. După redimensionare, următorul pas constă în stabilirea titlului slide-ului.

Pentru aceasta, din paleta **Tools**, se selectează butonul **Edit Text** și se efectuează click în zona unde este afișat textul **Slide**. Textul devine accesibil și se poate edita. Se denumește **Numarul 1**. Trebuie menționat că, pentru a activa schimbarea efectuată, există trei opțiuni:

- se acționează butonul **Ok**, care apare înaintea lui **Run**, în momentul când textul devine editabil, așa cum se poate vedea în fig.2.2;
- se execută click cu mouse-ul în altă zonă a panoului;
- se apasă **Ctrl+Enter** (dacă se apasă tasta **Enter** când editați un text se va insera o linie nouă).

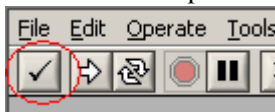


Fig.2.2. Butonul **Ok** pentru acceptarea schimbării

Următorul pas este rescalarea controlului. În acest moment, se pot selecta numere pe o scară de la 0 la 10. Cu butonul **Edit Text** selectat din paleta **Tools**, se efectuează click în zona numărului 10, și acesta va deveni editabil. În loc de 10 se scrie 100 și se activează schimbarea folosind

una dintre metodele descrise mai sus.

LabVIEW permite mai multe tipuri de date. În continuare se va descrie

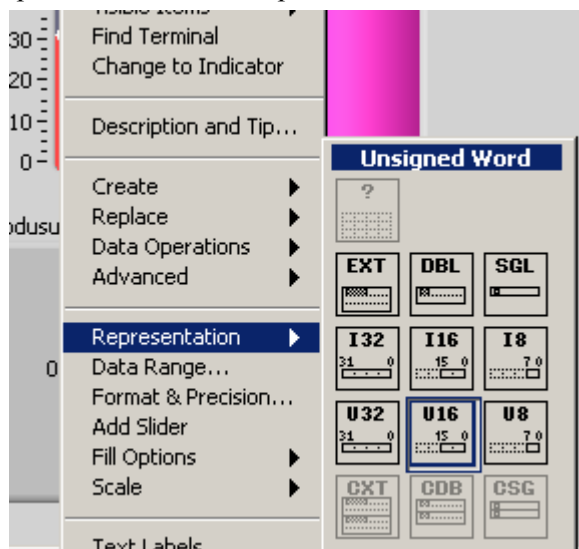


Fig.2.3. Selecția tipului de date **Unsigned Byte**

procedura de operare numai cu numere întregi. În cazul indicatoarelor numerice, tipul de date selectat implicit este **Double**. Pentru a schimba această opțiune, se selectează din nou săgeata din panoul **Tools** și se execută click dreapta pe slide-ul introdus. Se deschide un meniu tip pop-up de unde se alege opțiunea **Representation**, unde se găsesc tipurile de date, similare cu cele din programare. Se selectează **Unsigned Word**, așa cum se poate vedea în fig.2.3. Deși numerele pot varia între 0 și 100, deci ar fi fost de ajuns **Unsigned Byte**, totuși în cazul produsului, numărul va depăși valoarea 255 – maximul **Unsigned Byte**. **LabVIEW** nu face cast-ul automat și, ca atare, dacă intrările sunt **Unsigned Byte** automat și ieșirea va avea același tip. Din acest motiv se vor defini numerele ca fiind **Unsigned Word**.

În cazul unui control sau indicator care are și parte fracționară este permisă selectarea preciziei. În acest sens, în meniul pop-up, deschis cu click dreapta, există opțiunea **Format & Precision**. Fereastra care apare în urma alegerii acestei opțiuni are selectat câmpul **Digits of Precision** implicit; **LabVIEW** alocă două cifre de precizie pentru un terminal (control sau indicator).

LabVIEW permite stabilirea unor valori implicite pentru fiecare terminal care se adaugă într-o schemă. În general nu deranjează dacă valorile controalelor sau indicatoarelor sunt 0 în momentul pornirii execuției. În practică însă, se vor întâlni cazuri în care aplicația îndeplinește automat o condiție de oprire dacă valoarea unui control este 0. Pentru exemplul considerat se pot stabili numere inițiale nenule, ceea ce va conduce la valori diferite, de asemenea nenule, pentru sumă și produs.

Stabilirea valorii implicite a unui control sau a unui indicator se poate realiza selectând din paleta **Tools** butonul **Operate Value**. Cursorul se transformă într-o mână, cu care se acționează asupra indicatorului sau controlului. Pentru slide-ul **Numarul 1**

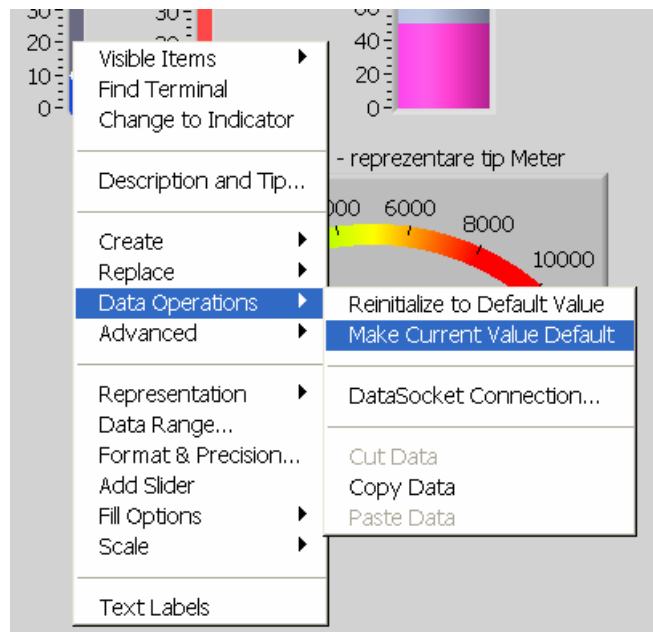


Fig.2.4. Stabilirea valorii implicite a unui control

se crește valoarea inițială de la 0 la 10. După ce s-a stabilit valoarea dorită, cu click dreapta în zona slide-ului, se alege **Data Operations >> Make Current Value Default** din meniul care se deschide, așa cum se poate observa în fig.2.4.

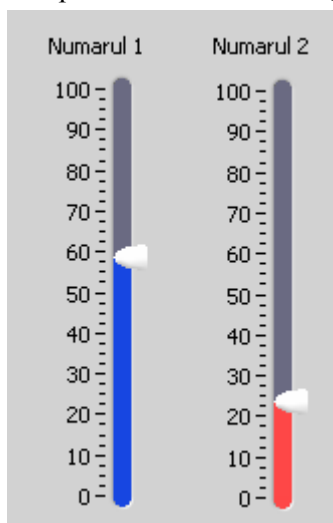


Fig.2.5. Cele două controale pentru numere

Trebuie menționat că meniul de tip pop-up, ce apare în urma unui click dreapta în zona unui terminal, se deschide indiferent de tipul de cursor cu care se lucrează (săgeată, dacă a fost selectat **Size Tool**, mână pentru **Operate Value** etc).

În acest moment s-a terminat de configurat controlul primului număr. Pentru a nu repeta toate operațiile de mai sus, cu săgeata din paleta **Tools** activată, se selectează slide-ul, și apoi **Copy** sau **Ctrl+C**. Se mută săgeata într-o zonă unde urmează să se plaseze al doilea control, se execută click pentru a deselecta slide-ul anterior și apoi **Paste** sau **Ctrl+V**. Va apare cel de-al doilea control, ce are deja numele **Numarul 2**. Pentru a face o diferență vizuală între cele două controale, "indicatoarele de umplere" se pot colora diferit, așa cum se vede în fig.2.5. Pentru aceasta, în paleta **Tools**, există subpaleta **Set Colors**, care are două

zone: **Text** și **Fundal**. Dacă se apasă cursorul mouse-ului pe butonul "din față" se poate selecta culoarea pentru **Text** sau, în general, culoarea elementelor aplicate din cadrul unui control (de ex. indicatorul de umplere). După ce s-a ales nuanța dorită, se observă schimbarea cursorului într-o pensulă, care se deplasează deasupra zonei a cărei culoare urmează a fi schimbată și se efectuează click.

După ce s-au inserat în **VI** controalele pentru numere, urmează indicatoarele. Mai întâi trebuie să se știe care este valoarea numerelor și pentru aceasta se alege **Digital Indicator** din subpaleta **Numeric** a paletii de **Controls** și se introduce în panoul frontal. Apoi, cu săgeata din paleta **Tools** activă, se selectează numai zona unde se găsește numele indicatorului și se mută în stânga acestuia. Se redefiniște câmpul **Numeric** în **Numarul 1** și se setează precizia la 0 digiți. În continuare se selectează întreg controlul, cu eticheta redenumită și câmpul de afișare și se efectuează **Copy**. Acest indicator se multiplică de trei ori, redenumind de fiecare dată eticheta, pentru a fi afișată valoarea numărului 2, precum și cele ale sumei și produsului.

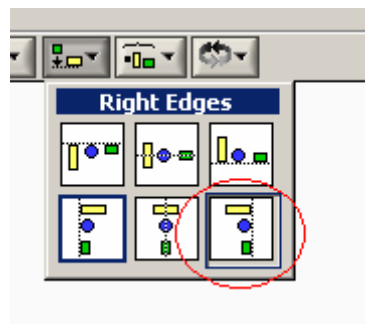


Fig.2.6. Alinierea obiectelor

Se selectează apoi grupul de 4 indicatoare astfel create pentru a le alinia. Din bara de butoane, se acționează **Align objects**, și apoi **Right Edges**, așa cum se poate observa în fig.2.6.

Teoretic, în acest moment, ar trebui să fie aliniate și etichetele celor patru indicatoare. Însă în practică, datorită unui click sau a altui eveniment, s-ar

putea să nu fie așa. De aceea se selectează doar etichetele și se mai face odată aliniere la dreapta.

După ce s-au stabilit indicatoarele numerice, urmează să se aleagă și două indicatoare vizuale (analogice), de exemplu, un **Tank Indicator** pentru sumă și un **Meter** pentru produs. Se redefinesc etichetele celor două indicatoare și scalele acestora. Evident, suma va trebui să permită o variație până la 200, în timp ce produsul până la 10000. Panoul frontal trebuie să arate ca în fig.2.7.

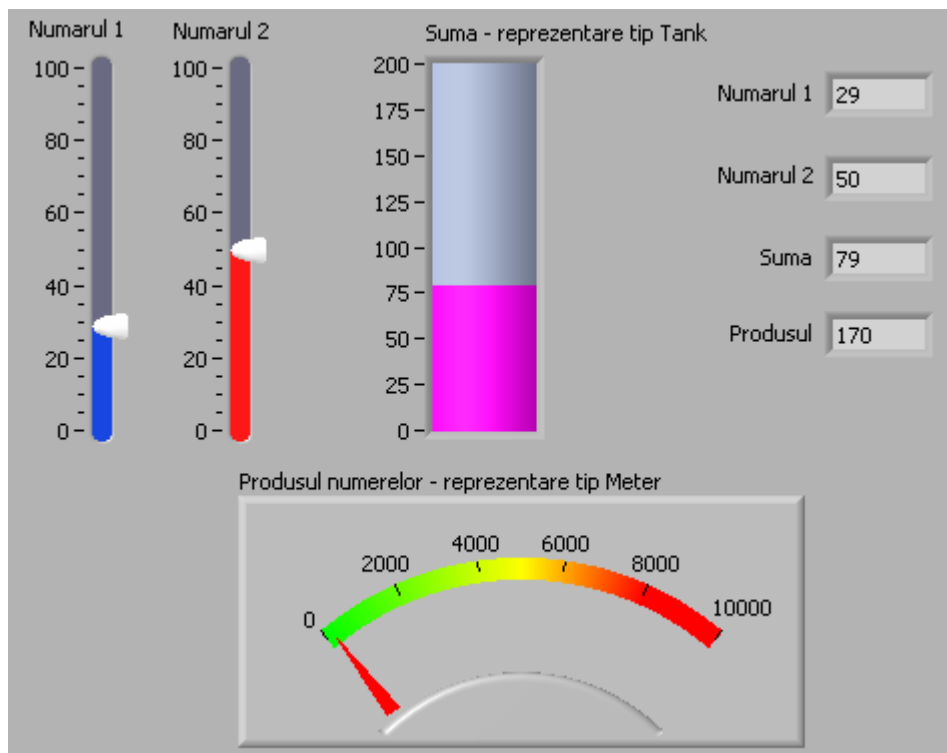


Fig.2.7. Panoul frontal al aplicației

Diagrama bloc

După ce a fost realizat panoul, se procedează la conectarea elementor grafice și adăugarea funcțiilor pentru ca aplicația să realizeze suma și produsul celor două numere. Pentru aceasta, din meniul **Window**, acționând **Show Diagram** sau apăsând **Ctrl + E**, se va deschide diagrama bloc a aplicației.

După cum se observă, aici elementele care corespund controalelor și indicatoarelor sunt “aruncate” la întâmplare. Evident, alinierea s-a aplicat doar în cazul panoului frontal. Deci primul pas este reprezentat de aranjarea convenabilă a elementelor, pentru a putea realiza apoi conexiunile între ele. O aranjare utilă ar fi cea în care controalele pentru numerele 1 și 2 sunt în partea stângă a diagramei, indicatoarele corespunzătoare în dreptul lor, indicatoarele grafice urmând a se poziționa în dreapta, așa cum se poate observa în fig.2.8.

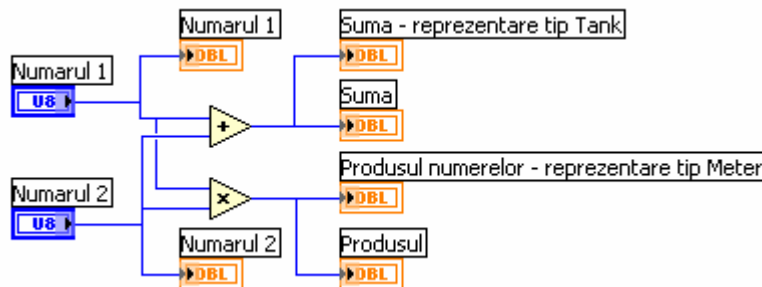


Fig.2.8. Diagrama bloc a aplicației

Aranjarea se face similar ca în cazul panoului frontal, folosind săgeata pentru poziționare, din paleta **Tools**, rezultând dispoziția din fig.2.8.

În acest moment, la o prima vedere pot apărea două confuzii:

- există două elemente **Numarul 1** și **Numarul 2** pentru care s-au folosit același nume și în cazul controlului și în cazul indicatorului. Diferența se face în funcție de poziția săgeții: la control, săgeata este în dreapta în partea de **Out**, deoarece controlul generează un număr, în timp ce la indicator săgeata este în stânga, în partea de **In**, deoarece, evident, acesta preia un număr și îl afișează;
- controalele sunt **Unsigned Byte - U8**, în timp ce indicatoarele sunt **Double**. Nu deranjează acest aspect, întrucât setarea s-a făcut cu 0 digiți la precizia indicatorului. Însă, în practică, atât în **LabVIEW**, ca și în alte limbaje de programare, este de preferat să se lucreze cu aceleași tipuri de date, deoarece în caz contrar au loc cast-uri implicite, care pot fi generatoare de erori. Aceste erori sunt foarte greu de depistat, în majoritatea cazurilor.

Un alt aspect important este că un element ce corespunde unui control sau unui indicator din panoul frontal nu poate fi șters din diagramă. Dacă nu mai este util, atunci trebuie revenit în panoul frontal și se șterge cu **Ctrl+E**, acesta dispărând automat și din diagramă.

După ce elementele corespondente celor din panoul frontal au fost aranjate, se adaugă funcțiile necesare pentru sumă și produs. Acestea se găsesc în subpaleta **Numeric** din paleta **Functions** și includerea lor în cadrul diagramei bloc se face similar cu modul în care au fost inserate controalele și indicatoarele în panoul frontal.

Un aspect foarte important în cadrul realizării unei aplicații sau simulări **LabVIEW** este reprezentat de conectarea elementelor pentru ca informația să fie prelucrată cu succes. Pentru acest pas, se selectează butonul **Connect Wire** din paleta **Tools**. Mai întâi se execută o conexiune între controalele celor două numere și indicatoarele aferente acestora făcând click pe controlul de unde se stabilește începutul conexiunii și click de sfârșit la indicatorul pentru finalizarea acesteia. După aceea se conectează slide-urile de variație a numerelor cu modulele care realizează suma și produsul. Pentru aceasta se execută click pe conexiunea dintre control și număr, pentru a marca începutul, și apoi pe indicatorul final. La fel se conectează **Tank**-ul și

indicatorul numeric pentru sumă la ieșirea modulului sumă, iar **Meter**-ul și indicatorul de produs la ieșirea modulului produs. Schema de conexiuni trebuie să arate ca aceea din fig.2.8.

Rularea aplicației

Dacă au fost respectați toți pașii, aplicația este în acest moment funcțională și poate fi rulată. Există mai multe opțiuni de rulare a unei aplicații **LabVIEW**:

- rulare o singură dată;
- rulare continuă;
- rulare cu observarea valorilor transmise între diferite blocuri în diagramă.

Dacă se acționează **Ctrl+R**, sau butonul **Run** ori **Operate/Run** din meniu, se va parcurge schema realizată o singură dată. Practic, nu se va vedea nimic, pentru că din cele două controale vor pleca valorile 0 și 0, deci suma și produsul vor fi tot 0. O altă modalitate de a rula aplicațiile, potrivită pentru acest exemplu, este rulare continuă, care se obține prin apăsarea butonului **Run Continuously**, localizat în bara de comenzi lângă butonul **Run**. Acesta are ca efect rulare continuă a aplicației de un număr nelimitat de ori, până se apasă butonul de **Abort Execution**.

O aplicație **LabVIEW** poate fi gândită în mai multe moduri. Astfel, se poate introduce întreaga schemă într-o buclă tip **While**, care se va termina cu acționarea unui buton de **Stop**, așa cum se va vedea în exemplele ulterioare. Pentru simplitate,

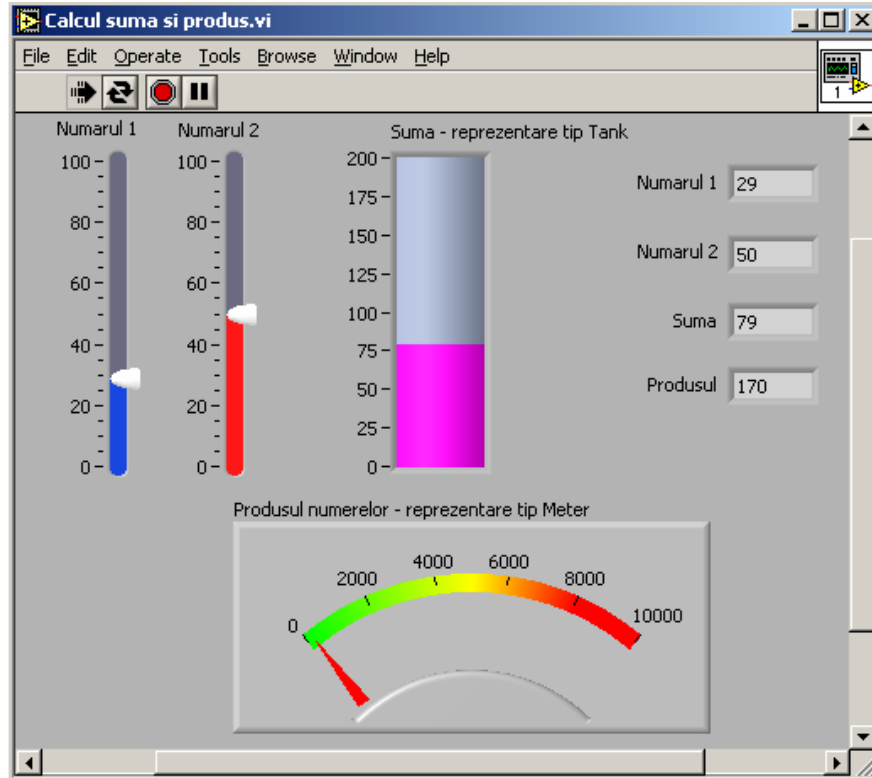


Fig. 2.9. Rularea aplicației

acest prim exemplu se va rula folosind comanda **Run Continuously** și se va termina folosind butonul **Abort Execution** (fig.2.9).

Depanarea (debug-ul) aplicației

Din multe puncte de vedere, **LabVIEW** se aseamănă cu un mediu de dezvoltare de tip WYSIWYG (What You See Is What You Get). De aceea, nu putea să lipsească din **LabVIEW** posibilitatea de depanare a unei scheme sau a unei diagrame. Debug-ul este orientat pe mai multe direcții:

- urmărirea valorilor în cadrul unei diagrame;
- urmărirea valorilor într-un punct;
- stabilirea unor break point-uri;
- detectarea de **VI**-uri sau module incomplete sau incomplet cablate.

Urmărirea valorilor în cadrul unei diagrame se poate face activând butonul **Highlight Execution** din diagramă, localizat în **Toolbar**, care are simbol un bec. Activarea acestui buton permite afișarea valorilor care sunt furnizate de către fiecare modul din cadrul diagramei. Folosirea acestui mod de debug are ca efect micșorarea vizibilă a vitezei de execuție a aplicației.

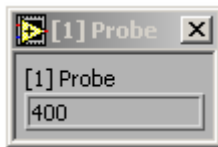


Fig.2.10. Fereastra **Probe**

Dacă se dorește verificarea valorilor furnizate de către un anumit modul, atunci se poate folosi opțiunea **Probe**. Pentru aceasta se acționează din paleta **Tools** butonul **Probe Data** ce va avea ca efect schimbarea cursorului într-un cerc. Se selectează modulul sau firul unde se dorește vizualizarea valorilor instantanee, și în acest moment se va deschide o fereastră asemănătoare cu

cea din fig.2.10.

Pentru fiecare valoare care se monitorizează în cadrul execuției se pot obține astfel de ferestre.

Breakpoint-urile reprezintă un instrument important în procesul de debug al unei aplicații. Sunt utile, în special, în cazul unor aplicații foarte complexe, în care se dorește oprirea rulării într-un anumit punct, pentru a nu sta să se execute și restul schemei. Setarea unui **breakpoint** se face cu ajutorul butonului **Set/Clear Breakpoint** din paleta **Tools**. În momentul selecției acestui buton, cursorul se transformă într-un semn asemănător **stop**-ului și, în cadrul diagramei, se poate selecta locul unde urmează să se oprească execuția. Trebuie menționat că setarea unui **breakpoint** nu duce la modificarea cursorului în alt obiect, de aceea trebuie avut grijă ca să nu se seteze, din greșală, mai multe **breakpoint**-uri.

O mare parte din erorile care apar în **LabVIEW** provin din legarea necorespunzătoare a terminalelor sau modulelor. Prin terminale se înțeleg elementele care au fost adăugate în scheme în panoul frontal, adică controalele și indicatoarele. Următoarele două situații se întâlnesc cel mai frecvent:

- module incomplet cablate. Un modul poate avea mai multe intrări și, de asemenea, mai multe ieșiri. Încă din stadiul de proiectare al modulului se stabilesc intrările și ieșirile care trebuie neaparat cablate. Pe lângă acestea, se mai pot întâlni intrări sau ieșiri care pot fi sau nu cablate în funcție de opțiunile și necesitățile utilizatorului. Un modul trebuie să aibă fie o intrare, fie o ieșire care să fie conectată în

schemă. Există module care au doar o singură intrare care este necesar să se cableză, în timp ce altele au doar o ieșire.

- cablarea între module care suportă date diferite. În general, când se încearcă să se cupleze module care schimbă date diferite, **LabVIEW**-ul atenționează desenând legătura dintre ele ca un fir întrerupt de culoare neagră. Trebuie avut în vedere că “întreruperea” unei legături între două module se datorează conectării unui al treilea, care nu procesează același tip de date ca primele două, așa cum se poate observa în fig.2.11.

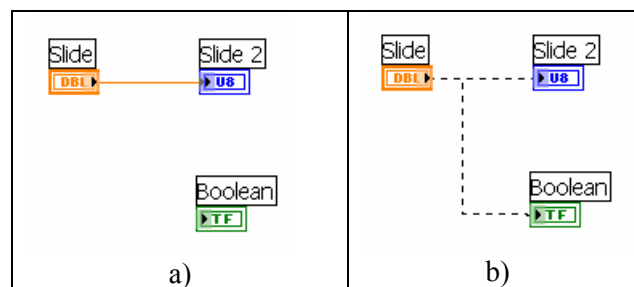


Fig. 2.11. a) două terminale ce pot procesa aceleași date conectate; b) conectarea unui terminal diferit duce la întreruperea firului și între primele două

Situația de mai sus se poate rezolva selectând legătura care duce către terminalul de tip boolean și apăsând tasta **Delete**, pentru a o șterge. Un shortcut util în **LabVIEW** este **Ctrl+B**, care are ca efect ștergerea tuturor legăturilor întrerupte din cadrul unei scheme.

2.2. Utilizarea funcțiilor Context Help și Timer în crearea VI-urilor

Help-ul în LabVIEW

După cum s-a arătat până acum, **LabVIEW** nu oferă explicit o fereastră tip **Properties**, de unde să se poată seta toate opțiunile, așa cum se întâmplă în majoritatea aplicațiilor Windows. În schimb, opțiunile sunt împărțite pe diferite categorii și sunt disponibile, în general, în meniul tip pop-up care apare când se execută click dreapta pe un terminal sau pe un modul.

O altă problemă este identificarea unui modul. După familiarizarea cu mediul **LabVIEW**, majoritatea elementelor de conectică se pot recunoaște după imaginea tip **Always on** caracteristică. Însă până atunci, neexistând o fereastră de tip **Properties**, la o primă privire va fi greu de recunoscut un anumit modul în panoul diagramei bloc. Pentru a întâmpina acest neajuns, dezvoltatorii **LabVIEW** au pus la dispoziția utilizatorilor **help**-ul de context, care este o fereastră **Top** și în care sunt actualizate instantaneu informațiile despre elementele peste care se poziționează mouse-ul, indiferent de cursorul acestuia.

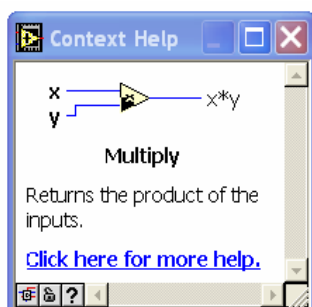


Fig.2.12. Fereastra **Context Help** – help de context

mod deosebit tipurile de date pe care trebuie să le primească un anumit modul, se pot afla informații minimale despre intrările și ieșirile acestuia atunci când se execută cablarea propriu-zisă a schemei.

Fereastra **Context Help** poate fi accesată fie din meniul **Help** al mediului **LabVIEW**, fie cu combinația de taste **Ctrl+H**. În această fereastră se pot vedea intrările și ieșirile unui anumit modul (fig.2.12).

Dacă se doresc informații detaliate, prin accesarea link-ului **Click here for more help** se obține o redirectionare către **help-ul** detaliat al **LabVIEW-ului**, unde, spre exemplu, modulul **Multiply** este prezentat ca în fig2.13.

Dacă imaginile diferitelor module din **LabVIEW** se recunosc ușor și nu interesează în

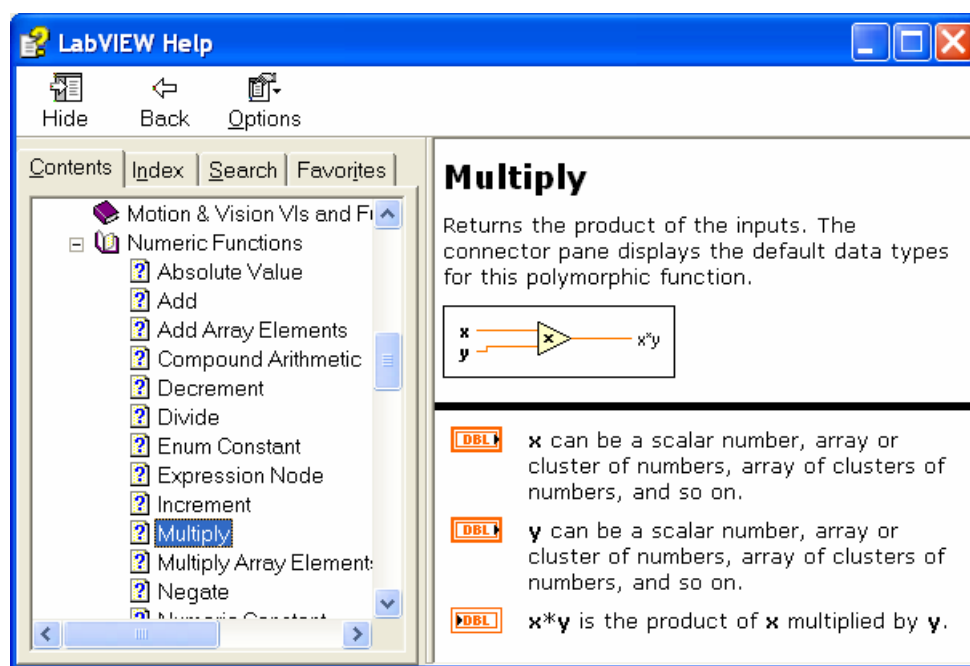


Fig.2.13. **Help-ul** detaliat

Ca o concluzie, dacă se dorește să se efectueze operații asupra unor module sau terminale (stabilirea scalei, tipul de date, valoare implicită etc) se efectuează click dreapta, iar dacă se doresc informații, se apelează fie la fereastra de **Context Help**, fie la **help-ul** **LabVIEW-ului**.

Temporizarea unei aplicații

În mediul industrial există cazuri în care se cere ca achiziția unor date să se realizeze la intervale prestabilite de timp. În aplicațiile care se vor studia, trebuie să se

analizeze variația unor valori, nu neaparat instanatenee. De aceea este necesar să se temporizeze aplicațiile respective. Aceasta înseamnă adăugarea unui **timer** (temporizator), astfel încât un ciclu se va produce la intervalul prestabilit de către acel **timer**.

Timer-ele se adaugă în diagrama bloc a aplicației. În panoul **Functions** există un meniu intitulat **Time & Dialog**, unde se pot găsi diferite funcții ce permit lucrul cu ceasul sistemului, așa cum se arată în fig.2.14.

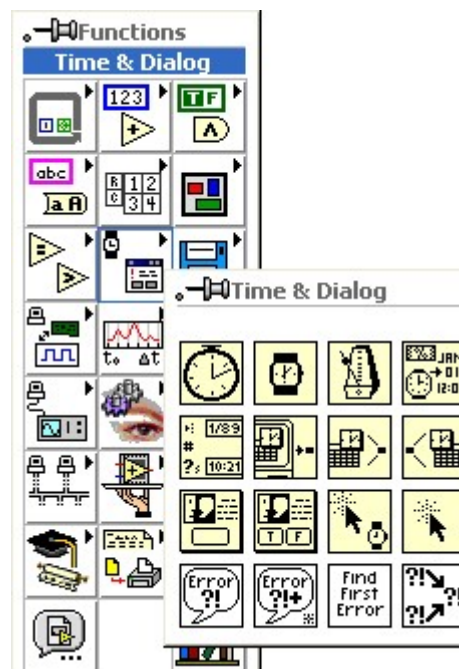


Fig. 2.14. Tipuri de **timer**-e

Spre exemplu, cu ajutorul **timer**-ului **Tick Count** (simbolul ceas de masă) se poate afla numărul de milisekunde, dar trebuie atenție, deoarece acesta generează de obicei numere foarte mari.

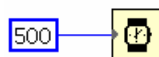


Fig.2.15. **Wait (ms)**. Constanta legată la modul indică timpul de așteptat în milisekunde

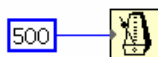


Fig.2.16. **Wait Until Next ms Multiple**

Pentru temporizarea aplicațiilor există două metode:

- **Wait (ms);**
- **Wait Until Next ms Multiple.**

Modulul **Wait (ms)** întrerupe execuția pentru un număr delimitat de milisekunde, stabilit printr-un control sau printr-o constantă. **Wait (ms)** realizează apeluri de sistem asincrone, în timp ce restul de module funcționează sincron (fig.2.15).

Deși este similară cu **Wait (ms)**, funcția **Wait Until Next ms Multiple** (fig.2.16) se folosește mai des în

cazul sincronizării activităților. Această funcție realizează apeluri asincrone la ceasul sistemului și oprește activitatea schemei până când se scurge numărul de secunde multiplu al valorii de intrare. Ambele **timer**-e depind de ceasul sistemului și nu asigură precizie la nivel de milisecundă.

În cadrul meniului **Time & Dialog** se mai pot întâlni diferite module de prelucrare a datei și ceasului, precum și o serie de module de dialog cu utilizatorul.

2.3. Crearea de SubVI-uri

Modularizarea aplicațiilor în **LabVIEW** se realizează folosind **SubVI**-uri, care se aseamănă cu **.dll**-urile Windows-ului. **SubVI**-urile au date de intrare și generează rezultate, pe baza funcției programate. Fiecare modul din cadrul paletelor **Functions** din **LabVIEW** este astfel un **SubVI**.

Crearea unui **SubVI** se realizează simplu, pentru ilustrare fiind considerat un modul care să calculeze volumul unui cilindru, știindu-se diametrul și înălțimea acestuia.

Se procedează la crearea unui nou **VI** în panoul frontal adăugându-se două **slide**-uri, unul destinat înălțimii și celălalt pentru diametru. Ca indicatoare se folosesc unul analogic de tip **tank** și altul numeric.

După ce s-au stabilit terminalele, cu **Ctrl+E** se adaugă în diagrama bloc

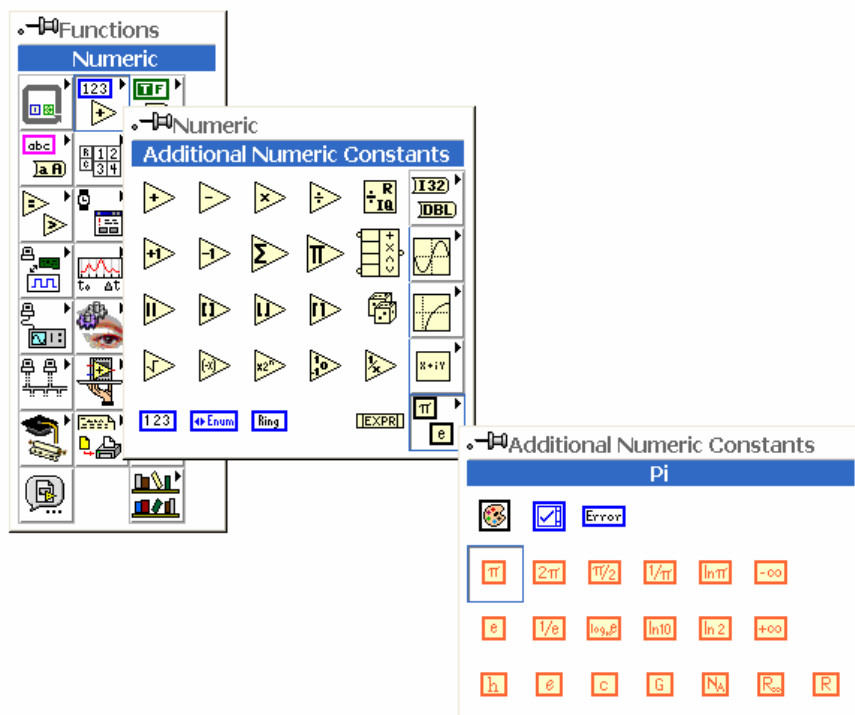


Fig.2.17. Constanta Pi

funcțiile și conexiunile necesare. Se știe că volumul cilindrului este $V = \frac{\pi d^2}{4} * h$ și,

ca urmare, vor trebui trei module de înmulțire (pentru ridicarea lui d la pătrat, pentru înmulțire cu h și cu π) și unul de împărțire la 4. Constanta π se găsește în meniul **Numeric** >> **Additional Numeric Constants** >> **Pi**, așa cum se poate observa în fig.2.17.

După ce s-au introdus cele trei module, se execută conexiunile între ele în așa fel încât să se obțină formula volumului cilindrului. Schema de conexiuni este redată în fig.2.18.

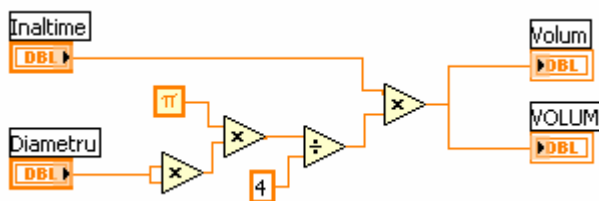


Fig.2.18. Diagrama bloc pentru calculul volumului unui cilindru

Se selectează toate elementele schemei și apoi, cu opțiunea **Edit / Create SubVI**, se realizează noul modul. În acest moment, schema va arăta ca în fig.2.19.

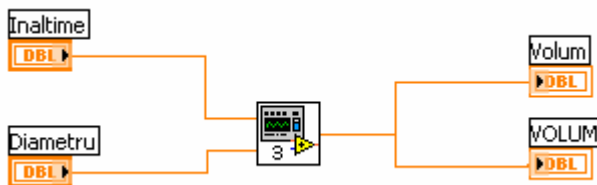


Fig.2.19. Simplificarea schemei folosind un SubVI

Revenind în panoul frontal se observă că nu există nici o diferență. Dacă se efectuează dublu click pe **SubVI**-ul nou creat, se va deschide fereastra acestuia, care conține atât un panou frontal cât și o diagramă bloc. Spre deosebire de un **.dll** tipic de Windows, un **SubVI** este tot un instrument virtual, deci trebuie să conțină terminale. De aceea, în panoul frontal al **SubVI**-ului nou creat se vor găsi două controale și două indicatoare numerice, astfel încât **SubVI**-ul este practic un **VI** de sine stătător.

În acest mod, pe măsură ce se dezvoltă o aplicație complexă **LabVIEW**, părți din aceasta pot fi transformate în **SubVI**-uri, atât pentru modularitate cât și pentru economia de spațiu în diagramă.

SubVI-ul nou creat, deschis prin dublu click, se salvează preferabil în directorul în care există aplicația la care se lucrează. În cazul de față acesta se salvează sub numele **Volum_SubVI.vi**, pentru a-l folosi în aplicațiile ulterioare. Pentru a insera un **SubVI** existent într-o aplicație există următoarele două metode:

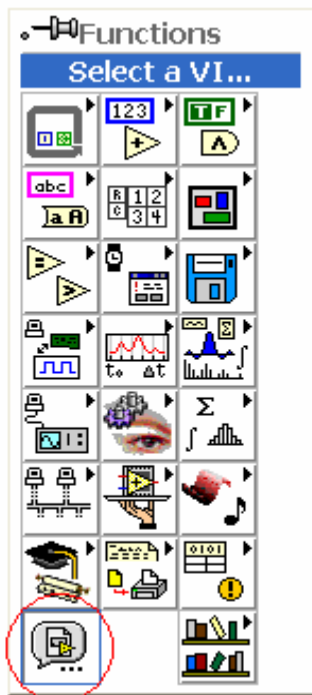


Fig.2.20. Butonul Select a VI

- se folosește butonul **Select a VI** din paleta **Functions**, ceea ce are ca rezultat deschiderea unei casete de dialog de tip **Open File**, care deschide automat directorul **My Documents** (sau alt director în care s-au salvat). În această casetă se navighează până la aplicația care se dorește a fi inserată (fig.2.20);

- se aplică un procedeu destul de utilizat în mediul Windows. În **Windows Explorer**, sau **Total Commander**, sau în orice alt file browser se navighează până la **SubVI**-ul care trebuie inserat. Se efectuează click stânga pe fișierul respectiv și se ține butonul stâng apăsat. După ce fișierul a fost selectat, cu butonul stâng al mouse-ului încă apăsat, se deplasează spre taskbar-ul Windows-ului, spre tab-ul ferestrei diagramei bloc a aplicației care se dorește a fi dezvoltată. Menținând mouse-ul deasupra acesteia, după 1-2 secunde se activează fereastra și se maximizează. După aceea, se mută cursorul în interiorul schemei și se eliberează butonul stâng al mouse-ului. Trebuie menționat că în tot acest timp butonul stâng al mouse-ului trebuie să fie apăsat.

Dacă fereastra **Context Help** este activată, se poate observa că, atunci când mouse-ul se află deasupra modulului nou inserat, se afișează date despre intrările și ieșirile acestuia. De aceea este util să se denumească sugestiv terminalele **SubVI**-ului, pentru ca un alt utilizator să își poată da seama ușor

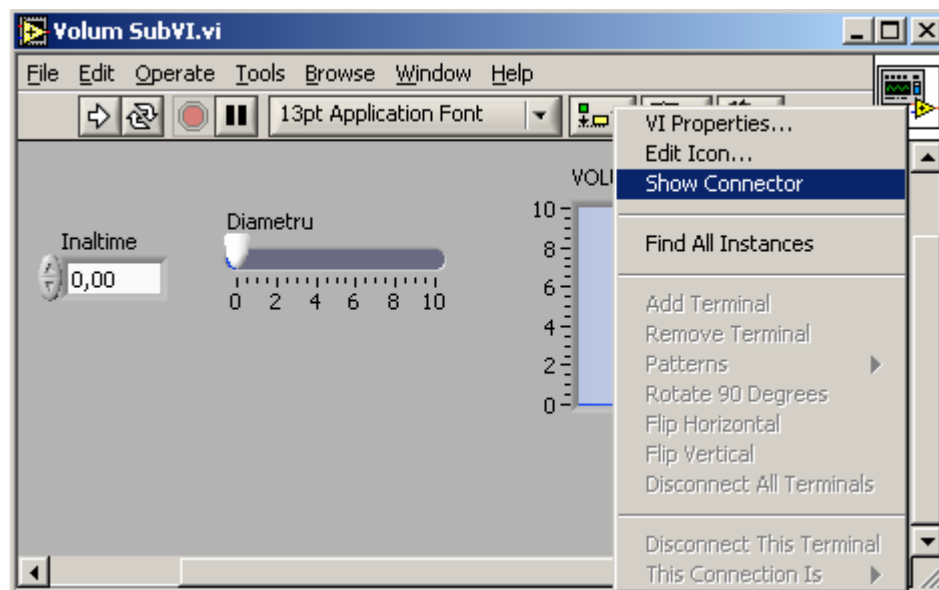


Fig.2.21. Conexiunile unui SubVI

cum să poată folosi modulul.

Odată ce a fost creat un **SubVI**, se poate modifica numărul de intrări sau de ieșiri. Pentru aceasta se deschide **SubVI**-ul și în panoul frontal se efectuează click pe icoana aplicației situată în dreapta sus. Se selectează **Show Connector**, ca în fig.2.21 și se observă că icoana se transformă într-un caroiaj, în care se pot distinge numărul de intrări și numărul de ieșiri. Intrările se găsesc în jumătatea stângă a caroiajului, în timp ce ieșirile ocupă jumătatea dreaptă.

După ce a fost selectată opțiunea **Show Connector** și icoana aplicației devine un caroiaj corespunzător intrărilor și ieșirilor, cursorul se transformă în cel pentru conectare de fire. Când se alege o intrare sau o ieșire, în panoul frontal apare selectat controlul sau indicatorul care corespunde respectivei intrări sau ieșiri.

Pentru a modifica ordinea terminațiilor se selectează una dintre ele și apoi, în meniul care apare în urma apăsării butonului dreapta al mouse-ului, se alege opțiunea **Disconect This Terminal**. În acest moment, terminația devine albă. Cu un click stânga se selectează și apoi se execută click pe noul terminal, care va corespunde terminației respective. Trebuie menționat că două terminații ale unui **SubVI** nu pot avea același terminal în panoul frontal.

Dacă se efectuează click dreapta pe o intrare - după ce înainte s-a activat opțiunea **Show Connector** și icoana **SubVI**-ului s-a transformat în caroiaj, se observă opțiunea **This Connection is**:

- **Required** – dacă nu este conectat un fir la această intrare, se generează eroare;
- **Recommended** – neconectarea unui fir poate produce erori, însă aplicația va rula fără o conexiune la această intrare;
- **Optional** – conexiunea la intrare sau la ieșire nu este neapărat necesară și nu produce eroare. Spre exemplu, în cazul salvării rezultatelor într-un fișier, dacă intrarea lipsește, se va deschide o casetă de dialog tipică Windows, care va solicita numele fișierului în care să se salveze datele.

În cazul ieșirilor, există doar opțiunile **Recommended** și **Optional**. Atât pentru intrări, cât și pentru ieșiri, opțiunea implicită este **Recommended**.

Când o intrare, de tipul **Recommended** sau **Optional**, nu este conectată, atunci intrarea implicită va fi 0.

În cazul calculului volumului, ambele intrări sunt necesare, atât înălțimea, cât și diametrul. Pentru aceasta se va selecta intrarea corespunzătoare înălțimii și din meniul **This Connection Is** se alege **Required**. Se repetă procedeul pentru diametru.

O altă aplicație ar fi aceea de a înmulți volumul cu un alt număr variabil, de exemplu densitatea pentru a determina masa. Pentru aceasta se adaugă în panoul frontal un slide care va reprezenta noul număr. Se redenumeste eticheta pentru a ști ulterior ce anume corespunde intrării care va fi creată. După ce s-au efectuat aceste operații, cu click dreapta pe icoana din colțul din dreapta sus se selectează **Show Connector**, dacă aceasta nu era deja selectată. Apoi, în partea stângă a caroiajului unde este zona destinată intrărilor, din nou cu click dreapta, se alege opțiunea **Add Terminal**. Se observă apariția unei noi intrări, marcată printr-o zonă albă. Prin selectare aceasta devine neagră, după care, în panoul frontal, se efectuează click stânga

pe slide-ul nou creat. Intrarea a devenit portocalie, și astfel s-a adăugat **SubVI**-ului încă o intrare. Scalarea volumului este o intrare opțională și se realizează acest lucru din submeniul **This Connection Is** (fig.2.22).

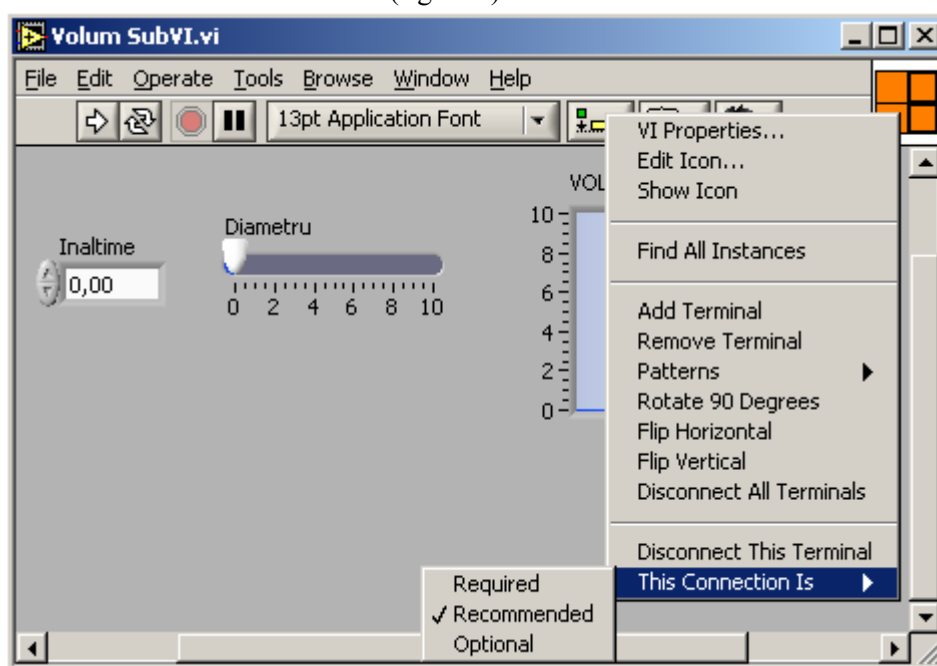


Fig.2.22. Stabilirea tipului de conexiune a terminației unui **SubVI**

2.4. VI pentru deteminarea volumului ocupat, la variația aleatoare a înălțimii agentului într-un rezervor

Următoarea aplicație se referă la realizarea unui **VI** care să măsoare volumul ocupat de lichidul stocat într-un **Tank** (rezervor) în condițiile în care înălțimea (nivelul) acestuia variază aleator în timp. Aplicația se cere să poată fi oprită de către utilizator prin intermediul unui buton – altul decât cel din **Toolbar**. Valoarea înălțimii va putea fi aleasă la o valoare impusă de către operator, la care se va adăuga o variație aleatoare într-un interval $[0,1]$ generată la un interval de 500 milisecunde.

În panoul frontal se plasează un **slide**, care este denumit **Inaltime**. Pentru a vedea și valoarea aleasă de operator se mai adaugă un indicator digital. Ambele terminale se selectează din meniul **Numeric** al paletei **Controls**. Cu ajutorul butonului **Operate Value** din paleta **Tools** se setează înălțimea la valoarea 5.00, după care, cu click dreapta, se selectează **Data Operations** și apoi **Make Current Value Default**. La fel se procedează și cu indicatorul digital. Este de observat că, atunci când cursorul se află deasupra câmpului indicatorului, se transformă automat într-un cursor tipic de editare text, cu care se poate modifica valoarea indicatorului. Deci se scrie valoarea 5.00 și apoi se setează să fie implicită.

Pentru estetica aplicației, se alege un dreptunghi de încadrare din meniul **Decorations**, și anume **Thick Lowered Box**. Se aranjează indicatorul sub slide și apoi peste ele se trage caseta de încadrare, care a fost în prealabil dimensionată corespunzător. Când se re poziționează caseta peste slide și indicator, se poate constata că, deși cele două terminale sunt vizibile, asupra lor nu se poate acționa. Se poate verifica acest lucru cu ajutorul butonului **Operate Value** din paleta **Tools**. Evident că, dacă în panoul frontal nu sunt accesibile, nu vor fi accesibile nici la rularea aplicației.

Decorația trebuie să fie “în spatele” controalelor și pentru aceasta, după selectare, cu ajutorul butonului **Reorder** din **Toolbar**, se re poziționează în spatele

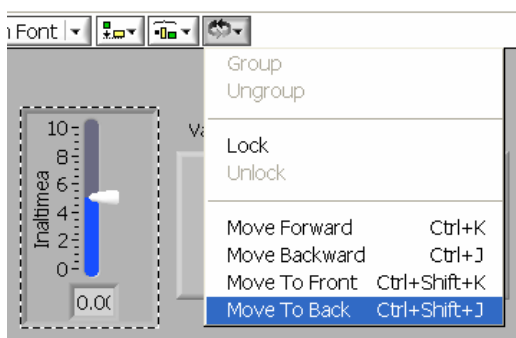


Fig.2.23. Stabilirea ordinii elementelor pe ecran

slide-ului și a indicatorului. Din meniul tip pop-up, care se deschide când se execută click pe butonul **Reorder**, se alege opțiunea **Move To Back**, așa cum se poate vedea în fig.2.23.

După ce s-a stabilit ordinea celor trei elemente, se selectează cu ajutorul **Size/Select Tool** și tot din meniul **Reorder** se alege opțiunea **Group**. Această opțiune are efect doar asupra terminalelor din panoul frontal; dacă se comută cu **Ctrl+ E** în diagrama bloc nu se observă nici o schimbare.

Mai mult, chenarul decorativ de încadrare care a fost adăugat nu apare aici.

Obiectele grupate, la fel ca și în Microsoft Word de exemplu, sunt privite ca un singur obiect. Grupul creat poate fi folosit în orice aplicație **LabVIEW** folosind procedura **Copy/Paste**. Totodată, obiectele pot fi separate folosind opțiunea **Ungroup** din meniul **Reorder**.

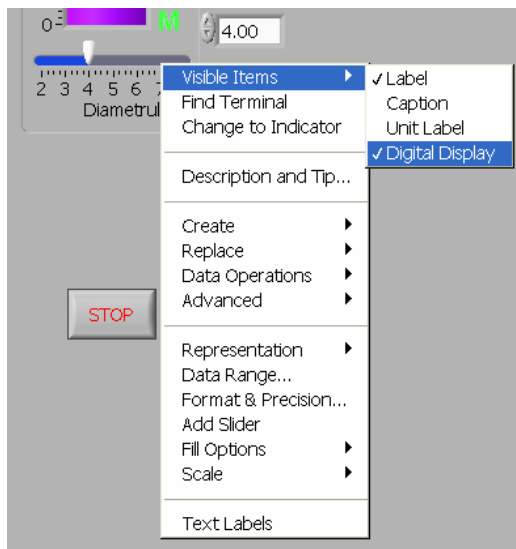


Fig.2.24. Afișarea controlului digital atașat slide-ului

După ce s-a finalizat grupul pentru variația înălțimii, se realizează, similar, un nou grup pentru afișarea volumului. Grupul va fi format dintr-un **Tank**, cu scară 0...500, un indicator digital pentru afișarea valorii instantanee a volumului și un control pentru variația diametrului.

Controlul **Diametru** este un **Horizontal Pointer Slide**, ales din meniul **Numeric** al paletii **Tools**. Se stabilește scala de variație a diametrului între 2 și 8 și se selectează valoarea implicită ca fiind 4. Pentru a ajunge la o valoare exactă, se efectuează click dreapta pe slide și din meniul care se deschide se alege **Visible Items** >>

Digital Display, așa cum se poate observa din fig.2.24. În cadrul acestui control se selectează valoarea 4.00. Trebuie menționat că săgețile de variație ale controlului digital determină schimbarea valorii întregi a numărului. De aceea, o modalitate mai facilă și mai rapidă este aceea bazată pe folosirea butonului **Operate Value** din paleta **Tools**. În acest moment, se schimbă și valoarea de pe slide. După ce s-a realizat această operație, se poate ascunde acest control, întrucât este redundant. Operația este similară cu cea pentru afișare.

Ca o paranteză, dacă acest **Digital Display** care a fost afișat pentru a stabili valoarea exactă este transformat în indicator, atunci tot slide-ul se transformă în indicator. De aceea, dacă se dorește vizualizarea valorii instantanee a unui slide, trebuie să i se atașeze un indicator, așa cum s-a procedat în cazul înălțimii. **Digital Display**-ul este tot un control și practic valoarea slide-ului poate fi modificată fie normal, cu ajutorul săgeții slide-ului, fie prin intermediul **Digital Display**-ului. Prin procedeul descris s-a finalizat controlul pentru diametru.

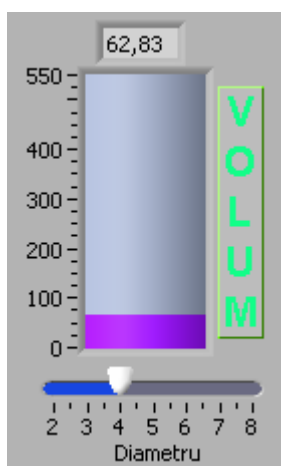


Fig.2.25. Grupul de afișare a volumului

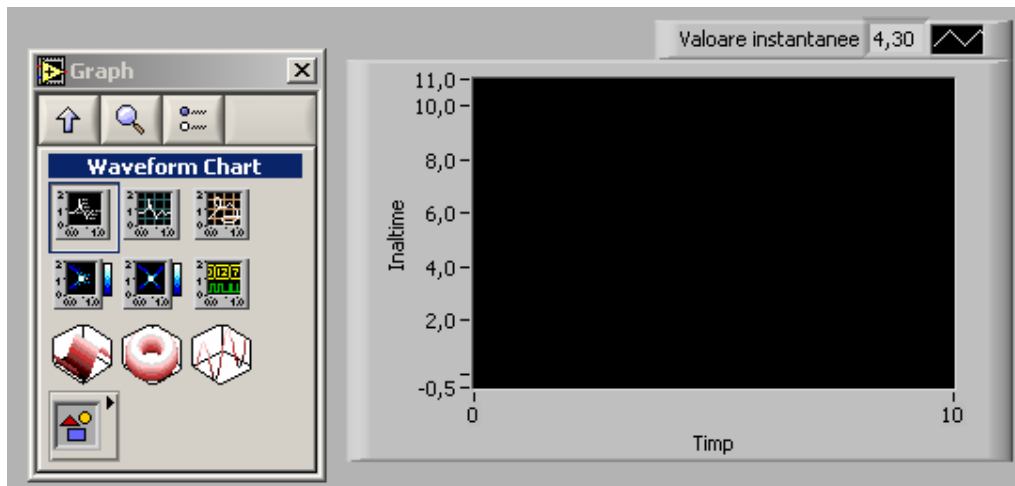
Tank-ul reprezintă valoarea volumului, de aceea valoarea sa va trebui să fie volumul cilindrului cu înălțimea 5 și diametrul 4, adică 62.83. Se realizează această valoare implicită la fel ca în cazul slide-ului, cu ajutorul **Digital Display**-ului. **Tank**-ul, fiind un indicator digital, display-ul va fi de asemenea un indicator. Precizia implicită a **Tank**-ului este de 2 zecimale. Dacă se modifică această precizie, schimbarea se va reflecta și pe scala de valori, de aceea se va menține la valoarea 2. Se scrie în indicator valoarea 62.83 și se setează ca valoare implicită, după care se selectează opțiunea **Digital Display**. Pentru a îmbunătăți partea grafică a aplicației, se selectează eticheta **Tank**-ului, se redenumesc **Volum** și, cu click dreapta, se alege **Vertical Arrangement >> Stacked**. În acest moment, eticheta va fi afișată vertical, în lungul **Tank**-lui. După aceea, se adoptă pentru etichetă culoarea verde și mov pentru culoarea de umplere a **Tank**-ului.

Pentru o precizie mărită a volumului, opțional se poate selecta indicatorul numeric, se stabilește pentru aceasta trei zecimale în meniul **Format & Precision** și se modifică valoarea sa implicită la 62.832.

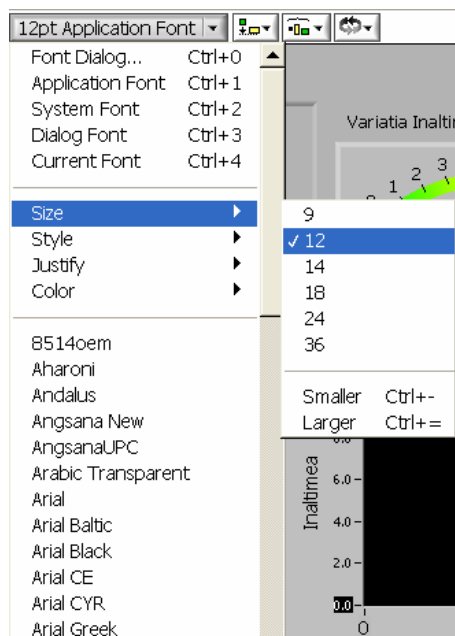
După ce s-au realizat toate aceste operații, se adaugă o decorație de tipul **Lowered** și se încadrează grupul de elemente care au fost prelucrate mai sus. Se stabilește ordinea de afișare pe ecran, în așa fel încât să se poată opera asupra controalelor și apoi acestea se grupează. Rezultatul final este prezentat în fig.2.25.

Pentru variația instantanee a înălțimii se aleg trei tipuri de indicatoare. Mai întâi unul de tip **Meter**, care se denumește **Variatia inaltimii**, și a cărui valoare implicită se setează la 5,00.

Următorul indicator va fi un grafic pentru a putea observa variația în timp a valorilor instantanee. Pentru aceasta, din paleta **Functions** se selectează grupul **Graph** și apoi **Waveform Chart** (fig.2.26).

Fig.2.26. Adaugarea unui **Waveform Chart**

Asupra **Waveform Chart**-ului nou introdus se vor efectua câteva operații de personalizare. Mai întâi, folosind butonul **Edit Text** din paleta **Tools**, se schimbă denumirile celor două axe în **Inaltime** și **Timp**. Apoi se schimbă scala pe înălțime în așa fel încât să ajungă în intervalul -0,5...11. Se observă că fontul este prea mare și pe scală apar prea puține valori. Pentru a rezolva această problemă, va trebui să se micșoreze fontul, în acest scop selectându-se o valoare de pe scală și apoi din meniul **Font** dimensiunea convenabilă. O altă modalitate, mai simplă, pentru a redimensiona

Fig.2.27. Stabilirea fontului în **LabVIEW**

fontul este prin folosirea combinațiilor de taste **Ctrl +** sau **Ctrl -** pentru incrementarea, respectiv decrementarea fontului, după ce în prealabil a fost selectată o valoare de pe scală, așa cum se poate observa în fig.2.27.

Eticheta **Chart**-ului se ascunde acționând butonul dreapta al mouse-ului în zona acestuia și selectând **Visible Items / Show Label**.

Următorul indicator este unul digital care va fi denumit **H Instanatneu**. Eticheta sa se ascunde printr-o procedură similară cu cea de la **chart**. Se setează valoarea implicită a indicatorului la valoarea 5,00.

În partea de sus a **Chart**-ului există o zonă unde este indicat tipul de grafic și culoarea acestuia, zonă care poate fi redimensionată. Se redimensionează în partea stângă și se înlocuiește textul **Plot 0**

cu **Valoare instantanee**, după care, în spațiul rămas liber între text și tipul de grafic, se aduce indicatorul de valoare instantanee. Următorul pas este reprezentat de gruparea celor două elemente, prin aceasta rezultând un nou obiect, asemănător cu cel din fig.2.26.

În continuare se adaugă un buton de tip **Stop**, care se redimensionează. În final se va obține un panou frontal asemănător cu cel din fig.2.28.

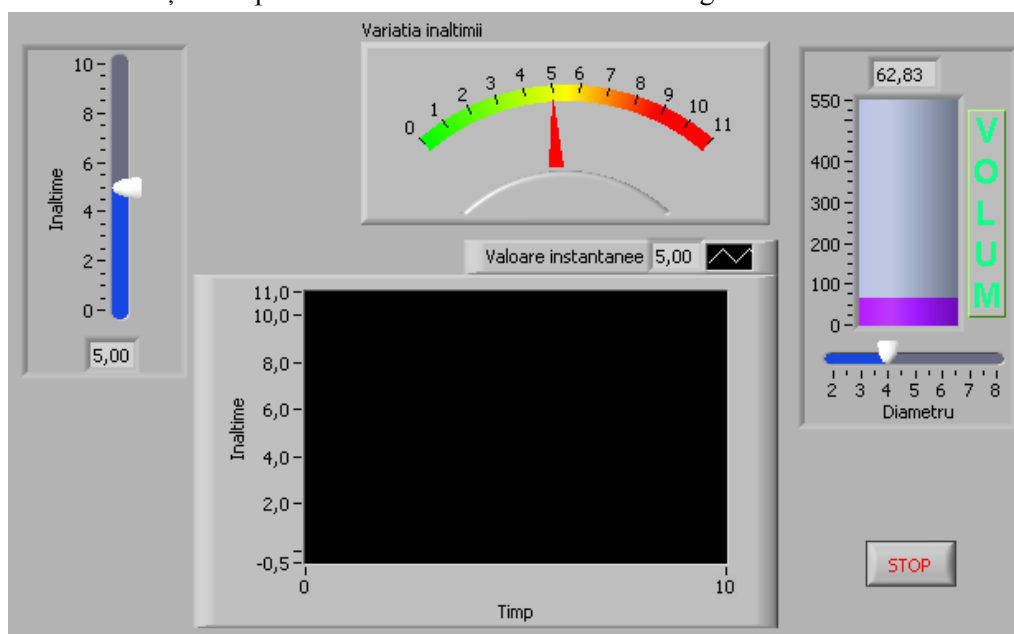


Fig.2.28. Panoul frontal al aplicației

După ce s-a terminat aranjarea obiectelor în panoul frontal, cu **Ctrl+E** se trece în diagrama bloc a aplicației. Primul pas va consta în aranjarea indicatoarelor și controalelor în funcție de necesități.

În partea stângă se aduce controlul pentru variația înălțimii și, lângă el, indicatorul corespunzător. Având în vedere cerința de variație aleatoare a înălțimii pe un interval $[0,1]$, se selectează subpaleta **Numeric** din paleta **Functions** și de aici se alege generatorul de numere aleatoare **Random Number (0-1)**, care se plasează în vecinătatea controlului pentru înălțime. Mai trebuie un modul de sumare care se alege tot din subpaleta **Numeric**. Se conectează controlul înălțimii cu modulul de sumare și cu indicatorul corespunzător. Generatorul de numere aleatoare se leagă doar la modulul de sumare.

În dreapta modulului de sumare se plasează indicatorul tip **meter**, care are eticheta **Variatia Inaltimeii**, **chart**-ul și indicatorul pentru înălțimea instantanee. Toate aceste indicatoare se conectează la ieșirea modulului sumă.

Urmează conectarea terminalilor și modulelor necesare pentru calculul și afișarea volumului. Pentru aceasta, din paleta **Functions** se alege **Select a VI**. Rezultatul va consta în deschiderea unei ferestre de dialog Windows tip Open File, de

unde se selectează **SubVI**-ul dorit, în acest caz **Volum_SubVI.vi**, dezvoltat anterior, care se plasează convenabil, așa cum se poate observa în fig.2.29.

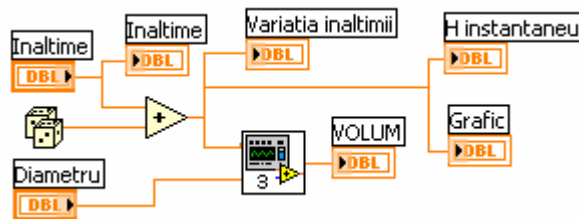


Fig.2.29. Schema de conectare a modulelor

După cum a fost definit **SubVI**-ul, se cunosc intrările necesare. La aceste intrări se conectează controlul pentru diametru și ieșirea de la modulul sumă. Ieșirea va fi conectată la indicatorul de tip **tank** și – implicit - la indicatorul digital pentru volumul instantaneu.

Pentru partea de temporizare a aplicației se alege un modul tip **Wait Until Next ms Multiple** din grupul **Time & Dialog**. La intrarea acestuia se conectează o constantă selectată din grupul **Numeric** și căreia i se atribuie valoarea 200 (a se vedea fig.2.16).

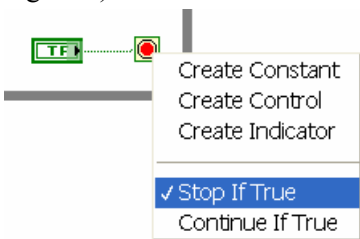


Fig.2.30. Oprirea buclei **While**

Pentru ca aplicația să poată fi oprită de utilizator este necesar să se includă (toată aplicația) într-o buclă de tipul **While** (pentru detalii privind bucla **While** a se vedea **aplicația II.3**). Se selectează tipul buclei din subpaleta **Structures** și se observă schimbarea formei cursorului într-un pătrat. Similar selecției mai multor obiecte într-o aplicație, “se trage” un dreptunghi care să încadreze atât aplicația, cât și partea de temporizare. După ce s-a efectuat această operație, se aduce terminalul

corespunzător butonului **Stop** în cadrul buclei, dacă nu se afla deja acolo, și se conectează la indicatorul de terminare a buclei **While**. Se efectuează click dreapta pe acest indicator și se selectează condiția **Stop If True** ca în fig.2.30, pentru ca aplicația să se sfârșească atunci când a fost apăsat butonul de tipul control boolean, adică s-a

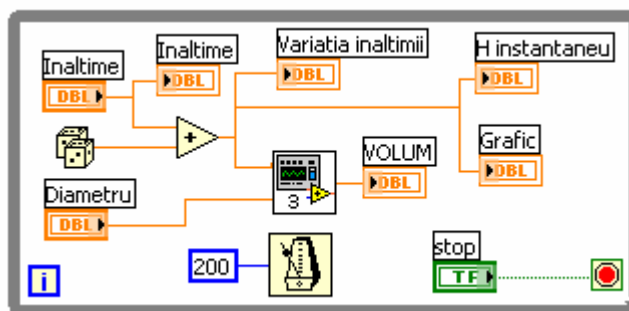


Fig.2.31. Diagrama finală a aplicației

efectuat o schimbare de stare din **False** în **True**.

Dacă operațiile au fost executate conform celor descrise mai sus, schema are forma prezentată în fig.2.31, este funcțională și acum aplicația se poate rula. Spre deosebire de exemplele anterioare, în cazul de față aplicația rulează doar dacă este apăsat butonul **Run** sau combinația de taste **Ctrl+R**. Nu mai este necesară rularea folosind comanda **Run Continuously**. În general, în dezvoltarea aplicațiilor **LabVIEW**, mai puțin în cazul celor de test, este de preferat să se ofere utilizatorului posibilitatea de a opri aplicația fără a folosi butoane din **Toolbar**, ceea ce implică introducerea aplicației într-o buclă de tip **While**.

Dacă se comută înapoi în panoul frontal al aplicației, se poate observa variația înălțimii și a volumului. Dacă se acționează asupra comutatorului înălțimii, se va schimba valoarea de bază, variația va continua și, evident, se va reflecta asupra volumului.

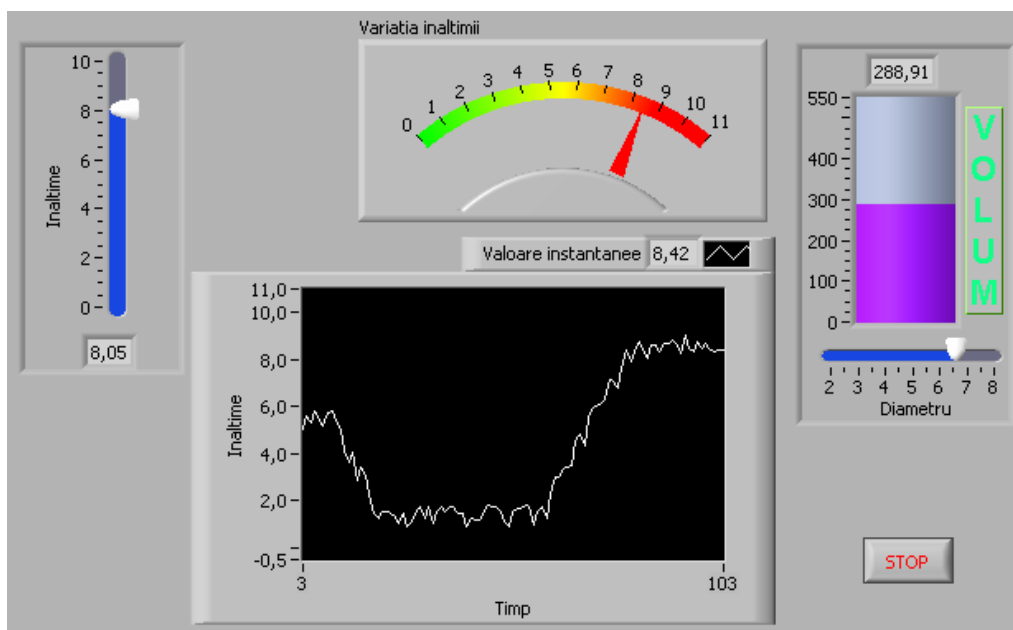


Fig.2.32. Rularea aplicației

În fig.2.32 este arătată imaginea panoului frontal care redă rularea aplicației. Salvarea se va face sub denumirea **Calcul volum cilindru_[nume / prenume student]_[grupa].vi** în directorul aferent seriei/grupeii.

2.5. Chestiuni de studiat

1. Se va realiza **VI**-ul pentru calculul sumei și produsului, urmărindu-se aranjarea cât mai convenabilă a elementelor în panoul frontal și în diagrama bloc.
2. După crearea **VI**-ului, se va executa rularea aplicației în toate modurile, inclusiv cu afișarea valorilor transmise între diversele blocuri din diagramă. Se vor

utiliza atât numere întregi cât și fracționare. Se va efectua depanarea schemei prin verificarea valorilor furnizate de modulele existente în diagrama bloc. Se va întrerupe una din conexiuni (sau se va lega eronat) și se va observa ce se întâmplă.

3. Se va deschide fereastra **Context Help**, se vor reține informațiile despre elementele componente care se pot obține pe această cale și utilitatea acestora. Se va accesa și **LabVIEW Help** (help-ul general) și se vor studia informațiile pe care acesta le poate furniza, precum și comparația cu **Context Help**.

4. Folosind cunoștințele însușite, se va crea un nou **VI** care să efectueze conversia valorilor temperaturii din grade Fahrenheit în grade Celsius. Se va ține seama că la intervalul de 100 grade Celsius corespunde pe scara Fahrenheit intervalul de 212 grade și că la referința de 0 grade pe scara Celsius corespund 32 grade Fahrenheit.

5. Se va crea **SubVI**-ul pentru calculul volumului unui cilindru, panoul frontal și digrama bloc, în conformitate cu indicațiile din secțiunea 2.3. De asemenea, **VI**-ul de la punctul precedent, de conversie a temperaturii din grade Fahrenheit în grade Celsius, se va transforma în **subVI**.

6. Utilizând **SubVI**-ul pentru calculul volumului unui cilindru, se va realiza un **VI** care să permită determinarea cantității de lichid dintr-un rezervor cilindric în funcție de înălțimea (nivelul) de umplere, conform celor menționate în secțiunea 2.4.

7. Aplicația de la punctul precedent se va completa în sensul ca **VI**-ul creat să poată calcula masa de lichid ținând seama de densitatea acestuia. Se va exemplifica pentru o valoare a densității $\rho=1,200$.

2.6. Modul de lucru și prezentarea rezultatelor

Pentru punctele 1 și 2 de la **Chestiuni de studiat** se vor urmări pas cu pas indicațiile cuprinse în secțiunea 2.1 și se vor experimenta toate posibilitățile prevăzute atât în ceea ce privește realizarea panoului frontal și a diagramei bloc, precum și pentru rularea aplicației. Se vor experimenta procedurile de redimensionare, de poziționare, de scalare, de aliniere a elementelor de control și a celor indicatoare, precum și cele de alegere a tipurilor de date, a reprezentării acestora și stabilirea valorilor implicite. În diagrama bloc se va acorda atenție operației de cablare. Pentru toate experimentările executate se vor formula observații și comentarii.

Referitor la punctul 3, se va compara **Context Help** cu **LabVIEW Help** din punctul de vedere al accesibilității și conținutului informațiilor furnizate și se vor face aprecieri cu privire la oportunitatea utilizării lor.

La punctele 4 și 5 se vor efectua toate operațiile și experimentările asemănătoare cu cele prevăzute pentru **VI** în secțiunile 2.1, 2.2 și 2.4, ținând seama și de cele specifice, de simplificare a schemei, precizate în secțiunea 2.3.

Pentru punctele 6 și 7, se vor exploata avantajele utilizării de **subVI**-uri și se vor comenta rezultatele obținute în cazurile respective.

Cu privire la punctul 8, se va crea un **VI** care să permită efectuarea de operații aritmetice de adunare și înmulțire cu mai mult de doi operanzi, urmărind adoptarea celor mai simple soluții.

În ceea ce privește prezentarea rezultatelor, fiecare student, în fișierul cu numele său (deschis anterior), urmărind punctele de la chestiunile de studiat și de la modul de lucru, va înscrie răspunsurile, rezultatele și comentariile. Pentru **VI**-urile descrise în lucrare sunt suficiente rezultatele din experimentări cu explicațiile aferente. Pentru **VI**-urile și **subVI**-urile nou create, se vor prezenta și panourile frontale și diagramele bloc respective, însoțite de imagini cu grafice, eventuale explicații și observații. O atenție specială se va acorda generării și afișării sub formă de grafic a valorilor instantanee ale înălțimii care variază aleator și includerii **VI**-ului într-o buclă **While**.