

1. EXPRESII LOGICE sau BOOLEENE

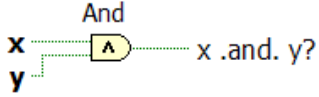
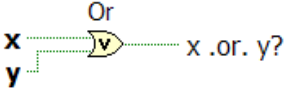
Expresia logică este o înlănțuire de:

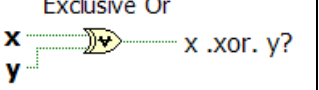
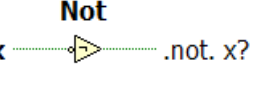
operanzi logici (True , False) și

operatori logici AND, OR, XOR, NOT etc.

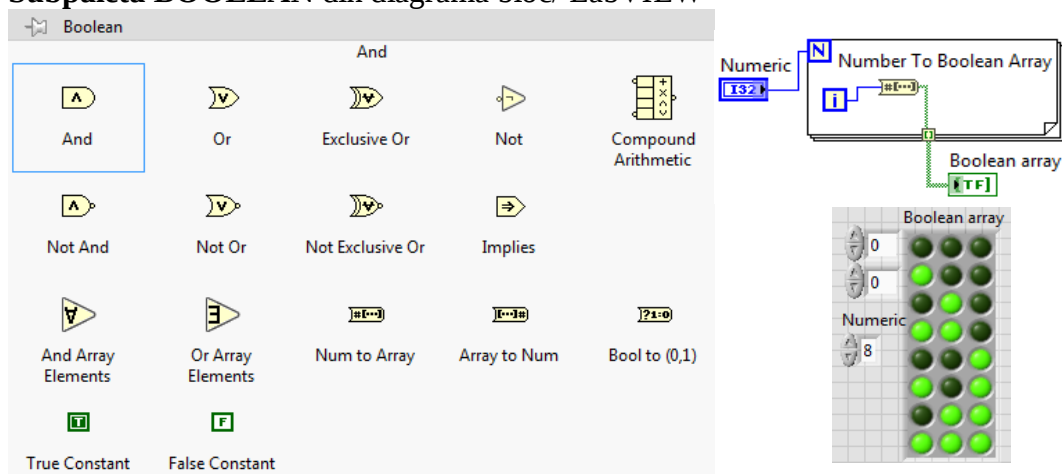
În urma evaluării expresiei logice se obține o **valoare logică (T, F)**

Exemple de expresii simple logice (tabele de adevăr):

T AND T → T F AND T → F T AND F → F F AND F → F	T OR T → T F OR T → T T OR F → T F OR F → F
COD grafic Labview - vezi simbol logic	
	

T XOR T → F F XOR T → T T XOR F → T F XOR F → F	NOT T → F NOT F → T
COD grafic LabVIEW	
	

Subpaleta BOOLEAN din diagrama bloc/ LabVIEW



2. EXPRESII RELATIONALE

Expresia relațională este o înlănțuire de:

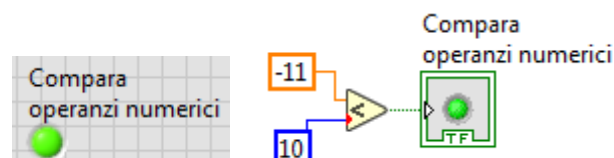
operanzi numerici, logici, șir de caractere și

operatori relaționali <, >, =, ≤, ≥ etc.

În urma evaluării expresiei relaționale se obține o **valoare logică (T, F)**

Comparații valide:

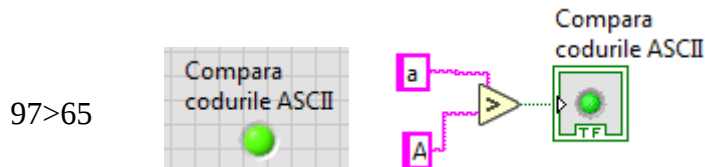
1. Operanzi valori reale



2. Operanzi valori logice:



3. Operanzi șiruri de caractere



3. Operatorul SELECT

Dacă valoarea logică primită pe intrarea de selecție (s) este True, operatorul Select permite trecerea valorii de pe ramura T (True) iar dacă valoarea primită este False permite trecerea valorii de pe ramura F (False).

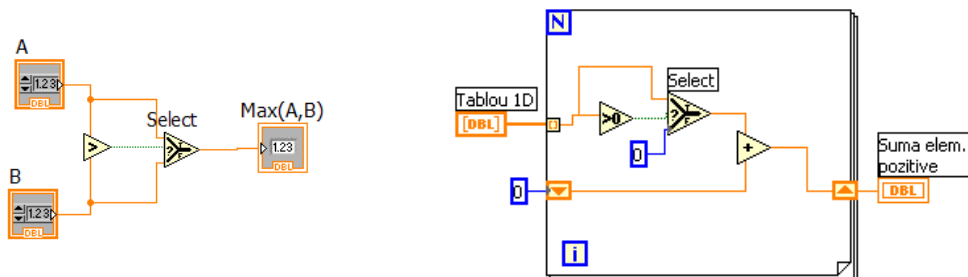


s=T	→	t trece
s=F	→	f trece

1. Selectarea unei valori numerice.

Exemplul#1: Calculul maxim dintre două valori numerice.

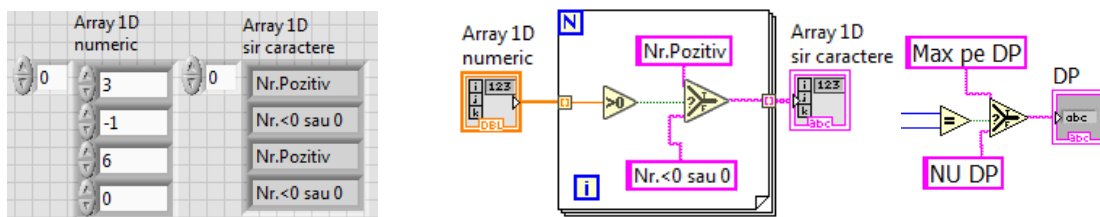
Exemplul#2: adunare selectivă, numai elemente pozitive dintr-un șir.



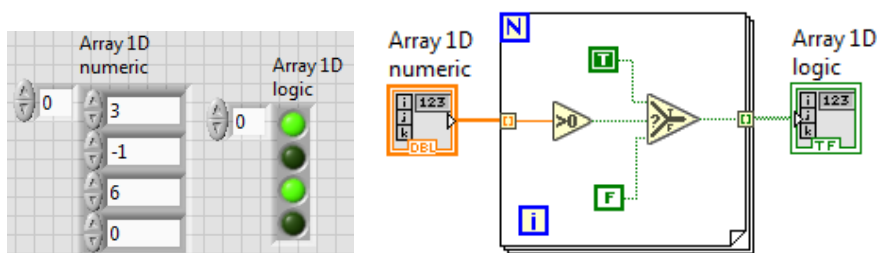
2. Selecție între două șiruri de caractere.

constante **șir de caractere** +

indicator șir de caractere →



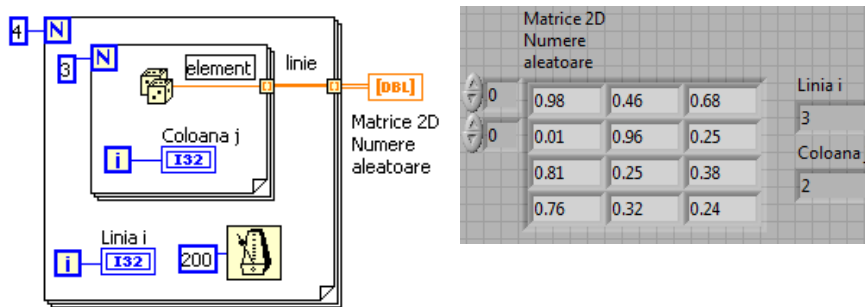
3. Selecție între două valori logice.



2. For loop în For loop (imbricat)

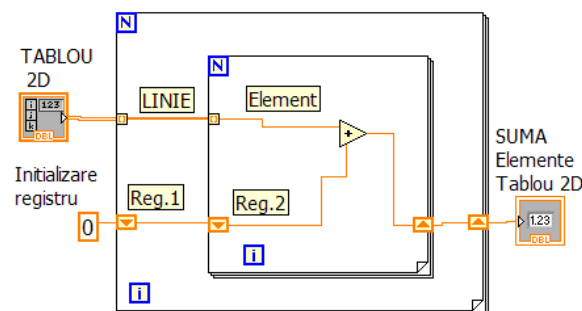
2.1. Generare matrice 2D (numere aleatoare)

- două tuneluri de ieșire din ciclul For cu indexare



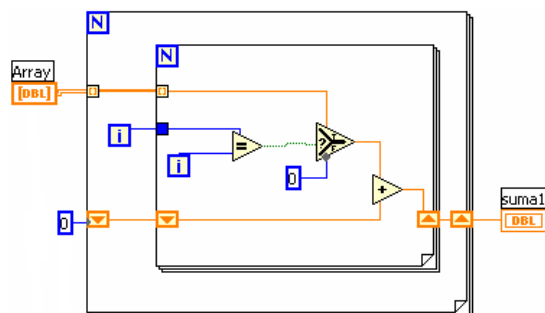
2.2. Suma elementelor dintr-o matrice:

For Loop imbricat și regiștrii de transfer asociați.



2.3. Suma elementelor de pe **diag. principală**.

(DP: $i = j$) într-o matrice A pătratică cu SELECT.



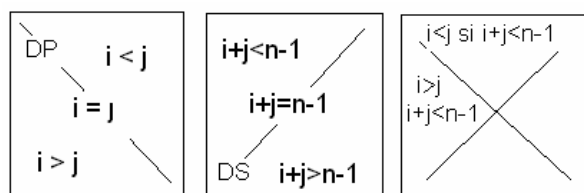
3. Suma elementelor $a_{ij} > 0$ situate subDP și subDS

într-o matrice pătratică (cu SELECT).

i, j - indice linie, coloana;

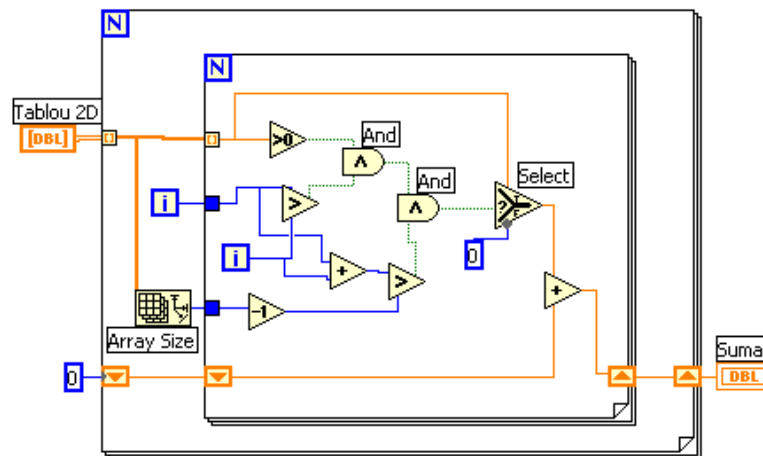
DP: $i = j$;
DS: $i + j = n - 1$;

SubDP: $i > j$;
SubDS: $i + j > n - 1$;

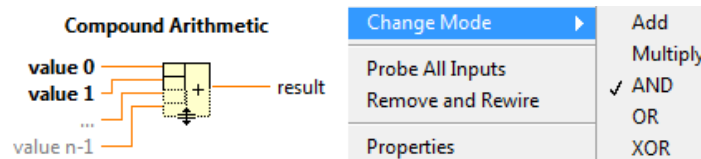


Expresii relaționale și logice

**$(a_{ij} > 0) \text{ AND } (i > j)$
 $\text{AND } (i + j > N - 1)$**

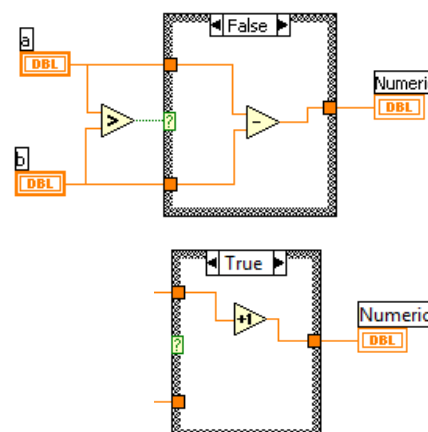


Tema #1: înlocuim cei 2 operatori AND → Comp. Arithm. AND:



4. Structura de selecție multiplă CASE

4.1. #cazuri = 2: caz T / caz F:
Selector Boolean, cazurile sunt identificate prin valori logice
(T sau F → circulă prin fir verde)

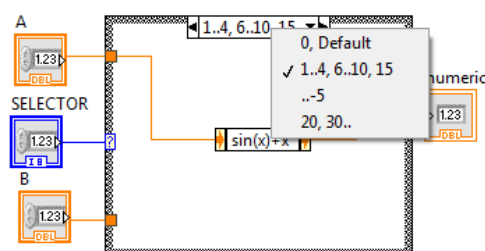


4.2. #cazuri ≥ 2:

Cazurile se identifică prin numere întregi (pozitive și negative);
exemplu: -4, 1, -2, 0, ... ⇒ selector întreg (I8),

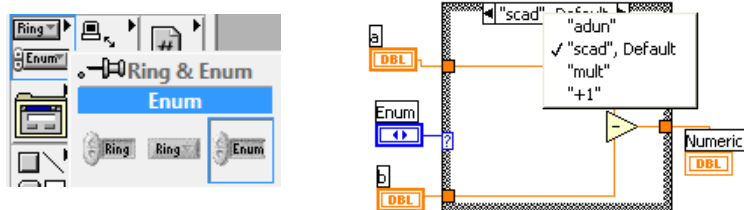
Ex.: Calcul și afișare:

Selector=0,(Default) →	Incrementare a
Selector=1..4, 6..10, 15 →	$\sin(a) + a$
Selector=..-5 →	Dif (a-b)
Selector=20, 30.. →	Suma (a+b)



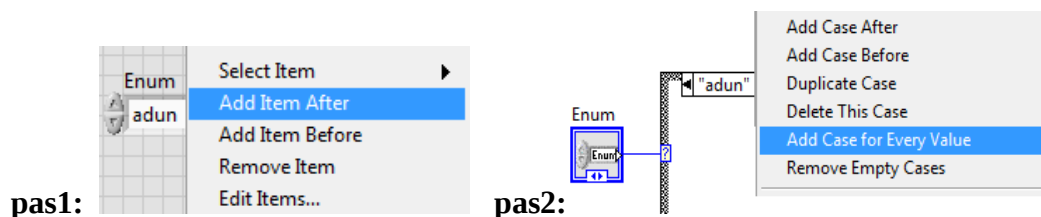
4.3. #cazuri ≥ 2: cazuri identificabile prin șiruri de caractere, selectorul de tip Enum (enumerare) de tip întreg fără semn. Se asociază fiecărui șir de caractere în ordinea din lista, valori întregi (crescător) începând cu 0 (0,1,2...)

Aplicația 1: lista șirurilor de caractere este "adun", "scad", "mult", "incr+1";

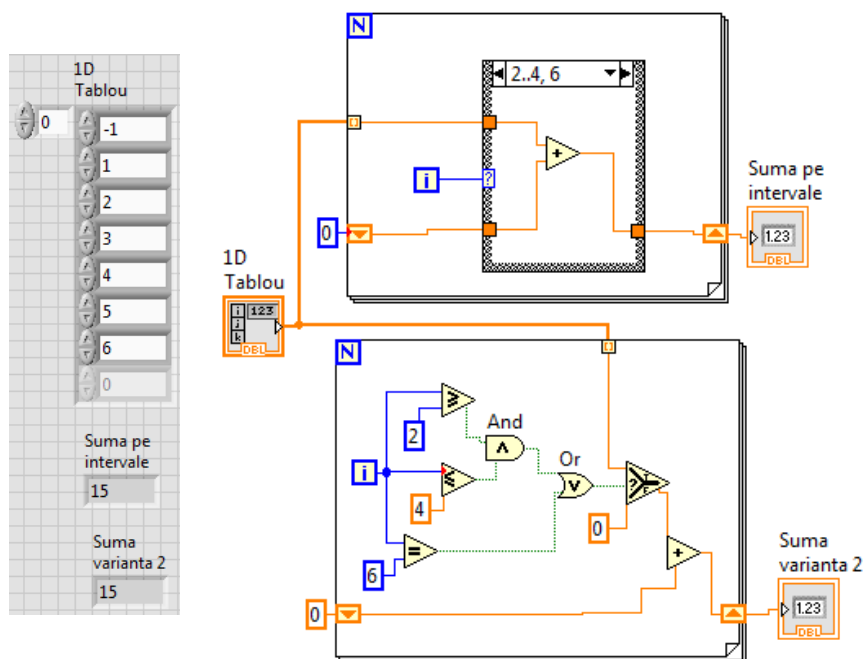


pas#1: se adauga articolele (item) 'Add Item After' dorite în controlul Enum

pas#2: se comanda 'Add Case for Every Value' din structura Case



5. Selectare (sumă) **elemente** de pe pozițiile dorite din șirul de elemente reale 1D



Tema: rescrieți exemplul folosind Case cu selector T/F

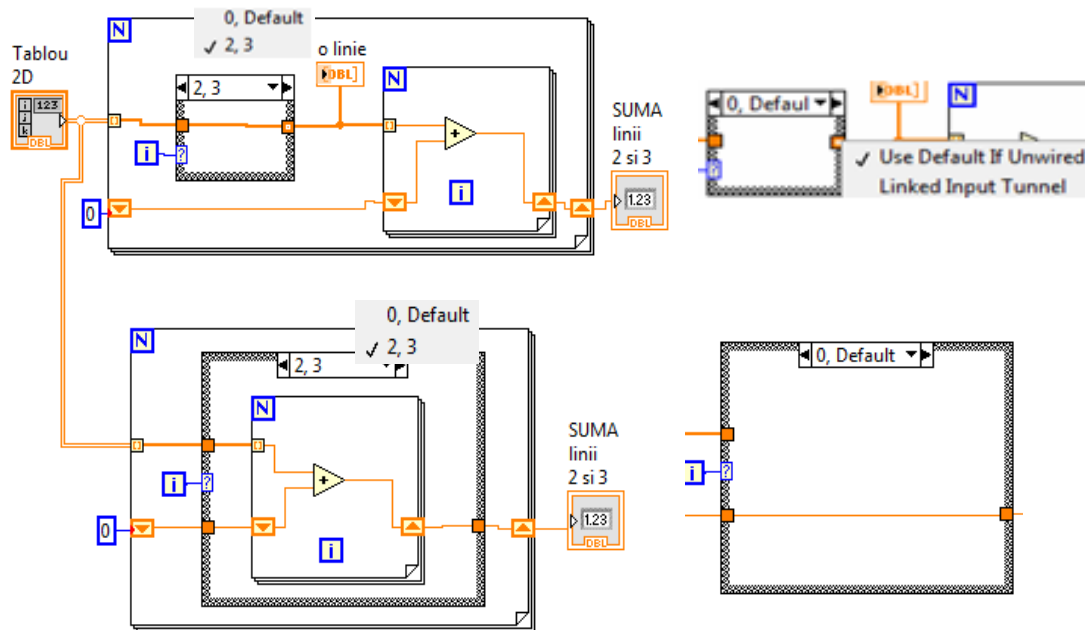
6. Selectare linii/(sumă de pe **linii**) de interes din matrice 2D

Var#1: Liniile 2 și 3 sunt trecute neschimbate **spre ciclul For**

-liniile **diferite de 2 și 3** se transferă de 0 elemente deci ciclul For nu se repetă dar valoarea din registru se transferă (sertar st -> sert. dr)

In CASE nu toate cazurile sunt conectate la **tunelul de ieșire**. Valoarea implicită transferată (Use Default if Unwired) pentru aceste cazuri este **valoarea implicită a tipului de dată** astfel:

- pentru **numeric este 0**;
- pentru șir caractere este **șirul vid**;
- pentru boolean **este False**.



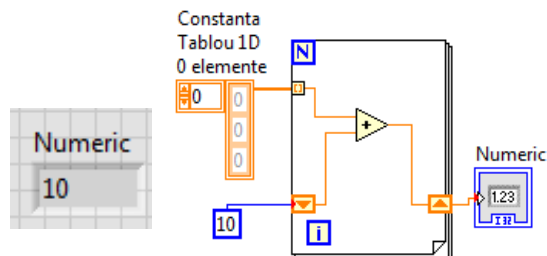
Var#2: **Contorul i** este selector.

Cazul **2,3**: liniile indice 2,3 intră în Case și sunt prelucrate.

Cazul **0,Default**: linii blocate iar suma parțială trece neafectată.

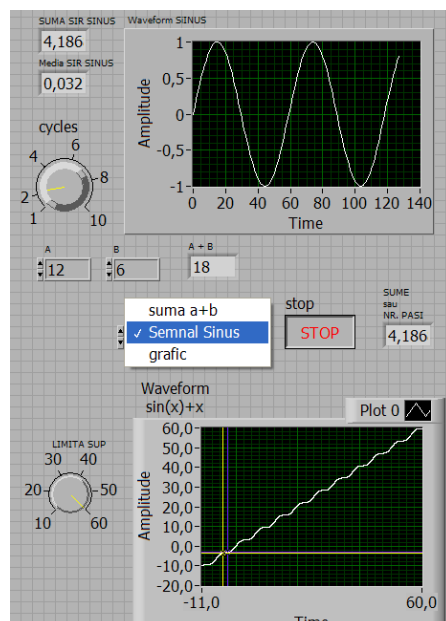
6.2: Ciclul For cu **0 iterații**

dar **reg. Shift transferă valoarea** de inițializare

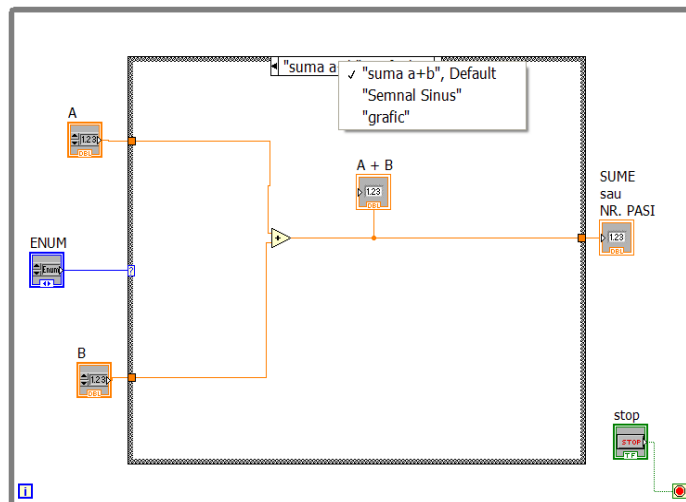


Aplicație

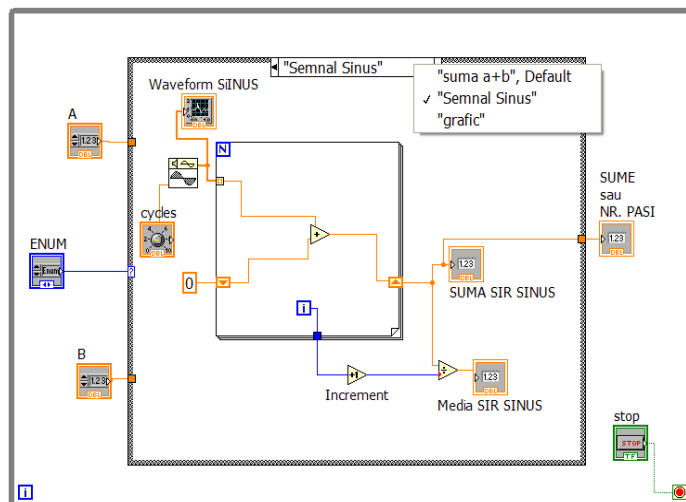
7. Acțiuni **diverse** în fiecare caz din CASE:



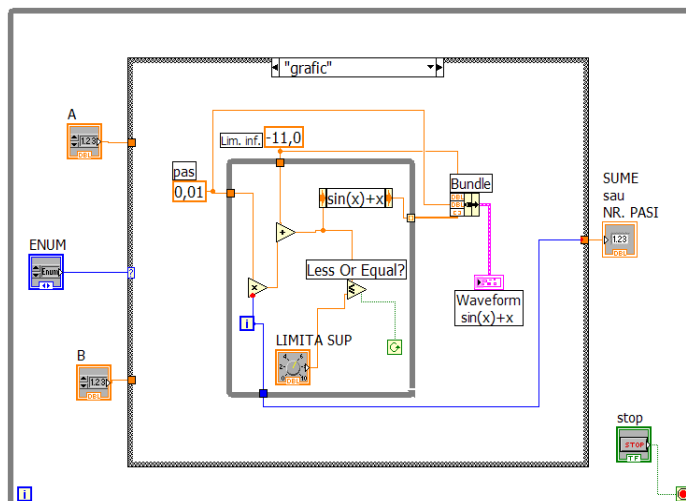
Caz 1: **suma a+b**
suma a două numere
A+B



Caz 2: **Semnal Sinus**
generare **Sine pattern**
+ grafic Sine pattern;
◦ nr. cicluri este control
◦ calcul sumă valori din semnal;
◦ calcul media aritmetică
valori din semnal



Caz 3: **grafic**
Calcul funcție **$\sin(x)+x$**
în interval: $\text{LimInf} = \text{constanta } (-11.0)$
 Limita SUP este control
în PF
+ reprezentare grafică
funcție



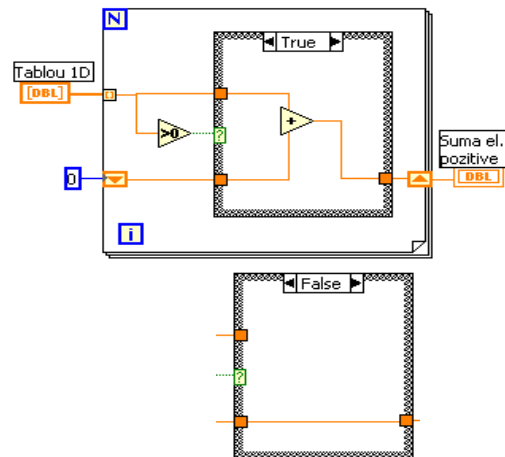
Tunel de ieșire din CASE:

Caz 1: suma A+B

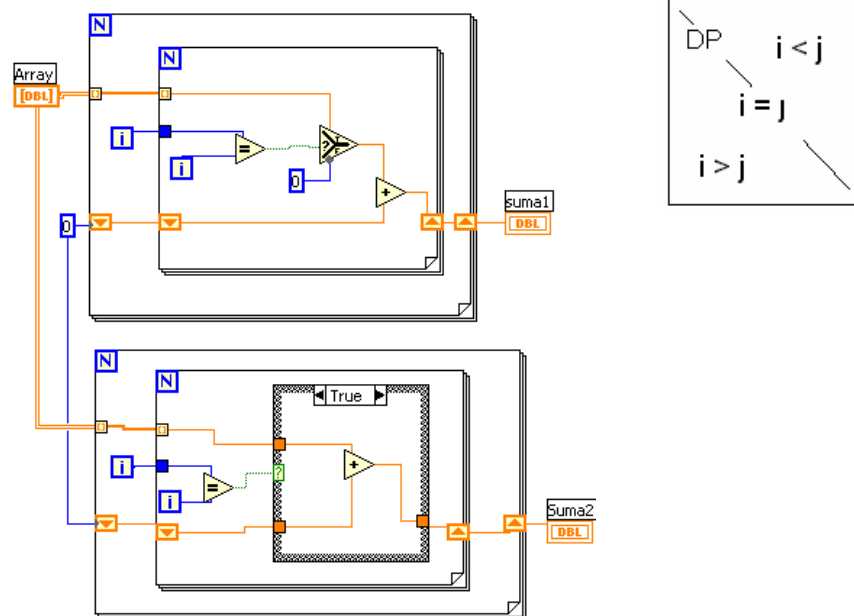
Caz 2: suma șir sinus

Caz 3: număr de pași (iterații)

8. Adunare selectivă,
numai **elemente pozitive**
dintr-un șir (Tablou 1D)
folosind structura **CASE**



9. Sumă elemente de pe DP (matrice **2D pătratică**) folosind:
CASE comparativ cu **Select**

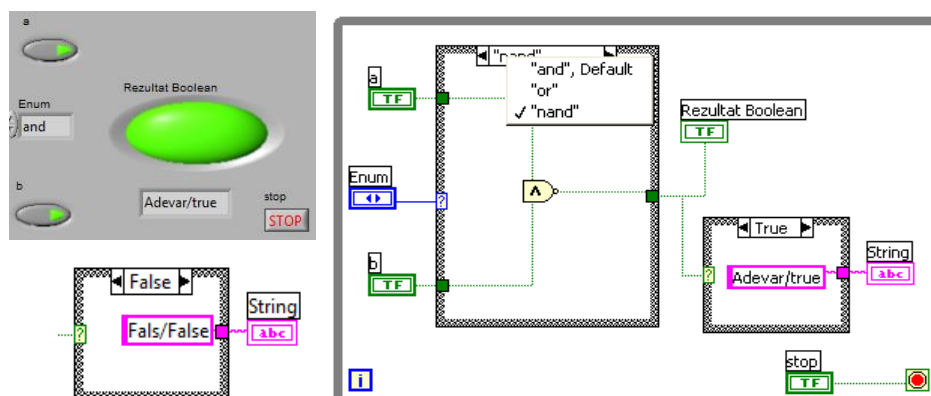


10. Implementarea unor expresii logice folosind CASE

Operandi logici: **T, F**

Operatori logici:

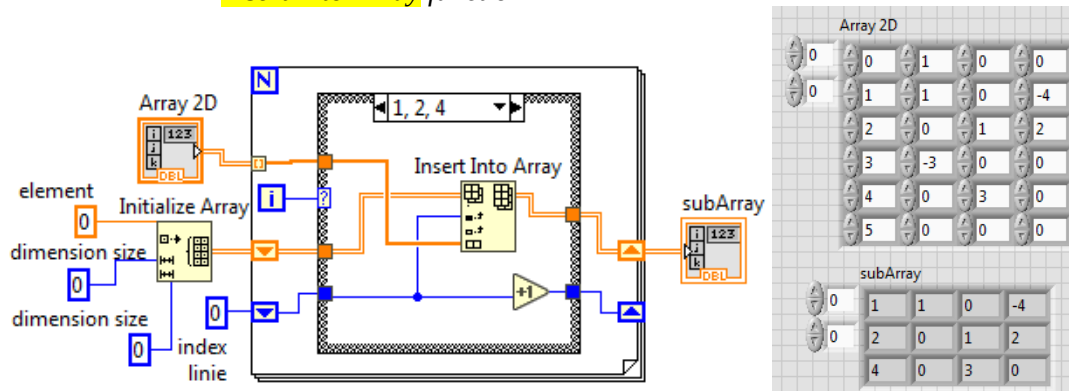
AND, OR, XOR, NOT, NAND, etc. și expresii logice:



- selectăm din controlul Enum cazul dorit (unul din 3 cazuri)
- setăm fiecare control logic pe valoarea logică T/F dorită
- rulăm aplicația și se va evalua expresia logică selectată
- rezultă o valoare logică pentru indicatorul Boolean “Rezultat Boolean” și o valoare șir de caractere pentru indicatorul “String”; se oprește aplicația (când se dorește) apăsând butonul STOP.

11. Salvare linii dorite din matrice 2D:

Insert Into Array function



Cazul 0 Default asigură trecerea neafectată a celor două registre Shift

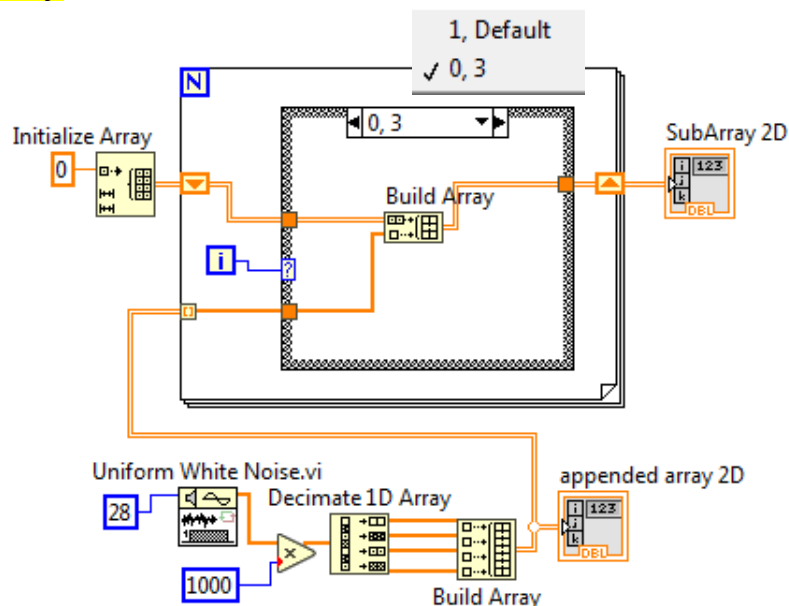
Registrul#1: noul tablou 2D(subArray)

Registrul#2: indice linie din noul tablou 2D(subArray) la care se inserează linia curentă

Varianta 2 - salvare linii dorite:

Build Array

Inițializare
reg.shift cu
matrice 2D vidă



Generare
matrice 2D
prin 'decimare'
valori din șir 1D

Sir 1D (28 valori aleatoare)

este distribuit pe 4 linii a câte 7 valori pe linie (=>2D)

Iulian Lupea, Programare, UTCN

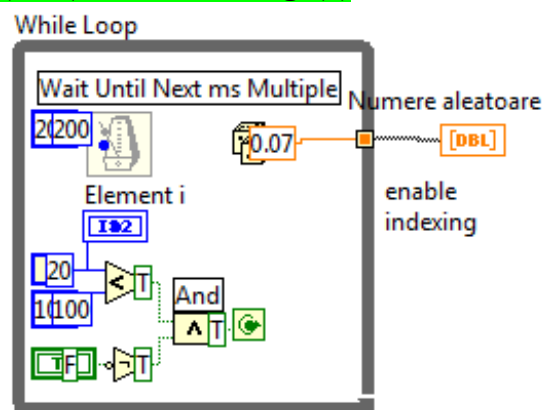
appended array 2D								SubArray 2D							
0	595.	52.7	-125	429.	-268	789.	-113	0	595.	52.7	-125	429.	-268	789.	-113
0	-545	-499	-632	-520	-576	-226	-131	0	-987	-371	780.	-579	288.	-837	-117
	148.	965.	-431	-926	-708	-382	-883		0	0	0	0	0	0	0
	-987	-371	780.	-579	288.	-837	-117								

12. Ciclul WHILE - oprire pe expresie logică

Generare de maxim 100 numere aleatoare cu oprire prin Buton de STOP

Terminalul WHILE (*continue if True*) primește valoarea expresiei:

$i < 100$ AND (not (Buton STOP logic))



Buton STOP Neapăsător → F

Buton STOP Apăsător → T

De exemplu la iterația $i=77$ și buton de Stop neapăsător se obține :

$(i < 100)$ AND Not (F) se evaluează:

$(77 < 100)$ AND T $\Rightarrow$

T AND T $\Rightarrow$ T $\Rightarrow$ ciclul While continuă

13. VARIABILA LOCALĂ

Var.Loc. este asociată unui anume *indicator sau control* din Panoul Frontal.

Cu ajutorul variabilei locale se poate **scrie în** sau **citi din** acel **indicator sau control** în cadrul instrumentului virtual curent. Var.Loc apare în diagramă dar nu apare în panoul frontal.

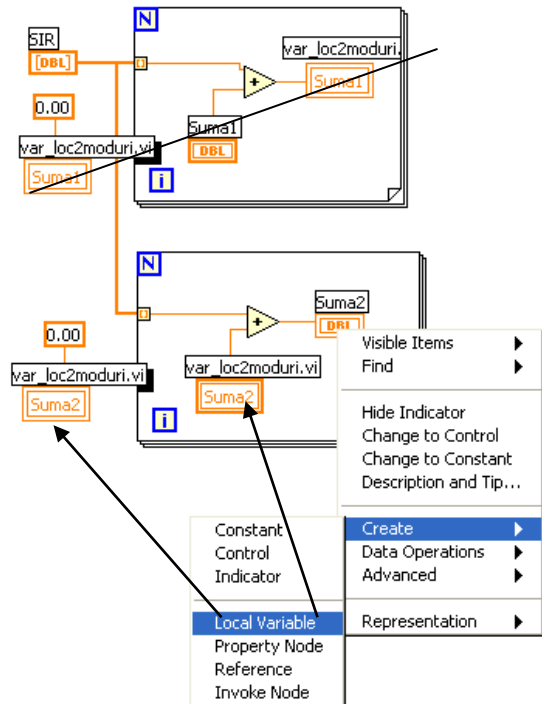
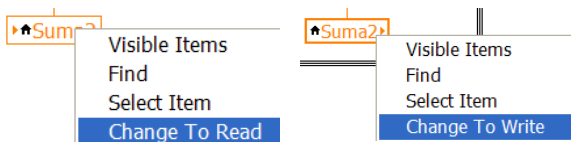
Variabila locală pentru însumarea elementelor din șir/matrice - va înlocui reg. shift

13.1. Se crează un **indicator numeric** (cu eticheta Suma2) care va memora succesiv sumele parțiale ale elementelor și suma finală.

Asociat indicatorului numeric sunt create două variabile locale numite “Suma2” – ambele fac referire la indicatorul numeric.

a) variabila locală plasată în corpul ciclului For poate **citi valoarea indicatorului** și o poate trimite la operatorul plus (**cadru bold=generează valoare în diagramă**).

b) variabila locală dinafara ciclului **primește valoarea 0 de la constantă** și totodată atribuie valoarea de inițializare zero, indicatorului (etichetat suma2).



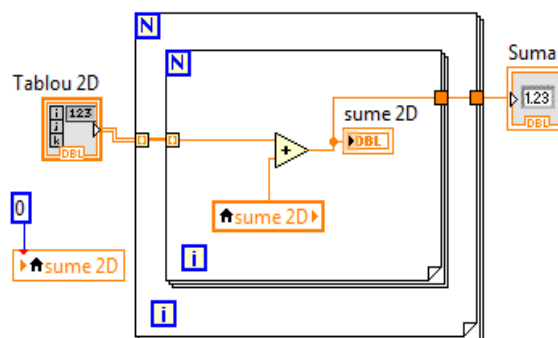
La rulare primul element din tablou pătrunde în ciclu iar prin intermediul unei variabile locale este inițializată Suma cu zero. La prima iterație se adună primul element cu zero și se transferă rezultatul spre indicatorul numeric. La a doua iterație al doilea element al șirului se adună cu suma parțială citită prin variabila locală din corpul ciclului.

4.2. Se poate rescrie (varianta sus) prin folosirea unui **control numeric Suma1** și două variabile locale **asociate** (dar valoarea este cunoscută dintr-un control în PF – nu din indicator - > diagrama tăiată).

4.3 Suma elementelor din tabloul 2D numeric

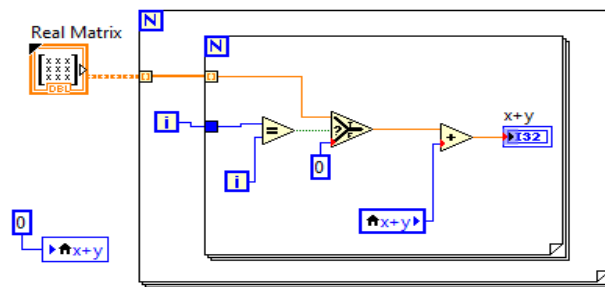
Variabila *sume 2D* indică toate sumele parțiale în timpul procesului de însumare. Variabila *Suma* indică numai suma finală a elementelor tabloului.

Tunelurile de intrare în cele două cicluri For sunt cu **cu indexare** pentru Tabloul 2D. **La tunelurile de ieșire** din cicluri **se asigură transferul ultimei valori** (Last value).



Suma tablou 2D: VarLoc asociată

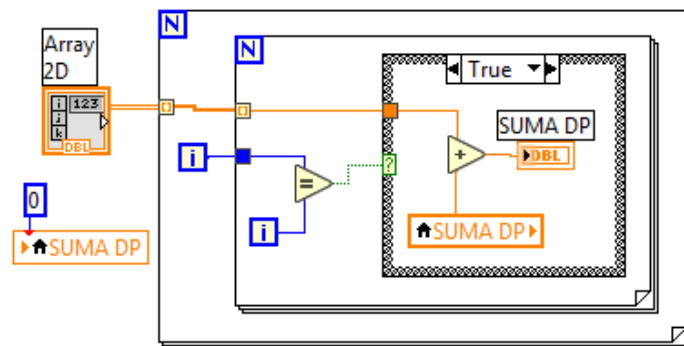
Matrice → suma pe DP (Diag.Principală)
cu **Var. locală + SELECT**



Matrice: suma DP cu var. locala + Select

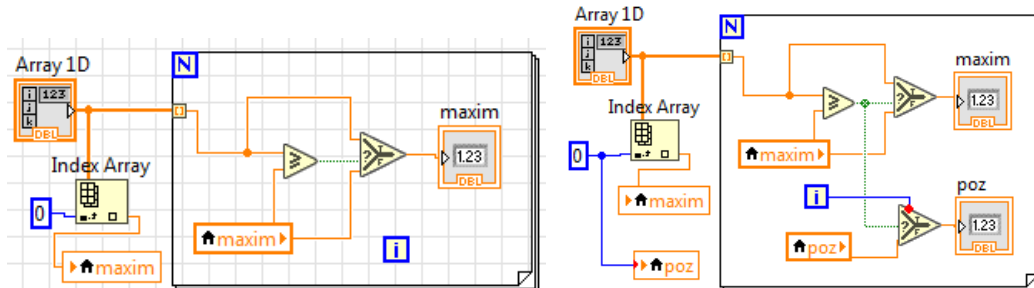
Obs: se va converti din I32 la DBL

→ Sumă DP cu **Var. locală + CASE**

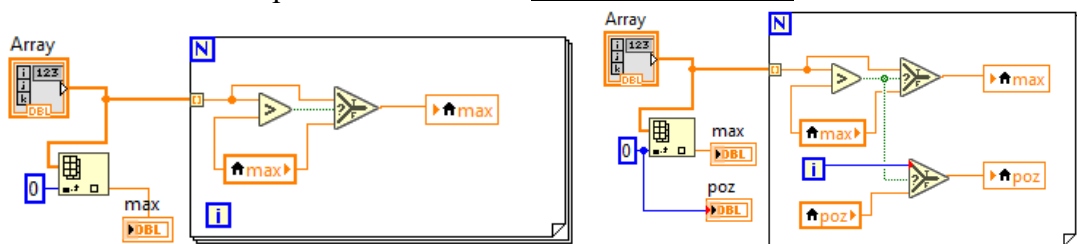


4.4. Calcul **maxim** din șir numeric - **Variabila locală + SELECT**

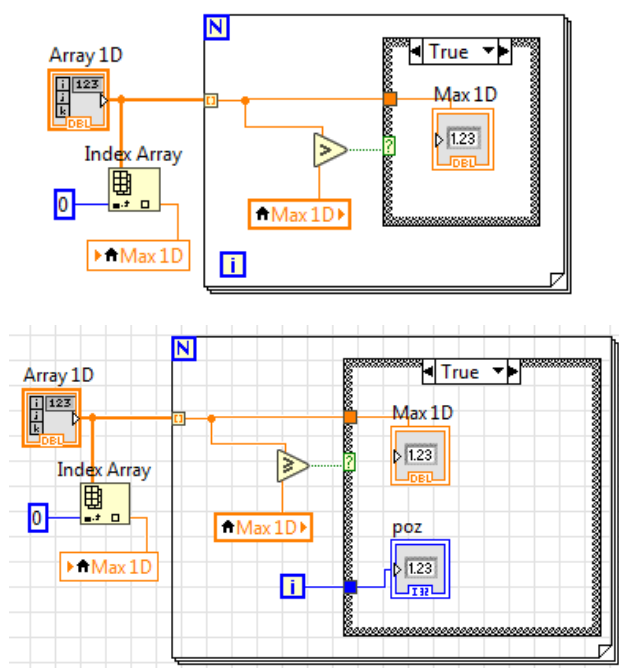
Varianta 1 = preferabil cu indicator maxim plasat în ciclu:



Varianta 2: NU se preferă cu indicator max în afara ciclului:



4.4.2. Max valoare din şir cu Var. locală + Case

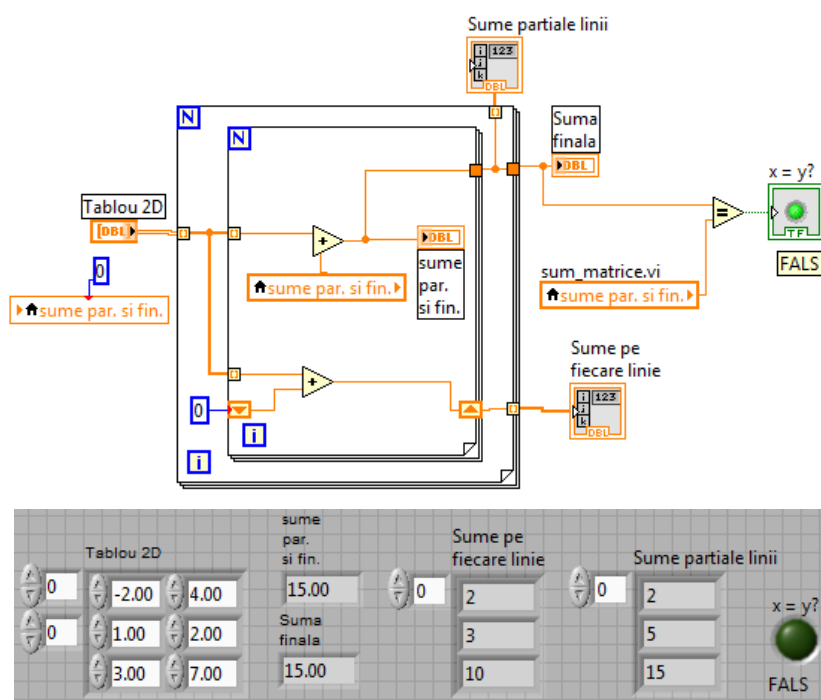


Max 1D este un indicator numeric. Prima valoare din tabloul Array 1D inițializează Max 1D prin intermediul variabilei locale asociate (scriere). În ciclul For valoarea curentă intrată în ciclu este comparată cu valoarea variabilei locale setate pe citire. La selecția cazului True este actualizat indicatorul Max 1D și indicatorul poz (poziția maximului în șir).

4.5. Particularități Variabila Locală - optional

1. În afara ciclurilor indicatorul '**sume par. si fin**' se pune pe 0 prin variabila locală asociată și se citește acel zero care apoi se conectează imediat la operatorul egal. În ciclul interior fiecare valoare din tablou 2D se adună cu variabila locală asociată indicatorului '**sume par. si fin**'. După parcurgerea unei linii în indicatorul '**sume par. si fin**' se memorează suma iar primul element din linia următoare se adună cu valoarea variabilei locale care citește indicatorul '**sume par. si fin**'.

2. **Suma finală** la ieșire din cicluri (last value, last value)



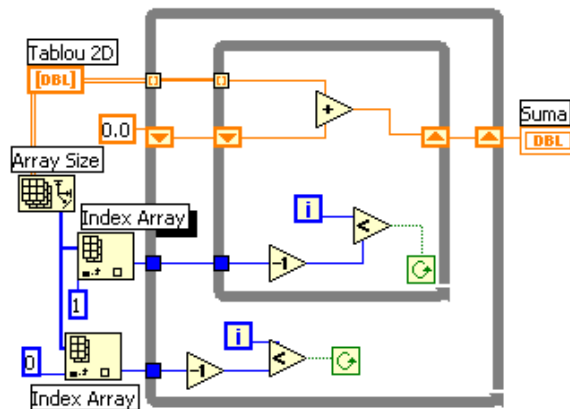
3. Indicatorul boolean indică **FALS** deoarece **VarLoc** exterioră ciclurilor este citită înainte de execuția ciclurilor

4. registrul shift pe ciclul interior: calculează sumele pe fiecare linie

Reg shift este pus pe 0 după ce realizează suma pe fiecare linie.

Rezultă un tablou de sume pe fiecare linie.

4.6. Suma elementelor din matrice prin 2 cicluri While (cu regiștri pentru comparație)

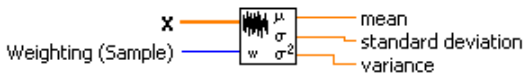


Temă: modificați diagrama folosind variabila locală

14. MĂRIMI STATISTICE aplicate unui ȘIR DE VALORI NUMERICE: media aritmetică, media pătratelor, varianța și deviația standard

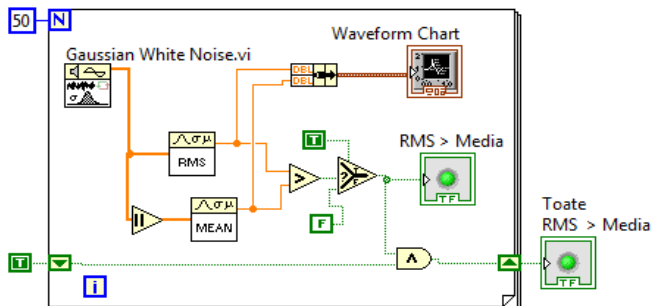
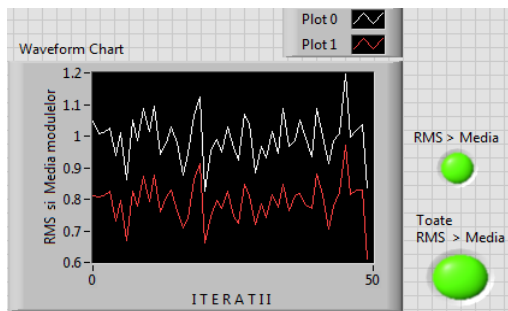
1. Funcții statistice elementare (predefinite)

$\bar{x}$ = <u>media</u> aritm. a semnalului x_k , $k=0, \dots, n-1$, dat prin valori discrete, pe o înregistrare (notație alternativă: μ).	$\bar{x} = \frac{1}{n} \sum_{k=1}^n x_k$	
$\overline{x^2}$ = <u>media pătratelor</u> (mean square value), este media pătratelor valorilor semnalului x_k .	$\overline{x^2} = \frac{1}{n} \sum_{k=1}^n x_k^2$	
x_{RMS} = <u>rădăcina din media pătratelor</u> (root mean square) sau valoarea eficace.	$x_{RMS} = \sqrt{\overline{x^2}}$	
σ^2 = <u>varianța</u> descrie fluctuația valorilor față de medie, fiind media aplicată pătratelor abaterilor de la medie.	$\sigma^2 = \frac{1}{n} \sum_{k=1}^n (x_k - \bar{x})^2$	
*abaterea de la medie a valorii indice k este de forma $x_k - \bar{x}$ și poate avea valori pozitive sau negative *varianța observă pătratul abaterilor. σ = <u>deviația standard</u> sau dispersia	$\sigma = \sqrt{\frac{1}{n} \sum_{k=1}^n (x_k - \bar{x})^2}$	

σ = radical ordinul 2 din varianța	
Standard Deviation and Variance.vi Input: șirului X (intrare) Returnează: media, deviația standard și varianța șirului X	 Standard Deviation and Variance

2. Comparație între RMS și Media Modulelor valorilor

Reg. Shift de tip LOGIC



În cadrul ciclului se compară valorile RMS și media modulelor celor 128 valori generate de GWN. $RMS > Media$ modulelor deci se luminează ledul. Dacă toate comparațiile sunt True atunci în registrul de transfer se menține valoarea True și se afișează True la ieșire din ciclu.

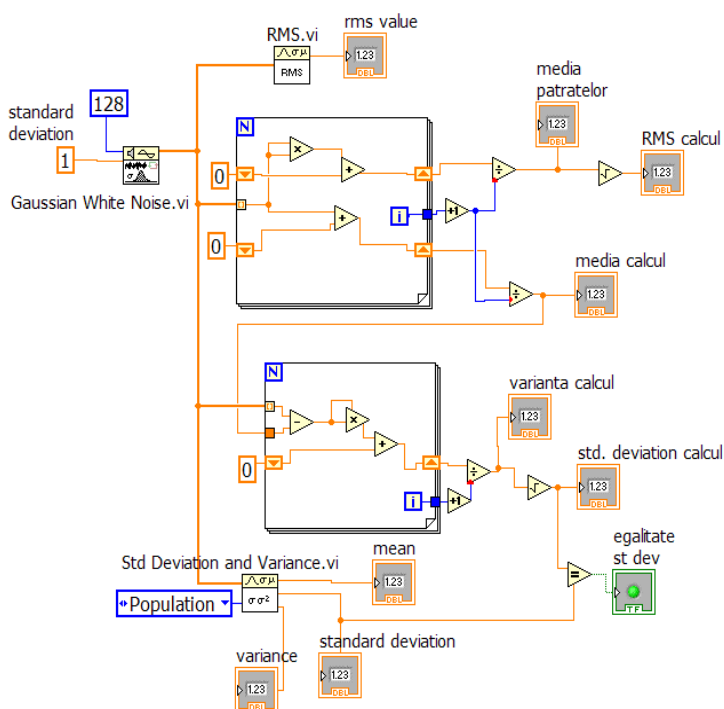
3. Calcul mărimi statistice - diagrama

registrul #1:
suma pătratelor
Parcursere#1 a șirului (for #1)

registrul #2: suma șir

Parcursere#2 a șirului (for #2)

registrul #3:
suma abateri la pătrat



Calcul: RMS, media, varianța, dev. standard

4. Varianța este egală cu **media pătratelor** din care se scade **pătratul mediei** valorilor.

$$\sigma^2 = \frac{1}{n} \sum_{k=1}^n (x_k - \mu)^2 = \frac{1}{n} \sum_{k=1}^n x_k^2 - 2\mu \frac{1}{n} \sum_{k=1}^n x_k + \frac{1}{n} \sum_{k=1}^n \mu^2 = \frac{1}{n} \sum_{k=1}^n x_k^2 - \mu^2$$

$$\sigma^2 = \overline{x^2} - \bar{x}^2$$

5. **Temă:** Pe baza relației deduse se va calcula **varianța** folosind un singur ciclu.

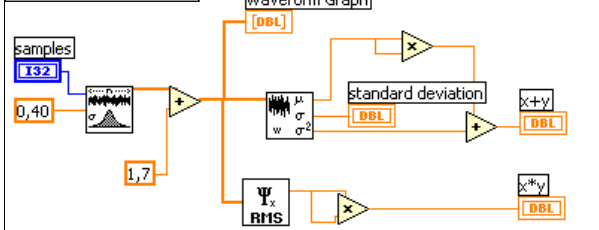
Să se verifice
relația:

$$\sigma^2 = \overline{x^2} - \bar{x}^2$$

Obs: media Gwn (Gaussian White Noise.vi) fiind 0 se introduce un offset de 1.7 pentru a evita cazul particular.

Intrarea 0.40 este deviația standard impusă șirului generat de Gwn.

$$\overline{x^2} = \bar{x}^2 + \sigma^2$$



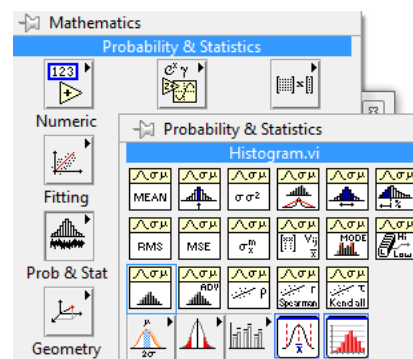
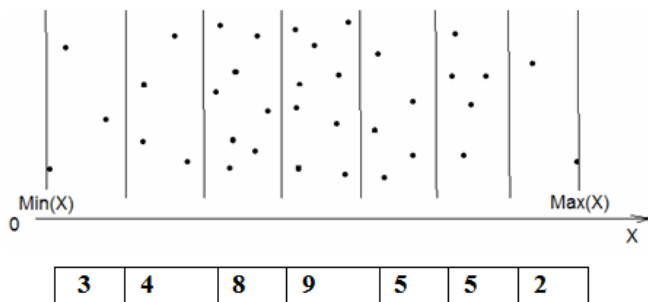
Standard Deviation and Variance.vi

6. Funcția **Histogram.vi**

Histograma $h(x)$ indică câte valori din șirul de intrare (X) sunt în fiecare din cele **m intervale (intervals)** egale.

$$\text{delta}_x = \frac{\max(X) - \min(X)}{m}$$

$$\text{center}[i] = \min(X) + \text{delta}_x/2 + i * \text{delta}_x, \quad i=0,1,2,\dots,m-1;$$



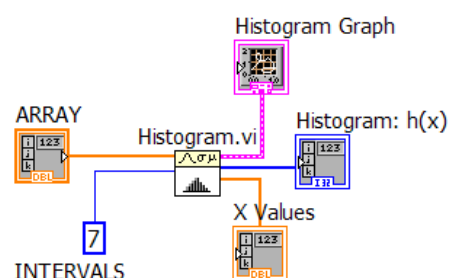
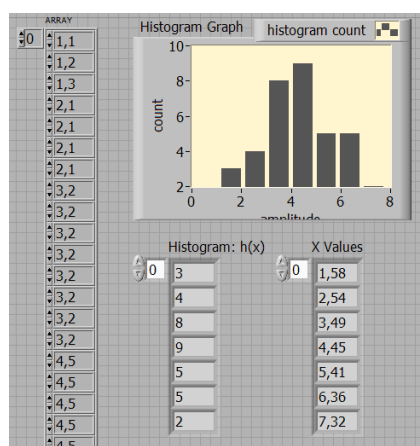
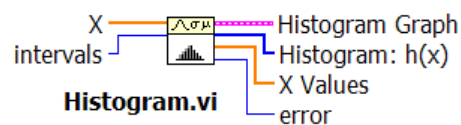
Număr valori în fiecare interval; **m=7**, $i=0,\dots,6$

Ieșiri din funcție:

Histogram Graph (structură de 2 tablouri 1D) se afișează histograma în grafic de bare

Histogram: $h(x)$ = șirul 1D al numerelor de valori din fiecare interval

X Values: $\text{center}[i]$ = șirul 1D al **valorilor centrale** ale intervalelor



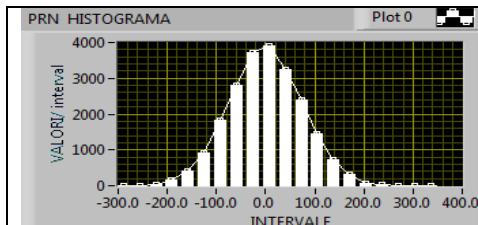
Obs: cele 7 bare indică numărul de valori din cele 7 intervale.

Tema #2. Să se calculeze histograma unui șir de valori folosind **programul propriu** pe baza relațiilor de mai sus.

13. Surse de șiruri numerice ALEATOARE:

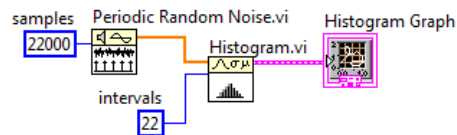
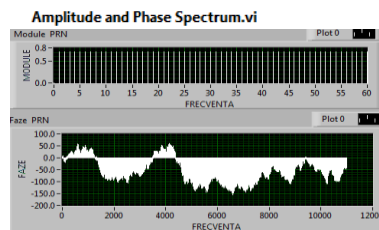
1. <i>Gaussian White Noise.vi</i>	2. <i>Uniform White Noise.vi</i>
GWN generează un tablou 1D de n valori (samples) aleatoare. Densitatea de probabilitate a amplitudinii=șantioanele cu distribuție Gauss (clopot) și împrăștierea valorilor în jurul mediei dictată de intrarea standard deviation ($\sigma = 1$ implicit); medie zero, corelare zero.	UWN ideal are densitatea de putere spectrală PSD constantă pe tot domeniul de frecvență (sunt necesare multe medieri). Valorile semnalului au aceeași probabilitate de apariție. Media valorilor este zero.
Samples: 22000 20 intervale; media=0	Samples: 22000 20 intervale; media=0

3. <i>Periodic Random Noise.vi</i>	4. <i>Binary MLS.vi</i>
PRN este o <u>suma de sinusoid</u>e, fiecare sinusoidă are <u>număr întreg de cicluri</u>, de aceeași amplitudine dar faze aleatoare => este random periodic deci numai o înregistrare este necesară pentru testare. Media valorilor este zero.	Funcția <u>Maximum Length Sequence</u> generează o succesiune aleatoare de 0 și 1 cu <u>perioada $2^n - 1$</u> unde n=polynomial order



Samples: 22000
20 intervale;
media=0

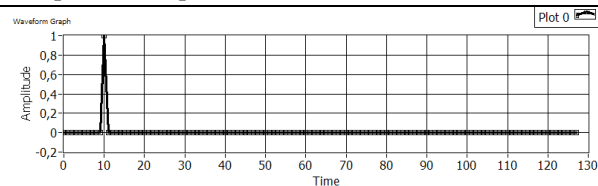
Signal (V) — Amp Spectrum Mag (Vrms)
unwrap phase (T) — Amp Spectrum Phase (radians)
dt — df



5. Impulse Pattern.vi

samples — Impulse Pattern
amplitude — error
delay —

Eșantioanele=0 mai puțin unul,
Delay=10 (indicele pulsului)
Amplitudinea pulsului

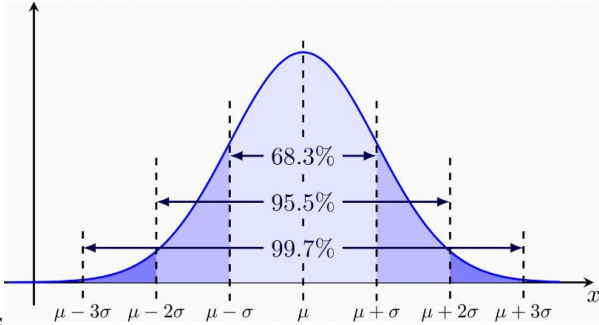
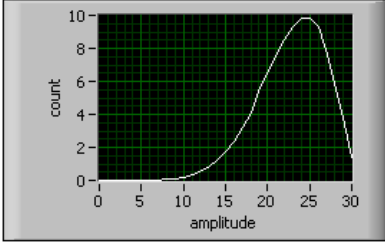
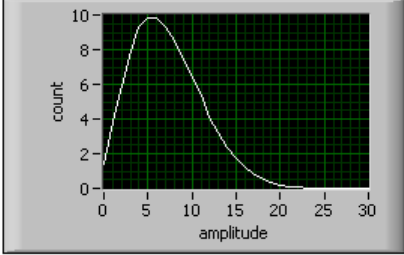


14. Calcul **Skewness** și **Kurtosis** pentru un șir $X_t(i)$ de valori numerice

$$skewness = \frac{\frac{1}{n} \sum_{i=0}^{n-1} (X_t(i) - \mu)^3}{\sigma^3}$$

$$kurtosis = \frac{\frac{1}{n} \sum_{i=0}^{n-1} (X_t(i) - \mu)^4}{\sigma^4}$$

Skewness = măsura **simetriei distribuției** șirului de valori (time series).
(distribuția de probabilitate discretă poate fi reprezentată printr-o histogramă)

 Distribuția normală Gauss	
Skewness $<0 \Rightarrow$ left skewness The following front panel image shows negative skewness. 	Skewness $>0 \Rightarrow$ right skewness The following front panel image shows positive skewness. 
Kurtosis = măsura ascuțimii / planității distribuției șirului de valori (time series)	
Kurtosis ≈ 3 indică vârf normal al distribuției Kurtosis < 3 indică a distribuție mai aplatază decât distribuția normală. Kurtosis > 3 indică a distribuție mai ascuțită decât distribuția normală	