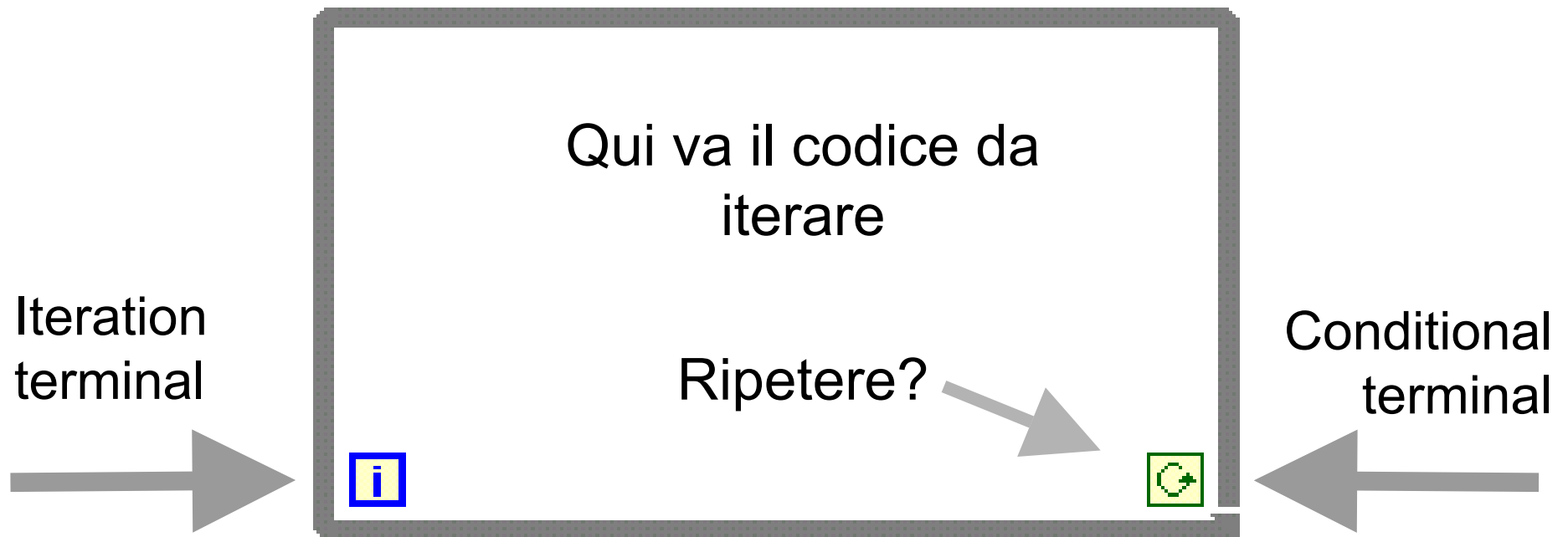


Corso di Sistemi di Misura Distribuiti

QUARTA LEZIONE:

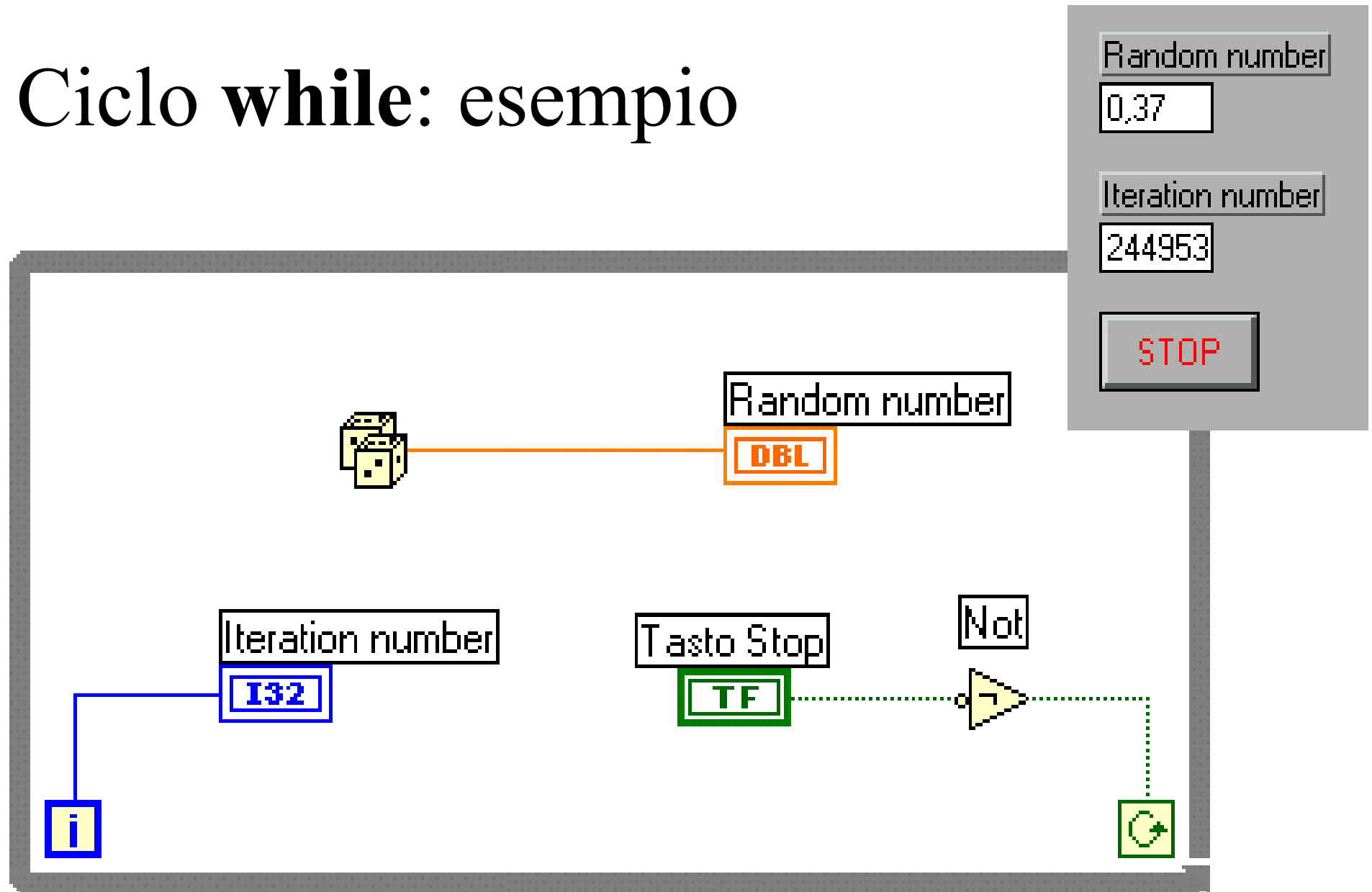
RICHIAMI DI LABVIEW

Il ciclo **while**



- „ Il ciclo termina quando si fornisce FALSE al conditional terminal
- „ Il codice è eseguito almeno una volta
- „ L'iteration terminal dà il numero di iterazioni raggiunto

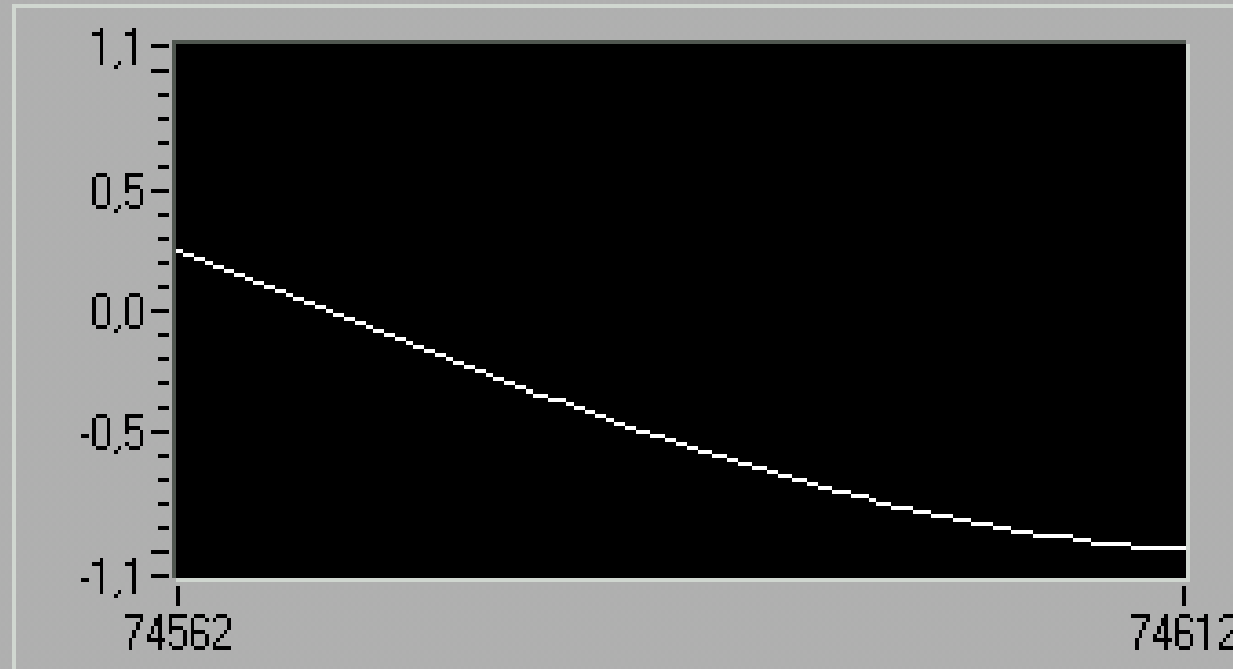
Ciclo while: esempio



Waveform chart

Label

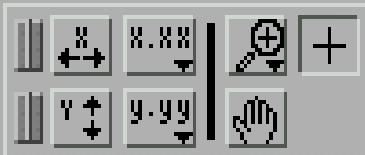
Waveform chart



Plot 0

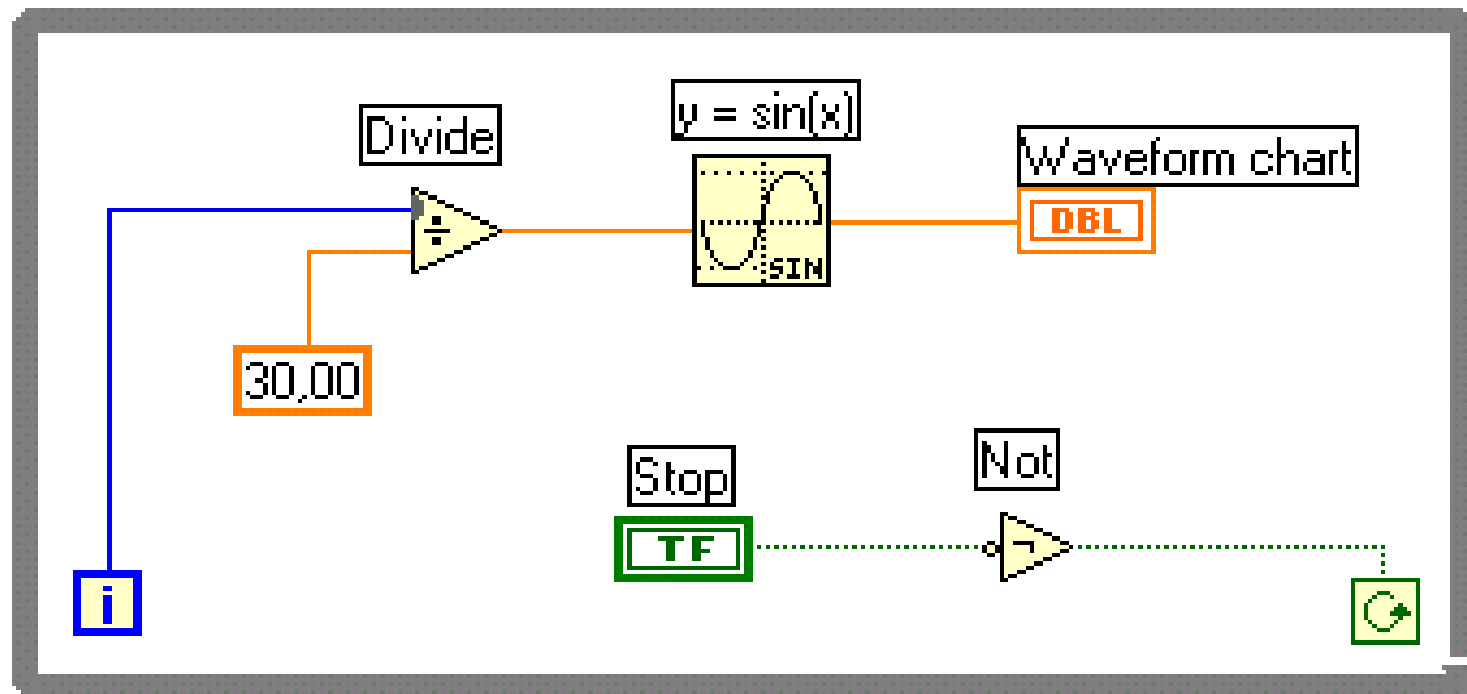


Legenda



Palette

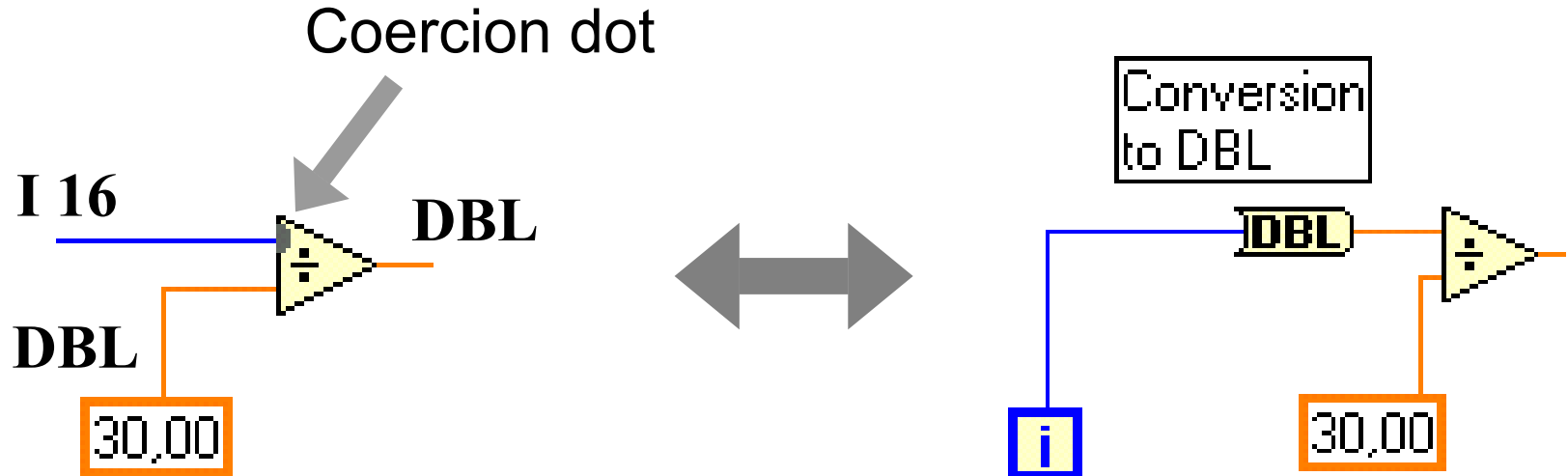
Visualizzazione con waveform chart



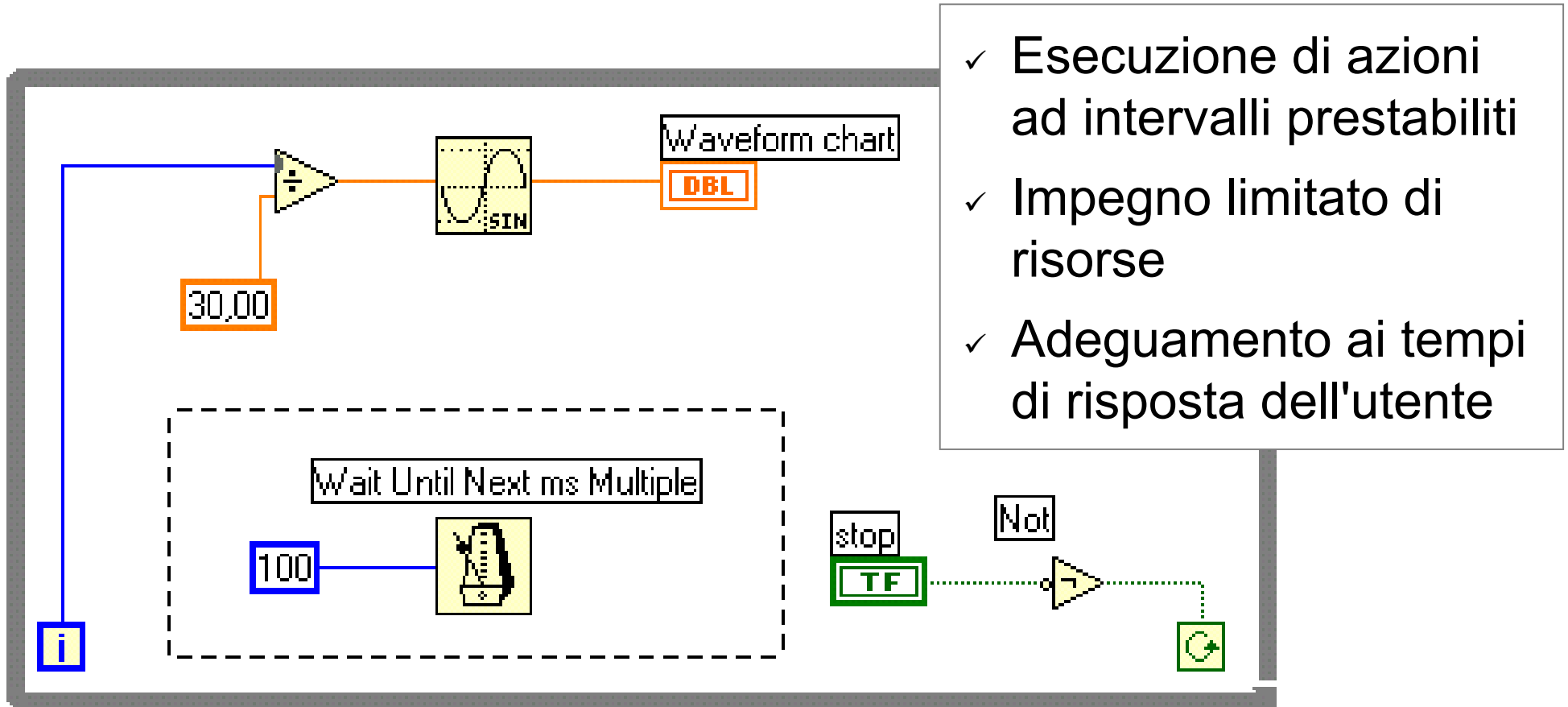
- Ad ogni iterazione il waveform chart riceve uno scalare

Tipi numerici e conversioni

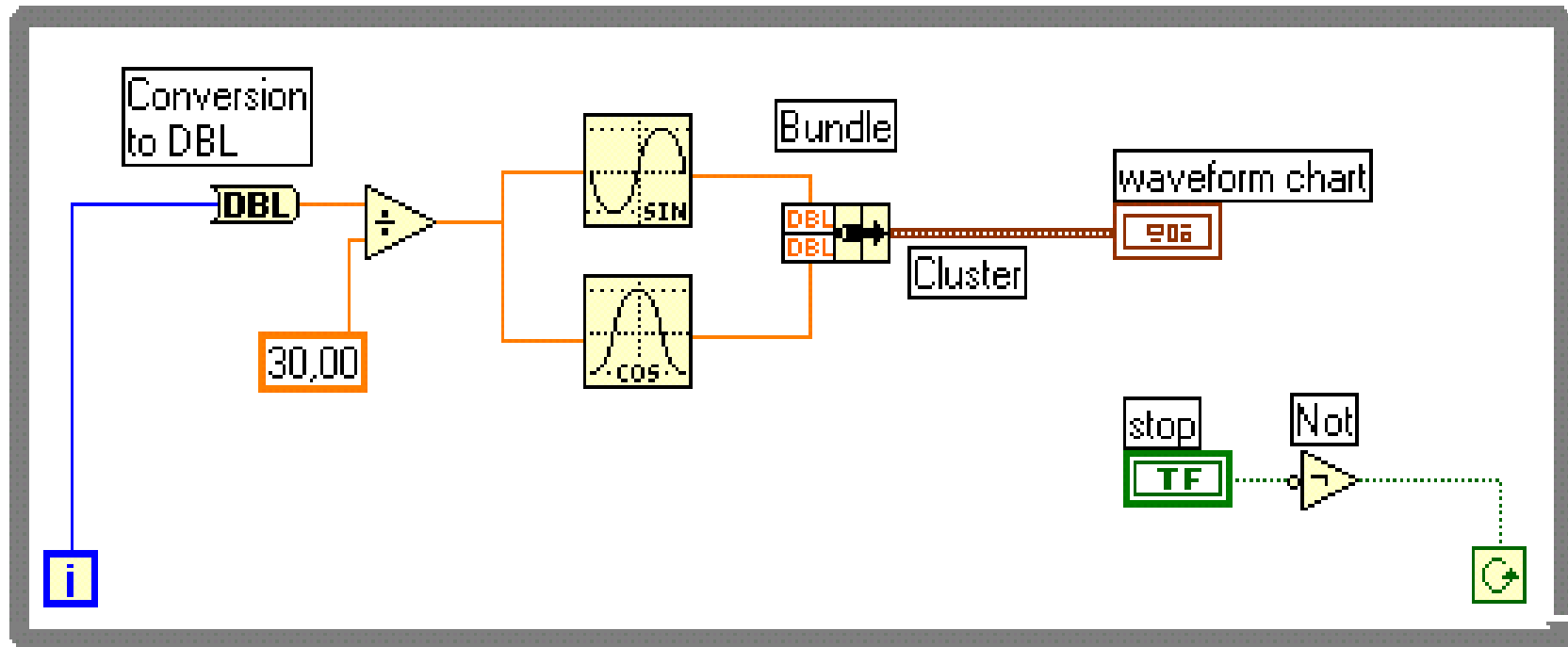
- „ Interi con segno: I8, I16, I32
- „ Interi senza segno: U8, U16, U32
- „ In virgola mobile: SGL (32), DBL (64), EXT
- „ Complessi



Ciclo while temporizzato



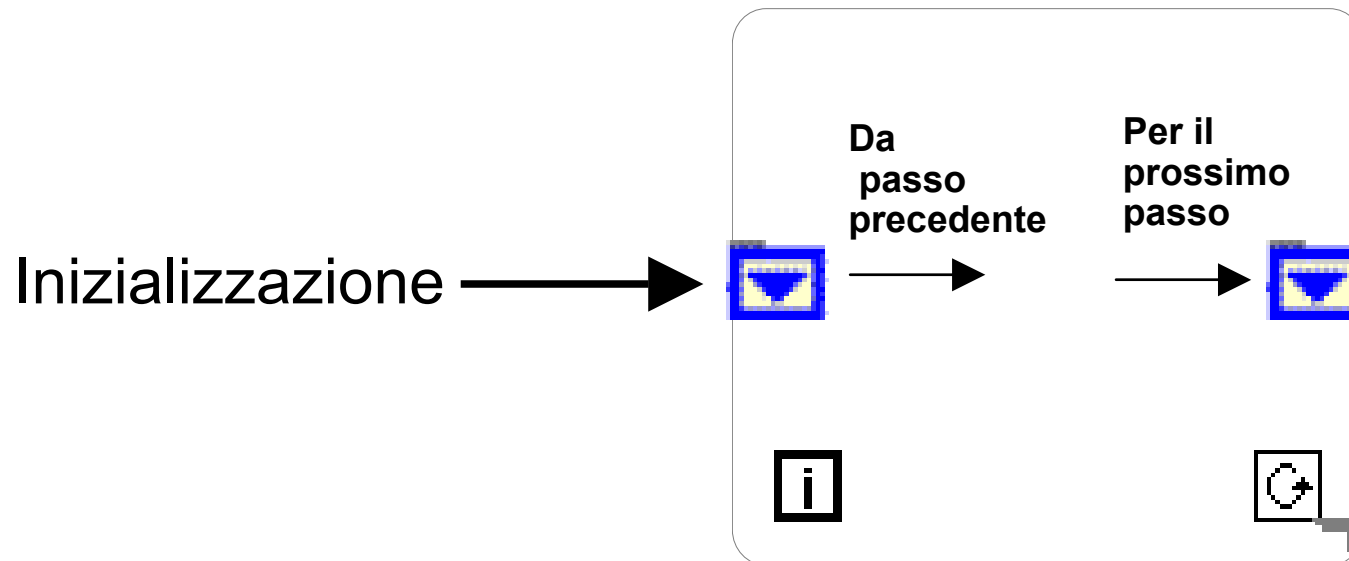
Grafici *multiple – plot*



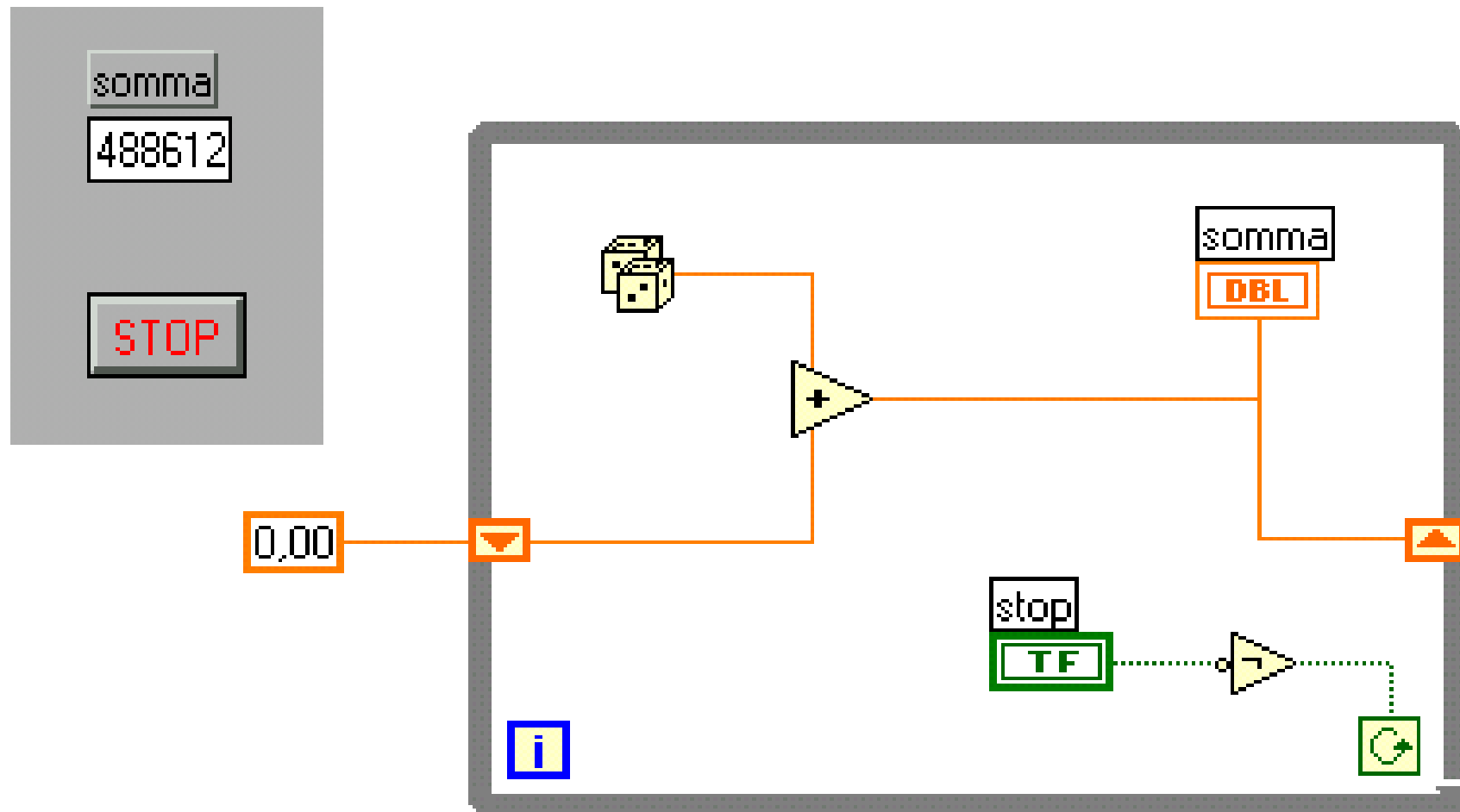
- **Bundle** (nella palette Functions >> Cluster): assembla i suoi ingressi in un unico *cluster*
- Un **cluster** è una struttura che raccoglie più elementi in un nuovo tipo di dato

Shift Registers

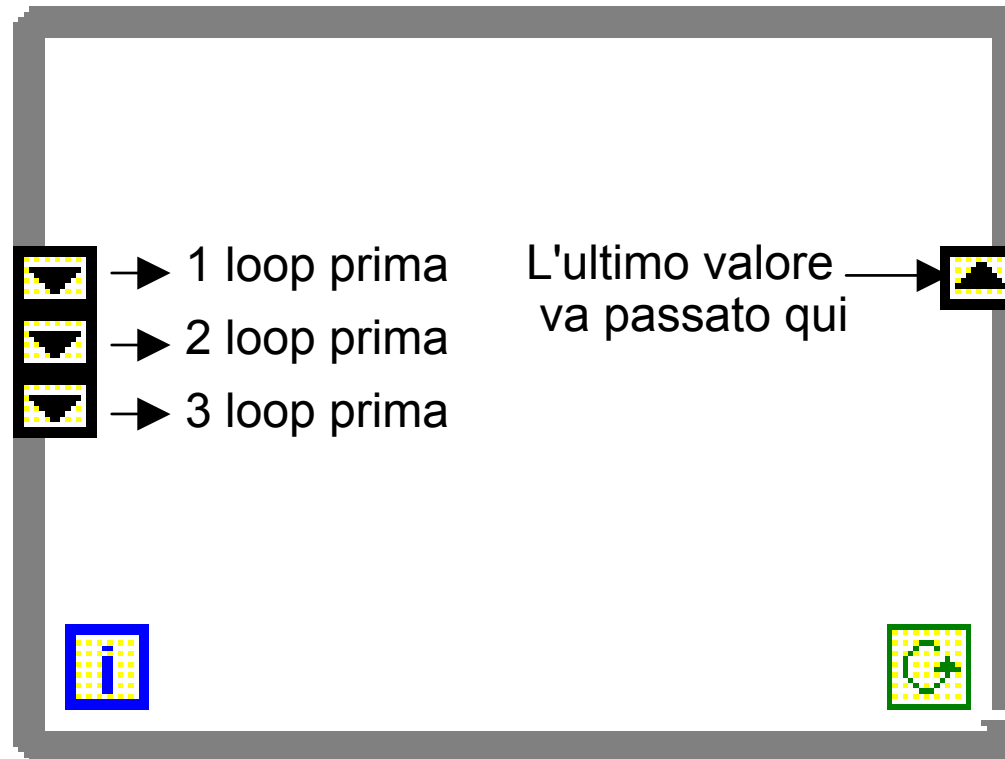
- Trasferiscono dati da una iterazione alla successiva
- Si inseriscono con un *right-click* sul bordo del *loop*
- Il terminale destro conserva il dato alla fine della iterazione
- Il terminale sinistro fornisce il dato alla iterazione successiva



Shift Registers: esempio

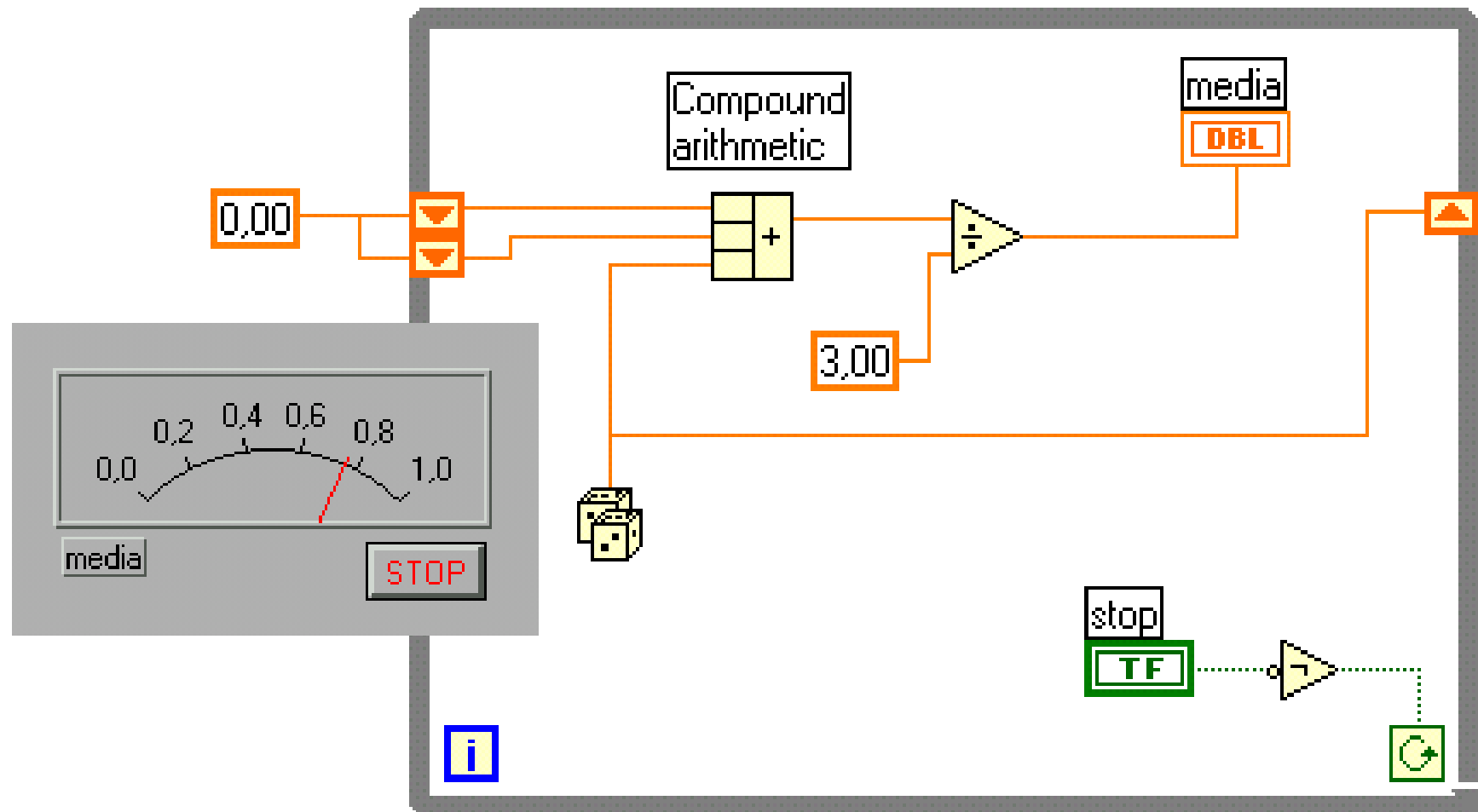


Shift Registers



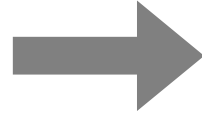
Si possono aggiungere altri *shift register* per ottenere valori di iterazioni precedenti

Esempio: *Running average*



Il ciclo for

Numero di iterazioni
da eseguire



N

Numero corrente di
iterazioni



i

Esegue il codice al suo interno un numero
prestabilito di volte

Arrays

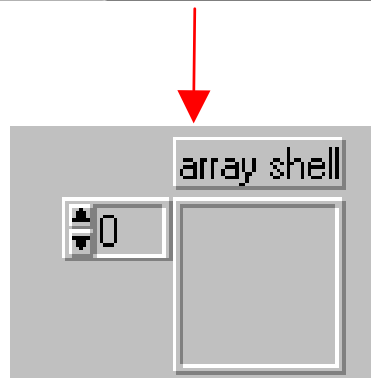
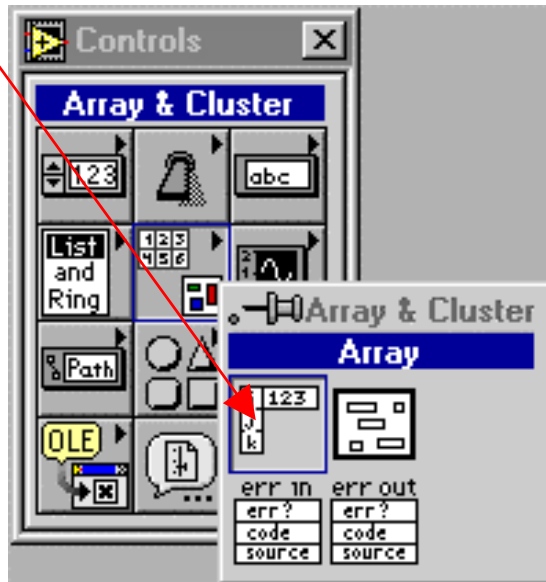
- „ Collezioni di elementi (dati) dello stesso tipo
- „ Una o più dimensioni, fino a 2^{31} elementi per dimensione
- „ Si accede agli elementi con un indice per dimensione
- „ Il primo elemento ha sempre indice 0

	Indice	0	1	2	3	4	5	6	7	8	9
array di 10 elementi		1.2	3.2	8.2	8.0	4.8	5.1	6.0	1.0	2.5	1.7

		0	1	2	3	4	5	6
Array	0							
bidimensionale	1							
	2							
	3							
	4							

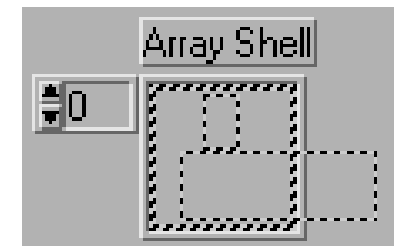
Controlli e indicatori per gli array

1. Selezionare **Array** dalla palette dei controlli



Procedura analoga per le costanti array

2. Inserire un controllo del tipo desiderato nella **Array Shell**



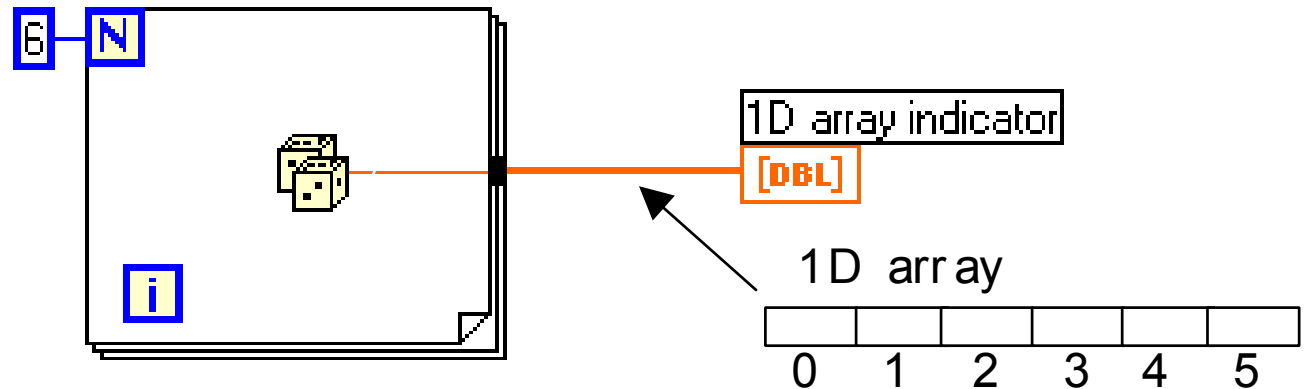
Add Dimension per 2D arrays



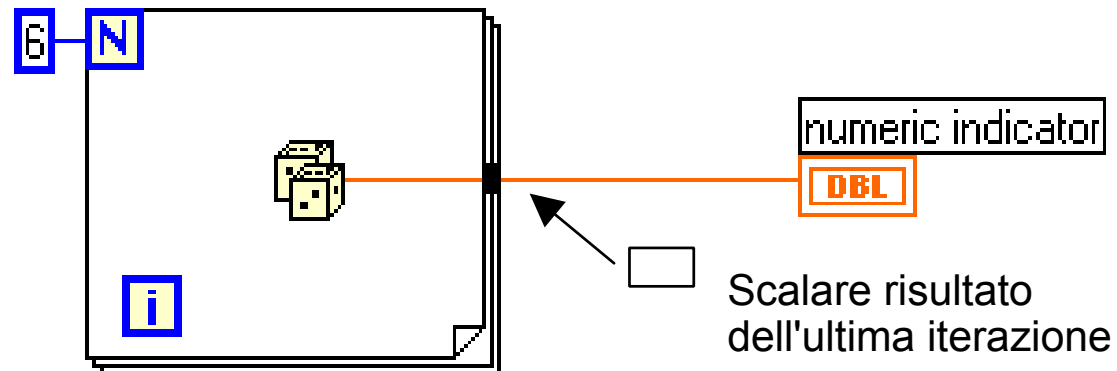
Autoindexing

Il loop accumula l'array al suo margine

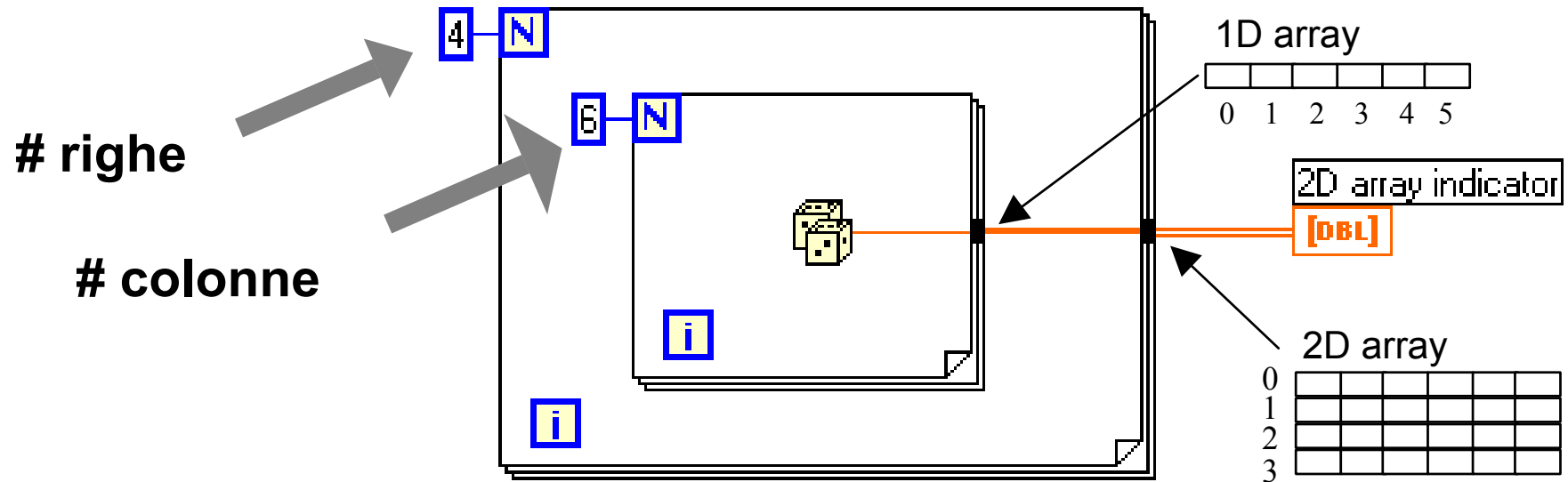
Auto-Indexing
abilitato
(Default nel For)



Auto-Indexing
disabilitato
(Default nel While)



Creazione di array 2D

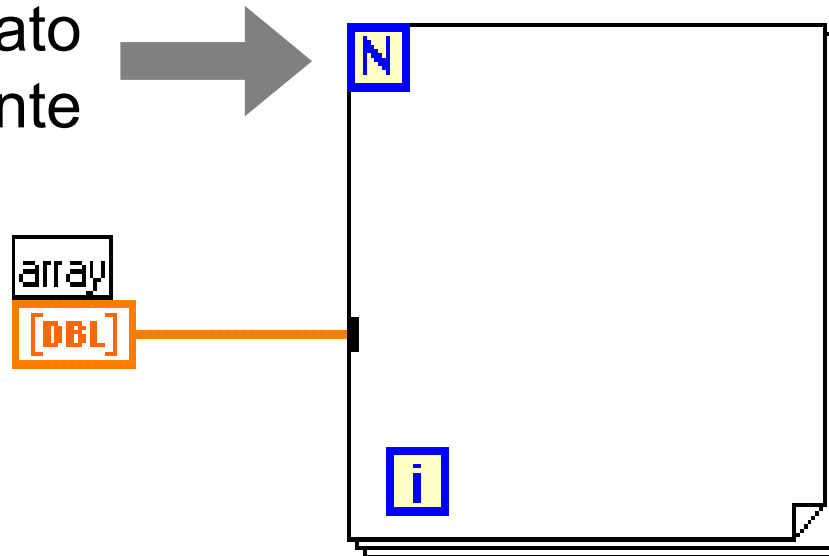


Il loop interno crea gli elementi di ogni riga

Il loop esterno sovrappone le righe

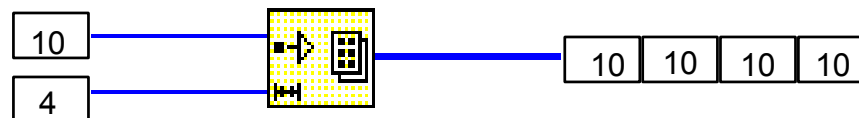
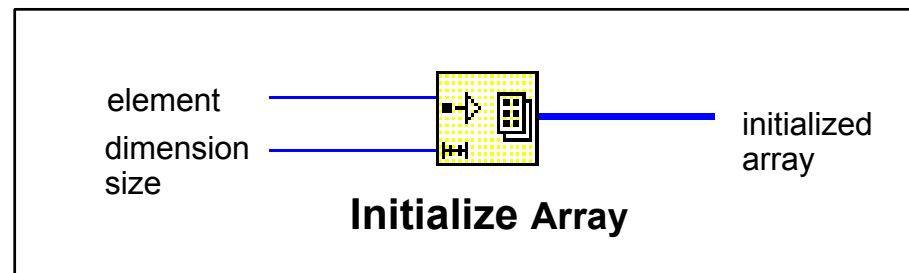
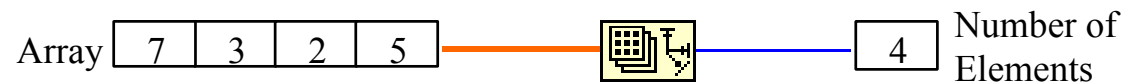
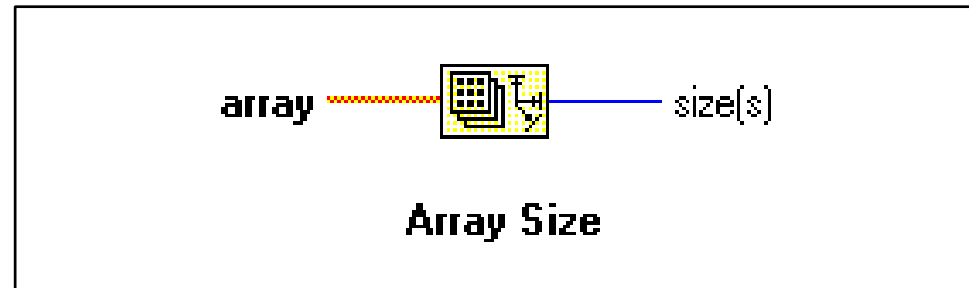
Autoindexing e lettura di array

Non deve essere impostato
direttamente



Il ciclo viene eseguito per un numero di volte pari
al numero di elementi dell'array collegato

Funzioni per gli array (1)



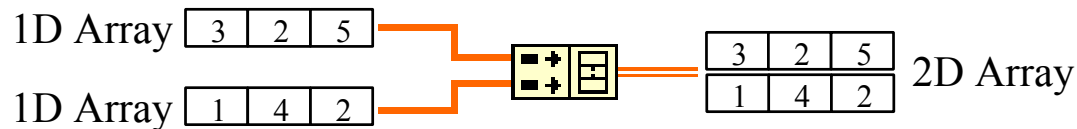
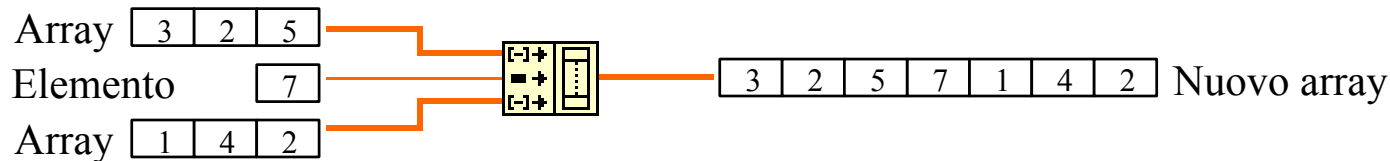
Funzioni per gli array (2)



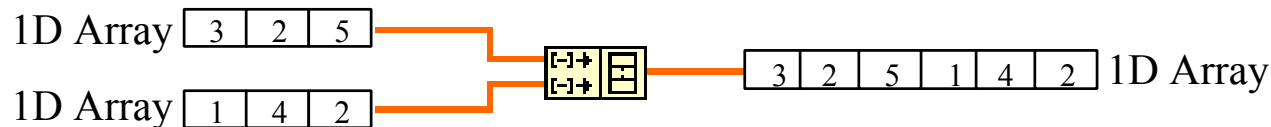
Build Array

La funzione **Build Array**

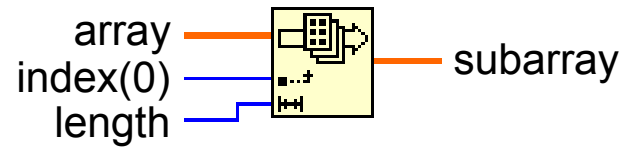
☐ *resizable*



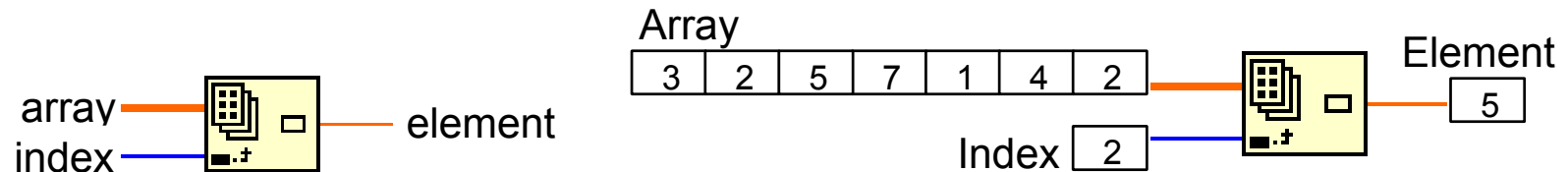
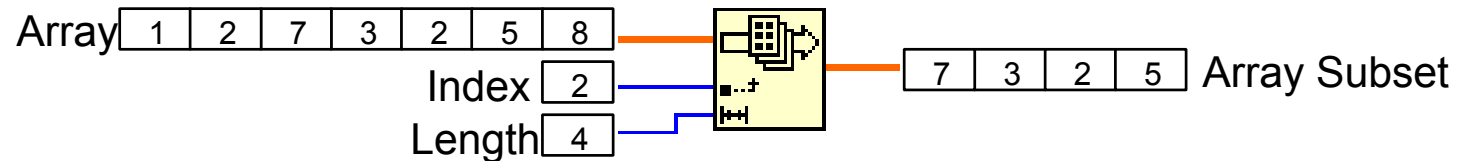
Dal menu di contesto:
“change to element”
“change to array”



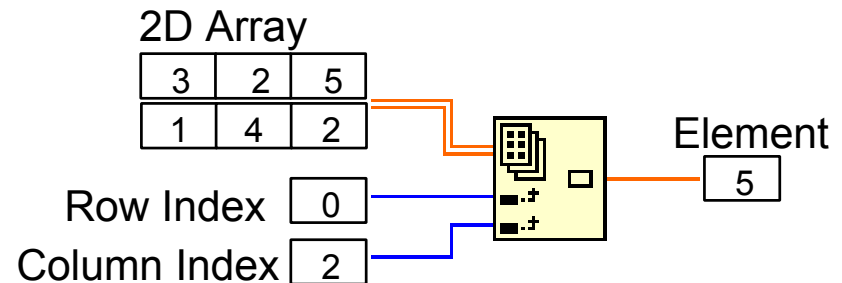
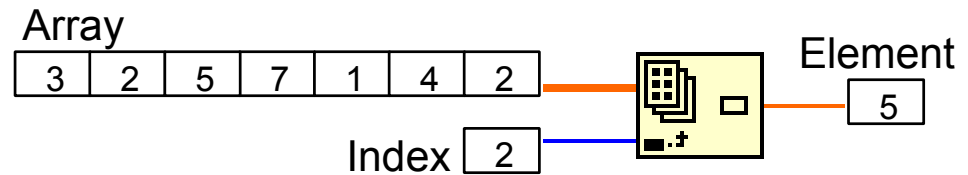
Funzioni per gli array (3)



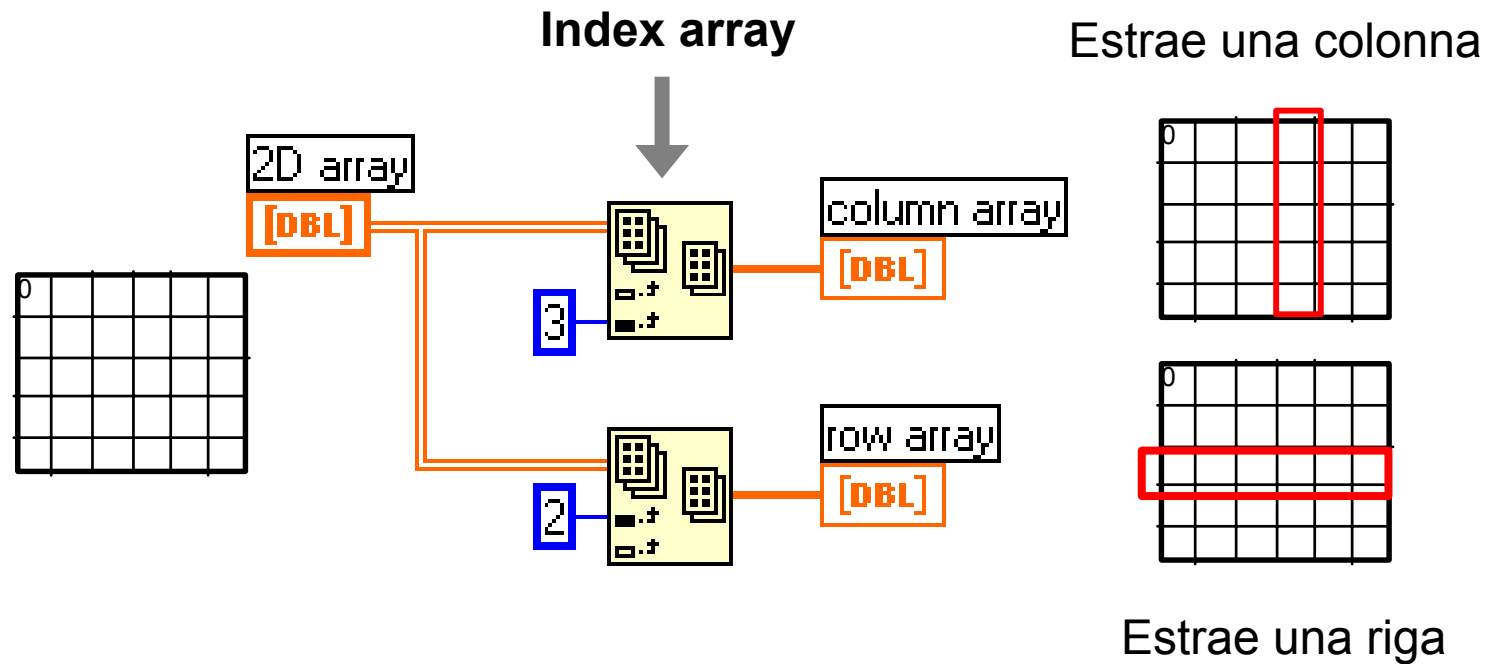
Array Subset



Index Array



Funzioni per gli array (4)

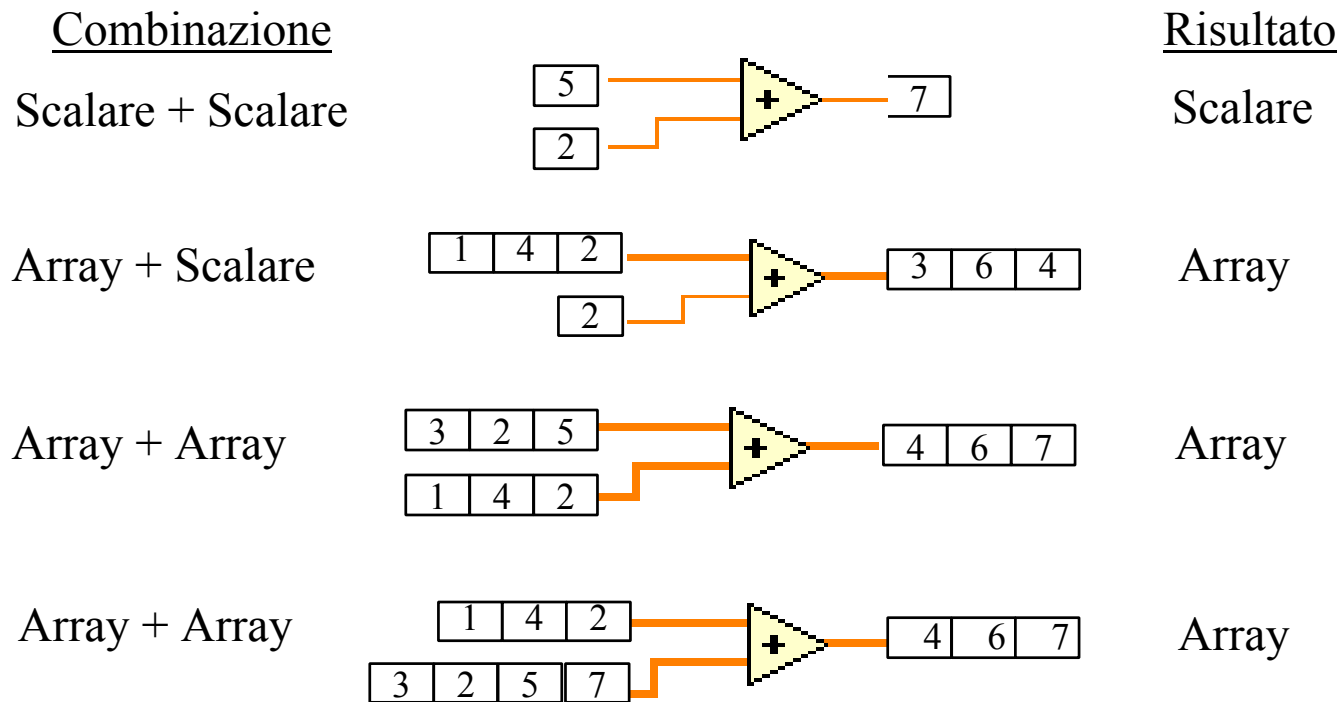


Dal menu di contesto della **Index Array**:

- **Add dimension**
- **Disable input**

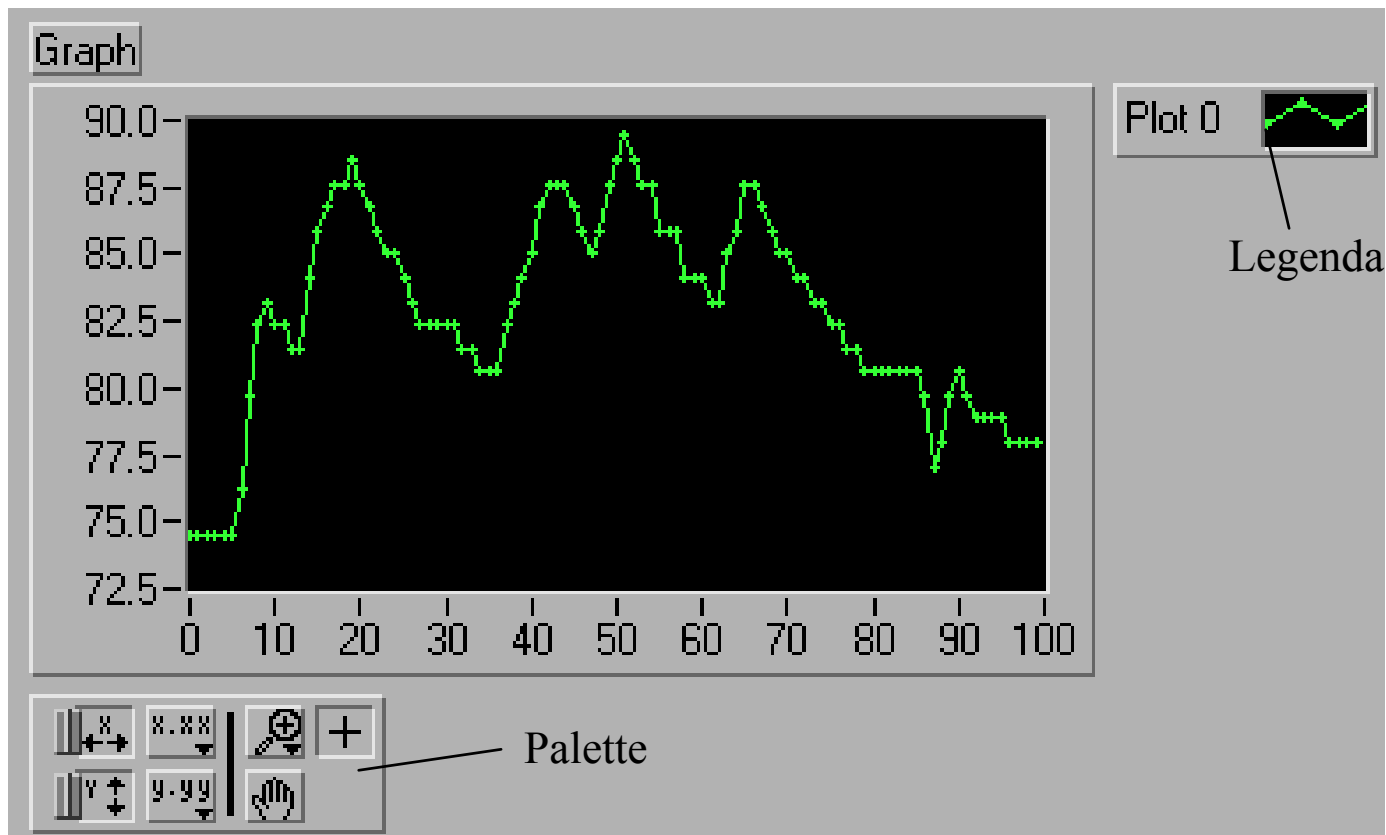
Polimorfismo

- Ingressi delle funzioni di tipo diverso
- Comportamento dipendente dal tipo degli ingressi
- Tutte le funzioni aritmetiche sono polimorfiche

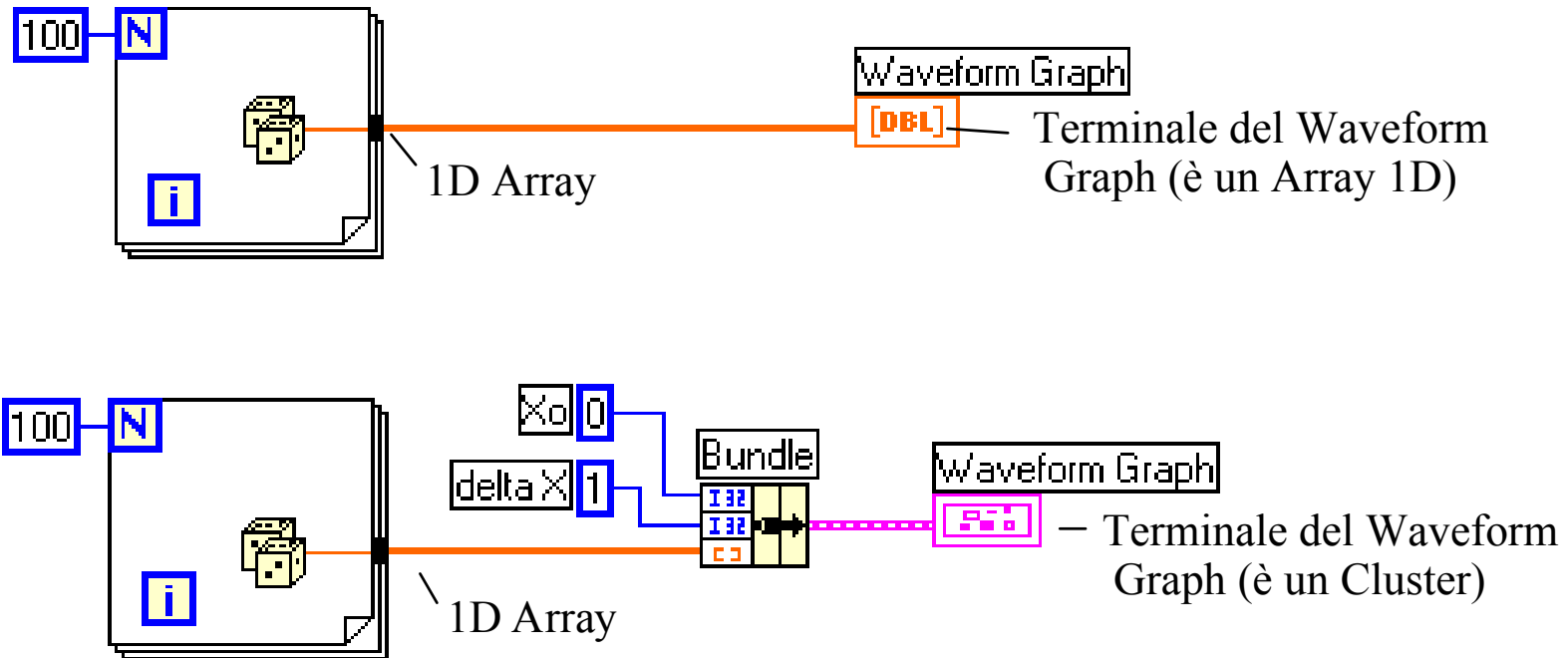


Waveform Graph

Tracciano un array di valori numerici rispetto all'indice

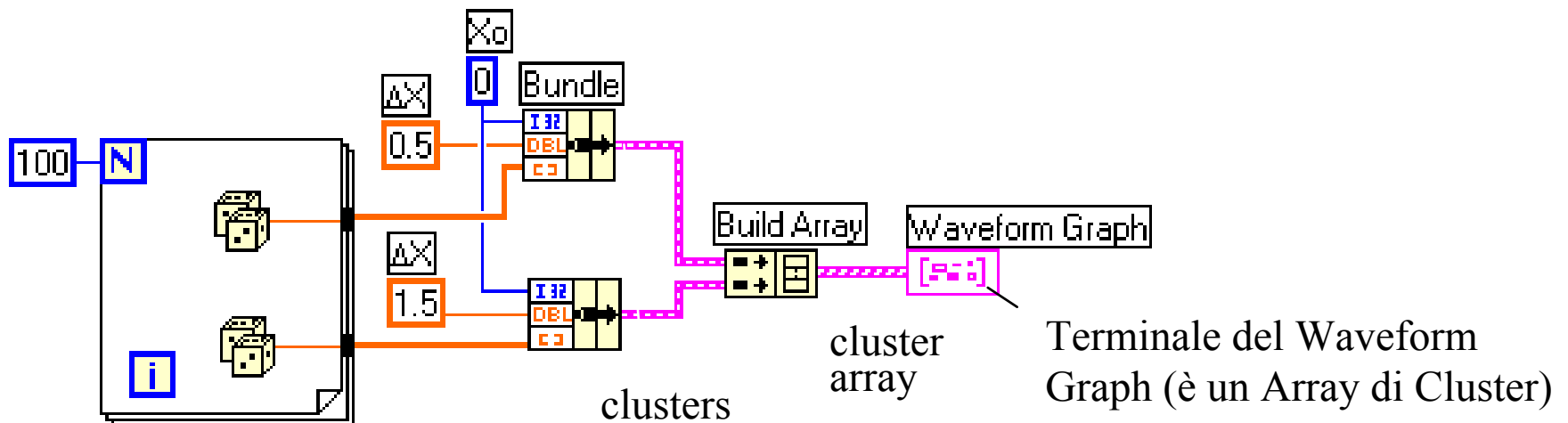
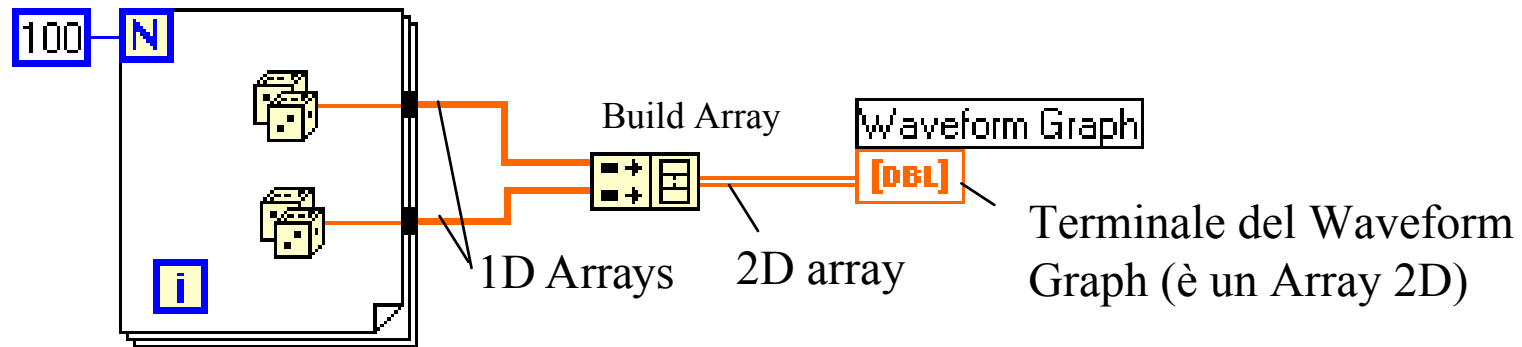


Single-Plot Waveform Graph



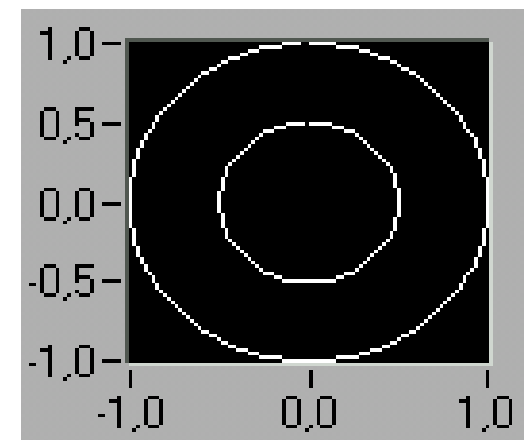
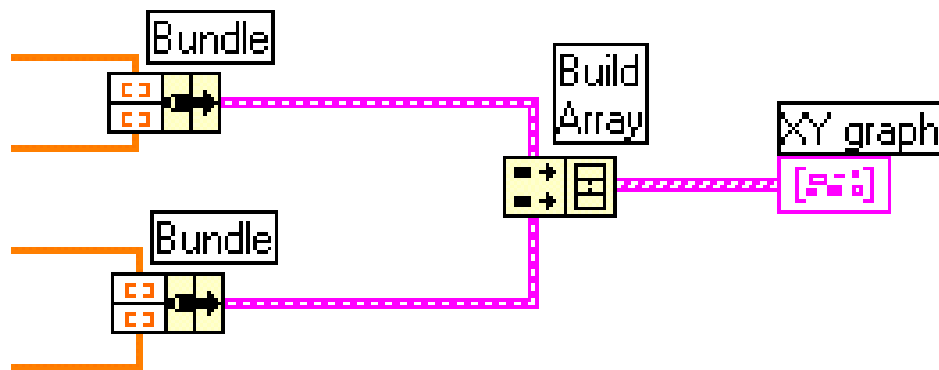
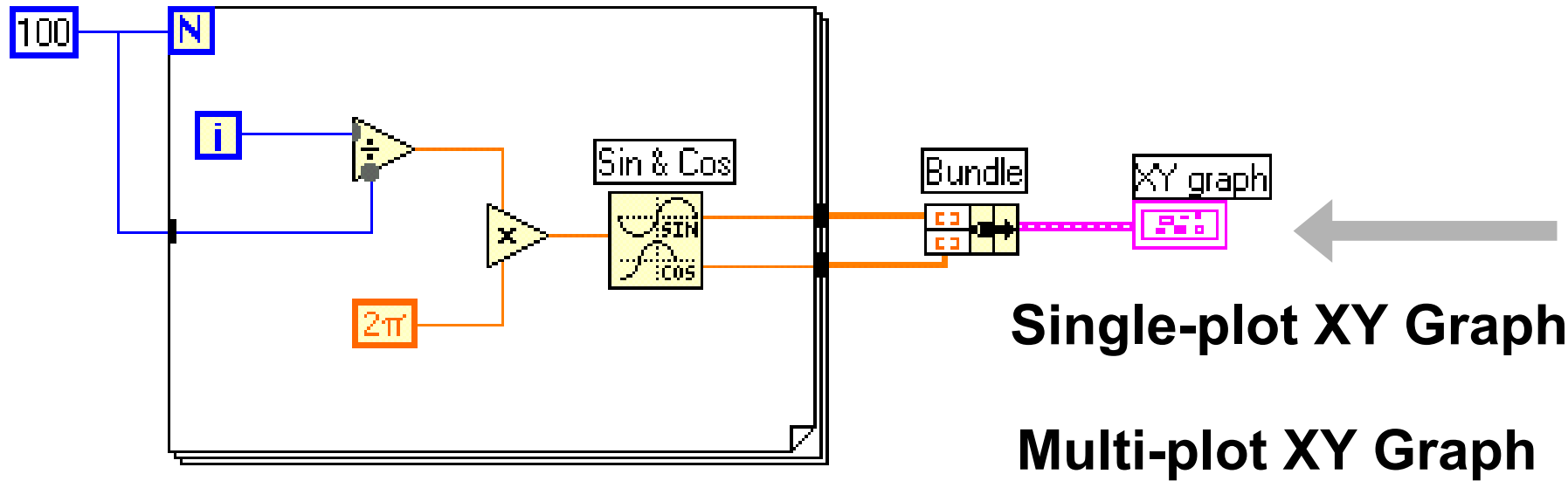
- X_0 è il valore iniziale per le ascisse
- $\text{delta}X$ è la distanza orizzontale tra i punti del grafico

Multiple-Plot Waveform Graph



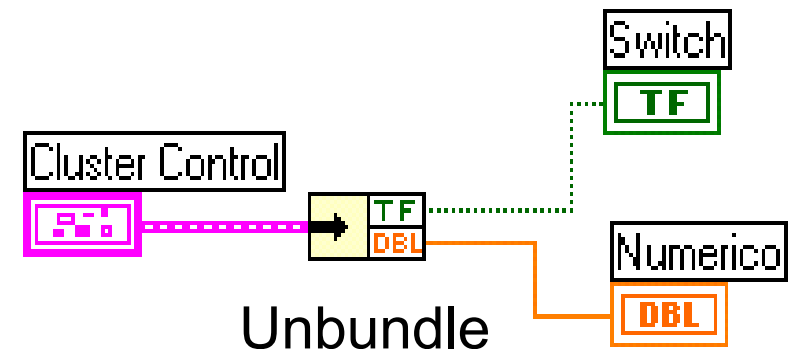
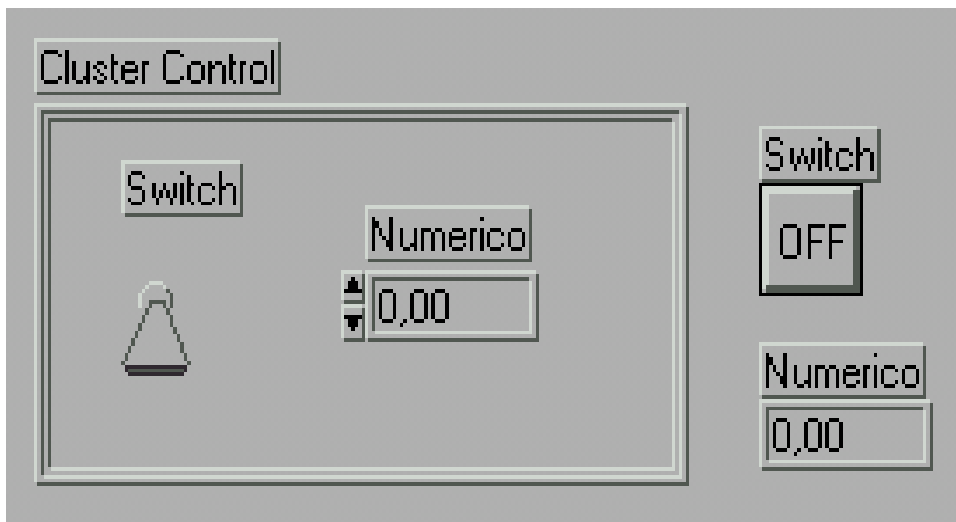
XY Graph

Traccia un array rispetto ad un altro

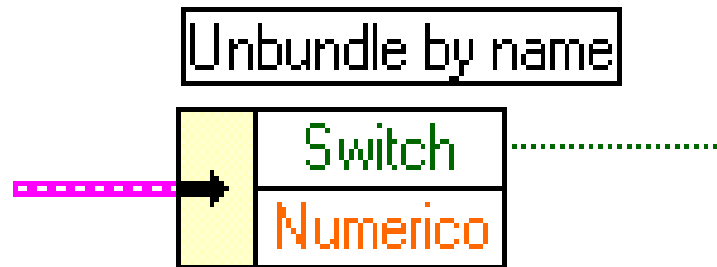


I Cluster

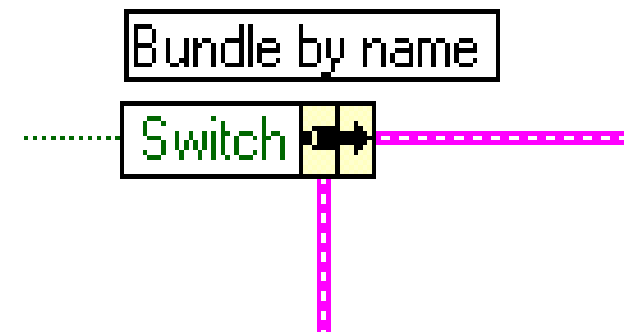
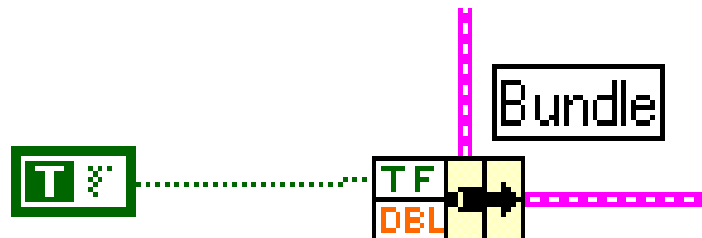
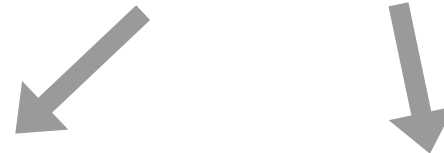
- „ Combinano più strutture dati in un nuovo tipo dati
- „ Possono contenere elementi di tipo diverso
- „ Si possono definire costanti, controlli e indicatori di tipo Cluster
- „ L'ordine degli elementi è importante



Altre funzioni per i Cluster



Modifica di un elemento
di un Cluster:

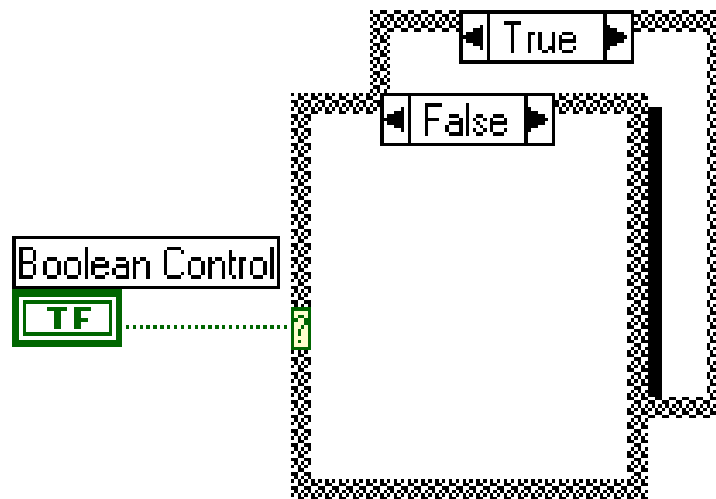


La struttura Case

- „ Solo un sottodiagramma viene eseguito
- „ Il valore passato al selettore determina il codice da eseguire
- „ Il selettore può essere booleano, numerico o stringa

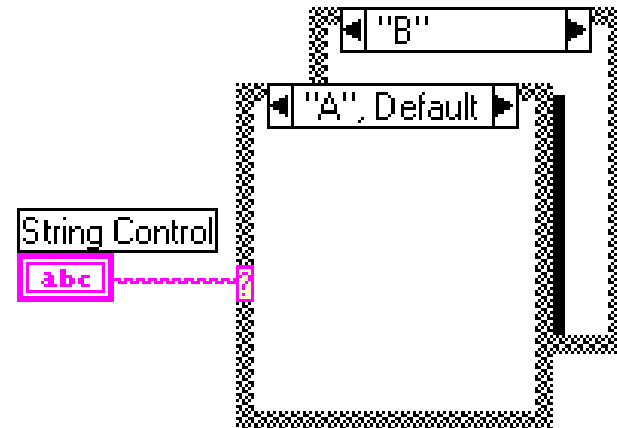
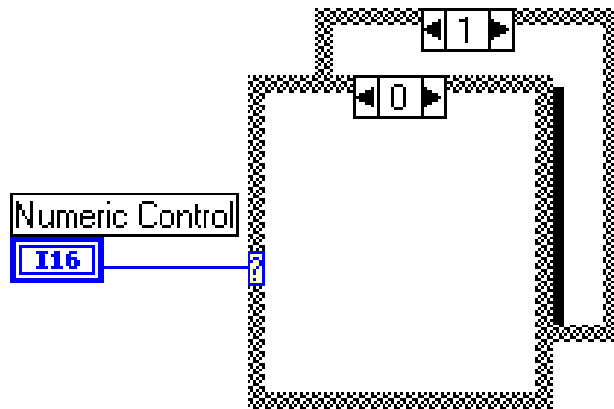
Selettore booleano:

if . . . then . . . else



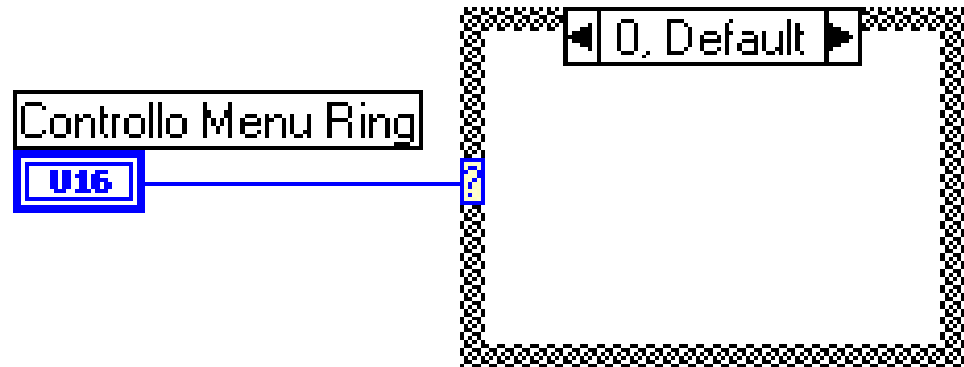
La struttura Case

- „ Con selettori di tipo numerico o stringa si possono gestire scelte multiple
- „ “Add Case” dal menu di contesto per aggiungere nuove scelte
- „ Un *Case* deve essere quello di *default*

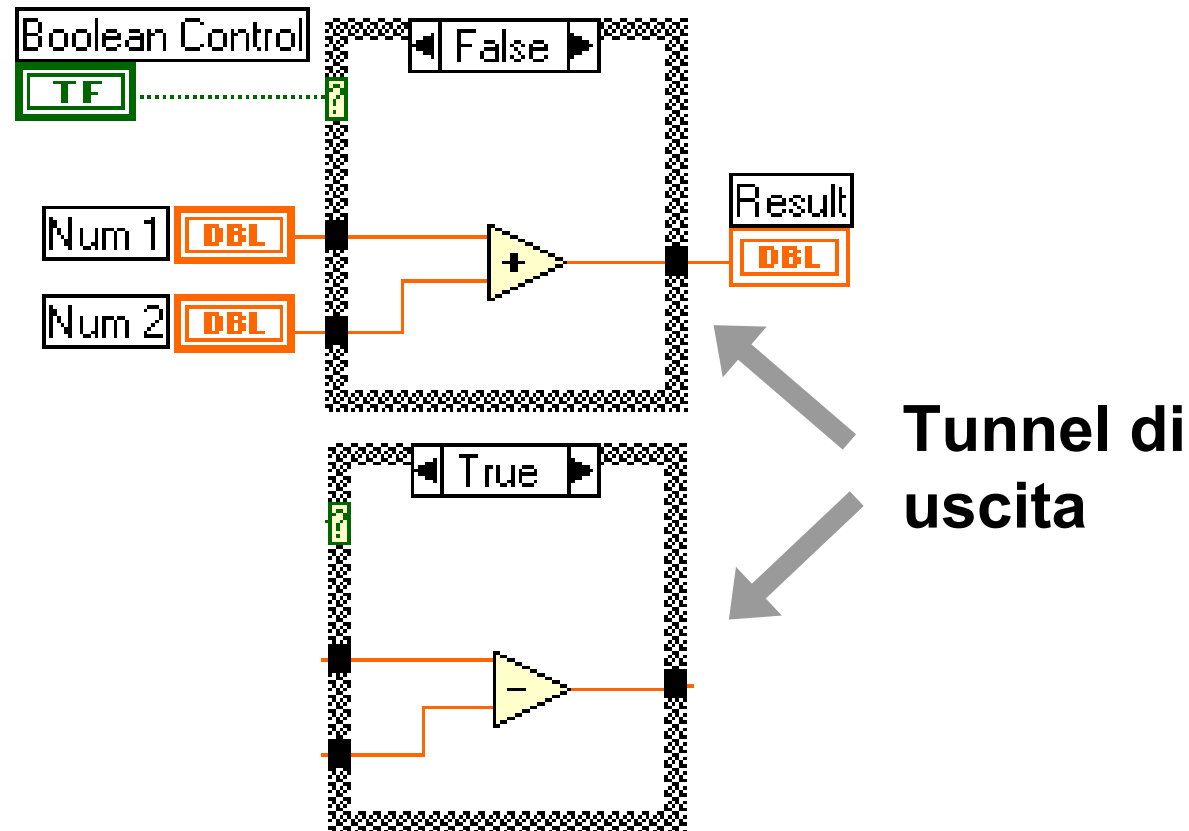


Menu di scelta

- I controlli “List & Ring” associano un intero ad una voce selezionabile



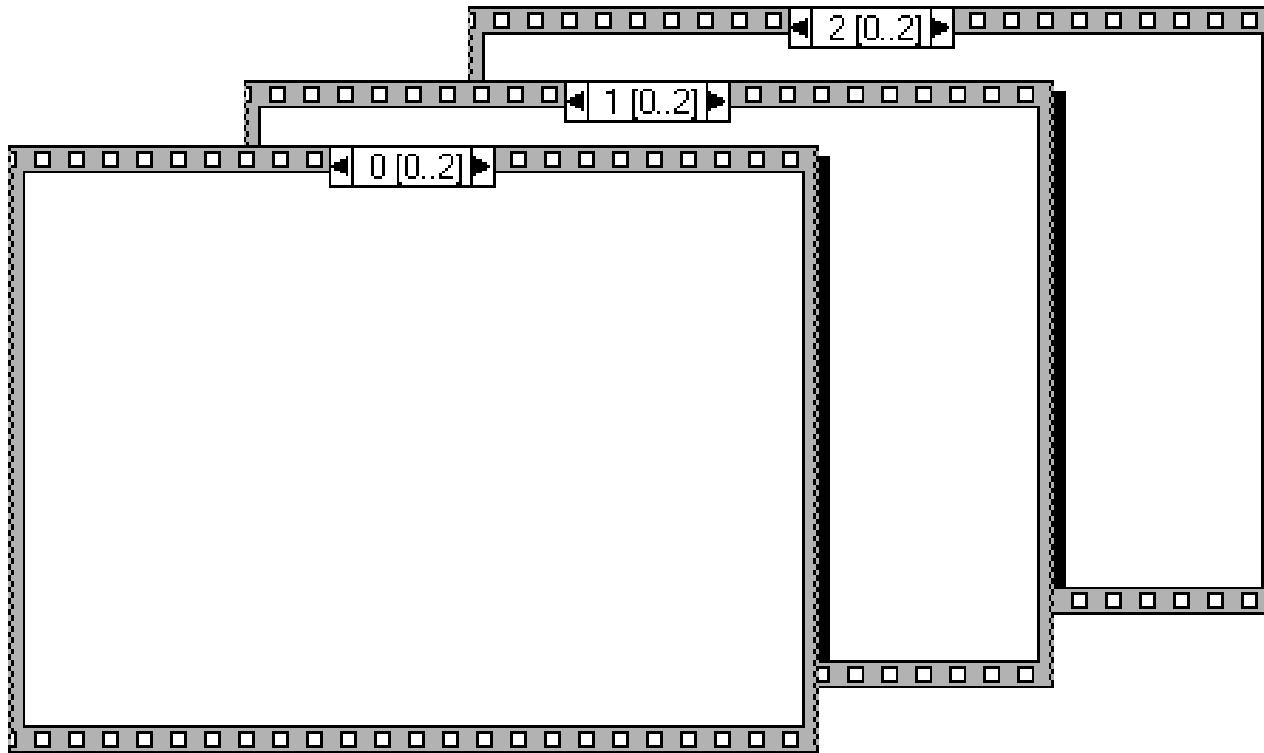
Ingressi ed uscite del **case** (tunnel)



- In TUTTI i case devono essere collegati TUTTI i tunnel

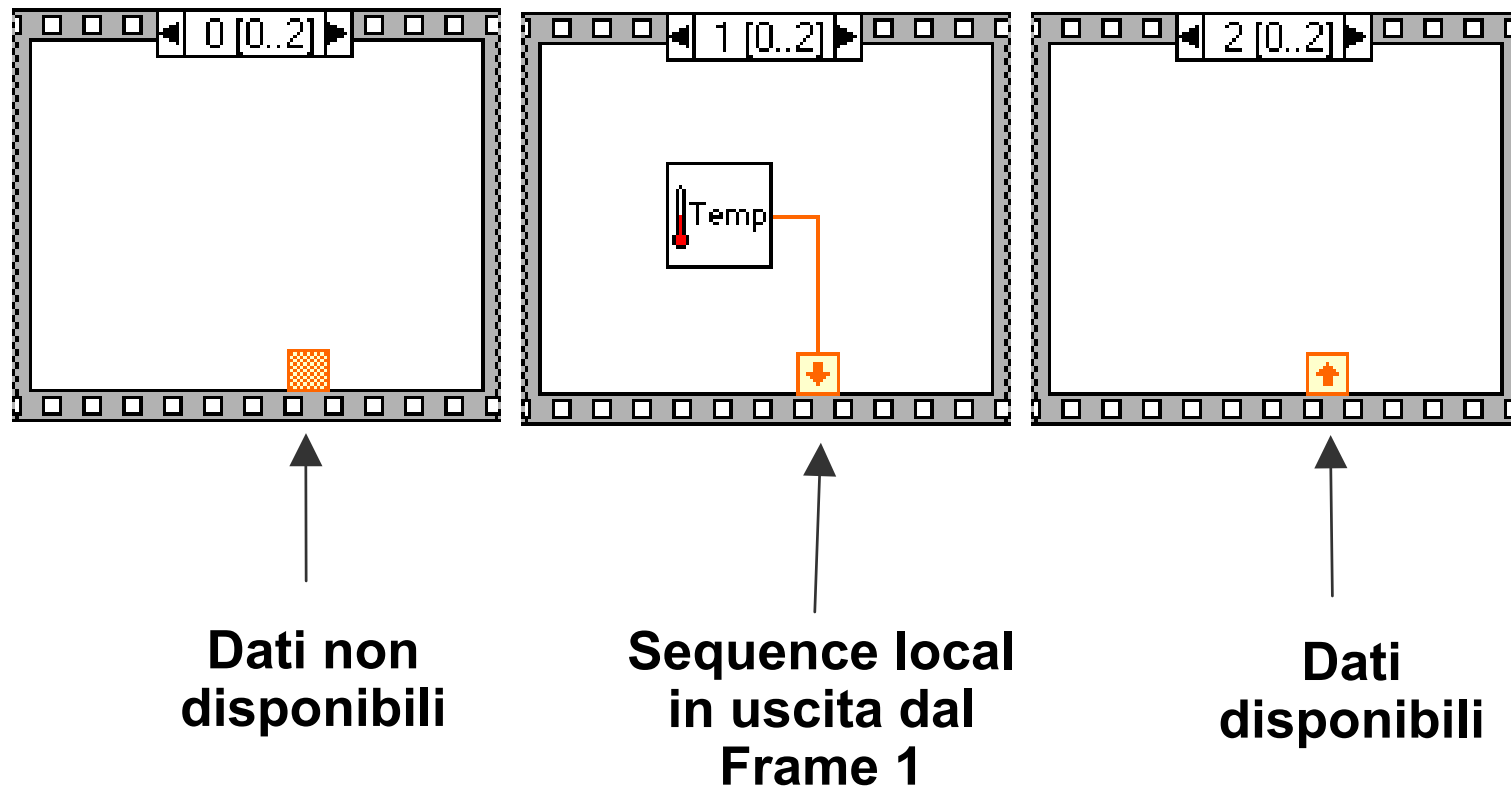
La struttura Sequence

- „ I sottodiagrammi vengono eseguiti uno dopo l'altro
- „ Consente di fissare l'ordine di esecuzione di VI non dipendenti



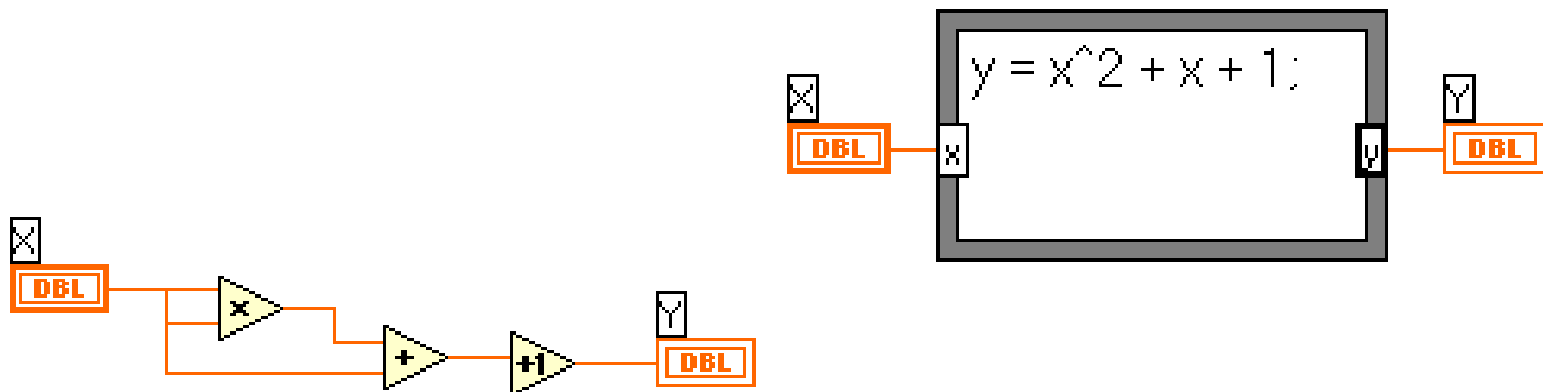
Sequence Locals

- Passano dati da un frame al solo frame successivo
- Si definiscono sul bordo della Sequence



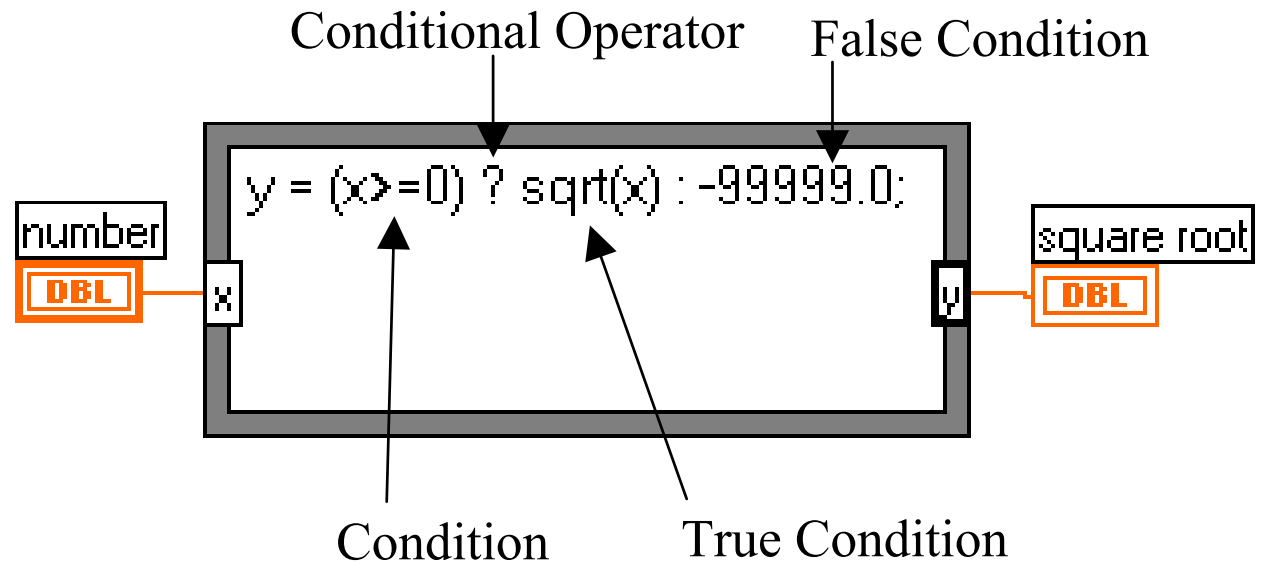
Formula Node

- Implementazione di equazioni complicate
- Variabili create sul bordo
- I nomi delle variabili sono *case sensitive*
- Ogni istruzione deve terminare con un punto e virgola (;)
- <Ctrl-H> per l'elenco delle funzioni disponibili



Formula Node: Conditional Branching

```
if (x >= 0) then  
    y = sqrt(x)  
else  
    y = -99999.0  
end if
```



Formula Node: Variabili temporanee

- „ Si definiscono come uscite, ma non vengono collegate

