

Κεφάλαιο

2

ΔΟΜΕΣ του LabVIEW

Σκοπός

Σκοπός του κεφαλαίου είναι να παρουσιάσει τις βασικές δομές (structures) προγραμματισμού που χρησιμοποιούνται στο LabVIEW.

Λέξεις Κλειδιά:

- While Loop
 - For Loop
 - Case
 - Flat Sequence
 - Stacked Sequence
 - Stacked Sequence
 - Event
 - Local
 - Global
 - Formula Node
 - MathScript Node
 - Shfit Registers
-

2.1 Δομές

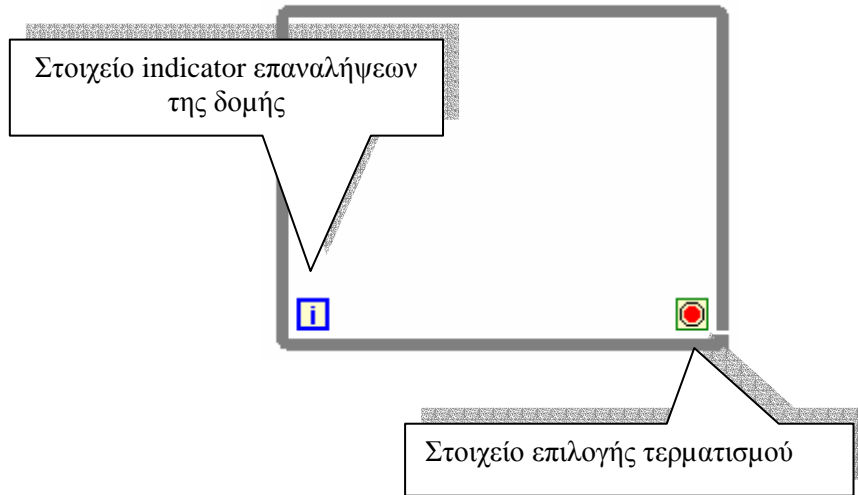
Κατά τον προγραμματισμό με το LabVIEW δομούμε τη λειτουργία του κώδικα στη χρήση των δομών (structures) που περιέχει. Οι βασικότερες δομές που καλύπτουμε στην ενότητα αυτή για το επίπεδο προγραμματισμού στο LabVIEW στο οποίο είναι γραμμένο το παρών βιβλίο είναι οι παρακάτω:

- Δομή While Loop
- Δομή For Loop
- Δομή Case
- Δομή Stacked/ Flat Sequence
- Δομή Event
- Δομή Formula Node

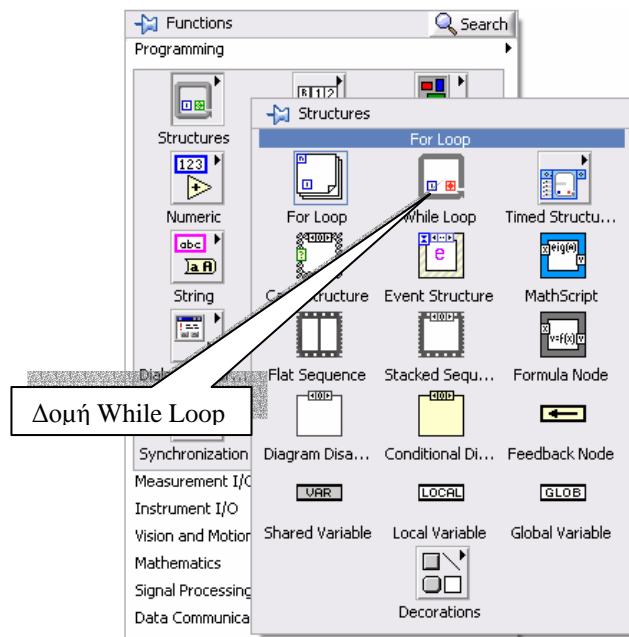
2.2 Δομή While Loop

Η δομή While Loop εκτελεί τις εντολές που περικλείει *Έως ότου...* αυτή τερματιστεί από μία λογική συνθήκη.

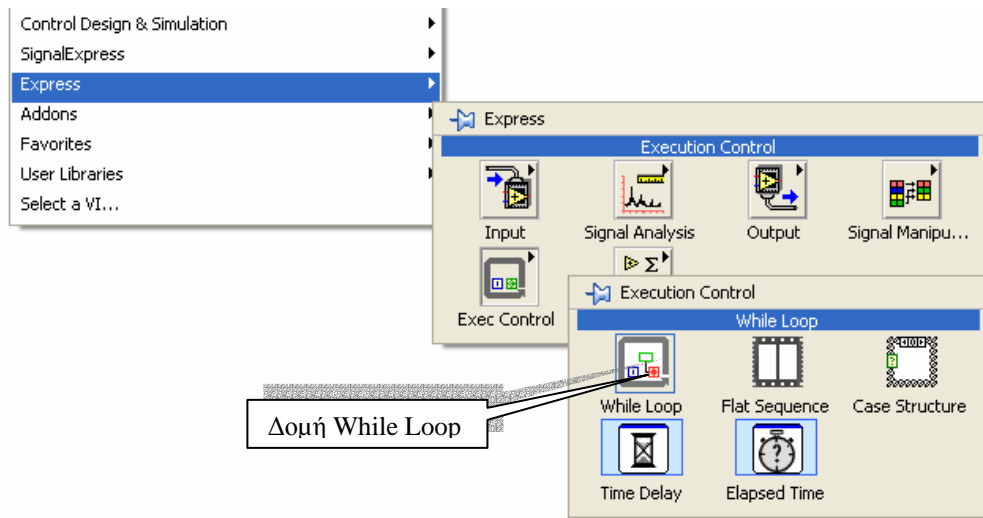
Η δομή While Loop (Έως ότου...)



Τη δομή While Loop την επιλέγουμε από την παλέτα **Function** → **Structures** στο block diagram.



Μπορούμε να επιλέξουμε τη δομή While Loop και από την παλέτα **Express**→**Execution Control**. Στην παλέτα αυτή στο τερματικό στοιχείο της δομής υπάρχει ως προεπιλεγμένο στοιχείο το πλήκτρο STOP



Δομικά στοιχεία δομής

Στη δομή While Loop συναντάμε τα δύο παρακάτω τερματικά στοιχεία:

- α) το τερματικό στοιχείο επανάληψης (iteration terminal)
- β) το τερματικό στοιχείο συνθήκης τερματισμού (conditional terminal)

- **Τερματικό στοιχείο επανάληψης (iteration terminal)**

Το τερματικό επανάληψης αποτελεί σημείο αρίθμησης των επαναλήψεων (iteration terminal) της δομής σε πραγματικό χρόνο που έχουν εκτελεστεί από τη δομή While Loop.



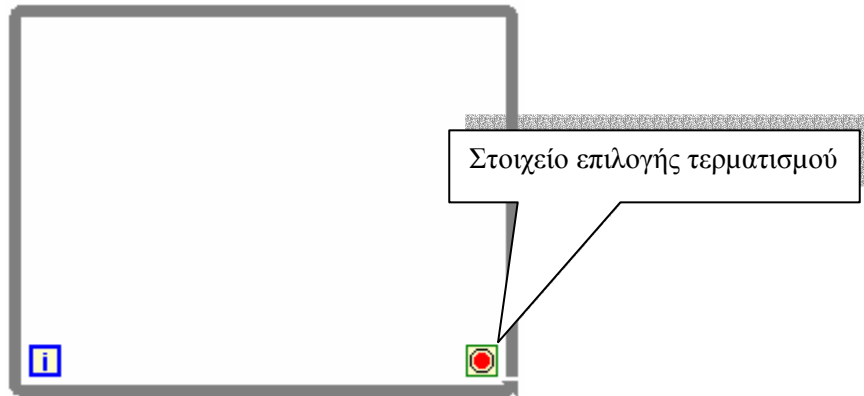
- **Τερματικό στοιχείο συνθήκης τερματισμού (conditional terminal)**

Το τερματικό συνθήκης τερματισμού αποτελεί σημείο τερματισμού της δομής While Loop (conditional terminal) όταν σε αυτό εφαρμοστεί λογικό δεδομένο.

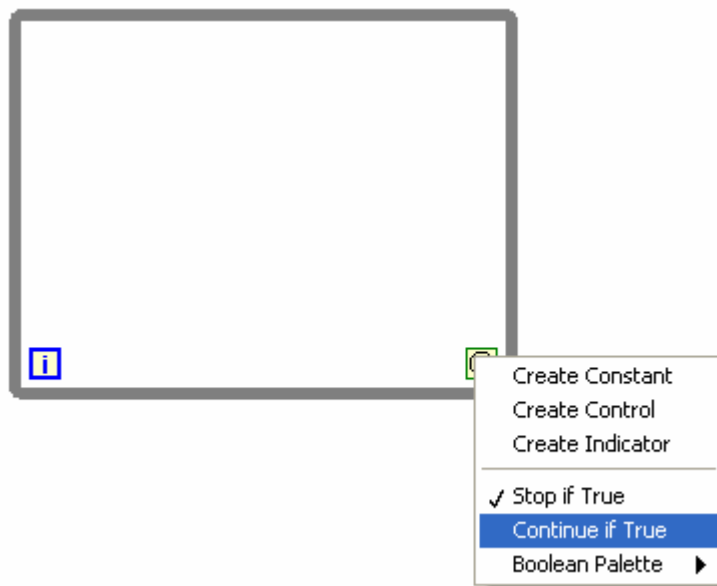


Η δομή While Loop εκτελεί τον κώδικα που είναι γραμμένος μέσα στα όρια του και κατόπιν, το LabVIEW ελέγχει την τιμή στο τερματικό στοιχείο συνθήκης όπου εάν είναι αληθής, τότε επαναλαμβάνει την εκτέλεση του. Εάν όμως η τιμή αλλάξει σε ψευδή (false) τότε η δομή While Loop σταματά να εκτελείται.

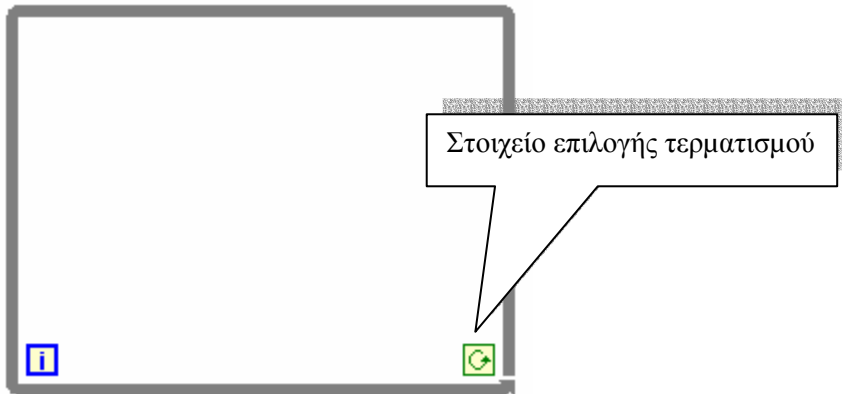
Η δομή παύει να εκτελείται για δεδομένο λογικού αληθές (Stop if True) στην είσοδο του τερματικού στοιχείου.



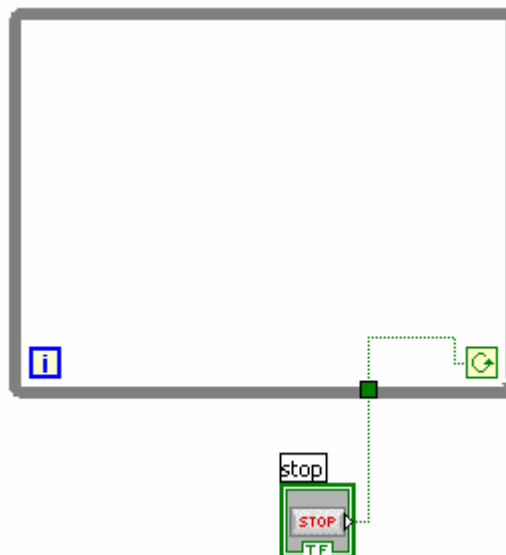
Μπορούμε να αλλάξουμε την κατάσταση συνθήκης τερματισμού επιλέγοντας όπως παρακάτω με αριστερό κλικ την συνθήκη Continue If True.



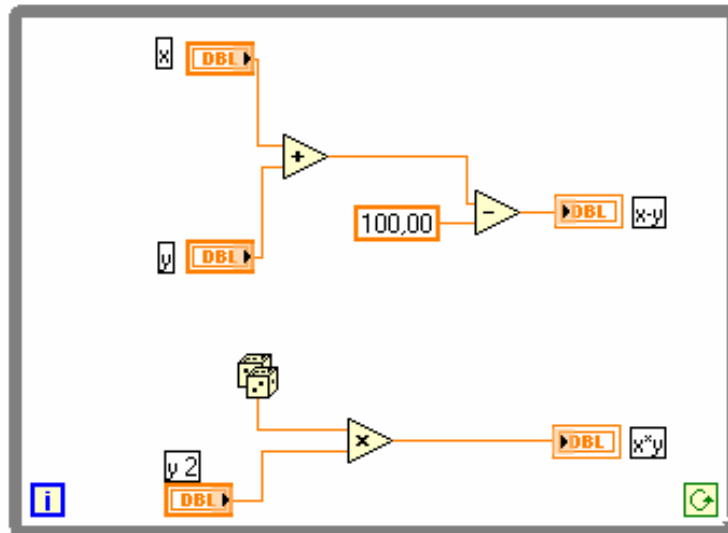
Η δομή συνεχίζει να εκτελείται για δεδομένο λογικού αληθές (Continue if True) στην είσοδο του τερματικού στοιχείου.




Όταν ένα boolean control στοιχείο έχει τοποθετηθεί έξω από τη δομή While Loop, δημιουργείται ένας ατέρμονος βρόγχος (Loop) ή ένας βρόγχος που εκτελείται μόνο μία φορά, ανάλογα με την αρχική τιμή του Boolean στοιχείου.

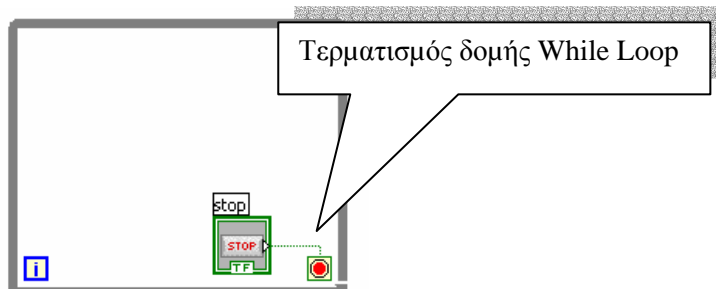


Όταν δεν συνδέουμε λογικό στοιχείο στο τερματικό συνθήκης, ο κώδικας τρέχει συνεχώς, αφού ποτέ δεν θα δεχτεί λογική τιμή (Boolean) στο τερματικό στοιχείο για να τερματιστεί ο βρόγχος.



2.2.1 Παράδειγμα χρήσης δομής While Loop

Στο παρακάτω παράδειγμα η δομή While Loop συνεχίζει να εκτελείται  έως ότου ο χρήστης πατήσει το πλήκτρο STOP στο front panel και σηματοδοτήσει με λογικό αληθές το τερματικό στοιχείο τερματισμού. Το πλήκτρο STOP το έχουμε τοποθετήσει από την παλέτα των αντικειμένων που βρίσκεται στο front panel.

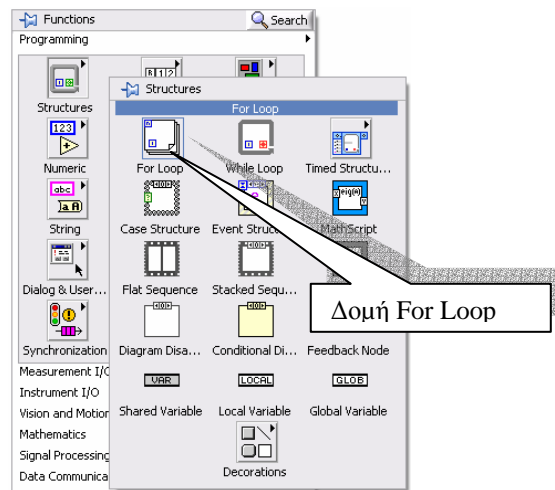


2.3 Δομή For Loop

Η δομή For Loop εκτελεί τις εντολές που περικλείει με συνθήκη επαναλήψεων:

$$\underline{\underline{I=0 \text{ έως } N-1}}$$


Τη δομή For Loop μπορούμε να την επιλέξουμε από την παλέτα **function** → **Structures** στο block diagram.




Δομικά στοιχεία δομής

Στη δομή For Loop συναντάμε τα δύο παρακάτω τερματικά στοιχεία:

- α) το τερματικό στοιχείο εισαγωγής του αριθμού επαναλήψεων της δομής (Number of Loops)
- β) το τερματικό στοιχείο επανάληψης (iteration terminal)

- Το τερματικό στοιχείο εισαγωγής του αριθμού επανάληψης της δομής (Number of Loops) 

Ο αριθμός των επαναλήψεων που θα εκτελέσει η δομή For Loop προσδιορίζεται από την τιμή του τερματικού στοιχείου N. Μπορούμε να χρησιμοποιήσουμε σταθερά ή στοιχείο control για να εισάγουμε τον αριθμό των επαναλήψεων του βρόγχου.

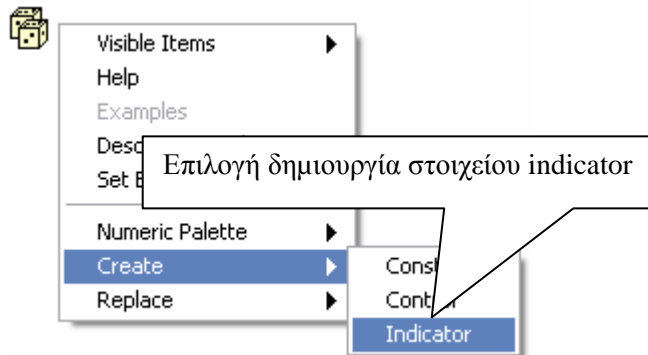
- Το τερματικό επανάληψης (iteration terminal) 

Ο αριθμός της επανάληψης στην οποία η δομή βρίσκεται ανά χρονική στιγμή προσδιορίζεται από το τερματικό στοιχείο επανάληψης του βρόγχου (iteration terminal – i).

Στο LabVIEW μπορούμε σε κάθε ακροδέκτη σύνδεσης να κάνουμε δεξί κλικ και να επιλέξουμε στοιχείο σύνδεσης στην επιλογή Create του pop up μενού του ακροδέκτη για το σημείο αυτό επιλέγοντας κατάλληλα:

- δημιουργίας σταθεράς
- δημιουργίας στοιχείου control
- δημιουργία στοιχείου indicator

Παράδειγμα δημιουργίας σύνδεσης indicator της λειτουργίας δημιουργίας τυχαίου αριθμού.

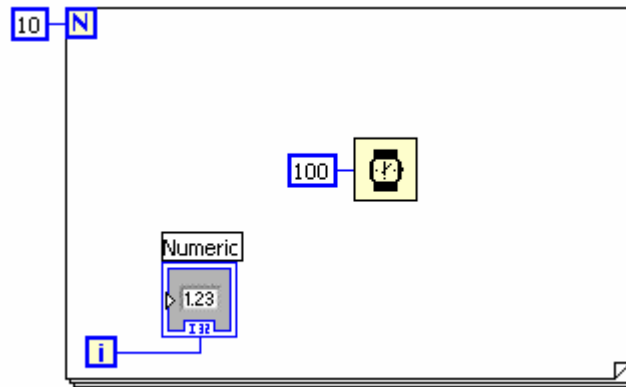


Παρόμοια μπορούμε να εφαρμόσουμε αυτή την τεχνική και για τη δημιουργία στοιχείου control στο σημείο ορισμού των επαναλήψεων της δομής For Loop, μειώνοντας κατ' αυτόν τον τρόπο τον χρόνο προγραμματισμού, αφού το LabVIEW θα επιλέξει το αντικείμενο και θα το τοποθετήσει αυτόματα στο front panel.

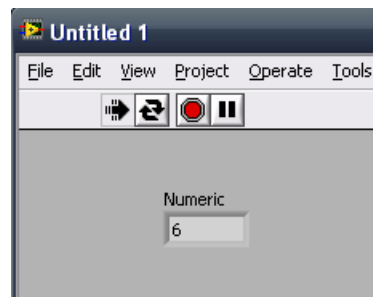
2.3.1 Παράδειγμα χρήσης δομής For Loop.

Στο παρακάτω παράδειγμα η δομή For Loop πραγματοποιεί 10 επαναλήψεις (δήλωση του αριθμού των επαναλήψεων με χρήση σταθεράς 10) με χρονική καθυστέρηση 100 msec όπως ορίζει η λειτουργία της χρονο-καθυστέρησης (Time delay). Μετά το πέρας των 10 επαναλήψεων η δομή τερματίζεται αυτομάτως. Ο αριθμός των επαναλήψεων N-1 εμφανίζεται σε στοιχείο indicator στο front panel του LabVIEW.

Block diagram



Front panel



2.4 Δομή Case

Η δομή Case περιέχει για τον προσδιορισμό της κατάστασης που θα εκτελεστεί ένα στοιχείο ελέγχου που καλείται επιλογέας (**selector terminal**),



Επίσης η δομή Case περιέχει και στοιχείο δείκτη κατάστασης ο οποίος απεικονίζει την κατάσταση στην οποία βρίσκεται η δομή

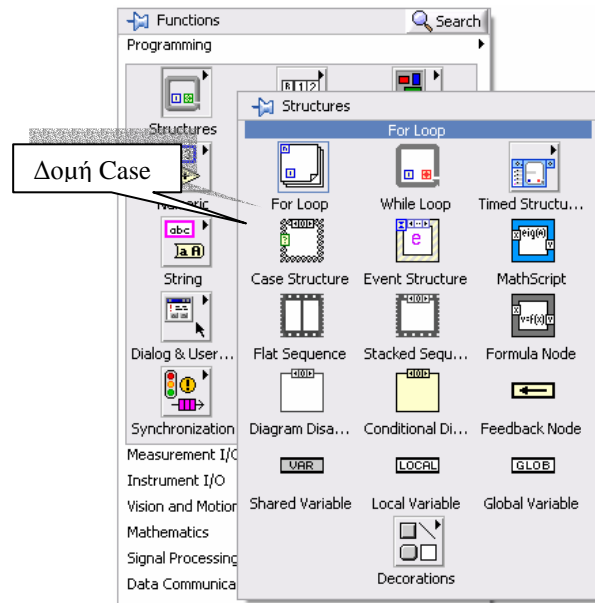


Στην είσοδο του επιλογέα μπορούμε να συνδέσουμε διαφορετικού τύπου δεδομένο. Στον επιλογέα μπορεί να συνδεθεί τύπος λογικού δεδομένου, ακέραιου αριθμού ή

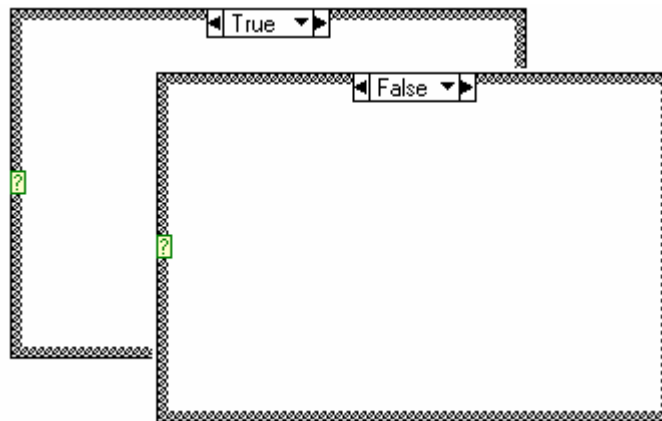
συμβολοσειρά. Ανάλογα με την τιμή του δεδομένου στον επιλογή αλλάζει και η ροή εκτέλεσης του κώδικα, ισχύει δηλαδή η συνθήκη:

Εάν – τότε - αλλιώς

Τη δομή Case μπορούμε να την επιλέξουμε από την παλέτα **function** → **Structures** στο block diagram.



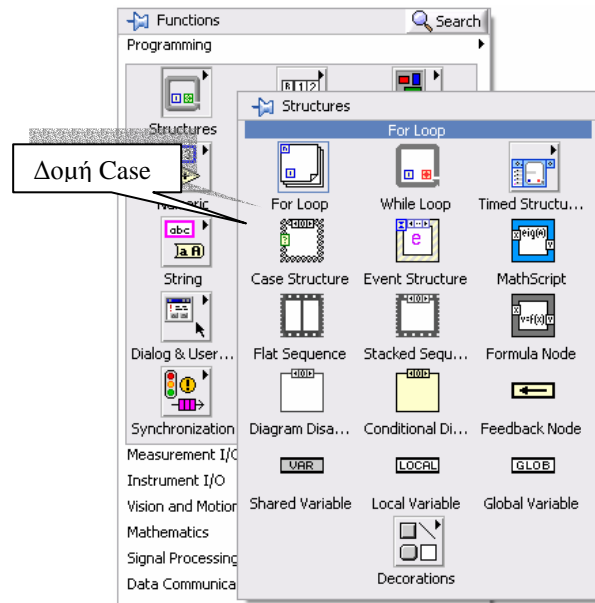
Λογικές συνθήκες (True/False) της δομής Case



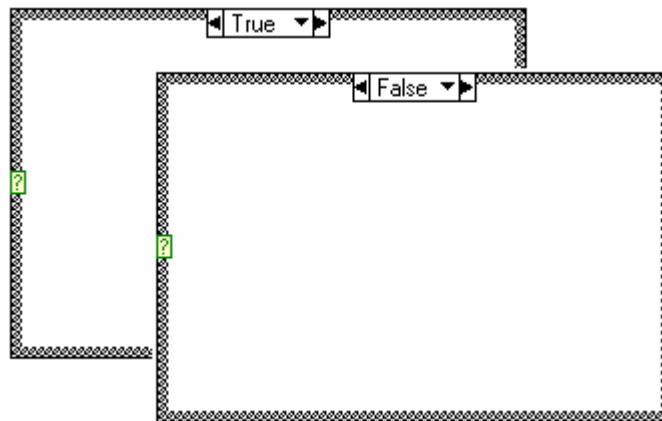
συμβολοσειρά. Ανάλογα με την τιμή του δεδομένου στον επιλογή αλλάζει και η ροή εκτέλεσης του κώδικα, ισχύει δηλαδή η συνθήκη:

Εάν – τότε - αλλιώς

Τη δομή Case μπορούμε να την επιλέξουμε από την παλέτα **function** → **Structures** στο block diagram.

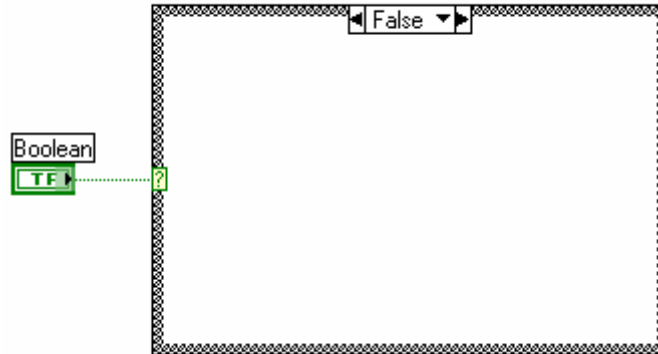


Λογικές συνθήκες (True/False) της δομής Case



2.4.1 Περίπτωση σύνδεση λογικής μεταβλητής στον επιλογέα

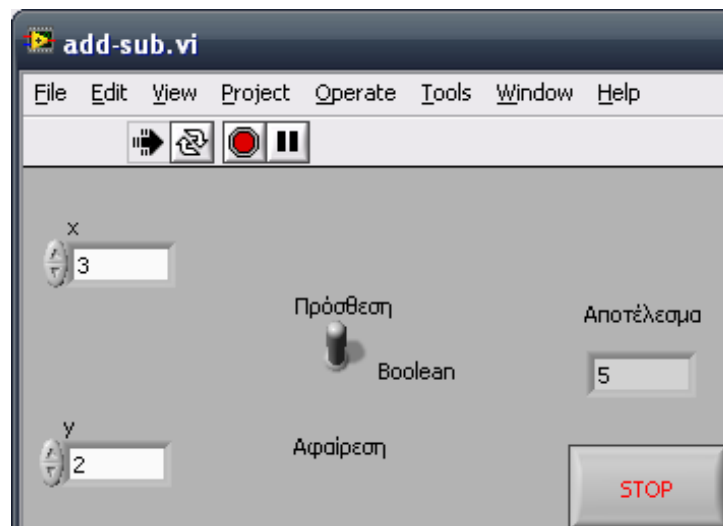
Σε αυτή τη περίπτωση ο κώδικας που θα εκτελεστεί εξαρτάται από τη μεταβλητή εισόδου ένα αυτή είναι λογικό αληθές (1) ή λογικό ψεύδος (0), όποτε και εκτελείται και ο αντίστοιχος κώδικας που περιβάλλεται στην συνθήκη με το δείκτη αληθές (True) ή ψευδές (False).



Παράδειγμα χρήσης δομής Case

Το παρακάτω εικονόργανο με διακόπτη επιλογής καθορίζει την πράξη της άθροισης ή της αφαίρεσης μεταξύ δύο μεταβλητών x και y και απεικονίζει το αποτέλεσμα σε indicator. (add-sub.vi)

Front panel



Block diagram

