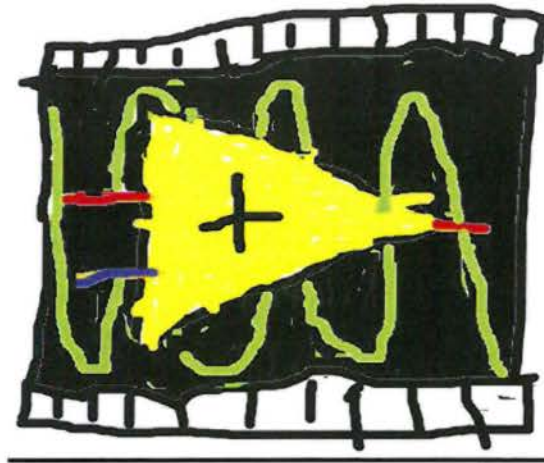


ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΠΕΙΡΑΙΑ



**ΚΑΤΑΣΚΕΥΗ MULTIMEDIA
ΕΦΑΡΜΟΓΗΣ ΚΑΙ AUDIOBOOK ΜΕ
ΘΕΜΑ “LABVIEW”**

ΡΙΤΣΑΤΟΣ ΑΓΓΕΛΟΣ

ΚΟΛΛΙΑΣ ΣΑΡΑΝΤΗΣ

**Επιβλέπουσα Καθηγήτρια
Αναστασία Ν. Βελώνη**

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
Ηλεκτρονικών Υπολογιστικών Συστημάτων**

ΙΟΥΝΙΟΣ 2014

ΤΕΙ ΠΕΙΡΑΙΑ
Π. Ραλλή & Θηβών 250, 12244 Αιγάτω , Αθήνα – Ελλάδα Τηλ. 210-5381488

**ΒΙΒΛΙΟΘΗΚΗ
ΤΕΙ ΠΕΙΡΑΙΑ**

ΠΕΡΙΛΗΨΗ

Η πτυχιακή εργασία "Κατασκευή Multimedia Εφαρμογής και AudioBook με θέμα 'LabVIEW'", περιλαμβάνει το πρόγραμμα «E-learning LabVIEW» και το εγχειρίδιο αυτό, τα οποία έχουν να κάνουν με την πλατφόρμα-πρόγραμμα LabVIEW. Αναφέρεται στη μαθησιακή χρήση τόσο σε επίδοξους μηχανικούς όσο και σε άτομα που απλώς θέλουν να γνωρίσουν το LabVIEW και να ασχοληθούν με τα συστήματα συλλογής δεδομένων.

Το πρόγραμμα «E-learning LabVIEW» που δημιουργήσαμε, περιέχει οπτικοακουστικό υλικό πάνω σε εφαρμογές LabVIEW, οι οποίες έχουν βιντεοσκοπηθεί και ηχογραφηθεί άμεσα από Η/Υ. Οι εφαρμογές μας δείχνουν τόσο απλές δραστηριότητες, όπως την δημιουργία εικονογράφων προσομοίωσης για την θερμοκρασία και την τάση, όσο και πιο πολύπλοκες, όπως τη δημιουργία συστημάτων αυτομάτου ελέγχου και επεξεργασία σήματος. Κάθε βιντεοσκοπημένη δραστηριότητα ακολουθείται από ηχογραφημένη ομιλία η οποία εξηγεί βήμα προς βήμα κάθε τι που κάνουμε καθώς και το λόγο της δραστηριότητας αυτής. Ο χρήστης του προγράμματός μας μπορεί επίσης να ακούσει ορισμούς του LabVIEW από ένα 'Audioobook' που δημιουργήσαμε και να μάθει πληροφορίες σχετικά με τις πηγές που χρησιμοποιήσαμε.

Ακολούθως μαζί με το «E-learning LabVIEW», παραδίδεται και αυτό το κείμενο, το οποίο είναι ένα εγχειρίδιο πάνω στο πρόγραμμα που δημιουργήθηκε. Περιέχει τρία κεφάλαια, με το πρώτο να μας δίνει πληροφορίες σχετικά με το LabVIEW, το δεύτερο να αναφέρεται στις εφαρμογές οι οποίες δημιουργήθηκαν και βιντεοσκοπήθηκαν και το τρίτο κεφάλαιο αφορά τα συστήματα συλλογής δεδομένων (DAS), με τα οποία έχει σχέση η πλατφόρμα του LabVIEW. Οι εφαρμογές στο δεύτερο κεφάλαιο μας δείχνουν βήμα προς βήμα τον τρόπο εκτέλεσης των δραστηριοτήτων μας, δείχνοντάς μας και εικόνες με το τί έχουμε κάνει. Επιπλέον υπάρχουν εικόνες πληροφοριών στα άλλα κεφάλαια. Στο τέλος της εργασίας υπάρχει παράρτημα με ορισμούς και εικόνες αντικειμένων του LabVIEW και πλήρης βιβλιογραφία.

ΠΕΡΙΕΧΟΜΕΝΑ

| | |
|-----------------|---|
| ΕΙΣΑΓΩΓΗ | 1 |
|-----------------|---|

ΚΕΦΑΛΑΙΟ 1: ΕΙΣΑΓΩΓΗ ΣΤΟ LABVIEW

| | | |
|-------|-------------------------------------|----|
| 1.1 | LabVIEW | 4 |
| 1.1.1 | Το Περιβάλλον LabVIEW | 4 |
| 1.1.2 | Το Πεδίο Παρουσίασης (Front Panel) | 6 |
| 1.1.3 | Το Λογικό Διάγραμμα (Block Diagram) | 7 |
| 1.1.4 | Εκτέλεση των Vis | 8 |
| 1.2 | Εργαλεία | 9 |
| 1.2.1 | Tools Palette | 9 |
| 1.2.2 | Controls Palette | 10 |
| 1.2.3 | Functions Palette | 11 |
| 1.2.4 | Μενού Pop-Up | 15 |
| 1.3 | Βιβλιοθήκες από Vis (VI Libraries) | 16 |
| 1.3.1 | Φορτώνοντας Vis | 16 |
| 1.3.2 | Μεταφορά VIs σε άλλες πλατφόρμες | 17 |
| 1.4 | Η χρήση βοήθειας στο LabVIEW | 18 |

ΚΕΦΑΛΑΙΟ 2 ΕΦΑΡΜΟΓΕΣ ΣΤΟ LABVIEW

| | | |
|------|--|----|
| 2.1 | Θερμόμετρο | 20 |
| 2.2 | Βολτόμετρο | 23 |
| 2.3 | Κύκλωμα RLC σε σειρά | 26 |
| 2.4 | Δομές Επανάληψης | 30 |
| 2.5 | Απεικόνιση δεδομένων - Γραφήματα | 34 |
| 2.6 | Αποθήκευση Δεδομένων | 39 |
| 2.7 | Δημιουργία Σήματος | 42 |
| 2.8 | Φίλτρο | 49 |
| 2.9 | Δημιουργία συστήματος Ανοιχτού Βρόγχου | 55 |
| 2.10 | Δημιουργία συστήματος Κλειστού Βρόγχου | 59 |
| 2.11 | PID Controller σε σύστημα Κλειστού Βρόγχου | 60 |
| 2.12 | Γεωμετρικός Τόπος Ριζών | 63 |
| 2.13 | Δημιουργία SubVi | 67 |

ΚΕΦΑΛΑΙΟ ΤΡΙΤΟ:

Ανάπτυξη & Προγραμματισμός Συστημάτων Συλλογής Δεδομένων

| | | |
|-----|--|----|
| 3.1 | Συστήματα συλλογής δεδομένων & LabVIEW | 73 |
| 3.2 | Υλικό συστημάτων μετρήσεων και ελέγχου | 75 |
| 3.3 | Ενσωμάτωση προγραμμάτων και συστημάτων | 78 |
| 3.4 | Χρήση συσκευών καταγραφής δεδομένων | 80 |

| | |
|---------------------|----|
| ΒΙΒΛΙΟΓΡΑΦΙΑ | 86 |
|---------------------|----|

ΠΑΡΑΡΤΗΜΑ:

| | |
|------------------------------------|----|
| Tools Pallet | 87 |
| Controls Pallet | 88 |
| Functions Pallet | 89 |
| Τα Εργαλεία του Front Panel | 90 |
| Τα εργαλεία του μπλόκ Διαγράμματος | 92 |
| Icon Editor | 92 |

ΕΙΣΑΓΩΓΗ

LabVIEW

Τα συστήματα συλλογής δεδομένων (Data acquisition systems: DAS), υπάρχουν σχεδόν από την αρχή των πληροφορικών συστημάτων και την μηχανική υπολογιστών. Αισθητήρες, ηλεκτρονικά κυκλώματα και μετατροπής αναλογικού σε ψηφιακό σήμα δεν αρκούν καθώς πρέπει να ελέγχονται από το κατάλληλο λογισμικό. Γλώσσες προγραμματισμού όπως η BASIC, C, Fortan, Pascal από μόνες τους δυσκολεύουν περισσότερο τις εργασίες γι' αυτό το λόγο αναπτύχθηκαν προγράμματα βασισμένα σε αυτές για να κάνουν αυτή την δουλειά. Ένα τέτοιο λογισμικό είναι το LabVIEW.

Το LabVIEW (Laboratory Virtual Instrument Engineering Workbench) αποτελεί ένα δυνατό και ευέλικτο πρόγραμμα ανάπτυξης εφαρμογών σχεδιασμού και ανάλυσης που αναπτύχθηκε από την National Instruments. Το περιβάλλον ανάπτυξής του χρησιμοποιεί μια γραφική-οπτική γλώσσα προγραμματισμού (visual programming language) που είναι γνωστή ως G. Το LabVIEW ξεφεύγει από την παραδοσιακή φύση γλωσσών προγραμματισμού, εισάγοντας τον χρήστη σε ένα γραφικό περιβάλλον για instrumentation με όλα τα εργαλεία για συλλογή μετρήσεων, έλεγχο αυτόνομων οργάνων, ανάλυση και παρουσίαση δεδομένων. Περιλαμβάνει συμβατικά όργανα διόρθωσης λαθών στα οποία μπορούμε να θέσουμε σύμβολα σημείων διακοπής, εκτέλεση προγράμματος βήμα-βήμα και εκτέλεση κίνησης για την παρακολούθηση της ροής δεδομένων. Για την ανάλυση δεδομένων, υπάρχουν ρουτίνες επεξεργασίας σήματος, φίλτρων, στατιστικής παρεμβολής, γραμμικής άλγεβρας κ.ο.κ. ενώ παρουσιάζει πολλά πλεονεκτήματα σαν σύστημα παρουσίασης αποτελεσμάτων. Φυσικά όλα αυτά θα ήταν δύσκολο να γίνουν αν το LabVIEW δεν υποστήριζε για την συλλογή μετρήσεων και τον έλεγχο οργάνων, μέσα ελέγχου όπως σειριακές θύρες RS-232/422/485, IEEE488(GPIB), VISA, VXI, καθώς επίσης και κάρτες δειγματοληψίας. Μία ολοκληρωμένη βιβλιοθήκη από drivers για όργανα διευκολύνει τον έλεγχο αυτόνομων οργάνων.

ΣΚΟΠΟΣ ΚΑΙ ΣΤΟΧΟΣ

Βασικό πλεονέκτημα του LabVIEW σε σχέση με άλλα περιβάλλοντα ανάπτυξης είναι η υποστήριξή του με υλικολογισμικό και συνεχής ανανέωση του.

Αυτό έχει ως αποτέλεσμα την εξοικονόμηση χρόνου ανάπτυξης του προγράμματος. Άτομα με περιορισμένη εμπειρία μπορούν να γράψουν προγράμματα και να αναπτύξουν λύσεις δοκιμής σε ένα μειωμένο χρονικό διάστημα σε σύγκριση με πιο συμβατικά ή ανταγωνιστικά συστήματα. Βοηθάει λοιπόν με αυτό τον τρόπο τους μαθητές-μηχανικούς να κατανοήσουν πιο εύκολο την μέθοδο της συλλογής δεδομένων, τόσο στο προγραμματιστικό όσο και στο hardware κομμάτι του, όπως τα ηλεκτρικά κυκλώματα και τους αισθητήρες. Στην εργασία που ακολουθεί τόσο στο γραπτό όσο και στο πρόγραμμα του «E-learning LabVIEW», παρουσιάζονται εύκολες εφαρμογές που μπορεί ο χρήστης να κατανοήσει την σημασία του LabVIEW στην μηχανική της συλλογής δεδομένων.

Στο *Κεφάλαιο 1* κάνουμε μια εισαγωγή στην πλατφόρμα του LabVIEW. Δίνονται στον χρήστη-μαθητή ορισμοί του προγράμματος και του αντικείμενου που χρησιμοποιεί το LabVIEW. Χωρίς την κατανόηση αυτών δεν θα μπορεί να κατανοήσει τις δραστηριότητες που έχουμε δημιουργήσει. Επιπλέον υπάρχουν εικόνες οι οποίες παρουσιάζουν τα αντικείμενα-παλέτες του προγράμματος και εκτενής αναφορά για κάθε μενού σε αυτό.

Το *Κεφάλαιο 2* παρουσιάζει τις 13 δραστηριότητες που έχουμε δημιουργήσει στο πρόγραμμα «E-learning LabVIEW». Αναλύονται με ακρίβεια τα βήματα που πρέπει να ακολουθήσει ο χρήστης ώστε να δημιουργήσει τις εφαρμογές αυτές. Στην συνέχεια υπάρχουν εικόνες οι οποίες αποτυπώνουν τα πάνελ του LabVIEW κατά την διάρκεια πραγματοποιήσεις των εφαρμογές. Οι εφαρμογές αφορούν απλές δραστηριότητες όπως την προσομοίωση εικονικών οργάνων αλλά και πιο περίπλοκες όπως την μελέτη συστημάτων αυτομάτου ελέγχου και στην αναλογική-ψηφιακή επεξεργασία σημάτων. Αυτό έχει να κάνει με την συλλογή και καταγραφή δεδομένων. Οι εφαρμογές αυτές είναι: Θερμόμετρο, Βολτόμετρο, Κύκλωμα RLC σε σειρά, Δομές Επανάληψης, Απεικόνιση δεδομένων – Γραφήματα, Αποθήκευση Δεδομένων, Δημιουργία Σήματος, Φίλτρο, Δημιουργία συστήματος Ανοιχτού Βρόγχου, Δημιουργία συστήματος Κλειστού Βρόγχου, PID Controller σε σύστημα Κλειστού Βρόγχου, Γεωμετρικός Τόπος Ριζών, Δημιουργία SubVi. Είναι αξιοσημείωτο να αναφέρουμε ότι το παρόν κεφάλαιο δεν λειτουργεί μόνο σαν εγχειρίδιο των εφαρμογών του προγράμματός μας αλλά διαβάζοντας το και μόνο ο χρήστης μπορεί να πραγματοποιήσει τις εφαρμογές στον δικό του υπολογιστή.

Το *Κεφάλαιο 3* αναφέρεται στα συστήματα συλλογής και καταγραφής δεδομένων(DAS). Δίνονται ορολογίες σχετικά με το συγκεκριμένο αντικείμενο και στη συνέχεια γίνεται λόγο για το hardware και τα πρότυπα που χρησιμοποιεί το LabVIEW όπως κάρτες δειγματοληψίας, αισθητήρες, θύρες διασύνδεσης και τα προγράμματα με τα οποία μπορεί να συνεργαστεί και ενσωματώσει σε αυτό. Τέλος, δίνεται ένα παράδειγμα χρήσης συσκευής καταγραφής δεδομένων στα πρότυπα του LabVIEW.

ΕΡΓΑΣΙΑ

Η συγκεκριμένη πτυχιακή εργασία αποτελείται από μια e-learning πλατφόρμα στην οποία περιέχεται οπτικοακουστικό υλικό βασισμένο σε εφαρμογές του LabVIEW. Σκοπός αυτών των εφαρμογών είναι να δείξει στους μηχανικούς τις βασικές λειτουργίες και τα πλεονεκτήματα του. Εφαρμογές όπως η δημιουργία συστήματος ανοιχτού και κλειστού βρόγχου, PID ελεγκτές, πρακτικές εφαρμογές όπως το πως δουλεύει ένα βολτόμετρο, η δημιουργία ενός θερμόμετρου και άλλα παρόμοια βοηθάνε στο σκοπό αυτό. Για την υλοποίηση του πρακτικού μέρους της πτυχιακής μας χρησιμοποιήσαμε τα προγράμματα Mediator, Camptasia Studio και το LabVIEW 2011.

Το Camptasia Studio είναι λογισμικό με το οποίο μπορούμε να δημιουργήσουμε βίντεο (οποιαδήποτε ανάλυση) καταγράφοντας την οθόνη του υπολογιστή μας (Camptasia Recorder) καθώς και να ηχογραφήσουμε ήχο. Με αυτόν τον τρόπο καταγράψαμε όλες τις εφαρμογές που δημιουργήσαμε στο LabVIEW. Με το Studio Editor του προγράμματος ενοποιήσαμε τα βίντεο που καταγράψαμε με την σειρά που τα θέλαμε και προσαρμόσαμε τον επιπλέον ήχο που μπορεί να χρειαστήκαμε. Στην δική μας εργασία χρησιμοποιήσαμε την έκδοση 7. Τα βίντεο τα οποία δημιουργήθηκαν από το Camptasia τα εισαγάγαμε στην πλατφόρμα μας, η οποία δημιουργήθηκε με το πρόγραμμα Mediator.

Το πρόγραμμα Matchware Mediator είναι ένα εργαλείο με το οποίο μπορούμε να δημιουργήσουμε multimedia εφαρμογές όπως παρουσιάσεις με οπτικοακουστικό υλικό, επισύναψη video, quiz τα οποία μπορούμε να ενσωματώσουμε σε ένα CD-DVD. Ένα από τα βασικά πλεονεκτήματα του προγράμματος είναι ότι το πρόγραμμα το οποίο θα δημιουργηθεί μπορεί να αναπαραχθεί σε οποιοδήποτε υπολογιστή σαν αρχείο .exe.

ΚΥΡΙΟΣ ΜΕΡΟΣ

ΚΕΦΑΛΑΙΟ 1: ΕΙΣΑΓΩΓΗ ΣΤΟ LABVIEW

1.1 LabVIEW

Το LabVIEW είναι η πιο κομψή και δυνατή γλώσσα προγραμματισμού για τη συλλογή δεδομένων, την ανάλυση δεδομένων, την προσομοίωση και τον έλεγχο οργάνων και μετρήσεων μέσω υπολογιστή. Στηρίζεται στον γραφικό προγραμματισμό μέσω αντικειμένων και αποτελεί ένα καλό παράδειγμα του «αντικειμενοστραφή προγραμματισμού» (object oriented programming). Αυτός ο όρος χρησιμοποιείται στην πληροφορική σε αντιδιαστολή με τον λεγόμενο «προγραμματισμό διαδικασιών», όπου ο προγραμματιστής γράφει κώδικα εντολών που εκτελούνται με γραμμική διαδοχή. Στο γραφικό περιβάλλον του LabView ο προγραμματιστής δεν χειρίζεται κώδικα, αλλά γραφικά αντικείμενα, όπως κουμπιά, ενδείκτες, οθόνες ή τετραγωνίδια που παριστάνουν συναρτήσεις ή εκτελούν συγκεκριμένες λειτουργίες με τη μορφή υπορουτινών. Αυτά τα εικονίδια έχουν εισόδους και εξόδους και επιδέχονται προγραμματισμό των ιδιοτήτων τους.

Προγραμματίζοντας με τα αντικείμενα που μας δίνει το περιβάλλον του LabView δημιουργούμε τα λεγόμενα «εικονικά όργανα» (Virtual Instruments ή απλώς VIs). Η γραφική γλώσσα που χρησιμοποιεί το LabView για τον προγραμματισμό και τη δημιουργία εικονικών οργάνων ονομάζεται γλώσσα G. Είναι πάντως δυνατό να χρησιμοποιήσει κανείς το LabView σε απλό επίπεδο, χρησιμοποιώντας έτοιμα εικονικά όργανα, χωρίς να μπαίνει σε λεπτομέρειες με τη γλώσσα G. Ως παράδειγμα, φανταστείτε ότι θα μπορούσε κανείς να δημιουργήσει ένα εικονικό όργανο που θα προσομοίωνε έναν παλμογράφο. Η λειτουργία του οργάνου (με ποιον τρόπο θα αποθηκεύονται π.χ. οι μετρήσεις), θα οριζόνταν από το λογικό διάγραμμα του VI. Από την άλλη, οι εισοδοί στο πρόγραμμα (π.χ. συνολικός αριθμός μετρήσεων κλπ.), αλλά και οι εξοδοί (π.χ. τα γραφήματα), θα οριζόνταν / εμφανίζονταν, στο front panel.

1.1.1 Το Περιβάλλον LabVIEW

Το Περιβάλλον LabVIEW αποτελείται από την εφαρμογή LabVIEW και από μερικά άλλα αρχεία. Θα εξετάσουμε μόνο τί συμβαίνει για τα λειτουργικά συστήματα Windows.

Αρχεία και Φάκελλοι

Το LabVIEW χρησιμοποιεί διάφορα αρχεία και φακέλλους για να σώσει χρήσιμες πληροφορίες.

- Ο Φάκελλος vi.lib. Περιλαμβάνει βιβλιοθήκες από VIs όπως αυτούς της συλλογής μετρήσεων (DAQ) και ανάλυσης. Μην αλλάξετε ποτέ το όνομα του vi.lib καθώς το LabVIEW ψάχνει για αυτό, όταν ξεκινά. Με την αλλαγή του ονόματος, το LabVIEW δεν θα γνωρίζει που θα βρεί πολλές από τις ρουτίνες που χρειάζεται για να τρέξει.
- Ο Φάκελλος με τα παραδείγματα(Examples). Εδώ θα βρείτε πλήθος από έτοιμα να τρέξουν παραδείγματα για πολλές και διαφορετικές εφαρμογές.
- Ο Φάκελλος με τα cintools. Περιέχει αρχεία για την σύνδεση και τρέξιμο ρουτινών σε γλώσσα C, μέσα από το LabVIEW.
- Ο Φάκελλος με τα Μενού(Menus), στον οποίο αποθηκεύονται πληροφορίες σχετικές με τα διάφορα μενού.
- Βοήθεια(Help). Περιλαμβάνει όλα τα απαραίτητα αρχεία. Τοποθετώντας εδώ διάφορα VIs και βιβλιοθήκες από VIs, μπορείτε να τα βλέπετε μέσα από το Μενού Βοήθειας(Help Menu).
- Ο Φάκελλος user.lib, που περιέχει όλες τις βιβλιοθήκες στις οποίες θέλετε να έχετε πρόσβαση μέσα από την παλέττα Functions στο μπλόκ διάγραμμα. Εδώ, αποθηκεύσαμε και τους επιπλέον VIs που χρησιμοποιούμε στο μάθημα.
- Η Βιβλιοθήκη BASCLASS.LLB. Περιέχει όλους τους VIs που χρειάζεστε για το μάθημα.

Η Οθόνη Getting Started

Όταν ξεκινάτε το LabVIEW, με διπλο κλικ στην εικόνα του, εμφανίζεται το παραθυρο Getting Started, η αρχική οθονη δηλαδή. Εδώ, μπορείτε να βρείτε έτοιμα παραδείγματα, την on-line βοήθεια κ.λ.π. Επιλέξτε Blank VI.

Τα Παράθυρα γιά το Front Panel και Block Diagram

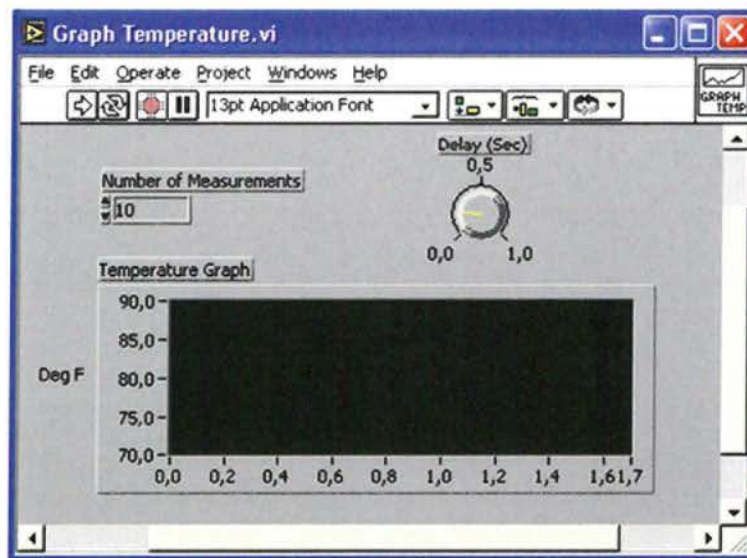
Όταν επιλέξετε NEW VI, ανοίγει αυτόματα ένα νέο παράθυρο χρώματος γκρι, το οποίο είναι το Front Panel του νέου σας VI. Το άλλο παράθυρο είναι το παράθυρο που θα περιέχει το μπλόκ Διάγραμμα. Το Front Panel περιέχει διάφορους τύπους από αντικείμενα, σε κάθε ένα από τα οποία αντιστοιχεί και ένα ακροδεκτη (terminal) στο μπλόκ διάγραμμα. Εκτός από τα διάφορα terminals στο μπλόκ διάγραμμα, θα εναποθέσετε και διάφορες σταθερές, ρουτίνες, αλγόριθμους, υπόVIs και καλώδια τα οποία θα μεταφέρουν τα δεδομένα από τον ένα Κόμβο στον άλλο, περιγράφοντας την διαγραμματική ροή του προγράμματός μας.

1.1.2 Το Πεδίο Παρουσίασης (Front Panel)

Το πεδίο παρουσίασης ενός VI είναι ένας συνδυασμός οργάνων χειρισμού και οργάνων ένδειξης αποτελώντας κατά αυτόν τον τρόπο το μέσο επικοινωνίας του προγράμματος με το χρήστη. Τα όργανα χειρισμού εξομοιώνουν συσκευές εισαγωγής δεδομένων τις οποίες μπορούμε να βρούμε σε οποιοδήποτε συμβατικό όργανο, όπως είναι οι μοχλοί και οι διακόπτες, και παρέχουν ένα μηχανισμό για την μεταφορά της εισόδου από το πεδίο παρουσίασης στο λογικό διάγραμμα. Τα εκάστοτε δεδομένα μπορεί να είναι είδη γραφημάτων και διαγραμμάτων, καθώς και αριθμητικά, Boolean και αλφαριθμητικά. Από την άλλη πλευρά, τα όργανα ένδειξης παρέχουν μηχανισμούς για την προβολή στο πεδίο παρουσίασης δεδομένων που προέρχονται από το λογικό διάγραμμα. Για αυτό το λόγο όταν χρησιμοποιούμε τον όρο *ελεγκτές (controls)* αναφερόμαστε σε «εισόδους» και όταν λέμε *ενδείκτες (indicators)* αναφερόμαστε σε «εξόδους».

Τοποθετούμε τους ελεγκτές και τους ενδείκτες στο πεδίο παρουσίασης επιλέγοντας τα και «αφήνοντας» τα από την παλέτα Controls Palette . Όταν επιλέγουμε έναν ελεγκτή (ή ενδείκτη) από την παλέτα και απελευθερώσουμε το κουμπί του ποντικιού, ο κέρσορας θα αλλάξει σε ένα εικονίδιο «χέρι» (Operating Tool), το οποίο μετά χρησιμοποιούμε για να μεταφέρουμε το αντικείμενο στην επιθυμητή θέση στο πεδίο παρουσίασης και να το «αφήσουμε» πιέζοντας ξανά το κουμπί του ποντικιού. Όταν ένα αντικείμενο τοποθετηθεί στο πεδίο παρουσίασης, μπορούμε εύκολα να ρυθμίσουμε το μέγεθος, το σχήμα και την θέση. Όπως βλέπουμε στην παρακάτω εικόνα, απεικονίζεται το Front Panel ενός εικονικού οργάνου του LabVIEW, το οποίο περιέχει έναν περιστροφικό διακόπτη (knob) για

την επιλογή καθυστέρησης, έναν ελεγκτή για την εισαγωγή δεδομένων και ένα γράφημα για την απεικόνιση της μετρούμενης τιμής.

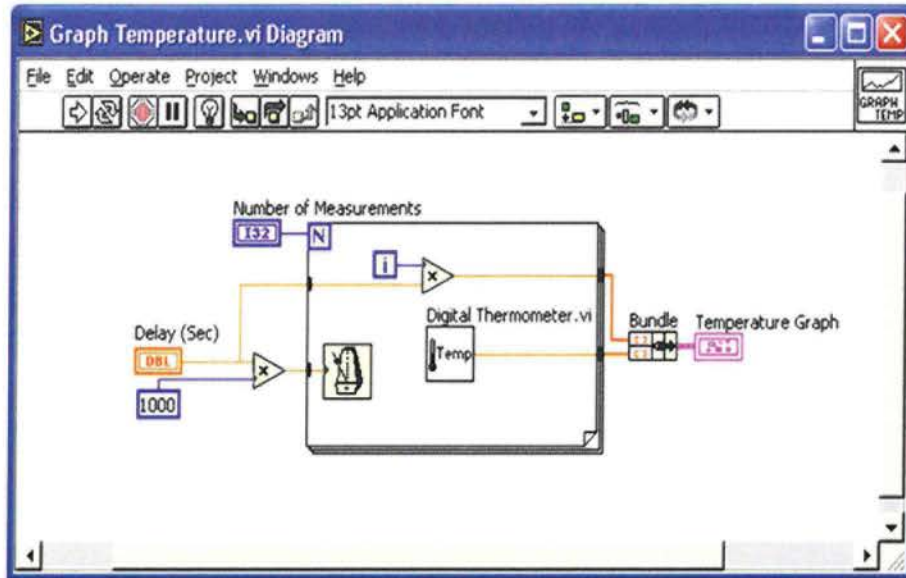


Εικόνα 1.1

Από τα τρία στοιχεία που παρουσιάζονται στο συγκεκριμένο πεδίο παρουσίασης, το γράφημα αποτελεί 'ενδείκτη' ενώ τα υπόλοιπα δύο αποτελούν 'ελεγκτές'.

1.1.3 Το Λογικό Διάγραμμα (Block Diagram)






Όλα τα γραφικά αντικείμενα που συνθέτουν το λογικό διάγραμμα, μαζί αποτελούν αυτό που συνήθως ονομάζουμε πηγαίο κώδικα. Το λογικό διάγραμμα (που οπτικά μοιάζει με ένα διάγραμμα ροής προγράμματος) ανταποκρίνεται στις γραμμές του κειμένου που βλέπουμε στις βασισμένες σε κείμενο γλώσσες προγραμματισμού. Στην πραγματικότητα, το λογικό διάγραμμα είναι ο πραγματικός εκτελέσιμος κώδικας. Χτίζεται καλωδιώνοντας μαζί αντικείμενα που εκτελούν συγκεκριμένες λειτουργίες. Στην παρακάτω εικόνα απεικονίζεται το λογικό διάγραμμα, ή ο πηγαίος κώδικας, της παραπάνω εφαρμογής.







Εικόνα 1.2

Παρατηρείστε ότι η απαιτούμενη καθυστέρηση (Delay) και ο αριθμός των μετρήσεων (Number of Measurements) που ορίζονται στο πεδίο παρουσίασης του VI, παρουσιάζονται ως εισοδοί στο λογικό διάγραμμα. Μέσω των δύο αυτών δεδομένων, ορίζεται η χρονική διάρκεια μεταξύ δύο διαδοχικών μετρήσεων όπως και ο αριθμός των συνολικών μετρήσεων που απαιτούνται. Στη συνέχεια, μέσω της υπορουτίνας 'Digital Thermometer.vi', η οποία είναι σχεδιασμένη για να ελέγχει σε συνεχή βάση τη θερμοκρασία, αποκτούνται οι τιμές της θερμοκρασίας στις καθορισμένες από το χρήστη χρονικές στιγμές. Το σύνολο των θερμοκρασιών καθώς και τα αντίστοιχα χρονικά σημεία παρουσιάζονται γραφικά μέσω του γραφήματος 'Temperature Graph', το οποίο και έχει οριστεί ως 'ενδείκτης' στο πεδίο παρουσίασης.

1.1.4 Εκτέλεση των Vis

Τα VIs συνήθως εκτελούνται από το **Front Panel**. Για να τρέξουμε ένα πρόγραμμα στο LabVIEW πατάμε το κουμπί **Run** , που βρίσκεται στην παλέτα στο πάνω μέρος του παραθύρου. Κατά την διάρκεια της εκτέλεσης του προγράμματος το *Run Button* αλλάζει σε μαύρο χρώμα . Επίσης για την εκτέλεση ή τον τερματισμό ενός προγράμματος υπάρχουν ακόμα τα κουμπιά **Run Continuously** , **Abort Execution** , και **Pause** .

Εάν το βέλος *Run* εμφανιστεί σαν σπασμένο βέλος , τότε σημαίνει πως υπάρχουν λάθη στο πρόγραμμα και αυτό δεν μπορεί να εκτελεστεί. Εάν κάνουμε click πάνω στο σπασμένο βέλος, τότε θα μας εμφανιστεί μια λίστα από λάθη που πρέπει να διορθώσουμε. Όταν διορθωθούν όλα τα λάθη τότε το βέλος επανέρχεται στην κατάσταση *Run* και το πρόγραμμα μπορεί να εκτελεστεί.

Μπορούμε να εκτελέσουμε ένα πρόγραμμα και από το **Block Diagram**, όπου στην παλέτα στο πάνω μέρος του παραθύρου υπάρχουν όλα τα παραπάνω κουμπιά όπως και τα ακόλουθα **Highlight Execution** , **Start Single Stepping**  και **Step Out** . Αυτό συμβαίνει συνήθως στη φάση σχεδιασμού του VI όταν θέλουμε να ελέγξουμε τη σωστή λειτουργία του.

1.2 Εργαλεία

Το LabVIEW περιέχει γραφικές παλέτες εργασίας σε μορφή menu με σκοπό την περαιτέρω βοήθεια στη δημιουργία και εκτέλεση των εικονικών οργάνων. Οι τρεις διαθέσιμες παλέτες είναι **Tools**, **Controls** και **Functions** κάθε μια από τις οποίες περιέχει τα αντίστοιχα εργαλεία. Τα εργαλεία με τη σειρά τους μπορούν να περιέχουν υπο-εργαλεία σχεδιασμένα για κάποια συγκεκριμένη χρήση

1.2.1 Tools Palette

Η παλέτα αυτή περιέχει εργαλεία που είναι κατάλληλα στην δημιουργία και εκτέλεση των εικονικών οργάνων. Εάν η tools palette δεν είναι ορατή, τότε για την εμφάνισή της επιλέγουμε *Show Tools Palette* από το Windows μενού. Η παλέτα εργασίας παρουσιάζεται στην παρακάτω εικόνα.



Εικόνα 1.3

Στο παράρτημα της εργασίας ακολουθεί η λειτουργικότητα των επιμέρους υπο-εργαλείων του Tools Pallet.

1.2.2 Controls Palette

Μπορούμε να εισάγουμε ελεγκτές και ενδείκτες στο πεδίο παρουσίασης από την Controls Palette κάνοντας δεξί-click. Κάθε επιλογή στην παλέτα εμπεριέχει μια υποεπιλογή με επιπλέον κουμπιά ελέγχου και ένδειξης, που σχετίζονται με την αρχική επιλογή σας. Επισημαίνεται ότι αν δεν είναι ορατό το μενού Controls, μπορείτε είτε να αναδυθείτε σε μια ελεύθερη περιοχή του πεδίου παρουσίασης και να εμφανιστεί, ή να επιλέξετε από το καταδυόμενο μενού Windows το Show Controls Palette. Επίσης πρέπει να τονιστεί ότι η παλέτα αυτή είναι ενεργή μόνο στο πεδίο παρουσίασης-Front Panel. Στην παρακάτω εικόνα απεικονίζεται η Controls Palette.

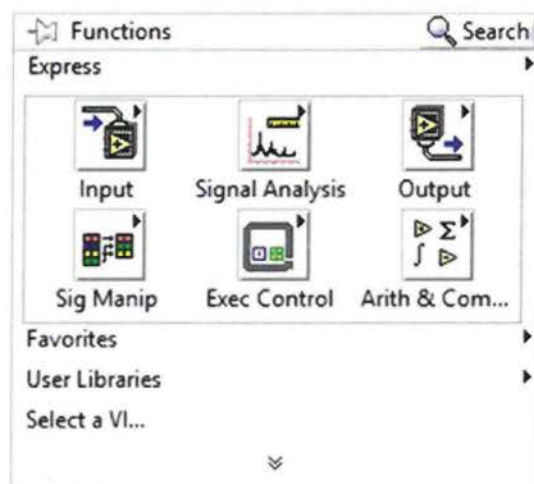


Εικόνα 1.4

Στο παράρτημα της εργασίας ακολουθεί αναλυτικός οδηγός κάθε εργαλείου του Control Palette.

1.2.3 Functions Palette

Η παλέτα εργασίας Function Palette είναι ενεργή μόνο όταν βρισκόμαστε στο Block Diagram. Κάθε επιλογή της παλέτας περιέχει έναν αριθμό από υπο-εργαλεία. Η Functions Palette απεικονίζεται παρακάτω.



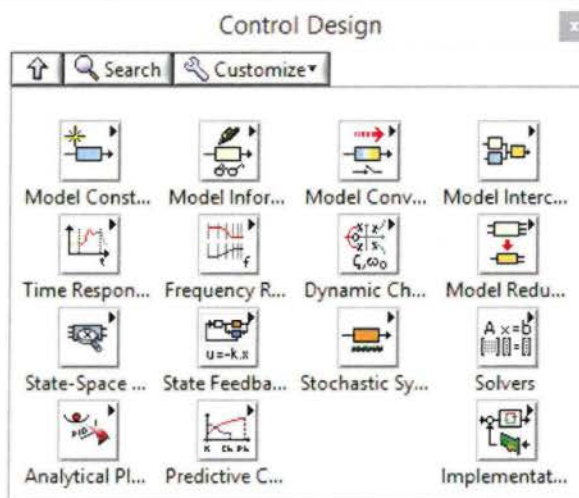
Εικόνα 1.5

Στο παράρτημα της πτυχιακής εργασίας παραθέτουμε τις έννοιες των πιο σημαντικών υπορουτινών.

Control Design Toolkit

Στην εργασία μας χρησιμοποιούμε πολύ συχνά την παλέτα Control Design Toolkit που βρίσκεται στο Function Palette, και την οποία έχουμε εγκαταστήσει μετά την εγκατάσταση του LabVIEW στον υπολογιστή. Ειδικά στις εφαρμογές αυτόματου ελέγχου όπως αυτή του Ανοιχτού και Βρόγχου, ο σχεδιασμός και η υλοποίηση τους γίνεται μέσω αυτής της βιβλιοθήκης, που την βρίσκουμε στο Function Palette. Το Control Design Toolkit υποστηρίζει γραμμικά, χρονικά αμετάβλητα, συνεχή και διακριτά μοντέλα που μπορούν να αναπαρασταθούν ως μοντέλα συνάρτησης μεταφοράς, χώρου κατάστασης ή κέρδους-πόλων-μηδενικών. Με τα εργαλεία που παρέχει μπορούμε να πραγματοποιήσουμε μετατροπές ανάμεσα στις διάφορες μορφές των μοντέλων καθώς και να κάνουμε σύνδεση μεταξύ των μοντέλων σε σειρά, παράλληλα, με ανάδραση κ.α. Μπορούμε να κάνουμε ανάλυση της χρονικής απόκρισης, της απόκρισης συχνότητας και των χαρακτηριστικών ενός συστήματος όπως είναι η ευστάθεια, το κέρδος και ο συντελεστής απόσβεσης, χρησιμοποιώντας ενσωματωμένα VIs (Virtual Instruments-Εικονικά Όργανα) όπως είναι το CD Bode.vi, CD Nyquist.vi και το CD Root Locus.vi.

Το Control Design Toolkit παρέχει το διαδραστικό εργαλείο Control Design Assistant που είναι διαθέσιμο από το Tools / Control Design Toolkit και μια βιβλιοθήκη με VIs, για τη μελέτη και σχεδίαση ενός συστήματος αυτομάτου ελέγχου. Όταν εγκαθίσταται το Control Design Toolkit, η Control Design Palette είναι διαθέσιμη από την παλέτα Functions και φαίνεται στο σχήμα που ακολουθεί.



Εικόνα 1.6

Ακολουθούν μερικοί από τους βασικότερους τύπους των δομικών στοιχείων στην εργαλειοθήκη:

• *Model Construction palette, με τις παρακάτω συναρτήσεις και υποπαλέτες:*

- Construct State-Space Model
- Construct Transfer Function Model
- Construct Zero-Pole-Gain Model
- Construct Random Model
- Construct Special Model
- Draw Transfer Function Equation
- Draw Zero-Pole-Gain Equation
- Read Model From File
- Write Model From File
- Model Information palette

• *Model Conversion palette, με τις παρακάτω συναρτήσεις:*

- Convert to State-Space Model
- Convert to Transfer Function Model
- Convert to Zero-Pole-Gain Model
- Convert Delay with Pade Approximation
- Convert Delay to Poles at Origin
- Convert Continuous to Discrete (με διάφορες μεθόδους όπως Tustin, zero order hold κ.α)
- Convert Discrete to Discrete
- Convert Discrete to Continuous
- Convert Control Design to Simulation
- Convert Simulation to Control Design

• *Model Interconnection palette, με τις παρακάτω συναρτήσεις:*

- Serial
- Parallel
- Feedback
- Append

• *Time Response palette, με τις παρακάτω συναρτήσεις:*

- Step Response
- Impulse Response
- Initial Response
- Linear Simulation
- Get Time Response Data

• *Frequency Response palette, με τις παρακάτω συναρτήσεις: ο Bode Nyquist*

- Nichols
- Singular Values
- All Margins
- Gain and Phase Margin
- Evaluate at Frequency
- Bandwidth
- Get Frequency Response Data

• *The Dynamic Characteristics palette, με τις παρακάτω συναρτήσεις:*

- Root Locus
- Pole-Zero Map
- Damping Ratio and Natural Frequency
- DC Gain
- Stability
- Norm
- Covariance Response
- Total Delay
- Distribute Delay
- Parametric Time Response

• *The State Space Model Analysis palette, με τις παρακάτω συναρτήσεις:*

- Controllability Matrix
- Observability Matrix
- Grammians
- Canonical State-Space Realization
- Balance State-Space Model (Diagonal)

- Balance State-Space Model (Grammians)
- Controllability Staircase
- Observability Staircase
- State Similarity Transform

1.2.4 Μενού Pop-Up

Σχεδόν όλα τα αντικείμενα που χρησιμοποιούμε για την κατασκευή ενός VI έχουν Pop-Up Μενού. Τα μενού εμφανίζονται όταν ο κέρσορας βρίσκεται πάνω από το αντίστοιχο αντικείμενο ή παράθυρο και πατήσουμε το δεξί ποντίκι.

Μενού Pull-Down

Η μπάρα στην κορυφή της οθόνης του LabVIEW περιέχει τα ακόλουθα Pull-Down μενού:

- **Μενού Αρχείων (File Menu):** Χρησιμοποιούμε τις επιλογές σε αυτό το μενού κυρίως για να σώσουμε, ανοίξουμε, κλείσουμε, εκτυπώσουμε VIs.
- **Μενού Αλλαγών (Edit Menu):**
 - Remove Broken Wires:* Σβήνει αχρηστες ή ελαττωματικές συνδέσεις
 - Cut / Copy / Paste:* Επιλογές με την ίδια ακριβώς λειτουργία όπως και σε άλλα προγράμματα windows
 - Make Current Values Default:* Θέτει τις τωρινές τιμές ως αρχικές τιμές λειτουργίας στο μέλλον
- **Μενού Λειτουργίας (Operate Menu):**
 - Data Logging:* Επιλογές για καταγραφή των δεδομένων,
 - Connect to remote Panel:* η δυνατότητα να συνδεθούμε σε άλλες εφαρμογές του LabVIEW από απόσταση
- **Μενού Εργαλείων (Tools):**
 - MAX:* Το πρόγραμμα αυτό μας βοηθά να προσδιορίσουμε τα χαρακτηριστικά των καναλιών συλλογής Μετρήσεων,
 - Instrumentation:* για την εύρεση Instrument Drivers,
 - Build Executable:* αν έχετε την έκδοση Professional, η επιλογή αυτή δημιουργεί αυτονομες εφαρμογές
 - Show Profile Window:* Εδώ βλέπουμε αν το πρόγραμμα είναι βέλτιστα γραμμένο
- **Μενού Παραθύρων και Εμφάνισης (Windows και View Menu):**

Show VI Info...: Μας δίνει πληροφορίες για τον αντίστοιχο VI.

Show Functions / Tools Palette: Εμφανίζει τις διάφορες παλέτες

Tile Left and Right: Τα παράθυρα του Front Panel και Διαγράμματος μοιράζονται την οθόνη

- **Μενού Βοήθειας(Help Menu):**

Simple Help: Συνοπτική Βοήθεια On-Line

Online Reference: Πλήρης Βοήθεια On-Line

Internet Links: Σύνδεση με το Web Site της National Instruments

1.3 Βιβλιοθήκες από Vis (VI Libraries)

Μπορείτε να φορτώσετε και να καταχωρήσετε VIs σε μία ειδική δομή αρχείου στο labVIEW την οποία ονομάζουμε VI Library (Βιβλιοθήκη VIs). Συνήθως αυτό το αρχείο έχει την προσθήκη .llb. Τα πλεονεκτήματα χρήσης βιβλιοθηκών είναι αρκετά:

- Με τα VI libraries, μπορείτε να χρησιμοποιήσετε μέχρι 255 χαρακτήρες για να ονομάσετε τα VIs σας.
- Τα VI libraries συμπιέζουν τα VIs και γλυτώνουν χώρο στο σκληρό δίσκο.
- Έχοντας πολλά VIs μέσα σε ένα μοναδικό αρχείο είναι πιο εύκολη η μεταφορά αρχείων μεταξύ υπολογιστών.

Σημείωση:

- Τα VI libraries δεν είναι ιεραρχικά. Δηλαδή, δεν είναι δυνατόν να υπάρχει ένα VI Library μέσα σε άλλο VI library.
- Η καταχώρηση VIs απευθείας στο δίσκο είναι πιο γρήγορη από ότι μέσα σε ένα VI Library.

1.3.1 Φορτώνοντας Vis

Φορτώνετε έναν VI στη μνήμη επιλέγοντας **Open** από το μενού **File**. Βιβλιοθήκες από VIs καθώς και Vis εύκολα αναγνωρίζονται στο διαλογικό παράθυρο που εμφανίζεται, καθώς οι Βιβλιοθήκες VI μοιάζουν με φάκελλους με την εικόνα VI γραμμένη επάνω τους, ενώ τα VIs διατηρούν την προσθήκη .vi. Ανοίγετε ένα VI Library, με τον ίδιο τρόπο που θα ανοίγατε έναν οποιοδήποτε φάκελλο. Είτε επιλέγοντας τον, και πατώντας **OK** είτε πατώντας διπλά το ποντίκι. Καθώς

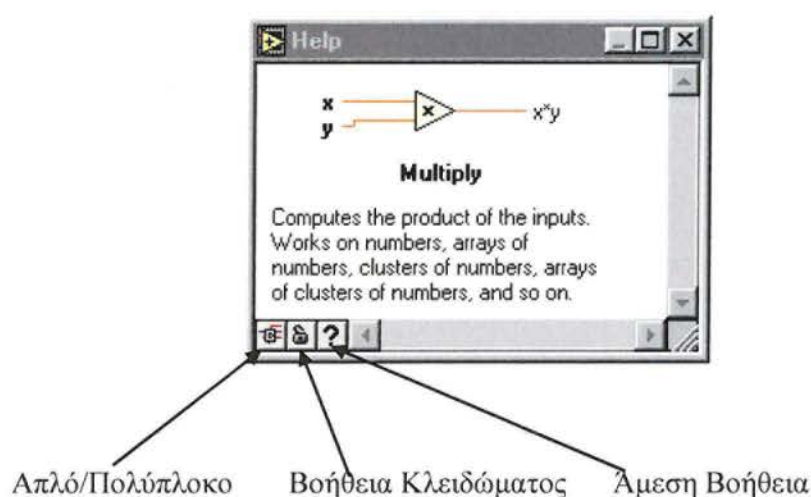
φορτώνεται ο VI, ένα παράθυρο εμφανίζεται στην οθόνη και μας πληροφορεί για τους υπόVIs που φορτώνονται μαζί με το κύριο πρόγραμμα στην μνήμη. Εάν το LabVIEW δεν μπορεί αυτόματα να βρει ένα συγκεκριμένο υπόVI, ψάχνει σε όλους τους φάκελλους που έχουμε προσδιορίσει στο μονοπάτι ανίχνευσης **VI Search Path** (**Tools** menu » **Options** » **Paths**). Μπορείτε να επιλέξετε να αγνοήσετε το υπόVI, πατώντας **Ignore SubVI**, ή μπορείτε να πατήσετε **Browse** και να συνεχίσετε την έρευνα. Η καταχώρηση ενός VI μπορεί να γίνει σε ένα φάκελλο ή Βιβλιοθήκη από VIs όπως είπαμε, επιλέγοντας **Save, Save As..., Save a Copy As...** από το μενού **File**. Για να δημιουργήσετε μία καινούργια βιβλιοθήκη από VIs επιλέξατε **Save As...** και πατήστε το κουμπί **New LLB**. Στο διαλογικό παράθυρο που εμφανίζεται, δίνετε ένα όνομα της αρεσκείας σας στην νέα βιβλιοθήκη και ύστερα πατάτε **Create**. Δεν χρειάζεται να γράψετε την προσθήκη **.llb**, το LabVIEW την τοποθετεί αυτόματα. Μπορείτε να διαγράψετε VIs από μία βιβλιοθήκη με την βοήθεια του **LLB Manager** από το μενού **Tools**.

1.3.2 Μεταφορά VIs σε άλλες πλατφόρμες

Μπορείτε να μεταφέρετε VIs από ένα λειτουργικό σύστημα σε άλλο (για παράδειγμα από LabVIEW for Macintosh σε LabVIEW for Windows). Το LabVIEW αυτόματα μεταφράζει τον κώδικα στην νέα πλατφόρμα. Η χρήση Βιβλιοθηκών Vis (VI Libraries) διευκολύνει την διαδικασία, καθώς μπορούμε να μεταφέρουμε πολλά VIs μαζί, ενώ μπορούμε να διατηρήσουμε μεγάλα ονόματα για τα αρχεία μας. Μπορείτε να χρησιμοποιήσετε οποιαδήποτε μέθοδο για μεταφορά των αρχείων ανάμεσα σε πλατφόρμες. Οι πιο συνηθισμένες είναι μέσω FTP, Z- ή Xmodem. Τέτοιες μεταφορές μέσα από το δίκτυο, είναι πολύ διαδεδομένες καθώς δεν χρειάζεται επιπλέον λογισμικό μετάφρασης. Εάν χρησιμοποιήσετε μαγνητικά μέσα, θα χρειαστείτε κάποιο από τα ακόλουθα προγράμματα : Windows— MacDisk , TransferPro για Macintosh αρχεία σε PC format και αντίστροφα. Sun—PC File System (PCFS) για μετατροπή αρχείων από PC Format σε Solaris και αντίστροφα. HP-UX—H εντολή doscp φορτώνει PC δισκέττες και αντιγράφει τα αρχεία τους. Macintosh—DOS Mounter και Apple File Exchange. Σημείωση : Μερικά VIs όντας εξαρτημένα από το συγκεκριμένο λειτουργικό σύστημα, δεν μπορούν να μεταφερθούν. Τέτοια παραδείγματα είναι Code Interface Nodes (CINs), ActiveX AppleEvents.

1.4 Η χρήση βοήθειας στο LabVIEW

Το LabVIEW, παρέχει έναν online οδηγό παροχής βοήθειας για όποια λειτουργία θέλουμε να πραγματοποιήσουμε. Η λειτουργία αυτή ενεργοποιείται από το Help Menu επιλέγοντας Show Help. Η ενεργοποίηση της λειτουργίας αυτής εμφανίζει ένα παράθυρο βοήθειας δείχνοντας ένα εικονίδιο που σχετίζεται με το επιλεγμένο αντικείμενο και δείχνει τα καλώδια που είναι ενωμένα σε κάθε ακροδέκτη. Ένα τυπικό παράθυρο βοήθειας απεικονίζεται παρακάτω, στην παρακάτω εικόνα.



Εικόνα 1.7

Επιλογή Απλής / Αναλυτικής Βοήθειας(Simple/Complex Diagram Help). Πατήστε για να διαλέξετε μεταξύ των δύο διαφορετικών επιλογών. Η Απλή βοήθεια δίνει έμφαση στις απαραίτητες συνδέσεις. Οι υπόλοιποι ακροδέκτες απλά φαίνονται, πληροφορώντας τον χρήστη ότι υπάρχουν και άλλες συνδέσεις. Η αναλυτική βοήθεια παρουσιάζει όλους τους ακροδέκτες. Η επιλογή της Αναλυτικής βοήθειας γίνεται και από το μενού **Help**.

Κλειδώμα του παράθυρου της βοήθειας. Με αυτόν τον τρόπο κλειδώνετε το περιεχόμενο του παράθυρου της βοήθειας πάνω στο συγκεκριμένο αντικείμενο, έτσι ώστε να μην αλλάζει παρότι μετακινείτε το ποντίκι πάνω από άλλα αντικείμενα. Ξεκλειδώνετε το παράθυρο είτε μέσα από το μενού **Help** είτε πατώντας ξανά την επιλογή **Lock** στο κάτω μέρος του παράθυρου της βοήθειας.

Βοήθεια On-line (άμεση βοήθεια). Με αυτόν τον τρόπο έχετε πρόσβαση On-Line στα εγχειρίδια χρήσης του LabVIEW και τις αναλυτικές περιγραφές για κάθε αντικείμενο.

ΚΕΦΑΛΑΙΟ 2: ΕΦΑΡΜΟΓΕΣ ΣΤΟ LABVIEW

2.1 ΘΕΡΜΟΜΕΤΡΟ

Στο συγκεκριμένο παράδειγμα θα δημιουργήσουμε ένα εικονόργανο το οποίο θα μετράει την θερμοκρασία.

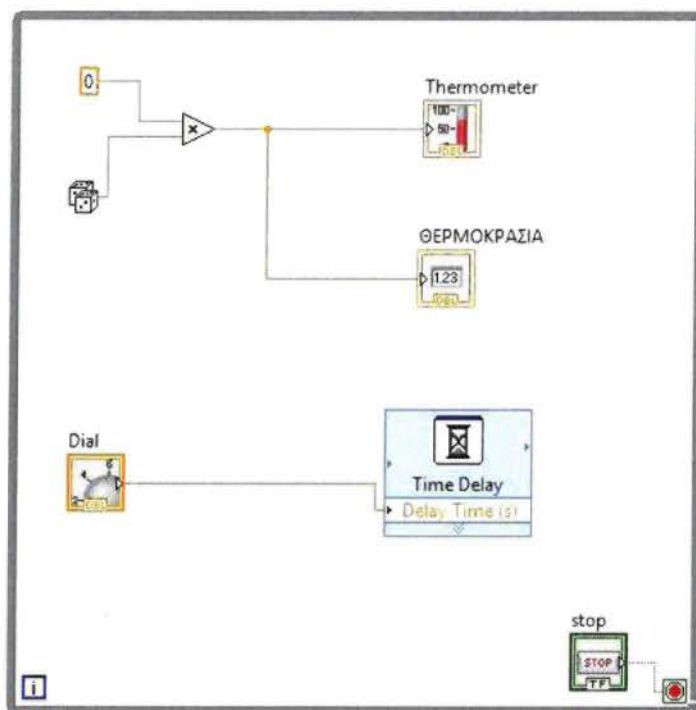
Τα βήματα που ακολουθούμε είναι τα εξής:

1. Από την οθόνη “Getting Started” ανοίγουμε ένα νέο “Blank VI”.
2. Πηγαίνουμε στο front panel (Ctrl+E).
3. Πατάμε δεξί-κλικ οπουδήποτε στο front panel για να εμφανιστεί η παλέτα Controls.
4. Από την παλέτα Controls πηγαίνουμε Modern>> Numeric και επιλέγουμε τα όργανα Numeric Indicator, Thermometer και Dial.
5. Αλλάζουμε όνομα στο Numeric σε ΘΕΡΜΟΚΡΑΣΙΑ.
6. Από την παλέτα Controls πηγαίνουμε Modern>> Boolean και τοποθετούμε ένα Stop Button.
7. Πηγαίνουμε στο block diagram (Ctrl+E).
8. Πατάμε δεξί-κλικ οπουδήποτε στο block diagram και για να εμφανιστεί η παλέτα Functions.
9. Από την παλέτα Functions πηγαίνουμε στο Programming>> Structures >> While Loop , και περικυκλώνουμε όλα τα στοιχεία στο block diagram.
10. Από την παλέτα Functions πηγαίνουμε στο Mathematics>> Numeric>> Multiply, και το τοποθετούμε μέσα στο While Loop.
11. Στην είσοδο x του Multiply κάνουμε δεξί-click Create και Constant.
12. Από την παλέτα Functions πηγαίνουμε στο Mathematics>> Numeric>> Random Number (0-1), και το τοποθετούμε μέσα στο While Loop. Αυτό το νί παράγει τυχαίους αριθμούς από 0 σε 1.
13. Ενώνουμε το Random Number (0-1) με την είσοδο y του Multiply.
14. Ενώνουμε την έξοδο $x*y$ του Multiply στην είσοδο του Thermometer.
15. Ενώνουμε την είσοδο του ΘΕΡΜΟΚΡΑΣΙΑ στην έξοδο $x*y$ του Multiply.
16. Από την παλέτα Functions πηγαίνουμε στο Express>> Execution Control>> Time Delay, και το τοποθετούμε μέσα στο While Loop. Αυτό το νί θα μας δώσει την απαραίτητη καθυστέρηση που χρειαζόμαστε.

17. Ενώνουμε την έξοδο του Dial με την είσοδο του Time Delay.

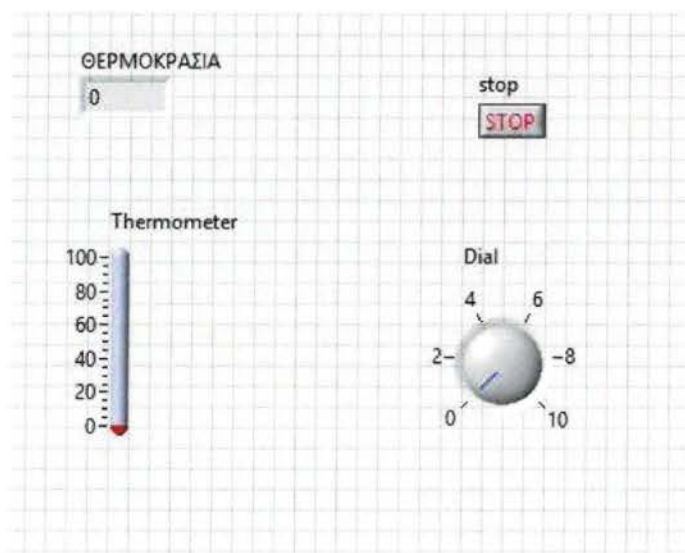
18. Ενώνουμε το STOP button με το Loop Condition.

Στο σχήμα απεικονίζεται το block diagram μετά την ολοκλήρωση των παραπάνω βημάτων.



Εικόνα 2.1.1

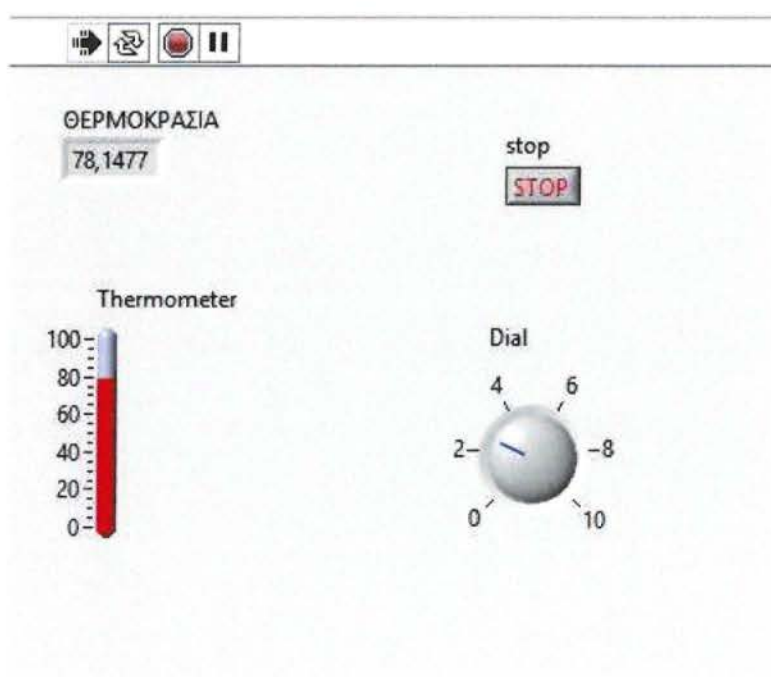
Στο παρακάτω σχήμα απεικονίζεται το Front Panel.



Εικόνα 2.1.2

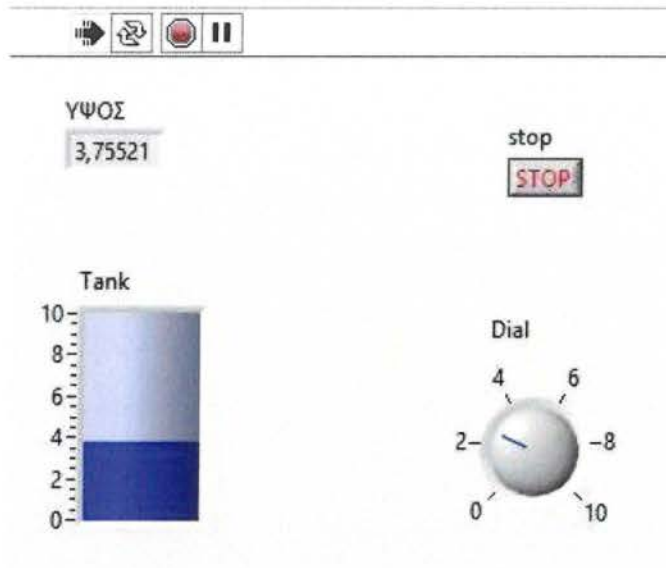
Με το συγκεκριμένο πρόγραμμα προσομοιώνουμε έναν αισθητήρα θερμοκρασίας. Μεταβάλλοντας την τιμή του ποτενσιόμετρου Dial μεταβάλλουμε και τον χρόνο μέτρησης της θερμοκρασίας. Αν κάνουμε διπλό-click πάνω στο εικονίδιο Time Delay εμφανίζεται η καθυστέρηση την οποία θα θέλαμε. Το αφήνουμε στο ένα δευτερόλεπτο. Στην είσοδο x του Multiply δίνουμε την τιμή 100. Αυτή η τιμή είναι η ανώτερη που μπορεί να φτάσει η θερμοκρασία μας.

Πατώντας το κουμπί Run στο Front Panel και μεταβάλλοντας όπως θέλουμε το χρόνο καθυστέρησης βλέπουμε πως μεταβάλλεται η θερμοκρασία σε τυχαίες χρονικές στιγμές.



Εικόνα 2.1.3

Πατώντας το κουμπί STOP σταματάμε την εφαρμογή. Την συγκεκριμένη εφαρμογή μπορούμε να την χρησιμοποιήσουμε και ως πρότυπο προσομοιώσεις και άλλων αισθητηρίων. Χαρακτηριστικό παράδειγμα είναι η προσομοίωση μέτρησης του ύψους μιας δεξαμενής. Αντί λοιπόν για το Thermometer μπορούμε να επιλέξουμε πάλι από την παλέτα Numeric το Tank και το αποτέλεσμα μας θα είναι:



Εικόνα 2.1.4

2.2 ΒΟΛΤΟΜΕΤΡΟ

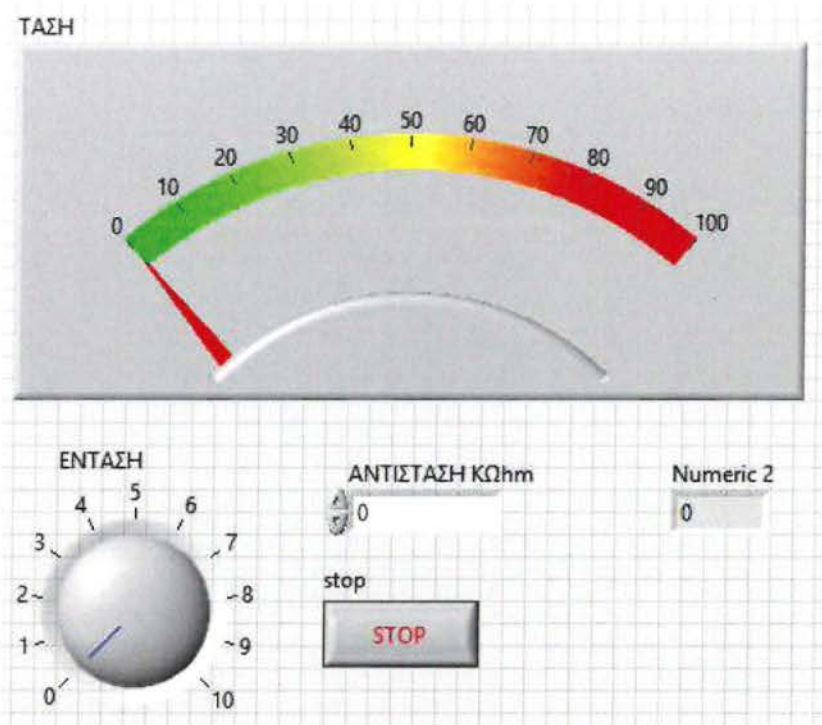
Στη δραστηριότητα αυτή θα κατασκευάσουμε εικονόργανο που θα προσομοιώνει τη μέτρηση τάσης στα άκρα αντίστασης στην οποία το ρεύμα που τι διαρρέει θα μεταβάλλεται από ποτενσιόμετρο στην οθόνη του μμικού παραθύρου καθώς και η τιμή της αντίστασης θα δίνεται από τον χρήστη. Το εικονόργανο θα τερματίζεται από πάτημα σε πλήκτρο STOP.

Τα βήματα που ακολουθούμε είναι τα εξής:

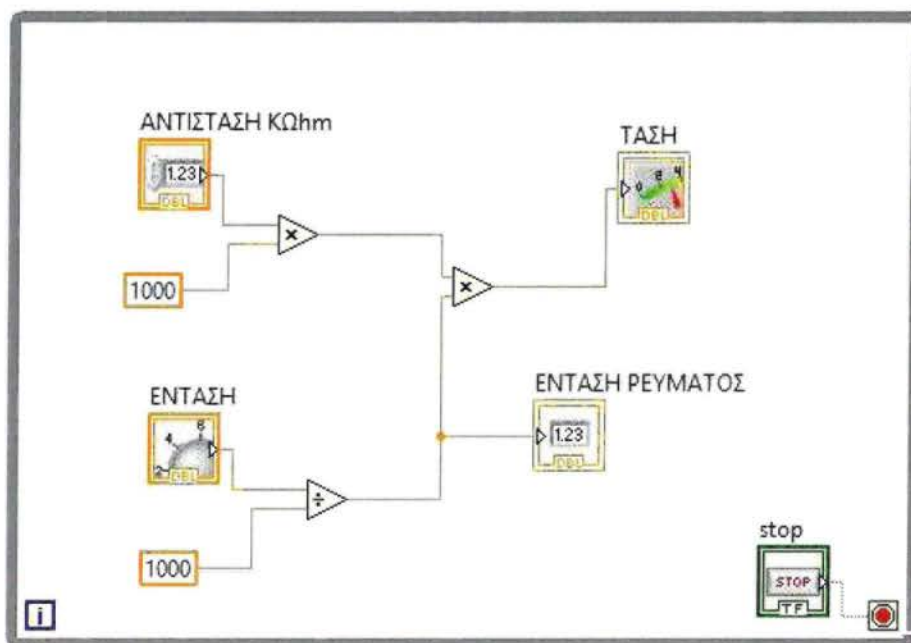
1. Από την οθόνη "Getting Started" ανοίγουμε ένα νέο "Blank VI".
2. Πηγαίνουμε στο block diagram (Ctrl+E).
3. Πατάμε δεξί-κλικ οπουδήποτε στο block diagram και για να εμφανιστεί η παλέτα Functions.
4. Από την παλέτα Functions πηγαίνουμε στο Programming>> Structures >> While Loop , και ανοίγουμε ένα πλαίσιο στο block diagram.
5. Πηγαίνουμε στο front panel (Ctrl+E).
6. Πατάμε δεξί-κλικ οπουδήποτε στο front panel για να εμφανιστεί η παλέτα Controls.
7. Από παλέτα Controls πηγαίνουμε Modern>>Boolean, και επιλέγουμε το Stop Button.

8. Από την παλέτα Controls πηγαίνουμε Modern>>Numeric>>Dial επιλέγουμε και τοποθετούμε ένα ποτενσιόμετρο και του δίνουμε το όνομα «ΕΝΤΑΣΗ».
9. Από την παλέτα Controls πηγαίνουμε Modern>>Numeric επιλέγουμε τα Numeric Control στο οποίο δίνουμε το όνομα «ΑΝΤΙΣΤΑΣΗ ΚΩhm» και Numeric Indicator το οποίο δίνουμε το όνομα «ΕΝΤΑΣΗ ΡΕΥΜΑΤΟΣ».
10. Από την παλέτα Controls πηγαίνουμε Modern>>Numeric>>Meter επιλέγουμε και τοποθετούμε ένα βολτόμετρο και του δίνουμε το όνομα «ΤΑΣΗ».
11. Διαμορφώνουμε το Front Panel έτσι ώστε να είναι ευδιάκριτα τα όργανά μας.
12. Πηγαίνουμε στο block diagram (Ctrl+E).
13. Από την παλέτα Functions πηγαίνουμε στο Mathematics>> Numeric και επιλέγουμε δύο Multiply για πολλαπλασιασμό και ένα Divide για διαίρεση, και τα τοποθετούμε μέσα στο While Loop.
14. Από την παλέτα Functions πηγαίνουμε στο Mathematics>> Numeric επιλέγουμε δύο DBL Numeric Constant και τα τοποθετούμε μέσα στο While Loop.
15. Συνδέουμε το STOP button με το Loop Condition.
16. Συνδέουμε την έξοδο του «ΑΝΤΙΣΤΑΣΗ ΚΩhm» με την είσοδο x του ενός Multiply και την έξοδο του ενός DBL Numeric Constant με την είσοδο y του Multiply δίνοντας του την τιμή 1000.
17. Συνδέουμε την έξοδο $x*y$ του Multiply με την είσοδο x του άλλου Multiply.
18. Συνδέουμε την έξοδο του «ΕΝΤΑΣΗ» με την είσοδο x του ενός Divide και την έξοδο του άλλου DBL Numeric Constant με την είσοδο y του Divide δίνοντας του και σε αυτό την τιμή 1000.
19. Συνδέουμε την έξοδο x/y του Divide με την είσοδο y του Multiply και την είσοδο του «ΕΝΤΑΣΗ ΡΕΥΜΑΤΟΣ».
20. Τέλος συνδέουμε την έξοδο $x*y$ του Multiply με την είσοδο του «ΤΑΣΗ».

Στο σχήμα απεικονίζεται το Block Diagram και το Front Panel μετά την ολοκλήρωση των παραπάνω βημάτων:

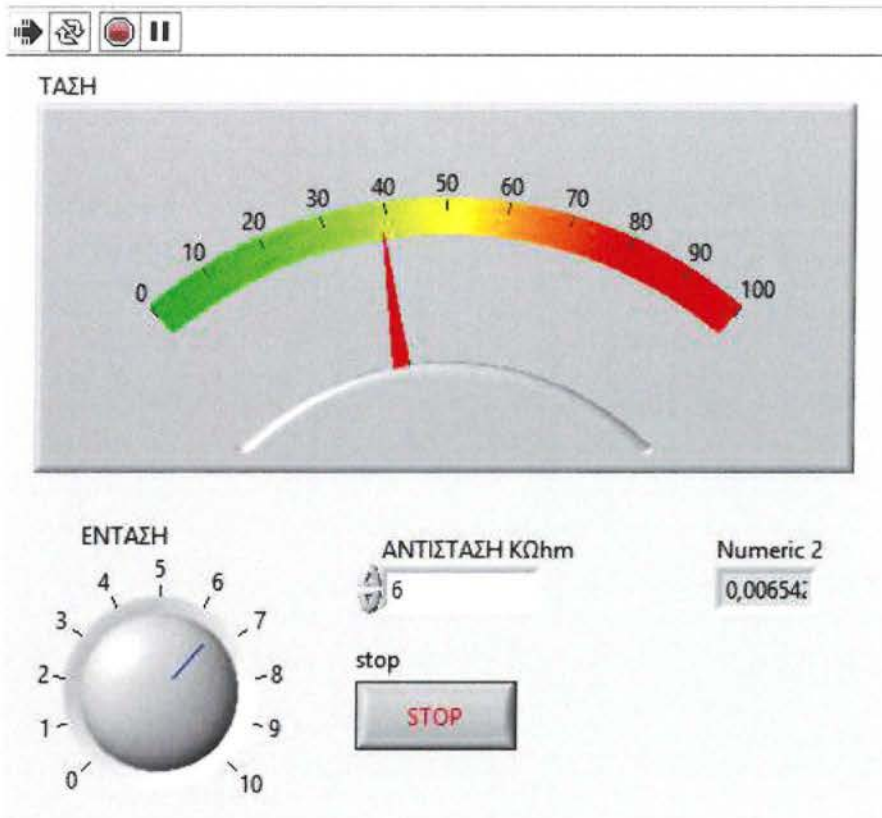


Εικόνα 2.2.1



Εικόνα 2.2.2

Πατώντας το πλήκτρο Run στο Front Panel και μεταβάλλοντας τις τιμές βλέπουμε το αποτέλεσμα μας.

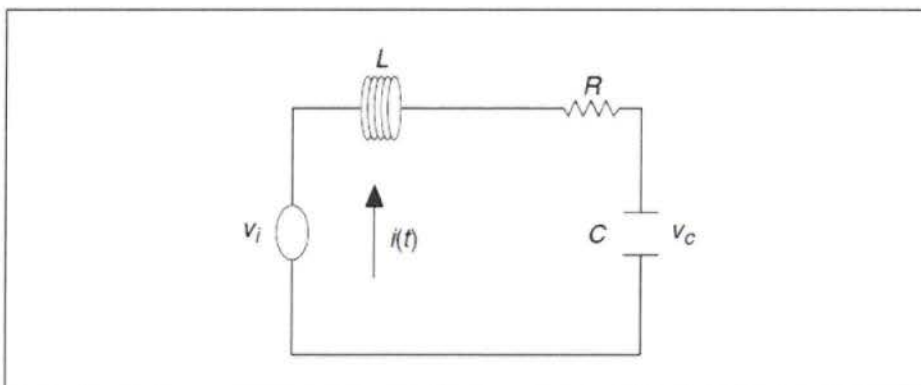


Εικόνα 2.2.3

Όπως βλέπουμε η αντίσταση μας η οποία μετριέται σε ΚΩhm (πολλαπλασιαζόμενη με το 1000), διαρρέεται από ρεύμα έντασης έως δέκα mile Ampère, (διαίρεται με το 1000), το οποίο εμφανίζεται στο Numeric 2.

2.3 Κύκλωμα RLC σε σειρά

Στην εφαρμογή αυτή υλοποιούμε ένα κύκλωμα RLC σε σειρά όπως φαίνεται στη παρακάτω εικόνα:



Η συνάρτηση μεταφοράς ανάμεσα στην τάση εισόδου V_i και την τάση του πυκνωτή V_c (τάση εξόδου) είναι η εξίσωση:

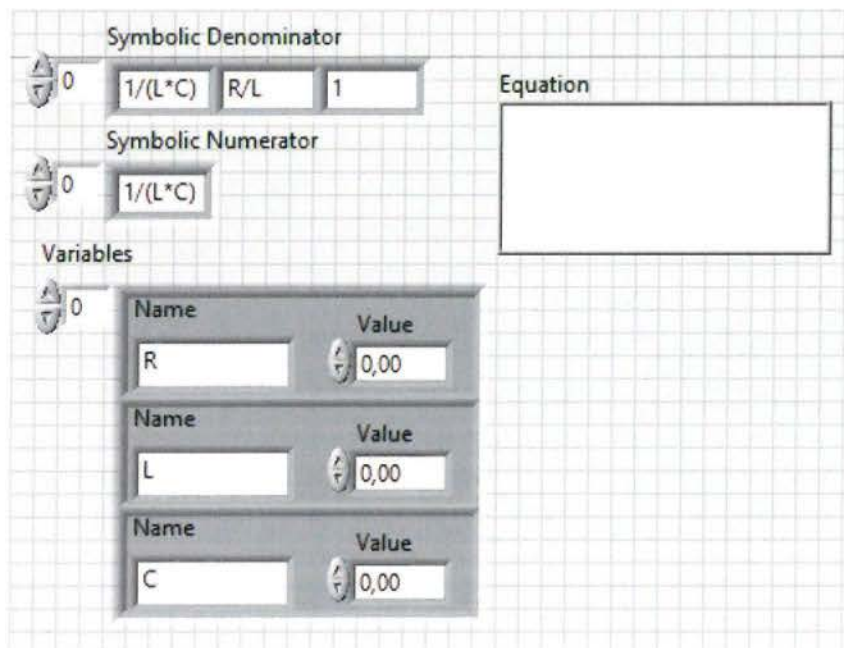
$$H(s) = \frac{1/LC}{s^2 + \frac{Rs}{L} + \frac{1}{LC}}$$

Τα βήματα που ακολουθούμε είναι τα εξής:

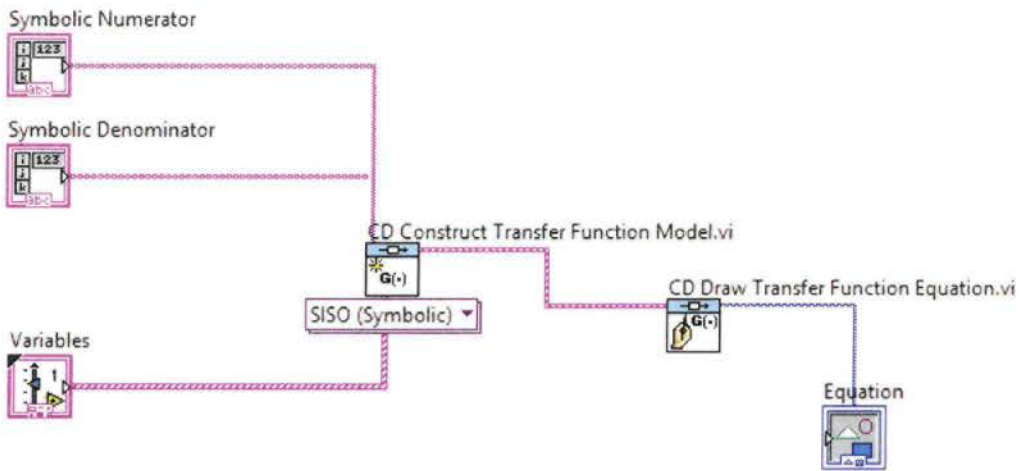
1. Από την οθόνη “Getting Started” ανοίγουμε ένα νέο “Blank VI”.
2. Πηγαίνουμε στο block diagram (Ctrl+E).
3. Πατάμε δεξί-κλικ οπουδήποτε στο block diagram για να εμφανιστεί η παλέτα Functions.
4. Από την παλέτα Functions πηγαίνουμε στο Control Design & Simulation >> Control Design >> Model Construction >> CD Construct Transfer Function Model.vi, επιλέγουμε και τοποθετούμε στο block diagram. Από το μενού που υπάρχει κάτω από το vi επιλέγουμε Single-Input Single-Output (Symbolic).
5. Δεξί-κλικ πάνω στο CD Construct Transfer Function Model.vi >> Visible Items >> Label, για να εμφανιστεί η ετικέτα με το όνομα του vi.
6. Δεξί-κλικ στην είσοδο Symbolic Numerator του CD Construct Transfer Function Model.vi και επιλέξτε Create >> Control.
7. Δεξί-κλικ στην είσοδο Symbolic Denominator του CD Construct Transfer Function Model.vi και επιλέγουμε Create >> Control.
8. Δεξί-κλικ στην είσοδο Variables του CD Construct Transfer Function Model.vi και επιλέγουμε Create >> Control.
9. Για να αλλάξουν μορφή τα Control που μόλις δημιουργήσαμε και να φαίνονται πιο μικρά, κάντε ξεχωριστά πάνω στο καθένα δεξί-κλικ και ξεκλικάρετε την επιλογή View As Icon.
10. Από την παλέτα Functions πηγαίνουμε στο Control Design & Simulation >> Control Design >> Model Construction >> CD Draw Transfer Function Equation.vi, επιλέγουμε και τοποθετούμε στο block diagram.

11. Συνδέουμε την έξοδο Transfer Function Model του CD Construct Transfer Function Model.vi στην είσοδο Transfer Function Model του CD Draw Transfer Function Equation.vi.
12. Δεξί-κλικ στην έξοδο Equation του CD Draw Transfer Function Equation.vi και επιλέγουμε Create >> Indicator.
13. Διπλό-κλικ στην ετικέτα του Indicator και την ονομάζουμε σε Plant Equation.
14. Επιλέξτε File >> Save As.. και σώστε το VI με το όνομα “RLC Series”.
15. Αλλάξτε στο front panel του VI (Ctrl+E).
16. Δεξί-κλικ στο κελί Value, μέσα στο Variables, πατήστε Display Format..., και επιλέξτε Automatic Formatting. Αυτό επιτρέπει να φαίνονται οι μικροί αριθμοί σωστά.
17. Δώστε στα Symbolic Numerator και Symbolic Denominator τις τιμές τους σύμφωνα με την εξίσωση και τα δεδομένα που δίνονται στην αρχή του παραδείγματος. Συμβουλευτείτε το σχήμα.
18. Ορίστε τις τιμές ως προεπιλογή. Επιλέξτε Edit >> Make Current Values Default.

Μετά τα παραπάνω βήματα το Front Panel και Block Diagram έχουν ως εξής:



Εικόνα 2.3.1



Εικόνα 2.3.2

Έστω ότι οι τιμές της αντίστασης είναι $R=20\Omega$, του πηνίου $L=50\text{mH}$ και του πυκνωτή $C=10\mu\text{F}$. Τότε το αποτέλεσμα στο Front Panel θα είναι :

Symbolic Denominator

Symbolic Numerator

Variables

| Name | Value |
|------|---------|
| R | 20,00 |
| L | 0,05 |
| C | 1,00E-5 |

Equation

$$\frac{2E+6}{s^2 + 400s + 2E+6}$$

Εικόνα 2.3.3

2.4 Δομές Επανάληψης

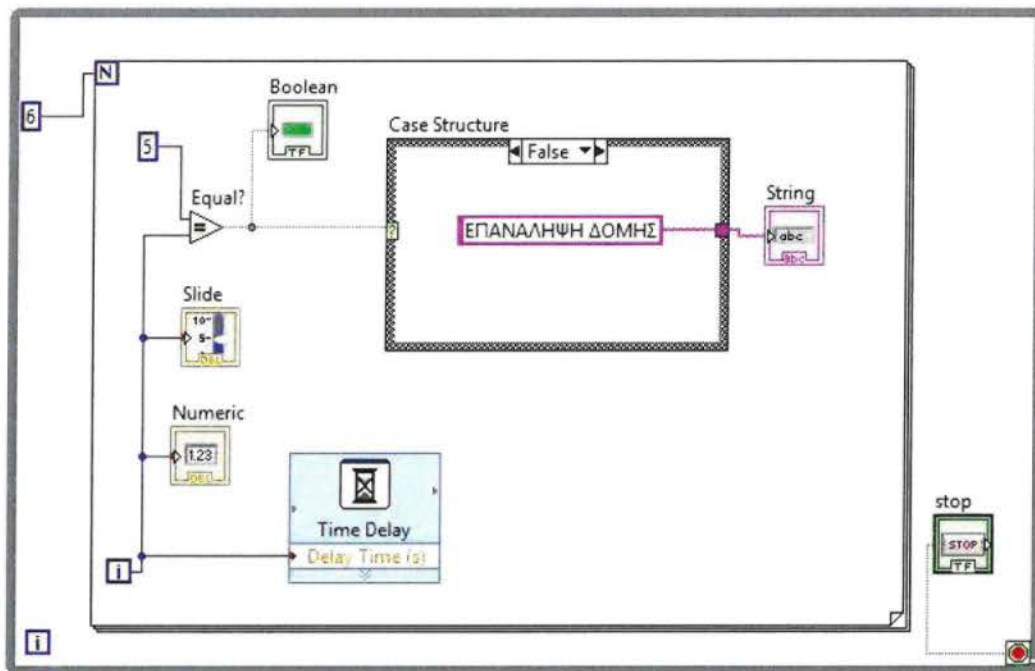
Όλες οι γλώσσες προγραμματισμού χρησιμοποιούν δομές επανάληψης για την πραγματοποίηση των εφαρμογών τους. Αυτές φυσικά δεν λείπουν και από την πλατφόρμα του LabVIEW για την απλούστευση των εφαρμογών. Σε αυτό το παράδειγμα θα σας δείξουμε τις βασικές δομές καθώς και πως τις χρησιμοποιούμε.

Τα βήματα που ακολουθούμε είναι τα εξής:

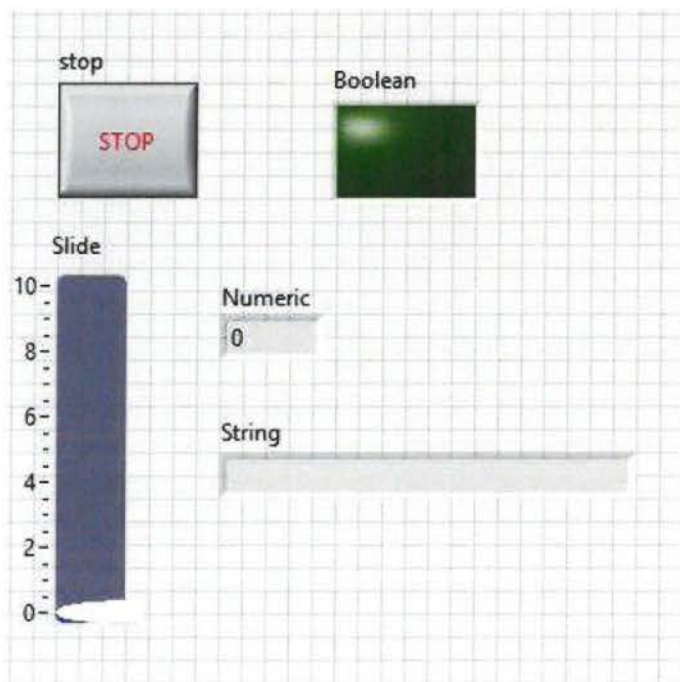
1. Από την οθόνη “Getting Started” ανοίγουμε ένα νέο “Blank VI”.
2. Πηγαίνουμε στο front panel (Ctrl+E).
3. Πατάμε δεξί-κλικ οπουδήποτε στο front panel για να εμφανιστεί η παλέτα Controls.
4. Από την παλέτα Controls πηγαίνουμε Modern>> Numeric και επιλέγουμε τα όργανα Numeric Indicator, και Vertical Pointet Slide.
5. Από την παλέτα Controls πηγαίνουμε Modern>>Boolean και επιλέγουμε το Stop Button και Square Led.
6. Από την παλέτα Controls πηγαίνουμε Modern>>String & Path και επιλέγουμε το όργανο String Indicator.
7. Διαμορφώνουμε το Front Panel έτσι ώστε να είναι ευδιάκριτα τα όργανά μας.
8. Πηγαίνουμε στο block diagram (Ctrl+E).
9. Πατάμε δεξί-κλικ οπουδήποτε στο block diagram και για να εμφανιστεί η παλέτα Functions.
10. Από την παλέτα Functions πηγαίνουμε στο Programming>> Structures >> While Loop , και περικυκλώνουμε όλα τα στοιχεία στο block diagram.
11. Ενώνουμε το STOP button με το Loop Condition.
12. Από την παλέτα Functions πηγαίνουμε στο Programming>> Structures >> For Loop , και περικυκλώνουμε όλα τα στοιχεία στο While Loop εκτός του Stop Button.
13. Από την παλέτα Functions πηγαίνουμε στο Programming>> Structures >> Structures Case και το τοποθετούμε μέσα στο For Loop.

14. Από την παλέτα Functions πηγαίνουμε στο Programming>> String >> String Constant το τοποθετούμε από ένα μέσα σε κάθε κατάσταση (True ή False) του Structures Case.
15. Στο πρώτο String Constant με τιμή False γράφουμε «ΕΠΑΝΑΛΗΨΗ ΤΗΣ ΔΟΜΗΣ» και στο δεύτερο με τιμή True γράφουμε «ΤΕΛΟΣ».
16. Συνδέουμε τα String Constant με το τοίχωμα των Structures Case αντίστοιχα.
17. Από την παλέτα Functions πηγαίνουμε στο Programming>> Numeric >> Numeric Constant και τοποθετούμε ένα μέσα στο For Loop και ένα εκτός.
18. Στο Numeric Constant που είναι εκτός δίνουμε την τιμή 6 και σε αυτό που είναι εντός την τιμή 5.
19. Συνδέουμε το Numeric Constant που είναι εκτός του For Loop με την είσοδο Loop Count του For Loop.
20. Από την παλέτα Functions πηγαίνουμε στο Programming>> Comparison >> Equal? και το τοποθετούμε μέσα στο For Loop.
21. Από την παλέτα Functions πηγαίνουμε στο Express>> Execution Control>>Time Delay, και το τοποθετούμε μέσα στο For Loop αφήνοντάς το 1sec.
22. Συνδέουμε το Loop Iteration της δομής for με την είσοδο του Numeric. Αυτό θα μας δείχνει σε ποια επανάληψη βρισκόμαστε.
23. Κάνουμε δεξί click πάνω στο Slide και επιλέγουμε Change to Indicator και το συνδέουμε και αυτό στο Loop Iteration ώστε να μας δείχνει και αυτό το όργανο σε ποια επανάληψη βρισκόμαστε.
24. Συνδέουμε την είσοδο του Time Delay με το Loop Iteration έτσι ώστε να δέχεται η επανάληψή μας την καθυστέρηση 1 sec.
25. Συνδέουμε το Loop Iteration της δομής for με μία από τις εισόδους του Equal?.
26. Συνδέουμε το Numeric Constant με τιμή 5 με την άλλη είσοδο του Equal?.
27. Συνδέουμε την έξοδο του Equal? Με την είσοδο Case Selector του Case Structure.
28. Συνδέουμε το Boolean με την έξοδο του Equal?.
29. Συνδέουμε το String με το Case Structure.

Στο σχήμα απεικονίζεται το block diagram και το front panel μετά την ολοκλήρωση των παραπάνω βημάτων



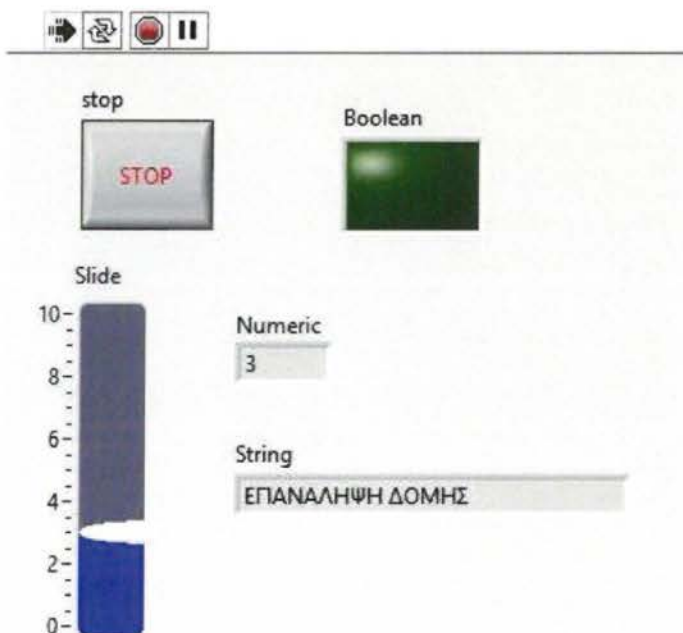
Εικόνα 2.4.1



Εικόνα 2.4.2

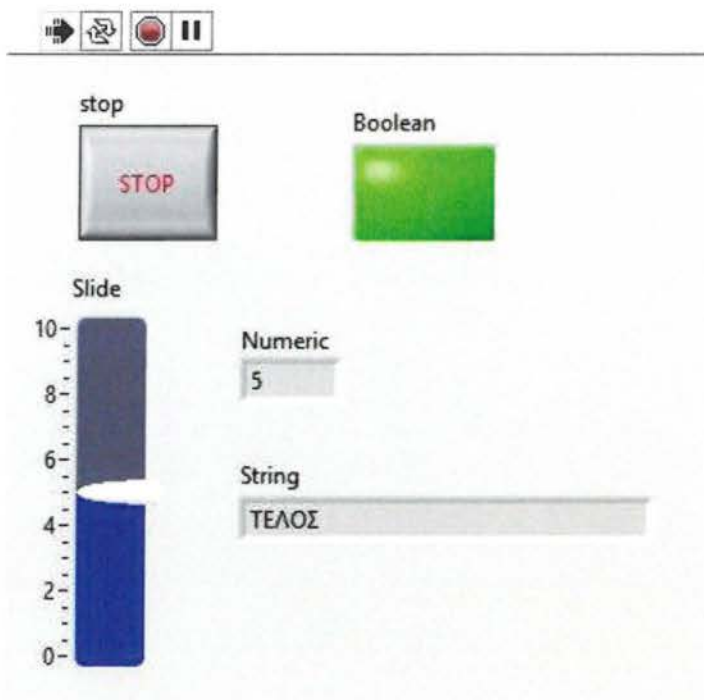
Με το συγκεκριμένο πρόγραμμα βλέπουμε πως μπορούμε να χειριστούμε κάποιες από τις πιο συχνές δομές επανάληψης και επιλογής του προγραμματισμού, με την πλατφόρμα του Labview. Η συνθήκη While Loop ισχύει έως ότου πατηθεί το κουμπί STOP στο Front Panel. Επαναλαμβάνεται δηλαδή όσο είναι TRUE η κατάσταση και με το πάτημα του STOP η κατάσταση γίνεται FALSE και σταματά το πρόγραμμα. Η συνθήκη του For Loop, όπως και στον κλασικό προγραμματισμό, γνωρίζουμε εξ αρχής των αριθμό των επαναλήψεων που θέλουμε να κάνει το πρόγραμμα μας και που έχουμε δώσει από το Numeric Constant με το οποίο συνδέουμε το For Loop. Στην προκυμμένη περίπτωση θέλουμε η συνθήκη μας να επαναληφθεί 6 φορές (από 0 έως 5) γι' αυτό του έχουμε δώσει την τιμή 6. Θέλουμε λοιπόν να επαναλαμβάνεται πέντε φορές το μήνυμα «ΕΠΑΝΑΛΗΨΗ ΤΗΣ ΔΟΜΗΣ» και την τελευταία επανάληψη δηλαδή την έκτη να εμφανιστεί το μήνυμα «ΤΕΛΟΣ». Έτσι ορίσαμε ένα Case Structure το οποίο αναλόγως την κατάσταση του (TRUE ή FALSE) δίνει ως αποτέλεσμα το String Capture. Ο έλεγχος από ότι βλέπουμε γίνεται με τη συνθήκη του Equal το οποίο συγκρίνει τον αριθμό τις επανάληψης i με τον τελευταίο αριθμό του πλήθους των επαναλήψεων (αν $n=6$ το πλήθος του επαναλήψεων τότε $n-1=5$ ο αριθμός σύγκρισης).

Σε μια τυχαία στιγμή αφού πατήσουμε το Run θα έχουμε:



Εικόνα 2.4.3

Όταν φτάσουμε στην πέμπτη επανάληψη το Boolean θα γίνει πράσινο ενώ θα εμφανιστεί το μήνυμα «ΤΕΛΟΣ» όπως την παρακάτω εικόνα:



Εικόνα 2.4.4

2.5 Απεικόνιση δεδομένων - Γραφήματα

Για να μπορέσουμε να απεικονίσουμε γραφήματα στο περιβάλλον του LabVIEW χρησιμοποιούμε τις ειδικές λειτουργίες γραφημάτων (βλέπε βιβλιογραφία) που βρίσκονται στην παλέτα Graphs στο μιμικό παράθυρο. Στο παράδειγμα λοιπόν αυτό θα ασχοληθούμε με την χρήση τριών γραφημάτων, του Waveform Chart, του Waveform Graph και του XY Graph. Θα κατασκευάσουμε εικονόργανο που παρουσιάζει και τα τρία είδη γραφημάτων με πληροφορία που θα λαμβάνει από γεννήτρια τυχαίων αριθμών (ζάρι). Το εικονόργανο θα τερματίζεται από πάτημα σε πλήκτρο STOP.

Τα βήματα που ακολουθούμε είναι τα εξής:

1. Από την οθόνη “Getting Started” ανοίγουμε ένα νέο “Blank VI”.
2. Πηγαίνουμε στο block diagram (Ctrl+E).

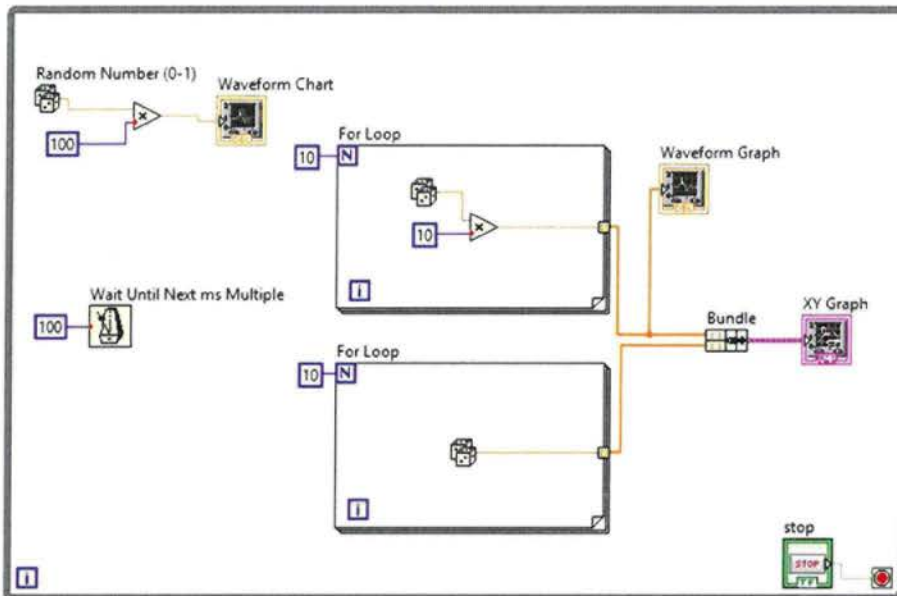
3. Πατάμε δεξί-κλικ οπουδήποτε στο block diagram για να εμφανιστεί η παλέτα Functions.
4. Από την παλέτα Functions πηγαίνουμε στο Programming>> Structures >> While Loop , και ανοίγουμε ένα πλαίσιο στο block diagram.
5. Πηγαίνουμε στο front panel (Ctrl+E).
6. Πατάμε δεξί-κλικ οπουδήποτε στο front panel για να εμφανιστεί η παλέτα Controls.
7. Από την παλέτα Controls πηγαίνουμε Modern>>Boolean, και επιλέγουμε το Stop Button.
8. Από την παλέτα Controls πηγαίνουμε Modern>>Graphs επιλέγουμε και τοποθετούμε τα τρία γραφήματα Waveform Chart, Waveform Graph και XY Graph.
9. Διαμορφώνουμε το Front Panel έτσι ώστε να είναι ευδιάκριτα τα γραφήματα μας.
10. Πηγαίνουμε στο block diagram (Ctrl+E).
11. Από την παλέτα Functions πηγαίνουμε στο Programming>> Structures >> For Loop και ανοίγουμε δύο πλαίσια μέσα στο While Loop.
12. Από την παλέτα Functions πηγαίνουμε στο Mathematics>> Numeric>> Random Number (0-1), και τοποθετούμε ένα μέσα στο While Loop και άλλα δύο αντίστοιχα σε κάθε For Loop. Αυτό το νι παράγει τυχαίους αριθμούς από 0 σε 1.
13. Από την παλέτα Functions πηγαίνουμε στο Programming>> Cluster, Class & Variant>> Bundle και το τοποθετούμε μέσα στο While Loop. Αυτό είναι το στοιχείο της συστάδας που χρειαζόμαστε.
14. Από την παλέτα Functions πηγαίνουμε στο Programming>> Numeric>> Multiply και το τοποθετούμε μέσα στο While Loop.
15. Από την παλέτα Functions πηγαίνουμε στο Programming>> Timing>> Wait Until Next ms Multiply και το μέσα στο While Loop για την χρονό-καθυστέρηση την οποία χρειαζόμαστε.

16. Από την παλέτα Functions πηγαίνουμε στο Programming>> Numeric>> Numeric Constant και το τοποθετούμε τρεις φορές μέσα στο While Loop.
17. Από την παλέτα Functions πηγαίνουμε στο Programming>> Numeric>> DBL Numeric Constant και το τοποθετούμε μέσα στο While Loop.
18. Συνδέουμε το DBL Numeric Constant με την είσοδο y του Multiply δίνοντάς του την τιμή 100.
19. Συνδέουμε το Random Number (0-1) με την είσοδο x του Multiply.
20. Συνδέουμε την έξοδο $x*y$ του Multiply στην είσοδο του Waveform Chaph.
21. Από την παλέτα Functions πηγαίνουμε στο Programming>> Numeric>> Multiply και το τοποθετούμε μέσα στο πρώτο For Loop.
22. Συνδέουμε την είσοδο x του Multiply με το Random Number (0-1)
23. Από την παλέτα Functions πηγαίνουμε στο Programming>> Numeric>> DBL Numeric Constant και το τοποθετούμε μέσα στο For Loop που βρίσκεται το Multiply.
24. Συνδέουμε το DBL Numeric Constant με την είσοδο y του Multiply δίνοντάς του την τιμή 10.
25. Συνδέουμε την έξοδο $x*y$ του Multiply στα τοιχώματα του For Loop.
26. Συνδέουμε το Random Number (0-1) που βρίσκεται στο δεύτερο For Loop με τα τοιχώματα του.
27. Δίνουμε την τιμή 10 στα δύο από τα τρία Numeric Constant και τα συνδέουμε με την είσοδο Loop Count των For Loop αντίστοιχα.
28. Δίνουμε την τιμή 100 στο τρίτο Numeric Constant και το συνδέουμε με το Wait Until Next ms Multiply.
29. Συνδέουμε τα For Loop με τις εισόδους element του Bundle.
30. Συνδέουμε την έξοδο output cluster με την είσοδο του XY Graph.

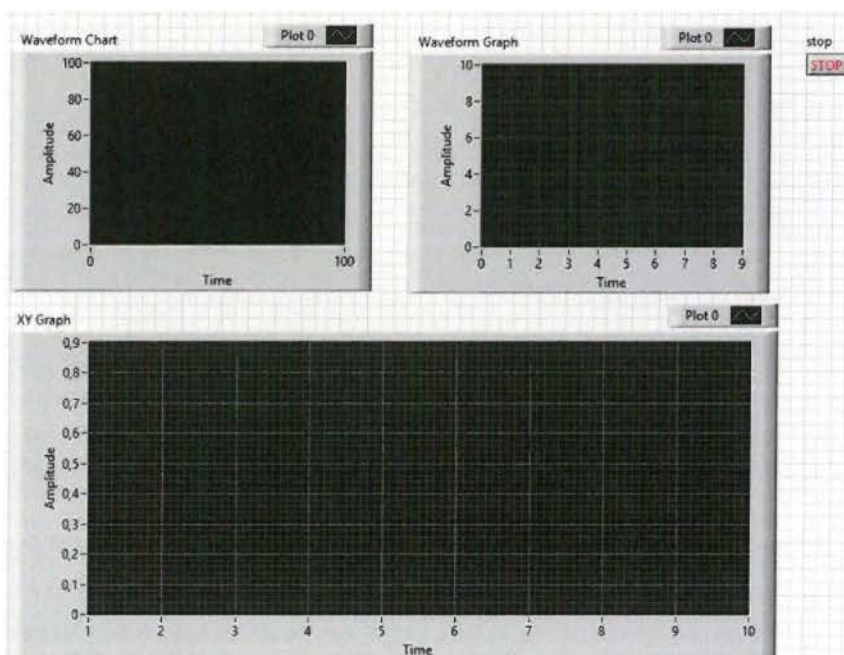
31. Συνδέουμε το Waveform Graph με την έξοδο του For Loop το οποίο έχει το Multiply.

32. Συνδέουμε το STOP button με το Loop Condition.

Στο σχήμα απεικονίζεται το block diagram και το front panel μετά την ολοκλήρωση των παραπάνω βημάτων:

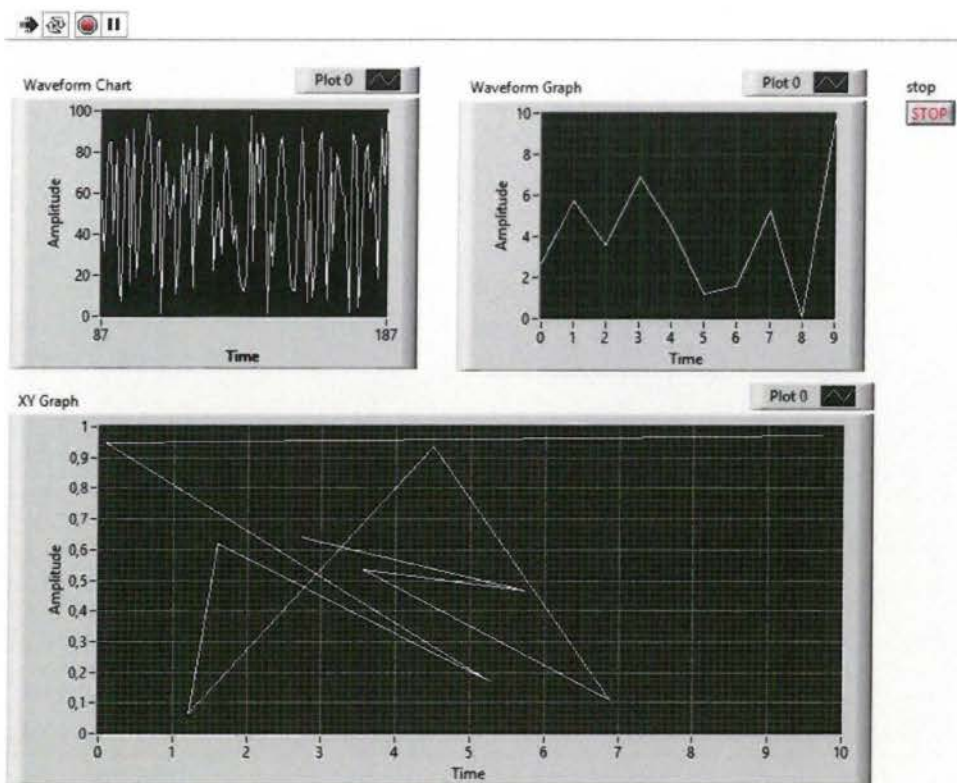


Εικόνα 2.5.1



Εικόνα 2.5.2

Πατώντας το κουμπί Run στο Front Panel το αποτέλεσμα που παίρνουμε σε τυχαία χρονική στιγμή είναι το ακόλουθο:



Εικόνα 2.5.3

Όπως βλέπουμε το αποτέλεσμα των γραφημάτων του πλάτους με τον χρόνο είναι ανάλογο με αυτό που δημιουργήσαμε στο Block Diagram. Ο συγχρονισμός των γραφημάτων γίνεται με το Wait Until Next ms Multiply το οποίο έχουμε ρυθμίσει στα 100ms. Το πρώτο γράφημα είναι το αποτέλεσμα του πολλαπλασιασμού τυχαίων αριθμών (από το 0 έως το 1) με το 100, σε συνάρτηση με το χρόνο. Τα άλλα δυο γραφήματα επαναλαμβάνονται κάθε 10 φορές μέσω της For Loop, με το ένα να λαμβάνει τιμές έως το 10 ενώ το άλλο δέχεται τις τιμές του σαν άξονα Y και σαν άξονα X δέχεται τυχαίους αριθμούς έως το 1.

2.6 Αποθήκευση Δεδομένων

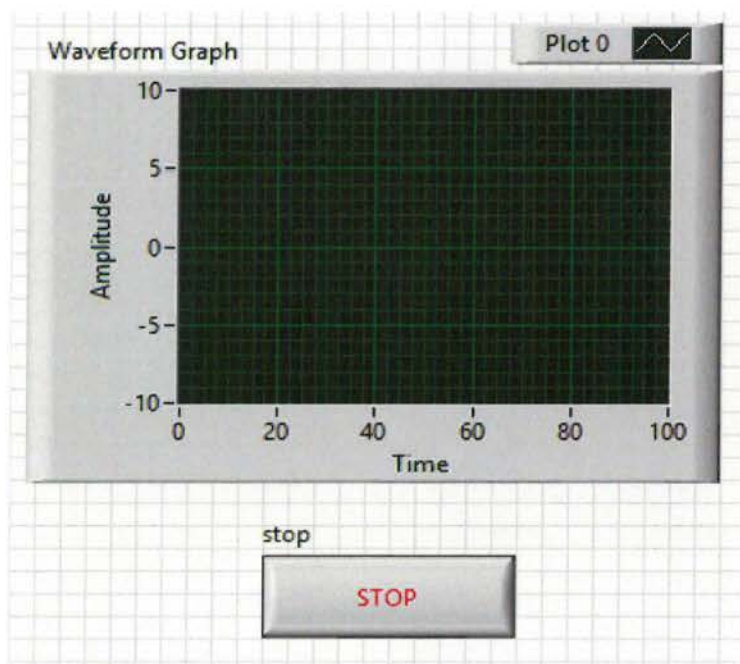
Στην δραστηριότητα αυτή θα δημιουργήσουμε εικονόργανο το οποίο θα προσομοιώνει την παραγωγή ενός ημιτονοειδούς σήματος και θα την αποθηκεύει σε αρχείο κειμένου. Έτσι μπορούμε να επεξεργαστούμε τις πληροφορίες του σήματος με άλλα λογισμικά όπως το Excel.

Τα βήματα που ακολουθούμε είναι τα εξής:

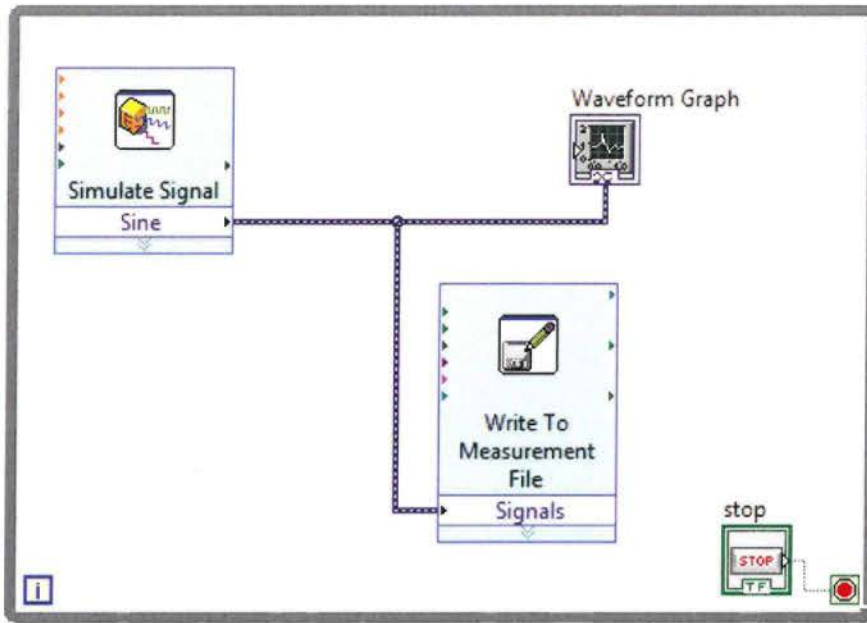
1. Από την οθόνη “Getting Started” ανοίγουμε ένα νέο “Blank VI”.
2. Πηγαίνουμε στο block diagram (Ctrl+E).
3. Πατάμε δεξί-κλικ οπουδήποτε στο block diagram για να εμφανιστεί η παλέτα Functions.
4. Από την παλέτα Functions πηγαίνουμε στο Programming>> Structures >> While Loop , και ανοίγουμε ένα πλαίσιο στο block diagram.
5. Πηγαίνουμε στο front panel (Ctrl+E).
6. Πατάμε δεξί-κλικ οπουδήποτε στο front panel για να εμφανιστεί η παλέτα Controls.
7. Από παλέτα Controls πηγαίνουμε Modern>>Boolean, και επιλέγουμε το Stop Button.
8. Από την παλέτα Controls πηγαίνουμε Modern>>Graph επιλέγουμε και τοποθετούμε στο front panel το γράφημα Waveform Graph.
9. Πηγαίνουμε στο block diagram (Ctrl+E).
10. Από την παλέτα Functions πηγαίνουμε στο Express>>Input επιλέγουμε το Simulate Signal και το τοποθετούμε μέσα στο While Loop.
11. Στο παράθυρο Configure Simulate Signal που μας εμφανίζεται, διαμορφώνουμε τις παραμέτρους ως εξής: Signal type: Sine (Ημιτονοηδής), Frequency (Hz): 10.1, Phase (deg): 0, Amplitude: 1, Offset: 0, και Samples per second (Hz): 1000 και πατάμε OK.
12. Από την παλέτα Functions πηγαίνουμε στο Express>>Output επιλέγουμε το Write To Measurement File και το τοποθετούμε μέσα στο While Loop.

13. Στο παράθυρο Configure Write To Measurement File που μας εμφανίζεται, το Filename μας δείχνει το φάκελο και την μορφή που θα αποθηκευτεί το σήμα μας.
14. Συνδέουμε την έξοδο του Simulate Signal με την είσοδο του Waveform Graph.
15. Συνδέουμε το Write To Measurement File με την είσοδο του Waveform Graph.
16. Συνδέουμε το STOP button με το Loop Condition.

Το Front Panel και το Block Diagram μετά την ολοκλήρωση των παραπάνω βημάτων είναι:

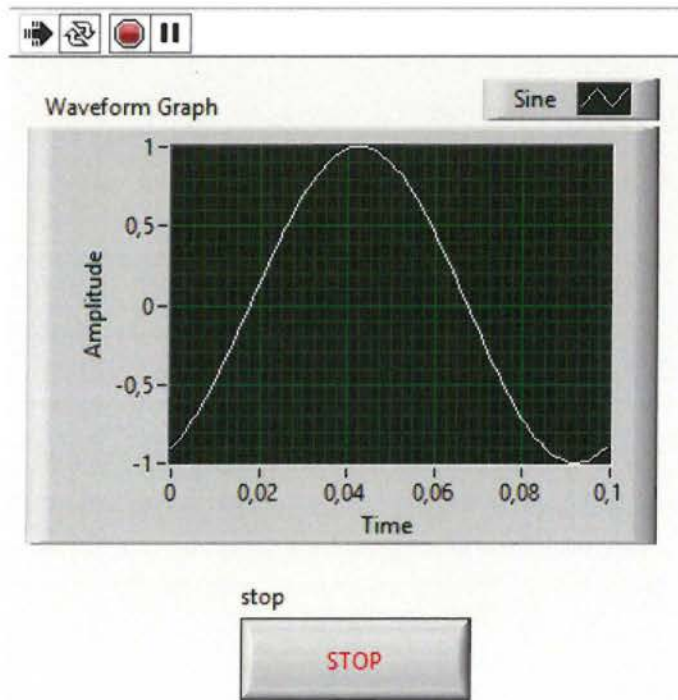


Εικόνα 2.6.1



Εικόνα 2.6.2

Πατώντας το κουμπί Run το αποτέλεσμα της γραφική μας παράσταση σε τυχαία στιγμή θα είναι:



Εικόνα 2.6.3

Αν πάμε στο block diagram και κάνουμε διπλό-click πάνω στο Write To Measurement File θα δούμε το path που έχει αποθηκευτή το αρχείο μας, σε μορφή LVM. Μπορούμε να το ανοίξουμε με το Notepad, και να δούμε πληροφορίες για την κυματομορφή μας. Μπορείτε να πειραματιστείτε να ανοίξετε το αρχείο στο λογισμικό excel με επιλογή εισαγωγής στοιχείων κειμένου όπως θα σας ορίζει ο οδηγός στο excel κατά το άνοιγμα του αρχείου.

2.7 Δημιουργία Σήματος

Η τεχνολογία των μετρήσεων αποκτά αξία όταν γίνεται η ακριβής μέτρηση του μεγέθους που επιθυμούμε. Για να είμαστε ικανοί να αξιοποιήσουμε κάθε παράμετρο του σήματος θα πρέπει να γνωρίζουμε τους ορισμούς των ηλεκτρικών χαρακτηριστικών του σήματος. Στη συνέχεια θα αναφερθούμε επιγραμματικά στα κυριότερα ηλεκτρικά μεγέθη:

- Σήμα καλούμε κάθε ηλεκτρική μεταβολή στη μονάδα του χρόνου (sec). Τα σήματα τα διακρίνουμε σε αναλογικά ψηφιακά.
- Αναλογικά καλούμε τα σήματα εκείνα τα οποία η μεταβολή τους είναι συνεχή στο χρόνο
- Ψηφιακά τα σήματα εκείνα τα οποία η μεταβολή τους είναι διακριτή στο χρόνο.
- Τα σήματα διακρίνονται σε περιοδικά και μη περιοδικά.
- Συχνότητα (f) ενός περιοδικού σήματος καλούμε τον αριθμό των επαναλαμβανόμενων περιόδων στη μονάδα του χρόνου.
- Περίοδος (T) σήματος καλούμε το χρόνο επανάληψης ενός περιοδικού σήματος.
- Τάση από κορυφή σε κορυφή (U_p-p) ονομάζουμε το δυναμικό μεταξύ της μέγιστης και ελάχιστης τιμής του περιοδικού σήματος. Η τάσης από κορυφή σε κορυφή μετριέται σε Volt.
- Πλάτος (U_0) σήματος καλούμε το ήμισυ της τάσης από κορυφή σε κορυφή του σήματος. Το πλάτος του σήματος μετριέται σε Volt.

- Ενεργός ένταση (I_{εν}) ενός εναλλασσομένου ρεύματος καλείται η ένταση εκείνου του συνεχούς ρεύματος που προκαλεί το ίδιο θερμικό αποτέλεσμα με την ένταση του εναλλασσομένου ρεύματος, που διαρρέει στον ίδιο χρόνο τον ίδιο αντιστάτη.
- Ενεργός τάση (U_{εν}) μιας εναλλασσόμενης τάσης, καλείται η τιμή εκείνης της συνεχούς τάσης, που προκαλεί συνεχές ρεύμα έντασης ίσης με την ενεργό ένταση του εναλλασσομένου ρεύματος στα άκρα αντιστάτη (R), που θα προκαλούσε η εναλλασσόμενη τάση στον ίδιο αντιστάτη.

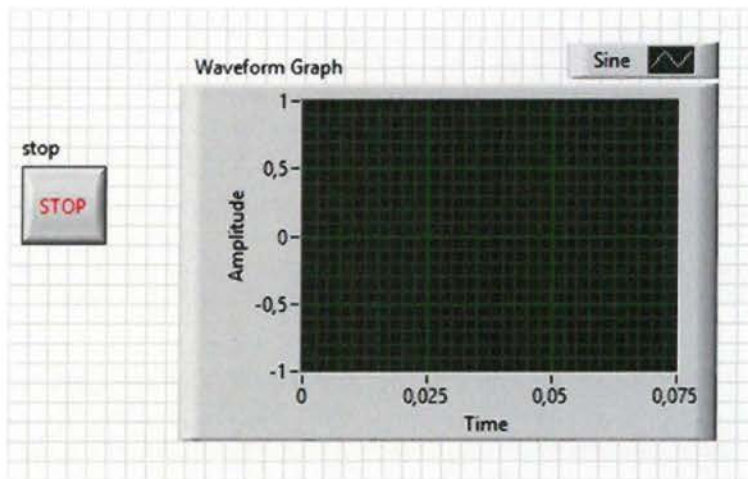
Σε αυτό το παράδειγμα θα κατασκευάσουμε εικονόργανο το οποίο θα προσομοιώνει την ανάγνωση ημιτονοειδούς σήματος βάσει της συχνότητας δειγματοληψίας του.

Τα βήματα που ακολουθούμε είναι τα εξής:

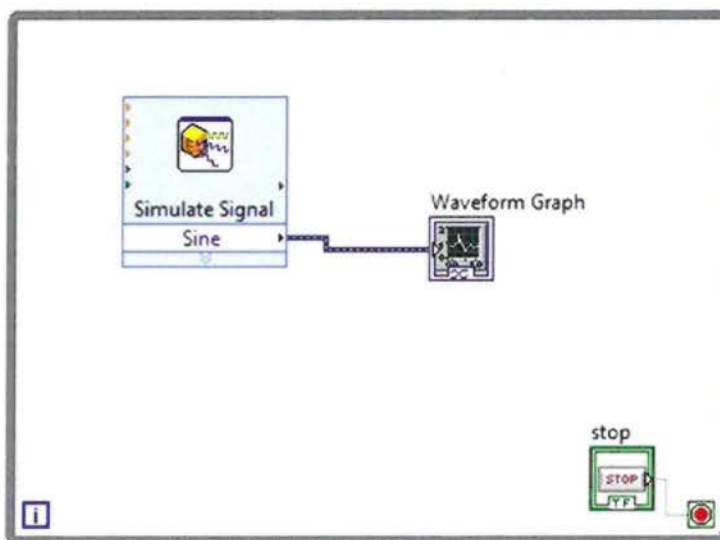
1. Από την οθόνη “Getting Started” ανοίγουμε ένα νέο “Blank VI”.
2. Πηγαίνουμε στο block diagram (Ctrl+E).
3. Πατάμε δεξί-κλικ οπουδήποτε στο block diagram και για να εμφανιστεί η παλέτα Functions.
4. Από την παλέτα Functions πηγαίνουμε στο Programming>> Structures >> While Loop , και ανοίγουμε ένα πλαίσιο στο block diagram.
5. Πηγαίνουμε στο front panel (Ctrl+E).
6. Πατάμε δεξί-κλικ οπουδήποτε στο front panel για να εμφανιστεί η παλέτα Controls.
7. Από την παλέτα Controls πηγαίνουμε στο Modern>>Boolean, και επιλέγουμε το Stop Button.
8. Από την παλέτα Controls πηγαίνουμε Modern>>Graphs επιλέγουμε και τοποθετούμε το γράφημα Waveform Graph.
9. Πηγαίνουμε στο block diagram (Ctrl+E).
10. Από την παλέτα Functions πηγαίνουμε στο Express >> Signal Analysis >> Simulate Signal , και το τοποθετούμε μέσα στο While Loop.

11. Στο παράθυρο που μας εμφανίζεται, διαμορφώνουμε τις παραμέτρους ως εξής: Signal type: Sine (Ημιτονοειδής), Frequency (Hz): 10.1, Phase (deg): 0, Amplitude: 1, Offset: 0, και Samples per second (Hz): 40 και πατάμε OK.
12. Συνδέουμε το STOP button με το Loop Condition.
13. Συνδέουμε την έξοδο του Simulate Signal με την είσοδο του Waveform Graph.

Το Front Panel και το Block Diagram μετά την ολοκλήρωση των παραπάνω βημάτων είναι:

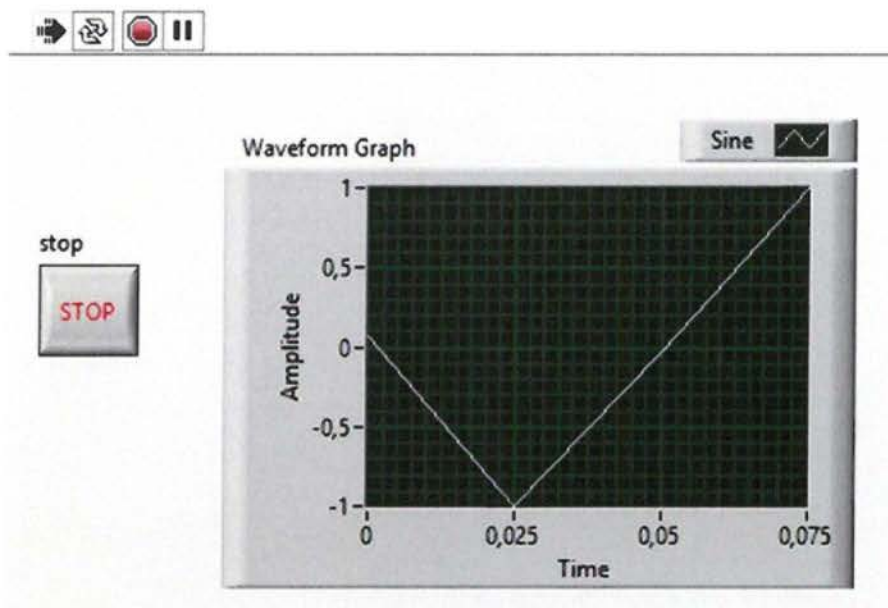


Εικόνα 2.7.1



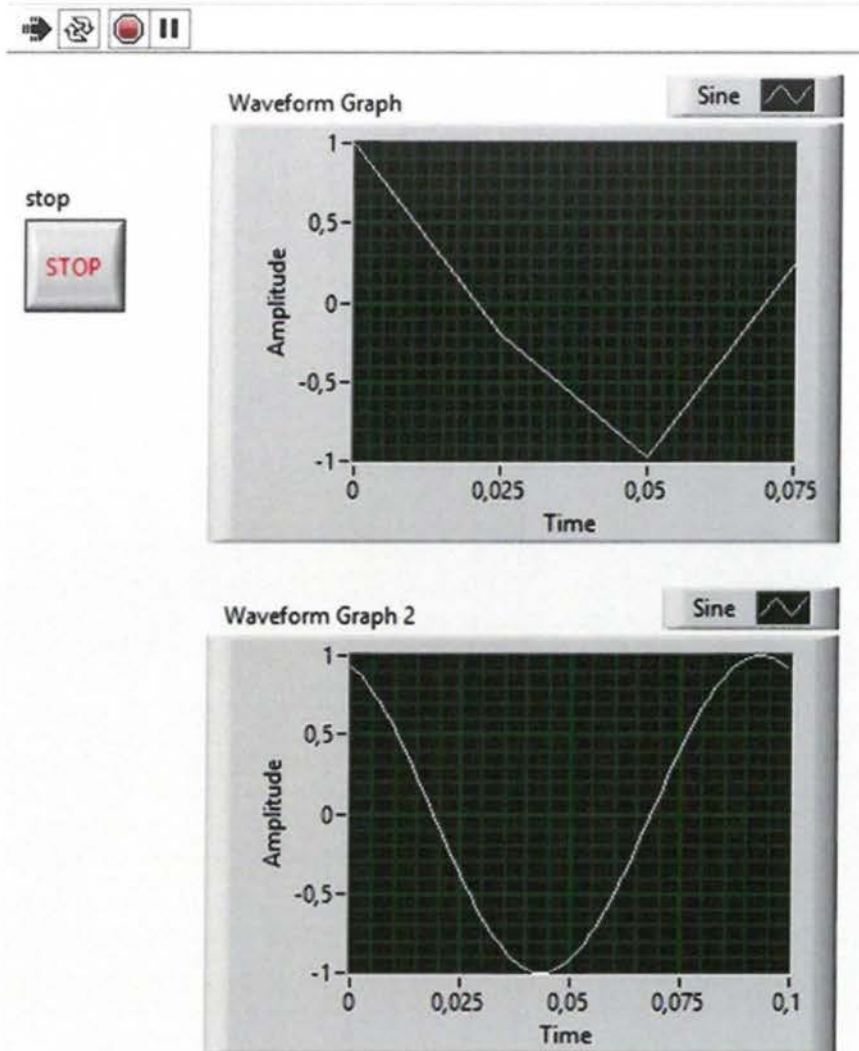
Εικόνα 2.7.2

Αν πατήσουμε Run το αποτέλεσμα του προγράμματός μας σε μια τυχαία στιγμή θα είναι:



Εικόνα 2.7.3

Αν θέλουμε να συγκρίνουμε δύο ημιτονοειδής σήματα αλλά με διαφορετική συχνότητα δειγματοληψίας τότε στο υπάρχον πρόγραμμα κάνουμε Copy paste (Ctrl+C) το Simulate Signal και του Waveform Graph αλλά στο δεύτερο βάζουμε Samples per second (Hz): 1000. Τότε το αποτέλεσμα θα είναι:



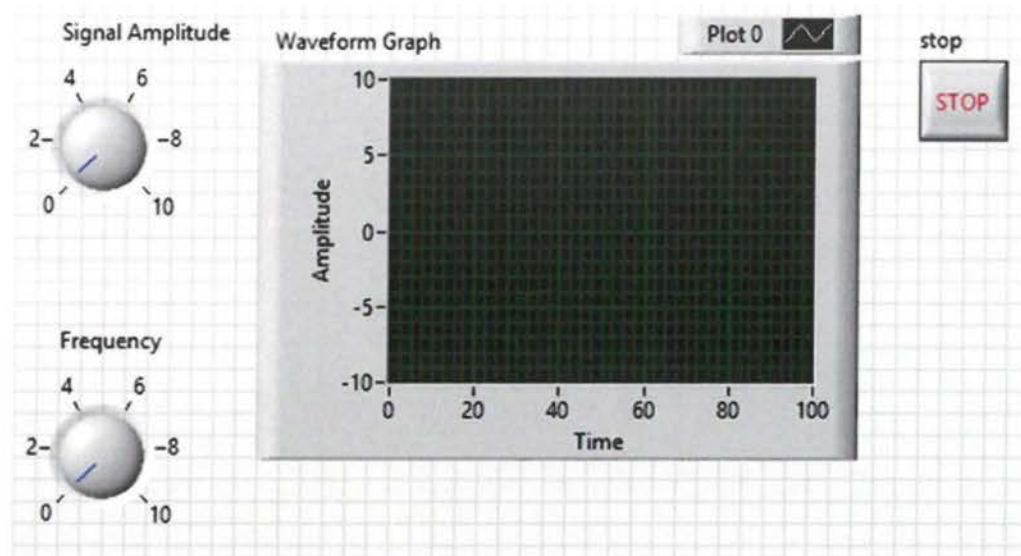
Εικόνα 2.7.4

Ένα άλλο παράδειγμα είναι η κατασκευή εικονόργανου που θα προσομοιώνει τη παραγωγή τριγωνικού σήματος με μεγέθη πλάτους και συχνότητας σήματος που θα ορίσει ο χρήστης του εικονόργανου στο μιμικό παράθυρο. Έτσι με το ίδιο σκεπτικό και χρησιμοποιώντας πάλι το Simulate Signal ακολουθούμε τα εξής βήματα:

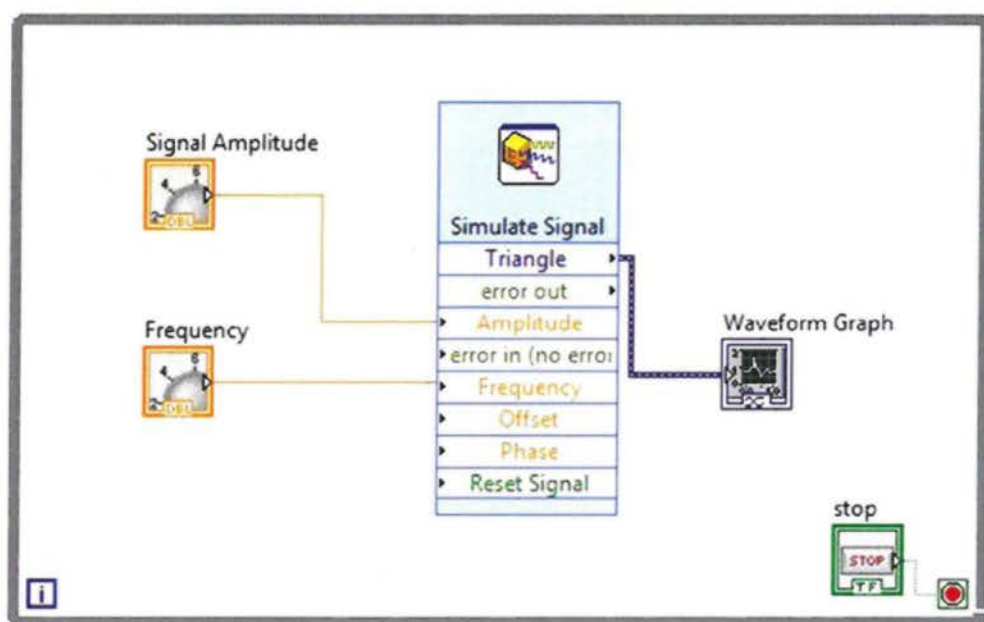
1. Από την οθόνη "Getting Started" ανοίγουμε ένα νέο "Blank VI".
2. Πηγαίνουμε στο block diagram (Ctrl+E).
3. Πατάμε δεξί-κλικ οπουδήποτε στο block diagram και για να εμφανιστεί η παλέτα Functions.

4. Από την παλέτα Functions πηγαίνουμε στο Programming>> Structures >> While Loop , και ανοίγουμε ένα πλαίσιο στο block diagram.
5. Πηγαίνουμε στο front panel (Ctrl+E).
6. Πατάμε δεξί-κλικ οπουδήποτε στο front panel για να εμφανιστεί η παλέτα Controls.
7. Από την παλέτα Controls πηγαίνουμε στο Modern>>Boolean, και επιλέγουμε το Stop Button.
8. Από την παλέτα Controls πηγαίνουμε στο Modern>>Graphs επιλέγουμε και τοποθετούμε το γράφημα Waveform Graph.
9. Από την παλέτα Controls πηγαίνουμε στο Modern>>Numeric>>Dial επιλέγουμε και τοποθετούμε δύο ποτενσιόμετρα και τους δίνουμε αντίστοιχα τα ονόματα Frequency και Signal Amplitude.
10. Πηγαίνουμε στο block diagram (Ctrl+E).
11. Από την παλέτα Functions πηγαίνουμε στο Express >> Signal Analysis>>Simulate Signal , και το τοποθετούμε μέσα στο While Loop.
12. Στο παράθυρο που μας εμφανίζεται, διαμορφώνουμε τις παραμέτρους ως εξής: Signal type: Triangle (Τριγωνικό), Frequency (Hz): 10.1, Phase (deg): 0, Amplitude: 1, Offset: 0, και Samples per second (Hz): 1000 και πατάμε OK.
13. Συνδέουμε το STOP button με το Loop Condition.
14. Συνδέουμε την έξοδο του Simulate Signal, Triangle με την είσοδο του Waveform Graph.
15. Συνδέουμε το Signal Amplitude με το Amplitude του Simulate Signal.
16. Συνδέουμε το Frequency με το Frequency του Simulate Signal.

Το Front Panel και το Block Diagram μετά την ολοκλήρωση των παραπάνω βημάτων είναι:

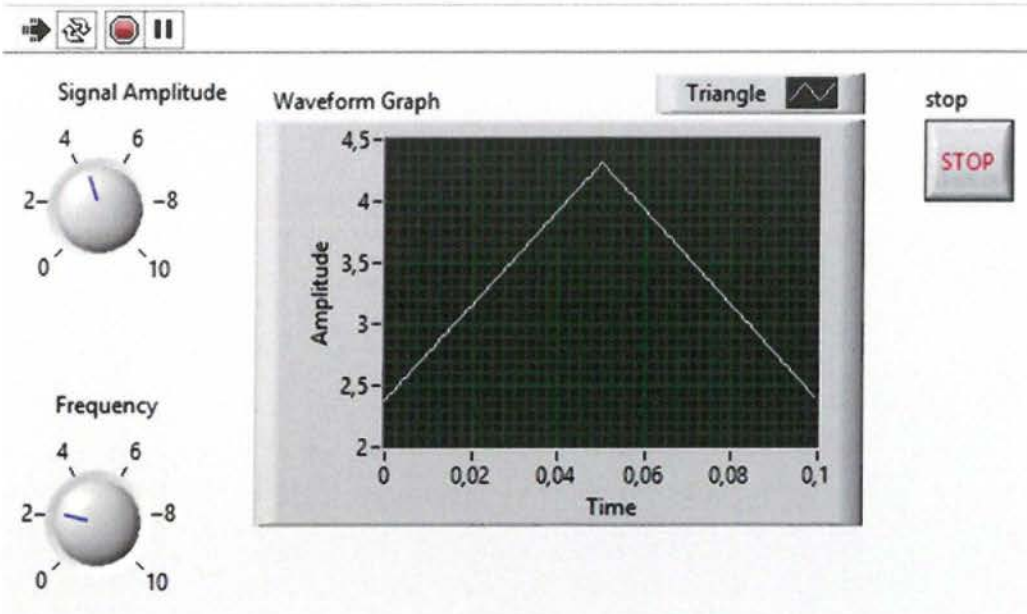


Εικόνα 2.7.5



Εικόνα 2.7.6

Αν πατήσουμε Run το αποτέλεσμα του προγράμματός μας σε μια τυχαία στιγμή θα είναι:



Εικόνα 2.7.7

Όπως βλέπουμε το Signal Amplitude και Frequency μεταβάλουν το πλάτος και την συχνότητα του τριγωνικού μας σήματος αντίστοιχα.

2.8 Φίλτρο

Η National Instruments διαθέτει πλήθος ειδικών μονάδων για συστήματα φίλτρων, ενίσχυσης, ο ψαλιδισμού, ψηφιακής επεξεργασίας σήματος, συμπίεσης θορύβου κ.α.. Στη δραστηριότητα αυτή θα κατασκευάσουμε εικονόργανο που θα προσομοιώνει ένα ημιτονοειδές σήμα που περιέχει στοιχεία λευκού θορύβου και θα φιλτράρεται από κατάλληλο Express VI.

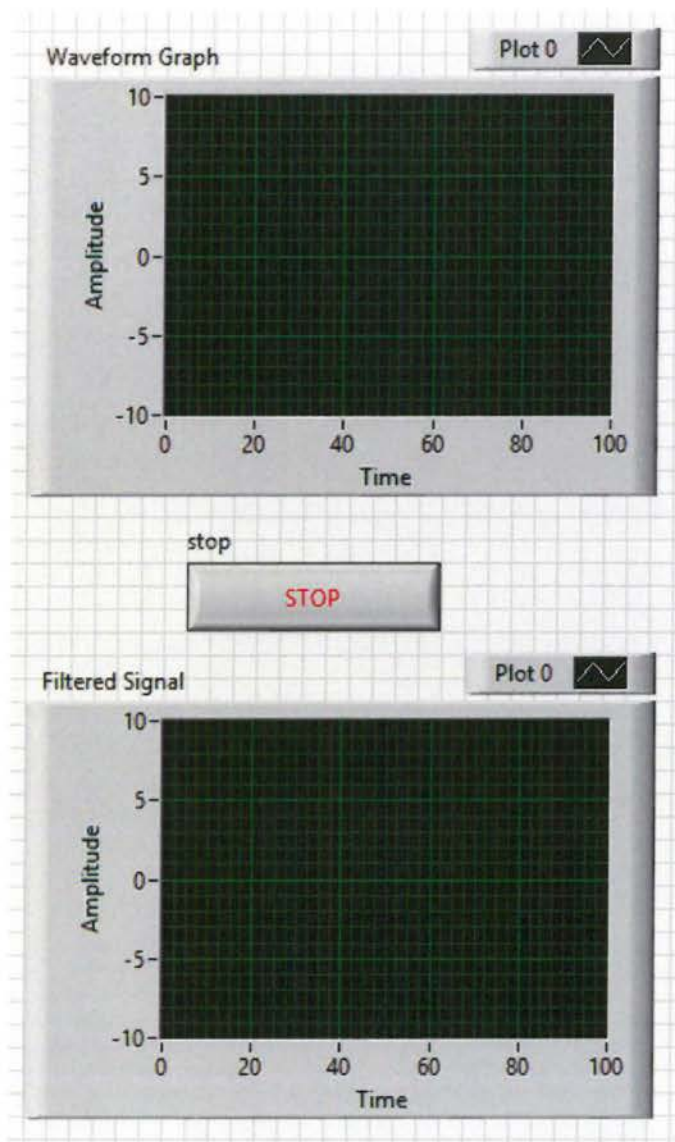
Τα βήματα που ακολουθούμε είναι τα εξής:

1. Από την οθόνη "Getting Started" ανοίγουμε ένα νέο "Blank VI".
2. Πηγαίνουμε στο block diagram (Ctrl+E).
3. Πατάμε δεξί-κλικ οπουδήποτε στο block diagram και για να εμφανιστεί η παλέτα Functions.
4. Από την παλέτα Functions πηγαίνουμε στο Programming>> Structures >> While Loop , και ανοίγουμε ένα πλαίσιο στο block diagram.

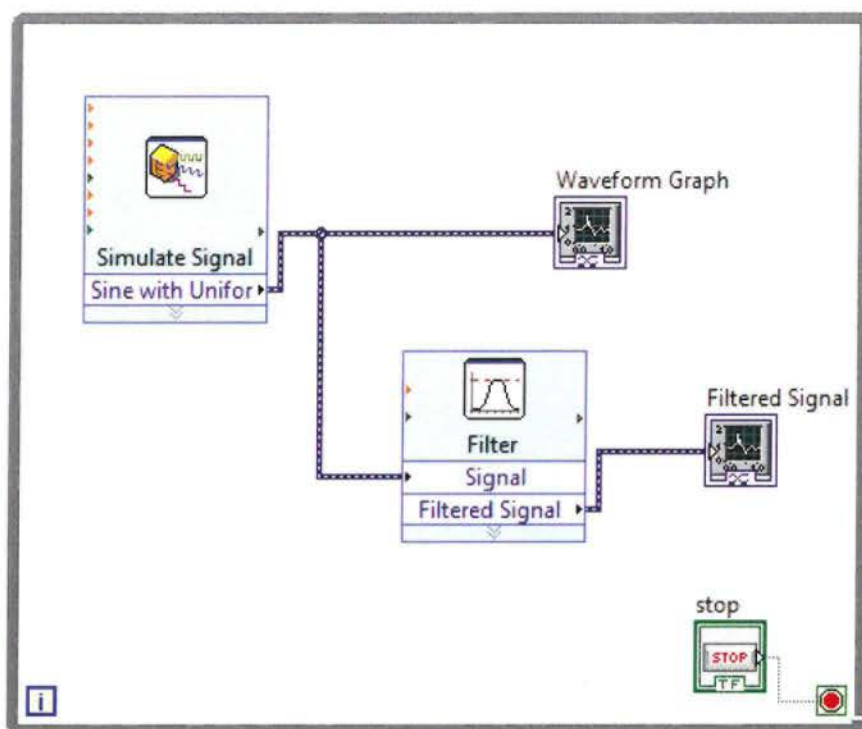
5. Πηγαίνουμε στο front panel (Ctrl+E).
6. Πατάμε δεξί-κλικ οπουδήποτε στο front panel για να εμφανιστεί η παλέτα Controls.
7. Από την παλέτα Controls πηγαίνουμε στο Modern>>Boolean, και επιλέγουμε το Stop Button.
8. Από την παλέτα Controls πηγαίνουμε Modern>>Graphs επιλέγουμε και τοποθετούμε το γραφήματα Waveform Graph
9. Κάνουμε αντιγραφή επικόλληση άλλο ένα Waveform Graph και του δίνουμε το όνομα «Filtered Signal».
10. Πηγαίνουμε στο block diagram (Ctrl+E).
11. Από την παλέτα Functions πηγαίνουμε στο Express >> Signal Analysis >> Simulate Signal , και το τοποθετούμε μέσα στο While Loop.
12. Στο παράθυρο που μας εμφανίζεται, διαμορφώνουμε τις παραμέτρους ως εξής: Signal type: Sine (Ημιτονοηδής),
Frequency (Hz): 10.1,
Phase (deg): 0,
Amplitude: 1,
Offset: 0,
Samples per second (Hz): 40
Κάνουμε click το Add noise με Noise type: Uniform White Noise,
Noise amplitude: 0.6,
Seed number:-1, και πατάμε OK.
13. Από την παλέτα Functions πηγαίνουμε στο Express >> Signal Analysis >> Filter , και το τοποθετούμε μέσα στο While Loop.
14. Στο παράθυρο που μας εμφανίζεται, διαμορφώνουμε τις παραμέτρους ως εξής: Filtering type: Lowpass (Χαμηλοπερατό),
Cutoff Frequency (Hz): 11 και πατάμε OK.
15. Συνδέουμε την έξοδο του Simulate Signal με την είσοδο του Waveform Graph.
16. Συνδέουμε την έξοδο Signal του Filter με την είσοδο του Waveform Graph.
17. Συνδέουμε την έξοδο Filtered Signal του Filter με το Filtered Signal.

18. Συνδέουμε το STOP button με το Loop Condition.

Το Front Panel και το Block Diagram μετά την ολοκλήρωση των παραπάνω βημάτων είναι:

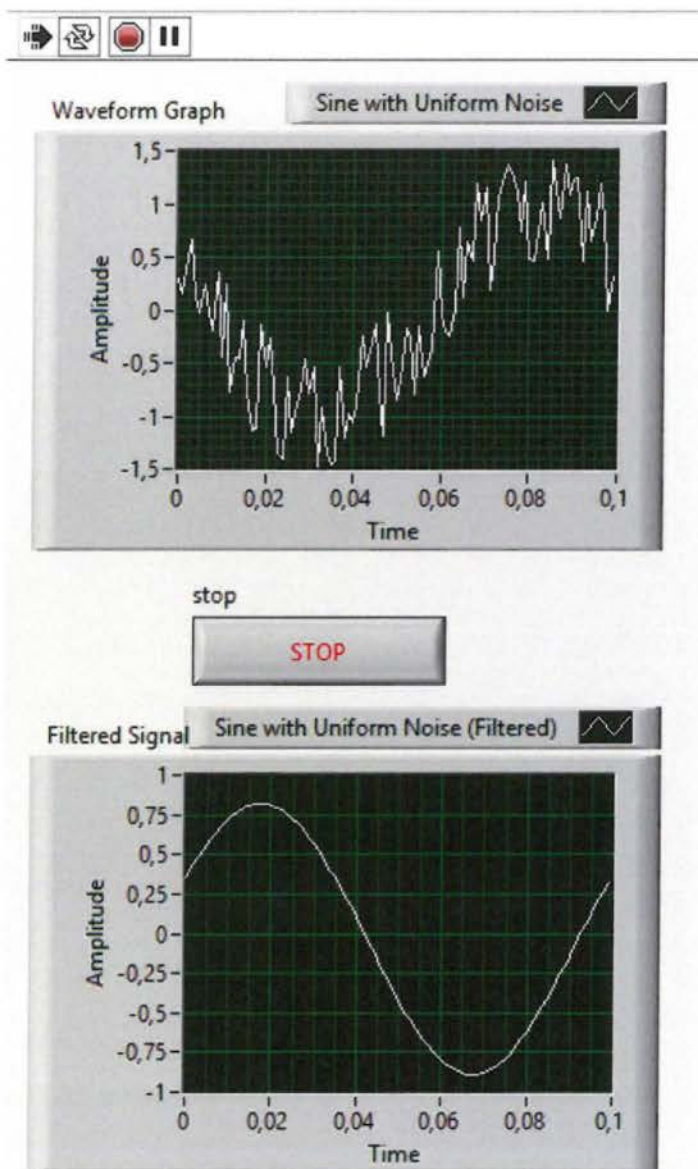


Εικόνα 2.8.1



Εικόνα 2.8.2

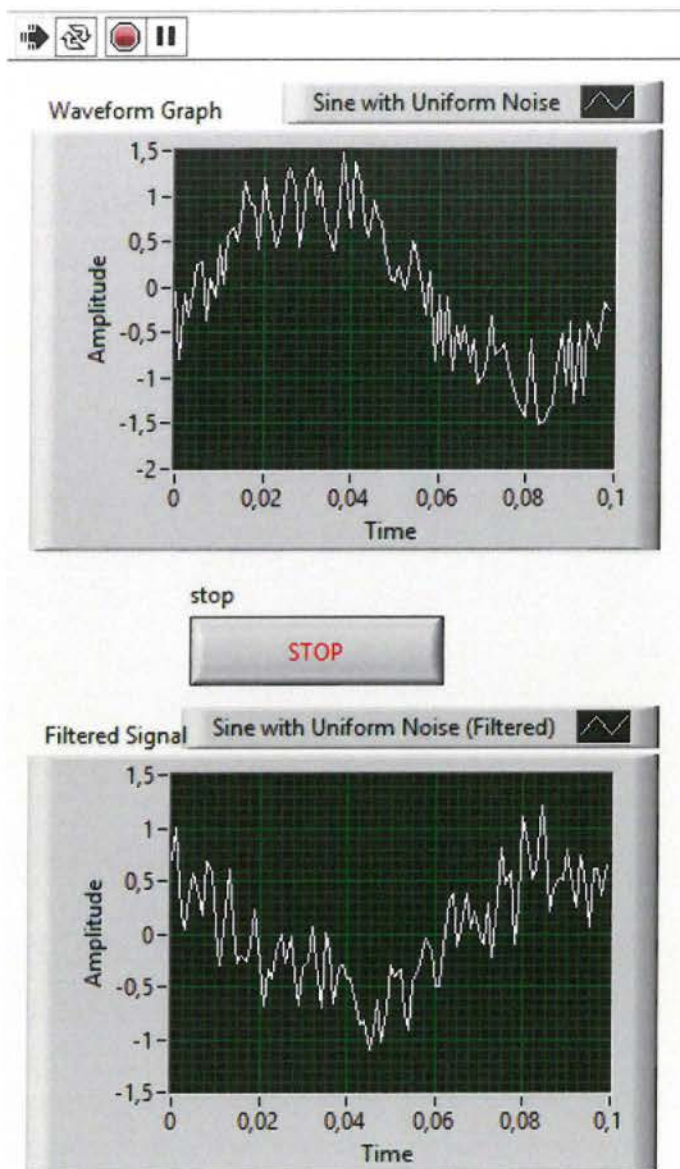
Αν πατήσουμε Run το αποτέλεσμα του προγράμματός μας σε μια τυχαία στιγμή θα είναι:



Εικόνα 2.8.3

Αν θέλουμε να αλλάξουμε το τύπο του φίλτρου, κάνουμε διπλό-click στο Filter και στο Filtering Type από Lowpass το κάνουμε Bandpass (ζωνοπερατό).

Πατώντας το Run το αποτέλεσμά μας θα είναι:



Εικόνα 2.8.4

Το αποτέλεσμά μας από χαμηλοπερατό σε ζωνοπερατό φέναται στα γραφήματά μας.

2.9 Δημιουργία συστήματος Ανοιχτού Βρόγχου

Στο συγκεκριμένο παράδειγμα θα σχεδιάσουμε ένα πρόγραμμα προσομοιώσεις ανοιχτού βρόχου στο LabVIEW. Θα δίνουμε ως είσοδο την συνάρτηση μεταφοράς και θα παίρνουμε την κυματομορφή της συχνότητας ως προς το χρόνο. Η συνάρτηση μεταφοράς που θα δώσουμε θα είναι η:

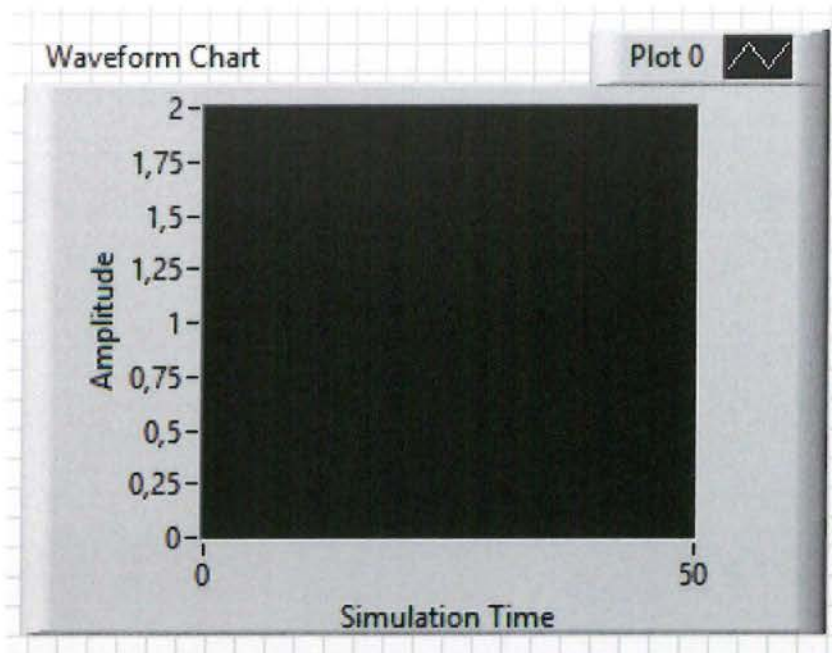
$$TF(s) = \frac{2}{50s^2 + 15s + 1}$$

Τα βήματα που ακολουθούμε είναι τα εξής:

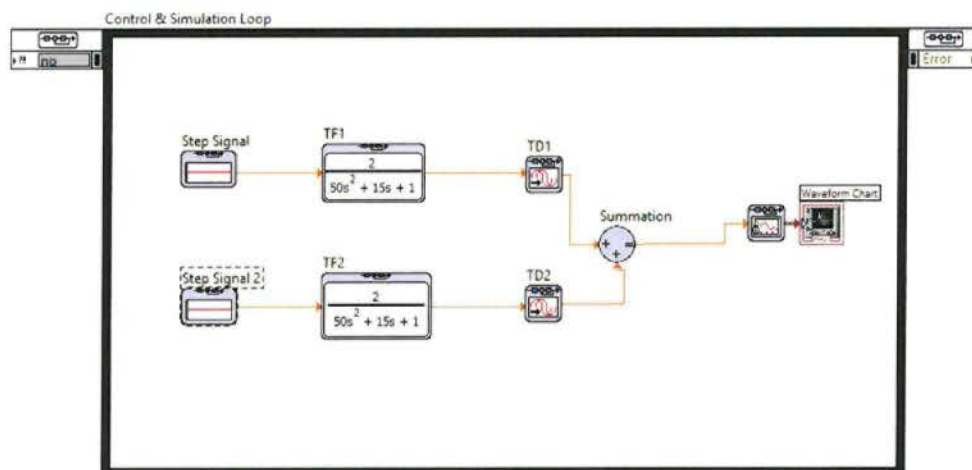
1. Από την οθόνη “Getting Started” ανοίγουμε ένα νέο “Blank VI”.
2. Πηγαίνουμε στο block diagram (Ctrl+E).
3. Πατάμε δεξί-κλικ οπουδήποτε στο block diagram και για να εμφανιστεί η παλέτα Functions.
4. Από την παλέτα Functions πηγαίνουμε στο Control Design & Simulation >> Simulation >> Control & Simulation Loop, επιλέγουμε και ανοίγουμε ένα πλαίσιο στο block diagram.
5. Από την παλέτα Functions πηγαίνουμε στο Control Design & Simulation >> Simulation >> Continuous Linear Systems >> Transfer Function, επιλέγουμε και το τοποθετούμε μέσα στο πλαίσιο του Control & Simulation Loop.
6. Κάνουμε διπλό-click στο Transfer Function και μορφοποιούμε την συνάρτηση που θέλουμε βάζοντας αριθμητή 2 και παρονομαστή $50s^2 + 15s + 1$.
7. Από την παλέτα Functions πηγαίνουμε στο Control Design & Simulation >> Simulation >> Continuous Linear Systems >> Transport Delay, επιλέγουμε και το τοποθετούμε μέσα στο πλαίσιο του Control & Simulation Loop.
8. Κάνουμε διπλό-click στο Transport Delay και ορίζουμε καθυστέρηση 1.
9. Κάνουμε αντιγραφή-επικόλληση το Transfer Function και Transport Delay.
10. Δίνουμε στο πρώτο ζεύγος Transfer Function και Transport Delay τα ονόματα TF1 και TD1 αντίστοιχα, και στο δεύτερο ζεύγος τα ονόματα TF2 και TD2.

11. Από την παλέτα Functions πηγαίνουμε στο Control Design & Simulation >>Simulation>>Signal Arithmetic επιλέγουμε τον αθροιστή Sumation και τον τοποθετούμε μέσα στο πλαίσιο του Control & Simulation Loop.
12. Κάνουμε διπλό-click στο Sumation και το μορφοποιούμε έτσι ώστε να προσθέσουμε το TF1 και TF2. Γι' αυτό βάζουμε δύο + και πατάμε OK.
13. Συνδέουμε την έξοδο output του TF1 με την είσοδο input του TD1, και αντίστοιχα του TF2 με του TD2.
14. Συνδέουμε την έξοδο output του TD1 με την μία είσοδο Operand1 του Sumation, και αντίστοιχα του TD2 με το Operand2.
15. Από την παλέτα Functions πηγαίνουμε στο Control Design & Simulation>>Simulation>>Graph Utilities, επιλέγουμε το Sim Time Waveform και το τοποθετούμε μέσα στο πλαίσιο του Control & Simulation Loop.
16. Συνδέουμε την έξοδο Result του αθροιστή Summation, με την είσοδο Value του Sim Time Waveform.
17. Από την παλέτα Functions πηγαίνουμε στο Control Design & Simulation>>Simulation>>Signal Generator, επιλέγουμε το Step Signal και το τοποθετούμε δύο μέσα στο πλαίσιο του Control & Simulation Loop.
18. Κάνουμε διπλό-click στο πρώτο Step Signal και κάνουμε τις τιμές initial value, final value και step time όλες 0.
19. Κάνουμε διπλό-click στο δεύτερο Step Signal και κάνουμε τις τιμές initial value, και step time μηδέν και final value ένα.
20. Συνδέουμε την έξοδο output του πρώτου Step Signal με την είσοδο input του TF1, και αντίστοιχα του δεύτερου Step Signal με το input του TF2.
21. Κάνουμε διπλό-click στο Configure Simulation Parameters και βλέπουμε ότι στο Simulation Time έχουμε όρια από 0 έως 50.
22. Πηγαίνουμε στο front panel (Ctrl+E).
23. Στο Waveform Graph το οποίο έχει εμφανιστεί αλλάζουμε τα όρια του Amplitude από 0 έως 2, και του Simulation Time από 0 έως 50.

Το Front Panel και το Block Diagram μετά την ολοκλήρωση των παραπάνω βημάτων είναι:

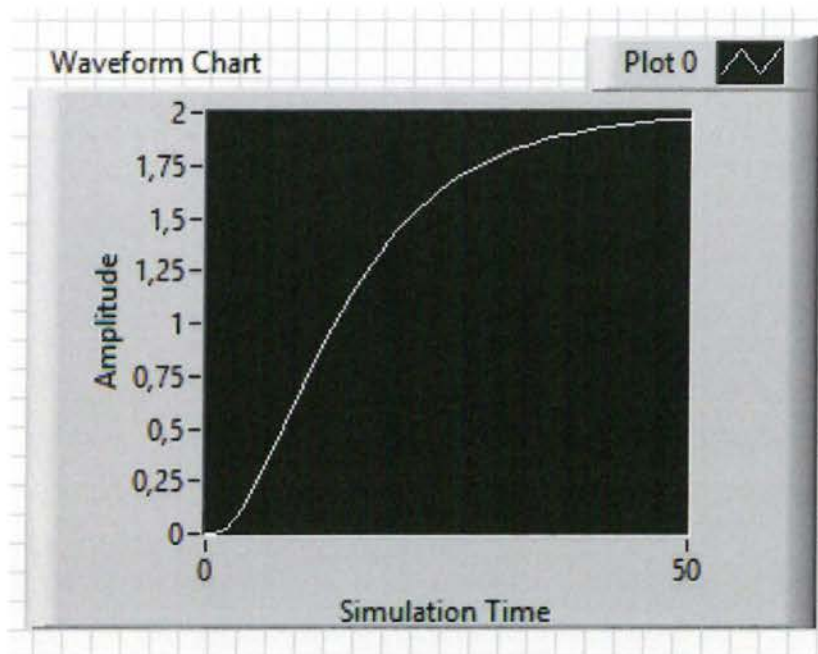


Εικόνα 2.9.1



Εικόνα 2.9.2

Αν πατήσουμε Run το αποτέλεσμα του προγράμματός μας θα είναι:



Εικόνα 2.9.3

Υλοποιήσαμε ένα σύστημα ανοιχτού βρόγχου προσθέτοντας δύο συναρτήσεις με συνάρτηση μεταφοράς την παραπάνω.

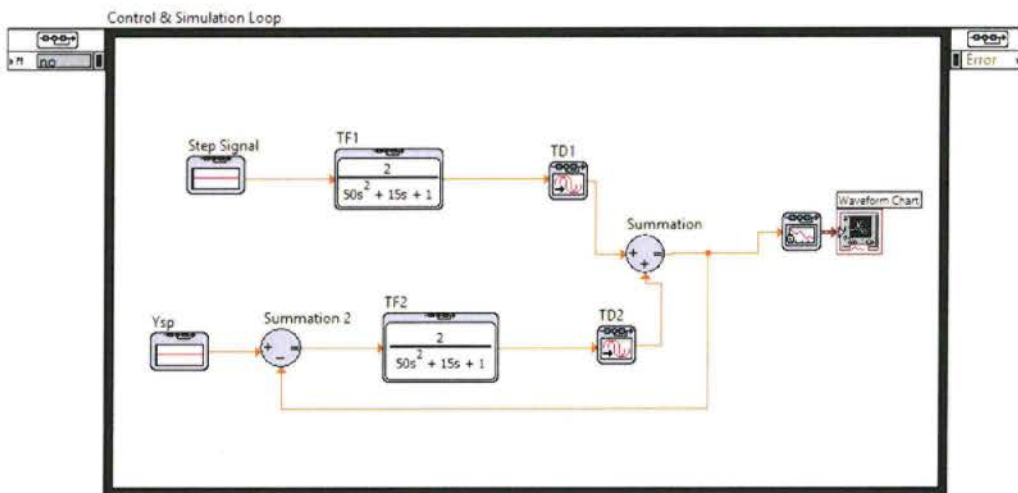
2.10 Δημιουργία συστήματος Κλειστού Βρόγχου.

Στην εφαρμογή αυτή, για την κατασκευή ενός συστήματος κλειστού βρόγχου, θα κάνουμε αλλαγές στο ήδη υπάρχον πρόγραμμα που έχουμε κατασκευάσει για τον ανοιχτό βρόγχο και έχουμε ονομάσει Open Loop.

Τα βήματα που ακολουθούμε είναι τα εξής:

1. Ανοίγουμε το VI με όνομα Open Loop.
2. Πηγαίνουμε στο block diagram (Ctrl+E).
3. Διαγράφουμε την σύνδεση του Step Signal 2 με το TF2.
4. Μετανομαζουμε το Step Signal 2 σε Ysp.
5. Κάνουμε αντιγραφή-επικόλληση το Summation και κάνουμε το Operand2 μύων.
6. Συνδέουμε την έξοδο output του Ysp με την Operand1 του Summation 2
7. Συνδέουμε το Result του Summation 2 με την είσοδο input του TF2.
8. Συνδέουμε το Operand2 του Summation 2 με το Result του Summation.

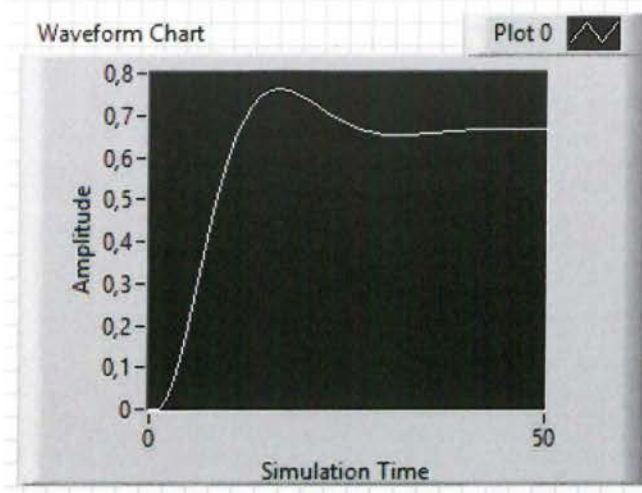
Το block diagram μετά τα παραπάνω βήματα είναι:



Εικόνα 2.10.1

Πατώντας το πλήκτρο Run το αποτέλεσμα αυτή την φορά του κλειστού βρόγχου με

συνάρτηση $TF(s) = \frac{2}{50s^2 + 15s + 1}$ θα είναι:



Εικόνα 2.10.2

Όπως βλέπουμε το σήμα σταθεροποιήτε στον χρόνο μετά την ανάδραση.

2.11 PID Controller σε σύστημα Κλειστού Βρόγχου

Όπως και ο κλειστός βρόγχος δημιουργήθηκε πάνω στον ανοιχτό, έτσι και ο PID Controller θα δημιουργηθεί πάνω σε αυτόν. Αρχικά πρέπει να πούμε ότι το LabView έχει πολλά PID Toolkit. Το PID που χρησιμοποιούμε το βρήκαμε στο Search του LabVIEW. Αυτό που θέλουμε είναι να ελέγχουμε την γραφική παράσταση του κλειστού βρόγχου μεταβάλλοντας τις τιμές των τριών controller, Proportional Gain (K_c), Integral time [s] (T_i) και Derivative time [s] (T_d).

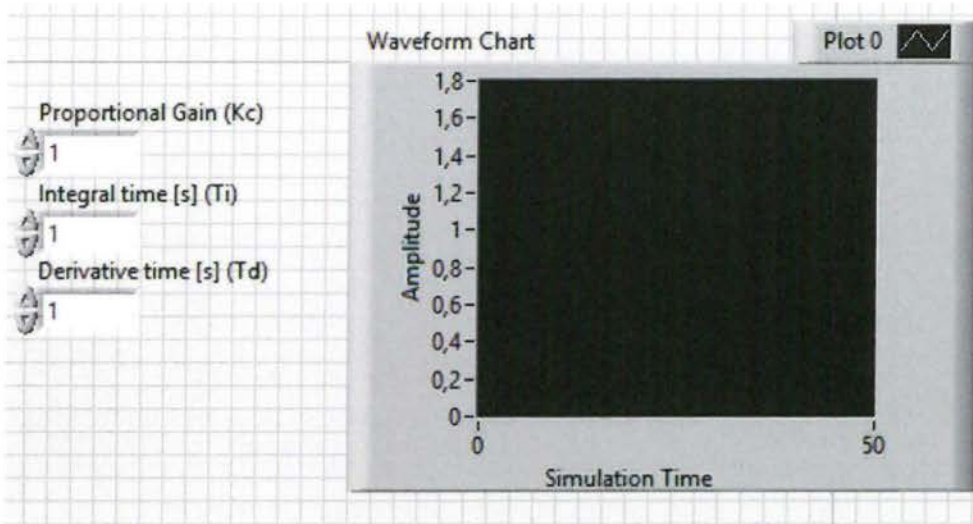
Τα βήματα που ακολουθούμε είναι τα εξής:

1. Ανοίγουμε τον VI με όνομα Closed Loop (κλειστός βρόγχος)
2. Πηγαίνουμε στο block diagram (Ctrl+E).
3. Πατάμε PID στο Search του Labview, βρίσκουμε το πρώτο PID και το τοποθετούμε στο πλαίσιο του Control & Simulation Loop
4. Πατάμε διπλό-click στο PID και κάνουμε όλες τις παράμετρους Terminal έτσι ώστε να τις ελέγχουμε από το front panel.
5. Διαγράφω την σύνδεση του Y_{sp} με το TF1.
6. Συνδέω το Y_{sp} με την είσοδο του PID και την έξοδο του PID με την είσοδο του TF2.

7. Κάνω δεξί-click στο PID στις εισόδους Proportional Gain (Kc), Integral time [s] (Ti) και Derivative time [s] (Td) και πάω Crate>>Control στο καθένα.

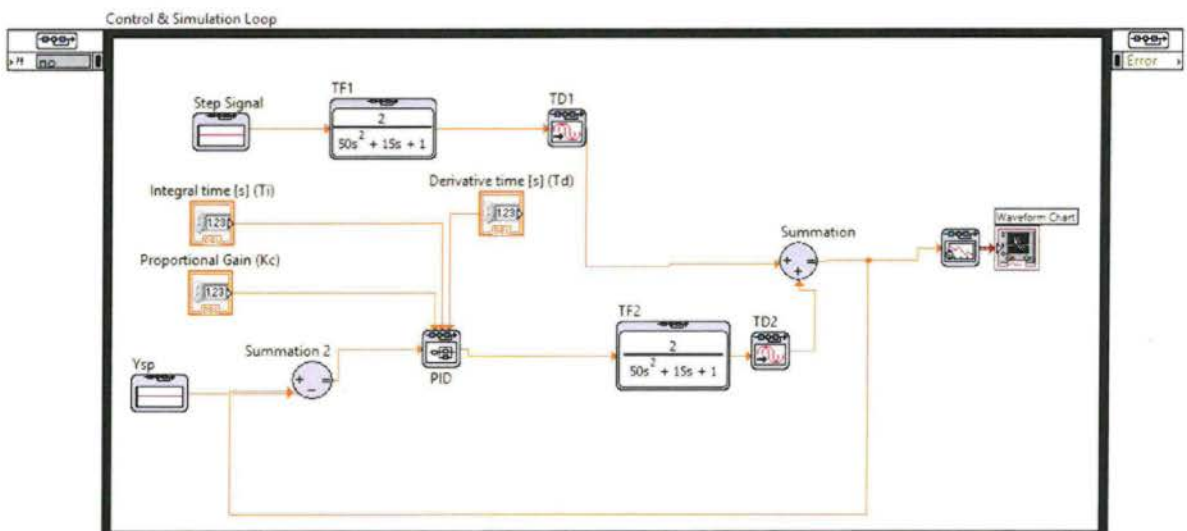
8. Πηγαίνουμε στο Front Panel (Ctrl+E).

όπως βλέπουμε εμφανίστηκαν οι τρεις έλεγχει του PID Controller



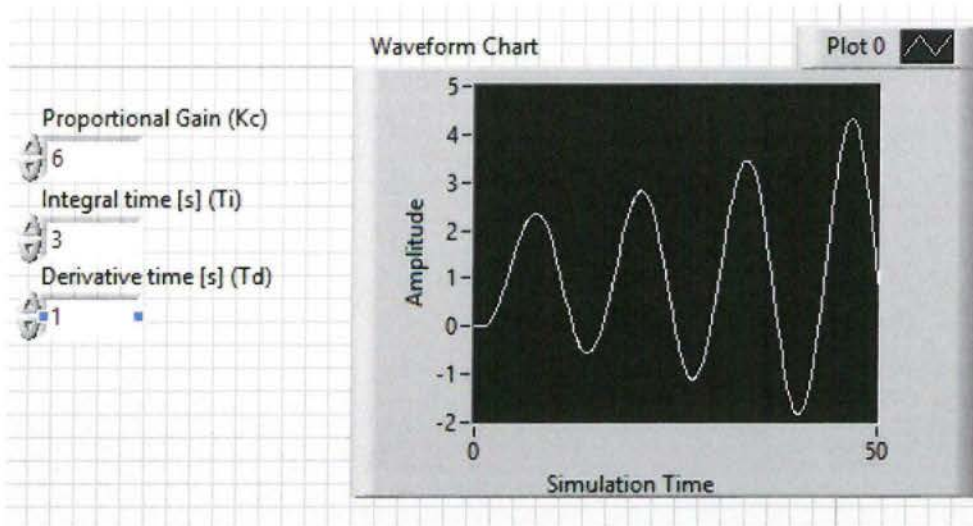
Εικόνα 2.11.1

Το block diagram μετά τα παραπάνω βήματα είναι:



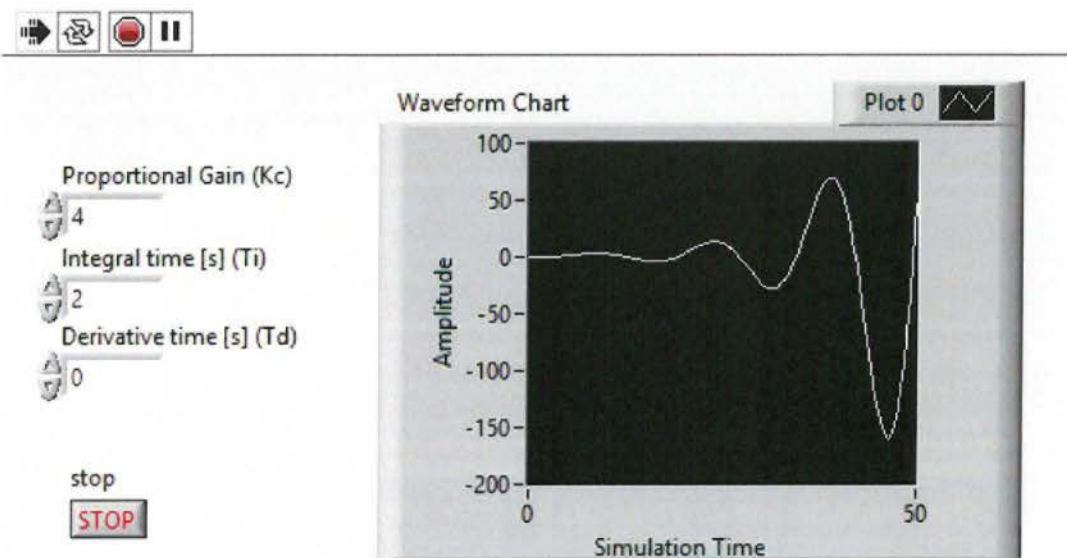
Εικόνα 2.11.2

Ο έλεγχος όπως βλέπουμε γίνεται αφού γίνει η ανάδραση. Αν αλλάξουμε τιμές στις παραμέτρους στο front panel και πατήσουμε Run το αποτέλεσμα μας θα είναι:

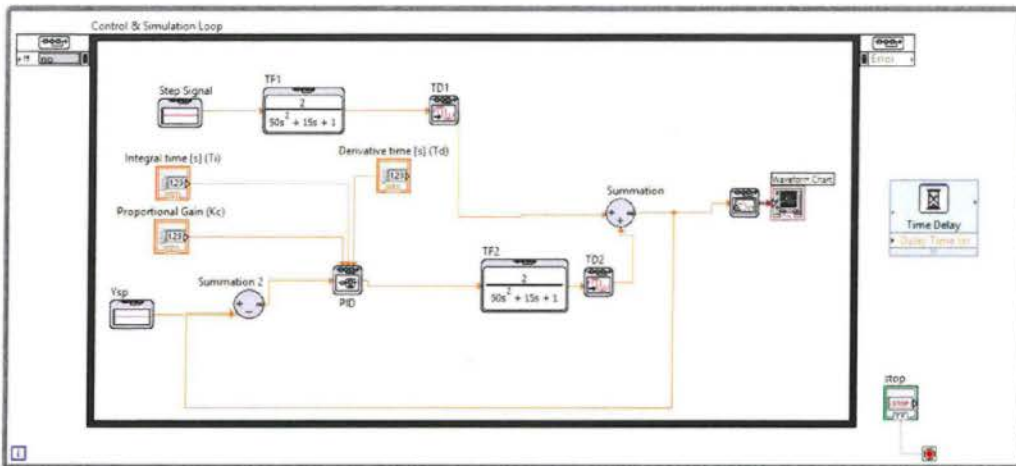


Εικόνα 2.11.3

Αν θέλουμε να μεταβάλλουμε συνεχώς τις τιμές του εκλεκτή PID, βλέποντας το αποτέλεσμα, τότε πηγαίνουμε στο block diagram και περικυκλώνουμε το Control & Simulation Loop με ένα While Loop, πηγαίνοντας στο Programming>> Structures >> While Loop. Τοποθετούμε και μία καθυστέρηση πηγαίνοντας Express>> Execution Control>> Time Delay και την τοποθετούμε μέσα στο While Loop. Το αποτέλεσμα μας θα είναι:



Εικόνα 2.11.4



Εικόνα 2.11.5

2.12 Γεωμετρικός Τόπος Ριζών

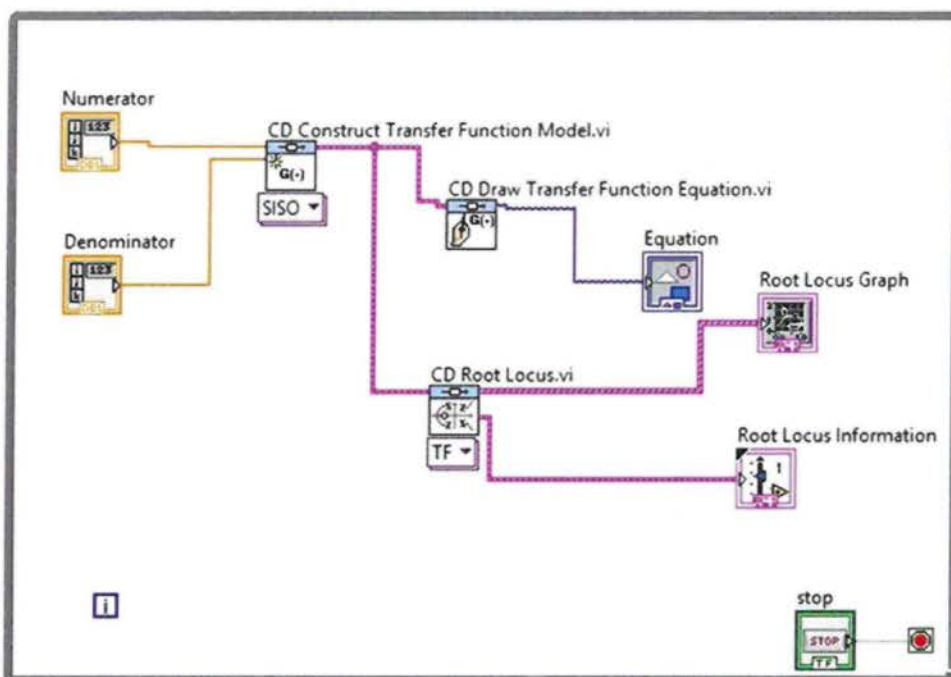
Στο συγκεκριμένο παράδειγμα σχεδιάζουμε τον Γεωμετρικό Τόπο Ριζών δίνοντας την χαρακτηριστική εξίσωση στο front panel.

Τα βήματα που ακολουθούμε είναι:

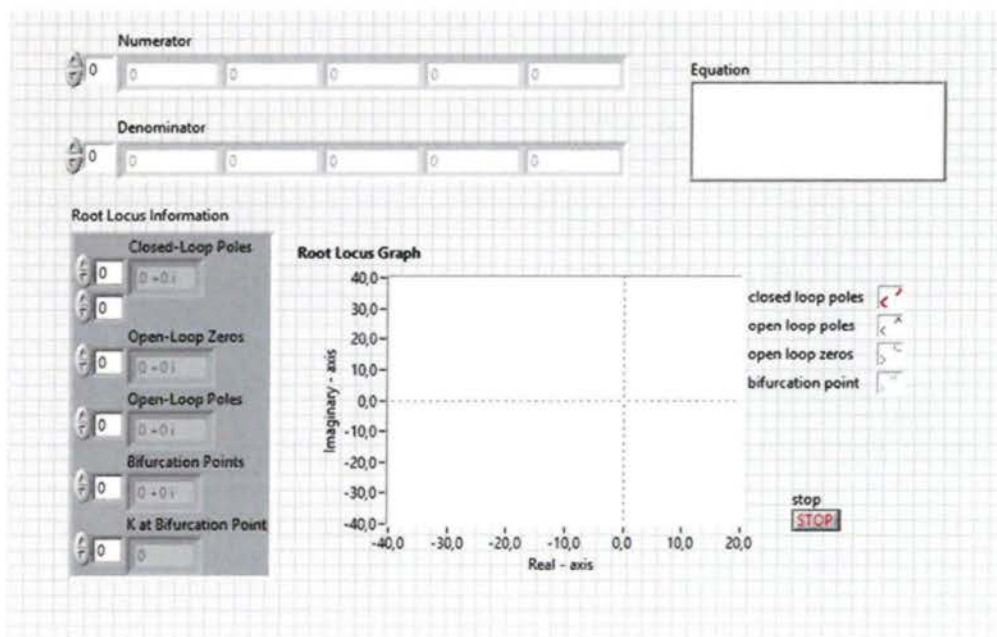
1. Από την οθόνη “Getting Started” ανοίγουμε ένα νέο “Blank VI”.
2. Πηγαίνουμε στο block diagram (Ctrl+E).
3. Πατάμε δεξί-κλικ οπουδήποτε στο block diagram και για να εμφανιστεί η παλέτα Functions.
4. Από την παλέτα Functions πηγαίνουμε στο Control Design & Simulation >> Control Design >> Model Construction >> CD Construct Transfer Function Model.vi, επιλέγουμε και το τοποθετούμε στο block diagram.
5. Δεξί-κλικ στην είσοδο Numerator του CD Construct Transfer Function Model.vi και επιλέγουμε Create >> Control.
6. Δεξί-κλικ στην είσοδο Denominator του CD Construct Transfer Function Model.vi και επιλέγουμε Create >> Control.
7. Από την παλέτα Functions πηγαίνουμε στο Control Design & Simulation >> Control Design >> Model Construction >> CD Draw Transfer Function Equation.vi, επιλέγουμε και το τοποθετούμε στο block diagram.

8. Δεξί-κλικ στην έξοδο Equation του CD Draw Transfer Function Equation.vi και επιλέγουμε Create >> Indicator για να δημιουργηθεί το Equation.
9. Συνδέουμε την έξοδο Transfer Function Model του CD Construct Transfer Function Model.vi στην είσοδο Transfer Function Model του CD Draw Transfer Function Equation.vi
10. Από την παλέτα Functions πηγαίνουμε στο Control Design & Simulation >> Control Design >> Dynamic Characteristics >> CD Root Locus.vi, επιλέγουμε και το τοποθετούμε στο block diagram.
11. Στην επιλογή που έχουμε στο CD Root Locus.vi διαλέγουμε Transfer Function
12. Συνδέουμε την είσοδο Transfer Function Model του CD Root Locus.vi στην έξοδο Transfer Function Model του CD Construct Transfer Function Model.vi
13. Δεξί-κλικ στην έξοδο Root Locus Graph του CD Root Locus.vi και επιλέγουμε Create >> Indicator για να δημιουργηθεί το Root Locus Graph.
14. Δεξί-κλικ στην έξοδο Root Locus Information του CD Root Locus.vi και επιλέγουμε Create >> Indicator για να δημιουργηθεί το Root Locus Information.
15. Από την παλέτα Functions πηγαίνουμε στο Programming>> Structures >> While Loop , και περικυκλώνουμε ότι έχουμε κάνει στο block diagram.
16. Στην έξοδο της επιλογής Loop Condition του While Loop που δημιουργήσαμε κάνουμε δεξί-κλικ και επιλέγουμε Create Control ώστε να δημιουργηθεί το Stop Button στο Front Panel.

Στο σχήμα απεικονίζεται το block diagram και το front panel, μετά την ολοκλήρωση των παραπάνω βημάτων:



Εικόνα 2.12.1

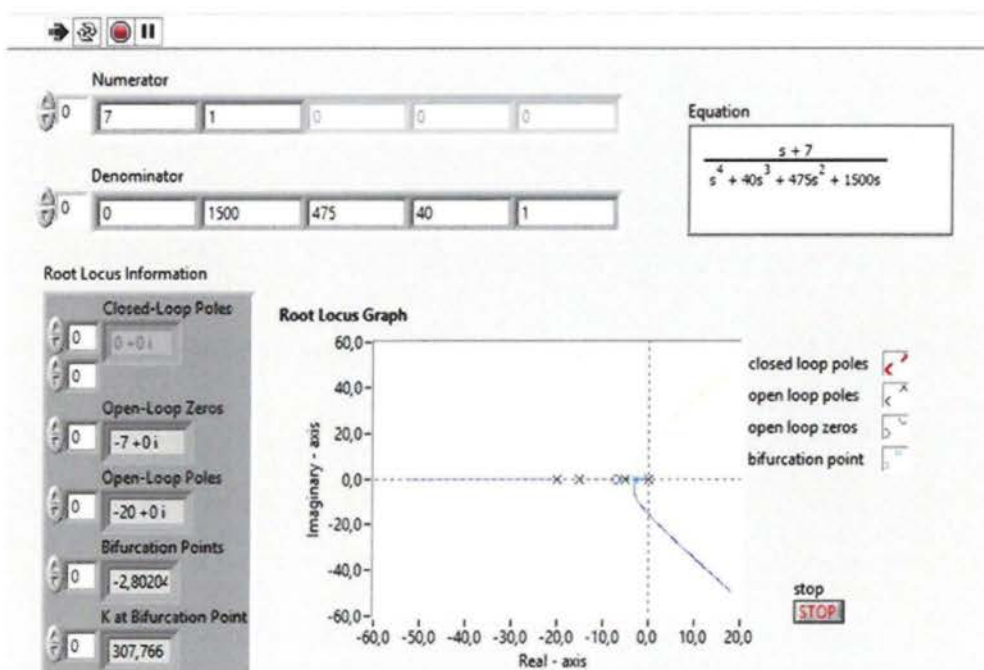


Εικόνα 2.12.2

Στο Front Panel δίνουμε στο τιμές στο Numerator (αριθμητής) και στο Denominator (παρονομαστής) και πατάμε το πλήκτρο Run. Παράδειγμα για την συνάρτηση μεταφοράς:

$$H(s) = \frac{Y(s)}{U(s)} = \frac{s + 7}{s^4 + 40s^3 + 475s^2 + 1500s}$$

Το αποτέλεσμα της σχεδίασης θα είναι :



Εικόνα 2.12.3

2.13 Δημιουργία SubVI

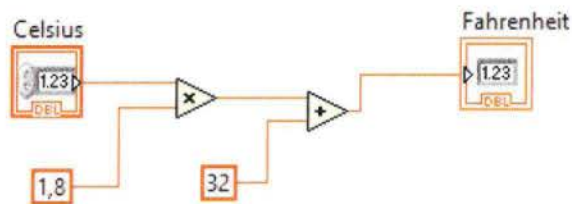
Όπως έχουμε καταλάβει μέχρι τώρα το LabVIEW μπορεί να δημιουργήσει SubVI δηλαδή ΥποVI. Το κλειδί στην χρήση του περιβάλλοντος LabVIEW είναι η κατανόηση της ιεραρχικής φύσης ενός Εικονόργανου. Με άλλα λόγια, μετά την δημιουργία ενός Εικονόργανου(VI), μπορείτε να το χρησιμοποιήσετε ως υπόVI(subVI) στο Μπλόκ Διάγραμμα ενός VI σε ανώτερο επίπεδο. Εάν ένα Μπλόκ Διάγραμμα έχει μεγάλο αριθμό από εικόνες μπορείτε να τις ταξινομήσετε (ανάλογα με την λειτουργικότητά τους) και να δημιουργήσετε υπόVIs με την ίδια λειτουργικότητα ώστε να απλοποιήσετε το Διάγραμμα σας. Αυτή η προσέγγιση διευκολύνει την κατανόηση, τροποποίηση και αναβάθμιση εφαρμογών στο LabVIEW. Ένας ΥπόVI είναι ανάλογος μίας υπορουτίνας σε άλλα περιβάλλοντα προγραμματισμού.

Στο παράδειγμα μας θα κατασκευάσουμε ένα απλό SubVI, το οποίο θα μετατρέπει τους βαθμούς Κελσίου σε Φαρενάιτ, και θα το συνδέσουμε στην δραστηριότητα που έχουμε δημιουργήσει για το θερμόμετρο στο 2.1. Σας θυμίζουμε ότι για να μετατρέψουμε τους βαθμούς Κελσίου, πολλαπλασιάζουμε τους βαθμούς με το 1,8 και στο γινόμενο προσθέτουμε το 32.

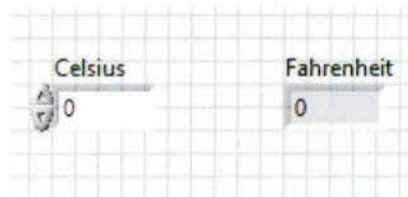
Τα βήματα που ακολουθούμε είναι:

1. Από την οθόνη “Getting Started” ανοίγουμε ένα νέο “Blank VI”.
2. Πηγαίνουμε στο front panel (Ctrl+E).
3. Πατάμε δεξί-κλικ οπουδήποτε στο front panel για να εμφανιστεί η παλέτα Controls.
4. Από την παλέτα Controls πηγαίνουμε στο Modern>>Numeric και τοποθετούμε ένα Numeric Indicator στο front panel και του δίνουμε το όνομα «Fahrenheit».
5. Από την παλέτα Controls πηγαίνουμε στο Modern>>Numeric και τοποθετούμε ένα Numeric Control στο front panel και του δίνουμε το όνομα «Celsius».
6. Πηγαίνουμε στο block diagram (Ctrl+E).
7. Πατάμε δεξί-κλικ οπουδήποτε στο block diagram και για να εμφανιστεί η παλέτα Functions.

8. Από την παλέτα Functions πηγαίνουμε στο Programming>>Numeric και τοποθετούμε ένα Multiply στο block diagram.
9. Από την παλέτα Functions πηγαίνουμε στο Programming>>Numeric και τοποθετούμε ένα Add στο block diagram.
10. Συνδέουμε την έξοδο του Celsius με τη είσοδο x του Multiply.
11. Κάνουμε δεξί-click στην είσοδο y του Multiply και πάμε Create>>Constant, και δίνουμε την τιμή 1,8.
12. Συνδέουμε την έξοδο x*y του Multiply με την είσοδο x του Add.
13. Κάνουμε δεξί-click στην είσοδο y του Add και πάμε Create>>Constant, και δίνουμε την τιμή 32.
14. Συνδέουμε την έξοδο x+y του Add με την είσοδο του Fahrenheit.

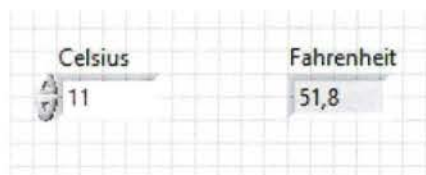


Εικόνα 2.13.1



Εικόνα 2.13.2

Αν μεταβάλουμε τις τιμές στο front panel και πατήσουμε το κουμπί Run βλέπουμε ότι το πρόγραμμά μας λειτουργεί.




Εικόνα 2.13.3

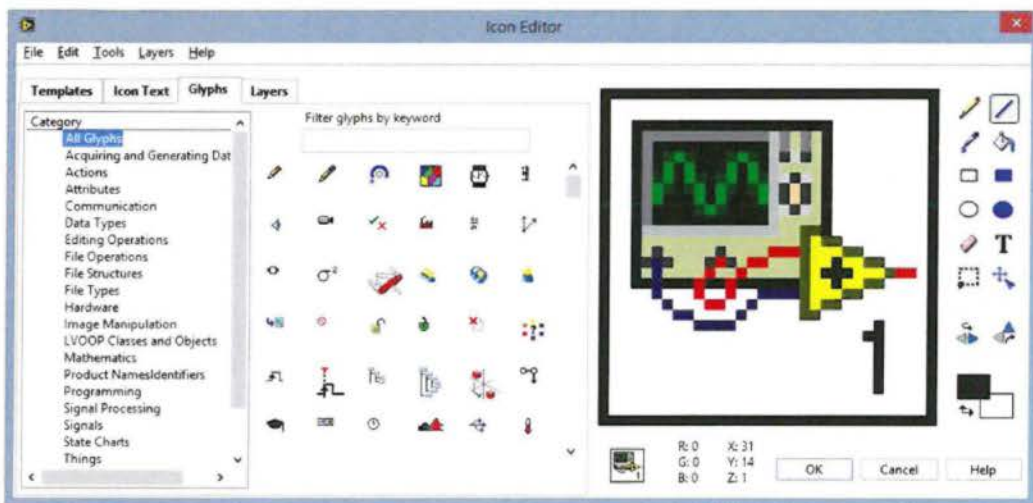
Αφού κάνουμε όλα τα παραπάνω βήματα είναι στιγμή να φτιάξουμε το SubVI.

Αρχικά πρέπει να φτιάξουμε την εικόνα του SubVI και στην συνέχεια τους κοννέκτορές τις.


15. Πηγαίνουμε στο front panel (Ctrl+E).

16. Πάνω δεξιά υπάρχει μια εικόνα με το σήμα του LabVIEW . Κάνουμε δεξί-click και πατάμε Edit Icon.

17. Στο παράθυρο Icon Editor που μας εμφανίζεται πάμε στο μενού Glyphs και φτιάχνουμε την εικόνα που θέλουμε για το SubVI (μία σύντομη περιγραφή της λειτουργίας των σχεδιαστικών Εργαλείων βρίσκεται στο Παράρτημα την πτυχιακής)



18. Αφού κάνουμε το σχέδιο που θέλουμε εμφανίζεται στο front panel μας.

Πάμε στην εικόνα  κάνουμε δεξί-click, πάμε Patterns και διαλέγουμε πως θέλουμε να τοποθετήσουμε τους κοννέκτορες δηλαδή στις εισόδους και τις εξόδους του SubVI μας.

19. Διαλέγουμε που θέλουμε να τοποθετηθεί η είσοδό μας Celsius και κάνουμε click πάνω σε αυτή και αντίστοιχα το ίδιο για το Fahrenheit.

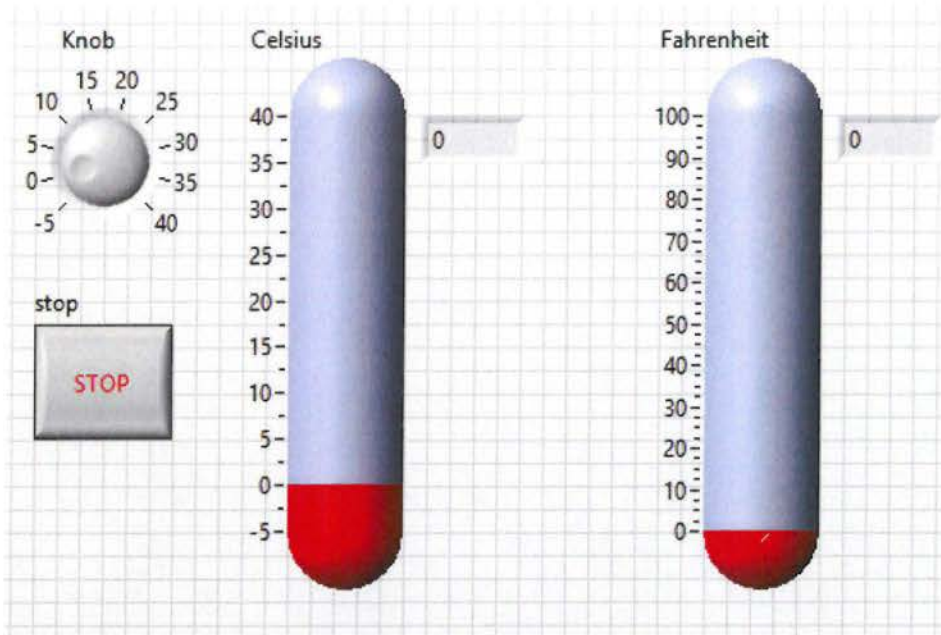
20. Αν θέλουμε από το μενού File>>VI Properties και βάζουμε πληροφορίες για το SubVI μας καθώς και σε κάθε όργανό μας ξεχωριστά κάνοντας δεξί-click και διαλέγοντας Description and Tip. Αυτά θα βοηθήσουν στην βοήθεια (Help) του LabVIEW αν το θελήσει κάποιος χρήστης.

21. Αποθηκεύουμε το SubVI μας σε όποιο φάκελο θελήσουμε σαν CtoF.

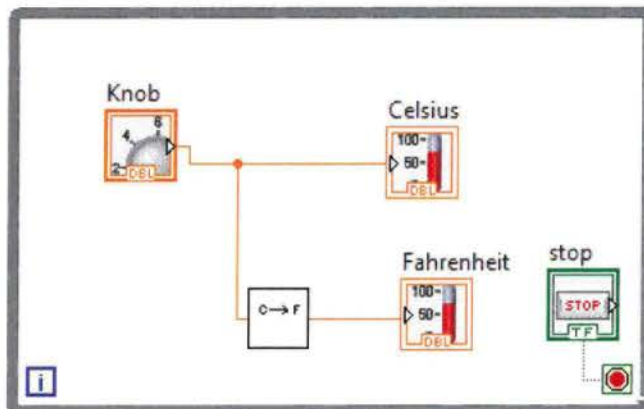
Έχουμε δημιουργήσει το SubVI μας και είμαστε έτοιμοι να το χρησιμοποιήσουμε. Θα δημιουργήσουμε ένα εικονόργανο το οποίο θα προσομοιώνει δύο θερμομέτρα, τα οποία θα μεταβάλλονται με έναν αντιστροφέα.

22. Από την οθόνη “Getting Started” ανοίγουμε ένα νέο “Blank VI”.
23. Πηγαίνουμε στο front panel (Ctrl+E)
24. Πατάμε δεξί-κλικ οπουδήποτε στο front panel για να εμφανιστεί η παλέτα Controls.
25. Από την παλέτα Controls πηγαίνουμε Modern>>Numeric και διαλέγουμε Knob.
26. Από την παλέτα Controls πηγαίνουμε Modern>>Numeric και διαλέγουμε δύο Thermometer.
27. Δίνουμε στο ένα το όνομα Celsius και στο άλλο το όνομα Fahrenheit.
28. Διαμορφώνουμε το Front Panel έτσι ώστε να είναι ευδιάκριτα τα γραφήματα μας.
29. Πηγαίνουμε στο block diagram (Ctrl+E).
30. Πατάμε δεξί-κλικ οπουδήποτε στο block diagram για να εμφανιστεί η παλέτα Functions.
31. Από την παλέτα Functions πηγαίνουμε στο Programming>> Structures >> While Loop , και περιβάλλουμε ότι έχουμε στο block diagram.
32. Κάνουμε δεξί-click στο Loop Condition και πατάμε Create>>Control.
33. Από την παλέτα Functions πηγαίνουμε στο Select VI βρίσκουμε το CtoF και το τοποθετούμε στο While Loop.
34. Συνδέουμε την έξοδο του Knob με την είσοδο του Celsius.
35. Συνδέουμε την είσοδο Celsius του CtoF με την είσοδο του Knob.
36. Συνδέουμε την έξοδο Fahrenheit του CtoF με το Fahrenheit.

Το Front Panel και το Block Diagram μετά την ολοκλήρωση των παραπάνω βημάτων είναι:

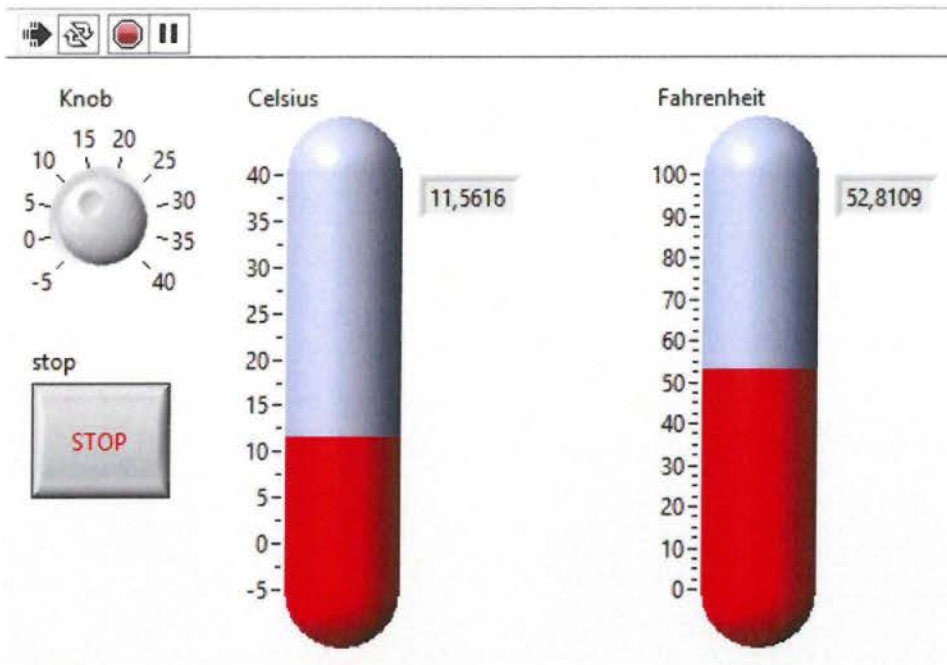


Εικόνα 2.13.4



Εικόνα 2.13.5

Αν πατήσουμε Run και μεταβάλουμε το ποτενσιόμετρο το αποτέλεσμα του προγράμματός μας σε μια τυχαία στιγμή θα είναι:



Εικόνα 2.13.6

Όπως βλέπουμε μέσω του CtoF δηλαδή του SubVI μεταβάλλεται η θερμοκρασία σε βαθμούς Κελσίου μεταβάλλοντας το σε Φαρενάιτ.

ΚΕΦΑΛΕΟ 3:

Ανάπτυξη & Προγραμματισμός

Συστημάτων Συλλογής Δεδομένων

3.1 Συστήματα συλλογής δεδομένων & LabVIEW

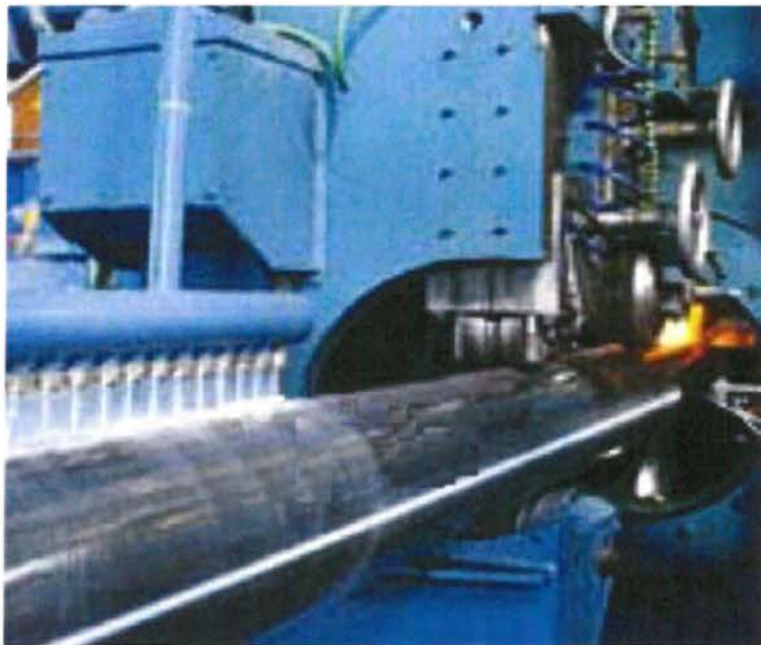
Η σχεδίαση και ανάπτυξη ηλεκτρονικών διατάξεων για την πραγματοποίηση μετρήσεων τεχνικών μεγεθών ορίζει τα *ηλεκτρονικά συστήματα μετρήσεων*. Τα ηλεκτρονικά συστήματα μετρήσεων αποτελούν ένα από τα κυριότερα τμήματα ενός συστήματος ελέγχου διότι σε μεγάλο βαθμό η ευστοχία του συστήματος αποτελεί συνάρτηση της ακρίβειας των μετρήσιμων μεγεθών. Σήμερα, με την εξέλιξη της τεχνολογίας μιλάμε για συλλογή δεδομένων (Data Acquisition, DAQ).

Συλλογή δεδομένων καλείται η μέθοδος μέσω της οποίας μεγέθη όπως: η τάση, το ρεύμα, η πίεση, η ροή, η θερμοκρασία, κ.α., συλλέγονται, απεικονίζονται, και καταγράφονται μέσω υπολογιστικής μονάδας. Για το σκοπό αυτό ένα *σύστημα συλλογής δεδομένων* (Data Acquisition System) μετατρέπει το εισερχόμενο σήμα σε ψηφιακό δεδομένο. Υπεύθυνο για την επεξεργασία των αποκτημένων πληροφοριών είναι το λογισμικό που φέρει το σύστημα DAQ μέσω του οποίου εκτελούνται διεργασίες όπως: μετρήσεις ηλεκτρικών μεγεθών, ανάλυση σήματος στο πεδίο της συχνότητας/ χρόνου, στατιστική ανάλυση σήματος, και πλήθος άλλων επεξεργασιών.

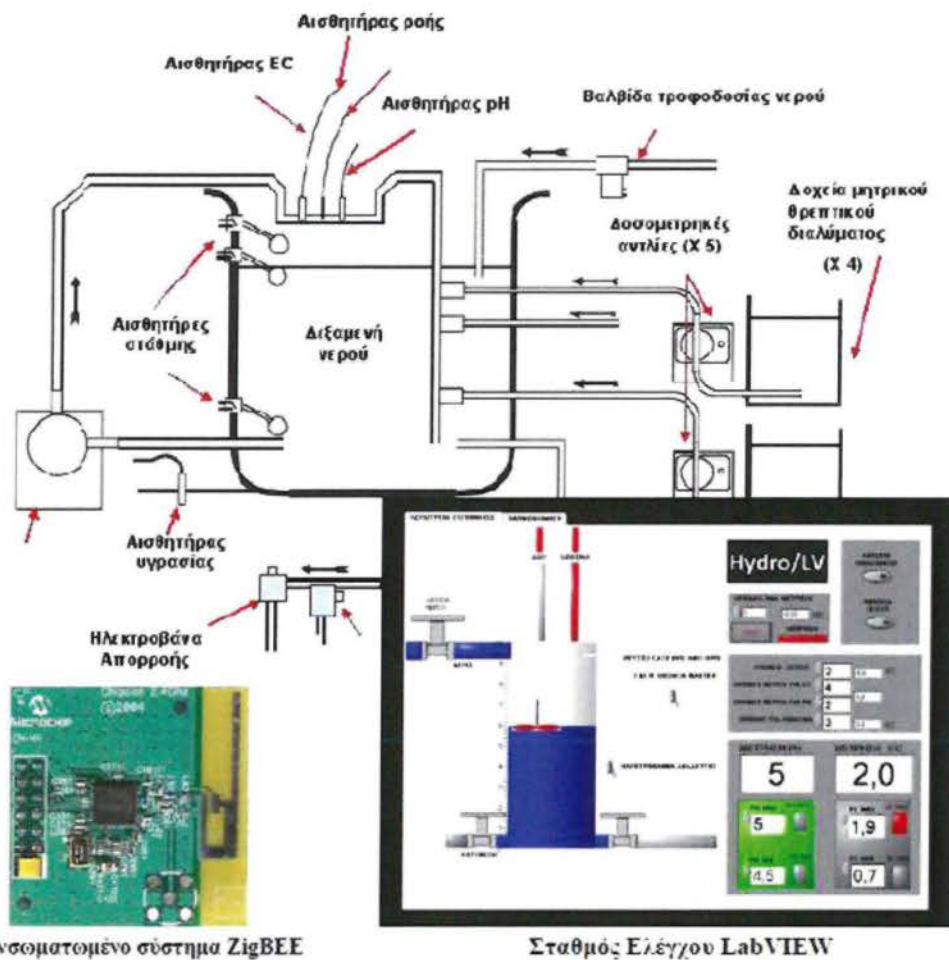
Με τη χρησιμοποίηση μονάδων υψηλής τεχνολογίας, και λογισμικών επεξεργασίας σημάτων, όπως το LabVIEW μπορεί να επιτευχθεί υψηλή ακρίβεια μέτρησης και ελέγχουτης πληροφορίας του σήματος σε εργαστηριακή ή βιομηχανική εφαρμογή. Το λογισμικό LabVIEW (Laboratory Virtual Instrument Engineering Workbench) είναι μία πανίσχυρη γλώσσα προγραμματισμού μετρήσεων, έλεγχου και ανάλυσης για συστήματα συλλογής δεδομένων. Σε αυτή τη γραφική γλώσσα προγραμματισμού που καλείτε “**G**”, ο προγραμματισμός επιτυγχάνεται σε μπλοκ διαγράμματα (block diagram). Μέσω του LabVIEW που συνδυάζει λογισμικό και υλικό για την απόκτηση δεδομένων (data acquisition) και έλεγχου εφαρμογών (system control) μέσω καρτών-μονάδων (devices) ο μηχανικός μπορεί να αναπτύξει το δικό του σύστημα μετρήσεων και ελέγχου καλύπτοντας

πλήρως τις απαιτήσεις του. Η μορφή των εντολών του LabVIEW δεν περιλαμβάνει χρήση λεκτικών εντολών, αλλά γραφικών. Με τη χρησιμοποίηση των γραφικών εντολών αναπτύσσουμε τον κώδικα της εφαρμογής μας κατά τον οποίο μπορούμε να κάνουμε συλλογή, έλεγχο, επεξεργασία και καταγραφή των μετρήσεων προκειμένου του ελέγχου του συστήματος μας.

Σε συνδυασμό με το αυτοματοποιημένο λογισμικό εργαλείο DAQ Assistant στο περιβάλλον του LabVIEW μπορούμε να προγραμματίσουμε και να ελέγξουμε λειτουργίες των συσκευών όπως: ρυθμό δειγματοληψίας (Sample Rate) αναλογικών I/O, καθορισμός ορίων I/O (Voltage limit), καθορισμός τρόπου σύνδεσης I/O (RSE, DIFF, NRSE), προσδιορισμός χρονιστών και ψηφιακών I/O, καθορισμό τεχνικού μεγέθους μέτρησης, και πλήθος άλλων επιλογών σε κάθε συγκεκριμένη συσκευή DAQ. Για την ταχύτερη ανάπτυξη εφαρμογών το LabVIEW παρέχει ειδικά εικονικά όργανα τα οποία καλούνται Express VIs καθιστώντας το χρόνο προγραμματισμού και της ανάπτυξης των εφαρμογών ταχύτερο. Παρακάτω δίνονται παραδείγματα συλλογής και επεξεργασίας δεδομένων.



Ψηφιακή κάμερα φωτογραφίζει σε καρτέ (frames) μία λευκή γραμμή που είναι αποτυπωμένη πάνω στο σωλήνα. Σε κάθε απόκλιση της λευκής γραμμής από τα σημεία που έχουν καθοριστεί από το μηχανισμό επεξεργασίας πραγματικού χρόνου μέσω εντολής του συστήματος συλλογής δεδομένων διορθώνει την κλίση των σημείων επαφής του σωλήνα περιστρέφοντας τον ώστε να οδηγηθούν τα άκρα του υλικού και να συγκολληθεί η συρραφή του.



Παράδειγμα συστήματος ελέγχου διαλυμάτων σε δεξαμενή, με LabVIEW, ανάπτυξη ενσωματωμένων συστημάτων, και ασύρματων αισθητήρων ZigBee

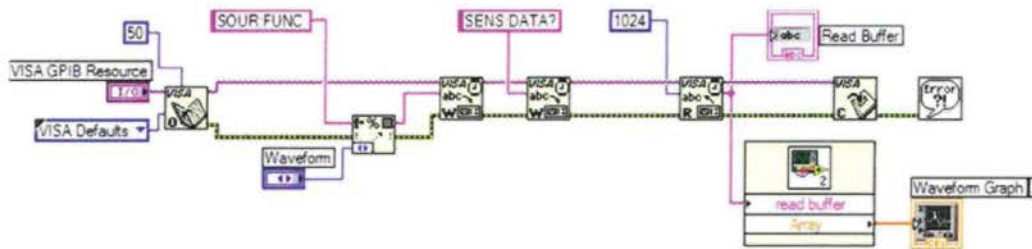
3.2 Υλικό συστημάτων μετρήσεων και ελέγχου

Για να μπορέσουμε να αποκτήσουμε να αναλύσουμε και να καταγράψουμε πραγματικά σήματα στο περιβάλλον του LabVIEW, χρειαζόμαστε συσκευές απόκτησης δεδομένων. Αυτές οι συσκευές απόκτησης δεδομένων μπορούν να συνδεθούν στον υπολογιστή μέσω των εσωτερικών διαύλων ή άλλες εξωτερικά μέσω της θύρας USB και δίνουν τη δυνατότητα λήψης μετρήσεων (acquiring), βελτίωσης σήματος (signal conditioning) ανάλυσης (analyzing), καταγραφής δεδομένων (data logger) και ελέγχου (control) συστημάτων μέσω προγραμματισμού. Το πρότυπο που παρέχει τη διασύνδεση για τον προγραμματισμό

αυτών των συστημάτων που φέρουν πρότυπα όπως GPIB, VXI, PXI, Ethernet, και USB ονομάζεται VISA.



Παράδειγμα διασύνδεσης οργάνων που φέρουν πρότυπο GPIB.



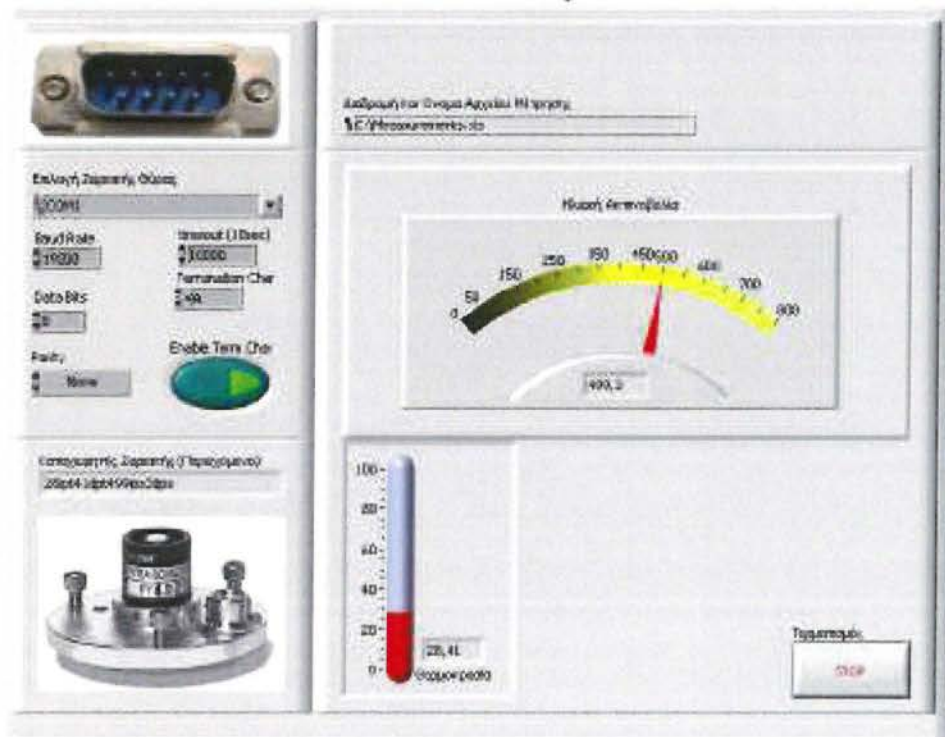
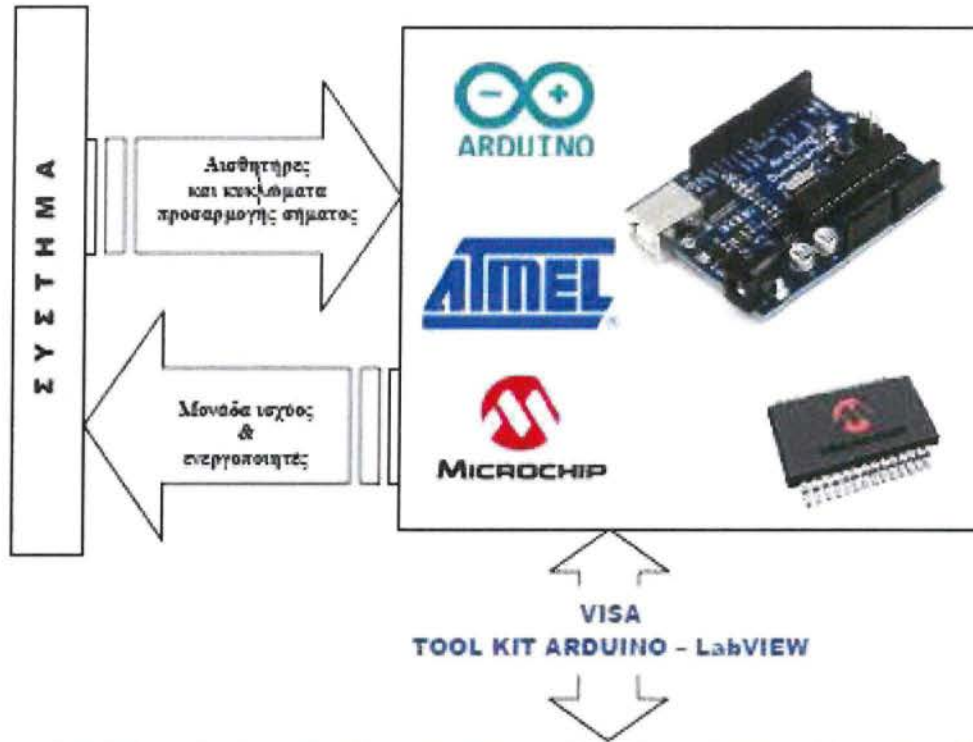
Παράδειγμα κώδικας επικοινωνίας οργάνου GPIB με το περιβάλλον LabVIEW

Όταν οι απαιτήσεις του συστήματος προσδιορίζουν αυτόνομες λειτουργίες μετρήσεων και ελέγχου χωρίς διασύνδεση με ηλεκτρονικό υπολογιστή καταφεύγουμε σε τεχνολογία FPGA για λήψη, επεξεργασία και έλεγχο σήματος σε πραγματικό χρόνο (RealTime Input Output). Χαρακτηριστικά των συστημάτων FPGA αποτελούν:

- Ο ενσωματωμένος επεξεργαστής πραγματικού χρόνου για αυτόνομη και διανεμημένη λειτουργία με ενσωματωμένο περιβάλλον διεπαφής με τον χρήστη,
- Οι αποσπώμενες ψηφιακές και αναλογικές μονάδες εισόδου/εξόδου για άμεση σύνδεση με βιομηχανικούς αισθητήρες και μετατροπείς για ανάπτυξη αυτόνομων συστημάτων μετρήσεων και ελέγχου.

Σε περιπτώσεις αυτόνομης καταγραφής (data Logger) περιβαλλοντικών μεγεθών πληροφοριακά συστήματα καταγράφουν και αποθηκεύουν δεδομένα σε ρυθμό απόκτησης που προγραμματίζει ο χρήστης, ενώ η επεξεργασία τους γίνεται με μεταφορά των πληροφοριών σε ηλεκτρονικό υπολογιστή μέσω θύρα RS232 ή USB.

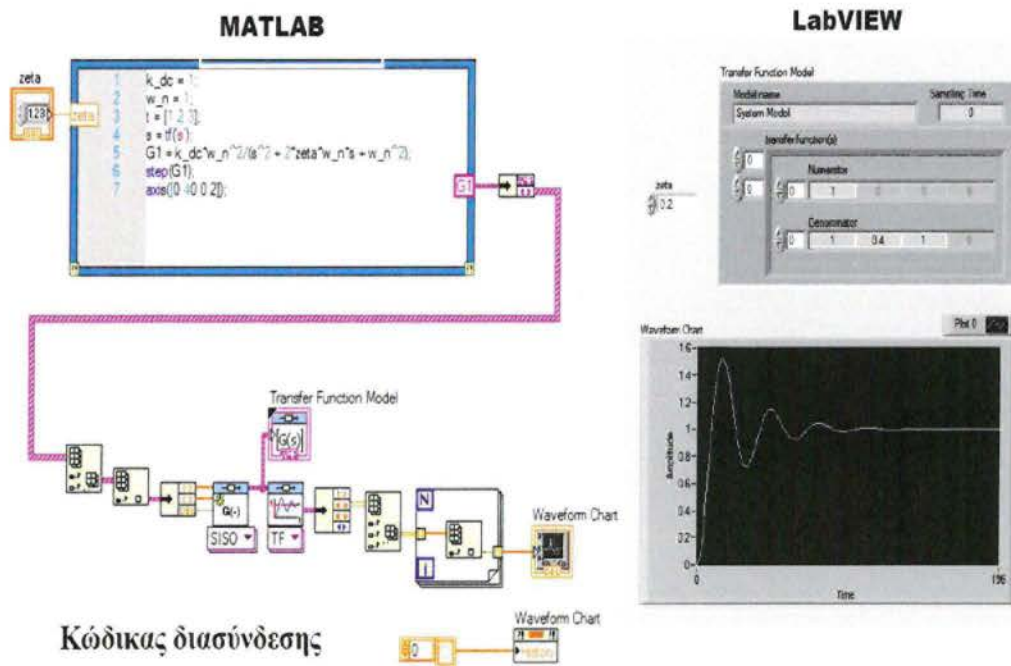
Ενσωματωμένα συστήματα και LABVIEW



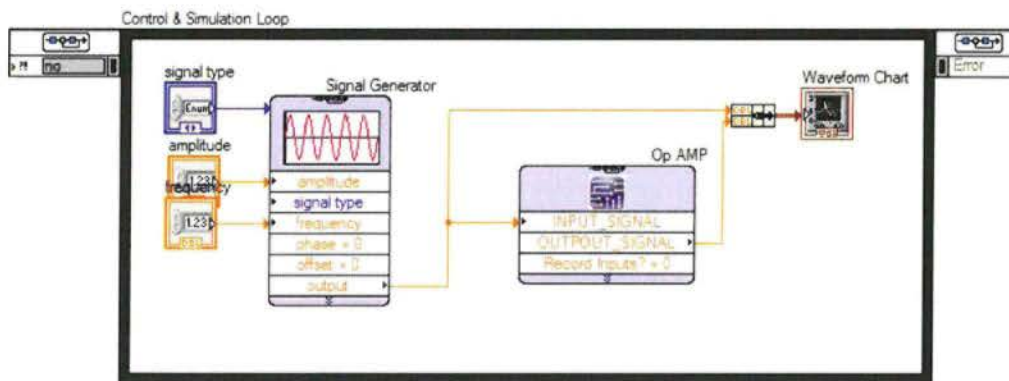
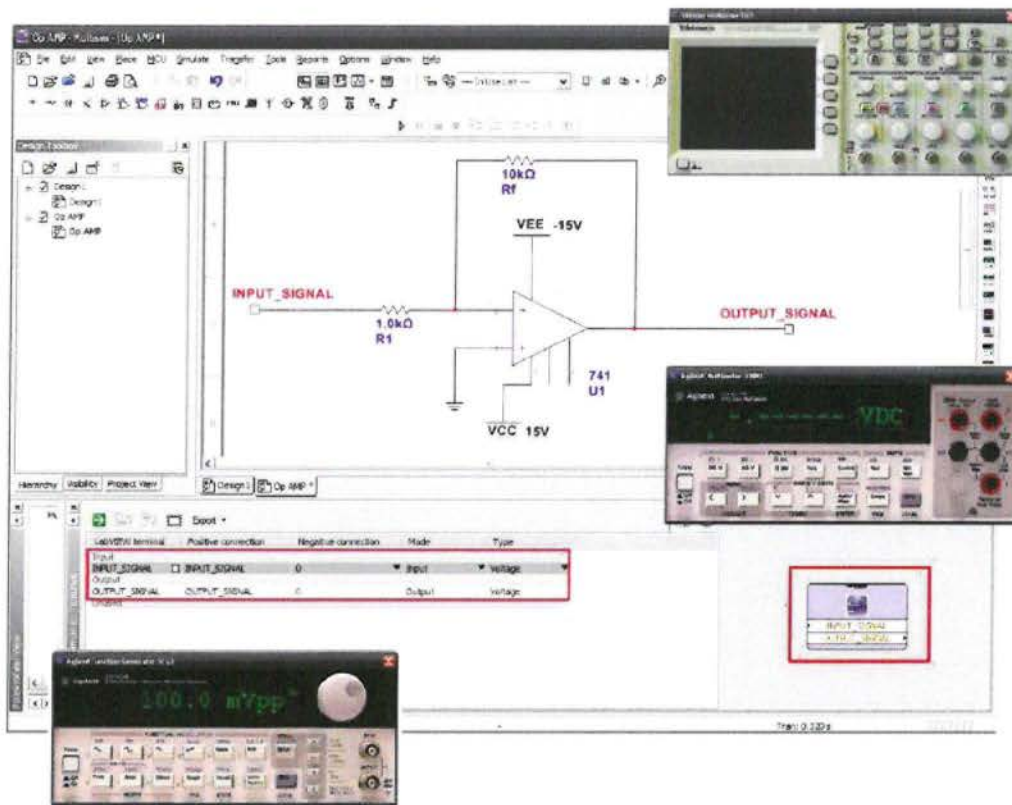
3.3 Ενσωμάτωση προγραμμάτων και συστημάτων

Η πλατφόρμα του LabVIEW πολλές φορές μπορεί να ενσωματώσει κώδικα από άλλη γλώσσα προγραμματισμού η και ακόμα σχεδιαστικό πρόγραμμα για την βελτιστοποίηση των εφαρμογών του. Παρακάτω αναφέρονται τα πιά σημαντικά:

MATLAB: Η επεξεργασία σήματος μέσω εντολών MATLAB μπορεί να ενσωματωθεί σε κώδικα LABVIEW με πραγματικά σήματα στα σύγχρονα συστήματα συλλογής δεδομένων.

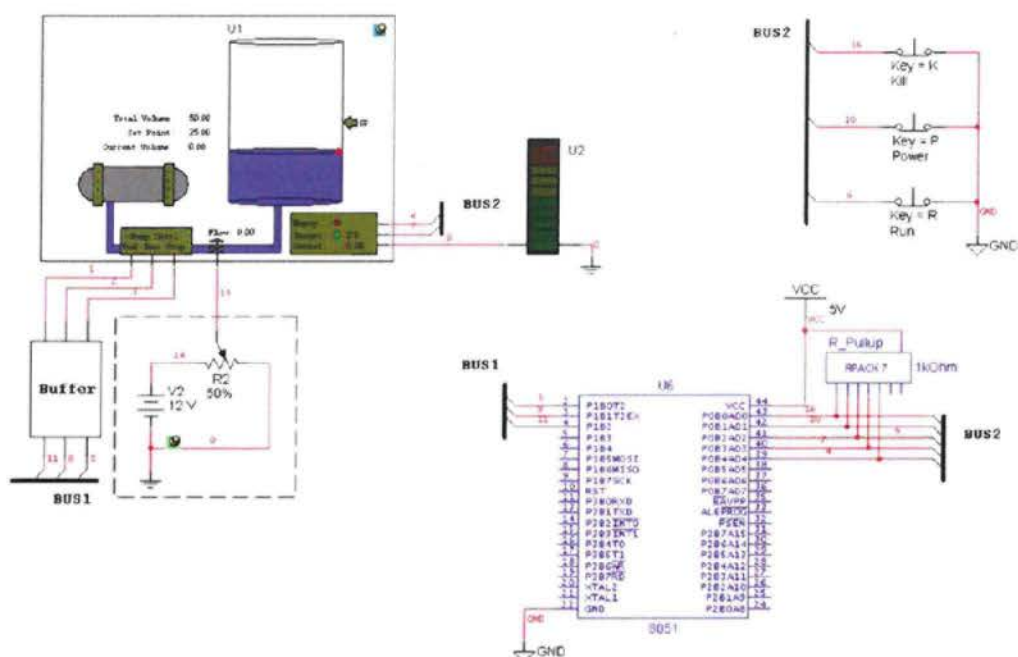


MULTISIM: Το MultiSIMTM αποτελεί λογισμικό ηλεκτρονικής σχεδίασης (Electronics Design Automation, EDA) στηριζόμενο σε μορφή SPICE και βοηθά στην πραγματοποίηση ολοκληρωμένων διαδικασιών σχεδίασης και προσομοίωσης κυκλωμάτων.



MultiVHDL : Το σύνολο των εργαλείων της συγκεκριμένης βιβλιοθήκης παρέχει τη δυνατότητα σχεδιασμού και προσομοίωσης συστημάτων που έχουν γίνει σε γλώσσα περιγραφής υλικού VHDL.

MultiMCU: Δίνει στο MultiSIM τη δυνατότητα σχεδιασμού και προσομοίωσης συστημάτων μικροεπεξεργαστών και μικροελεγκτών σε γλώσσα assembly.



3.4 Χρήση συσκευών καταγραφής δεδομένων

Η **National Instruments** έχει ένα πλήθος συσκευών για κάθε είδος Μέτρησης. Ας εξετάσουμε τη μονάδα USB 6009. Αυτή η μονάδα έχει οκτώ (8) αναλογικές εισόδους που μπορούν να συνδεθούν σε μονοκάναλη ή διαφορική σύνδεση, οκτώ (8) ψηφιακές εισοδοι/ έξοδοι και δύο (2) αναλογικές έξοδοι. Για να μπορέσουμε να εργαστούμε με τη μονάδα USB 6009 θα πρέπει να έχουμε εγκαταστήσει στον υπολογιστή μας μια πρόσφατη έκδοση του λογισμικού οδηγών μονάδων DAQmx το οποίο είναι υπεύθυνο για την αναγνώριση της μονάδας από το σύστημα του υπολογιστή. Μπορούμε να κατεβάσουμε τον οδηγό συσκευών NI-DAQmx 8.8 από την επίσημη ιστοσελίδα της **National Instruments** στο www.ni.com Καθώς συνδέουμε τη μονάδα ο DAQmx της National Instruments εμφανίζεται το παράθυρο διαλόγου ελέγχου σωστής λειτουργία της μονάδας.



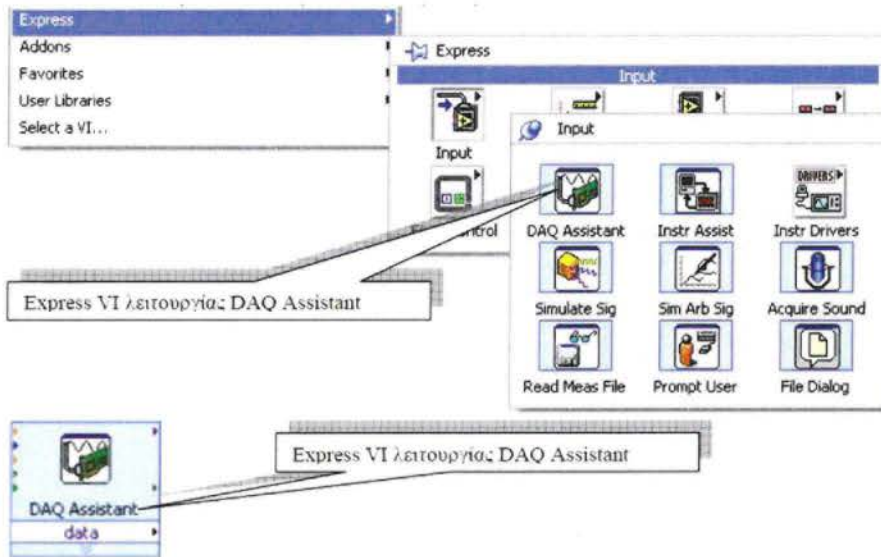
USB 6009



Μπορούμε επίσης να ελέγξουμε τη λειτουργία της μονάδας από τον MAX (measurement & Automation Explorer) στο φάκελο της National Instruments που βρίσκεται στα προγράμματα.

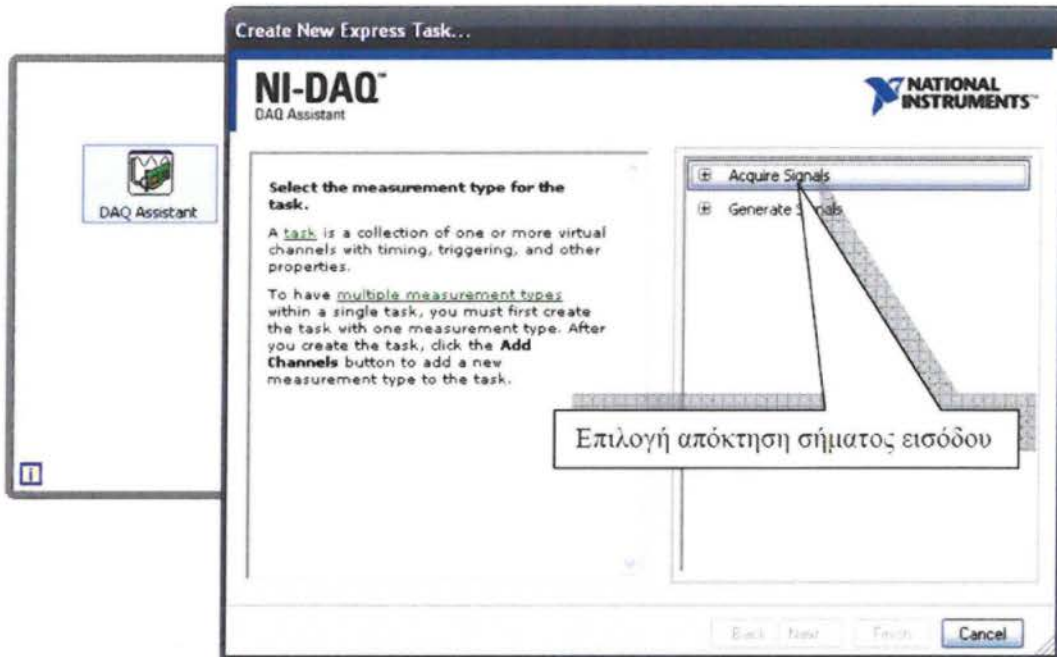
Εργασίες δραστηριότητας

Για να αποκτήσουμε δεδομένα από πραγματικό σήμα στον κώδικα του LabVIEW χρησιμοποιούμε τη λειτουργία του **DAQ Assistance** από την παλέτα **Functions**→**Express**→**Input**. Για να ακολουθήσουμε τα παρακάτω βήματα θα πρέπει πρώτα να έχουμε «κατεβάσει» και εγκαταστήσει το λειτουργικό οδηγών συσκευών **NI-DAQmx 8.8** από την επίσημη ιστοσελίδα της **National Instruments**.

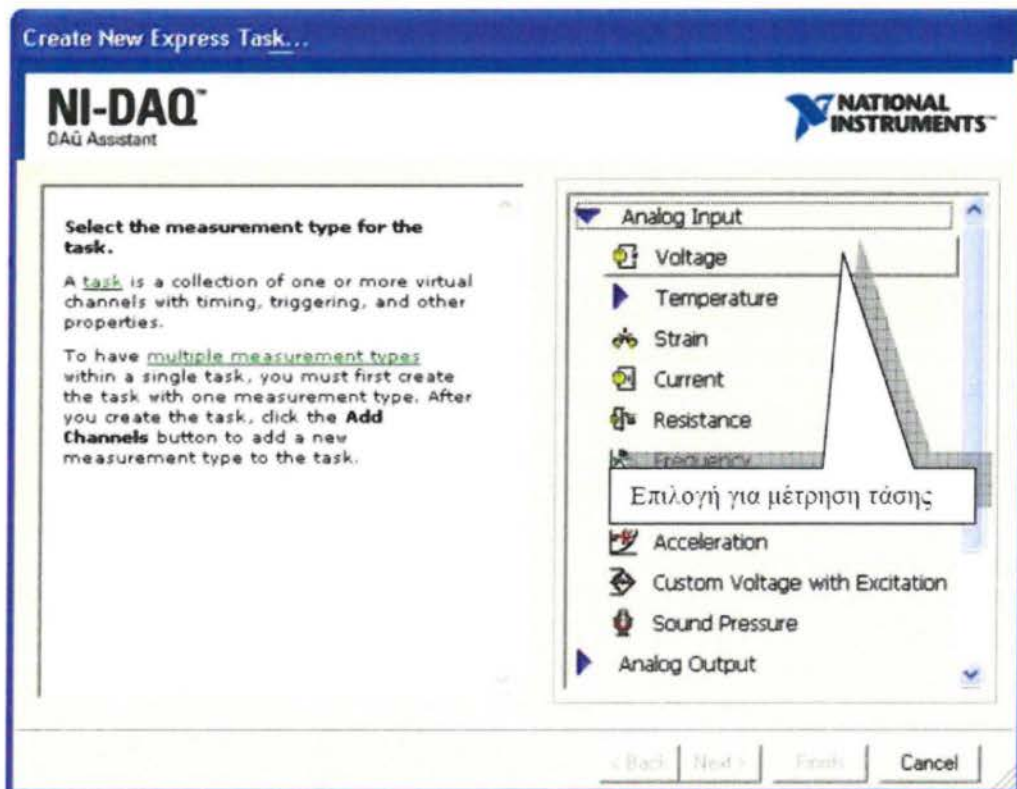


Στο προγραμματιστικό παράθυρο του DAQ Assistance που μας ανοίγει αυτομάτως με την τοποθέτηση του στο δια-γραμμικό μπλοκ ορίζουμε το είδος του σήματος καθώς και την είσοδο / έξοδο που θέλουμε να χρησιμοποιήσουμε από την μονάδα USB 6009. Ακολουθώντας τις παρακάτω εικόνες προγραμματίζουμε τη μονάδα σύμφωνα με τις απαιτήσεις της εφαρμογή μας.

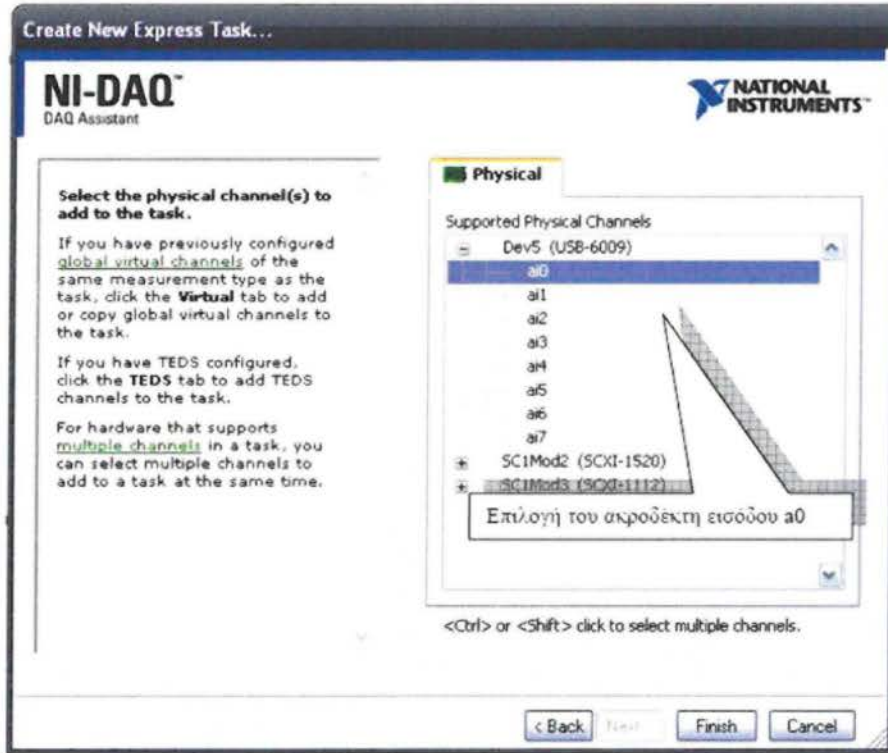
1. Επιλογή παραγωγής ή απόκτησης σήματος



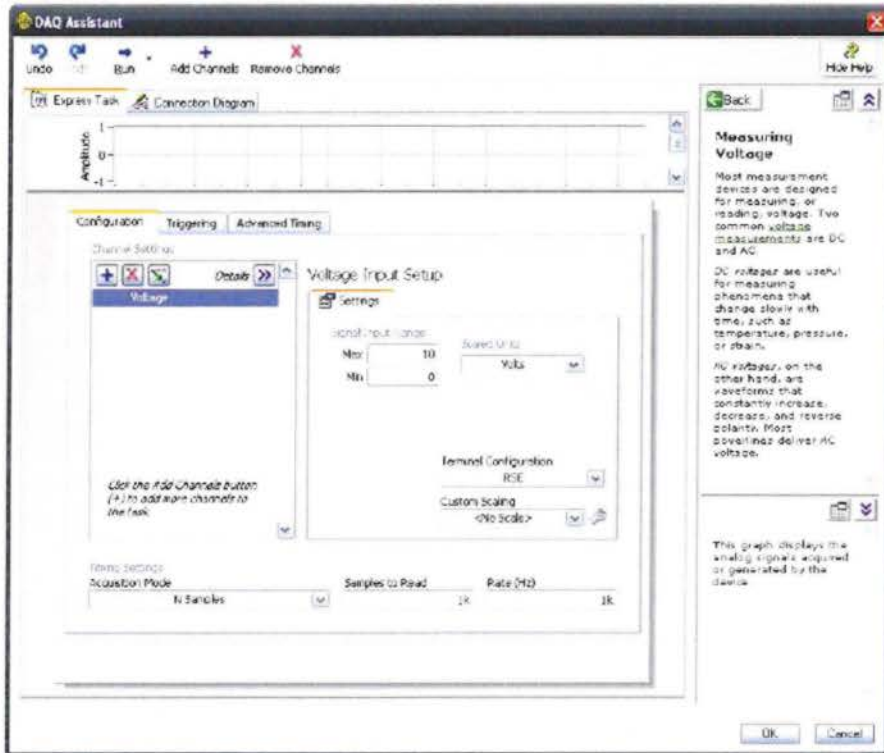
2. Επιλογή του τύπου μέτρησης



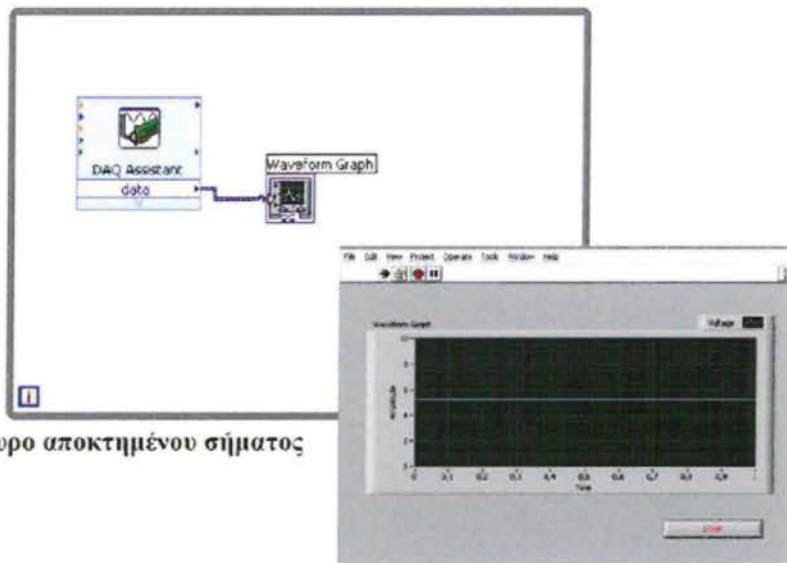
3. Επιλογή του ακροδέκτη σύνδεσης



4. Προγραμματισμός χαρακτηριστικών του ακροδέκτη εισόδου σύμφωνα με τις ανάγκες της εφαρμογής μας



5. Τέλος συνδέουμε στο δια-γραμμικό μπλοκ σύνδεσης τον DAQ Assistant όπως στην παρακάτω εικόνα ένα καταγραφικό τύπου WaveGraph για την απεικόνιση του σήματος που εφαρμόσαμε στον ακροδέκτη εισόδου της μονάδας.



Μιμικό παράθυρο αποκτημένου σήματος

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Πάριντας Θέμης , 2006 " ΤΕΙ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ, ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΙΑΣ Σημειώσεις εργαστήριο Scada : Summary,Tips and Tricks on introduction to LABVIEW "
2. Nasser Kehtarnavaz , Namzin Kim , "Digital Signal Processing System-Level Design Using Labview"
3. ΑΤΕΙ Πειραιά -Τμήμα ΗλεκτρονικώνΥπολογιστικών Συστημάτων - Εργαστήριο Ψ.Σ.Α.Ε , "ΘΕΩΡΗΤΙΚΟ ΜΕΡΟΣ LABVIEW"
4. National Instruments , "PID Control Toolset User Manual"
5. Sarah Fenney, Arisona state university "National Instrument:The LabVlew Programming Enviroment and Basic Operations"
6. Barry Paton ,1998 Dalhousie University "Fundamentals of digital Electronics"
7. National Instruments , "PID Cotrol Toolkit for G Reference Manual"
8. Eric Shaffer and Michael Kleinigger, RPI Undergraduate Engineering Students
9. Dr. Kevin Craig, Professor of Mechanical Engineering
10. RPI Mechatronics Laboratory "How to Create a System Simulation using LabVIEW 8.2"
11. Ricardo Dunia (NI), Eric Dean (NI), and Dr. Thomas Edgar (UT) "Introduction to LabVIEW for Control Design & Simulation" Reference Text : Process Dynamics and Control 2nd edition, by Seborg, Edgar, Mellichamp, Wiley 2004

Διαδικτυακή τόποι:

<https://www.youtube.com/user/nationalinstruments>

<http://www.ni.com/>

http://en.wikipedia.org/wiki/Main_Page

<http://www.mccdaq.com/>


<http://sae.teipir.gr/>


<http://www.labviewsolutions.com/>


www.elabview.gr/


ΠΑΡΑΡΤΗΜΑ


Tools Pallet


-  **Operating Tool.** Ο Δείκτης χρησιμεύει για την επιλογή θέσεων και τιμών για αντικείμενα του Front Panel.


-  **Positioning tool** (Εργαλείο Τοποθέτησης). Αυτό το εργαλείο μας βοηθά να επιλέξουμε ένα αντικείμενο, να το μετακινήσουμε, να το μεγενθύνουμε. Για την μεγένθυση ενός αντικειμένου πρέπει να το τοποθετήσουμε το Εργαλείο σε μία από τις γωνίες του αντικειμένου.


-  **Labeling tool** (Εργαλείο Ονομασίας). Με αυτό το εργαλείο μπορούμε να ονομάσουμε τα διάφορα αντικείμενα ή και να τοποθετήσουμε ετικέτες από μόνες τους τόσο στον χώρο του Front Panel όσο και στο μπλόκ Διάγραμμα.

-  **Wiring tool** (Εργαλείο Σύνδεσης). Με αυτό το εργαλείο συνδέουμε τους διάφορους κόμβους μεταξύ τους, στο Διάγραμμα. Τοποθετώντας το εργαλείο αυτό πάνω από οποιοδήποτε καλώδιο, βλέπουμε τον τύπο των δεδομένων που μεταφέρει το καλώδιο αυτό. Θα πρέπει πρώτα να επιλέξουμε **Show Help Window** από το μενού **Windows**.

-  **Object pop-up menu tool.** Με το εργαλείο αυτό ανοίγουμε το μενού pop-up ενός αντικειμένου πατώντας το αριστερό ποντίκι.

-  **Scrolling tool.** Κάνουμε εύκολα Scroll μεταξύ διαφόρων Παραθύρων.

-  **Breakpoint tool.** Με αυτό το εργαλείο θέτουμε Breakpoints στην λειτουργία VIs και ρουτινών.

 **Probe tool.** Τοποθετείται πάνω σε καλώδια, και χρησιμεύει για την απεικόνιση των δεδομένων που ρέουν σε αυτά κατά την λειτουργία ενός VI.



Color Copy tool. Χρησιμεύει για την αντιγραφή χρωμάτων.



Coloring tool. Χρησιμοποιείται για τον χρωματισμό ενός αντικειμένου.

Controls Pallet



Η Υποπαλέττα **Numeric.** Περιέχει controls και indicators για αριθμητικά Δεδομένα (ακέραιους, δεκαδικούς).



Η Υποπαλέττα **Boolean.** Περιέχει controls και indicators για λογικά / ψηφιακά δεδομένα.



Η Υποπαλέττα **String & Path.** Περιέχει controls και indicators για κείμενο μορφής ASCII ή Binary καθώς και Paths για την ονομασία αρχείων.



Η Υποπαλέττα **List & Table.** Περιέχει controls και indicators για δημιουργία μενού εναλλακτικών επιλογών.



Η Υποπαλέττα **Array & Cluster.** Περιέχει controls και indicators για ομαδοποίηση δεδομένων.



Η Υποπαλέττα **Graph.** Περιέχει indicators για γραφική απεικόνιση δεδομένων.



Η Υποπαλέττα **Refnum.** Περιέχει controls και indicators για επεξεργασία αρχείων.



Η Υποπαλέττα **Decorations.** Περιέχει γραφικά για εξωραισμό των Front Panels.



Η Υποπαλέττα **User Controls**. Εδώ τοποθετεί ο χρήστης τα δικά του Controls.



Η Υποπαλέττα **Select a Control**. Μας δίνει την δυνατότητα να επιλέξουμε και να φορτώσουμε άλλα Controls.

Functions Pallet



Υποπαλέττα **Structures**. Περιέχει προγραμματιστικές δομές όπως While & For Loops.



Υποπαλέττα **Numeric**. Περιέχει ρουτίνες αριθμητικές, λογαριθμικές, τριγωνομετρικές.



Υποπαλέττα **Boolean**. Περιέχει VIs για λογικές πράξεις.



Υποπαλέττα **String**. Περιέχει VIs για επεξεργασία κειμένου σε μορφή ASCII ή Binary.



Υποπαλέττα **Array**. Χρησιμοποιείται για την επεξεργασία πινάκων.



Υποπαλέττα **Cluster**. Περιέχει ρουτίνες για την επεξεργασία δομών από ανομοιογενή στοιχεία. Οι Δομές αυτές στο LabVIEW καλούνται Clusters.



Υποπαλέττα **Comparison**. Περιέχει ρουτίνες για την σύγκριση στοιχείων, τα οποία μπορεί να είναι αριθμητικά, λογικά, text.



Υποπαλέττα **Time & Dialog**. Χρησιμοποιείται για Διαλογικά Παράθυρα, Χρονισμό.



Υποπαλέττα **File I/O**. Πολύ χρήσιμη για καταχώρηση δεδομένων και επεξεργασία αρχείων.



Υποπαλέττα **Communication**. Περιέχει VIs για επικοινωνία με πρωτόκολα όπως TCP, OLE, DDE.



Υποπαλέττα **Instrument I/O**. Χρησιμοποιείται για επικοινωνία με αυτόνομα όργανα και υποστηρίζει πρωτόκολα όπως GPIB, RS-232, VISA.



Υποπαλέττα **Data Acquisition** κάτω από το **Measurement I/O**. Περιέχει VIs για έλεγχο καρτών συλλογής μετρήσεων.



Υποπαλέττα **Analysis**. Περιέχει 600 διαφορετικούς VIs ανάλυσης δεδομένων.



Υποπαλέττα **Tutorial**. Περιέχει διάφορα VIs που χρησιμοποιούνται στο LabVIEW tutorial.



Υποπαλέττα **Advanced**. Περιέχει διάφορες ρουτίνες για προχωρημένους χρήστες.



Υποπαλέττα **Select a VI...**. Εμφανίζει ένα διαλογικό παράθυρο για την εύρεση και επιλογή υπό VIs.



Υποπαλέττα **Users Library**. Εδώ μπορεί ο χρήστης να τοποθετήσει τα δικά του VIs.











Υποπαλέττα **Instrument Drivers**. Εδώ τοποθετούνται συνήθως Driver VIs για επικοινωνία με όργανα.

Τα Εργαλεία του Front Panel



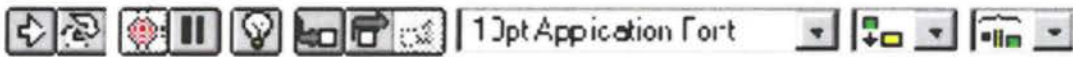
Το κουμπί Run. Πατήστε το για να τρέξετε το VI. Όταν ο VI τρέχει το κουμπί αλλάζει στα παρακάτω

-  εάν ο VI είναι ο κύριος VI (top-level)
-  εάν πρόκειται για έναν υπόVI ο οποίος καλείται από κάποιο VI σε ανώτερο επίπεδο
-  Όταν ο VI τρέχει, εμφανίζεται το κουμπί Stop Πατώντας το, σταματάτε το τρέξιμο του προγράμματος αμέσως.
-  Το κουμπί σπασμένης λειτουργίας (Broken-Run). Αντικαθιστά το κουμπί Run και υποδηλώνει ότι ο VI δεν μπορεί να τρέξει λόγω σφαλμάτων στο διάγραμμα. Πατώντας το κουμπί, βλέπουμε ποιά ακριβώς είναι τα σφάλματα.
-  Το κουμπί Συνεχούς Λειτουργίας (Continuous Run). Πατώντας το, ο VI εκτελείται συνεχώς.
-  Το κουμπί Προσωρινής Διακοπής (Pause). Πατώντας το ξανά, επαναφέρουμε τον VI στην κανονική του λειτουργία.
-  Η επιλογή Ευθυγράμμισης. Μπορούμε εδώ, να επιλέξουμε την μέθοδο ισοστοίχισης αντικειμένων: Κάθετα, Οριζόντια, Αριστερή Ακρη κ.ο.κ.
-  Επιλογή Ισοστοίχισης Αντικειμένων.

 Επι Application Font

Η επιλογή φόντου. Εδώ, διαλέγουμε τον τύπο, μέγεθος, στύλ, χρώμα.

Τα εργαλεία του μπλόκ Διαγράμματος



Το κουμπί Αργής Κίνησης (Execution Highlighting). Πατώντας το, παρακολουθούμε την ροή των δεδομένων μεταξύ των κόμβων, σε αργή κίνηση, και μπορούμε εύκολα να διαγνώσουμε λάθη στον προγραμματισμό του διαγράμματος.



Το κουμπί - Single Step/Step Over. Πατώντας το, τρέχουμε το πρόγραμμα από κόμβο σε κόμβο. Φτάνοντας σε έναν κόμβο, μπορούμε να αποφύγουμε να τον τρέξουμε, γιά να γλυτώσουμε χρόνο με το κουμπί Step Over, όπως γιά παράδειγμα σε ένα Loop 1000 κύκλων.



Το κουμπί Step Into, αντίθετα με το Step Over μας επιτρέπει να εισχωρήσουμε στον κόμβο και να παρακολουθήσουμε την λειτουργία του.



Κουμπί Step Out μας επιτρέπει εχοντας εισχωρήσει μέσα στον κόμβο με το Step Into να εξέλθουμε.



Η Ενδειξη Προειδοποίησης (Warning Indicator). Παρουσιάζεται όταν υπάρχει πιθανό πρόβλημα στο διάγραμμα, αλλά παρόλα αυτά ο VI μπορεί να τρέξει.

Icon Editor



pencil Σχεδιάζει με ακρίβεια pixel ανά pixel.



line Σχεδιάζει σε Ευθείες Γραμμές. Πατώντας <shift> περιορίζετε την κίνηση οριζόντια, ή κάθετα.



dropper Επιλέγει το χρώμα στο προσκήνιο.



fill bucket Γεμίζει την επιλεγμένη επιφάνεια με το χρώμα επιλογής για το προσκήνιο.



Rectangle Σχεδιάζει ένα παραλληλόγραμμο περίγραμμα στο επιλεγμένο χρώμα για το προσκήνιο.



filled Σχεδιάζει ένα παραλληλόγραμμο το περίγραμμα του οποίου έχει το χρώμα το προσκήνιου και το εσωτερικό το χρώμα του παρασκήνιου.



select Επιλέγει μία περιοχή της εικόνας για οποιαδήποτε αλλαγή ή μετακίνηση.



text Εισάγει κείμενο στην εικόνα.



Foreground.. Επιλογή χρώματος για το προσκήνιο, παρασκήνιο.