



ΑΝΩΤΑΤΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΠΕΙΡΑΙΑ
ΤΕΧΝΟΛΟΓΙΚΟΥ ΤΟΜΕΑ

**ΑΕΙ ΠΕΙΡΑΙΑ Τ.Τ.
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ Τ.Ε.**

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

LabView and Vi Server Connectibility



**Μισύρης Χριστόδουλος
Στρατής Αλβέρτος**

Εισηγητής: Αναστασία Ν. Βελώνη, Καθηγήτρια

**ΑΘΗΝΑ
ΝΟΕΜΒΡΙΟΣ 2016**

(Κενό φύλλο)

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

LabView and Vi Server Connectibility

**Μισύρης Χριστόδουλος 40736
Στρατής Αλβέρτος 40204**

Εισηγητής:

Αναστασία Ν. Βελώνη, Καθηγήτρια

Εξεταστική Επιτροπή:

Έλληνας Ν. Ιωάννης, Δροσινόπουλος Παναγιώτης

Ημερομηνία εξέτασης: 2016

(Κενό φύλλο)

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Οι κάτωθι υπογεγραμμένοι Μισύρης Χριστόδουλος και Στρατής Αλβέρτος , του Βασίλειου και του Αιμίλιου αντίστοιχα, με αριθμό μητρώου 40736 και 40204, φοιτητές του Τμήματος Μηχανικών Η/Υ Συστημάτων Τ.Ε. του Α.Ε.Ι. Πειραιά Τ.Τ. πριν αναλάβουν την εκπόνηση της πτυχιακής εργασίας τους, δηλώνουν ότι ενημερώθηκαν για τα παρακάτω:

«Η Πτυχιακή Εργασία (Π.Ε.) αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο του συγγραφέα, όσο και του Ιδρύματος και θα πρέπει να έχει μοναδικό χαρακτήρα και πρωτότυπο περιεχόμενο.

Απαγορεύεται αυστηρά οποιοδήποτε κομμάτι κειμένου της να εμφανίζεται αυτούσιο ή μεταφρασμένο από κάποια άλλη δημοσιευμένη πηγή. Κάθε τέτοια πράξη αποτελεί προϊόν λογοκλοπής και εγείρει θέμα ηθικής τάξης για τα πνευματικά δικαιώματα του άλλου συγγραφέα. Αποκλειστικός υπεύθυνος είναι ο συγγραφέας της Π.Ε., ο οποίος φέρει και την ευθύνη των συνεπειών, ποινικών και άλλων, αυτής της πράξης.

Πέραν των όποιων ποινικών ευθυνών του συγγραφέα σε περίπτωση που το Ίδρυμα του έχει απονείμει Πτυχίο, αυτό ανακαλείται με απόφαση της Συνέλευσης του Τμήματος. Η Συνέλευση του Τμήματος με νέα απόφασής της, μετά από αίτηση του ενδιαφερόμενου, του αναθέτει εκ νέου την εκπόνηση της Π.Ε. με άλλο θέμα και διαφορετικό επιβλέποντα καθηγητή. Η εκπόνηση της εν λόγω Π.Ε. πρέπει να ολοκληρωθεί εντός τουλάχιστον ενός ημερολογιακού εξαμήνου από την ημερομηνία ανάθεσής της. Κατά τα λοιπά εφαρμόζονται τα προβλεπόμενα στο άρθρο 18, παρ. 5 του ισχύοντος Εσωτερικού Κανονισμού.»

(Κενό φύλλο)

ΕΥΧΑΡΙΣΤΙΕΣ

Με την ολοκλήρωση της παρούσας πτυχιακής εργασίας σηματοδοτείται το τέλος του πιο εποικοδομητικού μέχρι τώρα κύκλου της ζωής μας και το ξεκίνημα ενός καινούργιου. Ένας κύκλος που μας έδωσε τα εφόδια για να μπορέσουμε να συνεχίσουμε και να κατακτήσουμε τα όνειρά μας.

Ευχαριστίες αποστέλλονται στους γονείς μας, Βασίλειο, Φωτεινή Μισύρη και Αιμίλιο, Χρυσάνθη Στρατή για τον καθημερινό αγώνα που δίνουν, με σκοπό την δική μας πρόοδο. Ευχαριστούμε τις αδελφές μας Ιωάννα Μισύρη και Μέλανη Στρατή, όπως και την υπόλοιπη οικογένειά μας για την στήριξή τους.

Θέλουμε να ευχαριστήσουμε, ακόμη, το Διαδικτυακό Κέντρο της σχολής για την παραχώρηση στατικής IP, αλλά και την παραχώρηση χώρου για την φιλοξενία του WebServer.

Επίσης, θα θέλαμε να ευχαριστούμε ιδιαιτέρως, την Καθηγήτρια κα Βελώνη Αναστασία που μας έδειξε εμπιστοσύνη και μας έδωσε την ευκαιρία να ασχοληθούμε με αυτό το πρακτικό και ενδιαφέρον θέμα, αλλά και για την βοήθεια, την πολύτιμη καθοδήγηση και τις υποδείξεις που μας πρόσφερε κατά την εκπόνηση της πτυχιακής εργασίας.

Τέλος, ευχόμαστε στην κα Βελώνη Αναστασία καλή συνέχεια τόσο στην ζωή της, όσο και στην καριέρα της.

(Κενό φύλλο)

ΠΕΡΙΛΗΨΗ

Η παρούσα πτυχιακή εργασία ασχολείται με τις διασυνδέσεις που μπορούν να επιτευχθούν με την χρήση του LabView. Γίνεται ανάλυση των τεχνολογιών που δίνουν την δυνατότητα αποστολής και λήψης δεδομένων, αλλά και ελέγχου του LabView, όπως VI Server, Web Server (HTTP) και Remote Panels όπως επίσης και επικοινωνία μίας συσκευής PLC με αυτό.

ABSTRACT

The present thesis deals with interfaces that can be achieved by using LabView. Analyzes the technologies that enable sending, receiving data and controlling LabView such as VI Server, Web Server (HTTP) and Remote Panels as well as the communication with a PLC device.

ΕΠΙΣΤΗΜΟΝΙΚΗ ΠΕΡΙΟΧΗ: Σήματα και Συστήματα

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Συνδεσιμότητα, Connectibility, WebServer, Vi Server, LabView

ΠΕΡΙΕΧΟΜΕΝΑ

1. Εισαγωγή στο LabView	15
1.1. Τι μπορεί να κάνει το LabView	18
1.2. Πως λειτουργεί το LabView	18
1.3. Τα τρία διαφορετικά τμήματα ενός VI	19
1.3.1. Το εμπρόσθιο πλαίσιο (Front panel)	19
1.3.1.1. Παλέτες	21
1.3.1.2 Η Παλέτα των Εργαλείων (Tools)	21
1.3.2. Η Παλέτα των Αντικειμένων (Controls) και Ρουτινών (Functions)	23
1.3.2.1. Η Παλέτα των Αντικειμένων (Controls)	23
1.3.2.2. Η Παλέτα των Ρουτινών (Functions)	25
1.3.3. Τα εργαλεία του μπλοκ Διαγράμματος (Block Diagram)	26
1.3.3.1. Wires (Καλώδια)	28
1.3.3.2. Δομές	28
1.3.3.3. Arrays and Clusters	29
1.3.3.4. Μενού Pop-Up	30
1.3.3.5. Μενού Pull-Down	30
1.3.4. Ο Κοννέκτορας και η Εικόνα (Icon & Connector)	32
1.3.4.1. Connector Panel	32
1.4. Βιβλιοθήκες από Vis (VI Libraries)	33
1.5. Φορτώνοντας VIs	34
1.6. Μεταφορά VIs σε άλλες πλατφόρμες	35
2. Ο εξυπηρετητής LabView (VI Server)	37
2.1. Ενεργοποίηση απομακρυσμένης πρόσβασης στο VI Server	39
2.1.1. Property Node	42
2.1.2. Invoke Node	43
2.2. Application References (Αναφορές Εφαρμογής)	45
2.3. Από την θεωρία στην πράξη (Παράδειγμα)	45
2.4. Παράδειγμα VI Server	48
3. Δημιουργία μιας υπηρεσίας Web με το LabView	55
4. Σύνδεση HTTP Request με το LabView	59

4.1. Μεταφορά δεδομένων μεταξύ των HTTP VIs και του WebClient	59
5. Remote Panel Server	65
5.1. Μέθοδος Embedded	66
5.2. Μέθοδος Snapshot	69
5.3. Μέθοδος Monitor.....	69
5.4. Connect to Remote Panel.....	69
5.5. Remote Panel Connection Manager	71
6. Σύνδεση PLC με LabView	73
6.1. Μέρος 1 ^ο (Δημιουργία Εικονικού PLC και προετοιμασία αυτού για σύνδεση με OPC Servers).....	74
6.1.1. Λίγα λόγια για το Simatic	74
6.2. Μέρος 2 ^ο (Σύνδεση OPC Server με PLC).....	83
6.3. Μέρος 3 ^ο (Σύνδεση OPC Server με το LabView).....	87
ΠΑΡΑΡΤΗΜΑ	97
1. Απομακρυσμένος έλεγχος ενός VI	97
2. Εργαστηριακή Άσκηση.....	99
ΒΙΒΛΙΟΓΡΑΦΙΑ	101

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1.1: Περιβάλλον Χρήσης.	16
Εικόνα 1.2: Γραφικός Κώδικας.	17
Εικόνα 1.3: Front Panel.....	19
Εικόνα 1.4: Εμπρόσθιο πλαίσιο (Front panel) του LabView. Αριστερά φαίνεται η παλέτα εργαλείων (Tools) και η παλέτα ελέγχου (Controls).	23
Εικόνα 1.5: Η παλέτα των Controls.	24
Εικόνα 1.6: Η Παλέτα των Ρουτινών (Functions).	25
Εικόνα 1.7 Δομικό διάγραμμα (Block diagram).....	27
Εικόνα 1.8: Τα καλώδια (wires).....	28
Εικόνα 1.9: Το For Loop.....	28
Εικόνα 1.10: Το While Loop.	29
Εικόνα 1.11: Το Case Structure.	29
Εικόνα 1.12: Εικονίδιο.	32
Εικόνα 1.13: Connector.....	32
Εικόνα 1.14: Το Connector Panel.	32
Εικόνα 1.15: Patterns στο Connector Panel.....	33
Εικόνα 2.1: Ο εξυπηρετητής LabView (Vi Server).....	37
Εικόνα 2.2: Εκτέλεση του testing.vi που βρίσκεται στον τοπικό δίσκο	38
Εικόνα 2.3: Εκτέλεση του testing.vi που βρίσκεται στην απομακρυσμένη διεύθυνση viserver.psae.lab.....	39
Εικόνα 2.4: Δομή VI Server	39
Εικόνα 2.5: Επιλογές διαμόρφωσης VI Server.....	40
Εικόνα 2.6: Η παλέτα Application Controls	41
Εικόνα 2.7: Property Node και Invoke Node	42
Εικόνα 2.8: Property Node	42
Εικόνα 2.9: Invoke Node	43
Εικόνα 2.10: Κλάση αντικειμένου	44
Εικόνα 2.11: VI Server Reference.....	46
Εικόνα 2.12: Reference ασύνδετη.....	47
Εικόνα 2.13: Open Application Reference	47
Εικόνα 2.14: Close Reference.....	48

Εικόνα 2.15: Παράδειγμα VI Server	48
Εικόνα 2.16: Open App Reference.....	49
Εικόνα 2.17: Path, IP, Port Constant.....	49
Εικόνα 2.18: Invoke Node (FP.Open Method).....	49
Εικόνα 2.19: Invoke Node (Ctrl.Set Method)	50
Εικόνα 2.20: Invoke Node (Run VI).....	51
Εικόνα 2.21: Close Reference.....	51
Εικόνα 2.22: Server's Front Panel.....	52
Εικόνα 2.23: Invoke Node (Ctrl.Val.Get Method)	52
Εικόνα 2.24 Αποτέλεσμα Ctrl.Val.Get	53
Εικόνα 3.1: Δημιουργία Web Service	55
Εικόνα 3.2: Παραμετροποίηση Web Server	56
Εικόνα 3.3: Visible Vis, Browser Access	56
Εικόνα 3.4: Δημιουργία Web Resource VI	57
Εικόνα 3.5: Επιλογή μεθόδων HTTP	57
Εικόνα 4.1: Προβολή URL.....	59
Εικόνα 4.2: Front Panel, Connector Panel	60
Εικόνα 4.3: JSON δεδομένα.....	60
Εικόνα 4.4: Επιλογή μορφής δεδομένων	61
Εικόνα 4.5: XML δεδομένα.....	61
Εικόνα 4.6: Προσθήκη Δημόσιου Φακέλου	62
Εικόνα 4.7: Web Application	62
Εικόνα 4.8: Προβολή URL.....	63
Εικόνα 5.1: Remote Panel Server	65
Εικόνα 5.2: Παραμετροποίηση Remote Panel Server	65
Εικόνα 5.3: Web Publishing Tool	66
Εικόνα 5.4: Ρυθμίσεις Embedded	67
Εικόνα 5.5: Επιλογές Path, Filename, URL για το αρχείο	68
Εικόνα 5.6: Web Embedded.....	68
Εικόνα 5.7: Απόκτηση ελέγχου VI	69
Εικόνα 5.8: Επιλογή ρυθμού ανανέωσης	69
Εικόνα 5.9: Σύνδεση σε Remote Panel	70
Εικόνα 5.10: Παράθυρο συμπλήρωσης στοιχείων Remote Server	70

Εικόνα 5.11: Front Panels (Client, Server).....	70
Εικόνα 5.12: Remote Panel Connection Manager	71
Εικόνα 6.1: Σύνδεση PLC με LabVIEW.....	73
Εικόνα 6.2: Διαχωρισμός διασύνδεσης σε δύο μέρη.....	73
Εικόνα 6.3: Simatic Step 7	74
Εικόνα 6.4: Simatic Manager	75
Εικόνα 6.5: Εισαγωγή Simatic Station.....	76
Εικόνα 6.6: Εισαγωγή Hardware	76
Εικόνα 6.7: Εισαγωγή Rack-300	77
Εικόνα 6.8: Ethernet Interface.....	77
Εικόνα 6.9: Rack μετά την εισαγωγή CPU	78
Εικόνα 6.10: Configure Network.....	78
Εικόνα 6.11: Επιλογή Γλώσσας	79
Εικόνα 6.12: Παράθυρο σχεδίασης γλώσσας Ladder	80
Εικόνα 6.13: Πρόγραμμα Ladder	80
Εικόνα 6.14: Save και Download	81
Εικόνα 6.15: Simulation ON	81
Εικόνα 6.16: Monitor ON	81
Εικόνα 6.17: Διάγραμμα Ladder και Simulation	82
Εικόνα 6.18: Ρυθμίσεις NetToPLCSim.....	83
Εικόνα 6.19: Επιλογή Device Driver.....	83
Εικόνα 6.20: Επιλογή Δικτύου.....	84
Εικόνα 6.21: Επιλογή συσκευής.....	84
Εικόνα 6.22: Εισαγωγή εισόδων εξόδων	85
Εικόνα 6.23: Quick Client.....	85
Εικόνα 6.24: Είσοδοι και έξοδοι του PLC	86
Εικόνα 6.25: Simatic και OPC Server.....	86
Εικόνα 6.26: Δημιουργία νέου Project.....	87
Εικόνα 6.27: Δημιουργία I/O Server	87
Εικόνα 6.28: Επιλογή OPC Client	88
Εικόνα 6.29: Παραμετροποίηση OPC Client	88
Εικόνα 6.30: Δημιουργία μεταβλητών	89
Εικόνα 6.31: Επιλογή μεταβλητών	89

Εικόνα 6.32: Δημιουργία Front Panel.....	90
Εικόνα 6.33: Δημιουργία Block Diagram	90
Εικόνα 6.34: Επιλογή Shared Variable	91
Εικόνα 6.35: Επιλογή Access Mode.....	91
Εικόνα 6.36: Εισαγωγή While Loop	94
Εικόνα 6.37: Συγκεντρωτικά αποτελέσματα.	95
Εικόνα Π1.1: Σύνδεση με FTP Client στον Server	97
Εικόνα Π1.2: Φόρτωση του Vi από τον Server.....	98
Εικόνα Π1.3: Εργαστηριακές Ασκήσεις	99

1. Εισαγωγή στο LabView

Το LabView είναι μια 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 «αντικειμενοστραφούς 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, δείκτες, 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.



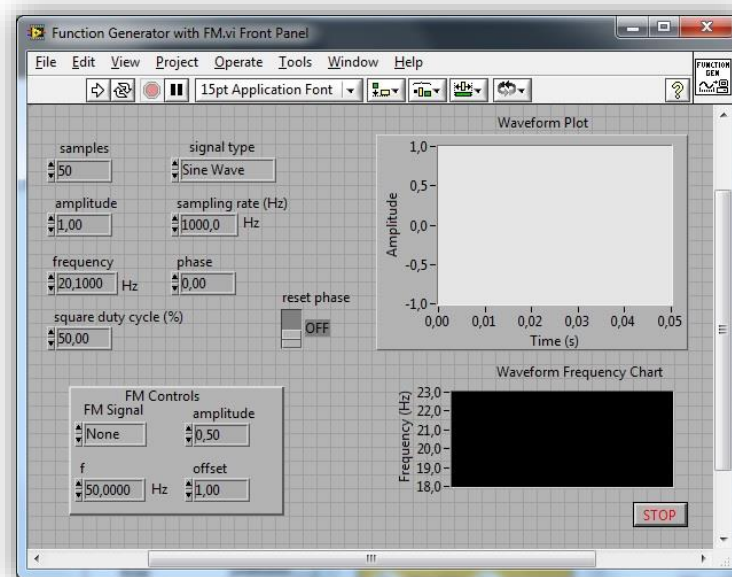
Το όνομα LabView είναι το ακρωνύμιο των λέξεων «Laboratory Virtual Instrument Engineering Workbench» (Σχεδιαστήριο για την Κατασκευή Εργαστηριακών Εικονικών Οργάνων) και αναπτύχθηκε κατά το τέλος της δεκαετίας του '80 από την εταιρία National Instruments (βλέπε www.ni.com). Η εταιρία αυτή ειδικεύεται σε συστήματα συλλογής δεδομένων, σε αισθητήρες, αυτοματισμούς και λογισμικό μετρήσεων και ελέγχου.

Προγραμματίζοντας με τα αντικείμενα που δίνει το περιβάλλον του LabView δημιουργούνται τα λεγόμενα «εικονικά όργανα» (Virtual Instruments ή απλώς VIs). Η γραφική γλώσσα που χρησιμοποιεί το LabView για τον προγραμματισμό και τη

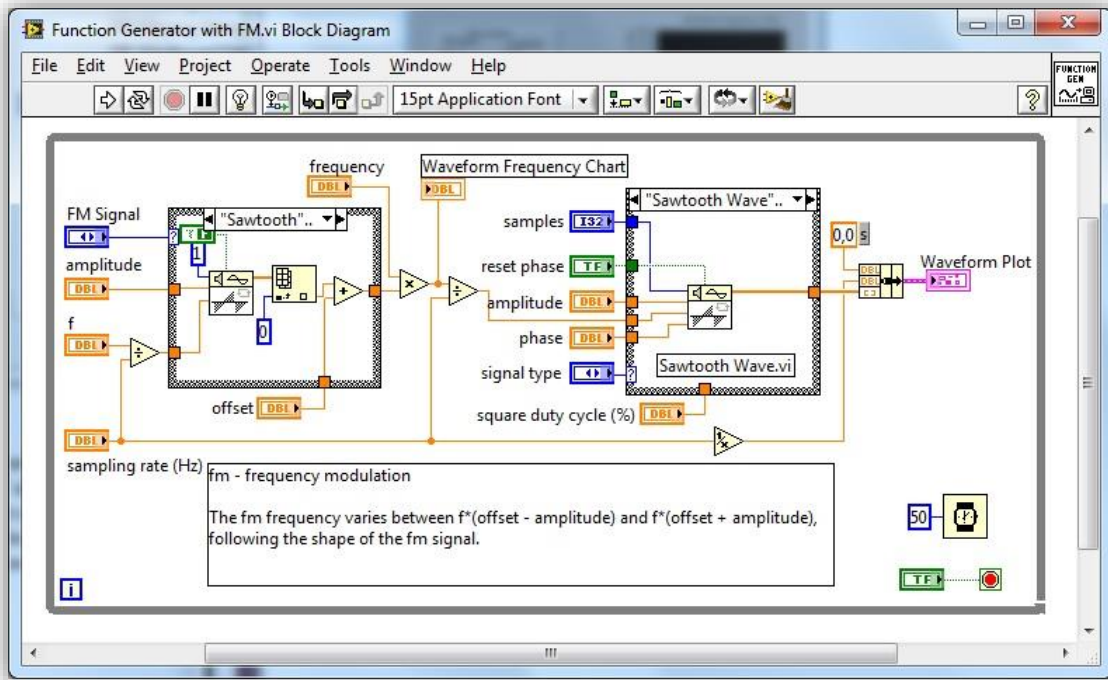
δημιουργία εικονικών οργάνων ονομάζεται γλώσσα G. Είναι πάντως δυνατό να χρησιμοποιήσει κανείς το LabView σε απλό επίπεδο, χρησιμοποιώντας έτοιμα εικονικά όργανα, χωρίς να μπαίνει σε λεπτομέρειες με τη γλώσσα G.

Ένα εικονικό όργανο μπορεί να προσομοιώνει απλώς μια λειτουργία και να την παρουσιάζει στην οθόνη του υπολογιστή, για εκπαιδευτικούς σκοπούς. Όμως, είναι δυνατό να συνδέεται με τις θύρες εισόδου/εξόδου του υπολογιστή ή με επιπρόσθετες κάρτες επέκτασης, προκειμένου να κάνει πραγματική εισαγωγή ή εξαγωγή δεδομένων. Στην περίπτωση αυτή ο υπολογιστής με τη βοήθεια των εισόδων και των εξόδων μετατρέπεται σε ένα ισχυρό εργαλείο μετρήσεων, με πολλές δυνατότητες επεξεργασίας δεδομένων.

Το LabView διαθέτει έναν αριθμό από έτοιμα VIs και ορισμένα εικονίδια συναρτήσεων που επιτρέπουν την επικοινωνία με όλα τα γνωστά πρωτόκολλα μετάδοσης δεδομένων. Έτσι, υπάρχουν έτοιμες λειτουργίες που επιτρέπουν τη συλλογή και μετάδοση δεδομένων μέσω της σειριακής θύρας του υπολογιστή, καθώς και μέσω της παράλληλης θύρας.



Εικόνα 1.1: Περιβάλλον Χρήσης.



Εικόνα 1.2: Γραφικός Κώδικας.

Επίσης, υπάρχουν λειτουργίες για την ανταλλαγή δεδομένων με την κάρτα ήχου, καθώς και με κάρτες επέκτασης που χρησιμοποιούν το πρωτόκολλο GPIB ή IEEE 488. Εξάλλου, όλες οι κάρτες συλλογής δεδομένων της εταιρίας National Instruments είναι συμβατές με το LabView με τη βοήθεια ειδικών οδηγών που ενσωματώνονται στο λογισμικό. Το ίδιο συμβαίνει και με σημαντικό αριθμό άλλων οργάνων, για τα οποία κυκλοφορούν οδηγοί συμβατοί με το LabView.

Έτσι, μέσω των πρωτοκόλλων επικοινωνίας (RS232, Centronics, IEEE488 ή TCP/IP) η πληροφορία που δημιουργείται στην οθόνη του υπολογιστή συνδέεται με πραγματικά όργανα, μέσω του λογισμικού. Όταν πατούμε ένα εικονικό κουμπί στην οθόνη, ενεργοποιείται ένας πραγματικός διακόπτης σε ένα εργαστηριακό όργανο. Σ' αυτήν ακριβώς τη δυνατότητα, που επεκτείνει την απλή προσομοίωση ώστε να γίνεται εφικτός ο έλεγχος αληθινών οργάνων, βρίσκεται και η δύναμη του LabView ως λογισμικό μετρήσεων και ελέγχου.

1.1. Τι μπορεί να κάνει το LabView

Το LabView απλοποιεί, στο μέτρο του δυνατού, την δημιουργία διάφορων διατάξεων. Περιέχει εκτεταμένο αριθμό βιβλιοθηκών, λειτουργιών και υπορουτίνων, ώστε να βοηθήσει σε όλες τις προγραμματιστικές εργασίες ενός μηχανικού.

Επίσης, περιέχει βιβλιοθήκες για συγκεκριμένες εφαρμογές, όπως εξόρυξη δεδομένων, σειριακά όργανα ελέγχου και GPIB, ανάλυση δεδομένων, παρουσίαση και αποθήκευση δεδομένων. Υπάρχουν λειτουργίες δημιουργίας σημάτων, επεξεργασίας σημάτων, φίλτρων, παραθύρων, στατιστικών, γραμμικής άλγεβρας, πινάκων, και παρεμβολών.

Το LabView περιλαμβάνει όλα τα συνηθισμένα εργαλεία ανάπτυξης με τα οποία μπορούμε να εισάγουμε σημεία διακοπών (break points), ροής ενός βήματος (single-step through), και γραφικής απεικόνισης της ροής εκτέλεσης ενός προγράμματος, ώστε να είναι εύκολη η παρακολούθηση της ροής και της επεξεργασίας των δεδομένων.

Λόγω της γραφικής φύσης του, το LabView είναι κυρίως ένα πακέτο παρουσίασης δεδομένων. Η έξοδος των δεδομένων μπορεί να πάρει οποιαδήποτε μορφή. Γραφήματα, πίνακες και οποιαδήποτε άλλης μορφής γραφική απεικόνιση που έχει δημιουργηθεί από τον χρήστη είναι μερικές από τις μορφές που μπορεί να πάρει η έξοδος.

Με τα εργαλεία εξόρυξης, ανάλυσης και παρουσίασης, το LabView είναι λειτουργικά ολοκληρωμένο. Οποιοδήποτε υπολογιστικό πρόβλημα είναι πραγματοποιήσιμο σε μια συμβατική γλώσσα προγραμματισμού, είναι γενικά ευκολότερο να υλοποιηθεί με την προσέγγιση των εικονικών οργάνων του LabView.

1.2. Πως λειτουργεί το LabView

Τα προγράμματα του LabView ονομάζονται εικονικά όργανα (virtual instruments ή VIs για συντομία) γιατί η εμφάνισή τους και η λειτουργία τους μιμείται τα πραγματικά όργανα. Παρόλα αυτά, στο παρασκήνιο είναι ανάλογα με τα κυρίως προγράμματα, τις λειτουργίες και τις υπορουτίνες από τις δημοφιλείς γλώσσες προγραμματισμού, όπως η C και η Java. Τα VIs έχουν και το διαδραστικό περιβάλλον χρήστη αλλά και τον ισοδύναμο πηγαίο κώδικά τους, και μπορούμε να περνάμε δεδομένα ανάμεσα στα δύο.

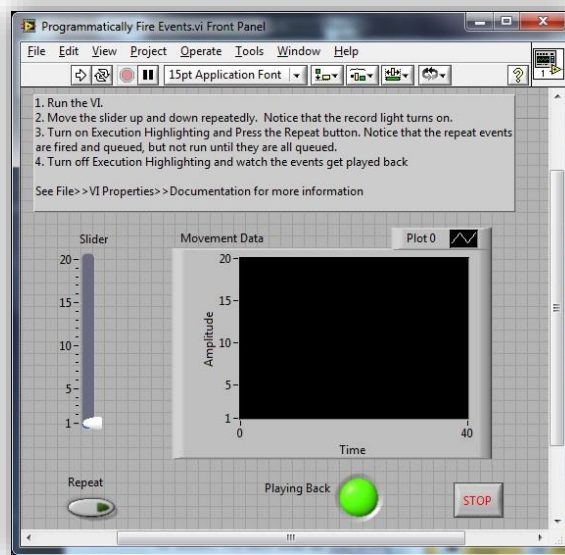
1.3. Τα τρία διαφορετικά τμήματα ενός VI

Κάθε VI αποτελείται από τρία διαφορετικά τμήματα. Το εμπρόσθιο πλαίσιο ή αλλιώς Front Panel, το παράθυρο του διαγράμματος ή Block Diagram, και το παράθυρο του Κοννέκτορα και της Εικόνας, γνωστό ως Icon/Connector.

1.3.1. Το εμπρόσθιο πλαίσιο (Front panel)

Μοιάζει με το μπροστινό μέρος ενός οργάνου. Μπορεί να περιέχει κουμπιά, διακόπτες, οθόνες γραφικών κ.ά. Τα στοιχεία του εμπρόσθιου πλαισίου παίρνουν τιμές με τη βοήθεια του ποντικιού ή του πληκτρολογίου. Για παράδειγμα, με το ποντίκι μπορούμε να πατήσουμε έναν διακόπτη και με το πληκτρολόγιο να ορίσουμε τιμή για μια τάση ή για μια θερμοκρασία. Το εμπρόσθιο πλαίσιο είναι αυτό που κυρίως χειρίζεται ο χρήστης. Το σχήμα 1.3 είναι παραδείγματα εμπρόσθιου πλαισίου εικονικών οργάνων. Στο σχ. 1.3 τοποθετείται στο εμπρόσθιο πλαίσιο ένας περιστροφικός επιλογέας και μια κάθετη μπάρα απεικόνισης.

Τα Εργαλεία του εμπρόσθιου πλαισίου (Front Panel)



Εικόνα 1.3: Front Panel.




Το κουμπί Run. Πατώντας το για να τρέξετε το VI, αλλάζει σε:



εάν ο VI είναι ο κύριος VI(top-level) ή



εάν πρόκειται για έναν υπόVI, ο οποίος καλείται από κάποιο VI σε ανώτερο επίπεδο.

Όταν το VI τρέχει, εμφανίζεται το κουμπί  Stop. Πατώντας το, σταματάει το τρέξιμο του προγράμματος αμέσως.



Το κουμπί σπασμένης λειτουργίας (Broken-Run). Αντικαθιστά το κουμπί Run και υποδηλώνει ότι το VI δεν μπορεί να τρέξει λόγω σφαλμάτων στο διάγραμμα. Πατώντας το κουμπί, παρατηρείται ποια 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 φόντου. Εδώ, γίνεται επιλογή του τύπου, του μεγέθους, του στυλ και του χρώματος.



Η επιλογή Ευθυγράμμισης. Εδώ υπάρχει η δυνατότητα επιλογής της μεθόδου ισοστοίχισης αντικειμένων: Κάθετη, Οριζόντια, Αριστερή Άκρη

κ.ο.κ.



Επιλογή Ισοστοίχισης Αντικειμένων.

Τοποθετώντας το εργαλείο πάνω από οποιοδήποτε καλώδιο βλέπουμε τον τύπο των δεδομένων που μεταφέρει το καλώδιο αυτό. Θα πρέπει πρώτα να γίνει επιλογή του **Show Help Window** από το μενού **Windows**.



Object pop-up menu tool. Με το εργαλείο αυτό εμφανίζεται το μενού pop-up ενός αντικειμένου πατώντας το αριστερό κλικ.



Scrolling tool. Γίνεται εύκολα Scroll μεταξύ διαφόρων Παραθύρων.



Breakpoint tool. Θέτονται Breakpoints στην λειτουργία VIs και ρουτινών.



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

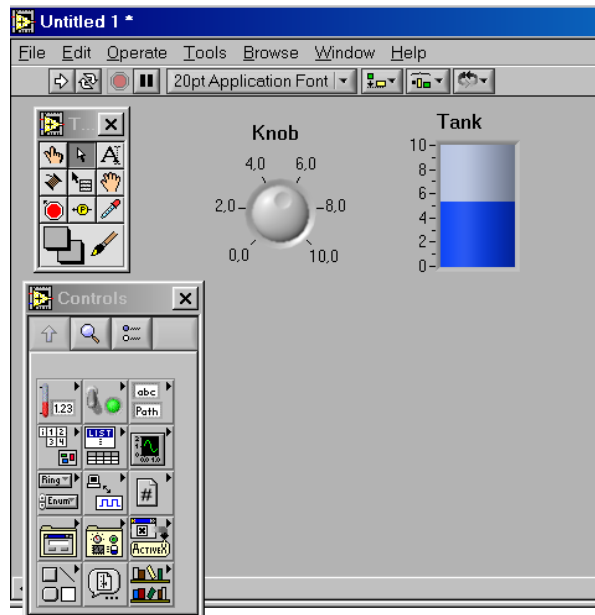


Color Copy tool. Χρησιμεύει στην αντιγραφή χρωμάτων.



Coloring tool. Χρησιμοποιείται για τον χρωματισμό ενός αντικειμένου.

Παρατηρείται, επομένως, ότι στα αριστερά του εμπρόσθιου πλαισίου στο σχ. 1.3 υπάρχουν οι παλέτες **Εργαλείων και Ελέγχου (Tools και Controls)**. Ανάμεσα στα εργαλεία εμφανίζεται ο δείκτης για τη μετακίνηση και την αλλαγή μεγέθους αντικειμένων, το εργαλείο λειτουργίας για την εισαγωγή δεδομένων και τη μεταβολή των τιμών των μεταβλητών (operate value), το εργαλείο για την εισαγωγή κειμένου και το εργαλείο για τη σύνδεση των στοιχείων του δομικού διαγράμματος (connect wire).



Εικόνα 1.4: Εμπρόσθιο πλαίσιο (Front panel) του LabView. Αριστερά φαίνεται η παλέτα εργαλείων (Tools) και η παλέτα ελέγχου (Controls).

1.3.2. Η Παλέτα των Αντικειμένων (Controls) και Ρουτινών (Functions)

Οι Παλέτες αυτές αποτελούνται από υποπαλέτες, καθεμία των οποίων εσωκλείει πλήθος προγραμματιζόμενων αντικειμένων-εικόνων που χρησιμοποιούνται στην κατασκευή ενός VI. Η πρόσβαση σε αυτά τα αντικείμενα γίνεται πατώντας την εικόνα της κάθε υποπαλέτας. Ταυτόχρονα, αν επιθυμείται η επιλογή από συγκεκριμένες υποπαλέτες μπορούμε να τις καρφισώσουμε, ώστε να παραμείνουν στο χώρο του Front Panel ή του Διαγράμματος.

1.3.2.1. Η Παλέτα των Αντικειμένων (Controls)

Τοποθετούνται Controls και Indicators αντικείμενα στο Front Panel μέσα από αυτήν την παλέτα, η οποία διαιρείται σε συγκεκριμένες υποπαλέτες που περιέχουν καθορισμένους τύπους αντικειμένων. Εάν η παλέτα δεν είναι ορατή, επιλέγεται **Show Controls Palette** από το μενού **View**. Εναλλακτικά, ανοίγεται η παλέτα των **Controls** πατώντας δεξί κλικ οπουδήποτε στο χώρο του Front Panel.



Εικόνα 1.5: Η παλέτα των Controls.

Σημείωση: Η παλέτα των Controls είναι διαθέσιμη, μόνο όταν το Front Panel είναι ενεργό.



Υποπαλέτα **Numeric**: Περιέχει controls και indicators για αριθμητικά δεδομένα (ακέραιους, δεκαδικούς).



Υποπαλέτα **Boolean**: Περιέχει controls και indicators για λογικά / ψηφιακά δεδομένα.



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



Υποπαλέτα **List & Table**: Περιέχει controls και indicators για δημιουργία μενού εναλλακτικών επιλογών.



Υποπαλέτα **Array & Cluster**: Περιέχει controls και indicators για ομαδοποίηση δεδομένων.



Υποπαλέτα **Graph**: Περιέχει indicators για γραφική απεικόνιση δεδομένων.



Υποπαλέτα **Refnum**: Περιέχει controls και indicators για επεξεργασία αρχείων.



Υποπαλέτα **Decorations**: Περιέχει γραφικά για εξωραϊσμό των Front Panels.



Υποπαλέτα **User Controls**: Εδώ τοποθετεί ο χρήστης τα δικά του Controls.



Υποπαλέτα **Select a Control**: Δίνεται η δυνατότητα επιλογής και φόρτωσης άλλων Controls.

1.3.2.2. Η Παλέτα των Ρουτινών (Functions)

Γίνεται κατασκευή του Μπλόκ Διαγράμματος με την βοήθεια της παλέτας των ρουτινών. Ομοίως με την παραπάνω παράγραφο, η παλέτα αυτή υποδιαιρείται σε υποπαλέτες που περικλείουν συγκεκριμένους τύπους από προγραμματιζόμενα αντικείμενα-εικόνες. Αν η παλέτα δεν είναι ορατή επιλέγεται **Show Functions Palette** από το μενού **Windows**. Εναλλακτικά, ανοίγεται η παλέτα των **Functions** κάνοντας δεξί κλικ οπουδήποτε στο χώρο του Front Panel.



Εικόνα 1.6: Η Παλέτα των Ρουτινών (Functions).

Σημείωση: Η παλέτα των Functions είναι διαθέσιμη, μόνο όταν το Μπλοκ Διάγραμμα είναι ενεργό.



Υποπαλέτα **Structures**: Περιέχει προγραμματιστικές δομές, όπως While & For Loops.



Υποπαλέτα **Numeric**: Περιέχει ρουτίνες αριθμητικές, λογαριθμικές και τριγωνομετρικές.



Υποπαλέτα **Boolean**: Περιέχει VIs για λογικές πράξεις.



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



Υποπαλέτα **Array**: Χρησιμοποιείται για την επεξεργασία πινάκων.



Υποπαλέτα **Cluster**: Περιέχει ρουτίνες για την επεξεργασία δομών από ανομοιογενή στοιχεία. Οι Δομές αυτές στο LabView καλούνται Clusters.



Υποπαλέτα **Comparison**: Περιέχει ρουτίνες για την σύγκριση στοιχείων,

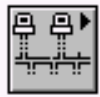
τα οποία μπορεί να είναι αριθμητικά, λογικά ή κείμενο (text).



Υποπαλέτα **Time & Dialog**: Χρησιμοποιείται για Διαλογικά Παράθυρα, Χρονισμό.



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



Υποπαλέτα **Communication**: Περιέχει VIs για επικοινωνία με πρωτόκολλα, όπως TCP, OLE, DDE.



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



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



Υποπαλέτα **Analysis**: Περιέχει 600 διαφορετικούς VIs ανάλυσης δεδομένων.



Υποπαλέτα **Tutorial**: Περιέχει διάφορα VIs που χρησιμοποιούνται στο LabView tutorial.



Υποπαλέτα **Advanced**: Περιέχει διάφορες ρουτίνες για προχωρημένους χρήστες.



Υποπαλέτα **Select a VI...:** Εμφανίζει ένα διαλογικό παράθυρο για την εύρεση και επιλογή υπόVIs.



Υποπαλέτα **Users Library**: Εδώ δίνεται η δυνατότητα στο χρήστη να τοποθετήσει τα δικά του VIs.

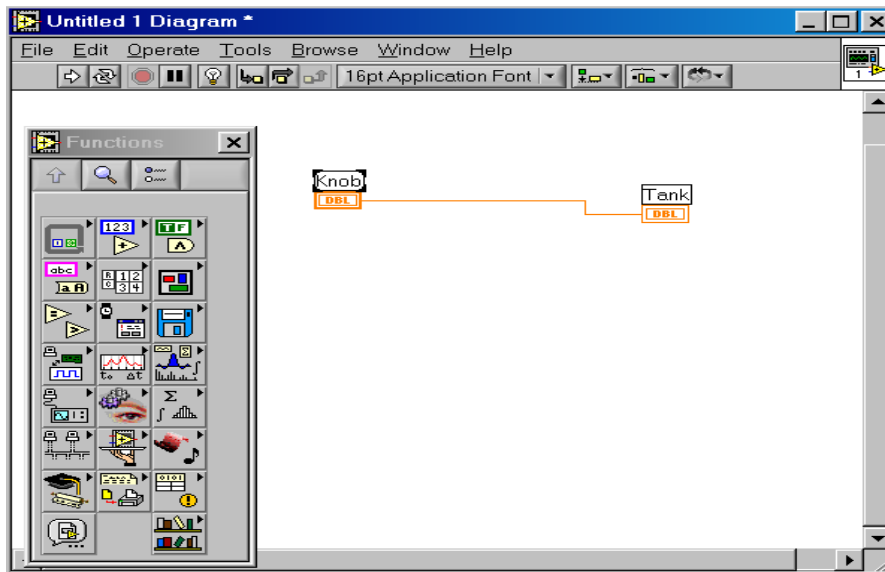


Υποπαλέτα **Instrument Drivers**: Εδώ τοποθετούνται συνήθως Driver VIs για επικοινωνία με όργανα.

1.3.3. Τα εργαλεία του μπλοκ Διαγράμματος (Block Diagram)

Το δομικό διάγραμμα (Block Diagram), υπάρχει πάντα μαζί με το εμπρόσθιο πλαίσιο και ισοδυναμεί με τον κώδικα προγράμματος στη γραφική γλώσσα G. Κάθε στοιχείο του δομικού διαγράμματος αναπαριστάται από ένα εικονίδιο. Ένα κουμπί μπορεί να είναι μια μεταβλητή που να παίρνει τιμές True/False ή μια μεταβλητή διπλής ακρίβειας για εισαγωγή δεδομένων, όπως φαίνεται στο δομικό





διάγραμμα με το αντίστοιχο εικονίδιο που συμβολίζει τη μεταβλητή. Μια συνάρτηση έχει το δικό της εικονίδιο, το ίδιο και κάθε ολοκληρωμένη λειτουργία.



Εικόνα 1.7 Δομικό διάγραμμα (Block diagram)

Αναφέρουμε εδώ, τα επιπλέον εργαλεία που βρίσκει κανείς στο μπλοκ διάγραμμα.



-  Το κουμπί Αργής Κίνησης (Execution Highlighting). Πατώντας το, παρακολουθείται η ροή των δεδομένων μεταξύ των κόμβων, σε αργή κίνηση, και μπορούν εύκολα να διαγνωστούν λάθη στον προγραμματισμό του διαγράμματος.
-  Το κουμπί - Single Step/Step Over. Πατώντας το, εκτελείται το πρόγραμμα από κόμβο σε κόμβο. Φτάνοντας σε έναν κόμβο, υπάρχει η δυνατότητα να αποφύγουμε την εκτέλεσή του, ώστε να εξοικονομήσουμε χρόνο με το κουμπί Step Over, όπως για παράδειγμα σε ένα Loop 1000 κύκλων.
-  Το κουμπί Step Into, αντίθετα με το Step Over επιτρέπει την εισχώρηση στον κόμβο και την παρακολούθηση της λειτουργίας του.
-  Το κουμπί Step Out επιτρέπει, έχοντας εισχωρήσει μέσα στον κόμβο με το Step Into, την έξοδο.



Η ένδειξη προειδοποίησης (Warning Indicator). Παρουσιάζεται όταν υπάρχει πιθανό πρόβλημα στο διάγραμμα, αλλά παρόλα αυτά ο VI μπορεί να εκτελεστεί.

1.3.3.1. Wires (Καλώδια)

Τα καλώδια μεταφέρουν δεδομένα από έναν κόμβο σε έναν άλλο. Ανάλογα τον τύπο μεταβλητής, το χρώμα και το πάχος των καλωδίων αλλάζουν. Τα καλώδια που χρησιμοποιούνται στις βασικές μεταβλητές του LabView φαίνονται στην παρακάτω εικόνα.

Wire-Type	Scalar	1-D Array	2-D Array	Color
Numeric				Orange for Floating point Blue for Integer
Boolean				Green
String				Pink

Εικόνα 1.8: Τα καλώδια (wires)

1.3.3.2. Δομές

Μια δομή αντιπροσωπεύεται από ένα γραφικό περίβλημα. Ο γραφικός κώδικα που περικλείει επαναλαμβάνεται ή εκτελείτε περιστασιακά.

1. For Loop

Η δομή “for loop” χρησιμοποιείται για να πραγματοποιήσουμε επαναλήψεις. Όπως φαίνεται και στην παρακάτω εικόνα το περίγραμμα ορίζει τα όρια της επανάληψης. Το “N” αντιπροσωπεύει τον αριθμό των επαναλήψεων που θα πραγματοποιηθούν, ενώ το “i” τον αριθμό των ολοκληρωμένων επαναλήψεων.



Εικόνα 1.9: Το For Loop.

2. While Loop

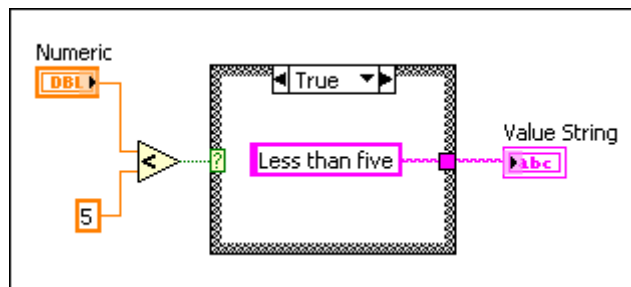
Η δομή “while loop” επιτρέπει τις επαναλήψεις μέχρις ότου μία συνθήκη που έχει οριστεί να γίνει αληθείς (true). Η κόκκινη κουκίδα δείχνει αν η συνθήκη είναι αληθείς, ενώ το “i” όπως και στην For Loop δείχνει τον αριθμό των ολοκληρωμένων επαναλήψεων.



Εικόνα 1.10: To While Loop.

3. Case Structure

Μία δομή Case, επιτρέπει την εκτέλεση διαφορετικών λειτουργιών ανάλογα με την τιμή που λαμβάνεται. Η είσοδος της δομής αντιπροσωπεύεται με το σύμβολο “?”. Στην είσοδο αυτής της δομής υπάρχουν μεταβλητές Boolean, Integer, αλλά και String.



Εικόνα 1.11: To Case Structure.

1.3.3.3. Arrays and Clusters

Ένα **Array** αντιπροσωπεύει μια ομάδα από στοιχεία τα οποία έχουν όλα τον ίδιο τύπο δεδομένων. Σε κάθε στοιχείο ενός πίνακα αντιστοιχεί και ένας δείκτης, έτσι οποιαδήποτε στιγμή υπάρχει πρόσβαση σε όποιο σημείο του πίνακα επιθυμούμε χρησιμοποιώντας τον δείκτη. Στο LabView ο δείκτης ξεκινάει από το 0 και φτάνει μέχρι $n - 1$ όπου είναι το τελευταίο στοιχείο του πίνακα.

Σε αντίθεση με τα arrays, τα **Clusters** μπορούν να περιέχουν δεδομένα από διαφορετικούς τύπους δεδομένων, όπως ακριβώς οι δομές (Structures) σε πολλές γλώσσες προγραμματισμού. Τα Clusters μας βοηθούν στην μείωση των πολλών καλωδίων, αφού για την κατασκευή ενός cluster απαιτείται μόνο ένα καλώδιο για την σύνδεση. Μπορεί να προστεθούν ή να αφαιρεθούν στοιχεία από ένα Cluster χρησιμοποιώντας τις συναρτήσεις Bundle by Name και Unbundle by Name.

Με αυτά τα χαρακτηριστικά, το LabView προωθεί τον προγραμματισμό σε ενότητες. Πρώτα, διαιρείται η κυρίως διάταξη, σε μια σειρά απλών εργασιών. Έπειτα, φτιάχνεται ένα VI για κάθε απλή εργασία και τέλος, συνδυάζονται μαζί για να δημιουργηθεί το τελικό πρόγραμμα (VI).

Ο προγραμματισμός σε ενότητες έχει το πλεονέκτημα ότι κάθε subVI μπορεί να εκτελεστεί αυτόνομα καθιστώντας την αποσφαλμάτωση ευκολότερη.

Επιπρόσθετα, πάρα πολλά απλά VIs εξυπηρετούν συνήθεις εργασίες σε πολλά διαφορετικά προβλήματα και έτσι μπορούν να χρησιμοποιηθούν ανεξάρτητα από κάθε εφαρμογή.

1.3.3.4. 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 δεξί κλικ.

1.3.3.5. Μενού Pull-Down

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

Μενού Αρχείων (File Menu)

Οι επιλογές αυτού του μενού χρησιμοποιούνται κυρίως για να αποθηκεύσουμε, ανοίξουμε, κλείσουμε και να εκτυπώσουμε Vis.

Μενού Αλλαγών (Edit Menu)

Remove Broken Wires: Σβήνει άχρηστες 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: Δυνατότητα σύνδεσης του LabView σε άλλες εφαρμογές από απόσταση.

Μενού Εργαλειων (Tools)

MAX: Το πρόγραμμα αυτό βοηθά στον προσδιορισμό των χαρακτηριστικών των καναλιών συλλογής μετρήσεων.

Instrumentation: Συμβάλλει στην εύρεση Instrument Drivers.

Build Executable: Αν η έκδοση είναι Professional, η επιλογή αυτή δημιουργεί αυτόνομες εφαρμογές.

Show Profile Window: Παρουσιάζεται αν το πρόγραμμα είναι βέλτιστα γραμμένο.

Μενού Παραθύρων και Εμφάνισης (Windows και View Menu)

Show VI Info...: Παροχή πληροφοριών 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

1.3.4. Ο Κοννέκτορας και η Εικόνα (Icon & Connector)

Το εικονίδιο και ο connector ενός VI επιτρέπουν σε άλλα VIs να περνούν δεδομένα στο VI. Το εικονίδιο αναπαριστά ένα VI στο μπλοκ διάγραμμα ενός άλλου VI. Αυτό σημαίνει ότι χρησιμοποιώντας πολλά απλά VIs, δημιουργούνται πολυπλοκότερα καινούρια VIs. Ο connector καθορίζει τις εισόδους και εξόδους του VI. Τα VIs είναι ιεραρχημένα και επεκτάσιμα. Μπορούν να χρησιμοποιηθούν σαν τελικά προγράμματα, σαν υποπρογράμματα μέσα σε άλλα προγράμματα, ακόμη και μέσα σε άλλα υποπρογράμματα. Ένα VI που χρησιμοποιείται μέσα σε ένα άλλο VI, αντίστοιχα με μια υπορουτίνα, ονομάζεται subVI.



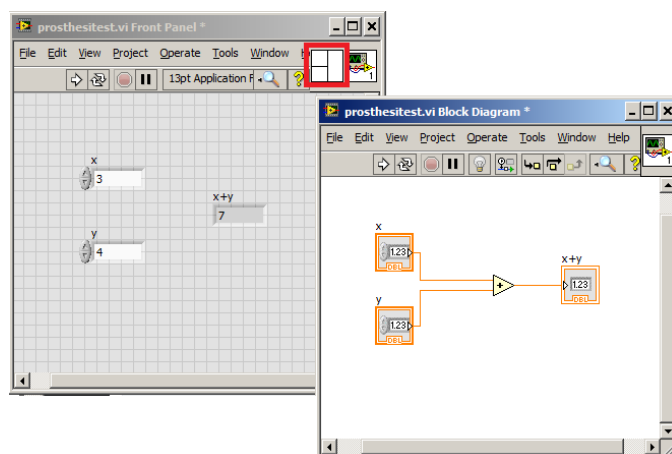
Εικόνα 1.12: Εικονίδιο.



Εικόνα 1.13: Connector.

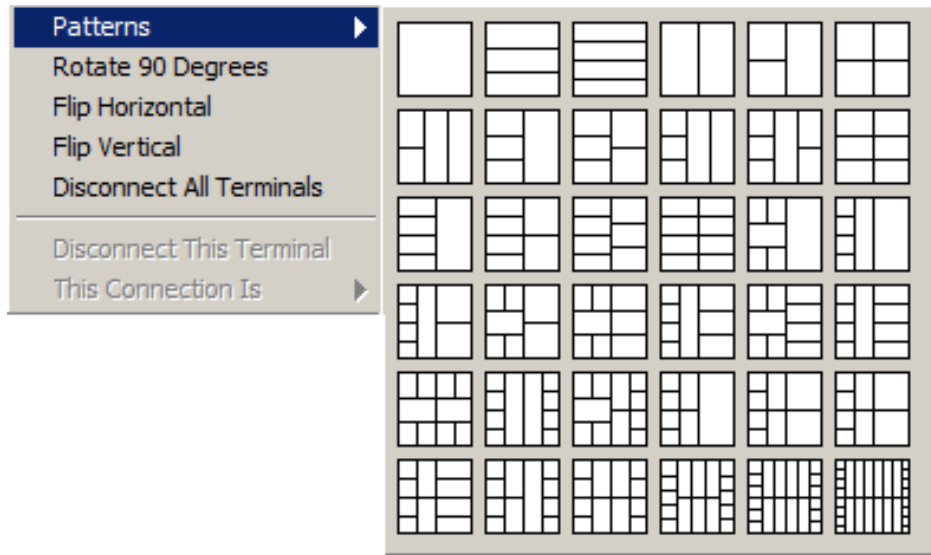
1.3.4.1. Connector Panel

Το Connector Panel βρίσκεται στην πάνω δεξιά γωνία του παραθύρου Front Panel.



Εικόνα 1.14: Το Connector Panel.

Στο Connector Panel ορίζονται οι είσοδοι (controls) και οι έξοδοι (indicators) ενός VI. Ο αριθμός των εισόδων και των εξόδων μπορεί να αλλάξει χρησιμοποιώντας διαφορετικό τύπο Pattern κάνοντας **δεξί Click -> Patterns στο Connector Panel**.



Εικόνα 1.15: Patterns στο Connector Panel.

Στην εικόνα 1.14 φαίνεται ένα εικονίδιο στην δεξιά γωνία του Block Diagram το οποίο αντιστοιχεί σε δύο εισόδους και μία έξοδο του Front Panel.

1.4. Βιβλιοθήκες από Vis (VI Libraries)

Υπάρχει η δυνατότητα φόρτωσης και καταχώρησης VIs σε μια 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, παρέχεται η δυνατότητα χρήσης μέχρι 255 χαρακτήρες για την ονομασία των VIs.
- Τα VI libraries συμπιέζουν τα VIs και εξοικονομούν 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.

1.5. Φορτώνοντας VIs

Η φόρτωση ενός VI στη μνήμη πραγματοποιείται 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 φακέλους με το σύμβολο VI να απεικονίζεται επάνω τους, ενώ τα VIs διατηρούν την κατάληξη .vi. Το άνοιγμα ενός VI Library πραγματοποιείται με τον ίδιο τρόπο ανοίγματος οποιουδήποτε φακέλου. Είτε επιλέγοντάς τον και πατώντας **OK**, είτε πατώντας διπλό κλικ πάνω του. 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 φάκελους που έχουν προσδιοριστεί στο μονοπάτι ανίχνευσης VI Search Path (**Tools** menu » **Options** » **Paths**). Επίσης, μπορεί να αγνοηθεί το υπόVI πατώντας **Ignore SubVI** ή το **Browse** και να συνεχιστεί η έρευνα. Η καταχώρηση ενός VI μπορεί να γίνει σε ένα φάκελο ή μία Βιβλιοθήκη από VIs, επιλέγοντας **Save**, **Save As...**, **Save a Copy As...** από το μενού **File**. Για να δημιουργηθεί μια καινούργια βιβλιοθήκη από VIs επιλέγεται το **Save As...** και το κουμπί **New LLB**. Στο διαλογικό παράθυρο που εμφανίζεται δίνεται ένα όνομα της αρεσκείας σας στην νέα βιβλιοθήκη και ύστερα επιλέγεται το **Create**. Δεν χρειάζεται να γίνει εγγραφή της κατάληξης .llb, γιατί το LabView την τοποθετεί αυτόματα. Μπορεί να γίνει διαγραφή του VIs από μία βιβλιοθήκη με την βοήθεια του **LLB Manager** από το μενού **Tools**.

1.6. Μεταφορά VIs σε άλλες πλατφόρμες

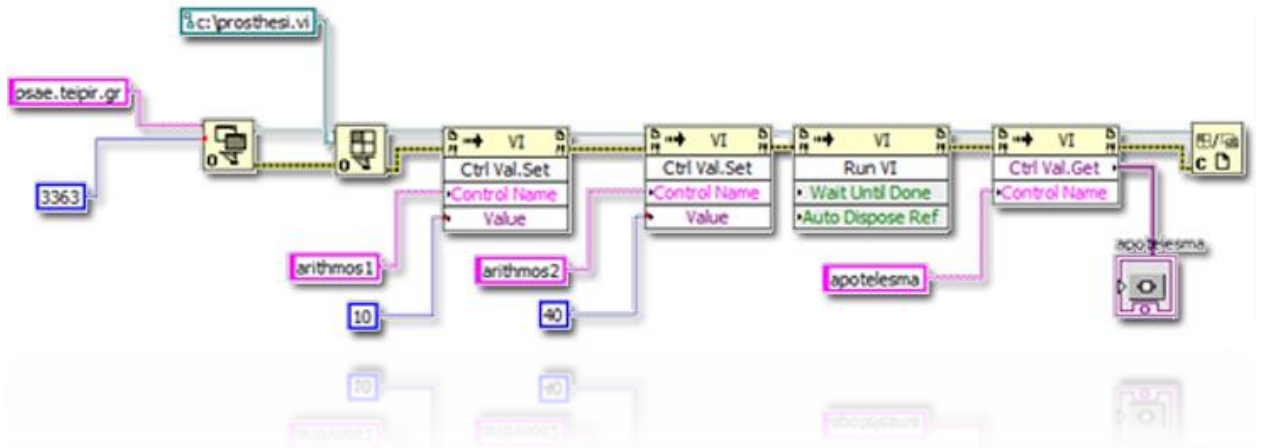
Μπορεί να μεταφερθεί VIs από ένα λειτουργικό σύστημα σε άλλο (για παράδειγμα από LabView for Macintosh σε LabView for Windows). Το LabView αυτόματα μεταφράζει τον κώδικα στην νέα πλατφόρμα.

Η χρήση Βιβλιοθηκών Vis (VI Libraries) διευκολύνει την διαδικασία, καθώς υπάρχει η δυνατότητα μεταφοράς πολλών VIs μαζί και μπορούν να διατηρηθούν, επίσης, μεγάλα ονόματα για τα αρχεία μας. Μπορεί να χρησιμοποιηθεί 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.

(Κενό φύλλο)

2. Ο εξυπηρετητής LabView (VI Server)

Ο VI Server είναι ένα ισχυρό χαρακτηριστικό του LabView που δίνει τη δυνατότητα προγραμματισμού σε διάφορες λειτουργίες του LabView, όπως το άνοιγμα και τη λειτουργία των VI, αλλαγή του χρώματος ή της τιμής ενός αντικειμένου του Front Panel, εκτύπωση του Front Panel και αρκετές άλλες.



Εικόνα 2.1: Ο εξυπηρετητής LabView (Vi Server)

Η λειτουργικότητα του VI Server είναι πραγματικά ένας τρόπος για να εκθέσει την δομή των αντικειμένων του LabView, τα οποία χρησιμοποιούνται για την οργάνωση των Vis, τους ελέγχους, και αρκετές άλλες λειτουργίες.

Για παράδειγμα, με ένα VI Server, δίνεται η επιλογή:

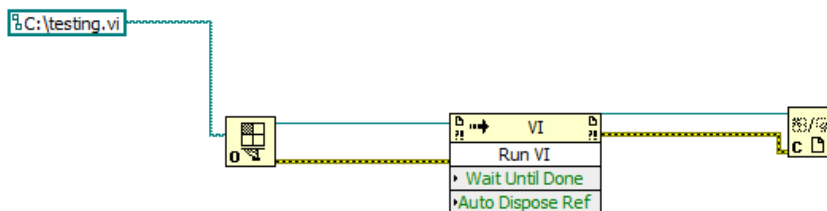
- Να φορτωθεί ένα VI στη μνήμη, να εκτελεστεί, και στη συνέχεια να αποφορτιστεί η μνήμη από το VI.
- Να εκτελεστεί ένα subVI γνωρίζοντας μόνο τη δομή και το όνομά του (γνωστό και ως VI by Reference).
- Αλλαγής ιδιοτήτων ενός συγκεκριμένου VI, όπως το μέγεθος και η θέση του παραθύρου Front Panel.
- Εμφάνισης των παραθύρων του LabView πάντα στην κορυφή της οθόνης. (In front of the screen)
- Δυναμικής αλλαγής των ιδιοτήτων, όπως το χρώμα και το μέγεθος ενός αντικειμένου του Front Panel.

- Κλήσης ενός SubVI από το Block Diagram χωρίς να περιμένουμε από αυτό να τελειώσει την εκτέλεση.

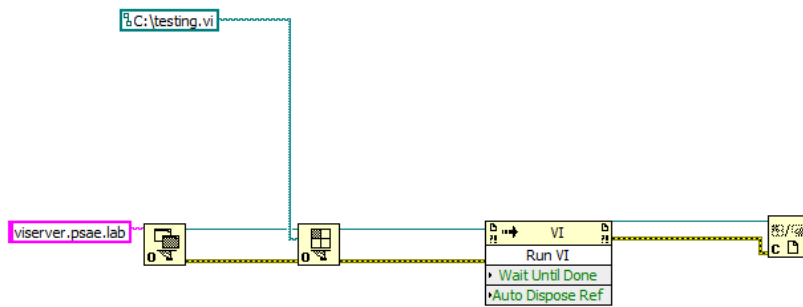
Επειδή τα εσωτερικά αντικείμενα του LabView όπως Vis Controls κλπ. (VIS, έλεγχοι, κλπ.) έχουν σχεδιαστεί χρησιμοποιώντας αρχές αντικειμενοστραφούς προγραμματισμού (OOP), παρατηρούνται πολλά χαρακτηριστικά του OOP να εμφανίζονται στα εργαλεία και στους μηχανισμούς που χρησιμοποιούνται στον VI Server. Για παράδειγμα, ο VI Server ασχολείται με τα αντικείμενα που έχουν ιδιότητες και μεθόδους.

Όλα όσα έχουν αναφερθεί παραπάνω εφαρμόζονται πάνω στο δίκτυο. Αυτό σημαίνει ότι τα παραπάνω μπορούν να εφαρμοστούν σε ένα VI ή και στο LabView που βρίσκεται σε ένα άλλο μηχάνημα μέσα στο δίκτυο (συμπεριλαμβανομένου και του διαδικτύου), ακριβώς με τον ίδιο τρόπο, σαν να ήταν στο δικό σας μηχάνημα. Έτσι, το απομακρυσμένο LabView μπορεί να εκτελεστεί ακόμα και σε ένα διαφορετικό λειτουργικό σύστημα.

Στο παρακάτω παράδειγμα θα περιγραφεί πώς μπορεί να εκτελεστεί ένα VI από απόσταση. (Τα περισσότερα από τα controls που χρησιμοποιήθηκαν βρίσκονται στην καρτέλα Application Controls).

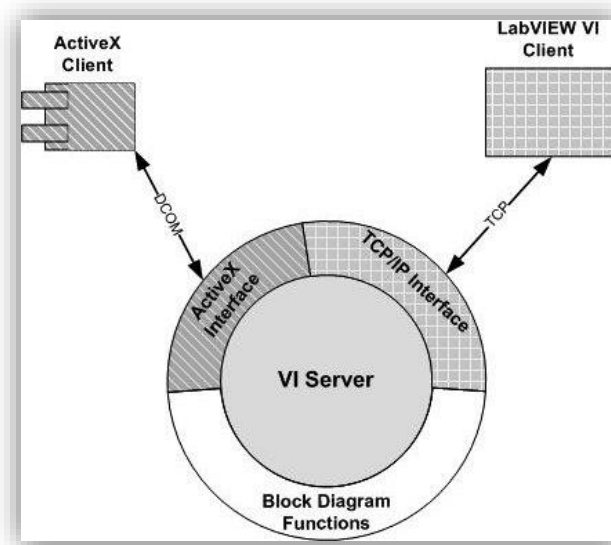


Εικόνα 2.2: Εκτέλεση του testing.vi που βρίσκεται στον τοπικό δίσκο



Εικόνα 2.3: Εκτέλεση του testing.vi που βρίσκεται στην απομακρυσμένη διεύθυνση viserver.psae.lab

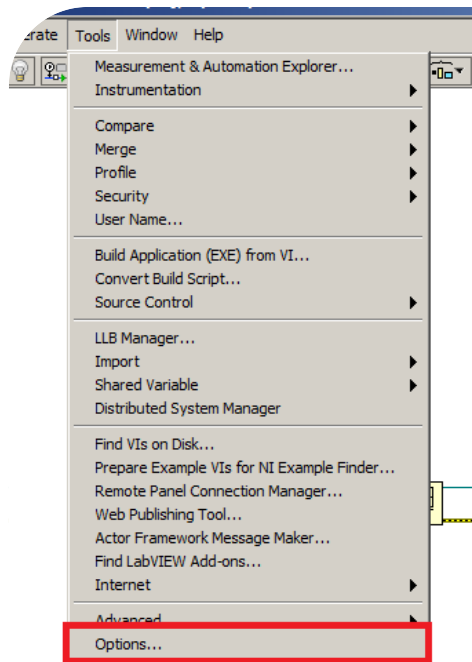
Ο VI Server εκθέτει την λειτουργικότητά του στο LabView με ActiveX αλλά και μέσω του πρωτοκόλλου TCP/IP.



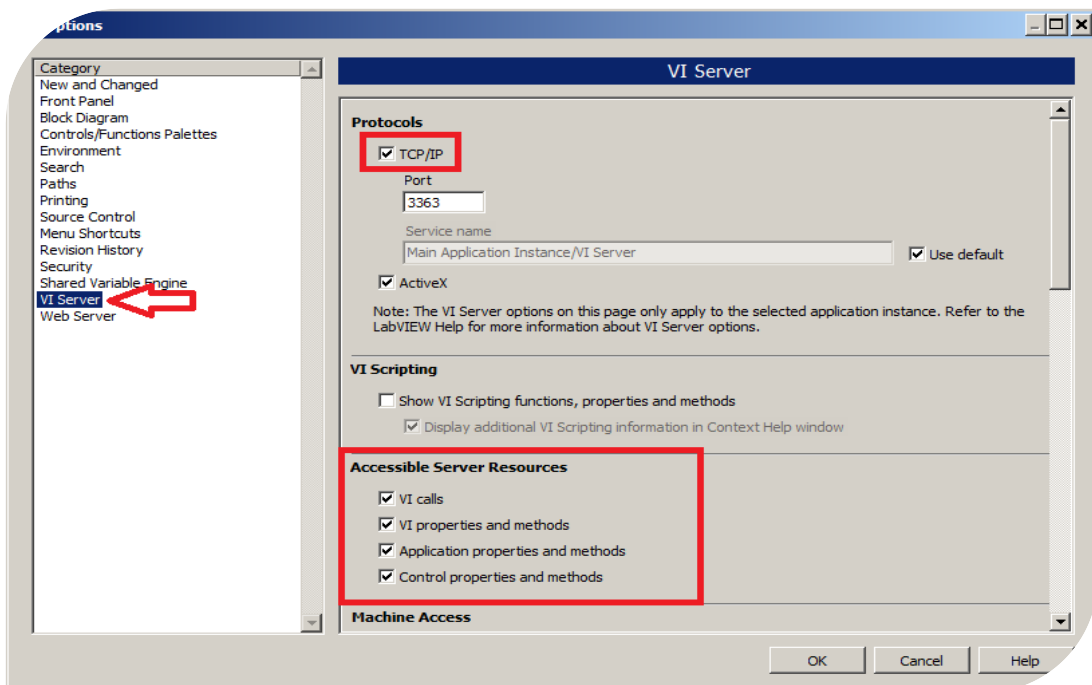
Εικόνα 2.4: Δομή VI Server

2.1. Ενεργοποίηση απομακρυσμένης πρόσβασης στο VI Server

Για να ενεργοποιηθεί η απομακρυσμένη πρόσβαση στο LabView για πρόσβαση στον VI Server, επιλέγονται **Tools** και μετά **Options**.



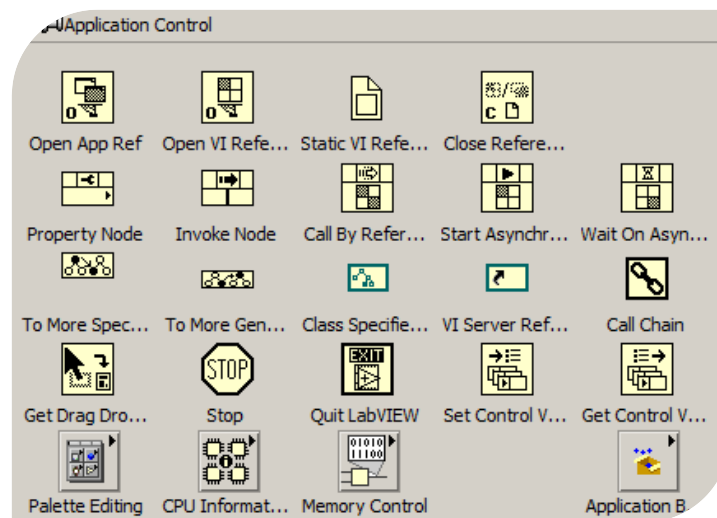
Στην συνέχεια, γίνεται μετάβαση στην καρτέλα VI Server, ενεργοποίηση του TCP/IP και έλεγχος για τον περιορισμό πρόσβασης σε ορισμένες λειτουργίες των VI.



Εικόνα 2.5: Επιλογές διαμόρφωσης VI Server.

Δίνεται πάντα πλήρης πρόσβαση στον VI Server του τοπικού LabView, στο οποίο το VI είναι σε λειτουργία (ανεξάρτητα από τις ρυθμίσεις του VI Server που διαμορφώθηκαν). Εάν είναι επιθυμητή η πρόσβαση στον τοπικό VI Server, χρησιμοποιείται μια τοπική αναφορά εφαρμογής (Local Application Reference).

Για παράδειγμα, εάν καλέσουμε την Open VI Reference και αφεθεί το application reference αποσυνδεδεμένο, τότε η τοπική αναφορά εφαρμογής (Local Application Reference) χρησιμοποιείται από προεπιλογή.



Εικόνα 2.6: Η παλέτα Application Controls

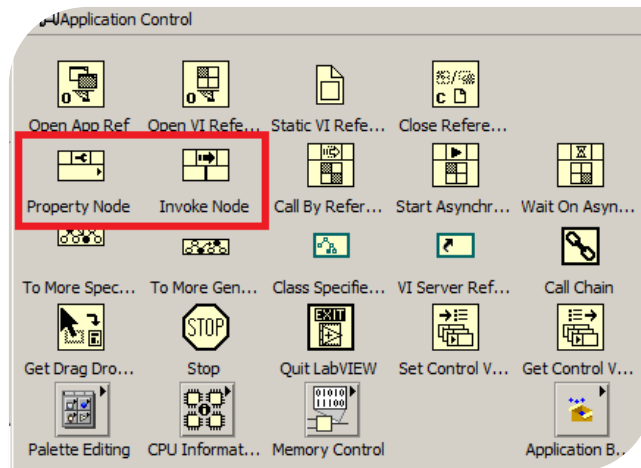
Υπάρχουν τρεις πολύ σημαντικές κατηγορίες των "αντικειμένων" του LabView τις οποίες ο VI Server επιτρέπει να χειριστούμε:

Application Class: Κλήση του ίδιου του περιβάλλοντος του LabView.

VI Class: Κλήση ενός VI που υπάρχει στην μνήμη ή στον δίσκο.

Control Class: Κλήση ενός Front Panel ή Indicator

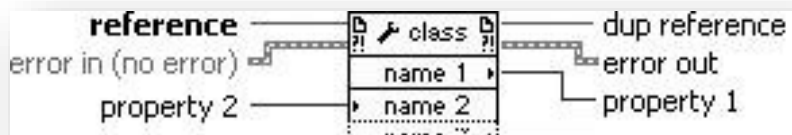
Για να χρησιμοποιηθούν αντικείμενα σε VI Server, γίνεται χρήση κυρίως του **Property Node** και του **Invoke Node** (Programming -> Application Control).



Εικόνα 2.7: Property Node και Invoke Node

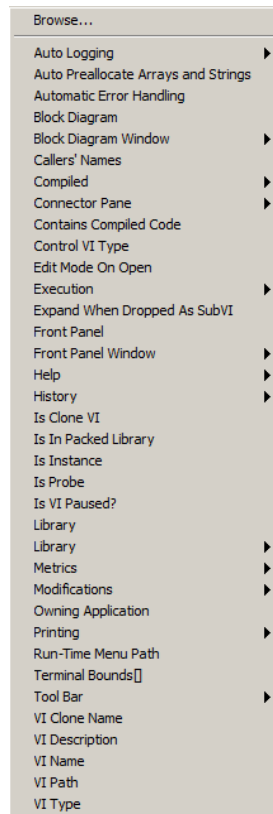
2.1.1. Property Node

Το Property Node καθορίζει (γράφει) ή παίρνει (διαβάζει) πληροφορίες για τις ιδιότητες του αντικειμένου το οποίο πέρασε στην είσοδο “reference” (Εικόνα 2.6). Μπορεί να ρυθμιστεί το Property Node έτσι ώστε να μπορεί να έχει πρόσβαση σε πολλές ιδιότητες.



Εικόνα 2.8: Property Node

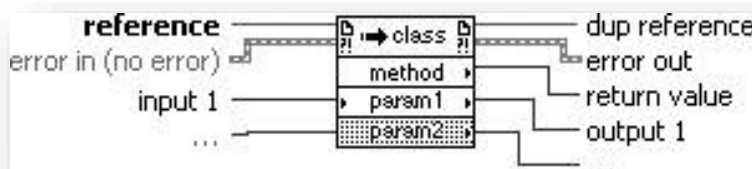
Στο παρακάτω σχήμα εμφανίζονται οι ιδιότητες του Property Node όταν αυτό συνδεθεί με ένα VI.



2.1.2. Invoke Node

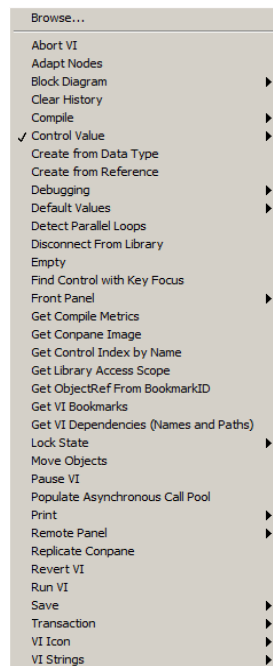
Το Invoke Node επικαλείται μια μέθοδο σχετικά με το αντικείμενο του οποίου περνά στην είσοδο αναφοράς (Reference). Μόλις επιλεγεί η μέθοδος, οι συνδεδεμένες παράμετροι θα εμφανίζονται κάτω από το όνομα της μεθόδου. Μπορεί να καθοριστεί η είσοδος και να ληφθούν ως έξοδος οι τιμές των παραμέτρων.

Παράμετροι με λευκό φόντο σημαίνει ότι είναι υποχρεωτικό να δηλωθούν εισοδοι ενώ οι παράμετροι με γκρι φόντο σημαίνει ότι συνιστάται να ενωθεί μια είσοδος. Ένα Invoke Node μπορεί να ρυθμιστεί, ώστε να καλεί μόνο μια μέθοδο.

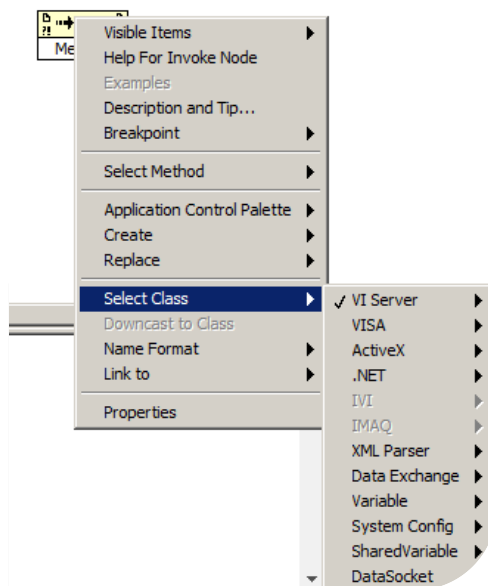


Εικόνα 2.9: Invoke Node

Στο παρακάτω σχήμα παρουσιάζονται οι ιδιότητες του Invoke Node όταν αυτό συνδεθεί με ένα VI.



Αν η είσοδος Reference είναι συνδεδεμένη, τότε το Property Node και το Invoke Node θα προσαρμοστούν αυτόματα στην κλάση του αντικειμένου που συνδέθηκε. Αν η είσοδος Reference δεν είναι συνδεδεμένη, τότε μπορεί να επιλεγεί η κλάση του αντικειμένου από το **Select Class** κάνοντας δεξί κλικ πάνω τους.



Εικόνα 2.10: Κλάση αντικειμένου

Αν επιλέξουμε το VI ή το Application Class, δεν χρειάζεται να συνδεθεί η είσοδος reference υποθέτοντας ότι αναφέρεστε στο καλούμενο VI ή την τοπική εφαρμογή αντίστοιχα.

2.2. Application References (Αναφορές Εφαρμογής)

Μια αναφορά εφαρμογής είναι μια αναφορά σε ένα στιγμιότυπο του Application Class στο LabView. Αυτό σημαίνει ότι μια αναφορά εφαρμογής δίνει πρόσβαση στην ανάγνωση και στη σύνταξη ιδιοτήτων, αλλά και στην πρόσβαση σε μεθόδους της ίδιας της εφαρμογής του LabView, σε αντίθεση με ένα VI.

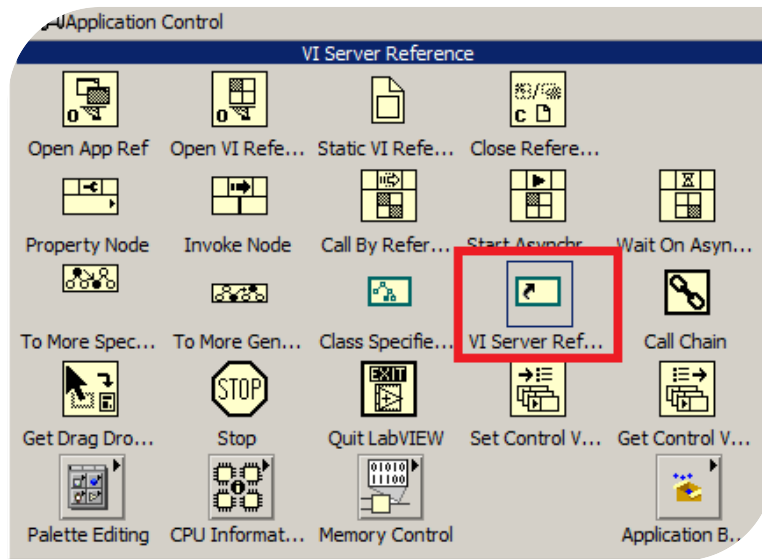
2.3. Από την θεωρία στην πράξη (Παράδειγμα)

Ας υποθέσουμε ότι το πρόγραμμα χρειάζεται να γνωρίζει την ανάλυση της οθόνης του χρήστη. Με ένα Application Reference, δίνεται η δυνατότητα αναγνώρισης της ανάλυσης της οθόνης που τρέχει το LabView διαβάζοντας την ιδιότητα "Display.AllMonitors".

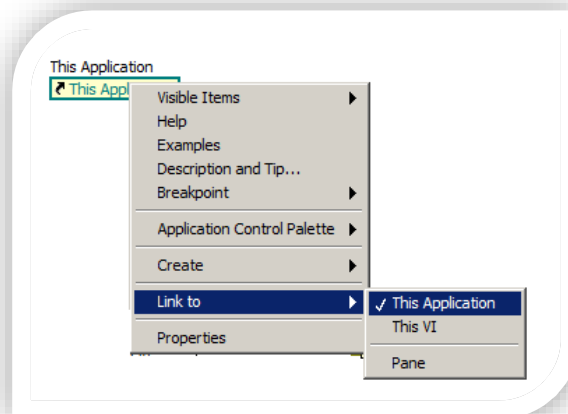
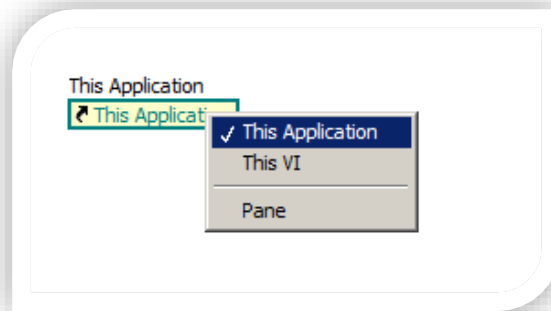
Ας υποθεθεί εναλλακτικά ότι το πρόγραμμα ανοίγει μια εξωτερική εφαρμογή, όπως το Microsoft Excel, αλλά επιδιώκεται να βεβαιωθεί ότι το παράθυρο του VI θα επανέλθει σαν πρώτο παράθυρο. Έτσι, καλείται η μέθοδος "Bring To Front", η οποία δίνει εντολή στο LabView να μεταφέρει το παράθυρο σε πρώτο πλάνο, δηλαδή μπροστά από όλα τα άλλα παράθυρα.

Πριν την ανάγνωση ή την εισαγωγή των ιδιοτήτων πρέπει να δημιουργηθεί ένα Application Reference. Συνήθως το Application Reference αναφέρεται στην τοπική εγκατάσταση του LabView, αλλά μπορεί να αναφέρεται και σε μία απομακρυσμένη εγκατάσταση.

Η απόκτηση μίας αναφοράς εφαρμογής είναι εύκολη και υπάρχουν πολλοί τρόποι να γίνει. Ένας τρόπος είναι να χρησιμοποιηθεί το VI Server Reference από την παλέτα **Programming -> Application Control** και να επιλεγεί το **This application** κάνοντας κλικ πάνω του και πηγαίνοντας στο **Link To**.



Εικόνα 2.11: VI Server Reference



Ένας άλλος τρόπος είναι να χρησιμοποιηθεί ένα Property Node (**Programming -> Application Control**) και να αφεθεί η είσοδος Reference ασύνδετη, όπως φαίνεται στο παρακάτω σχήμα. Έτσι, το LabVIEW υποθέτει ότι

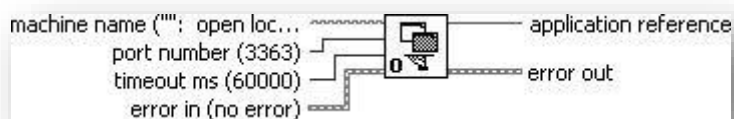
γίνεται αναφορά στην τοπική εφαρμογή και σαν έξοδος θα υπάρχει μία τοπική αναφορά εφαρμογής (**Local Application Reference**).



Εικόνα 2.12: Reference ασύνδετη

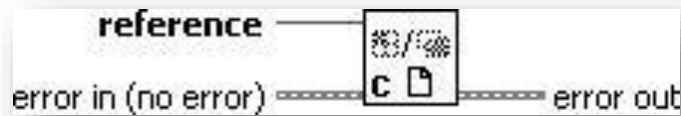
Ο τελευταίος τρόπος για να αποκτηθεί μία αναφορά εφαρμογής είναι να καλεστεί το **Open Application Reference** και να αφεθεί αποσυνδεδεμένη η είσοδος “Machine Name”. Με αυτό τον τρόπο το LabView υποθέτει ότι γίνεται λόγος για τοπική αναφορά εφαρμογής.

Βέβαια με το Open Application Reference αποκτάται και μια αναφορά εφαρμογής που είναι σε απομακρυσμένο μηχάνημα. Για να γίνει αυτό θα πρέπει να δηλωθεί το όνομα του μηχανήματος (**συνήθως μια IP Address**) και η πύλη, που χρησιμοποιεί το LabView στο απομακρυσμένο μηχάνημα, η οποία είναι υπεύθυνη για τις συνδέσεις TCP.



Εικόνα 2.13: Open Application Reference

Αφού τελειώσει το Open Application Reference πρέπει να κλείσει χρησιμοποιώντας το Close Reference (**Programming -> Application Control**).

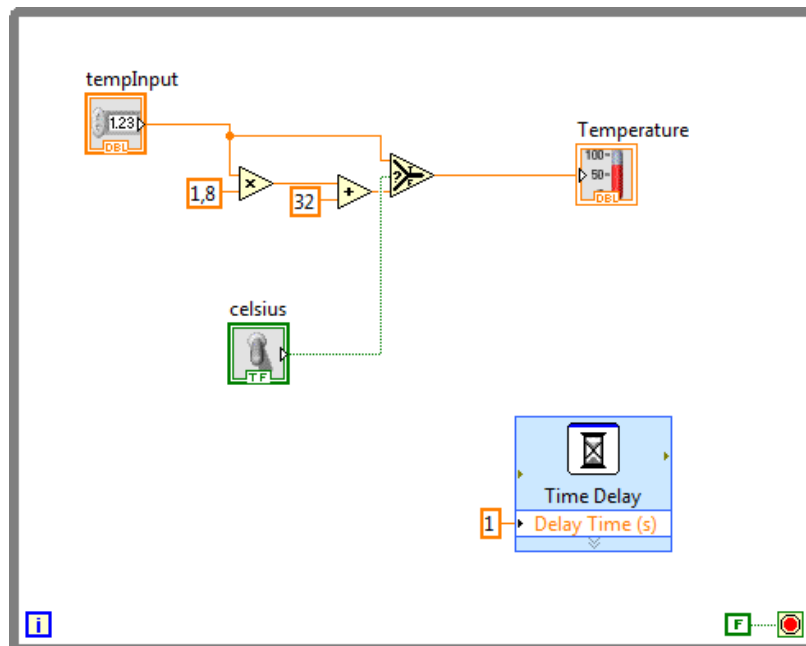


Εικόνα 2.14: Close Reference

2.4. Παράδειγμα VI Server

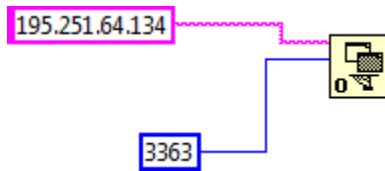
Στο παρακάτω παράδειγμα παρουσιάζεται πως χειρίζεται ένα απομακρυσμένο VI.

Για το συγκεκριμένο παράδειγμα δημιουργείται ένα πολύ απλό VI στη χρήση. Δέχεται μια θερμοκρασία για είσοδο και απεικονίζει την τιμή αυτή σε βαθμούς Celsius ή Fahrenheit ανάλογα με την τιμή της μεταβλητής Boolean Celsius.



Εικόνα 2.15: Παράδειγμα VI Server

Ξεκινώντας, κατασκευάζοντας το Block Diagram που βρίσκεται στον Client πρέπει να προστεθεί ένα Open App Reference και να ορισθεί η Port και η IP από τον server.

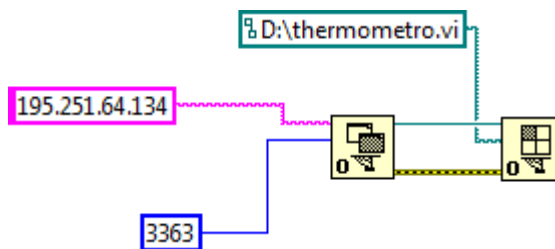


Εικόνα 2.16: Open App Reference

Χρησιμοποιείται ένα String Constant για την IP και ένα Number Constant για την πύλη.

Το Open App Reference πρέπει να συνδεθεί με VI Reference στο οποίο θα δωθεί το Path που βρίσκετε το απομακρυσμένο VI.

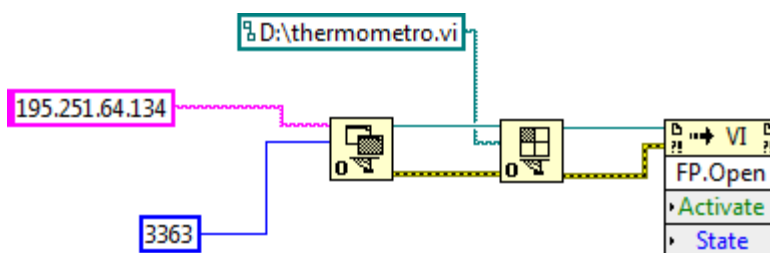
Για να ορισθεί το Path χρησιμοποιείται ένα **Path Constant** και το συνδέεται με την είσοδο Path του VI Reference.



Εικόνα 2.17: Path, IP, Port Constant

Πλέον μπορεί να χρησιμοποιηθεί το **Property Node** και το **Invoke Node** για να γίνει χειρισμός των μεθόδων και των αντικειμένων του VI αλλά και του Application.

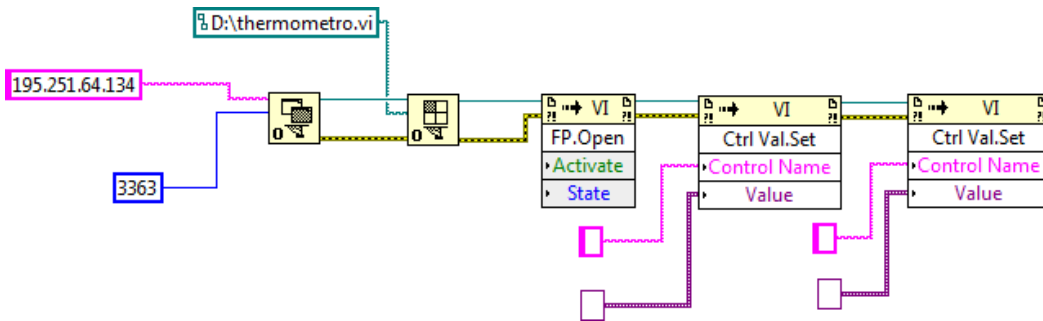
Προσθέτεται ένα Invoke Node το οποίο επιβεβαιώνει ότι το Front Panel του VI είναι ανοιχτό. Αυτό πραγματοποιείται χρησιμοποιώντας την μέθοδο **Front Panel - > Open**.



Εικόνα 2.18: Invoke Node (FP.Open Method)

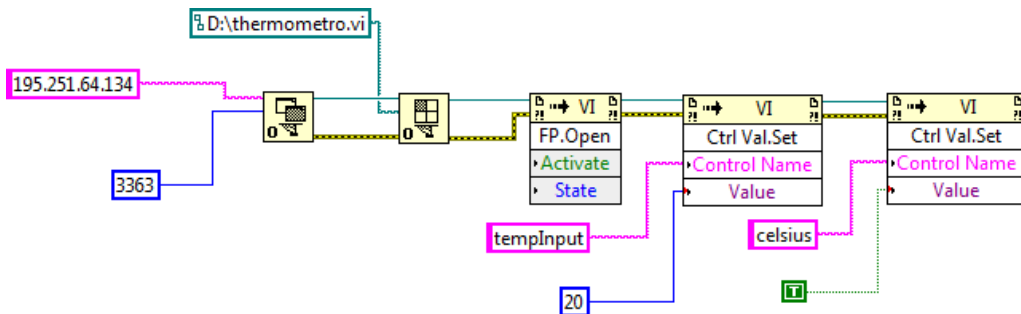
Πλέον αφού φαίνονται οι αλλαγές στο Front Panel, ορίζονται οι δύο μεταβλητές, το tempInput και το Celsius.

Για να γίνει αυτό χρησιμοποιούνται δύο Invoke Node για την κάθε μεταβλητή, και έπειτα επιλέγεται η μέθοδος **Control Value -> Set**.

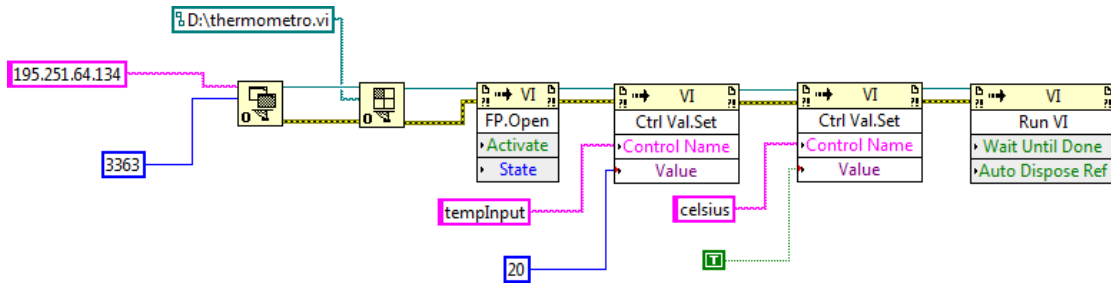


Εικόνα 2.19: Invoke Node (Ctrl.Set Method)

Η μέθοδος **Control Value -> Set** παίρνει δύο ορίσματα- το όνομα της μεταβλητής και την τιμή την οποία ορίζουμε σε αυτή. Χρησιμοποιούνται δύο **String Constant** ("tempInput", "celsius") για τα ονόματα των δύο μεταβλητών που έχουμε στο παράδειγμα, ένα **Numeric Constant** για την τιμή του **tempInput** και ένα **Boolean Constant** για την τιμή του **Celsius**.

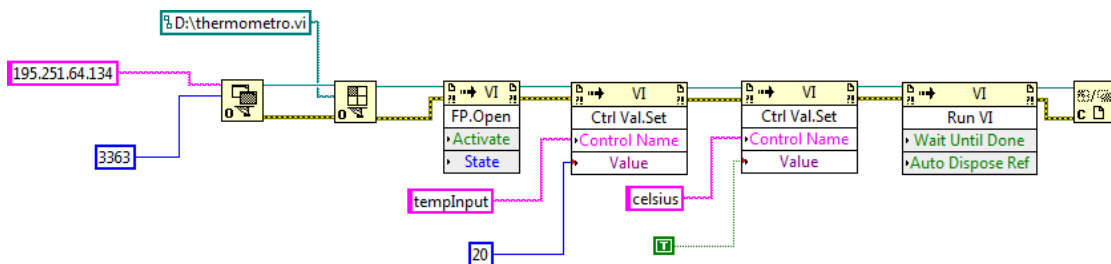


Εφόσον έχουν οριστεί οι δύο μεταβλητές που χρειάζεται το απομακρυσμένο VI για να λειτουργήσει, μπορεί να πραγματοποιηθεί η **εκτέλεση (απομακρυσμένα)**. Για την εκτέλεση θα χρησιμοποιηθεί η μέθοδος **Run VI**.



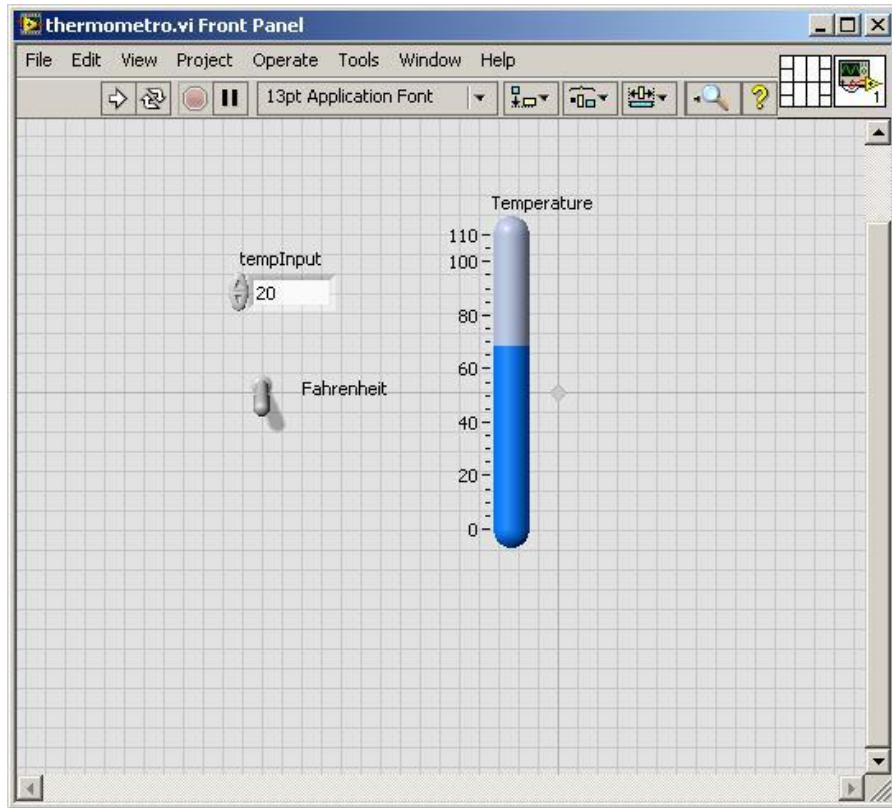
Εικόνα 2.20: Invoke Node (Run VI)

Έχοντας βάλει όλα τα απαραίτητα στοιχεία για να λειτουργήσει το VI, προστίθεται και το **Close Reference** για να οριστεί το τέλος.



Εικόνα 2.21: Close Reference

Έτσι, εκτελώντας το VI παρατηρούνται αλλαγές στον Server.

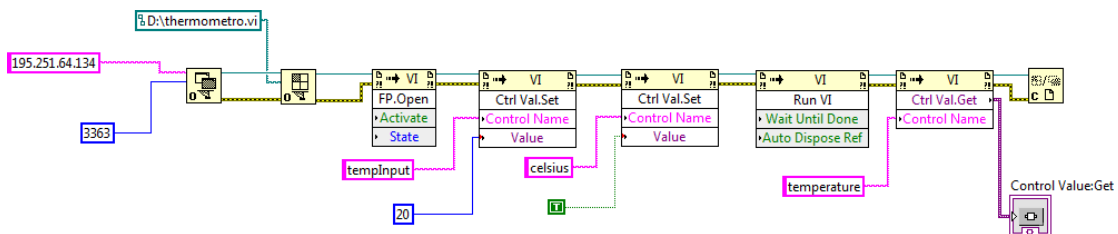


Εικόνα 2.22: Server's Front Panel

Να σημειώσουμε σε αυτό το σημείο ότι το παραπάνω παράδειγμα υποθέτει ότι υπάρχει φυσική επαφή για να διαπιστωθούν τα επιθυμητά αποτελέσματα.

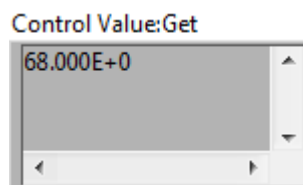
Ορισμένες φορές παρ' όλα αυτά **μπορεί να μην** υπάρχει φυσική επαφή, οπότε πρέπει να μεταφέρουμε τα αποτελέσματα στο δικό μας VI.

Επομένως, πρέπει να αφαιρεθεί το **Front Panel -> Open** και να προστεθεί μετά το Run VI method η μέθοδος **Control Value -> Get** για να αντλήσουμε το αποτέλεσμα της μεταβλητής «temperature».



Εικόνα 2.23: Invoke Node (Ctrl.Val.Get Method)

Εκτελώντας , εμφανίζεται το αποτέλεσμα που αναμενόταν.

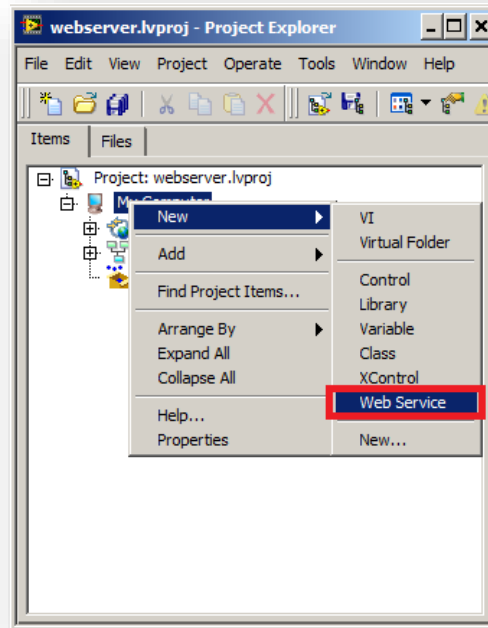


Εικόνα 2.24 Αποτέλεσμα Ctrl.Val.Get

(Κενό φύλλο)

3. Δημιουργία μιας υπηρεσίας Web με το LabView

Δημιουργείται ένα νέο project κάνοντας δεξί κλικ στο My Computer και έπειτα **New -> Web Service**. (Η επιλογή αυτή προστέθηκε στο LabView σε επόμενες εκδόσεις από την 2011).

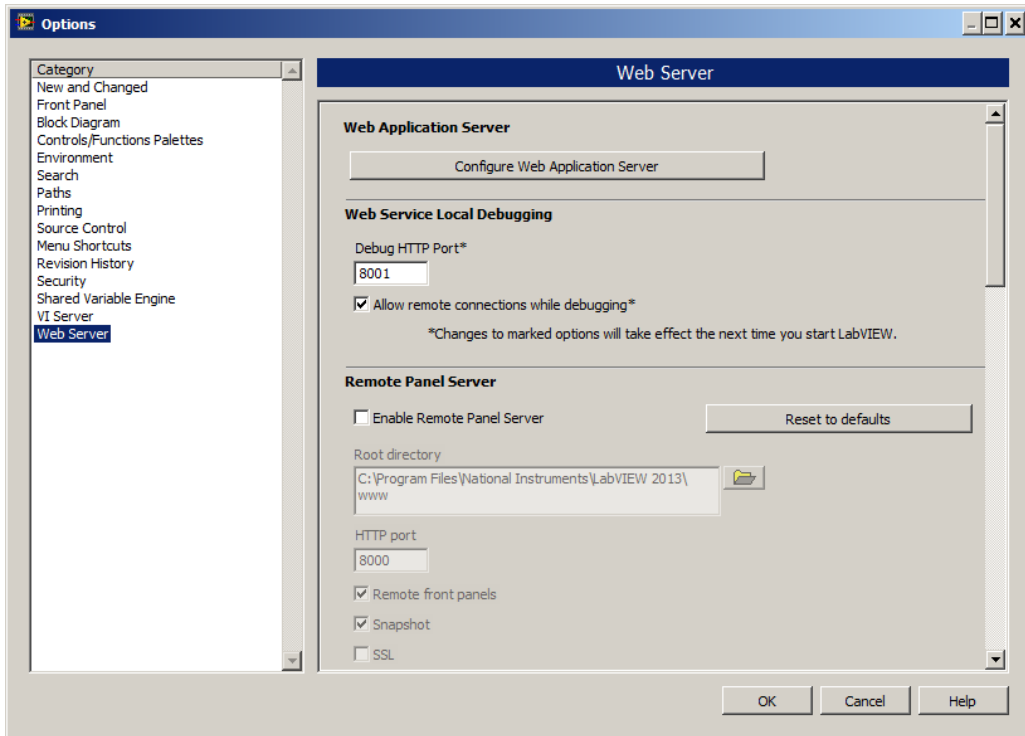


Εικόνα 3.1: Δημιουργία Web Service

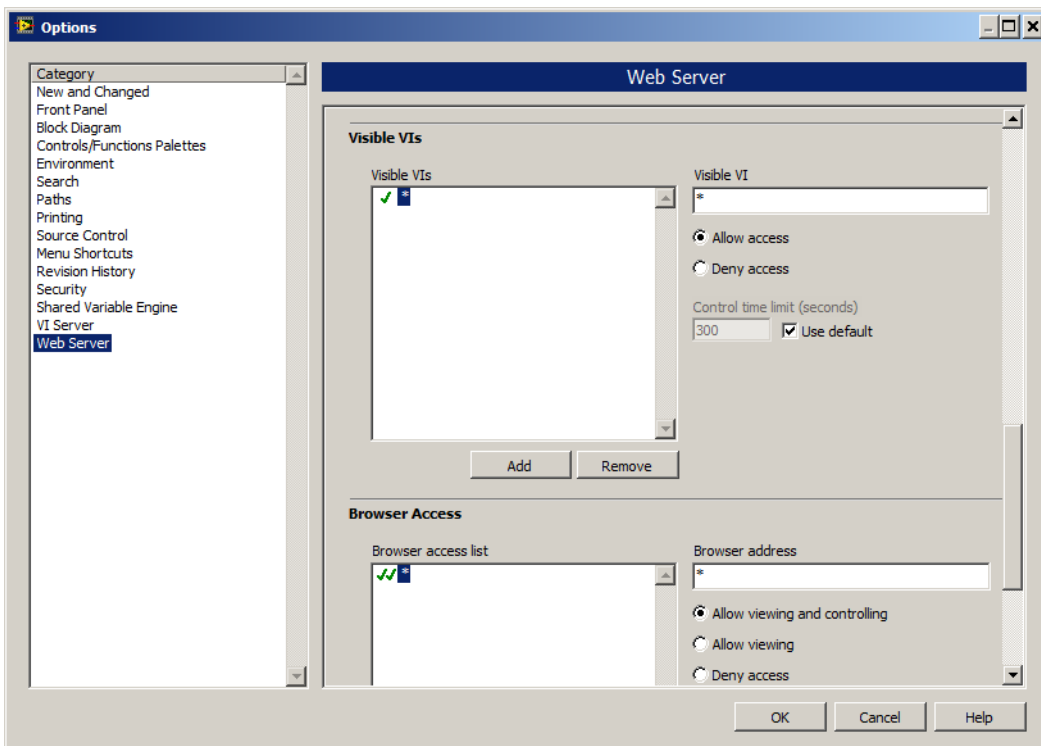
Έτσι το LabView δημιουργεί ένα νέο Web Service project το οποίο μπορεί να μετονομαστεί με **δεξί κλικ -> Rename**.

Γίνεται παραμετροποίηση του Web Server πηγαίνοντας **Tools -> Options -> WebServer**.

Εκεί βρίσκονται ρυθμίσεις για την πόρτα που θα χρησιμοποιεί, ποια VI θα είναι προσβάσιμα από τον WebServer, αλλά και να επιτρέπεται ή να αποτρέπεται η πρόσβαση σε συγκεκριμένες IPs.

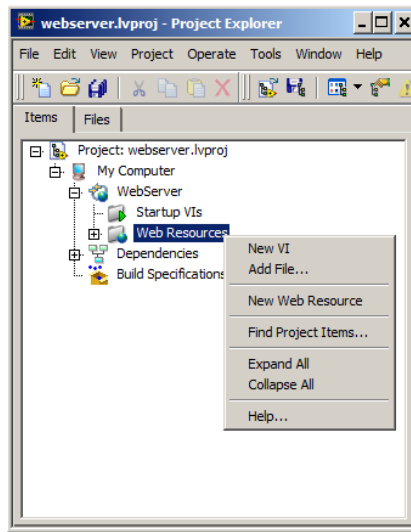


Εικόνα 3.2: Παραμετροποίηση Web Server



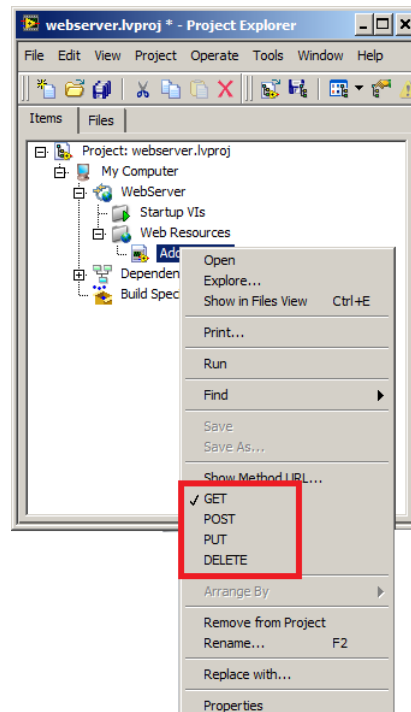
Εικόνα 3.3: Visible Vis, Browser Access

Κάνοντας δεξί κλικ στο Web Services δημιουργείται ένα νέο VI το οποίο θα παίρνει HTTP requests από έναν Client.



Εικόνα 3.4: Δημιουργία Web Resource VI

Το LabView δίνει τη δυνατότητα να γίνει επιλογή ανάμεσα από τέσσερις μεθόδους HTTP. GET, POST, PUT και DELETE.

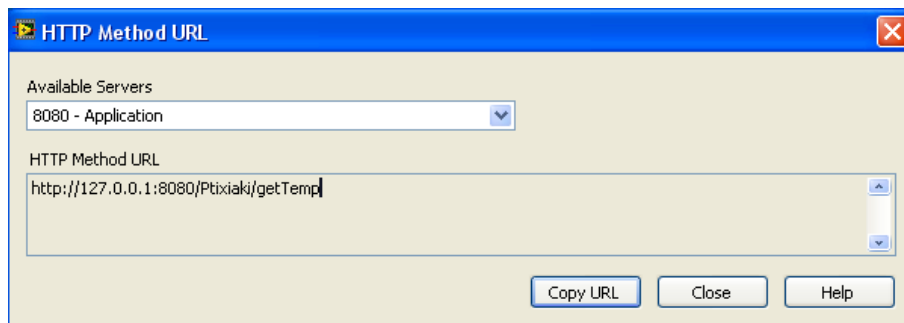


Εικόνα 3.5: Επιλογή μεθόδων HTTP

- Το **GET** χρησιμοποιείται για την ανάγνωση δεδομένων από μία εφαρμογή.
- Το **POST** χρησιμοποιείται για τη δημιουργία δεδομένων σε μία εφαρμογή.
- Το **PUT** χρησιμοποιείται για την ενημέρωση δεδομένων σε μία εφαρμογή.
- Το **DELETE** χρησιμοποιείται για τη διαγραφή δεδομένων σε μία εφαρμογή.

4. Σύνδεση HTTP Request με το LabVIEW

Αφού επιλεγεί η μέθοδος του HTTP Request το LabVIEW δίνει τη δυνατότητα κάνοντας δεξί κλικ στο VI και έπειτα Show Method URL να εμφανιστεί το URL στο οποίο είναι διαθέσιμο το συγκεκριμένο request.

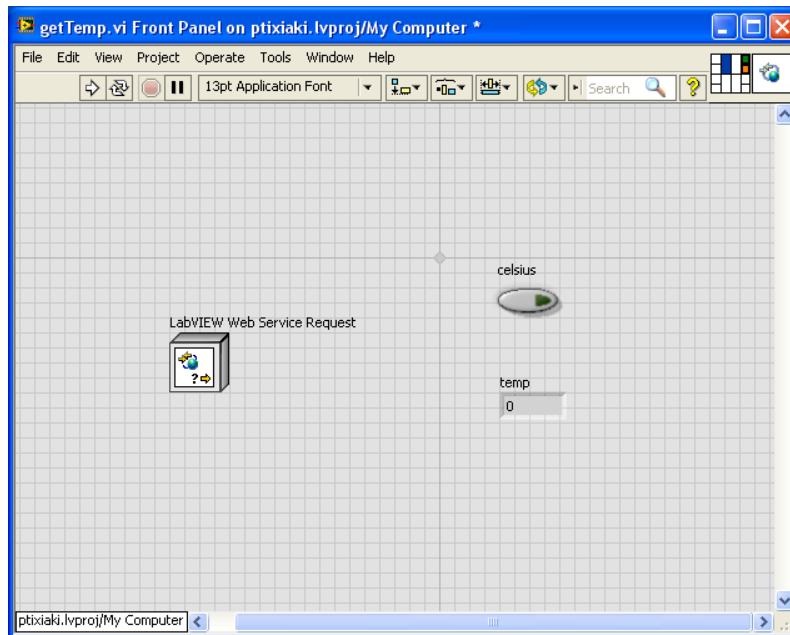


Εικόνα 4.1: Προβολή URL

4.1. Μεταφορά δεδομένων μεταξύ των HTTP VIs και του WebClient

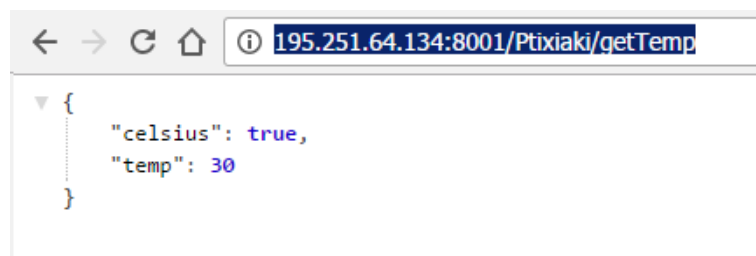
Τα HTTP VIs δέχονται και στέλνουν δεδομένα μέσω του Connector Panel. Όταν ένας WebClient στέλνει ένα HTTP Request σε ένα συγκεκριμένο URL το οποίο είναι συνδεδεμένο με ένα HTTP VI, το VI παίρνει τις τιμές των μεταβλητών και μέσω του Connector Panel τις επιστρέφει στον Client.

Στο παρακάτω παράδειγμα φαίνεται το Front Panel και το Connector Panel από ένα GET Request το οποίο επιστρέφει στον Client την θερμοκρασία και αν είναι Celsius ή Fahrenheit.



Εικόνα 4.2: Front Panel, Connector Panel

Πλέον θα επιστρέφονται στον Client οι δύο μεταβλητές που είναι συνδεδεμένες με το Connector Panel κάθε φορά που γίνεται ένα GET Request στο συγκεκριμένο URL.



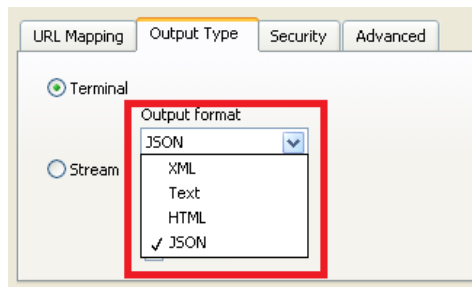
Εικόνα 4.3: JSON δεδομένα

Στην παραπάνω φωτογραφία φαίνεται ότι τα δεδομένα έχουν εμφανιστεί σε μία μορφή γνωστή ως JSON. Το LabView δίνει την δυνατότητα αλλαγής της μορφής της απάντησης που επιστρέφει ο LabVIEW WebServer.

Κάνοντας δεξί κλικ στο Webservice και έπειτα στο Properties, στην καρτέλα HTTP Method VI Settings εμφανίζονται όλα τα HTTP VIs την μέθοδο που χρησιμοποιούν αλλά και το URL που είναι συνδεδεμένα.

Επιλέγοντας ένα VI εμφανίζονται περισσότερες πληροφορίες για το κάθε ένα, όπως αλλαγή της μεθόδου, του URL, της μορφής της απάντησης και η προσθήκη ασφάλειας σε αυτό ώστε να μην είναι προσβάσιμο από όλους.

Οι διαθέσιμες μορφές που δίνει το LabVIEW είναι αυτές που φαίνονται στην παρακάτω εικόνα (XML, Text, HTML, JSON).



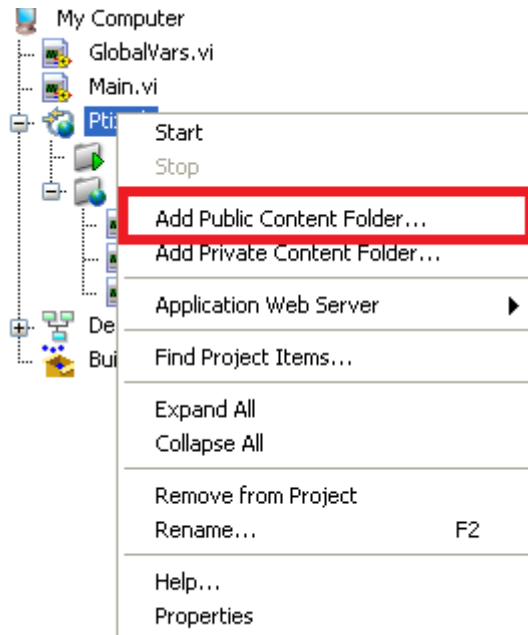
Εικόνα 4.4: Επιλογή μορφής δεδομένων

Επιλέγοντας την μορφή XML, ο Webserver επιστρέφει την αντίστοιχη μορφή απάντησης.



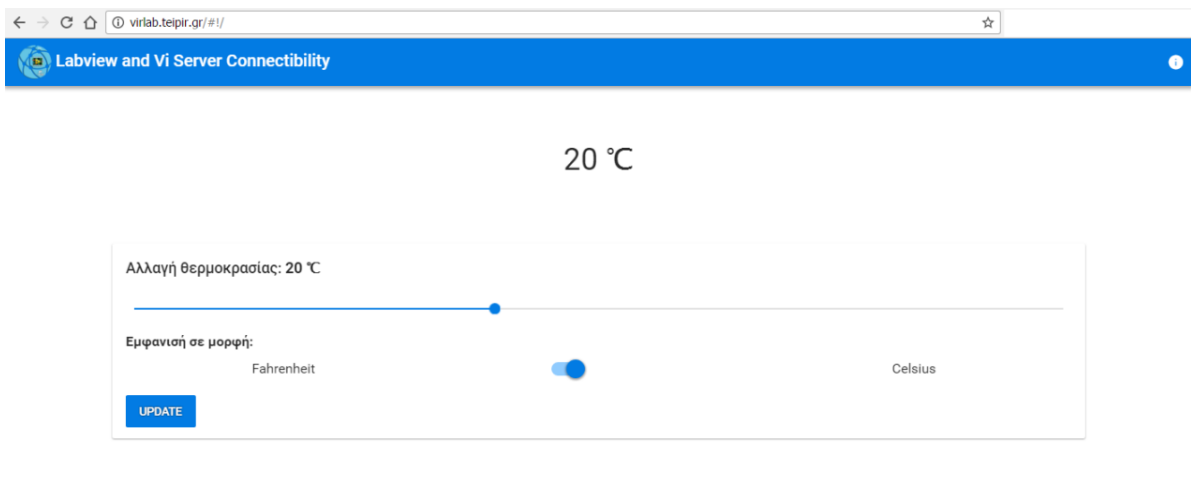
Εικόνα 4.5: XML δεδομένα

Το LabView δίνει την δυνατότητα δημιουργίας ιστοσελίδας προσθέτοντας αρχεία που μπορεί να χρησιμοποιήσει ένας WebClient (html, css, js κλπ.) κάνοντας δεξί κλικ στο Webservice και έπειτα στο Add Public Content Folder.



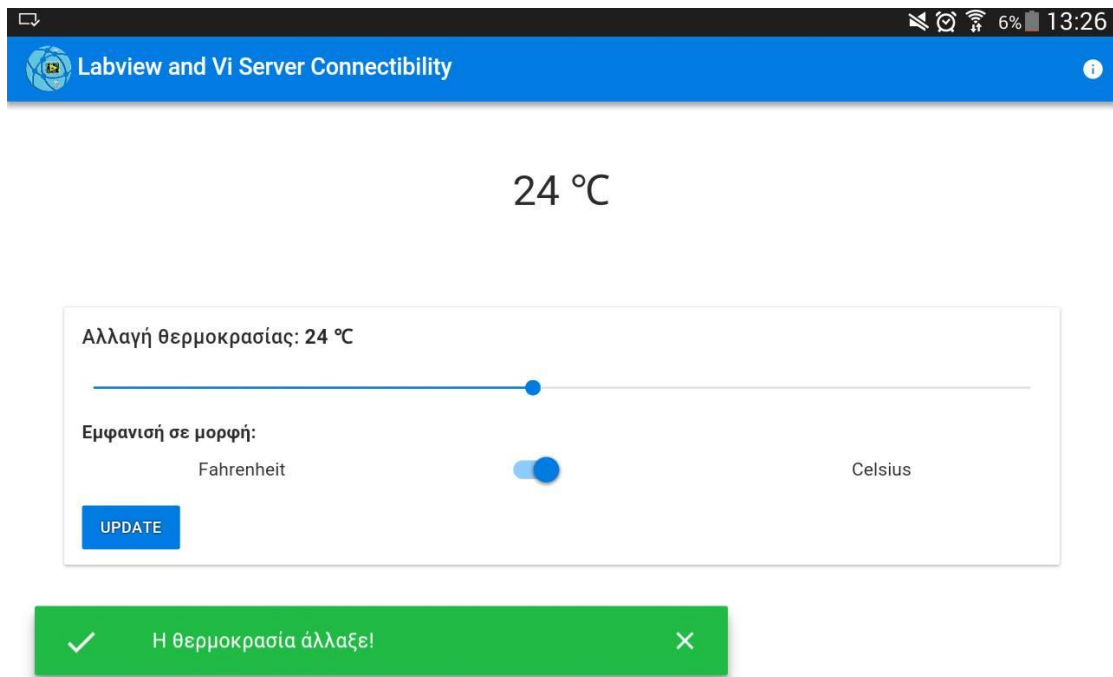
Εικόνα 4.6: Προσθήκη Δημόσιου Φακέλου

Έτσι, επιλέγοντας τον φάκελο με τα αρχεία που έχουν δημιουργηθεί, γίνονται προσβάσιμα μέσω του LabView Webserver.



Εικόνα 4.7: Web Application

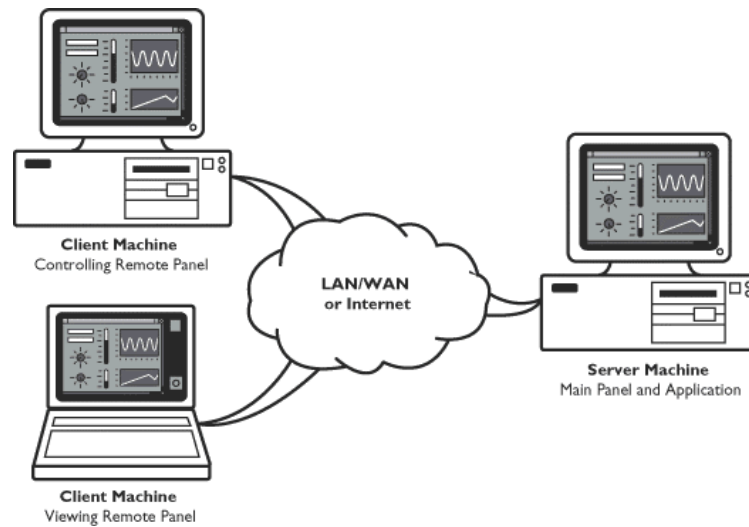
Τα HTTP Requests μπορούν να χρησιμοποιηθούν όχι μόνο από έναν WebBrowser. Πλέον οι περισσότερες γλώσσες προγραμματισμού έχουν την δυνατότητα να χειρίζονται HTTP Requests και έτσι προσφέρεται η δυνατότητα δημιουργίας εφαρμογών για κινητά, σύνδεση με Arduino – Raspberry κτλ.



Εικόνα 4.8: Προβολή URL

(Κενό φύλλο)

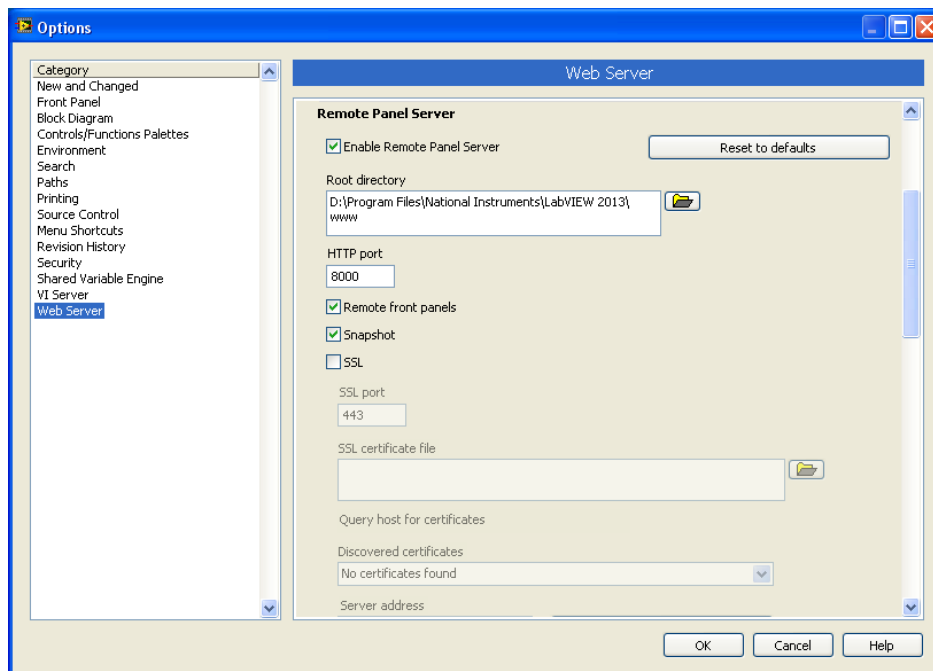
5. Remote Panel Server



Εικόνα 5.1: Remote Panel Server

Το **Remote Panel** επιτρέπει σε έναν υπολογιστή Client να δει και να ελέγξει το Front Panel ενός VI το οποίο βρίσκεται σε έναν υπολογιστή ή Server.

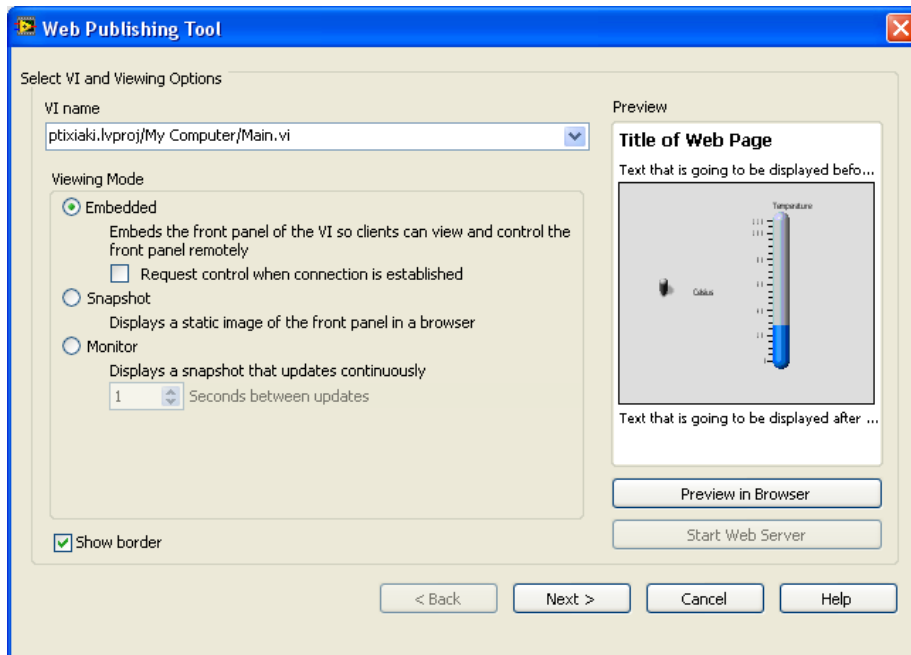
Η ενεργοποίηση αυτής της επιλογής βρίσκεται κάτω από την καρτέλα του WebServer στις ρυθμίσεις.



Εικόνα 5.2: Παραμετροποίηση Remote Panel Server

Με το Remote Panel δίνεται η δυνατότητα ενσωμάτωσης ενός Front Panel σε μία ιστοσελίδα και ο έλεγχος του από εκεί. Το μόνο που χρειάζεται για να γίνει αυτό είναι να υπάρχει το Run-time engine plugin.

Για την δημιουργία μίας σελίδας από την οποία θα υπάρχει πρόσβαση σε ένα VI μπορεί να γίνει χρήση του Web Publishing Tool (**Tools -> Web Publishing Tool**).



Εικόνα 5.3: Web Publishing Tool

Στο παράθυρο που ανοίγει εμφανίζεται η δυνατότητα της επιλογής του VI, το οποίο θα εμφανιστεί στην ιστοσελίδα, αλλά και η δυνατότητα επιλογής ενός από τους τρεις τρόπους προβολής του.

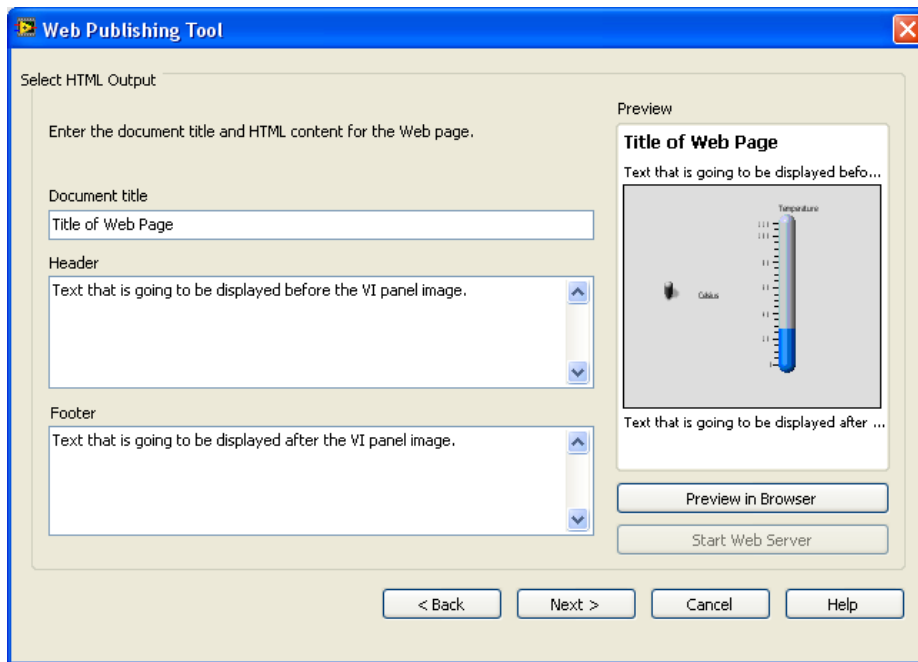
5.1. Μέθοδος Embedded

Η μέθοδος Embedded ενσωματώνει το VI στον Web Browser έτσι ώστε ο Client να μπορεί να δει σε πραγματικό χρόνο τις αλλαγές που γίνονται στο FrontPanel, αλλά και να πάρει τον έλεγχό του.

Για την τεχνολογία Embedded χρειάζεται η χρήση του Internet Explorer και του RunTimeEngine, (http://ftp.ni.com/support/softlib/labview/labview_runtime/2013/Windows/LVRTE2013min.exe).

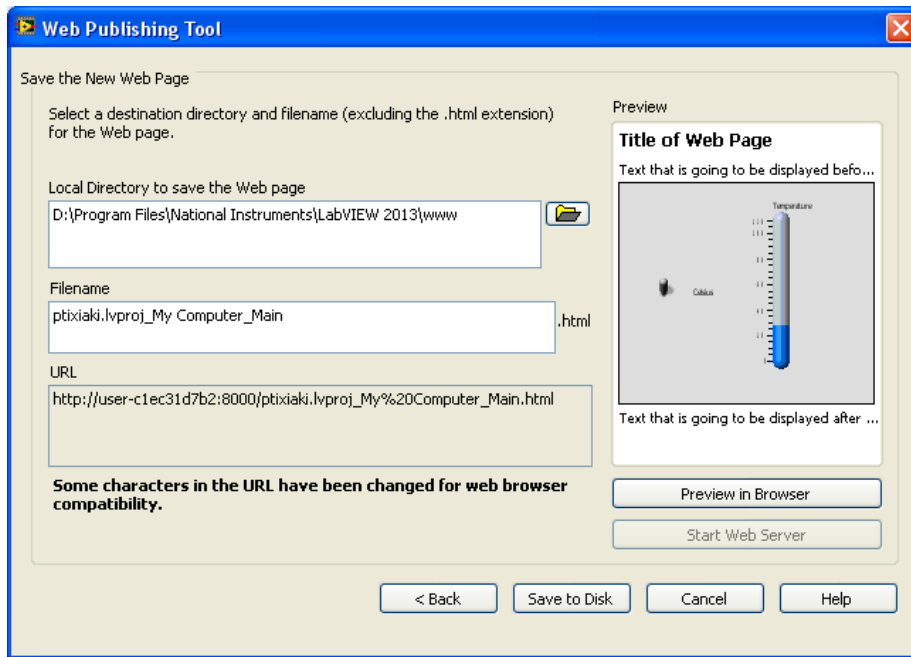
Επιλέγοντας την επιλογή “Request control when connection is established” μπορεί να ελεγχθεί το VI από τον Browser.

Σε επόμενο βήμα το LabView δίνει τη δυνατότητα αλλαγής κάποιων στοιχείων της ιστοσελίδας, όπως ο τίτλος, η επικεφαλίδα που βρίσκεται πάνω από το VI και η κατακλείδα που βρίσκεται κάτω από το VI, αλλά και μία προεπισκόπηση της ιστοσελίδας πριν γίνει Publish.



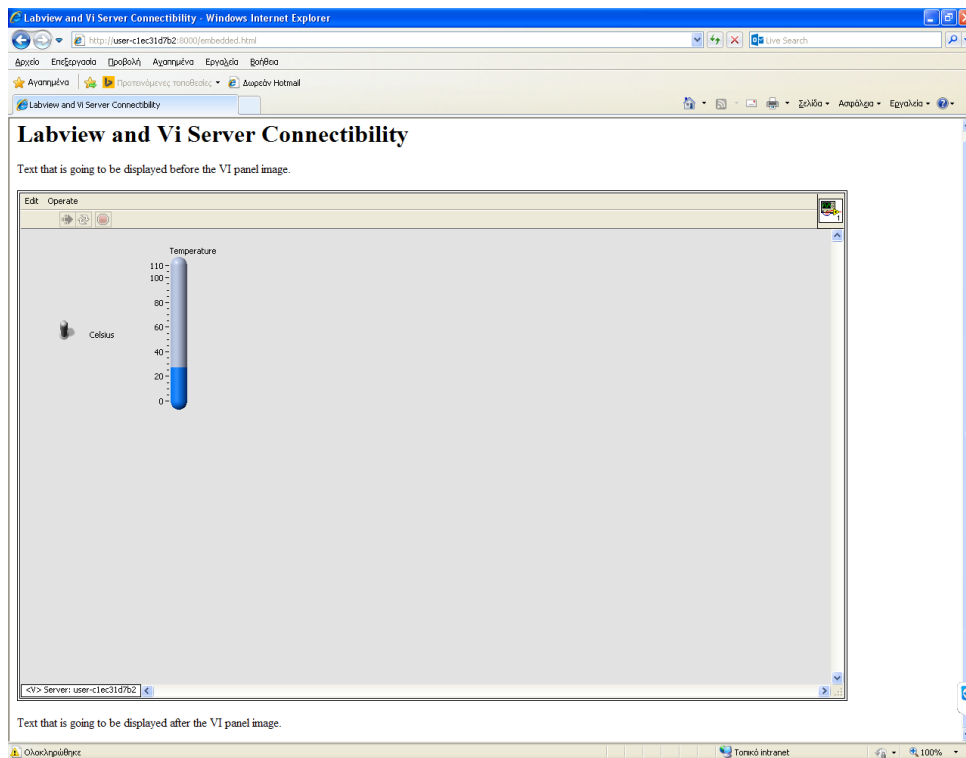
Εικόνα 5.4: Ρυθμίσεις Embedded

Μπορεί να αλλαχτεί το όνομα και η διαδρομή που θα αποθηκευτεί η σελίδα. Με βάση αυτά δημιουργείται και το αντίστοιχο URL που θα είναι προσβάσιμα το VI στον Web Browser.



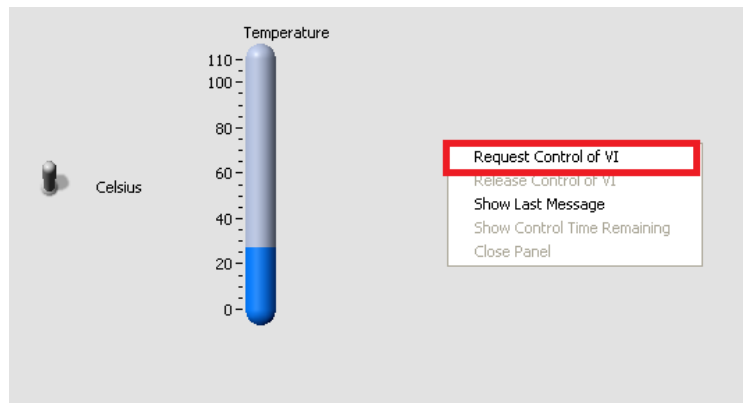
Εικόνα 5.5: Επιλογές Path, Filename, URL για το αρχείο

Αφού γίνει Save to Disk η πλοήγηση στο URL θα δώσει την επιλογή ελέγχου του VI.



Εικόνα 5.6: Web Embedded

Επιλέγοντας το Request Control of VI κάνοντας δεξί κλικ στο VI, δίνετε η δυνατότητα έλεγχου του VI.



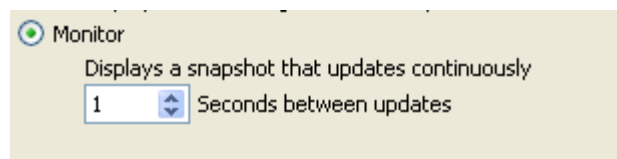
Εικόνα 5.7: Απόκτηση ελέγχου VI

5.2. Μέθοδος Snapshot

Η μέθοδος αυτή δημιουργεί μία φωτογραφία του Front Panel και την προβάλλει στον Web Browser. Σε αυτή την λειτουργία δεν υπάρχει ζωντανή προβολή του VI και δεν είναι δυνατός ο έλεγχος του VI.

5.3. Μέθοδος Monitor

Δημιουργεί μία φωτογραφία η οποία ανανεώνεται ανά X δευτερόλεπτα ανάλογα με την τιμή που έχει επιλεγεί. Ούτε εδώ υπάρχει ζωντανή προβολή του Front Panel και έλεγχος αυτού.

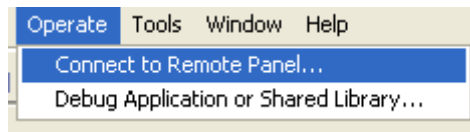


Εικόνα 5.8: Επιλογή ρυθμού ανανέωσης

5.4. Connect to Remote Panel

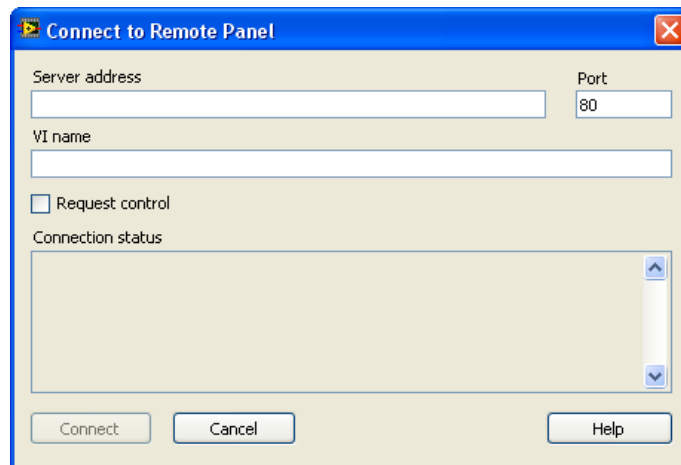
Το LabView δίνει την δυνατότητα ελέγχου ενός Front Panel σε ένα Server από ένα άλλο LabView (Client).

Για την σύνδεση σε ένα απομακρυσμένο LabView γίνεται επιλογή από την καρτέλα Operate το Connect to Remote Panel.

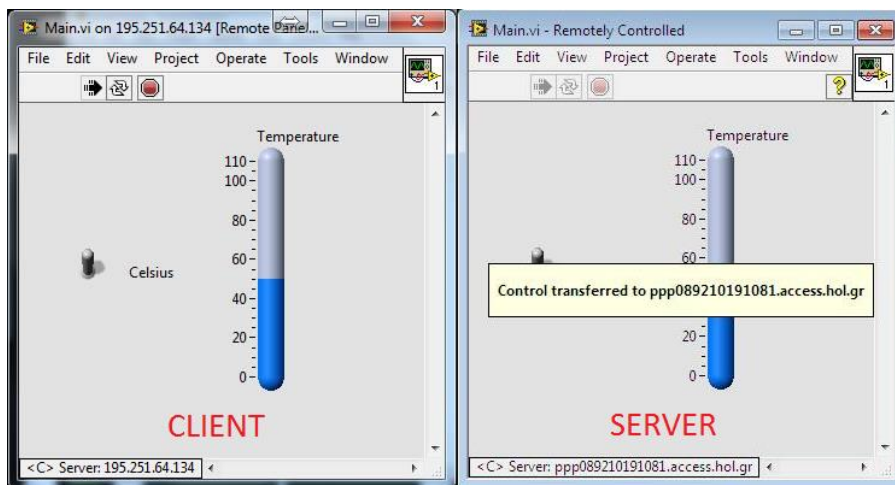


Εικόνα 5.9: Σύνδεση σε Remote Panel

Στο παράθυρο που ανοίγει γίνεται η συμπλήρωση των στοιχείων του Server που θα γίνει η σύνδεση (Server address, port και VI name). Στην περίπτωση που θέλουμε να ελέγξουμε το VI θα πρέπει να είναι επιλεγμένο το πεδίο Request Control.



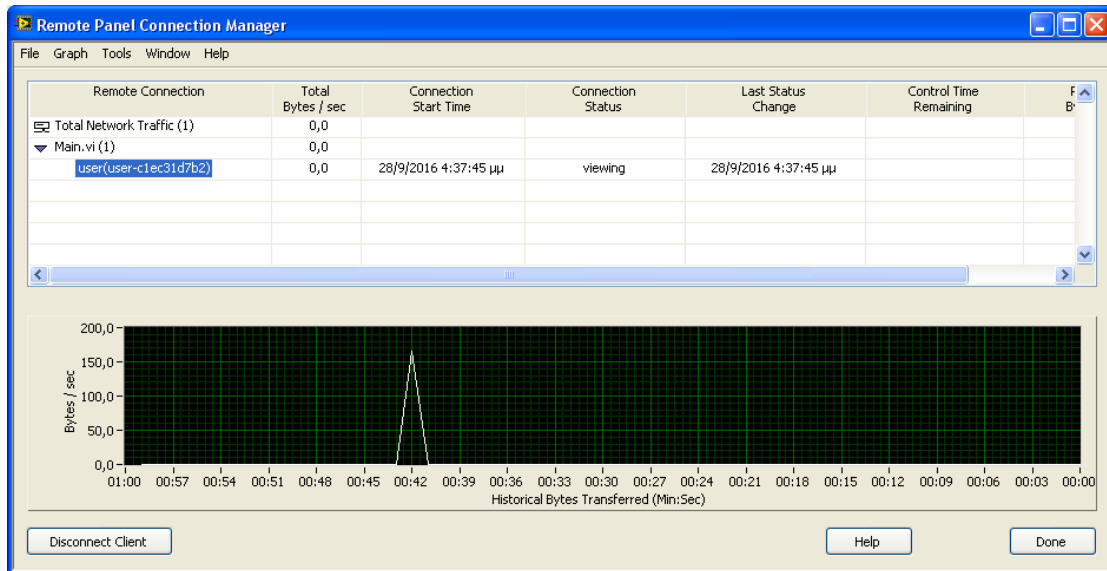
Εικόνα 5.10: Παράθυρο συμπλήρωσης στοιχείων Remote Server



Εικόνα 5.11: Front Panels (Client, Server)

5.5. Remote Panel Connection Manager

Στην καρτέλα Tools υπάρχει η επιλογή Remote Panel Connection Manager στην οποία φαίνονται στοιχεία για το ποιος είναι συνδεδεμένος μέσω Remote Panel, το πότε συνδέθηκε και αποσυνδέθηκε αλλά και πόσο όγκο δεδομένων έχει παραλάβει.

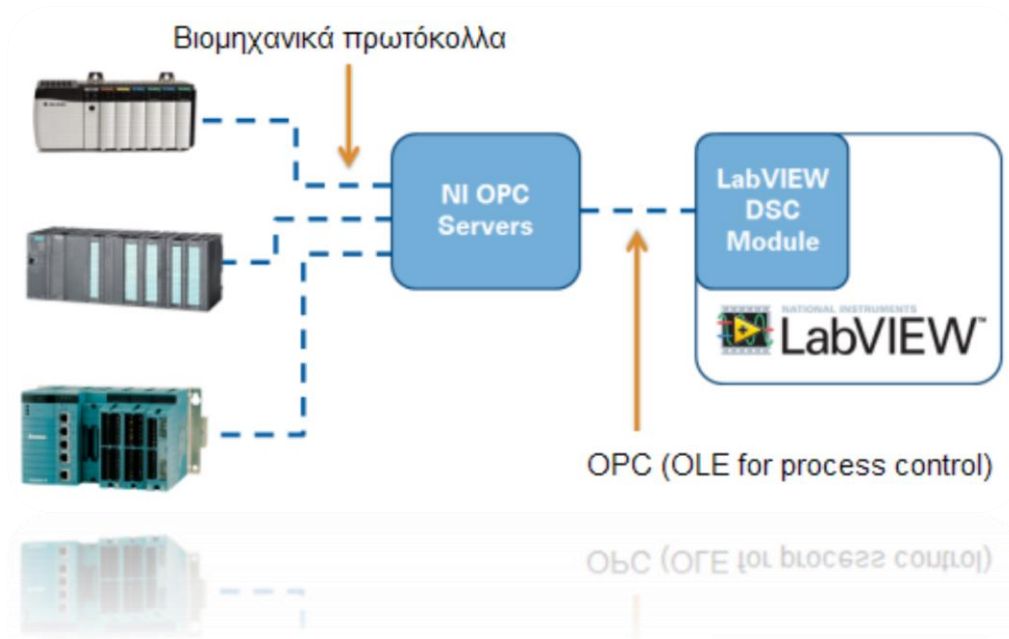


Εικόνα 5.12: Remote Panel Connection Manager

Επιλέγοντας κάποιον χρήστη υπάρχει η δυνατότητα αποσύνδεσής του πατώντας το κουμπί **Disconnect Client**.

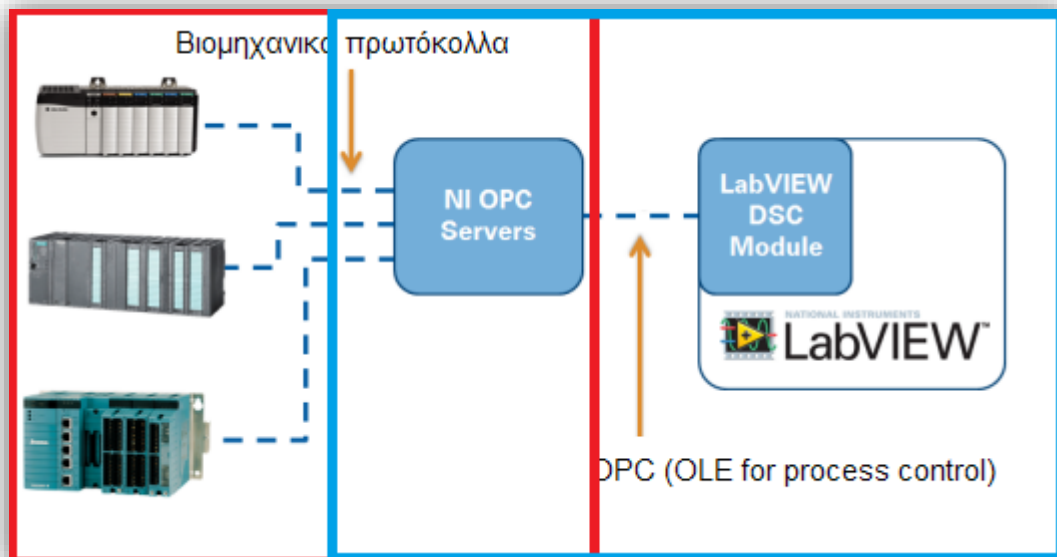
(Κενό φύλλο)

6. Σύνδεση PLC με LabView



Εικόνα 6.1: Σύνδεση PLC με LabVIEW

Η διαδικασία διασύνδεσης των δύο προγραμμάτων χωρίζεται σε δύο μέρη.



Εικόνα 6.2: Διαχωρισμός διασύνδεσης σε δύο μέρη

Το **πρώτο μέρος** περιέχει την σύνδεση ενός PLC με το πρόγραμμα της National Instruments OPC Servers.

Το **δεύτερο μέρος** περιέχει την σύνδεση του OPC Servers με το LabView. Για την σύνδεση αυτών των δύο απαραίτητη είναι η χρήση του Module DSC.

6.1. Μέρος 1^ο (Δημιουργία Εικονικού PLC και προετοιμασία αυτού για σύνδεση με OPC Servers)

Ξεκινώντας με την σύνδεση χρησιμοποιείται το πρόγραμμα Simatic Step 7 της Siemens.



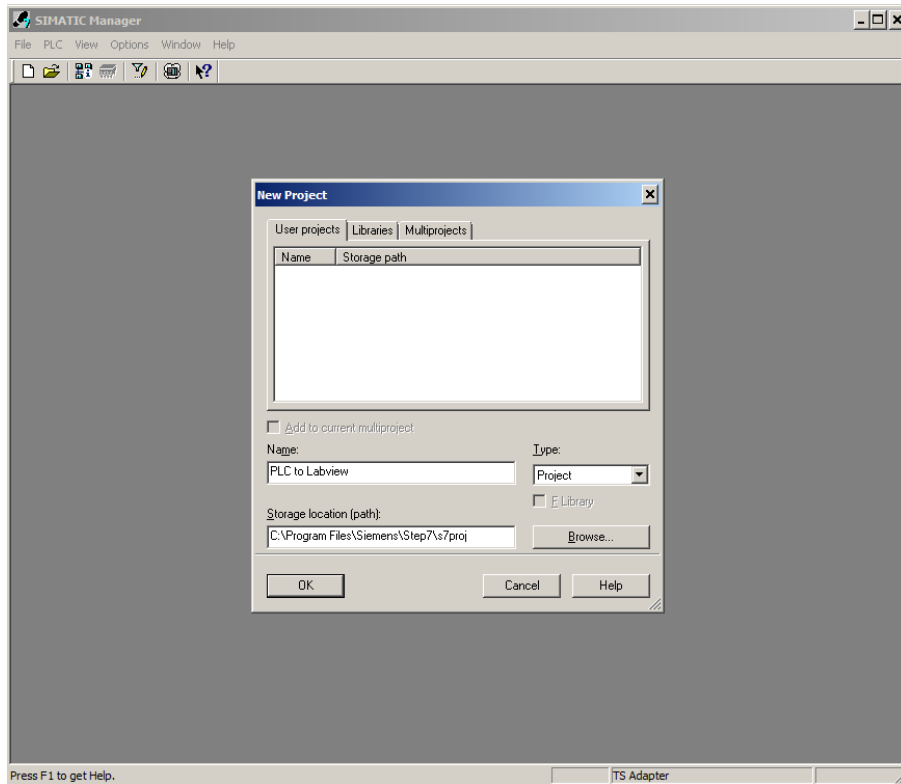
Εικόνα 6.3: Simatic Step 7

6.1.1. Λίγα λόγια για το Simatic

Το Simatic είναι ένα σύστημα αυτοματισμού που αναπτύχθηκε από την γερμανική Siemens. Το σύστημα αυτό ελέγχει μηχανές που χρησιμοποιούνται για την βιομηχανική παραγωγή. Το Simatic μπορεί να παρομοιαστεί με έναν ψηφιακό

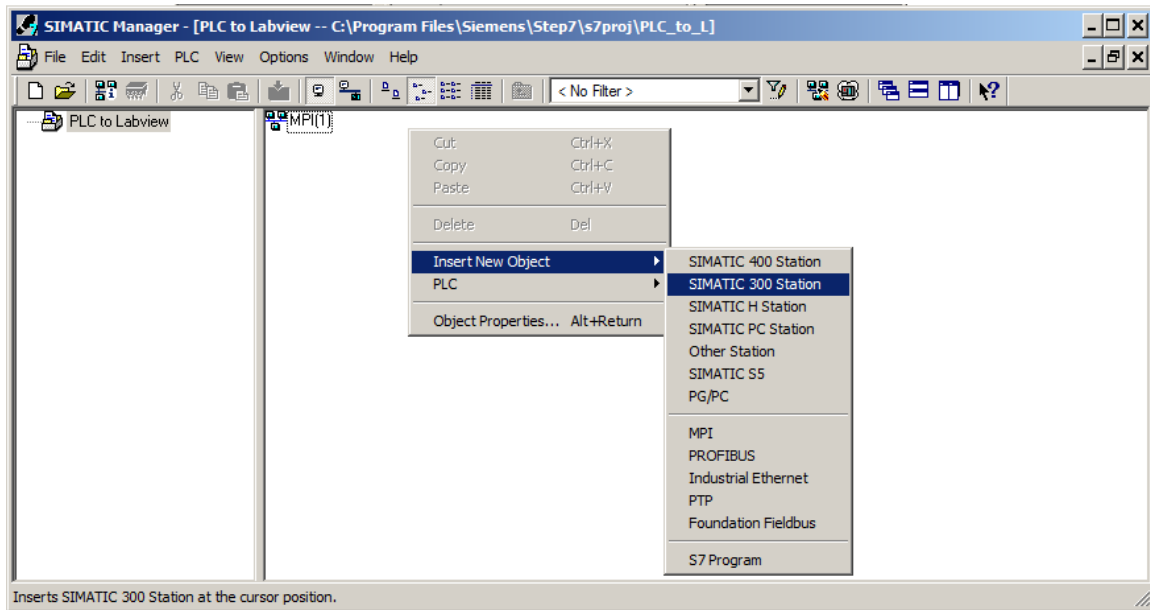
υπολογιστή που μπορεί να αποθηκεύσει και να εκτελέσει προγράμματα αλλά και να ελέγχει εισόδους και εξόδους που υπάρχουν σε αυτό.

Στο παράδειγμα χρησιμοποιείται το Simatic Manager για την **δημιουργία ενός PLC** και την δημιουργία εισόδων και εξόδων σε αυτό.



Εικόνα 6.4: Simatic Manager

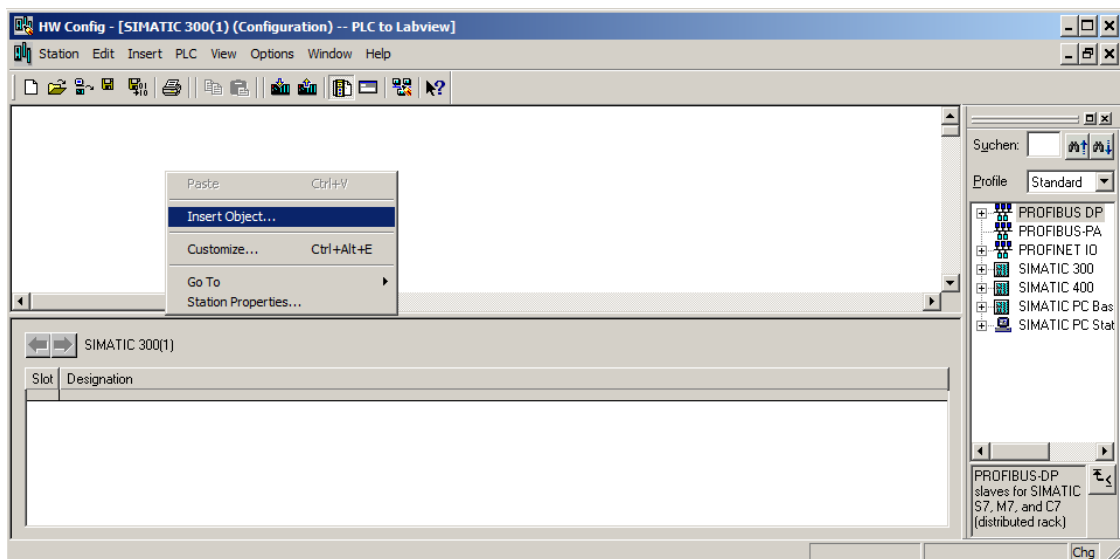
Δημιουργείται ένα νέο Project στο οποίο χρησιμοποιείται ένα s7-300 PLC.



Εικόνα 6.5: Εισαγωγή Simatic Station

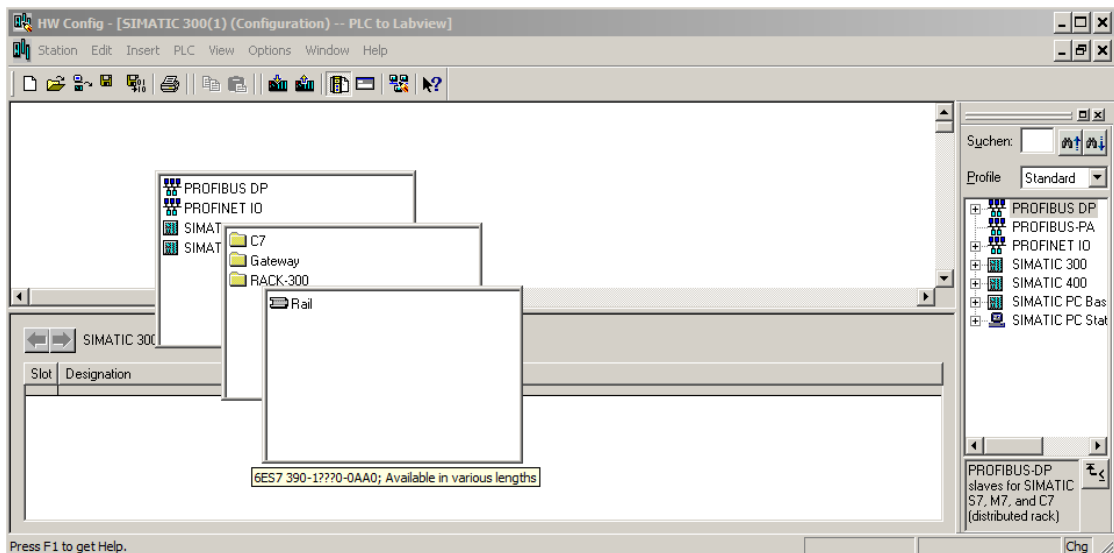
Κάνοντας διπλό Click σε αυτό, εισάγεται το Hardware .

Ανοίγει ένα καινούργιο παράθυρο στο οποίο κάνουμε εισαγωγή ενός νέου object.



Εικόνα 6.6: Εισαγωγή Hardware

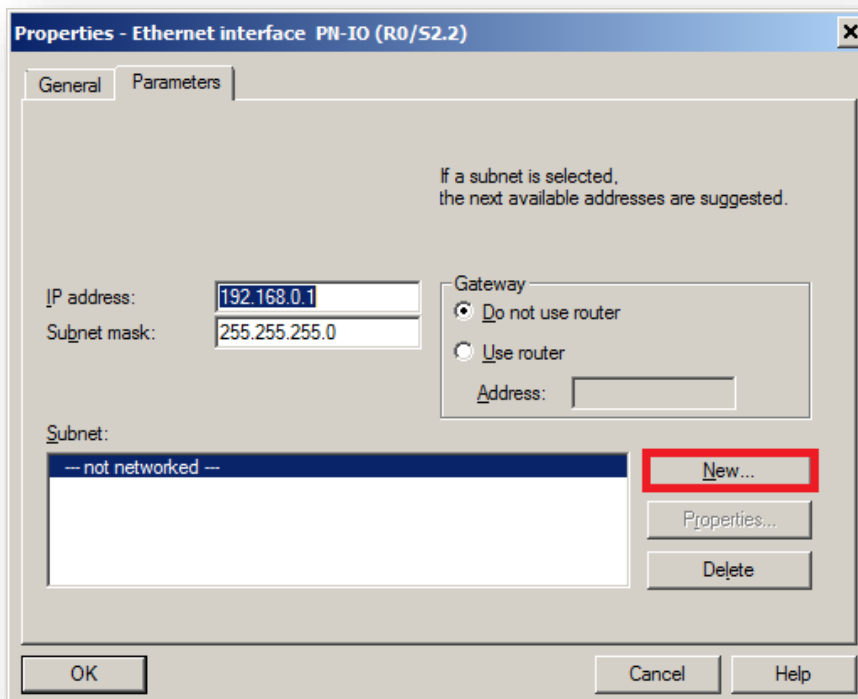
Εισάγεται ένα Rack-300 στο οποίο επιλέγεται τροφοδοτικό και CPU.



Εικόνα 6.7: Εισαγωγή Rack-300

Το τροφοδοτικό που επιλέγεται είναι το PS 307 5A.

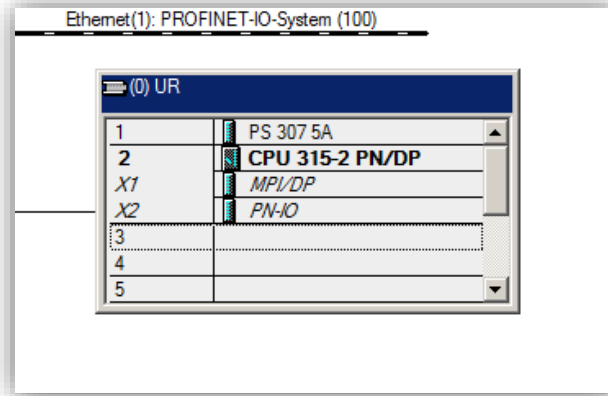
Επιλέγοντας CPU (315-2 PN/DP) μας ζητείται η δημιουργία ενός Ethernet Interface για την επικοινωνία με αυτό .



Εικόνα 6.8: Ethernet Interface

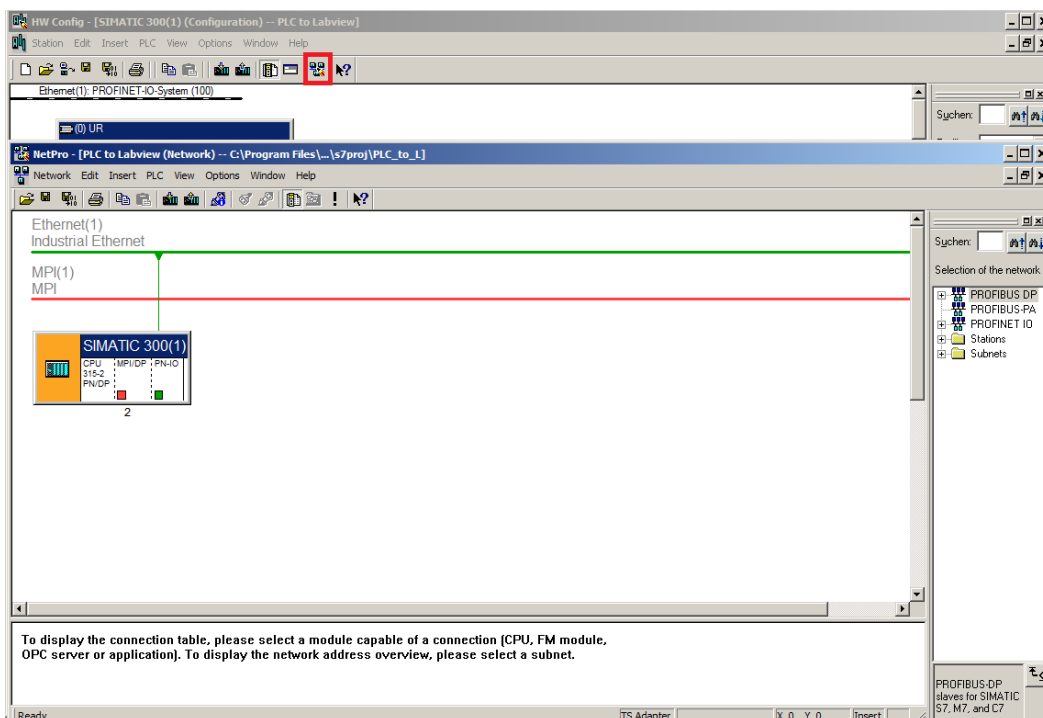
Δημιουργείται ένα νέο Ethernet Interface πατώντας το κουμπι “**New...**”

Μετά την εισαγωγή του CPU, το RACK παρουσιάζεται ως εξής:



Εικόνα 6.9: Rack μετά την εισαγωγή CPU

Κάνοντας Click στο **Configure Network** εμφανίζονται με γραφικό τρόπο ότι έχει δημιουργηθεί μέχρι στιγμής.

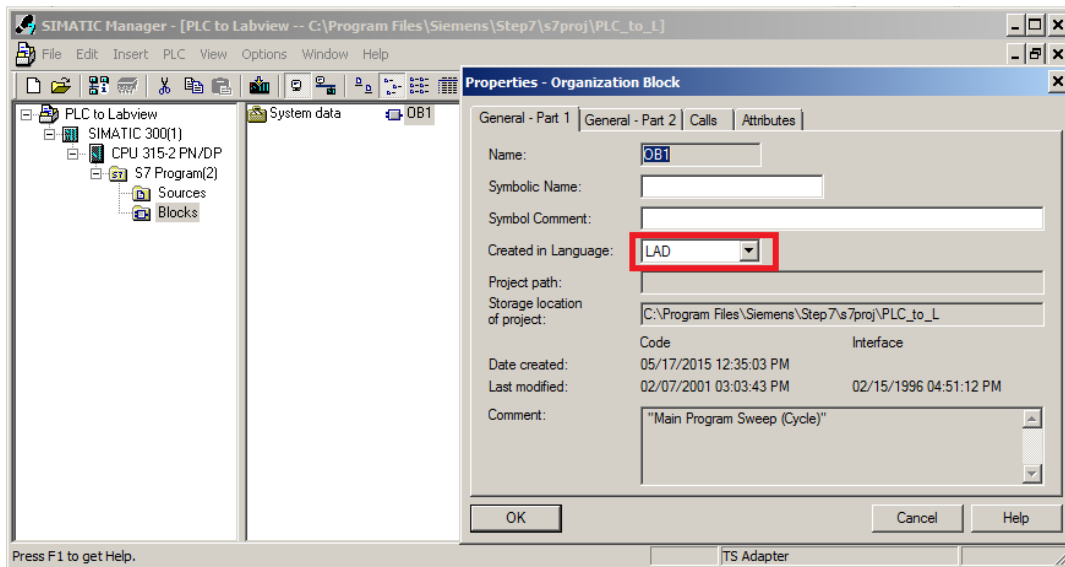


Εικόνα 6.10: Configure Network

Πριν κλείσει το παράθυρο παραμετροποίησης του Hardware πρέπει να γίνει **Save and Compile**.

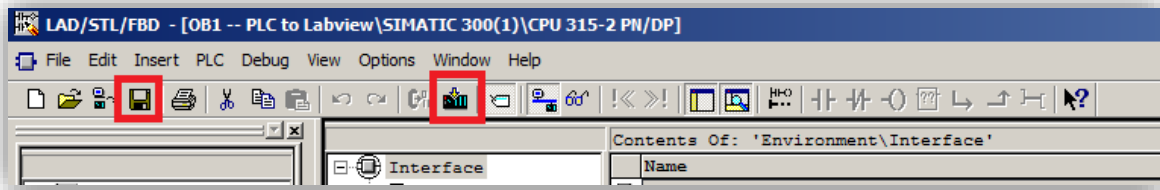
Στη συνέχεια, δημιουργούνται οι είσοδοι και οι έξοδοι του PLC.

Για την δημιουργία αυτών, γίνεται πλοήγηση στα Blocks και έπειτα διπλό κλικ στο **OB1**. Στο παράθυρο που εμφανίζεται γίνεται η επιλογή της γλώσσας.



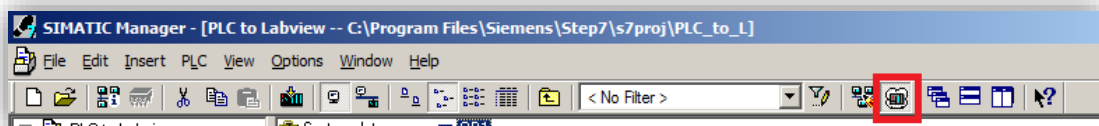
Εικόνα 6.11: Επιλογή Γλώσσας

Ύστερα ένα νέο παράθυρο ανοίγει στο οποίο σχεδιάζεται το πρόγραμμα σε γλώσσα **Ladder**.



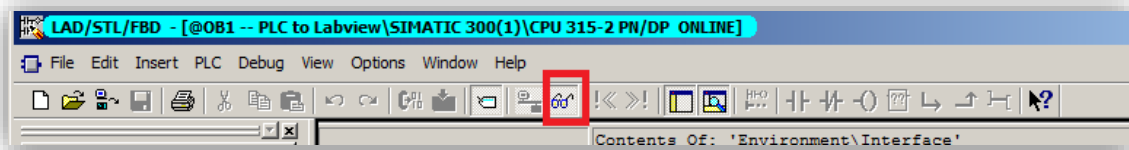
Εικόνα 6.14: Save και Download

Μετά από το αρχικό παράθυρο του Simatic επιλέγεται το **Simulation On**



Εικόνα 6.15: Simulation ON

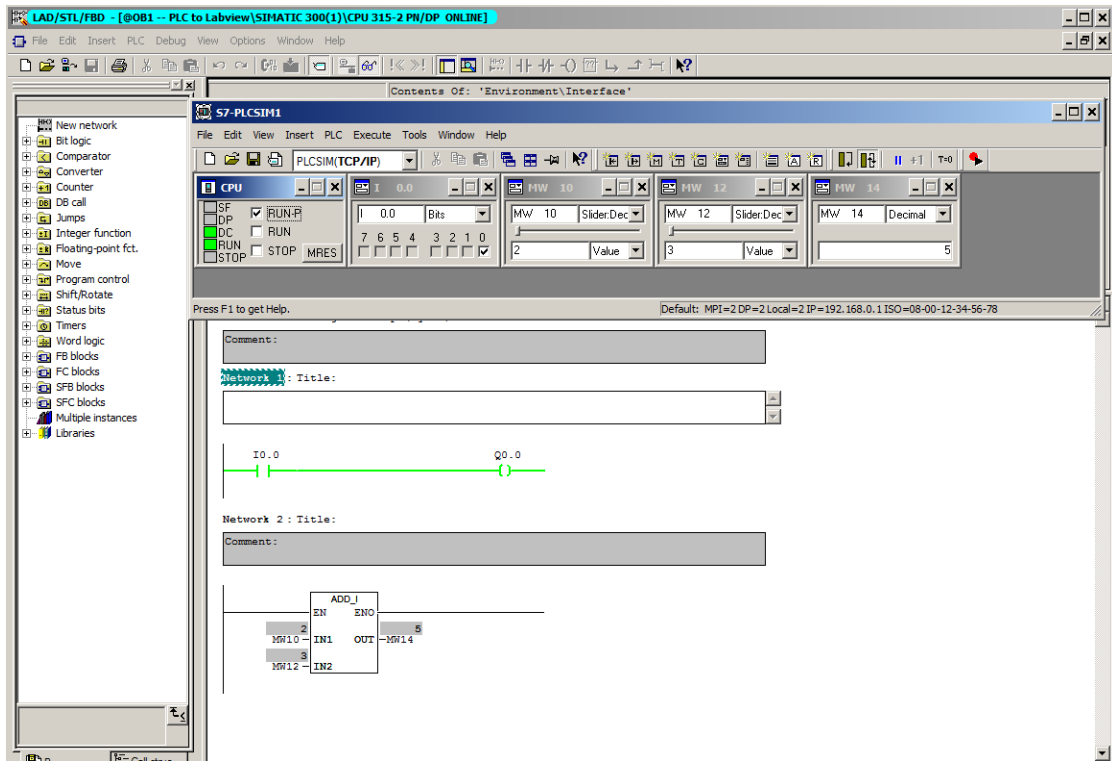
και από το παράθυρο της σχεδίασης του κώδικα επιλέγεται το Monitor On.



Εικόνα 6.16: Monitor ON

Στο παράθυρο του Simulation πρέπει να προστεθούν **όλες οι εισοδοι και έξοδοι** με τα ονόματα που δόθηκαν παραπάνω σχεδιάζοντας το πρόγραμμα (I0.0 , MW10 , MW12 , MW14).

Δίνοντας τιμές στις εισόδους και πατώντας το **RUN** στο simulation παρατηρούνται στο διάγραμμα τις γλώσσας Ladder οι αλλαγές που γίνονται.

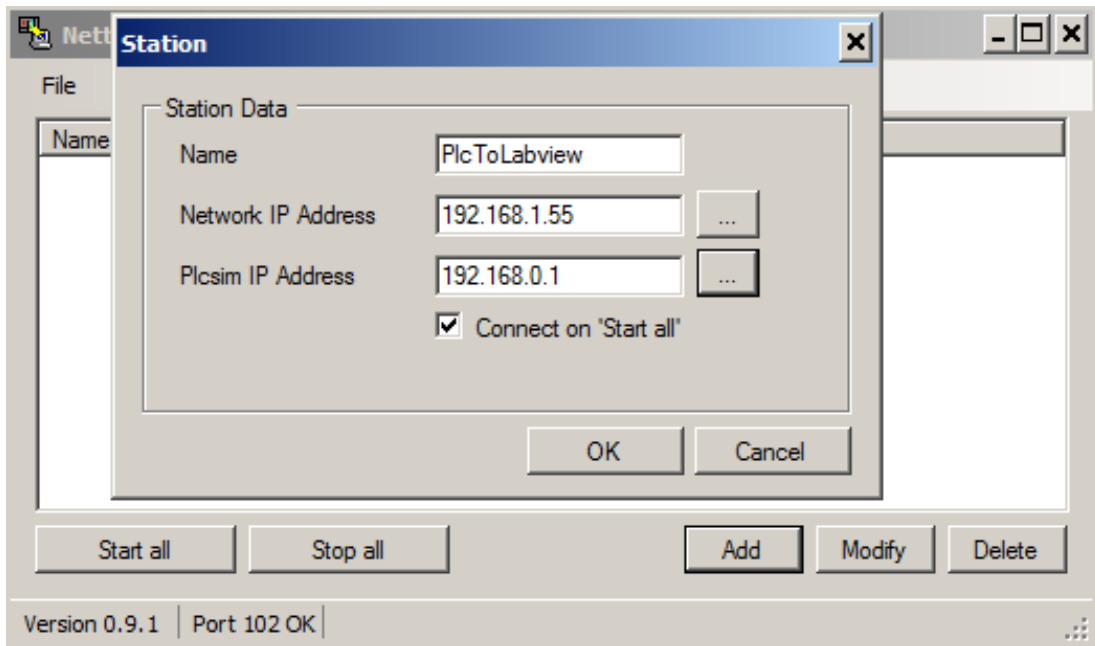


Εικόνα 6.17: Διάγραμμα Ladder και Simulation

Διαπιστώνεται ότι το PLC δουλεύει σύμφωνα με τον επιθυμητό τρόπο και ακολουθεί η σύνδεση αυτού με το **OPC Servers**.

Για την σύνδεση του εικονικού PLC με το OPC Server χρησιμοποιείται η επέκταση **NetToPLCSim**, η οποία επιτρέπει την προσομοίωση συστημάτων SCADA πάνω σε δίκτυο σε συνδυασμό με εικονικά PLC.

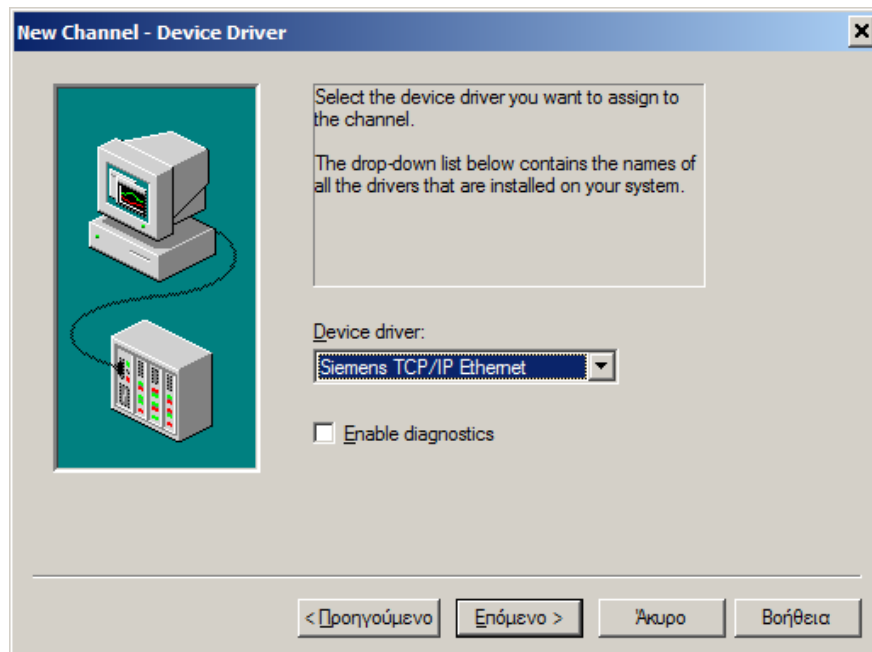
Ανοίγοντας το NetToPLCSim προστίθεται ένα νέο Station με την IP του δικτύου, αλλά και με την IP του προσαρμογέα Ethernet του εικονικού PLC.



Εικόνα 6.18: Ρυθμίσεις NetToPLCSim

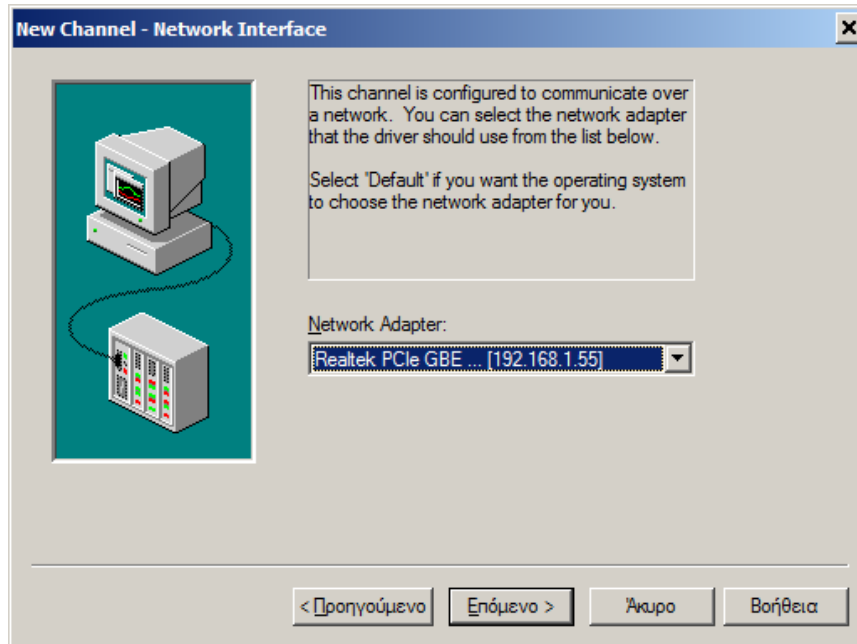
6.2. Μέρος 2^ο (Σύνδεση OPC Server με PLC)

Για την σύνδεση πρέπει να δημιουργηθεί ένα νέο κανάλι και να επιλεγεί ως Device driver: **Siemens TCP/IP Ethernet**



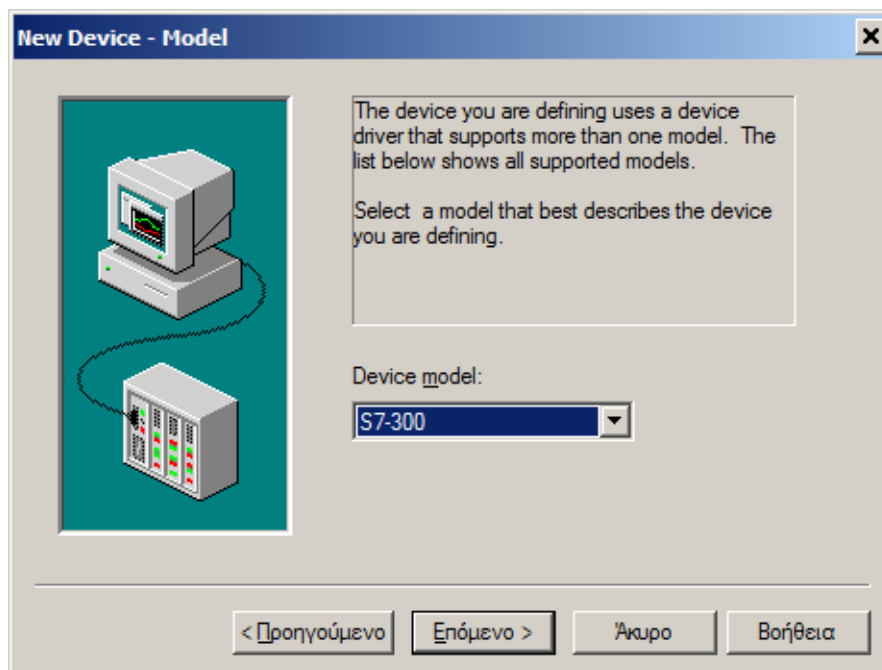
Εικόνα 6.19: Επιλογή Device Driver

Έπειτα, επιλέγεται το δίκτυο του υπολογιστή.



Εικόνα 6.20: Επιλογή Δικτύου

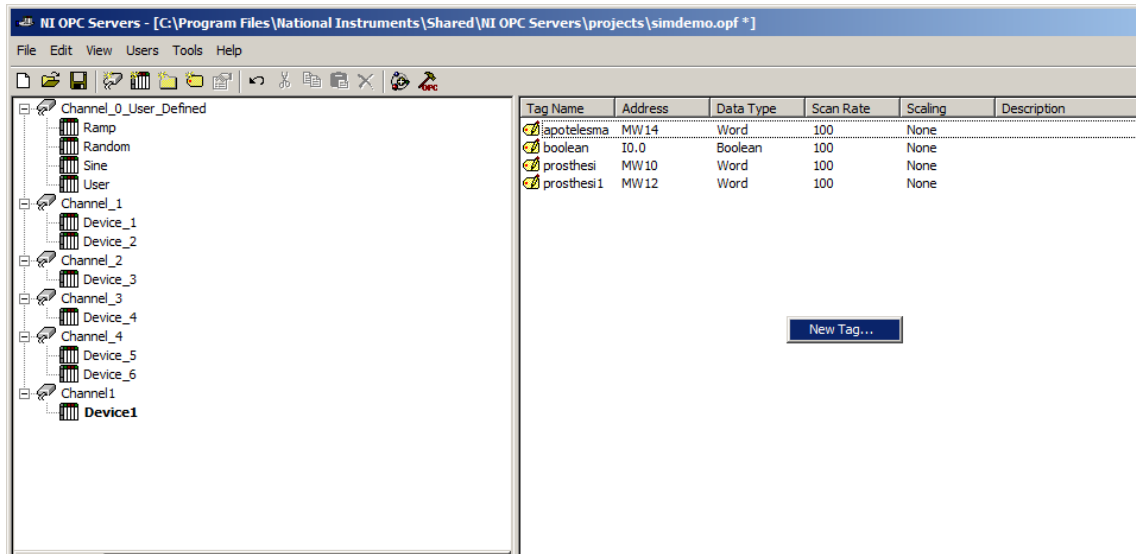
Με την ολοκλήρωση αυτού πρέπει να προστεθεί μια νέα συσκευή. Στο συγκεκριμένο παράδειγμα το **PLC S7-300**.



Εικόνα 6.21: Επιλογή συσκευής

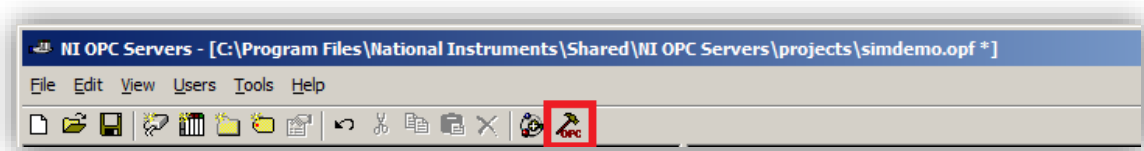
Εφόσον ζητηθεί διεύθυνση δικτύου, εισάγεται ξανά η διεύθυνση δικτύου του υπολογιστή.

Μετά την δημιουργία της συσκευής πρέπει να προστεθούν τυχόν εισοδοι και έξοδοι κάνοντας **δεξί click και Add new tag**.



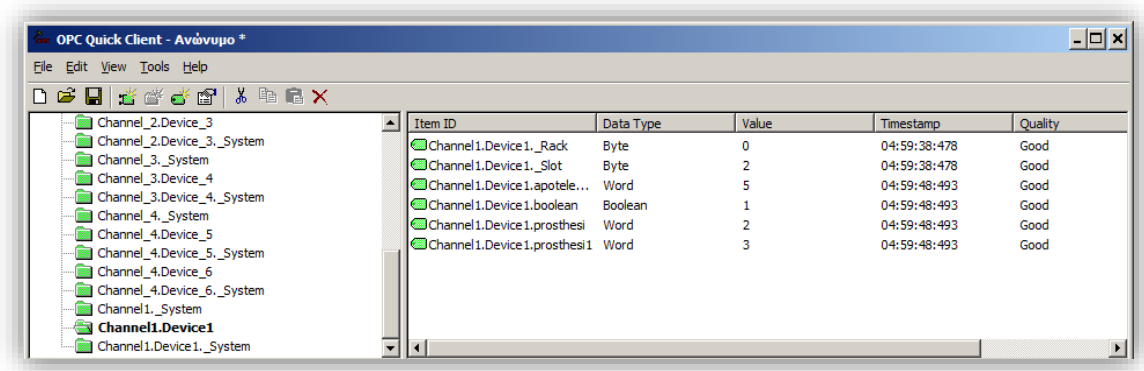
Εικόνα 6.22: Εισαγωγή εισόδων εξόδων

Αφού ολοκληρωθεί η προσθήκη των εισόδων εκτελείται το **Quick Client** από την γραμμή εργαλείων.



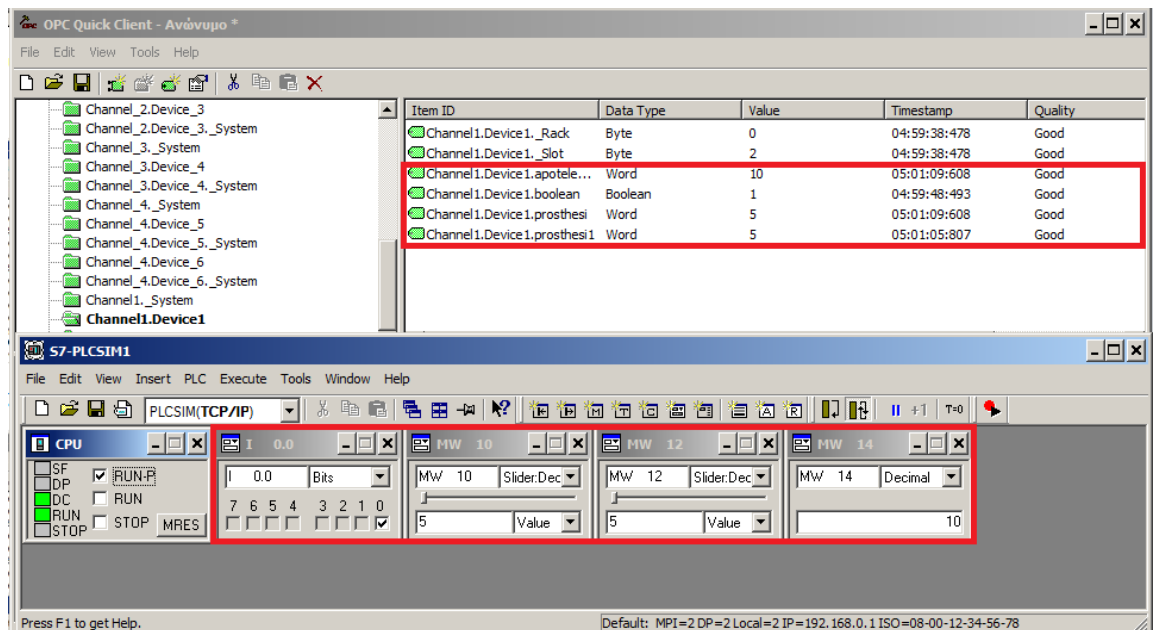
Εικόνα 6.23: Quick Client

Στο παράθυρο που εμφανίζεται γίνεται πλοήγηση στην συσκευή που δημιουργήθηκε, στην συγκεκριμένη περίπτωση η συσκευή **Device1**. Παρατηρείται ότι εμφανίζονται οι εισοδοι που έχουν φτιαχτεί με τις τιμές που έχουν δοθεί στο **εικονικό PLC**.



Εικόνα 6.24: Είσοδοι και έξοδοι του PLC

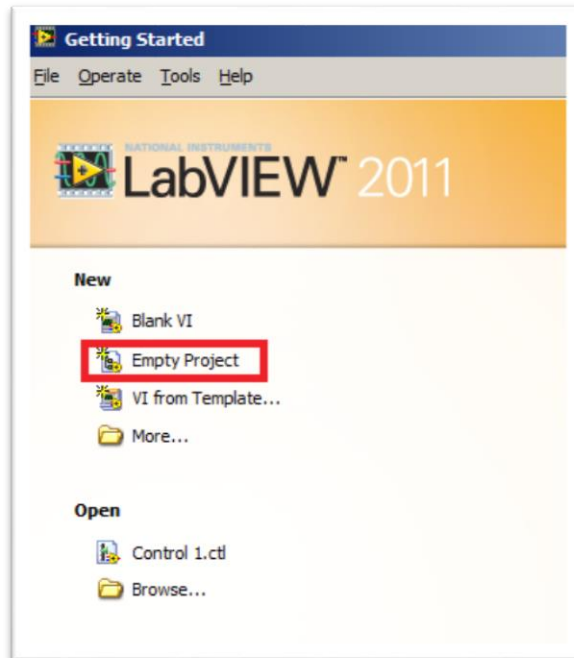
Αλλάζοντας τις τιμές στο PLC προκύπτουν και οι αντίστοιχες αλλαγές στον **OPC Server**.



Εικόνα 6.25: Simatic και OPC Server

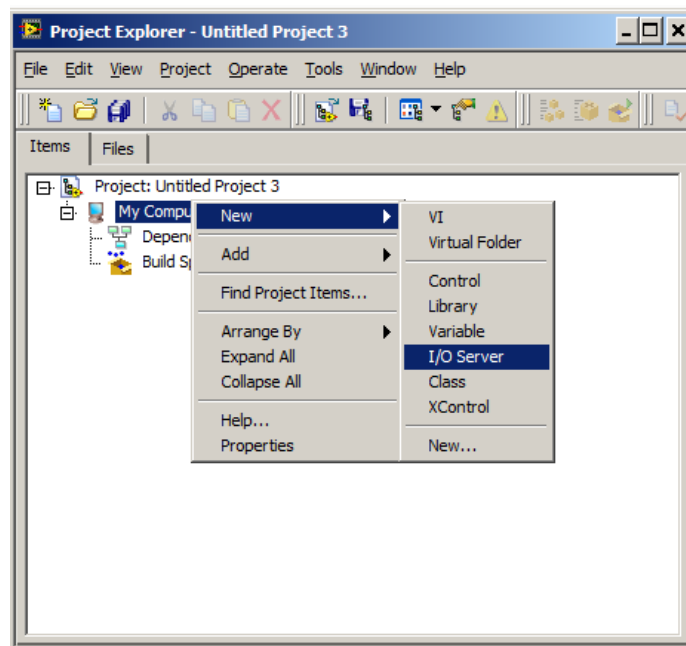
6.3. Μέρος 3^ο (Σύνδεση OPC Server με το LabView)

Γίνεται η δημιουργία ενός νέου **Empty Project**.



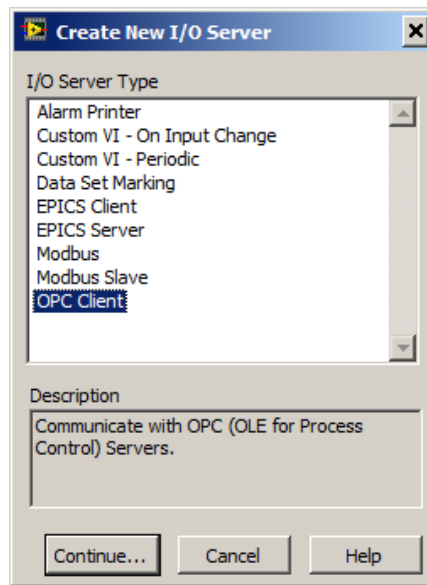
Εικόνα 6.26: Δημιουργία νέου Project

και στην συνέχεια ενός **νέου I/O Server**.



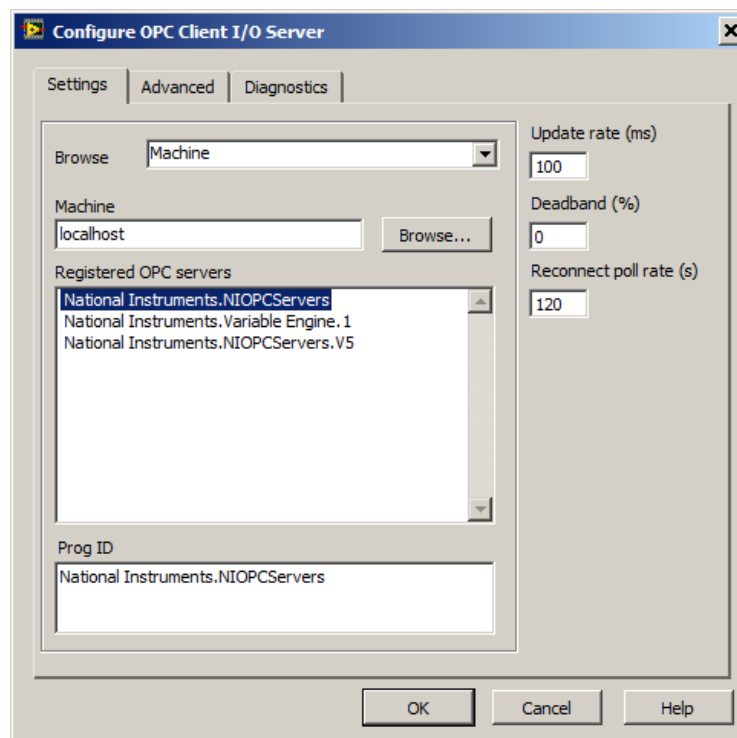
Εικόνα 6.27: Δημιουργία I/O Server

Στο παράθυρο που εμφανίζεται, επιλέγεται το **OPC Client**.



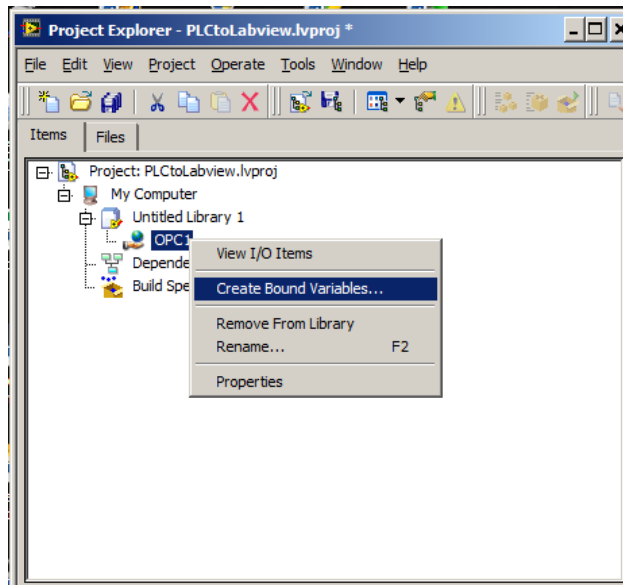
Εικόνα 6.28: Επιλογή OPC Client

Ορίζεται το **Update Rate σε 100ms**, τιμή ίδια με το PLC και στη συνέχεια, επιλέγεται ως **Registered OPC server** ο **National Instruments.NIOPServers**.



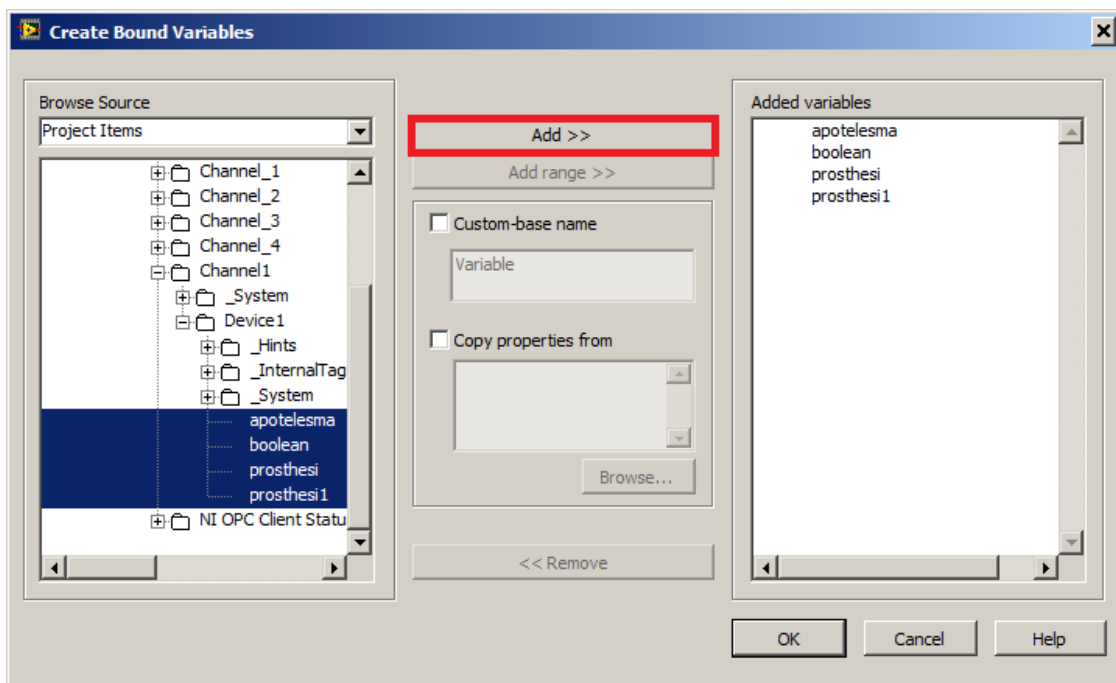
Εικόνα 6.29: Παραμετροποίηση OPC Client

Έπειτα, γίνεται η εισαγωγή των μεταβλητών κάνοντας **δεξί κλικ στο OPC1** και επιλέγοντας το **Create Bound Variables**.



Εικόνα 6.30: Δημιουργία μεταβλητών

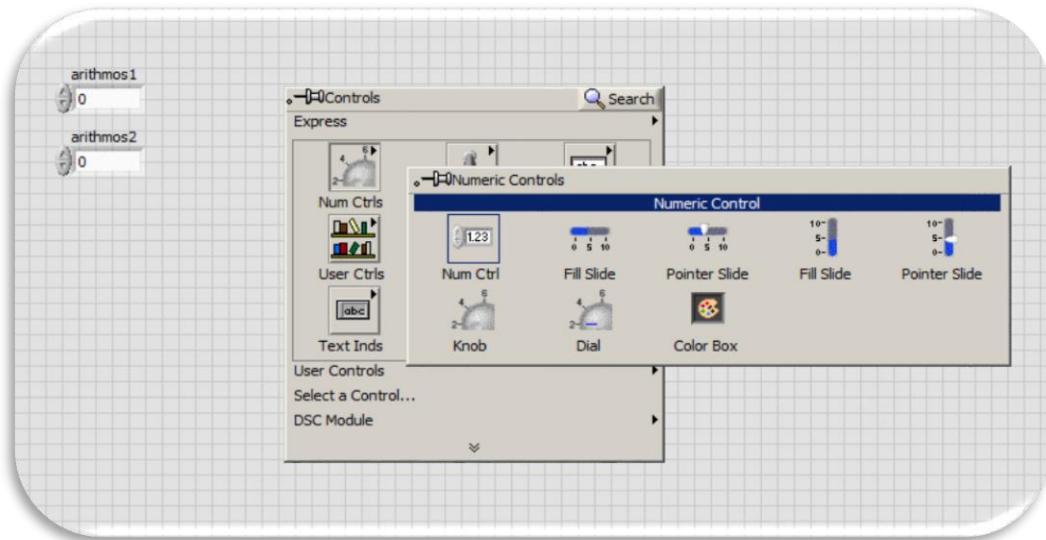
Στο παράθυρο που εμφανίζεται γίνεται πλοήγηση στα **Channel** και **Device** που έχουν δημιουργηθεί στο **OPC Server** και επιλέγονται **όλες** οι μεταβλητές.



Εικόνα 6.31: Επιλογή μεταβλητών

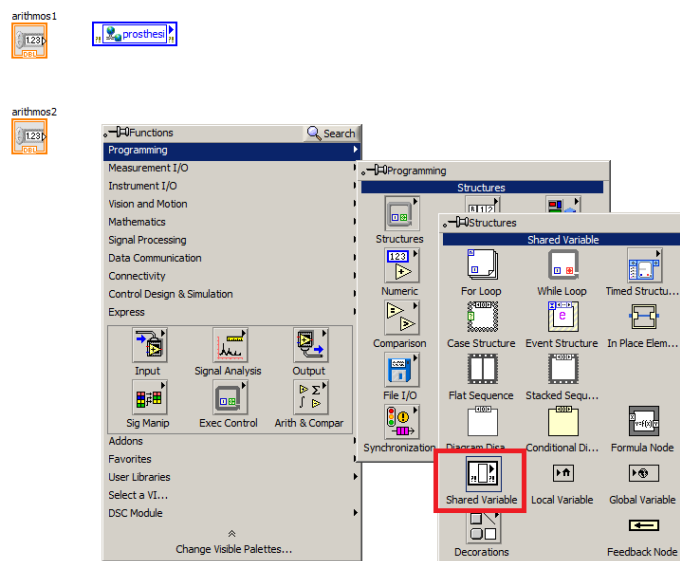
Εφόσον οι είσοδοι και οι έξοδοι μπορούν πλέον να χρησιμοποιηθούν από το LabView, δημιουργείται ένα **νέο VI**.

Στο **Front Panel** εισάγονται δύο **Num Ctrls** που ελέγχουν τις δυο εισόδους του PLC για την πρόσθεση.



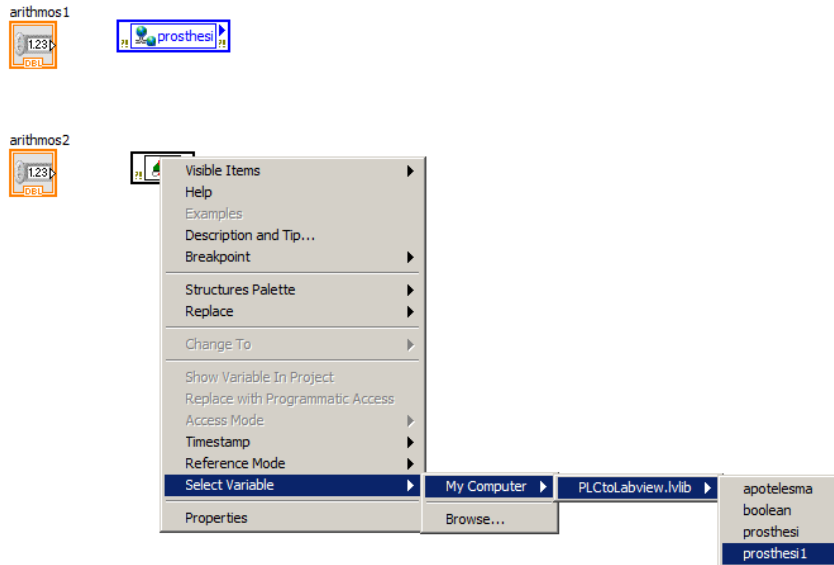
Εικόνα 6.32: Δημιουργία Front Panel

Στο **Block Diagram** εισάγονται δύο **Shared** μεταβλητές στις οποίες δίνονται οι δύο αριθμοί που θα προστεθούν.



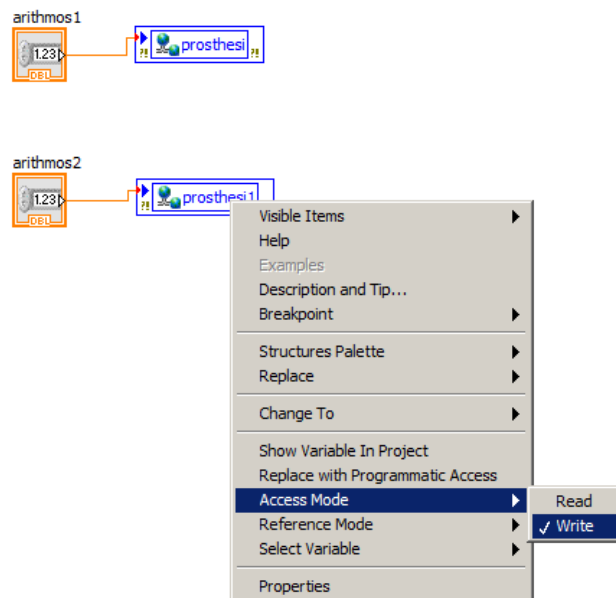
Εικόνα 6.33: Δημιουργία Block Diagram

Επιλέγεται η αντίστοιχη μεταβλητή κάνοντας **δεξί κλικ** στην κάθε μια **Shared Variable**.



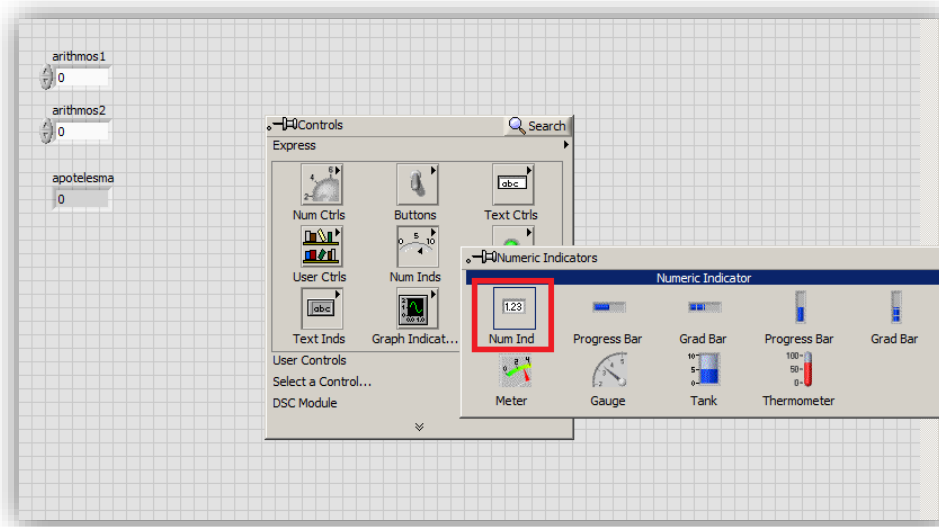
Εικόνα 6.34: Επιλογή Shared Variable

Συνδέεται η κάθε μια Shared Variable με το αντίστοιχο NumController.

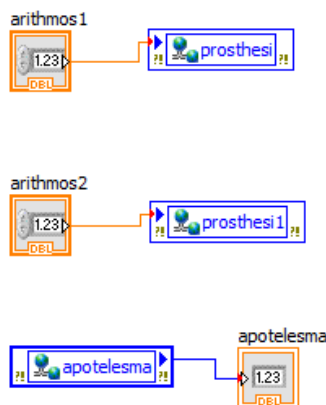


Εικόνα 6.35: Επιλογή Access Mode

Για το αποτέλεσμα της πρόσθεσης χρειάζεται ένα **Num Indicator**.

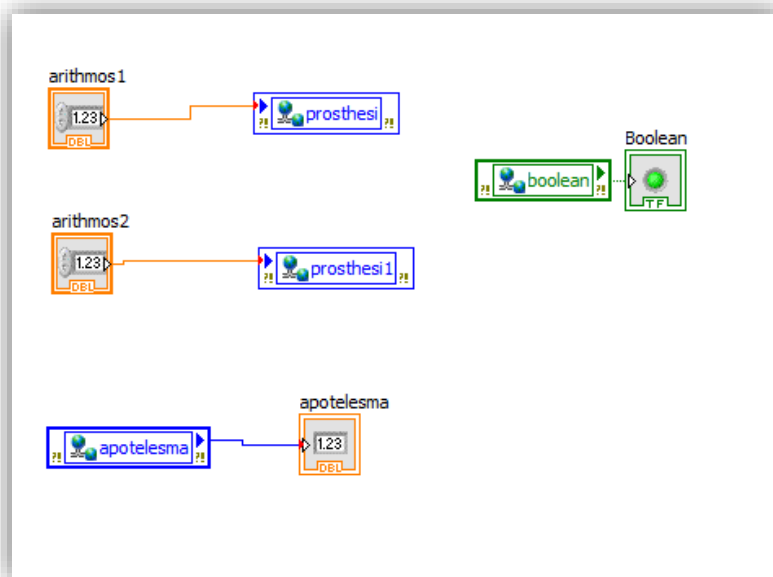
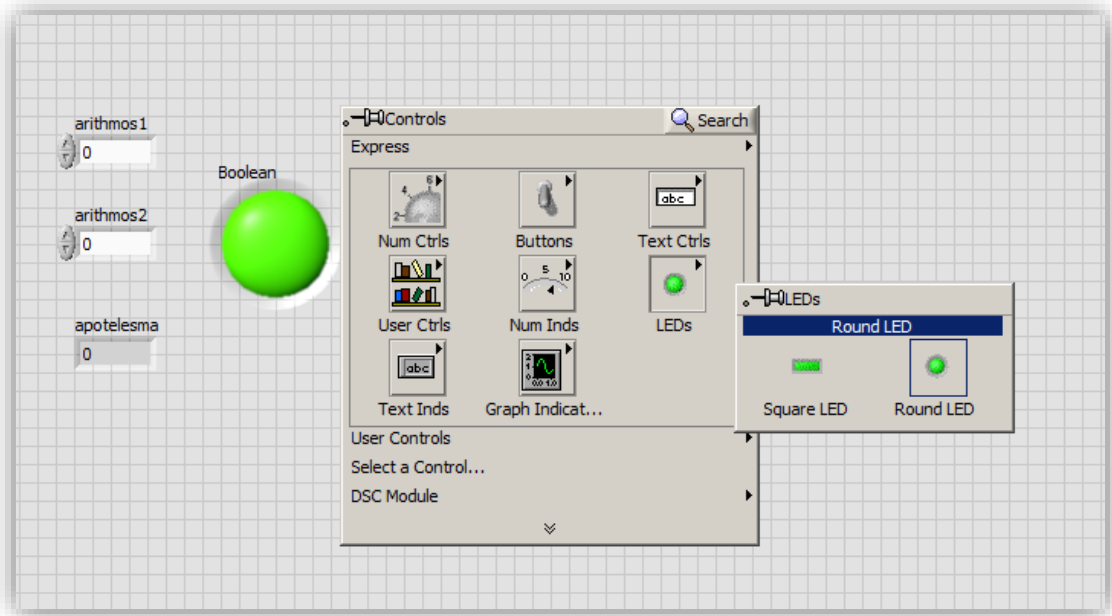


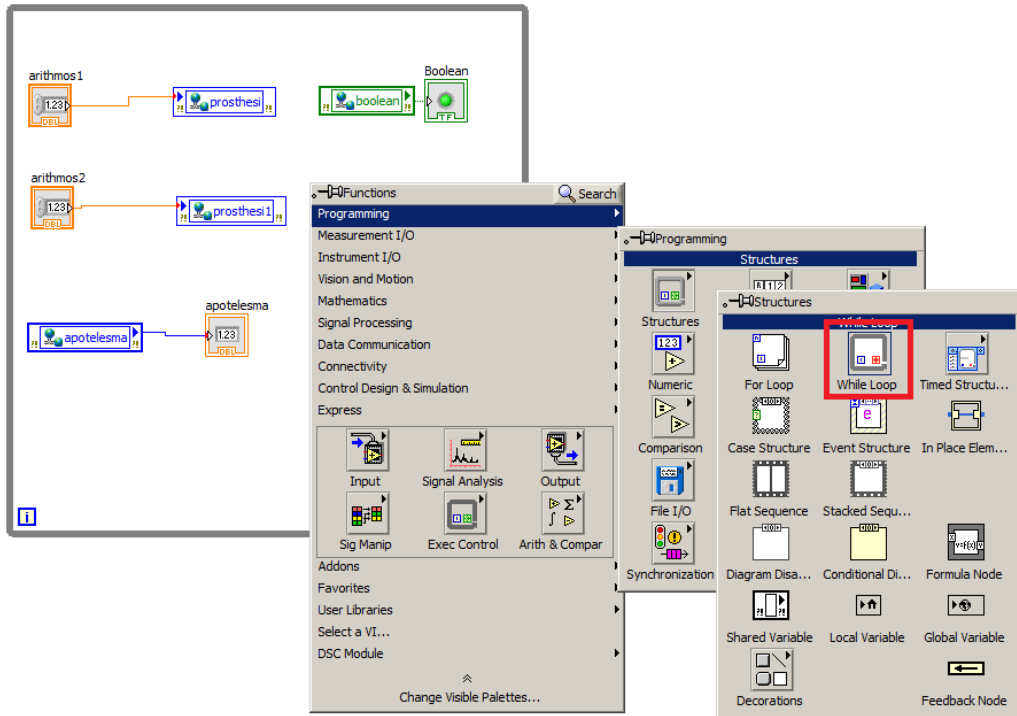
Δημιουργείται ακόμα μια **Shared Variable** (apotelesma), η οποία συνδέεται με το **Num Indicator** (apotelesma).



Επίσης, προστίθεται και **ένα Led** για τον έλεγχο της **Boolean** μεταβλητής.

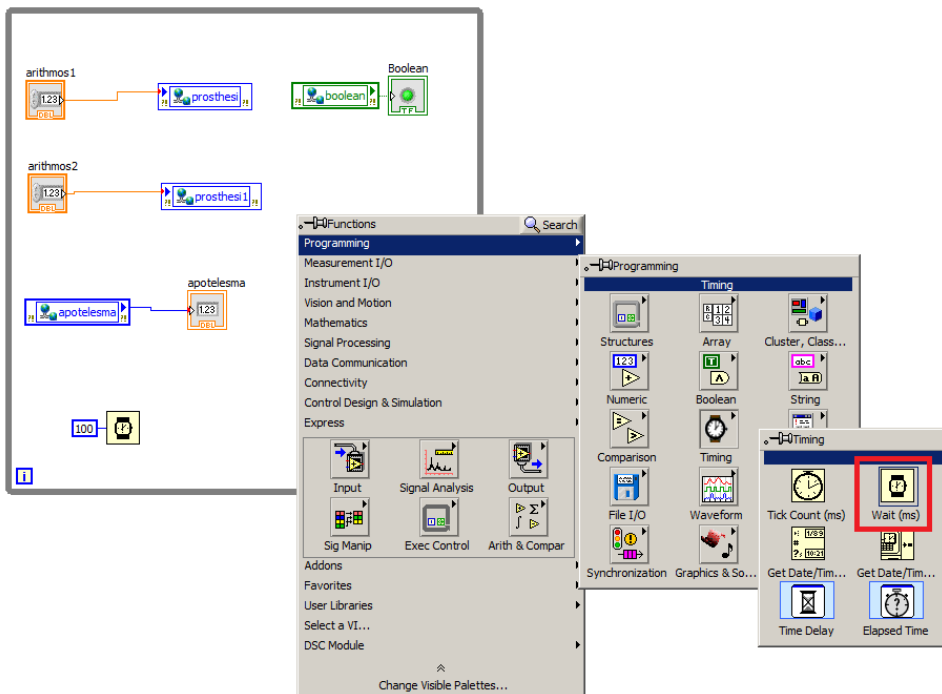
LabView and Vi Server Connectibility



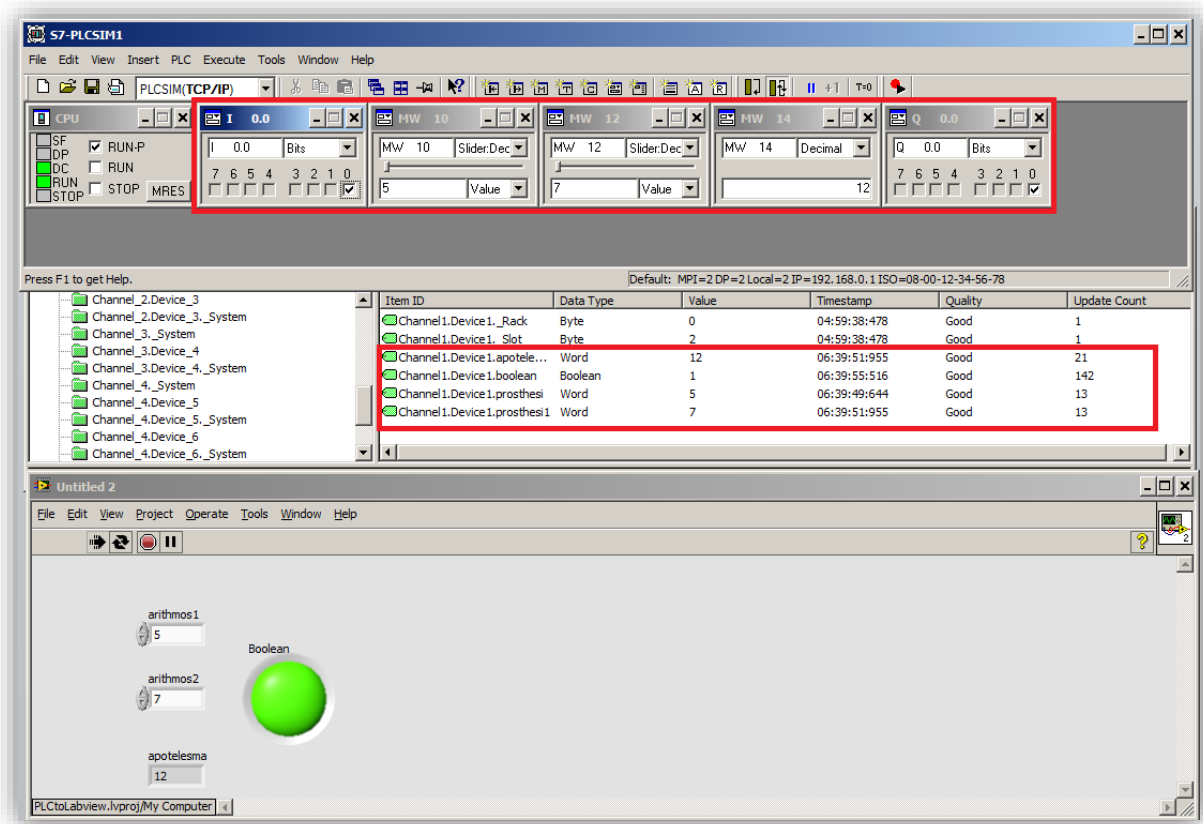


Εικόνα 6.36: Εισαγωγή While Loop

Προσθέτονται όλα μέσα σε ένα **While Loop** με **Wait Timer** το οποίο **ορίζεται στα 100ms**, τιμή στην οποία έχει ρυθμιστεί και το PLC.



Σε αυτό το σημείο παρατηρείται ότι αλλάζοντας τις τιμές του PLC συγχρονίζονται και οι αντίστοιχες του LabView. Το ίδιο συμβαίνει και αντίστροφα.



Εικόνα 6.37: Συγκεντρωτικά αποτελέσματα.

(Κενό φύλλο)

ΠΑΡΑΡΤΗΜΑ

Εργαστηριακό μέρος

Στην ενότητα αυτή παρουσιάζεται ένας σύντομος εργαστηριακός οδηγός προς τους φοιτητές με τα βήματα που πρέπει να ακολουθηθούν για τον απομακρυσμένο έλεγχο ενός VI.

1. Απομακρυσμένος έλεγχος ενός VI

- 1) Για να ανεβάσετε τα VIs στον Server πρέπει να χρησιμοποιήσετε έναν FTP Client. Μπορείτε να κατεβάσετε έναν FTP Client δωρεάν από τον παρακάτω σύνδεσμο:

FileZilla – FTP Client:

<https://filezilla-project.org/download.php?type=client>

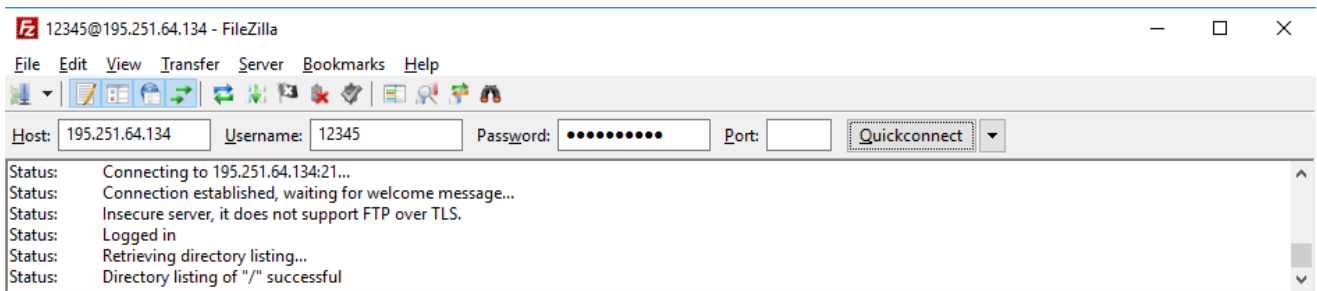
- 2) Αφού ληφθεί και εγκατασταθεί με επιτυχία, πρέπει συνδεθείτε στον Server με τα στοιχεία που σας έχουν δοθεί από το εργαστηριακό τμήμα στο οποίο είστε εγγεγραμμένοι.

Host: 195.251.64.134

Username: το όνομα χρήστη σας

Password: ο κωδικός πρόσβασής σας

Port: 21



Εικόνα Π1.1: Σύνδεση με FTP Client στον Server

3) Στη συνέχεια, ανεβάστε το VI της άσκησης με όνομα της μορφής:

Τμήμα_AM.vi (π.χ. deutera13-15_40736.vi)

Σημειώσεις:

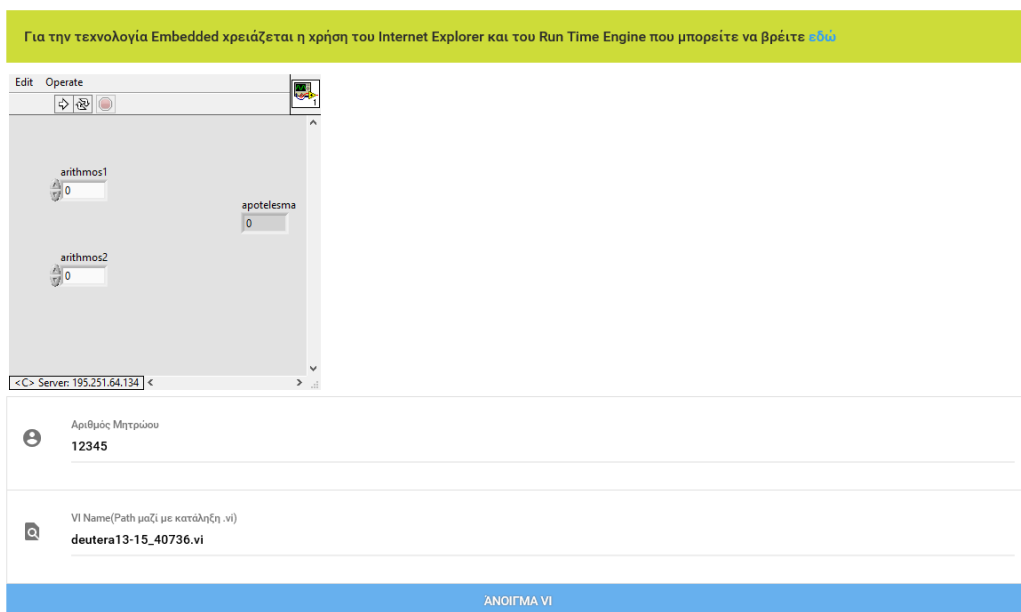
- Όπου **Τμήμα**, συμπληρώνετε το τμήμα όπου είστε εγγεγραμμένοι και την ώρα (π.χ. deutera13-15).
- Όπου **AM**, συμπληρώνετε τον αριθμό μητρώου σας, όπως ακριβώς στο παραπάνω παράδειγμα.
- Το αρχείο που ανεβάσατε πρέπει να έχει την μορφή **.vi**

ΠΡΟΣΟΧΗ!!!: ΜΗΝ ΧΡΗΣΙΜΟΠΟΙΕΙΤΕ ΕΛΛΗΝΙΚΟΥΣ ΧΑΡΑΚΤΗΡΕΣ ΣΤΑ ΟΝΟΜΑΤΑ ΤΩΝ VI.

4) Τέλος, αφού ανεβάσατε επιτυχώς το VI στον Server επισκεφτείτε τον παρακάτω σύνδεσμο με τη χρήση του Internet Explorer:

<http://virlab.teipir.gr/#!/load>

και συμπληρώστε τον **Αριθμό μητρώου** σας και το **όνομα του VI** που θέλετε να ελέγξετε.

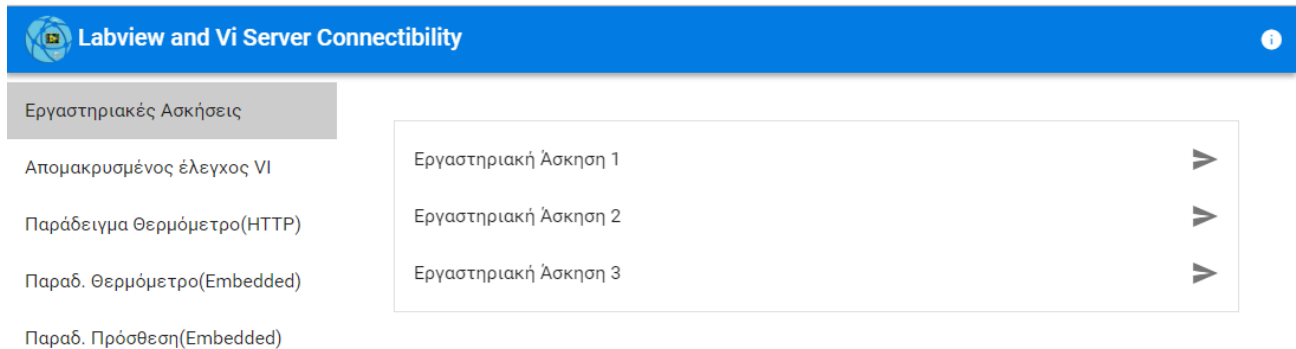


Εικόνα Π1.2: Φόρτωση του Vi από τον Server

2. Εργαστηριακή Άσκηση

Για να δείτε το VI στο οποίο έχετε την εργαστηριακή άσκησή σας, πλοηγηθείτε στον παρακάτω σύνδεσμο και επιλέξτε την εργαστηριακή άσκηση που σας έχει ανατεθεί.

<http://virlab.teipir.gr#!/ergastirio>



The screenshot shows a web interface for 'Labview and Vi Server Connectibility'. At the top is a blue header with the title and a small icon. Below the header is a navigation menu with the following items: 'Εργαστηριακές Ασκήσεις' (highlighted), 'Απομακρυσμένος έλεγχος VI', 'Παράδειγμα Θερμόμετρο(HTTP)', 'Παραδ. Θερμόμετρο(Embedded)', and 'Παραδ. Πρόσθεση(Embedded)'. To the right of the menu is a list of three laboratory exercises, each with a right-pointing arrow: 'Εργαστηριακή Άσκηση 1', 'Εργαστηριακή Άσκηση 2', and 'Εργαστηριακή Άσκηση 3'.

Εικόνα Π1.3: Εργαστηριακές Ασκήσεις

(Κενό φύλλο)

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] www.ni.com - *National Instruments: Test, Measurement, and Embedded Systems*
- [2] <http://www.ni.com/white-paper/2983/en/>
- [3] http://zone.ni.com/reference/en-XX/help/371361J-01/lvhowto/vi_server/
- [4] www.tetradio.gr - *Αυτοματισμοί και PLC*
- [5] http://courseware.mech.ntua.gr/ml23194/extras/PLC_Lesson1.pdf
- [6] <http://www.slideshare.net/ssuser9218dc/aftomat-kai-sae-enotita-2>
- [7] <http://repository.edulll.gr/edulll/retrieve/4971/1410.pdf>
- [8] www.wikipedia.com
- [9] www.google.com
- [10] <http://slideplayer.gr/slide/2023476/>
- [11] <http://sine.ni.com/nips/cds/view/p/lang/en/nid/209059> - *NI OPC Servers*
- [12] <http://www.pererikstrandberg.se/>