



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ ΚΑΙ ΔΙΟΙΚΗΣΗΣ

**ΤΗΛΕΜΕΤΡΙΑ ΚΑΙ ΑΠΟΜΑΚΡΥΣΜΕΝΟΣ ΕΛΕΓΧΟΣ ΣΕ
ΣΥΣΤΗΜΑΤΑ ΑΥΤΟΜΑΤΟΥ ΕΛΕΓΧΟΥ ΜΕ ΧΡΗΣΗ
ΥΠΗΡΕΣΙΩΝ ΔΙΑΔΙΚΤΥΟΥ**

Web-Based OPC Client

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
ΤΟΥ
ΚΑΤΣΙΓΑΡΑΚΗ ΚΥΡΙΑΚΟΥ**

Χανιά, Φεβρουάριος 2010

Πρόλογος

Η εργασία εκπονήθηκε στο Εργαστήριο Ελέγχου Βιομηχανικών Συστημάτων του τμήματος Μηχανικών Παραγωγής και Διοίκησης του Πολυτεχνείου Κρήτης. Το κύριο αντικείμενο που πραγματεύεται σχετίζεται με την διαδικασία ανάπτυξης εφαρμογών για την εξ' αποστάσεως διαχείριση συστημάτων αυτομάτου ελέγχου με την χρήση υπηρεσιών του διαδικτύου. Εξετάζει την προδιαγραφή Data Access του OPC (OLE for Process Control) που ορίζει τα αντικείμενα και τις διασυνδέσεις της τεχνολογίας COM που χρησιμοποιούνται για την ανταλλαγή πληροφοριών ανάμεσα σε πελάτες OPC και εξυπηρετητές τύπου Data Access.

Κύριος στόχος είναι η ανάλυση, η σχεδίαση και η ανάπτυξη ενός συστήματος το οποίο σαν βασική διασύνδεση χρήστη χρησιμοποιεί την διαδικτυακή εφαρμογή με την ονομασία OPC to Web, η οποία μέσω της τεχνολογίας automation, υποστηρίζει τα απαραίτητα αντικείμενα και τις διασυνδέσεις του Data Access V2.02 του OPC Foundation.

Σε αυτό το σημείο θα ήθελα να ευχαριστήσω όλους όσους με βοήθησαν με οποιονδήποτε τρόπο καθ' όλη την διάρκεια εκπόνησης της παρούσας διπλωματικής εργασίας. Ιδιαίτερα ευχαριστώ:

Τον επιβλέποντα καθηγητή μου κ. Αναστάσιο Πουλιέζο που μου έδωσε την ευκαιρία να ασχοληθώ με το συγκεκριμένο αντικείμενο και για την απόλυτη εμπιστοσύνη που μου έδειξε ως προς την χρήση του εξοπλισμού του εργαστηρίου που διευθύνει.

Τον κ. Νεκτάριο Αρναουτάκη για την παραχώρηση του απαραίτητου εργαστηριακού εξοπλισμού και την πολύτιμη βοήθεια του, στην πρώτη επαφή που είχα με την εγκατάσταση KNX/EIB του εργαστηρίου.

Τα μέλη της εξεταστικής επιτροπής, τον κ. Δημήτριο Ρόβα και τον κ. Ιωάννη Μαρινάκη οι οποίοι δέχτηκαν με πολύ ευχαρίστηση να διορθώσουν την παρούσα εργασία και να μου δώσουν πολύτιμες συμβουλές για την περαιτέρω βελτίωση της.

Τους λίγους φίλους μου, που με υποστήριξαν σε όλες τις δυσκολίες που συνάντησα κατά την υλοποίηση του συστήματος.

Τέλος ευχαριστώ θερμά την οικογένεια μου για την πολύπλευρη συμπαράσταση της όλα αυτά τα χρόνια των σπουδών μου.

Κατσιγάρκης Κυριάκος
Χανιά, Φεβρουάριος 2010

Επιβλέπων Καθηγητής

Αναστάσιος Πουλιέζος

Εξεταστική Επιτροπή

Αναστάσιος Πουλιέζος

Δημήτριος Ρόβας

Ιωάννης Μαρινάκης

Περιεχόμενα

Περιεχόμενα	5
1 Εισαγωγή	9
1.1 Συστήματα SCADA.....	9
1.2 Βασικά μέρη ενός SCADA.....	9
2 Λογισμικό ανοιχτής αρχιτεκτονικής	11
2.1 Επικοινωνία μεταξύ διεργασιών στο Win32	11
2.1.1 Νήματα και Διεργασίες.....	11
2.1.2 Μέθοδοι διαδιεργασιακής επικοινωνίας (InterProcess Comm.).....	13
2.1.3 DDE (Dynamic Data Exchange)	13
2.1.4 OLE (Object Linking and Embedding).....	14
2.2 Τεχνολογία COM (Component Object Model).....	14
2.2.1 Εισαγωγή.....	14
2.2.2 Χαρακτηριστικά COM	15
2.2.3 Αρχιτεκτονική COM.....	17
2.2.4 Είδη εξυπηρετητών COM.....	18
2.2.5 Επικοινωνία πελάτη με εξυπηρετητή out-of-process	18
2.2.6 Το εργαλείο Regsvr32	19
2.3 Τεχνολογία Automation	20
2.3.1 Δημιουργία και διαχείριση αντικειμένων	20
2.3.2 Κλίση μεθόδων	22
2.3.3 Χρήση ιδιοτήτων.....	22
2.3.4 Χρήση συλλογών.....	22
2.4 OLE for Process Control.....	23
2.4.1 Γενικά	23
2.4.2 OPC Data Access	24
2.4.3 Προγραμματισμός πελάτη OPC.....	27

3	Ανάλυση, Σχεδίαση & Ανάπτυξη Συστήματος	33
3.1	Εισαγωγή.....	33
3.2	Τεχνικό Υπόβαθρο Συστήματος	35
3.2.1	ODBC (Open Database Connectivity).....	35
3.2.2	SQL (Structured Query Language)	36
3.2.3	ASP (Active Server Pages)	37
3.2.4	AJAX (Asynchronous Javascript And XML).....	41
3.2.5	WMI (Windows Management Instrumentation).....	45
3.2.6	WSH (Windows Script Host)	48
3.3	Αρχιτεκτονική υλικού	48
3.4	Αρχιτεκτονική λογισμικού.....	51
3.4.1	OPC to Web.....	53
3.4.2	OPC to DB.....	63
3.4.3	OPC Real Time Control.....	64
3.4.4	Ρυθμίσεις αυτοματοποίησης.....	65
3.4.5	Κοινή χρήση μεταβλητών μεταξύ των εφαρμογών	66
4	Μελέτη Περίπτωσης.....	67
4.1	Το πρότυπο KNX/EIB	67
4.1.1	Τρόποι μετάδοσης	67
4.1.2	Μετάδοση μέσω συνεστραμμένου ζεύγους	68
4.1.3	Ασύρματη Μετάδοση	70
4.1.4	Μετάδοση με χρήση γραμμής ισχύος	70
4.1.5	Μετάδοση μέσω Ethernet	71
4.1.6	Τοπολογία και οργάνωση διαύλου	71
4.1.7	Μετάδοση πληροφορίας – τηλεγραφήματα.....	72
4.1.8	Συνδρομητές διαύλου	73
4.2	Εγκατάσταση KNX/EIB στο εργ. Ελέγχου Βιομηχανικών Συστημάτων.....	74
4.2.1	Αισθητήρες & Ενεργοποιητές.....	76
4.2.2	Συσκευές ελέγχου των ενεργοποιητών	79
4.2.3	Άλλες συσκευές	83

4.3	Βελτιώσεις συστήματος	85
4.3.1	Επαναπρογραμματισμός συσκευών στην εγκατάσταση	85
4.3.2	Ανάπτυξη εφαρμογής sHumidity	88
4.3.3	Αλγόριθμος προβλεπτικού ελέγχου	89
4.4	Εφαρμογές OPC Real Time Control	90
4.4.1	Έλεγχος οπτικής άνεσης	90
4.4.2	Έλεγχος θερμικής άνεσης	96
5	Συμπεράσματα & Μελλοντικές Κατευθύνσεις	100
5.1	Συμπεράσματα	100
5.2	Προτάσεις για μελλοντική ανάπτυξη	101
5.2.1	Βελτιώσεις λογισμικού	101
5.2.2	Βελτιώσεις εγκατάστασης	102
	Βιβλιογραφία	103
	Παράρτημα Α	105
A.1	Παρουσίαση εφαρμογής OPC to Web	105
A.2	Παράμετροι εφαρμογής	118
A.3	Κώδικας ελεγκτή οπτικής άνεσης	121
A.4	Κώδικας ελεγκτή θερμικής άνεσης	122
A.5	Αρχικοποίηση του μητρώου	125
A.6	Κώδικας εφαρμογής sHumidity	126
A.7	EIB OPC–Server	127
A.8	Χειριστές σφαλμάτων Try...Catch για την AJAX	128
A.9	Εφαρμογή OPC to DB	129
A.10	Εφαρμογή Checkbot	131
A.11	AJAX Feed	132
A.12	Κόστος εγκατάστασης	134

Περίληψη

Στο θεωρητικό μέρος της εργασίας γίνεται μία σύντομη εισαγωγή στο λογισμικό ανοιχτής αρχιτεκτονικής. Αναλύονται οι βασικοί τρόποι επικοινωνίας μεταξύ των διεργασιών στο Win32 ενώ στην συνέχεια παρουσιάζεται η τεχνολογία COM η οποία είναι η βάση του προτύπου OPC (OLE for Process Control). Το πρότυπο OPC αναπτύχθηκε για εφαρμογές βιομηχανικού αυτοματισμού. Δημιουργήθηκε για να αντιμετωπίσει το πρόβλημα των πολλών οδηγών (drivers), που εμφανίζεται συχνά όταν πρέπει να δημιουργηθεί ένα σύνθετο σύστημα αυτομάτου ελέγχου από ετερογενείς μονάδες διάφορων κατασκευαστών.

Στο τέλος του δεύτερου κεφαλαίου δίνεται ένα παράδειγμα χρήσης της τεχνολογίας OPC μέσα σε έγγραφο του Microsoft Excel. Στόχος είναι να υποδειχθεί η διαδικασία αρχικοποίησης υλικού και λογισμικού καθώς και να δοθεί ένα υπόδειγμα κώδικα για την ανάπτυξη πελατών OPC.

Στο τρίτο κεφάλαιο ακολουθεί η ανάλυση, η σχεδίαση και η ανάπτυξη ενός συστήματος το οποίο σαν βασική διασύνδεση χρήστη χρησιμοποιεί μία διαδικτυακή εφαρμογή, η οποία μέσω της τεχνολογίας automation της Microsoft, υποστηρίζει τα απαραίτητα αντικείμενα και τις διασυνδέσεις του Data Access V2.02 του OPC.

Στο τέταρτο κεφάλαιο ακολουθεί η πραγματική εφαρμογή του συστήματος που αναπτύχθηκε στην υπάρχουσα εγκατάσταση KNX/EIB του εργαστηρίου Ελέγχου Βιομηχανικών Συστημάτων. Επίσης αναπτύχθηκαν ελεγκτές οπτικής και θερμικής άνεσης και εξετάστηκε η αποτελεσματικότητά τους.

Τέλος παρουσιάζονται τα συμπεράσματα που προέκυψαν μετά την ολοκλήρωση της εργασίας καθώς και προτάσεις για περαιτέρω βελτίωση μέσω της χρήσης νέων ακόμα αποδοτικότερων τεχνολογιών.

1 Εισαγωγή

1.1 Συστήματα SCADA

Ο όρος SCADA αποτελεί τα αρχικά των λέξεων Supervisory Control and Data Acquisition, δηλαδή συστήματα εποπτείας, ελέγχου και συλλογής πληροφοριών. Έχει καθιερωθεί στη διεθνή ορολογία των συστημάτων αυτοματισμού και δηλώνει τα συστήματα εκείνα, που επιτρέπουν στο χειριστή μιας κατανεμημένης στο χώρο διεργασίας να συλλέγει πληροφορίες από διάφορα σημεία σε ένα κεντρικό Η/Υ, από τον οποίο μπορεί επίσης να εκτελεί χειρισμούς ή να στέλνει εντολές ελέγχου έχοντας την εποπτεία της διαδικασίας. Τα συστήματα SCADA είναι ιδιαίτερα δημοφιλή, τόσο σε σύνθετες βιομηχανικές εφαρμογές, που απαιτούνται αυτοματισμοί υψηλής αξιοπιστίας και ακρίβειας όσο και σε δίκτυα επιχειρήσεων και οργανισμών κοινής ωφελείας όπως σε δίκτυα μεταφοράς και διανομής ηλεκτρικής ενέργειας, νερού ή φυσικού αερίου, σε δίκτυα αποχέτευσης, σε συστήματα αυτοματισμού πολυώροφων κτιρίων ή άλλων μεγάλων εγκαταστάσεων, στα σιδηροδρομικά δίκτυα, σε δίκτυα άρδευσης και αλλού (1).

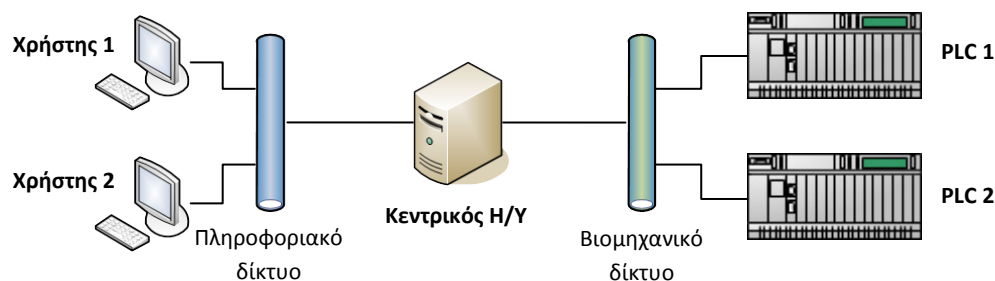
Παρόλο που τα συστήματα αυτά προορίζονται κυρίως για εφαρμογές μεγάλης κλίμακας, στις μέρες μας γνωρίζουν μία επιπλέον αλματώδη άνθηση, κυρίως λόγω της επέκτασης τους και στην καθημερινή ζωή. Τα λεγόμενα «έξυπνα σπίτια» στηρίζουν την λειτουργία τους, στις αρχές των συστημάτων αυτών και με τη χρήση κατάλληλων διατάξεων PLC και πακέτων λογισμικού δείχνουν το δρόμο στο πως η τεχνολογία μπορεί κάτω από συγκεκριμένες συνθήκες να κάνει την ανθρώπινη ζωή ακόμα πιο άνετη και ταυτόχρονα φιλική στο περιβάλλον.

1.2 Βασικά μέρη ενός SCADA

Τα μικρής κλίμακας σύστημα SCADA χρησιμοποιούν έναν κεντρικό Η/Υ συνήθως μεγάλης υπολογιστικής ισχύος, στον οποίο βρίσκεται εγκατεστημένο το πακέτο λογισμικού SCADA με τα διάφορα προγράμματα της εκάστοτε εφαρμογής.

Ένα πολύ σημαντικό κομμάτι στην ολοκλήρωση ενός συστήματος SCADA αποτελούν οι τύποι των δικτύων, με τα οποία επικοινωνούν τόσο ο κεντρικός Η/Υ με τους χρήστες (πληροφοριακό δίκτυο) όσο και οι διάφορες διατάξεις PLC μεταξύ τους (βιομηχανικό δίκτυο). Για μία σύνθετη εφαρμογή, συνήθως απαιτούνται αρκετές διατάξεις PLC κάθε μία από τις οποίες είναι επιφορτισμένη με ένα συγκεκριμένο τμήμα μιας κατανεμημένης στο χώρο διεργασίας. Ο τρόπος διασύνδεσης αυτών έχει άμεσες συνέπειες στην ορθή λειτουργία και ανταπόκριση του συστήματος. Συνήθως για την δικτύωση του κεντρικού Η/Υ με τους χρήστες χρησιμοποιείται το

πρωτόκολλο Ethernet ενώ για την δικτύωση των διατάξεων PLC χρησιμοποιούνται πρότυπα όπως το Profibus και το ModBus (2). Για την περίπτωση των «έξυπνων σπιτιών» έχει αναπτυχθεί το πρότυπο KNX/EIB ένα δίκτυο μεταφοράς και επεξεργασίας δεδομένων μεγάλης ευελιξίας, σχεδιασμένο για διεργασίες μικρής κλίμακας αλλά μεγάλης αξιοπιστίας.



Εικόνα 1-1 Δικτύωση ενός συστήματος SCADA

Ένα άλλο πολύ σημαντικό κομμάτι ενός συστήματος SCADA είναι το λογισμικό του κέντρου ελέγχου. Η ποιότητα του είναι καθοριστικής σημασίας για την καλή λειτουργία ολόκληρου του συστήματος SCADA. Συνήθως το λογισμικό αυτό είναι ένα εξαιρετικά πολύπλοκο. Χρειάζεται να διαχειρίζεται σωστά έναν όγκο πληροφοριών πραγματικού χρόνου και να τις δρομολογεί σωστά με τέτοιο τρόπο ώστε ο κάθε χρήστης του συστήματος και η κάθε μηχανή, που συνδέεται σε αυτό, να έχει την πληροφορία, που χρειάζεται, τη στιγμή, που τη χρειάζεται και στη μορφή, που τη χρειάζεται. Επιπλέον, καθώς τόσο οι ανάγκες της επιχείρησης όσο και οι διαθέσιμες τεχνολογικές λύσεις δεν παραμένουν σταθερές σε όλη τη διάρκεια της ζωής του συστήματος αλλά μεταβάλλονται με την πάροδο του χρόνου, το λογισμικό του κέντρου ελέγχου πρέπει να είναι αρκετά ευέλικτο και επεκτάσιμο, ώστε να μπορεί να προσαρμόζεται εύκολα και ομαλά στις καινούργιες κάθε φορά συνθήκες. Όπως έχει αποδειχτεί στην πράξη, οι απαιτήσεις από το λογισμικό των SCADA είναι πολλές φορές τόσο πολύπλοκες και τόσο πολύ διαφορετικές μεταξύ τους, ώστε να είναι αδύνατον να καλυφθούν από τις δυνατότητες ενός μόνου εργαλείου, όσο πλούσιο και αν είναι αυτό. Για το λόγο αυτό σήμερα θεωρείται πολύ σημαντικό ένα πακέτο λογισμικού SCADA να μπορεί να συνεργάζεται με άλλα προγράμματα και να ενσωματώνει στις εφαρμογές του εξειδικευμένο λογισμικό άλλων κατασκευαστών όποτε χρειαστεί (1).

Η αγορά σήμερα ζητά κέντρα ελέγχου ανοιχτής αρχιτεκτονικής, βασισμένα σε καθιερωμένα διεθνή πρότυπα, τα οποία επιτρέπουν τη συνένωση και τη συνεργασία ανεξάρτητων μεταξύ τους προϊόντων υλικού ή λογισμικού, τα οποία ενδεχομένως προέρχονται από διαφορετικούς κατασκευαστές. Η σύνδεση των διαφορετικών αυτών κομματιών και η ενσωμάτωσή τους σε ένα ολοκληρωμένο σύστημα πραγματοποιείται με τη βοήθεια ειδικών τεχνολογιών, που περιγράφονται συνολικά με τον όρο «Τεχνολογίες Ολοκλήρωσης» (Integration Technologies).

2 Λογισμικό ανοιχτής αρχιτεκτονικής

2.1 Επικοινωνία μεταξύ διεργασιών στο Win32

Στα Microsoft Windows, κάθε διεργασία έχει τον δικό της ιδιωτικό χώρο εικονικών διευθύνσεων στον οποίον έχει πρόσβαση μόνο αυτή και ο πυρήνας του λειτουργικού συστήματος. Προκειμένου να υπάρχει μία στοιχειώδης προστασία μνήμης μεταξύ των διαφορετικών διεργασιών, καμία διεργασία δεν έχει δικαίωμα ανάγνωσης ή εγγραφής στον χώρο διευθύνσεων των υπολοίπων. Αν χρειαστεί δύο διαφορετικές διεργασίες να επικοινωνήσουν μεταξύ τους ή να ανταλλάξουν δεδομένα, αυτό μπορεί να γίνει είτε μέσω του συστήματος αρχείων είτε μέσω μίας από τις διαθέσιμες μεθόδους διαδιεργασιακής επικοινωνίας IPC (InterProcess Communication) (3).

2.1.1 Νήματα και Διεργασίες

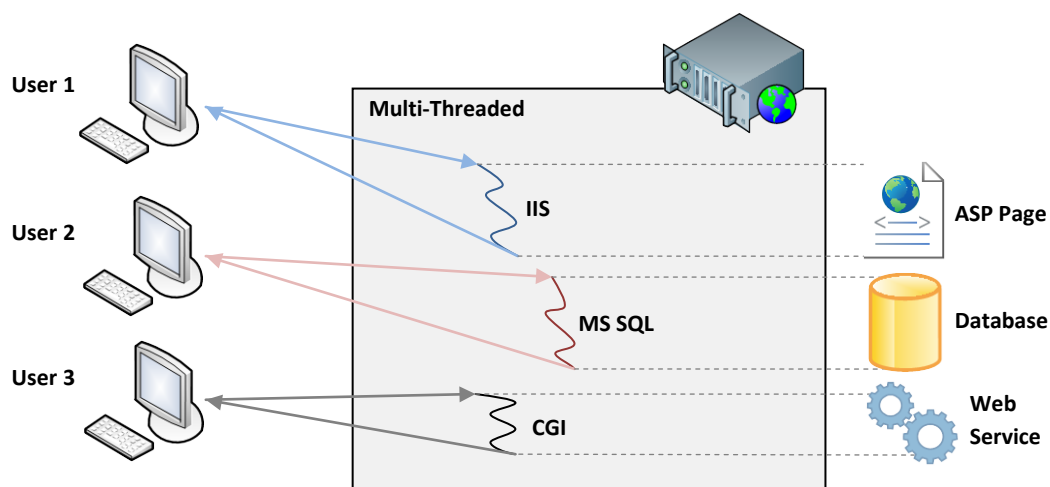
Τα λειτουργικά συστήματα που υλοποιούν το μοντέλο του Win32 υποστηρίζουν την πολυνηματική επεξεργασία (multithreaded processing). Αυτό σημαίνει ότι ο πυρήνας των Windows θεωρεί τα νήματα ως την κύρια μονάδα εκτέλεσης, τα οποία εκτελούνται εκ περιτροπής για ένα καθορισμένο χρονικό διάστημα. Στα πολυδιεργασιακά συστήματα ένας επεξεργαστής μπορεί να διαχειριστεί ένα νήμα κάθε στιγμή, έτσι το σύστημα εναλλάσσεται ταχύτατα μεταξύ των νημάτων για να επεξεργαστεί τα δεδομένα. Η Intel πριν μερικά χρόνια παρουσίασε την Hyper Threading Technology που υλοποιείται σε συστήματα με έναν επεξεργαστή και τα κάνει να εμφανίζονται σαν να διαθέτουν πολλαπλούς επεξεργαστές. Εκείνο το οποίο επιτυγχάνεται είναι η αύξηση του ρυθμού με τον οποίο το σύστημα μπορεί να εναλλάσσεται μεταξύ πολλών νημάτων.

Για κάθε νήμα τα Windows διατηρούν ένα σύνολο δεδομένων όπου κρατείται το περιεχόμενο του όσο το νήμα περιμένει να δρομολογηθεί για επεξεργασία. Το περιεχόμενο ενός νήματος περιλαμβάνει:

- Τις τιμές των καταχωρητών.
- Τη στοίβα του πυρήνα.
- Μια εγγραφή για το περιβάλλον του νήματος.
- Τη στοίβα του ίδιου του νήματος που βρίσκεται στο χώρο μνήμης.

Κατά την δημιουργία μιας νέας διεργασίας τα Windows αυτόματα δημιουργούν το πρώτο της νήμα το οποίο καλείται και το κύριο (primary) νήμα. Αυτό μπορεί μετά να δημιουργήσει επιπλέον νήματα της διεργασίας αυτής και εκείνα με την σειρά τους άλλα. Όλα τα νήματα της διεργασίας μοιράζονται τον εικονικό χώρο διευθύνσεων της και μπορούν να προσπελάσουν τις παγκόσμιες μεταβλητές της όπως και τους πόρους του συστήματος που κατέχει. Αν ένα οποιοδήποτε νήμα μιας διεργασίας δεν έχει τερματιστεί, η διεργασία δεν μπορεί να τερματίσει (3).

Η πολυνηματική επεξεργασία είναι χρήσιμη σε εφαρμογές που εκτελούν πολλές ανεξάρτητες μεταξύ τους διεργασίες που δεν είναι απαραίτητο να είναι σε σειρά, όπως για παράδειγμα οι διεργασίες σε έναν εξυπηρετητή ιστού (web server).



Σχήμα 2-1 Παράδειγμα πολυνημάτωσης σε έναν εξυπηρετητή ιστού

Το Win32 διαχειρίζεται διεργασίες (processes) κάθε μία από τις οποίες μπορεί να αποτελείται από πολλά νήματα εκτέλεσης. Μία διεργασία είναι το στιγμιότυπο μίας εκτελούμενης εφαρμογής και περιέχει τα εξής:

- Τουλάχιστον ένα νήμα εκτέλεσης.
- Ένα χώρο εικονικών διευθύνσεων.
- Τμήματα φυσικής μνήμης αντιστοιχισμένα σε εικονικές διευθύνσεις της διεργασίας.
- Τμήματα του Swap File αντιστοιχισμένα σε εικονικές διευθύνσεις της διεργασίας.
- Τμήματα του εκτελέσιμου αρχείου που εκτελεί η διεργασία αντιστοιχισμένα σε εικονικές διευθύνσεις της διεργασίας.
- Δεσμευμένους πόρους του συστήματος όπως ανοιχτά αρχεία κ.α.

2.1.2 Μέθοδοι διαδιεργασιακής επικοινωνίας (InterProcess Communication)

Το Win32 διαθέτει μεθόδους επικοινωνίας διεργασιών οι οποίες αφορούν έναν μόνο υπολογιστή, ή μπορούν να λειτουργήσουν και σε διαφορετικούς υπολογιστές ενός δικτύου. Οι μέθοδοι επικοινωνίας και ανταλλαγής δεδομένων στον ίδιο υπολογιστή είναι οι εξής:

- Ανώνυμες σωληνώσεις.
- Μοιραζόμενη μνήμη, μέσω του memory mapped files.
- Η ενσωμάτωση αρχείων με το OLE (Object Linking and Embedding). Το πρωτόκολλο αυτό παρέχει την δυνατότητα της δημιουργίας σύνθετων αρχείων.
- Το μήνυμα WM_COPYDATA. Με αυτό το μήνυμα μπορούν να αποσταλούν δεδομένα που βρίσκονται στον χώρο διεύθυνσεως της μίας διεργασίας και το λειτουργικό σύστημα αναλαμβάνει να τα «μεταφέρει» στο χώρο διεύθυνσεων της άλλης.

Ενώ οι δικτυακές μέθοδοι επικοινωνίας και ανταλλαγής δεδομένων μεταξύ διεργασιών που διαθέτει το Win32 είναι οι εξής:

- Ο Clipboard. Διεργασίες στον ίδιο ή σε διαφορετικούς υπολογιστές μπορούν να ανταλλάξουν δεδομένα μέσω του clipboard, συμφωνώντας μόνο για την κωδικοποίηση των δεδομένων.
- Το πρωτόκολλο DDE (Dynamic Data Exchange). Είναι η επέκταση του clipboard η οποία παρέχει υπηρεσίες συνεχούς ανταλλαγής δεδομένων.
- Οι σωληνώσεις (named pipes).
- Οι κλήσεις RPC (Remote Procedure Call), οι οποίες είναι συμβατές με το πρωτόκολλο DCE (Distributed Computing Environment) του Open Software Foundation και παρέχουν την δυνατότητα για την επικοινωνία των διεργασιών μεταξύ διαφορετικών λειτουργικών συστημάτων που υποστηρίζουν το DCE.
- Τα Windows sockets, τα οποία και αυτά παρέχουν την δυνατότητα επικοινωνίας με οποιοδήποτε λειτουργικό σύστημα υλοποιεί τα sockets.

2.1.3 DDE (Dynamic Data Exchange)

Το DDE πρωτοεμφανίστηκε στα Windows το 1987. Είναι ένας μηχανισμός που επιτρέπει σε μία εφαρμογή να στέλνει δεδομένα ή εντολές σε μία άλλη υπό την προϋπόθεση ότι και οι δυο έχουν σχεδιαστεί με τέτοιο τρόπο ώστε να το υποστηρίζουν. Με τον καιρό αποδείχτηκε ότι το DDE δεν ήταν πολύ ισχυρό, οι δυνατότητες του ήταν πολύ περιορισμένες και δεν μπορούσαν να καλύψουν τις ανάγκες της αγοράς. Αν και το DDE εξακολουθεί να υπάρχει ακόμη και στις

σύγχρονες εκδόσεις των Windows σε διάφορα σημεία τους, όπως για παράδειγμα στην αντιγραφή, την αποκοπή και αλλού, η Microsoft προχώρησε στην ανάπτυξη νεότερων τεχνολογιών όπως το πρότυπο COM (4).

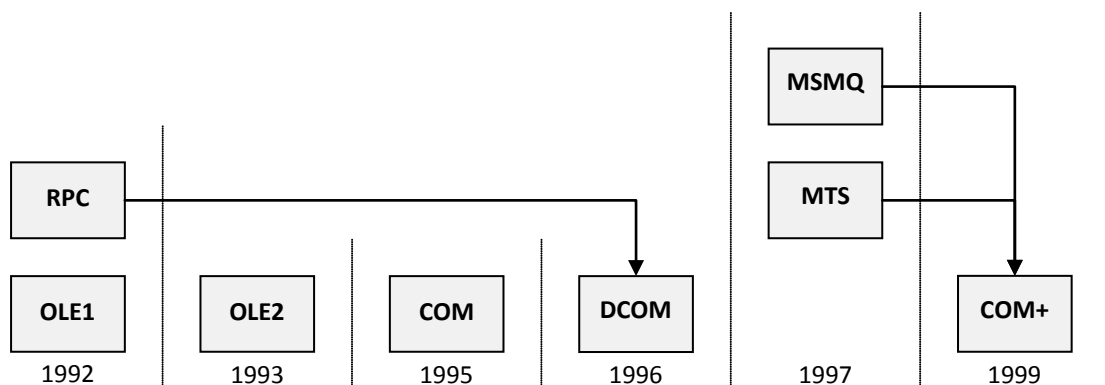
2.1.4 OLE (Object Linking and Embedding)

Πριν την εμφάνιση του πρότυπου COM η Microsoft είχε εισάγει στα Windows την τεχνολογία OLE. Η αρχική έκδοση του OLE βασιζόταν στην τεχνολογία του DDE. Το OLE επιτρέπει στον χρήστη να τοποθετεί «αντικείμενα περιεχομένου» που έχουν δημιουργηθεί από μία εφαρμογή μέσα σε έγγραφα μιας άλλης εφαρμογής. Το πρωτόκολλο του OLE είναι αρκετά περίπλοκο και περιγράφει πως ένας εξυπηρετητής OLE μπορεί να επικοινωνεί με ένα περίβλημα OLE και αντίστροφα. Κλασσικό παράδειγμα χρήσης του OLE είναι η ενσωμάτωση ενός φύλλου εργασίας του Excel μέσα σε ένα έγγραφο του Word. Η έννοια της ενσωμάτωσης (embedding) είναι κεντρικής σημασίας για την χρήση των πολυμέσων μέσα στις ιστοσελίδες, οι οποίες τείνουν να ενσωματώνουν διάφορων τύπων αρχεία μέσα στην περιγραφική γλώσσα HTML (HyperText Markup Language).

2.2 Τεχνολογία COM (Component Object Model)

2.2.1 Εισαγωγή

Με την εμφάνιση της έκδοσης 2.0 του OLE η Microsoft εισήγαγε την τεχνολογία COM για την δυνατότητα ανάπτυξης εφαρμογών από αντικείμενα που παρέχονται από διαφορετικούς προμηθευτές. Σε αντίθεση με την έκδοση 1.0 η νέα έκδοση του OLE δεν κάνει χρήση της τεχνολογίας DDE. Αν και αποτελεί πνευματική ιδιοκτησία της Microsoft το COM είναι ένα ανοιχτό πρότυπο που μπορεί να χρησιμοποιηθεί από οποιονδήποτε. Το πρότυπο COM συνδυάζει δύο βασικές αρχές που κυριαρχούν στην τεχνολογία του λογισμικού τα τελευταία χρόνια. Τον αντικειμενοστρεφή προγραμματισμό και την δημιουργία επαναχρησιμοποιήσιμου κώδικα. Με την χρήση του αντικειμενοστρεφή προγραμματισμού προσπαθούμε να λύσουμε ένα σύνθετο πρόβλημα σε απλούστερα υποπροβλήματα κάθε ένα από τα οποία αντιστοιχεί σε ένα ανεξάρτητο αντικείμενο λογισμικού. Από την άλλη μεριά ο επαναχρησιμοποιήσιμος κώδικας αφορά τμήματα λογισμικού που έχουν κατασκευαστεί με συγκεκριμένη τυποποίηση η οποία τους επιτρέπει να χρησιμοποιούνται από πολλές διαφορετικές εφαρμογές. Το COM κατόρθωσε να είναι η πιο καθιερωμένη από τις αντικειμενοστρεφείς τεχνολογίες επαναχρησιμοποιήσιμου κώδικα. Στην ουσία το COM είναι ένας τυποποιημένος τρόπος για να φτιάχνουμε αντικείμενα λογισμικού τα οποία μπορούν να συνεργάζονται μεταξύ τους έστω και αν έχουν γραφτεί σε διαφορετικές γλώσσες προγραμματισμού ή διαφορετικά περιβάλλοντα ανάπτυξης.



Σχήμα 2-2 Ιστορική εξέλιξη του μοντέλου COM

2.2.2 Χαρακτηριστικά COM

Τα βασικά χαρακτηριστικά και οι μηχανισμοί που παρέχει το πρότυπο COM συνοψίζονται στα παρακάτω:

- Καθορίζει δυαδικά πρότυπα για την συνεργασία (επικοινωνία) των διαφόρων διεργασιών όχι μόνο στα όρια ενός Η/Υ αλλά και μεταξύ δικτύων.
- Είναι ανεξάρτητο της γλώσσας ανάπτυξης και παρέχεται και σε λειτουργικά συστήματα εκτός των Microsoft Windows.
- Παρέχει αρκετές διασυνδέσεις και την βασική διασύνδεση IUnknown.
- Παρέχει τον μηχανισμό αναφοράς σφάλματος και κατάστασης HRESULT.
- Παρέχει τον μηχανισμό GUID ο οποίος προσδιορίζει μοναδικά τα στοιχεία και τις διασυνδέσεις τους.

2.2.2.1 Διασυνδέσεις (Interfaces) και το IUnknown

Κάθε αντικείμενο COM παρέχει στις εφαρμογές που το χρησιμοποιούν έναν αριθμό από μεθόδους δηλαδή υποπρογράμματα (συναρτήσεις) που υλοποιούν τις λειτουργίες του. Οι μέθοδοι ενός αντικειμένου COM είναι οργανωμένες σε μία ή περισσότερες ομάδες που ονομάζονται interfaces και συνήθως κάτω από το ίδιο interface υπάρχουν μέθοδοι που εκτελούν κοινές λειτουργίες. Η Microsoft έχει προκαθορίσει αρκετές βασικές διασυνδέσεις που υποστηρίζονται από οποιοδήποτε αντικείμενο COM και ονομάζονται standard interfaces. Ωστόσο οι προγραμματιστές των αντικειμένων COM μπορούν να ορίσουν και άλλες που πιθανό να χρειάζονται στην ανάπτυξη εφαρμογών. Κάθε διασύνδεση χαρακτηρίζεται από έναν μοναδικό αριθμό μεγέθους 128 bit που αποκαλείται IID (Interface Identifier) και υλοποιεί υποχρεωτικά το standard interface το οποίο ονομάζεται IUnknown (5).

Η διασύνδεση αυτή περιλαμβάνει τρεις μεθόδους όπως φαίνεται στον πίνακα 2-1.

Πίνακας 2-1: Μέθοδοι της βασικής διασύνδεσης IUnknown

Μέθοδοι	Ιδιότητες
QueryInterface	Επιστρέφει έναν δείκτη (pointer) δηλαδή την διεύθυνση μίας περιοχής μνήμης που δείχνει σε ένα άλλο interface του συγκεκριμένου ή άλλων αντικειμένων.
AddRef	Προσαυξάνει έναν μετρητή που μας δίνει τον αριθμό των ενεργών συνδέσεων με τις εφαρμογές ο οποίος μας αναφέρει ότι το συγκεκριμένο αντικείμενο COM χρησιμοποιείται.
Release	Μειώνει τον μετρητή αναφοράς κατά μία μονάδα όταν μια εφαρμογή σταματήσει να χρησιμοποιεί ένα αντικείμενο COM. Όταν ο μετρητής αυτός φτάσει στο μηδέν τότε το αντικείμενο δεν χρησιμοποιείται από κανένα πρόγραμμα και μπορεί να αφαιρεθεί από την μνήμη του Η/Υ.

2.2.2.2 Αναφορά σφάλματος και κατάστασης

Το HRESULT είναι μια κρυπτογραφημένη αριθμητική τιμή που χρησιμοποιείται σε πολλά είδη τεχνολογιών της Microsoft. Η τιμή που επιστρέφει περιγράφει τα σφάλματα και τις προειδοποιήσεις σε μία διεργασία (6). Μία τιμή HRESULT έχει 32bits και χωρίζεται σε τρία διαφορετικά πεδία:

- Το severity code που υποδεικνύει αν η τιμή είναι πληροφορία, προειδοποίηση ή σφάλμα.
- Το facility code που αναγνωρίζει την περιοχή του συστήματος που είναι υπεύθυνη για το σφάλμα.
- Το error code που είναι ένας μοναδικός αριθμός που προσδιορίζει την ταυτότητα της εξαίρεσης.

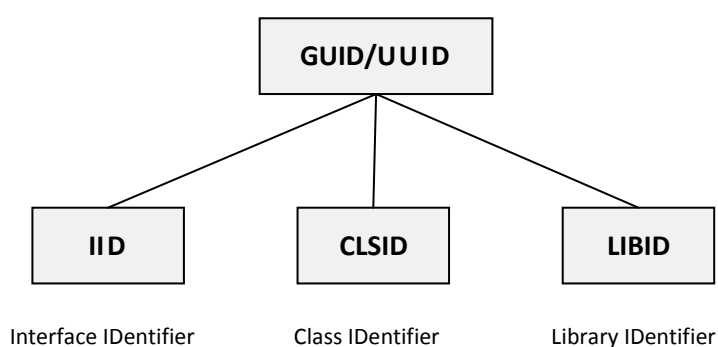
Sev	C	R	Facility code	Error code
31			15	0

Σχήμα 2-3 Δομή τιμής HRESULT 32bits

Ένα παράδειγμα επιστροφής μίας τιμής HRESULT είναι το μήνυμα σφάλματος «Απαγορεύεται η πρόσβαση» όταν μια εφαρμογή προσπαθεί να χρησιμοποιήσει ένα στοιχείο χωρίς να έχουν οριστεί τα κατάλληλα δικαιώματα πρόσβασης για τον χρήστη που εκτελεί την εφαρμογή. Δηλαδή το σύστημα θα επιστρέψει την τιμή 0x80070005 όπου το 8 υποδηλώνει "Failure", το 7 "Win32" και το 5 "Access Denied".

2.2.2.3 GUID (Globally Unique Identifier)

Το πρότυπο COM χρησιμοποιεί τον ευρύτερο όρο GUID, που δηλώνει έναν ακέραιο αριθμό 128 bit ο οποίος εγγυημένα είναι μοναδικός και χρησιμοποιείται είτε για να προσδιορίζει το interface (Interface Identifier, IID) είτε το κάθε interface την κλάση των αντικειμένων COM (Class IDs, CLSID). Το GUID είναι UUID (παγκοσμίως μοναδικό ID) όπως καθορίζεται από το Open Foundation Distributed Computer Environment (1). Η Microsoft παρέχει ένα εργαλείο που παράγει αυτόματα GUIDs. Έτσι οι ειδικοί ανάπτυξης αντικειμένων COM δημιουργούν τα δικά τους GUIDs με χρήση χαρακτήρων και χωρίς να χρειάζεται να χρησιμοποιούν τον πραγματικό 128 bit GUID.



Σχήμα 2-4 Κατηγορίες GUID/UUID

2.2.3 Αρχιτεκτονική COM

Το πρότυπο COM βασίζεται στην αρχιτεκτονική πελάτη/εξυπηρετητή (client/server). Το αντικείμενο COM είναι ο εξυπηρετητής (server) ενώ το πρόγραμμα που καλεί τις μεθόδους του COM είναι ο πελάτης (client). Για την ακρίβεια η Microsoft ορίζει ως εξυπηρετητή COM ένα εκτελέσιμο αρχείο, που υλοποιεί ένα αντικείμενο COM. Το μητρώο των Windows και συγκεκριμένα η περιοχή HKEY_CLASSES_ROOT\CLSID περιέχει πληροφορίες που σχετίζουν τα CLSIDs με τα εκτελέσιμα αρχεία των αντικειμένων COM.

Η αρχιτεκτονική αυτή ωφελεί σημαντικά στην διαχείριση των σφαλμάτων, για παράδειγμα αν ο εξυπηρετητής «κρεμάσει» ή αποσυνδεθεί από έναν πελάτη τότε ο πελάτης μπορεί να χειριστεί το σφάλμα αυτό εύκολα ή ακόμα και να επανεκινήσει τον εξυπηρετητή. Η δυνατότητα να βρίσκονται τα αντικείμενα πελάτη και τα αντικείμενα εξυπηρετητή σε διαφορετικούς Η/Υ είναι πολύ σημαντική σε ένα πρότυπο λογισμικού που πρόκειται να υποστηρίξει περίπλοκες εφαρμογές.

2.2.4 Είδη εξυπηρετητών COM

Υπάρχουν δύο βασικά είδη εξυπηρετητών COM. Οι in-process servers και οι out-of-process servers. Οι in-process servers είναι αρχεία τύπου DLL (Dynamically Linked Libraries). Χαρακτηρίζονται in-process διότι εκτελούνται στην ίδια διεργασία με τον πελάτη δηλαδή μοιράζονται τον ίδιο χώρο μνήμης. Από την άλλη πλευρά οι out-of-process servers είναι αρχεία exe που εκτελούνται σε ξεχωριστή διεργασία από αυτή του πελάτη. Συγκριτικά οι εξυπηρετητές in-process είναι πολύ γρηγορότεροι από τους out-of-process από την άλλη μεριά όμως οι out-of-process παρέχουν πολύ μεγαλύτερη ασφάλεια αφού χρησιμοποιούν διαφορετικές περιοχές μνήμης από αυτές των πελατών.

Με την εμφάνιση της τεχνολογίας DCOM (Distributed COM) οι εξυπηρετητές out-of-process μπορούν να εκτελούνται σε διαφορετικούς Η/Υ από αυτούς των πελατών. Οι ειδικοί του COM για να διακρίνουν τους εξυπηρετητές out-of-process που εκτελούνται στον ίδιο υπολογιστή με τους πελάτες από αυτούς που εκτελούνται σε διαφορετικό υπολογιστή, χρησιμοποιούν συχνά τους όρους τοπικός εξυπηρετητής (local server) και απομακρυσμένος εξυπηρετητής (remote server).

Έτσι συνολικά έχουμε τρεις κατηγορίες εξυπηρετητών COM όπως φαίνεται στον παρακάτω πίνακα (7):

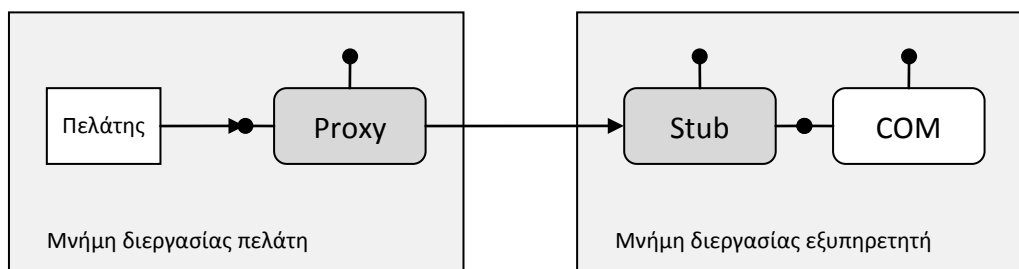
Πίνακας 2-2: Τύποι εξυπηρετητών COM

Τύπος εξυπηρετητή	Περιγραφή
In-process	Εκτελούνται κάτω από την ίδια διεργασία με τον πελάτη και μοιράζονται τον ίδιο χώρο μνήμης.
Local out-of-process	Εκτελούνται σε διαφορετικές διεργασίες με τους πελάτες αλλά στον ίδιο Η/Υ.
Remote out-of-process	Εκτελούνται σε διαφορετικές διεργασίες με τους πελάτες και σε διαφορετικούς Η/Υ.

2.2.5 Επικοινωνία πελάτη με εξυπηρετητή out-of-process

Όταν ένας πελάτης καλεί μια μέθοδο ενός εξυπηρετητή out-of-process ενεργοποιούνται οι μηχανισμοί οι οποίοι καθιστούν δυνατή την επικοινωνία μεταξύ των διαφορετικών διεργασιών. Στον χώρο μνήμης της διεργασίας του πελάτη υπάρχει ένα αντικείμενο εξουσιοδότησης (proxy) το οποίο διαθέτει τα ίδια interfaces με τον πελάτη. Αντίστοιχα στο χώρο μνήμης της διεργασίας του εξυπηρετητή υπάρχει ένα άλλο αντικείμενο (stub) το οποίο επικοινωνεί με το proxy χρησιμοποιώντας κάποιον από τους μηχανισμούς IPC (InterProcess Communication) που αναλύσαμε στην 2.1.2. Όταν ο πελάτης θέλει να χρησιμοποιήσει κάποια από τις

μεθόδους του εξυπηρετητή, καλεί την αντίστοιχη μέθοδο του proxy και αυτό με την σειρά του μεταβιβάζει κλήση στο stub το οποίο καλεί την μέθοδο του εξυπηρετητή. Με την ολοκλήρωση το stub επιστρέφει το αποτέλεσμα στο proxy και αυτό με την σειρά του στον πελάτη. Ο μηχανισμός κλήσης μίας μεθόδου ενός out-of-process εξυπηρετητή εικονίζεται στο σχήμα 2-5.



Σχήμα 2-5 Μηχανισμός κλήσης out-of-process

2.2.6 Το εργαλείο Regsvr32

Για την καταχώρηση ή την αφαίρεση στοιχείων OLE, όπως βιβλιοθήκες ή στοιχεία ελέγχου ActiveX χρησιμοποιούμε το εργαλείο regsvr32.exe το οποίο προσπαθεί να κάνει λήψη του στοιχείου και να καλέσει την λειτουργία DLLSelfRegister. Αν η προσπάθεια είναι επιτυχής θα εμφανιστεί ένα παράθυρο διαλόγου που δηλώνει επιτυχία διαφορετικά θα εμφανιστεί μήνυμα σφάλματος (7).

Για παράδειγμα τα βήματα για την καταχώρηση της υποθετικής βιβλιοθήκης name.dll με χρήση του εργαλείου regsvr32.exe είναι τα εξής:

1. Αρχικά αντιγραφούμε το στοιχείο name.dll στο φάκελο του λειτουργικού συστήματος WINDOWS\System32
2. Γράφουμε στην γραμμή εντολών την εντολή regsvr32 name.dll

Για την αφαίρεση της από το σύστημα γράφουμε την εντολή regsvr32 name.dll -u και στην συνέχεια διαγράφουμε το στοιχείο name.dll από τον φάκελο.

2.3 Τεχνολογία Automation

Για να καταφέρει ένας χρήστης να χρησιμοποιήσει το πρότυπο COM χρειάζεται να ξέρει αρκετές τεχνικές λεπτομέρειες για ένα πλήθος από διασυνδέσεις κάθε μια από τις οποίες αποτελείται από πολλές διαφορετικές συναρτήσεις. Για τον λόγο αυτό η Microsoft έχει αναπτύξει μια ολόκληρη οικογένεια από τεχνολογίες που βασίζονται στο πρότυπο COM αλλά είναι πιο φιλικές για τον χρήστη. Με τις τεχνολογίες αυτές είναι δυνατό οι προγραμματιστές των εφαρμογών να χρησιμοποιούν το COM χωρίς να ξέρουν όλες τις τεχνικές λεπτομέρειες. Η πιο γνωστή τεχνολογία είναι το automation, το οποίο επιτρέπει στους προγραμματιστές να επικοινωνούν με αντικείμενα COM χρησιμοποιώντας απλές εντολές σε συνηθισμένες γλώσσες προγραμματισμού.

Οι εξυπηρετητές automation είναι εξυπηρετητές COM που μπορεί να είναι είτε in-process είτε out-of-process. Το κοινό χαρακτηριστικό τους είναι ότι υλοποιούν μία συγκεκριμένη διασύνδεση που ονομάζεται IDispatch. Η σημαντικότερη μέθοδος του IDispatch είναι η Invoke η οποία επιτρέπει στους πελάτες να καλούν μεθόδους ή να διαχειρίζονται ιδιότητες του εξυπηρετητή. Έτσι όταν ο πελάτης θέλει να επικοινωνήσει με τον εξυπηρετητή, καλεί την μέθοδο Invoke δίνοντας της σαν παράμετρο ένα μοναδικό αναγνωριστικό αριθμό (DISPID) της συγκεκριμένης μεθόδου ή ιδιότητας που θέλει να χρησιμοποιήσει. Μία άλλη παράμετρος της Invoke ορίζει αν ο πελάτης θέλει να καλέσει μια μέθοδο του εξυπηρετητή, να διαβάσει ή να γράψει μία ιδιότητα (1).

Το πλεονέκτημα της τεχνολογίας automation είναι ότι οι προγραμματιστές δεν χρειάζεται να ξέρουν τίποτα για την διαδικασίες των διασυνδέσεων και των μεθόδων των αντικειμένων COM. Ο όλος χειρισμός γίνεται από το περιβάλλον της γλώσσας προγραμματισμού. Τα περιβάλλοντα που χρησιμοποιούν το automation επιτρέπουν στον χρήστη τις εξής δυνατότητες:

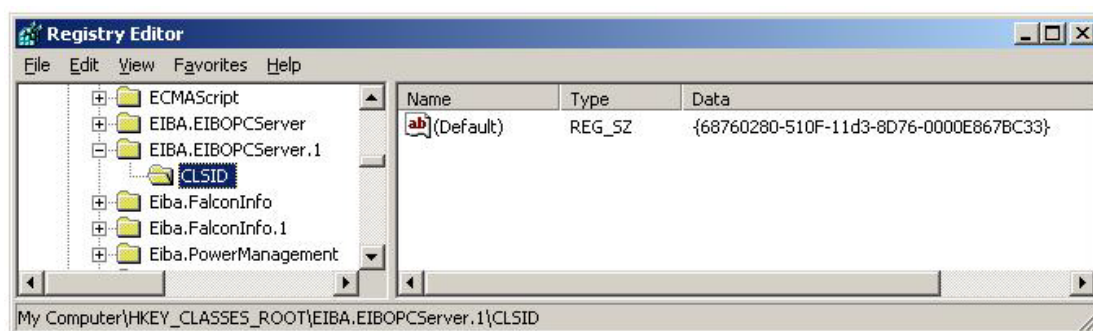
- Να δημιουργεί και να διαχειρίζεται αντικείμενα.
- Να καλεί μεθόδους των αντικειμένων αυτών.
- Να χρησιμοποιεί τις ιδιότητες τους.
- Να χρησιμοποιεί συλλογές των αντικειμένων.

Θα εξετάσουμε τις παραπάνω δυνατότητες του automation στην VBScript που είναι η βασική γλώσσα ανάπτυξης σε αυτή την διπλωματική εργασία.

2.3.1 Δημιουργία και διαχείριση αντικειμένων

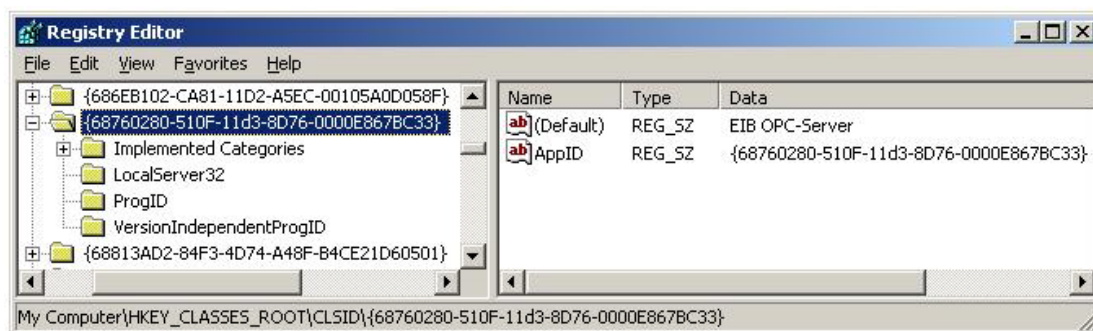
Οι γλώσσες προγραμματισμού που υποστηρίζουν την τεχνολογία automation επιτρέπουν την δημιουργία αντικειμένων που αντιστοιχούν σε εξυπηρετητές COM και την δυνατότητα να ορίζουμε μεταβλητές που να αναφέρονται στα αντικείμενα

αυτά. Για τις κλάσεις των αντικειμένων μπορούν να χρησιμοποιούνται συμβολικά ονόματα αντί για τους αριθμούς CLSID 128bit του COM. Αυτά τα ονόματα ονομάζονται PROGID (PROgram IDentifier) και φυλάσσονται στο μητρώο μαζί με τα αντίστοιχα CLSID. Για παράδειγμα στην εικόνα 2-1 βλέπουμε στο μητρώο το επιλεγμένο CLSID ενός εξυπηρετητή COM με την ονομασία EIBA.EIBOPCServer.1 (βλ. παράρτημα Α.7) την οποία ορίζει εξ αρχής ο προμηθευτής. Στην πραγματικότητα είναι ένα εκτελέσιμο αρχείο, που υλοποιεί ένα αντικείμενο COM όπου στην συγκεκριμένη περίπτωση παρέχει σε πελάτες OPC πρόσβαση στον δίαυλο KNX/EIB για επικοινωνία εισόδου/εξόδου.



Εικόνα 2-1: Εντοπισμός CLSID στο μητρώο

Στην συνέχεια ψάξαμε το συγκεκριμένο CLSID στην θέση που φυλάσσονται όλα τα CLSIDs δηλαδή στην HKEY_CLASSES_ROOT\CLSID για να εντοπίσουμε τις πληροφορίες συσχετισμού που περιέχει με το αντίστοιχο εκτελέσιμο αρχείο του EIB OPC-Server.



Εικόνα 2-2: Συσχετισμός CLSID με το αντίστοιχο εκτελέσιμο αρχείο

Στην VBScript η δημιουργία αντικειμένου automation γίνεται με την μέθοδο CreateObject με παράμετρο το PROGID του αντικειμένου. Στην περίπτωση που ο κώδικας VBScript εκτελείται μέσα σε δυναμικές σελίδες ASP η αντίστοιχη μέθοδος είναι η Server.CreateObject

Για παράδειγμα αν θέλουμε εκκινήσουμε το OPC Automation ως εξυπηρετητή automation και να τον συνδέσουμε σε μία μεταβλητή θα πρέπει να δώσουμε τις εξής εντολές:

```
Dim opcServer as Object  
Set opcServer = CreateObject("OPC.Automation.1")
```

Για τη απελευθέρωση της μνήμης που έχει δεσμευτεί για την επικοινωνία με τον εξυπηρετητή automation χρησιμοποιούμε την εντολή:

```
Set opcServer = Nothing
```

2.3.2 Κλίση μεθόδων

Οι προδιαγραφές του automation καθορίζουν τους τύπους δεδομένων, που μπορούν να λαμβάνουν ως παραμέτρους ή να επιστρέφουν ως αποτέλεσμα οι διάφορες μέθοδοι. Για να καλέσουμε μία μέθοδο γράφουμε το αντικείμενο ακολουθούμενο από μια τελεία και μετά το όνομα της μεθόδου και τυχών παραμέτρους μέσα σε παρένθεση.

Για παράδειγμα αν θέλουμε να συνδεθούμε σε έναν εξυπηρετητή OPC Data Access πρέπει να καλέσουμε την μέθοδο Connect του αντικειμένου OPC.Automation.1 η οποία παίρνει σαν παράμετρο το PROGID του εξυπηρετητή. Η εντολή που πρέπει να δώσουμε είναι:

```
opcServer.Connect("EIBA.EIBOPCServer.1")
```

2.3.3 Χρήση ιδιοτήτων

Οι εξυπηρετητές automation επιτρέπουν στους πελάτες τους να διαβάζουν τις ιδιότητες των αντικειμένων ή να δίνουν τιμές σε αυτές. Τα δεδομένα αυτά μπορούν να αφορούν την κατάσταση του αντικείμενου όπως για παράδειγμα την τελευταία τιμή που διάβασε ένας πελάτης από έναν αισθητήρα. Ένα παράδειγμα χρήσης ιδιοτήτων στο OPC Automation είναι η παρακάτω εντολή:

```
objGroup.IsSubscribed = True
```

2.3.4 Χρήση συλλογών

Στην τεχνολογία automation μια συλλογή είναι μια ομάδα από ομοειδή αντικείμενα. Όλες οι συλλογές έχουν τα εξής υποχρεωτικά χαρακτηριστικά:

- Μία ιδιότητα με την ονομασία count, η οποία είναι μόνο για ανάγνωση (read-only) και επιστρέφει το πλήθος των αντικειμένων της συλλογής.
- Μια μέθοδο με το όνομα item η οποία επιστρέφει ένα από τα αντικείμενα της συλλογής. Σαν παράμετρο η μέθοδος αυτή παίρνει έναν ακέραιο αριθμό που υποδηλώνει την θέση του αντικείμενου που θέλουμε να ανακτήσουμε.

Στο OPC Automation υποθέτοντας ότι έχουμε ορίσει μια συλλογή αντικειμένων π.χ. την `OPCItemsCollection` για να ανακτήσουμε το πρώτο αντικείμενο γράφουμε την εξής εντολή:

```
Set opcItem = OPCItemsCollection.Item(1)
```

Ενώ για να γράψουμε στην μεταβλητή `total` το πλήθος των αντικειμένων της συλλογής χρησιμοποιούμε την εντολή:

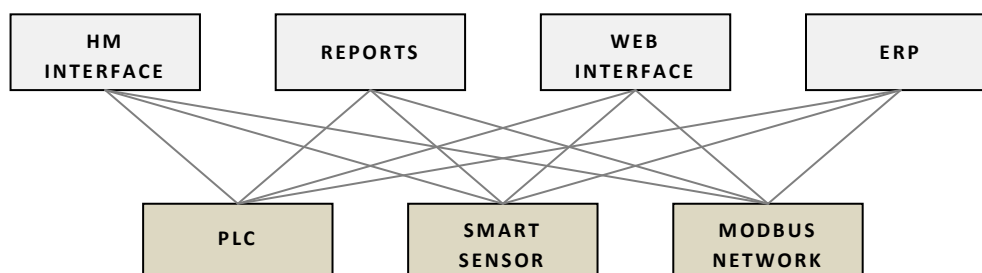
```
Total = OPCItemsCollection.Count
```

2.4 OLE for Process Control

2.4.1 Γενικά

Το OPC (OLE for Process Control) είναι μία τεχνολογία λογισμικού που στηρίζεται στο πρότυπο COM και αναπτύχθηκε ειδικά για εφαρμογές βιομηχανικού αυτοματισμού. Οι προδιαγραφές του OPC αναπτύσσονται και συντηρούνται από τον παγκόσμιο οργανισμό OPC foundation. Αρχικά το OPC υποστήριζε μόνο την ανταλλαγή δεδομένων πραγματικού χρόνου και συγκεκριμένα ψηφιακών και αναλογικών σημάτων. Σήμερα η λειτουργία αυτή αποτελεί ένα μόνο τμήμα του OPC, που ονομάζεται Data Access. Τα υπόλοιπα τμήματα του OPC αφορούν άλλα είδη ανταλλαγής πληροφοριών όπως μετάδοση συναγερμών και συμβάντων, ανάκτηση ιστορικών στοιχείων, αποστολή εντολών ελέγχου κ.τ.λ. (8).

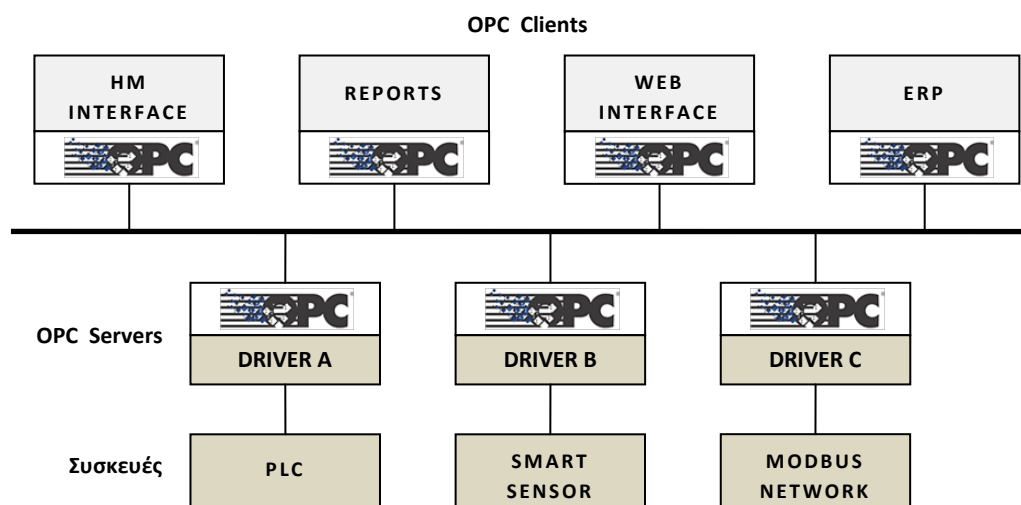
Το OPC δημιουργήθηκε για να αντιμετωπίσουμε το πρόβλημα των πολλών drivers, που εμφανίζεται συχνά όταν θέλουμε να δημιουργήσουμε ένα σύνθετο σύστημα συνδέοντας ετερογενείς μονάδες από διάφορους κατασκευαστές. Σε ένα κατακεντρωμένο σύστημα κάθε εφαρμογή του κέντρου ελέγχου θα πρέπει να έχει έναν ξεχωριστό driver για κάθε είδος συσκευής πεδίου με την οποία θέλει να επικοινωνήσει. Η ανάγκη υποστήριξης όλων αυτών των drivers αυξάνει σημαντικά την πολυπλοκότητα και το κόστος των εφαρμογών. Επιπλέον η συντήρηση και η αναβάθμιση του λογισμικού γίνεται εξαιρετικά δύσκολη αφού σε κάθε νέα έκδοση ενός πρωτόκολλου επικοινωνίας θα πρέπει να τροποποιούνται όλες οι εφαρμογές του κέντρου ελέγχου για να μπορέσουν να την υποστηρίξουν.



Σχήμα 2-6 Παράδειγμα υλοποίησης με πολλούς drivers

Μία καλύτερη προσέγγιση θα ήταν να υπάρχει ένας driver για κάθε συσκευή πεδίου. Οι drivers αυτοί θα φτιάχνονται από τους κατασκευαστές συσκευών πεδίου και θα ήταν προσβάσιμοι από όλες τις εφαρμογές του κέντρου ελέγχου. Επειδή όμως τα λειτουργικά συστήματα δεν προσφέρουν τυποποιημένους τρόπους ανταλλαγής δεδομένων, δημιουργήθηκε η τεχνολογία OPC εισάγοντας έναν τυποποιημένο τρόπο ανταλλαγής δεδομένων για εφαρμογές βιομηχανικού ελέγχου. Με λίγα λόγια το OPC λειτουργεί σαν «δίαυλος λογισμικού» που επιτρέπει σε οποιαδήποτε εφαρμογή του κέντρου ελέγχου να επικοινωνήσει με οποιαδήποτε συσκευή πεδίου.

Η τεχνολογία OPC χρησιμοποιεί την αρχιτεκτονική πελάτη/εξυπηρετητή. Οι εφαρμογές του κέντρου ελέγχου λειτουργούν ως πελάτες OPC ενώ το λογισμικό που οδηγεί τις συσκευές πεδίου λειτουργεί ως εξυπηρετητής OPC. Τόσο οι πελάτες όσο και ο εξυπηρετητής είναι αντικείμενα COM. Η ενσωμάτωση νέων συσκευών πεδίου στο σύστημα δεν απαιτεί καμία αλλαγή ή τροποποίηση του κώδικα των εφαρμογών αλλά γίνεται με απλή προσθήκη στους αντίστοιχους εξυπηρετητές. Θα πρέπει να σημειωθεί ότι το OPC επιτρέπει σε περισσότερους από έναν πελάτες να επικοινωνούν ταυτόχρονα με τον ίδιο εξυπηρετητή καθώς χειρίζεται αυτόματα όλες τις τεχνικές λεπτομέρειες της ταυτόχρονης πρόσβασης στα ίδια δεδομένα (8).



Σχήμα 2-7 Παράδειγμα υλοποίησης με την τεχνολογία OPC

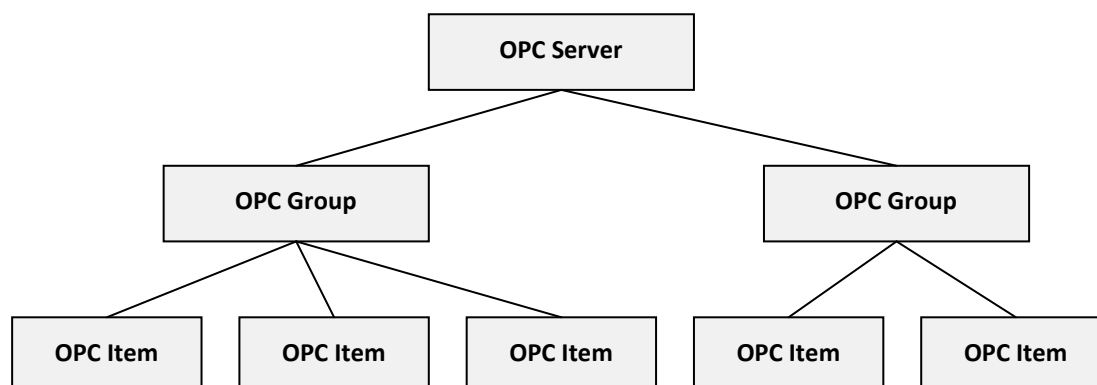
2.4.2 OPC Data Access

Η προδιαγραφή data access είναι η περισσότερο αναπτυγμένη από τις προδιαγραφές του OPC καθώς ήταν η πρώτη που δημιουργήθηκε. Ορίζει τα αντικείμενα και τις αντίστοιχες διασυνδέσεις που χρησιμοποιούνται για την ανταλλαγή πληροφοριών ανάμεσα σε έναν πελάτη OPC και έναν εξυπηρετητή τύπου data access.

Κάθε εξυπηρετητής data access παρέχει πρόσβαση σε κάποια σημεία της διαδικασίας. Αυτά τα σημεία συνήθως αντιστοιχούν σε πραγματικά αναλογικά ή ψηφιακά σήματα και είναι προκαθορισμένα για κάθε εξυπηρετητή. Οι πελάτες μπορούν να επιλέξουν με τα σημεία που θα επικοινωνήσουν αλλά δεν μπορούν να προσθέσουν καινούρια. Κάθε φορά που κάποια από τις συσκευές πεδίου αλλάζει (αναβάθμιση καρτών PLC κ.τ.λ.) θα πρέπει να ενημερώνεται ή να τροποποιείται και ο αντίστοιχος εξυπηρετητής.

Ένας πελάτης OPC αποκτάει πρόσβαση στα δεδομένα ενός εξυπηρετητή data access μέσω ενός ιεραρχικά δομημένου συνόλου από αντικείμενα COM. Το σύνολο αυτό αποτελείται από τρία είδη αντικειμένων, τα οποία είναι τα εξής (1):

- *OPC Servers*. Είναι τα βασικά αντικείμενα COM που υλοποιούν τους εξυπηρετητές OPC.
- *OPC Groups*. Αντιστοιχούν σε ομάδες από σήματα (σημεία διαδικασιών) που έχουν κάποιο κοινό χαρακτηριστικό.
- *OPC Items*. Αντιστοιχούν στα σήματα που μπορούν να διαβάσουν και να γράψουν οι πελάτες OPC.



Σχήμα 2-8 Μοντέλο αντικειμένων OPC

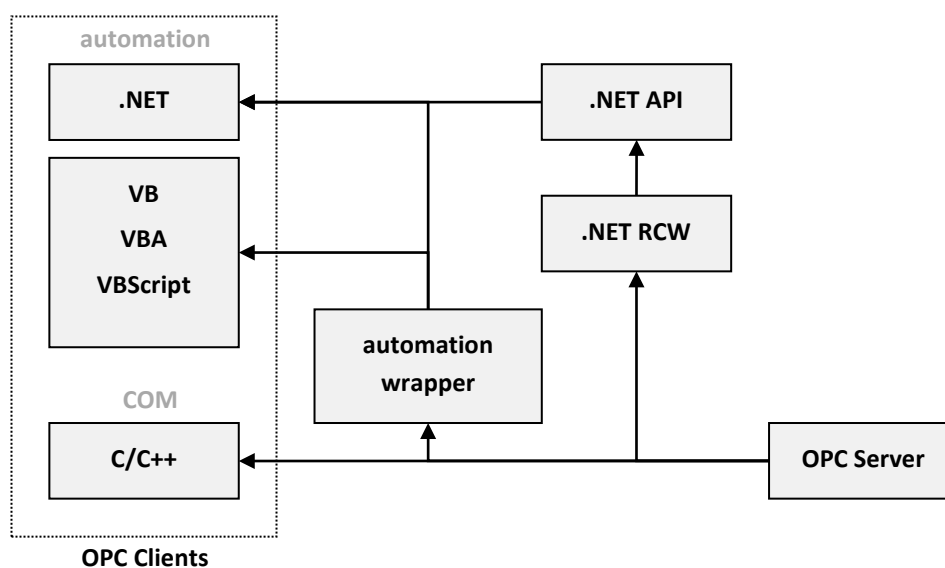
Τα groups και τα items δεν είναι προκαθορισμένα αλλά δημιουργούνται από τους πελάτες OPC ανάλογα με τις ανάγκες τους. Για παράδειγμα μια εφαρμογή HMI (Human Machine Interface) μπορεί να ορίζει ένα group για κάθε διαφορετική οθόνη, μια εφαρμογή αναφορών μπορεί να ορίζει ένα group για κάθε διαφορετική αναφορά κ.ο.κ. Επειδή όμως υπάρχουν και περιπτώσεις πελατών σε ένα σύστημα που ορίζουν τα ίδια group, το OPC προβλέπει ότι οι εξυπηρετητές μπορούν να έχουν κοινόχρηστα groups τα οποία είναι προσβάσιμα από όλους τους πελάτες. Μια άλλη υπηρεσία, που δεν είναι υποχρεωτική αλλά υποστηρίζεται από τους περισσότερους εξυπηρετητές OPC είναι η δυνατότητα εξερεύνησης των σημάτων. Η υπηρεσία αυτή επιτρέπει στους πελάτες να αναγνωρίζουν πια σήματα βλέπει ο εξυπηρετητής στον οποίο συνδέθηκαν. Επίσης οι πελάτες OPC μπορούν να ορίζουν items μόνο για εκείνα τα σημεία της διαδικασίας που θέλουν να χρησιμοποιήσουν.

Τα groups δίνουν τη δυνατότητα στους πελάτες OPC να προσπελάσουν μια ομάδα από σήματα με μία μόνο κίνηση ή να ορίσουν αυτοματοποιημένες διαδικασίες ανάγνωσης τιμών όπως για παράδειγμα χρονική επανάληψη ανάγνωσης κάθε x sec ή μόνο όταν η απόλυτη τιμή αλλάξει πάνω από ένα ποσοστό. Το OPC προβλέπει τρία είδη πρόσβασης στα σήματα ενός group:

- *Ασύγχρονη πρόσβαση για εγγραφή ή ανάγνωση.* Στην ασύγχρονη πρόσβαση ο πελάτης OPC δεν περιμένει μέχρι να ολοκληρωθεί η εγγραφή ή η ανάγνωση των δεδομένων.
- *Σύγχρονη πρόσβαση για εγγραφή ή ανάγνωση.* Ο πελάτης που δίνει εντολή για ανάγνωση ή εγγραφή δεδομένων περιμένει μέχρι να πάρει τα αποτελέσματα.
- *Ανάγνωση με ειδοποίηση από τον εξυπηρετητή.* Ο πελάτης ζητάει να ενημερώνεται κάθε φορά που αλλάζει η τιμή κάποιου σήματος ενός group ή όταν περάσει ένα ελάχιστο χρονικό διάστημα ή όταν η τιμή αλλάξει πάνω από ένα ποσοστό.

Το OPC αρχικά περιέγραφε μόνο αντικείμενα COM. Για να καταφέρουν οι προγραμματιστές να υλοποιήσουν την επικοινωνία μεταξύ πελατών και εξυπηρετητών έπρεπε να γνωρίζουν την χρήση του COM και να χρησιμοποιούν μεθόδους από τα διάφορα custom interfaces που όριζε η προδιαγραφή.

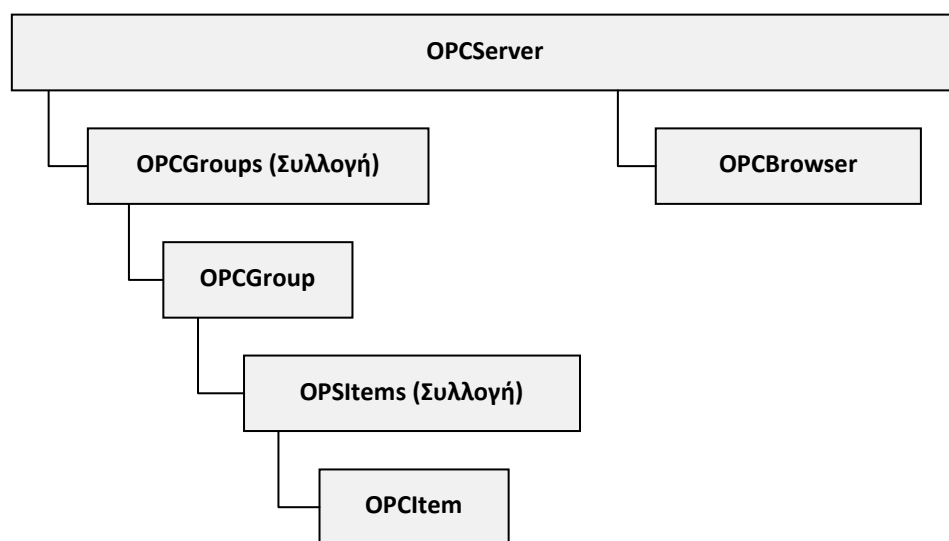
Λόγω των εξειδικευμένων γνώσεων που χρειάζονται για τον προγραμματισμό σε επίπεδο COM ο OPC Foundation ανέπτυξε το περίβλημα automation (automation wrapper) μια βιβλιοθήκη από εύχρηστα αντικείμενα τα οποία απλοποιούν σημαντικά την διαδικασία ανάπτυξης εφαρμογών automation (7).



Σχήμα 2-9 Το περίβλημα automation ανάμεσα σε πελάτες automation και εξυπηρετητές COM

Το περίβλημα automation (automation wrapper) δίνει τη δυνατότητα σε πελάτες automation όπως για παράδειγμα εφαρμογές Visual Basic, VBA και VBScript να επικοινωνούν με τους εξυπηρετητές OPC χωρίς να χρησιμοποιούν τα custom interfaces της προδιαγραφής OPC. Ωστόσο οι προγραμματιστές που είναι εξοικειωμένοι με την τεχνολογία COM μπορούν να χρησιμοποιήσουν αυτά τα custom interfaces στις εφαρμογές τους, συνήθως στο περιβάλλον C++ (1). Στην παρούσα διπλωματική εργασία θα εξετάσουμε την περίπτωση χρήσης του περιβλήματος automation μέσω της γλώσσας σεναρίων VBScript.

Τα κυριότερα αντικείμενα, που χρησιμοποιούνται κατά την πρόσβαση μέσω του automation φαίνονται στο σχήμα 2-3.



Σχήμα 2-10 Ιεραρχική δομή των αντικειμένων του OPC DA Automation Wrapper

2.4.3 Προγραμματισμός πελάτη OPC

2.4.3.1 Γενικά

Σε αυτή την ενότητα αναπτύχθηκε ένας πελάτης OPC στο Excel με χρήση της VBA (Visual Basic for Application) που συνδέεται με έναν εξυπηρετητή OPC και διαβάζει κάθε 2sec ένα σήμα που του έχει καθοριστεί εξαρχής. Η επιστροφή της τιμής γίνεται στο πρώτο κελί του πρώτου φύλλου εργασίας. Χρησιμοποιήθηκε ο εξυπηρετητής EIB OPC Version 2.0 της εταιρίας EIBA ο οποίος μέσω των βιβλιοθηκών eteC Falcon που παρέχει η ίδια εταιρία του δίνει την δυνατότητα εγγραφής και ανάγνωσης μηνυμάτων προς και από τον δίαυλο KNX/EIB (βλ. Παράρτημα Α.7).

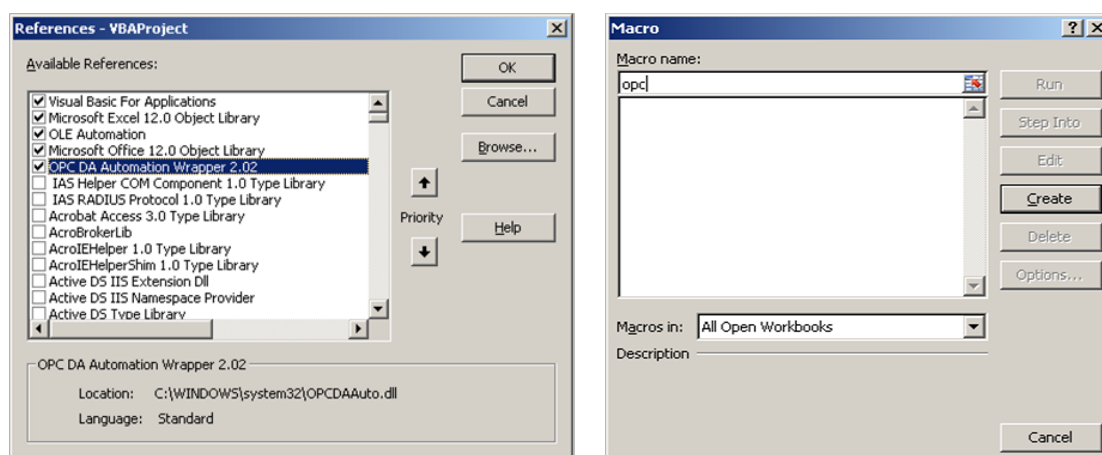
Πριν τον προγραμματισμό του πελάτη OPC δημιουργήθηκε μια λίστα από τις απαραίτητες προαπαιτούμενες ενέργειες που πρέπει να γίνουν στον κεντρικό Η/Υ ώστε να μπορεί ένα πρόγραμμα να συνδεθεί με έναν εξυπηρετητή OPC.

2.4.3.2 Προαπαιτούμενα

Για να επιτευχθεί σύνδεση του προγράμματος με τον εξυπηρετητή OPC ώστε να αποκτήσουμε πρόσβαση στα σήματα των συσκευών πεδίου θα πρέπει:

- Να έχουμε προμηθευτεί έναν εξυπηρετητή ο οποίος είτε θα είναι σε φυσική σύνδεση με τις συσκευές πεδίου μέσω της σειριακής θύρας RS-232 είτε θα τον έχουμε σε κατάσταση προσομοίωσης όπου θα δίνει τυχαίες τιμές στις διευθύνσεις που του έχουμε ορίσει.
- Να εγκατασταθεί η τελευταία έκδοση του .NET Framework που υπάρχει στο Download Center της Microsoft.
- Να εγκατασταθούν τα OPC SDK και OPC Core Component που υπάρχουν στον διαδικτυακό τόπο του OPC Foundation (8).
- Να αντικατασταθεί από το φάκελο WINDOWS/System32 το αρχείο OPCDAuto.dll V2.0.0.1 με την .NET έκδοση V2.2.5.20. Η αντικατάσταση γίνεται με το εργαλείο regsvr32.exe που αναπτύξαμε στην 2.2.6.
- Να εκτελέσουμε τον οδηγό "Using OPC via DCOM" του OPC (9) ώστε να έχουν οι εφαρμογές δικαιώματα πρόσβασης στο αντικείμενο COM του εξυπηρετητή OPC τοπικά αλλά και μέσα από το τοπικό δίκτυο.
- Από το μενού Tools του Microsoft Excel στην καρτέλα των References ενεργοποιούμε το "OLE Automation" και το "OPC DA Automation Wrapper".

Για τον έλεγχο της σωστής λειτουργίας των υπηρεσιών automation του OPC αρχικά δημιουργούμε ένα αρχείο VBScript, γράφουμε τον κώδικα της ενότητας 2.3.1 και το εκτελούμε. Αν δεν επιστραφεί κάποιο μήνυμα αποτυχίας HRESULT στη προσπάθεια δημιουργίας του αντικειμένου automation τότε το ο κεντρικός Η/Υ είναι σωστά ρυθμισμένος και έτοιμος για τον προγραμματισμό εφαρμογών OPC.



Εικόνα 2-3 Ενεργοποίηση του OPC DA Automation Wrapper 2.02 (αριστερά) και παράθυρο δημιουργίας νέου macro (δεξιά)

2.4.3.3 Περιβάλλον ανάπτυξης

Στο Microsoft Excel 2007 και συγκεκριμένα στην καρτέλα View πατάμε Macros → View Macros. Επιλέγουμε ένα κελί για παράδειγμα το A1 από το φύλλο εργασίας Sheet1, δίνουμε μία ονομασία στο πεδίο Macro name για παράδειγμα orc και έπειτα πατάμε το Create.

Ανοίγει ένα παράθυρο της Microsoft Visual Basic στο οποίο μπορούμε να ξεκινήσουμε την συγγραφή του κώδικα για τον προγραμματισμό του πελάτη OPC. Μέσα στην ρουτίνα orc() θα δημιουργηθούν τα αντικείμενα και θα οριστούν οι μεταβλητές για την πρόσβαση της εφαρμογής στο σήμα ενός αισθητήρα εσωτερικής θερμοκρασίας του εξυπηρετητή EIB OPC της εταιρίας EIBA που οδηγεί συσκευές κτιριακού αυτοματισμού συμβατές με το πρωτόκολλο KNX/EIB.

2.4.3.4 Κώδικας εφαρμογής

Με την χρήση του automation wrapper που αναπτύχθηκε από τον οργανισμό OPC είναι εξαιρετικά εύκολη η ανάπτυξη του κώδικα για την πρόσβαση στα σήματα των συσκευών πεδίου. Στην συνέχεια χωρίζουμε τον κώδικα που ακολουθεί σε πέντε τμήματα και εξετάζουμε το κάθε ένα ξεχωριστά.

Τμήμα 1: Ορίζουμε τις μεταβλητές με τα αντικείμενα του περιβλήματος automation του OPC τα οποία και θα χρησιμοποιήσουμε.

Τμήμα 2: Χρησιμοποιούμε την μέθοδο connect με παράμετρο το PROGID του εξυπηρετητή OPC για την σύνδεση του με την εφαρμογή.

Τμήμα 3: Ορίζουμε τις συλλογές AnOPCGroupCollention και AnOPCItemCollection και προσθέτουμε με τις μεθόδους Add και AddItem το group με την ονομασία "NewGroup" και το item με την ονομασία "sensors.indoor temperature.2/1/0" αντίστοιχα.

Τμήμα 4: Στο παράδειγμα της ενότητας 2.3.4 είδαμε την μέθοδο Item(1) με την οποία ανακτούμε το πρώτο item της συλλογής. Η μέθοδος Read δέχεται ως παράμετρο ένα αριθμό που δηλώνει αν η σύγχρονη ανάγνωση θα γίνει από device ή από cache. Εδώ υποθέτουμε ότι ο αριθμός αυτός έχει την τιμή 1.

Στις επόμενες σειρές το πρόγραμμα επιστρέφει την τιμή και το timestamp του σήματος στα κελιά A1 και B1 του Sheet1.

Τμήμα 5: Αποδεσμεύουμε τον χώρο μνήμης που καταλαμβάνουν οι μεταβλητές που ορίστηκαν στο τμήμα 1 και ζητάμε από την ρουτίνα αναδρομικά να επαναλάβει την εκτέλεση σε 2 sec.

```

Sub opc()
    'Τμήμα 1
    Dim handle As Long
    Dim opcServer As New OPCServer
    Dim AnOPCItem As OPCItem
    Dim AnOPCGroup As OPCGroup
    Dim AnOPCGroupCollection As OPCGroups
    Dim AnOPCItemCollection As OPCItems
    Dim Sheet1 As Worksheet
    Set Sheet1 = ThisWorkbook.Worksheets("Sheet1")
    'Τμήμα 2
    opcServer.Connect ("EIBA.EIBOPCServer.1")
    'Τμήμα 3
    Set AnOPCGroupCollection = opcServer.OPCGroups
    AnOPCGroupCollection.Add "NewGroup"
    Set AnOPCGroup = AnOPCGroupCollection.GetOPCGroup("NewGroup")
    Set AnOPCItemCollection = AnOPCGroup.OPCItems
    AnOPCItemCollection.AddItem "sensors.indoor temperature.2/1/0",
    handle
    'Τμήμα 4
    Set AnOPCItem = AnOPCItemCollection.Item(1)
    AnOPCItem.Read (1)
    Sheet1.Range("A1") = AnOPCItem.Value
    Sheet1.Range("B1") = AnOPCItem.TimeStamp
    'Τμήμα 5
    Set handle = Nothing
    Set opcServer = Nothing
    Set AnOPCItem = Nothing
    Set AnOPCGroup = Nothing
    Set AnOPCGroupCollection = Nothing
    Set AnOPCItemCollection = Nothing
    Set Sheet1 = Nothing
    Application.OnTime Now + TimeValue("00:00:02"), "opc"
End Sub

```

Κώδικας 2-1 Πελάτης OPC σε macro με χρήση VBA

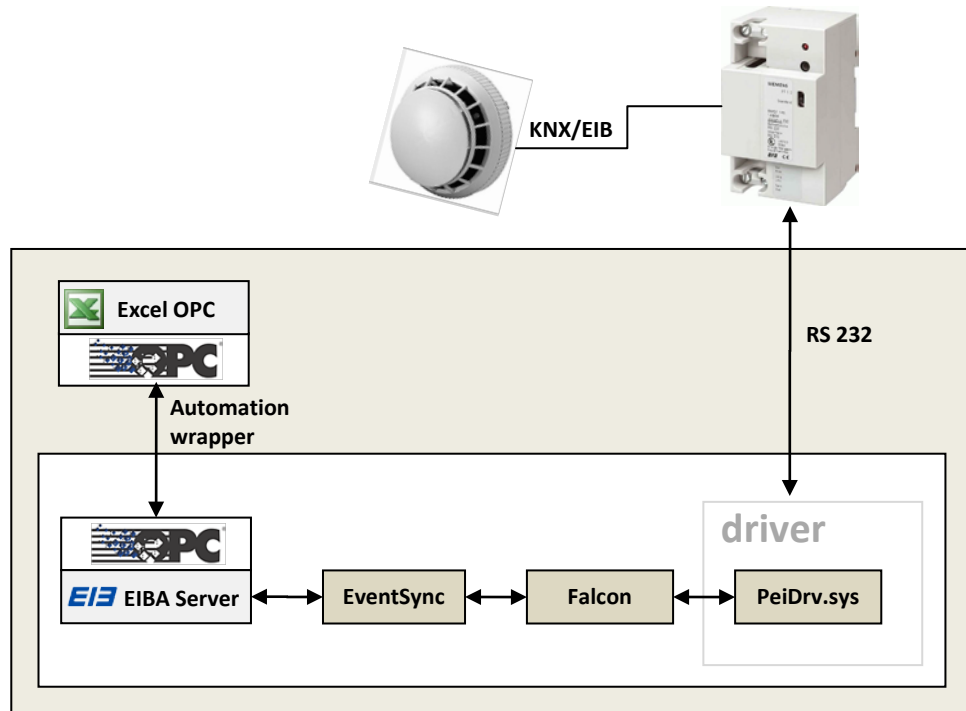
2.4.3.5 Πιθανά σφάλματα

Στην 2.4.3.2 αναφέρθηκε σαν προαπαιτούμενο η ενημέρωση του OPCDAAuto.dll στην .NET έκδοση η οποία δυστυχώς προς το παρόν είναι διαθέσιμη μόνο στα μέλη του OPC Foundation. Για την ανάπτυξη των πελατών OPC της παρούσας εργασίας καταφέραμε να εξασφαλίσουμε την ενημερωμένη .NET έκδοση. Στην πράξη φάνηκε ότι η ελεύθερη έκδοση της βιβλιοθήκης OPCDAAuto.dll έχει αρκετά προβλήματα τα οποία είναι γνωστά και έχουν διορθωθεί στις μεταγενέστερες εκδόσεις. Επίσης ένα άλλο πρόβλημα που αντιμετωπίστηκε κατά την εγκατάσταση και αρχικοποίηση του συστήματος είχε να κάνει με την αποτυχία δημιουργίας του αντικειμένου "OPC.Automation.1". Συνήθως αυτό διορθώνεται με χειροκίνητη επανεγκατάσταση

(repair) της βιβλιοθήκης orcproxy.dll που βρίσκεται στον κεντρικό φάκελο του λειτουργικού συστήματος WINDOWS/System32 με το εργαλείο regsvr32.exe (10).

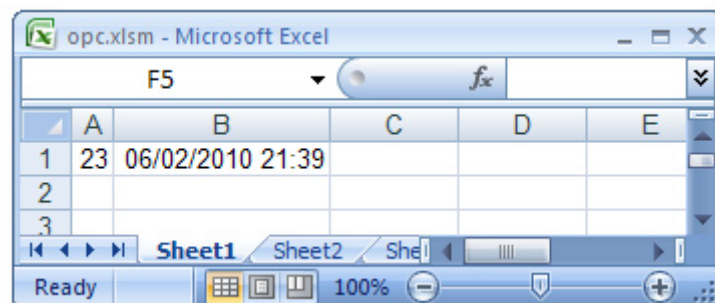
2.4.3.6 Η εφαρμογή

Στο σχήμα 2-11 φαίνονται τα στοιχεία υλικού και λογισμικού της εφαρμογής που αναπτύξαμε και οι μεταξύ τους διασυνδέσεις (11).



Σχήμα 2-11 Τα στοιχεία και οι μεταξύ τους διασυνδέσεις

Με την εκτέλεση της ρουτίνας orc() μπορούμε να δούμε ότι στο κελί A1 του Sheet1 υπάρχει η τιμή της θερμοκρασίας από το σήμα του αισθητήρα και στο κελί B1 το αντίστοιχο timestamp. Οι τιμές των κελιών ανανεώνεται κάθε 2 sec.



Εικόνα 2-4 Τρέχουσα τιμή του αισθητήρα εσωτερικής θερμοκρασίας στο Excel

3 Ανάλυση, Σχεδίαση & Ανάπτυξη Συστήματος

3.1 Εισαγωγή

Στην τελευταία ενότητα του προηγούμενου κεφαλαίου αναπτύχθηκε ένας πελάτης OPC μέσω της τεχνολογίας automation σε ένα έγγραφο του Microsoft Excel. Στόχος ήταν να υποδειχθεί η διαδικασία αρχικοποίησης υλικού και λογισμικού καθώς και να δοθεί ένα υπόδειγμα κώδικα για την ανάπτυξη τέτοιου είδους εφαρμογών.

Σε αυτό το κεφάλαιο υλοποιήθηκε μία multi-tiered εφαρμογή η οποία παρέχει στους χρήστες πρόσβαση στα σήματα ενός εξυπηρετητή OPC μέσω του διαδικτύου. Στο σύστημα αυτό οι χρήστες μπορούν να επικοινωνούν με την εφαρμογή χρησιμοποιώντας συνδέσεις ιστού μέσω του πρωτοκόλλου επικοινωνίας HTTP. Τα μηνύματα του HTTP παρέχουν ελεγχόμενη πρόσβαση στις εσωτερικές εφαρμογές του εξυπηρετητή υπηρεσιών ιστού (web server) ώστε να μπορεί να γίνει ακίνδυνα η κατανομή των διεργασιών. Ουσιαστικά οποιοδήποτε πρόγραμμα περιήγησης ιστού (web browser) μπορεί να χρησιμοποιηθεί σαν διαδικτυακός πελάτης για την πρόσβαση στα σημεία έλεγχου της διαδικασίας.

Η είσοδος στο σύστημα είναι ελεγχόμενη με λογαριασμούς χρηστών κάθε ένας από τους οποίους μπορεί να έχει διαφορετικά επίπεδα πρόσβασης. Ανάλογα με το υποδίκτυο στο οποίο ανήκουν οι υπολογιστές των χρηστών, εφαρμόζεται πολιτική ασφαλείας ώστε μερικές λειτουργίες να είναι ελεύθερες στην περίπτωση που οι χρήστες βρίσκονται με φυσικό τρόπο στον χώρο της διαδικασίας ελέγχου. Αυτό είναι πολύ χρήσιμο κυρίως σε συστήματα κτιριακού αυτοματισμού.



Εικόνα 3-1 Κεντρικό σημείο ελέγχου κτιριακού αυτοματισμού

Ο έλεγχος των συσκευών στα κτίρια γίνεται από ένα ή περισσότερα κεντρικά σημεία ελέγχου τα οποία συνήθως είναι όμοια με τις συσκευές HMI (Human Machine Interface) αποτελούμενα από οθόνες αφής. Αυτή η τεχνολογία διασύνδεσης βοηθάει σημαντικά στην απλοποίηση του χειρισμού μεγάλου πλήθους συσκευών ενώ ταυτόχρονα παρέχει δυνατότητα απεικόνισης συμβάντων, ιστορικών στοιχείων κ.τ.λ. Τα HMI είναι ολοκληρωμένα συστήματα ελέγχου τα οποία παρέχουν την δυνατότητα απεικόνισης συνθηκών λειτουργίας καθώς και την δυνατότητα παρεμβάσεων και αλλαγής των μεταβλητών της

διαδικασίας ελέγχου, ενώ σχεδόν πάντα έχουν ενσωματωμένο λειτουργικό σύστημα και ειδικές θύρες ή κάρτες που τους επιτρέπουν την απευθείας σύνδεση με τα διάφορα πρωτόκολλα δικτύων.

Το σύστημα σχεδιάστηκε και υλοποιήθηκε έτσι ώστε η διαδικτυακή εφαρμογή και τα βοηθητικά προγράμματα να εγκατασταθούν σε έναν κεντρικό Η/Υ ο οποίος παρέχει στους χρήστες την δυνατότητα εκτέλεσης χειρισμών και προβολής των καταστάσεων των σημείων ελέγχου σε δίκτυα Intranet ή στο Internet μέσω του πρωτοκόλλου επικοινωνίας HTTP, με την χρήση οποιουδήποτε προγράμματος περιήγησης ιστού. Στον κεντρικό Η/Υ εκτελούνται οι υπηρεσίες εξυπηρετητή ιστού καθώς και οι υπηρεσίες data access του OPC-Server της EIBA για την πρόσβαση στα σήματα των σημείων της διαδικασίας ελέγχου.

Επίσης ο μηχανισμός Real Time Control που αναπτύχθηκε και ενσωματώθηκε στην διαδικτυακή εφαρμογή και στο λειτουργικό σύστημα του κεντρικού Η/Υ επιτρέπει στους χρήστες την δημιουργία, τον έλεγχο εκτέλεσης και την διαχείριση αλγορίθμων, σεναρίων, ειδοποιήσεων, προειδοποιήσεων και συναγερμών. Στα πλαίσια την παρούσας εργασίας ο μηχανισμός Real Time Control δίνει την δυνατότητα αποστολής γραπτών μηνυμάτων σε κινητά τηλέφωνα μέσω SMS και σε ηλεκτρονικές διευθύνσεις μέσω Email που καθορίζονται κατά την συγγραφή των σεναρίων ελέγχου από τους ίδιους τους χρήστες.

Το σύστημα διαχειρίζεται δεδομένα που υπάρχουν αποθηκευμένα σε δύο βάσεις δεδομένων. Στην πρώτη με την ονομασία OPCWEB (system DSN) αποθηκεύονται πληροφορίες και δεδομένα που χρειάζονται για την σωστή λειτουργία της διαδικτυακής εφαρμογής (δεδομένα συσκευών πεδίου, ρυθμίσεις, λογαριασμοί χρηστών κ.τ.λ.) και είναι τεχνολογίας Microsoft SQL Server 2000, ενώ στην δεύτερη με την ονομασία OPCDATA (system DSN) αποθηκεύονται ανά τακτά χρονικά διαστήματα οι μετρήσιμες τιμές των σημείων της διαδικασίας ελέγχου με την αντίστοιχη χρονική στιγμή (timestamp) και είναι τεχνολογίας Microsoft Access 2003.

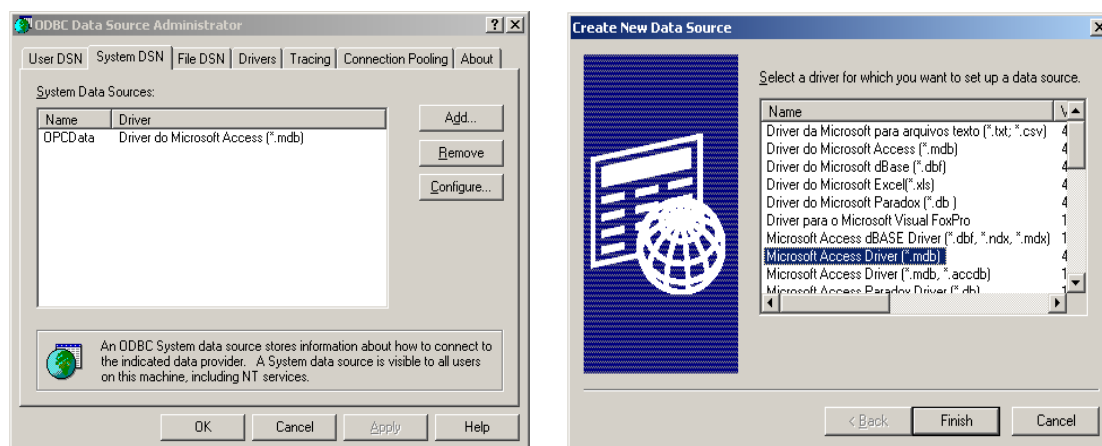
Συγκεκριμένα στα συστήματα κτιριακού αυτοματισμού το λογισμικό του κεντρικού Η/Υ που αναπτύχθηκε αποτελεί την κεντρική μονάδα ελέγχου στην οποία οι χρήστες μπορούν να συνδεθούν από οποιαδήποτε συσκευή που ενσωματώνει προγράμματα περιήγησης ιστού με ενεργοποιημένο τον μηχανισμό Javascript.

3.2 Τεχνικό Υπόβαθρο Συστήματος

3.2.1 ODBC (Open Database Connectivity)

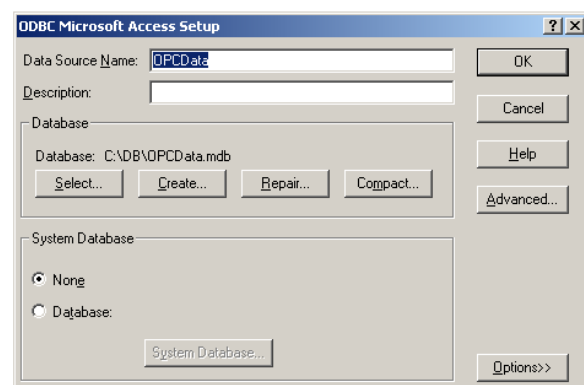
Το ODBC (Open Database Connectivity) είναι μία ανοιχτή διασύνδεση που επιτρέπει στις εφαρμογές να έχουν πρόσβαση σε βάσεις δεδομένων, οι οποίες χρησιμοποιούν την SQL ως πρότυπο για την πρόσβαση στα δεδομένα. Η σύνδεση σε βάσεις δεδομένων που υποστηρίζουν το πρότυπο ODBC γίνεται με την χρήση ειδικών drivers που παρέχουν οι κατασκευαστές. Οι ODBC drivers αποτελούν ουσιαστικά τη γέφυρα επικοινωνίας μεταξύ των εφαρμογών και των βάσεων δεδομένων.

Για να έχει δυνατότητα μία εφαρμογή να επικοινωνήσει με μία βάση δεδομένων θα πρέπει να κατασκευάσουμε ένα ODBC System DSN δίνοντας του ένα όνομα και να το συνδέσουμε με την βάση της εφαρμογής. Αυτό στην πλατφόρμα Windows Server 2003 γίνεται επιλέγοντας Start → Control Panel → ODBC Data Source Administrator.



Εικόνα 3-2 Παράθυρο ODBC και η επιλογή του κατάλληλου ODBC Driver

Στο παράθυρο που εμφανίζεται επιλέγουμε το Tab του System DSN και κάνουμε κλικ στο Add. Αν η εφαρμογή μας πρόκειται να διαχειριστεί τα δεδομένα μιας βάσης



Εικόνα 3-3 Παράθυρο εισαγωγής ονόματος DSN και επιλογής της βάσης δεδομένων

σε Access, σαν οδηγό επιλέγουμε το "Microsoft Access Driver" και πατάμε Finish.

Στην συνέχεια δίνουμε ένα όνομα DSN και πατάμε στο Select για να επιλέξουμε το αρχείο της Access από την τοποθεσία στην οποία βρίσκεται. Συνεχίζουμε πατώντας όσα OK χρειάζονται για να ολοκληρώσουμε τη διαδικασία και να κλείσουμε όλα τα παράθυρα. Για να συνδέσουμε την εφαρμογή μας με το DSN που μόλις

φτιάξαμε αρκεί να δημιουργήσουμε ένα αντικείμενο ADODB.Connection και να το ορίσουμε σε μία μεταβλητή. Για παράδειγμα στην γλώσσα σεναρίων VBScript γράφουμε:

```
Set dbConn = CreateObject("ADODB.Connection")
dbConn.Open "DSN_NAME", "DSN_USER", "DSN_PASSWORD"
'some code
dbConn.Close
Set dbConn = Nothing
```

Όπου η παράμετρος DSN_NAME είναι το όνομα DSN που δώσαμε στο system DSN. Αν η βάση δεδομένων δεν προστατεύεται με κωδικό τις παραμέτρους DSN_USER και DSN_PASSWORD δεν τις γράφουμε.

3.2.2 SQL (Structured Query Language)

Η πιο γνωστή γλώσσα που χρησιμοποιείται στα συστήματα διαχείρισης βάσεων δεδομένων είναι η SQL, της οποίας το όνομα προέρχεται από το Structured Query Language. Αρχικά η SQL ονομαζόταν SEQUEL από τα αρχικά των λέξεων Structured English Query Language και σχεδιάστηκε και υλοποιήθηκε στο ερευνητικό τμήμα της IBM ως διεπαφή για ένα πειραματικό σχεσιακό σύστημα βάσεων δεδομένων που ονομαζόταν SYSTEM-R. Σήμερα η γλώσσα SQL είναι η γλώσσα του συστήματος DB2 της IBM και του εμπορικού σχεσιακού συστήματος διαχείρισης βάσεων δεδομένων SQL/DS και αποτέλεσε, μαζί με την SEQUEL, μια από τις πρώτες υψηλού επιπέδου γλώσσες βάσεων δεδομένων.

Παραλλαγές της SQL έχουν υλοποιηθεί από τους κατασκευαστές των περισσότερων εμπορικών συστημάτων διαχείρισης βάσεων δεδομένων. Μια κοινή προσπάθεια του ANSI (American National Standards Ins.) και του ISO (International Standards Organization) οδήγησε σε μια τυποποιημένη έκδοση της SQL (ANSI 1986), που ονομάζεται SQL1. Ένα αναθεωρημένο και πολύ πιο εκτεταμένο πρότυπο έχει επίσης αναπτυχθεί με το όνομα SQL2 (αναφέρεται και ως SQL-92). Η SQL είναι μια πλήρης γλώσσα βάσεων δεδομένων. Διαθέτει εντολές για ορισμό δεδομένων, ερωτήσεις και ενημερώσεις.

Είναι ταυτόχρονα γλώσσα ορισμού δεδομένων αλλά και γλώσσα χειρισμού δεδομένων. Επιπλέον, περιλαμβάνει ευκολίες για τον ορισμό όψεων στη βάση δεδομένων και για την εμφύτευση εντολών της SQL σε γενικής χρήσης γλώσσες προγραμματισμού όπως της γλώσσας σεναρίων VBScript που χρησιμοποιείται στην παρούσα εργασία.

3.2.3 ASP (Active Server Pages)

Η τεχνολογία ASP αποτελεί την λύση της Microsoft όσον αφορά τον server side προγραμματισμό (scripting). Ο web server έχει την δυνατότητα δημιουργίας ή και μεταβολής του αρχείου HTML που θα επιστρέψει στον χρήστη. Συνεπώς σε κάθε αίτηση αρχείου με κατάληξη .asp ο web server εκτελεί το αρχείο αυτό μέσω της βιβλιοθήκης ASP.DLL η οποία επεξεργάζεται τον κώδικα. Οι σελίδες ASP μπορούν να αλληλεπιδρούν με μία βάση δεδομένων (SQL, ACCESS, ORACLE ή οποιαδήποτε συμβατή με το ODBC), ακόμα και να ορίζουν μεταβλητές οι οποίες αναφέρονται σε αντικείμενα COM.

Επίσης η ASP δίνει την δυνατότητα στον προγραμματιστή να επιλέξει την γλώσσα προγραμματισμού που θα χρησιμοποιήσει για την ανάπτυξη μίας εφαρμογής. Οι δύο πιο κοινές γλώσσες scripting για την ASP είναι η VBScript και η JScript. Είναι δυνατό στην ίδια σελίδα ASP να χρησιμοποιηθεί συνδυασμός και των δύο στην περίπτωση χρήσης κώδικα σε υπορουτίνες.

Για να μπορέσουμε να χρησιμοποιήσουμε την ASP ο web server πρέπει να είναι μία από τις διαθέσιμες εκδόσεις του IIS (Internet Information Services). Ο IIS είναι προϊόν της Microsoft και μπορεί να εκτελεσθεί στις νεότερες πλατφόρμες Windows NT. Η εγκατάσταση του γίνεται από την καρτέλα Add/Remove Programs του Control Panel στην ενότητα εγκατάστασης Πρόσθετα Στοιχεία των Windows. Υποστηρίζει υπηρεσίες WEB, FTP και SMTP. Είναι ολοκληρωμένος με το λειτουργικό σύστημα, γεγονός που του επιτρέπει να χρησιμοποιεί τον ίδιο κατάλογο χρηστών και έτσι παρέχει υψηλή ασφάλεια ελέγχοντας τα δικαιώματα που έχει κάθε χρήστης στις σελίδες των εφαρμογών. Υποστηρίζει SSL και παρέχει μια σειρά από τεχνολογίες που τον καθιστούν ικανό να ανταποκριθεί σε οποιαδήποτε απαίτηση. Για την ανάπτυξη του συστήματος επιλέχθηκε η έκδοση 6.0 η οποία περιέχεται στα Windows Server 2003.

Η ASP έχει αποδείξει ότι είναι ένα πετυχημένο και ολοκληρωμένο περιβάλλον ανάπτυξης εφαρμογών internet και intranet που επιτρέπει την ταυτόχρονη συνύπαρξη εγγράφων HTML, εντολών VBScript και Javascript, στοιχείων ActiveX καθώς και αντικειμένων COM.

3.2.3.1 Πλεονεκτήματα ASP

Τα βασικά πλεονεκτήματα που παρουσιάζει είναι τα εξής:

- Δεν απαιτεί μεταφραστή (Compiler).
- Μπορεί εύκολα να διορθωθεί και να επεκταθεί.
- Παρέχει υψηλή ασφάλεια.
- Συνεργάζεται με άλλες τεχνολογίες όπως ODBC, ActiveX, AJAX κ.τ.λ.

Δεν απαιτεί μεταφραστή (Compiler)

Η ASP δεν απαιτεί μεταφραστή διότι αποτελεί ενσωματωμένη λειτουργία του IIS οι οποίες φορτώνονται κάθε φορά μαζί του. Απλά μια βιβλιοθήκη dll επεξεργάζεται το αρχείο αυτό όταν υπάρξει κλήση.

Αποτέλεσμα αυτής της ιδιότητας είναι η αύξηση του αριθμού των ταυτόχρονων αιτήσεων στους οποίους μπορεί να ανταπεξέλθει ο IIS.

Μπορεί εύκολα να διορθωθεί και να επεκταθεί

Τα αρχεία asp είναι αρχεία κειμένου τα οποία περιέχουν κώδικα και δεν απαιτούν μεταφραστή κάθε φορά που γίνεται μια αλλαγή.

Παρέχει υψηλή ασφάλεια

Οι σελίδες ASP εκτελούνται αποκλειστικά στον IIS. Έτσι εξασφαλίζεται η απόκρυψη του κώδικα της γλώσσας προγραμματισμού που χρησιμοποιήθηκε για την ανάπτυξη της εφαρμογής.

Συνεργάζεται με άλλες τεχνολογίες

Η τεχνολογία ASP συμπληρώνουν ένα σημαντικό κομμάτι του πάζλ για να έχουμε δυνατές διαδικτυακές εφαρμογές. Σε συνδυασμό με άλλες τεχνολογίες όπως για παράδειγμα της Javascript και της XML μπορούμε να αυξήσουμε την αποδοτικότητα των εφαρμογών. Το πώς θα το εξετάσουμε αναλυτικά στην επόμενη ενότητα.

3.2.3.2 Αντικείμενα ASP

Η τεχνολογία ASP παρέχει τέσσερα αντικείμενα. Κάθε ένα από αυτά έχουν μία σειρά από μεθόδους και ιδιότητες που δίνουν πολλές δυνατότητες.

Τα αντικείμενα αυτά είναι τα εξής:

- *Request*. Χρησιμοποιείται για την αποστολή δεδομένων από το πρόγραμμα περιήγησης στον εξυπηρετητή IIS. Αυτό συνήθως γίνεται μέσα από φόρμες με τις δύο μεθόδους POST και GET.
- *Response*. Το αντίθετο αντικείμενο του Request είναι το Response και χρησιμοποιείται στις περιπτώσεις που ο εξυπηρετητής στέλνει πληροφορίες στο πρόγραμμα περιήγησης ιστού.
- *Session & Application*. Ένα από τα γνωστά μειονεκτήματα του πρωτοκόλλου HTTP, είναι ότι δεν διατηρεί την σύνδεση μεταξύ πελάτη και εξυπηρετητή κατά την διάρκεια της επικοινωνίας αλλά την διακόπτει κάθε φορά που ολοκληρώνεται η αποστολή δεδομένων από την μία μεριά στην άλλη. Η ASP δίνει την δυνατότητα μέσω του αντικειμένου session να δημιουργηθούν μεταβλητές στον server που να κρατάνε πληροφορίες για την σύνδεση και

δεδομένα τα οποία θα διαγραφούν όταν ο χρήστης θα τερματίσει την σύνδεση. Ένα παράδειγμα χρήσης του αντικειμένου session είναι ότι δίνει την δυνατότητα σε μία διαδικτυακή εφαρμογή να «θυμάται» ότι ο συνδεδεμένος χρήστης έχει περάσει με επιτυχία τον έλεγχο πρόσβασης σε μία προστατευμένη περιοχή δυναμικών σελίδων. Από την άλλη μεριά το αντικείμενο application περιέχει δεδομένα τα οποία μπορούν να τα μοιράζονται όλοι οι χρήστες που έχουν συνδεθεί. Ένα συνηθισμένο παράδειγμα χρήσης του application είναι ο υπολογισμός του αριθμού των ενεργών συνδέσεων σε μία διαδικτυακή εφαρμογή μέσω του global.asa

3.2.3.3 Παράδειγμα εφαρμογής ASP

Για να γίνει κατανοητός ο τρόπος προγραμματισμού διαδικτυακών εφαρμογών αναπτύχθηκε ένα πρόγραμμα σε μία σελίδα ASP με κώδικα VBScript, η οποία συνδέεται με μία βάση δεδομένων και μας επιστρέφει μία λίστα με τους αισθητήρες που ανήκουν σε ένα συγκεκριμένο δωμάτιο ενός συστήματος κτιριακού αυτοματισμού το αναγνωριστικό του οποίου το παίρνει σαν παράμετρο εισόδου.

Ο τρόπος με τον οποίο αποστέλλεται στον web server το αναγνωριστικό του δωματίου είναι με την μέθοδο GET στην γραμμή διευθύνσεων του προγράμματος περιήγησης. Δηλαδή:

`http://localhost/sensorsList.asp?Room=LivingRoom`

μέσω της οποίας ζητάμε από τον web server να επιστρέψει τους αισθητήρες του δωματίου με το αναγνωριστικό LivingRoom.

Προαπαιτούμενα

Η υλοποίηση του παραδείγματος προϋποθέτει ένα πίνακα σε μία βάση δεδομένων στην οποία υπάρχουν αποθηκευμένα τα δεδομένα των αισθητήρων. Το όνομα του είναι tbl_sensors και η δομή του φαίνεται στον πίνακα 3-1.

Πίνακας 3-1 Η δομή του tbl_sensors

tbl_sensors	
Id	Long integer, auto number, primary key
Name	Text 255 characters max
Room	Text 255 characters max

Στην συνέχεια προχωρήσαμε στην δημιουργία ενός system DSN με την ονομασία SENSORS βάση της διαδικασίας που αναπτύχθηκε στην 3.2.1 ώστε το πρόγραμμα να έχει την δυνατότητα να αντλεί δεδομένα από την βάση.

Δημιουργούμε ένα νέο αρχείο με κατάληξη .asp μέσα στον φάκελο wwwroot του IIS και του δίνουμε την ονομασία sensorsList.asp

```
<%@ LANGUAGE="VBSCRIPT" CODEPAGE="65001"%>
<html>
<head>
    <title>Sensors List</title>
</head>
<body>
<%
    Set dbConn = CreateObject("ADODB.Connection")
    Set Rs = CreateObject("ADODB.Recordset")
    Room = Request.QueryString("Room")
    dbConn.Open "SENSORS"
    SQL = "SELECT * FROM tbl_sensors WHERE Room='" & Room & "'"
    Rs.Open SQL, dbConn, 1, 3
    If Rs.RecordCount > 0 Then
        Do Until Rs.EOF
            Response.Write Rs("Name") & ", "
            Rs.MoveNext
        Loop
    End If
    Rs.Close
    dbConn.Close
    Set Rs = Nothing
    Set dbConn = Nothing
%>
</body>
</html>
```

Κώδικας 3-1 Σελίδα sensorsList.asp

Για να ελεγχθεί η σωστή λειτουργία του IIS μετά την εγκατάσταση του, ανοίγουμε την τοποθεσία <http://localhost/> από ένα πρόγραμμα περιήγησης ιστού. Αν όλα είναι εντάξει θα δούμε σχετικό μήνυμα καλωσορίσματος. Ο πίνακας `tbl_sensors` της βάσης δεδομένων περιέχει τυχαία δεδομένα όπως φαίνονται στον πίνακα 3-2.

Πίνακας 3-2 Τυχαία δεδομένα στον `tbl_sensors`

Id	Name	Room
1	Θερμοκρασία LV	LivingRoom
2	Θερμοκρασία	Bedroom
3	Φωτεινότητα	Bedroom
4	Φωτεινότητα LV	LivingRoom
5	Υγρασία	Kitchen

Έτσι αν δώσουμε την διεύθυνση <http://localhost/sensorsList.asp?Room=LivingRoom> στην γραμμή διευθύνσεων του προγράμματος περιήγησης η σελίδα ASP θα μας εκτυπώσει στην οθόνη, τις ονομασίες των αισθητήρων που έχουν στο πεδίο `Room` την τιμή `LivingRoom`. Το αποτέλεσμα της σελίδας θα είναι:

Θερμοκρασία LV, Φωτεινότητα LV

Στο παράδειγμα αυτό είδαμε σε γενικές γραμμές την δομή μίας σελίδας ASP και πως μπορεί αυτή να συνδεθεί με μία βάση δεδομένων που είναι συμβατή με το ODBC. Για να εκτελεστεί αυτός ο κώδικας, δόθηκε ένα αίτημα από τον χρήστη μέσω του οποίου δημιουργήθηκε η απόκριση από τον IIS. Για την αλληλεπίδραση του προγράμματος με τον χρήστη χρειάστηκε η ανανέωση της σελίδας ώστε να επιστραφούν τα αποτελέσματα.

Στην επόμενη ενότητα θα ενσωματωθεί στο παράδειγμα αυτό η τεχνολογία AJAX ώστε να μην χρειάζεται η ανανέωση της σελίδας σε κάθε αίτημα του χρήστη.

3.2.4 AJAX (Asynchronous Javascript And XML)

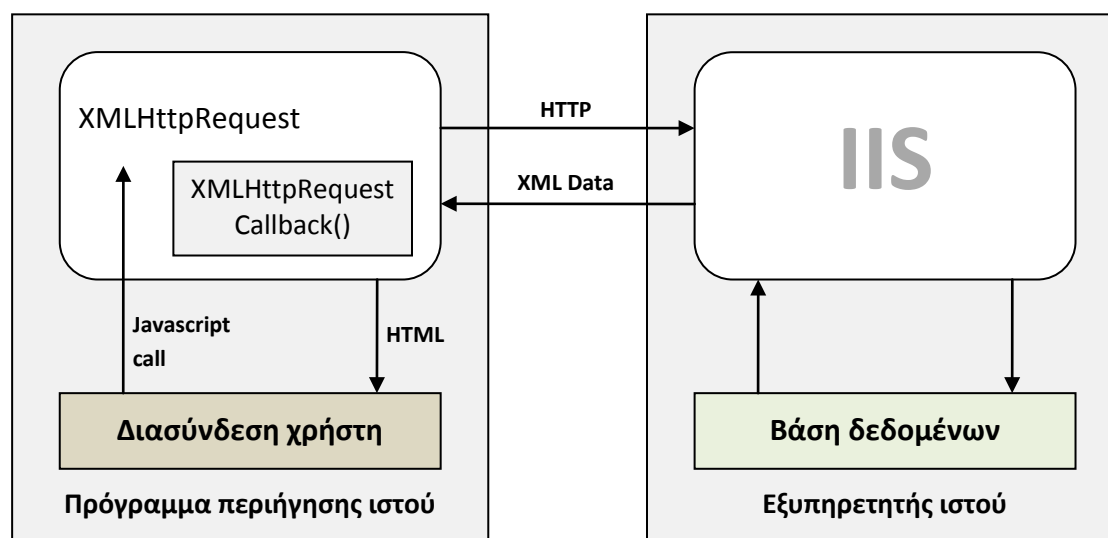
Οι διαδικτυακές εφαρμογές εμπλουτισμένων μέσων RIA (Rich Internet Applications) είναι εφαρμογές ιστού που προσεγγίζουν στην όψη και στην αισθητική τις εφαρμογές επιφάνειας εργασίας. Οι εφαρμογές αυτές έχουν δύο βασικά χαρακτηριστικά την υψηλή αποδοτικότητα και την αξιοπιστία.

Η αποδοτικότητα τους προέρχεται από την τεχνολογία AJAX η οποία χρησιμοποιεί προγράμματα σεναρίων στον πελάτη για να δώσει στην εφαρμογή μικρότερους χρόνους απόκρισης. Οι εφαρμογές RIA διαχωρίζουν την αλληλεπίδραση της πλευράς του πελάτη από την επικοινωνία με τον web server και τις εκτελούν παράλληλα, μειώνοντας τις καθυστερήσεις απόκρισης του εξυπηρετητή που υπομένει ο πελάτης (12).

Η AJAX χρησιμοποιεί την γλώσσα σεναρίων Javascript για την αποστολή ασύγχρονων αιτημάτων προς τον web server ενός συστήματος με απευθείας χρήση του αντικειμένου XMLHttpRequest. Κατόπιν μέσω της ιεραρχίας αντικειμένων του DOM (Document Object Model) ενημερώνεται το περιεχόμενο.

Σε γενικές γραμμές σκοπός της είναι η ασύγχρονη φόρτωση περιεχομένου των ιστοσελίδων στο πρόγραμμα περιήγησης. Για να επιτευχθεί αυτός ο σκοπός η AJAX κάνει χρήση των ακόλουθων τεχνολογιών:

- Της γλώσσας επισήμανσης ιστοσελίδων XHTML και μορφοποίησης CSS.
- Του DOM για την δυναμική προβολή και την αλληλεπίδραση των αντικειμένων.
- Της γλώσσας επισήμανσης XML (eXtensible Markup Language) και των μετασχηματισμών XSLT (XSL Transformations) για την ανταλλαγή και την διαχείριση των δεδομένων αντίστοιχα.
- Της γλώσσα σεναρίων Javascript και των αντικειμένων XMLHttpRequest() ActiveXObject("Msxml2.XMLHTTP") και ActiveXObject("Microsoft.XMLHTTP")



Σχήμα 3-1 Ασύγχρονη επικοινωνία διαδικτυακής εφαρμογής με χρήση της AJAX

Στο σχήμα 3-1 φαίνεται η διαδικασία της ασύγχρονης επικοινωνίας μίας διαδικτυακής εφαρμογής με τον IIS η οποία με κλήση της Javascript στέλνει ένα αίτημα ανάκτησης δεδομένων στον εξυπηρετητή. Όταν ο εξυπηρετητής ολοκληρώσει την διεργασία και έχει έτοιμη την απόκριση θα την επιστρέψει σε καθορισμένο τμήμα της διαδικτυακής εφαρμογής. Για να γίνει κατανοητός ο τρόπος με τον οποίο οι προγραμματιστές ενσωματώνουν την τεχνολογία AJAX στις διαδικτυακές εφαρμογές τους θα εμπλουτίσουμε το παράδειγμα της προηγούμενης ενότητας ώστε ένα αίτημα και η απόκριση του να γίνονται μέσω του αντικειμένου XMLHttpRequest().

3.2.4.1 Παράδειγμα χρήσης του αντικειμένου XMLHttpRequest()

Για την υλοποίηση του παραδείγματος δημιουργήθηκε μία νέα απλή στατική σελίδα HTML. Η σελίδα αυτή χρησιμοποιήθηκε για την καταγραφή του συμβάντος εκείνου, το οποίο όταν συμβεί η Javascript πυροδοτεί την αποστολή του αιτήματος στον IIS για την ανάκτηση δεδομένων από τον πίνακα tbl_sensors. Το συμβάν που χρησιμοποιήθηκε είναι το onKeyPress στο πεδίο κειμένου sensorName της φόρμας frm0. Καθώς αρχίζει η πληκτρολόγηση των χαρακτήρων η εφαρμογή ψάχνει μετά από κάθε πάτημα να βρει από τον πίνακα tbl_sensors τις καταχωρίσεις που ταιριάζουν με το αλφαριθμητικό και να τις επιστρέψει στην προκαθορισμένη περιοχή pData.

Δημιουργήθηκε ένα νέο αρχείο με κατάληξη .html μέσα στον φάκελο wwwroot και του δόθηκε η ονομασία sensors. Ο κώδικας της σελίδας φαίνεται στην συνέχεια με την συνάρτηση getSnsList() η οποία αναλαμβάνει την αποστολή του ασύγχρονου αιτήματος καθώς και την επιστροφή των αποτελεσμάτων σε προκαθορισμένη περιοχή της σελίδας.

```
<html>
<head>
<title>Sensors</title>
<script language="javascript">
function getSnsList(Room){
    var url = "sensorsList.asp?Room=" + Room;
    var xmlHttp;
    try{
        xmlHttp = new XMLHttpRequest();
    }
    catch (e){
        return false;
    }
    xmlHttp.onreadystatechange=function(){
        document.getElementById("pData").innerHTML+xmlHttp.responseText;
    }
    xmlHttp.open("GET", url, true);
    xmlHttp.send(null);
}
</script>
</head>
<body>
<form name="frm0">
<input type="text" id="sensorName" onKeyPress="getSnsList(this.value);">
</form>
<div id="pData"></div>
</body>
</html>
```

Κώδικας 3-2 Σελίδα sensors.html

Στην συνέχεια τροποποιήθηκε ο Κώδικας 3-1 της σελίδας sensorsList.asp έτσι ώστε να μην εκτυπώνει τις ετικέτες html, body κ.τ.λ. επειδή το περιεχόμενο της τώρα θα επιστρέφεται μέσα στην sensors.html η οποία περιλαμβάνει εξ αρχής αυτές τις ετικέτες.

Επίσης τροποποιήθηκε η εντολή SQL ώστε να επιστρέφονται οι καταχωρίσεις του πίνακα που αρχικά ταιριάζουν με την αριστερή πλευρά των αλφαριθμητικών τιμών του πεδίου Room. Με κόκκινο και έντονη γραφή εμφανίζονται οι προσθήκες στον Κώδικα 3-3.

Η μέθοδος ExpiresAbsolute της ASP χρησιμοποιείται για να δηλώσει ότι το περιεχόμενο της συγκεκριμένης σελίδας λήγει αμέσως μετά την εκτύπωση και για τον λόγο αυτό δεν θα πρέπει να αποθηκεύεται σαν προσωρινό αρχείο στον ειδικό φάκελο των προγραμμάτων περιήγησης ιστού.

Η συγκεκριμένη έκδοση της συνάρτησης getSnsList() αποτελεί ένα απλοποιημένο παράδειγμα του μηχανισμού αποστολής ασύγχρονου μηνύματος προς τους εξυπηρετητές ιστού και λειτουργεί μόνο στις σύγχρονες εκδόσεις του ΙΕ. Για την συμβατότητα του κώδικα με τα περισσότερα σύγχρονα προγράμματα περιήγησης

ιστού (Mozilla, Chrome και Safari) θα χρειαστεί να την εμπλουτίσουμε με αναδρομική χρήση των σύνθετων χειριστών σφαλμάτων Try...Catch ώστε στην περίπτωση που αποτυγχάνει η δημιουργία του XMLHttpRequest() να προσπαθεί την δημιουργία άλλων αντικειμένων όπως το ActiveXObject("Msxml2.XMLHTTP") ή και το ActiveXObject("Microsoft.XMLHTTP") (βλ. Παράρτημα Α.8).

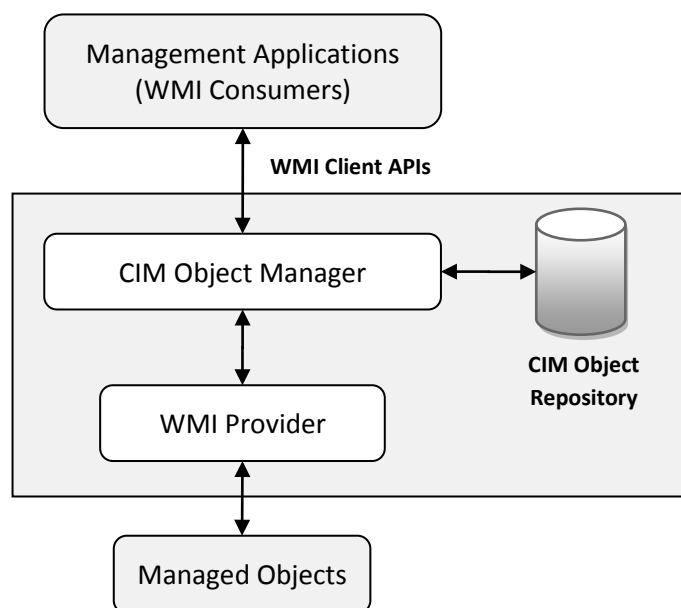
```
<%@ LANGUAGE="VBSCRIPT" CODEPAGE="65001"%>
<%
    Response.ExpiresAbsolute = Now() - 1
    Set dbConn = CreateObject("ADODB.Connection")
    Set Rs = CreateObject("ADODB.Recordset")
    Room = Request.QueryString("Room")
    dbConn.Open "SENSORS"
    SQL = "SELECT * FROM tbl_sensors WHERE Room Like '%" & Room & "'"
    Rs.Open SQL, dbConn, 1, 3
    If Rs.RecordCount > 0 Then
        Do Until Rs.EOF
            Response.Write Rs("Name") & ", "
            Rs.MoveNext
        Loop
    End If
    Rs.Close
    dbConn.Close
    Set Rs = Nothing
    Set dbConn = Nothing
%>
```

Κώδικας 3-3 Τροποποιημένη σελίδα sensorsList.asp

3.2.5 WMI (Windows Management Instrumentation)

Το WMI είναι ένα σύνολο επεκτάσεων του WDM (Windows Driver Model) που παρέχει μια διεπαφή στα λειτουργικά συστήματα της Microsoft. Μέσω της τεχνολογίας του WMI καθίσταται δυνατή η παρατήρηση και η εξαγωγή πληροφορίας από τα Windows. Πρόκειται για μία αρθρωτή διαχείριση συστήματος που χρησιμοποιεί μια αντικειμενοστραφή λογική για την εξαγωγή των πληροφοριών. Το WMI είναι μια υλοποίηση της Microsoft του WBEM (Web-Based Enterprise Management) και του CIM (Common Information Model) υπό τις προδιαγραφές του DMTF (Distributed Management Task Force) (13).

Στην πράξη, το WMI επιτρέπει σε γλώσσες προγραμματισμού καθώς και σε γλώσσες σεναρίων όπως για παράδειγμα της VBScript, να διαχειριστούν υπολογιστές σε δίκτυα της Microsoft. Το WMI είναι προεγκατεστημένο και παρέχει πρόσβαση και έλεγχο σε στοιχεία του λειτουργικού συστήματος μέσω API εφαρμογών, δεσμών ενεργειών και γραμμής εντολών καθώς και την διαχείριση τοπικών και απομακρυσμένων συστημάτων. Η αρχιτεκτονική του WMI φαίνεται στο σχήμα 3-2.



Σχήμα 3-2 Τριών επιπέδων αρχιτεκτονική του WMI

Μία εφαρμογή χρήστη μέσω της τεχνολογίας WMI API εξάγει πληροφορία από το WMI core. Η διαδικασία γίνεται λειτουργική με την χρήση των WMI παρόχων (WMI Providers). Ένας WMI Provider είναι ένα λογικό αντικείμενο το οποίο επιτελεί την λειτουργία της παρακολούθησης και διαχείρισης διαφόρων λογικών ή φυσικών τμημάτων ενός υπολογιστή. Είναι η οντότητα εκείνη η οποία ανακτά τις απαραίτητες πληροφορίες από τα επιμέρους συστήματα του υπολογιστή ή του δικτύου. Η εφαρμογή του χρήστη αναλαμβάνει τον ρόλο του WMI καταναλωτή (WMI Consumer).

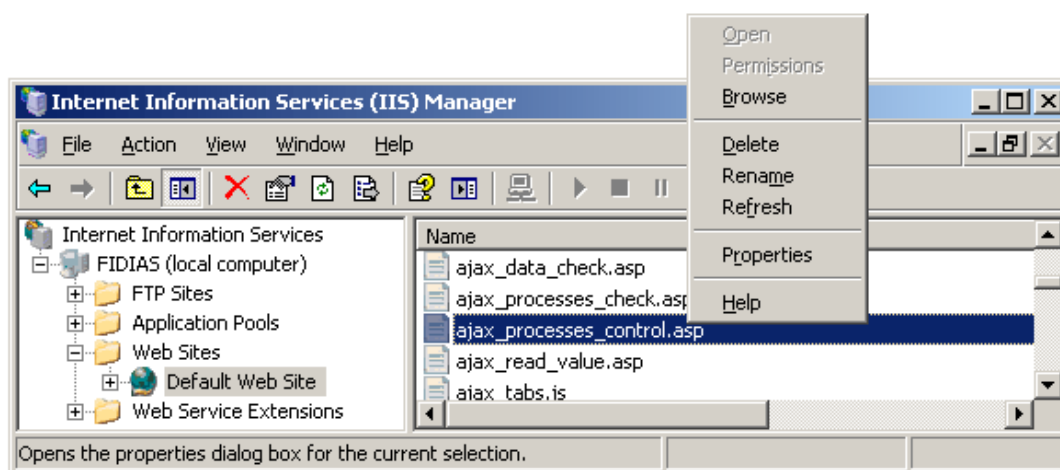
Για να γίνει κατανοητή η λειτουργία του μηχανισμού αναπτύχθηκε ένα πρόγραμμα σε VBScript όπου με την χρήση του WMI κλείνει μία διεργασία από την καρτέλα Processes του Task Manager. Η διεργασία που επιλέχθηκε για το παράδειγμα αυτό είναι ο EIB OPC Server της EIBA (βλ. Παράρτημα Α.7). Σε ένα αρχείο με κατάληξη .vbs γράφουμε:

```
Set objWMI =
GetObject("winmgmts:{impersonationLevel=impersonate}!\root\cimv2")
Set colProcessList =
objWMI.ExecQuery("SELECT * FROM Win32_Process WHERE Name = 'EIBSrvr.exe'")
For Each objProcess in colProcessList
    objProcess.Terminate()
Next
Set objWMI = Nothing
Set colProcessList = Nothing
```

Κώδικας 3-4 Παράδειγμα χρήσης του WMI για τον τερματισμό μίας διεργασίας

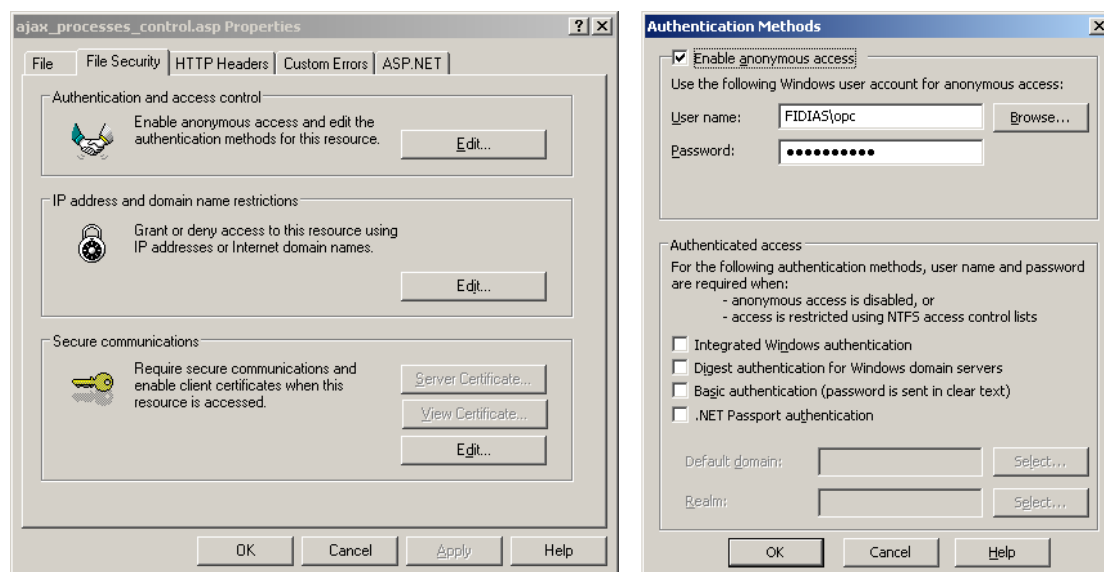
Με την προϋπόθεση ότι ο OPC-Server λειτουργεί, εκτελούμε τον κώδικα απευθείας με διπλό κλικ πάνω στο αρχείο. Παρατηρούμε αμέσως ότι ο OPC-Server τερματίζεται. Βασική προϋπόθεση για την επιτυχή εκτέλεση του script είναι ότι πρέπει να γίνεται από τον ίδιο χρήστη στον λογαριασμό του οποίου τρέχει η εφαρμογή.

Υπάρχει επίσης η δυνατότητα εκτέλεσης τέτοιου είδους κώδικα από τον web server δηλαδή μια δυναμική σελίδα τεχνολογίας ASP να ελέγξει μια εφαρμογή στον υπολογιστή μέσω του IIS. Για να γίνει αυτό εφικτό πρέπει να τροποποιηθούν τα δικαιώματα της σελίδας αυτής ώστε να εκτελείται σε ανώνυμη πρόσβαση, αλλά όχι από τον προεπιλεγμένο χρήστη του IIS, δηλαδή τον ISUR_computername αλλά από τον χρήστη κάτω από τον οποίο τρέχει η εφαρμογή που θέλουμε να ελέγχουμε. Στον IIS επιλέγουμε την σελίδα που περιέχει τον κώδικα WMI, δεξί κλικ και πατάμε Properties.



Εικόνα 3-4 Η επιλογή Properties μίας σελίδας στον IIS 6.0

Στο δεύτερο Tab δηλαδή το File Security πατάμε Edit στο "Authentication and access control" και στην συνέχεια διαγράφουμε τον χρήστη ISUR_computername και επιλέγουμε αυτόν στον οποίο εκτελείται η εφαρμογή που θέλουμε να ελέγξουμε.



Εικόνα 3-5 Ρυθμίσεις ασφαλείας σελίδων ASP που ενσωματώνουν WMI

Συνεχίζουμε πατώντας όσα OK χρειάζονται για να ολοκληρώσουμε τη διαδικασία και να κλείσουμε όλα τα παράθυρα. Με αυτές τις ρυθμίσεις μία δυναμική σελίδα που εμπεριέχει την τεχνολογία WMI μπορεί να αποκτήσει τον έλεγχο σε στοιχεία του λειτουργικού συστήματος (14).

3.2.5.1 PsExec

Για λόγους ασφαλείας το WMI δεν επιτρέπει την απομακρυσμένη εκκίνηση στο console session, των προγραμμάτων εκείνων που αλληλεπιδρούν με το χρήστη μέσω παραθύρων διαλόγου. Για να επιλύσουμε αυτό το πρόβλημα χρησιμοποιήσαμε το εργαλείο PsExec της εταιρίας Sysinternals (15).

Έτσι όταν θέλουμε να εκκινήσουμε ένα πρόγραμμα όπως για παράδειγμα τον OPC Server της EIBA, χρησιμοποιούμε το WMI για να εκκινήσει το PsExec το οποίο με την σειρά του εκκινεί τον OPC-Server. Για την εκκίνηση του PsExec μέσω της υπηρεσίας WMI σε VBScript γράφουμε:

```
Set objWMI = GetObject("winmgmts:\\.\root\cimv2:Win32_Process")
objWMI.Create "PSEXEC -d -u fidias\opc -p **** -i 0 "C:\Program Files\EIB
OPC-Server\EIBServr.exe" ", null, null, sintProcessID
Set objWMI = Nothing
```

Όπου οι παράμετροι -u και -p είναι τα στοιχεία του λογαριασμού που εκτελεί την εφαρμογή και το -i αναφέρεται στο session, όπου 0 είναι το console session.

3.2.6 WSH (Windows Script Host)

Το WSH (Windows Script Host) είναι μία τεχνολογία αυτοματισμού για τα λειτουργικά συστήματα Microsoft Windows η οποία παρέχει δέσμες ενεργειών με δυνατότητες συγκρίσιμες των batch files, αλλά με μεγαλύτερο φάσμα υποστηριζόμενων χαρακτηριστικών. Μπορεί να κάνει χρήση των active scripting μηχανισμών όπως της JScript και της VBScript ενώ τρέχει απλά αρχεία κειμένου με καταλήξεις .vbs, .js και .wsf (windows script file). Το πλεονέκτημα του .wsf είναι ότι επιτρέπει στον προγραμματιστή να χρησιμοποιήσει ένα συνδυασμό γλωσσών σεναρίων μέσα σε ένα ενιαίο αρχείο (16).

Ένα παράδειγμα χρήσης του WSH είναι όταν χρειαστεί να γραφτεί μια τιμή στο μητρώο των Windows. Με χρήση της VBScript γράφουμε:

```
Dim WSHShell
Set WSHShell = CreateObject("WScript.Shell")
WSHShell.RegWrite strKey, val
Set WSHShell = Nothing
```

Όπου η μεταβλητή strKey περιέχει την περιοχή και την ονομασία της εγγραφής και το val την τιμή η οποία θα αποθηκευτεί (βλ. Παράρτημα Α.6).

3.3 Αρχιτεκτονική υλικού

Το σύστημα που αναπτύχθηκε, σχεδιάστηκε έτσι ώστε όλες οι υπηρεσίες να φιλοξενούνται σε έναν κεντρικό ηλεκτρονικό υπολογιστή, ο οποίος παρέχει στους χρήστες την δυνατότητα εκτέλεσης χειρισμών και προβολής των καταστάσεων των σημείων ελέγχου σε δίκτυα Intranet ή στο Internet μέσω του πρωτοκόλλου επικοινωνίας HTTP. Επίσης με εσωτερικούς μηχανισμούς αυτοματοποίησης λογισμικού ο κεντρικός εξυπηρετητής καταγράφει τα συμβάντα και τις τιμές των σημείων ελέγχου στις βάσεις δεδομένων που φιλοξενεί. Βασική προϋπόθεση και πρωτεύων στόχος στο σχεδιασμό, είναι η συνεχή λειτουργία και η σταθερότητα του μηχανήματος και για το λόγο αυτό πρέπει να το εφοδιάσουμε με συσκευή παροχής εναλλασσόμενου ρεύματος σε περίπτωση διακοπής (UPS) και προστασίας από υπέρ-ανύψωση τάσης.

Επίσης η μητρική κάρτα του εξυπηρετητή πρέπει να έχει εξειδικευμένες ρυθμίσεις, όπως αυτόματη επανεκκίνηση μετά από διακοπή ρεύματος, και υποστήριξη της τεχνολογίας RAID 1 με εγκατεστημένους τουλάχιστον δύο σκληρούς δίσκους ώστε να αποφευχθεί η κατάρρευση του συστήματος από τυχόν αποτυχία του ενός εκ των δύο.

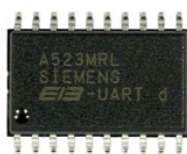
Ανάλογα με το πρωτόκολλο επικοινωνίας συσκευών με το οποίο σχεδιάζουμε να συνδέσουμε τον εξυπηρετητή (EIBus, Profibus, CANBus κ.τ.λ.) θα χρειαστεί να

προμηθευτούμε μία συσκευή διασύνδεσης σε μορφή εσωτερικής ή εξωτερικής κάρτας για την φυσική τους σύνδεση. Αν ο εξυπηρετητής εγκατασταθεί κοντά στον χώρο του πίνακα ελέγχου των συσκευών μπορεί να χρησιμοποιηθεί μία συσκευή διασύνδεσης τοποθετημένη σε ράγα (DINRAIL).

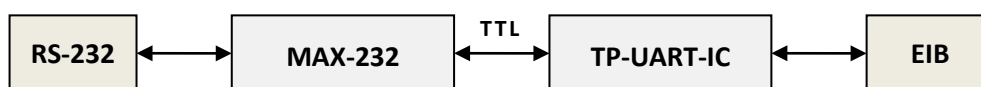
Οι κάρτες αυτές ουσιαστικά μετατρέπουν το σήμα της θύρας του υπολογιστή με τον οποίο συνδέονται (RS-232, USB κ.τ.λ.) σε σήμα του πρωτόκολλου επικοινωνίας των συσκευών αυτοματισμού.

Για παράδειγμα στην περίπτωση φυσικής σύνδεσης του εξυπηρετητή με τον δίαυλο KNX/EIB μέσω της θύρας RS-232 χρειάζονται δύο ολοκληρωμένα:

- Το πρώτο ένα είναι το γνωστό MAX-232 το οποίο στην ουσία είναι ένας μετατροπέας των σημάτων της RS-232 σε TTL τάσεις 0-5V ώστε να μπορούν να επικοινωνούν με τον υπολογιστή μικρό-ελεγκτές και ολοκληρωμένα που λειτουργούν στις τάσεις αυτές.
- Το δεύτερο είναι το EIB-TP-UART-IC (Twisted Pair – Universal Asynchronous Receive Transmit - IC) της Siemens το οποίο υποστηρίζει την επικοινωνία μεταξύ των μικρό-ελεγκτών των αισθητήρων ή των ενεργοποιητών (σήμα TTL) με το δίαυλο KNX/EIB (11).



Στο σχήμα 3-3 φαίνεται η διαδικασία μεταφοράς του σήματος από την RS-232 στο KNX/EIB και αντίστροφα. Τόσο το ολοκληρωμένο TP-UART όσο και το MAX-232 προϋποθέτουν την ύπαρξη κυκλωμάτων υποστήριξης για να λειτουργήσουν. Υπάρχουν εταιρίες που κατασκευάζουν τέτοια κυκλώματα τα οποία τα προσφέρουν σε αρκετά χαμηλές τιμές.

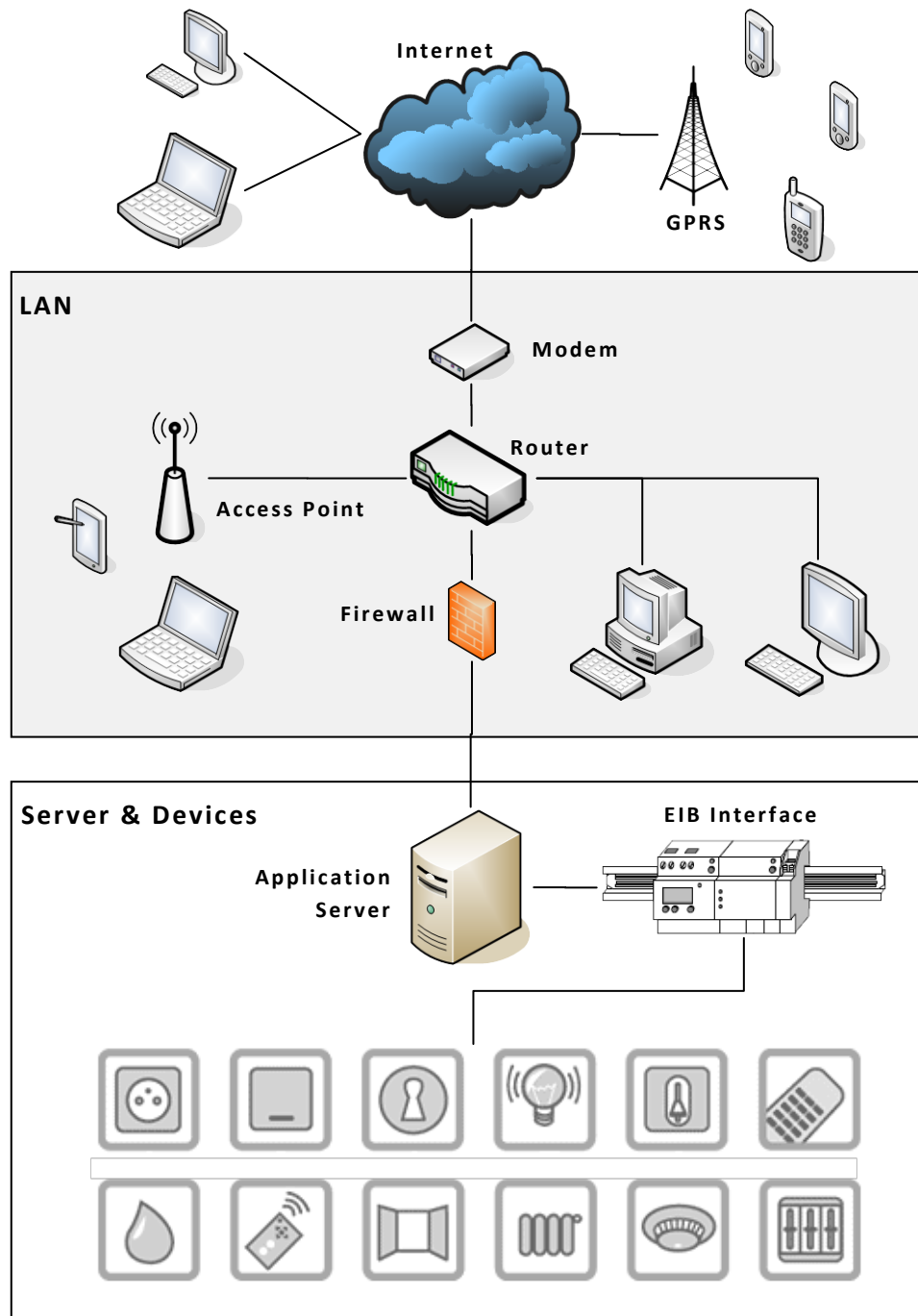


Σχήμα 3-3 Μετατροπή σήματος από RS-232 σε EIB και αντίστροφα

Ο κεντρικός εξυπηρετητής πρέπει να διαθέτει υποχρεωτικά μια κάρτα δικτύου Ethernet και μόνιμη σύνδεση με το δίκτυο LAN ή το Internet ώστε να είναι εφικτή η απομακρυσμένη επικοινωνία μέσω του πρωτοκόλλου HTTP.

Επίσης μπορεί να χρησιμοποιηθεί οποιοδήποτε software ή hardware firewall αρκεί να εξαιρεθεί η πόρτα 80 καθώς και άλλες πόρτες που πιθανόν να χρειάζονται, όπως η πόρτα 110 της υπηρεσίας SMTP στην περίπτωση που ο εξυπηρετητής αναλαμβάνει να κοινοποιήσει κάποια ειδοποίηση στις ηλεκτρονικές διευθύνσεις

των χρηστών. Η εγκατάσταση του κεντρικού Η/Υ σε συστήματα κτιριακού αυτοματισμού ακολουθεί την λογική του σχήματος 3-4.



Σχήμα 3-4 Αρχιτεκτονική υλικού

Με τις κατάλληλες ρυθμίσεις στο router και στο firewall μπορούμε να αποκτήσουμε πρόσβαση στην διαδικτυακή εφαρμογή από όλους τους Η/Υ του δικτύου LAN ή και του Internet. Στην περίπτωση που επιθυμούμε απομακρυσμένη σύνδεση από το διαδίκτυο θα χρειαστεί τουλάχιστον μία στατική διεύθυνση IP έτσι ώστε να είναι εφικτό να συνδεθούμε με την εφαρμογή, ρυθμίζοντας το router να προωθεί την πόρτα 80 στην εσωτερική διεύθυνση IP του κεντρικού εξυπηρετητή.

3.4 Αρχιτεκτονική λογισμικού

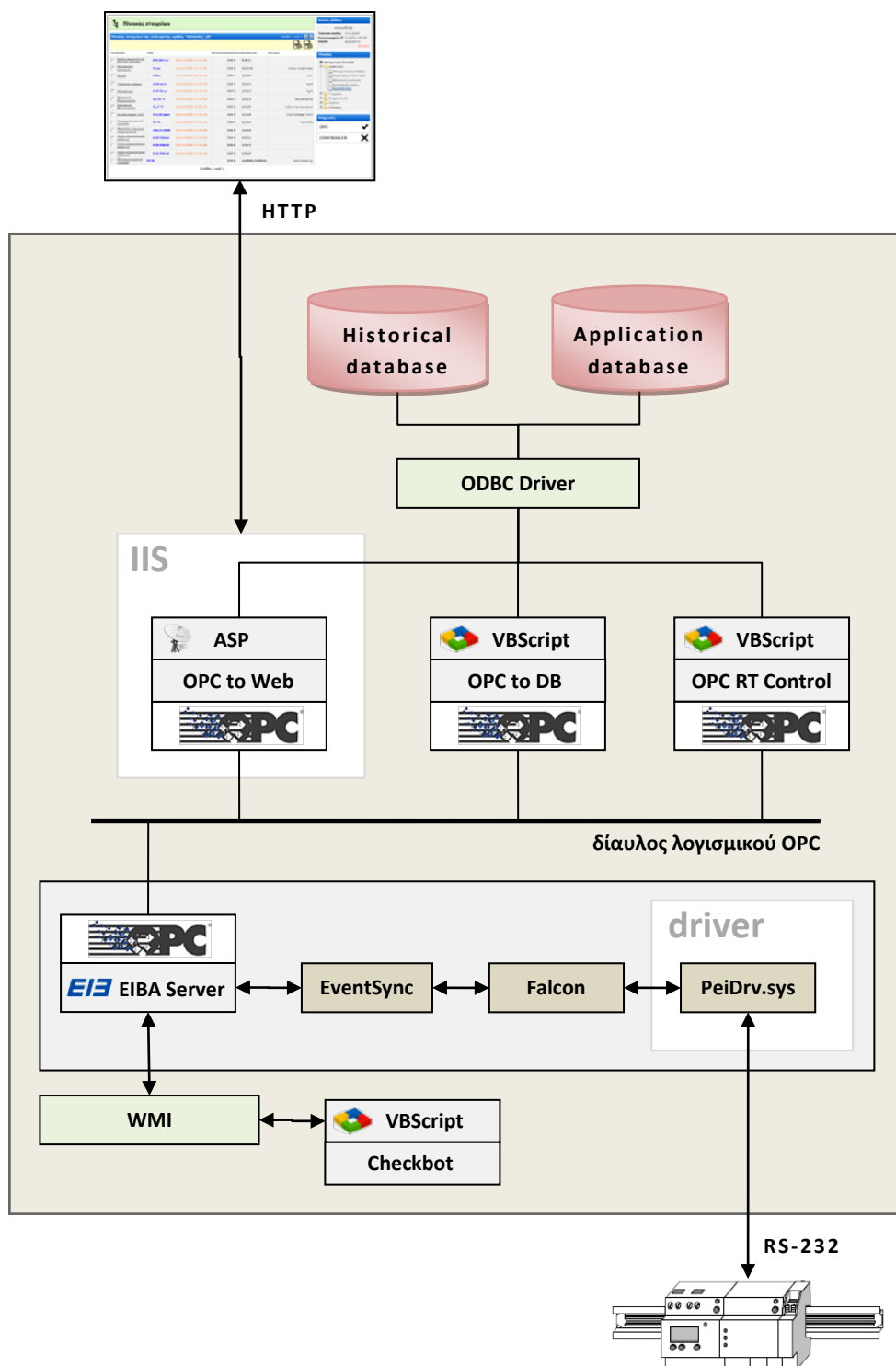
Όλα τα υποπρογράμματα που αναπτύχθηκαν στα πλαίσια αυτής της εργασίας καθώς και οι υπηρεσίες που τα διαχειρίζονται εγκαταστάθηκαν πάνω στην πλατφόρμα Microsoft Windows Server 2003. Η συγκεκριμένη έκδοση των Windows περιέχει ενσωματωμένο τον εξυπηρετητή ιστού IIS και δεν θέτει κανένα περιορισμό στον αριθμό των ενεργών συνδέσεων (sessions) σε αντίθεση με τα Microsoft Windows XP Professional. Με την χρήση του περιβλήματος automation (automation wrapper) του OPC που παρουσιάστηκε στο δεύτερο κεφάλαιο υλοποιήθηκαν οι παρακάτω πελάτες OPC:

- *OPC to Web*. Πρόκειται για την διαδικτυακή εφαρμογή του συστήματος η οποία συνδέεται με εξυπηρετητές τύπου data access. Η εφαρμογή αυτή διαβάζει και γράφει τα σημεία ελέγχου, κατασκευάζει ιστορικά διαγράμματα, διαχειρίζεται τους λογαριασμούς των χρηστών, την πολιτική ασφαλείας, εφαρμογές τρίτων, εντολές ελέγχου, ελεγκτές, ειδοποιήσεις κ.α.
- *OPC to DB*. Είναι μία αυτόνομη εφαρμογή γραμμένη σε VBScript η οποία διαβάζει και αποθηκεύει τις τιμές των σημείων της διαδικασίας ελέγχου σε μια βάση δεδομένων τύπου Microsoft Access 2003.
- *OPC Real Time Control*. Πρόκειται για έναν μηχανισμό πραγματικού χρόνου ο οποίος μέσω της βασικής διασύνδεσης χρήση της εφαρμογής OPC to Web, επιτρέπει στους χρήστες την ανάπτυξη και την εκτέλεση νέων αλγορίθμων και σεναρίων ελέγχου, ειδοποιήσεων σε κινητά τηλέφωνα, ηλεκτρονικές διευθύνσεις κ.τ.λ. στην γλώσσα σεναρίων VBScript. Επίσης διαχειρίζεται ένα κρυφό σενάριο (βλ. Παράρτημα A.11) που ενημερώνει την βάση δεδομένων της εφαρμογής OPC to Web με τις τρέχουσες τιμές των σημείων ελέγχου.

Με τον ODBC είναι συνδεδεμένες δύο βάσεις δεδομένων στις οποίες έχουν πρόσβαση όλα τα υποπρογράμματα που αναπτύχθηκαν. Η πρώτη είναι τεχνολογίας Microsoft SQL Server 2000 και αποθηκεύει τα δεδομένα που διαχειρίζεται η εφαρμογή OPC to Web (όπως πληροφορίες εκκίνησης, λογαριασμούς χρηστών κ.τ.λ.) καθώς και τις τρέχουσες τιμές των σημείων ελέγχου, ενώ η δεύτερη που είναι μία Microsoft Access 2003 περιέχει πίνακες με τα ιστορικά δεδομένα των σημείων ελέγχου.

Με χρήση της τεχνολογίας WMI και της γλώσσας σεναρίων VBScript υλοποιήθηκε επίσης το Checkbot (βλ. Παράρτημα A.10) το οποίο είναι μια αυτόνομη εφαρμογή που ελέγχει αν ο εγκατεστημένος OPC-Server είναι σε λειτουργία. Σε περίπτωση που δεν είναι, τον εκκινεί αυτόματα. Το πρόγραμμα αυτό χρησιμοποιεί υπηρεσίες και εργαλεία λογισμικού των Microsoft Windows για την εκτέλεση συγκεκριμένων ενεργειών.

Στο σχήμα 3-5 παρουσιάζεται η γενική δομή του λογισμικού, όπου διακρίνονται τα υποπρογράμματα, οι υπηρεσίες και οι μεταξύ τους συνδέσεις. Κάθε πελάτης OPC για να αποκτήσει πρόσβαση στα σήματα των συσκευών πεδίου συνδέεται με τον εξυπηρετητή OPC τύπου data access. Στην συνέχεια ο εξυπηρετητής μέσω κατάλληλου λογισμικού συγχρονισμού και οδηγών συνδέεται με την συσκευή ή την κάρτα διασύνδεσης.



Σχήμα 3-5 Αρχιτεκτονική λογισμικού του συστήματος

Το πλαίσιο που περιβάλλει τα στοιχεία EIB Server, EventSync, Falcon και τον οδηγό PeiDrv.sys αποτελεί το ολοκληρωμένο παράδειγμα χρήσης του συστήματος σε εγκαταστάσεις κτιριακού αυτοματισμού KNX/EIB με χρήση του EIBA OPC-Server. Με την αντικατάσταση αυτού και των αντίστοιχων drivers, το σύστημα μπορεί να εγκατασταθεί σε οποιοδήποτε βιομηχανικό δίκτυο που υποστηρίζει το OPC.

3.4.1 OPC to Web

3.4.1.1 Γενικά

Για την ανάπτυξη της διαδικτυακής εφαρμογής OPC to Web χρησιμοποιήθηκαν σε συνδυασμό όλες τεχνολογίες λογισμικού και οι τεχνικές που παρουσιάστηκαν στις προηγούμενες ενότητες της εργασίας. Τα αρχεία κώδικα πηγής της εφαρμογής ανάλογα με την γλώσσα σεναρίων στην οποία αναπτύχθηκαν και το περιβάλλον στο οποίο εκτελούνται διαχωρίζονται σε δύο βασικές κατηγορίες:

- *Server Side Scripting.* Σε αυτή την κατηγορία ανήκουν οι τεχνολογίες και οι γλώσσες σεναρίων που εκτελούν τις διεργασίες στον εξυπηρετητή κατόπιν αίτησης του πελάτη, στον οποίο και επιστρέφονται σε εύλογο χρονικό διάστημα τα αποτελέσματα, χωρίς να είναι σε θέση ο πελάτης να γνωρίζει τον εσωτερικό μηχανισμό της διεργασίας. Τεχνολογίες όπως η ASP με VBScript ή JScript, το CGI κ.τ.λ. αποτελούν παραδείγματα τεχνολογιών αυτής της κατηγορίας.
- *Client Side Scripting.* Μέσω του προγράμματος περιήγησης ιστού η διαδικτυακή εφαρμογή αποτελεί μια ολοκληρωμένη εφαρμογή διασύνδεσης για τον εποπτικό έλεγχο, την διαχείριση συμβάντων και την άντληση ιστορικών δεδομένων. Χρησιμοποιεί την γλώσσα Javascript για την δημιουργία των χειριστηρίων ελέγχου, την διαχείριση συμβάντων κατά την αλληλεπίδραση του interface με τον χρήστη αλλά και για την αποστολή ασύγχρονων αιτημάτων προς τον web server. Τα αρχεία κώδικα Javascript που αναπτύχθηκαν αρχικά μεταφέρονται στο πρόγραμμα περιήγησης ιστού και κατόπιν αίτησης εκτελούνται τοπικά στον υπολογιστή του κάθε χρήστη. Επίσης η εφαρμογή ενσωματώνει διάφορα στοιχεία ActiveX όπως για παράδειγμα ένα στοιχείο SWF για την δημιουργία δυναμικών διαγραμμάτων από ιστορικά δεδομένα που αντλεί από την αντίστοιχη βάση ιστορικών δεδομένων. Η μεταφορά των δεδομένων από τον web server στο στοιχείο αυτό γίνεται με την γλώσσα επισήμανσης XML.

Σε κάθε νέα αίτηση η διαδικτυακή εφαρμογή εκτελεί την δυναμική σελίδα default.asp η οποία με την σειρά της εκτελεί την αντίστοιχη υποσελίδα ASP που σχετίζεται με την αίτηση μέσω ενός μηχανισμού ελέγχου και της μεθόδου Server.Execute. Με αυτό τον τρόπο η σελίδα default.asp καλεί ανάλογα με το

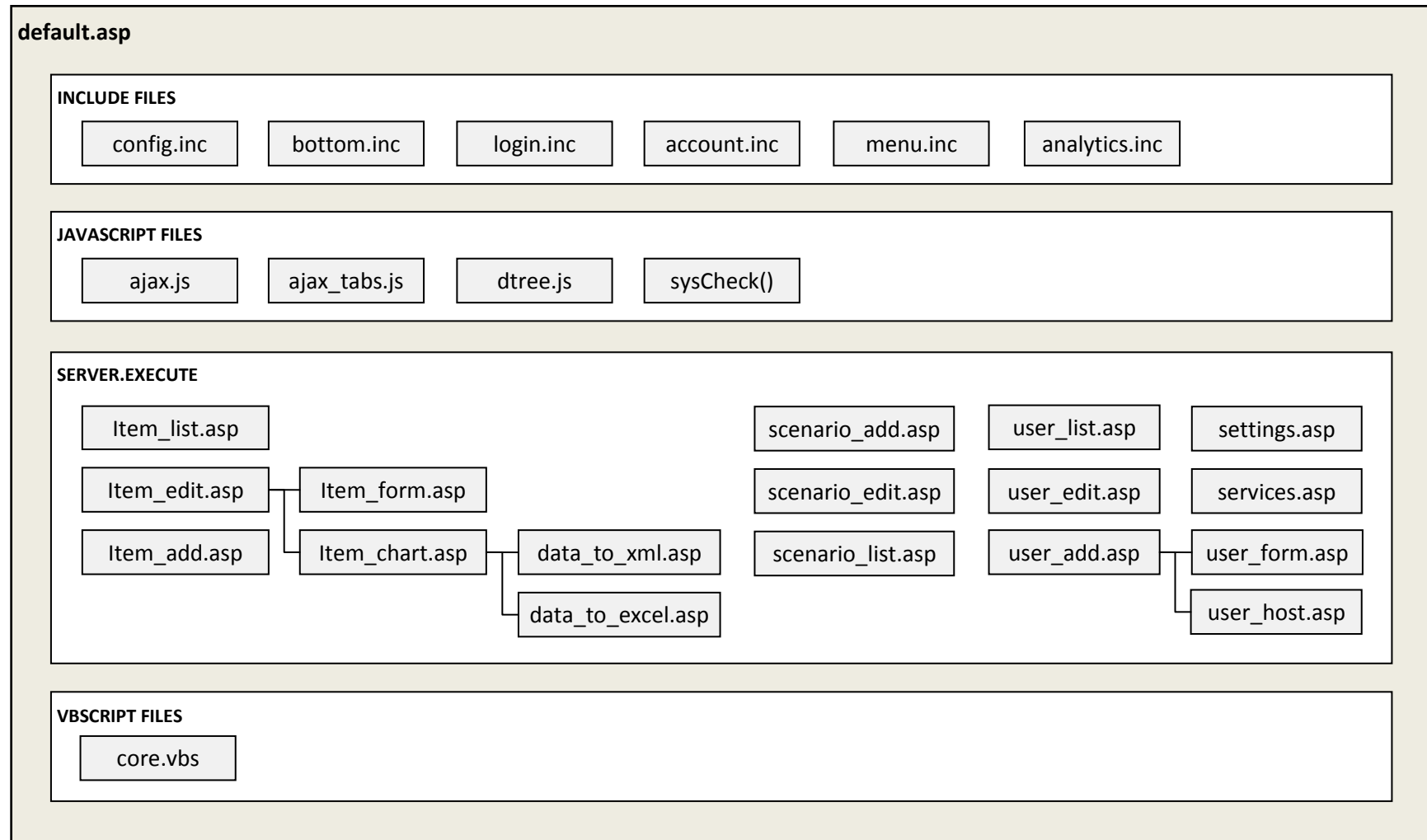
αίτημα την αντίστοιχη υποσελίδα η οποία περιέχει τον μηχανισμό και τον αντίστοιχο κώδικα για την εκτέλεση του αιτήματος. Τα πλεονεκτήματα αυτής της αρχιτεκτονικής εκτός της οργανωμένης της δομής του κώδικα πηγής είναι τα εξής:

- Λόγω του ότι η διαδικτυακή εφαρμογή διαχειρίζεται λογαριασμούς χρηστών με διαφορετικά επίπεδα πρόσβασης είναι εξαιρετικά εύκολο και γρήγορο να οριστούν ή να τροποποιηθούν μέσω του κώδικα τα δικαιώματα για κάθε υποσελίδα.
- Ο έλεγχος των session μεταβλητών που επαληθεύουν την κατάσταση εισόδου του χρήστη και το επίπεδο πρόσβασης του λογαριασμού του γίνεται σε κάθε νέα αίτηση μέσω της σελίδας default.asp και μόνο σ' αυτήν.
- Τα αρχεία διαμόρφωσης των σελίδων, όπως για παράδειγμα τα CSS καθώς και οι συναρτήσεις Javascript που μεταφέρονται στο πρόγραμμα περιήγησης ιστού του χρήστη εμπεριέχονται στην σελίδα default.asp και ισχύουν σε όλες της υποσελίδες της εφαρμογής δηλαδή, κληρονομούν τις ιδιότητες της.

Το κύριο μειονέκτημα αυτής της αρχιτεκτονικής είναι το ότι σε πολλές περιπτώσεις οι αιτήσεις για να εκτελεστούν χρειάζονται ένα μέρος μόνο του client side κώδικα πηγής που μεταφέρεται από τον εξυπηρετητή στο πρόγραμμα περιήγησης του χρήστη. Έτσι στην περίπτωση που πολλοί χρήστες συνδέονται ταυτόχρονα στην διαδικτυακή εφαρμογή η ταχύτητα μεταφοράς των δεδομένων δεν είναι η βέλτιστη, αφού μεταφέρονται τμήματα κώδικα και αρχεία μορφοποίησης γραφικών που πιθανώς να μην χρειάζονται για την εκτέλεση των αιτημάτων.

Όμως τα περισσότερα προγράμματα περιήγησης ιστού αποθηκεύουν τα αρχεία αυτά σε ειδικό φάκελο σαν προσωρινά αρχεία. Δηλαδή σε κάθε νέα αίτηση το πρόγραμμα περιήγησης χρησιμοποιεί τα προσωρινά αρχεία για να ελαχιστοποιήσει το χρόνο απόκρισης της διαδικτυακής εφαρμογής. Επίσης λόγω των ειδικών λειτουργικών χαρακτηριστικών της διαδικτυακής εφαρμογής δεν ενδείκνυται η ταυτόχρονη χρήση από μεγάλο αριθμό χρηστών.

Στο σχήμα 3-6 παρουσιάζεται η εσωτερική δομή της σελίδας default.asp όπου διακρίνονται οι τέσσερις βασικές ομάδες περιλαμβανόμενων αρχείων. Μόνο τα αρχεία και οι συναρτήσεις της Javascript εκτελούνται τοπικά στο πρόγραμμα περιήγησης ιστού του χρήστη ενώ τα αρχεία κώδικα πηγής των υπολοίπων ομάδων δηλαδή τα αρχεία με τις καταλήξεις .inc, .asp και .vbs αναγνωρίζονται από τον web server σαν περιλαμβανόμενα αρχεία της default.asp και εκτελούνται από αυτόν κατά την διαδικασία σάρωσης και εκτέλεσης του κώδικα της σελίδας. Μέσω του μηχανισμού ελέγχου Select...Case που υποστηρίζεται από την ASP η μέθοδος server.execute εκτελεί την υποσελίδα ASP που θα της υποδείξει το query string του αντίστοιχου URL (Uniform Resource Link).



Σχήμα 3-6 Τα περιλαμβανόμενα αρχεία της σελίδας default.asp

3.4.1.2 Δομή συνδέσμων

Κάθε σύνδεσμος της διαδικτυακής εφαρμογής είτε αυτός έχει δημιουργηθεί δυναμικά από την ίδια την εφαρμογή είτε στατικά από τον προγραμματιστή έχει υποχρεωτικά δύο query strings με τις ονομασίες node και sub οι τιμές των οποίων αποστέλλονται από το πρόγραμμα περιήγησης ιστού στον IIS μέσω της δυναμικής σελίδας default.asp για δρομολόγηση και εκτέλεση του αντίστοιχου κώδικα. Στην περίπτωση που τα node και sub δεν έχουν πάρει έγκυρες τιμές το σύστημα επιστρέφει την αρχική σελίδα της εφαρμογής. Η δομή του ενεργού συνδέσμου φαίνεται παρακάτω όπου στην συγκεκριμένη διαδικτυακή εφαρμογή το node έχει πάντα την τιμή app από την λέξη application (δίνει περιθώρια για μελλοντικές επεκτάσεις του συστήματος) ενώ το sub μπορεί να πάρει μία από τις έγκυρες τιμές του server.execute που δόθηκαν στο αντίστοιχο πλαίσιο του σχήματος 3-6 όπως για παράδειγμα item_list, item_edit, item_add, user_list, user_edit κτλ.

`http://<domain>/default.asp?node=app&sub=<υποσελίδα>`

Ανάλογα με τα λειτουργικά χαρακτηριστικά και τις απαιτήσεις της κάθε υποσελίδας υπάρχει το ενδεχόμενο να χρησιμοποιούνται περισσότερα από τα δύο βασικά query strings. Αυτό για παράδειγμα συμβαίνει στις σελίδες προβολής και επεξεργασίας των δεδομένων των στοιχείων της διαδικασίας ελέγχου όπου εκεί ο ενεργός σύνδεσμος εμπεριέχει το αντίστοιχο μοναδικό αναγνωριστικό id του στοιχείου. Στην συγκεκριμένη διαδικτυακή εφαρμογή αυτό είναι πάντα ένας ακέραιος αριθμός.

`http://<domain>/default.asp?node=app&sub=<υποσελίδα>&id=<αριθμός>`

Δύο πραγματικά παραδείγματα URL της διαδικτυακής εφαρμογής με τα αντίστοιχα query strings που χρησιμοποιούν είναι τα εξής:

- Προβολή και επεξεργασία των στοιχείων του χρήστη, με αναγνωριστικό id τον ακέραιο αριθμό 253:
`http://ermis.smartlab.tuc.gr/default.asp?node=app&sub=user_edit&id=253`
- Προβολή και επεξεργασία των δεδομένων του αισθητήρα εσωτερικής θερμοκρασίας, με αναγνωριστικό id τον ακέραιο αριθμό 22:
`http://ermis.smartlab.tuc.gr/default.asp?node=app&sub=item_edit&id=22`

3.4.1.3 Έλεγχος υπηρεσιών

Η διαδικτυακή εφαρμογή OPC to Web έχει την δυνατότητα να ελέγχει τις καταστάσεις λειτουργίας, να εκκινεί ή και να τερματίζει εφαρμογές που σχετίζονται με τις υπηρεσίες που εποπτεύει. Αυτό το επιτυγχάνει μέσω των τεχνολογιών αυτοματισμού WMI και WSH που περιλαμβάνει η Microsoft στα Windows, σε συνδυασμό με την χρήση αρχείων Batch και του εργαλείου PsExec της Sysinternals.

Ο εξυπηρετητής OPC είναι μία υπηρεσία τρίτου κατασκευαστή που είναι απαραίτητη για την λειτουργία του συστήματος δεδομένου ότι σε αυτήν συνδέονται μέσω της τεχνολογίας automation όλοι οι πελάτες OPC που αναπτύχθηκαν στα πλαίσια αυτής της εργασίας. Στην ενότητα 3.2.5 διατυπώθηκε η ιδέα της εκτέλεσης κώδικα VBScript που περιλαμβάνει αντικείμενα και μεθόδους της τεχνολογίας WMI από τον IIS μέσω μίας δυναμικής σελίδας ASP και επίσης δόθηκαν όλες οι απαραίτητες ρυθμίσεις ασφαλείας που πρέπει να γίνουν στον IIS. Με την χρήση της τεχνολογίας AJAX είναι δυνατό η εφαρμογή OPC to Web να ενημερώνει τους συνδεδεμένους χρήστες μέσω του προγράμματος περιήγησης ιστού για την κατάσταση λειτουργίας της υπηρεσίας χωρίς την υποχρεωτική ανανέωση (Refresh) των σελίδων. Αυτό το επιτυγχάνει μέσω μίας συνάρτησης Javascript με την ονομασία sysCheck() η οποία εμπεριέχεται στον client side κώδικα της σελίδας default.asp και ανά τακτά χρονικά διαστήματα «πυροδοτεί» την αποστολή ασύγχρονου αιτήματος προς τον IIS και συγκεκριμένα προς την δυναμική σελίδα ASP που ενσωματώνει την τεχνολογία WMI. Τα αποτελέσματα των αιτημάτων που επιστρέφει η δυναμική σελίδα ASP είναι οι τιμές true ή false ανάλογα με την κατάσταση λειτουργίας. Με αυτό τον μηχανισμό η εφαρμογή OPC to Web εποπτεύει κάθε χρονική στιγμή την κατάσταση λειτουργίας του εξυπηρετητή OPC. Εκτός όμως του εξυπηρετητή OPC η εφαρμογή OPC to Web σχεδιάστηκε έτσι ώστε να εποπτεύει τις καταστάσεις λειτουργίας για δύο ακόμα υπηρεσίες.

Η πρώτη από αυτές είναι το πρόγραμμα Matlab και συγκεκριμένα η έκδοση R2006b η οποία περιλαμβάνει την εργαλειοθήκη OPC Toolbox. Το πακέτο αυτό παρέχει την δυνατότητα στους χρήστες που έχουν βασικές γνώσεις προγραμματισμού να προχωρήσουν στην δημιουργία ελεγκτών - πελατών OPC όπως για παράδειγμα αλγορίθμων ελέγχου καλύτερης διαχείρισης της ενέργειας στα κτίρια και να τους συνδέσουν με ένα εξυπηρετητή OPC. Με αυτό τον τρόπο η εφαρμογή OPC to Web μπορεί να εποπτεύει ελεγκτές οι οποίοι έχουν υλοποιηθεί στο Matlab (βλ. Παράρτημα Α.1, Εικόνα Α.1-2). Επίσης μέσω ειδικών συναρτήσεων η εφαρμογή μπορεί να διαβάζει κοινόχρηστες μεταβλητές που είναι απαραίτητες για την λειτουργία των ελεγκτών, όπως για παράδειγμα τα επιθυμητά set point και εντολές τμηματικού ελέγχου (βλ. Ενότητα 3.4.5 και Παράρτημα Α.5).

Η δεύτερη υπηρεσία είναι ο μηχανισμός OPC Real Time Control που υλοποιήθηκε στα πλαίσια αυτής της εργασίας και αποτελεί ένα αυτόνομο υποπρόγραμμα το οποίο εποπτεύει η εφαρμογή OPC to Web. Πρόκειται για ένα αρχείο VBScript με την ονομασία scenario_exe.vbs το οποίο περιέχει ένα βρόχο While(1)...Wend ο οποίος κρατάει την υπηρεσία σε συνεχή λειτουργία. Εσωτερικά του βρόχου εκτελείται το αρχείο scenario_list.vbs που το ενημερώνει η εφαρμογή OPC to Web με τους ελεγκτές και τα σενάρια που αναπτύσσουν και διαχειρίζονται οι χρήστες της εφαρμογής.

3.4.1.4 Σήματα συσκευών πεδίου

Τα σήματα των συσκευών πεδίου κατηγοριοποιούνται από την εφαρμογή OPC to Web σε δύο βασικές κατηγορίες. Η πρώτη κατηγορία περιλαμβάνει τα σήματα που προέρχονται από τους αισθητήρες της διαδικασίας ελέγχου (βλ. Παράρτημα Α.1, Εικόνα Α.1-3). Τα σήματα αυτά είναι μόνο για ανάγνωση και έτσι η εφαρμογή απλά εκτυπώνει τις τιμές που αποστέλλει ο εξυπηρετητής OPC του συστήματος. Η δεύτερη κατηγορία περιλαμβάνει τα σήματα που προέρχονται από τους ενεργοποιητές της διαδικασίας ελέγχου (βλ. Παράρτημα Α.1, Εικόνα Α.1-8) στα οποία η εφαρμογή έχει δικαιώματα ανάγνωσης αλλά και εγγραφής τιμών.

Οι υποσελίδες που εκτυπώνουν τις τρέχουσες τιμές των σημάτων είναι δύο και είναι οι ίδιες είτε πρόκειται για αισθητήρες είτε για ενεργοποιητές. Όλα τα σήματα των συσκευών πεδίου που «βλέπει» η εφαρμογή OPC to Web εμφανίζονται σαν στοιχεία μίας λίστας στην οποία περιλαμβάνονται διάφορα δεδομένα τα βασικότερα των οποίων είναι η ονομασία και η τρέχουσα τιμή του σήματος με το αντίστοιχο timestamp. Την λίστα αυτή την δημιουργεί η υποσελίδα `item_list.asp` η οποία κατηγοριοποιεί τα σήματα στις αντίστοιχες κατηγορίες και υποκατηγορίες ανάλογα με την τιμή ενός query string που ονομάζεται `flag`. Το συγκεκριμένο query string χρησιμοποιείται από τους στατικούς συνδέσμους του κεντρικού μενού της εφαρμογής. Στην περίπτωση που στην λίστα βρεθεί το σήμα ενός ενεργοποιητή, ένας εσωτερικός μηχανισμός δημιουργεί δυναμικά το χειριστήριο ελέγχου για τον έλεγχο του συγκεκριμένου σήματος. Ο μηχανισμός αυτός αντλεί από την βάση δεδομένων της εφαρμογής το εύρος των επιτρεπτών τιμών, τις ετικέτες του χειριστηρίου, το βήμα και όλα τα δεδομένα που χρειάζεται ώστε να δημιουργήσει με ακρίβεια το χειριστήριο ελέγχου που θα ανταποκρίνεται ακριβώς στις ρυθμίσεις της αντίστοιχης διεύθυνσης του εξυπηρετητή OPC.

Η εφαρμογή OPC to Web παρέχει επίσης την δυνατότητα επεξεργασίας όλων εκείνων των δεδομένων που είναι απαραίτητα για κάθε στοιχείο (διεύθυνση, ομάδα και υποομάδα) ώστε να μπορεί να συνδεθεί επιτυχώς με τον εξυπηρετητή OPC και να αποκτήσει πρόσβαση στο αντίστοιχο σήμα του (βλ. Παράρτημα Α.1, Εικόνα Α.1-4). Αυτό γίνεται μέσω της υποσελίδας `item_edit.asp` στην οποία υπάρχουν δύο καρτέλες. Η πρώτη ενσωματώνει την υποσελίδα `item_form.asp` που περιέχει την φόρμα επεξεργασίας των δεδομένων ενώ η δεύτερη ενσωματώνει την υποσελίδα `item_chart.asp` η οποία περιέχει ένα στοιχείο ActiveX SWF για την δημιουργία δυναμικών διαγραμμάτων από ιστορικά δεδομένα (βλ. Παράρτημα Α.1, Εικόνα Α.1-5). Πανομοιότυπη με την υποσελίδα `item_form.asp` είναι η υποσελίδα `item_add.asp` η οποία παρέχει την δυνατότητα προσθήκης (αντί για επεξεργασία) νέων στοιχείων στο σύστημα είτε πρόκειται για αισθητήρες είτε για ενεργοποιητές. Η υποσελίδα αυτή δημιουργεί το στοιχείο και το αποθηκεύει στην βάση δεδομένων της εφαρμογής. Αν τα απαραίτητα δεδομένα (διεύθυνση, ομάδα και υποομάδα)

καταχωρηθούν σωστά τότε η εφαρμογή θα επιστρέψει απευθείας την τρέχουσα τιμή του νέου σήματος.

Οι υποσελίδες `item_list.asp` και `item_edit.asp` με την χρήση της τεχνολογίας AJAX ανανεώνουν αυτόματα ανά τακτά χρονικά διαστήματα τις τρέχουσες τιμές των συσκευών πεδίου και τις νέες θέσεις των χειριστηρίων ελέγχου με ένα μηχανισμό που ακολουθεί την λογική που αναπτύχθηκε στην ενότητα 3.4.1.3. Δύο συναρτήσεις Javascript με τις ονομασίες `Refresh()` και `RefreshControls()` οι οποίες εμπεριέχονται στον client side κώδικα των υποσελίδων αυτών ανά τακτά χρονικά διαστήματα «πυροδοτούν» την αποστολή ασύγχρονων αιτημάτων προς τις δυναμικές σελίδες που διαβάζουν (βλ. Παράρτημα Α.11) και γράφουν τιμές στα σήματα των σημείων ελέγχου. Τα αποτελέσματα των σελίδων αυτών είναι οι τρέχουσες τιμές των σημάτων που είτε εκτυπώνονται σε προκαθορισμένες θέσεις μέσα στην εφαρμογή είτε ενημερώνουν τα χειριστήρια ελέγχου των ενεργοποιητών στις νέες τους θέσεις.

3.4.1.5 Λογαριασμοί χρηστών

Για την είσοδο στην εφαρμογή OPC to Web το σύστημα εφαρμόζει μια πολιτική ασφαλείας με λογαριασμούς εισόδου έτσι ώστε να προστατεύει τις σελίδες ελέγχου του συστήματος. Η εφαρμογή διαθέτει ένα μηχανισμό που αναγνωρίζει τον τερματικό Η/Υ του χρήστη ελέγχοντας την διεύθυνση IP που του έχει δοθεί από το δίκτυο. Έτσι σε περίπτωση που ο εποπτικός έλεγχος εφαρμόζεται σε συγκεκριμένο χώρο όπως για παράδειγμα σε ένα εργοστάσιο ή σε ένα κτίριο, μερικές βασικές λειτουργίες χρήσης ή λειτουργίες έκτακτης ανάγκης μπορούν να είναι ελεύθερης πρόσβασης χωρίς να είναι υποχρεωτική η αναγνώριση των χρηστών από σύστημα. Διαφορετικά οι χρήστες της εφαρμογής θα πρέπει να δώσουν τα προσωπικά στοιχεία εισόδου (βλ. Παράρτημα Α.1, Εικόνα Α.1-1).

Υπάρχουν δύο επίπεδα λογαριασμών (βλ. Παράρτημα Α.1, Εικόνα Α.1-10). Τα δικαιώματα των επιπέδων αυτών είναι προκαθορισμένα από την φάση σχεδιασμού και υλοποίησης της εφαρμογής και είναι τα παρακάτω:

- *Επισκέπτης*: Τα δικαιώματα επιπέδου «Επισκέπτη» επιτρέπουν στο χρήστη να εισέρχεται στην εφαρμογή να διαβάζει ή και να γράφει τα σήματα της διαδικασίας ελέγχου, να δημιουργεί διαδικασίες ελέγχου, ειδοποιήσεις κ.τ.λ. και να διαχειρίζεται τις υπηρεσίες που εποπτεύει η εφαρμογή.
- *Διαχειριστής*: Τα δικαιώματα επιπέδου «Διαχειριστή» επιτρέπουν στο χρήστη να πραγματοποιεί όλες τις ενέργειες του προηγούμενου επιπέδου και ακόμα να διαγράφει, να επεξεργάζεται ή και να προσθέτει νέες συσκευές πεδίου, να διαχειρίζεται τους λογαριασμούς των χρηστών και τις γενικές ρυθμίσεις της εφαρμογής.

3.4.1.6 Βάση δεδομένων εφαρμογής

Τα δεδομένα των στοιχείων καθώς και όλα τα υπόλοιπα δεδομένα που χρειάζονται για την λειτουργία του συστήματος αποθηκεύονται σε μία βάση δεδομένων Microsoft SQL 2000 που συνδέεται μέσω του ODBC με system DSN το OPCWEB. Αυτή η βάση δεδομένων περιέχει τους εξής πίνακες:

tbl_items id (auto number) address (text) description (text) format (number) priority (number) groupName (text) subGroupName (text) name (text) flag (text) min_value (text) min_label (text) max_value (text) max_label (text) step (text) gain (text) width (text) relative (text) level (number) unit (text) writeProtection (text) valueRead(text) timestampR(text)	tbl_format id (auto number) name (text) description (text) chartFlash (text) chartPage (text) chartHeight (text)	tbl_users id (auto number) username (text) password (text) lastname (text) firstname (text) level (number) email (text) block (number) phone (text) remote_host (text) last_login (date)
tbl_sSms id (auto number) time (date) subject (text)	tbl_priority id (auto number) name (text)	
tbl_sEmail id (auto number) time (date) subject (text)	tbl_scenarios id (auto number) title (text) body (memo) active (text) start (date) stop (date) duration (text) mon (text) tue (text) wed (text) thu (text) fri (text) sat (text) sun (text) time1 (text) time2 (text)	tbl_settings id (auto number) name (text) value (text) description (text)
	tbl_hosts id (auto number) userID (number) host (text)	
	tbl_levels id (auto number) title (text) level (number)	
	tbl_groups groupID (text) title (text)	

Σχήμα 3-7 Δομή της βάσης δεδομένων OPCWEB

Στον πίνακα tbl_items αποθηκεύονται τα δεδομένα των στοιχείων όπου οι τιμές των πεδίων address, format, priority, groupName, subGroupName είναι απαραίτητες για

την επικοινωνία με τον εξυπηρετητή OPC ενώ οι τιμές των πεδίων `min_value`, `min_label`, `max_value`, `max_label`, `step`, `gain`, `width` κ.τ.λ. χρησιμοποιούνται μόνο από τα στοιχεία τύπου ενεργοποιητών για την δημιουργία των αντίστοιχων χειριστηρίων ελέγχου. Οι τιμές του πεδίου `flag` χρησιμοποιούνται για την κατηγοριοποίηση των στοιχείων στις κατηγορίες και υποκατηγορίες του κεντρικού μενού της εφαρμογής. Κάθε εγγραφή του πίνακα `tbl_items` δηλαδή κάθε σημείο ελέγχου συσχετίζεται με μία εγγραφή του πίνακα `tbl_format` που περιέχει τους τύπος δεδομένων που υποστηρίζονται από την εφαρμογή. Αυτοί είναι οι εξής:

- EIS 1 (Switching)
- EIS 2 (Dimming control)
- EIS 5 (Value)
- EIS 6 (Scaling - percent)
- EIS 9 (Float data)

Υπάρχει δυνατότητα αναβάθμισης του συστήματος ώστε να υποστηρίζει και τους υπόλοιπους τύπους δεδομένων αν αυτό χρειαστεί.

Στον πίνακα `tbl_scenarios` αποθηκεύονται οι αλγόριθμοι, οι ειδοποιήσεις και τα σενάρια λειτουργίας που υλοποιούν οι χρήστες της εφαρμογής. Υπάρχουν πεδία όπως τα `start`, `stop`, `duration`, `mon`, `tue`, `wed` κτλ. που αποθηκεύουν ρυθμίσεις σχετικές με το ποιες μέρες θα εκτελούνται, ποιες ώρες της ημέρας κτλ.

Στον πίνακα `tbl_users` αποθηκεύονται τα προσωπικά δεδομένα των χρηστών καθώς και πληροφορίες της τελευταίας χρήσης του λογαριασμού, όπως το τελευταίο IP και ημερομηνία. Οι υπόλοιποι πίνακες είναι βοηθητικοί και σε αυτούς αποθηκεύονται τα δεδομένα ρυθμίσεων του συστήματος.

3.4.1.7 Κώδικας τρίτου κατασκευαστή

Η εφαρμογή OPC to Web ενσωματώνει κώδικα τρίτου κατασκευαστή σε τρία σημεία. Ο κώδικας αυτός διατίθεται δωρεάν και η άδεια χρήσης *GNU General Public License* επιτρέπει την ελεύθερη χρήση και τροποποίηση του.

- Στην σελίδα `default.asp` περιλαμβάνεται το αρχείο Javascript `dtree.js` του *Christopher Hwang* το οποίο δημιουργεί το στατικό κεντρικό μενού της εφαρμογής με μορφή δέντρου ανάπτυξης.
- Στην υποσελίδα `item_edit.asp` περιλαμβάνεται το *Ajax Tabs Content Script v2.2* της *DynamicDrive.com* ένας μηχανισμός που δημιουργεί καρτέλες από υπάρχουσες σελίδες με δυναμικό περιεχόμενο χωρίς την ανανέωση των σελίδων.
- Στην υποσελίδα `item_chart.asp` περιλαμβάνεται το αντικείμενο Flash *FusionCharts v3* της *InfoSoft Global* που εκτυπώνει δυναμικά διαγράμματα από τα δεδομένα που στέλνει η εφαρμογή μέσω ενός αρχείου XML.

3.4.1.8 Παράμετροι εφαρμογής

Στην εφαρμογή OPC to Web αναπτύχθηκε ένας μηχανισμός για την επεξεργασία, όλων των παραμέτρων που χρειάζονται για την εγκατάσταση της σε διαφορετικούς κεντρικούς Η/Υ (βλ. Παράρτημα Α.2 και Παράρτημα Α.1, Εικόνα Α.1-12).

Δηλαδή η εφαρμογή δίνει την δυνατότητα επεξεργασίας όλων εκείνων των δεδομένων που της επιτρέπουν να επικοινωνεί με το περιβάλλον της. Συνοπτικά η εφαρμογή έχει τις εξής δυνατότητες παραμετροποίησης:

- Ρύθμιση λειτουργίας on ή off όλων των μηχανισμών τεχνολογίας AJAX μέσω της μεταβλητής AjaxState και των χρόνων αναμονής των συναρτήσεων sysCheck(), Refresh() και RefreshControls(). Αν έχουμε στην διάθεση μας έναν ταχύτερο κεντρικό Η/Υ μπορούμε να ελαττώσουμε αυτούς τους χρόνους. Το βέλτιστο σημείο λειτουργίας το βρίσκουμε εμπειρικά.
- Ρύθμιση των παραμέτρων σύνδεσης με τον εξυπηρετητή OPC, που είναι ο φάκελος εγκατάστασης του εξυπηρετητή, η ονομασία της διεργασίας στο task manager, η ονομασία του PROGID του αντίστοιχου αντικειμένου COM και ένα ψευδώνυμο αναγνώρισης.
- Ρύθμιση των παραμέτρων σύνδεσης του βασικού ελεγκτή (στην παρούσα εργασία αυτός είναι το Matlab) που είναι αντίστοιχες με τις ρυθμίσεις του εξυπηρετητή OPC αλλά διαφέρουν στο ότι δεν υπάρχει η ρύθμιση για COM.
- Ρύθμιση των παραμέτρων σύνδεσης του μηχανισμού OPC Real Time Control που είναι αντίστοιχες με τις ρυθμίσεις του βασικού ελεγκτή.
- Ρύθμιση της διεύθυνσης IP για το υποδίκτυο μέσα στο οποίο εφαρμόζεται πολιτική ασφαλείας ελεύθερης πρόσβασης σε βασικές λειτουργίες άμεσης πρόσβασης. Η συγκεκριμένη υπηρεσία κλειδώνει τις βασικές λειτουργίες άμεσης πρόσβασης όταν δοθεί η διεύθυνση IP του κεντρικού Η/Υ.
- Τέλος ρυθμίσεις που αφορούν την σύνδεση της εφαρμογής με τον SMTP Mail Server, το SMS Gateway κ.τ.λ.

Όλα τα δεδομένα των ρυθμίσεων της εφαρμογής αποθηκεύονται στην βάση δεδομένων OPCWEB και συγκεκριμένα στον πίνακα tbl_settings.

3.4.2 OPC to DB

Το OPC to DB είναι μία εφαρμογή VBScript η οποία αναλαμβάνει την αποθήκευση των σημάτων των συσκευών πεδίου σε μία βάση δεδομένων MS Access 2003 με την ονομασία OPCDATA στο system DSN του ODBC. Η εφαρμογή χρησιμοποιεί την τεχνολογία WMI για να επιβεβαιώσει ότι ο εξυπηρετητής OPC λειτουργεί κανονικά και στην συνέχεια με την χρήση συναρτήσεων της εφαρμογής OPC to Web αποκτά πρόσβαση στα σήματα, διαβάζει τις τιμές τους και τις αποθηκεύει στην βάση δεδομένων (βλ. Παράρτημα Α.9). Αυτή η διαδικασία γίνεται ανά πέντε λεπτά μέσω μίας εφαρμογής προγραμματισμού εργασιών του λειτουργικού συστήματος Windows Server 2003 που αναλαμβάνει την εκκίνηση της εφαρμογής OPC to DB αυτόματα κάθε φορά που θα περάσει το διάστημα παύσης των πέντε λεπτών.

Ανεξάρτητα από την κατάσταση λειτουργίας της εφαρμογής OPC to Web η αυτόνομη εφαρμογή OPC to DB ενημερώνει την βάση ιστορικών δεδομένων. Η συγκεκριμένη βάση αποτελείται από έναν πίνακα με την ονομασία tbl_data τα δεδομένα του οποίου είναι τα εξής:

tbl_data
id (auto number)
timestamp (date/time)
2_0_0 (text)
2_0_1 (text)
2_0_2 (text)
. (text)
. (text)
. (text)
set_tem (text)
set_bri (text)
set_co2 (text)
control_aircondition (text)
control_windows (text)
control_lights (text)

Σχήμα 3-8 Δομή της βάσης δεδομένων OPCDATA

Το πεδίο timestamp περιέχει την χρονική στιγμή την οποία έτρεξε η εφαρμογή, ενώ τα πεδία 2_0_0, 2_0_1 κ.τ.λ. περιέχουν τις τιμές των σημάτων της διαδικασίας ελέγχου. Τα υπόλοιπα πεδία περιέχουν τις καταστάσεις λειτουργίας ενός ελεγκτή προβλεπτικού ελέγχου που τρέχει στο Matlab και ο οποίος χρησιμοποιείται για την εξοικονόμηση ενέργειας σε εγκαταστάσεις κτιριακού αυτοματισμού KNX/EIB (17). Τις τιμές των συγκεκριμένων πεδίων η εφαρμογή τις αντλεί έμμεσα από το μητρώο του λειτουργικού συστήματος (βλ. Ενότητα 3.4.5 και Παράρτημα Α.5). Η λειτουργία του αλγόριθμου θα παρουσιαστεί στο επόμενο κεφάλαιο.

3.4.3 OPC Real Time Control

Το υποπρόγραμμα OPC Real Time Control έχει αναπτυχθεί σε γλώσσα VBScript και περιέχει έναν βρόχο While(1)...Wend που το κρατάει σε συνεχή λειτουργία. Η εκκίνηση και ο τερματισμός γίνεται από τον πίνακα ελέγχου υπηρεσιών της εφαρμογής OPC to Web με χρήση της τεχνολογίας WMI. Εκτός του βρόχου το υποπρόγραμμα περιέχει έτοιμες συναρτήσεις και υπορουτίνες με σκοπό να χρησιμοποιηθούν από τους χρήστες κατά την ανάπτυξη των δικών τους ελεγκτών, σεναρίων, προειδοποιήσεων κ.τ.λ.

Εσωτερικά ο βρόχος εκτελεί ένα αρχείο το οποίο το ενημερώνει η εφαρμογή OPC to Web με τα σενάρια που υπάρχουν αποθηκευμένα στον πίνακα tbl_scenarios. Σε αυτό το αρχείο περιέχονται όλα τα σενάρια που έχουν αναπτύξει οι χρήστες του συστήματος. Πραγματικά παραδείγματα τέτοιων σεναρίων θα παρουσιαστούν στο επόμενο κεφάλαιο (βλ. Ενότητα 4.4).

Οι συναρτήσεις που υποστηρίζονται και που αναπτύχθηκαν στα πλαίσια αυτής της διπλωματικής εργασίας και είναι οι εξής:

- sWrite(address, value, delay σε δευτερόλεπτα)
- sRead(address)
- sReadReg(key)
- sEmail(email, subject, body, delay σε λεπτά)
- sSms(phone, subject, body, delay σε λεπτά)

Όλα τα ορίσματα των παραπάνω συναρτήσεων εκτός του value της sWrite και του delay, είναι αλφαριθμητικά και για τον λόγο αυτό κατά την διαδικασία ανάπτυξης των σεναρίων θα πρέπει να δίδονται μέσα σε δίπλα λατινικά εισαγωγικά.

Η συνάρτηση sReadReg επιστρέφει την τιμή που διαβάζει από το αντίστοιχο κλειδί του μητρώου. Την χρησιμοποιούμε όταν θέλουμε να διαβάσουμε την τρέχουσα τιμή του αντίστοιχου set point που έχει εκείνη την στιγμή το χειριστήριο έλεγχου της εφαρμογής OPC to Web. Η συνάρτηση αυτή χρησιμοποιεί την τεχνολογία WSH που παρουσιάστηκε αναλυτικά στην 3.2.6.

Η συναρτήσεις sEmail και sSms χρησιμοποιούνται για την απευθείας αποστολή ειδοποιήσεων σε ηλεκτρονικές διευθύνσεις και κινητά τηλέφωνα. Χρησιμοποιούν το όρισμα delay σε λεπτά, για το μπλοκάρισμα αποστολής πιθανών επαναληπτικών ειδοποιήσεων πριν περάσει το χρονικό διάστημα που ορίζεται. Για την αποστολή των γραπτών μηνυμάτων το υποπρόγραμμα στέλνει HTTP μηνύματα σε έναν SMS Gateway ο οποίος στην συνέχεια προωθεί τα μηνύματα στις εταιρίες κινητής τηλεφωνίας ενώ για την αποστολή των ηλεκτρονικών μηνυμάτων το υποπρόγραμμα χρησιμοποιεί έναν mail server ο οποίος αναλαμβάνει την προώθηση παράδοση τους στους pop3 mail servers των λογαριασμών των παραληπτών.

3.4.4 Ρυθμίσεις αυτοματοποίησης

Δεδομένου ότι ο κεντρικός Η/Υ του συστήματος χρησιμοποιείται αποκλειστικά για τις υπηρεσίες και τις εφαρμογές της παρούσας εργασίας κρίνεται απαραίτητο να αναπτυχθεί βοηθητικό λογισμικό και να γίνουν στο λειτουργικό σύστημα μερικές ρυθμίσεις αυτοματοποίησης διαδικασιών.

Ο κεντρικός Η/Υ όπως αναφέρθηκε συνοπτικά και στην ενότητα 3.3 μετά από μία πιθανή διακοπή ρεύματος και αποφόρτισης των μπαταριών του UPS επανέρχεται αυτόματα σε κανονική κατάσταση λειτουργίας. Εκτός από τις ρυθμίσεις του BIOS στην μητρική κάρτα που της επιτρέπουν να αντιληφτεί την επαναφορά της τάσης και να εκκινήσει τον Η/Υ πρέπει μα γίνουν μία σειρά από περαιτέρω ενέργειες αυτοματοποίησης που έχουν ως εξής:

1. Αυτόματη είσοδος του λογαριασμού του χρήστη που χρησιμοποιούν οι εφαρμογές του συστήματος. Αυτές οι εφαρμογές είναι ο εξυπηρετητής OPC, οι σελίδες του IIS που χρησιμοποιούν τις τεχνολογίες WMI και WSH, τα υποπρογράμματα OPC to DB και OPC Real Time Control καθώς και άλλα προγράμματα όπως το Matlab και το εργαλείο PsExec. Η διαδικασία ρύθμισης της αυτόματης εισόδου του προεπιλεγμένου λογαριασμού γίνεται στο λειτουργικό σύστημα Windows Server 2003 με την προσθήκη νέων κλειδιών στο μητρώο υπό τις υποδείξεις του MSDN (18). Η αυτόματη είσοδος φέρνει στο προσκήνιο πιθανά προβλήματα ασφαλείας καθώς ο Η/Υ κινδυνεύει από πιθανή κακόβουλη φυσική πρόσβαση και για τον λόγο αυτό θα πρέπει να βρίσκεται σε κλειδωμένο και ειδικά διαμορφωμένο χώρο.
2. Αυτόματη εκκίνηση του εξυπηρετητή OPC. Για την αυτοματοποίηση αυτής της ενέργειας αναπτύχθηκε ένα υποπρόγραμμα με την ονομασία Checkbot (βλ. Παράρτημα Α.10) το οποίο με χρήση της τεχνολογίας WMI ψάχνει την διεργασία του εξυπηρετητή OPC στην λίστα προγραμμάτων του Task Manager (13). Αν δεν την βρει, εκτελεί την ρουτίνα εκκίνησης της εφαρμογής κάτω από το console session του προεπιλεγμένου λογαριασμού του συστήματος.
3. Αυτόματη ρύθμιση του AjaxState (βλ. Παράρτημα Α.2) της εφαρμογής OPC to Web στην κατάσταση *on*. Ο πρώτος χρήστης που θα επισκεφτεί την αρχική σελίδα της εφαρμογής OPC to Web μετά από μία επανεκκίνηση του κεντρικού Η/Υ είναι αυτός που θέτει το σύστημα στο 100% της λειτουργίας. Όπως αναφέραμε στην ενότητα 3.4.1.3 τα αποτελέσματα των αιτημάτων που επιστρέφει η δυναμική σελίδα για την κατάσταση λειτουργίας του εξυπηρετητή OPC είναι *true* ή *false*. Η πρώτη επιστροφή *true* για την κατάσταση λειτουργίας του εξυπηρετητή OPC είναι αυτή που θα αλλάξει την κατάσταση του AjaxState στο *on* και θα επιτρέψει στην εφαρμογή OPC to Web την πρόσβαση στα σήματα των συσκευών πεδίου.

3.4.5 Κοινή χρήση μεταβλητών μεταξύ των εφαρμογών

Στο σύστημα υπάρχουν εφαρμογές και υποπρογράμματα που χρησιμοποιούν από κοινού μεταβλητές οι οποίες είναι απαραίτητες για την λειτουργία τους. Για παράδειγμα το χειριστήριο ελέγχου επιθυμητής φωτεινότητας της εφαρμογής OPC to Web πρέπει να είναι σε θέση να το διαβάζουν ο ελεγκτής στο Matlab καθώς και ο μηχανισμός OPC Real Time Control όταν το χρειαστούν, ανεξάρτητα από το αν η διαδικτυακή εφαρμογή OPC to Web λειτουργεί.

Το πρόβλημα της κοινής χρήσης μεταβλητών λύνεται εύκολα με την χρήση του συστήματος αρχείων ή την χρήση του μητρώου του λειτουργικού συστήματος. Μέσω του συστήματος των αρχείων θα ήταν εύκολο όλες οι εφαρμογές και τα υποπρογράμματα να ενημερώνουν ένα απλό αρχείο κειμένου με τις τρέχουσες τιμές των κοινόχρηστων μεταβλητών. Έτσι την στιγμή που κάποια εφαρμογή χρειαστεί την τιμή μίας κοινόχρηστης μεταβλητής αρκεί να ανοίξει και να διαβάσει την αντίστοιχη θέση του αρχείου.

Στην συγκεκριμένη υλοποίηση επιλέχθηκε η χρήση του μητρώου των Windows για την αποθήκευση των κοινόχρηστων μεταβλητών. Για την αρχικοποίηση τους στην registry αναπτύχθηκε ένα αρχείο M-File στο Matlab το οποίο δημιουργεί τα αντίστοιχα κλειδιά στην θέση HKEY_LOCAL_MACHINE\SOFTWARE\<ομάδα> όπου η ονομασία <ομάδα> μπορεί να είναι ένα ψευδώνυμο του συστήματος όπως για παράδειγμα η ονομασία Smartlab. Όταν ο διαχειριστής «τρέξει» το M-file στον κεντρικό Η/Υ αυτό αυτόματα δημιουργεί τις θέσεις και τα κλειδιά στο μητρώο και δίνει στις κοινόχρηστες μεταβλητές τις προεπιλεγμένες τιμές (βλ. Παράρτημα Α.5).

Η εφαρμογή OPC to Web με τα χειριστήρια ελέγχου που ενσωματώνει στην βασική διασύνδεση χρήστη, ενημερώνει τις κοινόχρηστες μεταβλητές στο μητρώο έτσι ώστε τα υπόλοιπα υποπρογράμματα του συστήματος την στιγμή που θα τις χρειαστούν να μπορούν να τις προσπελάσουν.

4 Μελέτη Περίπτωσης

4.1 Το πρότυπο KNX/EIB

Το KNX/EIB (European Installation Bus) αποτελεί ένα πρότυπο δικτύωσης κτιρίων βασισμένο σε ένα ή περισσότερους διαύλους (19). Πρόκειται για ένα βιομηχανικό πρότυπο, το οποίο προτάθηκε από ένα σύνολο εταιριών ηλεκτρολογικών εγκαταστάσεων, οι οποίες συνέστησαν τον ανεξάρτητο επιστημονικό φορέα EIBA (EIB Association). Στην συνέχεια, το 1999 δημιουργήθηκε η Konnex Association ως συνένωση τριών αυτόνομων ενώσεων δικτύων διαχείρισης κτιρίων, της EIBA, της BCI και της EHSA. Έδρα της έγινε το Βέλγιο και τα αντίστοιχα δίκτυα των ενώσεων αυτών ήταν τα EIB, Batibus και EHS. Σήμερα η Konnex Association αποτελείται από περισσότερες από 110 ευρωπαϊκές εταιρίες που ασχολούνται με την ηλεκτρονική, τους αυτοματισμούς, τις ηλεκτρολογικές εγκαταστάσεις, τον κλιματισμό, τις λευκές ηλεκτρικές οικιακές συσκευές κ.α. Επίσης υποστηρίζεται από πολλά πανεπιστήμια και τεχνικά εκπαιδευτήρια σε όλη την Ευρώπη.

Το παλιό πρωτόκολλο EIB και το νέο KNX είναι αμφίδρομα συμβατά γι' αυτό και αναφέρονται με το κοινό όνομα KNX/EIB (στην πραγματικότητα το KNX/EIB είναι στην ουσία το EIB με την δυνατότητα επικοινωνίας με όλες τις KNX συσκευές). Το KNX/EIB λοιπόν, είναι μία σύγχρονη μέθοδος ηλεκτρικών εγκαταστάσεων για τον κτιριακό αυτοματισμό. Πολλοί αναφέρουν τον όρο «έξυπνο σπίτι» για να περιγράψουν τέτοια συστήματα. Η αρχή λειτουργίας είναι η εξής: υπάρχουν σε όλο το κτίριο διάφοροι αισθητήρες (θερμοκρασίας, φωτισμού, παρουσίας ανθρώπων κ.α.), ένας ή περισσότεροι ελεγκτές (PLC, PC) και διάφοροι ενεργοποιητές (φώτα, ηλεκτρικά ρολά, συστήματα θέρμανσης/ψύξης κ.α.). Όλα αυτά διαθέτουν ένα ενσωματωμένο μικρό-ελεγκτή καθώς και το ολοκληρωμένο EIB-TP-UART-IC (11) ώστε να μπορούν να επικοινωνήσουν μεταξύ τους μέσω του δικτύου KNX/EIB με σκοπό να δημιουργηθεί ένα έξυπνο σύστημα κτιριακού αυτοματισμού το οποίο θα μετατρέπει ένα απλό σπίτι σε ένα «έξυπνο σπίτι».

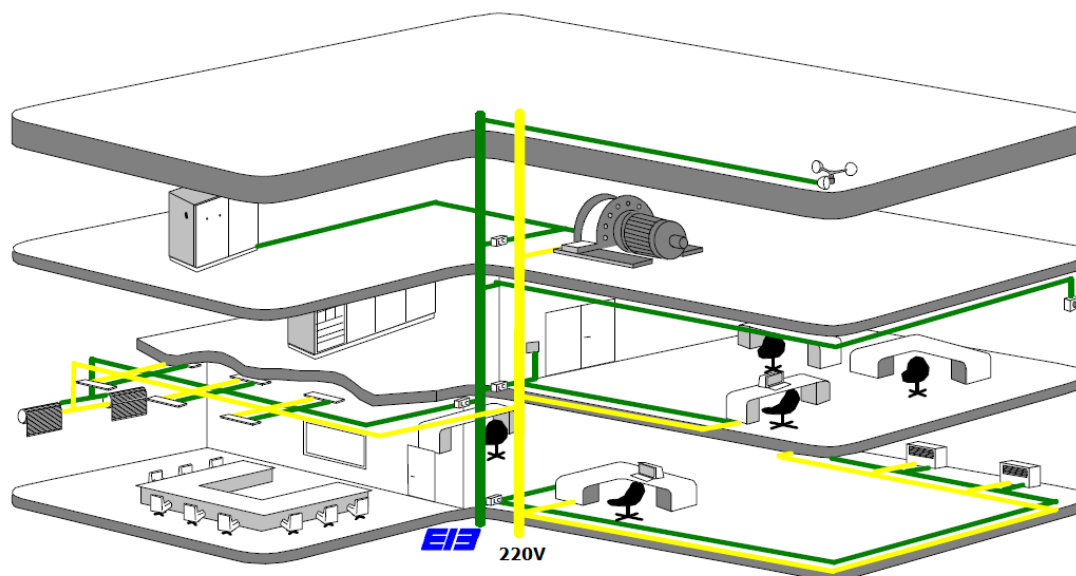
4.1.1 Τρόποι μετάδοσης

Οι κυριότερες υλοποιήσεις εγκαταστάσεων με το πρότυπο KNX/EIB γίνονται με τη χρήση τεσσάρων αρκετά διαδεδομένων φυσικών μέσων:

- Συνεστραμμένο ζεύγος καλωδίων (Twisted Pair)
- Ασύρματη μετάδοση (Radio Transmission)
- Γραμμής ισχύος (Power Line Transmission)
- Ethernet (KNX over IP)

4.1.2 Μετάδοση μέσω συνεστραμμένου ζεύγους

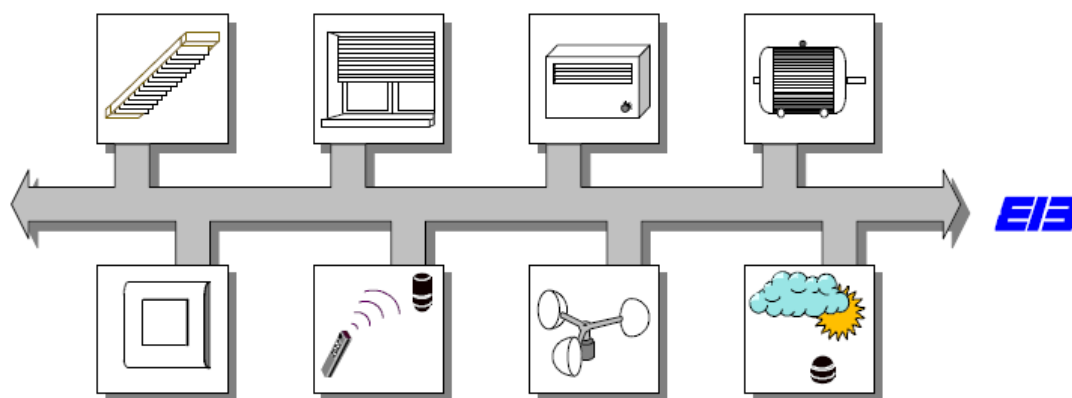
Η μέθοδος αυτή είναι η πιο διαδεδομένη για την υλοποίηση εγκαταστάσεων βασισμένων στο πρότυπο KNX/EIB. Το κόστος χρήσης συνεστραμμένου ζεύγους είναι αρκετά χαμηλό και πληρεί όλες τις προδιαγραφές για ανάπτυξη κτιριακών εγκαταστάσεων. Προσφέρει αναισθησία στο θόρυβο και στις ηλεκτρομαγνητικές παρεμβολές και δεν επηρεάζει την γραμμή μεταφοράς ηλεκτρικού ρεύματος του κτιρίου, εφόσον τοποθετείται παράλληλα με αυτήν. Τέλος, υπάρχει η δυνατότητα της απευθείας τροφοδοσίας των συσκευών από το δίαυλο με την προϋπόθεση ότι καταναλώνουν λιγότερο από 100mW (17). Για τους παραπάνω λόγους, το συνεστραμμένο ζεύγος προτιμάται για καλωδίωση κτιρίων μεγάλου μήκους όταν η εγκατάσταση γίνεται κατά την διάρκεια κατασκευής του κτιρίου. Στο σχήμα 4-1, δίνεται ένα παράδειγμα εγκατάστασης με συνεστραμμένο ζεύγος όπου με πράσινο χρώμα απεικονίζεται ο δίαυλος KNX/EIB και με κίτρινο η γραμμή μεταφοράς του ηλεκτρικού ρεύματος.



Εικόνα 4-1 Εγκατάσταση KNX/EIB με συνεστραμμένο ζεύγος

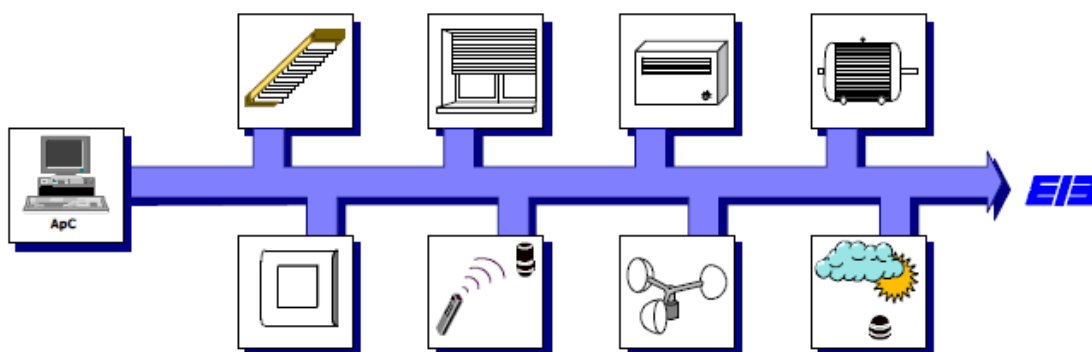
Οι συσκευές που είναι συμβατές με το πρότυπο KNX/EIB, επικοινωνούν μέσω του διαύλου και ονομάζονται συνδρομητές. Το KNX/EIB εξασφαλίζει στον σχεδιαστή, αλλά και στον τελικό καταναλωτή, το πλεονέκτημα της συμβατότητας των παρερχομένων προϊόντων, τώρα, αλλά και στο μέλλον. Μπορούν να τοποθετηθούν σε ράγα ηλεκτρολογικού πίνακα (DINRAIL) ή χωνευτές στον τοίχο, εφόσον αυτό επιτρέπεται. Οι συνδρομητές μπορούν να έχουν αμφίδρομη επικοινωνία μεταξύ τους μέσω μηνυμάτων. Οι προδιαγραφές του διαύλου ορίζουν ένα ανοιχτό σύστημα, στο οποίο κάθε συνδρομητής μπορεί να επικοινωνήσει με οποιοδήποτε άλλο, δίνοντας έτσι την δυνατότητα ο έλεγχος της εγκατάστασης να είναι αποκεντρωμένος και μοιρασμένος στους συνδρομητές που την αποτελούν, ή

Κεντρικός με την ύπαρξη ενός κεντρικού συνδρομητή-ελεγκτή. Στην παρακάτω εικόνα, δίνεται ένα τυπικό παράδειγμα αποκεντρωτικού ελέγχου, όπου οι συνδρομητές έχουν αμφίδρομη επικοινωνία μεταξύ τους.



Εικόνα 4-2 Αποκεντρωτικός έλεγχος εγκατάστασης

Στην περίπτωση κεντρικού ελέγχου, ο κύριος συνδρομητής αναλαμβάνει τον έλεγχο της εγκατάστασης και συχνά είναι ένας ηλεκτρονικός υπολογιστής ή ένα PLC. Στην παρακάτω εικόνα, δίνεται ένα παράδειγμα κεντρικού ελέγχου, όπου κύριος συνδρομητής είναι ένας κεντρικός εξυπηρετητής, στον οποίο είναι συνδεδεμένα τα αισθητήρια και οι ενεργοποιητές μέσω του διαύλου KNX/EIB (11). Αυτή η δομημένη προσέγγιση χρησιμοποιείται για την τοπολογία της εγκατάστασης, κάνοντας εύκολη την διαχείριση των συνδεδεμένων στον δίαυλο συσκευών.



Εικόνα 4-3 Κεντρικός έλεγχος εγκατάστασης

Υπάρχουν δύο είδη συνεστραμμένων ζευγών στο KNX, το TP-0 το οποίο έχει ρυθμό μετάδοσης 4800bits/s και είναι κατάλοιπο του παλιού Batibus και το TP-1 το οποίο έχει ρυθμό μετάδοσης 9600bits/s. Το TP-1 χρησιμοποιείται ευρέως και στην ουσία είναι το παλιό EIB συνεστραμμένου ζεύγους. Το TP-0 και το TP-1 είναι ασύμβατα μεταξύ τους και για ενωθούν χρειάζονται συγκεκριμένη συνδεσμολογία. Για τους περισσότερους ηλεκτρολόγους εγκαταστάτες και μελετητές το TP-1 είναι συνώνυμο του KNX/EIB. Μέσω της τεχνολογίας KNX/EIB γίνεται δυνατός ο έλεγχος μίας συσκευής από πολλαπλά σημεία της εγκατάστασης ενώ παράλληλα, δεν υφίσταται πρόβλημα στη λειτουργία της εγκατάστασης αν ένας κόμβος παρουσιάσει βλάβη.

Είναι εύκολος ο εντοπισμός βλαβών, εφόσον υπάρχει η δυνατότητα δυναμικής επέμβασης στην εγκατάσταση με τη χρήση διαφόρων εργαλείων, αλλά και η πρόσθεση, αφαίρεση ή αλλαγή ενός συνδρομητή.

Με τη χρήση του λογισμικού ETS είναι εύκολος ο επαναπρογραμματισμός και η αλλαγή της τοπολογίας της εγκατάστασης, αν αυτό είναι απαραίτητο να γίνει. Το ETS είναι ένα εργαλείο λογισμικού, που λειτουργεί τόσο σε επίπεδο προγραμματισμού και οργάνωσης της εγκατάστασης, όσο και σε επίπεδο διαχείρισης έργου. Βασικό χαρακτηριστικό του προγράμματος είναι ότι μπορεί να δεχθεί την βάση δεδομένων συνδρομητών οποιουδήποτε συμβατού κατασκευαστή με το πρότυπο EIB. Αυτό επιτρέπει στο σχεδιαστή να χρησιμοποιήσει συσκευές διαφορετικών κατασκευαστών στον σχεδιασμό της εγκατάστασης του.

4.1.3 Ασύρματη Μετάδοση

Είναι γνωστή σαν τεχνολογία EIB-RF και σύμφωνα με αυτή δεν είναι απαραίτητη η εγκατάσταση καινούργιων γραμμών Bus. Χρησιμοποιείται ράδιο-κανάλι (868 MHz), ώστε να μεταδοθούν τα δεδομένα με ασύρματο τρόπο και τόσο τα αισθητήρια, όσο και οι τελικοί αποδέκτες, μπορούν να λειτουργούν με μπαταρίες. Ο ρυθμός μετάδοσης είναι 38,4kbits/s και η κάθε συσκευή μπορεί να απέχει περίπου 300m ελεύθερου χώρου. Εάν απαιτείται μεγαλύτερη απόσταση, τότε επαναλήπτες (repeaters) αναλαμβάνουν την επανάληψη των ράδιο-τηλεγραφημάτων. Η τεχνική αυτή είναι εξαιρετικά κατάλληλη για επέκταση ήδη υπαρχόντων εγκαταστάσεων που έχουν υλοποιηθεί με διαφορετική τεχνολογία. Επίσης το μήκος καλωδίων είναι μικρό έως μηδαμινό. Το κύριο μειονέκτημα της είναι η ανάγκη χρήσης μπαταριών καθώς και η έλλειψη τέτοιων συσκευών στην αγορά (17).

4.1.4 Μετάδοση με χρήση γραμμής ισχύος

Είναι μία τεχνική στην οποία η μεταφορά δεδομένων, αλλά και η τροφοδοσία γίνεται, χρησιμοποιώντας μία διαθέσιμη τροφοδοσία 220/400V. Με αυτή την τεχνολογία δεν είναι πλέον απαραίτητο να απλώνονται παράλληλα οι γραμμές του διαύλου με τις γραμμές ισχύος και είναι η πιο κατάλληλη για την αναβάθμιση ηλεκτρολογικών εγκαταστάσεων παλαιού τύπου χωρίς να χρειάζονται νέα καλώδια. Υπάρχουν δύο είδη αυτής της τεχνικής στο KNX, το PL-110 και το PL-132kHz. Το πρώτο με ρυθμό μετάδοσης 1200bits/s χρησιμοποιεί μετάδοση στα 110kHz και είναι στην ουσία το EIB Powerline. Το δεύτερο με ρυθμό μετάδοσης 2400bits/s χρησιμοποιεί μετάδοση στα 132kHz και είναι στην ουσία το πρωτόκολλο EHS. Το PL-110 και το PL-132 είναι ασύμβατα μεταξύ τους και για ενωθούν χρειάζονται συγκεκριμένη συνδεσμολογία. Το κύριο μειονέκτημα και της τεχνικής αυτής είναι η έλλειψη τέτοιων συσκευών στην αγορά καθώς κάθε χώρα ενδέχεται να έχει διαφορετικής μορφής και αξιοπιστίας παροχή ηλεκτρικής ενέργειας.

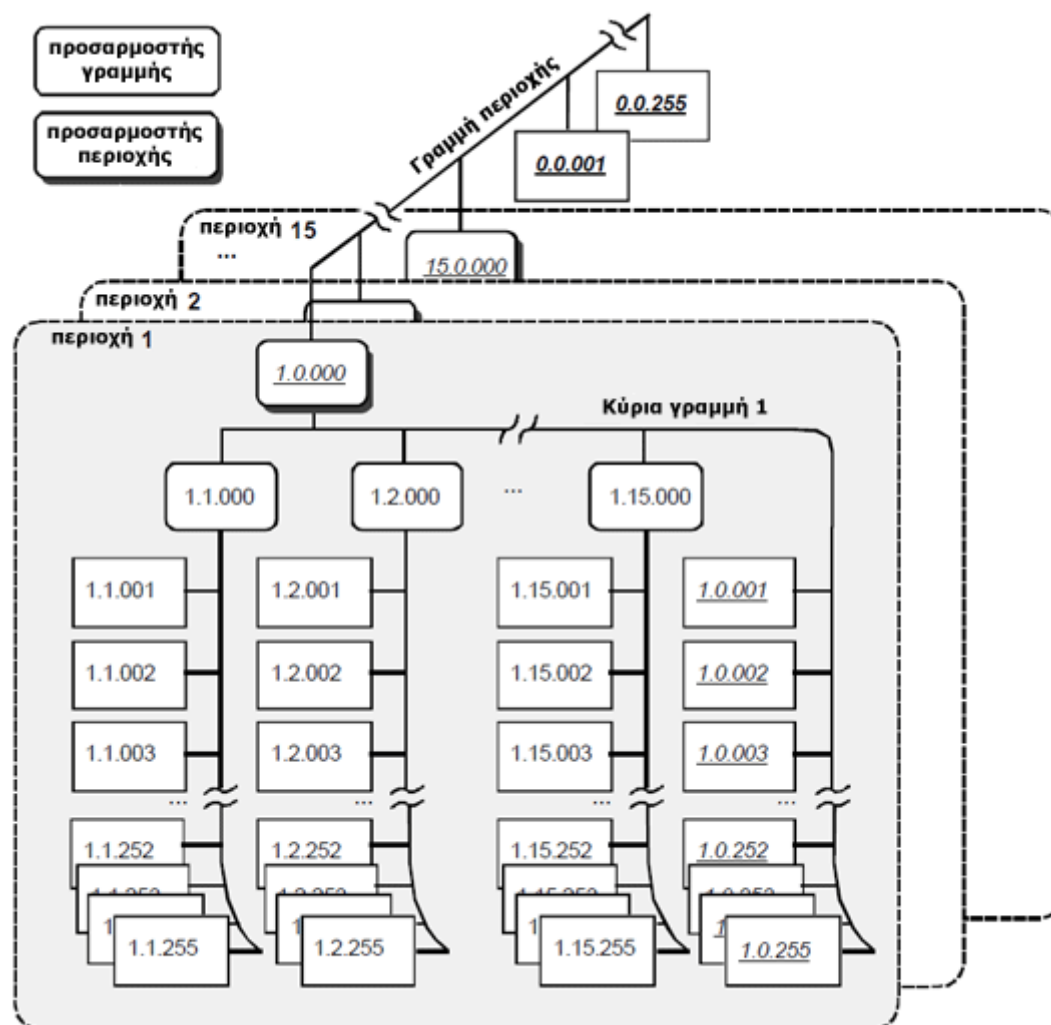
4.1.5 Μετάδοση μέσω Ethernet

Αυτή η τεχνική, γνωστή και ως KNX over IP, είναι η πιο πρόσφατη προσθήκη. Ο ρυθμός μετάδοσης είναι ο ρυθμός μετάδοσης του Ethernet ο οποίος συνήθως είναι της τάξης του Mbit. Το κύριο μειονέκτημα αυτής της τεχνικής είναι το υψηλό της κόστος αφού ένας προσαρμοστής δικτύου Ethernet π.χ. στα 10Mbit/s είναι ακριβότερος από έναν απλό προσαρμοστή για συνεστραμμένο ζεύγος στα 9,6Kbit/s. Γι' αυτό δεν μπορεί να χρησιμοποιηθεί για κάθε συνδρομητή αλλά κυρίως μόνο για την επικοινωνία μιας συμβατικής KNX/EIB εγκατάστασης με υπολογιστές ενός δικτύου Intranet (ίσως και Internet) ή για την επικοινωνία μεταξύ συμβατικών KNX/EIB εγκαταστάσεων με σκοπό να δημιουργηθεί μια μεγαλύτερη εγκατάσταση.

4.1.6 Τοπολογία και οργάνωση διαύλου

Σημαντική παράμετρος στη σχεδίαση και δημιουργία εγκαταστάσεων που βασίζονται σε τεχνολογία KNX/EIB, είναι η ιεράρχηση και η οργάνωση της τοπολογίας της. Αυτό που έχει προταθεί για το σχεδιασμό της καλωδίωσης είναι μία ιεραρχική δενδροειδής δομή. Η γραμμή (line) αποτελεί τη μικρότερη και απλούστερη μονάδα διαύλου σε όλο το σύστημα KNX/EIB. Ο μέγιστος αριθμός συνδρομητών διαύλου που μπορούν να εγκατασταθούν σε μία γραμμή είναι 64. Σε περίπτωση που απαιτείται εγκατάσταση περισσότερων από 64, τότε θα πρέπει να γίνει η χρήση επαναληπτών (repeaters), τότε ο μέγιστος αριθμός των συνδρομητών γίνεται 255. Η μέγιστη απόσταση συσκευών σε γραμμή με χρήση συνεστραμμένου ζεύγους είναι 700 μέτρα με ταχύτητα 9600bps. Η αμέσως μεγαλύτερη δομική μονάδα στην ιεραρχία είναι περιοχή (area). Σε μία περιοχή με τη χρήση ειδικών συσκευών, που λέγονται προσαρμοστές γραμμής (line couplers), μπορούν να συνδεθούν έως 15 γραμμές. Ο συνολικός αριθμός συνδρομητών που μπορούν να φιλοξενηθούν χωρίς επαναλήπτες φτάνει τους 960. Για τη λειτουργία κάθε γραμμής απαιτείται ένα πιστοποιημένο KNX/EIB τροφοδοτικό. Το κάθε τροφοδοτικό αναλαμβάνει την τροφοδοσία των συνδρομητών, που είναι συνδεδεμένοι στην εν λόγω γραμμή. Τέλος, με τη χρήση προσαρμοστών περιοχών (area couplers) μπορούν να συνδεθούν μέχρι και 15 περιοχές μαζί. Έτσι ο συνολικός αριθμός συνδρομητών που μπορούν να φιλοξενηθούν είναι 14400. Κάθε γραμμή λειτουργεί ανεξάρτητα από τις άλλες, έτσι αν συμβεί οποιοδήποτε πρόβλημα σε μια γραμμή (δυσλειτουργία τροφοδοτικού ή συσκευής, βραχυκύκλωμα, προσθήκη νέων συσκευών) τότε δεν θα επηρεαστούν οι υπόλοιπες. Επίσης ο διαχωρισμός του διαύλου σε περιοχές και γραμμές, παρέχει το πλεονέκτημα ότι η κίνηση δεδομένων, σε μία γραμμή ή περιοχή, είναι απομονωμένη και δεν επηρεάζει τη ροή των πληροφοριών στις υπόλοιπες γραμμές και περιοχές, έτσι αυξάνεται η φαινομενική ταχύτητα του δικτύου. Γι' αυτό, όταν οι συσκευές είναι περισσότερες από 64, είναι

προτιμότερο να τοποθετούνται σε πολλές γραμμές και όχι σε μόνο μία με την χρήση επαναληπτών. Το παρακάτω σχήμα αναπαριστά την τοπολογία του KNX/EIB (11).

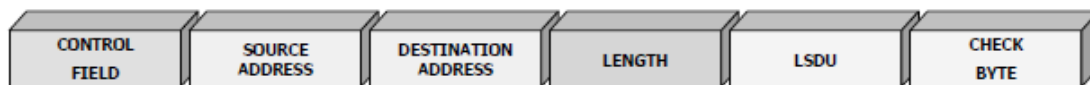


Σχήμα 4-1 Τοπολογία δικτύου KNX/EIB

4.1.7 Μετάδοση πληροφορίας – τηλεγραφήματα

Οι πληροφορίες ανταλλάσσονται μεταξύ των συνδρομητών του διαύλου με τη μορφή τηλεγραφημάτων. Οι πληροφορίες μεταδίδονται συμμετρικά στη γραμμή του διαύλου με ρυθμό μετάδοσης 9600bits/s καθορίζοντας έτσι το μέσο χρόνο μετάδοσης ενός τηλεγραφήματος στα 25ms. Για λόγους ηλεκτρομαγνητικής θωράκισης χρησιμοποιείται, για τη μετάδοση, διαφορά δυναμικού μεταξύ των δύο γραμμών και όχι διαφορά δυναμικού σε σχέση με το δυναμικό της γης. Το KNX/EIB επιβάλλει τα ξεχωριστά μηνύματα πληροφορίας να μεταδίδονται στη γραμμή του διαύλου ακολουθώντας συγκεκριμένη σειρά. Επιπλέον, επιβάλλεται κάθε χρονική στιγμή, ο δίαυλος να χρησιμοποιείται για εκπομπή πληροφορίας από ένα μόνο συνδρομητή. Ο μηχανισμός ελέγχου πρόσβασης που χρησιμοποιείται, είναι ο CSMA/CA, μία παραλλαγή του CSMA/CD του Ethernet. Υπάρχει βέβαια πρόβλεψη

για τα επείγοντα τηλεγραφήματα, σύμφωνα με την οποία δίνεται υψηλή προτεραιότητα σε αυτά έναντι άλλων συμβατικών.



Σχήμα 4-2 Δομή τηλεγραφήματος

Ένα τηλεγράφημα αποτελείται από μία σειρά από χαρακτήρες, οργανωμένους σε πεδία. Η χρήσιμη πληροφορία βρίσκεται στο πεδίο των δεδομένων. Το τηλεγράφημα ελέγχεται για την ορθότητα του μέσω του πεδίου check byte. Τα πεδία διεύθυνσης περιλαμβάνουν τις διευθύνσεις του αποστολέα (source address) και την διεύθυνση των παραληπτών (target address).

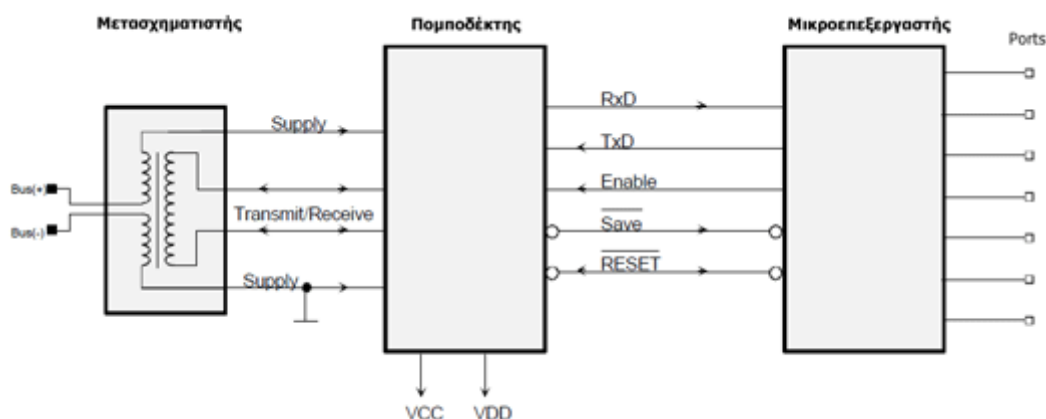
Η διεύθυνση της πηγής είναι πάντα η φυσική διεύθυνση (physical address), η οποία είναι μοναδική για κάθε συνδρομητή. Αυτή καθορίζει την περιοχή και τη γραμμή που βρίσκεται ο κάθε συνδρομητής. Η διεύθυνση προορισμού καθορίζει τη συσκευή ή τις συσκευές, οι οποίες είναι δέκτες του τηλεγραφήματος. Για τη διεύθυνση προορισμού χρησιμοποιούνται διευθύνσεις ομάδας (group address), οι οποίες επιτρέπουν μία ή περισσότερες συσκευές, να παίρνουν μέρος στην ίδια λήψη, ανεξαρτήτως της τοποθεσίας τους στο δίκτυο.

Οι συνδρομητές διαύλου χωρίζονται σε τρεις κατηγορίες: τους αισθητήρες, τους ελεγκτές και τους ενεργοποιητές. Οι αισθητήρες δίνουν εντολές ή πληροφορίες στο σύστημα, οι ελεγκτές αξιοποιώντας τις πληροφορίες δίνουν εντολές (PC, PLC, θερμοστάτες, χρονοδιακόπτες κ.τ.λ.) ενώ οι ενεργοποιητές (κινητήρες, καλοριφέρ, κλιματιστικά κ.τ.λ.) τις εκτελούν. Η διασύνδεση αυτών γίνεται με την αντιστοίχιση της ίδιας διεύθυνσης ομάδας στα αντικείμενα επικοινωνίας τους κάνοντας έτσι δυνατή την αποκατάσταση της μεταξύ τους επικοινωνίας.

4.1.8 Συνδρομητές διαύλου

Ο κάθε συνδρομητής διαύλου αποτελείται από το υποσύστημα προσαρμογής διαύλου (BCU) το οποίο αναλαμβάνει την αποστολή και την λήψη των τηλεγραφημάτων στον δίαυλο και στο υποσύστημα της εφαρμογής. Το υποσύστημα εφαρμογής διαθέτει μικρό-επεξεργαστή και αξιοποιεί την επικοινωνία με τις άλλες συσκευές μέσω του δικτύου για να κάνει κάτι χρήσιμο. Αν η συσκευή είναι τύπου αισθητήρα αναλαμβάνει να δειγματοληπτεί τον αναλογικό αισθητήρα που διαθέτει και να στέλνει το σήμα του περιοδικά στο δίκτυο. Αν είναι συσκευή τύπου ενεργοποιητή τότε για παράδειγμα αναλαμβάνει να ανοίγει και να κλείνει τα ρελέ που διαθέτει με βάση τα μηνύματα που δέχεται και ένα πρόγραμμα που τρέχει στον επεξεργαστή του. Ένας ηλεκτρονικός υπολογιστής μπορεί να γίνει ελεγκτής ή απλά

επόπτης του δικτύου με την προσθήκη ειδικής συσκευής σε μορφή εσωτερικής κάρτας όπως αναπτύχθηκε αναλυτικά στην ενότητα 3.3.1.



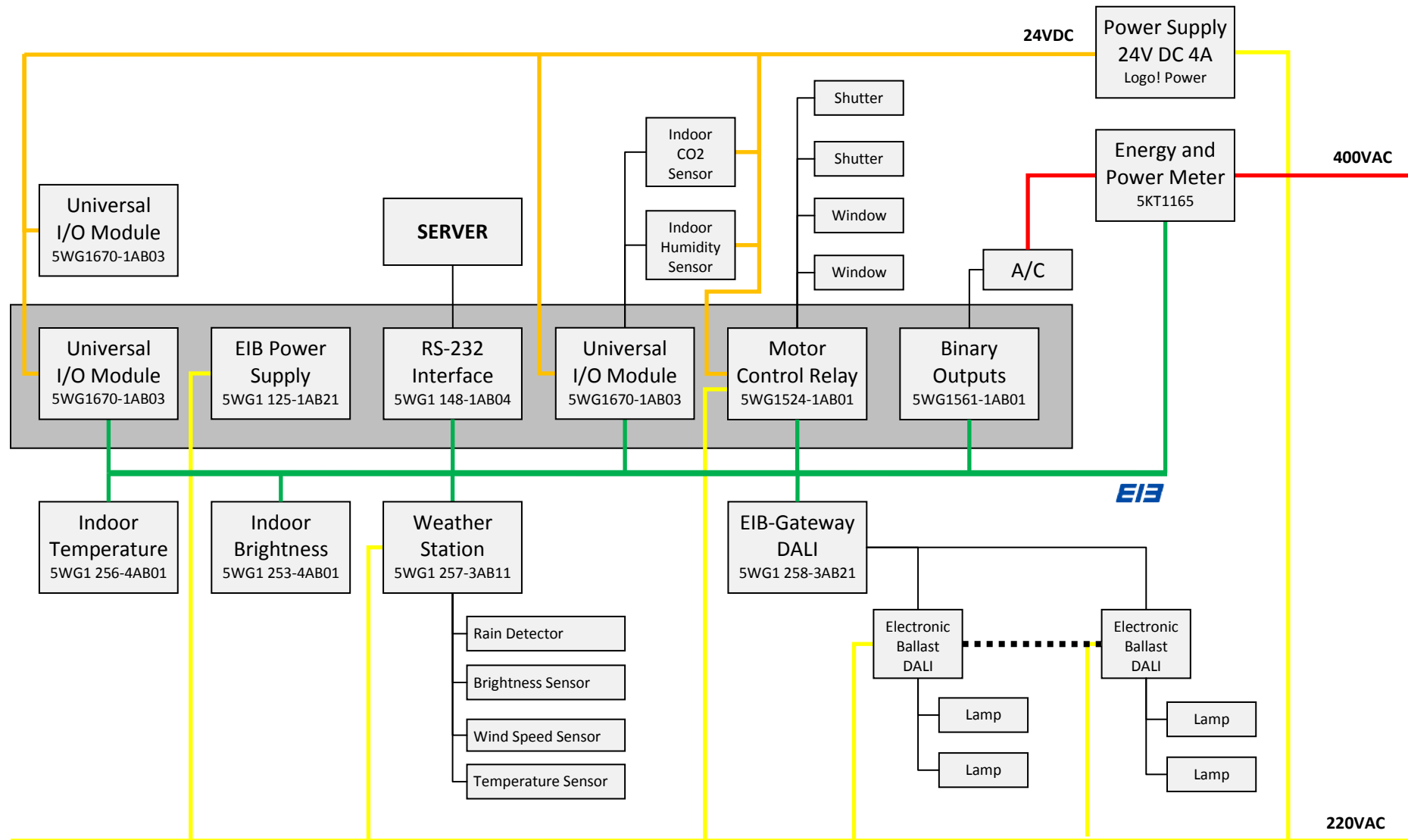
Σχήμα 4-3 Διάταξη BCU

Κάθε συσκευή KNX/EIB μπορεί να προγραμματιστεί με σκοπό να αλλάζουν οι επιμέρους παράμετροι της, όπως πόσο συχνά να στέλνει μηνύματα, τον τύπο δεδομένων που θα χρησιμοποιεί για την αποστολή, που να στέλνει τα μηνύματα κ.τ.λ. Αυτό γίνεται μέσω του λογισμικού ETS.

4.2 Εγκατάσταση KNX/EIB στο εργαστήριο Ελέγχου Βιομηχανικών Συστημάτων

Η εγκατάσταση KNX/EIB υπάρχει στο εργαστήριο Ελέγχου Βιομηχανικών Συστημάτων από το 2007. Όλες οι συσκευές είναι συνδεδεμένες σε μία γραμμή δικτύου KNX/EIB, έτσι το όλο δίκτυο εκφυλίστηκε σε απλό δίαυλο συνεστραμμένου ζεύγους. Η εγκατάσταση περιλαμβάνει συνολικά 11 αισθητήρες και 6 ενεργοποιητές. Ο αριθμός των συσκευών KNX/EIB είναι όμως διαφορετικός επειδή υπάρχουν συσκευές οι οποίες για παράδειγμα ελέγχουν ταυτόχρονα δύο ενεργοποιητές ή είναι συνδεδεμένες με περισσότερους από έναν αισθητήρες. Η αρχιτεκτονική του υλικού της εγκατάστασης όπως καταγράφηκε κατά την διάρκεια εκπόνησης της παρούσας εργασίας δίνεται στο σχήμα 4-4 ενώ το κόστος των συσκευών ανέρχεται περίπου στο ποσό των €11.000 (βλ. παράρτημα Α.12).

Την διαδικασία ελέγχου την διαχειριζόταν ένας Η/Υ με λειτουργικό σύστημα Windows XP Professional που φιλοξενούσε τις υπηρεσίες του εξυπηρετητή EIB OPC της EIBA καθώς και το Matlab R2006b όπου μέσω του OPC Toolbox που περιέχει η συγκεκριμένη έκδοση, έδινε στους χρήστες την δυνατότητα υλοποίησης νέων πελατών OPC (17). Ο συγκεκριμένος Η/Υ αντικαταστήθηκε από τον κεντρικό Η/Υ του συστήματος με τις προδιαγραφές υλικού και λογισμικού που παρουσιάστηκαν στις ενότητες 3.3 και 3.4 αντίστοιχα.



Σχήμα 4-4 Σύνδεση μεταξύ συσκευών πεδίου και γραμμών τροφοδοσίας

4.2.1 Αισθητήρες & Ενεργοποιητές

4.2.1.1 Αισθητήρας εσωτερικής φωτεινότητας

Ο αισθητήρας εσωτερικής φωτεινότητας της Siemens με κωδικό 5WG1 252-4AB02 συνδέεται απευθείας στο δίκτυο KNX/EIB, έχει εύρος μέτρησης 200-1900 lux και ανάλυση 7.68 lux. Είναι τοποθετημένο ψηλά στο κέντρο του εργαστηρίου με κατεύθυνση προς τα κάτω έτσι ώστε να μετράει σωστά την εσωτερική φωτεινότητα του εργαστηρίου. Η συσκευή αυτή τροφοδοτείται από το δίαυλο (20).



Εικόνα 4-4 Αισθητήρας εσωτερικής φωτεινότητας

4.2.1.2 Αισθητήρας εσωτερικής θερμοκρασίας

Για την μέτρηση της θερμοκρασίας χρησιμοποιήθηκε μια συσκευή KNX/EIB η οποία ονομάζεται Combined Fire Alarm. Αυτή η συσκευή της Siemens με το κωδικό 5WG1256-4AB01 ειδικεύεται στην μέτρηση της θερμοκρασίας και του καπνού με σκοπό την αναγνώριση ύπαρξης φωτιάς. Οι έξοδοι της (οι οποίες είναι είσοδοι στο σύστημα μας) μπορεί να είναι η τιμή της θερμοκρασίας, με εύρος τιμών από -25°C έως 70°C με ανάλυση 0.5°C, και η τιμή της μέτρησης του καπνού. Αν και η μέτρηση της θερμοκρασίας δεν είναι καθόλου ακριβής, μπορεί να μας οδηγήσει σε ένα σύστημα που δουλεύει χρησιμοποιώντας λογική τύπου θερμοστάτη όπως στις συμβατικές. Η μέτρηση του καπνού γίνεται μετρώντας την οπτική καθαρότητα του αέρα και για αυτό δεν μπορεί να χρησιμοποιηθεί για την μέτρηση της συγκέντρωσης του CO₂ (το CO₂ είναι διάφανο). Η μέτρηση του καπνού μπορεί να χρησιμοποιηθεί για την αναγνώριση φωτιάς ως ένα επιπλέον χαρακτηριστικό του συστήματος μας. Η συσκευή αυτή τροφοδοτείται από τον δίαυλο.



Εικόνα 4-5 Αισθητήρας εσωτερικής θερμοκρασίας

4.2.1.3 Αισθητήρας συγκέντρωσης CO₂

Ο αισθητήρας συγκέντρωσης CO₂ της Siemens με κωδικό QPA63.2 μπορεί να μετράει δύο μεγέθη, την συγκέντρωση του CO₂ με εύρος μέτρησης 0-2000ppm και ανάλυση 20ppm, και την συγκέντρωση VOC σε κάποιο σχετικό σύστημα μέτρησης. Το VOC είναι η μέτρηση πολλών πτητικών ουσιών και γι' αυτό δεν υπάρχει απόλυτη μονάδα μέτρησης. Αυτός ο αισθητήρας δεν συνδέεται απευθείας στο δίκτυο, αλλά έχει δύο αναλογικές εξόδους μία για το CO₂ και μία για το VOC με τιμές 0-10V για την αναπαράσταση των τιμών που μετράει. Τροφοδοτείται από τροφοδοτικό 24V DC (20).



Εικόνα 4-6 Εσωτερικός αισθητήρας CO₂

4.2.1.4 Αισθητήρας εσωτερικής σχετικής υγρασίας

Ο αισθητήρας εσωτερικής σχετικής υγρασίας της Siemens με κωδικό QFA2000 μπορεί να μετράει τη σχετική υγρασία εσωτερικού χώρου με εύρος μέτρησης 0-100% και ανάλυση 1%. Αυτός ο αισθητήρας δεν συνδέεται απευθείας στο δίκτυο, αλλά έχει μία αναλογική έξοδο με τιμές 0-10V για την αναπαράσταση των τιμών που μετράει. Επίσης τροφοδοτείται από τροφοδοτικό 24V DC.



Εικόνα 4-7 Αισθητήρας εσωτερικής σχετικής υγρασίας

4.2.1.5 Universal I/O module

Για το διάβασμα των τιμών τάσης 0-10V του CO₂ και της σχετικής υγρασίας πρέπει να χρησιμοποιηθεί μια συσκευή η οποία θα έχει δύο αναλογικές εισόδους τάσης DC 0-10V και η οποία θα συνδέεται στο δίκτυο KNX/EIB με σκοπό να στέλνει την τιμή

αυτή στις υπόλοιπες συσκευές. Για τον σκοπό αυτό χρησιμοποιήθηκε το Universal I/O Module N670 της Siemens με κωδικό 5WG1 670-1AB03 το οποίο διαθέτει 2 θύρες που μπορούν να προγραμματιστούν ως είσοδοι ή ως εξόδοι είτε αναλογικές είτε ψηφιακές 2 εισόδους για αισθητήρες τύπου Pt1000 και 2 εξόδους τύπου ρελέ. Η συσκευή αυτή τροφοδοτείται από τον δίαυλο.



Εικόνα 4-8 Συσκευή I/O τύπου DINRAIL

4.2.1.6 Μετεωρολογικός σταθμός

Ο μετεωρολογικός σταθμός της Siemens με κωδικό 5WG1 257-3AB11 είναι μια KNX/EIB συσκευή η οποία μπορεί να συνδεθεί με πολλούς και διαφορετικούς αισθητήρες με σκοπό να μετρηθούν διάφορες περιβαλλοντικές μεταβλητές. Είναι εγκατεστημένος στην ταράτσα ακριβώς πάνω από το εργαστήριο με σκοπό να παίρνει μετρήσεις για το εξωτερικό περιβάλλον του εργαστηρίου. Η συγκεκριμένη συσκευή διαθέτει 4 εισόδους, στις εισόδους αυτές έχουν συνδεθεί ένας αισθητήρας βροχής, ένας φωτεινότητας, ένας θερμοκρασίας και ένας ταχύτητας του αέρα. Η συσκευή αυτή τροφοδοτείται από την συμβατική παροχή ηλεκτρικής ενέργειας 220VAC, 50 Hz και η οποία στην συνέχεια μετασχηματίζεται και τροφοδοτεί τους αισθητήρες (17).



Εικόνα 4-9 Ο μετεωρολογικός σταθμός και οι αισθητήρες του

Ο αισθητήρας βροχής της Siemens με κωδικό 5WG1 258-3AB41 μπορεί να δώσει μόνο δύο τιμές True όταν βρέχει και False όταν δεν βρέχει, έτσι είναι μια ψηφιακή είσοδος στο σύστημα μας.

Ο αισθητήρας φωτεινότητας της Siemens με κωδικό 5WG1 258-3AB31 έχει εύρος μέτρησης από 0 έως 40.000 lux με ανάλυση 156,24 lux και είναι ιδανικός για μέτρηση της εξωτερικής φωτεινότητας.

Ο αισθητήρας θερμοκρασίας της Siemens με κωδικό 5WG1 258-3AB21 έχει εύρος μέτρησης -20°C έως 40°C βαθμούς κελσίου και ανάλυση 0.24°C.

Τέλος ο αισθητήρας ταχύτητας ανέμου της Siemens με κωδικό 5WG1 258-7AB12 έχει εύρος μέτρησης 0-35m/s και ανάλυση 0.27m/s (20).

4.2.1.7 Αισθητήρας εξωτερικής σχετικής υγρασίας

Ο αισθητήρας εξωτερικής σχετικής υγρασίας της Siemens με κωδικό QFA3160 μπορεί να μετράει τη σχετική υγρασία με εύρος μέτρησης 0-100% και ανάλυση 1%. Αυτός ο αισθητήρας δεν συνδέεται απευθείας στο δίκτυο, αλλά έχει μία αναλογική έξοδο με τιμές 0-10V για την αναπαράσταση του επιπέδου της υγρασίας που μετράει. Τροφοδοτείται από τροφοδοτικό 24V DC. Το αναλογικό σήμα 0-10V της εξωτερικής σχετικής υγρασίας το διαβάζει μία συσκευή Universal I/O Module η οποία στέλνει την τιμή στο δίαυλο KNX/EIB (17).



Εικόνα 4-10 Αισθητήρας εξωτερικής σχετικής υγρασίας

4.2.2 Συσκευές ελέγχου των ενεργοποιητών

Οι συσκευές ελέγχου των ενεργοποιητών είναι τρεις και ελέγχουν τον φωτισμό, το σύστημα κλιματισμού, τα παράθυρα και τα σκίαστρα.

4.2.2.1 Ενεργοποιητής φωτισμού

Η συσκευή αυτή με την ονομασία EIB DALI Interface της Siemens με κωδικό 5WG1 141-4AB01 είναι στην ουσία μια γέφυρα επικοινωνίας μεταξύ ενός δικτύου KNX/EIB με ένα σύστημα DALI (Digital Addressable Lighting Interface).

Το DALI είναι ένα σύστημα το οποίο μπορεί να ελέγξει μέχρι 64 διαφορετικές ομάδες φωτιστικών σωμάτων. Στην ουσία τα φωτιστικά σώματα του εργαστηρίου 51 συνολικά χωρισμένα σε 10 ομάδες είναι συνδεδεμένα σε ένα σύστημα DALI και μέσω της συσκευής DALI Interface μπορούν να ελεγχθούν από το δίκτυο KNX/EIB. Τα φώτα ελέγχονται μέσω του KNX/EIB από κοινού ως μία ομάδα και μπορούν είτε απλά να ανοίγουν και να κλείνουν είτε να ρυθμίζεται η φωτεινότητά τους σε 256 διακριτές στάθμες. Όταν ο εξωτερικός φωτισμός είναι 0 lux και εσωτερικά όλα τα φώτα είναι ανοιχτά τότε ο εσωτερικός φωτισμός μπορεί να φτάσει μέχρι τα 300 lux. Η συσκευή αυτή τροφοδοτείται από συμβατική παροχή ηλεκτρικής ενέργειας 220VAC, 50Hz (17).



Εικόνα 4-11 EIB DALI Interface

Στο εργαστήριο υπάρχουν συνολικά 102 λαμπτήρες φθορισμού τύπου T8, οι 94 είναι ονομαστικής ισχύος 58W και οι 8 ονομαστικής ισχύος 18W. Ανά 2 συνδέονται σε μια συσκευή electronic ballast DALI. Συνολικά υπάρχουν εγκατεστημένες 47 συσκευές TRIDONIC pca excel one4all 2x58 W και 4 συσκευές TRIDONIC pca excel one4all 2x18 W.



Εικόνα 4-12 Electronic ballast DALI

4.2.2.2 Ενεργοποιητής παραθύρων και σκιάστρων

Η συσκευή shutter switch N524 N561 της Siemens με κωδικό 5WG1 524-1AB01 συνδέεται με τους ηλεκτρικούς κινητήρες που είναι εγκατεστημένοι στα δύο παράθυρα και στα δύο ηλεκτρικά σκιάστρα. Η συσκευή αυτή προορίζεται για τον έλεγχο ηλεκτρικών κινητήρων για σκιάστρα. Η κίνηση του κάθε παραθύρου και του

κάθε σκιάστρου γίνεται με έναν DC ηλεκτρικό κινητήρα ενώ η πολικότητα της DC τάσης είναι αυτή που καθορίζει το αν θα γίνει άνοιγμα ή κλείσιμο. Επίσης το πόσο θα ανοίξει ή το πόσο θα κλείσει εξαρτώνται από το πόσο χρόνο εφαρμόζεται τάση στον κινητήρα.

Η συσκευή αυτή με την κατάλληλη συνδεσμολογία μπορεί να δέχεται ως είσοδο από το KNX/EIB μια τιμή για την θέση των σκιάστρων και αυτόματα να ανοίγει και να κλείνει τα κατάλληλα ρελέ για τον χρόνο που χρειάζεται με σκοπό τα σκιάστρα να έρθουν στην επιθυμητή θέση. Με παρόμοιο τρόπο μπορεί να ελέγχεται ακόμα και η θέση των περσίδων. Ακόμα, σε αυτήν την συσκευή μπορεί να συνδεθεί ακόμα και κινητήρας για άνοιγμα και για κλείσιμο παραθύρων (αν και το όνομα της συσκευής περιγράφει ότι προορίζεται μόνο για σκιάστρα) αφού τα παράθυρα ελέγχονται με τον ίδιο τρόπο που ελέγχονται τα σκιάστρα. Η συσκευή shutter switch μπορεί να ελέγξει μέχρι 4 ηλεκτρικούς κινητήρες, έτσι σε αυτήν έχουν συνδεθεί οι κινητήρες από δύο παράθυρα και από δύο σκιάστρα. Η συσκευή αυτή τροφοδοτείται από την συμβατική παροχή ηλεκτρικής ενέργειας για να λειτουργήσει και από τροφοδοτικό 24V DC για λειτουργήσουν οι κινητήρες.



Εικόνα 4-13 Shutter switch

4.2.2.3 Ενεργοποιητής συστήματος κλιματισμού

Η συσκευή KNX/EIB της Siemens με την ονομασία Binary Output N561 και κωδικό 5WG1 561-1AB01 είναι στην ουσία τέσσερις (4) έξοδοι τύπου ρελέ οι οποίες μπορούν να ελεγχθούν μέσω του δικτύου KNX/EIB. Η συγκεκριμένη συσκευή δεν απαιτεί γραμμή τροφοδοσίας αλλά τροφοδοτείται από τον δίαυλο. Με τον έλεγχο τεσσάρων (4) ρελέ μπορεί να ελεγχθεί το σύστημα κλιματισμού ρυθμίζοντας έτσι αν και για πόσο θα λειτουργεί στην κατάσταση θέρμανσης ή ψύξης.



Εικόνα 4-14 Binary Output N561

4.2.2.4 Παράθυρα

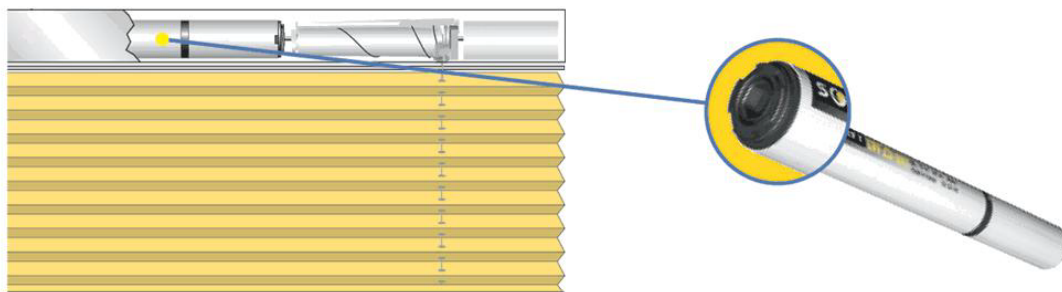
Τα δύο υπό έλεγχο παράθυρα είναι διαστάσεων 3000 0cm το καθένα. Σε κάθε παράθυρο είναι τοποθετημένος ο ηλεκτρικός μηχανισμός ανοίγματος, Turning Sash Drive SHD 50/450 της εταιρίας D+H. Οι κινητήρες λειτουργούν με τάση 24V DC και έχουν μέγιστη ένταση ρεύματος 1A.



Εικόνα 4-15 Μηχανισμός παραθύρων SHD 50/450

4.2.2.5 Σκίαστρα

Στην αίθουσα του εργαστηρίου υπάρχουν δύο υπό έλεγχο σκίαστρα. Το ένα καλύπτει ένα σταθερό άνοιγμα διαστάσεων 175×125 cm και το άλλο ένα άνοιγμα διαστάσεων 175×85 cm. Τα σκίαστρα Somfy Concept 25 λειτουργούν με τάση 24V DC και έχουν μέγιστη ένταση ρεύματος 0.5A



Εικόνα 4-16 Ο ηλεκτρικός μηχανισμός ελέγχου των σκίαστρων

4.2.2.6 Το σύστημα κλιματισμού

Το σύστημα κλιματιστικό της εταιρίας Fyrogenis με κωδικό MFS D12 είναι εγκατεστημένο στο εσωτερικό του εργαστηρίου. Δεν είναι διαθέσιμα τα πλήρη και ακριβή χαρακτηριστικά του συγκεκριμένου συστήματος. Τα στοιχεία τα οποία γνωρίζουμε είναι τα εξής:

- Λειτουργεί με τριφασική παροχή ηλεκτρικής ενέργειας και καταναλώνει περίπου 10KW.
- Σύμφωνα με τον κατασκευαστή η δυνατότητα ψύξης είναι στα 150.000 BTU.

Το συγκεκριμένο σύστημα κλιματισμού διαθέτει χειριστήριο-θερμοστάτη από τον οποίο ο χρήστης μπορεί να το ενεργοποιήσει στο ζεστό ή στο κρύο καθώς και να ενεργοποιήσει την εσωτερική ανακύκλωση του αέρα για να έχει όλος ο χώρος ομοιόμορφη θερμοκρασία. Δεν υπάρχει η δυνατότητα να ρυθμιστεί η ισχύς για παράδειγμα το πλήθος των αντιστάσεων σε περίπτωση θέρμανσης ή οι στρόφες του κινητήρα του συμπιεστή σε περίπτωση ψύξης. Με κατάλληλη διάταξη ρελέ και χειροκίνητο διακόπτη το σύστημα κλιματισμού μπορεί να λειτουργήσει είτε με το παραπάνω χειριστήριο-θερμοστάτη είτε με την KNX/EIB συσκευή με όνομα Binary Output N561 η οποία περιγράφεται παρακάτω.

Το σύστημα κλιματισμού ελέγχεται με 4 καλώδια, την φάση 230VAC, το on/off, το heat/cool και το compressor on/off. Βραχυκυκλώνοντας κάποιο από τα καλώδια με την φάση τότε η αντίστοιχη λειτουργία παίρνει την λογική τιμή 1, αλλιώς παίρνει την τιμή 0. Όταν το on/off είναι στο 0 τότε δεν λειτουργεί τίποτα, όταν είναι στο 1 τότε λειτουργεί ο κινητήρας ανακύκλωσης του αέρα.

Με την προϋπόθεση ότι το compressor on/off είναι στο 1 τότε ανάλογα με την τιμή του heat/cool το σύστημα κλιματισμού μπαίνει σε λειτουργία θέρμανσης ή ψύξης. Αν το heat/cool είναι στο 0 το κλιματιστικό είναι σε λειτουργία ψύξης διαφορετικά αν είναι στο 1 είναι σε λειτουργία θέρμανσης.

Η ανακύκλωση του αέρα είναι αναγκαία μετά από την λειτουργία στο κρύο για να μην βγει εκτός λειτουργίας το σύστημα κλιματισμού από πιθανή δημιουργία πάγου.

4.2.3 Άλλες συσκευές

4.2.3.1 Συσκευή διασύνδεσης RS-232

Η συσκευή διασύνδεσης της Siemens με κωδικό 5WG1 148-1AB04 είναι η διεπαφή μεταξύ του KNX/EIB και του κεντρικού ηλεκτρονικού υπολογιστή μέσω της σειριακής θύρας RS-232. Η συσκευή αυτή χρησιμοποιείται για την ρύθμιση του δικτύου δηλαδή για τον προγραμματισμό των συσκευών μέσω του λογισμικού ETS και για την σύνδεση του δίαυλου KNX/EIB με τον εξυπηρετητή EIB OPC (βλ. Παράρτημα Α.7) ώστε να μπορούν οι πελάτες OPC να αποκτούν πρόσβαση στα σήματα των συσκευών πεδίου. Η συσκευή αυτή τροφοδοτείται από το δίαυλο.



Εικόνα 4-17 Συσκευή διασύνδεσης RS-232

4.2.3.2 Τροφοδοτικά

Εκτός από την τροφοδοσία των συσκευών ισχύος μέσω της συμβατικής παροχής ηλεκτρισμού χρησιμοποιήθηκαν τρία διαφορετικά τροφοδοτικά για τις συσκευές χαμηλής ισχύος. Μέσω της συμβατικής παροχής τροφοδοτούνται τα φώτα, το σύστημα κλιματισμού κάποιες KNX/EIB συσκευές και προφανώς τα υπόλοιπα τροφοδοτικά καθώς και ο κεντρικός Η/Υ.

Χρησιμοποιήθηκε το πιστοποιημένο KNX/EIB τροφοδοτικό EIB Power Supply της Siemens με κωδικό 5WG1 125-1AB21 για να τροφοδοτήσει με συνεχή τάση 24V DC τον δίαυλο. Το μέγιστο ρεύμα του τροφοδοτικού αυτού είναι 640mA και είναι υπεραρκετό για το σύστημα αυτό.



Εικόνα 4-18 EIB Power Supply και το Logo! Power

Επίσης χρησιμοποιήθηκε το τροφοδοτικό "LOGO! Power" για την τροφοδοσία των κινητήρων στα παράθυρα και στα σκίαστρα, καθώς και για τους αισθητήρες CO₂ και εσωτερικής σχετικής υγρασίας. Η τάση αυτού του τροφοδοτικού είναι 24V DC και το μέγιστο ρεύμα είναι 4A.

4.2.3.3 Τριφασικός μετρητής ισχύος και κατανάλωσης ενέργειας

Για την καταγραφή της κατανάλωσης του κλιματιστικού χρησιμοποιήθηκε η συσκευή Energy and Power Meter N165 της Siemens με κωδικό 7KT1 165. Ο μετρητής αυτός είναι συσκευή KNX/EIB. Καταγράφει την συνολική κατανάλωση σε Wh και την τιμή της ισχύος σε W σε κάθε φάση. Η συνδεσμολογία γίνεται με τη χρήση μετασχηματιστών έντασης ρεύματος. Συγκεκριμένα χρησιμοποιήθηκαν τρεις (3) μετασχηματιστές 100/5 A.



Εικόνα 4-19 Energy and Power Meter

4.3 Βελτιώσεις συστήματος

Για μεγαλύτερη αξιοπιστία της λειτουργίας του συστήματος από απόσταση έγιναν κάποιες βελτιώσεις στην υπάρχουσα εγκατάσταση KNX/EIB. Επίσης αναπτύχθηκε βοηθητικό λογισμικό για την αντιμετώπιση κάποιων τεχνικών προβλημάτων στον αισθητήρα εξωτερικής σχετικής υγρασίας και τέλος έγινε βελτίωση στον έλεγχο λειτουργίας των σκιάστρων κατά τις απογευματινές ώρες με την προσθήκη ενός ακόμα αισθητήρα μέτρησης της εσωτερικής φωτεινότητας προσανατολισμένο στα δυτικά παράθυρα της αίθουσας.

4.3.1 Επαναπρογραμματισμός συσκευών στην εγκατάσταση

Ο επαναπρογραμματισμός μερικών συσκευών της εγκατάστασης έγινε με σκοπό να δημιουργηθούν νέες διευθύνσεις που θα βοηθήσουν στην καλύτερη λειτουργία και εποπτεία του συστήματος από απόσταση. Για τον επαναπρογραμματισμό των συσκευών χρησιμοποιήθηκε το λογισμικό ETS (EIB Tool Software). Οι συσκευές που επαναπρογραμματίστηκαν ήταν ο τριφασικός μετρητής ισχύος και κατανάλωσης ενέργειας, ο δεύτερος αισθητήρας μέτρησης εσωτερικής φωτεινότητας και ο αισθητήρας εσωτερικής θερμοκρασίας ενώ έγινε και μετονομασία της ομάδας διευθύνσεων "energy & power" σε "energy and power" λόγω του ότι ο χαρακτήρας & είναι δεσμευμένος από την γλώσσα προγραμματισμού Visual Basic.

4.3.1.1 Το λογισμικό ETS (EIB Tool Software)

Για τον σχεδιασμό και την ανάπτυξη εγκαταστάσεων KNX/EIB χρησιμοποιείται το εργαλείο λογισμικού ETS, το οποίο λειτουργεί τόσο σε επίπεδο προγραμματισμού και οργάνωσης όσο και σε επίπεδο διαχείρισης έργου.

Το ETS είναι ένα πρόγραμμα το οποίο έχει σχεδιαστεί με γνώμονα την ευελιξία και την επεκτασιμότητα, έχοντας τέτοια δομή ώστε να υποστηρίζει και μελλοντικές υλοποιήσεις της τεχνολογίας KNX/EIB. Βασικό χαρακτηριστικό του προγράμματος είναι ότι μπορεί να δεχτεί την βάση δεδομένων συνδρομητών οποιουδήποτε κατασκευαστή επιτρέποντας στο σχεδιαστή να χρησιμοποιήσει σε μια εγκατάσταση συσκευές διαφορετικών κατασκευαστών συμβατές με το πρότυπο KNX/EIB, οι οποίες θα διαχειρίζονται με ενιαία διεπαφή.

Για να μπορεί να επικοινωνήσει το πρόγραμμα με την εγκατάσταση πρέπει να υπάρχει φυσική διασύνδεση του Η/Υ που έχει εγκατεστημένο το ETS με την συσκευή διασύνδεσης KNX/EIB μέσω της θύρας RS-232.

Ένα από τα σπουδαιότερα τμήματα του ETS είναι η επιλογή σχεδιασμού έργου. Από εκεί ξεκινά η δημιουργία κάθε νέου έργου και μπορούν να γίνουν όλες οι αλλαγές και οι τροποποιήσεις ενός υπάρχοντος. Επιπλέον στο τμήμα αυτό δημιουργούνται

οι διευθύνσεις ομάδας όπως επίσης δηλώνεται η τοπολογία των συσκευών και ορίζονται οι σχέσεις των αντικειμένων επικοινωνίας με τις διευθύνσεις ομάδας. Παράλληλα, μπορεί κανείς να δει σε πιο σημείο του κτιρίου βρίσκεται κάποια συσκευή και να του παρουσιαστούν οι σχέσεις των στοιχείων επικοινωνίας με τις διευθύνσεις ομάδας.

Στο σημείο αυτό πρέπει να αναφερθεί ότι δεν είναι αναγκαίο να γίνει η όλη σχεδίαση στον Η/Υ όταν αυτός είναι συνδεδεμένος στην εγκατάσταση, αλλά κάπου αλλού και στο τέλος να γίνει η μεταφορά του προγράμματος σε κάθε μια συσκευή μεταφέροντας τον φορητό υπολογιστή στο χώρο εγκατάστασης.

Πρέπει να τονιστεί ότι το ETS χρησιμοποιείται μόνο κατά την εγκατάσταση ή την αναβάθμιση του συστήματος και δεν είναι απαραίτητο για την λειτουργία του συστήματος σε πραγματικό χρόνο. Ο προγραμματισμός στο ETS συνίσταται σε:

- Προσδιορισμό των συσκευών.
- Καθορισμό της τοπολογίας.
- Διευθυνσιοδότηση των συσκευών.
- Καθορισμό των παραμέτρων λειτουργίας των συσκευών.
- Καθορισμό των διευθύνσεων ομάδας και της ιεραρχίας τους.

Για κάθε συσκευή KNX/EIB αντλούμε το πρόγραμμα της από μια βάση δεδομένων, καθορίζουμε τις παραμέτρους, δίνουμε μία διεύθυνση και προσδιορίζουμε την τοπολογία της. Στην συνέχεια ακολουθεί ο προγραμματισμός της συσκευής και η ενημέρωση του αρχείου .esf του εξυπηρετητή OPC της EIBA με την διεύθυνση της νέας συσκευής.

4.3.1.2 Τριφασικός μετρητής ισχύος και κατανάλωσης ενέργειας

Ο τριφασικός μετρητής ισχύος και κατανάλωσης ενέργειας του συστήματος κλιματισμού δίνει την δυνατότητα καταγραφής της τιμής ισχύος σε Watt για κάθε φάση. Έτσι με τις τρεις νέες διευθύνσεις που προστέθηκαν μέσω του ETS, μπορεί να επαληθευτεί άμεσα από την εφαρμογή OPC to Web η κατάσταση λειτουργίας του κλιματιστικού από τις τρέχουσες τιμές ισχύος της κάθε φάσης.

Πριν το επαναπρογραμματισμό της συσκευής και την προσθήκη των νέων διευθύνσεων η άμεση επαλήθευση της λειτουργίας του κλιματιστικού από απόσταση δεν ήταν δυνατή αφού έπρεπε ο χρήστης να περιμένει την μεταβολή της θερμοκρασίας του χώρου.

Την δυνατότητα μέτρησης της τιμής ισχύος την αξιοποιεί και ο ελεγκτής θερμικής άνεσης ώστε να είναι σε θέση να επαληθεύει την κατάσταση λειτουργίας του συστήματος κλιματισμού με δύο ανεξάρτητους τρόπους (βλ. Ενότητα 4.4.2).

4.3.1.3 Αισθητήρας εσωτερικής φωτεινότητας με προσανατολισμό στα δυτικά παράθυρα

Για την βελτίωση της λειτουργίας των σκιάστρων τις απογευματινές ώρες όπου παρατηρείται το φαινόμενο της απευθείας εισόδου των ακτίνων του ηλίου στην αίθουσα, εγκαταστάθηκε σε δυτικό παράθυρο ο δεύτερος αισθητήρας εσωτερικής φωτεινότητας. Ο προγραμματισμός της συσκευής και η προσθήκη της διεύθυνσης στον εξυπηρετητή EIB OPC δίνει στο σύστημα ένα ακόμα σήμα με την τρέχουσα τιμή σε lux της φωτεινότητας της δυτικής πλευράς του κτιρίου (βλ. Ενότητα 4.4.1 και Παράρτημα Α.3).

4.3.1.4 Αισθητήρας εσωτερικής θερμοκρασίας

Για την ανάπτυξη του σεναρίου που αναλαμβάνει την αποστολή ειδοποιήσεων σε περίπτωση ανίχνευσης καπνού στον χώρο του εργαστηρίου προχωρήσαμε σε επαναπρογραμματισμό του αισθητήρα εσωτερικής θερμοκρασίας και στην προσθήκη νέων διευθύνσεων στον εξυπηρετητή OPC. Ο αισθητήρας όταν αντιληφθεί την ύπαρξη καπνού δίνει την τιμή TRUE στην νέα διεύθυνση του OPC Server. Ο μηχανισμός Real Time Control που εκτελεί τα σενάρια του συστήματος ακόμα και όταν το σύστημα είναι κλειστό αναλαμβάνει να εκτελέσει τον κώδικα του αντίστοιχου σεναρίου. Στην έκδοση V1.01-02 του συγκεκριμένου σεναρίου οι ενέργειες που εκτελούνται από τον μηχανισμό OPC Real Time Control είναι τρεις:

- Ηχεί συναγερμός στο εσωτερικό του εργαστηρίου.
- Αποστολή σχετικού Email σε προεπιλεγμένη ηλεκτρονική διεύθυνση.
- Αποστολή σχετικού SMS σε προεπιλεγμένο αριθμό κινητού τηλεφώνου.

Για την ενεργοποίηση του συναγερμού το σενάριο στέλνει την τιμή TRUE σε μια συγκεκριμένη διεύθυνση του αισθητήρα εσωτερικής θερμοκρασίας που εμπεριέχει το Buzzer, ενώ για τις αποστολές των ειδοποιήσεων χρησιμοποιούνται οι συναρτήσεις sSms και sEmail που περιλαμβάνονται στον μηχανισμό OPC Real Time Control.

```
If sRead("2/1/1") = True Then  
sWrite "2/1/6",1,0  
sEmail "katsigarakis@gmail.com","Fire Alarm","Indoor Temp:" &_  
sRead("2/1/0") & "C, Time:" & Now(), 180  
sSms "306938321974","Fire Alarm","Indoor Temp:" & sRead("2/1/0") &_  
"C, Time:" & Now(), 180  
End If
```

Και στις δύο συναρτήσεις το χρονικό διάστημα των 180 λεπτών ορίζεται έτσι ώστε η ίδια ειδοποίηση να μην ξανασταλθεί αν δεν περάσουν οι τρεις ώρες. Σε περίπτωση που μετά από τρεις ώρες ισχύει η κατάσταση ανίχνευσης καπνού τότε θα την ξαναστείλει.

4.3.2 Ανάπτυξη εφαρμογής sHumidity

Λόγω κάποιων τεχνικών προβλημάτων που αντιμετώπιζε στην αρχή ο αισθητήρας εξωτερικής σχετικής υγρασίας της Siemens αναπτύχθηκε μία εφαρμογή σε VBScript η οποία διαβάζει ανά τακτά χρονικά διαστήματα την τρέχουσα τιμή της σχετικής υγρασίας που καταγράφεται από τον κοντινότερο μετεωρολογικό σταθμό. Την τιμή αυτή η εφαρμογή sHumidity την αποθηκεύει στην προκαθορισμένη περιοχή κοινόχρηστων μεταβλητών του μητρώου του λειτουργικού συστήματος.

Όταν ένα υποπρόγραμμα του συστήματος ζητήσει την τρέχουσα τιμή της εξωτερικής σχετικής υγρασίας, του επιστρέφεται η τελευταία αποθηκευμένη τιμή από την προκαθορισμένη περιοχή του μητρώου. Ο κοντινότερος μετεωρολογικός σταθμός βρίσκεται εντός του Πολυτεχνείου Κρήτης και η εφαρμογή sHumidity επισκέπτεται την διεύθυνση <http://www.meteo.gr/stations/chania/> από την οποία και αντλεί την αντίστοιχη τιμή.

Ο κώδικας VBScript δημιουργεί το αντικείμενο msxml2.xmlhttp.3.0 με την μέθοδο Create.Object για την σάρωση της σελίδας και στην συνέχεια αφού εντοπίσει την τιμή την αποθηκεύει με χρήση της τεχνολογίας WSH στην προκαθορισμένη θέση του μητρώου (βλ. Παράρτημα Α.6).

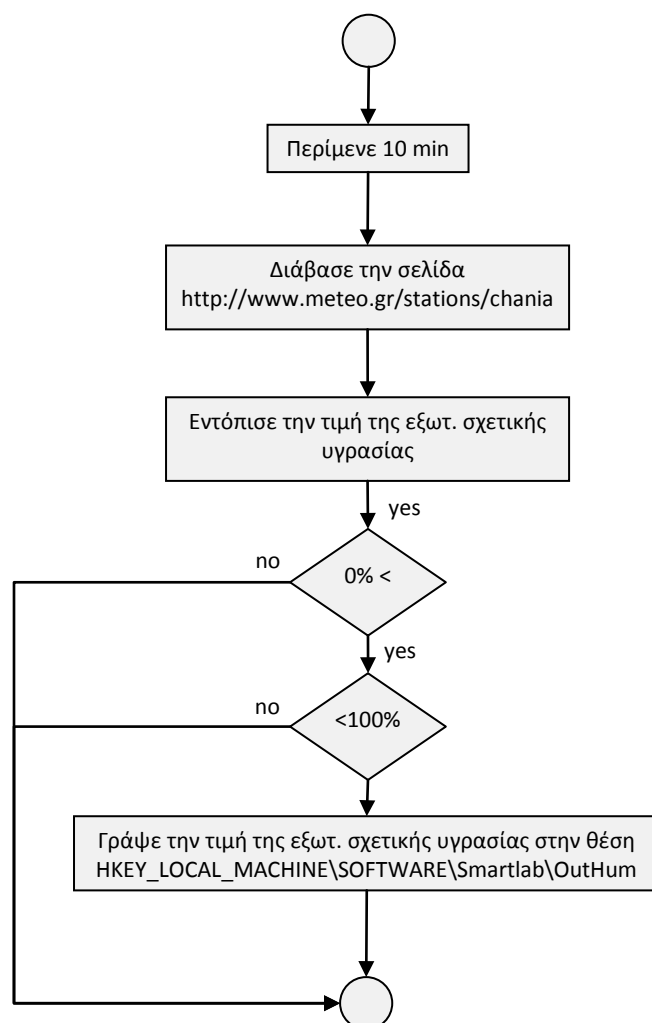
Η διαδικασία αυτή γίνεται ανά δέκα λεπτά μέσω της εφαρμογής προγραμματισμού εργασιών του λειτουργικού συστήματος Windows Server 2003 που αναλαμβάνει την εκκίνηση της εφαρμογής sHumidity αυτόματα κάθε φορά που θα περάσει το διάστημα παύσης των δέκα λεπτών.

Σε περίπτωση προβλήματος της σελίδας ή της σύνδεσης η εφαρμογή δεν στέλνει τίποτα και έτσι τα υποπρογράμματα του συστήματος χρησιμοποιούν την τελευταία τιμή που είχε αποθηκευτεί.

Κατά την διάρκεια εκπόνησης της διπλωματικής εργασίας το πρόβλημα του αισθητήρα εξωτερικής σχετικής υγρασίας επιδιορθώθηκε και έτσι το σύστημα τελικά διαθέτει δύο σήματα εξωτερικής σχετικής υγρασίας.

Το σήμα της εφαρμογής sHumidity το αφήσαμε ενεργό στο σύστημα για να παρουσιαστεί μια εναλλακτική οικονομική μέθοδος με την οποία εφαρμογές κτιριακού αυτοματισμού μπορούν να αντλούν από τους κοντινότερους μετεωρολογικούς σταθμούς μέσω του διαδικτύου, τα κλιματολογικά δεδομένα της ευρύτερης περιοχής και να τα στέλνουν στους ελεγκτές εξοικονόμησης ενέργειας που διαχειρίζονται. Με την ευρύτερη έννοια η συγκεκριμένη υλοποίηση μπορεί να χρησιμοποιηθεί από ελεγκτές εξοικονόμησης ενέργειας σε κτίρια που διαχειρίζονται διαδικασίες ελέγχου με διαφορετικά πρότυπα δικτύωσης. Για παράδειγμα ένας μετεωρολογικός σταθμός που υποστηρίζει το πρότυπο KNX/EIB μπορεί να στέλνει

τα κλιματολογικά δεδομένα στα υποσυστήματα ενός βιομηχανικού δικτύου όπως για παράδειγμα το Profibus.



Σχήμα 4-5 Εφαρμογή sHumidity

4.3.3 Αλγόριθμος προβλεπτικού ελέγχου

Στο εργαστήριο Ελέγχου Βιομηχανικών Συστημάτων εφαρμόζεται ένας αλγόριθμος προβλεπτικού ελέγχου, ο οποίος έχοντας γνώση της παρούσας κατάστασης προβλέπει τη συμπεριφορά του συστήματος σε όλα τα δυνατά σενάρια λειτουργίας των ενεργοποιητών και επιλέγει το καλύτερο από αυτά με γνώμονα την ελάχιστη κατανάλωση ενέργειας και τις καλύτερες δυνατές συνθήκες μέσα στην αίθουσα (17). Ο αλγόριθμος έχει υλοποιηθεί στο Matlab και τον διαχειρίζεται η εφαρμογή OPC to Web σαν τον βασικό προεπιλεγμένο ελεγκτή. Υποστηρίζεται η τμηματική ενεργοποίηση του σε τρία ανεξάρτητα τμήματα (κλιματισμός, φωτεινότητα και ποιότητα αέρα) ενώ μπορεί να εκτελείται ταυτόχρονα με τον μηχανισμό OPC Real Time Control χωρίς καμία δυσλειτουργία αφού πρόκειται για δύο διαφορετικούς πελάτες OPC (βλ. Παράρτημα Α.1, Εικόνα Α.1-2).

4.4 Εφαρμογές OPC Real Time Control

4.4.1 Έλεγχος οπτικής άνεσης

Για τον έλεγχο της φωτεινότητας της αίθουσας υλοποιήθηκε ένας ελεγκτής πραγματικού χρόνου που τον διαχειρίζεται ο μηχανισμός OPC Real Time Control. Χρησιμοποιεί την διασύνδεση χρήστη (interface) και τα χειριστήρια ελέγχου της εφαρμογής OPC to Web για την εκκίνηση, τον τερματισμό και τον καθορισμό των παραμέτρων που χρειάζεται. Το σύστημα αυτόματου ελέγχου που υλοποιήθηκε έχει τρεις εισόδους οι οποίες είναι:

- Η επιθυμητή τιμή εσωτερικής φωτεινότητας από το αντίστοιχο χειριστήριο καθορισμού του set point.
- Η μετρήσιμη τιμή εσωτερικής φωτεινότητας από το σήμα του αντίστοιχου αισθητήρα.
- Η μετρήσιμη τιμή εξωτερικής φωτεινότητας από τον μετεωρολογικό σταθμό της εγκατάστασης.

Ενώ οι εξόδοι του συστήματος ελέγχου είναι οι εξής:

- Ο καθορισμός της νέα θέσης στα στόρια των παραθύρων.
- Ο καθορισμός της νέας τιμής στο dimmer τεχνικού φωτισμού.

Ο αισθητήρας που μετράει την τιμή της εσωτερικής φωτεινότητας είναι τοποθετημένος ψηλά στο κέντρο του εργαστηρίου. Ο ελεγκτής με την χρήση των ειδικών συναρτήσεων που κατασκευάστηκαν και περιλαμβάνονται στον μηχανισμό OPC Real Time Control μπορεί να αποκτήσει πρόσβαση στο σήμα του αισθητήρα μέσω του εξυπηρετητή OPC της EIBA. Την εξωτερική φωτεινότητα την μετράει ο αισθητήρας εξωτερικής φωτεινότητας που περιλαμβάνεται στους αισθητήρες του μετεωρολογικού σταθμού της εγκατάστασης που βρίσκεται στην ταράτσα.

Η πρώτη από τις δύο εξόδους του ελεγκτή αφορά την νέα θέση στα στόρια των δύο παραθύρων και παίρνει τιμές 1 – 100% με βήμα 25% και με προστασία εγγραφής του αντίστοιχου σήματος τα 3 δευτερόλεπτα. Επειδή ο μηχανισμός OPC Real Time Control εκτελεί τον ελεγκτή συνεχώς μέσα σε βρόχο While(1)...Wend η προστασία εγγραφής δεν αφήνει την συνάρτηση sWrite να γράψει νέα τιμή στο σήμα του ενεργοποιητή αν δεν περάσει το χρονικό διάστημα που ορίζεται στο όρισμα delay της συνάρτησης. Επίσης να σημειωθεί ότι για τον ενεργοποιητή των σκιάστρων η τιμή 100% σημαίνει πλήρως κλειστά ενώ η τιμή 1% σημαίνει πλήρως ανοιχτά. Η δεύτερη έξοδος αφορά την τιμή του dimmer του εσωτερικού φωτισμού της αίθουσας το οποίο παίρνει τιμές 0 – 100% με βήμα 1% ή 5% ενώ η προστασία εγγραφής για τον συγκεκριμένο ενεργοποιητή έχει οριστεί στο 1 δευτερόλεπτο. Αν το απόλυτο σφάλμα της μετρήσιμης τιμής του αισθητήρα εσωτερικής φωτεινότητας

από το προκαθορισμένο set point του χειριστήριο ελέγχου είναι μεγαλύτερο από 50 lux τότε το βήμα ορίζεται στο 5% ενώ διαφορετικά όταν ο ελεγκτής πλησιάζει την επιθυμητή τιμή του set point ορίζεται αυτόματα στο 1%. Επειδή αρκετά φωτιστικά σώματα της αίθουσας είναι εκτός λειτουργίας λόγω καμένων λαμπτήρων ή χαλασμένων εκκινητών η δυνατότητα φωτισμού που έχει είναι πολύ περιορισμένη ειδικά κατά τις απογευματινές και νυχτερινές ώρες που με δυσκολία ξεπερνά τα 240 lux με το dimmer στην θέση 100%. Σε μία ηλιόλουστη μέρα με τα σκίαστρα πλήρως ανοιχτά και τα φωτιστικά σώματα σε πλήρη λειτουργία η μετρήσιμη τιμή με δυσκολία ξεπερνά τα 450 lux. Το χειριστήριο του set point δέχεται τιμές από 0 – 600 lux γιατί ενδέχεται μελλοντικά να επισκευαστούν μερικά από τα φωτιστικά σώματα. Προς το παρόν αν θέσουμε το χειριστήριο σε τιμές μεγαλύτερες των μέγιστων τιμών που αναφέραμε το σύστημα αυτόματου ελέγχου απλά θα λειτουργεί τα φωτιστικά σώματα στο 100% χωρίς ποτέ να καταφέρει να φτάσει την επιθυμητή τιμή.

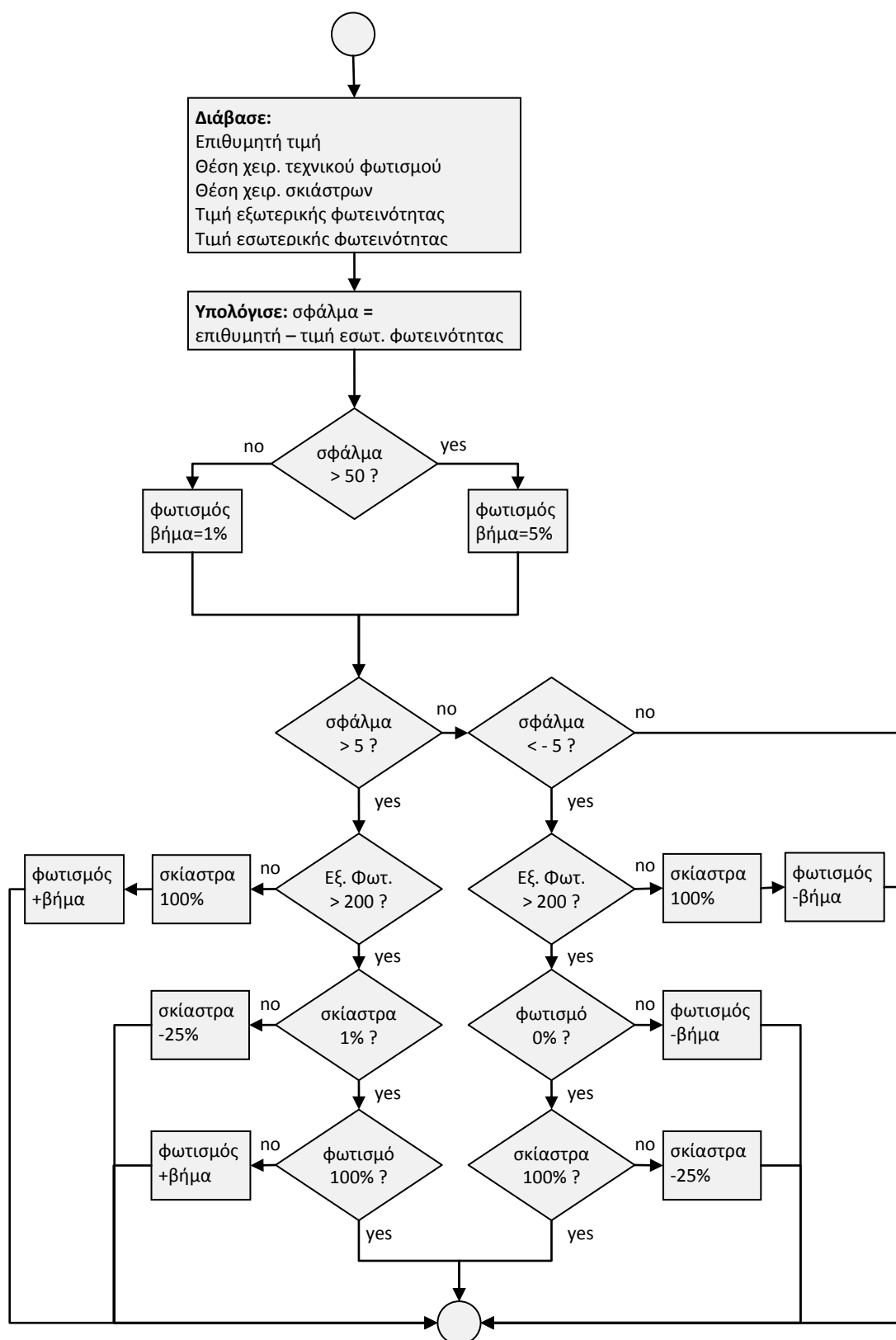
4.4.1.1 Έλεγχος σκιάστρων

Χρησιμοποιώντας τα σκίαστρα των παραθύρων μπορούμε να κάνουμε το χώρο του εργαστηρίου να φωτίζεται ή όχι με φυσικό τρόπο. Αυτού του είδους ο φωτισμός ονομάζεται φυσικός φωτισμός και προσφέρει εξοικονόμηση ενέργειας. Στον αλγόριθμο ελέγχου για την οπτική άνεση δίνουμε προτεραιότητα στο φυσικό φωτισμό έναντι των φωτιστικών. Αυτό σημαίνει ότι τα φώτα ανάβουν μόνο όταν δεν επαρκεί το φως της ημέρας και χρειαζόμαστε επιπλέον φωτισμό για να φτάσουμε την επιθυμητή τιμή που έχουμε ορίσει στο χειριστήριο του set point.

Τα σκίαστρα ρυθμίστηκαν ώστε όταν χρειαστεί να προβούν σε κάποια ενέργεια (ανοιχτά / κλειστά) αυτή να γίνεται σε ποσοστό 25% σε σχέση με την πλήρη διαδρομή που μπορούν να διανύσουν, συνολικά έχουμε 4 θέσεις από τις οποίες μπορούν να περάσουν. Έτσι, αν για παράδειγμα βρίσκονται μισάνοιχτα 50% τότε οι επόμενες πιθανές θέσεις που μπορούν να πάνε είναι ή στο 25% αν χρειαστεί να αυξήσουμε τη φωτεινότητα του χώρου ή στο 75% αν χρειαστεί να μειώσουμε τη φωτεινότητα του χώρου.

4.4.1.2 Έλεγχος φωτιστικών

Ο φυσικός φωτισμός μπορεί να προσφέρει καλύτερη ποιότητα φωτός και εξοικονόμηση ενέργειας όμως δεν επαρκεί για να επιτευχθεί η τιμή της επιθυμητής εσωτερικής φωτεινότητας. Για το λόγο αυτό, και αφού έχουμε ανοίξει πλήρως τα σκίαστρα, αρχίζει ο ελεγκτής να ανοίγει σταδιακά τον τεχνικό φωτισμό. Αυτό γίνεται με βήματα της τάξεως του 5% ή 1% (ανάλογα με την τιμή του απόλυτου σφάλματος της διαφοράς της μετρήσιμης τιμής από την επιθυμητή τιμή) μέχρις ότου φτάσει την επιθυμητή τιμή εσωτερικής φωτεινότητας. Ο αλγόριθμος ελέγχου οπτικής άνεσης παρουσιάζεται στο σχήμα 4-6 (V3.03-02) (βλ. Παράρτημα Α.3).



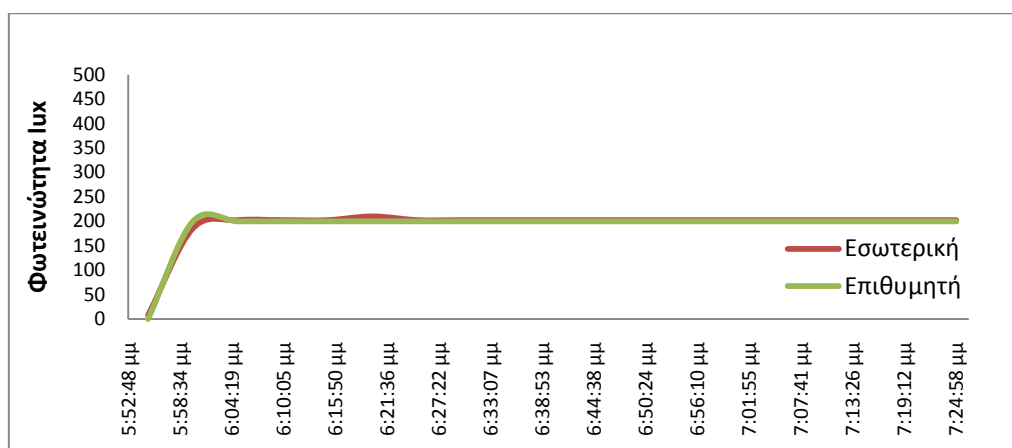
Σχήμα 4-6 Ελεγκτής οπτικής άνεσης (V02.01-02)

Στην διαδικασία ελέγχου φαίνεται η προτεραιότητα που έχει ο φυσικός φωτισμός έναντι του τεχνητού. Η παραπάνω εκδοχή του ελεγκτή είναι αρκετά αυστηρή αφού η ανοχή σφάλματος κυμαίνεται στο εύρος από -5 έως 5 lux δηλαδή στα 10 lux. Όλοι

οι παράμετροι που χρησιμοποιεί ο ελεγκτής, όπως για παράδειγμα τα σφάλματα, οι χρόνοι προστασίας εγγραφής και τα βήματα των ενεργοποιητών ορίζονται εξ' αρχής στις πρώτες γραμμές αρχικοποίησης των μεταβλητών. Με αυτό τον τρόπο το σύστημα δίνει την δυνατότητα στους χρήστες να αλλάζουν τις τιμές αυτές και να εξετάζουν άμεσα τα αποτελέσματα της διαδικασίας ελέγχου. Στα πλαίσια βελτίωσης του ελεγκτή οπτικής άνεσης αναπτύχθηκε μία βελτιωμένη έκδοση (V03.03-02) η οποία λαμβάνει υπόψη την απευθείας είσοδο των ακτίνων του ηλίου τις απογευματινές ώρες από τα δυτικά παράθυρα (βλ. Παράρτημα Α.3). Για τον σκοπό αυτό προστέθηκε στην εγκατάσταση ο δεύτερος αισθητήρας εσωτερικής φωτεινότητας που αναφέρθηκε στην 4.3.1.3. Ο ελεγκτής δίνει προτεραιότητα στα δυτικά σκίαστρα έναντι του τεχνικού φωτισμού μόνο στην περίπτωση που δεν καταγράφει μεγάλη τιμή στον δεύτερο αισθητήρα εσωτερικής φωτεινότητας.

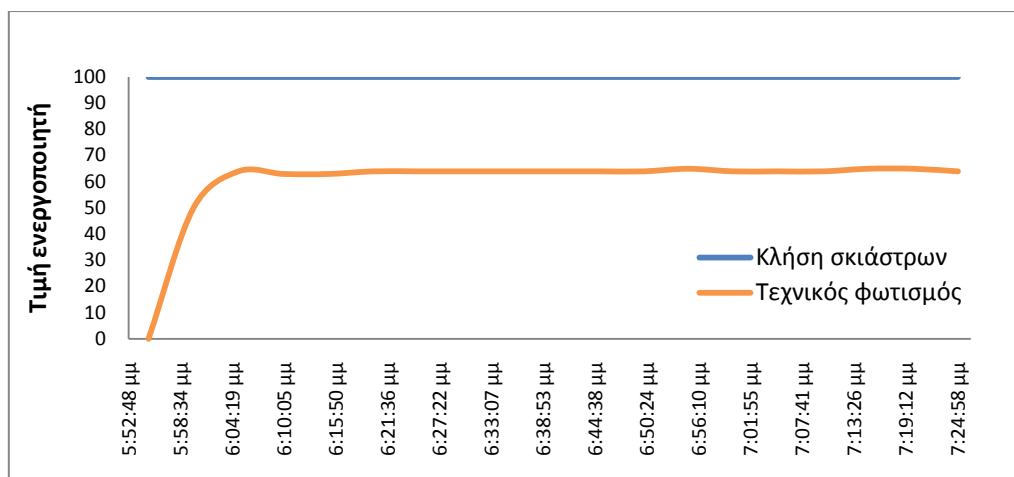
4.4.1.3 Πειραματικές μετρήσεις

Κατά τις απογευματινές ή βραδινές ώρες που η εξωτερική φωτεινότητα είναι μηδενική η εσωτερική φωτεινότητα δεν μπορεί να ξεπεράσει τα 250 lux ακόμα και όταν το χειριστήριο ελέγχου του τεχνικού φωτισμού βρίσκεται στην θέση 100%. Έτσι ορίζοντας ένα set point μικρότερο των 250 lux ο ελεγκτής μπορεί να το προσεγγίσει γρήγορα και με ακρίβεια ενώ σε αντίθετη περίπτωση το σύστημα φωτισμού λειτουργεί στο 100% των δυνατοτήτων του χωρίς να μπορεί να φτάσει την επιθυμητή τιμή. Η επιθυμητή τιμή εσωτερικού φωτισμού οριστική στα 200 lux και όπως φαίνεται στο παρακάτω διάγραμμα ο ελεγκτής την προσεγγίζει με μεγάλη ακρίβεια, πρακτικά το επίπεδο εσωτερικής φωτεινότητας ταυτίζεται με την επιθυμητή τιμή.



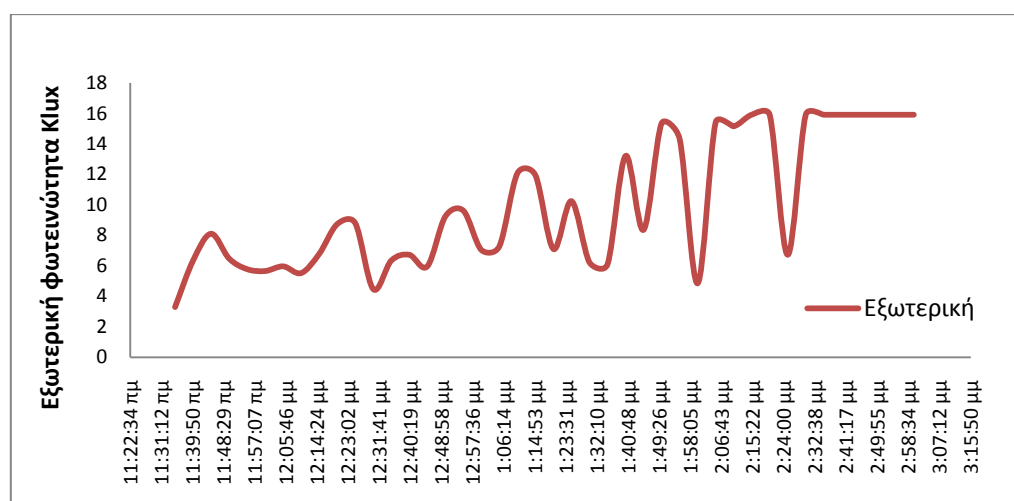
Εικόνα 4-20 Εσωτερική φωτεινότητα

Στην συνέχεια παρουσιάζεται το διάγραμμα των τιμών των ενεργοποιητών στο οποίο διακρίνεται η τιμή του ενεργοποιητή τεχνικού φωτισμού ίση περίπου με 65% ενώ τα σκίαστρα καθ' όλη την διάρκεια της δοκιμής παρέμεναν κλειστά αφού είχαμε μηδενική τιμή εξωτερικής φωτεινότητας.



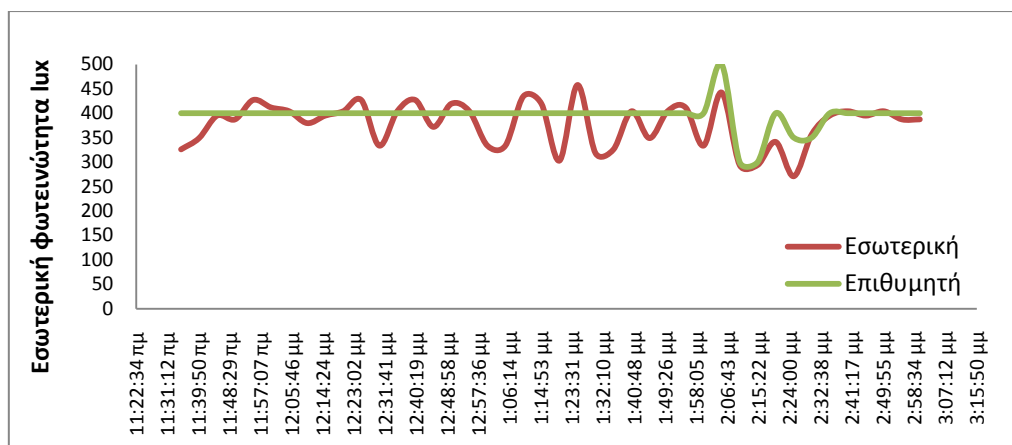
Εικόνα 4-21 Τιμές ενεργοποιητών

Κατά τις πρωινές και μεσημεριανές ώρες που οι τιμές της εξωτερικής φωτεινότητας είναι μεγάλες, μπορούμε να ορίσουμε την επιθυμητή τιμή πάνω από τα 400 lux και να εξετάσουμε την λειτουργία του ελεγκτή οπτικής άνεσης. Η μέρα που επιλέχθηκε παρουσιάζει ιδιαίτερο ενδιαφέρον επειδή είχε αρκετές «βίαιες» αυξομειώσεις στις τιμές της εξωτερικής φωτεινότητας.



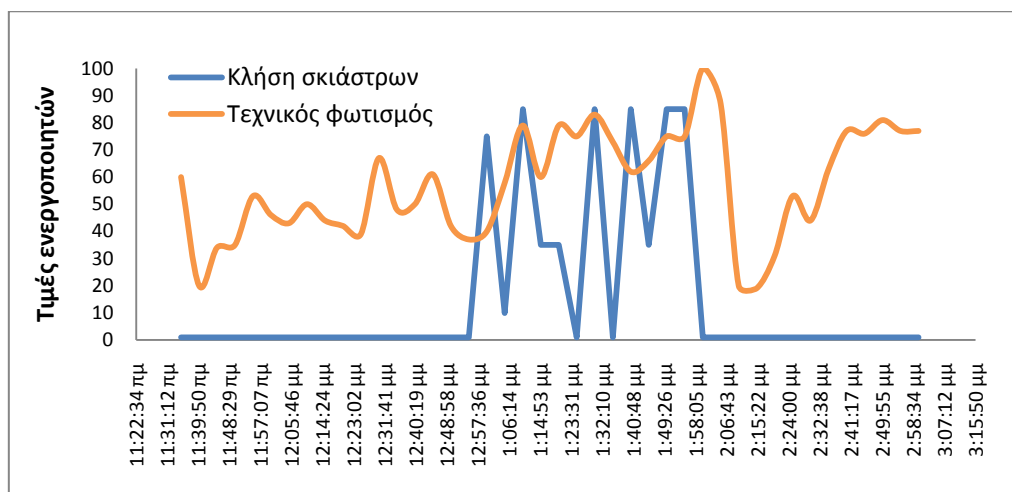
Σχήμα 4-7 Εξωτερική φωτεινότητα

Η βελτιωμένη έκδοση του ελεγκτή οπτικής άνεσης (βλ. Παράρτημα Α.3) καταφέρνει με επιτυχία να ελέγξει το επίπεδο εσωτερικής φωτεινότητας και να το διατηρήσει κοντά στην επιθυμητή τιμή. Ακόμα και στην περίπτωση που προχωρούσαμε σε αλλαγή της επιθυμητής τιμής εσωτερικής φωτεινότητας ο ελεγκτής ακολούθησε την νέα τιμή με ικανοποιητική ακρίβεια. Η συμπεριφορά του ελεγκτή παρουσιάζεται καθ' όλη την διάρκεια λειτουργίας του στο σχήμα 4-8. Τα χρονικά διαστήματα μεταξύ των μετρήσεων είναι της τάξεως των 5 λεπτών. Ενδέχεται να έχει χαθεί πληροφορία μεταξύ των μετρήσεων η οποία να επηρέαζε την απεικόνιση των διαγραμμάτων λόγω του ότι η αυξομειώσεις της εξωτερικής φωτεινότητας ήταν πολύ συχνές και απότομες.



Σχήμα 4-8 Εσωτερική φωτεινότητα

Όπως φαίνεται και από το σχήμα 4-9 ο ελεγκτής οπτικής άνεσης κλείνει το δυτικό σκίαστρο στις περιπτώσεις που καταγράφονται μεγάλες τιμές από τον αισθητήρα εσωτερικής φωτεινότητας της δυτικής πλευράς.



Σχήμα 4-9 Τιμές ενεργοποιητών

Το ψηφιακό dimmer της συσκευής EIB DALI Interface επιτρέπει την αυξομείωση του επιπέδου εσωτερικού φωτισμού χωρίς κάποια σημαντική φθορά στα φωτιστικά σώματα ή στις 48 συσκευές EIB El. Ballast. Έτσι οριστικό ο χρόνος προστασίας εγγραφής του σήματος του ενεργοποιητή EIB DALI Interface στο 1 ή 5 sec ανάλογα το σφάλμα.

Στον ενεργοποιητή των σκιάστρων ο αντίστοιχος χρόνος προστασίας εγγραφής ορίστηκε στα 5 ή 30 sec ανάλογα με την τιμή του απόλυτου σφάλματος. Αν κατά την λειτουργία του ελεγκτή διαπιστωθεί ότι η συχνή κίνηση των σκιάστρων ενοχλεί π.χ. λόγω του θορύβου των κινητήρων μπορούμε να μεγαλώσουμε τους αντίστοιχους χρόνους προστασίας εγγραφής του αντίστοιχου ενεργοποιητή.

4.4.2 Έλεγχος θερμικής άνεσης

4.4.2.1 Γενικά

Για τον έλεγχο της θερμοκρασίας στο εργαστήριο υλοποιήθηκε ένας ελεγκτής που τον διαχειρίζεται ο μηχανισμός OPC Real Time Control. Χρησιμοποιεί την βασική διασύνδεση χρήστη (interface) και τα χειριστήρια ελέγχου της εφαρμογής OPC to Web για την εκκίνηση, τον τερματισμό και τον καθορισμό των παραμέτρων που χρειάζεται. Το σύστημα ελέγχου θερμικής άνεσης που υλοποιήθηκε έχει τρεις εισόδους οι οποίες είναι οι εξής:

- Η τιμή της εξωτερικής θερμοκρασίας από τον μετεωρολογικό σταθμό της εγκατάστασης.
- Η τιμή της εσωτερικής θερμοκρασίας από το σήμα του αντίστοιχου αισθητήρα.
- Η τιμή της επιθυμητής εσωτερικής θερμοκρασίας που ορίζεται αυτόματα από την εφαρμογή OPC to Web.

Οι έξοδοι του ελεγκτή θερμικής άνεσης είναι οι εξής:

- Κατάσταση λειτουργίας του κλιματιστικού.
- Άνοιγμα / κλείσιμο παραθύρων.
- Άνοιγμα / κλείσιμο σκιάστρων.

Με τον ελεγκτή θερμικής άνεσης γίνεται προσπάθεια να διατηρηθεί η θερμοκρασία στα επιθυμητά επίπεδα χρησιμοποιώντας μεθόδους ενεργητικής ή παθητικής θέρμανσης και ψύξης.

Για να κάνει ο ελεγκτής θερμικής άνεσης εξοικονόμηση ενέργειας θα πρέπει να ρυθμίζεται κατάλληλα η επιθυμητή εσωτερική θερμοκρασία. Πολλοί δεν γνωρίζουν ποια είναι η κατάλληλη θερμοκρασία ανά εποχή του χρόνου, έτσι πολλές φορές βλέπουμε συστήματα κλιματισμού να είναι ρυθμισμένα το καλοκαίρι για ψύξη στους 20°C και το χειμώνα για θέρμανση στους 27°C. Αυτό όχι μόνο οδηγεί σε σπατάλη ενέργειας, αλλά είναι και επικίνδυνο για την υγεία.

Επίσης, ο ρουχισμός των ανθρώπων είναι διαφορετικός από εποχή σε εποχή. Έτσι για παράδειγμα, κάποια θερμοκρασία που τον χειμώνα θεωρείται ιδανική 21 - 23°C, το καλοκαίρι θεωρείται χαμηλή. Για αυτό, ο ελεγκτής σχεδιάστηκε έτσι ώστε να αποφασίζει αυτόματα ποια θα είναι η επιθυμητή θερμοκρασία λαμβάνοντας υπ' όψιν τα αποτελέσματα της σύγκρισης μεταξύ των τιμών εσωτερικής και εξωτερικής θερμοκρασίας. Η εφαρμογή OPC to Web στην καρτέλα Πίνακας Ελέγχου διαθέτει ένα χειριστήριο για την ρύθμιση της νεκρής ζώνης η οποία ορίζει το εύρος τιμών της εσωτερικής θερμοκρασίας στο οποίο δεν λειτουργεί η ενεργητική θέρμανση και ψύξη. Από προεπιλογή το σύστημα θεωρεί ότι θερμοκρασίες μεταξύ 21°C και 26°C

είναι ανεκτές και δεν απαιτείται η χρήση ενεργητικής θέρμανσης και ψύξης. Όμως οι χρήστες του συστήματος έχουν την δυνατότητα να μετατοπίσουν το εύρος αυτό κατά ένα βαθμό κελσίου προς τα πάνω ή προς τα κάτω. Συγκεκριμένα αν πατήσουν την επιλογή [μεταξύ 20°C και 25°C] το εύρος παύσης λειτουργίας της ενεργητικής θέρμανσης και ψύξης μετατοπίζεται προς τα κάτω και ο ελεγκτής θεωρεί τις θερμοκρασίες μεταξύ των 20°C και 25°C ανεκτές. Διαφορετικά αν οι χρήστες του συστήματος πατήσουν την επιλογή [μεταξύ 22°C και 27°C] το εύρος παύσης λειτουργίας της ενεργητικής θέρμανσης και ψύξης γίνεται μεταξύ των 22°C και 27°C.

4.4.2.2 Ενεργητικός κλιματισμός

Αν η εσωτερική θερμοκρασία είναι χαμηλότερη από το κατώτερο όριο του εύρους παύσης τότε ο ελεγκτής καταλαβαίνει ότι η τρέχουσα εποχή είναι χειμώνας, ορίζει την τιμή της επιθυμητής θερμοκρασίας ίση με το κατώτερο όριο και το σύστημα μπαίνει σε κατάσταση ενεργητικής θέρμανσης μέχρι να το φτάσει.

Αν η εσωτερική θερμοκρασία είναι μεγαλύτερη από το ανώτερο όριο του εύρος παύσης τότε ο ελεγκτής καταλαβαίνει ότι η τρέχουσα εποχή είναι καλοκαίρι, ορίζει την τιμή της επιθυμητής θερμοκρασίας ίση με το ανώτερο όριο και το σύστημα μπαίνει σε κατάσταση ενεργητικής ψύξης μέχρι να το φτάσει.

Όταν η τιμή της εσωτερική θερμοκρασίας εξισωθεί με την επιθυμητή το σύστημα ενεργητικού κλιματισμού συνεχίζει να λειτουργεί μέχρι η απόλυτη διαφορά τους να γίνει ίση με 0.5°C. Επίσης ο ελεγκτής θερμικής άνεσης χρησιμοποιεί την συνάρτηση sWrite για να γράψει τις εντολές στα σήματα ελέγχου του συστήματος κλιματισμού. Οι μεταβλητές delayAC και delayCL ορίζουν τα χρονικά διαστήματα προστασίας εγγραφής του ενεργοποιητή του συστήματος κλιματισμού τα οποία πρέπει να είναι της τάξεως των μερικών λεπτών.

Επομένως για να αποφευχθούν οι συχνές εκκινήσεις του συστήματος κλιματισμού όταν η εσωτερική θερμοκρασία είναι γύρω από το σημείο ισορροπίας της διαδικασίας ελέγχου χρησιμοποιούνται δύο ανεξάρτητες τεχνικές όπου η πρώτη μοιάζει με την αρχή λειτουργίας του θερμοστάτη ενώ η δεύτερη χρησιμοποιεί την λειτουργία της προστασίας εγγραφής του μηχανισμού OPC Real Time Control.

Για να μπορέσει ο ελεγκτής να γράψει στα σήματα του συστήματος κλιματισμού θα πρέπει να περάσει με επιτυχία και τις δύο τεχνικές ελέγχου. Με αυτό τον τρόπο εξασφαλίζεται η καλή λειτουργία του συστήματος κλιματισμού ακόμα και στην περίπτωση που ο αισθητήρας εσωτερικής θερμοκρασίας αρχίσει να δίνει λανθασμένες τιμές.

4.4.2.3 Παθητικός κλιματισμός

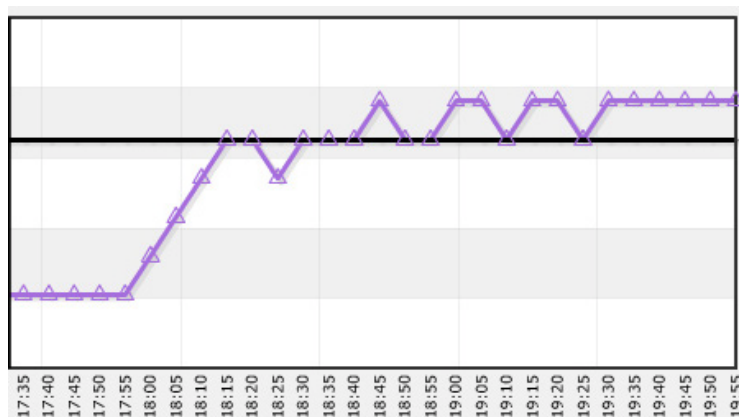
Ο παθητικός κλιματισμός είναι η διαδικασία κατά την οποία αλλάζει η θερμοκρασία του χώρου χωρίς κατανάλωση ενέργειας όπως για παράδειγμα με το άνοιγμα των παραθύρων και των σκιάστρων. Σε αντίθεση με τον ενεργητικό κλιματισμό όπου μπορεί να εφαρμόζεται πάντα εφόσον υπάρχει σύστημα κλιματισμού, ο παθητικός κλιματισμός δεν μπορεί να εφαρμόζεται το ίδιο συχνά αφού εξαρτάται από την εξωτερική θερμοκρασία, από την ύπαρξη ηλιακού φωτός, από την ταχύτητα του ανέμου και από την ύπαρξη βροχής (20).

Έχει όμως το πλεονέκτημα ότι δεν καταναλώνει καθόλου ενέργεια αφού η ενέργεια για την κίνηση των κινητήρων που ανοίγουν τα παράθυρα και τα σκιάστρα θεωρείται αμελητέα. Οι βασικές αρχές που πρέπει να ακολουθεί ο ελεγκτής για την παθητική θέρμανση και ψύξη της αίθουσας είναι οι εξής:

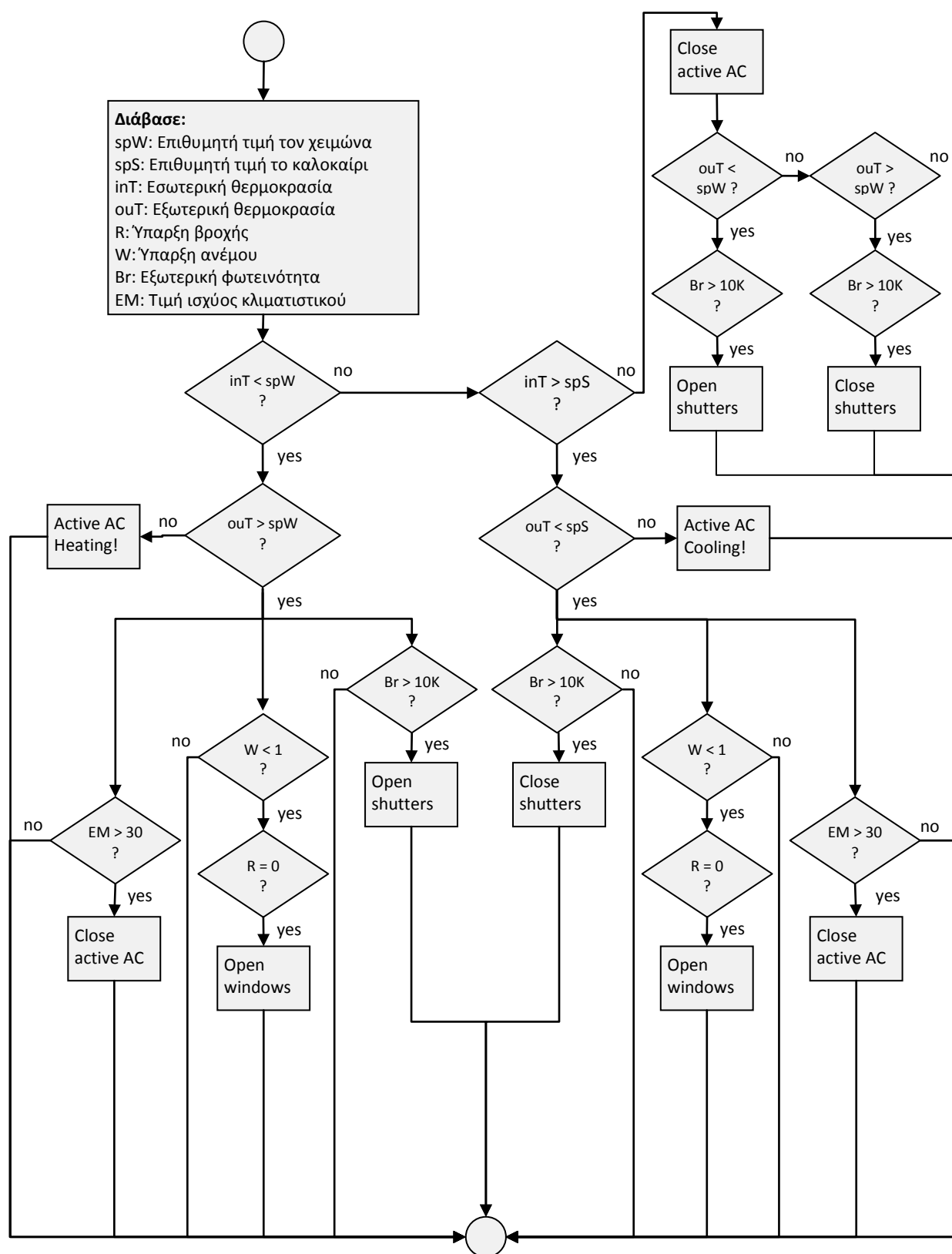
- Κατά τους καλοκαιρινούς μήνες και στην περίπτωση έντονης ηλιοφάνειας προτιμάται τα σκιάστρα να είναι κλειστά, ενώ τους χειμερινούς προτιμάται να είναι ανοιχτά.
- Τα παράθυρα ανοίγουν μόνο στην περίπτωση που η εξωτερική θερμοκρασία μπορεί να βοηθήσει το σύστημα να φτάσει την επιθυμητή εσωτερική θερμοκρασία, και μόνο όταν δεν υπάρχουν έντονα καιρικά φαινόμενα, όπως η ύπαρξη βροχής και μεγάλες ταχύτητες ανέμου.

4.4.2.4 Έλεγχος λειτουργίας

Ο ελεγκτής θερμικής άνεσης ενεργοποιήθηκε από την εφαρμογή OPC to Web στις 17:55 μίας συγκεκριμένης μέρας όπως φαίνεται και από το παρακάτω διάγραμμα. Η εσωτερική θερμοκρασία ήταν κάτω από το επιθυμητό επίπεδο των 21°C και έτσι το σύστημα μπήκε απλά σε κατάσταση ενεργητικής θέρμανσης μέχρι να το φτάσει. Η εφαρμογή OPC to Web δίνει την δυνατότητα παραγωγής δυναμικών διαγραμμάτων από τα οποία μπορεί να αξιολογηθεί η σωστή λειτουργία του ελεγκτή θερμικής άνεσης από τα ιστορικά δεδομένα (βλ. Παράρτημα Α.1, Εικόνα Α.1-5).



Εικόνα 4-22 Σταθεροποίηση της εσωτερικής θερμοκρασίας μεταξύ 21°C και 21.5°C



Σχήμα 4-10 Ελεγκτής θερμικής άνεσης (V4.03-02)

5 Συμπεράσματα & Μελλοντικές Κατευθύνσεις

5.1 Συμπεράσματα

Κατά την υλοποίηση των συστημάτων και της διαδικτυακής εφαρμογής OPC to Web προέκυψαν προβλήματα που χρειάστηκαν να αντιμετωπιστούν. Η εύρεση της .NET έκδοσης V2.2.5.20 του OPCDAuto.dll ήταν πραγματική δυσκολία αφού διατίθεται μόνο στα περίπου 400 μέλη του οργανισμού OPC. Η αρχικοποίηση του λογισμικού στον κεντρικό Η/Υ (βλ. Ενότητα 2.4.3.2) και η επιτυχής δημιουργία του αντικειμένου OPC.Automation.1 καθώς και η σύνδεση του με τον εξυπηρετητή OPC έγινε μετά από πολλές ώρες πειραματισμών και μελέτης λόγω του ότι δεν υπάρχει πουθενά διαθέσιμη πληροφορία για τέτοιου είδους υλοποιήσεις.

Το σύστημα ικανοποιεί σε πολύ μεγάλο βαθμό τις αρχικές σχεδιαστικές επιλογές. Μετά από μεγάλο διάστημα συνεχούς λειτουργίας παρατηρήθηκε εξαιρετική σταθερότητα τόσο του υλικού, όσο και του λογισμικού. Το λειτουργικό σύστημα Windows Server 2003 σε συνδυασμό με τον MS SQL Server 2000 που φιλοξενεί την βάση δεδομένων της εφαρμογής OPC to Web αποτελούν το καλύτερο πακέτο για συστήματα υψηλής αξιοπιστίας. Η εφαρμογή δίνει την δυνατότητα προσθήκης νέων στοιχείων στην διαδικασία ελέγχου χωρίς να χρειάζονται εξειδικευμένες γνώσεις προγραμματισμού.

Επίσης η ανάπτυξη του μηχανισμού που ελέγχει και διαχειρίζεται τα στοιχεία του λειτουργικού συστήματος και των προγραμμάτων που είναι εγκατεστημένα σε αυτό, από δυναμικές ιστοσελίδες ήταν μία πραγματική πρόκληση.

Ενσωματώθηκε με επιτυχία η τεχνολογία AJAX ώστε να εκτυπώνονται άμεσα οι τρέχουσες τιμές των σημάτων της διαδικασίας ελέγχου, να ενημερώνονται οι θέσεις των χειριστηρίων των ενεργοποιητών αλλά και για να ελέγχεται η κατάσταση λειτουργίας για κάθε υπηρεσία που σχετίζεται με την διαδικασία ελέγχου.

Ο μηχανισμός OPC Real Time Control αποτελεί ένα πραγματικό εργαλείο αφού δίνει την δυνατότητα στους χρήστες να αναπτύξουν σενάρια, ελεγκτές, ειδοποιήσεις κ.τ.λ. με απλές βασικές γνώσεις προγραμματισμού. Τα αποτελέσματα των ελεγκτών οπτικής και θερμικής άνεσης είναι πολύ ικανοποιητικά καθώς εκτελούνται με επιτυχία και επιλέγουν πραγματικά πολύ καλές λύσεις με γνώμονα πάντα την προσέγγιση των επιθυμητών τιμών και την εξοικονόμηση ενέργειας. Το σύστημα χρησιμοποιεί απλούς αλγόριθμους ελέγχου οι οποίοι εκτελούνται από τον μηχανισμό OPC Real Time Control του συστήματος.

5.2 Προτάσεις για μελλοντική ανάπτυξη

5.2.1 Βελτιώσεις λογισμικού

Από τις βασικότερες βελτιώσεις που μπορούν να γίνουν στην εφαρμογή είναι η υποστήριξη της δυνατότητας εξερεύνησης των σημάτων του εξυπηρετητή OPC. Δηλαδή στις φόρμες προσθήκης και επεξεργασίας νέου στοιχείου (βλ. Παράρτημα Α.1, Εικόνες Α.1-3 και Α.1-4) η εφαρμογή να μπορεί να εξερευνήσει τα OPC Groups του εξυπηρετητή OPC με τον οποίο έχει συνδεθεί.

Η εφαρμογή στην καρτέλα του Πίνακα Ελέγχου θα μπορούσε επίσης να εμφανίζει πληροφορίες σχετικές με την κατάσταση χρήσης του Η/Υ, όπως πληροφορίες για την χρήση της μνήμης, του επεξεργαστή κ.τ.λ.

Βελτίωση στον τρόπο διαχείρισης και εκτέλεσης των σεναρίων του μηχανισμού OPC Real Time Control του συστήματος έτσι ώστε ελεγκτές να μπορούν να εκτελούνται παράλληλα κάτω από διαφορετικές διεργασίες.

Βελτίωση στον μηχανισμό τεχνολογίας AJAX, που ανανεώνει τα χειριστήρια ελέγχου των ενεργοποιητών. Στην παρούσα έκδοση ο μηχανισμός ψάχνει για τις νέες θέσεις των χειριστηρίων ελέγχου κάθε φορά που θα περάσει ένα συγκεκριμένο χρονικό διάστημα.

Δυνατότητα ανεβάσματος (upload) αρχείων M-File (ελεγκτών οπτικής άνεσης, θερμικής άνεσης και άλλων) στον κεντρικό Η/Υ και η απομακρυσμένη εκτέλεση τους από το διαδίκτυο. Οι χρήστες θα μπορούσαν να πάρουν έτοιμα τα διαγράμματα με τις πειραματικές μετρήσεις από την εφαρμογή OPC to Web ή ακόμα και να κατεβάσουν τα αποτελέσματα σε αρχεία τύπου Excel χωρίς να χρειάζεται να βρίσκονται με φυσικό τρόπο στον χώρο της διαδικασίας ελέγχου.

Δυνατότητα καταγραφής όλων των συμβάντων εγγραφής τιμών στα σήματα των ενεργοποιητών της διαδικασίας ελέγχου.

Οι ελεγκτές οπτικής και θερμικής άνεσης αρχικά σχεδιάστηκαν για την βελτίωση και την αποσφαλμάτωση της λειτουργίας του μηχανισμού OPC Real Time Control. Στα πλαίσια της εργασίας εξελίχτηκαν αρκετά, τόσο ώστε να δίνουν πολύ ικανοποιητικά αποτελέσματα κοντά στις επιθυμητές τιμές, με όσο το δυνατό λιγότερη κατανάλωση ενέργειας. Θα μπορούσαν να δοκιμαστούν και άλλοι αλγόριθμοι και να γίνουν συγκρίσεις για την απόδοσή τους.

Ο μηχανισμός OPC Real Time Control δίνει την δυνατότητα στους χρήστες να δημιουργούν με ευκολία σεσνάκια και ελεγκτές έτσι ώστε να εκμεταλλεύονται πλήρως τις δυνατότητες του συστήματος. Θα μπορούσαν να αναπτυχθούν τέτοια σεσνάκια ανάλογα με τις ιδέες και την φαντασία του κάθε χρήστη.

5.2.2 Βελτιώσεις εγκατάστασης

Η ελεύθερη τοπολογία δικτύωσης του συστήματος αφήνει περιθώρια για την προσθήκη περισσότερων αισθητήρων και ενεργοποιητών χωρίς σημαντικές αλλαγές.

Στην περίπτωση του φωτισμού προτείνεται η εγκατάσταση περισσότερων ελεγχόμενων σκιάστρων ώστε να υπάρχει η δυνατότητα για περισσότερο ηλιακό φως, το οποίο είναι καλύτερο ποιοτικά σε σχέση με το τεχνητό και δεν κοστίζει. Επίσης προτείνεται η εγκατάσταση περισσότερων λαμπτήρων και η αντικατάσταση των καμμένων ώστε να μπορούμε να πετύχουμε πιο υψηλά επίπεδα εσωτερικής φωτεινότητας.

Στην περίπτωση της ενεργητικής θέρμανσης και ψύξης, προτείνεται η αντικατάσταση του συστήματος κλιματισμού με άλλο, που να παρέχει την δυνατότητα κλιμακωτής λειτουργίας (με inverter). Ακόμη θα μπορούσε να αντικατασταθεί ο αισθητήρας εσωτερικής θερμοκρασίας με άλλον μεγαλύτερης ανάλυσης σε σχέση με την ανάλυση του υπάρχοντα αισθητήρα, που είναι 0.5°C.

Βιβλιογραφία

1. Οικονομάκος Χ. *Κατανεμημένα Συστήματα Ελέγχου*. ΤΕΙ Χαλκίδας, 2009. Σημειώσεις μαθήματος.
2. Κορρές Γ. *Στοιχεία Συστημάτων SCADA*. Αθήνα : ΕΜΠ, 2001. Σημειώσεις ΕΜΠ
3. ΠΑΠΑΚΩΝΣΤΑΝΤΙΝΟΥ Γ. και ΤΣΑΝΑΚΑΣ Π. *Εισαγωγή στον πολυπρογραμματισμό σε περιβάλλον Win32*. Αθήνα : ΕΡΓΑΣΤΗΡΙΟ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ, 2004. Σημειώσεις ΕΜΠ.
4. *Dynamic Data Exchange*. [Ηλεκτρονικό]
http://en.wikipedia.org/wiki/Dynamic_Data_Exchange.
5. IUnknown Interface. [Ηλεκτρονικό] Microsoft Corporation. MSDN Library.
[http://msdn.microsoft.com/en-us/library/ms680509\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/ms680509(VS.85).aspx).
6. HRESULT. [Ηλεκτρονικό] <http://en.wikipedia.org/wiki/HRESULT>.
7. Stavic Ondrej. *OPC Server*. Praha, 2006.
8. The OPC Foundation. [Ηλεκτρονικό] OPC Foundation.
<http://www.opcfoundation.org/>.
9. *Using OPC via DCOM with Microsoft Windows XP Service Pack 2*. OPC Foundation, 2006.
10. Errors Connecting to OPC Server When Using DataSocket as an OPC Client. [Ηλεκτρονικό] National Instruments Corporation, 2008. <http://digital.ni.com>.
11. EIBA Handbook Series. *Introduction to the System*. SIEMENS, 1999.
12. Deitel, P. J. και Deitel, H. M. *Rich Internet Applications, and Web Development*. Boston : Pearson Education, 2008.
13. *Windows Managment Instrumentation* . [Ηλεκτρονικό]
[http://msdn.microsoft.com/en-us/library/aa394582\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/aa394582(VS.85).aspx).
14. Configuring IIS 5.0 and Later for WMI ASP Scripting. [Ηλεκτρονικό] Microsoft Corporation, 2009. MSDN Library.
[http://msdn.microsoft.com/en-us/library/aa389281\(VS.85,printer\).aspx](http://msdn.microsoft.com/en-us/library/aa389281(VS.85,printer).aspx).
15. Mark Russinovich. *PsExec*. [Ηλεκτρονικό] Windows Sysinternals.
<http://technet.microsoft.com/en-us/sysinternals/bb897553.aspx>.

16. Windows Script Host. [Ηλεκτρονικό] Wikipedia.
http://en.wikipedia.org/wiki/Windows_Script_Host.
17. Λάζος Χ. *Συστήμα διαχείρισης ενέργειας κτιρίου με χρήση του πρωτοκόλλου KNX/EIB και Αλγόριθμο Προβλεπτικού Ελέγχου*. Χανιά : Πολυτεχνείο Κρήτης, 2008.
18. How to turn on automatic logon in Windows. [Ηλεκτρονικό] 2007. Microsoft Support. <http://support.microsoft.com/kb/324737>.
19. Τούλογλου Σ. *KNX/EIB Τεχνική Ηλεκτρικών Εγκαταστάσεων*. Αθήνα : Εκδόσεις Ίων, 2006.
20. Χατζηλάμπρος Π. *Σύστημα ελέγχου οπτικής - θερμικής άνεσης και ποιότητας αέρα ζώνης κτιρίου με την χρήση PLC και δικτύου KNX/EIB*. Χανιά : Πολυτεχνείο Κρήτης, 2007.
21. Mihai Moldovan. Matlab Central Settings. [Ηλεκτρονικό] Mathworks, 2005.
<http://www.mathworks.com/matlabcentral/fileexchange/6914-settings>.
22. Command-line Tools Included in IIS. [Ηλεκτρονικό] Microsoft Corporation, 2009. MSDN Library. [http://msdn.microsoft.com/en-us/library/ms525006\(printer\).aspx](http://msdn.microsoft.com/en-us/library/ms525006(printer).aspx).
23. Creating Active Server Pages for WMI. [Ηλεκτρονικό] Microsoft Corporation, 8 1 2009. MSDN Library.
[http://msdn.microsoft.com/en-us/library/aa389395\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/aa389395(VS.85).aspx).
24. Papadopoulos, George A. *Αρχές Σπονδυλωτού Προγραμματισμού σε Κατανεμημένα Συστήματα*. ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ.
25. *DESIGO™ INSIGHT Open OPC Client*. SIEMENS, 2006.

Παράρτημα Α

Α.1 Παρουσίαση εφαρμογής OPC to Web

Η διαδικτυακή εφαρμογή OPC to Web είναι διαθέσιμη στην ηλεκτρονική διεύθυνση <http://ermis.smartlab.tuc.gr> την οποία οι χρήστες πρέπει να επισκέπτονται μόνο από τις ενημερωμένες εκδόσεις των προγραμμάτων περιήγησης ιστού.



Επειδή η εφαρμογή OPC to Web έχει σχεδιαστεί έτσι ώστε να χρησιμοποιείται σε καθημερινή βάση από το προσωπικό του εργαστηρίου Ελέγχου Βιομηχανικών Συστημάτων αναπτύχθηκε στην πλατφόρμα Microsoft Visual Basic 2008 μία απλή εκτελέσιμη εφαρμογή με την ίδια ονομασία που ενσωματώνει ένα μόνο αντικείμενο τύπου browser (ουσιαστικά πρόκειται για τον Internet Explorer 8) με προεπιλεγμένη διεύθυνση και λογαριασμό χρήστη. Η εγκατάσταση μπορεί να γίνει σε οποιοδήποτε Η/Υ σε λειτουργικό σύστημα της Microsoft με τις παρακάτω ελάχιστες απαιτήσεις:

- Windows XP με Service Pack 2.
- Windows Installer 3.1 και το Framework .NET 3.5 τα οποία είναι διαθέσιμα στο Download Center της Microsoft.

Η εγκατάσταση της εφαρμογής OPC to Web εκτελεί τις παρακάτω διαδικασίες:

- Δημιουργεί και αντιγράφει τα αρχεία της εγκατάστασης στον φάκελο C:\TUC\Smartlab.
- Δημιουργεί εικονίδιο στην επιφάνεια εργασίας.
- Δημιουργεί στο All Programs του Start menu τα αντίστοιχα Shortcuts.

Η εφαρμογή ελέγχει για ενημερώσεις σε κάθε έναρξη. Με αυτό τον τρόπο οι διαχειριστές του μπορούν να αλλάξουν την ηλεκτρονική διεύθυνση του συστήματος χωρίς να χρειαστεί να ενημερώσουν τους χρήστες για την αλλαγή. Αυτό μπορεί να γίνει για απλά τεχνικούς λόγους ή σε περίπτωση πιθανής αναβάθμισης.



Εικόνα Α.1-1 Αρχική σελίδα της εφαρμογής OPC to Web

The screenshot shows the 'OPC to Web' control panel. The interface is divided into several sections:

- Header:** 'Υπηρεσίες' (Services) with a logo.
- Right Sidebar:**
 - Καλώς ήλθατε (Welcome):** Displays user information for 'smartlab', including 'Last login 4/2/2010', 'Remote host 147.27.10.152', 'Level Διαχειριστής', and 'Ρυθμίσεις λογαριασμού' (Account settings).
 - Επιλογές (Options):** A tree view showing a hierarchy: Εξυπηρετητής (Server) > Αισθητήρες (Sensors) > Υπηρεσίες (Services) > Σενάρια (Scenarios) > Ενεργοποιητές (Activators) > Χρήστες (Users) > Ρυθμίσεις (Settings).
 - Υπηρεσίες (Services):** A list of services with status icons:

Service	Status
OPC	✓ (Active)
PR.CONTROL	✗ (Inactive)
MAIN SWITCH	🏠 (Home/Default)
- Main Control Area (Πίνακας ελέγχου):**
 - OPC Section:** Contains a 'START/STOP' button for the OPC service and a 'START/STOP' button for the basic controller.
 - PR.CONTROL Section:** Includes checkboxes for 'Κλιματισμός' (Air conditioning), 'Εσωτερικός φωτισμός' (Indoor lighting), and 'Ποιότητα αέρα' (Air quality). Below these is a '600 Επανάληψη' (Repeat) button.
 - TERMINATE Section:** Contains a 'STOP' button for the OPC Real Time Control mechanism.
 - RT.CONTROL Section:** Contains a 'START/STOP' button for the OPC Real Time Control mechanism.
 - Central Display:** Shows real-time data and settings:
 - Ρύθμιση νεκρής ζώνης (Dead band setting):** A dropdown menu set to 'μεταξύ 21°C και 26°C'.
 - Εσωτερικός φωτισμός (Indoor lighting):** A slider set to 300 Lux, with a range from 0 to 600 Lux.
 - Ποιότητα αέρα (Air quality):** A slider set to 650 ppm, with a range from 500 to 800 ppm.
 - Υγρασία (Humidity):** A slider set to 50%, with a range from 40% to 60%.

Callouts provide additional context:

- 'Εκκίνηση / τερματισμός του εξυπηρετητή OPC' points to the OPC START/STOP button.
- 'Εκκίνηση / τερματισμός του βασικού ελεγκτή' points to the basic controller START/STOP button.
- 'Χειριστήρια ρύθμισης των επιθυμητών τιμών' points to the central display sliders.
- 'Βασικές πληροφορίες του ενεργού λογαριασμού' points to the user information in the sidebar.
- 'Καθυστέρηση επανάληψης σε seconds' points to the '600 Επανάληψη' button.
- 'Τμηματική ενεργοποίηση βασικού ελεγκτή' points to the 'Κλιματισμός', 'Εσωτερικός φωτισμός', and 'Ποιότητα αέρα' checkboxes.
- 'Τερματισμός σεναρίων και ελεγκτών του μηχανισμού OPC Real Time Control που δεν ανήκουν στην ομάδα συστήματος καθώς και αποστολή τιμών αρχικής κατάστασης στους ενεργοποιητές. Η συγκεκριμένη επιλογή δεν τερματίζει τον βασικό ελεγκτή.' points to the TERMINATE button.
- 'Εκκίνηση / τερματισμός του μηχανισμού OPC Real Time Control' points to the RT.CONTROL START/STOP button.
- 'Βασικό μενού επιλογών' points to the 'Επιλογές' sidebar section.

Εικόνα Α.1-2 Σελίδα του πίνακα ελέγχου, πρόσβαση από το μενού Υπηρεσίες → Πίνακας Ελέγχου

OPC to Web

Πίνακας στοιχείων

Σελίδα 1 από 1

Πίνακας στοιχείων της επιλεγμένης ομάδας "telemetry_light"

Τρέχουσες τιμές στοιχείων

Timestamp τελευταίας αλλαγής των σημάτων

Τύπος δεδομένων των σημάτων

Προσθήκη νέου στοιχείου

Διαγραφή επιλεγμένων

Ονομασία	Τιμή	Προστασία	Δεδομένα	Διεύθυνση	Έλεγχος
<input type="checkbox"/> Άμεση φωτεινότητα δυτικής πλευράς	0,00 KLux	04/02/2010 18:17:47	0 sec	EIS 5	0/0/1
<input type="checkbox"/> Εσωτερικός φωτισμός	225,28 Lux	04/02/2010 21:51:09	1 sec	EIS 5	0/0/10

Ονομασίες των στοιχείων και ενεργοί σύνδεσμοι προς τις καρτέλες επεξεργασίας

Επιλογή στοιχείων προς διαγραφή

Σελίδα 1 από 1

Προστασία εγγραφής των σημάτων σε δευτερόλεπτα

Διευθύνσεις των στοιχείων

Κατηγοριοποίηση των στοιχείων σε δύο βασικές κατηγορίες (Αισθητήρες & Ενεργοποιητές) και σε διάφορες βοηθητικές υποκατηγορίες

Καλώς ήλθατε

smartlab

Last login 4/2/2010

Remote host 147.27.10.152

Level Διαχειριστής

Ρυθμίσεις λογαριασμού

ΕΞΟΔΟΣ

Επιλογές

Εξυπηρετητής

- Αισθητήρες
 - Μετεωρολογικός σταθμός
 - Κλιματισμός / Ποιοτ. αέρα
 - Εσωτερικός φωτισμός
 - Κατανάλωση / Ισχύς
 - Προβολή όλων
- Υπηρεσίες
- Σενάρια
- Ενεργοποιητές
- Χρήστες
- Ρυθμίσεις

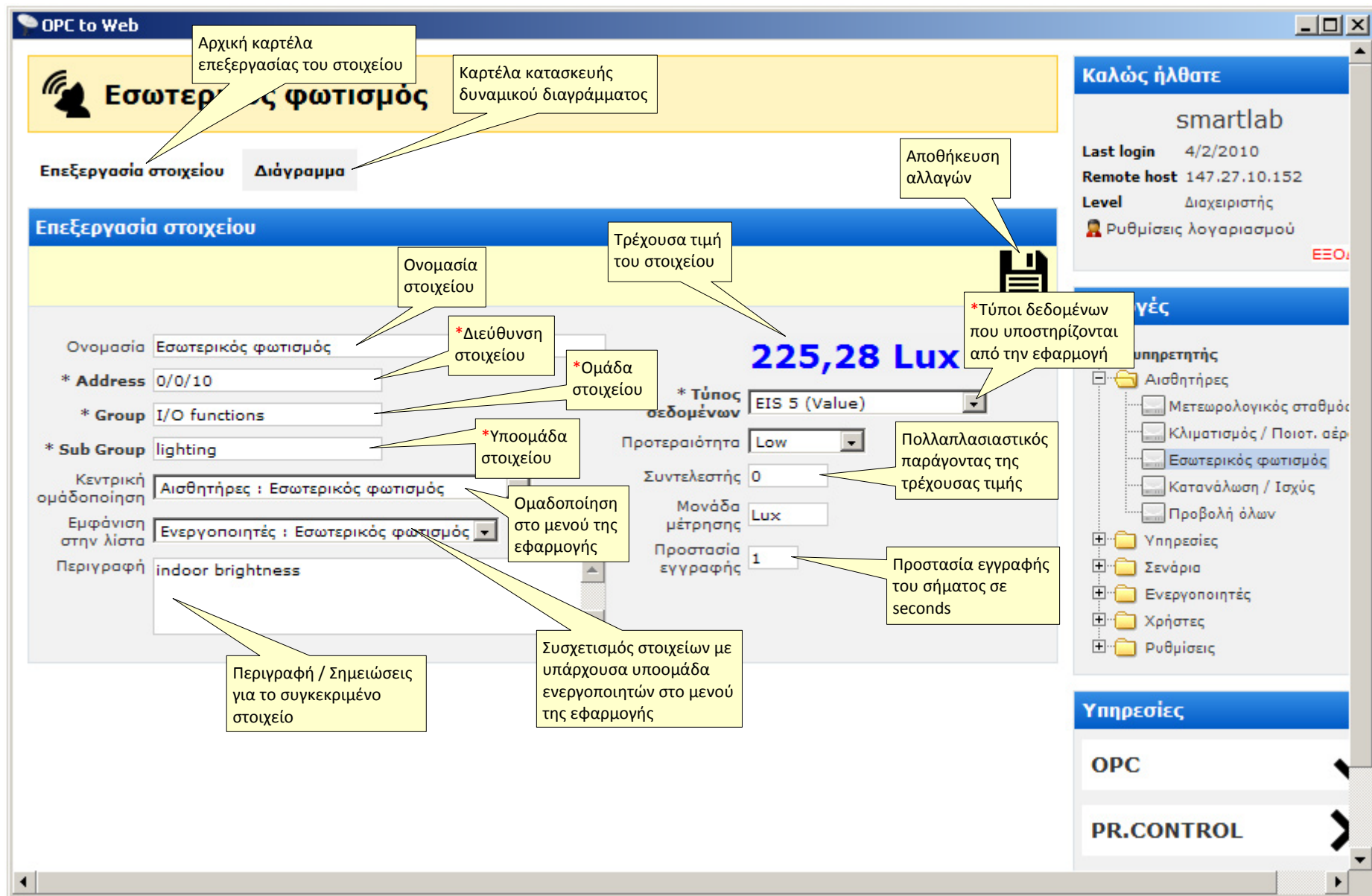
Υπηρεσίες

OPC ✓

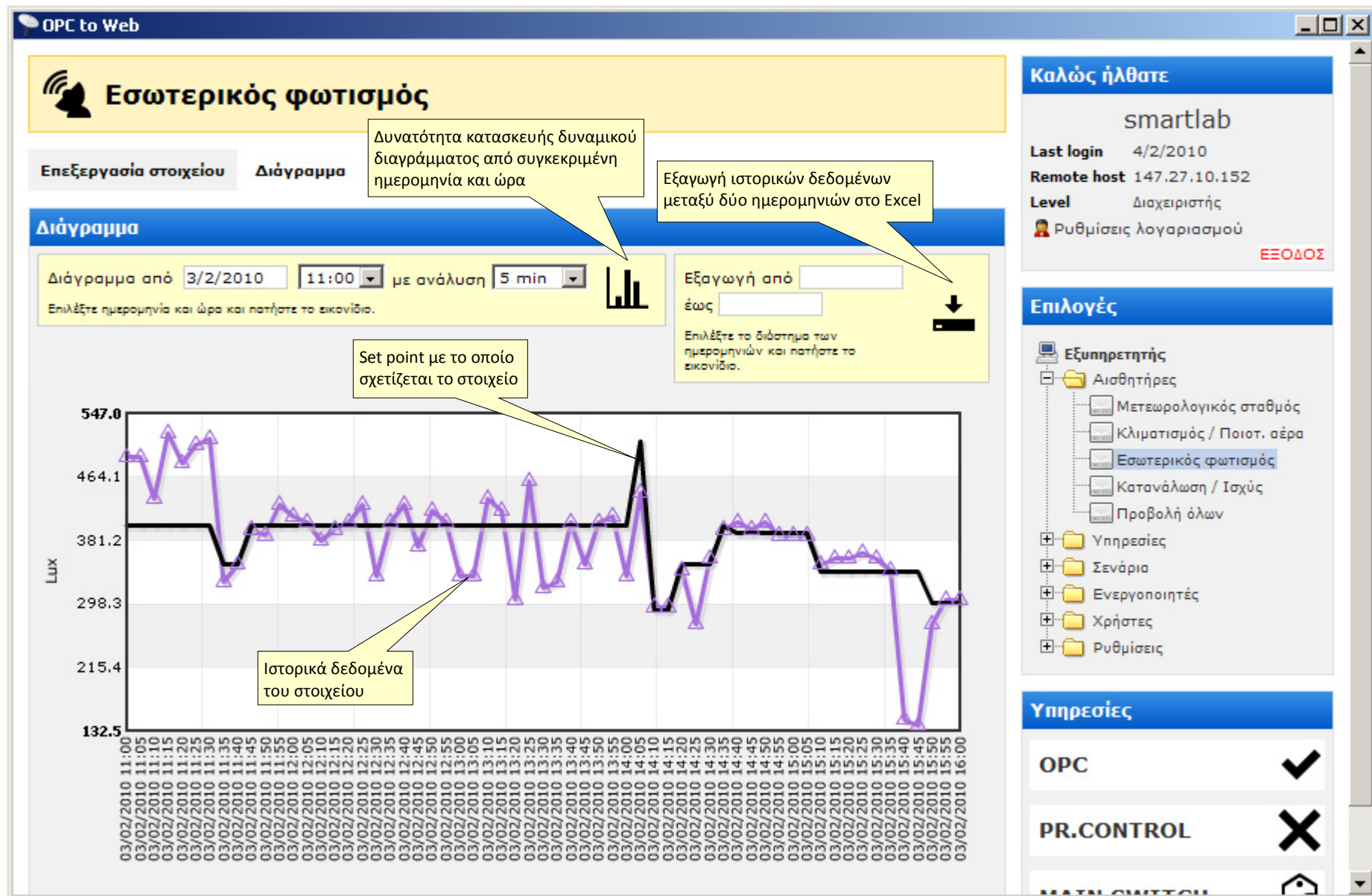
PR.CONTROL ✗

MAIN SWITCH

Εικόνα Α.1-3 Σελίδα με την λίστα των στοιχείων, περιλαμβάνει τα στοιχεία των αισθητήρων κατηγοριοποιημένα σε ομάδες.



Εικόνα Α.1-4 Σελίδα επεξεργασίας των δεδομένων για το επιλεγμένο στοιχείο. Τα πεδία με τον αστερίσκο υποδηλώνουν τα υποχρεωτικά πεδία.



Εικόνα Α.1-5 Σελίδα κατασκευής δυναμικού διαγράμματος και εξαγωγής ιστορικών δεδομένων του επιλεγμένου στοιχείου.

OPC to Web

Σενάρια

Πίνακας σεναρίων

Ονομασίες σεναρίων και ελεγκτών του μηχανισμού OPC Real Time Control

Προσθήκη νέου σεναρίου

Αποθήκευση αλλαγών και ταυτόχρονα τερματισμός ή εκκίνηση των αντιστοιχων ελεγκτών

Σελίδα 1 από 1

Τίτλος	Ενεργό Σύστημα Έναρξη		Λήξη	Καθημερινά
Οπτική άνεση V04.04-02	<input type="checkbox"/>	<input type="checkbox"/>	03/02/2010 21:00:00	03/02/2040 21:00:00 00:00:00 - 23:59:00
Ανιχνευση καπνού V01.01-02	<input type="checkbox"/>	<input checked="" type="checkbox"/>	02/02/2010 21:00:00	02/02/2040 21:00:00 00:00:00 - 23:59:59
Οπτική άνεση V02.01-02	<input type="checkbox"/>	<input type="checkbox"/>	11/11/2009 17:00:00	11/11/2040 17:00:00 00:00:00 - 23:59:00
Οπτική άνεση V03.03-02	<input checked="" type="checkbox"/>	<input type="checkbox"/>	14/11/2009 12:00:00	31/12/2030 12:00:00 09:30:00 - 22:00:00
Κλειστά φώτα το βράδυ	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	14/11/2009 12:00:00	14/11/2040 12:00:00 00:00:00 - 08:30:00
Test 01	<input type="checkbox"/>	<input type="checkbox"/>	14/11/2009 12:00:00	31/12/2040 12:00:00 00:00:00 - 23:59:59
Θερμική άνεση V04.03-02	<input checked="" type="checkbox"/>	<input type="checkbox"/>	11/11/2009 17:00:00	31/12/2040 17:00:00 00:00:00 - 23:59:59

Τα σενάρια ελέγχου που είναι ενεργοποιημένα

Τα σενάρια που παραμένουν σε λειτουργία στην περίπτωση που ζητηθεί τερματισμός των διαδικασιών αυτομάτου ελέγχου

Προσθήκη νέου σεναρίου

Επιλογές

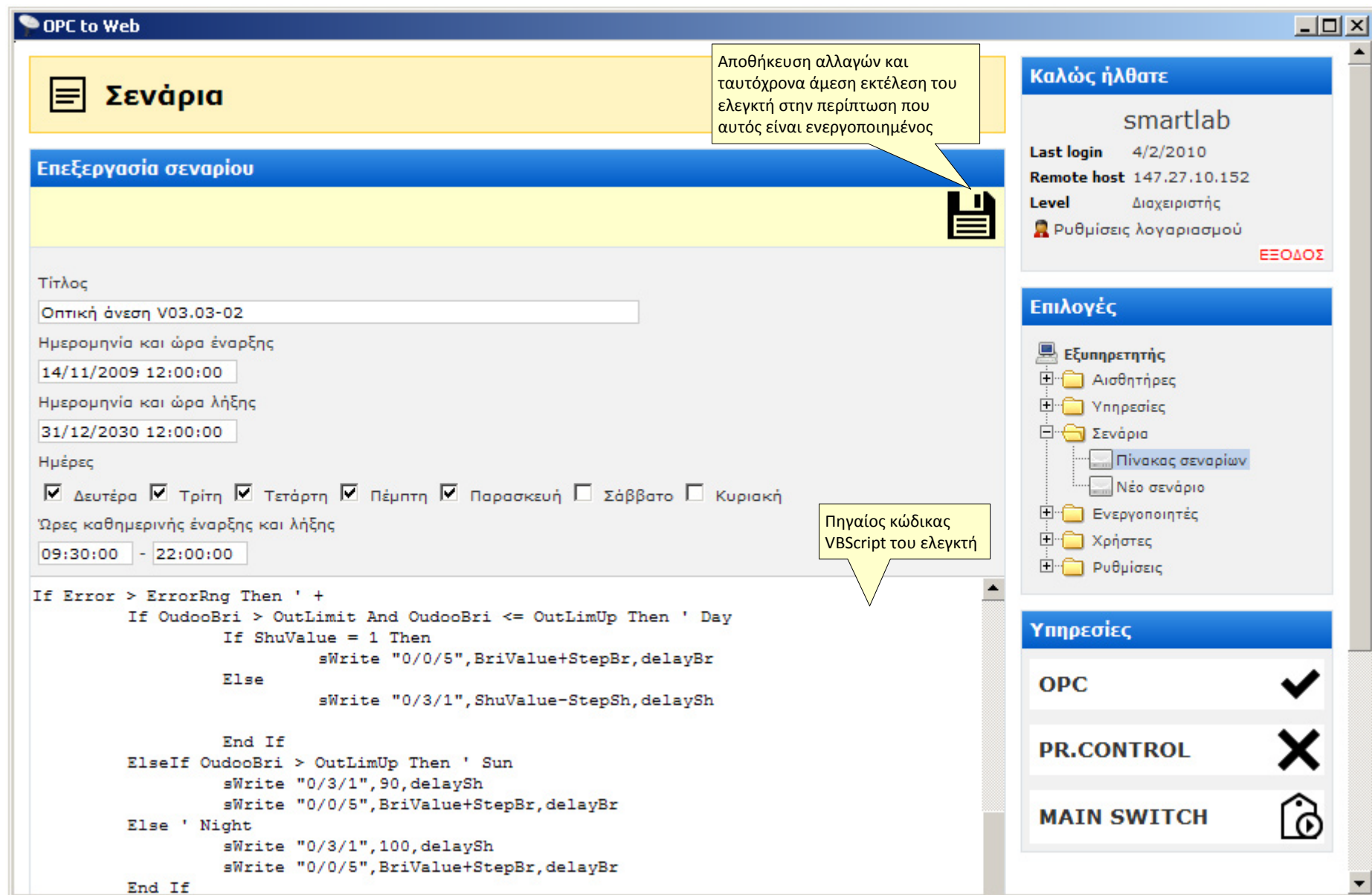
- Εξυπηρετητής
 - Αισθητήρες
 - Υπηρεσίες
 - Σενάρια
 - Πίνακας σεναρίων
 - Νέο σενάριο
 - Ενεργοποιητές
 - Χρήστες
 - Ρυθμίσεις

Υπηρεσίες

- OPC ✓
- PR.CONTROL ✗
- MAIN SWITCH 🏠

Σελίδα 1 από 1

Εικόνα Α.1-6 Σελίδα με την λίστα των σεναρίων, με δυνατότητες προβολής και ενεργοποίησης σεναρίων ελέγχου και ειδοποιήσεων.



Εικόνα Α.1-7 Σελίδα επεξεργασίας του επιλεγμένου σεναρίου.

113

OPC to Web

Τεχνικός φωτισμός

Επεξεργασία στοιχείου Διάγραμμα

Επεξεργασία στοιχείου

Ονομασία: Τεχνικός φωτισμός

* Address: 0/0/5

* Group: I/O functions

* Sub Group: lighting

Κεντρική ομάδοποίηση: Ενεργοποιητές : Εσωτερικός φωτισμός

Εμφάνιση στην λίστα: Καμία

Περιγραφή: Set brightness value

100

* Τύπος δεδομένων: EIS 6 (Scaling - percent)

Ονομασία Min: ΚΛΕΙΣΤΟΣ

Τιμή Min: 0

Ονομασία Max: ΑΝΟΙΧΤΟΣ

Τιμή Max: 100

Βήμα: 1

Μήκος: 140

Προτεραιότητα: Low

Συντελεστής: 0

Μονάδα μέτρησης:

Προστασία εγγραφής: 0

Αποθήκευση αλλαγών

Εμφάνιση πρόσθετων επιλογών μόνο για τα στοιχεία των ενεργοποιητών

Μήκος χειριστηρίου σε pixels. Δέχεται την τιμή 80 για τα χειριστήρια true / false και την τιμή 140 για τα υπόλοιπα

Καλώς ήλθατε

smartlab

Last login: 4/2/2010

Remote host: 147.27.10.152

Level: Διαχειριστής

Ρυθμίσεις λογαριασμού

Υπηρεσίες

OPC

PR.CONTROL

MAIN SWITCH

Εικόνα Α.1-9 Σελίδα επεξεργασίας των δεδομένων του επιλεγμένου στοιχείου τύπου ενεργοποιητή.

OPC to Web

Χρήστες

Πίνακας χρηστών

Σελίδα 1 από 1

Προσθήκη νέου χρήστη

Διαγραφή επιλεγμένων χρηστών

Εκτύπωση λίστας

Επίπεδο πρόσβασης

Επιλογή χρηστών προς διαγραφή

Προσθήκη νέου χρήστη

Όνομα χρήστη	Επίπεδο	Τηλέφωνο	Email	Τελευταίο IP	Τελευταία είσοδος	Κατάσταση λογαριασμού
<input type="checkbox"/> Default user	Διαχειριστής			147.27.10.160	4/2/2010	Access granted
<input type="checkbox"/> Guest account	Επισκέπτης			147.27.5.47	4/2/2010	Access granted
<input type="checkbox"/> Administrator account	Διαχειριστής			147.27.10.152	4/2/2010	Access granted
<input type="checkbox"/> Κατσιγαράκης Κυριάκος	Διαχειριστής	6938321974		79.107.134.142	2/2/2010	Access granted

Σελίδα 1 από 1

Ελάτε ήλθατε

smartlab

Last login 4/2/2010

Remote host 147.27.10.152

Level Διαχειριστής

Ρυθμίσεις λογαριασμού

ΕΞΟΔΟΣ

Εξυπηρετητής

- Αισθητήρες
- Υπηρεσίες
- Σενάρια
- Ενεργοποιητές
- Χρήστες
 - Πίνακας χρηστών
 - Νέος χρήστης
- Ρυθμίσεις

Υπηρεσίες

OPC ✓

PR.CONTROL ✗

MAIN SWITCH

Εικόνα Α.1-10 Σελίδα με την λίστα των χρηστών της εφαρμογής

OPC to Web

Guest account

Επεξεργασία χρήστη **Hostnames**

Αποθήκευση αλλαγών

Η καρτέλα δεν υποστηρίζεται προς το παρόν. Μελλοντικά οι χρήστες που θα χρησιμοποιούν συγκεκριμένο IP θα μπορούν να αναγνωρίζονται αυτόματα

Επεξεργασία χρήστη

Επίθετο: Guest Τηλέφωνο:

Όνομα: account Email:

Username: guest Τελευταίο IP: 147.27.5.47

Password: ---- Τελευταίο Login: 4/2/2010

Επίπεδο: Επισκέπτης

Κλειδωμένος: ☐

Ορισμός επίπεδου πρόσβασης

Καλώς ήλθατε

smartlab

Last login: 4/2/2010

Remote host: 147.27.10.152

Level: Διαχειριστής

Ρυθμίσεις λογαριασμού

Επιλογές

Εξυπηρετητής

- Αισθητήρες
- Υπηρεσίες
- Σενάρια
- Ενεργοποιητές
- Χρήστες
 - Πίνακας χρηστών
 - Νέος χρήστης
- Ρυθμίσεις

Υπηρεσίες

OPC ✓

PR.CONTROL ✗

MAIN SWITCH

Εικόνα Α.1-11 Σελίδα επεξεργασίας των προσωπικών δεδομένων του επιλεγμένου χρήστη.

117

Α.2 Παράμετροι εφαρμογής

Παράμετρος	Ιδιότητα
OPCServerProgID	Ορίζει το PROGID του εξυπηρετητή COM με τον οποίο πρόκειται να συνδεθεί η διαδικτυακή εφαρμογή. Την παράμετρο αυτή στην περίπτωση που δεν την αναφέρει ο προμηθευτής του OPC-Server υπάρχει διαθέσιμη στο μητρώο του λειτουργικού συστήματος (βλ. Ενότητα 2.3.1)
AjaxState	Δέχεται τιμές <i>on</i> ή <i>off</i> . Διαχειρίζεται τον μηχανισμό Ajax του συστήματος. Στην περίπτωση που γίνει <i>off</i> δεν θα ενημερώνει τις τρέχουσες τιμές των στοιχείων και δεν θα εκτελεί τα σενάρια και τους ελεγκτές του μηχανισμού OPC Real Time Control. Δημιουργήθηκε για την προστασία του κεντρικού Η/Υ από πιθανή δυσλειτουργία κάποιου υποπρογράμματος.
ItemListAjaxRefreshTime	Χρόνος παύσης σε ms μέχρι την επόμενη αποστολή ασύγχρονων αιτημάτων προς τον web server για την ανανέωση των τιμών των στοιχείων. Η τιμή εξαρτάται από την επεξεργαστική ισχύ, τον μέγιστο πιθανό αριθμό ταυτόχρονων συνδέσεων, την ταχύτητα του πληροφοριακού δικτύου και γενικά από τις απαιτήσεις της διαδικασίας ελέγχου.
ItemEditAjaxRefreshTime	Χρόνος παύσης σε ms μέχρι την επόμενη αποστολή ασύγχρονου αιτήματος προς τον web server για την ανανέωση της τιμής του στοιχείου. Η τιμή της πρέπει να είναι μικρότερη από την παράμετρο ItemListAjaxRefreshTime.
OPCServerFullPath	Πλήρης διαδρομή του εκτελέσιμου αρχείου του εξυπηρετητή OPC.
OPCServerExe	Ονομασία της διεργασίας του εξυπηρετητή OPC. Όταν «τρέχει» εμφανίζεται στο task manager του λειτουργικού συστήματος.
OPCServerName	Ψευδώνυμο του εξυπηρετητή OPC. Απαγορεύεται η χρήση του χαρακτήρα space.
ControllerFullPath	Πλήρης διαδρομή του εκτελέσιμου αρχείου του βασικού ελεγκτή. Στο σύστημα που αναπτύχθηκε ο βασικός ελεγκτής είναι το Matlab.
ControllerExe	Ονομασία της διεργασίας του βασικού ελεγκτή. Την ονομασία την εντοπίζουμε από το task manager του λειτουργικού συστήματος.

ControllerName	Ψευδώνυμο του βασικού ελεγκτή. Απαγορεύεται η χρήση του χαρακτήρα space.
ItemListAjaxRefreshControls	Χρόνος παύσης σε ms μέχρι την επόμενη αποστολή ασύγχρονων αιτημάτων για την ανανέωση των θέσεων των χειριστηρίων ελέγχου των ενεργοποιητών.
systemCheckRefreshTime	Χρόνος παύσης σε ms μέχρι την επόμενη αποστολή ασύγχρονων αιτημάτων για την ανανέωση των καταστάσεων λειτουργίας των υπηρεσιών του κεντρικού Η/Υ.
subHost	Υποδίκτυο IP στο οποίο ισχύουν βασικές λειτουργίες άμεσης πρόσβασης χωρίς την υποχρεωτική αναγνώριση των χρηστών.
subHostLock	Δέχεται τιμές <i>on</i> ή <i>off</i> . Στην τιμή <i>off</i> αφήνει τις βασικές λειτουργίες άμεσης πρόσβασης ανοιχτές σε όλο το δίκτυο. Προσοχή, η τιμή αυτή πρέπει πάντα να είναι <i>on</i> .
emailServer	Η διεύθυνση IP ή το domain του mail server (SMTP εισερχόμενης και POP3 εξερχόμενης αλληλογραφίας) που αναλαμβάνει την αποστολή των ειδοποιήσεων στις ηλεκτρονικές διευθύνσεις που ορίζονται στην συνάρτηση sEmail του μηχανισμού OPC Real Time Control.
emailAddress	Ηλεκτρονική διεύθυνση από την οποία θα αποστέλλονται οι ειδοποιήσεις.
emailUsername	Το username για την πρόσβαση στον λογαριασμό της παραπάνω ηλεκτρονικής διεύθυνσης
emailPassword	Ο αντίστοιχος κωδικός πρόσβασης του λογαριασμού
emailTitle	Δεν υποστηρίζεται. Η συγκεκριμένη ρύθμιση ορίζεται από την συνάρτηση sEmail σαν όρισμα. (βλ. Ενότητα 3.4.3)
scenarioFolder	Κεντρικός φάκελος στον οποίο αποθηκεύονται τα σενάρια που αναπτύσσουν οι χρήστες του συστήματος. Προσοχή, ο φάκελος αυτός πρέπει να διαθέτει δικαιώματα εγγραφής για τον χρήστη IUSR_computername.
batFolder	Δεν υποστηρίζεται στην παρούσα έκδοση.
scenarioRealTime	Χρόνος παύσης σε ms μέχρι την επόμενη εκτέλεση του μηχανισμού OPC Real Time Control.
scriptControllerFullPath	Πλήρης διαδρομή του αρχείου VBScript του μηχανισμού OPC Real Time Control.

scriptControllerExe	Ονομασία της διεργασίας του μηχανισμού OPC Real Time Control. Λόγω του ότι εκτελείται σε γλώσσα σεναρίων VBScript η τιμή της παραμέτρου είναι πάντα η διεργασία wscript.exe.
scriptControllerName	Ψευδώνυμο του μηχανισμού OPC Real Time Control. Απαγορεύεται η χρήση του χαρακτήρα space.
smsGatewayFrom	Δεν υποστηρίζεται. Η συγκεκριμένη ρύθμιση ορίζεται από την συνάρτηση sSms σαν όρισμα. (βλ. Ενότητα 3.4.3)
smsGatewayUser	Το username του λογαριασμού για την υπηρεσία δρομολόγησης και αποστολής των SMS.
smsGatewayPassword	Ο αντίστοιχος κωδικός πρόσβασης του λογαριασμού.
OPCServerSource	Δεν υποστηρίζεται. Η συγκεκριμένη ρύθμιση ορίζεται προγραμματιστικά και έχει πάντα την τιμή OPC_DS_CACHE (σε ακέραιο: 1).
OPCServerDriver	Ονομασία διεργασίας στο task manager η οποία τρέχει στο παρασκήνιο και περιέχει τις βιβλιοθήκες για την εγγραφή και ανάγνωση τιμών από τον εξυπηρετητή OPC προς τον δίαυλο. Το πεδίο αυτό δεν είναι υποχρεωτικό.

A.3 Κώδικας ελεγκτή οπτικής άνεσης

Ο πηγαίος κώδικας της βελτιωμένης έκδοσης (V03.03-02) του ελεγκτή οπτικής άνεσης περιλαμβάνει έλεγχο των σκιάστρων σε περίπτωση εισόδου των ηλιακών ακτίνων απευθείας στο εσωτερικό της αίθουσας. Για να το πετύχει χρησιμοποιεί το σήμα του δεύτερου αισθητήρα εσωτερικής φωτεινότητας που βρίσκεται στα δυτικά παράθυρα. Τον ελεγκτή τον εκτελεί ο μηχανισμός OPC Real Time Control μέσω της διασύνδεσης της εφαρμογής OPC to Web.

```
SetPoBri = sReadReg("HKEY_LOCAL_MACHINE\SOFTWARE\Smartlab\set_bri")
BriValue = sRead("0/0/5")
ShuValue = sRead("0/3/1")
IndooBri = sRead("0/0/10")
OudooBri = sRead("0/0/1")
ErrorRng = 5 ' Lux
OutLimit = 200 ' Lux
OutLimUp = 14500 ' Lux
Error = SetPoBri - IndooBri
If abs(Error) > 50 Then
    StepBr = 5
    delayBr = 1 ' second
    delaySh = 3 ' second
Else
    StepBr = 1
    delayBr = 5 ' second
    delaySh = 30 ' second
End If
StepSh = 50
If Error > ErrorRng Then ' +
    If OudooBri > OutLimit And OudooBri <= OutLimUp Then ' Day
        If ShuValue = 1 Then
            sWrite "0/0/5",BriValue+StepBr,delayBr
        Else
            sWrite "0/3/1",ShuValue-StepSh,delaySh
        End If
    ElseIf OudooBri > OutLimUp Then ' Sunshine
        sWrite "0/3/1",90,delaySh
        sWrite "0/0/5",BriValue+StepBr,delayBr
    Else ' Night
        sWrite "0/3/1",100,delaySh
        sWrite "0/0/5",BriValue+StepBr,delayBr
    End If
End If
If Error < -ErrorRng Then ' -
    If OudooBri > OutLimit And OudooBri < OutLimUp Then ' Day
        If BriValue = 0 Then
```

```

        sWrite "0/3/1",ShuValue+StepSh,delaySh
    Else
        sWrite "0/0/5",BriValue-StepBr,delayBr
        If ShuValue > 1 Then ' ? [fix bug]
            sWrite "0/3/1",1,delaySh
        End If
    End If
ElseIf OudooBri > OutLimUp Then ' Sunshine
    sWrite "0/3/1",90,delaySh
    sWrite "0/0/5",BriValue-StepBr,delayBr
Else ' Night
    sWrite "0/3/1","100",delaySh
    sWrite "0/0/5",BriValue-StepBr,delayBr
End If
End If

```

A.4 Κώδικας ελεγκτή θερμικής άνεσης

Ο πηγαίος κώδικας (V5.06-02) του ελεγκτή θερμικής άνεσης είναι σε θέση να διαχωρίσει τον χειμώνα από το καλοκαίρι αυτόματα ορίζοντας την νεκρή ζώνη θερμοκρασιών ανάμεσα στις επιθυμητές θερμοκρασίες του χειμώνα και του καλοκαιριού. Επίσης αν είναι εφικτό θα προτιμήσει να εκτελέσει ενέργειες παθητικού κλιματισμού παρά ενεργητικού με κύριο στόχο την εξοικονόμηση ενέργειας. Στον κώδικα χρησιμοποιείται το όρισμα delay της συνάρτησης sWrite για να ορίσει τους ελάχιστους χρόνους εκκίνησης και τερματισμού του ενεργητικού κλιματισμού. Δηλαδή σύμφωνα με την τρέχουσα παραμετροποίηση ο ελεγκτής δεν μπορεί να εκκινήσει τον ενεργητικό κλιματισμό αν δεν περάσει το χρονικό διάστημα delayAC (10 λεπτών) από την τελευταία φορά που αυτός τερμάτισε και επίσης δεν μπορεί να τον τερματίσει αν δεν περάσει το χρονικό διάστημα delayCL (4 λεπτών) από την χρονική στιγμή που αυτός εκκίνησε. Και αυτόν τον ελεγκτή τον εκτελεί ο μηχανισμός OPC Real Time Control μέσω της διασύνδεσης χρήστη της εφαρμογής OPC to Web.

```

SetPointWinter = CInt(sReadReg("HKEY_LOCAL_MACHINE\SOFTWARE\Smartlab\set_tem"))
SetPointSummer = CInt(sReadReg("HKEY_LOCAL_MACHINE\SOFTWARE\Smartlab\set_tem")) + 5
IndoorTemp = sRead("2/1/0")
OutdoorTemp = sRead("2/0/3")
Rain = sRead("2/0/0")
Wind = sRead("2/0/1")
OutBri = sRead("0/0/1")

delayAC = 600 ' seconds
delayCL = 240 ' seconds
delaySH = 60 ' seconds
delayWI = 180 ' seconds

```

Παράρτημα Α

```
bLimit = 10000 ' lux
wLimit = 1 ' m/s
eLimit = 30 ' Watt
actSta = 0 ' Heat:1 or Cool:0

If IndoorTemp < SetPointWinter Then
' Winter active air-conditioning
    actSta = 1
    If OutdoorTemp > SetPointWinter Then
        ' Passive air-conditioning Heating!

        ' 1. Close air-conditioning if working
        If sRead("3/0/1") > eLimit Then
            sWrite "4/0/1",actSta,0
            sWrite "4/0/0",0,0
            sWrite "4/0/2",0,0
        End If

        ' 2. Open windows
        If wind < wLimit And Rain = 0 Then
            sWrite "0/7/0",100,delayWI
        End If

        ' 3. Open shutters
        If OutBri > bLimit Then
            sWrite "0/3/1",1,delaySH
        End If

    Else
        ' Active air-conditioning Heating!
        sWrite "4/0/1",actSta,delayAC
        sWrite "4/0/0",1,delayAC
        sWrite "4/0/2",1,delayAC
    End If

ElseIf IndoorTemp > SetPointSummer Then
' Summer active air-conditioning
    actSta = 0
    If OutdoorTemp < SetPointSummer Then
        ' Passive air-conditioning Cooling!

        ' 1. Close air-conditioning if working
        If sRead("3/0/1") > eLimit Then
            sWrite "4/0/1",actSta,0
            sWrite "4/0/0",0,0
            sWrite "4/0/2",0,0
        End If
```

```
        ' 2. Open windows
        If wind < wLimit And Rain = 0 Then
            sWrite "0/7/0",100,delayWI
        End If

        ' 3. Close shutters
        If OutBri > bLimit Then
            sWrite "0/3/1",85,delaySH
        End If
    Else
        ' Active air-conditioning Cooling!
        sWrite "4/0/1",actSta,delayAC
        sWrite "4/0/0",1,delayAC
        sWrite "4/0/2",1,delayAC
    End If
Else
    ' Only passive air-contitioning

    If OutdoorTemp < setPointWinter Then
        actSta = 1
        ' 1. Open shutters
        If OutBri > bLimit Then
            sWrite "0/3/1",1,delaySH
        End If
    ElseIf OutdoorTemp > setPointSummer Then
        actSta = 0
        ' 1. Close shutters
        If OutBri > bLimit Then
            sWrite "0/3/1",85,delaySH
        End If
    Else
        ' nothing to change
        actSta = 0
    End If

    ' Close active air-contitioning
    If sRead("3/0/1") > eLimit Then
        sWrite "4/0/1",actSta,delayCL
        sWrite "4/0/0",0,delayCL
        sWrite "4/0/2",0,delayCL
    End If

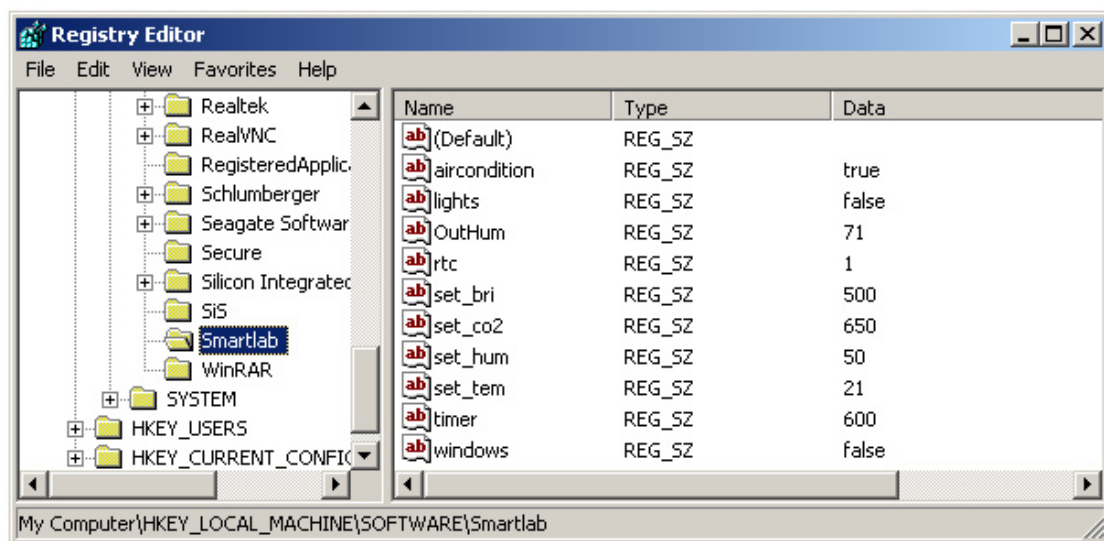
End If
```

A.5 Αρχικοποίηση του μητρώου

Ο παρακάτω πηγαίος κώδικας του Matlab αρχικοποιεί το μητρώο του λειτουργικού συστήματος με τις κατάλληλες θέσεις ώστε να έχουν την δυνατότητα οι εφαρμογές του συστήματος OPC to Web και sHumidity να γράψουν τιμές σε αυτές τις θέσεις. Πρέπει να δημιουργηθεί ένα αρχείο M-File με την ονομασία fix_registry.m και να γραφτεί η παρακάτω συνάρτηση:

```
function fix_registry
%Create structure
s=[];
s.set_tem = 26;
s.set_bri = 450;
s.set_hum = 50;
s.set_co2 = 500;
s.aircondition = 'true';
s.lights = 'true';
s.windows = 'true';
s.timer = 600;
s.rtc = 0;
%Save settings
Settings('Smartlab', s);
```

Η συνάρτηση Settings που περιέχεται στον παραπάνω κώδικα υπάρχει διαθέσιμη με ελεύθερη άδεια χρήσης στις online βιβλιοθήκες της Mathworks με τον τίτλο Windows Registry Access From Matlab (21).



Εικόνα A.5-1 Τιμές τύπου string στην θέση HKEY_LOCAL_MACHINE\SOFTWARE\Smartlab

A.6 Κώδικας εφαρμογής sHumidity

Παρακάτω δίνεται ο πηγαίος κώδικας της εφαρμογής sHumidity που μας επιστρέφει την τιμή της εξωτερικής σχετικής υγρασίας. Ο προγραμματιστής ανάπτυξης αφού μελετήσει τον κώδικα προέλευσης της ιστοσελίδας ορίζει το σημείο που θα πρέπει να επιστρέψει η εφαρμογή. Μπορεί να χρησιμοποιηθεί συνδρομητική υπηρεσία όπου σε αυτή την περίπτωση η τιμή θα δίνεται σε προκαθορισμένη θέση μέσα σε ειδικό αρχείο XML.

```
On Error Resume Next
```

```
WebSite = "http://penteli.meteo.gr/stations/chania/"
```

```
Set http = CreateObject("msxml2.xmlhttp.3.0")
```

```
http.Open "GET", WebSite, False
```

```
http.Send
```

```
str = http.responsetext
```

```
startFrom = InStr(str,"<small><strong><font color=Brown>Humidity</font></strong><br>")
```

```
txtHum = ""
```

```
txtHum = Mid(str,startFrom + 307,2)
```

```
If Len(txtHum) = 0 Then
```

```
    Hum = -1
```

```
Else
```

```
    Hum = CInt(txtHum)
```

```
End If
```

```
Set http = Nothing
```

```
If Hum >= 0 And Hum <= 100 Then
```

```
    write_registry "HKEY_LOCAL_MACHINE\SOFTWARE\Smartlab\OutHum",Hum
```

```
End If
```

```
Function write_registry(strKey,val)
```

```
    Dim WSHShell
```

```
    Set WSHShell = CreateObject("WScript.Shell")
```

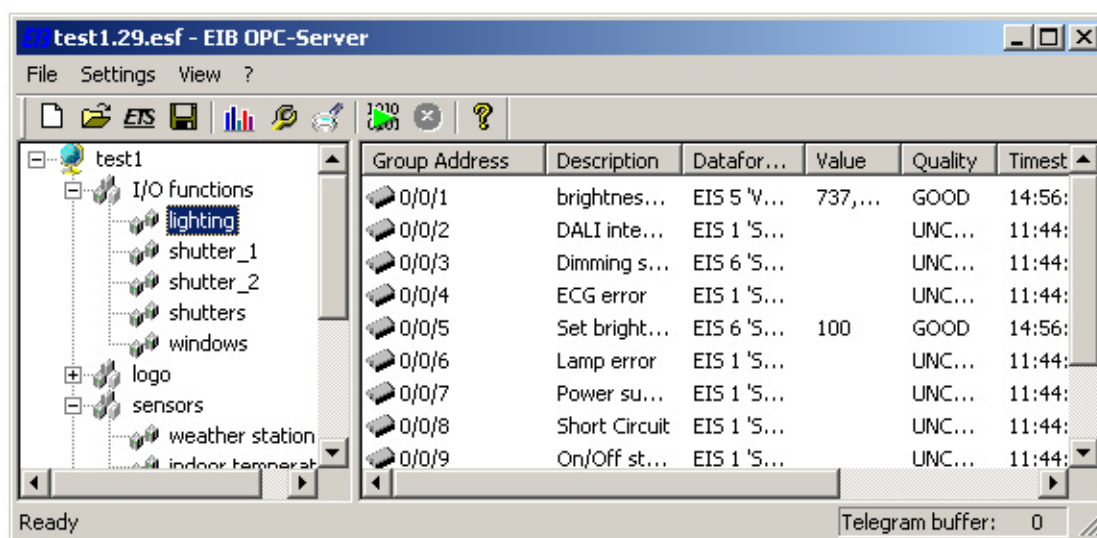
```
    WSHShell.RegWrite strKey, val
```

```
    Set WSHShell = Nothing
```

```
End Function
```

A.7 EIB OPC-Server

Ο EIB OPC-Server, είναι μια 32 bit εφαρμογή για λειτουργικά συστήματα Microsoft Windows η οποία κάνει χρήση της τεχνολογίας OPC (OLE for Process Control) και παρέχει μέσω της τεχνολογίας automation, στις εφαρμογές Microsoft Windows που την υποστηρίζουν, πρόσβαση στον δίαυλο για επικοινωνία εισόδου/εξόδου.



Εικόνα A.7-1 Εξυπηρετητής EIB OPC της EIBA

Μέσω του εξυπηρετητή EIB OPC και των σχετικών πελατών OPC ο χρήστης αποκτά πρόσβαση στους συνδρομητές μέσω μεταβλητών στις οποίες διαβάζει και γράφει τιμές οι οποίες μπορούν να μεταφερθούν σε άλλους αποδέκτες στον δίαυλο. Η διαδικασία επικοινωνίας με το δίαυλο και η ανάπτυξη δοκιμαστικών εφαρμογών είναι αρκετά γρήγορη και εύκολη διαδικασία. Ο χρήστης δεν ενδιαφέρεται για την τοπολογία και την λειτουργία του διαύλου. Αντίθετα, έχει ως προγραμματιστική διεπαφή τις μεταβλητές που παρέχονται από τον εξυπηρετητή EIB OPC.

Το αρχείο με τις διευθύνσεις ομάδας που παράχθηκε στο ETS, εισάγεται στον εξυπηρετητή EIB OPC. Αυτές οι διευθύνσεις ομάδας αποτελούν τις μεταβλητές στις οποίες αποκτά πρόσβαση ο πελάτης OPC.

Στα πλαίσια της παρούσας εργασίας χρησιμοποιήθηκε η έκδοση 2.0 του EIB OPC Server της EIBA ο οποίος μέσω των βιβλιοθηκών eteC Falcon που περιέχονται, του δίνει την δυνατότητα εγγραφής και ανάγνωσης μηνυμάτων προς και από τον δίαυλο KNX/EIB. Το αντίστοιχο PROGID του εξυπηρετητή COM του EIB OPC Server είναι το EIBA.EIBOPCServer.1 το οποίο χρησιμοποιείται από τους πελάτες OPC στην μέθοδο connect σαν παράμετρος.

A.8 Χειριστές σφαλμάτων Try...Catch για την AJAX

Για την αποστολή ασύγχρονου μηνύματος από όλα τα προγράμματα περιήγησης ιστού (Mozilla, Chrome, Safari, Opera κ.τ.λ.) προς τον εξυπηρετητή ιστού πρέπει να ενσωματωθούν σύνθετοι χειριστές σφαλμάτων Try...Catch στην Javascript ώστε να κάνει χρήση άλλων αντικειμένων όπως του XMLHttpRequest("Msxml2.XMLHTTP") ή του XMLHttpRequest("Microsoft.XMLHTTP") όταν αποτύχει το XMLHttpRequest().

```
<script language="javascript">
function getSnsList(Room){
    var url = "sensorsList.asp?Room=" + Room;
    var xmlhttp;
        try{xmlhttp = new XMLHttpRequest();
        }catch (e)
        {
            try{xmlhttp = new XMLHttpRequest("Msxml2.XMLHTTP");
            }catch (e)
            {
                try{xmlhttp = new XMLHttpRequest("Microsoft.XMLHTTP");
                }catch (e)
                {
                    alert("Αποτυχία χρήσης Ajax!");
                    return false;
                }
            }
        }

    xmlhttp.onreadystatechange=function(){
        document.getElementById("pData").innerHTML=xmlhttp.responseText;
        xmlhttp.open("GET", url, true);
        xmlhttp.send(null);
    }
}
</script>
```


A.9 Εφαρμογή OPC to DB

Η εφαρμογή OPC to DB για κάθε στοιχείο του πίνακα tbl_tems της εφαρμογής OPC to Web διαβάζει τα υποχρεωτικά δεδομένα (διεύθυνση, ομάδα και υποομάδα) και συνδέεται με το αντίστοιχο σήμα του OPC Server. Έπειτα αποθηκεύει την τιμή που διάβασε στην βάση ιστορικών δεδομένων με το DSN, OPCDATA. Επίσης διαβάζει και αποθηκεύει στην ίδια βάση δεδομένων όλες τις τρέχουσες τιμές από το μητρώο.

```
on error resume next
dbUser1 = ""
dbPwd1 = ""
dbDns1 = "OPCWEB"
dbUser2 = ""
dbPwd2 = ""
dbDns2 = "OPCDATA"
If rtn_setting("AjaxState",1) = "on" Then
Set dbConnWEB = CreateObject("ADODB.Connection")
Set dbConnDAT = CreateObject("ADODB.Connection")
Set Rs = CreateObject("ADODB.Recordset")
Set Rsl = CreateObject("ADODB.Recordset")
dbConnDAT.Open dbDns2,dbUser2,dbPwd2
Rsl.Open "tbl_data", dbConnDAT,1,3,2
Rsl.AddNew
Rsl("time") = Now()
Set objSrv = CreateObject("OPC.Automation.1")
objSrv.Connect "EIBA.EIBOPCServer.1"
dbConnWEB.Open dbDns1,dbUser1,dbPwd1
Rs.Open "SELECT * FROM tbl_items WHERE address Not Like 'xmlhttp.%' ",
dbConnWEB,1,3
If Rs.RecordCount <> 0 Then
Do until Rs.EOF
Set MyGroups = objSrv.OPCGroups
MyGroups.DefaultGroupIsActive = True
Set objGroup = MyGroups.Add("NewGroup1")
objGroup.IsSubscribed = False
Set objItem = objGroup.OPCItems.AddItem(Rs("groupName") & "." &
Rs("subGroupName") & "." & Rs("address"), 1)
objItem.Read 1, val
If Len(val) > 0 Then
If Rs("gain") > 0 Then
Rsl(CStr(Replace(Rs("address"),"/","_"))) = val * Rs("gain")
Else
Rsl(CStr(Replace(Rs("address"),"/","_"))) = val
End if
Else
Rsl(CStr(Replace(Rs("address"),"/","_"))) = 0
End If
MyGroups.RemoveAll
Set objItem = Nothing
Set objGroup = Nothing
Set MyGroups = Nothing
Rs.MoveNext
Loop
End If
Rs.Close
```

```
dbConnWEB.Close
objSrv.Disconnect
Set objSrv = Nothing
Rs1("set_tem") =
read_registry("HKEY_LOCAL_MACHINE\SOFTWARE\Smartlab\set_tem")
Rs1("set_bri") =
read_registry("HKEY_LOCAL_MACHINE\SOFTWARE\Smartlab\set_bri")
Rs1("set_co2") =
read_registry("HKEY_LOCAL_MACHINE\SOFTWARE\Smartlab\set_co2")
Rs1("set_hum") =
read_registry("HKEY_LOCAL_MACHINE\SOFTWARE\Smartlab\set_hum")
Rs1("control_aircondition") =
read_registry("HKEY_LOCAL_MACHINE\SOFTWARE\Smartlab\aircondition")
Rs1("control_windows") =
read_registry("HKEY_LOCAL_MACHINE\SOFTWARE\Smartlab\windows")
Rs1("control_lights") =
read_registry("HKEY_LOCAL_MACHINE\SOFTWARE\Smartlab\lights")
Rs1.UPDATE
Rs1.Close
dbConnDAT.Close
Set Rs = Nothing
Set Rs1 = Nothing
Set dbConnDAT = Nothing
Set dbConnWEB = Nothing
End If

Function read_registry(strRegistryKey)
    Dim WSHShell
    Set WSHShell = CreateObject("WScript.Shell")
    read_registry = WSHShell.RegRead(strRegistryKey)
    Set WSHShell = Nothing
End Function

Function rtn_setting(nameID,dbc)
    If nameID <> "" Then
        If dbc = 1 Then
            Set dbConn = CreateObject("ADODB.Connection")
            dbConn.ConnectionTimeout = 20
            dbConn.Open dbDns1,dbUser1,dbPwd1
        End If
        Set Rs2 = CreateObject("ADODB.Recordset")
        Rs2.Open "SELECT * FROM tbl_settings WHERE name='" & _
nameID & "'", dbConn,1,3
        If Rs2.Recordcount <> 0 Then
            rtn_setting = Rs2("value")
        Else
            rtn_setting = "NF" ' NOT FOUND
        End if
        Rs2.Close
        Set Rs2 = Nothing
        If dbc = 1 Then
            dbConn.Close
            Set dbConn = Nothing
        End If
        Else
            rtn_setting = "FE" ' FUNCTION ERROR
        End If
    End Function
```

A.10 Εφαρμογή Checkbot

Ο παρακάτω κώδικας εκτελείται αυτόματα με κάθε εκκίνηση του κεντρικού Η/Υ ή σε κάθε αίτηση επανεκκίνησης του OPCServer. Τερματίζει πιθανόν κάθε προβληματική διεργασία του εξυπηρετητή ή των βιβλιοθηκών eteC Falcon και τις επανεκκινεί.

```
dbUser = ""
dbPwd = ""
dbDns = "OPCWEB"

WScript.Sleep(1000)

Set objWMIService =
GetObject("winmgmts:{impersonationLevel=impersonate}!\root\cimv2")
Set colProcessList = objWMIService.ExecQuery("SELECT * FROM Win32_Process
WHERE Name = 'EIBSrvr.exe'")

For Each objProcess in colProcessList
    objProcess.Terminate()
Next

Set objWMIService = Nothing
Set colProcessList = Nothing

WScript.Sleep(1000)

Set objWMIService =
GetObject("winmgmts:{impersonationLevel=impersonate}!\root\cimv2")
Set colProcessList = objWMIService.ExecQuery("SELECT * FROM Win32_Process
WHERE Name = 'Falcon.exe'")

For Each objProcess in colProcessList
    objProcess.Terminate()
Next

Set objWMIService = Nothing
Set colProcessList = Nothing

WScript.Sleep(1000)

Set objWMIService = GetObject("winmgmts:\\.\root\cimv2:Win32_Process")
sintReturn = objWMIService.Create("C:\Program Files\EIB OPC-
Server\EIBSrvr.exe", null, null, sintProcessID)
Set objWMIService = Nothing

WScript.Sleep(2000)
```

A.11 AJAX Feed

Ο παρακάτω πηγαίος κώδικας τροφοδοτεί ανά τακτά χρονικά διαστήματα την βάση δεδομένων της εφαρμογής OPC to Web με τις τρέχουσες τιμές των σημμάτων των σημείων ελέγχου. Αποτελεί μέρος του μηχανισμού OPC Real Time Control και εκτελείται από αυτόν σαν κρυφό σενάριο του συστήματος. Η εφαρμογή OPC to Web εμφανίζει τις τρέχουσες τιμές των στοιχείων από τα αντίστοιχα πεδία της βάσης δεδομένων. Δεν συνδέεται απευθείας με τον εξυπηρετητή OPC για την ανάγνωση τιμών των σημείων ελέγχου. Μόνο στην περίπτωση εγγραφής συνδέεται απευθείας με τον εξυπηρετητή OPC.

```
dbUser1 = ""
dbPwd1 = ""
dbDns1 = "OPCWEB"

Set dbConnWEB = CreateObject("ADODB.Connection")
Set Rs = CreateObject("ADODB.Recordset")

dbConnWEB.Open dbDns1,dbUser1,dbPwd1
Rs.Open "tbl_items WHERE address Not Like 'xmlhttp.%" ", dbConnWEB,1,3,2
If Rs.RecordCount <> 0 Then

Do until Rs.EOF
    Set MyGroups = objSrv.OPCGroups
    MyGroups.DefaultGroupIsActive = True
    Set objGroup = MyGroups.Add("NewGroup1")
    objGroup.IsSubscribed = False
    Set objItem = objGroup.OPCItems.AddItem(Rs("groupName") & "." &
Rs("subGroupName") & "." & Rs("address"), 1)
    objItem.Read 1, val
    If Len(CStr(val)) > 0 Then
        Rs("valueRead") = CStr(val)
        Rs("timeStampR") = objItem.TimeStamp
    Else
        Rs("valueRead") = "0"
    End If
    Wscript.Sleep(50)
    MyGroups.RemoveAll
    Set objItem = Nothing
    Set objGroup = Nothing
    Set MyGroups = Nothing
    Rs.MoveNext
Loop

End If
Rs.Close

Rs.Open "tbl_items WHERE address Like 'xmlhttp.%" ", dbConnWEB,1,3,2
If Rs.RecordCount <> 0 Then

Do until Rs.EOF
    aSp = Split(Rs("address"),".")
```

```
Rs("valueRead") =  
read_registry("HKEY_LOCAL_MACHINE\SOFTWARE\Smartlab\" & aSp(1))  
Rs("timeStampR") = Now()  
Wscript.Sleep(50)  
Rs.MoveNext  
  
Loop  
  
End If  
Rs.Close  
dbConnWEB.Close  
  
Set Rs = Nothing  
Set dbConnWEB = Nothing
```

A.12 Κόστος εγκατάστασης

Το κόστος της εγκατάστασης KNX/EIB με τιμές 2007 (17). Τα αυξημένα κόστη εντοπίζονται στους μηχανισμούς των παραθύρων και των σκιάστρων αλλά και στο μεγάλο αριθμό συσκευών EI. Ballast για τον έλεγχο του φωτισμού.

Πίνακας A.12-1 Κόστος εγκατάστασης

Είδος	Ποσότητα	Τιμή
Αισθητήρας Φωτεινότητας 5WG1 152-4AB02	1	143,17
Binary output 5WG1 561-1AB01	1	129,12
Motor Control Relay 5WG1 524-1AB01	1	198,65
Universal I/O Module 5WG1 670-1AB03	2	344,96
EIB Gateway DALI 5WG1 141-4AB01	1	319,97
Μετρητής Ενέργειας και Ισχύος 7KT1 165	1	422,97
EI. Ballast TRIDONIC PCA one4all 2x58W	47	2726,00
EI. Ballast TRIDONIC PCA one4all 2x18W	4	232,00
Τροφοδοτικό LOGO! Power 24V DC/4A	1	71,14
Αισθητήριο CO ₂ QPA63.2	1	261,82
Αισθητήριο Θερμοκρασίας 5WG1 256-3AB01	1	124,42
Μηχανισμός παραθύρων SHD 50/450-V	2	1241,10
Ηλεκτρικά στόρια Somfy (Concept 25)	2	1134,45
Αισθητήρας Υγρασίας QFA3160	1	326,29
Αισθητήρας Υγρασίας QFA2000	1	138,09
Μετασχηματιστής Έντασης Ρεύματος 50/5 A	3	23,59
Τροφοδοτικό EIB 5WG1 125-1AB21	1	226,80
Interface RS 232 5WG1 148-1AB04	1	173,52
Data Rail 5WG1 190-8AB52	1	19,73
Weather Station 5WG1 257-3AB11	1	398,88
Αισθητήρας ταχύτητας ανέμου 5WG1 258-7AB02	1	236,16
Αισθητήρας θερμοκρασίας 5WG1 258-3AB21	1	70,13
Αισθητήρας φωτεινότητας 5WG1 258-3AB31	1	66,46
Αισθητήρας βροχής 5WG1 258-3AB41	1	146,88
Hating Transformer for W. St. 5WG1 258-8AB01	1	41,04
Mast Mounting for W. St. 5WG1 258-8AB21	1	31,38
Ηλεκτρολογικό υλικό και λοιπά έξοδα (εκτίμηση)	1	250,00
Μερικό σύνολο		€9208,72
Φ.Π.Α. 19%		€1749,66
Σύνολο		€10958,38