

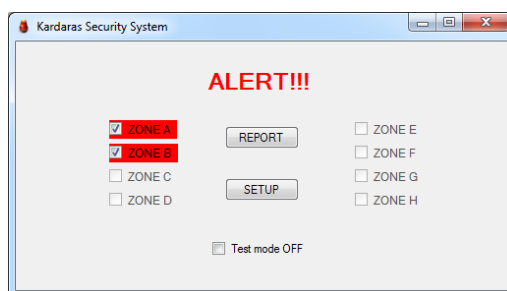


ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Υλοποίηση κεντρικού εγκεφάλου συναγερμού οικίας σε υπολογιστή με λειτουργικό σύστημα Windows ή Linux ή FreeBSD



Του φοιτητή

Καρδάρα Γεώργιου

Αρ. Μητρώου: 03/2330

Επιβλέπων καθηγητής

Δασυγένης Μηνάς

Θεσσαλονίκη 2011

ΠΡΟΛΟΓΟΣ

Η εφαρμογή και η κατασκευή δημιουργήθηκε το ακαδημαϊκό έτος 2010 – 2011 από τον Καρδάρα Γεώργιο, φοιτητή του τμήματος Πληροφορικής της Σχολής Τεχνολογικών Εφαρμογών του Α.Τ.Ε.Ι. Θεσσαλονίκης, υπό την εποπτεία του καθηγητή της σχολής κ. Μήνα Δασυγένη, στα πλαίσια εκπόνησης πτυχιακής εργασίας με τίτλο: «Υλοποίηση κεντρικού εγκεφάλου συναγερμού οικίας σε υπολογιστή με λειτουργικό σύστημα Windows ή Linux ή FreeBSD».

Στόχος μας ήταν με την ολοκλήρωση της πτυχιακής αυτής να έχουμε δημιουργήσει ένα πλήρως παραμετροποιήσιμο από εμάς, σύστημα συναγερμού οικίας του οποίου οι ενέργειες και οι λειτουργίες θα υλοποιούνται σε έναν Ηλεκτρονικό Υπολογιστή. Αφού μελετήσαμε τις λειτουργίες από τα συστήματα συναγερμού , κατασκευάσαμε μια διεπαφή υλικού από αισθητήρες που συνδέσαμε στην παράλληλη θύρα του Η/Υ και δημιουργήσαμε μια εφαρμογή λογισμικού εκπληρώνοντας σε μεγάλο βαθμό τον σκοπό μας.

ΠΕΡΙΛΗΨΗ

Στην παρούσα πτυχιακή εργασία κατασκευάστηκε ένα σύστημα το οποίο αποτελείται από υλικό και λογισμικό, και θα υλοποιεί τη κεντρική μονάδα επεξεργασίας σημάτων των αισθητήρων, αφού αρχικά δώσαμε μια περιγραφή των Συστημάτων συναγερμών, των αισθητηρίων και των μέρων τους. Συγκεκριμένα, κατασκευάστηκε μια διεπαφή υλικού η οποία συνδέεται με τον υπολογιστή μέσω της παράλληλης θύρας, και αποτελείται από έναν παθητικό αισθητήριο υπερύθρων (PIR) και επτά μαγνητικούς αισθητήρες. Επιπλέον, κατασκευάστηκε λογισμικό(με το Microsoft Visual Studio 2008) που εκτελείται στον υπολογιστή, δέχεται και επεξεργάζεται τα σήματα από τα αισθητήρια, απεικονίζει οπτικά και ηχητικά τις πιθανές παραβιάσεις ασφαλείας, και τις καταγράφει σε ιστορικό (αρχείο "logfile.txt"). Το σύστημα μπορεί να διαχωρίζει ζώνες, να ενεργοποιεί συγκεκριμένα αισθητήρια, να κατηγοριοποιεί τις ενδεχόμενες παραβιάσεις ασφαλείας αναλόγως της ζώνης ελέγχου και του είδους, και να ειδοποιεί είτε e-mail, είτε με SMS κατάλληλα το χρήστη. Υπάρχει ένα γραφικό περιβάλλον στο οποίο ο χρήστης θα μπορεί να παραμετροποιήσει το σύστημα.

ABSTRACT

In this thesis we construct a system consisting of hardware and software, and will implement the central processing unit of the sensor signals, having first a description of alarm systems, sensors and their parts. Specifically, we constructed a hardware interface which is connected to the computer via the parallel port, and consists of a passive infrared sensor (PIR) and seven magnetic sensors. In addition, we built software (in Microsoft Visual Studio 2008) running on the computer, which receives and processes signals from sensors, displays visual and audio, possible security breaches, and record a historical file ("logfile.txt"). The system can distinguish zones to trigger these sensors to classify the possible security breaches according to the control area and type, and to warn with e-mail or SMS with the appropriate user. There is a setup form where the user can customize the system.

ΕΥΧΑΡΙΣΤΙΕΣ

Δεν θα μπορούσα να παραλείψω τις ευχαριστίες προς την οικογένειά μου για την κάθε είδους στήριξη που μου παρείχε σε όλη την διάρκεια των σπουδών μου και όχι μόνο. Επίσης θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή Δρ. Μήνα Δασυγένη για την βοήθεια και την ώθηση που μου προσέφερε κατά την δημιουργία της πτυχιακής εργασίας παρόλο τον περιορισμένο χρόνο και την μεγάλη χιλιομετρική απόσταση που υπήρχε.

Περιεχόμενα

ΠΡΟΛΟΓΟΣ	3
ΠΕΡΙΛΗΨΗ	5
ABSTRACT	6
ΕΥΧΑΡΙΣΤΙΕΣ	7
Περιεχόμενα	8
Ευρετήριο Εικόνων	10
ΕΙΣΑΓΩΓΗ	13
ΚΕΦΑΛΑΙΟ 1	15
«Τα Συστήματα Συναγερμών στις μέρες μας»	15
ΕΙΣΑΓΩΓΗ	15
1.1 Κεντρική μονάδα ελέγχου	15
1.2 Πληκτρολόγιο	16
1.3 Αισθητήρια	16
1. Παθητικοί υπέρυθροι αισθητήρες (PIR)	17
2. Ενεργητικοί υπέρυθροι αισθητήρες	18
3. Μαγνητικές επαφές	20
4. Ανιχνευτές θραύσης κρυστάλλων	21
5. Ανιχνευτές καπνού	21
6. Ανιχνευτές Θερμοκρασίας	22
7. Ανιχνευτές διαρροής αερίου	23
8. Ανιχνευτές διαρροής νερού	23
1.4 Μέσα ειδοποίησης	24
1. Σειρήνες	24
2. Modem	24
3. Φωτεινές ενδείξεις	24
ΕΠΙΛΟΓΟΣ	25
ΚΕΦΑΛΑΙΟ 2	27
«Κατασκευή του Υλικού(“Hardware”)»	27
ΕΙΣΑΓΩΓΗ	27
2.1 Η Παράλληλη Θύρα	27

2.1.1 Ιδιότητες παράλληλης θύρας	29
2.1.2 Διευθύνσεις θυρών	31
Καταχωρητές λογισμικού τυποποιημένης παράλληλης θύρας (SSP) 2.1.3 ..	32
2.1.4 Αμφίδρομες θύρες.....	34
2.2 Σύνδεση αισθητηρίων	35
ΕΠΙΛΟΓΟΣ.....	38
ΚΕΦΑΛΑΙΟ 3.....	39
«Κατασκευή του Λογισμικού(“Software”）」.....	39
ΕΙΣΑΓΩΓΗ.....	39
3.1 Περιγραφή κατασκευής της εφαρμογής.....	40
3.1.1 Περιγραφή κώδικα φόρμας ρυθμίσεων(form2)	46
3.1.2 Περιγραφή κώδικα της κυρίως φόρμας (form1)	53
3.2 Περιγραφή κατασκευής οδηγού εγκατάσταση	60
ΕΠΙΛΟΓΟΣ.....	61
ΣΥΜΠΕΡΑΣΜΑΤΑ.....	63
ΒΙΒΛΙΟΓΡΑΦΙΑ	64
ΠΑΡΑΡΤΗΜΑΤΑ	65
Κώδικας κύριας φόρμας.....	65
Κώδικας φόρμας ρυθμίσεων.....	76
ΟΔΗΓΟΣ ΧΡΗΣΗΣ ΛΟΓΙΣΜΙΚΟΥ	84

Ευρετήριο Εικόνων

Εικόνα 1: Ένα τυπικό πληκτρολόγιο.....	16
Εικόνα 2: Αισθητήρας PIR	17
Εικόνα 3: Ενεργητικοί υπέρυθροι αισθητήρες.....	19
Εικόνα 4 : Μαγνητικές επαφές.....	20
Εικόνα 5: Ανιχνευτής θραύσης κρυστάλλων.....	21
Εικόνα 6: Ανιχνευτής καπνού	22
Εικόνα 7: Ανιχνευτής Θερμοκρασίας	22
Εικόνα 8: Ανιχνευτής διαρροής αερίου	23
Εικόνα 9: Ανιχνευτής διαρροής νερού	23
Εικόνα 10: Εξωτερική σειρήνα.....	24
Εικόνα 11:Παράλληλη θύρα	28
Εικόνα 12:Πίνακας των pins Παράλληλη θύρας	30
Εικόνα 13:Διευθύνσεις παράλληλων θυρών.....	31
Εικόνα 14:Πίνακας αντιστοίχισης διευθύνσεων BIOS.....	32
Εικόνα 15:Data port παράλληλης θύρας	32
Εικόνα 16:Status Port παράλληλης θύρας	33
Εικόνα 17:Control Port παράλληλης θύρας.....	33
Εικόνα 18:Πολύμετρο, καλάι και κολλητήρι	35
Εικόνα 20:Κύκλωμα της κατασκευής.....	36
Εικόνα 19:Αρσενικός συνδετήρας παράλληλης.....	36
Εικόνα 21:Σύνδεση καλωδίων και αισθητηρίων με κλέμα.....	37
Εικόνα 22:Συνδέσεις στον αισθητήρα PIR.....	37
Εικόνα 23: Μαγνητικές επαφές.....	38
Εικόνα 24:Το πάνελ με τους αισθητήρες	38
Εικόνα 25: Αρχική σελίδα του Microsoft Visual Studio 2008.....	40
Εικόνα 26: Δημιουργία νέου πρότζεκτ	40
Εικόνα 27: Παράθυρο επιλογών νέου πρότζεκτ	41
Εικόνα 28: Παράθυρο κατασκευής νέου πρότζεκτ.....	42
Εικόνα 29: Μπάρα μενού και εργαλείων.....	42
Εικόνα 31: Πλαίσιο νέας φόρμας.....	43
Εικόνα 30: Εργαλειοθήκη του Microsoft Visual Studio 2008.....	43
Εικόνα 34: Πλαίσιο Output.....	44
Εικόνα 32: Πλαίσιο Solution Explorer	44
Εικόνα 33: Πλαίσιο ιδιοτήτων	44
Εικόνα 35: Παράθυρο ρυθμίσεων του πρότζεκτ	45
Εικόνα 36:Φόρμα ρυθμίσεων (form2	45
Εικόνα 37: Παράθυρο επιλογών νέου πρότζεκτ	60
Εικόνα 38: Δημιουργία KardarasSS setup.....	61
Εικόνα 39: Φόρμα ρυθμίσεων(“SETUP”).....	85

Εικόνα 40: Επιλογή αριθμού ζωνών.....	86
Εικόνα 41: Παράθυρο υπενθύμισης επιλογής ζωνών	86
Εικόνα 42:Επιλογή αισθητηρίων	87
Εικόνα 43: Παράθυρο υπενθύμισης επιλογής αισθητηρίων	88
Εικόνα 44:Ενεργοποίηση drop down box ειδών αισθητηρίων	88
Εικόνα 45:Παράθυρο φορτώματος αρχείου ειδών αισθητηρίων.....	89
Εικόνα 46: Επιλογή είδους αισθητηρίου	90
Εικόνα 47: Παράθυρο υπενθύμισης επιλογής είδους αισθητηρίου	90
Εικόνα 48: Επιλογή τρόπου ειδοποίησης και εισαγωγή στοιχείων	91
Εικόνα 49:Παράθυρο υπενθύμισης εισαγωγής τηλεφωνικού αριθμού.....	92
Εικόνα 50:Παράθυρο υπενθύμισης εισαγωγής hostname.....	92
Εικόνα 51:Παράθυρο υπενθύμισης εισαγωγής ηλεκτρονικής διεύθυνσης.....	93
Εικόνα 52:Παράθυρο υπενθύμισης εισαγωγής κωδικού	93
Εικόνα 53:Παράθυρο αποθήκευσης ρυθμίσεων	94
Εικόνα 54:Παράθυρο φορτώματος ρυθμίσεων.....	95
Εικόνα 55:Παράθυρο κυρίας φόρμας	96
Εικόνα 56:Παραβίαση ζωνών και ενεργοποίηση συναγερμού.....	97
Εικόνα 57:Ενεργοποίηση Test Mode και εμφάνιση Test buttons.....	98

ΕΙΣΑΓΩΓΗ

Ο αρχικός στόχος της πτυχιακής εργασίας αυτής είναι η κατασκευή ενός συστήματος το οποίο αποτελείται από υλικό και λογισμικό, και θα υλοποιεί τη κεντρική μονάδα επεξεργασίας σημάτων των αισθητήρων ενός συναγερμού οικίας. Τα τυπικά συστήματα συναγερμού για την απομακρυσμένη ειδοποίηση μας βασίζονται στα σταθερά τηλεφωνικά δίκτυα κυρίως, (μερικά πιο προηγμένα στα δίκτυα της κινητής τηλεφωνίας τα οποία είναι και πιο ακριβά) και η ρύθμιση τους γίνεται από τους τεχνικούς, ενώ η ενεργοποίηση συνήθως απαιτεί την παρουσία μας. Με την ολοκλήρωση της εργασίας μας και με πολύ χαμηλό κόστος, εκτός από την διεπαφή υλικού με τους αισθητήρες, θα έχουμε μία κεντρική μονάδα εγκεφάλου συναγερμού η οποία θα βρίσκεται σε έναν απλό Ηλεκτρονικό Υπολογιστή, που θα εκτελεί όλες τις λειτουργίες θα είναι πλήρως παραμετροποιήσιμη από εμάς και θα υπάρχει καταγραφή ιστορικού. Επιπλέον με την σύνδεση του Η/Υ στο διαδίκτυο θα υπάρχει δυνατότητα ειδοποίηση μέσω mail και sms καθώς και η απομακρυσμένη διαχείριση και ενεργοποίηση του μέσω του διαδικτύου.

Στο πρώτο κεφάλαιο θα περιγράψουμε αναλυτικά τα συστήματα συναγερμών, καθώς και τα μέρη από τα οποία αποτελούνται και τις λειτουργίες και τα είδη των αισθητήριων που συνδέονται σε αυτά. Στην συνέχεια στο δεύτερο κεφάλαιο θα δώσουμε μια περιγραφή της διεπαφής του υλικού μας, αναφέροντας τις λειτουργίες της παράλληλης θύρας, τα εργαλεία και τα αισθητήρια που χρησιμοποιήσαμε και τον τρόπο σύνδεσης τους και κατασκευής του υλικού μας. Έπειτα στο τρίτο κεφάλαιο θα περιγράψουμε την δημιουργία της εφαρμογής μας, με την βοήθεια του “Microsoft Visual Studio 2008”, και των κώδικα αυτής.

Τέλος θα παραθέσουμε τυχόν επεκτάσεις και βελτιώσεις, καθώς και σε ξεχωριστή ενότητα ακολουθεί αναλυτικός «ΟΔΗΓΟΣ ΧΡΗΣΗΣ ΛΟΓΙΣΜΙΚΟΥ», όπου γίνεται αναλυτική περιγραφή εγκατάστασης, παραμετροποίησης και χρήσης του προγράμματός μας.

ΚΕΦΑΛΑΙΟ 1

«Τα Συστήματα Συναγερμών στις μέρες μας»

ΕΙΣΑΓΩΓΗ

Ο συναγερμός είναι ένα σύστημα ασφαλείας που αποτρέπει τους ανεπιθύμητους επισκέπτες να κάνουν οποιοσδήποτε κακόβουλες ενέργειες ή βανδαλισμούς προς την οικία ή την επιχείρηση μας, και επίσης μας κρατούν ενήμερους για κάθε τυχόν κίνδυνο που μπορεί να διατρέχει το κτίριο μας, όπως προειδοποίηση για διαρροή νερού ή αερίου και ανίχνευση καπνού ή αύξηση θερμοκρασίας, και γενικότερα μας βοηθάει στο να αυξήσουμε τα επίπεδα της ασφάλειας του χώρου που θέλουμε να προστατεύσουμε. Η προστασία ενός σπιτιού απέναντι σε κακόβουλες ενέργειες καθίσταται αναγκαία στις μέρες μας που η οικονομική κρίση έχει ως αποτέλεσμα την αύξηση της εγκληματικότητας. Όταν υπάρχει ένα σύστημα συναγερμού λειτουργεί αποτρεπτικά σε κάθε κακόβουλη ενέργεια, σε αντίθεση με ένα κτίριο που είναι απροστάτευτο και αποτελεί πόλο έλξης για τους ανεπιθύμητους επισκέπτες. Τα αισθητήρια των συναγερμών συνήθως εγκαθίστανται στις πόρτες, στα παράθυρα, στο γκαράζ και σε άλλα μέρη όπου είναι πιθανή μια παραβίαση. Λόγω της μεγάλης ανάπτυξης της τεχνολογίας τα συστήματα συναγερμών έχουν ηλεκτρονική μορφή στις μέρες μας. Έτσι, ένα τυπικό σύστημα συναγερμού αποτελείται μία κεντρική μονάδα ελέγχου (κεντρικός εγκέφαλος), ένα πληκτρολόγιο, διάφορα αισθητήρια και διάφορα μέσα ειδοποίησης παραβίασης ή κινδύνου.

1.1 Κεντρική μονάδα ελέγχου

Η κεντρική μονάδα ελέγχου είναι ο εγκέφαλος του συστήματος. Σε αυτήν τη συσκευή συνδέονται όλα τα υπόλοιπα στοιχεία, όπως οι αισθητήρες, το πληκτρολόγιο, οι συσκευές ειδοποίησης και οι κάρτες επικοινωνίας. Είναι ο εγκέφαλος που λαμβάνει και διαχειρίζεται τα σήματα από τους αισθητήρες, κάνει καταγραφή των συμβάντων και στη συνέχεια ενεργοποιεί τις συσκευές ειδοποίησης και ενημερώνει είτε τηλεφωνικώς είτε με κάποιον άλλο τρόπο, ανάλογα βέβαια και με τον προγραμματισμό της από το χειριστή του συστήματος.

Στην αγορά υπάρχουν πολλοί και διαφορετικοί τύποι κεντρικών μονάδων, που διαχωρίζονται βάσει των δυνατοτήτων τους. Όμως, η βασική αρχή κατασκευής τους είναι σε γενικές γραμμές η ίδια. Αποτελούνται από ένα εξωτερικό κουτί, ένα μετασχηματιστή για την ηλεκτρική τροφοδοσία του συστήματος με 6 ή 12 Volt DC,

μια μπαταρία που αποτελεί την εφεδρική ηλεκτρική παροχή σε περίπτωση εσκεμμένης ή όχι διακοπής του ρεύματος και ένα πληκτρολόγιο, αν δεν είναι ξεχωριστό.

Τα ελάχιστα βασικά χαρακτηριστικά μιας κεντρικής μονάδας που θα μπορεί να ανταποκριθεί στις τυπικές προδιαγραφές ενός αξιόπιστου συστήματος συναγερμού, είναι τα ακόλουθα: Καταρχήν, το τροφοδοτικό της μονάδας θα πρέπει να είναι σε θέση να αντέχει ρεύματα έντασης τουλάχιστον 2A για την ομαλή διαχείριση όλων των φορτίων του συστήματος. Η κεντρική μονάδα πρέπει να διαθέτει ενδείξεις της κατάστασης των μπαταριών, ώστε να είναι εφικτή ανά κάθε στιγμή η παρακολούθησή τους και να γίνεται η αντικατάστασή τους ή η επαναφόρτισή τους, πριν ολοκληρωθεί ο κύκλος ζωής τους. Μια σύγχρονη συσκευή θα πρέπει να υποστηρίζει τουλάχιστον τέσσερις διαφορετικές ζώνες, ενώ άλλο σημαντικό χαρακτηριστικό είναι η ύπαρξη ρυθμιζόμενης χρονοκαθυστέρησης, ώστε να υπάρχει το κατάλληλο χρονικό περιθώριο για την είσοδο ή έξοδο από το χώρο που προστατεύεται.

Ένα άλλο στοιχείο που θα πρέπει να λαμβάνεται υπόψη κατά τη διάρκεια της υλοποίησης του συστήματος συναγερμού είναι ότι η τροφοδοσία με 230/400 Volt AC θα πρέπει να γίνεται από ανεξάρτητη γραμμή, στην οποία δεν θα συνδέονται άλλες καταναλώσεις.

1.2 Πληκτρολόγιο

Το πληκτρολόγιο μπορεί να είναι είτε ενσωματωμένο με την κεντρική μονάδα ελέγχου, το οποίο μπορεί και να αντικατασταθεί με διακόπτη κλειδαριά (keyswitch) ή και τηλεχειρισμό, είτε ξεχωριστό. Τα πληκτρολόγια λειτουργούν σαν διακόπτες οι οποίοι ενεργοποιούνται και απενεργοποιούνται με το πάτημα ενός συνδυασμού πλήκτρων (προσωπικός κωδικός). Τα πιο κοινά πληκτρολόγια αποτελούνται από 16 πλήκτρα (10 αριθμητικά, 4 αλφαβητικά, το clear και το enter) αυξάνοντας έτσι τους πιθανούς συνδυασμούς και συνεπώς την ασφάλεια που προσφέρουν.



Εικόνα 1: Ένα τυπικό πληκτρολόγιο.

1.3 Αισθητήρια

Για να λειτουργήσει σωστά ένα σύστημα συναγερμού απαραίτητη προϋπόθεση είναι να υπάρχουν αισθητήρια στον χώρο που θέλουμε να προστατεύσουμε. Τα

αισθητήρια αυτά έχουν τη δυνατότητα να ανιχνεύουν κάποια παραβίαση ή ύποπτη κίνηση από έναν τυχόν εισβολέα, κάποια διαρροή νερού ή αερίου και την ύπαρξη καπνού ή την αύξηση θερμοκρασίας στον χώρο που θέλουμε να προστατεύσουμε. Τοποθετημένα στρατηγικά στον χώρο, μπορούν να προσφέρουν την μέγιστη ασφάλεια. Για ευκολότερη διαχείριση τους, καθώς και παρακολούθησης των σημάτων τους, ο χώρος χωρίζεται σε ζώνες στις οποίες μπορούν να ενεργοποιούνται ή να απενεργοποιούνται ξεχωριστά οι κάθε αισθητήρες. Ανάλογα με το τι ανιχνεύει το κάθε αισθητήριο υπάρχουν και διαφορετικοί τύποι αυτών. Στην συνέχεια παρουσιάζονται οι βασικότεροι τύποι αισθητηρίων από αυτούς, οι τέσσερεις πρώτοι προφυλάσσουν το κτίριο μας από τυχόν ανεπιθύμητη εισβολή, ενώ οι τέσσερεις επόμενοι μας προειδοποιούν για τυχόν κίνδυνο καταστροφής που διατρέχει το κτίριο μας :

1. Παθητικοί υπέρυθροι αισθητήρες (PIR)

Οι παθητικοί υπέρυθροι αισθητήρες που έχει επικρατήσει να αποκαλούνται PIR έχουν ευρύτατη χρήση σε συστήματα ασφαλείας. Όπως υποδηλώνει το όνομά τους, οι συγκεκριμένοι αισθητήρες είναι παθητικοί, το οποίο σημαίνει ότι δεν εκπέμπουν κανέναν είδος σήμα, αλλά δέχονται σήματα. Αναλυτικότερα, η κεφαλή του αισθητήρα είναι διαχωρισμένη σε τομείς, με τον κάθε τομέα να καθορίζεται από συγκεκριμένα όρια. Οι παθητικοί υπέρυθροι αισθητήρες ανιχνεύουν ηλεκτρομαγνητική ενέργεια η οποία δημιουργείται από πηγές οι οποίες παράγουν θερμοκρασίες χαμηλότερες απ' αυτές του ορατού φωτός. Οι αισθητήρες αυτοί μετράνε την αλλαγή της θερμικής ακτινοβολίας. Οι παθητικοί υπέρυθροι ανιχνευτές δεν εκπέμπουν κανένα σήμα. Αντιθέτως, ελέγχουν την ακτινοβολία κάθε κίνησης που γίνεται στο δωμάτιο όπου είναι εγκατεστημένοι.



Εικόνα 2: Αισθητήρας PIR

Οι παθητικοί υπέρυθροι ανιχνευτές είναι σε θέση να διακρίνουν εάν μια υπέρυθρη ακτίνα που εκπέμπει ο εισβολέας είναι παρούσα. Αυτό το επιτυγχάνει μαθαίνοντας πρώτα τη θερμοκρασία του περιβάλλοντος και έπειτα ανιχνεύοντας μια αλλαγή στη θερμοκρασία που προκαλείται από την παρουσία ενός αντικείμενου, δηλαδή του εισβολέα. Χρησιμοποιώντας την αρχή της διαφοροποίησης, η οποία είναι ένας έλεγχος της παρουσίας ή της απουσίας, οι παθητικοί υπέρυθροι ανιχνευτές ελέγχουν αν ένας εισβολέας ή κάποιο αντικείμενο βρίσκεται στο σημείο που εξετάζεται. Η αρχή της διαφοροποίησης επιτυγχάνεται δημιουργώντας μεμονωμένες ζώνες ανίχνευσης όπου κάθε ζώνη περιλαμβάνει ένα ή περισσότερα

στρώματα. Μεταξύ των ζωνών αυτών υπάρχουν νεκρές ζώνες οι οποίες χρησιμοποιούνται από τον αισθητήρα για τη σύγκριση. Δηλαδή, εντοπίζουν μια υπέρυθη εικόνα, ανιχνεύοντας την αντίθεση που υπάρχει μεταξύ της εικόνας και του ψυχρότερου περιβάλλοντος. Μονάδα μέτρησης της υπέρυθρης ακτινοβολίας είναι τα “microns”. Η εκπεμπόμενη ενέργεια από το ανθρώπινο σώμα κυμαίνεται μεταξύ των 7 έως 14 “microns”. Έτσι, όπως είναι λογικό οι παθητικοί υπέρυθροι αισθητήρες ρυθμίζονται έτσι, ώστε να ανιχνεύουν αυτής της τάξης ενέργεια.

Όμως θεωρητικά, βάση αυτής της αρχής, εάν η ενέργεια που εκπέμπει κάποιος έχει την ίδια θερμοκρασία με το περιβάλλον, τότε οι ανιχνευτές δεν θα μπορούν να τον εντοπίσουν(π.χ. αν η θερμοκρασία του δωματίου είναι γύρω στους 30 με 38 βαθμούς Κελσίου θα κάνει τους παθητικούς υπέρυθρους αισθητήρες λιγότερο αποτελεσματικούς). Για να αντιμετωπιστεί η συγκεκριμένη αδυναμία των ανιχνευτών τύπου PIR, θα πρέπει να χρησιμοποιούνται σε συνδυασμό και αισθητήρες άλλου είδους, όπως μαγνητικές επαφές και αισθητήρες ανίχνευσης σπασίματος τζαμιού, ανάλογα με το χώρο που θέλουμε να προστατέψουμε.

Μια κατάσταση που μπορεί να προκαλέσει κάποιον εσφαλμένο συναγερμό, είναι η παρουσία μικρών ζώων ή τρωκτικών στο δωμάτιο που έχουν τοποθετηθεί οι αισθητήρες. Σόμπες, φούρνοι και καυτές σωλήνες μπορούν επίσης να προκαλέσουν κάποιον εσφαλμένο συναγερμό εάν βρίσκονται στο πεδίο που “βλέπουν” οι αισθητήρες. Επιπλέον, ένα άλλο μειονέκτημα είναι ότι οι ανιχνευτές PIR δεν είναι σε θέση να φιλτράρουν το ορατό φως, οπότε μπορεί η λειτουργία τους να επηρεαστεί από πηγές εστιασμένου φωτός, όπως τους προβολείς των αυτοκινήτων.

Οι παθητικοί υπέρυθροι αισθητήρες τοποθετούνται σε τοίχους ή στην οροφή του δωματίου που θέλουμε να προστατεύσουμε, με τη διάταξη ανίχνευσης να καλύπτει τις πιθανές ζώνες διείσδυσης..

2. Ενεργητικοί υπέρυθροι αισθητήρες

Οι ενεργητικοί υπέρυθροι αισθητήρες, δημιουργούν μια ζώνη ανίχνευσης εκπέμποντας υπέρυθη ακτινοβολία. Ο αισθητήρας αντιδράει σε μια αλλαγή της εναρμόνισης της συχνότητας ή σε μια διακοπή της λαμβανόμενης ενέργειας. Αυτά τα περιστατικά συμβαίνουν όταν ο εισβολέας περάσει από την περιοχή που προστατεύεται από τον αισθητήρα με αποτέλεσμα να ενεργοποιηθεί ο συναγερμός. Οι ενεργητικοί υπέρυθροι αισθητήρες αποτελούνται από έναν πομπό και έναν δέκτη οι οποίοι βρίσκονται μέσα στην ίδια μονάδα.



Εικόνα 3: Ενεργητικοί υπέρυθροι αισθητήρες

Ο πομπός χρησιμοποιεί ένα λέιζερ για να δημιουργήσει μια προστατευτική ζώνη. Το λέιζερ εκπέμπεται σε ένα ειδικό κάτοπτρο το οποίο το οποίο χρησιμοποιείται για να καθορισθεί το φάσμα της προστατευτικής ζώνης. Η ενέργεια που εκπέμπεται από τον πομπό αντανακλάται πίσω στον δέκτη ο οποίος είναι τοποθετημένος στο ίδιο σημείο που βρίσκεται ο πομπός. Η ενέργεια αυτή, μετατρέπεται σε ένα ηλεκτρικό σήμα και όταν το σήμα αυτό κάνει περισσότερο ή λιγότερο χρόνο να επιστρέψει από τον προκαθορισμένο χρόνο που έχουμε ρυθμίσει εμείς, ενεργοποιείται ο συναγερμός.

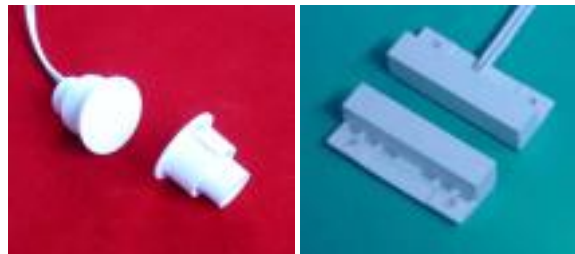
Οι ενεργητικοί υπέρυθροι αισθητήρες μπορούν να “νικηθούν” από έναν κάποιον, εάν μπορέσει να “δει” με ειδικά μηχανήματα την ακτίνα δράσης των ανιχνευτών και αποφύγει την πορεία της δέσμης του λέιζερ.

Την ανιχνευτική ικανότητα του συστήματος μπορούν να την επηρεάσουν παράγοντες που διαταράσσουν τη μετάδοση της φωτεινής δέσμης, όπως ομίχλη, καπνός ή σκόνη. Επίσης, κάθε αντικείμενο ή ζώο που παρεμβαίνει στην πορεία της δέσμης μπορεί να ενεργοποιήσει το συναγερμό και να παραπλανήσει τους υπεύθυνους ασφαλείας του χώρου.

Συνήθως, οι συγκεκριμένοι ανιχνευτές χρησιμοποιούνται για την προστασία εισόδων, προθάλαμων, περιμέτρων ή ακόμα και την κάλυψη ενός τοίχου με ιδιαίτερα αυξημένο μήκος. Η απόσταση μεταξύ δέκτη και πομπού, ώστε το σύστημα ανίχνευσης να παρέχει ικανοποιητική κάλυψη μπορεί να είναι μέχρι κάποιες εκατοντάδες μέτρα. Οι ανιχνευτές αυτοί, δεν επηρεάζονται από τυχόν εκπομπές θερμότητας, από λαμπτήρες φθορισμού ή από διάφορες ηλεκτρονικές παρεμβολές. Έχουν καλά ποσοστά ανίχνευσης, με ταυτόχρονα, μικρό δείκτη εμφάνισης λανθασμένων συναγερμών. Επίσης, η πορεία των δεσμών μπορεί να μεταβληθεί με τη χρήση καθρεπτών, κάνοντας ακόμα δυσκολότερη την προσέγγιση στον προστατευόμενο χώρο. Βέβαια πρέπει να ληφθεί υπόψη ότι η χρήση καθρεπτών εξασθενίζει την ένταση της δέσμης και μειώνει την εμβέλεια δράσης της.

3. Μαγνητικές επαφές

Οι μαγνητικές επαφές χρησιμοποιούνται για την ανίχνευση του ανοίγματος μιας πόρτας ή ενός παραθύρου. Το αισθητήριο αυτό είναι ένας διακόπτης δύο θέσεων και σχεδιάζεται ώστε να ανοίγει και να κλείνει κανονικά. Ο διακόπτης είναι στην κανονική του θέση όταν η πόρτα ή το παράθυρο είναι κλειστό. Ένα μαγνητικό πεδίο δημιουργείται και το μαγνητικό πεδίο αυτό κρατάει τον διακόπτη ώστε ο συναγερμός να είναι ανενεργός. Όταν η πόρτα ή το παράθυρο ανοίξει, ο μαγνήτης ελευθερώνει το διακόπτη και δίνει το σήμα του συναγερμού. Οι μαγνητικές επαφές τοποθετούνται συνήθως στα πλαίσια των πορτών και των παραθύρων και χρησιμοποιούνται σε συνδυασμό τους αισθητήρες κίνησης για να ανιχνεύεται όποια παραβίαση γίνει. Υπάρχει μεγάλη ποικιλία σχημάτων μαγνητικών επαφών ανάλογα με το μέρος το οποίο θα εγκατασταθούν.



Εικόνα 4 : Μαγνητικές επαφές

Η λανθασμένη τοποθέτηση της μαγνητικής επαφής με μη ευθυγραμμισμένο τον μαγνήτη που είναι πάνω στην πόρτα ή στο παράθυρο με τον μαγνητικό διακόπτη που βρίσκεται στο πλαίσιο της πόρτας ή του παραθύρου μπορεί να προκαλέσει εσφαλμένους συναγερμούς. Η εσφαλμένη ευθυγράμμιση μπορεί να προκληθεί από ένα στραβωμένο ή από ένα παραμορφωμένο πλαίσιο πόρτας ή παραθύρου. Οι διακυμάνσεις στο πλαίσιο της πόρτας ή του παραθύρου λόγω της υπερβολικής χρήσης, ή ακόμα και των αλλαγών της θερμοκρασίας το καλοκαίρι και τον χειμώνα, μπορούν, επίσης να κάνουν την μαγνητική επαφή να προκαλέσει ένα εσφαλμένο συναγερμό.

Η κύρια μέθοδος για να “ξεγελαστεί” μία μαγνητική επαφή από κάποιον είναι να χρησιμοποιήσει έναν δυνατό μαγνήτη ώστε να κρατήσει την επαφή και να του επιτρέψει να ανοίξει την πόρτα ή το παράθυρο χωρίς να ενεργοποιηθεί ο συναγερμός. Αυτό επιτυγχάνεται επειδή η θέση που βρίσκεται ο διακόπτης είναι ορατή στον διαρρήκτη, επιτρέποντας του να μπορεί να τοποθετήσει κατάλληλα τον ισχυρότερο μαγνήτη ώστε να παραβιάσει τον μαγνητικό διακόπτη της επαφής μιας πόρτας ή ενός παραθύρου.

Για την αξιόπιστη λειτουργία τους, θα πρέπει να δοθεί ιδιαίτερη προσοχή κατά την κατασκευή των πόρτων και των παραθύρων και στην εφαρμογή τους με την κάσα. Ακόμα υψηλότερο επίπεδο ασφάλειας παρέχουν οι ισοσταθμισμένοι μαγνητικοί ανιχνευτές, που αποτελούνται από δύο μαγνήτες. Ανάμεσα στους δύο μαγνήτες εκ των οποίων ο ένας είναι τοποθετημένος στο σταθερό πλαίσιο και ο άλλος στο κινητό μέρος της πόρτας ή του παραθύρου δημιουργείται ένα ισοσταθμισμένο και σταθερό μαγνητικό πεδίο. Τυχόν προσπάθεια διάρρηξης διαταράσσει τη σταθερότητα του πεδίου, με αποτέλεσμα να ενεργοποιείται ο συναγερμός.

4. Ανιχνευτές θραύσης κρυστάλλων

Οι αισθητήρες ανίχνευσης θραύσης κρυστάλλων μπορούν να χρησιμοποιηθούν για την εσωτερική ή για την περιμετρική προστασία ενός κτηρίου. Οι αισθητήρες θραύσης κρυστάλλων χρησιμοποιούν είτε μικρόφωνα ώστε να μπορούν να “ακούν” ήχους που σχετίζονται με το σπάσιμο ενός τζαμιού, είτε αισθητήρες δόνησης που “αισθάνονται” τις συχνότητες κλονισμού και τους ήχους συγκεκριμένων συχνοτήτων που παράγονται όταν σπάσει το τζάμι, οι οποίοι ταξιδεύουν μέσω του γυαλιού και συχνά μέσω του πλαισίου παραθύρων, των περιβαλλόντων τοίχων και της οροφή, είτε συνδυασμό και των δύο παραπάνω για την μείωση πιθανοτήτων λανθασμένων συναγερμών.



Εικόνα 5: Ανιχνευτής θραύσης κρυστάλλων

Τα σημεία που συνήθως τοποθετούνται οι αισθητήρες θραύσης κρυστάλλων είναι το παράθυρο(τζάμι), το πλαίσιο του παραθύρου, ο τοίχος ή η οροφή. Τέλος, για καλύτερα αποτελέσματα καλό θα ήταν οι αισθητήρες σπασίματος τζαμιού να χρησιμοποιούνται σε συνδυασμό με μαγνητικές επαφές.

5. Ανιχνευτές καπνού

Οι ανιχνευτές καπνού είναι αισθητήρες που μας προειδοποιούν για τυχόν ύπαρξη καπνού φωτιάς και μας παρέχουν ένα σωτήριο προβάδισμα για την περιουσία μας αλλά και για την ίδια μας την ζωή. Οι ανιχνευτές καπνού λειτουργούν σύμφωνα με την αρχή της σκέδασης του φωτός. Στο θάλαμο ανίχνευσης, που είναι στο εσωτερικό του αισθητηρίου, εκπέμπονται ακτίνες φωτός από μία διοδική λυχνία. Όταν εισέλθει καπνός οι ακτίνες αυτές διασκορπίζονται και κατευθύνονται σε έναν

φωτογραφικό φακό, ο οποίος αναγνωρίζει τον καπνό φωτιάς και ο αισθητήρας δίνει σήμα και ενεργοποιείται ο συναγερμός.



Εικόνα 6: Ανιχνευτής καπνού

Συνήθως τοποθετούνται στα ταβάνια δωματίων των οποίων θέλουμε να ελέγχουμε και όπου υπάρχει υψηλός κίνδυνος εκδήλωσης πυρκαγιάς.

6. Ανιχνευτές Θερμοκρασίας

Οι ανιχνευτές θερμοκρασίας μας προειδοποιούν για επικίνδυνη άνοδο θερμοκρασίας και τυχόν ύπαρξη φωτιάς. Οι ανιχνευτές θερμοκρασίας είναι θερμοδιαφορικός αισθητήρας με ανώτατο όριο ενεργοποίησης (συνήθως τους 60°C). Χρησιμοποιεί ένα “thermistor” (κάτι σαν μία θερμική αντίσταση), εγκατεστημένο εξωτερικά και ηλεκτρονικό κύκλωμα ώστε να μπορεί να παρέχει ακριβή γραμμική ανίχνευση και μέτρηση της θερμοκρασίας του χώρου. Ανάλογα με την θερμοκρασία του χώρου το “thermistor” δημιουργεί τάση. Ανάλογα με την ταχύτητα ανόδου της θερμοκρασίας αυτής ενεργοποιείται ή όχι ο ανιχνευτής. Εάν η θερμοκρασία φτάσει το ανώτατο όριο, οποιαδήποτε στιγμή, ο ανιχνευτής ενεργοποιείται.



Εικόνα 7: Ανιχνευτής Θερμοκρασίας

Και αυτοί τοποθετούνται στα ταβάνια δωματίων των οποίων θέλουμε να ελέγχουμε και υπάρχει υψηλός κίνδυνος εκδήλωσης πυρκαγιάς. Τέλος είναι ιδανικοί για εγκατάσταση σε χώρους με μέση θερμοκρασία περιβάλλοντος, όπως χώρους καυστήρων, στεγνωτήρια/καθαριστήρια (υδρατμοί) ή σε χώρους που οι ανιχνευτές καπνού δεν είναι δυνατό να εγκατασταθούν λόγω παρουσίας ατμών ή αερίων που προέρχονται από μαγείρεμα, όπως οι κουζίνες.

7. Ανιχνευτές διαρροής αερίου

Οι ανιχνευτές διαρροής αερίων μας ενημερώνουν για ύπαρξη αερίων, τα οποία είναι είτε τοξικά είτε εύφλεκτα και κατά συνέπεια βλαβερά και επικίνδυνα προς εμάς. Οι ανιχνευτές αυτοί χρησιμοποιούν τρεις μέθοδοι μέτρησης και παρακολούθησης αερίων:

- την μέθοδο με την χρήση ηλεκτροχημικών αισθητήρων, οι οποίοι αρμόζουν στις περιπτώσεις ανίχνευσης τοξικών αερίων έχοντας κατώτατο εύρος περιεκτικότητας της τάξης του εκατομμυριοστού(ppm) κυρίως,
- και τις μεθόδους με την χρήση καταλυτικών αισθητήρων και με την χρήση αισθητήρων τεχνολογίας IR, οι οποίες χρησιμοποιούνται για την ανίχνευση συγκέντρωσης εύφλεκτων και εκρηκτικών ουσιών κάτω από χαμηλότερο σημείο έκρηξης.



Εικόνα 8: Ανιχνευτής διαρροής αερίου

Τοποθετούνται κοντά σε αγωγούς αερίων και σε χώρους που μπορεί να υπάρξει διαρροή όπως εργαστήρια, καυστήρες φυσικού αερίου κ.α. και χρησιμοποιούνται κυρίως σε βιομηχανίες και διυλιστήρια όπου υπάρχει υψηλός κίνδυνος διαρροής αερίων, και η ανάγκη επιτήρησης και ελέγχου των διαρροών έρχεται σε πρώτη προτεραιότητα.

8. Ανιχνευτές διαρροής νερού

Οι ανιχνευτές διαρροής νερού μας ειδοποιούν για τυχόν ύπαρξη νερού ή γενικότερα υγρού σε χώρο που δεν είναι επιθυμητό, μπορούν να χρησιμοποιηθούν και για την ενημέρωση για τυχόν άνοδο στάθμης υγρών πάνω από επιτρεπτό όριο για το συγκεκριμένο χώρο. Οι ανιχνευτές αυτοί αποτελούνται από ένα αισθητήριο με δυο ακροδέκτες, οι οποίοι μόλις έρθουν σε επαφή με υγρό τότε στέλνει σήμα και ενεργοποιείται ο συναγερμός.



Εικόνα 9:
Ανιχνευτής
διαρροής νερού

Ένα από τα πιο ανεπιθύμητα φαινόμενα που μπορούν να

επηρεάσουν έναν χώρο είναι αυτό της διαρροής νερού. Καταστροφή της περιουσίας, φθορά επίπλων, ατυχήματα από γλίστρημα ή ηλεκτροπληξία είναι μόνο μερικοί από τους κινδύνους ενός τέτοιου συμβάντος.

1.4 Μέσα ειδοποίησης

Τα μέσα ειδοποίησης αποτελούν, ένα εξίσου σημαντικό στοιχείο των συστημάτων συναγερμού, καθώς μας ενημερώνουν ή μας προειδοποιούν ότι υπάρχει πρόβλημα ασφάλειας ή παραβίασης στον προστατευόμενο χώρο. Συνήθως, αυτά τα μέσα αποτελούνται από:

1. Σειρήνες

Συνήθως, χρησιμοποιούνται τουλάχιστον δύο σειρήνες: η μία τοποθετείται εσωτερικά και η άλλη στον εξωτερικό χώρο. Η επιλογή του σημείου που θα τοποθετηθεί η εξωτερική σειρήνα είναι ιδιαίτερης σημασίας, καθώς αποτελεί το μόνο σημείο του συστήματος που είναι εκτεθειμένο. Συνιστάται να τοποθετείται σε σημεία μεγάλου ύψους, ώστε να εξασφαλίζεται ο συνδυασμός της μέγιστης ορατότητας και του μικρότερου βαθμού προσβασιμότητας. Ο ήχος τους είναι πολύ δυνατός και κυμαίνεται από 110 db έως 125 db, καθώς κάνει αισθητή την παραβίαση στον γειτονικούς χώρους με την ενεργοποίησή της. Η τοποθέτηση μιας εσωτερικής σειρήνας έχει σκοπό να, κυρίως να πανικοβάλλει τον ανεπιθύμητο επισκέπτη.



Εικόνα 10:
Εξωτερική σειρήνα

2. Modem

Το modem στέλνει τα σήματα συναγερμού στα κέντρα λήψεως σημάτων. Τα κέντρα λήψεως σημάτων αποκωδικοποιούν τα σήματα που στέλνει το σύστημα ασφαλείας και προβαίνουν στις κατά περίπτωση ενέργειες (ειδοποιούν την αστυνομία, την πυροσβεστική, άτομα που εμείς έχουμε επιλέξει κ.λπ.). Σε περίπτωση που δεν υπάρχει τέτοιο κέντρο, χρησιμοποιούνται εναλλακτικά τα GSM Modems που παρέχουν την δυνατότητα στον χρήστη να ενημερώνεται να πάσα στιγμή για τα σήματα του συναγερμού, μέσω τεχνολογίας SMS ή MMS.

3. Φωτεινές ενδείξεις

Οι φωτεινές ενδείξεις μπορεί να είναι προβολείς, περιστρεφόμενες λάμπες (φάρος), φλας κ.α..

Όλα τα παραπάνω έχουν σαν σκοπό να αποτρέψουν πιθανές εισβολές κάνοντας αισθητή την παρουσία του συστήματος (Εξωτερική σειρήνα σε εμφανές σημείο), να

πανικοβάλουν τους διαρρήκτες και τέλος να ειδοποιούν μέσω 24ώρου κεντρικού σταθμού λήψεως σημάτων τον ιδιοκτήτη, τις Αρχές ή τους αρμόδιους

ΕΠΙΛΟΓΟΣ

Τελειώνοντας με την περιγραφή των τυπικών συστημάτων συναγερμού στις μέρες μας και των βασικών χαρακτηριστικών τους θα περάσουμε στην περιγραφή της χρησιμοποίησης και της κατασκευής του υλικού μας. Η κατασκευή αυτή θα αποτελείται από τα αισθητήρια τα οποία θα συνδέονται με τον Ηλεκτρονικό Υπολογιστή μας ο οποίος με την χρησιμοποίηση του προγράμματος μας θα αντικαταστήσει την λειτουργία της κεντρικής μονάδας ελέγχου, του πληκτρολογίου και μερικών εκ των μέσων ειδοποίησης.

ΚΕΦΑΛΑΙΟ 2

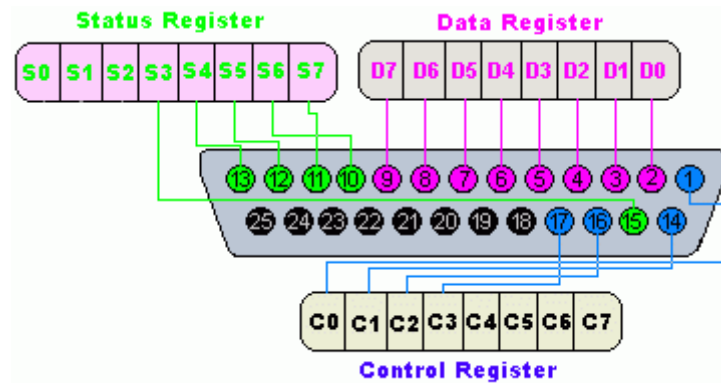
«Κατασκευή του Υλικού(“Hardware”)»

ΕΙΣΑΓΩΓΗ

Στο κεφάλαιο αυτό θα γίνει περιγραφή στον τρόπο κατασκευής του υλικού και στην χρήση του, καθώς και στο τι εργαλεία χρησιμοποιήθηκαν και στον τρόπο που έγινε αυτό. Επιπλέον θα αναφερθούμε και στον τρόπο επικοινωνίας του υλικού με τον Ηλεκτρονικό Υπολογιστή, που στην περίπτωσή μας είναι η παράλληλη θύρα του Η/Υ. Επομένως για την αντικατάσταση της κεντρικής μονάδας ελέγχου του συναγερμού χρησιμοποιήσαμε έναν απλό Η/Υ με αμφίδρομη παράλληλη θύρα και χωρίς ιδιαίτερα μεγάλη επεξεργαστική ισχύ. Ακόμη χρειαστήκαμε ένα καλώδιο παράλληλης θύρας, ένα ψηφιακό πολύμετρο, για την μέτρηση των τιμών αμπέρ και βολτ, ένα κολλητήρι και καλάι, για την συγκόλληση των καλωδίων, ένα κλέμα, για την ένωση των καλωδίων με τα αισθητήρια, και χρησιμοποιήσαμε επτά μαγνητικές επαφές και έναν παθητικό υπέρυθρο αισθητήρα (PIR), τους οποίους προσαρμόσαμε πάνω σε μια ξύλινη κατασκευή με βίδες και μεντεσέδες, που φτιάξαμε για πάνελ, και μια μικρή μπαταρία εμπορίου 12v για τον αισθητήρα PIR.

2.1 Η Παράλληλη Θύρα

Η παράλληλη θύρα είναι η συνηθέστερα χρησιμοποιημένη θύρα για τη διασύνδεση απλών εφαρμογών. Αυτή η θύρα επιτρέπει την εισαγωγή μέχρι 9 μπιτ ή την εξαγωγή 12 μπιτ σε οποιοδήποτε δεδομένο χρόνο, απαιτώντας κατά συνέπεια ελάχιστα εξωτερικά στοιχεία κυκλώματος για να εφαρμοστεί σε πολλές απλές εφαρμογές. Η θύρα αποτελείται από 4 γραμμές ελέγχου (Control register), 5 γραμμές κατάστασης (Status register), 8 γραμμές δεδομένων (Data register) και 8 γραμμές γείωσης (Εικόνα 11). Βρίσκεται συνήθως στο πίσω μέρος του PC ως D-type 25 θηλυκός σύνδεσμος.



Εικόνα 11: Παράλληλη θύρα

Οι νεώτερες παράλληλες θύρες είναι τυποποιημένες κάτω από το IEEE 1284 πρότυπο που παρουσιάστηκε το 1994, Αυτό το πρότυπο καθορίζει 5 τρόπους λειτουργίας που είναι οι ακόλουθοι:

1. Compatibility Mode.
2. Nibble Mode.
3. Byte Mode.
4. EPP Mode (Enhanced Parallel Port).
5. ECP Mode (Extended Capabilities Port).

Ο στόχος ήταν να σχεδιαστούν νέοι οδηγοί και συσκευές που θα ήταν συμβατοί μεταξύ τους και προς τα πίσω συμβατοί στη διασύνδεση τους με μια τυποποιημένη παράλληλη θύρα (SSP). Οι λειτουργίες Compatibility, Nibble & Byte χρησιμοποιούν ακριβώς το τυποποιημένο διαθέσιμο υλικό στις αρχικές παράλληλες κάρτες ενώ οι λειτουργίες του EPP & ECP απαιτούν πρόσθετο υλικό που μπορεί να τρέξει με γρηγορότερες ταχύτητες, ενώ ακόμη όντας προς τα πίσω συμβατό με την τυποποιημένη παράλληλη θύρα.

Η λειτουργία Compatibility ή "Centronics" όπως είναι περισσότερο γνωστή, μπορεί μόνο να στείλει τα στοιχεία εμπρός με μια χαρακτηριστική ταχύτητα 50 kbyte ανά δευτερόλεπτο αλλά μπορεί να είναι και υψηλότερη όπως 150 + kbyte το δευτερόλεπτο. Προκειμένου να παραληφθούν τα στοιχεία, πρέπει να αλλάξετε σε λειτουργία nibble ή byte. Η λειτουργία nibble μπορεί να εισαγάγει (4 μπιτ) στην αντίστροφη κατεύθυνση. Π.χ. από τη συσκευή στον υπολογιστή. Η λειτουργία byte χρησιμοποιεί το αμφίδρομο χαρακτηριστικό γνώρισμα της παράλληλης (που βρίσκεται μόνο σε μερικές κάρτες) για να εισαγάγει μια ψηφιολέξη (8 μπιτ) στην αντίστροφη κατεύθυνση.

Οι παράλληλες EPP & ECP χρησιμοποιούν πρόσθετο υλικό για να παραγάγουν και να διαχειριστούν χειραψία (handshaking). Για να στείλετε ένα byte σε έναν εκτυπωτή (ή σε κάτι παρόμοιο) που χρησιμοποιεί την λειτουργία Compatibility, το λογισμικό πρέπει να κάνει τα εξής:

1. γράψτε το byte στη θύρα δεδομένων.
2. ελέγξτε για να δείτε αν ο εκτυπωτής είναι απασχολημένος. Εάν ο εκτυπωτής είναι απασχολημένος, δεν θα δεχτεί οποιαδήποτε στοιχεία, κατά συνέπεια οποιαδήποτε στοιχεία που γράφονται θα χαθούν.
3. κάντε το strobe (pin 1) χαμηλό. Αυτό λέει στον εκτυπωτή ότι υπάρχει ένα σωστό στοιχείο όσον αφορά τις γραμμές δεδομένων. (pin 2-9)
4. κάντε το strobe υψηλό πάλι αφού περιμένετε περίπου 5 s αφότου κάνατε το strobe χαμηλό. (Βήμα 3)

Αυτό περιορίζει την ταχύτητα με την οποία η θύρα μπορεί να λειτουργήσει. Οι θύρες του EPP & ECP το προσπερνούν αυτό αφήνοντας τον έλεγχο στο υλικό για να δουν εάν ο εκτυπωτής είναι απασχολημένος και παράγουν ένα strobe ή/και μια κατάλληλη χειραψία. Αυτό σημαίνει ανάγκη για μόνο μια I/O οδηγία να εκτελεσθεί, αυξάνοντας κατά συνέπεια την ταχύτητα. Αυτές οι θύρες μπορούν να στείλουν περίπου 1-2 μεγαμπίτ ανά δευτερόλεπτο. Η θύρα ECP έχει επίσης το πλεονέκτημα ότι χρησιμοποιεί κανάλια DMA και buffers FIFO, κατά συνέπεια τα στοιχεία μπορούν να μεταφερθούν χωρίς τη χρησιμοποίηση των I/O οδηγιών.

2.1.1 Ιδιότητες παράλληλης θύρας

Ο D-type 25 συνδετήρας είναι ο πιο κοινός συνδετήρας που βρίσκεται στις παράλληλες θύρες των υπολογιστών. Στην Εικόνα 12 ,ο πίνακας μας παρουσιάζει το όνομα, την κατεύθυνση και τον καταχωρητή που ανήκει κάθε pin της παράλληλης θύρας.

Το IEEE 1284 πρότυπο εντούτοις διευκρινίζει 3 διαφορετικούς συνδετήρες για τη χρήση με την παράλληλη θύρα. Ο πρώτος, 1284 type A είναι ο d-type 25 συνδετήρας που βρίσκεται στο πίσω μέρος των περισσότερων υπολογιστών. Ο 2ος είναι ο 1284 type B που είναι ο συνδετήρας Centronics 36 pin που βρίσκεται στους περισσότερους εκτυπωτές.

Ο IEEE 1284 type C εντούτοις, είναι ένας συνδετήρας 36 αγωγών όπως το Centronics, αλλά μικρότερος. Αυτός ο συνδετήρας θεωρείται ότι έχει έναν καλύτερο σύρτη συνδετήρων (ασφαλίζει καλύτερα), καλύτερες ηλεκτρικές ιδιότητες και είναι ευκολότερος να κατασκευαστεί. Περιέχει επίσης δύο περισσότερα pin για σήματα που μπορούν να χρησιμοποιηθούν για να δουν εάν η άλλη συσκευή που

συνδέθηκε, έχει τάση. Οι συνδετήρες type C συστήνονται για νέα σχέδια, έτσι μπορούμε να βρούμε αυτούς τους νέους συνδετήρες στις νέες συσκευές.

Pin No (DB25)	Signal name	Direction	Register - bit	Inverted
1	nStrobe	Out	Control-0	Yes
2	Data0	In/Out	Data-0	No
3	Data1	In/Out	Data-1	No
4	Data2	In/Out	Data-2	No
5	Data3	In/Out	Data-3	No
6	Data4	In/Out	Data-4	No
7	Data5	In/Out	Data-5	No
8	Data6	In/Out	Data-6	No
9	Data7	In/Out	Data-7	No
10	nAck	In	Status-6	No
11	Busy	In	Status-7	Yes
12	Paper-Out	In	Status-5	No
13	Select	In	Status-4	No
14	Linefeed	Out	Control-1	Yes
15	nError	In	Status-3	No
16	nInitialize	Out	Control-2	No
17	nSelect-Printer	Out	Control-3	Yes
18-25	Ground	-	-	-

Εικόνα 12: Πίνακας των pins Παράλληλη θύρα

Ο ανωτέρω πίνακας χρησιμοποιεί το "n" μπροστά από το όνομα σημάτων για να δείξει ότι το σήμα είναι ενεργό στο χαμηλό π.χ. nError. Εάν ο εκτυπωτής έχει εμφανίσει ένα λάθος αυτή η γραμμή είναι χαμηλή. Αυτή η γραμμή είναι κανονικά υψηλή, εάν λειτουργεί σωστά ο εκτυπωτής. Το "Hardware Inverted" σημαίνει ότι το σήμα αναστρέφεται από το υλικό της παράλληλης κάρτας. Ένα τέτοιο παράδειγμα είναι η Busy γραμμή. Εάν + 5v (ή Logic 1) εφαρμοστεί σε αυτό το pin και το Status register read, θα επιστρέψει 0 στο bit 0 του Status Register.

Οι έξοδοι των παράλληλων θυρών είναι κανονικά επίπεδα λογικής TTL. Το επίπεδο τάσης είναι το εύκολο μέρος. Το ρεύμα που μπορεί να βυθίσει και να στείλει ποικίλλει από θύρα σε θύρα. Οι περισσότερες παράλληλες θύρες υλοποιούνται σε ASIC, μπορούν να τραβήξουν και να στείλουν γύρω στα 12mA. Εντούτοις αυτοί είναι μερικοί από τους αριθμούς που λαμβάνονται από τα φύλλα δεδομένων, sink/source 6mA, source 12mA/ sink 20mA, sink 16mA/ source 4mA, sink / source 12mA. Όπως μπορούμε να δούμε ποικίλλουν αρκετά. Η καλύτερη περίπτωση είναι να χρησιμοποιηθεί ένας buffer, έτσι ώστε να απαιτείται ελάχιστο ρεύμα από την θύρα .

2.1.2 Διευθύνσεις θυρών

Η παράλληλη θύρα έχει συνήθως τρεις χρησιμοποιούμενες διευθύνσεις. Αυτές παρατίθενται στην Εικόνα 13, παρακάτω. Η διεύθυνση 3BCh εισήχθη αρχικά χρησιμοποιημένη για τις παράλληλες θύρες στις παλαιότερες κάρτες βίντεο. Αυτή η διεύθυνση εξαφανίστηκε έπειτα για μια στιγμή, όταν αφαιρέθηκαν οι παράλληλες θύρες από τις κάρτες βίντεο. Έχουν επανεμφανιστεί τώρα ως επιλογή για τις παράλληλες θύρες που ενσωματώνονται επάνω στις μητρικές κάρτες, στις οποίες η διαμόρφωσή τους μπορεί να αλλάξει χρησιμοποιώντας το BIOS.

LPT1 ορίζεται κανονικά η διεύθυνση βάσεων 378h, ενώ LPT2 ορίζεται η 278h. Εντούτοις αυτό μπορεί να μην συμβαίνει πάντα όπως εξηγείται αργότερα. Η 378h & 278h χρησιμοποιούνται συνήθως για τις παράλληλες θύρες. Ο δείκτης h δείχνει ότι είναι σε δεκαεξάρικο σύστημα. Αυτές οι διευθύνσεις μπορούν να αλλάξουν από μηχανήμα σε μηχανήμα.

Address	Notes:
3BCh - 3BFh	Used for Parallel Ports which were incorporated in to Video Cards and now, commonly an option for Ports controlled by BIOS. - Doesn't support ECP addresses.
378h - 37Fh	Usual Address For LPT 1
278h - 27Fh	Usual Address For LPT 2

Table 2 Port Addresses

Εικόνα 13: Διευθύνσεις παράλληλων θυρών

Όταν ο υπολογιστής ανοίγεται αρχικά, το BIOS (Basic System Input/Output) θα καθορίσει τον αριθμό θυρών που έχουμε και ορίζει τις ετικέτες των συσκευών σε LPT1, LPT2 & LPT3. Το BIOS εξετάζει αρχικά τη διεύθυνση 3BCh. Εάν μια παράλληλη θύρα βρίσκεται εκεί, ορίζεται ως LPT1, έπειτα ψάχνει στη θέση 378h. Εάν μια παράλληλη κάρτα βρεθεί εκεί παίρνει το αμέσως επόμενο διαθέσιμο όνομα. Αυτό ορίζεται σε LPT1 εάν δε βρεθεί κάρτα στην 3BCh, στη συνέχεια συμβαίνει το ίδιο και για την 278h. Επομένως είναι δυνατό να υπάρχει LPT2 στην 378h και όχι στην αναμενόμενη διεύθυνση 278h. Αυτό που μπορεί να κάνει αυτήν την διαδικασία ακόμη πιο πολύπλοκη, είναι ότι μερικοί κατασκευαστές των παράλληλων καρτών, έχουν jumpers που επιτρέπουν να τεθεί η θύρα σε LPT1, LPT2, LPT3. Στην πλειοψηφία των καρτών η LPT1 είναι η 378h, και LPT2 η 278h,

αλλά μερικοί θα χρησιμοποιήσουν την 3BCh ως LPT1, 378h ως LPT2 και 278h σαν LPT3.

Η ανάθεση των ονομασιών LPT1, LPT2 & LPT3 δεν πρέπει να ανησυχούν αυτούς που επιθυμούν να διασυνδέσουν τις συσκευές στα PC τους. Τις περισσότερες φορές η διεύθυνση βάσης χρησιμοποιείται για να διασυνδέσει τη θύρα, παρά το όνομα LPT1 κ.λπ.. Εντούτοις εάν θέλουμε να βρούμε τη διεύθυνση LPT1 ή οποιοσδήποτε από τις συσκευές, μπορούμε να χρησιμοποιήσουμε έναν πίνακα αντιστοίχισης που παρέχεται από το BIOS(Εικόνα 14). Όταν το BIOS ορίζει τις διευθύνσεις στις συσκευές εκτυπωτών σας, αποθηκεύει τη διεύθυνση στις συγκεκριμένες θέσεις στη μνήμη, έτσι μπορούμε να τους βρούμε.

Start Address	Function
0000:0408	LPT1's Base Address
0000:040A	LPT2's Base Address
0000:040C	LPT3's Base Address
0000:040E	LPT4's Base Address (Note 1)

Table 3 - LPT Addresses in the BIOS Data Area

Εικόνα 14:Πίνακας αντιστοίχισης διευθύνσεων BIOS

Στην εικόνα 14, ο πίνακας παρουσιάζει τη διεύθυνση στην οποία μπορούμε να βρούμε τις διευθύνσεις της θύρας των εκτυπωτών στην περιοχή δεδομένων του BIOS. Κάθε διεύθυνση θα καταλάβει 2 byte.

Καταχωρητές λογισμικού τυποποιημένης παράλληλης θύρας (SSP) 2.1.3

Offset	Name	Read/Write	Bit No.	Properties
Base + 0	Data Port	Write (Note-1)	Bit 7	Data 7 (Pin 9)
			Bit 6	Data 6 (Pin 8)
			Bit 5	Data 5 (Pin 7)
			Bit 4	Data 4 (Pin 6)
			Bit 3	Data 3 (Pin 5)
			Bit 2	Data 2 (Pin 4)
			Bit 1	Data 1 (Pin 3)
			Bit 0	Data 0 (Pin 2)

Table 4 Data Port

Εικόνα 15:Data port παράλληλης θύρας

Η διεύθυνση βάσης, αποκαλούμενη συνήθως Data Port ή Data Register χρησιμοποιείται απλά για την αποστολή δεδομένων μέσω της παράλληλης θύρας (pin 2-9). Αυτός ο καταχωρητής είναι κανονικά μόνο εγγράψιμος (write only). Εάν διαβάζουμε από τη θύρα, θα πάρουμε το τελευταίο byte που στάλθηκε. Ωστόσο εάν η θύρα μας είναι αμφίδρομης επικοινωνίας μπορούμε να λάβουμε δεδομένα σε αυτήν την διεύθυνση.

Base + 1	Status Port	Read Only	Bit 7	Busy
			Bit 6	Ack
			Bit 5	Paper Out
			Bit 4	Select In
			Bit 3	Error
			Bit 2	IRQ (Not)
			Bit 1	Reserved
			Bit 0	Reserved

Table 5 Status Port

Εικόνα 16: Status Port παράλληλης θύρας

Η Status Port (η διεύθυνση βάσης + 1) είναι μια θύρα που μπορεί μόνο να διαβαστεί. Οποιαδήποτε στοιχεία γράφονται σε αυτήν την θύρα θα αγνοηθούν. Η Status Port αποτελείται από 5 γραμμές εισόδου (pin 10,11,12,13 & 15), έναν καταχωρητή IRQ και δύο δεσμευμένα μπιτ. Σημειώστε ότι το bit 7 (busy) είναι μια ενεργός χαμηλή είσοδος. Π.χ. Εάν το bit 7 παρουσιάσει μια λογική 0, αυτό σημαίνει πως στο pin 11 υπάρχει μια τάση +5V. Πιθανόν και στο bit 2 (nIRQ). Εάν αυτό το bit είναι "1" σημαίνει ότι δεν έχει εμφανιστεί διακοπή.

Base + 2	Control Port	Read/Write	Bit 7	Unused
			Bit 6	Unused
			Bit 5	Enable bi-directional Port
			Bit 4	Enable IRQ Via Ack Line
			Bit 3	Select Printer
			Bit 2	Initialize Printer (Reset)
			Bit 1	Auto Linefeed
			Bit 0	Strobe

Table 6 Control Port

Εικόνα 17: Control Port παράλληλης θύρας

Η Control Port (ή διεύθυνση βάσης + 2) προορίζεται μόνο για εγγραφή δεδομένων. Όταν ένας εκτυπωτής συνδέεται με την παράλληλη θύρα, τέσσερα "control" χρησιμοποιούνται. Αυτά είναι το strobe, auto Linefeed, initialize και Select Printer, οι οποίοι είναι όλοι ανάστροφοι εκτός από το initialize.

Ο εκτυπωτής δεν θα έστελνε ένα σήμα για να εκκινήσει τον υπολογιστή, ούτε θα έλεγε στον υπολογιστή να χρησιμοποιήσει το Auto Line Feed. Εντούτοις αυτά τα τέσσερα pins μπορούν επίσης να χρησιμοποιηθούν για είσοδοι. Εάν ο υπολογιστής έχει τοποθετήσει ένα pin υψηλό (π.χ. + 5v) και η συσκευή σας θέλει να την κάνει χαμηλή, εξορισμού θα προκαλέσει βραχυκύκλωμα, προκαλώντας μια σύγκρουση(conflict) σε εκείνο το pin. Επομένως αυτές οι γραμμές είναι είσοδοι "ανοικτού συλλέκτη" (open collector) (ή open drain για τις συσκευές CMOS). Αυτό σημαίνει ότι έχει δύο καταστάσεις. Μια χαμηλή κατάσταση (0v) και μια υψηλής αντίστασης (ανοικτό κύκλωμα).

Κανονικά η θύρα του εκτυπωτή θα έχει εσωτερικές pull-up αντιστάσεις, αλλά όπως θα αναμένατε, όχι όλες. Μερικές μπορεί απλά να έχουν εξόδους open-collector, ενώ άλλες μπορούν ακόμη και να έχουν απλές εξόδους. Προκειμένου να γίνει η εργασία των συσκευών σας σωστά για όσο το δυνατόν περισσότερες θύρες εκτυπωτών, μπορούμε να χρησιμοποιήσουμε μια εξωτερική αντίσταση.

Μια εξωτερική αντίσταση 4.7k μπορεί να χρησιμοποιηθεί για να τραβήξει το pin υψηλά. Δεν θα πρέπει να χρησιμοποιήσουμε χαμηλότερη τιμή, ακριβώς για την περίπτωση που έχουμε ήδη μια εσωτερική pull-up αντίσταση, δεδομένου ότι η εξωτερική αντίσταση θα ενεργούσε παράλληλα δίνοντας έτσι, μια χαμηλότερη συνολική αντίσταση. Όταν βρίσκετε σε κατάσταση υψηλής αντίστασης, το pin στην παράλληλη θύρα είναι σε υψηλό (+ 5v). Σε αυτήν την κατάσταση αν η εξωτερική συσκευή τραβήξει το pin χαμηλά, τότε η τιμή που θα διαβαστεί θα είναι 0 . Με αυτόν τον τρόπο τα 4 pin του control port μπορεί να χρησιμοποιηθούν για την αμφίδρομη μεταφορά δεδομένων. Εντούτοις το control port πρέπει να τεθεί σε xxxx0100 για να είναι σε θέση να διαβάσει τα δεδομένα, το οποίο σημαίνει ότι όλα τα pins θα είναι + 5v στη θύρα έτσι ώστε να μπορούμε να το τραβήξουμε χαμηλά στη γείωση gnd (λογική 0).

Τα pin 4 & 5 είναι εσωτερικοί έλεγχοι. Το pin 4 θα επιτρέψει το IRQ και το pin 5 θα επιτρέψει την αμφίδρομη επικοινωνία των θυρών. ότι μπορούμε δηλαδή να εισάγουμε 8 μπιτ χρησιμοποιώντας τα DATA 0-7. Αυτός ο τρόπος είναι μόνο δυνατός εάν η κάρτα μας τον υποστηρίζει. Τα pin 6 & 7 είναι δεσμευμένα. Οποιοσδήποτε εγγραφή σε αυτά τα δύο μπιτ θα αγνοηθεί.

2.1.4 Αμφίδρομες θύρες

Οι αμφίδρομες θύρες χρησιμοποιούν το bit 5 της Control Port συνδεδεμένη με το OE των 374, έτσι ώστε οδηγοί εξόδους μπορούν να κλείσουν. Με αυτόν τον τρόπο μπορούμε να διαβάσουμε τα δεδομένα που βρίσκονται στα Data Pins της θύρας, χωρίς την ύπαρξη συγκρούσεων(conflict) και υπερβολικών ρευμάτων.

Το bit 5 της Control Port επιτρέπει ή θέτει εκτός λειτουργίας την αμφίδρομη λειτουργία της παράλληλης θύρας. Αυτό είναι μόνο διαθέσιμο στις αληθινές

αμφίδρομες θύρες. Όταν αυτό το bit τίθεται σε ένα, τα pin 2 έως 9 πηγαίνουν σε κατάσταση υψηλής αντίστασης. Σε αυτήν την κατάσταση μπορούμε να εισάγουμε τα δεδομένα σε αυτές τις γραμμές και να τα ανακτήσουμε από το DataPort (διεύθυνση βάσεων). Οποιαδήποτε δεδομένα γράφονται στη Data Port θα αποθηκευτούν αλλά δεν θα είναι διαθέσιμα στα Data Pin. Για να κλείσουμε την αμφίδρομη επικοινωνία, κάνουμε το pin 5 της Control Port "0".

Ωστόσο δεν συμπεριφέρονται όλες οι θύρες με τον ίδιο τρόπο. Άλλες θύρες μπορεί να απαιτήσουν το bit 6 του Control Port να γίνει "1" για να επιτρέψουν τον αμφίδρομο τρόπο και το pin 5 για να απενεργοποιήσουν τον αμφίδρομο τρόπο, διαφορετικοί κατασκευαστές εφαρμόζουν τις αμφίδρομες θύρες τους με διαφορετικούς τρόπους. Εάν επιθυμούμε να χρησιμοποιήσουμε την αμφίδρομο θύρα σας για είσοδο δεδομένων, εξετάσουμε με έναν ελεγκτή λογικής ή το πολύμετρο για να σιγουρευτούμε πρώτα ότι είναι σε κατάσταση εισόδου.

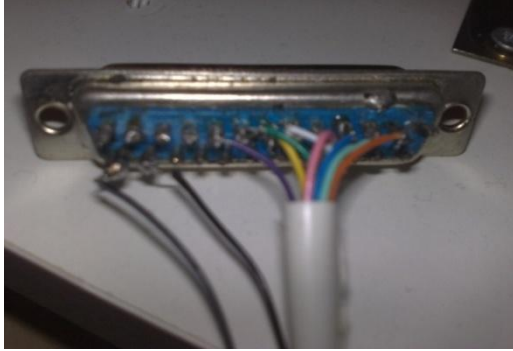
2.2 Σύνδεση αισθητηρίων

Στην κατασκευή μας χρησιμοποιήσαμε έναν παθητικό υπέρυθρο αισθητήρα(PIR) και επτά μαγνητικές επαφές που προμηθευτήκαμε από κατάστημα με ηλεκτρικές συσκευές και κόστιζαν 20€ ο αισθητήρας PIR και από 1,5€ η κάθε μαγνητική επαφή. Επίσης πήραμε ένα πολύμετρο που κόστισε 9€ και ένα κολλητήρι και καλάνι που κόστισαν από 5€ το καθένα (Εικόνα 18).

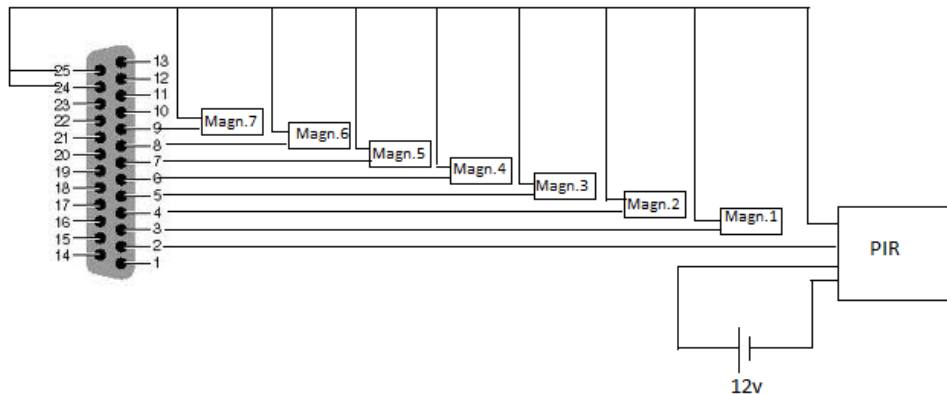


Εικόνα 18:Πολύμετρο, καλάνι και κολλητήρι

Από ένα καλώδιο παράλληλης θύρας αφού το ανοίξαμε πήραμε τον αρσενικό συνδετήρα D-type 25 και τα καλωδιάκια που περιείχε. Στην συνέχεια με βάση της ιδιότητες της παράλληλης θύρας που προαναφέραμε θα χρησιμοποιήσουμε μόνο τα pins των δεδομένων και μερικά των γειώσεων και θα υλοποιήσουμε το κύκλωμα τις Εικόνας 20. Έτσι με το κολλητήρι και το καλάι κολλήσαμε τα καλωδιάκια στα αντίστοιχα Data pins του αρσενικού συνδετήρα και σε δυο pins γείωσης (Εικόνα 19).

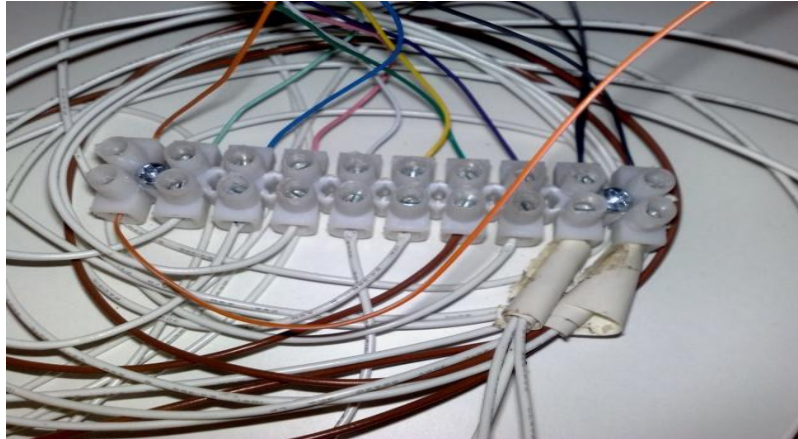


Εικόνα 19:Αρσενικός συνδετήρας παράλληλης



Εικόνα 20:Κύκλωμα της κατασκευής

Στην συνέχεια συνδέσαμε τα άλλα άκρα από αυτά τα καλώδια τα συνδέσαμε, κατά αύξουσα σειρά αυτά που αντιστοιχούν στα δεδομένα και έπειτα τα δυο των γειώσεων για να ξέρουμε που αντιστοιχεί το καθένα, στη μία πλευρά από ένα κλέμα 10 θέσεων. Στην άλλη του πλευρά συνδέσαμε εκεί που αντιστοιχούν τα δεδομένα ένα καλώδιο, το οποίο θα συνδέσουμε τον αισθητήρα PIR, και από ένα καλώδιο κάθε μαγνητικής επαφής και εκεί που αντιστοιχούν οι γειώσεις συνδέσαμε τα δεύτερα καλώδια από τις μαγνητικές επαφές καθώς και ένα ακόμη καλώδιο που θα συνδέσουμε επίσης στον αισθητήρα PIR(Εικόνα 21).



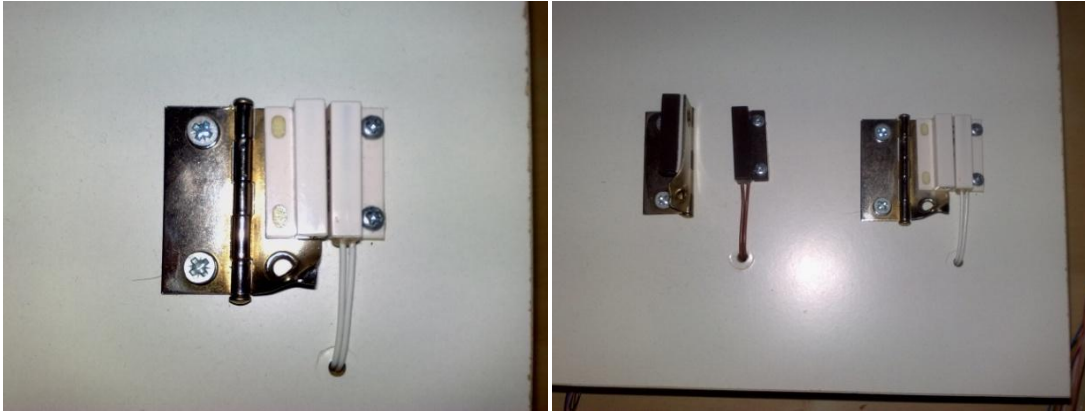
Εικόνα 21:Σύνδεση καλωδίων και αισθητήριων με κλέμα.

Τα δύο καλώδια που είπαμε ότι θα είναι για τον αισθητήρα PIR αφού ανοίξαμε τον αισθητήρα τα συνδέσαμε στις υποδοχές “ALARM” που έχει η πλακέτα του αισθητήρα(αριστερά στην Εικόνα 22) ενώ στις υποδοχές “+/-12v” συνδέσαμε δυο καλώδια που ενώσαμε με του αντίστοιχους πόλους μια μικρής δωδεκάβολτης μπαταρίας (δεξιά στην Εικόνα 22).



Εικόνα 22:Συνδέσεις στον αισθητήρα PIR

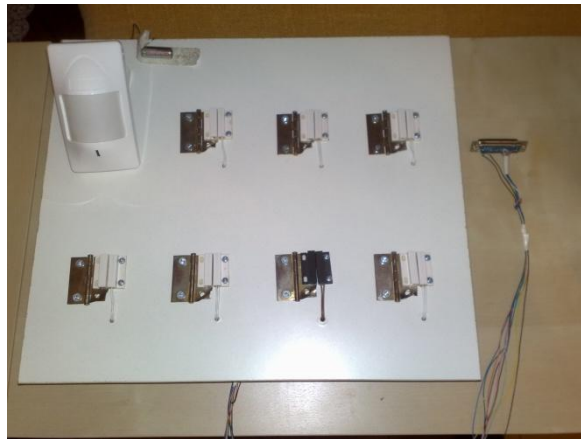
Σε έναν κοντραπλακέ (37cm X 31cm), αφού το τρυπήσαμε με ένα τρυπάνι και περάσαμε τα καλώδια, βιδώσαμε στο κάτω μέρος δυο πηγάκια (3cm) και το κλέμα. Στο πάνω μέρος του βιδώσαμε τον αισθητήρα PIR και τα επτά σταθερά μέρη από τις μαγνητικές επαφές και δίπλα σε αυτές βιδώσαμε επτά μεντεσέδες μικρούς που πάνω τους είχαμε κολλήσει τα κινούμενα μέρη των μαγνητικών επαφών για να προσομοιώσουμε το άνοιγμα και το κλείσιμο της πόρτας ή των παραθύρων (Εικόνα 23).



Εικόνα 23: Μαγνητικές επαφές

ΕΠΙΛΟΓΟΣ

Μετά από πολλές και συνέχεις μετρήσεις και μερικούς χαλασμένους αισθητήρες, διότι αρχικά είχαμε επιλέξει αισθητήρες με μπαταρίες και είχαν χαμηλότερη τάση στο κύκλωμά τους, ολοκληρώσαμε την κατασκευή του υλικού μας, που έχει την μορφή πάνελ της Εικόνας 24. Αφού ολοκληρώσαμε την περιγραφή της κατασκευής του υλικού μας θα προχωρήσουμε στο επόμενο κεφάλαιο στην αναλυτική περιγραφή του προγράμματος μας και των μεθόδων που συγγράψαμε, καθώς και της γλώσσας και τους λόγους που την επιλέξαμε.



Εικόνα 24: Το πάνελ με τους αισθητήρες

ΚΕΦΑΛΑΙΟ 3

«Κατασκευή του Λογισμικού(“Software”)

ΕΙΣΑΓΩΓΗ

Σε αυτό το κεφάλαιο θα περιγράψουμε το πώς κατασκευάστηκε το λογισμικό μας καθώς και ποια γλώσσα προγραμματισμού χρησιμοποιήσαμε. Η γλώσσα η οποία επιλέξαμε είναι η “ Visual Basic .NET” και χρησιμοποιήσαμε την πλατφόρμα του “Microsoft Visual Studio 2008” για την κατασκευή του προγράμματος. Η επιλογή αυτή έγινε διότι το “Microsoft Visual Studio 2008” μας διευκολύνει αρκετά στην κατασκευή της εικόνας της διεπαφής μας και επίσης θέλαμε να διευρύνουμε τις γνώσεις με μια νέα γλώσσα προγραμματισμού που δεν είχαμε ασχοληθεί στο παρελθόν.

Η “Visual Basic .NET” (VB.NET), που επιλέξαμε είναι μια αντικειμενοστραφής γλώσσα προγραμματισμού που μπορεί να θεωρηθεί ως εξέλιξη του της κλασικής “Visual Basic” (VB), η οποία υλοποιείται στο .NET Framework.

Η “Visual Basic” (VB), είναι γλώσσα προγραμματισμού τρίτης γενιάς, οδηγούμενη από συμβάντα και έχει ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) από τη Microsoft για το μοντέλο προγραμματισμού COM. Η VB θεωρείται επίσης μία σχετικά εύκολη γλώσσα προγραμματισμού στην εκμάθηση και τη χρησιμοποίηση, λόγω των χαρακτηριστικών της, καθώς έχει Γραφικό Περιβάλλον Χρήστη και συγγένεια με την γλώσσα προγραμματισμού BASIC (Beginner’s All Purpose Symbolic Instruction Code). Η VB επιτρέπει την ταχεία ανάπτυξη εφαρμογών (RAD) με Γραφικό Περιβάλλον Χρήστη (GUI), πρόσβαση σε βάσεις δεδομένων χρησιμοποιώντας αντικείμενα (Data Access Objects, Remote Data Objects, ή ActiveX Data Objects), και τη δημιουργία στοιχείων ελέγχου ActiveX και αντικειμένων.

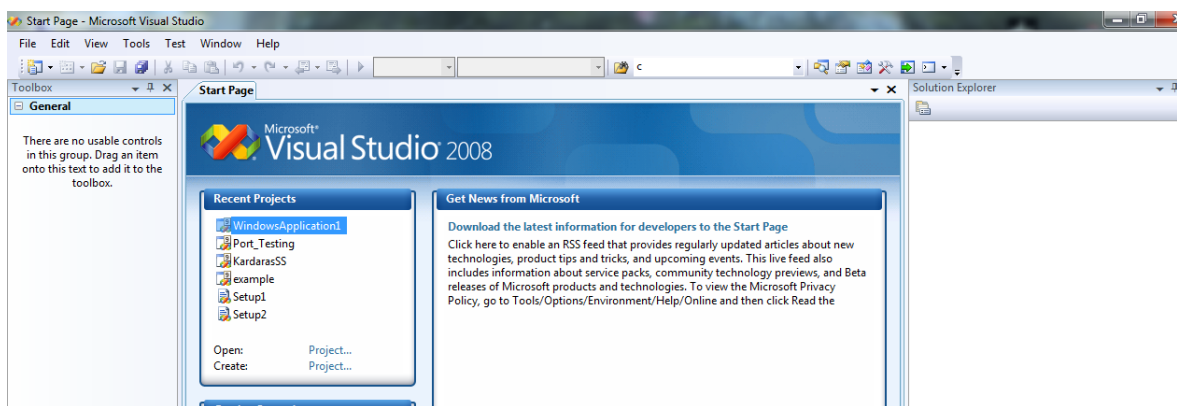
Το .NET Framework (προφέρεται dot net) είναι ένα πλαίσιο λογισμικού για τα Windows λειτουργικά συστήματα. Περιλαμβάνει μια μεγάλη βιβλιοθήκη , και υποστηρίζει πολλές γλώσσες προγραμματισμού που επιτρέπει τη διαλειτουργικότητα τους. Η βιβλιοθήκη του .NET είναι διαθέσιμη σε όλες τις γλώσσες προγραμματισμού που το υποστηρίζουν. Η Base Class Library (BCL) του .NET παρέχει διεπαφή χρήστη , πρόσβαση σε δεδομένα , σύνδεσης με βάσεις δεδομένων , κρυπτογραφία, web εφαρμογή ανάπτυξη, την αριθμητική αλγόριθμους και επικοινωνιών δικτύου. Προγράμματα που γράφτηκαν για το .NET Framework εκτελούνται σε ένα λογισμικό περιβάλλον (σε αντιπαραβολή του hardware), γνωστό ως η Common Language Runtime (CLR). Το CLR είναι

μια εφαρμογή εικονικής μηχανής έτσι ώστε οι προγραμματιστές δεν χρειάζεται να εξετάσει τις δυνατότητες του συγκεκριμένου επεξεργαστή που θα εκτελέσει το πρόγραμμα. Το CLR παρέχει επίσης και άλλες σημαντικές υπηρεσίες, όπως η ασφάλεια, διαχείριση μνήμης, και το χειρισμό εξαίρεση. Η BCL και η CLR αποτελούν μαζί το .NET Framework.

Η Microsoft παρέχει σήμερα δύο σημαντικές εφαρμογές της VB.NET: το Microsoft Visual Studio, το οποίο είναι εμπορικό λογισμικό και το Microsoft Visual Studio Express, το οποίο είναι δωρεάν και μπορούμε να κατεβάσουμε από την ιστοσελίδα της (<http://www.microsoft.com/express/>).

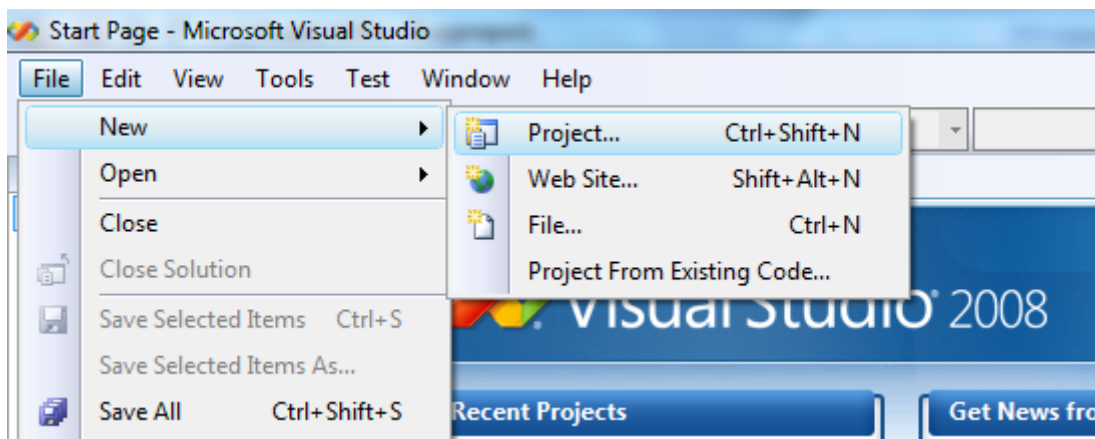
3.1 Περιγραφή κατασκευής της εφαρμογής

Ξεκινώντας το Microsoft Visual Studio 2008, ανοίγει η αρχική του σελίδα, όπου βλέπουμε ότι υπάρχουν επιλογές για παλαιότερα πρότζεκτ μας.



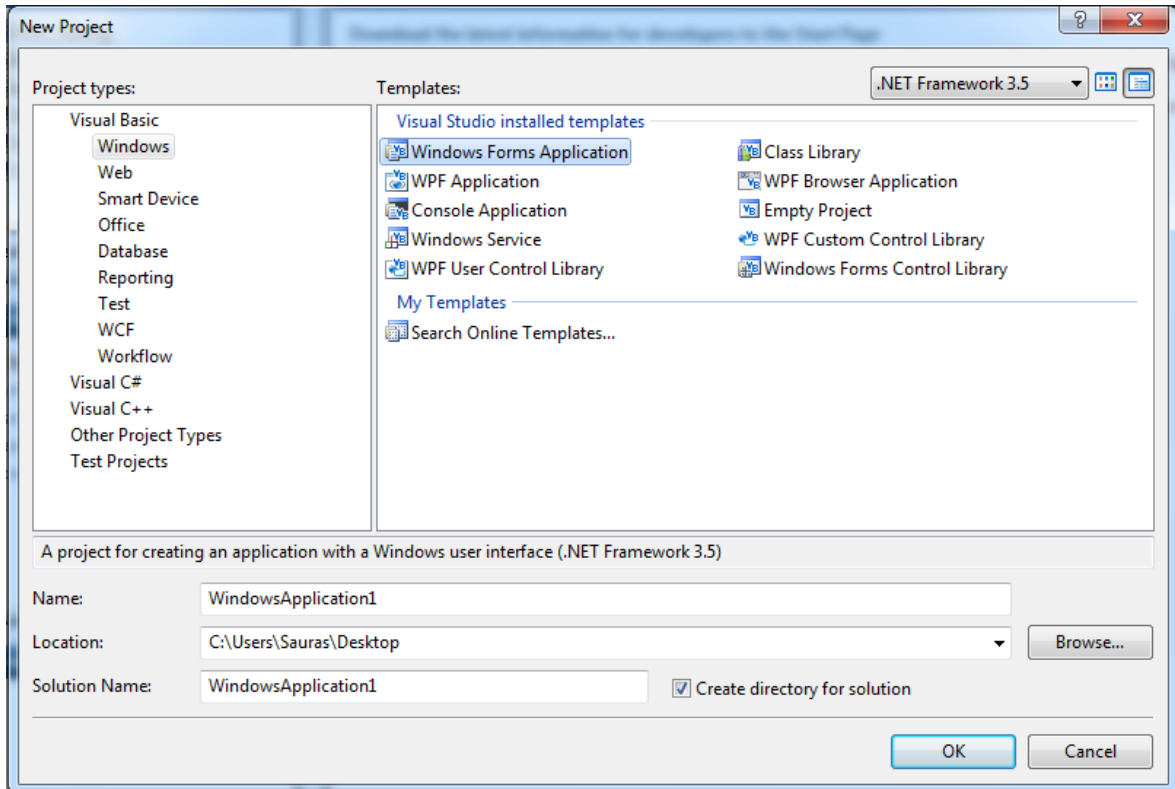
Εικόνα 25: Αρχική σελίδα του Microsoft Visual Studio 2008

Για να δημιουργήσουμε ένα νέο πρότζεκτ, επιλέγουμε “File->New->Project..”.



Εικόνα 26: Δημιουργία νέου πρότζεκτ

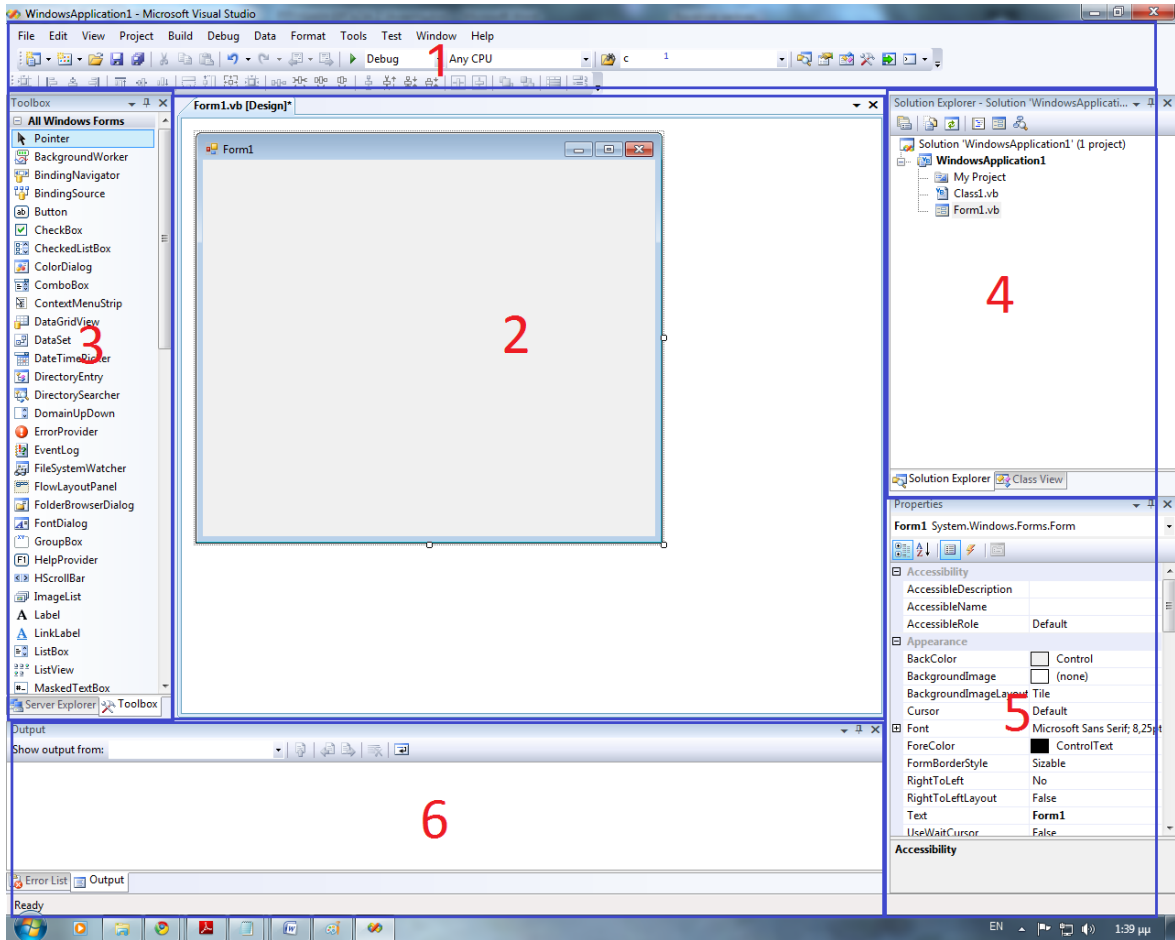
Έπειτα μας εμφανίζεται το παράθυρο “New Project” όπου επιλέγουμε το είδος του πρότζεκτ που θέλουμε να δημιουργήσουμε, την γλώσσα προγραμματισμού που θέλουμε καθώς και το πώς θα το ονομάσουμε.



Εικόνα 27: Παράθυρο επιλογών νέου πρότζεκτ

Όπως βλέπουμε στην περιοχή που ονομάζεται, Project types μπορούμε να διαλέξουμε την γλώσσα προγραμματισμού που θέλουμε να δουλέψουμε πχ. Visual Basic, Visual C#, Visual C++ . Επιλέγοντας μια από αυτές, στην συγκεκριμένη περίπτωση εμείς διαλέξαμε την Visual Basic, μας δίνεται η δυνατότητα να διαλέξουμε αν θα δημιουργήσουμε μια εφαρμογή για Windows που θα τρέχει σε σταθερό υπολογιστή ή μια εφαρμογή που θα τρέχει σε φορητές συσκευές(Smart Device) , όπως Pocket PC και smart phone. Στο δίπλα περιθώριο (Templates) διαλέγουμε πιο συγκεκριμένα την εφαρμογή που θέλουμε να δημιουργήσουμε, αν θα είναι ένα κενό Project ,μια Class Library κ.ά.

Συμπληρώνοντας το όνομα και την θέση που θέλουμε να αποθηκεύσουμε το Project, ανοίγει το βασικό περιβάλλον του Microsoft Visual Studio 2008. Στην συγκεκριμένη περίπτωση εμείς έχουμε διαλέξει μια Windows Form Application με την γλώσσα Visual Basic.



Εικόνα 28: Παράθυρο κατασκευής νέου πρότζεκτ

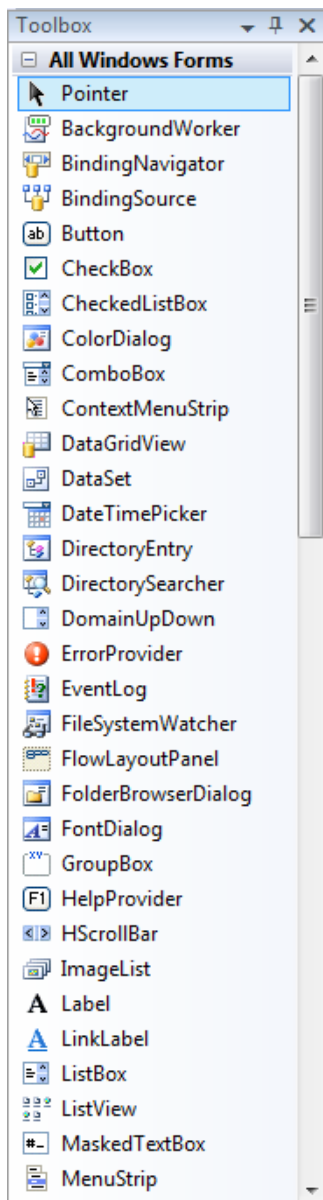
Όπως μπορούμε εύκολα να διακρίνουμε, και έχουμε σημειώσει στην εικόνα 28, υπάρχουν έξι πλαίσια που χωρίζεται το περιβάλλον αυτό.

Ξεκινώντας από το πλαίσιο 1 βλέπουμε την μπάρα μενού, με τις επιλογές file, open, και γενικά όλα τα εργαλεία που υποστηρίζει η πλατφόρμα. Από κάτω βρίσκεται η μπάρα εργαλείων, που αποτελείται από κουμπιά εργαλείων, όπως προσθήκης νέων αντικειμένων, δημιουργίας νέου πρότζεκτ, το Debug κ.ά. Το Debug κάνει Compile το πρόγραμμα που έχουμε δημιουργήσει και το τρέχει ώστε να δει ο προγραμματιστής αν είναι σωστή η εφαρμογή του.



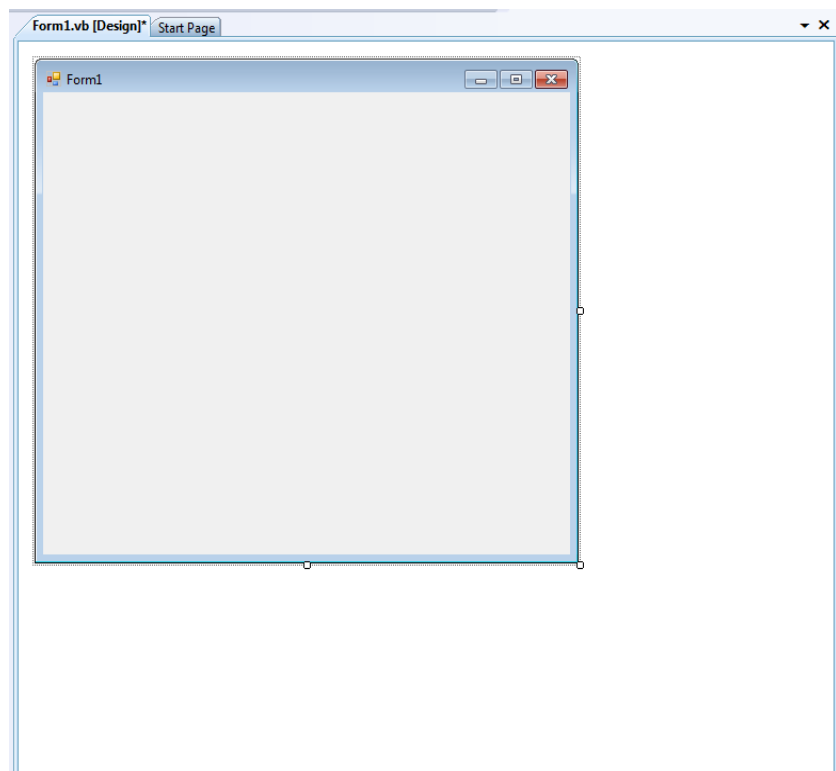
Εικόνα 29: Μπάρα μενού και εργαλείων

Στην συνέχεια βλέπουμε αριστερά το πλαίσιο 3 το οποίο περιέχει όλα τα εργαλεία για την κατασκευή της φόρμας μας, δηλαδή του παραθύρου μας(Εικόνα).



Εικόνα 30: Εργαλειοθήκη του Microsoft Visual Studio 2008

Επιλέγοντας ένα από τα εργαλεία που θέλουμε όπως ένα button, ή ένα checkbox και ότι μπορεί να υπάρχει σε μια Windows form, με “drag and drop” μπορούμε να το προσθέσουμε στο δικό μας Form. Συνεχίζοντας στο πλαίσιο 2, το Microsoft Visual Studio έχει δημιουργήσει μια κενή φόρμα όπου σε αυτήν εμείς θα προσθέσουμε τα button, τα checkboxes και γενικώς όποιο εργαλείο θέλουμε από το Toolbox (Εικόνα 30). Ο σκοπός μας είναι να δημιουργήσουμε την δική μας φόρμα ανάλογα με τις ανάγκες μας. (Εικόνα 31)

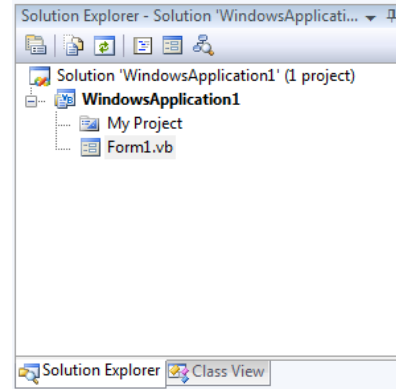


Εικόνα 31: Πλαίσιο νέας φόρμας

Μόλις τοποθετήσουμε ένα εργαλείο μέσα στην φόρμα, το Microsoft Visual Studio μας δίνει την δυνατότητα πατώντας διπλό κλικ πάνω σε αυτό, να βλέπουμε τον κώδικα που βρίσκεται από πίσω του, έτσι ώστε να έχουμε το έλεγχο και να

προγραμματίζουμε το κάθε ένα εργαλείο ξεχωριστά. Πατώντας δυο φορές πάνω στην φόρμα, εμφανίζεται όλος ο κώδικας αυτής.

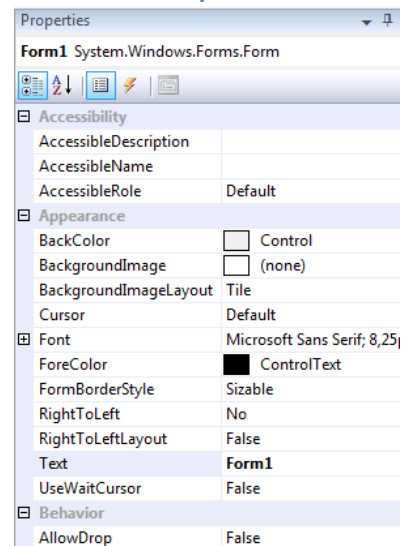
Δημιουργώντας ένα καινούριο πρότζεκτ Microsoft Visual Studio δημιουργεί ένα δέντρο με τις βιβλιοθήκες, τις κλάσεις και τις ιδιότητες του, πλαίσιο 4. Με λίγα λόγια το Solution Explorer (Εικόνα 32) μας διευκολύνει στην αναζήτηση του κώδικα, γιατί ταχτοποιεί το πρότζεκτ ανά κλάση.



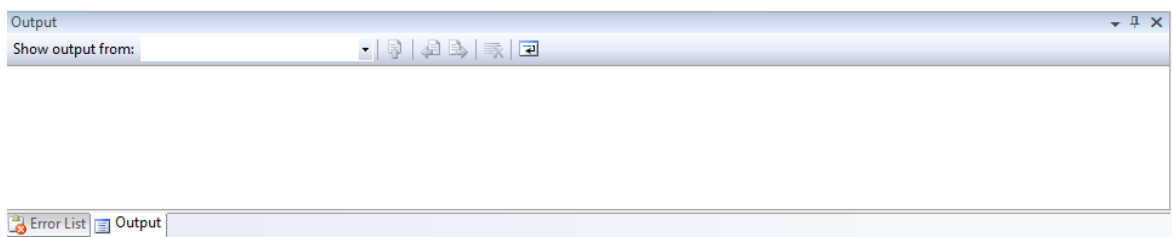
Εικόνα 32: Πλαίσιο Solution Explorer

Στο πλαίσιο 5 βρίσκεται οι επιλογές των ιδιοτήτων "Properties" (Εικόνα 33) όπου εκεί ανάλογα με το αντικείμενο που έχουμε επιλέξει με το ποντίκι, μπορούμε να ρυθμίσουμε τις ιδιότητές του, το χρώμα το όνομα και πολλά άλλα ανάλογα με το είδος του αντικειμένου.

Τέλος βλέπουμε το πλαίσιο 6 όπου εμφανίζονται τυχόν λάθη ή προειδοποιήσεις όταν κάνουμε Compile το πρόγραμμα μας (Εικόνα 34).



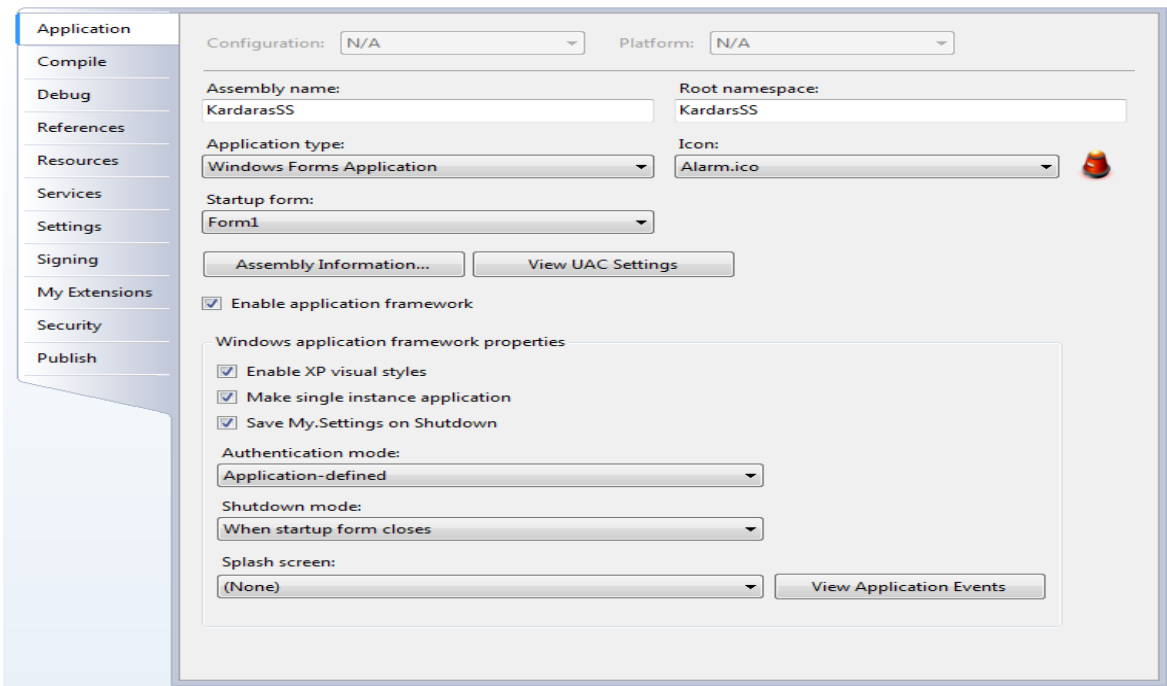
Εικόνα 33: Πλαίσιο ιδιοτήτων



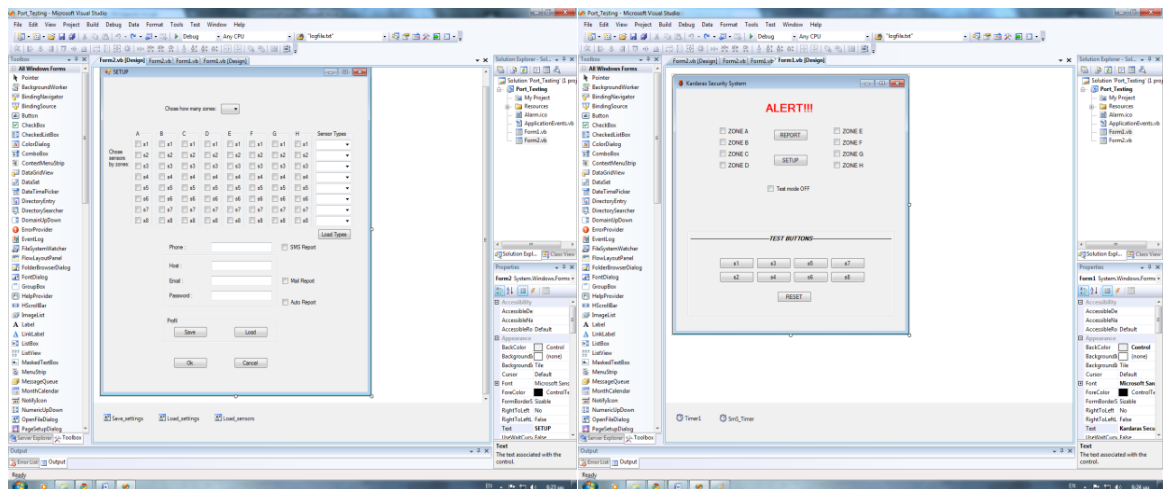
Εικόνα 34: Πλαίσιο Output

Η εφαρμογή μας αποτελείται από δυο φόρμες, την κυρίως φόρμα(form1) και την φόρμα όπου θα γίνονται οι ρυθμίσεις(form2). Έτσι αφού δημιουργήσαμε ένα νέο

πρότζεκτ σχεδιάσαμε τις φόρμες με την εργαλειοθήκη του, όπως τις θέλαμε.(Εικόνα 36). Επιπλέον πατώντας δυο φορές στο Solution Explorer το όνομα του πρότζεκτ μας, ανοίγει μια καρτέλα με το όνομά του με επιλογές για αυτό. Στην συνέχεια εισάγαμε στην επιλογή “resources”, το αρχείο attention.wav το οποίο θα αναπαράγουμε όταν γίνεται κάποια παραβίαση και επιλέξαμε για εικονίδιο της εφαρμογής το alarm.ico στην επιλογή “Application”, καθώς κάναμε και διάφορες άλλες επιλογές όπως το όνομα του εκτελέσιμου και ποια φόρμα θα εμφανίζεται κατά την εκκίνηση του προγράμματος μας κ.ά.(Εικόνα 35). Παράλληλα με τον σχεδιασμό δημιουργήσαμε και τον κώδικα της κάθε φόρμας.



Εικόνα 35: Παράθυρο ρυθμίσεων του πρότζεκτ



Εικόνα 36:Φόρμα ρυθμίσεων (form2

3.1.1 Περιγραφή κώδικα φόρμας ρυθμίσεων(form2)

Έτσι στην αρχή του κώδικα της φόρμας των ρυθμίσεων δηλώσαμε τις μεταβλητές που χρησιμοποιήσαμε:

```
Public c(8) As Short
Private se(8, 8) As CheckBox
Public tse(8) As ComboBox
Public nu_z(8) As String
```

Ο c(8) είναι ένας μονοδιάστατος πίνακας από αριθμούς short που μας δείχνουν πόσους αισθητήρες έχουμε επιλέξει σε κάθε ζώνη. Ο se(8, 8) είναι ένας διδιάστατος πίνακας από checkboxes όπου θα περάσουμε τα checkboxes που αντιστοιχούν στις επιλογές των αισθητήρων κάθε ζώνης για να μπορούμε να τα χειριζόμαστε ομαδικά. Ο tse(8) είναι ένας μονοδιάστατος πίνακας από comboboxes όπου θα περάσουμε τα comboboxes που αντιστοιχούν στις επιλογές των τύπων των αισθητήρων για να μπορούμε να τα χειριζόμαστε ομαδικά, επίσης. Τέλος nu_z(8) είναι ένας μονοδιάστατος πίνακας από string όπου θα περιέχει τα ονόματα των ζωνών, για παράδειγμα σε εμάς «A-Z».

Στην συνέχεια υπάρχει η sub μέθοδος se_name() όπου “γεμίζουμε” τους παραπάνω πίνακες, εκτός από τον c(8), με τα αντίστοιχα checkboxes, comboboxes και ονόματα των ζωνών. Ακολουθεί η sub μέθοδος CB_SelectedIndexChanged() η οποία σε κάθε αλλαγή της επιλογής μας στον αριθμό των ζωνών ενεργοποιεί τις αντίστοιχες δυνατές επιλογές.

Συνεχίζοντας υπάρχει η sub μέθοδος Checked_changed(), η οποία όταν επιλέξουμε κάποιο checkbox που αντιστοιχεί σε αισθητήριο σε κάποια ζώνη, απενεργοποιεί την επιλογή του αισθητήρα από τις άλλες ζώνες, ενεργοποιεί το combobox των τύπων των αισθητήρων, δηλώνει σε ποια ζώνη ανήκει κάθε αισθητήρας (Form1.n()) και υπολογίζει πόσους αισθητήρες έχουμε επιλέξει για κάθε ενεργή ζώνη (c()). Αντίστροφα όταν “ξεκλικάρουμε” κάποιο checkbox που αντιστοιχεί σε αισθητήριο σε κάποια ζώνη ακολουθούν οι ανάποδες διαδικασίες από αυτές που προείπαμε.

```
Private Sub Checked_changed(ByVal sender As System.Object, ByVal e As
System.EventArgs)
    If sender.checked = True Then
        For j = 1 To 8
            For i = 1 To 8
                If se(i, j).Checked = True And se(i, j) Is sender Then
                    c(j) = c(j) + 1
                    Form1.n(i) = j
                    tse(i).Enabled = True
                For z = 1 To 8
                    If se(i, z) Is sender Then
                        se(i, z).Enabled = True
                    Else
```

```

                se(i, z).Enabled = False
            End If
        Next z
    End If
Next i
Next j
Else
    For j = 1 To 8
        For i = 1 To 8
            If se(i, j) Is sender Then
                Form1.n(i) = 0
                c(j) = c(j) - 1
                tse(i).Enabled = False
                For z = 1 To 8
                    se(i, z).Enabled = True
                Next z
            End If
        Next i
    Next j
End If
End Sub

```

Μετά έχουμε την sub μέθοδο L_types_Click() που ενεργοποιείται όταν πατηθεί το κουμπί “Load Types” και εμφανίζει ένα παραθύρο διαλόγου για να επιλέξουμε το αρχείο “.txt” όπου έχουμε γράψει τα είδη των αισθητήρων και πατώντας το κουμπί “Ok” αυτού του παραθύρου διαλόγου ενεργοποιείται η sub μέθοδος Load_sensors_FileOk() η οποία διαβάζει γραμμή-γραμμή το αρχείο που επιλέξαμε και προσθέτει τις γραμμές ως επιλογές ειδών αισθητήρων στα comboboxes των αισθητήρων.

```

Private Sub L_types_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles L_types.Click
    Load_sensors.ShowDialog()
End Sub

Private Sub Load_sensors_FileOk(ByVal sender As System.Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles Load_sensors.FileOk
    Dim rdr As System.IO.StreamReader
    For i = 1 To 8
        tse(i).Items.Clear()
        rdr =
My.Computer.FileSystem.OpenTextFileReader(Load_sensors.FileName)
        While Not rdr.EndOfStream
            Dim line As String = rdr.ReadLine()
            tse(i).Items.Add(line)
        End While
        rdr.Close()
    Next i
End Sub

```

Στην συνέχεια έχουμε την sub μέθοδο sms_report_CheckedChanged(), που ενεργοποιείται όταν μεταβληθεί η επιλογή του checkbox “SMS Report” και ενεργοποιεί ή απενεργοποιεί το πεδίο του τηλεφώνου, και την sub μέθοδο Mail_report_CheckedChanged(), που ενεργοποιείται όταν μεταβληθεί η επιλογή

του checkbox “Mail Report” και αντίστοιχα ενεργοποιεί ή απενεργοποιεί τα πεδία “Host”, “Email” και “Password”.

Έπειτα δημιουργήσαμε δυο function μεθόδους τις encryptString() και decryptString() που μας επιστρέφουν ένα κρυπτογραφημένο ή αποκρυπτογραφημένο String αντίστοιχα. Η μέθοδος encryptString() παίρνει ως όρισμα ένα String και χρησιμοποιώντας ένα κλειδί που θα πρέπει να είναι κρυφό (στο παράδειγμα μας το “encoderkey”) το κρυπτογραφεί Byte-Byte και επιστρέφει το κρυπτογραφημένο String. Η μέθοδος τώρα decryptString() κάνει την ανάποδη διαδικασία, δηλαδή παίρνει ως όρισμα ένα κρυπτογραφημένο String και χρησιμοποιώντας το ίδιο κλειδί που χρησιμοποιήθηκε κατά την κρυπτογράφηση (αν το κλειδί δεν είναι ίδιο η σωστή αποκρυπτογράφηση δεν είναι δυνατή), αποκρυπτογραφεί Byte-Byte και επιστρέφει το αποκρυπτογραφημένο αυτό String. Οι μέθοδοι αυτή θα μας χρησιμέψουν οι μέθοδοι αυτές στη κωδικοποιημένη αποθήκευση του κωδικού του mail μας και στην ανάκτησή του στην συνέχεια.

```
Dim Hash As New Security.Cryptography.MD5CryptoServiceProvider
Dim Des As New Security.Cryptography.TripleDESCryptoServiceProvider

Private Function encryptString(ByVal s As String) As String
    Des.Key =
Hash.ComputeHash(System.Text.Encoding.ASCII.GetBytes("encoderkey"))
    Des.Mode = Security.Cryptography.CipherMode.ECB
    Dim DESEncrypter As System.Security.Cryptography.ICryptoTransform =
Des.CreateEncryptor
    Dim ByteString() As Byte = System.Text.Encoding.ASCII.GetBytes(s)
    Dim enString As String =
Convert.ToBase64String(DESEncrypter.TransformFinalBlock(ByteString, 0,
ByteString.Length))
    Return enString
End Function

Private Function decryptString(ByVal st As String) As String
    Des.Key =
Hash.ComputeHash(System.Text.Encoding.ASCII.GetBytes("encoderkey"))
    Des.Mode = Security.Cryptography.CipherMode.ECB
    Dim DESDecrypter As System.Security.Cryptography.ICryptoTransform =
Des.CreateDecryptor
    Dim ByteString() As Byte = Convert.FromBase64String(st)
    Dim deString As String =
System.Text.Encoding.ASCII.GetString(DESDecrypter.TransformFinalBlock(ByteString,
0, ByteString.Length))
    Return deString
End Function
```

Ακολουθούν οι sub μέθοδοι Save_Click() που ενεργοποιείται όταν πατηθεί το κουμπί “Save” και ανοίγει ένα παράθυρο διαλόγου για να αποθηκεύσουμε τις ρυθμίσεις που έχουμε επιλέξει ,και Save_settings_FileOk() που ενεργοποιείται όταν πατηθεί το κουμπί “Ok” του παραθύρου διαλόγου που έχει ανοίξει η Save_Click() και γράφει με σειρά γραμμή-γραμμή, σε ένα αρχείο “.txt” που δημιουργήσε ή επιλέξαμε, τις επιλογές που έχουμε κανεί. Στην αρχή γράφει μία γραμμή τον αριθμό του αντικειμένου που έχουμε επιλέξει στο combobox με τον

αριθμό των ζωνών. Μετά γράφει πόσους αισθητήρες έχει η κάθε ζώνη και τον δείκτη που μας δείχνει σε ποια ζώνη ανοίκει ο κάθε αισθητήρας. Έπειτα γράφει το path του αρχείου των ειδών των αισθητήρων που έχουμε φορτώσει. Στην συνέχεια γραμμή-γραμμή για τα αισθητήρια αν είναι επιλεγμένα ή όχι γράφει 1 ή 0 αντίστοιχα και τον τύπο του αισθητήρα που έχουμε επιλέξει από τα αντίστοιχα comboboxes. Στην συνέχεια γράφει 1 ή 0 αν είναι επιλεγμένα ή όχι τα checkboxes των sms_report και mail_report και ενδιάμεσα το κείμενο του textbox “Phone”, και έπειτα τα κείμενα των textboxes “Host” και “Email” και κωδικοποιημένο το κείμενο του textbox του “Password”. Τέλος γράφει 1 ή 0 αν είναι επιλεγμένα ή όχι το checkbox Auto_report.

```

Private Sub Save_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Save.Click
    Save_settings.ShowDialog()
End Sub
Private Sub Save_settings_FileOk(ByVal sender As System.Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles Save_settings.FileOk
    Dim sfile As System.IO.StreamWriter
    sfile =
My.Computer.FileSystem.OpenTextFileWriter(Save_settings.FileName(), True)

    sfile.WriteLine(Zone_num.SelectedIndex)
    For i = 1 To 8
        sfile.WriteLine(c(i))
        sfile.WriteLine(Form1.n(i))
    Next i
    sfile.WriteLine(Load_sensors.FileName)
    For j = 1 To 8
        sfile.WriteLine(tse(j).SelectedItem)
        For i = 1 To 8
            If se(i, j).Checked = True Then
                sfile.WriteLine(1)
            Else
                sfile.WriteLine(0)
            End If
        Next i
    Next j
    If sms_report.Checked = True Then
        sfile.WriteLine(1)
    Else
        sfile.WriteLine(0)
    End If
    sfile.WriteLine(Phone_text.Text)
    If Mail_report.Checked = True Then
        sfile.WriteLine(1)
    Else
        sfile.WriteLine(0)
    End If
    sfile.WriteLine(Host_text.Text)
    sfile.WriteLine(Email_text.Text)
    sfile.WriteLine(encryptString(Pass_text.Text))
    If Auto_Report.Checked = True Then
        sfile.WriteLine(1)
    Else
        sfile.WriteLine(0)
    End If
    sfile.Close()
End Sub

```

Ακολουθούν οι sub μέθοδοι Load_Click() και Load_settings_FileOk() που κάνουν ακριβώς την αντίστροφη διαδικασία από αυτήν που προαναφέραμε για τις Save_Click() και Save_settings_FileOk() μεθόδους, όταν πατηθεί του κουμπι “Load” και επιλεχθεί ένα αρχείο “.txt” που έχουμε αποθηκεύσει προηγούμενες ρυθμίσεις. Η Load_settings_FileOk() διαβάζει γραμμή-γραμμή ακριβώς με την ίδια σειρά που η Save_settings_FileOk() έχει γράψει στο αρχείο “.txt”. Στην αρχή επιλέγει στο combobox τον αριθμό των ζωνών που αντιστοιχεί στον αριθμό της γραμμής που διαδάζει. Μετά περνάει στις αντίστοιχες μεταβλητές, πόσους αισθητήρες έχει η κάθε ζώνη και τον δείκτη που μας δείχνει σε ποια ζώνη ανοίκει ο κάθε αισθητήρας. Έπειτα φορτώνει το path του αρχείου των ειδών των αισθητήρων που έχουμε γράψει. Στην συνέχεια διαβάζει γραμμή-γραμμή 1 ή 0 και επιλέγει ή όχι τα checkboxes των αισθητηρίων αντίστοιχα και επιλέγει τον τύπο του αισθητήρα που έχουμε γράψει στα αντίστοιχα comboboxes. Στην συνέχεια διαβάζει 1 ή 0 και επιλέγει ή όχι τα checkboxes των sms_report και mail_report και ενδιάμεσα γράφει το κείμενο στο textbox “Phone”, και έπειτα τα κείμενα στα textboxes “Host” και “Email” και αποκωδικοποιημένο το κείμενο στο textbox του “Password”. Τέλος διαβάζει 1 ή 0 και επιλέγει ή όχι το checkbox Auto_report.

```

Private Sub Load_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Load.Click
    Load_settings.ShowDialog()
End Sub
Private Sub Load_settings_FileOk(ByVal sender As System.Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles Load_settings.FileOk
    Dim lfile As System.IO.StreamReader
    lfile =
My.Computer.FileSystem.OpenTextFileReader(Load_settings.FileName())

    Zone_num.SelectedIndex = lfile.ReadLine()
    For i = 1 To 8
        c(i) = lfile.ReadLine()
        Form1.n(i) = lfile.ReadLine()
    Next i
    Load_sensors.FileName = lfile.ReadLine()
    Load_sensors_FileOk(Load_sensors, e)
    For j = 1 To 8
        tse(j).SelectedItem = lfile.ReadLine()
        For i = 1 To 8
            If lfile.ReadLine() = 1 Then
                se(i, j).Checked = True
            Else
                se(i, j).Checked = False
            End If
        Next i
    Next j

    If lfile.ReadLine() = 1 Then
        sms_report.Checked = True
    Else
        sms_report.Checked = False
    End If
    Phone_text.Text = lfile.ReadLine()
    If lfile.ReadLine() = 1 Then
        Mail_report.Checked = True
    
```

```
Else
    Mail_report.Checked = False
End If
Host_text.Text = lfile.ReadLine()
Email_text.Text = lfile.ReadLine()
Pass_text.Text = decryptString(lfile.ReadLine)
If lfile.ReadLine() = 1 Then
    Auto_Report.Checked = True
Else
    Auto_Report.Checked = False
End If
lfile.Close()
End Sub
```

Στην συνέχεια έχουμε την sub μέθοδο Ok_Click(), η οποία ενεργοποιείται όταν πατηθεί το κουμπί “Ok”, καταχωρεί τις ρυθμίσεις μας και μας περνάει στην κυρίως φόρμα. Αν έχουμε ξεχάσει να επιλέξουμε πόσες ζώνες θέλουμε μας ειδοποιεί με το αντίστοιχο μήνυμα με ένα messagebox. Στην συνέχεια ελέγχει αν στις ενεργοποιημένες ζώνες έχουμε επιλέξει αισθητήρες, διαφορετικά μας πάλι ειδοποιεί με το αντίστοιχο μήνυμα με ένα messagebox για την συγκεκριμένη ζώνη. Έπειτα ελέγχει αν στους επιλεγμένους αισθητήρες έχουμε διαλέξει είδος αλλιώς πάλι μας εμφανίζει αντίστοιχο μήνυμα. Μετά ελέγχει αν τα checkboxes sms_report και Mail_report είναι επιλεγμένα και αν έχουν συμπληρωθεί τα πεδία που είναι ενεργοποιημένα, σε αντίθετη περίπτωση εμφανίζει και εδώ ένα messagebox με το ανάλογο μήνυμα . Τέλος έχουμε την sub μέθοδο Cancel_Click() ενεργοποιείται όταν πατηθεί το κουμπί “Cancel” το οποίο κλείνει την εφαρμογή μας.

```
Private Sub Ok_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Ok.Click
    Form1.phone = Phone_text.Text

    Form1.host = Host_text.Text
    Form1.email = Email_text.Text
    Form1.pass = Pass_text.Text

    Dim q As Short
    For i = 0 To Zone_num.SelectedIndex + 1
        If c(i) > 0 Then
            q = q + 1
        End If
    Next i
    If Zone_num.SelectedIndex = -1 Then
        MessageBox.Show("Chose how many zones you want!")
        Zone_num.Focus()
        GoTo break
    ElseIf q <> Zone_num.SelectedIndex + 1 Then
        For j = 0 To 7
            If Zone_num.SelectedIndex = j Then
                For i = 0 To j
```

Πτυχιακή εργασία του φοιτητή Καρδάρη Γεώργιου

```
1))
        If c(i + 1) = 0 Then
            MessageBox.Show("Chose sensors for zone " + nu_z(i +
                GoTo break
            End If
        Next i
    End If
Next j
Else
    For i = 1 To 8
        If tse(i).Enabled = True And tse(i).SelectedItem = "" Then
            MessageBox.Show("Select the type of sensor " + CStr(i))
            GoTo break
        End If
    Next i
End If
If sms_report.Checked = True And Phone_text.Text = "" Then
    MessageBox.Show("Insert a phone number for sms report!")
    Phone_text.Focus()
ElseIf Mail_report.Checked = False Then
    Me.Close()
End If
If Mail_report.Checked = True Then
    If Host_text.Text = "" Then
        MessageBox.Show("Insert your hostname for mail report!")
        Host_text.Focus()
    ElseIf Email_text.Text = "" Then
        MessageBox.Show("Insert your E-mail for mail report!")
        Email_text.Focus()
    ElseIf Pass_text.Text = "" Then
        MessageBox.Show("Insert your password for mail report!")
        Pass_text.Focus()
    Else
        Me.Close()
    End If
End If
break:
    End Sub

    Private Sub Cancel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Cancel.Click
        Form1.Close()
    End Sub
```

Τέλος έχουμε την sub μέθοδο Form2_Load() η οποία ενεργοποιείται όταν φορτώνεται η φόρμα των ρυθμίσεων και ενεργοποιεί την μέθοδο se_name() που “γεμίζει” τους πίνακες c(8), se(8, 8), tse(8) και προσθέτει “Handlers” στην μέθοδο Checked_changed() οι οποίοι την ενεργοποιούν.

```
Private Sub Form2_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

    se_named()
    For j = 1 To 8
        For i = 1 To 8
            AddHandler se(i, j).CheckedChanged, AddressOf Checked_changed
        Next i
    Next j
```

```
End Sub
```

3.1.2 Περιγραφή κώδικα της κυρίως φόρμας (form1)

Στον κώδικα της κυρίως φόρμας στην αρχή κάναμε τις παρακάτω επιλογές και εισάγαμε το System.Net.Mail. Το κομμάτι αυτό της βιβλιοθήκης System μας παρέχει μια διεπαφή διαχείρισης για το SMTP με κυριότερες κλάσεις τις: MailMessage, που αναπαριστά τα μηνύματα, MailAttachment, που αναπαριστά τα συνημμένα αρχεία και SmtpClient, που χειρίζεται την υπηρεσία μηνυμάτων SMTP.

Επίσης δηλώσαμε τις μεθόδους Inp() και Out() που θα χρησιμοποιήσουμε από την βιβλιοθήκη του inpout32.dll,(το οποίο το κατεβάσαμε από την ιστοσελίδα http://logix4u.net/Legacy_Ports/Parallel_Port/) για την επικοινωνία με την παράλληλη θύρα. Η Inp() παίρνει ως όρισμα τον αριθμό της διεύθυνσης της παράλληλης θύρας και επιστρέφει έναν αριθμό short που είναι η τιμή των συγκεκριμένων bits και η Out() ως όρισμα έναν short αριθμό που αντιστοιχεί στην διεύθυνση της παράλληλης θύρας και έναν αριθμό short που είναι η τιμή που θέτει τα bits τις συγκεκριμένης διεύθυνσης.

```
Option Strict Off
Option Explicit On
Imports System.Net.Mail
```

```
Module InpOut32_Declarations
```

```
    'Inp and Out declarations for port I/O using inpout32.dll.
```

```
    Public Declare Function Inp Lib "inpout32.dll" Alias "Inp32" (ByVal
PortAddress As Short) As Short
```

```
    Public Declare Sub Out Lib "inpout32.dll" Alias "Out32" (ByVal PortAddress As
Short, ByVal Value As Short)
```

```
End Module
```

```
Public Class Form1
    Public Shared host, email, pass, phone, msg_text, tim As String
    Public Shared n(8) As Short
    Private sb(8) As Button
    Private zone(8) As CheckBox
    Public Shared ti, sms_ti As Integer
    Dim x, y, c, lo(8), lo8, lo7, lo6, lo5, lo4, lo3, lo2, lo1 As Short
    Dim file As System.IO.StreamWriter
```

Στις μεταβλητές host, email, pass, phone καταχωρούνται οι τιμές από τα αντίστοιχα πεδία της φόρμας των ρυθμίσεων. Η msg_text έχει το κείμενο που στέλνεται κάθε φορά και η tim την τρέχουσα ημερομηνία και ώρα. Ο πίνακας n(8) δείχνει σε ποια ζώνη ανήκει ο κάθε αισθητήρας. Οι μονοδιάστατοι πίνακες sb(8) και zone(8) περιέχουν τα κουμπιά των αισθητήρων και τα checkboxes των ζωνών αντίστοιχα για να τα χειζόμαστε ομαδικά. Οι ti και sms_ti μας θα μας χρειαστούν ως μετρήτες στους timers που θα χρησιμοποιήσουμε. Ο c θα χρησιμοποιηθεί ως flag για την εμφάνιση του μηνύματος "ALERT" και την αναπαραγωγή του ήχου ή

όχι. Οι μεταβλητές x,y θα μας βοηθήσουν να υπολογίσουμε μέσα από μεθόδους ποια data pins της παράλληλης είναι ενεργοποιημένα. Ο πίνακας lo(8) και οι μεταβλητές lo1-8 χρησιμοποιούνται ως flags για την εγγραφή ή μη των αναφορών παραβίασης στο αρχείο "logfile" που κάνουμε με την χρήση της μεταβλητής file.

Ακολουθούν οι sub μέθοδοι bu_named() και che_named() με τις οποίες "γεμίζουμε" τους πίνακες sb(8) και zone(8) με τα ονόματα των κουμπιών και των checkboxes αντίστοιχα.

Στην συνέχεια έχουμε τις sub μεθόδους zone_enable() και sensor_enable() οι οποίες ενεργοποιούν ή απενεργοποιούν ανάλογα με της επιλογές μας στην φόρμα των ρυθμίσεων τα checkboxes των ζωνών και τα κουμπιά των αισθητήρων, αντίστοιχα.

Ακολουθεί η sub μέθοδος s_Click() η οποία ενεργοποιείται όταν βρισκόμαστε σε κατάσταση "Test Mode" και πατηθεί κάποιο κουμπί που αντιστοιχεί σε αισθητήρα και στην συνέχεια ενεργοποιεί την ζώνη που αντιστοιχεί αυτός και γράφει την αναφορά στο αρχείο logfile.

```

Private Sub s_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)
    For i = 1 To 8
        If sb(i) Is sender Then
            For j = 1 To 8
                If n(i) = j Then
                    zone(j).Checked() = True
                    c = 1
                    If lo(i) = 0 Then
                        lo(i) = 1
                        msg_text += " *TEST MODE* Violation of the " +
Form2.tse(i).SelectedItem + " sensor " + CStr(i) + " in Zone " + Form2.nu_z(j) +
"!" & vbNewLine
                        file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
                        file.WriteLine(trim + " *TEST MODE* Violation of the "
+ Form2.tse(i).SelectedItem + " sensor " + CStr(i) + " in Zone " + Form2.nu_z(j)
+ "!")
                        file.Close()
                    End If
                End If
            Next j
        End If
    Next i
End Sub

```

Ακολουθεί η sub μέθοδος reset_Click() η οποία ενεργοποιείται όταν βρισκόμαστε σε κατάσταση "Test Mode" και πατηθεί το κουμπί "RESET" και απενεργοποιεί τυχόν παραβιάσεις που έχουμε προκαλέσει και γράφει την αναφορά στο αρχείο logfile.

Έπειτα έχουμε την sub μέθοδος report_Click() η οποία ενεργοποιείται αν πατηθεί, αυτόματα ή μη, το κουμπί "REPORT". Αρχικά ελέγχει αν υπάρχει κάποιο μήνυμα παραβίασης ή όχι , μετά ελέγχει τον μετρητή sms_ti και τέλος αν έχουμε επιλέξει

στην φόρμα των ρυθμίσεων τα checkboxes sms_report και mail_report προχωρά στην αποστολή sms και mail αντίστοιχα.

```
Private Sub report_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles report.Click
    Dim msg_f As Short
    If msg_text <> "" Then
        If sms_ti > 60 Or sms_ti = 0 Then
            If Form2.sms_report.Checked = True Then
                Dim Command As String
                Command = "ipconfig " + phone + " " + msg_text
                Shell("cmd /k " & Command, 1, True)
            End If
            msg_f = 1
        End If
        If sms_ti > 5 Or sms_ti = 0 Then
            If Form2.Mail_report.Checked = True Then
                Dim smtpServer As New SmtplibClient()
                Dim mail As New MailMessage
                smtpServer.Credentials = New Net.NetworkCredential(email,
pass)

                smtpServer.Port = 587
                smtpServer.Host = host
                smtpServer.EnableSsl = False
                mail.From = New MailAddress(email)
                mail.To.Add(email)
                mail.Subject = "ALARM"
                mail.Body = msg_text

                smtpServer.Send(mail)
            End If
            msg_f = 1
        End If
        If msg_f = 1 Then
            msg_text = ""
            sms_ti = 0
            msg_f = 0
        End If
        SmS_Timer.Start()
    End If
End Sub
```

Στην συνέχεια έχουμε την sub μέθοδο setup_Click() η οποία ενεργοποιείται όταν πατηθεί το κουμπί "SETUP" και μας εμφανίζει την φόρμα ρυθμίσεων (form2).

Έπειτα έχουμε τις sub μεθόδους για την ανάγνωση των αισθητηρίων s8_zone(), s7_zone(), s6_zone(), s5_zone(), s4_zone(), s3_zone(), s2_zone(), s1_zone() οι οποίες θα καλεστούν με αυτήν την σειρά στην συνέχεια. Αφού ελέγξουν σε ποια ζώνη ανήκει ο αισθητήρας ενεργοποιεί ή όχι την αντίστοιχη ζώνη και γράφει αναφορά στο αρχείο logfile και στην msg_text για αποστολή. Ο έλεγχος αυτός γίνεται με την βοήθεια των μεταβλητών x, y. Η μεταβλητή x αρχικά έχει την δεκαδική τιμή που διαβάζουμε από τα Data pins της παράλληλης θύρας και αφού το συγκρίνουμε αρχικά το 128, που στο δυαδικό είναι το 10000000, αν είναι μεγαλύτερο ή ίσο, τότε ο αισθητήρας που είναι συνδεδεμένος με το Data pin 8 της παράλληλης θύρας είναι ενεργοποιημένος και ενεργοποιείται η αντίστοιχη ζώνη,

και το y παίρνει την τιμή 128, αλλιώς ο αισθητήρας δεν είναι ενεργοποιημένος και το y παίρνει την τιμή 0. Στο τέλος αφαιρεί το y από το x και συνεχίζουμε στην επόμενη μέθοδο και συγκρίνοντας με τον 64 (1000000) για το Data pin 7 κάνοντας την αντίστοιχη διαδικασία και για τα υπόλοιπα Data pins συγκρίνοντας αντίστροφα κάθε φορά με τον αντίστοιχο αριθμό (32, 16 , 8, κ.ο.κ.).

```

Private Sub s8_zone()
    For i = 1 To 8
        If n(8) = i Then
            If x >= 128 Then
                zone(i).Checked() = True
                If ti Mod 2 = 0 Then
                    zone(i).BackColor = Color.Red
                Else
                    zone(i).BackColor = Color.Transparent
                End If
                y = 128
                If lo8 = 0 Then
                    lo8 = 1
                    msg_text += tim + " Violation of the " +
Form2.tse(8).SelectedItem + " sensor 8 in Zone " + Form2.nu_z(i) + "!" &
vbNewLine
                    file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
                    file.WriteLine(tim + " Violation of the " +
Form2.tse(8).SelectedItem + " sensor 8 in Zone " + Form2.nu_z(i) + "!" )
                    file.Close()
                End If
            Else
                If lo8 = 1 Then
                    lo8 = 0
                    file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
                    file.WriteLine(tim + " End violation of the " +
Form2.tse(8).SelectedItem + " sensor 8 in Zone " + Form2.nu_z(i) + "!" )
                    file.Close()
                End If
                If zone(i).Checked() = True Then
                    y = 0
                Else
                    y = 0
                    zone(i).BackColor = Color.Transparent
                    zone(i).Checked() = False
                End If
            End If
        End If
    Next i
    x = x - y
End Sub
Private Sub s7_zone()
    For i = 1 To 8
        If n(7) = i Then
            If x >= 64 Then
.
.
                x = x - y
            End Sub
Private Sub s6_zone()
    For i = 1 To 8
        If n(6) = i Then
            If x >= 32 Then

```



```

.
.
    x = x - y
End Sub
Private Sub s5_zone()
    For i = 1 To 8
        If n(5) = i Then
            If x >= 16 Then
.
.
.
End Sub
Private Sub s4_zone()
    For i = 1 To 8
        If n(4) = i Then
            If x >= 8 Then
.
.
.
    x = x - y
End Sub
Private Sub s3_zone()
    For i = 1 To 8
        If n(3) = i Then
            If x >= 4 Then
.
.
.
End Sub
Private Sub s2_zone()
    For i = 1 To 8
        If n(2) = i Then
            If x >= 2 Then
.
.
.
    x = x - y
End Sub
Private Sub s1_zone()
    For i = 1 To 8
        If n(1) = i Then
            If x >= 1 Then
.
.
End Sub

```

Στην συνέχεια έχουμε την sub μέθοδο Test_mode_CheckedChanged() η οποία ενεργοποιείται όταν το checkbox “Test Mode” επιλεγεί, και μεγαλώνει την φόρμα μας εμφανίζοντας τα “Test Buttons” ,κάνει το χρώμα του πράσινο και τον τίτλο του “Test Mode ON” και γράφει αναφορά στο αρχείο logfile και στην msg_text για αποστολή, και αντίστροφα μικραίνει την φόρμα μας αποκρύπτοντας τα “Test Buttons” , και κάνει τον τίτλο του “Test Mode OFF” και γράφει αναφορά στο αρχείο logfile.

```

Private Sub Test_mode_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Test_mode.CheckedChanged
    If Test_mode.Checked = True Then
        Height() = 550
        .
        .
        Test_Area.Enabled = True

```

```

Else
    Height() = 300
    .
    .
    Test_Area.Enabled = False
End If
End Sub

```

Ακολουθούν οι sub μέθοδοι SmS_Tick() και Timer1_Tick(). Η πρώτη ενεργοποιείται με τον SmS_Timer, ανά 1 δευτερόλεπτο, όταν σταλθεί κάποιο mail ή sms και περιέχει τον μετρητή sms_ti και τον αυξάνει.

Η δεύτερη ενεργοποιείται με τον Timer1, ανά μισό δευτερόλεπτο, όταν φορτωθεί η κυρίως φόρμα μας και περιέχει τον κύριο αλγόριθμο της εφαρμογής μας. Αρχικά τρέχει τις μεθόδους zone_enable() και sensor_enable() για να ενεργοποιήσει τις ζώνες και τα κουμπιά των αισθητήρων. Έπειτα αυξάνει τον μετρητή ti και περνάει στην tim την τρέχουσα ημερομηνία και ώρα. Στην συνέχεια αφού ελέγξει αν βρισκόμαστε σε Test Mode OFF ή ON θέτει αντίστοιχα, τα data pins της παράλληλης θύρας σε κατάσταση εισόδου ή όχι και ακολουθεί τις αντίστοιχες μεθόδους κάθε φορά για την ανάγνωση των αισθητηρίων και την ενεργοποίηση του συναγερμού και των ζωνών του. Επίσης αν ενεργοποιηθεί ο συναγερμός εμφανίζει το αντίστοιχο μήνυμα, αλλάζει περιοδικά το background χρώμα των παραβιασμένων ζωνών και αναπαράγει το attention.wav . Τέλος αν ελέγχει αν έχουμε επιλέξει αυτόματη ειδοποίηση ή όχι και ενεργεί ανάλογα.

```

Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
    zone_enable()
    sensor_enable()
    ti = ti + 1
    tim = System.DateTime.Now
    If Test_mode.Checked = False Then
        Out(&H378S + 2, 224)
        If ti Mod 3 Then
            reset_sub()
        End If
        x = Inp(&H378S)

        s8_zone()
        '=====
        s7_zone()
        '=====
        s6_zone()
        '=====
        s5_zone()
        '=====
        s4_zone()
        '=====
        s3_zone()
        '=====
        s2_zone()
        '=====
        s1_zone()

```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
        If ZONE_A.Checked = True Or ZONE_B.Checked = True Or ZONE_C.Checked =  
True Or ZONE_D.Checked = True Or ZONE_E.Checked = True Or ZONE_F.Checked = True  
Or ZONE_G.Checked = True Or ZONE_H.Checked = True Then  
            Label2.Visible = True  
            If ti Mod 2 = 0 Then  
                My.Computer.Audio.Play(My.Resources.attention,  
AudioPlayMode.Background)  
            End If  
        Else  
            Label2.Visible = False  
        End If  
  
    Else  
        Out(&H378S + 2, 192)  
        If c = 1 Then  
            Label2.Visible = True  
            If ti Mod 2 = 0 Then  
                My.Computer.Audio.Play(My.Resources.attention,  
AudioPlayMode.Background)  
            End If  
        Else  
            Label2.Visible = False  
        End If  
        For j = 1 To 8  
            For i = 1 To 8  
                If n(j) = i Then  
                    If zone(i).Checked = True Then  
                        If ti Mod 2 = 0 Then  
                            zone(i).BackColor = Color.Red  
                        Else  
                            zone(i).BackColor = Color.Transparent  
                        End If  
                    End If  
                End If  
            End If  
        Next i  
    Next j  
End If  
If ti Mod 8 - 7 = 0 Then  
    If Form2.Auto_Report.Checked = True Then  
        report_Click(True, AcceptButton)  
    End If  
End If  
End Sub
```

Τελειώνοντας έχουμε την μέθοδο Form1_Load() η οποία τρέχει με το φόρτωμα της κυρίως φόρμας που στην αρχή μας εμφανίζει την φόρμα των ρυθμίσεων ενεργοποιεί τις μεθόδους bu_named() και che_named() που “γεμίζουν” τους πίνακες sb(8) και zone(8) και προσθέτει “Handlers” στην μέθοδο s_Click ()οι οποίοι την ενεργοποιούν,και τέλος ενεργοποιεί τον Timer1.

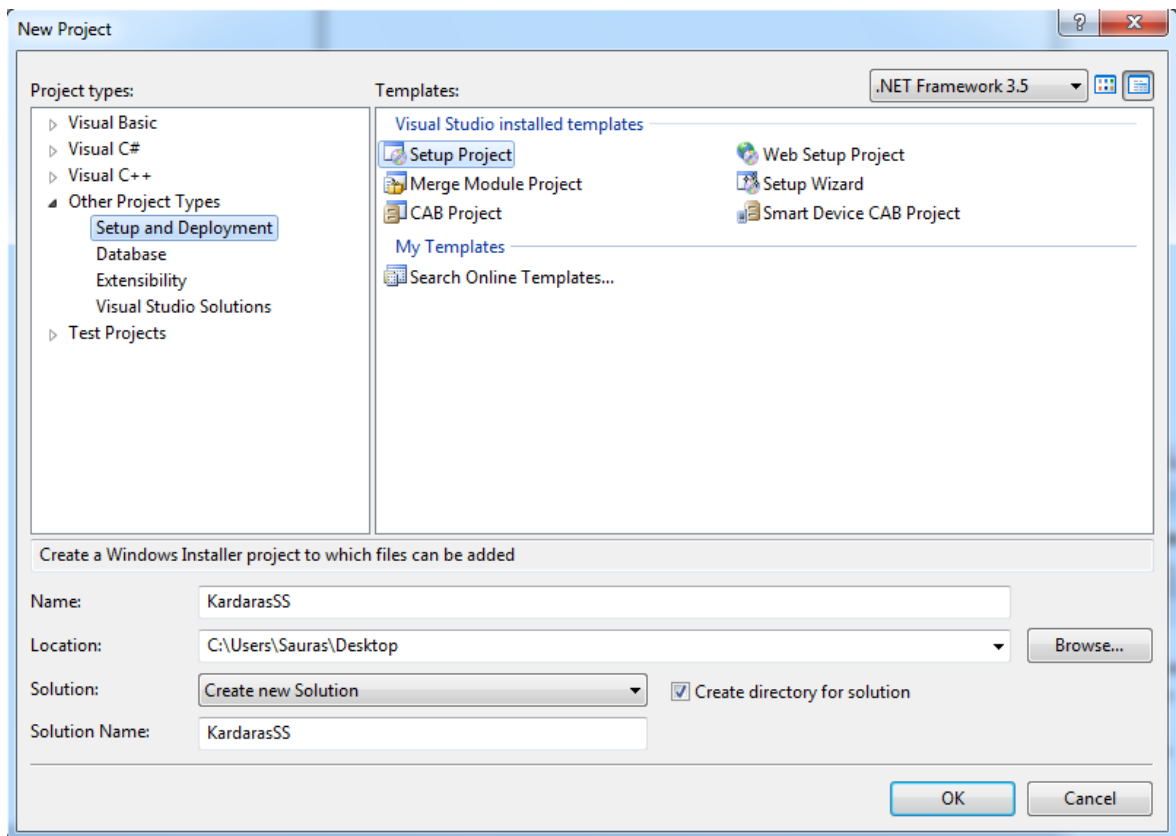
```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles MyBase.Load  
    Form2.ShowDialog()  
    Height() = 300  
    bu_named()  
    che_named()  
    ti = 0  
    For i = 1 To 8  
        AddHandler sb(i).Click, AddressOf s_Click
```

```
Next i  
Timer1.Start ()  
End Sub
```

3.2 Περιγραφή κατασκευής οδηγού εγκατάσταση

Το Microsoft Visual Studio 2008 μας παρέχει την δυνατότητα δημιουργίας ενός οδηγού εγκατάστασης της εφαρμογής και προαπαιτούμενων αρχείων.

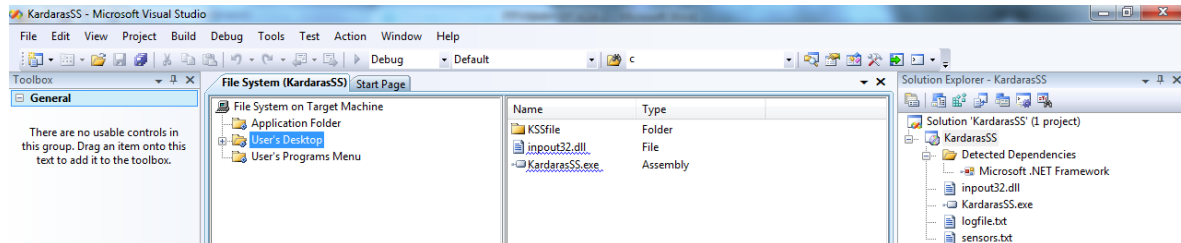
Δημιουργώντας ένα νέο πρότζεκτ όπως έχουμε δείξει στο παράθυρο New Project στην περιοχή Project types επιλέγουμε Others Projects Types->Setup and Deployment. Στην περιοχή Templates επιλέγουμε Setup Project και δίνουμε τον κατάλογο και το όνομα που θέλουμε να σώσουμε (Εικόνα 37).



Εικόνα 37: Παράθυρο επιλογών νέου πρότζεκτ

Στην συνέχεια ανοίγει το βασικό περιβάλλον του Microsoft Visual Studio 2008 έχοντας δημιουργήσει ένα δέντρο με κόμβους τους φακέλους Application Folder, User's Desktop και User's Program Menu. Κάνοντας δεξί κλικ πάνω σε κάποιον από αυτούς μπορούμε να προσθέσουμε αρχεία, φακέλους και ότι άλλο θέλουμε να εγκατασταθεί σε αυτούς του H/Y που θα κάνουμε εγκατάσταση το πρόγραμμά μας. Έτσι λοιπόν εισάγαμε το inrout32.dll, KardarasSS.exe, που δημιουργήσαμε

κάνοντας debug τον κώδικά μας, και τον φάκελο KSSfile με τα αρχεία logfile.txt, όπου θα καταγράφεται το ιστορικό, και sensors.txt, όπου θα γράψουμε τα είδη των αισθητήρων, στον φάκελο User's Desktop για να εγκατασταθούν στην επιφάνεια εργασίας του Η/Υ που θα κάνουμε την εγκατάσταση (Εικόνα 38).



Εικόνα 38: Δημιουργία KardarasSS setup

Πατώντας στην μπάρα μενού το Build->Build KardarasSS δημιουργείται το KardarasSS setup το οποίο θα εγκαταστήσει τα παραπάνω αρχεία μας στην επιφάνεια εργασίας του Η/Υ που θα το τρέξουμε.

ΕΠΙΛΟΓΟΣ

Στόχος μας ήταν ολοκληρώνοντας το πρόγραμμα μας, η εφαρμογή να υλοποιεί τις λειτουργίες του κεντρικού εγκεφάλου ενός συναγερμού οικίας. Η εφαρμογή μας κατασκευάστηκε για Η/Υ με λειτουργικό σύστημα Windows γιατί είναι πιο διαδεδομένο στην χώρα μας. Για να τρέξει σε Η/Υ με λειτουργικό σύστημα Linux υπάρχει ένα πρόγραμμα το ονομαζόμενο "Mono" (βλ. http://www.mono-project.com/VisualBasic.NET_support), αν και θα ήταν προτιμότερο να κατασκευαστεί από την αρχή αν θέλαμε να το υλοποιήσουμε σε Η/Υ με Linux για να γλιτώσουμε πιθανά προβλήματα ασυμβατότητας. Επίσης θα μπορούσε να γραφτεί και σε άλλες γλώσσες προγραμματισμού, όπως σε Java και σε Visual C# ή Visual C++ όμως επιλέξαμε την Visual Basic.NET και το Microsoft Visual Studio 2008 και έπειτα από συνεχής αναζήτηση πληροφοριών στο διαδίκτυο σε forums και σε ιστοσελίδες με διάφορα tutorials καταφέραμε την κατασκευή μιας αρκετά ολοκληρωμένης εφαρμογής, παρόλο που οι γνώσεις σε αυτή τη γλώσσα προγραμματισμού ήταν περιορισμένες.

ΣΥΜΠΕΡΑΣΜΑΤΑ

Για την δημιουργία της εφαρμογής της πτυχιακής εργασίας χρειάστηκαν γνώσεις προγραμματισμού που προήλθαν από τα αντίστοιχα μαθήματα όλων των εξαμήνων της σχολής Προγραμματισμός Η/Υ I, Προγραμματισμός Η/Υ II, Δομές Δεδομένων, Ανάπτυξη και Διαχείριση Εφαρμογών κ.ά. και από συνεχής αναζήτηση στο διαδίκτυο , και για την κατασκευή της διεπαφής του υλικού γνώσεις που προήλθαν από το μάθημα της σχολής Ψηφιακά Συστήματα και μεράκι.

Στην διεπαφή υλικού θα μπορούσαν να αντικατασταθούν αισθητήρες οποιαδήποτε άλλου είδους, καθώς και ασύρματοι αισθητήρες, αλλά και να αλλάξει ο τρόπος σύνδεσης με τον Η/Υ, αντί για παράλληλη θα μπορούσε να χρησιμοποιηθεί σειριακή ή USB, με τις κατάλληλες τροποποιήσεις σε συνδέσεις και στο λογισμικό. Αλλά και η εφαρμογή θα μπορούσε να δημιουργηθεί διαδικτυακά και η απομακρυσμένη διαχείριση της να γίνεται άμεσα μέσω του διαδικτύου.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- http://www.eyecat.gr/services/security_systems (Ιανουάριος 2011)
- <http://www.thiefbusters.gr/> (Ιανουάριος 2011)
- http://www.kpsystems.gr/systimata_asfaleias.php (Ιανουάριος 2011)
- <http://www.doorado.gr/news/synagermoi-spition/> (Ιανουάριος 2011)
- http://logix4u.net/Legacy_Ports/Parallel_Port/ (Σεπτέμβριος 2010)
- <http://www.mechatronics.gr/el/rd/10-interfacing-the-parallel-port.htm>
- <http://www.codeproject.com/> (Φόρουμ Σεπτέμβριος 2010-Μαρτιος 2011)
- <http://www.tutorialized.com/tutorials/Visual-Basic/1> (Σεπτέμβριος 2010-Μαρτιος 2011)

ΠΑΡΑΡΤΗΜΑΤΑ

Κώδικας κύριας φόρμας

```
Option Strict Off
Option Explicit On
Imports System.Net.Mail

Module InpOut32_Declarations

    'Inp and Out declarations for port I/O using inpout32.dll.

    Public Declare Function Inp Lib "inpout32.dll" Alias "Inp32" (ByVal
PortAddress As Short) As Short

    Public Declare Sub Out Lib "inpout32.dll" Alias "Out32" (ByVal PortAddress As
Short, ByVal Value As Short)

End Module

Public Class Form1
    Public Shared host, email, pass, phone, msg_text, tim As String
    Public Shared n(8) As Short
    Private sb(8) As Button
    Private zone(8) As CheckBox
    Public Shared ti, sms_ti As Integer
    Dim x, y, c, lo(8), lo8, lo7, lo6, lo5, lo4, lo3, lo2, lo1 As Short
    Dim file As System.IO.StreamWriter

    '=====
    Private Sub bu_named()
        sb(1) = s1
        sb(2) = s2
        sb(3) = s3
        sb(4) = s4
        sb(5) = s5
        sb(6) = s6
        sb(7) = s7
        sb(8) = s8
    End Sub
    Private Sub che_named()
        zone(1) = ZONE_A
        zone(2) = ZONE_B
        zone(3) = ZONE_C
        zone(4) = ZONE_D
        zone(5) = ZONE_E
        zone(6) = ZONE_F
        zone(7) = ZONE_G
        zone(8) = ZONE_H
    End Sub
    '=====
    Private Sub zone_enable()
        For i = 0 To 7
            If Form2.Zone_num.SelectedIndex = i Then
                For j = 1 To i + 1
                    zone(j).Enabled = True
                Next j
                For j = i + 2 To 8
                    zone(j).Enabled = False
                Next j
            End If
        Next i
    End Sub
End Class
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
        Next j
    End If
Next i

End Sub
Private Sub sensor_enable()
    For i = 1 To 8
        If n(i) = 0 Then
            sb(i).Enabled = False
        Else
            sb(i).Enabled = True
        End If
    Next i
End Sub

'=====TEST BUTTONS=====
Private Sub s_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)
    For i = 1 To 8
        If sb(i) Is sender Then
            For j = 1 To 8
                If n(i) = j Then
                    zone(j).Checked() = True
                    c = 1
                    If lo(i) = 0 Then
                        lo(i) = 1
                        msg_text += " *TEST MODE* Violation of the " +
Form2.tse(i).SelectedItem + " sensor " + CStr(i) + " in Zone " + Form2.nu_z(j) +
"! " & vbNewLine
                        file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
                        file.WriteLine(TIM + " *TEST MODE* Violation of the "
+ Form2.tse(i).SelectedItem + " sensor " + CStr(i) + " in Zone " + Form2.nu_z(j)
+ "! ")
                        file.Close()
                    End If
                End If
            Next j
        End If
    Next i
End Sub

'=====RESET BUTTON=====
Private Sub reset_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles reset.Click
    Out(&H378S, 0)
    Label2.Visible = False
    My.Computer.Audio.Stop()
    c = 0
    msg_text += " *TEST MODE* Reset of Violations!" & vbNewLine
    file = My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt",
True)
    file.WriteLine(TIM + " *TEST MODE* Reset of Violations!")
    For i = 1 To 8
        lo(i) = 0
        zone(i).Checked() = False
        zone(i).BackColor = Color.Transparent
    Next i
    file.Close()
End Sub

'=====
Private Sub reset_sub()
    For i = 1 To 8
        zone(i).Checked() = False
    Next i
End Sub
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
        zone(i).BackColor = Color.Transparent
    Next
End Sub
'=====REPORT BUTTON=====
Private Sub report_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles report.Click
    Dim msg_f As Short
    If msg_text <> "" Then
        If sms_ti > 300 Or sms_ti = 0 Then
            If Form2.sms_report.Checked = True Then
                Dim Command As String
                Command = "sms.cmd " + phone + " " + msg_text
                Shell("cmd /k " & Command, 1, True)
            End If
            msg_f = 1
        End If
        If sms_ti > 5 Or sms_ti = 0 Then
            If Form2.Mail_report.Checked = True Then
                Dim smtpServer As New SmtplibClient()
                Dim mail As New MailMessage
                smtpServer.Credentials = New Net.NetworkCredential(email,
pass)

                smtpServer.Port = 587
                smtpServer.Host = host
                smtpServer.EnableSsl = False
                mail.From = New MailAddress(email)
                mail.To.Add(email)
                mail.Subject = "ALARM"
                mail.Body = msg_text

                smtpServer.Send(mail)
            End If
            msg_f = 1
        End If
        If msg_f = 1 Then
            msg_text = ""
            sms_ti = 0
            msg_f = 0
        End If
        SmS_Timer.Start()
    End If
End Sub
'=====SETUP BUTTON=====
Private Sub setup_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles setup.Click
    Form2.ShowDialog()
End Sub
'=====
Private Sub s8_zone()
    For i = 1 To 8
        If n(8) = i Then
            If x >= 128 Then
                zone(i).Checked() = True
                If ti Mod 2 = 0 Then
                    zone(i).BackColor = Color.Red
                Else
                    zone(i).BackColor = Color.Transparent
                End If
            End If
            y = 128
            If lo8 = 0 Then
                lo8 = 1
            End If
        End If
    Next
End Sub
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
        msg_text += tim + " Violation of the " +
Form2.tse(8).SelectedItem + " sensor 8 in Zone " + Form2.nu_z(i) + "!" &
vbNewLine
        file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
        file.WriteLine(tim + " Violation of the " +
Form2.tse(8).SelectedItem + " sensor 8 in Zone " + Form2.nu_z(i) + "!" &
vbNewLine)
        file.Close()
    End If
Else
    If lo8 = 1 Then
        lo8 = 0
        file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
        file.WriteLine(tim + " End violation of the " +
Form2.tse(8).SelectedItem + " sensor 8 in Zone " + Form2.nu_z(i) + "!" &
vbNewLine)
        file.Close()
    End If
    If zone(i).Checked() = True Then
        y = 0
    Else
        y = 0
        zone(i).BackColor = Color.Transparent
        zone(i).Checked() = False
    End If
End If
End If
Next i
x = x - y
End Sub
Private Sub s7_zone()
    For i = 1 To 8
        If n(7) = i Then
            If x >= 64 Then
                zone(i).Checked() = True
                If ti Mod 2 = 0 Then
                    zone(i).BackColor = Color.Red
                Else
                    zone(i).BackColor = Color.Transparent
                End If
                y = 64
                If lo7 = 0 Then
                    lo7 = 1
                    msg_text += " Violation of the " +
Form2.tse(7).SelectedItem + " sensor 7 in Zone " + Form2.nu_z(i) + "!" &
vbNewLine
                    file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
                    file.WriteLine(tim + " Violation of the " +
Form2.tse(7).SelectedItem + " sensor 7 in Zone " + Form2.nu_z(i) + "!" &
vbNewLine)
                    file.Close()
                End If
            Else
                If lo7 = 1 Then
                    lo7 = 0
                    file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
                    file.WriteLine(tim + " End violation of the " +
Form2.tse(7).SelectedItem + " sensor 7 in Zone " + Form2.nu_z(i) + "!" &
vbNewLine)
                    file.Close()
                End If
                If zone(i).Checked() = True Then
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
        y = 0
    Else
        y = 0
        zone(i).BackColor = Color.Transparent
        zone(i).Checked() = False
    End If
End If
End If
Next i
x = x - y
End Sub
Private Sub s6_zone()
    For i = 1 To 8
        If n(6) = i Then
            If x >= 32 Then
                zone(i).Checked() = True
                If ti Mod 2 = 0 Then
                    zone(i).BackColor = Color.Red
                Else
                    zone(i).BackColor = Color.Transparent
                End If
                y = 32
                If lo6 = 0 Then
                    lo6 = 1
                    msg_text += " Violation of the " +
Form2.tse(6).SelectedItem + " sensor 6 in Zone " + Form2.nu_z(i) + "!" &
vbNewLine
                    file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
                    file.WriteLine(ti + " Violation of the " +
Form2.tse(6).SelectedItem + " sensor 6 in Zone " + Form2.nu_z(i) + "!" )
                    file.Close()
                End If
            Else
                If lo6 = 1 Then
                    lo6 = 0
                    file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
                    file.WriteLine(ti + " End violation of the " +
Form2.tse(6).SelectedItem + " sensor 6 in Zone " + Form2.nu_z(i) + "!" )
                    file.Close()
                End If
                If zone(i).Checked() = True Then
                    y = 0
                Else
                    y = 0
                    zone(i).BackColor = Color.Transparent
                    zone(i).Checked() = False
                End If
            End If
        End If
    Next i
    x = x - y
End Sub
Private Sub s5_zone()
    For i = 1 To 8
        If n(5) = i Then
            If x >= 16 Then
                zone(i).Checked() = True
                If ti Mod 2 = 0 Then
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
        zone(i).BackColor = Color.Red
    Else
        zone(i).BackColor = Color.Transparent
    End If
    y = 16
    If lo5 = 0 Then
        lo5 = 1
        msg_text += " Violation of the " +
Form2.tse(5).SelectedItem + " sensor 5 in Zone " + Form2.nu_z(i) + "!" &
vbNewLine
        file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
        file.WriteLine(TIM + " Violation of the " +
Form2.tse(5).SelectedItem + " sensor 5 in Zone " + Form2.nu_z(i) + "!")
        file.Close()
    End If

    Else
        If lo5 = 1 Then
            lo5 = 0
            file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
            file.WriteLine(TIM + " End violation of the " +
Form2.tse(5).SelectedItem + " sensor 5 in Zone " + Form2.nu_z(i) + "!")
            file.Close()
        End If
        If zone(i).Checked() = True Then
            y = 0
        Else
            y = 0
            zone(i).BackColor = Color.Transparent
            zone(i).Checked() = False
        End If
    End If
End If
Next i
x = x - y
End Sub
Private Sub s4_zone()
    For i = 1 To 8
        If n(4) = i Then
            If x >= 8 Then
                zone(i).Checked() = True
                If ti Mod 2 = 0 Then
                    zone(i).BackColor = Color.Red
                Else
                    zone(i).BackColor = Color.Transparent
                End If
            End If
            y = 8
            If lo4 = 0 Then
                lo4 = 1
                msg_text += " Violation of the " +
Form2.tse(4).SelectedItem + " sensor 4 in Zone " + Form2.nu_z(i) + "!" &
vbNewLine
                file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
                file.WriteLine(TIM + " Violation of the " +
Form2.tse(4).SelectedItem + " sensor 4 in Zone " + Form2.nu_z(i) + "!")
                file.Close()
            End If

        Else
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
        If lo4 = 1 Then
            lo4 = 0
            file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
            file.WriteLine(TIM + " End violation of the " +
Form2.tse(4).SelectedItem + " sensor 4 in Zone " + Form2.nu_z(i) + "!")
            file.Close()
        End If
        If zone(i).Checked() = True Then
            y = 0
        Else
            y = 0
            zone(i).BackColor = Color.Transparent
            zone(i).Checked() = False
        End If
    End If
End If
Next i
x = x - y
End Sub
Private Sub s3_zone()
    For i = 1 To 8
        If n(3) = i Then
            If x >= 4 Then
                zone(i).Checked() = True
                If ti Mod 2 = 0 Then
                    zone(i).BackColor = Color.Red
                Else
                    zone(i).BackColor = Color.Transparent
                End If
                y = 4
                If lo3 = 0 Then
                    lo3 = 1
                    msg_text += " Violation of the " +
Form2.tse(3).SelectedItem + " sensor 3 in Zone " + Form2.nu_z(i) + "! " &
vbNewLine
                    file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
                    file.WriteLine(TIM + " Violation of the " +
Form2.tse(3).SelectedItem + " sensor 3 in Zone " + Form2.nu_z(i) + "!")
                    file.Close()
                End If
            Else
                If lo3 = 1 Then
                    lo3 = 0
                    file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
                    file.WriteLine(TIM + " End violation of the " +
Form2.tse(3).SelectedItem + " sensor 3 in Zone " + Form2.nu_z(i) + "!")
                    file.Close()
                End If
                If zone(i).Checked() = True Then
                    y = 0
                Else
                    y = 0
                    zone(i).BackColor = Color.Transparent
                    zone(i).Checked() = False
                End If
            End If
        End If
    End If
End Sub
```

Πτυχιακή εργασία του φοιτητή Καρδάρα Γεώργιου

```
Next i
x = x - y
End Sub
Private Sub s2_zone()
For i = 1 To 8
If n(2) = i Then
If x >= 2 Then
zone(i).Checked() = True
If ti Mod 2 = 0 Then
zone(i).BackColor = Color.Red
Else
zone(i).BackColor = Color.Transparent
End If
y = 2
If lo2 = 0 Then
lo2 = 1
msg_text += " Violation of the " +
Form2.tse(2).SelectedItem + " sensor 2 in Zone " + Form2.nu_z(i) + "!" &
vbNewLine
file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
file.WriteLine(tim + " Violation of the " +
Form2.tse(2).SelectedItem + " sensor 2 in Zone " + Form2.nu_z(i) + "!")
file.Close()
End If
Else
If lo2 = 1 Then
lo2 = 0
file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
file.WriteLine(tim + " End violation of the " +
Form2.tse(2).SelectedItem + " sensor 2 in Zone " + Form2.nu_z(i) + "!")
file.Close()
End If
If zone(i).Checked() = True Then
y = 0
Else
y = 0
zone(i).BackColor = Color.Transparent
zone(i).Checked() = False
End If
End If
End If
Next i
x = x - y
End Sub
Private Sub s1_zone()
For i = 1 To 8
If n(1) = i Then
If x >= 1 Then
zone(i).Checked() = True
If ti Mod 2 = 0 Then
zone(i).BackColor = Color.Red
Else
zone(i).BackColor = Color.Transparent
End If
y = 1
If lo1 = 0 Then
lo1 = 1
```


Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
        msg_text += " Violation of the " +
Form2.tse(1).SelectedItem + " sensor 1 in Zone " + Form2.nu_z(i) + "!" &
vbNewLine
        file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
        file.WriteLine(tim + " Violation of the " +
Form2.tse(1).SelectedItem + " sensor 1 in Zone " + Form2.nu_z(i) + "!" )
        file.Close()
    End If

    Else
        If lo1 = 1 Then
            lo1 = 0
            file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
            file.WriteLine(tim + " End violation of the " +
Form2.tse(1).SelectedItem + " sensor 1 in Zone " + Form2.nu_z(i) + "!" )
            file.Close()
        End If
        If zone(i).Checked() = True Then
            y = 0
        Else
            y = 0
            zone(i).BackColor = Color.Transparent
            zone(i).Checked() = False
        End If
    End If
End If
Next i
End Sub
'=====
'=====
Private Sub Test_mode_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles Test_mode.CheckedChanged
    If Test_mode.Checked = True Then
        Height() = 550
        Test_mode.BackColor = Color.LightGreen
        Test_mode.Text = "Test mode ON"
        Label2.Visible = False
        My.Computer.Audio.Stop()
        c = 0
        msg_text += " *TEST MODE ON* Reset of Violations!" & vbNewLine
        file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
        file.WriteLine(tim + " *TEST MODE ON* Reset of Violations!")
        For i = 1 To 8
            lo(i) = 0
            zone(i).Checked() = False
            zone(i).BackColor = Color.Transparent
        Next i
        file.Close()
        Test_Area.Enabled = True
    Else
        Height() = 300
        Test_mode.BackColor = Color.Transparent
        Test_mode.Text = "Test mode OFF"
        Label2.Visible = False
        My.Computer.Audio.Stop()
        c = 0
        msg_text += " *TEST MODE OFF* Reset of Violations!" & vbNewLine
        file =
My.Computer.FileSystem.OpenTextFileWriter("KSSfile\logfile.txt", True)
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
file.WriteLine(tim + " *TEST MODE OFF* Reset of Violations!")
For i = 1 To 8
    lo(i) = 0
    zone(i).Checked() = False
    zone(i).BackColor = Color.Transparent
Next i
file.Close()
Test_Area.Enabled = False
End If
End Sub

Private Sub SmS_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles SmS_Timer.Tick
    sms_ti = sms_ti + 1
End Sub
Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
    zone_enable()
    sensor_enable()
    ti = ti + 1
    tim = System.DateTime.Now
    If Test_mode.Checked = False Then
        Out(&H378S + 2, 224)

        If ti Mod 3 Then
            reset_sub()
        End If

        x = Inp(&H378S)

        s8_zone()
        '=====
        s7_zone()
        '=====
        s6_zone()
        '=====
        s5_zone()
        '=====
        s4_zone()
        '=====
        s3_zone()
        '=====
        s2_zone()
        '=====
        s1_zone()

        If ZONE_A.Checked = True Or ZONE_B.Checked = True Or ZONE_C.Checked =
True Or ZONE_D.Checked = True Or ZONE_E.Checked = True Or ZONE_F.Checked = True
Or ZONE_G.Checked = True Or ZONE_H.Checked = True Then
            Label2.Visible = True
            If ti Mod 2 = 0 Then
                My.Computer.Audio.Play(My.Resources.attention,
AudioPlayMode.Background)
            End If
        Else
            Label2.Visible = False
        End If

    Else
        Out(&H378S + 2, 192)
        If c = 1 Then
            Label2.Visible = True
            If ti Mod 2 = 0 Then
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
        My.Computer.Audio.Play(My.Resources.attention,
AudioPlayMode.Background)
    End If
Else
    Label2.Visible = False
End If
For j = 1 To 8
    For i = 1 To 8
        If n(j) = i Then
            If zone(i).Checked = True Then
                If ti Mod 2 = 0 Then
                    zone(i).BackColor = Color.Red
                Else
                    zone(i).BackColor = Color.Transparent
                End If
            End If
        End If
    End If
Next i
Next j
End If
If ti Mod 8 - 7 = 0 Then
    If Form2.Auto_Report.Checked = True Then
        report_Click(True, AcceptButton)
    End If
End If
End Sub

Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    Form2.ShowDialog()
    Height() = 300
    bu_named()
    che_named()
    ti = 0
    For i = 1 To 8
        AddHandler sb(i).Click, AddressOf s_Click
    Next i
    Timer1.Start()
End Sub
End Class
```

Κώδικας φόρμας ρυθμίσεων

```
Public Class Form2
    Public c(8), lts As Short
    Private se(8, 8) As CheckBox
    Public tse(8) As ComboBox
    Public nu_z(8) As String

    Private Sub se_named()
        tse(1) = ComboBox1
        tse(2) = ComboBox2
        tse(3) = ComboBox3
        tse(4) = ComboBox4
        tse(5) = ComboBox5
        tse(6) = ComboBox6
        tse(7) = ComboBox7
        tse(8) = ComboBox8

        se(1, 1) = As1
        se(2, 1) = As2
        se(3, 1) = As3
        se(4, 1) = As4
        se(5, 1) = As5
        se(6, 1) = As6
        se(7, 1) = As7
        se(8, 1) = As8

        se(1, 2) = Bs1
        se(2, 2) = Bs2
        se(3, 2) = Bs3
        se(4, 2) = Bs4
        se(5, 2) = Bs5
        se(6, 2) = Bs6
        se(7, 2) = Bs7
        se(8, 2) = Bs8

        se(1, 3) = Cs1
        se(2, 3) = Cs2
        se(3, 3) = Cs3
        se(4, 3) = Cs4
        se(5, 3) = Cs5
        se(6, 3) = Cs6
        se(7, 3) = Cs7
        se(8, 3) = Cs8

        se(1, 4) = Ds1
        se(2, 4) = Ds2
        se(3, 4) = Ds3
        se(4, 4) = Ds4
        se(5, 4) = Ds5
        se(6, 4) = Ds6
        se(7, 4) = Ds7
        se(8, 4) = Ds8

        se(1, 5) = Es1
        se(2, 5) = Es2
        se(3, 5) = Es3
        se(4, 5) = Es4
        se(5, 5) = Es5
    End Sub
End Class
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
se(6, 5) = Es6
se(7, 5) = Es7
se(8, 5) = Es8

se(1, 6) = Fs1
se(2, 6) = Fs2
se(3, 6) = Fs3
se(4, 6) = Fs4
se(5, 6) = Fs5
se(6, 6) = Fs6
se(7, 6) = Fs7
se(8, 6) = Fs8

se(1, 7) = Gs1
se(2, 7) = Gs2
se(3, 7) = Gs3
se(4, 7) = Gs4
se(5, 7) = Gs5
se(6, 7) = Gs6
se(7, 7) = Gs7
se(8, 7) = Gs8

se(1, 8) = Hs1
se(2, 8) = Hs2
se(3, 8) = Hs3
se(4, 8) = Hs4
se(5, 8) = Hs5
se(6, 8) = Hs6
se(7, 8) = Hs7
se(8, 8) = Hs8

nu_z(1) = "A"
nu_z(2) = "B"
nu_z(3) = "C"
nu_z(4) = "D"
nu_z(5) = "E"
nu_z(6) = "F"
nu_z(7) = "G"
nu_z(8) = "H"
End Sub

Private Sub CB_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Zone_num.SelectedIndexChanged
    For j = 1 To 8
        tse(j).SelectedIndex = -1
        For i = 1 To 8
            c(i) = 0
            Form1.n(i) = 0
            se(i, j).Checked = False
            se(i, j).Enabled = True
        Next i
    Next j
    Select Case Zone_num.SelectedIndex
        Case 0
            Az.Enabled = True
            Bz.Enabled = False
            Cz.Enabled = False
            Dz.Enabled = False
            Ez.Enabled = False
            Fz.Enabled = False
            Gz.Enabled = False
            Hz.Enabled = False
        Case 1
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
Az.Enabled = True
Bz.Enabled = True
Cz.Enabled = False
Dz.Enabled = False
Ez.Enabled = False
Fz.Enabled = False
Gz.Enabled = False
Hz.Enabled = False
Case 2
Az.Enabled = True
Bz.Enabled = True
Cz.Enabled = True
Dz.Enabled = False
Ez.Enabled = False
Fz.Enabled = False
Gz.Enabled = False
Hz.Enabled = False
Case 3
Az.Enabled = True
Bz.Enabled = True
Cz.Enabled = True
Dz.Enabled = True
Ez.Enabled = False
Fz.Enabled = False
Gz.Enabled = False
Hz.Enabled = False
Case 4
Az.Enabled = True
Bz.Enabled = True
Cz.Enabled = True
Dz.Enabled = True
Ez.Enabled = True
Fz.Enabled = False
Gz.Enabled = False
Hz.Enabled = False
Case 5
Az.Enabled = True
Bz.Enabled = True
Cz.Enabled = True
Dz.Enabled = True
Ez.Enabled = True
Fz.Enabled = True
Gz.Enabled = False
Hz.Enabled = False
Case 6
Az.Enabled = True
Bz.Enabled = True
Cz.Enabled = True
Dz.Enabled = True
Ez.Enabled = True
Fz.Enabled = True
Gz.Enabled = True
Hz.Enabled = False
Case 7
Az.Enabled = True
Bz.Enabled = True
Cz.Enabled = True
Dz.Enabled = True
Ez.Enabled = True
Fz.Enabled = True
Gz.Enabled = True
Hz.Enabled = True
End Select
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
End Sub

Private Sub Checked_changed(ByVal sender As System.Object, ByVal e As
System.EventArgs)
    If sender.checked = True Then
        For j = 1 To 8
            For i = 1 To 8
                If se(i, j).Checked = True And se(i, j) Is sender Then
                    c(j) = c(j) + 1
                    Form1.n(i) = j
                    tse(i).Enabled = True
                    For z = 1 To 8
                        If se(i, z) Is sender Then
                            se(i, z).Enabled = True
                        Else
                            se(i, z).Enabled = False
                        End If
                    Next z
                End If
            Next i
        Next j
    Else
        For j = 1 To 8
            For i = 1 To 8
                If se(i, j) Is sender Then
                    Form1.n(i) = 0
                    c(j) = c(j) - 1
                    tse(i).Enabled = False
                    For z = 1 To 8
                        se(i, z).Enabled = True
                    Next z
                End If
            Next i
        Next j
    End If
End Sub

Private Sub L_types_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles L_types.Click
    Load_sensors.ShowDialog()
End Sub

Private Sub Load_sensors_FileOk(ByVal sender As System.Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles Load_sensors.FileOk
    Dim rdr As System.IO.StreamReader
    For i = 1 To 8
        tse(i).Items.Clear()
        rdr =
My.Computer.FileSystem.OpenTextFileReader(Load_sensors.FileName)
        While Not rdr.EndOfStream
            Dim line As String = rdr.ReadLine()
            tse(i).Items.Add(line)
        End While
        rdr.Close()
    Next i
End Sub

Private Sub sms_report_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles sms_report.CheckedChanged
    If sms_report.Checked = True Then
        Auto_Report.Enabled = True
        Phone.Enabled = True
        Phone_text.Enabled = True
    End If
End Sub
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
        Phone_text.Focus()
    ElseIf Mail_report.Checked = False Then
        Auto_Report.Enabled = False
        Phone_text.Text = ""
        Phone.Enabled = False
        Phone_text.Enabled = False
    Else
        Phone_text.Text = ""
        Phone.Enabled = False
        Phone_text.Enabled = False
    End If
End Sub
Private Sub Mail_report_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles Mail_report.CheckedChanged
    If Mail_report.Checked = True Then
        Auto_Report.Enabled = True
        Mail_Area.Enabled = True
        Host_text.Focus()
    ElseIf sms_report.Checked = False Then
        Auto_Report.Enabled = False
        Mail_Area.Enabled = False
        Host_text.Text = ""
        Email_text.Text = ""
        Pass_text.Text = ""
    Else
        Mail_Area.Enabled = False
        Host_text.Text = ""
        Email_text.Text = ""
        Pass_text.Text = ""
    End If
End Sub

Dim Hash As New Security.Cryptography.MD5CryptoServiceProvider
Dim Des As New Security.Cryptography.TripleDESCryptoServiceProvider

Private Function encryptString(ByVal s As String) As String
    Des.Key =
Hash.ComputeHash(System.Text.Encoding.ASCII.GetBytes("encoderkey"))
    Des.Mode = Security.Cryptography.CipherMode.ECB
    Dim DESEncrypter As System.Security.Cryptography.ICryptoTransform =
Des.CreateEncryptor
    Dim ByteString() As Byte = System.Text.Encoding.ASCII.GetBytes(s)
    Dim enString As String =
Convert.ToBase64String(DESEncrypter.TransformFinalBlock(ByteString, 0,
ByteString.Length))
    Return enString
End Function
Private Function decryptString(ByVal st As String) As String
    Des.Key =
Hash.ComputeHash(System.Text.Encoding.ASCII.GetBytes("encoderkey"))
    Des.Mode = Security.Cryptography.CipherMode.ECB
    Dim DESDecrypter As System.Security.Cryptography.ICryptoTransform =
Des.CreateDecryptor
    Dim ByteString() As Byte = Convert.FromBase64String(st)
    Dim deString As String =
System.Text.Encoding.ASCII.GetString(DESDecrypter.TransformFinalBlock(ByteString,
0, ByteString.Length))
    Return deString
End Function

Private Sub Save_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Save.Click
    Save_settings.ShowDialog()
```


Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
End Sub
Private Sub Save_settings_FileOk(ByVal sender As System.Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles Save_settings.FileOk
    Dim sfile As System.IO.StreamWriter
    sfile =
My.Computer.FileSystem.OpenTextFileWriter(Save_settings.FileName(), True)

    sfile.WriteLine(Zone_num.SelectedIndex)
    For i = 1 To 8
        sfile.WriteLine(c(i))
        sfile.WriteLine(Form1.n(i))
    Next i
    sfile.WriteLine(Load_sensors.FileName)
    For j = 1 To 8
        sfile.WriteLine(tse(j).SelectedItem)
        For i = 1 To 8
            If se(i, j).Checked = True Then
                sfile.WriteLine(1)
            Else
                sfile.WriteLine(0)
            End If
        Next i
    Next j
    If sms_report.Checked = True Then
        sfile.WriteLine(1)
    Else
        sfile.WriteLine(0)
    End If
    sfile.WriteLine(Phone_text.Text)
    If Mail_report.Checked = True Then
        sfile.WriteLine(1)
    Else
        sfile.WriteLine(0)
    End If
    sfile.WriteLine(Host_text.Text)
    sfile.WriteLine(Email_text.Text)
    sfile.WriteLine(encryptString(Pass_text.Text))
    If Auto_Report.Checked = True Then
        sfile.WriteLine(1)
    Else
        sfile.WriteLine(0)
    End If
    sfile.Close()
End Sub

Private Sub Load_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Load.Click
    Load_settings.ShowDialog()
End Sub
Private Sub Load_settings_FileOk(ByVal sender As System.Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles Load_settings.FileOk
    Dim lfile As System.IO.StreamReader
    lfile =
My.Computer.FileSystem.OpenTextFileReader(Load_settings.FileName())

    Zone_num.SelectedIndex = lfile.ReadLine()
    For i = 1 To 8
        c(i) = lfile.ReadLine()
        Form1.n(i) = lfile.ReadLine()
    Next i
    Load_sensors.FileName = lfile.ReadLine()
    Load_settings_FileOk(Load_sensors, e)
    For j = 1 To 8
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
tse(j).SelectedItem = lfile.ReadLine()
For i = 1 To 8
    If lfile.ReadLine() = 1 Then
        se(i, j).Checked = True
    Else
        se(i, j).Checked = False
    End If
Next i
Next j

If lfile.ReadLine() = 1 Then
    sms_report.Checked = True
Else
    sms_report.Checked = False
End If
Phone_text.Text = lfile.ReadLine()
If lfile.ReadLine() = 1 Then
    Mail_report.Checked = True
Else
    Mail_report.Checked = False
End If
Host_text.Text = lfile.ReadLine()
Email_text.Text = lfile.ReadLine()
Pass_text.Text = decryptString(lfile.ReadLine())
If lfile.ReadLine() = 1 Then
    Auto_Report.Checked = True
Else
    Auto_Report.Checked = False
End If
lfile.Close()
End Sub

Private Sub Ok_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Ok.Click
    Form1.phone = Phone_text.Text

    Form1.host = Host_text.Text
    Form1.email = Email_text.Text
    Form1.pass = Pass_text.Text

    Dim q As Short
    For i = 0 To Zone_num.SelectedIndex + 1
        If c(i) > 0 Then
            q = q + 1
        End If
    Next i
    If Zone_num.SelectedIndex = -1 Then
        MessageBox.Show("Chose how many zones you want!")
        Zone_num.Focus()
        GoTo break
    ElseIf q <> Zone_num.SelectedIndex + 1 Then
        For j = 0 To 7
            If Zone_num.SelectedIndex = j Then
                For i = 0 To j
                    If c(i + 1) = 0 Then
                        MessageBox.Show("Chose sensors for zone " + nu_z(i +
1))

                                GoTo break
                            End If
                        Next i
                    End If
                Next j
            Else
```

Πτυχιακή εργασία του φοιτητή Καρδάρ Γεώργιου

```
        For i = 1 To 8
            If tse(i).Enabled = True And tse(i).SelectedItem = "" Then
                MessageBox.Show("Select the type of sensor " + CStr(i))
                GoTo break
            End If
        Next i
    End If
    If sms_report.Checked = True And Phone_text.Text = "" Then
        MessageBox.Show("Insert a phone number for sms report!")
        Phone_text.Focus()
    ElseIf Mail_report.Checked = False Then
        Me.Close()
    End If
    If Mail_report.Checked = True Then
        If Host_text.Text = "" Then
            MessageBox.Show("Insert your hostname for mail report!")
            Host_text.Focus()
        ElseIf Email_text.Text = "" Then
            MessageBox.Show("Insert your E-mail for mail report!")
            Email_text.Focus()
        ElseIf Pass_text.Text = "" Then
            MessageBox.Show("Insert your password for mail report!")
            Pass_text.Focus()
        Else
            Me.Close()
        End If
    End If
break:
    End Sub

    Private Sub Cancel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Cancel.Click
        Form1.Close()
    End Sub

    Private Sub Form2_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

        se_named()
        For j = 1 To 8
            For i = 1 To 8
                AddHandler se(i, j).CheckedChanged, AddressOf Checked_changed
            Next i
        Next j

    End Sub

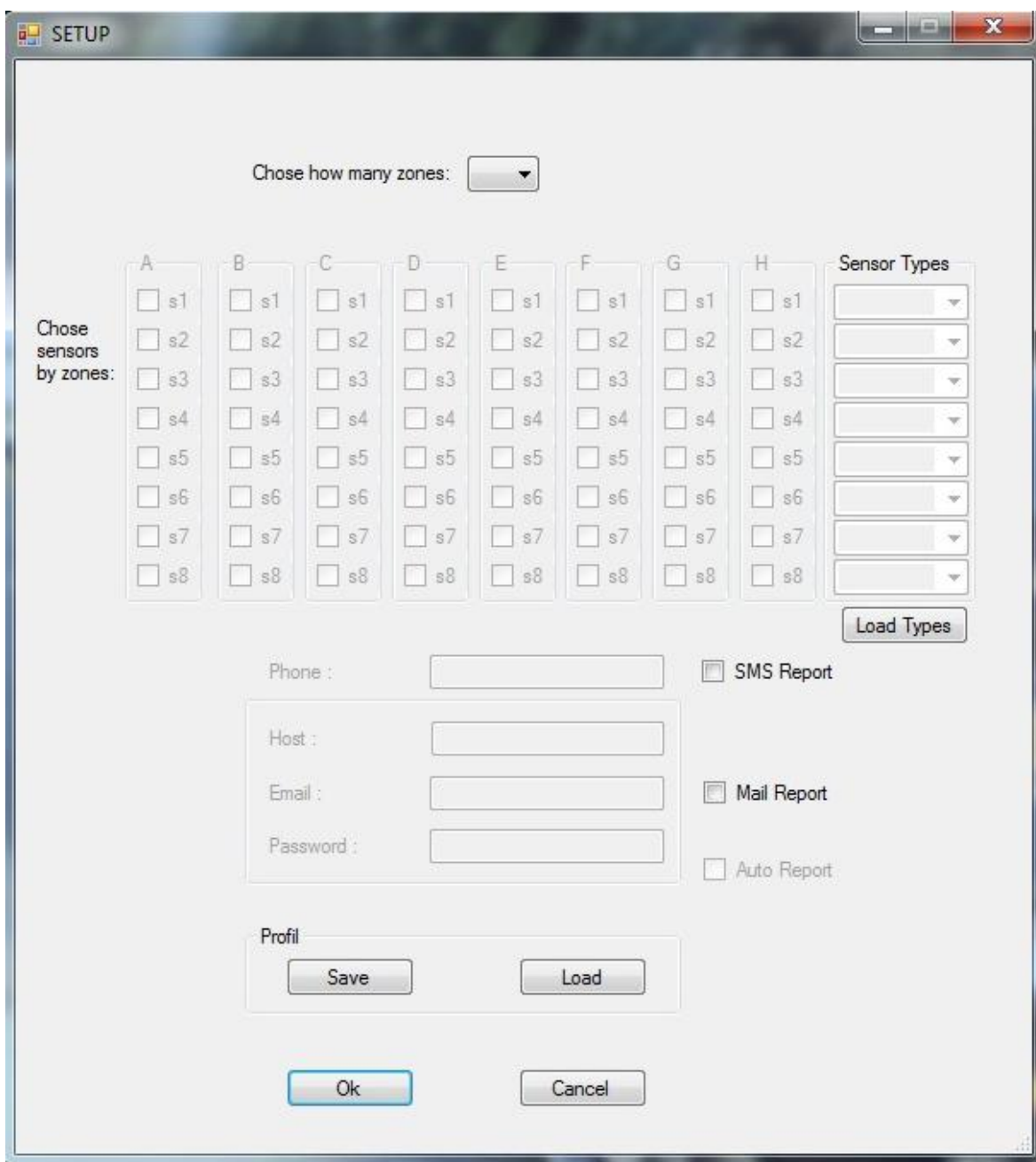
End Class
```

ΟΔΗΓΟΣ ΧΡΗΣΗΣ ΛΟΓΙΣΜΙΚΟΥ

Ο Η/Υ θα πρέπει να έχει επεξεργαστή τουλάχιστον 1.0 GHz και μνήμη RAM 265MB και άνω. Επίσης θα πρέπει να έχουμε εγκαταστήσει το .NET Frameworks v2.0 ή νεότερο, το οποίο μπορούμε να κατεβάσουμε από την ιστοσελίδα της Microsoft δωρεάν (<http://www.microsoft.com/net/>). Κάνουμε εγκατάσταση το πρόγραμμα KardarasSS και εμφανίζονται στην Επιφάνεια Εργασίας μας τρία εικονίδια:

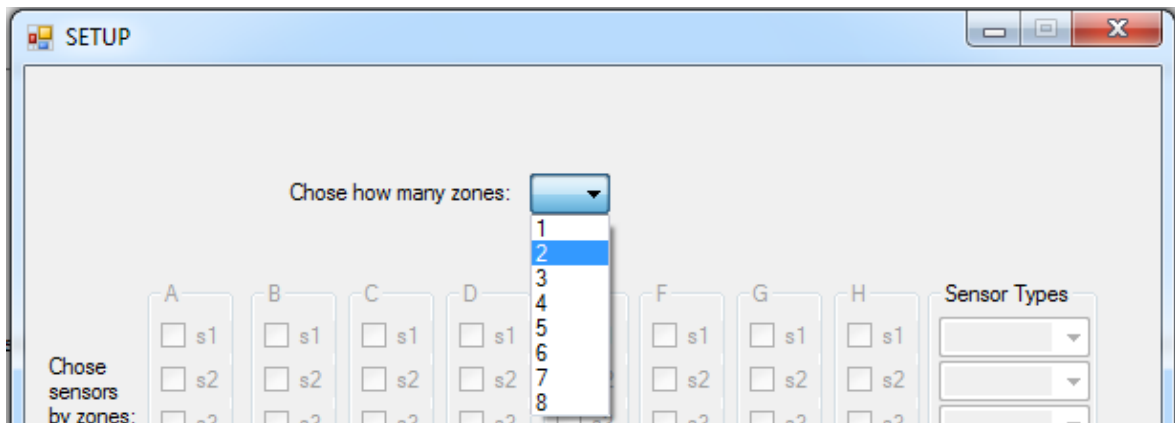
1. το KardarasSS.exe, το πρόγραμμα,
2. ένας φάκελος με όνομα KSSfile, ο οποίος περιέχει δύο έγγραφα κειμένου το logfile.txt, στο οποίο καταγράφεται το ιστορικό χρήση του προγράμματος (ώρα, ημερομηνία, παραβίαση ή μη αισθητηρίου, τύπος αισθητηρίου και ζώνη) και το sensors.txt, το οποίο περιέχει είδη αισθητηρίων και μπορούμε να το τροποποιήσουμε ανάλογα με τις ανάγκες μας,
3. και το inrout32.dll, το οποίο περιέχει τις μεθόδους επικοινωνίας με την παράλληλη θύρα και θα πρέπει να το αντιγράψουμε στον φάκελο System32 των Windows .

Ανοίγουμε το KardarasSS.exe και μας εμφανίζεται η φόρμα "SETUP", στην οποία παραμετροποιούμε το πρόγραμμά μας(Εικόνα 39).



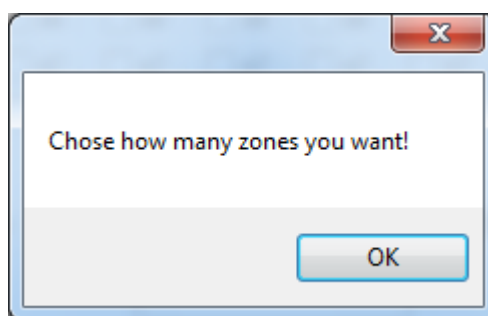
Εικόνα 39: Φόρμα ρυθμίσεων("SETUP")

Αρχικά επιλέγουμε πόσες ζώνες θέλουμε να έχει το σύστημά μας (Εικόνα 40) ,



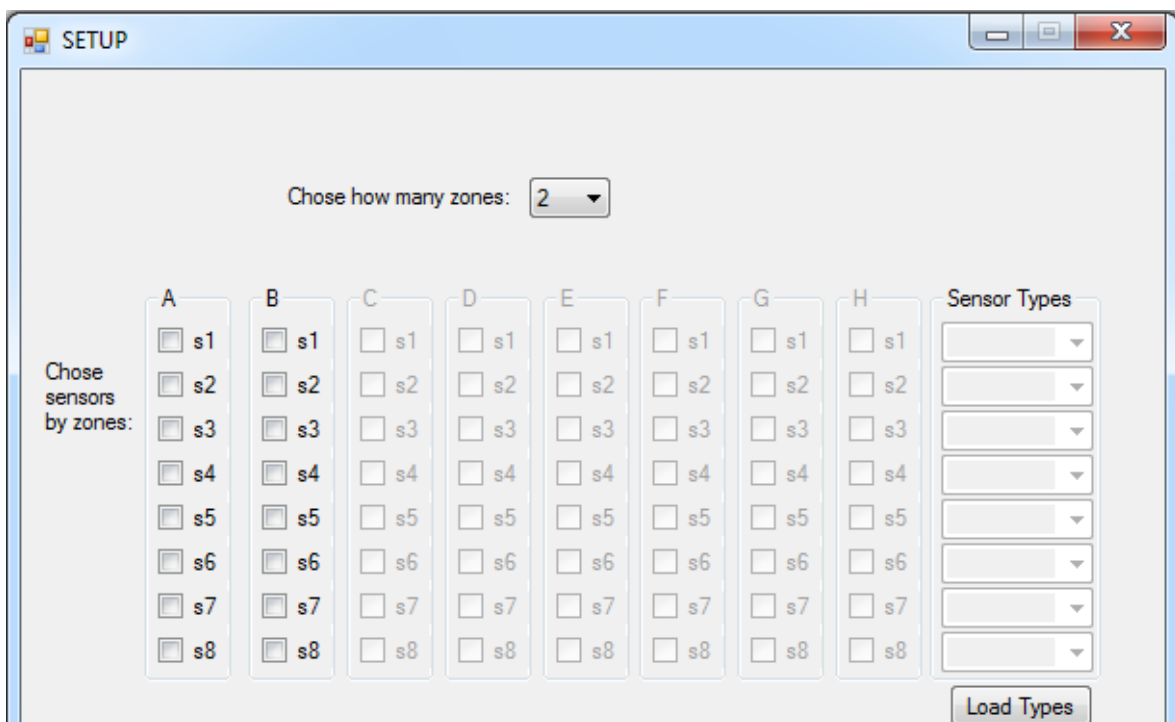
Εικόνα 40: Επιλογή αριθμού ζωνών

εάν προσπαθήσουμε να συνεχίσουμε χωρίς να επιλέξουμε τότε εμφανίζεται το παρακάτω παράθυρο μηνύματος:



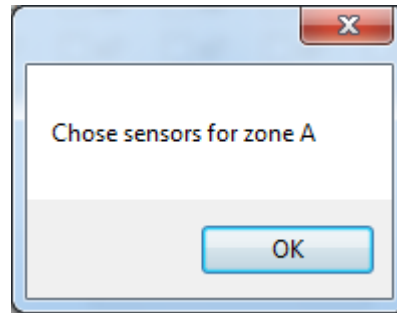
Εικόνα 41: Παράθυρο υπενθύμισης επιλογής ζωνών

Αφού έχουμε επιλέξει πόσες ζώνες θέλουμε βλέπουμε ότι έχουν ενεργοποιηθεί οι αντίστοιχες ζώνες με τις επιλογές των αισθητηρίων που θα τις ενεργοποιούν(Εικόνα 42).



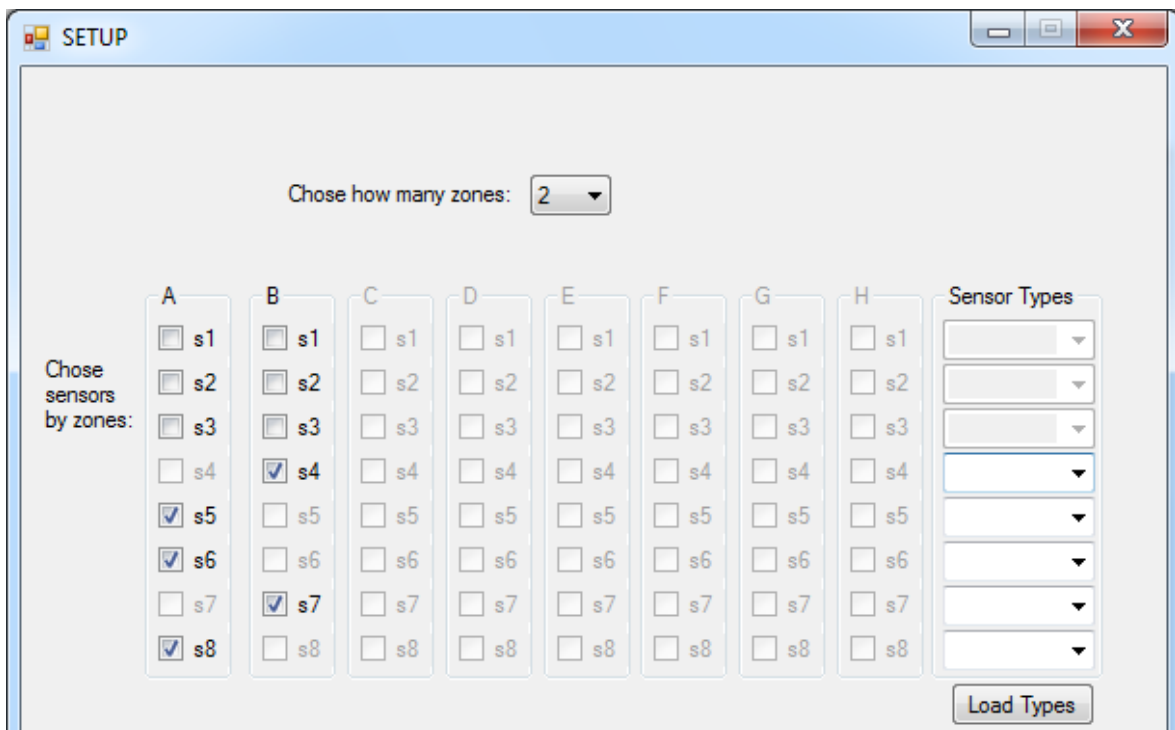
Εικόνα 42:Επιλογή αισθητηρίων

Υπάρχει και εδώ παράθυρο ειδοποίησης σε περίπτωση που δεν έχουμε επιλέξει αισθητήρια για κάποια ενεργοποιημένη ζώνη(π.χ. η Α):



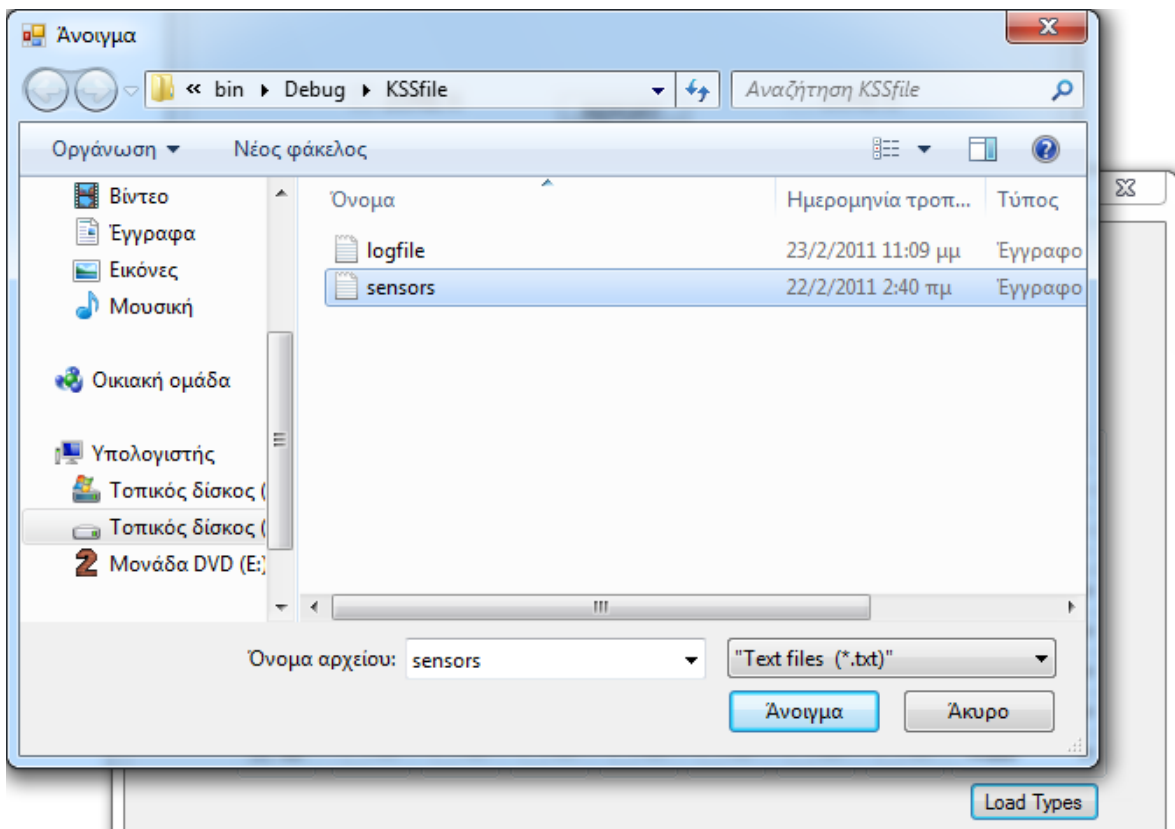
Εικόνα 43: Παράθυρο υπενθύμισης επιλογής αισθητήριων

Στην συνέχεια επιλέγοντας κάποιο αισθητήριο βλέπουμε ότι απενεργοποιείται από τις άλλες ζώνες, αφού δεν γίνεται ένα αισθητήριο να είναι σε δυο διαφορετικές ζώνες, ενώ παράλληλα ενεργοποιείται το αντίστοιχο “drop down box” από την λίστα “Sensor Types” αριστερά(Εικόνα 44).



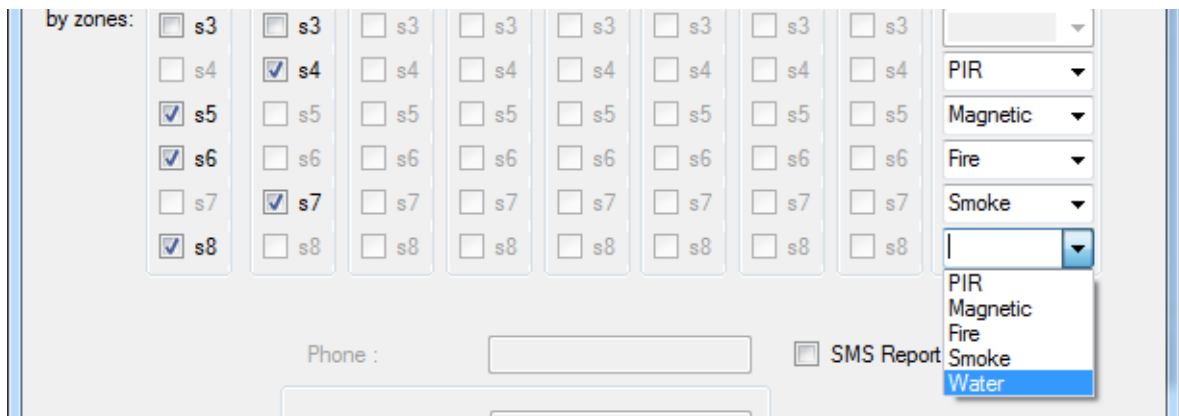
Εικόνα 44:Ενεργοποίηση drop down box ειδών αισθητήριων

Πατώντας το κουμπί “Load Types”, που βρίσκεται κάτω από την λίστα “Sensor Types” ανοίγει παράθυρο διαλόγου για να φορτώσουμε το έγγραφο κειμένου που περιέχει τους τύπους αισθητήριων που έχουμε ορίσει, οι οποίοι περνούν ως επιλογές στα “drop down box” από την λίστα “Sensor Types”(Εικόνα 45).



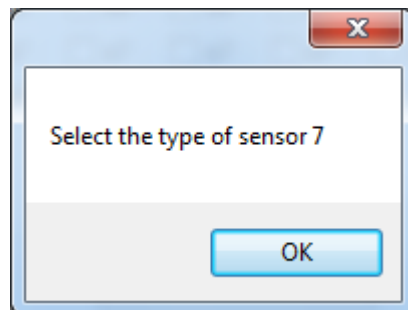
Εικόνα 45:Παράθυρο φόρτωσης αρχείου ειδών αισθητήριων

Έχοντας φορτώσει το έγγραφο κειμένου που περιέχει τους τύπους αισθητηρίων επιλέγουμε τον αντίστοιχο τύπο που θέλουμε για κάθε ενεργό αισθητήρα (Εικόνα 46).



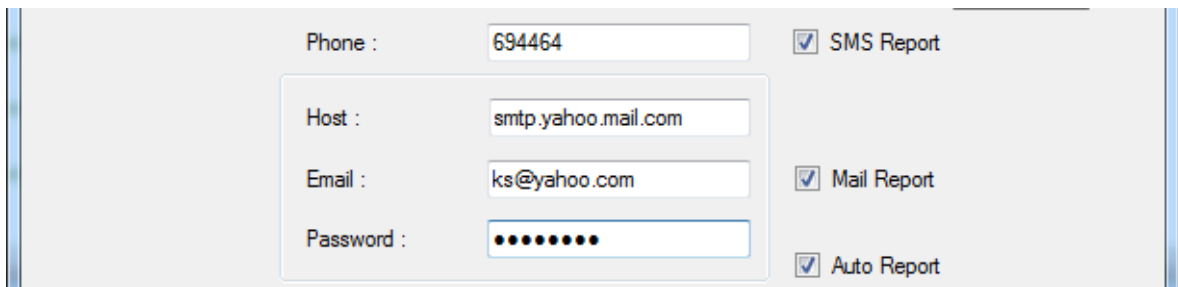
Εικόνα 46: Επιλογή είδους αισθητηρίου

Σε περίπτωση μη επιλογής τύπου για κάποιο ενεργό αισθητήριο τότε εμφανίζεται παράθυρο ειδοποίησης για τον αντίστοιχο αισθητήριο(π.χ. το 7):



Εικόνα 47: Παράθυρο υπενθύμισης επιλογής είδους αισθητηρίου

Συνεχίζοντας βλέπουμε ότι υπάρχουν δύο “Checkboxes” τα “SMS Report” και “Mail Report”, που αν τα επιλέξουμε ενεργοποιούν τα “Textboxes” με τίτλο “Phone” και “Host”, “Email”, “Password” αντίστοιχα, όπου καταχωρούμε το τηλέφωνο και τα στοιχεία της ηλεκτρονικής διεύθυνσης που θέλουμε να ειδοποιηθούμε με sms ή mail αντίστοιχα. Επίσης επιλέγοντας κάποιο από αυτά τα “Checkboxes” ενεργοποιείται το “Checkbox” “Auto Report” με το οποίο ρυθμίζουμε αν θέλουμε αυτόματη ειδοποίηση ή όχι (Εικόνα 48).

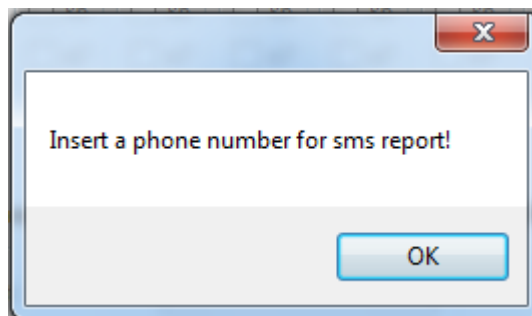


Phone :	<input type="text" value="694464"/>	<input checked="" type="checkbox"/> SMS Report
Host :	<input type="text" value="smtp.yahoo.mail.com"/>	
Email :	<input type="text" value="ks@yahoo.com"/>	<input checked="" type="checkbox"/> Mail Report
Password :	<input type="password" value="••••••••"/>	<input checked="" type="checkbox"/> Auto Report

Εικόνα 48: Επιλογή τρόπου ειδοποίησης και εισαγωγή στοιχείων

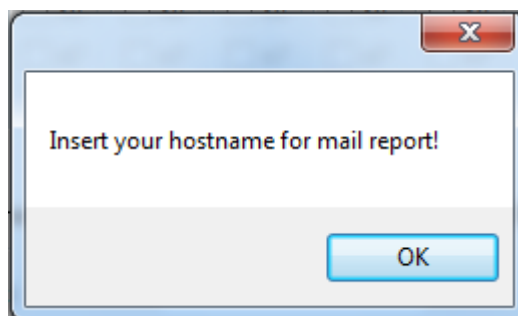
Η επιλογή κάποιου τρόπου ειδοποίησης τηλεφωνικής ή ηλεκτρονικής και η μη εισαγωγή στοιχείων εμφανίζει τα αντίστοιχα παράθυρα ειδοποίησης:

- Για την μη εισαγωγή τηλεφωνικού αριθμού.



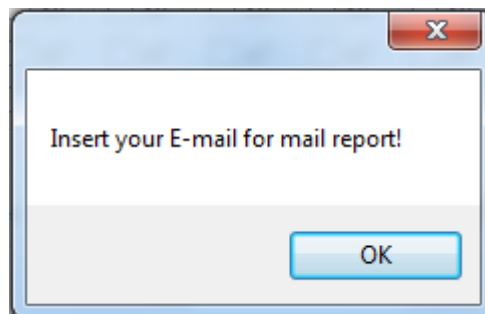
Εικόνα 49:Παράθυρο υπενθύμισης εισαγωγής τηλεφωνικού αριθμού

- Για την μη εισαγωγή "hostname" του ηλεκτρονικού ταχυδρομείου.



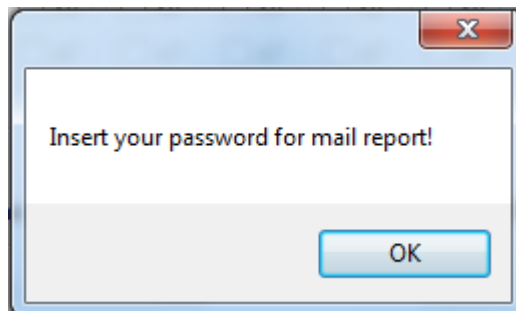
Εικόνα 50:Παράθυρο υπενθύμισης εισαγωγής hostname

- Για την μη εισαγωγή ηλεκτρονικής διεύθυνσης.



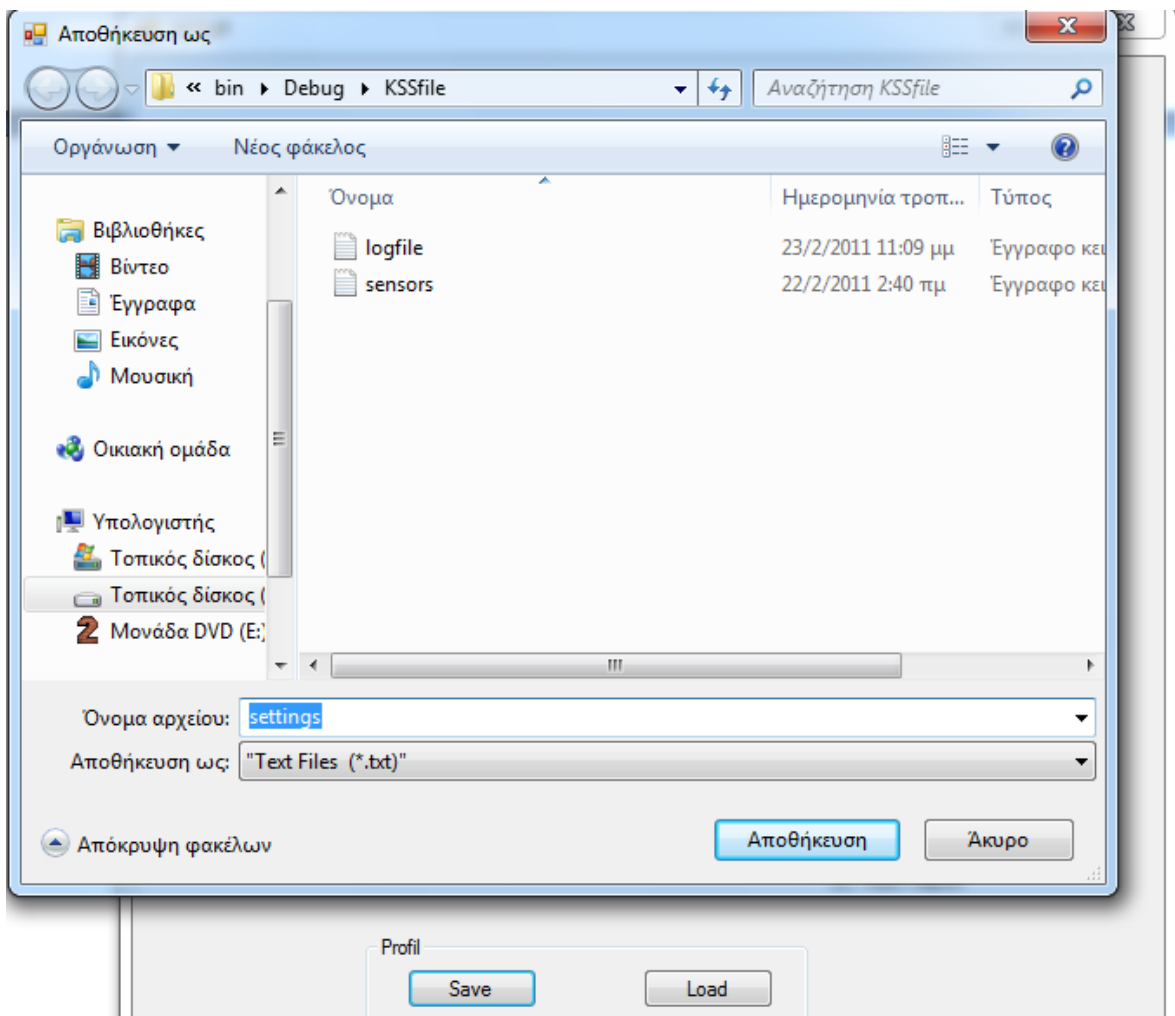
Εικόνα 51:Παράθυρο υπενθύμισης εισαγωγής ηλεκτρονικής διεύθυνσης

- Για την μη εισαγωγή κωδικού ηλεκτρονικής διεύθυνσης.



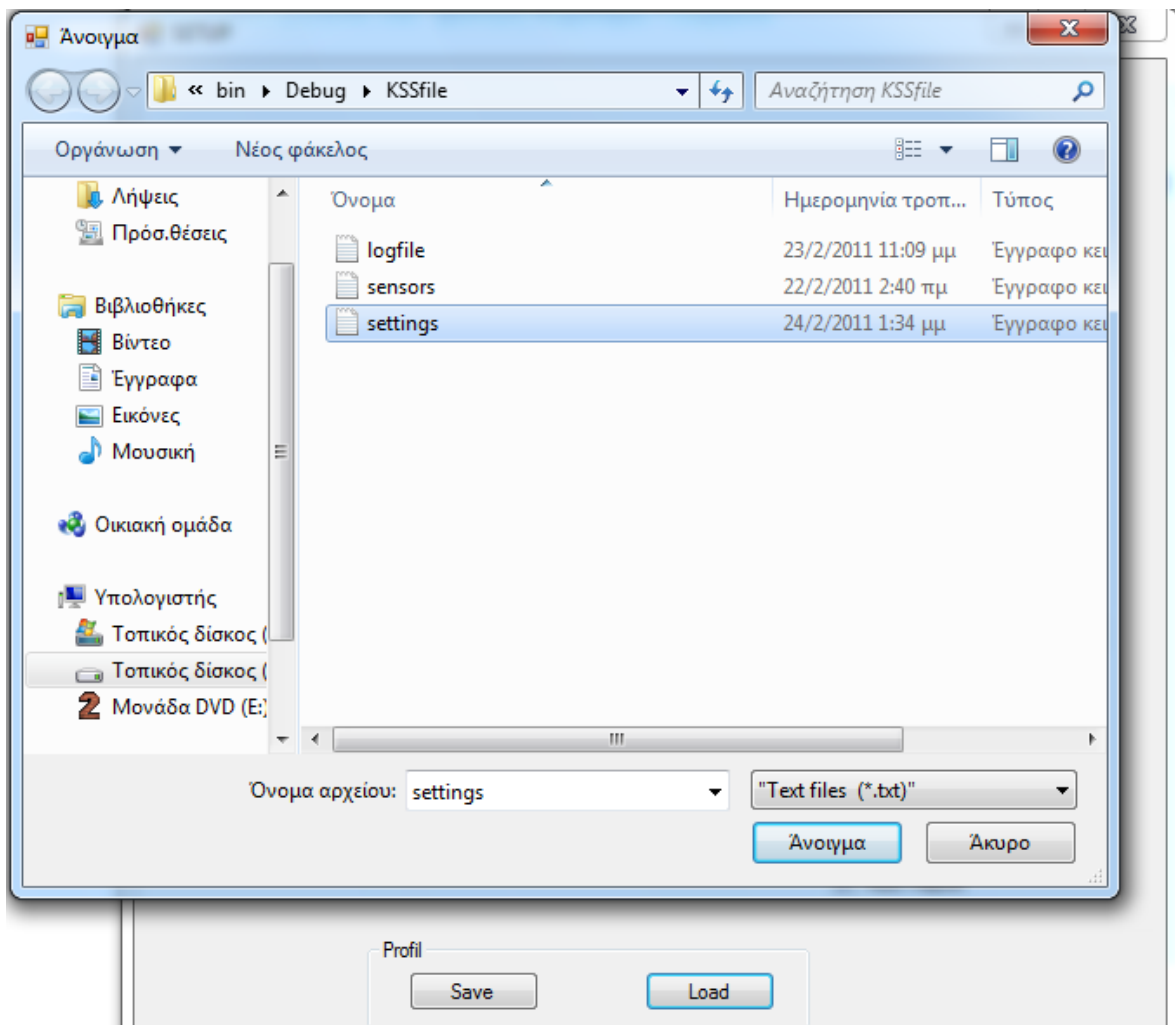
Εικόνα 52:Παράθυρο υπενθύμισης εισαγωγής κωδικού

Στην συνέχεια υπάρχουν δύο κουμπιά “Save” και “Load” στο “Groupbox” “Profil”. Πατώντας το κουμπί “Save” εμφανίζεται παράθυρο διαλόγου που μας δίνει την δυνατότητα να σώσουμε τις επιλογές μας σε ένα έγγραφο κειμένου (Εικόνα 53).



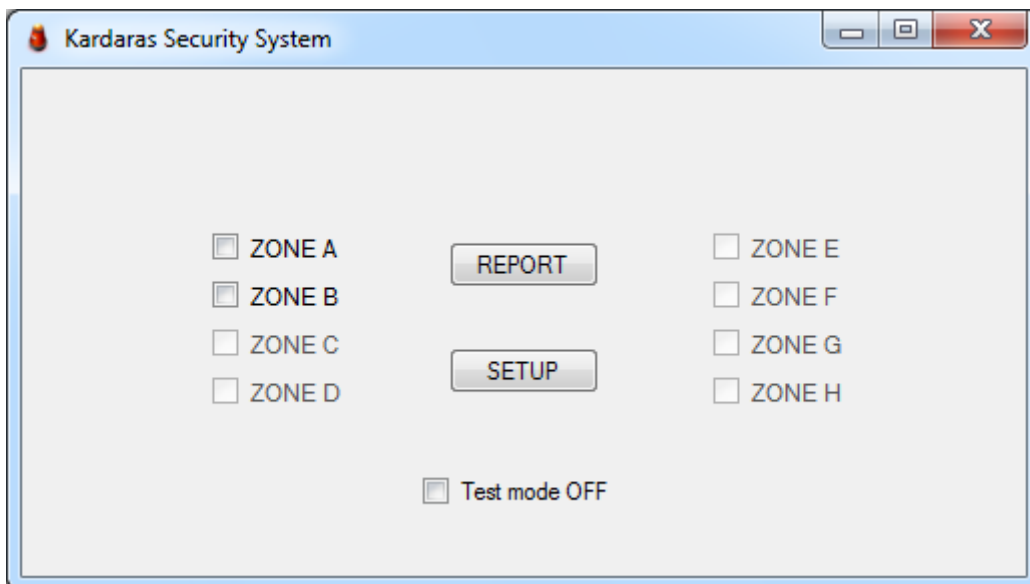
Εικόνα 53:Παράθυρο αποθήκευσης ρυθμίσεων

Πατώντας το κουμπί “Load” εμφανίζεται παράθυρο διαλόγου που μας δίνει την δυνατότητα να φορτώσουμε ένα έγγραφο κειμένου με επιλογές που έχουμε αποθηκεύσει στο παρελθόν (Εικόνα 54).



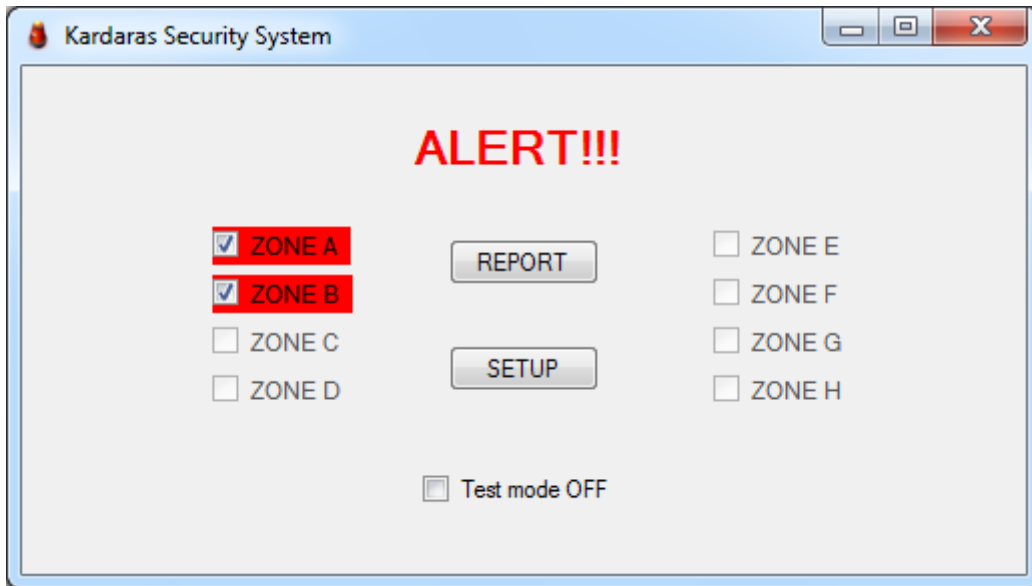
Εικόνα 54: Παράθυρο φόρτωσης ρυθμίσεων

Τέλος πιέζοντας το κουμπί “Οκ” εμφανίζεται η κυρίως φόρμα με όνομα “Kardaras Security System”, στην οποία είναι ενεργοποιημένα τα “Checkboxes” που αντιστοιχούν στις ενεργές ζώνες. Όταν δεν έχει παραβιαστεί κάποιο αισθητήριο που έχουμε επιλέξει και αντιστοιχίσει σε ενεργή ζώνη τότε το παράθυρο μας έχει την παρακάτω μορφή(Εικόνα 55).



Εικόνα 55:Παράθυρο κυρίας φόρμας

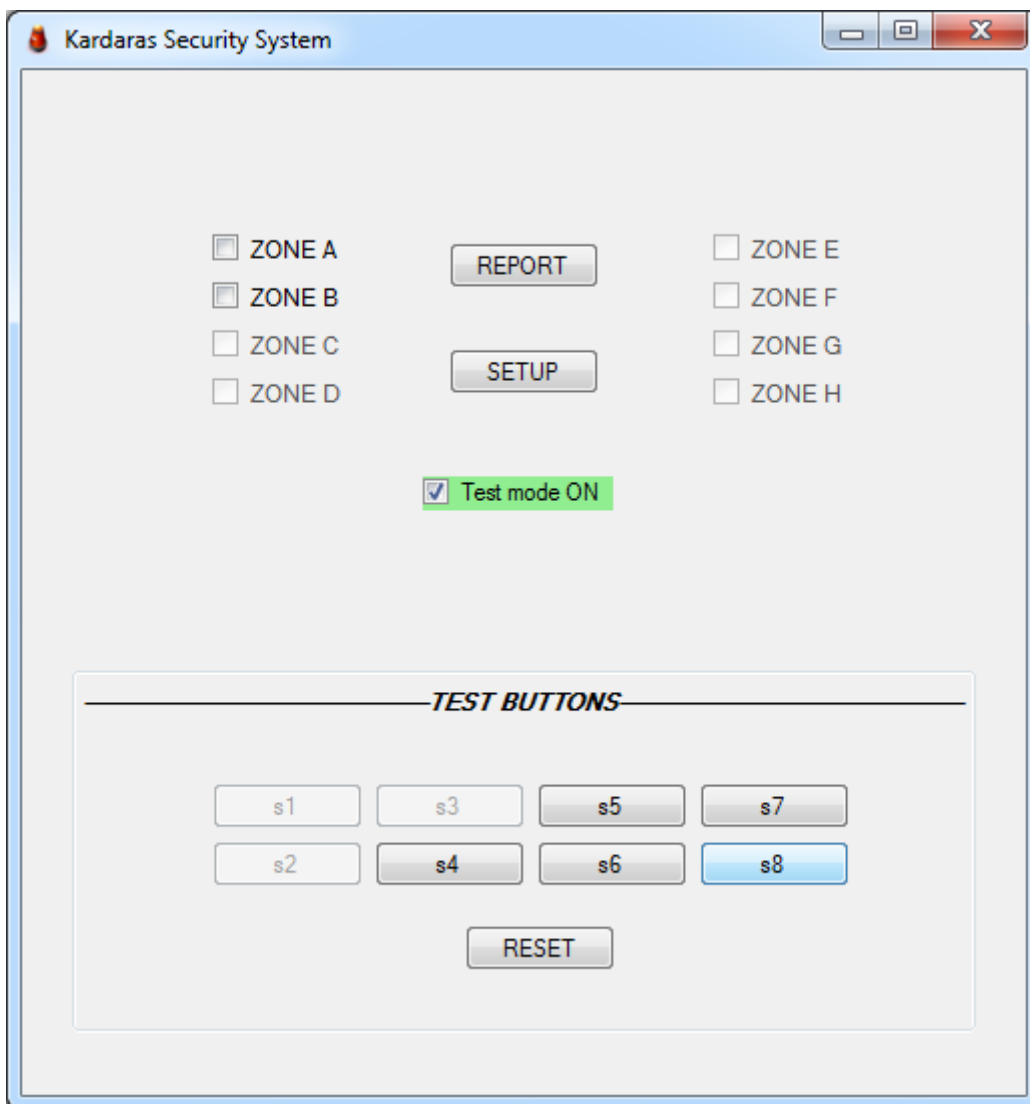
Μόλις παραβιαστεί κάποιο ενεργό αισθητήριο εμφανίζεται με κόκκινα γράμματα στο επάνω και κεντρικό μέρος του παραθύρου το μήνυμα “ALERT!!!” και αρχίζει να αναβοσβήνει κόκκινο χρώμα στο φόντο του “Checkbox” της αντίστοιχης ζώνης που ανήκει το παραβιασμένο αισθητήριο, ηχώντας περιοδικά “Attention”(Εικόνα 56).



Εικόνα 56: Παραβίαση ζωνών και ενεργοποίηση συναγερμού.

Κεντρικά υπάρχουν δυο κουμπιά τα "REPORT" και "SETUP". Πιέζοντας το κουμπί "REPORT" αποστέλλεται sms ή mail ανάλογα με τις ρυθμίσεις μας ενώ πιέζοντας το κουμπί "SETUP" εμφανίζεται η αρχική φόρμα "SETUP" με τις ρυθμίσεις μας.

Τέλος υπάρχει και ένα "Checkbox" "Test mode OFF" οποίο όταν επιλεγεί και γίνει "Test mode ON" με πράσινο φόντο, η εφαρμογή μας τίθεται σε κατάσταση δοκιμής, το παράθυρο μας μεγαλώνει και εμφανίζεται ένα "Groupbox" που αποτελείται από κουμπιά που αντιστοιχούν στα αισθητήρια μας και είναι ενεργοποιημένα τα αντίστοιχα και το κουμπί "RESET". Πατώντας κάποιο κουμπί αισθητηρίου ενεργοποιείται η ζώνη που ανήκει το αισθητήριο αυτό ενώ πατώντας το "RESET" απενεργοποιεί όλες οι ενεργές ζώνες (Εικόνα 57).



Εικόνα 57:Ενεργοποίηση Test Mode και εμφάνιση Test buttons.