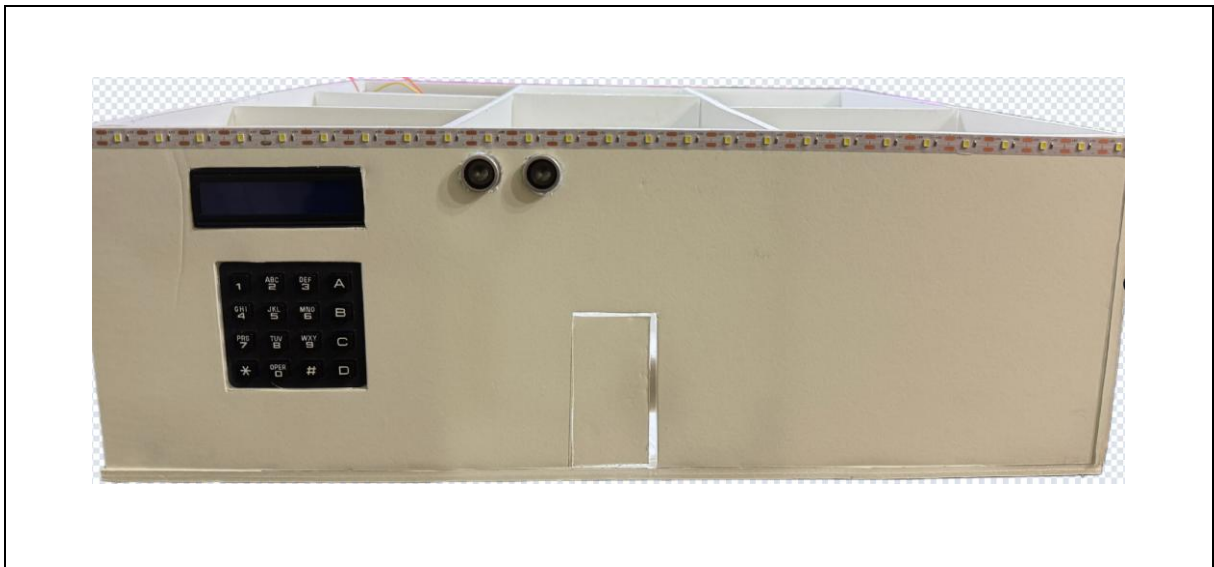


ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ  
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ  
«ΟΙΚΙΑΚΟΣ ΣΥΝΑΓΕΡΜΟΣ ΜΕ ΧΡΗΣΗ  
ARDUINO»



Του φοιτητή  
Αρβανιτάκη Χαράλαμπου  
Αρ. Μητρώου: 518013

Επιβλέπων  
Δρ. Τσιακμάκης Κυριάκος

Ημερομηνία 23/12/2024

## ΟΙΚΙΑΚΟΣ ΣΥΝΑΓΕΡΜΟΣ ΜΕ ΧΡΗΣΗ ARDUINO

Κωδικός Π.Ε. 24220

Όνοματεπώνυμο φοιτητή ΑΡΒΑΝΙΤΑΚΗΣ ΧΑΡΑΛΑΜΠΟΣ

Όνοματεπώνυμο εισηγητή Τσιακμάκης Κυριάκος

Ημερομηνία ανάληψης Π.Ε. 24/8/2024

Ημερομηνία περάτωσης Π.Ε. 24/1/2025

*Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως πτυχιακή εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.*

*Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Αρβανιτάκη Χαράλαμπου που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.*

Η έγκριση της πτυχιακής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

*«Στην οικογένειά μου»*



## Πρόλογος

Ο τίτλος της παρούσας πτυχιακής εργασίας είναι "Οικιακός συναγερμός με χρήση Arduino" και πραγματεύεται την ανάπτυξη ενός σύγχρονου συστήματος ασφαλείας για οικιακές χρήσεις. Σκοπός της εργασίας είναι η σχεδίαση και υλοποίηση ενός οικονομικά προσιτού, εύχρηστου και αποτελεσματικού συστήματος συναγερμού, το οποίο βασίζεται στην πλατφόρμα ανοιχτού κώδικα Arduino. Τα συστήματα ασφαλείας αποτελούν πλέον κομμάτι της καθημερινότητας, προσφέροντας προστασία από απρόσκλητους επισκέπτες και συμβάλλοντας στην αύξηση του αισθήματος ασφάλειας σε κάθε χώρο.

Το Arduino, λόγω της ευελιξίας και της απλότητάς του, προσφέρει μια ιδανική βάση για την ανάπτυξη ενός τέτοιου συστήματος, καθώς επιτρέπει τη σύνδεση αισθητήρων κίνησης, σειρήνων, και άλλων απαραίτητων εξαρτημάτων για τη λειτουργία ενός συναγερμού. Στην εργασία αυτή θα παρουσιαστούν όλα τα στάδια της κατασκευής, από τον σχεδιασμό του κυκλώματος και την επιλογή των κατάλληλων αισθητήρων, έως την προγραμματιστική υλοποίηση του συστήματος. Η εργασία καταδεικνύει πως η χρήση σύγχρονων, ανοιχτών τεχνολογιών μπορεί να προσφέρει ασφαλείς και προσιτές λύσεις για κάθε σπίτι.

## Περίληψη

Αυτή η πτυχιακή εργασία ασχολείται με τη δημιουργία ενός ολοκληρωμένου συστήματος ασφαλείας για οικιακή χρήση, το οποίο βασίζεται στην πλατφόρμα Arduino. Ο σκοπός του συστήματος είναι να παρέχει ολοκληρωμένη προστασία μέσω ανίχνευσης κινδύνων και παραβιάσεων, ενώ επιτρέπει την απομακρυσμένη παρακολούθηση και έλεγχο μέσω της πλατφόρμας Blynk. Για την υλοποίηση του συστήματος χρησιμοποιήθηκε ένα Arduino Mega ως κεντρική μονάδα επεξεργασίας, το οποίο συντονίζει την επικοινωνία μεταξύ των αισθητήρων και των εξαρτημάτων του συστήματος. Επίσης, ένα ESP-01 παρέχει δυνατότητα ασύρματης σύνδεσης στο διαδίκτυο, επιτρέποντας την απομακρυσμένη πρόσβαση και διαχείριση του συστήματος μέσω της εφαρμογής Blynk. Το σύστημα χρησιμοποιεί διάφορους αισθητήρες για την ανίχνευση πιθανών απειλών. Ο PIR sensor ανιχνεύει κίνηση στον χώρο, ενώ ο knock sensor ανιχνεύει δονήσεις που μπορεί να προκληθούν από απόπειρες διάρρηξης. Ο ultrasonic sensor μετρά την απόσταση και μπορεί να εντοπίζει την παρουσία αντικειμένων ή ανθρώπων σε προκαθορισμένη περιοχή. Ο DHT22 sensor μετρά τη θερμοκρασία και την υγρασία, επιτρέποντας τον έλεγχο των περιβαλλοντικών συνθηκών στο χώρο. Η πλατφόρμα Blynk χρησιμοποιείται για τον απομακρυσμένο έλεγχο και την παρακολούθηση του συστήματος μέσω κινητής συσκευής, καθιστώντας εύκολη τη διαχείριση του συναγερμού σε πραγματικό χρόνο. Η εργασία καταδεικνύει την αποτελεσματικότητα της συνδυασμένης χρήσης του Arduino με την τεχνολογία IoT για την κατασκευή ενός προσιτού και αποδοτικού συστήματος οικιακής ασφάλειας, το οποίο μπορεί να ενισχύσει το επίπεδο προστασίας σε οποιονδήποτε χώρο.

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ

Συναγερμός, σύστημα ασφαλείας, Arduino, Blynk, ESP-01

# «Home Security System with Arduino»

«Charalampos Arvanitakis»

## **Abstract**

This thesis focuses on the design and creation of an integrated security system for home use, based on the Arduino platform. The system provides comprehensive protection through danger and intrusion detection, while allowing remote monitoring and control through the Blynk platform. To implement the system, an Arduino Mega was used as the central processing unit, which synchronizes the communication between sensors and system components. Also, an ESP-01 provides wireless internet connectivity, allowing remote access and management of the system via the Blynk application. The system uses various sensors to detect potential threats. The PIR sensor detects movement in the room, while the knock sensor detects vibrations that can be caused by burglary attempts. The ultrasonic sensor measures distance and can detect the presence of objects or people in a predefined area. The DHT22 sensor measures temperature and humidity, allowing control of the environmental conditions in the room. The Blynk platform is used to remotely control and check the system via a mobile device, making it easy to manage the alarm in real time. The paper shows the effectiveness of combining Arduino with IoT technology to build an affordable and efficient home security system that can enhance the level of protection in any space.

## **KEY WORDS**

Alarm, Security systems, Arduino, Blynk, ESP-01

## **Ευχαριστίες**

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή μου κ. Τσιακμάκη Κυριάκο, για την καθοδήγηση, την υποστήριξη και τις πολύτιμες συμβουλές του καθ' όλη τη διάρκεια της πτυχιακής μου εργασίας. Επίσης, ευχαριστώ τους συμφοιτητές μου για τη συνεργασία και την αλληλοϋποστήριξη που είχαμε, καθώς και την οικογένειά μου για την αδιάκοπη συμπαράσταση και ενθάρρυνση. Τέλος, εκφράζω την ευγνωμοσύνη μου σε όλους όσους συνέβαλαν με οποιονδήποτε τρόπο στην ολοκλήρωση αυτής της εργασίας.

# Περιεχόμενα

Πρόλογος.....	v
Περίληψη.....	vi
Abstract .....	vii
Ευχαριστίες .....	viii
Περιεχόμενα .....	ix
Κατάλογος Σχημάτων .....	xiii
Κατάλογος Πινάκων.....	xiv
Συντομογραφίες.....	xvi
Κεφάλαιο 1ο: Συστήματα ασφαλείας .....	1
1.1 Εισαγωγή.....	1
1.2 Ηλεκτρονικά συστήματα ασφαλείας.....	1
1.2.1 Κατηγορίες συστημάτων ασφαλείας .....	1
1.2.2 Συστήματα συναγερμού.....	1
1.2.3 Συστήματα πυρανίχνευσης.....	2
1.2.3.1 Συμβατικά συστήματα πυρανίχνευσης .....	2
1.2.3.2 Διευθυνσιοδοτούμενα συστήματα πυρανίχνευσης .....	2
1.2.4 Συστήματα παρακολούθησης.....	3
1.2.4.1 Βασικά συστατικά ενός συστήματος CCTV.....	3
1.2.5 Συστήματα ελέγχου πρόσβασης .....	4
1.2.5.1 Βασικά συστατικά ενός συστήματος ελέγχου πρόσβασης .....	4
1.3 Επίλογος.....	4
Κεφάλαιο 2ο: Συστήματα συναγερμού.....	5
2.1 Εισαγωγή.....	5
2.2 Αρχή λειτουργίας .....	5
2.3 Βασικές κατηγορίες συστημάτων συναγερμού .....	6
2.4 Δομή συστήματος συναγερμού .....	6
2.4.1 Κεντρική μονάδα ελέγχου .....	7
2.4.2 Πληκτρολόγιο.....	7
2.4.3 Ανιχνευτές κίνησης .....	8
2.4.4 Σειρήνες.....	9
Κεφάλαιο 3ο: Arduino .....	10

3.1	Εισαγωγή.....	10
3.2	Εφαρμογές Arduino.....	10
3.3	Πλακέτες Arduino .....	10
3.3.1	Arduino UNO .....	10
3.3.2	Arduino NANO .....	11
3.3.3	Arduino MEGA 2560 .....	11
3.3.4	Arduino MICRO.....	12
3.3.5	Arduino Leonardo.....	12
Κεφάλαιο 4ο:	Εφαρμογή Blynk.....	14
4.1	Εισαγωγή.....	14
4.2	Ορισμός.....	14
4.3	Χαρακτηριστικά.....	14
Κεφάλαιο 5ο:	Σχεδίαση και υλοποίηση εργασίας.....	15
5.1	Εισαγωγή.....	15
5.2	Σχεδίαση κατασκευής.....	15
5.3	Υλοποίηση κατασκευής .....	15
5.3.1	Οθόνη .....	15
5.3.1.1	Συσκευή διεπαφής I2C.....	15
5.3.2	Πληκτρολόγιο.....	16
5.3.2.1	Αρχή λειτουργίας πληκτρολογίου.....	17
5.3.2.2	Συνδεσμολογία με το Arduino .....	17
5.3.3	ESP8266-01 .....	17
5.3.3.1	Χαρακτηριστικά ESP-01 .....	18
5.3.3.2	Συνδεσμολογία με το Arduino .....	19
5.3.4	Απομακρυσμένος έλεγχος Arduino .....	19
5.3.4.1	Επικοινωνία Blynk και Arduino .....	19
5.3.4.2	Αποστολή ειδοποιήσεων.....	20
5.3.5	Μαγνητική επαφή.....	21
5.3.5.1	Συνδεσμολογία μαγνητικής επαφής.....	21
5.3.5.2	Συνδεσμολογία με το Arduino .....	22
5.3.6	Ανιχνευτής φωτιάς.....	22
5.3.6.1	Συνδεσμολογία με το Arduino .....	22
5.3.7	Αισθητήρας λέιζερ.....	23
5.3.7.1	Συνδεσμολογία με το Arduino .....	23

5.3.8	Δέκτης λέιζερ.....	24
5.3.8.1	Συνδεσμολογία με το Arduino .....	24
5.3.9	Ανιχνευτής κίνησης.....	24
5.3.9.1	Πλακέτα ανιχνευτή κίνησης (PIR) HC-SR501 .....	25
5.3.9.2	Τεχνικά χαρακτηριστικά (PIR) HC-SR501 .....	26
5.3.9.3	Συνδεσμολογία με το Arduino .....	26
5.3.10	Αισθητήρας υπερήχων .....	27
5.3.10.1	Λειτουργία αισθητήρα HC-SR04 .....	27
5.3.10.2	Τεχνικά χαρακτηριστικά HC-SR04 .....	28
5.3.11	Αισθητήρας κραδασμών.....	28
5.3.11.1	Συνδεσμολογία αισθητήρα κραδασμών με το Arduino .....	28
5.3.12	Αισθητήρας θερμοκρασίας και υγρασίας.....	28
5.3.12.1	Λειτουργία αισθητήρα DHT22 .....	29
5.3.12.2	Συνδεσμολογία DHT22 με το Arduino.....	31
5.3.13	Κατασκευή μακέτας.....	31
Κεφάλαιο 6ο:	Κώδικας .....	33
6.1	Εισαγωγή.....	33
6.2	Κώδικας συστήματος συναγερμού.....	33
6.3	Διάγραμμα ροής .....	33
6.4	Λειτουργία συστήματος συναγερμού.....	35
6.5	Σενάρια.....	37
6.5.1	Πρώτο σενάριο .....	37
6.5.2	Δεύτερο σενάριο.....	38
6.5.3	Τρίτο σενάριο .....	38
6.5.4	Τέταρτο σενάριο .....	39
6.5.5	Πέμπτο σενάριο .....	39
6.5.6	Έκτο σενάριο .....	40
6.6	Επίλογος.....	40
Κεφάλαιο 7ο:	Συμπεράσματα ή/και προτάσεις βελτίωσης.....	41
BIBΛΙΟΓΡΑΦΙΑ.....		43
ΠΑΡΑΡΤΗΜΑ Α: Κατάλογος υλικών.....		44
ΠΑΡΑΡΤΗΜΑ Β: Κώδικας .....		45



## Κατάλογος Σχημάτων

Σχήμα 1.1: Ενσύρματο/Ασύρματο σύστημα συναγερμού

Σχήμα 1.2: Συμβατικό σύστημα πυρανίχνευσης

Σχήμα 1.3: Διευθυνσιοδοτούμενο σύστημα πυρανίχνευσης

Σχήμα 1.4: Κλειστό κύκλωμα καμερών (CCTV)

Σχήμα 1.5: Σύστημα ελέγχου πρόσβασης

Σχήμα 2.1: Σύστημα συναγερμού

Σχήμα 2.2: Πίνακας συναγερμού Siemens Sintony ic60

Σχήμα 2.3: Πληκτρολόγιο Siemens Sintony IKP6-03

Σχήμα 2.4: Ανιχνευτής κίνησης υπερθύρων ADM-Q12 Siemens

Σχήμα 2.5: Ασύρματη σειράνα ISR6-12R Siemens

Σχήμα 3.1: Arduino LOGO

Σχήμα 3.2: Arduino UNO

Σχήμα 3.3: Arduino NANO

Σχήμα 3.4: Arduino MEGA 2560

Σχήμα 3.5: Arduino MICRO

Σχήμα 3.6: Arduino LEONARDO

Σχήμα 4.1: Αρχιτεκτονική Blynk

Σχήμα 5.1: 16x2 LCD display with I2C interface

Σχήμα 5.2: I2C Module

Σχήμα 5.3: Πληκτρολόγιο 4x4 matrix

Σχήμα 5.4: Εσωτερική διάταξη πληκτρολογίου

Σχήμα 5.5: ESP8266 Wi-Fi Module ESP-01

Σχήμα 5.6: Τεχνικά χαρακτηριστικά ESP-01

Σχήμα 5.7: Δημιουργία ψηφιακού ακροδέκτη

Σχήμα 5.8: Δημιουργία γραφικού στοιχείου (widget)

Σχήμα 5.9: Λήψη ειδοποιήσεων από το Blynk

Σχήμα 5.10: Κανονικά κλειστή μαγνητική επαφή

Σχήμα 5.11: Ανιχνευτής φωτιάς KY-026

Σχήμα 5.12: Πομπός λέιζερ

Σχήμα 5.13: Δέκτης λέιζερ

Σχήμα 5.14: Παθητικός ανιχνευτής υπέρυθρης ακτινοβολίας (PIR) HC-SR501

Σχήμα 5.15: Κάτω μέρος πλακέτας, επάνω μέρος πλακέτας  
Σχήμα 5.16: Σχηματικό (PIR) HC-SR501  
Σχήμα 5.17: Αισθητήρας υπερήχων HC-SR04  
Σχήμα 5.18: Λειτουργία HC-SR04  
Σχήμα 5.19: Αισθητήρας κραδασμών  
Σχήμα 5.20: Αισθητήρας θερμοκρασίας και υγρασίας DHT22  
Σχήμα 5.21: Διάγραμμα σχέσης μεταξύ θερμοκρασίας και υγρασίας, Θερμίστορ  
Σχήμα 5.22: Εσωτερική δομή του αισθητήρα υγρασίας DHT22  
Σχήμα 5.23: Μελέτη κάτοψης μακέτας  
Σχήμα 5.24: Κάτοψη μακέτας  
Σχήμα 6.1: Διάγραμμα ροής  
Σχήμα 6.2: Αρχικοποίηση συστήματος και έλεγχος Wi-Fi  
Σχήμα 6.3: Σύστημα έτοιμο, Ειδοποίηση Blynk  
Σχήμα 6.4: Αλλαγή κωδικού πρόσβασης  
Σχήμα 6.5: Ακύρωση αλλαγής κωδικού  
Σχήμα 6.6: Όπλιση και αφόπλιση συστήματος συναγερμού  
Σχήμα 6.7: Ενεργοποίηση αντίστροφης μέτρησης και ειδοποιήσεις  
Σχήμα 6.8: Ειδοποίηση φωτιάς  
Σχήμα 6.9: Ειδοποίηση υψηλής υγρασίας  
Σχήμα 6.10: Ειδοποίηση για κίνηση στο χώρο  
Σχήμα 6.11: Ειδοποίηση από διακοπή δέσμης λέιζερ  
Σχήμα 6.12: Ειδοποίηση για διάρρηξη παραθύρου

## **Κατάλογος Πινάκων**

Πίνακας 5.1: Σύνδεση LCD I2C οθόνης με arduino  
Πίνακας 5.2: Συνδεσμολογία ακίδων  
Πίνακας 5.3: Συνδεσμολογία με το Arduino  
Πίνακας 5.4: Συνδεσμολογία με το Arduino  
Πίνακας 5.5: Τεχνικά χαρακτηριστικά πομπού  
Πίνακας 5.6: Συνδεσμολογία πομπού λέιζερ  
Πίνακας 5.7: Συνδεσμολογία δέκτη λέιζερ  
Πίνακας 5.8: Τεχνικά χαρακτηριστικά (PIR) HC-SR501

Πίνακας 5.9: Συνδεσμολογία με το Arduino

Πίνακας 5.10: Τεχνικά χαρακτηριστικά αισθητήρα HC-SR04

Πίνακας 5.11: Συνδεσμολογία με το Arduino

Πίνακας 5.12: Τεχνικά χαρακτηριστικά DHT22

Πίνακας 5.13: Συνδεσμολογία με το Arduino

## Συντομογραφίες

ΚΜΕ	Κεντρική Μονάδα Ελέγχου
Η/Υ	Ηλεκτρονικός Υπολογιστής

# Εισαγωγή

Η ανάγκη για ασφάλεια αποτελεί μια από τις βασικότερες προτεραιότητες στη σύγχρονη κοινωνία. Τα συστήματα ασφαλείας, είτε για προσωπικούς χώρους είτε για επιχειρήσεις, έχουν εξελιχθεί ραγδαία τα τελευταία χρόνια, παρέχοντας περισσότερες δυνατότητες και αυτοματισμούς. Ταυτόχρονα, η ραγδαία ανάπτυξη της τεχνολογίας και η εξάπλωση των συσκευών χαμηλού κόστους, όπως οι πλατφόρμες ανοιχτού κώδικα τύπου Arduino, δίνουν τη δυνατότητα δημιουργίας έξυπνων και οικονομικά προσιτών συστημάτων ασφαλείας που μπορούν να ενσωματωθούν εύκολα στην καθημερινή ζωή.

Η παρούσα εργασία επικεντρώνεται στην ανάπτυξη ενός ολοκληρωμένου οικιακού συστήματος συναγερμού με τη χρήση της πλατφόρμας Arduino και συγκεκριμένα του μοντέλου Arduino Mega. Το σύστημα έχει ως κύριο στόχο την ανίχνευση παραβιάσεων ή άλλων ύποπτων ενεργειών σε έναν οικιακό χώρο, προσφέροντας έγκαιρες ειδοποιήσεις στον χρήστη μέσω της πλατφόρμας Blynk. Η δυνατότητα απομακρυσμένου ελέγχου και παρακολούθησης προσθέτει ένα σημαντικό πλεονέκτημα, καθώς επιτρέπει στον χρήστη να παρακολουθεί το σπίτι του από οποιαδήποτε τοποθεσία με πρόσβαση στο διαδίκτυο.

Στο πρώτο κεφάλαιο της εργασίας γίνεται η ανάλυση των συστημάτων ασφαλείας, τι ονομάζουμε σύστημα ασφαλείας και σε ποιες μεγάλες κατηγορίες χωρίζονται. Στη συνέχεια αναλύονται τα ηλεκτρονικά συστήματα ασφαλείας και αναλύονται οι κατηγορίες τους.

Στο δεύτερο κεφάλαιο γίνεται αναλυτική παρουσίαση του συστήματος συναγερμού. Αναφέρεται η δομή του συστήματος και οι πιο βασικοί ανιχνευτές μαζί με την αρχή λειτουργίας τους.

Στο τρίτο κεφάλαιο γίνεται μια σύντομη αναφορά στο Arduino, τί είναι το Arduino και ποιες είναι οι πιο δημοφιλείς πλακέτες Arduino.

Στο τέταρτο κεφάλαιο αναφέρεται σύντομα η πλατφόρμα Blynk, τί είναι το Blynk και ποια τα βασικά χαρακτηριστικά του.

Στο πέμπτο κεφάλαιο γίνεται αναλυτική περιγραφή στην σχεδίαση και υλοποίηση του συστήματος συναγερμού. Αναφέρονται όλα τα εξαρτήματα που χρησιμοποιήθηκαν για την δημιουργία του συστήματος, η συνεργασία με την πλατφόρμα Blynk καθώς και η υλοποίηση της μακέτας.

Το έκτο κεφάλαιο αφορά τον κώδικα του συστήματος και προγραμματίζουμε το Arduino MEGA με την χρήση του Arduino IDE και δίνεται το διάγραμμα ροής του κώδικα. Τέλος, παρουσιάζεται με εικόνες η λειτουργία του καθώς και τα σενάρια.

Στο έβδομο και τελευταίο κεφάλαιο παρουσιάζονται τα προβλήματα που προέκυψαν κατά την διάρκεια της πτυχιακής εργασίας αλλά και πώς αντιμετωπίστηκαν. Τέλος, παραχωρούνται τα συμπεράσματα και οι μελλοντικές βελτιώσεις που μπορούν να γίνουν στο σύστημα συναγερμού για οικιακή χρήση.

# Κεφάλαιο 1ο: Συστήματα ασφαλείας

## 1.1 Εισαγωγή

Τα συστήματα ασφαλείας διαδραματίζουν σημαντικό ρόλο στην προστασία ατόμων και περιουσιών. Η ραγδαία ανάπτυξη της τεχνολογίας έχει οδηγήσει στη δημιουργία καινοτόμων ηλεκτρονικών λύσεων, οι οποίες ανταποκρίνονται στις σύγχρονες προκλήσεις ασφαλείας. Στο παρόν κεφάλαιο, θα παρουσιαστούν οι κύριες κατηγορίες ηλεκτρονικών συστημάτων ασφαλείας, τα χαρακτηριστικά τους και τα κριτήρια επιλογής τους.

## 1.2 Ηλεκτρονικά συστήματα ασφαλείας

Ως ηλεκτρονικά συστήματα ασφαλείας ορίζονται οι τεχνολογικές υποδομές που στοχεύουν στην προστασία από διάρρηξη, ληστεία, πυρκαγιά ή άλλες απειλές. Τα συστήματα αυτά βασίζονται σε ηλεκτρονικά εξαρτήματα, αισθητήρες και δικτυακή συνδεσιμότητα, διασφαλίζοντας άμεση απόκριση σε περιστατικά.

### 1.2.1 Κατηγορίες συστημάτων ασφαλείας

- ❖ Συστήματα συναγερμού (Alarm Systems)
- ❖ Συστήματα πυρανίχνευσης (Fire Detection Systems)
- ❖ Συστήματα παρακολούθησης (CCTV)
- ❖ Συστήματα ελέγχου πρόσβασης (Access Control Systems)

### 1.2.2 Συστήματα συναγερμού

Τα συστήματα συναγερμού χρησιμοποιούνται για την προστασία από εισβολείς. Περιλαμβάνουν αισθητήρες κίνησης, μαγνητικές επαφές και σειρήνες και μπορεί να είναι ενσύρματα ή ασύρματα ανάλογα τις ανάγκες του χώρου. [11]



Σχήμα 1.1: Ενσύρματο/Ασύρματο σύστημα συναγερμού

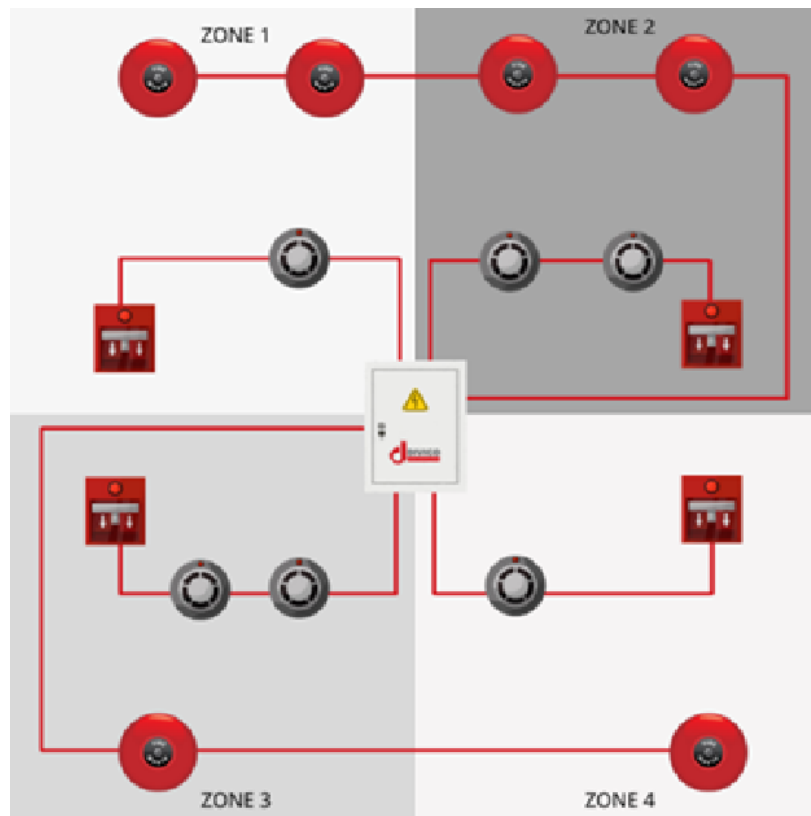
<https://sat-alarm.gr/shop/%ce%b1%cf%83%cf%8d%cf%81%ce%bc%ce%b1%cf%84%ce%bf%cf%82-%cf%83%cf%85%ce%bd%ce%b1%ce%b3%ce%b5%cf%81%ce%bc%cf%8c%cf%82-siemens-ic60-compact/>

### 1.2.3 Συστήματα πυρανίχνευσης

Τα συστήματα πυρανίχνευσης αποτελούνται από ομάδες συσκευών που είναι σχεδιασμένες για την ανίχνευση καπνού ή θερμότητας και περιλαμβάνουν αισθητήρες φωτιάς, καπνού, αυτοματισμούς απενεργοποίησης συσκευών και μηχανισμούς ειδοποίησης με σκοπό να ανιχνεύσουν έγκαιρα μία εστία φωτιάς και να δώσουν το σήμα κινδύνου με ηχητικά, οπτικά και άλλα μέσα.

#### 1.2.3.1 Συμβατικά συστήματα πυρανίχνευσης

Σε ένα συμβατικό σύστημα πυρανίχνευσης ο χώρος χωρίζεται σε ζώνες. Κάθε ζώνη αντιπροσωπεύεται από μια ομάδα ανιχνευτών όπως ανιχνευτές καπνού, θερμότητας ή φλόγας. Όταν ενεργοποιηθεί κάποιος ανιχνευτής, το σύστημα υποδεικνύει την ζώνη όπου εντοπίστηκε το συμβάν αλλά όχι το ακριβές σημείο. Είναι απλό στην εγκατάσταση και την χρήση του και χρησιμοποιείται κυρίως σε μικρές εγκαταστάσεις όπου η ακριβής τοποθεσία της πυρκαγιάς δεν είναι κρίσιμη. [12]

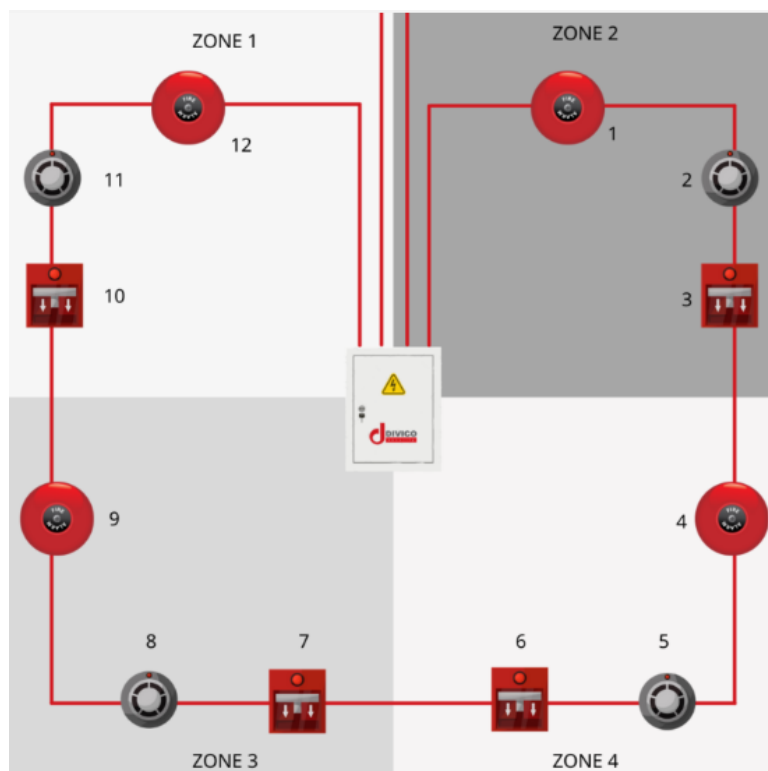


Σχήμα 1.2: Συμβατικό σύστημα πυρανίχνευσης

<https://divico.gr/en/products/fire-detection-systems/>

#### 1.2.3.2 Διευθυνσιοδοτούμενα συστήματα πυρανίχνευσης

Στα διευθυνσιοδοτούμενα συστήματα πυρανίχνευσης κάθε ανιχνευτής και συσκευή διαθέτει μοναδική διεύθυνση. Όταν ενεργοποιηθεί κάποιος ανιχνευτής, το σύστημα αναγνωρίζει την ακριβή του θέση στον χώρο παρέχοντας αναλυτική πληροφορία για το σημείο πυρκαγιάς. Μπορεί να διαχειριστεί μεγάλο αριθμό ανιχνευτών και παρέχει προηγμένες ρυθμίσεις, όπως καθορισμό ειδικών συναγερμών για συγκεκριμένους χώρους, το οποίο το κάνει κατάλληλο για μεγάλους χώρους ή κτίρια. [12]



Σχήμα 1.3: Διευθυνσιοδοτούμενο σύστημα πυρανίχνευσης

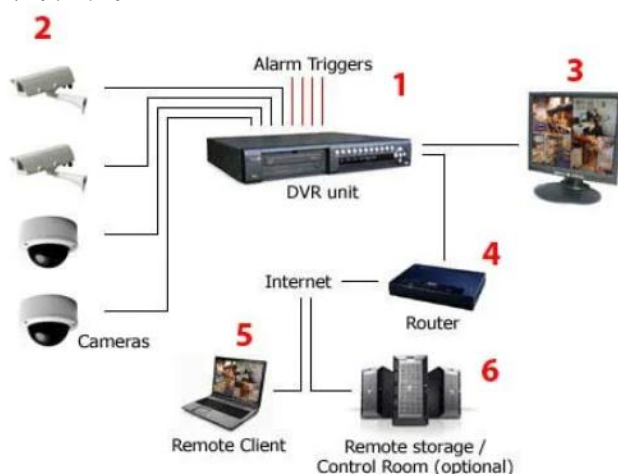
<https://divico.gr/en/products/fire-detection-systems/>

## 1.2.4 Συστήματα παρακολούθησης

Τα συστήματα παρακολούθησης CCTV (Closed-Circuit Television) χρησιμοποιούνται για την ασφάλεια και την παρακολούθηση χώρων, τόσο για οικιακή όσο και για επαγγελματική χρήση. Αυτά τα συστήματα περιλαμβάνουν κάμερες, καταγραφικά, καλώδια και λογισμικό, που συνεργάζονται για τη συνεχή καταγραφή εικόνας και ήχου σε πραγματικό χρόνο ή για αναπαραγωγή αργότερα.

### 1.2.4.1 Βασικά συστατικά ενός συστήματος CCTV

- ❖ Κάμερες
- ❖ Καταγραφικά DVR/NVR
- ❖ Οθόνες
- ❖ Λογισμικό και εφαρμογές



## Σχήμα 1.4: Κλειστό κύκλωμα καμερών (CCTV)

<https://chzafarhabib.wordpress.com/2013/06/21/professional-design-of-cctv-systems/>

### 1.2.5 Συστήματα ελέγχου πρόσβασης

Εκτός από την καταγραφή δεδομένων και την πρόσβαση σε εξουσιοδοτημένους χρήστες, τα συστήματα ελέγχου πρόσβασης είναι κατασκευασμένα για να προστατεύουν χώρους όπου τα μέλη του προσωπικού και πολλοί επισκέπτες κινούνται συχνά. Οι επιχειρήσεις, τα ξενοδοχεία, οι αποθήκες και τα νοσοκομεία είναι οι κύριες τοποθεσίες για τη χρήση τους, επιτρέποντας την ενισχυμένη ασφάλεια σε περιοχές που είναι ζωτικής σημασίας για τον οργανισμό. Επιπλέον, μπορεί να χρησιμεύσει ως κάρτα ελέγχου χρόνου εργασίας για τους υπαλλήλους.

#### 1.2.5.1 Βασικά συστατικά ενός συστήματος ελέγχου πρόσβασης

- ❖ Αναγνώστες κάρτας ή βιομετρικών στοιχείων
- ❖ Ελεγκτές
- ❖ Λογισμικό διαχείρισης
- ❖ Ηλεκτρονικές κλειδαριές και μηχανισμοί πόρτας
- ❖



Σχήμα 1.5: Σύστημα ελέγχου πρόσβασης

<http://www.homesec.gr/services/access-control-services/>

### 1.3 Επίλογος

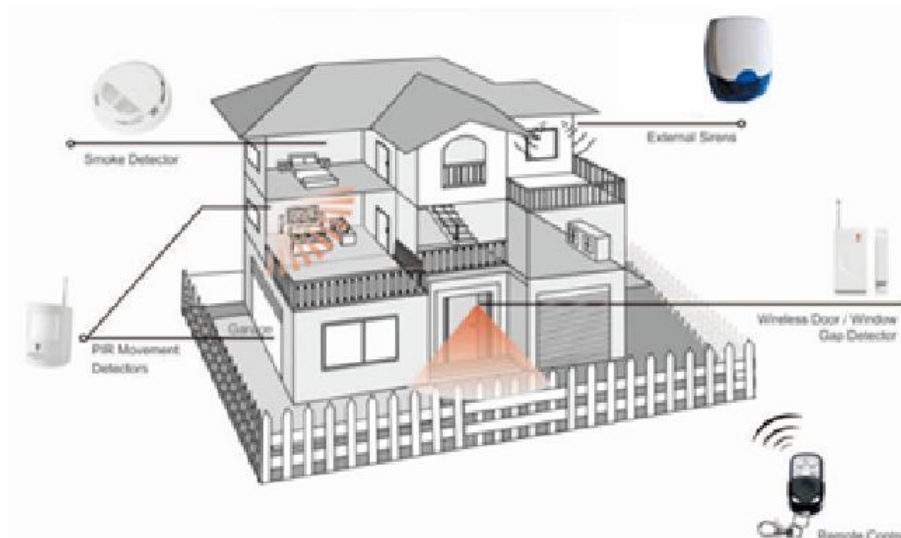
Τα συστήματα ασφαλείας συμβάλλουν στην ασφάλεια διαφόρων χώρων και περιβαλλόντων. Λόγω της τεχνολογικής προόδου, είναι δυνατή η δημιουργία μιας ολοκληρωμένης λύσης για κάθε είδους απειλή. Ο συνδυασμός αυτών των συστημάτων σε μια υποδομή ασφαλείας μπορεί να προσφέρει μια ολοκληρωμένη προστασία για τους ανθρώπους και την περιουσία και να δημιουργήσει ένα αποτέλεσμα όπου η ασφάλεια και η λειτουργικότητα συνεργάζονται.

## Κεφάλαιο 2ο: Συστήματα συναγερμού

### 2.1 Εισαγωγή

Τα συστήματα συναγερμών αποτελούν τεχνολογικές λύσεις που έχουν σχεδιαστεί για να ανιχνεύουν και να προειδοποιούν για πιθανές απειλές, όπως παραβιάσεις, κλοπές, πυρκαγιές ή άλλους κινδύνους σε κτίρια και χώρους. Βασική τους λειτουργία είναι η άμεση ανίχνευση ανεπιθύμητων ενεργειών ή αλλαγών στον χώρο και η ειδοποίηση των χρηστών μέσω ηχητικών σημάτων, φωτεινών ενδείξεων ή αποστολής ειδοποιήσεων σε κινητές συσκευές.

Τα συστήματα αυτά ενσωματώνουν αισθητήρες, όπως ανιχνευτές κίνησης, κραδασμών ή καπνού, που εντοπίζουν οποιαδήποτε ασυνήθιστη δραστηριότητα. Στη σύγχρονη εποχή, τα συστήματα συναγερμών έχουν εξελιχθεί με την εισαγωγή της τεχνολογίας IoT και τη δυνατότητα απομακρυσμένης διαχείρισης μέσω διαδικτύου, προσφέροντας αυξημένη ασφάλεια και ευκολία στη χρήση.



Σχήμα 2.1: Σύστημα συναγερμού

<https://alexandropoulos-systems.blogspot.com/2020/09/blog-post.html>

### 2.2 Αρχή λειτουργίας

Η αρχή λειτουργίας των συστημάτων συναγερμού βασίζεται στην ανίχνευση οποιωνδήποτε περιεργων δραστηριοτήτων ή συνθηκών που υποδηλώνουν απειλή, όπως εισβολή, πυρκαγιά και διαρροή αερίου. Ο εξοπλισμός αποτελείται από έναν κεντρικό πίνακα ελέγχου και έναν αριθμό αισθητήρων, οι οποίοι εγκαθίστανται σε στρατηγικά σημεία των εγκαταστάσεων. Αυτοί οι αισθητήρες ανιχνεύουν συγκεκριμένες δραστηριότητες όπως κίνηση, άνοιγμα πόρτας, ανίχνευση καπνού, δόνηση κ.λπ. και στέλνουν σήμα στον πίνακα ελέγχου. Μόλις ο κεντρικός πίνακας ελέγχου λάβει σήμα από έναν αισθητήρα, στέλνει μια απάντηση. Αυτή η απάντηση στέλνει μια ηχητική ειδοποίηση μέσω σειρήνας ή φώτων, ή μπορεί να στείλει μια ειδοποίηση στον χρήστη μέσω κινητού τηλεφώνου, ή μπορεί να στείλει μια ειδοποίηση σε μια υπηρεσία ασφαλείας. Τα συστήματα σε πολλές περιπτώσεις μπορούν να

συνδεθούν στο διαδίκτυο επιτρέποντας στον χρήστη να παρακολουθεί και να διαχειρίζεται εξ αποστάσεως μέσω εφαρμογών. Το σύστημα συναγερμού θα λειτουργήσει καλά μόνο εάν οι αισθητήρες λειτουργούν καλά, είναι σωστά εγκατεστημένοι και οι ειδοποιήσεις ενεργοποιούνται γρήγορα. Με αυτόν τον τρόπο, παρέχουν άμεση ανταπόκριση εάν υπάρχει οποιαδήποτε απειλή, ελαχιστοποιώντας έτσι τις πιθανότητες απώλειας περιουσίας ή ζημιών. [11]

### 2.3 Βασικές κατηγορίες συστημάτων συναγερμού

- ❖ Συστήματα ενσύρματου συναγερμού
- ❖ Συστήματα ασύρματου συναγερμού
- ❖ Υβριδικά συστήματα συναγερμού
- ❖ Συστήματα αυτοματοποιημένου σπιτιού

### 2.4 Δομή συστήματος συναγερμού

Τα βασικά στοιχεία ενός συστήματος συναγερμού είναι:

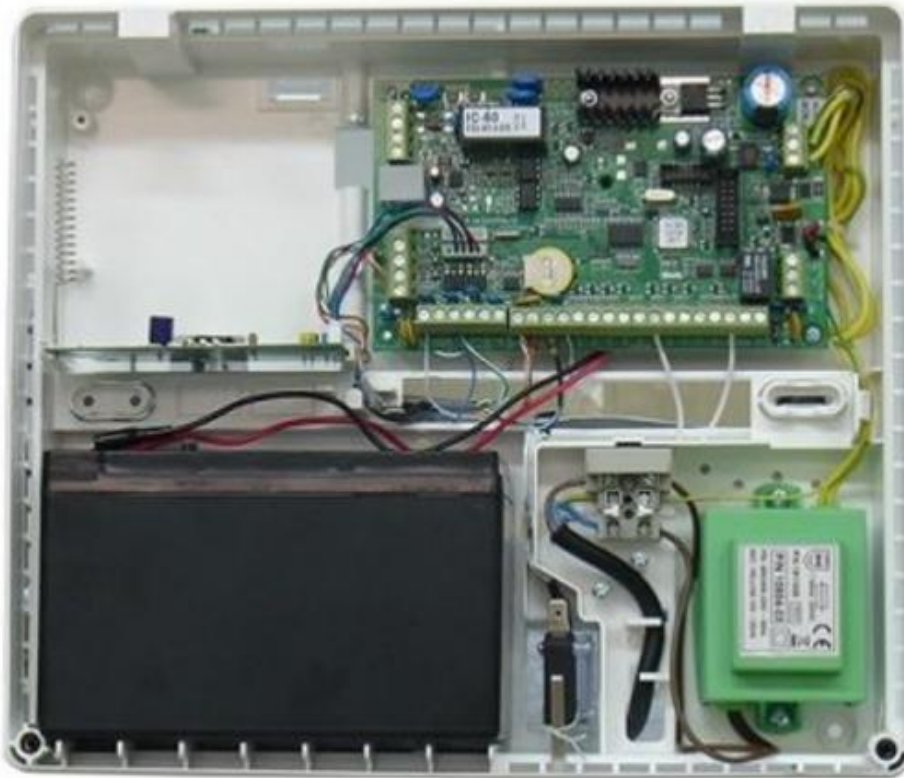
- ❖ Κεντρική μονάδα ελέγχου (ΚΜΕ)
- ❖ Πληκτρολόγιο
- ❖ RFID κάρτες
- ❖ Είσοδοι του συστήματος
  - Αισθητήρες
    - Ανιχνευτές καπνού
    - Ανιχνευτές φωτιάς
    - Ανιχνευτές κρούσης
    - Ανιχνευτές θραύσης κρυστάλλων
    - Ανιχνευτές κίνησης
    - Μαγνητικές επαφές
    - Ανιχνευτές πλημμύρας
    - Υπέρυθροι αισθητήρες δέσμης
    - Αισθητήρες θερμοκρασίας
    - Ανιχνευτές υπερήχων
- ❖ Έξοδοι του συστήματος
  - Σειρήνες
    - Εσωτερικές
    - Εξωτερικές
  - Οπτικές ενδείξεις
    - Φάροι
    - Φώτα
  - Ειδοποιήσεις μέσω δικτύου
    - SMS
    - Email
    - Push Notification
  - Ενεργοποίηση καμερών παρακολούθησης
  - Ειδικός αυτοματισμός
    - Κλείδωμα πορτών
    - Απενεργοποίηση συσκευών

### 2.4.1 Κεντρική μονάδα ελέγχου

Η κεντρική μονάδα ελέγχου είναι η "καρδιά" ενός συστήματος συναγερμού, καθώς συγκεντρώνει και διαχειρίζεται όλα τα δεδομένα από τους αισθητήρες, τους ανιχνευτές και τις εξόδους του συστήματος. Ο ρόλος της είναι να λαμβάνει τα σήματα από τους αισθητήρες, να τα επεξεργάζεται και να αποφασίζει τις κατάλληλες ενέργειες, όπως την ενεργοποίηση της σειρήνας ή την αποστολή ειδοποίησης. [11]

Αποτελείται από:

- ❖ Πλακέτα κεντρικής μονάδας ελέγχου
  - Μικροελεγκτής
  - Τροφοδοτικό
  - Υποδοχές συνδέσεων
  - Μετασχηματιστής 230V AC σε 12V DC
- ❖ Μπαταρίες 12V/7Ah
- ❖ Συσκευές επικοινωνίας



Σχήμα 2.2: Πίνακας συναγερμού Siemens Sintony ic60

<https://sat-alarm.gr/shop/%CF%80%CE%AF%CE%BD%CE%B1%CE%BA%CE%B1%CF%82-%CF%83%CF%85%CE%BD%CE%B1%CE%B3%CE%B5%CF%81%CE%BC%CE%BF%CF%8D-siemens-sintony-ic60/>

### 2.4.2 Πληκτρολόγιο

Το πληκτρολόγιο σε ένα σύστημα συναγερμού είναι το βασικό μέσο αλληλεπίδρασης του χρήστη με την κεντρική μονάδα ελέγχου. Πρόκειται για μια συσκευή μέσω της οποίας ο χρήστης μπορεί να

ενεργοποιεί, να απενεργοποιεί και να διαχειρίζεται το σύστημα, εισάγοντας εντολές ή κωδικούς. Ανάλογα με τον τύπο του συστήματος, το πληκτρολόγιο μπορεί να είναι ενσύρματο ή ασύρματο.



Σχήμα 2.3: Πληκτρολόγιο Siemens Sintony IKP6-03

<https://www.skrouz.gr/s/23292140/SIEMENS-IC60M-8-kit-04-Kalodiako-Set-synagermou-mepinaka-pliktrologio-2-esoterikous-anichneytes-kinisis.html>

### 2.4.3 Ανιχνευτές κίνησης

Οι ανιχνευτές κίνησης είναι βασικά στοιχεία ενός συστήματος συναγερμού, σχεδιασμένα να εντοπίζουν την παρουσία ή την κίνηση ανθρώπων (ή αντικειμένων) σε έναν προστατευόμενο χώρο. Ενεργοποιούνται όταν ανιχνεύσουν κίνηση και στέλνουν σήμα στην κεντρική μονάδα, η οποία αποφασίζει πώς θα αντιδράσει (π.χ., ενεργοποίηση σειρήνας ή ειδοποίηση στον χρήστη).



Σχήμα 2.4: Ανιχνευτής κίνησης υπερθύρων ADM-Q12 Siemens

<https://akatos.net/product/el-adm-i12w1-en-adm-i12w1/>

#### 2.4.4 Σειρήνες

Οι σειρήνες ειδοποίησης είναι βασικά εξαρτήματα ενός συστήματος συναγερμού και έχουν ως κύριο σκοπό να παρέχουν ακουστικές και, συχνά, οπτικές ενδείξεις όταν ενεργοποιείται ο συναγερμός. Αποτελούν το μέσο άμεσης ειδοποίησης για τον χρήστη, τους γείτονες ή τους περαστικούς, ενώ ταυτόχρονα λειτουργούν αποτρεπτικά για τους εισβολείς.



Σχήμα 2.5: Ασύρματη σειρήνα ISR6-12R Siemens

<https://www.sourcesecurity.com/vanderbilt-isrw6-12r-technical-details.html>

## Κεφάλαιο 3ο: Arduino

### 3.1 Εισαγωγή

Το Arduino αποτελεί μια δημοφιλή πλατφόρμα ανάπτυξης ανοιχτού κώδικα, που επιτρέπει την κατασκευή και τον προγραμματισμό ηλεκτρονικών εφαρμογών. Προσφέρει μια απλή και προσιτή προσέγγιση για την υλοποίηση διαφόρων έργων, από βασικά πειράματα μέχρι πολύπλοκες διατάξεις αυτοματισμού και ρομποτικής. Με ανοιχτό υλικό (hardware) και λογισμικό (software), επιτρέπει την εύκολη διασύνδεση αισθητήρων και άλλων ηλεκτρονικών εξαρτημάτων, καθιστώντας το ιδανικό εργαλείο για εκπαιδευτικούς, φοιτητές και χομπίστες, προσφέροντας δυνατότητες για δημιουργία πρωτοτύπων και μάθηση βασικών εννοιών της ηλεκτρονικής και του προγραμματισμού. [15]



Σχήμα 3.1: Arduino LOGO

[https://el.wikipedia.org/wiki/%CE%91%CF%81%CF%87%CE%B5%CE%AF%CE%BF:Arduino\\_Logo.svg](https://el.wikipedia.org/wiki/%CE%91%CF%81%CF%87%CE%B5%CE%AF%CE%BF:Arduino_Logo.svg)

### 3.2 Εφαρμογές Arduino

Με το Arduino υλοποιούνται πολλές εφαρμογές που βοηθούν στην αυτοματοποίηση της καθημερινότητας. Ένα παράδειγμα είναι ο έξυπνος φωτισμός. Όταν οι αισθητήρες κίνησης δεν ανιχνεύουν ανθρώπινη παρουσία, τα φώτα δεν ανάβουν. Επιπλέον, μπορεί να χρησιμοποιηθεί ένα Arduino για το αυτόματο πότισμα των φυτών με έναν αισθητήρα υγρασίας. Θα ενεργοποιούν το πότισμα μόνο όταν το χώμα είναι στεγνό. Για λόγους ασφαλείας, μπορεί να σχεδιαστεί ο συναγερμός που ανιχνεύει κίνηση ή καπνό και ειδοποιεί τον ιδιοκτήτη. Τέλος, ένα ακόμη παράδειγμα είναι ο μετεωρολογικός σταθμός Arduino, ο οποίος καταγράφει δεδομένα σε πραγματικό χρόνο, όπως η θερμοκρασία, η υγρασία και η πίεση, τα οποία είναι ιδανικά για την παρακολούθηση του καιρού.

### 3.3 Πλακέτες Arduino

#### 3.3.1 Arduino UNO

Η πιο γνωστή και ευρέως χρησιμοποιούμενη πλακέτα, ιδανική για αρχάριους αλλά και πιο προχωρημένα projects. Χρησιμοποιεί τον μικροελεγκτή ATmega328P και έχει αρκετές ψηφιακές και αναλογικές εισόδους/εξόδους.

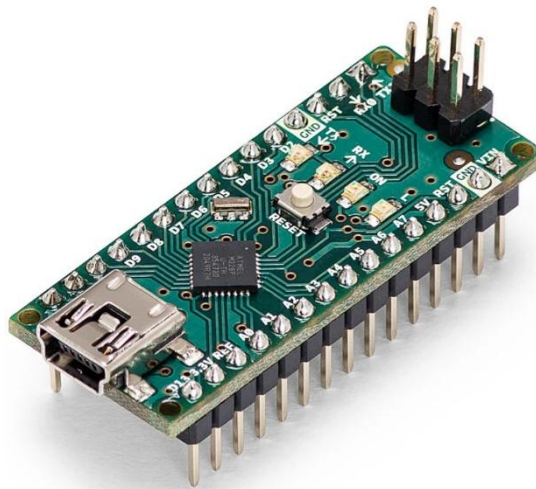


Σχήμα 3.2: Arduino UNO

<https://store.arduino.cc/en-gr/products/arduino-uno-rev3-smd?queryID=e91d5131f109188a797382025d33b354>

### 3.3.2 Arduino NANO

Είναι μια μικρού μεγέθους και ευέλικτη πλατφόρμα, δημοφιλής για έργα που απαιτούν μικρό χώρο και ελαφρύ βάρος. Βασίζεται στον μικροελεγκτή ATmega328, ενώ οι νεότερες εκδόσεις της έρχονται με το ATmega328P ή το ATmega168.

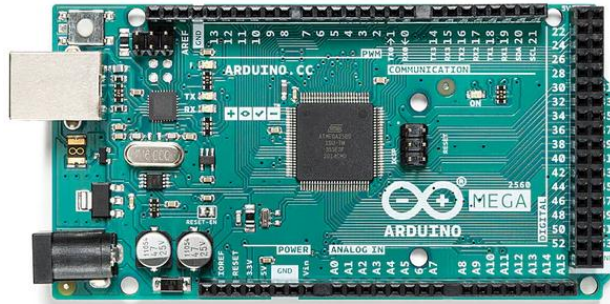


Σχήμα 3.3: Arduino NANO

<https://store.arduino.cc/products/arduino-nano?queryID=undefined>

### 3.3.3 Arduino MEGA 2560

Σχεδιασμένη για πιο απαιτητικές εφαρμογές, με 54 ψηφιακές εισόδους/εξόδους και 16 αναλογικές εισόδους. Χρησιμοποιείται σε projects που απαιτούν πολλαπλές συνδέσεις.

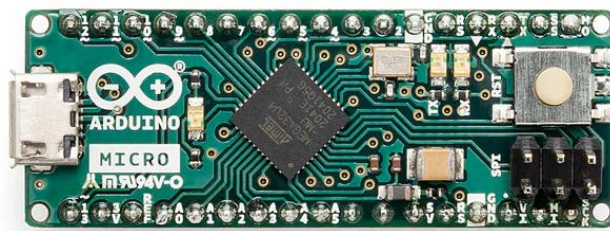


Σχήμα 3.4: Arduino MEGA 2560

<https://store.arduino.cc/products/arduino-mega-2560-rev3?queryID=f947280e252c02ed1b8042fff9966d9f>

### 3.3.4 Arduino MICRO

Είναι σχεδιασμένη για έργα που απαιτούν μικρό μέγεθος και υψηλή συνδεσιμότητα. Βασίζεται στον μικροελεγκτή ATmega32U4, ο οποίος επιτρέπει στο Micro να επικοινωνεί απευθείας με τον υπολογιστή μέσω USB και να λειτουργεί ως συσκευή HID.

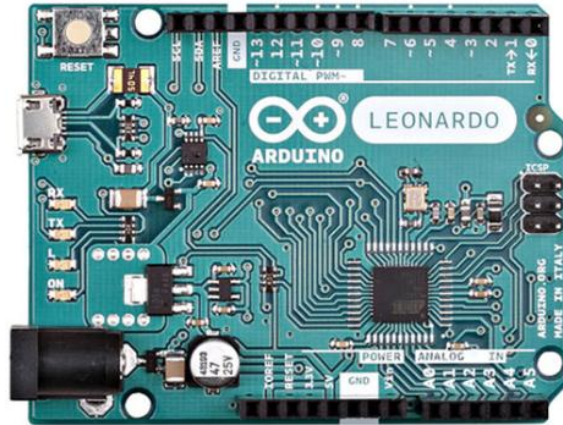


Σχήμα 3.5: Arduino MICRO

<https://store.arduino.cc/products/arduino-micro?queryID=86602614e1f4f128918b166b571e4bf1>

### 3.3.5 Arduino Leonardo

Χρησιμοποιεί τον μικροελεγκτή ATmega32U4, ο οποίος διαθέτει ενσωματωμένη δυνατότητα USB επικοινωνίας. Αυτό του επιτρέπει να λειτουργεί ως HID (Human Interface Device), δηλαδή να προσομοιώνει συσκευές όπως πληκτρολόγιο, ποντίκι ή joystick χωρίς την ανάγκη πρόσθετου hardware. Διαθέτει 20 ψηφιακές εισόδους/εξόδους (εκ των οποίων οι 7 υποστηρίζουν PWM), 12 αναλογικές εισόδους και μια micro USB θύρα για προγραμματισμό και τροφοδοσία. Είναι κατάλληλο για έργα που απαιτούν αλληλεπίδραση με τον υπολογιστή, όπως προσαρμοσμένα πληκτρολόγια, game controllers ή αυτοματισμοί που χρησιμοποιούν προσομοιώσεις πληκτρολογίου/ποντικιού.



Σχήμα 3.6: Arduino LEONARDO

<https://store.arduino.cc/products/arduino-leonardo-with-headers?queryID=062ee89672a8ebf19fcbaf1fd3b10786d>

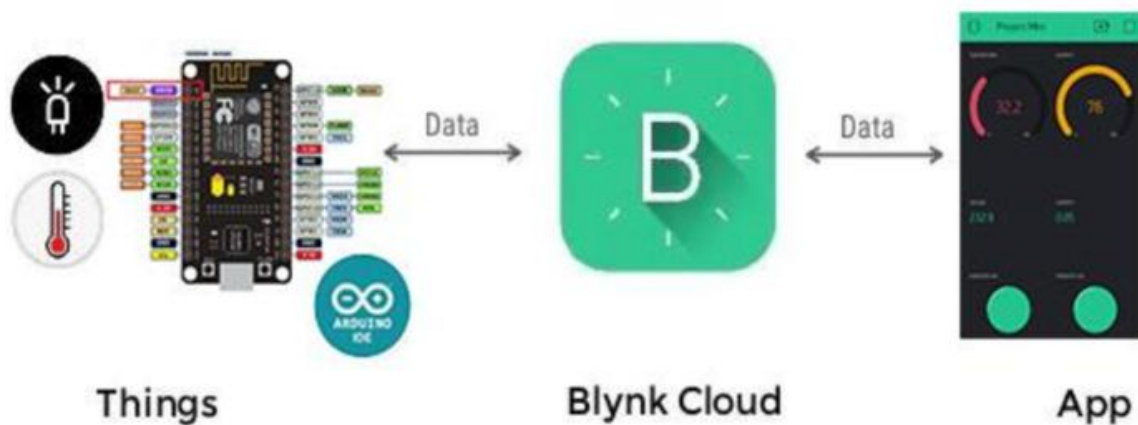
## Κεφάλαιο 4ο: Εφαρμογή Blynk

### 4.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα αναφερθεί η εφαρμογή Blynk που χρησιμοποιήθηκε για τον απομακρυσμένο έλεγχο του συστήματος.

### 4.2 Ορισμός

Η πλατφόρμα Blynk είναι ένα εργαλείο για την ανάπτυξη εφαρμογών Internet of Things (IoT) που επιτρέπει στους χρήστες να δημιουργούν και να ελέγχουν έξυπνες συσκευές μέσω κινητών συσκευών. Χρησιμοποιείται ευρέως για έργα που περιλαμβάνουν μικροελεγκτές, αισθητήρες και άλλες IoT συσκευές, παρέχοντας ευκολία στη σύνδεση και την απομακρυσμένη παρακολούθηση. [14]



Σχήμα 4.1: Αρχιτεκτονική Blynk

[https://www.researchgate.net/figure/Blynk-architecture-Things-Cloud-Apps\\_fig4\\_364534592](https://www.researchgate.net/figure/Blynk-architecture-Things-Cloud-Apps_fig4_364534592)

### 4.3 Χαρακτηριστικά

- ❖ Απλή διασύνδεση χρήστη
- ❖ Ποικιλία εργαλείων για παρουσίαση δεδομένων
- ❖ Απομακρυσμένη πρόσβαση και έλεγχος
- ❖ Υποστήριξη για ποικιλία συσκευών και πρωτοκόλλων
- ❖ Ασφαλής επικοινωνία
- ❖ Αποθήκευση δεδομένων με ιστορικό
- ❖ Υποστήριξη για υπηρεσίες cloud
- ❖ Ενσωμάτωση με άλλες πλατφόρμες

## Κεφάλαιο 5ο: Σχεδίαση και υλοποίηση εργασίας

### 5.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα αναλυθεί ο σχεδιασμός και η υλοποίηση του συστήματος ασφαλείας για οικιακή χρήση καθώς και η κατασκευή της μακέτας που προσομοιώνει ένα σπίτι.

### 5.2 Σχεδίαση κατασκευής

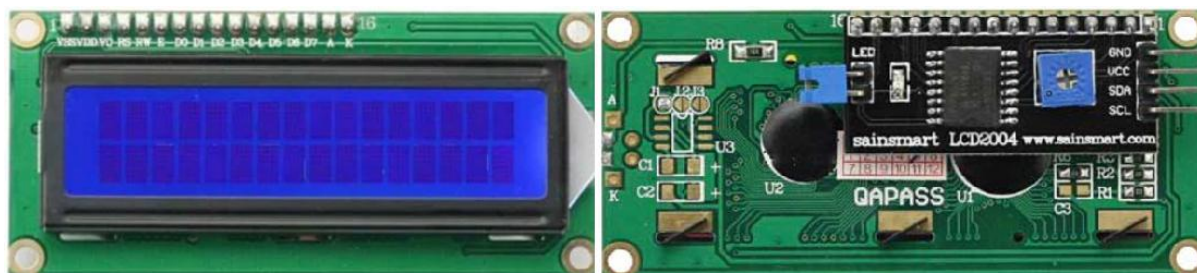
Για τον σχεδιασμό του συστήματος ασφαλείας χρησιμοποιήθηκε για κεντρική μονάδα ελέγχου το Arduino MEGA 2560 και το ESP8266 (ESP-01) για την σύνδεσή του στο διαδίκτυο μέσω Wi-Fi.

Η σκέψη για τον σχεδιασμό του συστήματος είναι να μπορεί ο χρήστης με την χρήση του πληκτρολογίου να εισάγει τον προσωπικό του κωδικό και να οπλίζει ή να αφοπλίζει τον συναγερμό, να έχει την δυνατότητα του πλήρη ελέγχου απομακρυσμένα από οποιαδήποτε συσκευή και να ενημερώνεται σε πραγματικό χρόνο αν υπάρχει κάποια παραβίαση ή αλλαγή στον χώρο του. Οι ανιχνευτές του συστήματος για την επιτήρηση του χώρου είναι: Μία μαγνητική επαφή που ανιχνεύει το άνοιγμα της κεντρικής πόρτας, ένας αισθητήρας υπερήχων όπου θα ανάβει αυτόματα τον περιμετρικό φωτισμό του σπιτιού όταν κάποιος πλησιάσει, ένας αισθητήρας φωτιάς, ένας αισθητήρας θερμοκρασίας και υγρασίας, ένας αισθητήρας κίνησης, μία δέσμη λέιζερ και ένας αισθητήρας κρούσης όπου θα ειδοποιούν τον χρήστη για τυχόν πυρκαγιά ή παραβίαση του χώρου του. Επιπλέον, θα υπάρχει μία οθόνη ενδείξεων για τοπικό χειρισμό του συστήματος.

### 5.3 Υλοποίηση κατασκευής

#### 5.3.1 Οθόνη

Για την εμφάνιση δεδομένων και τοπικό χειρισμό του συστήματος χρησιμοποιήθηκε μία οθόνη υγρών κρυστάλλων LCD με ενσωματωμένη λειτουργία I2C. Έχει μέγεθος 16 χαρακτήρων ανά 2 γραμμές (16x2). Η τάση λειτουργίας της είναι 2,5-6V DC. [2][3]

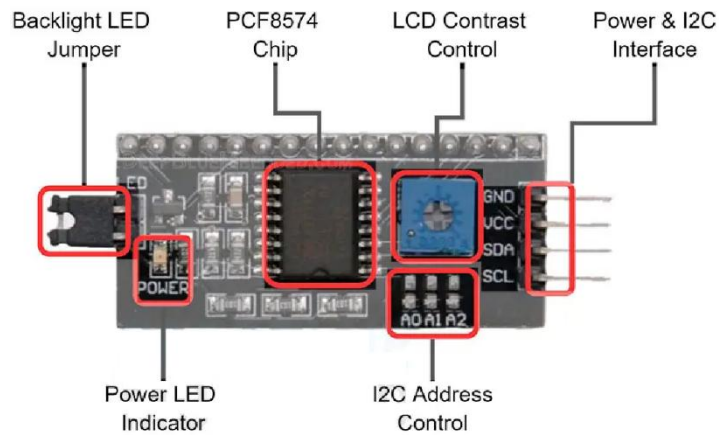


Σχήμα 5.1: 16x2 LCD display with I2C interface

<https://store-usa.arduino.cc/products/16x2-lcd-display-with-i-c-interface>

##### 5.3.1.1 Συσκευή διεπαφής I2C

Ο προσαρμογέας LCD I2C έχει ως επεξεργαστή τον PCF8574 8-bit που μετατρέπει τα δεδομένα από το arduino σε παράλληλα δεδομένα για να τα απεικονίσει η οθόνη. Με αυτό τον τρόπο μειώνει τις απαιτούμενες εισόδους που χρειάζεται για να επικοινωνήσουν η οθόνη με το arduino από 8 σε 4. [2][3]



Σχήμα 5.2: I2C Module

<https://deepbluembedded.com/arduino-i2c-lcd/>

Πίνακας 5.1: Σύνδεση LCD I2C οθόνης με arduino

I2C LCD	Arduino
VCC	5V
GND	GND
SCL	SCL
SDA	SDA

### 5.3.2 Πληκτρολόγιο

Για να μπορεί να ελέγχει ο χρήστης το σύστημα συναγερμού, επιλέχθηκε ένα πληκτρολόγιο 16 πλήκτρων τύπου matrix. Αποτελείται από 4 γραμμές και 4 στήλες όπου κάθε στοιχείο του πίνακα είναι ένα πλήκτρο που αντιστοιχεί σε μια μοναδική στήλη και μια μοναδική γραμμή.



Σχήμα 5.3: Πληκτρολόγιο 4x4 matrix

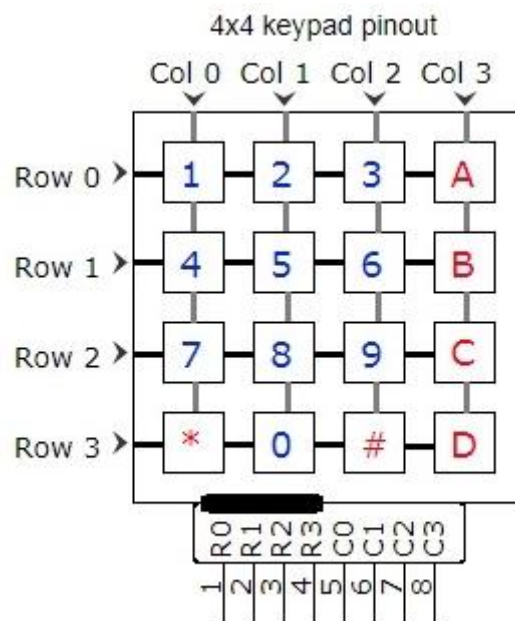
<https://grobotronics.com/keypad-matrix-4x4.html?sl=en>

### 5.3.2.1 Αρχή λειτουργίας πληκτρολογίου

Το πληκτρολόγιο 4x4 matrix αποτελείται από 8 καλώδια, 4 σειρές και 4 στήλες. Ο κάθε ακροδέκτης σειράς και στήλης συνδέεται με ένα συγκεκριμένο πλήκτρο. Όταν ένα πλήκτρο πατιέται, ο ακροδέκτης της σειράς και ο αντίστοιχος ακροδέκτης της στήλης συνδέονται μεταξύ τους. Ο μικροελεγκτής σαρώνει τη συστοιχία συνεχώς, ενεργοποιώντας διαδοχικά τις γραμμές ή τις στήλες και ανιχνεύοντας ποιο πλήκτρο συνδέεται στην αντίστοιχη στήλη και σειρά. Όταν το πλήκτρο στην αντίστοιχη σειρά και στήλη είναι πατημένο, το κύκλωμα κλείνει, και ο μικροελεγκτής ανιχνεύει τη σύνδεση. Ανάλογα με τη σειρά και τη στήλη, μπορεί να προσδιοριστεί το ακριβές πλήκτρο που πατήθηκε. Κάθε πλήκτρο αντιστοιχεί σε μια μοναδική συνδυασμένη θέση (σειρά-στήλη) και έτσι, μέσω των εντολών του προγράμματος, ο μικροελεγκτής μπορεί να καταλάβει ποιο πλήκτρο πατήθηκε και να εκτελέσει την αντίστοιχη ενέργεια, όπως εμφάνιση χαρακτήρων ή αριθμών στην οθόνη. [16]

### 5.3.2.2 Συνδεσμολογία με το Arduino

Η συνδεσμολογία ενός keypad matrix 4x4 είναι μια κοινή μέθοδος για την ανάγνωση πλήκτρων σε ένα πληκτρολόγιο με 16 πλήκτρα (4 σειρές x 4 στήλες). Η βασική ιδέα πίσω από αυτή τη διάταξη είναι ότι μειώνεται ο αριθμός των απαιτούμενων ακροδεκτών σε σύγκριση με το να έχει κάθε πλήκτρο τη δική του γραμμή και στήλη. [16]

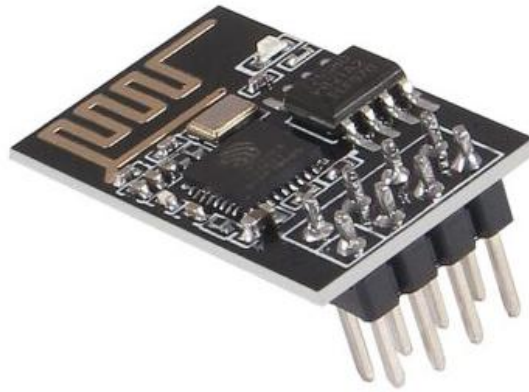


Σχήμα 5.4: Εσωτερική διάταξη πληκτρολογίου

<https://keszoox.com/products/membrane-keypad-4x4-matrix>

### 5.3.3 ESP8266-01

Ο ESP8266 (ESP-01) είναι μια δημοφιλής και οικονομική πλακέτα Wi-Fi, η οποία επιτρέπει την ενσωμάτωση ασύρματης συνδεσιμότητας σε μικροελεγκτές και IoT εφαρμογές. Παρά την απλότητά του, προσφέρει ισχυρές δυνατότητες και έχει χρησιμοποιηθεί ευρέως σε έργα που απαιτούν ασύρματη συνδεσιμότητα. [4]



Σχήμα 5.5: ESP8266 Wi-Fi Module ESP-01

<https://www.ebay.com/itm/186338482031>

### 5.3.3.1 Χαρακτηριστικά ESP-01

Βασίζεται στο τσιπ ESP8266EX της Espressif, το οποίο διαθέτει έναν πυρήνα Tensilica L106 (32-bit RISC) με συχνότητα έως 80 MHz (ή 160 MHz σε υπερχρονισμό). Υποστηρίζει 802.11 b/g/n Wi-Fi και Μπορεί να λειτουργήσει ως σταθμός (station), access point (AP) ή και τα δύο ταυτόχρονα. Η τάση λειτουργίας του είναι στα 3V. [4]

Categories	Items	Values
WiFi Paramters	WiFi Protocoles	802.11 b/g/n
	Frequency Range	2.4GHz-2.5GHz (2400M-2483.5M)
Hardware Paramaters	Peripheral Bus	UART/HSPI/I2C/I2S/Ir Remote Contorl
		GPIO/PWM
	Operating Voltage	3.0~3.6V
	Operating Current	Average value: 80mA
	Operating Temperature Range	-40~125°
	Ambient Temperature Range	Normal temperature
	Package Size	14.3mm*24.8mm*3mm
	External Interface	N/A
Software Paramaters	Wi-Fi mode	station/softAP/SoftAP+station
	Security	WPA/WPA2
	Encryption	WEP/TKIP/AES
	Firmware Upgrade	UART Download / OTA (via network) / download and write firmware via host
	Ssoftware Development	Supports Cloud Server Development / SDK for custom firmware development
	Network Protocols	IPv4, TCP/UDP/HTTP/FTP
	User Configuration	AT Instruction Set, Cloud Server, Android/IOS App

## Σχήμα 5.6: Τεχνικά χαρακτηριστικά ESP-01

<https://teplo-energetika.ru/instruktion/arduino-chip/esp-12e-datasheet-eng.pdf>

### 5.3.3.2 Συνδεσμολογία με το Arduino

Πίνακας 5.2: Συνδεσμολογία ακίδων

Arduino MEGA	ESP-01
TX1	RX0
RX1	TX0

### 5.3.4 Απομακρυσμένος έλεγχος Arduino

Για τον απομακρυσμένο έλεγχο του arduino και την παρακολούθηση των αισθητήρων χρησιμοποιούμε την βιβλιοθήκη του Blynk <BlynkSimpleShieldEsp8266.h> και μέσω της βιβλιοθήκης <ESP8266\_Lib.h> παρέχουμε σύνδεση του arduino με το Blynk. Για την αποστολή και λήψη δεδομένων στο περιβάλλον του Blynk προστέθηκαν κάποιες γραμμές κώδικα μέσω του Arduino IDE για να αλληλοεπιδρούν οι δύο εφαρμογές μεταξύ τους. Στη συνέχεια στην εφαρμογή του Blynk προστέθηκαν γραφικά στοιχεία για την απεικόνιση των δεδομένων και την λήψη ειδοποιήσεων.

#### 5.3.4.1 Επικοινωνία Blynk και Arduino

Για την ανταλλαγή δεδομένων μεταξύ των δύο εφαρμογών δημιουργήθηκαν ψηφιακοί ακροδέκτες (Virtual Pins). Κάθε ψηφιακός ακροδέκτης είναι ένα κανάλι επικοινωνίας μεταξύ του Arduino και του Blynk. Παράδειγμα, για την αποστολή δεδομένων του ανιχνευτή κίνησης (PIR Sensor) δημιουργήθηκε στο Blynk μία ροή δεδομένων (Datastream) στον ψηφιακό ακροδέκτη V31 και έπειτα έγιναν οι απαραίτητες ρυθμίσεις για την ανταλλαγή δεδομένων και την λήψη ειδοποιήσεων.

**Virtual Pin Datastream**

General Expose to Automations

NAME: motion ALIAS: motion

PIN: V31 DATA TYPE: Integer

UNITS: None

MIN: 0 MAX: 1 DEFAULT VALUE: 0

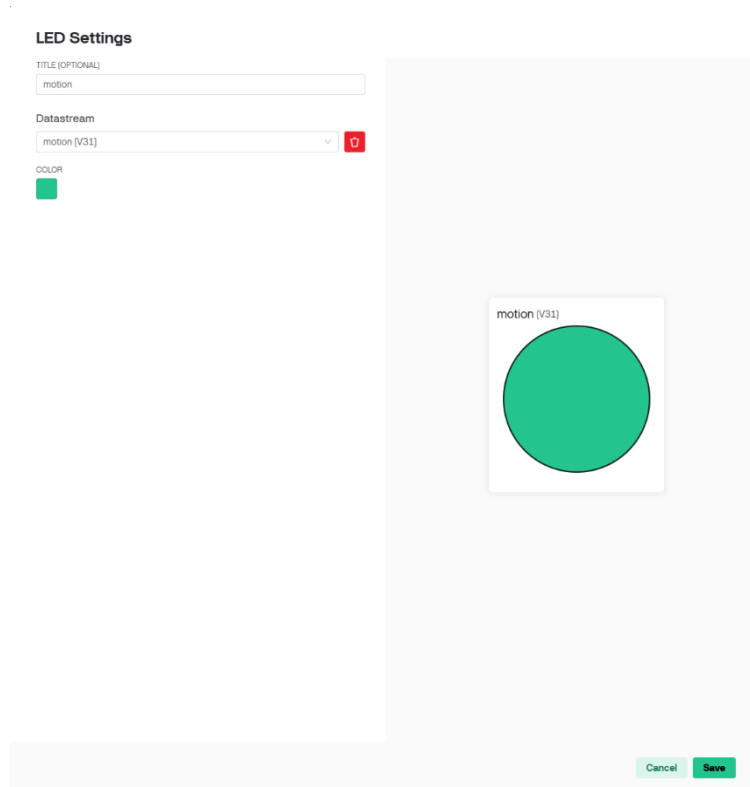
Enable history data

Cancel Save

Σχήμα 5.7: Δημιουργία ψηφιακού ακροδέκτη

<https://blynk.io/>

Με αυτό τον τρόπο δημιουργήσαμε την ροή δεδομένων που χρειαζόμαστε και ως επόμενο βήμα είναι η απεικόνιση των δεδομένων στην εφαρμογή με γραφικό στοιχείο (widget) και το ρυθμίζουμε να δέχεται δεδομένα από το συγκεκριμένο datastream.



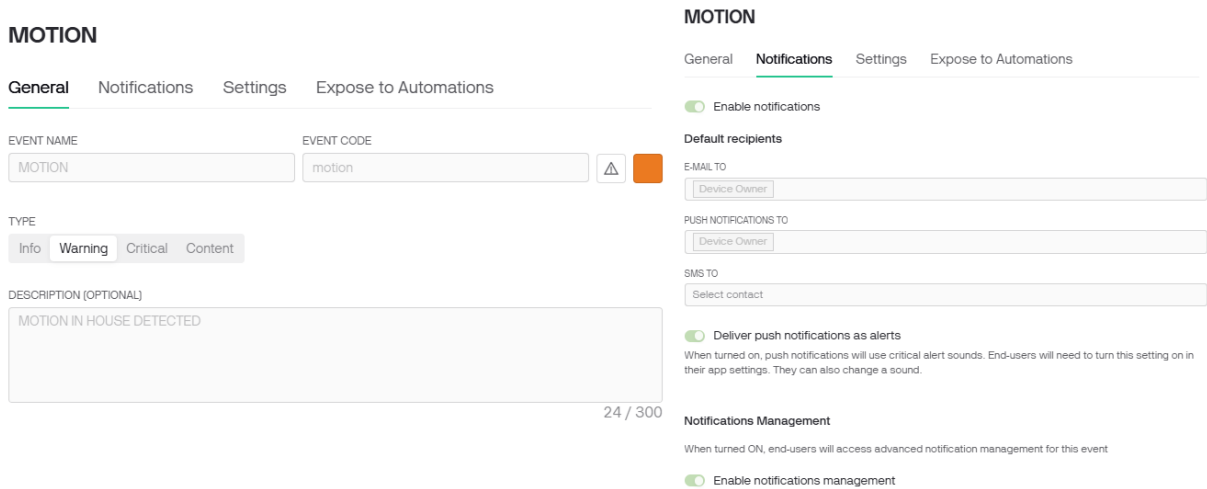
Σχήμα 5.8: Δημιουργία γραφικού στοιχείου (widget)

<https://blynk.io/>

Για να μπορέσουμε να στείλουμε δεδομένα από το Arduino, πρέπει να χρησιμοποιήσουμε κάποιες εντολές από την βιβλιοθήκη Blynk. Η εντολή για να στέλνουμε δεδομένα προς την εφαρμογή είναι η «`Blynk.virtualWrite(Vpin, sensorData);`», όπου `Vpin` είναι το νούμερο του ψηφιακού ακροδέκτη που δημιουργήσαμε και όπου `sensorData` είναι η μεταβλητή που έχουμε δημιουργήσει στον κώδικα για να αποθηκεύουμε τιμές.

#### 5.3.4.2 Αποστολή ειδοποιήσεων

Η εφαρμογή Blynk μας δίνει την δυνατότητα αποστολής ειδοποιήσεων σε πραγματικό χρόνο δημιουργώντας συμβάντα (events) και να τα αντιστοιχίσουμε σε συγκεκριμένα datastream. Οι ειδοποιήσεις που μας προσφέρει είναι μέσω email, push notification στο κινητό τηλέφωνο και SMS. Για να χρησιμοποιήσουμε την λειτουργία των ειδοποιήσεων θα πρέπει πρώτα να προσθέσουμε στον κώδικα του Arduino την εντολή «`Blynk.logEvent(event_code);`», όπου «`event_code`» είναι ο κωδικός του συμβάν που έχουμε δώσει στην εφαρμογή.



Σχήμα 5.9: Λήψη ειδοποιήσεων από το Blynk

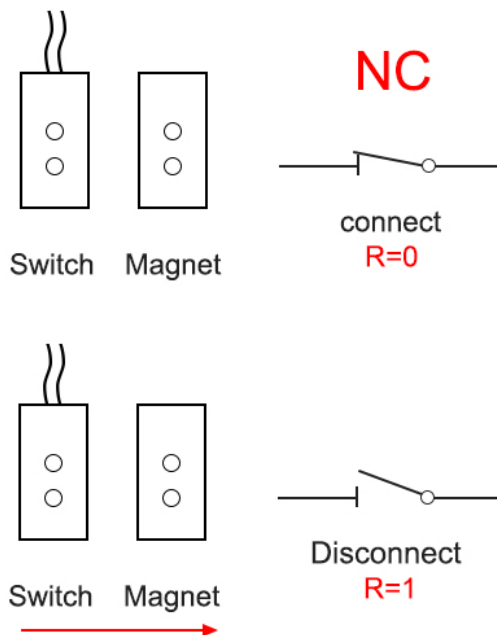
<https://blynk.io/>

### 5.3.5 Μαγνητική επαφή

Για τον έλεγχο της κεντρικής πόρτας εισόδου στο σύστημα χρησιμοποιήθηκε μία μαγνητική επαφή normally close.

#### 5.3.5.1 Συνδεσμολογία μαγνητικής επαφής

Η μαγνητική επαφή έχει δύο καλώδια. Το ένα καλώδιο συνδέεται στην γείωση (GND) του Arduino και το δεύτερο καλώδιο συνδέεται στα 5V μέσω μίας αντίστασης 10KΩ και στην ένωσή τους συνδέεται ένα καλώδιο που πάει σε μία ψηφιακή είσοδο του Arduino που έχουμε ρυθμίσει. Το Arduino διαβάζει λογικό “1” όσο η πόρτα είναι κλειστή και μόλις ανοίξει διαβάζει λογικό “0” και καταλαβαίνει ότι υπάρχει παραβίαση.



Σχήμα 5.10: Κανονικά κλειστή μαγνητική επαφή

<https://www.amazon.in/Gebildet-Normally-Induction-2-5mm%C3%9714mm-Multi-Use/dp/B0BPYGGTBS>

### 5.3.5.2 Συνδεσμολογία με το Arduino

Πίνακας 5.3: Συνδεσμολογία με το Arduino

Μαγνητική επαφή	Arduino
Καλώδιο 1	Γείωση (GND)
Καλώδιο 2	10KΩ + 5V (ακροδέκτης 32)

### 5.3.6 Ανιχνευτής φωτιάς

Για την ανίχνευση φωτιάς στο σύστημα συναγερμού χρησιμοποιήθηκε ένας KY-026. Ο αισθητήρας διαθέτει έναν υπέρυθρο φωτοανιχνευτή που ανιχνεύει το υπέρυθρο φως που εκπέμπεται από τις φλόγες. Περιλαμβάνει έναν αναλογικό και έναν ψηφιακό ακροδέκτη, όπου ο ψηφιακός ακροδέκτης (D0) ενεργοποιείται όταν ανιχνευθεί φλόγα πάνω από ένα προκαθορισμένο όριο και ο αναλογικός ακροδέκτης (A0) παρέχει μια συνεχόμενη τιμή που αντιστοιχεί στην ένταση της φλόγας. Περιέχει επίσης μια μεταβλητή αντίσταση που επιτρέπει τη ρύθμιση της ευαισθησίας του αισθητήρα. Όταν ανιχνεύσει φλόγα ανάβει ένα LED στο module, η έξοδος D0 γίνεται HIGH και δίνει την εντολή σε μια ψηφιακή είσοδο στο Arduino. [13]



Σχήμα 5.11: Ανιχνευτής φωτιάς KY-026

<https://www.electronicclinic.com/arduino-fire-alarm-system-with-gsm-alert-text-message-buzzer/>

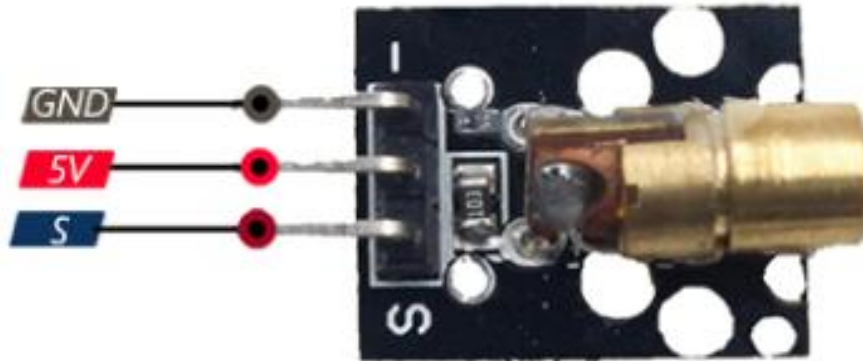
### 5.3.6.1 Συνδεσμολογία με το Arduino

Πίνακας 5.4: Συνδεσμολογία με το Arduino

KY-026	Arduino
A0	Ακροδέκτης A0
GND	Γείωση (GND)
VCC	5V
D0	Ακροδέκτης 33

### 5.3.7 Αισθητήρας λέιζερ

Χρησιμοποιήθηκε ένας πομπός λέιζερ σε συνδυασμό με έναν δέκτη λέιζερ για την ανίχνευση εμποδίων. Ο πομπός λέιζερ εκπέμπει μια εστιασμένη ακτίνα φωτός προς τον δέκτη, σχηματίζοντας μια "γραμμή ασφαλείας". Όταν ένας άνθρωπος ή αντικείμενο διακόψει την ακτίνα, ο δέκτης εντοπίζει την απουσία του σήματος και ενημερώνει το Arduino. Το Arduino, με τη σειρά του, ενεργοποιεί τον συναγερμό και στέλνει ειδοποίηση. Ο δέκτης συνδέεται σε μια ψηφιακή είσοδο για την ανίχνευση της διακοπής της ακτίνας. [5]



Σχήμα 5.12: Πομπός λέιζερ

<https://www.instructables.com/Laser-Diode-Module-Tutorial/>

Πίνακας 5.5: Τεχνικά χαρακτηριστικά πομπού

Τάση λειτουργίας	5V
Ισχύς εξόδου	5mW
Μήκος δέσμης	650nm
Ρεύμα λειτουργίας	<40mA
Θερμοκρασία λειτουργίας	-10°C ~ 40°C
Διαστάσεις πλακέτας	18.5mm x 15mm

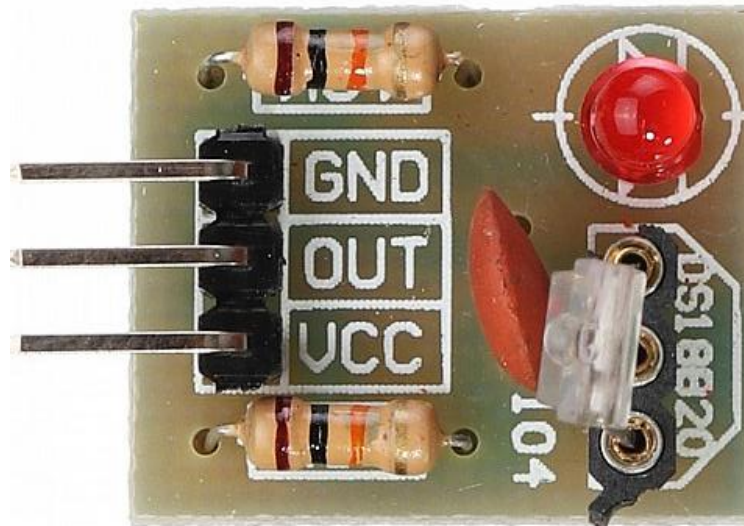
#### 5.3.7.1 Συνδεσμολογία με το Arduino

Πίνακας 5.6: Συνδεσμολογία πομπού λέιζερ

Laser sensor	Arduino
VCC	5V
GND	Γείωση (GND)
S	ακροδέκτης 28

### 5.3.8 Δέκτης λέιζερ

Ο δέκτης λέιζερ διαθέτει τρεις ακροδέκτες σύνδεσης (VCC, OUT, και GND) για την τροφοδοσία και την έξοδο σήματος, ενώ είναι σχεδιασμένο να ανιχνεύει την παρουσία ή την απουσία μιας ακτίνας φωτός. Περιλαμβάνει μία φωτοδίοδο που ανιχνεύει το φως και το μετατρέπει σε ηλεκτρικό σήμα. Διαθέτει επίσης αντιστάσεις για τη σταθεροποίηση του κυκλώματος, έναν κεραμικό πυκνωτή για την αποφυγή ηλεκτρικού θορύβου και τη σταθερότητα της τροφοδοσίας, καθώς και ένα LED ένδειξης που ανάβει όταν ανιχνεύεται ακτίνα φωτός ή αλλάζει η κατάσταση εξόδου.



Σχήμα 5.13: Δέκτης λέιζερ

<https://arduino.me/a/2297>

#### 5.3.8.1 Συνδεσμολογία με το Arduino

Πίνακας 5.7: Συνδεσμολογία δέκτη λέιζερ

Laser sensor	Arduino
VCC	5V
GND	Γείωση (GND)
S	ακροδέκτης 28

### 5.3.9 Ανιχνευτής κίνησης

Για την ανίχνευση κίνησης στο σύστημα συναγερμού θα χρησιμοποιήθηκε ένας παθητικός ανιχνευτής υπέρυθρης ακτινοβολίας (PIR) HC-SR501.



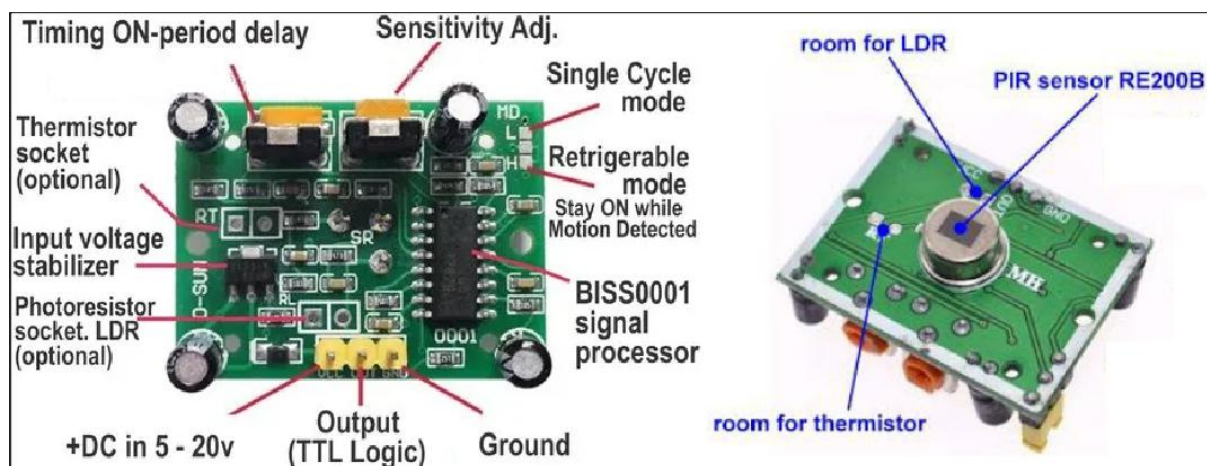
Σχήμα 5.14: Παθητικός ανιχνευτής υπέρυθρης ακτινοβολίας (PIR) HC-SR501

<https://smartxprokits.in/pir-sensor/>

### 5.3.9.1 Πλακέτα ανιχνευτή κίνησης (PIR) HC-SR501

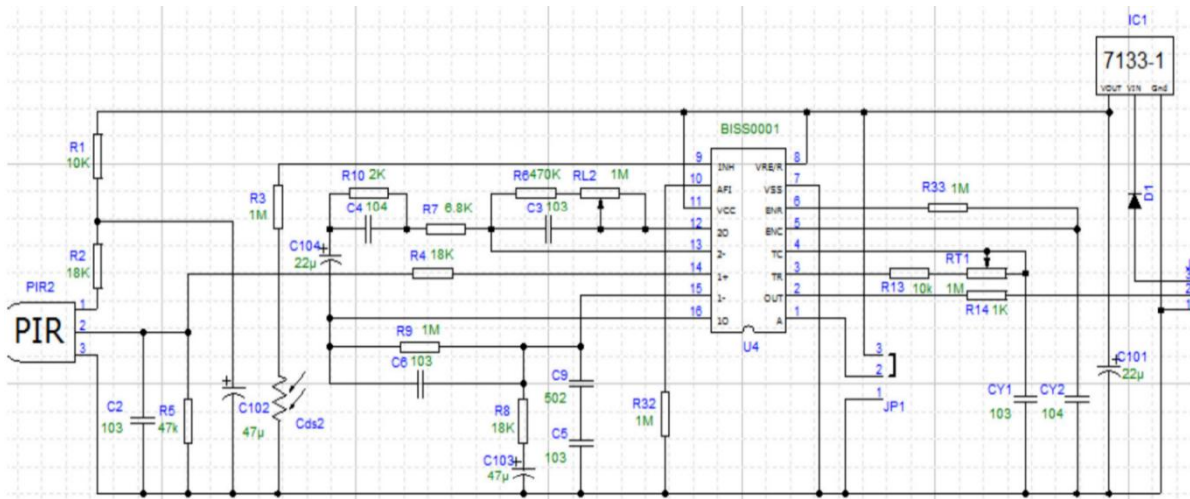
Στο κάτω μέρος της πλακέτας του HC-SR501 σύμφωνα με την Εικόνα 5.15, υπάρχει το ολοκληρωμένο BISS0001 που είναι ένας ελεγκτής σχεδιασμένος ειδικά για χρήση σε παθητικούς ανιχνευτές υπέρυθρης ακτινοβολίας, δύο ποτενσιόμετρα για ρύθμιση της απόστασης κάλυψης του αισθητήρα και την ρύθμιση του χρόνου που θα παραμένει σε κατάσταση HIGH η έξοδος, ενδεικτικά LED και ακροδέκτες διασύνδεσης.

Στην επάνω πλευρά σύμφωνα με την Εικόνα 5.15, βρίσκεται ο φακός Fresnel που βοηθάει στην αύξηση της εμβέλειας και το πεδίο δράσης του αισθητήρα. Στο εσωτερικό του υπάρχει ένας παθητικός ανιχνευτής υπέρυθρης ακτινοβολίας τύπου RE200B όπου τροφοδοτείται από έναν σταθεροποιητή τάσης στα 3.3V και αποτελείται από 2 στοιχεία ανίχνευσης συνδεδεμένα με μια διάταξη αντιστοίχισης τάσης (Voltage bucking configuration). Αυτή η διάταξη είναι σχεδιασμένη να αναιρεί τα σήματα που προκαλούνται από αλλαγές θερμοκρασίας, αλλά όταν ανιχνεύσει ένα σώμα ή αντικείμενο, τότε ενεργοποιούνται διαδοχικά τα στοιχεία και στέλνει το σήμα στο Arduino. [6] [7]



Σχήμα 5.15: Κάτω μέρος πλακέτας, επάνω μέρος πλακέτας

<https://chinese-electronics-products-tested.blogspot.com/p/hc-sr501-motion-sensor-tested.html>



Σχήμα 5.16: Σχηματικό (PIR) HC-SR501

<https://www.alldatasheet.com/html-pdf/1131987/ETC2/HC-SR501/231/2/HC-SR501.html>

### 5.3.9.2 Τεχνικά χαρακτηριστικά (PIR) HC-SR501

Πίνακας 5.8: Τεχνικά χαρακτηριστικά (PIR) HC-SR501

Τάση λειτουργίας	5-20VDC
Ρεύμα ηρεμίας	<50μΑ
Επίπεδα εξόδου	HIGH 3.3V LOW 0V
Χρόνος καθυστέρησης	5-300s
Γωνία αντίχενσης	<100°
Θερμοκρασία λειτουργίας	-15°C ~ 70°C
Μέγεθος φακού αισθητήρα	23mm

### 5.3.9.3 Συνδεσμολογία με το Arduino

Πίνακας 5.9: Συνδεσμολογία με το Arduino

HC-SR501	Arduino
VCC	5V
GND	Γείωση (GND)
OUTPUT	Ακροδέκτης 31

### 5.3.10 Αισθητήρας υπερήχων

Για την αυτόματη ενεργοποίηση του εξωτερικού φωτισμού χρησιμοποιήθηκε ο HC-SR04 αισθητήρας υπερήχων. Είναι ένας αισθητήρας απόστασης που χρησιμοποιεί υπερηχητικά κύματα για την ανίχνευση αντικειμένων. Προσφέρει εύρος μέτρησης από 2 εκατοστά έως 4 μέτρα με ακρίβεια περίπου  $\pm 3\text{mm}$ . Η λειτουργία του βασίζεται στη μέτρηση του χρόνου που απαιτείται για την ανακλαστική επιστροφή ενός υπερηχητικού σήματος, το οποίο παράγεται και λαμβάνεται από τον ίδιο τον αισθητήρα. [8]



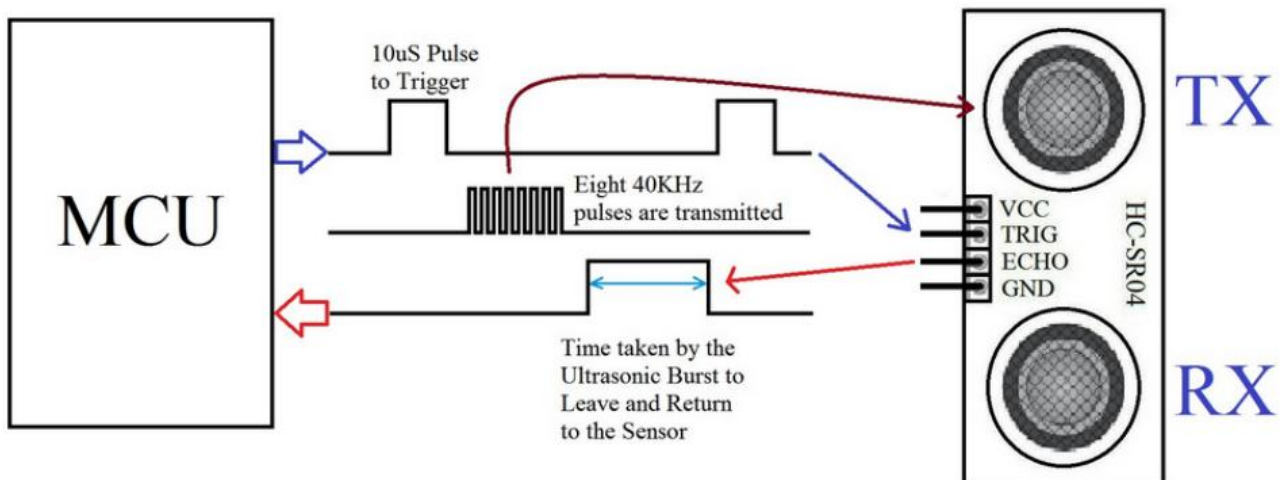
Σχήμα 5.17: Αισθητήρας υπερήχων HC-SR04

<https://netsonic.fi/en/hc-sr04-datasheet-and-pinout-ultrasonic-sensor-noncontact-range-detection/>

#### 5.3.10.1 Λειτουργία αισθητήρα HC-SR04

Ο αισθητήρας υπερήχων HC-SR04 λειτουργεί στέλνοντας έναν παλμό ενεργοποίησης (trigger) στον ακροδέκτη Trig με ελάχιστη διάρκεια 10 $\mu\text{s}$ . Ο πομπός εκπέμπει υπερηχητικά κύματα συχνότητας 40kHz τα οποία ταξιδεύουν στον αέρα και όταν συναντήσουν ένα αντικείμενο, ανακλώνται και επιστρέφουν στον δέκτη του αισθητήρα, ο οποίος ενεργοποιεί τον ακροδέκτη Echo για χρονικό διάστημα τόσο, όσο διαρκεί η επιστροφή του ανακλώμενου κύματος. Το Arduino μετρά τον χρόνο αυτόν και υπολογίζει την απόσταση με τον τύπο [8]

$$\text{Απόσταση} = \frac{\text{Χρόνος} \cdot 0.034}{2}$$



Σχήμα 5.18: Λειτουργία HC-SR04

<https://www.handsontec.com/dataspecs/HC-SR04-Ultrasonic.pdf>

### 5.3.10.2 Τεχνικά χαρακτηριστικά HC-SR04

Πίνακας 5.10: Τεχνικά χαρακτηριστικά αισθητήρα HC-SR04

Τάση λειτουργίας	3.3VDC ~ 5VDC
Ρεύμα ηρεμίας	<2mA
Ρεύμα λειτουργίας	15mA
Συχνότητα λειτουργίας	40KHz
Απόσταση λειτουργίας και ακρίβεια	2cm ~ 300cm ±3mm
Ευαισθησία	-65dB
Ηχητική πίεση	112dB
Γωνία	15°

### 5.3.11 Αισθητήρας κραδασμών

Για τον εντοπισμό θραύσης κρυστάλλων χρησιμοποιήθηκε ένας αισθητήρας κραδασμών VS-01. Στο εσωτερικό του περιέχει δύο εύκαμπτες μεταλλικές λάμες οι οποίες όταν χτυπηθεί ο αισθητήρας κάνουν επαφή μεταξύ τους και στέλνει κατάσταση HIGH στο Arduino.



Σχήμα 5.19: Αισθητήρας κραδασμών

<https://www.skroutz.gr/s/25721137/Aisthitiras-THraysia-Krystallon-se-Leyko-CHroma-VS-01.html>

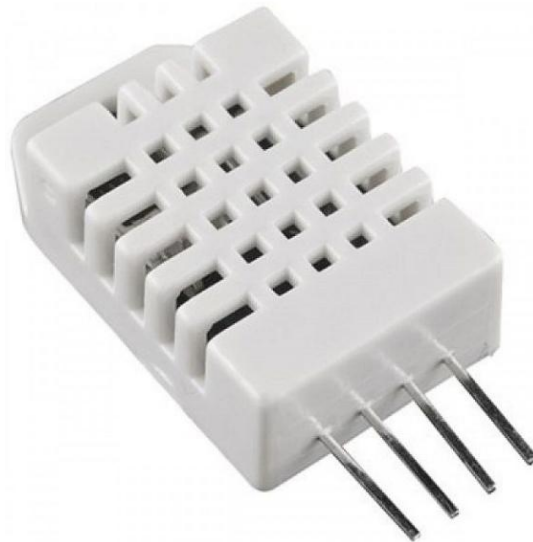
#### 5.3.11.1 Συνδεσμολογία αισθητήρα κραδασμών με το Arduino

Πίνακας 5.11: Συνδεσμολογία με το Arduino

Αισθητήρας κραδασμών	Arduino
Καλώδιο 1	Γείωση (GND)
Καλώδιο 2	ακροδέκτης 32

### 5.3.12 Αισθητήρας θερμοκρασίας και υγρασίας

Για την μέτρηση της θερμοκρασίας και της στον χώρο χρησιμοποιήθηκε ο αισθητήρας DHT22 που έχει δύο αισθητήρες εσωτερικά μαζί με έναν 8-bit μικροελεγκτή για την αποστολή των δεδομένων.



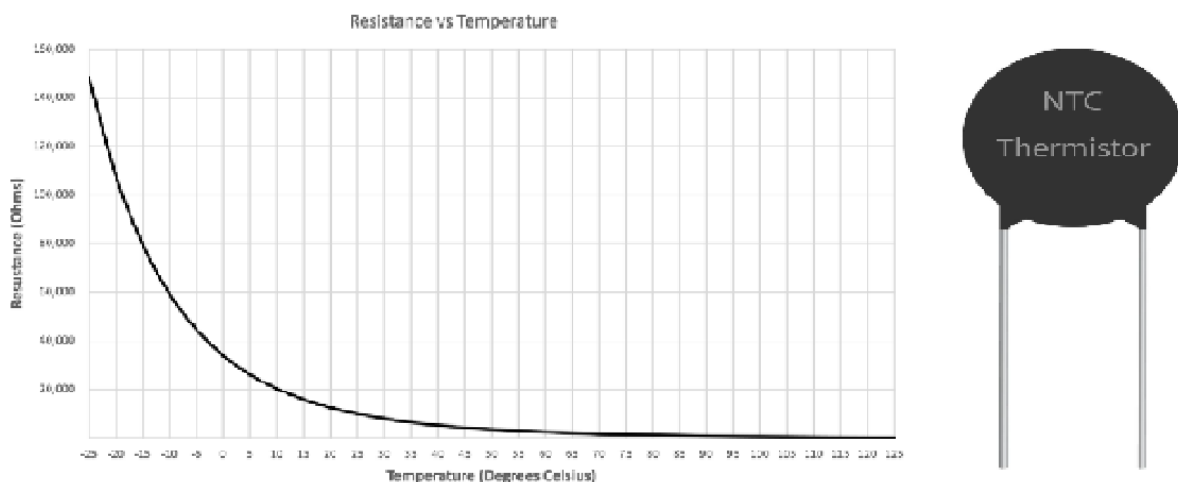
Σχήμα 5.20: Αισθητήρας θερμοκρασίας και υγρασίας DHT22

<https://grobotronics.com/rht03-dht22.html>

### 5.3.12.1 Λειτουργία αισθητήρα DHT22

Ο αισθητήρας DHT22 μπορεί να μετρήσει θερμοκρασίες από  $-40^{\circ}\text{C}$  έως  $+80^{\circ}\text{C}$  με ακρίβεια  $\pm 0.5^{\circ}\text{C}$  και υγρασία από 0% έως 100% RH με ακρίβεια  $\pm 2\%$ -5%, ενώ ενημερώνει τις μετρήσεις του κάθε 2 δευτερόλεπτα. Λειτουργεί με τάση 3.3V-5V. Ο ενσωματωμένος μικροελεγκτής του μετατρέπει τις αναλογικές μετρήσεις σε ψηφιακές τιμές και τις μεταδίδει μέσω μίας μόνο γραμμής δεδομένων, χρησιμοποιώντας το πρωτόκολλο Single Wire Interface (SWI), καθιστώντας την επικοινωνία με το Arduino απλή και αποδοτική.

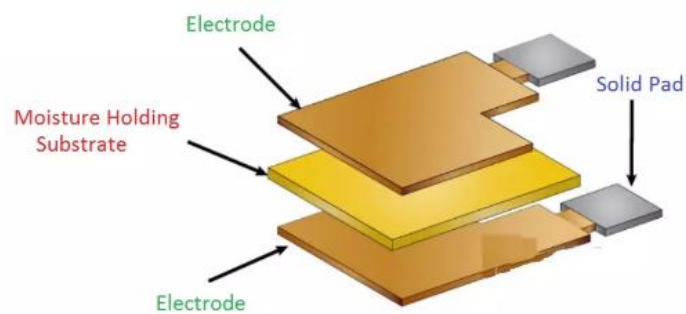
Για την μέτρηση της θερμοκρασίας, έχει μια θερμοαντίσταση με αρνητικό συντελεστή (Negative Temperature Coefficient-NTC) ή αλλιώς θερμίστορ (Thermistor) που είναι στην επιφάνεια της εσωτερικής πλακέτας. Τα θερμίστορ είναι αισθητήρες μεταβλητής αντίστασης και η αντίστασή τους μειώνεται με την αύξηση της περιβαλλοντικής θερμοκρασίας. [9] [10]



Σχήμα 5.21: Διάγραμμα σχέσης μεταξύ θερμοκρασίας και υγρασίας, Θερμίστορ

<https://fritzenlab.net/2024/06/12/ntc-thermistor-temperature-sensor-arduino/>

Για την μέτρηση της υγρασίας ο DHT22 χρησιμοποιεί έναν αισθητήρα χωρητικότητας με πολυμερικό υλικό το οποίο είναι ευαίσθητο στην υγρασία του περιβάλλοντος. Ο αισθητήρας αποτελείται από δύο ηλεκτρόδια και ένα στρώμα πολυμερικού υλικού ανάμεσά τους. Το πολυμερικό υλικό είναι υγροσκοπικό, που σημαίνει ότι απορροφά ή απελευθερώνει υγρασία από το περιβάλλον, ανάλογα με τα επίπεδα της σχετικής υγρασίας (Relative Humidity - RH). Όταν η υγρασία αυξάνεται, το πολυμερικό υλικό απορροφά υδατμούς αλλάζοντας την διηλεκτρική σταθερά του και η αλλαγή αυτή επηρεάζει τη χωρητικότητα του αισθητήρα. Η μεταβολή της χωρητικότητας ανιχνεύεται από τον μικροελεγκτή που είναι ενσωματωμένος στο DHT22 και χρησιμοποιεί τη μεταβολή αυτή για να υπολογίσει τη σχετική υγρασία (RH) ως ποσοστό %. [9] [10]



Σχήμα 5.22: Εσωτερική δομή του αισθητήρα υγρασίας DHT22

<https://www.theengineeringprojects.com/2019/02/introduction-to-dht22.html>

Πίνακας 5.12: Τεχνικά χαρακτηριστικά DHT22

Τάση λειτουργίας	3.3-6VDC
Σήμα εξόδου	Ψηφιακό σήμα μέσω ενιαίου διαύλου
Στοιχείο ανίχνευσης	Πυκνωτής πολυμερούς
Εύρος λειτουργίας	Υγρασία: 0-100% Θερμοκρασία: -40°C ~ 80°C
Ευαισθησία	Υγρασία: 0.1% RH Θερμοκρασία: 0.1°C
Επαναληψιμότητα	Υγρασία: $\pm 0.1\%$ RH Θερμοκρασία: $\pm 0.2^\circ\text{C}$
Περίοδος ανίχνευσης	2s

### 5.3.12.2 Συνδεσμολογία DHT22 με το Arduino

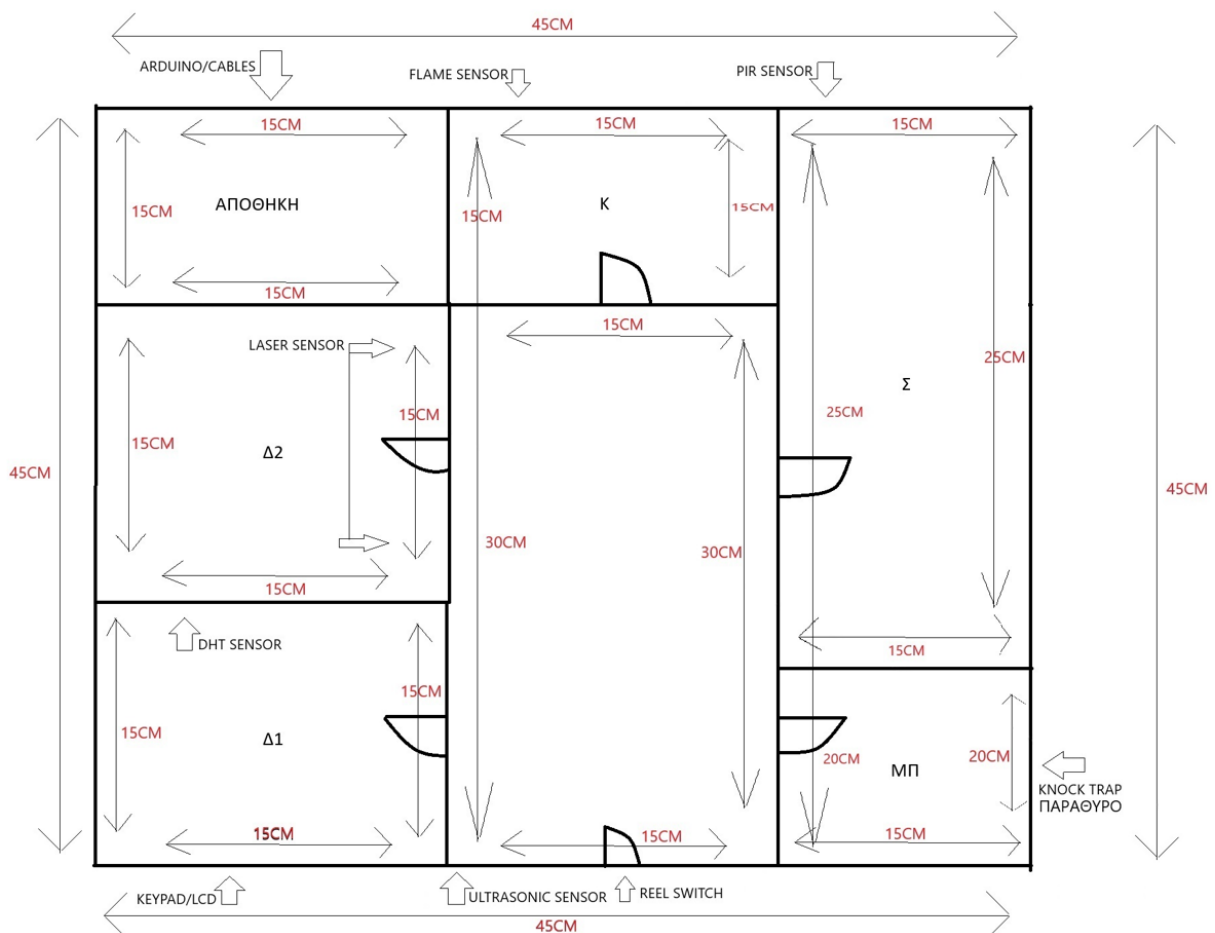
Πίνακας 5.13: Συνδεσμολογία με το Arduino

DHT22	Arduino
PIN 1	10KΩ (με το PIN 2) + 5V
PIN 2	Ακροδέκτης 11
PIN 3	-
PIN 4	Γείωση (GND)

### 5.3.13 Κατασκευή μακέτας

Για την κατασκευή της μακέτας χρησιμοποιήθηκε χαρτόνι μακέτας 100cm x70cm x 5mm. Οι διαστάσεις της μακέτας για την πτυχιακή εργασία, αποφασίστηκαν έπειτα από τον σχεδιασμό της κάτοψης σπιτιού.

Όπως φαίνεται στο Σχήμα 5.23, ο σχεδιασμός της κάτοψης έγινε στο πρόγραμμα Windows Paint και μελετήθηκε πού και πώς θα τοποθετηθούν οι αισθητήρες, οι καλωδιώσεις και τα επιμέρους εξαρτήματα.



Σχήμα 5.23: Μελέτη κάτοψης μακέτας

Στη συνέχεια φτιάχτηκε η βάση και τα τοιχώματα τα οποία κολλήθηκαν μεταξύ τους με κυανοακρυλική κόλλα και στηρίχθηκαν με πλαστικές γωνίες ραφιών. Έπειτα τοποθετήθηκαν όλα τα υλικά και οι καλωδιώσεις μπήκαν στα κανάλια.



Σχήμα 5.24: Κάτοψη μακέτας

## **Κεφάλαιο 6ο: Κώδικας**

### **6.1 Εισαγωγή**

Σε αυτό το κεφάλαιο θα αναλυθεί ο προγραμματισμός του Arduino MEGA 2560 στο περιβάλλον Arduino IDE, η συνολική λειτουργία του καθώς και τα σενάρια ασφαλείας ώστε να δούμε πως αντιδρά το σύστημα σε αυτά.

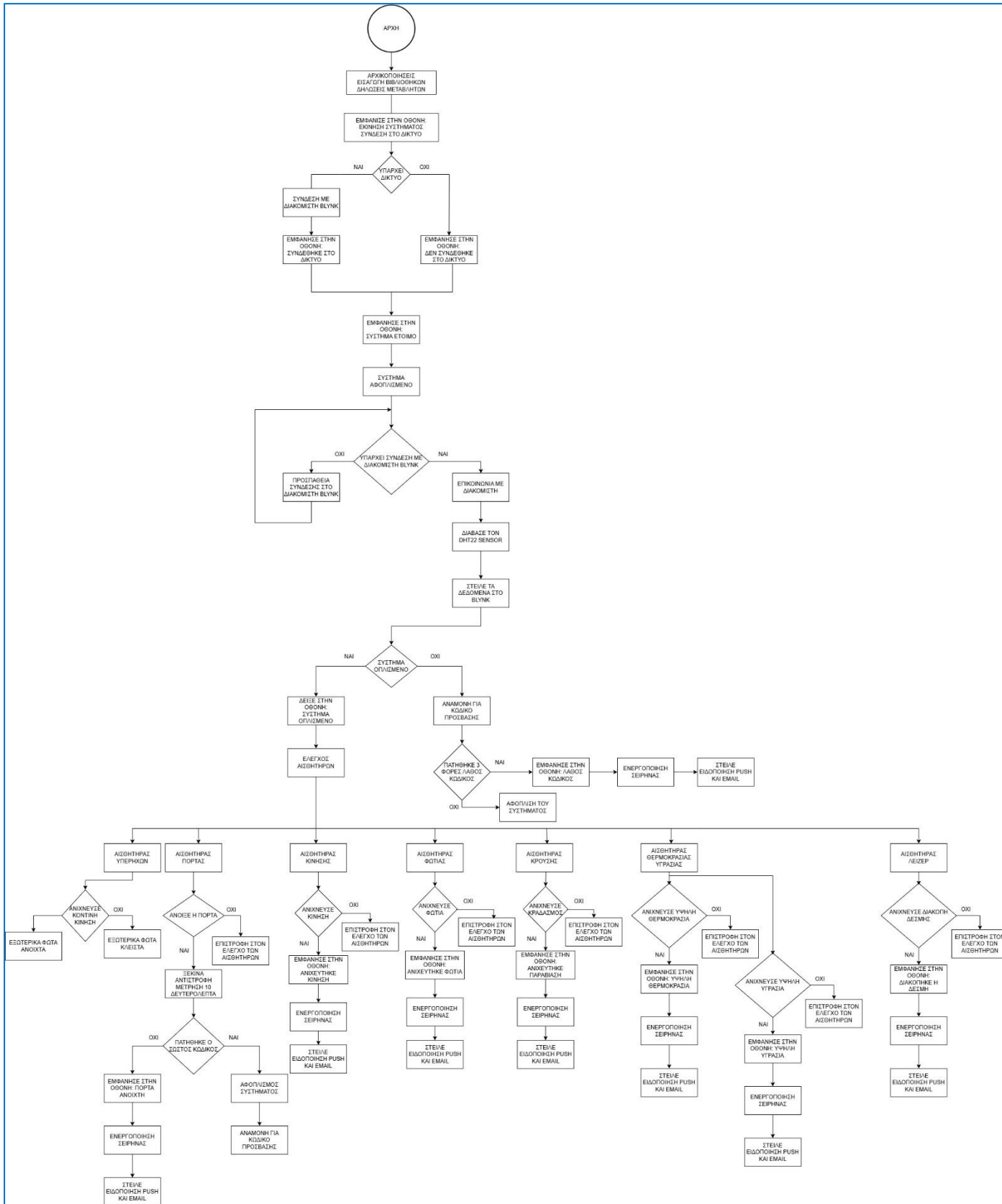
### **6.2 Κώδικας συστήματος συναγερμού**

Για τον προγραμματισμό του Arduino MEGA 2560 χρησιμοποιήθηκε το πρόγραμμα Arduino IDE το οποίο διατίθεται δωρεάν για λήψη.

Αρχικά δηλώθηκαν οι απαραίτητες πληροφορίες που χρειάζεται το Blynk για να εκτελέσει την επικοινωνία με το Arduino. Έπειτα προστέθηκαν οι βιβλιοθήκες για το hardware του συστήματος, τα διαπιστευτήρια για να συνδεθεί το ESP-01 στο Wi-Fi και ρυθμιστικό το baud rate του για να επικοινωνεί σωστά με το Arduino.

### **6.3 Διάγραμμα ροής**

Παρακάτω, είναι το διάγραμμα ροής του κώδικα στην γενική μορφή του.



Σχήμα 6.1: Διάγραμμα ροής

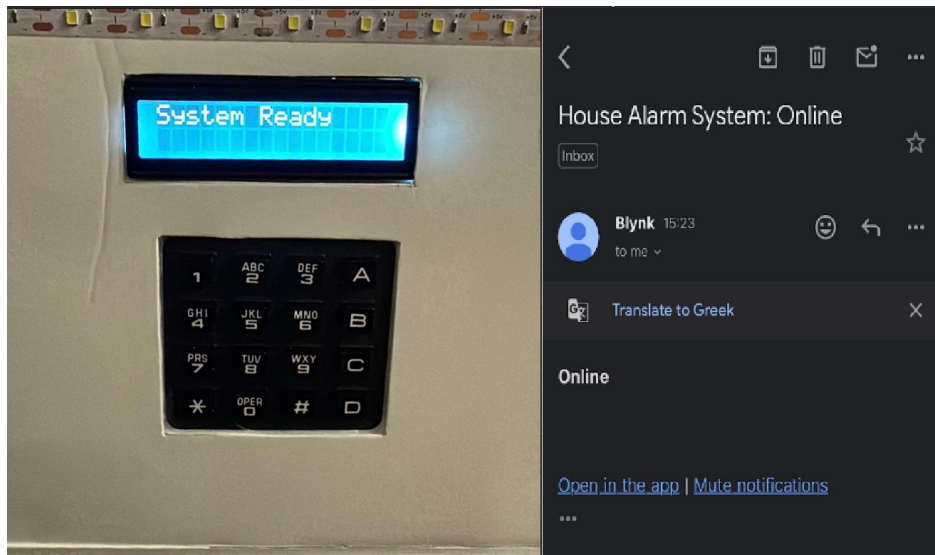
## 6.4 Λειτουργία συστήματος συναγερμού

Κατά την εκκίνηση του συστήματος εμφανίζονται στην οθόνη πληροφορίες σχετικά με την κατάσταση του Wi-Fi μέχρι να ολοκληρωθεί η αρχικοποίησή του.



Σχήμα 6.2: Αρχικοποίηση συστήματος και έλεγχος Wi-Fi

Μόλις εμφανιστεί στην οθόνη το μήνυμα “System Ready” όπως φαίνεται στο Σχήμα 6.3 τότε μπορεί ο χρήστης να βάλει τον κωδικό πρόσβασης και να τον οπλίσει. Παράλληλα, ειδοποιείται ο χρήστης με email και push ειδοποίηση από το Blynk ότι το σύστημα είναι έτοιμο για χρήση.



Σχήμα 6.3: Σύστημα έτοιμο, Ειδοποίηση Blynk

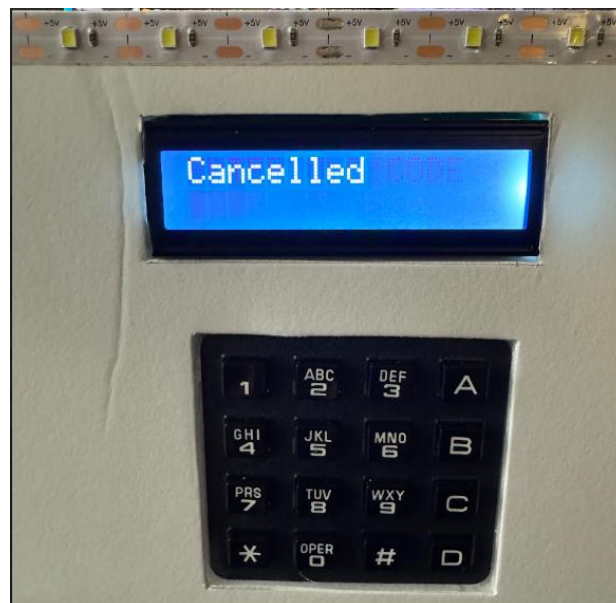
Στη συνέχεια ο χρήστης μπορεί να οπλίσει τον συναγερμό πληκτρολογώντας τον κωδικό πρόσβασης ή να αλλάξει σε νέο κωδικό πρόσβασης πατώντας το πλήκτρο «D». Η διαδικασία είναι πολύ απλή

καθώς πατώντας το πλήκτρο για αλλαγή κωδικού πρόσβασης το σύστημα καθοδηγεί τον χρήστη βήμα-βήμα με μηνύματα στην οθόνη όπως φαίνεται στο Σχήμα 6.4.



Σχήμα 6.4: Αλλαγή κωδικού πρόσβασης

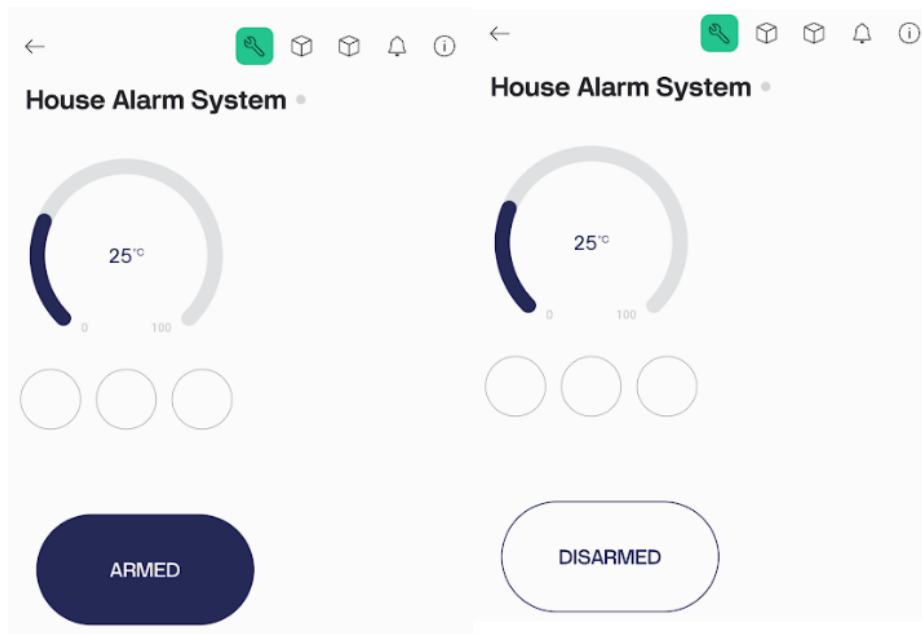
Στην περίπτωση που ο χρήστης πιέσει λάθος πλήκτρο, μπορεί να πιέσει το πλήκτρο «\*» για ακύρωση της διαδικασίας και έπειτα ξανά το «D» για εκ νέου αλλαγή κωδικού.



Σχήμα 6.5: Ακύρωση αλλαγής κωδικού

Αφού ο χρήστης ολοκληρώσει την διαδικασία αλλαγής κωδικού, ο σπλισμός του συστήματος μπορεί να γίνει με δύο τρόπους. Ο πρώτος τρόπος είναι να πληκτρολογήσει τον κωδικό από το πληκτρολόγιο και έπειτα να πιέσουμε το πλήκτρο «#». Ο δεύτερος τρόπος είναι να επιλέξει τον σπλισμό του συστήματος από την εφαρμογή Blynk στο κινητό τηλέφωνο. Και στις δυο περιπτώσεις, γίνεται 15 δευτερόλεπτα αντίστροφη μέτρηση μέχρι το σύστημα να σπλίσει.

Για να αφοπλίσει το σύστημα συναγερμού, υπάρχουν δύο τρόποι με παρόμοια διαδικασία. Ο πρώτος τρόπος είναι, αφού εισέλθει στον χώρο από την είσοδο που έχει δηλωθεί ως κεντρική, το σύστημα παρέχει 15 δευτερόλεπτα για να πληκτρολογήσει τον κωδικό και να πιάσει το «#» για να αφοπλίσει. Ο δεύτερος τρόπος είναι να επιλέξει τον αφοπλισμό από την εφαρμογή Blynk στο κινητό τηλέφωνο πριν εισέλθουμε στον χώρο.



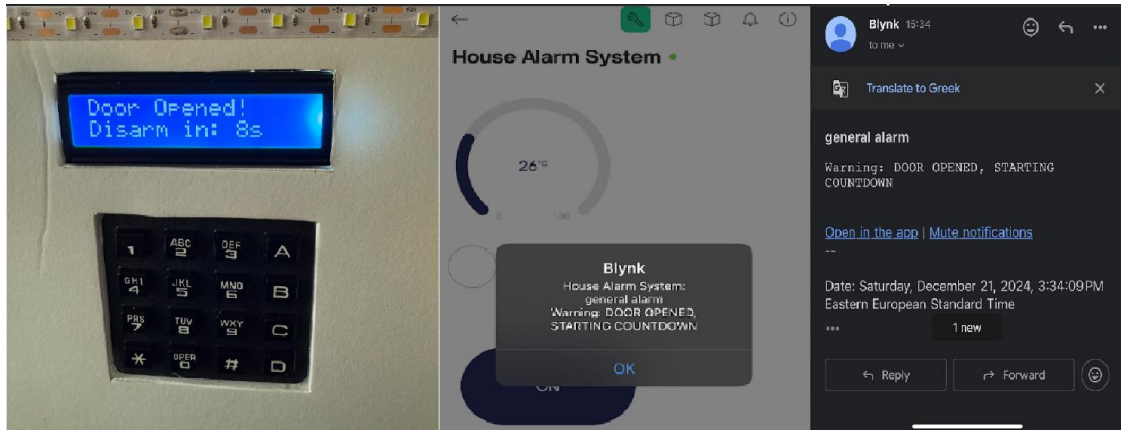
Σχήμα 6.6: Όπλιση και αφοπλιση συστήματος συναγερμού

## 6.5 Σενάρια

Έχει οριστεί ότι η ζώνη της κεντρικής εισόδου έχει χρόνο εισόδου πριν χτυπήσει ο συναγερμός, ενώ οι άλλες ζώνες θα ενεργοποιούν κατευθείαν τον συναγερμό. Για την σωστή λειτουργία του συστήματος συναγερμού θα δοκιμαστούν 6 σενάρια. Σε όλα τα σενάρια το σύστημα θα είναι οπλισμένο. Στο πρώτο θα διεγερθεί η μαγνητική επαφή της κεντρικής εισόδου, στο δεύτερο σενάριο θα διεγερθεί ο αισθητήρας φωτιάς, στο τρίτο θα αυξηθεί η υγρασία στον χώρο, στο τέταρτο θα ενεργοποιηθεί ο αισθητήρας κίνησης, στο πέμπτο θα διακοπή η δέσμη λέιζερ και στο έκτο θα διεγερθεί ο αισθητήρας κρούσης στο παράθυρο.

### 6.5.1 Πρώτο σενάριο

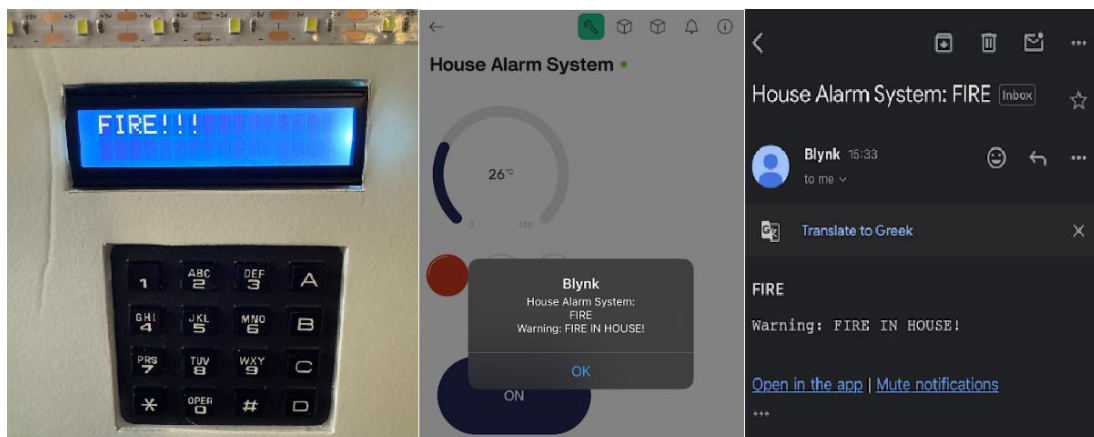
Σε αυτό το σενάριο, ένας διαρρήκτης πλησιάζει την κεντρική είσοδο και ανάβουν αυτόματα τα εξωτερικά φώτα περιμετρικά του σπιτιού. Ανοίγει την κεντρική είσοδο και ενεργοποιείται η αντίστροφη μέτρηση 15 δευτερολέπτων στον συναγερμό. Παράλληλα έρχεται στον χρήστη email και push ειδοποίηση ότι η κεντρική πόρτα είναι ανοιχτή. Στο τέλος της αντίστροφης μέτρησης ενεργοποιείται η σειρήνα.



Σχήμα 6.7: Ενεργοποίηση αντίστροφης μέτρησης και ειδοποιήσεις

### 6.5.2 Δεύτερο σενάριο

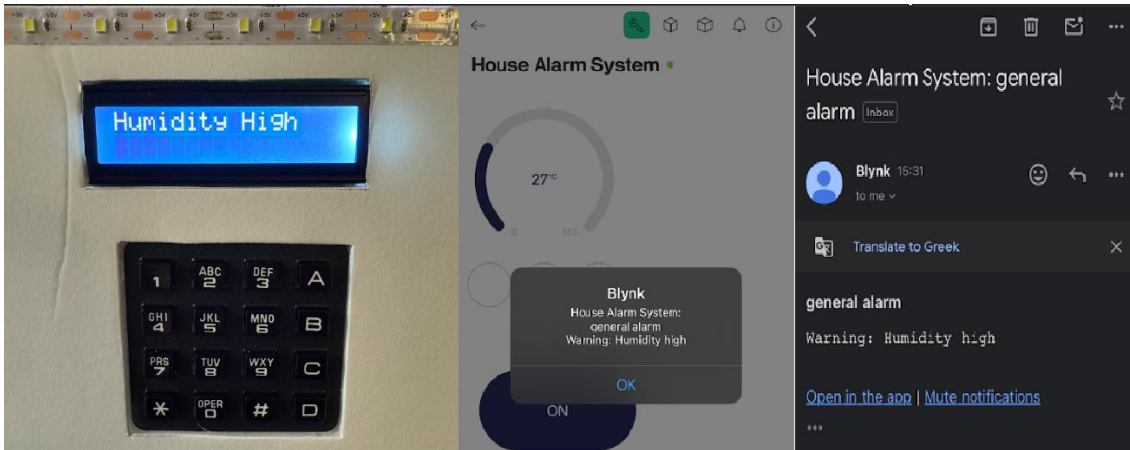
Αυτό το σενάριο προσομοιώνει μία πιθανή φωτιά στον χώρο που έχει προκληθεί είτε εσκεμμένα είτε από άλλη αιτία. Μόλις ο ανιχνευτής φωτιάς ενεργοποιηθεί, στέλνει με email και push ειδοποίηση στον χρήστη την ειδοποίηση φωτιάς και ενεργοποιεί την σειρήνα.



Σχήμα 6.8: Ειδοποίηση φωτιάς

### 6.5.3 Τρίτο σενάριο

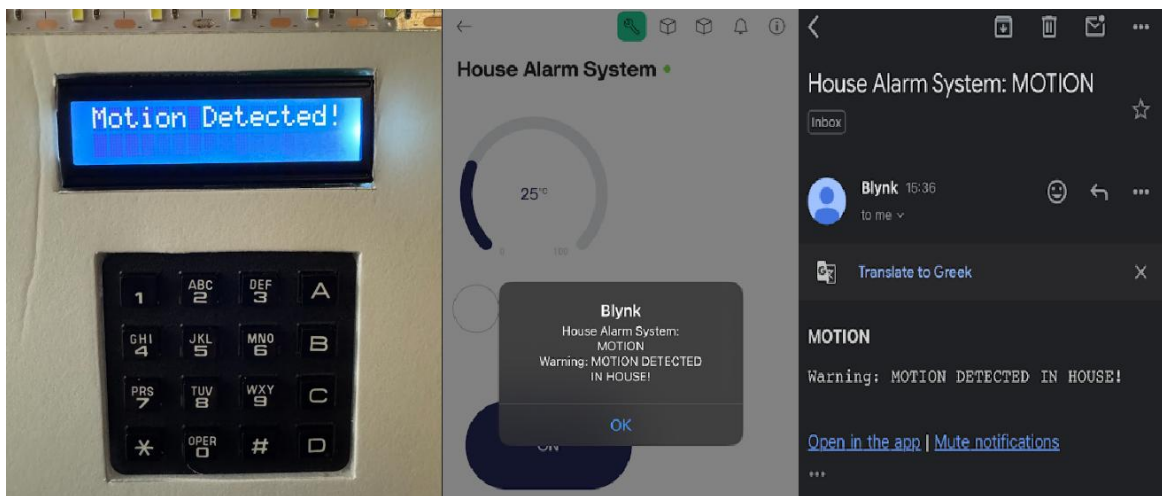
Σε αυτό το σενάριο θα προσομοιωθεί το αυξημένο ποσοστό υγρασίας στο χώρο, ενδεχομένως λόγω μιας πλημμύρας που προκλήθηκε. Μόλις ο αισθητήρας υγρασίας ανιχνεύσει πάνω από 75% υγρασία στο χώρο, θα στείλει με email και push ειδοποίηση στον χρήστη την ειδοποίηση αυξημένης υγρασίας και ενεργοποιήσει την σειρήνα.



Σχήμα 6.9: Ειδοποίηση υψηλής υγρασίας

#### 6.5.4 Τέταρτο σενάριο

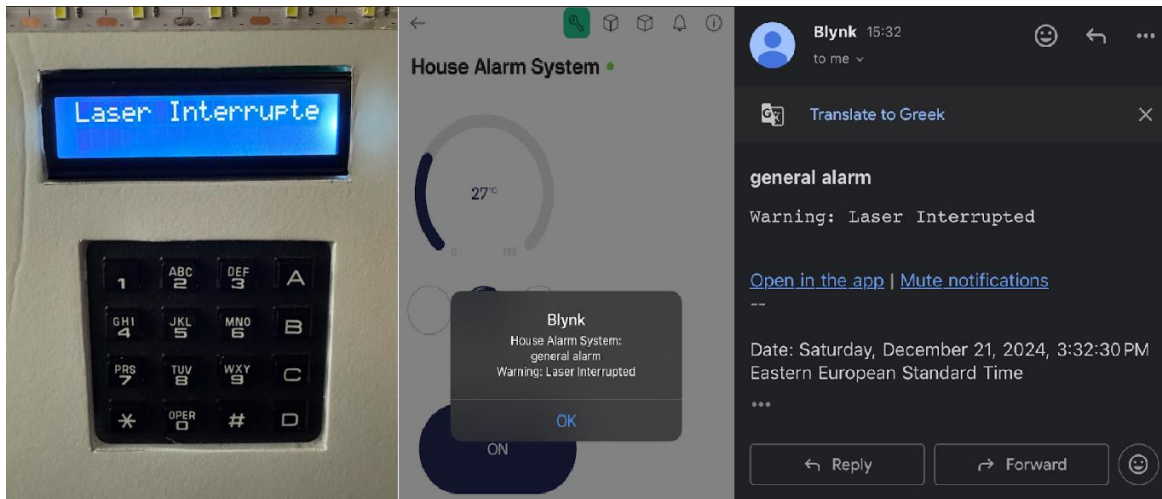
Σε αυτό το σενάριο, ένας διαρρήκτης εισέρχεται στον χώρο και ενεργοποιεί τον αισθητήρα κίνησης. Κατευθείαν ενεργοποιείται η σειρήνα και στέλνει με email και push ειδοποίηση στον χρήστη την ειδοποίηση ότι κάποιος έχει εισβάλει στον χώρο.



Σχήμα 6.10: Ειδοποίηση για κίνηση στο χώρο

#### 6.5.5 Πέμπτο σενάριο

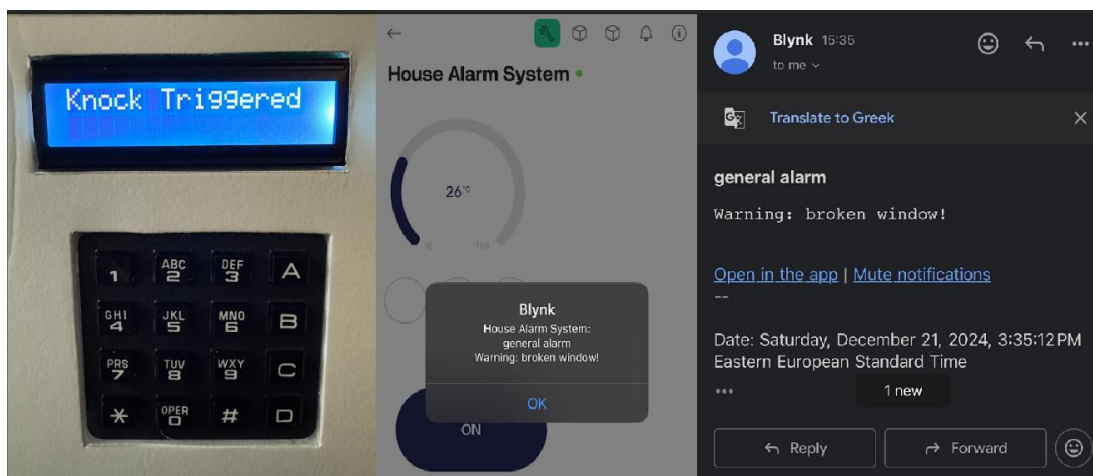
Σε αυτό το σενάριο, ένας διαρρήκτης εισέρχεται στον χώρο και διακόπτει την δέσμη λέιζερ. Κατευθείαν ενεργοποιείται η σειρήνα και στέλνει με email και push ειδοποίηση στον χρήστη την ειδοποίηση ότι κάποιος έχει εισβάλει στον χώρο.



Σχήμα 6.11: Ειδοποίηση από διακοπή δέσμης λέιζερ

### 6.5.6 Έκτο σενάριο

Σε αυτό το σενάριο, ένας διαρρήκτης σπάει το παράθυρο για να εισβάλει στον χώρο προκαλώντας την διέγερση του αισθητήρα κρούσης. Κατευθείαν ενεργοποιείται η σειρήνα και στέλνει με email και push ειδοποίηση στον χρήστη την ειδοποίηση ότι κάποιος έχει σπάσει το παράθυρο.



Σχήμα 6.12: Ειδοποίηση για διάρρηξη παραθύρου

## 6.6 Επίλογος

Σε αυτό το κεφάλαιο, αρχικά αναφέρθηκε πως έγινε ο προγραμματισμός του Arduino MEGA και σε ποιο περιβάλλον. Έπειτα, παρουσιάστηκε το διάγραμμα ροής του κώδικα που έχει σημαντικό ρόλο για την κατανόηση του προγράμματος και τέλος τέθηκε σε λειτουργία το σύστημα συναγερμού και έγιναν κάποια σενάρια για την σωστή λειτουργία του.

## Κεφάλαιο 7ο: Συμπεράσματα ή/και προτάσεις βελτίωσης

Για την εκπόνηση της πτυχιακής εργασίας, προηγήθηκαν συζητήσεις με τον επιβλέποντα καθηγητή για την επιλογή του θέματος. Έπειτα, τέθηκαν με την σειρά οι στόχοι που πρέπει να επιτευχθούν για την ολοκλήρωση της. Στη συνέχεια, ακολούθησε έρευνα για την επιλογή του μικροελεγκτή καθώς και για τα περιφερειακά του, τα υλικά κατασκευής της μακέτας και της πλακέτας.

Το κύριο πρόβλημα που υπήρχε στην αρχή της εργασίας ήταν η επικοινωνία του Arduino MEGA με το ESP01 καθώς το ESP01 δεν έρχεται με ενσωματωμένο firmware. Ύστερα από εκτεταμένη έρευνα βρέθηκε η λύση με ένα USB Module για το ESP01 το οποίο κάνει εσωτερικά τις απαραίτητες ενώσεις των ακίδων για να μπει το σε λειτουργία προγραμματισμού όταν συνδεθεί στον υπολογιστή. Στη συνέχεια με τον υπολογιστή εγκαθίσταται εύκολα και γρήγορα το λογισμικό του και γίνονται οι απαραίτητες ρυθμίσεις στις ζώνες επικοινωνίας για να λειτουργήσει σωστά με το Arduino.

Το κόστος για όλη την κατασκευή ανέρχεται συνολικά στα 116,05€. Ακριβότερο υλικό ήταν η κυανοακρυλική κόλλα, το χαρτί μακέτας και το Arduino, ενώ ο κόστος των υπόλοιπων υλικών ήταν μικρότερο από 5€.

Το συμπέρασμα από την παρούσα πτυχιακή εργασία είναι ότι με μπορούν όλοι να υλοποιήσουν ή να αγοράσουν έτοιμο ένα σύστημα συναγερμού ή ένα σύστημα ασφαλείας προσαρμοσμένο ακριβώς στις ανάγκες τους, σε πολλές περιπτώσεις με αρκετά χαμηλό κόστος.

Τέλος, θα μπορούσαν να υπάρξουν μελλοντικές βελτιώσεις σε αυτήν την κατασκευή, όπως για παράδειγμα να προστεθεί στο σύστημα ένας ανιχνευτής καπνού, σε συνεργασία με τον ανιχνευτή φλόγας, για πιο ακριβής ειδοποιήσεις πυρκαγιάς. Επιπλέον, θα μπορούσε το σύστημα να συνδεθεί απευθείας με την τηλεφωνική γραμμή του σπιτιού για να δέχεται ο χρήστης τηλεφωνικές κλήσεις όταν ανιχνευτεί κάτι από το σύστημα για πιο άμεση ενημέρωση. Ακόμα μια βελτίωση που θα μπορούσε να γίνει, είναι να προστεθεί ένα RFID σύστημα με το οποίο θα μπορεί ο χρήστης να οπλίζει και να αποπλίζει τον συναγερμό με μία προγραμματιζόμενη κάρτα εισόδου για μεγαλύτερη ευκολία. Κάτι ακόμα που θα μπορούσε να γίνει, είναι να βελτιωθεί και να εμπλουτισθεί ο κώδικας.



# ΒΙΒΛΙΟΓΡΑΦΙΑ

## Βιβλία

- [1] Μαγκανιάρη Α. Μαρία, *ΣΥΣΤΗΜΑΤΑ ΕΛΕΓΧΟΥ ΚΑΙ ΑΣΦΑΛΕΙΑΣ*. Γ'ΕΠΙΑΛ: ΙΝΣΤΙΤΟΥΤΟ ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ ΚΑΙ ΕΚΔΟΣΕΩΝ «ΔΙΟΦΑΝΤΟΣ»  
[11] Σταύρος Καμινάρης, *ΣΥΣΤΗΜΑΤΑ ΣΥΝΑΓΕΡΜΟΥ – ΑΣΦΑΛΕΙΑΣ*. ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ: ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

## Data Sheet

- [5] ArduinoModules, “KY-008 laser transmitter module,” datasheet, Oct 2021  
[4] Microchip, “ESP-01 WiFi Module,” datasheet  
[6] ALLDATASHEET, “HC-SR501 PIR MOTION DETECTOR,” datasheet  
[7] ALLDATASHEET, “BISS0001 PIR CONTROLER,” datasheet  
[8] Handson Technology, “HC-SR04 Ultrasonic Sensor Module User Guide,” datasheet  
[9] Aosong Electronics CoLtd, “DHT22 sensor module,” datsheet  
[13] ArduinoModules, “KY-026 FLAME SENSOR MODULE FRITZING PART,” datasheet, Oct 2021  
[16] Adafruit, “Matrix Keypad,” datasheet Jun 2024

## Internet Site

- [2] Last Minute ENGINEERS, “Interface an I2C LCD with Arduino,” [Online]. Available: <https://lastminuteengineers.com/i2c-lcd-arduino-tutorial/>  
[3] DEEPBLUEMBEDDED, “Arduino I2C LCD 16x2 Interfacing,” [Online]. Available: <https://deepbluembedded.com/arduino-i2c-lcd/>  
[10] THE ENGINEERING PROJECTS, “Introduction to DHT22,” [Online]. Available: <https://www.theengineeringprojects.com/2019/02/introduction-to-dht22.html>  
[12] DIVICO SECURITY, “ΣΥΣΤΗΜΑΤΑ ΠΥΡΑΝΙΧΝΕΥΣΗΣ,” [Online]. Available: <https://divico.gr/proionta/sistimata-piranixneusis/>  
[14] Blynk.Documentation, “Introduction to Blynk,” [Online]. Available: [Introduction | Blynk Documentation](#)  
[15] Arduino, “What is Arduino,” [Online]. Available: [What is Arduino? | Arduino](#)

## ΠΑΡΑΡΤΗΜΑ Α: Κατάλογος υλικών

Προϊόν	Τιμή	Ποσότητα	Υποσύνολο
Arduino MEGA 2560	17€	1	17€
16x2 I2C Οθόνη	4€	1	4€
Buzzer	0,5€	1	0,5€
Πληκτρολόγιο	3€	1	3€
Jumper Wires set	2,5€	4	10€
ESP01	2,5€	1	2,5€
ESP01 USB Module	1,7€	1	1,7€
Ultrasonic sensor	2,4€	1	2,4€
DHT22 sensor	1,13€	1	1,13€
KY-026 Flame sensor	3,2€	1	3,2€
Laser sensor	1,97€	1	1,97€
Laser receiver	0,97€	1	0,97€
Αισθητήρας κρούσης	0,33€	1	0,33€
Αντίσταση 2ΚΩ	0,02	4	0,08€
PIR sensor	3€	1	3€
Reed switch	0,73€	1	0,73€
Pin header 1x8	0,2€	1	0,2€
Χαρτόνι μακέτας 70x100cm 5mm	7,85€	2	15,70€
Πλακέτα διάτρητη	0,3€	1	0,3€
Θερμοσυστελόμενα set	11€	1	11€
Ταινία LED	2,9€	1	2,9€
Κανάλι διανομής Αυτοκόλλητο 12X13mm	3,44€	1	3,44€
Καλάι 0,5mm	5€	1	5€
Κυανοακρυλική κόλλα	25€	1	25€
		<b>Σύνολο:</b>	116,05€

## ΠΑΡΑΡΤΗΜΑ Β: Κώδικας

```
#define BLYNK_TEMPLATE_ID "TMPL4hdeoLS8M"
#define BLYNK_TEMPLATE_NAME "test1"
#define BLYNK_AUTH_TOKEN "kcqN_SQGrMMFq4odnTyEC1tyTs8y6oUp"
#define BLYNK_PRINT Serial

#include <NewPing.h>
#include <ESP8266_Lib.h>
#include <BlynkSimpleShieldEsp8266.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h> // For I2C LCD
#include <Keypad.h> // For Keypad
#include <DHT.h> // For DHT22 sensor

// WiFi credentials.
char ssid[] = "Charis";
char pass[] = "haris123";

// Hardware Serial on Mega
#define EspSerial Serial1

// Your ESP8266 baud rate
#define ESP8266_BAUD 38400

ESP8266 wifi(&EspSerial);

// LCD settings
LiquidCrystal_I2C lcd(0x27, 16, 2); // Set the LCD I2C address (0x27 or 0x3F)

// Keypad settings
const byte ROWS = 4; // Four rows
const byte COLS = 4; // Four columns
char keys[ROWS][COLS] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};

byte rowPins[ROWS] = {10, 9, 8, 7}; // Rows connected to pins 9, 8, 7, 6
byte colPins[COLS] = {6, 5, 4, 3}; // Columns connected to pins 5, 4, 3, 2
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);

// Magnetic door switch, buzzer, DHT sensor, laser module, laser sensor, outdoor
light, PIR sensor pins
#define DOOR_SWITCH_PIN 32 // Magnetic door switch is connected to pin 32
```

```

#define BUZZER_PIN 12 // Buzzer is connected to pin 12
#define DHTPIN 11 // DHT22 sensor is connected to pin 11
#define DHTTYPE DHT22 // DHT22 sensor
#define LASER_MODULE_PIN 28 // Laser module connected to pin 28
#define LASER_SENSOR_PIN 29 // Laser sensor connected to pin 29
#define OUTDOOR_LIGHT_PIN 2 // Outdoor light connected to pin 2
#define PIR_SENSOR_PIN 31 // PIR sensor connected to pin 31
#define KNOCK_TRAP_PIN 30 // Pin 30 connected to the knock trap
#define TRIG_PIN 22 // Trig pin for HC-SR04 connected to pin 22
#define ECHO_PIN 24 // Echo pin for HC-SR04 connected to pin 24
#define FLAME_SENSOR_PIN 33 // Flame sensor KY-026 connected to pin 33
#define FLAME_SENSOR_ANALOG_PIN A0 // Analog input flame sensor in PIN A0
#define PIR_SENSOR_PIN 31 //PIR sensor pin 31
DHT dht(DHTPIN, DHTTYPE);

// Knock trap
bool knockTriggered = false; // Tracks if the knock trap has been triggered
unsigned long lastKnockTime = 0; // Last time the knock was detected (for
debouncing)
const unsigned long knockDebounceDelay = 1; // Reduce to 1ms for more
responsiveness
unsigned long knockWindow = 1; // Time window for detecting multiple knocks
int knockCount = 0; // Counter for consecutive knocks
const int knockThreshold = 1; // Number of knocks required to trigger

bool alarmArmed = false; // Tracks if the alarm is armed
bool doorOpened = false; // Tracks if the door is opened
bool lastDoorState = HIGH; // Tracks the last state of the magnetic door
switch
String inputPassword = ""; // String to hold entered password
String correctPassword = "0000"; // Default password
int failedAttempts = 0; // Counter for incorrect password attempts

bool alarmTriggered = false; // Indicates if the alarm is triggered
bool countdownActive = false; // Tracks if the countdown is active

void setup() {
// Initialize the LCD
lcd.begin(16, 2);
lcd.backlight();
lcd.setCursor(0, 0);
lcd.print("Initializing...");

// Initialize the DHT sensor
dht.begin();

```

```

// Initialize the door switch, laser module, laser sensor, buzzer, outdoor
light, and PIR sensor pins
pinMode(DOOR_SWITCH_PIN, INPUT_PULLUP); // Magnetic switch (closed = LOW,
opened = HIGH)
pinMode(LASER_MODULE_PIN, OUTPUT); // Laser module pin
pinMode(LASER_SENSOR_PIN, INPUT); // Laser sensor pin
pinMode(BUZZER_PIN, OUTPUT);
pinMode(OUTDOOR_LIGHT_PIN, OUTPUT); // Outdoor light pin
pinMode(PIR_SENSOR_PIN, INPUT); // PIR sensor pin
pinMode(FLAME_SENSOR_PIN, INPUT); // Ορισμός του ακροδέκτη του αισθητήρα
φωτιάς ως είσοδο
digitalWrite(BUZZER_PIN, LOW); // Buzzer off initially
digitalWrite(LASER_MODULE_PIN, HIGH); // Turn on the laser module
digitalWrite(OUTDOOR_LIGHT_PIN, LOW); // Ensure outdoor light is off
initially

// Initialize the ultrasonic sensor pins
pinMode(TRIG_PIN, OUTPUT); // Trig pin
pinMode(ECHO_PIN, INPUT); // Echo pin

// Initialize the knock trap pin
pinMode(KNOCK_TRAP_PIN, INPUT_PULLUP); // Set knock trap pin as input with
internal pull-up

// Debug console
Serial.begin(9600);

// Set ESP8266 baud rate
EspSerial.begin(ESP8266_BAUD);
delay(10);
Blynk.begin(BLYNK_AUTH_TOKEN, wifi, ssid, pass);
lcd.clear();
lcd.print("Blynk/Wi-Fi OK");
delay(1500);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("System Ready");
}

void loop() {

handleKeypadInput(); // Check for keypad input

handleKnockTrap(); // Check knock sensor

checkPirSensor(); // Check the PIR sensor

```

```

checkFlameSensor(); //Check the flame sensor

// Check the status of the magnetic door switch
doorOpened = digitalRead(DOOR_SWITCH_PIN) == HIGH; // HIGH means the door is
opened (triggered)

if (alarmArmed && doorOpened && !alarmTriggered && !countdownActive) {
  // Trigger the alarm with a 10-second countdown before the buzzer is enabled
  countdownActive = true;
  startCountdown(10); // 10-second countdown
}

// Check the laser sensor (if laser beam is cut off, sensor reads HIGH)
if (alarmArmed && digitalRead(LASER_SENSOR_PIN) == HIGH && !alarmTriggered) {
  triggerAlarm("Laser Interrupted!");
  alarmTriggered = true;
  Blynk.logEvent("gen", "Warning: Laser Interrupted");
}

// Check DHT22 temperature and humidity
float temperature = dht.readTemperature();
float humidity = dht.readHumidity();
Blynk.virtualWrite(V11, temperature);
if (alarmArmed && temperature > 30.0 && !alarmTriggered) {
  triggerAlarm("Temperature High");
  alarmTriggered = true;
  Blynk.logEvent("gen", "Warning: temperature high");
}

if (alarmArmed && humidity > 75.0 && !alarmTriggered) {
  triggerAlarm("Humidity High");
  alarmTriggered = true;
  Blynk.logEvent("gen", "Warning: Humidity high");
}

// Check distance from the ultrasonic sensor
float distance = getDistance();
if (distance <= 30.0 && distance > 0) {
  digitalWrite(OUTDOOR_LIGHT_PIN, HIGH); // Turn on the outdoor light
} else {
  digitalWrite(OUTDOOR_LIGHT_PIN, LOW); // Turn off the outdoor light
}
if ( alarmArmed &&((humidity > 75.0) || (temperature > 30.0) ||
(digitalRead(LASER_SENSOR_PIN) == HIGH) || (knockTriggered == true)) ) {
  Blynk.virtualWrite(V1, 1) ;
} else {
  Blynk.virtualWrite(V1, 0) ;
}

```

```

    }
}

BLYNK_WRITE(V52) {
    alarmArmed = param.asInt(); // Port V52 Arm/Disarm via Blynk

    if (alarmArmed) {
        lcd.clear();
        lcd.print("ARMED");
    } else {
        lcd.clear();
        lcd.print("DISARMED");
        alarmTriggered = false;
        knockTriggered = false;
        digitalWrite(BUZZER_PIN, LOW); // Turn off buzzer
    }
}

// Function to check the PIR sensor for motion detection
void checkPirSensor() {
    // Check if the PIR sensor detects motion (HIGH signal)
    char key = keypad.getKey();
    if (alarmArmed && digitalRead(PIR_SENSOR_PIN) == HIGH) {
        triggerAlarm("Motion Detected!");
        Blynk.virtualWrite(V31, PIR_SENSOR_PIN);
        Blynk.logEvent("motion", "Warning: MOTION DETECTED IN HOUSE!");
        {
            if (key >= '0' && key <= '9') {
                inputPassword += key;
                lcd.setCursor(0, 1);
                lcd.clear();
                lcd.print("Enter: ");
                lcd.print(inputPassword);
            }
        }
    } else { // If no move detect
        Blynk.virtualWrite(V31, 0); // Reset alarm on Blynk
    }
}

// Function to handle knock sensor input
void handleKnockTrap() {
    if (digitalRead(KNOCK_TRAP_PIN) == LOW) { // Knock detected (LOW when triggered)
        lastKnockTime = millis();
        while (digitalRead(KNOCK_TRAP_PIN) == LOW) { // Knock detected (LOW when triggered)

```

```

unsigned long currentTime = millis();

if (currentTime - lastKnockTime > knockDebounceDelay) {
    knockCount++; // Register a knock
    lastKnockTime = currentTime;
}
if (knockCount >= knockThreshold) { // If enough knocks detected in the
window
    knockTriggered = true;
    if (alarmArmed) {
        triggerAlarm("Knock Triggered");
        Blynk.logEvent("gen", "Warning: broken window!");
    }
    knockCount = 0; // Reset knock counter
}
// Reset knock count if time window expires
if (millis() - lastKnockTime > knockWindow) {
    knockCount = 0;
}
}
}
}

// Function to handle keypad input
void handleKeypadInput() {
char key = keypad.getKey();
static bool changePasswordMode = false; // Flag to track password change mode
static String tempPassword = ""; // Temporary storage for new password

if (key) {
    if (!changePasswordMode) {
        if (key >= '0' && key <= '9') {
            inputPassword += key;
            lcd.setCursor(0, 1);
            lcd.clear();
            lcd.print("Enter: ");
            lcd.print(inputPassword);
        }

        if (key == '#') {
            if (inputPassword == correctPassword) {
                alarmArmed = !alarmArmed;
                alarmTriggered = false; // Reset alarm trigger
                countdownActive = false; // Reset countdown
                digitalWrite(BUZZER_PIN, LOW); // Turn off buzzer
                lcd.clear();
                lcd.setCursor(0, 0);
            }
        }
    }
}
}

```

```

    lcd.print(alarmArmed ? "Armed" : "Disarmed");
    Blynk.virtualWrite(V52, alarmArmed); // Update state on Blynk
} else {
    failedAttempts++;
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Wrong Password!");
    if (failedAttempts >= 3) {
        triggerAlarm("3 Failed Attempts");
        Blynk.logEvent("gen", "Warning: 3 times wrong password");
        failedAttempts = 0;
    }
}
inputPassword = ""; // Clear password after attempt
}

if (key == '*') {
    inputPassword = ""; // Clear input if * is pressed
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Password Cleared");
}

if (key == 'D') { // Initiate password change
    lcd.clear();
    lcd.print("ENTER OLD CODE");
    inputPassword = ""; // Clear inputPassword
    changePasswordMode = true; // Enable password change mode
}
} else { // Password change mode is active
if (key >= '0' && key <= '9') {
    inputPassword += key;
    lcd.setCursor(0, 1);
    lcd.clear();
    lcd.print("Enter: ");
    lcd.print(inputPassword);
}

if (key == '#') {
    if (tempPassword.length()==0) { // Verifying old password
        if (inputPassword == correctPassword) {
            lcd.clear();
            lcd.print("ENTER NEW CODE");
            inputPassword = ""; // Clear inputPassword for new code
            tempPassword = "valid"; // Mark old code as valid
        } else {
            lcd.clear();
            lcd.print("Wrong Password!");

```

```

        changePasswordMode = false; // Exit change password mode
        inputPassword = ""; // Clear inputPassword
    }
} else { // Saving new password
    correctPassword = inputPassword; // Update the correct password
    lcd.clear();
    lcd.print("Code Changed!");
    delay(1500);
    lcd.clear();
    lcd.print("System Ready");
    changePasswordMode = false; // Exit change password mode
    tempPassword = ""; // Clear tempPassword
    inputPassword = ""; // Clear inputPassword
}
}

if (key == '*') { // Cancel password change
    changePasswordMode = false;
    tempPassword = "";
    inputPassword = "";
    lcd.clear();
    lcd.print("Cancelled");
    delay(1000);
    lcd.clear();
    lcd.print("System Ready");
}
}
}

// Function to trigger the alarm
void triggerAlarm(String reason) {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(reason); // Display the reason for the alarm
    digitalWrite(BUZZER_PIN, HIGH); // Activate the buzzer
    alarmTriggered = true;
}

// Function to start countdown before triggering alarm and handle keypad input
void startCountdown(int seconds) {
    unsigned long previousMillis = 0; // Tracks the last time a second passed
    unsigned long currentMillis; // Tracks the current time
    int remainingSeconds = seconds; // Countdown time
    inputPassword = ""; // Clear the password input
    bool passwordStarted = false; // Flag to track if password entry has
started

```

```

// Start the countdown loop
while (remainingSeconds >= 0) {
    currentMillis = millis(); // Get the current time

    // Check if a second has passed (non-blocking delay using millis)
    if (currentMillis - previousMillis >= 1000) {
        previousMillis = currentMillis;
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("Door Opened!");
        lcd.setCursor(0, 1);
        lcd.print("Disarm in: ");
        lcd.print(remainingSeconds);
        lcd.print("s");
        remainingSeconds--;
        Blynk.virtualWrite(V1, DOOR_SWITCH_PIN); //DOOR ON BLYNK
        Blynk.logEvent("gen", "Warning: DOOR OPENED, STARTING COUNTDOWN");
        Blynk.virtualWrite(V1, 0); // Reset the alarm on Blynk
    }

    // Check if a key is pressed during the countdown
    char key = keypad.getKey();
    if (key) {
        if (key >= '0' && key <= '9') { // Only append number keys to the password
            inputPassword += key;
        }

        if (key == '*') { // Clear password if '*' is pressed
            inputPassword = "";
        }

        // If '#' is pressed, check the password
        if (key == '#') {
            if (inputPassword == correctPassword) {
                alarmArmed = false; // Disarm the system
                alarmTriggered = false;
                countdownActive = false;
                inputPassword = ""; // Clear the password after successful disarm
                lcd.clear();
                lcd.setCursor(0, 0);
                lcd.print("Disarmed");
                digitalWrite(BUZZER_PIN, LOW); // Turn off the buzzer
                delay(2000);
                return; // Exit the countdown if the system is disarmed
            }
            else {
                lcd.clear();
            }
        }
    }
}

```

```

        lcd.setCursor(0, 0);
        lcd.print("Wrong Password!");
        delay(1000);
        inputPassword = ""; // Clear the input password after a wrong attempt
    }
}

// Display the entered password on the first line
lcd.setCursor(0, 0);
lcd.print("Enter: ");
lcd.print(inputPassword);
}

// Check if the system is disarmed during the countdown
if (!alarmArmed) { // If disarmed, stop the countdown
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Alarm Disarmed");
    countdownActive = false;
    delay(2000);
    return;
}
}

// If the countdown finishes without disarming, trigger the alarm
alarmTriggered = true;
digitalWrite(BUZZER_PIN, HIGH); // Turn on the buzzer after the countdown
Blynk.logEvent("gen", "Warning: No password inserted. Door Open!");
}

// Function to measure distance with ultrasonic sensor
float getDistance() {
    digitalWrite(TRIG_PIN, LOW);
    delayMicroseconds(2);
    digitalWrite(TRIG_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIG_PIN, LOW);
    long duration = pulseIn(ECHO_PIN, HIGH);
    float distance = duration * 0.034 / 2;
    return distance;
}

void checkFlameSensor() {
    // If the alarm is Armed and the flame sensor see fire go HIGH
    if (alarmArmed && digitalRead(FLAME_SENSOR_PIN) == HIGH) {
        triggerAlarm("FIRE!!!"); // enable alarm with message "FIRE!!!"
        Blynk.virtualWrite(V33, FLAME_SENSOR_PIN);
    }
}

```

```
Blynk.logEvent("fire_", "Warning: FIRE IN HOUSE!");
} else { // If the fire is over
  // Reset
  Blynk.virtualWrite(V33, 0); // turn of LED on blynk
  char key = keypad.getKey();
  if (key) {
    if (key >= '0' && key <= '9') {
      inputPassword += key;
      lcd.setCursor(0, 1);
      lcd.clear();
      lcd.print("Enter: ");
      lcd.print(inputPassword);
    }
  }
}
}
```