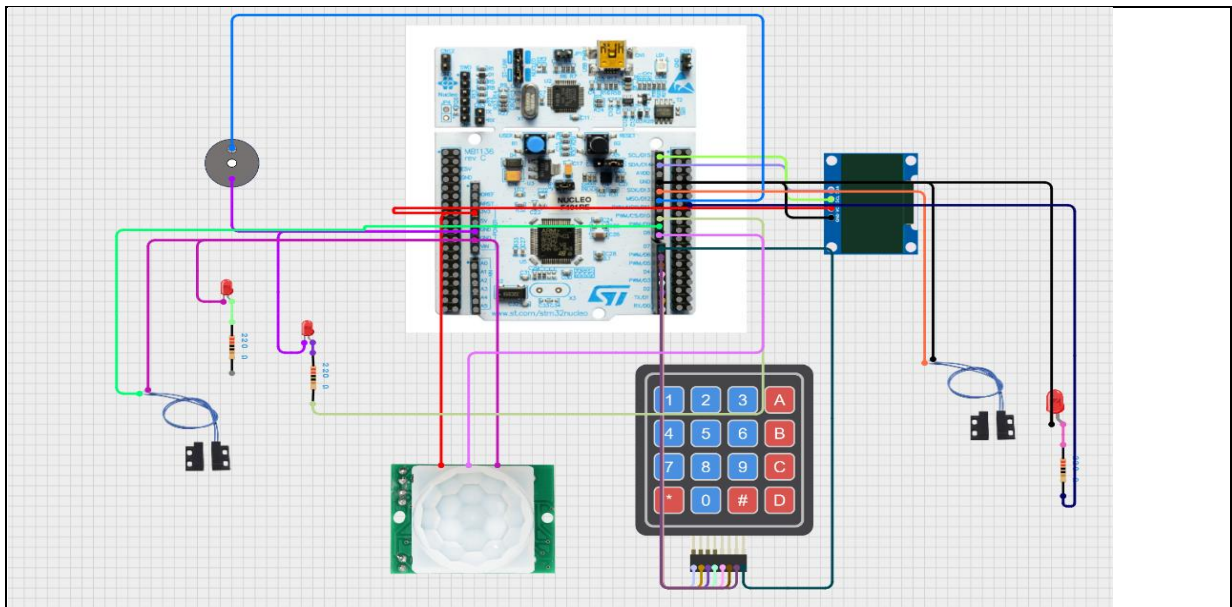


ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

«Σχεδίαση και Ανάπτυξη Πρωτότυπου Πολυζωνικού
Συστήματος Συναγερμού με Μικροελεγκτή STM32 και
Καταγραφή Συμβάντων»



Του φοιτητή
Παπαδόπουλου Χρήστου
Αρ. Μητρώου: 518106

Επιβλέπων
Δημήτριος Κ. Παπακόστας
Καθηγητής

Ημερομηνία ...05/09/2026.....

Σχεδίαση και Ανάπτυξη Πρωτότυπου Πολυζωνικού Συστήματος Συναγερμού με Μικροελεγκτή STM32 και Καταγραφή Συμβάντων

Κωδικός Π.Ε. 26229

Ονοματεπώνυμο φοιτητή Χρήστος Παπαδόπουλος

Ονοματεπώνυμο εισηγητή Δημήτριος Κ. Παπακώστας

Ημερομηνία ανάληψης Π.Ε. 29-03-2026

Ημερομηνία περάτωσης Π.Ε. ...

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 διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Χρήστου Παπαδόπουλου 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 διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητα και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

«Στην οικογένειά μου»

Πρόλογος

Η παρούσα πτυχιακή εργασία έχει ως στόχο τη σχεδίαση ενός πολυζωνικού συστήματος συναγερμού, το οποίο στηρίζεται στην αρχιτεκτονική των μικροελεγκτών STM32(ARM Cortex -M) δίνοντας στην εργασία την πολυπλοκότητα που χρειάζεται τόσο στο κομμάτι του hardware όσο και του software. Αρχικά όσον αφορά τα συστήματα ασφαλείας είναι ένας αρκετά ενδιαφέρον κλάδος καθώς συνδυάζουν άψογα πολλές λειτουργίες που λαμβάνουν χώρα την ίδια στιγμή. Δεύτερον ο μικροελεγκτής ήταν και αυτός ένας ακόμη λόγος επιλογής του θέματος διότι παρέχει αρκετές δυνατότητες και ευελιξία. Κατά τη διάρκεια εκπόνησης της εργασίας μου δόθηκε η ευκαιρία να ανακαλύψω εις βάθος το πως λειτουργεί ένα σύστημα ασφαλείας, η ταυτόχρονη εκτέλεση πολλών εργασιών και η επεξεργασία δεδομένων. Τέλος μπόρεσα να εμβαθύνω αρκετά στη συγγραφή κώδικα σε ένα πιο απαιτητικό αντικείμενο.

Περίληψη

Η παρούσα πτυχιακή εργασία επικεντρώνεται στη δημιουργία ενός συστήματος συναγερμού το οποίο βασίζεται στον μικροελεγκτή STM32. Στο συγκεκριμένο συναγερμό υπάρχει δυνατότητα όπλισης της κάθε ζώνης ξεχωριστά, σε ζευγάρια και φυσικά τη συνολική όπλιση του συστήματος καταγράφοντας το κάθε συμβάν σε πραγματικό χρόνο. Για την υλοποίηση του συναγερμού χρησιμοποιήθηκε το STM32 Nucleo F401RE Development Board ως κεντρική μονάδα επεξεργασίας το οποίο επικοινωνεί με τον αισθητήρα, τις μαγνητικές επαφές, την οθόνη, το πληκτρολόγιο, τον βομβητή και τα leds. Ως αποτέλεσμα συνοπτικά σε κάθε αλλαγή κατάστασης είτε στον αισθητήρα είτε στις μαγνητικές επαφές ο μικροελεγκτής τις ανιχνεύει και εμφανίζει τις ανάλογες ειδοποιήσεις στην οθόνη ενώ παράλληλα παρατηρείται οπτική και ακουστική ειδοποίηση κατευθείαν. Επιπλέον η όπλιση καθώς και η αφόπλιση γίνεται μέσω του πληκτρολογίου, ενώ υπάρχει και η δυνατότητα αλλαγής κωδικού και αποθήκευσής του νέου κωδικού. Τέλος, το σύστημα κατασκευάστηκε και δοκιμάστηκε σε δοκιμαστική πλακέτα (breadboard). Τα σενάρια ελέγχου έδειξαν ότι ανταποκρίνεται σωστά σε πραγματικό χρόνο, διατηρεί με ακρίβεια τη σειρά των παραβιάσεων και επιτρέπει την ασφαλή αλλαγή κωδικού από τον χρήστη, επιβεβαιώνοντας την αξιοπιστία του συναγερμού.

«Design and Development of a Prototype Multi-Zone Alarm System with an STM32 Microcontroller and Event Logging»

CHRISTOS PAPADOPOULOS

Abstract

This thesis focuses on the development of an alarm system based on the STM32 microcontroller. The alarm allows each zone to be armed separately, in pairs, and of course the whole system to be armed, with each event recorded in real time. To implement the alarm, the STM32 Nucleo F401RE Development Board was used as the main processing unit, communicating with the sensor, the magnetic contacts, the display, the keypad, the buzzer, and the LEDs. In summary, whenever there is any change in state in either the sensor or the magnetic contacts, the microcontroller detects it and displays the corresponding notifications on the screen, while visual and audible alerts are triggered immediately. In addition, arming and disarming are carried out via the keypad, and there is also the option to change the code and store the new one. Finally, the system was built and tested on a breadboard. The test scenarios showed that it responds correctly in real time, accurately preserves the sequence of breaches, and allows the user to change the code securely, confirming the alarm's reliability.

Ευχαριστίες

Ευχαριστώ την οικογένειά μου αλλά και τον επιβλέποντα καθηγητή μου.

Περιεχόμενα

Πρόλογος.....	vi
Περίληψη.....	vii
Abstract	viii
Ευχαριστίες	ix
Περιεχόμενα	x
Κατάλογος Σχημάτων	xii
Κατάλογος Πινάκων.....	xii
Κεφάλαιο 1ο: Εισαγωγή	1
1.1 Αντικείμενο και Σκοπός της Εργασίας.....	1
1.2 Κίνητρο	1
1.3 Μεθοδολογία Υλοποίησης	2
1.4 Δομή της Εργασίας.....	3
Κεφάλαιο 2ο: Θεωρητικό Υπόβαθρο.....	4
2.1 Συστήματα Συναγερμού και Πολυζωνική Προστασία	4
2.2 Η Οικογένεια Μικροελεγκτών STM32 και η Πλακέτα Nucleo	4
2.3 Αισθητήρες και Τεχνολογίες Ανίχνευσης	5
2.4 Το Πρωτόκολλο Επικοινωνίας I2C (Inter-Integrated Circuit)	5
Κεφάλαιο 3ο: Σχεδίαση και Επιλογή Υλικού (Hardware).....	6
3.1 Αρχιτεκτονική Συστήματος.....	6
3.2 Σχεδίαση των Ζωνών Εισόδου	6
3.3 Διεπαφή Χρήστη(User interface)	9
3.4 Μονάδα Καταγραφής Συμβάντων.....	12
3.5 Σύστημα Τροφοδοσίας	12
Κεφάλαιο 4ο: Ανάπτυξη Λογισμικού (Software)	14
4.1 Περιβάλλον Ανάπτυξης (IDE)	14
4.2 Λογική Συστήματος(State Machine)	14
4.3 Διαχείριση Περιφερειακών (Drivers & Interrupts)	16
4.4 Διαχείριση Κωδικών και Ασφάλεια	16
4.5 Υλοποίηση Καταγραφής(Logging Algorith).....	17
Κεφάλαιο 5ο: Υλοποίηση, Δοκιμές και Αποτελέσματα	18
5.1 Κατασκευή Πρωτοτύπου σε Breadboard	18
5.2 Σενάρια Ελέγχου(Test Cases)	24

Κεφάλαιο 6ο:	Συμπεράσματα και Μελλοντικές Επεκτάσεις.....	33
6.1	Σύνοψη Αποτελεσμάτων	33
6.2	Περιορισμοί της Υλοποίησης.....	33
6.3	Προτάσεις για Μελλοντική Βελτίωση.....	34
ΒΙΒΛΙΟΓΡΑΦΙΑ.....		35
ΠΑΡΑΡΤΗΜΑ Α: ΚΑΤΑΛΟΓΟΣ ΥΛΙΚΩΝ.....		37
ΠΑΡΑΡΤΗΜΑ Β : ΣΧΗΜΑΤΙΚΟ ΑΠΟ KICAD.....		37
ΠΑΡΑΡΤΗΜΑ Γ : ΚΩΔΙΚΑΣ.....		37
ΠΑΡΑΡΤΗΜΑ Δ : STM32CUBEMX		52

Κατάλογος Σχημάτων

Εικόνα 1 Αναπτυξιακή Πλακέτα Nucleo-F401RE.....	2
Εικόνα 2: Αισθητήρας Κίνησης HC-SR501.....	7
Εικόνα 3: Κύκλωμα του αισθητήρα κίνησης HC-SR501.....	8
Εικόνα 4: Καταστάσεις της Μαγνητικής Επαφής.....	9
Εικόνα 5: Οθόνη LCD 20x4.....	10
Εικόνα 6: Πληκτρολόγιο 4x4 matrix membrane.....	11
Εικόνα 7: Διάγραμμα ροής.....	15
Εικόνα 8: Έλεγχος για 2 ^η παραβίαση.....	17
Εικόνα 9: Πίσω Μέρος Οθόνης. Πλακέτα Διεπαφής I2C.....	18
Εικόνα 10: Μπροστά Μέρος Οθόνης.....	19
Εικόνα 11: Μαγνητικές Επαφές (Πόρτα & Παράθυρο).....	19
Εικόνα 12: Πληκτρολόγιο.....	20
Εικόνα 13: Εσωτερική Συνδεσμολογία Πληκτρολογίου.....	21
Εικόνα 14: Αισθητήρας Κίνησης.....	21
Εικόνα 15: Led Αισθητήρα Κίνησης /Led Ειδοποίησης Καταστάσεων.....	22
Εικόνα 16: Led Μαγνητικής Επαφής Παραθύρου.....	22
Εικόνα 17: Led Μαγνητικής Επαφής Πόρτας.....	23
Εικόνα 18: Βομβητής.....	23
Εικόνα 19: Μήνυμα της οθόνης στο delay 15s.....	25
Εικόνα 20: Μήνυμα της οθόνης μόλις οπλιστεί πλήρως.....	25
Εικόνα 21: Μήνυμα οθόνης μετά την εισαγωγή λάθος κωδικού πρώτη φορά.....	26
Εικόνα 22: Μήνυμα οθόνης μετά την εισαγωγή λάθος κωδικού δεύτερη φορά.....	26
Εικόνα 23: Μήνυμα οθόνης μετά την εισαγωγή λάθος κωδικού τρίτη φορά.....	27
Εικόνα 24: Επιλογή Ζωνών.....	28
Εικόνα 25: Ανίχνευση Κίνησης.....	28
Εικόνα 26: Παραβίαση Πόρτας.....	29
Εικόνα 27: Ανίχνευση Κίνησης – Παραβίαση Πόρτας.....	29
Εικόνα 28: Παραβίαση Πόρτας – Ανίχνευση Κίνησης.....	30
Εικόνα 29: Μήνυμα της οθόνης στο delay 15s για πλήρη όπλιση.....	31
Εικόνα 30: Ανίχνευση Κίνησης και delay 15s.....	31
Εικόνα 31: Μήνυμα της οθόνης στο delay 15s για όπλιση ζωνών.....	32

Κατάλογος Πινάκων

Πίνακας 1: Συνδεσμολογία με NUCLEO-F401.....	8
Πίνακας 2: Συνδεσμολογία με NUCLEO-F401.....	10
Πίνακας 3: Συνδεσμολογία με NUCLEO-F401.....	11
Πίνακας 4: Συνδεσμολογία με NUCLEO-F401.....	12
Πίνακας 5: Συνδεσμολογία με NUCLEO-F401.....	12
Πίνακας 6: Συνδεσμολογία Οθόνης.....	18
Πίνακας 7: Συνδεσμολογία Μαγνητικών Επαφών.....	19
Πίνακας 8: Συνδεσμολογία Πληκτρολογίου.....	20

Πίνακας 9: Συνδεσμολογία του Αισθητήρα Κίνησης	21
Πίνακας 10: Κατάλογος-Κοστολόγηση Υλικών	37

Κεφάλαιο 1ο: Εισαγωγή

1.1 Αντικείμενο και Σκοπός της Εργασίας

Στη σημερινή κοινωνία, η προστασία του ανθρώπου αλλά και τα κτίρια στα οποία υπάρχει η ανθρώπινη παρουσία όπως είναι οι εργασιακοί χώροι, τα σπίτια κ.α έχουν οδηγήσει στη ραγδαία ανάπτυξη και τοποθέτηση των συστημάτων ασφαλείας. Τα ψηφιακά συστήματα παρέχουν στους χρήστες δυνατότητα παρακολούθησης, καταγραφή συμβάντων σε πραγματικό χρόνο καθώς και υψηλή αξιοπιστία.

Η συγκεκριμένη εργασία έχει ως στόχο τον σχεδιασμό, τη κατασκευή και την καταγραφή γεγονότων σε πραγματικό χρόνο ενός πολυζωνικού συστήματος συναγερμού. Το σύστημα διαθέτει:

- Πολλαπλές Ζώνες: περιλαμβάνουν μαγνητικές επαφές και αισθητήρα κίνησης
- Οπτική και Ακουστική ένδειξη: βομβητής και leds
- Διεπαφή Χρήστη: οθόνη lcd και πληκτρολόγιο
- Επικοινωνία και Τροφοδοσία: USB θύρα

Ο κύριος σκοπός της εργασίας είναι η δημιουργία ενός ψηφιακού συστήματος ασφαλείας το οποίο θα καλύπτει τις ανάγκες του χρήστη ενώ παράλληλα, είναι εύκολο και γρήγορο όσον αφορά τη λειτουργία του [15], [16].

1.2 Κίνητρο

Οι λόγοι επιλογής του STM32 και συγκεκριμένα NUCLEO-F401RE όπου είναι αναπτυξιακή πλακέτα έγινε διότι συγκριτικά με άλλες παλαιότερες ή απλούστερες παρέχει τα εξής:

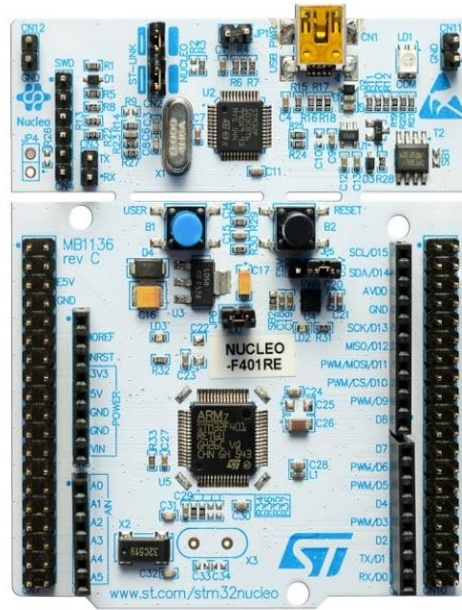
Επεξεργαστής: Ο πυρήνας ARM Cortex-M4 προσφέρει μια αρκετά υψηλή συχνότητα συγκεκριμένα 84MHz αυξάνοντας την ταχύτητα, και κατ' επέκταση την άμεση απόκριση του συστήματος με τα περιφερειακά που συνδέονται πάνω στη πλακέτα.

Αρχιτεκτονική ARM Cortex-M4 (32-bit): Η αρχιτεκτονική των 32 bit συγκριτικά με πυρήνες στα 8 bit μπορεί να υποστηρίξει μεγαλύτερο όγκο εντολών και καλύτερη διαχείριση δεδομένων κάτι που κάνει αυτή την αρχιτεκτονική ιδανική για ενσωματωμένα συστήματα [1].

Ενσωματωμένη Αποσφαλμάτωση (Hardware Debugging): Το NUCLEO-F401RE διαθέτει τον προγραμματιστή ST-LINK που επιτρέπει την παρακολούθηση του κώδικα σειρά προς σειρά για την αντιμετώπιση λαθών πολύ πιο εύκολα.

Κόστος-Απόδοση: Το κόστος της συγκεκριμένης πλακέτας με βάση τα χαρακτηριστικά δηλαδή μεγάλη μνήμη, συχνότητα επεξεργαστή είναι μια αρκετά οικονομική επιλογή σε σχέση με άλλα μοντέλα (π.χ. Arduino mega 2560) [11].

Εν κατακλείδι η επιλογή της αναπτυξιακής πλακέτας NUCLEO STM32F4 STM32F401RE έγινε με στόχο μια πιο επαγγελματική προσέγγιση και την κάλυψη των αναγκών του συστήματος [10].



Εικόνα 1 Αναπτυξιακή Πλακέτα Nucleo-F401RE

<https://www.st.com/en/evaluation-tools/nucleo-f401re.html>

1.3 Μεθοδολογία Υλοποίησης

Τα βήματα που ακολουθήθηκαν μέχρι το τελικό αποτέλεσμα είναι τα παρακάτω:

Θεωρητική Σχεδίαση: Πρωταρχικό στάδιο ήταν η καταγραφή της κεντρικής ιδέας του συναγερμού και των υλικών που χρησιμοποιήθηκαν, ενώ στη συνέχεια ήταν η σχεδίαση για αρχή στο χαρτί όλου του κυκλώματος.

Υλοποίηση σε Breadboard: Μετά τη συλλογή όλων των υλικών τοποθετήθηκαν στο breadboard με βάση την αρχική τοποθέτηση που είχε γίνει στο χαρτί με μικρές τροποποιήσεις όσον αφορά τη θέση τους στο κύκλωμα. Ωστόσο τα υλικά τοποθετήθηκαν αφού ελέγχθηκε η σωστή λειτουργία τους με τη χρήση τροφοδοτικού και πολυμέτρου.

Ανάπτυξη Κώδικα: Ακολούθησε η συγγραφή του κώδικα με στόχο την οριστικοποίηση των λειτουργιών όλων των περιφερειακών. Αυτό περιλαμβάνει τα μηνύματα που εμφανίζονται στην οθόνη μέσω του πρωτοκόλλου I2C, τις λειτουργίες των μαγνητικών επαφών, του αισθητήρα κίνησης του βομβητή, των led και του πληκτρολογίου.

Δοκιμές: Τέλος αφού έγινε η υλοποίηση του hardware και του software έγιναν αρκετά πειράματα έτσι ώστε να εξαλειφθούν τα σφάλματα αλλά και να προστεθούν επιπλέον λειτουργίες ή να βελτιωθούν οι ήδη υπάρχουσες.

1.4 Δομή της Εργασίας

Κεφάλαιο 1: Εισαγωγή στο αντικείμενο της πτυχιακής, στόχος της κατασκευής και λόγοι επιλογής του NUCLEO STM32F4 STM32F401RE.

Κεφάλαιο 2: Στο δεύτερο μέρος γίνεται παράθεση των απαραίτητων εννοιών για την κατανόηση του συστήματος. Συγκεκριμένα γίνεται μια αναφορά στα συστήματα συναγερμού και η έννοια της ζώνης σε ένα συναγερμό. Ακόμη αναλύονται τα χαρακτηριστικά του μικροελεγκτή και των περιφερειακών του κυκλώματος καθώς και τα πρωτόκολλα επικοινωνίας που χρησιμοποιήθηκαν.

Κεφάλαιο 3: Το τρίτο κεφάλαιο αναφέρεται στην αρχιτεκτονική του συστήματος με block diagram και πιο αναλυτικά η συνδεσμολογία των περιφερειακών, η διεπαφή με τον χρήστη η καταγραφή συμβάντων και τέλος το σύστημα τροφοδοσίας. Εν ολίγοις αναλύεται η σχεδίαση και επιλογή υλικού.

Κεφάλαιο 4: Με το τέταρτο κεφάλαιο αναλύεται η “καρδιά” της λογικής του συστήματος δηλαδή η ανάπτυξη λογισμικού. Πιο συγκεκριμένα γίνεται αναφορά στο περιβάλλον ανάπτυξης όπως και στη λογική λειτουργίας του συστήματος. Επιπλέον παρατίθεται η διαχείριση των περιφερειακών και των κωδικών του συναγερμού, αλλά και την υλοποίηση καταγραφής.

Κεφάλαιο 5: Στο συγκεκριμένο κεφάλαιο υπάρχουν φωτογραφίες και επεξήγηση του κυκλώματος σε συνδυασμό με όλα τα σενάρια ελέγχου και τέλος η αξιολόγηση καταγραφής.

Κεφάλαιο 6: Στο τελευταίο κεφάλαιο αναλύονται τα συμπεράσματα αλλά προτείνονται μελλοντικές επεκτάσεις του πρότυπου αυτού συναγερμού.

Κεφάλαιο 2ο: Θεωρητικό Υπόβαθρο

2.1 Συστήματα Συναγερμού και Πολυζωνική Προστασία

Η έννοια της “ζώνης” σε ένα ψηφιακό σύστημα αποτελεί μια περιοχή ή μια ομάδα από αισθητήρων (π.χ. μαγνητικές επαφές, αισθητήρες κίνησης), και διαχειρίζονται από τον μικροελεγκτή. Η χρήση πολλαπλών ζωνών δίνει τη δυνατότητα στη Κεντρική Μονάδα να ανιχνεύει την παραβίαση αλλά και την ακριβή τοποθεσία της. Γενικά υπάρχουν τρεις βασικές κατηγορίες ζωνών που εφαρμόζονται για χρήση σε κτίρια ή σε οικίες αυτές είναι:

- Άμεσης Ενεργοποίησης: Η συγκεκριμένη κατηγορία επι της ουσίας ενεργοποιείται κατευθείαν μόλις ανιχνευθεί κάποια παραβίαση όταν το σύστημα είναι οπλισμένο και ενεργοποιείται είτε κάποια σειρήνα είτε στέλνει σήμα σε αρμόδιες υπηρεσίες (π.χ. Κέντρο Άμεσης Δράσης)
- Με Χρονοκαθυστέρηση: Στη περίπτωση αυτή όταν ενεργοποιηθεί κάποια από τις περιφερειακές συσκευές τότε υπάρχει μια μικρή καθυστέρηση κάποια δευτερόλεπτα πριν δοθούν οι προηγούμενες εντολές έτσι ώστε ο χρήστης να μπορέσει να αποπλύνει τον συναγερμό εγκαίρως.
- 24ωρη Ζώνη: Τέλος σε αυτή τη κατηγορία η Κεντρική Μονάδα Επεξεργασίας ελέγχει συνεχώς τη ζώνη είτε είναι οπλισμένο το σύστημα είτε όχι. Σε αυτού του τύπου ζώνες οι αισθητήρες είναι συνήθως υψηλής σημασίας (π.χ. αισθητήρας καπνού και φωτιάς).

2.2 Η Οικογένεια Μικροελεγκτών STM32 και η Πλακέτα Nucleo

Στον σύγχρονο ηλεκτρονικό σχεδιασμό, η σειρά STM32 της STMicroelectronics κυριαρχεί. Ο βασικός λόγος εντοπίζεται στους πυρήνες ARM Cortex-M. Επιτυγχάνουν υψηλή υπολογιστική ταχύτητα με εξαιρετικά χαμηλή κατανάλωση ρεύματος, συγκεντρώνοντας πληθώρα εσωτερικών περιφερειακών μονάδων σε ένα ενιαίο chip.

Για την υλοποίηση του συναγερμού επιλέχθηκε η πλακέτα STM32 Nucleo-64 (NUCLEO-F401RE), η οποία ενσωματώνει τον μικροελεγκτή STM32F401RET6 [7]. Ο συγκεκριμένος επεξεργαστής ARM Cortex-M4 απαντά πλήρως στις αυστηρές απαιτήσεις συνεχούς επιτήρησης σε πραγματικό χρόνο, προσφέροντας τα εξής χαρακτηριστικά:

Αρχιτεκτονική 32-bit: Ξεφεύγοντας από τα όρια των παλαιότερων δομών 8-bit, ο Cortex-M4 επεξεργάζεται δεδομένα 32 bit ανά κύκλο ρολογιού. Με συχνότητα που φτάνει τα 84 MHz, η εκτέλεση εντολών γίνεται ταχύτατα [14]. Υπάρχει διαθέσιμη υποστήριξη πράξεων κινητής υποδιαστολής (FPU). Ωστόσο, η ουσιαστική διαφορά στο σύστημα ασφαλείας προκύπτει από την άμεση αξιολόγηση των λογικών συνθηκών.

Ελεγκτής Διακοπών NVIC (Nested Vectored Interrupt Controller): Το υποσύστημα αυτό καθορίζει την άμεση συμπεριφορά του πίνακα. Ελαχιστοποιεί την καθυστέρηση απόκρισης σε εξωτερικά συμβάντα. Εάν παραβιαστεί μια επαφή, η ροή εκτέλεσης αλλάζει επιτόπου. Ο επεξεργαστής σταματά άμεσα την αναμονή στο μενού και εκτελεί τη ρουτίνα συναγερμού.

Χωρητικότητα Μνήμης: Το εξάρτημα διαθέτει 512 KB Flash για την αποθήκευση του κώδικα και 96 KB SRAM για τις μεταβλητές εκτέλεσης [9]. Ο διαθέσιμος χώρος επαρκεί άνετα. Επιτρέπει την απρόσκοπτη λειτουργία της μηχανής καταστάσεων, αφήνοντας περιθώριο για μελλοντική καταγραφή εκτενούς ιστορικού συμβάντων.

Περιφερειακά και Ανοχή Τάσεων: Το ολοκληρωμένο περιλαμβάνει αυτόνομους ελεγκτές για διαύλους I2C, SPI και USART. Το κρίσιμότερο σχεδιαστικό πλεονέκτημα αφορά τους ακροδέκτες 5V Tolerant.

Ο πυρήνας τροφοδοτείται με 3.3V. Παρόλα αυτά, οι είσοδοι δέχονται σήματα 5V χωρίς κίνδυνο βλάβης. Αυτό επιτρέπει τη σύνδεση των εμπορικών αισθητήρων απευθείας στον μικροελεγκτή, χωρίς να απαιτούνται εξωτερικά κυκλώματα μετατροπής επιπέδων τάσης (level shifters).

2.3 Αισθητήρες και Τεχνολογίες Ανίχνευσης

Η αξιοπιστία ενός συστήματος ασφαλείας εξαρτάται άμεσα από την ποιότητα των αισθητήρων του. Αποτελούν τα αισθητήρια όργανα της κεντρικής μονάδας. Μέσω αυτών, ο μικροελεγκτής αντιλαμβάνεται τις μεταβολές του φυσικού περιβάλλοντος αναλύοντας συνεχώς τα δεδομένα προκειμένου να εντοπίσει εγκαίρως οποιαδήποτε απόπειρα παραβίασης. Στη συγκεκριμένη υλοποίηση αξιοποιήθηκαν δύο διαφορετικές τεχνολογίες ανίχνευσης. Η καθεμία εξυπηρετεί έναν εντελώς ξεχωριστό σκοπό.

Μαγνητικές Επαφές (Reed Switches): Η περιμετρική προστασία βασίζεται κυρίως στις μαγνητικές επαφές. Λειτουργούν με εξαιρετική απλότητα. Αποτελούνται από έναν μόνιμο μαγνήτη και έναν διακόπτη κλειστού τύπου τοποθετημένο μέσα σε μια μικρή γυάλινη αμπούλα. Όσο η πόρτα παραμένει κλειστή, το ισχυρό μαγνητικό πεδίο κρατά τα μεταλλικά ελάσματα σταθερά ενωμένα. Σε περίπτωση διάρρηξης τα δεδομένα αλλάζουν. Η βίαιη απομάκρυνση του μαγνήτη προκαλεί την ακαριαία διακοπή του κυκλώματος. Αυτή η στιγμιαία μηχανική μεταβολή μεταφράζεται απευθείας σε ψηφιακό σήμα το οποίο διαβάζεται από τον επεξεργαστή. Πρόκειται για μια εξόχως αξιόπιστη λύση. Δεν απαιτεί καν δική της τροφοδοσία.

Ανιχνευτές Κίνησης (PIR): Για την ογκομετρική κάλυψη των εσωτερικών χώρων επιστρατεύονται οι παθητικοί υπέρυθροι αισθητήρες. Η αρχή λειτουργίας τους διαφέρει ριζικά. Δεν εκπέμπουν ακτινοβολία. Αντιθέτως, ανιχνεύουν την υπέρυθρη ενέργεια που εκπέμπει φυσιολογικά το ανθρώπινο σώμα με τη μορφή θερμότητας. Καθώς ένας εισβολέας διασχίζει το πεδίο κάλυψης, ο αισθητήρας καταγράφει τη ραγδαία θερμική μεταβολή σε σχέση με το σταθερό υπόβαθρο του δωματίου. Οι σύγχρονοι ανιχνευτές PIR ενσωματώνουν συνήθως ειδικούς φακούς Fresnel. Αυτοί εστιάζουν την θερμική ακτινοβολία στο πυροηλεκτρικό στοιχείο, μεγιστοποιώντας την εμβέλεια και περιορίζοντας δραστηκά τους ενοχλητικούς ψευδείς συναγερμούς που προκαλούνται συχνά από μικρά κατοικίδια. Η απόκριση είναι ταχύτατη.

2.4 Το Πρωτόκολλο Επικοινωνίας I2C (Inter-Integrated Circuit)

Η ψηφιακή διασύνδεση του μικροελεγκτή με τα περιφερειακά εξαρτήματα αποτελεί τον κορμό κάθε ενσωματωμένου συστήματος. Χωρίς αξιόπιστη μεταφορά δεδομένων το σύστημα καταρρέει. Για τις ανάγκες της συγκεκριμένης κατασκευής κρίθηκε απαραίτητη η χρήση αποκλειστικά του πρωτοκόλλου I2C. Ήταν μια καθαρά πρακτική απόφαση. Έπρεπε να οδηγηθεί η οθόνη ενδείξεων. Πρόκειται για έναν σειριακό δίαυλο σύγχρονης επικοινωνίας ο οποίος αναπτύχθηκε αρχικά από τη Philips και απαιτεί μόλις δύο καλώδια για την πλήρη λειτουργία του, γεγονός που τον καθιστά εξαιρετικά δημοφιλή στους σχεδιαστές ηλεκτρονικών. Αποτελείται από μία γραμμή χρονισμού και μία δεύτερη γραμμή δεδομένων. Η αρχιτεκτονική του βασίζεται στο μοντέλο κυρίου και εξαρτημένου κόμβου. Ο μικροελεγκτής STM32 αναλαμβάνει τον ρόλο του απόλυτου συντονιστή. Υπαγορεύει τον ρυθμό μετάδοσης. Κάθε συνδεδεμένη συσκευή στο δίκτυο διαθέτει μια μοναδική διεύθυνση υλικού αναγνωρίζοντας αμέσως τα πακέτα πληροφοριών που προορίζονται αποκλειστικά για την ίδια, αγνοώντας πλήρως την υπόλοιπη κυκλοφορία του διαύλου. Αυτή η προσέγγιση εξοικονομεί πάρα πολλούς πολύτιμους ακροδέκτες από την αναπτυξιακή πλακέτα Nucleo. Οι συγκεκριμένες θύρες απελευθερώνονται. Μπορούν έτσι να διατεθούν αποκλειστικά για τη σύνδεση των πολλαπλών αισθητήρων του συναγερμού ή τη σάρωση του πληκτρολογίου χειρισμού. Μια απλή σχεδιαστική λύση που μεγιστοποιεί την αποδοτικότητα [17].

Κεφάλαιο 3ο: Σχεδίαση και Επιλογή Υλικού (Hardware)

3.1 Αρχιτεκτονική Συστήματος

Η αποτύπωση της αρχιτεκτονικής συνιστά το πρώτο βήμα κατανόησης. Το διάγραμμα βαθμίδων αποσαφηνίζει τον τρόπο με τον οποίο τα επιμέρους εξαρτήματα της κατασκευής αλληλοεπιδρούν μεταξύ τους γύρω από τον κεντρικό πυρήνα ελέγχου, διαμορφώνοντας ένα ενιαίο, λειτουργικό σύνολο. Κάθε μπλοκ αντιπροσωπεύει μια διακριτή φυσική μονάδα. Βάσει των υλικών που χρησιμοποιήθηκαν αποκλειστικά στη δική μας υλοποίηση, το σύστημα χωρίζεται σε πέντε βασικούς άξονες.

Κεντρική Μονάδα Επεξεργασίας :Στο κέντρο της τοπολογίας δεσπόζει ο μικροελεγκτής STM32F401RET6. Αναλαμβάνει τον πλήρη συντονισμό. Δέχεται τα σήματα εισόδου, επεξεργάζεται τα δεδομένα βάσει της προγραμματισμένης λογικής και οδηγεί τις κατάλληλες εξόδους.

Υποσύστημα Ανίχνευσης :Το συγκεκριμένο τμήμα τροφοδοτεί διαρκώς τον επεξεργαστή με πληροφορίες για την κατάσταση του χώρου. Απαρτίζεται από τις μαγνητικές επαφές για την περιμετρική προστασία και τον αισθητήρα κίνησης PIR για τον ογκομετρικό έλεγχο. Αυτά τα περιφερειακά συνδέονται απευθείας σε ψηφιακές εισόδους της πλακέτας. Η παραμικρή διέγερσή τους προκαλεί ακαριαία αλλαγή δυναμικού.

Διεπαφή Χρήστη :Η αλληλεπίδραση με τον χειριστή πραγματοποιείται μέσω δύο βασικών στοιχείων: Το πληκτρολόγιο και την οθόνη ενδείξεων. Μέσω του πληκτρολογίου εισάγονται οι κωδικοί πρόσβασης για την όπλιση ή την αφόπλιση του συναγερμού. Η οθόνη υγρών κρυστάλλων επικοινωνεί μέσω του διαύλου I2C παρέχοντας οπτική ανατροφοδότηση σχετικά με την τρέχουσα κατάσταση των ζωνών.

Στάδιο Εξόδου :Η συγκεκριμένη βαθμίδα αφορά την ηχητική και οπτική ειδοποίηση. Ο βομβητής συνδέεται κατευθείαν σε ψηφιακή είσοδο και ενεργοποιείται μόλις το λογισμικό εντοπίσει παραβίαση, ενώ ταυτόχρονα το ίδιο σήμα είναι που οδηγεί στην ενεργοποίηση των led. Απαιτείται προσεκτική ηλεκτρική απομόνωση προκειμένου να προστατευθεί η ευαίσθητη λογική του μικροελεγκτή από ανεπιθύμητες επιστροφές ρεύματος κατά την ενεργοποίηση του ηχείου.

Διαχείριση Τροφοδοσίας: Η μονάδα ενέργειας αποτελεί τον πυρήνα της κατασκευής. Το κύκλωμα της πλακέτας Nucleo φροντίζει για τη σταθεροποίηση της τάσης, διανέμοντας τα απαραίτητα επίπεδα των 3.3V και 5V στα διάφορα υποσυστήματα. Εξασφαλίζεται έτσι η αρμονική ηλεκτρική συμβίωση εξαρτημάτων με διαφορετικές απαιτήσεις λειτουργίας.

3.2 Σχεδίαση των Ζωνών Εισόδου

Στο σύστημα των ζωνών εισόδου χρησιμοποιήθηκαν μαγνητικές επαφές και ένας αισθητήρας κίνησης, προσομοιάζοντας ένα σύστημα συναγερμού για οικιακή χρήση.

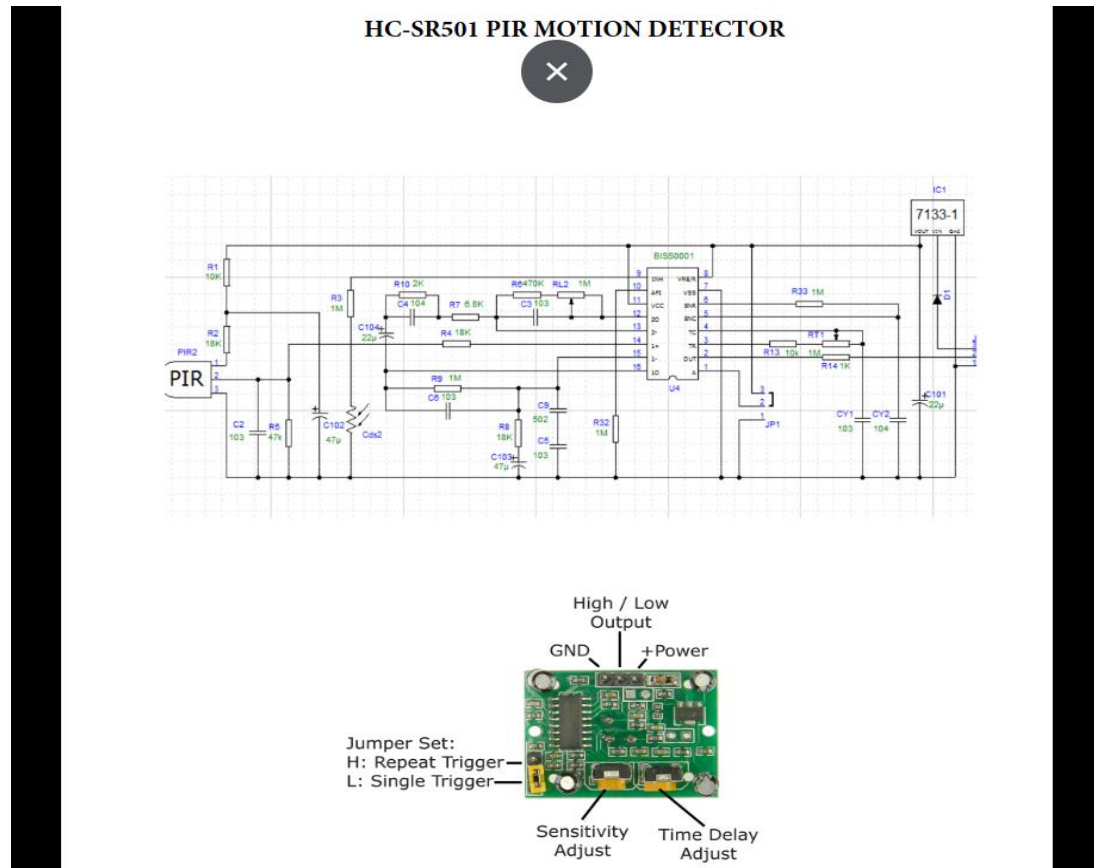
Αισθητήρας Κίνησης



Εικόνα 2: Αισθητήρας Κίνησης HC-SR501

<https://en.hwlibre.com/pir-sensor/>

Για την ανίχνευση κίνησης χρησιμοποιήθηκε ο αισθητήρας HC-SR501 που αποτελεί μια μονάδα παθητικού υπέρυθρου η οποία δεν εκπέμπει καμία ακτινοβολία. Αντιθέτως, αντιλαμβάνεται τη θερμική ενέργεια που εκλύεται φυσιολογικά από το ανθρώπινο σώμα, καταγράφοντας τις απότομες θερμοκρασιακές μεταβολές στο πεδίο εμβέλειάς της μέσω ενός εξαιρετικά ευαίσθητου πυροηλεκτρικού στοιχείου. Το λευκό ημισφαιρικό κάλυμμα διαδραματίζει σπουδαίο ρόλο. Δεν αποτελεί απλώς ένα πλαστικό περίβλημα προστασίας από τη σκόνη και λειτουργεί ως φακός Fresnel συγκεντρώνοντας την υπέρυθρη ακτινοβολία εστιάζοντάς την απευθείας στο κεντρικό αισθητήριο. Πάνω στην ίδια την πλακέτα του εξαρτήματος υπάρχουν δύο ενσωματωμένα ποτενσιόμετρα. Το πρώτο ρυθμίζει την ευαισθησία ελέγχοντας πρακτικά τη μέγιστη απόσταση εντοπισμού ενός εισβολέα. Το δεύτερο καθορίζει τον χρόνο καθυστέρησης, δηλαδή την ακριβή χρονική διάρκεια κατά την οποία η ψηφιακή έξοδος θα παραμείνει ενεργοποιημένη αμέσως μετά από μια επιτυχημένη ανίχνευση κίνησης [3].



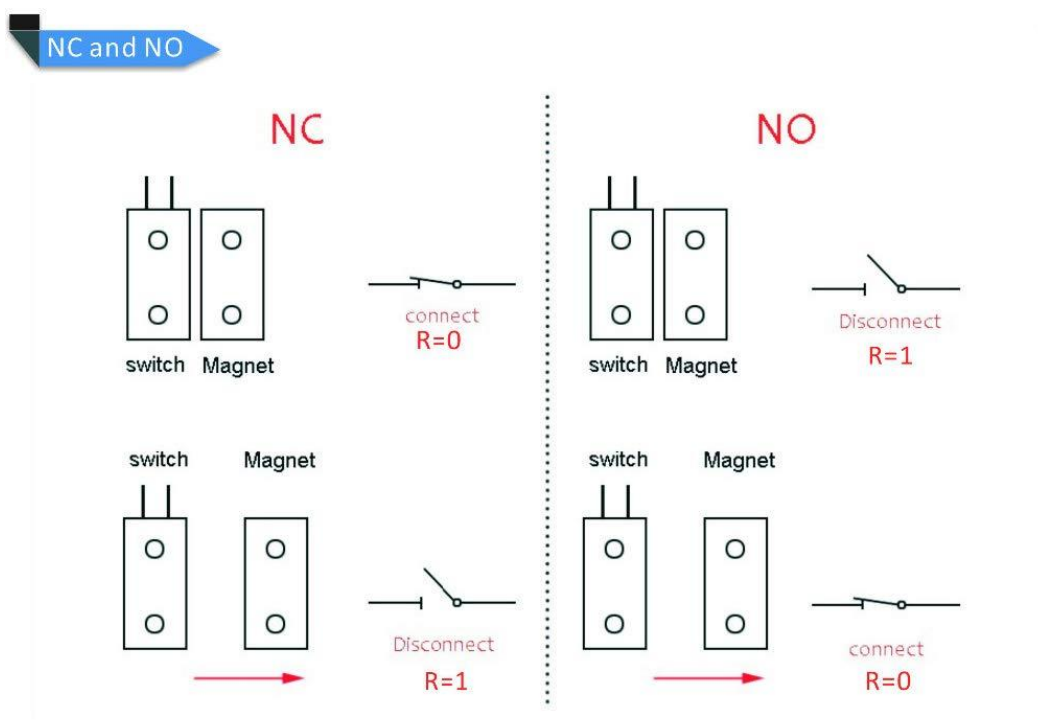
Εικόνα 3: Κύκλωμα του αισθητήρα κίνησης HC-SR501[4]

Πίνακας 1: Συνδεσμολογία με NUCLEO-F401

HC-SR501	NUCLEO -F401
POWER	+5V
OUTPUT	PC0
GND	GND

Μαγνητικές Επαφές

Οι μαγνητικές επαφές έχουν δύο καλώδια τα οποία συνδέονται το ένα στη γείωση και το άλλο σε ένα ψηφιακό pin κατευθειάν πάνω στη πλακέτα. Ο μικροελεγκτής διαβάζει λογικό "0" όταν είναι κλειστή η επαφή και λογικό "1" όταν ανοίξει.



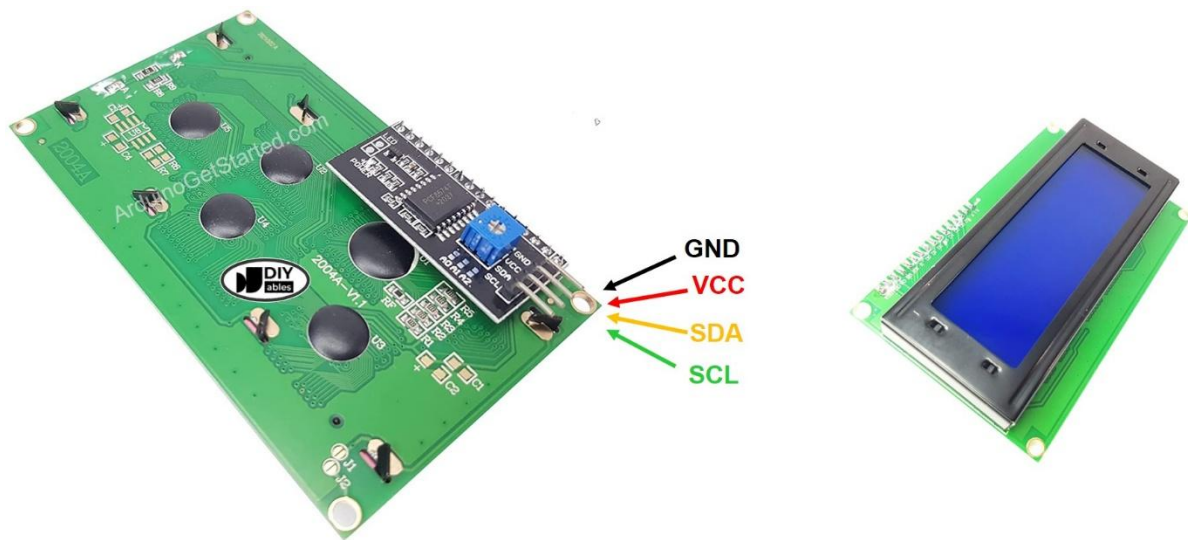
Εικόνα 4: Καταστάσεις της Μαγνητικής Επαφής

https://www.amazon.in/Magnetic-Normally-Security-Contact-Lamp%EF%BC%88White%EF%BC%89/dp/B086GYGCJ8/ref=pd_bxgy_thbs_d_sccl_1/524-5704938-9279301?pd_rd_w=7719b&content-id=amzn1.sym.142c0d31-3594-485b-b879-def9251f61c0&pf_rd_p=142c0d31-3594-485b-b879-def9251f61c0&pf_rd_r=PB0E953YF9TK9CH0NRAQ&pd_rd_wg=l0Y6t&pd_rd_r=146b70b4-901b-4e63-8f28-62e10edf59cf&pd_rd_i=B086GYGCJ8&psc=1

3.3 Διεπαφή Χρήστη (User interface)

Οθόνη LCD20x4 I2C

Η διεπαφή με τον χρήστη γίνεται μέσω μιας οθόνης LCD 20x4 όπου επιτρέπει την εμφάνιση 20 χαρακτήρων σε κάθε μια από τις 4 γραμμές. Επιπλέον η οθόνη διαθέτει στο πίσω μέρος μια μικρή πλακέτα προσαρμογής η οποία αξιοποιεί το πρωτόκολλο I2C. Η συγκεκριμένη επιλογή έγινε διότι δεν χρειάζεται να καταναλωθούν αμέτρητοι ψηφιακοί ακροδέκτες του μικροελεγκτή για την οδήγησή της, καθώς τώρα χρειάζονται μόλις τέσσερα καλώδια για πλήρη και αποτελεσματική σύνδεση [5],[13].



Εικόνα 5:Οθόνη LCD 20x4

<https://arduinogetstarted.com/tutorials/arduino-lcd-20x4>

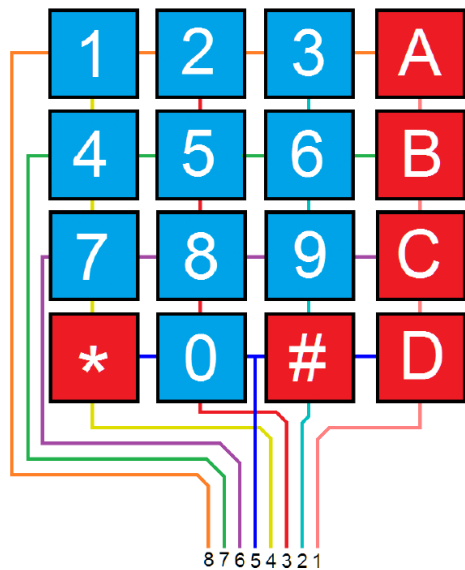
Πίνακας 2: Συνδεσμολογία με NUCLEO-F401

ΟΘΟΝΗ LCD 20x4	NUCLEO-F401
VCC	+5V
GND	GND
SDA	PB9
SCL	PB8

Πληκτρολόγιο

Για τον πλήρη έλεγχο του συναγερμού από τον χρήστη χρησιμοποιήθηκε ένα πληκτρολόγιο 4x4 matrix membrane το οποίο αποτελείται από 16 πλήκτρα (4 σειρές και 4 στήλες). Το πληκτρολόγιο έχει στο σύνολο του 8 ενώσεις οι οποίες είναι επι της ουσίας οι 4 σειρές και οι 4 στήλες αντίστοιχα και

συνδέονται κατευθείαν στα ψηφιακά pin του μικροελεκτή. Η λογική πίσω από το πληκτρολόγιο είναι ότι το κάθε πλήκτρο έχει δυο ενώσεις η μια είναι σε σειρά και η άλλη σε μια στήλη [6].



Εικόνα 6: Πληκτρολόγιο 4x4 matrix membrane

<https://www.sunrom.com/p/matrix-keypad-4x4-membrane-type-self-adhesive>

Πίνακας 3: Συνδεσμολογία με NUCLEO-F401

Πληκτρολόγιο 4x4	NUCLEO-F401
R1	PA4
R2	PA5
R3	PA6
R4	PA7
C1	PC4
C2	PC5
C3	PC6
C4	PC7

Βομβητής και Led

Το οπτικοακουστικό σκέλος ολοκληρώνει την τοπική αλληλεπίδραση. Τα led είναι τρία στο σύνολο ένα για κάθε μια ζώνη, έχουν διαφορετικά χρώματα και τοποθετήθηκαν στο κύκλωμα σε συνδυασμό με αντιστάσεις με τιμή 330Ω σε σειρά για να αποφευχθεί η καταστροφή τους παρέχοντας μια γρήγορη εποπτική εικόνα για τις τρέχουσες παραβιάσεις που μπορεί να συμβούν. Η ηχητική ειδοποίηση ανατέθηκε σε ένα τοπικό βομβητή. Το συγκεκριμένο εξάρτημα αναλαμβάνει την ειδοποίηση του χρήστη εκπέμποντας ένα συνεχές διαπεραστικό σήμα σε περίπτωση που καταγραφεί παραβίαση κάποιας ζώνης, ενώ η ακουστική απόκριση είναι άμεση.

Πίνακας 4: Συνδεσμολογία με NUCLEO-F401

Βομβητής	NUCLEO- F401
Positive	PC3
Negative	GND

Πίνακας 5: Συνδεσμολογία με NUCLEO-F401

Led	NUCLEO-F401
ANODE (PIR SENSOR/MAGNETIC D/MAGNETIC W)	PA1/PC1/PC2
CATHODE	GND

3.4 Μονάδα Καταγραφής Συμβάντων

Η διατήρηση ιστορικού και ρυθμίσεων μετατρέπει μια απλή κατασκευή σε ένα ολοκληρωμένο σύστημα ασφαλείας. Κάθε παραβίαση πρέπει να καταγράφεται. Το ζήτημα του χρονισμού επιλύεται μέσα από τον ίδιο τον μικροελεγκτή. Ο επεξεργαστής STM32F401RET6 ενσωματώνει εγγενώς ένα κύκλωμα πραγματικού χρόνου (RTC). Λειτουργεί εντελώς ανεξάρτητα από τον κεντρικό πυρήνα ελέγχου. Χρησιμοποιώντας τους εσωτερικούς καταχωρητές του τσιπ το σύστημα μετράει δευτερόλεπτα, λεπτά και ώρες με χαρακτηριστική ακρίβεια, αποδίδοντας έτσι μια σωστή χρονική σφραγίδα σε οποιοδήποτε συμβάν ανιχνευτεί από τους αισθητήρες.

Η αποθήκευση των ευαίσθητων δεδομένων αποτελεί μια διαφορετική πρόκληση. Ο κωδικός ασφαλείας του χρήστη δεν νοείται να χάνεται κάθε φορά που η πλακέτα αποσυνδέεται από την τροφοδοσία. Η μνήμη SRAM είναι αυστηρά πτητική. Εφόσον η πλακέτα Nucleo δεν φέρει κάποιο ξεχωριστό τσιπ EEPROM, η λύση δόθηκε αποκλειστικά μέσω λογισμικού. Αξιοποιήθηκε η τεχνική της εξομοίωσης EEPROM πάνω στην ίδια την εσωτερική μνήμη Flash του μικροελεγκτή. Ο κώδικας δεσμεύει έναν συγκεκριμένο τομέα της Flash προκειμένου να εγγράφει μόνιμα τις ρυθμίσεις. Μια εξαιρετικά κομψή προσέγγιση η οποία καταργεί την ανάγκη προσθήκης εξωτερικών εξαρτημάτων στο κύκλωμα, εξασφαλίζοντας την απόλυτη ακεραιότητα των δεδομένων μετά από κάθε επανεκκίνηση.

3.5 Σύστημα Τροφοδοσίας

Η ενεργειακή διαχείριση καθορίζει τη σταθερότητα ολόκληρης της κατασκευής. Αποτελεί τη ραχοκοκαλιά του συστήματος. Συνήθως η πλακέτα Nucleo λαμβάνει ρεύμα μέσω της θύρας USB, τροφοδοτώντας στη συνέχεια τον κεντρικό γραμμικό ρυθμιστή ο οποίος αναλαμβάνει το κρίσιμο έργο της υποβίβασης στα επίπεδα των 3.3V για την απρόσκοπτη λειτουργία του μικροελεγκτή STM32. Μια

τέτοια συνθήκη θεωρείται απολύτως θεμελιώδης. Ο κεντρικός πυρήνας λειτουργεί αυστηρά με χαμηλή τάση. Παρόμοιες ηλεκτρικές απαιτήσεις εμφανίζουν οι μαγνητικές επαφές μαζί με το πληκτρολόγιο.

Ορισμένα περιφερειακά ωστόσο διαφοροποιούνται. Ο ανιχνευτής κίνησης HC-SR501 και η οθόνη υγρών κρυστάλλων ανήκουν ξεκάθαρα σε αυτή την κατηγορία, απαιτώντας για την ορθή λειτουργία τους τάση 5V, η οποία παρέχεται ευτυχώς απευθείας από την αντίστοιχη έξοδο ισχύος της αναπτυξιακής πλατφόρμας χωρίς να απαιτείται απολύτως κανένα εξωτερικό κύκλωμα μετατροπής.

Εκείνο όμως που κρύβει παγίδες είναι η ενεργοποίηση της σειρήνας. Ένας τοπικός βομβητής καταναλώνει απότομα ρεύμα κατά την παραβίαση, προκαλώντας ενδεχομένως μια στιγμιαία πτώση τάσης η οποία θα μπορούσε ίσως να οδηγήσει τον ευαίσθητο μικροελεγκτή σε ανεπιθύμητη επανεκκίνηση. Κάθε σωστός σχεδιασμός προλαμβάνει τέτοια καταστροφικά φαινόμενα. Διαπιστώθηκε πως το ενσωματωμένο τροφοδοτικό της πλακέτας ανταπεξέρχεται άριστα διαχειριζόμενο ταυτόχρονα το σύνολο των ηλεκτρικών φορτίων διατηρώντας τα επίπεδα τάσης συμπαγή κάτω από οποιεσδήποτε συνθήκες [7].

Κεφάλαιο 4ο: Ανάπτυξη Λογισμικού (Software)

4.1 Περιβάλλον Ανάπτυξης (IDE)

Η μετάβαση από τον σχεδιασμό του υλικού στη συγγραφή του κώδικα προϋποθέτει τη χρήση εξειδικευμένων ψηφιακών εργαλείων. Αποτελεί το κρίσιμότερο βήμα. Για την ανάπτυξη του συγκεκριμένου συστήματος ασφαλείας υιοθετήθηκε το επίσημο οικοσύστημα λογισμικού της STMicroelectronics, το οποίο παρέχει στους μηχανικούς μια απολύτως ενοποιημένη πλατφόρμα εργασίας καταργώντας πρακτικά την ανάγκη αναζήτησης εφαρμογών από τρίτους κατασκευαστές. Όλα βρίσκονται στο ίδιο περιβάλλον.

Η διαδικασία ξεκινά με την παραμετροποίηση. Εδώ αναλαμβάνει δράση το λογισμικό STM32CubeMX. Πρόκειται για ένα πανίσχυρο γραφικό εργαλείο διαμόρφωσης που επιτρέπει τον οπτικό καθορισμό της λειτουργίας κάθε ακροδέκτη του μικροελεγκτή, τη ρύθμιση του ρολογιού συστήματος και την ενεργοποίηση των απαραίτητων διαύλων επικοινωνίας. Η προσέγγιση αυτή εκμηδενίζει τα λάθη. Αντί να αναλώνεται χρόνος στη μελέτη αμέτρητων καταχωρητών ο χρήστης ορίζει τις παραμέτρους μέσω ενός φιλικού γραφικού περιβάλλοντος, αφήνοντας το πρόγραμμα να παράγει αυτόματα τον αρχικό κορμό του κώδικα σε γλώσσα C.

Αμέσως μετά ακολουθεί η κύρια φάση της προγραμματιστικής ανάπτυξης. Το παραγόμενο έργο εισάγεται στο STM32CubeIDE. Αυτό το λογισμικό βασίζεται στη στιβαρή αρχιτεκτονική του Eclipse, προσφέροντας έναν πανίσχυρο μεταγλωττιστή μαζί με προηγμένες δυνατότητες αποσφαλμάτωσης σε πραγματικό χρόνο. Εξαιρετικά βολικό. Ενσωματώνει πλήρως τις βιβλιοθήκες Hardware Abstraction Layer διευκολύνοντας δραματικά την πρόσβαση στα εσωτερικά περιφερειακά του τσιπ μέσω αφηρημένων και εύκολα κατανοητών συναρτήσεων. Ο σχεδιαστής ασχολείται πλέον αποκλειστικά με τη λογική της εφαρμογής [8].

4.2 Λογική Συστήματος(State Machine)

Η αρχιτεκτονική του λογισμικού δεν ακολουθεί μια απλή γραμμική εκτέλεση εντολών. Βασίζεται σε συγκεκριμένες καταστάσεις. Ο σχεδιασμός υλοποιήθηκε αυστηρά σύμφωνα με το πρότυπο μιας πεπερασμένης μηχανής καταστάσεων, η οποία εξασφαλίζει ότι ο μικροελεγκτής γνωρίζει ανά πάσα στιγμή την ακριβή λειτουργική φάση της κατασκευής αποφεύγοντας κρίσιμα λογικά σφάλματα κατά την αλληλεπίδραση με τον χρήστη. Κάθε μετάβαση ελέγχεται απόλυτα. Η οπτικοποίηση αυτής της προγραμματιστικής ροής αποτυπώνεται συνήθως μέσω ενός διαγράμματος ελέγχου περιγράφοντας αναλυτικά τις τέσσερις βασικές συνθήκες λειτουργίας.

Κατάσταση Αφόπλισης (Disarmed). Αποτελεί τη φυσιολογική συνθήκη ηρεμίας. Κατά τη διάρκεια αυτής της φάσης ο κεντρικός πίνακας αγνοεί τα σήματα παραβίασης από την περιμετρική προστασία ή τα εσωτερικά ραντάρ, επιτρέποντας την εντελώς ελεύθερη κίνηση των ατόμων στον χώρο. Το λογισμικό παραμένει σε συνεχή εγρήγορση αναμένοντας την εισαγωγή του σωστού κωδικού από το πληκτρολόγιο, ενώ ταυτόχρονα η οθόνη προβάλλει το αντίστοιχο μήνυμα ετοιμότητας.

Κατάσταση Όπλισης (Armed). Το επίπεδο ασφαλείας μεγιστοποιείται. Ο μικροελεγκτής αρχίζει να σαρώνει αδιάκοπα τις ψηφιακές του εισόδους παρακολουθώντας στενά τις μαγνητικές επαφές μαζί με τους αισθητήρες κίνησης για οποιαδήποτε απότομη μεταβολή του ηλεκτρικού δυναμικού. Μηδενική ανοχή σε διαρρήξεις. Αν μια ζώνη ενεργοποιηθεί η μηχανή καταστάσεων εγκαταλείπει αμέσως τη συγκεκριμένη ρουτίνα μεταβαίνοντας ακαριαία στο επόμενο δραστικό στάδιο.

Συνθήκη Συναγερμού (Alarm). Πρόκειται για την πιο κρίσιμη λειτουργία της κατασκευής. Η ηχητική ειδοποίηση ενεργοποιείται αμέσως μέσω του τοπικού βομβητή υποδεικνύοντας σαφώς την παρουσία κάποιου εισβολέα. Το σύστημα εγκλωβίζεται σκοπίμως σε αυτόν τον βρόχο. Αρνείται πεισματικά οποιαδήποτε άλλη εξωτερική εντολή μέχρι ο νόμιμος χρήστης να πληκτρολογήσει το σωστό συνθηματικό, οπότε και το λογισμικό επιστρέφει ομαλά στην αρχική συνθήκη ηρεμίας.

```

graph TD
    Start([ΑΡΧΗ]) --> Init[ΕΦΑΓΚΟΤΟΙΧΙΣΕ  
ΔΙΑΡΧΗΣΕ ΜΕΤΑΒΛΗΤΩΝ]
    Init --> MainMenu[ΕΜΦΑΝΙΣΕ ΣΤΗΝ ΟΘΟΝΗ:  
HELLO  
ENTER PASSWORD  
OR  
PRESS THE BUTTON]
    
    MainMenu --> SystemStatus{ΣΥΣΤΗΜΑ  
ΟΠΙΣΘΕΜΕΝΟ}
    
    SystemStatus -- ΝΑΙ --> MainMenu
    SystemStatus -- ΟΧΙ --> EnterPIN[ΠΑΤΗΝΗΣΕ ΠΛΑΚΙΔΙΟ]
    
    EnterPIN --> PINValid{ΠΑΤΗΝΗΣΕ ΑΡΙΘ}
    EnterPIN --> PINInvalid[ΠΑΤΗΝΗΣΕ ΚΑΠΟΙΟΥ Η  
ΚΑΠΟΙΑ ΑΠΟ ΤΑ ΑΔΣ]
    
    PINValid --> NewPIN[ΕΜΦΑΝΙΣΕ ΣΤΗΝ ΟΘΟΝΗ:  
ENTER NEW PIN]
    PINValid --> PINWrong[ΠΑΤΗΝΗΣΕ ΛΑΘΟΣ ΚΩΔΙΚΟΣ]
    
    PINWrong -- ΝΑΙ --> Locked[ΕΜΦΑΝΙΣΕ ΣΤΗΝ ΟΘΟΝΗ:  
WRONG CODE]
    PINWrong -- ΟΧΙ --> Armed[ΕΜΦΑΝΙΣΕ ΣΤΗΝ ΟΘΟΝΗ:  
FULL SYSTEM ARMED]
    
    NewPIN --> SavePIN[ΑΤΟΠΗΚΕΥΣΕ ΚΩΔΙΚΟ  
ΕΜΦΑΝΙΣΕ ΠΑΡΟΛΗ SAVED]
    SavePIN --> MainMenu
    
    PINInvalid --> SelectZone[ΠΑΤΗΝΗΣΕ ΖΩΝΑ  
ΚΑΠΟΙΟΥ Η ΚΑΠΟΙΑ  
ΑΠΟ ΤΑ ΠΑΡΑΒΛΗΤΑ ΠΛΑΚΙΔΙΑ]
    PINInvalid --> NoZone[ΠΑΤΗΝΗΣΕ TO #]
    
    NoZone --> NoZoneMsg[ΕΜΦΑΝΙΣΕ ΣΤΗΝ ΟΘΟΝΗ:  
NO ZONE SELECTED]
    NoZoneMsg --> MainMenu
    
    SelectZone --> ZoneList[ΟΤΙΩΣ ΤΗΝ Ή ΤΩΝ ΖΩΝΩΝ]
    ZoneList --> Zone1[ΕΙΣΑΓΩΓΗ ΓΩΝΙΟΥ  
4ΨΗΦΙΟΥ ΚΩΔΙΚΟΥ]
    ZoneList --> Zone2[ΕΙΣΑΓΩΓΗ ΑΒΟΥΤΣ  
4ΨΗΦΙΟΥ ΚΩΔΙΚΟΥ]
    
    Zone1 --> MotionAlarm[ΑΝΙΧΝΕΥΣΗ ΚΙΝΗΣΗΣ]
    Zone2 --> DoorAlarm[ΑΝΙΧΝΕΥΣΗ ΠΟΡΤΑΣ]
    Zone2 --> WindowAlarm[ΑΝΙΧΝΕΥΣΗ ΠΑΡΑΒΛΗΤΟΥ]
    
    MotionAlarm --> MotionDetected{ΑΝΙΧΝΕΥΕΤΕ  
ΚΙΝΗΣΗ}
    MotionDetected -- ΝΑΙ --> Delay[DELAY 25s]
    Delay --> MotionSounded[ΕΚΦΕΡΟΝΤΕΣ ΣΕΡΦΗΝΑΓΕΙ]
    MotionSounded --> MotionReached[ΕΜΦΑΝΙΣΕ ΣΤΗΝ ΟΘΟΝΗ:  
MOTION DETECTED]
    MotionReached --> MainMenu
    
    MotionDetected -- ΟΧΙ --> Locked
    
    DoorAlarm --> DoorOpen{ΑΝΙΧΝΕΙ Η  
ΠΟΡΤΑ}
    DoorOpen -- ΝΑΙ --> DoorSounded[ΕΚΦΕΡΟΝΤΕΣ ΣΕΡΦΗΝΑΓΕΙ]
    DoorSounded --> DoorReached[ΕΜΦΑΝΙΣΕ ΣΤΗΝ ΟΘΟΝΗ:  
DOOR REACHED]
    DoorReached --> MainMenu
    
    DoorOpen -- ΟΧΙ --> Locked
    
    WindowAlarm --> WindowOpen{ΑΝΙΧΝΕΙ ΤΟ  
ΠΑΡΑΒΛΗΤΟ}
    WindowOpen -- ΝΑΙ --> WindowSounded[ΕΚΦΕΡΟΝΤΕΣ ΣΕΡΦΗΝΑΓΕΙ]
    WindowSounded --> WindowReached[ΕΜΦΑΝΙΣΕ ΣΤΗΝ ΟΘΟΝΗ:  
WINDOW REACHED]
    WindowReached --> MainMenu
    
    WindowOpen -- ΟΧΙ --> Locked
    
    Locked --> Unlocked[ΕΜΦΑΝΙΣΕ ΣΤΗΝ ΟΘΟΝΗ:  
SYSTEM UNLOCKED]
    Unlocked --> MainMenu
  
```

15

4.3 Διαχείριση Περιφερειακών (Drivers & Interrupts)

Η αλληλεπίδραση του κεντρικού επεξεργαστή με τα εξωτερικά κυκλώματα καθορίζει τη συνολική ταχύτητα απόκρισης. Είναι ένα κρίσιμο σχεδιαστικό μέγεθος. Η συλλογή δεδομένων από τις περιφερειακές μονάδες μπορεί να υλοποιηθεί μέσω δύο εκ διαμέτρου αντίθετων προγραμματιστικών προσεγγίσεων. Την τεχνική της συνεχούς ερώτησης (polling) και τον μηχανισμό των εξωτερικών διακοπών (interrupts).

Στα συστήματα ασφαλείας ο χρόνος θεωρείται αμείλικτος. Μια παραβίαση πρέπει να καταγραφεί ακαριαία. Η μέθοδος polling αναγκάζει τον μικροελεγκτή να ελέγχει αδιάκοπα και κυκλικά την κατάσταση κάθε αισθητήρα καταναλώνοντας τεράστιους υπολογιστικούς πόρους για να διαπιστώσει συνήθως ότι τίποτα δεν έχει αλλάξει. Αντίθετα η χρήση εξωτερικών διακοπών (EXTI) αλλάζει ριζικά την αρχιτεκτονική του λογισμικού. Ο πυρήνας εκτελεί ανενόχλητος την κύρια ρουτίνα του. Όταν μια μαγνητική επαφή ανοίξει το ίδιο το υλικό παράγει ένα ασύγχρονο ηλεκτρικό σήμα το οποίο αναγκάζει τον επεξεργαστή να εγκαταλείψει αμέσως την τρέχουσα διεργασία του μεταπηδώντας στην αντίστοιχη ρουτίνα εξυπηρέτησης. Εξασφαλίζεται έτσι η απόλυτη ντετερμινιστική συμπεριφορά που απαιτεί η ανίχνευση διάρρηξης.

Η λογική διαφοροποιείται αισθητά στη διαχείριση του πληκτρολογίου. Εκεί ακολουθείται αυστηρά η μέθοδος της διαδοχικής σάρωσης. Η κατασκευή διαθέτει δεκαέξι πλήκτρα οργανωμένα σε ένα πλέγμα τεσσάρων γραμμών και τεσσάρων στηλών [12]. Αντιστοιχούν σε οκτώ μόλις ακροδέκτες. Ο αλγόριθμος τροφοδοτεί αρχικά όλες τις γραμμές με υψηλό δυναμικό. Στη συνέχεια μηδενίζει διαδοχικά μία προς μία την κάθε γραμμή ελέγχοντας ταυτόχρονα τις επιστροφές από τις στήλες.

Μόλις εντοπιστεί χαμηλό δυναμικό σε κάποια στήλη το σύστημα αποκρυπτογραφεί τις ακριβείς συντεταγμένες της πίεσης. Υπάρχει ωστόσο ένας φυσικός περιορισμός. Οι μηχανικές επαφές των πλήκτρων πάλλονται κατά το κλείσιμο. Για τον λόγο αυτόν ο κώδικας ενσωματώνει μια μικρή προγραμματιστική παύση (delay) αμέσως μετά την πρώτη ανίχνευση προκειμένου να φιλτράρει τον ηλεκτρικό θόρυβο αγνοώντας τις πολλαπλές ψευδείς καταγραφές του ίδιου πατήματος.

4.4 Διαχείριση Κωδικών και Ασφάλεια

Η ασφάλεια της κατασκευής βασίζεται αποκλειστικά στον έλεγχο πρόσβασης. Αποτελεί τον πυρήνα του συστήματος. Η διαχείριση των κωδικών υλοποιείται μέσω μιας σειράς προγραμματιστικών ελέγχων οι οποίοι διασφαλίζουν την ορθή ταυτοποίηση του χρήστη πριν από οποιαδήποτε κρίσιμη ενέργεια όπλισης. Κατά τη διαδικασία αυτή ο κώδικας συγκεντρώνει τα πατήματα των αριθμών αποθηκεύοντάς τα προσωρινά σε έναν μονοδιάστατο πίνακα χαρακτήρων. Η επιβεβαίωση γίνεται ακαριαία. Αξιοποιείται η κλασική συνάρτηση σύγκρισης συμβολοσειρών της γλώσσας προγραμματισμού C. Ο επεξεργαστής ελέγχει τον πίνακα της τρέχουσας εισαγωγής με το αποθηκευμένο συνθηματικό απαιτώντας την απόλυτη ταύτιση όλων των ψηφίων.

Ο νόμιμος κάτοχος διατηρεί το δικαίωμα αλλαγής του κλειδιού ασφαλείας. Η διαδικασία ξεκινάει με το πάτημα του ειδικού συμβόλου του αστερίσκου. Το λογισμικό αντιλαμβάνεται την πρόθεση τροποποίησης ενεργοποιώντας μια ειδική μεταβλητή κατάστασης η οποία αλλάζει ριζικά τη συμπεριφορά του αλγορίθμου διαβάζοντας πλέον τις εισόδους ως νέα δεδομένα. Το σύστημα αναμένει τέσσερα καινούργια ψηφία. Μόλις ολοκληρωθεί η πληκτρολόγηση ο νέος αριθμός αντικαθιστά τον παλιό μέσω μιας απλής εντολής αντιγραφής. Η οθόνη επιβεβαιώνει την επιτυχή καταχώρηση.

Η φυσική αποθήκευση των δεδομένων ακολουθεί μια απολύτως ξεκάθαρη προσέγγιση. Αποφεύχθηκαν οι πολύπλοκες ρουτίνες μόνιμης εγγραφής. Βάσει του τελικού πηγαίου κώδικα το ενεργό συνθηματικό φιλοξενείται αποκλειστικά στην πτητική μνήμη SRAM του μικροελεγκτή. Η συγκεκριμένη επιλογή απλοποιεί δραματικά την ανάπτυξη του λογισμικού απαλλάσσοντας τον σχεδιαστή από τη χρονοβόρα

διαχείριση των σελίδων της μνήμης Flash. Υπάρχει ωστόσο ένας περιορισμός. Σε περίπτωση γενικής διακοπής της κεντρικής τροφοδοσίας ο τροποποιημένος κωδικός χάνεται οριστικά. Η μητρική πλακέτα επανέρχεται αναγκαστικά στην εργοστασιακή της προεπιλογή κατά την επόμενη επανεκκίνηση.

4.5 Υλοποίηση Καταγραφής(Logging Algorithm)

Η παρακολούθηση των παραβιάσεων δεν απαιτεί απαραίτητα πολύπλοκα συστήματα εξωτερικής μνήμης. Η σχεδιαστική προσέγγιση είναι εντελώς διαφορετική. Στον συγκεκριμένο κώδικα αποφεύχθηκε σκόπιμα η δημιουργία ενός κλασικού αρχείου κειμένου με πολύπλοκες χρονοσφραγίδες, δίνοντας αντίθετα το απόλυτο βάρος στην άμεση οπτική ενημέρωση του χειριστή μέσω της οθόνης.

Πρόκειται για μια δυναμική αποτύπωση, κατά την οποία ο αλγόριθμος αξιοποιεί τρεις διακριτές μεταβλητές μανδάλωσης προκειμένου να αποθηκεύσει με αυστηρή χρονολογική σειρά την αλληλουχία των γεγονότων κατά τη διάρκεια ενός συναγερμού. Αυτό ακριβώς αποτελεί τον πυρήνα της καταγραφής. Μόλις το ραντάρ ή κάποια μαγνητική επαφή διεγερθεί, το λογισμικό κλειδώνει την πηγή του σφάλματος στην πρώτη διαθέσιμη θέση, συνεχίζοντας ταυτόχρονα να σαρώνει το περιβάλλον αναζητώντας ενδεχόμενη δεύτερη ή και τρίτη συνεχόμενη διάρρηξη από τις υπόλοιπες προστατευόμενες ζώνες. Το συγκριμένο σενάριο αποδίδεται από το παρακάτω τμήμα του κώδικα.

```
//ΕΛΕΓΧΟΣ ΓΙΑ 2η ΠΑΡΑΒΙΑΣΗ |
if (alarm_source_2 == 0)
{
    if (alarm_source != 1 && (zone_A_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOC, GPIO_PIN_0) == GPIO_PIN_SET)
        alarm_source_2 = 1;
    else if (alarm_source != 2 && (zone_B_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOB, DOOR_Pin) == GPIO_PIN_RESET)
        alarm_source_2 = 2;
    else if (alarm_source != 4 && (zone_C_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOB, WINDOW_Pin) == GPIO_PIN_RESET)
        alarm_source_2 = 4;
}
```

Εικόνα 8: Έλεγχος για 2^η παραβίαση

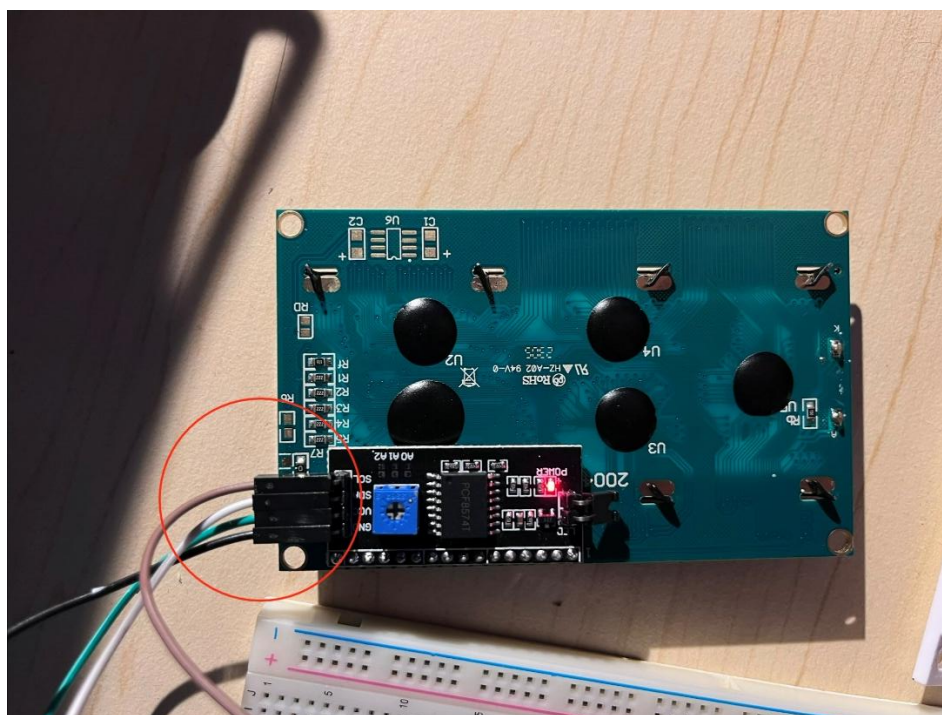
Η πληροφορία διατηρείται απολύτως ανέπαφη. Αντί να εξάγεται μια τυπική γραμμή ημερολογίου στον σειριακό δίαυλο, η μορφή της εγγραφής μεταφράζεται ζωντανά στο περιβάλλον διεπαφής με ξεκάθαρα μηνύματα κειμένου. Τα συμβάντα αριθμούνται διαδοχικά. Η πρώτη γραμμή υποδεικνύει την αρχική αιτία της ενεργοποίησης, ενώ οι επόμενες συμπληρώνονται σταδιακά σχηματίζοντας ένα πλήρες τοπικό ιστορικό της πορείας του εισβολέα μέσα στον χώρο, το οποίο παραμένει ορατό και εγκλωβισμένο μέχρι ο νόμιμος κάτοχος να καταχωρήσει το σωστό συνθηματικό πρόσβασης.

Κεφάλαιο 5ο: Υλοποίηση, Δοκιμές και Αποτελέσματα

5.1 Κατασκευή Πρωτοτύπου σε Breadboard

Η θεωρία οφείλει να δοκιμάζεται στην πράξη. Το αρχικό στάδιο υλοποίησης αφορά αποκλειστικά τη συναρμολόγηση των εξαρτημάτων πάνω σε μια πλακέτα δοκιμών. Εκεί αποκαλύπτονται τα πραγματικά προβλήματα. Η φυσική τοποθέτηση του μικροελεγκτή STM32 και η χωροταξική κατανομή των περιφερειακών μονάδων στο περιορισμένο πλέγμα οπών απαιτεί αυστηρή μεθοδολογία προκειμένου να διασφαλιστεί η ορθή τροφοδοσία όλων των ανεξάρτητων υποσυστημάτων. Κάθε καλώδιο διαδραματίζει τον δικό του κρίσιμο ρόλο.

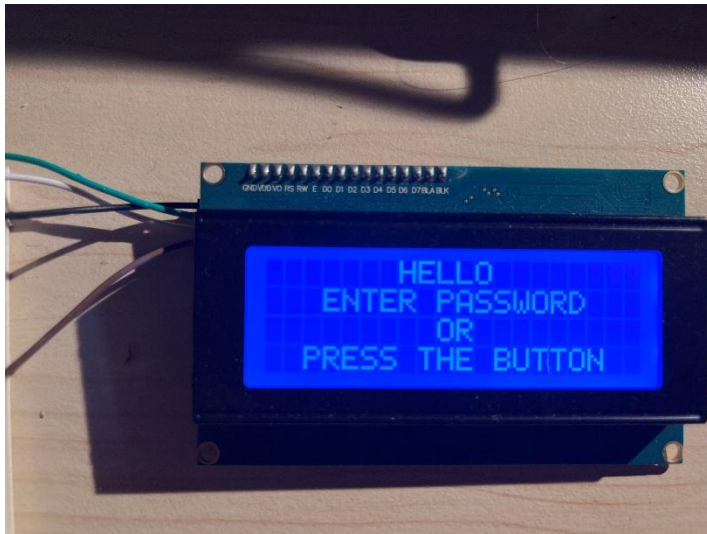
Τα συνοδευτικά φωτογραφικά στιγμιότυπα αποτυπώνουν την τελική μορφή του έργου, απεικονίζοντας την πολυπλοκότητα του εγχειρήματος. Η καλωδίωση οργανώθηκε με γνώμονα τον αυστηρό διαχωρισμό των γραμμών ισχύος από τα ευαίσθητα σήματα δεδομένων. Η τάση διανέμεται περιμετρικά. Αντίθετα οι ψηφιακές επαφές των αισθητήρων οδεύουν κεντρικά προς τους ακροδέκτες της πλακέτας Nucleo σχηματίζοντας ένα πυκνό αλλά απολύτως ανιχνεύσιμο δίκτυο επικοινωνίας. Η χρωματική κωδικοποίηση των αγωγών κρίθηκε απαραίτητη. Βοήθησε αφάνταστα.



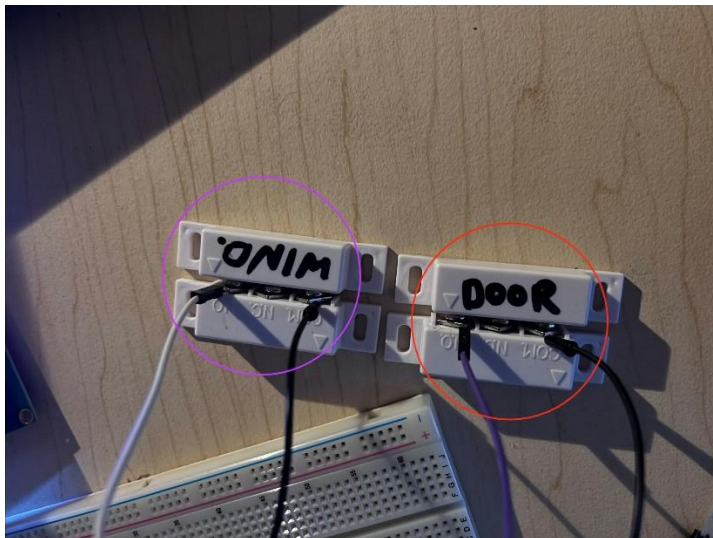
Εικόνα 9: Πίσω Μέρος Οθόνης. Πλακέτα Διεπαφής I2C

Πίνακας 6: Συνδεσμολογία Οθόνης

Μαύρο Καλώδιο	GND
Πράσινο Καλώδιο	VCC(+5V)
Άσπρο Καλώδιο	SDA(PB9)
Καφέ Καλώδιο	SCL(PB8)



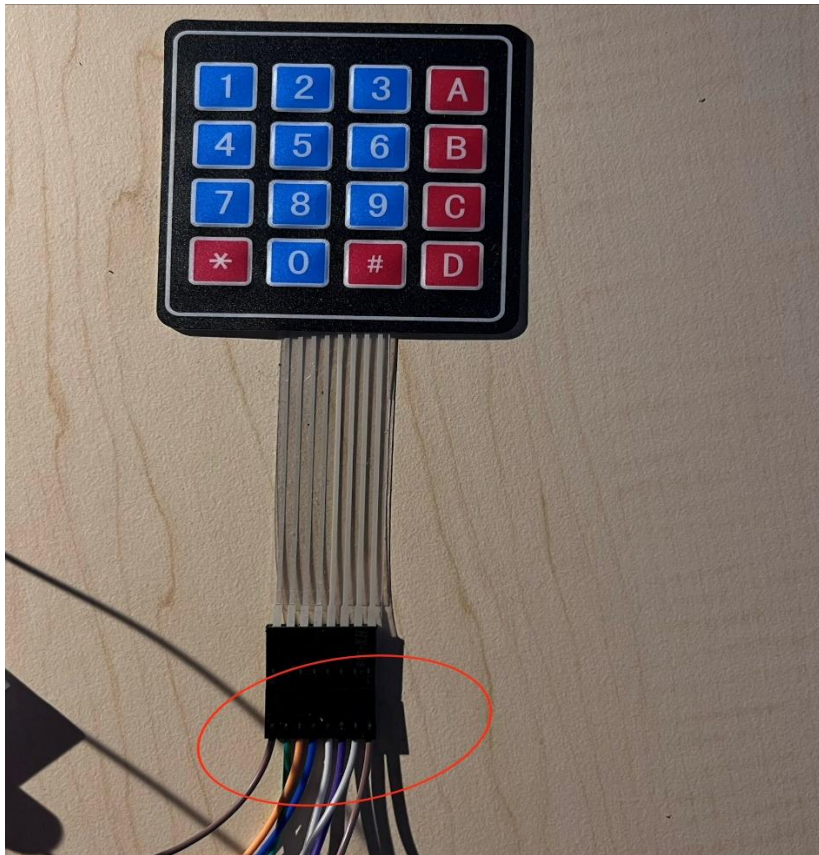
Εικόνα 10: Μπροστά Μέρος Οθόνης



Εικόνα 11: Μαγνητικές Επαφές (Πόρτα & Παράθυρο)

Πίνακας 7: Συνδεσμολογία Μαγνητικών Επαφών

Μαύρο Καλώδιο	GND
Μωβ/Άσπρο Καλώδιο	PC1/PC2

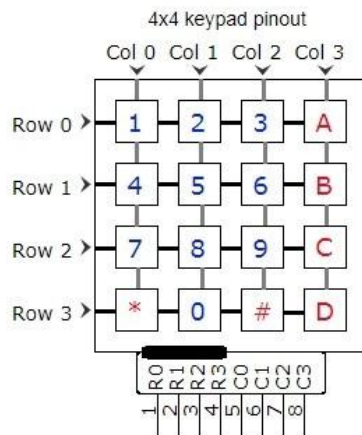


Εικόνα 12: Πληκτρολόγιο

Η συνδεσμολογία του πληκτρολογίου ξεκινάει από τα αριστερά προς τα δεξιά και πιο συγκεκριμένα τα πρώτα 4 pin είναι σειρές και τα επόμενα 4 pin είναι στήλες.

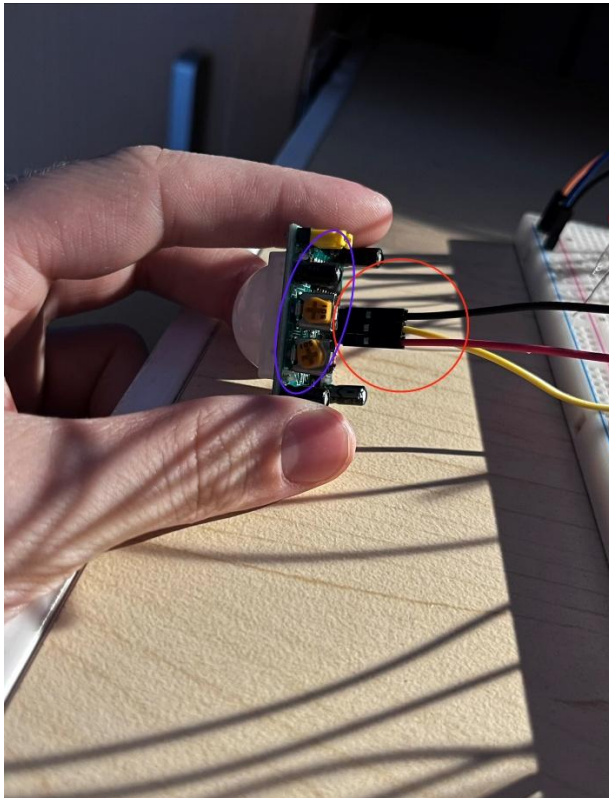
Πίνακας 8: Συνδεσμολογία Πληκτρολογίου

R0-R3	PA4-PA7
C0-C3	PC4-PC7



Εικόνα 13: Εσωτερική Συνδεσμολογία Πληκτρολογίου

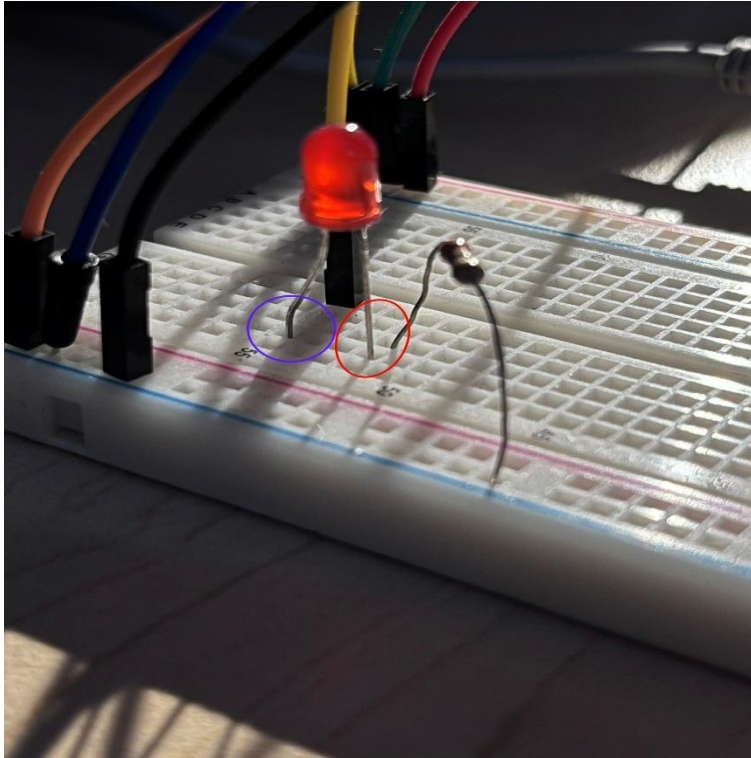
Στη παρακάτω εικόνα στο κόκκινο κύκλο απεικονίζεται η συνδεσμολογία του αισθητήρα κίνησης με τον μικροελεγκτή, ενώ στο μωβ κύκλο βρίσκονται η χρονοκαθυστέρηση και η ευαισθησία του αισθητήρα.



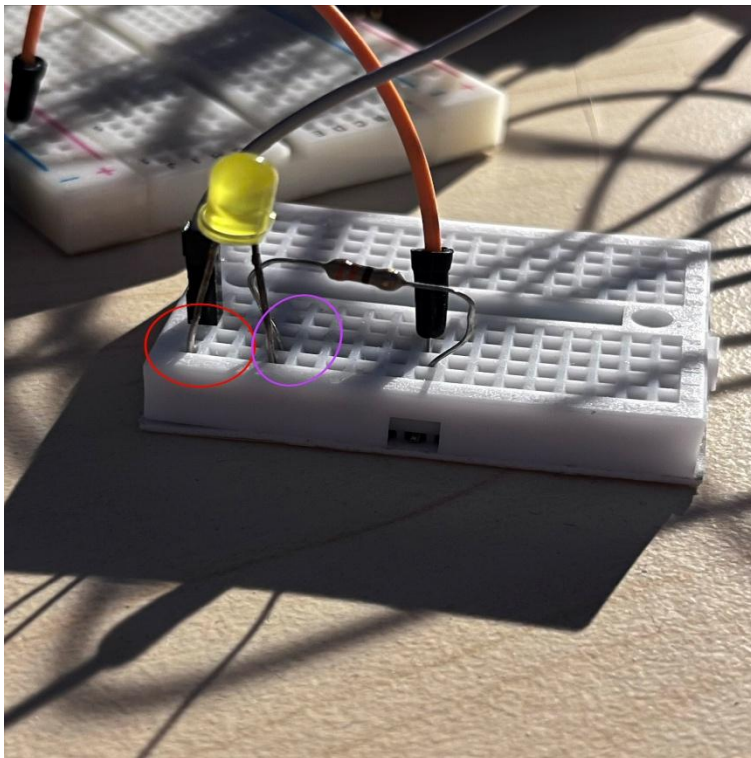
Εικόνα 14: Αισθητήρας Κίνησης

Πίνακας 9: Συνδεσμολογία του Αισθητήρα Κίνησης

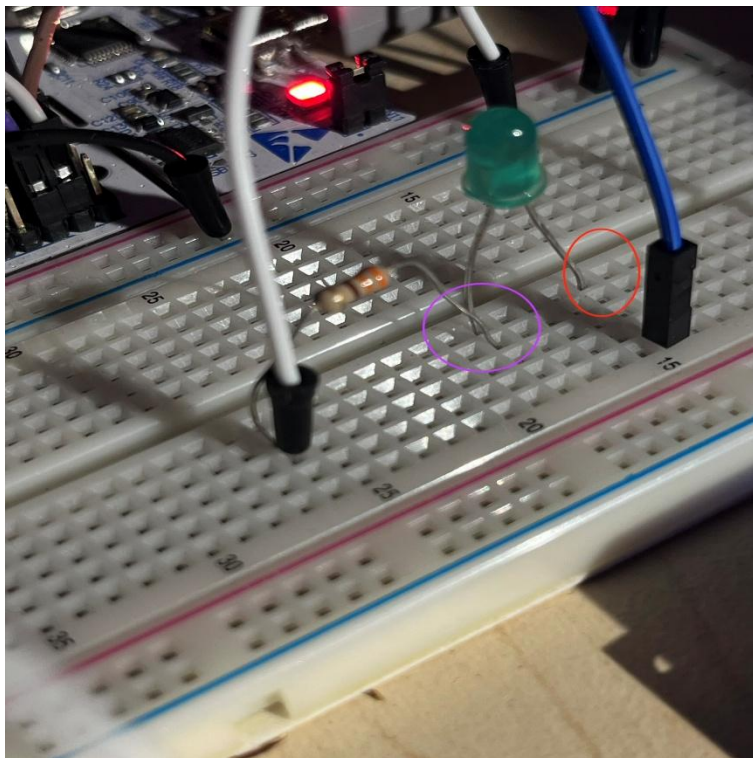
Μαύρο Καλώδιο	GND
Κίτρινο Καλώδιο	OUTPUT (PC0)
Κόκκινο Καλώδιο	VCC(+5V)



Εικόνα 15: Led Αισθητήρα Κίνησης /Led Ειδοποίησης Καταστάσεων

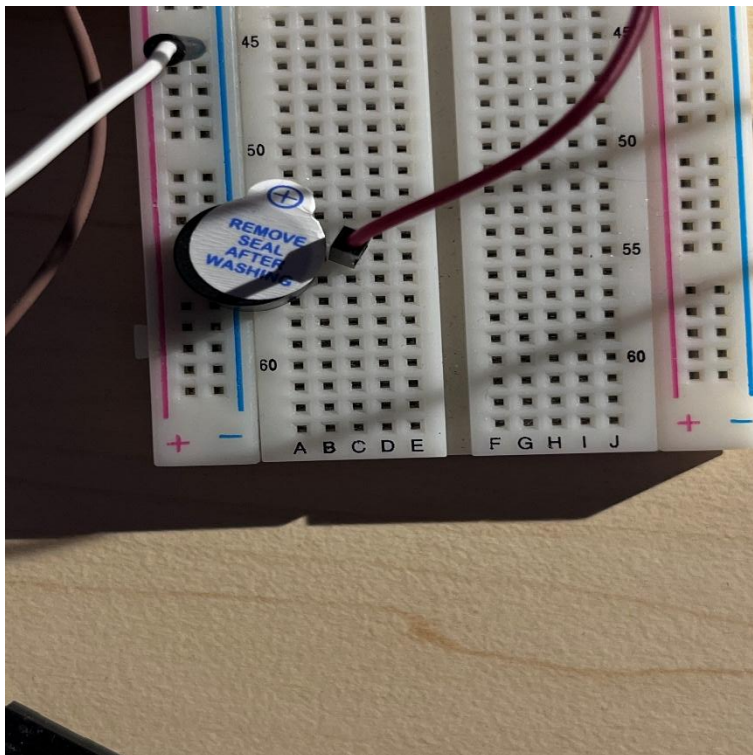


Εικόνα 16: Led Μαγνητικής Επαφής Παραθύρου



Εικόνα 17: Led Μαγνητικής Επαφής Πόρτας

Τα led συνδέονται όλα με τον εξής τρόπο η άνοδος συνδέεται κατευθείαν σε ψηφιακό pin του μικροελεγκτή και η κάθοδος συνδέεται μέσω μιας αντίστασης 330Ω στη γείωση.



Εικόνα 18: Βομβητής

Ο βομβητής συνδέεται κατευθείαν στον μικροελεγκτή σε ψηφιακό pin (positive) και στη γείωση (negative).

Η πρακτική εφαρμογή κρύβει σημαντικές ηλεκτρικές παγίδες. Μια χαλαρή σύνδεση αρκεί. Μπορεί να καταρρεύσει ολόκληρη η προγραμματιστική λογική. Οι αδέσποτες χωρητικότητες που αναπτύσσονται φυσιολογικά ανάμεσα στις παράλληλες μεταλλικές λωρίδες της βάσης προκαλούν συχνά ανεπιθύμητες ταλαντώσεις μεταβάλλοντας ακαριαία το ηλεκτρικό δυναμικό. Ο ηλεκτρομαγνητικός θόρυβος αποτελεί τον μεγαλύτερο εχθρό. Τα μακριά καλώδια μετατρέπονται εύκολα σε μικρές κεραίες συλλέγοντας παράσιτα από τον περιβάλλοντα χώρο, γεγονός που επέβαλε την άμεση υιοθέτηση αυστηρών προγραμματιστικών τεχνικών εξομάλυνσης στον τελικό κώδικα προκειμένου ο επεξεργαστής να αγνοήσει πλήρως τις ψευδείς αναπηδήσεις. Η σταθερότητα τελικά επιτεύχθηκε.

5.2 Σενάρια Ελέγχου (Test Cases)

Η ολοκληρωμένη λειτουργία του εν λόγω συστήματος συναγερμού (όπλιση-αφόπλιση, ανίχνευση κίνησης καθώς και ανίχνευση παραβίασης μέσω μαγνητικών επαφών) παρουσιάζονται συνολικά σε βίντεο, ακολουθώντας τους παρακάτω συνδέσμους.

https://drive.google.com/file/d/1jZ7mUkjR6i15hJ8VaoLCwHlkyrSvcRl0/view?usp=drive_link

https://drive.google.com/file/d/1SNKsGSyot1eEH4KJiP5ugznfbDQDsXU/view?usp=drive_link

https://drive.google.com/file/d/1tBRUKCLetx3cAB1ApZdkjtTORKwyOwRA/view?usp=drive_link

Εν συνεχεία, ακολουθεί η αναλυτική περιγραφή των επιμέρους συνθηκών. Το συγκεκριμένο σύστημα συναγερμού διαθέτει ορισμένα σενάρια τα οποία χωρίζονται σε τρεις βασικές κατηγορίες και παρουσιάζονται συνολικά :

Δοκιμή οπλισμού/αφοπλισμού με σωστό και λάθος PIN: Μόλις ο χρήστης βάλει τον σωστό κωδικό το σύστημα εισέρχεται στην κατάσταση του πλήρη οπλισμού. Εάν όμως δεν μπει ο σωστός κωδικός τρεις φορές το σύστημα ξεκινάει να ηχεί ο βομβητής και ο μονός τρόπος να σταματήσει είναι να μπει ο σωστός κωδικός. Όσον αφορά την αφόπλιση ισχύουν ακριβώς τα ίδια με την πλήρη όπλιση.



Εικόνα 19:Μήνυμα της οθόνης στο delay 15s



Εικόνα 20:Μήνυμα της οθόνης μόλις οπλιστεί πλήρως



Εικόνα 21:Μήνυμα οθόνης μετά την εισαγωγή λάθος κωδικού πρώτη φορά



Εικόνα 22: Μήνυμα οθόνης μετά την εισαγωγή λάθος κωδικού δεύτερη φορά



Εικόνα 23: Μήνυμα οθόνης μετά την εισαγωγή λάθος κωδικού τρίτη φορά

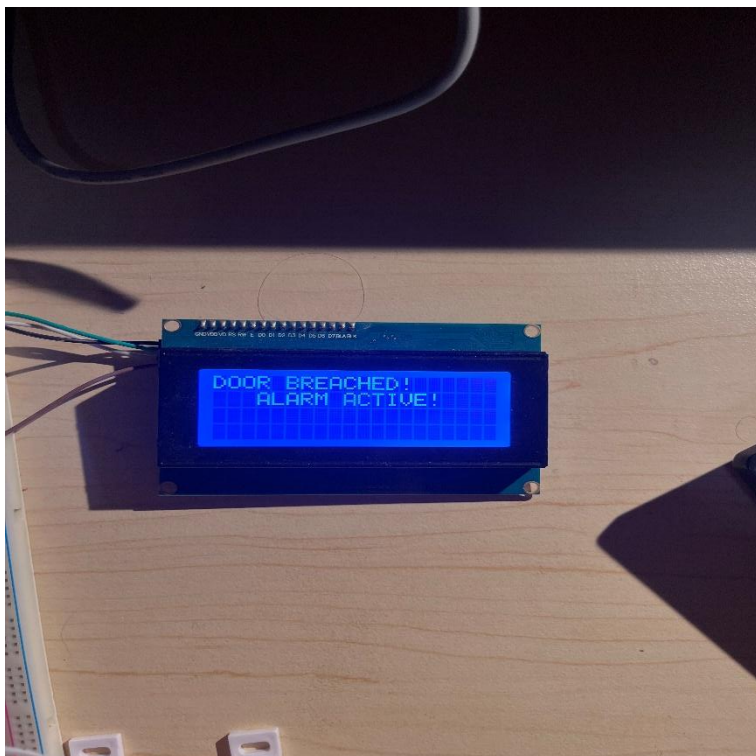
Προσομοίωση παραβίασης μεμονωμένων και πολλαπλών ζωνών ταυτόχρονα: Ο χρήστης επιπλέον μπορεί να οπλίσει ξεχωριστά την κάθε ζώνη αλλά και σε ζευγάρια. Ωστόσο στη περίπτωση που οπλιστούν δύο ζώνες ή και τρεις τα μηνύματα παραβιάσεων εμφανίζονται με τη σειρά που συνέβησαν. Παραδείγματος χάρη αν οπλιστούν οι ζώνες της πόρτας και το αισθητήρα οι παραβιάσεις θα εμφανιστούν με τη σωστή χρονολογική σειρά.



Εικόνα 24:Επιλογή Ζωνών



Εικόνα 25:Ανίχνευση Κίνησης



Εικόνα 26:Παραβίαση Πόρτας



Εικόνα 27:Ανίχνευση Κίνησης – Παραβίαση Πόρτας



Εικόνα 28: Παραβίαση Πόρτας – Ανίχνευση Κίνησης

Δοκιμή λειτουργίας χρονοκαθυστέρησης (Entry/Exit delay): Τέλος σε περίπτωση όπλισης υπάρχει μια μικρή χρονοκαθυστέρηση 15 δευτερολέπτων πριν οπλίσει δίνοντας τον απαραίτητο χρόνο στον χρήστη να απομακρυνθεί από το χώρο. Ακόμη και όταν είναι οπλισμένο και ανιχνευθεί κίνηση συμβαίνει ακριβώς το ίδιο σενάριο έτσι ώστε ο χρήστης αυτή τη φορά να εισέλθει άνετος στον χώρο του.



Εικόνα 29:Μήνυμα της οθόνης στο delay 15s για πλήρη όπλιση



Εικόνα 30:Ανίχνευση Κίνησης και delay 15s



Εικόνα 31: Μήνυμα της οθόνης στο delay 15s για όπλιση ζωνών

Κεφάλαιο 6ο: Συμπεράσματα και Μελλοντικές Επεκτάσεις

6.1 Σύνοψη Αποτελεσμάτων

Η ολοκλήρωση της πειραματικής διάταξης επιβεβαιώνει την πλήρη επίτευξη των σχεδιαστικών στόχων. Το σύστημα λειτουργεί ακριβώς όπως προβλέφθηκε. Ο κεντρικός αλγόριθμος ανταποκρίνεται με απόλυτη συνέπεια στις προδιαγραφές της μηχανής καταστάσεων, διαχειριζόμενος τις πολύπλοκες μεταβάσεις από την απλή αναμονή έως την ενεργοποίηση της σειράς χωρίς λογικά σφάλματα. Η επικοινωνία με τον χειριστή κατά την εισαγωγή δεδομένων εκτελείται ακαριαία.

Η δοκιμή του κυκλώματος πάνω σε breadboard αποτέλεσε το πιο απαιτητικό στάδιο της αξιολόγησης. Προέκυψαν αρκετές κατασκευαστικές προκλήσεις. Οι προσωρινές καλωδιώσεις δημιουργούν εγγενώς ένα ασταθές ηλεκτρικό περιβάλλον το οποίο ευνοεί την ανάπτυξη παρασιτικών τάσεων. Κατά τη φάση της αποσφαλμάτωσης εντοπίστηκε μια ελαττωματική μαγνητική επαφή. Το συγκεκριμένο εξάρτημα παρήγαγε λανθασμένα σήματα προς τον μικροελεγκτή διαταράσσοντας την ομαλή ροή του κώδικα. Η αντιμετώπιση του προβλήματος απαίτησε ενδελεχή διαγνωστικό έλεγχο στο φυσικό επίπεδο του υλικού. Μέσα από αυτή τη διαδικασία αναδείχθηκε ξεκάθαρα η αξία των εσωτερικών αντιστάσεων του επεξεργαστή για την απομόνωση του περιβαλλοντικού θορύβου.

Συνολικά το ηλεκτρονικό πρωτότυπο επέδειξε εξαιρετική συμπεριφορά. Η κεντρική μονάδα παρέμεινε ενεργή για μεγάλα χρονικά διαστήματα χωρίς να παρατηρηθεί καμία αυθόρμητη επανεκκίνηση. Αποδείχθηκε στην πράξη πως η παρούσα υλοποίηση συνιστά μια απολύτως αξιόπιστη βάση για τη μελλοντική μεταφορά της κατασκευής σε ένα επαγγελματικό τυπωμένο κύκλωμα.

6.2 Περιορισμοί της Υλοποίησης

Κάθε πειραματική διάταξη παρουσιάζει συγκεκριμένες αδυναμίες. Το περιβάλλον του breadboard εξυπηρετεί άριστα τους σκοπούς της αρχικής ανάπτυξης, συνοδεύεται ωστόσο από εγγενείς τεχνικούς περιορισμούς. Μια βασική δυσκολία εντοπίζεται στην απουσία μόνιμων ηλεκτρικών κολλήσεων. Η φυσική επαφή των εξαρτημάτων βασίζεται αποκλειστικά σε ελατήρια συγκράτησης, γεγονός που καθιστά το σύστημα εξαιρετικά ευάλωτο σε οποιουδήποτε μηχανικούς κραδασμούς του χώρου. Ελάχιστες μετακινήσεις πάνω στον πάγκο εργασίας αρκούν για να προκαλέσουν στιγμιαία απώλεια αγωγιμότητας. Αυτές ακριβώς οι απότομες διακοπές ρεύματος εκλαμβάνονται από τον κεντρικό επεξεργαστή ως πραγματικά γεγονότα διάρρηξης, δημιουργώντας ανεπιθύμητους ψευδείς συναγερμούς.

Σε αυστηρά ηλεκτρονικό επίπεδο παρατηρούνται σοβαρές αλλοιώσεις του σήματος. Οι παράλληλες μεταλλικές λωρίδες στο εσωτερικό της μήτρας δοκιμών λειτουργούν ακούσια ως μικροί πυκνωτές. Αναπτύσσονται επομένως παρασιτικές χωρητικότητες μεταξύ γειτονικών γραμμών μεταφοράς δεδομένων. Κατά τη διάρκεια απότομων αλλαγών δυναμικού μέρος της ενέργειας μεταφέρεται επαγωγικά στους παρακείμενους αγωγούς. Παρόμοια ηλεκτρικά φαινόμενα υποβαθμίζουν δραματικά την καθαρότητα των ψηφιακών εντολών. Ευτυχώς η προσεκτική αρχικοποίηση των εσωτερικών αντιστάσεων αποτρέπει την πλήρη κατάρρευση της προγραμματιστικής λογικής.

Ένας κρίσιμος αποτρεπτικός παράγοντας αφορά τη διαχείριση των φυσικών αποστάσεων. Μακριά τμήματα καλωδίωσης συμπεριφέρονται αναπόφευκτα ως κεραίες λήψης ραδιοκυμάτων. Συλλέγουν

διαρκώς ηλεκτρομαγνητικές παρεμβολές από τον περιβάλλοντα χώρο. Επειδή ολόκληρο το πρωτότυπο στερείται εξωτερικής μεταλλικής θωράκισης, η συγκεκριμένη αόρατη ακτινοβολία προστίθεται απευθείας στις γραμμές των ευαίσθητων αισθητήρων. Πρακτικά η απόσταση μεταξύ της πλακέτας Nucleo και των περιφερειακών μαγνητικών επαφών δεν επιτρέπεται να υπερβαίνει τα ελάχιστα εκατοστά. Για τη μελλοντική εγκατάσταση της μονάδας σε ένα πραγματικό κτίριο απαιτείται οπωσδήποτε η κατασκευή ειδικού τυπωμένου κυκλώματος, ικανού να οδηγήσει σωστά μεγάλες αποστάσεις μέσω κατάλληλων καλωδίων.

6.3 Προτάσεις για Μελλοντική Βελτίωση

Η τρέχουσα υλοποίηση προσφέρει μια στέρεη λειτουργική βάση, ωστόσο οι προοπτικές εξέλιξης της κατασκευής παραμένουν πολυάριθμες. Το πλέον κρίσιμο βήμα αφορά την οριστική μετάβαση από το προσωρινό περιβάλλον του breadboard σε ένα επαγγελματικά σχεδιασμένο τυπωμένο κύκλωμα. Αυτή η μηχανική αναβάθμιση θα εξαλείψει πλήρως τις ηλεκτρικές αστάθειες. Μια προσαρμοσμένη πλακέτα εγγυάται απόλυτη σταθερότητα των επαφών, ελαχιστοποιεί τον παρασιτικό θόρυβο και επιτρέπει τη δραστική χωροταξική συρρίκνωση της κεντρικής μονάδας. Ακόμη μια οθόνη αφής θα έδινε ένα πιο σύγχρονο τόνο στο σύστημα του συναγερμού.

Η διασύνδεση του εξοπλισμού με τον έξω κόσμο συνιστά μια αυτονόητη τεχνολογική απαίτηση. Προτείνεται η ενσωμάτωση κατάλληλων μονάδων ασύρματης επικοινωνίας, αξιοποιώντας δίκτυα κινητής τηλεφωνίας ή τοπικούς πομποδέκτες ασύρματου διαδικτύου. Με αυτόν τον τρόπο η διάταξη μετατρέπεται σε έναν ενεργό κόμβο του Διαδικτύου των Πραγμάτων. Ο τελικός χρήστης θα λαμβάνει ακαριαία ενημερωτικά μηνύματα στην προσωπική του έξυπνη συσκευή τη στιγμή ακριβώς του συμβάντος, διευρύνοντας τον έλεγχο ασφαλείας πέρα από τα στενά όρια της φυσικής του παρουσίας στον προστατευόμενο χώρο.

Το λογισμικό ελέγχου μπορεί να εμπλουτιστεί με έξυπνες αναλυτικές ρουτίνες, αντί να βασίζεται αποκλειστικά στην ψηφιακή κατάσταση των ακροδεκτών, το σύστημα θα μπορούσε να αξιοποιήσει καινοτόμους αλγόριθμους μηχανικής μάθησης. Τέτοια πολύπλοκα μαθηματικά μοντέλα κατανοούν τα μοτίβα συμπεριφοράς. Εκπαιδεύονται διαρκώς ώστε να διαχωρίζουν μια απλή περιβαλλοντική διακύμανση από μια πραγματική προσπάθεια διάρρηξης [2]. Η υιοθέτηση αυτής της στρατηγικής εξαλείφει τους ψευδείς συναγερμούς αναβαθμίζοντας κατακόρυφα τη συνολική αξιοπιστία της προτεινόμενης αρχιτεκτονικής.

ΒΙΒΛΙΟΓΡΑΦΙΑ

Βιβλία

- [1] C. Ünsalan, H. D. Gürhan, and M. E. Yücel, *Embedded System Design with ARM Cortex-M Microcontrollers*. Springer, 2022.
- [2] J. Wang, W. Hu, and T. Chen, *Intelligent Industrial Alarm Systems: Advanced Analysis and Design Methods*. Springer, 2024.

Data Sheet

- [3] ALLDATASHEET, HC-SR501 Datasheet
- [4] ALLDATASHEET, BISS0001 Datasheet
- [5] HandsOn Tech, I2C Interface 20x4 LCD Module User Guide & Datasheet
- [6] PARALLAX, 4x4 Matrix Membrane Keypad Documentation
- [7] STMicroelectronics, Nucleo-F401 Documentation UM1724 30 Sep 2025
- [8] STMicroelectronics, STM32CubeIDE Documentation DB3871 05 Mar 2026
- [9] STMicroelectronics, STM32F401 Documentation DS10086 17 Mar 2026

Internet Site

- [10] Microcontrollerslab, “STM32 Nucleo F401 RE Development Board” Available:
<https://microcontrollerslab.com/stm32-nucleo-board-pinout-features-applications-programming-peripherals/>
- [11] Piest Systems, “Top Differences Between Arduino VS STM32- Complete Comparison Guide” Available: <https://piestsystems.com/arduino-vs-stm32/>
- [12] CONTROLLERSTECH, “How to Use 4x4 keypad with STM32” Available:
<https://controllerstech.com/use-4x4-keypad-with-stm32/>
- [13] CONTROLLERSTECH, “LCD 20x4 using I2C with STM32” Available:
<https://controllerstech.com/lcd-20x4-using-i2c-with-stm32/>

Paper in Conference Proceedings

- [14] H. P B, S. R. Anireddy, J. F T and V. R, "Introduction to ARM processors & its types and Overview to Cortex M series with deep explanation of each of the processors in this Family," *2022 International Conference on Computer Communication and Informatics (ICCCI)*, Coimbatore, India, 2022, pp. 1-8. <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9740768&isnumber=9740653>
- [15] Z. Xiaodong and Z. Jie, "Design and implementation of smart home control system based on STM32," *2018 Chinese Control And Decision Conference (CCDC)*, Shenyang, China, 2018, pp. 3023-3027. <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8407643&isnumber=8407034>
- [16] Lele Niu, Zunan Ou, and Wencheng Wang. 2021. Design of Indoor Intelligent Alarm System Based on STM32. In *Proceedings of the 6th International Conference on Intelligent Information Processing (ICIIP '21)*. Association for Computing Machinery, New York, NY, USA, 100–103. <https://doi.org/10.1145/3480571.3480587>

Application Note

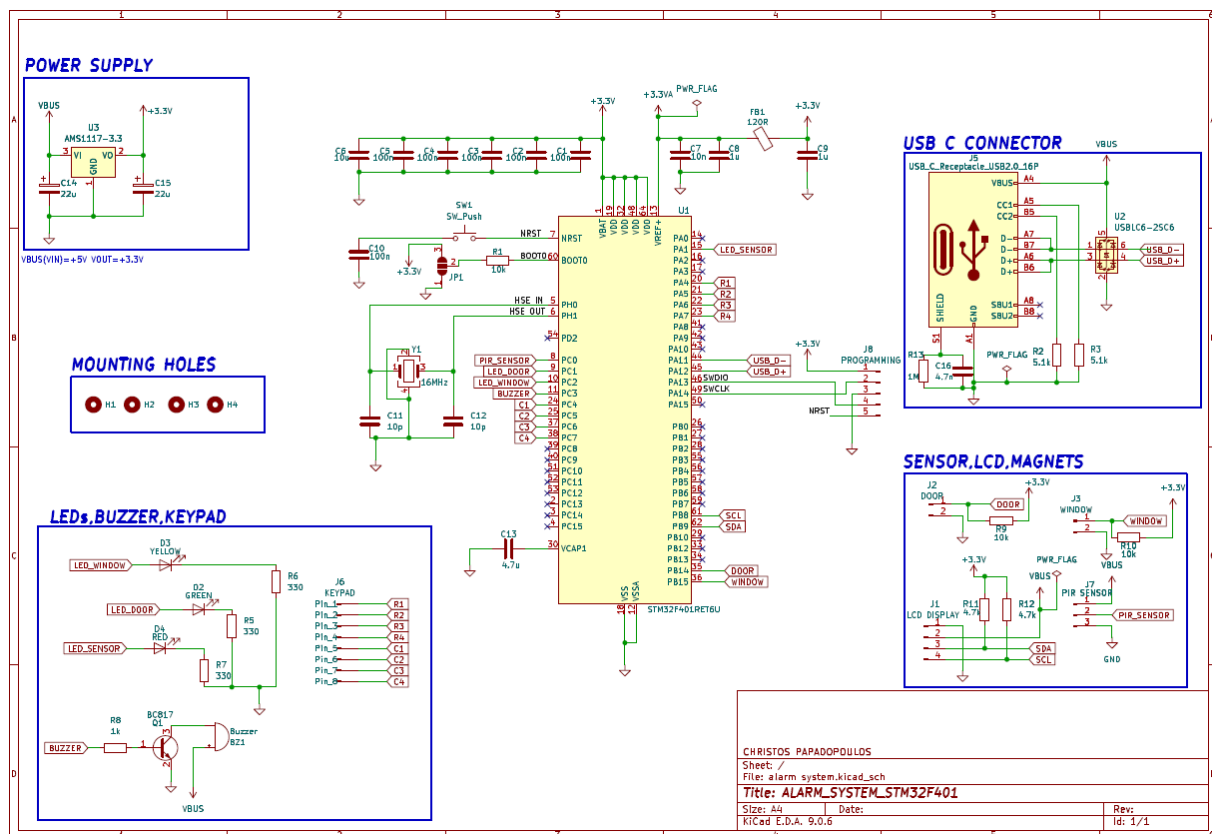
- [17] J. Wu, "A basic guide to I2C," Texas Instruments, 2022. <https://www.ti.com/lit/an/sbaa565/sbaa565.pdf>

ΠΑΡΑΡΤΗΜΑ Α: ΚΑΤΑΛΟΓΟΣ ΥΛΙΚΩΝ

Πίνακας 10: Κατάλογος-Κοστολόγηση Υλικών

ΠΡΟΙΟΝ	ΠΟΣΟΤΗΤΑ	ΤΙΜΗ
NUCLEO- F401RE	1	24,20€
LED	3	$0,10 \times 3 = 0,30€$
BUZZER	1	0,30€
KEYPAD	1	1,35€
PIR SENSOR	1	2,50€
LCD 20x4 I2C	1	7,00€
MAGNETIC REED SWITCHES	2	$3,25 \times 2 = 6,50€$
ΣΥΝΟΛΟ	10	42,15€

ΠΑΡΑΡΤΗΜΑ Β : ΣΧΗΜΑΤΙΚΟ ΑΠΟ KICAD



ΠΑΡΑΡΤΗΜΑ Γ : ΚΩΔΙΚΑΣ

```

#include "main.h"

#include "i2c-lcd.h"
#include <string.h>

I2C_HandleTypeDef hi2c1;

void SystemClock_Config(void);
static void MX_GPIO_Init(void);
static void MX_I2C1_Init(void);

char read_keypad(void)
{
    char keys[4][4] = {
        {'1', '2', '3', 'A'},
        {'4', '5', '6', 'B'},
        {'7', '8', '9', 'C'},
        {'*', '0', '#', 'D'}
    };

    // Τα Pins των γραμμών (PA4 έως PA7) και των στηλών (PC4 έως PC7)
    uint16_t row_pins[4] = {GPIO_PIN_4, GPIO_PIN_5, GPIO_PIN_6, GPIO_PIN_7};
    uint16_t col_pins[4] = {GPIO_PIN_4, GPIO_PIN_5, GPIO_PIN_6, GPIO_PIN_7};

    for (int r = 0; r < 4; r++)
    {
        // 1. Στέλνουμε 1 (HIGH) σε όλες τις γραμμές
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4 | GPIO_PIN_5 | GPIO_PIN_6 | GPIO_PIN_7, GPIO_PIN_SET);

        // 2. Στέλνουμε 0 (LOW) μόνο στη γραμμή που ελέγχουμε τώρα
        HAL_GPIO_WritePin(GPIOA, row_pins[r], GPIO_PIN_RESET);

        // Μικρή παύση για να προλάβει να πέσει η τάση
        HAL_Delay(5);

        // 3. Ελέγχουμε μία-μία τις στήλες
        for (int c = 0; c < 4; c++)
        {
            if (HAL_GPIO_ReadPin(GPIOC, col_pins[c]) == GPIO_PIN_RESET)

```

```

        if (HAL_GPIO_ReadPin(GPIOC, col_pins[c]) == GPIO_PIN_RESET)
        {
            // Το κουμπι πατήθηκε!
            // Αντί για το while που μπλοκάρει, βάζουμε μια μικρή παύση (debounce)
            HAL_Delay(250);
            return keys[r][c]; // Επιστρέφει το γράμμα/αριθμό που πάτησε
        }
    }
}

return '\0'; // /if nothing is pressed
}

int main(void)
{
    /* USER CODE BEGIN 1 */
    uint8_t system_state = 0; // Ξεκινάμε από την Κατάσταση 0 (Αφοπλισμένο)
    uint32_t timestamp = 0; // Εδώ θα αποθηκεύουμε την ώρα που ξεκινάει η αντίστροφη μέτρηση
    char entered_code[5] = {0}; // Εδώ θα αποθηκεύονται τα 4 ψηφία που πατάς (το 5ο είναι για τον τερματισμό)
    char current_password[5] = "1234";
    uint8_t password_change_mode = 0;
    uint8_t code_index = 0; // Μετράει πόσα ψηφία έχεις πατήσει μέχρι στιγμής
    uint8_t zone_A_selected = 0; // Θυμάται ότι επιλέξαμε τη Ζώνη A (PIR)
    uint8_t zone_B_selected = 0; // Για τη μαγνητική επαφή της πόρτας
    uint8_t zone_C_selected = 0; // Για την μαγνητική επαφή του παραθύρου
    uint8_t failed_attempts = 0; // Μετράει τους λάθος κωδικούς
    uint8_t full_arm_selected = 0;
    uint8_t alarm_source = 0;
    uint8_t alarm_source_2 = 0;
    uint8_t alarm_source_3 = 0;

    HAL_Init();

    SystemClock_Config();

```

```

MX_GPIO_Init();
MX_I2C1_Init();

lcd_init();

    lcd_send_cmd (0x80|0x07);
    lcd_send_string("HELLO");// first line

    lcd_send_cmd (0x80|0x43);
    lcd_send_string("ENTER PASSWORD");//second line

    lcd_send_cmd (0x80|0x1D);//third line
    lcd_send_string("OR");

    lcd_send_cmd (0x80|0x56);//fourth line
    lcd_send_string("PRESS THE BUTTON");

while (1)
{
    // 1. Διαβάζουμε το πληκτρολόγιο
    char key_pressed = read_keypad();
    // 2. ΛΟΓΙΚΗ ΕΙΣΑΓΩΓΗΣ ΚΩΔΙΚΟΥ
    if (key_pressed != '\0')
    {
        // --- ΚΟΜΜΑΤΙ Α: ΕΙΣΑΓΩΓΗ ΨΗΦΙΩΝ ---
        if (key_pressed >= '0' && key_pressed <= '9')
        {
            if (code_index == 0)
            {
                lcd_clear();
                // Διαχωρισμός οθόνης ανάλογα με το αν αλλάζουμε κωδικό ή όχι
                if (password_change_mode == 1) {
                    lcd_send_cmd(0x80 | 0x01); lcd_send_string("NEW PASSWORD:");
                } else {
                    lcd_send_cmd(0x80 | 0x05); lcd_send_string("Press key:");
                }
            }

            entered_code[code_index] = key_pressed;
            lcd_send_cmd((0x80 | 0x48)+ code_index);
            lcd_send_data(key_pressed);
            code_index++;
        }
    }
}

```

```

if(code_index == 4)
{
    entered_code[4] = '\0';

    // -- ΝΕΟ: ΑΝ ΕΙΜΑΣΤΕ ΣΕ ΛΕΙΤΟΥΡΓΙΑ ΑΛΛΑΓΗΣ ΚΩΔΙΚΟΥ --
    if (password_change_mode == 1)
    {
        strcpy(current_password, entered_code); // Αντιγράφει τον νέο στον τρέχοντα
        password_change_mode = 0;               // Απενεργοποίηση λειτουργίας αλλαγής

        lcd_clear();
        lcd_send_cmd(0x80 | 0x03);
        lcd_send_string("PASSWORD SAVED");
        HAL_Delay(1500);

        // Επιστροφή στην οθόνη προείρας
        lcd_clear();
        lcd_send_cmd(0x80 | 0x07); lcd_send_string("HELLO");
        lcd_send_cmd(0x80 | 0x43); lcd_send_string("ENTER PASSWORD");
        lcd_send_cmd(0x80 | 0x1D); lcd_send_string("OR");
        lcd_send_cmd(0x80 | 0x56); lcd_send_string("PRESS THE BUTTON");
    }
    // -- ΚΑΝΟΝΙΚΗ ΛΕΙΤΟΥΡΓΙΑ (ΟΠΛΙΣΜΟΣ / ΑΦΟΠΛΙΣΜΟΣ) --
    else
    {
        // ΠΛΕΟΝ ΕΛΕΓΧΕΙ ΤΗΝ current_password ΜΕΤΑΒΛΗΤΗ!
        if(strcmp(entered_code, current_password) == 0)
        {
            failed_attempts = 0;
            if(system_state == 0)
            {
                system_state = 1;
                timestamp = HAL_GetTick();
                full_arm_selected = 1;
                zone_A_selected = 0;
                zone_B_selected = 0;
                zone_C_selected = 0;
                lcd_clear();
                lcd_send_cmd(0x80 | 0x01);
                lcd_send_string("FIII I SYSTEM ARMING");
            }
        }
    }
}

```

```

    }
    else
    {
        system_state = 0;
        zone_A_selected =0; zone_B_selected =0; zone_C_selected =0; full_arm_selected =0;
        alarm_source = 0; alarm_source_2 =0; alarm_source_3 =0;
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOC, GPIO_PIN_3, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOC, LED_DOOR_Pin, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOC, LED_WINDOW_Pin, GPIO_PIN_RESET);

        lcd_clear();
        lcd_send_cmd(0x80 | 0x07); lcd_send_string("HELLO");
        lcd_send_cmd(0x80 | 0x43); lcd_send_string("ENTER PASSWORD");
        lcd_send_cmd(0x80 | 0x1D); lcd_send_string("OR");
        lcd_send_cmd(0x80 | 0x56); lcd_send_string("PRESS THE BUTTON");
    }
}
else // Λάθος κωδικός
{
    failed_attempts++;
    if(failed_attempts>=3)
    {
        system_state = 4; // ΣΥΝΑΓΕΡΜΟΣ!
        alarm_source = 3;
        lcd_clear();
        lcd_send_cmd(0x80 | 0x00); lcd_send_string("  SYSTEM LOCKED!  ");
        lcd_send_cmd(0x80 | 0x40); lcd_send_string("  ALARM ACTIVE!  ");
    }
    else
    {
        lcd_clear();
        lcd_send_cmd(0x80 | 0x05); lcd_send_string("WRONG CODE");
        lcd_send_cmd(0x80 | 0x43);
        if (failed_attempts == 1) lcd_send_string("2 TRIES LEFT");
        if (failed_attempts == 2) lcd_send_string("1 TRY LEFT");

        HAL_Delay(1500);
        lcd_clear();
        if (system_state == 0) {
            lcd_send_cmd(0x80 | 0x07); lcd_send_string("HELLO");
            lcd_send_cmd(0x80 | 0x43); lcd_send_string("ENTER PASSWORD");
            lcd_send_cmd(0x80 | 0x1D); lcd_send_string("OR");

```

```

        lcd_send_cmd(0x80 | 0x56); lcd_send_string("PRESS THE BUTTON");
    }
}
}
code_index = 0;
}
}
// --- ΚΟΜΜΑΤΙ Β: ΕΠΙΛΟΓΗ ΖΩΝΩΝ (Α, Β, C, *, #) ΟΤΑΝ ΕΙΝΑΙ ΑΦΟΠΛΙΣΜΕΝΟ ---
else if (system_state == 0)
{
    // ΝΕΟ: Έναρξη αλλαγής κωδικού με το '*'
    if (key_pressed == '*')
    {
        password_change_mode = 1;
        code_index = 0; // Μηδενίζουμε για να πάρει 4 νέα ψηφία
        lcd_clear();
        lcd_send_cmd(0x80 | 0x01);
        lcd_send_string("ENTER NEW PIN:");
    }
    else if (key_pressed == 'A')
    {
        if (zone_A_selected == 1) { zone_A_selected = 0; }
        else if (zone_B_selected == 0 || zone_C_selected == 0) { zone_A_selected = 1; }
    }
    else if (key_pressed == 'B')
    {
        if (zone_B_selected == 1) { zone_B_selected = 0; }
        else if (zone_A_selected == 0 || zone_C_selected == 0) { zone_B_selected = 1; }
    }
    else if (key_pressed == 'C')
    {
        if (zone_C_selected == 1) { zone_C_selected = 0; }
        else if (zone_A_selected == 0 || zone_B_selected == 0) { zone_C_selected = 1; }
    }
}

if(key_pressed == 'A' || key_pressed == 'B' || key_pressed == 'C')
{
    lcd_clear();
    lcd_send_cmd(0x80 | 0x01);
    if (zone_A_selected && zone_B_selected) { lcd_send_string("ZONE A+B SELECTED "); }
    else if (zone_A_selected && zone_C_selected) { lcd_send_string("ZONE A+C SELECTED "); }
    else if (zone_B_selected && zone_C_selected) { lcd_send_string("ZONE B+C SELECTED "); }
}

```

```

        else if (zone_A_selected)          { lcd_send_string("ZONE A SELECTED "); }
        else if (zone_B_selected)          { lcd_send_string("ZONE B SELECTED "); }
        else if (zone_C_selected)          { lcd_send_string("ZONE C SELECTED "); }
        else                               { lcd_send_string("NO ZONE SELECTED "); }

        if (zone_A_selected || zone_B_selected || zone_C_selected)
        {
            lcd_send_cmd(0x80 | 0x41);
            lcd_send_string("PRESS # TO ARM");
        }
    }
    else if (key_pressed == '#')
    {
        if (zone_A_selected || zone_B_selected || zone_C_selected)
        {
            system_state = 1;
            timestamp = HAL_GetTick();
            lcd_clear();
            lcd_send_cmd(0x80 | 0x03);
            lcd_send_string("SYSTEM ARMING...");
        }
    }
}
}
}

```

// 2. Η ΜΗΧΑΝΗ ΚΑΤΑΣΤΑΣΕΩΝ ΤΟΥ ΣΥΝΑΓΕΡΜΟΥ

```

switch (system_state)
{
    case 0:// ΑΦΟΠΛΙΣΜΕΝΟ (Idle)
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOC, GPIO_PIN_3, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOC, LED_DOOR_Pin, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOC, LED_WINDOW_Pin, GPIO_PIN_RESET);
        break;
}

```

```

case 1:// ΧΡΟΝΟΣ ΕΞΟΔΟΥ
if (HAL_GetTick() - timestamp >= 15000)
{
    if(full_arm_selected == 1)
    {
        lcd_clear();
        lcd_send_cmd(0x80 | 0x01);
        lcd_send_string("FULL SYSTEM ARMED!");
        HAL_Delay(1500);
    }
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_3, GPIO_PIN_SET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_SET);
    HAL_Delay(500);
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_3, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_RESET);
    lcd_clear();
    system_state = 2;
}
break;

case 2:// ΠΛΗΡΩΣ ΟΠΛΙΣΜΕΝΟ
// ΕΛΕΓΧΟΣ 1: PIR
if ((zone_A_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOC, GPIO_PIN_0) == GPIO_PIN_SET)
{
    alarm_source = 1; // ΚΛΕΙΔΩΝΕΙ ΑΜΕΣΩΣ ΩΣ 1η ΠΑΡΑΒΙΑΣΗ
    timestamp = HAL_GetTick();
    lcd_clear();
    lcd_send_cmd(0x80 | 0x02);
    lcd_send_string("Motion Detected!");
    system_state = 3;
}
// ΕΛΕΓΧΟΣ 2: ΠΟΡΤΑ
else if ((zone_B_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOB, DOOR_Pin) == GPIO_PIN_RESET)
{
    alarm_source = 2;
    system_state = 4;
}
// ΕΛΕΓΧΟΣ 3: ΠΑΡΑΘΥΡΟ
else if ((zone_C_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOB, WINDOW_Pin) == GPIO_PIN_RESET)
{
    alarm_source = 4;
    system_state = 4;
}
}

```

```

    }
    break;

case 3: // ΧΡΟΝΟΣ ΕΙΣΟΔΟΥ (Από PIR - Το alarm_source είναι ήδη 1)
    if ((zone_B_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOB, DOOR_Pin) == GPIO_PIN_RESET)
    {
        alarm_source_2 = 2; // Καταγράφεται ως 2η παραβίαση
        system_state = 4;
    }
    else if ((zone_C_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOB, WINDOW_Pin) == GPIO_PIN_RESET)
    {
        alarm_source_2 = 4; // Καταγράφεται ως 2η παραβίαση
        system_state = 4;
    }
    else if (HAL_GetTick() - timestamp >= 15000)
    {
        system_state = 4;
    }
    break;

case 4: // ΣΥΝΑΓΕΡΜΟΣ! (Latching με Αυτόματο Κλείδωμα)
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_3, GPIO_PIN_SET); // Buzzer ON

    // --- 1. LIVE ΕΛΕΓΧΟΣ ΓΙΑ 2η ΠΑΡΑΒΙΑΣΗ (Μόνο αν η 2η θέση είναι κενά) ---
    if (alarm_source_2 == 0)
    {
        if (alarm_source != 1 && (zone_A_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOC, GPIO_PIN_0) == GPIO_PIN_SET)
            alarm_source_2 = 1;
        else if (alarm_source != 2 && (zone_B_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOB, DOOR_Pin) == GPIO_PIN_RESET)
            alarm_source_2 = 2;
        else if (alarm_source != 4 && (zone_C_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOB, WINDOW_Pin) == GPIO_PIN_RESET)
            alarm_source_2 = 4;
    }

    // --- 2. LIVE ΕΛΕΓΧΟΣ ΓΙΑ 3η ΠΑΡΑΒΙΑΣΗ (Μόνο αν η 2η έχει ενεργήσει και η 3η είναι κενά) ---
    if (alarm_source_2 != 0 && alarm_source_3 == 0)
    {
        if (alarm_source != 1 && alarm_source_2 != 1 && (zone_A_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOC, GPIO_PIN_0) == GPIO_PIN_SET)
            alarm_source_3 = 1;
        else if (alarm_source != 2 && alarm_source_2 != 2 && (zone_B_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOB, DOOR_Pin) == GPIO_PIN_RESET)
            alarm_source_3 = 2;
        else if (alarm_source != 4 && alarm_source_2 != 4 && (zone_C_selected == 1 || full_arm_selected == 1) && HAL_GPIO_ReadPin(GPIOB, WINDOW_Pin) == GPIO_PIN_RESET)
            alarm_source_3 = 4;
    }

```

```

}

// --- 3. ΑΝΑΒΟΥΜΕ ΤΑ ΑΝΤΙΣΤΟΙΧΑ LEDs ---
if (alarm_source == 1 || alarm_source_2 == 1 || alarm_source_3 == 1) HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_SET);
if (alarm_source == 2 || alarm_source_2 == 2 || alarm_source_3 == 2) HAL_GPIO_WritePin(GPIOC, LED_DOOR_Pin, GPIO_PIN_SET);
if (alarm_source == 4 || alarm_source_2 == 4 || alarm_source_3 == 4) HAL_GPIO_WritePin(GPIOC, LED_WINDOW_Pin, GPIO_PIN_SET);

// --- 4. ΕΚΤΥΠΩΣΗ ΣΤΗΝ ΟΘΟΝΗ ---
if (code_index == 0)
{
    // --- ΣΕΝΑΡΙΟ ΜΠΛΟΚΑΡΙΣΜΑΤΟΣ ---
    if (alarm_source == 3)
    {
        lcd_send_cmd(0x80 | 0x00); lcd_send_string("  SYSTEM LOCKED!  ");
        lcd_send_cmd(0x80 | 0x40); lcd_send_string("  ALARM ACTIVE!  ");
        lcd_send_cmd(0x80 | 0x14); lcd_send_string("                ");
        lcd_send_cmd(0x80 | 0x54); lcd_send_string("                ");
    }
    // --- ΣΕΝΑΡΙΟ Α: Μόνο 1 Παραβίαση ---
    else if (alarm_source_2 == 0)
    {
        lcd_send_cmd(0x80 | 0x00);
        if (alarm_source == 1)    lcd_send_string("MOTION DETECTED!  ");
        else if (alarm_source == 2) lcd_send_string("DOOR BREACHED!  ");
        else if (alarm_source == 4) lcd_send_string("WINDOW BREACHED!  ");

        lcd_send_cmd(0x80 | 0x40); lcd_send_string("  ALARM ACTIVE!  ");
        lcd_send_cmd(0x80 | 0x14); lcd_send_string("                ");
        lcd_send_cmd(0x80 | 0x54); lcd_send_string("                ");
    }
    // --- ΣΕΝΑΡΙΟ Β: 2 Παραβιάσεις ---
    else if (alarm_source_3 == 0)
    {
        lcd_send_cmd(0x80 | 0x00);
        if (alarm_source == 1)    lcd_send_string("1.MOTION DETECTED!  ");
        else if (alarm_source == 2) lcd_send_string("1.DOOR BREACHED!  ");
        else if (alarm_source == 4) lcd_send_string("1.WINDOW BREACHED!  ");

        lcd_send_cmd(0x80 | 0x40);
        if (alarm_source_2 == 1)    lcd_send_string("2.MOTION DETECTED!  ");
        else if (alarm_source_2 == 2) lcd_send_string("2.DOOR BREACHED!  ");
        else if (alarm_source_2 == 4) lcd_send_string("2.WINDOW BREACHED!  ");
    }
}

```

```

        lcd_send_cmd(0x80 | 0x14); lcd_send_string("  ALARM ACTIVE!  ");
        lcd_send_cmd(0x80 | 0x54); lcd_send_string("                ");
    }
    // --- ΣΕΝΑΡΙΟ Γ: 3 Παραβιάσεις ---
    else
    {
        lcd_send_cmd(0x80 | 0x00);
        if (alarm_source == 1)    lcd_send_string("1.MOTION DETECTED! ");
        else if (alarm_source == 2) lcd_send_string("1.DOOR BREACHED!  ");
        else if (alarm_source == 4) lcd_send_string("1.WINDOW BREACHED! ");

        lcd_send_cmd(0x80 | 0x40);
        if (alarm_source_2 == 1)    lcd_send_string("2.MOTION DETECTED! ");
        else if (alarm_source_2 == 2) lcd_send_string("2.DOOR BREACHED!  ");
        else if (alarm_source_2 == 4) lcd_send_string("2.WINDOW BREACHED! ");

        lcd_send_cmd(0x80 | 0x14);
        if (alarm_source_3 == 1)    lcd_send_string("3.MOTION DETECTED! ");
        else if (alarm_source_3 == 2) lcd_send_string("3.DOOR BREACHED!  ");
        else if (alarm_source_3 == 4) lcd_send_string("3.WINDOW BREACHED! ");

        lcd_send_cmd(0x80 | 0x54); lcd_send_string("  ALARM ACTIVE!  ");
    }
}
break;
}

```

```

void SystemClock_Config(void)
{
    RCC_OscInitTypeDef RCC_OscInitStruct = {0};
    RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};

    __HAL_RCC_PWR_CLK_ENABLE();
    __HAL_PWR_VOLTAGESCALING_CONFIG(PWR_REGULATOR_VOLTAGE_SCALE2);

    RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSI;
    RCC_OscInitStruct.HSISState = RCC_HSI_ON;
    RCC_OscInitStruct.HSICalibrationValue = RCC_HSICALIBRATION_DEFAULT;
    RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;
    RCC_OscInitStruct.PLL.PLLSource = RCC_PLLSOURCE_HSI;
    RCC_OscInitStruct.PLL.PLLM = 16;
    RCC_OscInitStruct.PLL.PLLN = 336;
    RCC_OscInitStruct.PLL.PLLP = RCC_PLLP_DIV4;
    RCC_OscInitStruct.PLL.PLLQ = 7;
    if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)
    {
        Error_Handler();
    }

    RCC_ClkInitStruct.ClockType = RCC_CLOCKTYPE_HCLK|RCC_CLOCKTYPE_SYSCLK
                                  |RCC_CLOCKTYPE_PCLK1|RCC_CLOCKTYPE_PCLK2;
    RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_PLLCLK;
    RCC_ClkInitStruct.AHBCLKDivider = RCC_SYSCLK_DIV1;
    RCC_ClkInitStruct.APB1CLKDivider = RCC_HCLK_DIV2;
    RCC_ClkInitStruct.APB2CLKDivider = RCC_HCLK_DIV1;

    if (HAL_RCC_ClockConfig(&RCC_ClkInitStruct, FLASH_LATENCY_2) != HAL_OK)
    {
        Error_Handler();
    }
}

static void MX_I2C1_Init(void)
{

```

```

hi2c1.Instance = I2C1;
hi2c1.Init.ClockSpeed = 100000;
hi2c1.Init.DutyCycle = I2C_DUTYCYCLE_2;
hi2c1.Init.OwnAddress1 = 0;
hi2c1.Init.AddressingMode = I2C_ADDRESSINGMODE_7BIT;
hi2c1.Init.DualAddressMode = I2C_DUALADDRESS_DISABLE;
hi2c1.Init.OwnAddress2 = 0;
hi2c1.Init.GeneralCallMode = I2C_GENERALCALL_DISABLE;
hi2c1.Init.NoStretchMode = I2C_NOSTRETCH_DISABLE;
if (HAL_I2C_Init(&hi2c1) != HAL_OK)
{
    Error_Handler();
}
}

static void MX_GPIO_Init(void)
{
    GPIO_InitTypeDef GPIO_InitStruct = {0};

    __HAL_RCC_GPIOC_CLK_ENABLE();
    __HAL_RCC_GPIOH_CLK_ENABLE();
    __HAL_RCC_GPIOA_CLK_ENABLE();
    __HAL_RCC_GPIOB_CLK_ENABLE();

    HAL_GPIO_WritePin(GPIOC, LED_DOOR_Pin|LED_WINDOW_Pin|BUZZER_Pin, GPIO_PIN_RESET);

    HAL_GPIO_WritePin(GPIOA, STATUS_LED_SENSOR_Pin|R1_Pin|R2_Pin|R3_Pin
|R4_Pin, GPIO_PIN_RESET);

    GPIO_InitStruct.Pin = PIR_SENSOR_Pin;
    GPIO_InitStruct.Mode = GPIO_MODE_INPUT;
    GPIO_InitStruct.Pull = GPIO_PULLDOWN;
    HAL_GPIO_Init(PIR_SENSOR_GPIO_Port, &GPIO_InitStruct);

    GPIO_InitStruct.Pin = LED_DOOR_Pin|LED_WINDOW_Pin|BUZZER_Pin;

```

```

GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
HAL_GPIO_Init(GPIOC, &GPIO_InitStruct);

GPIO_InitStruct.Pin = STATUS_LED_SENSOR_Pin|R1_Pin|R2_Pin|R3_Pin
                    |R4_Pin;
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);

GPIO_InitStruct.Pin = C1_Pin|C2_Pin|C3_Pin|C4_Pin;
GPIO_InitStruct.Mode = GPIO_MODE_INPUT;
GPIO_InitStruct.Pull = GPIO_PULLUP;
HAL_GPIO_Init(GPIOC, &GPIO_InitStruct);

GPIO_InitStruct.Pin = WINDOW_Pin|DOOR_Pin;
GPIO_InitStruct.Mode = GPIO_MODE_INPUT;
GPIO_InitStruct.Pull = GPIO_PULLUP;
HAL_GPIO_Init(GPIOB, &GPIO_InitStruct);
}

void Error_Handler(void)
{
    __disable_irq();
    while (1)
    {
    }
}

#ifdef USE_FULL_ASSERT

void assert_failed(uint8_t *file, uint32_t line)

#endif /* USE_FULL_ASSERT */

```

ΠΑΡΑΡΤΗΜΑ Δ : STM32CUBEMX

