



ΔΗΜΟΚΡΙΤΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΡΑΚΗΣ

DEMOCRITUS
UNIVERSITY
OF THRACE

ΣΥΣΤΗΜΑ ΑΝΙΧΝΕΥΣΗΣ ΦΩΤΙΑΣ ΜΕ ΟΠΤΙΚΟΠΟΙΗΣΗ ΔΕΔΟΜΕΝΩΝ ΣΤΟ ΔΙΑΔΙΚΤΥΟ

Ομάδα 3

ΜΕΛΗ ΟΜΑΔΑΣ

Μάριος Μανουσάκης

Στυλιανός Γαλεγαλίδης

Στέλλα Ειρήνη Εγγλεξή

Χρυσούλα Αθανασία Δαλδαρα

Επιβλέπων Καθηγητής:

Δρ. Λάγκας Θωμάς

ΚΑΒΑΛΑ

ΜΑΪΟΣ 2025

Περιεχόμενα

Περίληψη.....	2
Abstract.....	3
Εισαγωγή.....	4
1. Απαιτήσεις – Προδιαγραφές.....	4
1.1 Διάγραμμα περιβάλλοντος.....	5
1.2 Λειτουργικές απαιτήσεις.....	5
1.3 Προδιαγραφές.....	6
2. Σχεδιασμός.....	8
2.1 Διάγραμμα Ροής Δεδομένων 1 ^{ου} επιπέδου.....	9
3. Υλοποίηση.....	9
4. Συμπεράσματα.....	20
Βιβλιογραφία.....	21
Παράρτημα 1 Documentation.....	21

Περίληψη

Αυτό το έργο πραγματεύεται τη δημιουργία ενός προηγμένου συστήματος πυρανίχνευσης για έξυπνους χώρους, βασισμένου στην ευέλικτη πλατφόρμα ESP32 C6 και την ενσωμάτωση πολλαπλών αισθητήρων για την ανίχνευση καπνού και θερμοκρασίας. Η λειτουργία του συστήματος εστιάζει στην συνεχή παρακολούθηση του περιβάλλοντος για την αναγνώριση δυνητικά επικίνδυνων συγκεντρώσεων καπνού ή αιφνίδιων αυξήσεων της θερμοκρασίας, σηματοδοτώντας καταστάσεις έκτακτης ανάγκης. Επιπρόσθετα, παρέχεται η δυνατότητα χειροκίνητης ενεργοποίησης του συναγερμού μέσω ενός φυσικού κουμπιού, προσφέροντας ένα επιπλέον επίπεδο ασφάλειας.

Σε περίπτωση που ανιχνευθεί αυξημένος κίνδυνος πυρκαγιάς, το σύστημα αντιδρά άμεσα μέσω ενός συνδυασμού ηχητικών και οπτικών ειδοποιήσεων – ενεργοποιώντας έναν διαπεραστικό ηχητικό συναγερμό και μια φωτεινή ένδειξη LED για την άμεση προειδοποίηση των παρευρισκόμενων. Παράλληλα, αξιοποιείται η συνδεσιμότητα IoT για την αποστολή ειδοποιήσεων σε πραγματικό χρόνο στην εφαρμογή Blynk, επιτρέποντας την ενημέρωση των χρηστών ανεξαρτήτως της τοποθεσίας τους. Η ενσωματωμένη οθόνη LCD αναλαμβάνει την εμφάνιση κρίσιμων μηνυμάτων και πληροφοριών σχετικά με την κατάσταση του συστήματος και την πιθανή απειλή.

Ο απώτερος σκοπός αυτού του έργου είναι η δημιουργία ενός αξιόπιστου και αποτελεσματικού συστήματος έγκαιρης ανίχνευσης πυρκαγιάς σε οικιακούς ή μικρούς επαγγελματικούς χώρους, συμβάλλοντας στην προστασία της ανθρώπινης ζωής και της υλικής περιουσίας μέσω της άμεσης ενημέρωσης και της δυνατότητας έγκαιρης αντίδρασης.

Λέξεις – Κλειδιά: ESP32 C6, Ευφυές σύστημα πυρανίχνευσης IoT, Ανίχνευση πολλαπλών παραμέτρων, Ειδοποιήσεις σε πραγματικό χρόνο, Διασύνδεση Blynk, Άμεση ανταπόκριση.

Abstract

This project deals with the creation of an advanced fire detection system for smart spaces, based on the flexible ESP32 C6 platform and the integration of multiple sensors for smoke and temperature detection. The operation of the system focuses on continuous monitoring of the environment to identify potentially dangerous concentrations of smoke or sudden temperature increases, signalling emergency situations. In addition, the ability to manually activate the alarm via a physical button is provided, offering an additional level of safety.

Should an increased fire risk be detected, the system reacts immediately through a combination of audible and visual alerts - activating a piercing audible alarm and an LED light to immediately warn bystanders. At the same time, IoT connectivity is leveraged to send real-time alerts to the Blynk app, allowing users to be notified regardless of their location. The integrated LCD screen takes care of displaying critical messages and information about the system status and potential threat.

The ultimate goal of this project is the creation of a reliable and effective early fire detection system in residential or small business premises, contributing to the protection of human life and property through immediate information and the possibility of early reaction.

Keywords: ESP32 C6, IOT Intelligent Fire Detection System, Multi-parameter detection, Real-time alerts, Blynk interface, Immediate response.

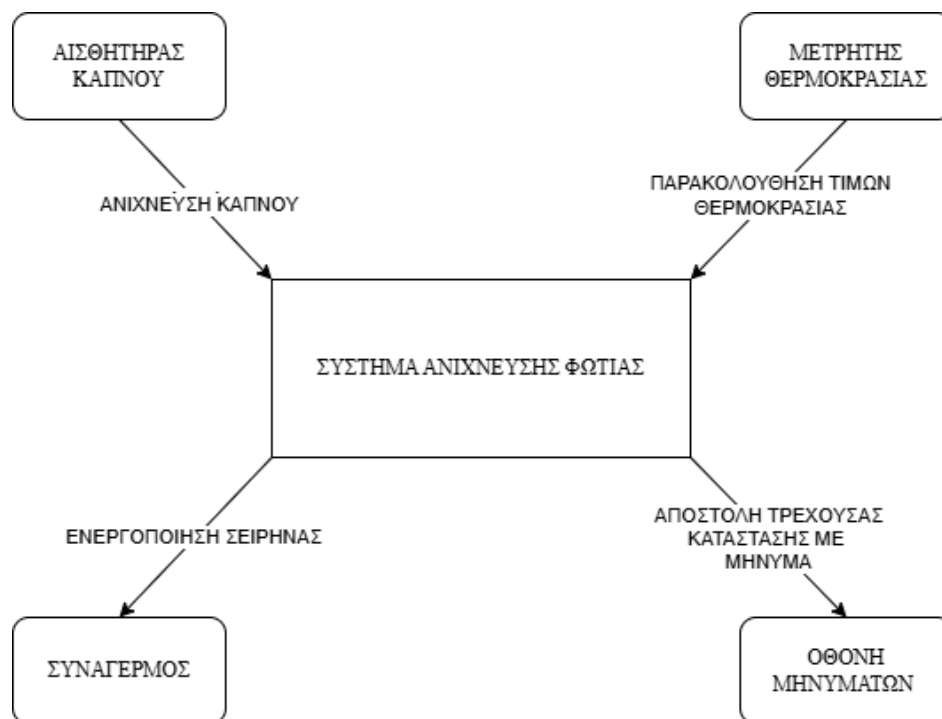
Εισαγωγή

Η πρόληψη και η έγκαιρη ανίχνευση πυρκαγιών αποτελούν κρίσιμους παράγοντες για την προστασία ανθρώπινων ζώων και περιουσιών. Αντίθετα με τα παραδοσιακά συστήματα ανίχνευσης που βασίζονται σε απλούς αισθητήρες καπνού ή θερμοκρασίας και συχνά παρουσιάζουν καθυστερήσεις ή ψευδείς συναγερμούς, η πρόοδος της τεχνολογίας IoT επιτρέπει την ανάπτυξη πιο έξυπνων και αποδοτικών λύσεων. Το παρόν έργο επικεντρώνεται στην ανάπτυξη ενός τέτοιου συστήματος ανίχνευσης πυρκαγιάς, το οποίο βασίζεται στο ESP32-C6 και αξιοποιεί αισθητήρες θερμοκρασίας και καπνού, μια οθόνη LCD για τοπική απεικόνιση και την πλατφόρμα Blynk για απομακρυσμένη παρακολούθηση, με στόχο την παροχή έγκαιρων ειδοποιήσεων σε περίπτωση ανίχνευσης επικίνδυνων συνθηκών και τη βελτίωση της απόκρισης σε περιστατικά πυρκαγιάς. Η επιλογή του θέματος υπαγορεύτηκε από την ανάγκη για προσιτά και αποτελεσματικά συστήματα ανίχνευσης πυρκαγιάς, ιδίως σε περιοχές με περιορισμένους πόρους, όπου η χρήση του ESP32-C6 προσφέρει πλεονεκτήματα όπως χαμηλό κόστος, ενσωματωμένη συνδεσιμότητα Wi-Fi, καθώς και ικανοποιητική υπολογιστική ισχύ. Οι κύριοι στόχοι της εργασίας είναι η σχεδίαση και υλοποίηση ενός συστήματος ανίχνευσης πυρκαγιάς με χρήση του ESP32-C6, η ενσωμάτωση αισθητήρων θερμοκρασίας και καπνού για την παρακολούθηση του περιβάλλοντος, η ανάπτυξη διεπαφής χρήστη μέσω οθόνης LCD και της εφαρμογής Blynk για την απεικόνιση δεδομένων και την αποστολή ειδοποιήσεων, καθώς και η αξιολόγηση της απόδοσης του συστήματος σε συνθήκες προσομοίωσης. Η δομή της εργασίας περιλαμβάνει την ανασκόπηση της σχετικής βιβλιογραφίας, την περιγραφή των χρησιμοποιούμενων υλικών και μεθόδων, την παρουσίαση των αποτελεσμάτων και τη συζήτηση επί των ευρημάτων, καταλήγοντας σε συμπεράσματα.

1. Απαιτήσεις – Προδιαγραφές

Σε αυτό το κεφάλαιο παρουσιάζονται οι αρχικές απαιτήσεις και οι προδιαγραφές του συστήματος. Αρχικά, στο διάγραμμα περιβάλλοντος (1.1) αποτυπώνεται το σύστημα σε σχέση με τις εξωτερικές του οντότητες (όπως χρήστες ή άλλες συσκευές), περιγράφοντας τις μεταξύ τους αλληλεπιδράσεις. Έπειτα, οι λειτουργικές απαιτήσεις (1.2) καταγράφουν τις βασικές λειτουργίες που πρέπει να υποστηρίζονται από το σύστημα. Τέλος, στην ενότητα των προδιαγραφών (1.3) αναφέρονται τα τεχνικά χαρακτηριστικά, οι περιορισμοί και οι παραδοχές, καθώς και πληροφορίες για τις τεχνολογίες που θα χρησιμοποιηθούν.

1.1 Διάγραμμα περιβάλλοντος



1.2 Λειτουργικές απαιτήσεις

Στο συγκεκριμένο σύστημα οι λειτουργικές απαιτήσεις που το υποστηρίζουν αποτελούνται από:

- ❖ Ανίχνευση καπνού: το σύστημα μέσω αισθητήρων πρέπει να συλλέγει τα δεδομένα που του δίνουν οι αισθητήρες
- ❖ Μέτρηση θερμοκρασίας: πάλι μέσω των αισθητήρων το σύστημα οφείλει να λαμβάνει τις μετρήσεις της θερμοκρασίας ανά τακτικά διαστήματα
- ❖ Ανάλυση και Αξιολόγηση δεδομένων: το σύστημα πρέπει να επεξεργαστεί τα στοιχεία που σύλλεξε και να τα αξιολογήσει, ως προς τα όρια που έχουν τεθεί από τον χρήστη.
- ❖ Αποστολή / Εμφάνιση στοιχείων στον χρήστη: το σύστημα χρησιμοποιώντας την οθόνη που διαθέτει, εμφανίζει όλα τα δεδομένα και τις μετρήσεις που εκτελεί, ώστε να γίνονται ορατά από τον χρήστη.
- ❖ Ενεργοποίηση συστήματος: σε περίπτωση φωτιάς/ πυρκαγιάς το σύστημα ενεργοποιείται και αμέσως ξεκινά να ηχεί ο βομβητής και να αναβοσβήνει το LED, ενώ παράλληλα αποστέλλει το ανάλογο μήνυμα μέσω της οθόνης.

1.3 Προδιαγραφές

Τεχνικά Χαρακτηριστικά

- ❖ Αισθητήρας καπνού (MQ2 gas sensor) για ανίχνευση καπνού
- ❖ Αισθητήρας θερμοκρασίας NTC(temper sensor) για μέτρηση θερμοκρασίας
- ❖ Κουμπί (Button) για χειροκίνητη ενεργοποίηση συστήματος
- ❖ Οθόνη (LCD2004) για εμφάνιση δεδομένων και μηνυμάτων
- ❖ Φωτάκι (LED) φωτεινή ειδοποίηση για ενεργοποίηση συστήματος
- ❖ Βομβητής (Buzzer) ηχητική ειδοποίηση για ενεργοποίηση συστήματος
- ❖ Μικροελεγκτής (ESP32)
- ❖ Επικοινωνία (WiFi)

Περιορισμοί

Το σύστημα λόγω μιας συνάρτησης του (millis()), απαιτεί επανεκκίνηση κάθε 49 μέρες για να λειτουργεί στο 100% της αποδοτικότητας του. Συγκεκριμένα, η συνάρτηση αυτή επανέρχεται στο 0 και δεν επιτρέπει στον τακτικό έλεγχο για ραγδαίες ανόδους της θερμοκρασίας και της πυκνότητας καπνού. Αυτό μπορεί να αποφέρει και θετικά αποτελέσματα, καθώς αναγκάζει τον χρήστη για τακτική συντήρηση αλλά καθιστά το σύστημα ακατάλληλο για δύσβατες και απομακρυσμένες περιοχές.

Τεχνολογίες που χρησιμοποιούνται

<u>Τμήμα Συστήματος</u>	<u>Τεχνολογία</u>
Μικροελεγκτής	ESP32
Αισθητήρας καπνού	MQ2 gas sensor
Αισθητήρας θερμοκρασίας	Temperature sensor pin
Οθόνη	ICD2004 12C 20x4
Φωτάκι	LED pin
Βομβητής	SOUND pin
Επικοινωνία	WiFi

Προδιαγραφές

Υπηρεσία	Μέτρηση επιπέδων θερμοκρασίας και τυχόν ανίχνευση καπνού
Περιγραφή	Όσο οι μετρήσεις της θερμοκρασίας είναι εντός ορίων και δεν ανιχνεύεται καπνός το σύστημα μένει ανενεργό
Δεδομένα Εισόδου	Τιμές επιπέδων καπνού και θερμοκρασίας
Προέλευση	Αισθητήρες καπνού και μέτρησης θερμοκρασίας
Δεδομένα Εξόδου	Εμφάνιση δεδομένων στην οθόνη μηνυμάτων
Προορισμός	Κεντρικό χρήστη
Ενέργεια	Αποστολή τρέχουσας κατάστασης με μήνυμα
Απαίτηση	Καμία
Προϋπόθεση	Οι μετρήσεις των αισθητήρων να παραμένουν στα καθορισμένα όρια
Αποτέλεσμα	Το σύστημα παραμένει ανενεργό
Πλευρικά Φαινόμενα	Κανένα

Υπηρεσία	Ενεργοποίηση συναγερμού ανίχνευσης φωτιάς
Περιγραφή	Όταν τα επίπεδα θερμοκρασίας και καπνού περάσουν τα όρια που έχουν ορισθεί το σύστημα ενεργοποιείται
Δεδομένα Εισόδου	Τιμή θερμοκρασίας, επίπεδα καπνού
Προέλευση	Αισθητήρες θερμοκρασίας και καπνού
Δεδομένα Εξόδου	Ηχητικό σήμα και κόκκινη ένδειξη
Προορισμός	Κεντρικό χρήστη μέσω της οθόνης μηνυμάτων
Ενέργεια	Αξιολόγηση των μετρήσεων και ενεργοποίηση συναγερμού

Απαίτηση	Καμία
Προϋπόθεση	Έχει πιάσει φωτιά και οι αισθητήρες έχουν εισάγει στο σύστημα τιμές άνω των καθορισμένων ορίων
Αποτέλεσμα	Αρχίζει να ηχεί ο συναγερμός και να αναβοσβήνει το κόκκινο λαμπάκι
Πλευρικά Φαινόμενα	Κανένα

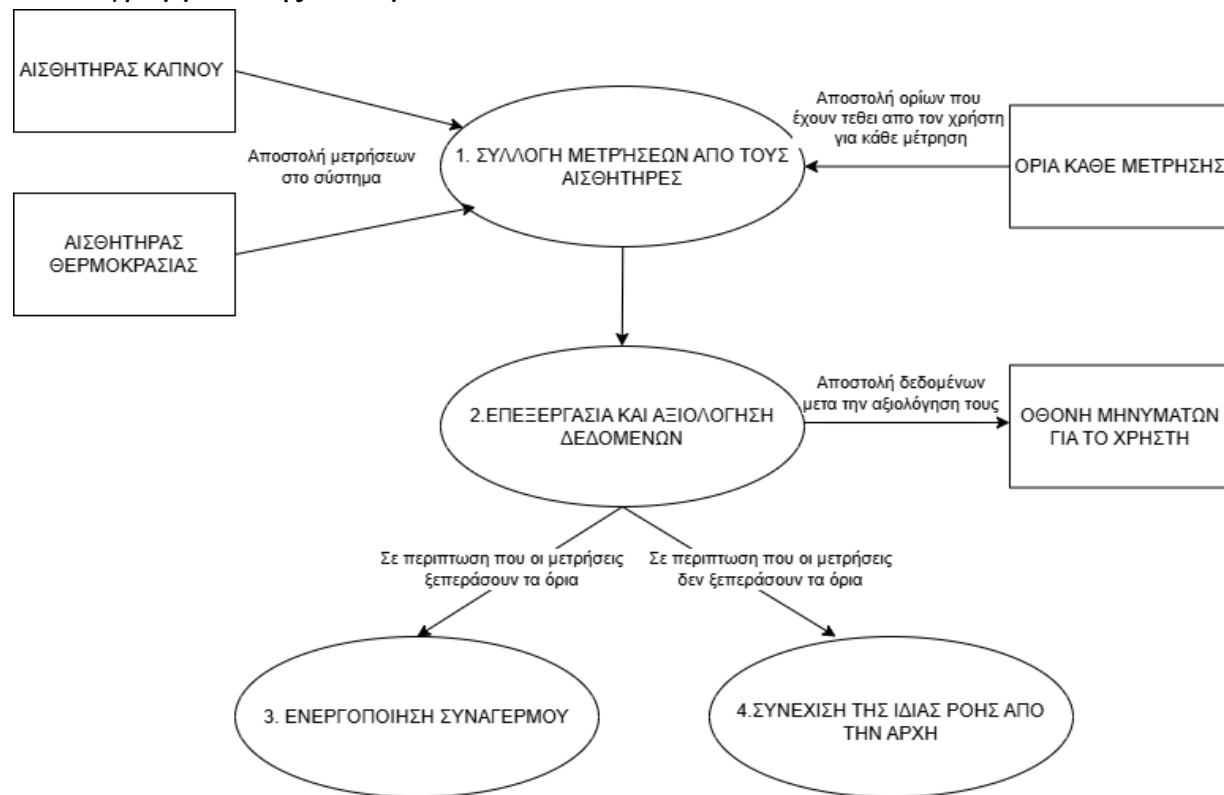
2. Σχεδιασμός

Στον σχεδιασμό του συστήματος IoT που παρουσιάζεται χρησιμοποιήθηκαν αισθητήρες, κουμπί, φωτάκι LED, μικροελεγκτής, βομβητής και οθόνη. Τα παραπάνω στοιχεία με κατάλληλο σχεδιασμό και δομή έφεραν αυτό το αποτέλεσμα.

Όπως φαίνεται και στο διάγραμμα ροής δεδομένων παρακάτω, η διαδικασία εκκίνησής του, αρχίζει από τους αισθητήρες και τις τιμές που στέλνουν στον μικροελεγκτή. Έπειτα, γίνεται επεξεργασία και αξιολόγηση των δεδομένων για να διαπιστωθεί αν οι τιμές τους είναι εντός των ορίων που έχει θέσει ο χρήστης ή αν τα υπερβαίνουν. Αφού, ολοκληρωθεί η παραπάνω διαδικασία, οι μετρήσεις στέλνονται ως μήνυμα στην οθόνη για να τα δει και ο χρήστης του συστήματος. Τέλος, εάν οι τιμές είναι εντός των ορίων η παραπάνω διαδικασία, γίνεται ξανά από την αρχή, εάν, όμως οι τιμές περάσουν τα όρια, τότε ο συναγερμός ενεργοποιείται, το φωτάκι αναβοσβήνει και ο βομβητής ηχεί.

Γενικότερα, αποτελεί ένα αξιόπιστο σύστημα ανίχνευσης φωτιάς, διότι, παρέχει συνεχή ενημέρωση και αξιολογεί την κατάσταση μέσα από δύο αισθητήρες διαφορετικής χρήσης. Αυτό, όμως, δεν το περιορίζει στο ζήτημα της επεκτασιμότητας, αφού μελλοντικά θα μπορούσαν να προστεθούν επιπλέον αισθητήρες και σύστημα GPS με μετάδοση συντεταγμένων, στην περίπτωση που το σύστημα αισθητήρων εγκατασταθεί σε περισσότερες από μία τοποθεσίες.

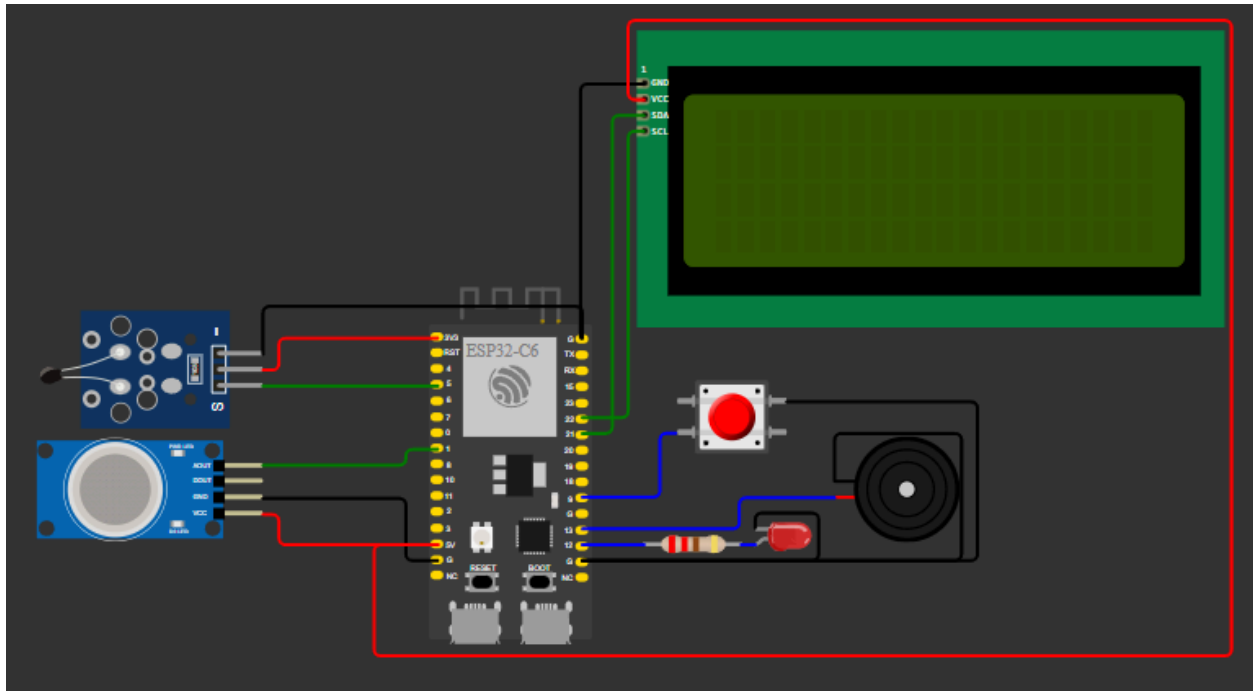
2.1 Διάγραμμα Ροής Δεδομένων 1^{ου} επιπέδου



3. Υλοποίηση

Επιλέχθηκε και χρησιμοποιήθηκε το ESP32 - C6 καθώς είναι η νεότερη έκδοση του ESP32 και προσφέρει γρηγορότερη σύνδεση στο διαδίκτυο (Wi-Fi 6) που είναι αναγκαία σε συστήματα άμεσης δράσης όπως η αναγνώριση μιας πυρκαγιάς. Για τον ίδιο λόγο χρησιμοποιήθηκε η γλώσσα προγραμματισμού C++ έναντι της micropython.

Το πρώτο βήμα ήταν η συνδεσμολογία των αισθητήρων, του κουμπιού, το λαμπάκι LED, του βομβητή και της οθόνης Wokwi-ICD2004. Ξεκινώντας με τους αισθητήρες, επιλέχθηκαν να χρησιμοποιηθούν αισθητήρες αερίων και θερμοκρασίας. Ο αισθητήρας θερμοκρασίας είναι αναλογικός ενώ ο αισθητήρας αερίων έχει δυνατότητα αναλογικής και ψηφιακής χρήσης, επιλέχθηκε μόνο η ψηφιακή του δυνατότητα καθώς ήταν επαρκής για τις απαιτήσεις του συστήματος. Προχωρώντας με το κουμπί, επιλέχθηκε για να μπορεί να υπάρξει χειροκίνητη ενεργοποίηση του συστήματος σε περίπτωση κάποιας μελλοντικής βλάβης του συστήματος. Έπειτα προστέθηκε το λαμπάκι LED και ο βομβητής ως μέτρα προειδοποίησης ότι το σύστημα ενεργοποιήθηκε. Τέλος, προστέθηκε η οθόνη Wokwi-ICD2004 με την έκδοση I2C 80 χαρακτήρων για μεγαλύτερη ευκολία σύνδεσης και δυνατότητα μεγαλύτερων μηνυμάτων.



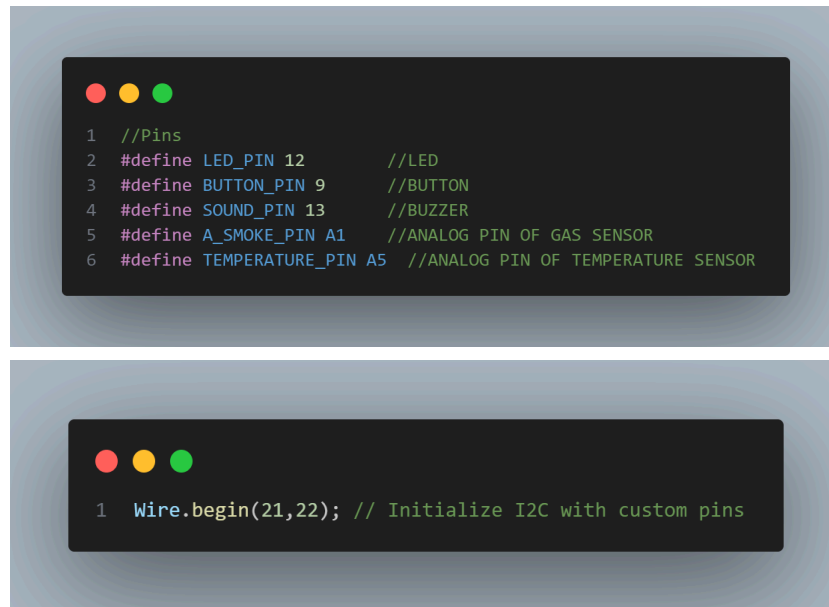
3.1 Συνδεσμολογία του συστήματος

Τα καλώδια είναι χρωματοποιημένα ανάλογα με την λειτουργία που παρέχουν, όπως φαίνεται στον πίνακα 3.2.

Χρώμα καλωδίου	Λειτουργία
Red	Παροχή ενέργειας 3.3V ή 5V
Blue	Ψηφιακή Σύνδεση
Green	Αναλογική Σύνδεση
Black	Γείωση

3.2 Πίνακας λειτουργίας καλωδίων

Το δεύτερο βήμα ήταν ο προγραμματισμός αυτών των εργαλείων για να ανταποκρίνονται στις απαιτήσεις του συστήματος. Ξεκινώντας με τον καθορισμό των πυλών στον κώδικα και τον διαχωρισμό τους σε αναλογικές και ψηφιακές όπως φαίνεται στον πίνακα 3.3.



Πίνακας 3.3

Κουμπί (Button)

Το κουμπί λειτουργεί με latching (μανδάλωση), όταν δεν είναι πατημένο, εκπέμπει 1(ένα) και όταν είναι πατημένο εκπέμπει 0(μηδέν). Ανάλογα με την κατάσταση το σύστημα εκτελεί το αντίστοιχο μέρος κώδικα. Στον πίνακα 3.4 φαίνεται ο κώδικας που εκτελείται με την αλλαγή κατάστασης.



```

1 //Button check
2 if (buttonReading) { //0 If pressed, 1 if not pressed
3     FIRE_DANGER = LOW;
4     message = "No Fire"; //Blynk message
5 }
6 else{
7     FIRE_DANGER = HIGH;
8     threat = 1;
9     printScreen(threat,celsius,smokeValue); //I2C Screen display
10    message = "Fire Danger: Button Pressed"; //Blynk message
11    counter +=1;
12 }

```

Πίνακας 3.4

Φωτάκι LED και βομβητής (Buzzer)

Τα δύο αυτά εργαλεία χρησιμοποιούνται ταυτόχρονα σε περίπτωση ενεργοποίησης του συστήματος. Στην εικόνα 3.5 φαίνεται η συνάρτηση στον κώδικα που τα ενεργοποιεί.

```

1 void beep(){
2     digitalWrite(LED_PIN, HIGH);
3     tone(SOUND_PIN, 262, 250); //262Hz for 250ms
4     delay(250);
5     digitalWrite(LED_PIN, LOW);
6     delay(250);
7 }

```

Εικόνα 3.5

Αισθητήρας αερίων (MQ2 gas sensor)

Ο αισθητήρας χρησιμοποιείται με την αναλογική του λειτουργία, ανιχνεύει την πυκνότητα του καπνού και αν ξεπεραστεί το επιτρεπόμενο όριο (479 ppm) ενεργοποιείται το σύστημα. Επίσης, αν η άνοδος της πυκνότητας είναι απότομη (μεγαλύτερη από 50 ppm σε 10 δευτερόλεπτα) ενεργοποιείται το σύστημα και ας μην είναι μεγαλύτερο το επίπεδο καπνού από το όριο. Στον πίνακα 3.6 παρουσιάζεται ο κώδικας για τις λειτουργίες του αισθητήρα.

```
1  int smokeValue = analogRead(A_SMOKE_PIN);
```

```
1  //gas sensor check
2  if (smokeValue > SMOKE_THRESHOLD_TEMP){
3    FIRE_DANGER = HIGH;
4    threat = 3;
5    printScreen(threat,celsius,smokeValue);
6    message = "Fire Danger (Fixed Smoke Concentration)";
7    counter +=1;
8  }
```

```
1  if(smokeValue - lastSmoke > RATE_OF_RISE_SMOKE){
2    FIRE_DANGER = HIGH; //Set fire danger to high
3    threat = 5; //Sudden rise in gas concentration
4    printScreen(threat,celsius,smokeValue); //I2C Screen display
5    message = "Fire Danger (Rate of Rise / Smoke)"; //Blynk message
6    sendMessages(smokeValue,celsius); //Blynk message
7    for(int i=0; i<5; i++){
8      beep(); //call the beep function (led and buzzer beep)
9    }
10 }
```

Πίνακας 3.6

Αισθητήρας θερμοκρασίας (temperature sensor)

Ο αισθητήρας χρησιμοποιείται για να ανιχνεύει την θερμοκρασία και αν ξεπεραστεί το επιτρεπόμενο όριο (55 °C) ενεργοποιείται το σύστημα. Επίσης, αν η άνοδος της θερμοκρασίας είναι απότομη (μεγαλύτερη από 5 °C σε 10 δευτερόλεπτα) ενεργοποιείται το σύστημα και ας μην είναι μεγαλύτερο το επίπεδο θερμοκρασίας από το όριο. Στον πίνακα 3.7 παρουσιάζεται ο κώδικας για τις λειτουργίες του αισθητήρα.

```
1 int temperatureValue = analogRead(TEMPERATURE_PIN);
2 float celsius = 1 / (log(1 / (4095. / temperatureValue - 1)) / BETA + 1.0 / 298.15) - 273.15;
```



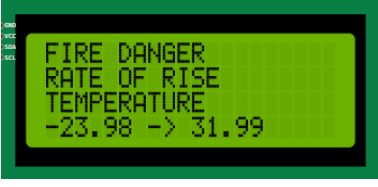


```
1 //temperature sensor check
2 if (celsius > FIRE_THRESHOLD_TEMP) {
3     FIRE_DANGER = HIGH;
4     threat = 2;
5     printScreen(threat,celsius,smokeValue);
6     message = "Fire Danger (Fixed Temperature)";
7     counter +=1;
8 }
```

```
1 if (celsius - lastTemp > RATE_OF_RISE_TEMP) {
2     FIRE_DANGER = HIGH; //Set fire danger to high
3     threat = 4; //Sudden rise in temperature
4     printScreen(threat,celsius,smokeValue); //I2C Screen display
5     message = "Fire Danger (Rate of Rise / Temperature)"; //Blynk message
6     sendMessages(smokeValue,celsius); //Blynk message
7     for(int i=0; i<5; i++){
8         beep(); //call the beep function (led and buzzer beep)
9     }
10 }
```

Πίνακας 3.7

Οθόνη wokwi-ICD2004 I2C 20x4

Η οθόνη της επιφάνειας του Wokwi προστέθηκε για την αναπαράσταση των δεδομένων και συγκεκριμένων μηνυμάτων. Στον πίνακα 3.8 παρουσιάζονται τα πιθανά μηνύματα που μπορεί να εμφανιστούν, στην εικόνα 3.9 ο κώδικας για την εμφάνιση των κατάλληλων μηνυμάτων βάσει τις διαφορετικές πιθανές αιτίες και στην εικόνα 3.10 η ανανέωση της οθόνης πριν την κάθε εκτύπωση.

Μήνυμα οθόνης Wokwi	Αίτιο
	Πατήθηκε το κουμπί
	Η θερμοκρασία ξεπέρασε τους 55°C
	Η θερμοκρασία ξεπέρασε την επιτρεπόμενη άνοδο των 5°C ανά 10 δευτερόλεπτα
	Η πυκνότητα αερίων ξεπέρασε τα 400 ppm
	Η πυκνότητα αερίων ξεπέρασε την επιτρεπόμενη άνοδο των 50 ppm ανά 10 δευτερόλεπτα

Πίνακας 3.8


```

1 //Print the threat message depending on the threat
2 void printScreen(int x,float temperature,int gas){
3     resetScreen();
4     switch(x){
5         //button
6         case 1:
7             lcd.setCursor(0,1);
8             lcd.print("FIRE DANGER:");
9             lcd.setCursor(0,2);
10            lcd.print("BUTTON PRESSED");
11            break;
12        //temperature
13        case 2:
14            lcd.setCursor(0,0);
15            lcd.print("FIRE DANGER:");
16            lcd.setCursor(0,1);
17            lcd.print("FIXED TEMPERATURE:");
18            lcd.setCursor(0,2);
19            lcd.print(temperature);
20            break;
21        //gas
22        case 3:
23            lcd.setCursor(0,0);
24            lcd.print("FIRE DANGER:");
25            lcd.setCursor(0,1);
26            lcd.print("FIXED SMOKE");
27            lcd.setCursor(0,2);
28            lcd.print("CONCENTRATION:");
29            lcd.setCursor(0,3);
30            lcd.print(gas);
31            lcd.print(" PPM");
32            break;
33        //Sudden rise in temperature
34        case 4:
35            lcd.setCursor(0,0);
36            lcd.print("FIRE DANGER");
37            lcd.setCursor(0,1);
38            lcd.print("RATE OF RISE");
39            lcd.setCursor(0,2);
40            lcd.print("TEMPERATURE");
41            lcd.setCursor(0,3);
42            lcd.print(lastTemp);
43            lcd.print(" -> ");
44            lcd.print(temperature);
45            break;
46        //sudden rise in gas concentration
47        case 5:
48            lcd.setCursor(0,0);
49            lcd.print("FIRE DANGER");
50            lcd.setCursor(0,1);
51            lcd.print("RATE OF RISE");
52            lcd.setCursor(0,2);
53            lcd.print("SMOKE");
54            lcd.setCursor(0,3);
55            lcd.print(lastSmoke);
56            lcd.print(" PPM");
57            lcd.print(" -> ");
58            lcd.print(gas);
59            lcd.print(" PPM");
60            break;
61        default:
62            lcd.setCursor(0,0);
63            lcd.print("FALSE ALARM");
64        }
65        delay(200);
66    }

```

Εικόνα 3.9

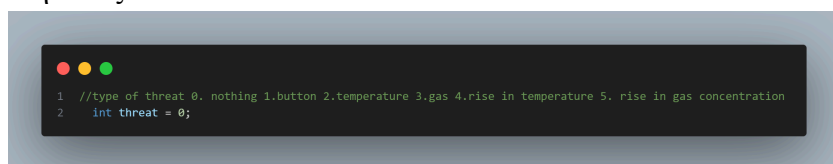


Εικόνα 3.10

Περιγραφή αναγκαίων μεταβλητών:

currentTime - lastCheckTime: σε συνδυασμό με την συνάρτηση millis() με την χρήση των μεταβλητών δημιουργήθηκε το σύστημα ελέγχου για απότομη αύξηση της θερμοκρασίας/πυκνότητα αερίων και αποστολής μηνυμάτων κάθε 10 δευτερόλεπτα στην πλατφόρμα blynk.

threat: αποθήκευση είδος κινδύνου

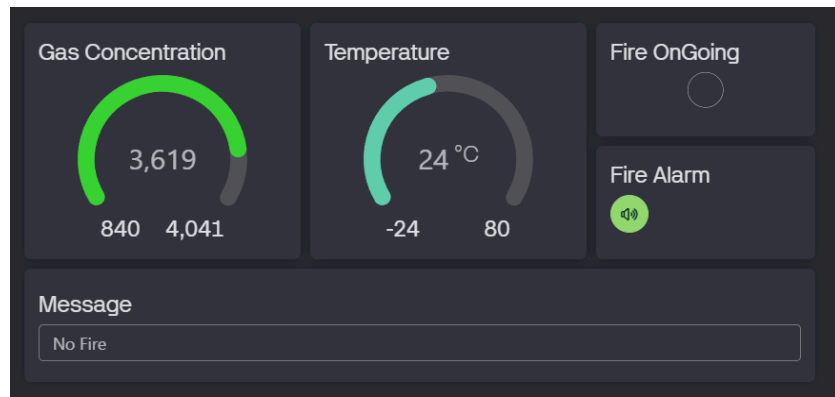


Εικόνα 3.11

counter: ανίχνευση για παραπάνω από 1 ενεργό κίνδυνο. Η μεταβλητή χρειάζεται για να μπορεί το σύστημα να λειτουργεί σωστά αν εξαφανιστεί ο ένας κίνδυνος αλλά παραμένει κάποιος άλλος.

Πλατφόρμα Blynk

Η πλατφόρμα Blynk χρησιμοποιήθηκε για την αναπαράσταση των δεδομένων στο διαδύκτιο. Στην εικόνα 3.12 φαίνεται το περιβάλλον που βλέπει ο χρήστης που λαμβάνει δεδομένα.



Εικόνα 3.12

Παρακάτω δίνεται ο πίνακας 3.13 με το τι αντιστοιχεί το κάθε πεδίο.

Πεδίο	Λειτουργία
	Επίπεδο συγκέντρωσης αερίων, αλλάζει χρώματα βάση την τιμή.
	Επίπεδο θερμοκρασίας, αλλάζει χρώματα βάση την τιμή
	LED, αναβοσβήνει κόκκινο αν το σύστημα ενεργοποιηθεί
	Βομβητής, δημιουργεί ήχο και αλλάζει χρώμα αν το σύστημα ενεργοποιηθεί
	Εμφανίζει το κατάλληλο μήνυμα βάση το τι προκάλεσε ενεργοποίηση του συστήματος.

Πίνακας 3.13

Στον παρακάτω πίνακα 3.14 φαίνεται ο κώδικας που χρειάστηκε για να γίνει ένωση της πλατφόρμας Wokwi με την πλατφόρμα blynk καθώς και σύντομη περιγραφή του τι αποσκοπεί το κάθε σκέλος.

```
1 //Blynk Initialise
2 #define BLYNK_TEMPLATE_ID "TMPL4IK18E-Cz"
3 #define BLYNK_TEMPLATE_NAME "Iot Fire Detection"
4 #define BLYNK_AUTH_TOKEN "MpgZyJ_uzKBV1m5mUymcdXP_B8Zdr1vi"
```

καθορισμός τιμών για ενεργοποίηση ζεύξης

```
1 //Wifi
2 char SSID[] = "Wokwi-GUEST";
3 char PASS[] = "";
```

Στοιχεία Wifi για σύνδεση στο διαδίκτυο

```
1 #define V_SMOKE_VALUE V1 // Virtual Pin for Smoke Value
2 #define V_CELSIUS_VALUE V2 // Virtual Pin for Celsius Temperature
3 #define V_FIRE_ALARM V3 // Virtual Pin for Fire Alarm Status (0 for no fire, 1 for fire)
4 #define V_SCREEN_MESSAGE V4 // Virtual Pin for the message of the LCD Screen
```

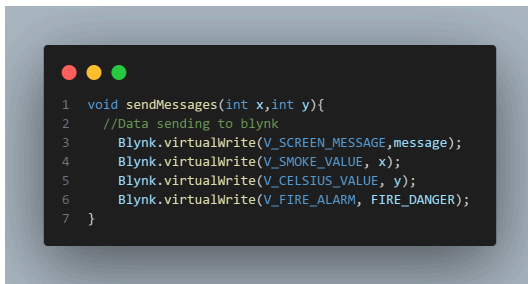
Δημιουργία datastreams

```
1 Blynk.begin(BLYNK_AUTH_TOKEN, SSID, PASS);
```

Εκκίνηση του blynk και σύνδεση στο διαδίκτυο

```
1 Blynk.run();
```

Εκκίνηση ζεύξης blynk



Αποστολή των datastreams στο blynk

Πίνακας 3.14

4. Συμπεράσματα

Η αξιολόγηση του τελικού αποτελέσματος κατέδειξε ότι το σύστημα ανταποκρίθηκε επιτυχώς στις προκαθορισμένες προσδοκίες και στους στόχους. Επιτεύχθηκε η έγκαιρη αναγνώριση πιθανής πυρκαγιάς και η άμεση ειδοποίηση του χρήστη. Το σύστημα επέδειξε αξιόλογη σταθερότητα, ταχεία απόκριση και συνέπεια στη λειτουργία του, υποστηρίζοντας τόσο την τοπική όσο και την απομακρυσμένη επιτήρηση. Η ικανότητα συνεχούς παρακολούθησης του χώρου, η αυτόματη ενεργοποίηση συναγερμού σε κρίσιμες καταστάσεις και η υποστήριξη ειδοποιήσεων σε πραγματικό χρόνο ενίσχυσαν σημαντικά την αποτελεσματικότητά του.

Ένα επιπλέον θετικό στοιχείο ήταν η προσβασιμότητα της κατασκευής, τόσο από πλευράς κόστους όσο και ως προς την κατανόηση της λειτουργίας του, γεγονός που καθιστά την εφαρμογή πρακτική και με δυνατότητα επέκτασης σε μελλοντικές χρήσεις. Η συνολική εμπειρία από την υλοποίηση ανέδειξε τη χρησιμότητα των τεχνολογιών αυτοματισμού και επικοινωνίας για την ενίσχυση της ασφάλειας.

Παρά τα ενθαρρυντικά αποτελέσματα, εντοπίστηκαν ορισμένοι περιορισμοί, κυρίως σε σχέση με την ακρίβεια των αισθητήρων σε απαραίτητα περιβάλλοντα, την εξάρτηση της απομακρυσμένης λειτουργίας από τη σταθερότητα της σύνδεσης και την ανάγκη για τακτική συντήρηση. Οι παρατηρήσεις αυτές υποδεικνύουν πιθανές κατευθύνσεις για μελλοντική βελτιστοποίηση, με στόχο τη δημιουργία ακόμη πιο αξιόπιστων και ευφρών λύσεων.

Βιβλιογραφία

Διαδικτυακοί Πόροι:

CodeSnap - VS code extension: Για την δημιουργία των screenshot του κώδικα

[CodeSnap Extended - Visual Studio Marketplace](#)

Wokwi: Απεικόνιση hardware σε ψηφιακό περιβάλλον

[Wokwi - World's most advanced ESP32 Simulator](#)

Blynk: Δημιουργία User Interface για την απεικόνιση δεδομένων στο διαδίκτυο

[Blynk: a low-code IoT software platform for businesses and developers](#)

Παράρτημα 1. Τεκμηρίωση

Documentation για το πως λειτουργεί το κάθε εργαλείο στο Wokwi:

[Welcome to Wokwi! | Wokwi Docs](#)

Documentation για το πως λειτουργεί η πλατφόρμα Blynk:

[Introduction | Blynk Documentation](#)