

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΤΜΗΜΑΤΟΣ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ**



**Ηλεκτρονική Πλατφόρμα Παρακολούθησης Δεικτών
κατά τη Μεταφορά Ευπαθών Προϊόντων με την
χρήση τεχνολογιών IoT**

**Διπλωματική Εργασία του φοιτητή
*Παπόγλου Βασιλείου***

Επιβλέπων Καθηγητής: *Χατζηγεωργίου Αλέξανδρος*

Θεσσαλονίκη, Φεβρουάριος 2022

Ηλεκτρονική Πλατφόρμα Παρακολούθησης Δεικτών κατά τη Μεταφορά Ευπαθών Προϊόντων με την χρήση τεχνολογιών IoT

Βασίλειος Παπόγλου

Προηγούμενο πτυχίο: Πληροφορική, ΑΠΘ, 2017

Διπλωματική Εργασία

υποβαλλόμενη για τη μερική εκπλήρωση των απαιτήσεων του

ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΤΙΤΛΟΥ ΣΠΟΥΔΩΝ ΣΤΗΝ ΕΦΑΡΜΟΣΜΕΝΗ ΠΛΗΡΟΦΟΡΙΚΗ

Επιβλέπων Καθηγητής:
Αλέξανδρος Χατζηγεωργίου

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την ηη/μμ/εεεε

Όνοματεπώνυμο 1

Όνοματεπώνυμο 2

Όνοματεπώνυμο 3

.....

.....

.....

Βασίλειος Παπόγλου

ΠΕΡΙΛΗΨΗ

Αυτή η εργασία εξετάζει τη χρήση της τεχνολογίας IoT στην ασφάλεια της μεταφοράς προϊόντων στο πλαίσιο του παγκόσμιου εμπορίου. Διεξήχθη βιβλιογραφική ανασκόπηση χρησιμοποιώντας ακαδημαϊκά έγγραφα γραμμένα στην αγγλική γλώσσα και δημοσιευμένα σε επιστημονικά περιοδικά. Βασικά ευρήματα και συμπεράσματα: Το IoT στην ασφάλεια των προϊόντων είναι μια σχετικά νέα προσέγγιση με το πρώτο άρθρο να χρονολογείται το 2009 και από το 2013 και μετά, να έχουν πολλαπλασιαστεί. Οι μελέτες αυτές αφορούν θεωρητικές αναλύσεις, εννοιολογικά μοντέλα αλλά και πραγματικές εφαρμογές IoT που αφορούν τις αλυσίδες εφοδιασμού και συγκεκριμένα την παρακολούθηση της ασφάλειας και της ποιότητας των προϊόντων τους. Η συντριπτική πλειοψηφία των δημοσιεύσεων αφορούσε τα τρόφιμα, το κρέας, τα γεωργικά και ιατροφαρμακευτικά προϊόντα. Αυτές οι μελέτες χρησιμοποίησαν αισθητήρες για την παρακολούθηση κυρίως της θερμοκρασίας, της υγρασίας και της τοποθεσίας. Αυτή η εργασία συνοψίζει τις γνώσεις και τις τεχνολογίες που έχει αναπτύξει και εξελίξει η επιστημονική κοινότητα και επιδιώκει, μέσω της υλοποίησης ενός πραγματικού συστήματος παρακολούθησης μεταφερόμενων προϊόντων, να προσθέσει χρήσιμη γνώση και ιδέες για την περαιτέρω ανάπτυξη του παγκόσμιου εμπορίου, της βιομηχανίας και της κρατικής αποτελεσματικότητας μέσω χρήσης σύγχρονης τεχνολογίας IoT. Πραγματοποιήθηκε αναζήτηση βιβλιογραφίας σε πολλές βάσεις δεδομένων που περιέχουν επιστημονικές δημοσιεύσεις και πρακτικά συνεδρίων (π.χ. IEEE, Scopus, Google Scholar). Αυτή η αναζήτηση οδήγησε σε 336 δυνητικά σχετικές εργασίες από τις οποίες μετά από περαιτέρω αξιολόγηση προέκυψαν 56 σχετικές δημοσιεύσεις, οι οποίες χρησιμοποιήθηκαν γι' αυτήν την εργασία.

Λέξεις Κλειδιά: Διαδίκτυο των Πραγμάτων, Εφοδιαστική Αλυσίδα, Ψυχρή Αλυσίδα, Μεταφορά Ευπαθών Προϊόντων, Ασφάλεια Προϊόντων, Παρακολούθηση Θερμοκρασίας, Εξυπηρετητές Νέφους

ABSTRACT

This paper examines the use of IoT technology in the security of product shipments in the context of world trade. A literature review was conducted using academic papers written in English and published in scientific journals. Key Findings and Conclusions: The IoT in product safety is a relatively new approach with the first article dating back to 2009 and articles multiplied after 2013. These studies concern theoretical analyzes, conceptual models and real IoT applications and systems related to supply chains and in particular the monitoring of the safety and quality of their products. The vast majority of publications were on food, meat, agricultural and medical products. These studies used sensors to monitor mainly temperature, humidity and location. This paper summarizes the knowledge and technologies developed by the scientific community and seeks, through the implementation of a real tracking system, to add useful knowledge and ideas for the further development of world trade, industry and government efficiency through the use of modern IoT technology. Bibliography sources were searched in many databases containing scientific publications and conference proceedings (e.g. IEEE, Scopus, Google Scholar). This search resulted in 336 potentially relevant papers, of which further evaluation yielded 56 relevant publications, which were used for this work.

Key Words: Internet of Things, Supply Chain, Cold Chain, Transportation of Vulnerable Products, Product Safety, Temperature Monitoring, Cloud Servers

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ	5
1. ΕΙΣΑΓΩΓΗ	7
Σκοπός και διάρθρωση εργασίας	9
2. ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΑΝΑΣΚΟΠΗΣΗ	11
3. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΟΣ	19
3.1 Αρχικό σύστημα	19
Συνδεσμολογία	21
3.2 Αναβαθμισμένο σύστημα	23
3.3 Αυτόματος χειρισμός κλιματισμού	25
3.3.1 Αποκωδικοποίηση σήματος	26
Συνδεσμολογία	26
Προγραμματισμός	28
3.3.2 Κλωνοποίηση και αποστολή σήματος	33
Συνδεσμολογία	33
Προγραμματισμός	33
3.4 Αποστολή δεδομένων στο Cloud και απομακρυσμένος έλεγχος	36
Συνδεσμολογία	37
Προγραμματισμός	38
3.5 Αποστολή email alerts	42
3.5 Γεωγραφικός εντοπισμός μέσω GPS	46
4. ΜΕΛΕΤΗ ΠΕΡΙΠΤΩΣΗΣ & ΑΠΟΤΕΛΕΣΜΑΤΑ	49
5. ΜΕΛΛΟΝΤΙΚΗ ΕΡΕΥΝΑ	58
6. ΣΥΜΠΕΡΑΣΜΑΤΑ	60
ΒΙΒΛΙΟΓΡΑΦΙΑ	61

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ

Εικόνα 1	Αισθητήρας θερμοκρασίας LM35
Εικόνα 2	Καλώδια σύνδεσης ή jumper wire
Εικόνα 3	MB-102 Breadboard
Εικόνα 4	HC-06 ασύρματη bluetooth συσκευή
Εικόνα 5	Arduino UNO R3
Εικόνα 6	USB καλώδιο
Εικόνα 7	Αποτελέσματα θερμοκρασίας στο Arduino IDE Serial Monitor
Εικόνα 8	DHT11 Αισθητήρας θερμοκρασίας και υγρασίας
Εικόνα 9	Αποτελέσματα θερμοκρασίας και υγρασίας στο Arduino IDE Serial Monitor
Εικόνα 10	IR Receiver Infrared Module
Εικόνα 11	IR Transmitter Infrared
Εικόνα 12	Το κύκλωμα του IR Receiver
Εικόνα 13	Αποτελέσματα ανεπιτυχούς αναγνώρισης IR σήματος στο Arduino IDE Serial Monitor
Εικόνα 14	Αποτελέσματα επιτυχούς αναγνώρισης IR σήματος στο Arduino IDE Serial Monitor
Εικόνα 15	NodeMcu V3 Lua CH340G ESP8266
Εικόνα 16	USB cable for data
Εικόνα 17	Συνδεσμολογία NodeMCU με DHT11
Εικόνα 18	Αποτελέσματα αποστολής θερμοκρασίας και υγρασίας στο Cloud
Εικόνα 19	Γραφικά αποτελέσματα θερμοκρασίας και υγρασίας στο Cloud
Εικόνα 20	Εναλλαγή των γραφικών αποτελεσμάτων θερμοκρασίας και υγρασίας στο Cloud

Εικόνα 21	Γραφικά αποτελέσματα GPS στο Cloud
Εικόνα 22	Αποτελέσματα επιτυχούς σύνδεσης του NodeMCU στο Wifi και ανανέωσης των αποτελεσμάτων στο Cloud στο Arduino IDE Serial Monitor για την μελέτη περίπτωσης
Εικόνα 23	Απεικόνιση γραφικών αποτελεσμάτων θερμοκρασίας και υγρασίας στο Cloud για την μελέτη περίπτωσης
Εικόνα 24	Αποτελέσματα επιτυχούς ανανέωσης αποτελεσμάτων θερμοκρασίας και υγρασίας στο Arduino IDE Serial Monitor για την μελέτη περίπτωσης
Εικόνα 25	Αποτελέσματα επιτυχούς ανανέωσης γραφικών αποτελεσμάτων θερμοκρασίας και υγρασίας στο Cloud για την μελέτη περίπτωσης
Εικόνα 26	Αποτελέσματα 2ης επιτυχούς ανανέωσης αποτελεσμάτων θερμοκρασίας και υγρασίας στο Arduino IDE Serial Monitor για την μελέτη περίπτωσης
Εικόνα 27	Αποτελέσματα 2ης επιτυχούς ανανέωσης γραφικών αποτελεσμάτων θερμοκρασίας και υγρασίας στο Cloud για την μελέτη περίπτωσης
Εικόνα 28	Ειδοποιήσεις email για θερμοκρασία που είναι διαφορετική των 20° C
Εικόνα 29	Φωτογραφία ολοκληρωμένου συστήματος

1. ΕΙΣΑΓΩΓΗ

Τα τελευταία χρόνια, λόγω της ραγδαίας ανάπτυξης της τεχνολογίας και της εύκολης πρόσβασης κάθε ανθρώπου σε σύγχρονα μέσα επικοινωνίας έχει παρατηρηθεί μια αλματώδης ανάπτυξη του παγκοσμίου εμπορίου και των μεταφορών προϊόντων από την μια άκρη του κόσμου στην άλλη. Αυτή η ανάπτυξη έχει δημιουργήσει νέες προκλήσεις για τον επιχειρηματικό και ερευνητικό κόσμο. Καλούνται πλέον να βρουν αποτελεσματικούς τρόπους διαχείρισης της μεταφοράς ευπαθών προϊόντων και φαρμάκων. Ειδικότερα αυτή η ανάγκη έγινε επιτακτική με την έξαρση της πανδημίας Covid-19 όπου η επιστημονική κοινότητα και οι κρατικοί οργανισμοί κλήθηκαν άμεσα να βρουν αξιόπιστα μέσα αποθήκευσης και μεταφοράς ώστε τα εμβόλια αντιμετώπισης της πανδημίας να συντηρηθούν με ασφάλεια καθώς αυτά απαιτούσαν συντήρηση σε θερμοκρασίες έως και -70°C και η ζήτηση ήταν πρωτόγνωρα υψηλή. Αυτή η ανάγκη καλύπτεται με την ανάπτυξη συστημάτων που παρέχουν βέλτιστη θερμοκρασία κατά την αποθήκευση και τη μεταφορά προϊόντων σε κάθε στάδιο της διαδρομής τους, γνωστών ως ψυχρή αλυσίδα ή cold-chain. Η ψυχρή αλυσίδα αναφέρεται ως «μεταφορά ευαίσθητων στη θερμοκρασία προϊόντων κατά μήκος μιας αλυσίδας εφοδιασμού μέσω θερμικών μεθόδων συσκευασίας και ψύξης και υλικοτεχνικός σχεδιασμός για την προστασία της ακεραιότητας αυτών των αποστολών». Είναι μια διαδικασία κατά την οποία πρέπει να εκτελεστεί μια σειρά εργασιών για την προετοιμασία, αποθήκευση, μεταφορά και παρακολούθηση των ευαίσθητων στη θερμοκρασία προϊόντων, από τους χώρους παροχής στους χώρους υποδοχής (Luo et al., 2016).

Σε αντίθεση με τις παραδοσιακές αλυσίδες εφοδιασμού τα προϊόντα στο cold-chain έχουν μικρότερη διάρκεια ζωής και μεγαλύτερη ευαισθησία στις περιβαλλοντικές συνθήκες όπως για παράδειγμα στην θερμοκρασία, την υγρασία και τον φωτισμό. Απαιτούνται συστήματα ψύξης και αφύγρανσης σε κάθε σημείο του cold-chain πράγμα το οποίο έχει οδηγήσει στο αυξανόμενο ερευνητικό ενδιαφέρον για την διαχείριση του (Cold-Chain Management - CCM). Το CCM αποσκοπεί στο να προβλέπει τους κινδύνους στην αλλοίωση της ποιότητας των προϊόντων ώστε να αποφευχθεί η οικονομική ζημία αλλά και η έκθεση του καταναλωτή σε κίνδυνο.

Για να διασφαλιστεί ότι τα ευπαθή προϊόντα δεν θα υποστούν αλλοίωση ή απώλεια κατά τη διάρκεια της διαδικασίας αποστολής τους, οι επιχειρήσεις στις βιομηχανίες τροφίμων και ιατροφαρμακευτικών προϊόντων βασίζονται ολοένα και περισσότερο στην ψυχρή αλυσίδα. Πλέον, ο έλεγχος της θερμοκρασίας στην αποστολή τροφίμων γίνεται ιδιαίτερα ζωτικής σημασίας για τη βιομηχανία τροφίμων, ειδικά στις αναπτυσσόμενες χώρες ενώ η συνεχιζόμενη άνοδος του βιοτικού επιπέδου σε αυτές τις χώρες οδηγεί στην αυξανόμενη ζήτηση. Για παράδειγμα, περίπου το 70% του συνόλου των τροφίμων που καταναλώνονται στις ΗΠΑ μεταφέρονται μέσω ψυχρής αλυσίδας, ενώ αντίστοιχα στην Κίνα λιγότερο από το 25% του κρέατος και περίπου το 5% των φρούτων και λαχανικών που καταναλώνονται (Luo et al., 2016). Από την άλλη πλευρά, οι περισσότερες αναπτυσσόμενες χώρες εστιάζουν την εξαγωγική τους οικονομία στα τρόφιμα και την παραγωγή οπότε χρειάζονται κατάλληλη μεταφορά προϊόντων διατροφής για να μειώσουν την πιθανότητα μόλυνσης από βακτήρια, μικρόβια και μύκητες. Ως εκ τούτου, η διατήρηση των προϊόντων διατροφής για παρατεταμένες χρονικές περιόδους έχει αποκτήσει σημασία για εμπορικούς λόγους και λόγους υγείας. Οι επιχειρήσεις πρέπει να αντιμετωπίσουν διάφορα προβλήματα, όπως η

έλλειψη προτύπων θερμοκρασίας, οι ελλειπείς εγκαταστάσεις υποδομής, οι ξεπερασμένες τεχνολογίες και η έλλειψη επαγγελματιών.

Για παράδειγμα, το 2017, η επιχείρηση Lucky's Market αναγκάστηκε να πετάξει όλα τα τρόφιμα επειδή η συντήρησή τους έγινε στους 16°C αντί για τους κάτω από 4.4°C που απαιτούνταν (Nerbonig, 2017) ενώ περίπου 1 στους 6 Αμερικανούς πολίτες ετησίως, πάσχει από τροφικές ασθένειες λόγω μολυσμένων τροφίμων που συντηρούνται σε ακατάλληλη θερμοκρασία (Wein, 2014). Ειδικότερα, σχετικά με τις επιπτώσεις της πανδημίας, το Κέντρο Ασθενειών και Πρόληψης στις Ηνωμένες Πολιτείες της Αμερικής ανέφερε 16.233 περιπτώσεις COVID-19 και 239 θανάτους μεταξύ εργαζομένων στις ΗΠΑ σε 115 εγκαταστάσεις επεξεργασίας κρέατος και πουλερικών σε 19 πολιτείες. Στην Κίνα που είναι ένας από τους μεγαλύτερους εισαγωγείς κατεψυγμένου κρέατος, κυρίως από τη Νέα Ζηλανδία, την Αυστραλία και τη Βραζιλία, οι αρχές βρήκαν ίχνη κορονοϊού στο βόειο κρέας και στη συσκευασία του από τη Βραζιλία, τη Βολιβία, την Αυστραλία και τη Νέα Ζηλανδία. Ίχνη του COVID-19 βρέθηκαν επίσης σε συσκευασίες χοιρινού κρέατος από την Αργεντινή. Οι κινεζικές αρχές σφράγισαν τις αποθήκες, ακύρωσαν τις άδειες εισαγωγής των εταιρειών και απαγόρευαν τις εισαγωγές από διάφορες εταιρείες επηρεάζοντας πολύ αρνητικά την εργασία στην Κίνα και στο εξωτερικό (Iftekhhar et al., 2021).

Σύμφωνα με τον FAO (Οργανισμός Τροφίμων και Γεωργίας), αναφέρονται ετησίως 1,3 δισεκατομμύρια τόνοι απωλειών τροφίμων που αντιπροσωπεύουν το 33% της συνολικής παραγωγής. Η ζήτηση τροφίμων αυξάνεται συνεχώς και θα μπορούσε να φτάσει περίπου το 150-170% της παρούσης μέχρι το 2050. Επιπλέον, σύμφωνα με εκτίμηση του Παγκόσμιου Οργανισμού Υγείας (ΠΟΥ), περίπου 1,7 εκατομμύρια θάνατοι ετησίως σε όλο τον κόσμο συνδέονται με χαμηλή πρόσληψη φρέσκων λαχανικών. Το ποσοστό απωλειών φρέσκων λαχανικών είναι το ψηλότερο μεταξύ όλων των άλλων τροφίμων και συγκεκριμένα στο 45%, συμπεριλαμβανομένου του κρέατος (20%), των ελαιόσπορων (20%), των γαλακτοκομικών (20%), των δημητριακών (30%) και των θαλασσινών (35%) (Afreem et al., 2021).

Αντίστοιχα, για τον τερματισμό της πανδημίας του Covid-19 είναι αναγκαίος ο εμβολιασμός όπου τα εμβόλια πρέπει να βρίσκονται υπό τον αυστηρό έλεγχο της θερμοκρασίας από την κατασκευή μέχρι τα χέρια των επαγγελματιών υγείας που θα ανοσοποιήσουν τον πληθυσμό καθώς μπορεί να χάσουν γρήγορα την αποτελεσματικότητά τους εάν αλλοιωθεί η θερμοκρασία τους ανά πάσα στιγμή, ειδικά κατά τη μεταφορά και την αποθήκευση. Υπάρχουν αρκετές αναφορές στις ειδήσεις για απώλεια εμβολίων λόγω προβλημάτων θερμοκρασίας. Για παράδειγμα, στις 20 Ιανουαρίου 2021, οι αρχές στο Μίσιγκαν ανακοίνωσαν ότι σχεδόν 12.000 δόσεις του εμβολίου Moderna Covid-19 είχαν καταστραφεί λόγω δυσλειτουργίας του ελέγχου θερμοκρασίας κατά τη διάρκεια της αποστολής και στο Maine, περισσότερες από 16.000 δόσεις εμβολίου είχαν αλλοιωθεί. Στη Βοστώνη, 1.900 δόσεις εμβολίου κατά του Covid καταστράφηκαν λόγω μιας χαλαρής τάπας κατάψυξης. Στο Μιλγουόκι, ένας υπάλληλος νοσοκομείου κατέστρεψε περισσότερες από 500 δόσεις αφαιρώντας τα εμβόλια από ένα ψυγείο. Στην πόλη του Ρίο ντε Τζανέιρο, εκατοντάδες εμβόλια για τον COVID-19, που πρέπει να διατηρούνται μεταξύ 2 °C και 8 °C, χάθηκαν μετά από διακοπή ρεύματος. Ο Παγκόσμιος Οργανισμός Υγείας έχει εκδώσει συγκεκριμένες οδηγίες για την ψυχρή αλυσίδα του εμβολίου προκειμένου να αντιμετωπιστεί το πρόβλημα (Mendonca et al., 2021). Επίσης η πρόοδος τα τελευταία χρόνια για τις κυτταρικές και γενετικές θεραπείες, οι οποίες περιλαμβάνουν ζωντανά κύτταρα, έχει οδηγήσει σε

αυξανόμενη ανάγκη για αποθήκευση σε ακραίες συνθήκες κατάψυξης: -80°C ή ακόμα και -180°C που απαιτεί υγρό άζωτο (LN₂)(Rusnack, 2021).

Πέραν των προϊόντων, το CCM μπορεί να χρησιμεύσει και στην πρόβλεψη του κινδύνου εργατικών ατυχημάτων. Η υπερβολική έκθεση των διαχειριστών των προϊόντων στο κρύο μπορεί να προκαλέσει σοβαρές επιπτώσεις στην υγεία τους (Rice, 2014). Οι επιπτώσεις αυτές ποικίλουν από ένα απλό κρυολόγημα και συμπτώματα που σχετίζονται με αυτό, μέχρι και τραυματισμούς από το κρύο όπως κρυοπαγήματα (Mäkinen and Hassi, 2009). Για παράδειγμα, συνολικά 15 εργαζόμενοι έχασαν την ζωή τους και 26 εργάτες ασθένησαν όταν εργάζονταν σε ψυγείο της Σαγκάης λόγω διαρροής αμμωνίας (Laurence, 2013). Ένα αποτελεσματικό σύστημα cold-chain λοιπόν, που θα παρακολουθούσε και θα αξιολογούσε το επίπεδο του κινδύνου σε όλα τα σημεία της αλυσίδας, θα συνέβαλε όχι μόνο στην ποιότητα των προϊόντων και την ασφάλεια των καταναλωτών αλλά και στην ασφάλεια του προσωπικού που εργάζεται σε αυτά τα περιβάλλοντα.

Επομένως, κρίνεται αναγκαίο σε κάθε σημείο του cold-chain να υπάρχει έλεγχος και διασφάλιση ότι τα προϊόντα βρίσκονται στις κατάλληλες περιβαλλοντικές συνθήκες ώστε να διατηρείται η ποιότητα τους και να ανιχνεύονται άμεσα οποιοσδήποτε μεταβολές των περιβαλλοντικών συνθηκών καθ'όλη την διάρκεια της μεταφοράς ή αποθήκευσης. Γι' αυτούς τους σκοπούς έχει χρησιμοποιηθεί ευρέως τα τελευταία χρόνια η τεχνολογία Internet of Things (IoT) ή αλλιώς "Δίκτυο των Πραγμάτων". Το IoT είναι ένα δομημένο δίκτυο για την διασύνδεση καθημερινών αντικειμένων με την χρήση έξυπνων τεχνολογιών αναγνώρισης και ανίχνευσης (Yang, 2014). Με την χρήση του IoT αλλά και Τεχνητής Νοημοσύνης (AI) μπορεί να παρακολουθείται σε άμεσο χρόνο το περιβάλλον των προϊόντων και οι συνθήκες του, προσφέροντας αποτελεσματικό Cold-Chain Management ώστε η ποιότητα τους να παραμένει άρτια και η εργασία σε τέτοια περιβάλλοντα να είναι ασφαλής. Μετά την επανάσταση του διαδικτύου, η ανάπτυξη του IoT βρίσκεται σε συνεχή εξέλιξη τα τελευταία χρόνια. Η ιδέα αρχικά ονομάστηκε και συζητήθηκε από τον Kevin Ashton το 1999 και στην συνέχεια έγινε ολοένα και περισσότερο αναγνωρισμένη μέσω της έρευνας και της συζήτησης από το MIT Auto-ID Center (Shang et al., 2015). Το IoT θολώνει τη γραμμή μεταξύ του ψηφιακού και του φυσικού κόσμου και θεωρείται το δίκτυο επόμενης γενιάς ή το μελλοντικό Διαδίκτυο που συμπεριλαμβάνει ποικιλία λειτουργικών αντικειμένων όπως ετικέτες RFID, αισθητήρες, ενεργοποιητές, κινητά τηλέφωνα κ.λπ.

Σκοπός και διάρθρωση εργασίας

Ο σκοπός της παρούσας εργασίας είναι η διερεύνηση των σύγχρονων τρόπων διαχείρισης του cold-chain καθώς και η ανάλυση των συστημάτων IoT που χρησιμοποιούνται και καθιστούν αυτή την διαχείριση αποτελεσματική αλλά και των τεχνολογιών που τα συνοδεύουν. Οι πάροχοι και οι διαχειριστές των συστημάτων IoT προσφέρουν τις υπηρεσίες τους στις εμπορικές επιχειρήσεις, οι οποίες με την σειρά τους καταβάλουν κάποιο συμφωνηθέν αντίτιμο για τις σύγχρονες αυτές υπηρεσίες που διασφαλίζουν την ποιότητα των προϊόντων τους.

Η τυπική λύση για τη διατήρηση της θερμοκρασίας των ευαίσθητων στη θερμότητα προϊόντων είναι η χρήση ψυκτικών μηχανών και ψυγείων, που χρησιμοποιούν σημαντική ποσότητα ηλεκτρικής ενέργειας και ενέργειας καυσίμου που κοστίζει και εκλύει τεράστια

ποσότητα διοξειδίου του άνθρακα (CO₂) στην ατμόσφαιρα. Καθώς το επίπεδο εκπαίδευσης των ανθρώπων έχει αυξηθεί, οι άνθρωποι συνειδητοποιούν περισσότερο τα περιβαλλοντικά ζητήματα και λαμβάνοντας υπόψη τις περιβαλλοντικές επιπτώσεις, οι κυβερνήσεις έχουν εκδώσει ορισμένες αυστηρές οδηγίες και κανονισμούς για τις εκπομπές από τη μεταποίηση και άλλους βιομηχανικούς τομείς. Εξαιτίας αυτού, οι βιομηχανίες παραγωγής και οι ψυχρές αλυσίδες έχουν υποστεί αυξανόμενες πιέσεις για να μειώσουν τις περιβαλλοντικές επιπτώσεις τους (Kumar et al., 2021). Παρά τις οικονομικές και περιβαλλοντικές ανησυχίες, η ασφάλεια των τροφίμων και η ασφάλεια της υγείας είναι οι δύο βασικές ανάγκες κάθε έθνους. Η βελτίωση αυτών των υπηρεσιών θα προσφέρει πολλαπλά οφέλη και στις δύο πλευρές και θα βοηθήσει στην περαιτέρω ανάπτυξη του παγκόσμιου εμπορίου, της υγείας και της ευημερίας. Σύμφωνα με αυτό το πλαίσιο λοιπόν, σε αυτή την διπλωματική εργασία θα διερευνηθούν τυχόν ερευνητικά κενά στην διαχείριση του cold-chain αλλά και προοπτικές που μπορεί να κάνουν την χρήση του IoT αποδοτικότερη και αποτελεσματικότερη στο μέλλον.

Μέσα από βιβλιογραφική έρευνα, παρουσιάζονται οι έννοιες του Cold-Chain Management και οι μεταβλητές που συνθέτουν το Internet of Things αλλά και οι ευρύτερες τεχνολογίες που συνδυάζονται με τα παραπάνω όπως η Τεχνητή Νοημοσύνη (AI), η Μηχανική Μάθηση (Machine Learning), το blockchain και άλλα. Επίσης μέσω της βιβλιογραφικής ανασκόπησης, διερευνάται σε ποιο σημείο έχουν προχωρήσει οι έρευνες πάνω στο αντικείμενο και μελετώνται αντίστοιχα συστήματα και εργαλεία που έχουν δημιουργηθεί προς τον σκοπό του ελέγχου και της παρακολούθησης του Supply Chain.

Τέλος, αναπτύσσεται μια νέα πλατφόρμα ηλεκτρονικής παρακολούθησης και εξαγωγής δεικτών του περιβάλλοντος για την υποστήριξη λήψης αποφάσεων που αφορούν την δρομολόγηση και χρήση των μεταφερόμενων προϊόντων. Πραγματοποιείται έλεγχος και δοκιμές αυτού του συστήματος για να διαπιστωθεί η αποτελεσματικότητα, η ακρίβεια και η χρησιμότητα του.

Η εργασία χωρίζεται σε κεφάλαια τα οποία έχουν ως εξής:

Η ενότητα 1 είναι η εισαγωγή. Στην Ενότητα 2, μελετάται η σχετική εργασία και βιβλιογραφία σχετικά με τα συστήματα ψυχρών αλυσίδων, το σχεδιασμό πειραμάτων και πιλοτικών συστημάτων, το πλαίσιο και τις εξελίξεις του IoT και άλλων τεχνολογιών και τις προσεγγίσεις βελτιστοποίησης στο πρόβλημα απώλειας προϊόντων κατά την μεταφορά. Η ενότητα 3 παρουσιάζει την αρχιτεκτονική συστήματος. Μια μελέτη περίπτωσης για την εφαρμογή του προτεινόμενου συστήματος και την απεικόνιση της κατάστασης περιβάλλοντος μέσω αποστολής δεδομένων σε server απεικονίζεται στην Ενότητα 4. Η Ενότητα 5 παρέχει τα αποτελέσματα των ελέγχων και των δοκιμών του συστήματος και την παρουσίαση μιας μελέτης περίπτωσης. Στην Ενότητα 6 αναφέρεται ο ιστότοπος που μπορεί οποιοςδήποτε να δει τα αποτελέσματα στο cloud από την δοκιμή του συστήματος ενώ στην Ενότητα 7 παρουσιάζονται ιδέες και προοπτικές για μελλοντική έρευνα και επέκταση του δημιουργηθέντος συστήματος. Τα συμπεράσματα εξάγονται στην Ενότητα 8.

2. ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΑΝΑΣΚΟΠΗΣΗ

Οι Jia-Wei Han, Min Zuo, Wen-Ying Zhu, Jin-Hua Zuo, En-Li Lü, Xin-Ting Yang(2021) εστιάζουν την έρευνα τους στις επιπτώσεις των περιβαλλοντικών παραγόντων στην ποιότητα των αγροτικών προϊόντων στην Κίνα και την κακή χρήση της ψυχρής αλυσίδας σε σχέση με τις ανεπτυγμένες χώρες προτείνοντας ως λύση κυβερνητικές ρυθμίσεις και οικονομικές παρεμβάσεις για την ανάπτυξη επιχειρήσεων και υπηρεσιών CCL και την συνεργασία μεταξύ εθνικών φορέων, πανεπιστημίων, βιομηχανίας και επιστημονικών εμπειρογνομώνων για την ζωτικής σημασίας σταθερή ανάπτυξη των CCL τεχνολογιών. Οι Bouzembrak, Klüche, Gavai, Marvin(2019) διεξήγαγαν μια βιβλιογραφική ανασκόπηση σχετικά με τις λύσεις IoT στον τομέα των τροφίμων η οποία έδειξε ότι το IoT στην ασφάλεια των τροφίμων αναφέρθηκε για πρώτη φορά το 2011 και από τότε αυξήθηκε κατακόρυφα με έντονη παρουσία των κινεζικών πανεπιστημίων αλλά και ότι οι δημοσιεύσεις εστιάζουν στον σχεδιασμό αρχιτεκτονικών IoT που χρησιμοποιούνται κυρίως πιλοτικά και όχι στην πράξη με την πλειοψηφία να χρησιμοποιεί αισθητήρες περιβάλλοντος και τεχνολογίες επικοινωνίας RFID και WSN.

Στην κακή χρήση της ψυχρής αλυσίδας εστιάζουν και οι Kumar, Neeraj & Tyagi, Mohit & Sachdeva, Anish(2021), οι οποίοι αναλύουν πως η υψηλή κατανάλωση ενέργειας της ψυχρής αλυσίδας οδηγεί σε υψηλότερο ποσοστό εκπομπών αερίων θερμοκηπίου αυξάνοντας το φαινόμενο της υπερθέρμανσης του πλανήτη αλλά και σε υψηλό λειτουργικό κόστος και έτσι την καθιερώνει ως το πιο σημαντικό κριτήριο για την βελτίωση της απόδοσης της ψυχρής αλυσίδας ενώ αποδεικνύουν πως η βελτίωση επιτυγχάνεται με την εφαρμογή των Παθητικών Συσκευών Ψύξης (PCDs) και το IoT. Πάνω στο θέμα της κατανάλωσης ενέργειας οι Karani, Krishna Prasad & Muraleedhar, Rajeshwari & Kirlaya, Vinayachandra & Poornima K, Geetha(2020) εξηγούν την ενσωμάτωση του IoT στην ψυχρή αλυσίδα που αξιοποιεί ηλιακή ενέργεια και τις προοπτικές ανάπτυξης που υπάρχουν παρουσιάζοντας ένα θεωρητικό εννοιολογικό μοντέλο διαχείρισης ολοκληρωμένης ηλιακής ψυχρής αλυσίδας.

Οι Aamer, Ammar & Al-Awlaqi, Mohammed & Affia, Ifadhila & Arumsari, Silvia & Mandahawi, Nabeel(2021) διερεύνησαν και εντόπισαν τις βασικές προκλήσεις που σχετίζονται με την υιοθέτηση του IoT στην ψυχρή αλυσίδα τροφίμων χρησιμοποιώντας τη δημοσιευμένη βιβλιογραφία από το 2010 έως το 2020, αντιμετωπίζοντας το κενό της μη διαθεσιμότητας δεδομένων και την ασυνέπεια στη μέτρηση και την παρακολούθηση της απώλειας τροφίμων και συνέβαλαν στο σύνολο των γνώσεων καθιερώνοντας ένα πολυεπίπεδο πλαίσιο που συνδέει τις προκλήσεις με διάφορα μέρη της ψυχρής αλυσίδας με σκοπό την αποτελεσματικότερη βελτίωση της. Για να πραγματοποιηθούν οι απαραίτητες βελτιώσεις είναι σημαντικό να διερευνηθούν τα περιθώρια κερδών σε κάθε μεγάλη αλλαγή και σε αυτό το πλαίσιο οι Sun, Lu & Zhao, Yuanjun & Sun, Wenqi & Liu, Zhengkai(2020) παρέχουν μια θεωρητική βάση για τη λήψη επενδυτικών αποφάσεων των επιχειρήσεων στο IoT. Αναλύονται η μείωση χρόνου μεταφοράς της ψυχρής αλυσίδας, το κόστος μεταφοράς, ο ρυθμός διατήρησης της ποιότητας των προϊόντων, δημιουργώντας ένα μοντέλο κόστους-κέρδους πριν και μετά την υιοθέτηση του IoT το οποίο επιτρέπει σε όλους τους κύριους φορείς της εφοδιαστικής αλυσίδας σε διαφορετικά επίπεδα να αποκομίσουν κέρδη. Οι Shang, Zhang, Zhu and Zhou(2014) προτείνουν τρεις αρχές για το σχεδιασμό υπηρεσιών IoT – ταξινόμηση υπηρεσιών, συντονισμός και συμβατότητα – για να βοηθήσουν στο

σχεδιασμό ενός συστήματος υπηρεσιών IoT. Προτείνουν επίσης ένα μοντέλο για την κατανομή πόρων προκειμένου να βελτιωθεί η απόδοση της υπηρεσίας IoT και μια εφαρμογή που περιλαμβάνει μια πλατφόρμα ανίχνευσης ψυχρής αλυσίδας για την επεξήγηση της έρευνας. Ο Cui(2021) ανέλυσε και μελέτησε εμπειρικά τον αντίκτυπο των βασικών τεχνολογιών του IoT στον συντονισμό ψυχρής αλυσίδας από την προοπτική της οικοδόμησης ενός ευφυούς συστήματος συντονισμού. Εξηγεί πως η τεχνολογία IoT μπορεί να βελτιώσει το επίπεδο “νοημοσύνης” της ψυχρής αλυσίδας συνδυάζοντας τα logistics της ψυχρής αλυσίδας με την αρχιτεκτονική τριών επιπέδων της τεχνολογίας IoT, δηλαδή το επίπεδο IoT, το επίπεδο δικτύου και το επίπεδο νοημοσύνης.

Πέρα από την συμβολή των IoT τεχνολογιών στον τομέα των τροφίμων και των επιχειρήσεων, το μεγαλύτερο ίσως στοίχημα αυτής της τεχνολογίας είναι η συμβολή της στον ιατροφαρμακευτικό τομέα, ειδικότερα στην εποχή του COVID-19 που διανύουμε. Οι Hasanat, Raisa & Rahman, MD & Mansoor, Nafees & Mohammed, Nabeel & Rahman, Shahriar & Rasheduzzaman, Mirza(2020) αναλύουν τα προβλήματα στην διανομή των εμβολίων και μελετούν ένα προϋπάρχον σύστημα παρακολούθησης που διασφαλίζει ότι η ψυχρή αλυσίδα του εμβολίου παρακολουθείται συνεχώς κατά τη διάρκεια ολόκληρης της διαδικασίας μεταφοράς στα κέντρα υγείας και καθώς το σύστημα είναι πλήρως αυτοματοποιημένο, διασφαλίζει διαφάνεια και αποτελεσματικότητα σε όλη τη διαδικασία. Αντίστοιχα, στην έρευνα των Monteleone S., Sampraio M. and Maia R. F.(2017) προτείνεται ένα θεωρητικό, εννοιολογικό μοντέλο ψυχρής αλυσίδας για παρακολούθηση της θερμοκρασίας με προοπτική την βελτίωση της πρόληψης, την διόρθωση και την συντήρηση των ήδη υπαρχόντων συστημάτων και με σκοπό την μελλοντική πιλοτική εφαρμογή του σε μεταφορές φαρμάκων και εμβολίων και την διευκόλυνση λήψης αποφάσεων σε εταιρείες και ιδρύματα σχετικά με την εφοδιαστική αλυσίδα και την ποιότητα των προϊόντων. Ο Rusnack(2021) αναλύει τα πρότυπα παρακολούθησης και τον εξοπλισμό που χρησιμοποιούνται στην ψυχρή αλυσίδα του VFC προγράμματος της Αμερικής που αφορά την παροχή εμβολίων σε παιδιά που δεν έχουν ασφάλιση υγείας ή οικονομικές δυνατότητες και πως μπορεί να είναι επαρκή για τέτοιου είδους προγράμματα αλλά δεν είναι το ίδιο για το εμβόλιο κατά του COVID. Η λύση που περιγράφεται έχει παραλληλισμούς με το πρόγραμμα VFC ενώ προτείνει την υιοθέτηση προηγμένων τεχνολογιών του 2021. Οι Lawrence, Phippard, Ramachandran and Hughes(P.W. Lawrence et al. 2018) αναλύουν την εφαρμογή μιας αναδυόμενης τεχνολογίας δικτύωσης, του δικτύου LoRa Wide Area (LoRaWAN) για ιατρική επιτήρηση ψυχρής αλυσίδας σε περιβάλλον με αναξιόπιστο ηλεκτρισμό και κατανεμημένο σύστημα υγείας, το οποίο ταιριάζει ιδανικά σε περιβάλλοντα με περιορισμένους πόρους όπως αναπτυσσόμενων χωρών λόγω του χαμηλού κόστους, της χαμηλής χρήσης ενέργειας και της μεγάλης εμβέλειας. Περιγράφεται η υλοποίηση ενός πιλοτικού έργου στο Kikwit της Α.Δ. του Κονγκό για την ανάπτυξη ενός ασύρματου δικτύου που βασίζεται σε LoRa για την παρακολούθηση της θερμοκρασίας των προϊόντων αίματος, διασφαλίζοντας την ασφάλεια και τη βιωσιμότητά τους μέσω ενός αποκεντρωμένου συστήματος παρακολούθησης χαμηλής κατανάλωσης και χαμηλού κόστους.

Το IoT αποτελεί αναμφίβολα μια τεχνολογία του μέλλοντος και ήδη πραγματοποιούνται πολλές έρευνες για την σύνδεση του με άλλες ραγδαία αναπτυσσόμενες τεχνολογίες που απασχολούν σε μεγάλο βαθμό τις επιχειρήσεις και τον ακαδημαϊκό κόσμο όπως είναι η τεχνητή νοημοσύνη, η μηχανική μάθηση(machine learning) και το blockchain. Οι Loisel, Julie & Duret, Steven & Cornuéjols, Antoine & Cagnon, Dominique & Tardet, Margot & Derens-Bertheau, Evelyne & Laguerre, Onrawee(2021) πραγματοποιούν μια ανασκόπηση η

οποία σκοπεύει να προσδιορίσει πώς μπορεί να χρησιμοποιηθεί η τεχνητή νοημοσύνη για την ανίχνευση βλαβών της ψυχρής αλυσίδας και περιγράφονται μέθοδοι μηχανικής μάθησης που εφαρμόζονται στις ψυχρές αλυσίδες προκειμένου να επισημανθούν τα ζητήματα που σχετίζονται με αυτά τα δεδομένα. Στον τομέα του blockchain, οι Zhang, Liu, Jiong, Zhang, Li and Chen(2021) προτείνουν λύσεις στην διαχείριση ιχνηλασιμότητας της ψυχρής αλυσίδας μέσω της ενσωμάτωσης του blockchain και του IoT για τη βελτίωση διαχείρισης των εύκολα αλλοιώσιμων τροφίμων υδατοκαλλιέργειας με αξιόπιστες εφαρμογές. Χρησιμοποιώντας την τεχνολογία IoT, το blockchain διαθέτει όλες τις λειτουργίες ανίχνευσης που καλύπτουν κάθε σύνδεσμο στα logistics υδατοκαλλιέργειας, και έτσι υλοποιείται μια ενοποιημένη εποπτεία ανίχνευσης και αξιόπιστη διαχείριση. Όσον αφορά τον ιατροφαρμακευτικό τομέα, η έρευνα των Hosseini Bamakan, Seyed Mojtaba & Ghasemzadeh, Shima & Manshadi, Sajedeh(2021) έχει ως κύριο στόχο να εξετάσουν τον τρόπο με τον οποίο το blockchain θα ανταποκρίνεται στις απαιτήσεις μιας φαρμακευτικής ψυχρής αλυσίδας, όπως η φαρμακευτική ψηφιακή ταυτότητα, η σειριοποίηση και η ιχνηλασιμότητα, η ακεραιότητα των δεδομένων, η διαφάνεια και η διαχείριση αποβλήτων και αναλύονται μελέτες περιπτώσεων φαρμακευτικών έργων που βασίζονται σε blockchain καταγράφοντας τα πλεονεκτήματα τους αλλά και τις προκλήσεις που πρέπει να ξεπεραστούν. Σχετικά με τα προβλήματα που επιδείνωσε η πανδημία του COVID-19 στον τομέα των τροφίμων, οι Iftekhar, Adnan & Cui, Xiaohui(2021) παρουσιάζουν μια αρχιτεκτονική αλυσίδας εφοδιασμού με χρήση τεχνολογίας blockchain ώστε να διασφαλίζεται ο έλεγχος και η ελαχιστοποίηση του κινδύνου παρουσίας COVID-19 και άλλων βακτηρίων, μυκήτων και παρασίτων στην αλυσίδα εφοδιασμού κατεψυγμένου κρέατος. Βελτιώνει τη διαφάνεια σε όλη την αλυσίδα εφοδιασμού και συμβάλλει στην τεκμηρίωση που απαιτούν οι νομοθετικές αρχές για την εισαγωγή προϊόντων ψυχρής αλυσίδας και την πιστοποίηση της ποιότητας του τελικού προϊόντος.

Στο τεχνικό κομμάτι, οι Ping, Wang, Ma, Du(2018) κάνουν μια εισαγωγή στην έννοια της τεχνολογίας IoT και τις βασικές τεχνολογίες που χρησιμοποιούνται επί του παρόντος στην παρακολούθηση της ποιότητας και της ασφάλειας των γεωργικών προϊόντων στην παραγωγή, μεταποίηση, κυκλοφορία, πωλήσεις και ιχνηλασιμότητα και αναλύουν τις τρέχουσες προκλήσεις και τάσεις στο μέλλον. Οι Onwude, Daniel & Chen, Guangnan & Eke-Emezie, Nnanna & Kabutey, Abraham & Alkhaled, Alfadhl & Sturm, Barbara(2020) διερευνούν τρόπους για το πώς οι απώλειες τροφίμων μπορούν να μειωθούν περαιτέρω στην αλυσίδα εφοδιασμού φρούτων και λαχανικών χρησιμοποιώντας την τρέχουσα τεχνολογία αιχμής στην παρακολούθηση και τη βελτιστοποίηση συστημάτων ψυχρής αλυσίδας αναλύοντας τις δυνατότητες εφαρμογής αναδυόμενων τεχνολογιών όπως το IoT και τα digital twins. Για τις ρυθμίσεις συσκευασίας και το πρόβλημα δρομολόγησης οχημάτων προκειμένου να δημιουργηθούν οδηγίες συσκευασίας, αυτοματοποιημένη παράδοση και παρακολούθηση των προϊόντων σε πραγματικό χρόνο, οι Tsang, Y. P. & Choy, K.L. & Wu, Chun-Ho & Ho, G.T.s & Lam, H. & Tang, Valerie(2018) προτείνουν ένα σύστημα σχεδιασμού διαδρομής που βασίζεται στο IoT (IRPS) το οποίο ενσωματώνει τεχνολογίες IoT(πχ αισθητήρες SensorTag CC3200) και τη μέθοδο Taguchi. Με τη διεξαγωγή μιας μελέτης περίπτωσης, το προτεινόμενο σύστημα επιβεβαίωσε την μείωση του ποσοστού αλλοίωσης των τροφίμων και τη βελτίωση της ικανοποίησης των πελατών, καθώς και της λειτουργικής αποτελεσματικότητας. Η ομάδα με την ίδια περίπτωση σύνθεση, αποτελούμενη από τους Tsang, Y. P. & Choy, KL & Wu, Chun-Ho & Ho, G.T.s & Lam, H. & Koo, PS.(2017), προτείνει ένα σύστημα παρακολούθησης φορτίου που βασίζεται στο IoT για παρακολούθηση περιβαλλοντικών αλλαγών των ευαίσθητων προϊόντων προκειμένου να διασφαλιστεί η ποιότητα τους σε όλη την έκταση της ψυχρής αλυσίδας ενώ παράλληλα μέσω της

διεξαγωγής της μελέτης περίπτωσης σε μια εταιρεία 3PL(εταιρείες logistics τρίτων) στο Χονγκ Κονγκ, αξιολογεί την απόδοση στην ικανοποίηση των πελατών ως θετική μετά την υιοθέτηση του συστήματος. Παρομοίως, ο Yu Cui(2018) προτείνει μια αρχιτεκτονική συστήματος παρακολούθησης ψυχρής αλυσίδας χρησιμοποιώντας IoT τεχνολογίες και διεξάγει μια μελέτη περίπτωσης για την απεικόνιση της διαθεσιμότητας και της ευρωστίας του συστήματος στον τομέα προμήθειας τροφίμων. Στην έρευνα των Guillén, Antonio & Boada, Martí & Villarino, R. & Girbau, David(2019) εξετάζονται οι πρόσφατες εξελίξεις στις τεχνολογίες αισθητήρων που βασίζονται σε επικοινωνία κοντινού πεδίου (NFC) που επιτρέπει τη γρήγορη επικοινωνία μεταξύ συσκευών σε μικρή εμβέλεια χρησιμοποιώντας τη ζώνη RFID 13,56 MHz. Παρέχονται πρακτικές σκέψεις για το σχεδιασμό αισθητήρων που βασίζονται σε NFC και αναλύεται πως το RFID είναι μια από τις πιο σημαντικές τεχνολογίες για τη μαζική ανάπτυξη του IoT. Οι D. Bogataj, M. Bogataj, Hudoklin(2017) περιγράφουν πώς μπορεί να χρησιμοποιηθεί η θεωρία EMRP(extended material requirements planning) για την εκτίμηση των αλλαγών στο NPV(net present value) όταν περιλαμβάνονται στο μοντέλο κυψέλες κινούμενης δραστηριότητας(moving activity cells) και πως ένα τέτοιο σύστημα βελτιώνει το NPV με την ανάπτυξη διαδικασιών λήψης αποφάσεων σε πραγματικό χρόνο στο cloud, χρησιμοποιώντας το IoT με αισθητήρα που παρακολουθεί τις περιβαλλοντικές συνθήκες ως υποδομή και συμπεριλαμβάνοντας την αυτόματη αναδρομολόγηση στα logistics μετά τη συγκομιδή.

Πολλές ακαδημαϊκές έρευνες έχουν αναπτύξει δικά τους συστήματα IoT παρακολούθησης χρησιμοποιώντας πληθώρα τεχνολογιών και προσεγγίσεων τα οποία είναι πλήρως λειτουργικά. Αυτά τα συστήματα έχουν δοκιμαστεί στην πράξη και ολοένα και περισσότερες προσπάθειες και ιδέες προκύπτουν από ομάδες ανθρώπων. Οι Ruan, Shi(2015) παρουσιάζουν ένα σύστημα βασισμένο στο IOT με τεχνολογίες όπως GPS, πομποδέκτες RFID που συχνά ονομάζονται ετικέτες RFID και ασύρματα δίκτυα αισθητήρων για την παρακολούθηση των παραδόσεων ηλεκτρονικού εμπορίου φρούτων και προτείνουν μια προσέγγιση βασισμένη σε σενάρια για την αξιολόγηση της κατάστασης των φρούτων στην ψυχρή αλυσίδα παρουσιάζοντας τα πειραματικά αποτελέσματα που δείχνουν την αποτελεσματικότητα και τα πλεονεκτήματα του συστήματος. Ο Dae-Sik Ko(2016) σχεδιάζει και αναλύει τις διαδικασίες και την ροή δεδομένων με το CoAP πρωτόκολλο επικοινωνίας του IETF CoRE WG(Internet Engineering Task Force Constrained RESTful Environments Working Group) μεταξύ της IOT συσκευής και του middleware server ενός IOT Cold Chain Monitoring System. Ως συσκευές IOT χρησιμοποιεί αισθητήρες θερμοκρασίας και υγρασίας, για την επικοινωνία με την gateway συσκευή χρησιμοποιήθηκαν τεχνολογίες Zigbee, Bluetooth και RFID ενώ για την επικοινωνία με τον server χρησιμοποιήθηκαν Ethernet, WiFi, LTE και CDMA. Τελικώς, ερμηνεύει τα αποτελέσματα σε πραγματικό χρόνο αφού τρέξει μια διαδικασία επιβεβαίωσης παραλαβής δεδομένων από τον server όπου χρειάζεται. Οι Urbano, Oscar & Perles, Angel & Pedraza, Cesar & Rubio-Arreaez, Susana & Castelló, M. & Ortola, María & Mercado, Ricardo(2020) σχεδίασαν μια εφαρμογή ενός οικονομικά αποδοτικού συστήματος ιχνηλασιμότητας, βασισμένου στην τεχνολογία αναγνώρισης ραδιοσυχνότητας (RFID) με αισθητήρες θερμοκρασίας και στο IoT μέσω της σύνδεσης τους στην ίδια πλατφόρμα που προορίζεται να αντιμετωπίσει τα προβλήματα διασύνδεσης και κόστους εκτελώντας 2 πειράματα: το ένα εκτελείται σε ελεγχόμενες εργαστηριακές συνθήκες, παρακολουθώντας ψιλοκομμένη κολοκύθα και το άλλο σε πραγματικό σενάριο, παρακολουθεί πορτοκάλια που αποστέλλονται από τη Βαλένθια της Ισπανίας στο Κορκ της Ιρλανδίας. Οι Ko, Kwak, Choi, Song(2015) σχεδίασαν ένα σύστημα ψυχρής αλυσίδας ως PaaS (Platform as a service) και IaaS (Infrastructure as a service) με βάση το Cloud και το

IoT που επιτρέπει στους χρήστες να αναπτύξουν και να εφαρμόσουν συστήματα διαχείρισης ψυχρής αλυσίδας με χαμηλό κόστος και σε σύντομο χρονικό διάστημα και να χρησιμοποιήσουν κάθε τύπο συσκευών IoT, π.χ. ετικέτες RFID, κόμβους αισθητήρων WSN, κόμβους αισθητήρων BLE(Bluetooth Low Energy). Για τις απαιτήσεις παροχής μεγάλης ποσότητας ροής δεδομένων από πολλαπλές πηγές σε πραγματικό χρόνο, η ψυχρή αλυσίδα σχεδιάστηκε με βάση το Hadoop, το Spark και το Spark Streaming. Ο Guie Li(2021) μελέτησε πως η χρήση της τεχνολογίας προγνωστικής μετάδοσης δεδομένων (PDTT) και τα προηγμένα μοντέλα IoT στη μεταφορά προϊόντων μέσω ψυχρή αλυσίδα μπορούν να διατηρήσουν τα φρούτα και τα λαχανικά φρέσκα στο μέγιστο βαθμό και να μειώσουν το κόστος στη διαδικασία μεταφοράς. Χρησιμοποιούνται GPS και 5G ως συστήματα επικοινωνίας ενώ το σύστημα ελέγχου Wireless Systems Manager(WSM) υλοποιείται με την ενσωμάτωση κυρίως ενός ασύρματου μικροελεγκτή FPGA που βασίζεται στο λογισμικό Xilinx. Οι Mohsin, Yellampalli(2017) επιδιώκουν να καλύψουν τα κενά που αφορούν την παρακολούθηση τοποθεσίας και τον έλεγχο του περιβάλλοντος στον φαρμακευτικό τομέα μέσω μιας μονάδας GPS και μιας πλατφόρμας αισθητήρων που βασίζεται στο IOT αντίστοιχα. Στο σύστημα που προτείνουν τα δεδομένα του αισθητήρα αποστέλλονται μέσω πομπού Wi-Fi με το άκρο λήψης να περιλαμβάνει τον υπεύθυνο παρακολούθησης ο οποίος θα λαμβάνει τακτικές ενημερώσεις σχετικά με τις παραμέτρους του ελεγχόμενου περιβάλλοντος. Ο Xiao(2020) σχεδιάζει και υλοποιεί σύστημα παρακολούθησης για τα logistics ψυχρής αλυσίδας που βασίζεται στις πλατφόρμες IoT και GPS κάνοντας το δίκτυο μεταφοράς αποτελεσματικό και χαμηλού κόστους, διατηρώντας παράλληλα τη μέγιστη φρεσκάδα των ευπαθών προϊόντων ενώ οι Kramar, Vadim & Maatta, Harri & Hinkula, Henry & Thorsen, Oistein & Cox, Georgina(2017) εισάγουν την αρχιτεκτονική του συστήματος Smart-Fish για μεταφορά φρέσκων ψαριών που βασίζεται σε αισθητήρα θερμοκρασίας και τεχνολογία GPS και προσφέρει δυνατότητα λήψης δεδομένων σε πραγματικό χρόνο για τον έλεγχο της θερμοκρασίας και δυνατότητα λήψης αποφάσεων με καλύτερη ενημέρωση για την διατήρηση της ποιότητας και την μείωση των αποβλήτων. Οι Luo, Heng & Zhu, Minjie & Ye, Sengang & Hou, Hanping & Chen, Yong & Bulysheva, Larisa(2016) προτείνουν ένα σύστημα παρακολούθησης IoT για αποτελεσματική και γρήγορη ζωντανή παρακολούθηση των αγαθών στην ψυχρή αλυσίδα με το χαμηλότερο κόστος, μεγαλύτερη χωρητικότητα δικτύου και απλούστερα πρωτόκολλα. Το βασικό μέρος σε αυτό το προτεινόμενο σύστημα είναι ένα ασύρματο δίκτυο αισθητήρων που βασίζεται στο Zigbee. Οι Chandra, Abel & Lee, Seong. (2014) περιγράφουν την δημιουργία ενός συστήματος Cold Chain Monitoring με την χρήση Arduino hardware και ZigBee επικοινωνίας ώστε να πετύχει μικρότερη κατανάλωση ενέργειας αλλά και μικρότερο κόστος και πολυπλοκότητα από τα αντίστοιχα WiFi και Bluetooth. Στην συνέχεια χρησιμοποιεί την Xively sensor cloud πλατφόρμα για την καλύτερη οπτικοποίηση και έλεγχο των δεδομένων των αισθητήρων αλλά και της κατανάλωσης ενέργειας και μπαταρίας ενώ μέσω της πλατφόρμας δημιουργεί ειδοποιήσεις μέσω email σε κάθε τιμή δεδομένων που ξεφεύγει από τα καθορισμένα. Οι Afreen, Hina & Bajwa, Imran(2021) μελετούν και υλοποιούν ένα σύστημα IoT παρακολούθησης και ειδοποίησης για την ποιότητα φρούτων και λαχανικών που εμφανίζουν μεγάλα ποσοστά απωλειών κατά την μεταφορά, εστιάζοντας και σε περιβαλλοντικές παραμέτρους όπως η φωτεινότητα, η συγκέντρωση αερίων και η σκόνη αντί μόνο για τις συνήθειες όπως θερμοκρασία και υγρασία. Χρησιμοποιώντας το μοντέλο πρόβλεψης Τεχνητών Νευρωνικών Δικτύων(ANN) επιτεύχθηκε ακρίβεια 99% ενώ υπάρχοντα μοντέλα όπως αποστολή συμπίεσης (CS), Adaptive Naïve Bayes (ANB), Extreme Gradient Boosting (XGBoost), Εξόρυξη Δεδομένων (DM) πέτυχαν ακρίβεια 95,60%, 87,50%, 93,59%, 90% αντίστοιχα. Παρόμοια, στην έρευνα των Lorenc, Augustyn & Czuba, Michał & Szarata, Jakub. (2021) η αυτόματη ανάλυση Big Data και η

μαθηματική μοντελοποίηση χρησιμοποιήθηκαν για τον εντοπισμό προβλημάτων στην ψυχρή αλυσίδα. Το Τεχνητό Νευρωνικό Δίκτυο (ANN) χρησιμοποιήθηκε για την πρόβλεψη πιθανών διαταραχών που σχετίζονται με τη θερμοκρασία στη μεταφορά. Η έρευνα αποδεικνύει ότι είναι δυνατό να αποτραπεί πάνω από το 82% των διαταραχών στην ψυχρή αλυσίδα καθώς το ANN επιτρέπει αναλύσεις της καμπύλης θερμοκρασίας και πρόβλεψη της διαταραχής πριν αυτή συμβεί. Έτσι ο συνδυασμός αισθητήρων IoT για τη συλλογή δεδομένων και η ανάλυση Big Data και το ANN εξασφαλίζει την ακεραιότητα της ψυχρής αλυσίδας. Οι Sergi, Ilaria & Montanaro, Teodoro & Benvenuto, Fabrizio & Patrono, Luigi(2021) παρουσιάζουν ένα σύστημα διαχείρισης ψυχρής αλυσίδας που περιλαμβάνει συσκευές IoT με πιστοποίηση Azure Sphere και υπηρεσίες Azure Cloud, με ιδιαίτερη έμφαση στην εξασφάλιση ασφάλειας από άκρο σε άκρο. Ειδικότερα, η δουλειά τους επικεντρώνεται στη συλλογή δεδομένων σε πραγματικό χρόνο, στην τοπική επεξεργασία δεδομένων και στην αποστολή δεδομένων σε μια ασφαλή πλατφόρμα cloud. Καταλήγοντας, αναφέρουν ότι η διαδικασία μπορεί να βελτιωθεί περαιτέρω σε μελλοντικές εργασίες αξιοποιώντας άλλες υπηρεσίες που παρέχονται από το Microsoft Azure όπως για παράδειγμα, τις δυνατότητες Azure για τεχνητή νοημοσύνη. Από τους Tsang, Y. P. & Choy, K.L. & Wu, Chun-Ho & Ho, G.T.s & Lam, H. & Koo, P.S.(2018) προτείνεται ένα σύστημα που βασίζεται στο IoT ενσωματώνοντας ως κόμβους αισθητήρων τα SensorTag CC2650 και Microsoft band 2 για τη συλλογή δεδομένων περιβάλλοντος και υγείας σε πραγματικό χρόνο και το IBM Bluemix για τη δημιουργία της εφαρμογής που θα ελέγχει τα δεδομένα, με σκοπό να καλύψει το ερευνητικό κενό της αναποτελεσματικής παρακολούθησης και διαχείρισης κινδύνου για την ποιότητα των προϊόντων αλλά και την επαγγελματική ασφάλεια στις ψυχρές αλυσίδες. Οι Mourtzis, Angelopoulos, Panopoulos(2021) παρουσίασαν την έρευνα τους στη σχεδίαση και την ανάπτυξη ενός πλαισίου για την απομακρυσμένη παρακολούθηση της ψυχρής αλυσίδας με βάση την υλοποίηση ενός ασύρματου δικτύου αισθητήρων (WSN) και τεχνολογίας Cloud για την απόκτηση δεδομένων και ευφυών αλγορίθμων για προγνωστική συντήρηση(Predictive Maintenance). Οι Li, Yang, Duan, Wu and Zhang(2019) ανέπτυξαν και δοκίμασαν ένα σύστημα παρακολούθησης θερμοκρασίας, υγρασίας, οξυγόνου και διοξειδίου του άνθρακα στην ψυχρή αλυσίδα σε πραγματικό χρόνο χρησιμοποιώντας τεχνολογίες IoT για την παρακολούθηση μεταφοράς του T. matsutake, ενός ειδικού τύπου μύκητα γνωστού ως «ο βασιλιάς των βακτηρίων» που έχει πολύ υψηλή οικονομική αξία για την αγροτική περιοχή στη νοτιοδυτική Κίνα και είναι πολύ δύσκολη η μεταφορά λόγω της εύκολης αλλοίωσης του. Η μελέτη των Shih, Wang(2015) ανέπτυξε ένα καινοτόμο σύστημα ένδειξης χρόνου-θερμοκρασίας ψυχρής αλυσίδας (TTI) για ένα πολυκαναλικό κινεζικό σύστημα επεξεργασίας τροφίμων, το οποίο χρησιμοποιεί ασύρματους αισθητήρες για τη συλλογή δεδομένων θερμοκρασίας και τη διαμόρφωση κριτηρίων κρίσιμου σημείου ελέγχου(CCP). Στη συνέχεια, διαμορφώνονται χάρτες ελέγχου X-R για την παρακολούθηση κάθε σημείου της διαδικασίας. Αυτή η προσέγγιση βασίζεται σε μια αρχιτεκτονική IoT και ένα διεθνές πρότυπο τροφίμων ISO 22.000. Ειδικότερα, η προτεινόμενη μεθοδολογία εφαρμόστηκε σε ένα διάσημο franchise τροφίμων στην Ταϊβάν για να αξιολογηθεί η αποτελεσματικότητά του. Η προτεινόμενη λύση IoT χρησιμοποιεί αισθητήρες για τη συλλογή δεδομένων θερμοκρασίας σε κάθε σημείο της διαδικασίας για τα μεταγενέστερα κριτήρια CCP του προτύπου ISO 22000/HACCP και έτσι η ανάλυση των συλλεγόμενων δεδομένων διευκολύνει τη διαδικασία λήψης αποφάσεων σχετικά με την αποθήκευση. Οι Salunkhe, Nerkar(2016) δημιουργούν σύστημα που βασίζεται στο IoT παρακολουθεί το περιβάλλον μέσα στα φορτηγά της ψυχρής αλυσίδας, όπως τη θερμοκρασία και την υγρασία για υλικά όπως τα ψάρια και τα φαρμακευτικά προϊόντα. Χρησιμοποιείται ο μικροελεγκτής Arduino Uno για ένα οικονομικό και αξιόπιστο σύστημα παρακολούθησης φορτηγών με LM35D ως αισθητήρα θερμοκρασίας,

DHT11 ως αισθητήρα υγρασίας, αισθητήρα LDR, διακόπτη Reed, Optocoupler και αισθητήρα PIR. Η πλατφόρμα cloud Osmosis της Axelta αναλύει τα δεδομένα που συλλέγονται και σε οποιαδήποτε παρέκκλιση δημιουργεί συναγερμό. Η Angeles(2016) αναλύει το σύστημα PharmaChain της διεθνούς εφοδιαστικής Kuehne + Nagel για αποστολές φαρμακευτικών και βιολογικών προϊόντων ψυχρής αλυσίδας με χρήση IoT για παρακολούθηση θερμοκρασίας και εφαρμόζει το δομικό μοντέλο τεχνολογίας του Orlikowski προσφέροντας μεγαλύτερη κατανόηση της ανάπτυξης του PharmaChain χρησιμοποιώντας αισθητήρες U. Ramírez-Faz, J. & Fernández-Ahumada, Luis & Fernández-Ahumada, Elvira & López-Luque, Rafael(2020) παρουσιάζουν ένα χαμηλού κόστους σύστημα IoT, βασισμένο σε δωρεάν υλικό και λογισμικό, για την παρακολούθηση της θερμοκρασίας στην τροφική ψυχρή αλυσίδα, προτείνοντάς το για δημοσίευση ανοιχτής πρόσβασης και αναπτύσσοντάς το ως λογισμικό και υλικό ανοιχτού κώδικα. Συγκεκριμένα, προτείνεται η χρήση του μικροελεγκτή ESP-8266-Wi-Fi με αισθητήρες θερμοκρασίας DS18B20 και η πλατφόρμα ThingSpeak IoT για την αποθήκευση και επεξεργασία δεδομένων στο cloud. Οι Zhang, Liu(2019) προτείνουν ένα έξυπνο σύστημα ψυχρής αλυσίδας που βασίζεται στην τεχνολογία επικοινωνίας NB-IoT(NarrowBand-IoT) παρουσιάζοντας το σχήμα συστήματος, τη σχεδίαση υλικού και λογισμικού και σχεδιασμό χαμηλής ισχύος. Το σύστημα μεταφοράς χρησιμοποιεί το STM32 ως κύριο ελεγκτή, ενσωματώνει αισθητήρες θερμοκρασίας και υγρασίας, έντασης φωτός και πληροφορίας θέσης. Οι Bengionanni, Teresa & Rosito, Roberto & Lacasa, Vincenzo & Simone, Eugenio & Sergi, Ilaria & Iacovone, Vincenzo & Viggiano, Michele & Patrono, Luigi(2020) προτείνουν ένα πλήρες σύστημα υλικού και λογισμικού που παρακολουθεί συστήματα ψυγείων που μεταφέρουν ευαίσθητα στη θερμοκρασία βιολογικά δείγματα, φάρμακα ή εμβόλια. Το προτεινόμενο σύστημα βασίζεται σε τεχνολογίες IoT και υπηρεσίες cloud και παρέχει τόσο εργαλεία παρακολούθησης σε πραγματικό χρόνο όσο και ιστορικά δεδομένα αλλά και σύστημα συναγερμού που σχετίζεται με τη δυσλειτουργία των συστημάτων ψύξης.

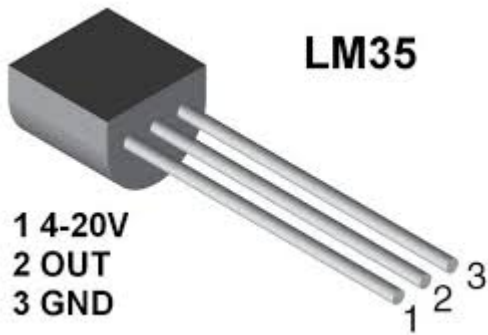
Κάνοντας ένα βήμα παραπάνω, πολλά πιλοτικά συστήματα που είναι λειτουργικά και έχουν σχεδιαστεί ως προϊόν ακαδημαϊκής έρευνας, έχουν συνδεθεί με την τεχνολογία του blockchain. Οι Sunny , Undralla , Pillai(2020) παρέχουν μια εικόνα σχετικά με τις δυνατότητες τεχνολογίας blockchain για τη διαφάνεια μιας ψυχρής αλυσίδας και αναλύουν την συνάφεια του IoT με τα smart contracts. Προκειμένου να καταδειχθεί πώς οι λύσεις ιχνηλασιμότητας blockchain βελτιώνουν τη διαφάνεια της εφοδιαστικής αλυσίδας, παρουσιάζεται ένα Proof of Concept (PoC) σενάριο ψυχρής αλυσίδας χρησιμοποιώντας το Microsoft Azure Blockchain Workbench για τον εντοπισμό και την παρακολούθηση της διαδρομής των φαρμάκων που είναι ευαίσθητα στη θερμοκρασία και την υγρασία. Οι Stach, Gritti, Przytarski, Mitschang(2020) παρουσίασαν το σύστημα SHEEPDOG που βασίζεται στο IoT και παρακολουθεί τα προϊόντα από την παραγωγή έως την αγορά εξασφαλίζοντας αξιόπιστη και ασφαλή παρακολούθηση των τροφίμων χρησιμοποιώντας τεχνολογίες blockchain που παρέχουν ασφαλή αποθήκευση δεδομένων και ακριβή έλεγχο πρόσβασης που διασφαλίζει την ανθεκτική και αμετάβλητη αρχειοθέτηση των δεδομένων. Οι Mendonça, Ronan & Gomes, Otávio & Vieira, Luiz & Vieira, Marcos & Borges, Alex & Miranda Nacif, José(2021) σχεδίασαν μια προσέγγιση ψυχρής αλυσίδας του εμβολίου που χρησιμοποιεί τεχνολογία IoT για ανίχνευση και blockchain για αποθήκευση που βοηθά στην καταπολέμηση της πανδημίας COVID-19. Συγκεκριμένα αισθητήρες IoT ανιχνεύουν την κατάσταση, τη θερμοκρασία και την τρέχουσα θέση του εμβολίου ενώ μια πλατφόρμα Ethereum με τη μορφή ενός ιδιωτικού δικτύου αποθηκεύει με ασφάλεια τα δεδομένα. Στο σύστημά τους, τα smart contracts είναι υπεύθυνα για τη διαχείριση των αισθητήρων IoT, των στοιχείων που

παρακολουθούνται και των δεδομένων που σχετίζονται με κάθε στοιχείο. Οι Baralla, Pinna, Tonelli, Marchesi, Ibba(2019) σχεδίασαν και ανέπτυξαν ένα σύστημα που βασίζεται σε blockchain που χρησιμοποιεί την πλατφόρμα Ethereum για τη διαχείριση της ψυχρής αλυσίδας αγροδιατροφικών προϊόντων, χρησιμοποιώντας smart contracts για διαφάνεια, αποτελεσματικότητα και αξιοπιστία ενώ συσκευές δικτύου IoT όπως αισθητήρες, παρέχουν δεδομένα παρακολούθησης όπως θερμοκρασία αποθήκευσης, υγρασία περιβάλλοντος και δεδομένα GPS. Το προτεινόμενο σύστημα έχει σχεδιαστεί σύμφωνα με τις μεθόδους Agile και ABCDE, για την απόκτηση υψηλότερης ποιότητας λογισμικού για το σχεδιασμό ενός γενικού συστήματος blockchain μέσω πρακτικών μηχανικής λογισμικού. Η παρουσία πιστοποιητικών επιτρέπει την αξιολόγηση της προέλευσης και της ακεραιότητας των τροφίμων σε ολόκληρη την αλυσίδα διανομής ενώ σε περίπτωση ανωμαλίας παρτίδας, θα είναι δυνατός ο εντοπισμός όλων των προγόνων της παρτίδας ή των διαδόχων της. Ο τελικός καταναλωτής μπορεί να είναι σίγουρος ότι το αγορασμένο προϊόν είναι πρωτότυπο επειδή μπορεί να ανακτήσει λεπτομερείς πληροφορίες, όπως για παράδειγμα την προέλευση από την αλυσίδα διανομής. Επίσης πραγματοποιείται μελέτη περίπτωσης που εφαρμόζεται σε τοπικά προϊόντα από τη Σαρδηνία της Ιταλίας. Οι Feng, Wang, Chen, Zhang(2020) παρουσίασαν το μοντέλο και την αξιολόγηση της παρακολούθησης πολλαπλών αισθητήρων (WSN) που βασίζονται σε blockchain για τη λήψη δεδομένων αλλαγών μικροπεριβάλλοντος και την αποτροπή παραβίασης δεδομένων αντίστοιχα αλλά και την υλοποίηση των αλγορίθμων K-means και SVM σε εφαρμογές αξιολόγησης ποιότητας για την ταξινόμηση και την πρόβλεψη της απώλειας ποιότητας των κατεψυγμένων θαλασσινών τροφίμων, παρουσιάζοντας αποτελέσματα που διασφαλίζουν την ασφάλεια και την αξιοπιστία των δεδομένων. Τέλος, οι A. Halim, M. Halim, Usman, Daud(2021) δημιούργησαν ένα πρότυπο σύστημα παρακολούθησης θερμοκρασίας με βάση το σύστημα IoT και το Blockchain για τη συλλογή κρίσιμων δεδομένων αποθήκευσης, όπως η παραβίαση θερμοκρασίας, για το CCL του εμβολίου COVID-19.

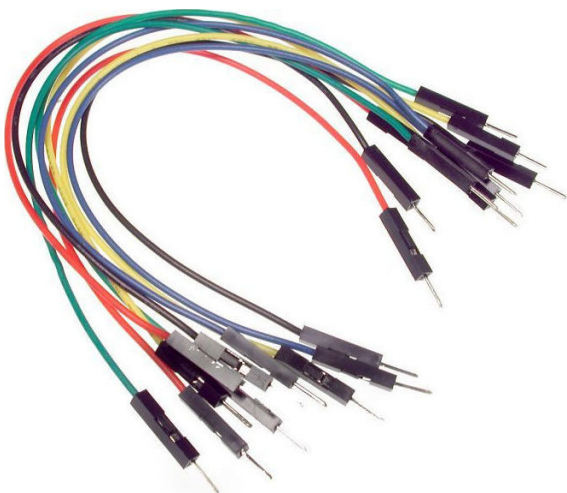
3. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΟΣ

3.1 Αρχικό σύστημα

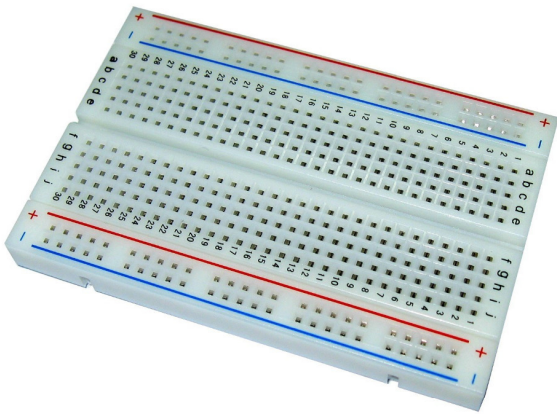
Σε αυτή την εργασία θα δημιουργηθεί ένα απλό σύστημα παρακολούθησης θερμοκρασίας περιβάλλοντος αποτελούμενο από απλά υλικά όπου με την κατάλληλη συνδεσμολογία θα παράγουν τα αποτελέσματα που χρειάζονται. Για τον σχεδιασμό λοιπόν, ενός ενδεικτικού συστήματος παρακολούθησης της θερμοκρασίας περιβάλλοντος χρησιμοποιήθηκαν τα εξής εξαρτήματα:



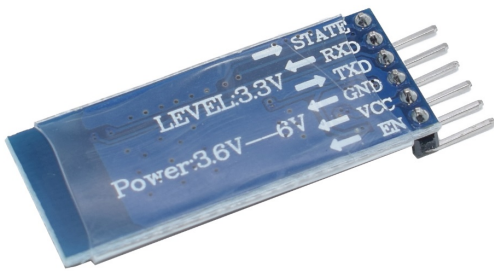
Εικόνα 1: Αισθητήρας θερμοκρασίας LM35



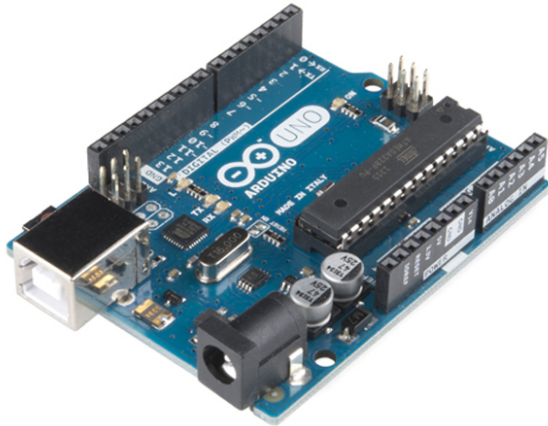
Εικόνα 2: Καλώδια σύνδεσης ή jumper wire



Εικόνα 3: MB-102 Breadboard



Εικόνα 4: HC-06 ασύρματη bluetooth συσκευή



Εικόνα 5: Arduino UNO R3



Εικόνα 6: USB καλώδιο

Ο αισθητήρας θερμοκρασίας έχει 3 πόδια τα οποία συνδέουμε στα αναλογικά pin του Breadboard. Η εσωτερική του αντίσταση αλλάζει ανάλογα με την θερμοκρασία. Όπως γνωρίζουμε από τον κατασκευαστή, όταν γίνεται αντιληπτός 1 βαθμός διαφοράς θερμοκρασίας, η εσωτερική αντίσταση αλλάζει ώστε να έχουμε 10 mV στην πτώση τάσης δηλαδή μια αναλογία απόκρισης 10mV/1C. Αυτή την αναλογία θα την χρησιμοποιήσουμε και στον προγραμματισμό. Ο αισθητήρας δουλεύει από -2° έως 150° C και προκειμένου να υπολογίσουμε την διαφορά θερμοκρασίας από την διαφορά στην τάση πρέπει να πολλαπλασιάσουμε την ένδειξη με το 500 και μετά να διαιρέσουμε με το 1024 για να πάρουμε τους βαθμούς Κελσίου.

Συνδεσμολογία

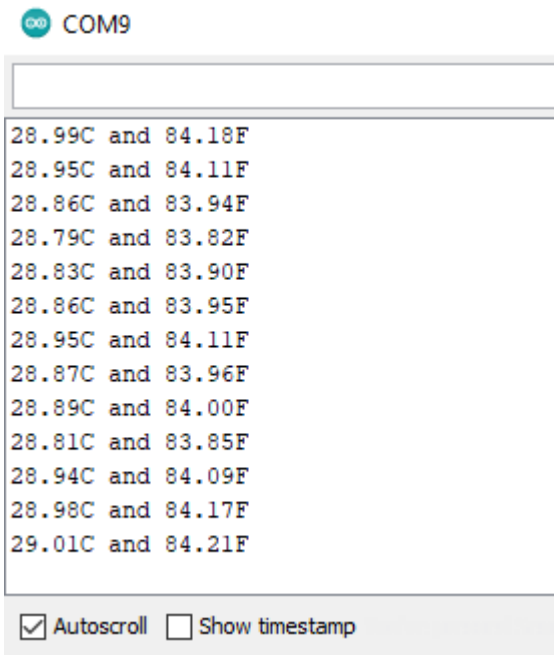
Το αριστερό ποδαράκι του αισθητήρα είναι η τροφοδοσία, το μεσαίο η ένδειξη τάσης και το δεξί η γείωση. Συνδέουμε τον αισθητήρα στο Breadboard και στην συνέχεια συνδέουμε τα

καλώδια στα αντίστοιχα pin του Breadboard και στις κατάλληλες θέσεις στο Arduino. Το μεσαίο το συνδέουμε στο αναλογικό pin A₀ του Arduino για να μας δώσει την ένδειξη της τάσης, το δεξί το συνδέουμε στο pin τροφοδοσίας που έχει την ένδειξη 5V και το αριστερό το συνδέουμε στο pin που αντιστοιχεί στην γείωση το οποίο έχει την ένδειξη GND.

Αφού τελειώσουμε με τις συνδέσεις, συνδέουμε το Arduino UNO στον υπολογιστή μέσω USB καλωδίου και χρησιμοποιούμε το IDE του Arduino προκειμένου να εμφανίσουμε τα αποτελέσματα του αισθητήρα στην κονσόλα μέσω του κατάλληλου κώδικα που παρατίθεται παρακάτω με τα αντίστοιχα σχόλια επεξήγησης.

```
/* Δηλώνουμε τις μεταβλητές στις οποίες θα αποθηκεύσουμε την τιμή από την
αναλογική είσοδο A0, την θερμοκρασία σε βαθμούς Κελσίου και
θερμοκρασία σε βαθμούς Fahrenheit αντίστοιχα */
int a;
float tempC;
float tempF;
/* Setup μέθοδος όπου καθορίζουμε το κανάλι του Serial Monitor στο
Arduino IDE που θα προβάλλουμε τα αποτελέσματα */
void setup()
{
  Serial.begin(9600);
}
// Έναρξη της επαναληπτικής διαδικασίας λήψης και εκτύπωσης τιμών
void loop()
{
  // Διάβασε την τιμή από την αναλογική είσοδο A0
  a = analogRead(A0);
  // Υπολόγισε την θερμοκρασία σε βαθμούς Κελσίου
  tempC = (5.0 * a * 100.0) / 1023;
  // Υπολόγισε την θερμοκρασία σε βαθμούς Fahrenheit
  tempF = tempC*1.8 + 32;
  // Απεικόνισε την θερμοκρασία στο σειριακό μόνιτορ του Arduino IDE
  Serial.print(tempC);
  Serial.print("°C and ");
  // Απεικόνισε την θερμοκρασία στο σειριακό μόνιτορ του Arduino IDE
  Serial.print(tempF);
  Serial.println("F");
  //περίμενε εδώ 1 δευτερόλεπτο
  delay(1000);
}
```

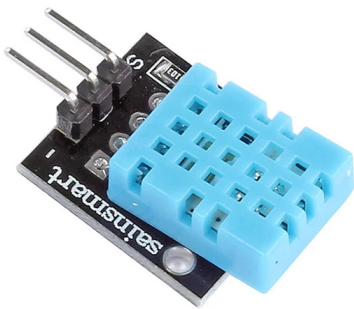
Στην συνέχεια ανοίγουμε το μόνιτορ του προγραμματιστικού περιβάλλοντος του Arduino για να δούμε τις ενδείξεις του αισθητήρα:



Εικόνα 7: Αποτελέσματα θερμοκρασίας στο Arduino IDE Serial Monitor

3.2 Αναβαθμισμένο σύστημα

Για τις ανάγκες παρακολούθησης του περιβάλλοντος είναι αναγκαίο το να ελέγχεται παράλληλα και η υγρασία εκτός της θερμοκρασίας. Επίσης απαραίτητη είναι και η δυνατότητα απομακρυσμένου ελέγχου ο οποίος θα πραγματοποιηθεί με την αποστολή των δεδομένων στο cloud ώστε να μπορούν οι υπεύθυνοι να έχουν πρόσβαση με ευκολία από οποιοδήποτε μέρος. Γι' αυτές τις ανάγκες θα χρειαστεί επιπλέον hardware:



Εικόνα 8: DHT11 Αισθητήρας θερμοκρασίας και υγρασίας

Οπότε απαραίτητη είναι μια νέα συνδεσμολογία η οποία θα έχει ως εξής:

Αρχικά τοποθετούμε τον αισθητήρα στο Breadboard. Τα pins στον αισθητήρα είναι 3. Το S για το σήμα, το μεσαίο που είναι για την τάση και το minus(-) που είναι για την γείωση. Οπότε αντίστοιχα τα καλώδια τα ενώνουμε συνδέοντας το pin του S με το A0 του Arduino, το μεσαίο με το pin 5V του Arduino και τέλος το minus(-) με το pin GND του Arduino.

Για τον νέο αισθητήρα θα χρειαστούμε την βιβλιοθήκη του στον κώδικα μας δηλαδή την <dht.h>. Ο υπόλοιπος κώδικας μας θα είναι παρόμοιος με τον προηγούμενο με την διαφορά ότι προσθέτουμε και την μέτρηση της υγρασίας:

```
// Η βιβλιοθήκη που χρειαζόμαστε για τον αισθητήρα DHT11
#include <dht.h>
#define dht_apin A0 // Η αναλογική είσοδος που θα διαβάζει τις τιμές
dht DHT; //Η μεταβλητή που θα αποθηκεύει τις τιμές

void setup(){

  Serial.begin(9600);
  delay(500); //Περίμενε 0.5 δευτερόλεπτα
  Serial.println("DHT11 Humidity & temperature Sensor\n\n");
  delay(1000); /*Περίμενε 1 δευτερόλεπτο πριν την ανάγνωση του
  αισθητήρα*/

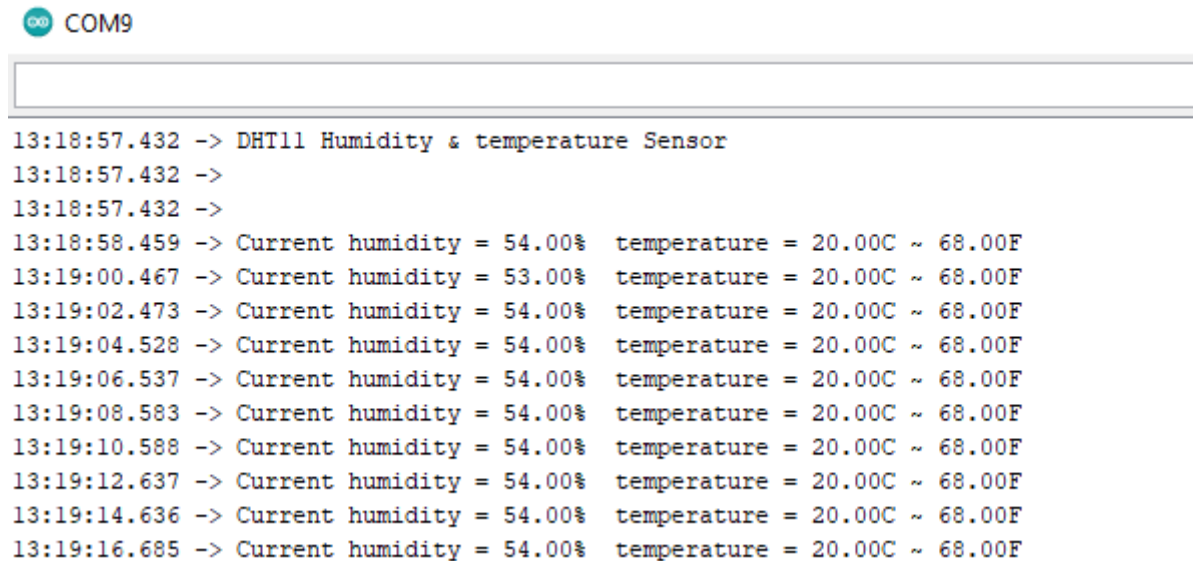
}

void loop(){
  //Έναρξη προγράμματος
  DHT.read11(dht_apin); //διάβασε από το αναλογικό pin A0

  Serial.print("Current humidity = ");
  Serial.print(DHT.humidity);
  Serial.print("% ");
  Serial.print("temperature = ");
  Serial.print(DHT.temperature);
  Serial.print("°C and ");
  Serial.print(DHT.temperature*1.8 + 32);
  Serial.println("F");

  delay(2000); /*Περίμενε 2 δευτερόλεπτα πριν την επόμενη λήψη
  πληροφοριών από τον αισθητήρα*/
}
```

Κατόπιν, πραγματοποιώντας compile και upload τον κώδικα, μπορούμε να ανοίξουμε το serial monitor του Arduino IDE όπου βλέπουμε τα αντίστοιχα αποτελέσματα με διαφορά 2 δευτερολέπτων:



```
COM9
13:18:57.432 -> DHT11 Humidity & temperature Sensor
13:18:57.432 ->
13:18:57.432 ->
13:18:58.459 -> Current humidity = 54.00%  temperature = 20.00C ~ 68.00F
13:19:00.467 -> Current humidity = 53.00%  temperature = 20.00C ~ 68.00F
13:19:02.473 -> Current humidity = 54.00%  temperature = 20.00C ~ 68.00F
13:19:04.528 -> Current humidity = 54.00%  temperature = 20.00C ~ 68.00F
13:19:06.537 -> Current humidity = 54.00%  temperature = 20.00C ~ 68.00F
13:19:08.583 -> Current humidity = 54.00%  temperature = 20.00C ~ 68.00F
13:19:10.588 -> Current humidity = 54.00%  temperature = 20.00C ~ 68.00F
13:19:12.637 -> Current humidity = 54.00%  temperature = 20.00C ~ 68.00F
13:19:14.636 -> Current humidity = 54.00%  temperature = 20.00C ~ 68.00F
13:19:16.685 -> Current humidity = 54.00%  temperature = 20.00C ~ 68.00F
```

Εικόνα 9: Αποτελέσματα θερμοκρασίας και υγρασίας στο Arduino IDE Serial Monitor

3.3 Αυτόματος χειρισμός κλιματισμού

Μετά τις μετρήσεις θερμοκρασίας και υγρασίας, το σύστημα θα πρέπει να διαχειρίζεται κατάλληλα αυτά τα δεδομένα έτσι ώστε να επιτευχθούν οι κατάλληλες συνθήκες που θα επιτρέπουν ασφαλή μεταφορά ευπαθών προϊόντων. Τα δεδομένα θα πρέπει να ελέγχονται και μέσω αυτού του ελέγχου να καθορίζεται ανάλογα ο κλιματισμός του χώρου έτσι ώστε οι συνθήκες να παραμένουν σταθερές στο εύρος τιμών που θεωρείται κατάλληλο. Για τις ανάγκες ελέγχου του κλιματισμού, χρειάζεται επιπλέον hardware και συγκεκριμένα:



Εικόνα 10: IR Receiver Infrared Module



Εικόνα 11: IR Transmitter Infrared

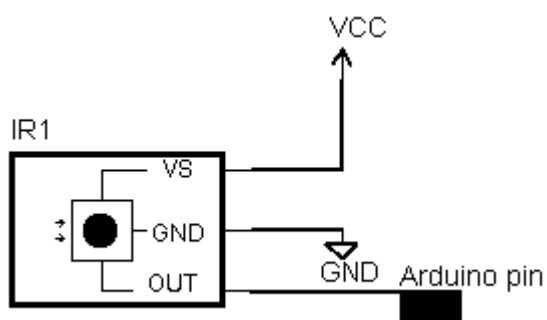
Τα δύο εξαρτήματα που χρειαζόμαστε είναι τηλεχειριστήρια υπερύθρων. Τα περισσότερα τηλεχειριστήρια υπερύθρων λειτουργούν γύρω στις συχνότητες των 38 KHz. Οι περισσότερες ηλεκτρονικές συσκευές ευρείας χρήσης όπως η τηλεόραση, το κλιματιστικό και άλλες συσκευές χρησιμοποιούν υπέρυθρα σήματα για τηλεχειρισμό. Κάθε κατασκευαστής έχει τα δικά του πρωτόκολλα για την κωδικοποίηση των δεδομένων, έτσι ώστε τα σήματα που προορίζονται για μια συσκευή να μην παρεμβαίνουν σε μια άλλη. Το IR Receiver είναι δέκτης υπερύθρων. Αποκωδικοποιεί το λαμβανόμενο σήμα υπερύθρων που θα δέχεται από το τηλεχειριστήριο του κλιματιστικού έτσι ώστε να μπορέσουμε να αναλύσουμε και να στείλουμε το ίδιο σήμα πίσω στο κλιματιστικό μέσω του IR Transmitter δηλαδή του πομπού υπερύθρων που είναι η δεύτερη συσκευή.

3.3.1 Αποκωδικοποίηση σήματος

Αρχικά θα πραγματοποιήσουμε την αποκωδικοποίηση του σήματος του τηλεχειριστηρίου μέσω του IR Receiver.

Συνδεσμολογία

Το IR Receiver συνδυάζει ένα φωτοκύτταρο ευαίσθητο στα υπέρυθρα σήματα και ένα φίλτρο ζώνης 38 kHz. Λειτουργεί σε μια ποικιλία τάσεων τροφοδοσίας συμπεριλαμβανομένων των 3,3V και 5V. Αποκωδικοποιεί το λαμβανόμενο σήμα υπέρυθρων και δίνει ένα τετραγωνικό κύμα παλμών απενεργοποίησης και ενεργοποίησης στο επίπεδο τάσης του τροφοδοτικού οπότε η τροφοδοσία της εξόδου του μπορεί να γίνει απευθείας στην ψηφιακή ακίδα εισόδου του Arduino. Το τοποθετούμε στο Breadboard και τοποθετούμε 3 jumper wires από κάτω του, ένα για κάθε άκρο. Το αριστερό του άκρο το συνδέουμε σε οποιαδήποτε ψηφιακή ακίδα εισόδου στο Arduino, την οποία θα την ορίσουμε στον κώδικα στην συνέχεια. Το μεσαίο του άκρο το συνδέουμε στην γείωση GND και το δεξί του άκρο στην υποδοχή 5V του Arduino.



Εικόνα 12: Το κύκλωμα του IR Receiver

Ένα τηλεχειριστήριο υπέρυθρων λειτουργεί ενεργοποιώντας και απενεργοποιώντας το LED σε ένα συγκεκριμένο μοτίβο. Ωστόσο, για να αποφευχθεί η παρεμβολή από πηγές υπέρυθρων, όπως το ηλιακό φως ή τα φώτα, το LED δεν ανάβει σταθερά, αλλά ανάβει και σβήνει σε συχνότητα διαμόρφωσης (συνήθως 36, 38 ή 40 KHz). Ο χρόνος αποστολής ενός διαμορφωμένου σήματος ονομάζεται "mark" και όταν το LED είναι σβηστό ονομάζεται "space".

Κάθε πλήκτρο στο τηλεχειριστήριο αντιστοιχεί σε έναν συγκεκριμένο κωδικό (συνήθως 12 έως 32 bit) και μεταδίδει αυτόν τον κωδικό όταν πατηθεί.

Τα σήματα υπέρυθρων λοιπόν, αποτελούνται από μια σειρά διαμορφωμένων παλμών που ονομάζονται "marks" που χωρίζονται με διαστήματα που ονομάζονται "spaces". Συνήθως υπάρχει ένα μεγάλο mark και ένα space στην αρχή κάθε σήματος που χρησιμεύει ως κεφαλίδα. Στη συνέχεια, μεταβάλλοντας το χρονισμό από marks και spaces, μεταδίδεται μια ακολουθία bit. Τα σήματα αποστέλλονται σύμφωνα με πολύ συγκεκριμένα πρωτόκολλα που επιτρέπουν την λήψη αυτών των επαναλαμβανόμενων δεδομένων χρονισμού και την μετατροπή σε έναν ενιαίο δυαδικό αριθμό έως και 32 bit. Η βιβλιοθήκη υπέρυθρων συλλέγει τις πληροφορίες χρονισμού σε ένα buffer και στη συνέχεια τις μετατρέπει σε μια τιμή 32 bit.

Ομοίως, για την μετάδοση σημάτων υπέρυθρων, χρειάζεται να περάσουμε την τιμή 32 bit στη βιβλιοθήκη και να επιλέξουμε το πρωτόκολλο που θέλουμε να

χρησιμοποιήσουμε. Μετατρέπει αυτήν την τιμή σε μια ροή marks και spaces με τις κατάλληλες κεφαλίδες, κωδικοποιήσεις bit και χρονισμούς. Η βιβλιοθήκη κώδικα που θα χρησιμοποιηθεί είναι η "IRremote" που δημοσίευσε ο Ken Shirriff στο GitHub.

Προγραμματισμός

Αφού κατεβάσουμε την βιβλιοθήκη "IRremote" και την εισάγουμε στις βιβλιοθήκες του Arduino μπορούμε να δημιουργήσουμε το νέο project στο Arduino IDE που θα κάνουμε upload στην πλατφόρμα. Ο κώδικας που χρησιμοποιήθηκε αρχικά είναι ο εξής:

```
// Η βιβλιοθήκη που χρειαζόμαστε για την χρήση των IR σημάτων
#include <IRremote.h>
// Το pin εισόδου που επιλέξαμε στην συνδεσμολογία
const int RECV_PIN = 6;
// Δημιουργία του object για τον IR Receiver
IRrecv irrecv(RECV_PIN);
/* Δημιουργία μεταβλητής που θα αποθηκευτούν τα αποτελέσματα του irrecv
*/
decode_results results;

void setup()
{
  Serial.begin(9600);
  // Έναρξη του IR Receiver
  irrecv.enableIRIn();
  /* Λέει στο αντικείμενο IRrecv να αναβοσβήνει το LED όταν λαμβάνεται
σήμα IR*/
  irrecv.blink13(true);
}

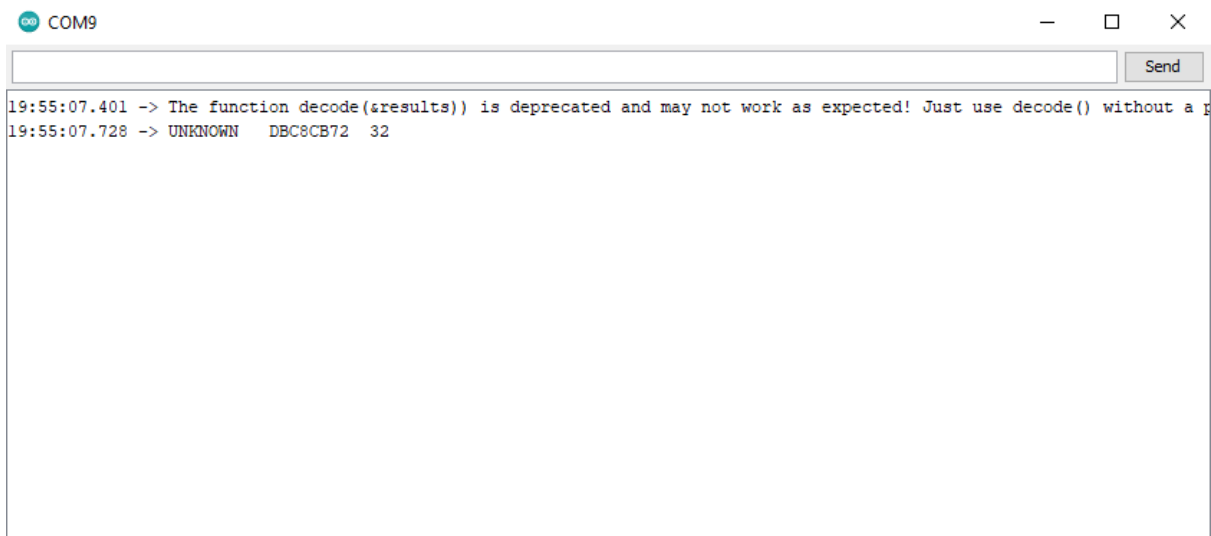
void loop() {
  /* Ανάγνωση του τύπου IR σήματος ανάλογα με την μάρκα του κλιματιστικού
  */
  if (irrecv.decode(&results)) {
    if (results.decode_type == NEC) {
      Serial.print("NEC ");
    } else if (results.decode_type == SONY) {
      Serial.print("SONY ");
    } else if (results.decode_type == RC5) {
      Serial.print("RC5 ");
    } else if (results.decode_type == RC6) {
      Serial.print("RC6 ");
    }
  }
}
```

```

    } else if (results.decode_type == UNKNOWN) {
      Serial.print("UNKNOWN ");
    }
    // Εκτύπωση των αποτελεσμάτων
    Serial.print(" ");
    Serial.print(results.value, HEX);
    Serial.print(" ");
    Serial.println(results.bits);
    // Επανεκκίνηση για ανάγνωση της επόμενης τιμής
    irrecv.resume();
  }
}

```

Αφού γίνει επιτυχώς το upload στην πλατφόρμα και ξεκινήσει η λειτουργία της, ανοίγουμε το serial monitor και ρυθμίζουμε στα 9600 baud. Στρέφουμε το τηλεχειριστήριο του κλιματιστικού προς τον IR αποδέκτη και πατάμε το κουμπί ON/OFF προκειμένου να αποκωδικοποιήσουμε αυτά τα σήματα και να μπορέσουμε να αυτοματοποιήσουμε την αποστολή τους. Αφού γίνει αυτό λαμβάνουμε το εξής αποτέλεσμα:



Εικόνα 13: Αποτελέσματα ανεπιτυχούς αναγνώρισης IR σήματος στο Arduino IDE Serial Monitor

Οι στήλες αποτελεσμάτων αποτελούν αντίστοιχα τα Code Type, Code Hex Value, Bits used. Το αποτέλεσμα “UNKNOWN” για το Code Type δείχνει ότι η βιβλιοθήκη δεν περιέχει τα σήματα της συγκεκριμένης συσκευής που χρησιμοποιείται. Άρα δεν υπάρχει και η κατάλληλη μέθοδος που μπορεί να χρησιμοποιηθεί στην συνέχεια ώστε να στείλει το σήμα μέσω του IR Transmitter. Μετά από αυτό το πρόβλημα,

χρησιμοποιήθηκε διαφορετικός κώδικας ο οποίος θα μπορεί να διαβάσει ένα raw σήμα και να το κλωνοποιεί, χωρίς ενδιάμεσες μεθόδους.

Ο νέος κώδικας που θα χρησιμοποιήσουμε είναι ο εξής:

```
#define IRpin_PIN PIND
#define IRpin 6
#define MAXPULSE 65000
#define RESOLUTION 20
// Ορισμός πίνακα για αποθήκευση του σήματος
uint16_t pulses[100][2];
uint8_t currentpulse = 0;

void setup(void) {
  Serial.begin(9600);
  Serial.println("Ready to decode IR!");
}

void loop(void) {
  uint16_t highpulse, lowpulse;
  highpulse = lowpulse = 0;
  // Ανάγνωση σήματος
  while (IRpin_PIN & (1 << IRpin)) {

    highpulse++;
    delayMicroseconds(RESOLUTION);

    if ((highpulse >= MAXPULSE) && (currentpulse != 0)) {
      printpulses(); // Κλήση μεθόδου για εκτύπωση αποτελεσμάτων
      currentpulse=0;
      return;
    }

    pulses[currentpulse][0] = highpulse;
    // Ανάγνωση σήματος
    while (!(IRpin_PIN & _BV(IRpin))) {

      lowpulse++;
      delayMicroseconds(RESOLUTION);
      if ((lowpulse >= MAXPULSE) && (currentpulse != 0)) {
        printpulses(); // Κλήση μεθόδου για εκτύπωση αποτελεσμάτων
```

```

    currentpulse=0;
    return;
}
}
pulses[currentpulse][1] = lowpulse;

currentpulse++;
}

// Μέθοδος για εκτύπωση της αποκωδικοποίησης του σήματος
void printpulses(void) {

    Serial.println("int IRsignal[] = {");
    for (uint8_t i = 0; i < currentpulse-1; i++) {
        Serial.print("\t");
        Serial.print(pulses[i][1] * RESOLUTION / 10, DEC);
        Serial.print(", ");
        Serial.print(pulses[i+1][0] * RESOLUTION / 10, DEC);
        Serial.println(",");
    }
    Serial.print("\t");
    Serial.print(pulses[currentpulse-1][1] * RESOLUTION / 10, DEC);
    Serial.print(", 0}");
}

```

Επαναλαμβάνουμε την διαδικασία, κάνοντας upload και ανοίγοντας το serial monitor. Στην συνέχεια πατάμε το κουμπί ON/OFF στο τηλεχειριστήριο του κλιματιστικού και παίρνουμε τα παρακάτω αποτελέσματα:

COM9	COM9	COM9	COM9
13:20:41.735 -> Ready to decode IR!	13:21:30.570 -> 1640 usec, 580 usec	13:21:31.637 -> int IRsignal[] = {	13:21:32.194 -> 56, 54,
13:21:29.503 ->	13:21:30.570 -> 5360 usec, 4560 usec	13:21:31.637 -> // ON, OFF (in 10's	13:21:32.194 -> 58, 164,
13:21:29.503 ->	13:21:30.570 -> 4540 usec, 580 usec	13:21:31.685 -> 456, 456,	13:21:32.194 -> 58, 164,
13:21:29.503 -> Received:	13:21:30.617 -> 1640 usec, 580 usec	13:21:31.731 -> 56, 166,	13:21:32.194 -> 58, 536,
13:21:29.503 -> OFF ON	13:21:30.617 -> 540 usec, 580 usec	13:21:31.731 -> 56, 54,	13:21:32.240 -> 456, 454,
13:21:29.503 -> 63504 usec, 4560 usec	13:21:30.664 -> 1640 usec, 580 usec	13:21:31.731 -> 58, 164,	13:21:32.240 -> 58, 164,
13:21:29.503 -> 4560 usec, 560 usec	13:21:30.664 -> 1640 usec, 560 usec	13:21:31.731 -> 58, 164,	13:21:32.287 -> 58, 54,
13:21:29.549 -> 1660 usec, 560 usec	13:21:30.664 -> 540 usec, 580 usec	13:21:31.731 -> 58, 52,	13:21:32.287 -> 58, 164,
13:21:29.549 -> 540 usec, 580 usec	13:21:30.710 -> 540 usec, 560 usec	13:21:31.731 -> 58, 54,	13:21:32.287 -> 58, 164,
13:21:29.597 -> 1640 usec, 580 usec	13:21:30.757 -> 1660 usec, 560 usec	13:21:31.777 -> 58, 164,	13:21:32.287 -> 56, 54,
13:21:29.597 -> 1640 usec, 580 usec	13:21:30.757 -> 540 usec, 580 usec	13:21:31.777 -> 58, 52,	13:21:32.334 -> 58, 54,
13:21:29.597 -> 520 usec, 580 usec	13:21:30.802 -> 540 usec, 560 usec	13:21:31.777 -> 58, 54,	13:21:32.334 -> 56, 166,
13:21:29.643 -> 540 usec, 580 usec	13:21:30.802 -> 1660 usec, 560 usec	13:21:31.825 -> 56, 166,	13:21:32.334 -> 56, 54,
13:21:29.689 -> 1640 usec, 580 usec	13:21:30.802 -> 540 usec, 580 usec	13:21:31.825 -> 58, 52,	13:21:32.334 -> 58, 54,
13:21:29.689 -> 520 usec, 580 usec	13:21:30.849 -> 520 usec, 580 usec	13:21:31.825 -> 58, 54,	13:21:32.334 -> 56, 166,
13:21:29.735 -> 540 usec, 560 usec	13:21:30.849 -> 1640 usec, 580 usec	13:21:31.825 -> 56, 166,	13:21:32.334 -> 56, 54,
13:21:29.735 -> 1660 usec, 580 usec	13:21:30.895 -> 1640 usec, 580 usec	13:21:31.825 -> 58, 164,	13:21:32.380 -> 58, 52,
13:21:29.735 -> 520 usec, 580 usec	13:21:30.895 -> 540 usec, 580 usec	13:21:31.825 -> 58, 52,	13:21:32.380 -> 58, 164,
13:21:29.781 -> 540 usec, 560 usec	13:21:30.895 -> 1640 usec, 580 usec	13:21:31.871 -> 58, 164,	13:21:32.380 -> 58, 164,
13:21:29.781 -> 1660 usec, 580 usec	13:21:30.941 -> 1640 usec, 580 usec	13:21:31.871 -> 58, 164,	13:21:32.426 -> 58, 54,
13:21:29.827 -> 1640 usec, 580 usec	13:21:30.987 -> 540 usec, 560 usec	13:21:31.871 -> 58, 54,	13:21:32.426 -> 58, 164,
13:21:29.827 -> 520 usec, 580 usec	13:21:30.987 -> 1660 usec, 560 usec	13:21:31.918 -> 58, 164,	13:21:32.426 -> 58, 164,
13:21:29.827 -> 1640 usec, 580 usec	13:21:31.034 -> 1660 usec, 560 usec	13:21:31.918 -> 56, 166,	13:21:32.426 -> 58, 54,
13:21:29.874 -> 1640 usec, 580 usec	13:21:31.034 -> 1660 usec, 560 usec	13:21:31.918 -> 56, 166,	13:21:32.426 -> 56, 166,
13:21:29.920 -> 540 usec, 580 usec	13:21:31.034 -> 1660 usec, 580 usec	13:21:31.963 -> 58, 164,	13:21:32.426 -> 56, 166,
13:21:29.920 -> 1640 usec, 560 usec	13:21:31.080 -> 1640 usec, 580 usec	13:21:31.963 -> 58, 164,	13:21:32.473 -> 56, 166,
13:21:29.966 -> 1660 usec, 560 usec	13:21:31.126 -> 1640 usec, 580 usec	13:21:31.963 -> 58, 164,	13:21:32.473 -> 56, 166,
13:21:29.966 -> 1660 usec, 580 usec	13:21:31.126 -> 520 usec, 580 usec	13:21:31.963 -> 58, 52,	13:21:32.473 -> 58, 164,
13:21:29.966 -> 1640 usec, 580 usec	13:21:31.126 -> 1640 usec, 580 usec	13:21:31.963 -> 58, 164,	13:21:32.518 -> 58, 164,
13:21:30.013 -> 1640 usec, 580 usec	13:21:31.173 -> 540 usec, 580 usec	13:21:31.963 -> 58, 54,	13:21:32.518 -> 58, 52,
13:21:30.013 -> 1640 usec, 580 usec	13:21:31.173 -> 520 usec, 580 usec	13:21:32.009 -> 58, 52,	13:21:32.518 -> 58, 164,
13:21:30.060 -> 520 usec, 580 usec	13:21:31.219 -> 540 usec, 560 usec	13:21:32.009 -> 58, 54,	13:21:32.565 -> 58, 54,
13:21:30.060 -> 1640 usec, 580 usec	13:21:31.219 -> 540 usec, 580 usec	13:21:32.009 -> 58, 52,	13:21:32.565 -> 58, 52,
13:21:30.060 -> 540 usec, 580 usec	13:21:31.266 -> 540 usec, 560 usec	13:21:32.056 -> 58, 54,	13:21:32.565 -> 58, 54,
13:21:30.106 -> 520 usec, 580 usec	13:21:31.266 -> 540 usec, 580 usec	13:21:32.056 -> 58, 52,	13:21:32.565 -> 56, 54,
13:21:30.154 -> 540 usec, 580 usec	13:21:31.266 -> 1640 usec, 580 usec	13:21:32.056 -> 58, 164,	13:21:32.565 -> 58, 54,
13:21:30.154 -> 520 usec, 580 usec	13:21:31.313 -> 540 usec, 560 usec	13:21:32.056 -> 58, 54,	13:21:32.565 -> 56, 54,
13:21:30.200 -> 540 usec, 580 usec	13:21:31.358 -> 1660 usec, 560 usec	13:21:32.056 -> 56, 166,	13:21:32.613 -> 58, 164,
13:21:30.200 -> 520 usec, 580 usec	13:21:31.358 -> 1640 usec, 580 usec	13:21:32.056 -> 56, 166,	13:21:32.613 -> 58, 54,
13:21:30.200 -> 1640 usec, 580 usec	13:21:31.358 -> 1640 usec, 580 usec	13:21:32.102 -> 58, 164,	13:21:32.613 -> 56, 166,
13:21:30.247 -> 540 usec, 560 usec	13:21:31.404 -> 1660 usec, 580 usec	13:21:32.102 -> 58, 164,	13:21:32.658 -> 56, 164,
13:21:30.247 -> 1660 usec, 560 usec	13:21:31.404 -> 520 usec, 580 usec	13:21:32.102 -> 58, 52,	13:21:32.658 -> 58, 164,
13:21:30.293 -> 1660 usec, 580 usec	13:21:31.453 -> 520 usec, 580 usec	13:21:32.149 -> 58, 54,	13:21:32.658 -> 58, 166,
13:21:30.293 -> 1640 usec, 580 usec	13:21:31.453 -> 540 usec, 580 usec	13:21:32.149 -> 58, 52,	13:21:32.658 -> 58, 52,
13:21:30.293 -> 1640 usec, 580 usec	13:21:31.497 -> 1640 usec, 580 usec	13:21:32.149 -> 58, 164,	13:21:32.658 -> 58, 52,
13:21:30.340 -> 520 usec, 580 usec	13:21:31.497 -> 520 usec, 580 usec	13:21:32.194 -> 58, 54,	13:21:32.705 -> 58, 54,
13:21:30.385 -> 540 usec, 580 usec	13:21:31.497 -> 540 usec, 580 usec	13:21:32.194 -> 58, 52,	13:21:32.705 -> 58, 164,
13:21:30.385 -> 520 usec, 580 usec	13:21:31.544 -> 520 usec, 580 usec	13:21:32.194 -> 58, 54,	13:21:32.705 -> 58, 52,
13:21:30.431 -> 1640 usec, 580 usec	13:21:31.590 -> 540 usec, 580 usec	13:21:32.194 -> 56, 54,	13:21:32.705 -> 58, 54,
13:21:30.431 -> 540 usec, 580 usec	13:21:31.590 -> 1640 usec, 580 usec	13:21:32.194 -> 58, 164,	13:21:32.751 -> 58, 52,
13:21:30.431 -> 520 usec, 580 usec	13:21:31.590 -> 1640 usec, 580 usec	13:21:32.194 -> 58, 164,	13:21:32.751 -> 58, 54,
13:21:30.478 -> 540 usec, 560 usec	13:21:31.637 -> int IRsignal[] = {	13:21:32.194 -> 58, 536,	13:21:32.751 -> 58, 164,
13:21:30.478 -> 540 usec, 580 usec	13:21:31.637 -> // ON, OFF (in 10's of	13:21:32.240 -> 456, 454,	13:21:32.751 -> 58, 164,
13:21:30.524 -> 1640 usec, 580 usec	13:21:31.685 -> 456, 456,	13:21:32.240 -> 58, 164,	13:21:32.751 -> 58, 0};

Εικόνα 14: Αποτελέσματα επιτυχούς αναγνώρισης IR σήματος στο Arduino IDE Serial Monitor

Από όλα αυτά τα αποτελέσματα, αυτό που θα χρειαστεί να αποθηκεύσουμε και να χρησιμοποιήσουμε στην συνέχεια είναι το array IRSignal[].

Πραγματοποιούμε την ίδια διαδικασία πατώντας μια ακόμη φορά το κουμπί ON/OFF για να αποκωδικοποιήσουμε και το σήμα που απενεργοποιεί τον κλιματισμό. Αφού αποθηκεύσουμε το νέο IRSignal ξανά τότε έχουμε ολοκληρώσει την αποκωδικοποίηση των σημάτων και συνεχίζουμε στην αποστολή τους.

3.3.2 Κλωνοποίηση και αποστολή σήματος

Τώρα θα πραγματοποιήσουμε την αποστολή του σήματος του τηλεχειριστηρίου που αναλύσαμε μέσω του IR Receiver, χρησιμοποιώντας το IR Transmitter.

Συνδεσμολογία

Μπορούμε να αποσυνδέσουμε το IR Receiver καθώς δεν χρειάζεται στην συνέχεια. Το IR Transmitter έχει δύο ακίδες, μια μικρή και μια μεγάλη. Η μεγάλη συνδέεται σε ένα pin εισόδου του Arduino και η μικρή στην γείωση GND. Στην συγκεκριμένη περίπτωση μπορεί να μην χρησιμοποιηθεί το Breadboard και jumper wires αλλά να συνδεθεί κατευθείαν στην πλατφόρμα του Arduino στις θέσεις pin 13 και GND που βρίσκονται η μία δίπλα στην άλλη.

Προγραμματισμός

Σύμφωνα με την αρχική βιβλιοθήκη κώδικα που είχαμε χρησιμοποιήσει θα έπρεπε να ορίσουμε τον κατασκευαστή της συσκευής για την αποστολή σήματος αφού διαφορετικοί κατασκευαστές συσκευών χρησιμοποιούν διαφορετικά πρωτόκολλα. Τώρα όμως απλά θα χρειαστεί να στείλουμε πίσω το ίδιο σήμα που παραλάβαμε.

Αντιγράφουμε τις κωδικοποιήσεις των σημάτων που παραλάβαμε μέσω των arrays και δημιουργούμε τις αντίστοιχες μεταβλητές στον κώδικα:

```
/* Οι βιβλιοθήκες που χρειαζόμαστε για την χρήση των IR σημάτων και αισθητήρα DHT11 */
```

```
#include <IRremote.hpp>
```

```
#include <dht.h>
```

```
//Ορισμός pin εισόδου για τον αισθητήρα DHT
```

```
#define dht_apin A0
```

```
//Ορισμός pin εισόδου για το IR Transmitter
```

```
#define IRledPin 13
```

```
#define NumIRsignals 200
```

```
dht DHT;
```

```
int IRsignalON[] = {
```

```
456, 454, 58, 164, 58, 52, 58, 166, 56, 164, 58, 54, 58, 52, 58,  
164, 58, 54, 58, 52, 58, 164, 58, 54, 56, 54, 58, 164, 58, 164, 58,  
54, 56, 166, 58, 164, 58, 52, 58, 164, 58, 164, 58, 164, 58, 164,  
58, 164, 58, 164, 58, 54, 58, 164, 58, 52, 58, 54, 58, 52, 58, 54,  
56, 54, 58, 52, 58, 166, 56, 54, 58, 164, 58, 164, 58, 164, 58, 164,
```

```

58, 54, 56, 54, 58, 54, 56, 164, 58, 54, 58, 52, 58, 54, 58, 52, 58,
164, 58, 164, 58, 536, 456, 456, 56, 166, 56, 54, 58, 164, 58, 164,
58, 52, 58, 54, 58, 164, 58, 52, 58, 54, 56, 166, 58, 52, 58, 54,
56, 166, 58, 164, 58, 52, 58, 164, 58, 164, 58, 54, 58, 164,
58, 164, 58, 164, 58, 164, 58, 164, 58, 164, 58, 54, 56, 164, 58,
54, 58, 52, 58, 54, 58, 52, 58, 54, 58, 52, 58, 164, 58, 54, 56,
166, 56, 166, 56, 166, 56, 166, 56, 54, 58, 52, 58, 54, 58, 164, 58,
52, 58, 54, 58, 52, 58, 54, 56, 166, 58, 164, 58, 0
};

```

```

int IRsignalOFF[] = {

```

```

458, 454, 58, 164, 58, 52, 58, 164, 58, 164, 58, 54, 58, 52, 58,
164, 58, 54, 56, 54, 58, 164, 58, 54, 56, 54, 58, 164, 58, 164, 58,
52, 58, 164, 58, 54, 58, 164, 58, 164, 58, 164, 58, 164, 58, 54, 56,
166, 56, 166, 58, 164, 58, 52, 58, 54, 58, 52, 58, 54, 56, 166, 58,
52, 58, 54, 56, 166, 56, 166, 56, 166, 56, 54, 58, 54, 56, 54, 58,
52, 58, 54, 58, 52, 58, 54, 58, 52, 58, 164, 58, 164, 58, 164, 58,
164, 58, 164, 58, 536, 456, 454, 58, 164, 58, 54, 58, 164, 58, 164,
58, 52, 58, 54, 58, 164, 58, 52, 58, 54, 58, 164, 58, 52, 58, 54,
58, 164, 58, 164, 58, 52, 58, 164, 58, 54, 58, 164, 56, 166, 56,
166, 58, 164, 58, 52, 58, 164, 58, 164, 58, 164, 58, 54, 58, 52, 58,
54, 58, 52, 58, 164, 58, 54, 56, 54, 58, 164, 58, 164, 58, 164, 58,
52, 58, 54, 58, 52, 58, 54, 58, 52, 58, 54, 58, 52, 58, 54, 58, 164,
56, 166, 58, 164, 58, 164, 58, 164, 58, 0
};

```

```

void setup() {

```

```

// Επιβεβαίωση ότι το IR είναι απενεργοποιημένο στην εκκίνηση
digitalWrite(IRledPin, LOW); pinMode(IRledPin, OUTPUT);
Serial.begin(9600);
}

```

```

void loop() {

```

```

// Ανάγνωση από αισθητήρα
DHT.read11(dht_apin);
// Εκτύπωση τιμών από αισθητήρα
Serial.print("Current humidity = ");
Serial.print(DHT.humidity);
Serial.print("% ");
Serial.print("temperature = ");
Serial.print(DHT.temperature);
Serial.print("°C ~ ");

```

```

Serial.print(DHT.temperature*1.8 + 32);
Serial.println("F");

/*Αν η θερμοκρασία είναι κάτω από 19°C → στείλε IR σήμα για
ενεργοποίηση του κλιματιστικού*/
if(DHT.temperature < 19)
{
Serial.println("Temperature low --> AC powered on");
for (int i = 0; i < NumIRsignals; i+=2)
{
//Στείλε σήμα ενεργοποίησης κλιματιστικού
pulseIR(IRsignalON[i]*10);
//Απενεργοποίηση του IR Transmitter
delayMicroseconds(IRsignalON[i+1]*10);
}
}
/*Αν η θερμοκρασία ξεπερνάει τους 19°C → στείλε IR σήμα για
απενεργοποίηση του κλιματιστικού*/
if(DHT.temperature > 19)
{
Serial.println("Temperature high --> AC powered off");
for (int i = 0; i < NumIRsignals; i+=2)
{
//Στείλε σήμα απενεργοποίησης κλιματιστικού
pulseIR(IRsignalOFF[i]*10);
//Απενεργοποίηση του IR Transmitter
delayMicroseconds(IRsignalOFF[i+1]*10);
}
}
//Περίμενε 60 δευτερόλεπτα για επανάληψη ελέγχου
delay(60000);
}
//Μέθοδος αποστολής σήματος
void pulseIR(long microsecs) {
//Απενεργοποίηση όλων των πιθανών διακοπών συστήματος
cli();

while (microsecs > 0) {
digitalWrite(IRledPin, HIGH);
delayMicroseconds(10);
digitalWrite(IRledPin, LOW);
delayMicroseconds(10);
microsecs -= 26;
}
}

```

```
}  
//Ενεργοποίηση των πιθανών διακοπών συστήματος  
sei();  
}
```

Χρησιμοποιώντας τον παραπάνω κώδικα λοιπόν, ορίζουμε συγκεκριμένες τιμές θερμοκρασίας και πραγματοποιούμε έλεγχο για τις συνθήκες περιβάλλοντος μέσω του αισθητήρα θερμοκρασίας και υγρασίας DHT. Ενδεικτικά, παραπάνω έχουμε ορίσει ως επιθυμητή θερμοκρασία χώρου τους 19° C και το σήμα αποστέλλεται ανά λεπτό. Εφόσον οι συνθήκες δεν είναι αυτές που χρειαζόμαστε, το Arduino μέσω του IR Transmitter στέλνει το ανάλογο σήμα είτε για να ενεργοποιήσει, είτε για να απενεργοποιήσει τον κλιματισμό. Έτσι οι συνθήκες του περιβάλλοντος θα παραμένουν πάντα σταθερές και η ποιότητα των ευπαθών προϊόντων θα προστατεύεται αυτόματα κατά την μεταφορά τους.

Σε αυτό το σημείο θα μπορούσαμε να πραγματοποιήσουμε μια επέκταση του project με την χρήση της HC-06 ασύρματης bluetooth συσκευής όπου μέσω ενός smartphone και μιας νέας εφαρμογής θα μπορούσαμε να στέλνουμε τα σήματα στο Arduino.

3.4 Αποστολή δεδομένων στο Cloud και απομακρυσμένος έλεγχος

Για την ολοκλήρωση του συστήματος, τα δεδομένα που λαμβάνονται από τον αισθητήρα εκτός από την χρησιμοποίησή τους για τον έλεγχο του κλιματισμού του χώρου, θα πρέπει να στέλνονται στο cloud ώστε να είναι δυνατός ο απομακρυσμένος έλεγχος όπως θα ήταν απαραίτητο σε ένα ρεαλιστικό σενάριο. Θα χρειαστεί επικοινωνία του συστήματος με ένα WiFi Router του χώρου που ελέγχουμε ώστε να μπορούν να σταλούν τα δεδομένα. Αυτή η σύνδεση θα πραγματοποιηθεί με την συσκευή Nodemcu V3. Οπότε το hardware που χρειάζεται να προστεθεί πλέον είναι:



Εικόνα 15: NodeMcu V3 Lua CH340G ESP8266

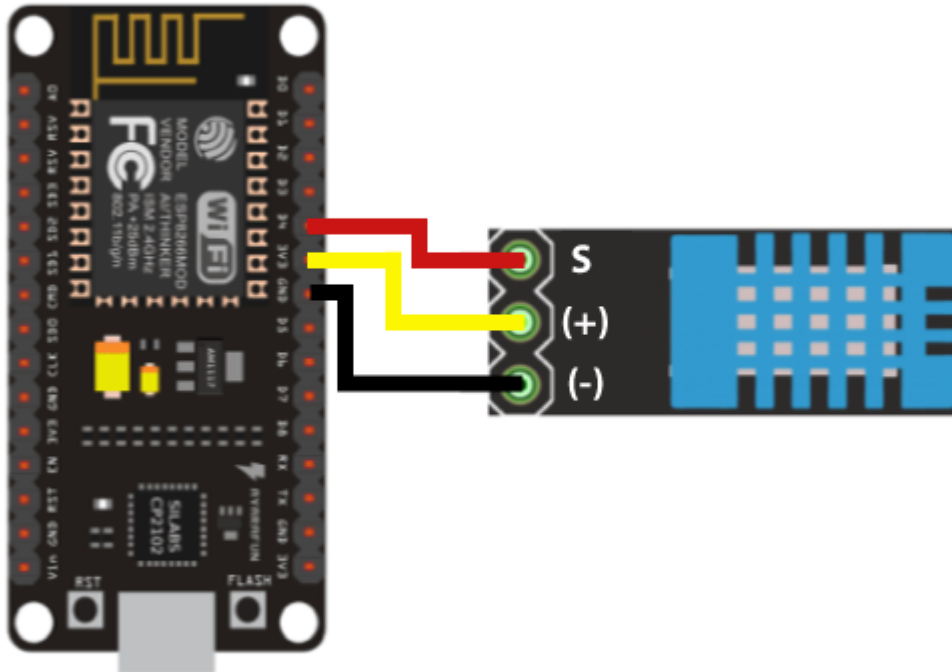


Εικόνα 16: USB cable for data

Συνδεσμολογία

Πρέπει να συνδέσουμε το NodeMcu με τον αισθητήρα DHT. Για να γίνει αυτό συνδέουμε την αριστερή ακίδα(S) του αισθητήρα σε μια ακίδα εισόδου του NodeMcu. Στο παρόν σύστημα έχει συνδεθεί στο D4 και έτσι λαμβάνει τα δεδομένα εισόδου. Η μεσαία ακίδα είναι η τροφοδοσία οπότε την συνδέουμε με την ακίδα 3V του NodeMcu και τέλος η δεξιά ακίδα(-) είναι η γείωση οπότε την συνδέουμε με το GND του NodeMcu. Στην προκειμένη περίπτωση ο αισθητήρας έχει συνδεθεί ήδη με την

τροφοδοσία και την γείωση του Arduino οπότε το μόνο που χρειάζεται είναι να συνδέσουμε την αριστερή ακίδα που αφορά την μεταφορά δεδομένων. Για λόγους αρτιότητας ωστόσο, πραγματοποιούμε σωστά τις συνδέσεις και πάλι. Τέλος συνδέουμε το καλώδιο USB του NodeMcu στον υπολογιστή.



Εικόνα 17: Συνδεσμολογία NodeMCU με DHT11

Προγραμματισμός

Για να συνδέσουμε το NodeMcu στο WiFi και να στείλουμε δεδομένα στο Cloud θα χρησιμοποιήσουμε τον παρακάτω κώδικα:

```
#include <ESP8266WiFi.h>
#include <dht.h>
#include <ThingSpeak.h>

dht DHT;
#define DHT11_PIN D4
/* δημιουργία object για σύνδεση με WiFi και μεταβλητές για το Channel ID
και το Write API Key του ThingSpeak */
WiFiClient client;

long myChannelNumber = yourChannelID;
const char myWriteAPIKey[] = "yourAPIKey";
```

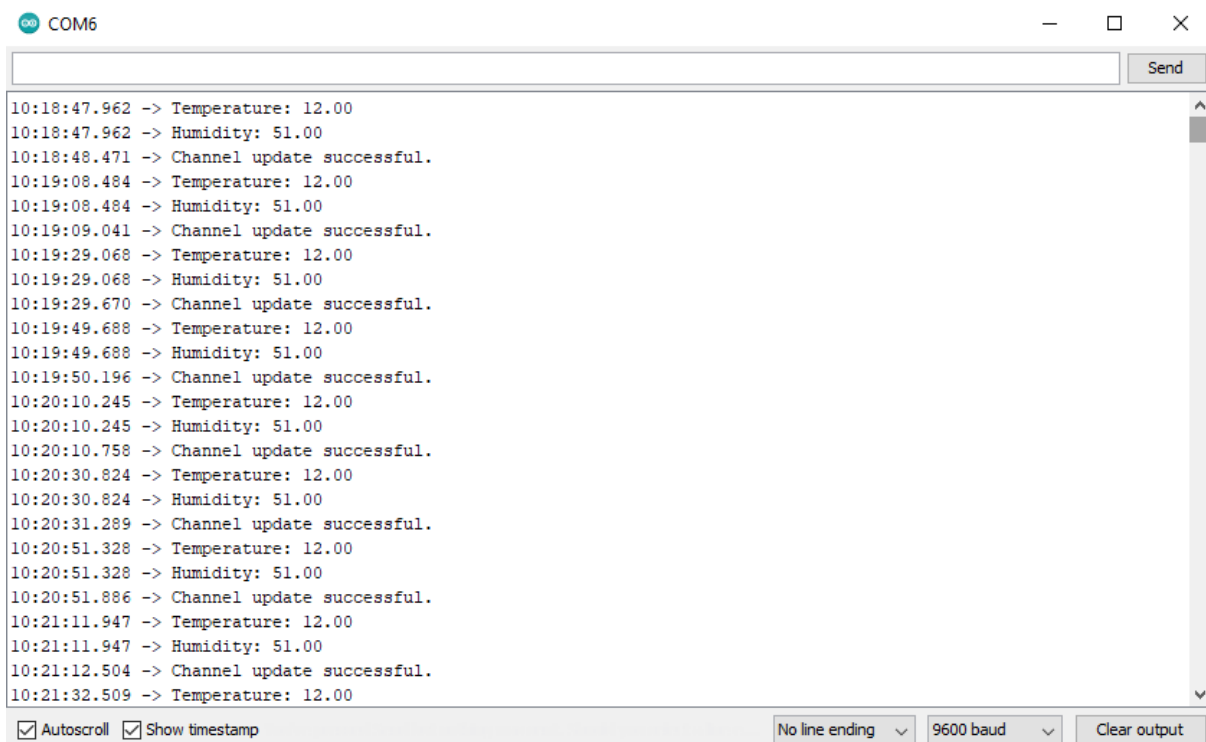
```

void setup() {
  Serial.begin(9600);
  //Σύνδεση με WiFi
  WiFi.begin("Your SSID", "Your Pass");
  while(WiFi.status() != WL_CONNECTED)
  {
    delay(200);
    Serial.print("..");
  }
  Serial.println();
  Serial.println("NodeMCU is connected!");
  Serial.println(WiFi.localIP());
  //Αρχιβοποίηση του ThingSpeak με τα στοιχεία του WiFi
  ThingSpeak.begin(client);
}

void loop() {
  int chk = DHT.read11(DHT11_PIN);
  float h = DHT.humidity;
  float t = DHT.temperature;
  Serial.println("Temperature: " + (String) t);
  Serial.println("Humidity: " + (String) h);
  /* Αποστολή των δεδομένων θερμοκρασίας και υγρασίας στα πεδία 1 και 2
του ThingSpeak* /
  ThingSpeak.setField(1, t);
  ThingSpeak.setField(2, h);
  int x = ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);
  if(x == 200){
    Serial.println("Channel update successful.");
  }
  else{
    Serial.println("Problem updating channel. HTTP error code " +
String(x));
  }
  delay(20000);
}

```

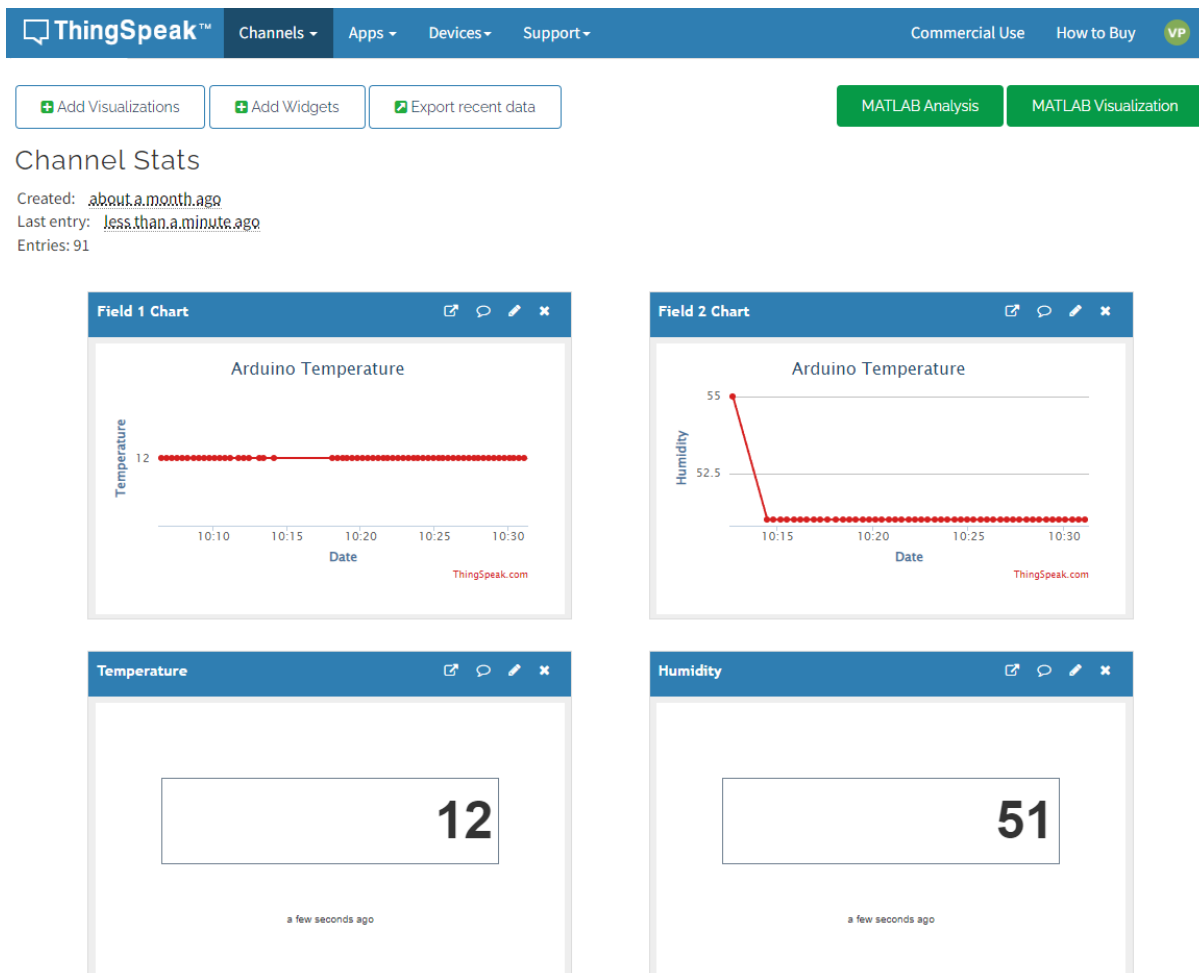

Ο κώδικας αυτός αρχικά συνδέει το NodeMcu στο WiFi και κατόπιν στέλνει τα δεδομένα στην σελίδα ThingSpeak. Στο <https://thingspeak.com/> μπορούμε να δημιουργήσουμε ένα Channel το οποίο θα έχει τα δικά του πεδία, γραφικές αναπαραστάσεις και API keys ώστε να μπορούμε να κάνουμε τα δικά μας updates. Οπότε στον κώδικα χρειάζεται να προσθέσουμε το όνομα και τον κωδικό του WiFi καθώς και το Channel ID και το Write API Key του ThingSpeak. Τέλος κάνουμε upload τον κώδικα στο NodeMcu επιλέγοντας στα tools ως board το “NodeMCU 1.0 (ESP-12E Module)” και ανοίγοντας το Serial Monitor του IDE παίρνουμε τα εξής αποτελέσματα:



```
COM6
10:18:47.962 -> Temperature: 12.00
10:18:47.962 -> Humidity: 51.00
10:18:48.471 -> Channel update successful.
10:19:08.484 -> Temperature: 12.00
10:19:08.484 -> Humidity: 51.00
10:19:09.041 -> Channel update successful.
10:19:29.068 -> Temperature: 12.00
10:19:29.068 -> Humidity: 51.00
10:19:29.670 -> Channel update successful.
10:19:49.688 -> Temperature: 12.00
10:19:49.688 -> Humidity: 51.00
10:19:50.196 -> Channel update successful.
10:20:10.245 -> Temperature: 12.00
10:20:10.245 -> Humidity: 51.00
10:20:10.758 -> Channel update successful.
10:20:30.824 -> Temperature: 12.00
10:20:30.824 -> Humidity: 51.00
10:20:31.289 -> Channel update successful.
10:20:51.328 -> Temperature: 12.00
10:20:51.328 -> Humidity: 51.00
10:20:51.886 -> Channel update successful.
10:21:11.947 -> Temperature: 12.00
10:21:11.947 -> Humidity: 51.00
10:21:12.504 -> Channel update successful.
10:21:32.509 -> Temperature: 12.00
```

Εικόνα 18: Αποτελέσματα αποστολής θερμοκρασίας και υγρασίας στο Cloud

Οι τιμές αυτές αντίστοιχα γίνονται update στο ThingSpeak:

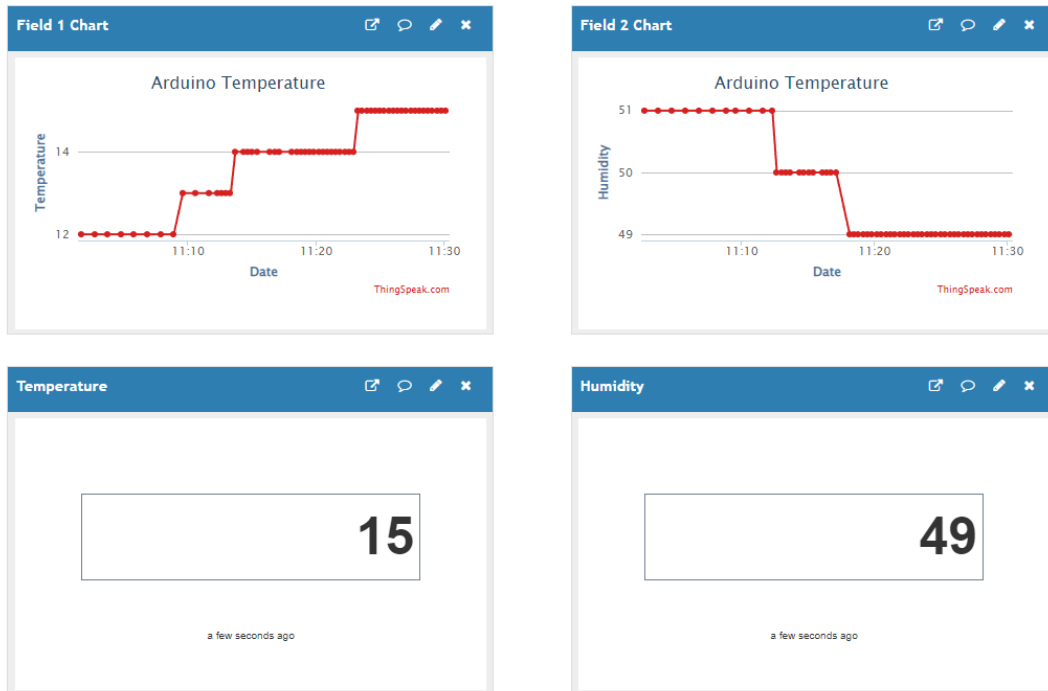


Εικόνα 19: Γραφικά αποτελέσματα θερμοκρασίας και υγρασίας στο Cloud

Στην συνέχεια ανοίγοντας τον κλιματισμό του χώρου μπορούμε να δούμε την ανανέωση των αποτελεσμάτων που στέλνονται στο cloud:

Channel Stats

Created: [about a month ago](#)
Last entry: [less than a minute ago](#)
Entries: 263



Εικόνα 20: Εναλλαγή των γραφικών αποτελεσμάτων θερμοκρασίας και υγρασίας στο Cloud

3.5 Αποστολή email alerts

Για την ολοκλήρωση του συστήματος, θα προσθέσουμε την λειτουργία της αποστολής email στην περίπτωση που η θερμοκρασία ξεφεύγει των 20° C. Για την πραγματοποίηση αυτής της λειτουργίας θα δημιουργήσουμε έναν νέο λογαριασμό gmail ο οποίος θα είναι και ο αποστολέας. Οπότε όταν στο loop που έχουμε δημιουργήσει, θα υπάρξει διαφορετική τιμή από τους 20° C τότε μέσω του NodeMCU θα στέλνεται από τον νέο λογαριασμό email alert. Ο κώδικας που χρησιμοποιήθηκε είναι ο παρακάτω:

```
#include <ESP_Mail_Client.h>
#include <ESP_Mail_FS.h>
#include <SDK_Version_Common.h>

#include <ESP8266WiFi.h>
#include <dht.h>
```

```

#include <ThingSpeak.h>
// Ορισμός των στοιχείων του WiFi για σύνδεση
#define WIFI_SSID "Your SSID"
#define WIFI_PASSWORD "Your pass"
// Ορισμός του host που θα χρησιμοποιήσουμε δηλαδή το gmail
#define SMTP_HOST "smtp.gmail.com"
#define SMTP_PORT 465

// Ορισμός του νέου λογαριασμού gmail που θα στέλνει ειδοποιήσεις
#define AUTHOR_EMAIL "your new gmail account"
#define AUTHOR_PASSWORD "your new password"

// Ορισμός του λήπτη των ειδοποιήσεων
#define RECIPIENT_EMAIL "your preferred recipient"

/* Το SMTP Session object που θα χρησιμοποιηθεί για την αποστολή των
email */
SMTPSession smtp;

// Ορισμός της κλάσης μηνύματος
SMTP_Message message;

// Η μέθοδος που θα στέλνει την κατάσταση αποστολής
void smtpCallback(SMTP_Status status);

dht DHT;
#define DHT11_PIN D4

WiFiClient client;

long myChannelNumber = yourChannelID;
const char myWriteAPIKey[] = "yourWriteAPIKey";

void setup() {
  // Σύνδεση με WiFi
  Serial.begin(9600);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  while(WiFi.status() != WL_CONNECTED)
  {
    delay(200);
    Serial.print("..");
  }
  Serial.println();
}

```

```

Serial.println("NodeMCU is connected!");
Serial.println(WiFi.localIP());
ThingSpeak.begin(client);

/* Ορισμός της μεθόδου που θα επιστρέφει πληροφορίες για την κατάσταση του μηνύματος */
smtp.callback(smtpCallback);

// Ορισμός του session αποστολής email
ESP_Mail_Session session;

// Ορισμός των επιμέρους χαρακτηριστικών του session
session.server.host_name = SMTP_HOST;
session.server.port = SMTP_PORT;
session.login.email = AUTHOR_EMAIL;
session.login.password = AUTHOR_PASSWORD;
session.login.user_domain = "";

// Ορισμός των κεφαλίδων του μηνύματος
message.sender.name = "ESP";
message.sender.email = AUTHOR_EMAIL;
message.subject = "ESP Alert Email";
message.addRecipient("Vasileios", RECIPIENT_EMAIL);

// Σύνδεση στον server μέσω του session
if (!smtp.connect(&session))
    return;
}

void loop() {

    int chk = DHT.read11(DHT11_PIN);
    float h = DHT.humidity;
    float t = DHT.temperature;

    if(t!=20){
        // Ο ορισμός του HTML μηνύματος που θα στέλνεται
        String htmlMsg = "<div style=\"color:#2f4468;\"><h1>Hello World!</h1><p>- Sent from ESP board: Temperature needs caution:!"</p></div>" + (String)t + "° C";
        message.html.content = htmlMsg.c_str();
        message.html.content = htmlMsg.c_str();
    }
}

```

```

    message.text.charset = "us-ascii";
    message.html.transfer_encoding =
Content_Transfer_Encoding::enc_7bit;

    // Εκκίνηση αποστολής email
    if (!MailClient.sendMail(&smtp, &message))
        Serial.println("Error sending Email, " + smtp.errorReason());
}

Serial.println("Temperature: " + (String) t);
Serial.println("Humidity: " + (String) h);
ThingSpeak.setField(1, t);
ThingSpeak.setField(2, h);
int x = ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);
if(x == 200){
    Serial.println("Channel update successful.");
}
else{
    Serial.println("Problem updating channel. HTTP error code " +
String(x));
}
delay(20000);
}

// Η μέθοδος που θα επιστρέφει την κατάσταση της αποστολής email
void smtpCallback(SMTP_Status status){

    Serial.println(status.info());

    if (status.success()){
        Serial.println("-----");
        ESP_MAIL_PRINTF("Message sent success: %d\n",
status.completedCount());
        ESP_MAIL_PRINTF("Message sent failed: %d\n",
status.failedCount());
        Serial.println("-----\n");
        struct tm dt;

        for (size_t i = 0; i < smtp.sendingResult.size(); i++){

            SMTP_Result result = smtp.sendingResult.getItem(i);
            time_t ts = (time_t)result.timestamp;
            localtime_r(&ts, &dt);

```

```

// Προβολή πληροφοριών σχετικά με το μήνυμα που εστάλη
ESP_MAIL_PRINTF("Message No: %d\n", i + 1);
ESP_MAIL_PRINTF("Status: %s\n", result.completed ? "success" :
"failed");
ESP_MAIL_PRINTF("Date/Time: %d/%d/%d %d:%d:%d\n", dt.tm_year +
1900, dt.tm_mon + 1, dt.tm_mday, dt.tm_hour, dt.tm_min, dt.tm_sec);
ESP_MAIL_PRINTF("Recipient: %s\n", result.recipients);
ESP_MAIL_PRINTF("Subject: %s\n", result.subject);
}
Serial.println("-----\n");
}
}

```

3.5 Γεωγραφικός εντοπισμός μέσω GPS

Μια σημαντική προσθήκη στο σύστημα που θα αναβάθμιζε την αξιοπιστία του αλλά και ίσως σε πρακτικό επίπεδο βοηθούσε σε μεγάλη εξοικονόμηση πόρων κατά την μεταφορά, είναι ο εντοπισμός του φορτίου στον χάρτη και η απεικόνιση της θερμοκρασίας του εκείνη την δεδομένη στιγμή. Αυτό μπορεί να πραγματοποιηθεί με ένα απλό GPS Module που θα συνδεθεί στο NodeMCU και το οποίο κατόπιν θα στέλνει γεωγραφικό μήκος, γεωγραφικό πλάτος και τους δείκτες θερμοκρασίας και υγρασίας στο Cloud. Στην συνέχεια, αφού δημιουργήσουμε τις δύο νέες μεταβλητές για γεωγραφικό μήκος και γεωγραφικό πλάτος στο ThingSpeak, δημιουργούμε ένα MATLAB Visualization και προσθέτουμε τον παρακάτω κώδικα:

```

/* Ορισμός του γεωγραφικού μήκους και πλάτους και της θερμοκρασίας όπως
έχουμε ορίσει τις μεταβλητές στο ThingSpeak. Στην προκειμένη περίπτωση το πεδίο 3
είναι το γεωγραφικό πλάτος, το πεδίο 4 το γεωγραφικό μήκος και η θερμοκρασία είναι
το πεδίο 1 */

```

```

lat =
thingSpeakRead(yourChannelID, 'Fields', 3, 'ReadKey', 'yourReadKey', 'N
umPoints', 3, 'Timeout', 50);
lon = thingSpeakRead(
, 'Fields', 4, 'ReadKey', 'yourReadKey', 'NumPoints', 3, 'Timeout', 50);
temp =
thingSpeakRead(yourChannelID, 'Fields', 1, 'ReadKey', 'yourReadKey', 'N
umPoints', 3, 'Timeout', 50);

```

```

/* Μορφοποίηση των χαρτών. Το 'red' δηλώνει χρώμα στίγματος, το
'filled' να είναι ο κύκλος κόκκινος και στο εσωτερικό εκτός από το
περίγραμμα */

```

```

geoscatteer(lat, lon, temp, 'red', 'filled');

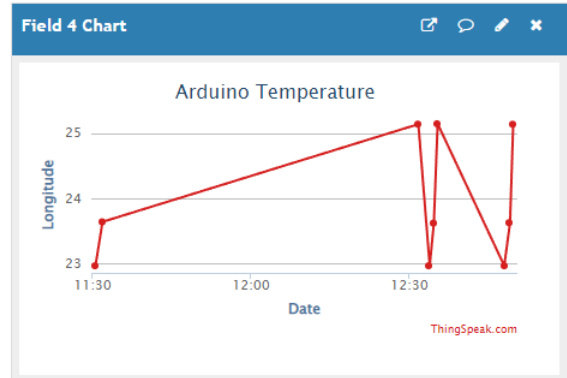
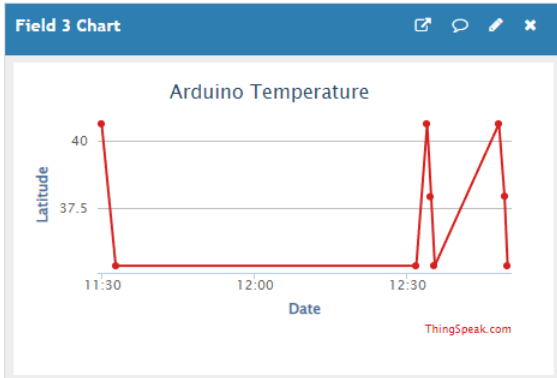
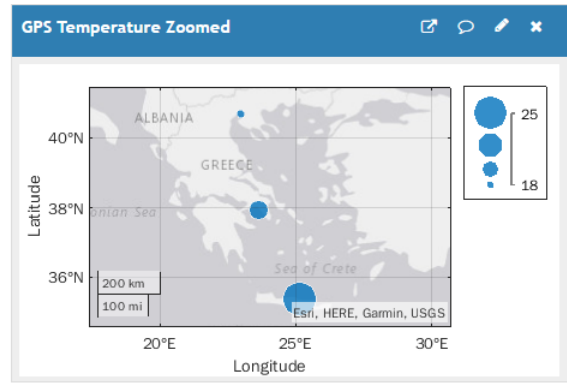
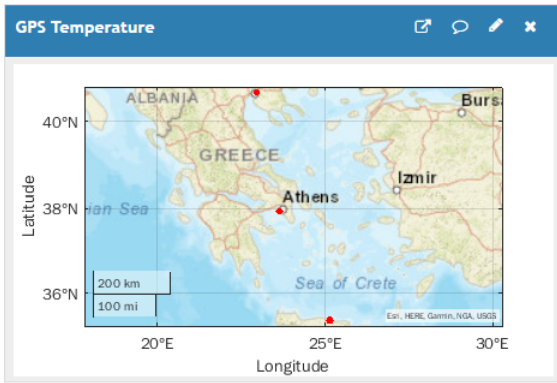
```

```
// Η έκδοση χάρτη που θα χρησιμοποιηθεί είναι η 'streets'  
geobasemap('streets');
```

Για να δημιουργήσουμε μια ακόμη γεωγραφική απεικόνιση αλλά περισσότερο εστιασμένη μπορούμε να χρησιμοποιήσουμε τον παρακάτω κώδικα:

```
lat =  
thingSpeakRead(yourChannelID, 'Fields', 3, 'ReadKey', 'yourReadKey', 'N  
umPoints', 3, 'Timeout', 50);  
lon =  
thingSpeakRead(yourChannelID, 'Fields', 4, 'ReadKey', 'yourReadKey', 'N  
umPoints', 3, 'Timeout', 50);  
temp =  
thingSpeakRead(yourChannelID, 'Fields', 1, 'ReadKey', 'yourReadKey', 'N  
umPoints', 3, 'Timeout', 50);  
geobasemap('streets');  
geobubble(lat, lon, temp, zoom=5);
```

Το τελικό αποτέλεσμα θα είναι δύο χάρτες όπου κάθε δεδομένη στιγμή θα μπορεί ο χρήστης να δει την τοποθεσία του φορτίου καθώς και την τιμή που έχει η θερμοκρασία στην συγκεκριμένη τοποθεσία και στιγμή. Στον εστιασμένο χάρτη, όσο μεγαλύτερη είναι η θερμοκρασία που εντοπίζεται, τόσο μεγαλύτερο θα είναι και το στίγμα στο χάρτη.



Εικόνα 21: Γραφικά αποτελέσματα GPS στο Cloud

4. ΜΕΛΕΤΗ ΠΕΡΙΠΤΩΣΗΣ & ΑΠΟΤΕΛΕΣΜΑΤΑ

Για τον συνολικό έλεγχο της λειτουργίας του συστήματος που δημιουργήθηκε, θα ενοποιηθούν οι κώδικες που γράφτηκαν για την κάθε λειτουργία σε ένα ενιαίο πρόγραμμα το οποίο:

- a. Θα εμφανίζει την θερμοκρασία και την υγρασία
- b. Θα στέλνει αυτές τις τιμές στο cloud.
- c. Θα ελέγχει τις τιμές της θερμοκρασίας και εφόσον είναι κάτω από 20° C θα ανοίγει τον κλιματισμό, ενώ αν είναι πάνω από 20° C θα τον κλείνει.

Για να πετύχουμε την ενοποίηση χρειάζεται να λειτουργούν ταυτόχρονα το Arduino Uno που χειρίζεται το IR Transmitter και στέλνει το σήμα στο κλιματιστικό και το NodeMcu το οποίο θα στέλνει τις θερμοκρασίες στο cloud. Τα συνδέουμε και τα 2 στον υπολογιστή και αντίστοιχα έχουν ports τα COM5 και COM6. Αφού έχουμε κάνει upload τα 2 αντίστοιχα κομμάτια κώδικα στο καθένα, ανοίγουμε το Serial Monitor για το COM6 το οποίο θα διαβάζει την θερμοκρασία και την υγρασία μέσω του NodeMcu και θα μας ενημερώνει αν τα δεδομένα αποστέλλονται επιτυχώς στο cloud. Στο COM5 που αντιστοιχεί στο Arduino, επίσης μπορούμε να δούμε την θερμοκρασία και την υγρασία αφού το έχουμε συνδεμένο κι αυτό στον αισθητήρα αλλά δεν μας προσφέρει κάποια παραπάνω πληροφορία οπότε απλά το συνδέουμε για να στέλνει τα κατάλληλα IR σήματα.

Οπότε ανοίγοντας αρχικά το Serial Monitor στο COM6 βλέπουμε ότι η θερμοκρασία είναι 19° C και η υγρασία 46. Οι τιμές στέλνονται στο cloud ενώ αντίστοιχα το Arduino τις διαβάζει και μέσω του ελέγχου που κάνουμε στον κώδικα, στέλνεται IR σήμα στο κλιματιστικό και ανοίγει.

```
COM6
13:33:12.471 -> .....
13:33:16.803 -> NodeMCU is connected!
13:33:16.803 -> 192.168.1.11
13:33:16.849 -> Temperature: 19.00
13:33:16.849 -> Humidity: 46.00
13:33:17.316 -> Channel update successful.
```

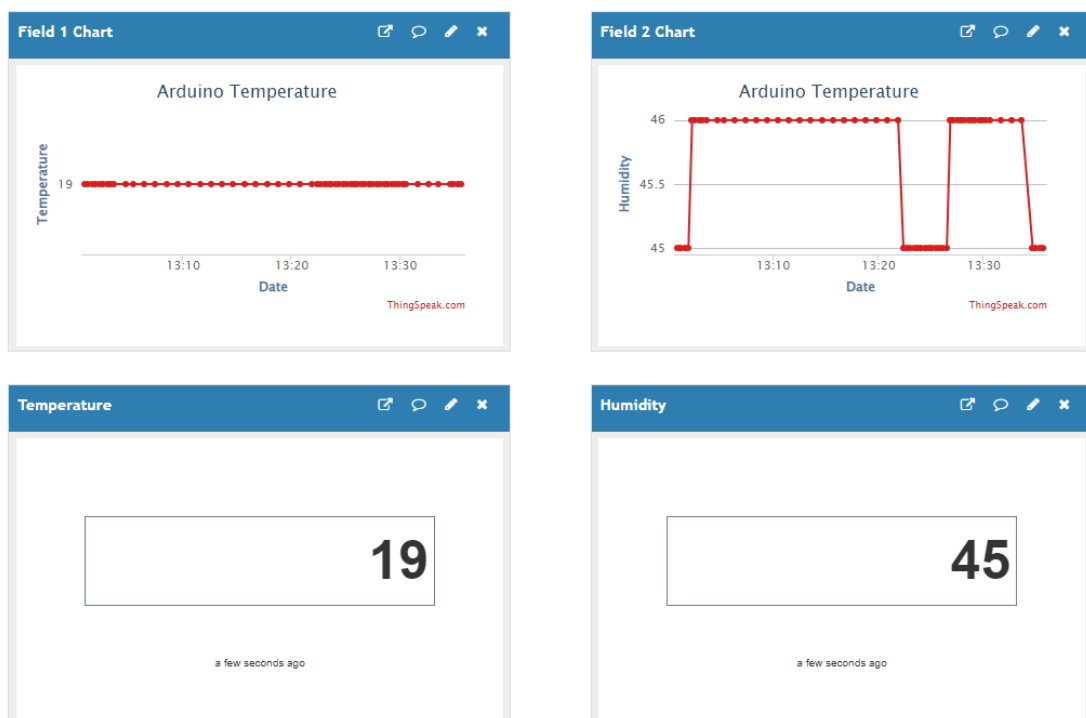
Autoscroll Show timestamp No line ending 9600 baud Clear output

Εικόνα 22: Αποτελέσματα επιτυχούς σύνδεσης του NodeMCU στο Wifi και ανανέωσης των αποτελεσμάτων στο Cloud στο Arduino IDE Serial Monitor για την μελέτη περίπτωσης

Οι τιμές στέλνονται στο ThingSpeak.

Channel Stats

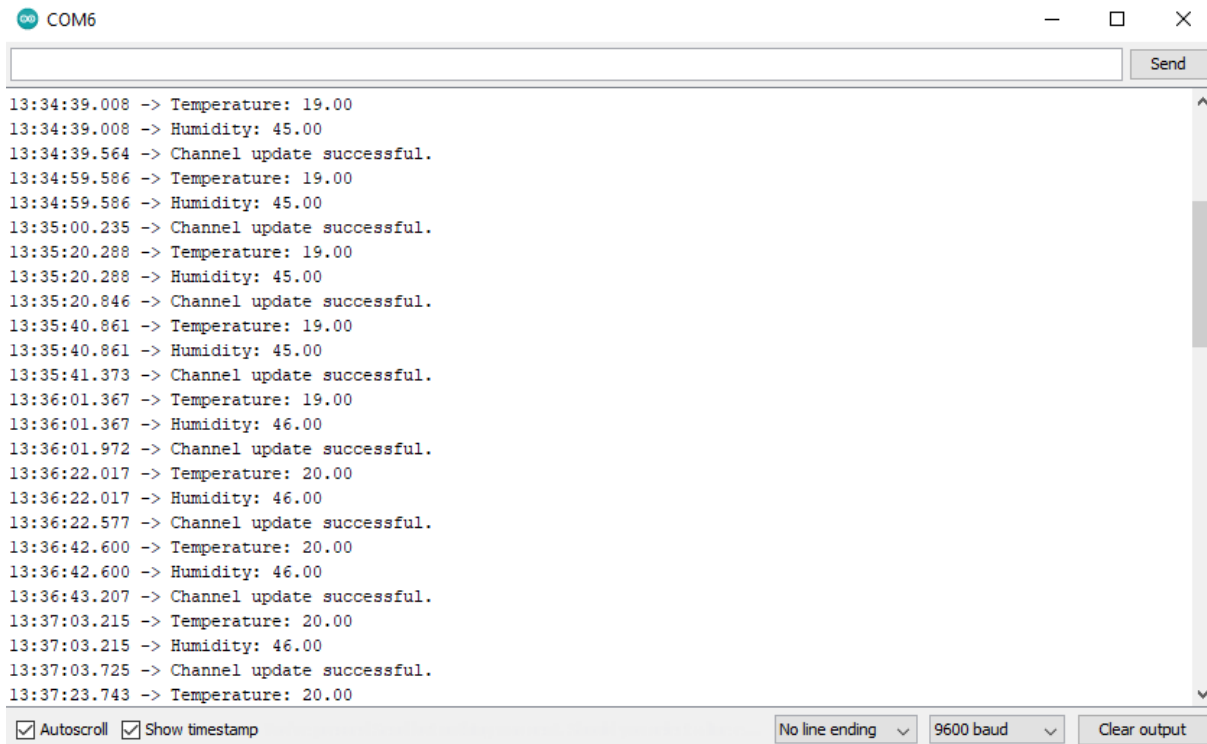
Created: [about a month ago](#)
Last entry: [less than a minute ago](#)
Entries: 629



Εικόνα 23: Απεικόνιση γραφικών αποτελεσμάτων θερμοκρασίας και υγρασίας στο Cloud για την μελέτη περίπτωσης

Οι πρώτες τιμές θερμοκρασίας που στάλθηκαν ενώ ταυτόχρονα ξεκίνησε να δουλεύει και το Arduino στέλνοντας το IR σήμα ήταν περίπου στις 13:00. Οπότε αναμένουμε τον χώρο να θερμανθεί και μόλις ξεπεράσει τους 20° C τότε θα πρέπει να σταλεί σήμα που να απενεργοποιεί τον κλιματισμό.

Παρατηρούμε αρχικά την αύξηση των βαθμών στους 20° C:



```
COM6
13:34:39.008 -> Temperature: 19.00
13:34:39.008 -> Humidity: 45.00
13:34:39.564 -> Channel update successful.
13:34:59.586 -> Temperature: 19.00
13:34:59.586 -> Humidity: 45.00
13:35:00.235 -> Channel update successful.
13:35:20.288 -> Temperature: 19.00
13:35:20.288 -> Humidity: 45.00
13:35:20.846 -> Channel update successful.
13:35:40.861 -> Temperature: 19.00
13:35:40.861 -> Humidity: 45.00
13:35:41.373 -> Channel update successful.
13:36:01.367 -> Temperature: 19.00
13:36:01.367 -> Humidity: 46.00
13:36:01.972 -> Channel update successful.
13:36:22.017 -> Temperature: 20.00
13:36:22.017 -> Humidity: 46.00
13:36:22.577 -> Channel update successful.
13:36:42.600 -> Temperature: 20.00
13:36:42.600 -> Humidity: 46.00
13:36:43.207 -> Channel update successful.
13:37:03.215 -> Temperature: 20.00
13:37:03.215 -> Humidity: 46.00
13:37:03.725 -> Channel update successful.
13:37:23.743 -> Temperature: 20.00
```

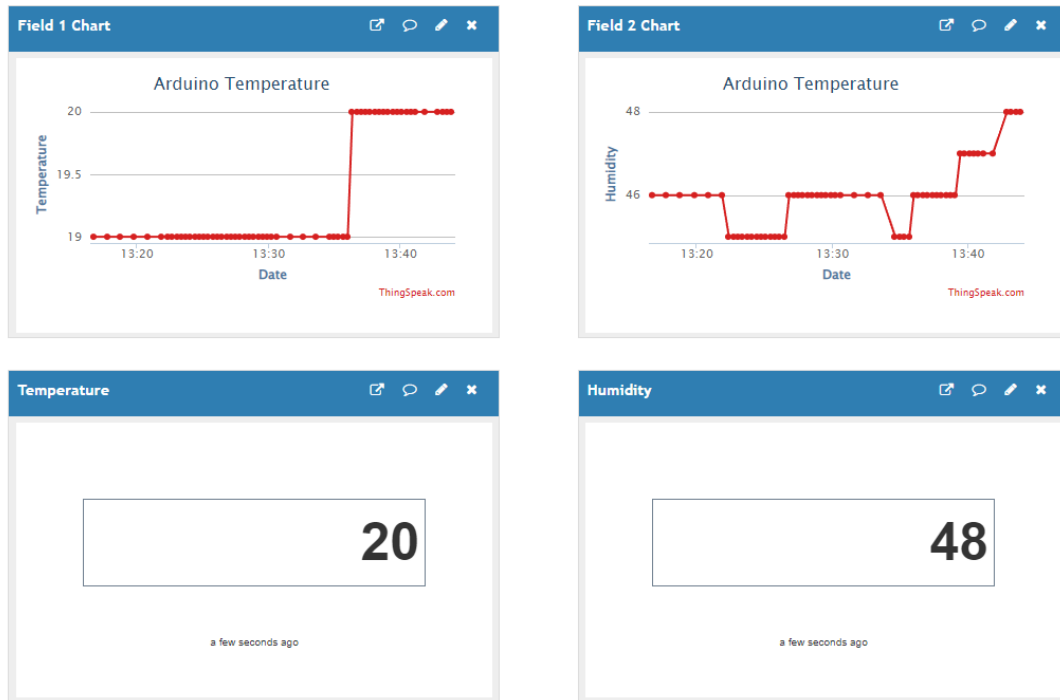
Εικόνα 24: Αποτελέσματα επιτυχούς ανανέωσης αποτελεσμάτων θερμοκρασίας και υγρασίας στο Arduino IDE Serial Monitor για την μελέτη περίπτωσης

Channel Stats

Created: [about a month ago](#)

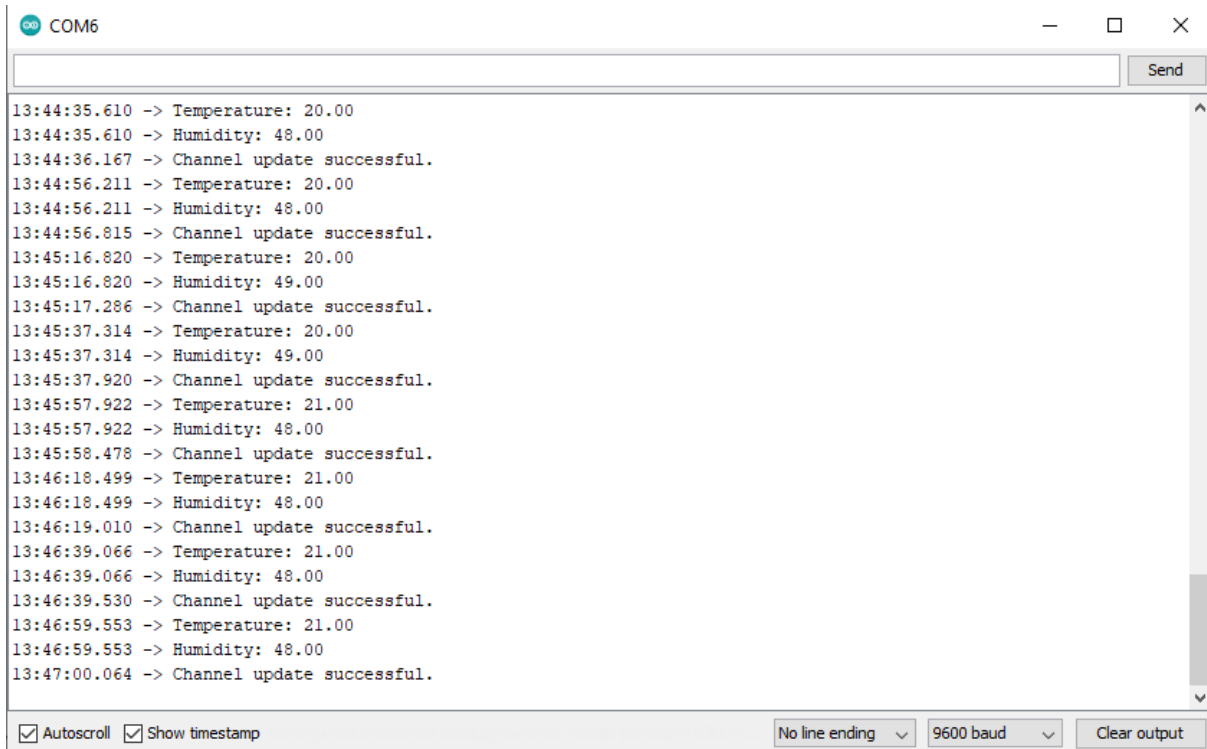
Last entry: [less than a minute ago](#)

Entries: 653



Εικόνα 25: Αποτελέσματα επιτυχούς ανανέωσης γραφικών αποτελεσμάτων θερμοκρασίας και υγρασίας στο Cloud για την μελέτη περίπτωσης

Τέλος όταν ο χώρος πιάσει τους 21° C, το Arduino στέλνει το σήμα και ο κλιματισμός απενεργοποιείται.



The screenshot shows the Serial Monitor window for COM6. The output text is as follows:

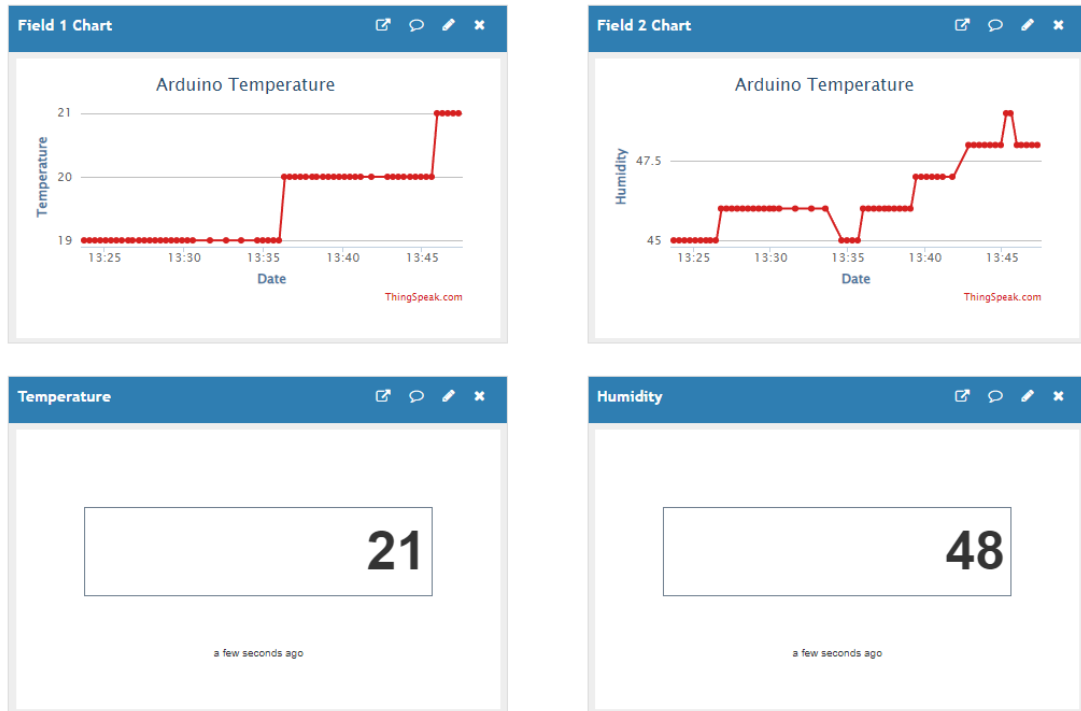
```
13:44:35.610 -> Temperature: 20.00
13:44:35.610 -> Humidity: 48.00
13:44:36.167 -> Channel update successful.
13:44:56.211 -> Temperature: 20.00
13:44:56.211 -> Humidity: 48.00
13:44:56.815 -> Channel update successful.
13:45:16.820 -> Temperature: 20.00
13:45:16.820 -> Humidity: 49.00
13:45:17.286 -> Channel update successful.
13:45:37.314 -> Temperature: 20.00
13:45:37.314 -> Humidity: 49.00
13:45:37.920 -> Channel update successful.
13:45:57.922 -> Temperature: 21.00
13:45:57.922 -> Humidity: 48.00
13:45:58.478 -> Channel update successful.
13:46:18.499 -> Temperature: 21.00
13:46:18.499 -> Humidity: 48.00
13:46:19.010 -> Channel update successful.
13:46:39.066 -> Temperature: 21.00
13:46:39.066 -> Humidity: 48.00
13:46:39.530 -> Channel update successful.
13:46:59.553 -> Temperature: 21.00
13:46:59.553 -> Humidity: 48.00
13:47:00.064 -> Channel update successful.
```

The interface includes a 'Send' button at the top right, a scroll bar on the right, and a control bar at the bottom with the following options: Autoscroll, Show timestamp, a dropdown menu for 'No line ending', a dropdown menu for '9600 baud', and a 'Clear output' button.

Εικόνα 26: Αποτελέσματα 2ης επιτυχούς ανανέωσης αποτελεσμάτων θερμοκρασίας και υγρασίας στο Arduino IDE Serial Monitor για την μελέτη περίπτωσης

Channel Stats


Created: [about a month ago](#)
Last entry: [less than a minute ago](#)
Entries: 663




Εικόνα 27: Αποτελέσματα 2ης επιτυχούς ανανέωσης γραφικών αποτελεσμάτων θερμοκρασίας και υγρασίας στο Cloud για την μελέτη περίπτωσης


Η θερμοκρασία αρχίζει να πέφτει ξανά και όταν πέσει στους 19° C ο κλιματισμός ανοίγει ξανά. Η παρακολούθηση αυτών των αλλαγών και του κλιματισμού του χώρου πλέον, ελέγχεται ζωντανά και απομακρυσμένα μέσω του Thingspeak. Επίσης μπορούμε να δούμε και τα αντίστοιχα email alerts που στέλνονται στην περίπτωση που η θερμοκρασία δεν είναι στους 20° C:

ESP Alert Email Εισερχόμενα x

 **ESP**
Hello World! - Sent from ESP board: Temperature needs caution!

26



 **ESP**
Hello World! - Sent from ESP board: Temperature needs caution:! 19.00

 **ESP** <arduino-temperature.alert@gmail.com>
προς εγώ ▾

🇬🇷 Αγγλικά ▾ > Ελληνικά ▾ [Μετάφραση μηνύματος](#)

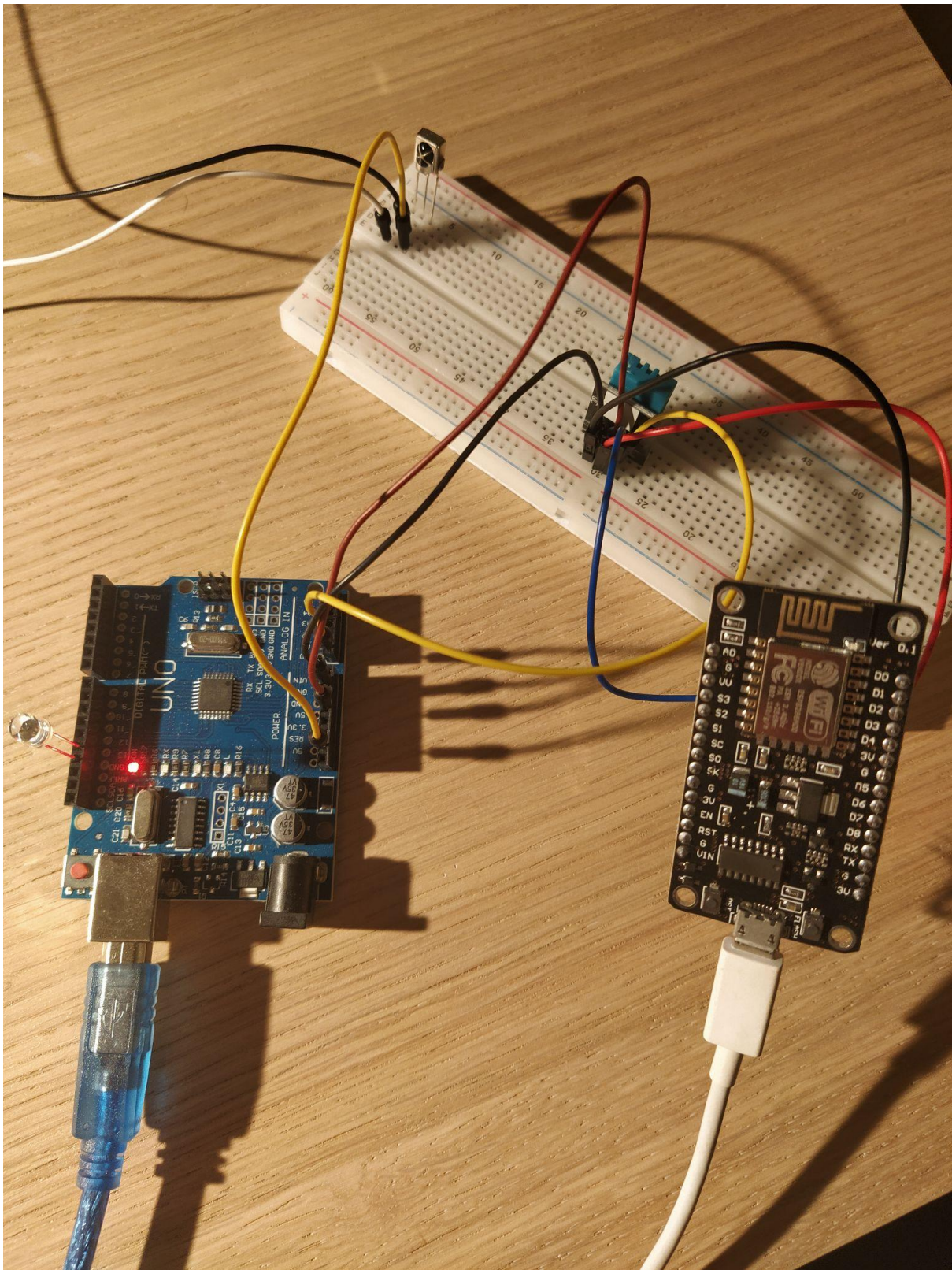
Hello World!

- Sent from ESP board: Temperature needs caution:!
19.00° C

 Απάντηση  Προώθηση

Εικόνα 28: Ειδοποιήσεις email για θερμοκρασία που είναι διαφορετική των 20° C

Η μορφή του τελικού συστήματος:



Εικόνα 29: Φωτογραφία ολοκληρωμένου συστήματος

Διακρίνουμε το Arduino Uno αριστερά, το NodeMCU δεξιά και στο πάνω μέρος το Breadboard όπου γίνονται οι συνδέσεις. Στο Breadboard διακρίνουμε τον IR Receiver στο αριστερό του μέρος, το οποίο είναι αποσυνδεδεμένο καθώς έχουν ήδη αποκωδικοποιηθεί τα σήματα του κλιματιστικού και τον αισθητήρα DHT11 ο οποίος συνδέεται ταυτόχρονα και στο Arduino και στο NodeMCU. Τέλος, διακρίνουμε τον IR Transmitter συνδεδεμένο απευθείας στο Arduino.

Τα αποτελέσματα της δοκιμής του συστήματος είναι μετρήσιμα και διαθέσιμα οποιαδήποτε στιγμή στο Cloud. Συγκεκριμένα, στην διεύθυνση <https://thingspeak.com/channels/1613446> έχει δημιουργηθεί το δημόσιο μέρος του καναλιού που αποθηκεύονται τα αποτελέσματα και είναι διαθέσιμα σε γραφικής αναπαράστασης.

5. ΜΕΛΛΟΝΤΙΚΗ ΕΡΕΥΝΑ

Η δημιουργία αυτού του συστήματος είναι μόνο μία αρχή για τον κόσμο της ψυχρής αλυσίδας και του ΙΟΤ. Το ίδιο το σύστημα μπορεί να επεκταθεί σε πολύ μεγάλο βαθμό αναβαθμίζοντας την αποτελεσματικότητα και την ακρίβεια του.

Ξεκινώντας από τον έλεγχο των αποτελεσμάτων, μια μελλοντική σημαντική προσθήκη θα ήταν η χρήση του αλγορίθμου K-means για τον καλύτερο έλεγχο της θερμοκρασίας και της υγρασίας. Ο αλγόριθμος K-means αποτελεί μια μέθοδο ανίχνευσης ανωμαλιών στα αποτελέσματα. Βοηθά στην κατανόηση και την ανάλυση των αποτελεσμάτων, ανιχνεύοντας τότε μια τιμή είναι πέραν των ορίων των υπολοίπων και έτσι μπορεί να προβλέψει τότε μπορεί να δημιουργηθεί πρόβλημα στην ασφάλεια της μεταφοράς των προϊόντων. Όσο υψηλότερος είναι ο συντελεστής διακύμανσης του, τόσο μεγαλύτερη είναι η αλλαγή στα περιβαλλοντικά δεδομένα. Μια περαιτέρω αναβάθμιση θα ήταν η χρησιμοποίηση της πλατφόρμας EdgeImpulse για την ανάλυση των αποτελεσμάτων μέσω μηχανικής μάθησης (Machine Learning). Η πλατφόρμα εφαρμόζει αλγορίθμους ML στα ακατέργαστα δεδομένα που δέχεται έτσι ώστε να διευκολύνει την βελτιστοποίηση του συστήματος, ανιχνεύοντας τυχόν ανωμαλίες στα αποτελέσματα που ανεβαίνουν στο Cloud. Μόλις τα δείγματα είναι αρκετά ώστε το μοντέλο να κάνει ακριβείς εκτιμήσεις τότε υπάρχει δυνατότητα να ορίσουμε την έξοδο του ως την είσοδο για τον αλγόριθμο ανίχνευσης ανωμαλιών K-means που βοηθά στην εύρεση ακραίων τιμών σε άγνωστα δεδομένα.

Στον τομέα της τεχνολογίας και της ασφάλειας, μπορεί να δοκιμαστεί η σύνδεση της παρούσας τεχνολογίας με την τεχνολογία του blockchain. Μέσω αποκεντρωμένης και αξιόπιστης τεχνολογίας blockchain και χρησιμοποιώντας τα λεγόμενα έξυπνα συμβόλαια (smart contracts), η πλατφόρμα θα μπορεί να εγγυάται την προέλευση, την ακεραιότητα, τη διαφάνεια και την αξιοπιστία των δεδομένων καθώς γνωρίζουμε ότι για να φτάσει στον τελικό καταναλωτή κάθε προϊόν υπάρχει ένα σύνθετο δίκτυο παραγωγής και διανομής. Δεδομένου ότι η ασφάλεια των πληροφοριών διαδραματίζει κρίσιμο ρόλο στην φαρμακευτική ψυχρή αλυσίδα, η μελλοντική έρευνα θα μπορούσε να επιδιώξει το σχεδιασμό πρωτοκόλλων για τη διαχείριση ιδιωτικών κρυπτογραφικών κλειδιών.

Στον τομέα του hardware, αρχικά, το σύστημα μπορεί να αναβαθμιστεί με τον έλεγχο των IR σημάτων και ενός αφυγραντήρα ώστε να γίνεται αυτόματη ρύθμιση των περιβαλλοντικών συνθηκών όπως ακριβώς έγινε με την θερμοκρασία. Μετέπειτα, υπάρχουν πολλές δυνατότητες για χρησιμοποίηση ακόμη ποιοτικότερου hardware και κυρίως αισθητήρων ώστε να επιτευχθεί μεγαλύτερη ακρίβεια στις τιμές, να υπάρχει δυνατότητα λειτουργίας σε ακραίες συνθήκες θερμοκρασίας όπως π.χ. στην μεταφορά εμβολίων για τον COVID-19 όπου η ψύξη πρέπει να παραμένει σε ακραίες τιμές και τέλος, να λαμβάνονται επιπλέον πληροφορίες όπως η ύπαρξη σωματιδίων στην ατμόσφαιρα, η ακτινοβολία και οτιδήποτε άλλο μπορεί να προκαλέσει αλλοίωση στην ποιότητα των προϊόντων.

Στον τομέα του software, χρησιμοποιώντας την συσκευή bluetooth που παρουσιάστηκε στην παρούσα εργασία, μπορεί να σχεδιαστεί μια εφαρμογή κινητού η οποία να προσομοιάζει ένα τηλεχειριστήριο και με το πάτημα κάθε πλήκτρου να στέλνεται το αντίστοιχο IR σήμα για μια συγκεκριμένη ενέργεια στην συσκευή bluetooth και κατόπιν μέσω του IR Transmitter να στέλνεται στο κλιματιστικό. Τέλος, θα πρέπει να διαμορφωθεί και να εφαρμοστεί ένα πλαίσιο δοκιμών που να λαμβάνει υπόψιν τη χρήση του συστήματος σε ένα πραγματικό και πολύπλοκο σύστημα μεταφορών που χρησιμοποιεί πραγματικά εμπορευματοκιβώτια τα οποία μπορούν να μεταφερθούν με διαφορετικούς τρόπους μεταφοράς (πλοία, φορτηγά, τρένα).

6. ΣΥΜΠΕΡΑΣΜΑΤΑ

Η παρούσα εργασία παρέχει μια θεωρητική αλλά κυρίως πρακτική συμβολή στην πρόοδο της γνώσης σχετικά με την ψυχρή αλυσίδα, δεδομένου ότι λαμβάνει υπόψιν τις ρεαλιστικές ανάγκες του εμπορίου και των εφοδιαστικών αλυσίδων σύμφωνα με την πραγματική βιομηχανία και ιδιαίτερα σύμφωνα με τις νέες ανάγκες που προέκυψαν και έγιναν ευρέως γνωστές και θέμα συζήτησης παγκοσμίως σχετικά με την διανομή των εμβολίων για τον COVID-19. Είναι σημαντικό να τονιστεί η συμβολή αυτού του τομέα στην κοινωνία, καθώς η ποιότητα των τροφίμων και των ιατροφαρμακευτικών προϊόντων έχουν σημαντικό αντίκτυπο στη ζωή των ανθρώπων και αντιπροσωπεύουν σημαντικό μέρος του κόστους υγειονομικής περίθαλψης και βιοτικού επιπέδου γενικότερα. Το πρότυπο σύστημα που δημιουργήθηκε θα μπορούσε να βοηθήσει εταιρείες και ιδρύματα στη διαδικασία λήψης αποφάσεων. Η παρακολούθηση και ο έλεγχος της θερμοκρασίας των προϊόντων με ένα έξυπνο σύστημα ψυχρής αλυσίδας παρέχει πληροφορίες σε όλους τους παράγοντες που εμπλέκονται στην αλυσίδα διανομής και μπορούν να τις χρησιμοποιήσουν για τη βελτίωση της ανταπόκρισης της ψυχρής αλυσίδας, μελετώντας τις αδυναμίες και αποφασίζοντας ποια χαρακτηριστικά της θα βελτιώσουν.

Με το παρόν σύστημα, διερευνήσαμε τη χρήση συσκευών χαμηλού κόστους για την γρήγορη δημιουργία προτύπου συστήματος για την μεταφορά εμπορευμάτων. Συγκεκριμένα, το σύστημα εστιάζει στην ενσωμάτωση υλικού, λογισμικού και υπηρεσιών cloud για να υποστηρίξει τη γρήγορη δημιουργία λύσεων, ικανών να επιτύχουν την ασφάλεια των εμπορευμάτων μέσω χρήσης συσκευών IoT και Cloud. Η χρήση τεχνολογιών IoT στον βιομηχανικό τομέα μπορεί να μεγιστοποιήσει την παραγωγικότητα και να μειώσει το κόστος, δημιουργώντας παράλληλα νέες πηγές εσόδων. Οι τεχνολογίες IoT καθιστούν δυνατή τη δημιουργία έξυπνων συστημάτων για τη διαχείριση πολύπλοκων αλυσίδων εφοδιασμού, για την πρόβλεψη κινδύνων και την ελαχιστοποίηση των απωλειών. Επίσης, οι τεχνολογίες IoT μπορούν να χρησιμοποιηθούν για τη βελτιστοποίηση της κατανάλωσης ενέργειας και της χρήσης χώρου. Επιπλέον, είναι δυνατή η δημιουργία ασφαλών συνθηκών για τους εργαζόμενους και η εξοικονόμηση χρημάτων λόγω του αποτελεσματικού ελέγχου της θέρμανσης. Όλες οι εταιρείες που θέλουν να επωφεληθούν από το IoT χρησιμοποιώντας σχετικές συσκευές, υπηρεσίες και εφαρμογές θα πρέπει να εξετάσουν και την ασφάλεια των δεδομένων τους καθώς οι συσκευές IoT είναι συνδεδεμένες στο διαδίκτυο και, ως εκ τούτου, ενδέχεται να δεχτούν επίθεση από κακόβουλους χρήστες κλέβοντας δεδομένα, διακόπτοντας τη διανομή υπηρεσιών ή πραγματοποιώντας μη αξιόπιστες λειτουργίες.

Τέλος, για να δώσουμε μια ιδέα για το τι σημαίνει «χαμηλό κόστος», αναφέρεται ότι το συνολικό ποσό που δαπανήθηκε για κάθε συσκευή που αγοράστηκε για την υλοποίηση ολόκληρου του συστήματος είναι περίπου 60€. Αυτές οι συσκευές προορίζονται για γρήγορη υλοποίηση πρότυπων συστημάτων για την αξιολόγηση μια σχεδιαζόμενης στρατηγικής. Ο τομέας των logistics μπορεί να έχει πολύ μεγάλες δυνατότητες και ευκαιρίες από την χρησιμοποίηση καινοτόμων τεχνολογιών IoT για την αυτοματοποίηση, την απλοποίηση και τη βελτίωση των διαδικασιών του και οι προοπτικές της εξέλιξης του παγκόσμιου εμπορίου συνεχώς θα αμβλύνονται όσο εξελίσσονται και οι τεχνολογίες που αναφέρθηκαν και χρησιμοποιήθηκαν στην παρούσα εργασία.

BIBΛΙΟΓΡΑΦΙΑ

1. Aamer, Ammar & Al-Awlaqi, Mohammed & Affia, Ifadhila & Arumsari, Silvia & Mandahawi, Nabeel (2021). The internet of things in the food supply chain: adoption challenges. Benchmarking: An International Journal. ahead-of-print
2. Abdul Halim, A. H., Abdul Halim, M. H., Usman, S., & mohd daud, salwani (2021). Implementation of IoT and Blockchain for Temperature Monitoring in Covid19 Vaccine Cold Chain Logistics. Open International Journal of Informatics, 9(1), 78–87
3. Afreen, Hina & Bajwa, Imran (2021). An IoT-Based Real-Time Intelligent Monitoring and Notification System of Cold Storage. IEEE Access. PP. 1-1
4. Angeles, R. (2016), 'Kuehne + Nagel's PharmaChain: IoT-Enabled Product Monitoring Using Radio Frequency Identification', World Academy of Science, Engineering and Technology, Open Science Index 119, International Journal of Economics and Management Engineering, 10(11), 3602 - 3609
5. Baralla, Gavina & Pinna, Andrea & Tonelli, Roberto & Marchesi, Michele & Ibba, Simona (2020). Ensuring transparency and traceability of food local products. A blockchain application to a Smart Tourism Region. Concurrency and Computation Practice and Experience. 33
6. Bengiovanni, Teresa & Rosito, Roberto & Lacasa, Vincenzo & Simone, Eugenio & Sergi, Ilaria & Iacovone, Vincenzo & Viggiano, Michele & Patrono, Luigi (2020). Risk Management and Healthcare: IoT Technologies and Smart Monitoring System for a Good Cold Chain Management. 1-6
7. Bogataj, David & Bogataj, Marija & Hudoklin, Domen (2017). Reprint of “Mitigating risks of perishable products in the cyber-physical systems based on the extended MRP model”. International Journal of Production Economics. 194
8. Bouzembrak, Yamine & Klüche, Marcel & Gavai, Anand & Marvin, H.J.P. (2019). Internet of Things in food safety: Literature review and a bibliometric analysis. Trends in Food Science & Technology. 94
9. Chandra A.A., Lee S.R. (2014). A method of WSN and sensor cloud system to monitor cold chain logistics as part of the IoT technology. Int. J. Multimed. Ubiquitous Eng. 9, 145–152

10. Cui, Hongxiu (2021). Intelligent Coordination Distribution of the Whole Supply Chain Based on the Internet of Things. *Complexity*. 2021. 1-12
11. Cui, Yu (2018). Supply Chain Innovation with IoT
12. Dae-Sik Ko (2016). A Design on the Communication Protocol for Data Collection of the IOT Based on Cold Chain System. *International Journal of Multimedia and Ubiquitous Engineering*, Vol 9, p. 221-230
13. Feng, Huanhuan & Wang, Wensheng & Chen, Bingqi & Xiaoshuan, Zhang (2020). Evaluation on Frozen Shellfish Quality by Blockchain Based Multi-Sensors Monitoring and SVM Algorithm During Cold Storage. *IEEE Access*. PP. 1-1
14. Guillén, Antonio & Boada, Martí & Villarino, R. & Girbau, David (2019). NFC Sensors Based on Energy Harvesting for IoT Applications
15. Hasanat, Raisa & Rahman, MD & Mansoor, Nafees & Mohammed, Nabeel & Rahman, Shahriar & Rasheduzzaman, Mirza (2020). An IoT based Real-time Data-centric Monitoring System for Vaccine Cold Chain. 1-5
16. Hosseini Bamakan, Seyed Mojtaba & Ghasemzadeh, Shima & Manshadi, Sajedah (2021). Blockchain-enabled pharmaceutical cold chain: Applications, key challenges, and future trends. *Journal of Cleaner Production*. 302. 127021
17. Iftekhhar, Adnan & Cui, Xiaohui (2021). Blockchain-Based Traceability System That Ensures Food Safety Measures to Protect Consumer Safety and COVID-19 Free Supply Chains. *Foods*. 10. 1289
18. Jia-Wei Han, Min Zuo, Wen-Ying Zhu, Jin-Hua Zuo, En-Li Lü, Xin-Ting Yang (2021). A comprehensive review of cold chain logistics for fresh agricultural products: Current status, challenges, and future trends. *Trends in Food Science & Technology*, Vol. 109, p. 536-551
19. Karani, Krishna Prasad & Muraleedhar, Rajeshwari & Kirlaya, Vinayachandra & Poornima K, Geetha (2020). An Integrated Solution for Solar Cold Chain Portfolio Management using Internet of Things. 4. 112-130
20. Ko, Dasik & Kwak, Yunsik & Choi, Dojin & Song, Seokil (2015). Design of Cold Chain Application Framework (CCAF) Based on IOT and Cloud. 11-13
21. Kramar, Vadim & Maatta, Harri & Hinkula, Henry & Thorsen, Oistein & Cox, Georgina (2017). Smart-fish system for fresh fish cold chain transportation — Overall approach and selection of sensor materials. *Proceedings of the XXth Conference of Open Innovations Association FRUCT*. 562. 202-209
22. Kumar, Neeraj & Tyagi, Mohit & Sachdeva, Anish (2021). Depiction of possible solutions to improve the cold supply chain performance system. *Journal of Advances in Management Research*. ahead-of-print

23. Laurence, J. (2013), "Ammonia leak at Shanghai refrigeration plant kills 15, injures 26", Reuters, New York, NY, p. 1, available at: www.reuters.com/article/us-china-accident-ammonia-idUSBRE97U04420130831
24. Lawrence, Piers & Phippard, Trisha & Ramachandran, Gowri & Hughes, Danny (2017). Developing IoT to support the health sector: A case study from Kikwit, DR Congo
25. Li, Guie. (2021). Development of cold chain logistics transportation system based on 5G network and Internet of things system. *Microprocessors and Microsystems*. 80. 103565
26. Li, Xinwu & Yang, Lin & Duan, Yanqing & Wu, Zhigang & Xiaoshuan, Zhang (2019). Developing a Real-time Monitoring Traceability System for Cold Chain of *Tricholoma matsutake*. *Electronics*. 8. 423
27. Loisel, Julie & Duret, Steven & Cornuéjols, Antoine & Cagnon, Dominique & Tardet, Margot & Derens-Bertheau, Evelyne & Laguerre, Onrawee (2021). Cold chain break detection and analysis: Can machine learning help?. *Trends in Food Science & Technology*. 112. 391-399
28. Lorenc, Augustyn & Czuba, Michał & Szarata, Jakub. (2021). Big data analytics and anomaly prediction in the cold chain to supply chain resilience. *FME Transactions*. 49. 315-326
29. Luo, Heng & Zhu, Minjie & Ye, Sengang & Hou, Hanping & Chen, Yong & Bulysheva, Larisa (2016). An intelligent tracking system based on internet of things for the cold chain. *Internet Research*. 26. 435-445
30. Mendonça, Ronan & Gomes, Otávio & Vieira, Luiz & Vieira, Marcos & Borges, Alex & Miranda Nacif, José (2021). BlockColdChain: Vaccine Cold Chain Blockchain.
31. Mohsin, Afreen & Yellampalli, Siva (2017). IoT based cold chain logistics monitoring. 1971-1974
32. Monteleone S., Sampaio M. and Maia R. F. (2017), A novel deployment of smart Cold Chain system using 2G-RFID-Sys temperature monitoring in medicine Cold Chain based on Internet of Things, 2017 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI), p. 205-210
33. Mourtzis, Dimitris & Angelopoulos, John & Panopoulos, Nikos (2021). Design and development of an IoT enabled platform for remote monitoring and predictive maintenance of industrial equipment. *Procedia Manufacturing*. 54. 166-171
34. Mäkinen, T.M. and Hassi, J. (2009), "Health problems in cold work", *Industrial Health*, Vol. 47 No. 3, pp. 207-220.

35. Nerbovig, A. (2017), "Cold weather causes food spoilage at Lucky's market; replenishment expected Saturday evening", Billings Gazette, January 6, p. 1, available at:
http://billingsgazette.com/business/cold-weather-causes-food-spoilage-at-lucky-s-market-replenishment/article_1b7cdce1-ccf5-58d1-a724-e405efa1db5a.html
36. Onwude, Daniel & Chen, Guangnan & Eke-Emezies, Nnanna & Kabutey, Abraham & Alkhaled, Alfadhl & Sturm, Barbara (2020). Recent Advances in Reducing Food Losses in the Supply Chain of Fresh Agricultural Produce. *Processes*. 8
37. Ping, Hua & Wang, Jihua & Ma, Zhihong & Du, Yuanfang (2018). Mini-review of application of IoT technology in monitoring agricultural products quality and safety. *International Journal of Agricultural and Biological Engineering*. 11. 35-45
38. Ramírez-Faz, J. & Fernández-Ahumada, Luis & Fernández-Ahumada, Elvira & López-Luque, Rafael (2020). Monitoring of Temperature in Retail Refrigerated Cabinets Applying IoT Over Open-Source Hardware and Software. *Sensors*. 20. 846
39. Rice, D. (2014), "Killer cold: Winter is deadlier than Summer in US", USA Today, July 30, available at:
www.usatoday.com/story/weather/2014/07/30/weather-death-statistics-cold-heat/13323173/
40. Ruan, Junhu & Shi, Yan (2016). Monitoring and assessing fruit freshness in IOT-based e-commerce delivery using scenario analysis and interval number approaches. *Information Sciences*. 373
41. Rusnack, Michael (2021). COVID Vaccine Transport, Storage, and Distribution: Cold Chain Management to Ensure Efficacy. *INNOVATIONS in pharmacy*. 12. 5
42. Salunkhe, P. & Nerkar, Rasika (2016). IoT driven smart system for best cold chain application. 64-67
43. Sergi, Ilaria & Montanaro, Teodoro & Benvenuto, Fabrizio & Patrono, Luigi (2021). A Smart and Secure Logistics System Based on IoT and Cloud Technologies. *Sensors*, 21, 2231
44. Shang, Xiaopu & Zhang, Runtong & Zhu, Xiaomin & Zhou, Quan (2015). Design theory, modelling and the application for the Internet of Things service. *Enterprise Information Systems*. 10. 1-19
45. Shih, Chih-Wen & Wang, Chih-Hsuan (2015). Integrating wireless sensor networks with statistical quality control to develop a cold chain system in food industries. *Computer Standards & Interfaces*. 45
46. Stach, Christoph & Gritti, Clémentine & Przytarski, Dennis & Mitschang, Bernhard (2020). Trustworthy, Secure, and Privacy-aware Food Monitoring Enabled by Blockchains and the IoT.

47. Sun, Lu & Zhao, Yuanjun & Sun, Wenqi & Liu, Zhengkai (2020). Study on supply chain strategy based on cost income model and multi-access edge computing under the background of the Internet of Things. *Neural Computing and Applications*. 32
48. Sunny, Justin & Undralla, Naveen & Pillai, Madhusudanan (2020). Supply chain transparency through blockchain-based traceability: An overview with demonstration. *Computers & Industrial Engineering*. 150
49. Tsang, Y. P. & Choy, K.L. & Wu, Chun-Ho & Ho, G.T.s & Lam, H. & Koo, P.S. (2018). An Internet of Things (IoT)-based risk monitoring system for managing cold supply chain risks. *Industrial Management & Data Systems*. 118
50. Tsang, Y. P. & Choy, K.L. & Wu, Chun-Ho & Ho, G.T.s & Lam, H. & Tang, Valerie (2018). An intelligent model for assuring food quality in managing a multi-temperature food distribution centre. *Food Control*. 90
51. Tsang, Y. P. & Choy, KL & Wu, Chun-Ho & Ho, G.T.s & Lam, H. & Koo, PS. (2017). An IoT-based cargo monitoring system for enhancing operational effectiveness under a cold chain environment. *International Journal of Engineering Business Management*. 9. 184797901774906
52. Urbano, Oscar & Perles, Angel & Pedraza, Cesar & Rubio-Arreaez, Susana & Castelló, M. & Ortola, María & Mercado, Ricardo (2020). Cost-Effective Implementation of a Temperature Traceability System Based on Smart RFID Tags and IoT Services. *Sensors*. 20. 1163
53. Wein, H. (2014), "Fight off food poisoning food safety for warmer weather", *News in Health*, July 1, pp. 1-4, available at:
<https://newsinhealth.nih.gov/issue/jul2014/feature2>
54. Xiao, Pengwen (2020). Real-time Tracking System for Freshness of Cold Chain Logistics based on IoT and GPS Platforms. 834-837
55. Zhang, Ning & Liu, Yingjie (2019). NB-IOT Drives Intelligent Cold Chain for Best Application. 1-4
56. Zhang, Yongjun & Liu, Yanfeng & Jiong, Zhang & Xiaoshuan, Zhang & Li, Baotian & Chen, Enxiu (2021). Development and assessment of blockchain-IoT-based traceability system for frozen aquatic product. *Journal of Food Process Engineering*. 44