

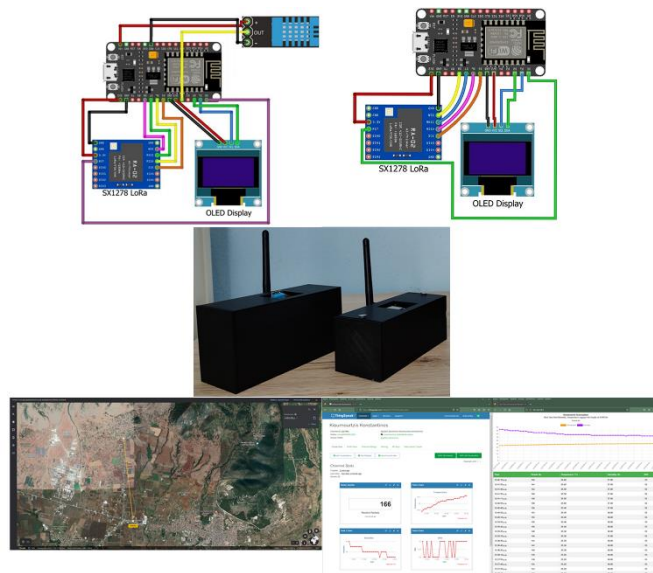
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΕΦΑΡΜΟΣΜΕΝΑ ΗΛΕΚΤΡΟΝΙΚΑ ΣΥΣΤΗΜΑΤΑ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

*«ΚΑΤΑΣΚΕΥΗ ΦΟΡΗΤΩΝ ΜΟΝΑΔΩΝ ΜΕΤΡΗΣΗΣ & ΜΕΤΑΔΟΣΗ
ΔΕΛΟΜΕΝΩΝ ΜΕ ΧΡΗΣΗ LoRa ΕΠΙΚΟΙΝΩΝΙΑΣ»*



Του φοιτητή:
Κιουμουρτζή Κωνσταντίνου
Αρ. Μητρώου: 52011M

Επιβλέπων:
Τσιακμάκης Κυριάκος

Ημερομηνία: 31/1/2023

Τίτλος Δ.Ε.: Κατασκευή φορητών μονάδων μέτρησης και μετάδοση δεδομένων με χρήση LoRa
επικοινωνίας

Κωδικός Δ.Ε.: 22329

Όνοματεπώνυμο φοιτητή: Κιουμουρτζής Κωνσταντίνος

Όνοματεπώνυμο εισηγητή: Τσιακμάκης Κυριάκος

Ημερομηνία ανάληψης Δ.Ε.: 31/10/2022

Ημερομηνία περάτωσης Δ.Ε.: 31/1/2023

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

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

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

«Στην οικογένεια μου»

Πρόλογος

Η εκπόνηση της παρούσας διπλωματικής εργασίας πραγματοποιήθηκε κατά την διάρκεια των μεταπτυχιακών σπουδών μου, στο τμήμα «Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων» του Διεθνούς Πανεπιστημίου της Ελλάδος, με σκοπό την απόκτηση του μεταπτυχιακού διπλώματος.

Η εργασία εστιάζει στην υλοποίηση ενός συστήματος φορητών μονάδων μέτρησης και μετάδοσης δεδομένων με την χρήση LoRa επικοινωνίας. Πιο συγκεκριμένα, ο στόχος της εργασίας είναι η πραγματοποίηση ενός συστήματος Internet of Things το οποίο αποτελείται από την συσκευή μέτρησης και μετάδοσης δεδομένων και από την συσκευή λήψης και καταχώρησης δεδομένων σε cloud υπηρεσία για αποστάσεις από δύο έως τρία χιλιόμετρα. Ως εκ τούτου, η εργασία διαπραγματεύεται τόσο σε θεωρητικό όσο και σε πρακτικό επίπεδο στο διαδίκτυο των πραγμάτων (IoT) και στα δίκτυα μεγάλης απόστασης (LoRa). Στην συνέχεια παρουσιάζει την υλοποίηση του συστήματος όπως επίσης και τις μετρήσεις που πραγματοποιήθηκαν υπό πραγματικές συνθήκες σε αποστάσεις από δύο έως τριών χιλιομέτρων.

Τέλος, καταλήγει στα συμπεράσματα και στην επίλυση των προβλημάτων που προέκυψαν κατά την εκπόνηση της, καθώς επίσης και στις εφαρμογές που μπορούν να έχουν τέτοιου είδους συστήματα επιτυγχάνοντας την κάλυψη υπηρεσιών ίντερνετ σε περιοχές που δεν διαθέτουν υποδομές διαδικτύου.

Περίληψη

Η παρούσα διπλωματική εργασία, αναφέρεται τόσο στην θεωρητική όσο και στην κατασκευαστική υλοποίηση ενός συστήματος φορητών μονάδων μέτρησης, μετάδοσης και καταχώρησης δεδομένων σε cloud με την χρήση LoRa τεχνολογίας. Ως εκ τούτου, η εκπόνηση της εργασίας περιλαμβάνει την υλοποίηση ενός IoT συστήματος.

Η στοίβα πρωτοκόλλων της LoRa τεχνολογίας περιλαμβάνει 4 επίπεδα. Στην Ευρώπη τα δίκτυα LoRa χρησιμοποιούν τις I.S.M. ζώνες συχνοτήτων, ανάλογα με την ζώνη συχνοτήτων που λειτουργεί ένα δίκτυο LoRa επιβάλλονται ορισμένοι περιορισμοί. Η τεχνική διαμόρφωσης που χρησιμοποιείται βασίζεται στην τεχνική C.S.S., τα μεγέθη που καθορίζουν την απόσταση και την ποιότητα μιας ζεύξης είναι ο παράγοντας διασποράς, το εύρος ζώνης συχνοτήτων και οι τεχνικές κωδικοποίησης.

Το σύστημα που υλοποιήθηκε αποτελείται από δύο συσκευές, την συσκευή μέτρησης και εκπομπής δεδομένων και την συσκευή λήψης και καταχώρησης των δεδομένων σε τοπικό server και στο ThingSpeak. Η συσκευή μέτρησης και εκπομπής δεδομένων αποτελείται από έναν μικροεπεξεργαστή ESP8266, έναν αισθητήρα θερμοκρασίας και υγρασίας DHT11, μία OLED οθόνη και έναν LoRa πομπό – δέκτη. Η συσκευή λήψης και καταχώρησης δεδομένων περιλαμβάνει έναν μικροεπεξεργαστή ESP8266, μία OLED οθόνη και έναν LoRa πομπό – δέκτη. Επίσης η συσκευή λήψης και καταχώρησης δεδομένων, λειτουργεί ως τοπικός server καταχωρώντας ταυτόχρονα τα δεδομένα λήψης σε web σελίδα και στο ThingSpeak. Η μέγιστη απόσταση που πραγματοποιήθηκε ζεύξη μεταξύ των δύο συσκευών υπό πραγματικές συνθήκες για τις ανάγκες της εργασίας ήταν της τάξης των 3450 μέτρων.

Εν κατακλείδι, τα συμπεράσματα που προκύπτουν είναι πως προκειμένου να υπάρξει ζεύξη μεταξύ των συσκευών είναι απαραίτητη η ύπαρξη οπτικής επαφής, καθώς τα διάφορα φυσικά εμπόδια και οι κτηριακές εγκαταστάσεις δυσχεραίνουν την επικοινωνία των συσκευών. Το γεγονός αυτό καθιστά τις συσκευές ιδανικές για εφαρμογή σε αγροτικό περιβάλλον. Συνεπώς, επιτυγχάνεται ο σκοπός της εργασίας που είναι η μετάδοση δεδομένων μέτρησης σε αποστάσεις από δύο έως τριών χιλιομέτρων με την χρήση LoRa επικοινωνίας.

«Construction of portable measurement units with data transmission using LoRa communication»

«Kioumourtzis Konstantinos»

Abstract

This dissertation, refers to both the theoretical and the manufacturing implementation of a system of portable measuring units, transmission and recording of data in the cloud using LoRa technology. Therefore, the elaboration of the work includes the implementation of an IoT system.

The protocol stack of LoRa technology includes 4 layers. In Europe LoRa networks use I.S.M. frequency bands, depending on the frequency band in which a LoRa network operates certain limitations are imposed. The shaping technique used is based on the C.S.S. technique, the quantities that determine the distance and the quality of a link are the spread factor, the frequency bandwidth and the coding techniques.

The implemented system consists of two devices, the measuring and data transmitting device and the device for receiving and recording the data on a local server and in ThingSpeak. The measuring and data transmitting device consists of an ESP8266 microprocessor, a DHT11 temperature and humidity sensor, an OLED screen and a LoRa transmitter – receiver. The data receiving and logging device includes an ESP8266 microprocessor, an OLED screen and a LoRa transmitter – receiver as well. Also, the data acquisition and recording device, works as a local server simultaneously registering the receiving data on a web page and on ThingSpeak. The maximum distance was link between the two devices under actual conditions for the needs of the thesis was of the order of 3450 meters.

In conclusion, it follows that for the existence of a link between the devices, visual contact is necessary, as various physical obstacles and building facilities make it difficult for the devices to communicate. This fact makes the devices ideal for application in a rural environment. Therefore, the purpose of the thesis is achieved, which is the transmission of measurement data at distances of two to three kilometers using LoRa communication.

Περιεχόμενα

Κεφάλαιο 1: Διαδίκτυο των πραγμάτων (I.o.T.)	1
1.1 Εισαγωγή	1
1.2 Ορισμός του διαδικτύου των πραγμάτων (I.o.T.).....	1
1.3 Ιστορική αναδρομή του διαδικτύου των πραγμάτων (I.o.T.).....	1
1.4 Τεχνολογίες του διαδικτύου των πραγμάτων (I.o.T.).....	2
1.4.1 Τεχνολογία Radio Frequency Identification – R.F.I.D.	2
1.4.2 Τεχνολογία Wireless Sensor Networks – W.S.N.s	3
1.4.3 Τεχνολογία Low Power Wide Area Networks – L.P.W.A.N.s.....	3
1.4.3.1 Τεχνολογία Sigfox	4
1.4.3.2 Τεχνολογία Weightless.....	5
1.4.3.3 Τεχνολογία Narrow Band – I.o.T.....	6
1.4.3.4 Τεχνολογία LoRa	7
1.5 Εφαρμογές του διαδικτύου των πραγμάτων (I.o.T.).....	7
1.5.1 I.o.T. & μεταφορές (Logistics)	8
1.5.2 I.o.T. & αυτοκινητοβιομηχανία	8
1.5.3 I.o.T. & έξυπνες πόλεις	8
1.5.4 I.o.T. & ηλεκτρονική υγεία	9
1.6 Διαδικτυακά μοντέλα επικοινωνιών του διαδικτύου των πραγμάτων (I.o.T.).....	9
1.6.1 Μοντέλο επικοινωνίας συσκευής σε συσκευή (Device to Device).....	9
1.6.2 Μοντέλο επικοινωνίας συσκευής σε υπολογιστικό νέφος (Device to Cloud).....	10
1.6.3 Μοντέλο επικοινωνίας συσκευής σε πύλη (Device to Gateway).....	11
1.7 Επίλογος.....	12
Κεφάλαιο 2: Δίκτυα μεγάλης εμβέλειας (LoRa).....	13
2.1 Εισαγωγή	13
2.2 Στοιβά LoRa πρωτοκόλλων	13
2.3 Ραδιοσυχρότητες δικτύου LoRa.....	13
2.3.1 Ευρωπαϊκοί περιορισμοί των δικτύων LoRa.....	14
2.4 Φυσικό επίπεδο LoRa – LoRa PHY Layer	15
2.4.1 Υλοποίηση των chirp σημάτων στην τεχνική διαμόρφωσης LoRa	15
2.4.2 Παράγοντας διασποράς στην τεχνική διαμόρφωσης LoRa (Spreading Factor)	17
2.4.3 Υπολογισμός ευαισθησίας και Link Budget.....	18
2.4.4 Τεχνικές κωδικοποίησης στην διαμόρφωση LoRa	19
2.4.4.1 Data Whitening	19

2.4.4.2	Gray Indexing	19
2.4.4.3	Forward Error Correction	19
2.4.4.4	Interleaving	20
2.4.5	Μορφή LoRa πακέτων	20
2.5	Δίκτυα LoRa W.A.N.	22
2.5.1	Τοπολογία των δικτύων LoRa W.A.N.	22
2.5.1.1	Τερματικοί κόμβοι – End Nodes	23
2.5.1.2	Κεντρικοί κόμβοι – Gateways	23
2.5.1.3	Κεντρικοί διακομιστές δικτύου – Network Servers	24
2.5.2	Πρωτόκολλο δικτύου LoRa W.A.N.	24
2.6	Κλάσεις δικτύου LoRa W.A.N.	25
2.6.1	Κλάση A (Class A)	25
2.6.2	Κλάση B (Class B)	25
2.6.3	Κλάση C (Class C)	26
2.7	Δομή πακέτου και MAC εντολές δικτύου LoRa W.A.N.	27
2.8	Επίλογος	29
Κεφάλαιο 3:	Δομή και υλοποίηση συστήματος	30
3.1	Εισαγωγή	30
3.2	Δομή συστήματος	30
3.2.1	Συσκευή μετρήσεων και εκπομπής δεδομένων	30
3.2.2	Συσκευή λήψης και καταχώρησης δεδομένων στο cloud	31
3.3	Κατασκευή συσκευής μέτρησης και εκπομπής δεδομένων	31
3.3.1	Συνδεσμολογία συστήματος μέτρησης και εκπομπής δεδομένων	31
3.3.2	Πειραματική υλοποίηση συστήματος μέτρησης και εκπομπής δεδομένων	33
3.3.3	Κατασκευαστική υλοποίηση συστήματος μέτρησης και εκπομπής δεδομένων	33
3.3.4	Σχεδίαση και υλοποίηση 3D πλαισίου	35
3.4	Κατασκευή συσκευής λήψης και καταχώρησης δεδομένων	38
3.4.1	Συνδεσμολογία συστήματος λήψης και καταχώρησης δεδομένων	38
3.4.2	Πειραματική υλοποίηση συστήματος λήψης και καταχώρησης δεδομένων	39
3.4.3	Κατασκευαστική υλοποίηση συστήματος λήψης και καταχώρησης δεδομένων	39
3.4.4	Σχεδίαση και υλοποίηση 3D πλαισίου	41
3.5	Λογισμικό συσκευών	42
3.5.1	Διάγραμμα ροής προγράμματος συσκευής μέτρησης και εκπομπής δεδομένων	42
3.5.2	Πρόγραμμα συσκευής μέτρησης και εκπομπής δεδομένων	43
3.5.3	Διάγραμμα ροής προγράμματος συσκευής λήψης και καταχώρησης δεδομένων	46

3.5.4	Πρόγραμμα συσκευής λήψης και καταχώρησης δεδομένων.....	47
3.6	Δημιουργία τοπικού server	52
3.7	Εύρος ζεύξης συσκευών	57
3.7.1	Ζεύξη 1 γήπεδο.....	58
3.7.2	Ζεύξη 2 φωτοβολταϊκό πάρκο.....	58
3.7.3	Ζεύξη 3 κατάσταση χρωμάτων.....	59
3.7.4	Ζεύξη 4 μηχανουργείο	60
3.7.5	Ζεύξη 5 εξωκλήσι.....	61
3.8	Καταχώρηση δεδομένων στον τοπικό Server και στο ThingSpeak.....	61
3.8.1	Καταχώρηση δεδομένων ζεύξης 1	62
3.8.2	Καταχώρηση δεδομένων ζεύξης 2.....	63
3.8.3	Καταχώρηση δεδομένων ζεύξης 3	64
3.8.4	Καταχώρηση δεδομένων ζεύξης 4.....	64
3.8.5	Καταχώρηση δεδομένων ζεύξης 5.....	65
3.9	Επίλογος.....	66
Κεφάλαιο 4:	Συμπεράσματα	67
4.1	Εισαγωγή	67
4.2	Συμπεράσματα	67
4.3	Επίλυση προβλημάτων	68
4.4	Εφαρμογές συστήματος.....	68
Βιβλιογραφία.....		69
ΠΑΡΑΡΤΗΜΑ Α: ΤΕΧΝΙΚΑ ΧΑΡΑΚΤΗΡΙΣΤΗΚΑ ΜΟΝΑΔΩΝ		70
ΠΑΡΑΡΤΗΜΑ Β: ΚΩΔΙΚΑΣ ΣΥΣΚΕΥΗΣ ΜΕΤΡΗΣΗΣ ΚΑΙ ΕΚΠΟΜΠΗΣ ΔΕΔΟΜΕΝΩΝ		73
ΠΑΡΑΡΤΗΜΑ Γ: ΚΩΔΙΚΑΣ ΣΥΣΚΕΥΗΣ ΛΗΨΗΣ ΚΑΙ ΚΑΤΑΧΩΡΗΣΗΣ ΔΕΔΟΜΕΝΩΝ		75
ΠΑΡΑΡΤΗΜΑ Δ: ΚΩΔΙΚΑΣ WEB ΣΕΛΙΔΑΣ ΤΟΠΙΚΟΥ SERVER		79

Κατάλογος Σχημάτων

Εικόνα 1.1: Kevin Ashton.....	1
Εικόνα 1.2: Σύνολο συνδεδεμένων συσκευών στο I.o.T. παγκοσμίως απο το 2015 ως το 2025 (σε δισεκατομμύρια).....	2
Εικόνα 1.3: Σύγκριση L.P.W.A.N. τεχνολογιών με υπάρχουσες.....	4
Εικόνα 1.4: Διαμόρφωση B.P.S.K.....	5
Εικόνα 1.5: Τρόποι λειτουργίας N.B. – I.o.T.....	6
Εικόνα 1.6: Σήματα chirp up & chirp down.....	7
Εικόνα 1.7: Μοντέλο επικοινωνίας device to device.....	9
Εικόνα 1.8: Μοντέλο επικοινωνίας device to cloud.....	10
Εικόνα 1.9: Μοντέλο επικοινωνίας device to gateway.....	11
Εικόνα 2.1: Δομή στοίβας πρωτοκόλλων LoRa δικτύου.....	13
Εικόνα 2.2: Παγκόσμιος χάρτης συχνοτήτων LoRa δικτύων.....	14
Εικόνα 2.3: Λογικό «0» και λογικό «1» chirp σήματος.....	16
Εικόνα 2.4: Διαδικασία κωδικοποίησης LoRa σήματος.....	16
Εικόνα 2.5: Κωδικοποιημένο σήμα LoRa.....	17
Εικόνα 2.6: Σύγκριση παραγόντων διασποράς από S.F. = 7 ως S.F. = 12.....	17
Εικόνα 2.7: Δομή πακέτου LoRa.....	20
Εικόνα 2.8: Πακέτο LoRa.....	21
Εικόνα 2.9: Απεικόνιση de – chirped σήματος.....	22
Εικόνα 2.10: Τοπολογία Αστέρα LoRa W.A.N.....	23
Εικόνα 2.11: LoRa Ra – 02 E.N.....	23
Εικόνα 2.12: Πρωτόκολλο δικτύου LoRa W.A.N.....	24
Εικόνα 2.13: Κλάση A αποστολή και λήψη δεδομένων.....	25
Εικόνα 2.14: Κλάση B αποστολή και λήψη δεδομένων.....	26
Εικόνα 2.15: Κλάση C αποστολή και λήψη δεδομένων.....	26
Εικόνα 2.16: Συνοπτική παρουσίαση λήψης και αποστολής δεδομένων των κλάσεων.....	26
Εικόνα 2.17: Συγκριτικό διάγραμμα κλάσεων.....	27
Εικόνα 2.18: Τυπική μορφή πακέτου LoRa.....	28
Εικόνα 3.1: Μπλοκ διάγραμμα συσκευής μετρήσεων και εκπομπής δεδομένων.....	30
Εικόνα 3.2: Μπλοκ διάγραμμα συσκευής λήψης και καταχώρησης δεδομένων σε cloud.....	31
Εικόνα 3.3: Συνδεσμολογία συσκευής μέτρησης και εκπομπής δεδομένων.....	32
Εικόνα 3.4: Συσκευή μέτρησης και εκπομπής δεδομένων σε πειραματικό στάδιο.....	33
Εικόνα 3.5: Πάνω όψη διάτρητης πλακέτας.....	33
Εικόνα 3.6: Κάτω όψη διάτρητης πλακέτας.....	34
Εικόνα 3.7: Συνδεσμολογία των pins header (πάνω όψη).....	34
Εικόνα 3.8: Τοποθέτηση ενδεικτικού led.....	34
Εικόνα 3.9: Τελική μορφή κατασκευής συστήματος μέτρησης και εκπομπής δεδομένων.....	35
Εικόνα 3.10: Τελική μορφή κατασκευής dc - dc buck converter.....	35
Εικόνα 3.11: Τελική μορφή μοντέλου 3D.....	36
Εικόνα 3.12: Μετατροπή αρχείου .stl σε g - code.....	36
Εικόνα 3.13: Διαδικασία 3D εκτύπωσης πλαισίου συσκευής μέτρησης και εκπομπής δεδομένων.....	37
Εικόνα 3.14: Τελικό αποτέλεσμα 3D εκτύπωσης φορητής συσκευής μέτρησης και εκπομπής δεδομένων.....	37
Εικόνα 3.15: Συνδεσμολογία συσκευής λήψης και καταχώρησης δεδομένων.....	38

Εικόνα 3.16: Συσκευή λήψης και καταχώρησης δεδομένων σε πειραματικό στάδιο.....	39
Εικόνα 3.17: Πάνω όψη διάτρητης πλακέτας.....	40
Εικόνα 3.18: Κάτω όψη διάτρητης πλακέτας.....	40
Εικόνα 3.19: Συνδεσμολογία των pin headers (πάνω όψη).....	40
Εικόνα 3.20: Συνδεσμολογία των pin headers (κάτω όψη).....	40
Εικόνα 3.21: Τελική μορφή συστήματος λήψης και καταχώρησης δεδομένων.....	41
Εικόνα 3.22: Τελικό αποτέλεσμα 3D εκτύπωσης φορητής συσκευής λήψης και καταχώρησης δεδομένων.....	41
Εικόνα 3.23: Διάγραμμα ροής προγράμματος συσκευής μέτρησης και εκπομπής δεδομένων.....	42
Εικόνα 3.24: Περίληψη βιβλιοθηκών και αρχικοποιήσεις μεταβλητών.....	43
Εικόνα 3.25: Υπό – ρουτίνες αρχικοποίησης, ανάγνωσης και εμφάνισης δεδομένων.....	44
Εικόνα 3.26: Υπό - ρουτίνα αποστολής δεδομένων, ρουτίνα αρχικοποίησης, ρουτίνα λειτουργίας....	45
Εικόνα 3.27: Διάγραμμα ροής προγράμματος συσκευής λήψης και καταχώρησης δεδομένων.....	46
Εικόνα 3.28: Περίληψη βιβλιοθηκών και αρχικοποιήσεις μεταβλητών.....	47
Εικόνα 3.29: Υπό – ρουτίνες μετατροπής δεδομένων λήψης, δημιουργίας τοπικού server, καταχώρησης δεδομένων στον τοπικό server και αρχικοποίησης LoRa πομπό – δέκτη.....	48
Εικόνα 3.30:Υπό - ρουτίνες αρχικοποίησης και σύνδεσης Wi - Fi, αρχικοποίησης και εκκίνησης τοπικού server.....	49
Εικόνα 3.31: Υπό – ρουτίνες ανάλυσης δεδομένων λήψης, σύνθεσης και καταχώρησης δεδομένων στο ThingSpeak.....	50
Εικόνα 3.32: Υπό – ρουτίνα αρχικοποίησης OLED οθόνης, ρουτίνα αρχικοποίησης και ρουτίνα λειτουργίας.....	51
Εικόνα 3.33: Web σελίδα τοπικού server.....	52
Εικόνα 3.34: Διαμόρφωση κεφαλίδας, γραφικής παράστασης και πίνακα δεδομένων web σελίδας...	53
Εικόνα 3.35: Διαμόρφωση του σώματος της web σελίδας.....	54
Εικόνα 3.36: Εμφάνιση γραφήματος και αυτόματη ανανέωση web σελίδας.....	55
Εικόνα 3.37: Διαχωρισμός και καταχώρηση δεδομένων σε μορφή γραφήματος και στον πίνακα.....	56
Εικόνα 3.38: Συντεταγμένες 41°11'32''N 24°6'14''E.....	57
Εικόνα 3.39: Ζεύξη 1 γήπεδο.....	58
Εικόνα 3.40: Ζεύξη 2 φωτοβολταϊκό πάρκο.....	59
Εικόνα 3.41: Ζεύξη 3 κατάσταση χρωμάτων.....	59
Εικόνα 3.42: Ζεύξη 4 μηχανουργείο.....	60
Εικόνα 3.43: Ζεύξη 5 εξωκλήσι.....	61
Εικόνα 3.44: Δομή καναλιού στο thingspeak και στον τοπικό server.....	62
Εικόνα 3.45: Συσκευή μέτρησης στην περιοχή του γηπέδου σε απόσταση 2080 m.....	62
Εικόνα 3.46: Καταχώρηση δεδομένων ζεύξης 1.....	63
Εικόνα 3.47: Συσκευή μέτρησης στην περιοχή του φωτοβολταϊκού πάρκου σε απόσταση 2530 m ...	63
Εικόνα 3.48: Καταχώρηση δεδομένων ζεύξης 2.....	63
Εικόνα 3.49: Συσκευή μέτρησης στην περιοχή του καταστήματος χρωμάτων σε απόσταση 3450 m.	64
Εικόνα 3.50: Καταχώρηση δεδομένων ζεύξης 3.....	64
Εικόνα 3.51: Συσκευή μέτρησης στην περιοχή του μηχανουργείου σε απόσταση 2940 m.....	64
Εικόνα 3.52: Καταχώρηση δεδομένων ζεύξης 4.....	65
Εικόνα 3.53: Συσκευή μέτρησης στην περιοχή του εξωκλήσιου σε απόσταση 2550 m.....	65
Εικόνα 3.54: Καταχώρηση δεδομένων ζεύξης 5.....	65

Κατάλογος Πινάκων

Πίνακας 2.1: Κανονισμοί LoRa καναλιών όπως ορίστηκαν απο τον E.T.S.I.	15
Πίνακας 2.2: Παράμετροι που επηρεάζονται από τον S.F.	18
Πίνακας 3.1: Πίνακας συνδεσμολογίας συσκευής μέτρησης και εκπομπής	32
Πίνακας 3.2: Πίνακας συνδεσμολογίας συσκευής λήψης και καταχώρησης.....	38
Πίνακας 3.3: Σημεία μετρήσεων	58
Πίνακας 3.4: Ρυθμίσεις ζεύξης 1	58
Πίνακας 3.5: Ρυθμίσεις ζεύξης 2	59
Πίνακας 3.6: Ρυθμίσεις ζεύξης 3	60
Πίνακας 3.7: Ρυθμίσεις ζεύξης 4	60
Πίνακας 3.8: Ρυθμίσεις ζεύξης 5	61

Συντομογραφίες

Δ.Ε.	Διπλωματική Εργασία
ΔΙ.ΠΑ.Ε.	Διεθνές Πανεπιστήμιο Ελλάδος
I.o.T.	Internet of Things
R.F.ID.	Radio Frequency Identification
W.S.N.	Wireless Sensor Networks
L.P.W.A.N.	Low Power Wide Area Network
I.S.M.	Industrial, Scientific and Medical (Frequency Zones)
B.P.S.K.	Binary Phase Shift Keying (Modulation)
TVW.S.	TV White Space
F.H.	Frequency Hopping
A.E.S.	Advanced Encryption Standard (Encryption)
Q.P.S.K.	Quadrature Phase Shift Keying (Modulation)
G.M.S.K.	Gaussian Minimum Shift Keying (Modulation)
L.T.E.	Long Term Evolution
3G.P.P.	3 rd Generation Partnership Project
C.S.S.	Chirp Spread Spectrum
S.N.R.	Signal to Noise Ratio
I.o.V.	Internet of Vehicles
MEM.S.	Microelectromechanical Systems
I.A.B.	Internet Architecture Board
A.L.G.	Application Layer Gateway
A.P.I.	Application Programming Interface
I.T.U.	International Telecommunication Union
E.T.S.I.	European Telecommunications Standards Institute
E.I.R.P.	Effective Isotropic Radiated Power
E.R.P.	Effective Radiated Power
D.S.S.S.	Direct – sequence spread spectrum

S.F.	Spreading Factor
Q.o.S.	Quality of Service
C.R.C.	Cyclic Redundancy Check
T.o.A.	Time on Air
C.R.	Coding Rate
M.F.S.K.	Multiple Frequency – Shift Keying
E.N.s	End Nodes
S.o.S.	Stars – of – Stars Topology
G.W.s	Gateways

Κεφάλαιο 1: Διαδίκτυο των πραγμάτων (I.o.T.)

1.1 Εισαγωγή

Το κεφάλαιο αυτό έχει ως σκοπό να εισάγει τον αναγνώστη στο διαδίκτυο των πραγμάτων (Internet of Things – I.o.T.). Αρχικά, δίνεται ο ορισμός του I.o.T. και στην συνέχεια γίνεται αναφορά στην ιστορική του αναδρομή. Έπειτα, περιγράφονται οι τεχνολογίες I.o.T. που έχουν αναπτυχθεί και που χρησιμοποιούνται κατά κόρον. Επί προσθέτου, για να καταστεί σαφές η σπουδαιότητα του θέματος τούτου, γίνεται μία σύντομη αναφορά στις εφαρμογές που έχουν τα συστήματα αυτά στην καθημερινότητα μας. Τέλος, περιγράφονται τα υπάρχοντα διαδικτυακά μοντέλα επικοινωνιών που υπάρχουν στο I.o.T..

1.2 Ορισμός του διαδικτύου των πραγμάτων (I.o.T.)

Ο όρος «διαδίκτυο των πραγμάτων» αναφέρεται σε ένα δίκτυο το οποίο αποτελείται από συσκευές οι οποίες μπορούν και επικοινωνούν μεταξύ τους με σκοπό την ανταλλαγή δεδομένων. Αυτού του είδους οι συσκευές μπορεί να είναι είτε ηλεκτρικές, είτε ηλεκτρονικές και παρέχουν την δυνατότητα να συνδεθούν μεταξύ τους μέσω ενός τοπικού δικτύου ή με την χρήση του διαδικτύου. Κάθε μία από αυτές τις συσκευές ονομάζεται «πράγμα», στο διαδίκτυο των πραγμάτων κάθε «πράγμα» συνδέεται σε μία πλατφόρμα όπου εκεί γίνεται η συλλογή όλων των δεδομένων. Εφόσον πραγματοποιηθεί η επεξεργασία των δεδομένων γίνεται η εξυπηρέτηση συγκεκριμένων σκοπών ή αναγκών.[1]

1.3 Ιστορική αναδρομή του διαδικτύου των πραγμάτων (I.o.T.)

Η επινοήση του όρου «Internet of Things (I.o.T.)» έγινε από τον βρετανό Kevin Ashton (εικόνα 1.1) και την ομάδα του στα τέλη της δεκαετίας του 1990. Ο Kevin Ashton ήταν υπεύθυνος στα εργαστήρια Auto – ID του MIT όπου μαζί με την ομάδα του δημιούργησαν τα πρότυπα συστημάτων Radio Frequency Identification – R.F.ID., το R.F.ID. είναι μία από τις πρώτες τεχνολογίες που χρησιμοποιήθηκαν στο I.o.T., κάθε συσκευή που συνδέονταν στο δίκτυο είχε έναν κωδικό R.F.ID., ο κωδικός αυτός λειτουργούσε ως ηλεκτρονική ταυτότητα, μ' αυτόν τον τρόπο γινόταν η αναγνώριση και η καταγραφή τους από τους ηλεκτρονικούς υπολογιστές. Το 2003 η ιδέα του I.o.T. άρχισε να γνωστοποιείται εστιάζοντας το ενδιαφέρον και τις επενδύσεις διάφορων εταιριών, αναγνωρίζοντας έτσι από την πρώτη στιγμή την σημαντικότητα του.[1]

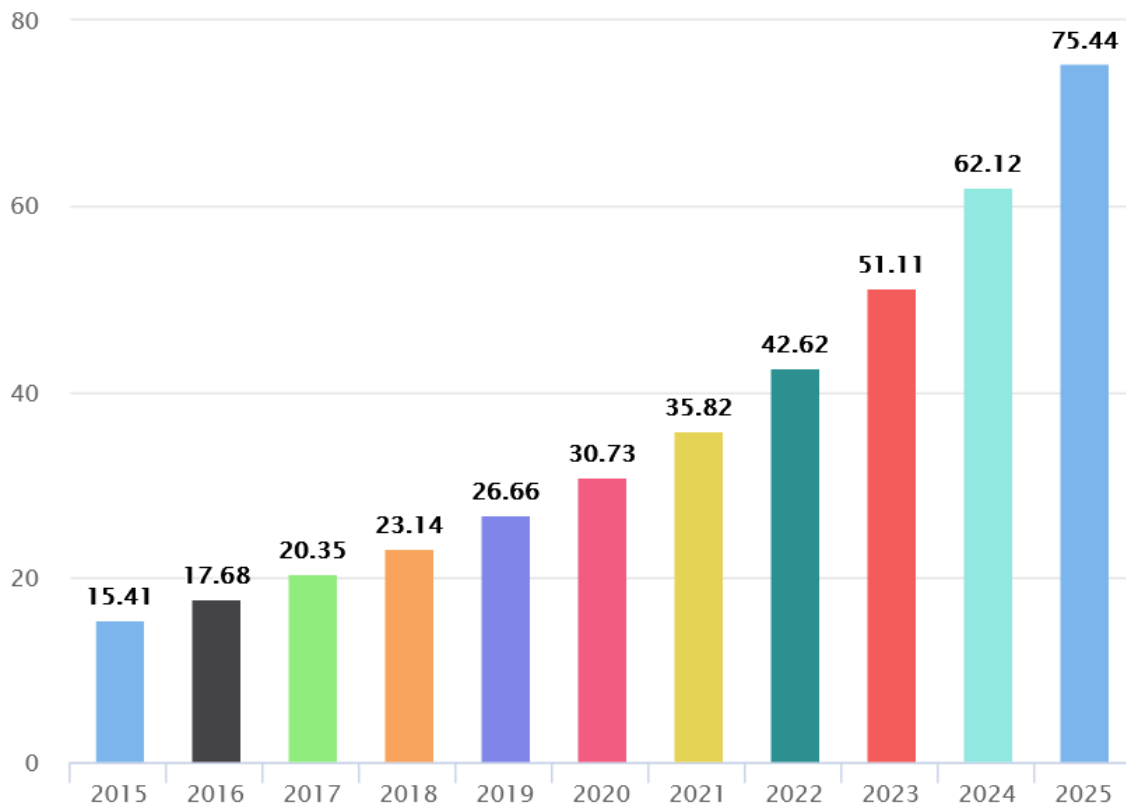


Εικόνα 1.1: Kevin Ashton

Τον Ιανουάριο του 2009 μια ομάδα κινέζων επιστημόνων έπειτα από εξέταση δεδομένων δρομολόγησης στο διαδίκτυο κατέληξαν στο συμπέρασμα ότι το μέγεθος του διαδικτύου διπλασιάζεται κάθε 5,3 χρόνια. Η Cisco IBSG όρισε την απαρχή του διαδικτύου των πραγμάτων μεταξύ του 2008 και του 2009. Το I.o.T. συνεχίζει απρόσκοπτα την εξέλιξή του με ραγδαίους ρυθμούς. Σύμφωνα με δεδομένα που διαθέτει η Cisco εκτιμάτε πως το 2019 περισσότερα από 24 δισεκατομμύρια αντικείμενα ήταν συνδεδεμένα στο internet.[2]

Σήμερα, το I.o.T. προκαλεί το ενδιαφέρον της κοινωνίας, καθώς συνιστά ένα ραγδαία εξελισσόμενο μέσο αλληλεπίδρασης οργανισμών και καταναλωτών – χρηστών, η τεχνολογία του I.o.T. βρίσκει εφαρμογή σε μία μεγάλη γκάμα δραστηριοτήτων, από οικιακές συσκευές και κτίρια μέχρι βιομηχανικές εγκαταστάσεις και αγροκτήματα. Η εικόνα 1.2 δείχνει τον αριθμό των συνδεδεμένων

συσκευών παγκοσμίως στο I.o.T. από το 2015 ως το 2025, σύμφωνα με έρευνα που έγινε τον Νοέμβριο του 2016 και δημοσιεύτηκε από το Statista Research Department στο τέλος του 2022 ο αριθμός το συνδεδεμένων συσκευών στο διαδίκτυο αναμένεται να φτάσει τα 42,62 δισεκατομμύρια.[3]



Εικόνα 1.2: Σύνολο συνδεδεμένων συσκευών στο I.o.T. παγκοσμίως από το 2015 ως το 2025 (σε δισεκατομμύρια)

1.4 Τεχνολογίες του διαδικτύου των πραγμάτων (I.o.T.)

Σ' αυτή την υπό – ενότητα θα αναφερθούν οι τεχνολογίες που έχουν αναπτυχθεί και επιτρέπουν τις διάφορες συσκευές να επικοινωνούν μεταξύ τους. Πρόκειται για την R.F.ID. τεχνολογία, τα ασύρματα δίκτυα των αισθητήρων (Wireless Sensor Networks – W.S.N.), την τεχνολογία των μεγάλων δεδομένων (Big Data), την υπολογιστική νέφους (Cloud Computing) και την υπολογιστική ομίχλης (Fog Computing).[1]

1.4.1 Τεχνολογία Radio Frequency Identification – R.F.I.D.

Πρόκειται για μία αυτόματη τεχνολογία η οποία επιτρέπει τα υπολογιστικά συστήματα να εντοπίζουν αντικείμενα και να καταγράφουν δεδομένα μέσω ραδιοκυμάτων. Ένα τυπικό σύστημα R.F.ID. αποτελείται από ετικέτες που έχουν τον ρόλο του πομπού και αναγνώστες που έχουν τον ρόλο του δέκτη. Κάθε ετικέτα, περιλαμβάνει ένα microchip το οποίο συνδέεται με μία κεραία και αποτελεί το αναγνωριστικό του συγκεκριμένου αντικειμένου. Η επικοινωνία του δέκτη με τον πομπό γίνεται με την χρήση ραδιοκυμάτων. Το βασικό πλεονέκτημα αυτής της τεχνολογίας είναι ότι η αναγνώριση και η συλλογή των δεδομένων επιτυγχάνεται αυτόματα. Για τον λόγο αυτό το R.F.ID. αποτέλεσε σημαντική τεχνολογία του I.o.T. τα πρώτα χρόνια της λειτουργίας του.[1]

1.4.2 Τεχνολογία Wireless Sensor Networks – W.S.N.s

Η ανάπτυξη του I.o.T. έγινε παράλληλα με τα ασύρματα δίκτυα των αισθητήρων (W.S.N.). Τα W.S.N., αποτελούνται από αυτόνομους αισθητήρες οι οποίοι συλλέγουν δεδομένα από το περιβάλλον στο οποίο βρίσκονται με σκοπό την παρακολούθηση φυσικών μεγεθών π.χ. θερμοκρασία, ήχος, πίεση, υγρασία, αγωγιμότητα κτλ., τα δεδομένα αυτά μεταφέρονται μέσω του διαδικτύου σε κατάλληλη τοποθεσία με σκοπό την ανάλυση και την επεξεργασία τους. Σήμερα, υπάρχουν δίκτυα που έχουν την ικανότητα να λαμβάνουν και να στέλνουν πληροφορίες μέσω του απομακρυσμένου ελέγχου.[1]

Ένα W.S.N., περιγράφεται επίσης ως ένα δίκτυο κόμβων που αντιλαμβάνεται και ελέγχει το περιβάλλον, επιτρέποντας μ' αυτόν τον τρόπο την αλληλεπίδραση μεταξύ προσώπων – υπολογιστών και του περιβάλλοντος. Η μεγαλύτερη πρόκληση είναι ο σχεδιασμός των αισθητήρων, οι οποίοι πρέπει να έχουν μικρό μέγεθος και η κατασκευή τους να είναι οικονομική. Επίσης, τα W.S.N., πρέπει να επιτυγχάνουν μεγάλη εξοικονόμηση ενέργειας προκειμένου να εξασφαλίζουν την μακροχρόνια λειτουργία τους. Για τον λόγο αυτό, χρησιμοποιούνται πρωτόκολλα τα οποία είναι αποδοτικά εξοικονομώντας κατ' αυτόν τον τρόπο ενεργειακούς πόρους.[1]

Σε μία εφαρμογή, τα W.S.N. επικοινωνούν μ' ένα τοπικό δίκτυο ή δίκτυο ευρείας περιοχής. Στην 2^η περίπτωση μία πύλη (Gateway), λειτουργεί ως μεσολαβητής για να ενώσει τους αισθητήρες που συνήθως έχουν χαμηλή κατανάλωση ενέργειας με το υπόλοιπο δίκτυο (server), στο οποίο γίνεται η επεξεργασία των δεδομένων. Αυτού του είδους τα δίκτυα ονομάζονται, δίκτυα χαμηλής κατανάλωσης και ευρείας περιοχής (Low Power Wide Area Network – L.P.W.A.N.).[1]

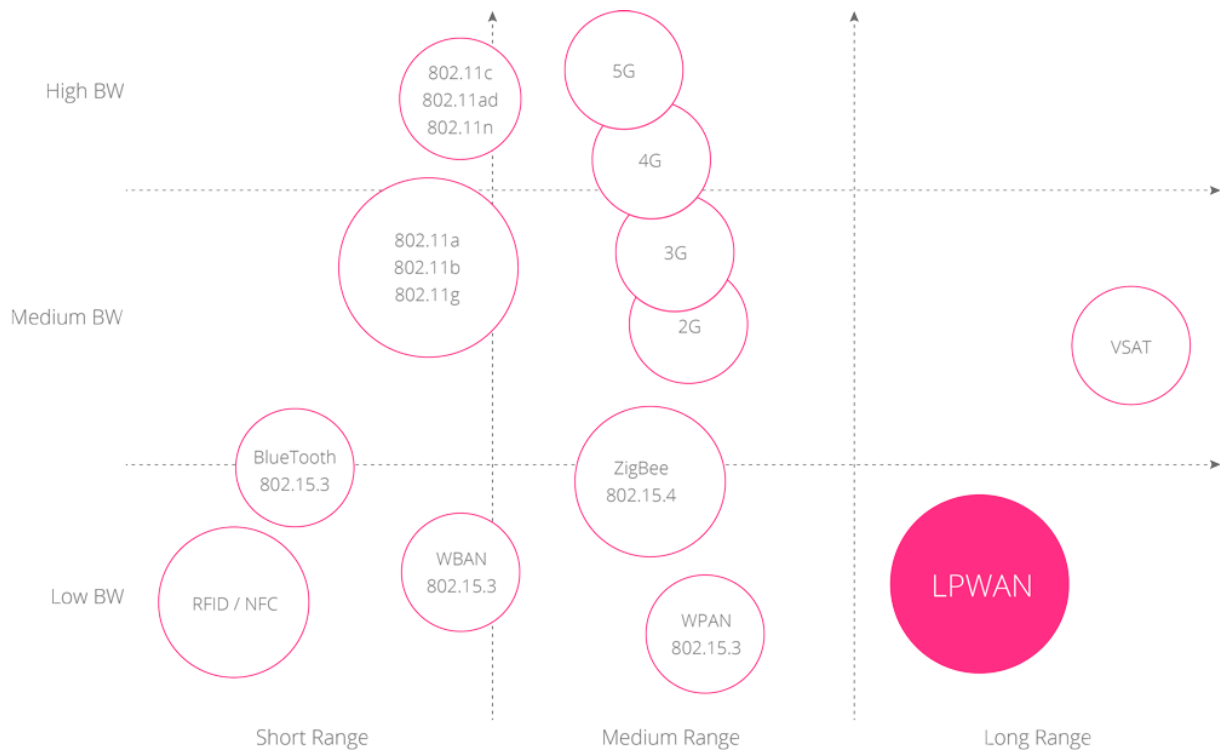
1.4.3 Τεχνολογία Low Power Wide Area Networks – L.P.W.A.N.s

Τα L.P.W.A.N., έχουν σχεδιαστεί για εφαρμογές – αισθητήρες που στέλνουν μικρό όγκο δεδομένων, σε μεγάλες αποστάσεις ανά τακτά χρονικά διαστήματα εστιάζοντας μ' αυτόν τον τρόπο στην χαμηλή κατανάλωση ενέργειας, σε αντίθεση με τα δίκτυα 4G/5G και τα Wi – Fi που έχουν ως σκοπό τους υψηλούς ρυθμούς δεδομένων και την ελαχιστοποίηση της καθυστέρησης.[1]

Η κατηγορία των L.P.W.A.N.s, έχει προσελκύσει το ενδιαφέρον τόσο από την επιστημονική – ερευνητική κοινότητα, όσο και από την βιομηχανία προκειμένου να δοθούν λύσεις για την ικανοποίηση των απαιτήσεων των I.o.T. εφαρμογών. Η πλειοψηφία των τερματικών συσκευών στα I.o.T. δίκτυα λειτουργεί με μπαταρίες καθώς παρέχουν μεγάλη αυτονομία. Το γεγονός αυτό, εξασφαλίζει την κάλυψη περιοχών της τάξης των 10 – 40 km σε αγροτικές εφαρμογές και 1 – 5 km σε αστικές για μεγάλα χρονικά διαστήματα. Η εικόνα 1.3 δείχνει πως οι L.P.W.A.N. τεχνολογίες καλύπτουν μεγαλύτερη έκταση σε σχέση με τις υπάρχουσες, μεταδίδοντας όμως μικρό όγκο δεδομένων. Επιγραμματικά, ορισμένες L.P.W.A.N. τεχνολογίες είναι:[1]

- Η Sigfox
- Η Weightless
- Η Narrow Band – I.o.T.
- Η LoRa/LoRa WAN

Οι παραπάνω τεχνολογίες αναλύονται συνοπτικά στην συνέχεια. Τέλος, οι τεχνολογίες L.P.W.A.N. παρουσιάζουν ορισμένες προκλήσεις που αφορούν τον περιορισμό του φάσματος, την συνύπαρξη με άλλες τεχνολογίες, την κινητικότητα, την επεκτασιμότητα, την κάλυψη, την ασφάλεια και τις απαιτήσεις που παρουσιάζει η εκάστοτε εφαρμογή, όπως οι ταχύτητες μετάδοσης δεδομένων και οι επικοινωνίες σε πραγματικούς χρόνους.[1]



Εικόνα 1.3: Σύγκριση L.P.W.A.N. τεχνολογιών με υπάρχουσες

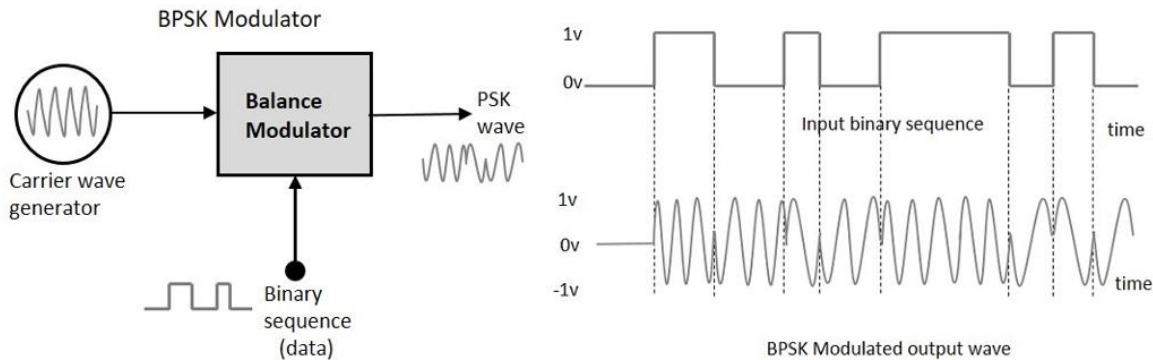
1.4.3.1 Τεχνολογία Sigfox

Η Sigfox στην ουσία είναι μία διαχειρίστρια δικτύου I.o.T. η οποία παρέχει επικοινωνία μεταξύ συσκευών σημείο προς σημείο (point – point). Η επέκτασή της εκτείνεται σε παγκόσμια κλίμακα καθώς υπάρχει συνεργασία με άλλους παρόχους δικτύων. Η λήψη δεδομένων από τις συνδεδεμένες συσκευές στην Sigfox γίνεται μέσω εγκατεστημένων κεραιών. Για την μετάδοση των δεδομένων χρησιμοποιούνται οι ελεύθερες βιομηχανικές, επιστημονικές και ιατρικές μπάντες συχνοτήτων (Industrial, Scientific and Medical – I.S.M.) οι οποίες βρίσκονται στα 868MHz για την Ευρώπη και 915MHz για την Αμερική και διασυνδέουν μέσω ενός IP δικτύου τα σήματα στον server.[1]

Για την μετάδοση του σήματος των συσκευών στους σταθμούς βάσης χρησιμοποιείται η δυαδική διαμόρφωση μετατόπισης φάσης (Binary Phase Shift Keying – B.P.S.K.) σε εξαιρετικά στενές ζώνες των 100Hz. Με την χρησιμοποίηση των στενών ζωνών, η Sigfox καταφέρνει να διατηρεί πολύ χαμηλά επίπεδα θορύβου, εξοικονόμηση ενεργειακών πόρων, υψηλή ευαισθησία στους δέκτες και χαμηλό κόστος κεραιών αξιοποιώντας έτσι όλο το εύρος ζώνης συχνοτήτων, με αντάλλαγμα την μέγιστη απόδοση μετάδοσης η οποία ανέρχεται στα 100 bps.[1]

Ο αριθμός μετάδοσης των μηνυμάτων μέσω ανερχόμενης ζεύξης περιορίζεται στα 140 μηνύματα την ημέρα. Το μέγιστο μήκος του ωφέλιμου φορτίου για κάθε μήνυμα ανερχόμενης ζεύξης είναι 12 byte και παραμένει στον αέρα κατά μ.ό. 2 second. Παρ' όλα αυτά, ο αριθμός των μηνυμάτων στην κατερχόμενη ζεύξη περιορίζεται στα 4 μηνύματα την ημέρα, αυτό σημαίνει ότι δεν υποστηρίζεται η αναγνώριση κάθε μηνύματος ανερχόμενης ζεύξης. Το μέγιστο μήκος του ωφέλιμου φορτίου για κάθε μήνυμα κατερχόμενης ζεύξης είναι 8 byte. Η διασφάλιση της αξιοπιστίας στην επικοινωνία ανερχόμενης ζεύξης, επιτυγχάνεται με την χρησιμοποίηση ποικιλίας χρόνου, συχνότητας και διπλότυπου μετάδοσης. Γι' αυτόν τον λόγο, κάθε μήνυμα τελικής συσκευής μεταδίδεται πολλές φορές (τουλάχιστον τρεις από προεπιλογή), σε διαφορετικά κανάλια συχνότητας.[1]

Σε αντίθεση με τους σταθμούς βάσης οι οποίοι μπορούν να λαμβάνουν ταυτόχρονα μηνύματα σε όλα τους τα κανάλια, μία τερματική συσκευή μπορεί να επιλέξει τυχαία μόνο ένα κανάλι συχνότητας για να μεταδώσει τα μηνύματά του. Αυτό έχει ως αποτέλεσμα την απλοποίηση του σχεδιασμού της εκάστοτε τερματικής συσκευής και την μείωση του κόστους της. Το πρωτόκολλο της Sigfox μειώνει το μέγεθος πλαισίου του σήματος και δεν επιβάλλει τον συγχρονισμό με το δίκτυο. Τέλος, η χαμηλή στάθμη και η μικρή διάρκεια των εκπομπών επιτρέπει στις συσκευές την μέγιστη αυτονομία.[1]



Εικόνα 1.4: Διαμόρφωση B.P.S.K.

1.4.3.2 Τεχνολογία Weightless

Η τεχνολογία weightless, επιτρέπει την ασύρματη επικοινωνία για L.P.W.A.N. δίκτυα και έχει σχεδιαστεί για I.o.T.. Η weightless αποτελεί μέρος της επωνυμίας του οργανισμού, Weightless Special Interest Group (S.I.G.). Ο οργανισμός Weightless S.I.G. προσφέρει τρία διαφορετικά πρωτόκολλα επικοινωνίας τα οποία χρησιμοποιούνται σε διαφορετικές περιπτώσεις και είναι τα εξής:[1]

- Weightless – N
- Weightless – W
- Weightless – P

Το Weightless – W, λειτουργεί στο φάσμα που δεν χρησιμοποιείται και διατίθεται για την τηλεόραση (TV White Space – TVW.S.), σ' αυτό το φάσμα οι κανονισμοί αλλάζουν ανάλογα με την περιοχή λειτουργίας, επομένως το εύρος εκπομπής διαφοροποιείται μεταξύ των 400MHz – 800MHz. Οι τερματικές συσκευές κατασκευάζονται ώστε να λειτουργούν σε ένα μικρό εύρος του φάσματος, όπως γίνεται αντιληπτό, η χρησιμοποίηση του TVW.S. σε I.o.T. εφαρμογές αντιμετωπίζει δυσκολίες καθώς είναι αδύνατο να χρησιμοποιηθεί σε όλες τις περιοχές λόγω των διαφόρων περιορισμών που ισχύουν στην καθεμία.[1]

Το Weightless – N είναι παρόμοιο με το πρωτόκολλο της Sigfox, η διαμόρφωση του σήματος γίνεται με B.P.S.K. και χρησιμοποιεί άλμα συχνοτήτων (Frequency Hopping – F.H.) προκειμένου να μειώσει τις παρεμβολές. Το σύστημα δεν είναι κλειστό και αποτελείται από ένα δίκτυο συνεργατών. Τέλος, επιτρέπει την κινητικότητα και παρέχει προηγμένο πρότυπο κρυπτογράφησης (Advanced Encryption Standard – A.E.S.) 128bit.[1]

Το Weightless – P περιλαμβάνει χαρακτηριστικά αμφίδρομης επικοινωνίας και καλή ποιότητα υπηρεσιών. Η λειτουργία του γίνεται σε κανάλι συχνότητας 12.5kHz, επίσης η διαμόρφωση που χρησιμοποιείται είναι τετραγωνική διαμόρφωση μετατόπισης φάσης (Quadrature Phase Shift Keying – Q.P.S.K.) ή ελάχιστη γκαουσιανή διαμόρφωση μετατόπιση φάσης (Gaussian Minimum Shift Keying – G.M.S.K.). Το Weightless – P είναι ιδανικό για εφαρμογή σε ιδιωτικά δίκτυα, στα οποία είναι αναγκαίος

ο έλεγχος των uplink και downlink δεδομένων, αντίθετα δεν κρίνεται κατάλληλο για την εφαρμογή σε δίκτυα ευρείας περιοχής.[1]

Τέλος, η περιοχή κάλυψης των πρωτοκόλλων Weightless – W και Weightless – N κυμαίνεται περίπου 5km σε αστικό περιβάλλον ενώ το Weightless – P έχει εμβέλεια περίπου 2km.[1]

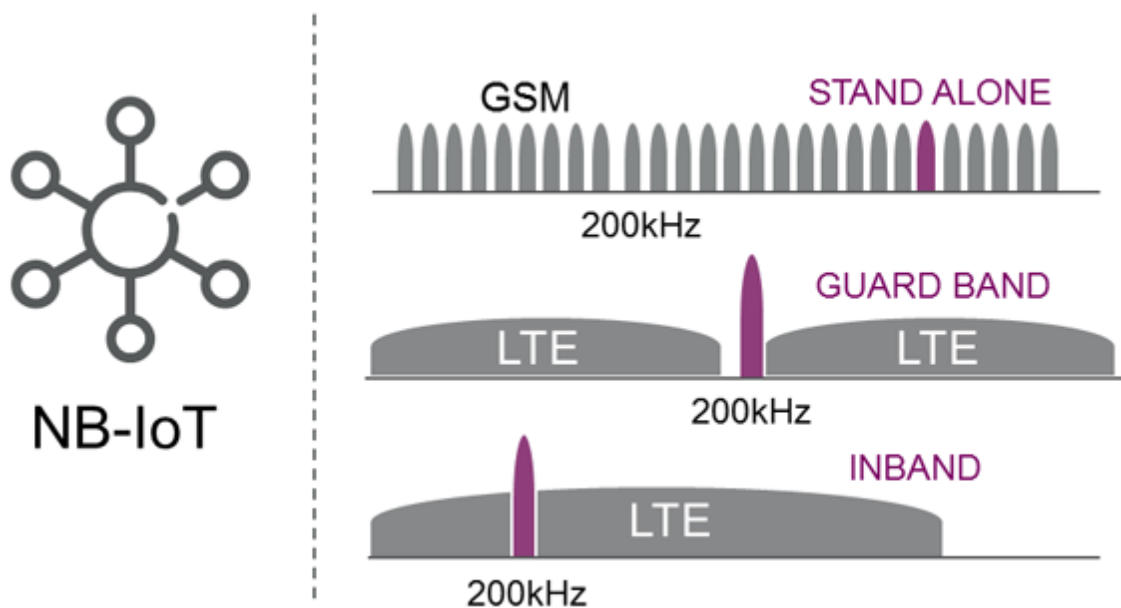
1.4.3.3 Τεχνολογία Narrow Band – I.o.T.

Η τεχνολογία Narrow Band I.o.T. είναι βασισμένη στην ήδη υπάρχουσα τεχνολογία Long Term Evolution – L.T.E. η οποία βρίσκει εφαρμογή στην ασύρματη δικτύωση και την επικοινωνία κινητών συσκευών με υψηλές ταχύτητες και χρησιμοποιεί τις ίδιες μη ελεύθερες ζώνες συχνοτήτων. Η ανάπτυξη της προκύπτει από έναν όμιλο συνεργασίας μεταξύ τηλεπικοινωνιακών οργανισμών ανά τον κόσμο, το οποίο ονομάζεται 3rd Generation Partnership Project – 3G.P.P., ο οποίος είναι υπεύθυνος για την εισαγωγή του N.B. – I.o.T. στην αγορά.[1]

Το N.B. – I.o.T. επιτρέπει την μετάδοση δεδομένων με χαμηλό ρυθμό, την μεγάλη αυτονομία, το χαμηλό κόστος των συσκευών και την επέκταση της εμβέλειας κάλυψης, δίχως να επηρεάζονται οι εφαρμογές από την καθυστέρηση δεδομένων. Το N.B. – I.o.T. χρειάζεται ένα ελάχιστο εύρος ζώνης 180kHz και αναπτύσσεται με 3 διαφορετικούς τρόπους λειτουργίας που είναι:[1]

- Σαν αυτόνομο φέρων σήμα (Stand Alone)
- Σαν ενσωματωμένο σε κάποιο άλλο L.T.E. σήμα (In – Band)
- Σαν ενσωματωμένο στην ζώνη ασφαλείας κάποιου L.T.E. σήματος (Guard Band)

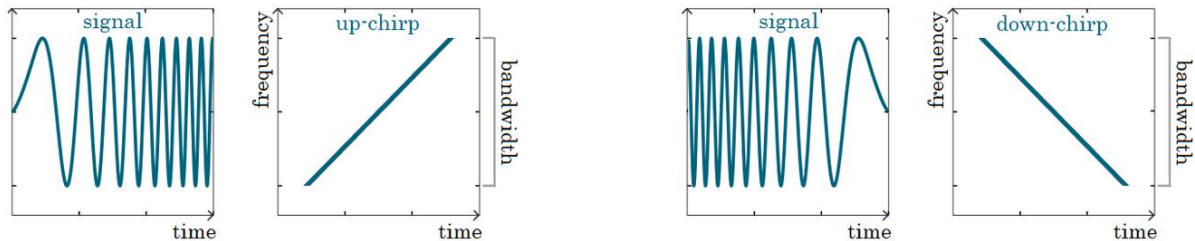
Η εικόνα 1.5 απεικονίζει τους τρεις διαφορετικούς τρόπους λειτουργίας του N.B. – I.o.T., αξίζει να σημειωθεί πως για την μείωση του κόστους των συσκευών και την απλούστευση της μορφής του πρωτοκόλλου αφαιρούνται πολλά από τα χαρακτηριστικά που περιλαμβάνει ένα L.T.E. σήμα, όπως η διαδικασία αλλαγής σε άλλο κανάλι μετάδοσης, οι μετρήσεις που αφορούν την παρακολούθηση ποιότητας του καναλιού και της τεχνικής Carrier Aggregation που εφαρμόζεται από το L.T.E. προκειμένου να επιτευχθούν υψηλές ταχύτητες δεδομένων στο 4G και της διπλής συνδεσιμότητας (Dual Connectivity).[1]



Εικόνα 1.5: Τρόποι λειτουργίας N.B. – I.o.T.

1.4.3.4 Τεχνολογία LoRa

Η ονοματολογία της τεχνολογίας LoRa προέρχεται από την συντομογραφία «μεγάλη εμβέλεια (Long Range)», η ανάπτυξη και προώθηση της LoRa τεχνολογίας γίνεται από την Semtech και την LoRa Alliance. Χρησιμοποιεί τεχνολογία ασύρματης διαμόρφωσης σήματος Chirp Spread Spectrum – C.S.S.. Το Chirp είναι ένα σήμα του οποίου η συχνότητα είτε αυξάνεται (up – chirp), είτε μειώνεται (down – chirp) συναρτήσει του χρόνου, όπως φαίνεται στην εικόνα 1.6.[1]



Εικόνα 1.6: Σήματα chirp up & chirp down

Το C.S.S. είναι μία τεχνική εξάπλωσης του φάσματος η οποία χρησιμοποιεί ευρυζωνικές γραμμικές συχνότητες που διαμορφώνονται με συχνότητα για κωδικοποίηση πληροφοριών. Η τεχνική εξάπλωσης του φάσματος περιλαμβάνει μεθόδους με τις οποίες ένα σήμα διασκορπίζεται σκοπίμως στην περιοχή συχνοτήτων, αυτό έχει ως αποτέλεσμα την δημιουργία ενός σήματος με μεγαλύτερο εύρος ζώνης. Μ' αυτού του είδους την τεχνική επιτυγχάνεται καλύτερη ποιότητα του σήματος και καλύτερη επικοινωνία μεταξύ των συσκευών, αυξάνοντας παράλληλα την αντίσταση σε φυσικές παρεμβολές και στον θόρυβο.[1]

Η χρήση της LoRa τεχνολογίας επιτρέπει στις συσκευές την μετάδοση δεδομένων σε μεγάλες αποστάσεις καταναλώνοντας λιγιστά ποσοστά ενέργειας, πράγμα που επιτρέπει την ενσωμάτωση μπαταριών. Η λειτουργία πραγματοποιείται στις ελεύθερες συχνότητες και δεν απαιτείται άδεια από την πολιτεία για την μετάδοση δεδομένων. Οι ελεύθερες ζώνες συχνοτήτων είναι για την Ευρώπη τα 868MHz, για την Αμερική τα 915MHz και για την Ασία τα 433MHz. Από το θεώρημα Shannon – Hartley είναι γνωστό πως, η αύξηση στο εύρος ζώνης ενός καναλιού μετάδοσης είναι ένας τρόπος ώστε να βελτιωθεί η κακή αναλογία σήματος προς θόρυβο (Signal to Noise Ratio – S.N.R.).[1]

Τέλος, μία ακόμα βελτίωση με την χρήση του πρωτοκόλλου LoRa είναι η προσαρμοστικότητα του στο να λειτουργεί σε αισθητήρες οι οποίοι διαθέτουν περιορισμένους ενεργειακούς πόρους. Αυτό συμβαίνει λόγω του ότι η κίνηση στο uplink τις περισσότερες φορές είναι μεγαλύτερη απ' ότι στο downlink σε I.o.T. εφαρμογές. Επιπλέον, η LoRa διαθέτει την κρυπτογράφηση A.E.S. – 128bit με ζεύγος κλειδιών πράγμα που καθιστά ασφαλές το κανάλι μετάδοσης από υποκλοπές δεδομένων. Στην συνέχεια της εργασίας πρόκειται να γίνει εκτενέστερη αναφορά στους διαφορετικούς τύπους μετάδοσης δεδομένων που έχει ορίσει η LoRa.[1]

1.5 Εφαρμογές του διαδικτύου των πραγμάτων (I.o.T.)

Οι εφαρμογές που έχει το I.o.T. έχουν ως σκοπό την διευκόλυνση της καθημερινότητας και την μεγιστοποίηση του κέρδους. Υπό αυτό το πρίσμα, οι εφαρμογές του I.o.T. μπορεί να προσαρμόζονται στις ανάγκες των χρηστών ή να ενσωματώνονται στις νέες τεχνολογίες. Όπως γίνεται κατανοητό, το I.o.T. είναι ένα πεδίο το οποίο διαθέτει πολλές δυνατότητες, όμως πάντα θα υπάρχουν προκλήσεις για την ανάπτυξη νέων εφαρμογών καθώς η εξέλιξη του αυξάνεται ραγδαία. Ορισμένοι τομείς ανάπτυξης του I.o.T. είναι οι μεταφορές, οι αυτοκινητοβιομηχανίες, η υγεία κ.α.[1]

1.5.1 I.o.T. & μεταφορές (Logistics)

Με την χρησιμοποίηση του I.o.T. στον τομέα των μεταφορών υπάρχει αναβάθμιση στην μεταφορά προϊόντων – αγαθών και υλικών από τον ένα σημείο στο άλλο. Επίσης, γίνεται ευκολότερη η παρακολούθηση κατά την διαδρομή τους (tracking) καθώς και ο αυτόματος προσδιορισμός για την κατάλληλη αποθήκευση τους αυξάνοντας έτσι την ενεργειακή απόδοση αυτών των συστημάτων καθώς μειώνονται οι ώρες εργασίας και η κατανάλωση ενέργειας.[1]

Το I.o.T. έχει επιφέρει σημαντικές αλλαγές στην παγκόσμια αλυσίδα εφοδιασμού, αυτό έχει πραγματοποιηθεί λόγω του συνεχούς συγχρονισμού των πληροφοριών, μ' αυτόν τον τρόπο επιτυγχάνεται η ανίχνευση των αντικειμένων σε πραγματικό χρόνο καθιστώντας την σύνθετη διαδικασία των μεταφορών διαφανή, ορατή και ελεγχόμενη.[1]

1.5.2 I.o.T. & αυτοκινητοβιομηχανία

Μία από τις πρώτες εφαρμογές του I.o.T. ήταν η εισαγωγή του στα αυτοκίνητα, επιτρέποντας έτσι την εμφάνιση της αυτόνομης οδήγησης (auto drive) και άλλων «έξυπνων» συστημάτων μεταφοράς. Με την εγκατάσταση αυτών των συστημάτων οι αυτοκινητοβιομηχανίες εστιάζουν στην αποτροπή θανατηφόρων τροχαίων ατυχημάτων, την μείωση της κυκλοφοριακής συμφόρησης και την ελαχιστοποίηση των επιπτώσεων του περιβάλλοντος.[1]

Όλα τα παραπάνω, μπορούν να πραγματοποιηθούν μετατρέποντας τα οχήματα σε κινητούς κόμβους προκειμένου να επιτρέπεται η μεταξύ τους σύνδεση με αποτέλεσμα την δημιουργία δικτύου μεγάλης εμβέλειας γνωστό και ως Internet of Vehicles – I.o.V., το I.o.V. χρησιμοποιεί αισθητήρες που υπάρχουν εγκαταστημένοι στα αυτοκίνητα συλλέγοντας δεδομένα από το περιβάλλον του κάθε αυτοκινήτου, μ' αυτόν τον τρόπο δημιουργείται ένα σύστημα το οποίο αποτελείται από έξυπνες συσκευές. Τα αυτοκίνητα μπορούν και επικοινωνούν μεταξύ τους συλλέγοντας πληροφορίες από τον δρόμο, τους πεζούς και από αισθητήρες που διαθέτουν τα ίδια, αυτά τα δεδομένα αποστέλλονται σε servers όπου εκεί γίνεται η επεξεργασία τους. Τα αποτελέσματα που προκύπτουν από την επεξεργασία συνδυάζονται με σκοπό την λήψη αποφάσεων σχετικά με όλες τις πιθανές αιτίες κινδύνου που μπορεί να υπάρξουν. Όλα τα παραπάνω έχουν σαν αποτέλεσμα να συμβάλουν στην ασφαλέστερη πλοήγηση του κάθε αυτοκινήτου, την διαχείριση της κυκλοφορίας καθώς επίσης στον έλεγχο της ρύπανσης.[1]

1.5.3 I.o.T. & έξυπνες πόλεις

Ένας ακόμη μεγάλος τομέας στον οποίο το I.o.T. βρίσκει εφαρμογή είναι η χρήση του στις καθημερινές ανάγκες των κοινωνιών. Οι «έξυπνες πόλεις» καλύπτουν μια μεγάλη ποικιλία των περιπτώσεων χρήσης του I.o.T., από την διαχείριση της κυκλοφορίας έως την διανομή νερού, την διαχείριση αποβλήτων, την αστική ασφάλεια και την παρακολούθηση του περιβάλλοντος. Για να επιτευχθούν όλα τα παραπάνω πρέπει όλες οι συσκευές που επιτηρούν μία πόλη να διαθέτουν πρόσβαση στο internet. Έτσι λοιπόν, επιγραμματικά οι βασικές εφαρμογές σε μία «έξυπνη πόλη» είναι:[1]

- Τα έξυπνα σπίτια (smart homes)
- Οι έξυπνες θέσεις στάθμευσης (smart parking)
- Τα έξυπνα συστήματα καιρού (smart weather systems)
- Τα συστήματα παρακολούθησης της κυκλοφορίας των οχημάτων (traffic monitoring systems)
- Παρακολούθηση της μόλυνσης του περιβάλλοντος (air pollution systems)
- Τα συστήματα ασφάλειας των πολιτών (citizens security systems)

1.5.4 I.o.T. & ηλεκτρονική υγεία

Μία ακόμα εφαρμογή του I.o.T. μπορεί να είναι η προσθήκη του στους τομείς της υγείας. Η ιατρική και η υγειονομική περίθαλψη εστιάζει στο έλεγχο και την πρόληψη διαφόρων καταστάσεων υγείας. Με την προσθήκη του I.o.T. στα δημόσια και ιδιωτικά νοσοκομεία, η ανίχνευση του ιστορικού των ασθενών γίνεται ευκολότερα, κάνοντας έτσι την ανθρώπινη αλληλεπίδραση αποτελεσματικότερη διότι επιτρέπει την παρακολούθηση των ασθενών.[1]

Επίσης, είναι καίριας σημασίας η ανάπτυξη συσκευών που επιτρέπουν στους ασθενείς να πραγματοποιούν τις καθημερινές τους ανάγκες, όπως για παράδειγμα την παρακολούθηση και τον μετρισμό των διατροφικών διαταραχών. Στο σενάριο αυτό οι ασθενείς με την νόσο Parkinson μπορούν να τρώνε με ευκολία χρησιμοποιώντας τα απτικά συστήματα. Πιο συγκεκριμένα, ένα γάντι το οποίο είναι εξοπλισμένο με μικροσκοπικούς δονητικούς κινητήρες (Microelectromechanical Systems – MEM.S.) μπορεί να εξουδετερώσει την αστάθεια της κίνησης των χεριών η οποία καταγράφεται από επιταχυνσιόμετρα.[1]

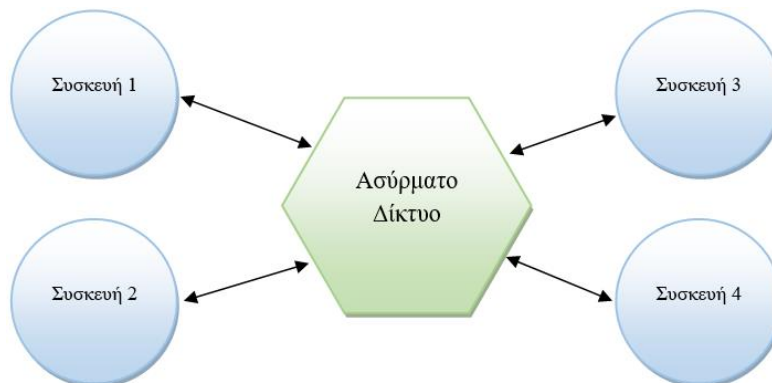
1.6 Διαδικτυακά μοντέλα επικοινωνιών του διαδικτύου των πραγμάτων (I.o.T.)

Το Μάρτιο του 2015, το συμβούλιο αρχιτεκτονικής διαδικτύου (Internet Architecture Board – I.A.B.) εξέδωσε έγγραφη αναφορά για την δικτύωση των έξυπνων αντικειμένων – συσκευών. Η RFC 7452, έχει ως σκοπό την περιγραφή ενός πλαισίου κοινών μοντέλων επικοινωνίας που χρησιμοποιούνται από τις συσκευές I.o.T., επιγραμματικά το πλαίσιο των κοινών μοντέλων επικοινωνίας είναι:[2]

- Συσκευής σε συσκευή (Device to Device)
- Συσκευής σε υπολογιστικό νέφος (Device to cloud)
- Συσκευής σε πύλη (Device to Gateway)

1.6.1 Μοντέλο επικοινωνίας συσκευής σε συσκευή (Device to Device)

Το μοντέλο επικοινωνίας συσκευής σε συσκευή αποτελεί την περίπτωση στην οποία απουσιάζει η χρήση του διακομιστή εφαρμογών. Πιο συγκεκριμένα, οι συσκευές είναι σε θέση να επικοινωνούν άμεσα μεταξύ τους, η επικοινωνία επιτυγχάνεται με την χρησιμοποίηση διαφόρων τύπων δικτύων. Επίσης, το σύνηθες σ' αυτό το μοντέλο επικοινωνίας είναι η χρησιμοποίηση πρωτοκόλλων, ορισμένα παραδείγματα του συγκεκριμένου μοντέλου επικοινωνίας είναι το Bluetooth, το Z – Wave ή το ZigBee 4. Στην εικόνα 1.7 απεικονίζεται το μπλοκ διάγραμμα του μοντέλου επικοινωνίας Device to Device.[2]



Εικόνα 1.7: Μοντέλο επικοινωνίας device to device

Για την λειτουργία αυτών των δικτύων, χρησιμοποιούνται πρωτόκολλα επικοινωνίας μέσω των οποίων οι συσκευές γίνονται ικανές να αποστέλλουν μηνύματα. Μία εφαρμογή του εν λόγω μοντέλου επικοινωνίας από συσκευή σε συσκευή είναι οι συσκευές που χρησιμοποιούνται σ' ένα «έξυπνο σπίτι» (smart home) όπως για παράδειγμα αισθητήρες συναγερμών, αυτόματοι διακόπτες ρολών, κάμερες, συστήματα φωτισμού κ.α., τέτοιου είδους εφαρμογές απαιτούν μικρό όγκο δεδομένων ενώ οι απαιτήσεις στην ταχύτητά τους είναι χαμηλές.[2]

1.6.2 Μοντέλο επικοινωνίας συσκευής σε υπολογιστικό νέφος (Device to Cloud)

Σ' αυτά τα μοντέλα επικοινωνίας πραγματοποιείται απευθείας σύνδεση με υπηρεσίες νέφους μέσω του internet για την ανταλλαγή μηνυμάτων ελέγχου με μια συσκευή I.o.T., η προσέγγιση αυτή εκμεταλλεύεται συχνά υπάρχον εγκαταστάσεις επικοινωνίας όπως οι ενσύρματες συνδέσεις ethernet ή οι ασύρματες συνδέσεις Wi – Fi προκειμένου να δημιουργηθεί σύνδεση μεταξύ μιας συσκευής και μιας διεύθυνσης δικτύου (IP address) η οποία συνδέεται με την ανάλογη cloud υπηρεσία. Στην εικόνα 1.8 παρουσιάζεται ένα μοντέλο επικοινωνίας Device to Cloud.[2]

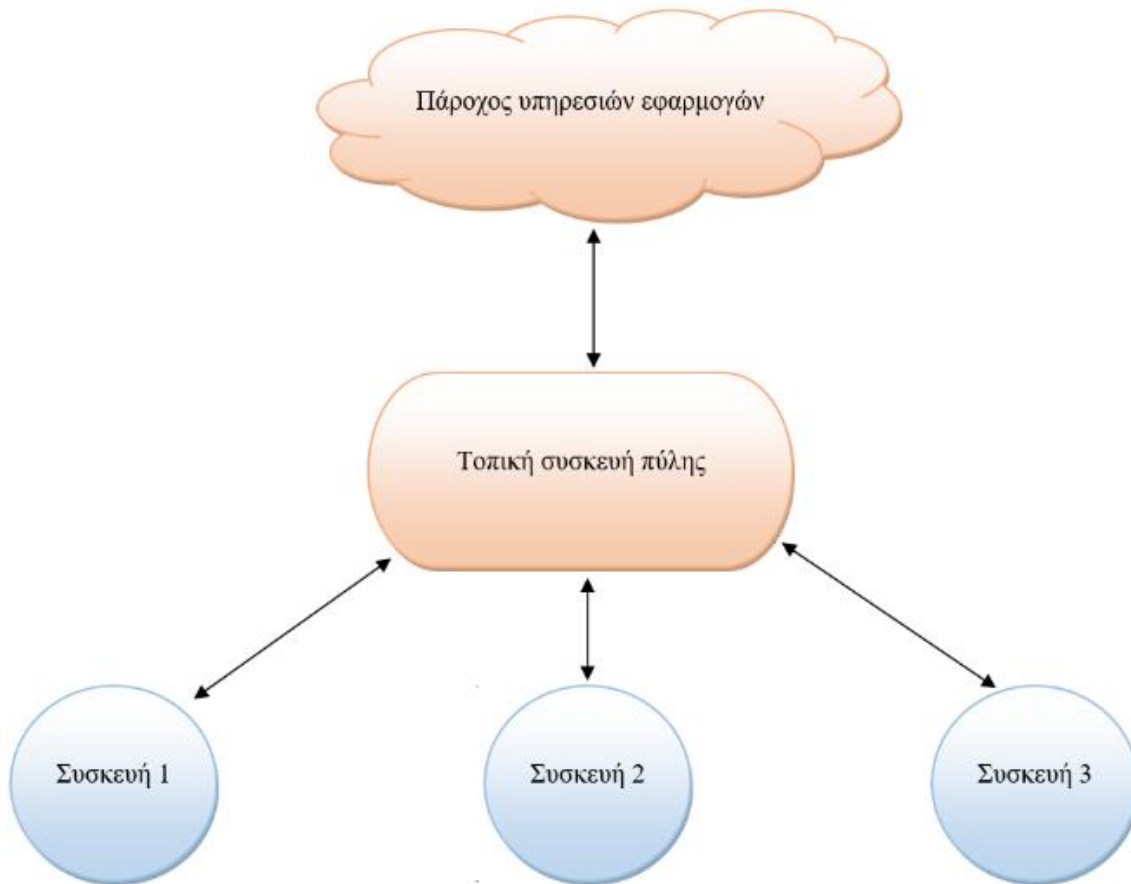


Εικόνα 1.8: Μοντέλο επικοινωνίας device to cloud

Οι συσκευές και οι υπηρεσίες cloud συνήθως προέρχονται από τον ίδιο τον πάροχο. Ένα παράδειγμα εφαρμογής αυτού του μοντέλου επικοινωνίας είναι οι «έξυπνες τηλεοράσεις» (smart T.V.s). Αυτού του είδους οι τηλεοράσεις, χρησιμοποιούν την σύνδεση στο διαδίκτυο προκειμένου να πραγματοποιηθεί η μετάδοση πληροφοριών μεταξύ της συσκευής και της αντίστοιχης cloud υπηρεσίας. Συχνά, προκύπτουν προκλήσεις δια λειτουργικότητας κατά την προσπάθεια ενσωμάτωσης συσκευών που είναι κατασκευασμένες από διαφορετικούς κατασκευαστές.[2]

1.6.3 Μοντέλο επικοινωνίας συσκευής σε πύλη (Device to Gateway)

Σ' αυτή την κατηγορία μοντέλων η πρόσβαση της I.o.T. συσκευής στην cloud υπηρεσία, γίνεται μέσω της πύλης επιπέδου εφαρμογής (Application Layer Gateway – A.L.G.), πρόκειται για μια κεντρική συσκευή η οποία λειτουργεί σαν πύλη μέσω της οποίας περνούν τα δεδομένα μετάδοσης προκειμένου να καταλήξουν στο cloud. Πιο συγκεκριμένα, η συσκευή αυτή παρεμβάλλεται μεταξύ των τοπικών συσκευών και της cloud υπηρεσίας και διαθέτει εγκατεστημένο λογισμικό εφαρμογών το οποίο είναι υπεύθυνο για την συλλογή και την αποστολή των δεδομένων στο cloud. Το συγκεκριμένο μοντέλο επικοινωνίας, παρέχει αυξημένα επίπεδα ασφάλειας κατά την αποστολή και την λήψη δεδομένων από το cloud. Η εικόνα 1.9 παρουσιάζει το μπλοκ διάγραμμα του μοντέλου επικοινωνίας device to gateway.[2]



Εικόνα 1.9: Μοντέλο επικοινωνίας device to gateway

Αυτό το μοντέλο επικοινωνίας επιτρέπει την ενσωμάτωση νέων συσκευών σ' ένα υπάρχον σύστημα που χρησιμοποιεί παλαιότερου τύπου συσκευές χωρίς να υπάρχει δια λειτουργικότητα μεταξύ τους. Η ανάπτυξη αυτών των μοντέλων επικοινωνίας αυξάνουν την πολυπλοκότητα και το κόστος συνολικά. Ένα παράδειγμα εφαρμογής αυτού του μοντέλου επικοινωνίας, μπορεί να είναι ένα smartphone το οποίο μπορεί να επικοινωνεί με μία τοπική συσκευή και μέσω μιας εφαρμογής να έχει τον ρόλο της τοπικής συσκευής πύλης μέσα από την οποία να μεταδίδει τα δεδομένα προς το cloud.[2]

1.7 Επίλογος

Ο σκοπός του κεφαλαίου αυτού είναι η διατύπωση του ορισμού του διαδικτύου των πραγμάτων σύμφωνα με τον οποίο πρόκειται για ένα δίκτυο το οποίο αποτελείται από συσκευές οι οποίες μπορούν και επικοινωνούν μεταξύ τους με σκοπό την ανταλλαγή δεδομένων. Η ανακάλυψη του διαδικτύου των πραγμάτων έγινε από τον Kevin Ashton και την ομάδα του στα τέλη της δεκαετίας του 1990. Οι εκτιμήσεις μιλούν για ραγδαία εξέλιξη που παρατηρείται στον τομέα αυτό, καθώς στο τέλος του τρέχοντος έτους, κάνουν λόγο για περισσότερες από 51 δισεκατομμύρια συσκευές που πρόκειται να βρίσκονται συνδεδεμένες στο διαδίκτυο.

Οι βασικές τεχνολογίες του διαδικτύου των πραγμάτων είναι η τεχνολογία Radio Frequency Identification – R.F.ID., η τεχνολογία Wireless Sensor Networks – W.S.N.s και η τεχνολογία Low Power Wide Area Networks – L.P.W.A.N.s η οποία περιλαμβάνει διάφορα είδη όπως είναι η Sigfox, η Weightless, η Narrow Band – I.o.T. και η LoRa/LoRa WAN.

Η ονομασία της LoRa τεχνολογίας προέρχεται από την συντομογραφία «μεγάλη εμβέλεια (Long Range)». Χρησιμοποιεί τεχνολογία ασύρματης διαμόρφωσης σήματος Chirp Spread Spectrum – C.S.S.. Το Chirp είναι ένα σήμα του οποίου η συχνότητα είτε αυξάνεται (up – chirp), είτε μειώνεται (down – chirp) συναρτήσει του χρόνου. Η λειτουργία πραγματοποιείται στις ελεύθερες συχνότητες και δεν απαιτείται άδεια από την πολιτεία για την μετάδοση των δεδομένων. Οι ελεύθερες ζώνες συχνοτήτων για την Ευρώπη είναι τα 868MHz, για την Αμερική τα 915MHz και για την Ασία τα 433MHz.

Κλείνοντας, αναλύονται τα διαδικτυακά μοντέλα επικοινωνιών του διαδικτύου των πραγμάτων τα οποία προέκυψαν το Μάρτιο του 2015 από το συμβούλιο αρχιτεκτονικής διαδικτύου (Internet Architecture Board – I.A.B.). Πρόκειται για τα διαδικτυακά μοντέλα συσκευής σε συσκευή (Device to Device), συσκευής σε υπολογιστικό νέφος (Device to cloud) και συσκευής σε πύλη (Device to Gateway).

Κεφάλαιο 2: Δίκτυα μεγάλης εμβέλειας (LoRa)

2.1 Εισαγωγή

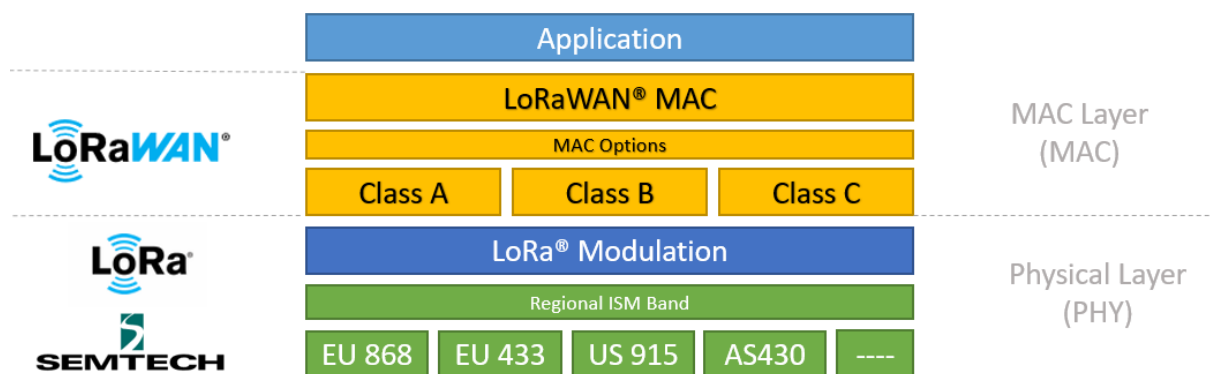
Αυτό το κεφάλαιο, αρχικά αναφέρεται στην στοίβα των πρωτοκόλλων των δικτύων LoRa. Έπειτα αναλύονται οι ραδιοσυχνότητες που λειτουργούν τα δίκτυα αυτά, καθώς επίσης και οι περιορισμοί που επιβάλλονται στα δίκτυα LoRa από την ευρωπαϊκή νομοθεσία. Στην συνέχεια, γίνεται αναφορά στο φυσικό επίπεδο LoRa και κατ' επέκταση στην υλοποίηση των σημάτων, στον παράγοντα διασποράς, στον υπολογισμό της ευαισθησίας του δέκτη, στις τεχνικές κωδικοποίησης και στην μορφή των LoRa πακέτων που χρησιμοποιούνται κατά την διαδικασία της διαμόρφωσης. Επί προσθέτου, για να καταστεί σαφές ο τρόπος λειτουργίας των δικτύων LoRa γίνεται αναφορά στην τοπολογία αυτών των δικτύων, στα πρωτόκολλα επικοινωνία τους καθώς επίσης και στις κλάσεις που ταξινομούνται. Κλείνοντας, γίνεται συνοπτική αναφορά στη δομή των πακέτων και των MAC εντολών που περιλαμβάνουν τα δίκτυα αυτά.

2.2 Στοίβα LoRa πρωτοκόλλων

Η στοίβα πρωτοκόλλων του δικτύου LoRa, αποτελείται συνολικά από 4 επίπεδα, καθ' ένα από τα οποία πραγματοποιεί συγκεκριμένες λειτουργίες χωρίς όμως να παρεμβαίνει στις λειτουργίες του άλλου. Πιο συγκεκριμένα, ο διαχωρισμός τους γίνεται ως εξής:

- Επίπεδο εφαρμογής (LoRa Application Layer)
- Επίπεδο ελέγχου προσπέλασης μέσω (LoRa M.A.C. Layer)
- Φυσικό επίπεδο (LoRa PHY Layer)
- Επίπεδο ραδιοσυχνοτήτων (LoRa RF Layer)

Όλα τα επίπεδα είναι αυστηρά διαχωρισμένα μεταξύ τους και έχουν ως ρόλο να αξιοποιούν τις υπηρεσίες του κατώτερου επιπέδου τους και να προσφέρουν υπηρεσίες στο ανώτερο τους επίπεδο. Αυτό έχει ως αποτέλεσμα, την διευκόλυνση τόσο της μελέτης τους, όσο και της σωστής σχεδίασής τους. Κάθε επίπεδο είναι υπεύθυνο για την υλοποίηση καθορισμένων πρωτοκόλλων, εστιάζοντας πάντα στην αυξημένη διάρκεια ζωής των τελικών συσκευών (End – Nodes) αλλά και στην ανθεκτικότητα του σήματος. Στην εικόνα 2.1 φαίνεται η δομή της στοίβας πρωτοκόλλων ενός LoRa δικτύου.[4]

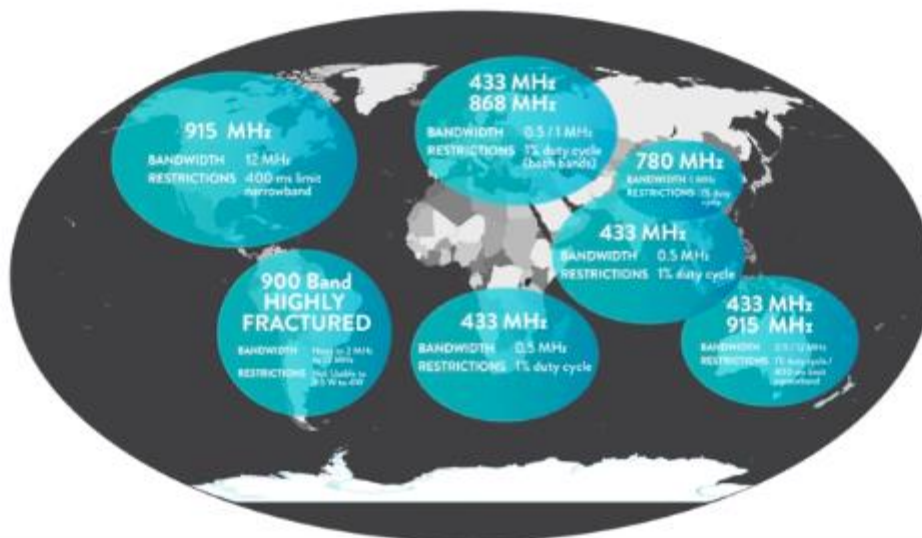


Εικόνα 2.1: Δομή στοίβας πρωτοκόλλων LoRa δικτύου

2.3 Ραδιοσυχνότητες δικτύου LoRa

Όπως αναφέρθηκε στο πρώτο κεφάλαιο της εργασίας, τα δίκτυα LoRa χρησιμοποιούν τις I.S.M. ζώνες συχνοτήτων. Η χρήση των I.S.M. ραδιοσυχνοτήτων δεν απαιτεί χορήγηση άδειας, οι I.S.M.

προορίζονται για χρήση εκπαιδευτικών, βιομηχανικών, επιστημονικών και ιατρικών σκοπών. Υπάρχει διεθνής αναγνώριση και η υιοθέτησή τους έγινε από την διεθνή ένωση τηλεπικοινωνιών (International Telecommunication Union – I.T.U.), παρ’ όλα αυτά υπάρχουν διαφοροποιήσεις σε κάθε χώρα λόγω διαφορετικών κανονισμών που υπάρχουν στην κατανομή των ραδιοσυχνοτήτων. Η χρήση των I.S.M. πραγματοποιείται κάτω από αυστηρούς κανονισμούς οι οποίοι αφορούν την ισχύ εκπομπής καθώς επίσης και τα όρια εκπομπής, προκειμένου να αποφευχθούν τυχόν παρεμβολές με ζώνες συχνοτήτων που χρειάζονται άδεια χρήσης. Στην Ευρώπη, οι περιορισμοί ορίστηκαν από τον ευρωπαϊκό οργανισμό τηλεπικοινωνιακών προτύπων (European Telecommunications Standards Institute – E.T.S.I.) και ανατέθηκε η μπάντα συχνοτήτων 867 – 869MHz. Η μέγιστη εκπεμπόμενη ισχύς τέθηκε στα +14 dBm, επίσης υπάρχει περιορισμός και στον αριθμό των καναλιών.[4]



Εικόνα 2.2: Παγκόσμιος χάρτης συχνοτήτων LoRa δικτύων

2.3.1 Ευρωπαϊκοί περιορισμοί των δικτύων LoRa

Η Ελλάδα εμπίπτει στους ευρωπαϊκούς κανονισμούς. Αυτό σημαίνει ότι επιτρέπεται η χρησιμοποίηση της ζώνης των 863 – 870MHz, κατά την εκπομπή ο εκάστοτε πομπός πρέπει να διαθέτει κάποια Listen – Before – Talk τεχνική ή να έχει περιορισμένο κύκλο λειτουργίας (duty cycle) μετάδοσης με σκοπό την μείωση του ρυθμού παραγωγής μηνυμάτων. Επίσης, υπάρχουν περιορισμοί στην μέγιστη ισχύ εκπομπής και περιγράφονται με τρεις διαφορετικούς τρόπους ως εξής:

- Το μέτρο της ακτινοβολούμενης ισχύς του ηλεκτρικού πεδίου σε συγκεκριμένη απόσταση από την κεραία εκπομπής
- Το μέτρο της ισότροπα ακτινοβολούμενης ισχύς (Effective Isotropic Radiated Power – E.I.R.P.), το μέγεθος αυτό αποτελεί μία μέτρηση της ισχύς που πρέπει να ακτινοβολεί μία υποθετική ιστροπική κεραία, προκειμένου να λαμβάνει το ίδιο ακριβώς επίπεδο σήματος στην κατεύθυνση της μέγιστης ακτινοβολίας μιας συγκεκριμένης κεραίας. Το μέγεθος αυτό προκύπτει από την σχέση:

$$E.I.R.P._{dBm} = 10 \log \left(\frac{E^2 r^2}{0.03} \right)$$

- Το μέτρο της ακτινοβολούμενης ισχύς (Effective Radiated Power – E.R.P.), το οποίο ορίζεται όπως η E.I.R.P. με την διαφορά ότι θεωρείται μία κεραία half – wave dipole, αντί μίας ιστροπικής. Το μέγεθος αυτό προκύπτει από την σχέση:

$$E.I.R.P._{(w)} = 1.64E.R.P._{(w)} \text{ ή } E.I.R.P._{dBm} = E.R.P._{dBm} + 2.15$$

Για την εκτίμηση της ισχύς εκπομπής και τον έλεγχο της αν βρίσκεται σε επιτρεπτά όρια χρησιμοποιείται το μέτρο της E.R.P. και αυτό δεν πρέπει να υπερβαίνει τα +14 dBm.

Ένας επίσης περιορισμός είναι του κύκλου λειτουργίας (duty cycle) της μετάδοσης. Το duty cycle είναι το μέγιστο ποσοστό του χρόνου κατά το οποίο ένας πομπός μπορεί να καταλαμβάνει ένα συγκεκριμένο κανάλι. Το γεγονός αυτό, καθορίζει την επιλογή του καναλιού να γίνεται με τυχαίο τρόπο για κάθε μετάδοση, προκειμένου αυτή να είναι σύμφωνη με το μέγιστο duty cycle. Παρ' όλα αυτά, ένας πομπός μπορεί να χρησιμοποιήσει για την μετάδοσή του περισσότερα από ένα κανάλια, εφόσον αυτά είναι ελεύθερα. Στον πίνακα που ακολουθεί αναγράφονται τα διάφορα κανάλια που υπάρχουν για την Ευρώπη και τους αντίστοιχους περιορισμούς που αυτά έχουν. Τέλος, ο περιορισμός των +27dBm αποτελεί εξαίρεση για την Ελλάδα καθώς απαγορεύεται η εκπεμπόμενη ισχύς άνω των +20dBm.[4]

Πίνακας 2.1: Κανονισμοί LoRa καναλιών όπως ορίστηκαν από τον E.T.S.I.

#	f	BW	% of time on air	Max E.R.P.
1	868.1	125kHz	1% ToA = 36s/h	14dBm
2	868.3	125kHz	1% ToA = 36s/h	14dBm
3	868.5	125kHz	1% ToA = 36s/h	14dBm
4	868.85	125kHz	0.1% ToA = 3.6s/h	14dBm
5	869.05	125kHz	0.1% ToA = 3.6s/h	14dBm
6	868.525	125kHz	1% ToA = 360s/h	27dBm

2.4 Φυσικό επίπεδο LoRa – LoRa PHY Layer

Αρχικά, να σημειωθεί πως δεν υπάρχει σαφής περιγραφή της τεχνικής διαμόρφωσης που χρησιμοποιείται στα συστήματα LoRa, αυτό συμβαίνει γιατί υπάρχει κατοχυρωμένο δίπλωμα ευρεσιτεχνίας από την εταιρία Semtech. Ωστόσο, έχουν πραγματοποιηθεί αρκετές προσπάθειες αποκρυπτογράφησης προκειμένου να γίνει γνωστή η τεχνική που χρησιμοποιείται. Σύμφωνα με έρευνα που διεξήγαγε ο Matthew Knight προκύπτει ότι η διαμόρφωση LoRa βασίζεται στην τεχνική C.S.S., για να γίνει κατανοητός ο τρόπος λειτουργίας των LoRa συστημάτων πρέπει να αναλυθούν τα εξής:[4]

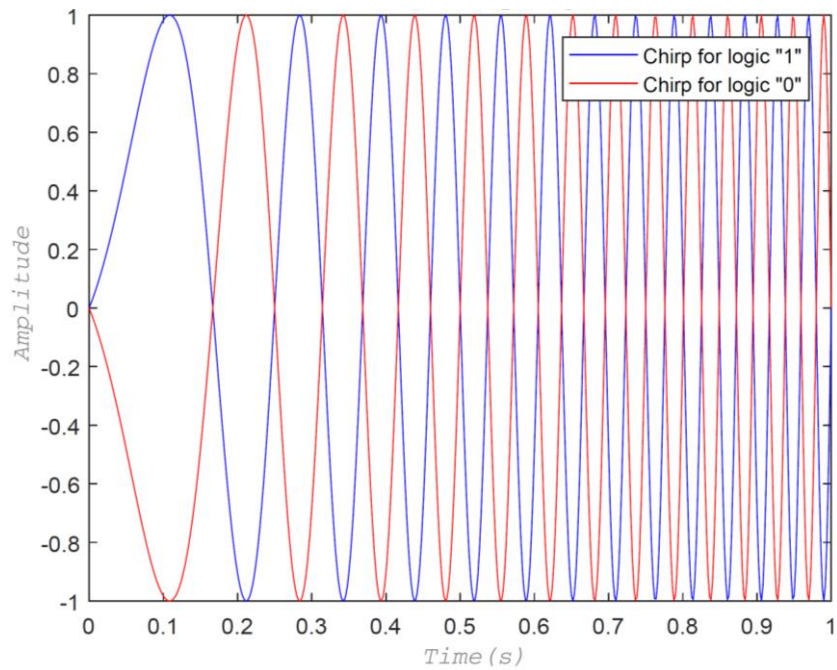
- Η υλοποίηση των σημάτων chirp
- Ο παράγοντας διασποράς (Spreading Factor)
- Ο υπολογισμός ευαισθησίας
- Οι τεχνικές κωδικοποίησης
- Η μορφή των πακέτων

2.4.1 Υλοποίηση των chirp σημάτων στην τεχνική διαμόρφωσης LoRa

Όπως έχει αναφερθεί στην υπό – ενότητα 1.4.3.4, η LoRa τεχνολογία χρησιμοποιεί chirp σήματα, το χαρακτηριστικό αυτών των σημάτων είναι ότι η συχνότητα τους μπορεί είτε να αυξάνεται, είτε να μειώνεται συναρτήσει του χρόνου. Επίσης, η μεταβολή της συχνότητας αυτών των σημάτων μπορεί να μεταβάλλεται είτε γραμμικά, είτε εκθετικά. Η κωδικοποίηση των δεδομένων γίνεται σε ημιτονοειδές γραμμικά μεταβαλλόμενης συχνότητας παλμούς, πιο συγκεκριμένα το λογικό «1» και το λογικό «0» παριστάνονται από ίδιο πλάτος σήματος με αντίθετη φάση, όπως απεικονίζεται στην εικόνα 2.3.

Ο γραμμικά μεταβαλλόμενης συχνότητας παλμός σταθερής διάρκειας, μπορεί για την διάδοση της πληροφορίας να καταλάβει ευρύτερο φάσμα απ' ότι θα καταλάμβανε κανονικά. Αυτό το

χαρακτηριστικό της ομοιόμορφης κατανομής οποιουδήποτε συμβόλου σε μεγαλύτερο εύρος ζώνης παρουσιάζει σημαντικά πλεονεκτήματα. Πιο συγκεκριμένα, η χρησιμοποίηση όλου του διαθέσιμου εύρους ζώνης που διατίθεται για την μετάδοση του σήματος, κάνει το διαμορφωμένο σήμα πιο ανθεκτικό στον θόρυβο του καναλιού. Επί προσθέτου, η αξιοποίηση όλου του εύρους ζώνης, προσδίδει αντίσταση σε συχνοτικά επιλεκτικό θόρυβο αλλά και σε κάθε είδους παρεμβολή, καθώς η περίοδος των συμβόλων LoRa, μπορεί να είναι μεγαλύτερη σε σχέση με τις άλλες τεχνικές διαμόρφωσης άλματος συχρότητας. Τέλος, είναι μία καλή λύση σε σχέση με τις υπόλοιπες διαμορφώσεις διασποράς φάσματος (Direct – sequence spread spectrum – D.S.S.S.), καθώς μπορεί και επιτυγχάνει καλές επιδόσεις διατηρώντας το κόστος και την πολυπλοκότητα σε χαμηλά επίπεδα.[4]



Εικόνα 2.3: Λογικό «0» και λογικό «1» chirp σήματος

Για κάθε bit, γίνεται πολλαπλασιασμός μεταξύ της δυαδικής ακολουθίας πληροφορίας με μία ακολουθία chirp. Να σημειωθεί πως ο ρυθμός των bits μιας chirp ακολουθίας είναι μεγαλύτερος σε σχέση με τον ρυθμό της ακολουθίας πληροφορίας, αυτό σημαίνει πως κάθε σύμβολο πληροφορίας αποτελείται από πολλούς chirp παλμούς. Στην εικόνα 2.4, φαίνονται οι ακολουθίες chirp για λογικό «0» και «1» οι οποίες αποτελούνται από 7 παλμούς η κάθε μία, καθώς επίσης και η δυαδική ακολουθία πληροφορίας.[4]



Εικόνα 2.4: Διαδικασία κωδικοποίησης LoRa σήματος

Στην έξοδο του κωδικοποιητή ο παλμός chirp έχει την μορφή της ακολουθίας πληροφορίας όπως απεικονίζεται στην εικόνα 2.5.



Εικόνα 2.5: Κωδικοποιημένο σήμα LoRa

2.4.2 Παράγοντας διασποράς στην τεχνική διαμόρφωσης LoRa (Spreading Factor)

Ο παράγοντας διασποράς (Spreading Factor – S.F.) στην διαμόρφωση LoRa αποτελεί ένα πολύ σημαντικό δεδομένο, ορίζεται από την μαθηματική σχέση:

$$S.F. = \frac{R_{chirp}}{R_S}$$

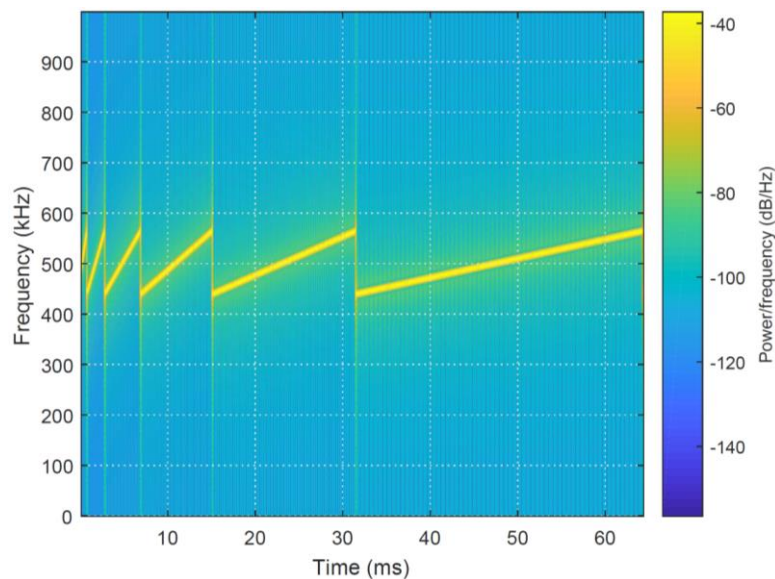
όπου, το R_{chirp} είναι ο ρυθμός του σήματος chirp και το R_S είναι ο ρυθμός του συμβόλου. Το μέγεθος της τιμής του S.F. προσδιορίζει τον ρυθμό των κωδικοποιημένων bits ανά σύμβολο και έχει εύρος τιμών από 7 ως 12. Ο ρυθμός του σήματος chirp ορίζεται ως η πρώτη παράγωγος της συχνότητας του chirp παλμού, ο οποίος αποτελεί συνάρτηση του παράγοντα διασποράς και του εύρους ζώνης. Δίνεται από την μαθηματική σχέση:

$$R_{chirp} = \frac{df}{dt} = \frac{BW}{2^{S.F.}}$$

Σύμφωνα με την αντιστροφή της παραπάνω μαθηματικής σχέσης προκύπτει πως, ο S.F. είναι υπεύθυνος για την διάρκεια του συμβόλου LoRa, καθώς ισχύει ότι:

$$T_S = \frac{1}{R_{chirp}} = \frac{2^{S.F.}}{BW}$$

Υποθέτοντας ότι χρησιμοποιείται ένα καθορισμένο εύρος ζώνης, τότε αυτό που ισχύει είναι ότι για κάθε αύξηση του S.F. κατά μία μονάδα ισοδυναμεί σε διπλάσια αύξηση διάρκειας ενός συμβόλου, όπως φαίνεται στην εικόνα 2.6.



Εικόνα 2.6: Σύγκριση παραγόντων διασποράς από S.F. = 7 ως S.F. = 12

Η μεγαλύτερη διάρκεια του συμβόλου, εξασφαλίζει την αξιόπιστη μεταφορά της πληροφορίας, καθώς το διαμορφωμένο σήμα έχει υψηλότερη ανθεκτικότητα στις παρεμβολές και στον θόρυβο. Συνεπώς, η αύξηση του S.F. οδηγεί σε αύξηση της ευαισθησίας του δέκτη μεγαλώνοντας μ' αυτόν τον τρόπο την ακτίνα κάλυψης του συστήματος. Η επιλογή ενός μεγάλου S.F. μεγαλώνει την πιθανότητα μετάδοσης λάθος πληροφορίας, όπως επίσης μπορεί να δημιουργηθούν συγκρούσεις μηνυμάτων, επομένως αυξάνοντας την τιμή του S.F. συνιστάται η αποστολή μικρής έκτασης μηνυμάτων.[4]

Να αναφερθεί ότι ο S.F. πρέπει να είναι γνωστός τόσο στον πομπό όσο και στον δέκτη, επίσης το S.N.R. είναι το ελάχιστο δυνατό όριο στο οποίο ο δέκτης μπορεί να λάβει ικανοποιητικά το σήμα προκειμένου να προχωρήσει σε αποδιαμόρφωση. Τέλος, εξαιτίας της γραμμικότητας που παρουσιάζουν οι chirp παλμοί έχει παρατηρηθεί ισοδυναμία μεταξύ συχνοτικών (frequency offsets) και χρονικών αντισταθμίσεων (timing offsets), αυτού του είδους οι αντισταθμίσεις μπορούν εύκολα να εξαλειφθούν από τον αποδιαμορφωτή όταν υπάρχει σωστός συγχρονισμός μεταξύ του πομπού και του δέκτη. Συνεπώς, η διαμόρφωση LoRa είναι εξίσου καλή για την αντιμετώπιση του φαινομένου Doppler. Στον πίνακα 2.2 παρουσιάζονται οι παράμετροι της LoRa διαμόρφωσης και πως επηρεάζονται από τον S.F.[4]

Πίνακας 2.2: Παράμετροι που επηρεάζονται από τον S.F.

Spreading Factor	Chirps/Symbol	S.N.R. limit	Time - on - air (10-byte packet)	Bitrate
7	128	-7,5	56 ms	5469 bps
8	256	-10	103 ms	3125 bps
9	512	-12,5	205 ms	1758 bps
10	1024	-15	371 ms	977 bps
11	2048	-17,5	741 ms	537 bps
12	4096	-20	1483 ms	293 bps

2.4.3 Υπολογισμός ευαισθησίας και Link Budget

Η μαθηματική σχέση που ορίζει την ευαισθησία του δέκτη ορίζεται παρακάτω:

$$S = N_t + 10 \log_{10}(BW) + N.F. + S.N.R. \text{ dBm}$$

στην σχέση αυτή ο όρος N_t αντιπροσωπεύει τον θερμικό θόρυβο στον δέκτη, ο δεύτερος όρος προσδιορίζει την μείωση της ευαισθησίας λόγω του απαιτούμενου εύρους ζώνης για την μετάδοση της πληροφορίας, ο τρίτος όρος προσδιορίζει την εικόνα θορύβου και είναι συγκεκριμένη για κάθε ηλεκτρονικό σύστημα, τέλος ο όρος S.N.R. προσδιορίζει το απαιτούμενο λόγο σήματος προς θόρυβο που πρέπει να υπάρχει στον αποδιαμορφωτή προκειμένου να γίνει αξιόπιστη λήψη. Η παρακάτω σχέση εκφράζει τον θερμικό θόρυβο στον δέκτη και είναι:

$$N_t = 10 \log_{10}(1000kT) \text{ dBm} + 10 \log_{10}(\Delta_f) \text{ dBm}$$

η μεταβλητή k είναι ο σταθερά του Boltzmann, η μεταβλητή T (Kelvin) είναι η θερμοκρασία του περιβάλλοντα χώρου, ο δεύτερος όρος είναι το εύρος ζώνης του θορύβου. Η μέγιστη ενδεικτική τιμή της ευαισθησίας του δέκτη σε μια διαμόρφωση LoRa φτάνει ως και -148 dBm.[4]

Το link budget προσδιορίζει όλα τα κέρδη και όλες τις απώλειες που υπάρχουν σε μία τηλεπικοινωνιακή ζεύξη και δίνεται από την σχέση:

$$P_{RX} \text{ (dBm)} = P_{TX} \text{ (dBm)} + G_{system} \text{ (dB)} - l_{system} \text{ (dB)} - l_{channel} \text{ (dB)} - M \text{ (dB)}$$

το P_{RX} είναι η αναμενόμενη ισχύ στον δέκτη, το P_{TX} είναι η ισχύς του πομπού, το G_{system} είναι τα κέρδη που έχει το σύστημα και προκύπτουν από κεραιές κτλ., το l_{system} είναι οι απώλειες του συστήματος και προκύπτουν από γραμμές μεταφοράς κτλ., το $l_{channel}$ είναι οι απώλειες του καναλιού ο υπολογισμός τους γίνεται εμπειρικά με βάση μετρήσεις που έχουν γίνει και το M είναι το fading margin. Το link budget μπορεί να υπολογιστεί πιο απλά από την σχέση:

$$Link\ Budget = P_{TX} - S$$

Αν ληφθούν υπόψη ότι η μέγιστη επιτρεπόμενη ισχύς μετάδοσης είναι τα +20 dBm, τότε το μέγιστο link budget θα είναι 168 dB, το νούμερο αυτό επαληθεύεται από τα data sheet που παρέχει η Semtech σε αντίστοιχα chip. Στην πράξη, η ευαισθησία του δέκτη και το link budget έχουν μικρότερη επίπτωση στην ποιότητα των υπηρεσιών (Quality of Service – Q.o.S.) σε σχέση με τις παρεμβολές. Τέλος, να αναφερθεί πως δεν υπάρχει εγγυημένο Q.o.S. λόγω του ότι η χρησιμοποίηση συχνοτήτων που λειτουργούν σε L.P.W.A.N.s δεν χρειάζονται άδεια.[4]

2.4.4 Τεχνικές κωδικοποίησης στην διαμόρφωση LoRa

Οι τεχνικές κωδικοποίησης εξασφαλίζουν την μετάδοση της πληροφορίας προκειμένου να επιτευχθεί η αξιόπιστη μεταφορά τους, επιγραμματικά οι τεχνικές που χρησιμοποιούνται στην διαμόρφωση LoRa είναι:

- Data Whitening
- Gray Indexing
- Forward Error Correction
- Interleaving

2.4.4.1 Data Whitening

Σ' αυτή την τεχνική, εφαρμόζεται ένας γραμμικός μετασχηματισμός στα δεδομένα και έχει ως σκοπό την τυχαιοποίησή τους. Κατά την αρχική μορφή των δεδομένων παρουσιάζεται συσχέτιση μεταξύ τους η οποία προκύπτει αυτόματα, ο μετασχηματισμός αυτός αναίρει την συσχέτιση που εμφανίζεται και τα μετατρέπει σε ασυσχέτιστα. Συνεπώς, με την παρουσία μιας τυχαίας ακολουθίας (whitening), επιτυγχάνεται η ομοιόμορφη κατανομή της πληροφορίας σε όλο το εύρος ζώνης τους καναλιού. Να αναφερθεί πως η whitening ακολουθία είναι γνωστή και στον πομπό και στον δέκτη.[4]

2.4.4.2 Gray Indexing

Η τεχνική αυτή, ορίζει τα κωδικοποιημένα σύμβολα με τέτοιο τρόπο προκειμένου να πετύχει μέγιστη διαφορά κατά 1 bit μεταξύ δύο διαδοχικών συμβόλων. Η ενέργεια αυτή, έχει ως σκοπό την αποτροπή των off – by – one σφαλμάτων κατά την διαδικασία ανάκτησης των συμβόλων. Μ' αυτόν τον τρόπο, αυξάνονται οι πιθανότητες να διορθωθούν σφάλματα από τον κώδικα του καναλιού.[4]

2.4.4.3 Forward Error Correction

Η τεχνική αυτή, επιτυγχάνει την προς τα εμπρός διόρθωση των σφαλμάτων αποτελώντας μία από τις πιο διαδεδομένες τεχνικές που χρησιμοποιούνται για τον έλεγχο σφαλμάτων στην ψηφιακή μετάδοση πληροφορίας σ' ένα θορυβώδες κανάλι. Η εφαρμογή της εν λόγω τεχνικής στην διαμόρφωση LoRa, γίνεται με την μορφή κώδικα Hamming(M,N). Πιο συγκεκριμένα, το μήκος της λέξης ενός κώδικα πληροφορίας μήκους N είναι καθορισμένο και ισούται με 4 bits, επίσης η κωδική λέξη μήκους M είναι μία παράμετρος η οποία ρυθμίζεται από το σύστημα με μήκος $\frac{5}{8}$ bits. Συνεπώς, ο ρυθμός bit του

σήματος πληροφορίας επηρεάζεται λόγω της συγκεκριμένης απαραίτητης υλοποίησης. Ο ρυθμός bit δίνεται σύμφωνα με την παρακάτω μαθηματική σχέση:

$$R_b = S.F. \cdot \frac{4 + CR}{2^{S.F.}} \cdot \frac{4}{BW}$$

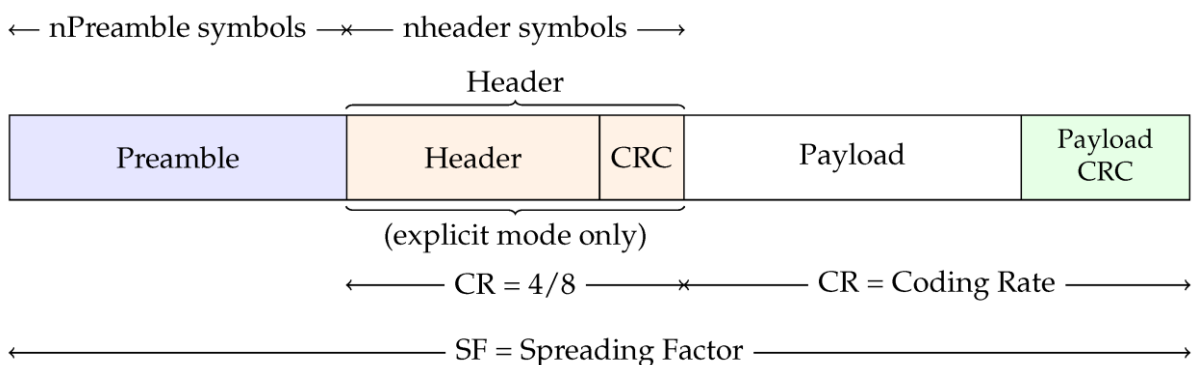
όπου το CR είναι το code rate (1...4).[4]

2.4.4.4 Interleaving

Στην τεχνική αυτή, είναι απαραίτητη η χρήση αναδιατάκτη καθώς βελτιώνει την επίδοση forward error correction κωδικών. Στην ουσία, γίνεται αναδιάταξη της εξόδου του forward error correction κωδικοποιητή, κάνοντας τον κώδικα πιο ανθεκτικό σε «λάθη ριπών» (burst errors). Η εμφάνιση των σφαλμάτων γίνονται κατά ομάδες σε μεμονωμένα χρονικά διαστήματα με αποτέλεσμα να μην είναι ίσο – κατανομημένα σε όλη την διάρκεια του μηνύματος με τυχαίο τρόπο. Επομένως, η αναδιάταξη των bits της πληροφορίας από – συσχετίζει το μήνυμα από τον χρόνο, κατ' αυτόν τον τρόπο τα λάθη που προκύπτουν σε χρονικά συνεχόμενα bits δεν είναι συνεχόμενα μέσα στο πραγματικό σήμα. Στην LoRa διαμόρφωση γίνεται η χρησιμοποίηση διαγώνιου αναδιατάκτη, αντιστρέφοντας τα δύο πιο σημαντικά δυαδικά ψηφία. Τα bits μέσα σε κάθε κωδική λέξη είναι ανεστραμμένα, επίσης κάθε διαγώνια λέξη πληροφορίας είναι μετατοπισμένη ή μπορεί να έχει περιστραφεί.[4]

2.4.5 Μορφή LoRa πακέτων

Ένα πακέτο LoRa περιλαμβάνει δύο μορφές, την μορφή explicit και την μορφή implicit. Η βασική διαφορά των δύο μορφών, είναι ότι στην μορφή explicit το πακέτο περιλαμβάνει επί πρόσθετα μία κεφαλίδα (header) η οποία συμπεριλαμβάνει πληροφορίες όπως τον αριθμό των bytes, το coding rate και την ύπαρξη ή όχι ενός δεκαεξάμπτου κυκλικού ελέγχου απόρριψης (16 bits Cyclic Redundancy Check – C.R.C.) για το Payload. Στην μορφή implicit στον δέκτη, είναι γνωστά όλα τα παραπάνω επομένως η χρησιμοποίηση της κεφαλίδας δεν είναι απαραίτητη γι' αυτό αφαιρείται. Συνεπώς, η διάρκεια του συμβόλου είναι μικρότερη αυτό έχει ως αποτέλεσμα την ελάττωση του χρόνου μετάδοσης. Η δομή ενός πακέτου LoRa απεικονίζεται στην εικόνα 2.7.[4]



Εικόνα 2.7: Δομή πακέτου LoRa

Η βασική χρησιμοποίηση του προοιμίου (Preamble) γίνεται για τον σωστό συγχρονισμό του δέκτη με την εισερχόμενη ροή των δεδομένων. Το payload είναι ένα πεδίο μεταβλητού μήκους το οποίο περιλαμβάνει κωδικοποιημένα δεδομένα, τα οποία έχουν καθοριστεί στην κεφαλίδα (header). Όπως γίνεται κατανοητό, ο χρόνος διάρκειας της μετάδοσης ενός πακέτου (Time on Air – T.o.A.), είναι άμεσα συνυφασμένος με τον S.F., το BW και τον ρυθμό του κώδικα (Coding Rate – C.R.).[4]

Πιο συγκεκριμένα, ο χρόνος μετάδοσης είναι ένα σημαντικό στοιχείο που αφορά την διαμόρφωση LoRa, αυτό συμβαίνει καθώς υπάρχουν περιορισμοί που προκύπτουν από το πρωτόκολλο LoRa W.A.N., για τον υπολογισμό της διάρκειας ενός πακέτου LoRa χρησιμοποιούνται οι παρακάτω σχέσεις:

$$t_{packet} = t_{preamble} + t_{payload}$$

όπου:

$$t_{preamble} = (n_{preamble} + 4)T_s$$

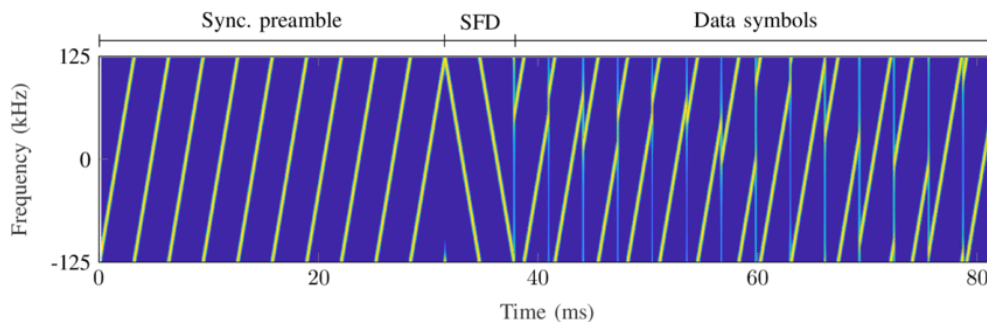
$$t_{payload} = n_{payload}T_s$$

Το $n_{preamble}$ είναι το μήκος του preamble και αποτελεί παράμετρο η οποία μπορεί να ρυθμιστεί από τον σχεδιαστή και επιδρά στον αριθμό των συμβόλων του preamble. Επίσης το $n_{payload}$ εξαρτάται από την μορφή της κεφαλίδας και αποτελεί τον αριθμό των payload συμβόλων. Ο υπολογισμός του $n_{payload}$ είναι αρκετά πολύπλοκος και ορίζεται από την σχέση:[4]

$$n_{payload} = 8 + \max \left(\text{ceil} \left[\frac{8PL - 4SF + 44 - 20H}{4(SF - 2DE)} \right] (CR + 4), 0 \right)$$

Το PL είναι ο αριθμός των payload bytes, το SF είναι ο παράγοντας διασποράς, το H μπορεί να λάβει την τιμή 0 όταν η μορφή του πακέτου είναι explicit ή την τιμή 1 όταν η μορφή του πακέτου είναι implicit, το DE μπορεί να λάβει την τιμή 0 όταν ο αλγόριθμος βελτιστοποίησης low data rate είναι ενεργοποιημένος (Low Data Rate Optimize = 1) ή την τιμή 1 όταν ο αλγόριθμος βελτιστοποίησης low data rate είναι απενεργοποιημένος (Low Data Rate Optimize = 0), τέλος το CR είναι ο αριθμός των bit ισοτιμίας (parity bits) που προστίθενται. Η βελτιστοποίηση Low Data Rate είναι μία μέτρηση που πραγματοποιείται για τον συγχρονισμό μεταξύ του πομπού και του δέκτη όταν υπάρχει αποστολή μεγάλης διάρκειας συμβόλων.[4]

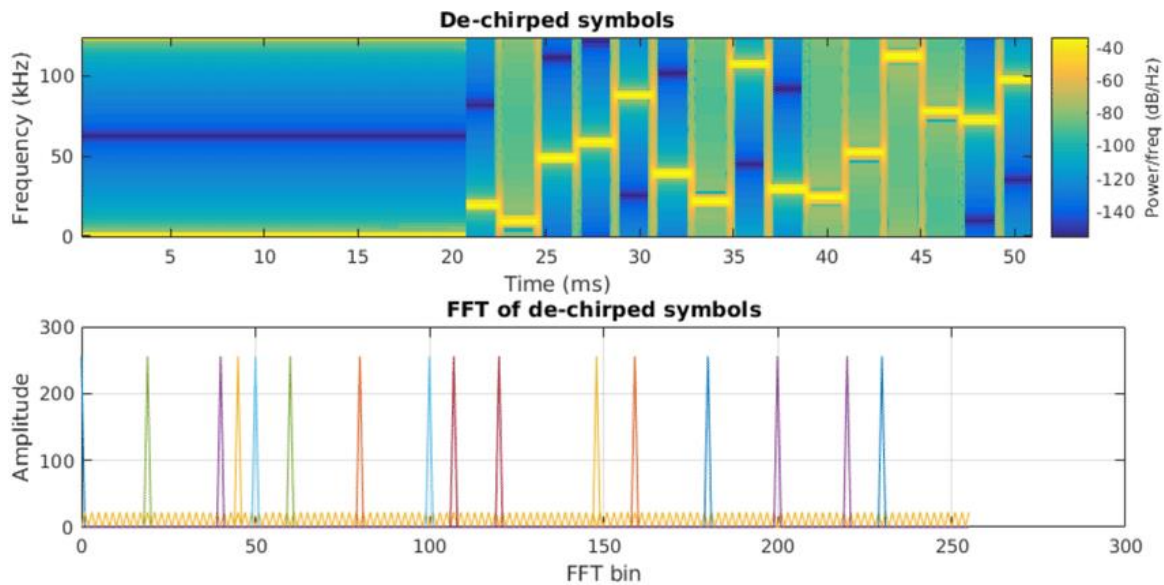
Ένα παράδειγμα ενός LoRa πακέτου παρουσιάζεται στην εικόνα 2.8 η οποία δείχνει ένα φασματογράφημα το οποίο απεικονίζει τον χρόνο στον οριζόντιο άξονα και την συχνότητα στον κατακόρυφο.[4]



Εικόνα 2.8: Πακέτο LoRa

Αυτό που παρατηρείται σε φυσικό επίπεδο είναι τα σήματα chirp, τα οποία σαρώνουν την ζώνη συχνοτήτων. Αρχικά, πραγματοποιούνται μερικές επαναλήψεις της συγκεκριμένης συχνότητας που σαρώνεται και αποτελεί το preamble του μηνύματος πληροφορίας, στην συνέχεια τα δεδομένα κωδικοποιούνται στο σήμα ως απότομες αλλαγές στην συχνότητα του σήματος chirp, όπως φαίνονται στην παραπάνω εικόνα. Η πιο διαδεδομένη διαδικασία αποδιαμόρφωσης είναι, ο προσδιορισμός της αρχής του πακέτου (preamble synchronization), ο προσδιορισμός της αρχής της κωδικοποιημένης πληροφορίας, η εξαγωγή της πληροφορίας από τις διαφορετικές μεταβάσεις της φάσης των σημάτων chirp.[4]

Πιο συγκεκριμένα, το σήμα σε πρώτη φάση υπόκειται σε διαδικασία de – chirping, στην συνέχεια εφαρμόζεται FFT στο de – chirped σήμα όπως φαίνεται παρακάτω στην εικόνα 2.9.



Εικόνα 2.9: Απεικόνιση de – chirped σήματος

Ο αριθμός των frequency bits είναι ίσος με τον αριθμό των συμβολών M που αντιστοιχεί στον S.F. που χρησιμοποιείται. Το de – chirped σήμα είναι έτοιμο να αποδιαμορφωθεί σύμφωνα με την τεχνική αποδιαμόρφωσης Multiple Frequency – Shift Keying (M.F.S.K.), η διαδικασία αυτή είναι πολύπλοκη καθώς απαιτείται η εφαρμογή πολλαπλών overlapping FFTs, ώστε να είναι ξεκάθαρη η ανίχνευση των συμβόλων.[4]

2.5 Δίκτυα LoRa W.A.N.

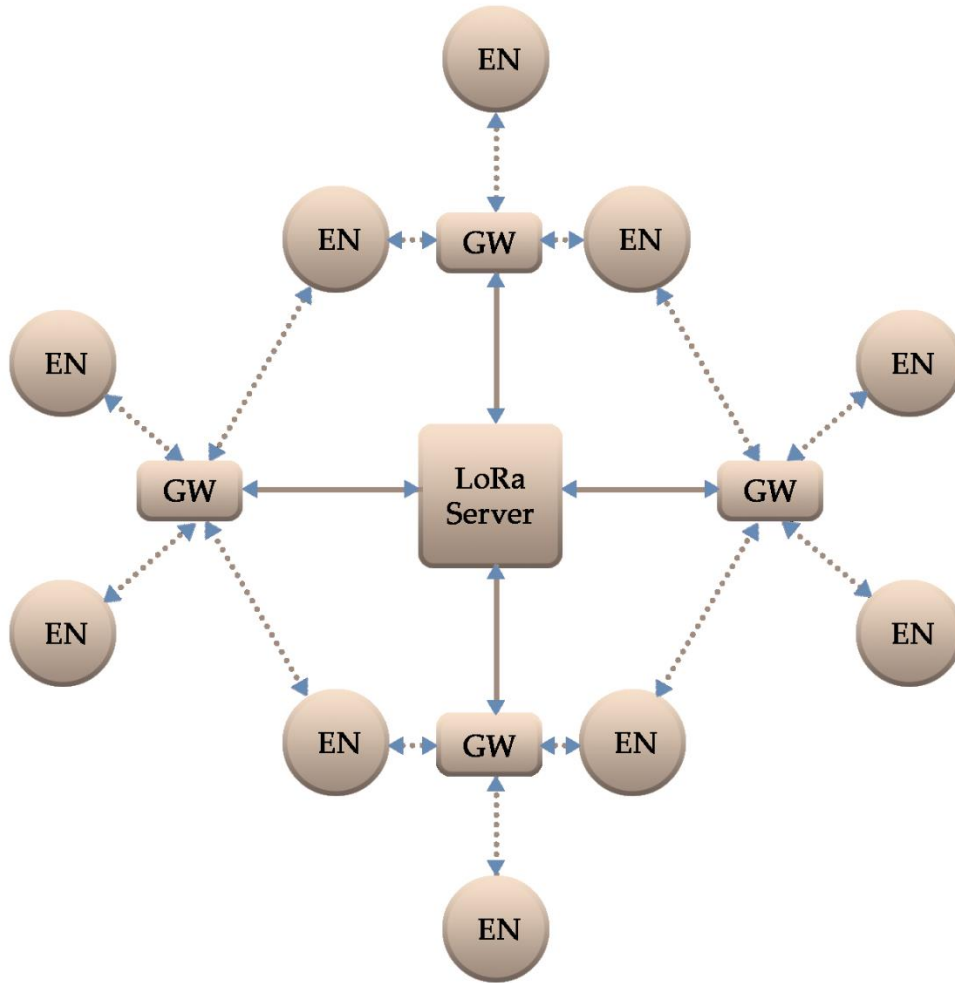
Το LoRa W.A.N. είναι ένα πρωτόκολλο δικτύου το οποίο αποσκοπεί στην βέλτιστη ενεργειακή κατανάλωση των τερματικών κόμβων που το αποτελούν (End Nodes – E.N.s). Οι E.N.s, είναι σχεδιασμένοι για να λειτουργούν με την χαμηλότερη δυνατή κατανάλωση ενέργειας και συνήθως τροφοδοτούνται από μπαταρίες. Σε σύγκριση με την τεχνολογία LoRa – PHY, το πρότυπο LoRa W.A.N. διαφέρει στο ότι παρέχει προσβασιμότητα από τον καθένα μας και έχει συνταχθεί από μία ομάδα εταιριών και ερευνητικών κέντρων, την LoRa Alliance. Παρακάτω, θα εξετασθούν εν συντομία οι βασικές πτυχές του πρωτοκόλλου, αλλά και το πως αυτές επηρεάζουν την λειτουργία του δικτύου.[4]

2.5.1 Τοπολογία των δικτύων LoRa W.A.N.

Οι πιο διαδεδομένη τοπολογία που χρησιμοποιείται σε δίκτυα LoRa W.A.N.s, είναι η τοπολογία αστέρα (Stars – of – Stars Topology – S.o.S.). Η τοπολογία αστέρα είναι μία εύκολη λειτουργικά τοπολογία λόγω του ότι είναι δυνατή η επέκταση του δικτύου με απλό τρόπο. Ως εκ τούτου, υπάρχουν περιπτώσεις δικτύων LoRa W.A.N.s που περιλαμβάνουν διαφορετική τοπολογία, όπως η τοπολογία πλέγματος (Mesh Topology), παρότι η πρόταση αυτή ανεβάζει την πολυπλοκότητα του δικτύου.[4]

Μία τυπική μορφή του δικτύου LoRa W.A.N. σε τοπολογία αστέρα, παρουσιάζεται στην εικόνα 2.10. Όπως γίνεται εύκολα αντιληπτό από την εικόνα 2.10, ένα LoRa W.A.N. δίκτυο αποτελείται από:

- Τους τερματικούς κόμβους (End Nodes – E.N.s)
- Τους κεντρικούς κόμβους (Gateways – G.W.s)
- Τον κεντρικό διακομιστή δικτύου (Network Server)



Εικόνα 2.10: Τοπολογία Αστέρα LoRa W.A.N.

2.5.1.1 Τερματικοί κόμβοι – End Nodes

Οι τερματικοί κόμβοι είναι συσκευές οι οποίες περιλαμβάνουν αισθητήρες για την συλλογή δεδομένων και μία τεχνολογία διασύνδεσης. Ένα E.N., προκειμένου να εξοικονομήσει ενέργεια και να μεγιστοποιήσει την αυτονομία του συλλέγει και αποστέλλει δεδομένα ανά χρονικά διαστήματα που ορίζονται από τις ανάγκες της εκάστοτε εφαρμογής τους, όταν δεν πραγματοποιείται η συλλογή και η αποστολή των δεδομένων τότε εισέρχονται σε κατάσταση ύπνου (sleep mode). Στην εικόνα 2.11 παρουσιάζεται ένα E.N., το οποίο περιλαμβάνει το chip SX1278 της Semtech.[5]



Εικόνα 2.11: LoRa Ra – 02 E.N.

2.5.1.2 Κεντρικοί κόμβοι – Gateways

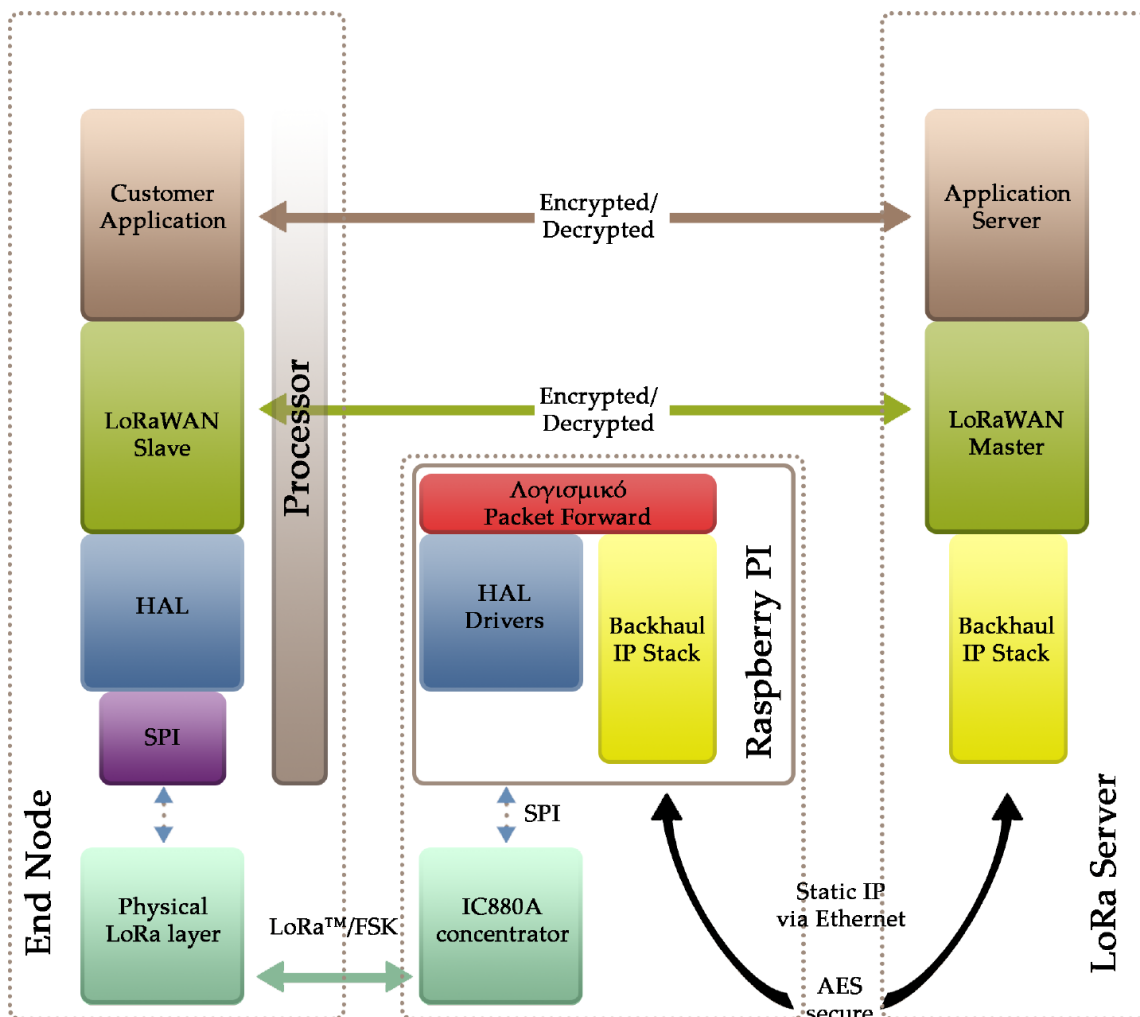
Ο σκοπός των κεντρικών κόμβων είναι η συλλογή των δεδομένων που προκύπτουν από τα E.N.s που βρίσκονται στο πεδίο της εκάστοτε εφαρμογής. Τα gateways διαθέτουν μεγάλη χωρητικότητα δικτύου, καταλαμβάνουν σταθερή θέση στον χώρο και βρίσκονται πάντα σε ενεργή κατάσταση (always listening) προκειμένου να λαμβάνουν συνέχεια πακέτα δεδομένων. Επίσης, τα gateways έχουν μεγαλύτερη επεξεργαστική ισχύ και συνήθως χρησιμοποιούν lightweight software με σκοπό την ευκολότερη παραμετροποίηση του χρήστη.[5]

2.5.1.3 Κεντρικοί διακομιστές δικτύου – Network Servers

Οι κεντρικοί διακομιστές δικτύου είναι υπολογιστές οι οποίοι συλλέγουν και επεξεργάζονται τα δεδομένα που προκύπτουν από τα E.N.s με αποτέλεσμα την εξαγωγή αποτελεσμάτων. Οι network servers συνήθως βρίσκονται σε αδιάληπτη λειτουργία με σκοπό την άμεση εξυπηρέτηση των E.N.s. Ανάλογα με τα δεδομένα που εισέρχονται για επεξεργασία μπορούν να δώσουν εντολές στα E.N.s μέσω των gateways ή να αποθηκεύσουν τα δεδομένα αυτά σε μία βάση δεδομένων με σκοπό την αξιοποίησή τους από τον χρήστη. Τέλος, ένας διακομιστής μπορεί να δέχεται πολλαπλές αφίξεις μηνυμάτων από διαφορετικά gateways.[4]

2.5.2 Πρωτόκολλο δικτύου LoRa W.A.N.

Στην εικόνα 2.12 παρουσιάζεται ένα παράδειγμα της στοίβας πρωτοκόλλου του δικτύου LoRa W.A.N., να αναφερθεί ότι παρ' όλο που τα E.N.s αλλά και οι servers περιλαμβάνουν στα υψηλότερα επίπεδά τους κάποιου είδους application layer, τα gateways απασχολούνται μόνο με την προώθηση των μηνυμάτων μεταξύ των E.N.s και των network servers, η διαδικασία αυτή υλοποιείται κυρίως σε επίπεδο firmware. Συνεπώς, υπάρχει μία στενή διασύνδεση των E.N.s και των network servers.[4]



Εικόνα 2.12: Πρωτόκολλο δικτύου LoRa W.A.N.

2.6 Κλάσεις δικτύου LoRa W.A.N.

Προκειμένου οι συσκευές E.N.s να ακολουθούν το πρωτόκολλο LoRa W.A.N., είναι απαραίτητη η ένταξή τους σε κλάσεις (classes). Αρχικά, να αναφερθεί πως όλα τα E.N.s είναι συμβατά με την κλάση A, ενώ ανάλογα με τις λειτουργικές διαδικασίες που εκτελούν εντάσσονται σε διαφορετικές κλάσεις όπως την B ή την C. Επί προσθέτου, να σημειωθεί πως η κάθε κλάση επιτρέπει την αμφίδρομη επικοινωνία μεταξύ των gateways και των E.N.s συσκευών, επίσης η επίτευξη της ενεργειακής αποδοτικότητας διαφέρει ανάλογα με την επιλογή της κλάσης που ανήκει κάθε E.N. συσκευή. Πιο συγκεκριμένα οι κλάσεις του δικτύου LoRa W.A.N. κατανέμονται ως εξής:[4]

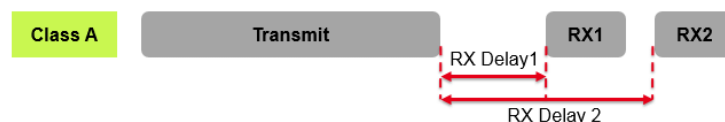
- Class A
- Class B
- Class C

2.6.1 Κλάση A (Class A)

Η κλάση A είναι προπάντων τρόπος λειτουργίας των E.N.s, στην συγκεκριμένη κατηγορία κάθε E.N. συσκευή αποστέλλει την πληροφορία με τυχαίο και ασύγχρονο τρόπο όταν αυτή υπάρχει, χρησιμοποιώντας έναν «aioha» τρόπο πρόσβασης στο κοινό κανάλι. Μετά από κάθε up – link μετάδοση, «ανοίγουν» δύο μικρά παράθυρα λήψης τα οποία έχουν χρονική διάρκεια 20ms, αναμένοντας την απόκριση από τον network server (down – link), η οποία μπορεί να είναι είτε κάποια εντολή αποδοχής του αιτήματος προκειμένου να γίνει αποστολή μηνύματος, είτε κάποιο πακέτο που θα κρίνει απαραίτητο να επιστρέψει ο network server.[4]

Το πρώτο παράθυρο «ανοίγει» στο ίδιο ακριβώς κανάλι που πραγματοποιήθηκε η up – link μετάδοση, το δεύτερο παράθυρο ανοίγει σε διαφορετικό κανάλι το οποίο επιλέγει ο network server. Η παρουσία του δεύτερου παραθύρου αυξάνει την αξιοπιστία της μετάδοσης, αντιμετωπίζοντας μ' αυτόν τον τρόπο τις διάφορες διακυμάνσεις του καναλιού. Να αναφερθεί επίσης, πως δεν υπάρχει κάποιος μηχανισμός αποστολής της πληροφορίας αν δεν πραγματοποιηθεί το down – link, δηλαδή αν δεν επιτύχει η αμφίδρομη επικοινωνία σε κάποιο από τα υπάρχοντα κανάλια λήψεως.[4]

Τέλος, η κλάση A αποτελεί την πιο ενεργειακά αποδοτική κλάση, αυτό σημαίνει πως κάθε συσκευή E.N. θα πρέπει να καταναλώνει την ελάχιστη δυνατή ενέργεια. Συνεπώς, πρέπει να διατηρεί ανενεργό τον δέκτη για όσο τον δυνατόν μεγαλύτερο χρονικό διάστημα, προκειμένου αυτός να μην καταναλώνει ενέργεια, αυτό μπορεί να επιτευχθεί είτε μπαίνοντας σε κατάσταση ύπνου (sleep mode), είτε σε κάποια κατάσταση απενεργοποίησης (off mode).[4]

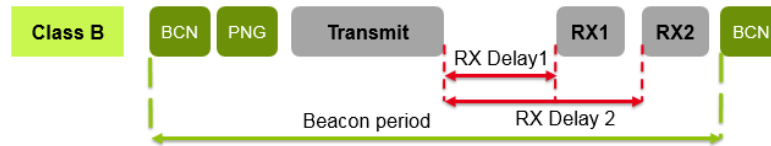


Εικόνα 2.13: Κλάση A αποστολή και λήψη δεδομένων

2.6.2 Κλάση B (Class B)

Στην κλάση B, οι συσκευές E.N.s «ανοίγουν» επιπλέον παράθυρα λήψης τα οποία έχουν προκαθορισμένες χρονικές στιγμές, σε αντίθεση με την κλάση A η οποία όπως αναφέρθηκε τα παράθυρα λήψης της είναι τυχαία. Για τον προκαθορισμό των χρονικών στιγμών των παραθύρων λήψεως, οι E.N.s συσκευές συγχρονίζονται με τον network server. Ο network server, στέλνει μέσω των gateways πιλοτικά σήματα (beacons), με σκοπό την αμφίδρομη επικοινωνία και το «άνοιγμα» των παραθύρων λήψεως.[4]

Να σημειωθεί ότι ο μηχανισμός του beacon λειτουργεί ανεξάρτητα από το up – link, το γεγονός αυτό καθιστά τις συσκευές που εντάσσονται σ’ αυτήν την κλάση ιδανικές για λειτουργίες ελέγχου, όπως για παράδειγμα είναι οι ενεργοποιητές (actuators) ή οι διακόπτες (switches). Τέλος, η ενεργειακή κατανάλωση των E.N.s που χρησιμοποιούνται σ’ αυτήν την κατηγορία είναι αυξημένη σε σχέση με τις συσκευές που χρησιμοποιούνται στην κλάση A.[4]

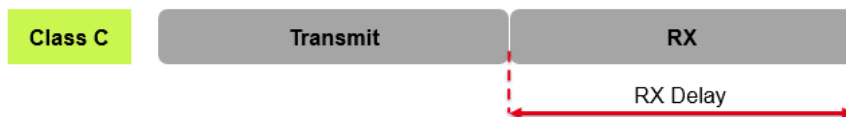


Εικόνα 2.14: Κλάση B αποστολή και λήψη δεδομένων

2.6.3 Κλάση C (Class C)

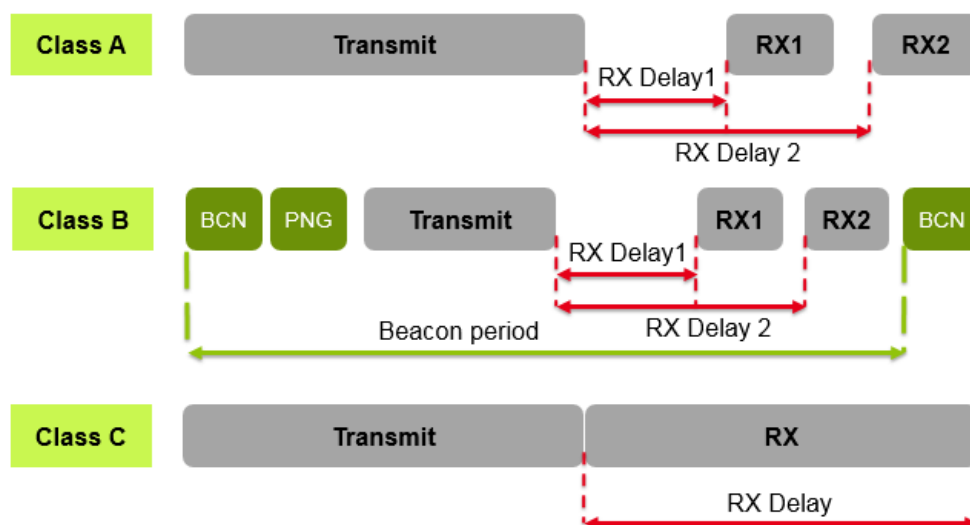
Στην κλάση C, οι συσκευές E.N.s βρίσκονται σε συνεχόμενη επαγρύπνηση προκειμένου να δεχτούν κάποιο πακέτο από τον network server μέσω των gateways, η διαδικασία αυτή όμως δεν μπορεί να πραγματοποιηθεί όταν μεταδίδουν κάποια μορφή πακέτου. Συνεπώς, οι E.N.s συσκευές που χρησιμοποιούνται στην κλάση αυτή δεν έχουν αυστηρούς περιορισμούς που αφορούν την ενεργειακή κατανάλωσή τους, καθώς δεν μπορούν να μεταβούν σε κάποιου είδους κατάσταση ύπνου (sleep mode).[4]

Συνήθως, διαθέτουν σχεδόν πάντα «ανοιχτά» παράθυρα λήψης, το γεγονός αυτό συμβάλλει στην μικρότερη χρονική καθυστέρηση (latency) της επικοινωνίας μεταξύ του network server και των E.N.s συσκευών. Αυτού του είδους οι E.N.s χρησιμοποιούνται σε εφαρμογές που απαιτούν μικρό latency και η ενεργειακή τους κατανάλωση δεν αποτελεί μείζων θέμα.[4]



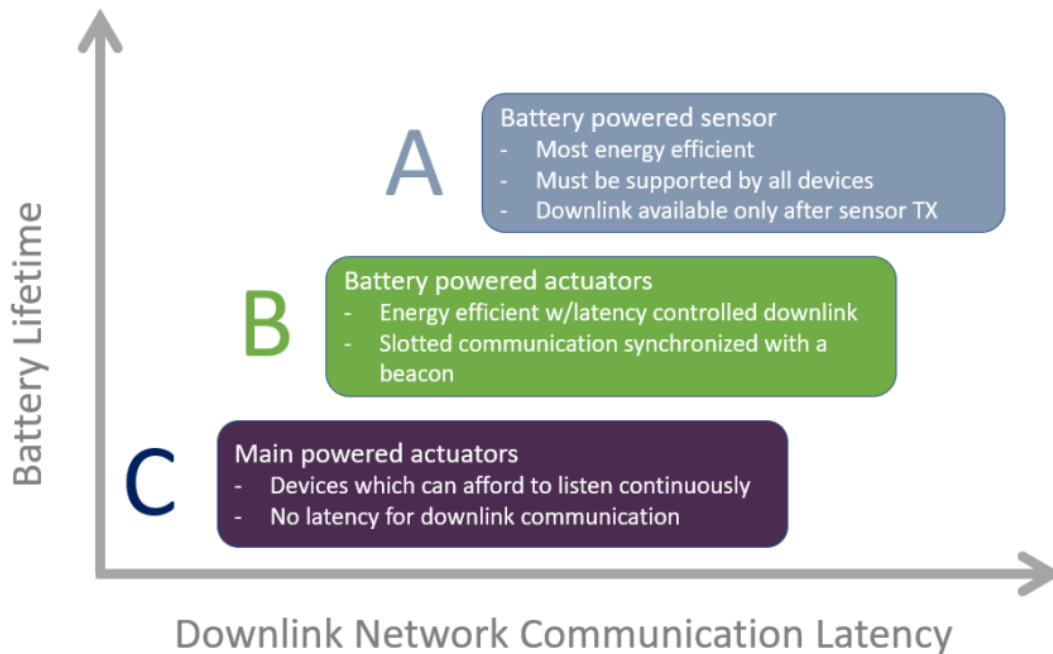
Εικόνα 2.15: Κλάση C αποστολή και λήψη δεδομένων

Συνοψίζοντας, η εικόνα 2.16 παρουσιάζει τις κλάσεις A, B και C και πώς αυτές λαμβάνουν και αποστέλλουν δεδομένα.



Εικόνα 2.16: Συνοπτική παρουσίαση λήψης και αποστολής δεδομένων των κλάσεων

Με την ανάλυση των παραπάνω γίνεται κατανοητό ότι ανάλογα σε ποια κλάση βρίσκεται μία E.N. συσκευή παρουσιάζει πλεονεκτήματα και μειονεκτήματα. Τα χαρακτηριστικά αυτά σε σχέση πάντα με τις απαιτήσεις της εκάστοτε εφαρμογής έχουν ως αποτέλεσμα τον ορισμό της κλάσης που θα βρίσκεται μία E.N. συσκευή. Η εικόνα 2.17 παρουσιάζει συγκριτικό διάγραμμα της ενεργειακής κατανάλωσης συναρτήσει της χρονικής καθυστέρησης που υπάρχει στο down – link.



Εικόνα 2.17: Συγκριτικό διάγραμμα κλάσεων

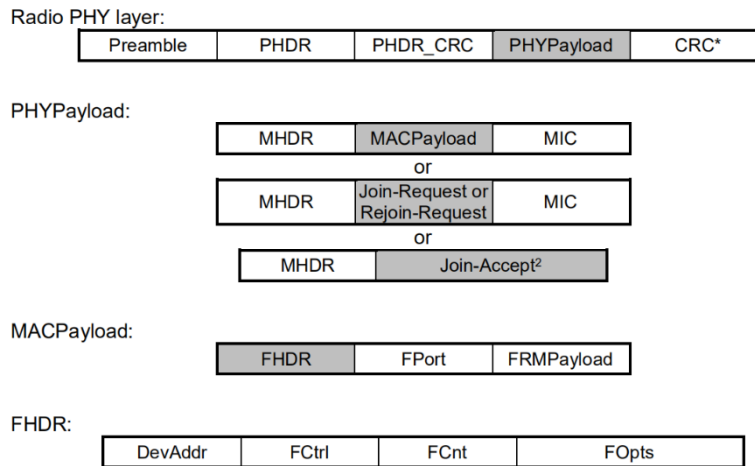
Αρχικά, αυτό που συμπεραίνετε είναι πως, η κλάση A διαθέτει την χαμηλότερη ενεργειακή κατανάλωση σε σύγκριση με τις κλάσεις B και C, αλλά υστερεί στην χρονική απόκριση έναντι της κλάσης C. Η κλάση B, παρουσιάζει μία ισορροπία μεταξύ της ενεργειακής κατανάλωσης και της χρονικής απόκρισης σε σύγκριση με την κλάση A και C. Τέλος, η κλάση C διαθέτει την χαμηλότερη χρονική απόκριση αλλά την υψηλότερη ενεργειακή κατανάλωση σε συγκρίνοντάς την με τις κλάσεις A και B.

2.7 Δομή πακέτου και MAC εντολές δικτύου LoRa W.A.N.

Οι MAC εντολές καταλαμβάνουν μεγάλο μέρος του πρωτοκόλλου LoRa W.A.N., η δομή ενός πακέτου δημιουργείται από την μορφή του PHY layer που έχει αναφερθεί στην υπό – ενότητα 2.4 και ένα σύνολο από παραμέτρους που επιβάλλει το δίκτυο LoRa W.A.N., αυτό του είδους οι παράμετροι αποτελούν τις MAC εντολές. Οι πληροφορίες που περιλαμβάνονται στις MAC εντολές αφορούν τον τύπο του μηνύματος, αλλά και την ίδια E.N. συσκευή, ενώ οι περισσότερες μεταβλητές αποτελούν τις παραμέτρους του συστήματος.

Η επεξεργασία καθώς επίσης και πολλές αλλαγές των παραμέτρων MAC γίνεται στον network server. Στην εικόνα 2.18 φαίνονται αυτές οι αλλαγές, το μέρος του πακέτου που περιλαμβάνει το MAC payload αποτελείται από τα εξής υπό – στρώματα (subframes):

- Frame header (FHDR)
- Frame port (FPort)
- Frame payload (FRMPayload)



Εικόνα 2.18: Τυπική μορφή πακέτου LoRa

Όπως φαίνεται στην παραπάνω εικόνα, το frame header (FHDR) αποτελείται από τέσσερα πεδία τα οποία σχετίζονται με διαφορετικές πτυχές του LoRa W.A.N.. Πιο συγκεκριμένα, υπάρχει ένα subframe με την ονομασία DevAddr το οποίο σημαίνει device address και αποτελείται από μία αλληλουχία τεσσάρων bytes, το DevAddr είναι υπεύθυνο για την αναγνώριση του network server στον οποίο ανήκει η E.N. συσκευή. Έπειτα, υπάρχει το subframe FCtrl το οποίο σημαίνει frame control και αποτελείται από ένα byte, το subframe FCnt το οποίο σημαίνει frame counter και αποτελείται από δύο byte και τέλος ακολουθεί το subframe FOpts το οποίο σημαίνει frame options και αποτελείται από δεκαέξι bytes, τα οποία περιλαμβάνουν διάφορες MAC εντολές, οι οποίες ανταλλάσσονται μεταξύ του network server και των E.N.s συσκευών.[4]

Η χρησιμοποίηση του frame port (FPort), είναι προαιρετική και γίνεται αποκλειστικά για την εφαρμογή στην οποία περιορίζεται το μήνυμα. Τέλος, το frame payload (FRMPayload) περιλαμβάνει δεδομένα τα οποία προέρχονται από το Application Layer. Γενικά, οι MAC εντολές που ανταλλάσσονται μεταξύ E.N.s συσκευών και του network server αφορούν διάφορες λειτουργίες του LoRa W.A.N., οι πιο σημαντικές λειτουργίες έχουν σχέση:

- Με τον προσδιορισμό της απόστασης μεταξύ του gateway και της E.N. συσκευής, δηλαδή η κάθε E.N. συσκευή είναι σε θέση να ζητήσει από τον network server την λαμβανόμενη ισχύ του gateway (Link Margin)
- Με τα δεδομένα της εφαρμογής αλγορίθμων βελτιστοποίησης, όπως για παράδειγμα είναι ο ADR αλγόριθμος (Adaptive Data Rate Algorithm)
- Με τους περιορισμούς που μπορεί να θέσει ο network server και αφορούν το duty cycle το οποίο όπως έχει αναφερθεί στην υπό – ενότητα 2.3.1 παίζει πολύ σημαντικό ρόλο
- Με τις ρυθμίσεις των παραμέτρων που αφορούν τα παράθυρα λήψεων
- Με τα δεδομένα τα οποία παρέχουν πληροφορίες για την κατάσταση της E.N. συσκευής σε πραγματικό χρόνο όπως είναι για παράδειγμα η κατάσταση της μπαταρίας και το demodulation margin
- Με τυχόν δημιουργία νέων καναλιών επικοινωνίας, που έχουν σαν σκοπό την αύξηση της αξιοπιστίας της προς μετάδοσης πληροφορίας

Η σχέση μεταξύ των MAC εντολών και του network server είναι στενή, επίσης να σημειωθεί πως οποιαδήποτε εντολή υπάρχει στο firmware και στο software της E.N. συσκευής προϋποθέτει την παρουσία αντίστοιχων εντολών στον network server προκειμένου η λειτουργία τους να είναι σωστή.[4]

2.8 Επίλογος

Ο σκοπός αυτού του κεφαλαίου είναι η εισαγωγή στην δομή και την στοίβα των πρωτοκόλλων που χρησιμοποιούνται τα δίκτυα μεγάλης εμβέλειας LoRa. Τα οποία αποτελούνται από 4 επίπεδα τα οποία είναι το επίπεδο εφαρμογής (LoRa Application Layer), το επίπεδο ελέγχου προσπέλασης μέσω (LoRa M.A.C. Layer), το φυσικό επίπεδο (LoRa PHY Layer) και το επίπεδο ραδιοσυχνότητας (LoRa RF Layer).

Στην Ευρώπη τα δίκτυα LoRa χρησιμοποιούν τις I.S.M. ζώνες συχνοτήτων, αυτό έχει ως αποτέλεσμα να λειτουργούν στις μπάντες συχνοτήτων 867 – 869 MHz, ενώ σε ορισμένες χώρες που ανήκουν στην ευρωπαϊκή ένωση επιτρέπεται η χρησιμοποίηση της ζώνης συχνοτήτων των 433MHz, σ' αυτές τις χώρες συμπεριλαμβάνεται και η Ελλάδα. Ανάλογα με την ζώνη συχνοτήτων που λειτουργεί ένα δίκτυο LoRa επιβάλλονται ορισμένοι περιορισμοί που θα πρέπει να λαμβάνονται υπ' όψη κατά τον σχεδιασμό των δικτύων αυτών.

Δεν υπάρχει σαφής περιγραφή της τεχνικής διαμόρφωσης που χρησιμοποιείται στην LoRa επικοινωνία, παρά μόνο ορισμένα paper που δεν εξηγούν αναλυτικά περί τίνος πρόκειται. Ωστόσο, σύμφωνα με έρευνα που εκπόνησε ο Matthew Knight προκύπτει ότι η διαμόρφωση LoRa βασίζεται στην τεχνική C.S.S., χρησιμοποιώντας chirp σήματα. Ο παράγοντας διασποράς αποτελεί ένα πολύ βασικό μέγεθος σ' αυτού του είδους τα συστήματα, στην ουσία η τιμή του προσδιορίζει τον ρυθμό των κωδικοποιημένων bits ανά σύμβολο και έχει εύρος τιμών από 7 ως 12. Επίσης, η αύξηση του παράγοντα διασποράς οδηγεί σε αύξηση της ευαισθησίας του δέκτη μεγάλωνοντας μ' αυτόν τον τρόπο την ακτίνα κάλυψης του συστήματος. Ωστόσο η επιλογή μεγάλης τιμής του παράγοντα διασποράς μεγάλώνει την πιθανότητα μετάδοσης λάθος πληροφορίας. Τέλος, η τιμή του παράγοντα διασποράς πρέπει να είναι γνωστή τόσο στον πομπό, όσο και στον δέκτη.

Οι τεχνικές κωδικοποίησης εξασφαλίζουν την μετάδοση της πληροφορίας προκειμένου να επιτευχθεί η αξιόπιστη μεταφορά τους, επιγραμματικά οι τεχνικές που χρησιμοποιούνται στην διαμόρφωση LoRa είναι η Data Whitening, η Gray Indexing, η Forward Error Correction και η Interleaving.

Ένα πακέτο LoRa περιλαμβάνει δύο μορφές, την μορφή explicit και την μορφή implicit. Η βασική διαφορά των δύο μορφών, είναι ότι στην μορφή explicit το πακέτο περιλαμβάνει επί πρόσθετα μία κεφαλίδα (header) η οποία συμπεριλαμβάνει πληροφορίες όπως τον αριθμό των bytes, το coding rate και την ύπαρξη ή όχι ενός δεκαεξάμπτου κυκλικού ελέγχου απόρριψης για το Payload. Στην μορφή implicit στον δέκτη, είναι γνωστά όλα τα παραπάνω επομένως η χρησιμοποίηση της κεφαλίδας δεν είναι απαραίτητη γι' αυτό αφαιρείται.

Το LoRa W.A.N. είναι ένα πρωτόκολλο δικτύου το οποίο αποσκοπεί στην βέλτιστη ενεργειακή κατανάλωση των τερματικών κόμβων που το αποτελούν. Οι πιο διαδεδομένη τοπολογία που χρησιμοποιείται σε δίκτυα LoRa W.A.N.s, είναι η τοπολογία αστέρα. Ωστόσο, υπάρχουν περιπτώσεις δικτύων LoRa W.A.N.s που περιλαμβάνουν διαφορετική τοπολογία, όπως η τοπολογία πλέγματος παρ' ότι αυξάνει την πολυπλοκότητα του συστήματος. Ένα LoRa W.A.N. δίκτυο αποτελείται από τους τερματικούς κόμβους, τους κεντρικούς κόμβους και τον κεντρικό διακομιστή δικτύου.

Τέλος, προκειμένου οι τερματικές συσκευές να ακολουθούν το πρωτόκολλο LoRa W.A.N., είναι απαραίτητη η ένταξή τους σε κλάσεις. Όλες οι τερματικές συσκευές είναι συμβατές με την κλάση A, ενώ ανάλογα με τις λειτουργικές διαδικασίες που εκτελούν εντάσσονται σε διαφορετικές κλάσεις όπως την B ή την C. Η κάθε κλάση επιτρέπει την αμφίδρομη επικοινωνία μεταξύ των gateways και των τερματικών συσκευών.

Κεφάλαιο 3: Δομή και υλοποίηση συστήματος

3.1 Εισαγωγή

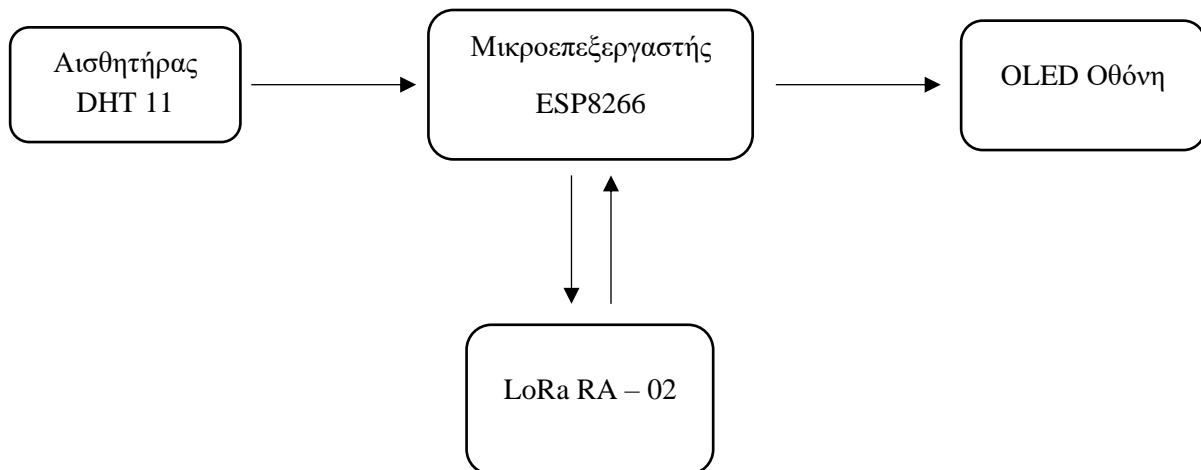
Σ' αυτό το κεφάλαιο αναλύεται η δομή, η υλοποίηση, η κατασκευή και η λειτουργία των μονάδων που αναπτύχθηκαν. Αρχικά, δίνονται τα μπλοκ διαγράμματα των δύο συσκευών, έπειτα παρουσιάζονται οι συνδεσμολογίες που σχεδιάστηκαν και υλοποιήθηκαν κατά το πειραματικό και κατασκευαστικό μέρος της εργασίας. Στην συνέχεια, περιγράφονται οι διαδικασίες που ακολουθήθηκαν κατά την κατασκευή των μονάδων. Επίσης, περιγράφεται ο κώδικας του προγράμματος που περιλαμβάνουν οι μονάδες του συστήματος προκειμένου να γίνεται η μέτρηση, η αποστολή, η λήψη και η καταχώρηση των μετρούμενων φυσικών μεγεθών σε απόσταση 2 έως 3 χιλιομέτρων. Τέλος, καταλήγει στις μετρήσεις που πραγματοποιήθηκαν υπό πραγματικές συνθήκες και παρουσιάζει τα αποτελέσματα των μετρήσεων που καταχωρήθηκαν τόσο σε τοπικό server, όσο και στην πλατφόρμα του ThingSpeak.

3.2 Δομή συστήματος

Η βασική δομή του συστήματος, αποτελείται από δύο συσκευές, την συσκευή μετρήσεων και εκπομπής δεδομένων και την συσκευή λήψης και καταχώρησης των δεδομένων σε τοπικό server και στην πλατφόρμα ThingSpeak.

3.2.1 Συσκευή μετρήσεων και εκπομπής δεδομένων

Στην εικόνα 3.1, παρουσιάζεται το μπλοκ διάγραμμα της διάταξης που ως σκοπό έχει την μέτρηση ορισμένων μεγεθών όπως η θερμοκρασία και την αποστολή των δεδομένων μέτρησης που προέκυψαν σε απόσταση από 2 έως 3 χιλιομέτρων.



Εικόνα 3.1: Μπλοκ διάγραμμα συσκευής μετρήσεων και εκπομπής δεδομένων

Όπως φαίνεται και στην παραπάνω εικόνα η συσκευή μέτρησης και εκπομπής δεδομένων περιλαμβάνει έναν αισθητήρα θερμοκρασίας και υγρασίας (DHT11), έναν μικροεπεξεργαστή (ESP8266), μία OLED οθόνη και τον πομπό – δέκτη LoRa. Τα τεχνικά χαρακτηριστικά των μονάδων που περιλαμβάνονται στην συσκευή αναλύονται στο παράρτημα Α.

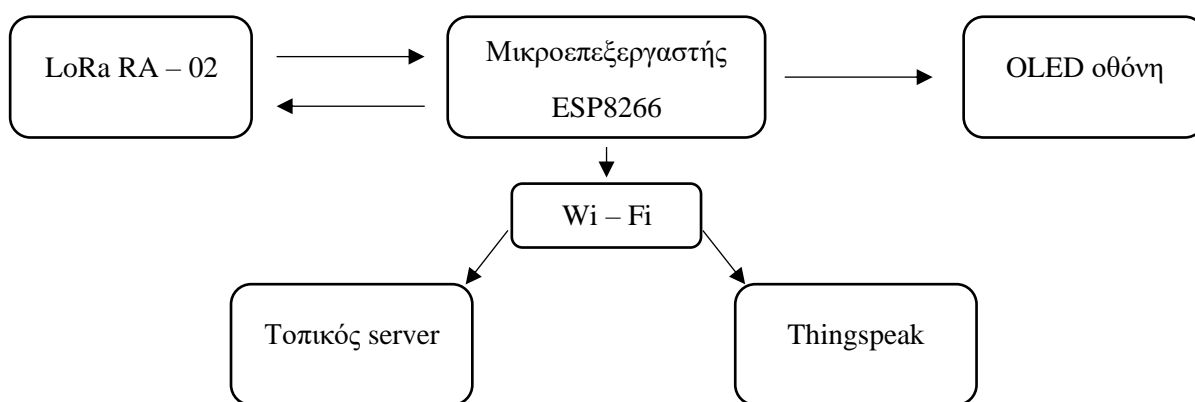
Σύμφωνα με όλα όσα αναφέρθηκαν στο 2^ο κεφάλαιο, η υλοποίηση παρόμοιων συστημάτων εστιάζει στην μεγάλη εμβέλεια αλλά και στην χαμηλότερη ενεργειακή κατανάλωση. Συνεπώς, η επιλογή των μονάδων που αποτελούν την συσκευή μέτρησης έγινε λαμβάνοντας υπ' όψη αυτές τις δύο βασικές παραμέτρους εξασφαλίζοντας παράλληλα το χαρακτηριστικό της φορητότητας.

Ο τρόπος λειτουργίας της μονάδας είναι σχετικά απλός, όμως υπόκειται σε διάφορους περιορισμούς που θα εξηγηθούν στην συνέχεια της εργασίας. Αρχικά, ο αισθητήρας θερμοκρασίας μετράει την θερμοκρασία και την υγρασία στον περιβάλλοντα χώρο που βρίσκεται η συσκευή και στην συνέχεια στέλνει τα δεδομένα που προκύπτουν στον μικροεπεξεργαστή.

Ο μικροεπεξεργαστής με την σειρά του εφόσον λάβει και επεξεργαστεί τα δεδομένα τα αναπαράγει στην OLED οθόνη προκειμένου να υπάρχει τοπική ένδειξη των μετρήσεων. Ταυτόχρονα με την αναπαραγωγή των δεδομένων στην οθόνη, τα δεδομένα αποστέλλονται μέσω του πομπό – δέκτη στην συσκευή λήψης η οποία βρίσκεται σε ακτίνα από 2 έως 3 χιλιόμετρα μακριά.

3.2.2 Συσκευή λήψης και καταχώρησης δεδομένων στο cloud

Στην εικόνα 3.2, απεικονίζεται το μπλοκ διάγραμμα της διάταξης η οποία έχει σαν σκοπό την λήψη των δεδομένων και την καταχώρηση τους σε ιδιωτικό τοπικό server, αλλά και σε δημόσια πλατφόρμα όπως το ThingSpeak.



Εικόνα 3.2: Μπλοκ διάγραμμα συσκευής λήψης και καταχώρησης δεδομένων σε cloud

Τα τεχνικά χαρακτηριστικά των μονάδων που περιλαμβάνονται στην συσκευή αυτή υπάρχουν στο παράρτημα Α. Όπως παρουσιάζεται στην εικόνα 3.2, η συσκευή λήψης και καταχώρησης δεδομένων αποτελείται από έναν πομπό – δέκτη LoRa, έναν μικροεπεξεργαστή (ESP8266) και μία OLED οθόνη.

Επίσης, για την καταχώρηση των δεδομένων είτε στον server, είτε στο ThingSpeak είναι απαραίτητη η σύνδεση του μικροεπεξεργαστή με το ίντερνετ, αυτό μπορεί να επιτευχθεί χάρη στην ενσωμάτωση του Wi – Fi που περιλαμβάνει ο ίδιος ο μικροεπεξεργαστής από τον κατασκευαστή του.

Ο τρόπος λειτουργίας της συσκευής είναι ο εξής, εφόσον ο πομπό – δέκτης λάβει τα δεδομένα τα στέλνει στον μικροεπεξεργαστή, ο οποίος με την σειρά του τα επεξεργάζεται και μέσω Wi – Fi σύνδεσης τα καταχωρεί τοπικά και στο ThingSpeak.

3.3 Κατασκευή συσκευής μέτρησης και εκπομπής δεδομένων

Η διαδικασία της υλοποίησης της συσκευής μέτρησης και εκπομπής δεδομένων περιελάμβανε τρία στάδια, την σχεδίαση της συνδεσμολογίας του συστήματος, το πειραματικό στάδιο και το κατασκευαστικό στάδιο, τα οποία αναλύονται στις παρακάτω υπό – ενότητες.

3.3.1 Συνδεσμολογία συστήματος μέτρησης και εκπομπής δεδομένων

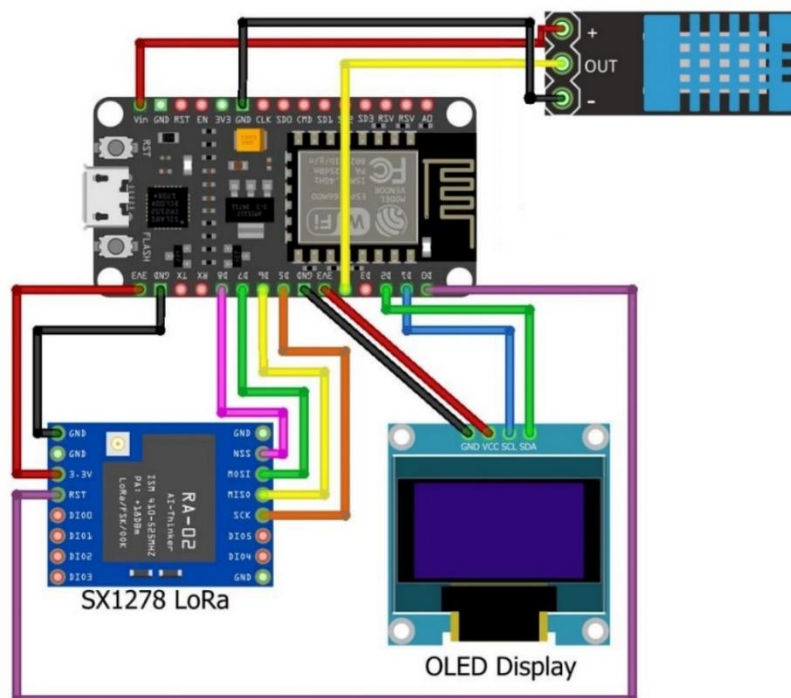
Η κατασκευή της συσκευής, ακολουθεί την συνδεσμολογία του μικροεπεξεργαστή με τις μονάδες όπως απεικονίζεται στην εικόνα 3.3. Για να καταστεί σαφές ποια pins του μικροεπεξεργαστή συνδέονται με τις μονάδες του συστήματος δημιουργήθηκε ο πίνακας 3.1.

Πιο συγκεκριμένα, τα pins D0, D5, D6, D7 και D8 είναι συνδεδεμένα με τον πομπό – δέκτη LoRa, τα pins D1 και D2 συνδέονται στην OLED οθόνη, ενώ το pin D4 συνδέεται στην έξοδο του αισθητήρα. Επίσης, να αναφερθεί πως για την γνωστοποίηση της διαδικασίας της αποστολής των δεδομένων μεταξύ των συσκευών εκπομπής και λήψης δεδομένων τοποθετήθηκε ενδεικτικό LED στο pin D3, το οποίο αναβοσβήνει υποδηλώνοντας ότι η συσκευή μέτρησης εξέπεμψε το πακέτο που περιλαμβάνει τα δεδομένα της μέτρησης προς την συσκευής λήψης. Αν η συσκευή μέτρησης και εκπομπής δεδομένων ξεπεράσει τα όρια της σύνδεσης με αποτέλεσμα να χαθεί η ζεύξη το LED συνεχίζει να αναβοσβήνει υποδηλώνοντας την εκπομπή πακέτων και συσκευή συνεχίζει απρόσκοπτα την λειτουργία της.

Τέλος, προκειμένου η συσκευή να είναι φορητή περιλαμβάνει δύο επαναφορτιζόμενες μπαταρίες τύπου 18650 των 3,7V – 3000mAh και έναν dc – dc buck μετατροπέα με ρυθμισμένη τάση εξόδου στα 5V με την οποία τροφοδοτείται όλη η συσκευή. Επί προσθέτου, η έξοδος του μετατροπέα μπορεί να δώσει μέγιστο ρεύμα 2A το οποίο καλύπτει πλήρως την συνολική κατανάλωση της συσκευής.

Πίνακας 3.1: Πίνακας συνδεσμολογίας συσκευής μέτρησης και εκπομπής

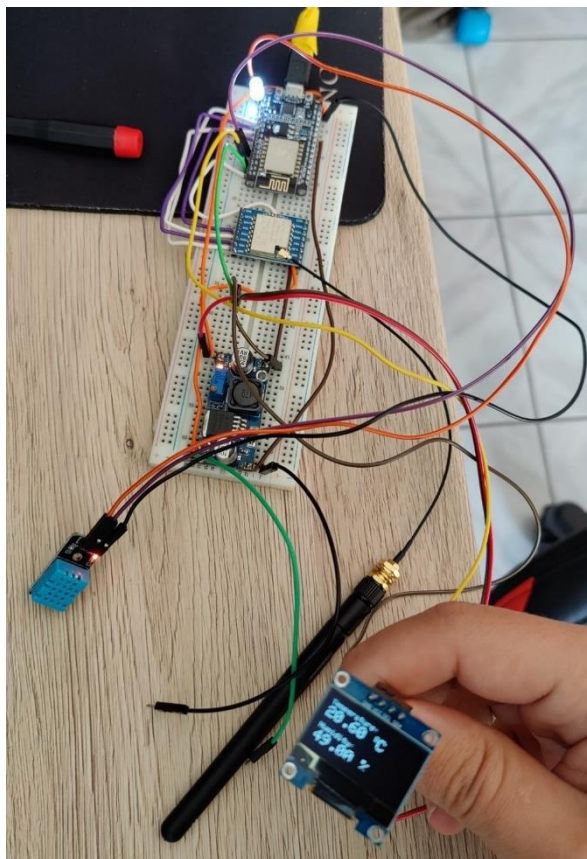
ESP8266	LoRa Ra - 02	OLED	DHT 11	LED
Pins				
D0	RST			
D1		SCL		
D2		SDA		
D3				+
D4			OUT	
D5	SCK			
D6	MISO			
D7	MOSI			
D8	NSS			
VIN			+	
3V3	3V3	VCC		
GND	GND	GND	-	-



Εικόνα 3.3: Συνδεσμολογία συσκευής μέτρησης και εκπομπής δεδομένων

3.3.2 Πειραματική υλοποίηση συστήματος μέτρησης και εκπομπής δεδομένων

Στο πειραματικό στάδιο, έγινε η υλοποίηση του συστήματος στο ράστερ, σύμφωνα με την συνδεσμολογία της εικόνας 3.3. Στην εικόνα 3.4 παρουσιάζεται η πειραματική διάταξη του συστήματος στο ράστερ.

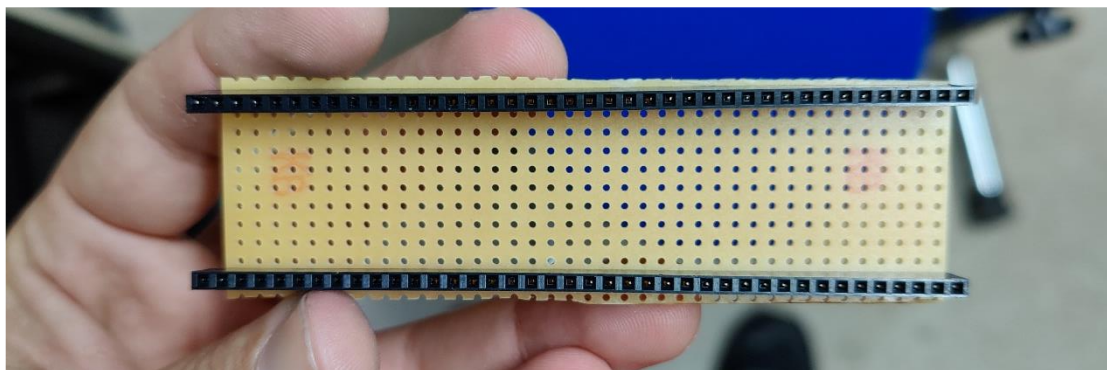


Εικόνα 3.4: Συσκευή μέτρησης και εκπομπής δεδομένων σε πειραματικό στάδιο

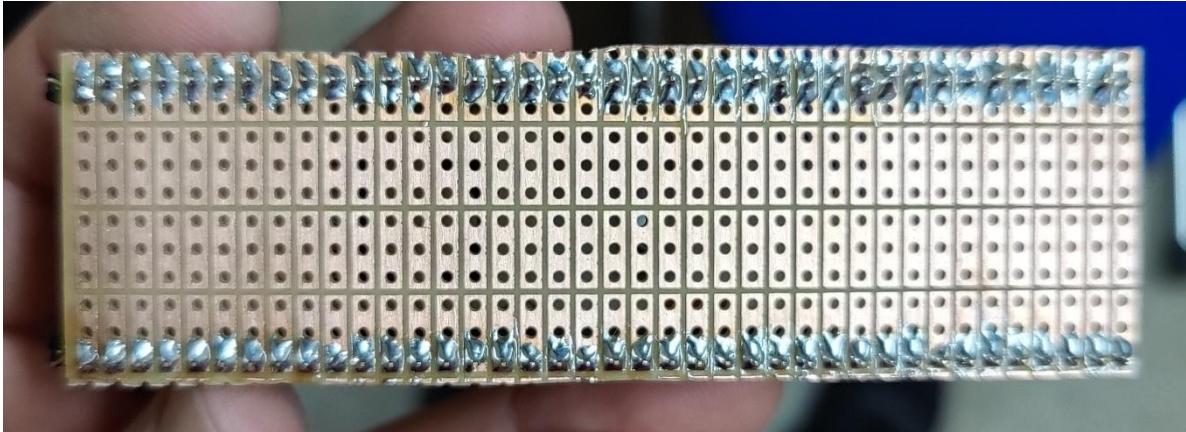
Εφόσον η λειτουργία της συσκευής στο στάδιο αυτό ήταν ομαλή και τα αποτελέσματα τα αναμενόμενα, ακολουθεί η κατασκευαστική υλοποίηση της συσκευής.

3.3.3 Κατασκευαστική υλοποίηση συστήματος μέτρησης και εκπομπής δεδομένων

Για την υλοποίηση της κατασκευής του συστήματος μέτρησης και εκπομπής δεδομένων χρησιμοποιήθηκε μία διάτρητη πλακέτα, στις άκρες της οποίας κολλήθηκαν δύο θηλυκά pin header των 40 θέσεων, όπως φαίνεται στις εικόνες 3.5 και 3.6.

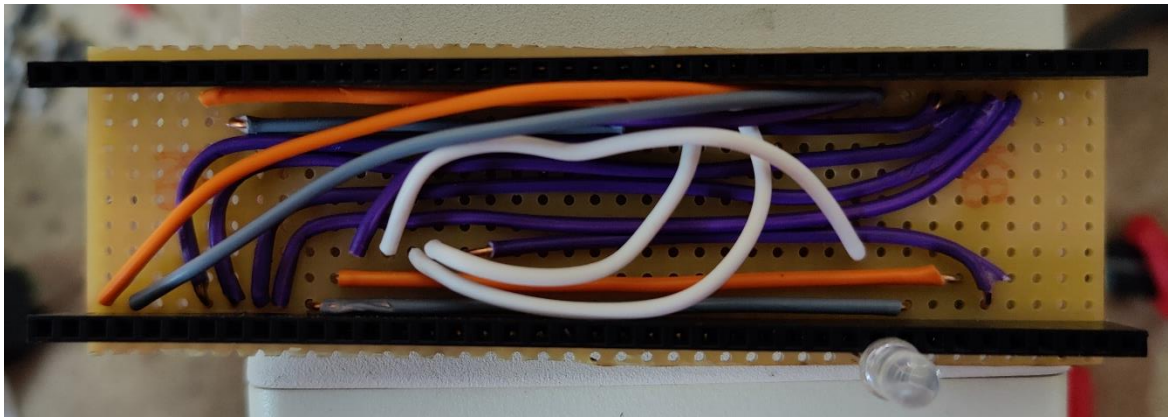


Εικόνα 3.5: Πάνω όψη διάτρητης πλακέτας

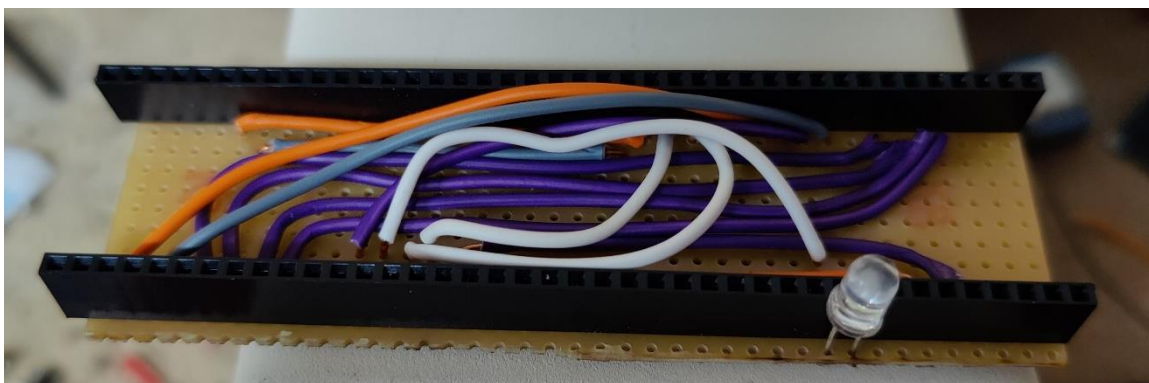


Εικόνα 3.6: Κάτω όψη διάτρητης πλακέτας

Στην συνέχεια, πραγματοποιήθηκε η διαδικασία της συνδεσμολογίας μεταξύ των pin header σύμφωνα με την συνδεσμολογία του συστήματος προκειμένου να υπάρχει επικοινωνία των μονάδων που αποτελούν το σύστημα μέτρησης και εκπομπής δεδομένων. Στις εικόνες 3.7 και 3.8 απεικονίζεται το τελικό αποτέλεσμα αυτής της διαδικασίας.



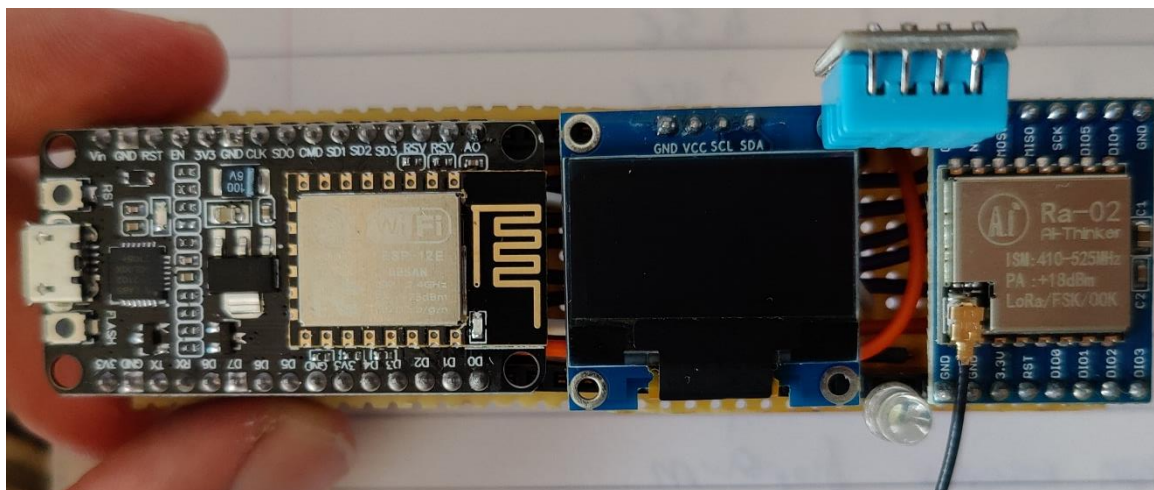
Εικόνα 3.7: Συνδεσμολογία των pins header (πάνω όψη)



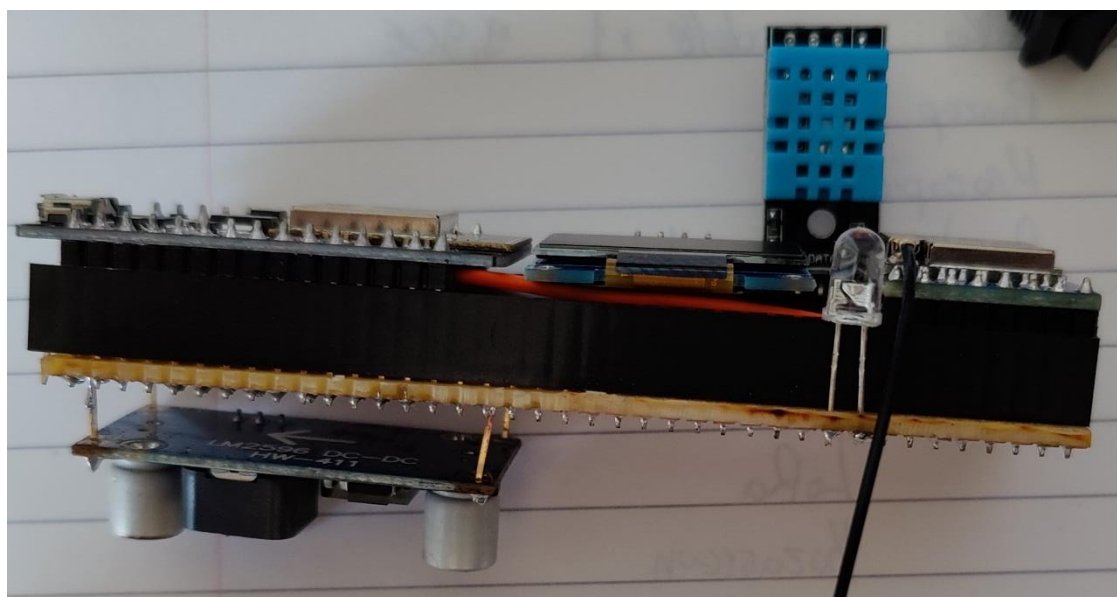
Εικόνα 3.8: Τοποθέτηση ενδεικτικού led

Έπειτα από την ολοκλήρωση της διαδικασίας της συνδεσμολογίας των pins header ακολούθησε έλεγχος συνέχειας (continuity test) σύμφωνα με την συνδεσμολογία του συστήματος (εικόνα 3.3) προκειμένου να διαπιστωθεί ότι όλες η συνδέσεις είναι σύμφωνες με την συνδεσμολογία που σχεδιάστηκε.

Τέλος, στις αντίστοιχες θέσεις των θηλυκών pins headers, τοποθετήθηκαν οι μονάδες που περιλαμβάνει το σύστημα μέτρησης και εκπομπής δεδομένων και είναι το ESP8266, το LoRa module, η OLED οθόνη, ο αισθητήρας DHT 11 και ο dc – dc buck converter. Στην εικόνα 3.9 απεικονίζεται η τελική μορφή της κατασκευής του συστήματος (πάνω όψη), ενώ στην εικόνα 3.10 παρουσιάζεται η τελική μορφή της κατασκευής του συστήματος (πλάγια όψη) στην οποία απεικονίζεται ο dc – dc buck converter.



Εικόνα 3.9: Τελική μορφή κατασκευής συστήματος μέτρησης και εκπομπής δεδομένων



Εικόνα 3.10: Τελική μορφή κατασκευής dc - dc buck converter

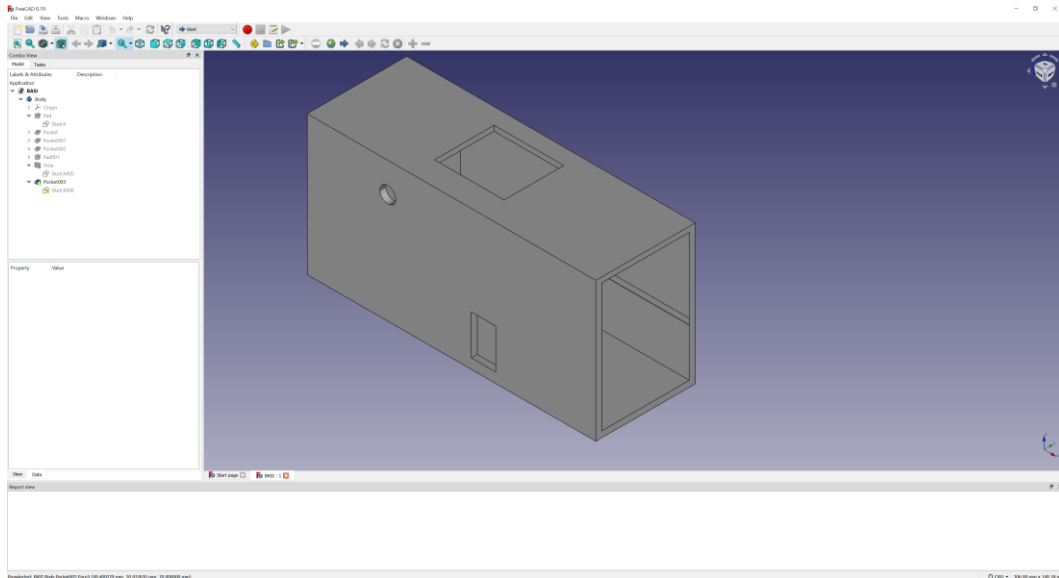
Η συσκευή μέτρησης και εκπομπής δεδομένων πρέπει να είναι φορητή, για τον σκοπό αυτό επί προσθέτου τοποθετήθηκαν μία μπαταριοθήκη δύο θέσεων τύπου 18650, που φέρει δύο μπαταρίες των 3,7V – 3000mAh, καθώς και έναν on – off διακόπτη.

3.3.4 Σχεδίαση και υλοποίηση 3D πλαισίου

Κλείνοντας, είναι απαραίτητη η κατασκευή ενός πλαισίου μέσα στο οποίο θα τοποθετηθεί η φορητή κατασκευή του συστήματος μέτρησης και εκπομπής δεδομένων. Για τον σκοπό αυτό, χρησιμοποιήθηκε το λογισμικό σχεδίασης 3D μοντέλων freeCAD.

Κεφάλαιο 3

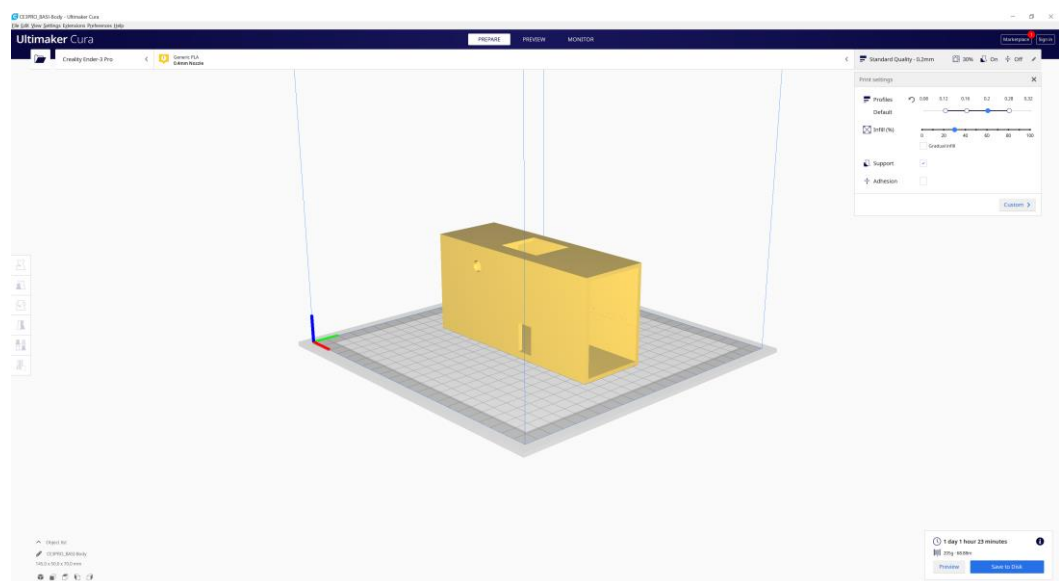
Το freeCAD είναι ένα δωρεάν λογισμικό ανοιχτού κώδικα τρισδιάστατης σχεδίασης το οποίο χρησιμοποιείται για τον σχεδιασμό μηχανολογικών, αρχιτεκτονικών και ηλεκτρικών εξαρτημάτων. Δημιουργήθηκε από τους Jürgen Riegel, Werner Mayer και Yorik van Havre τον Οκτώβριο του 2002, βασίζεται στις γλώσσες C++ και Python και είναι συμβατό με τα λειτουργικά συστήματα Linux, macOS, Windows, Unix και FreeBSD. Η τελική μορφή της σχεδίασης του 3D μοντέλου που υλοποιήθηκε στο freeCAD φαίνεται στην εικόνα 3.11.



Εικόνα 3.11: Τελική μορφή μοντέλου 3D

Το freeCAD επιτρέπει την εξαγωγή του μοντέλου σε μορφή αρχείου με κατάληξη .stl, τα αρχεία με κατάληξη .stl δημιουργήθηκαν από την 3D Systems. Περιγράφουν μόνο την γεωμετρία της επιφάνειας ενός τρισδιάστατου αντικειμένου χωρίς καμία αναπαράσταση χρώματος, υφής ή άλλων κοινών χαρακτηριστικών.

Εφόσον πραγματοποιηθεί η εξαγωγή του μοντέλου σε μορφή αρχείου με κατάληξη .stl, χρησιμοποιείται το λογισμικό Ultimaker Cura στο οποίο γίνεται η εισαγωγή του αρχείου που φαίνεται στην εικόνα 3.12.

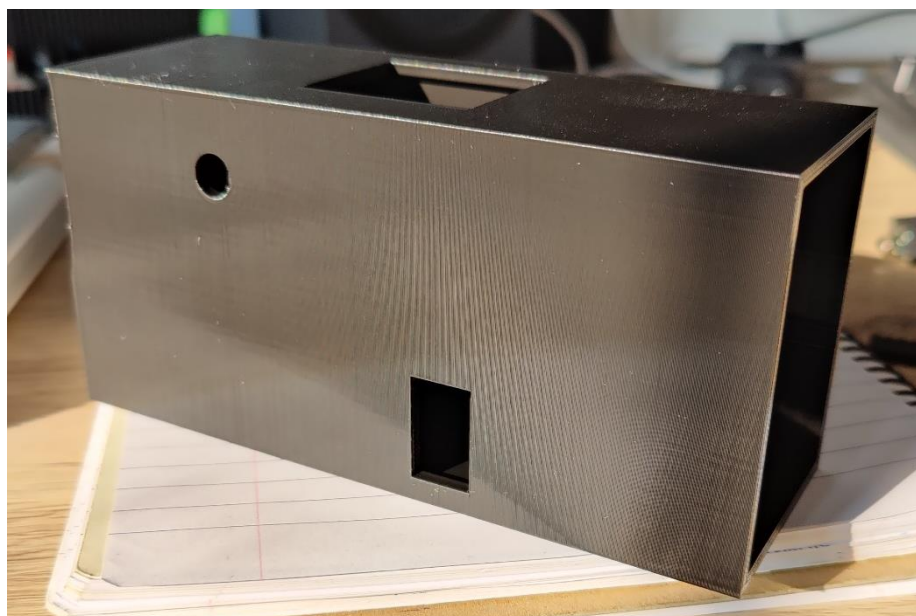


Εικόνα 3.12: Μετατροπή αρχείου .stl σε g - code

Το λογισμικό Ultimaker Cura είναι επίσης ένα δωρεάν λογισμικό ανοιχτού κώδικα που χρησιμοποιείται για την μετατροπή των αρχείων με κατάληξη .stl σε g – code. Η μετατροπή αυτή δημιουργεί αντίστοιχες εντολές σε επίπεδα x, y, z όταν ολοκληρωθεί η διαδικασία αυτή το αρχείο g – code μπορεί να εισαχθεί στον εκτυπωτή, ξεκινώντας έτσι η διαδικασία της εκτύπωσης που απεικονίζεται στην εικόνα 3.13, ενώ στην εικόνα 3.14 φαίνεται το τελικό αποτέλεσμα της 3D εκτύπωσης.



Εικόνα 3.13: Διαδικασία 3D εκτύπωσης πλαισίου συσκευής μέτρησης και εκπομπής δεδομένων



Εικόνα 3.14: Τελικό αποτέλεσμα 3D εκτύπωσης φορητής συσκευής μέτρησης και εκπομπής δεδομένων

3.4 Κατασκευή συσκευής λήψης και καταχώρησης δεδομένων

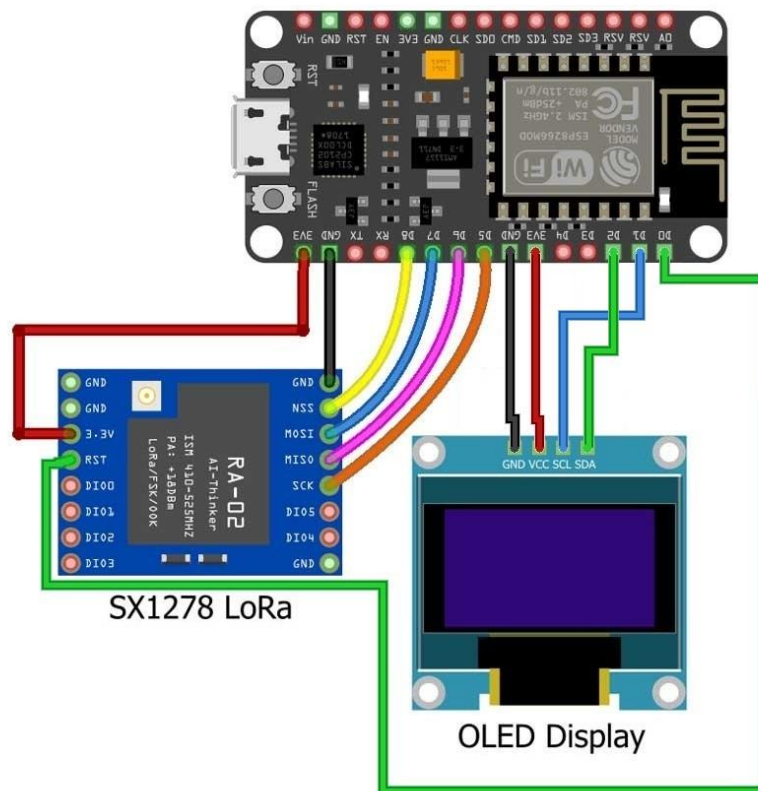
Η διαδικασία της υλοποίησης της συσκευής λήψης και καταχώρησης των δεδομένων περιελάμβανε επίσης τρία στάδια, την σχεδίαση της συνδεσμολογίας του συστήματος, το πειραματικό στάδιο και το κατασκευαστικό στάδιο, τα οποία αναλύονται στις παρακάτω υπό – ενότητες.

3.4.1 Συνδεσμολογία συστήματος λήψης και καταχώρησης δεδομένων

Η συνδεσμολογία του κατασκευαστικού μέρους της συσκευής φαίνεται στην εικόνα 3.15. Για να γίνουν κατανοητές οι συνδέσεις των pins των μονάδων υλοποιήθηκε ο πίνακας 3.2 που φαίνεται παρακάτω.

Πίνακας 3.2: Πίνακας συνδεσμολογίας συσκευής λήψης και καταχώρησης

ESP8266	LoRa Ra - 02	OLED	LED
Pins			
D0	RST		
D1		SCL	
D2		SDA	
D3			+
D4			
D5	SCK		
D6	MISO		
D7	MOSI		
D8	NSS		
VIN			
3V3	3V3	VCC	
GND	GND	GND	-



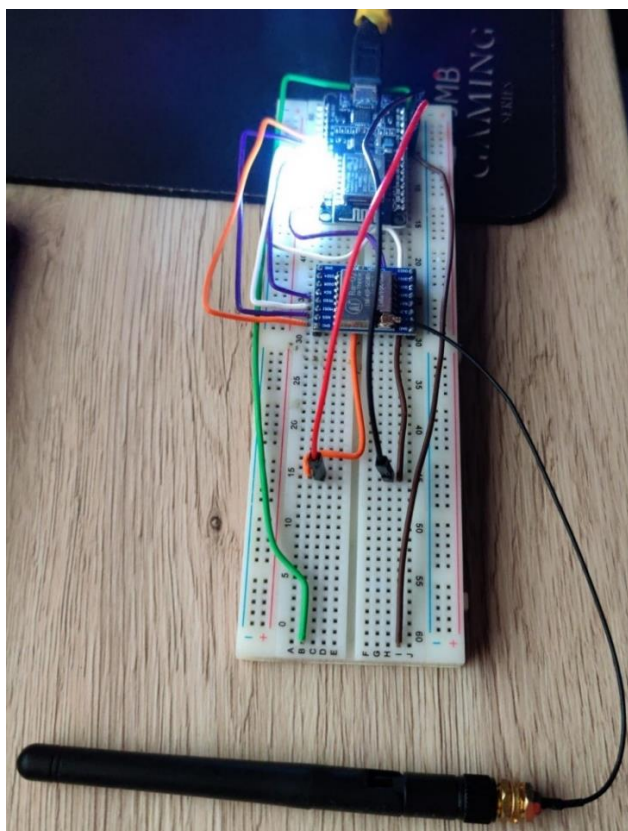
Εικόνα 3.15: Συνδεσμολογία συσκευής λήψης και καταχώρησης δεδομένων

Πιο συγκεκριμένα η συνδεσμολογία των μονάδων της συσκευής είναι η εξής, τα pins D1, D5, D6, D7 και D8 του μικροεπεξεργαστή συνδέονται με τον πομπό – δέκτη LoRa, το ενδεικτικό LED συνδέεται με το pin D2, τέλος η OLED οθόνη συνδέεται στα pins D3 και D4. Όπως συμβαίνει και στην συσκευή μέτρησης και εκπομπής δεδομένων, έτσι και σ’ αυτήν την συσκευή το LED ενεργοποιείται μόνο όταν υπάρχει ζεύξη με την συσκευή μέτρησης και εκπομπής δεδομένων. Σε αντίθετη περίπτωση, το LED παραμένει απενεργοποιημένο.

Τέλος, η συσκευή λήψης και καταχώρησης δεδομένων μπορεί να είναι κι αυτή φορητή όπως η συσκευή μέτρησης και εκπομπής δεδομένων ή μπορεί να έχει σταθερή θέση. Στην περίπτωση που η συσκευή είναι φορητή τότε περιλαμβάνει επίσης δύο επαναφορτιζόμενες μπαταρίες τύπου 18650 των 3,7V – 3000mAh και έναν dc – dc buck μετατροπέα με ρυθμισμένη τάση εξόδου στα 5V με την οποία τροφοδοτείται όλη η συσκευή, διαφορετικά η τροφοδοσία της συσκευής μπορεί να πραγματοποιείται με ένα τροφοδοτικό.

3.4.2 Πειραματική υλοποίηση συστήματος λήψης και καταχώρησης δεδομένων

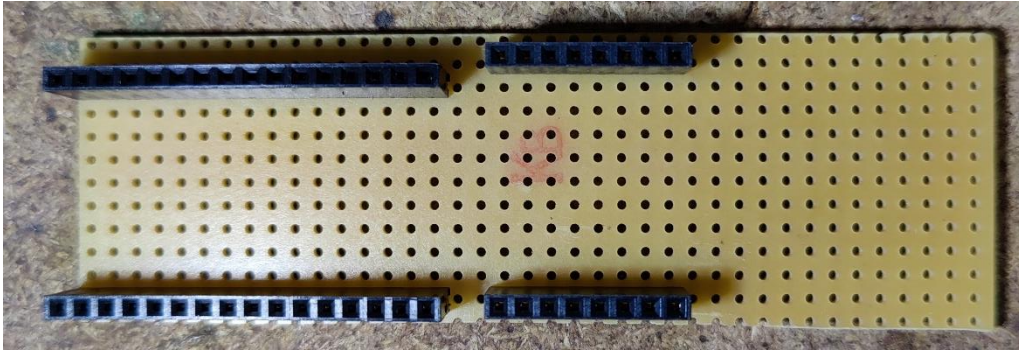
Στο στάδιο αυτό, έγινε η υλοποίηση του συστήματος στο ράστερ, σύμφωνα με την συνδεσμολογία της εικόνας 3.15. Στην εικόνα 3.16 παρουσιάζεται η πειραματική διάταξη του συστήματος στο ράστερ.



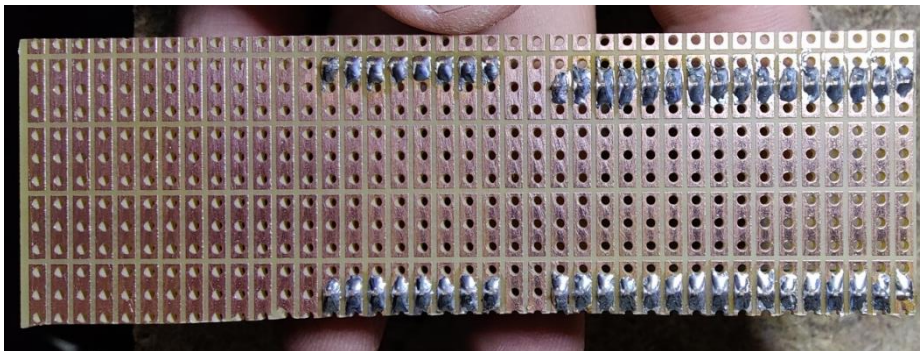
Εικόνα 3.16: Συσκευή λήψης και καταχώρησης δεδομένων σε πειραματικό στάδιο

3.4.3 Κατασκευαστική υλοποίηση συστήματος λήψης και καταχώρησης δεδομένων

Για την υλοποίηση της κατασκευής του συστήματος λήψης και καταχώρησης δεδομένων χρησιμοποιήθηκε μία διάτρητη πλακέτα, στις άκρες της οποίας κολλήθηκαν δύο θηλυκά pin header των 16 θέσεων προκειμένου να τοποθετηθεί ο μικροεπεξεργαστής ESP8266, επίσης κολλήθηκαν δύο θηλυκά pin header των 8 θέσεων πάνω στα οποία τοποθετείται ο πομπό – δέκτης LoRa, όπως φαίνεται στις εικόνες 3.17 και 3.18.

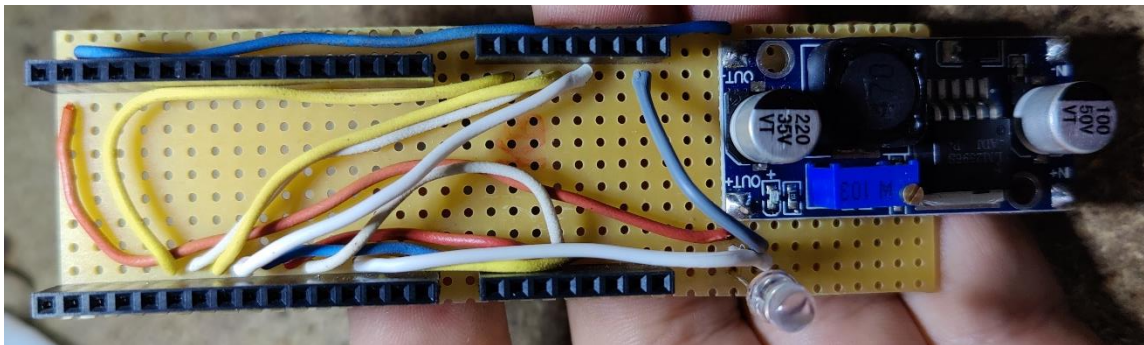


Εικόνα 3.17: Πάνω όψη διάτρησης πλακέτας

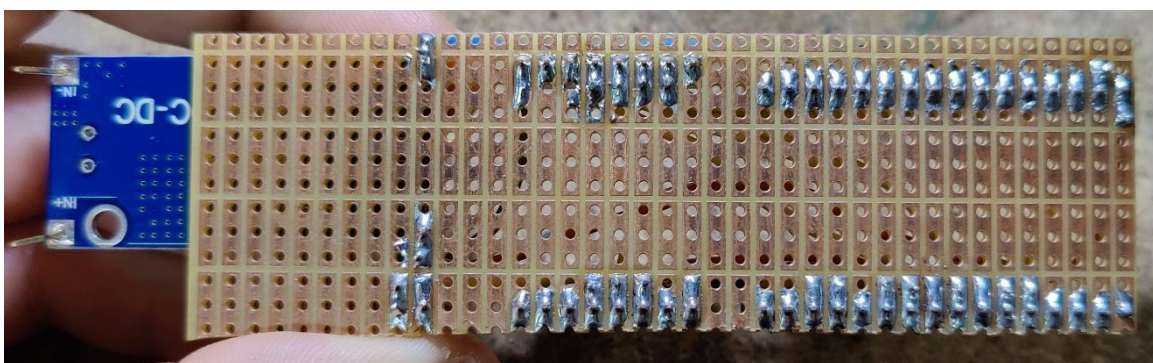


Εικόνα 3.18: Κάτω όψη διάτρησης πλακέτας

Στην συνέχεια, πραγματοποιήθηκε η διαδικασία της συνδεσμολογίας μεταξύ των pin header σύμφωνα με την συνδεσμολογία του συστήματος προκειμένου να υπάρχει επικοινωνία των μονάδων που αποτελούν το σύστημα λήψης και καταχώρησης δεδομένων. Στις εικόνες 3.19 και 3.20 απεικονίζεται το τελικό αποτέλεσμα αυτής της διαδικασίας.



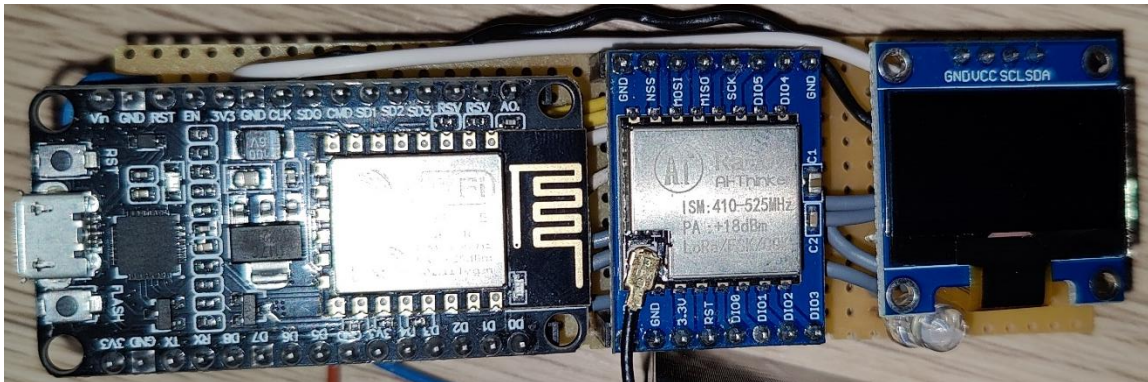
Εικόνα 3.19: Συνδεσμολογία των pin headers (πάνω όψη)



Εικόνα 3.20: Συνδεσμολογία των pin headers (κάτω όψη)

Έπειτα από την ολοκλήρωση της διαδικασίας της συνδεσμολογίας των pins header ακολούθησε έλεγχος συνέχειας (continuity test) σύμφωνα με την συνδεσμολογία του συστήματος (εικόνα 3.15) προκειμένου να διαπιστωθεί ότι όλες η συνδέσεις είναι σύμφωνες με την συνδεσμολογία που σχεδιάστηκε.

Τέλος, στις αντίστοιχες θέσεις των θηλυκών pins headers, τοποθετήθηκαν οι μονάδες που περιλαμβάνει το σύστημα λήψης και καταχώρησης δεδομένων και είναι το ESP8266, το LoRa module και ο dc – dc buck converter. Στην εικόνα 3.21 απεικονίζεται η τελική μορφή της κατασκευής του συστήματος (πάνω όψη), στην συνέχεια τοποθετήθηκε και η OLED οθόνη, σύμφωνα πάντα με την συνδεσμολογία του συστήματος (εικόνα 3.15).



Εικόνα 3.21: Τελική μορφή συστήματος λήψης και καταχώρησης δεδομένων

3.4.4 Σχεδίαση και υλοποίηση 3D πλαισίου

Τέλος, η διαδικασία της σχεδίασης και της υλοποίησης του 3D πλαισίου πραγματοποιήθηκε με τον ίδιο τρόπο που παρουσιάζεται στην υπό – ενότητα 3.3.4. Για την σχεδίαση του πλαισίου χρησιμοποιήθηκε το λογισμικό freeCAD, ενώ για την μετατροπή του αρχείου από .stl σε g – code χρησιμοποιήθηκε το λογισμικό Ultimaker Cura. Η τελική μορφή του πλαισίου του συστήματος λήψης και καταχώρησης δεδομένων φαίνεται στην εικόνα 3.22.



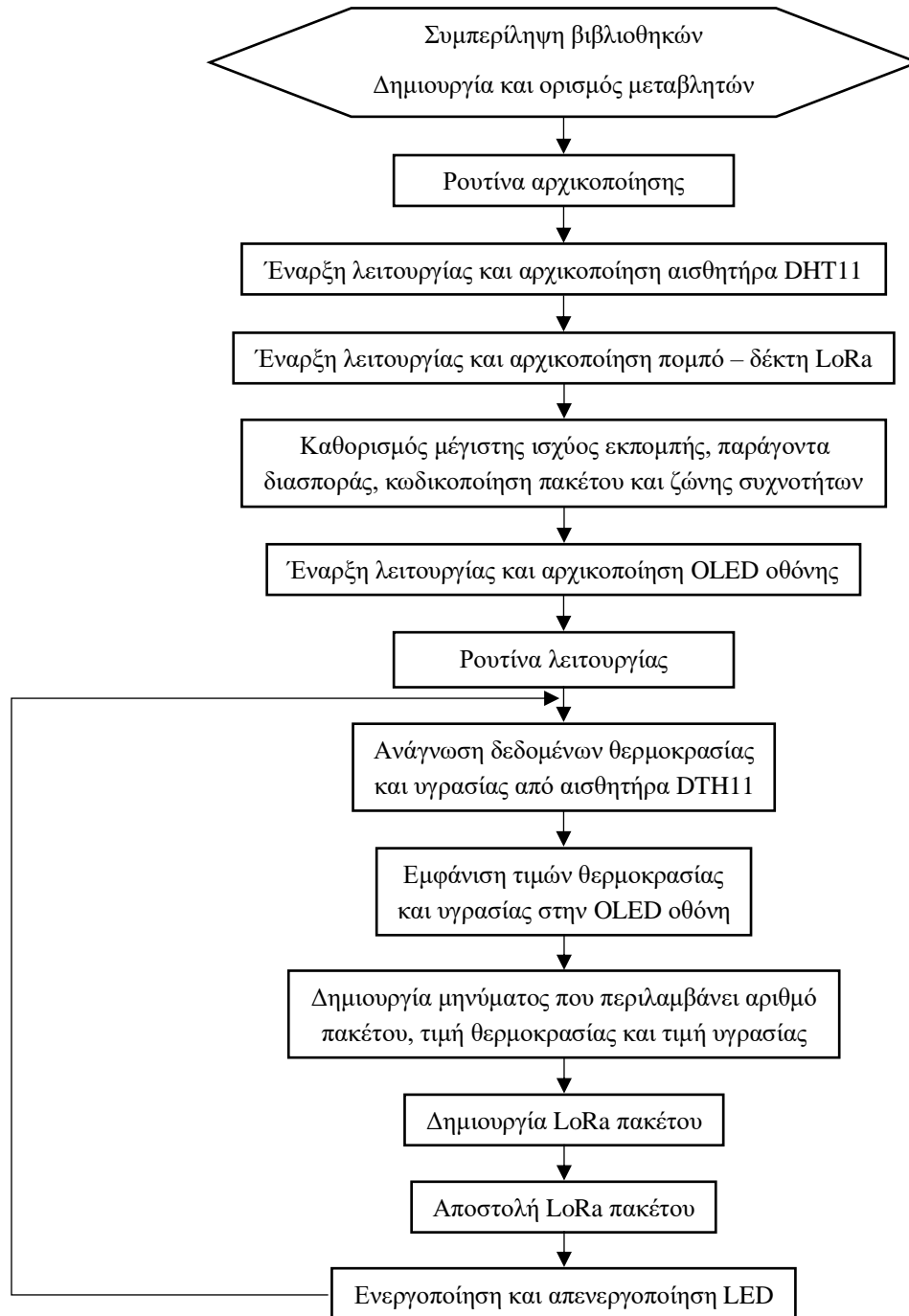
Εικόνα 3.22: Τελικό αποτέλεσμα 3D εκτύπωσης φορητής συσκευής λήψης και καταχώρησης δεδομένων

3.5 Λογισμικό συσκευών

Στην υπό – ενότητα αυτή, παρουσιάζονται τα λογισμικά που βρίσκονται εγκατεστημένα στους μικροεπεξεργαστές των συσκευών. Η ανάπτυξη των προγραμμάτων για την κάθε συσκευή βασίζεται στην κατασκευαστική δομή του αντίστοιχου συστήματος.

3.5.1 Διάγραμμα ροής προγράμματος συσκευής μέτρησης και εκπομπής δεδομένων

Στην παρακάτω εικόνα 3.23 παρουσιάζεται το διάγραμμα ροής του προγράμματος της συσκευής μέτρησης και εκπομπής δεδομένων. Το διάγραμμα ροής παρουσιάζει τον τρόπο σκέψης που αναπτύχθηκε προκειμένου να υλοποιηθεί ο κώδικας της συσκευής.



Εικόνα 3.23: Διάγραμμα ροής προγράμματος συσκευής μέτρησης και εκπομπής δεδομένων

Αρχικά, συμπεριλαμβάνονται στην μνήμη του μικροεπεξεργαστή όλες οι κατάλληλες βιβλιοθήκες, έπειτα δημιουργούνται και ορίζονται οι μεταβλητές που χρησιμοποιούνται κατά την εκτέλεση του προγράμματος. Ακολουθεί η εκτέλεση της ρουτίνας αρχικοποίησης η οποία προσδιορίζει τα pin εισόδων και εξόδων και τίθενται σε εκκίνηση λειτουργίας τα περιφερειακά του μικροεπεξεργαστή. Τέλος, εκτελείται η ρουτίνα λειτουργίας του προγράμματος όπου γίνεται η μέτρηση, η απεικόνιση, η σύνθεση και η αποστολή των δεδομένων που προκύπτουν.

3.5.2 Πρόγραμμα συσκευής μέτρησης και εκπομπής δεδομένων

Σύμφωνα με το διάγραμμα ροής, προκύπτει το πρόγραμμα λειτουργίας της συσκευής που απεικονίζεται στις παρακάτω εικόνες. Προκειμένου να γίνει κατανοητός ο τρόπος λειτουργίας του προγράμματος, δίπλα από κάθε εντολή υπάρχουν σχολιασμοί που ως σκοπό έχουν την περιγραφή της εκτέλεσης που πραγματοποιεί η αντίστοιχη εντολή.

```
#include <SPI.h> // περίληψη βιβλιοθήκης
#include <LoRa.h> // περίληψη βιβλιοθήκης
#include <DHT.h> // περίληψη βιβλιοθήκης
#include <Adafruit_Sensor.h> // περίληψη βιβλιοθήκης
#include <Adafruit_GFX.h> // περίληψη βιβλιοθήκης
#include <Adafruit_SSD1306.h> // περίληψη βιβλιοθήκης

#define DHTPIN 2 // ονοματοδοσία pin 2 σε DHTPIN
DHT dht(DHTPIN, DHT11); // ορισμός DHTPIN ως pin εισόδου
#define led 0 // ονοματοδοσία pin 0 σε led

const int csPin = 15; // ονοματοδοσία pin 15 σε csPin
const int resetPin = 16; // ονοματοδοσία pin 16 σε resetPin
const int irqPin = 10; // ονοματοδοσία pin 10 σε irqPin
int state = 0; // δημιουργία μεταβλητής state και αρχικοποίηση τιμής
int counter = 0; // δημιουργία μεταβλητής counter και αρχικοποίηση τιμής
int SyncWord = 0x22; // δημιουργία μεταβλητής SyncWord και αρχικοποίηση τιμής

int readingID = 0; // δημιουργία μεταβλητής readingID και αρχικοποίηση τιμής
String LoRaMessage = ""; // δημιουργία μεταβλητής LoRaMessage και αρχικοποίηση τιμής
float temperature = 0; // δημιουργία μεταβλητής temperature και αρχικοποίηση τιμής
float humidity = 0; // δημιουργία μεταβλητής humidity και αρχικοποίηση τιμής

#define SCREEN_WIDTH 128 // αρχικοποίηση τιμής του πλάτους της OLED οθόνης σε pixels
#define SCREEN_HEIGHT 64 // αρχικοποίηση τιμής του ύψους της OLED οθόνης σε pixels
#define OLED_RESET -1 // αρχικοποίηση του reset pin της OLED οθόνης
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET); // ορισμός των pins και του μεγέθους της OLED οθόνης
```

Εικόνα 3.24: Περίληψη βιβλιοθηκών και αρχικοποιήσεις μεταβλητών

Στην εικόνα 3.24 φαίνονται οι περιλήψεις των απαραίτητων βιβλιοθηκών καθώς επίσης και η δημιουργία διαφόρων μεταβλητών που χρησιμοποιούνται στις ρουτίνες και τις υπό – ρουτίνες του προγράμματος. Στην εικόνα 3.25 παρουσιάζονται οι υπό – ρουτίνες αρχικοποίησης του αισθητήρα DHT11 και του πομπό – δέκτη LoRa. Επίσης, απεικονίζεται και η υπό – ρουτίνα ανάγνωσης και απεικόνισης των δεδομένων στην OLED οθόνη.

```

void startLoRa() // υπορουτίνα αρχικοποίησης πομπό - δέκτη LoRa
{
  LoRa.setPins(csPin, resetPin, irqPin); // ορισμός pin πομπό - δέκτη LoRa
  while (!LoRa.begin(433E6) && counter < 10) // εκκίνηση του πομπό - δέκτη LoRa
    στην συχνότητα 433MHz
  {
    Serial.print("."); // εμφάνιση συμβόλου αναμονής
    counter++; // αύξηση της μεταβλητής counter κατά 1
    delay(500); // χρονοκαθυστέρηση 500ms
  }
  if (counter == 10) // αν η μεταβλητή counter ισούται με 10
  {
    readingID++; // αύξηση της μεταβλητής readingID κατά 1
    Serial.println("Starting LoRa failed!"); // εμφάνιση μηνύματος αποτυχίας
    αρχικοποίησης
  }
  Serial.println("LoRa Initialization OK!"); // εμφάνιση μηνύματος επιτυχίας
  αρχικοποίησης
  delay(2000); // χρονοκαθυστέρηση 2000ms
}

void startDHT() // υπορουτίνα αρχικοποίησης αισθητήρα DHT11
{
  if (isnan(humidity) || isnan(temperature)) // εκκίνηση αισθητήρα DHT11
  {
    Serial.println("Failed to read from DHT sensor!"); // εμφάνιση μηνύματος
    αποτυχίας εκκίνησης
    return; // επιστροφή στην void setup
  }
}

void getReadings() // υπορουτίνα ανάγνωσης και εμφάνισης δεδομένων απο τον
αισθητήρα DHT11
{
  humidity = dht.readHumidity(); // ανάγνωση και καταχώρηση τιμής υγρασίας στην
  μεταβλητή humidity
  temperature = dht.readTemperature(); // ανάγνωση και καταχώρηση τιμής
  θερμοκρασίας στην μεταβλητή temperature
  Serial.print(F("Humidity: ")); // εμφάνιση μηνύματος
  Serial.print(humidity); // εμφάνιση τιμής της μεταβλητής humidity
  Serial.print(F("% Temperature: ")); // εμφάνιση μηνύματος
  Serial.print(temperature); // εμφάνιση τιμής της μεταβλητής temperature
  Serial.println(F("°C ")); // εμφάνιση μηνύματος
  display.clearDisplay(); // καθαρισμός OLED οθόνης
  display.setTextSize(1); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
  display.setCursor(0, 0); // επιλογή θέσης του κέρσορα OLED οθόνης
  display.print("Temperature: "); // εμφάνιση μηνύματος στην OLED οθόνη
  display.setTextSize(2); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
  display.setCursor(0, 10); // επιλογή θέσης του κέρσορα OLED οθόνης
  display.print(temperature); // εμφάνιση τιμής της μεταβλητής temperature στην
  OLED οθόνη
  display.print(" "); // εμφάνιση μηνύματος στην OLED οθόνη
  display.setTextSize(1); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
  display.cp437(true); // επιλογή γραμματοσειράς OLED οθόνης
  display.write(167); // εμφάνιση συμβόλου στην OLED οθόνη
  display.setTextSize(2); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
  display.print("C"); // εμφάνιση μηνύματος στην OLED οθόνη
  display.setTextSize(1); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
  display.setCursor(0, 35); // επιλογή θέσης του κέρσορα OLED οθόνης
  display.print("Humidity: "); // εμφάνιση μηνύματος στην OLED οθόνη
  display.setTextSize(2); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
  display.setCursor(0, 45); // επιλογή θέσης του κέρσορα OLED οθόνης
  display.print(humidity); // εμφάνιση τιμής της μεταβλητής humidity στην OLED
  οθόνη
  display.print(" %"); // εμφάνιση συμβόλου στην OLED οθόνη
  display.display(); // εφαρμογή των αλλαγών OLED οθόνης
}

```

Εικόνα 3.25: Υπό – ρουτίνες αρχικοποίησης, ανάγνωσης και εμφάνισης δεδομένων

```

void sendReadings() // υπορουτίνα αποστολής δεδομένων
{
    LoRaMessage = String(readingID) + "/" + String(temperature) + "&" +
String(humidity); // σύνθεση μηνύματος πακέτου αποστολής με τα δεδομένα μέτρησης
    LoRa.beginPacket(); // έναρξη πακέτου
    LoRa.print(LoRaMessage); // μήνυμα
    LoRa.endPacket(); // τέλος πακέτου
    Serial.print("Sending packet: "); // εμφάνιση μηνύματος
    Serial.println(readingID); // εμφάνιση της τιμής μεταβλητής readingID
    readingID++; // αύξηση της μεταβλητής readingID κατά 1
    Serial.println(LoRaMessage); // εμφάνιση μηνύματος
}

void setup() // έναρξη ρουτίνας αρχικοποιήσεων
{
    pinMode(led,OUTPUT); // ορισμός pin led ως έξοδο
    Serial.begin(115200); // ορισμός τιμής baud rate
    dht.begin(); // ενεργοποίηση αισθητήρα DHT11
    startDHT(); // κλήση υπό - ρουτίνας αρχικοποίησης αισθητήρα DHT11
    startLoRA(); // κλήση υπό - ρουτίνας αρχικοποίησης πομπό - δέκτη LoRa
    LoRa.setTxPower(20); // ορισμός μέγιστης ισχύος εκπομπής
    LoRa.setSpreadingFactor(12); // ορισμός μέγιστου παράγοντα διασποράς
    LoRa.setSignalBandwidth(20.8E3); // ορισμός ζώνης BW σε 20.8MHz
    LoRa.setCodingRate4(8); // ορισμός μέγιστου ρυθμού κωδικοποίησης
    LoRa.setSyncWord(SyncWord); // ορισμός byte της λέξης συγχρονισμού
    display.begin(SSD1306_SWITCHCAPVCC, 0x3C); // ενεργοποίηση OLED οθόνης
    display.clearDisplay(); // καθαρισμός OLED οθόνης
    display.setTextColor(WHITE); // ορισμός χρώματος γραμματοσειράς OLED οθόνης
}

void loop() // έναρξη ρουτίνας λειτουργίας
{
    getReadings(); // κλήση υπορουτίνας ανάγνωσης και εμφάνισης δεδομένων απο τον
    αισθητήρα DHT11
    sendReadings(); // κλήση υπορουτίνας αποστολής δεδομένων
    digitalWrite(led, HIGH); // ενεργοποίηση led
    delay(100); // χρονοκαθυστέρηση 100ms
    digitalWrite(led, LOW); // απενεργοποίηση led
    delay(100); // χρονοκαθυστέρηση 100ms
}

```

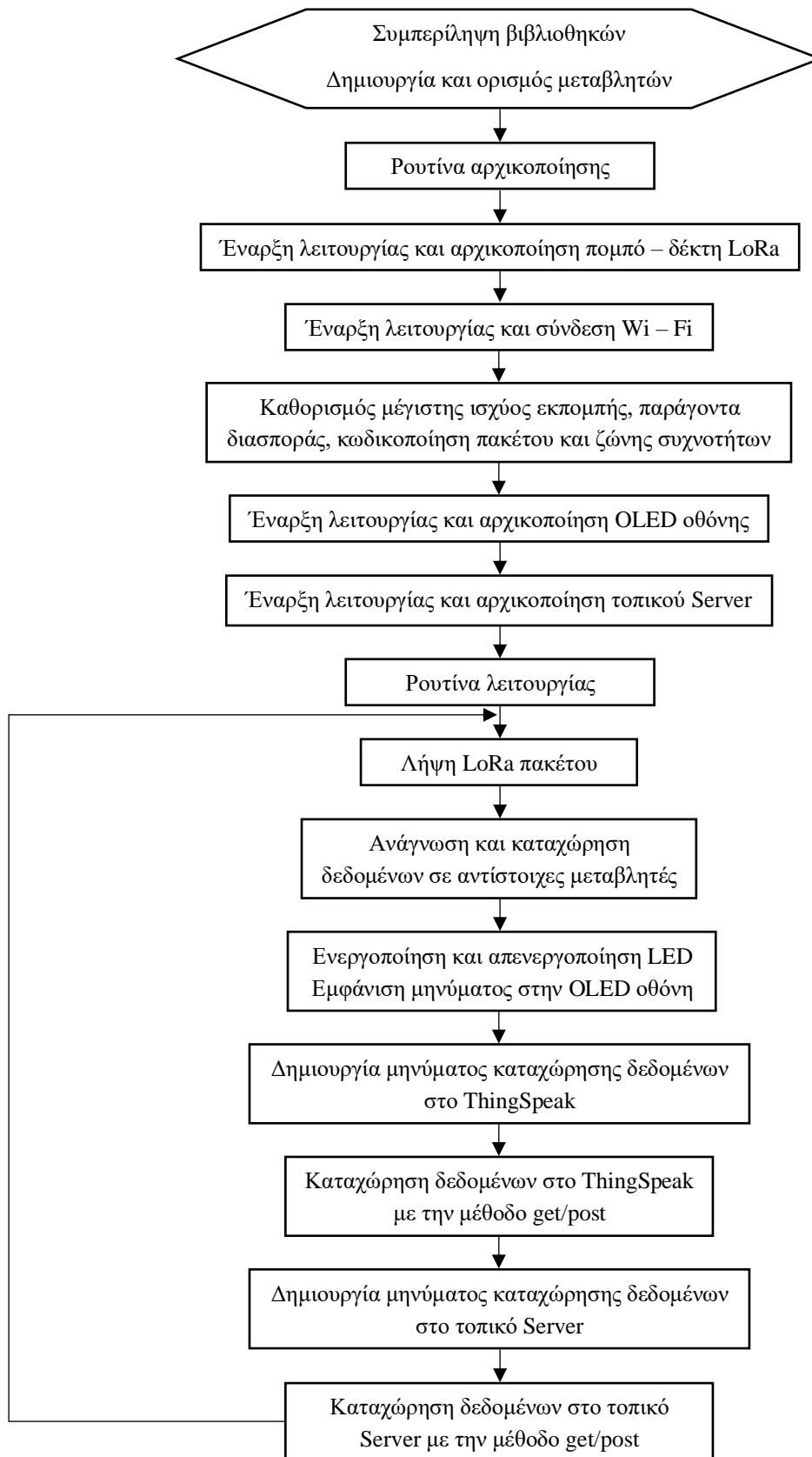
Εικόνα 3.26: Υπό - ρουτίνα αποστολής δεδομένων, ρουτίνα αρχικοποίησης, ρουτίνα λειτουργίας

Στην εικόνα 3.26 παρουσιάζεται η υπό – ρουτίνα που είναι υπεύθυνη για την σύνταξη και την αποστολή του μηνύματος που περιλαμβάνει τα δεδομένα μετρήσεων. Επίσης, απεικονίζεται η ρουτίνα αρχικοποίησης και η ρουτίνα εκτέλεσης, κατά την εκτέλεση του προγράμματος ο μικροεπεξεργαστής εκκινεί από την ρουτίνα αρχικοποίησης (void setup), μέσα από την οποία κάνει κλήσεις στις υπό – ρουτίνες αρχικοποιήσεων που προηγούνται. Εφόσον εκτελέσει τις εντολές που βρίσκονται μέσα στην υπό – ρουτίνα αρχικοποίησης τότε εισέρχεται στην ρουτίνα εκτέλεσης (void loop), και παραμένει εκεί εκτελώντας τις εντολές για όσο διάστημα βρίσκεται ενεργοποιημένο το σύστημα.

Σημείωση: Το πρόγραμμα σπάστηκε σε επί μέρους κομμάτια προκειμένου να ενσωματωθεί στην δομή της εργασίας. Ολόκληρη η δομή του προγράμματος της συσκευής μέτρησης και εκπομπής δεδομένων παρουσιάζεται στο παράρτημα Β.

3.5.3 Διάγραμμα ροής προγράμματος συσκευής λήψης και καταχώρησης δεδομένων

Στην εικόνα 3.27 που ακολουθεί, απεικονίζεται το διάγραμμα ροής της συσκευής.



Εικόνα 3.27: Διάγραμμα ροής προγράμματος συσκευής λήψης και καταχώρησης δεδομένων

Όπως αναφέρθηκε και στην υπό – ενότητα 3.4.1, η συσκευή καταχωρεί τα δεδομένα που λαμβάνει σε έναν τοπικό server και στο ThingSpeak. Αρχικά, συμπεριλαμβάνονται στην μνήμη του μικροεπεξεργαστή όλες οι κατάλληλες βιβλιοθήκες, έπειτα δημιουργούνται και ορίζονται οι μεταβλητές που χρησιμοποιούνται κατά την εκτέλεση του προγράμματος. Ακολουθεί η εκτέλεση της ρουτίνας αρχικοποίησης η οποία προσδιορίζει τα pin εισόδων και εξόδων και τίθενται σε εκκίνηση λειτουργίας τα περιφερειακά του μικροεπεξεργαστή. Τέλος, εκτελείται η ρουτίνα λειτουργίας του προγράμματος στην οποία γίνεται η λήψη, η ανάγνωση, ο διαχωρισμός, η σύνθεση και η καταχώρηση των δεδομένων που ελήφθησαν προκειμένου να πραγματοποιηθεί η καταχώρηση τους πρώτα στο ThingSpeak και έπειτα στον τοπικό server.

3.5.4 Πρόγραμμα συσκευής λήψης και καταχώρησης δεδομένων

Σύμφωνα με το διάγραμμα ροής του συστήματος λήψης και καταχώρησης δεδομένων (εικόνα 3.27) προκύπτει το πρόγραμμα λειτουργίας της συσκευής που απεικονίζεται στις παρακάτω εικόνες.

```
#include <ESP8266WiFi.h> // περίληψη βιβλιοθήκης
#include <WiFiClient.h> // περίληψη βιβλιοθήκης
#include <ESP8266WebServer.h> // περίληψη βιβλιοθήκης
#include <SPI.h> // περίληψη βιβλιοθήκης
#include <LoRa.h> // περίληψη βιβλιοθήκης
#include <Adafruit_Sensor.h> // περίληψη βιβλιοθήκης
#include <Adafruit_GFX.h> // περίληψη βιβλιοθήκης
#include <Adafruit_SSD1306.h> // περίληψη βιβλιοθήκης
#include "index.h" //περίληψη αρχείου σελίδας τοπικού server

#define led 0 // ονοματοδοσία pin 0 σε led
#define SCREEN_WIDTH 128 // αρχικοποίηση τιμής του πλάτους της OLED οθόνης σε pixels
#define SCREEN_HEIGHT 64 // αρχικοποίηση τιμής του ύψους της OLED οθόνης σε pixels
#define OLED_RESET -1 // αρχικοποίηση του RST pin της OLED οθόνης
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
// ορισμός των pins και του μεγέθους της OLED οθόνης

const int csPin = 15; // ονοματοδοσία pin 15 σε csPin
const int resetPin = 16; // ονοματοδοσία pin 16 σε resetPin
const int irqPin = 10; // ονοματοδοσία pin 10 σε irqPin
int state = 0; // δημιουργία μεταβλητής state και αρχικοποίηση τιμής
int SyncWord = 0x22; // δημιουργία μεταβλητής SyncWord και αρχικοποίηση τιμής
int counter = 0; // δημιουργία μεταβλητής counter και αρχικοποίηση τιμής

String apiKey = "SEYK51W0DHAPLUE7"; // δημιουργία μεταβλητής apiKey και αρχικοποίηση τιμής
const char* ssid = "ACTROS_1840"; // δημιουργία μεταβλητής ssid και αρχικοποίηση τιμής
const char* password = "0481_SORTCA"; // δημιουργία μεταβλητής password και αρχικοποίηση τιμής
const char* server1 = "api.thingspeak.com"; // δημιουργία μεταβλητής server και αρχικοποίηση τιμής

WiFiClient client; // αρχικοποίηση βιβλιοθήκης client
ESP8266WebServer server(80); // άνοιγμα πόρτας 80 στον server

int rssi; // δημιουργία μεταβλητής rssi
String loRaMessage; // δημιουργία μεταβλητής loRaMessage
String temperature; // δημιουργία μεταβλητής temperature
String humidity; // δημιουργία μεταβλητής humidity
String readingID; // δημιουργία μεταβλητής readingID
```

Εικόνα 3.28: Περίληψη βιβλιοθηκών και αρχικοποιήσεις μεταβλητών

```

String processor(const String& var) // υπό - ρουτίνα μετατροπής των δεδομένων λήψης
{
  if (var == "TEMPERATURE")      // αν var = TEMPERATURE
  {
    return temperature;         // επιστροφή temperature
  } else if (var == "HUMIDITY")  // αν var = HUMIDITY
  {
    return humidity;           // επιστροφή humidity
  } else if (var == "RRSI")     // αν var = RRSI
  {
    return String(rssi);       // επιστροφή rssi
  }
  return String();             // επιστροφή
}

void handleRoot() // ενεργοποίηση υπό - ρουτίνας όταν ανοίγει η IP του τοπικού
server σε browser
{
  String s = MAIN_page; // διαβάζει τον κώδικα απο το html αρχείο
  server.send(200, "text/html", s); // στέλνει τα δεδομένα στην σελίδα
}

void handleADC() // ενεργοποίηση υπό - ρουτίνας για να γίνει η καταχώρηση
δεδομένων στον τοπικό server
{
  String data = "{\"Packet\": \""+readingID+"\", \"Temperature\": \""+ temperature
+ "\", \"Humidity\": \""+ humidity + "\", \"RSSI\": \""+String(rssi)+"\"}";
  // δημιουργία συμβολοσειράς για καταχώρηση τιμών στον τοπικό server
  server.send(200, "text/plane", data); // αποστολή και καταχώρηση συμβολοσειράς
στον τοπικό server με την μέθοδο get/post
}

void startLoRa() // υπορουτίνα αρχικοποίησης πομπό - δέκτη LoRa
{
  LoRa.setPins(csPin, resetPin, irqPin); // ορισμός pin πομπό - δέκτη LoRa
  while (!LoRa.begin(433E6) && counter < 10) // εκκίνηση του πομπό - δέκτη LoRa
στην συχνότητα 433MHz
  {
    Serial.print("."); // εμφάνιση συμβόλου αναμονής
    counter++; // αύξηση της μεταβλητής counter κατά 1
    delay(500); // χρονοκαθυστέρηση 500ms
  }
  if (counter == 10) // αν η μεταβλητή counter ισούται με 10
  {
    Serial.println("Starting LoRa failed!"); // εμφάνιση μηνύματος αποτυχίας
αρχικοποίησης
  }
  Serial.println("LoRa Initialization OK!"); // εμφάνιση μηνύματος επιτυχίας
αρχικοποίησης
  display.setCursor(0,10); // επιλογή θέσης του κέρσορα OLED οθόνης
  display.print("LoRa OK!"); // εμφάνιση μηνύματος στην OLED οθόνη
  display.display(); // εφαρμογή των αλλαγών OLED οθόνης
  delay(2000); // χρονοκαθυστέρηση 2000ms
}

```

Εικόνα 3.29: Υπό – ρουτίνες μετατροπής δεδομένων λήψης, δημιουργίας τοπικού server, καταχώρησης δεδομένων στον τοπικό server και αρχικοποίησης LoRa πομπό – δέκτη

```

void startWiFi() // υπορουτίνα αρχικοποίησης και σύνδεσης Wi - Fi
{
  Serial.print("Connecting to "); // εμφάνιση μηνύματος σύνδεσης Wi - Fi
  Serial.println(ssid);           // εμφάνιση ονόματος Wi - Fi
  WiFi.begin(ssid, password);     // χρησιμοποίηση ονόματος και κωδικού Wi - Fi για σύνδεση
  Serial.println("");             // εμφάνιση μηνύματος
  while (WiFi.status() != WL_CONNECTED) // αναμονή σύνδεσης στο Wi - Fi
  {
    delay(500);                  // χρονοκαθυστέρηση 500ms
    Serial.print(".");           // εμφάνιση συμβόλου αναμονής
  }
  Serial.println("");            // εμφάνιση μηνύματος
  Serial.print("Connected to "); // εμφάνιση μηνύματος ότι συνδέθηκε στο Wi - Fi
  Serial.println(ssid);         // εμφάνιση ονόματος Wi - Fi
  Serial.print("IP address: "); // εμφάνιση μηνύματος για την IP
  Serial.println(WiFi.localIP()); // εμφάνιση IP που έχει δωθεί στον ESP
  display.clearDisplay();       // καθαρισμός OLED οθόνης
  display.print("Wi - Fi connect"); // εμφάνιση μηνύματος στην OLED οθόνη
  display.clearDisplay();       // καθαρισμός OLED οθόνης
  display.setCursor(0, 20);     // επιλογή θέσης του κέρσορα OLED οθόνης
  display.print("Access web server at: "); // εμφάνιση μηνύματος στην OLED οθόνη
  display.setCursor(0, 30);     // επιλογή θέσης του κέρσορα OLED οθόνης
  display.print(WiFi.localIP()); // εμφάνιση της IP στην OLED οθόνη
  display.display();           // εφαρμογή των αλλαγών OLED οθόνης
  delay(2000);                 // χρονοκαθυστέρηση 2000ms
}

void startServer() // υπορουτίνα αρχικοποίησης και εκκίνησης του τοπικού server
{
  server.on("/", handleRoot);   // ενεργοποίηση σελίδας τοπικού server
  server.on("/readADC", handleADC); // κλήση σελίδας
  server.begin();               // εκκίνηση τοπικού server
  Serial.println("HTTP server started"); // εμφάνιση μηνύματος για την εκκίνηση του τοπικού
server
  display.setCursor(0, 40);     // επιλογή θέσης του κέρσορα OLED οθόνης
  display.print("Server OK"); // εμφάνιση μηνύματος στην OLED οθόνη
  display.display();           // εφαρμογή των αλλαγών OLED οθόνης
  delay(2000);                 // χρονοκαθυστέρηση 2000ms
}

```

Εικόνα 3.30:Υπό - ρουτίνες αρχικοποίησης και σύνδεσης Wi - Fi, αρχικοποίησης και εκκίνησης τοπικού server

Στην εικόνα 3.28, απεικονίζονται οι περιλήψεις των απαραίτητων βιβλιοθηκών προκειμένου η συσκευή να μπορέσει να πραγματοποιήσει την λήψη και την καταχώρηση των δεδομένων μέτρησης σε τοπικό server και στο ThingSpeak. Στην εικόνα 3.29 παρουσιάζονται η υπό – ρουτίνα μετατροπής των δεδομένων λήψης, η υπό – ρουτίνα δημιουργίας τοπικού server, η υπό – ρουτίνα καταχώρησης των δεδομένων στον τοπικό server και η υπό – ρουτίνα αρχικοποίησης του πομπό – δέκτη LoRa. Στην εικόνα 3.30 φαίνονται η υπό – ρουτίνα αρχικοποίησης και σύνδεσης με το Wi – Fi και η υπό – ρουτίνα αρχικοποίησης και εκκίνησης του τοπικού server.

Στην εικόνα 3.31 που ακολουθεί παρουσιάζονται η υπό – ρουτίνα ανάλυσης των δεδομένων λήψης όπως επίσης η υπό – ρουτίνα σύνθεσης και καταχώρησης δεδομένων στο ThingSpeak. Κλείνοντας, στην εικόνα 3.32 φαίνονται οι υπό – ρουτίνα αρχικοποίησης της OLED οθόνης καθώς επίσης η ρουτίνα αρχικοποιήσεων η οποία εκτελείται κάθε φορά που γίνεται εκκίνηση ή επανεκκίνηση του συστήματος και η ρουτίνα εκτέλεσης του προγράμματος μέσα από την οποία πραγματοποιείται η λήψη, η επεξεργασία και η καταχώρηση των δεδομένων στον τοπικό server και στο ThingSpeak.

```

void receiveData() // υπορουτίνα ανάλυσης και καταχώρησης δεδομένων
{
    Serial.print("Lora packet received: "); // εμφάνιση μηνύματος
    while (LoRa.available()) // ενεργοποίηση πομπό - δέκτη LoRa
    {
        String LoRaData = LoRa.readString(); // ανάγνωση και καταχώρηση τιμής LoRa πακέτου
        Serial.print(LoRaData); // εμφάνιση τιμής της μεταβλητής LoRaData
        int pos1 = LoRaData.indexOf('/'); // καταχώρηση τιμής του αριθμού των χαρακτήρων ως το
        int pos2 = LoRaData.indexOf('&'); // καταχώρηση τιμής του αριθμού των χαρακτήρων ως το
        readingID = LoRaData.substring(0, pos1); // ανάγνωση και καταχώρηση τιμής αριθμού
        temperature = LoRaData.substring(pos1 + 1, pos2); // ανάγνωση και καταχώρηση τιμής
        humidity = LoRaData.substring(pos2 + 1, LoRaData.length()); // ανάγνωση και καταχώρηση
        rssi = LoRa.packetRssi(); // ανάγνωση και καταχώρηση τιμής ένδειξης της ισχύος του
        Serial.print(" with RSSI "); // εμφάνιση μηνύματος
        Serial.println(rssi); // εμφάνιση τιμής της μεταβλητής rssi
    }
}

void sendThingSpeak() // υπορουτίνα σύνθεσης και καταχώρησης δεδομένων στο ThingSpeak
{
    if (client.connect(server1, 80)) // εκκίνηση καταχώρησης δεδομένων στο ThingSpeak
    {
        String postStr = apiKey; // καταχώρηση στην μεταβλητή postStr την τιμή της μεταβλητής
        postStr += "&field1="; // καταχώρηση στην μεταβλητή postStr για την επιλογή του field 1
        postStr += String(readingID); // καταχώρηση στο field 1 την τιμή της μεταβλητής readingID
        postStr += "&field2="; // καταχώρηση στην μεταβλητή postStr για την επιλογή του field 2
        postStr += String(temperature); // καταχώρηση στο field 2 την τιμή της μεταβλητής
        postStr += "&field3="; // καταχώρηση στην μεταβλητή postStr για την επιλογή του field 3
        postStr += String(humidity); // καταχώρηση στο field 2 την τιμή της μεταβλητής humidity
        postStr += "&field4="; // καταχώρηση στην μεταβλητή postStr για την επιλογή του field 4
        postStr += String(rssi); // καταχώρηση στο field 2 την τιμή της μεταβλητής rssi
        postStr += "\r\n\r\n\r\n\r\n"; // καταχώρηση στην μεταβλητή postStr το μήνυμα
        client.print("POST /update HTTP/1.1\n"); // δημιουργία συνδέσμου για ThingSpeak εμφάνιση
        client.print("Host: api.thingspeak.com\n"); // δημιουργία συνδέσμου για ThingSpeak εμφάνιση
        client.print("Connection: close\n"); // δημιουργία συνδέσμου για ThingSpeak εμφάνιση
        client.print("X-THINGSPEAKAPIKEY: " + apiKey + "\n"); // καταχώρηση τιμής της μεταβλητής
        client.print("Content-Type: application/x-www-form-urlencoded\n"); // ορισμός τύπου
        client.print("Content-Length: "); // ορισμός μεγέθους του περιεχομένου των δεδομένων
        client.print(postStr.length()); // ανάγνωση του μεγέθους του περιεχομένου των δεδομένων
        client.print("\n\n"); // καταχώρηση δύο enter
        client.print(postStr); // καταχώρηση της τιμής της μεταβλητής postStr
    }
}

```

Εικόνα 3.31: Υπό – ρουτίνες ανάλυσης δεδομένων λήψης, σύνθεσης και καταχώρησης δεδομένων στο ThingSpeak


```

void startOLED() // υπορουτίνα αρχικοποίησης και εκκίνησης της OLED οθόνης
{
    display.begin(SSD1306_SWITCHCAPVCC, 0x3C); // ενεργοποίηση OLED οθόνης
    display.clearDisplay(); // καθαρισμός OLED οθόνης
    display.setTextColor(WHITE); // ορισμός χρώματος γραμματοσειράς OLED οθόνης
    display.setTextSize(1); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
    display.setCursor(0,0); // επιλογή θέσης του κέρσορα OLED οθόνης
    display.print("LORA RECEIVER"); // εμφάνιση μηνύματος στο OLED οθόνη
    display.display(); // εφαρμογή των αλλαγών OLED οθόνης
    delay(2000); // χρονοκαθυστέρηση 2000ms
}

void setup()
{
    pinMode(led,OUTPUT); // ορισμός pin led ως έξοδο
    Serial.begin(115200); // ορισμός τιμής baud rate
    Serial.println(); // αλλαγή γραμμής
    delay(10); // χρονοκαθυστέρηση 10ms
    startOLED(); // κλήση υπό - ρουτίνας αρχικοποίησης και εκκίνησης OLED οθόνης
    startLoRa(); // κλήση υπό - ρουτίνας αρχικοποίησης και ενεργοποίησης πομπό -
δέκτη LoRa
    startWiFi(); // κλήση υπό - ρουτίνας αρχικοποίησης και ενεργοποίησης Wi - Fi
    startServer(); // κλήση υπό - ρουτίνας αρχικοποίησης και εκκίνησης τοπικού server
    LoRa.setSpreadingFactor(12); // ορισμός μέγιστου παράγοντα διασποράς
    LoRa.setSignalBandwidth(20.8E3); // ορισμός ζώνης BW σε 20.8MHz
    LoRa.setCodingRate4(8); // ορισμός μέγιστου ρυθμού κωδικοποίησης
    LoRa.setSyncWord(SyncWord); // ορισμός byte της λέξης συγχρονισμού
    Serial.println("End Setup"); // εμφάνιση μηνύματος
}

void loop()
{
    int packetSize = LoRa.parsePacket(); // καταχώρηση τιμής μεγέθους του πακέτου
στην μεταβλητή packetSize
    if (packetSize) // έλεγχος καταχώρησης τιμής στην μεταβλητή packetSize
    {
        receiveData(); // κλήση υπορουτίνας ανάλυσης και καταχώρησης δεδομένων
        sendThingSpeak(); // κλήση υπορουτίνας σύνθεσης και καταχώρησης δεδομένων
στο ThingSpeak
        digitalWrite(led, HIGH); // ενεργοποίηση led
        delay(100); // χρονοκαθυστέρηση 100ms
        digitalWrite(led, LOW); // απενεργοποίηση led
        delay(100); // χρονοκαθυστέρηση 100ms
        display.clearDisplay(); // καθαρισμός OLED οθόνης
        display.setCursor(0, 0); // επιλογή θέσης του κέρσορα OLED οθόνης
        display.print(WiFi.localIP()); // εμφάνιση της IP στην OLED οθόνη
        display.setCursor(0, 10); // επιλογή θέσης του κέρσορα OLED οθόνης
        display.print("Packet No: "); // εμφάνιση μηνύματος στην OLED οθόνη
        display.setCursor(0, 20); // επιλογή θέσης του κέρσορα OLED οθόνης
        display.print(readingID); // εμφάνιση του αριθμού πακέτου στην OLED οθόνη
        display.display(); // εφαρμογή των αλλαγών OLED οθόνης
    }
    server.handleClient(); // κλήση υπορουτίνας καταχώρησης δεδομένων στον τοπικό
server
}

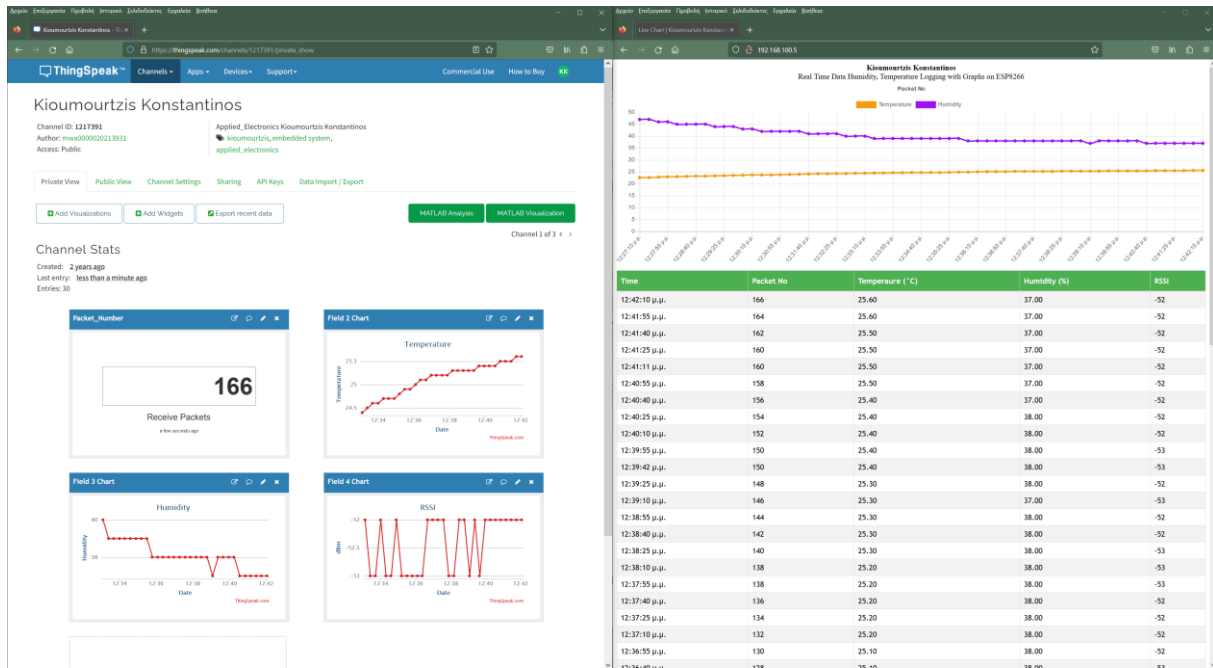
```

Εικόνα 3.32: Υπό – ρουτίνα αρχικοποίησης OLED οθόνης, ρουτίνα αρχικοποίησης και ρουτίνα λειτουργίας

Σημείωση: Το πρόγραμμα σπάστηκε σε επί μέρους κομμάτια προκειμένου να ενσωματωθεί στην δομή της εργασίας. Ολόκληρη η δομή του προγράμματος της συσκευής λήψης και καταχώρησης δεδομένων παρουσιάζεται στο παράρτημα Γ.

3.6 Δημιουργία τοπικού server

Την λειτουργία τοπικού server την εκτελεί ο μικροεπεξεργαστής ESP8266 της συσκευής λήψης και καταχώρησης δεδομένων. Πιο συγκεκριμένα, ο μικροεπεξεργαστής κάνει κλήση του αρχείου `index.h` το οποίο περιλαμβάνει τον κατάλληλο κώδικα προκειμένου να δημιουργηθεί μία web σελίδα όπως φαίνεται στην εικόνα 3.33.



Εικόνα 3.33: Web σελίδα τοπικού server

Όπως διαπιστώνει κανείς, η σελίδα αποτελείται από την περιοχή του γραφήματος όπου απεικονίζεται η καταχώρηση των δεδομένων με την μορφή της γραφικής παράστασης και την περιοχή του πίνακα όπου πραγματοποιείται η καταχώρηση των δεδομένων στα αντίστοιχα κελιά. Η ενημέρωση της σελίδας και η καταχώρηση των δεδομένων γίνεται αυτόματα κάθε 15 δευτερόλεπτα, ο χρόνος αυτός επιλέχθηκε να είναι ίδιος με τον χρόνο της πλατφόρμας ThingSpeak προκειμένου να υπάρχει συσχετισμός.

Στην γραφική παράσταση με μωβ χρώμα απεικονίζονται τα δεδομένα καταχώρησης της υγρασίας, ενώ με πορτοκαλί χρώμα παρουσιάζονται τα δεδομένα καταχώρησης της θερμοκρασίας συναρτήσει του χρόνου. Ο πίνακας περιλαμβάνει πέντε στήλες που είναι η στήλη της ώρας, η στήλη του αριθμού του πακέτου, η στήλη της θερμοκρασίας, η στήλη της υγρασίας και η στήλη του δείκτη RSSI που προσδιορίζει την τιμή της ισχύος του λαμβανόμενου σήματος σε dBm.

Τέλος, να αναφερθεί πως η web σελίδα φαίνεται σε συσκευές που είναι συνδεδεμένες στο ίδιο δίκτυο Wi-Fi, πληκτρολογώντας την IP διεύθυνση που απεικονίζεται στην OLED οθόνη. Προκειμένου να καλυφθεί η ανάγκη για την δημοσιοποίηση των δεδομένων τα δεδομένα καταχωρούνται ταυτόχρονα τόσο τοπικά όσο και δημόσια στην πλατφόρμα του ThingSpeak, όπως αναφέρεται στην συνέχεια της εργασίας. Στις εικόνες που φαίνονται παρακάτω, παρουσιάζεται ο κώδικας που περιλαμβάνει το αρχείο `index.h`.

Στην εικόνα που ακολουθεί 3.34, φαίνονται οι εντολές που είναι υπεύθυνες για την διαμόρφωση της κεφαλίδας της web σελίδας, καθώς επίσης και για την μορφή του πίνακα που θα εμφανίζεται. Πιο συγκεκριμένα, καθορίζεται το γραφικό περιβάλλον στο οποίο θα αποτυπώνονται οι γραφικές παραστάσεις καθώς επίσης και η επιλογή γραμματοσειράς, η στοίχιση, τα χρώματα φόντου και οι αποστάσεις στα κελιά που θα εμφανίζονται οι καταχωρήσεις στον πίνακα.

```
const char MAIN_page[] PROGMEM = R"====(
<!doctype html>
<html>
/* Διαμόρφωση κεφαλίδας */
<head>
  <title>Line Chart | Kioumourtzis Konstantinos</title> // εμφάνιση μηνύματος
  κεφαλίδας
  <script src =
"https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.3/Chart.min.js"></script> //
  κλήση κώδικα για την εμφάνιση του γραφικού περιβάλλοντος
  <style>
  canvas
  {
    -moz-user-select: none;
    -webkit-user-select: none;
    -ms-user-select: none;
  }
  /* Διαμόρφωση πίνακα δεδομένων */
  #dataTable
  {
    font-family: "Trebuchet MS", Arial, Helvetica, sans-serif; // επιλογή
  γραμματοσειράς
    border-collapse: collapse; // σύμπτυξη περιγραμμάτων κελιών
    width: 100%; // καθορισμός πλάτους κελιών
  }
  #dataTable td, #dataTable th
  {
    border: 1px solid #ddd; // διαμόρφωση περιγραμμάτων κελιών
    padding: 8px; // διαμόρφωση συμπλήρωσης κελιών
  }

  #dataTable tr:nth-child(even){background-color: #f2f2f2;} // διαμόρφωση χρώματος
  φόντου κελιών

  #dataTable tr:hover {background-color: #ddd;} // διαμόρφωση χρώματος φόντου
  κελιών

  #dataTable th
  {
    padding-top: 12px; // διαμόρφωση συμπλήρωσης κελιών κορυφής από πάνω
    padding-bottom: 12px; // διαμόρφωση συμπλήρωσης κελιών κορυφής από κάτω
    text-align: left; // αριστερή στοίχιση κειμένου κελιών κορυφής
    background-color: #4CAF50; // διαμόρφωση χρώματος φόντου κελιών κορυφής
    color: white; // διαμόρφωση χρώματος γραμματοσειράς κελιών κορυφής
  }
  </style>
</head>
```

Εικόνα 3.34: Διαμόρφωση κεφαλίδας, γραφικής παράστασης και πίνακα δεδομένων web σελίδας

```

/* Διαμόρφωση σώματος σελίδας */
<body>
  <div style="text-align:center;"><b>Κioumourtzis Konstantinos</b><br>Real Time Data
Humidity, Temperature Logging with Graphs on ESP8266</div> // διαμόρφωση και εμφάνιση
μηνύματος με την λειτουργία που πραγματοποιεί η σελίδα
  <div class="chart-container" position: relative; height:350px; width:100%">
    <canvas id="Chart" width="400" height="400"></canvas>
  </div>
<div>
  <table id="dataTable">
    <tr><th>Time</th><th>Packet No</th><th>Temperaure (&deg;C)</th><th>Humidity
(%)</th><th>RSSI</th></tr> // εμφάνιση μηνύματος στα κελιά κορυφής σε κάθε στήλη με την
πληροφορία που θα περιλαμβάνουν
  </table>
</div>
<br>
<br>
<script>
/* Ορισμός μεταβλητών */
var RSSI = [];
var PacketNo = [];
var Tvalues = [];
var Hvalues = [];
var timeStamp = [];

/* Διαμόρφωση γραφήματος δεδομένων */
function showGraph()
{
  var ctx = document.getElementById("Chart").getContext('2d'); // επιλογή δισδιάστατου
  γραφήματος
  var Chart2 = new Chart(ctx,
  {
    type: 'line', // επιλογή εμφάνισης γραμμής μεταξύ των σημείων
    data:
    {
      labels: timeStamp, // εμφάνιση ετικέτας με την ώρα καταχώρησης
      datasets:
      [{
        label: "Temperature", // εμφάνιση ετικέτας θερμοκρασίας
        fill: false,
        backgroundColor: 'rgba( 243, 156, 18 , 1)', // επιλογή χρώματος
        καταχώρησης σημείου θερμοκρασίας
        borderColor: 'rgba( 243, 156, 18 , 1)', // επιλογή χρώματος γραμμής
        θερμοκρασίας
        data: Tvalues, // καταχώρηση τιμής θερμοκρασίας
      },{
        label: "Humidity", // εμφάνιση ετικέτας υγρασίας
        fill: false,
        backgroundColor: 'rgba(156, 18, 243 , 1)', // επιλογή χρώματος
        καταχώρησης σημείου υγρασίας
        borderColor: 'rgba(156, 18, 243 , 1)', // επιλογή χρώματος γραμμής
        υγρασίας
        data: Hvalues, // καταχώρηση τιμής υγρασίας
      }
    ]
  },
},

```

Εικόνα 3.35: Διαμόρφωση του σώματος της web σελίδας

Στην εικόνα 3.36 που ακολουθεί αρχικά ενεργοποιείται ο τίτλος της γραφικής παράστασης καθώς επίσης επιλέγεται το πόσο ομαλή θα είναι η καμπύλη της γραμμής που ενώνει τα σημεία και το σημείο εκκίνησης της αρίθμησης του κατακόρυφου άξονα. Στην συνέχεια, εκτελείται η συνάρτηση που εμφανίζει το γράφημα στην web σελίδα με όλες τις ρυθμίσεις που περιλαμβάνει.

Τέλος, εκτελείται η συνάρτηση που είναι υπεύθυνη για την καταχώρηση των δεδομένων και την εμφάνισή τους με αυτόματη ανανέωση της web σελίδας κάθε 15 δευτερόλεπτα. Αυτό μπορεί να επιτευχθεί χρησιμοποιώντας την τεχνική AJAX (AJAX – Asynchronous JavaScript and XML). Η τεχνική AJAX είναι μία τεχνική ανάπτυξης δυναμικών ιστοσελίδων, που τις επιτρέπει να ενημερώνονται ασύγχρονα ανταλλάσσοντας μικρές ποσότητες δεδομένων με τον server στο παρασκήνιο. Αυτό επιτρέπει την ενημέρωση ορισμένων τμημάτων μιας ιστοσελίδας να πραγματοποιείται αυτόματα, χωρίς ανανέωση από τον χρήστη.

```

/* εμφάνιση τίτλου, προσαρμογή, καμπύλη μεταξύ σημείων και έναρξη άξονα y από το μηδέν*/
options:
{
  title:
  {
    display: true,
    text: "Web Server on ESP"
  },
  maintainAspectRatio: false,
  elements:
  {
    line:
    {
      tension: 0.5 //Smoothing (Curved) of data lines
    }
  },
  scales:
  {
    yAxes:
    [{
      ticks:
      {
        beginAtZero:true
      }
    }]
  }
}
});
}
/* Εμφάνιση περιοχής γραφήματος στην σελίδα */
window.onload = function()
{
  console.log(new Date().toLocaleTimeString());
};

/* Ανανέωση σελίδας και καταχώρησης δεδομένων κάθε 15 sec */
setInterval(function(){getData();}, 15000);

```

Εικόνα 3.36: Εμφάνιση γραφήματος και αυτόματη ανανέωση web σελίδας

Κλείνοντας, στην εικόνα 3.37 παρουσιάζεται η συνάρτηση που είναι υπεύθυνη για τον διαχωρισμό και την καταχώρηση των δεδομένων στην web σελίδα, τόσο υπό την μορφή γραφήματος όσο και υπό την μορφή πίνακα δεδομένων. Εφόσον υπάρχει νέο αίτημα καταχώρησης δεδομένων πραγματοποιείται ο διαχωρισμός τους και στην συνέχεια η καταχώρηση τους στις αντίστοιχες θέσεις.

```

/* Καταχώρηση δεδομένων σε μορφή γραφήματος και στο πίνακα */
function getData()
{
    var xhttp = new XMLHttpRequest(); // ορισμός μεταβλητής στην οποία καταχωρούνται τα
    νέα αιτήματα
    xhttp.onreadystatechange = function() // ενεργοποίηση συνάρτησης κάθε φορά υπάρχει
    νέο αίτημα
    {
        if (this.readyState == 4 && this.status == 200) // έλεγχος νέου αιτήματος
        {
            /* διαχωρισμός δεδομένων */
            var time = new Date().toLocaleTimeString();
            var txt = this.responseText;
            var obj = JSON.parse(txt);
            /* Καταχώρηση δεδομένων σε μορφή γραφήματος και στον πίνακα */
            PacketNo.push(obj.Packet);
            Tvalues.push(obj.Temperature);
            Hvalues.push(obj.Humidity);
            timeStamp.push(time);
            showGraph();
            var table = document.getElementById("dataTable");
            var row = table.insertRow(1);
            var cell1 = row.insertCell(0);
            var cell2 = row.insertCell(1);
            var cell3 = row.insertCell(2);
            var cell4 = row.insertCell(3);
            var cell5 = row.insertCell(4);
            cell1.innerHTML = time;
            cell2.innerHTML = obj.Packet;
            cell3.innerHTML = obj.Temperature;
            cell4.innerHTML = obj.Humidity;
            cell5.innerHTML = obj.RSSI;
        }
    };
    xhttp.open("GET", "readADC", true);
    xhttp.send();
}

</script>
</body>

</html>

)====";

```

Εικόνα 3.37: Διαχωρισμός και καταχώρηση δεδομένων σε μορφή γραφήματος και στον πίνακα

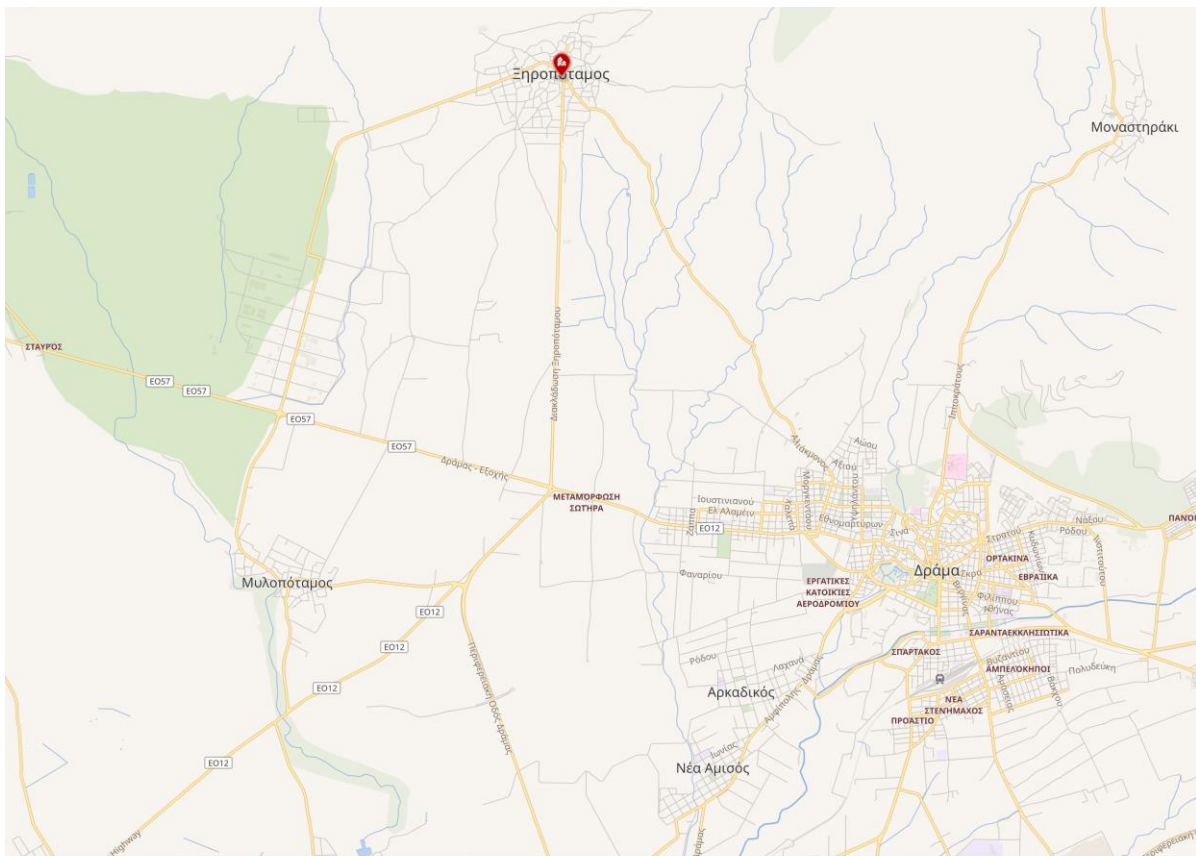
Σημείωση: Το πρόγραμμα σπάστηκε σε επί μέρους κομμάτια προκειμένου να ενσωματωθεί στην δομή της εργασίας. Ολόκληρη η δομή του προγράμματος παρουσιάζεται στο παράρτημα Δ.

3.7 Εύρος ζεύξης συσκευών

Σκοπός της εργασίας είναι η ασύρματη επικοινωνία μεταξύ της συσκευής μέτρησης και αποστολής δεδομένων με την συσκευή λήψης και καταχώρησης δεδομένων σε απόσταση 2 έως 3 χιλιομέτρων. Το εύρος ζεύξης μεταξύ των συσκευών δεν είναι σταθερό και εξαρτάται πολύ από την ύπαρξη φυσικών εμποδίων π.χ. οικίες, δέντρα, ρέματα, βουνά κτλ. Κατ' αυτόν τον τρόπο, οι μετρήσεις που πραγματοποιήθηκαν για το εύρος ζεύξης μεταξύ των συσκευών δεν δύναται να είναι οι ίδιες αν οι συσκευές τοποθετηθούν σε άλλο περιβάλλον.

Συνεπώς, η μέτρηση της εμβέλειας μεταξύ των συσκευών πραγματοποιήθηκε στην ευρύτερη περιοχή του Ξηροποτάμου Δράμας. Πιο συγκεκριμένα, ο Ξηροπόταμος είναι μία κωμόπολη του νομού Δράμας, ο οικισμός είναι χτισμένος αμφιθεατρικά στους πρόποδες του όρους Φαλακρό και βρίσκεται σε απόσταση 11 χιλιομέτρων από την πόλη της Δράμας.

Ο προσανατολισμός του χωριού είναι νοτιοδυτικός και από τα βόρεια περιβάλλεται από βουνά. Οι συντεταγμένες του οικισμού του Ξηροποτάμου σύμφωνα με το google earth είναι: $41^{\circ}11'32''\text{N}$ $24^{\circ}6'14''\text{E}$. Στην εικόνα 3.38 φαίνεται ο σχετικός χάρτης με τις συντεταγμένες του χωριού.



Εικόνα 3.38: Συντεταγμένες $41^{\circ}11'32''\text{N}$ $24^{\circ}6'14''\text{E}$

Η συσκευή λήψης και καταχώρησης δεδομένων εγκαταστάθηκε στο σημείο με συντεταγμένες: $41^{\circ}11'14''\text{N}$ $24^{\circ}6'11''\text{E}$, η θέση της ήταν σταθερή καθ' όλη την διάρκεια των μετρήσεων.

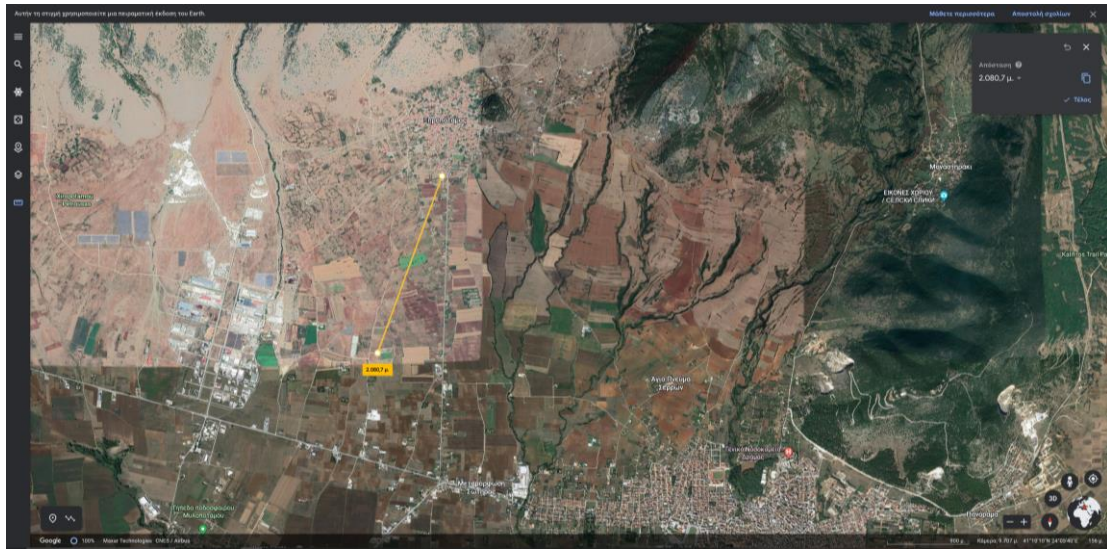
Η συσκευή μέτρησης και αποστολής δεδομένων δεν είχε σταθερή θέση, οι συντεταγμένες των σημείων στα οποία πραγματοποιήθηκαν οι μετρήσεις καθώς και η χιλιομετρική απόσταση μεταξύ των συσκευών παρουσιάζονται στον πίνακα 3.3 που ακολουθεί.

Πίνακας 3.3: Σημεία μετρήσεων

A/A	Ονομασία τοποθεσίας	Συντεταγμένες τοποθεσίας	Χιλιομετρική απόσταση
Ζεύξη 1	Γήπεδο	41°10'11"N 24°05'39"E	2080 m
Ζεύξη 2	Φωτοβολταϊκό πάρκο	41°09'38"N 24°06'10"E	2530 m
Ζεύξη 3	Κατάστημα χρωμάτων	41°09'22"N 24°06'27"E	3450 m
Ζεύξη 4	Μηχανουργείο	41°10'10"N 24°05'03"E	2940 m
Ζεύξη 5	Εξωκλήσι	41°09'55"N 24°06'44"E	2550 m

3.7.1 Ζεύξη 1 γήπεδο

Οι χιλιομετρικές αποστάσεις μετρήθηκαν σε ευθεία με την χρήση του google earth. Στην εικόνα 3.39 φαίνεται ο χάρτης μεταξύ των σημείων της ζεύξης 1, η συσκευή μέτρησης και αποστολής δεδομένων τοποθετήθηκε στο γήπεδο του Ξηροποτάμου. Η απόσταση που μεσολαβεί μεταξύ των δύο σημείων περιλαμβάνει αγροτεμάχια αλλά και εγκαταστάσεις συλών. Παρ' όλα αυτά, η αποστολή και η λήψη των δεδομένων πραγματοποιήθηκε επιτυχώς.



Εικόνα 3.39: Ζεύξη 1 γήπεδο

Οι ρυθμίσεις των πομπό – δεκτών LoRa για την πραγματοποίηση της ζεύξης 1 παρουσιάζονται στον παρακάτω πίνακα 3.4.

Πίνακας 3.4: Ρυθμίσεις ζεύξης 1

Μεταβλητές	Τιμές
Εύρος ζώνης (BW)	62.5kHz
Παράγοντας διασποράς (SF)	7
Κωδικοποίηση πακέτου (CR)	5

3.7.2 Ζεύξη 2 φωτοβολταϊκό πάρκο

Στην εικόνα 3.40 παρουσιάζεται ο χάρτης μεταξύ των σημείων της ζεύξης 2, το σημείο που τοποθετήθηκε η συσκευή μέτρησης και αποστολής δεδομένων είναι ένα φωτοβολταϊκό πάρκο. Η απόσταση μεταξύ των συσκευών περιλαμβάνει παρεμβολές κτηρίων και φυσικών εμποδίων καθώς επίσης και μη οπτική επαφή μεταξύ των συσκευών, καθώς το φωτοβολταϊκό πάρκο βρίσκεται σε πλαγιά και δεν είναι εφικτή η οπτική επαφή.



Εικόνα 3.40: Ζεύξη 2 φωτοβολταϊκό πάρκο

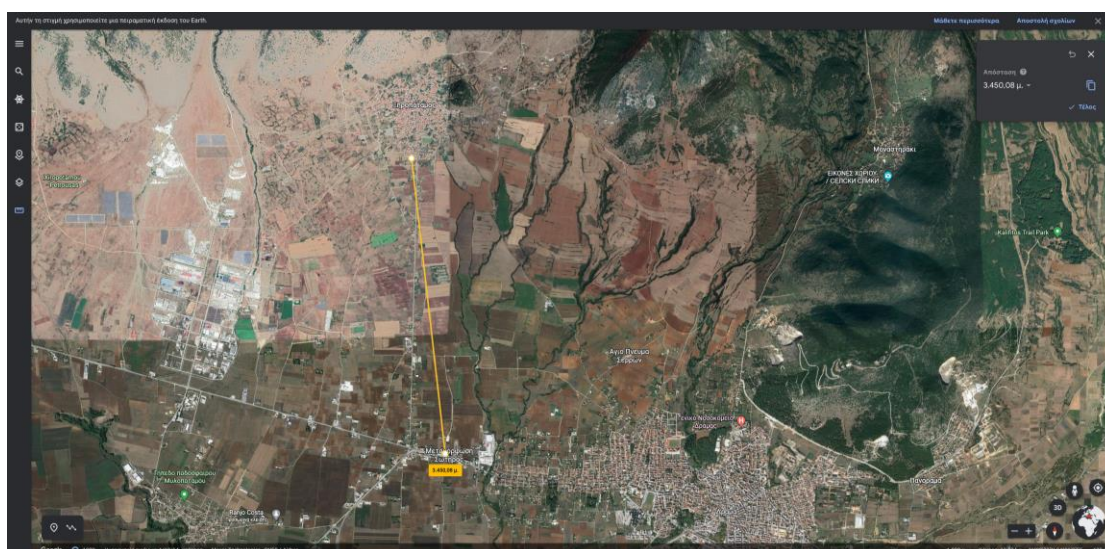
Όλα τα παραπάνω επηρέασαν αρνητικά ως προς την επικοινωνία μεταξύ των συσκευών. Οι ρυθμίσεις των μεταβλητών της ζεύξης 2 περιλαμβάνουν τις μέγιστες τιμές οι οποίες τέθηκαν σε εφαρμογή και απεικονίζονται στον πίνακα 3.5, χωρίς ωστόσο να υπάρξει επιτυχής ζεύξη.

Πίνακας 3.5: Ρυθμίσεις ζεύξης 2

Μεταβλητές	Τιμές
Εύρος ζώνης (BW)	20.8kHz
Παράγοντας διασποράς (SF)	12
Κωδικοποίηση πακέτου (CR)	8

3.7.3 Ζεύξη 3 κατάστημα χρωμάτων

Στην εικόνα 3.41 απεικονίζεται ο χάρτης μεταξύ των σημείων της ζεύξης 3, το σημείο που τοποθετήθηκε η συσκευή μέτρησης και αποστολής δεδομένων είναι ένα κατάστημα χρωμάτων της περιοχής.



Εικόνα 3.41: Ζεύξη 3 κατάστημα χρωμάτων

Κεφάλαιο 3

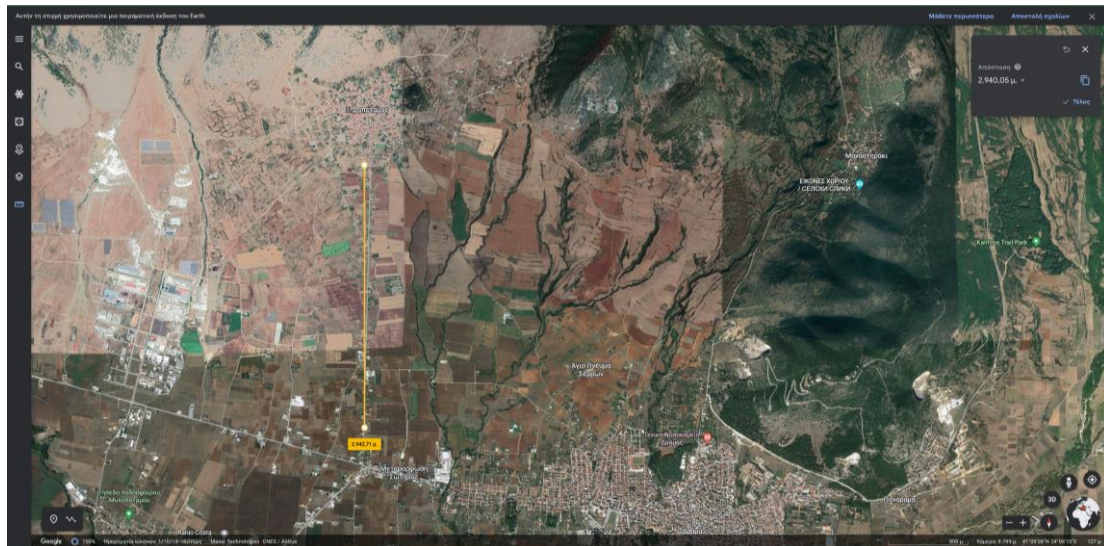
Πρόκειται για την πιο απομακρυσμένη ζεύξη που πραγματοποιήθηκε με επιτυχία, η απόσταση μεταξύ των δύο συσκευών περιλαμβάνει παρεμβολές κτηρίων και φυσικών εμποδίων. Η επικοινωνία μεταξύ των συσκευών δεν ήταν εφικτή όταν η συσκευή μέτρησης και αποστολής δεδομένων τοποθετούταν στον εσωτερικό χώρο του καταστήματος χρωμάτων. Γι' αυτόν τον λόγο οι μετρήσεις πραγματοποιήθηκαν στον εξωτερικό χώρο του κτηρίου. Οι ρυθμίσεις του παράγοντα διασποράς, η κωδικοποίηση των πακέτων καθώς επίσης και η ισχύς εκπομπής ήταν οι μέγιστες δυνατές, όπως παρουσιάζονται στον παρακάτω πίνακα 3.6.

Πίνακας 3.6: Ρυθμίσεις ζεύξης 3

Μεταβλητές	Τιμές
Εύρος ζώνης (BW)	20.8kHz
Παράγοντας διασποράς (SF)	12
Κωδικοποίηση πακέτου (CR)	8

3.7.4 Ζεύξη 4 μηχανουργείο

Στην εικόνα 3.42 παρουσιάζεται ο χάρτης μεταξύ των σημείων της ζεύξης 4, το σημείο που τοποθετήθηκε η συσκευή μέτρησης και αποστολής δεδομένων είναι ένα μηχανουργείο της περιοχής.



Εικόνα 3.42: Ζεύξη 4 μηχανουργείο

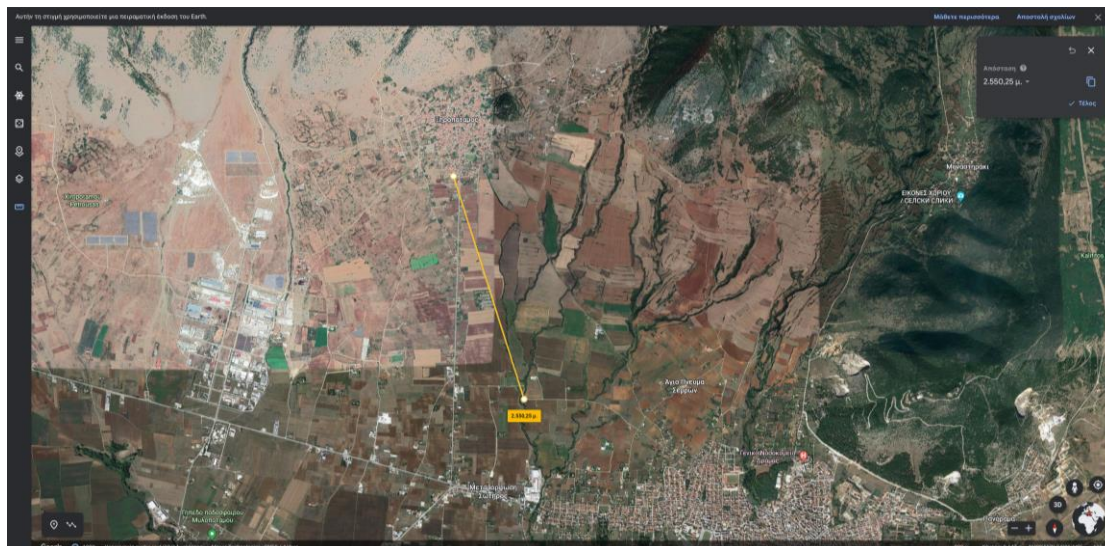
Στην απόσταση μεταξύ των συσκευών μεσολαβούν διάφοροι οικισμοί που αρχικά δυσχεραίνουν την αναμετάδοση δεδομένων. Η επικοινωνία μεταξύ των συσκευών δεν ήταν εφικτή όταν η συσκευή μέτρησης και αποστολής δεδομένων τοποθετούταν στον εσωτερικό χώρο του καταστήματος του μηχανουργείου. Γι' αυτόν τον λόγο οι μετρήσεις πραγματοποιήθηκαν στον εξωτερικό χώρο του κτηρίου. Οι ρυθμίσεις του παράγοντα διασποράς, η κωδικοποίηση των πακέτων καθώς επίσης και η ισχύς εκπομπής κατά την πραγματοποίηση της ζεύξης ήταν οι μέγιστες δυνατές και παρουσιάζονται στον πίνακα 3.7.

Πίνακας 3.7: Ρυθμίσεις ζεύξης 4

Μεταβλητές	Τιμές
Εύρος ζώνης (BW)	20.8kHz
Παράγοντας διασποράς (SF)	12
Κωδικοποίηση πακέτου (CR)	8

3.7.5 Ζεύξη 5 εξωκλήσι

Τέλος, στην παρακάτω εικόνα 3.43 παρουσιάζεται ο χάρτης μεταξύ των συσκευών της ζεύξης 5, το σημείο που τοποθετήθηκε η συσκευή μέτρησης και αποστολής δεδομένων είναι ένα εξωκλήσι της περιοχής.



Εικόνα 3.43: Ζεύξη 5 εξωκλήσι

Η απόσταση μεταξύ των συσκευών περιλαμβάνει παρεμβολές φυσικών εμποδίων όπως δέντρα. Στον πίνακα 3.8 που ακολουθεί φαίνονται οι μέγιστες δυνατές ρυθμίσεις με τις οποίες η επικοινωνία μεταξύ των συσκευών ήταν εφικτή.

Πίνακας 3.8: Ρυθμίσεις ζεύξης 5

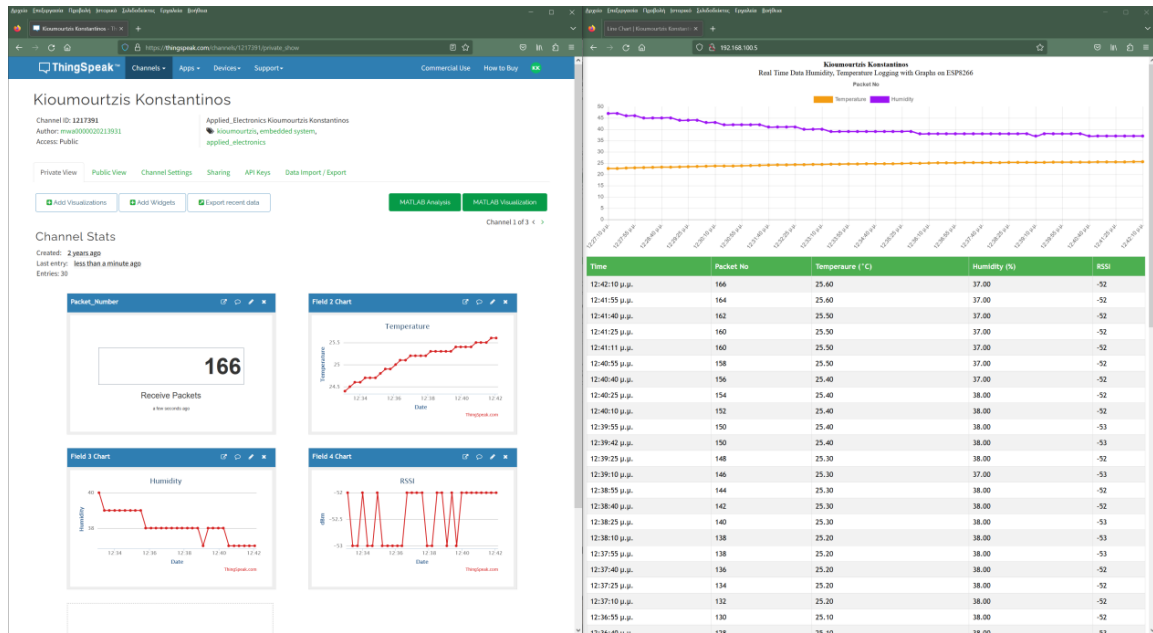
Μεταβλητές	Τιμές
Εύρος ζώνης (BW)	20.8kHz
Παράγοντας διασποράς (SF)	12
Κωδικοποίηση πακέτου (CR)	8

3.8 Καταχώρηση δεδομένων στον τοπικό Server και στο ThingSpeak

Τα δεδομένα που λαμβάνονται από την συσκευή λήψης και καταχώρησης δεδομένων εφόσον επεξεργαστούν αποστέλλονται μέσω Wi – Fi στον τοπικό server και στην πλατφόρμα του ThingSpeak. Η καταχώρηση των δεδομένων στο ThingSpeak γίνεται στο δημόσιο κανάλι το οποίο βρίσκεται στον παρακάτω σύνδεσμο:

<https://thingspeak.com/channels/1217391>

Η καταχώρηση των δεδομένων στον τοπικό server γίνεται στην IP που εμφανίζεται στην OLED οθόνη της συσκευής. Οι καταχωρήσεις γίνονται με την τεχνική get – post. Ανοίγοντας τον σύνδεσμο του καναλιού στο ThingSpeak απεικονίζονται 4 fields όπως φαίνεται στην εικόνα που ακολουθεί 3.44. Στο πρώτο field απεικονίζεται ο αριθμός του πακέτου δεδομένων που καταχωρήθηκε τελευταία καθώς επίσης και την χρονική διάρκεια της καταχώρησης. Στο δεύτερο field απεικονίζεται σε γραφική παράσταση η αυξομείωση της τιμής της θερμοκρασίας, ενώ στο τρίτο field απεικονίζεται σε γραφική παράσταση η αυξομείωση της τιμής της υγρασίας. Τέλος, στο τέταρτο field απεικονίζεται σε γραφική παράσταση η αυξομείωση της τιμής της ισχύος του λαμβανόμενου σήματος σε dBm.



Εικόνα 3.44: Δομή καναλιού στο thingspeak και στον τοπικό server

Πληκτρολογώντας την IP που εμφανίζεται στην OLED οθόνη της συσκευής ανοίγει η web σελίδα του server. Στο επάνω μέρος της σελίδας εμφανίζονται πληροφορίες που κατευθύνουν τον χρήστη στο να αντιληφθεί την λειτουργία της σελίδας. Έπειτα ακολουθεί ο χώρος του γραφήματος μέσα στον οποίο καταχωρούνται οι τιμές της θερμοκρασίας και της υγρασίας συναρτήσει του χρόνου. Τέλος, στο κάτω μέρος της σελίδας υπάρχει ο πίνακας καταχωρήσεων ο οποίος περιλαμβάνει πέντε στήλες μέσα στις οποίες καταχωρούνται τα αντίστοιχα δεδομένα όπως φαίνεται στην εικόνα 3.44.

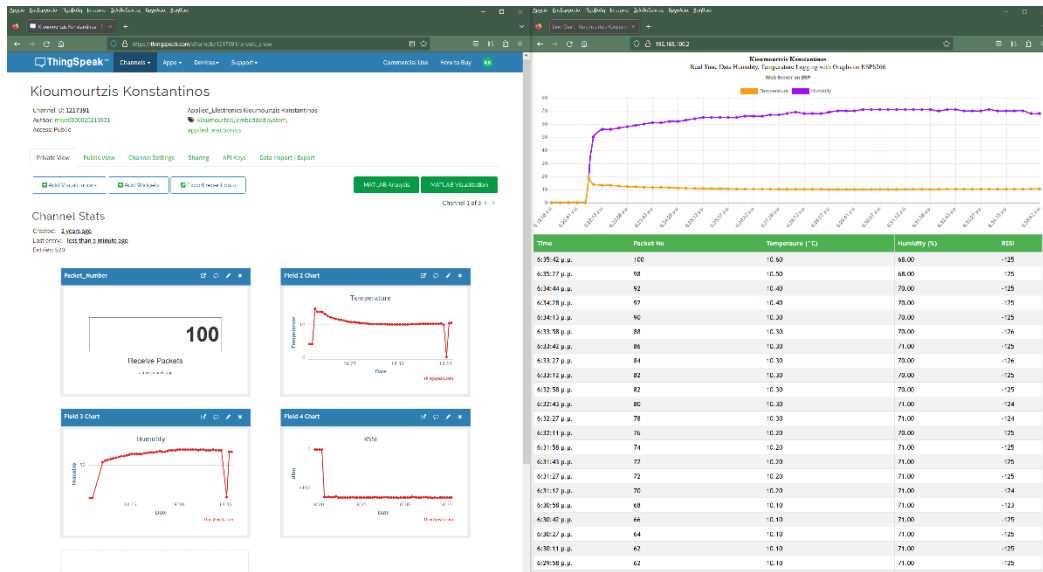
3.8.1 Καταχώρηση δεδομένων ζεξής 1

Στην εικόνα 3.45 παρουσιάζεται η συσκευή μέτρησης στην τοποθεσία του γηπέδου κατά την διάρκεια των μετρήσεων και της αποστολής δεδομένων, η απόσταση που απέχει από την συσκευή λήψης και καταχώρησης είναι 2080 μέτρα.



Εικόνα 3.45: Συσκευή μέτρησης στην περιοχή του γηπέδου σε απόσταση 2080 m

Συνολικά, κατά την διάρκεια της σύνδεσης στάλθηκαν 100 πακέτα δεδομένων τα οποία ελήφθησαν από την συσκευή λήψης και καταχωρήθηκαν τόσο τοπικά όσο και στην πλατφόρμα του ThingSpeak επιτυχώς όπως φαίνεται στην εικόνα 3.46. Κατά την διάρκεια της ζεξής τα λαμβανόμενα σήματα ήταν αρκετά εξασθενημένα καθώς ο δείκτης RSSI ήταν -125dBm , ενώ η θερμοκρασία ήταν 10°C και η υγρασία ήταν 68%.



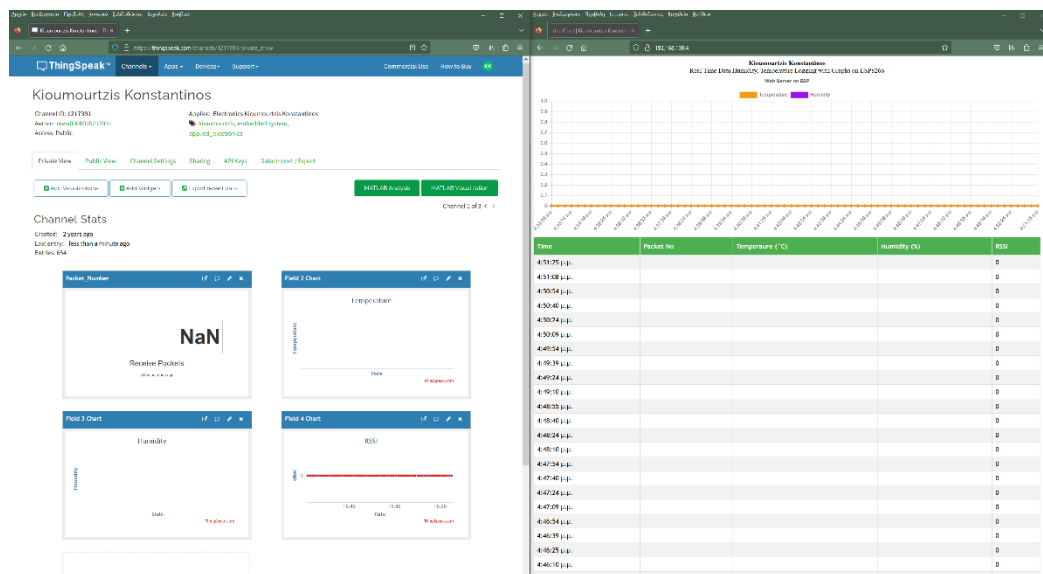
Εικόνα 3.46: Καταχώρηση δεδομένων ζεύξης 1

3.8.2 Καταχώρηση δεδομένων ζεύξης 2

Στην εικόνα 3.47 απεικονίζεται η συσκευή μέτρησης στην τοποθεσία του φωτοβολταϊκού πάρκου κατά την διάρκεια των μετρήσεων, η απόσταση που απέχει από την συσκευή λήξης είναι 2530 μέτρα. Όπως αναλύθηκε και στην υπό – ενότητα 3.7.2 η τοποθεσία του φωτοβολταϊκού πάρκου δεν επιτρέπει την οπτική επαφή μεταξύ των συσκευών με αποτέλεσμα να μην υπάρξει ανταλλαγή δεδομένων. Στην εικόνα 3.48 φαίνεται η μη επικοινωνία μεταξύ των συσκευών, στο field 1 δεν υπάρχει λήψη πακέτων, τα fields 2 και 3 δεν περιλαμβάνουν καταχωρήσεις, ενώ στο field 4 δείχνει την τιμή 0dBm καθώς η συσκευή λήξης δεν λάμβανε σήματα.



Εικόνα 3.47: Συσκευή μέτρησης στην περιοχή του φωτοβολταϊκού πάρκου σε απόσταση 2530 m



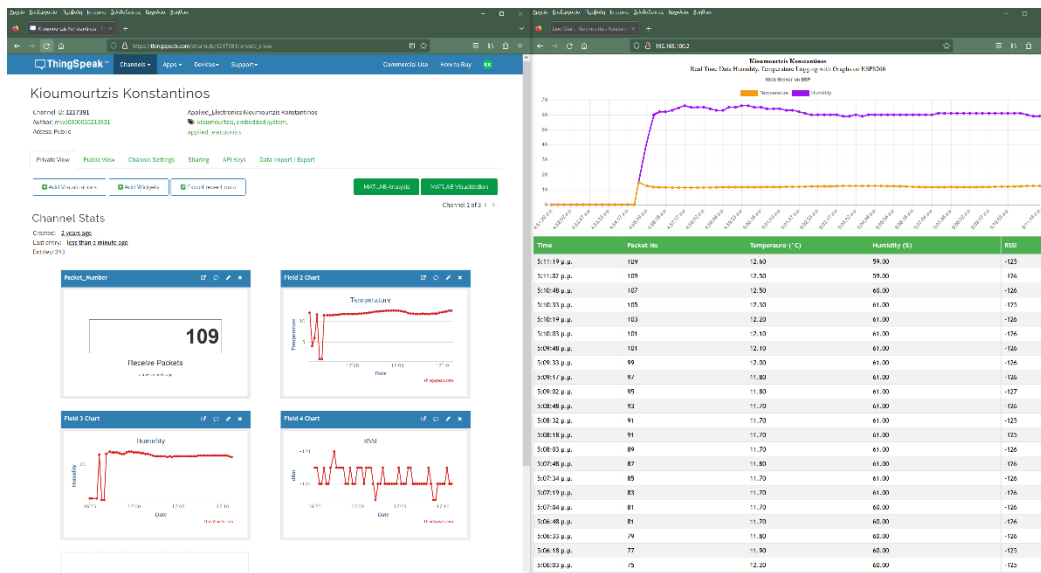
Εικόνα 3.48: Καταχώρηση δεδομένων ζεύξης 2

3.8.3 Καταχώρηση δεδομένων ζεύξης 3

Στην εικόνα 3.49 φαίνεται η συσκευή μέτρησης στην τοποθεσία του καταστήματος χρωμάτων κατά την διάρκεια των μετρήσεων, η απόσταση που απέχει από την συσκευή λήψης είναι 3450 μέτρα. Πρόκειται για την πιο απομακρυσμένη απόσταση ζεύξης που πραγματοποιήθηκε επιτυχής επικοινωνία. Στην εικόνα 3.50 απεικονίζονται τα δεδομένα που καταχωρήθηκαν. Όπως φαίνεται κατά την διάρκεια της ζεύξης στάλθηκαν 109 πακέτα δεδομένων, η ισχύς των λαμβανόμενων σημάτων ήταν εξασθενημένη στα -125dBm , ενώ η θερμοκρασία ήταν 12.6°C και η υγρασία ήταν 59%.



Εικόνα 3.49: Συσκευή μέτρησης στην περιοχή του καταστήματος χρωμάτων σε απόσταση 3450 m



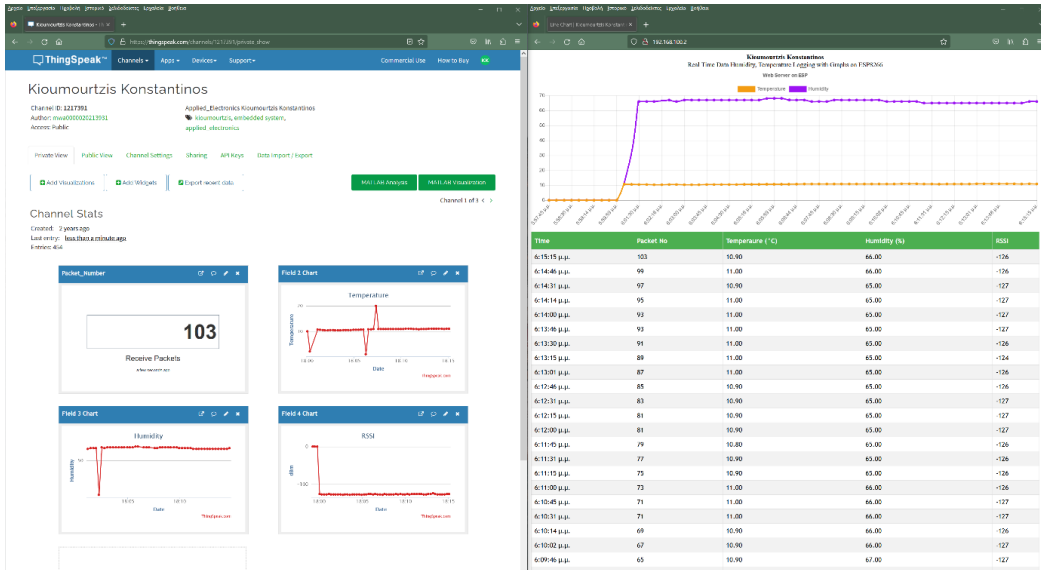
Εικόνα 3.50: Καταχώρηση δεδομένων ζεύξης 3

3.8.4 Καταχώρηση δεδομένων ζεύξης 4

Στην εικόνα 3.51 φαίνεται η συσκευή μέτρησης στην τοποθεσία του μηχανουργείου κατά την διάρκεια των μετρήσεων, η απόσταση που απέχει από την συσκευή λήψης είναι 2940 μέτρα. Συνολικά, κατά την διάρκεια της σύνδεσης στάλθηκαν 103 πακέτα δεδομένων τα οποία ελήφθησαν από την συσκευή λήψης και καταχωρήθηκαν επιτυχώς όπως φαίνεται στην εικόνα 3.52. Κατά την διάρκεια της ζεύξης η ισχύς των λαμβανόμενων σημάτων ήταν -126dBm , ενώ η θερμοκρασία ήταν 10.9°C και η υγρασία ήταν 66%.



Εικόνα 3.51: Συσκευή μέτρησης στην περιοχή του μηχανουργείου σε απόσταση 2940 m



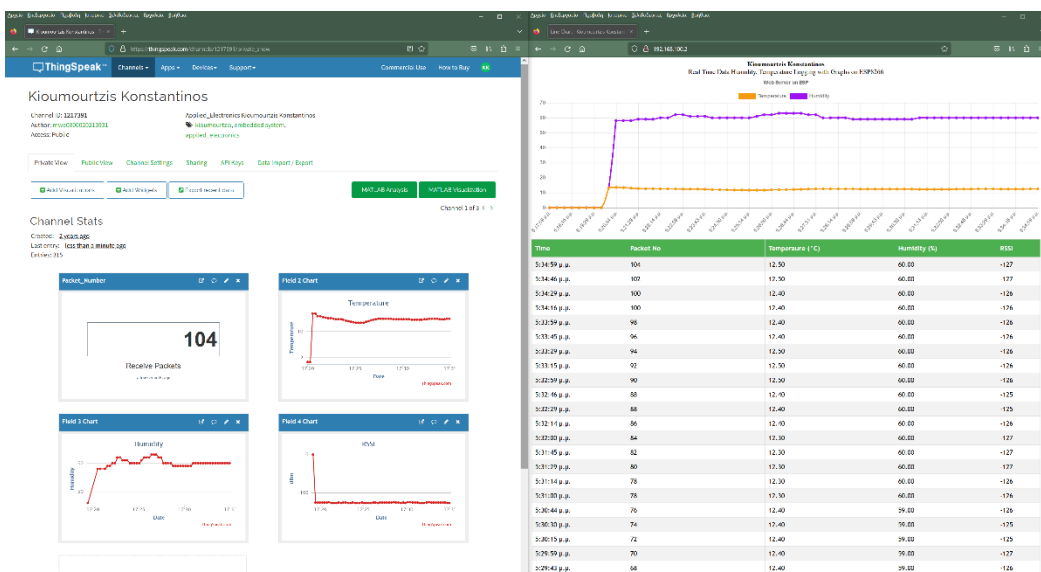
Εικόνα 3.52: Καταχώρηση δεδομένων ζεύξης 4

3.8.5 Καταχώρηση δεδομένων ζεύξης 5

Στην εικόνα 3.53 φαίνεται η συσκευή μέτρησης στην τοποθεσία του εξωκκλησιού κατά την διάρκεια των μετρήσεων, η απόσταση που απέχει από την συσκευή λήψης είναι 2550 μέτρα. Συνολικά, κατά την διάρκεια της σύνδεσης στάλθηκαν 104 πακέτα δεδομένων τα οποία ελήφθησαν από την συσκευή λήψης και καταχωρήθηκαν επιτυχώς όπως φαίνεται στην εικόνα 3.54. Κατά την διάρκεια της ζεύξης τα λαμβανόμενα σήματα ήταν αρκετά εξασθενημένα καθώς ο δείκτης RSSI ήταν -127dBm, ενώ η θερμοκρασία ήταν 12.5°C και η υγρασία ήταν 60%.



Εικόνα 3.53: Συσκευή μέτρησης στην περιοχή του εξωκκλησιού σε απόσταση 2550 m



Εικόνα 3.54: Καταχώρηση δεδομένων ζεύξης 5

3.9 Επίλογος

Ο σκοπός αυτού το κεφαλαίου είναι να αναδείξει την ανάπτυξη και υλοποίηση των μονάδων του συστήματος οι οποίες είναι σχεδιασμένες για να επικοινωνούν σε αποστάσεις 2 έως 3 χιλιομέτρων. Συνεπώς, το σύστημα απαρτίζεται από δύο μονάδες την μονάδα μετρήσεων, ένδειξης και αποστολής δεδομένων και την μονάδα λήψης και καταχώρησης δεδομένων σε τοπικό server και στο ThingSpeak.

Η μονάδα μετρήσεων, ένδειξης και αποστολής δεδομένων περιλαμβάνει έναν αισθητήρα θερμοκρασίας και υγρασίας (DHT11), έναν μικροεπεξεργαστή (ESP8266), μία OLED οθόνη και τον πομπό – δέκτη LoRa. Η μονάδα σχεδιάστηκε για να είναι φορητή, αυτό έχει ως αποτέλεσμα να καταναλώνει χαμηλή ηλεκτρική ισχύ και να προσδίδει μεγάλη αυτονομία. Ο τρόπος λειτουργίας της μονάδας είναι ο εξής, αρχικά ο αισθητήρας θερμοκρασίας μετράει την θερμοκρασία και την υγρασία στον περιβάλλοντα χώρο που βρίσκεται η συσκευή και στην συνέχεια στέλνει τα δεδομένα που προκύπτουν στον μικροεπεξεργαστή. Ο μικροεπεξεργαστής με την σειρά του εφόσον λάβει και επεξεργαστεί τα δεδομένα τα αναπαράγει στην OLED οθόνη προκειμένου να υπάρχει τοπική ένδειξη των μετρήσεων. Ταυτόχρονα με την αναπαραγωγή των δεδομένων στην οθόνη, αποστέλλονται μέσω του πομπό – δέκτη στην συσκευή λήψης η οποία βρίσκεται σε ακτίνα 2 έως 3 χιλιομέτρων μακριά.

Η μονάδα λήψης και καταχώρησης δεδομένων αποτελείται από έναν πομπό – δέκτη LoRa, μία OLED οθόνη και έναν μικροεπεξεργαστή (ESP8266). Ο σχεδιασμός της μονάδας έγινε ώστε να είναι φορητή και να προσδίδει μεγάλη αυτονομία. Ο τρόπος λειτουργίας της συσκευής είναι ο εξής, εφόσον ο πομπό – δέκτης λάβει τα δεδομένα από την μονάδα μετρήσεων, στέλνει τα δεδομένα αυτά στον μικροεπεξεργαστή, ο οποίος με την σειρά του τα επεξεργάζεται και μέσω Wi – Fi σύνδεσης τα καταχωρεί στον τοπικό server και στην πλατφόρμα του ThingSpeak.

Η web σελίδα του τοπικού server δεν μπορεί να φαίνεται δημόσια καθώς απαιτείται η αγορά DNS διεύθυνσης μέσα από την οποία θα μπορούσε η web σελίδα να φαίνεται δημόσια σε κάθε συσκευή. Για τον λόγο αυτό, τα δεδομένα καταχωρούνται ταυτόχρονα τόσο στην σελίδα που βρίσκεται εγκατεστημένη στον τοπικό server όσο και στην πλατφόρμα του ThingSpeak. Μ' αυτόν τον τρόπο τα δεδομένα μπορούν να είναι διαθέσιμα σε οποιαδήποτε συσκευή είναι συνδεδεμένη με το δίκτυο.

Κατά το κατασκευαστικό μέρος της εργασίας, δημιουργήθηκαν πρώτα πειραματικά οι δύο μονάδες με σκοπό να αποδειχθεί η εύρυθμη λειτουργία τους. Στην συνέχεια πραγματοποιήθηκε το κατασκευαστικό μέρος των δύο μονάδων, έγιναν έλεγχοι (continuity tests) για να διαπιστωθεί η ορθότητα των συνδέσεων. Επί προσθέτου, σχεδιάστηκαν και υλοποιήθηκαν με 3d εκτύπωση ειδικά πλαίσια τα οποία περιλαμβάνουν τα ηλεκτρονικά μέρη του συστήματος. Τέλος, αναπτύχθηκαν κατάλληλοι κώδικες προγράμματος προκειμένου να επιτευχθεί ο σκοπός της εργασίας που είναι η μέτρηση, η ένδειξη η αποστολή και η λήψη δεδομένων σε αποστάσεις 2 έως 3 χιλιόμετρα και η καταχώρηση των δεδομένων μέτρησης σε τοπικό server και στην πλατφόρμα του ThingSpeak.

Στην συνέχεια οι μονάδες τέθηκαν σε λειτουργία υπό πραγματικές συνθήκες προκειμένου να διαπιστωθεί αν επιτυγχάνεται ο σκοπός της εργασίας. Ως εκ τούτου, ορίστηκαν πέντε τοποθεσίες οι οποίες σύμφωνα με τους χάρτες google earth απείχαν σε ευθεία 2 με 3 χιλιόμετρα από την τοποθεσία που βρισκόταν η μονάδα λήψης και καταχώρησης δεδομένων.

Εν κατακλείδι, αποδεικνύεται ότι ο σκοπός της εργασίας επιτυγχάνεται με τις μονάδες αυτές, καθώς υπήρξε αποστολή και λήψη δεδομένων σε απόσταση ίση με 3450 μέτρα. Να σημειωθεί ότι οι μονάδες δεν μπορούν να επιτύχουν τέτοια εμβέλεια εντός κατοικημένων περιοχών καθώς τα φυσικά εμπόδια όπως επίσης και τα κτήρια δεν επιτρέπουν την αποστολή και λήψη δεδομένων σε τόσες μεγάλες αποστάσεις.

Κεφάλαιο 4: Συμπεράσματα

4.1 Εισαγωγή

Σκοπός του κεφαλαίου τούτου είναι η καταγραφή των συμπερασμάτων που προέκυψαν κατά την εκπόνηση της εργασίας. Επίσης, αποτυπώνονται προβλήματα που προέκυψαν κατά την εφαρμογή των συστημάτων και πως αυτά επιλύθηκαν. Τέλος, καταλήγει στις εφαρμογές που μπορούν να βρουν τα συστήματα αυτά.

4.2 Συμπεράσματα

Η τεχνολογία LoRa επιτρέπει την σύνδεση των συσκευών στο δίκτυο σε μεγάλες αποστάσεις, η ονομασία LoRa προέρχεται από την συντομογραφία «μεγάλη εμβέλεια (Long Range)». Χρησιμοποιεί τεχνολογία ασύρματης διαμόρφωσης σήματος Chirp Spread Spectrum – C.S.S.. Το Chirp είναι ένα σήμα του οποίου η συχνότητα είτε αυξάνεται (up – chirp), είτε μειώνεται (down – chirp) συναρτήσει του χρόνου.

Τα δίκτυα LoRa αποτελούνται από 4 επίπεδα τα οποία είναι το επίπεδο εφαρμογής (LoRa Application Layer), το επίπεδο ελέγχου προσπέλασης μέσω (LoRa M.A.C. Layer), το φυσικό επίπεδο (LoRa PHY Layer) και το επίπεδο ραδιοσυχνότητας (LoRa RF Layer). Στην Ευρώπη τα δίκτυα LoRa χρησιμοποιούν τις I.S.M. ζώνες συχνοτήτων, αυτό έχει ως αποτέλεσμα να λειτουργούν στις μάντες συχνοτήτων 867 – 869 MHz, ενώ σε ορισμένες χώρες που ανήκουν στην ευρωπαϊκή ένωση επιτρέπεται η χρησιμοποίηση της ζώνης συχνοτήτων των 433MHz, σ' αυτές τις χώρες συμπεριλαμβάνεται και η Ελλάδα. Ανάλογα με την ζώνη συχνοτήτων που λειτουργεί ένα δίκτυο LoRa επιβάλλονται ορισμένοι περιορισμοί που θα πρέπει να λαμβάνονται υπ' όψη κατά τον σχεδιασμό των δικτύων αυτών.

Ο παράγοντας διασποράς αποτελεί ένα πολύ βασικό μέγεθος σ' αυτού του είδους τα συστήματα, στην ουσία η τιμή του προσδιορίζει τον ρυθμό των κωδικοποιημένων bits ανά σύμβολο. Επίσης, η αύξηση του παράγοντα διασποράς οδηγεί σε αύξηση της ευαισθησίας του δέκτη μεγάλωνοντας μ' αυτόν τον τρόπο την ακτίνα κάλυψης του συστήματος. Ωστόσο η επιλογή μεγάλης τιμής του παράγοντα διασποράς μεγάλωνει την πιθανότητα μετάδοσης λάθος πληροφορίας. Τέλος, η τιμή του παράγοντα διασποράς πρέπει να είναι γνωστή τόσο στον πομπό, όσο και στον δέκτη. Οι τεχνικές κωδικοποίησης εξασφαλίζουν την μετάδοση της πληροφορίας προκειμένου να επιτευχθεί η αξιόπιστη μεταφορά τους, επιγραμματικά οι τεχνικές που χρησιμοποιούνται στην διαμόρφωση LoRa είναι η Data Whitening, η Gray Indexing, η Forward Error Correction και η Interleaving.

Κατά την δοκιμασία των συσκευών σε πραγματικές συνθήκες διαπιστώθηκε ότι προκειμένου να υπάρχει επικοινωνία και ανταλλαγή δεδομένων μεταξύ των συσκευών θα πρέπει να υπάρχει οπτική επαφή. Διάφορα φυσικά εμπόδια καθώς και κτηριακές εγκαταστάσεις δυσχεραίνουν την μεταξύ επικοινωνία των συσκευών. Αυτό γίνεται εύκολα αντιληπτό καθώς από τις πέντε τοποθεσίες που πραγματοποιήθηκαν οι μετρήσεις στις τέσσερις τοποθεσίες έγινε επιτυχημένα η αποστολή και η λήψη των δεδομένων. Η τοποθεσία στην οποία δεν πραγματοποιήθηκε ζεύξη όπως διαπιστώνεται δεν υπήρχε οπτική επαφή ανάμεσα στον πομπό και τον δέκτη.

Αν και ο σκοπός της εργασίας ήταν η μετάδοση δεδομένων με χρήση LoRa επικοινωνίας σε αποστάσεις 2 έως 3 χιλιομέτρων, εν τέλει πραγματοποιήθηκε επιτυχημένη μετάδοση δεδομένων σε απόσταση 3450 μέτρων. Αυτό σημαίνει ότι οι μονάδες έχουν την δυνατότητα για την μετάδοση δεδομένων στις αποστάσεις που τέθηκαν εξ αρχής, αρκεί να μην μεσολαβούν ανάμεσα στις συσκευές κτηριακές εγκαταστάσεις και εμπόδια που δεν επιτρέπουν την οπτική επαφή τους. Συνεπώς, εντός

κατοικημένων περιοχών η μετάδοση δεδομένων δεν μπόρεσε να πραγματοποιηθεί για αποστάσεις μεγαλύτερες των 200 μέτρων.

Η συσκευή λήψης είναι υπεύθυνη για την καταχώρηση των δεδομένων που λαμβάνει σε ιστοσελίδα και στην πλατφόρμα του ThingSpeak. Γι' αυτόν τον λόγο, η συσκευή λήψης λειτουργεί και ως τοπικός server καταχωρώντας ταυτόχρονα τα δεδομένα τόσο τοπικά όσο και δημόσια στην πλατφόρμα ThingSpeak. Εν κατακλείδι, οι προς υλοποίηση συσκευές επιτυγχάνουν τον σκοπό της εργασίας που είναι η μετάδοση δεδομένων σε αποστάσεις 2 έως 3 χιλιομέτρων με την χρήση LoRa επικοινωνίας, μόνο όταν το περιβάλλον στο οποίο χρησιμοποιούνται δεν είναι αστικό.

4.3 Επίλυση προβλημάτων

Αρχικά το πρόβλημα που υφίσταντο, ήταν πως κατά την υλοποίηση της πειραματικής κατασκευής διαπιστώθηκε η λήψη παρεμβολών, αυτό είχε ως αποτέλεσμα την αρνητική επίδραση στην διαδικασία της καταχώρησης των δεδομένων, καθώς εμφανίζονταν διάφορα σύμβολα. Η επίλυση αυτού του προβλήματος πραγματοποιήθηκε θέτοντας μέσω του προγράμματος μέγιστη τιμή κωδικοποίησης των πακέτων, κατ' αυτόν τον τρόπο το φαινόμενο εξαλείφθηκε.

Ένα ακόμη μείζον πρόβλημα που υπήρξε, ήταν ότι αρχικά υπήρχε ζεύξη μεταξύ των συσκευών για αποστάσεις μικρότερες των 1000 μέτρων. Η επίλυση αυτού του προβλήματος διήρκησε για μεγάλο χρονικό διάστημα καθώς υπήρχε δυσκολία στον εντοπισμό του προβλήματος. Εν τέλη, το πρόβλημα επιλύθηκε μέσω του προγράμματος με τις αλλαγές των προεπιλεγμένων τιμών του παράγοντα διασποράς και του εύρους ζώνης στις μέγιστες δυνατές. Το γεγονός αυτό επέτρεψε να υπάρξει μέγιστη μετάδοση δεδομένων σε απόσταση 3450 μέτρων.

Ένα ακόμη σοβαρό πρόβλημα που υπήρξε κατά την υλοποίηση του συστήματος ήταν πως η ιστοσελίδα που βρίσκεται στον server δεν μπορεί να φανεί δημόσια από οποιαδήποτε συσκευή. Συνεπώς, η ιστοσελίδα φαίνεται σε όσες συσκευές βρίσκονται συνδεδεμένες στο ίδιο Wi – Fi δίκτυο. Οι ενέργειες που πραγματοποιήθηκαν για την επίλυση του προβλήματος ήταν να πραγματοποιηθεί port forwarding στον router, προκειμένου να υπάρξει πρόσβαση στην συσκευή. Παρ' όλα αυτά η μέθοδος αυτή δεν λειτούργησε γι' αυτόν τον λόγο τα δεδομένα καταχωρούνται ταυτόχρονα τόσο στην ιστοσελίδα του τοπικού server όσο και στην πλατφόρμα ThingSpeak προκειμένου να παρουσιάζονται δημόσια προς οποιαδήποτε συσκευή.

4.4 Εφαρμογές συστήματος

Η ευελιξία που παρέχουν οι συσκευές επιτρέπει την χρησιμοποίησή τους σε μεγάλο εύρος δραστηριοτήτων. Πιο συγκεκριμένα οι συσκευές μπορούν να χρησιμοποιηθούν ως μετρητικές συσκευές φυσικών μεγεθών παρέχοντας έτσι στον χρήστη άμεση πληροφόρηση για τα προς μέτρηση φυσικά μεγέθη. Ένα παράδειγμα εφαρμογής του συστήματος είναι οι έξυπνες καλλιέργειες, με τροποποιήσεις το σύστημα μπορεί να μετράει την υγρασία του εδάφους, όπως επίσης και διάφορα χαρακτηριστικά και να ενημερώνει τον χρήστη για την λίπανση ή το πότισμα του αγροτεμαχίου.

Ένα επίσης παράδειγμα εφαρμογής στο οποίο οι συσκευές μπορούν να χρησιμοποιηθούν είναι η κάλυψη αναγκών σε μία δασική έκταση, με τις κατάλληλες τροποποιήσεις του συστήματος μπορεί να φέρει διατάξεις που να καταμετρούν διάφορα χαρακτηριστικά που έχουν σχέση με την χλωρίδα ενός δάσους. Τα συστήματα αυτά μπορούν να βρίσκονται διάσπαρτα εγκατεστημένα μέσα στο δάσος (τερματικές συσκευές) η κάθε συσκευή μπορεί να στέλνει τα δεδομένα μέσω μιας gateway συσκευής σε έναν server μέσα από τον οποίο θα είναι δυνατή η ανάγνωση των δεδομένων για διάφορα φαινόμενα όπως πυρκαγιές κ.ο.κ..

Βιβλιογραφία

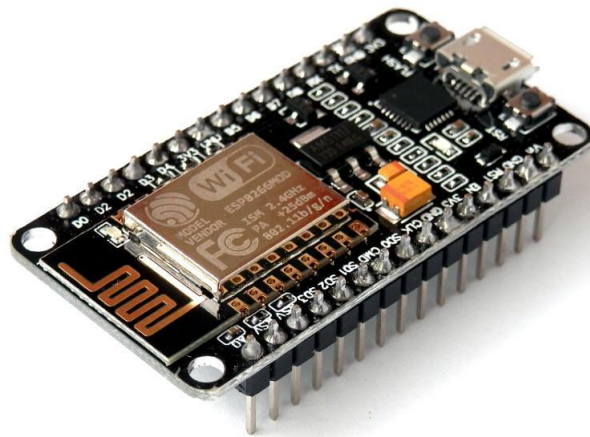
- Γ. Μπαγκατζούνης, «DSpace,» 3 2020. [Ηλεκτρονικό]. Available:
[1] https://dspace.uowm.gr/xmlui/bitstream/handle/123456789/2329/dipl_mpagkatzoynis_george_529.pdf?sequence=1&isAllowed=y. [Πρόσβαση 1 2023].
- Β. Θεόδωρος, «telematics.upatras,» 4 2022. [Ηλεκτρονικό]. Available:
[2] http://telematics.upatras.gr/telematics/system/files/bouras_site/ergasies/diplwmatikes/Study%20for%20energy%20consumption%20in%20%20LoRa%20networks_VLAMIS.pdf?language=el. [Πρόσβαση 1 2023].
- Τ. Κωνσταντίνος, «Ιδρυματικό Αποθετήριο Πολυτεχνείο Κρήτης,» 7 2018. [Ηλεκτρονικό].
[3] Available: <https://dias.library.tuc.gr/view/78314?show=full&locale=el>. [Πρόσβαση 1 2023].
- Α. Σ. Νίκος Παπαδημητρίου, «Ιδρυματικό Καταθετήριο Επιστημονικών Εργασιών,» 6 2018.
[4] [Ηλεκτρονικό]. Available: <https://ikee.lib.auth.gr/record/298808/files/Thesis%20LoRa.pdf>. [Πρόσβαση 1 2023].
- Μ. Ποταμιάς, «Άρτεμις,» 11 2017. [Ηλεκτρονικό]. Available:
[5] <http://artemis.cslab.ece.ntua.gr:8080/jspui/bitstream/123456789/13640/1/DT2017-0300.pdf>. [Πρόσβαση 1 2023].

ΠΑΡΑΡΤΗΜΑ Α: ΤΕΧΝΙΚΑ ΧΑΡΑΚΤΗΡΙΣΤΗΚΑ ΜΟΝΑΔΩΝ

Οι συσκευές μέτρησης και λήψης δεδομένων περιλαμβάνουν από κοινού από έναν μικροεπεξεργαστή ESP8266 και από έναν πομπό – δέκτη LoRa RA – 02 που διαθέτει το τσιπ της semtech SX 1278. Επίσης η συσκευή μέτρησης περιλαμβάνει μία OLED οθόνη 0.96’’ ιντσών και έναν DHT 11 αισθητήρα. Παρακάτω φαίνονται τα τεχνικά χαρακτηριστικά που ενσωματώνονται στις συσκευές.

ESP 8266

Στην παρακάτω εικόνα φαίνεται ο μικροεπεξεργαστής ESP8266 που χρησιμοποιείται στις δύο συσκευές. Ο ESP8266 είναι ένας ανοιχτού κώδικα μικροεπεξεργαστής που συνήθως συνδυάζεται με αισθητήρες οι οποίοι επιλέγονται ανάλογα με τις απαιτήσεις τις εκάστοτε εφαρμογής για την επίτευξη ενός σκοπού.



Παρακάτω παρουσιάζονται τα βασικά τεχνικά χαρακτηριστικά του ESP8266 τα οποία είναι τα εξής:

- Ενσωματωμένος μικροεπεξεργαστής χαμηλής ισχύος 32 – bit
- Ενσωματωμένος μετατροπέας A/D 10 – bit
- Ενσωματωμένη στοίβα πρωτοκόλλου TCP/IP
- Ενσωματωμένα PLL, ρυθμιστές και διαχειριστές ισχύος
- Wi – Fi 2.4 GHz, υποστήριξη WPA/WPA2
- Υποστήριξη καταστάσεων λειτουργίας STA/AP/STA+AP
- Υποστήριξη smart link για android και iOS συσκευές
- Κατανάλωση deep sleep μικρότερη των 10μΑ
- Αφύπνιση και αποστολή πακέτων μικρότερο από 2ms
- Κατανάλωση ισχύος σε standby mode μικρότερη από 1mW
- Εύρος θερμοκρασίας λειτουργίας από – 40°C έως 125°C
- Ελάχιστη τάση λειτουργίας 3.6V
- Πιστοποιήσεις FCC, CE, TELEC, Wi – Fi Alliance και SRRC
- Ισχύς εξόδου σε κατάσταση 802.11 b/g/n, +20dBm
- SDIO 2.0, SPI, I2C, UART, I2S, IR remote control, PWM, GPIO
- STBC, 1x1 MIMO, 2x1 MIMO

LoRa RA – 02

Στην παρακάτω εικόνα παρουσιάζεται ο πομπό – δέκτης LoRa RA – 02 ο οποίος περιλαμβάνει τσιπ από την semtech. Πιο συγκεκριμένα το τσιπ είναι το SX 1278 και διαθέτει τα εξής τεχνικά χαρακτηριστικά:

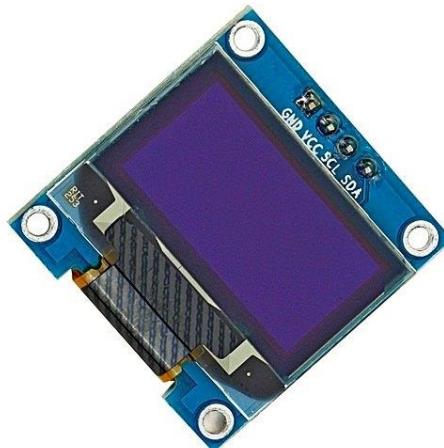
- Απόσταση επικοινωνίας 10km
- Ευαισθησία -148dBm
- Προγραμματιζόμενο bit rate 300kbps
- Δυναμικό εύρος RSSI 127dB
- Συχνότητα συντονισμού 433MHz
- Τάση λειτουργίας 3.3V
- Θερμοκρασία λειτουργίας – 40°C έως 80°C
- Διαστάσεις 28x20.3x5mm



OLED 0.96''

Στην παρακάτω εικόνα απεικονίζεται η οθόνη OLED που χρησιμοποιείται στην συσκευή μέτρησης και αποστολής δεδομένων. Πιο συγκεκριμένα διαθέτει ένα εσωτερικό τσιπ το SSD136 και περιλαμβάνει τα εξής τεχνικά χαρακτηριστικά:

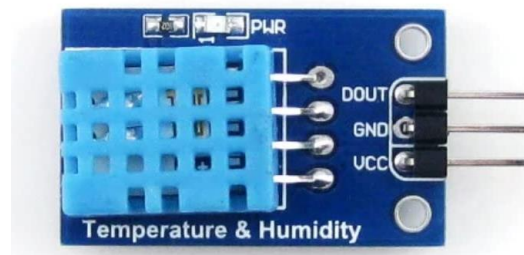
- Διεπαφή IIC/I2C
- Ανάλυση 128x64
- Ισχύς λειτουργίας 60mW
- Τάση λειτουργίας από 3V έως 5V
- Θερμοκρασία λειτουργίας – 30°C έως 70°C
- Διαστάσεις 29.28x27.1mm



DHT 11

Στην παρακάτω εικόνα φαίνεται ο αισθητήρας DHT 11 ο οποίος ενσωματώνεται στην συσκευή μέτρησης και αποστολής δεδομένων. Πρόκειται για έναν σύνθετο αισθητήρα ο οποίος διαθέτει ψηφιακή έξοδο και μετράει θερμοκρασία και υγρασία. Τα τεχνικά χαρακτηριστικά του διατυπώνεται παρακάτω:

- Εύρος μέτρησης θερμοκρασίας 0°C έως 50°C
- Ακρίβεια μέτρησης θερμοκρασίας $\pm 0.5^{\circ}\text{C}$
- Εύρος μέτρησης υγρασίας 20% έως 80%
- Ακρίβεια μέτρησης υγρασίας 2% έως 5%
- Τάση λειτουργίας 3V έως 5V
- Μέγιστο ρεύμα λειτουργίας 2.5mA



ΠΑΡΑΡΤΗΜΑ Β: ΚΩΔΙΚΑΣ ΣΥΣΚΕΥΗΣ ΜΕΤΡΗΣΗΣ ΚΑΙ ΕΚΠΟΜΠΗΣ ΔΕΔΟΜΕΝΩΝ

```
#include <SPI.h> // περίληψη βιβλιοθήκης
#include <LoRa.h> // περίληψη βιβλιοθήκης
#include <DHT.h> // περίληψη βιβλιοθήκης
#include <Adafruit_Sensor.h> // περίληψη βιβλιοθήκης
#include <Adafruit_GFX.h> // περίληψη βιβλιοθήκης
#include <Adafruit_SSD1306.h> // περίληψη βιβλιοθήκης
-----
#define DHTPIN 2 // ονοματοδοσία pin 2 σε DHTPIN
DHT dht(DHTPIN, DHT11); // ορισμός DHTPIN ως pin εισόδου
#define led 0 // ονοματοδοσία pin 0 σε led
-----
const int csPin = 15; // ονοματοδοσία pin 15 σε csPin
const int resetPin = 16; // ονοματοδοσία pin 16 σε resetPin
const int irqPin = 10; // ονοματοδοσία pin 10 σε irqPin
int state = 0; // δημιουργία μεταβλητής state και αρχικοποίηση τιμής
int counter = 0; // δημιουργία μεταβλητής counter και αρχικοποίηση τιμής
int SyncWord = 0x22; // δημιουργία μεταβλητής SyncWord και αρχικοποίηση τιμής
-----
int readingID = 0; // δημιουργία μεταβλητής readingID και αρχικοποίηση τιμής
String LoRaMessage = ""; // δημιουργία μεταβλητής LoRaMessage και αρχικοποίηση τιμής
float temperature = 0; // δημιουργία μεταβλητής temperature και αρχικοποίηση τιμής
float humidity = 0; // δημιουργία μεταβλητής humidity και αρχικοποίηση τιμής
-----
#define SCREEN_WIDTH 128 // αρχικοποίηση τιμής του πλάτους της OLED οθόνης σε pixels
#define SCREEN_HEIGHT 64 // αρχικοποίηση τιμής του ύψους της OLED οθόνης σε pixels
#define OLED_RESET -1 // αρχικοποίηση του reset pin της OLED οθόνης
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET); // ορισμός των pins και του μεγέθους της
OLED οθόνης
-----
void startLoRa() // υπορουτίνα αρχικοποίησης πομπό - δέκτη LoRa
{
  LoRa.setPins(csPin, resetPin, irqPin); // ορισμός pin πομπό - δέκτη LoRa
  while (!LoRa.begin(433E6) && counter < 10) // εκκίνηση του πομπό - δέκτη LoRa στην συχνότητα 433MHz
  {
    Serial.print("."); // εμφάνιση συμβόλου αναμονής
    counter++; // αύξηση της μεταβλητής counter κατά 1
    delay(500); // χρονοκαθυστέρηση 500ms
  }
  if (counter == 10) // αν η μεταβλητή counter ισούται με 10
  {
    readingID++; // αύξηση της μεταβλητής readingID κατά 1
    Serial.println("Starting LoRa failed!"); // εμφάνιση μηνύματος αποτυχίας αρχικοποίησης
  }
  Serial.println("LoRa Initialization OK!"); // εμφάνιση μηνύματος επιτυχίας αρχικοποίησης
  delay(2000); // χρονοκαθυστέρηση 2000ms
}

void startDHT() // υπορουτίνα αρχικοποίησης αισθητήρα DHT11
{
  if (isnan(humidity) || isnan(temperature)) // εκκίνηση αισθητήρα DHT11
  {
    Serial.println("Failed to read from DHT sensor!"); // εμφάνιση μηνύματος αποτυχίας εκκίνησης
    return; // επιστροφή στην void setup
  }
}

void getReadings() // υπορουτίνα ανάγνωσης και εμφάνισης δεδομένων από τον αισθητήρα DHT11
{
  humidity = dht.readHumidity(); // ανάγνωση και καταχώρηση τιμής υγρασίας στην μεταβλητή humidity
  temperature = dht.readTemperature(); // ανάγνωση και καταχώρηση τιμής θερμοκρασίας στην μεταβλητή temperature
  Serial.print(F("Humidity: ")); // εμφάνιση μηνύματος
  Serial.print(humidity); // εμφάνιση τιμής της μεταβλητής humidity
  Serial.print(F("% Temperature: ")); // εμφάνιση μηνύματος
  Serial.print(temperature); // εμφάνιση τιμής της μεταβλητής temperature
  Serial.println(F("°C ")); // εμφάνιση μηνύματος
}
```

```

display.clearDisplay(); // καθαρισμός OLED οθόνης
display.setTextSize(1); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
display.setCursor(0, 0); // επιλογή θέσης του κέρσορα OLED οθόνης
display.print("Temperature: "); // εμφάνιση μηνύματος στην OLED οθόνη
display.setTextSize(2); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
display.setCursor(0, 10); // επιλογή θέσης του κέρσορα OLED οθόνης
display.print(temperature); // εμφάνιση τιμής της μεταβλητής temperature στην OLED οθόνη
display.print(" "); // εμφάνιση μηνύματος στην OLED οθόνη
display.setTextSize(1); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
display.cp437(true); // επιλογή γραμματοσειράς OLED οθόνης
display.write(167); // εμφάνιση συμβόλου στην OLED οθόνη
display.setTextSize(2); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
display.print("C"); // εμφάνιση μηνύματος στην OLED οθόνη
display.setTextSize(1); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
display.setCursor(0, 35); // επιλογή θέσης του κέρσορα OLED οθόνης
display.print("Humidity: "); // εμφάνιση μηνύματος στην OLED οθόνη
display.setTextSize(2); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
display.setCursor(0, 45); // επιλογή θέσης του κέρσορα OLED οθόνης
display.print(humidity); // εμφάνιση τιμής της μεταβλητής humidity στην OLED οθόνη
display.print(" %"); // εμφάνιση συμβόλου στην OLED οθόνη
display.display(); // εφαρμογή των αλλαγών OLED οθόνης
}

-----

void sendReadings() // υπορουτίνα αποστολής δεδομένων
{
  LoRaMessage = String(readingID) + "/" + String(temperature) + "&" + String(humidity); // σύνθεση μηνύματος πακέτου
  αποστολής με τα δεδομένα μέτρησης
  LoRa.beginPacket(); // έναρξη πακέτου
  LoRa.print(LoRaMessage); // μήνυμα
  LoRa.endPacket(); // τέλος πακέτου
  Serial.print("Sending packet: "); // εμφάνιση μηνύματος
  Serial.println(readingID); // εμφάνιση της τιμής μεταβλητής readingID
  readingID++; // αύξηση της μεταβλητής readingID κατά 1
  Serial.println(LoRaMessage); // εμφάνιση μηνύματος
}

-----

void setup() // έναρξη ρουτίνας αρχικοποιήσεων
{
  pinMode(led,OUTPUT); // ορισμός pin led ως έξοδο
  Serial.begin(115200); // ορισμός τιμής baud rate
  dht.begin(); // ενεργοποίηση αισθητήρα DHT11
  startDHT(); // κλήση υπό - ρουτίνας αρχικοποίησης αισθητήρα DHT11
  startLoRa(); // κλήση υπό - ρουτίνας αρχικοποίησης πομπό - δέκτη LoRa
  LoRa.setTxPower(20); // ορισμός μέγιστης ισχύος εκπομπής
  LoRa.setSpreadingFactor(12); // ορισμός μέγιστου παράγοντα διασποράς
  LoRa.setSignalBandwidth(20.8E3); // ορισμός ζώνης BW σε 20.8MHz
  LoRa.setCodingRate4(8); // ορισμός μέγιστου ρυθμού κωδικοποίησης
  LoRa.setSyncWord(SyncWord); // ορισμός byte της λέξης συγχρονισμού
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C); // ενεργοποίηση αισθητήρα DHT11
  display.clearDisplay(); // καθαρισμός OLED οθόνης
  display.setTextColor(WHITE); // ορισμός χρώματος γραμματοσειράς OLED οθόνης
}

-----

void loop() // έναρξη ρουτίνας λειτουργίας
{
  getReadings(); // κλήση υπορουτίνας ανάγνωσης και εμφάνισης δεδομένων από τον αισθητήρα DHT11
  sendReadings(); // κλήση υπορουτίνας αποστολής δεδομένων
  digitalWrite(led, HIGH); // ενεργοποίηση led
  delay(100); // χρονοκαθυστέρηση 100ms
  digitalWrite(led, LOW); // απενεργοποίηση led
  delay(100); // χρονοκαθυστέρηση 100ms
}

```


ΠΑΡΑΡΤΗΜΑ Γ: ΚΩΔΙΚΑΣ ΣΥΣΚΕΥΗΣ ΛΗΨΗΣ ΚΑΙ ΚΑΤΑΧΩΡΗΣΗΣ ΔΕΔΟΜΕΝΩΝ

```
#include <ESP8266WiFi.h> // περίληψη βιβλιοθήκης
#include <WiFiClient.h> // περίληψη βιβλιοθήκης
#include <ESP8266WebServer.h> // περίληψη βιβλιοθήκης
#include <SPI.h> // περίληψη βιβλιοθήκης
#include <LoRa.h> // περίληψη βιβλιοθήκης
#include <Adafruit_Sensor.h> // περίληψη βιβλιοθήκης
#include <Adafruit_GFX.h> // περίληψη βιβλιοθήκης
#include <Adafruit_SSD1306.h> // περίληψη βιβλιοθήκης
#include "index.h" //περίληψη αρχείου σελίδας τοπικού server
-----
#define led 0 // ονοματοδοσία pin 0 σε led
#define SCREEN_WIDTH 128 // αρχικοποίηση τιμής του πλάτους της OLED οθόνης σε pixels
#define SCREEN_HEIGHT 64 // αρχικοποίηση τιμής του ύψους της OLED οθόνης σε pixels
#define OLED_RESET -1 // αρχικοποίηση του RST pin της OLED οθόνης
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET); // ορισμός των pins και του μεγέθους της OLED οθόνης
-----
const int csPin = 15; // ονοματοδοσία pin 15 σε csPin
const int resetPin = 16; // ονοματοδοσία pin 16 σε resetPin
const int irqPin = 10; // ονοματοδοσία pin 10 σε irqPin
int state = 0; // δημιουργία μεταβλητής state και αρχικοποίηση τιμής
int SyncWord = 0x22; // δημιουργία μεταβλητής SyncWord και αρχικοποίηση τιμής
int counter = 0; // δημιουργία μεταβλητής counter και αρχικοποίηση τιμής
String apiKey = "SEYK51W0DHAPLUE7"; // δημιουργία μεταβλητής apiKey και αρχικοποίηση τιμής
const char* ssid = "ACTROS_1840"; // δημιουργία μεταβλητής ssid και αρχικοποίηση τιμής
const char* password = "0481_SORTCA"; // δημιουργία μεταβλητής password και αρχικοποίηση τιμής
const char* server1 = "api.thingspeak.com"; // δημιουργία μεταβλητής server και αρχικοποίηση τιμής
-----
WiFiClient client; // αρχικοποίηση βιβλιοθήκης client
ESP8266WebServer server(80); // άνοιγμα πόρτας 80 στον server
-----
int rssi; // δημιουργία μεταβλητής readingID
String loRaMessage; // δημιουργία μεταβλητής loraMessage
String temperature; // δημιουργία μεταβλητής temperature
String humidity; // δημιουργία μεταβλητής humidity
String readingID; // δημιουργία μεταβλητής readingID
-----
String processor(const String& var) // υπό - ρουτίνα μετατροπής των δεδομένων λήψης
{
  if (var == "TEMPERATURE") // αν var = TEMPERATURE
  {
    return temperature; // επιστροφή temperature
  } else if (var == "HUMIDITY") // αν var = HUMIDITY
  {
    return humidity; // επιστροφή humidity
  } else if (var == "RRSI") // αν var = RRSI
  {
    return String(rssi); // επιστροφή rssi
  }
  return String(); // επιστροφή
}
-----
void handleRoot() // ενεργοποίηση υπό - ρουτίνας όταν ανοίγει η IP του τοπικού server σε browser
{
  String s = MAIN_page; // διαβάζει τον κώδικα απο το html αρχείο
  server.send(200, "text/html", s); // στέλνει τα δεδομένα στην σελίδα
}

void handleADC() // ενεργοποίηση υπό - ρουτίνας για να γίνει η καταχώρηση δεδομένων στον τοπικό server
{
  String data = "{\"Packet\": \""+readingID+"\", \"Temperature\": \""+ temperature +"\", \"Humidity\": \""+ humidity
  +"\", \"RRSI\": \""+String(rssi)+"\"}"; // δημιουργία συμβολοσειράς για καταχώρηση τιμών στον τοπικό server
  server.send(200, "text/plain", data); // αποστολή και καταχώρηση συμβολοσειράς στον τοπικό server με την μέθοδο
  get/post
}
```

```

void startLoRa() // υπορουτίνα αρχικοποίησης πομπό - δέκτη LoRa
{
  LoRa.setPins(csPin, resetPin, irqPin); // ορισμός pin πομπό - δέκτη LoRa
  while (!LoRa.begin(433E6) && counter < 10) // εκκίνηση του πομπό - δέκτη LoRa στην συχνότητα 433MHz
  {
    Serial.print("."); // εμφάνιση συμβόλου αναμονής
    counter++; // αύξηση της μεταβλητής counter κατά 1
    delay(500); // χρονοκαθυστέρηση 500ms
  }
  if (counter == 10) // αν η μεταβλητή counter ισούται με 10
  {
    Serial.println("Starting LoRa failed!"); // εμφάνιση μηνύματος αποτυχίας αρχικοποίησης
  }
  Serial.println("LoRa Initialization OK!"); // εμφάνιση μηνύματος επιτυχίας αρχικοποίησης
  display.setCursor(0,10); // επιλογή θέσης του κέρσορα OLED οθόνης
  display.print("LoRa OK!"); // εμφάνιση μηνύματος στην OLED οθόνη
  display.display(); // εφαρμογή των αλλαγών OLED οθόνης
  delay(2000); // χρονοκαθυστέρηση 2000ms
}

-----

void startWiFi() // υπορουτίνα αρχικοποίησης και σύνδεσης Wi - Fi
{
  Serial.print("Connecting to "); // εμφάνιση μηνύματος σύνδεσης Wi - Fi
  Serial.println(ssid); // εμφάνιση ονόματος Wi - Fi
  WiFi.begin(ssid, password); // χρησιμοποίηση ονόματος και κωδικού Wi - Fi για σύνδεση
  Serial.println(""); // εμφάνιση μηνύματος
  while (WiFi.status() != WL_CONNECTED) // αναμονή σύνδεσης στο Wi - Fi
  {
    delay(500); // χρονοκαθυστέρηση 500ms
    Serial.print("."); // εμφάνιση συμβόλου αναμονής
  }
  Serial.println(""); // εμφάνιση μηνύματος
  Serial.print("Connected to "); // εμφάνιση μηνύματος ότι συνδέθηκε στο Wi - Fi
  Serial.println(ssid); // εμφάνιση ονόματος Wi - Fi
  Serial.print("IP address: "); // εμφάνιση μηνύματος για την IP
  Serial.println(WiFi.localIP()); // εμφάνιση IP που έχει δωθεί στον ESP
  display.clearDisplay(); // καθαρισμός OLED οθόνης
  display.print("Wi - Fi connect"); // εμφάνιση μηνύματος στην OLED οθόνη
  display.clearDisplay(); // καθαρισμός OLED οθόνης
  display.setCursor(0, 20); // επιλογή θέσης του κέρσορα OLED οθόνης
  display.print("Access web server at: "); // εμφάνιση μηνύματος στην OLED οθόνη
  display.setCursor(0, 30); // επιλογή θέσης του κέρσορα OLED οθόνης
  display.print(WiFi.localIP()); // εμφάνιση της IP στην OLED οθόνη
  display.display(); // εφαρμογή των αλλαγών OLED οθόνης
  delay(2000); // χρονοκαθυστέρηση 2000ms
}

-----

void startServer() // υπορουτίνα αρχικοποίησης και εκκίνησης του τοπικού server
{
  server.on("/", handleRoot); // ενεργοποίηση σελίδας τοπικού server
  server.on("/readADC", handleADC); // κλήση σελίδας
  server.begin(); // εκκίνηση τοπικού server
  Serial.println("HTTP server started"); // εμφάνιση μηνύματος για την εκκίνηση του τοπικού server
  display.setCursor(0, 40); // επιλογή θέσης του κέρσορα OLED οθόνης
  display.print("Server OK"); // εμφάνιση μηνύματος στην OLED οθόνη
  display.display(); // εφαρμογή των αλλαγών OLED οθόνης
  delay(2000); // χρονοκαθυστέρηση 2000ms
}

-----

void receiveData() // υπορουτίνα ανάλυσης και καταχώρησης δεδομένων
{
  Serial.print("Lora packet received: "); // εμφάνιση μηνύματος
  while (LoRa.available()) // ενεργοποίηση πομπό - δέκτη LoRa
  {
    String LoRaData = LoRa.readString(); // ανάγνωση και καταχώρηση τιμής LoRa πακέτου λήψης
    Serial.print(LoRaData); // εμφάνιση τιμής της μεταβλητής LoRaData
    int pos1 = LoRaData.indexOf('/'); // καταχώρηση τιμής του αριθμού των χαρακτήρων ως το σύμβολο /
    int pos2 = LoRaData.indexOf('&'); // καταχώρηση τιμής του αριθμού των χαρακτήρων ως το σύμβολο &
    readingID = LoRaData.substring(0, pos1); // ανάγνωση και καταχώρηση τιμής αριθμού πακέτου αποστολής
    temperature = LoRaData.substring(pos1 + 1, pos2); // ανάγνωση και καταχώρηση τιμής αριθμού θερμοκρασίας
    humidity = LoRaData.substring(pos2 + 1, LoRaData.length()); // ανάγνωση και καταχώρηση τιμής αριθμού υγρασίας
  }
}

```

```

}
rssi = LoRa.packetRssi(); // ανάγνωση και καταχώρηση τιμής ένδειξης της ισχύος του σήματος λήψης
Serial.print(" with RSSI "); // εμφάνιση μηνύματος
Serial.println(rssi); // εμφάνιση τιμής της μεταβλητής rssi
}

-----

void sendThingSpeak() // υπορουτίνα σύνθεσης και καταχώρησης δεδομένων στο ThingSpeak
{
if (client.connect(server1, 80)) // εκκίνηση καταχώρησης δεδομένων στο ThingSpeak
{
String postStr = apiKey; // καταχώρηση στην μεταβλητή postStr την τιμή της μεταβλητής apiKey
postStr += "&field1="; // καταχώρηση στην μεταβλητή postStr για την επιλογή του field 1
postStr += String(readingID); // καταχώρηση στο field 1 την τιμή της μεταβλητής readingID
postStr += "&field2="; // καταχώρηση στην μεταβλητή postStr για την επιλογή του field 2
postStr += String(temperature); // καταχώρηση στο field 2 την τιμή της μεταβλητής temperature
postStr += "&field3="; // καταχώρηση στην μεταβλητή postStr για την επιλογή του field 3
postStr += String(humidity); // καταχώρηση στο field 2 την τιμή της μεταβλητής humidity
postStr += "&field4="; // καταχώρηση στην μεταβλητή postStr για την επιλογή του field 4
postStr += String(rssi); // καταχώρηση στο field 2 την τιμή της μεταβλητής rssi
postStr += "\r\n\r\n\r\n\r\n"; // καταχώρηση στην μεταβλητή postStr το μήνυμα
client.print("POST /update HTTP/1.1\n"); // δημιουργία συνδέσμου για ThingSpeak εμφάνιση μηνύματος
client.print("Host: api.thingspeak.com\n"); // δημιουργία συνδέσμου για ThingSpeak εμφάνιση μηνύματος
client.print("Connection: close\n"); // δημιουργία συνδέσμου για ThingSpeak εμφάνιση μηνύματος
client.print("X-THINGSPEAKAPIKEY: " + apiKey + "\n"); // καταχώρηση τιμής της μεταβλητής apiKey για καταχώρηση
δεδομένων μέτρησης
client.print("Content-Type: application/x-www-form-urlencoded\n"); // ορισμός τύπου περιεχομένου των δεδομένων
μέτρησης
client.print("Content-Length: "); // ορισμός μεγέθους του περιεχομένου των δεδομένων μέτρησης
client.print(postStr.length()); // ανάγνωση του μεγέθους του περιεχομένου των δεδομένων μέτρησης
client.print("\n\n"); // καταχώρηση δύο enter
client.print(postStr); // καταχώρηση της τιμής της μεταβλητής postStr
}
}

-----

void startOLED() // υπορουτίνα αρχικοποίησης και εκκίνησης της OLED οθόνης
{
display.begin(SSD1306_SWITCHCAPVCC, 0x3C); // ενεργοποίηση OLED οθόνης
display.clearDisplay(); // καθαρισμός OLED οθόνης
display.setTextColor(WHITE); // ορισμός χρώματος γραμματοσειράς OLED οθόνης
display.setTextSize(1); // επιλογή μεγέθους γραμματοσειράς OLED οθόνης
display.setCursor(0,0); // επιλογή θέσης του κέρσορα OLED οθόνης
display.print("LORA RECEIVER"); // εμφάνιση μηνύματος στο OLED οθόνη
display.display(); // εφαρμογή των αλλαγών OLED οθόνης
delay(2000); // χρονοκαθυστέρηση 2000ms
}

-----

void setup()
{
pinMode(led,OUTPUT); // ορισμός pin led ως έξοδο
Serial.begin(115200); // ορισμός τιμής baud rate
Serial.println(); // αλλαγή γραμμής
delay(10); // χρονοκαθυστέρηση 10ms
startOLED(); // κλήση υπό - ρουτίνας αρχικοποίησης και εκκίνησης OLED οθόνης
startLoRa(); // κλήση υπό - ρουτίνας αρχικοποίησης και ενεργοποίησης πομπό - δέκτη LoRa
startWiFi(); // κλήση υπό - ρουτίνας αρχικοποίησης και ενεργοποίησης Wi - Fi
startServer(); // κλήση υπό - ρουτίνας αρχικοποίησης και εκκίνησης τοπικού server
LoRa.setSpreadingFactor(12); // ορισμός μέγιστου παράγοντα διασποράς
LoRa.setSignalBandwidth(20.8E3); // ορισμός ζώνης BW σε 20.8MHz
LoRa.setCodingRate4(8); // ορισμός μέγιστου ρυθμού κωδικοποίησης
LoRa.setSyncWord(SyncWord); // ορισμός byte της λέξης συγχρονισμού
Serial.println("End Setup"); // εμφάνιση μηνύματος
}

-----

void loop()
{
int packetSize = LoRa.parsePacket(); // καταχώρηση τιμής μεγέθους του πακέτου στην μεταβλητή packetSize
if (packetSize) // έλεγχος καταχώρησης τιμής στην μεταβλητή packetSize
{
receiveData(); // κλήση υπορουτίνας ανάλυσης και καταχώρησης δεδομένων
sendThingSpeak(); // κλήση υπορουτίνας σύνθεσης και καταχώρησης δεδομένων στο ThingSpeak
digitalWrite(led, HIGH); // ενεργοποίηση led
}
}

```

```

delay(100);           // χρονοκαθυστέρηση 100ms
digitalWrite(led, LOW); // απενεργοποίηση led
delay(100);           // χρονοκαθυστέρηση 100ms
display.clearDisplay(); // καθαρισμός OLED οθόνης
display.setCursor(0, 0); // επιλογή θέσης του κέρσορα OLED οθόνης
display.print(WiFi.localIP()); // εμφάνιση της IP στην OLED οθόνη
display.setCursor(0, 10); // επιλογή θέσης του κέρσορα OLED οθόνης
display.print("Packet No: "); // εμφάνιση μηνύματος στην OLED οθόνη
display.setCursor(0, 20); // επιλογή θέσης του κέρσορα OLED οθόνης
display.print(readingID); // εμφάνιση του αριθμού πακέτου στην OLED οθόνη
display.display(); // εφαρμογή των αλλαγών OLED οθόνης
}
server.handleClient(); // κλήση υπορουτίνας καταχώρησης δεδομένων στον τοπικό server
}

```

ΠΑΡΑΡΤΗΜΑ Δ: ΚΩΔΙΚΑΣ WEB ΣΕΛΙΔΑΣ ΤΟΠΙΚΟΥ SERVER

```
const char MAIN_page[] PROGMEM = R"====(
<!doctype html>
<html>
/* Διαμόρφωση κεφαλίδας */
<head>
  <title>Line Chart | Kioumourtzis Konstantinos</title> // εμφάνιση μηνύματος κεφαλίδας
  <script src = "https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.3/Chart.min.js"></script> // κλήση κώδικα για την
  εμφάνιση του γραφικού περιβάλλοντος
  <style>
  canvas
  {
    -moz-user-select: none;
    -webkit-user-select: none;
    -ms-user-select: none;
  }
  /* Διαμόρφωση πίνακα δεδομένων */
  #dataTable
  {
    font-family: "Trebuchet MS", Arial, Helvetica, sans-serif; // επιλογή γραμματοσειράς
    border-collapse: collapse; // σύμπτυξη περιγραμμάτων κελιών
    width: 100%; // καθορισμός πλάτους κελιών
  }
  #dataTable td, #dataTable th
  {
    border: 1px solid #ddd; // διαμόρφωση περιγραμμάτων κελιών
    padding: 8px; // διαμόρφωση συμπλήρωσης κελιών
  }

  #dataTable tr:nth-child(even){background-color: #f2f2f2;} // διαμόρφωση χρώματος φόντου κελιών

  #dataTable tr:hover {background-color: #ddd;} // διαμόρφωση χρώματος φόντου κελιών

  #dataTable th
  {
    padding-top: 12px; // διαμόρφωση συμπλήρωσης κελιών κορυφής από πάνω
    padding-bottom: 12px; // διαμόρφωση συμπλήρωσης κελιών κορυφής από κάτω
    text-align: left; // αριστερή στοίχιση κειμένου κελιών κορυφής
    background-color: #4CAF50; // διαμόρφωση χρώματος φόντου κελιών κορυφής
    color: white; // διαμόρφωση χρώματος γραμματοσειράς κελιών κορυφής
  }
  </style>
</head>
/* Διαμόρφωση σώματος σελίδας */
<body>
  <div style="text-align:center;"><b>Kioumourtzis Konstantinos</b><br>Real Time Data Humidity, Temperature Logging
  with Graphs on ESP8266</div> // διαμόρφωση και εμφάνιση μηνύματος με την λειτουργία που πραγματοποιεί η σελίδα
  <div class="chart-container" position: relative; height:350px; width:100%">
    <canvas id="Chart" width="400" height="400"></canvas>
  </div>
  <div>
    <table id="dataTable">
      <tr><th>Time</th><th>Packet No</th><th>Temperaure (&deg;C)</th><th>Humidity (%)</th><th>RSSI</th></tr> // εμφάνιση
      μηνύματος στα κελιά κορυφής σε κάθε στήλη με την πληροφορία που θα περιλαμβάνουν
    </table>
  </div>
  <br>
  <br>
  <script>
  /* Ορισμός μεταβλητών */
  var RSSI = [];
  var PacketNo = [];
  var Tvalues = [];
  var Hvalues = [];
  var timeStamp = [];
```

```

/* Διαμόρφωση γραφήματος δεδομένων */
function showGraph()
{
    var ctx = document.getElementById("Chart").getContext('2d'); // επιλογή διαστάσεων γραφήματος
    var Chart2 = new Chart(ctx,
    {
        type: 'line', // επιλογή εμφάνισης γραμμής μεταξύ των σημείων
        data:
        {
            labels: timeStamp, // εμφάνιση ετικέτας με την ώρα καταχώρησης
            datasets:
            [
                {
                    label: "Temperature", // εμφάνιση ετικέτας θερμοκρασίας
                    fill: false,
                    backgroundColor: 'rgba( 243, 156, 18 , 1)', // επιλογή χρώματος καταχώρησης σημείου θερμοκρασίας
                    borderColor: 'rgba( 243, 156, 18 , 1)', // επιλογή χρώματος γραμμής θερμοκρασίας
                    data: Tvalues, // καταχώρηση τιμής θερμοκρασίας
                },
                {
                    label: "Humidity", // εμφάνιση ετικέτας υγρασίας
                    fill: false,
                    backgroundColor: 'rgba(156, 18, 243 , 1)', // επιλογή χρώματος καταχώρησης σημείου υγρασίας
                    borderColor: 'rgba(156, 18, 243 , 1)', // επιλογή χρώματος γραμμής υγρασίας
                    data: Hvalues, // καταχώρηση τιμής υγρασίας
                }
            ]
        },
        /* εμφάνιση τίτλου, προσαρμογή, καμπύλη μεταξύ σημείων και έναρξη άξονα y από το μηδέν*/
        options:
        {
            title:
            {
                display: true,
                text: "Web Server on ESP"
            },
            maintainAspectRatio: false,
            elements:
            {
                line:
                {
                    tension: 0.5 //Smoothing (Curved) of data lines
                }
            },
            scales:
            {
                yAxes:
                [
                    [
                        {
                            ticks:
                            {
                                beginAtZero:true
                            }
                        }
                    ]
                ]
            }
        }
    });
}

/* Εμφάνιση περιοχής γραφήματος στην σελίδα */
window.onload = function()
{
    console.log(new Date().toLocaleTimeString());
};

/* Ανανέωση σελίδας και καταχώρησης δεδομένων κάθε 15 sec */
setInterval(function(){getData();}, 15000);
/* Καταχώρηση δεδομένων σε μορφή γραφήματος και στο πίνακα */
function getData()
{
    var xhttp = new XMLHttpRequest(); // ορισμός μεταβλητής στην οποία καταχωρούνται τα νέα αιτήματα
    xhttp.onreadystatechange = function() // ενεργοποίηση συνάρτησης κάθε φορά υπάρχει νέο αίτημα
    {
        if (this.readyState == 4 && this.status == 200) // έλεγχος νέου αιτήματος
        {

```

```

/* διαχωρισμός δεδομένων */
var time = new Date().toLocaleTimeString();
var txt = this.responseText;
var obj = JSON.parse(txt);
/* Καταχώρηση δεδομένων σε μορφή γραφήματος και στον πίνακα */
PacketNo.push(obj.Packet);
Tvalues.push(obj.Temperature);
Hvalues.push(obj.Humidity);
timeStamp.push(time);
showGraph();
var table = document.getElementById("dataTable");
var row = table.insertRow(1);
var cell1 = row.insertCell(0);
var cell2 = row.insertCell(1);
var cell3 = row.insertCell(2);
var cell4 = row.insertCell(3);
var cell5 = row.insertCell(4);
cell1.innerHTML = time;
cell2.innerHTML = obj.Packet;
cell3.innerHTML = obj.Temperature;
cell4.innerHTML = obj.Humidity;
cell5.innerHTML = obj.RSSI;
}
};
xhttp.open("GET", "readADC", true);
xhttp.send();
}

</script>
</body>

</html>

)=====";

```