

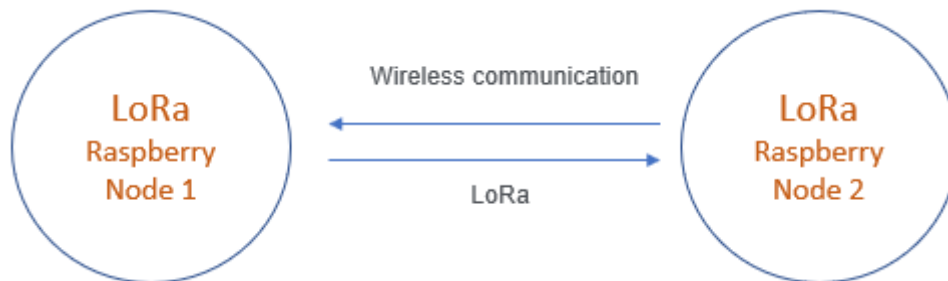
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ
ΣΥΣΤΗΜΑΤΩΝ

ΠΜΣ «Εφαρμοσμένα Ηλεκτρονικά Συστήματα»

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Μελέτη και υλοποίηση συστήματος επικοινωνίας μεταξύ δύο συσκευών με LoRa



Φοιτητής

Λιάπης Κωνσταντίνος 51810m

Επιβλέπων

Δρ. Κυριάκος Τσιακμάκης

Σεπτέμβριος 2021

Μελέτη και υλοποίηση συστήματος επικοινωνίας μεταξύ δύο συσκευών με LoRa

Κωδικός: 21127

Φοιτητής: Λιάπης Κωνσταντίνος

Εισηγητής: Δρ Κυριάκος Τσιακμάκης

Ημερομηνία ανάληψης Π.Ε. 24-02-2021

Ημερομηνία περάτωσης Π.Ε. 01-09-2021

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

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

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

Περίληψη

Η εργασία αυτή αφορά τη μελέτη και υλοποίηση συστήματος επικοινωνίας μεταξύ δύο συσκευών με LoRa. Χρησιμοποιήθηκε ειδικό LoRa module το SX1278 με συγκεκριμένη κεραία. Στους κόμβους χρησιμοποιήθηκε το Raspberry Pi για την λήψη και μετάδοση μέσω του SX1278. Αρχικά μελετήθηκαν τα χαρακτηριστικά αυτής της επικοινωνίας για να εκτιμηθούν η μέγιστη απόσταση επικοινωνίας σε ανοιχτό χώρο αλλά και σε χώρο με συγκεκριμένα εμπόδια. Επιπλέον έγιναν μετρήσεις που αφορούν το μέγεθος RSSI, SNR και το Loss Packet Ration.

Στη συνέχεια υλοποιήθηκε ένα είδος απλού πρωτοκόλλου επικοινωνίας για μια απλή τοπολογία αστέρα και επικοινωνία κόμβων μέσω ενός κεντρικού με το διαδίκτυο. Σχεδιάστηκε τοπολογία που αφορά ένα θερμοκήπιο αλλά και για πολλά θερμοκήπια και πως μπορούν να επικοινωνούν οι κόμβοι μεταξύ τους. Αναλύθηκε η προσέγγιση που πραγματοποιήθηκε για την ασφάλεια της μετάδοσης. Από την πλευρά της ασφάλειας η εργασία επικεντρώθηκε στην ταυτοποίηση του κόμβου, στην κρυπτογράφηση του πακέτου μετάδοσης από κόμβο σε κόμβο και στην ασφαλή μετάδοση στους servers. Υλοποιήθηκε server σε python και χρήση του cloud server ThingSpeak. Στη συνέχεια γίνεται προβολή των δεδομένων σε ιστοσελίδα.

« Design and implementation of a communication system between two devices with LoRa»

Abstract

This work concerns the study and implementation of a communication system between two devices with LoRa. A special LoRa module SX1278 with a specific antenna was used. The Raspberry Pi was used at the nodes to receive and transmit via the SX1278. Initially, the characteristics of this communication were studied in order to estimate the maximum communication distance in open space but also in space with specific obstacles. In addition, RSSI size, SNR and Loss Packet Ration measurements have been performed.

A kind of simple communication protocol was implemented using a simple star topology and a communication of nodes through a central node via the internet has been implemented. A topology was designed for greenhouses. The approach taken for transmission safety was analyzed. On the security side, the work focused on node authentication, node-to-node transmission packet encryption, and secure transmission to servers. A python server has been measurements have been implemented. Additionally, we used the cloud server ThingSpeak. All data displayed on a web page.

Ευχαριστίες

Θέλω να ευχαριστήσω του γονείς μου και τον κ. Τσιακμάκη Κ. για την πολύτιμη συμβολή του σε κάθε μου βήμα.

Περιεχόμενα

Περίληψη.....	iv
Abstract	v
Ευχαριστίες.....	vi
Περιεχόμενα	vii
Κατάλογος Σχημάτων	viii
Κεφάλαιο 1ο: Εισαγωγή.....	10
1.1 Εισαγωγή.....	10
1.2 Δομή της εργασίας	11
Κεφάλαιο 2ο: Τεχνολογίες.....	12
2.1 Εισαγωγή στο LoRa	12
2.1.1 LPWAN.....	13
2.1.2 LoRaWAN	15
2.1.3 Μετρούμενα μεγέθη	17
2.2 Παρόμοια συστήματα.....	19
2.3 LoRa modules	21
2.4 Raspberry	24
2.5 ThingSpeak	25
2.6 Python.....	28
2.7 Flask.....	29
Κεφάλαιο 3ο: Περιγραφή Συστήματος και Λειτουργιών	33
3.1 Το σύστημα.....	33
Κεφάλαιο 4ο: Πειραματικό Μέρος.....	54
4.1 Μετρήσεις σε ανοιχτό πεδίο χωρίς εμπόδια.....	54
4.2 Μετρήσεις σε ανοιχτό πεδίο με εμπόδια	58
Κεφάλαιο 5ο: Συμπεράσματα και προτάσεις βελτίωσης.....	66
ΒΙΒΛΙΟΓΡΑΦΙΑ	67

Κατάλογος Σχημάτων

Εικόνα 2.1: Σύγκριση εύρους και απόστασης	14
Εικόνα 2.2: SX1278 LoRa module	22
Εικόνα 2.3: Antenna RF 434MHz 2dBi 63.6mm u.FL για το SX1278 LoRa module.....	23
Εικόνα 2.4: ThingSpeak – Κανάλια – Νέο Κανάλι	26
Εικόνα 2.5: Ρυθμίσεις καναλιού.....	27
Εικόνα 2.6: Τα κλειδιά API	27
Εικόνα 3.1: Το σύστημα με τους δύο κόμβους.....	33
Εικόνα 3.2: Η διάταξη-κόμβος που χρησιμοποιήθηκε	33
Εικόνα 3.3: Φωτογραφία διάταξης – κόμβος server	34
Εικόνα 3.4: Φωτογραφία διάταξης – κόμβος client	34
Εικόνα 3.5: Το πρώτο βασικό πείραμα με τους δυο κόμβους με τον client να απομακρύνεται....	35
Εικόνα 3.6: Απλή τοπολογία που δείχνει την αποστολή των δεδομένων σε server	35
Εικόνα 3.7: Επιπρόσθετοι κόμβοι-client που αποστέλλουν δεδομένα στον κεντρικό κόμβο-server	36
Εικόνα 3.8: Τοπολογία με πολλούς clients που μπορούν να τοποθετηθούν σε έναν χώρο, όπως ένα Θερμοκήπιο	37
Εικόνα 3.9: Τοπολογία με τέσσερα Θερμοκήπια που έχουν πολλούς clients	38
Εικόνα 3.10: Τοπολογία με πολλά θερμοκήπια	39
Εικόνα 3.11: Κεντρικό διάγραμμα λειτουργίας στον Client.....	40
Εικόνα 3.12: Μορφοποίηση του βασικού πακέτου μετάδοσης	43
Εικόνα 3.13: Μορφοποίηση του πακέτου επιβεβαίωσης	45
Εικόνα 3.14: Βασικές λειτουργίες που εκτελούνται στον Server-κόμβο-Raspberry.....	46
Εικόνα 3.15: Περιγραφή επικοινωνίας κόμβου με τους servers ThingSpeak.....	49
Εικόνα 3.16: Απευθείας αποστολή των δεδομένων-πακέτου από το κόμβο-server στον υπολογιστή server χωρίς την χρήση ThingSpeak	51
Εικόνα 3.17: Αναπαράσταση δεδομένων σε ιστοσελίδα με framework Flask.....	51
Εικόνα 3.18: Αναπαράσταση δεδομένων σε ThingSpeak	52
Εικόνα 4.1: Μετρήσεις σε ανοιχτό πεδίο στη νέα παραλία Θεσσαλονίκης	54
Εικόνα 4.2: Σχέση ανάμεσα στην απόσταση των δύο συσκευών και το RSSI σε ανοιχτό πεδίο.....	55
Εικόνα 4.3: Σχέση ανάμεσα στην απόσταση των δύο συσκευών και το SNR σε ανοιχτό πεδίο.....	56
Εικόνα 4.4: Η σχέση ανάμεσα στον κανονικοποιημένο λόγω απώλειας πακέτων (Packet Loss Ratio) και στην απόσταση μεταξύ των δύο κόμβων σε ανοιχτό πεδίο	57
Εικόνα 4.5: Μετρήσεις σε ανοιχτό πεδίο με εμπόδια σε χώρο με πολυκατοικίες.....	58
Εικόνα 4.6: Σχέση ανάμεσα στην απόσταση των δύο συσκευών και το RSSI σε ανοιχτό πεδίο με συγκεκριμένα εμπόδια	59
Εικόνα 4.7: Σχέση ανάμεσα στην απόσταση των δύο συσκευών και το SNR σε ανοιχτό πεδίο με συγκεκριμένα εμπόδια	60
Εικόνα 4.8: Η σχέση ανάμεσα στον κανονικοποιημένο λόγω απώλειας πακέτων (Packet Loss Ratio) και στην απόσταση μεταξύ των δύο κόμβων σε ανοιχτό πεδίο με συγκεκριμένα εμπόδια	61
Εικόνα 4.9: Μετρήσεις σε δεύτερο ανοιχτό χώρο με εμπόδια	62
Εικόνα 4.10: Μετρήσεις σε τρίτο ανοιχτό χώρο με εμπόδια	63

Εικόνα 4.11: Μετρήσεις σε ανοιχτό χώρο με τη μια συσκευή να βρίσκεται σε μεγαλύτερο ύψος 64
Εικόνα 4.12: Η σχέση ανάμεσα στον κανονικοποιημένο λόγω απώλειας πακέτων (Packet Loss Ratio) και στον χρόνο αποστολής μεταξύ των δύο κόμβων σε ανοιχτό πεδίο 65

Κεφάλαιο 1ο: Εισαγωγή

1.1 Εισαγωγή

Το Διαδίκτυο των Πραγμάτων (IoT) είναι μια αναδύομενη τεχνολογία που επιτρέπει την επικοινωνία μεταξύ ηλεκτρονικών συσκευών και αισθητήρων μέσω του Διαδικτύου για να διευκολύνει τη ζωή μας. Το IoT χρησιμοποιεί έξυπνες συσκευές και διαδίκτυο για να παρέχει καινοτόμες λύσεις σε διάφορες προκλήσεις και ζητήματα που σχετίζονται με διάφορες επιχειρηματικές, κυβερνητικές και δημόσιες/ιδιωτικές βιομηχανίες σε όλο τον κόσμο. Το IoT γίνεται σταδιακά μια σημαντική πτυχή της ζωής μας που μπορεί να γίνει αντιληπτή παντού γύρω μας. Συνολικά, το IoT είναι μια καινοτομία που συνδυάζει μεγάλη ποικιλία έξυπνων συστημάτων, πλαισίων και ευφών συσκευών και αισθητήρων.

Σήμερα, όλο και περισσότερες βιομηχανικές, εμπορικές εφαρμογές και εφαρμογές πελατών βασίζονται για δεδομένα που συλλέγονται από πληθώρα ετερογενών συσκευών που βρίσκονται στα όρια του δικτύου. Δεδομένης της υψηλής πυκνότητας τους, της περιορισμένης διαθεσιμότητας ενέργειας-μπαταρίας και της δύσκολης πρόσβασης τους στις περισσότερες περιπτώσεις, η ερευνητική κοινότητα επικεντρώνεται στη σχεδίαση λύσεων για την εξασφάλιση ασύρματων μεταδόσεων μεγάλης εμβέλειας, χαμηλής ισχύος και χαμηλού ρυθμού bit από και προς αυτές τις συσκευές δηλαδή στην ανάπτυξη των λεγόμενων LPWAN (χαμηλής ισχύος δίκτυα ευρείας περιοχής).[1]

Μια γνωστή ασύρματη τεχνολογία για τα LPWAN είναι το LoRa (μεγάλης εμβέλειας). Το LoRa υποστηρίζει τη δυνατότητα ισχυρών επικοινωνιών μεγάλης εμβέλειας σε μη αδειοδοτημένες βιομηχανικές, επιστημονικές και ιατρικές ζώνες συχνοτήτων. Δεδομένου της αξιοπιστίας και της ευελιξίας, το LoRa έγινε γρήγορα η πιο ευρέως υιοθετημένη λύση για το φυσικό επίπεδο για το LPWAN. Όσον αφορά τα ανώτερα στρώματα, η LoRa Alliance προτείνει το LoRaWAN [9], ένα ειδικό πρωτόκολλο δικτύου που επιτρέπει την επικοινωνία συσκευών που βασίζονται σε LoRa σε ένα καλά καθορισμένο δίκτυο.

Η εργασία αυτή αφορά τη μελέτη και υλοποίηση συστήματος επικοινωνίας μεταξύ δύο συσκευών με LoRa. Χρησιμοποιήθηκε ειδικό LoRa module το SX1278 με συγκεκριμένη κεραία. Στους κόμβους χρησιμοποιήθηκε το Raspberry Pi για την λήψη και μετάδοση μέσω του SX1278. Αρχικά μελετήθηκαν τα χαρακτηριστικά αυτής της επικοινωνίας για να εκτιμηθούν η μέγιστη απόσταση επικοινωνίας σε ανοιχτό χώρο αλλά και σε χώρο με συγκεκριμένα εμπόδια. Επιπλέον έγιναν μετρήσεις που αφορούν το μέγεθος RSSI, SNR και το Loss Packet Ration.

Στη συνέχεια υλοποιήθηκε ένα είδος απλού πρωτοκόλλου επικοινωνίας για μια απλή τοπολογία αστέρα και επικοινωνία κόμβων μέσω ενός κεντρικού με το διαδίκτυο. Σχεδιάστηκε τοπολογία που αφορά ένα θερμοκήπιο αλλά και για πολλά θερμοκήπια και πως μπορούν να επικοινωνούν οι κόμβοι μεταξύ τους.

Αναλύθηκε η προσέγγιση που πραγματοποιήθηκε για την ασφάλεια της μετάδοσης. Από την πλευρά της ασφάλειας η εργασία επικεντρώθηκε στην ταυτοποίηση του κόμβου, στην κρυπτογράφηση του πακέτου μετάδοσης από κόμβο σε κόμβο και στην ασφαλή μετάδοση στους servers. Υλοποιήθηκε server σε python και χρήση του cloud server ThingSpeak. Στη συνέχεια γίνεται προβολή των δεδομένων σε ιστοσελίδα.

1.2 Δομή της εργασίας

Στο πρώτο κεφάλαιο παρουσιάζεται μια μικρή εισαγωγή της εργασίας και οι στόχοι της.

Στο δεύτερο κεφάλαιο παρουσιάζονται οι τεχνολογίες που χρησιμοποιήθηκαν στην εργασία. Επίσης, παρουσιάζεται μια μικρή εισαγωγή σε παρόμοια συστήματα.

Στο τρίτο κεφάλαιο περιγράφεται το σύστημα και οι λειτουργίες του μέσω διαγραμμάτων και επεξήγηση κώδικα.

Στο τέταρτο κεφάλαιο παρουσιάζονται τα πειράματα που πραγματοποιήθηκαν και τα αποτελέσματα αυτών.

Στο πέμπτο κεφάλαιο παρουσιάζονται τα συμπεράσματα της εργασίας και σκέψεις για μελλοντική έρευνα.

Στο τέλος της εργασίας υπάρχει το παράρτημα με ορισμένους από τους κώδικες που χρησιμοποιήθηκαν στην εργασία.

Κεφάλαιο 2ο: Τεχνολογίες

2.1 Εισαγωγή στο LoRa

Το LoRa (Long Range) είναι μια τεχνική διαμόρφωσης δικτύου ευρείας περιοχής χαμηλής ισχύος. Βασίζεται σε τεχνικές διαμόρφωσης φάσματος εξάπλωσης που προέρχονται από την τεχνολογία chirp spread spectrum (CSS).[1][2][3]

Η τεχνολογία LoRa δημιουργήθηκε από μια γαλλική εταιρεία που ονομάζεται Cycleo, η οποία αποκτήθηκε στη συνέχεια από την Semtech το 2012. Η Semtech ήταν το ιδρυτικό μέλος της LoRa Alliance που είναι τώρα το διοικητικό όργανο της LoRa Technology.

Η LoRa Alliance είναι μία από τις ταχύτερα αναπτυσσόμενες τεχνολογικές συμμαχίες. Αυτή η μη κερδοσκοπική ένωση αποτελείται από περισσότερες από 500 εταιρείες μέλη, δεσμευμένες να επιτρέψουν την ανάπτυξη μεγάλης κλίμακας IoT Low Power Wide Area Networks (LPWAN) μέσω της ανάπτυξης και προώθησης του ανοιχτού προτύπου LoRaWAN.

Το LoRa (συντομογραφία: μεγάλη εμβέλεια) της Semtech είναι μια ασύρματη πλατφόρμα μεγάλης εμβέλειας, χαμηλής ισχύος που έχει γίνει η de facto ασύρματη πλατφόρμα του Internet of Things (IoT). Οι συσκευές και τα δίκτυα LoRa όπως το LoRaWAN επιτρέπουν έξυπνες εφαρμογές IoT που λύνουν μερικές από τις μεγαλύτερες προκλήσεις που αντιμετωπίζει ο πλανήτης όπως διαχείριση ενέργειας και φυσικών πόρων, έλεγχος και πρόληψη καταστροφών. Οι συσκευές LoRa της Semtech έχουν εγκατασταθεί σε αρκετές περιπτώσεις χρήσης για έξυπνες πόλεις, σπίτια και κτίρια, κοινότητες, μετρήσεις, αλυσίδα εφοδιασμού και γεωργία. [4]

Το LoRa είναι μια ασύρματη τεχνολογία που προσφέρει μεγάλη εμβέλεια, χαμηλή ισχύ και ασφαλή μετάδοση δεδομένων για εφαρμογές M2M (machine to machine) και IoT. Το LoRa αφού βασίζεται στη διαμόρφωση του φάσματος chirp spread, έχει χαρακτηριστικά χαμηλής ισχύος όπως η διαμόρφωση FSK αλλά μπορεί να χρησιμοποιηθεί για επικοινωνίες μεγάλου βεληνεκούς. Το LoRa επίσης μπορεί να χρησιμοποιηθεί για τη σύνδεση ασύρματων αισθητήρων, πυλών, μηχανών, συσκευών, ζώων, ανθρώπων κλπ. [5]

Λειτουργεί σε διαφορετικές ζώνες συχνοτήτων σε διαφορετικές περιοχές: Στις Ηνωμένες Πολιτείες λειτουργεί στη ζώνη των 915 MHz, στην Ευρώπη στη ζώνη των **EU433 (433.05-434.79 MHz) και EU863-870 (863-870/873 MHz)** και στην Ασία στη ζώνη των 865 έως 867 MHz, 920 έως 923 MHz.

Στον Πίνακα 2.1 παρουσιάζονται κάποια από τα πιο σημαντικά χαρακτηριστικά της LoRa τεχνολογίας.

Πίνακας 2.1 Χαρακτηριστικά της LoRa τεχνολογίας

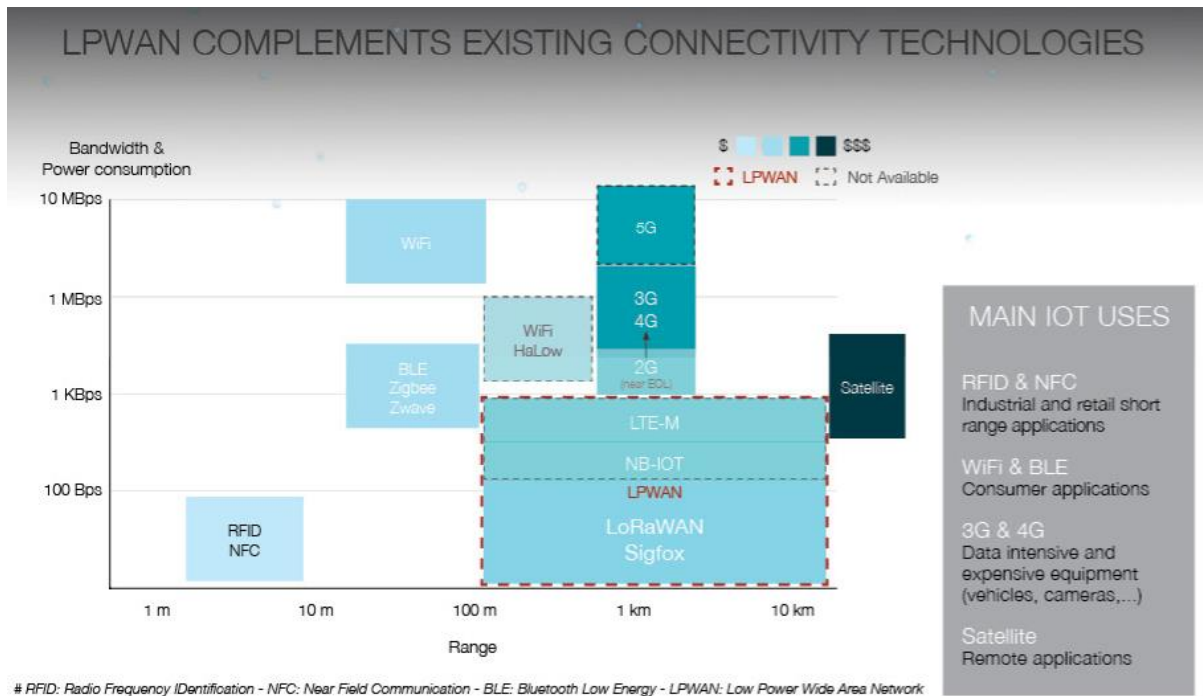
Governing Body	LoRa Alliance
Standard	801.15.4g
Frequency	ISM 433/863 MHz
Range	Up to 15 km (Ανοιχτό)
Datarate	27 kbps
Modulation	Chirp Spread-spectrum modulation type based on FM Modulation Technology
Security	32-bit CRC

2.1.1 LPWAN

Ένα δίκτυο χαμηλής ισχύος ευρείας περιοχής (LPWAN) ή χαμηλής ισχύος δίκτυο ευρείας περιοχής (LPWA) ή δίκτυο χαμηλής ισχύος (LPN) είναι ένας τύπος ασύρματου δικτύου ευρείας περιοχής τηλεπικοινωνιών που έχει σχεδιαστεί για να επιτρέπει επικοινωνίες μεγάλης εμβέλειας σε χαμηλό ρυθμό μετάδοσης μεταξύ των πραγμάτων-συσκευές, όπως αισθητήρες που λειτουργούν με μπαταρία. Η χαμηλή ισχύς, ο χαμηλός ρυθμός bit και η προοριζόμενη χρήση διακρίνουν αυτόν τον τύπο δικτύου από ένα ασύρματο WAN που έχει σχεδιαστεί για να συνδέει χρήστες ή επιχειρήσεις και να μεταφέρει περισσότερα δεδομένα, χρησιμοποιώντας περισσότερη ενέργεια. Ο ρυθμός δεδομένων LPWAN κυμαίνεται από 0,3 kbit/s έως 50 kbit/s ανά κανάλι. [6] [7] [8]

Ένα LPWAN μπορεί να χρησιμοποιηθεί για τη δημιουργία ενός ιδιωτικού ασύρματου δικτύου αισθητήρων, αλλά μπορεί επίσης να είναι μια υπηρεσία ή υποδομή που κάποιος χρησιμοποιεί, επιτρέποντας στους ιδιοκτήτες αισθητήρων να τους αναπτύξουν στο χώρο χωρίς να επενδύσουν σε τεχνολογία πύλης.

Στην Εικόνα 2.1 παρουσιάζεται το σύγκριση εύρους και απόστασης για διάφορες ασύρματες τεχνολογίες.



Εικόνα 2.1: Σύγκριση εύρους και απόστασης

[<https://ipcarrier.blogspot.com/2017/07/does-lpwan-complement-or-compete-with.html>]

Χαρακτηριστικά

Μεγάλη εμβέλεια: Το εύρος λειτουργίας της τεχνολογίας LPWAN είναι από μερικά χιλιόμετρα στις αστικές περιοχές έως και πάνω από 10 χιλιόμετρα σε αγροτικές περιοχές. Μπορεί επίσης να επιτρέψει την αποτελεσματική επικοινωνία δεδομένων σε κάποιες ανέφικτες εσωτερικές ή υπόγειες τοποθεσίες.

Χαμηλή ισχύς: Βελτιστοποιημένοι για κατανάλωση ενέργειας, οι πομποδέκτες LPWAN μπορούν να λειτουργούν με μικρές, φθηνές μπαταρίες για πολλά χρόνια.

Χαμηλό κόστος: Τα απλοποιημένα και ελαφριά πρωτόκολλα της LPWAN μειώνουν την πολυπλοκότητα στο σχεδιασμό υλικού και μειώνουν το κόστος της συσκευής. Το μεγάλο εύρος του σε συνδυασμό με την τοπολογία αστέρων μειώνουν τις δαπανηρές απαιτήσεις υποδομής. Η χρήση ζωνών χωρίς άδεια μειώνει το κόστος δικτύου.

Τεχνολογίες

Ορισμένα ανταγωνιστικά πρότυπα και προμηθευτές στον χώρο LPWAN περιλαμβάνουν: [8]

LoRa-LoRaWAN,

DASH7,

Wize,

Sigfox

MIoTy

NarrowBand IoT (NB-IoT), μια προσπάθεια τυποποίησης από 3GPP για LPWAN που χρησιμοποιείται σε κυψελοειδή δίκτυα

2.1.2 LoRaWAN

Το LoRa είναι το φυσικό στρώμα ή η τεχνική διαμόρφωσης που χρησιμοποιείται για συνδέσεις επικοινωνιών μεγάλης εμβέλειας και χαμηλής ισχύος. Το LoRaWAN είναι ένα πρωτόκολλο επικοινωνίας και αρχιτεκτονική δικτύου που βρίσκεται πάνω από το φυσικό επίπεδο LoRa. [9][10]

Το LoRaWAN είναι ένα δίκτυο (πρωτόκολλο) που χρησιμοποιεί LoRa. Τα LoRa και LoraWAN ανήκουν στην κατηγορία των μη κυψελοειδών πρωτοκόλλων και φορέων ασύρματου δικτύου επικοινωνίας LPWAN, που λειτουργούν στο φάσμα χωρίς άδεια. [11]

Η κατοχυρωμένη με δίπλωμα ευρεσιτεχνίας τεχνολογία ασύρματων ραδιοσυχνοτήτων LoRa αντιπροσωπεύει το πρωτόκολλο φυσικής στρώσης ενώ το LoRaWAN, που αναπτύχθηκε από την LoRa Alliance, σημαίνει το πρωτόκολλο επιπέδου ελέγχου πρόσβασης στα μέσα, το οποίο αξιοποιεί και περιλαμβάνει τη φυσική διαμόρφωση LoRa της Semtech.

Η αρχιτεκτονική δικτύου LoRaWAN αναπτύσσεται σε μια τοπολογία αστέρι-αστέρων, στην οποία οι gateways μεταδίδουν μηνύματα μεταξύ τελικών συσκευών και κεντρικού διακομιστή δικτύου. Οι gateways συνδέονται με τον διακομιστή δικτύου μέσω τυπικών συνδέσεων IP και λειτουργούν ως γέφυρες, μετατρέποντας απλά τα πακέτα RF σε πακέτα IP και αντίστροφα. Η ασύρματη επικοινωνία εκμεταλλεύεται τα χαρακτηριστικά μεγάλης εμβέλειας του φυσικού επιπέδου, επιτρέποντας έναν σύνδεσμο μονής μετάβασης μεταξύ της τελικής συσκευής και μιας ή πολλών gateways. Όλες οι λειτουργίες είναι ικανές για αμφίδρομη επικοινωνία και υπάρχει υποστήριξη για ομάδες διευθύνσεων πολλαπλών εκπομπών για αποτελεσματική χρήση του φάσματος κατά τη διάρκεια εργασιών. [12]

Πολλά υπάρχοντα ανεπτυγμένα δίκτυα χρησιμοποιούν αρχιτεκτονική δικτύου πλέγματος. Σε ένα πλέγμα δίκτυο, οι μεμονωμένοι τελικοί κόμβοι προωθούν τις πληροφορίες άλλων κόμβων για να αυξήσει το εύρος επικοινωνίας και το μέγεθος κυψέλης του δικτύου. Ενώ αυτό αυξάνεται αυξάνει την πολυπλοκότητα, μειώνει τη χωρητικότητα δικτύου και μειώνει την μπαταρία εφόσον οι κόμβοι λαμβάνουν και προωθούν πληροφορίες από άλλους κόμβους που είναι πιθανών άσχετες για αυτούς. Η αρχιτεκτονική αστέρων μεγάλης εμβέλειας έχει μεγαλύτερη χρησιμότητα για τη διατήρηση της διάρκειας ζωής της μπαταρίας όταν μπορεί να επιτευχθεί συνδεσιμότητα μεγάλης εμβέλειας.

Σε ένα LoRaWAN οι κόμβοι δικτύου δεν σχετίζονται με μια συγκεκριμένη πύλη. Αντι αυτού, τα δεδομένα που μεταδίδονται από έναν κόμβο λαμβάνονται συνήθως από πολλές πύλες. Κάθε πύλη θα

προωθήσει το λαμβανόμενο πακέτο από τον τελικό κόμβο στο cloud διακομιστή δικτύου μέσω κινητής τηλεφωνίας, Ethernet, Wi-Fi ή κάτι άλλο.

Η πολυπλοκότητα μεταφέρεται στο διακομιστή δικτύου, ο οποίος διαχειρίζεται το δίκτυο και θα φιλτράρει τα περιττά ληφθέντα πακέτα, πραγματοποιεί ελέγχους ασφαλείας, προγραμματίζει τις αναγνωρίσεις μέσω της βέλτιστης πύλης κλπ.

Διάρκεια ζωής μπαταρίας

Οι κόμβοι σε ένα δίκτυο LoRaWAN είναι ασύγχρονοι και επικοινωνούν όταν είναι έτοιμα τα δεδομένα για αποστολή είτε βάσει συμβάντος είτε προγραμματισμένου. Αυτός ο τύπος πρωτοκόλλου συνήθως αναφέρεται ως μέθοδος Aloha. Σε δίκτυο πλέγματος ή με σύγχρονο δίκτυο, όπως το κυψελοειδές, οι κόμβοι πρέπει συχνά να «ξυπνούν» για συγχρονισμό. Αυτός ο συγχρονισμός καταναλώνει σημαντικά ποσά ενέργειας και είναι ο νούμερο ένα παράγοντας μείωσης της διάρκειας ζωής της μπαταρίας.

Για να καταστεί βιώσιμο ένα δίκτυο αστερών μεγάλης εμβέλειας, η πύλη πρέπει να έχει πολύ υψηλή ικανότητα ή δυνατότητα λήψης μηνυμάτων από πολύ μεγάλο όγκο κόμβων. Αυτό επιτυγχάνεται με τη χρήση προσαρμοστικών δεδομένων δηλαδή να προσαρμοστεί ο ρυθμός και ο διμρασμός με τη χρήση ενός πολυκαναλου πομποδέκτη πολλαπλών μόντεμ στην πύλη έτσι ώστε μπορούν να ληφθούν ταυτόχρονα μηνύματα σε πολλά κανάλια. Οι κρίσιμοι παράγοντες για αυτήν την ικανότητα είναι ο αριθμός των ταυτόχρονων καναλιών, ο ρυθμός δεδομένων, το μήκος ωφέλιμου φορτίου και πόσο συχνά μεταδίδονται οι κόμβοι. Η πύλη εκμεταλλεύεται τη δυνατότητα λήψης με πολλαπλούς διαφορετικούς ρυθμούς δεδομένων στο ίδιο κανάλι ταυτόχρονα. Ο προσαρμοζόμενος ρυθμός δεδομένων βελτιστοποιεί επίσης τη διάρκεια ζωής της μπαταρίας του κόμβου. Αυτές οι δυνατότητες δίνουν σε ένα LoRaWAN να έχει πολύ υψηλή χωρητικότητα και να κάνει το δίκτυο επεκτάσιμο. Ένα δίκτυο μπορεί να αναπτυχθεί με ελάχιστο όγκο υποδομής και χωρητικότητα. Επίσης, μπορούν να προστεθούν περισσότερες πύλες, αλλάζοντας τους ρυθμούς δεδομένων.

Ασφάλεια

Το LoRaWAN χρησιμοποιεί δύο επίπεδα ασφάλειας: ένα για το δίκτυο και ένα για την εφαρμογή.

Η ασφάλεια δικτύου διασφαλίζει την αυθεντικότητα του κόμβου στο δίκτυο ενώ το επίπεδο ασφάλειας εφαρμογής διασφαλίζει ότι ο χειριστής δικτύου δεν έχει πρόσβαση στα δεδομένα εφαρμογής του τελικού χρήστη. Η κρυπτογράφηση AES χρησιμοποιείται με την ανταλλαγή κλειδιών με ένα αναγνωριστικό IEEE EUI64.

Από δίκτυα LPWAN το LoRaWAN είναι το πιο γνωστό αλλά υπάρχουν και άλλα αξιόλογα όπως το NB-IoT. Άλλα παρόμοια δίκτυα χαμηλής ενέργειας ευρείας περιοχής είναι τα NB-IoT, LTE-M, SigFox και συγκριτικά αποτελέσματα μπορεί ο αναγνώστης να βρει στην πηγή [13][14].

Το Narrowband Internet of things (NB-IoT) είναι ένα πρότυπο ραδιοφωνικής τεχνολογίας χαμηλής ισχύος δίκτυο ευρείας περιοχής (LPWAN) που αναπτύχθηκε από την 3GPP για κυψελοειδείς συσκευές και υπηρεσίες. Οι προδιαγραφές έγιναν από 3GPP Release 13 (LTE Advanced Pro), τον Ιούνιο του 2016.

Το NB-IoT επικεντρώνεται ειδικά στην εσωτερική κάλυψη, το χαμηλό κόστος, τη μεγάλη διάρκεια ζωής της μπαταρίας και την υψηλή πυκνότητα σύνδεσης. Το NB-IoT χρησιμοποιεί ένα υποσύνολο του προτύπου LTE, αλλά περιορίζει το εύρος ζώνης σε μία μόνο στενή ζώνη 200kHz. Οι εφαρμογές IoT που απαιτούν συχνότερες επικοινωνίες θα εξυπηρετούνται καλύτερα από το NB-IoT, το οποίο δεν έχει περιορισμούς κύκλου λειτουργίας που λειτουργούν στο φάσμα με άδεια χρήσης [15].

2.1.3 Μετρούμενα μεγέθη

Στην εργασία μελετήθηκαν δύο περιπτώσεις και υλοποιήθηκαν πειραματικά. Η μια αφορά την χρήση δυο απλών και φθηνών συσκευών LoRa συνδεδεμένα σε Raspberry και η μέτρηση κάποιων από των χαρακτηριστικών επικοινωνίας τους. Η δεύτερη περίπτωση αφορά την εγκατάσταση ενός δικτύου επικοινωνίας με τεχνικές ασφαλείας.

Στην πρώτη περίπτωση πραγματοποιήθηκαν μετρήσεις που αφορούν τα εξής μεγέθη:

- Packet Loss Ratio
- RSSI
- SNR

Σε αυτήν την παράγραφο θα περιγράψουμε θεωρητικά αυτά τα μεγέθη που μετράμε στο Κεφάλαιο 4.

2.1.3.1 Packet Loss Ratio

Ο λόγος απώλειας πακέτων αντιπροσωπεύει την αναλογία του αριθμού των χαμένων πακέτων προς το συνολικό αριθμό των αποσταλμένων πακέτων. Κάθε πακέτο έχει προθεσμία πριν από την οποία πρέπει να εκτελεστεί και εάν αυτό δεν είναι δυνατό, ο προγραμματιστής προσπαθεί να ελαχιστοποιήσει τον αριθμό των χαμένων πακέτων λόγω λήξης της προθεσμίας. Οι εγγυήσεις για την επίτευξη αυτών των περιορισμών χρονισμού και πώς το σύστημα χειρίζεται εκείνα τα πακέτα που δεν μπορούν να τηρήσουν την προθεσμία τους είναι στόχοι ενός αλγορίθμου προγραμματισμού. Το κλάσμα των πακέτων που

έχουν χαθεί, λόγω απώλειας της προθεσμίας, χρησιμοποιείται για την αξιολόγηση της απόδοσης ζημίας ενός αλγορίθμου προγραμματισμού.. Όποτε όταν αυτό το κλάσμα είναι μικρό, ο αλγόριθμος είναι κατάλληλος για την υλοποίηση σε πραγματικό χρόνο. [16]

2.1.3.2 RSSI

Η Ένδειξη Ισχύος Λήψης Σήματος (RSSI- Received Signal Strength Indicator) είναι η ισχύς του λαμβανόμενου σήματος σε milliwatt και μετριέται σε dBm. [17][18]

Αυτή η τιμή μπορεί να χρησιμοποιηθεί ως μέτρηση του πόσο καλά ένας δέκτης μπορεί να "ακούσει" ένα σήμα από έναν αποστολέα. Το RSSI μετριέται σε dBm και είναι αρνητική τιμή. Όσο πιο κοντά στο 0 τόσο καλύτερο είναι το σήμα.

Οι τυπικές τιμές LoRa RSSI είναι:

Ελάχιστο RSSI = -120 dBm.

πχ

Εάν RSSI = -20dBm: το σήμα είναι ισχυρό.

Εάν RSSI = -110dBm: το σήμα είναι ασθενές.

2.1.3.3 SNR

Ο λόγος σήματος προς θόρυβο (SNR- Signal-to-noise ratio) είναι η αναλογία μεταξύ του λαμβανόμενου σήματος ισχύος και του επιπέδου ισχύος του δαπέδου του θορύβου. [17]

Το δάπεδο θορύβου είναι μια περιοχή όλων των ανεπιθύμητων πηγών σήματος παρεμβολής που μπορούν να καταστρέψουν το μεταδιδόμενο σήμα και συνεπώς θα επανεκδοθούν.

Εάν το SNR είναι μεγαλύτερο από 0, το λαμβανόμενο σήμα λειτουργεί πάνω από το δάπεδο θορύβου.

Εάν το SNR είναι μικρότερο από 0, το λαμβανόμενο σήμα λειτουργεί κάτω από το δάπεδο θορύβου.

Κανονικά το δάπεδο θορύβου είναι το φυσικό όριο ευαισθησίας, ωστόσο το LoRa λειτουργεί κάτω από το επίπεδο θορύβου.

2.2 Παρόμοια συστήματα

Τελευταία έχουν πραγματοποιηθεί συστήματα μέτρησης απόστασης με LoRa. [19][20][21]

Η εργασία του Augustin A και λοιπών [22] παρέχει μια επισκόπηση του LoRa και μια σε βάθος ανάλυση των λειτουργικών συστατικών του. Η απόδοση του επιπέδου φυσικής και δεδομένων σύνδεσης αξιολογείται με δοκιμές πεδίου και προσομοιώσεις. Με βάση την ανάλυση και τις αξιολογήσεις, προτείνονται κάποιες πιθανές λύσεις για βελτιώσεις της απόδοσης

Η ομάδα του Dieng [23] προτείνουν μια απόσταση που βασίζεται σε RSSI μέτρηση για τον εντοπισμό των περιλαίμων σε βοοειδή που επικοινωνούν ασύρματα με μεγάλης εμβέλειας LoRa. Ο εντοπισμός βοοειδών έχει σχεδιαστεί για την πρόληψη κλοπών σε ζώα. Η προτεινόμενη λύση μειώνει το κόστος εντοπισμού βοοειδών ελαχιστοποιώντας τον αριθμό των περιλαίμων με GPS και επιτρέπει τον ακριβή εντοπισμό των περιλαίμων χωρίς GPS. Προτείνουν μια εκτίμηση απόστασης βάσει RSSI χρησιμοποιώντας προσαρμογή σε πραγματικό χρόνο της χαρτογράφησης RSSI απόστασης, αξιοποιώντας επικοινωνία μεταξύ κόμβων κολάρου και πύλης. Το μοντέλο διαδρομής-απώλειας χρησιμοποιείται επίσης ως διάσωση όταν ο χάρτης δεν το κάνει. Τα αποτελέσματα του πειράματος δείχνουν την εγκυρότητα της προσέγγισης με εξαιρετικά ακριβή εντοπισμό των περιλαίμων χωρίς GPS.

Ο κύριος στόχος αυτής της εργασίας είναι να γίνει μια εκτίμηση της απόστασης που μπορούν να επικοινωνούν δύο φθηνά χαμηλής ισχύος όπως το SX1278.

Έχουν αναφερθεί αρκετές εργασίες σε μετρήσεις με SX1278. Οι M. Aref and A. Sikora [24] με την οικογένεια SX127 παρουσίασαν αποτελέσματα που αφορούν την μεταξύ τους απόσταση και την απώλεια πακέτων κατά τη μετάδοση αλλά και μετρήσεις για το RSSI.

Οι Gui και λοιποί περιγράφουν τη διαδικασία ανάπτυξης του LoRa, εισάγουν τα κύρια χαρακτηριστικά της ασύρματης τεχνολογίας LoRa, σχεδίασαν κύκλωμα ασύρματου πομποδέκτη βασισμένο στο τσιπ SX1278 της εταιρείας Semtech και ανάλυσαν τις προοπτικές εφαρμογής και την προοπτική της τεχνολογίας LoRa στον τομέα των πλοίων. [25]

Οι Wang και λοιποί εστίασαν στην υλοποίηση ενός μικρού ασύρματου συστήματος παρακολούθησης ασύρματου ad hoc αισθητήρα δικτύου βασισμένου στο LoRa (Long Range) το οποίο παρέχει ένα σύστημα παρακολούθησης μεγάλων αποστάσεων χαμηλού κόστους σε πραγματικό χρόνο. Το σύστημα χρησιμοποιεί το Heltec WiFi LoRa 32 ως πύλη και κόμβος. Υιοθετεί το τσιπ ESP32 ως MCU το οποίο έχει υψηλό επίπεδο χαμηλής απόδοσης στη βιομηχανία. Το σύστημα ενσωματώνει τόσο τη μονάδα LoRa (SX1278) όσο και τη μονάδα WiFi, που κάνουν το σύστημα να συνδυάζει τη μετάδοση μεγάλων

αποστάσεων και τη χαμηλή κατανάλωση ενέργειας της επικοινωνίας LoRa με την υποστήριξη επικοινωνίας δικτύου TCP/IP. Το σύστημά μας μπορεί να παρέχει ένα φιλικό προς το χρήστη μπροστινό μέρος που είναι βολικό για την παρακολούθηση του συστήματος σε πραγματικό χρόνο από τους χρήστες. Χρησιμοποιούνε ένα σύνολο πρωτοκόλλων δικτύου αυτοοργάνωσης για να ικανοποιήσουν τις απαιτήσεις της καθημερινής παρακολούθησης και προειδοποίησης έκτακτης ανάγκης.

Από τις πιο σημαντικές εργασίες είναι του Shuhaizar Daud [27] και της ομάδας του όπου αξιολογεί ορισμένες μονάδες χαμηλού κόστους LoRa που διατίθενται στην αγορά και την καταλληλότητά τους, την ενεργειακή απόδοση και την απόδοση κατά τη λειτουργία. Δύο χαμηλού κόστους πομποδέκτες LoRa από την Semtech Industries, οι SX1272 και SX1278 δοκιμάστηκαν για την κατανάλωση ενέργειας και τη μέγιστη εμβέλεια μετάδοσης. Αυτή η μελέτη αξιολόγησε τις δύο λύσεις LoRa και διαπίστωσε ότι το SX1278 έχει καλύτερη εμβέλεια μετάδοσης και χρησιμοποιεί χαμηλότερη ενέργεια σε σύγκριση με το SX1272 καθιστώντας το πιο κατάλληλο για ενσωματωμένη εφαρμογή ως πύλη δεδομένων.

Όσον αφορά τη δεύτερη προσέγγιση που πραγματοποιήθηκε για την υλοποίηση ενός μικρού δικτύου κόμβων που επικοινωνούν με έναν κεντρικό server-LoRa ή/και με έναν server-PC μέσω διαδικτύου σε θερμοκήπια ή αγροτικές περιοχές έχουν αναφερθεί αξιολογικές εργασίες, όπως του Vu και λοιπών. [28]

Η μελέτη τους αφορά τον σχεδιασμό αυτόματου συστήματος άρδευσης για τη γεωργία θερμοκηπίου με βάση την τεχνολογία LoRa με εξαιρετικά πλεονεκτήματα όσον αφορά το εύρος μετάδοσης και την κατανάλωση ενέργειας. Το σύστημα αποτελείται από κόμβους αισθητήρων που συλλέγουν τα δεδομένα της υγρασίας, της θερμοκρασίας και της υγρασίας του εδάφους. Αυτά τα δεδομένα διαβιβάζονται σε έναν κεντρικό σταθμό που ονομάζεται Συμπυκνωτής. Ο Συμπυκνωτής θα τα χρησιμοποιήσει για τον έλεγχο της διαδικασίας άρδευσης μέσω των κόμβων ελέγχου που βρίσκονται στο επίπεδο του πεδίου. Τα δεδομένα αποστέλλονται επίσης στον Υπολογιστή Επόπτη και στον Διακομιστή Ιστού για αποθήκευση για μελλοντική ανάλυση. Οι χρήστες μπορούν να παρακολουθούν, να ρυθμίζουν τη λειτουργία άρδευσης, να διαμορφώνουν το σύστημα, να έχουν πρόσβαση σε δεδομένα και να διαχειρίζονται καλλιέργειες τόσο μέσω της διασύνδεσης υπολογιστή όσο και της διεπαφής ιστού.

Ο Thair και λοιποί [29] χρησιμοποίησαν την τεχνολογία LoRa ως την ραχοκοκαλιά του σχεδιασμού για το σύστημα ασύρματου δικτύου για την παρακολούθηση και τον έλεγχο αισθητήρων θερμοκηπίου. Ο σχεδιασμός αποτελείται από δύο μέρη: Το πρώτο είναι να προετοιμάσει μια ομάδα αισθητήρων που υποστηρίζουν το (IoT), μετρήσεις και αποστολή δεδομένων χρησιμοποιώντας την τεχνολογία LoRa. Υλοποιεί παρακολούθηση και έλεγχο των δεδομένων, χρησιμοποιώντας την τεχνολογία LoRa για τη λήψη δεδομένων και τη σύνδεσή τους στο Διαδίκτυο. Οι αναγνώσεις μπορούν να ληφθούν μετά τη

μεταφόρτωσή τους στο cloud computing και τα δεδομένα προβάλλονται σε έναν ιστότοπο που επιτρέπει την παρακολούθηση των αισθητήρων όπου είναι διαθέσιμη η υπηρεσία διαδικτύου. Το σύστημα δίνει επίσης τη δυνατότητα αποστολής σήματος ανατροφοδότησης στο θερμοκήπιο. Το έργο παρέχει ολοκληρωμένο έλεγχο θερμοκηπίου σε απόσταση 2 έως 15 χλμ.

Υπάρχουν και εργασίες με LoRa με εφαρμογή αυτοματισμού σε σπίτια όπως του Rahabu και λοιποί [30] που προτείνει μια αρχιτεκτονική αυτοματισμού σπιτιού τόσο για μικρής εμβέλειας όσο και για μεγάλης εμβέλειας χρησιμοποιώντας πολλαπλές τεχνολογίες επικοινωνίας, συγκεκριμένα LoRaWAN, πύλη LoRa που βασίζεται σε διακομιστές και συνδεσιμότητα Bluetooth. Αυτό το ολοκληρωμένο σύστημα ελέγχει διαφορετικούς τύπους οικιακών συσκευών και κάνει έξυπνη διαχείριση σε όλα τα ηλεκτρονικά εξαρτήματα. Ένας κανονικός χρήστης μπορεί να διαχειριστεί αυτά τα ενοποιημένα συστήματα χρησιμοποιώντας μια εφαρμογή Android. Αυτή η εργασία παρουσιάζει επίσης πειραματική ανάλυση δεδομένων. Η ενότητα αποτελεσμάτων περιέχει ένα σύνολο πειραμάτων όπως εκτιμώμενος υπολογισμός καθυστέρησης μετάδοσης για LoRa, Wi-Fi και Bluetooth, υπολογισμός περιοχής κάλυψης για LoRa με τιμές RSSI και SNR και κλίμακα χρηστικότητας συστήματος (SUS). Το πρόγραμμα έχει επιτύχει βαθμολογία SUS 93%.

2.3 LoRa modules

Στην αγορά υπάρχουν διάφορες συσκευές LoRa και στην εργασία του Rudy Susanto [31] παρουσιάζονται συγκριτικά αποτελέσματα από τρεις συσκευές. Από τις συσκευές UM402, Dragino και RA-02 συμπεράνε ότι το UM402 έχει τη μεγαλύτερη απόσταση έως 1 χλμ. Μέχρι 250 m το LoRa Dragino είναι η γρηγορότερο με χρόνο παράδοσης πακέτων δεδομένων σε 0,77 δευτερόλεπτα. Το UM402 LoRa έχει τη μεγαλύτερη τιμή RSSI στα 50 m, περίπου -34. Όλες οι συσκευές LoRa λειτουργούν το ίδιο για απόσταση <35 m.

Χρησιμοποιήθηκε το SX1278 lora module επειδή είναι φθινό και μικρό, μπορεί να βρεθεί εύκολα στην Ελληνική αγορά και υπάρχουν βιβλιοθήκες για σύνδεση με το raspberry και esp32.

Το Ra-02 ολοκληρωμένο που χρησιμοποιεί είναι μια μονάδα ασύρματης μετάδοσης που βασίζεται στον ασύρματο πομποδέκτη SEMTECH. Έχει προηγμένη τεχνολογία διασποράς φάσματος LoRa, με απόσταση επικοινωνίας 10 χιλιομέτρων. Το SX1278 που το χρησιμοποιεί έχει υψηλή ευαισθησία -148 dBm με ισχύ εξόδου +20 dBm, και μεγάλη απόσταση μετάδοσης και υψηλή αξιοπιστία.



Εικόνα 2.2: SX1278 LoRa module

[<https://www.e-wireless.gr/sx1278-lora-module-433m-ra-02-wireless-module/>]

Μπορεί να καλύπτει χιλιάδες ανθρώπους στο περιβάλλον της περιοχής, ιδιαίτερα κατάλληλο για ανάγνωση μετρητών, έξυπνο σπίτι, εξοπλισμό συναγερμού κτλ

Χαρακτηριστικά [32]:

Απόσταση επικοινωνίας: 15KM

Ευαισθησία: έως -148dBm

Προγραμματιζόμενοι ρυθμοί bit: έως 300kbps

Δυναμικό εύρος RSSI: 127dB

Ασύρματη συχνότητα: 433MHz

Τάση λειτουργίας: 1,8 - 3,7v

Θερμοκρασία λειτουργίας: -40 - 80

Διαστάσεις: 28x20,3x5mm

Τεχνολογία διαμόρφωσης LoRa™ Spread Spectrum

Σταθερή ισχύς RF σε αλλαγή τάσης + 20dBm-100mW

Ημι-διπλή επικοινωνία SPI

Υποστηρίζει λειτουργίες διαμόρφωσης FSK, GFSK, MSK, GMSK, LoRa και OOK

Αυτόματη ανίχνευση σήματος RF, λειτουργία CAD και AFC πολύ υψηλής ταχύτητας

Η κεραία που χρησιμοποιήθηκε μαζί με το SX1278 είναι η
Antenna RF 434MHz 2dBi 63.6mm u.FL



Εικόνα 2.3: Antenna RF 434MHz 2dBi 63.6mm u.FL για το SX1278 LoRa module

Με τα παρακάτω χαρακτηριστικά:

Είδος κεραίας: RF

Συχνότητα: 433.050 ... 434.790MHz

Θερμοκρασία λειτουργίας: -40 ... 85 ° C

Κέρδος κεραίας: 2dBi

Είδος σύνδεσης: I-PEX (u.FL)

Αντίσταση: 50Ω

Τοποθέτηση: Τύπος βίδας

Πόλωση: Γραμμική

Είδος σύρματος: RG178

2.4 Raspberry

Το Raspberry Pi, είναι ένας αποδοτικός και οικονομικός υπολογιστής μεγέθους πιστωτικής κάρτας, με ένας από τους στόχους του είναι να διαφωτίσει και να ενδυναμώσει τη διδασκαλία της επιστήμης των υπολογιστών σε σχολεία και άλλες αναπτυσσόμενες χώρες. Από την ίδρυσή του, διάφορες κοινότητες ανοιχτού κώδικα έχουν συνεισφέρει σε εφαρμογές ανοιχτού κώδικα, λειτουργικά συστήματα και διάφορους άλλους υπολογιστές μικρού παράγοντα παρόμοιου με το Raspberry Pi. Μέχρι σήμερα, πολλοί ερευνητές και άλλοι λάτρεις των ενσωματωμένων συστημάτων σε ολόκληρο τον πλανήτη υλοποιούν καταπληκτικά έργα χρησιμοποιώντας το Pi Το Raspberry Pi από την έναρξή του βρίσκεται τακτικά υπό συνεχή ανάπτυξη με βελτίωση τόσο από πλευράς υλικού όσο και λογισμικού, το οποίο με τη σειρά του καθιστά το Pi έναν «Πλήρη Υπολογιστή» με δυνατότητα να λαμβάνεται υπόψη για σχεδόν όλες τις εργασίες υπολογιστικής. [33]

Μέχρι σήμερα, έχουν πωληθεί 5 εκατομμύρια κομμάτια διαφόρων γενεών Raspberry Pi που κυμαίνονται από το μοντέλο A έως το μοντέλο B και ακόμη και το πιο πρόσφατο Raspberry Pi Zero και χιλιάδες έργα που κυμαίνονται από ρομποτική, ανάπτυξη, κονσόλες παιχνιδιών, tablet, εφαρμογές για Linux. Τα θεμέλια για την ανάπτυξη του Raspberry Pi ξεκίνησε το 2006, από τους ερευνητές: Eben Upton, Rob Mullins, Jack Lang και Alan Mycroft στο Εργαστήριο Υπολογιστών του Πανεπιστημίου του Cambridge. Η κύρια ιδέα πίσω από την εξέλιξή τους ήταν να δώσουν στα παιδιά ένα μικροσκοπικό και προσιτό υπολογιστή την περίοδο όπου οι υπολογιστές ήταν ακριβοί και η πρακτική προγραμματισμού μεταξύ των παιδιών δεν υποστηριζόταν από γονείς των παιδιών. Η ομάδα με επικεφαλής τον Eben Upton ανέπτυξε διάφορες εκδόσεις πρωτοτύπων εργασίας από έτος 2006 έως 2008 και η τελική έκδοση που κυκλοφόρησε ονομάστηκε "RASPBerry Pi".

Το 2008, οι επεξεργαστές που χρησιμοποιήθηκαν σε φορητές συσκευές έγιναν φθηνότεροι και ισχυρότεροι και είχαν πλήρη δυνατότητα να υποστηρίξουν και να εκτελέσουν πολυμέσα και κάθε είδους προγραμματισμό. Το Raspberry Pi δημιουργήθηκε επίσημα το έτος 2012 και ο αναπτυσσόμενος λίθος τοποθετήθηκε από το ίδρυμα Raspberry Pi και μέσα σε 3 χρόνια, το μοντέλο B μπήκε σε μαζική παραγωγή.

Το Raspberry Pi διαθέτει εξαιρετική υπολογιστική ισχύ και ικανό να αναπτύξει εκπληκτικά έργα. Ο υπολογιστής κοστίζει από 5 ευρώ έως 50 ευρώ και είναι τέλειος για την εκτέλεση κάθε είδους υπολογιστικών εργασιών και διασύνδεσης διαφόρων ειδών συσκευών μέσω GPIO. Το Raspberry Pi περιέχει επεξεργαστή ARM με βάση το Broadcom, τσιπ γραφικών, RAM, GPIO και άλλες δυνατότητες σύνδεσης για εξωτερικές συσκευές. Η διαδικασία λειτουργίας του Raspberry Pi είναι πολύ παρόμοια σε σύγκριση με τον υπολογιστή και απαιτεί πρόσθετο υλικό όπως πληκτρολόγιο, ποντίκι, μονάδα οθόνης, τροφοδοσία, κάρτα SD με εγκατεστημένο λειτουργικό σύστημα για λειτουργία. Το Raspberry Pi έχει επίσης τις θύρες USB, Ethernet για Internet/Network. Το Raspberry Pi έχει βάση το λειτουργικό σύστημα ανοιχτού κώδικα το Linux. Μέχρι σήμερα έχουν ξεκινήσει περισσότερα από 30 λειτουργικά

συστήματα που βασίζονται σε διαφορετικές εκδόσεις Linux. Το ίδρυμα Raspberry Pi κυκλοφόρησε επίσης διάφορα αξεσουάρ όπως Camera, Gertboard και Compute Model Kit για την ανάπτυξη πρόσθετων μονάδων υλικού. [34][35]

Το Raspberry Pi περιλαμβάνει τα ακόλουθα στοιχεία:

- CPU: Quad-core 64-bit ARM Cortex A53 clocked at 1.2 GHz
- GPU: 400MHz VideoCore IV multimedia
- Memory: 1GB LPDDR2-900 SDRAM (i.e. 900MHz)
- USB ports: 4
- Video outputs: HDMI, composite video (PAL and NTSC) via 3.5 mm jack
- Network: 10/100Mbps Ethernet and 802.11n Wireless LAN
- Peripherals: 17 GPIO plus specific functions, and HAT ID bus
- Bluetooth: 4.1
- Power source: 5 V via MicroUSB or GPIO header
- Size: 85.60mm × 56.5mm
- Weight: 45g (1.6 oz)

2.5 ThingSpeak

Το ThingSpeak είναι μια υπηρεσία Web (REST API) που σας επιτρέπει να συλλέγετε και αποθηκεύστε δεδομένα αισθητήρων στο cloud και αναπτύξτε Internet of Things εφαρμογές.[36][37]

Λειτουργεί με Arduino, Raspberry Pi και MATLAB (προπαρασκευασμένο υπάρχουν βιβλιοθήκες και API). Λειτουργεί με κάθε είδους γλώσσας προγραμματισμού, αφού χρησιμοποιεί REST API και HTTP.

Η υπηρεσία ThingSpeak σας επιτρέπει επίσης να πραγματοποιείτε διαδικτυακή ανάλυση και εκτελέσετε ενέργειες ανάλυσης με βάση τα δεδομένα σας. Τα δεδομένα αισθητήρα μπορούν να σταλούν στο ThingSpeak από οποιοδήποτε υλικό που μπορεί να επικοινωνήσει χρησιμοποιώντας ένα REST API Το ThingSpeak είναι μια υπηρεσία Web (REST API) που σας επιτρέπει να συλλέγετε και αποθηκεύστε δεδομένα αισθητήρων στο cloud και αναπτύξτε Internet of Things εφαρμογές.

Δημιουργήστε έναν λογαριασμό ThingSpeak επισκεπτόμενοι τη διεύθυνση URL:

<https://thingspeak.com/>

Κάντε κλικ στο κουμπί "Ξεκινήστε δωρεάν" και εγγραφείτε ως νέος χρήστης.

Δημιουργήστε ένα νέο κανάλι κάνοντας κλικ στο κουμπί όπως φαίνεται στην παρακάτω Εικόνα 2.4.




thingspeak.com/channels

ThingSpeak™ Channels Apps Devices St

My Channels

New Channel

Search by tag

Name
 bme/bmp280
Private Public Settings Sharing API Keys Data Import / Export
 liappis
Private Public Settings Sharing API Keys Data Import / Export
 IoT
Private Public Settings Sharing API Keys Data Import / Export

Εικόνα 2.4: ThingSpeak – Κανάλια – Νέο Κανάλι

Ένα κανάλι είναι πηγή για τα δεδομένα σας. Πού μπορείτε να αποθηκεύσετε και να ανακτήσετε δεδομένα. Ένα κανάλι μπορεί να έχει το πολύ 8 πεδία. Αυτό σημαίνει ότι μπορείτε να αποθηκεύσετε 8 διαφορετικά δεδομένα σε ένα κανάλι. Δημιουργήστε κανάλια για να αποθηκεύσετε δεδομένα από αισθητήρες. Το αναγνωριστικό καναλιού είναι η ταυτότητα του καναλιού σας, όπως φαίνεται στην Εικόνα 2.5.

IoT

Channel ID: **1527755**

Author: **korkonas**

Access: Private

Private View Public View Channel Settings Sharing API Keys Data Import / Export

+ Add Visualizations + Add Widgets Export recent data

Channel Stats

Created: **4 days ago**

Last entry: **4 days ago**

Entries: 2

Εικόνα 2.5: Ρυθμίσεις καναλιού

Τα κλειδιά API (Διεπαφή προγραμματισμού εφαρμογών) είναι τα κλειδιά για πρόσβαση στο κανάλι σας, όπως φαίνεται στην Εικόνα 2.6.

Author: korkonas
Access: Private

Private View Public View Channel Settings Sharing API Keys Data Import / Export

Write API Key

Key: T5A2HCB80Z4FHV8L

Generate New Write API Key

Read API Keys

Key: U6EH75GL3TC9967I

Note:

Save Note Delete API Key

Help

API keys enable you to write data to a channel or read data from a private channel. API keys are auto-generated when you create a new channel.

API Keys Settings

- Write API Key:** Use this key to write data to a channel. If you feel your key has been compromised, click **Generate New Write API Key**.
- Read API Keys:** Use this key to allow other people to view your private channel feeds and charts. Click **Generate New Read API Key** to generate an additional read key for the channel.
- Note:** Use this field to enter information about channel read keys. For example, add notes to keep track of users with access to your channel.

API Requests

Write a Channel Feed

```
GET https://api.thingspeak.com/update?api_key=T5A2HCB80Z4FHV8L&field=
```

Read a Channel Feed

```
GET https://api.thingspeak.com/channels/1527755/feeds.json?api_key=U
```

Εικόνα 2.6: Τα κλειδιά API

Πρόκειται για κωδικούς πρόσβασης για πρόσβαση στο κανάλι σας. Μπορείτε να έχετε πρόσβαση στο κανάλι σας με δύο τρόπους.

A. Για ενημέρωση της καταγραφής καναλιών / δεδομένων: Το κλειδί εγγραφής API θα χρησιμοποιηθεί για πρόσβαση σε αυτήν τη λειτουργία.

B. Για ανάκτηση δεδομένων: Το κλειδί ανάγνωσης API θα χρησιμοποιηθεί για πρόσβαση σε αυτήν τη λειτουργία.

2.6 Python

Η Python είναι μια εύχρηστη, ισχυρή γλώσσα προγραμματισμού. Διαθέτει αποτελεσματικές δομές δεδομένων υψηλού επιπέδου και μια απλή αλλά αποτελεσματική προσέγγιση στον αντικειμενοστραφή προγραμματισμό. Η σύνταξη και η δυναμική πληκτρολόγηση της Python την καθιστούν ιδανική γλώσσα για script ενεργειών και γρήγορη ανάπτυξη εφαρμογών σε πολλούς τομείς στις περισσότερες πλατφόρμες.[38][39]

Ο διερμηνέας Python και η εκτεταμένη τυπική βιβλιοθήκη διατίθενται ελεύθερα σε πηγή ή δυαδική μορφή για όλες τις κύριες πλατφόρμες από την ιστοσελίδα της Python και μπορούν να διανεμηθούν ελεύθερα. Ο ίδιος ιστότοπος περιέχει επίσης διανομές και υποδείξεις σε πολλές δωρεάν ενότητες, προγράμματα και εργαλεία Python τρίτων μερών και πρόσθετη τεκμηρίωση.

Ο διερμηνέας Python επεκτείνεται εύκολα με νέες λειτουργίες και τύπους δεδομένων που εφαρμόζονται σε C ή C ++.

Η Python είναι μια ευρέως χρησιμοποιούμενη γλώσσα προγραμματισμού με εύκολη, φιλική για αρχάριους σύνταξη (διάταξη λέξεων, φράσεων, σε προτάσεις). Λόγω της ευρείας υιοθέτησής του, έχει μια τεράστια κοινότητα, παρέχοντας πρόσβαση σε βιβλιοθήκες, πλαίσια και εργαλεία για να βοηθήσει τους χρήστες να ξεκινήσουν.

Δεδομένου ότι η Python είναι μια γλώσσα προγραμματισμού πολλαπλών χρήσεων, μπορεί να χρησιμοποιηθεί για πολλές χρήσιμες εφαρμογές σε διαφορετικούς κλάδους, όπως:

Ανάπτυξη Ιστού και εφαρμογές

Μηχανική μάθηση και Τεχνητή Νοημοσύνη

Ανάπτυξη παιχνιδιών

Δημιουργία εφαρμογών GUI

Αυτοματοποίηση

Έργα όπως ενεργοποίηση και απενεργοποίηση ηλεκτρονόμενων κτλ

Raspberry και Python

Για την εκτέλεση της Python οποιασδήποτε πλατφόρμα απαιτείται ένας διερμηνέας Python και αφού το Raspberry Pi έχει λειτουργικό σύστημα Linux, η εκτέλεση του Python Interpreter δεν παρουσιάζει προβλήματα.

Αυτό που τρέχει (ή υπερέχει) η python στο Raspberry Pi αντί για τον τυπικό υπολογιστή είναι ότι μπορείτε να εκτελέσουμε κώδικα ειδικά για τον έλεγχο εξαρτημάτων υλικού μέσω των ακροδεκτών GPIO.

2.7 Flask

Το Flask είναι ένα API της Python που μας επιτρέπει να δημιουργούμε διαδικτυακές εφαρμογές[40]. Το πλαίσιο του Flask είναι πιο σαφές από το πλαίσιο του Django και είναι επίσης ευκολότερο να το μάθετε επειδή έχει λιγότερο βασικό κώδικα για την υλοποίηση μιας απλής διαδικτυακής εφαρμογής. [41]. Το Web-Application Framework ή το Web Framework είναι η συλλογή ενοτήτων και βιβλιοθηκών που βοηθούν τον προγραμματιστή να γράφει εφαρμογές χωρίς να γράφει κώδικα χαμηλού επιπέδου, όπως πρωτόκολλα, διαχείριση νήματος κ.λπ. Το Flask βασίζεται στην εργαλειοθήκη WSGI (Web Server Gateway Interface) και πρότυπο κινητήρα Jinja2.

Για την εγκατάσταση του Flask απαιτείται Python 2.6 ή νεότερη έκδοση. Μπορούμε να ξεκινήσουμε με την εισαγωγή Flask από το πακέτο σε οποιοδήποτε IDE python.

Ξεκινάμε με την γραφή του παρακάτω κομματιού κώδικα.

```
from flask import Flask
app = Flask(myname)
```

```
@app.route('/')
def hi():
    return 'Hi'
```

```
if myname == '__main__':
    app.run()
```

Το URL '/' συνδέεται με τη συνάρτηση hi(). Όταν ανοίξει η αρχική σελίδα του διακομιστή ιστού στο πρόγραμμα περιήγησης, η έξοδος αυτής της προβληθεί.

Η εφαρμογή Flask ξεκινά καλώντας τη συνάρτηση `run()`. Η μέθοδος θα πρέπει να επανεκκινηθεί με μη αυτόματο τρόπο για οποιαδήποτε αλλαγή στον κώδικα. Για να ξεπεραστεί αυτό, η υποστήριξη εντοπισμού σφαλμάτων είναι ενεργοποιημένη, έτσι ώστε να εντοπίζεται οποιοδήποτε σφάλμα.

```
app.debug = True
app.run()
app.run(debug = True)
```

Τα πλαίσια ιστού παρέχουν τεχνική δρομολόγησης, ώστε ο χρήστης να μπορεί να θυμάται τις διευθύνσεις URL. Η χωρίς πλοήγηση από την αρχική σελίδα γίνεται μέσω του ακόλουθου διακοσμητή διαδρομής `()`, για να συνδέσει τη διεύθυνση URL σε μια συνάρτηση.

```
@app.route('/hi')
```

```
def hi():
    return 'hi'
```

Εάν ένας χρήστης επισκεφτεί τη διεύθυνση `http://localhost:5000/hi` URL, η έξοδος της συνάρτησης `hi()` θα εμφανιστεί στο πρόγραμμα περιήγησης.

Οι μεταβλητές στο Flask χρησιμοποιούνται για τη δημιουργία δυναμικής διεύθυνσης URL προσθέτοντας τα μεταβλητά μέρη στην παράμετρο. Αυτό το μεταβλητό μέρος είναι όρισμα λέξης - κλειδιού, όπως φαίνεται στο παράδειγμα

```
from flask import Flask
app = Flask(__name__)

@app.route('/hi/<name>')
def hi(name):
    return 'Hi %s!' % name

if __name__ == '__main__':
    app.run(debug = True)
```

Αποθηκεύουμε το παραπάνω παράδειγμα ως `hi.py` και τρέχουμε το power shell και πατάμε

```
python hi.py
```

Στη συνέχεια, ανοίγουμε το πρόγραμμα περιήγησης και εισαγουμε τη διεύθυνση URL

```
http://localhost:5000/hi/liappis.
```

Στο παραπάνω παράδειγμα, η παράμετρος του διακοσμητή διαδρομής `()` περιέχει το μεταβλητό τμήμα που είναι προσαρτημένο στη διεύθυνση URL `/hi` ως όρισμα. Επομένως, εάν εισαχθεί διεύθυνση URL όπως το `http://localhost:5000/hi/liappis`, τότε το «liappis» θα περάσει στη λειτουργία `hi()` ως παράμετρο.

Εκτός από το τμήμα μεταβλητής συμβολοσειράς, χρησιμοποιούνται και άλλοι τύποι δεδομένων, όπως int, float κτλ. Οι κανόνες URL του Flask βασίζονται στην ενότητα δρομολόγησης του Werkzeug.[43] Αυτό διασφαλίζει ότι οι διευθύνσεις URL που σχηματίζονται είναι μοναδικές και βασίζονται σε προηγούμενα που καθορίζονται από το Apache.

Το Flask υποστηρίζει διάφορα πρωτόκολλα HTTP για ανάκτηση δεδομένων από το καθορισμένο URL, αυτά μπορούν να οριστούν ως:

Πίνακας 2.2 Πρωτόκολλα HTTP του Flask

Μέθοδος Περιγραφή

GET	Αυτό χρησιμοποιείται για την αποστολή των δεδομένων σε διακομιστή χωρίς κρυπτογράφηση της φόρμας.
HEAD	Παρέχει σώμα ανταπόκρισης στη φόρμα
POST	Στέλνει τα δεδομένα της φόρμας στο διακομιστή. Τα δεδομένα που λαμβάνονται με τη μέθοδο POST δεν αποθηκεύονται προσωρινά στον διακομιστή.
PUT	Αντικαθιστά την τρέχουσα αναπαράσταση του πόρου στόχου με διεύθυνση URL.
DELETE	Διαγράφει τον πόρο -στόχο ενός δεδομένου URL

Μια εφαρμογή ιστού συχνά απαιτεί ένα στατικό αρχείο όπως javascript ή ένα αρχείο CSS για να κάνει την εμφάνιση της ιστοσελίδας στο πρόγραμμα περιήγησης. Συνήθως, ο διακομιστής ιστού έχει διαμορφωθεί για να τα ορίζει, αλλά κατά την ανάπτυξη, αυτά τα αρχεία εξυπηρετούνται ως στατικοί φάκελοι στο πακέτο ή δίπλα στη μονάδα, όπως παρουσιάζεται παρακάτω:

```
from flask import Flask, render_template
app = Flask(myname)

@app.route("/")
def index():
    return render_template('index.html', title='Κόμβος', field1=field1, field2=field2)

if myname == '__main__':
    app.run(debug = True)
```

Ο ακόλουθος κώδικας HTML θα είναι μέσα στο φάκελο προτύπων.

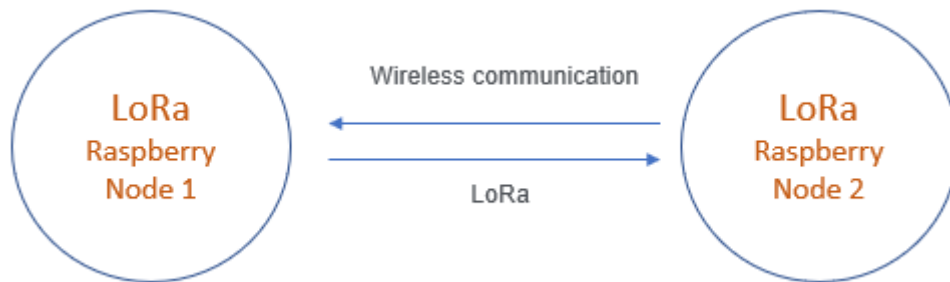
```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Liappis</title>
</head>
<body>

<div class="container">
  <br>
  <h2>{{title}} </h2>
  <table class="table">
    <thead>
      <tr>
        <th>Θερμοκρασία</th>
        <th>Υγρασία</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>{{field1}}</td>
        <td>{{field2}}</td>
      </tr>
    </tbody>
  </table>
</div>
</body>
</html>
```

Κεφάλαιο 3ο: Περιγραφή Συστήματος και Λειτουργιών

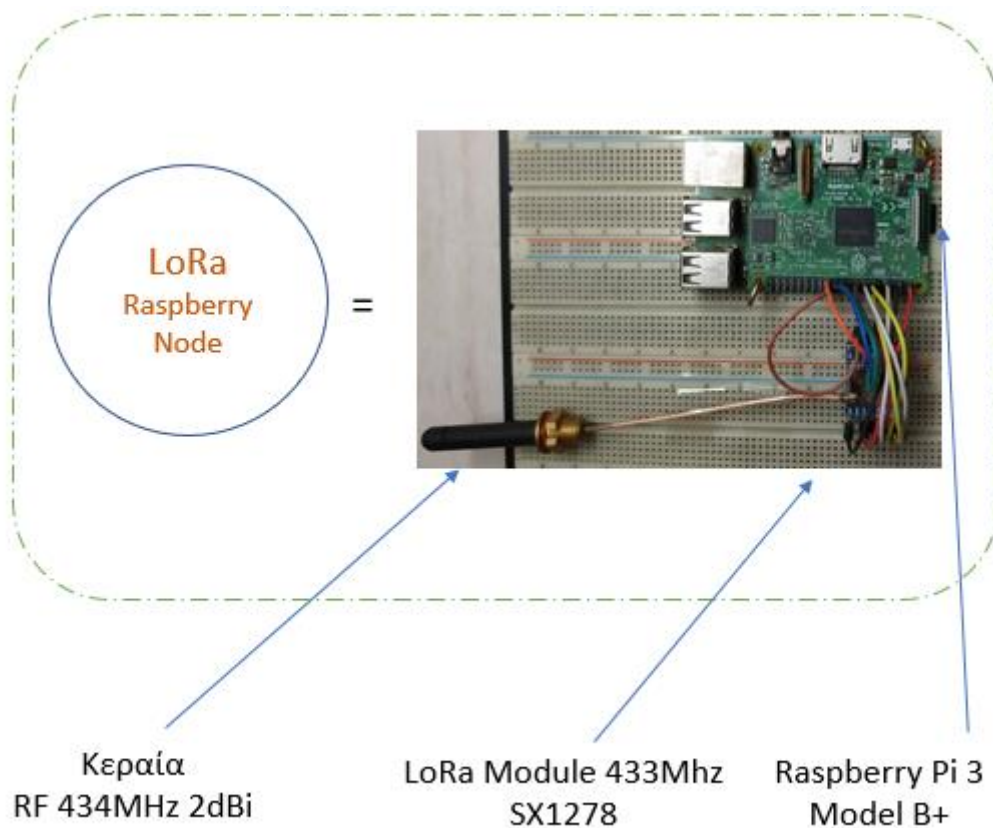
3.1 Το σύστημα

Το σύστημα που υλοποιήθηκε παρουσιάζεται στην Εικόνα 3.1 με δυο συσκευές Raspberry Pi με LoRa module το καθένα να επικοινωνούν αμφίδρομα μεταξύ τους ασύρματα.

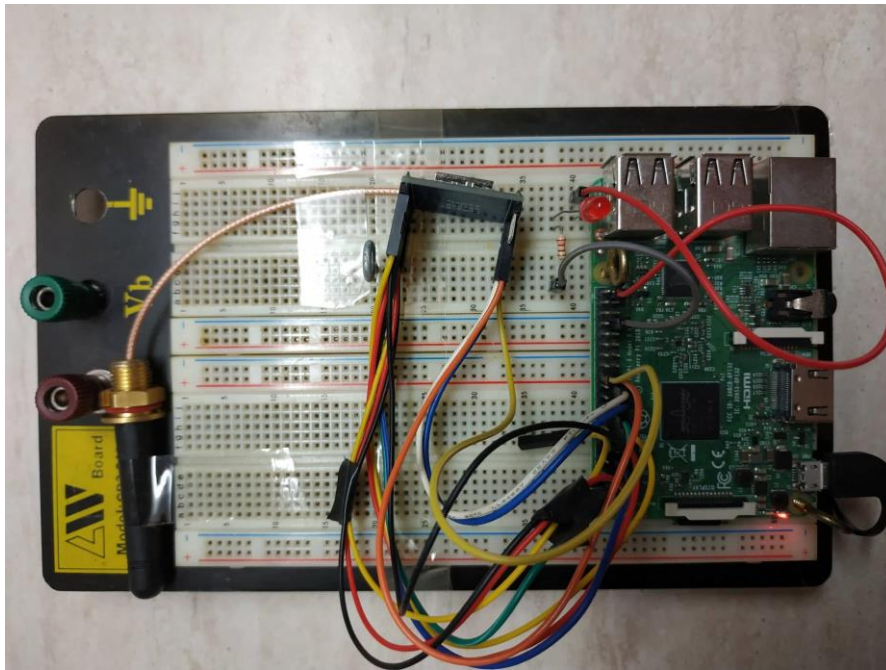


Εικόνα 3.1: Το σύστημα με τους δύο κόμβους

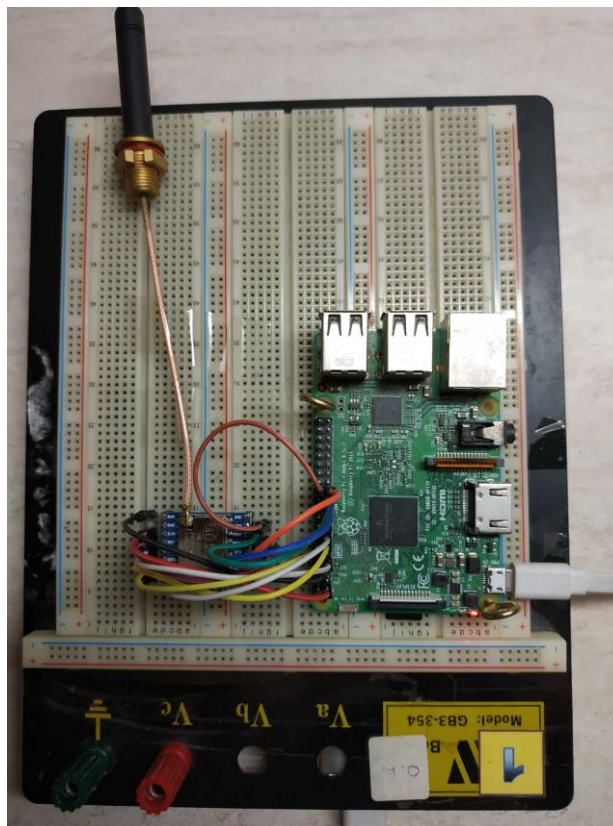
Στην Εικόνα 3.2 παρουσιάζεται σε πραγματική φωτογραφία η διάταξη-κόμβος που χρησιμοποιήθηκε με το κάθε εξάρτημα, όπως η κεραία, το LoRa module και το Raspberry.



Εικόνα 3.2: Η διάταξη-κόμβος που χρησιμοποιήθηκε



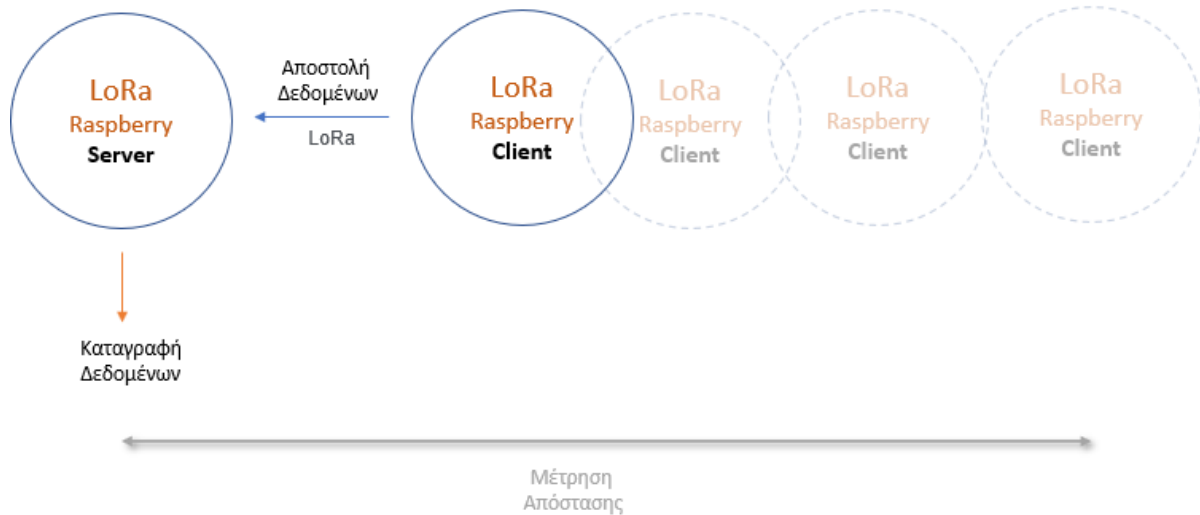
Εικόνα 3.3: Φωτογραφία διάταξης – κόμβος server



Εικόνα 3.4: Φωτογραφία διάταξης – κόμβος client

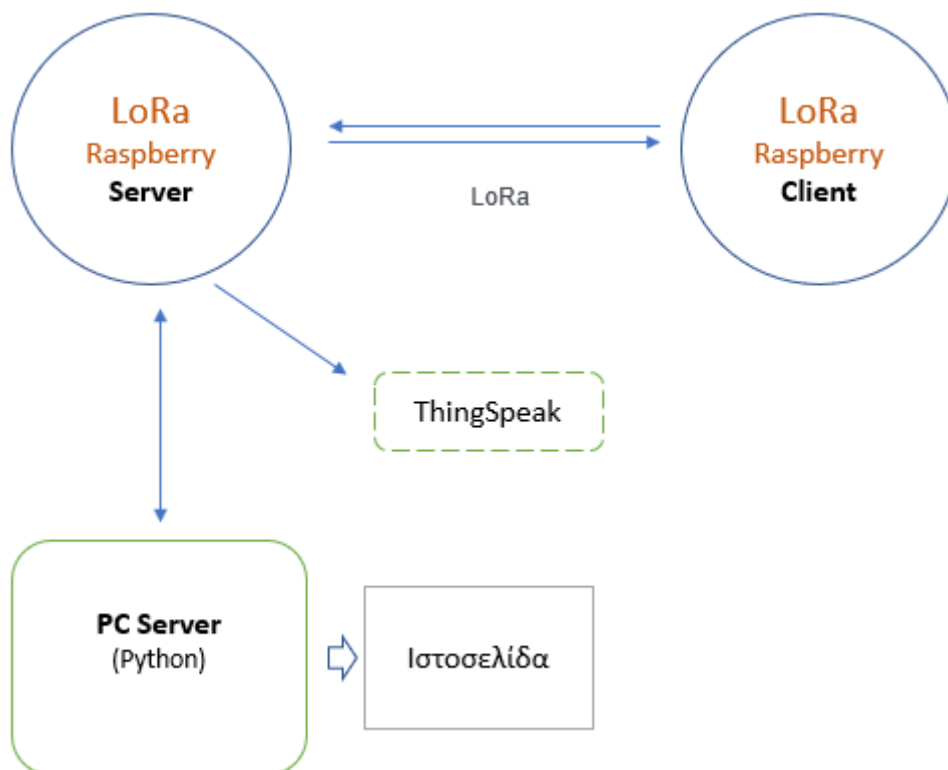
Στις εικόνες 3.3 και 3.4 παρουσιάζονται οι εικόνες από τους δύο κόμβους, client και server.

Να σημειωθεί ότι χρησιμοποιήθηκαν power banks για κάθε διάταξη για να είναι φορητό το κάθε σύστημα.



Εικόνα 3.5: Το πρώτο βασικό πείραμα με τους δυο κόμβους με τον client να απομακρύνεται

Το πρώτο βασικό πείραμα που πραγματοποιήθηκε περιγράφεται στην Εικόνα 3.5. Ο Server βρίσκεται σε σταθερή τοποθεσία.

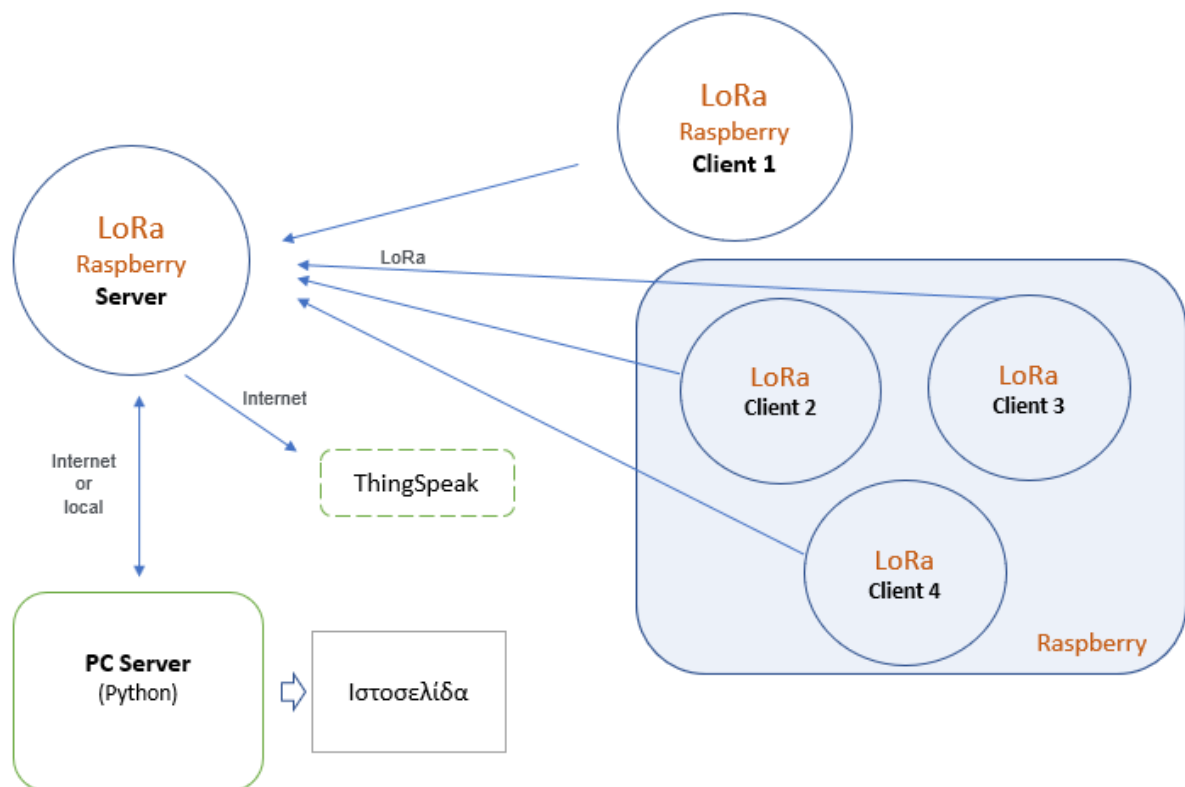


Εικόνα 3.6: Απλή τοπολογία που δείχνει την αποστολή των δεδομένων σε server

Ο client κινείται και αποστέλλει ανα σταθερό χρονικό διάστημα ένα σταθερό πακέτο από bytes. Ο server κόμβος καταγράφει την επικοινωνία ενώ ο χρήστης που έχει τον κόμβο-client καταγράφει την απόσταση.

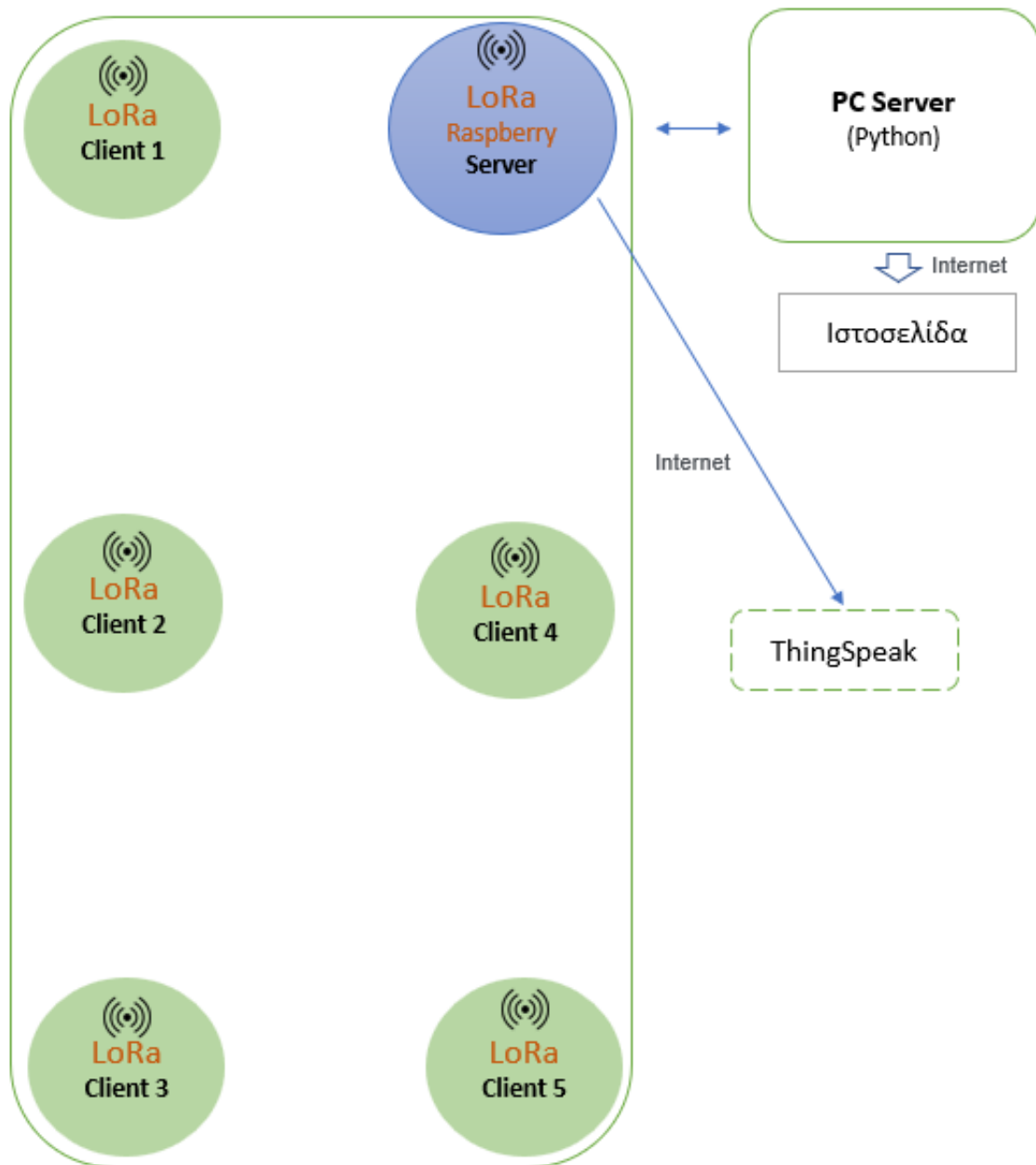
Στο δεύτερο πείραμα πραγματοποιήθηκε η παρακάτω τοπολογία της Εικόνας 3.6. Ο server -raspberry επικοινωνεί μέσω LoRa με ένα άλλο κόμβο-client. Τα δεδομένα που του στέλνει ο τελευταίος τα στέλνει σε έναν server που είναι εγκαταστημένος σε ένα άλλο υπολογιστικό σύστημα και δημιουργήθηκε με python. Αυτά τα δεδομένα τα απεικονίζει σε μια ιστοσελίδα. Όλες οι λειτουργίες περιγράφονται αναλυτικά σε επόμενες παραγράφους. Η λειτουργία αυτή μπορεί να γίνει σε τοπικό server ή σε απομακρυσμένο αν υπάρχει δυνατότητα σύνδεσης με το διαδίκτυο.

Παράλληλα, ο server-raspberry κόμβος μπορεί να στέλνει τα δεδομένα στο ThingSpeak μέσω διαδικτύου.



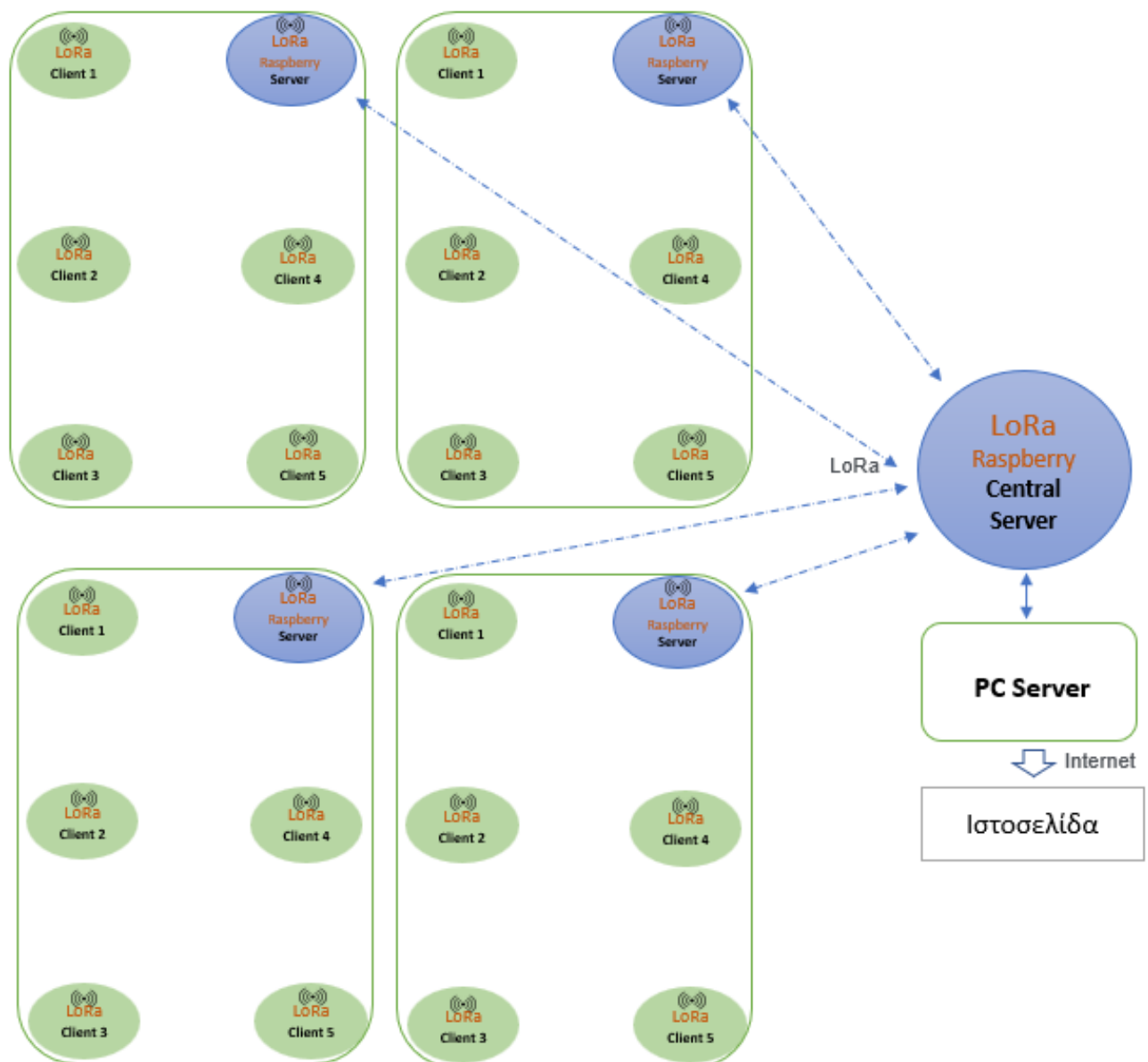
Εικόνα 3.7: Επιπρόσθετοι κόμβοι-client που αποστέλλουν δεδομένα στον κεντρικό κόμβο-server

Στην Εικόνα 3.7 παρουσιάζεται μια εκδοχή με επιπρόσθετους κόμβους-client που αποστέλλουν δεδομένα στον κεντρικό κόμβο-server. Οι κόμβοι αυτοί είναι εικονικοί, δηλαδή είναι προγράμματα-python ίδια με το κεντρικό client 1, βρίσκονται στο ίδιο Raspberry και χρησιμοποιούν το ίδιο LoRa module.



Εικόνα 3.8: Τοπολογία με πολλούς clients που μπορούν να τοποθετηθούν σε έναν χώρο, όπως ένα Θερμοκήπιο

Στην Εικόνα 3.8 παρουσιάζεται μια τοπολογία με πολλούς clients που μπορούν να τοποθετηθούν σε έναν χώρο, όπως ένα Θερμοκήπιο και να επικοινωνούν με έναν κεντρικό κόμβο Raspberry που έχει τον ρόλο του Server και εκτελεί παρόμοιες λειτουργίες που περιγράφηκαν παραπάνω.



Εικόνα 3.9: Τοπολογία με τέσσερα Θερμοκήπια που έχουν πολλούς clients

Στην Εικόνα 3.9 παρουσιάζεται μια τοπολογία με τέσσερα Θερμοκήπια που έχουν πολλούς clients οι οποίοι επικοινωνούν με το δικό τους κεντρικό κόμβο Raspberry που έχει τον ρόλο του τοπικού Server. Όλοι αυτοί οι Server επικοινωνούν με τον κεντρικό Server ο οποίος είναι Raspberry και εκτελεί λειτουργίες που περιγράφηκαν παραπάνω.

Κάθε κόμβος, client και server, έχουν μοναδικό id το οποίο ονομάζεται guid και η αποστολή κάθε πακέτου από κόμβο σε κόμβο περιέχει πληροφορίες για τον αποστολέα και το δέκτη.

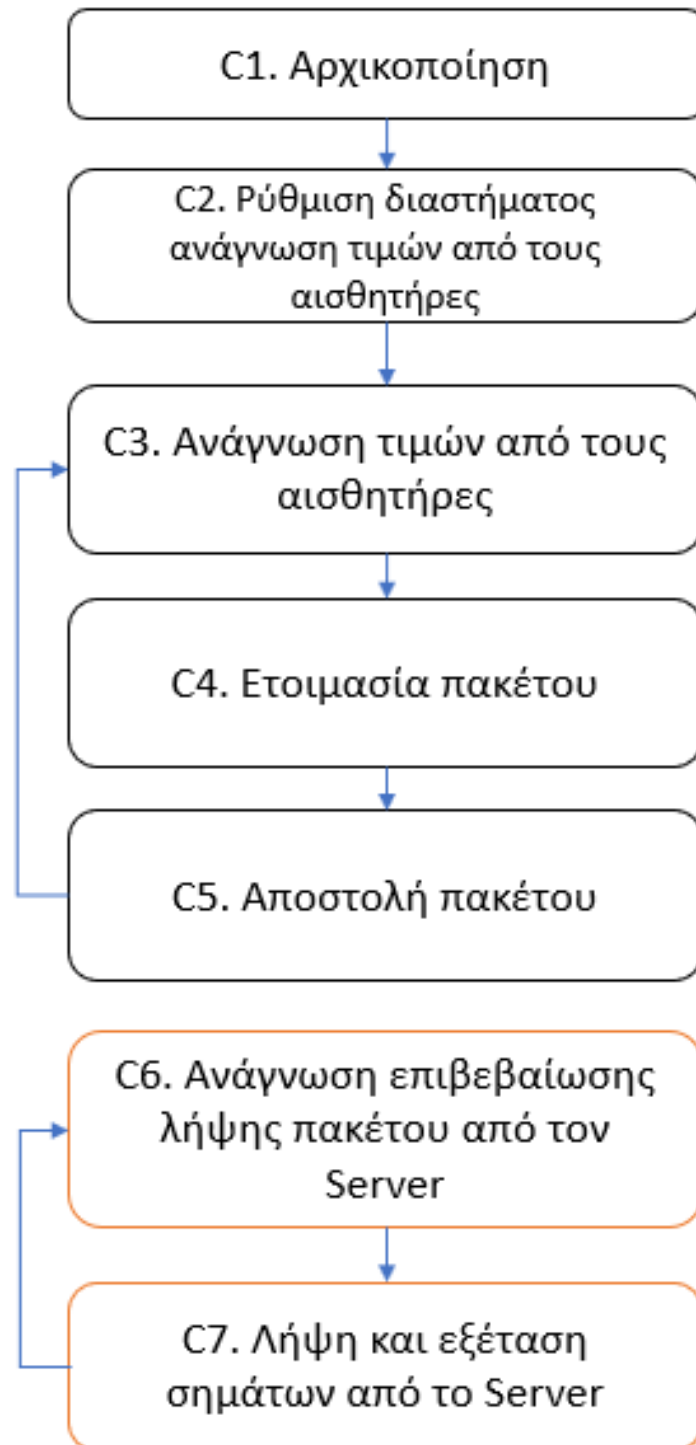
Να τονισθεί σε αυτό το σημείο ότι το LoRa αποστέλλει το σήμα-πακέτο και όλοι οι κόμβοι το λαμβάνουν. Με έναν ειδικό τρόπο που παρουσιάζεται σε αυτή την εργασία γίνεται ταυτοποίηση και κωδικοποίηση της επικοινωνίας, η οποία περιγράφεται σε επόμενες παραγράφους.



Εικόνα 3.10: Τοπολογία με πολλά θερμοκήπια

Η τοπολογία της Εικόνας 3.9 μπορεί να βρει εφαρμογή σε εγκαταστάσεις με πολλά θερμοκήπια στα οποία έχουν τοποθετηθεί αισθητήρες-κόμβοι-clients σε διάφορα σημεία στο εσωτερικό του, όπως φαίνεται στην Εικόνα 3.10. Κάθε θερμοκήπιο μπορεί να περιέχει έναν τοπικό server ο οποίος να επικοινωνεί με ένα κεντρικό ή μπορεί να αποστέλλουν τα δεδομένα απευθείας στον κεντρικό (κίτρινο κτίριο), όταν μπορεί να είναι εφικτή η επικοινωνία λόγω απόστασης,

Το LoRa φαίνεται ότι καλύπτει μεγάλες αποστάσεις που μπορούν να καλύψουν μεγάλα θερμοκήπια κάτω από 500 μέτρα.



Εικόνα 3.11: Κεντρικό διάγραμμα λειτουργίας στον Client

Στην Εικόνα 3.11 παρουσιάζεται το κεντρικό διάγραμμα λειτουργίας στον Client.

Το πρόγραμμα είναι γραμμένο σε γλώσσα Python και ξεκινάει με την αρχικοποίηση με υποστήριξη βιβλιοθηκών και μεταβλητών. Στη συνέχεια προχωράει σε ρύθμιση διαστήματος ανάγνωση τιμών από τους αισθητήρες και σε ανάγνωση τιμών από τους αισθητήρες σε σταθερό χρονικό διάστημα.

Στη συνέχεια ετοιμάζεται το πακέτο για αποστολή και αποστέλλεται αφού κωδικοποιηθεί. Μια σημαντική λειτουργία του προγράμματος είναι η επιβεβαίωση της λήψης πακέτου από τον κόμβο που το έλαβε, συνήθως ο κόμβος-server.

Επίσης, υπάρχει η δυνατότητα λήψης και εξέτασης επιπλέον σημάτων που μπορεί να στείλει ο server και αυτά συνήθως αποτελούν ειδικές εντολές για ενεργοποίηση συγκεκριμένων λειτουργιών, όπως ενεργοποίηση ενός relay-ηλεκτρονόμου(για να ενεργοποίηση μια συσκευής) στον client.

Όλες οι παραπάνω λειτουργίες θα παρουσιαστούν με κομμάτια κώδικα για να βοηθήσουν τον αναγνώστη στην καλύτερη κατανόηση τους.

C1.

```
myguid = 'A12' #my id
toguid = 'S15' #Server id
```

Κάθε κόμβος έχει το δικό του id με τρία γράμματα (3 bytes) το καθένα.

Οι απλοί κόμβοι-clients ξεκινούν με το 'A' ενώ οι servers με το 'S'.

C2.

```
from SX127x.LoRa import *
from SX127x.LoRaArgumentParser import LoRaArgumentParser
...
parser.add_argument('--wait', '-w', dest='wait', default=1, action="store", type=float, help="Waiting
time between transmissions (default is 0s)")
def on_tx_done(self):
    global args
    ...
    sleep(args.wait)
```

Η αποστολή γίνεται παράλληλα με την ανάγνωση των αισθητήρων κάθε t s.

Αυτό ρυθμίζεται μέσω της πρώτης εντολής parser.add_argument του C2 και η διεργασία πραγματοποιείται στην ειδική συνάρτηση on_tx_done που έχει η βιβλιοθήκη SX127x.LoRa για την αποστολή δεδομένων.

C3.

```
import RPi.GPIO as GPIO
import Adafruit_DHT
```

```

...
DHT_SENSOR = Adafruit_DHT.DHT11
DHT_PIN = 4
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BOARD)

    humidity, temperature = Adafruit_DHT.read(DHT_SENSOR, DHT_PIN)
    sensor1 = humidity
    sensor2 = temperature

```

Στο C3 παρουσιάζεται το κομμάτι που πραγματοποιείται η ανάγνωση των αισθητήρων με τη βοήθεια ειδικών βιβλιοθηκών, όπως η Adafruit_DHT.

Ρυθμίζονται οι ακροδέκτες όπως επιθυμεί ο χρήστης.

Στο πείραμα που πραγματοποιήσαμε απενεργοποιήθηκαν οι αισθητήρες γιατί ο σκοπός του πειράματος ήταν η μέτρηση χαρακτηριστικών της επικοινωνίας μεταξύ των κόμβων και η διαχείριση του πακέτου μετάδοσης.

C4.

```

# Η βιβλιοθήκη Fernet υποστηρίζει την κρυπτογράφηση-αποκρυπτογράφηση.
from cryptography.fernet import Fernet

# Για παραγωγή τυχαίων αριθμών
import random
import string
...
# Δημιουργούμε ένα κλειδί που θα το έχει ο κάθε client και server του δικτύου μας
key = b'vyY29vFL_KTVWC3FpCkgdOo5G2cFiqn4_svU79oBoI8='
f = Fernet(key)
...
# Διαβάζουμε τις τιμές των αισθητήρων
humidity, temperature = ...

# Δημιουργούμε ένα μοναδικό id για το πακέτο που θα μεταδοθεί πχ 57C
signal = str(random.randint(10,99))+random.choice(string.ascii_uppercase)

# Αποθηκεύουμε το id σε global μεταβλητή
storedsignal=signal
...
sensor1 = humidity

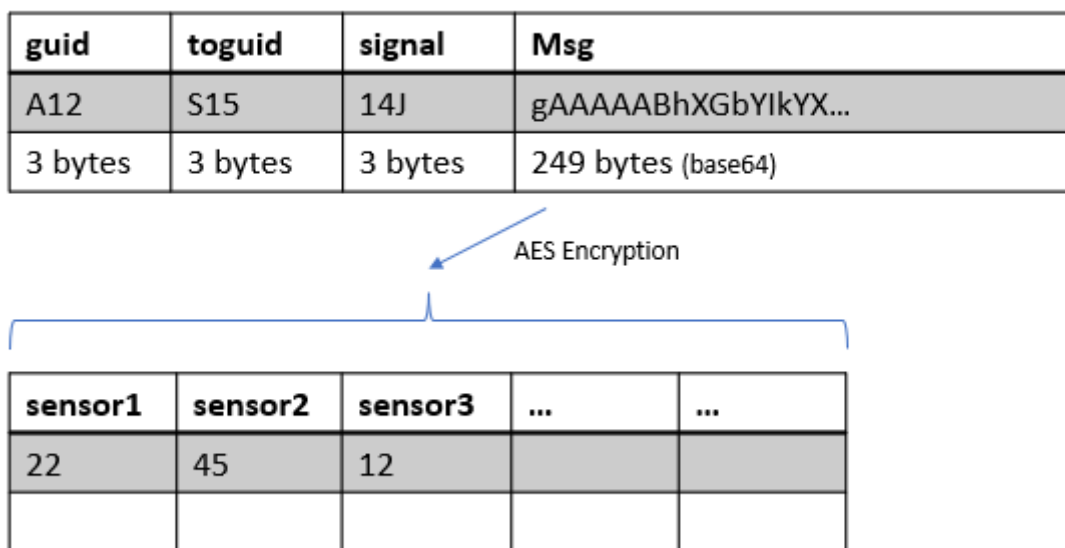
```

```

sensor2 = temperature
#Δημιουργούμε το αρχικό πακέτο με τις τιμές
sensordata = "sensor1="+str(sensor1)+"#sensor2="+str(sensor2)
#Κρυπτογραφούμε το αρχικό πακέτο με τις τιμές
ciphertext = f.encrypt(msg.encode())
#Δημιουργούμε το τελικό πακέτο
paketo = myguid+"#" + toguid+"#" + signal

```

Στο C4 μέσα στο κώδικα περιγράφεται το κάθε σημαντικό βήμα.



Εικόνα 3.12: Μορφοποίηση του βασικού πακέτου μετάδοσης

Κατά την κρυπτογράφηση δημιουργείται το Msg της Εικόνας 3.12 το οποίο μπορεί να είναι μέχρι 249 bytes, αφού το πακέτο μετάδοσης μέσω αυτής της LoRa επικοινωνίας είναι 255 bytes.

Παράδειγμα για μορφή

```
sensor1=22.3#sensor2=44.6#sensor3=-76.8
```

προκύπτει μέγεθος 140 bytes:

```

b'gAAAAABhXGbYIkYXTuXKheYacmcn8vQWE3CLXWpm5sOBDZc6grr1P1SbMUEViR2ZKdt
3mdN6-yjpImH6UmzzZ2O5G_Cd-
oHrHBTnqTACij_gW9eN6yjbWmXPEGu8QdGNpdcu2uO2fX6V'

```

Το format πρέπει να είναι ακριβώς όπως ορίζεται στην Εικόνα 4.12 για να αναγνωρίζεται από τους άλλους κόμβους και μπορούν να τοποθετηθούν sensor1 ... sensorN μέχρι το κρυπτογραφημένο πακέτο να φτάσει σε μέγεθος 249 bytes (δεν ελέγχεται στο παρόν).

C5.

```
paketo_as_bytes = str.encode(paketo)
bpaketo = bytes(paketo, 'utf-8')

lista = list(bpaketo)
self.write_payload(lista)

self.set_mode(MODE.TX)
```

Στο κομμάτι κώδικα C5 γίνεται αποστολή του πακέτου με ένα σύνολο εντολών που μας παρέχει το παράδειγμα της βιβλιοθήκης SX127x.LoRa.

Να τονισθεί η λειτουργία σε MODE.TX για την αποστολή.

C6.

```
#Ρυθμίσεις για λήψη σύμφωνα με τη βιβλιοθήκη SX127x.LoRa
self.clear_irq_flags(RxDone=1)
payload = self.read_payload(nocheck=True)
message = bytes(payload).decode("utf-8", 'ignore')

#form of msg from server: S15#A12#V#14J
verifiedsignal = 0
# Το πακέτο που λαμβάνει ο Client περιέχει είτε σήμα επιβεβαίωσης είτε σήμα εντολής
msg=message.split("#")
    if msg[1] == myguid and msg[2] == toguid:
        if msg[3] == 'V':
            if msg[4] == storedsignal:
                verifiedsignal = 1
        if msg[5] == 'C':
            command = msg[3]
```

Στο κομμάτι κώδικα C5 παρουσιάζεται η Ανάγνωση επιβεβαίωσης λήψης πακέτου από τον Server.

Κάθε πακέτο που λαμβάνει ο Server-κόμβο από τον Client-κόμβο το ελέγχει ότι απευθύνεται σε αυτόν, το αποκωδικοποιεί με το ίδιο key και αφού το format του πακέτου είναι σωστό τότε αποστέλλει στον Client σήμα επιβεβαίωσης λήψης πακέτου.

Το πακέτο που λαμβάνει ο Client περιέχει είτε σήμα επιβεβαίωσης είτε σήμα εντολής

guid	togroupid	mode	signal
S15	A12	V	14J
3 bytes	3 bytes	1 byte	3 bytes

Εικόνα 3.13: Μορφοποίηση του πακέτου επιβεβαίωσης

Το πακέτο που λαμβάνει ο Client περιέχει ως σήμα επιβεβαίωσης παρουσιάζεται στην Εικόνα 3.13.

Τα πρώτα 3 bytes (guid) είναι το id του αποστολέα, που στέλνει το σήμα – συνήθως ο server.

Τα επόμενα 3 bytes (togroupid) είναι το id του δέκτη, συνήθως ο client.

Το επόμενο 1 byte (mode) είναι το mode της εντολής.

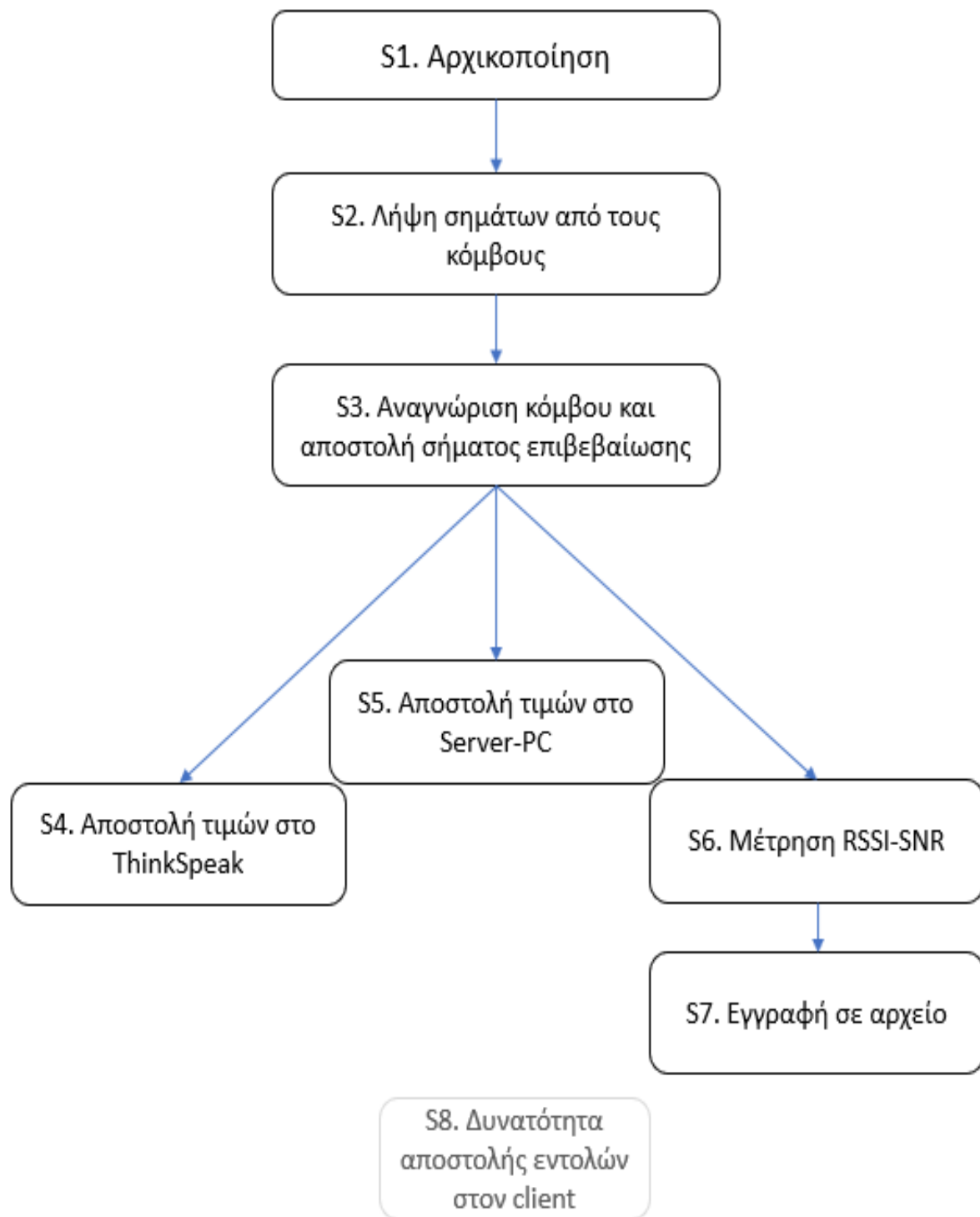
Αν είναι C αφορά σήμα εντολής.

Αν είναι V αφορά σήμα επιβεβαίωσης.

Τα επόμενα 3 bytes (signal) αφορούν το σήμα

Όταν mode=C είναι η εντολή για κάποια ενέργεια

Όταν mode=V είναι ο κωδικός που είχε αποστείλει ο client στην τελευταία αποστολή του και συμπεριλάμβανε αυτόν τον κωδικό.



Εικόνα 3.14: Βασικές λειτουργίες που εκτελούνται στον Server-κόμβο-Raspberry

Στο διάγραμμα της Εικόνας 3.14 παρουσιάζονται οι βασικές λειτουργίες που εκτελούνται στον Server-κόμβο-Raspberry

S2.

```
node_list = ['A12','B10','A04']
...
msg=message.split("#")
    if msg[1] == serverguid:
        if msg[0] in node_list:
            signal = msg[2]
            sensorsdata = msg[3]
            try:
                sensorsdata = f.decrypt(sensorsdata)
                sensorsdata = sensorsdata.decode()
                sensors = sensorsdata.split("#")
            except:
                print("Invalid")
```

Στο κομμάτι κώδικα S2 παρουσιάζεται η λήψη σημάτων από τους κόμβους.

Το σήμα έχει τη μορφή της Εικόνας 3.12.

Το δεύτερο 3 bytes πρέπει να περιέχει το id του server αλλιώς δεν το δέχεται.

Το πρώτο 3 bytes πρέπει να περιέχει το id του κόμβου (client), από τον οποίο δέχεται το σήμα-πακέτο και πρέπει να βρίσκεται μέσα στη λίστα node_list.

Αφού ικανοποιούνται τα παραπάνω προχωράει σε αποκωδικοποίηση των σημάτων.

S3.

```
paketo_for_verification = myguid+"#"+toguid+"#"+mode+"#"+signal

paketo_as_bytes = str.encode(paketo_for_verification)
bpaketo = bytes(paketo, 'utf-8')

lista = list(bpaketo)
self.write_payload(lista)
```

```
self.set_mode(MODE.TX)
```

Στο κομμάτι κώδικα S3 μετά την αναγνώριση κόμβου πραγματοποιείται η αποστολή του σήματος επιβεβαίωσης για το πακέτο που έλαβε από τον client.

S4.

Στο κομμάτι κώδικα S4 παρουσιάζεται η

Αποστολή στο ThingSpeak απευθείας από το Raspberry Server που έχει πρόσβαση στο διαδίκτυο

```
import requests
...
#https://api.thingspeak.com/update?api_key=T5A2HCB80Z4FHV8L&field1=25&field2=52
requests.get("https://api.thingspeak.com/update?api_key= T5A2HCB80Z4FHV8L &field1="+ sensors
[0]+ "&field2="+ sensors [1])
```

S5.

Στο κομμάτι κώδικα S5 παρουσιάζεται η

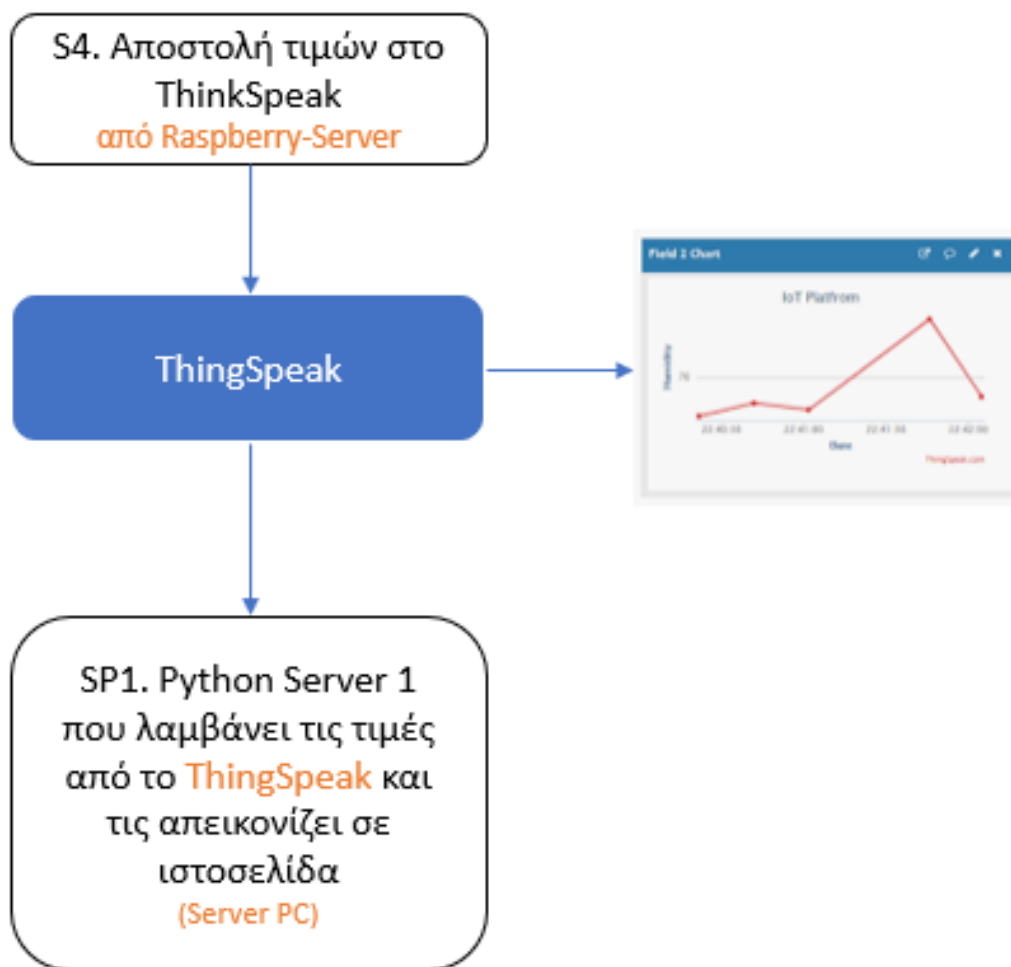
Αποστολή στο Server-PC απευθείας από το Raspberry Server που έχει πρόσβαση στο διαδίκτυο

```
import requests
...
requests.get("http://localhost:105/addvalue/?guid=A12&field1="+ sensors [0]+ "&field1=" sensors
[1]+ "&token=4324563")
"
```

S6-S7.

```
    rssi_value = self.get_pkt_rssi_value()
    sys.stdout.write("rssi= %d snr= %d\n" % (rssi_value, self.get_pkt_snr_value()))
    now = datetime.datetime.now()
    file1 = open('mydata.txt', 'a+')
    file1.write(now.strftime("%Y-%m-%d %H:%M:%S"))
    file1.write(" "+str(rssi_value)+" "+str(self.get_pkt_snr_value())+" \n")
    file1.close()
```

Στο κομμάτι κώδικα S6-7 παρουσιάζεται η μέτρηση μέσω της βιβλιοθήκης SX127x.LoRa και το module SX1278 των τιμών rssi και snr και πραγματοποιείται εγγραφή σε αρχείο με ειδικό format.



Εικόνα 3.15: Περιγραφή επικοινωνίας κόμβου με τους servers ThingSpeak

Όπως φαίνεται στην Εικόνα 3.15 κάθε κόμβος-Server-Raspberry μπορεί να στέλνει τα δεδομένα του απευθείας στο ThingSpeak. Μέσω αυτού μπορεί να βλέπει σε γραφικό περιβάλλον διαγράμματα και να επεξεργαστεί τα δεδομένα.

Έχει δημιουργηθεί ειδικό πρόγραμμα σε υπολογιστή γραμμένο σε Python για να μπορεί να διαβάζει τις τιμές από το ThingSpeak.

Έχει χαρακτηριστικά server και υποστηρίζει δημιουργία και προβολή ιστοσελίδας και αυτό φαίνεται στο παρακάτω κομμάτι κώδικα SP1.

SP1.

```
import requests
from flask import render_template
from flask import Flask
app = Flask(__name__)

@app.route('/index')
def index():

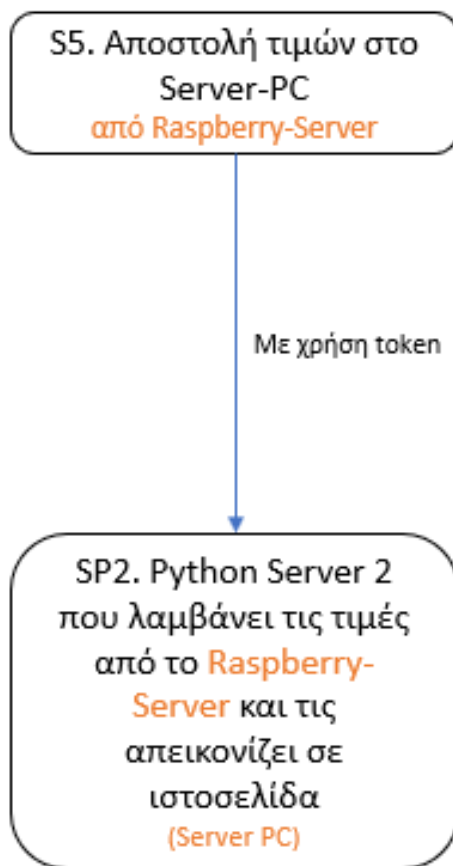
#https://api.thingspeak.com/update?api_key=T5A2HCB80Z4FHV8L&field1=25&field2=52

r=requests.get("https://api.thingspeak.com/channels/1527755/feeds.json?api_key=U6EH75GL3TC99
67I&results=1")
    print(r.json())
    rj = r.json()
    print("")
    print(rj['feeds'][0]['field1'])
    print(rj['feeds'][0]['field2'])
    field1 = rj['feeds'][0]['field1']
    field2 = rj['feeds'][0]['field2']
    return render_template('index.html', title='Κόμβος', field1=field1, field2=field2)

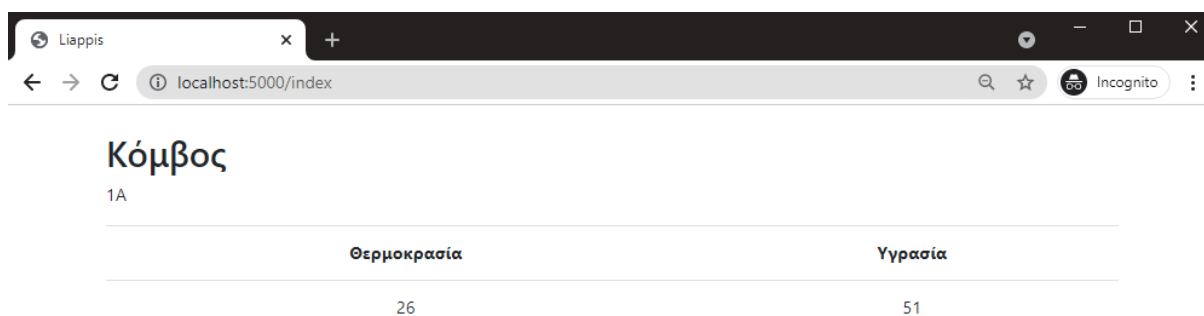
if __name__ == '__main__':
    app.run()
```

Μια άλλη επιπρόσθετη και σημαντική λειτουργία είναι η απευθείας αποστολή των δεδομένων-πακέτου από το κόμβο-server στον υπολογιστή server χωρίς την χρήση ThingSpeak, όπως παρουσιάζεται στην Εικόνα 3.16.

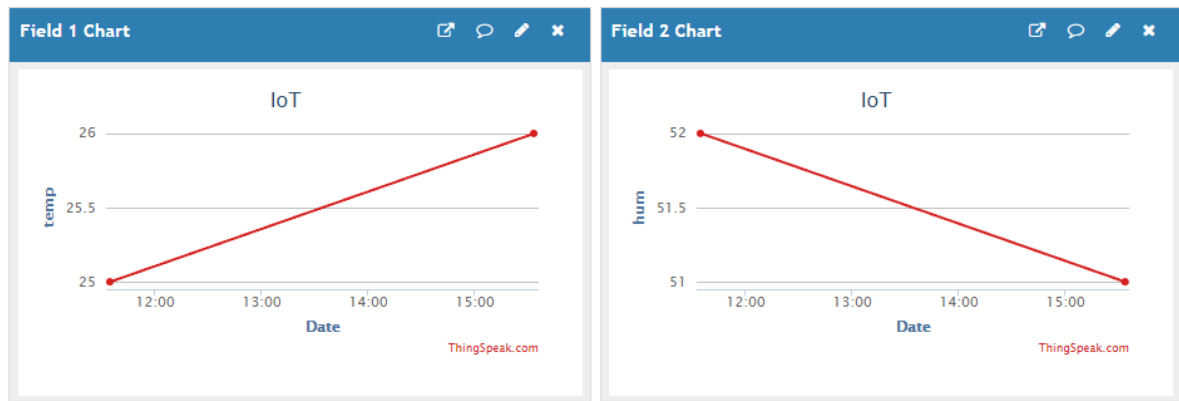
Τα δεδομένα τα αναπαριστά σε ιστοσελίδα, Εικόνα 3.17 με τη βοήθεια του framework Flask και στο ThingSpeak όπως φαίνεται στην Εικόνα 3.18.



Εικόνα 3.16: Απευθείας αποστολή των δεδομένων-πακέτου από το κόμβο-server στον υπολογιστή server χωρίς την χρήση ThingSpeak



Εικόνα 3.17: Αναπαράσταση δεδομένων σε ιστοσελίδα με framework Flask



Εικόνα 3.18: Αναπαράσταση δεδομένων σε ThingSpeak

PS2.

```

from flask import Flask
from flask import request,jsonify
import json

app = Flask(__name__)

@app.route('/', methods=['GET', 'POST'])
def main():
    # http://localhost:105/
    return "Main"

@app.route('/addvalue/', methods=['GET'])
def addvalue():
    # http://localhost:105/addvalue/?guid=A12&field1=28&field1=56&token=4324563
    token = request.args.get("token");
    if token == '4324563':
        guid = request.args.get("guid");
        field1 = request.args.get("field1");
        field2 = request.args.get("field2");
        print(field1)
        return guid

```

```
else:
    return "Not Authorized"

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=105)
```

Στον κώδικα SP2 παρουσιάζεται η δημιουργία Python server που δέχεται τα δεδομένα από τον κόμβο-server.

Για να λειτουργήσει ο python server πρέπει να ενεργοποιηθούν τα endpoints και εμείς ορίζουμε το να δέχεται GET requests στο όνομα addvalue.

```
@app.route('/addvalue/', methods=['GET'])
```

Ο κόμβος-server-raspberry μπορεί να αποστείλει αν καλέσει ένα ειδικό URL όπως το

<http://localhost:105/addvalue/?guid=A12&field1=28&field1=56&token=4324563>

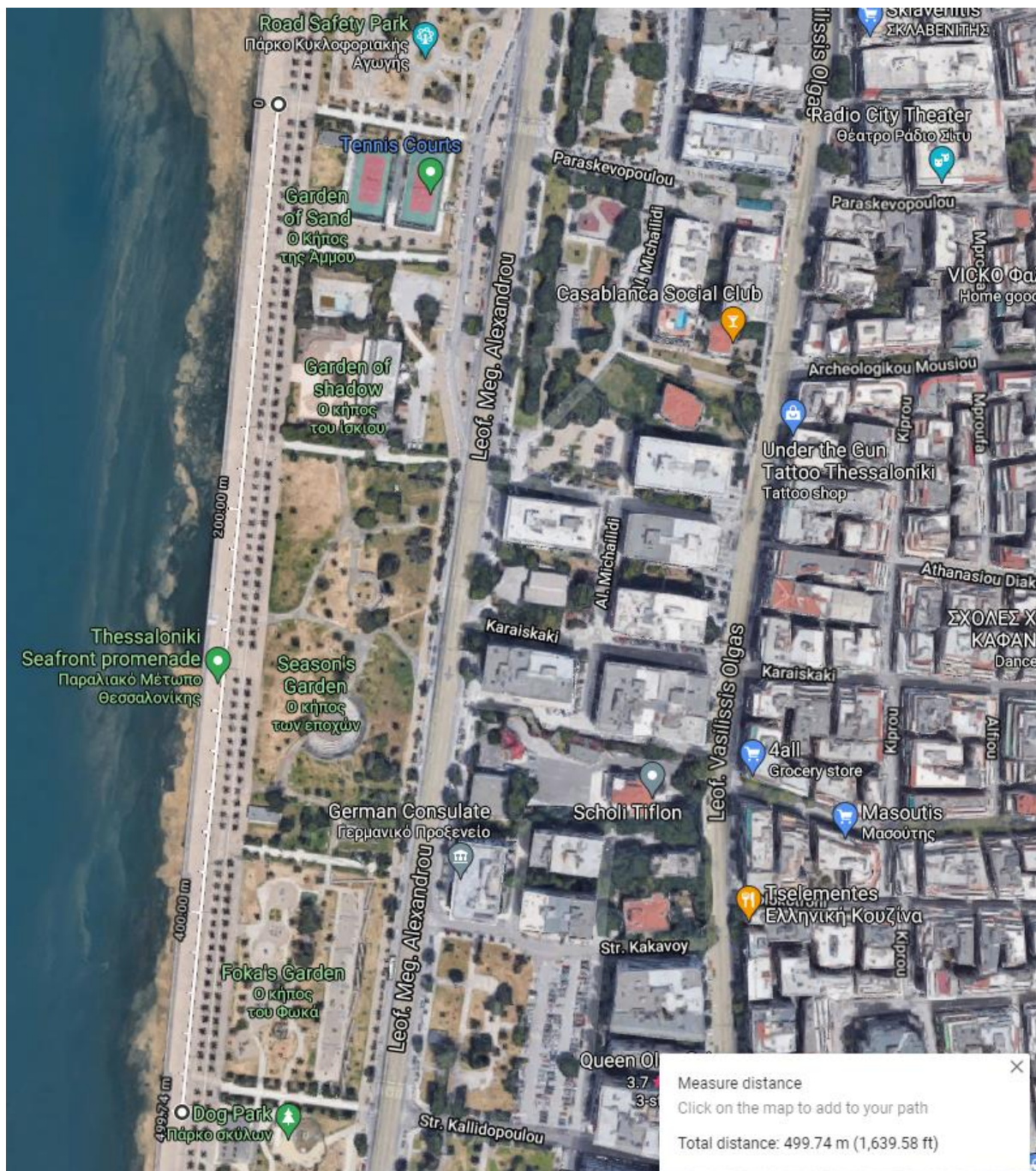
Θα γίνει έλεγχος του κόμβου μέσω ενός token που το περιέχουν μόνο

κόμβος-server-raspberry και python-server-pc.

Κεφάλαιο 4ο: Πειραματικό Μέρος

4.1 Μετρήσεις σε ανοιχτό πεδίο χωρίς εμπόδια

Οι μετρήσεις σε ανοιχτό πεδίο πραγματοποιήθηκαν κυρίως στη νέα παραλία Θεσσαλονίκης όπως φαίνεται στην Εικόνα 4.1.

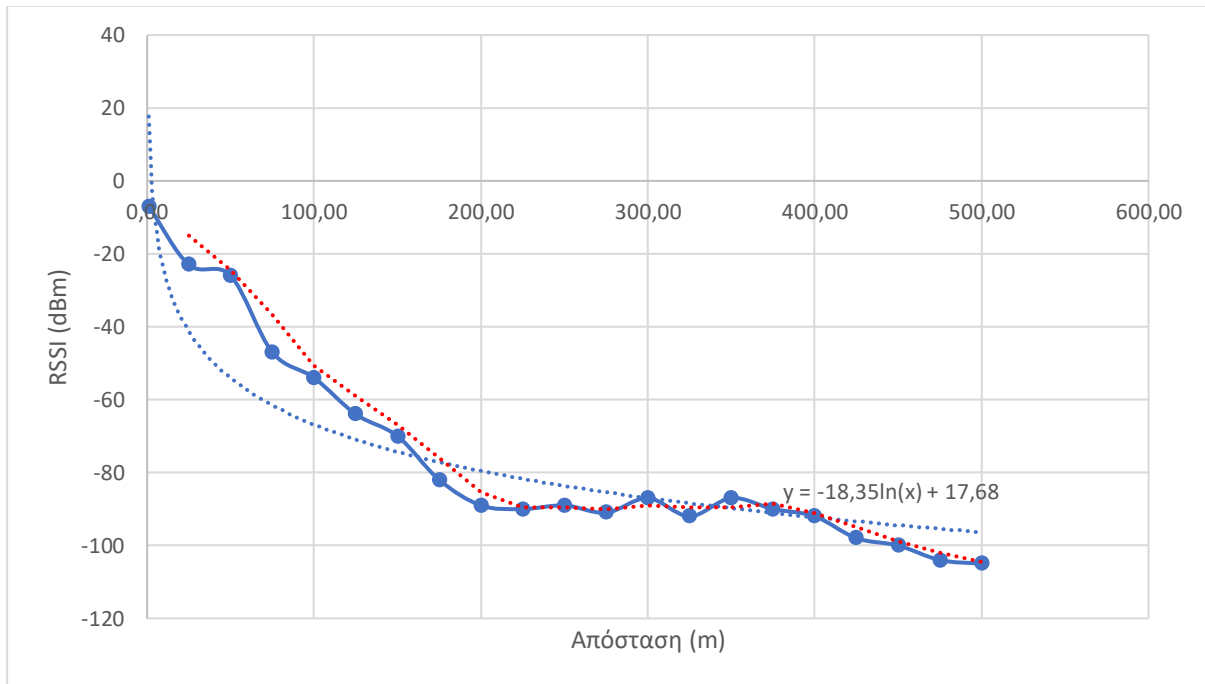


Εικόνα 4.1: Μετρήσεις σε ανοιχτό πεδίο στη νέα παραλία Θεσσαλονίκης

Πραγματοποιήθηκαν και άλλες μετρήσεις σε ανοιχτούς χώρους αλλά θεωρήσαμε ότι η συγκεκριμένη περιοχή είναι ένα καλό σημείο αναφοράς, χωρίς αυτοκίνητα, με ομοιόμορφο δάπεδο χωρίς κλίση.

Το ύψος της κάθε συσκευής είναι στο 1.5 μέτρο από το δάπεδο.

Ο server-device έμεινε ακίνητος στο σημείο 0 και ο client-device απομακρύνεται από αυτόν σε ευθεία γραμμή.



Εικόνα 4.2: Σχέση ανάμεσα στην απόσταση των δύο συσκευών και το RSSI σε ανοιχτό πεδίο

Στην Εικόνα 4.2 παρουσιάζεται η σχέση ανάμεσα στην απόσταση των δύο συσκευών και το RSSI.

Κάθε τελεία αντιπροσωπεύει το μέσο όρο 5 μετρήσεων για κάθε απόσταση με διαφορά 25 μέτρα και η τελευταία 10 μετρήσεις.

Η διακεκομμένη μπλε γραμμή παριστάνει την λογαριθμική γραμμή τάση και αναγράφεται και η εξίσωση που προκύπτει.

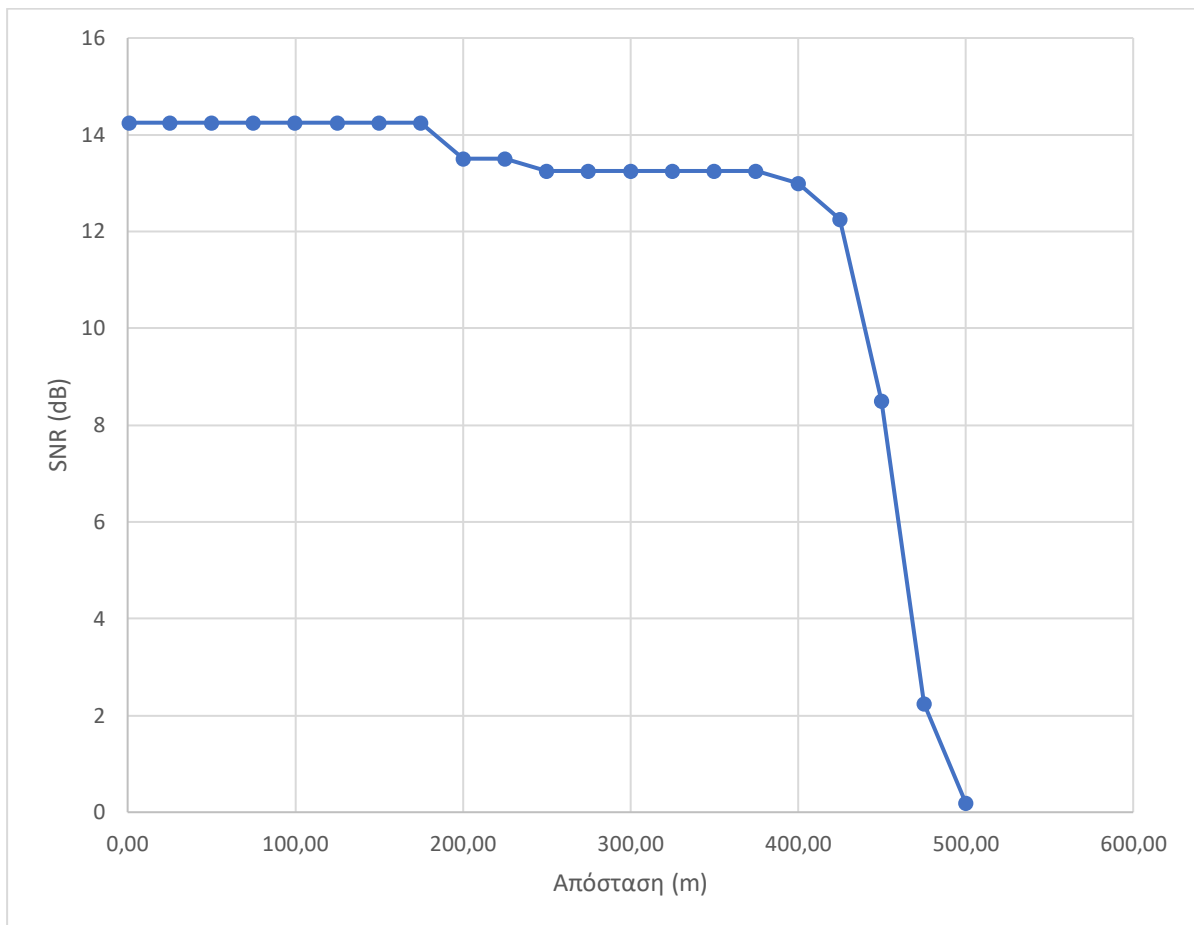
Η διακεκομμένη κόκκινη γραμμή προκύπτει από τον κυλιόμενο μέσο.

Προκύπτει ότι σε ανοιχτό χώρο η απόσταση (μέση τιμή) μεταξύ δύο κόμβων (server και client) είναι:

499.47 m

με τυπική απόκλιση

2.03 m



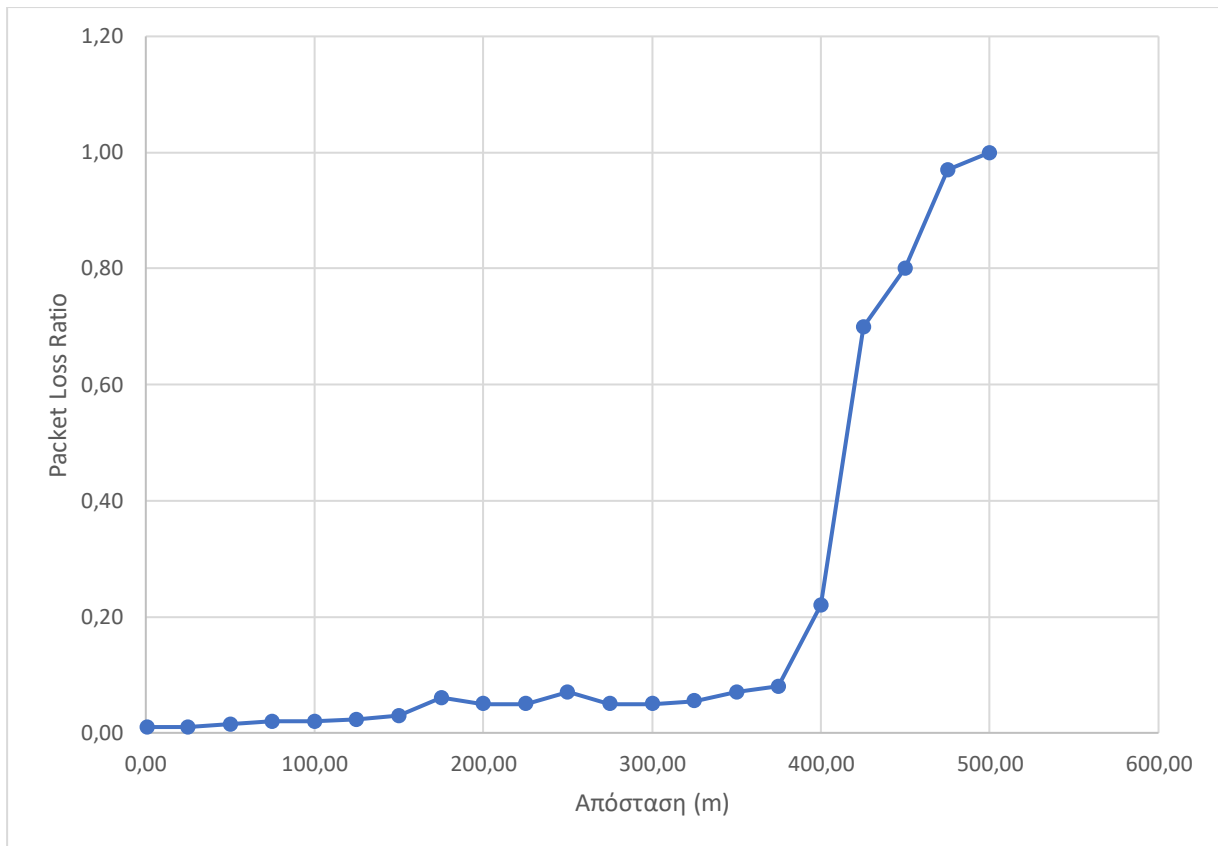
Εικόνα 4.3: Σχέση ανάμεσα στην απόσταση των δύο συσκευών και το SNR σε ανοιχτό πεδίο

Φαίνεται ότι μετά στα 200 μέτρα το RSSI βρίσκεται σε χαμηλά επίπεδα και παραμένει σε αυτά μέχρι τα 500μ όπου το σήμα-σύνδεση χάνεται και δεν λαμβάνεται σωστά το πακέτο.

Στην Εικόνα 4.3 παρουσιάζεται η σχέση ανάμεσα στην απόσταση των δύο συσκευών και το SNR.

Κάθε τελεία αντιπροσωπεύει το μέσο όρο 5 μετρήσεων για κάθε απόσταση με διαφορά 25 μέτρα.

Παρατηρούμε ότι μετά τα 200 μέτρα το SNR μειώνεται με μεγαλύτερη μείωση μετά τα 400μ και περίπου στα 500μ παρατηρούμε ότι ο λόγος σήματος προς θόρυβο γίνεται 1 και το πακέτο που λαμβάνεται παρουσιάζει την μέγιστη απώλεια.



Εικόνα 4.4: Η σχέση ανάμεσα στον κανονικοποιημένο λόγο απώλειας πακέτων (Packet Loss Ratio) και στην απόσταση μεταξύ των δύο κόμβων σε ανοιχτό πεδίο

Στην Εικόνα 4.4 παρουσιάζεται η σχέση ανάμεσα στον κανονικοποιημένο λόγο απώλειας πακέτων (Packet Loss Ratio) και στην απόσταση μεταξύ των δύο κόμβων.

Φαίνεται ότι ο λόγος τείνει στην 1 όταν η απόσταση πλησιάζει τα 500μ.

4.2 Μετρήσεις σε ανοιχτό πεδίο με εμπόδια

Οι μετρήσεις σε ανοιχτό πεδίο με εμπόδια πραγματοποιήθηκαν σε χώρο με πολυκατοικίες δίπλα στη νέα παραλία Θεσσαλονίκης όπως φαίνεται στην Εικόνα 4.5.



Εικόνα 4.5: Μετρήσεις σε ανοιχτό πεδίο με εμπόδια σε χώρο με πολυκατοικίες

Πραγματοποιήθηκαν πολλές μετρήσεις σε ανοιχτούς χώρους με εμπόδια και θεωρήσαμε ότι η συγκεκριμένη περιοχή είναι ένα καλό σημείο αναφοράς με ομοιόμορφο δάπεδο χωρίς κλίση.

Το ύψος της κάθε συσκευής είναι στο 1.5 μέτρο από το δάπεδο.

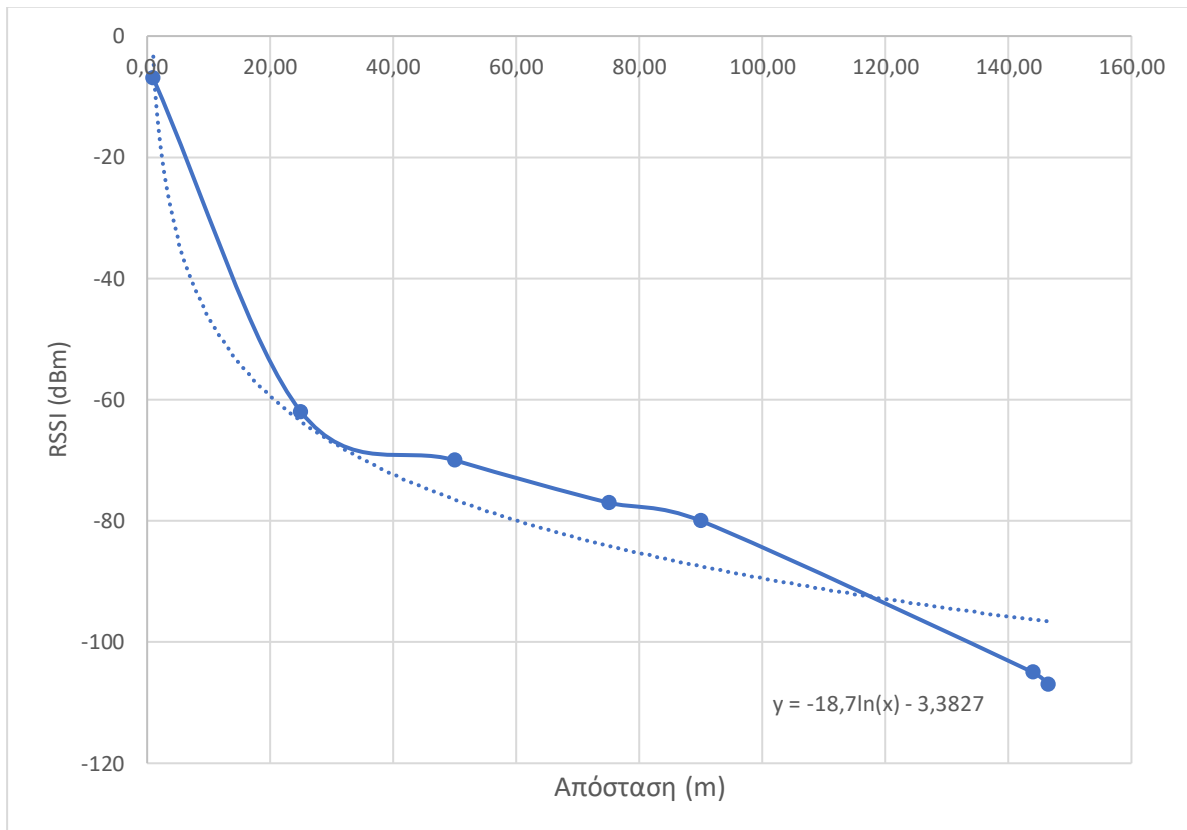
Ο server-device έμεινε ακίνητος στο σημείο 0 και ο client-device απομακρύνεται από αυτόν σε ευθεία γραμμή.

Ανάλογα τα εμπόδια και τις πολυκατοικίες η μέτρηση της απόστασης διέφερε συνεχώς με σημαντικές μεταβολές.

Προφανώς πολύ σημαντικό ρόλο στην απόσταση έχει το πλήθος, ο όγκος, η ύπαρξη διπλανών κτιρίων για αυτό οι μετρήσεις δεν μπορούν να οδηγήσουν σε αξιόπιστα αποτελέσματα.

Μπορούμε όμως να κάνουμε μια εκτίμηση της εμβέλειας μεταξύ των δύο κόμβων σε διάφορα σημεία χωρίς αυτό να εκφράζει με ακρίβεια την εμβέλεια μέσα σε κατοικημένη περιοχή.

Μια σειρά μετρήσεων πραγματοποιήθηκε στην περιοχή που φαίνεται στην Εικόνα 4.5.



Εικόνα 4.6: Σχέση ανάμεσα στην απόσταση των δύο συσκευών και το RSSI σε ανοιχτό πεδίο με συγκεκριμένα εμπόδια

Στην Εικόνα 4.6 παρουσιάζεται η σχέση ανάμεσα στην απόσταση των δύο συσκευών και το RSSI.

Κάθε τελεία αντιπροσωπεύει το μέσο όρο 5 μετρήσεων για κάθε απόσταση με διαφορά 25 μέτρα και η τελευταία 10 μετρήσεις.

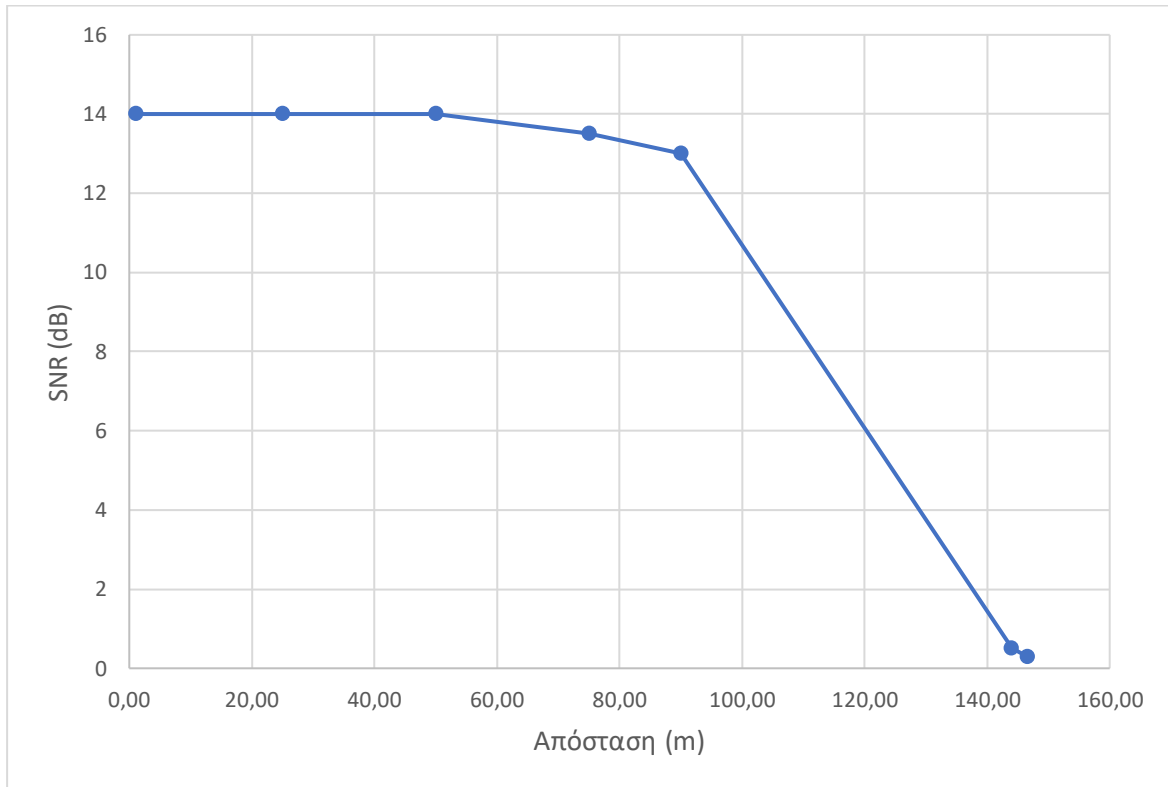
Η διακεκομμένη μπλε γραμμή παριστάνει την λογαριθμική γραμμή τάση και αναγράφεται και η εξίσωση που προκύπτει.

Προκύπτει ότι στο συγκεκριμένο ανοιχτό χώρο με εμπόδια η μέγιστη απόσταση (μέση τιμή) μεταξύ δύο κόμβων (server και client) είναι:

150.54 m

με τυπική απόκλιση

1.12 m



Εικόνα 4.7: Σχέση ανάμεσα στην απόσταση των δύο συσκευών και το SNR σε ανοιχτό πεδίο με συγκεκριμένα εμπόδια

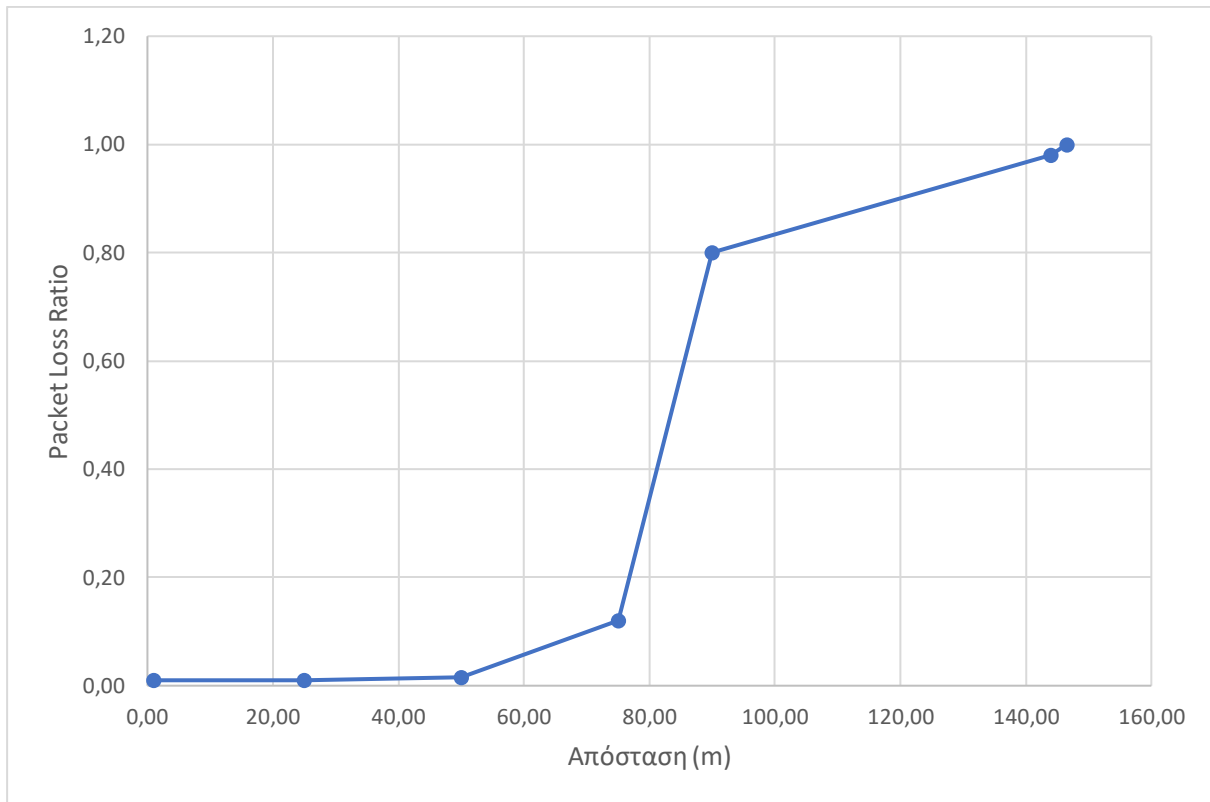
Φαίνεται ότι μετά στα 100 μέτρα το RSSI βρίσκεται σε χαμηλά επίπεδα και παραμένει σε αυτά μέχρι τα 150μ όπου το σήμα-σύνδεση χάνεται και δεν λαμβάνεται σωστά το πακέτο.

Στην Εικόνα 4.7 παρουσιάζεται η σχέση ανάμεσα στην απόσταση των δύο συσκευών και το SNR.

Κάθε τελεία αντιπροσωπεύει το μέσο όρο 5 μετρήσεων για κάθε απόσταση σε μέτρα:

- 1,00
- 25,00
- 50,00
- 75,00
- 90,00
- 144,00
- 146,50

Παρατηρούμε ότι μετά τα 80 μέτρα το SNR μειώνεται με μεγαλύτερη κλίση μέχρι ο λόγος σήματος προς θόρυβο γίνεται 1 και το πακέτο που λαμβάνεται παρουσιάζει την μέγιστη απώλεια στην λήψη.



Εικόνα 4.8: Η σχέση ανάμεσα στον κανονικοποιημένο λόγο απώλειας πακέτων (Packet Loss Ratio) και στην απόσταση μεταξύ των δύο κόμβων σε ανοιχτό πεδίο με συγκεκριμένα εμπόδια

Στην Εικόνα 4.8 παρουσιάζεται η σχέση ανάμεσα στον κανονικοποιημένο λόγο απώλειας πακέτων (Packet Loss Ratio) και στην απόσταση μεταξύ των δύο κόμβων.

Φαίνεται ότι ο λόγος τείνει στην 1 όταν η απόσταση πλησιάζει τα 146μ.

Πραγματοποιήθηκαν και άλλες μετρήσεις στη συγκεκριμένη περιοχή. Δύο από αυτές φαίνονται στις εικόνες 4.9 και 4.10.

Παρατηρούμε ότι η μέγιστη εμβέλεια διαφέρει ανα περίπτωση.

Προκύπτει ότι για τον ανοιχτό χώρο με εμπόδια της Εικόνας 4.9 η μέγιστη απόσταση (μέση τιμή) μεταξύ δύο κόμβων (server και client) είναι:

146.5 m

με τυπική απόκλιση

0.97 m



Εικόνα 4.9: Μετρήσεις σε δεύτερο ανοιχτό χώρο με εμπόδια

Προκύπτει ότι για τον ανοιχτό χώρο με εμπόδια της Εικόνας 4.10 η μέγιστη απόσταση (μέση τιμή) μεταξύ δύο κόμβων (server και client) είναι:

110.05 m

με τυπική απόκλιση

0.8 m



Εικόνα 4.10: Μετρήσεις σε τρίτο ανοιχτό χώρο με εμπόδια

Να σημειωθεί ότι σε όλες τις εικόνες έχουν ληφθεί από το maps.google και οι αποστάσεις που φαίνονται δεν είναι οι τελικές.

Επίσης πραγματοποιήθηκαν μετρήσεις σε ανοιχτό χώρο τοποθετώντας την μια συσκευή-ακίνητη(server) σε γέφυρα με ύψος 6 μέτρα από το έδαφος ενώ η άλλη ήταν στο 1.5 μέτρο.

Ο server-device έμεινε ακίνητος στο σημείο 0 και ο client-device απομακρυνόταν από αυτόν σε ευθεία γραμμή.

Προκύπτει ότι για τον ανοιχτό χώρο με τη μια συσκευή πιο ψηλά όπως φαίνεται στην Εικόνας 4.11 η μέγιστη απόσταση (μέση τιμή) μεταξύ δύο κόμβων (server και client) είναι:

506 m

με τυπική απόκλιση

2.1 m

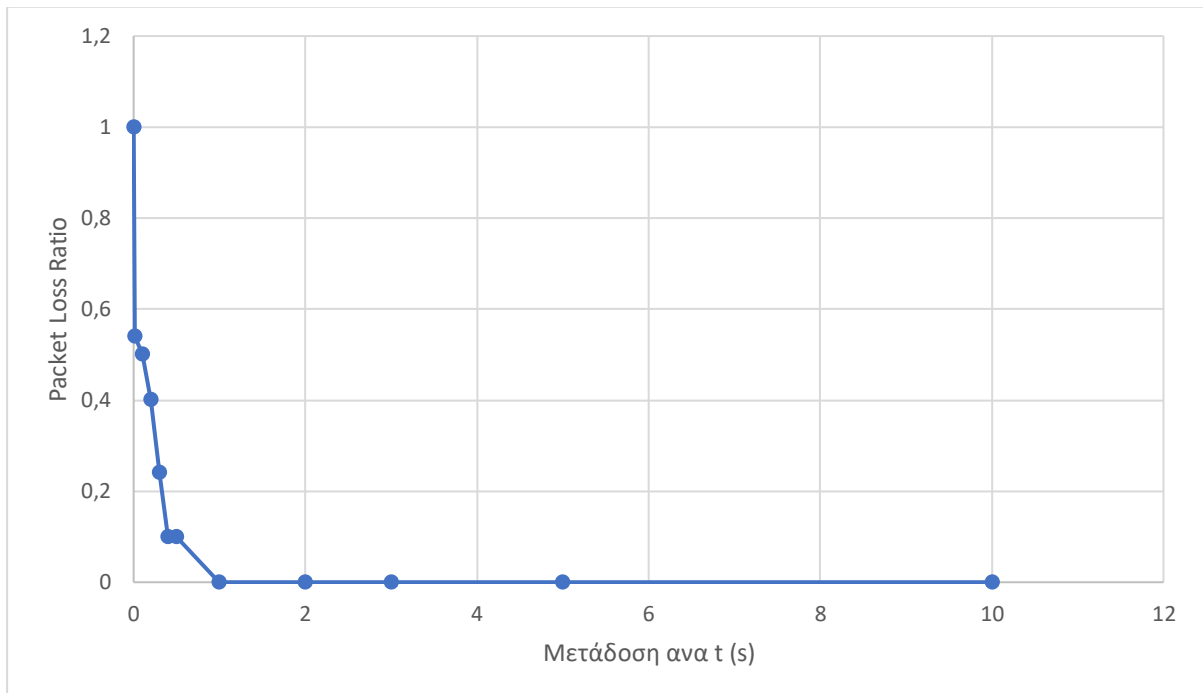


Εικόνα 4.11: Μετρήσεις σε ανοιχτό χώρο με τη μια συσκευή να βρίσκεται σε μεγαλύτερο ύψος

Πραγματοποιήθηκαν πειράματα με ένα επιπλέον χρήστη-client σε διαφορετική συσκευή και με τρεις επιπλέον clients σε μια συσκευή με διαφορετικό πρόγραμμα.

Στάλθηκαν δεδομένα από τον κάθε client για να εξεταστεί πόσα πακέτα χάνονται όταν στέλνουν όλοι μαζί σε διάφορους χρόνους χωρίς επιβεβαίωση.

Βρέθηκε ότι σε απόσταση 50m.



Εικόνα 4.12: Η σχέση ανάμεσα στον κανονικοποιημένο λόγο απώλειας πακέτων (Packet Loss Ratio) και στον χρόνο αποστολής μεταξύ των δύο κόμβων σε ανοιχτό πεδίο

Όπως φαίνεται στην Εικόνα 4.12 όταν στέλνουν πακέτο με διάστημα πάνω από 1s δεν χάνεται κανένα πακέτο.

Κάτω από αυτόν τον χρόνο χάνονται αρκετά πακέτα με κορύφωση στο 0,001s.

Κεφάλαιο 5ο: Συμπεράσματα και προτάσεις βελτίωσης

Παρουσιάστηκε η μελέτη και υλοποίηση ενός συστήματος LoRa επικοινωνίας μεταξύ δύο συσκευών με το module SX1278 και συγκεκριμένη κεραία που χρησιμοποιήθηκε μαζί με το Raspberry Pi για την λήψη και μετάδοση.

Μελετήθηκαν τα χαρακτηριστικά αυτής της επικοινωνίας για την εκτίμηση της μέγιστης απόστασης της επικοινωνίας σε ανοιχτό χώρο αλλά και σε χώρο με συγκεκριμένα εμπόδια. Σε ανοιχτό η μέγιστη απόσταση (μέση τιμή) βρέθηκε 499.47 m με τυπική απόκλιση 2.03 m, ενώ σε συγκεκριμένο ανοιχτό χώρο με εμπόδια η μέγιστη απόσταση (μέση τιμή) μεταξύ δύο κόμβων (server και client) είναι: 150.54 m με τυπική απόκλιση 1.12 m.

Υλοποιήθηκε ένα είδος απλού πρωτοκόλλου επικοινωνίας για μια απλή τοπολογία αστέρα και επικοινωνία κόμβων μέσω ενός κεντρικού με το διαδίκτυο. Σχεδιάστηκε τοπολογία που αφορά ένα θερμοκήπιο αλλά και για πολλά θερμοκήπια και πως μπορούν να επικοινωνούν οι κόμβοι μεταξύ τους.

Το σύστημα που υλοποιήθηκε μπορεί να βρει εφαρμογή σε εγκαταστάσεις με πολλά θερμοκήπια στα οποία έχουν τοποθετηθεί αισθητήρες-κόμβοι-clients σε διάφορα σημεία στο εσωτερικό του. Κάθε θερμοκήπιο μπορεί να περιέχει έναν τοπικό server ο οποίος να επικοινωνεί με ένα κεντρικό ή μπορεί να αποστέλλουν τα δεδομένα απευθείας στον κεντρικό, όταν μπορεί να είναι εφικτή η επικοινωνία λόγω απόστασης, Το LoRa φαίνεται ότι καλύπτει μεγάλες αποστάσεις που μπορούν να καλύψουν μεγάλα θερμοκήπια κάτω από 500 μέτρα.

Θεωρήσαμε ότι το πιο σημαντικό στοιχείο είναι η αναγνώριση και ασφάλεια στη μετάδοση και η προβολή των δεδομένων μέσω διαδικτύου και όχι η μέτρηση της ταχύτητας αφού η ανάγνωση των αισθητήρων και η αποστολή τους γίνεται σε αραιά χρονικά διαστήματα (>1 min).

Όσον αφορά τις βελτιώσεις από τις πιο σημαντικές είναι η προσθήκη λειτουργιών-εντολών του server μέσω ή όχι σε κάθε κόμβο. Επίσης, είναι πολύ σημαντικό να τοποθετηθεί μηχανισμός που να 'κοιμίζει' και να 'ξυπνάει' τον κάθε κόμβο όταν χρειάζεται να διαβάσει τις τιμές των αισθητήρων. Αυτό βέβαια θα μπορούσε να πραγματοποιηθεί με χρήση ενσωματωμένων συστημάτων, όπως esp32. Επίσης, πρέπει να μελετηθεί η τεχνική που μπορεί να ξυπνήσει τον κόμβο client ο κόμβος server μέσω του LoRa module.

Τέλος, σε μια πραγματική εγκατάσταση μπορεί να γίνει μια εκτίμηση της κατανάλωσης ενέργειας και τι είδους μπαταρία θα χρειαζόταν για την βέλτιστη λειτουργία του. Η κατανάλωση θα μπορούσε να μειωθεί αν επιλεχθούν χαμηλότερης ενέργειας, όπως ενσωματωμένα συστήματα όπως το esp32.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Augustin A, Yi J, Clausen T, Townsley WM. A Study of LoRa: Long Range & Low Power Networks for the Internet of Things. *Sensors*. 2016; 16(9):1466. <https://doi.org/10.3390/s16091466>
- [2] <https://en.wikipedia.org/wiki/LoRa>
- [3] Kim S, Chong J-W. Chirp Spread Spectrum Transceiver Design and Implementation for Real Time Locating System. *International Journal of Distributed Sensor Networks*. August 2015. doi:10.1155/2015/572861
- [4] <https://www.semtech.com/lora/what-is-lora>
- [5] S. Persia, C. Carciofi and M. Faccioli, "NB-IoT and LoRA connectivity analysis for M2M/IoT smart grids applications," *2017 AEIT International Annual Conference*, 2017, pp. 1-6, doi: 10.23919/AEIT.2017.8240558.
- [6] S. C. Gaddam and M. K. Rai, "A Comparative Study on Various LPWAN and Cellular Communication Technologies for IoT Based Smart Applications," *2018 International Conference on Emerging Trends and Innovations In Engineering And Technological Research (ICETIETR)*, 2018, pp. 1-8, doi: 10.1109/ICETIETR.2018.8529060.
- [7] D. Patel and M. Won, "Experimental Study on Low Power Wide Area Networks (LPWAN) for Mobile Internet of Things," *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*, 2017, pp. 1-5, doi: 10.1109/VTCSpring.2017.8108501.
- [8] https://en.wikipedia.org/wiki/Low-power_wide-area_network
- [9] <https://lora-alliance.org/about-lorawan/>
- [10] <https://www.thethingsnetwork.org/docs/lorawan/>
- [11] <https://www.i-scoop.eu/internet-of-things-iot/lpwan/iot-network-lora-lorawan/>
- [12] <https://lora-alliance.org/wp-content/uploads/2020/11/what-is-lorawan.pdf>
- [13] <https://www.iotforall.com/lpwan-benefits-vs-iot-connectivity-options>
- [14] <https://www.iotforall.com/iot-connectivity-comparison-lora-sigfox-rpma-lpwan-technologies>
- [15] https://en.wikipedia.org/wiki/Narrowband_IoT
- [16] <https://www.sciencedirect.com/topics/computer-science/packet-loss-ratio>
- [17] <https://lora.readthedocs.io/en/latest/>
- [18] Z A Tan et al, ' Analysis on LoRa RSSI in Urban, Suburban, and Rural Area for Handover Signal Strength-Based Algorithm.', 2019 IOP Conf. Ser.: Mater. Sci. Eng. 705 012012
- [19] W. Choi, Y.-S. Chang, Y. Jung and J. Song. Low-power LoRa signal-based outdoor positioning using fingerprint algorithm. *ISPRS International Journal of Geo-Information*, 7(11):440, 2018.

- [20] LoRa Alliance Strategy Committee. LoRaWAN geolocation whitepaper. 2018.
- [21] J. Danebjer and V. Halldorsson. A hybrid approach to GPS-free geolocation over LoRa. 2018.
- [22] Augustin A, Yi J, Clausen T, Townsley WM. A Study of LoRa: Long Range & Low Power Networks for the Internet of Things. *Sensors*. 2016; 16(9):1466. <https://doi.org/10.3390/s16091466>
- [23] O. DIENG, C. PHAM and O. THIARE, "Outdoor Localization and Distance Estimation Based on Dynamic RSSI Measurements in LoRa Networks: Application to Cattle Rustling Prevention," *2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2019, pp. 1-6, doi: 10.1109/WiMOB.2019.8923542.
- [24] M. Aref and A. Sikora, "Free space range measurements with Semtech Lora™ technology," 2014 2nd International Symposium on Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems, 2014, pp. 19-23, doi: 10.1109/IDAACS-SWS.2014.6954616.]
- [25] F. Gui, Y. Yin and A. Lu, "Research on Ship Wireless Communication Technology Based on LoRa," *2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, 2018, pp. 976-979, doi: 10.1109/IMCEC.2018.8469621
- [26] J. Wang, S. Yi, D. Zhan and W. Zhang, "Design and Implementation of small monitoring wireless network system based on LoRa," *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, 2019, pp. 296-299, doi: 10.1109/IAEAC47372.2019.8997877.
- [27] Shuhaizar Daud et al 2018 IOP Conf. Ser.: Mater. Sci. Eng. 318 012053, IOP Conference Series: Materials Science and Engineering, Volume 318, Malaysian Technical Universities Conference on Engineering and Technology 2017 (MUCET 2017) 6–7 December 2017, Penang, Malaysia
- [28] V. A. Vu, D. Cong Trinh, T. C. TRUVANT and T. Dang Bui, "Design of automatic irrigation system for greenhouse based on LoRa technology," *2018 International Conference on Advanced Technologies for Communications (ATC)*, 2018, pp. 72-77, doi: 10.1109/ATC.2018.8587487.
- [29] Thair A. Salih and Mohammed S. Noori 2020 *IOP Conf. Ser.: Mater. Sci. Eng.* **928** 032058
- [30] Rahabul Islam, Md. Wahidur Rahman, Rahmina Rubaiat, Md. Mahmudul Hasan, Md. Mahfuz Reza, Mohammad Motiur Rahman, LoRa and server-based home automation using the internet of things (IoT), *Journal of King Saud University - Computer and Information Sciences*, 2021
- [31] Rudy Susanto et al, Comparison of three LoRa devices and its application on street light monitoring system, 2018 IOP Conf. Ser.: Earth Environ. Sci. **195** 012066

- [32] <https://www.semtech.com/products/wireless-rf/lora-core/sx1278>
- [33] Nayyar, Anand and Vikram Puri. “Raspberry PiA Small , Powerful , Cost Effective and Efficient Form Factor Computer : A Review.” (2016).
- [34] <https://www.wired.co.uk/article/raspberry-pi-three-wifi-bluetooth-release-price-cost>
- [35] https://en.wikipedia.org/wiki/Raspberry_Pi
- [36] <https://www.halvorsen.blog/documents/technology/iot/thingspeak/resources/ThingSpeak.pdf>
- [37] <https://iot4beginners.com/thingspeak-tutorial/>
- [38] <https://docs.python.org/3/tutorial/>
- [39] <https://www.seeedstudio.com/blog/2020/01/14/how-to-create-and-run-a-raspberry-pi-python-program-beginners-tutorial/>
- [40] <https://www.geeksforgeeks.org/python-introduction-to-web-development-using-flask/>
- [41] <https://www.guru99.com/flask-vs-django.html>
- [42] <https://www.educative.io/courses/flask-develop-web-applications-in-python/qVyQAA6y7A2>
- [43] <https://werkzeug.palletsprojects.com/en/2.0.x/>