



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ

Διπλωματική Εργασία

Σχεδίαση και ανάπτυξη συστήματος Γεωργίας Ακριβείας με εφαρμογή τεχνολογιών του Διαδικτύου των Πραγμάτων



Φοιτητής: ΠΑΠΑΝΙΚΟΛΑΟΥ ΑΠΟΣΤΟΛΟΣ

ΑΜ: 210017146

Επιβλέπων Καθηγητής

ΔΙΟΝΥΣΗΣ ΚΑΝΔΡΗΣ

Καθηγητής

ΑΘΗΝΑ-ΑΙΓΑΛΕΩ, ΙΟΥΝΙΟΣ 2022



UNIVERSITY OF WEST ATTICA
FACULTY OF ENGINEERING
DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING

Diploma Thesis

Design and development of Precision Agriculture system by using IoT technologies



Student: APOSTOLOS PAPANIKOLAOU
Registration Number: 212017146

Supervisor:
DIONISIS KANDRIS,
Professor

ATHENS-EGALEO, JUNE 2022

Η Διπλωματική Εργασία έγινε αποδεκτή και βαθμολογήθηκε από την εξής τριμελή επιτροπή:

Διονύσης Κανδρής, Καθηγητής	Παναγιώτης Παπαγέωργας, Καθηγητής	Γρηγόρης Κουλούρας, Αναπληρωτής Καθηγητής
(Υπογραφή)	(Υπογραφή)	(Υπογραφή)

Copyright © Με επιφύλαξη παντός δικαιώματος. All rights reserved.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ και Απόστολος Παπανικολάου, Ιούνιος, 2022

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τους συγγραφείς.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα του και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις θέσεις του επιβλέποντος, της επιτροπής εξέτασης ή τις επίσημες θέσεις του Τμήματος και του Ιδρύματος.

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

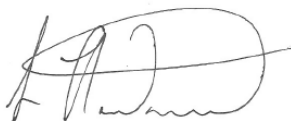
Ο κάτωθι υπογεγραμμένος Παπανικολάου Απόστολος του Γεωργίου, με αριθμό μητρώου 212017146 φοιτητής του Πανεπιστημίου Δυτικής Αττικής της Σχολής ΜΗΧΑΝΙΚΩΝ του Τμήματος ΗΛΕΚΤΡΟΛΟΓΩΝ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ,

δηλώνω υπεύθυνα ότι:

«Είμαι συγγραφέας αυτής της διπλωματικής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος.

Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του διπλώματός μου.»

Ο Δηλών
Απόστολος Παπανικολάου



Περίληψη

Η σημερινή εξέλιξη της τεχνολογίας αλλάζει δραστικά την ζωή των ανθρώπων, τις επιχειρήσεις αλλά και την παγκόσμια οικονομία.

Μετά την σπουδαία ανακάλυψη του διαδικτύου ακολουθεί ακόμη και στις μέρες μας η ανάπτυξη του IoT που έχει απώτερο σκοπό να θέσει το τεχνολογικό πλαίσιο της επικοινωνίας μεταξύ αντικειμένων.

Έτσι, από τις αρχές της δεκαετίας του 90 και την ανακάλυψη της RFID ετικέτας, πραγματοποιήθηκαν σημαντικότητα βήματα στον τομέα της επικοινωνίας συσκευών τα οποία οδήγησαν σε ασύρματα δίκτυα ευρείας περιοχής χαμηλής ισχύος LPWAN (Low Power Wide Area Network). Μέρος των LPWAN αποτελεί και το LORAWAN το οποίο είναι ένα πρωτόκολλο σύνδεσης ασυρμάτων συσκευών μέσω της τεχνολογίας LORA.

Η τεχνολογία LoRa χρησιμοποιείται σήμερα από διάφορους τομείς όπου απαιτείται επικοινωνία μικρού όγκου δεδομένων με ελάχιστη ενέργεια και μεγάλη εμβέλεια κάλυψης.

Ένας από τους τομείς όπου έχει ήδη ξεκινήσει να εφαρμόζεται είναι ο τομέας της γεωργίας. Η ιδέα είναι ότι σύγχρονοι αλλά και έξυπνοι αισθητήρες διαφόρων ειδών συλλέγουν σημαντικές πληροφορίες με εξαιρετική ακρίβεια από διαφορετικά σημεία του χώρου. Στην συνέχεια, με την χρήση της τεχνολογίας LoRa όλα αυτά τα δεδομένα αποστέλλονται ασύρματα προς επεξεργασία με σκοπό να αυτοματοποιηθούν διαδικασίες για την μεγιστοποίηση της παραγωγής.

Συνεπώς, σιγά-σιγά πραγματοποιείται μετάβαση από ένα κλασικό μοντέλο γεωργίας σε ένα νέο μοντέλο “γεωργίας ακριβείας” όπου η τεχνολογία υπόσχεται να συμβάλει τα μέγιστα με απόλυτη ακρίβεια στους χειρισμούς που απαιτούνται.

Στην πορεία της παρούσας εργασίας θα γίνει ανάλυση της τεχνολογίας LoRa και θα εξηγήσουμε τους λόγους για τους οποίους θεωρείται σήμερα η βέλτιστη επιλογή για τις αγροκαλλιέργειες. Επίσης, υλοποιείται μια κατασκευή από διάφορους αισθητήρες μετρήσεων έτσι ώστε να λαμβάνονται δυναμικά δεδομένα, τα οποία σε τακτά χρονικά διαστήματα θα αποστέλλονται προς αποθήκευση και επεξεργασία σε μια πλατφόρμα διαχείρισης με απώτερο σκοπό την αυτόματη λήψη αποφάσεων βελτιστοποίησης.

Λέξεις – κλειδιά

LPWAN, LoRa, End-devices, Sensors, LoRaWAN, Backend Infrastructure, InfluxDB, Grafana, Web Interface

Abstract

Today's development of technology drastically changes the lives of people, businesses and the global economy.

After the great discovery of the internet follows even today the development of IoT which has the ultimate goal to set the technological framework of communication between objects.

Thus, since the beginning of the 90's and the discovery of the RFID tag, significant steps have been taken in the field of device communication which have led to LPWAN (Low Power Wide Area Network) wireless networks. Part of LPWAN is LORAWAN which is a protocol for connecting wireless devices through LORA technology.

LoRa technology is currently used in various fields where small data communication with minimal energy and wide coverage is required.

One of the areas where it has already started to be implemented is the agricultural sector. The idea is that modern as well as smart sensors of various types collect important information with great accuracy from different parts of the space. Then, using LoRa technology, all this data is sent wirelessly for processing in order to automate processes to maximize production.

Therefore, there is a gradual transition from a classic model of agriculture to a new model of "precision agriculture" where technology promises to contribute the maximum with absolute precision to the manipulations required.

In the course of this work, LoRa technology will be analyzed and we will explain the reasons why it is currently considered the best choice for agricultural crops. Also, a construction is implemented by various measurement sensors so as to receive dynamic data, which at regular intervals will be sent for storage and processing on a management platform with the ultimate goal of automatic optimization decisions.

Keywords

LPWAN, LoRa, End-devices, Sensors, LoRaWAN, Backend Infrastructure, InfluxDB, Grafana, Web Interface

Περιεχόμενα

Κατάλογος Πινάκων.....	9
Κατάλογος Εικόνων	10
Αλφαβητικό Ευρετήριο.....	19
1. Γεωργία Ακριβείας.....	20
1.1 Εισαγωγή κεφαλαίου 1	20
1.2 Παράγοντες που επηρεάζουν τη Γεωργία	20
1.3 Γεωργία ακριβείας	21
1.4 Τεχνολογία και IoT στη γεωργία	22
2. IoT και Ασύρματα Δίκτυα Αισθητήρων (WSN).....	26
2.1 Εισαγωγή	26
2.2 Αναφορά στις τεχνολογίες ασύρματης δικτύωσης LPWAN	26
2.2.1 Τι είναι η τεχνολογία LPWAN – Low Power Wide Area Network	26
2.2.2 LTE-M (Long Term Evolution – Machine Type Communication) & NB-IoT (Narrow Band – Internet of Things)	27
2.2.3 Sigfox	28
2.2.4 Ορισμός της Τεχνολογίας LoRa	32
2.3 Ορισμός και Αρχιτεκτονική πρωτοκόλλου LoRaWAN	35
2.3.1 Τελικές Συσκευές	36
2.3.2 Gateways - Πύλες Δικτύου	37
2.3.3 Εξωτερικές Επιλογές	38
2.3.4 Διακομιστής Δικτύου - Network Server	40
2.3.5 Διακομιστής εφαρμογών - Application Server	41
2.3.6 Διακομιστής σύνδεσης - Join Server	42
2.4 Τεχνικές λεπτομέρειες τεχνολογίας στο LoRa	42
2.4.1 Τοπικές παράμετροι.....	42
2.4.2 Μηνύματα	43
2.4.3 Ασφάλεια	46
2.4.4 Frame Counters - Μετρητές πλαισίων.....	47
2.4.5 Spread Spectrum - Ευρύ φάσμα	48
2.4.6 Classes Συσκευών.....	49
2.4.7 Ενεργοποίηση συσκευών.....	52
2.4.8 Activation by Personalization (ABP).....	56
2.4.9 Spreading Factors - Παράγοντες διάδοσης.....	57
2.4.10 Προσαρμοστικός αλγόριθμος διάδοσης (ADR – Adaptive Data Rate)	62
3. Υλοποίηση του Συστήματος.....	64
3.1 Εισαγωγή κεφαλαίου 3	64
3.2 Σύνδεση με το διαδίκτυο των πραγμάτων TTN – Things of Network	65
3.2.1 Διαδικασία εγγραφής στο TTN	65
3.2.2 Διαδικασία καταχώρησης δρομολογητή στο TTN	68
3.2.3 Δημιουργία εφαρμογών στο TTN	71
3.2.4 Δημιουργία Τελικής Συσκευής στο TTN.....	73
3.3 Δρομολογητής LoRa Gateway - Mikrotik wAP LR8 kit	76
3.3.1 Τεχνικά χαρακτηριστικά του wAP LR8 kit	77
3.3.2 Εγκατάσταση & Σύνδεση με το διαδίκτυο των πραγμάτων (TTN).	78

3.4 Αισθητήρες & Μικροελεγκτές	82
3.4.1 Ψηφιακός αισθητήρας θερμοκρασίας DHT 22.....	83
3.4.2 Σύνδεση και προγραμματισμός του DHT 22.....	85
3.4.3 Αναλογικός αισθητήρας θερμοκρασίας εδάφους DS18B20	87
3.4.4 Σύνδεση και προγραμματισμός του DS18B20.....	88
3.4.5 Ψηφιακός αισθητήρας BMP280 μέτρησης ύψους & βαρομετρικής πίεσης.....	90
3.4.6 Σύνδεση και προγραμματισμός του BMP280.....	91
3.4.7 Αναλογικός αισθητήρας υγρασίας εδάφους.....	93
3.4.8 Σύνδεση και προγραμματισμός του αισθητήρα υγρασίας εδάφους.....	93
3.4.9 Αναλογικός αισθητήρας φωτός – Photo resistor 5mm	94
3.4.10 Σύνδεση και προγραμματισμός του αισθητήρα φωτός	95
3.4.11 Ψηφιακός αισθητήρας βροχής – YL 38.....	96
3.4.12 Σύνδεση και προγραμματισμός του αισθητήρα βροχής.....	97
3.5 Επιλογή Tranceiver και σύνδεση με έναν μικροελεγκτή Arduino Uno	98
3.5.1 LoRa Tranceiver RFM95W	98
3.6 Δημιουργία πολυαισθητήρα	107
3.7 Δεύτερη τελική συσκευή - Ενεργοποιητές του συστήματος (Πότισμα, υδρονέφωση, Λάμπες Θερμότητας)	114
3.7.1 Διαδικασία κατασκευής δεύτερης τελικής συσκευής	114
3.7.2 Καταχώρηση δεύτερης τελικής συσκευής στον TTN	115
3.7.3 Κώδικας λειτουργίας δεύτερης τελικής συσκευής.....	116
3.8 Σύνδεση ενεργοποιητών συστήματος	123
3.8.1 Δημιουργία ενός VM Ubuntu 20.04 στο Digital Ocean	130
3.8.2 Εγκατάσταση του προγράμματος NodeRed και όλων των απαραίτητων modules για την δημιουργία του Web Interface.....	135
3.8.3 Εγκατάσταση της βάσης InfluxDB και σύνδεση με το The things Network.....	140
3.8.4 Εγκατάσταση του λογισμικού Grafana για την προσθήκη γραφικών παραστάσεων	142
3.8.5 Δημιουργία ενός Web interface	144
4. Επίλογος	160
4.1 Εισαγωγή κεφαλαίου	160
4.2 Σύνοψη εργασίας	160
4.3 Προβλήματα – Αντιμετώπιση	161
4.4 Συμπεράσματα – Παρατηρήσεις	164
4.5 Προτάσεις μελλοντικής εξέλιξης	165
Βιβλιογραφικές αναφορές	167

Κατάλογος Πινάκων

Πίνακας 2.1 – Τεχνικές λεπτομέρειες τεχνολογίας Sigfox

Πίνακας 2.2 – Περιφερειακά πλάνα παραμέτρων για την περιοχή της Ασίας

Πίνακας 2.3 – Συχνότητες λειτουργίας LoRa για την Ολλανδία

Πίνακας 2.4 – Μηνύματα LoRaWAN

Πίνακας 2.5 – Παράγοντες διάδοσης ανά συντελεστή διασποράς

Πίνακας 2.6 – Ρυθμός μετάδοσης σε σχέση με το εύρος ζώνης (bandwidth)

Πίνακας 2.7 – Ευαίσθησία δέκτη με σταθερό bandwidth για διαφορετικό συντελεστή διασποράς

Πίνακας 3.1 – Βασικά χαρακτηριστικά του DS18B20

Πίνακας 3.2 – Τεχνικά χαρακτηριστικά BMP280

Πίνακας 3.3 – Βασικές προδιαγραφές φωτοαντίστασης

Πίνακας 3.4 - Τεχνικά Χαρακτηριστικά του ΥL-38

Πίνακας 3.5 – Βασικά χαρακτηριστικά του RFM95W

Πίνακας 3.6 – Έξοδοι ενεργοποιητών στο Arduino

Κατάλογος Εικόνων

Εικόνα 1.1 – Χρήση ρομποτικών διατάξεων στο πεδίο των αγροκαλλιιεργειών

Εικόνα 1.2 – Χαρτογράφηση αγροτικής περιοχής μέσω GIS

Εικόνα 2.1 – Συγκριτικό διάγραμμα ασύρματων τεχνολογιών

Εικόνα 2.2 – Ζώνες συχνοτήτων (Radio configuration) Sigfox

Εικόνα 2.3 – Τοπολογία δικτύου Sigfox

Εικόνα 2.4 - Ανοδικό σήμα sigfox με εμφανές το frequency hopping

Εικόνα 2.5 – LoRa & LoRaWAN

Εικόνα 2.6 – Γραφική απεικόνιση ενός Chirp σήματος

Εικόνα 2.7 – Παγκόσμια κάλυψη δικτύου LoRa

Εικόνα 2.8 – Εκδόσεις του πρωτοκόλλου LoRaWAN

Εικόνα 2.9 – Αρχιτεκτονική του πρωτοκόλλου LoRaWAN

Εικόνα 2.10 - Πολυαισθητήρας γενικής χρήσης

Εικόνα 2.11 – Αισθητήρας ποιότητας ατμοσφαιρικού αέρα

Εικόνα 2.12 – Μετρητής νερού

Εικόνα 2.13 - The Things Indoor gateway

Εικόνα 2.14 - Wirnet iFemtoCell

Εικόνα 2.15 - LG308 Indoor LoRaWAN Gateway

Εικόνα 2.16 - MultiTech Conduit® IP67 Base Station

Εικόνα 2.17 - Cisco IXM-LPWA-800-16-K9

Εικόνα 2.18 - MikroTik RBwAPR-2nD&R11e-LoRa8

Εικόνα 2.19 - PC31 868/915MHz FR4 PCB Internal Antenna

Εικόνα 2.20 - LoRa omnidirectional fiberglass antenna - 8 dBi

Εικόνα 2.21 - Lora 868Mhz single band Fiberglass antenna

Εικόνα 2.22 – Διακομιστής Δικτύου (Network Server)

Εικόνα 2.23 – Διακομιστής εφαρμογών (Application server)

Εικόνα 2.24 – Δομή μηνύματος LoRaWAN

Εικόνα 2.25 – Λειτουργίας τελικής συσκευής με κλάση A

Εικόνα 2.26 – Συμπεριφορά συσκευών Κλάσης A κατά την κατερχόμενη ζεύξη (Downlink)

Εικόνα 2.27 – Συμπεριφορά συσκευών Κλάσης B κατά την κατερχόμενη ζεύξη (Downlink)

Εικόνα 2.28 – Συμπεριφορά συσκευών Κλάσης C κατά την κατερχόμενη ζεύξη (Downlink)

Εικόνα 2.29 – Μέθοδος σύνδεσης OTAA, διαδικασία join-request

Εικόνα 2.30 – Μέθοδος σύνδεσης OTAA, διαδικασία join-accept

Εικόνα 2.31 – Μέθοδος σύνδεσης OTAA, συνολική διαδικασία

Εικόνα 2.32 – Ανταλλαγή κλειδιών μεταξύ τελικών συσκευών και διακομιστών κατά την χρήση της μεθόδου ενεργοποίησης OTAA

Εικόνα 2.33 – Μέθοδος ενεργοποίησης ABP, απαιτούμενα κλειδιά λειτουργίας

Εικόνα 2.34 - Μη Διαμορφωμένο σήμα LoRa

Εικόνα 2.35 - Διαμορφωμένο σήμα LoRa

Εικόνα 2.36 – Σήμα LoRa από παλμογράφο

Εικόνα 2.37 – Χρονική διάρκεια συμβόλου LoRa

Εικόνα 2.38 – Χρονική διάρκεια συμβόλου με βάση τον συντελεστή διασποράς (sf)

Εικόνα 3.1 – Βηματική παρουσίαση συνολικής κατασκευής

Εικόνα 3.2 – Δημιουργία λογαριασμού TTN

Εικόνα 3.3 – Επιτυχής ολοκλήρωση δημιουργίας νέου λογαριασμού στο TTN

Εικόνα 3.4 – Επιλογή τοποθεσίας διακομιστή στο TTN

Εικόνα 3.5 – Οθόνη εισαγωγής (login) στο TTN

Εικόνα 3.6 – Επιλογή εφαρμογής ή δρομολογητή

Εικόνα 3.7 (α) – Οθόνη καταχώρησης δρομολογητή

Εικόνα 3.7 (β) – Οθόνη καταχώρησης δρομολογητή

Εικόνα 3.7 (γ) – Οθόνη καταχώρησης δρομολογητή

Εικόνα 3.8 – Λίστα καταχωρημένων δρομολογητών στο TTN

Εικόνα 3.9 – Δημιουργία εφαρμογής στο TTN

Εικόνα 3.10 – Δημιουργία εφαρμογής στο TTN

Εικόνα 3.11 – Πληροφορίες εφαρμογής

Εικόνα 3.12 – Καταχώρηση τελικής συσκευής στο TTN

Εικόνα 3.13(α) – Καταχώρηση στοιχείων τελικής συσκευής

Εικόνα 3.13 (β) – Καταχώρηση στοιχείων τελικής συσκευής

Εικόνα 3.14 – Λίστα καταχωρημένων τελικών συσκευών στο TTN

Εικόνα 3.15 – Χαρακτηριστικά LoRa Module R11e-LR8

Εικόνα 3.16 – Χαρακτηριστικά δρομολογητή wAP LR8

Εικόνα 3.17 (α) – Εσωτερικό δρομολογητή wAP LR8 kit

Εικόνα 3.16 (β) - Εσωτερικό δρομολογητή wAP LR8 kit

Εικόνα 3.17 – Οθόνη σύνδεσης winBox

Εικόνα 3.18 – Winbox διαδικασία login

Εικόνα 3.19 – Κεντρικό μενου δρομολογητή wAP LR8 kit

Εικόνα 3.20 – Διαθέσιμες συσκευές LoRa στο wAP LR8 kit

Εικόνα 3.21 – Διαθέσιμοι διακομιστές LoRa στο wAP LR8 kit

Εικόνα 3.22 – Καταχώρηση LoRa server στο wAP LR8 kit

Εικόνα 3.23 – Καταχώρηση LoRa server στο wAP LR8 kit

Εικόνα 3.24 – Λίστα καταχωρημένων δρομολογητών στο TTN

Εικόνα 3.25 – Αισθητήρας DHT22 (Θερμοκρασίας, υγρασίας)

Εικόνα 3.26 – DHT 22 με τρεις ακροδέκτες

Εικόνα 3.27 – Εσωτερικά μέρη του αισθητήρα DHT-22

Εικόνα 3.28 – Σύνδεση αισθητήρα DHT-22

Εικόνα 3.29 – Αποτέλεσμα αισθητήρα μέσω σειριακής επικοινωνίας

Εικόνα 3.30 – Ψηφιακός αισθητήρας θερμοκρασίας - DS18B20 (Waterproof)

Εικόνα 3.31 – Διάγραμμα δομής του αισθητήρα θερμοκρασίας - DS18B20

Εικόνα 3.32 – Σύνδεση του DS18B20 σε Arduino Uno

Εικόνα 3.33 – Αισθητήρας BMP280 (Θερμοκρασίας, Βαρομετρικής πίεσης)

Εικόνα 3.34 – Σύνδεση του BMP280 σε Arduino Uno

Εικόνα 3.35 – Αναλογικός αισθητήρας Waveshare μέτρησης υγρασίας εδάφους

Εικόνα 3.36 – Σύνδεση του αισθητήρα WaveShare σε Arduino Uno

Εικόνα 3.36 – Αναλογική Φωτοαντίσταση

Εικόνα 3.37 – Σύνδεση φωτοαντίστασης σε Arduino Uno

Εικόνα 3.38 – Ψηφιακός αισθητήρας βροχής YL-38

Εικόνα 3.39 – Σύνδεση του αισθητήρα YL-38 σε arduino Uno

Εικόνα 3.40 – LoRa Module RFM95W

Εικόνα 3.41 – Ακροδέκτες σύνδεσης του RFM95W

Εικόνα 3.42 – Τρόπος σύνδεσης RFM95W με Arduino Uno

Εικόνα 3.43 – Upload κώδικα μέσω Arduino Uno

Εικόνα 3.44 – Διασύνδεση RFM95W με Arduino Uno

Εικόνα 3.45 – Παράθυρο εφαρμογής Smart Agriculture στο TTN

Εικόνα 3.46 – Χαρακτηριστικά τελικής συσκευής στο TTN

Εικόνα 3.47 – Δεδομένα τελικής συσκευής στο TTN σε πραγματικό χρόνο

Εικόνα 3.48 – Μετασχηματισμός δεδομένων μέσω Payload formatter στο TTN

Εικόνα 3.49 (α),(β),(γ),(δ) – Πολυαισθητήρας

Εικόνα 3.50 (α),(β),(γ) – Πολυαισθητήρας με μπαταρίες

Εικόνα 3.51 – Κατασκευή δεύτερης τελικής συσκευής (Arduino nano & RFM95W)

Εικόνα 3.52 – Κατασκευή δεύτερης τελικής συσκευής (Arduino nano & RFM95W & YL-38)

Εικόνα 3.53 – Στοιχεία δεύτερης τελικής συσκευής

Εικόνα 3.54 – Upload κώδικα δεύτερης τελικής συσκευής

Εικόνα 3.55 – Upload κώδικα δεύτερης τελικής συσκευής

Εικόνα 3.56 – Upload κώδικα δεύτερης τελικής συσκευής

Εικόνα 3.57 – Module Ρελέ 12V με 8 κανάλια λειτουργίας

Εικόνα 3.58 – Θερμαντική Λυχνία τύπου LED

Εικόνα 3.59 – Ενδεικτική λυχνία υδρονέφωσης

Εικόνα 3.60 – Αντλία Ποτίσματος 12V

Εικόνα 3.61 – Power Bank 12v – 5V

Εικόνα 3.62 – Τοποθέτηση δεύτερης τελικής συσκευής με ορισμένους ενεργοποιητές

Εικόνα 3.63 – Δοκιμή downlink στην δεύτερη τελική συσκευή

Εικόνα 3.64 – Live Data δεύτερης τελικής συσκευής – Μήνυμα Downlink

Εικόνα 3.65 – Τελική συσκευή μετά από καθοδικό μήνυμα 07 Hex.

Εικόνα 3.66 – Τελική συσκευή με ενεργοποιημένο το ρελέ θερμοαντικής λυχνίας.

Εικόνα 3.67(α),(β),(γ) – Συνολική κατασκευή

Εικόνα 3.68 – Δημιουργία ενός project στο Digital Ocean

Εικόνα 3.69 – Δημιουργία εικονικής μηχανής στο Digital Ocean

Εικόνα 3.70 – Πληροφορίες για την δημιουργία μιας εικονικής μηχανής στο Digital Ocean

Εικόνα 3.71 – Λίστα εικονικών μηχανών που ανήκουν στο project Smart Agriculture

Εικόνα 3.72 – Απομακρυσμένη σύνδεση στο VM μέσω τερματικού με χρήση ssh

Εικόνα 3.73 – Δημιουργία νέου χρήστη και προσθήκη στην ομάδα sudo

Εικόνα 3.74 – Δημιουργία κλειδιών ssh στο τοπικό μας μηχάνημα win10

Εικόνα 3.75 – Μεταφορά δημόσιου κλειδιού στην εικονική μηχανή

Εικόνα 3.76 – Ενημέρωση των authorized_keys της εικονικής μηχανής

Εικόνα 3.77 – Απομακρυσμένη σύνδεση στο VM με ssh και χρήστη 'par'

Εικόνα 3.78 – Εκκίνηση του NodeRed

Εικόνα 3.79 – Δοκιμή του NodeRed μέσω διεύθυνσης Ip και πόρτας 1880

Εικόνα 3.80 – Καταχώρηση domain-name στην υπηρεσία duckdns

Εικόνα 3.81 – Δημιουργία hash password για το NodeRed

Εικόνα 3.82 – Ενεργοποίηση username & password στο NodeRed

Εικόνα 3.83 – Εγκατάσταση του pm2 στον Virtual Server

Εικόνα 3.84 – Ρύθμιση εκκίνησης του NodeRed μέσω pm2

Εικόνα 3.85 – Ενημέρωση του firewall για την πόρτα 1880

Εικόνα 3.86 – Έλεγχος κατάστασης της Influx DB

Εικόνα 3.87 – Δημιουργία χρήστη στην βάση InfluxDB

Εικόνα 3.88 – Enable authenticate στην βάση

Εικόνα 3.89 – Login στην InfluxDB και δημιουργία βάσης agriculture_db

Εικόνα 3.90 – Έλεγχος κατάστασης του Grafana

Εικόνα 3.91 – Εκκίνηση του grafana από browser

Εικόνα 3.92 – Δημιουργία σύνδεσης μέσω πρωτοκόλλου mqtt στο TTN

Εικόνα 3.93 – Εισαγωγή κόμβου mqtt-in

Εικόνα 3.94 – Ιδιότητες κόμβου mqtt-in

Εικόνα 3.95 – Κόμβος mqtt-in συνδεδεμένος στο TTN

Εικόνα 3.96 – Προσθήκη κόμβου json

Εικόνα 3.97 – Κόμβος “change node” με τις ιδιότητές του

Εικόνα 3.98 – Συνολική ροή δεδομένων πολυαισθητήρα από το TTN στο NodeRed

Εικόνα 3.99 – Συνολική ροή δεδομένων πολυαισθητήρα από το TTN στο NodeRed με influxDB

Εικόνα 3.100 – Συνολική ροή δεδομένων αισθητήρα βροχής από το TTN στο NodeRed με influxDB

Εικόνα 3.101 – Κόμβος mqtt-out

Εικόνα 3.102 – Ιδιότητες mqtt-out

Εικόνα 3.103 – Κόμβος mqtt out συνδεδεμένος στο TTN

Εικόνα 3.104 – Καταχώρηση ορίων αυτοματισμών

Εικόνα 3.105 – Ρύθμιση ενεργοποιητών μέσω dashboard

Εικόνα 3.106 – Ρύθμιση ενεργοποιητών του συστήματος

Εικόνα 3.107 – Συνολική ροή προς τους ενεργοποιητές του συστήματος

Εικόνα 3.108 – Δημιουργία γραφικών με το grafana

Εικόνα 3.109 – Query προς την influxDB από το περιβάλλον του Grafana για την κατασκευή γραφικής παράστασης που αφορά την υγρασία της ατμόσφαιρας.

Εικόνα 3.110 – Query προς την influxDB από το περιβάλλον του Grafana για την κατασκευή γραφικής παράστασης που αφορά την υγρασία του εδάφους.

Εικόνα 3.111 – Εικόνα 3.110 – Query προς την influxDB από το περιβάλλον του Grafana για την κατασκευή γραφικής παράστασης που αφορά την θερμοκρασία του εδάφους και της ατμόσφαιρας.

Εικόνα 3.112 – Query προς την influxDB από το περιβάλλον του Grafana για την κατασκευή γραφικής παράστασης που αφορά την ένταση του φωτός.

Εικόνα 3.113 – Query προς την influxDB από το περιβάλλον του Grafana για την κατασκευή γραφικής παράστασης που αφορά την ατμοσφαιρική πίεση.

Εικόνα 3.114 – Δημιουργία ‘embedding iframe’ για την εισαγωγή των γραφικών στο NodeRed

Εικόνα 3.115 – Εισαγωγή του iframe στο NodeRed

Εικόνα 3.116 – Συνολικό user interface της εφαρμογής

Εικόνα 3.117 – Ρυθμίσεις ενεργοποιητών στο NodeRed

Εικόνα 3.118 (α),(β),(γ) – Grafana dashboard

Εικόνα 4.1 – Κατασκευή δρομολογητή LoRa με 2 κανάλια λειτουργίας

Εικόνα 4.2 – Εξωτερικός δρομολογητής με δυο εξωτερικές κεραίες

Εικόνα 4.3 – Οξειδωμένος αισθητήρας υγρασίας εδάφους

Εικόνα 4.4 – Μπαταρία πολυαισθητήρα

Αλφαβητικό Ευρετήριο

3GPP – 3rd Generation Partnership Project

ABP - Activation by Personalization

ADR - Adaptive Data Rate

API – Application Interface

AppEUI - Application Extended Unique Identifier

AppKey - Application Key

AppSKey - Application Session Key

CSS - Chirp Spread Spectrum

DevEUI - Device Extended Unique Identifier

DSSS - Direct Sequence Spread Spectrum

FDMA - Frequency Division Multiple Access

FHSS - Frequency Hopping Spread Spectrum

IoT – Internet of Things

ISM – Industrial, Scientific, Medical

LPWAN – Low Power Wide Area Network

LTE-M - Long Term Evolution – Machine Type Communication

M2M – Machine to Machine

MAC – Media Access Control

MIC - Message Integrity Code

NB-IoT - Narrow Band – Internet of Things

NetID - Network Identifier

NTC - Negative Temperature Coefficient

NwkSkey - Network Session Key

OFDMA - Orthogonal Frequency-Division Multiple Access

OTAA - Over-The-Air-Activation

PoE - Power Over Ethernet

QPSK - Quadrature Phase Shift Keying

SF – Spreading Factors

SNR – Signal to Noise Ratio

WSN – Wireless Sensors Network

TTN - The Things Network

1. Γεωργία Ακριβείας

1.1 Εισαγωγή κεφαλαίου 1

Η ιστορία της Γεωργικής Τεχνολογίας ξεκινάει το 4000 Π.χ με την ανακάλυψη του αρότρου στην Αίγυπτο και την Μεσοποταμία και περιλαμβάνει όλα εκείνα τα τεχνολογικά μέσα που χρησιμοποιεί ο άνθρωπος για την καλλιέργεια της γης.

Έως το 2050, ο παγκόσμιος πληθυσμός αναμένεται να αυξηθεί από 7 σε 10 δισεκατομμύρια, πρόκειται δηλαδή για μια αύξηση της τάξης του 40%. Για να υπάρξει ανταπόκριση, θα πρέπει να βρεθούν οι κατάλληλοι τρόποι ώστε να οδηγηθεί η παγκόσμια γεωργική παραγωγή σε αύξηση της τάξης μεταξύ 70 έως 100% με παράλληλη παροχή καλής ποιότητας τροφής.

Συνεπώς, η πρόκληση για την μετάβαση του κλασικού μοντέλου που εφαρμόζεται μέχρι και σήμερα στον τομέα της γεωργίας είναι εδώ, εισάγοντας καινοτόμες λύσεις που μπορεί να μας προσφέρει η σημερινή τεχνολογία με σκοπό να πετύχουμε μια πιο αποδοτική παραγωγή με υγιεινά και βιώσιμα προϊόντα.

1.2 Παράγοντες που επηρεάζουν τη Γεωργία

Η ανάπτυξη των καλλιεργειών και οι αποδόσεις είναι συνάρτηση πολλών παραγόντων. Οι σημαντικότεροι από αυτούς τους παράγοντες διακρίνονται σε γενετικούς, εδαφολογικούς και κλιματολογικούς [1.1].

Οι γενετικοί παράγοντες για μια καλλιέργεια είναι συγκεκριμένοι και καθορίζουν τη δυναμικότητα της ως προς το μέγιστο της αύξησης της, υπό ευνοϊκές συνθήκες. Ο άνθρωπος παρεμβαίνει στους παράγοντες αυτούς χρησιμοποιώντας τις μεθόδους της γενετικής μηχανικής με σκοπό την δημιουργία πιο ανθεκτικών και βιώσιμων προϊόντων.

Οι εδαφολογικοί παράγοντες έχουν να κάνουν με την υφή, την δομή, τα διαθέσιμα θρεπτικά στοιχεία και τους διάφορους τύπους φυτοφάγων οργανισμών και μικροοργανισμών του εδάφους που επηρεάζουν θετικά ή αρνητικά τη φυτική παραγωγή. Μέσω κατάλληλων εδαφολογικών αναλύσεων ο παραγωγός μπορεί να προλάβει τη μείωση της γονιμότητας του εδάφους, εντοπίζοντας όλους τους παράγοντες εκείνους που επηρεάζουν θετικά ή αρνητικά τη φυτική παραγωγή.

Οι κλιματολογικοί παράγοντες σχετίζονται με την θερμοκρασία του αέρα, το νερό, την διάρκεια της ημέρας, την δύναμη του ανέμου και την ύπαρξη ορισμένων αέριων (CO₂, O₂, SO₂, NO_x, F_l, Cl, O₃) .

Η θερμοκρασία και το φως, ασκούν ιδιαίτερη επίδραση στην ανάπτυξη των καλλιεργειών καθώς μια ευνοϊκή θερμοκρασία επιταχύνει αύξηση των φυτών ενώ το πολύ έντονο φως την επιβραδύνει.

Η υγρασία αποτελεί μια παράμετρο η οποία μπορεί εύκολα να ελεγχθεί και να γίνουν οι κατάλληλες παρεμβάσεις έτσι ώστε να διατηρείται στα επιθυμητά πλαίσια. Η έλλειψη της υγρασίας, συνήθως προκαλεί γρήγορη ωρίμανση των φυτών διότι τα εξαναγκάζει να επισπεύσουν το βιολογικό τους κύκλο με αποτέλεσμα να υπάρχει σημαντικότερο πρόβλημα στην καλλιέργεια.

Τα αέρια είναι εξίσου σημαντικά και θα πρέπει να ελέγχεται η συγκέντρωσή τους έτσι ώστε να λαμβάνονται τα κατάλληλα μέτρα. Τα πρώτα τρία (CO_2 , O_2 , SO_2) δεν βλάπτουν τις καλλιέργειες ενώ η αύξηση της συγκέντρωσης των υπολοίπων (NO_x , FI , Cl , O_3) που θεωρούνται ρυπαντές, μπορεί να οδηγήσει σε αναστολή της ανάπτυξής τους.

1.3 Γεωργία ακριβείας

Η γεωργία ακριβείας [1.2] (Precision Agriculture) πρόκειται για ένα σύστημα διαχείρισης αγροτεμαχίων μέσω του οποίου αξιοποιείται η εδαφική ετερογένεια του αγρού έτσι ώστε να μεγιστοποιηθεί η αποτελεσματικότητα των καλλιεργητικών εισροών. Στοχεύει στην αύξηση της παραγωγής αλλά και στην βελτίωση της ποιότητας παραγωγής.

Αποτελεί μια μέθοδο που έχει ξεκινήσει να εφαρμόζεται τα τελευταία χρόνια, η οποία χρησιμοποιεί πληροφορία προσδιορισμένη με σαφήνεια ως προς το χώρο και το χρόνο, προκειμένου να μεγιστοποιήσει την απόδοση της παραγωγής και να ελαχιστοποιήσει τις περιβαλλοντικές επιπτώσεις.

Βασίζεται σε σύγχρονες τεχνολογίες ικανές να καταγράψουν με ακρίβεια την υπάρχουσα κατάσταση σε μια καλλιέργεια, όπου κατόπιν επεξεργασίας αυτών των πληροφοριών πρόκειται να εφαρμοστούν κατάλληλες ενέργειες έτσι ώστε, να καλύψουν τις ανάγκες κάθε σημείου για κάθε χρονική στιγμή ξεχωριστά.

Σε αντίθεση με τις συμβατικές μεθόδους αγροτικής παραγωγής, όπου οι ενέργειες παρέχονται ενιαία σε όλη την έκταση της καλλιέργειας, θεωρώντας ότι υπάρχει μια ομοιογένεια στις εδαφολογικές ιδιότητες και τη γονιμότητα του εδάφους, την εδαφική υγρασία, τους πληθυσμούς των ζιζανίων και εντόμων, τα χαρακτηριστικά των φυτών, η γεωργία ακριβείας διαχειρίζεται την συνολική έκταση σε μικρότερες περιοχές (διαχειριστικές ζώνες) που εμφανίζουν μια σχετική ομοιομορφία με αποτέλεσμα να ανταποκρίνεται περισσότερο στην πραγματικότητα.

Έτσι, με τη γεωργία ακριβείας, γίνεται αποτελεσματικότερη διαχείριση της τοπικής παραλλακτικότητας μιας καλλιεργήσιμης έκτασης αλλά και παροχής των ακόλουθων πλεονεκτημάτων:

- Βελτιστοποίηση των αποτελεσμάτων (ποσοτικά και ποιοτικά) του γεωργικού τομέα κάνοντας παράλληλα τη μέγιστη δυνατή οικονομία στους πόρους όπως είναι το νερό, τα απαιτούμενα φυτοφάρμακα-λιπάσματα καθώς επίσης και η συνολική ενέργεια (ηλεκτρική, καύσιμα, κτλ).
- Παροχή δεδομένων από διαφορετικά σημεία της αγροκαλλιέργειας με απόλυτη ακρίβεια έτσι ώστε για μεγάλες εκτάσεις να υπάρχει διαφοροποίηση των απαιτούμενων ενεργειών από σημείο σε σημείο. Αυτό παρέχει την δυνατότητα σποράς και φύτευσης διαφορετικών ποικιλιών στην ίδια καλλιέργεια με διαφορετικές ιδιότητες (Π.χ υγρασία χώματος, pH εδάφους κτλ) για την κάθε μία ποικιλία ξεχωριστά
- Προστασία της αγροκαλλιέργειας από αντίξοες καιρικές συνθήκες. Ο εκάστοτε χρήστης-αγρότης μπορεί μέσω ηλεκτρονικής διαχείρισης να καθορίζει το πότε θα γίνουν ενέργειες μιας συγκεκριμένης δράση στο πεδίο της αγροκαλλιέργειας. Έτσι θα είναι δυνατόν Π.χ να ενεργοποιηθούν θερμαντικές λάμπες σε περίπτωση παγετού ή να τεθεί σε εφαρμογή το σύστημα υδρονέφωσης όταν η υγρασία είναι χαμηλή και η θερμοκρασία αρκετά υψηλή ή να ξεκινήσει αυτόματο πότισμα όταν η υγρασία του εδάφους γίνει κατώτερη ενός συγκεκριμένου ορίου, κτλ.
- Δυνατότητα προγραμματισμένης παροχής διαφόρων φυτοφαρμάκων-λιπασμάτων στο πεδίο της αγροκαλλιέργειας μέσω ακριβής δοσολογίας έτσι ώστε αποφεύγεται η ρύπανση του εδάφους και κατά συνέπεια του περιβάλλοντος από περιπτώσεις υπερδοσολογίας.
- Μείωση του ανθρώπινου δυναμικού αλλά και του χρόνου ενασχόλησης με το πεδίο της αγροκαλλιέργειας, κατ' επέκταση την αύξηση του κέρδους του εκάστοτε παραγωγού.
- Παροχή άμεσης ενημέρωσης του καλλιεργητή μέσω οπτικοποιημένων δεδομένων σε 24ωρη βάση έτσι ώστε να είναι εύκολη και γρήγορη ή λήψη αποφάσεων χωρίς να απαιτείται εξειδικευμένη γνώση για την κατανόηση των αποτελεσμάτων.

1.4 Τεχνολογία και ΙοΤ στη γεωργία

Η καινοτομία και η τεχνολογία έχουν προχωρήσει αρκετά τα τελευταία χρόνια στον τομέα της αγροτικής παραγωγής, κυρίως μέσω μεγάλων επιχειρήσεων. Θα μπορούσαμε να πούμε ότι οι τεχνολογικές λύσεις που εφαρμόζονται σήμερα στον αγροτικό τομέα έχουν ήδη συμβάλει στην αύξηση της παραγωγικότητας της γεωργίας και στη μείωση του περιβαλλοντικού αποτυπώματος.

Η επανάσταση του διαδικτύου των πραγμάτων ΙοΤ που έχει απώτερο σκοπό να θέσει το τεχνολογικό πλαίσιο της επικοινωνίας μεταξύ αντικειμένων καθώς και η ρομποτική έρχονται να αλλάξουν εντελώς τον τρόπο λειτουργίας στην γεωργία μέσω συμβατικών εργασιών που θα

βοηθήσουν στην κάλυψη των αυξανόμενων απαιτήσεων και τη μείωση των απωλειών παραγωγής.

Σήμερα χρησιμοποιούνται ρομπότ, drones, απομακρυσμένοι αισθητήρες, υπολογιστικά συστήματα για την απεικόνιση των δεδομένων και αναλυτικά εργαλεία για την τοπογραφία και τη χαρτογράφηση των αγρών με σκοπό την παροχή δεδομένων μέσω ορθολογικών σχεδίων διαχείρισης καλλιεργειών για την εξοικονόμηση χρόνου και χρήματος .

Οι βασικές τεχνολογίες που λειτουργούν ως καταλύτης στον τομέα της γεωργίας είναι οι ακόλουθες [1.3] :

Ρομποτική / Drones:

Η χρήση των ρομπότ - drones κερδίζει μέρα με την μέρα ολόένα και περισσότερο έδαφος, αφού τα μεγάλα οφέλη που προσφέρουν συνθέτουν τις πιο σύγχρονες μεθόδους γεωργικής παραγωγής, μεγιστοποιώντας την απόδοση της καλλιεργήσιμης έκτασης. Οι εξελίξεις στην πλοήγηση και την αναγνώριση, επιτρέπουν τη χρήση των ρομπότ για σχετικά πολύπλοκες εργασίες στον τομέα της αγροκαλλιέργειας, όπως η ανίχνευση ασθενειών, η ακριβή κατανομή λιπάσματος, οι αεροψεκασμοί, η καταμέτρηση των φυτών και η επαρκή άρδευση αυτών.

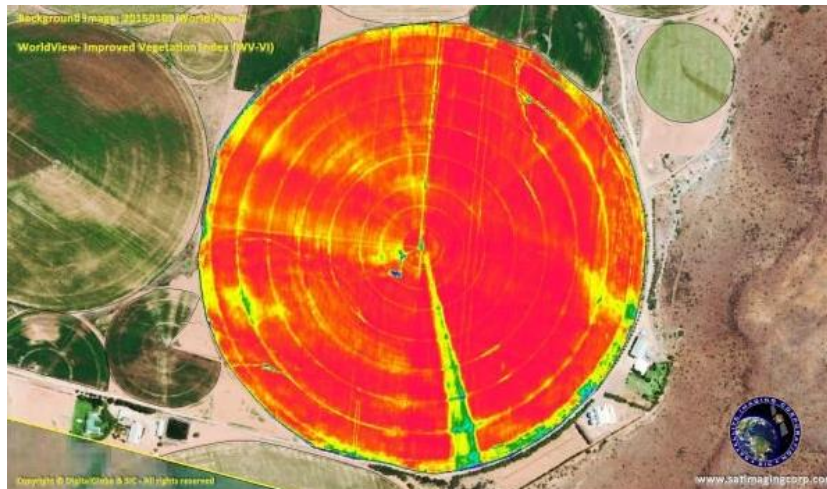


Εικόνα 1.1 – Χρήση ρομποτικών διατάξεων στο πεδίο των αγροκαλλιεργειών
(Πηγή: www.sweeper-robot.eu)

Λογισμικό GIS και χρήση GPS

Το λογισμικό GIS επιτρέπει στους αγρότες να χαρτογραφούν τη θερμοκρασία, τις βροχοπτώσεις, τις αποδόσεις των καλλιεργειών και πολλά άλλα χαρακτηριστικά όπως φαίνεται στην εικόνα 1.3. Αυτά τα δεδομένα επιτρέπουν μελλοντικές προβλέψεις με βάση τις προηγούμενες συνθήκες. Σε συνδυασμό με τη χρήση GPS, αυτές οι τεχνολογίες μπορούν να καθοδηγήσουν αυτόνομα τρακτέρ και μηχανές θερισμού ώστε να πραγματοποιήσουν προσχεδιασμένες εργασίες ακολουθώντας μια συγκεκριμένη στρατηγική. Το λογισμικό GIS

βρίσκεται επίσης πίσω από τη χρήση drones σε γεωργικό περιβάλλον, επιτρέποντας την εναέρια εκτίμηση της βιομάζας και του ύψους των καλλιεργειών, της παρουσίας ζιζανίων του εδάφους και του καιρού.

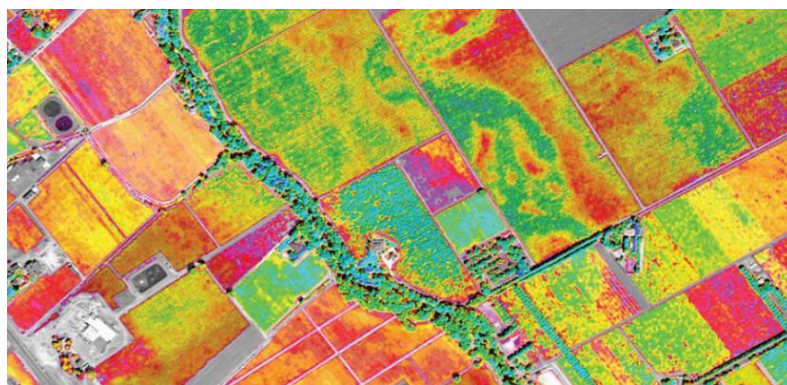


Εικόνα 1.2 – Χαρτογράφηση αγροτικής περιοχής μέσω GIS

(Πηγή: <https://www.satimagingcorp.com/services/geographic-information-systems/gis-maps-agriculture-mapping/>)

Δορυφορική Απεικόνιση

Οι δορυφόροι χρησιμοποιούνται για πολλούς σκοπούς αλλά τα τελευταία χρόνια χρησιμοποιούνται επίσης και στον γεωργικό τομέα. Οι σύγχρονες δορυφορικές εικόνες όπως φαίνεται από την εικόνα 1.3, επιτρέπουν στους αγρότες να ενσωματώνουν δεδομένα από σχετικούς φασματικούς δείκτες προκειμένου να εφαρμόσουν ένα μοντέλο γεωργίας ακριβείας. Ορισμένοι από τους δείκτες αναλύουν την ποσότητα χλωροφύλλης στις καλλιέργειες η οποία σηματοδοτήσει την έλλειψη άζωτου. Άλλοι δείκτες ανιχνεύουν την βλάστηση σε πρόσφατα σπαρμένες περιοχές ή περιοχές με μεγάλες ποσότητες γυμνού εδάφους.



Εικόνα 1.3 – Δορυφορική εικόνα

(Πηγή: <https://www.xyht.com/enviroag/satellite-imagery-precision-agriculture/>)

Αισθητήρες

Παρέχουν συστηματική μέτρηση διαφόρων ιδιοτήτων από το πεδίο της αγροκαλλιέργειας όπως η θερμοκρασία, τα επίπεδα υγρασίας, η ένταση του φωτός, η ταχύτητα του ανέμου κτλ. Η κάλυψη και η πολυπλοκότητα των αισθητήρων για γεωργικές εφαρμογές αυξάνονται συνεχώς, με ταυτόχρονη μείωση του κόστους του εξοπλισμού, εγκατάστασης αλλά και συντήρησης.

Cloud computing

Η μείωση του κόστους πρόσβασης αλλά και η αύξηση των δυνατοτήτων που παρέχονται στις μέρες μας στον τομέα της πληροφορικής επιτρέπει την χρήση σύγχρονου εξοπλισμού, απαραίτητο για την συλλογή, αποθήκευση και ανάλυση δεδομένων χωρίς την ύπαρξη τοπικού εξοπλισμού.

Blockchain

Η γεωργία πέρα από την παραγωγή τροφίμων παράγει και δεδομένα τα οποία πιστοποιούν διαφορές ιδιότητες των τροφίμων όπως το είδος, η προέλευση κτλ τα οποία είναι οικονομικά πολύτιμα. Ωστόσο, ενώ η αποθήκευση των δεδομένων είναι φθηνή, η επαλήθευσή τους είναι αρκετά δαπανηρή.

Κατά συνέπεια, ένα σημαντικό ποσοστό του τελικού κόστους της γεωργικής παραγωγής πηγαίνει στο κόστος καθορισμού προέλευσης, απόδειξης συμμόρφωσης με τα πρότυπα και κανονισμούς, ελέγχων και παρακολούθησης της διαδικασίας, καθώς και στο κόστος διαμεσολάβησης, διασφάλισης ποιότητας και επωνυμίας.

Το Blockchain[1.4] στον τομέα της γεωργίας, είναι μια νέα τεχνολογία που δίνει τη δυνατότητα σε διαφορετικά μέρη σε μια αλυσίδα εφοδιασμού να εμπιστεύονται τα ψηφιακά δεδομένα. Συνεπώς, έχει την δυνατότητα να μειώσει το κόστος συναλλαγών και να βελτιώσει την αποτελεσματικότητα των γεωργικών αλυσίδων εφοδιασμού μειώνοντας την ανάγκη παρακολούθησης και επαλήθευσης δεδομένων.

2. IoT και Ασύρματα Δίκτυα Αισθητήρων (WSN)

2.1 Εισαγωγή

Ο όρος IoT πραγματεύεται την διασύνδεση μεταξύ συσκευών με στόχο την αμφίδρομη μεταφορά δεδομένων. Η αλματώδης ανάπτυξη της τεχνολογίας στις μέρες μας κάνει αναγκαία την ύπαρξη πρακτικών εφαρμογών που κάνουν χρήση διαφόρων τεχνολογιών IoT σε πολλούς τομείς όπως η γεωργία, η βιομηχανία, η έξυπνοι μετρητές, η ασφάλεια, οι έξυπνες πόλεις, τα έξυπνα σπίτια κτλ.

Ο τομέας του IoT και γενικότερα οι εφαρμογές του έχουν συγκεκριμένες απαιτήσεις όπως, η χαμηλή κατανάλωση ενέργειας, μεγάλη εμβέλεια ασύρματης επικοινωνίας, χαμηλό ρυθμό μετάδοσης δεδομένων και χαμηλό κόστος εγκατάστασης και λειτουργίας. Πολλές τεχνολογικές λύσεις της προηγούμενης δεκαετίας δεν κάλυπταν αυτές τις απαιτήσεις και έτσι, σιγά-σιγά δημιουργήθηκε μια νέα τεχνολογία ασύρματης επικοινωνίας, το δίκτυο ευρείας περιοχής χαμηλής ισχύος (LPWAN).

2.2 Αναφορά στις τεχνολογίες ασύρματης δικτύωσης LPWAN

2.2.1 Τι είναι η τεχνολογία LPWAN – Low Power Wide Area Network

Η τεχνολογία ευρείας περιοχής χαμηλής κατανάλωσης (LPWAN - Low Power Wide Area Network) εμφανίστηκε αρχικά ως όρος το 2013, όχι ως νέο τεχνολογικό πρότυπο, αλλά ως μια κατηγορία ασύρματων τεχνολογιών για συγκεκριμένες ανάγκες επικοινωνίας μεταξύ μηχανών M2M (Machine To Machine) και κατ' επέκταση σχετική με τον τομέα του IoT.

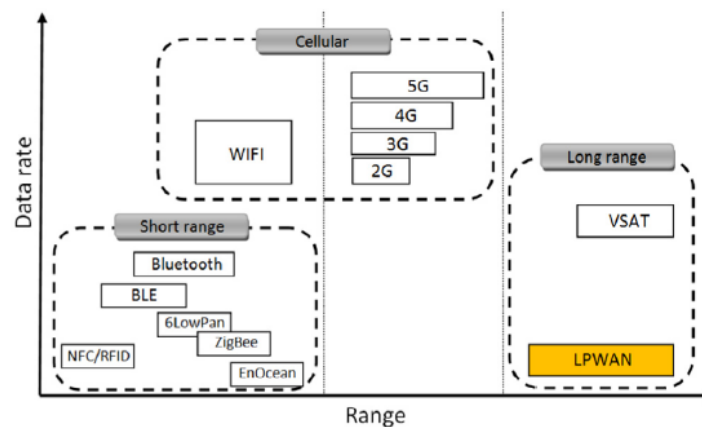
Στην πορεία απέκτησε εξέχουσα θέση ως η προτιμώμενη επιλογή για εφαρμογές IoT το 2015, όταν η ένωση βιομηχανίας ασύρματων δικτύων GSMA καθόρισε μια σειρά προτύπων δικτύου για να βοηθήσει τους φορείς εκμετάλλευσης δικτύων να ανταποκριθούν στις απαιτήσεις του IoT.

Τα γενικά χαρακτηριστικά της τεχνολογίας LPWAN είναι η παροχή χαμηλής ισχύος εκπομπής καθότι οι πομποδέκτες θα πρέπει να λειτουργούν με μικρές μπαταρίες για μεγάλο χρονικό διάστημα, μεγάλη εμβέλεια κάλυψης (10-40Km σε αγροτικές ζώνες και 1-5km σε αστικές ζώνες) και τέλος μειωμένη πολυπλοκότητα στον σχεδιασμό έτσι ώστε να είναι χαμηλό το κόστος υλοποίησης.

Συνοπτικά, η τεχνολογία LPWAN είναι κατάλληλη για εφαρμογές τηλεμετρίας IoT όπου μικρές ποσότητες δεδομένων μεταδίδονται περιοδικά σε μεγάλη εμβέλεια. Όμως, οι ανάγκες ποικίλουν και δεν είναι πάντα οι ίδιες για όλες τις εφαρμογές του IoT. Έτσι, υπάρχουν διάφορες

LPWAN τεχνολογίες όπου η κάθε μια από αυτές κάνει τους δικούς της συμβιβασμούς μεταξύ του ρυθμού δεδομένων, της εμβέλειας, της κατανάλωσης ενέργειας, της ασφάλειας κ.λπ.

Στο ακόλουθη εικόνα 2.1 εμφανίζεται η τεχνολογία LPWAN σχετικά με τον ρυθμό δεδομένων και την εμβέλεια σε σχέση με τις υπόλοιπες υπάρχουσες τεχνολογίες.



Εικόνα 2.1 – Συγκριτικό διάγραμμα ασύρματων τεχνολογιών
(Πηγή: <https://www.sciencedirect.com/science/article/pii/S2405959517302953>)

Ορισμένες τεχνολογίες που αναπτυχθήκαν πάνω στην LPWAN είναι οι ακόλουθες:

SigFox, ZigBee, Nwave, LTE-M, NB-IoT και LoRa.

Παρακάτω γίνεται μια συνοπτική αναφορά στην τεχνολογία NB-IoT και Sigfox και αναλύεται η τεχνολογία LoRa και το πρωτόκολλο LoRaWAN που χρησιμοποιήθηκαν στο πλαίσιο αυτής της εργασίας.

2.2.2 LTE-M (Long Term Evolution – Machine Type Communication) & NB-IoT (Narrow Band – Internet of Things)

Το LTE-M είναι ο απλοποιημένος βιομηχανικός όρος για το LTE-MTC, ένα πρότυπο LPWAN που έχει εκδοθεί από τον οργανισμό προτύπων 3GPP (3rd Generation Partnership Project). Συχνά θεωρείται μία από τις τεχνολογίες LPWAN κινητής τηλεφωνίας πρώτης γενιάς. Το LTE-M αξιοποιεί τα υπάρχοντα δίκτυα LTE επιτρέποντας εξαιρετικά αποδοτική συνδεσιμότητα με εκτεταμένη κάλυψη σε εσωτερικούς και υπόγειους χώρους. Μειώνει σημαντικά την πολυπλοκότητα της συσκευής επιτρέποντας αποτελεσματική συνδεσιμότητα σε πολλές ζώνες GSM, εξαλείφοντας έτσι την ανάγκη για πολλαπλές μονάδες και παραλλαγές.

Η λύση LTE-M είναι ενεργειακά αποδοτική καθώς επιτρέπει τη διάρκεια ζωής της μπαταρίας για μεγάλο χρονικό διάστημα, εκτεταμένη κάλυψη, βελτιωμένη διείσδυση εντός κτιρίων καθώς επίσης και σε σημεία κάτω από το έδαφος και τέλος βελτιστοποιημένες

ταχύτητες σε σύγκριση με άλλες τεχνολογίες, έως 200 Kbps καθοδικής ζεύξης και 375 Kbps ανοδικής ζεύξης.

Η εξέλιξη του LTE-M έρχεται με το NB-IoT το οποίο ανεβάζει την αποτελεσματικότητα του δικτύου που το χρησιμοποιεί, αυξάνοντας την ικανότητα υποστήριξης ενός τεράστιου αριθμού νέων συνδέσεων χρησιμοποιώντας μόνο ένα μέρος του διαθέσιμου φάσματος. Γενικά, το NB-IoT επιτρέπει την συνδεσιμότητα έως και 100K τελικών συσκευών ανά κυψέλη με δυνατότητα κλιμάκωσης της χωρητικότητας με την προσθήκη περισσότερων φορέων NB-IoT.

Το NB-IoT χρησιμοποιεί την πολλαπλή πρόσβαση διαίρεσης συχνότητας ενός φορέα (FDMA - Frequency Division Multiple Access) στην ανερχόμενη ζεύξη και την ορθογώνια FDMA (OFDMA - Orthogonal Frequency-Division Multiple Access) στην κατερχόμενη ζεύξη κάνοντας χρήση της διαμόρφωσης μετατόπισης φάσης τετραγωνισμού (QPSK - Quadrature Phase Shift Keying).

Τα γενικά χαρακτηριστικά του είναι παρόμοια με αυτά της LTE-M τεχνολογίας. Δηλαδή:

- Έχει μικρή κατανάλωση που επιτρέπει διάρκεια ζωής της μπαταρίας για μεγάλο χρονικό διάστημα. Μπορεί να επιτύχει διάρκεια ζωής της μπαταρίας έως δέκα χρόνια όταν μεταδίδει 200 byte την ημέρα κατά μέσο όρο.
- Ταχύτητες κάλυψης που μπορούν να φτάσουν έως 200 Kbps για την κατερχόμενη ζεύξη και 20 Kbps για την ανερχόμενη ζεύξη.
- Μέγιστο μέγεθος ωφέλιμου φορτίου του κάθε μηνύματος είναι 1600 byte.
- Πολύ καλή διείσδυση σήματος σε βαθιά υπόγεια και σε κλειστούς χώρους.

Και οι δύο αυτές παραπάνω τεχνολογίες (LTE-M & NB-IoT) έχουν αρκετά πλεονεκτήματα όπως ήδη έχει αναφερθεί. Μειονεκτούν όμως καθότι κάνουν χρήση του δικτύου της κινητής τηλεφωνίας. Και αυτό διότι αφενός υπάρχουν ακόμη και σήμερα περιοχές όπου δεν καλύπτονται επαρκώς ή και καθόλου από κάποιον πάροχο κινητής τηλεφωνίας και αφετέρου από την στιγμή που γίνεται χρήση της κινητής τηλεφωνίας, υπάρχει ένα κόστος που είναι αναπόφευκτο. Το κόστος αυτό μπορεί να είναι μικρό σε ένα δίκτυο ορισμένων δεκάδων τελικών συσκευών όμως μπορεί να γίνει αρκετά μεγάλο στην περίπτωση όπου το δίκτυο αποτελείται από χιλιάδες τελικές συσκευές.

2.2.3 Sigfox

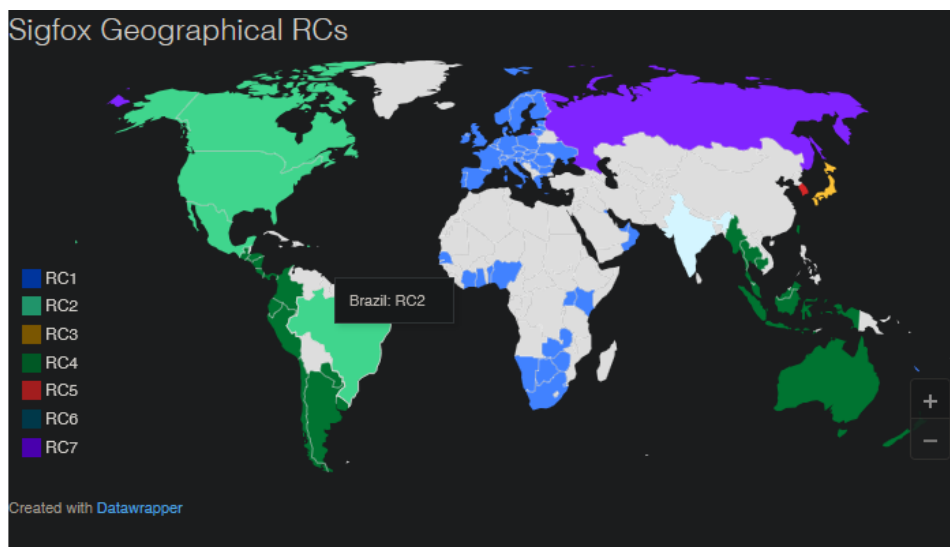
Η Sigfox [1] είναι μια γαλλική εταιρεία που εδρεύει στο Labège κοντά στην Τουλούζη της Γαλλίας και έχει πάνω από 375 υπαλλήλους. Ιδρύθηκε το 2010 από την start-up Sigfox με σκοπό την κατασκευή και δημιουργία ασύρματου δικτυακού εξοπλισμού και λογισμικού για τη σύνδεση αντικειμένων χαμηλής κατανάλωσης όπως είναι οι έξυπνοι μετρητές και

αισθητήρες οι οποίοι πρέπει να είναι συνεχώς ενεργοποιημένοι και να εκπέμπουν μικρές ποσότητες δεδομένων ανά τακτά χρονικά διαστήματα.

Η ανάπτυξη έγινε με τέτοιο τρόπο ώστε να ανταποκρίνεται στις απαιτήσεις μαζικών εφαρμογών IoT. Δηλαδή, μεγάλος κύκλος ζωής μπαταρίας της συσκευής, χαμηλό κόστος συσκευής, χαμηλά λειτουργικά έξοδα με χρήση ελεύθερων συχνοτήτων για μηδενικό τέλος χρήσης, υψηλή χωρητικότητα δικτύου και τέλος μεγάλη εμβέλεια.

Διαθέτει ένα παγκόσμιο δίκτυο κάλυψης LPWAN σε 31 χώρες και εξακολουθεί να είναι υπό ανάπτυξη παγκοσμίως. Λειτουργεί σε διάφορες ζώνες σε όλο τον κόσμο, με ραδιοσυχνότητες που κυμαίνονται από 862 έως 928 MHz. Έτσι, κάνοντας χρήση διαφόρων ζωνών συχνοτήτων χωρίς άδεια επιτρέπει στις συσκευές να στέλνουν δεδομένα σε ένα υπολογιστικό νέφος (cloud) ενώ χρησιμοποιούν μικρή ισχύ, με πολύ χαμηλό κόστος.

Ωστόσο, λόγω τοπικών κανονισμών, νόμων ή άλλων λειτουργικών περιορισμών, η διαμόρφωση του δικτύου μπορεί να διαφέρει από τη μια χώρα στην άλλη.



Εικόνα 2.2 – Ζώνες συχνοτήτων (Radio configuration) Sigfox
(Πηγή: <https://build.sigfox.com/sigfox-radio-configurations-rc>)

Οι δραστηριότητες εκπομπής χωρίζονται μέχρι στιγμής σε επτά γεωγραφικές ζώνες (Radio Configuration από RC1 έως RC7) όπως φαίνεται και στην παραπάνω εικόνα 2.2. Κάθε ζώνη έχει ένα διαφορετικό σύνολο παραμέτρων που ορίζουν την υλοποίηση του υλικού της συσκευής: εύρος συχνοτήτων, μέγιστη ακτινοβολούμενη ισχύς και διάφορα άλλα χαρακτηριστικά. Βεβαίως, στο μέλλον μπορούν να προστεθούν περισσότερες ζώνες καθώς το δίκτυο αυξάνεται ολοένα και νέες χώρες υιοθετούν την συγκεκριμένη τεχνολογία.

Στον πίνακα 2.1 απεικονίζονται με ακρίβεια οι τεχνικές λεπτομέρειες ανά περιοχή.

	RC1	RC2	RC3	RC4	RC5	RC6	RC7
Uplink center frequency (MHz)	868.130	902.200	923.200	920.800	923.300	865.200	868.800
Downlink center frequency (MHz)	869.525	905.200	922.200	922.300	922.300	866.300	869.100
Uplink data rate (bit/s)	100	600	100	600	100	100	100
Downlink data rate (bit/s)	600	600	600	600	600	600	600
Sigfox recommended EIRP (dBm)	16	24	16	24	14	16	16
Specifics	DC 1% *	FH **	LBT and/or DC 1%	FH **	LBT ***	DC 1% *	DC 1% *

Πίνακας 2.1 – Τεχνικές λεπτομέρειες τεχνολογίας Sigfox
(Πηγή: <https://build.sigfox.com/sigfox-radio-configurations-rc>)

EIRP (Effective Isotropic Radiated Power): Το EIRP είναι ένας υπολογισμός που χρησιμοποιείται για την εκτίμηση της ακτινοβολούμενης ισχύος εξόδου μιας ιστροπικής κεραίας ο οποίος λαμβάνει υπόψη την ισχύ εξόδου του πομπού, την απώλεια καλωδίου και το κέρδος κεραίας. (Με τον όρο ιστροπική κεραία θεωρείται μια διπολική κεραία μισού κύματος που ακτινοβολεί τέλεια προς όλες τις κατευθύνσεις)

DC (Duty cycle): Το DC είναι 1% του χρόνου ανά ώρα (36 δευτερόλεπτα). Για ωφέλιμο φορτίο 8 έως 12 byte, αυτό σημαίνει 6 μηνύματα την ώρα, άρα 140 την ημέρα.

FH (Frequency hopping): Η συσκευή εκπέμπει κάθε μήνυμα 3 φορές σε 3 διαφορετικές συχνότητες. Μέγιστος χρόνος ενεργοποίησης 400 ms ανά κανάλι. Καμία νέα εκπομπή πριν από 20 δευτερόλεπτα.

LBT (Listen Before Talk): Οι συσκευές πρέπει να επαληθεύσουν ότι το φάσμα συχνοτήτων των 200 kHz που ορίζεται από το πρωτόκολλο της Sigfox είναι απαλλαγμένο από σήμα ισχυρότερο από -80 dBm για την γεωγραφική ζώνη RC3 και -65 dBm αντίστοιχα για την RC5 πριν από τη μετάδοση.

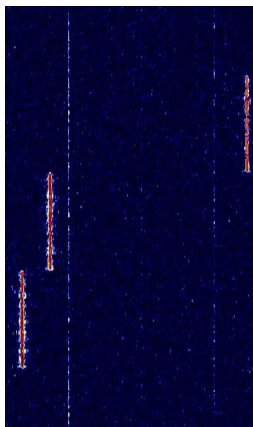
Στην ακόλουθη εικόνα φαίνεται η τοπολογία ενός δικτύου Sigfox.



Εικόνα 2.3 – Τοπολογία δικτύου Sigfox
(Πηγή: <https://build.sigfox.com/sigfox-radio-configurations-rc>)

Η Sigfox αναπτύσσει ιδιόκτητους σταθμούς βάσης εξοπλισμένους με συσκευές εκπομπής και δικό της λογισμικό και τους συνδέει με τους διακομιστές υποστήριξης χρησιμοποιώντας ένα δίκτυο που βασίζεται σε τεχνολογία IP. Οι τελικές συσκευές συνδέονται με αυτούς τους σταθμούς βάσης χρησιμοποιώντας διαμόρφωση δυαδικής μετατόπισης φάσης (BPSK) σε φορέα ζώνης ISM εξαιρετικά στενής ζώνης (100 Hz). Οι ζώνες δεν απαιτούν άδεια, για παράδειγμα, 868 MHz στην Ευρώπη, 915 MHz στη Βόρεια Αμερική και 433 MHz στην Ασία. Χρησιμοποιώντας την εξαιρετικά στενή ζώνη των 100Hz, η Sigfox χρησιμοποιεί αποτελεσματικά το εύρος ζώνης συχνοτήτων και έχει πολύ χαμηλά επίπεδα θορύβου, οδηγώντας σε πολύ χαμηλή κατανάλωση ενέργειας, υψηλή ευαισθησία δέκτη και χαμηλό link budget σε βάρος της μέγιστης απόδοσης μόνο 100 bps. [2]

Το Sigfox αρχικά υποστήριζε μόνο επικοινωνία ανοδικής ζεύξης, αλλά αργότερα εξελίχθηκε σε αμφίδρομη τεχνολογία με σημαντική ασυμμετρία ζεύξης. Η κατερχόμενη επικοινωνία ζεύξης, δηλαδή δεδομένα από τους σταθμούς βάσης προς τις τελικές συσκευές, μπορεί να πραγματοποιηθεί μόνο μετά από επικοινωνία ανερχόμενης ζεύξης. Ο αριθμός των μηνυμάτων μέσω του uplink περιορίζεται στα 140 μηνύματα την ημέρα. Το μέγιστο μήκος ωφέλιμου φορτίου για κάθε μήνυμα uplink είναι 12 byte με ρυθμό μετάδοσης δεδομένων έως και 100 bit ανά δευτερόλεπτο. Ωστόσο, ο αριθμός των μηνυμάτων μέσω του κατερχόμενου συνδέσμου περιορίζεται σε τέσσερα μηνύματα την ημέρα, πράγμα που σημαίνει ότι δεν υποστηρίζεται η επιβεβαίωση κάθε μηνύματος ανόδου. Το μέγιστο μήκος ωφέλιμου φορτίου για κάθε μήνυμα downlink είναι οκτώ byte. Χωρίς την επαρκή υποστήριξη των επιβεβαιώσεων, η αξιοπιστία της επικοινωνίας ανοδικής ζεύξης διασφαλίζεται με τη χρήση διαφοροποίησης χρόνου και συχνότητας καθώς και διπλασιασμού της μετάδοσης. Κάθε μήνυμα τελικής συσκευής μεταδίδεται πολλές φορές (τρεις από προεπιλογή) μέσω διαφορετικών καναλιών συχνότητας (frequency hopping) όπως φαίνεται από την εικόνα 2.4 .



Εικόνα 2.4 - Ανοδικό σήμα sigfox με εμφανές το frequency hopping
(Πηγή: <https://www.sigfoxwiki.com/wiki/SIGFOX>)

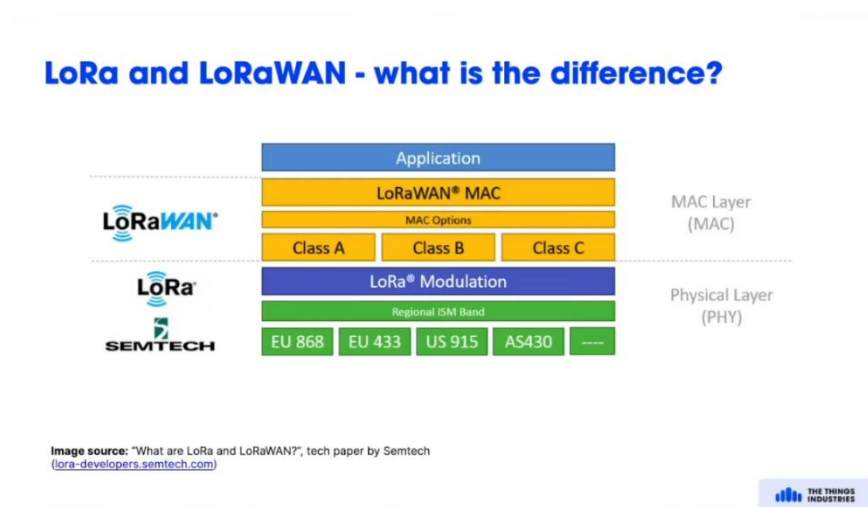
Για το σκοπό αυτό, στην Ευρώπη για παράδειγμα, η ζώνη μεταξύ 868.180 MHz και 868.220 MHz χωρίζεται σε 400 κανάλια 100 Hz (μεταξύ αυτών 40 κανάλια είναι

δεσμευμένα και δεν χρησιμοποιούνται). Καθώς οι σταθμοί βάσης μπορούν να λαμβάνουν μηνύματα ταυτόχρονα σε όλα τα κανάλια, η τελική συσκευή μπορεί να επιλέξει τυχαία ένα κανάλι συχνότητας για τη μετάδοση των μηνυμάτων τους. Αυτό απλοποιεί τη σχεδίαση της τελικής συσκευής και μειώνει το κόστος της.

Η Sigfox έχει συνεργαστεί με έναν αριθμό εταιρειών στη βιομηχανία LPWAN, όπως η Texas Instruments, η Silicon Labs και η ON Semiconductor. Οι ραδιοφωνικές ζώνες ISM υποστηρίζουν περιορισμένη αμφίδρομη επικοινωνία. Το υπάρχον πρότυπο για τις επικοινωνίες Sigfox υποστηρίζει έως και 140 μηνύματα uplink την ημέρα, καθένα από τα οποία μπορεί να μεταφέρει ωφέλιμο φορτίο 12 bytes με ρυθμό μετάδοσης δεδομένων έως και 100 bit ανά δευτερόλεπτο.

2.2.4 Ορισμός της Τεχνολογίας LoRa

Το όνομα της τεχνολογίας LoRa προέρχεται από το “Long Range” που σημαίνει «μακράς εμβέλειας» και αφορά μια τεχνική διαμόρφωσης κατοχυρωμένη με δίπλωμα ευρεσιτεχνίας. Αρχικά αναπτύχθηκε από την Cycleo στην Γκρενόμπλ της Γαλλίας και αργότερα αποκτήθηκε από τη εταιρεία Semtech η οποία συνεχίζει ακόμη και σήμερα να εξελίσσει την συγκεκριμένη τεχνολογία έχοντας δημιουργήσει μια συμμαχία εταιριών (LoRa Alliance) με σκοπό την υποστήριξη και την ανάπτυξη της συγκεκριμένης τεχνολογίας σε διάφορους τομείς.



Εικόνα 2.5 – LoRa & LoRaWAN

(Πηγή: <https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/>)

Ο όρος LoRa καθορίζει το φυσικό επίπεδο (PHY) για την τεχνολογία LoRa (εικόνα 2.5). Με βάση το μοντέλο OSI που ορίζεται από επτά διαφορετικά επίπεδα το φυσικό επίπεδο είναι το πρώτο και χαμηλότερο. Το επίπεδο αυτό σχετίζεται περισσότερο με τη φυσική διασύνδεση μεταξύ των συσκευών και η υλοποίηση του επιπέδου αυτού, πραγματοποιείται μέσω ενός τσιπ σιλικόνης. Με τον όρο φυσική διασύνδεση εννοείται η σύνδεση του σταθμού με το μέσο μετάδοσης. Συνεπώς, εδώ καθορίζονται όλα τα ηλεκτρικά και μηχανικά χαρακτηριστικά της

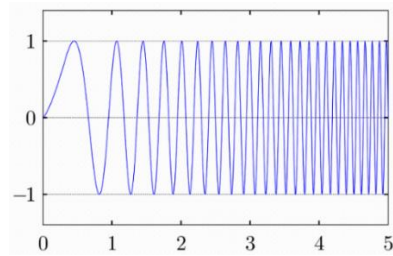
σύνδεσης έτσι ώστε να μπορεί να πραγματοποιείται η μετάδοση των δεδομένων μέσα από το τηλεπικοινωνιακό κανάλι.

Η τεχνολογία LoRa απαρτίζει σημαντικά χαρακτηριστικά που την κάνουν να υπερτερεί σε σύγκριση με άλλες παρεμφερείς τεχνολογίες. Ασφαλώς, ορισμένα χαρακτηριστικά μπορεί να βρεθούν και σε άλλες τεχνολογίες ευρείας περιοχής χαμηλής κατανάλωσης (LPWAN). Όμως ο συνδυασμός του συνόλου των ακόλουθων χαρακτηριστικών είναι εξαιρετικά ισχυρός έτσι ώστε να δημιουργήσει ένα μεγάλο ενδιαφέρον για την χρήση της τεχνολογίας LoRa. Έτσι, στις μέρες μας κατατάσσετε στην πρώτη θέση σε σχέση με άλλες παρεμφερείς τεχνολογίες δικτύωσης.

- Κύριο και βασικό χαρακτηριστικό είναι η πολύ χαμηλή κατανάλωση ισχύος. Αισθητήρες που κάνουν χρήση της τεχνολογίας LoRa μπορούν να λειτουργούν για πολύ μεγάλο χρονικό διάστημα (1-2 έτη) με μια απλή μπαταρία.
- Το σήμα μπορεί να ληφθεί σε μεγάλες αποστάσεις, δηλαδή έχει πολύ μεγάλη εμβέλεια. Άλλωστε το όνομα της συγκεκριμένης τεχνολογίας πηγάζει από το συγκεκριμένο χαρακτηριστικό (LoRa -> Long Range).
- Βαθιά διείσδυση σήματος σε εσωτερικούς χώρους.
- Η μετάδοση σήματος μέσω διαμόρφωσης LoRa είναι εξαιρετικά ανθεκτική έναντι διαταραχών από άλλα σήματα τα οποία θα μπορούσαν να δημιουργήσουν σημαντικό θόρυβο στο σήμα.
- Η τεχνολογία LoRa κάνει χρήση ελευθέρου φάσματος συχνοτήτων, συνεπώς δεν απαιτείται ειδική άδεια χρήσης και λειτουργείας για την ανάπτυξη ενός δικτύου.
- Υποστηρίζει γεωγραφικό εντοπισμό δηλαδή μπορεί να προσδιορίσει τη θέση των συσκευών αποστολής δεδομένων [4].
- Δίνει την δυνατότητα ανάπτυξης δημόσιων καθώς και ιδιωτικών δικτύων κάνοντας χρήση του ίδιου εξοπλισμού.
- Παρέχει άμεσα ενσωματωμένη ασφάλεια αποστολής δεδομένων από το ένα άκρο της επικοινωνίας έως το άλλο.
- Υποστηρίζει ασύρματες ενημερώσεις υλικολογισμικού (over-the-air updates) [5]
- Παρέχεται πρόγραμμα πιστοποίησης.
- Και τέλος υπάρχει ένα πολύ μεγάλο οικοσύστημα από κατασκευαστές και προγραμματιστές που παράγουν, αναπτύσσουν και υποστηρίζουν συσκευές που αφορούν την τεχνολογία LoRa.

Όλα τα παραπάνω χαρακτηριστικά καθιστούν την τεχνολογία LoRa μια πρωτοποριακή τεχνολογία κατάλληλη για αισθητήρες και ενεργοποιητές που λειτουργούν με χαμηλή ισχύ.

Η τεχνολογία LoRa είναι μια τεχνική ασύρματης διαμόρφωσης που χρησιμοποιεί παλμούς chirp. Οι παλμοί chirp χρησιμοποιούνται και στην φύση από τα δελφίνια και τις νυχτερίδες για την μεταξύ τους επικοινωνία. Το chirp είναι ένα ημιτονοειδές σήμα του οποίου η συχνότητα αυξάνεται ή μειώνεται με την πάροδο του χρόνου (συχνά με μια πολυωνυμική έκφραση μεταξύ χρόνου και συχνότητας). Μια γραφική απεικόνιση ενός σήματος chirp φαίνεται στην εικόνα 2.6:

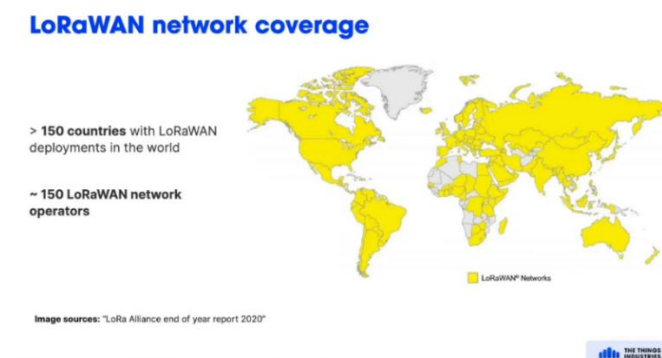


Εικόνα 2.6 – Γραφική απεικόνιση ενός Chirp σήματος
(Πηγή: https://en.wikipedia.org/wiki/Chirp_spread_spectrum)

Το LoRa είναι ιδανικό για εφαρμογές που μεταδίδουν μικρά κομμάτια δεδομένων με χαμηλούς ρυθμούς bit. Τα δεδομένα μπορούν να μεταδοθούν σε μεγαλύτερη εμβέλεια σε σύγκριση με άλλες ασύρματες τεχνολογίες όπως WiFi, Bluetooth ή ZigBee.

Το LoRa μπορεί να λειτουργήσει σε ζώνες χωρίς άδεια, π.χ. 915 MHz, 868 MHz και 433 MHz. Αυτές οι συχνότητες εμπίπτουν σε ζώνες ISM που είναι δεσμευμένες διεθνώς για βιομηχανικούς, επιστημονικούς και ιατρικούς σκοπούς.

Ορισμένες από τις εφαρμογές που εφαρμόζεται η τεχνολογία LoRa είναι η πρόληψη φυσικών καταστροφών, η γεωργία ακριβείας, διάφορες βιομηχανικές εφαρμογές όπως η παρακολούθηση παραμέτρων σε μεγάλη απόσταση, αλλά επίσης και μια πληθώρα εφαρμογών του διαδικτύου των πραγμάτων (IOT) όπως είναι η ανάπτυξη έξυπνων πόλεων, η παρακολούθηση κτιρίων, ο έλεγχος της ατμοσφαιρικής ρύπανσης, η επιτήρηση της αλυσίδας εφοδιασμού, έξυπνοι μετρητές και άλλα.



Εικόνα 2.7 – Παγκόσμια κάλυψη δικτύου LoRa
Πηγή: The Things Network

Τα δίκτυα LoRa όπως φαίνεται από την εικόνα 2.7 είναι πραγματικά μια παγκόσμια τεχνολογία. Η δυνατότητα τους να προσαρμόζουν το φάσμα συχνοτήτων τους αλλά και άλλα χαρακτηριστικά του σήματος εκπομπής τους, τα φέρνει στην πρώτη θέση επιλογής όταν πρόκειται για ασύρματη μετάδοση χαμηλού bandwidth και ιδιαίτερα για εφαρμογές που ανήκουν στο πεδίο του IoT (Internet of Things). Έτσι, υπάρχουν ανά τον κόσμο αρκετοί πάροχοι

που καλύπτουν την δικτυακή υποδομή και διαχείριση ενός δικτύου LoRaWAN με το ανάλογο κόστος. Υπάρχει βέβαια και εναλλακτική επιλογή, να αναπτυχθεί ένα ιδιωτικό οικοσύστημα διαχείρισης ή να χρησιμοποιηθεί μια λύση ελεύθερου κώδικα όπως είναι το chirpStack όπου έχει μεγαλύτερη ευελιξία διαχείρισης και μηδενικό κόστος.

2.3 Ορισμός και Αρχιτεκτονική πρωτοκόλλου LoRaWAN

Με τον όρο LoRaWAN εννοούμε το πρωτόκολλο επικοινωνίας το οποίο είναι χτισμένο με βάση την τεχνολογία LoRa. Όπως φαίνεται από την εικόνα 2.5 πρόκειται για ένα πρωτόκολλο επιπέδου ελέγχου πρόσβασης μέσω (MAC – Media Access Control), δηλαδή το αμέσως επόμενο από το φυσικό επίπεδο και αφορά λογισμικό που ορίζει τον τρόπο με τον οποίο οι συσκευές χρησιμοποιούν την τεχνολογία LoRa όπως π.Χ. κατά την μετάδοση πια θα πρέπει να είναι η μορφή των μηνυμάτων, κατά την λήψη πως θα γίνει η αποκωδικοποίηση της πληροφορίας κτλ.

Το πρωτόκολλο LoRaWAN αναπτύσσεται και διατηρείται από την μη κερδοσκοπική ένωση-οργανισμό που ονομάζεται LoRa Alliance (Συμμαχία LoRa).

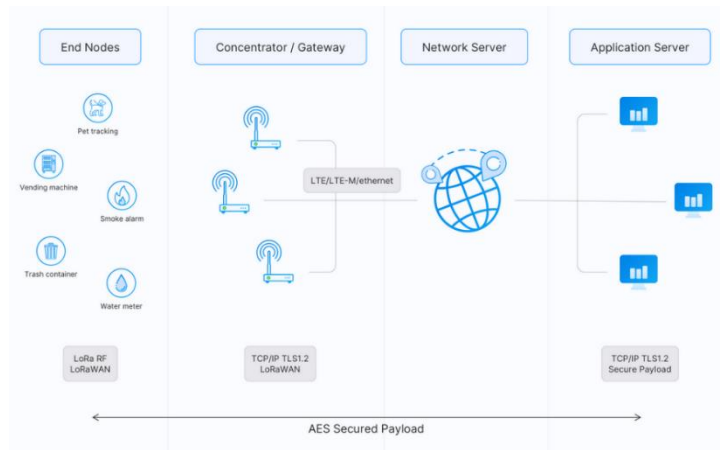
Η πρώτη προδιαγραφή LoRaWAN κυκλοφόρησε τον Ιανουάριο του 2015 με την έκδοση 1.0 (ver. 1.0) και από τότε μέχρι και σήμερα έχουν υπάρξει 4 διαφορετικές ενημερώσεις που φαίνονται στην εικόνα 2.8. Τελευταία έκδοση είναι η 1.1 αλλά ακόμη χρησιμοποιείται ευρέως η έκδοση 1.0.4 καθώς η αναβάθμιση σε έκδοση πρωτοκόλλου 1.1 ορισμένες φορές απαιτεί και αλλαγή εξοπλισμού.



Εικόνα 2.8 – Εκδόσεις του πρωτοκόλλου LoRaWAN

Πηγή: The Things Network

Η συνολική εικόνα της αρχιτεκτονικής ενός δικτύου LoRaWAN έχει την μορφή που φαίνεται στην εικόνα 2.9 .



Εικόνα 2.9 – Αρχιτεκτονική του πρωτοκόλλου LoRaWAN

Πηγή: The Things Network

Από τα αριστερά, φαίνονται οι τελικές συσκευές (End Nodes) όπου αποτελούνται από διάφορους αισθητήρες και συνδέονται ασύρματα με τις πύλες (Gateways). Οι πύλες προσδιορίζουν την τελική συσκευή και λαμβάνουν τα μηνύματα από την τελική συσκευή. Μόλις μια πύλη λάβει κάποιο μήνυμα το προωθεί σε έναν διακομιστή δικτύου (Network Server). Ο διακομιστής δικτύου ο οποίος είναι εγκατεστημένος και λειτουργεί σε έναν εξυπηρετητή (Server), αρχικά ενεργοποιεί ορισμένες δικλίδες ασφαλείας αναγνώρισης της τελικής συσκευής και στην συνέχεια εκτελεί ορισμένες διαδικασίες διαχείρισης δικτύου οι οποίες αναλύονται στη συνέχεια. Μετά τον διακομιστή δικτύου ακολουθεί ο διακομιστής εφαρμογών ο οποίος είναι υπεύθυνος αρχικά να διαχειρίζεται τα δεδομένων που έχει λάβει από τις τελικές συσκευές και στην συνέχεια να πραγματοποιεί αποστολή δεδομένων προς τις συσκευές όταν απαιτείται, έτσι ώστε να υπάρχει αμφίδρομη επικοινωνία.

Μεταξύ διακομιστή δικτύου και διακομιστή εφαρμογών υπάρχει και ένας server όπου δεν παρουσιάζεται στην εικόνα 2.9, ονομάζεται Join Server και αποτελείται από λογισμικό το οποίο είναι υπεύθυνο για να αποδέχεται ή να απορρίπτει τις κλήσεις σύνδεσης (join-request) που έρχονται από τις τελικές συσκευές. Ο Join Server παρουσιάστηκε για πρώτη φορά με το LoRaWAN v1.1. και είναι επίσης διαθέσιμος στο LoRaWAN v1.0.4.

Στην συνέχεια εξετάζεται λεπτομερώς κάθε ένα στοιχείο του δικτύου LoRaWAN ξεχωριστά.

2.3.1 Τελικές Συσκευές

Μια τελική συσκευή LoRaWAN μπορεί να είναι ένας αισθητήρας, ένας ενεργοποιητής ή και τα δύο μαζί. Οι συσκευές αυτές συνήθως τροφοδοτούνται με μπαταρίες και συνδέονται ασύρματα στο δίκτυο μέσω των πυλών (Gateways) χρησιμοποιώντας διαμόρφωση σήματος LoRa.

Στις ακόλουθες εικόνες 2.10, 2.11, 2.12 εμφανίζονται ορισμένα παραδείγματα τελικών συσκευών.

		
<p>Εικόνα 2.10 - Πολυαισθητήρας γενικής χρήσης [6] (Πηγή: https://www.genericnode.com/)</p> <p>Αντιλαμβάνεται Κίνηση, Ελευθερη πτώση, Προσανατολισμό και μετράει Υγρασία, Θερμοκρασία</p>	<p>Εικόνα 2.11 - Αισθητήρας ποιότητας ατμοσφαιρικού αέρα [7] (Πηγή: https://iotfactory.eu/products/iot-sensors/indoor-environment-quality-monitoring-lorawan-sensor/)</p> <p>Αυτός ο αισθητήρας επιτρέπει την παρακολούθηση της ποιότητας του αέρα και γενικότερα του περιβάλλοντος στο οποίο είναι εγκατεστημένος. (CO₂, Θερμοκρασία, Υγρασία, Θόρυβος και Φως)</p>	<p>Εικόνα 2.12 – Μετρητής νερού [8] (Πηγή: https://iotfactory.eu/products/iot-sensors/lorawan-water-meter/)</p> <p>Ηλεκτρονικός μετρητής νερού για μέτρηση ροής κρύου ή ζεστού νερού, με συσσώρευση και μετάδοση μετρήσεων με χρήση του δικτύου LORAWAN.</p>

2.3.2 Gateways - Πύλες Δικτύου

Η πύλη δικτύου LoRa είναι η συσκευή που συνδέει τις τελικές συσκευές (sensors) με το διακομιστή δικτύου (Network server). Σκοπός της πύλης είναι να λαμβάνει τα ασύρματα μηνύματα LoRa από τελικές συσκευές και απλώς να τα προωθεί στον διακομιστή δικτύου LoRaWAN. Σε κάθε δίκτυο LoRa μπορεί να υπάρχουν παραπάνω από μια πύλη. Όσο περισσότερες πύλες υπάρχουν εγκατεστημένες σε μια λογική απόσταση ή μια από την άλλη τόσο μεγαλύτερη κάλυψη παρέχεται στο δίκτυο. Οι πύλες συνδέονται με τον διακομιστή δικτύου χρησιμοποιώντας είτε ασύρματες συνδέσεις όπως Cellular (3G/4G/5G) ή WiFi, είτε απ' ευθείας εάν βρίσκονται σε κοντινή απόσταση με καλωδιακή σύνδεση Ethernet ή οπτικών ινών. Γενικότερα, χωρίζονται σε δύο κατηγορίες. Στις εσωτερικές που χρησιμοποιούνται για εσωτερικούς χώρους και στις εξωτερικές για αντίστοιχη χρήση στους εξωτερικούς χώρους.

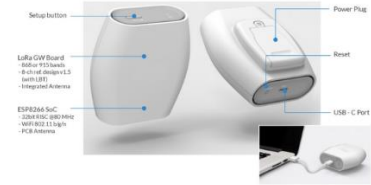


Ορισμένες επιλογές από πύλες είναι οι ακόλουθες:

Εσωτερικές Επιλογές

Οι πύλες εσωτερικών χώρων αποτελούν συνήθως μια οικονομική επιλογή και είναι κατάλληλες να παρέχουν κάλυψη σε εσωτερικούς χώρους, οι οποίοι τις περισσότερες φορές καλύπτονται από αρκετούς τοίχους, υπόγεια αλλά και πολλούς ορόφους. Αυτές οι πύλες έχουν εσωτερικές κεραίες «internal antennas» οι οποίες είναι τυπωμένες άλλοτε επάνω στην πλακέτα με το υπόλοιπο κύκλωμα και άλλοτε σε μια ξεχωριστή πλακέτα ή αλλιώς διαθέτουν εξωτερικές κεραίες τύπου «rigtail».

Ωστόσο, η εμβέλεια λήψης έχει να κάνει περισσότερο με το φυσικό περιβάλλον του εσωτερικού χώρου. Έτσι, σε ορισμένες περιπτώσεις, πύλες εσωτερικών χώρων είναι σε θέση να λαμβάνουν μηνύματα από αισθητήρες που βρίσκονται μερικά χιλιόμετρα μακριά.

Ακολουθούν ορισμένα παραδείγματα πυλών εσωτερικών χώρων LoRa (Internal gateways) στις εικόνες 2.13, 2.14, 2.15.

		
<p>Εικόνα 2.13 - The Things Indoor gateway [9] (Πηγή: https://wiki.seeedstudio.com/The-Things-Indoor-Gateway/)</p> <p>Gateway - The Things Indoor που έχει σχεδιαστεί για να συνδέεται απ' ευθείας σε μια πρίζα AC τάσης.</p>	<p>Εικόνα 2.14 - Wirnet iFemtoCell [10] (Πηγή: https://www.kerlink.com/iot-solutions/iot-gateways/wirnet-indoor-lorawan-gateways/)</p> <p>Το Wirnet iFemtoCell είναι μια έξυπνη, διαμορφώσιμη και αποδοτική εσωτερική πύλη LoRaWAN για έξυπνο δίκτυο IoT με Wi-Fi, 2,4 GHz και Ethernet (RJ45).</p>	<p>Εικόνα 2.15 - LG308 Indoor LoRaWAN Gateway [11] (Πηγή: https://www.dragino.com/products/loralorawan-gateway/item/140-lg308.html)</p> <p>Το LG308 είναι μια πύλη ανοιχτού κώδικα LoRaWAN Pico. Επιτρέπει την γεφύρωση του ασύρματου δικτύου LoRa σε ένα δίκτυο IP μέσω WiFi, Ethernet ή 3G/4G κινητής τηλεφωνίας μέσω προαιρετικής μονάδας LTE.</p>

2.3.3 Εξωτερικές Επιλογές

Οι πύλες εξωτερικού χώρου είναι κατασκευασμένες ώστε να αντέχουν σε δύσκολες καιρικές συνθήκες και παρέχουν μεγαλύτερη κάλυψη από τις εσωτερικές πύλες καθότι

συνήθως, η ευαισθησία του δέκτη μιας πύλης εξωτερικού χώρου είναι υψηλότερη από την ευαισθησία του δέκτη μιας εσωτερικής πύλης. Είναι κατάλληλες για κάλυψη τόσο σε αγροτικές όσο και σε αστικές περιοχές. Αυτές οι πύλες μπορούν να τοποθετηθούν σε στέγες ψηλών κτιρίων, μεταλλικούς σωλήνες (ιστούς) κ.λπ. Συνήθως κάθε εξωτερική πύλη έχει μια εξωτερική κεραία (δηλαδή κεραία τύπου Fiberglass) που συνδέεται μέσω ομοαξονικού καλωδίου.

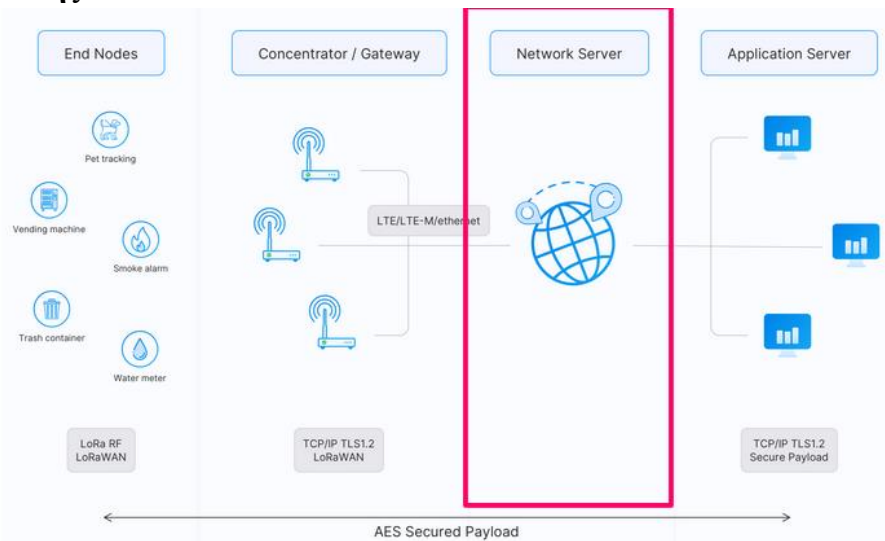
Ακολουθούν ορισμένα παραδείγματα εξωτερικών πυλών LoRa (external gateways) στις εικόνες 2.16, 2.17, 2.18 .

		
<p>Εικόνα 2.16 - MultiTech Conduit® IP67 Base Station [12] (Πηγή: https://www.multitech.com/brands/multico-nnect-conduit-ip67)</p>	<p>Εικόνα 2.17 - Cisco IXM-LPWA-800-16-K9 [13] (Πηγή: https://it-market.com/en/router/wireless/cisco/ixm-lpwa-800-16-k9)</p>	<p>Εικόνα 2.18 - MikroTik RBwAPR-2nD&R11e-LoRa8 [14] (Πηγή: https://mikrotik.com/product/wap_1)</p>

Αντίστοιχα, ακολουθούν ορισμένα παραδείγματα εσωτερικών και εξωτερικών κεραιών τύπου fiberglass στις εικόνες 2.19, 2.20, 2.21 .

		
<p>Εικόνα 2.19 - PC31 868/915MHz FR4 PCB Internal Antenna [15] (Πηγή: https://www.taoglas.com/product-category/lorasigfoxlpwa-antennas/internal-lor-antennas/)</p>	<p>Εικόνα 2.20 - LoRa omnidirectional fiberglass antenna - 8 dBi [16] (Πηγή: https://cryptostation.store/product/lora-fiberglass-antenna-kit-863-870mhz-8dbi-1300mm-rf-explorer-rfela-4-88x/)</p>	<p>Εικόνα 2.21 - Lora 868Mhz single band Fiberglass antenna [17] (Πηγή: https://www.globalsources.com/Rfid-antennas/Lora-antenna-1184407539p.htm)</p>

2.3.4 Διακομιστής Δικτύου - Network Server



Εικόνα 2.22 – Διακομιστής Δικτύου (Network Server)

Πηγή: The Things Network

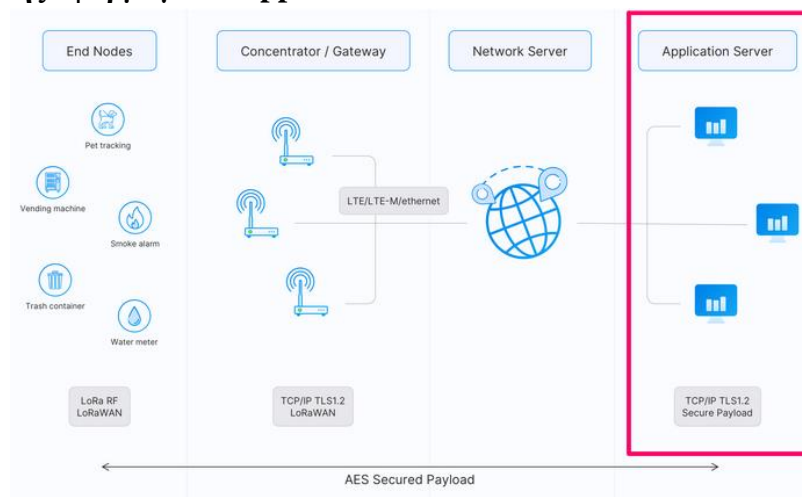
Ο διακομιστής δικτύου μπορεί να θεωρηθεί ότι αποτελεί την καρδιά του συστήματος καθώς διαχειρίζεται πύλες, τελικές συσκευές, εφαρμογές και χρήστες σε ολόκληρο το δίκτυο LoRaWAN. Στην εικόνα 2.22 φαίνεται που τοποθετείται ο διακομιστής δικτύου σε σχέση με τη συνολική αρχιτεκτονική του δικτύου.

Ένας τυπικός διακομιστής δικτύου LoRaWAN έχει τις ακόλουθες δυνατότητες:

- Είναι υπεύθυνος για την ασφάλεια του συστήματος από άκρο σε άκρο. Δημιουργεί ασφαλές συνδέσεις με βάση τον αλγόριθμο AES 128-bit (Advanced Encryption System), που είναι απαραίτητες για τη μεταφορά μηνυμάτων μεταξύ των τελικών συσκευών και του διακομιστή εφαρμογών.
- Ελέγχει και επικυρώνει την αυθεντικότητα των τελικών συσκευών καθώς επίσης την ακεραιότητα των μηνυμάτων που λαμβάνονται ή αποστέλλονται.
- Στην περίπτωση όπου ένα μήνυμα από μία τελική συσκευή ληφθεί από περισσότερες από μια πύλες (Gateways) που βρίσκονται στην ακτίνα λειτουργίας του, τότε ο διακομιστής δικτύου είναι υπεύθυνος να ταυτοποιήσει αυτά τα πολλαπλά μηνύματα και να προωθήσει μόνο ένα από αυτά προς τον διακομιστή εφαρμογών.
- Όταν πρόκειται για κατερχόμενη ζεύξη δηλαδή αποστολή μηνύματος ή εντολής από το δίκτυο ή από τον χρήστη προς την τελική συσκευή ή προς έναν ενεργοποιητή, ο διακομιστής δικτύου είναι υπεύθυνος για την βέλτιστη δρομολόγηση του μηνύματος επιλέγοντας την πύλη εκείνη που βρίσκεται πιο κοντά στον τελικό προορισμό.

- Αποστολή εντολών για βελτιστοποίηση του ρυθμού δεδομένων των τελικών συσκευών στην περίπτωση όπου έχουμε ενεργοποιήσει τον αλγόριθμο ADR. Εξετάζεται αναλυτικά το ADR στην παράγραφο 2.4.6 .
- Έλεγχος διεύθυνσης συσκευής.
- Παρέχει επιβεβαίωση αν του ζητηθεί για μηνύματα που ελήφθησαν από τελικές συσκευές (Downlink).
- Προώθηση δεδομένων εφαρμογής (payload) που ελήφθησαν από τις τελικές συσκευές προς τον κατάλληλο διακομιστή εφαρμογών.
- Προώθηση μηνυμάτων αίτησης συμμετοχής (Join-request) και αποδοχής σύνδεσης (Join-accept) μεταξύ των τελικών συσκευών και του διακομιστή σύνδεσης (Join Server).
- Είναι έτοιμος να ανταποκριθεί σε όλες τις εντολές του επιπέδου MAC.

2.3.5 Διακομιστής εφαρμογών - Application Server



Εικόνα 2.23 – Διακομιστής εφαρμογών (Application server)

Πηγή: The Things Network

Ο διακομιστής εφαρμογών αποθηκεύει και επεξεργάζεται όλα τα μηνύματα δεδομένων που έχει λάβει από τις τελικές συσκευές. Επίσης, για τις περιπτώσεις όπου απαιτείται απάντηση κατόπιν της λήψης δεδομένων, δημιουργεί όλα τα απαραίτητα δεδομένα για την κατερχόμενη ζεύξη (downlink) και δίνει εντολή να σταλούν προς τις συνδεδεμένες τελικές συσκευές μέσω του διακομιστή δικτύου. Ένα δίκτυο LoRaWAN μπορεί να έχει περισσότερους από έναν διακομιστή Εφαρμογών είτε για να μετριάξει τον φόρτο εργασίας, έτσι ώστε να υπάρχει γρήγορη απόκριση λειτουργίας, είτε για λόγους ασφαλείας δηλαδή αν συμβεί μια βλάβη σε έναν από αυτούς (application server) να αναλάβει ο δεύτερος, τρίτος κτλ. Η θέση του συγκεκριμένου διακομιστή σε σχέση με την αρχιτεκτονική του δικτύου φαίνεται στην εικόνα 2.23.

2.3.6 Διακομιστής σύνδεσης - Join Server

Ο διακομιστής σύνδεσης (Join Server) βοηθά στην ασφαλή ενεργοποίηση της τελικής συσκευής δημιουργώντας και αποθηκεύοντας τα απαραίτητα κλειδιά ασφαλείας. Η διαδικασία σύνδεσης ξεκινά από την τελική συσκευή με την αποστολή ενός μηνύματος αίτησης σύνδεσης προς τον διακομιστή σύνδεσης μέσω του διακομιστή δικτύου. Ο διακομιστής σύνδεσης επεξεργάζεται το μήνυμα αίτησης σύνδεσης, και ως απάντηση αποδοχής δημιουργεί κλειδιά περιόδου σύνδεσης-λειτουργίας (NwkSKey και το AppSKey) τα οποία μεταφέρει στον διακομιστή δικτύου και στον διακομιστή εφαρμογής αντίστοιχα.

2.4 Τεχνικές λεπτομέρειες τεχνολογίας στο LoRa

2.4.1 Τοπικές παράμετροι

Μια πολύ σημαντική τεχνική παράμετρος του LoRaWAN είναι το φάσμα συχνοτήτων λειτουργίας του καθότι λειτουργεί σε συχνότητες χωρίς άδεια. Αυτό σημαίνει ότι ο καθένας μπορεί να χρησιμοποιήσει αυτές τις συχνότητες χωρίς να χρειάζεται να πληρώσει τέλη μετάδοσης ή να απαιτείται η έκδοση άδειας εκπομπής.

Εν αντιθέσει με άλλα ελεύθερα δίκτυα όπου οποιοσδήποτε επιτρέπεται να ενεργοποιήσει συσκευές εκπομπής και δρομολογητές και να μεταδώσει σήμα - όπως για παράδειγμα το WiFi, το οποίο χρησιμοποιεί τις ζώνες ISM 2,4 GHz και 5 GHz παγκοσμίως χωρίς την ανάγκη κάποιας άδειας - το LoRaWAN χρησιμοποιεί χαμηλότερες συχνότητες έτσι ώστε να μπορεί να πετύχει μεγαλύτερη εμβέλεια. Όμως, το γεγονός ότι στις συχνότητες αυτές είναι δυνατό να επιτευχθεί μεγαλύτερη εμβέλεια συνοδεύεται επίσης από περισσότερους περιορισμούς που συνήθως αφορούν την κάθε μια χώρα ξεχωριστά. Για να μην υπάρχουν μεγάλες διαφοροποιήσεις από χώρα σε χώρα η ομάδα που ανέπτυξε το πρωτόκολλο προσπάθησε έτσι ώστε το αποτέλεσμα να είναι όσο το δυνατόν πιο ομοιόμορφο γίνεται για κάθε μια διαφορετική περιοχή λειτουργίας.

Έτσι, το LoRaWAN καθορίζεται για έναν αριθμό ζωνών [18] οι οποίες είναι παρόμοιες ώστε να μπορούν να υποστηρίξουν το ίδιο ακριβώς πρωτόκολλο σε κάθε μια περιοχή με την ίδια ακριβώς υλοποίηση αναφορικά με τα συστήματα υποστήριξης. Συνεπώς, το LoRaWAN έχει επίσημες περιφερειακές παραμέτρους (Regional Parameters), οι οποίες είναι διαθέσιμες στον ιστότοπο του LoRa Alliance.

Αυτές οι περιφερειακές παράμετροι για το LoRaWAN προσδιορίζουν πρωτίστως ένα κοινό υποσύνολο καναλιών. Όμως, για ορισμένες χώρες δύναται να υπάρχουν και ορισμένες παραλλαγές μεταξύ των κανονισμών. Ένα παράδειγμα είναι οι περιφερειακές ρυθμίσεις που ισχύουν για την Ασία. Στην Ασία γενικά ορίζεται το φάσμα συχνοτήτων μεταξύ 915 – 928 MHz και μέσα σε αυτό το φάσμα ορίζονται 4 διαφορετικά περιφερειακά πλάνα παραμέτρων (AS923-1, AS923-2, AS923-3, AS923-4) όπως ακριβώς παρουσιάζεται στον πίνακα 2.2.

Plan	AS923-1	AS923-2	AS923-3	AS923-4
Default Freq band	915 to 928 MHz	915 to 928 MHz	915 to 928 MHz	917 to 920 MHz
	923.20 MHz	921.4 MHz	916.6 MHz	917.3 MHz
Mandatory Channel Freq (Join Req)	923.40 MHz	921.6 MHz	916.8 MHz	917.5 MHz

Πίνακας 2.2 – Περιφερειακά πλάνα παραμέτρων για την περιοχή της Ασίας
(Πηγή: https://loro-alliance.org/resource_hub/rp2-1-0-3-lorawan-regional-parameters/)

Σε ορισμένες χώρες, ενδέχεται να χρησιμοποιούνται περισσότερα από ένα φάσμα συχνοτήτων. Για παράδειγμα, στην Ολλανδία, μπορούν να χρησιμοποιηθούν τόσο το EU868-870 όσο και το EU433 όπως φαίνεται στον πίνακα 2.3.

Netherlands (NL)	433.05 – 434.79 MHz	EU433
	863 – 870 MHz	EU863-870

Πίνακας 2.3 – Συχνότητες λειτουργίας LoRa για την Ολλανδία
(Πηγή: <https://www.thethingsnetwork.org/docs/lorawan/frequencies-by-country/>)

Επιπλέον, κάθε διαχειριστής ενός δικτύου LoRa είναι ελεύθερος να επιλέξει τις πρόσθετες παραμέτρους που θέλει να ενεργοποιήσει, όπως πρόσθετα κανάλια εκπομπής τα οποία να ανήκουν μέσα στο επιτρεπτό φάσμα εκπομπής. Αυτές οι περαιτέρω παράμετροι είναι γνωστές ως «Άλλες παράμετροι» (other parameters) και είναι διαθέσιμες στο site του παρόχου του δικτύου LoRa. Π.χ. για το TTN, είναι διαθέσιμες στο αποθετήριο GitHub [19].

2.4.2 Μηνύματα

Η επικοινωνία με την χρήση της τεχνολογίας LoRa είναι ασύρματη. Η πραγματοποίηση αυτής της ασύρματης επικοινωνίας πραγματοποιείται δια μέσω μηνυμάτων.

Τα μηνύματα χρησιμοποιούνται για τη μεταφορά εντολών MAC (Media Access Control) και την μεταφορά δεδομένων εφαρμογών. Με βάση την κατεύθυνση τους χωρίζονται σε δύο μεγάλες κατηγορίες. Στα μηνύματα ανοδικής ζεύξης και στα μηνύματα καθοδικής ζεύξης.

Μηνύματα ανοδικής ζεύξης - Τα μηνύματα ανοδικής ζεύξης αποστέλλονται από τελικές συσκευές προς τον διακομιστή δικτύου (Network Server) δια μέσω μίας ή περισσοτέρων πυλών (Gateways). Ανάλογα εάν το μήνυμα ανοδικής σύνδεσης ανήκει στον διακομιστή εφαρμογών ή στο διακομιστή σύνδεσης, ο διακομιστής δικτύου μόλις λάβει το μήνυμα θα προβεί στην απαιτούμενη προώθηση.

Μηνύματα κατερχόμενης ζεύξης – Τα μηνύματα κατερχόμενης ζεύξης ξεκινούν από τον διακομιστή εφαρμογών ή από τον διακομιστή σύνδεσης και αποστέλλονται σε μία μόνο τελική συσκευή περνώντας από τον διακομιστή δικτύου και από μία μόνο πύλη.

Στον ακόλουθο πίνακα αναφέρονται όλα τα μηνύματα που αφορούν και τις δύο ζεύξεις σχετικά με το πρωτόκολλο LoRaWAN για τις εκδόσεις 1.0.x και 1.1 .

LoRaWAN 1.0.x	LoRaWAN 1.1	Περιγραφή - Επεξήγηση
Μηνύματα MAC (Media Access Control)		
Join-request Αίτημα σύνδεσης	Join-request Αίτημα σύνδεσης	<p>Κάθε φορά που μία τελική συσκευή θέλει να συνδεθεί σε ένα δίκτυο LoRa στέλνεται ένα μήνυμα αίτησης σύνδεσης (join-request) προς το δίκτυο και συγκεκριμένα προς τον διακομιστή δικτύου που δηλώνει την επιθυμία της συσκευής να συνδεθεί.</p> <p>Σε εκδόσεις LoRaWAN προγενέστερες της 1.0.4, το μήνυμα της αίτησης σύνδεσης προωθείται από τον Διακομιστή Δικτύου στον Διακομιστή Εφαρμογών. Στο LoRaWAN 1.1 και 1.0.4+, ο διακομιστής δικτύου προωθεί το μήνυμα αίτησης σύνδεσης στον διακομιστή σύνδεσης της συσκευής.</p> <p>Και για τις δύο εκδόσεις πρωτοκόλλου το μήνυμα αίτησης σύνδεσης δεν είναι κρυπτογραφημένο.</p>
Join-accept Αποδοχή	Join-accept Αποδοχή	<p>Σε συνέχεια της αίτησης σύνδεσης και αφού πληρούνται οι προϋποθέσεις ένταξης της τελικής συσκευής στο δίκτυο, ακολουθεί κρυπτογραφημένο μήνυμα αποδοχής (join-accept) προς την τελική συσκευή.</p> <p>Σε εκδόσεις LoRaWAN παλαιότερες από την 1.0.4, το μήνυμα αποδοχής (Join-accept) δημιουργείται από τον διακομιστή εφαρμογών. Από την έκδοση 1.0.4+ έως και την 1.1 το μήνυμα σύνδεσης (Join-accept) δημιουργείται από τον διακομιστή συνδέσεων (Join Server). Και στις δύο περιπτώσεις το μήνυμα περνά από τον διακομιστή δικτύου. Στη συνέχεια, ο διακομιστής δικτύου δρομολογεί το μήνυμα αίτημα σύνδεσης (Join-accept) στη σωστή τελική συσκευή μέσω μιας πύλης (gateway).</p>

RFU (Reserved for future usage) Δεσμευμένο για μελλοντική χρήση	Rejoin-request Αίτημα επανασύνδεσης	Το μήνυμα αίτησης επανασύνδεσης (Rejoin-request) ξεκινά πάντα από μια τελική συσκευή και αποστέλλεται στον διακομιστή δικτύου στην περίπτωση όπου για κάποιο λόγο διακοπεί η σύνδεση της τελικής συσκευής με το δίκτυο ή λόγω περιαγωγής. Κατά την αποδοχή ενός αιτήματος επανασύνδεσης από το σύστημα, το δίκτυο απαντά με ένα μήνυμα αποδοχής (Join-accept).
Data Messages - Μηνύματα δεδομένων		
Proprietary Ιδιωτικό	Proprietary Ιδιωτικό	Χρησιμοποιείται για την υλοποίηση μη τυπικών μορφών μηνυμάτων.
Unconfirmed Data Up Ανοδικό ανεπιβεβαίωτο	Unconfirmed Data Up Ανοδικό ανεπιβεβαίωτο	Ανοδικό μήνυμα δεδομένων από την τελική συσκευή προς το δίκτυο χωρίς να απαιτείται επιβεβαίωση λήψης.
Unconfirmed Data Down Καθοδικό ανεπιβεβαίωτο	Unconfirmed Data Down Καθοδικό ανεπιβεβαίωτο	Καθοδικό μήνυμα δεδομένων από την τελική συσκευή προς το δίκτυο χωρίς να απαιτείται επιβεβαίωση λήψης.
Confirmed Data Up Ανοδικό επιβεβαιωμένο	Confirmed Data Up Ανοδικό επιβεβαιωμένο	Ανοδικό μήνυμα δεδομένων από την τελική συσκευή προς το δίκτυο με απαίτηση επιβεβαίωσης λήψης.

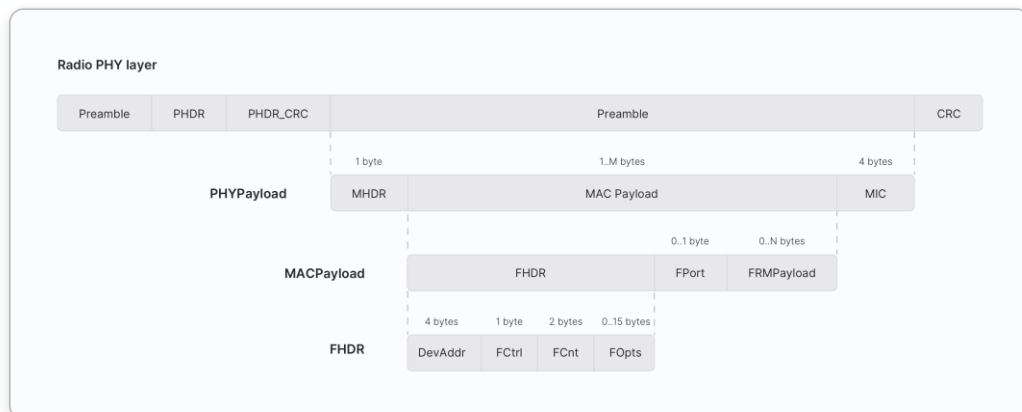
Πίνακας 2.4 – Μηνύματα LoRaWAN

(Πηγή: <https://www.thethingsnetwork.org/docs/lorawan/message-types/>)

Όπως φαίνεται στον παραπάνω πίνακα 2.4 υπάρχουν 4 τύποι μηνυμάτων δεδομένων που χρησιμοποιούνται τόσο στο LoRaWAN 1.0.x όσο και στο 1.1. Αυτοί οι τύποι χρησιμοποιούνται για τη μεταφορά τόσο εντολών MAC (Media Access Control) όσο και δεδομένων εφαρμογών και μπορούν να συνδυαστούν μαζί σε ένα μόνο μήνυμα.

Τα μηνύματα δεδομένων μπορούν να επιβεβαιωθούν ή να μην επιβεβαιωθούν. Τα επιβεβαιωμένα μηνύματα δεδομένων πρέπει να επιβεβαιώνονται από το δίκτυο.

Η δομή ενός μηνύματος δεδομένων είναι αυτή που παρουσιάζεται στην ακόλουθη εικόνα 2.24 :



Εικόνα 2.24 – Δομή μηνύματος LoRaWAN

Πηγή: The Things Network

Το ωφέλιμο φορτίο (MAC Payload) των μηνυμάτων δεδομένων αποτελείται από μια κεφαλίδα πλαισίου (FHDR) ακολουθούμενη από ένα προαιρετικό πεδίο θύρας (FPort) και ένα προαιρετικό ωφέλιμο φορτίο πλαισίου (FRMPayload).

Για κάθε ένα από αυτά τα πεδία το μέγεθος είναι καθορισμένο σε bytes όπως ακριβώς φαίνεται στην φωτογραφία εκτός από το πεδίο ωφέλιμου φορτίου (MAC Payload) του οποίου το μέγεθος είναι συγκεκριμένο και ορίζεται ανά περιοχή λειτουργίας και με βάση τον ρυθμό δεδομένων που έχει επιλεγεί. Στις τοπικές παραμέτρους του πρωτοκόλλου LoRaWAN αναφέρονται όλα τα μεγέθη για κάθε μια περιοχή λειτουργίας.

Αποστολή εντολών MAC και δεδομένων για κάποια συγκεκριμένη εφαρμογή.

Ένα μήνυμα δεδομένων μπορεί να περιέχει οποιαδήποτε ακολουθία εντολών MAC. Ένα μήνυμα δεδομένων μπορεί να μεταφέρει ταυτόχρονα εντολές MAC και δεδομένα εφαρμογής σε ξεχωριστά πεδία. Οι εντολές MAC μπορούν να αποσταλούν είτε στο πεδίο επιλογών πλαισίου (FOpts) είτε στο πεδίο ωφέλιμου φορτίου πλαισίου (FRMPayload) ενός μηνύματος δεδομένων, αλλά όχι και τα δύο ταυτόχρονα. Δεδομένα εφαρμογής μπορούν να σταλούν στο πεδίο ωφέλιμο φορτίο πλαισίου (FRMPayload) ενός μηνύματος δεδομένων. Το πεδίο FRMPayload δεν μπορεί να περιέχει εντολές MAC και δεδομένα εφαρμογής ταυτόχρονα.

2.4.3 Ασφάλεια

Για να παρέχεται η απαιτούμενη ασφάλεια κατά την επικοινωνία μέσω του πρωτοκόλλου LoRaWAN έχουν καθοριστεί τα ακόλουθα τρία κλειδιά ασφαλείας:

- a) Κλειδί συνεδρίας δικτύου - NwkSKey (Network Session Key)

- b) Κλειδί συνεδρίας εφαρμογής - AppSKey (Application Session Key)
- c) Κλειδί εφαρμογής – AppKey (Application Key)

Και τα τρία παραπάνω κλειδιά έχουν μήκος 128bit. Τα δύο από αυτά είναι κλειδιά συνεδρίας δηλαδή δημιουργούνται με την εκκίνηση της επικοινωνίας και λήγουν με το πέρας αυτής. Η κωδικοποίηση των δεδομένων με τα παραπάνω κλειδιά πραγματοποιείται μέσω του αλγορίθμου AES-128 (Advanced Encryption Standard).

Κλειδιά συνεδρίας - Session Keys

Κατά την σύνδεση μιας τελικής συσκευής στο δίκτυο δημιουργείται ένα κλειδί συνεδρίας εφαρμογής AppSKey και ένα κλειδί συνεδρίας δικτύου NwkSKey. Το NwkSKey είναι κοινόχρηστο με το δίκτυο δηλαδή επιτρέπεται η διαμοίραση του με τις υπόλοιπες συσκευές του δικτύου, ενώ το AppSKey παραμένει ιδιωτικό δηλαδή είναι γνωστό μόνο από τον διακομιστή εφαρμογών. Αυτά τα κλειδιά συνεδρίας θα χρησιμοποιηθούν κατά την διάρκεια της συνεδρίας.

Το κλειδί συνεδρίας δικτύου (NwkSKey) χρησιμοποιείται για την αλληλεπίδραση μεταξύ της τελικής συσκευής και του διακομιστή δικτύου και πιο συγκεκριμένα χρησιμοποιείται για την επικύρωση της ακεραιότητας-ορθότητας κάθε μηνύματος μέσω του κώδικα ακεραιότητας μηνύματος (MIC). Ο κώδικας MIC είναι παρόμοιος με ένα άθροισμα ελέγχου, με τη διαφορά ότι αποτρέπει την σκόπιμη παραβίαση ενός μηνύματος.

Το κλειδί συνεδρίας εφαρμογής (AppSKey) χρησιμοποιείται για την κρυπτογράφηση και την αποκρυπτογράφηση του ωφέλιμου φορτίου του μηνύματος δηλαδή των πραγματικών δεδομένων που αποστέλλει η τελική συσκευή. Το ωφέλιμο φορτίο είναι πλήρως κρυπτογραφημένο μεταξύ της τελικής συσκευής και του διαχειριστή του διακομιστή εφαρμογών (Application Server). Αυτό σημαίνει ότι κανείς εκτός από τον διαχειριστή δεν μπορεί να διαβάσει το περιεχόμενο των μηνυμάτων που στέλνεται ή λαμβάνεται.

Αυτά τα δύο κλειδιά συνεδρίας (NwkSKey και AppSKey) είναι μοναδικά ανά συσκευή και ανά περίοδο λειτουργίας. Σε κάθε δυναμική ενεργοποίηση της συσκευής (OTAA – Over the air activation), αυτά τα κλειδιά δημιουργούνται εκ νέου, ενώ εάν η ενεργοποίηση της συσκευής γίνεται στατικά με την μέθοδο ABP (Activation by Personalisation) αυτά τα κλειδιά παραμένουν ίδια.

Το κλειδί εφαρμογής (AppKey) είναι γνωστό από την τελική συσκευή και από την εφαρμογή που καταγράφει τα στοιχεία της. Οι δυναμικά ενεργοποιημένες συσκευές (OTAA) χρησιμοποιούν το Application Key (AppKey) για να δημιουργήσουν τα δύο κλειδιά περιόδου λειτουργίας κατά τη διαδικασία ενεργοποίησης.

2.4.4 Frame Counters - Μετρητές πλαισίων

Επειδή η μετάδοση των μηνυμάτων γίνεται μέσω ασύρματου πρωτόκολλο, οποιοσδήποτε βρεθεί στην ίδια συχνότητα θα μπορεί να καταγράψει και να αποθηκεύει αυτά τα μηνύματα.

Όμως δεν θα είναι δυνατή η ανάγνωση αυτών των μηνυμάτων χωρίς το κλειδί συνεδρίας εφαρμογής (AppSKey), καθότι είναι κρυπτογραφημένα. Επίσης, δεν θα είναι δυνατό να τα παραβιαστούν-τροποποιηθούν χωρίς το κλειδί συνεδρίας δικτύου (NwkSKey), καθότι ο έλεγχος ακεραιότητας MIC θα αποτύχει. Ωστόσο, είναι δυνατόν να συμβεί καταγραφή και επαναμετάδοση μηνυμάτων σε μεταγενέστερο χρόνο ως μια προσπάθεια αλλοίωσης των πραγματικών δεδομένων. Αυτές οι επιθέσεις επανάληψης μπορούν να εντοπιστούν και να αποκλειστούν χρησιμοποιώντας μετρητές πλαισίων (Frame Counters).

Υπάρχουν δύο μετρητές πλαισίων. Ένας για τα ανερχόμενα μηνύματα (FCntUp – Frame Counter Up) και ένας για τα κατερχόμενα (FCntDown – Frame Counter Down). Όταν ενεργοποιείται μια συσκευή, αυτοί οι μετρητές πλαισίων ορίζονται αρχικά στο 0. Κάθε φορά που η συσκευή εκπέμπει ένα μήνυμα ανόδου, το FCntUp αυξάνεται κατά μία μονάδα και κάθε φορά που το δίκτυο στέλνει ένα μήνυμα κατερχόμενης ζεύξης, το FCntDown αυξάνεται αντίστοιχα. Εάν είτε η συσκευή είτε το δίκτυο λάβει ένα μήνυμα με χαμηλότερο μετρητή από τον τελευταίο, το σύστημα αντιλαμβάνεται ότι πρόκειται για επανάληψη μηνύματος και το αγνοεί.

Παρόλο την ασφάλεια που προσφέρει το συγκεκριμένο επίπεδο ασφαλείας έχει ορισμένες αρνητικές συνέπειες όταν οι τελικές συσκευές ενεργοποιούνται στατικά μέσω ABP (Activation by Personalization). Σε αυτήν την περίπτωση κάθε φορά που η συσκευή επανεκκινείται, όλοι οι μετρητές πλαισίων (FCntUp και FCntDown) επανέρχονται στο 0. Συνεπώς, όλα τα μηνύματα από τη συσκευή έως ότου το FCntUp γίνει υψηλότερο από το προηγούμενο FCntUp θα απορριφθούν. Άρα για να μπορούν οι τελικές συσκευές να στέλνουν άμεσα μηνύματα προς τον διακομιστή εφαρμογών μετά από επανεκκίνηση θα πρέπει να γίνεται άμεση διαγραφή της τελικής συσκευής και επανακαταχώρηση στον διακομιστή εφαρμογών του δικτύου. Για να μην παρουσιάζονται τέτοιου είδους προβλήματα η λύση είναι να χρησιμοποιούνται τελικές συσκευές οι οποίες μπορούν και ενεργοποιούνται μέσω OTAA (Over the air activation). Σε αυτήν την περίπτωση αποφεύγονται τέτοιου είδους προβλήματα.

2.4.5 Spread Spectrum - Ευρύ φάσμα

Η τεχνολογία LoRa κάνει χρήση της τεχνικής διασποράς φάσματος[20] (Spread Spectrum). Η τεχνική αυτή αποτελείται από διάφορους τρόπους με τους οποίους το φάσμα ενός σήματος κατανέμεται εσκεμμένα σε ένα μεγαλύτερο φάσμα με σκοπό την αποφυγή υποκλοπών αλλά και την αύξηση της αντοχής σε τυχόν παρεμβολές.

Υπάρχουν δύο σημαντικές τεχνικές διασποράς φάσματος:

1. Διασπορά ευθείας ακολουθίας - Direct Sequence Spread Spectrum (DSSS)
2. Διασπορά με αναπήδηση συχνότητας - Frequency Hopping Spread Spectrum (FHSS)

Το πρωτόκολλο LoRaWAN κάνει χρήση της διασποράς με αναπήδηση συχνότητας[21] (FHSS). Σε αυτήν την περίπτωση το διαθέσιμο εύρος ζώνης συχνοτήτων χωρίζεται σε μικρότερες περιοχές με τέτοιο εύρος που να μπορεί να υποστηρίξει τη μετάδοση του σήματος πληροφορίας. Κάθε μια από αυτές τις περιοχές χαρακτηρίζεται από ένα φέρον σήμα. Το αρχικό σήμα πληροφορίας διαμορφώνεται περιοδικά σε διαφορετική περιοχή συχνοτήτων και η επιλογή πραγματοποιείται με βάση ένα κώδικα που ονομάζεται κώδικας διασποράς. Το ευρύ πλέον σήμα έχει διασκορπισμένη την συνολική ισχύ του αρχικού σήματος με αποτέλεσμα να προσεγγίζει ένα σήμα θορύβου και κατ' επέκταση να ξεγελάει τυχόν προσπάθειες υποκλοπής ότι πρόκειται για σήμα θορύβου. Όμως, ακόμη και αν κάποιος γνωρίζει ότι πραγματοποιείται εκπομπή σήματος μέσω διασποράς αναπήδησης, για να μπορέσει να ανακτήσει το αρχικό σήμα θα πρέπει να γνωρίζει την αρχική ραδιοσυχνότητα, τον κώδικα διασποράς και τον ρυθμό σε bit/s του κώδικα διασποράς.

Αυτά τα τρία στοιχεία γνωρίζει και ο δέκτης για να μπορέσει να επαναφέρει το σήμα από τα επίπεδα θορύβου στο αρχικό σήμα που εξέπεμψε η τελική συσκευή.

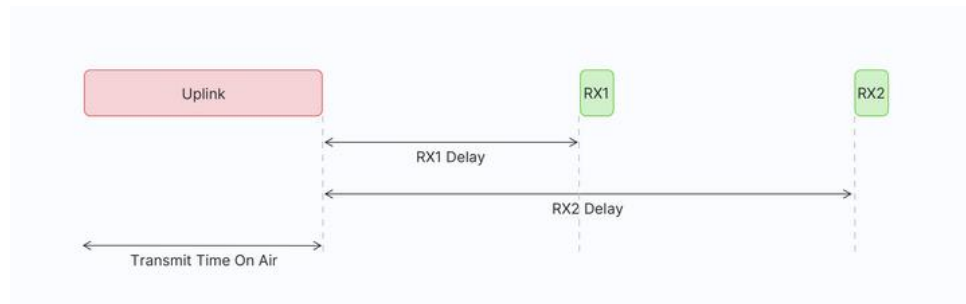
2.4.6 Classes Συσκευών

Το πρωτόκολλο LoRaWAN ορίζει τρεις κλάσεις τελικών συσκευών: Κλάση A, Κλάση B και Κλάση Γ. Όλες οι συσκευές πρέπει να υποστηρίζουν τις λειτουργίες της κλάσης A, ενώ οι λειτουργίες για την κλάση B και την κλάση Γ αποτελούν επεκτάσεις των λειτουργιών της κλάσης A.

Όλες οι κλάσεις τελικών συσκευών υποστηρίζουν αμφίδρομη επικοινωνία (uplink και downlink) όμως είναι πολύ σημαντικό να σημειωθεί ότι κατά την διάρκεια ενός ανερχόμενου μηνύματος η τελική συσκευή δεν μπορεί να λάβει κατερχόμενο μήνυμα.

Κλάση A

Η επικοινωνία κλάσης A ξεκινά πάντα από την τελική συσκευή. Μια συσκευή μπορεί να στείλει ένα μήνυμα ανοδικής ζεύξης (uplink) ανά πάσα στιγμή. Μόλις ολοκληρωθεί η μετάδοση ανοδικής ζεύξης, η συσκευή ανοίγει δύο σύντομα παράθυρα λήψης (downlink) όπως ακριβώς φαίνεται στην εικόνα 2.25. Υπάρχει μια καθυστέρηση μεταξύ του τέλους της μετάδοσης ανοδικής ζεύξης και της έναρξης των παραθύρων λήψης καθοδικής ζεύξης (RX1 και RX2) αντίστοιχα. Εάν ο διακομιστής δικτύου δεν ανταποκρίνεται κατά τη διάρκεια αυτών των δύο παραθύρων λήψης, η επόμενη κατερχόμενη ζεύξη θα είναι μετά την επόμενη μετάδοση ανερχόμενης ζεύξης δηλαδή μετά την αποστολή ενός νέου μηνύματος.

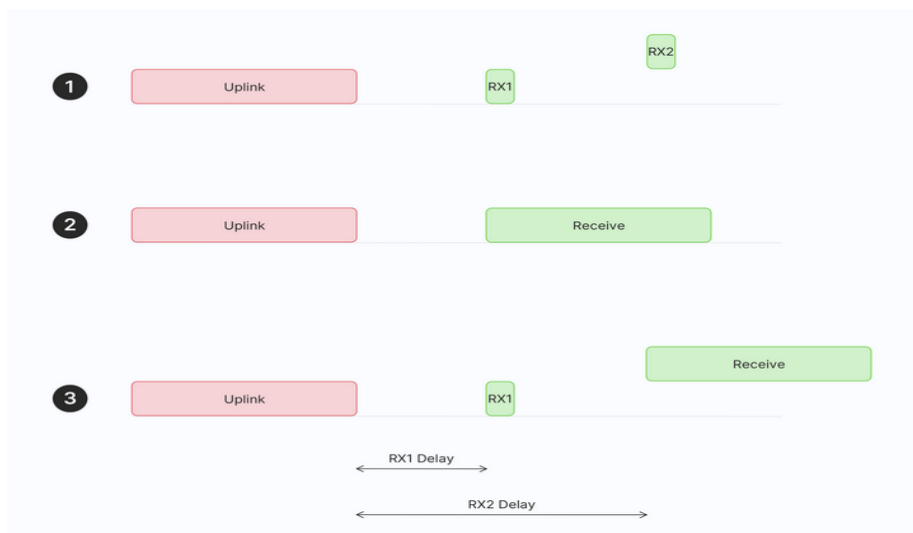


Εικόνα 2.25 – Λειτουργίας τελικής συσκευής με κλάση A

Πηγή: The Things Network

Κλάση A – Παράθυρα λήψης

Ο διακομιστής δικτύου μπορεί να ανταποκριθεί κατά το πρώτο παράθυρο λήψης (RX1) ή κατά το δεύτερο παράθυρο λήψης (RX2), αλλά δεν μπορεί να ανταποκριθεί και στα δύο παράθυρα λήψης. Ακολουθούν οι τρεις διαφορετικές περιπτώσεις που φαίνονται στην εικόνα 2.26, που μπορούν να συμβούν όταν πρόκειται για μηνύματα κατερχόμενης ζεύξης.



Εικόνα 2.26 – Συμπεριφορά συσκευών Κλάσης A κατά την κατερχόμενη ζεύξη (Downlink)

Πηγή: The Things Network

Περίπτωση 1: Μετά από μια ανοδική ζεύξη (uplink) η τελική συσκευή ανοίγει και τα δύο παράθυρα λήψης, αλλά δεν λαμβάνει μήνυμα κατερχόμενης σύνδεσης κατά τη διάρκεια κανενός παραθύρου λήψης.

Περίπτωση 2: Μετά από μια ανοδική ζεύξη (uplink) η τελική συσκευή λαμβάνει μια κατερχόμενη σύνδεση κατά το πρώτο παράθυρο λήψης και επομένως δεν ανοίγει το δεύτερο παράθυρο λήψης.

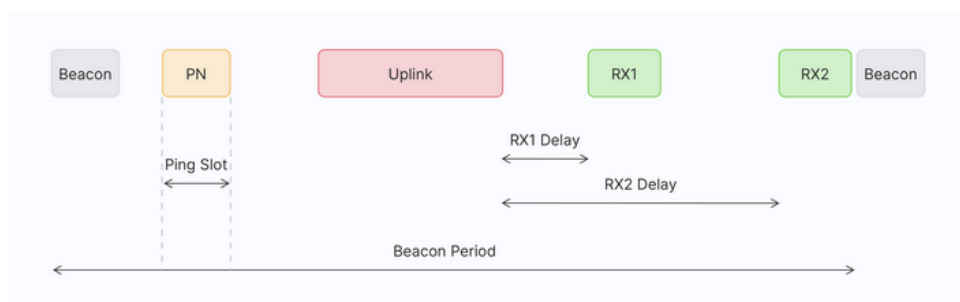
Περίπτωση 3: Μετά από μια ανοδική ζεύξη (uplink) η τελική συσκευή ανοίγει το πρώτο παράθυρο λήψης αλλά δεν λαμβάνει κάποια κατερχόμενη ζεύξη (downlink). Επομένως ανοίγει το δεύτερο παράθυρο λήψης και πραγματοποιείται λήψη μιας κατερχόμενης ζεύξης.

Τα γενικά χαρακτηριστικά των τελικών συσκευών που ανήκουν στην κλάση A είναι ότι συχνά τροφοδοτούνται με μπαταρίες, έχουν τη χαμηλότερη κατανάλωση ενέργειας σε σύγκριση με τις άλλες δύο κλάσεις, περνούν τον περισσότερο χρόνο τους σε κατάσταση ύπνου (sleep mode), συνήθως διατηρούν μεγάλα διαστήματα μεταξύ των uplinks και τέλος για να λάβει η συσκευή μια κατερχόμενη ζεύξη θα πρέπει πρώτα να εκπέμψει μια ανοδική ζεύξη.

Χρησιμοποιούνται συνήθως για εφαρμογές περιβαλλοντικής παρακολούθησης, παρακολούθηση ζώων, ανίχνευση πυρκαγιάς, ανίχνευση διαρροής νερού, ανίχνευση σεισμών, παρακολούθηση τοποθεσίας.

Κλάση B

Η Κλάση B αποτελεί επέκταση της κλάσης A, έτσι τα παράθυρα λήψης που υπάρχουν στην κλάση A ισχύουν και για την κλάση B. Οι τελικές συσκευές κλάσης B ανοίγουν προγραμματισμένα παράθυρα λήψης για να μπορούν να λάβουν μηνύματα κατερχόμενης ζεύξης από τον διακομιστή δικτύου όπως παρουσιάζεται από την εικόνα 2.27. Αυτό γίνεται με τη βοήθεια χρονικά συγχρονισμένων «beacon» που μεταδίδονται μέσω των πυλών (gateways) του δικτύου. Ο χρόνος μεταξύ δύο «beacon» είναι γνωστός ως περίοδος beacon. Η συσκευή ανοίγει υποδοχές «ring slot» για τη λήψη μηνυμάτων κατερχόμενης ζεύξης σε συγκεκριμένο χρόνο από τον διακομιστή δικτύου.



Εικόνα 2.27 – Συμπεριφορά συσκευών Κλάσης B κατά την κατερχόμενη ζεύξη (Downlink)

Πηγή: The Things Network

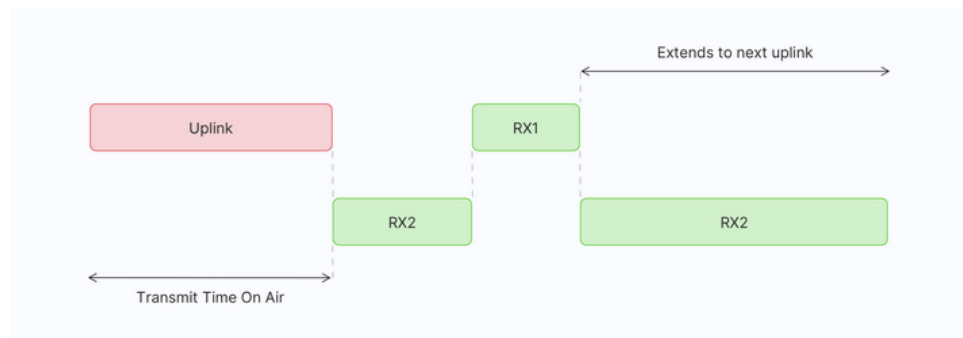
Λειτουργία τελικής συσκευής κλάσης B

Οι τελικές συσκευές κλάσης B έχουν χαμηλότερο χρόνο κατερχόμενης ζεύξης από τις τελικές συσκευές κλάσης A, επειδή είναι προσβάσιμες σε προρυθμισμένους χρόνους και δεν είναι απαραίτητο να στείλουν μια ανοδική σύνδεση για να λάβουν μια κατερχόμενη σύνδεση. Η διάρκεια ζωής της μπαταρίας είναι μικρότερη στην κλάση B από την κλάση A, καθώς λόγω της χρήσης των beacon και των υποδοχών ring slot η συσκευή βρίσκεται για μεγαλύτερο χρονικό διάστημα σε ενεργή λειτουργία.

Οι τελικές συσκευές κλάσης B συνήθως χρησιμοποιούνται ως μετρητές κοινής ωφέλειας και ως συσκευές αναφοράς της θερμοκρασίας.

Κλάση C

Οι συσκευές κλάσης C όπου η συμπεριφορά τους φαίνεται από την εικόνα 2.28 επεκτείνουν την κλάση A διατηρώντας τα παράθυρα λήψης ανοιχτά καθ' όλη την χρονική διάρκεια εκτός από τα διαστήματα εκείνα όπου πραγματοποιείται εκπομπή ανοδικής ζεύξης. Αυτό επιτρέπει την πραγματοποίηση μίας επικοινωνίας χαμηλού χρόνου για τις καθοδικές ζεύξεις αυξάνοντας το κόστος ενέργειας σε σχέση με τις συσκευές κλάσης A.



Εικόνα 2.28 – Συμπεριφορά συσκευών Κλάσης C κατά την κατερχόμενη ζεύξη (Downlink)

Πηγή: The Things Network

Έτσι, οι τελικές συσκευές κλάσης C συχνά τροφοδοτούνται από το δίκτυο καθότι είναι αρκετά ενεργοβόρες λόγω του παράθυρου συνεχούς λήψης και βρίσκουν εφαρμογή συνήθως σε μετρητές κοινής ωφέλειας με δυνατότητα άμεσης διακοπής και ενεργοποίησης καθώς επίσης και σε τελικές συσκευές για την ενεργοποίηση και απενεργοποίηση δημόσιου φωτισμού.

2.4.7 Ενεργοποίηση συσκευών

Κάθε τελική συσκευή πρέπει να είναι εγγεγραμμένη στο δίκτυο που πρόκειται να λειτουργήσει πριν από την αποστολή και λήψη μηνυμάτων. Αυτή η διαδικασία είναι γνωστή ως ενεργοποίηση.

Υπάρχουν δύο διαθέσιμες μέθοδοι ενεργοποίησης:

1. Over-The-Air-Activation (OTAA) είναι η συνιστώμενη μέθοδος ενεργοποίησης για τελικές συσκευές καθότι είναι η πιο ασφαλής ως αναφορά τα δεδομένα.

Για να γίνει ενεργοποίηση μιας τελικής συσκευής θα πρέπει αυτή να έχει αποθηκευμένα τα ακόλουθα κλειδιά:

1. DevEUI (Device Extended Unique Identifier), το οποίο είναι μοναδικό για κάθε συσκευή έχει μέγεθος 64bit και χρησιμοποιείται για την αναγνώριση της συσκευής από το δίκτυο. Θα μπορούσε με ευκολία να παρομοιαστεί ως η MAC address (Media Access Control) της τελικής συσκευής. Ορισμένες τελικές συσκευές που υπάρχουν στην αγορά έχουν ένα προκαθορισμένο DevEUI από τον κατασκευαστή τους. Εάν όχι, το DevEUI καθορίζεται μέσω κώδικα από τον χρήστη.

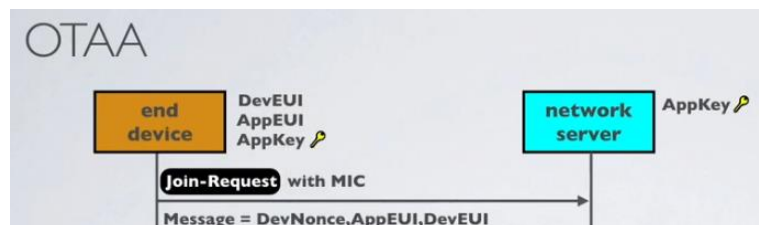
2. AppEUI (Application Extended Unique Identifier), αποτελεί μοναδιαίο χαρακτηριστικό για τον διακομιστή δικτύου (Network Server). Θα μπορούσε να παρομοιαστεί με την πόρτα που χρησιμοποιείται σε μια σύνδεση TCP/IP socket για να την επικοινωνία με ένα sever.

3. AppKey (Application Key) είναι ένα συμμετρικός κωδικός AES (Advanced Encryption Standard) με μέγεθος 128bit το οποίο επίσης λέγεται και rootkey. Χρησιμοποιείται για τη δημιουργία του integrity code ενός μηνύματος έτσι ώστε να διασφαλίζεται η ακεραιότητα και η αυθεντικότητα του.

Από την άλλη πλευρά ο διακομιστής δικτύου (Network Server) πρέπει να έχει αποθηκευμένο το AppKey που έχει και η τελική συσκευή.

Κατά την μέθοδο OTAA οι συσκευές εκτελούν μια διαδικασία σύνδεσης με το δίκτυο, κατά την οποία εκχωρείται μια δυναμική διεύθυνση συσκευής και διαπραγματεύονται τα κλειδιά ασφαλείας με τη συσκευή.

Θα εξετάσουμε αναλυτικά την μέθοδο ενεργοποίησης OTAA που περιγράφεται γραφικά στις εικόνες 2.29, 2.30 και 2.31, για το πρωτόκολλο LoraWAN έκδοσης 1.02 .



Εικόνα 2.29 – Μέθοδος σύνδεσης OTAA, διαδικασία join-request

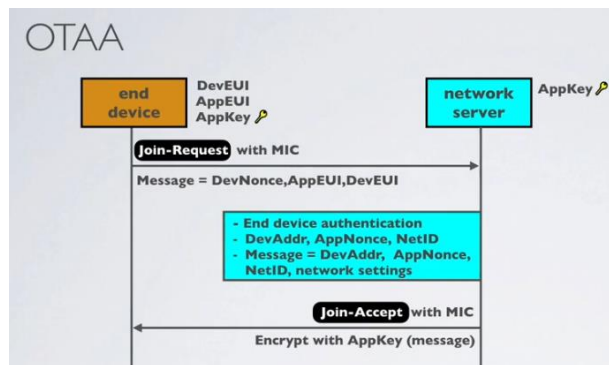
(Πηγή: <https://www.youtube.com/watch?v=KrNDOBzhxeM>)

Αρχικά η τελική συσκευή δημιουργεί ένα τυχαίο κωδικό με ονομασία DevNonce. Στην συνέχεια δημιουργεί ένα μήνυμα join-Request που περιλαμβάνει τον τυχαίο κωδικό DevNonce, και τους δύο κωδικούς που βρίσκονται καταχωρημένοι σε αυτήν AppEUI και DevEUI. Αυτό το join-Request μήνυμα το οποίο δεν είναι κωδικοποιημένο περιλαμβάνει και ένα MIC (Message Integrity Code) για τον έλεγχο της ακεραιότητας το οποίο δημιουργήθηκε από το AppKey που είναι γνωστό στην συσκευή μας.

Μόλις ο διακομιστής δικτύου λάβει το join-request μήνυμα θα εξετάσει εάν ο κωδικός DevNonce έχει χρησιμοποιηθεί από παλαιότερα μηνύματα. Στην συνέχεια θα προχωρήσει στον

έλεγχο της αυθεντικότητας του μηνύματος υπολογίζοντας εκ νέου το MIC (Message Integrity Code) του μηνύματος που έλαβε και συγκρίνοντας τον με το MIC που περιλαμβάνεται στο μήνυμα που έχει λάβει.

Στην περίπτωση όπου και οι δύο MIC είναι ίδιοι τότε ο διακομιστής δικτύου πιστοποιεί την αυθεντικότητα του μηνύματος και προχωράει στην έκδοση του DevAddr (Device Address), του AppNonce και του NetID (Network Identifier).



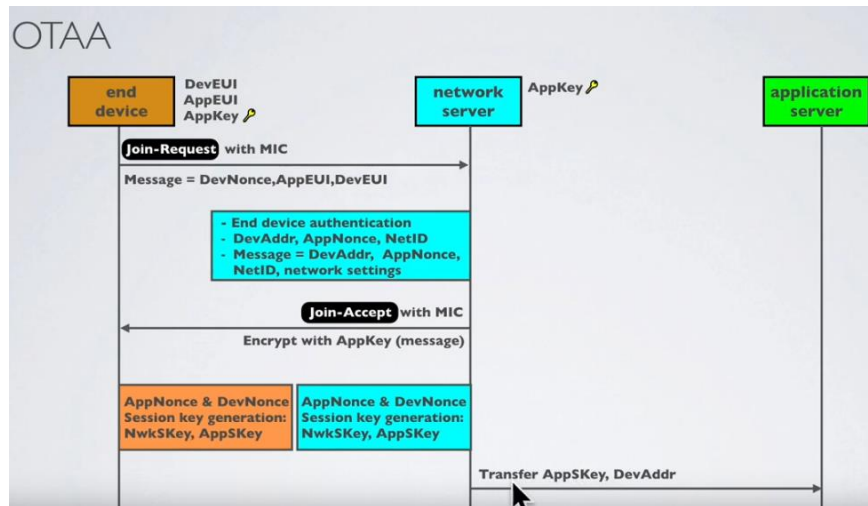
Εικόνα 2.30 – Μέθοδος σύνδεσης OTAA, διαδικασία join-accept (Πηγή: <https://www.youtube.com/watch?v=KrNDOBzhxeM>)

Το DevAddr αποτελεί έναν παραλληλισμό του DevEUI μετασχηματίζοντας το σε μια σύντομη διεύθυνση με μέγεθος 32 bit με σκοπό να μειωθεί το μέγεθος των εκπεμπόμενων παραθύρων. Το DevAddr είναι παρόμοιο με την IP (Internet Protocol) διεύθυνση ενός υπολογιστή ή μιας συσκευής σε ένα δίκτυο υπολογιστών.

Το AppNonce είναι ένας αριθμός που δημιουργείται τυχαία από τον διακομιστή δικτύου και το NetID είναι ένας μοναδιαίος κωδικός που αφορά επίσης τον διακομιστή δικτύου.

Ακολουθεί η έκδοση ενός απαντητικού μηνύματος join-accept όπου συμπεριλαμβάνει τους κωδικούς DevAddr, AppNonce, NetID και ορισμένες παραμέτρους του δικτύου (Network Settings) όπου περιλαμβάνουν το data-rate που χρησιμοποιείται κατά την λήψη των μηνυμάτων, το receive delay που είναι ο χρόνος μεταξύ transmit και receive και μια λίστα συχνοτήτων που δηλώνουν τις συχνότητες του κάθε καναλιού.

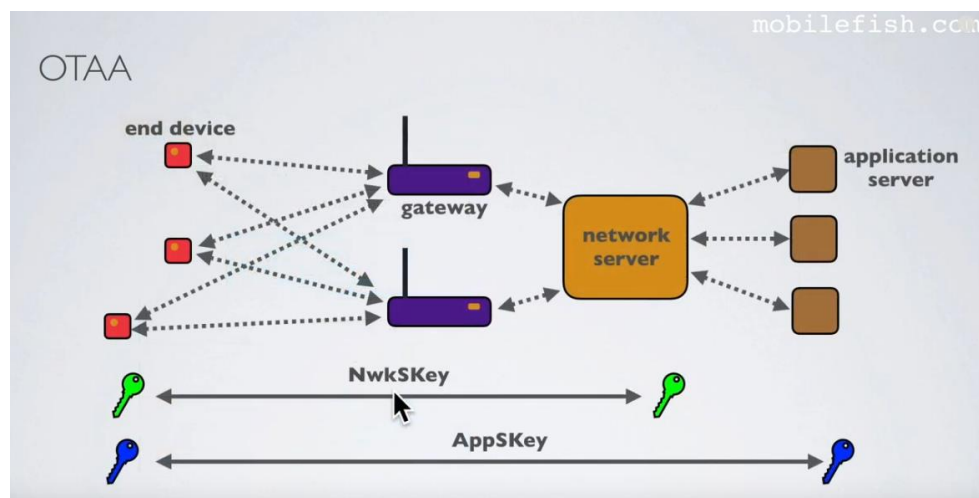
Ακολουθεί η έκδοση του MIC (Message Integrity Code) κάνοντας χρήση του AppKey που είναι γνωστό στον διακομιστή δικτύου και τέλος το μήνυμα κωδικοποιείται με τη χρήση του AppKey και στέλνεται προς την τελική συσκευή μέσω ενός gateway.



Εικόνα 2.31 – Μέθοδος σύνδεσης OTAA, συνολική διαδικασία
(Πηγή: <https://www.youtube.com/watch?v=KrNDOBzhxeM>)

Άρα πλέον η τελική συσκευή και ο διακομιστής δικτύου έχουν από κοινού τα ίδια AppNonce και DevNonce τα οποία χρησιμεύουν και στους δύο για την δημιουργία των κλειδιών συνεδρίας δηλαδή το NwkSKey (Network Session Key) και το AppSKey (Application Session Key).

Τέλος, ο διακομιστής δικτύου μεταφέρει και ενημερώνει τον διακομιστή εφαρμογών για την ύπαρξη των κλειδιών συνεδρίας, τα οποία διασφαλίζουν την από άκρο σε άκρο επικοινωνία μεταξύ τελικής συσκευής και διακομιστή εφαρμογών όπως παρουσιάζεται στην εικόνα 2.32.



Εικόνα 2.32 – Ανταλλαγή κλειδιών μεταξύ τελικών συσκευών και διακομιστών κατά την χρήση της μεθόδου ενεργοποίησης OTAA
(Πηγή: <https://www.youtube.com/watch?v=KrNDOBzhxeM>)

Έτσι υπάρχουν τα δύο ακόλουθα κλειδιά συνεδρίας:

Το NwkSKey (Network Session Key) χρησιμοποιείται από την τελική συσκευή και τον διακομιστή δικτύου και χρησιμεύει στην διασφάλιση της επικοινωνίας. Επίσης, χρησιμοποιείται για την κωδικοποίηση και αποκωδικοποίηση της πληροφορίας (payload) του μηνύματος.

Το AppSKey (Application Session Key) χρησιμοποιείται για την διασφάλιση απ' άκρο σε άκρο της επικοινωνίας μας, δηλαδή από την τελική συσκευή έως τον διακομιστή εφαρμογών.

2.4.8 Activation by Personalization (ABP)

Κατά την χρήση της ενεργοποίησης μιας τελικής συσκευής με την μέθοδο ABP (Activation by personalization) δεν υπάρχουν μηνύματα join-request και join-accept όπως φαίνεται από την εικόνα 2.33. Η τελική συσκευή δεν αποθηκεύει τα DevEUI, AppEUI και AppKey όπως αναλύθηκε προηγουμένως με την χρήση της OTAA. Επίσης, ο διακομιστής δικτύου δεν αποθηκεύει το AppKey καθώς δεν θα χρειαστεί να δημιουργήσει κλειδιά συνεδρίας αφού καταχωρούνται στην τελική συσκευή εξ' αρχής το AppSKey και το NwkSKey, και στον διακομιστή δικτύου το DevAddr και το NwkSKey. Τέλος στον διακομιστή εφαρμογών καταχωρείται το DevAddr και το AppSKey.



Εικόνα 2.33 – Μέθοδος ενεργοποίησης ABP, απαιτούμενα κλειδιά λειτουργίας
(Πηγή: <https://www.youtube.com/watch?v=KrNDOBzhxM>)

Έτσι, όταν μια τελική συσκευή προσπαθεί να επικοινωνήσει με τον διακομιστή δικτύου στέλνει το μήνυμα και η υποδομή του δικτύου έχει ότι χρειάζεται για να μπορέσει να το αποκωδικοποιήσει.

Όλες οι συσκευές που κάνουν χρήση του πρωτοκόλλου LoRaWAN υποχρεούνται να κωδικοποιούν την επικεφαλίδα (header) και το περιεχόμενο του μηνύματος με βάση τον AES (Advanced Encryption Standard) αλγόριθμο χρησιμοποιώντας 128bit encryption. Το LoRaWAN πρωτόκολλο προσφέρει ασφάλεια δύο επιπέδων. Στο επίπεδο δικτύου η ακεραιότητα των δεδομένων διασφαλίζεται από τον κώδικα MIC ο οποίος δημιουργείται μέσω του NwkSKey. Τα δεδομένα που αποστέλλονται είναι κωδικοποιημένα από την τελική συσκευή έως τον διακομιστή δικτύου. Στο επίπεδο εφαρμογής τα δεδομένα κωδικοποιούνται με την χρήση του AppSKey από την τελική συσκευή έως τον διακομιστή εφαρμογών που σημαίνει ότι υπάρχει κωδικοποίηση δεδομένων από άκρη σε άκρη για τα δεδομένα LoRa.

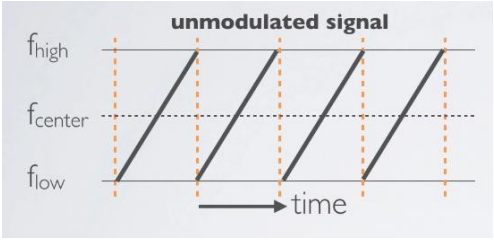
Το ABP είναι λιγότερο ασφαλές από το OTAA και έχει επίσης το μειονέκτημα ότι οι συσκευές δεν μπορούν να αλλάξουν παρόχους δικτύου χωρίς χειροκίνητη αλλαγή των κλειδιών στην συσκευή.

2.4.9 Spreading Factors - Παράγοντες διάδοσης

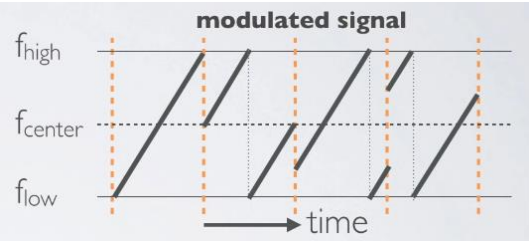
Όπως αναφέρθηκε, το LoRa βασίζεται στην τεχνολογία Chirp Spread Spectrum (CSS), όπου chirps (σύμβολα) είναι ο φορέας δεδομένων.

Τα chirps είναι κυκλικές μετατοπισμένες επιλογές ανάμεσα σε δύο συχνότητες (f_{low} – f_{high}) όπου το κάθε ένα από αυτά χαρακτηρίζεται από ένα κωδικοποιημένο σύμβολο.

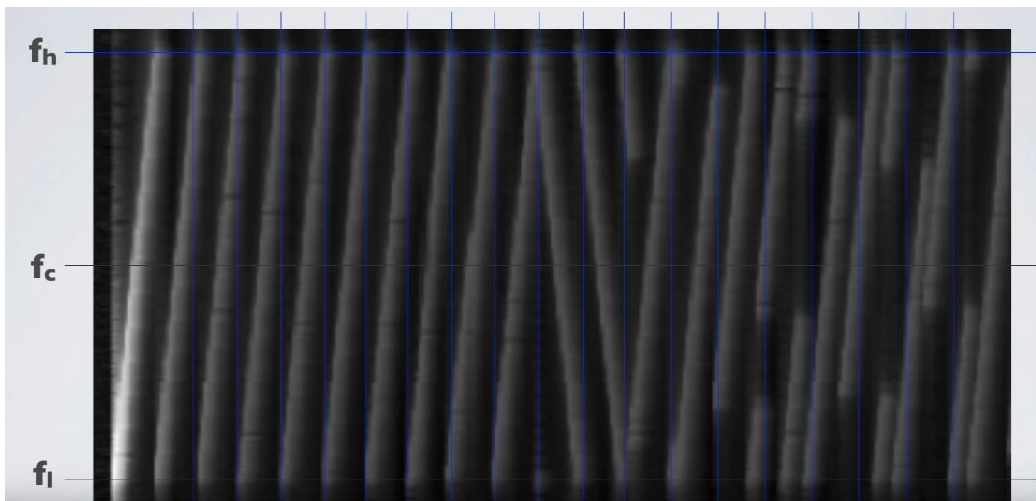
Για να γίνει κατανοητό και σχηματικά αρχικά παρατηρείται γραφικά ένα μη διαμορφωμένο σήμα LoRa που παρουσιάζεται στην εικόνα 2.34.

<p>Μη Διαμορφωμένο σήμα LoRa</p>  <p>Εικόνα 2.34 - Μη Διαμορφωμένο σήμα LoRa</p> <p>(Πηγή: https://www.rfwireless-world.com/Terminology/What-is-difference-between-Chip-and-Chirp-in-LoRaWAN.html)</p>	<p>Αποτελείται από πολλαπλά up-chirps, διαγώνιες γραμμές οι οποίες ξεκινούν από την χαμηλή συχνότητα f_{min} και καταλήγουν στην f_{max} συχνότητα του συνολικού διαθέσιμου bandwidth σε συγκεκριμένο σταθερό χρόνο. Το συγκεκριμένο μη διαμορφωμένο σήμα δεν κουβαλάει δεδομένα και μπορεί να το χαρακτηριστεί ως ένα φέρον σήμα LoRa.</p>
---	---

Στην συνέχεια εμφανίζεται ένα διαμορφωμένο σήμα LoRa στην εικόνα 2.35.

<p>Διαμορφωμένο σήμα LoRa</p>  <p>Εικόνα 2.35 - Διαμορφωμένο σήμα LoRa</p> <p>(Πηγή: https://www.rfwireless-world.com/Terminology/What-is-difference-between-Chip-and-Chirp-in-LoRaWAN.html)</p>	<p>Στο διαμορφωμένο σήμα παρατηρούνται με ευκολία οι κυκλικές αναπηδήσεις (chirps) που συμβαίνουν σε σταθερά χρονικά διαστήματα.</p> <p>Στο πρώτο χρονικό διάστημα εμφανίζεται ένα up-chirp και στην συνέχεια ακολουθούν τρία διαφορετικά chirps όπου κάθε ένα από αυτά αναπαριστά και ένα διαφορετικό σύμβολο.</p>
--	---

Στην εικόνα 2.36 ακολουθεί ένα πραγματικό σήμα LoRa όπως ακριβώς αποτυπώνεται μέσω ενός παλμογράφου.



Εικόνα 2.36 – Σήμα LoRa από παλμογράφο

Πηγή: The Things Network

Αρχικά φαίνονται 10 up-chirps και 2 down-chirps όπου με αυτόν τον τρόπο δηλώνεται η εκκίνηση ενός σήματος LoRa. Στην συνέχεια παρατηρούνται τα διαμορφωμένα δεδομένα του μηνύματος που στέλνει η τελική συσκευή μέσω των chirps που ακολουθούν.

Η παράμετρος που καθορίζει από πόσα bits αποτελείται το κάθε σύμβολο κατά την διαμόρφωση του σήματος είναι ο συντελεστής διασποράς (spreading factor - SF) ο οποίος επίσης μας δείχνει ότι το κάθε σύμβολο αποτελείται από 2^{SF} chips (τιμές). Προσοχή στην διαφορά μεταξύ chirp και chip [22]. Κάθε σύμβολο αποτελείται από διαφορετικές τιμές chips (2^{SF}), ενώ με τον όρο chirps εννοείται απλά μία ανοδική ράμπα από μια χαμηλή συχνότητα flow προς την υψηλή fhigh ή και το αντίθετο.

Η διαμόρφωση LoRa έχει συνολικά 6 παράγοντες διασποράς από SF7 σε SF12 που παρουσιάζονται στον πίνακα 2.5 .

Spreading factor	Chips/Symbol	SNR Limit	Time on air (10 byte packet)	Data Rate Bit Rate
7	128	-7.5	56 ms	5469 bps
8	256	-10	103	3125
9	512	-12.5	205	1758
10	1024	-15	371	977
11	2048	-17.5	741	537
12	4096	-20	1483	293

Πίνακας 2.5 – Παράγοντες διάδοσης ανά συντελεστή διασποράς

(Πηγή: <https://www.rfwireless-world.com/Terminology/What-is-difference-between-Chip-and-Chirp-in-LoRaWAN.html>)

Το δίκτυο χρησιμοποιεί επίσης τον συντελεστή διασποράς για τον έλεγχο της συμφόρησης. Τα σήματα που διαμορφώνονται με διαφορετικούς συντελεστές διασποράς και μεταδίδονται στο ίδιο κανάλι την ίδια στιγμή δεν παρεμβαίνουν(παρεμβάλλονται) μεταξύ τους.

Οι παράγοντες διάδοσης επηρεάζουν τον ρυθμό δεδομένων, το χρόνο μετάδοσης, τη διάρκεια ζωής της μπαταρίας και την ευαισθησία του δέκτη [23].

- **Ρυθμός δεδομένων - Data rate ή bit rate**

Ο συντελεστής διασποράς (spreading factor) ελέγχει τον ρυθμό chips/Symbol και κατά συνέπεια ελέγχει την ταχύτητα μετάδοσης δεδομένων. Χαμηλότεροι συντελεστές διασποράς σημαίνουν ταχύτερα chips και συνεπώς υψηλότερο ρυθμό μετάδοσης δεδομένων (bit rate).

$$\text{Το Data Rate ή αλλιώς bit Rate } R_b \text{ (bits / sec) ισούται με } SF \cdot \frac{4}{\frac{2^{SF}}{BW}} = \frac{4 \cdot SF \cdot BW}{2^{SF} (4 + CR)}$$

$$\text{όπου } \frac{2^{SF}}{BW} = \text{Symbol duration} = T_s \text{ (sec),}$$

και CR = ρυθμός κωδικοποίησης των bits που αφορούν ενεργή πληροφορία (payload).

Σε σύγκριση με έναν υψηλότερο συντελεστή διασποράς (sf), ένας χαμηλότερος συντελεστής διασποράς παρέχει υψηλότερο ρυθμό μετάδοσης bit για σταθερό εύρος ζώνης (BW) και ρυθμό κωδικοποίησης (CR) όπως φαίνεται άλλωστε και από τον πίνακα 4. Αυτό είναι λογικό διότι με την αύξηση του sf ο παρονομαστής του κλάσματος R_b μεγαλώνει πιο γρήγορα από τον αριθμητή.

Στον ακόλουθο πίνακα 2.6 παρατηρείται η μεταβολή του ρυθμού δεδομένων για τιμές εύρους ζώνης, 125, 250 και 500 kHz.

Spreading Factor	Bandwidth	Bit rate (kbits/s)
7	125	5.5
7	250	10.9
7	500	21.9

Πίνακας 2.6 – Ρυθμός μετάδοσης σε σχέση με το εύρος ζώνης (bandwidth)

(Πηγή: <https://www.thethingsnetwork.org/docs/lorawan/spreading-factors/>)

Ο διπλασιασμός του εύρους ζώνης σχεδόν διπλασιάζει επίσης τον ρυθμό μετάδοσης bit για σταθερό συντελεστή διασποράς (sf) και σταθερό ρυθμό κωδικοποίησης (CR).

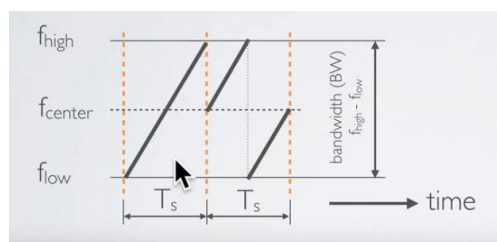
- **Distance**

Μεγαλύτεροι συντελεστές διασποράς σημαίνουν ότι η επεξεργασία των δεδομένων του μηνύματος γίνεται με περισσότερα bits άρα και περισσότερες τιμές (chips), άρα ένα σήμα που διαμορφώνεται με μεγαλύτερο συντελεστή διασποράς είναι πιο πλούσιο και συνεπώς σε ενδεχόμενες παρεμβολές μπορεί να ληφθεί με λιγότερα σφάλματα σε σύγκριση με ένα σήμα με χαμηλότερο συντελεστή διασποράς. Συνεπώς, μπορεί να διανύσει και μεγαλύτερη απόσταση χωρίς να υπάρξουν προβλήματα αλλοίωσης του σήματος.

- **Time-On-Air – Χρόνος Εκπομπής**

Σε σύγκριση με έναν χαμηλότερο συντελεστή διασποράς, η αποστολή σταθερού όγκου δεδομένων (ωφέλιμο φορτίο) με υψηλότερο συντελεστή διασποράς και σταθερό εύρος ζώνης απαιτεί μεγαλύτερο χρόνο μετάδοσης.

Η χρονική διάρκεια T_s ενός σήματος φαίνεται στην εικόνα 2.37:



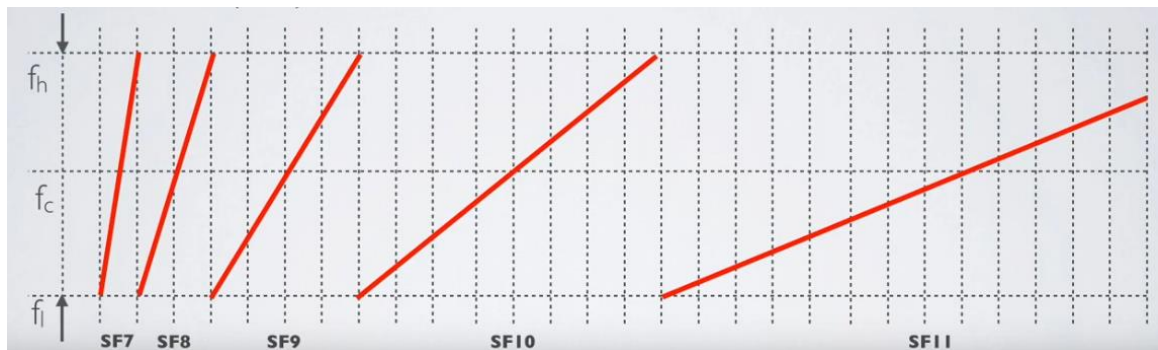
Εικόνα 2.37 - Χρονική διάρκεια συμβόλου LoRa

(Πηγή: <https://www.rfwireless-world.com/Terminology/What-is-difference-between-Chip-and-Chirp-in-LoRaWAN.html>)

Αρχικά η μονάδα μέτρησης του Bandwidth είναι τα Hz (κύκλοι/sec). Το bandwidth εναλλακτικά για τα σήματα Lora με διαμόρφωση Chirp Spread Spectrum (CSS) είναι το chip rate CR (chips/sec) δηλαδή οι τιμές που λαμβάνει ανά δευτερόλεπτο. Δηλαδή, $BW = R_c = \text{chip rate (chips/sec)}$.

$$\text{Symbol duration } T_s \text{ (sec)} = \frac{2^{SF}}{BW}$$

Στην εικόνα 2.38 εμφανίζεται το symbol duration σε σχέση με διαφορετικές τιμές του spreading factor.



Εικόνα 2.38 - Χρονική διάρκεια συμβόλου με βάση τον συντελεστή διασποράς (sf)

(Πηγή: <https://www.rfwireless-world.com/Terminology/What-is-difference-between-Chip-and-Chirp-in-LoRaWAN.html>)

Παρατηρείται ότι εάν το spreading factor αυξηθεί κατά 1 τότε το symbol duration διπλασιάζεται.

Η αλλαγή του συντελεστή διασποράς επιτρέπει στο δίκτυο να αυξάνει ή να μειώνει τον ρυθμό δεδομένων για κάθε τελική συσκευή θυσιάζοντας βέβαια ένα μέρος της εμβέλειας λειτουργίας του.

Receiver Sensitivity – Ευαισθησία Δέκτη

Οι υψηλότεροι συντελεστές διασποράς παρέχουν υψηλότερη ευαισθησία του δέκτη. Συνήθως, το LoRa χρησιμοποιεί υψηλότερους συντελεστές διασποράς όταν το σήμα είναι αδύναμο.

Ο παρακάτω πίνακας 2.7 δείχνει πώς οι παράγοντες διασποράς επηρεάζουν την ευαισθησία του δέκτη για σταθερό bandwidth σήματος.

Spreading factor	Receiver sensitivity for bandwidth fixed at 125 kHz
SF7	-123 dBm
SF8	-126 dBm
SF9	-129 dBm
SF10	-132 dBm
SF11	-134.5 dBm
SF12	-137 dBm

Πίνακας 2.7 – Ευαισθησία δέκτη με σταθερό bandwidth για διαφορετικό συντελεστή διασποράς

(Πηγή: <https://www.thethingsnetwork.org/docs/lorawan/spreading-factors/>)

Battery Life – Χρόνος διάρκειας μπαταρίας

Η διάρκεια ζωής της μπαταρίας μιας τελικής συσκευής εξαρτάται σε μεγάλο βαθμό από τον παράγοντα διασποράς που χρησιμοποιείται. Οι υψηλότεροι συντελεστές διασποράς έχουν ως αποτέλεσμα μεγαλύτερους χρόνους ενεργοποίησης για τους πομποδέκτες και συνεπώς μικρότερη διάρκεια ζωής της μπαταρίας.

2.4.10 Προσαρμοστικός αλγόριθμος διάδοσης (ADR – Adaptive Data Rate)

Το Adaptive Data Rate (ADR) αποτελεί έναν προσαρμοστικό αλγόριθμο βελτιστοποίησης του ρυθμού δεδομένων, του χρόνου που διαρκεί ένα μήνυμα για να φτάσει από την τελική συσκευή στο δίκτυο (χρόνος εκπομπής) καθώς επίσης και της κατανάλωσης ενέργειας.

Ο μηχανισμός ADR ελέγχει τις ακόλουθες παραμέτρους μετάδοσης μιας τελικής συσκευής.

- Συντελεστής διασποράς - Spreading factor
- Εύρος ζώνης μετάδοσης - Bandwidth
- Ισχύς μετάδοσης

Σκοπός του ADR είναι να βελτιστοποιήσει την κατανάλωση ενέργειας μίας τελικής συσκευής διασφαλίζοντας παράλληλα ότι τα μηνύματα εξακολουθούν να λαμβάνονται στις πύλες (gateways). Έτσι, κάνοντας χρήση του ADR, ο διακομιστής δικτύου με βάση τα μηνύματα που λαμβάνει από τις τελικές συσκευές, θα είναι σε θέση να υποδείξει προς αυτές εάν πρέπει να αυξομειώσουν την ισχύ μετάδοσης ή αντίστοιχα να μεταβάλλουν τον ρυθμό δεδομένων έτσι ώστε να διασφαλιστεί η ορθή λήψη των μηνυμάτων από τις πύλες καταναλώνοντας την ελάχιστη δυνατή ενέργεια από την συνολική που διαθέτουν.

Γενικότερα οι τελικές συσκευές που βρίσκονται κοντά στις πύλες (gateways) του δικτύου, θα πρέπει να χρησιμοποιούν χαμηλότερο συντελεστή διασποράς άρα υψηλότερο ρυθμό δεδομένων, ενώ οι συσκευές που βρίσκονται πιο μακριά θα πρέπει να χρησιμοποιούν υψηλότερο συντελεστή διασποράς καθότι θα πρέπει να εκπέμπουν με δυνατότερο σήμα.

Ο αλγόριθμος ADR θα πρέπει να ενεργοποιείται για τις στατικές συσκευές που έχουν σταθερές συνθήκες εκπομπής. Εάν γίνει χρήση του ADR για συσκευές οι οποίες βρίσκονται σε κίνηση υπάρχουν δύο σημαντικά προβλήματα α) επειδή η βελτιστοποίηση γίνεται σε δύο χρόνους, δηλαδή πρώτα εξετάζονται τα δεδομένα από τις 10 προηγούμενες λήψεις και στην συνέχεια αποστέλλεται μια βελτιωτική ενέργεια προς την τελική συσκευή, κατά την λήψη αυτής της βελτιωτικής ενέργειας τα δεδομένα θα έχουν αλλάξει και πάλι λόγω αλλαγής θέσης της συσκευής, β) εάν η τελική συσκευή βρίσκεται συνέχεια σε κίνηση τότε ο αλγόριθμος θα είναι μόνιμα ενεργοποιημένος προσπαθώντας να βελτιστοποιήσει τα αποτελέσματα και έτσι σε ένα δίκτυο με χιλιάδες αισθητήρες εν κινήσει η επεξεργαστική ισχύς που θα απαιτούνταν από την πλευρά του δικτύου θα ήταν πολύ μεγάλη.

Συνεπώς, σωστό είναι να απενεργοποιείται ο αλγόριθμος ADR από τις ίδιες τις τελικές συσκευές όταν βρίσκονται σε κίνηση και να ενεργοποιείται εκ νέου όταν είναι σταθερές.

3. Υλοποίηση του Συστήματος

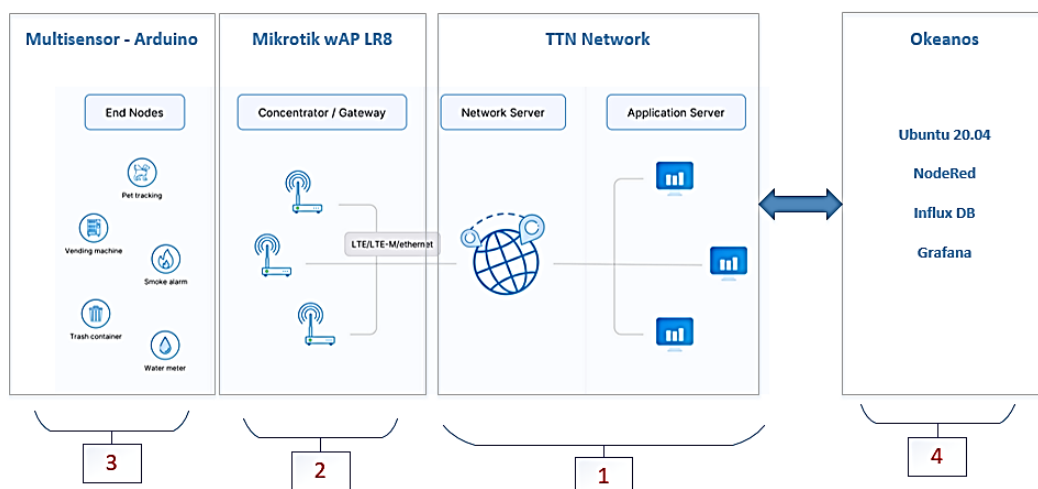
3.1 Εισαγωγή κεφαλαίου 3

Το κεφάλαιο αυτό ασχολείται με την εφαρμογή της τεχνολογίας LoRa στο χώρο της γεωργίας. Πιο συγκεκριμένα κατασκευάζεται ένα σύστημα το οποίο μπορεί να επιτηρεί διάφορες παραμέτρους μιας γεωργικής καλλιέργειας με απώτερο σκοπό την βελτιστοποίηση της απόδοσης παραγωγής καθώς επίσης την προστασία σε αντίξοες καιρικές συνθήκες.

Για να επιτευχθεί αυτό, διάφορα μετρήσιμα δείγματα από διαφορετικές παραμέτρους όπως θερμοκρασία αέρα, θερμοκρασία εδάφους, υγρασία εδάφους, ατμοσφαιρική πίεση, φωτεινότητα ατμόσφαιρας, λαμβάνονται ανά τακτά χρονικά διαστήματα από διάφορους αισθητήρες και στην συνέχεια αποστέλλονται μέσω ενός δρομολογητή (gateway) που βρίσκεται σε κοντινή περιοχή εντός της εμβέλειας των αισθητήρων, προς ένα δικτυακό νέφος (cloud). Εκεί πραγματοποιείται επεξεργασία και αποκωδικοποίηση των δεδομένων αυτών. Το διαδικτυακό νέφος που χρησιμοποιείται είναι το TTN (The Things Network) το οποίο αναπτύχθηκε για IoT εφαρμογές και λειτουργεί με βάση την τεχνολογία LoRa.

Τέλος, για να είναι άμεσα κατανοητά τα δεδομένα που λαμβάνονται από τους αισθητήρες προς τους χειριστές του συστήματος, μεταφέρονται μέσω API διεπαφής προς έναν virtual server που διατίθεται η εφαρμογή ανοικτού κώδικα “NodeRed”. Εκεί πραγματοποιείται αρχικά η οπτικοποίηση των δεδομένων αλλά επίσης και η επεξεργασία τους για τυχόν ενέργειες που απαιτούνται με σκοπό την βελτιστοποίηση της παραγωγής. Στην περίπτωση που προκύψει ενδεχόμενο συγκεκριμένης ενέργειας, αυτόματα καθοδικά μηνύματα είναι δυνατόν να αποσταλούν από το περιβάλλον της εφαρμογής προς ορισμένους ενεργοποιητές του συστήματος.

Μια σχηματική απεικόνιση του συστήματος φαίνεται στην εικόνα 3.1.



Εικόνα 3.1 – Βηματική παρουσίαση συνολικής κατασκευής

Τα νούμερα με κόκκινο χρώμα στο κάτω μέρος της παραπάνω εικόνας υποδηλώνουν την σειρά των απαραίτητων βημάτων με σκοπό την ανάπτυξη της κατασκευής.

Έτσι, το πρώτο βήμα αφορά τις ενέργειες που πρέπει να γίνουν στο TTN ώστε να μπορεί να δέχεται πακέτα δεδομένων μέσω ενός δρομολογητή LoRa και στην συνέχεια αυτά τα πακέτα να καταλήγουν σε μία εφαρμογή με ένα συγκεκριμένο format.

Το δεύτερο βήμα αφορά τις ρυθμίσεις που πρέπει να γίνουν στον δρομολογητή έτσι ώστε να λαμβάνονται και να στέλνονται πακέτα δεδομένων προς το TTN.

Το τρίτο βήμα αφορά τις τελικές συσκευές. Διάφοροι αισθητήρες μετρήσεων προστίθενται σε έναν μικροελεγκτή μαζί με ένα module ασύρματης εκπομπής και λήψης. Με αυτόν τον τρόπο δημιουργείται ένας πολυαισθητήρας.

Παράλληλα, χρησιμοποιείται ένας δεύτερος μικροελεγκτής όπου σκοπός του είναι να διαχειρίζεται τους ενεργοποιητές του συστήματος μέσω ασύρματης επικοινωνίας.

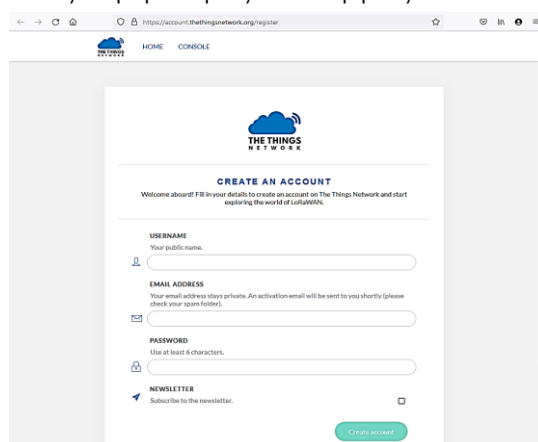
Τέλος το τέταρτο βήμα αφορά την οπτικοποίηση των δεδομένων που λαμβάνονται από τις τελικές συσκευές και την αυτοματοποίηση ορισμένων λειτουργιών.

3.2 Σύνδεση με το διαδίκτυο των πραγμάτων TTN – Things of Network

3.2.1 Διαδικασία εγγραφής στο TTN

Αυτή η ενότητα ασχολείται με το πρώτο βήμα δηλαδή με τις απαραίτητες ρυθμίσεις που πρέπει να γίνουν από την πλευρά του TTN έτσι ώστε να είναι έτοιμο να δεχθεί δεδομένα από τις τελικές συσκευές.

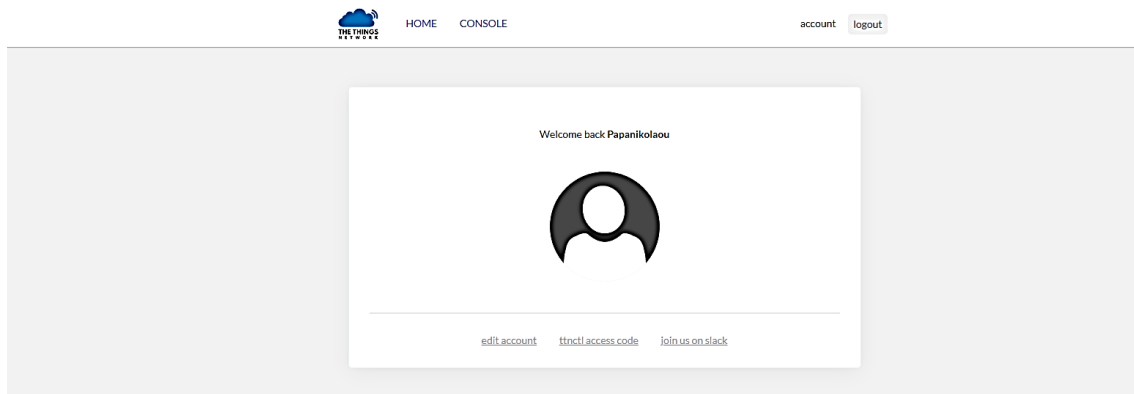
Αρχικά δημιουργείται ένας λογαριασμός στο TTN μέσω της σελίδας <https://www.thethingsnetwork.org/> επιλέγοντας την διαδικασία register από το βασικό μενού. Αμέσως μετά, εμφανίζεται η ακόλουθη φόρμα που παρουσιάζεται γραφικά στην εικόνα 3.2, στην οποία συμπληρώνοντας τα απαραίτητα στοιχεία και επιλέγοντας 'create Account' δημιουργείται ένας λογαριασμός λειτουργίας.

The image shows a web browser window displaying the 'CREATE AN ACCOUNT' page on the Things Network website. The page has a white background with a blue header containing the TTN logo and navigation links for 'HOME' and 'CONSOLE'. The main content area contains a form with the following fields: 'USERNAME' (with a person icon and the instruction 'Your public name.'), 'EMAIL ADDRESS' (with an envelope icon and the instruction 'Your email address stays private. An activation email will be sent to you shortly (please check your spam folder).'), and 'PASSWORD' (with a key icon and the instruction 'Use at least 8 characters.'). Below these fields is a 'NEWSLETTER' section with a checked checkbox and the text 'Subscribe to the newsletter.'. At the bottom right of the form is a green 'Create account' button.

Εικόνα 3.2 – Δημιουργία λογαριασμού TTN

Πηγή: The Things Network

Αμέσως μετά την δημιουργία ενός λογαριασμού από το TTN εμφανίζεται η οθόνη που φαίνεται στην εικόνα 3.3. Παράλληλα, ζητείται να γίνει επιβεβαίωση της διεύθυνσης του ηλεκτρονικού ταχυδρομείου (email) που καταχωρήθηκε έτσι ώστε να γίνει η απαιτούμενη ενεργοποίηση του λογαριασμού.



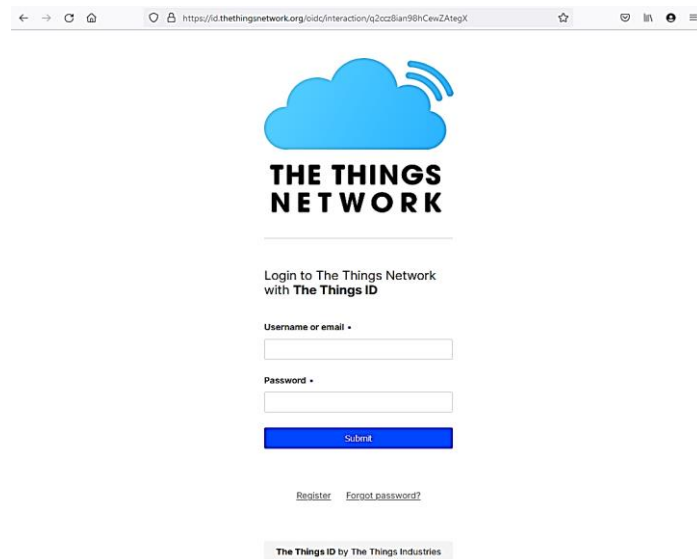
Εικόνα 3.3 – Επιτυχής ολοκλήρωση δημιουργίας νέου λογαριασμού στο TTN
Πηγή: The Things Network

Επόμενο βήμα είναι η επιλογή διακομιστή TTN στον οποίο πραγματοποιείται η επεξεργασία των δεδομένων. Η συγκεκριμένη διαδικασία υλοποιείται από το κεντρικό μενού μέσω της επιλογής 'Console'. Υπάρχουν συνολικά τρεις διαφορετικές επιλογές που παρουσιάζονται στην εικόνα 3.4, Ευρώπη, Αμερική και Αυστραλία. Επιλέχτηκε ο διακομιστής που βρίσκεται στην Ευρώπη ως βέλτιστη επιλογή καθότι, βρίσκεται σε κοντινότερη απόσταση από την τοποθεσία που πρόκειται να λειτουργήσουν οι τελικές συσκευές.



Εικόνα 3.4 – Επιλογή τοποθεσίας διακομιστή στο TTN
Πηγή: The Things Network

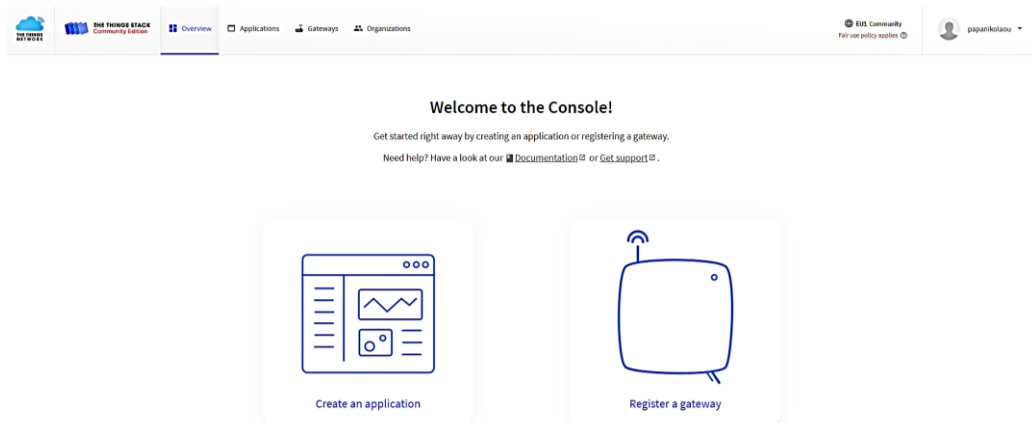
Μετά την επιλογή του server το σύστημα ζητάει να γίνει εισαγωγή των στοιχείων του λογαριασμού που μόλις δημιουργήθηκε μέσω της οθόνης 3.5.



Εικόνα 3.5 – Οθόνη εισαγωγής (login) στο TTN

Πηγή: The Things Network

Αμέσως μετά την εισαγωγή εμφανίζεται η οθόνη που φαίνεται στην εικόνα 3.6 όπου μας δίνεται η δυνατότητα να επιλέξουμε ανάμεσα στην καταχώρηση ενός gateway ή την δημιουργία μιας νέας εφαρμογής.



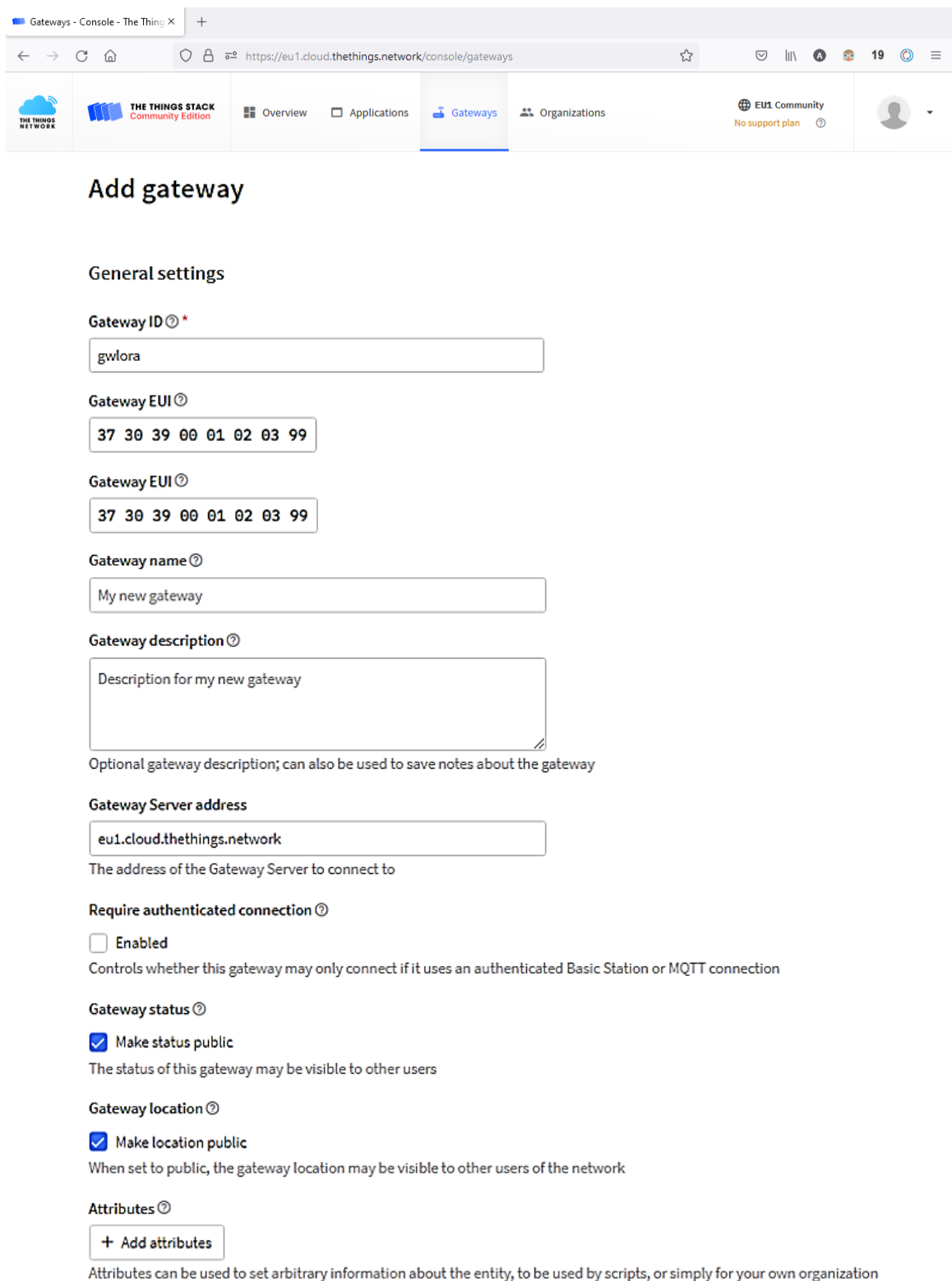
Εικόνα 3.6 – Επιλογή δημιουργίας εφαρμογής ή δρομολογητή

Πηγή: The Things Network

Αρχικά επιλέγεται “Register a gateway” και αφού ολοκληρωθεί η καταχώρηση του δρομολογητή ακολουθεί η διαδικασία ‘Create an application’ για την δημιουργία μιας εφαρμογής.

3.2.2 Διαδικασία καταχώρησης δρομολογητή στο TTN

Κατά την διαδικασία καταχώρησης ενός δρομολογητή στο δίκτυο TTN (v3) εμφανίζεται η οθόνη που φαίνεται στις εικόνες 3.7 (α),(β),(γ).



The screenshot shows the 'Add gateway' form in the TTN console. The form includes the following fields and options:

- Gateway ID**: gwlor
- Gateway EUI**: 37 30 39 00 01 02 03 99
- Gateway name**: My new gateway
- Gateway description**: Description for my new gateway
- Gateway Server address**: eu1.cloud.thethings.network
- Require authenticated connection**: Enabled
- Gateway status**: Make status public
- Gateway location**: Make location public
- Attributes**: + Add attributes

Εικόνα 3.7(α)– Οθόνη καταχώρησης δρομολογητή

Πηγή: The Things Network

Γενικές ρυθμίσεις

Στην εικόνα 3.7 (α) ζητούνται οι γενικές ρυθμίσεις που αφορούν τον δρομολογητή και αναλύονται ακολούθως:

- **Gateway ID** : Μοναδικό αναγνωριστικό το οποίο πρέπει να αποτελείται από μικρά λατινικά γράμματα, αριθμούς και παύλες.
- **Gateway EUI** : Ένα μοναδικό αναγνωριστικό 64-bit όπου καθορίζεται από την κατασκευάστρια εταιρεία του δρομολογητή.
- **Gateway name** : Μη υποχρεωτικό περιγραφικό πεδίο που δύναται να βοηθήσει στην διάκρισή μεταξύ καταχωρημένων δρομολογητών.
- **Gateway description** : Καθαρά περιγραφικό πεδίο, μη υποχρεωτικό.
- **Gateway Server address**: Πρόκειται για την διεύθυνση του server που θα χρησιμοποιείται από την συγκεκριμένη συσκευή έτσι ώστε να μπορεί να επικοινωνεί με το TTN. Συνήθως, το πεδίο αυτό συμπληρώνεται αυτόματα με την διεύθυνση του διακομιστή που επιλέχθηκε σε προηγούμενο βήμα κατά την διαδικασία της καταχώρησης.
- **Require authenticated connection (Enable/Disable)**: Επιβάλλει ασφαλή σύνδεση μεταξύ δρομολογητή και δικτύου TTN.
- **Gateway status (Make status public – Enable/Disable)**: Επιτρέπει σε όλους τους χρήστες του δικτύου TTN να μπορούν να κάνουν χρήση του συγκεκριμένου gateway.
- **Gateway location (Make location public – Enable/Disable)**: Επιτρέπει σε όλους τους χρήστες του δικτύου TTN να έχουν πρόσβαση στις πληροφορίες τοποθεσίας του δρομολογητή.

LoRaWAN options

Frequency plan ⓘ *

 | v

Schedule downlink late ⓘ

Enabled

Enable server-side buffer of downlink messages

Enforce duty cycle ⓘ

Enabled

Recommended for all gateways in order to respect spectrum regulations

Schedule any time delay ⓘ *

 | v

Configure gateway delay (minimum: 130ms, default: 530ms)

Εικόνα 3.7(β) – Οθόνη καταχώρησης δρομολογητή

Πηγή: The Things Network

Ρυθμίσεις LoRaWAN

Αντίστοιχα στην εικόνα 3.7 (β) ζητούνται οι ρυθμίσεις που αφορούν το πρωτόκολλο LoRaWAN του δρομολογητή και αναλύονται ακολούθως:

- **Frequency plan (Dropdown):** Επιλέγεται το εύρος συχνοτήτων λειτουργίας του δρομολογητή.
- **Schedule downlink late (Enable/Disable) :** Η συγκεκριμένη ρύθμιση αφορά ορισμένους δρομολογητές που δεν διαθέτουν buffer για τα κατερχόμενα μηνύματα. Σε αυτήν την περίπτωση πρέπει να τίθεται σε ενεργή κατάσταση.
- **Enforce duty cycle (Enable/Disable) :** Όταν τίθεται σε ενεργή κατάσταση, ο διακομιστής δικτύου εφαρμόζει τους περιορισμούς που ισχύουν από τον νόμο για το εύρος συχνοτήτων λειτουργίας που επιλέχθηκαν στην ρύθμιση frequency plan.
- **Schedule anytime delay :** Η συγκεκριμένη ρύθμιση αφορά χρονική καθυστέρηση και πρέπει να ενεργοποιείται μόνο όταν γίνεται χρήση τελικών συσκευών class C. Το δίκτυο που παρεμβαίνει μεταξύ gateway και δικτύου TTN παρουσιάζει μια εμφανή καθυστέρηση (latency).



Gateway updates

Automatic updates

Enabled
Gateway can be updated automatically

Channel

Stable

Channel for gateway automatic updates

Create gateway

Εικόνα 3.7(γ) – Οθόνη καταχώρησης δρομολογητή (γ)

Πηγή: The Things Network

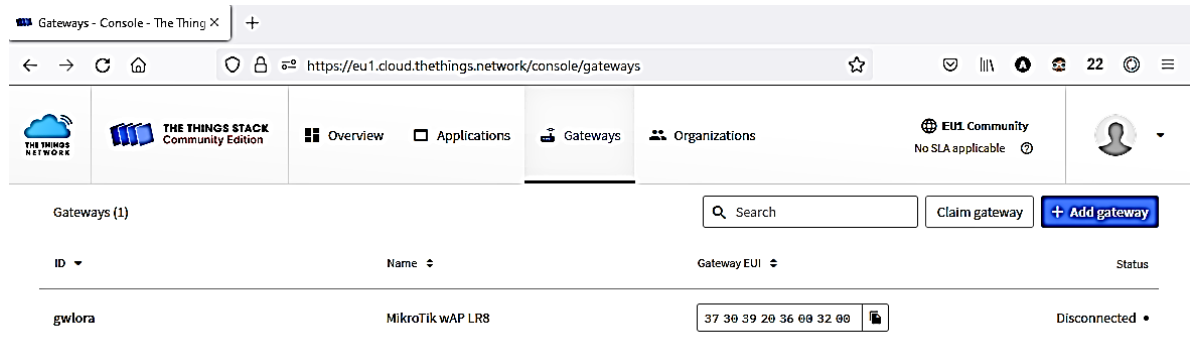
Ρυθμίσεις ενημέρωσης δρομολογητή

Στην εικόνα 3.7(γ) παρουσιάζονται οι ρυθμίσεις ενημέρωσης δρομολογητή και αναλύονται ακολούθως:

- **Automatic updates (enable/disable) :** Όταν ενεργοποιείται η συγκεκριμένη ρύθμιση ο δρομολογητής ενημερώνεται αυτόματα υπό την προϋπόθεση ότι η συγκεκριμένη λειτουργία υποστηρίζεται από τον ίδιο τον δρομολογητή (gateway).
- **Channel :** Επιλέγεται το κανάλι μέσω του οποίου πραγματοποιείται η αυτόματη ενημέρωση.

Μόλις συμπληρωθούν οι παραπάνω πληροφορίες μέσω της επιλογής “Create gateway” ολοκληρώνεται η καταχώρηση του δρομολογητή στο δίκτυο TTN.

Με την ολοκλήρωση της καταχώρησης το σύστημα εμφανίζει την οθόνη που φαίνεται στην εικόνα 3.8, όπου παρουσιάζονται ορισμένα βασικά στοιχεία από τους καταχωρημένους δρομολογητές καθώς και η τρέχουσα κατάσταση (status) στην οποία βρίσκονται.



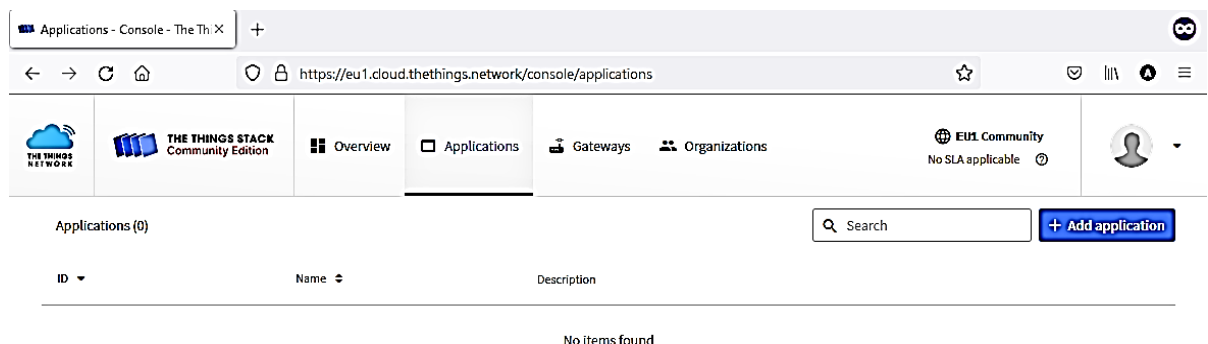
Εικόνα 3.8 – Λίστα καταχωρημένων δρομολογητών στο TTN

Πηγή: The Things Network

3.2.3 Δημιουργία εφαρμογών στο TTN

Οι εφαρμογές αποτελούν το χώρο που καταλήγουν τα δεδομένα από τις τελικές συσκευές μετά την αποδιαμόρφωση και την αποκωδικοποίηση τους χωρίς όμως να πραγματοποιείται η αποθήκευση των δεδομένων αυτών.

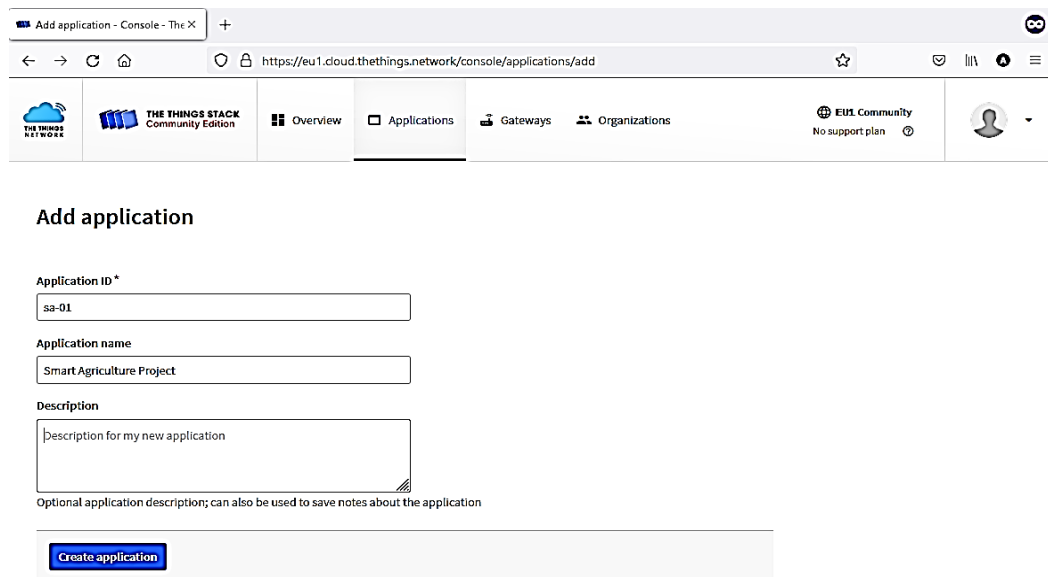
Επιλέγοντας “Overview” από το κεντρικό μενού εμφανίζεται η οθόνη επιλογής ανάμεσα σε δρομολογητή ή εφαρμογή που είδαμε στην Εικόνα 3.6. Στην συνέχεια επιλέγοντας “Create an application” εμφανίζεται η οθόνη που βλέπουμε στην εικόνα 3.9 η οποία αποτελεί μια λίστα που περιλαμβάνει τις καταχωρημένες εφαρμογές που υπάρχουν ήδη στο σύστημα. Στην παρούσα περίπτωση δεν υπάρχουν καταχωρημένες εφαρμογές οπότε εμφανίζεται ‘No items found’.



Εικόνα 3.9 – Δημιουργία εφαρμογής στο TTN

Πηγή: The Things Network

Επιλέγοντας “Add application” στην οθόνη της Εικόνας 3.9 ξεκινάει η διαδικασία προσθήκης μιας νέας εφαρμογής. Τα στοιχεία που απαιτούνται για την δημιουργία της εφαρμογής είναι αυτά που φαίνονται στην Εικόνα 3.10.

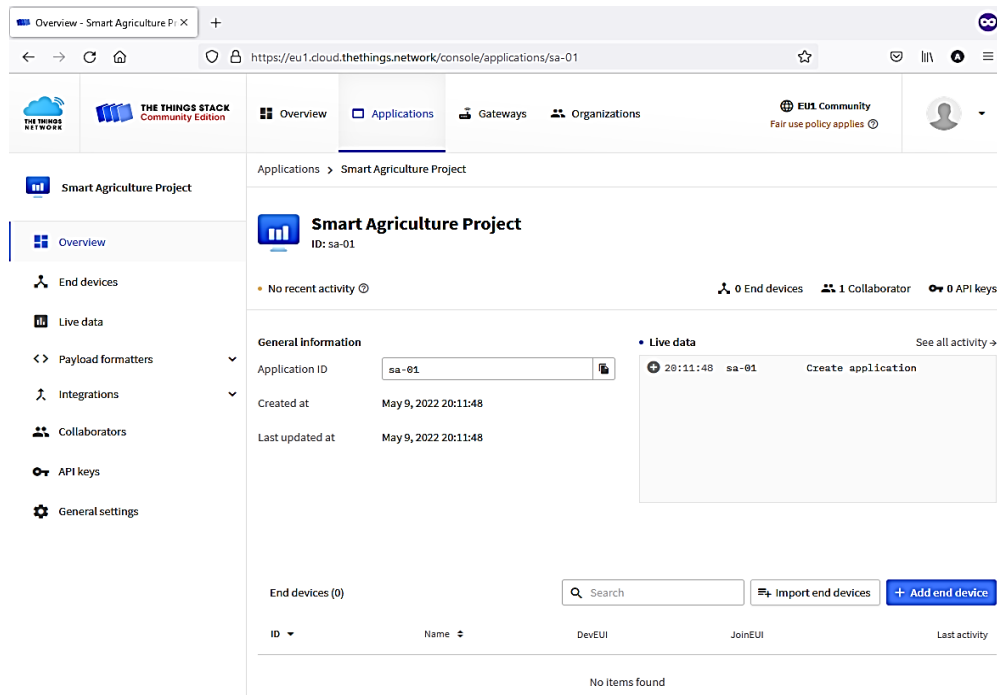
The image shows a web browser window with the URL https://eu1.cloud.thethings.network/console/applications/add. The page title is "Add application - Console - The X". The browser's address bar shows the URL. The page has a navigation bar with "THE THINGS STACK Community Edition", "Overview", "Applications", "Gateways", and "Organizations". There is also a "EUI Community No support plan" link and a user profile icon. The main content area is titled "Add application" and contains three input fields: "Application ID*" with the value "sa-01", "Application name" with the value "Smart Agriculture Project", and "Description" with the value "Description for my new application". Below the description field is a note: "Optional application description; can also be used to save notes about the application". At the bottom of the form is a blue button labeled "Create application".

Εικόνα 3.10 – Δημιουργία εφαρμογής στο TTN

Πηγή: The Things Network

- **Application ID:** Χαρακτηρίζει την εφαρμογή και μπορεί να αποτελείται μόνο από μικρούς αλφαριθμητικούς χαρακτήρες και παύλες.
- **Application name:** Αποτελεί το όνομα της εφαρμογής που πρόκειται να δημιουργηθεί.
- **Description:** Πρόκειται για ένα καθαρά περιγραφικό πεδίο που βοηθάει στην περιγραφή της εφαρμογής.

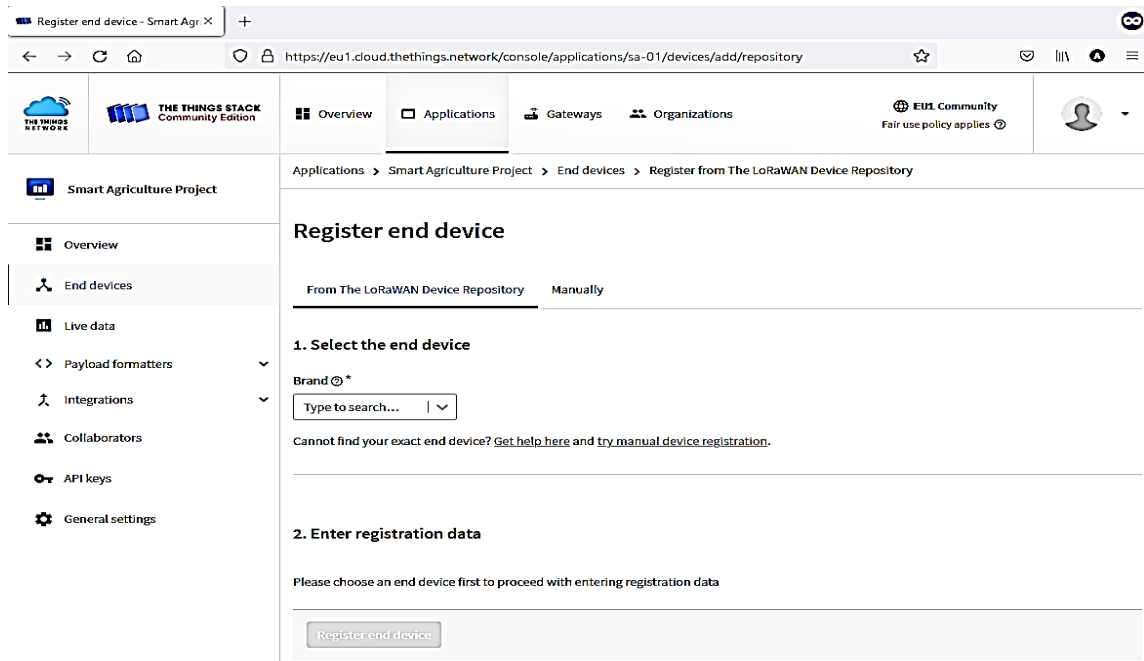
Επιλέγοντας “Create application” δημιουργείται η εφαρμογή και εμφανίζεται η οθόνη της εικόνας 3.11, η οποία περιλαμβάνει βασικές πληροφορίες για την εφαρμογή μαζί με μια λίστα τελικών συσκευών που ανήκουν σε αυτήν.



Εικόνα 3.11 – Πληροφορίες εφαρμογής
Πηγή: The Things Network

3.2.4 Δημιουργία Τελικής Συσκευής στο TTN

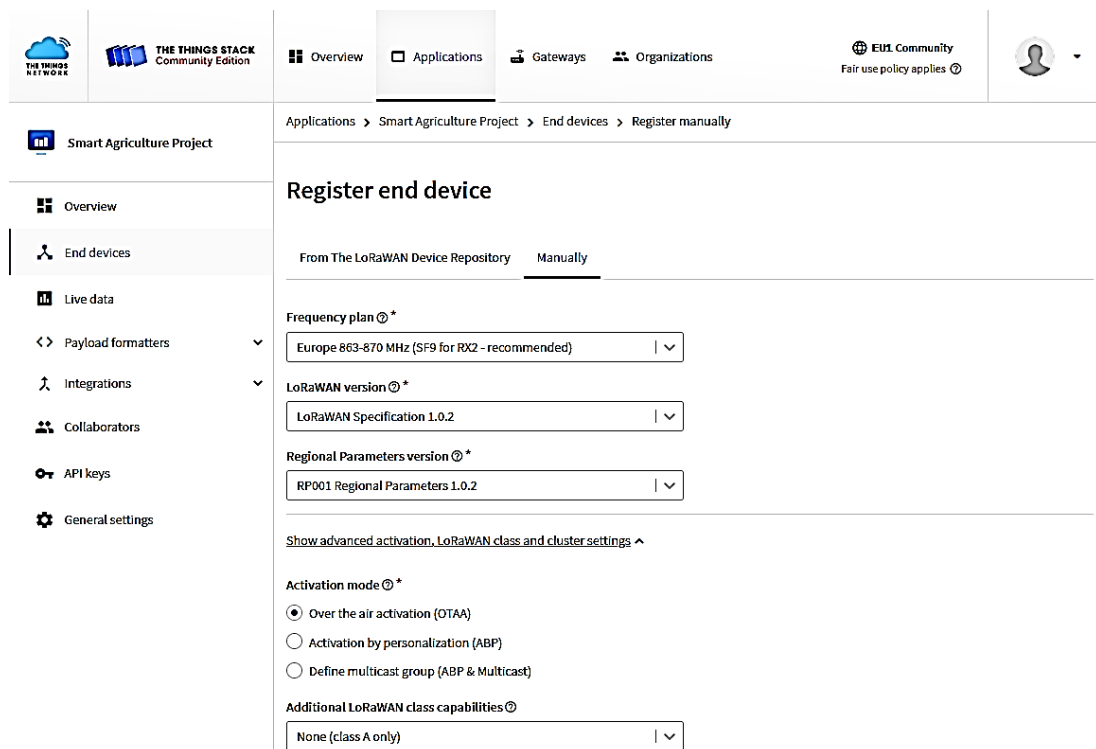
Για την δημιουργία μιας νέας τελικής συσκευής επιλέγεται “Add end device” στην οθόνη της εικόνας 3.11. Εμφανίζεται η οθόνη που φαίνεται στην εικόνα 3.12, στην οποία παρέχεται η δυνατότητα επιλογής ανάμεσα σε μια βάση δεδομένων τελικών συσκευών (from The LoRaWAN Device Repository) όταν πρόκειται για συσκευές που πωλούνται στο εμπόριο, ή την προσθήκη μιας άγνωστης συσκευής (Manually) όταν πρόκειται για συσκευές που δεν είναι γνωστές.



Εικόνα 3.12 – Καταχώρηση τελικής συσκευής στο TTN

Πηγή: The Things Network

Για την περίπτωση της εργασίας επιλέγεται 'Manually' και εμφανίζεται η οθόνη που παρουσιάζεται στις εικόνες 3.13(α) και 3.13(β) όπου πρέπει να συμπληρωθούν οι απαραίτητες πληροφορίες που ζητούνται.



Εικόνα 3.13(α) – Καταχώρηση στοιχείων τελικής συσκευής

Πηγή: The Things Network

The screenshot shows the 'General settings' page for registering a final device in The Things Network. The sidebar on the left contains navigation links: Live data, Payload formatters, Integrations, Collaborators, API keys, and General settings. The main content area is titled 'Network defaults' and includes the following sections:

- Network defaults**: Includes a checked checkbox for 'Use network's default MAC settings' and an unchecked checkbox for 'Use external LoRaWAN backend servers'.
- DevEUI**: A text input field containing '70 B3 D5 7E D0 05 06 E2', a 'Generate' button, and a '1/50 used' indicator.
- AppEUI**: A text input field containing '00 00 00 00 00 00 00 00' and a 'Fill with zeros' button.
- AppKey**: A text input field containing '5D DE 9D 7F A9 CF D6 74 2B 1D CB 6E 52 5C 64 9B' and a 'Generate' button.
- End device ID**: A text input field containing 'eui-70b3d57ed00506e2' with a note below it stating 'This value is automatically prefilled using the DevEUI'.
- After registration**: Two radio buttons: 'View registered end device' (selected) and 'Register another end device of this type'.

At the bottom of the form, there is a blue 'Register end device' button.

Εικόνα 3.13 (β) – Καταχώρηση στοιχείων τελικής συσκευής

Πηγή: The Things Network

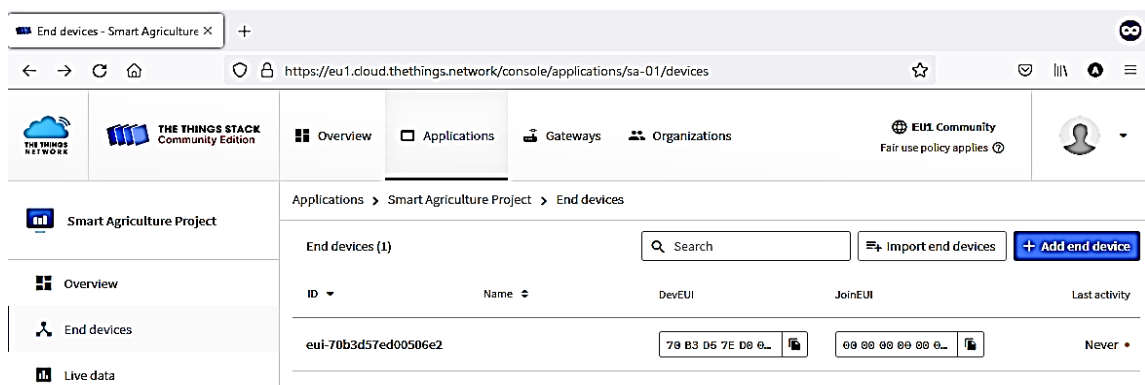
Παρακάτω αναλύονται τα πεδία από τις παραπάνω εικόνες καταχώρησης τελικής συσκευής 3.13(α) και 3.13(β).

- **Frequency plan:** Επιλέγονται οι συχνότητες που επιτρέπονται για την περιοχή λειτουργίας της τελικής συσκευής. Για την Ελλάδα επιλέγεται (863 – 870MHz).
- **LoRaWAN version:** Αφορά το version του πρωτοκόλλου LoRaWAN που πρόκειται να επιλεγεί προς χρήση. Επιλέχθηκε η version 1.0.2 καθότι υποστηρίζεται από τον κώδικα που πρόκειται να χρησιμοποιηθεί για τις τελικές συσκευές.
- **Regional parameters:** Σχετίζεται με την επιλογή ορισμένων τοπικών παραμέτρων αλλά επίσης και με το LoRaWAN version που επιλέχθηκε στην προηγούμενη επιλογή.
- **Activation mode:** Παρέχεται δυνατότητα επιλογής ανάμεσα σε Over the Air Activation (OTAA) ή Activation by personalization (ABP). Επιλέχθηκε η μέθοδος OTAA καθότι παρέχει αυξημένη ασφάλεια σε σχέση με την μέθοδο ABP.
- **Additional LoRaWAN class capabilities:** Επιλέγεται η κλάση λειτουργίας της τελικής συσκευής που πρόκειται να χρησιμοποιηθεί. Επιλέχθηκε η κλάση A.5
- **Network defaults:** Με το Network defaults ορίζεται ο ρυθμός και η συχνότητα λειτουργίας του παραθύρου Rx2 για τις συσκευές κλάσης A. Ο εκάστοτε κατασκευαστής τελικών συσκευών ορίζει αυτές τις λεπτομέρειες. Για τις τελικές συσκευές που πρόκειται να

χρησιμοποιηθούν επιλέχθηκαν οι προκαθορισμένες τιμές που προτείνονται από το TTN (RX2 data rate 0 και Rx2 frequency 869.525 MHz).

- **Cluster settings:** Μέσω της επιλογής Cluster settings παρέχεται η δυνατότητα χρήσης ενός διακομιστή δικτύου εκτός του TTN. Επιλέγεται ο διακομιστής του TTN.
- **DevEUI, AppEUI:** Πρόκειται για κωδικούς οι οποίοι διατίθενται από τους κατασκευαστές των τελικών συσκευών. Επιλέγεται να γίνει χρήση της γεννήτριας τυχαίων κωδικών που διατίθεται από το TTN για την δημιουργία αυτών των κωδικών μέσω της επιλογής 'Generate'.
- **AppKey:** Ο κωδικός AppKey παρέχεται επίσης από τον κατασκευαστή της τελικής συσκευής. Για συσκευές που δεν διατίθεται το AppKey, συμπληρώνεται με μηδενικά μέσω της επιλογής "Fill with zeros".
- **End DeviceID:** Το End Device ID συμπληρώνεται αυτόματα από το σύστημα μόλις συμπληρωθεί το DevEUI, αλλά παρέχεται στον χρήστη η δυνατότητα τροποποίησης.
- **After Registration:** Επιλέγεται 'Register another end device of this type' για περαιτέρω καταχώρηση τελικών συσκευών, αλλιώς 'View registered end device' για τον τερματισμό της καταχώρησης.

Τέλος, με την χρήση της επιλογής "Register end Device", η τελική συσκευή καταχωρείται στο TTN όπως φαίνεται στην Εικόνα 3.14.



Εικόνα 3.14 – Λίστα καταχωρημένων τελικών συσκευών στο TTN

Πηγή: The Things Network

3.3 Δρομολογητής LoRa Gateway - Mikrotik wAP LR8 kit

Η ενότητα αυτή ασχολείται με το δεύτερο βήμα δηλαδή με την ρύθμιση του δρομολογητή LoRa που πρόκειται να χρησιμοποιηθεί. Αναλύονται διεξοδικά όλες οι απαραίτητες λεπτομέρειες που πρέπει να ακολουθηθούν έτσι ώστε ο δρομολογητής να είναι σε θέση να στέλνει δεδομένα προς το TTN.

3.3.1 Τεχνικά χαρακτηριστικά του wAP LR8 kit

Το wAP LR8 kit της Mikrotik[24] αποτελεί μια ολοκληρωμένη λύση για χρήση της τεχνολογίας LoRa. Χρησιμοποιεί το chipset SX1301 της Semtech και υποστηρίζει συχνότητα 863-870 MHz που χρησιμοποιείται στην Ευρωπαϊκή Ένωση. Διαθέτει 8 διαφορετικά κανάλια και τεχνολογία Listen before Talk (LBT) μέσω της οποίας γίνεται έλεγχος της συχνότητας που πρόκειται να χρησιμοποιηθεί αν είναι ελεύθερη, πριν από κάθε εκπομπή.

Χρησιμοποιεί πρωτόκολλο UDP για προώθηση πακέτων σε οποιονδήποτε δημόσιο ή ιδιωτικό διακομιστή. Διαθέτει θύρα Ethernet και υποστηρίζει σύνδεση Power Over Ethernet (PoE) έτσι ώστε να μπορεί να τροφοδοτηθεί μέσω ενός καλωδίου δικτύου. Πέρα από την σύνδεση Ethernet διαθέτει και σύνδεση WLAN 2.4 GHz κάνοντας χρήση του πρωτοκόλλου 802.11b/g/n.

Είναι αδιάβροχο, σταθερό στη λειτουργία του και παρέχει εξαιρετική απόδοση σχεδόν σε όλες τις καιρικές συνθήκες.

Διαθέτει εσωτερική κεραία 2 dBi τυπωμένου κυκλώματος αλλά για επιπλέον κάλυψη παρέχει ειδικό connector έτσι ώστε να μπορεί να υποστηρίξει και εξωτερική κεραία.

Τα τεχνικά χαρακτηριστικά του δρομολογητή wAP LR8 kit καθώς και του LoRa module που περιλαμβάνει, ορίζονται από τον κατασκευαστή και παρουσιάζονται στις εικόνες 3.15 και 3.16.

R11e-LR8/R11e-LR9 specifications

Product code	R11e-LR8	R11e-LR9
Interface	Mini-PCle	
Supported class	A and C	
Frequency	863-870 MHz (EU863-870, RU864-870, IN865-867)	902-928 MHz (AU915-928, US902-928, AS923, KR920-923)
RF Output power	863-870 MHz 20 dBm	902-928 MHz 23 dBm
Receive max sensitivity	-137 dB @ SF12	
Range	Up to 15 km in rural environment and up to 2 km in urban environment when using MikroTik LoRa® 6.5 dBi antenna kit	
Operating ambient temperature	-40°C .. +70°C	
Max power consumption	2 W	



Εικόνα 3.15 – Χαρακτηριστικά LoRa Module R11e-LR8

(Πηγή: https://mikrotik.com/product/r11e_lr8)

wAP LR8/LR9 kit specifications

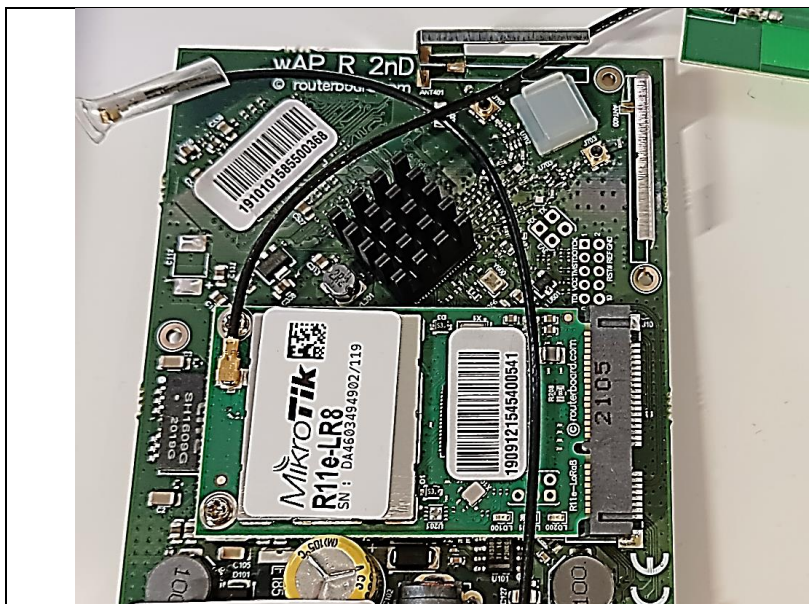


Product code	RBwAPR-2nD&R11e-LR8	RBwAPR-2nD&R11e-LR9
CPU	QCA9531 650 MHz	
Size of RAM	64 MB	
10/100 Ethernet ports	1	
Wireless	Built-in 2.4 GHz 802.11b/g/n, dual-chain	
Antenna gain	2 dBi	
PoE in	Yes	
Supported input voltage	9 V - 30 V (Passive PoE)	
Dimensions	185 x 85 x 30 mm	
Operating ambient temperature	-40°C .. +60°C	
Operating system	RouterOS, License level 4	
Max power consumption	7 W	

Εικόνα 3.16 – Χαρακτηριστικά δρομολογητή wAP LR8
(Πηγή: https://mikrotik.com/product/r11e_lr8)

3.3.2 Εγκατάσταση & Σύνδεση με το διαδίκτυο των πραγμάτων (TTN).

Αρχικά πριν την σύνδεση του gateway με ένα τοπικό δίκτυο, πρέπει να γίνει αλλαγή της κεραίας από εξωτερική σε εσωτερική κάνοντας χρήση της κεραίας που διαθέτει η συσκευή υπό μορφή τυπωμένου κυκλώματος. Αυτό γίνεται αφαιρώντας το καπάκι και αλλάζοντας το κλιπ σύνδεσης της εξωτερικής κεραίας με το κλιπ σύνδεσης της εσωτερικής κεραίας όπως φαίνεται στις εικόνες 3.17 (α) και 3.17(β).



Εικόνα 3.17 (α) – Εσωτερικό δρομολογητή wAP LR8 kit

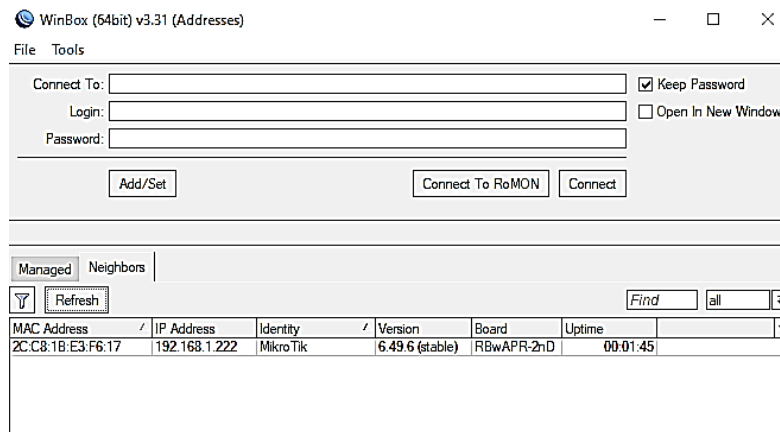


Εικόνα 3.17 (β) - Εσωτερικό δρομολογητή wAP LR8 kit

Στην συνέχεια γίνεται εγκατάσταση του λογισμικού winbox της Mikrotik μέσω του οποίου πρόκειται να γίνουν όλες οι απαραίτητες ενέργειες έτσι ώστε να συνδεθεί ο δρομολογητής με το TTN.

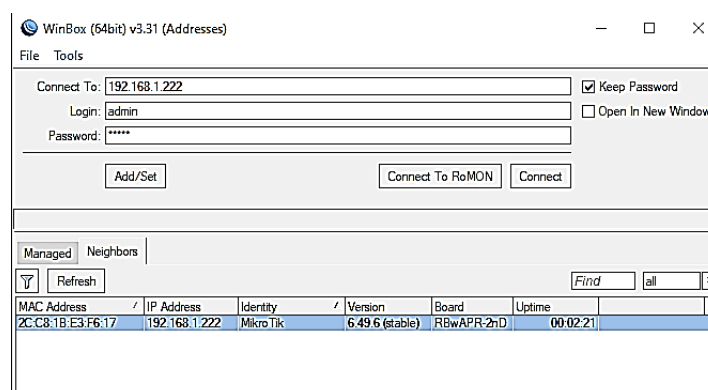
Link για download : <https://mt.lv/winbox64>

Αφότου ολοκληρωθεί η εγκατάσταση, με διπλό κλικ στο εικονίδιο του WinBox εμφανίζεται η οθόνη της εικόνας 3.17.



Εικόνα 3.17 – Οθόνη σύνδεσης winBox

Στο tab “Neighbors” εμφανίζονται όλες οι συσκευές Mikrotik μαζί με τα απαραίτητα στοιχεία (IP, MAC Address) που βρίσκονται στο τοπικό δίκτυο. Η σύνδεση με την συσκευή πραγματοποιείται συμπληρώνοντας τα στοιχεία εισόδου, login και password και εκτελώντας διπλό κλικ επάνω σε αυτήν ή απλά επιλέγοντας “Connect”, όπως παρουσιάζεται στην εικόνα 3.18.



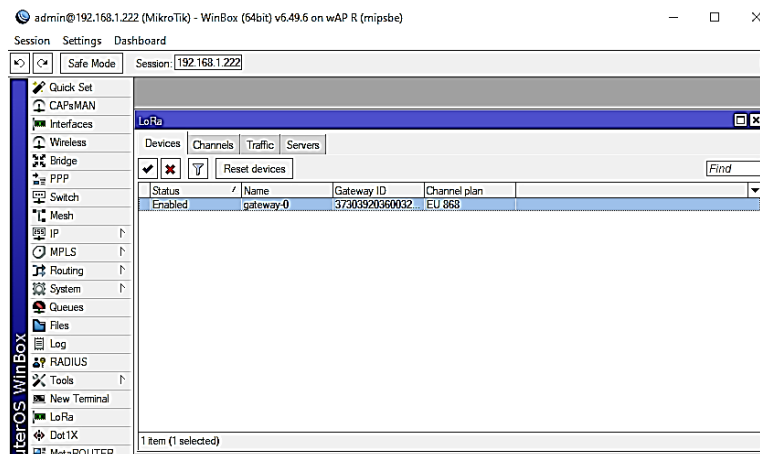
Εικόνα 3.18 – Winbox διαδικασία login

Στην συνέχεια, εάν τα στοιχεία που καταχωρήθηκαν είναι σωστά, μας εμφανίζεται το κεντρικό μενού ρυθμίσεων του δρομολογητή που φαίνεται στην εικόνα 3.19.



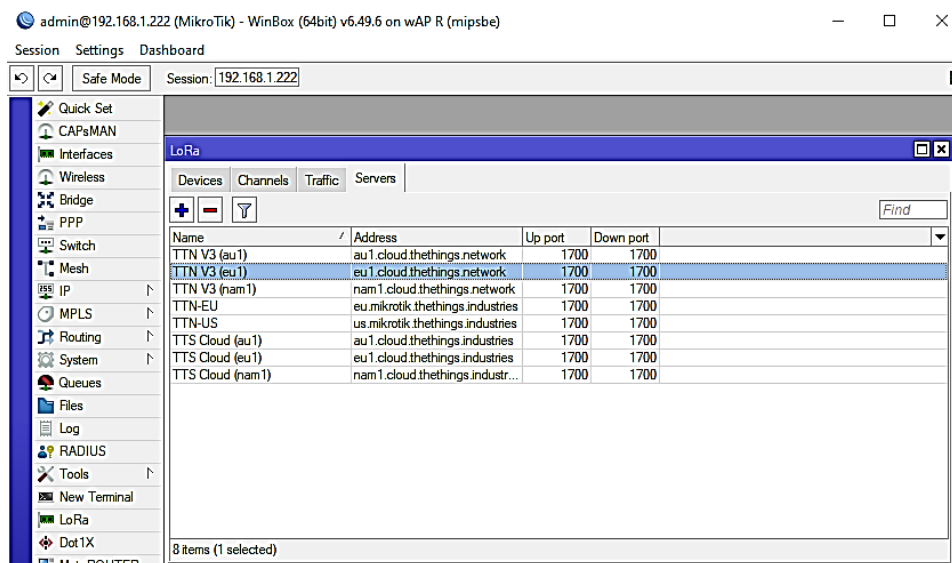
Εικόνα 3.19 – Κεντρικό μενού δρομολογητή wAP LR8 kit

Η πρώτη ρύθμιση που πρέπει να γίνει αφορά την καταχώρηση διακομιστή του TTN. Επιλέγοντας 'LoRa' από το μενού που βρίσκεται στην αριστερή στήλη ανοίγει το παράθυρο της εικόνας 3.20.



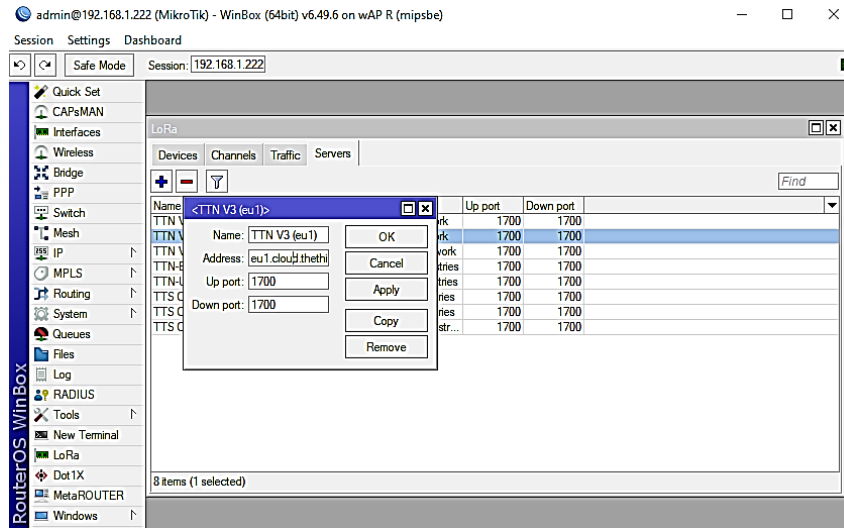
Εικόνα 3.20 – Διαθέσιμες συσκευές LoRa στο wAP LR8 kit

Επιλέγοντας στην συνέχεια το tab "Servers" μια λίστα με ορισμένους διακομιστές LoRa εμφανίζονται στην οθόνη μας (εικόνα 3.21).



Εικόνα 3.21 – Διαθέσιμοι διακομιστές LoRa στο wAP LR8 kit

Εδώ ελέγχεται εάν ο διακομιστής όπου επιλέχθηκε κατά την διαδικασία καταχώρησης στο TTN (eu1.cloud.thethings.network) υπάρχει στην συγκεκριμένη λίστα. Εάν όχι, πρέπει να προστεθεί πατώντας το μπλε εικονίδιο (+) όπως φαίνεται στην εικόνα 3.22.



Εικόνα 3.22 – Καταχώρηση LoRa server στο wAP LR8 kit

Κατά την περίπτωση καταχώρησης διακομιστή 'Lora' (εικόνας 3.22) απαιτούνται τα ακόλουθα πεδία:

Name: Καθαρά περιγραφικό πεδίο του διακομιστή που πρόκειται να γίνει καταχώρηση.

Address: Αφορά την διεύθυνση του διακομιστή LoRa που πρόκειται να χρησιμοποιηθεί. Επιλέγεται ο διακομιστής του TTN με διεύθυνση eu1.cloud.thethings.network

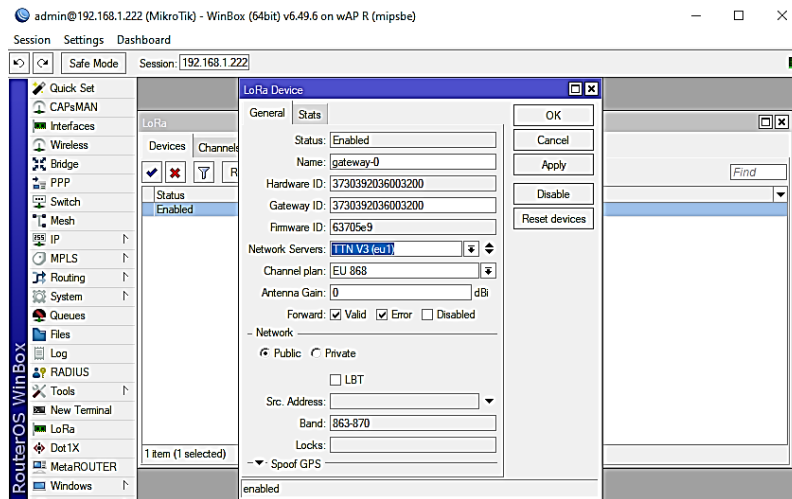
Up port / Down port: Δηλώνονται οι κατάλληλες πόρτες για τα ανερχόμενα και κατερχόμενα μηνύματα. Επιλέγεται η πόρτα 1700 και για τις δύο κατευθύνσεις μηνυμάτων.

Η καταχώρηση ολοκληρώνεται επιλέγοντας 'OK' από το παράθυρο της καταχώρησης.

Στην συνέχεια στην ίδια οθόνη αλλάζουμε tab και επιλέγουμε "Devices" όπου εμφανίζεται η οθόνη της εικόνας 3.20. Το μοναδικό device που βλέπουμε είναι το LoRa module που περιλαμβάνεται στον δρομολογητή. Εκτελούμε διπλό κλικ επάνω του και ανοίγει ένα νέο παράθυρο που φαίνεται στην εικόνα 3.23.

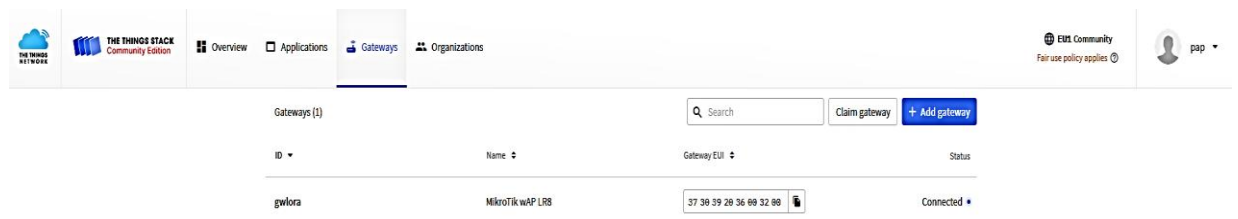
Στο πεδίο 'Network Servers' επιλέγεται το όνομα του διακομιστή που μόλις καταχωρήθηκε προηγουμένως. Αντίστοιχα, στο πεδίο Network επιλέγεται η τιμή public έτσι ώστε ο δρομολογητής να είναι δημόσιος, δηλαδή να επιτρέπεται η χρήση του σε οποιονδήποτε βρίσκεται σε μία κοντινή ακτίνα έτσι ώστε να μπορεί να στείλει πακέτα LoRa στις δικές του εφαρμογές.

Επιλέγουμε 'OK' και κρατάμε τις αλλαγές που πραγματοποιήθηκαν.



Εικόνα 3.23 – Καταχώρηση LoRa server στο wAP LR8 kit

Μετά από τις παραπάνω ρυθμίσεις το gateway πρέπει να είναι σε θέση να συνδεθεί με το TTN. Για επαλήθευση, κατόπιν εισαγωγής στο TTN επιλέγουμε “Go to gateways”. Μας εμφανίζεται η οθόνη της εικόνα 3.24 που μας παρέχει την κατάσταση σύνδεσης του δρομολογητή wAP LR8 kit στο TTN μέσω του πεδίου κατάστασης (status). Με πεδίο κατάστασης “Connected”, ο δρομολογητής είναι συνδεδεμένος και έτοιμος για αποστολή και λήψη πακέτων.



Εικόνα 3.24 – Λίστα καταχωρημένων δρομολογητών στο TTN

3.4 Αισθητήρες & Μικροελεγκτές

Η ενότητα αυτή ασχολείται με την κατασκευή των τελικών συσκευών που πρόκειται να χρησιμοποιηθούν στην παρούσα εργασία. Συνολικά κατασκευάζονται δύο τελικές συσκευές. Η πρώτη αποτελείται από διάφορους αισθητήρες όπως θερμοκρασίας αέρος και εδάφους, υγρασίας αέρος και εδάφους, βροχής κτλ και έχει κύρια αρμοδιότητα την συλλογή δεδομένων από το σημείο που θα γίνει η τοποθέτησή της. Η δεύτερη τελική συσκευή

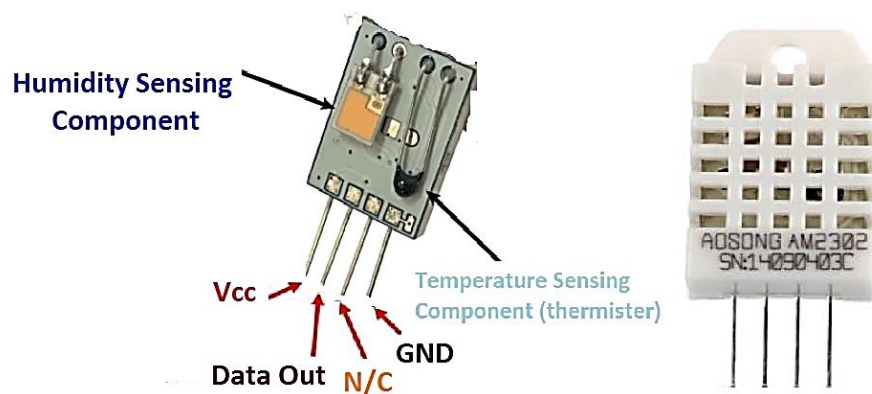
σχετίζεται με την λειτουργία των ενεργοποιητών αλλά παράλληλα μπορεί να διαχειρίζεται και ορισμένους αισθητήρες.

Η διαδικασία της ανάπτυξης της πρώτης τελικής συσκευής πρόκειται να γίνει τμηματικά. Δηλαδή, για κάθε έναν αισθητήρα παρουσιάζεται η απαιτούμενη σύνδεση με το Arduino καθώς επίσης και ο απαραίτητος κώδικας λειτουργίας. Στο τέλος, συνδυάζονται όλοι οι αισθητήρες μαζί σε έναν μικροελεγκτή και με την προσθήκη ενός LoRa module για την ασύρματη μετάδοση καθώς με τον απαραίτητο κώδικα λειτουργίας δημιουργείται ένας πολυαισθητήρας όπου η βασική του λειτουργία είναι η αποστολή δεδομένων προς το TTN.

Η δεύτερη τελική συσκευή αποτελείται από έναν αισθητήρα βροχής και είναι υπεύθυνη για τον έλεγχο των ενεργοποιητών του συστήματος.

3.4.1 Ψηφιακός αισθητήρας θερμοκρασίας DHT 22

Ο DHT22[3.1] είναι ένας αξιόπιστος και σταθερός ψηφιακός αισθητήρας θερμοκρασίας και υγρασίας χαμηλού κόστους. Χρησιμοποιεί έναν χωρητικό αισθητήρα για την μέτρηση της υγρασίας της ατμοσφαιράς και ένα θερμίστορ για τη μέτρηση της θερμοκρασίας του αέρα.



Εικόνα 3.25 – Αισθητήρας DHT22 (Θερμοκρασίας, υγρασίας)

(Πηγή: <https://www.theengineeringprojects.com/2019/02/introduction-to-dht22.html>)

Το εύρος τιμών για την θερμοκρασία είναι μεταξύ -40°C και 80°C ενώ για την υγρασία από 0% έως 100%. Έχει πολύ καλή πιστότητα (accuracy $\pm 1\%$), δηλαδή η διαφορά επι τοις εκατό που παρουσιάζει η έξοδος του αισθητήρα σε σχέση με την πραγματική τιμή μέτρησης είναι πολύ μικρή. Επίσης, έχει πολύ καλή ακρίβεια μέτρησης (precision) δηλαδή αν πάρουμε έναν μεγάλο αριθμό μετρήσεων με τις ίδιες συνθήκες τότε η μεταξύ τους διασπορά θα είναι μικρή.

Συνολικά διαθέτει 4 ποδαράκια (Vcc, Data out, N/C, GND) τα οποία γίνονται 3 εάν τοποθετηθεί σε πλάκeta που περιλαμβάνει μια pull up αντιστάση (4,7 K έως 10 K) μεταξύ Vcc και Data καθώς επίσης και έναν πυκνωτή.



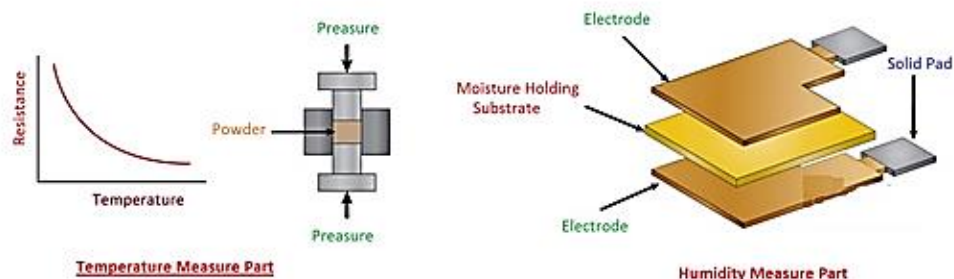
Εικόνα 3.26 – DHT 22 με τρεις ακροδέκτες

Τα τρία ποδαράκια σύνδεσης είναι τα ακόλουθα:

- Vcc - Θετική τάση τροφοδοσίας από 3.5V έως 5
- Data – Έξοδος ψηφιακών δεδομένων
- GND – Σύνδεση γείωσης

Ο αισθητήρας αποτελείται από τρία διαφορετικά μέρη.

Το πρώτο μέρος είναι το αισθητήριο για την μέτρηση της υγρασίας το οποίο αποτελείται από δύο πλάκες που ορίζουν τα δύο ηλεκτρόδια του αισθητήρα και στην μέση υπάρχει ένα υλικό κατάλληλο για να συγκρατεί την υγρασία όπως φαίνεται στην εικόνα 3.27. Καθώς σταδιακά η υγρασία της ατμοσφαιρας μεταβάλλεται, η αγωγιμότητα και κατά συνέπεια η αντίσταση ανάμεσα στις δύο πλάκες αλλάζει επίσης. Αυτή η αλλαγή της αντίστασης είναι μετρήσιμη και κατ'επέκταση αξιοποιήσιμη από ένα ολοκληρωμένο IC το οποίο αποτελεί το δεύτερο μέρος του αισθητήρα. Το ολοκληρωμένο αυτό είναι υπευθυνο για την αποστολή των δεδομένων προς έναν μικροελεγκτή. Το τρίτο μέρος είναι το αισθητήριο μέτρησης της θερμοκρασίας το οποίο αποτελείται από ένα θερμίστορ τύπου NTC (Negative Temperature Coefficient) το οποίο μεταβάλλει την αντίσταση του με την αλλαγή της θερμοκρασίας. Οι NTC αισθητήρες κατασκευάζονται με πυροσυσσωμάτωση ημιαγωγικών υλικών, όπως κεραμικά ή πολυμερή υλικά, με σκοπό την μεγάλη αλλαγή της αντίστασης για μικρές μεταβολές της θερμοκρασίας.



Εικόνα 3.27 – Εσωτερικά μέρη του αισθητήρα DHT-22

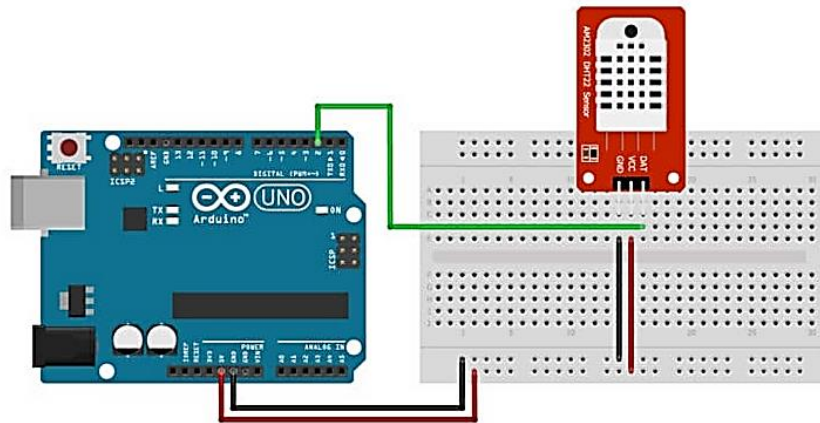
(Πηγή: <https://www.theengineeringprojects.com/2019/02/introduction-to-dht22.html>)

Η αποστολή των δεδομένων από τις μετρήσεις των δύο αισθητηρίων γίνεται μέσω σειριακής επικοινωνίας.

Το μόνο πραγματικό μειονέκτημα αυτού του αισθητήρα είναι ότι λαμβάνονται νέα δεδομένα από αυτόν μόνο μία φορά κάθε 2 δευτερόλεπτα.

3.4.2 Σύνδεση και προγραμματισμός του DHT 22

Η σύνδεση του DHT 22 με έναν μικροελεγκτή τύπου Arduino είναι απλή και παρουσιάζεται στην εικόνα 3.28. Αρχικά τροφοδοτείται ο αισθητήρας με συνεχή τάση συνδέοντας ένα καλώδιο από την παροχή 5V του Arduino προς το ποδαράκι με την ένδειξη Vcc και άλλο ένα από μια υποδοχή GND του arduino προς την ένδειξη GND του αισθητήρα. Η μεταφορά των δεδομένων από τον αισθητήρα προς το Arduino πραγματοποιείται συνδέοντας το pin Data του αισθητήρα σε μια από τις ψηφιακές υποδοχές του arduino. Η επιλογή μπορεί να είναι τυχαία αρκεί βέβαια να δηλωθεί στον κώδικά λειτουργίας.



Εικόνα 3.28 – Σύνδεση αισθητήρα DHT-22

(Πηγή: <https://www.makerguides.com/dht11-dht22-arduino-tutorial/>)

Ο κώδικας λειτουργίας είναι ο ακόλουθος:

```
// Ενσωμάτωση απαιτούμενων βιβλιοθηκών του κατασκευαστή
#include <DHT.h>
#include <DHT_U.h>
#include <Adafruit_Sensor.h>

// Καθορισμός ψηφιακής εισόδου και τύπου του αισθητήρα
#define DHT_PIN 2
#define DHTTYPE DHT22

// Αρχικοποίηση του αισθητήρα
DHT dht(DHT_PIN, DHTTYPE);

void setup() {
  // Ενεργοποίηση της σειριακή θύρας για την λήψη δεδομένων
  Serial.begin(115200);
  Serial.println(F("Starting ..... "));
}
```

```
// Ενεργοποίηση του αισθητήρα
dht.begin();
}

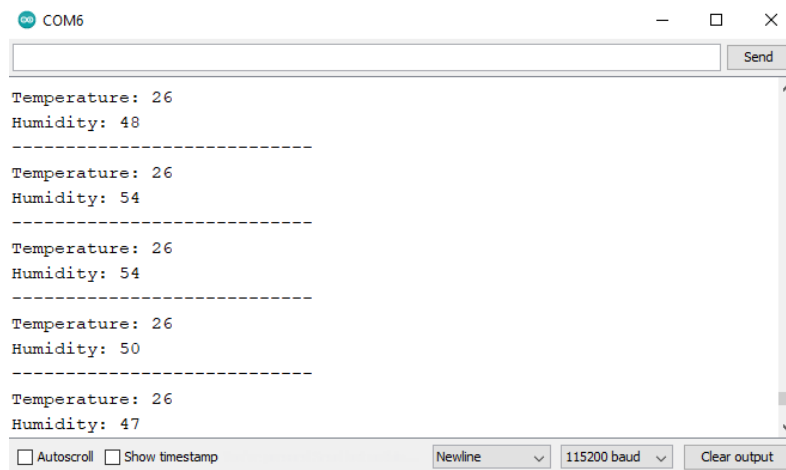
void loop() {
  //Ανάθεση τιμής θερμοκρασίας (Celsius) σε μεταβλητή
  int16_t t = dht.readTemperature();
  // Ανάθεση υγρασίας (%) σε μεταβλητή:
  uint16_t h = dht.readHumidity();

  // Έλεγχος ύπαρξης προβλήματος κατά την μέτρηση και
  // επανάληψη της διαδικασίας μέτρησης.
  if (isnan(t) || isnan(h) || t==0 || h==0) {
    Serial.println("Failed to read data from DHT sensor!");
    return;
  }

  //Εκτύπωση τιμών Θερμοκρασίας/Υγρασίας στο σειριακό παράθυρο
  Serial.print("Temperature: ");
  Serial.println(t);
  Serial.print("Humidity: ");
  Serial.println(h);
  Serial.println("-----");

  // Μικρή καθυστέρηση
  delay(3000);
}
```

Στην συνέχεια αφού πρώτα μεταφερθεί ο παραπάνω κώδικας μέσω του Arduino IDE στον μικροελεγκτή λαμβάνονται οι ακόλουθες τιμές που φαίνονται στην εικόνα 3.29 μέσω της σειριακής επικοινωνίας.

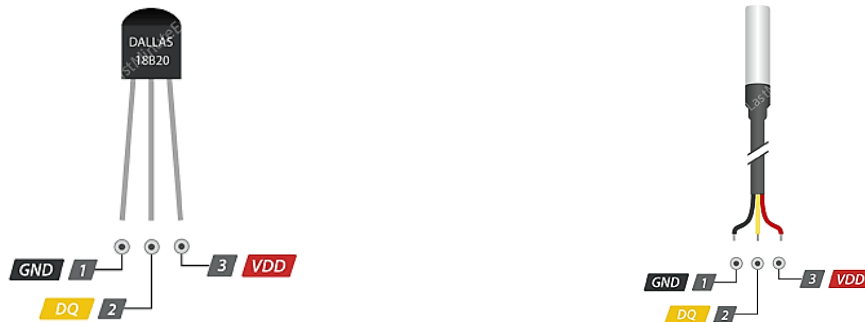


The screenshot shows the serial monitor window for COM6. The output displays four sets of data, each separated by a dashed line. The first set shows Temperature: 26 and Humidity: 48. The second set shows Temperature: 26 and Humidity: 54. The third set shows Temperature: 26 and Humidity: 54. The fourth set shows Temperature: 26 and Humidity: 50. The fifth set shows Temperature: 26 and Humidity: 47. The window includes a 'Send' button at the top right and control options at the bottom: 'Autoscroll', 'Show timestamp', 'Newline', '115200 baud', and 'Clear output'.

Εικόνα 3.29 – Αποτέλεσμα αισθητήρα μέσω σειριακής επικοινωνίας

3.4.3 Αναλογικός αισθητήρας θερμοκρασίας εδάφους DS18B20

Ο DS18B20[3.2] που φαίνεται στην εικόνα 3.30 είναι ένας έξυπνος αισθητήρας θερμοκρασίας που κυκλοφόρησε από την American DALLAS Semiconductor ο οποίος παρέχει μετρήσεις θερμοκρασίας σε βαθμούς Κελσίου (Celsius) με ακρίβεια 9 έως 12-bit.



Εικόνα 3.30 – Ψηφιακός αισθητήρας θερμοκρασίας - DS18B20 (Waterproof)

(Πηγή: <https://lastminuteengineers.com/ds18b20-arduino-tutorial/>)

Διαθέτει λειτουργία συναγερμού με προγραμματιζόμενα από το χρήστη πάνω και κάτω όρια θερμοκρασιών που διατηρούνται ακόμη και όταν ο αισθητήρας δεν τροφοδοτείται με τάση λειτουργίας.

Επικοινωνεί σειριακά μέσω ενός 1-Wire bus δηλαδή απαιτεί μια γραμμή δεδομένων και τη γείωση για να μπορεί να στέλνει δεδομένα προς έναν μικροελεγκτή.

Μπορεί να λειτουργεί στην περιοχή θερμοκρασιών από -55°C έως 125°C αλλά παρέχει ακρίβεια μέτρησης $\pm 0,5^{\circ}\text{C}$, μόνο για την περιοχή -10°C έως 85°C .

Ο DS18B20 μπορεί να τροφοδοτηθεί από εξωτερική τάση τροφοδοσίας ή μπορεί επίσης να λειτουργεί σε κατάσταση “parasite mode” στην οποία αντλεί ισχύ από την γραμμή δεδομένων. Σε αυτή την περίπτωση, όταν ο ακροδέκτης data βρίσκεται σε high κατάσταση παίρνει ισχύ διαμέσου μιας pull-up αντίστασης που είναι συνδεδεμένη στον ακροδέκτη και φορτίζει ένα εσωτερικό πυκνωτή (C_{pp}) που χρησιμεύει για να παρέχει ισχύ στον αισθητήρα όταν ο ακροδέκτης data βρίσκεται σε κατάσταση low.

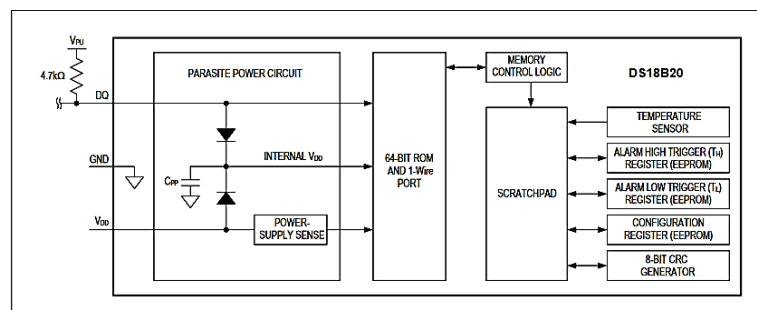


Figure 3. DS18B20 Block Diagram

Εικόνα 3.31 – Διάγραμμα δομής του αισθητήρα θερμοκρασίας - DS18B20

(Πηγή: <https://www.apogeeweb.net/circuitry/ds18b20-working-principle.html>)

Τα μέρη που φαίνονται στην εικόνα 3.31 από τα οποία αποτελείται ένας αισθητήρας DS18B20 περιγράφονται ακολούθως:

- Μία μνήμη ROM ενσωματώνει ένα 64-bit μοναδικό σειριακό κωδικό ο οποίος επιτρέπει την σύνδεση πολλαπλών τέτοιων αισθητήρων στο ίδιο (1-Wire) δίαυλο δεδομένων. Αυτό το χαρακτηριστικό παρέχει την δυνατότητα να μπορεί να γίνεται χρήση της ίδιας ψηφιακής θύρας ενός μικροελεγκτή από πολλούς αισθητήρες DS18B20 οι οποίοι μπορεί να είναι κατανεμημένοι γύρω από μια περιοχή.
- Μια περιοχή μνήμης SRAM στην οποία περιέχεται ένας 2-byte καταχωρητής θερμοκρασίας για να αποθηκεύει την ψηφιακή έξοδο του αισθητήρα θερμοκρασίας, δύο 1-byte καταχωρητές για το πάνω και το κάτω ορίου λειτουργίας συναγερμού θερμοκρασίας (TH και TL) καθώς και ένας 1-byte καταχωρητής διαμόρφωσης που επιτρέπει στον χρήστη να επιλέγει την ανάλυση του μετατροπέα θερμοκρασίας σε ψηφιακό σήμα των 9, 10, 11 ή 12 bits.
- Μια μνήμη EEPROM όπου αποθηκεύονται οι καταχωρητές TH και TL και ο καταχωρητής διαμόρφωσης, κάνοντας χρήση συγκεκριμένων εντολών έτσι ώστε τα δεδομένα να διατηρούνται στην περίπτωση που υπάρχει διακοπή τροφοδοσίας.
- Μια 8-bit γεννήτρια CRC που χρησιμεύει για την επιβεβαίωση της ακεραιότητας των δεδομένων κατά την μεταφορά τους από τον αισθητήρα προς τον μικροελεγκτή.
- Μια διάταξη τροφοδοσίας

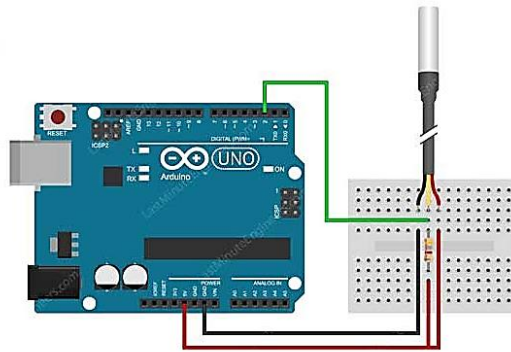
Στον ακόλουθο πίνακα 3.1 παρουσιάζονται τα βασικά χαρακτηριστικά του *DS18B20*.

Power Supply	3V to 5.5V
Current Consumption	1mA
Temperature Range	-55 to 125°C
Accuracy	±0.5°C
Resolution	9 to 12 bit (selectable)
Conversion Time	< 750ms

Πίνακας 3.1 – Βασικά χαρακτηριστικά του *DS18B20*
(Πηγή: <https://lastminuteengineers.com/ds18b20-arduino-tutorial/>)

3.4.4 Σύνδεση και προγραμματισμός του DS18B20

Ο τρόπος σύνδεσης ενός αισθητήρα DS18B20 με έναν μικροελεγκτή Arduino Uno παρουσιάζεται στην εικόνα 3.32. Είναι παρόμοιος με τον τρόπο σύνδεσης του DHT22 με την μόνη διαφορά ότι απαιτείται μία pull-up αντίσταση στον ακροδέκτη του σήματος.



Εικόνα 3.32 – Σύνδεση του DS18B20 σε Arduino Uno
(Πηγή: <https://lastminuteengineers.com/ds18b20-arduino-tutorial/>)

Ο απαιτούμενος κώδικας λειτουργίας είναι ο ακόλουθος:

```
//Ενσωμάτωση απαραίτητων βιβλιοθηκών
#include <OneWire.h>
#include <DallasTemperature.h>

// Ορισμός μεταβλητής ONE_WIRE_BUS ίση με 2
#define ONE_WIRE_BUS 2

// Δημιουργία νέο αντικείμενου της κλάσης OneWire με όνομα oneWire και όρισμα την μεταβλητή
// ONE_WIRE_BUS
OneWire oneWire(ONE_WIRE_BUS);

//Δημιουργία αντικείμενου sensors με όρισμα μέσω pointer
DallasTemperature sensors(&oneWire);

void setup(void)
{
  // Εκκίνηση της σειριακής επικοινωνίας
  Serial.begin(115200);
  Serial.println("Dallas Temperature IC Control Library Demo");

  //Εκκίνηση sensors
  sensors.begin();
}

void loop(void)
{
  // Ζητείται η θερμοκρασία από την κλάση sensors και τυπώνεται μέσω σειριακής επικοινωνίας
  Serial.print(" Requesting temperatures...");
  sensors.requestTemperatures();
  Serial.println("DONE");
}
```

```
// Παρεμβάλλεται μικρή καθυστέρηση  
delay(1000);  
}
```

3.4.5 Ψηφιακός αισθητήρας BMP280 μέτρησης ύψους & βαρομετρικής πίεσης

Ο BMP280 [3.3] που παρουσιάζεται στην εικόνα 3.33 είναι ένας αισθητήρας για την μέτρηση της βαρομετρικής πίεσης και της θερμοκρασίας, κατασκευασμένος από την εταιρεία Bosh. Λόγω των μικρών του διαστάσεων και της μικρής του κατανάλωσης σε ενέργεια είναι ιδιαίτερα κατάλληλος για κινητές συσκευές που τροφοδοτούνται από μπαταρίες και απαιτούν μέτρηση της πίεσης και της θερμοκρασίας.

Διαθέτει δύο διαφορετικά πρωτόκολλα επικοινωνίας. Για μια απλή και γρήγορη διασύνδεση του αισθητήρα μπορούμε να χρησιμοποιήσουμε το I2C, ενώ στην περίπτωση που έχουμε μια κατασκευή με ένα μεγαλύτερο αριθμό αισθητήρων μπορούμε να χρησιμοποιήσουμε το πρωτόκολλο SPI έτσι ώστε να μην υπάρχουν κολλήματα διευθύνσεων.

Ο BMP280 διαθέτει υψηλή ακρίβεια και γραμμικότητα, καθώς και μακροπρόθεσμη σταθερότητα στις μετρήσεις του. Είναι ένας υψηλής ακρίβειας αισθητήρας, με άριστη σχέση απόδοσης – κόστους, που μπορεί να μετρήσει την ατμοσφαιρική πίεση με απόλυτη ακρίβεια $\pm 1\text{hPa}$ και ακρίβεια στη μέτρηση της θερμοκρασίας $\pm 1.0^{\circ}\text{C}$.



Εικόνα 3.33 – Αισθητήρας BMP280 (Θερμοκρασίας, Βαρομετρικής Πίεσης)

(Πηγή: <https://electropeak.com/learn/interfacing-bmp280-barometric-pressure-sensor-module-with-arduino/>)

Τα βασικά τεχνικά χαρακτηριστικά του BMP280 περιγράφονται στον ακόλουθο πίνακα 3.2.

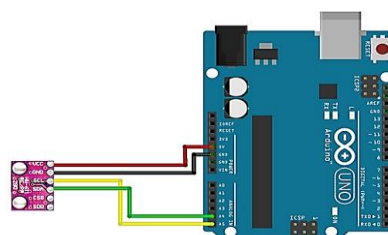
Βασικά Τεχνικά Χαρακτηριστικά	
Εύρος λειτουργίας	Πίεση: 300...1100 hPa Θερμοκρασία: -40...85°C
Απόλυτη ακρίβεια (950...1050 hPa, 0...+40°C)	~ ±1 hPa
Σχετική ακρίβεια p = 700...900hPa (Θερμ. @ 25°C)	± 0,12 hPa (τυπικό) ισοδυναμεί με ±1 m
Μέση τυπική κατανάλωση ρεύματος (ρυθμός δεδομένων 1 Hz)	3,4 μΑ @ 1 Hz
Μέση κατανάλωση ρεύματος (ρυθμός ανανέωσης δεδομένων 1 Hz)	2,74 μΑ, τυπικό (λειτουργία εξαιρετικά χαμηλής ισχύος)
Μέση κατανάλωση ρεύματος σε κατάσταση sleep mode	0,1 μΑ
Μέσος χρόνος μέτρησης	5,5 msec (προεπιλογή εξαιρετικά χαμηλής ισχύος)
Τάση τροφοδοσίας VDD	1,71 ... 3,6 V
Ανάλυση δεδομένων	Πίεση: 0,01 hPa (< 10 cm) Θερμοκρασία: 0,01°C
Θερμοκρασιακός συντελεστής offset (+25°...+40°C @900hPa)	1,5 Pa/K, ισοδύναμο με 12,6 cm/K
Interface	I ² C και SPI
Διαστάσεις αισθητήρα	8-Pin LGA με μέταλλο 2,0 x 2,5 x 0,95 mm ³

Πίνακας 3.2 – Τεχνικά χαρακτηριστικά BMP280

(Πηγή: <https://www.bosch-sensortec.com/products/environmental-sensors/pressure-sensors/bmp280/>)

3.4.6 Σύνδεση και προγραμματισμός του BMP280

Η σύνδεση του αισθητήρα BMP280 με έναν μικροελεγκτή arduino υπο παρουσιάζεται στην εικόνα 3.34.



Εικόνα 3.34 – Σύνδεση του BMP280 σε Arduino Uno

(Πηγή: <https://electropeak.com/learn/interfacing-bmp280-barometric-pressure-sensor-module-with-arduino/>)

Ο απαιτούμενος κώδικας λειτουργίας είναι ο ακόλουθος:

```
// Ενσωμάτωση απαραίτητων βιβλιοθηκών
#include "Seeed_BME280.h"
#include <Wire.h>

// Εκκίνηση του αισθητήρα
BME280 bme280;

void setup()
{

// Εκκίνηση της σειριακής επικοινωνίας
Serial.begin(115200);

// Ελέγχεται εάν έχει ξεκινήσει ο αισθητήρας
if(!bme280.init()){
    Serial.println("Device error!");
}
}

void loop()
{

// Δημιουργία μεταβλητής πίεσης
float pressure;

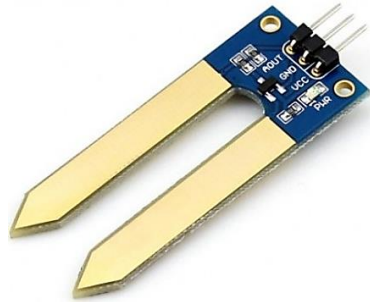
// Ζητείται και εκτυπώνεται η τιμή θερμοκρασίας
Serial.print("Temp: ");
Serial.print(bme280.getTemperature());
Serial.println("C");

// Ζητείται και εκτυπώνεται η τιμή της πίεσης
Serial.print("Pressure: ");
Serial.print(bme280.getPressure());
Serial.println("Pa");

// Μικρή χρονική καθυστέρηση
delay(1000);
}
```

3.4.7 Αναλογικός αισθητήρας υγρασίας εδάφους

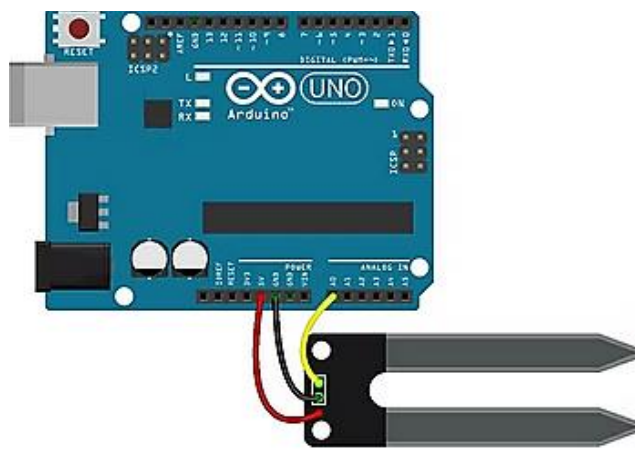
Ο αισθητήρας υγρασίας εδάφους που χρησιμοποιήθηκε είναι της εταιρείας Waveshare[3.4]. Πρόκειται για έναν απλό αισθητήρα στη χρήση. Αποτελείται από δύο μεγάλα ηλεκτρόδια τα οποία λειτουργούν ως μια μεταβλητή αντίσταση. Όσο περισσότερο νερό υπάρχει στο έδαφος τόσο καλύτερη είναι η αγωγιμότητα μεταξύ των ηλεκτροδίων του, ή αλλιώς χαμηλότερη αντίσταση άρα υψηλότερο σήμα στην έξοδο. Η συγκεκριμένη έκδοση του αισθητήρα υγρασίας εδάφους περιλαμβάνει τρεις ακροδέκτες (pins) για εύκολη καλωδίωση και εγκατάσταση.



Εικόνα 3.35 – Αναλογικός αισθητήρας Waveshare μέτρησης υγρασίας εδάφους
(Πηγή: <https://www.waveshare.com/moisture-sensor.htm>)

3.4.8 Σύνδεση και προγραμματισμός του αισθητήρα υγρασίας εδάφους

Για να λειτουργήσει ο αισθητήρας υγρασίας εδάφους WaveShare, χρειάζεται αρχικά να συνδεθούν οι ακροδέκτες VCC και GND στις αντίστοιχες υποδοχές του arduino καθώς και η αναλογική έξοδος (Aout) σε μια από τις αναλογικές εισόδους του, όπως παρουσιάζεται στην εικόνα 3.36.



Εικόνα 3.36 – Σύνδεση του αισθητήρα WaveShare σε Arduino Uno
(Πηγή: <https://create.arduino.cc/projecthub/electropeak/complete-guide-to-use-soil-moisture-sensor-w-examples-756b1f>)

Ο απαιτούμενος κώδικας λειτουργίας για τον αισθητήρα εδάφους είναι ο ακόλουθος:

```
// Καθορισμός εισόδου αναλογικού σήματος
#define SensorPin A0

// καθορισμός μεταβλητής sensorValue
float sensorValue = 0;

void setup() {
  //Εκκίνηση σειριακής επικοινωνίας
  Serial.begin(115200);
}

void loop() {

//Λαμβάνονται 100 διαδοχικές μετρήσεις και αθροίζονται στην μεταβλητή sensorValue
for (int i = 0; i <= 100; i++)
{
  sensorValue = sensorValue + analogRead(SensorPin);
  delay(1);
}

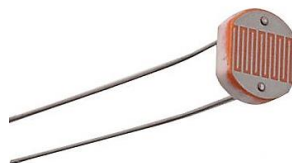
//Υπολογίζεται ο μέσος όρος της μεταβλητής sensorValue
sensorValue = sensorValue/100.0;

// Εκτύπωση μεταβλητής
  Serial.println(sensorValue);

// Μικρή καθυστέρηση
  delay(30);
}
```

3.4.9 Αναλογικός αισθητήρας φωτός – Photo resistor 5mm

Μία φωτοαντίσταση[3.5] έχει την ιδιότητα να αλλάζει την αντίστασή της ανάλογα με την ποσότητα φωτός στην οποία εκτίθεται.



Εικόνα 3.36 – Αναλογική Φωτοαντίσταση

Τα βασικά χαρακτηριστικά μιας φωτοαντίστασης παρουσιάζονται στον πίνακα 3.3.

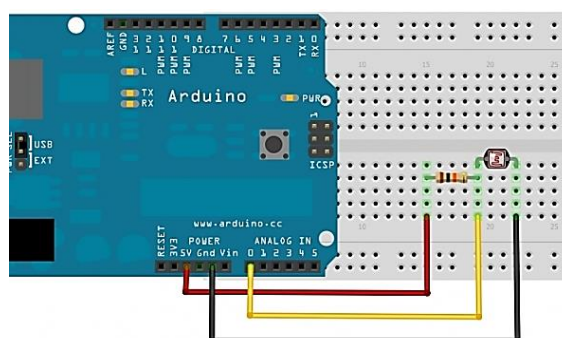
Βασικά χαρακτηριστικά

Διεπαφή:	Αναλογική
Πρωτόκολλο επικοινωνίας:	Χωρίς
Αντίσταση φωτός:	0-20KOhm
Αντοχή στο σκοτάδι:	1MOhm (Ελάχιστο)
Μέγιστη τάση:	150 V
Μέγιστη ισχύς:	100 mW
Διαστάσεις:	2 x 4 x 5mm
Βήμα μεταξύ των ακίδων:	4mm

Πίνακας 3.3 – Βασικές προδιαγραφές φωτοαντίστασης
(Πηγή: <https://grobotronics.com/photo-resistor-ldr-5mm.html>)

3.4.10 Σύνδεση και προγραμματισμός του αισθητήρα φωτός

Η σύνδεση του αισθητήρα φωτός με ένα arduino υπο παρουσιάζεται στην εικόνα 3.37. Το ένα άκρο του αισθητήρα τροφοδοτείται με θετική τάση από το arduino μέσω μιας αντίστασης 10K με παράλληλη σύνδεση σε μια από τις αναλογικές εισόδους του. Και το άλλο άκρο του γειώνεται στον μικροελεγκτή στην υποδοχή GND.



Εικόνα 3.37 – Σύνδεση φωτοαντίστασης σε Arduino Uno
(Πηγή: <https://grobotronics.com/photo-resistor-ldr-5mm.html>)

Ο απαιτούμενος κώδικας λειτουργίας της φωτοαντίστασης για το arduino είναι ο ακόλουθος:

```
void setup() {  
  // Εκκίνηση σειριακής επικοινωνίας  
  Serial.begin(115200);  
}  
  
void loop() {  
  // Ορισμός και ανάγνωση τιμής της μεταβλητής  
  int value = (analogRead(A0)*100)/1000;
```

```
// Εκτύπωση της μεταβλητής  
Serial.println("Analog Value :");  
Serial.println(value);  
  
// Μικρή καθυστέρηση  
delay(250);  
}
```

3.4.11 Ψηφιακός αισθητήρας βροχής – YL 38

Ο YL-38 που φαίνεται στην εικόνα 3.38 είναι ένας απλός αισθητήρας που μπορεί να ανιχνεύσει εάν υπάρχουν σταγονίδια νερού επάνω σε ένα τυπωμένο κύκλωμα που αποτελείται από αγωγικές λωρίδες δύο αγωγών πλεγμένες μεταξύ τους. Όταν υπάρχουν σταγονίδια νερού επάνω στην πλάκα των ηλεκτροδίων, μεταφέρεται ηλεκτρισμός από το ένα ηλεκτρόδιο προς το άλλο. Αυτή η ροή του ηλεκτρικού ρεύματος είναι ανιχνεύσιμη από τον αισθητήρα.



Εικόνα 3.38 – Ψηφιακός αισθητήρας βροχής YL-38

(Πηγή: <https://skycraftsurplus.com/products/fc-37-rain-sensor-with-yl-38-comparator-module-arduino-compatible.html>)

Ο συγκεκριμένος αισθητήρας μπορεί να λειτουργεί είτε με αναλογικό είτε με ψηφιακό τρόπο.

Για αναλογική λειτουργία συνδέεται η αναλογική έξοδος σε μια από τις αναλογικές εισόδους του arduino. Καθότι το σήμα που λαμβάνεται είναι αναλογικό παρέχεται μια ποσοστιαία ένδειξη της κατάστασης που εντοπίζει ο αισθητήρας.

Στην περίπτωση που γίνεται χρήση της ψηφιακή έξοδος λαμβάνεται ως σήμα ένα true ή false. Η ρύθμιση της ευαισθησίας σε αυτήν την περίπτωση γίνεται μέσω ενός ποτενσιόμετρου που ανήκει στον συγκριτή που βρίσκεται επάνω στην ηλεκτρονική πλακέτα που συνοδεύει τον αισθητήρα.

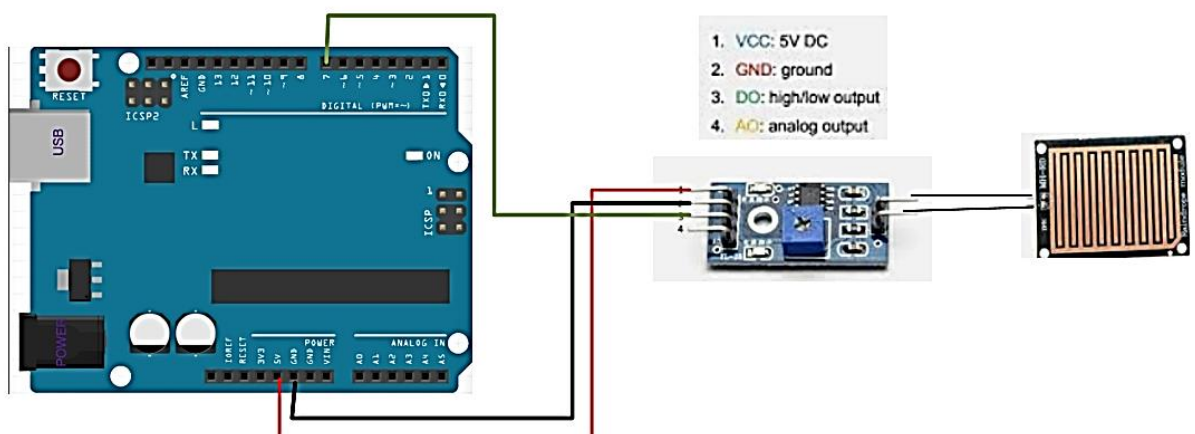
Τα τεχνικά χαρακτηριστικά του ΥL-38 περιγράφονται στον πίνακα 3.4.

Τεχνικά Χαρακτηριστικά του ΥL-38	
Ενσωματωμένος συγκριτής:	LM393
Τάση λειτουργίας:	3–5 Volt
Ρεύμα εξόδου σήματος:	15 mA
Τύπος εξόδου:	3–5 Volt (Ψηφιακό TTL)
Αναλογική Έξοδος:	0–5 Volt μέγ.
Ενσωματωμένη ένδειξη τροφοδοσίας:	LED
Ενσωματωμένη ένδειξη εξόδου:	LED
Accuracy:	Καλύτερη από +5%
Θερμοκρασία λειτουργίας:	-40 έως 80 °C
Κατανάλωση:	100 – 200 mA
Διαστάσεις:	3,2 cm x 1.4 cm

Πίνακας 3.4 - Τεχνικά Χαρακτηριστικά του ΥL-38

3.4.12 Σύνδεση και προγραμματισμός του αισθητήρα βροχής

Ο ψηφιακός τρόπος σύνδεσης του αισθητήρα ΥL-38 με ένα arduino υπο φαίνεται στην ακόλουθη εικόνα 3.39.



Εικόνα 3.39 – Σύνδεση του αισθητήρα ΥL-38 σε arduino Uno

Ο απαιτούμενος κώδικας λειτουργίας είναι ο ακόλουθος:

```
void setup() {
```

```
//Εκκίνηση σειριακής επικοινωνίας
Serial.begin(115200);
}

void loop() {
  // Καθορισμός και ανάθεση μεταβλητής
  int rain = digitalRead(7);

  // Εκτύπωση μεταβλητής
  Serial.print("Soil Humidity is: ");
  Serial.println(rain);

  //Μικρή καθυστέρηση
  delay(500);
}
```

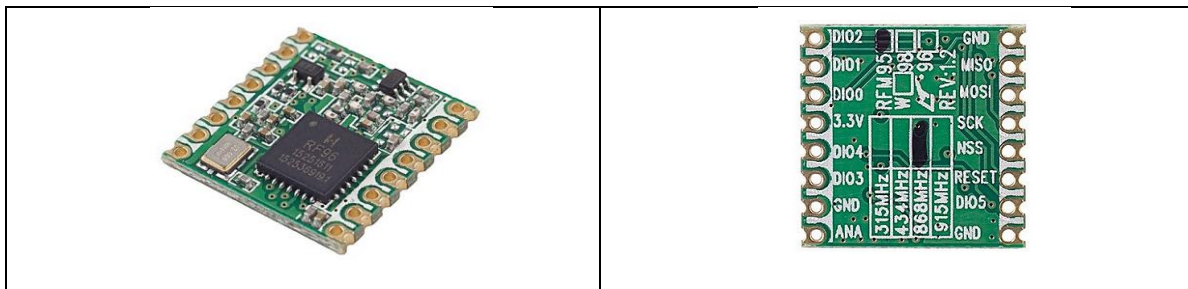
3.5 Επιλογή Transceiver και σύνδεση με έναν μικροελεγκτή Arduino Uno

Σε αυτήν την ενότητα παρουσιάζεται και δοκιμάζεται το LoRa transceiver που χρησιμοποιείται για την ασύρματη σύνδεση μεταξύ τελικών συσκευών και TTN.

3.5.1 LoRa Transceiver RFM95W

Τα δεδομένα που συλλέγονται από τους αισθητήρες πρέπει να αποστέλλονται μέσω ασύρματης σύνδεσης LoRa προς το δίκτυο TTN. Η συγκεκριμένη λειτουργία επιτυγχάνεται μέσω ενός LoRa module συνδεδεμένο σε έναν μικροελεγκτή με τον απαιτούμενο κώδικα λειτουργίας.

Για την παρούσα εργασία χρησιμοποιείται το module RFM95W που φαίνεται στην εικόνα 3.40. Το συγκεκριμένο module κατασκευάζεται από την HopeRF και χρησιμοποιεί το chipset SX1276 της Semtech. Υποστηρίζει συνολικά τέσσερις διαφορετικές συχνότητες λειτουργίας (315MHz, 434MHz, 868MHz, 915MHz). Επιλέγεται η συχνότητα των 868MHz που είναι η κατάλληλη για την περιοχή λειτουργίας των τελικών συσκευών.



Εικόνα 3.40 – LoRa Module RFM95W

(Πηγή: https://www.mobilefish.com/developer/lorawan/lorawan_quickguide_build_lora_node_rfm95_arduino_uno.html)

Ο transceiver RFM95W[3.7] διαθέτει μόντεμ μεγάλης εμβέλειας LoRaTM. Το σήμα του είναι ανθεκτικό στις παρεμβολές και έχει την χαμηλότερη κατανάλωση ρεύματος σχετικά με άλλες ανταγωνιστικές συσκευές.

Το RFM95W μπορεί να επιτύχει ευαισθησία άνω των -136 dBm χρησιμοποιώντας έναν κρύσταλλο χαμηλού κόστους. Η υψηλή ευαισθησία σε συνδυασμό με τον ενσωματωμένο ενισχυτή ισχύος +20 dBm τον καθιστά ως βέλτιστο transceiver για οποιαδήποτε εφαρμογή που απαιτεί εμβέλεια και στιβαρότητα.

Ορισμένα βασικά χαρακτηριστικά του RFM95W παρουσιάζονται στον πίνακα 3.5

Part Number	Frequency Band	Spreading Factor	Bandwidth	Effective Bitrate	Est. Sensitivity
RFM95W	868\915MHz	6 - 12	125 - 500 kHz	.293 - 37.5 kbps	-111 to -136 dBm

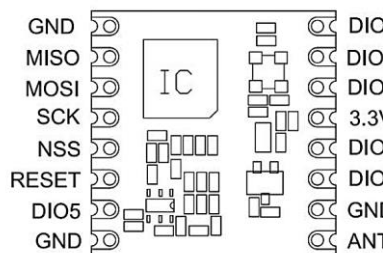
Πίνακας 3.5 – Βασικά χαρακτηριστικά του RFM95W

(Πηγή: <https://www.hoperf.com/data/upload/portal/20190801/RFM95W-V2.0.pdf>)

Τα συνολικά χαρακτηριστικά του καθορίζονται στο datasheet του κατασκευαστή στην ακόλουθη διεύθυνση:

<https://www.hoperf.com/data/upload/portal/20190801/RFM95W-V2.0.pdf>

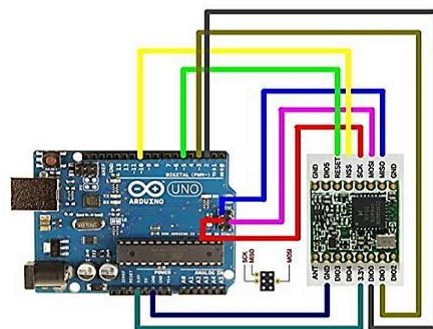
Οι ακροδέκτες του RFM95W φαίνονται στην εικόνα 3.41 και ο τρόπος σύνδεσης του RFM95W με το Arduino εμφανίζεται στην εικόνα 3.42.



Εικόνα 3.41 – Ακροδέκτες σύνδεσης του RFM95W

(Πηγή:

<https://www.hoperf.com/data/upload/portal/20190801/RFM95W-V2.0.pdf>)



Εικόνα 3.42 – Τρόπος σύνδεσης RFM95W με Arduino Uno

(Πηγή:

https://www.mobilefish.com/developer/lorawan/lorawan_quickguide_build_lora_node_rfm95_arduino_uno.html)

Τα pins διασύνδεσης μεταξύ RFM95W και arduino που αφορούν NSS, RST, DIO0, DIO1 καθορίζονται μέσα στο κώδικα λειτουργίας. Τα υπόλοιπα που έχουν να κάνουν με την θύρα

επικοινωνία SPI όπως MISO, MOSI, SCK αφορούν σταθερά σημεία (pins) στο arduino τα οποία δεν μπορούν να μεταβληθούν.

Για την πραγματοποίηση ορισμένων δοκιμών απαιτείται η διασύνδεση ενός RFM95W με ένα arduino υπο και τον απαραίτητο κώδικα λειτουργίας. Σχετικά με τον κώδικα επιλέγεται η βιβλιοθήκη LMIC (LoraWAN-MAC-in-C) που προτείνεται από το TTN και παρέχεται από την IBM. Έχει γραφτεί από τους Matthijs Kooijman, Terry Moore, ChaeHee Won, Frank Rose και υποστηρίζει δίκτυα LoRaWAN έκδοσης 1.0.2/1.0.3 για χρήση συσκευών class A.

Παρακάτω παρουσιάζεται και αναλύεται ο συγκεκριμένος κώδικας λειτουργίας.

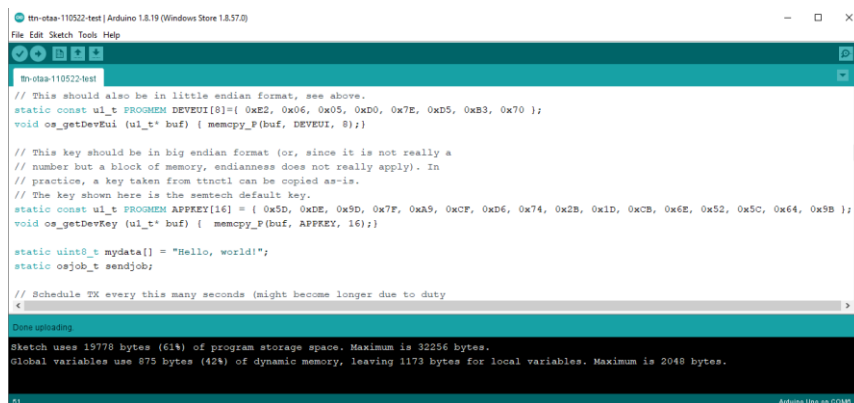
<pre>/* * Copyright (c) 2015 Thomas Telkamp and Matthijs Kooijman * * Permission is hereby granted, free of charge, to anyone * obtaining a copy of this document and accompanying files, * to do whatever they want with them without any restriction, * including, but not limited to, copying, modification and redistribution. * NO WARRANTY OF ANY KIND IS PROVIDED. * * This example sends a valid LoRaWAN packet with payload "Hello, * world!", using frequency and encryption settings matching those of * the The Things Network. * * This uses OTAA (Over-the-air activation), where where a DevEUI and * application key is configured, which are used in an over-the-air * activation procedure where a DevAddr and session keys are * assigned/generated for use with all further communication. * * Note: LoRaWAN per sub-band duty-cycle limitation is enforced (1% in * g1, 0.1% in g2), but not the TTN fair usage policy (which is probably * violated by this sketch when left running for longer)! * To use this sketch, first register your application and device with * the things network, to set or generate an AppEUI, DevEUI and AppKey. * Multiple devices can use the same AppEUI, but each device has its own * DevEUI and AppKey. * * Do not forget to define the radio type correctly in config.h. */ #include <lmic.h> #include <hal/hal.h> #include <SPI.h> // This EUI must be in little-endian format, so least-significant-byte // first. When copying an EUI from ttncpl output, this means to reverse // the bytes. For TTN issued EUIs the last bytes should be 0xD5, 0xB3, // 0x70. static const u1_t PROGMEM APPEUI[8]={ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 }; void os_getArtEui (u1_t* buf) { memcpy_P(buf, APPEUI, 8);} // This should also be in little endian format, see above. static const u1_t PROGMEM DEVEUI[8]={ 0xE2, 0x06, 0x05, 0xD0, 0x7E, 0xD5, 0xB3, 0x70 }; void os_getDevEui (u1_t* buf) { memcpy_P(buf, DEVEUI, 8);}</pre>	<p>Και τα τρία αυτά κλειδιά-κωδικοί καθορίζονται από το TTN την στιγμή που δηλώνεται η τελική συσκευή στην εφαρμογή που ανήκει.</p> <p>Τα κλειδιά αυτά έχουν δημιουργηθεί από το πρώτο βήμα της κατασκευής μας, συνεπώς τα</p>
--	--

<pre>// This key should be in big endian format (or, since it is not really a // number but a block of memory, endianness does not really apply). In // practice, a key taken from ttnc1 can be copied as-is. // The key shown here is the semtech default key. static const u1_t PROGMEM APPKEY[16] = { 0x5D, 0xDE, 0x9D, 0x7F, 0xA9, 0xCF, 0xD6, 0x74, 0x2B, 0x1D, 0xCB, 0x6E, 0x52, 0x5C, 0x64, 0x9B }; void os_getDevKey (u1_t* buf) { memcpy_P(buf, APPKEY, 16);} static uint8_t mydata[] = "Hello, world!"; static osjob_t sendjob; // Schedule TX every this many seconds (might become longer due to duty // cycle limitations). const unsigned TX_INTERVAL = 60; // Pin mapping const lmic_pinmap lmic_pins = { .nss = 10, .rxtx = LMIC_UNUSED_PIN, .rst = 5, .dio = {2, 3, LMIC_UNUSED_PIN }, }; void onEvent (ev_t ev) { Serial.print(os_getTime()); Serial.print(": "); switch(ev) { case EV_SCAN_TIMEOUT: Serial.println(F("EV_SCAN_TIMEOUT")); break; case EV_BEACON_FOUND: Serial.println(F("EV_BEACON_FOUND")); break; case EV_BEACON_MISSED: Serial.println(F("EV_BEACON_MISSED")); break; case EV_BEACON_TRACKED: Serial.println(F("EV_BEACON_TRACKED")); break; case EV_JOINING: Serial.println(F("EV_JOINING")); break; case EV_JOINED: Serial.println(F("EV_JOINED")); // Disable link check validation (automatically enabled // during join, but not supported by TTN at this time). LMIC_setLinkCheckMode(0); break; case EV_RFU1: Serial.println(F("EV_RFU1")); break;</pre>	<p>ενσωματώνονται στον κώδικα με την μορφή που τα ζητάει (msb ή lsb).</p> <p>Καθορίζεται το δοκιμαστικό μήνυμα το οποίο στέλνεται στο TTN (Hello, world!).</p> <p>Εδώ καθορίζεται μετά από πόσα δευτερόλεπτα μπορεί να στέλνει το επόμενο μήνυμα.</p> <p>Καθορίζονται ορισμένα από τα pins σύνδεσης που θα χρησιμοποιήσουμε για το RFM95W. Το NSS στο pin 10, το reset στο pin 5, το DIO0 στο 2 και το DIO1 στο 3.</p> <p>Ρουτίνα ελέγχου LoRa</p>
--	--

<pre> case EV_JOIN_FAILED: Serial.println(F("EV_JOIN_FAILED")); break; case EV_REJOIN_FAILED: Serial.println(F("EV_REJOIN_FAILED")); break; break; case EV_TXCOMPLETE: Serial.println(F("EV_TXCOMPLETE (includes waiting for RX windows)")); if (LMIC.txrxFlags & TXRX_ACK) Serial.println(F("Received ack")); if (LMIC.dataLen) { Serial.println(F("Received ")); Serial.println(LMIC.dataLen); Serial.println(F(" bytes of payload")); } // Schedule next transmission os_setTimedCallback(&sendjob, os_getTime()+sec2osticks(TX_INTERVAL), do_send); break; case EV_LOST_TSYNC: Serial.println(F("EV_LOST_TSYNC")); break; case EV_RESET: Serial.println(F("EV_RESET")); break; case EV_RXCOMPLETE: // data received in ping slot Serial.println(F("EV_RXCOMPLETE")); break; case EV_LINK_DEAD: Serial.println(F("EV_LINK_DEAD")); break; case EV_LINK_ALIVE: Serial.println(F("EV_LINK_ALIVE")); break; default: Serial.println(F("Unknown event")); break; } } void do_send(osjob_t* j){ // Check if there is not a current TX/RX job running if (LMIC.opmode & OP_TXRXPEND) { Serial.println(F("OP_TXRXPEND, not sending")); } else { // Prepare upstream data transmission at the next possible time. LMIC_setTxData2(1, mydata, sizeof(mydata)-1, 0); Serial.println(F("Packet queued")); } // Next TX is scheduled after TX_COMPLETE event. </pre>	<p>Ρουτίνα αποστολής μηνυμάτων</p>
---	------------------------------------

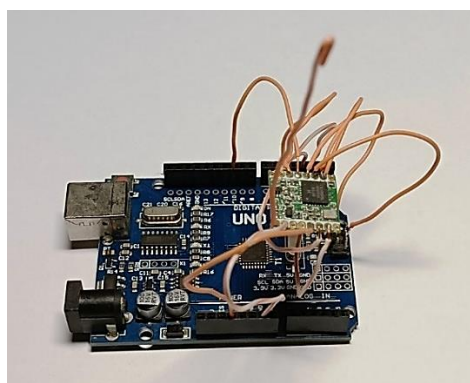
<pre> } void setup() { Serial.begin(115200); Serial.println(F("Starting")); // LMIC init os_init(); // Reset the MAC state. Session and pending data transfers will be discarded. LMIC_reset(); // Start job (sending automatically starts OTAA too) do_send(&sendjob); } void loop() { os_runloop_once(); } </pre>	<p>Αρχικές ρυθμίσεις</p>
---	--------------------------

Για την μεταφορά του παραπάνω κώδικα προς το Arduino γίνεται χρήση του Arduino IDE όπως φαίνεται στην εικόνα 3.43.



Εικόνα 3.43 – Upload κώδικα μέσω Arduino Uno

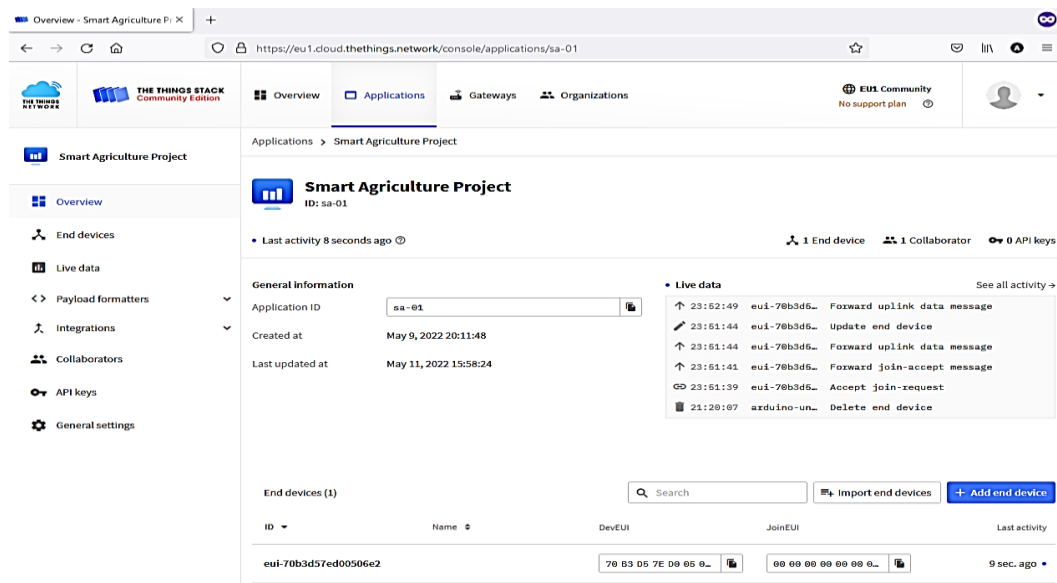
Στην εικόνα 3.44 παρουσιάζεται το arduino υπο συνδεδεμένο με το RFM95W με το τρόπο που περιγράφεται στην εικόνα 3.42, φορτωμένο με τον παραπάνω κώδικα λειτουργίας και έτοιμο για τις απαραίτητες δοκιμές.



Εικόνα 3.44 – Διασύνδεση RFM95W με Arduino Uno

Μόλις τροφοδοτηθεί με τάση ο μικροελεγκτής αρχίζει να στέλνει το μήνυμα δοκιμής ‘Hello, world’ προς το TTN κάθε 60 sec.

Από την σελίδα του TTN επιλέγεται από το βασικό μενού η διαδρομή ‘Console -> Go to Application’. Από την λίστα εφαρμογών που εμφανίζεται επιλέγεται η “smart Agriculture project” που παρουσιάζεται στην εικόνα 3.45. Από την συγκεκριμένη εικόνα παρατηρείται ότι το TTN έλαβε το πρώτο μήνυμα πριν από 8 δευτερόλεπτα (Last activity 8 seconds ago).

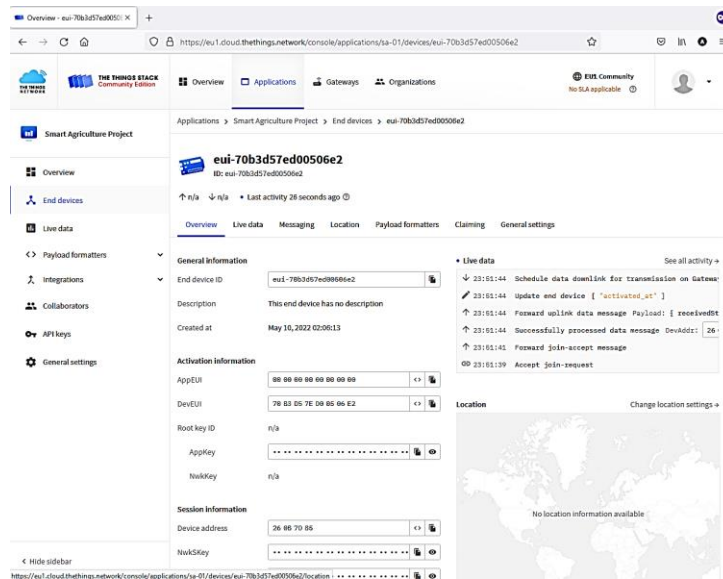


Εικόνα 3.45 – Παράθυρο εφαρμογής Smart Agriculture στο TTN

Πηγή: The Things Network

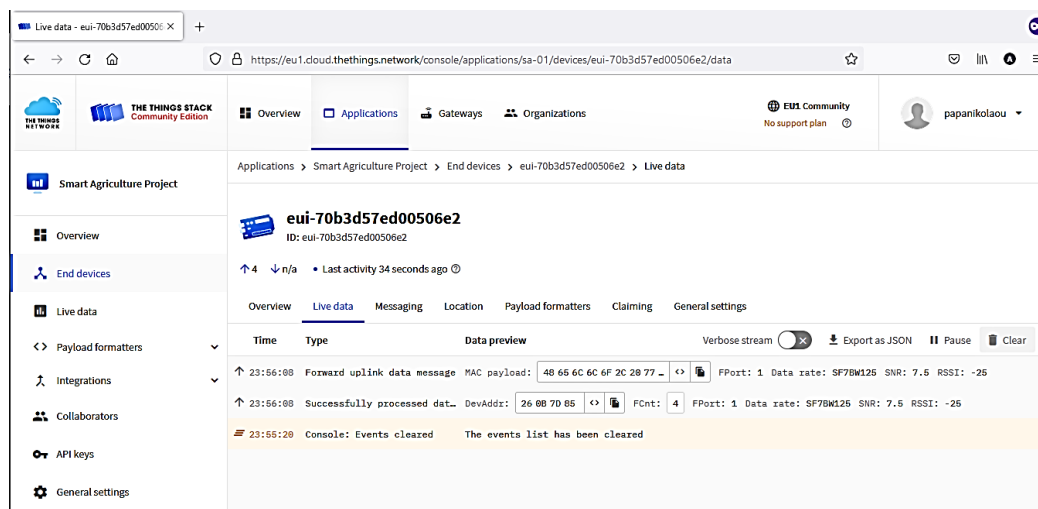
Για περισσότερη πληροφόρηση επιλέγεται με διπλό κλικ η συσκευή (eui-70b3d57ed00506e2) στο κάτω μέρος της εικόνας 3.45 από την λίστα ‘End devices’.

Εμφανίζεται η οθόνη που φαίνεται στην εικόνα 3.46 στην οποία περιλαμβάνονται γενικά στοιχεία της τελικής συσκευής μαζί με τα δεδομένα που φτάνουν προς αυτήν.



Εικόνα 3.46– Χαρακτηριστικά τελικής συσκευής στο TTN
Πηγή: The Things Network

Για μία αναλυτικότερη εικόνα των δεδομένων επιλέγεται από το κεντρικό μενού η επιλογή 'Live Data' όπου εμφανίζεται η εικόνα 3.47 στην οποία διακρίνονται τα δεδομένα σε δεκαεξαδική μορφή με την ακριβή ώρα λήψης.



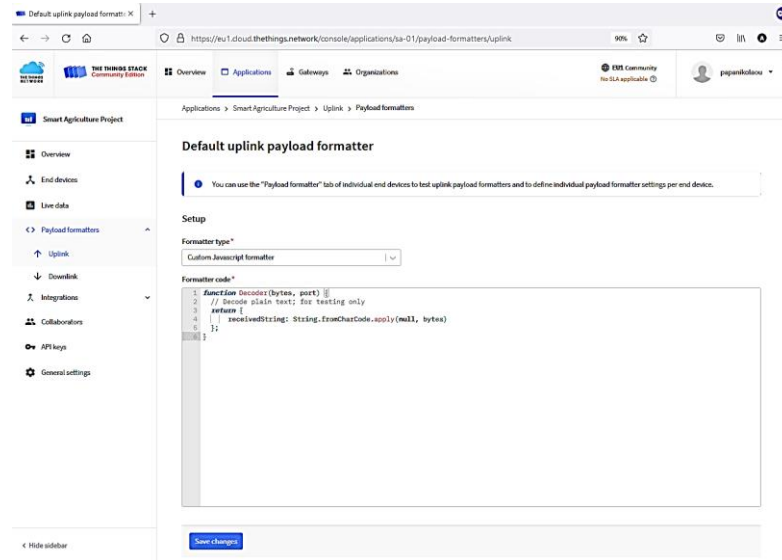
Εικόνα 3.47 – Δεδομένα τελικής συσκευής στο TTN σε πραγματικό χρόνο
Πηγή: The Things Network

Για να λαμβάνεται το σήμα υπό μορφή κειμένου και όχι υπό μορφή δεκαεξαδικής συμβολοσειράς πρέπει να χρησιμοποιηθεί μια συνάρτηση μετατροπής έτσι ώστε να γίνεται ο απαραίτητος μετασχηματισμός (από δεκαεξαδικό σε ascii).

Το TTN διαθέτει τους 'Payload formatters' μέσω των οποίων πραγματοποιείται ο μετασχηματισμός δεδομένων. Υπάρχει η δυνατότητα να γίνει χρήση διάφορων έτοιμων

τύπων μετασχηματισμού καθώς επίσης να δημιουργηθεί μια νέα ρουτίνα μετατροπής σε javascript που θα καλύπτει τις εκάστοτε απαιτήσεις.

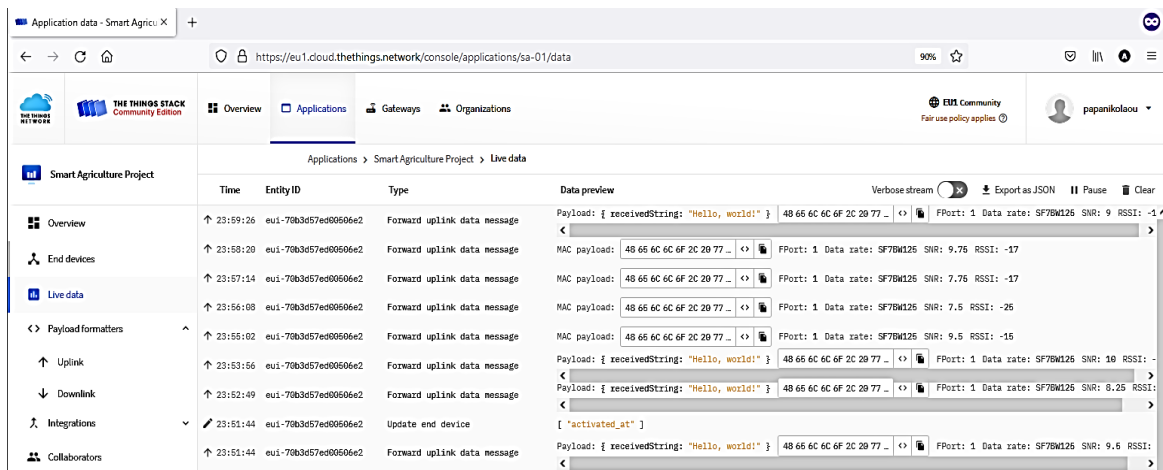
Για την δική μας περίπτωση επιλέγεται η δημιουργία μιας νέας ρουτίνας μετατροπής σε javascript όπως φαίνεται στην εικόνα 3.48.



Εικόνα 3.48 – Μετασχηματισμός δεδομένων μέσω Payload formatter στο TTN

Πηγή: The Things Network

Αφού ενημερωθεί το σύστημα με την ρουτίνα μετατροπής επιλέγεται από το κεντρικό μενού στην αριστερή στήλη η επιλογή 'Live data'. Εμφανίζεται η οθόνη της εικόνας 3.48 όπου παρατηρείται ότι τα μηνύματα εμφανίζονται υπό μορφή κειμένου "Hello, world!".



Εικόνα 3.48 – Μετασχηματισμός δεδομένων μέσω Payload formatter στο TTN

3.6 Δημιουργία πολυαισθητήρα

Αφού ολοκληρώθηκε στις ενότητες 3.4 και 3.5 η περιγραφή των διαφόρων αισθητήρων και του LoRa transceiver ως προς την λειτουργία και την διασύνδεση τους ξεχωριστά για το καθένα, ήρθε η στιγμή να συνδεθούν όλα μαζί σε έναν μικροελεγκτή με σκοπό την δημιουργία μιας τελικής συσκευής με πολλούς αισθητήρες. Δηλαδή την κατασκευή ενός πολυαισθητήρα.

Συνολικά, η τελική συσκευή θα περιλαμβάνει πέντε διαφορετικούς αισθητήρες (υγρασία εδάφους(Waveshare), υγρασία αέρα(DHT-22), θερμοκρασία εδάφους(DS18B20), ατμοσφαιρική πίεση-θερμοκρασία αέρα(BMP280), ένταση φωτός(Photo resistor 5mm)) οι οποίοι συνδέονται στον ίδιο μικροελεγκτή.

Για πρακτικούς λόγους χρησιμοποιείται ένα arduino nano στην θέση του arduino υπο καθότι σκοπός μας είναι η τελική συσκευή να είναι όσο το δυνατόν μικρότερη σε μέγεθος.

Ο κώδικας είναι ο ίδιος με αυτόν που χρησιμοποιήθηκε στην προηγούμενη παράγραφο 3.5 με την μόνη διαφορά ότι ενσωματώνεται και ο κώδικας λειτουργίας για κάθε έναν αισθητήρα που προστίθεται.

Παρακάτω ακολουθεί ο ενημερωμένος κώδικας με τις απαραίτητες προσθήκες που έγιναν για την λειτουργία των αισθητήρων καθώς επίσης και ο αναγκαίος σχολιασμός:

```
/******  
* Copyright (c) 2015 Thomas Telkamp and Matthijs Kooijman  
* https://github.com/matthijskooijman/arduino-  
lmic/blob/master/examples/ttn-otaa/ttn-otaa.ino  
* Permission is hereby granted, free of charge, to anyone  
* obtaining a copy of this document and accompanying files,  
* to do whatever they want with them without any restriction,  
* including, but not limited to, copying, modification and redistribution.  
* NO WARRANTY OF ANY KIND IS PROVIDED.  
*  
* This example sends a valid LoRaWAN packet with payload "Hello,  
* world!", using frequency and encryption settings matching those of  
* the The Things Network.  
*  
* This uses OTAA (Over-the-air activation), where where a DevEUI and  
* application key is configured, which are used in an over-the-air  
* activation procedure where a DevAddr and session keys are  
* assigned/generated for use with all further communication.  
*  
* Note: LoRaWAN per sub-band duty-cycle limitation is enforced (1% in  
* g1, 0.1% in g2), but not the TTN fair usage policy (which is probably  
* violated by this sketch when left running for longer!)  
*  
* To use this sketch, first register your application and device with  
* the things network, to set or generate an AppEUI, DevEUI and AppKey.  
* Multiple devices can use the same AppEUI, but each device has its own
```

<pre> * DevEUI and AppKey. * * Do not forget to define the radio type correctly in config.h. *****/ #include <lmic.h> #include <hal/hal.h> #include <SPI.h> #include <DHT.h> #include <DHT_U.h> #include <Adafruit_Sensor.h> #include "Seeed_BME280.h" #include <Wire.h> #include <OneWire.h> #include <DallasTemperature.h> // This EUI must be in little-endian format, so least-significant-byte // first. When copying an EUI from ttncnl output, this means to reverse // the bytes. For TTN issued EUIs the last bytes should be 0xD5, 0xB3, // 0x70. static const u1_t PROGMEM APPEUI[8] = { 0xDD, 0xCC, 0xBB, 0xAA, 0x04, 0x03, 0x02, 0x01 }; void os_getArtEui (u1_t* buf) { memcpy_P(buf, APPEUI, 8);} // This should also be in little endian format, see above. static const u1_t PROGMEM DEVEUI[8] = { 0xBA, 0x47, 0x04, 0xD0, 0x7E, 0xD5, 0xB3, 0x70 }; void os_getDevEui (u1_t* buf) { memcpy_P(buf, DEVEUI, 8);} // This key should be in big endian format (or, since it is not really a // number but a block of memory, endianness does not really apply). In // practice, a key taken from ttncnl can be copied as-is. // The key shown here is the semtech default key. static const u1_t PROGMEM APPKEY[16] = { 0x54, 0x4D, 0x9F, 0xC7, 0x44, 0x0F, 0x2F, 0x8D, 0x6C, 0xDB, 0x05, 0x13, 0x8F, 0xF8, 0xD4, 0x2E }; void os_getDevKey (u1_t* buf) { memcpy_P(buf, APPKEY, 16);} static uint8_t mydata[] = "Hello, world!"; static osjob_t sendjob; // Schedule TX every this many seconds (might become longer due to duty // cycle limitations). const unsigned TX_INTERVAL = 60; const lmic_pinmap lmic_pins = { .nss = 6, .rxtx = LMIC_UNUSED_PIN, .rst = 5, .dio = {2, 3, LMIC_UNUSED_PIN}, }; /* ----- Define Sensors ----- */ //DHT-22 #define DHT_PIN 7 #define DHTTYPE DHT22 DHT dht(DHT_PIN, DHTTYPE); </pre>	<p>Καταχωρούμε το APPEUI</p> <p>Καταχωρούμε το DEVEUI</p> <p>Καταχωρούμε το APPKEY</p> <p>Ορίζουμε το χρονικό διάστημα αποστολής μηνυμάτων</p> <p>Ορίζουμε τα pin που θα χρησιμοποιήσουμε για την σύνδεση του RFM95W</p> <p>Ορίζουμε τα pin σύνδεσης των αισθητήρων θερμοκρασίας & υγρασίας στην ψηφιακή είσοδο 7/</p>
---	--

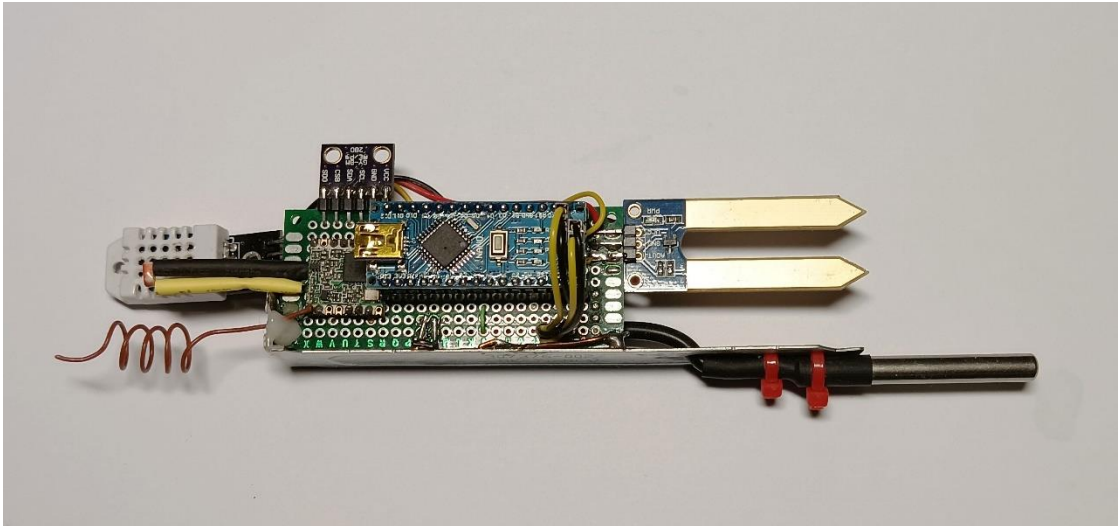
<pre> //bme-sensor BME280 bme280; //Soil Moisture Sensor #define SensorPin A2 //Soil Temperature #define ONE_WIRE_BUS 9 OneWire oneWire(ONE_WIRE_BUS); DallasTemperature sensors(&oneWire); /* ----- End-Defined Sensors ----- */ void onEvent (ev_t ev) { Serial.print(os_getTime()); Serial.print(": "); switch(ev) { case EV_SCAN_TIMEOUT: Serial.println(F("EV_SCAN_TIMEOUT")); break; case EV_BEACON_FOUND: Serial.println(F("EV_BEACON_FOUND")); break; case EV_BEACON_MISSED: Serial.println(F("EV_BEACON_MISSED")); break; case EV_BEACON_TRACKED: Serial.println(F("EV_BEACON_TRACKED")); break; case EV_JOINING: Serial.println(F("EV_JOINING")); break; case EV_JOINED: Serial.println(F("EV_JOINED")); // Disable link check validation (automatically enabled // during join, but not supported by TTN at this time). LMIC_setLinkCheckMode(0); break; /* case EV_RFU1: Serial.println(F("EV_RFU1")); break; */ case EV_JOIN_FAILED: Serial.println(F("EV_JOIN_FAILED")); break; case EV_REJOIN_FAILED: Serial.println(F("EV_REJOIN_FAILED")); break; break; case EV_TXCOMPLETE: Serial.println(F("EV_TXCOMPLETE (includes waiting for RX windows)")); if (LMIC.txxFlags & TXRX_ACK) Serial.println(F("Received ack")); if (LMIC.dataLen) { Serial.println(F("Received ")); Serial.println(LMIC.dataLen); Serial.println(F(" bytes of payload")); } } </pre>	<p>Αισθητήρας θερμοκρασίας & πίεσης</p> <p>Αισθητήρας υγρασίας εδάφους στην αναλογική είσοδο 2</p> <p>Αισθητήρας θερμοκρασίας εδάφους στην ψηφιακή είσοδο 9</p> <p>Συνάρτηση onEvent ελέγχει ότι συμβαίνει σε ένα ολοκληρωμένο κύκλο αποστολής και λήψης δεδομένων.</p>
--	---

<pre> // Schedule next transmission os_setTimedCallback(&sendjob, os_getTime()+sec2osticks(TX_INTERVAL), do_send); break; case EV_LOST_TSYNC: Serial.println(F("EV_LOST_TSYNC")); break; case EV_RESET: Serial.println(F("EV_RESET")); break; case EV_RXCOMPLETE: // data received in ping slot Serial.println(F("EV_RXCOMPLETE")); break; case EV_LINK_DEAD: Serial.println(F("EV_LINK_DEAD")); break; case EV_LINK_ALIVE: Serial.println(F("EV_LINK_ALIVE")); break; default: Serial.println(F("Unknown event")); break; } } void do_send(osjob_t* j){ // Check if there is not a current TX/RX job running if (LMIC.opmode & OP_TXRXPEND) { Serial.println(F("OP_TXRXPEND, not sending")); } else { // Prepare upstream data transmission at the next possible time. // Code for Sensor - DHT 22 int16_t t2 = dht.readTemperature() * 100; uint16_t h = dht.readHumidity() * 100; // Code for Sensor - BME280 int16_t t = bme280.getTemperature()* 100; uint32_t p = bme280.getPressure(); uint16_t alt = bme280.calcAltitude(p) * 100; p = p * 0.00750062 * 1000; // pascal to mmHg //Soil Moisture Sensor uint16_t soilMoisture = 0; for (int i = 0; i < 100; i++) { digitalWrite(8, HIGH); soilMoisture = soilMoisture + analogRead(SensorPin); delay(1); } soilMoisture = soilMoisture/100.0; //Serial.println(sensorValue); delay(30); //Soil-Temperature sensors.requestTemperatures(); // Send the command to get temperatures int16_t soilTemp = sensors.getTempCByIndex(0)*100; //Photo-Resistance </pre>	<p>Ρουτίνα αποστολής</p> <p>Ανάθεση τιμών από τους αισθητήρες σε μεταβλητές. Πριν την ανάθεση πολλαπλασιάζουμε τις τιμές τους με το 100 για να κρατήσουμε τα δύο πρώτα δεκαδικά ψηφία.</p> <p>Μετατρέπουμε την πίεση σε mmHg</p> <p>Αισθητήρας υγρασίας εδάφους. Λαμβάνουμε εκατό μετρήσεις και υπολογίζουμε τον μέσο όρο</p> <p>Λαμβάνουμε την μέτρηση θερμοκρασίας εδάφους και την αναθέτουμε σε μεταβλητή.</p>
--	---

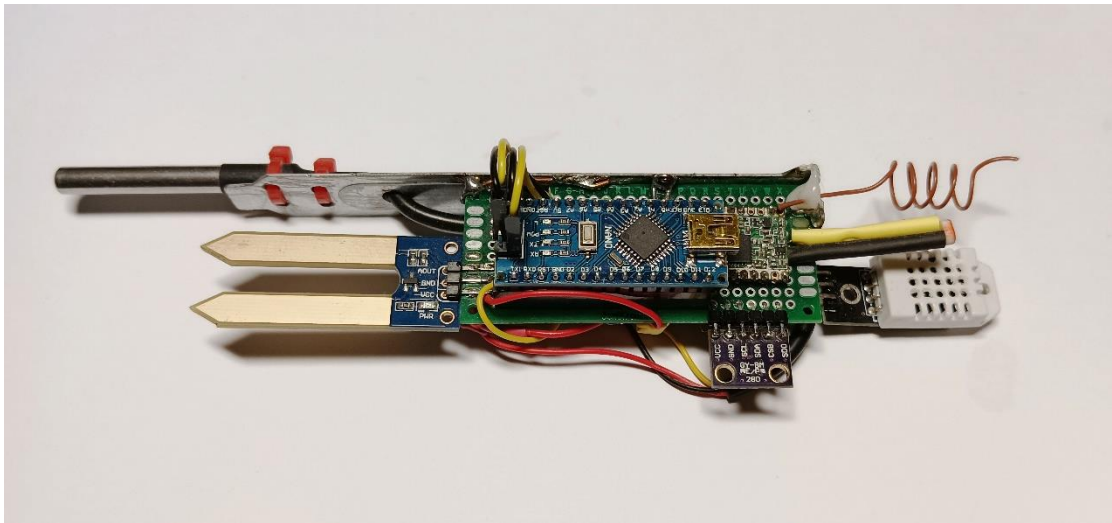
<pre> int16_t light = (analogRead(A0)*100)/1000; byte buffer[16]; buffer[0] = t2 >> 8; //Temperature buffer[1] = t2; buffer[2] = h >> 8; //Humidity buffer[3] = h; buffer[4] = alt >> 8; //Altitude buffer[5] = alt; buffer[6] = p >> 24; //32-bit number buffer[7] = p >> 16; buffer[8] = p >> 8; buffer[9] = p; buffer[10] = soilMoisture >> 8; buffer[11] = soilMoisture; buffer[12] = soilTemp >> 8; buffer[13] = soilTemp; buffer[14] = light >>8; buffer[15] = light; //Prepare upstream data transmission at the next possible time LMIC_setTxData2(1, buffer, sizeof(buffer), 0); //LMIC_setTxData2(1, mydata, sizeof(mydata)-1, 0); Serial.println(F("Packet queued")); } // Next TX is scheduled after TX_COMPLETE event. } void setup() { //Soil-Temp sensors.begin(); Serial.begin(115200); Serial.println(F("Starting")); //Check status of bme sensor (temp+pression) if(!bme280.init()){ Serial.println("Device error!"); } // LMIC init os_init(); // Reset the MAC state. Session and pending data transfers will be discarded. LMIC_reset(); // Start job (sending automatically starts OTAA too) do_send(&sendjob); } void loop() { os_runloop_once(); } </pre>	<p>Λαμβάνουμε την μέτρηση του αισθητήρα φωτός και την αναθέτουμε σε μεταβλητή.</p> <p>Ετοιμάζουμε την μεταβλητή byte που περιλαμβάνει τα δεδομένα των αισθητήρων σε δεκαεξαδική μορφή. Για όλες τις μεταβλητές εκτός από την πίεση χρησιμοποιούμε 16-bit μεταβλητή για να τις διαβάσουμε. Άρα και στην buffer δεσμεύουμε 16-bit επίσης. Για να την διαβάσουμε τον αισθητήρα πίεσης χρησιμοποιούμε μεταβλητή 32bit. Αντίστοιχα δεσμεύουμε 32-bit και στην buffer[].</p> <p>Εκκίνηση αισθητήρα υγρασίας εδάφους</p> <p>Εκκίνηση σειριακής επικοινωνίας</p> <p>Έλεγχος λειτουργίας του αισθητήρα πίεσης, θερμοκρασίας.</p>
---	---



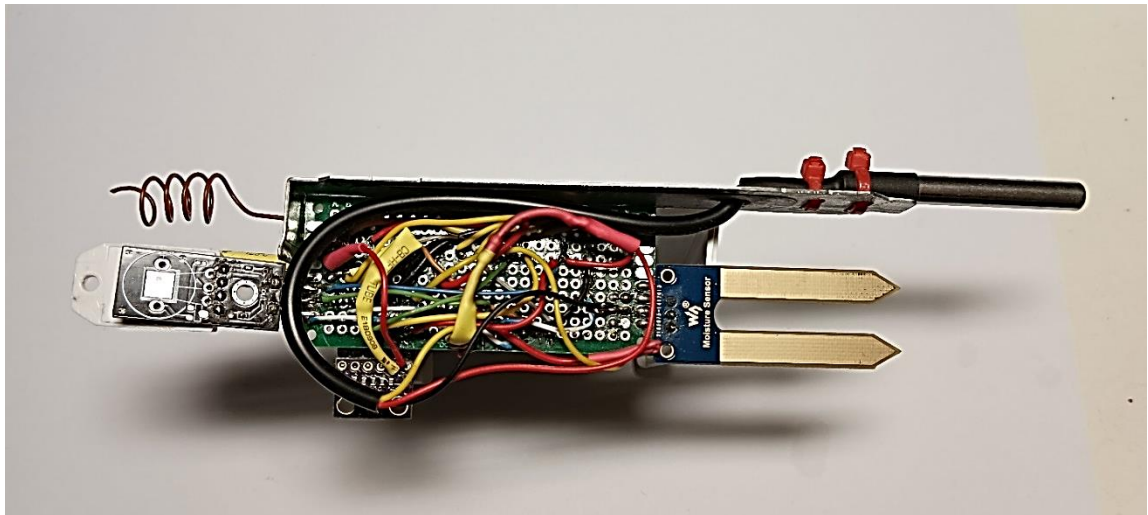
Η συνολική κατασκευή του πολυαισθητήρα παρουσιάζεται στην εικόνα 3.49 σε τέσσερις διαφορετικές όψεις (α),(β),(γ),(δ).



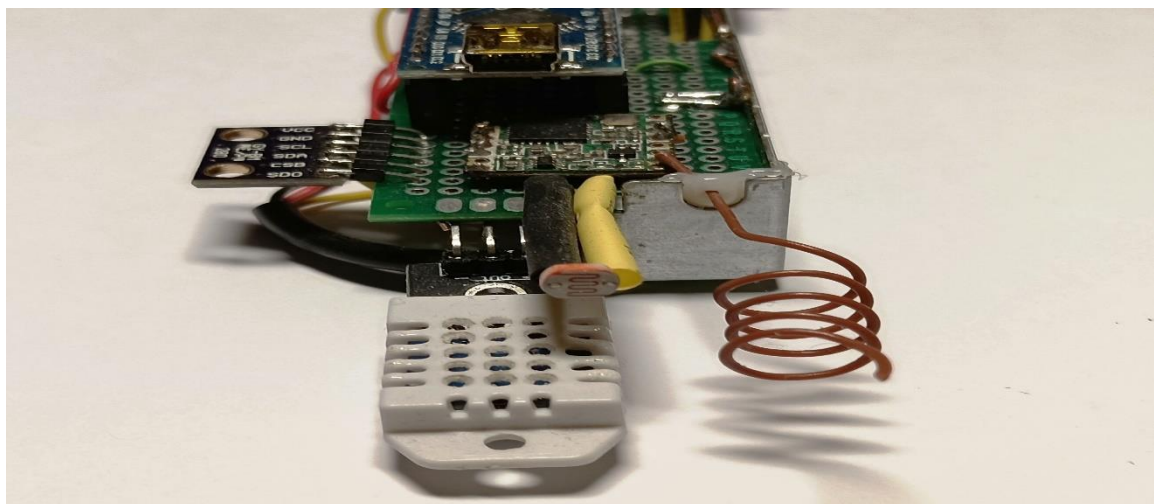
Εικόνα 3.49 (α) – Πολυαισθητήρας



Εικόνα 3.49 (β) – Πολυαισθητήρας

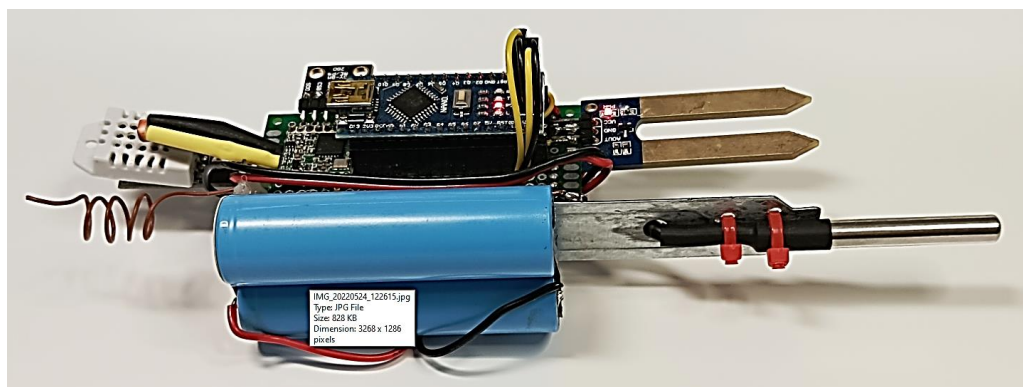


Εικόνα 3.49 (γ) – Πολυαισθητήρας

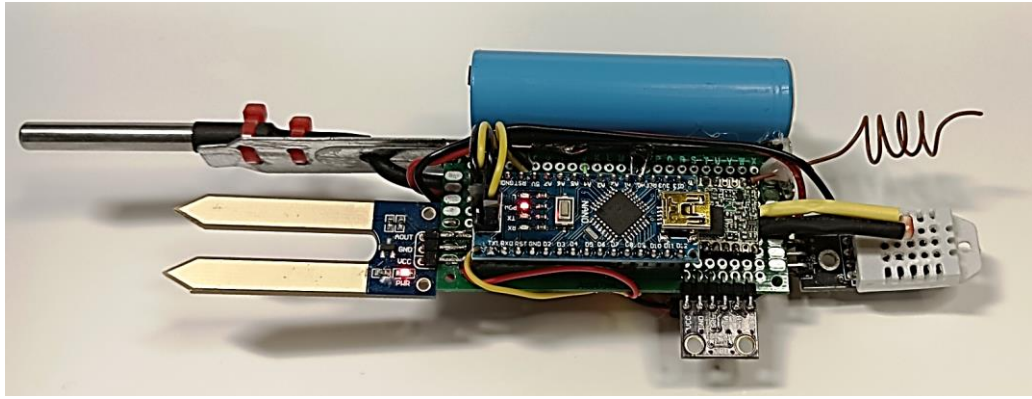


Εικόνα 3.49 (δ) – Πολυαισθητήρας

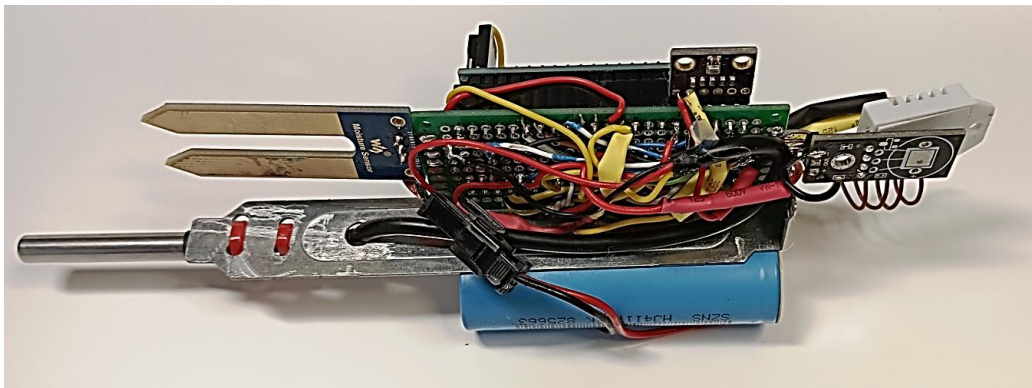
Τέλος, με την προσθήκη μιας μπαταρίας 3,3V ο πολυαισθητήρας είναι έτοιμος να τοποθετηθεί στο χώμα σε οποιοδήποτε σημείο μας ενδιαφέρει έτσι ώστε να λαμβάνονται οι απαιτούμενες μετρήσεις.



Εικόνα 3.50 (α) – Πολυαισθητήρας με μπαταρίες



Εικόνα 3.50 (β) – Πολυαισθητήρας με μπαταρίες



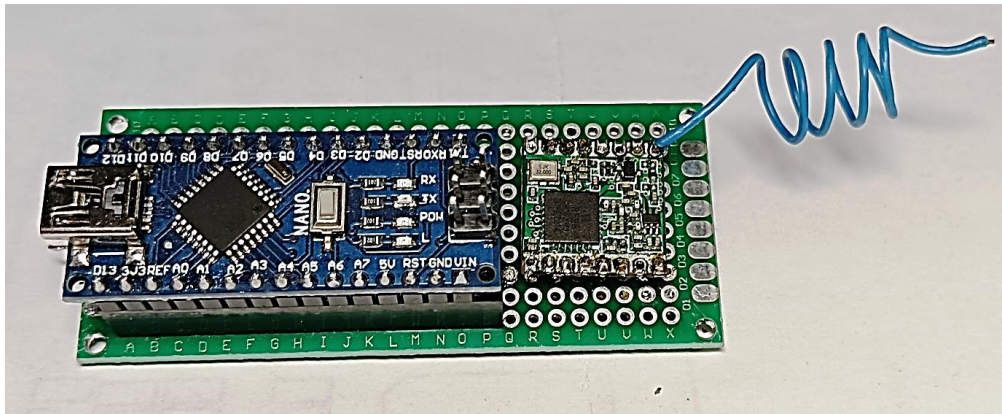
Εικόνα 3.50 (γ) – Πολυαισθητήρας με μπαταρίες

3.7 Δεύτερη τελική συσκευή - Ενεργοποιητές του συστήματος (Πότισμα, υδρονέφωση, Λάμπες θερμότητας)

Παράλληλα με τον πολυαισθητήρα μια δεύτερη τελική συσκευή θα τοποθετηθεί σε ένα σταθερό σημείο του χώρου και θα είναι υπεύθυνη για δύο λειτουργίες. Να παρέχει ενημέρωση σχετικά με την ύπαρξη βροχής από το σημείο λειτουργίας της με την βοήθεια ενός αισθητήρα ΥL-38 που παρουσιάστηκε στην παράγραφο 3.4.10 και να ελέγχει τους ενεργοποιητές του συστήματος όταν λαμβάνεται εντολή ενεργοποίησης.

3.7.1 Διαδικασία κατασκευής δεύτερης τελικής συσκευής

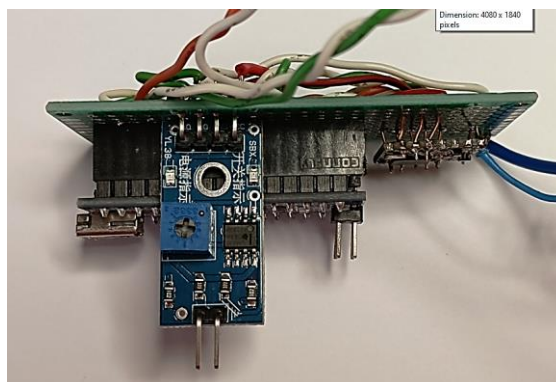
Αντίστοιχα όπως και με την προηγούμενη τελική συσκευή, γίνεται χρήση ενός arduino nano και ενός LoRa module RFM95W με τις απαραίτητες συνδέσεις μεταξύ τους που παρουσιάστηκαν στην εικόνα 3.42.



Εικόνα 3.51 – Κατασκευή δεύτερης τελικής συσκευής (Arduino nano & RFM95W)

Για να μπορεί να αντιλαμβάνεται η τελική συσκευή την βροχή προστίθεται σε αυτήν ο αισθητήρας YL-38 που παρουσιάστηκε στην παράγραφο 3.4.10. Η σύνδεση γίνεται μέσω της ψηφιακής εξόδου του αισθητήρα και συνδέοντας την στην ψηφιακή είσοδο D7 του arduino nano.

Στην εικόνα 3.52 παρουσιάζεται η τελική συσκευή με εγκατεστημένο τον αισθητήρα βροχής YL-38.



Εικόνα 3.52 – Κατασκευή δεύτερης τελικής συσκευής (Arduino nano & RFM95W & YL-38)

Τέλος, καθορίζονται στο Arduino nano οι έξοδοι που φαίνονται στον πίνακα 3.6, έτσι ώστε να μπορούν να στέλνονται μήνυμα ενεργοποίησης ή απενεργοποίησης από τον μικροελεγκτή προς τους ενεργοποιητές του συστήματος, όταν αυτό απαιτείται.

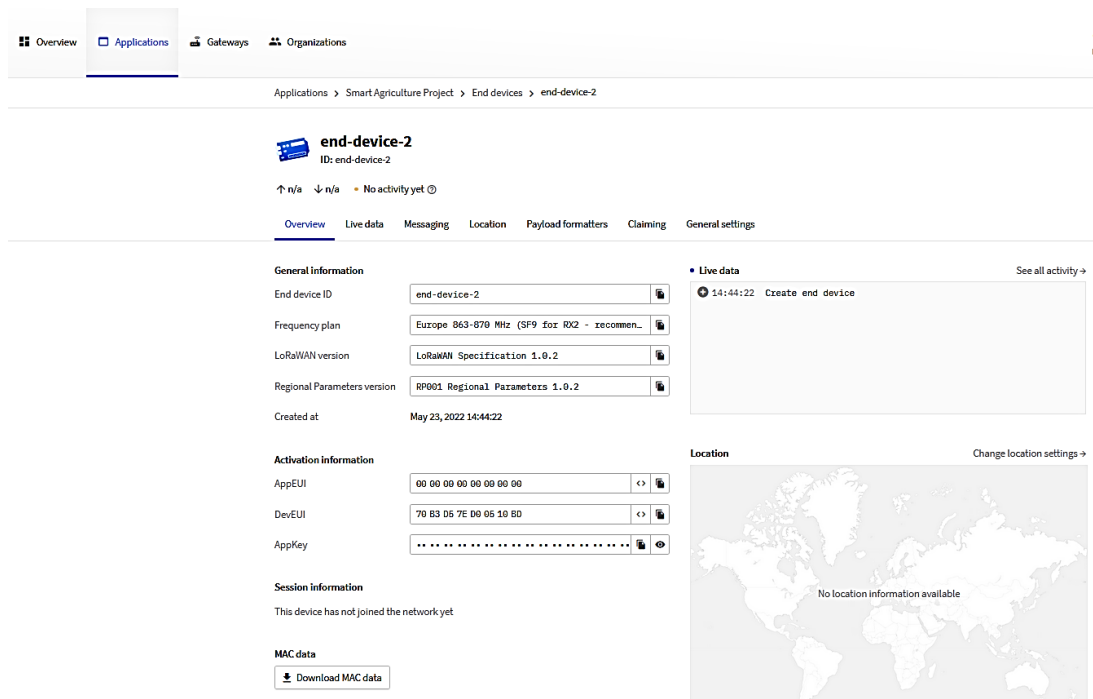
Λάμπες θερμότητας	D8
Αντλία Ποτίσματος	D9
Αντλία Υδρονέφωσης	D10

Πίνακας 3.6 – Έξοδοι ενεργοποιητών στο Arduino

3.7.2 Καταχώρηση δεύτερης τελικής συσκευής στον TTN

Για να μπορεί το TTN να δέχεται τα δεδομένα της δεύτερης τελικής συσκευής, πρέπει να γίνει καταχώρηση της συσκευής στην εφαρμογή “Smart Agriculture Project”. Πρόκειται για

ακριβώς την ίδια διαδικασία που παρουσιάστηκε στην παράγραφο 3.2.4 σχετικά με τον πολυαισθητήρα. Προς χάριν συντομίας και αφού έχουν συμπληρωθεί όλα τα απαραίτητα πεδία που είδαμε στις εικόνες 3.13 (α) & (β), ακολούθως παρουσιάζουμε απ' ευθείας την οθόνη της καταχωρημένης τελικής συσκευής στο TTN η οποία περιλαμβάνει όλα τα κατάλληλα κλειδιά που δημιουργήθηκαν κατά την διαδικασία καταχώρησης και φαίνεται στην εικόνα 3.53.



Εικόνα 3.53 – Στοιχεία δεύτερης τελικής συσκευής

Πηγή: The Things Network

3.7.3 Κώδικας λειτουργίας δεύτερης τελικής συσκευής

Ο κώδικας που χρησιμοποιείται σε μεγάλο ποσοστό είναι ο ίδιος με αυτόν που χρησιμοποιήθηκε για την πρώτη τελική συσκευή. Γίνεται απλά μια μικρή προσθήκη που έχει να κάνει με τα downlink μηνύματα, δηλαδή αυτά που αποστέλλονται από το TTN προς την τελική συσκευή για την ενεργοποίηση ορισμένων συσκευών.

<pre> /***** * Copyright (c) 2015 Thomas Telkamp and Matthijs Kooijman * https://github.com/matthijskooijman/arduino-lmic/blob/master/examples/ttn-otaa/ttn-otaa.ino * Permission is hereby granted, free of charge, to anyone * obtaining a copy of this document and accompanying files, * to do whatever they want with them without any restriction, </pre>	
---	--

<pre> * including, but not limited to, copying, modification and redistribution. * NO WARRANTY OF ANY KIND IS PROVIDED. * * This example sends a valid LoRaWAN packet with payload "Hello, * world!", using frequency and encryption settings matching those of * the The Things Network. * * This uses OTAA (Over-the-air activation), where where a DevEUI and * application key is configured, which are used in an over-the-air * activation procedure where a DevAddr and session keys are * assigned/generated for use with all further communication. * * Note: LoRaWAN per sub-band duty-cycle limitation is enforced (1% in * g1, 0.1% in g2), but not the TTN fair usage policy (which is probably * violated by this sketch when left running for longer)! * To use this sketch, first register your application and device with * the things network, to set or generate an AppEUI, DevEUI and AppKey. * Multiple devices can use the same AppEUI, but each device has its own * DevEUI and AppKey. * * Do not forget to define the radio type correctly in config.h. * *****/ #include <lmic.h> #include <hal/hal.h> #include <SPL.h> // This EUI must be in little-endian format, so least-significant-byte // first. When copying an EUI from ttncpl output, this means to reverse // the bytes. For TTN issued EUIs the last bytes should be 0xD5, 0xB3, // 0x70. static const u1_t PROGMEM APPEUI[8]={ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 }; void os_getArtEui (u1_t* buf) { memcpy_P(buf, APPEUI, 8);} // This should also be in little endian format, see above. static const u1_t PROGMEM DEVEUI[8]={ 0xBD, 0x10, 0x05, 0xD0, 0x7E, 0xD5, 0xB3, 0x70 }; void os_getDevEui (u1_t* buf) { memcpy_P(buf, DEVEUI, 8);} // This key should be in big endian format (or, since it is not really a // number but a block of memory, endianness does not really apply). In // practice, a key taken from ttncpl can be copied as-is. // The key shown here is the semtech default key. static const u1_t PROGMEM APPKEY[16] = { 0x91, 0xA9, 0x34, 0x4B, 0x6F, 0xCB, 0xCF, 0x4B, 0x1C, 0xA1, 0x2B, 0x69, 0x25, 0xFE, 0x22, 0x29 }; void os_getDevKey (u1_t* buf) { memcpy_P(buf, APPKEY, 16);} </pre>	<p>Καταχωρούμε το APPEUI</p> <p>Καταχωρούμε το DEVEUI</p> <p>Καταχωρούμε το APPKEY</p>
---	--

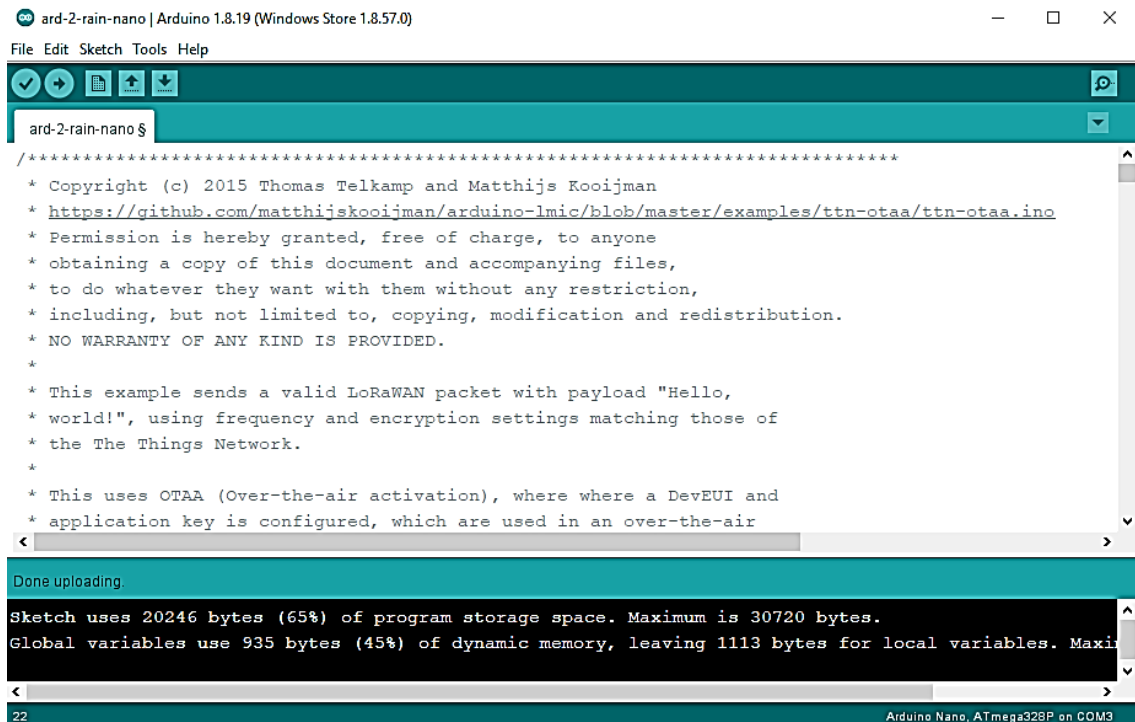
<pre> static osjob_t sendjob; // Schedule TX every this many seconds // (might become longer due to duty cycle limitations). const unsigned TX_INTERVAL = 60; // Pin mapping const lmic_pinmap lmic_pins = { .nss = 6, .rxtx = LMIC_UNUSED_PIN, .rst = 5, .dio = {2, 3, LMIC_UNUSED_PIN}, }; /* ----- Define Sensors ----- */ int rain = digitalRead(7); /* ----- Define Actuators ----- */ #define LIGHT 8 #define WATER_PUMP 9 #define YDRO_NEFOSI 10 int time_counter; void onEvent(ev_t ev) { Serial.print(os_getTime()); Serial.print(": "); switch(ev) { case EV_SCAN_TIMEOUT: Serial.println(F("EV_SCAN_TIMEOUT")); break; case EV_BEACON_FOUND: Serial.println(F("EV_BEACON_FOUND")); break; case EV_BEACON_MISSED: Serial.println(F("EV_BEACON_MISSED")); break; case EV_BEACON_TRACKED: Serial.println(F("EV_BEACON_TRACKED")); break; case EV_JOINING: Serial.println(F("EV_JOINING")); break; case EV_JOINED: Serial.println(F("EV_JOINED")); // Disable link check validation (automatically enabled // during join, but not supported by TTN at this time). LMIC_setLinkCheckMode(0); break; /* case EV_RFU1: Serial.println(F("EV_RFU1")); break; */ case EV_JOIN_FAILED: Serial.println(F("EV_JOIN_FAILED")); } } </pre>	<p>Ορίζουμε το χρονικό διάστημα αποστολής μηνυμάτων</p> <p>Ορίζουμε τα pin που θα χρησιμοποιήσουμε για την σύνδεση του RFM95W</p> <p>Ορίζουμε το pin σύνδεσης του αισθητήρα βροχής</p> <p>Ορίζουμε τα pins σύνδεσης των ενεργοποιητών.</p> <p>Συνάρτηση onEvent ελέγχει ότι συμβαίνει σ' ένα ολοκληρωμένο κύκλο αποστολής και λήψης δεδομένων.</p>
---	--

<pre> break; case EV_REJOIN_FAILED: Serial.println(F("EV_REJOIN_FAILED")); break; break; case EV_TXCOMPLETE: Serial.println(F("EV_TXCOMPLETE (includes waiting for RX windows)")); if (LMIC.txrxFlags & TXRX_ACK) Serial.println(F("Received ack")); if (LMIC.dataLen) { Serial.println(F("Received ")); Serial.println(LMIC.dataLen); Serial.println(F(" bytes of payload")); /* Send Actuators Data */ if (LMIC.dataLen == 1) { uint8_t result = LMIC.frame[LMIC.dataBeg + 0]; if (result == 0) { Serial.println("RESULT 0"); digitalWrite(LIGHT, LOW); //8 digitalWrite(WATER_PUMP, LOW); //9 digitalWrite(YDRO_NEFOSI, LOW); //10 } if (result == 1) { Serial.println("RESULT 1"); digitalWrite(LIGHT, LOW); digitalWrite(WATER_PUMP, LOW); digitalWrite(YDRO_NEFOSI, HIGH); } if (result == 2) { Serial.println("RESULT 2"); digitalWrite(LIGHT, LOW); digitalWrite(WATER_PUMP, HIGH); digitalWrite(YDRO_NEFOSI, LOW); } if (result == 3) { Serial.println("RESULT 3"); digitalWrite(LIGHT, LOW); digitalWrite(WATER_PUMP, HIGH); digitalWrite(YDRO_NEFOSI, HIGH); } if (result == 4) { Serial.println("RESULT 4"); digitalWrite(LIGHT, HIGH); digitalWrite(WATER_PUMP, LOW); digitalWrite(YDRO_NEFOSI, LOW); } if (result == 5) { Serial.println("RESULT 5"); digitalWrite(LIGHT, HIGH); digitalWrite(WATER_PUMP, LOW); digitalWrite(YDRO_NEFOSI, HIGH); } if (result == 6) { </pre>	<p>Μόλις ολοκληρωθεί η αποστολή των δεδομένων ακολουθούν δύο παράθυρα λήψης.</p> <p>Εδώ ελέγχουμε εάν το παράθυρο λήψης έχει δεδομένα.</p> <p>Ελέγχουμε το μήκος της συμβολοσειράς των δεδομένων και το μεταβιβάζουμε σε μια μεταβλητή.</p> <p>Οι τρεις ενεργοποιητές δημιουργούν επτά διαφορετικές καταστάσεις. Με ένα πολλαπλό βρόγχο if ελέγχουμε την μεταβλητή result και ενεργοποιούμε κάθε φορά τους κατάλληλους ενεργοποιητές.</p>
--	---

<pre> Serial.println("RESULT 6"); digitalWrite(LIGHT, HIGH); digitalWrite(WATER_PUMP, HIGH); digitalWrite(YDRO_NEFOSI, LOW); } if (result == 7) { Serial.println("RESULT 7"); digitalWrite(LIGHT, HIGH); digitalWrite(WATER_PUMP, HIGH); digitalWrite(YDRO_NEFOSI, HIGH); } if (result == 1 result == 2 result == 3 result == 5 result == 6 result == 7){ time_counter = 20000; Serial.println(time_counter); } } Serial.println(); //----- } // Schedule next transmission os_setTimedCallback(&sendjob, os_getTime()+sec2osticks(TX_INTERVAL), do_send); break; case EV_LOST_TSYNC: Serial.println(F("EV_LOST_TSYNC")); break; case EV_RESET: Serial.println(F("EV_RESET")); break; case EV_RXCOMPLETE: // data received in ping slot Serial.println(F("EV_RXCOMPLETE")); break; case EV_LINK_DEAD: Serial.println(F("EV_LINK_DEAD")); break; case EV_LINK_ALIVE: Serial.println(F("EV_LINK_ALIVE")); break; default: Serial.println(F("Unknown event")); break; } } void do_send(osjob_t* j){ // Check if there is not a current TX/RX job running if (LMIC.opmode & OP_TXRXPEND) { Serial.println(F("OP_TXRXPEND, not sending")); } else { // Prepare upstream data transmission at the next possible time. int8_t rr = rain; byte buffer[1]; buffer[0] = rr; </pre>	<p>Εφαρμογή χρονοκαθυστέρησης. Όταν ενεργοποιούνται οι αντλίες υδρονέφωσης και ποτίσματος καθορίζεται ένα χρονικό διάστημα ενεργοποίησης. Αυτό γίνεται με την μεταβλητή time_counter</p>
--	--

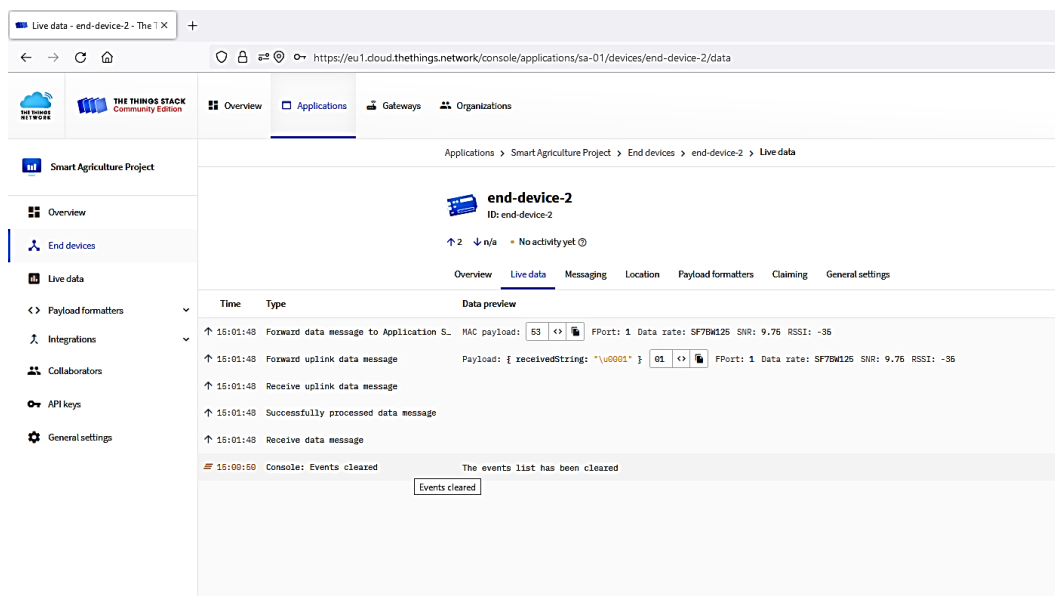
<pre> //Prepare upstream data transmission at the next possible time LMIC_setTxData2(1, buffer, sizeof(buffer), 0); Serial.println(F("Packet queued")); } // Next TX is scheduled after TX_COMPLETE event. } void setup() { Serial.begin(115200); Serial.println(F("Starting")); /* ----- Setup Actuators ----- */ pinMode(WATER_PUMP,OUTPUT); pinMode(YDRO_NEFOSI,OUTPUT); pinMode(LIGHT, OUTPUT); // LMIC init os_init(); // Reset the MAC state. Session and pending data transfers will be discarded. LMIC_reset(); // Start job (sending automatically starts OTAA too) do_send(&sendjob); } void loop() { if (time_counter > 0){ time_counter--; Serial.println(time_counter); } else if (time_counter <= 0){ digitalWrite(WATER_PUMP, LOW); digitalWrite(YDRO_NEFOSI, LOW); //Serial.println(time_counter); } } os_runloop_once();} </pre>	<p>Προετοιμασία ανοδικού μηνύματος από τον αισθητήρα βροχής. Δημιουργούμε μια μεταβλητή 8-bit (rr) και της αναθέτουμε την τιμή της μεταβλητής rain. Στην συνέχεια την φορτώνουμε σε μια μεταβλητή τύπου byte και αποστέλλεται με το επόμενο μήνυμα.</p> <p>Ορίζουμε την σειριακή επικοινωνία</p> <p>Καθορίζουμε τα pin των ενεργοποιητών ως σήματα εξόδου</p> <p>Μειώνουμε την τιμή της μεταβλητής time_counter κατά ένα κάθε φορά που το πρόγραμμα περνάει από αυτό το σημείο για να σταματήσει το πότισμα και η υδρονέφωση.</p>
---	---

Ο κώδικας μεταφέρεται στο arduino nano με την χρήση του arduino IDE όπως φαίνεται στην εικόνα 3.54.



Εικόνα 3.54 – Upload κώδικα δεύτερης τελικής συσκευής

Στην συνέχεια αφού πρώτα τροφοδοτηθεί η τελική συσκευή με τάση, ελέγχεται αν τα δεδομένα φτάνουν στο TTN. Από την οθόνη της εικόνας 3.53 επιλέγεται 'Live data' και εμφανίζεται η εικόνα 3.55 όπου παρουσιάζονται τα δεδομένα που στέλνει ο αισθητήρας της βροχής σε δεκαεξαδική μορφή με την ώρα λήψης.



Εικόνα 3.55 – Live data δεύτερης τελικής συσκευής

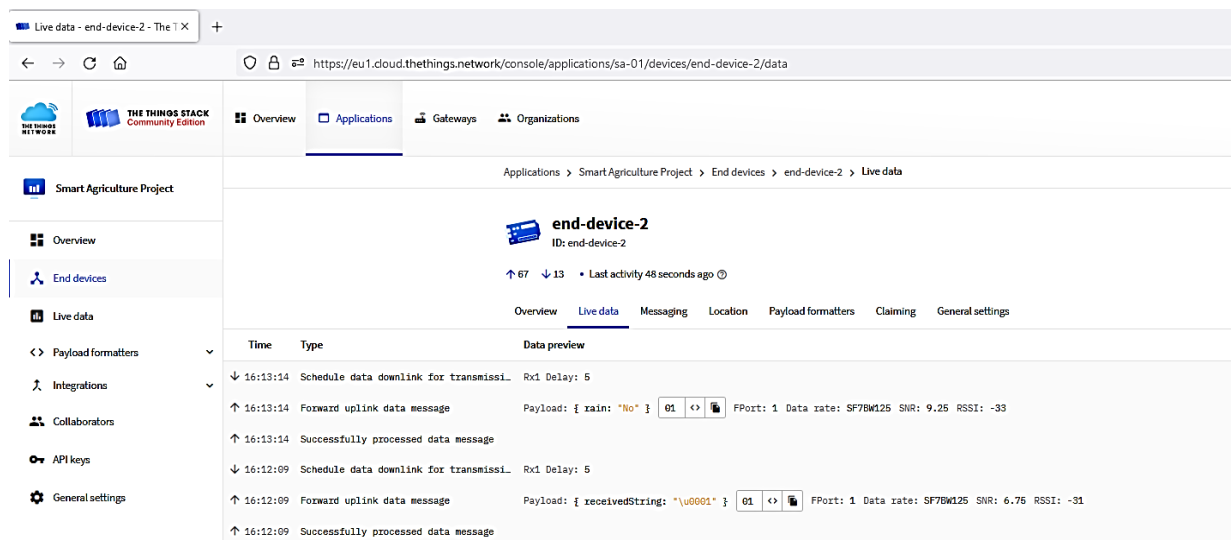
Πηγή: The Things Network

Για την μετατροπή των δεδομένων από δεκαεξαδικό σε κείμενο (ascii) απαιτείται να καταχωρηθεί μια συνάρτηση μορφοποίησης σε javascript μέσω της επιλογής “Payload formatters”, όπως ακριβώς έγινε και για την πρώτη τελική συσκευή.

Ακολουθως παρουσιάζεται η συνάρτηση για τα δεδομένα της βροχής:

```
function Decoder(bytes, port) {  
  //Ανάθεση τιμής σε μεταβλητή  
  var r = bytes[0];  
  // Εάν το δεδομένο είναι 1 βγάζει έξοδο rain: 'No' αλλιώς αν είναι 0 βγάζει έξοδο rain: 'Yes'  
  // Σε κάθε άλλη περίπτωση βγάζει rain: 'Undefined'  
  if (r==1){  
    return {rain: 'No'};  
  }else if (r === 0) {  
    return {rain: 'Yes'};  
  } else {  
    return {rain:'Undefined'};  
  }  
}
```

Επιβεβαιώνεται στην εικόνα 3.56 που ακολουθεί ότι τα δεδομένα του δεύτερου αισθητήρα εμφανίζονται υπό μορφή κειμένου.



Εικόνα 3.56 – Upload κώδικα δεύτερης τελικής συσκευής

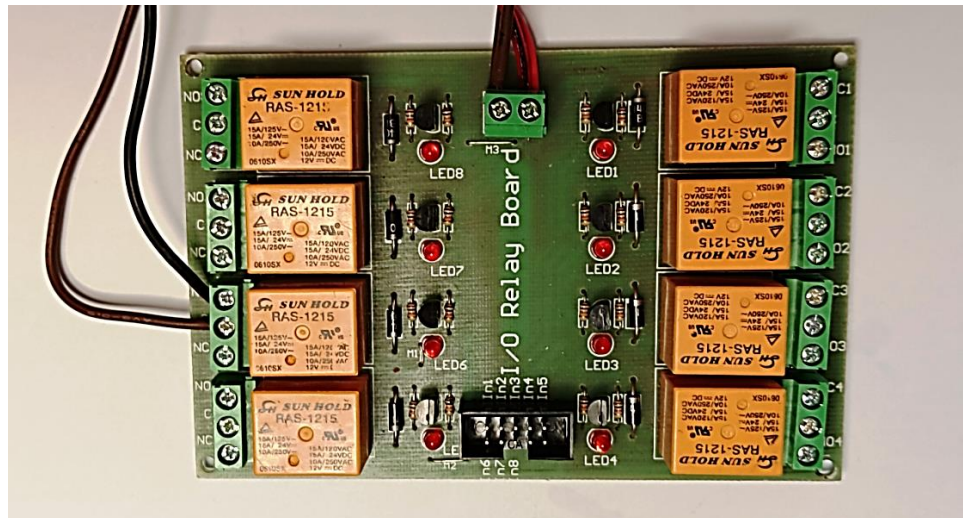
Πηγή: The Things Network

3.8 Σύνδεση ενεργοποιητών συστήματος

Για την ενεργοποίηση του ποτίσματος, της υδρονέφωσης και της θερμαντικής λυχνίας χρησιμοποιούνται ενεργοποιητές που λειτουργούν με τάση 12V. Για να μπορεί ο

μικροελεγκτής μέσω μιας εντολής να ενεργοποιεί και να απενεργοποιεί τις συγκεκριμένες συσκευές συνδέεται η πλακέτα που φαίνεται στην εικόνα 3.57.

Η συγκεκριμένη πλακέτα λειτουργεί με τάση 12V και διαθέτει συνολικά 8 ρελέ μέσω των οποίων πραγματοποιείται η τροφοδοσία των ενεργοποιητών από μια εξωτερική πηγή. Συνολικά πρόκειται να χρησιμοποιηθούν μόνο τα τρία από αυτά καθότι η κατασκευή περιλαμβάνει τρεις διαφορετικούς ενεργοποιητές.



Εικόνα 3.57 – Module Ρελέ 12V με 8 κανάλια λειτουργίας

Στην κατασκευή δεν είναι δυνατόν να χρησιμοποιηθούν κανονικές λάμπες θέρμανσης καθότι απαιτούν μεγάλη κατανάλωση ισχύος. Όμως, επειδή μας ενδιαφέρει να ελέγξουμε το κομμάτι του αυτοματισμού σχετικά με την ενεργοποίηση και την απενεργοποίηση κατά την λειτουργία του συστήματος χρησιμοποιείται αντί της θερμαντικής λάμπας μια λυχνία led με τάση λειτουργίας τα 12V όπως αυτή που φαίνεται στην ακόλουθη εικόνα 3.58 .



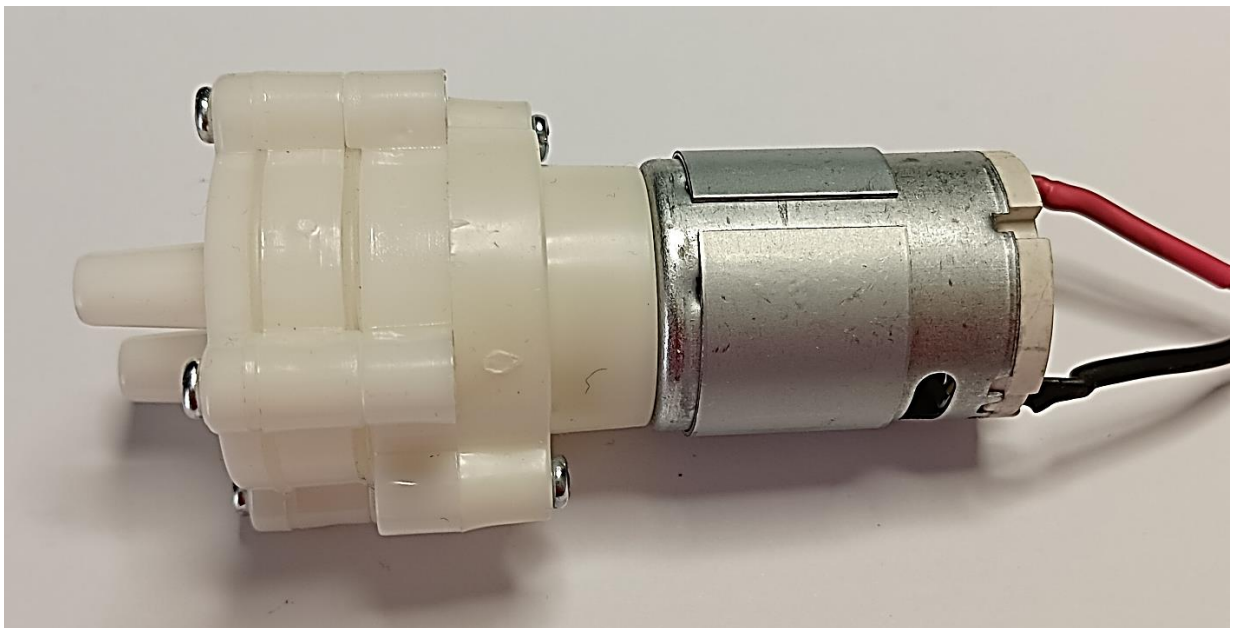
Εικόνα 3.58 – Θερμαντική Λυχνία τύπου LED

Αντίστοιχα, μια κανονική αντλία υδρονέφωσης απαιτεί μεγάλη πίεση νερού και δεν είναι εφικτό να δουλέψει με τάση 12V και ένταση 1A. Για τον λόγο αυτόν αντικαθίσταται με ένα ενδεικτικό λαμπάκι led όπως αυτό της εικόνας 3.59 .



Εικόνα 3.59 – Ενδεικτική λυχνία υδρονέφωσης

Η αντλία ποτίσματος που χρησιμοποιείται φαίνεται στην εικόνα 3.60, λειτουργεί με τάση 12V και ενεργοποιείται με ένα από τα ρελέ της πλακέτας που είδαμε στην εικόνα 3.57.



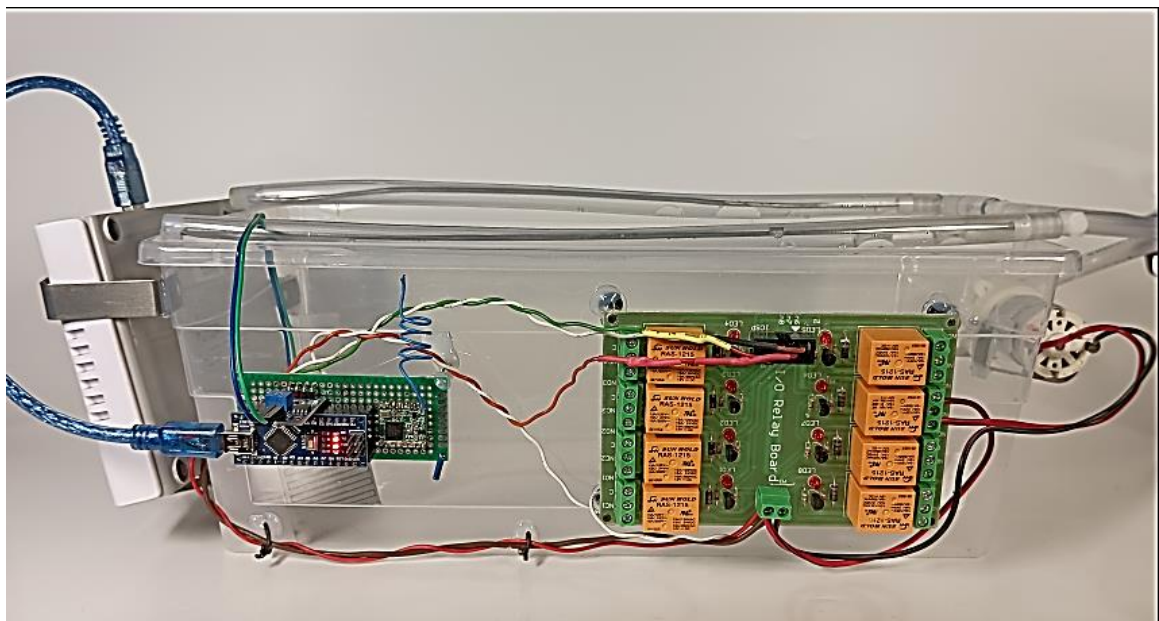
Εικόνα 3.60 – Αντλία Ποτίσματος 12V

Η τροφοδοσία της δεύτερης τελικής συσκευής καθώς και των ενεργοποιητών που συνδέονται σε αυτήν γίνεται μέσω ενός powerbank το οποίο φαίνεται στην εικόνα 3.61. Μας παρέχει διπλή έξοδο, 12V - 1A μέσω ενός θηλυκού βύσματος των 2,1mm και 5V μέσω ενός θηλυκού USB.



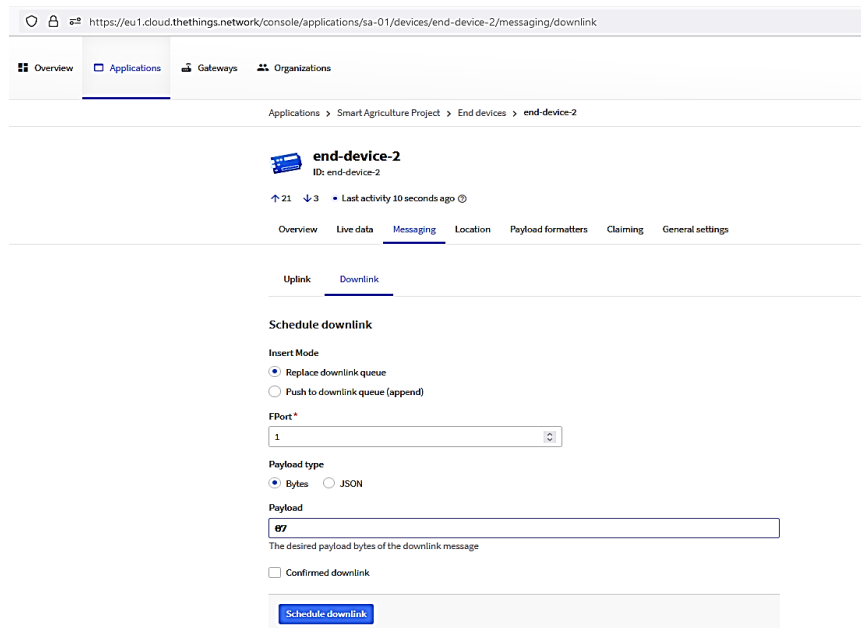
Εικόνα 3.61 – Power Bank 12v – 5V

Η τοποθέτηση της δεύτερης τελικής συσκευής μαζί με την πλακέτα ενεργοποίησης των ενεργοποιητών παρουσιάζεται στην εικόνα 3.62.



Εικόνα 3.62 – Τοποθέτηση δεύτερης τελικής συσκευής με ορισμένους ενεργοποιητές

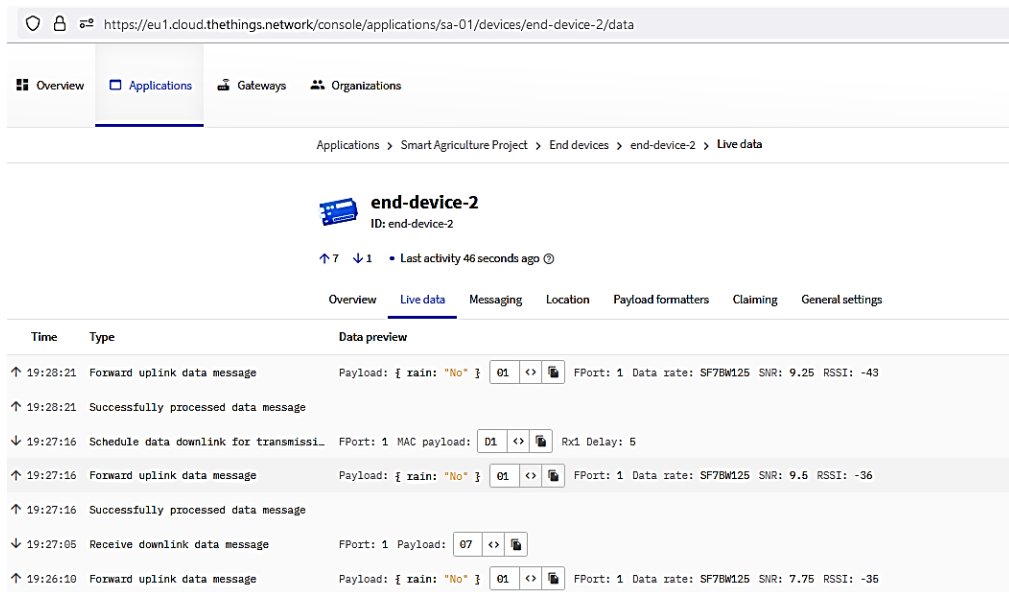
Για την πραγματοποίηση δοκιμών σχετικά με την λειτουργία downlink, στέλνεται δεκαεξαδικό 07 από το TTN προς την τελική συσκευή. Το δεκαεξαδικό 07 με βάση τον κώδικα λειτουργίας ενεργοποιεί και τα τρία ρελέ. Η αποστολή γίνεται από το TTN, μέσω του κεντρικού μενού της εφαρμογής “Smart agriculture”. Επιλέγεται Messaging → Downlink και στο payload δίνεται η τιμή 07. Για αποστολή επιλέγεται “schedule downlink” (εικόνα 3.63)



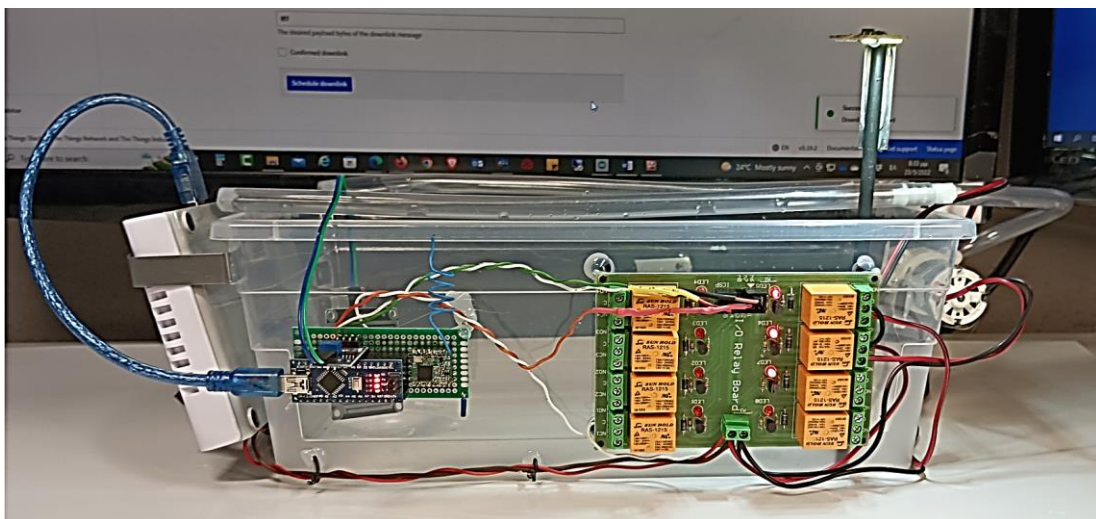
Εικόνα 3.63 – Δοκιμή downlink στην δεύτερη τελική συσκευή
Πηγή: The Things Network

Καθότι η τελική συσκευή που χρησιμοποιείται ανήκει στην κλάση A, λαμβάνει δεδομένα κατά την διάρκεια των δύο παραθύρων που ανοίγουν μετά από κάθε αποστολή δεδομένων (Κεφ2 - §2.4.4). Έτσι, για να πραγματοποιηθεί λήψη απαιτείται πρώτα η αποστολή ενός uplink μηνύματος προς το TTN από την τελική συσκευή.

Στην εικόνα 3.64 φαίνεται ότι το μήνυμα αποστέλλεται στις 19:27:05 (Receive downlink data message) μέσω της διαδικασίας που φαίνεται στην εικόνα 3.63. Στις 19:27:16 το μήνυμα λαμβάνεται από την τελική συσκευή κατά το πρώτο παράθυρο λήψης Rx1 (Schedule data downlink for transmission) και αμέσως ενεργοποιούνται τα τρία ρελέ που είναι υπεύθυνα για τους τρεις ενεργοποιητές (εικόνα 3.65).

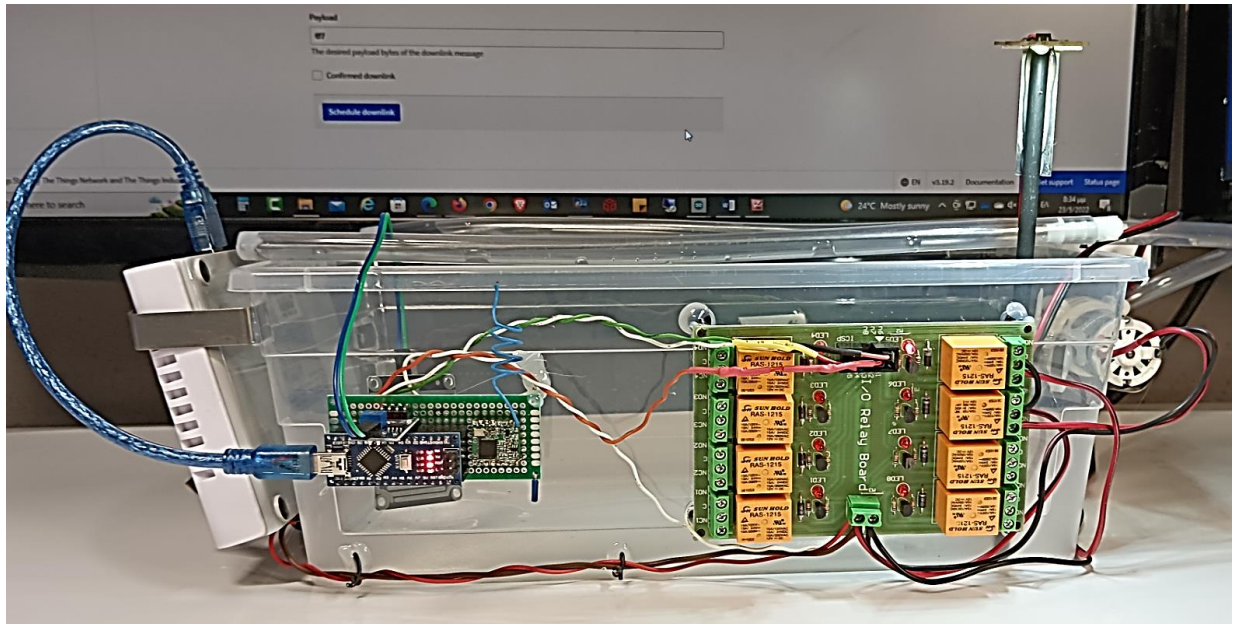


Εικόνα 3.64 – Live Data δεύτερης τελικής συσκευής – Μήνυμα Downlink
Πηγή: The Things Network



Εικόνα 3.65 – Τελική συσκευή μετά από καθοδικό μήνυμα 07 Hex.

Μετά από μερικά δευτερόλεπτα μηδενίζεται η μεταβλητή `time_counter` που αφορά το χρόνο ποτίσματος και υδρονέφωσης με αποτέλεσμα να απενεργοποιηθούν τα δύο ρελέ και να μείνει ενεργοποιημένο μόνο το ρελέ που αφορά την θερμαντική λυχνία όπως φαίνεται στην ακόλουθη εικόνα 3.66.

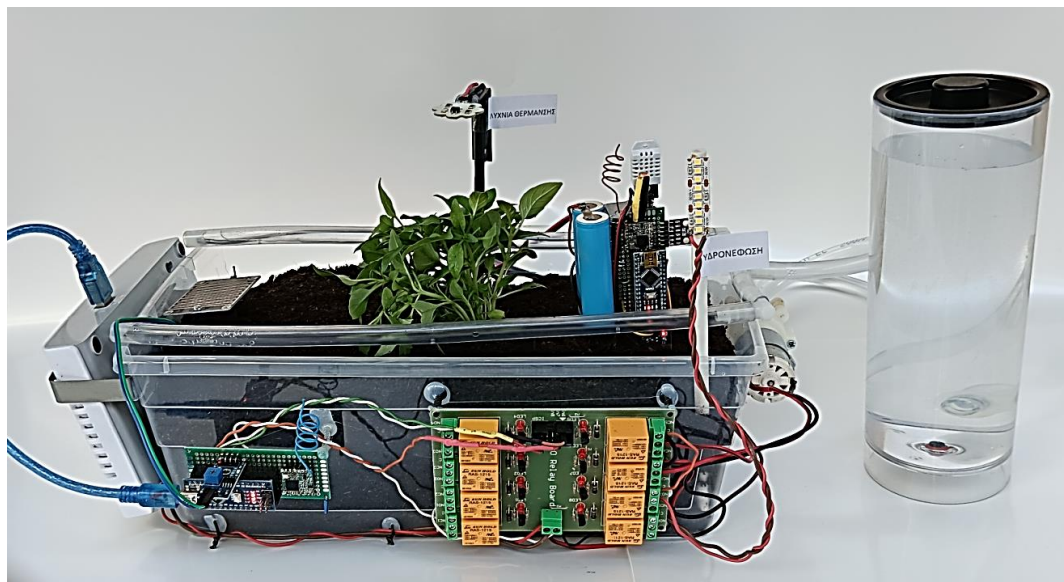


Εικόνα 3.66 – Τελική συσκευή με ενεργοποιημένο το ρελέ θερμαντικής λυχνίας.

Πλέον έχουν ολοκληρωθεί τα τρία πρώτα βήμα της κατασκευής. Μηνύματα δεδομένων στέλνονται από τις τελικές συσκευές προς το TTN και σήματα ελέγχου μπορούν να φτάνουν προς τους ενεργοποιητές του συστήματος.

Προσθέτουμε το φυτό με το χώμα και τοποθετούμε τον πολυαισθητήρα έτσι ώστε να ξεκινήσουν να λαμβάνονται πραγματικές μετρήσεις.

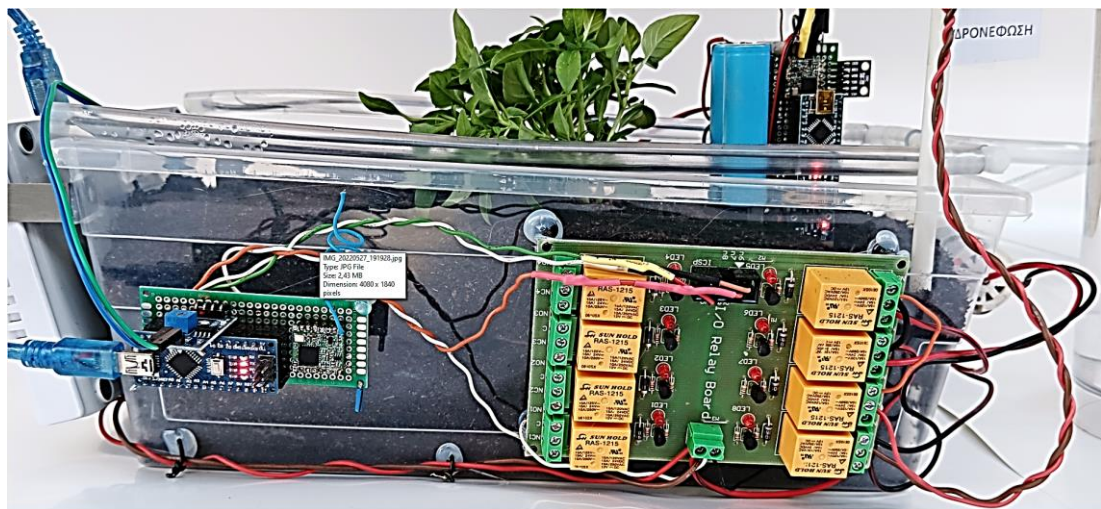
Στις ακόλουθες εικόνες 3.67(α),(β),(γ) παρουσιάζεται η συνολική κατασκευή.



Εικόνα 3.67(α) – Συνολική κατασκευή



Εικόνα 3.67(β) – Συνολική κατασκευή



Εικόνα 3.67(γ) – Συνολική κατασκευή

3.8 Αξιοποίηση και επεξεργασία των δεδομένων μέσω λογισμικού

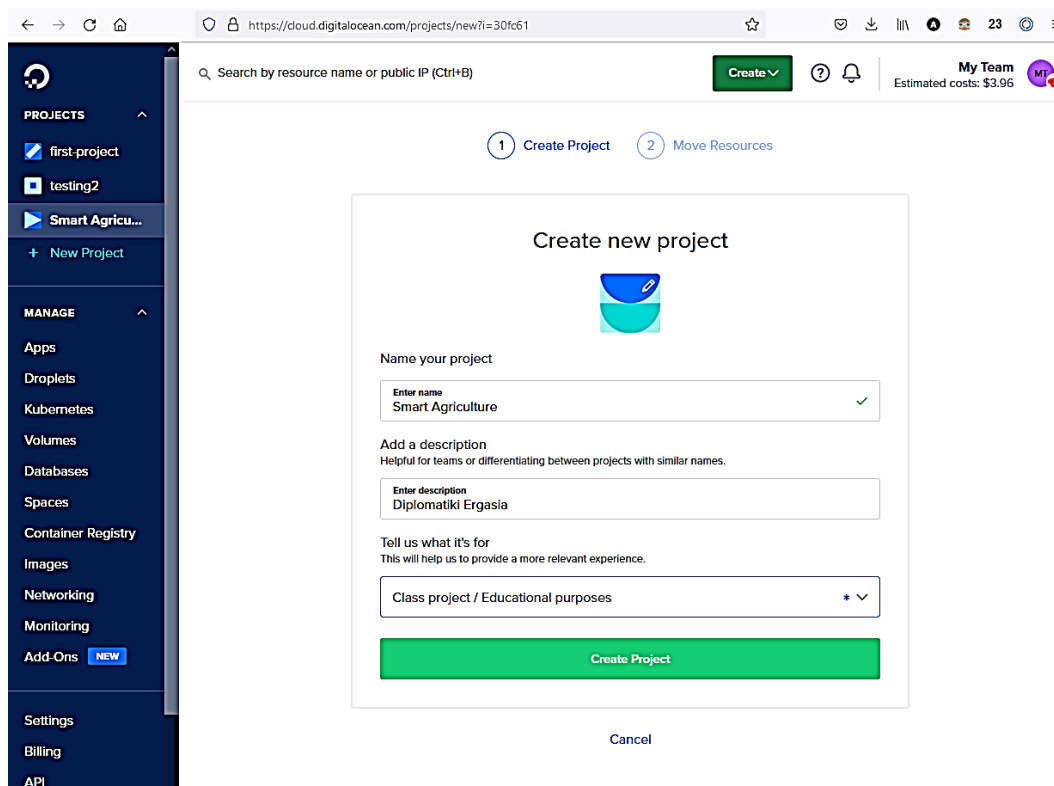
Σε αυτήν την ενότητα παρουσιάζεται όλο το backend περιβάλλον μέσω του οποίου πραγματοποιείται η παρουσίαση των δεδομένων και η ρύθμιση των απαραίτητων αυτοματισμών.

3.8.1 Δημιουργία ενός VM Ubuntu 20.04 στο Digital Ocean

Επιλέγεται η εταιρεία Digital Ocean ως χώρος φιλοξενίας για την δημιουργία μιας εικονικής μηχανής στη οποία πρόκειται να εγκατασταθούν ορισμένα προγράμματα τα οποία απαιτούνται για τα επόμενα βήματα της εργασίας.

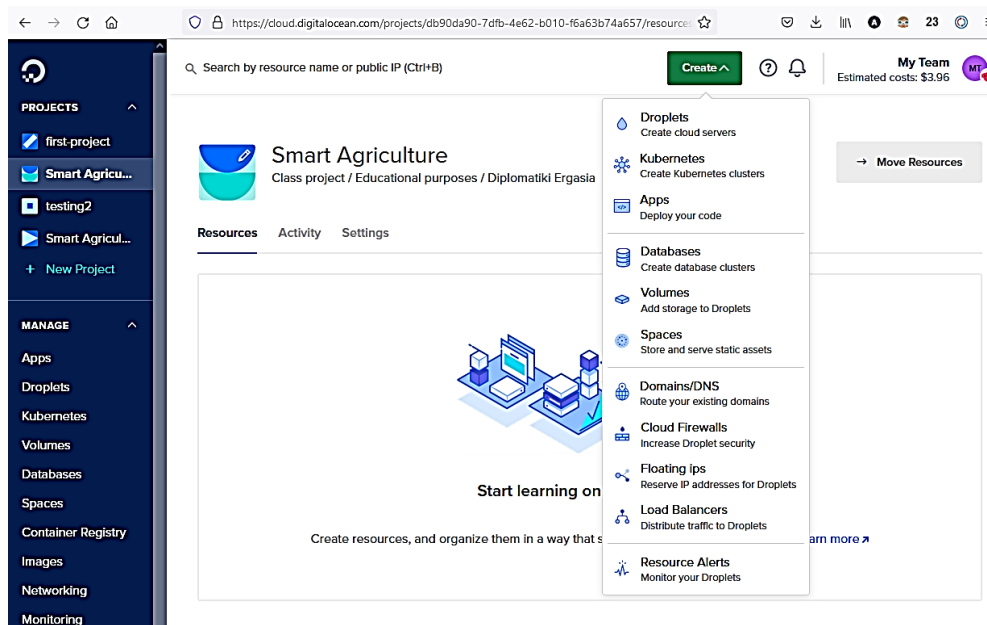
Αφού πρώτα ολοκληρωθεί η διαδικασία εισαγωγής (login) στην σελίδα Digital Ocean ακολουθεί η δημιουργία ενός νέου project μέσω της επιλογής “New project” καθώς και την συμπλήρωση των απαραίτητων πεδίων (εικόνα 3.67). Το project ορίζεται απο έναν συγκεκριμένο χώρο που μπορεί να στεγάσει μια η περισσότερες εικονικές μηχανές (virtual

servers). Μετά την συμπλήρωση των απαραίτητων πεδίων επιλέγεται 'Create project' και εμφανίζεται η οθόνη της εικόνας 3.69.



Εικόνα 3.68 – Δημιουργία ενός project στο Digital Ocean

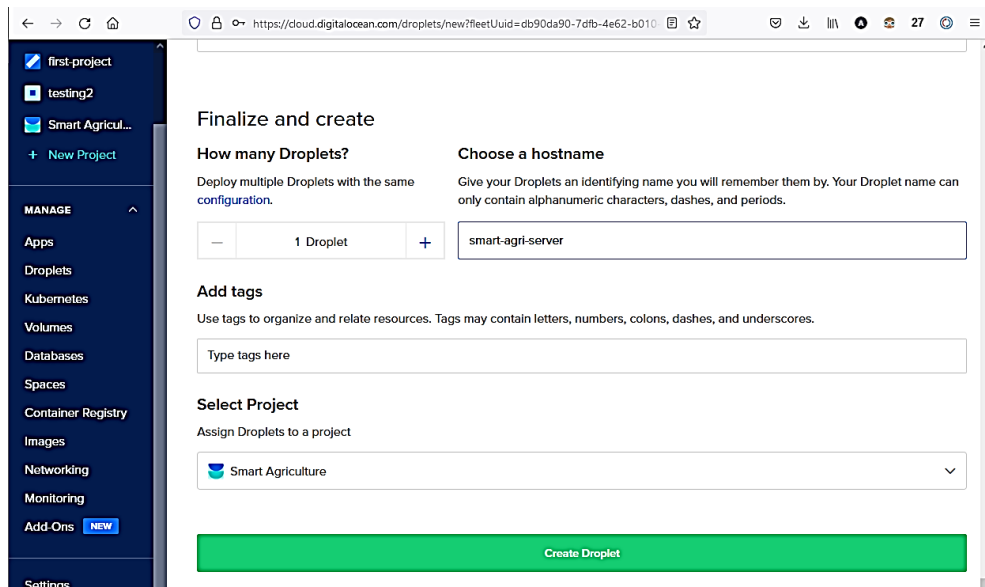
Πηγή: Digital Ocean



Εικόνα 3.69 – Δημιουργία εικονικής μηχανής στο Digital Ocean

Πηγή: Digital Ocean

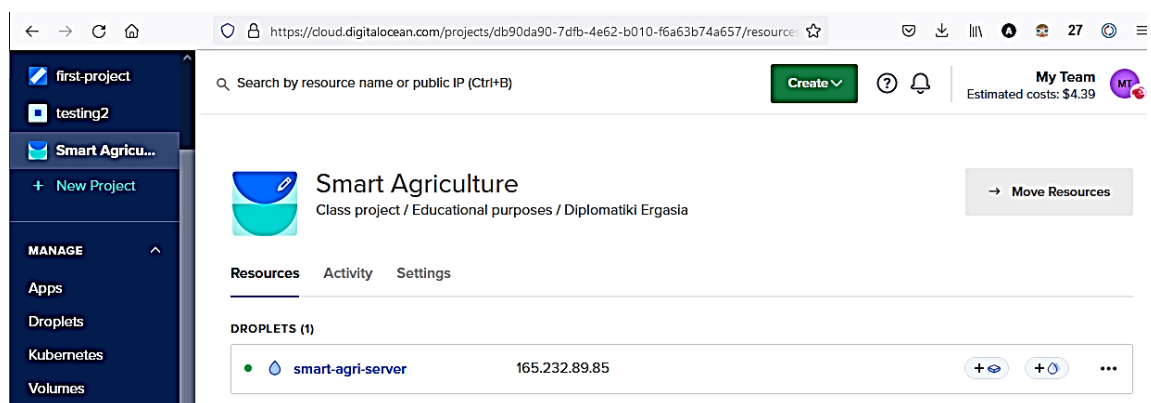
Για την δημιουργία της εικονικής μηχανής επιλέγεται “Create - Droplets” από το drop-down που φαίνεται στην εικόνα 3.69. Αρχικά ζητούνται ορισμένες τεχνικές λεπτομερείς σχετικά με τις δυνατότητες που πρέπει να έχει η εικονική μηχανή. Καθότι δεν υπάρχουν ιδιαίτερες απαιτήσεις, επιλέγονται οι ελάχιστες δυνατότητες έτσι ώστε να υπάρχει ένα χαμηλό κόστος.



Εικόνα 3.70 – Πληροφορίες για την δημιουργία μιας εικονικής μηχανής στο Digital Ocean

Πηγή: Digital Ocean

Μετά την συμπλήρωση των παραπάνω πεδίων επιλέγεται “Create Droplet” μέσω της εικόνας 3.70 και μέσα σε λίγα δευτερόλεπτα η εικονική μηχανή είναι έτοιμη προς χρήση (εικόνα 3.71).



Εικόνα 3.71 – Λίστα εικονικών μηχανών που ανήκουν στο project Smart Agriculture

Πηγή: Digital Ocean

Στην συνέχεια με την βοήθεια της διεύθυνσης ip πραγματοποιείται σύνδεση από έναν υπολογιστή που διαθέτει λειτουργικό σύστημα windows 10 μέσω ενός τερματικού

παράθυρου 'command prompt' χρησιμοποιώντας την εντολής ssh root@165.232.89.85 (εικόνα 3.72)



```
root@smart-agri-server: ~
Microsoft Windows [Version 10.0.19044.1706]
(c) Microsoft Corporation. All rights reserved.

C:\Users\apapa>ssh root@165.232.89.85
root@165.232.89.85's password:
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.4.0-107-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue May 17 11:24:30 UTC 2022

System load:  0.01          Users logged in:   0
Usage of /:   1.0% of 154.90GB IPv4 address for eth0: 165.232.89.85
Memory usage: 2%          IPv4 address for eth0: 10.18.0.7
Swap usage:   0%          IPv4 address for eth1: 10.110.0.4
Processes:   148

11 updates can be applied immediately.
7 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Last login: Tue May 17 11:23:18 2022 from 2.84.102.106
root@smart-agri-server:~#
```

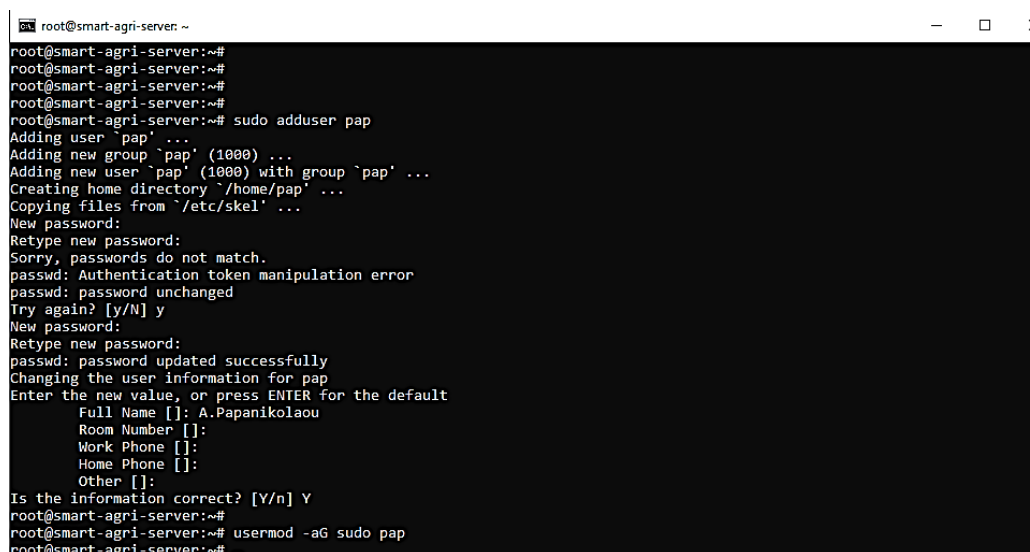
Εικόνα 3.72 – Απομακρυσμένη σύνδεση στο VM μέσω τερματικού με χρήση ssh

Αμέσως μετά την είσοδο πραγματοποιείται ενημέρωση του συστήματος μέσω των ακόλουθων εντολών :

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

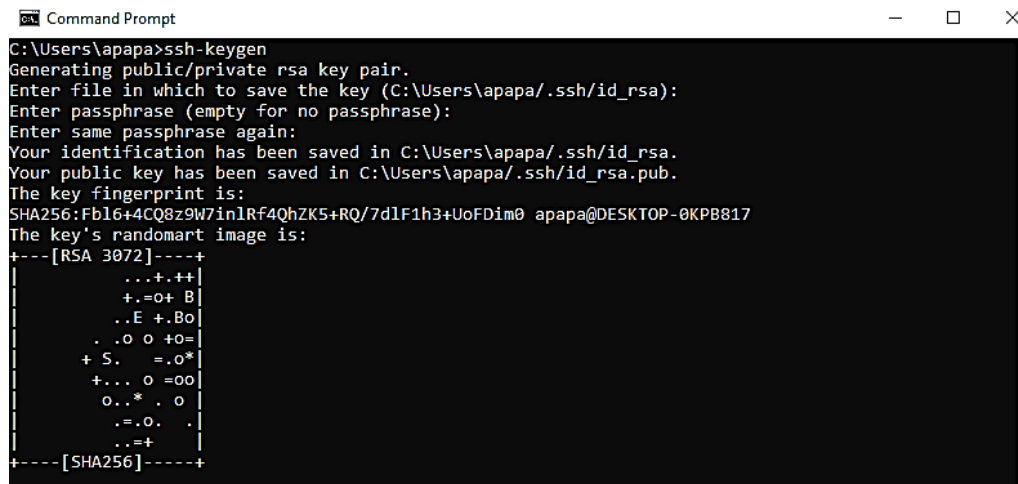
Μετά το πέρας της ενημέρωσης του συστήματος δημιουργείτε ένας νέος χρήστης με δικαιώματα sudo και κλειδιά ssh όπως φαίνεται στην εικόνα 3.73 για την ασφαλή πρόσβαση σε αυτό.



```
root@smart-agri-server:~#
root@smart-agri-server:~#
root@smart-agri-server:~#
root@smart-agri-server:~#
root@smart-agri-server:~# sudo adduser pap
Adding user `pap' ...
Adding new group `pap' (1000) ...
Adding new user `pap' (1000) with group `pap' ...
Creating home directory `/home/pap' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
Sorry, passwords do not match.
passwd: Authentication token manipulation error
passwd: password unchanged
Try again? [y/N] y
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for pap
Enter the new value, or press ENTER for the default
  Full Name []: A.Papanikolaou
   Room Number []:
   Work Phone []:
   Home Phone []:
   Other []:
Is the information correct? [Y/n] Y
root@smart-agri-server:~#
root@smart-agri-server:~# usermod -aG sudo pap
root@smart-agri-server:~#
```

Εικόνα 3.73 – Δημιουργία νέου χρήστη και προσθήκη στην ομάδα sudo

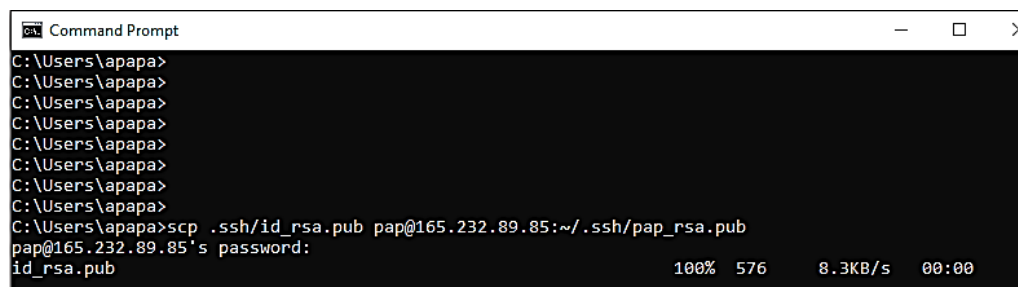
Ακολουθεί η διαδικασία δημιουργίας κλειδιών ssh που φαίνεται στην εικόνα 3.74. Πραγματοποιείται logout από την εικονική μηχανή και εκτελείται η εντολή ssh-keygen στον τοπικό υπολογιστή όπου επιχειρείτε η πρόσβαση.



```
Command Prompt
C:\Users\apapa>ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\apapa/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\apapa/.ssh/id_rsa.
Your public key has been saved in C:\Users\apapa/.ssh/id_rsa.pub.
The key's fingerprint is:
SHA256:Fb16+4CQ8z9W7inlRf4QhZK5+RQ/7d1F1h3+UoFDim0 apapa@DESKTOP-0KP817
The key's randomart image is:
+---[RSA 3072]---+
  . . . + . + +
  + . + 0 + B
  . . E + . B o
  . . O O + O =
  + S . = . O *
  + . . . O = O O
  O . * . O
  . = . O . .
  . = +
+----[SHA256]-----+
```

Εικόνα 3.74 – Δημιουργία κλειδιών ssh στο τοπικό μας μηχάνημα win10

Στην συνέχεια πραγματοποιείται μεταφορά του δημόσιου κλειδιού (public key) μέσω της εντολής scp (εικόνα 3.75) από το τοπικό περιβάλλον προς την εικονική μηχανή.



```
Command Prompt
C:\Users\apapa>
C:\Users\apapa>
C:\Users\apapa>
C:\Users\apapa>
C:\Users\apapa>
C:\Users\apapa>
C:\Users\apapa>
C:\Users\apapa>
C:\Users\apapa>
C:\Users\apapa>
C:\Users\apapa>scp .ssh/id_rsa.pub pap@165.232.89.85:~/ .ssh/pap_rsa.pub
pap@165.232.89.85's password:
id_rsa.pub          100% 576    8.3KB/s  00:00
```

Εικόνα 3.75 – Μεταφορά δημόσιου κλειδιού στην εικονική μηχανή

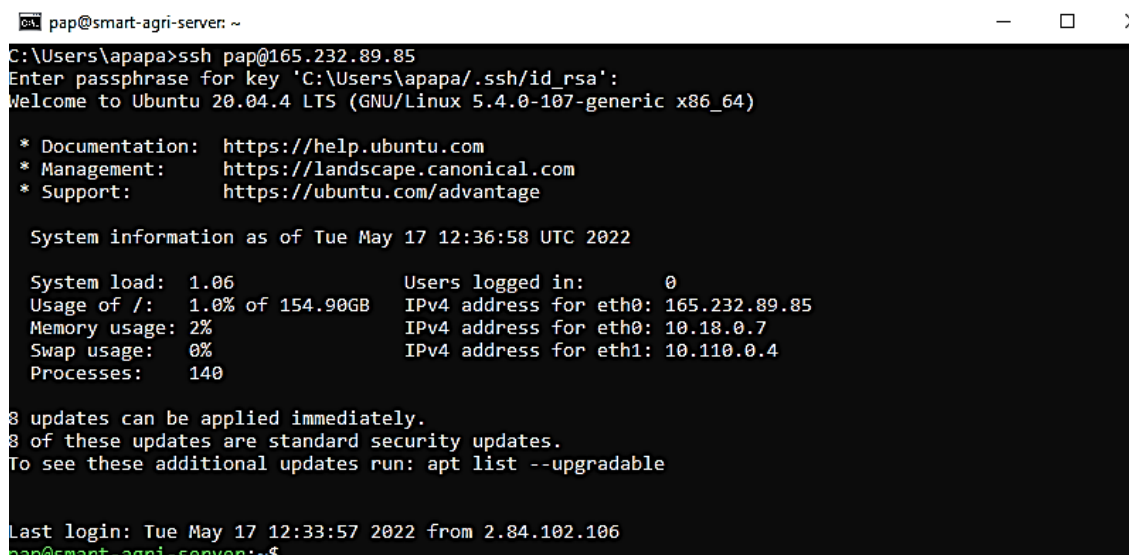
Για να είναι λειτουργικό το δημόσιο κλειδί που μεταφέρθηκε απαιτείται ενημέρωση των authorized keys της εικονικής μηχανής με το συγκεκριμένο δημόσιο κλειδί (εικόνα 3.76)



```
pap@smart-agri-server: ~
pap@smart-agri-server:~$
pap@smart-agri-server:~$
pap@smart-agri-server:~$
pap@smart-agri-server:~$
pap@smart-agri-server:~$
pap@smart-agri-server:~$
pap@smart-agri-server:~$
pap@smart-agri-server:~$ cat .ssh/pap_rsa.pub >> .ssh/authorized_keys
pap@smart-agri-server:~$
```

Εικόνα 3.76 – Ενημέρωση των authorized_keys της εικονικής μηχανής

Τέλος, απενεργοποιείται η διαδικασία εισαγωγής (login) για τον χρήστη root και πραγματοποιείται εισαγωγή στην εικονική μηχανή μέσω του χρήστη pap με την βοήθεια των ssh κλειδιών που δημιουργήθηκαν προηγουμένως (εικόνα 3.77).



```
pap@smart-agri-server: ~  
C:\Users\apapa>ssh pap@165.232.89.85  
Enter passphrase for key 'C:\Users\apapa/.ssh/id_rsa':  
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.4.0-107-generic x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:       https://ubuntu.com/advantage  
  
System information as of Tue May 17 12:36:58 UTC 2022  
  
System load:  1.06           Users logged in:      0  
Usage of /:   1.0% of 154.90GB IPv4 address for eth0: 165.232.89.85  
Memory usage: 2%           IPv4 address for eth0: 10.18.0.7  
Swap usage:  0%           IPv4 address for eth1: 10.110.0.4  
Processes:   140  
  
8 updates can be applied immediately.  
8 of these updates are standard security updates.  
To see these additional updates run: apt list --upgradable  
  
Last login: Tue May 17 12:33:57 2022 from 2.84.102.106  
pap@smart-agri-server:~$
```

Εικόνα 3.77 – Απομακρυσμένη σύνδεση στο VM με ssh και χρήστη ‘pap’

3.8.2 Εγκατάσταση του προγράμματος NodeRed και όλων των απαραίτητων modules για την δημιουργία του Web Interface

Το πρώτο πρόγραμμα που χρειάζεται να εγκατασταθεί στην εικονική μηχανή είναι το NodeRed το οποίο επιτρέπει την διαχείριση ροών δεδομένων και την δημιουργία γραφικών για την οπτικοποίηση τους.

Πριν ξεκινήσει η εγκατάσταση απαιτείται πρώτα να εγκατασταθεί το πρόγραμμα npm το οποίο είναι απαραίτητο για την μετέπειτα εγκατάσταση του NodeRed.

Ακολουθούν οι εντολές εγκατάστασης npm και NodeRed:

```
sudo apt-get install npm
```

```
sudo npm install -g --unsafe-perm node-red
```

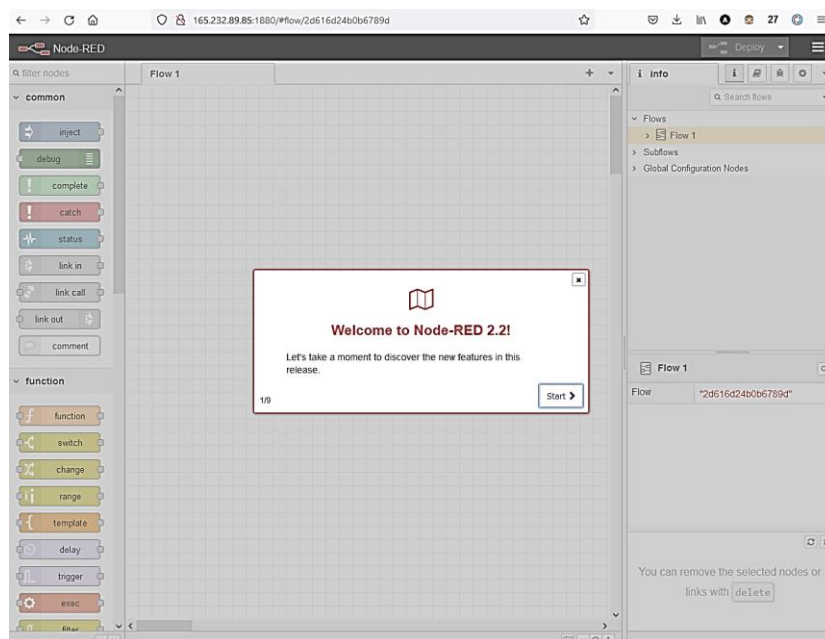
```
sudo npm install -g --unsafe-perm node-red-admin
```

Μετά την εκτέλεση των παραπάνω εντολών πραγματοποιείται εκκίνηση του NodeRed μέσω της εντολής node-red όπως φαίνεται στην εικόνα 3.78.

```
pap@smart-agri-server: ~
pap@smart-agri-server:~$
pap@smart-agri-server:~$
pap@smart-agri-server:~$ node-red
17 May 14:09:00 - [info]
Welcome to Node-RED
-----
17 May 14:09:00 - [info] Node-RED version: v2.2.2
17 May 14:09:00 - [info] Node.js version: v10.19.0
17 May 14:09:00 - [info] linux 5.4.0-107-generic x64 LE
17 May 14:09:01 - [info] Loading palette nodes
17 May 14:09:02 - [info] Settings file : /home/pap/.node-red/settings.js
17 May 14:09:02 - [info] context store : 'default' [module=memory]
17 May 14:09:02 - [info] User directory : /home/pap/.node-red
17 May 14:09:02 - [warn] Projects disabled : editorTheme.projects.enabled=false
17 May 14:09:02 - [info] Flows file : /home/pap/.node-red/flows.json
17 May 14:09:02 - [info] Creating new flow file
17 May 14:09:02 - [warn]
-----
Your flow credentials file is encrypted using a system-generated key.
If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.
You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
-----
17 May 14:09:02 - [info] Starting flows
17 May 14:09:02 - [info] Started flows
17 May 14:09:02 - [info] Server now running at http://127.0.0.1:1880/
```

Εικόνα 3.78 – Εκκίνηση του NodeRed

Για την δοκιμή της εγκατάστασης του προγράμματος NodeRed καλείται μέσω ενός φυλλομετρητή (browser) η ip της εικονικής μηχανής με πόρτα την 1880 (εικόνα 3.79).



Εικόνα 3.79 – Δοκιμή του NodeRed μέσω διεύθυνσης Ip και πόρτας 1880

Για να μην χρειάζεται κάθε φορά που ένας χρήστης συνδέεται στην εφαρμογή NodeRed να γνωρίζει απ’ έξω την διεύθυνση ip, χρησιμοποιείται μια δωρεάν υπηρεσία DNS που παρέχει δωρεάν ‘domain-name’. Η υπηρεσία είναι η DuckDNS και το όνομα που επιλέγεται είναι ‘smart-agri’ το οποίο πρέπει να συνοδεύεται από το duckdns.org.

Για την ενεργοποίηση της υπηρεσίας πραγματοποιείται είσοδος (login) στην σελίδα του duckdns (<https://www.duckdns.org/>). Στην συνέχεια καταχωρείται το όνομα που

επιλέχτηκε 'smart-agri' στο πεδίο sub-domain που φαίνεται στην εικόνα 3.80 και επιλέγεται "add domain".



Εικόνα 3.80 – Καταχώρηση domain-name στην υπηρεσία duckdns

Στην συνέχεια ενημερώνεται η διεύθυνση ip για το domain 'smart-agri' με την διεύθυνση ip της εικονικής μηχανής (165.232.89.85) που δημιουργήθηκε στο Digital Ocean και επιλέγεται 'update ip'.

Σε αυτό το σημείο η καταχώρηση έχει ολοκληρώθηκε και το NodeRed μπορεί να καλείται μέσω της ακόλουθης διεύθυνσης:

<http://smart-agri.duckdns.org:1880>

Επόμενο βήμα είναι να γίνουν ορισμένες αλλαγές έτσι ώστε να είναι ασφαλές το περιβάλλον λειτουργίας του NodeRed.

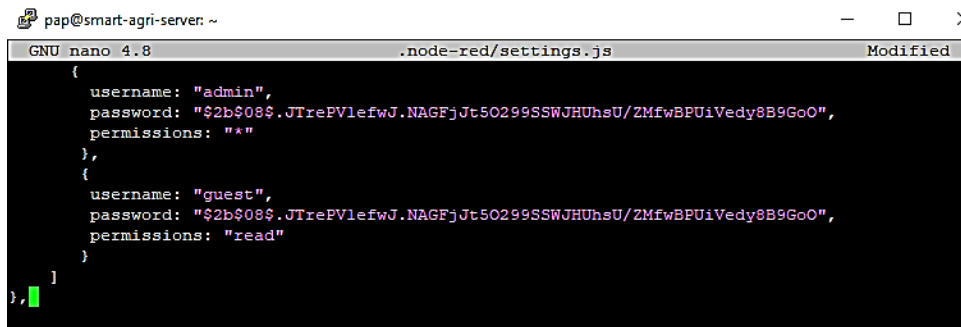
a) Προσθήκη username και password για την χρήση του NodeRed.

Δημιουργείται ένα κρυπτογραφημένο password για τον user "admin" με την εντολή **node-red-admin hash-pw** όπως φαίνεται στην εικόνα 3.81.

```
pap@smart-agri-server: ~  
pap@smart-agri-server:~$  
pap@smart-agri-server:~$  
pap@smart-agri-server:~$  
pap@smart-agri-server:~$  
pap@smart-agri-server:~$  
pap@smart-agri-server:~$  
pap@smart-agri-server:~$ node-red-admin hash-pw  
Password:  
$2b$08$/ikUQSNzbN5m0EWhEsbZm.Mh1aWRtHRTbvhdY35Idc3TdKc1AF9G  
pap@smart-agri-server:~$
```

Εικόνα 3.81 – Δημιουργία hash password για το NodeRed

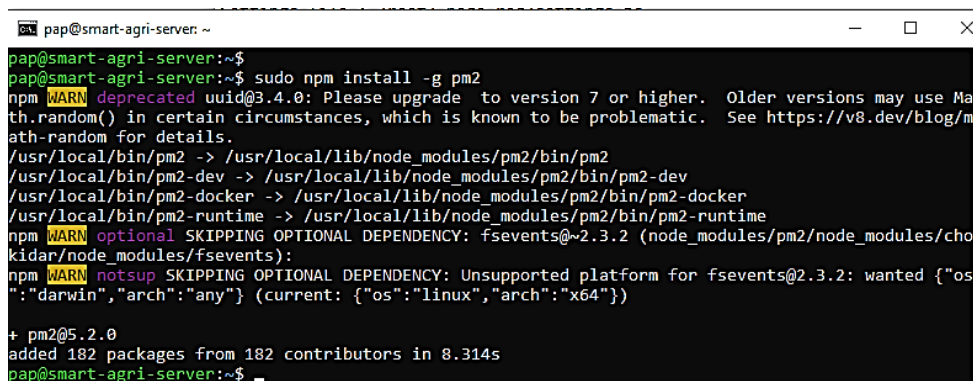
Προστίθεται στο αρχείο settings.js του NodeRed το password για τον admin που δημιουργήθηκε προηγούμενος μέσω της εντολής **sudo nano .node-red/settings.js** όπως φαίνεται στην εικόνα 3.82.



```
GNU nano 4.8 .node-red/settings.js Modified
{
  username: "admin",
  password: "$2b$08$.JTrePV1efwJ.NAGFjJt50299SSWJHUsU/ZMfwBPUiVedy8B9GoO",
  permissions: "*"
},
{
  username: "guest",
  password: "$2b$08$.JTrePV1efwJ.NAGFjJt50299SSWJHUsU/ZMfwBPUiVedy8B9GoO",
  permissions: "read"
}
}, ]
```

Εικόνα 3.82 – Ενεργοποίηση username & password στο NodeRed

b) Ρύθμιση εκκίνησης του NodeRed με κάθε επανεκκίνηση του συστήματος. Εγκαθίσταται το πρόγραμμα pm2 όπου αναλαμβάνει να εκκινεί το node red κάθε φορά που θα επανεκκινείτε η εικονική μηχανή. Για την εγκατάσταση χρησιμοποιείται η εντολή `sudo npm install -g pm2` όπως φαίνεται στην εικόνα 3.83.



```
pap@smart-agri-server: ~
pap@smart-agri-server:~$ sudo npm install -g pm2
npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.
/usr/local/bin/pm2 -> /usr/local/lib/node_modules/pm2/bin/pm2
/usr/local/bin/pm2-dev -> /usr/local/lib/node_modules/pm2/bin/pm2-dev
/usr/local/bin/pm2-docker -> /usr/local/lib/node_modules/pm2/bin/pm2-docker
/usr/local/bin/pm2-runtime -> /usr/local/lib/node_modules/pm2/bin/pm2-runtime
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@~2.3.2 (node_modules/pm2/node_modules/chokidar/node_modules/fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"linux","arch":"x64"})
+ pm2@5.2.0
added 182 packages from 182 contributors in 8.314s
pap@smart-agri-server:~$
```

Εικόνα 3.83 – Εγκατάσταση του pm2 στον Virtual Server

Με την ολοκλήρωση της εγκατάστασης του pm2 εκτελούνται οι ακόλουθες εντολές που δηλώνουν στο pm2 ότι πρέπει να γίνεται εκκίνηση του NodeRed για κάθε επανεκκίνηση της εικονικής μηχανής.

```
pm2 start /usr/local/bin/node-red -- -v
pm2 save
pm2 startup
pm2 startup systemd
```

Η εκτέλεση των παραπάνω εντολών φαίνεται στην ακόλουθη εικόνα 3.84 .

```
pap@smart-agri-server: ~  
17 May 16:00:18 - [info] Server now running at http://127.0.0.1:1880/  
17 May 16:00:18 - [info] Starting flows  
17 May 16:00:18 - [info] Started flows  
17 May 16:06:27 - [info] Stopping flows  
17 May 16:06:27 - [info] Stopped flows  
pap@smart-agri-server:~$  
pap@smart-agri-server:~$  
pap@smart-agri-server:~$  
pap@smart-agri-server:~$  
pap@smart-agri-server:~$  
pap@smart-agri-server:~$ which node-red  
/usr/local/bin/node-red  
pap@smart-agri-server:~$ pm2 start /usr/local/bin/node-red -- -v  
[PM2] Spawning PM2 daemon with pm2_home=/home/pap/.pm2  
[PM2] PM2 Successfully daemonized  
[PM2] Starting /usr/local/bin/node-red in fork_mode (1 instance)  
[PM2] Done.  


| id | name     | mode | ▢ | status | cpu | memory |
|----|----------|------|---|--------|-----|--------|
| 0  | node-red | fork | ▢ | online | 0%  | 20.6mb |

  
pap@smart-agri-server:~$ pm2 save  
[PM2] Saving current process list...  
[PM2] Successfully saved in /home/pap/.pm2/dump.pm2  
pap@smart-agri-server:~$ pm2 startup  
[PM2] Init System found: systemd  
[PM2] To setup the Startup Script, copy/paste the following command:  
sudo env PATH=$PATH:/usr/bin /usr/local/lib/node_modules/pm2/bin/pm2 startup systemd -u pap --hp /home/pap  
pap@smart-agri-server:~$ pm2 startup systemd  
[PM2] Init System found: systemd  
[PM2] To setup the Startup Script, copy/paste the following command:  
sudo env PATH=$PATH:/usr/bin /usr/local/lib/node_modules/pm2/bin/pm2 startup systemd -u pap --hp /home/pap  
pap@smart-agri-server:~$
```

Εικόνα 3.84 – Ρύθμιση εκκίνησης του NodeRed μέσω pm2

Τέλος, μέσω επανεκκίνησης του συστήματος επιβεβαιώνεται ότι η διαδικασία επανεκκίνησης του NodeRed λειτουργεί κανονικά.

reboot system

c) Ενημερώνεται το firewall του λειτουργικού συστήματος να επιτρέπει την χρήση της πόρτας 1880 που χρησιμοποιεί το NodeRed μέσω της ακόλουθης εντολής.

sudo ufw allow 1880

```
pap@smart-agri-server: ~  
pap@smart-agri-server:~$  
pap@smart-agri-server:~$  
pap@smart-agri-server:~$ sudo ufw allow 1880  
Rules updated  
Rules updated (v6)  
pap@smart-agri-server:~$
```

Εικόνα 3.85 – Ενημέρωση του firewall για την πόρτα 1880

Σε αυτό το σημείο η εγκατάσταση του NodeRed έχει ολοκληρωθεί και είναι έτοιμο για την δημιουργία και τη διαχείριση ροών δεδομένων.

3.8.3 Εγκατάσταση της βάσης InfluxDB και σύνδεση με το The things Network

Η InfluxDB πρόκειται για λογισμικό ανοιχτού κώδικα. Αποτελεί μια βάση δεδομένων χρονοσειρών και επιτρέπει την αποθήκευση και ανάκτηση δεδομένων σε πεδία όπως η παρακολούθηση λειτουργιών, οι μετρήσεις εφαρμογών, τα δεδομένα αισθητήρων του διαδικτύου των πραγμάτων και η ανάλυση σε πραγματικό χρόνο.

Στην δική μας εφαρμογή παρέχει την δυνατότητα της αποθήκευσης των δεδομένων που στέλνονται από τους αισθητήρες με χρονική σειρά (Ημερομηνία, Ώρα) έτσι ώστε να μπορούν να αξιοποιηθούν για συγκριτικά γραφικά ή ακόμη και αν χρειαστεί να αναζητηθούν μετρήσεις του παρελθόντος.

Για την εγκατάσταση της InfluxDB εκτελούνται οι ακόλουθες εντολές από ένα τερματικό παράθυρο της εικονικής μηχανής.

Αρχικά με τις δύο επόμενες εντολές μεταφέρονται τα απαραίτητα αρχεία για την εγκατάσταση στην εικονική μηχανή.

```
sudo curl -sL https://repos.influxdata.com/influxdb.key | sudo apt-key add -  
sudo echo "deb https://repos.influxdata.com/ubuntu bionic stable" | sudo tee  
/etc/apt/sources.list.d/influxdb.list
```

Ελέγχεται το σύστημα για τυχόν ενημερώσεις

```
sudo apt update
```

Εγκαθίσταται το λογισμικό για την InfluxDB

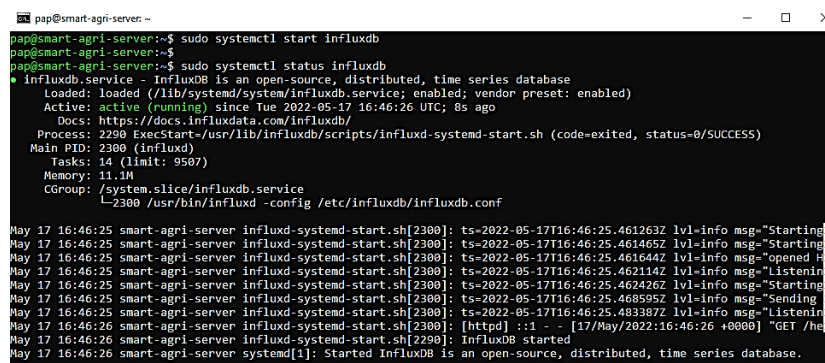
```
sudo apt install influxdb
```

Εκκινείτε η influxdb

```
sudo systemctl start influxdb
```

Ελέγχεται αν η Influxdb είναι έτοιμη για λειτουργία (εικόνα 3.86)

```
sudo systemctl status influxdb
```



```
pap@smart-agri-server:~$ sudo systemctl start influxdb  
pap@smart-agri-server:~$ sudo systemctl status influxdb  
● influxdb.service - InfluxDB is an open-source, distributed, time series database  
   Loaded: loaded (/lib/systemd/system/influxdb.service; enabled; vendor preset: enabled)  
   Active: active (running) since Tue 2022-05-17 16:46:26 UTC; 8s ago  
     Docs: https://docs.influxdata.com/influxdb/  
   Process: 2290 ExecStart=/usr/lib/influxdb/scripts/influxd-systemd-start.sh (code=exited, status=0/SUCCESS)  
   Main PID: 2300 (influxd)  
     Tasks: 14 (limit: 9507)  
    Memory: 11.1M  
   CGroup: /system.slice/influxdb.service  
           └─2300 /usr/bin/influxd -config /etc/influxdb/influxdb.conf  
May 17 16:46:25 smart-agri-server influxd-systemd-start.sh[2300]: ts=2022-05-17T16:46:25.461263Z lvl=info msg="Starting InfluxDB" data=map[hostname:smart-agri-server ip:172.17.0.2] source=influxd  
May 17 16:46:25 smart-agri-server influxd-systemd-start.sh[2300]: ts=2022-05-17T16:46:25.461465Z lvl=info msg="Starting InfluxDB" data=map[hostname:smart-agri-server ip:172.17.0.2] source=influxd  
May 17 16:46:25 smart-agri-server influxd-systemd-start.sh[2300]: ts=2022-05-17T16:46:25.461644Z lvl=info msg="opened HTTP listener" data=map[hostname:smart-agri-server ip:172.17.0.2] source=influxd  
May 17 16:46:25 smart-agri-server influxd-systemd-start.sh[2300]: ts=2022-05-17T16:46:25.462114Z lvl=info msg="Listening for connections" data=map[hostname:smart-agri-server ip:172.17.0.2] source=influxd  
May 17 16:46:25 smart-agri-server influxd-systemd-start.sh[2300]: ts=2022-05-17T16:46:25.462426Z lvl=info msg="Starting InfluxDB" data=map[hostname:smart-agri-server ip:172.17.0.2] source=influxd  
May 17 16:46:25 smart-agri-server influxd-systemd-start.sh[2300]: ts=2022-05-17T16:46:25.468895Z lvl=info msg="Sending heartbeat" data=map[hostname:smart-agri-server ip:172.17.0.2] source=influxd  
May 17 16:46:25 smart-agri-server influxd-systemd-start.sh[2300]: ts=2022-05-17T16:46:25.483387Z lvl=info msg="Listening for connections" data=map[hostname:smart-agri-server ip:172.17.0.2] source=influxd  
May 17 16:46:26 smart-agri-server influxd-systemd-start.sh[2300]: [httpd] :1 - - [17/May/2022:16:46:26 +0000] "GET /health/v1/status/ready HTTP/1.1" 200 100  
May 17 16:46:26 smart-agri-server influxd-systemd-start.sh[2290]: InfluxDB started  
May 17 16:46:26 smart-agri-server systemd[1]: Started InfluxDB is an open-source, distributed, time series database.
```

Εικόνα 3.86 – Έλεγχος κατάστασης της Influx DB

Από την εικόνα 3.86 φαίνεται ότι η εγκατάσταση έχει ολοκληρωθεί και είναι έτοιμη προς χρήση.

Υπολείπονται οι ακόλουθες ρυθμίσεις:

A) Δημιουργία ενός χρήστη 'admin' με password

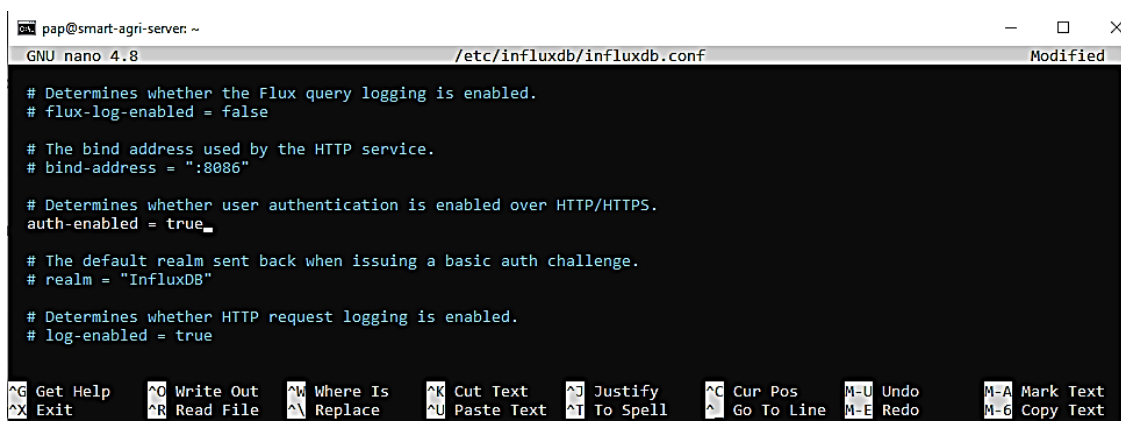
Αφού πρώτα πραγματοποιηθεί σύνδεση με την βάση μέσω της εντολής 'Influx' δημιουργείται ένας χρήστης 'admin' με ένα password της επιλογής μας όπως φαίνεται στην ακόλουθη εικόνα 3.87.



```
pap@smart-agri-server: ~
pap@smart-agri-server:~$
pap@smart-agri-server:~$ influx
Connected to http://localhost:8086 version 1.8.10
InfluxDB shell version: 1.8.10
> CREATE USER admin WITH PASSWORD '...' WITH ALL PRIVILEGES
> quit_
```

Εικόνα 3.87 – Δημιουργία χρήστη στην βάση InfluxDB

B) Στην συνέχεια πρέπει να επιτραπεί το authenticate στην βάση μέσω πρωτοκόλλου http και https, μέσω ενημέρωσης του αρχείου `influxdb.conf` που βρίσκεται στην διαδρομή `/etc/influxdb/influxdb.conf`, με την βοήθεια ενός text editor όπως φαίνεται στην εικόνα 3.88.



```
GNU nano 4.8 /etc/influxdb/influxdb.conf Modified
# Determines whether the Flux query logging is enabled.
# flux-log-enabled = false

# The bind address used by the HTTP service.
# bind-address = ":8086"

# Determines whether user authentication is enabled over HTTP/HTTPS.
auth-enabled = true_
# realm = "InfluxDB"

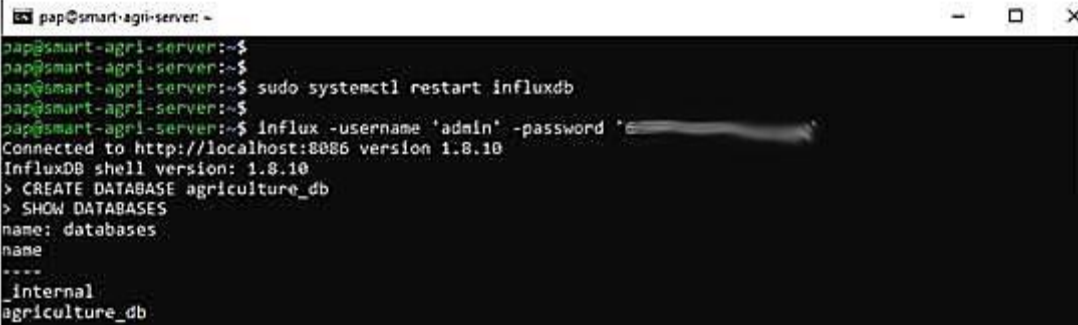
# Determines whether HTTP request logging is enabled.
# log-enabled = true

^G Get Help  ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos    M-U Undo     M-A Mark Text
^X Exit      ^R Read File  ^L Replace   ^U Paste Text ^I To Spell   ^_ Go To Line  M-E Redo     M-G Copy Text
```

Εικόνα 3.88 – Enable authenticate στην βάση

Γ) Τέλος, δημιουργείται μια βάση δεδομένων με όνομα "agriculture_db" για να αποθηκεύονται τα δεδομένα των αισθητήρων μέσω των ακόλουθων βημάτων τα οποία παρουσιάζονται και στην εικόνα 3.89:

- Πραγματοποιούμε εισαγωγή στην βάση μέσω της εντολής `influx -username 'admin' -password 'db_pass'`
- Δημιουργείται μια νέα βάση με την εντολή `CREATE DATABASE agriculture_db`.
- Ελέγχεται με την `SHOW DATABASES` αν πράγματι έχει δημιουργηθεί η βάση.



```
pap@smart-agri-server:~$
pap@smart-agri-server:~$
pap@smart-agri-server:~$ sudo systemctl restart influxdb
pap@smart-agri-server:~$
pap@smart-agri-server:~$ influx -username 'admin' -password '
Connected to http://localhost:8086 version 1.8.10
InfluxDB shell version: 1.8.10
> CREATE DATABASE agriculture_db
> SHOW DATABASES
name: databases
----
      _internal
      agriculture_db
```

Εικόνα 3.89 – Login στην InfluxDB και δημιουργία βάσης agriculture_db

3.8.4 Εγκατάσταση του λογισμικού Grafana για την προσθήκη γραφικών παραστάσεων

Το Grafana είναι μια εφαρμογή ανοιχτού κώδικα που επιτρέπει την οπτικοποίηση των δεδομένων μας με τη βοήθεια εντυπωσιακών γραφικών εργαλείων. Σημαντικό χαρακτηριστικό είναι ότι μας δίνει την δυνατότητα να το συνδέσουμε με οποιαδήποτε πηγή-βάση δεδομένων όπως Graphite, Prometheus, Influx DB, Elasticsearch, MySQL, PostgreSQL κτλ.

Για την εγκατάσταση του λογισμικού Grafana χρησιμοποιούνται οι ακόλουθες εντολές:

Μεταφέρονται τα απαραίτητα αρχεία για την εγκατάσταση στην εικονική μηχανή.

```
wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -
sudo add-apt-repository "deb https://packages.grafana.com/oss/deb stable main"
```

Ελέγχεται το σύστημα για τυχόν ενημερώσεις

```
sudo apt update
```

Πραγματοποιείται εγκατάσταση του grafana

```
sudo apt install grafana
```

Εκκίνηση του Grafana

```
sudo systemctl start grafana-server
```

Ελέγχεται αν το Grafana είναι έτοιμο για χρήση (εικόνα 3.90)

```
sudo systemctl status grafana-server
```

```
pap@smart-agri-server: ~  
● grafana-server.service - Grafana instance  
   Loaded: loaded (/lib/systemd/system/grafana-server.service; disabled; vendor preset: enabled)  
   Active: active (running) since Tue 2022-05-17 17:26:52 UTC; 7s ago  
     Docs: http://docs.grafana.org  
   Main PID: 3860 (grafana-server)  
     Tasks: 17 (limit: 9507)  
    Memory: 34.5M  
    CGroup: /system.slice/grafana-server.service  
           └─3860 /usr/sbin/grafana-server --config=/etc/grafana/grafana.ini --pidfile=/var/run/grafana-server.pid  
  
May 17 17:26:58 smart-agri-server grafana-server[3860]: logger=sqlstore t=2022-05-17T17:26:58.000Z  
May 17 17:26:58 smart-agri-server grafana-server[3860]: logger=plugin.manager t=2022-05-17T17:26:58.000Z  
May 17 17:26:58 smart-agri-server grafana-server[3860]: logger=plugin.finder t=2022-05-17T17:26:58.000Z  
May 17 17:26:58 smart-agri-server grafana-server[3860]: logger=query_data t=2022-05-17T17:26:58.000Z  
May 17 17:26:58 smart-agri-server grafana-server[3860]: logger=live.push_http t=2022-05-17T17:26:58.000Z  
May 17 17:26:58 smart-agri-server grafana-server[3860]: logger=server t=2022-05-17T17:26:58.000Z  
May 17 17:26:58 smart-agri-server grafana-server[3860]: logger=ngalert t=2022-05-17T17:26:58.000Z  
May 17 17:26:58 smart-agri-server grafana-server[3860]: logger=grafanaStorageLogger t=2022-05-17T17:26:58.000Z  
May 17 17:26:58 smart-agri-server grafana-server[3860]: logger=http.server t=2022-05-17T17:26:58.000Z  
May 17 17:26:58 smart-agri-server grafana-server[3860]: logger=ngalert.multiorg.alertmanager t=2022-05-17T17:26:58.000Z  
~  
lines 1-20/20 (END)
```

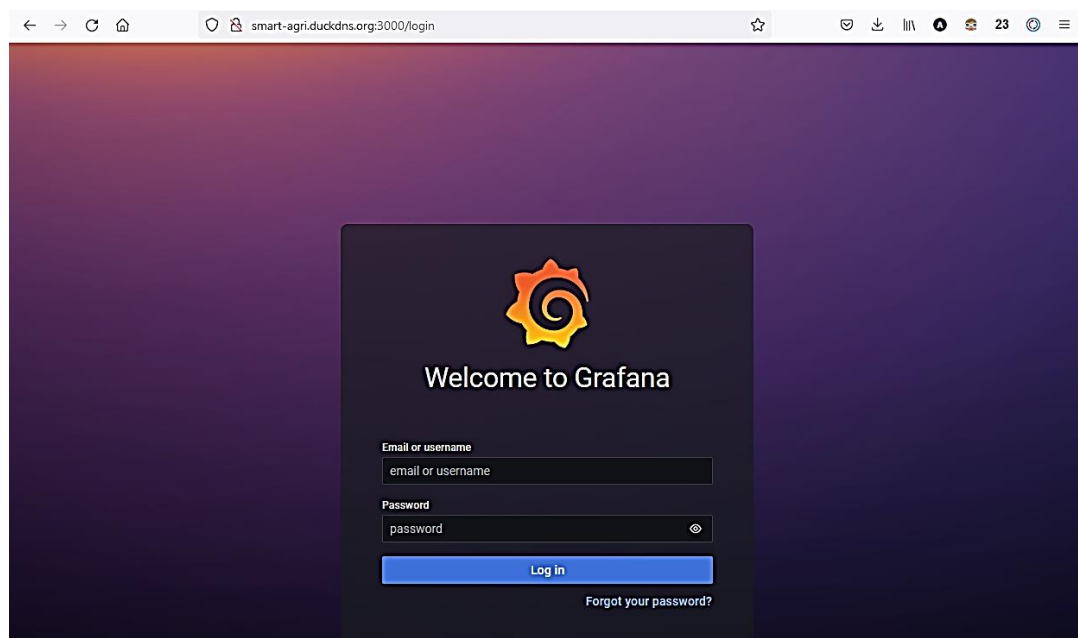
Εικόνα 3.90 – Έλεγχος κατάστασης του Grafana

Ενεργοποιείται η αυτόματη εκκίνηση του Grafana μετά από κάθε επανεκκίνηση του συστήματος.

```
sudo systemctl enable grafana-server
```

Η εγκατάσταση του Grafana έχει ολοκληρωθεί και μπορεί να γίνει η κλήση του μέσω ενός φυλλομετρητή (browser) χρησιμοποιώντας την ακόλουθη διεύθυνση :

<http://smart-agri.duckdns.org:3000> όπως φαίνεται στην εικόνα 3.91.



Εικόνα 3.91 – Εκκίνηση του grafana από browser

Την πρώτη φορά που πραγματοποιείται εισαγωγή (login) γίνεται με τα στοιχεία username = 'admin' και password = 'admin'. Αμέσως μετά, προτείνεται από το grafana η αλλαγή του password για τον χρήστη 'admin'.

Για να υπάρχει συνεργασία με το NodeRed, έτσι ώστε να μπορεί να πραγματοποιηθεί η ενσωμάτωση γραφικών μεταξύ τους πρέπει να γίνουν οι ακόλουθες αλλαγές στο αρχείο ρυθμίσεων που βρίσκεται στην διαδρομή `/etc/grafana/grafana.ini`

```
allow_embedding = true
```

```
[auth.anonymous]
```

```
enabled = true
```

```
org_name = <<org name>>
```

```
org_role = Viewer
```

Αυτό πραγματοποιείται με την βοήθεια της εντολής

```
sudo nano /etc/grafana/grafana.ini
```

Αφού ολοκληρωθούν οι αλλαγές γίνεται επανεκκίνηση του Grafana

```
sudo systemctl restart grafana-server
```

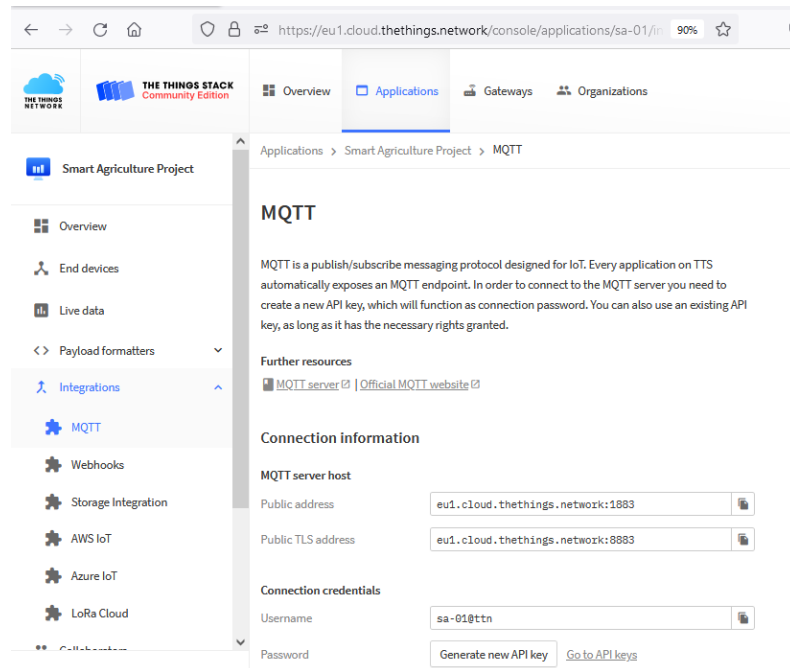
3.8.5 Δημιουργία ενός Web interface

Υπάρχουν πλέον όλα τα απαραίτητα εργαλεία που απαιτούνται για να γίνει η παρουσίαση των δεδομένων των αισθητήρων αλλά επίσης και η αυτοματοποίηση ορισμένων λειτουργιών.

Η πρώτη κίνηση αφορά την υλοποίηση μιας γέφυρας για την μεταφορά των δεδομένων από το TTN προς το NodeRed. Αυτό σημαίνει ότι κάθε φορά που φτάνουν νέα δεδομένα στο TTN από τους αισθητήρες που βρίσκονται στις τελικές συσκευές, πρόκειται να μεταφέρονται αυτόματα και στο περιβάλλον του NodeRed. Αυτή η γέφυρα μεταφοράς υλοποιείται με την βοήθεια του πρωτοκόλλου mqtt.

Για να μπορεί να λειτουργήσει το mqtt ανάμεσα στο TTN και στο NodeRed απαιτούνται δύο πράγματα. Από την πλευρά του TTN πρέπει να δημιουργηθούν οι απαραίτητοι κωδικοί διεπαφής API και από την πλευρά του NodeRed πρέπει να ρυθμιστεί ένας κόμβος έτσι ώστε να χρησιμοποιεί αυτήν την διεπαφή για να μπορεί να αντλεί δεδομένα.

Αρχικά δημιουργούνται τα απαιτούμενα κλειδιά διεπαφής API στο TTN. Από το βασικό menu της εφαρμογής μας στο TTN και επιλέγεται Integrations→MQTT και μας εμφανίζεται η ακόλουθη οθόνη που φαίνεται στην εικόνα 3.92.



Εικόνα 3.92 – Δημιουργία σύνδεσης μέσω πρωτοκόλλου mqtt στο TTN
(Πηγή: The Things Network)

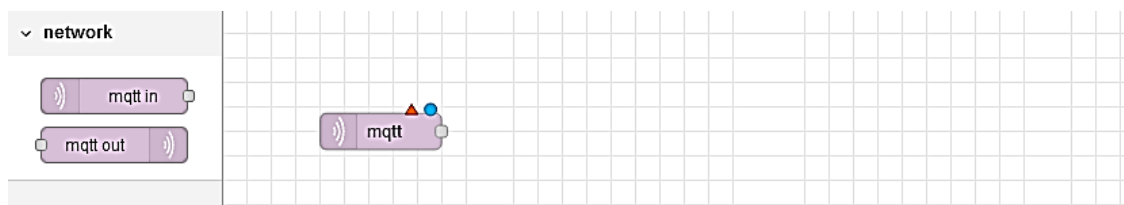
Η παραπάνω οθόνη μας ενημερώνει για την διεύθυνση του διακομιστή που αφορά το mqtt πρωτόκολλο η οποία πρόκειται να καταχωρηθεί αργότερα στο NodeRed. Από την πλευρά μας απαιτείται η συμπλήρωση των στοιχείων εισόδου για το mqtt δηλαδή, το username και το password.

Το username προτείνεται από το σύστημα και αποτελείται από το ID της εφαρμογής ακολουθούμενο από @ttn ενώ το password δημιουργείται από μία γεννήτρια κλειδιών API πατώντας επάνω στην επιλογή “Generate new API key”.

Αφού γίνει επιλογή του “Generate new API key” σημειώνεται το κλειδί που θα δημιουργηθεί καθότι πρόκειται να ζητηθεί αργότερα από το NodeRed.

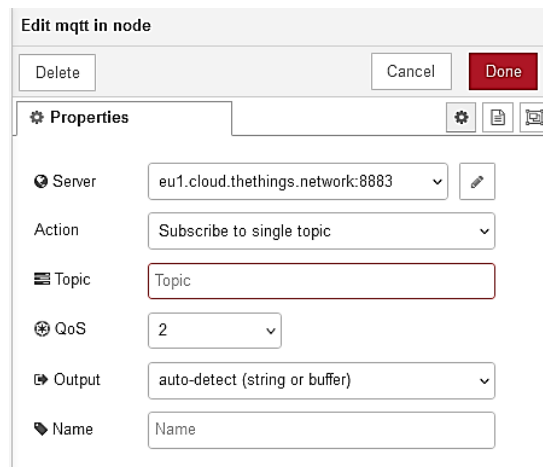
Στην συνέχεια επιλέγεται η διεύθυνση <http://smart-agri.duckdns.org:1880/> η οποία μας μεταφέρει στο περιβάλλον του NodeRed και πραγματοποιείται η διαδικασία εισαγωγής (login).

Για να μπορούν να λαμβάνονται δεδομένα μέσω MQTT στο NodeRed προστίθεται ένας κόμβος mqtt-in στην ροή (flow) που εργαζόμαστε μέσω διαδικασίας drag and drop.



Εικόνα 3.93 – Εισαγωγή κόμβου mqtt-in

Με διπλό κλικ επάνω του μας ανοίγει ένα παράθυρο “edit mqtt node” το οποίο φαίνεται στην εικόνα 3.94. Μέσω αυτού του παραθύρου μπορούν να γίνουν οι απαραίτητες ρυθμίσεις σύνδεσης.



Εικόνα 3.94 – Ιδιότητες κόμβου mqtt-in

Στο πεδίο Server (πατώντας επάνω στο μικρό μολυβάκι) συμπληρώνονται τα στοιχεία της διεπαφής API που δημιουργήθηκαν προηγούμενος στο TTN. (Username , API key)

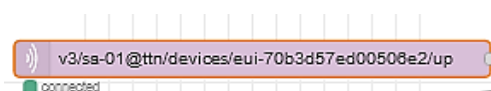
Στο πεδίο Topic δηλώνεται από ποιον αισθητήρα λαμβάνονται δεδομένα. Στην συγκεκριμένη περίπτωση μας ενδιαφέρει ο πολυαισθητήρας όπου έχει όνομα συσκευής στο TTN ‘eui-70b3d57ed00506e2’ και ανήκει στην εφαρμογή με id ‘sa-01’.

Το format που δέχεται το πεδίο topic είναι το ακόλουθο:

TTV Version	Application+ “@ttn”	“devices”	Device ID	up
Το version του TTN που χρησιμοποιείται	Από ποιο application αντλούνται δεδομένα	Συμπληρώνεται “devices” Όταν πρόκειται για τελική συσκευή	Id τελικής συσκευής	‘up’ για τα ανερχόμενα μηνύματα

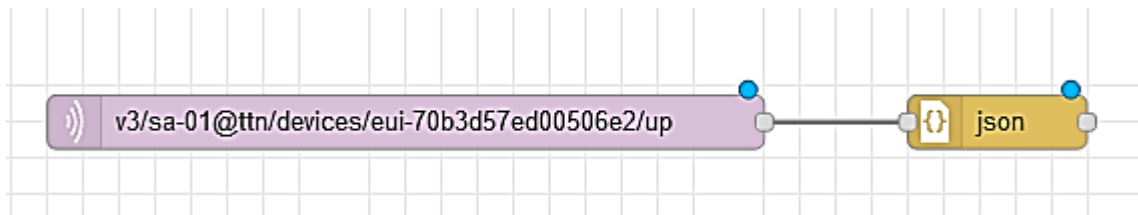
Άρα, στο πεδίο topic της εικόνας 3.94 συμπληρώνεται το ακόλουθο περιεχόμενο:
v3/sa-01@ttn/devices/eui-70b3d57ed00506e2/up

Ενημερώνεται ο κόμβος με το κουμπί “Done” και γίνεται Deploy το NodeRed έτσι ώστε να ενεργοποιηθούν οι αλλαγές. Παρατηρείται ότι κάτω από τον mqtt κόμβο εμφανίζει ‘connected’ που σημαίνει ότι έχει συνδεθεί στο TTN και είναι έτοιμο να λάβει δεδομένα μετρήσεων από αυτό.



Εικόνα 3.95 – Κόμβος mqtt-in συνδεδεμένος στο TTN

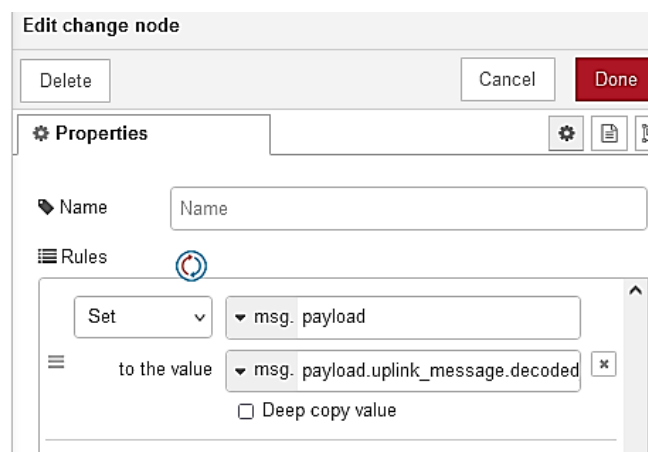
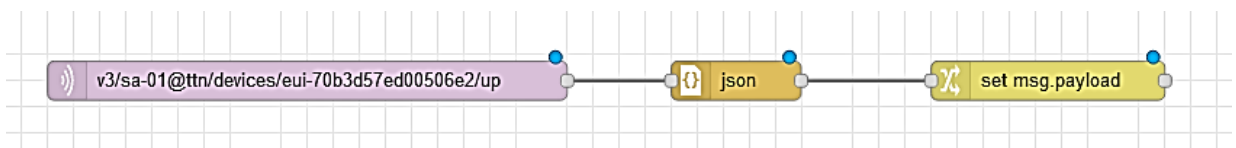
Το mqtt-in μεταφέρει τα δεδομένα των αισθητήρων στο περιβάλλον του NodeRed υπό μορφή json. Για την μετατροπή τους σε αντικείμενα (object) προστίθεται και συνδέεται ένας κόμβος json όπως φαίνεται στην ακόλουθη εικόνα 3.96.



Εικόνα 3.96 – Προσθήκη κόμβου json

Επειδή εντός της ροής json εκτός από τα δεδομένα των αισθητήρων περιλαμβάνονται αρκετές άλλες πληροφορίες που δεν είναι απαραίτητες όπως Π.χ το id του gateway που εισήλθαν τα δεδομένα στο TTN, ισχύς σήματος, session κλειδιά κτλ, προστίθεται ένας νέος κόμβος “change node” έτσι ώστε να παραμείνουν στην ροή μόνο τα δεδομένα που απαιτούνται, δηλαδή οι μετρήσεις των αισθητήρων.

Στις ρυθμίσεις αυτού του κόμβου περιορίζουμε το msg.payload της συνολικής ροής σε message.payload.uplink_message.decoded όπως φαίνεται στην εικόνα 3.97.



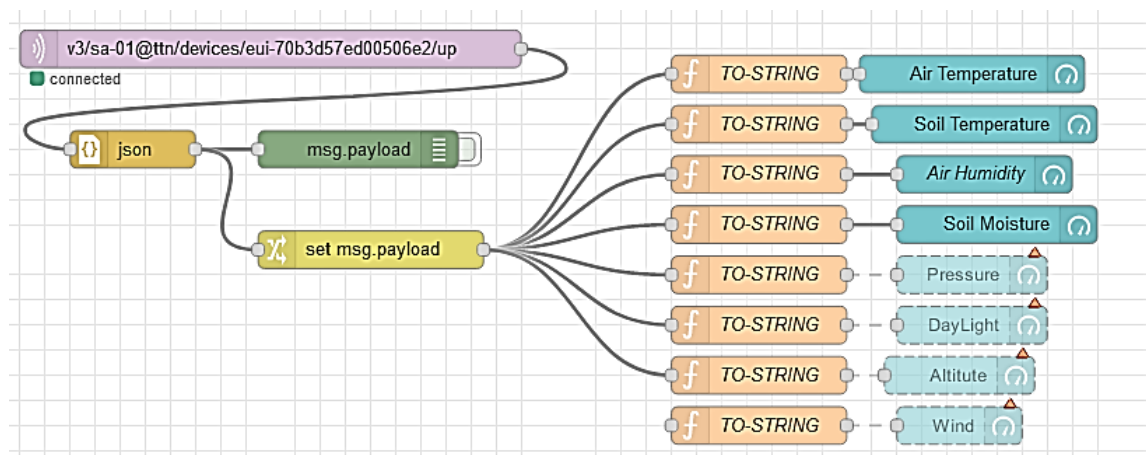
Εικόνα 3.97 – Κόμβος “change node” με τις ιδιότητές του

Έτσι, πλέον η ροή αποτελείται από ένα αντικείμενο που περιλαμβάνει μόνο τα δεδομένα των αισθητήρων.

Για την εμφάνιση αυτών των δεδομένων απαιτούνται ορισμένοι κόμβοι “gauge”, οι οποίοι δέχονται δεδομένα υπό μορφή string. Για την μετατροπή του object σε string προστίθεται

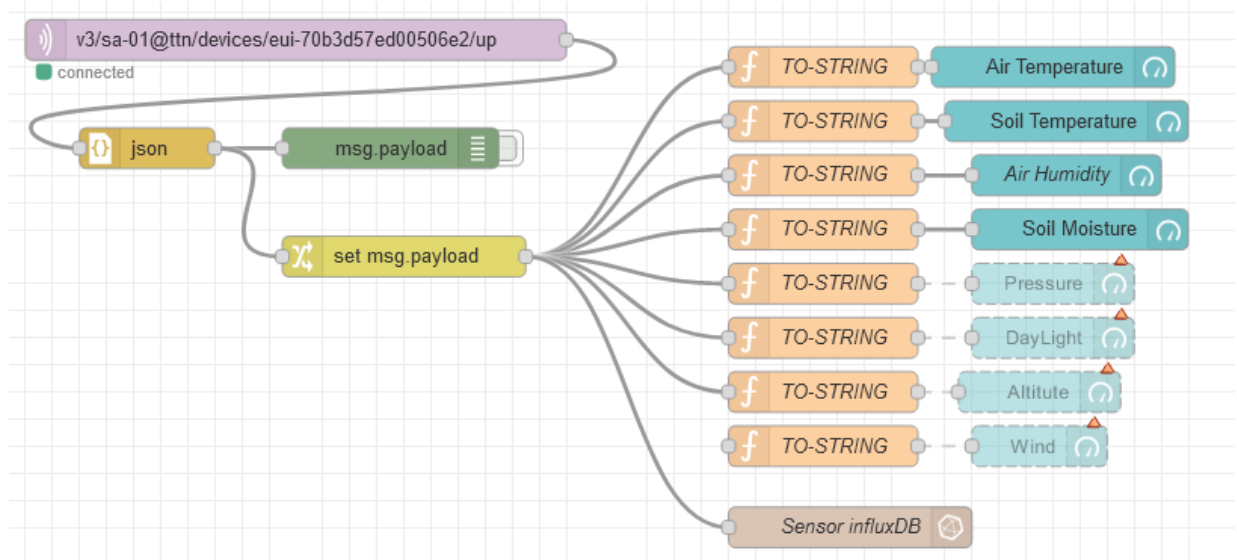
για κάθε ένα δεδομένο μέτρησης και ένας μετατροπέας κόμβος 'function' με τον απαραίτητο κώδικα μετατροπής.

Η συνολική εικόνα της ροής από το TTN προς το NodeRed για τον πολυαισθητήρα φαίνεται στην εικόνα 3.98:



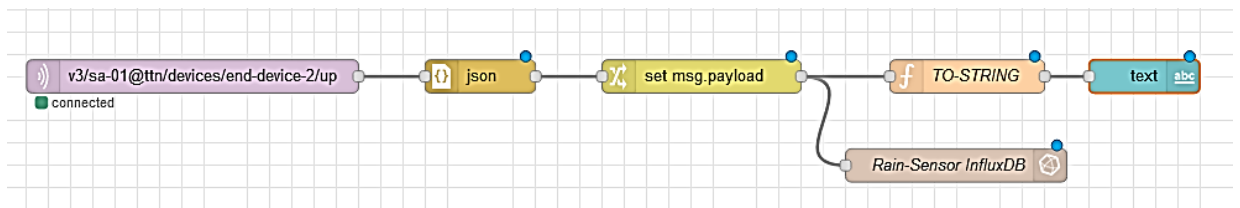
Εικόνα 3.98 – Συνολική ροή δεδομένων πολυαισθητήρα από το TTN στο NodeRed

Για να καταχωρούνται τα δεδομένα της ροής στην βάση InfluxDB που εγκαταστάθηκε στην παράγραφο 3.8.3, γίνεται προσθήκη ενός κόμβου 'InfluxDB' όπως φαίνεται στην εικόνα 3.99.



Εικόνα 3.99 – Συνολική ροή δεδομένων πολυαισθητήρα από το TTN στο NodeRed με influxDB

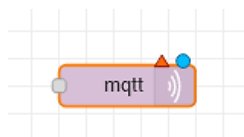
Με την ίδια ακριβώς λογική κατασκευάζεται η ροή εισαγωγής της δεύτερης τελικής συσκευής η οποία παρέχει τα δεδομένα βροχής και φαίνεται στην εικόνα 3.100.



Εικόνα 3.100 – Συνολική ροή δεδομένων αισθητήρα βροχής από το TTN στο NodeRed με influxDB

Η δεύτερη τελική συσκευή πέρα από τα δεδομένα βροχής που παρέχει είναι υπεύθυνη και για την λειτουργία των ενεργοποιητών του συστήματος. Άρα, λαμβάνει σήματα ενεργοποίησης για το πότισμα, τις θερμαντικές λάμπες, την υδρονέφωση και οποιαδήποτε άλλη μελλοντική συσκευή απαιτεί ενεργοποίηση.

Για να μπορεί να σταλεί ένα σήμα ενεργοποίησης από το NodeRed προς το TTN θα πρέπει δημιουργηθεί μια ροή εξαγωγής. Έτσι, τοποθετείται στο NodeRed ένας κόμβος mqtt-out όπως φαίνεται στην εικόνα 3.101, ο οποίος πρόκειται να στέλνει δεδομένα προς το TTN μέσω πρωτοκόλλου mqtt.

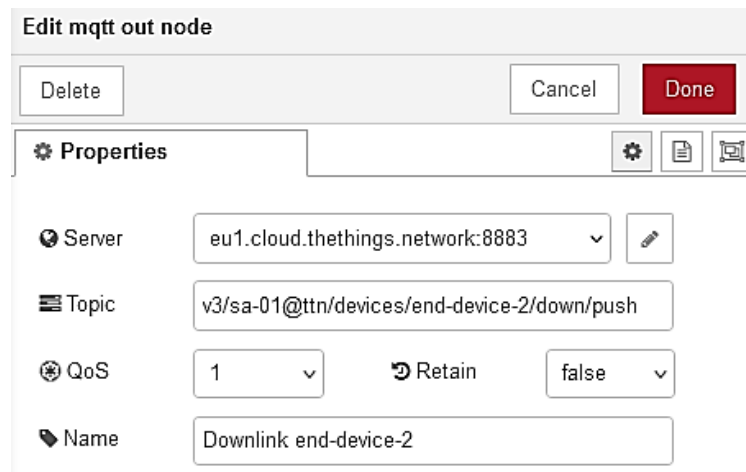


Εικόνα 3.101 – Κόμβος mqtt-out

Με δεξί κλικ επάνω στον κόμβο ρυθμίζονται οι ιδιότητες του με τρόπο αντίστοιχο που έγινε και για το mqtt-in οι οποίες φαίνονται στην εικόνα 3.101. Το μόνο πεδίο ιδιοτήτων που αλλάζει σε σχέση με το mqtt-out είναι το topic όπου δέχεται το ακόλουθο format:

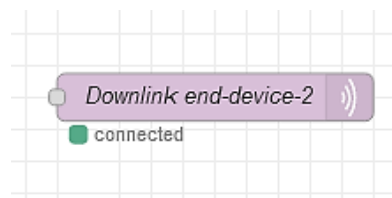
TTV Version	application +“@ttn”	“devices”	Device ID	Down/push
Το version του TTN που χρησιμοποιείται	Από ποιο application του TTN αντλούνται δεδομένα	Πρόκειται για τελική συσκευή	Id τελικής συσκευής	Για τα κατερχόμενα μηνύματα

Άρα, πρέπει να συμπληρωθεί με “ v3/sa-01@ttn/devices/end-device-2/down/push ”.



Εικόνα 3.102 – Ιδιότητες mqtt-out

Αμέσως μόλις ενημερωθεί το NodeRed παρατηρείται ότι ο κόμβος έχει συνδεθεί στην τελική συσκευή όπως φαίνεται στην εικόνα 3.103 και είναι έτοιμος να στείλει δεδομένα.



Εικόνα 3.103 – Κόμβος mqtt out συνδεδεμένος στο TTN

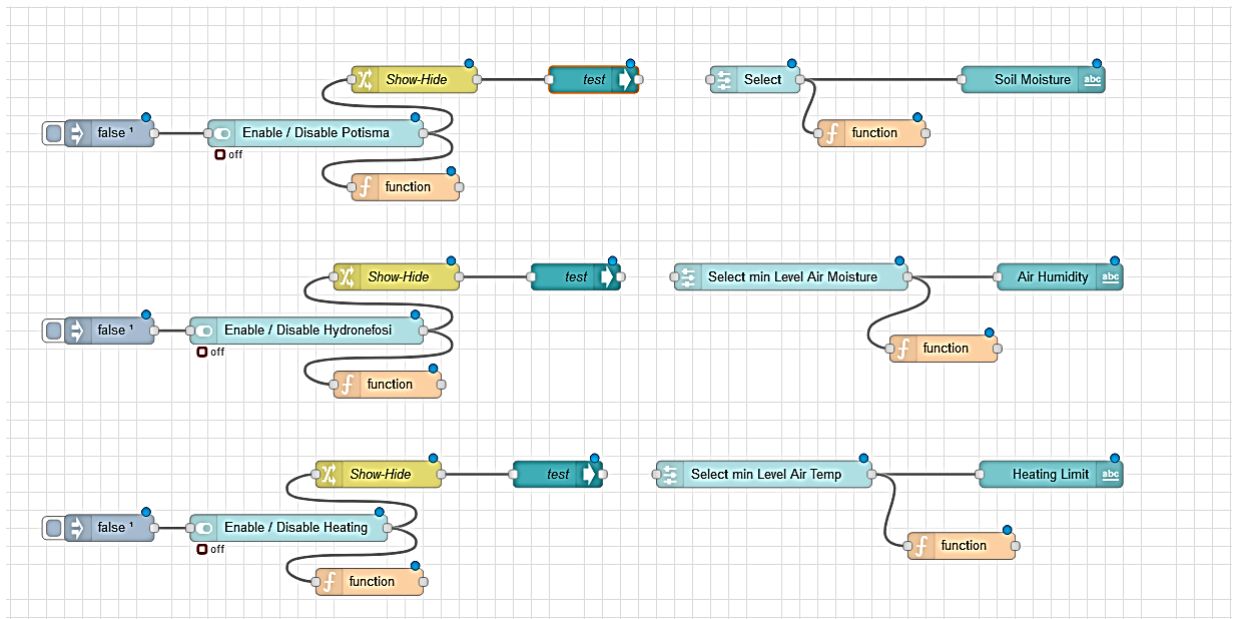
Για να μπορεί ο συγκεκριμένος κόμβος να στέλνει μηνύματα ενεργοποίησης απαιτείται μια ροή προς αυτόν με τα κατάλληλα δεδομένα.

Παρακάτω ετοιμάζεται μια ροή η οποία θα είναι υπεύθυνη για τρεις αυτοματισμούς.

- Το σύστημα θα πρέπει να ποτίζει αυτόματα για ένα ορισμένο χρονικό διάστημα όταν η υγρασία του χώματος βρίσκεται κάτω από ένα συγκεκριμένο όριο (ποσοστιαίο).
- Θα πρέπει να ενεργοποιείται αυτόματα η υδρονέφωση όταν η υγρασία της ατμόσφαιρας βρίσκεται κάτω από ένα συγκεκριμένο όριο (ποσοστιαίο).
- Θα πρέπει να ενεργοποιείται αυτόματα η θερμαντική λυχνία όταν η θερμοκρασία της ατμόσφαιρας βρίσκεται κάτω από ένα συγκεκριμένο όριο θερμοκρασίας.

Για να υλοποιηθούν οι παραπάνω αυτοματισμοί απαιτείται ένα όριο για τον καθένα το οποίο πρέπει να καθορίζεται από τον διαχειριστή του συστήματος. Για την συλλογή αυτών των ορίων κατασκευάζεται ένα user interface που αποτελείται από έναν διακόπτη ενεργοποίησης(Enable/Disable) και ένα slider επιλογής τιμής (Select).

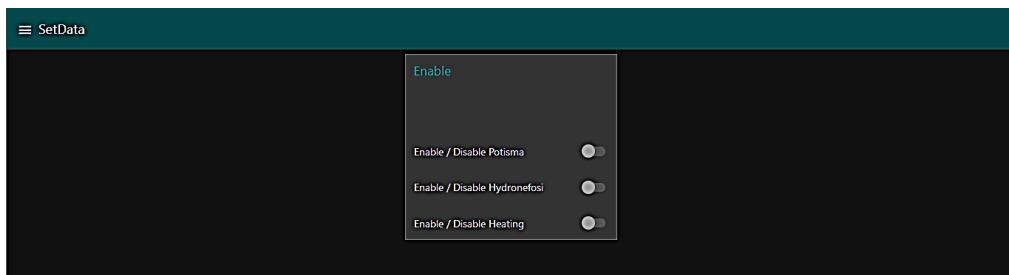
Οι ροές και τα στοιχεία που επιλέχθηκαν φαίνονται στην εικόνα 3.104 .



Εικόνα 3.104 – Καταχώρηση ορίων αυτοματισμών

Ανάλυση ροής της καταχώρησης ορίων:

Τοποθετείται ένα inject στοιχείο όπου κατά την εκκίνηση του nodeRed στέλνει ένα μήνυμα false στο στοιχείο switch (Enable/Disable) έτσι ώστε και οι τρεις ενεργοποιητές να είναι απενεργοποιημένοι κατά την εκκίνηση του προγράμματος όπως φαίνεται στην εικόνα 3.105.



Εικόνα 3.105 – Ρύθμιση ενεργοποιητών μέσω dashboard

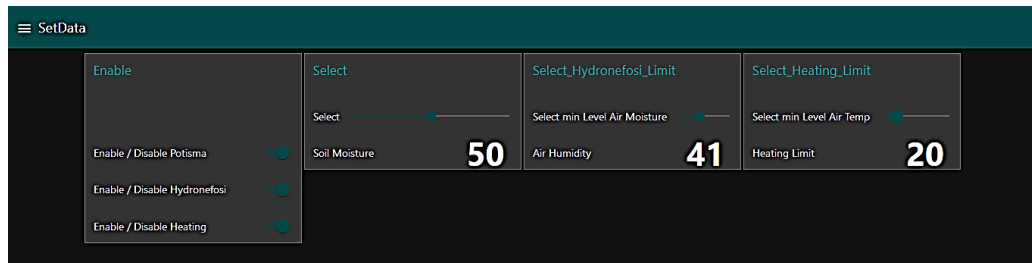
Το στοιχείο show-hide που συνδέεται μέσω ροής με το στοιχείο switch χρησιμεύει για να κρύβει και να εμφανίζει το slider επιλογής τιμής ανάλογα με την επιλογή του χρήστη. Αντίστοιχα, το στοιχείο function που συνδέεται με την ροή μετά το Switch χρησιμεύει για να αποθηκεύει την κατάσταση του switch σε μια global μεταβλητή για περαιτέρω έλεγχο της κατάστασης.

Το select στοιχείο δίνει στον χρήστη την επιλογή να καθορίζει την τιμή της ιδιότητας, όταν ο κόμβος show-hide βρίσκεται σε κατάσταση enable.

Η τιμή που επιλέγει ο χρήστης μέσω του στοιχείου select αποθηκεύεται από το στοιχείο function που ακολουθείται, σε μια global μεταβλητή και εμφανίζεται στον χρήστη μέσω ενός στοιχείου text.

Η τελική μορφή του user interface της παραπάνω ροής στοιχείων παρουσιάζεται στην εικόνα 3.106.

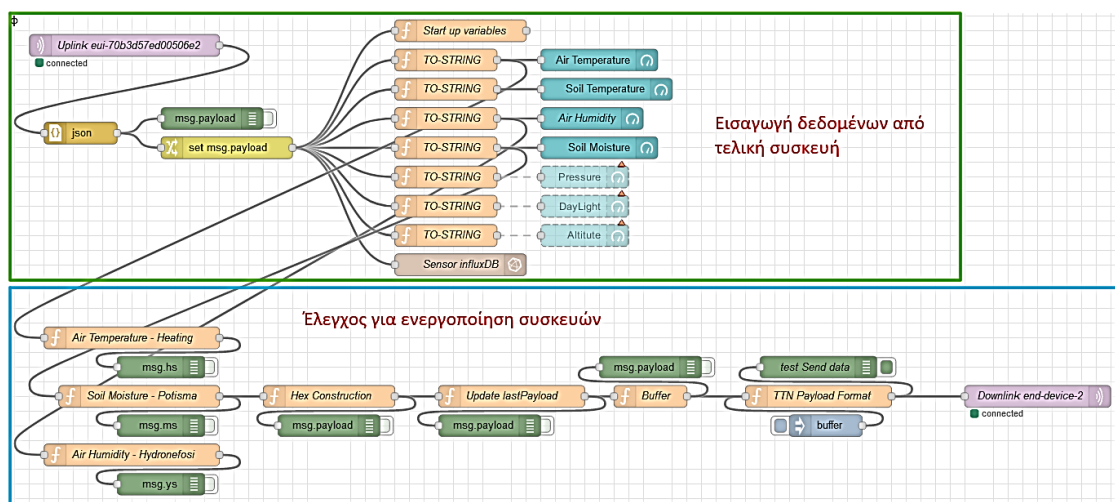
Για κάθε μια λειτουργία που ενεργοποιείται, εμφανίζεται το slider ρύθμισης, στο οποίο ο χρήστης πρέπει να ρυθμίσει στην επιθυμητή τιμή.



Εικόνα 3.106 – Ρύθμιση ενεργοποιητών του συστήματος

Αφού πλέον διατίθενται τα όρια και για τους τρεις αυτοματισμούς μπορούμε να τα συγκρίνουμε με τις πραγματικές τιμές που λαμβάνονται μέσω του πολυαισθητήρα και ανάλογα να σταλεί εντολή ενεργοποίησης ή απενεργοποίησης προς έναν ή περισσότερους ενεργοποιητές.

Η ροή υλοποίησης των ελέγχων είναι η ακόλουθη:



Εικόνα 3.107 – Συνολική ροή προς τους ενεργοποιητές του συστήματος

Στο επάνω μέρος της ροής που παρουσιάζεται στην εικόνα 3.107 φαίνεται η διασύνδεση ροής εισαγωγής με τον πολυαισθητήρα που εξηγήσαμε αναλυτικά προηγουμένως.

Το κάτω μέρος λαμβάνονται οι πραγματικές τιμές των τριών αισθητήρων που μας ενδιαφέρουν και συγκρίνονται με τις τιμές που έχει καταχωρήσει ο χειριστής του συστήματος. Αν οι πραγματικές τιμές είναι κάτω από το όριο που έχει τεθεί από τον

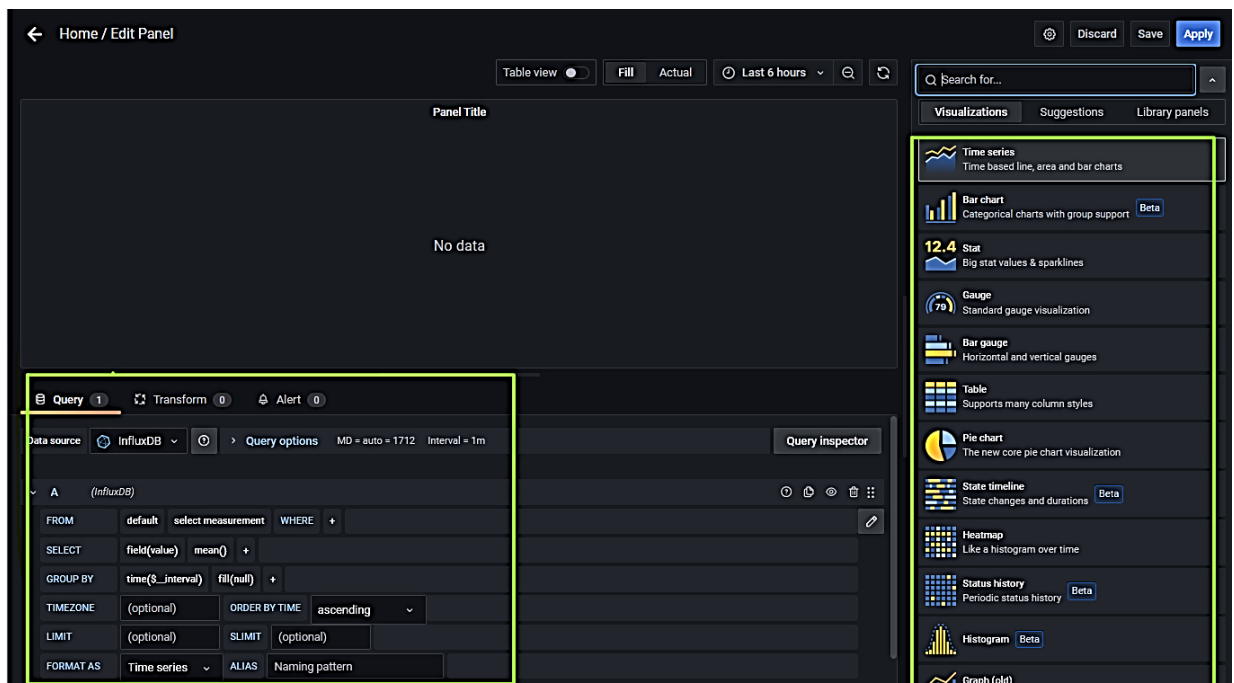
χειριστή, τότε το σύστημα ετοιμάζει μια δεκαεξαδική μεταβλητή (Hex Construction) η οποία στέλνεται στην δεύτερη τελική συσκευή έτσι ώστε να γίνει η απαραίτητη ενεργοποίηση.

Για να ολοκληρωθεί το user interface της εφαρμογής θα χρειαστούν ορισμένα γραφήματα τα οποία πρόκειται να σχεδιαστούν μέσω του Grafana καθότι μας παρέχει πολύ περισσότερες δυνατότητες ανάλυσης και σχεδιασμού από τα γραφήματα του NodeRed.

Καλείται το Grafana μέσω της ακόλουθης διεύθυνσης:

<http://smart-agri.duckdns.org:3000>

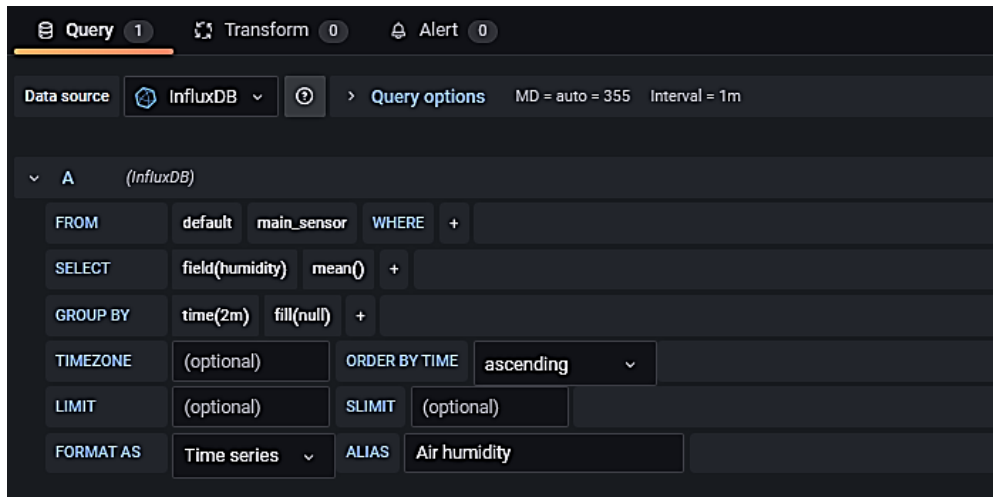
Επιλέγεται “add new panel” από το κεντρικό μενού του Grafana και εμφανίζεται η ακόλουθη οθόνη που φαίνεται στην εικόνα 3.108.



Εικόνα 3.108 – Δημιουργία γραφικών με το grafana

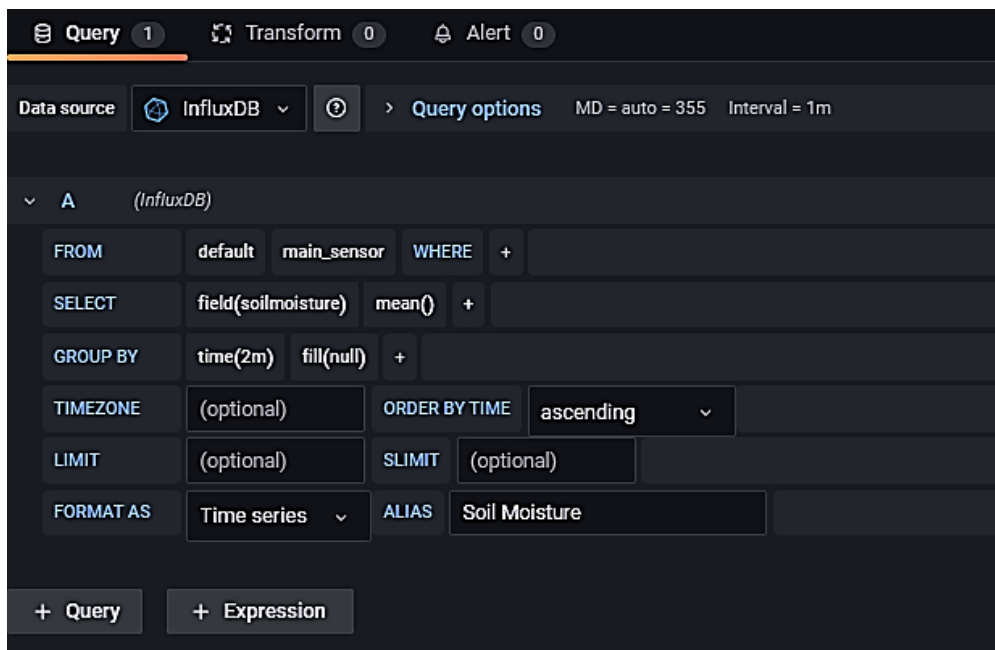
Πρώτα, στην δεξιά στήλη επιλέγεται η κατηγορία γραφικών που πρόκειται να δημιουργηθεί και στην συνέχεια κάτω αριστερά εμφανίζεται το παράθυρο με τις απαραίτητες ρυθμίσεις που πρέπει να γίνουν προς την βάση (query) έτσι ώστε να επιλέγονται τα δεδομένα που απαιτούνται.

Για την γραφική παράσταση της υγρασίας της ατμόσφαιρας δημιουργείται το ακόλουθο query προς την βάση που φαίνεται στην εικόνα 3.109 :



Εικόνα 3.109 – Query προς την influxDB από το περιβάλλον του Grafana για την κατασκευή γραφικής παράστασης που αφορά την υγρασία της ατμόσφαιρας.

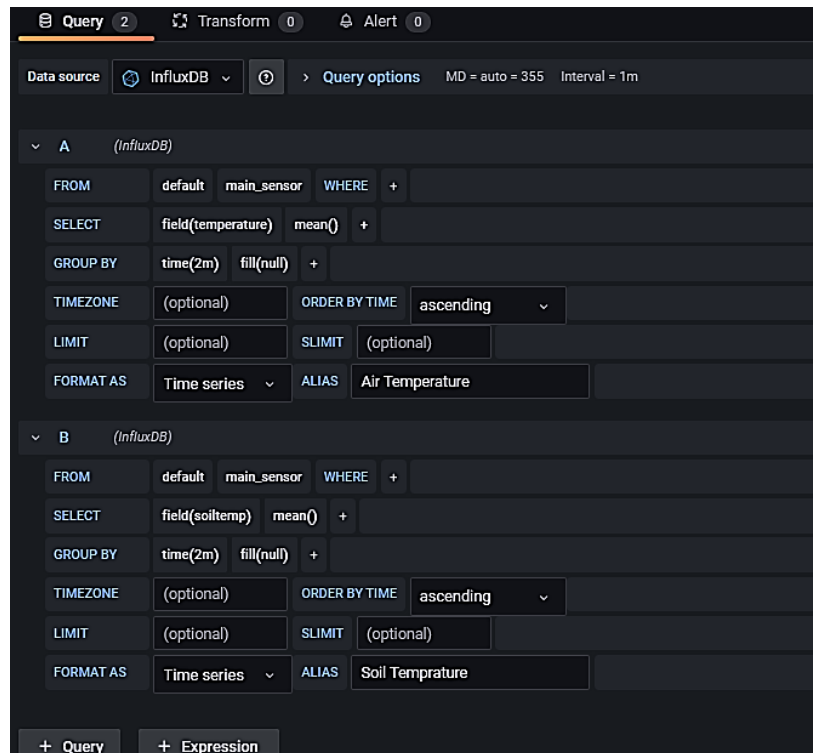
Αντίστοιχα, για την υγρασία του εδάφους το query φαίνεται στην εικόνα 3.110:



Εικόνα 3.110 – Query προς την influxDB από το περιβάλλον του Grafana για την κατασκευή γραφικής παράστασης που αφορά την υγρασία του εδάφους.

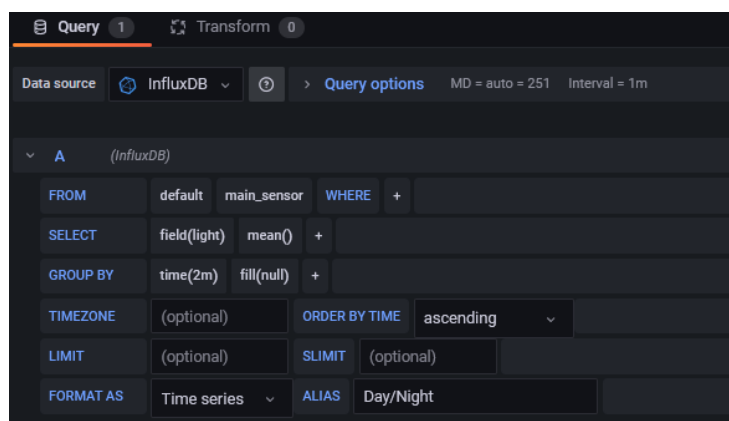
Για την θερμοκρασία του εδάφους και της ατμόσφαιρας κατασκευάζεται ένα γράφημα που να περιλαμβάνει και τις δύο γραφικές παραστάσεις έτσι ώστε, να είναι εύκολα συγκρίσιμες.

Τα πεδία που συμπληρώνονται σε αυτήν την περίπτωση παρουσιάζονται στην εικόνα 3.111.



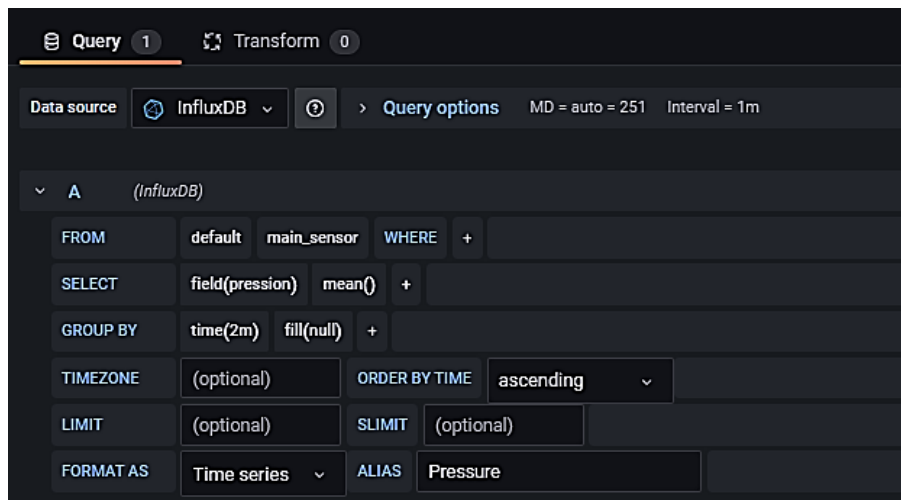
Εικόνα 3.111 – Query προς την influxDB από το περιβάλλον του Grafana για την κατασκευή γραφικής παράστασης που αφορά την θερμοκρασία του εδάφους και της ατμόσφαιρας.

Αντίστοιχα, για την ένταση του φωτός το query προς την βάση φαίνεται στην εικόνα 3.112:



Εικόνα 3.112 – Query προς την influxDB από το περιβάλλον του Grafana για την κατασκευή γραφικής παράστασης που αφορά την ένταση του φωτός.

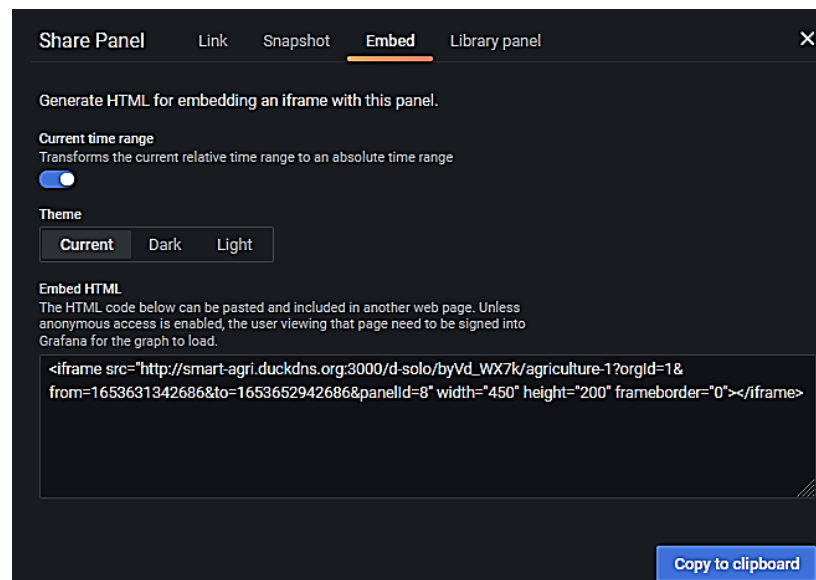
Τέλος, για την πίεση της ατμόσφαιρας επιλέγεται γραφικό τύπου 'stat', από την δεξιά στήλη της εικόνας 3.107 και στην συνέχεια δημιουργείται το ακόλουθο query προς την βάση που φαίνεται στην εικόνα 3.113.



Εικόνα 3.113 – Query προς την influxDB από το περιβάλλον του Grafana για την κατασκευή γραφικής παράστασης που αφορά την ατμοσφαιρική πίεση.

Τα γραφικά στο περιβάλλον του Grafana έχουν ολοκληρωθεί.

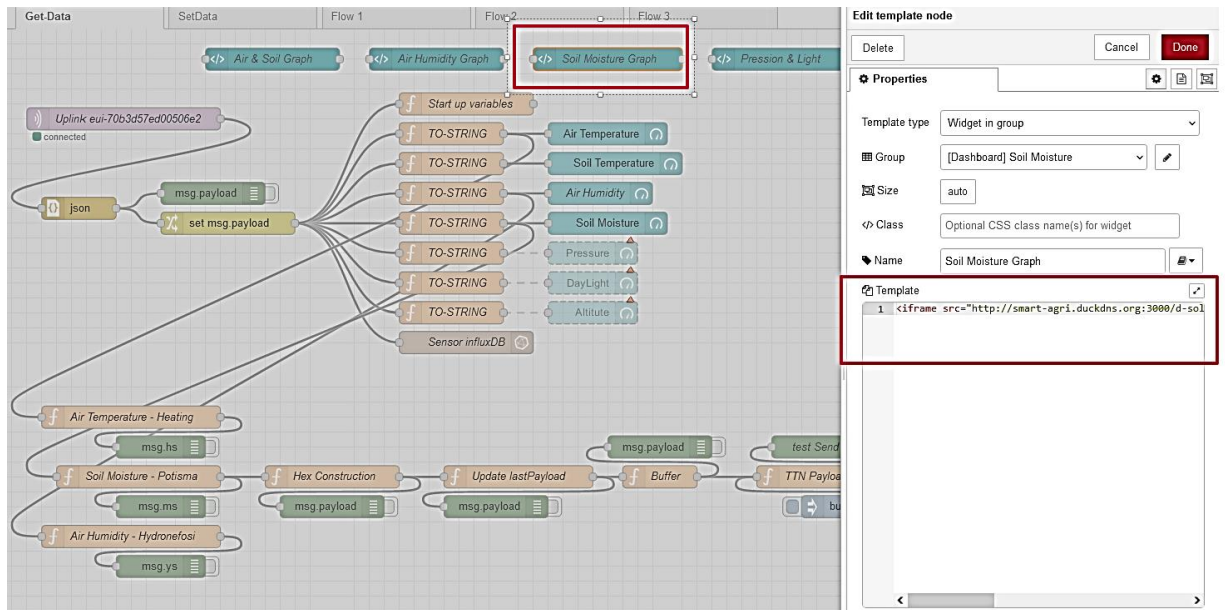
Υπολείπεται απλά η εισαγωγή τους στο περιβάλλον του NodeRed. Αυτό πραγματοποιείται μέσω της ιδιότητας embed που παρέχεται από το περιβάλλον του Grafana για κάθε γραφική παράσταση και φαίνεται στην εικόνα 3.114.



Εικόνα 3.114 – Δημιουργία 'embedding iframe' για την εισαγωγή των γραφικών στο NodeRed

Στο περιβάλλον του NodeRed για κάθε μια γραφική παράσταση που εισάγεται δημιουργείται ένα στοιχείο 'template' στο οποίο ενσωματώνεται ο κώδικας του embed

iframe όπως φαίνεται στην εικόνα 3.115. Συνολικά δημιουργούνται τέσσερα στοιχεία 'template' ένα για κάθε μια γραφική παράσταση.



Εικόνα 3.115 – Εισαγωγή του iframe στο NodeRed

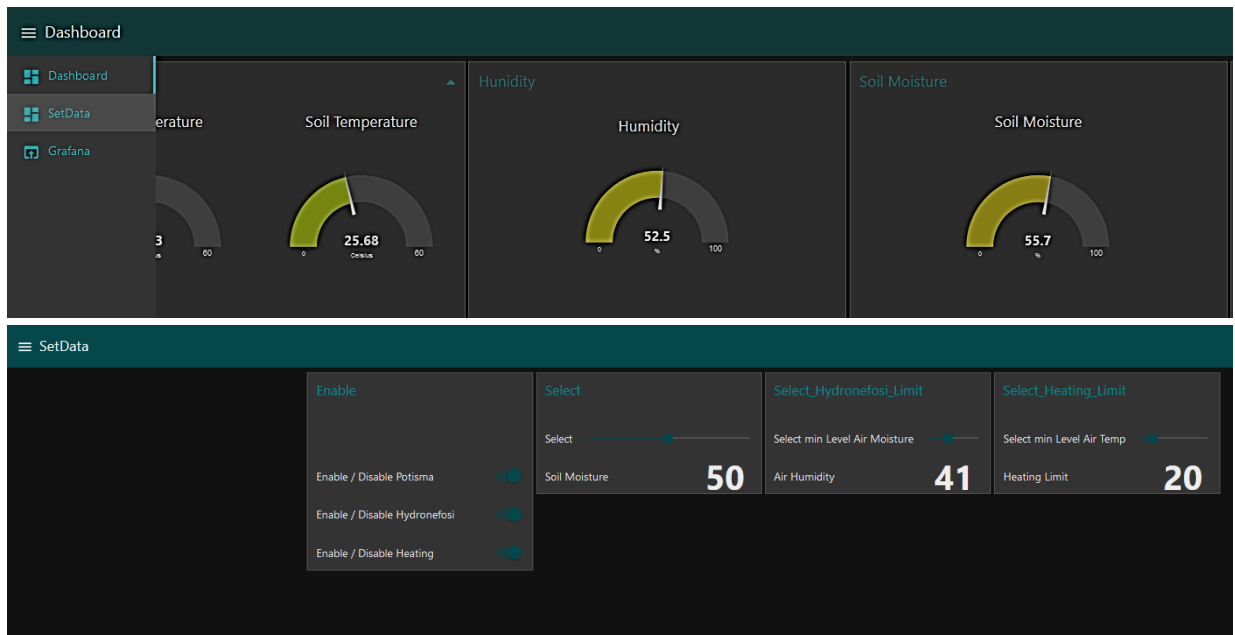
Το συνολικό user interface της εφαρμογής παρουσιάζεται στην εικόνα 3.116 και είναι διαθέσιμο διαδικτυακά με ζωντανά δεδομένα μέσω της ακόλουθης διεύθυνσης:

<http://smart-agri.duckdns.org:1880/ui> (username: user1 / password: user1@!@)



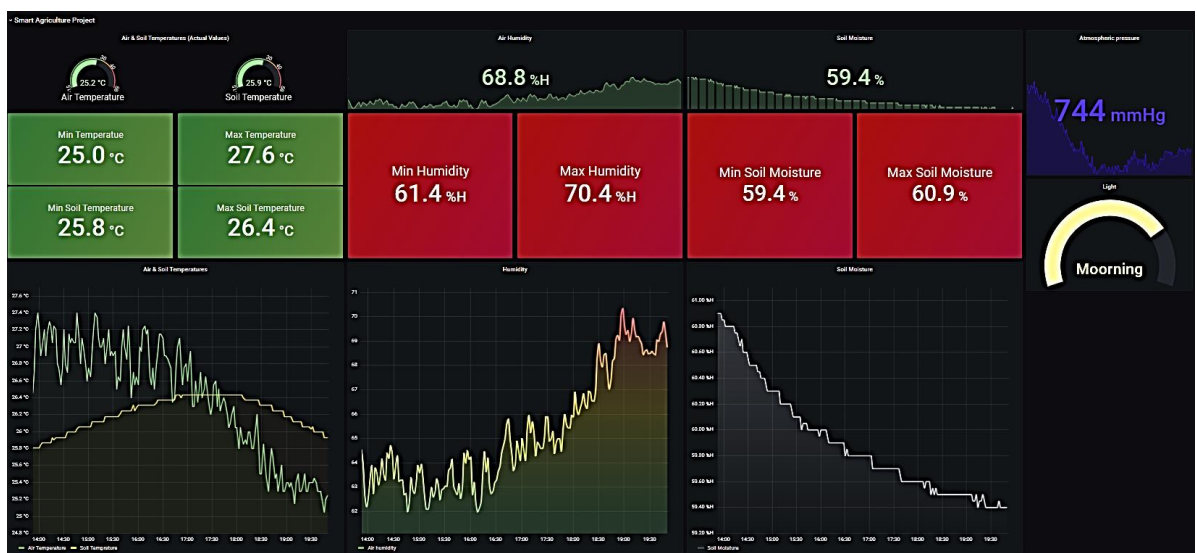
Εικόνα 3.116 – Συνολικό user interface της εφαρμογής

Οι ρυθμίσεις για τους αυτοματισμούς γίνονται από το κεντρικό μενού που βρίσκεται στην πάνω δεξιά γωνία επιλέγοντας 'SetData' όπως φαίνεται στην εικόνα 3.117.



Εικόνα 3.117 – Ρυθμίσεις ενεργοποιητών στο NodeRed

Επίσης, από το κεντρικό μενού μπορεί να πραγματοποιηθεί μετάβαση στο γραφικό περιβάλλον του 'Grafana'. Μεσω της μετάβαση αυτής παρέχεται η δυνατότητα χρονικής επιλογής των δεδομένων που εμφανίζονται στις γραφικές παραστάσεις όπως φαίνεται από τις εικόνες 3.118 (α),(β),(γ).



Εικόνα 3.118 (α) – Grafana dashboard, δεδομένα για τις έξι τελευταίες ώρες



Εικόνα 3.118 (β) – Grafana dashboard, δεδομένα για τις τρεις τελευταίες ώρες



Εικόνα 3.118 (γ) – Grafana dashboard, δεδομένα για τα 15 τελευταία λεπτά

4. Επίλογος

4.1 Εισαγωγή κεφαλαίου

Στην ενότητα αυτή παρουσιάζεται μια σύνοψη της παρούσας εργασίας δίνοντας έμφαση στα προβλήματα που αντιμετωπίστηκαν κατά την διάρκεια της κατασκευής και παράλληλα παρέχονται προτάσεις βελτίωσης για ενδεχόμενη μελλοντική εξέλιξη.

4.2 Σύνοψη εργασίας

Κατά την διάρκεια της εργασίας αφού μελετήθηκαν τα πλεονεκτήματα και μειονεκτήματα των διαφόρων τεχνολογιών LPWAN επιλέχθηκε η ανοιχτή τεχνολογία LoRa - ως η πιο πρόσφατη αλλά και πολλά υποσχόμενη - για την ασύρματη συλλογή δεδομένων από μια γεωργική περιοχή με στόχο την υλοποίηση ενός μοντέλου γεωργίας ακριβείας.

Ειδικότερα :

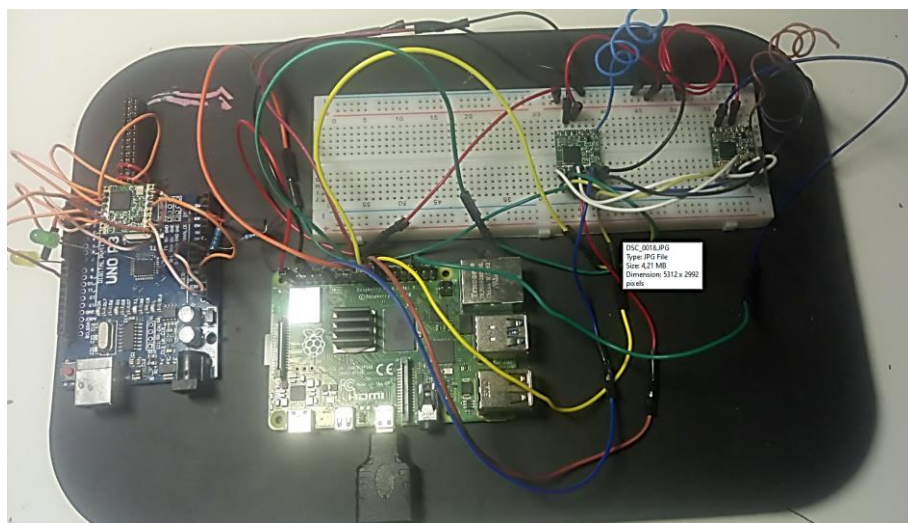
- Υλοποιήθηκε μια πρώτη ασύρματη τελική συσκευή μέσω συνδεσιμότητας LoRa με πέντε διαφορετικούς αισθητήρες. Η συγκεκριμένη συσκευή κάθε εξήντα δευτερόλεπτα παρέχει την δυνατότητα μέτρησης της θερμοκρασίας του αέρα, της θερμοκρασίας του εδάφους, της υγρασίας του αέρα, της υγρασίας του εδάφους, της έντασης του φωτός και της ατμοσφαιρικής πίεσης.
- Αντίστοιχα, υλοποιήθηκε μια δεύτερη ασύρματη τελική συσκευή με συνδεσιμότητα LoRa η οποία διαθέτει έναν ψηφιακό αισθητήρα βροχής και έχει ως κύρια αρμοδιότητα τον έλεγχο των ενεργοποιητών του συστήματος.
- Επιλέχθηκε ο δρομολογητής Mikrotik wAP LR8 kit και εφαρμόστηκαν οι κατάλληλες ρυθμίσεις που απαιτούνταν για να είναι σε θέση να συνδέεται στο TTN και να μπορεί να ανταλλάσσει πακέτων δεδομένων μέσω αυτού.
- Δημιουργήθηκε ένας νέος λογαριασμός λειτουργίας στο TTN.
- Καταχωρήθηκε ο δρομολογητής Mikrotik wAP LR8 kit στο δίκτυο TTN ως public.
- Δημιουργήθηκε η εφαρμογή 'smart-agri' στο TTN και εντάχθηκαν σε αυτήν οι δύο τελικές συσκευές.
- Τροποποιήθηκε η βιβλιοθήκη ανοικτού κώδικα LMIC (LoraWAN-MAC-in-C) έτσι ώστε να συνεργάζεται με τους αισθητήρες που εγκαταστάθηκαν στις τελικές συσκευές και να μπορούν να λαμβάνονται κατερχόμενα μηνύματα που απαιτούνται για τους ενεργοποιητές.
- Δημιουργήθηκε μια εικονική μηχανή (virtual server) και εγκαταστάθηκαν σε αυτή οι εφαρμογές NodeRed, InfluxDB και Grafana.
- Ενεργοποιήθηκε το πρωτόκολλο mqtt στο TTN και παράλληλα ρυθμίστηκαν οι απαραίτητοι κόμβοι ροών στο NodeRed.

- Αναπτύχθηκε ένα user interface στο NodeRed το οποίο με γραφικό τρόπο παρέχει άμεση πληροφόρηση των δεδομένων που λαμβάνονται από τις δύο τελικές συσκευές.
- Δημιουργήθηκαν γραφικά σύγχρονης αισθητικής μέσω της χρήσης του influxDB και του Grafana τα οποία ενσωματώθηκαν στο user interface του NodeRed.
- Αυτοματοποιήθηκε η διαδικασία ποτίσματος, υδρονέφωσης και θέρμανσης μέσω του NodeRed δίνοντας στον χρήστη την δυνατότητα επιλογής της ελάχιστης τιμής για την κάθε μια από αυτές λειτουργία.

4.3 Προβλήματα – Αντιμετώπιση

Παρακάτω γίνεται παράθεση ορισμένων προβλημάτων που εμφανίστηκαν κατά την διαδικασία της κατασκευής και ο τρόπος με τον οποίο αντιμετωπίστηκαν.

Αρχικά το μεγαλύτερο πρόβλημα που παρουσιάστηκε ήταν κατά την εκκίνηση της εργασίας όπου επιχειρήθηκε η κατασκευή ενός δρομολογητή (DIY) με την βοήθεια ενός Raspberry Pi4, δυο LoRa module RFM95W και την βιβλιοθήκη 'Dual_chan_pkt_fwd' ανοιχτού κώδικα. Η συνολική κατασκευή παρουσιάζεται στην εικόνα 4.1



Εικόνα 4.1 – Κατασκευή δρομολογητή LoRa με 2 κανάλια λειτουργίας

Ενώ έγιναν πολλές προσπάθειες για να λειτουργήσει σωστά ο συγκεκριμένος δρομολογητής δεν κατέστη δυνατόν λόγω του ότι έκανε χρήση μόνο δύο καναλιών (ένα για τα ανερχόμενα και ένα για τα κατερχόμενα) με αποτέλεσμα να μην ανεβαίνουν προς το TTN σχεδόν τα μισά πακέτα που αποστέλλονταν μέσω αυτού. Οι περισσότεροι έτοιμοι δρομολογητές χρησιμοποιούν οκτώ κανάλια για τα ανερχόμενα μηνύματα και ένα κανάλι για τα κατερχόμενα.

Τελικά, για την αποφυγή του προβλήματος επιλέχθηκε ένας έτοιμος δρομολογητής του εμπορίου και συγκεκριμένα ο Mikrotik wAP LR8 kit.

Ένα άλλο πρόβλημα που αντιμετωπίστηκε έχει να κάνει με την μέθοδο ενεργοποίησης των τελικών συσκευών. Αρχικά είχε επιλεγεί η μέθοδος ABP (Activation by Personalization) καθώς είναι πιο απλή – απαιτεί λιγότερα κλειδιά - από την μέθοδο OTAA (Over the Air Activation). Παρατηρήθηκε όμως ότι στην περίπτωση επανεκκίνησης της τελικής συσκευής το TTN δεν λάμβανε πακέτα δεδομένων για ένα μεγάλο χρονικό διάστημα ή για να μπορέσει να λάβει θα έπρεπε να γίνει διαγραφή και επανακαταχώρηση της συσκευής στο TTN.

Το πρόβλημα οφείλονταν στον έλεγχο ασφάλειας του πρωτοκόλλου LoRa σχετικά με τον frame counter που περιγράφεται στο Κεφ.2 – § 2.4.3. Δηλαδή, για κάθε μήνυμα που έστειλε η τελική συσκευή ο μετρητής FCntUp (frame counter up) αυξανόταν κατά μία μονάδα. Με την αποσύνδεση της συσκευής και την επανασύνδεση το δίκτυο λάβανε ένα μήνυμα με χαμηλότερο μετρητή από τον τελευταίο καθώς ο FCntUp ξεκινούσε πάλι από την αρχή. Έτσι, το σύστημα αντιλαμβανόταν ότι επρόκειτο για επανάληψη μηνύματος και το αγνοούσε. Η λύση που επιλέχθηκε για αυτό το πρόβλημα είναι η αλλαγή της μεθόδου ενεργοποίησης από ABP σε OTAA.

Πραγματοποιήθηκαν δοκιμές σχετικά με την εμβέλεια κάλυψης των δυο τελικών συσκευών και διαπιστώθηκε ότι εντός αστικής περιοχής με χρήση της εσωτερικής κεραίας του δρομολογητή η μέγιστη απόσταση κάλυψης είναι περίπου 250m. Για μεγαλύτερη απόσταση κάλυψης θα πρέπει να γίνει αλλαγή στην κεραία που χρησιμοποιεί ο δρομολογητής. Δηλαδή, να γίνει χρήση εξωτερικής κεραίας όπως αυτή που φαίνεται στην εικόνα 4.02 ή αλλιώς να υπάρχει κάλυψη από περισσότερους δρομολογητές.



Εικόνα 4.2 – Εξωτερικός δρομολογητής με δυο εξωτερικές κεραίες

Οι αισθητήρες παράγουν δεδομένα με δυο δεκαδικά ψηφία. Για την μεταφορά αυτών των δεδομένων κανονικά απαιτείται μεταβλητή float η οποία είναι κατάλληλη για δεκαδικούς αριθμούς μέχρι 7 ψηφία και χρειάζονται 32bit (4 bytes) για την μεταφορά της στο payload του μηνύματος. Από την στιγμή που υπάρχουν μόνο δύο δεκαδικά ψηφία μας είναι προτιμότερο να γίνει μετατροπή του δεκαδικού σε ακέραιο πολλαπλασιάζοντας με 100 και μετά την λήψη να πραγματοποιηθεί διαίρεση με το 100 έτσι ώστε να προκύψει και πάλι δεκαδικός αριθμός. Με αυτόν τον τρόπο πραγματοποιείται μεγάλη οικονομία στο payload του μηνύματος που στέλνεται.

Παρατηρήθηκε ότι ο αισθητήρας που μετράει την υγρασία του εδάφους (Waveshare) οξειδώνεται πολύ γρήγορα όπως φαίνεται στην εικόνα 4.3. Πρόκειται για έναν φτηνό αισθητήρα που δεν έχει τις απαραίτητες επικαλύψεις προστασίας με αποτέλεσμα να έχει μικρή διάρκεια ζωής όταν εκτίθεται σε υγρό περιβάλλον. Η εναλλακτικές είναι είτε να αντικατασταθεί με έναν πιο ποιοτικό αισθητήρα επικαλυμμένο με χρυσό φινίρισμα αλλά και μεγαλύτερο κόστος είτε να γίνεται αλλαγή με τον ίδιο αισθητήρα όταν κρίνεται απαραίτητο.



Εικόνα 4.3 – Οξειδωμένος αισθητήρας υγρασίας εδάφους

Ελέγχθηκε η αυτονομία της πρώτης τελική συσκευής (πολυαισθητήρας) ως προς την μπαταρία που διαθέτει και παρατηρήθηκε ότι η διάρκεια λειτουργίας της είναι περίπου 14 μέρες. Η τροφοδοσία πραγματοποιείται από δύο μπαταρίες 3,7V - 2200 mAh σε παράλληλη σύνδεση όπως αυτές που φαίνονται στην εικόνα 4.4.



Εικόνα 4.4 – Μπαταρία πολυαισθητήρα

Καθότι η αυτονομία της συσκευής είναι πολύ σημαντικός παράγοντας για την υιοθέτηση ενός τέτοιου συστήματος αναζητήθηκαν τρόποι διάφοροι τρόποι βελτίωσης.

Ένας πρώτος τρόπος είναι να χρησιμοποιηθούν μπαταρίες μεγαλύτερης χωρητικότητας οι οποίες παράλληλα θα αυξήσουν και το μέγεθος της τελικής συσκευής το οποίο δεν είναι επιθυμητό στην συγκεκριμένη περίπτωση. Ένας δεύτερος τρόπος βελτίωσης είναι η μεταβολή του duty-cycle της τελικής συσκευής, δηλαδή η μεταβολή της χρονικής απόστασης μεταξύ μηνυμάτων (Π.χ από ένα λεπτό σε πέντε, δέκα κτλ). Επίσης, ένας τρίτος τρόπος εξίσου αποτελεσματικός θα ήταν να τίθεται σε κατάσταση sleep-mode ο μικροελεγκτής για το χρονικό διάστημα που δεν στέλνει μηνύματα και να ενεργοποιείται πάλι λίγο πριν την επόμενη αποστολή.

Ορισμένες φορές παρατηρείται το φαινόμενο ότι κατά το πότισμα ο αισθητήρας υγρασίας εδάφους δεν προλαβαίνει να ενημερωθεί γρηγορά με την τα νέα δεδομένα με αποτέλεσμα να στέλνεται και δεύτερο σήμα ποτίσματος προς τον ενεργοποιητή.

Για να αποφευχθεί το συγκεκριμένο φαινόμενο προστέθηκε μια μεταβλητή (true/false) στο περιβάλλον του NodeRed ή οποία ενεργοποιείται σε κάθε πότισμα. Για όσο χρονικό διάστημα είναι ενεργοποιημένη εμποδίζει την ενεργοποίηση του επόμενου ποτίσματος. Η απενεργοποίηση αυτής της μεταβλητής πραγματοποιείται σε χρονικό διάστημα 2 λεπτών από την στιγμή της ενεργοποίησης μέσω μιας διαδικασίας τύπου cronjob που ορίζεται από το περιβάλλον του NodeRed.

Σε ότι αφορά τη χρονική καθυστέρηση ενεργοποιητών, η ελάχιστη χρονική απόσταση μεταξύ δύο σημάτων είναι 60 δευτερόλεπτα. Έτσι, αν σταλεί ένα σήμα ενεργοποίησης προς την αντλία ποτίσματος θα χρειαστούν 60 δευτερόλεπτα για να σταλεί το επόμενο που πρόκειται να απενεργοποιήσει την αντλία (Συσκευή κλάσης A). Η ποσότητα νερού για πότισμα διάρκειας ενός λεπτού είναι υπερβολικά μεγάλη σε σχέση με την έκταση της κατασκευής και λογικά θα καταστρέψει το φυτό. Για τον λόγο αυτόν χρησιμοποιείται μια χρονική μεταβλητή στην τελική συσκευή η οποία διατηρεί ενεργοποιημένη την αντλία ποτίσματος για 10 δευτερόλεπτα από την στιγμή της ενεργοποίησης.

Το ίδιο ακριβώς συμβαίνει και για την ενεργοποίηση της υδρονέφωσης.

4.4 Συμπεράσματα – Παρατηρήσεις

Η κατασκευή φαίνεται να λειτουργεί πολύ καλά. Με ένα μικρό κόστος υλοποίησης παρέχονται τα κρίσιμα δεδομένα που αντικατοπτρίζουν την κατάσταση που επικρατεί στον χώρο της καλλιέργειας, άμεσα και με κατανοητό τρόπο καθώς επίσης δίνεται η δυνατότητα αυτοματοποίησης ορισμένων διαδικασιών.

Η επιλογή της τεχνολογίας LoRa καλύπτει τις ανάγκες του συστήματος σχετικά με τον όγκο δεδομένων, την ασφάλεια και την εμβέλεια ενώ παράλληλα συμβάλλει στην μειωμένη κατανάλωση ενέργειας. Επιπρόσθετα, η χρήση του TTN είναι δωρεάν για τα ανοδικά μηνύματα για τις δέκα πρώτες τελικές συσκευές χωρίς περιορισμό σχετικά με τον αριθμό μηνυμάτων, ενώ για τα καθοδικά μηνύματα επιτρέπεται η αποστολή δέκα μηνυμάτων ανά ημέρα για κάθε τελική συσκευή. Σε κάθε περίπτωση πάντως, αν δεν μας καλύπτει το περιβάλλον του TTN μπορούμε να εγκαταστήσουμε το δικό μας περιβάλλον διαχείρισης LoRa χρησιμοποιώντας την εφαρμογή ανοικτού κώδικα 'Chirp Stack'.

Επίσης, η μεταφορά των δεδομένων στο περιβάλλον του NodeRed μέσω πρωτοκόλλου mqtt παρέχει απεριόριστες δυνατότητες εμφάνισης των δεδομένων μέσω user interface αλλά και σύνδεσης μέσω άλλων εφαρμογών στον τομέα του IoT.

4.5 Προτάσεις μελλοντικής εξέλιξης

Παρακάτω ακολουθούν ορισμένες βελτιωτικές προτάσεις του συστήματος:

1. Προσθήκη ένδειξης επιπέδου μπαταρίας για κάθε τελική συσκευή

Μια σημαντική προσθήκη είναι να μπορεί η κάθε τελική συσκευή να στέλνει δεδομένα σχετικά με το επίπεδο της μπαταρίας που διαθέτει μαζί με τα υπόλοιπα δεδομένα. Έτσι, ο εκάστοτε χειριστής του συστήματος μπορεί να γνωρίζει τις τελικές συσκευές που έχουν ανάγκη για αλλαγή μπαταρίας.

2. Προσθήκη περισσότερων αισθητήρων

Ανάλογα με την ανάγκες της καλλιέργειας μπορούν να προστεθούν και οι κατάλληλοι αισθητήρες στις τελικές συσκευές. Π.χ μπορεί να προστεθεί ένας αισθητήρας που να μετράει το pH του εδάφους, τα νιτρικά του εδάφους την δύναμη του αέρα κτλ.

3. Προσθήκη περισσότερων τελικών συσκευών με χρήση GPS

Όταν πρόκειται για μεγάλες εκτάσεις είναι απαραίτητο να υπάρχουν αρκετές τελικές συσκευές έτσι ώστε παρέχονται δεδομένα για την συνολική έκταση και να εντοπίζεται η παραλλακτικότητα του αγρού.

4. Προσθήκη υπηρεσίας μετεωρολογικών προγνώσεων

Στο NodeRed θα μπορούσε να προστεθεί μια υπηρεσία πρόγνωσης καιρού για την γύρω περιοχή έτσι ώστε να λαμβάνονται αυτόματα μέτρα μέσω ενεργοποιητών στις περιπτώσεις που προβλέπονται κακές καιρικές συνθήκες.

5. Σύνδεση με ολοκληρωμένα WSNs

Για αγροκαλλιέργειες που καλύπτουν μεγάλες εκτάσεις, ενδείκνυται η ανάπτυξη ασυρμάτων δικτύων αισθητήρων ευρείας κλίμακας καθώς η απουσία καλωδίωσης ελαχιστοποιείτο κόστος και το χρόνο ανάπτυξής τους [4.1]. Στην περίπτωση αυτή κρίνεται επιτακτική η υιοθέτηση πρωτοκόλλων ενεργειακά αποδοτικής δρομολόγησης δεδομένων ώστε να επιμηκύνεται χρονικά η ενεργειακή επάρκεια των ασύρματων κόμβων τους [4.2].

6. Υιοθέτηση μοντέλων αυτοματοποιημένης γεωπονίας

Για αγροκαλλιέργειες που καλύπτουν μεγάλες εκτάσεις, ενδείκνυται επίσης η χρησιμοποίηση ολοκληρωμένων μοντέλων αυτοματοποιημένης γεωπονίας που διέπουν τον κεντρικό έλεγχο των δικτύων ανάλογα με τις αρχές της γεωπονικής επιστήμης και τις επικρατούσες συνθήκες [4.3]

Βιβλιογραφικές αναφορές

[1.1]

<https://www.imathiotikigi.gr/index.php/kalliergies/item/3137-oi-3-paragontes-pou-epireazoun-tis-apodoseis-ton-kalliergeion>

[1.2]

<http://www.gaiapedia.gr/gaiapedia/index.php/%CE%93%CE%B5%CF%89%CF%81%CE%B3%CE%AF%CE%B1%CE%B1%CE%BA%CF%81%CE%B9%CE%B2%CE%B5%CE%AF%CE%B1%CF%82>

[1.3] - <https://www.cropin.com/iot-in-agriculture>

[1.4] - https://papers.ssrn.com/sol3/Delivery.cfm/SSRN_ID3397786_code2087841.pdf?abstractid=3397786&mirid=1

[2.1] - <https://build.sigfox.com/sigfox-radio-configurations-rc>

[2.2] - <https://www.sigfoxwiki.com/wiki/SIGFOX>

[2.3] - <https://www.ingenu.com/technology/rpma/>

[2.4] - <https://www.mokolora.com/el/lora-geolocation-is-a-trend/>

[2.5] - https://lora-developers.semtech.com/uploads/documents/files/LoRaWAN_Firmware_Updates_OTA_NSornin_Final.pdf

[2.6] - <https://www.genericnode.com/>

[2.7] - <https://iotfactory.eu/products/iot-sensors/indoor-environment-quality-monitoring-lorawan-sensor/>

[2.8] - <https://iotfactory.eu/products/iot-sensors/lorawan-water-meter/>

[2.9] - <https://wiki.seeedstudio.com/The-Things-Indoor-Gateway/>

[2.10] - <https://www.kerlink.com/iot-solutions/iot-gateways/wirnet-indoor-lorawan-gateways/>

[2.11] - <https://www.dragino.com/products/lora-lorawan-gateway/item/140-lg308.html>

[2.12] - <https://www.multitech.com/brands/multiconnect-conduit-ip67>

[2.13] - <https://it-market.com/en/router/wireless/cisco/ixm-lpwa-800-16-k9>

[2.14] - https://mikrotik.com/product/wap_lr8_kit

[2.15] - <https://www.taoglas.com/product-category/lorasigfoxlpwa-antennas/internal-lora-antennas/>

[2.16] - <https://cryptostation.store/product/lora-fiberglass-antenna-kit-863-870mhz-8dbi-1300mm-rf-explorer-rfela-4-88x/>

[2.17] - <https://www.globalsources.com/RFID-antennas/Lora-antenna-1184407539p.htm>

[2.18] - https://lora-alliance.org/resource_hub/rp2-1-0-3-lorawan-regional-parameters/

[2.19] - <https://github.com/TheThingsNetwork/gateway-conf>

[2.20] – https://el.wikipedia.org/wiki/Διασπορά_Φάσματος

[2.21] - https://en.wikipedia.org/wiki/Frequency-hopping_spread_spectrum

[2.22] – <https://www.rfwireless-world.com/Terminology/What-is-difference-between-Chip-and-Chirp-in-LoRaWAN.html>

[2.23] - <https://josefntd.com/2018/08/14/spreading-factor-bandwidth-coding-rate-and-bit-rate-in-lora-english/>

[2.24] - https://i.mt.lv/cdn/product_files/LoRa_200558.pdf

[3.1] - <https://www.theengineeringprojects.com/2019/02/introduction-to-dht22.html>

[3.2] - <https://learnelectronics.gr/ds18b20->

[%CE%B1%CE%B9%CF%83%CE%B8%CE%B7%CF%84%CE%AE%CF%81%CE%B1%CF%82-%CE%B8%CE%B5%CF%81%CE%BC%CE%BF%CE%BA%CF%81%CE%B1%CF%83%CE%AF%CE%B1%CF%82/](https://learnelectronics.gr/ds18b20-%CE%B1%CE%B9%CF%83%CE%B8%CE%B7%CF%84%CE%AE%CF%81%CE%B1%CF%82-%CE%B8%CE%B5%CF%81%CE%BC%CE%BF%CE%BA%CF%81%CE%B1%CF%83%CE%AF%CE%B1%CF%82/)

[3.3] - <https://www.bosch-sensortec.com/products/environmental-sensors/pressure-sensors/bmp280/>

[3.4] - <https://www.waveshare.com/moisture-sensor.htm>

[3.5] - <https://components101.com/resistors/ldr-datasheet>

[3.6] - <https://skycraftsurplus.com/products/fc-37-rain-sensor-with-yl-38-comparator-module-arduino-compatible.html>

[3.7] - <https://www.hoperf.com/modules/lora/RFM95.html>

[4.1] Kandris, D., Nakas, C., Vomvas, D., & Koulouras, G. (2020). Applications of wireless sensor networks: an up-to-date survey. *Applied System Innovation*, 3(1), 14.

[4.2] Nakas, C., Kandris, D., & Visvardis, G. (2020). Energy efficient routing in wireless sensor networks: a comprehensive survey. *Algorithms*, 13(3), 72.

[4.3] Nikolidakis, S. A., Kandris, D., Vergados, D. D., & Douligeris, C. (2015). Energy efficient automated control of irrigation in agriculture by using wireless sensor networks. *Computers and Electronics in Agriculture*, 113, 154-163.