



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ**

**ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ**

**ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ**

**Διπλωματική Εργασία**

**Μελέτη και προσομοίωση δικτύου 6LoWPAN**



**Φοιτητής: Δημήτριος Τσέλλος**

**ΑΜ: 06716**

**Επιβλέπων Καθηγητής**

**Καμινάρης Σταύρος**

**Καθηγητής του Τμήματος Ηλεκτρολόγων και Ηλεκτρονικών Μηχανικών της  
Σχολής Μηχανικών του Πανεπιστημίου Δυτικής Αττικής**

**ΑΘΗΝΑ-ΑΙΓΑΛΕΩ, ΜΑΡΤΙΟΣ 2024**



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

**Diploma Thesis**

**Design and simulation of a 6LoWPAN network**



**Student: Dimitrios Tsellos  
Registration Number: 06716**

**Supervisor**

**Kaminaris Stavros  
Professor at Department of Electrical and Electronics Engineering UniWA  
Faculty of Engineering**

**ATHENS-EGALEO, MARCH 2024**

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

Καμινάρης Σταύρος, Καθηγητής ΠΑ.Δ.Α	Πυρομάλης Δημήτριος Αν. Καθηγητής ΠΑ.Δ.Α	Ιωαννίδης Γεώργιος, Καθηγητής ΠΑ.Δ.Α

**Copyright ©** Με επιφύλαξη παντός δικαιώματος. All rights reserved.

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ Δημήτριος Τσέλλος,  
Μάρτιος, 2024**

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/την 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 ΤΣΕΛΛΟΣ ΔΗΜΗΤΡΙΟΣ του ΑΓΓΕΛΟΥ, με αριθμό μητρώου 06716 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.

Επιθυμώ την απαγόρευση πρόσβασης στο πλήρες κείμενο της εργασίας μου μέχρι ..... και έπειτα από αίτησή μου στη Βιβλιοθήκη και έγκριση του επιβλέποντος/ουσας καθηγητή/ήτριας.»

Ο Δηλών  
ΤΣΕΛΛΟΣ ΔΗΜΗΤΡΙΟΣ



## **ΕΥΧΑΡΙΣΤΙΕΣ**

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή ΠΑ.Δ.Α κ. Σταύρο Καμινάρη, για την ευκαιρία που μου έδωσε, να δουλέψω και να αναπτύξω τη θεματολογία της Διπλωματικής μου εργασίας.

Θα ήθελα επίσης να ευχαριστήσω ιδιαίτερος τον κ. Ορφανό Βασίλειο, Εργαστηριακό Διδακτικό Προσωπικό του «Ερευνητικού Εργαστηρίου Κτηριακών και Βιομηχανικών Ενεργειακών Συστημάτων» για την αμέριστη υποστήριξη, που μου παρείχε κατά τη διάρκεια της συγγραφής της παρούσας Διπλωματικής εργασίας.

## **Περίληψη**

Στην παρούσα Διπλωματική εργασία γίνεται μια ενδελεχής παρουσίαση των WPAN δικτύων. Πραγματοποιείται μια εκτενής παρουσίαση των χαρακτηριστικών του πρωτοκόλλου 802.15.4. Αναλύεται η δομή του όπως και οι λειτουργίες του. Σταχυολογούνται επίσης πρωτόκολλα τα οποία είναι βασισμένα στο πρότυπο 802.15.4, σχετικά με την προέλευσή τους, τη δομή τους και την πρακτική λειτουργία τους στην καθημερινότητά μας, υπό το πρίσμα του Διαδικτύου των Πραγμάτων. Παρουσιάζονται τα τεχνικά χαρακτηριστικά και οι προδιαγραφές κάτω από τις οποίες αυτά λειτουργούν. Ειδικότερα δίνεται έμφαση στο πρωτόκολλο 6LoWPAN και στην παρουσίαση της λειτουργίας του σε μια σειρά προσομοιώσεων με την χρήση της εφαρμογής Contiki COOJA. Η εφαρμογή μελετάται σε βάθος μέσω περιγραφής των λειτουργιών της, αλλά και των διαφορετικών επιπέδων ανάπτυξης των δυνατοτήτων της. Πιο συγκεκριμένα, η μελέτη γίνεται με χρήση τριών διαφορετικών σεναρίων ανάπτυξης προσομοιωμένων κόμβων της εφαρμογής. Επιπρόσθετα, η παρουσίασή γίνεται αναλυτικότερη με χρήση εικόνων από τις προσομοιώσεις, όπως επίσης και του κώδικα, κομμάτια του οποίου έχουν διαμορφωθεί καταλλήλως, προκειμένου να επιτευχθεί το τελικό αποτέλεσμα.

## **Λέξεις – κλειδιά**

Πρωτόκολλα  
Διαδίκτυο των Πραγμάτων (Internet of Things, IoT)  
Έξυπνο σπίτι  
6LoWPAN  
Contiki COOJA  
802.15.4  
Προσομοίωση

## **Abstract**

In this Diploma thesis a thorough presentation of WPAN networks is made. An extensive presentation of the features of the 802.15.4 protocol is performed. Its structure is analyzed as well as its functions. Protocols which are based on the 802.15.4 standard are also sampled, regarding their origin, their structure and their practical operation in our daily life, in the light of the Internet of Things. The technical characteristics and specifications under which they operate are presented. In particular, emphasis is placed on the 6LoWPAN protocol and the presentation of its operation in a series of simulations using the Contiki COOJA application. The application is studied in depth through a description of its functions, but also of the different levels of development of its capabilities. More specifically, the study is done using three different deployment scenarios of simulated nodes of the application. In addition, the presentation is made more detailed by using images from the simulations, as well as the code, parts of which have been properly configured, in order to achieve the final result.

## **Keywords**

Communication Protocols

Internet of Things (IoT)

Home Automation

6LoWPAN

Contiki COOJA

802.15.4

Simulation

## Περιεχόμενα

Κατάλογος Πινάκων .....	11
Κατάλογος Εικόνων .....	11
Αλφαβητικό Ευρετήριο .....	15
ΕΙΣΑΓΩΓΗ .....	16
ΚΕΦΑΛΑΙΟ 1ο : IoT - WSN .....	18
1.1 Internet of Things (IoT) .....	18
1.1.1 Η Δομή και η Λειτουργία των συσκευών .....	18
1.2 Ασύρματα Δίκτυα Αισθητήρων - Wireless Sensors Network.....	20
1.2.1 Τα σημεία εφαρμογής τους .....	20
1.2.2 Τοπολογίες.....	21
ΚΕΦΑΛΑΙΟ 2ο : Πρωτόκολλο 802.15.4 .....	24
2.1 Γενική Περιγραφή .....	24
2.2 IEEE 802.15.4 και μοντέλο OSI.....	25
2.3 Ανάλυση επιπέδου Physical.....	26
2.3.1 Εύρος Διακύμανσης Συχνοτήτων Δεδομένων .....	27
2.3.2 Αναθέσεις Καναλιού.....	28
2.3.3 Ανίχνευση Ενέργειας .....	29
2.3.4 Δείκτης Ποιότητας Σύνδεσης.....	29
2.4 Επίπεδο Ελέγχου Πρόσβασης Δεδομένων (MAC Media Access Control).....	30
2.4.1 Υπερπλαίσιο.....	30
2.4.2 Ανταλλαγή Δεδομένων .....	30
2.4.3 Δομή Πλαισίου .....	32
2.4.4 CSMA-CA .....	32
2.4.5 Υπηρεσίες .....	33
2.4.6 Κατανάλωση Ισχύος .....	34
ΚΕΦΑΛΑΙΟ 3ο : Πρωτόκολλα βασισμένα στο 802.15.4.....	35
3.1 Πρωτόκολλα βασισμένα στο 802.15.4.....	35
3.1.1 Το πρότυπο “ISA100.11a” .....	35
3.1.2 WirelessHART .....	35
3.1.3 Thread .....	36
3.2 Zigbee .....	38
3.1.1 Υπηρεσίες .....	39
3.1.2 Συσκευές.....	39
3.1.3 Σύνδεση συσκευών.....	41
3.1.4 Θεμελιικά Στοιχεία.....	41
3.1.5 Φορείς Δεδομένων - Διαχείριση – Σημεία Ελέγχου Υπηρεσιών .....	41
3.1.6 Μονάδες Δεδομένων Υπηρεσιών – Μονάδες Δεδομένων Πρωτοκόλλου .....	42
ΚΕΦΑΛΑΙΟ 4ο : Το πρωτόκολλο 6LoWPAN .....	44
4.1 Συσχέτιση με το 802.15.4 .....	44
4.2 Δομή 45	
4.3 Επίπεδο Εφαρμογής.....	46
4.3.1 Μορφή Επικεφαλίδας .....	47
4.3.2 Επικεφαλίδα Κατακερματισμού .....	48
4.3.3 Επικεφαλίδα Διευθυνσιοδότησης Πλέγματος.....	49
4.3.4 Συμπύεση Επικεφαλίδας – RFC-4944 .....	50
4.3.5 Συμπύεση Προθέματος UDP (RFC-4944) .....	51
4.3.6 Παραδείγματα Συμπύεσεων .....	52



4.3.7 Βελτιωμένη Συμπύεση Προθέματος - UDP/IPv6 .....	53
4.3.8 Βελτιωμένα Παραδείγματα Συμπύεσης UDP/IPv6.....	54
4.4 IPv6/6LoWPAN - Αρχιτεκτονική .....	55
4.4.1 IEEE 802.15.4 στην πράξη .....	55
4.4.2 Τύπος Δρομολόγησης.....	56
4.4.3 Διευθυνσιοδότηση -Αυτόματη Διαμόρφωση.....	58
4.4.4 Εύρεση γειτονικών σταθμών .....	58
4.4.5 Mobile Ad-hoc NETwork - MANET .....	59
4.4.6 Ασφάλεια.....	59
ΚΕΦΑΛΑΙΟ 5ο : Contiki COOJA .....	60
5.1 Προέλευση.....	60
5.2 Χαρακτηριστικά .....	60
5.3 Κόμβοι.....	61
5.3.1 Sky Mote.....	61
5.3.1.1 Περιγραφή υλικού.....	61
5.3.1.2 Βασικά Χαρακτηριστικά .....	61
5.3.1.3 Περιγραφή του Module .....	62
5.3.1.4 Τροφοδοσία .....	63
5.3.1.5 Τυπικές συνθήκες λειτουργίας.....	63
5.3.1.6 Μπλοκ Διάγραμμα Tmote Sky .....	64
5.3.2 Z1 Mote - (Zolertia Mote) .....	64
5.3.2.1 Περιγραφή υλικού.....	65
5.3.2.2 Βασικές λειτουργίες .....	65
5.3.3.3 Βασικά Χαρακτηριστικά .....	65
5.3.3.4 Το module .....	65
5.3.3.5 Μπλοκ Διάγραμμα Z1 Mote.....	66
5.4 Threads και Protothreads .....	66
5.5 Διασυνδέσεις COOJA.....	67
5.6 Δομές μνήμης .....	69
5.7 JNI 69	
5.8 Πρόσθετα .....	70
5.9 Μέσα Μετάδοσης .....	72
5.10 Σύστημα Διαμόρφωσης.....	73
5.11 G.U.I .....	73
5.12 Ρυθμιστές θέσης και διανομείς IP .....	74
5.13 Διαμόρφωση Περιβάλλοντος Εκκίνησης .....	74
ΚΕΦΑΛΑΙΟ 6ο : Εγκατάσταση – Λειτουργία - Simulation .....	75
6.1 Εγκατάσταση Contiki.....	75
6.2 Σενάριο 1 <sup>ο</sup> .....	77
6.2.1 Προετοιμασία Σεναρίου 1.....	78
Η διεργασία ξεκινά όπως φαίνεται στη φωτογραφία 6.5 στο περιβάλλον υποδοχής της εφαρμογής, αφού έχει επιλεγεί η δημιουργία μιας νέας προσομοίωσης. ....	78
6.2.2 Αλλαγές .....	78
6.2.3 Εφαρμογή Σεναρίου .....	83
6.3 Σενάριο 2 <sup>ο</sup> .....	88
6.3.1 Προετοιμασία Σεναρίου 2.....	88
6.3.2 Παρούσα κατάσταση.....	89
6.3.3 Αλλαγές .....	93
6.3.4 Εκτέλεση .....	96
6.4 Σενάριο 3 <sup>ο</sup> .....	96
6.4.1 Εκτέλεση .....	97

<b>6.4.1 Μια εναλλακτική ματιά .....</b>	<b>102</b>
<b>Συμπεράσματα .....</b>	<b>104</b>
<b>Βιβλιογραφία .....</b>	<b>105</b>
<b>ΠΑΡΑΡΤΗΜΑ Α .....</b>	<b>113</b>

## **Κατάλογος Πινάκων**

Πίνακας 2.1 Εύρος συχνοτήτων και ταχύτητα δεδομένων

Πίνακας 3.1 Χαρακτηριστικά Thread

Πίνακας 5.1 Τυπικές συνθήκες λειτουργίας

## **Κατάλογος Εικόνων**

Εικόνα 1.1 Διάγραμμα Ροής Αρχιτεκτονικής IoT συσκευών

Εικόνα 1.2 Τοπολογία αστέρα (Star)

Εικόνα 1.3 Τοπολογία Πλέγματος (Mesh)

Εικόνα 1.4 Τοπολογία Δένδρου (Tree)

Εικόνα 1.5 Τοπολογία Peer to Peer

Εικόνα 2.1 Αρχιτεκτονική συσκευής LR-WPAN

Εικόνα 2.2 Μοντέλο OSI και τύποι δεδομένων

Εικόνα 2.3 Η μετάδοση της πληροφορίας του Data link layer στο πλαίσιο του Physical Layer

Εικόνα 3.1 Επισκόπηση προδιαγραφών Thread

Εικόνα 3.2 Δομή Zigbee

Εικόνα 3.3 Devices

Εικόνα 3.4 Λειτουργίες primitive

Εικόνα 3.5 SAPs

Εικόνα 3.6 Ενθυλάκωση

Εικόνα 4.1 Δομή 6LoWPAN

Εικόνα 4.2 Δομή Overhead

Εικόνα 4.3 Overhead κατακερματισμού

Εικόνα 4.4 Overhead διευθυνσιοδότησης

Εικόνα 4.5 HC1 Encoding

Εικόνα 4.6 Κωδικοποίηση HC2

Εικόνα 4.7 Compression Examples

Εικόνα 4.8 Βελτιστοποιημένη κωδικοποίηση

Εικόνα 4.9: Examples of improved compression

Εικόνα 4.10: Δρομολόγηση στο επίπεδο δικτύου και ζεύξης

Εικόνα 5.1 : Περιγραφή Tmote (Μπροστά)

Εικόνα 5.2 : Περιγραφή Tmote (Πίσω)

Εικόνα 5.3 : Block diagram Tmote Sky

Εικόνα 5.4 : Z1 Mote

Εικόνα 5.5 : Z1 Mote – Block Diagram

Εικόνα 5.6 : Αρχικοποιημένα Plugins

Εικόνα 6.1 : Λήψη αρχείου – Επιλογή έκδοσης Contiki 2.7

Εικόνα 6.2 : Είσοδος στο VMWare Player – Επιλογή Contiki 2.7

Εικόνα 6.3 : Κωδικός πρόσβασης εισόδου

Εικόνα 6.4 : Εντολή ανοίγματος εφαρμογής μέσω terminal

Εικόνα 6.5 : Περιβάλλον υποδοχής νέου Simulation

Εικόνα 6.6 : Είσοδος στο φάκελο Home – Επιλογή Contiki

Εικόνα 6.7 : Δημιουργία φακέλου app\_button

Εικόνα 6.8 : Δημιουργία text αρχείου press\_button

Εικόνα 6.9 : Μέσα στο αρχείο button-sensor.h

Εικόνα 6.10 : Μέσα στο αρχείο sensors.h

Εικόνα 6.11 : Κείμενο κώδικα press\_button

Εικόνα 6.12 : Πρότυπο Makefile

Εικόνα 6.13 : Επεξεργασμένο Makefile

Εικόνα 6.14 : Εντολή make στο terminal

Εικόνα 6.15 : Δημιουργία απαραίτητων αρχείων

Εικόνα 6.16: Δημιουργία προσομοίωσης

Εικόνα 6.17: Επιλογές καρτέλας View

Εικόνα 6.18: Επιλογή Sky Mote

Εικόνα 6.19: Επιλογή app\_button

Εικόνα 6.20: Επιτυχές compil – Create

Εικόνα 6.21: Επιλογή Mote Interface Viewer

Εικόνα 6.22: Επιλογή Button

Εικόνα 6.23: Εμφάνιση παραθύρου Button

Εικόνα 6.24: Εμφάνιση παραθύρου Sky LED

Εικόνα 6.25: Εντολή Start

Εικόνα 6.26: Αναμμένο κόκκινο LED

Εικόνα 6.27: Αναμμένο κόκκινο LED – Παράθυρο Network

Εικόνα 6.28: Παράθυρο Mote output

Εικόνα 6.29: Δημιουργία νέας προσομοίωσης

Εικόνα 6.30: Επιλογή Contiki-2.7

Εικόνα 6.31: Επιλογή ipv6

Εικόνα 6.32: Επιλογή border-router.c

Εικόνα 6.33: Επιλογή sky-websense.c

Εικόνα 6.34: Μη ανταλλαγή δεδομένων

Εικόνα 6.35: Εντολή terminal – Έναρξη ανταλλαγής δεδομένων

Εικόνα 6.36: Έναρξη ανταλλαγής δεδομένων – Serial Socket (Server)

Εικόνα 6.37: Διευθύνσεις μέσω terminal

Εικόνα 6.38: Διευθύνσεις motes μέσω Mozilla

Εικόνα 6.39: Αποτελέσματα ενός εκ των motes

Εικόνα 6.40: 1<sup>η</sup> επεξεργασία Sky-websense

Εικόνα 6.41: 2<sup>η</sup> επεξεργασία Sky-websense

Εικόνα 6.42: 3<sup>η</sup> επεξεργασία Sky-websense

Εικόνα 6.43: 4<sup>η</sup> επεξεργασία Sky-websense

Εικόνα 6.44: 1<sup>η</sup> επεξεργασία Sht11.c

Εικόνα 6.45: 2<sup>η</sup> επεξεργασία Sht11.c

Εικόνα 6.46: 1<sup>η</sup> επεξεργασία Sht11.h

Εικόνα 6.47: 1<sup>η</sup> επεξεργασία Sht11-sensor.c

Εικόνα 6.48: 1<sup>η</sup> επεξεργασία Sht11-sensor.h

Εικόνα 6.49: Τελικό αποτέλεσμα

Εικόνα 6.50: Δημιουργία Simulation 3

Εικόνα 6.51: Επιλογές View

Εικόνα 6.52: 10m background grid

Εικόνα 6.53: Radio Environment (UDGM)

Εικόνα 6.54: Εμφάνιση παραθύρων collect view

Εικόνα 6.55: Εμφάνιση bytes στο collect view

Εικόνα 6.56: Average Power Consumption

Εικόνα 6.57: Battery Voltage

Εικόνα 6.58: Light

Εικόνα 6.59: Node info

Εικόνα 6.60: Temperature

Εικόνα 6.61: Sensor Map

Εικόνα 6.62: Average Radio Duty Cycle – Properties Γραφημάτων

Εικόνα 6.63: Καθορισμός Transmission ranges

Εικόνα 6.64: Κάτοψη Γραφείου

Εικόνα 6.65: Κάτοψη Γραφείου – Πεδίο Radio Environment (UDGM)

## **Αλφαβητικό Ευρετήριο**

**“IoT”** - “Internet of Things”

**“RFID”** - “Radio-frequency identification”

**“IEEE”** - “Institute of Electrical and Electronics Engineers”

**“GSM”** - “Global System for Mobile Communications”

**“DTLS”** - “Datagram Transport Layer Security”

**“IETF”** - “Internet Engineering Task Force”

**“IID”** – “Interface Identifier”

**“6LoWPAN”** – “IPv6 over Low-Power Wireless Personal Area Networks”

**“RPL”** - “Routing protocol for low power and lossy networks”

**“WPANs”** - “wireless personal area networks”

**“LR-WPANs”** - “Low-Rate Wireless Personal Area Networks”

**“PAN”** - “Personal area network”

## ΕΙΣΑΓΩΓΗ

Η επικοινωνία και η ανταλλαγή δεδομένων, αποτελούν δύο από τις βασικότερες ανάγκες, τόσο για διαβίωση όσο και για τη βελτιστοποίηση της ποιότητας ζωής. Οι ανάγκες αυτές είναι άμεσα συνυφασμένες με τις δυνατότητες και την εξέλιξη της τεχνολογίας, η οποία υπεισέρχεται με διαφορετικούς τρόπους στις δραστηριότητες της καθημερινότητάς.

Ένα πραγματικό επίτευγμα της τεχνολογίας ωστόσο, μπορεί να θεωρηθεί το γεγονός πως υπάρχει πλέον η δυνατότητα απλές συσκευές οι οποίες χρησιμοποιούνται σε σπίτια ή εργασιακούς χώρους, να επικοινωνούν μεταξύ τους και να δίνεται η δυνατότητα μέσω αυτών απομακρυσμένα να συλλέγονται δεδομένα και πληροφορίες. Ένα παράδειγμα αποτελεί το άναμμα της λάμπας σε ένα δωμάτιο από ένα κινητό, επίσης η λήψη δεδομένων της θερμοκρασίας του περιβάλλοντος από έναν μικροελεγκτή και την αποτύπωση της στην οθόνη ενός υπολογιστή ακόμα και η απομακρυσμένη ενεργοποίηση του κλιματιστικού στο γραφείο.

Η ανάγκη για μεγαλύτερη εξοικονόμηση ενέργειας αλλά και την μεμονωμένη ανάπτυξη αυτών των Ασυρμάτων Τοπικών Δικτύων (Wireless Personal Area Networks, WPAN), οδήγησε στην ανάπτυξη του, ευρέως πλέον διαδεδομένου, προτύπου 802.15.4. Παράγωγο της αρχιτεκτονικής αυτής αποτελεί το 6LoWPAN όπου αποτελεί το αντικείμενο της διπλωματικής αυτής.

Σε επόμενο κεφάλαιο πραγματοποιείται μια προσομοίωση δικτύου WPAN βασισμένη στο πρωτόκολλο 6LoWPAN. Το λογισμικό που θα χρησιμοποιηθεί είναι το Contiki COOJA, όπου είναι ιδανικό για την προσομοίωση ασύρματων επικοινωνιών βασισμένα στο πρότυπο 80215.4. Για την καλύτερη κατανόηση της διαδικασίας αυτής, έχουν επιλεγεί τρία σενάρια στα οποία περιγράφονται σε τρεις ξεχωριστές καταστάσεις. Η υλοποίηση των σεναρίων αυτών επιτυγχάνεται μέσω παραλλαγών των υφιστάμενων παραδειγμάτων όπου βρίσκονται ήδη εγκατεστημένα στις βιβλιοθήκες της εφαρμογής Contiki COOJA.

Σκοπός είναι η παρουσίαση του τρόπου επικοινωνίας μεταξύ των μικροελεγκτών, οι οποίοι κατά τη διάρκεια της εκάστοτε προσομοίωσης επιτελούν διαφορετικές λειτουργίες, καθώς η επικοινωνία τους στηρίζεται στο πρωτόκολλο 6LoWPAN. Η συλλογή πληροφοριών κατά την ώρα της επικοινωνίας, αλλά και η ανάδειξη του πλήθους των παραδειγμάτων που μπορούν να προκύψουν με τις σωστές αλλαγές στο ήδη υπάρχον υλικό, αποτελούν επίσης στόχο της εργασίας.

Με την υλοποίηση αυτή, έγινε προσπάθεια για την βελτίωση των αποτελεσμάτων που λαμβάνονται σε σχέση με εκείνα που θα αποτυπωνόντουσαν κατά την εκτέλεση του προγράμματος. Επίσης έγινε μια εκτίμηση της λειτουργίας των μικροελεγκτών και των ενδεχόμενων απωλειών στην επικοινωνία μεταξύ τους, μέσω της εισαγωγή μιας κάτοψης γραφείου μέσω της εφαρμογής AutoCad. Η χρήση του σχεδίου δείχνει την ύπαρξη φυσικών εμποδίων και το πως αυτά επηρεάζουν την επικοινωνία μεταξύ τους, χωρίς την ανάγκη χρησιμοποίησης επιπλέον συσκευών.



- Στο 1<sup>ο</sup> κεφάλαιο αναπτύσσεται η έννοια του Διαδικτύου των Πραγμάτων. Πραγματοποιείται επίσης μια εμπειριστατωμένη ανάλυση στα πλαίσια της αρχιτεκτονικής, της δομής και της λειτουργίας.
- Στο 2<sup>ο</sup> κεφάλαιο παρουσιάζεται το πρωτόκολλο 802.15.4, μέσα από μια βιβλιογραφική ανάλυση. Παρατίθενται επίσης πληροφορίες αναφορικά με τις ιδιότητες και τις εφαρμογές του.
- Στο 3<sup>ο</sup> κεφάλαιο παρατίθενται πρωτόκολλα τα οποία είναι βασισμένα στο πρότυπο 802.15.4. Αναγράφονται πληροφορίες αναφορικά τα χαρακτηριστικά και τις εφαρμογές τους.
- Στο 4<sup>ο</sup> κεφάλαιο γίνεται μια εκτενής αναφορά στη χρήση και τα χαρακτηριστικά του πρωτοκόλλου 6LoWPAN.
- Στο 5<sup>ο</sup> κεφάλαιο περιγράφονται οι δυνατότητες και οι λειτουργίες του προγράμματος Contiki COOJA.
- Στο 6<sup>ο</sup> κεφάλαιο παρατίθενται τρία διαφορετικά σενάρια προσομοίωσης στο Contiki Cooja με την αντίστοιχη περιγραφή και παράθεση των αντίστοιχων εικόνων.

## ΚΕΦΑΛΑΙΟ 1ο : IoT - WSN

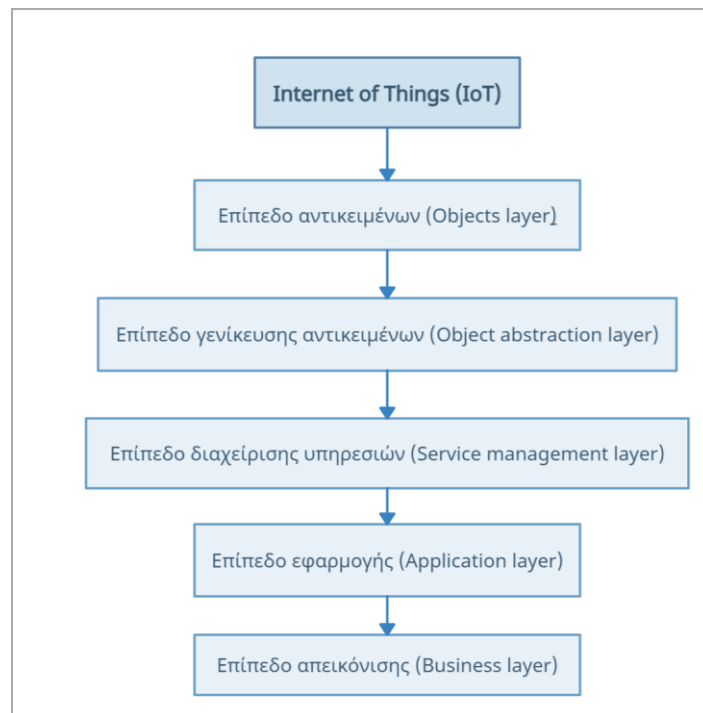
Στο κεφάλαιο αυτό παρουσιάζονται και περιγράφονται βασικές έννοιες και ορολογίες σχετικά με το Διαδίκτυο των Πραγμάτων και τα Ασύρματα Δίκτυα Αισθητήρων.

### 1.1 Internet of Things (IoT)

Το Διαδίκτυο των Πραγμάτων (Internet of Things, IoT) είναι ένα δίκτυο το οποίο αποτελείται από συσκευές (πράγματα, Things). Σκοπός των συσκευών που απαρτίζουν το σύνολο αυτό, είναι η συλλογή δεδομένων, η επεξεργασία και η εξασφάλιση της επικοινωνίας μεταξύ τους. Η δυνατότητα αυτή, απορρέει στην ραγδαία ανάπτυξη της τεχνολογίας και πιο συγκεκριμένα στον νόμο του Moore[1]. Μέσω της εφαρμογής του, μπορούν να παραχθούν συσκευές, οι οποίες να είναι μικρές σε μέγεθος όπως επίσης και με χαμηλές ενεργειακές απαιτήσεις. Η διαδικασία αυτή της ευρύτερης επικοινωνίας, της συλλογής δεδομένων - διαλογής είναι ήδη γνωστή με ήδη γνωστά πρωτόκολλα και υπηρεσίες όπως το Bluetooth Low Energy (BLE), το ασύρματο δίκτυό (Wi-Fi) και το δίκτυο κινητής τηλεφωνίας (LTE, 5G).[1-4]

#### 1.1.1 Η Δομή και η Λειτουργία των συσκευών

Η δομή του Διαδικτύου των Πραγμάτων αποτελείται από πέντε επίπεδα, όπως φαίνεται στην εικόνα 1.1 και παρατίθενται παρακάτω:



Εικόνα 1.1 : Διάγραμμα Ροής Αρχιτεκτονικής IoT συσκευών

A. Επίπεδο Αντικειμένων –Object Layers:

Το επίπεδο αυτό θα μπορούσε να χαρακτηριστεί ως η ομάδα εκείνη της οποίας αρμοδιότητα είναι να συλλέγει πληροφορίες από το περιβάλλον γύρω του και να τις παρέχει στο σύστημα IoT. Για τη λειτουργία αυτή χρησιμοποιούνται ειδικές συσκευές, όπως το Raspberry ή το Arduino, τα οποία έχουν τη δυνατότητα να πραγματοποιήσουν, με κατάλληλη υλοποίηση, διάφορων ειδών μετρήσεις όπως το πόσο ζεστό είναι ένα περιβάλλον ή πόσο φωτεινό είναι. Αυτές οι πληροφορίες που συλλέγονται είναι σημαντικό να αποστέλλονται με ασφάλεια στο επόμενο τμήμα του συστήματος.[1-3,4]

B. Επίπεδο Γενίκευσης Αντικειμένων – Object abstraction layer:

Το συγκεκριμένο επίπεδο είναι υπεύθυνο για την αποστολή μηνυμάτων από το ένα μέρος ενός συστήματος στο άλλο. Μπορεί να χρησιμοποιήσει μέσα όπως το Διαδίκτυο και τεχνολογίες όπως οι υπηρεσίες κινητής τηλεφωνίας (LTE, Long Term Evolution), το Wi-Fi και το BLE, προκειμένου να ανταλλάξει τα μηνύματα αυτά και να επιτελεστούν οι διαδικασίες που απαιτούνται.[2-3,6]

C. Επίπεδο Διαχείρισης Υπηρεσιών – Service management layer:

Στο επίπεδο αυτό, γίνεται δρομολόγηση των πακέτων από το αποστολέα στον αποδέκτη. Πιο συγκεκριμένα, γίνεται συσχέτιση μεταξύ των διαδικασιών εκείνων όπου αιτείται τη διαδικασία με βάση χρήσιμων πληροφοριών όπως το όνομα και η διεύθυνση δικτύου. Κατά τη διεργασία αυτή η αρχιτεκτονική του περιβάλλοντος αυτού μπορεί να ξεχωρίσει τις όποιες διαφορές που βρίσκονται ενδιάμεσα των πληροφοριών από διαφορετικές πηγές.[1,3,5]

D. Επίπεδο Εφαρμογής – Application Layer :

Στο επίπεδο εφαρμογής, παρουσιάζονται οι διαθέσιμες επιλογές που παρέχονται στον χρήστη, από το εκάστοτε IoT. Εδώ πραγματοποιείται η επεξεργασία των δεδομένων που έχουν ληφθεί από το επίπεδο του διαδικτύου. Το επίπεδο το οποίο βρίσκεται πιο κοντά από οποιοδήποτε άλλο, στην ανθρώπινη αντίληψη και συμπεριφορά. Η κάθε περίπτωση διαφέρει, αφού θα πρέπει κάθε φορά να είναι εναρμονισμένη με τις αντίστοιχες ανάγκες της εφαρμογής. Οι ανάγκες αυτές ποικίλλουν καθώς χρήζει διαφορετικών ενεργειών μια υποδομή για ένα έξυπνο σπίτι, μια έξυπνη πόλη, ένα έξυπνο αγρόκτημα.[1,3-4]

### E. Επίπεδο Απεικόνισης – Business layer :

Σε αυτό το επίπεδο επιτελείται η συνολική εποπτεία των υπηρεσιών και των διαδικασιών εντός του IoT περιβάλλοντος. Αυτό το επίπεδο είναι υπεύθυνο για την σχεδίαση, ανάλυση, αξιολόγηση, έλεγχο και εφαρμογή των προϋποθέσεων λειτουργίας επίσης IoT συστήματος. Ακόμα σε αυτό το επίπεδο γίνεται η απεικόνιση των δεδομένων σε σχεδιαγράμματα, μοντέλα και διάφορα διαγράμματα.

Το παρόν επίπεδο επιτελεί τη λειτουργία του καθολικού ελέγχου των διαδικασιών και των λειτουργιών μέσα σε ένα IoT δίκτυο. Έχει επίσης την αρμοδιότητα αναφορικά με το:

- Design - Σχεδιασμός
- Analysis - Ανάλυση
- Evaluation - Αξιολόγηση
- Control - Έλεγχος
- Implementation - Εφαρμογή

Επιπροσθέτως εδώ οι πληροφορίες απεικονίζονται με τη μορφή μοντέλων και σχεδιαγραμμάτων.[2,3,5]

## 1.2 Ασύρματα Δίκτυα Αισθητήρων - Wireless Sensors Network

Ασύρματο Δίκτυο Αισθητήρων (Wireless Network Sensors, WSN) μπορεί να χαρακτηριστεί ως μια ομάδα συσκευών που μπορούν να επικοινωνούν μεταξύ τους χωρίς τη χρήση καλωδίων. Οι συσκευές αυτές, ονομάζονται κόμβοι, μπορούν να πραγματοποιήσουν διαφόρων ειδών μετρήσεις. Όπως το πόσο ζεστό ή κρύο είναι ένα δωμάτιο ή το πόσο φως εισέρχεται σε αυτό. Μπορούν επίσης να ανταλλάξουν πληροφορίες με άλλες κοντινές συσκευές χρησιμοποιώντας κεραίες.[7,8]

### 1.2.1 Τα σημεία εφαρμογής τους

Καθώς η τεχνολογία βελτιώνεται και αναπτύσσεται, οι αισθητήρες χρησιμοποιούνται για τη συλλογή πληροφοριών του περιβάλλοντος, ενώ παράλληλα γίνονται όλο πιο προηγμένοι ενώ ταυτόχρονα και πιο προσιτοί οικονομικά. Ορισμένες από τις χρήσεις τους αναφορικά, με το περιβάλλον είναι οι κάτωθι:

- Γεωργία: έλεγχος των χαρακτηριστικών του χώματος, προκειμένου να εξακριβωθεί η ποιότητα του έτσι ώστε να διασφαλιστεί η καλύτερη δυνατή συγκομιδή.
- Νοσηλευτική: πρόληψη μέσω μετρήσεων των ζωτικών ενδείξεων του ανθρώπινου οργανισμού, με σκοπό την αποφυγή του κορεσμού.

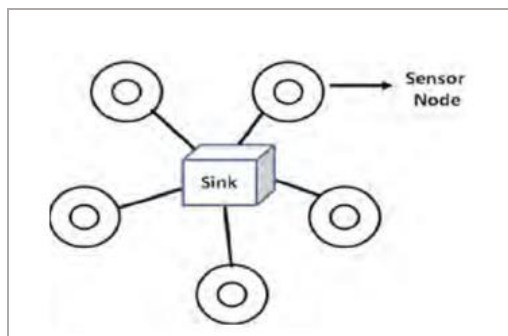
- Οικία: ενοποίηση και επικοινωνία μεταξύ των απλών καθημερινών οικιακών συσκευών με σκοπό την απλούστευση και τη βελτίωση του επιπέδου διαβίωσης.[1,7]

## 1.2.2 Τοπολογίες

Ανάλογα με τον τρόπο συνδεσμολογίας του κάθε δικτύου, οι πιο γνωστοί τρόποι που χρησιμοποιούνται είναι οι παρακάτω:

### 1.2.2.1 Τοπολογία Αστέρα - Star Topology :

Σε ένα δίκτυο με συνδεδεμένο με αυτήν την τοπολογία, υπάρχουν διαφορετικές συσκευές που ονομάζονται κόμβοι οι οποίοι δεν επικοινωνούν απευθείας μεταξύ τους. Αντίθετα, στέλνουν τα μηνύματά τους μέσω μιας κεντρικής συσκευής, δρομολογητής (πύλη, Gateway). Ο κεντρικός σταθμός επιτελεί τη λειτουργία της υποβοήθησης της μεταφοράς των μηνυμάτων στον προορισμό τους. Οι κόμβοι είναι οι πελάτες (clients) και ο σταθμός βάσης ο εξυπηρετητής Αυτή η ρύθμιση καταναλώνει λιγότερη ενέργεια, αλλά μερικές φορές τα μηνύματα δεν φτάνουν πάντα στον τελικό τους αποδέκτη επειδή ο σταθμός βάσης μπορεί να μιλήσει μόνο σε κοντινές συσκευές. Στην Εικόνα 1.2 παρουσιάζεται ένα σχηματικό παράδειγμα της τοπολογίας αυτής.[6,7]

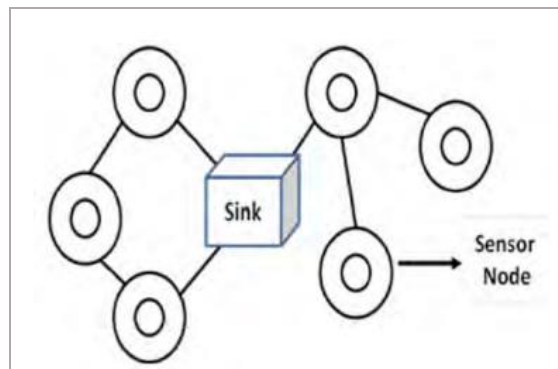


**Εικόνα 1.2.: Συνδεσμολογία αισθητήρων σε τοπολογία αστέρα (Star) [6]**

### 1.2.2.2 Τοπολογία Πλέγματος - Mesh Topology :

Στην τοπολογία αυτή, τα μηνύματα δύναται να διέλθουν μέσω πολλαπλών διαδρομών προκειμένου φτάσουν στον αποδέκτη τους. Οι κόμβοι σε αυτού του τύπου την συνδεσμολογία, μπορούν να ανταλλάσσουν πληροφορίες μεταξύ τους όπως επίσης και με έναν κεντρικό σταθμό. Έτσι, στην περίπτωση που χρειάζεται να επικοινωνήσουν δύο κόμβοι μεταξύ τους και δεν μπορούν να το επιτύχουν με

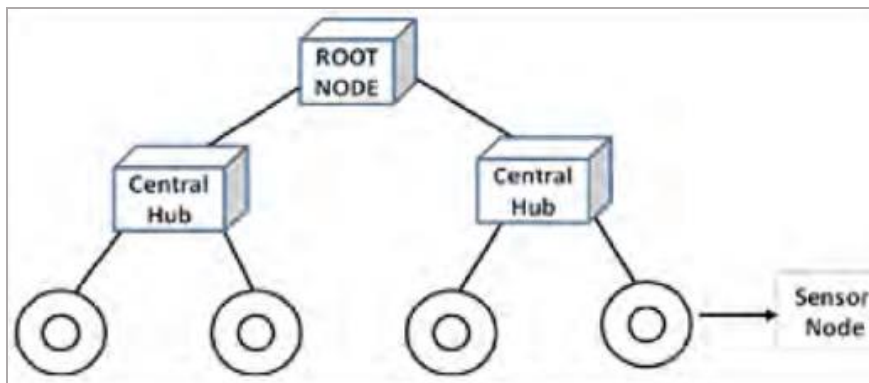
κάποια απευθείας σύνδεση, υπάρχει η δυνατότητα να γίνει χρήση ενός τρίτου κόμβου να μεσολαβήσει έτσι ώστε να φτάσει το μήνυμα στον προορισμό του. Αυτή η διαδικασία σίγουρα καθιστά το σύστημα πολύ πιο αξιόπιστο, δεδομένου ότι υπάρχει εναλλακτική διαδρομή στην μετάδοση της πληροφορίας, ωστόσο η κατανάλωση περισσότερης ενέργειας είναι μεγαλύτερη σε σύγκριση με μια τοπολογία αστέρα. Η εικόνα 1.3 δείχνει ένα δίκτυο αισθητήρων με βάση την τοπολογία αυτή. [2,7].



**Εικόνα 1.3: Συνδεσμολογία αισθητήρων σε τοπολογία Πλέγματος (Mesh) [6]**

#### 1.2.2.3 Τοπολογία Δέντρου - Tree Topology :

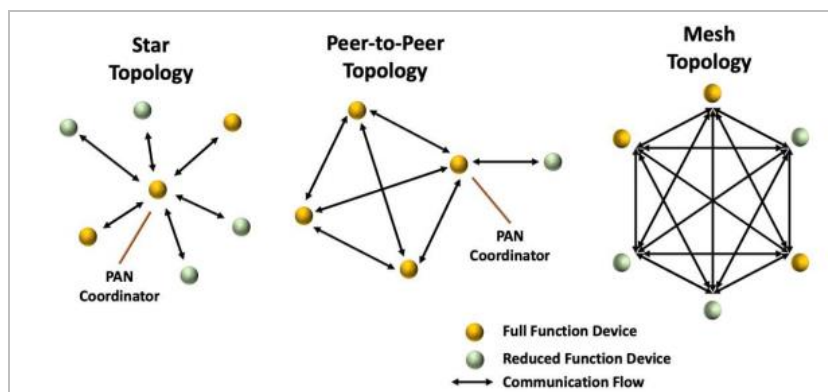
Ένα δίκτυο το οποίο είναι δομημένο με την τοπολογία δέντρου έχει την παρακάτω μορφή. Απαρτίζεται από έναν κύριο σταθμό που λειτουργεί ως Κεντρικός κόμβος (Root Node) ο οποίος είναι υπεύθυνος για όλη την επικοινωνία. Έπειτα συναντάται ένας ή περισσότεροι σταθμοί επικοινωνίας όπου ονομάζεται κόμβος/οι (Hub(s)). Σε κάθε έναν από αυτούς διαμορφώνεται μια συνδεσμολογία στη σε τοπολογία αστέρα. Η τοπολογία δέντρου αποτελεί ένα συνδυασμό δικτύων σε συνδεσμολογία αστέρα. Στην τοπολογία τύπου δέντρου, οι συσκευές στα άκρα του δικτύου δεν χρειάζεται να έχουν μεγάλη επεξεργαστική ισχύ εν αντιθέσει με εκείνες που βρίσκονται πιο κοντά στον κεντρικό κόμβο. Οι ισχυρότερες συσκευές συμβάλλουν στη μετάδοση μηνυμάτων μεταξύ των υπολοίπων και λειτουργούν σαν πύλες. Η ισχύς αυτή είτε υψηλότερη είτε χαμηλότερη, αντιστοιχεί και σε αντίστοιχα μεγέθη κατανάλωσης ενέργειας. Η εικόνα 1.4 δείχνει ένα σχηματικό παράδειγμα της τοπολογίας αυτής[5,7].



Εικόνα 1.4: Τοπολογία Δένδρου (Tree) [6]

#### 1.2.2.4 Τοπολογία από σημείο σε σημείο (Peer to Peer, P2P)

Με αυτό τον τρόπο σύνδεσης δίνεται η δυνατότητα στη συσκευή να επικοινωνήσει απευθείας με όποια συσκευή βρίσκεται στο περιβάλλον της. Μια συσκευή μπορεί να λειτουργήσει ως συντονιστής (PAN Coordinator), όταν χρονικά έρχεται πρώτη σε επαφή με το εκάστοτε κανάλι. Αντίστοιχες διαμορφώσεις δικτύου μπορούν να συνεχίσουν να δημιουργούνται με την αντίστοιχη τοπολογία με τους ίδιους περιορισμούς. Η εικόνα 1.5 απεικονίζει όλες τις τεχνολογίες που έχουν αναφερθεί [9,12].



Εικόνα 1.5: Τοπολογία Peer to Peer [12]

## ΚΕΦΑΛΑΙΟ 2ο : Πρωτόκολλο 802.15.4

Το IEEE 802.15, είναι ένα πρότυπο οποίο έχει οριστεί για την εφαρμογή κανόνων σχετικά με το πώς οι συσκευές μπορούν να επικοινωνούν ασύρματα σε μια μικρή περιοχή. Παρουσιάστηκε το 2003 ενώ οι βασικές εκδόσεις του προτύπου αυτού έχουν οριστεί το 2006 και το 2011. Η επικοινωνία αυτή επιτυγχάνεται μέσω των Ασύρματων Δικτύων Μικρής Εμβέλειας (Wireless Personal Area Networks, WPAN). Μέσω αυτού προτύπου αυτού, μια προδιαγραφή που ορίζεται, αφορά τις ταχύτητες των ασύρματων αυτών δικτύων. όπου κυμαίνονται από 20 kbps έως και 250 kbps. Επίσης, μέσω του προτύπου αυτού ορίζεται ο τρόπος επικοινωνίας των συσκευών στο Φυσικό Επίπεδο και στο Επικοινωνίας Δεδομένων.

### 2.1 Γενική Περιγραφή

Η προδιαγραφή IEEE 802.15.4 περιγράφει το φυσικό επίπεδο (Physical Layer), όπως και το επίπεδο Σύνδεσης Δεδομένων (Data Link) το οποίο περιγράφει την λειτουργία της κάρτας δικτύου (Medium Access Control, MAC). Η διεργασία αυτή πραγματοποιείται για τα Χαμηλής Ενέργειας Ασύρματα Δίκτυα Μικρής Εμβέλειας (Low Range Wireless Area Networks, LR-WAN). Στο πρότυπο αυτό βασίζονται πρωτόκολλα όπως είναι το ZigBee, το MiWi και το ISA100.11a. Καθένα από αυτά, αποτελεί μια διαφορετική υλοποίηση του προτύπου αυτού, με την διαφοροποίηση τους να βρίσκεται στα ανώτερα επίπεδα του OSI (Open System Interconnect).

Το LR-WPAN είναι μια τοπολογία ασύρματου δικτύου η οποία σαν σκοπό έχει την εξασφάλιση της επικοινωνίας των συσκευών της οι οποίες βρίσκονται σε κοντινή απόσταση μεταξύ τους. Οι κύριοι στόχοι του LR-WPAN είναι η ευκολία στη ρύθμιση συσκευών, η αξιόπιστη μεταφορά δεδομένων, η διατήρηση του κόστους σε χαμηλά επίπεδα και η διασφάλιση πως οι συσκευές μπορούν να διαρκέσουν πολύ χρόνο με μία μόνο φόρτιση μπαταρίας. Τα χαρακτηριστικά του προτύπου:

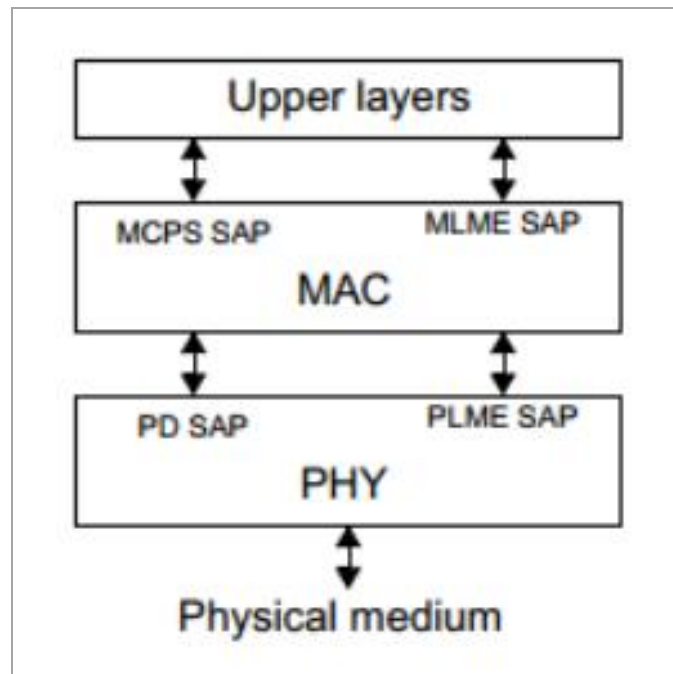
- Τοπολογία Αστέρα.
- Τοπολογία Πλέγματος
- Μοναδική 16-bit ή 64-bit address.
- Προαιρετική κατανομή των “guaranteed time slots”.
- Τεχνολογία διασφάλισης επικοινωνίας Carrier sense multiple access with collision avoidance (CSMA-CA)
- Πρόσβαση καναλιού ALOHA, μέσω του οποίου τα δεδομένα μεταφέρονται ταυτόχρονα σε ένα δίκτυο, όταν ένα πακέτο μεταφοράς είναι διαθέσιμο για μετάδοση. [14]
- Πρωτόκολλο σχετικά με την αξιοπιστία της μεταφοράς
- Μικρή κατανάλωση ενέργειας



- Εντοπισμός της ενέργειας (Energy Detection)
- Ένδειξη της ποιότητας Σύνδεσης (LQI). Στο πρότυπο αυτό ορίζονται τα πολλαπλά “φυσικά επίπεδα” τα οποία λειτουργούν σε ένα μεγάλο αριθμό από ζώνες συχνοτήτων [10,16].

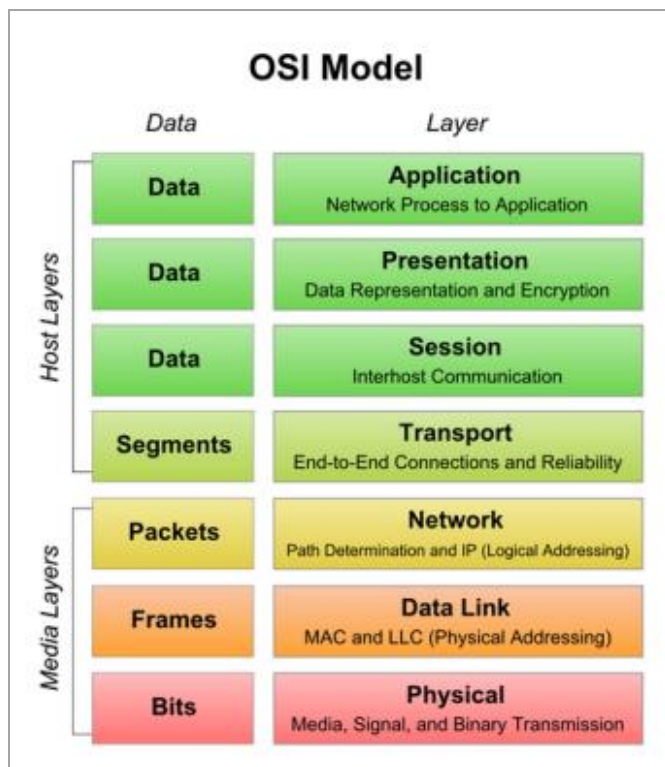
## 2.2 IEEE 802.15.4 και μοντέλο OSI

Όπως έχει ήδη αναφερθεί, τα πρώτα δυο επίπεδα OSI (το Φυσικής Διεπαφής και το Επικοινωνίας Δεδομένων) καθορίζονται από το πρότυπο 802.15.4. Τα επίπεδα αυτά έχουν καθοριστεί να εκτελούν ορισμένες διεργασίες σχετικά με τις εσωτερικές τους λειτουργίες το καθένα ξεχωριστά αλλά και μεταξύ τους. Η Εικόνα 2.1 δείχνει ένα σχηματικό παράδειγμα της λειτουργίας αυτής.

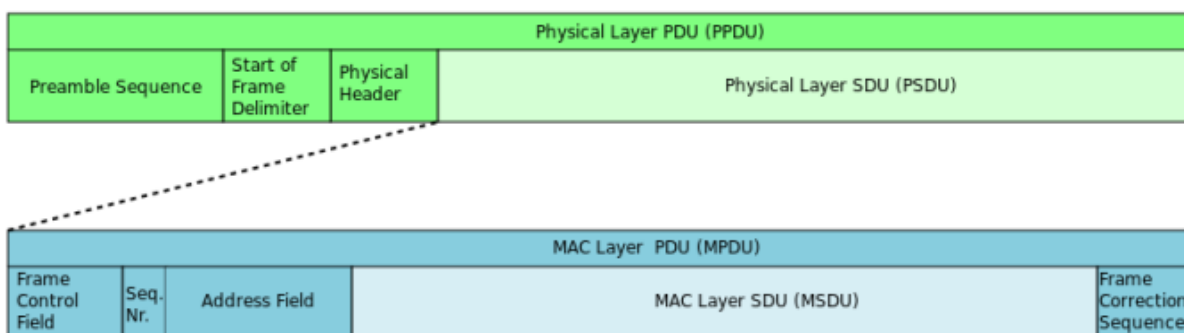


Εικόνα 2.1: Αρχιτεκτονική συσκευής LR-WPAN [5]

Το μοντέλο OSI-7 καθορίζεται από επτά επίπεδα επικοινωνίας, τα οποία, από κάτω προς τα πάνω, ξεκινούν από το επίπεδο της Φυσικής Επαφής και καταλήγουν σε αυτό της Εφαρμογής. Η Εικόνα 2.2 δείχνει μια γραφική απεικόνιση, πως γίνεται η μετάβαση της πληροφορίας από την προηγούμενη στην επόμενη βαθμίδα, ενώ στην εικόνα 2.3 παρουσιάζεται το πως αποτυπώνεται κάθε επίπεδο μέσα σε ένα πακέτο που μεταδίδεται[2,13].



Εικόνα 2.2: Μοντέλο OSI και τύποι δεδομένων [5]



Εικόνα 2.3: Μετάδοση πληροφορίας του Data link layer στο πλαίσιο του Physical Layer [17]

### 2.3 Ανάλυση επιπέδου Physical

Το φυσικό επίπεδο (Physical Layer) είναι αυτό, το οποίο είναι υπεύθυνο για τον τρόπο επικοινωνίας (ενσύρματος/ασύρματο). Αποτελεί τον σύνδεσμο επικοινωνίας μεταξύ συσκευών στο ασύρματο δίκτυο. Στο φυσικό επίπεδο συναντώνται δύο είδη υπηρεσιών: Η υπηρεσία δεδομένων και η υπηρεσία διαχείρισης PHY. Η δεύτερη είναι υπεύθυνη για τη διασύνδεση με το PLME ( Physical Layer Management Entity). Η υπηρεσία δεδομένων παρέχει τη δυνατότητα για μετάδοση και υποδοχή των Μονάδων Πρωτοκόλλου Δεδομένων (PHY Protocol Data Units), με χρήση του φυσικού καναλιού. Χαρακτηριστικά επίσης του φυσικού επιπέδου είναι τα ακόλουθα:

- On / Off ράδιο-πομποδέκτη.
- Ένδειξη Ποιότητας Σύνδεσης (Link Quality Indication -LQI).
- Επιλογή καναλιού.
- Εντοπισμός ενέργειας (Energy Detection - ED).
- Αξιολόγηση Καναλιού - “CCA”.
- Αποστολή και λήψη πακέτων [10,11].

### 2.3.1 Εύρος Διακύμανσης Συχνοτήτων Δεδομένων

Το 802.15.4 δρομολογεί λειτουργίες του φυσικού επιπέδου, οι οποίες βασίζονται σε ζώνες διαφορετικών συχνοτήτων και ταχυτήτων. Διαφορετικοί είναι επίσης οι τρόποι με τους οποίους επιτελείται η διαμόρφωση και η αποδιαμόρφωση. Η ζώνη των 2,4 GHz λόγω της διαθεσιμότητάς της, δύναται να χρησιμοποιηθεί σε όλες τις ηπείρους ανά τον κόσμο για ασύρματη επικοινωνία ανάμεσα σε συσκευές με μικρές αποστάσεις. Η περιοχή των 2,4 GHz φέρει την ονομασία της Βιομηχανικής Επιστημονικής και Ιατρικής Ζώνης (ISM – Industrial Scientific and Medical Band),. Αναφορικά με το εύρος ζώνης, διατίθεται ένα πλήθος διαφορετικών τεχνολογιών του φυσικού επιπέδου των δικτύων αισθητήρων. Οι τεχνολογίες ευρέως φάσματος χαρακτηρίζονται ως οι καταλληλότερες για τα δίκτυα αισθητήρων, καθώς μειώνουν την κατανάλωση ενέργειας, ενώ παράλληλα μειώνουν τα ήδη υψηλά ποσοστά σφαλμάτων. Στον πίνακα 2.1, γίνεται η παρουσίαση ενός πομποδέκτη,ο οποίος χρησιμοποιεί διαμόρφωση OQPSK (Offset Quadrature Phase Shift Keying) στην 2,45GHz ζώνη συχνοτήτων και με ρυθμό διάδοσης 250kbit/s. Αποτυπώνονται επίσης τα διάφορα PHYs και οι ζώνες συχνοτήτων στα οποία μπορεί να λειτουργήσει μία συσκευή [6,11].

PHY (MHz)	Ζώνη Συχνοτήτων (MHz)	Παράμετροι διάδοσης		Παράμετροι δεδομένων		
		Chip rate (kchip/s)	Διαμόρφωση	Bit rate (kb/s)	Symbol rate (ksymbol/s)	Σύμβολα
780	779-787	1000	O-QPSK	250	62.5	16-ary orthogonal
780	779-787	1000	MPSK	250	62.5	16-ary orthogonal
868/915	868-868.6	300	BPSK	20	20	Binary
	902-928	600	BPSK	40	40	Binary
868/915 (προαιρετικό)	868-868.6	400	ASK	250	12.5	20-bit PSSS
	902-928	1600	ASK	250	50	5-bit PSSS
868/915 (προαιρετικό)	868-868.6	400	O-QPSK	100	25	16-ary orthogonal
	902-928	1000	O-QPSK	250	62.5	16-ary orthogonal
950	950-956	-	GFSK	100	100	Binary
950	950-956	300	BPSK	20	20	Binary
2450 DSSS	2400-2483.5	2000	O-QPSK	250	62.5	16-ary orthogonal
UWB sub-gigahertz (προαιρετικό)	250-750	N/A	N/A	N/A	N/A	N/A

2450 CSS (προαιρετικό)	2400-2483.5	N/A	N/A	250	167	
		N/A	N/A	1000	167	
UWB low band (προαιρετικό)	3244-4742	N/A				
UWB high band (προαιρετικό)	5944-10234	N/A				

**Πίνακας 2.1: Εύρος συχνοτήτων και ταχύτητα δεδομένων [5]**

### 2.3.2 Αναθέσεις Καναλιού

Οι διάφορες ζώνες ασύρματης επικοινωνίας, έχουν διαφορετικά χαρακτηριστικά και χρησιμοποιούνται με διαφορετικούς τρόπους. Ανάλογα με τη συχνότητα που χρησιμοποιείται, η ακτίνα μετάδοσης, η ευαισθησία στις παρεμβολές και οι ταχύτητες μετάδοσης δεδομένων ποικίλλουν. Η εμβέλεια που παρέχει κάθε ζώνη συχνοτήτων χωρίζεται σε κανάλια. Η ζώνη 2,4GHz, χωρίζεται σε 16 κανάλια μεταξύ 2,400GHz και 2,483GHz, όπως φαίνεται στον πίνακα 2.1. Υπάρχει επαρκής χώρος περίπου 5MHz μεταξύ γειτονικών καναλιών, έτσι ώστε θεωρητικά 16 πομποί να μπορούν να χρησιμοποιούν την ίδια ζώνη ταυτόχρονα.

- Για τις ζώνες συχνοτήτων, 780 MHz, υπάρχει ένα σύνολο από 8 κανάλια, αριθμημένα από 0 έως 7. Η κεντρική συχνότητα αυτών των καναλιών καθορίζεται ως εξής:
  - $F_c = 780 + 2k$  σε megahertz, για  $k = 0, \dots, 3$
  - $F_c = 780 + 2(k-4)$  σε megahertz, για  $k = 4, \dots, 7$
 όπου,  $k$  είναι ο δείκτης του καναλιού.
- Για τις ζώνες συχνοτήτων, 868 MHz, 915 MHz, και 2450 MHz, υπάρχει ένα σύνολο από 27 κανάλια, αριθμημένα από 0 έως 26.
- Για τη ζώνη των 2450 MHz υπάρχουν 16 κανάλια διαθέσιμα, 10 στη 915 MHz και 1 στη 868 MHz. Η κεντρική συχνότητα αυτών των καναλιών καθορίζεται ως εξής:
  - $F_c = 868.3$  σε megahertz, για  $k = 0$
  - $F_c = 906 + 2(k-1)$  σε megahertz, για  $k = 1, 2, \dots, 10$  και
  - $F_c = 2405 + 5(k-11)$  σε megahertz, για  $k = 11, 12, \dots, 26$
 όπου,  $k$  είναι ο δείκτης του καναλιού.

- Για τις ζώνες συχνοτήτων, 950 MHz, έχουμε ένα σύνολο από 22 κανάλια, αριθμημένα από 0 έως 21. Τα κανάλια 0–7 χρησιμοποιούνται για 1mW BPSK, τα 8–9 χρησιμοποιούνται για 10mW BPSK, και τα 10–21 για GFSK. Η κεντρική συχνότητα αυτών των καναλιών καθορίζεται ως εξής:
  - $F_c = 951.2 + 0.6k$  σε megahertz, για  $k = 0, \dots, 7$
  - $F_c = 954.4 + 0.2(k-8)$  σε megahertz, για  $k = 8, 9$  και
  - $F_c = 951.1 + 0.4(k-10)$  σε megahertz, για  $k = 10, \dots, 21$όπου,  $k$  είναι ο δείκτης του καναλιού.
- Για όλα τα υπόλοιπα PHYs εκτός από το UWB PHY, οι μικρότερες περίοδοι LIFS και SIFS περιγράφονται από τις πιο κάτω μεταβλητές και είναι:
  - macLIFSPeriod: 40 symbols
  - macSIFSPeriod: 12 symbols[6,17]

### 2.3.3 Ανίχνευση Ενέργειας

Το Φυσικό Επίπεδο είναι απαραίτητο να παρέχει τον τρόπο εκείνο, βάσει του οποίου να μπορεί να επιτευχθεί η ανίχνευση ενέργειας και αντιστοίχως η μέτρησής της. Η διαδικασία αυτή είναι ένα βασικό κομμάτι του επιπέδου δικτύου όπως και του αλγόριθμου βάσει του οποίου πραγματοποιείται η επιλογή καναλιών. Το ED (Energy Detection) είναι μια εκτίμηση της ανιχνευόμενης ισχύος σήματος που λαμβάνει ο δέκτης. Η ελάχιστη τιμή που μπορεί να λάβει είναι το μηδέν, στοιχείο που υποδεικνύει πως η ανιχνευόμενη ισχύς σήματος, είναι λιγότερη από 10 dB, πάνω από το επιτρεπόμενο όριο για κάθε δέκτη PHY. Πραγματοποιείται επίσης μια αντιστοίχιση των τιμών dB σε ED με βάση την τιμή ισχύος του σήματος εισόδου σε dB. Η αντιστοίχιση είναι γραμμική, με ακρίβεια  $\pm 6$ dB [6,18,19].

### 2.3.4 Δείκτης Ποιότητας Σύνδεσης

Το πρότυπο δίνει ιδιαίτερη έμφαση στον Δείκτη Ποιότητας Σύνδεσης (Link Quality Indication, LQI), ως μια μέθοδο μέτρησης της ισχύος και του πόσο καθαρό είναι ένα σήμα κατά τη λήψη του. Αυτή η μέτρηση μπορεί να γίνει με την χρησιμοποίηση διαφορετικών μεθόδων για τον προσδιορισμό της ποιότητας της σύνδεσης. Ωστόσο, το πόσο χρήσιμη μπορεί να είναι η μέτρηση αυτή σε ανώτερα επίπεδα OSI του δικτύου δεν καθορίζεται πλήρως από το 802.15.4. Κάθε κομμάτι δεδομένων που διέρχεται πρέπει να έχει μια μέτρηση LQI και η τιμή του είναι μεταξύ 0x00 και 0xFF. Πρέπει να υπάρχουν τουλάχιστον οκτώ διαφορετικές τιμές LQI που να ορίζονται ομοιόμορφα εντός αυτού του εύρους. [5,6,17].

## 2.4 Επίπεδο Ελέγχου Πρόσβασης Δεδομένων (MAC Media Access Control)

Το επίπεδο αυτό αποτελείται από δύο διαδικασίες. Η πρώτη, είναι ονομάζεται Υπηρεσία Δεδομένων, ενώ η δεύτερη Υπηρεσία Διαχείρισης, η οποία αντιστοίχως συνδέεται με το επίπεδο MLME (MAC Layer Management Entity). Η πρώτη είναι ένας τρόπος μέσω του οποίου οι συσκευές στέλνουν και λαμβάνουν πληροφορίες. Χρησιμοποιεί τη διαδικασία Protocol Data Units (MPDU) για να επιτύχει το σκοπό αυτό. Το επίπεδο MAC επιτελεί διαφορετικές εργασίες, όπως τη διαχείριση του τρόπου με τον οποίο οι συσκευές επικοινωνούν μεταξύ τους, τη διασφάλιση ότι οι πληροφορίες είναι σωστές και τη διατήρηση της ασφάλειάς τους. [10,18].

### 2.4.1 Υπερπλαίσιο

Μέσω του προτύπου αυτού δίνεται η δυνατότητα για τη προαιρετική χρήση της δομής superframe. Αυτό έχει τη μορφή που ορίζεται από έναν συντονιστή ο οποίος ορίζεται για την μετάδοση των πακέτων. Καθορίζονται σημεία (beacons), όπου χωρίζονται σε δεκαέξι χρονικά διαστήματα, ίσης διάρκειας μεταξύ τους. Ο συντονιστής στέλνει σήματα για να σηματοδοτήσει την έναρξη και το τέλος του προγράμματος. Η μετάδοση της πληροφορίας δεν είναι συνεχής, αλλά διακόπτεται όταν δεν χρειάζεται να σταλεί πληροφορία, για εξοικονόμηση ενέργειας. Είναι προαιρετική η ύπαρξη ενός ενεργού και ενός ανενεργού τμήματος στο superframe. Η χρονική στιγμή της πρώτης υποδοχής σηματοδοτεί την αφετηρία για τη μετάδοση του πλαισίου beacon. Η μετάδοση αυτή απενεργοποιείται στις περιπτώσεις που δεν υπάρχει η επιθυμία για τη χρήση του superframe.

Κατά τη χρονική διάρκεια που επιτελείται το CAP (Contention Access Period), εκτυλίσσεται μια διαδικασία ανταγωνισμού μεταξύ των συσκευών στην προσπάθειά τους, για επικοινωνία, μεταξύ των δύο σημείων beacon. Η διαδικασία αυτή πραγματοποιείται μέσω των τεχνολογιών ALOHA ή CSMA-CA. Ο συντονιστής PAN διαμοιράζει τμήματα του χρονικού πλαισίου σε εφαρμογές οι οποίες πρέπει να υποστηρίζουν τη γρήγορη λειτουργία ή τη χρήση μεγάλου πλήθους δεδομένων. Τα τμήματα αυτά λαμβάνουν την ονομασία GTS (Guaranteed Time Slots).

Το σύνολο αυτό των GTS συντελεί μια χρονική περίοδο όπου ονομάζεται CFS (Contention Free Period). Η χρονική αυτή περίοδος κάνει την εμφάνισή της στο τελευταίο μέρος του υπερπλαισίου. Το άθροισμα των επτά GTS είναι το μέγιστο που μπορεί να διατεθεί στις συσκευές από έναν συντονιστή. Κατά την έναρξη της CFP θα πρέπει να είναι εξασφαλισμένο πως όλες οι επικοινωνίες CAP, έχουν ήδη ολοκληρωθεί. [9-10,16].

### 2.4.2 Ανταλλαγή Δεδομένων

Η ανταλλαγή δεδομένων δύναται να επιτευχθεί μέσω τριών διαφορετικών τρόπων. Ο πρώτος τρόπος περιγράφει μια συσκευή, η οποία στέλνει πληροφορίες σε έναν

κεντρικό συντονιστή. Ο δεύτερος τρόπος αναφέρεται σε μια συσκευή που λαμβάνει πληροφορίες από τον συντονιστή. Ο τρίτος τρόπος παρουσιάζεται στις περιπτώσεις που δύο συσκευές ανταλλάζουν απευθείας πληροφορίες μεταξύ τους από κόμβο σε κόμβο (Peer-to-Peer). Σε ένα δίκτυο το οποίο έχει στηθεί με τοπολογία αστέρα, δύνανται να χρησιμοποιηθούν μόνο οι δύο πρώτοι τρόποι επικοινωνίας, καθώς οι πληροφορίες μοιράζονται μόνο μεταξύ του συντονιστή και της εκάστοτε συσκευής.

Σε μια τοπολογία Peer to Peer, τα δεδομένα ανταλλάσσονται μεταξύ οποιωνδήποτε συσκευών εντός ενός δικτύου PAN. Επομένως και οι τρεις τρόποι επικοινωνίας μπορούν να αναπτυχθούν σε αυτήν την τοπολογία. Ο μηχανισμός λειτουργίας που ορίζεται για κάθε έναν τύπο μετάδοσης, εξαρτάται από τη δυνατότητα του δικτύου να υποστηρίζει την περιοδική μετάδοση των beacons. Τα beacon enabled PANs χρησιμοποιούνται σε δίκτυα που απαιτούν συγχρονισμό ή την υποστήριξη συσκευών χαμηλής καθυστέρησης μετάδοσης της πληροφορίας, όπως είναι για παράδειγμα τα περιφερειακά των υπολογιστών. Στις περιπτώσεις που το δίκτυο δεν απαιτεί συγχρονισμό ή υποστηρίζει συσκευές με μεγάλη καθυστέρηση, μπορεί να επιλεγεί η παράλειψη beacons για κανονική μετάδοση δεδομένων. Υποχρεωτική κρίνεται η παρουσία τους κατά την ανάγκη αναζήτησης δικτύου.

Όταν μια συσκευή πρέπει να μεταδώσει δεδομένα από έναν συντονιστή σε ένα beacon enabled PAN, αρχικά αναζητά το κατάλληλο χρονικό σημείο στο δίκτυο. Όταν βρεθεί, η συσκευή συγχρονίζεται με τη δομή του υπερπλαισίου. Στο κατάλληλο χρονικό διάστημα, η συσκευή μεταδίδει το πλαίσιο δεδομένων στον συντονιστή. Εάν ζητηθεί, ο συντονιστής θα επιβεβαιώσει την επιτυχή λήψη των δεδομένων στέλνοντας ένα πλαίσιο επιβεβαίωσης. Όταν μια συσκευή επιθυμεί να μεταδώσει δεδομένα σε ένα non beacon enabled PAN, απλά μεταδίδει το πλαίσιο δεδομένων της στον συντονιστή. Ο συντονιστής ολοκληρώνει την επικοινωνία στέλνοντας ένα προαιρετικό πλαίσιο επιβεβαίωσης για να επιβεβαιώσει την επιτυχή λήψη των δεδομένων.

Στην περίπτωση που πρέπει να πραγματοποιηθεί ανταλλαγή δεδομένων μεταξύ μιας συσκευής και ενός συντονιστή, στο πλαίσιο ενός beacon enabled PAN, πρώτα γίνεται αναζήτηση για τα σημεία αυτά του δικτύου. Εφόσον κάποιο από αυτά εντοπιστεί, ακολουθεί ο συγχρονισμός της συσκευής με τη δομή υπερπλαισίου. Ακολούθως δρομολογείται η αποστολή του πλαισίου δεδομένων προς τον συντονιστή, ο οποίος αναγνωρίζει την επιτυχή λήψη του αιτήματος δεδομένων μεταδίδοντας ένα πλαίσιο επιβεβαίωσης. Το εκκρεμών προς αποστολή πλαίσιο δεδομένων στη συνέχεια, αποστέλλεται από τον συντονιστή. Η συσκευή αναγνωρίζει την επιτυχή λήψη των δεδομένων μεταδίδοντας ένα πλαίσιο επιβεβαίωσης, εφόσον έχει ζητηθεί. Η μεταφορά δεδομένων έχει πλέον τελειώσει.

Μόλις ολοκληρωθεί επιτυχώς η ανταλλαγή δεδομένων, το μήνυμα αφαιρείται από τη λίστα των εκκρεμών μηνυμάτων που βρίσκονται στο beacon. Όταν ο συντονιστής επιθυμεί να μεταδώσει δεδομένα σε μια συσκευή στο πλαίσιο ενός μη ενεργοποιημένου beacon PAN, αποθηκεύει τα δεδομένα ώστε η κατάλληλη συσκευή

να επικοινωνήσει και να ζητήσει τα δεδομένα. Αυτό γίνεται με αποστολή μιας εντολής σε επίπεδο MAC που προορισμό του αιτήματος στον συντονιστή της. Εκείνος επιβεβαιώνει την επιτυχή παραλαβή ενός αιτήματος δεδομένων στέλνοντας ένα πακέτο επιβεβαίωσης. Εάν αυτό ακόμα εκκρεμεί, ο συντονιστής το αποστέλλει ξανά. Εάν δεν υπάρχουν πλαίσια δεδομένων σε εκκρεμότητα, ο συντονιστής απαντάει στο αίτημα με ένα πακέτο επιβεβαίωσης δεδομένων μηδενικού μήκους. Εάν ζητηθεί, η συσκευή θα επιβεβαιώσει την επιτυχή λήψη του πλαισίου δεδομένων στέλνοντας ένα πλαίσιο επιβεβαίωσης.

Σε ένα P2P PAN, οι συσκευές έχουν τη δυνατότητα να ανταλλάσσουν πληροφορίες απευθείας μεταξύ τους όταν βρίσκονται σε κοντινή απόσταση. Προκειμένου να μην υπάρχουν απώλειες, οι συσκευές πρέπει να βρίσκονται σε λειτουργία αναμονής για κάποιο σήμα έτσι ώστε να μπορούν να συγχρονίζονται μεταξύ τους. [10,17,19].

### 2.4.3 Δομή Πλαισίου

Οι “Δομές Πλαισίου” (Frame Structures) αποτελούν βασικά στοιχεία που έχουν σχεδιαστεί για να συμβάλουν στην απλούστευση των διαδικασιών και να έχουν σήμα μετάδοσης ισχυρό, έτσι ώστε να λειτουργούν σε ικανοποιητικό βαθμό ακόμα και σε ένα κανάλι γεμάτο θόρυβο. Κάθε επίπεδο OSI του πρωτοκόλλου προσθέτει και μια δική του επικεφαλίδα και υποσέλιδο στη δομή, με σκοπό την καλύτερη αποκωδικοποίηση και ασφάλεια σε όλη της διαδρομή. Υπάρχουν τέσσερις βασικοί τύποι πλαισίων MAC:

1. Τα πλαίσια beacon, τα οποία υποδεικνύουν την ανάγκη του συντονιστή για τη μετάδοσή τους,
2. Τα πλαίσια δεδομένων, τα οποία είναι υπεύθυνα για τη μεταφορά των πληροφοριών
3. Τα πλαίσια αναγνώρισης, τα οποία επιβεβαιώνουν ότι οι πληροφορίες ελήφθησαν σωστά.
4. Τα πλαίσια εντολών MAC, που ελέγχουν τον τρόπο αποστολής των πληροφοριών. Αυτά τα πλαίσια τοποθετούνται στο Physical layer ως ωφέλιμο φορτίο προς μετάδοση. [10,19-20].

### 2.4.4 CSMA-CA

Το πρότυπο IEEE 802.15.4 LR-WPAN διαθέτει δύο τρόπους εισόδου σε ένα κανάλι, ανάλογα με τον τρόπο κατά τον οποίο είναι διαμορφωμένο το δίκτυο.

Σε ένα nonbeacon-enabled PAN, οι συσκευές χρησιμοποιούν το CSMA-CA (Carrier Sense Multiple Access with Collision Avoidance), προκειμένου να επιτευχθεί η είσοδός τους στο κανάλι. Όταν μια συσκευή επιθυμεί να αποστείλει πληροφορίες μέσω πακέτου δεδομένων ή των εντολών MAC, τότε υπάρχει ένα συγκεκριμένο χρονικό περιθώριο, προκειμένου να διαπιστωθεί εάν το κανάλι είναι αδρανές. Εάν



είναι, τότε μετά το πέρας της αναμονής πραγματοποιείται η αποστολή των δεδομένων.. Αν όχι, επεκτείνεται ο χρόνος αναμονής και γίνεται εκ νέου προσπάθεια. Ο μηχανισμός CSMA-CA δεν χρησιμοποιείται κατά την αποστολή των πλαισίων επιβεβαίωσης.[9].

Σε ένα beacon-enabled PAN, χρησιμοποιείται ο μηχανισμός CSMA-CA για τη σύνδεση σε ένα κανάλι, με τις χρονικές περιόδους αναμονής να είναι εναρμονισμένες με την εκκίνηση μετάδοσης του σημείου αυτού. Οι χρόνοι αναμονής των συσκευών επίσης εναρμονίζονται με τον συντονιστή PAN. Στην περίπτωση που μια συσκευή πρέπει να επικοινωνήσει με μετάδοση των πλαισίων δεδομένων κατά τη χρονική περίοδο CAP, ανιχνεύει την περίοδο αναμονής (backoff period) και τίθεται σε αναμονή για έναν αριθμό χρονικών περιόδων αναμονής. Όταν το κανάλι δεν είναι ελεύθερο η αναμονή αυτή παρατείνεται, προκειμένου να γίνει σε μεταγενέστερο χρόνο, εκ νέου προσπάθεια. Όταν το κανάλι είναι διαθέσιμο, ακόμα και αν βρίσκεται σε κατάσταση αδράνειας, η συσκευή εκκινεί τη διαδικασία αποστολής δεδομένων με διάρκεια έως την επόμενη διαθέσιμη χρονική περίοδο. Όπως και στην προηγούμενη περίπτωση, ο μηχανισμός CSMA-CA δεν χρησιμοποιείται κατά την αποστολή των πλαισίων επιβεβαίωσης[10,19].

#### 2.4.5 Υπηρεσίες

Η υπηρεσία σε ένα επίπεδο αποτυπώνει τη λειτουργικότητα που παρέχεται στους χρήστες στο αμέσως επόμενο υψηλότερο ή χαμηλότερο επίπεδο OSI, με βάση τη λειτουργικότητα του χαμηλότερου επιπέδου που προηγείται. Μια υπηρεσία ορίζεται περιγράφοντας τη ροή πληροφοριών μεταξύ N-χρηστών και N-επιπέδων. Η ροή των πληροφοριών, συγκροτείται από διακριτά και στιγμιαία γεγονότα, τα οποία με τη σειρά τους διαμορφώνουν την υπηρεσία. Κάθε γεγονός βασίζεται στη μετάβαση μιας βασικής υπηρεσίας (Primitive Service) από ένα επίπεδο σε ένα άλλο μέσω του επιπέδου SAP (System Access Point) που σχετίζεται με N-χρήστες.

Μια υπηρεσία περιγράφεται μέσα από τις δράσεις της και λεπτομέρειές της, οι οποίες την κάνουν μοναδική. Υπάρχουν τέσσερις κύριοι τύποι βασικών υπηρεσιών:

- Αίτηση: Σκοπός της συγκεκριμένης υπηρεσίας είναι η αίτηση ενεργοποίησης κάποιας διαδικασίας.
- Ένδειξη: Σκοπός της είναι η ανάδειξη ενός γεγονότος στον χρήστη.
- Απάντηση: Σκοπός της είναι η ολοκλήρωση μια διαδικασίας που έχει προκληθεί προηγουμένως.
- Επιβεβαίωση: Σκοπός της είναι η μετάδοση των αποτελεσμάτων των υπηρεσιών αίτησης. [10,16,18].

#### 2.4.6 Κατανάλωση Ισχύος

Σε ένα μεγάλο αριθμό εφαρμογών στις οποίες χρησιμοποιείται το πρότυπο αυτό, οι συσκευές λειτουργούν με τροφοδοσία που παρέχεται μέσω μπαταριών, με την τακτική αντικατάσταση ή επαναφόρτισή τους να μην καθίσταται πάντοτε εφικτή. Δεδομένου αυτού, η κατανάλωση ενέργειας αποτελεί το σημαντικότερο στόχο αυτού του προτύπου. Τα πρωτόκολλα έχουν σχεδιαστεί κατά τέτοιο τρόπο, ώστε να δίνεται προτεραιότητα στις συσκευές οι οποίες λειτουργούν με μπαταρίες. Ωστόσο, ορισμένες υλοποιήσεις λειτουργούν μέσω συσκευών οι οποίες χρησιμοποιούν ηλεκτρική τροφοδοσία. Οι συσκευές που τροφοδοτούνται από μπαταρίες απαιτούν να είναι γνωστός ο Κύκλος Λειτουργίας (duty cycling) για τον καλύτερο υπολογισμό της κατανάλωσης ενέργειας. Οι συσκευές αυτές, έχουν την δυνατότητα τις περισσότερες ώρες λειτουργίας τους να βρίσκονται σε κατάσταση αναστολής λειτουργία. Η κάθε συσκευή ενεργοποιείται από ένα παλμό ενεργοποίησης, προκειμένου να τις θέσει σε λειτουργία αν εκκρεμεί κάποιο μήνυμα προς αποστολή. Αυτός ο μηχανισμός δίνει τη δυνατότητα στους προγραμματιστές εφαρμογών, να καθορίζουν την ισορροπία μεταξύ της κατανάλωσης ενέργειας και της καθυστέρησης των μηνυμάτων. [11,17,20].

## **ΚΕΦΑΛΑΙΟ 3ο : Πρωτόκολλα βασισμένα στο 802.15.4**

Στο κεφάλαιο αυτό, πρωτόκολλα τα οποία είναι βασισμένα στο πρότυπο 802.15.4 θα μελετηθούν. Μια εκτενής περιγραφή των χαρακτηριστικών τους θα παρουσιαστεί, με ιδιαίτερη έμφαση στην τεχνολογία Zigbee λόγω το ότι αποτελεί την πιο διαδεδομένη τεχνολογία η οποία βασίζεται στο πρότυπο αυτό.

### **3.1 Πρωτόκολλα βασισμένα στο 802.15.4**

#### **3.1.1 Το πρότυπο “ISA100.11a”**

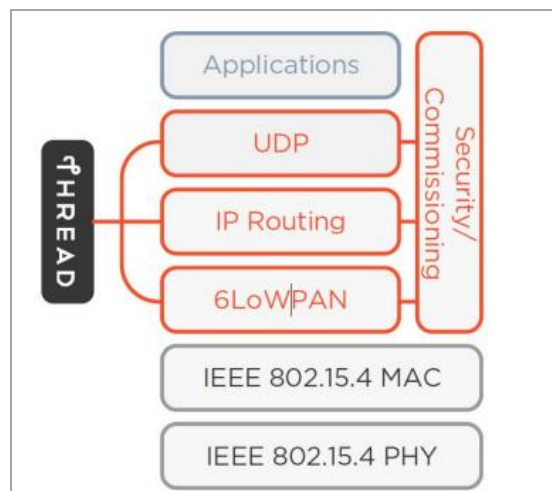
Το 2009 το International Society of Automation παρουσίασε ένα νέο πρότυπο, το ISA100.11a. Το πρότυπο αυτό, προκειμένου να επιτευχθεί η μεταφορά πακέτων και η πρόσβαση στα μέσα αυτών, χρησιμοποιεί του μηχανισμούς TDMA (Time Division Multiple Access) και CSMA/CA. Στους μηχανισμούς που χρησιμοποιεί προστίθενται επίσης εκείνοι της μεθόδου δρομολόγησης μέσω γραφήματος όπως και της εναλλαγής καναλιών μετάδοσης με σκοπό την αποφυγή των δικτυακών συγκρούσεων μεταξύ τους. Και στο παρόν πρότυπο πραγματοποιείται η ομαδοποίηση των χρονικών περιθωρίων σε υπερπλαίσια, των οποίων η επανεμφάνιση γίνεται τακτικά. Οι συσκευές δεν διαθέτουν τη δυνατότητα δρομολόγησης πακέτων, γεγονός που το καθιστά ευάλωτο στις εναλλαγές της τοπολογίας και το διαφοροποιεί από το πρότυπο Wireless HART. Συναντάται κυρίως σε βιομηχανικού τύπου διαδικασίες και αφορά επικοινωνίες μεταξύ των Ασυρμάτων Δικτύων. [13]

#### **3.1.2 WirelessHART**

Ένα άλλο πρωτόκολλο το οποίο βασίζεται στο πρότυπο 802.15.4, είναι το WirelessHart. Το πρωτόκολλο αυτό αποτελείται από τα επίπεδα Φυσικό, Μεταφοράς δεδομένων, δικτύου και εφαρμογής. Στο παρόν πρωτόκολλο γίνεται συνδυασμός της τεχνικής πρόσβασης TDMA και της υπερπήδησης των συχνοτήτων στο επίπεδο μεταφοράς δεδομένων. Για τον έλεγχο της κυκλοφορίας των πακέτων που διακινούνται, χρησιμοποιείται ένας διαχειριστής δικτύου, μέσω του οποίου δίνονται οι εντολές για την εκκίνηση της σύνδεσης και για τον προσδιορισμό των πόρων για τη μετάδοση των δεδομένων. Δίνεται η δυνατότητα στις συσκευές να εκπέμπουν παράλληλα μέσω μόνο ενός καναλιού για καθεμία από αυτές, ενώ κάθε συσκευή δύναται να εκτελέσει διαδικασία δρομολόγησης πακέτων. Οι μηχανισμοί απομόνωσης σε περιπτώσεις εμφάνισης θορύβου στο κανάλι και η υποστήριξη τοπολογιών πλέγματος, αποτελούν επίσης χαρακτηριστικά της τεχνολογίας αυτής. Πρόκειται για μια υλοποίηση WPAN με προδιαγραφές λειτουργίας σε βιομηχανικούς χώρους. Καταφέρνει επίσης, να αναδιοργανώνεται αυτόματα μόνο του και να παρουσιάζει τη δυνατότητα της αυτοϊασης[13,21].

### 3.1.3 Thread

Πρόκειται για ένα πρωτόκολλο με στόχο κυρίως την εξυπηρέτηση διαδικασιών αυτοματισμού σε οικιακό περιβάλλον με βάση το 6LoWPAN. Έχει άμεση σύνδεση με το Wi-Fi και μπορεί να λειτουργεί συμπληρωματικά, καθώς καλείται πρωτίστως να καλύψει τους περιορισμούς όπου μπορεί να έχει σε μια οικιακή εγκατάσταση αυτοματισμού. Είναι ένα πρωτόκολλο το οποίο χαρακτηρίζεται ανοιχτού κώδικα (open source), όπου κυκλοφόρησε πρώτη φορά το 2014 από την Thread Group. Έχει τη δυνατότητα να υποστηρίξει δίκτυα πλέγματος με τη χρήση πομποδεκτών 802.15.4 και να διαχειριστεί έως 250 κόμβους με επίπεδα κρυπτογράφησης και υψηλού ελέγχου ταυτότητας. Στην εικόνα 3.1 αποτυπώνεται μια επισκόπηση της προδιαγραφής Thread.



Εικόνα 3.1: Επισκόπηση προδιαγραφών Thread[15]

Το Thread χρησιμοποιεί από το IEEE 802.15.4 τα δύο πρώτα επίπεδα (Physical και MAC Layer), τα οποία έχουν ταχύτητες έως στα 250 kbps σε συχνότητες 2,4 GHz. Πιο συγκεκριμένα οι εκδόσεις IEEE 802.15.4-2006 και IEEE802.15.4-2015 περιγράφουν με περισσότερη σαφήνεια για το πρωτόκολλο αυτό. Το πρότυπο χρησιμοποιείται για την αξιόπιστη αποστολή μηνυμάτων μεταξύ μεμονωμένων συσκευών στο επίπεδο συνδέσμου (Link Level). Παρέχει επίσης τον μηχανισμό CSMA-CA (Carrier Sense Multiple Access – Collision Avoidance), προκειμένου να επιτραπεί σε πολλές συσκευές να χρησιμοποιήσουν το εύρος ζώνης 2.4 GHz χωρίς να υπάρχει ταυτόχρονη εκπομπή σήματος.

Στον πίνακα 3.1 αποτυπώνονται συγκεντρωτικά τα βασικά χαρακτηριστικά του Thread.

Προδιαγραφές	Thread Data
Εστίαση σχεδιασμού	Συνδεσιμότητα στο σπίτι στο IoT
Πρότυπο IEEE	802.15.4
Τύπος δικτύου	Star, Mesh
Δίκτυο	Thread
Απόσταση	20-30 μέτρα
Μέγιστος αριθμός κόμβων	250
Λειτουργική ζώνη	2,4 GHz (Η ζώνη χωρίς άδεια ISM)
Ευρύ φάσμα	Χρήση άμεσου φάσματος διασποράς ακολουθίας (DSSS)
Διακίνηση	250 kbps
Δεδομένα	Δεδομένα παρακολούθησης και ελέγχου
Δυνατότητα φωνής	Όχι
Ασφάλεια	Υψηλή, κρυπτογράφηση δημόσιου κλειδιού
Κατανάλωση ενέργειας	Χαμηλή
Διαμόρφωση	O-QPSK

Πίνακας 3.1: Χαρακτηριστικά Thread[15]

Οι συσκευές που χρησιμοποιούνται χωρίζονται σύμφωνα με τα χαρακτηριστικά τους, στις Full Thread Device (FTD) και στις Minimal Thread Device (MTD). Οι MTD συσκευές έχουν λιγότερες απαιτήσεις όπως το μέγεθος της μνήμης και η κατανάλωση της ενέργειας, εν αντιθέσει με τις συσκευές FTD οι οποίες δύνανται να είναι περισσότερο ευέλικτες αναφορικά με τους ρόλους που ενδέχεται να παίξουν εντός του δικτύου. Αναλυτικότερα οι συσκευές αναλογικά με τις ιδιότητές τους χωρίζονται ως εξής:

- Routing Full Thread Devices:
  - Router
  - Leader
- Non-Routing Full Thread Devices:
  - Router-Eligible End Device (REED)
  - Full End Device (FED)
- Non-Routing Minimal Thread Devices:
  - Minimal End Device (MED)
  - Sleepy End Device (SED)
  - Synchronized Sleepy End Device (SSED)
  - Bluetooth End Device (BED)
  - Border Router [22,23].

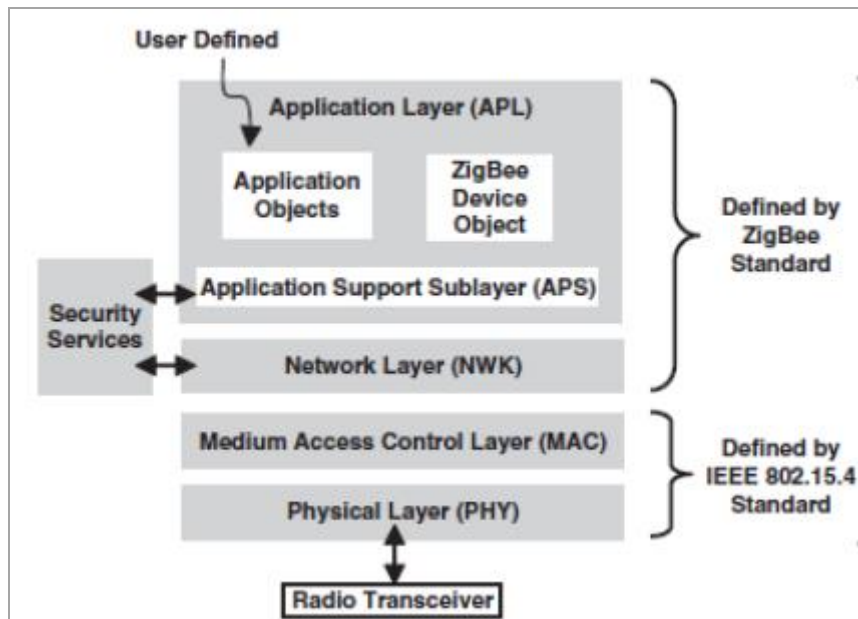
### 3.2 Zigbee

Το Zigbee είναι επίσης ένα πρότυπο το οποίο έχει σαν βάση το πρότυπο IEEE 802.15.4. Εκπέμπει κυρίως στη συχνότητα των 2,4 GHz με ξεχωριστές διαφοροποιήσεις να συναντιούνται ανά ήπειρο με την εκπομπή του στην Ευρώπη να είναι στα 868 MHz, σε Αμερική - Αυστραλία στα 915 MHz και 784 MHz στην Κίνα, με καθορισμένη ταχύτητα μετάδοσης στα 250 kbps. Χαρακτηρίζεται ως το απλούστερο και λιγότερο δαπανηρό σχετικά με τα υπόλοιπα WPAN με τις αποστάσεις μετάδοσης να κυμαίνονται από τα 10 έως τα 100 μέτρα. Χρησιμοποιείται σε εφαρμογές που απαιτούν χαμηλό ρυθμό μετάδοσης δεδομένων με αυτονομία μπαταρίας μεγάλης διάρκειας και προσφορά ασφάλειας επί του δικτύου.

Η σύσταση και η δομή του πρωτοκόλλου Zigbee, περιγράφεται μέσω τεσσάρων επιπέδων. Τα επίπεδα αυτά κατά βάση στηρίζονται πάνω στο μοντέλο δικτύωσης OSI-7 με ιδιέτερη έμφαση στα επίπεδα 4 και 7, λόγω της βαρύνουσας σημασίας τους στις διεργασίες του πρωτοκόλλου. Το επίπεδο μεταφοράς OSI-4, είναι εκείνο μέσω του οποίου επιτυγχάνεται η μεταφορά δεδομένων μεταξύ των χρηστών, απαλλάσσοντας με τον τρόπο αυτό τα ανώτερα επίπεδα από τη διαδικασία αυτή. Σε αυτό το επίπεδο διεκπεραιώνονται επίσης διεργασίες όπως ο έλεγχος της αξιοπιστίας του χρησιμοποιούμενου καναλιού με έλεγχο ροής, η κατάτμηση και η αποτμηματοποίηση και ο έλεγχος ροής. Στο επίπεδο εφαρμογών (OSI-7), προσφέρεται η ομαλή επικοινωνία μεταξύ χρήστη και εφαρμογής στα πλαίσια ενός «ανθρώπινου» περιβάλλοντος. Μέσω του επιπέδου αυτού ο χρήστης έχει τη δυνατότητα του ελέγχου των αποτελεσμάτων των πληροφοριών που μεταφέρονται και της πρωτοβουλίας της καθοδήγησης των επόμενων κινήσεων της εφαρμογής. Το σύνολο των επιπέδων ωστόσο επιτελούν το δικό τους ρόλο και καθίστανται απαραίτητα μέσω της αλληλουχίας τους για τη εύρυθμη λειτουργία του μοντέλου OSI.

Δεδομένου ότι όλα τα επίπεδα είναι ανεξάρτητα μεταξύ τους, δίνει το πλεονέκτημα της επεξεργασία καθενός από αυτά, δίχως να υπάρχει αυτομάτως και ενδεχόμενη αλλοίωση των υπολοίπων. Το γεγονός αυτό είναι πολύ σημαντικό μέσα σε μια δικτύωση, καθώς δεν είναι λίγες οι φορές που ο χρήστης καλείται να πραγματοποιήσει αλλαγές. Ο διαχωρισμός των επιπέδων αυτών, όπως έχω αναφερθεί, αποτυπώνεται με τα πρώτα δύο επίπεδα, το φυσικό και το επικοινωνίας δεδομένων. Η διαφοροποίηση που προσφέρεται μέσω του του πρωτοκόλλου Zigbee, έγκειται στην ανάπτυξη ενεργειών στα δύο ανώτερα επίπεδα. Συστά είναι το δικτύου και το υπόστρωμα υποστήριξης εφαρμογών. Επιπροσθέτως είναι το καθ' ύλην αρμόδιο και για τις υπηρεσίες ασφάλειας στο ενδιάμεσο των δύο αυτών επιπέδων.

Στην εικόνα 3.2 παρουσιάζονται τα επίπεδα που ορίζονται από το πρότυπο 802.15.4 και τα επίπεδα που ορίζονται από το Zigbee. [22,24]



Εικόνα 3.2: Δομή Zigbee [23]

### 3.1.1 Υπηρεσίες

Κάθε επίπεδο παρέχει μια σειρά από υπηρεσίες (Services) που εκτελούνται συνήθως για λογαριασμό του αμέσως ανώτερου επιπέδου OSI. Όλες οι υπηρεσίες πραγματοποιούνται από τον φορέα διαχείρισης εκτός από την μεταφορά δεδομένων που γίνεται από τον αντίστοιχο δεδομένων. Ένα υψηλότερο επίπεδο αποκτά πρόσβαση στις υπηρεσίες του χαμηλότερου από αυτό επίπεδο, με τη βοήθεια των σημείων πρόσβασης υπηρεσιών (Service Access Points). Για παράδειγμα στην περίπτωση που χρειαστεί η ενεργοποίηση του πομποδέκτη, θα πρέπει το MAC επίπεδο να χρησιμοποιήσει μέσω του PD-SAP (Physical Data – SAP) το PHY Data Service το οποίο θα επιτρέψει την αποστολή και λήψη των PDUs( Protocol Data Units)[23,25]

### 3.1.2 Συσκευές

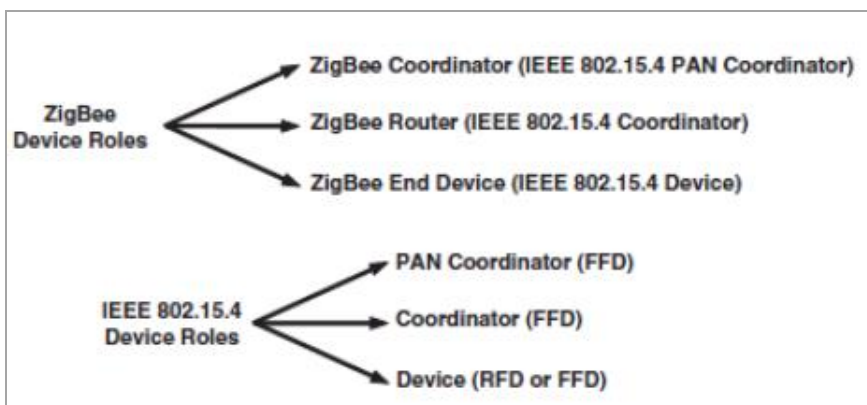
Οι συσκευές ενός δικτύου Zigbee χωρίζονται στις παρακάτω κατηγορίες:

- Συντονιστής (Coordinator): Αποτελεί την κεντρική συσκευή του δικτύου Zigbee. Είναι εκείνη που είναι αρμόδια για την έναρξη και τη λήξη μιας εφαρμογής. Επιτελεί τον πολύ σημαντικό ρόλο της σύνδεσης μεταξύ των συσκευών που απαρτίζουν το δίκτυο. Είναι επίσης αρμόδια για τη διάδοση της πληροφορίας εντός του δικτύου όπως και τα κλειδιά ασφαλείας (security keys).
- Δρομολογητής (Router): Βασική του λειτουργία αποτελεί η αναμετάδοση της εκάστοτε πληροφορίας εντός του δικτύου. Μπορεί επιπροσθέτως να επιτελέσει

εναλλακτικές λειτουργίες για την υποβοήθηση λειτουργιών που απαιτούνται, όπως και η οποιαδήποτε άλλη συσκευή που βρίσκεται μέσα στο δίκτυο αυτό.

- Τερματικές Συσκευές (End Devices): Μέσω αυτού του τύπου της συσκευής επιτελείται η μετάδοση της πληροφορίας απευθείας στον δρομολογητή και τον συντονιστή χωρίς να χρειαστεί η μετάδοση πρώτα σε οποιαδήποτε άλλη συσκευή εντός δικτύου. Στα πλεονεκτήματα της προστίθεται η χαμηλή κατανάλωση ενέργειας καθώς σύμφωνα με δυνατότητα που έχει ήδη αναλυθεί, για μεγάλο χρονικό διάστημα δύναται να παραμείνει αδρανής.

Στην Εικόνα 3.2 γίνεται η αποτύπωση των παραπάνω κατηγοριών, καθώς και η παρουσίαση των αντίστοιχων με αυτές συσκευών του προτύπου 802.15.4[23-25].



**Εικόνα 3.3: Τοπολογία ενός δικτύου Zigbee σε αναλογία με το πρότυπο 802.15.4 [23]**

Η αντιστοίχιση είναι η εξής :

- Συσκευή μειωμένης λειτουργίας (Reduced Function Device-RFD) ή Συσκευή πλήρους Λειτουργίας (Full Function Device-FFD / Zigbee End Device (IEEE 802.15.4 Device)  
Και στις δύο περιπτώσεις πρόκειται για μια συσκευή με τυπικές αρμοδιότητες. Στις αρμοδιότητές της δεν εμπεριέχονται διαδικασίες όπως η δρομολόγηση και τις περισσότερες εκ των περιπτώσεων βρίσκεται σε κατάσταση αδράνειας.
- Συντονιστής (FFD) / Zigbee Router (IEEE 802.15.4 Coordinator)  
Και στις δύο εκ των περιπτώσεων επιτελεί τη λειτουργία ενός τυπικού δρομολογητή, με αρμοδιότητα τη δρομολόγηση πακέτων δεδομένων προς όλες τις συσκευές και τους υπόλοιπους δρομολογητές.
- Συντονιστής PAN (FFD) / Συντονιστής Zigbee (IEEE 802.15.4 PAN Coordinator)  
Ο συντονιστής επίσης δεν παρουσιάζει ουσιαστικές διαφορές, καθώς είτε γίνεται αναφορά στο Zigbee είτε στο 802.15.4, οι βασικές διεργασίες ρύθμισης του δικτύου, παραμένουν ίδιες [23,26].



### 3.1.3 Σύνδεση συσκευών

Ως σύνδεση συσκευών (Binding) ορίζεται η κατάσταση κατά την οποία συγκροτούνται οι συνδέσεις μεταξύ των συσκευών. Πιο συγκεκριμένα, όταν γίνεται αναφορά για αυτού του τύπου τη σύνδεση, οι συνδέσεις χαρακτηρίζονται ως λογικές. Το σύνολο όλης αυτής της πληροφορίας αποθηκεύεται σε πίνακες. Η πίνακες αυτοί όπως είναι φυσικό λαμβάνουν την ονομασία Πίνακες Συνδέσεων (Binding Tables). Ένα τέτοιο παράδειγμα σύνδεσης θα μπορούσε να χαρακτηριστεί αυτό μεταξύ μια περιφερειακής συσκευής όπως ένας ανιχνευτής κίνησης (motion detector) ο οποίος θα πρέπει να μεταδώσει την πληροφορία στον πίνακα του συναγερμού. [23,29]

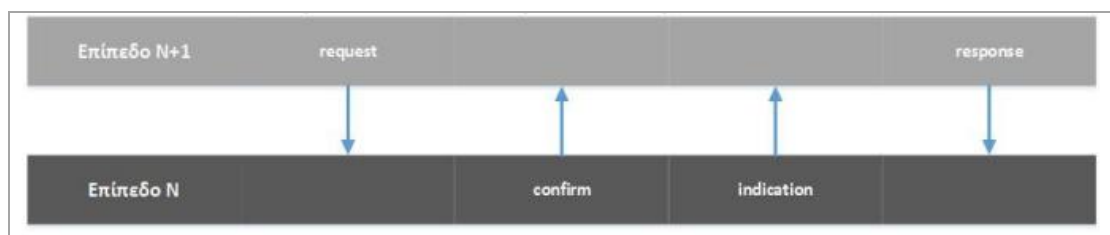
### 3.1.4 Θεμελιακά Στοιχεία

Η οποιαδήποτε υπηρεσία απαρτίζεται από ένα σύνολο επιλογών που ορίζονται ως θεμελιακά στοιχεία. Αυτές έχουν κάποια χαρακτηριστικά όπως και κάποιες συγκεκριμένες λειτουργίες όπως

- αίτηση (request)
- αποδοχή (confirm)
- ένδειξη (indication)
- απάντηση (response)

Η λειτουργία του καθενός από αυτά ορίζει σε ποιο εκ των επιπέδων είναι ορισμένο όπως και το ποιες είναι οι λειτουργίες του. Ο τρόπος δομής του περιλαμβάνει κάποια χαρακτηριστικά όπως το όνομα και ο τύπος

Στην Εικόνα 3.3 που ακολουθεί παρουσιάζεται η μεταβίβαση δεδομένων ανάμεσα σε δύο επίπεδα με τη συνδρομή ενός primitive.[23-24,26]

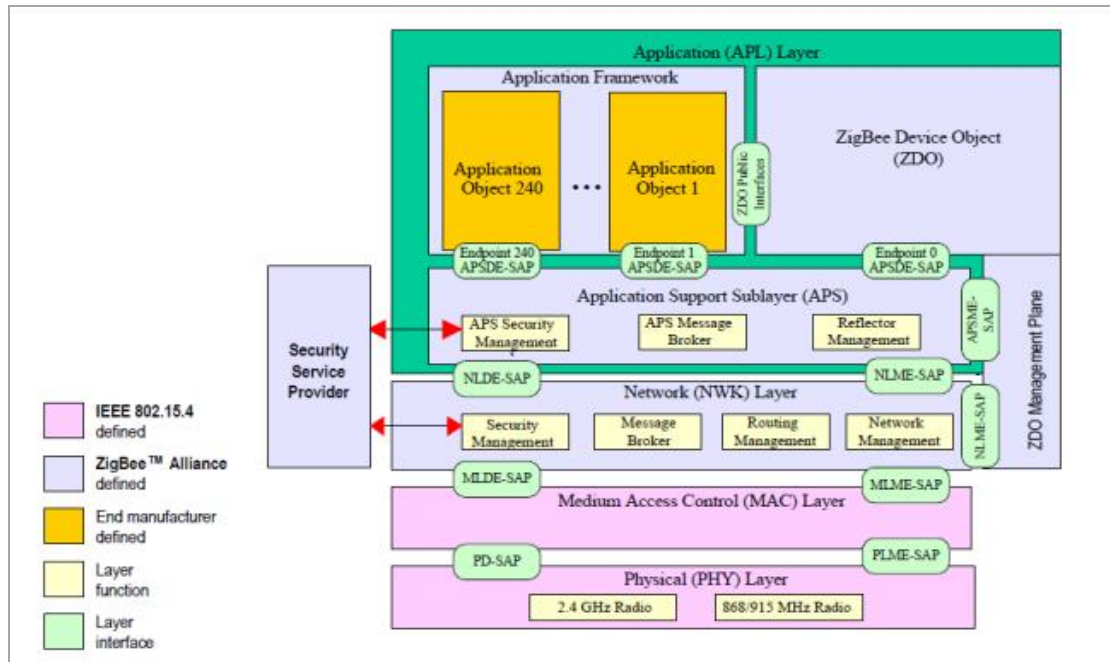


Εικόνα 3.4: Λειτουργίες primitive [23]

### 3.1.5 Φορείς Δεδομένων - Διαχείριση – Σημεία Ελέγχου Υπηρεσιών

Σε κάθε επίπεδο μπορεί να διακριθούν δυο χαρακτηριστικά, η οντότητα δεδομένων και εκείνη της διαχείρισης. Η πρώτη έχει ως σκοπό το χειρισμό της μετάδοσης των δεδομένων, ενώ η δεύτερη τον χειρισμό των υπολοίπων διεργασιών του επιπέδου. Το υψηλότερο επίπεδο μπορεί να έχει πρόσβαση στις διεργασίες του αμέσως

χαμηλότερου από αυτό. Αυτό επιτυγχάνεται μέσω των Σημείων Ελέγχου Υπηρεσιών (SAPs, Service Access Points), τα οποία διαχωρίζονται σε εκείνα που αφορούν την είσοδο σε διεργασίες αναφορικά με τα δεδομένα (DE-SAP) και σε αυτά που αφορούν την είσοδο σε διεργασίες διαχείρισης (ME-SAP). Η εικόνας εικόνα 3.4 αποτυπώνει την λειτουργία αυτή.[25,26]



Εικόνα 3.5: SAPs [25]

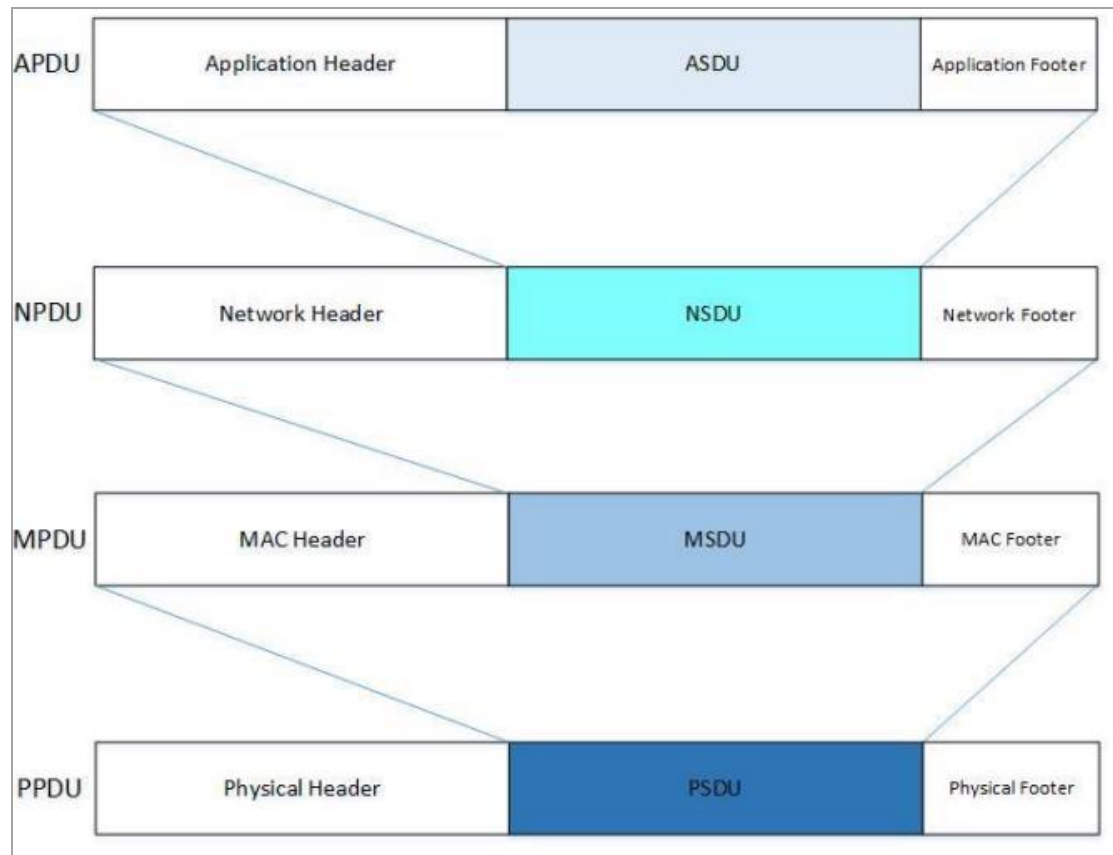
### 3.1.6 Μονάδες Δεδομένων Υπηρεσιών – Μονάδες Δεδομένων Πρωτοκόλλου

Η ομάδα των δεδομένων ενός επιπέδου λαμβάνει την ονομασία Protocol Data Unit (PDU) και για κάθε επόμενο επίπεδο προστίθεται ένα ακόμη γράμμα το οποίο χαρακτηρίζει την ιδιότητα του. Αναλυτικότερα παρατίθενται τα εξής :

- Μονάδα Δεδομένων Φυσικού Πρωτοκόλλου (Physical Protocol Data Unit)
- Μονάδα δεδομένων πρωτοκόλλου MAC (MAC Protocol Data Unit)
- Μονάδα δεδομένων πρωτοκόλλου δικτύου (Network Protocol Data Unit)
- Μονάδα δεδομένων πρωτοκόλλου εφαρμογής (Application Protocol Data Unit)

Όσο αναφορά την ωφέλιμη πληροφορία, είναι εκείνη που οδηγείται στο κατώτερο επίπεδο, προερχόμενη από το αμέσως πάνω από εκείνο. Παρατίθεται Η αντίστοιχη ονοματολογία που ακολουθείται είναι η παρακάτω:

- Μονάδα Δεδομένων Φυσικής Υπηρεσίας (Physical Service Data Unit)
- Μονάδα δεδομένων υπηρεσίας MAC (MAC Service Data Unit)
- Μονάδα δεδομένων πρωτοκόλλου δικτύου (Network Protocol Data Unit)
- Μονάδα δεδομένων υπηρεσίας εφαρμογής (Application Service Data Unit) [25]



**Εικόνα 3.6: Ενθυλάκωση [23]**

## ΚΕΦΑΛΑΙΟ 4ο : Το πρωτόκολλο 6LoWPAN

Η εξέλιξη της τεχνολογίας των WPANs οδήγησε στο πρότυπο 802.15.4, με ιδιαίτερη έμφαση στα χαμηλότερων απαιτήσεων ενεργειακά δίκτυα. Η μειωμένη κατανάλωση ενέργειας έχει ως ακόλουθο το αντίστοιχο αποτέλεσμα και στην απόδοση των δικτύων αυτών. Δεδομένου αυτού, μπορούν να χρησιμοποιηθούν σε όσες εφαρμογές όπου δεν χρειάζονται ιδιαίτερες απαιτήσεις. Με βάση το πρότυπο αυτό, δημιουργήθηκαν πρωτόκολλα για την υλοποίηση των δικτύων αυτών, με ποιο γνωστή την σουίτα TCP/IP. Η ιδιαιτερότητα όμως των πρωτοκόλλων αυτών αναφορικά με τη χαμηλή κατανάλωση ενέργειας δεν καθιστούσε εφικτή την επικοινωνία δίχως την παρουσία κάποιου μετατροπέα, καθώς ένα πρωτόκολλο όπως το IP έχει υψηλές απαιτήσεις σε ενέργεια και δεδομένα.

Στο σημείο αυτό τη λύση έδωσε ένα νέο για τα δεδομένα πρότυπο, το 6LoWPAN. Μέσω του προτύπου αυτού γίνεται η σύνδεση μεταξύ του πρωτοκόλλου IP και του προτύπου 802.15.4. Αυτό επιτυγχάνεται μέσω της επιτυχούς μετάδοσης των δεδομένων IPv6 από ένα επίπεδο, που σκοπό έχει να επιτελέσει το ρόλο αυτό. Σαν αποτέλεσμα, λαμβάνονται τα μεγέθη των επικεφαλίδων και με την απαραίτητη διαδικασία συμπίεσης και τον τεμαχισμό της πληροφορίας, επιτυγχάνεται η βέλτιστη επικοινωνία ανάμεσα στα επίπεδα και πιο συγκεκριμένα στο επίπεδο 2 του OSI. Με τον τρόπο αυτό, μέσω της χρήση του προτύπου 6LoWPAN, επιτυγχάνεται η καλύτερη επικοινωνία ανάμεσα στα επίπεδα.[23,30,31]

### 4.1 Συσχέτιση με το 802.15.4

Η βάση πάνω στην οποία έχει δομηθεί η αρχιτεκτονική του IPv6 η οποία θα αντικαταστήσει την υφιστάμενη έκδοση IPv4. Με την επέκταση της διεύθυνσης από τα 32 bits στα 128 bits, κατορθώθηκε να υπερκεραστεί το ζήτημα, αναφορικά με την έλλειψη διαθέσιμων διευθύνσεων. Με τη συμπίεση της αύξησης μεγέθους του εύρους ζώνης, αυξήθηκε η μεγέθυνση του όγκου των δεδομένων από τα 576 bytes στα 1280 bytes. Επιπροσθέτως με σκοπό την πιο απλούστευση των διαδικασιών των δρομολογητών όπως και την επίτευξη υψηλότερων επιδόσεων, δεν υλοποιείται ο τεμαχισμός των δεδομένων στους δρομολογητές, αλλά στα άκρα του δικτύου.

Το πρωτόκολλο IPv6 θεωρείται και ένα πολύ χρήσιμο εργαλείο αναφορικά με τη μετάδοση δεδομένων προς πολλούς παραλήπτες. Υπάρχουν όμως και άλλα χαρακτηριστικά του, τα οποία οδηγούν στην αναγνώριση της χρησιμότητάς του, με κάποια από αυτά να είναι :

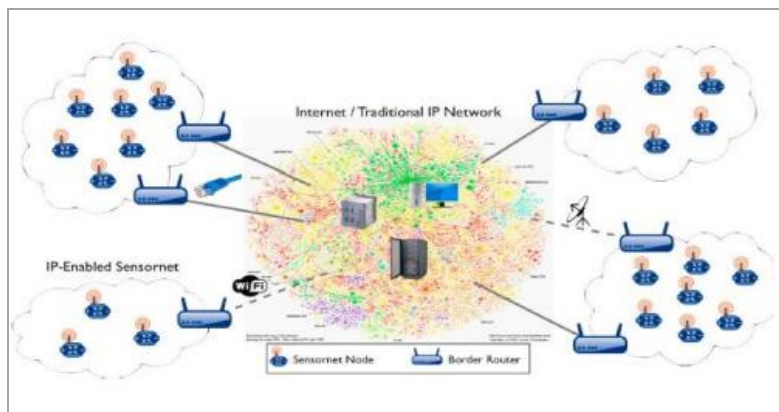
- Εύρεση Γειτονικού Σταθμού (Neighbor Discovery, ND)
- Ανίχνευση Διπλότυπης Διεύθυνσης (Duplicate Address Detection)
- Ανίχνευση δρομολογητή

Μέσω του IPv6 αποτυπώνεται η εξέλιξη της τεχνολογίας στον τομέα της επικοινωνίας μέσω του διαδικτύου. Ένας τομέας στον οποίο εδώ και χρόνια επικρατεί το Ethernet. Τα αποτελέσματα των ερευνών επί των Wireless Communications, οδηγούν στο συμπέρασμα πως δεν υστερούν σημαντικά έναντι του Ethernet. Έτσι μπορεί να υποστηριχθεί το επιχείρημα, πως πλέον είναι δύο οι τεχνολογίες οι οποίες μπορούν να ανταπεξέλθουν σε αντίστοιχες απαιτήσεις.

Στον αντίποδα ωστόσο υπάρχουν τα Προσωπικά Δίκτυα Ασύρματης Ζεύξης των οποίων όπως έχει ήδη αναλυθεί, χαρακτηριστικό είναι το να λειτουργούν σε συνθήκες χαμηλής κατανάλωσης ισχύος. Αναπτύχθηκε επομένως το πρότυπο 802.15.4, με σκοπό να εξυπηρετήσει τις ανάγκες αυτές, για μεγάλο χρονικό διάστημα αυτονομίας, αλλά και πλήθος από κόμβους χαμηλού κόστους. Με τα χαρακτηριστικά αυτά, ακόλουθος είναι ο περιορισμός αναφορικά με τους ρυθμούς ζεύξης των LoWPANs. Η ταχύτητα του ρυθμού μετάδοσης φτάνει έως τα 250 Kbps και ο όγκος των δεδομένων τα 128 Bytes, αφού οι μικροελεγκτές διαθέτουν μικρές δυνατότητες για την επεξεργασία και αποθήκευση των δεδομένων. Η διαφορά με τα υπόλοιπα προσωπικά ασύρματα δίκτυα, είναι πως τα προσωπικά ασύρματα δίκτυα χαμηλής ενέργειας λειτουργούν μέσω των Multiple Hops, με τους μικροελεγκτές να επιτυγχάνουν τη σύνδεσή τους σε διάφορους κόμβους χρησιμοποιώντας οκτώ 8 KB (KiloBytes) R.A.M και εξήντα τέσσερα 64 KB R.O.M.[31,33]

## 4.2 Δομή

Τα ασύρματα προσωπικά δίκτυα χαμηλής κατανάλωσης επικοινωνώντας με το πρωτόκολλο IP συνδέονται με άλλα αντίστοιχα δίκτυα μέσω δρομολογητών. Τα περιφερειακά μέρη του δικτύου να αποτελούν τα σημεία στα οποία τα LoWPANs λειτουργούν ως αυτόνομα δίκτυα. Συνδέσεις με άλλα IP δίκτυα μπορούν να υλοποιηθούν μέσω GPRS, Wi-Fi. Το πρωτόκολλο 6LoWPAN δύναται να λειτουργήσει αποκλειστικά με τη χρήση IPv6. Δεν είναι λίγες οι περιπτώσεις όπου είναι αναγκαία η επιλογή του ευρύτερου δικτυακού εξοπλισμού, έτσι ώστε να επιτευχθεί η προσαρμογή σε IPv4. Με τους δρομολογητές οι οποίοι εκτελούν διεργασίες στα άκρα, πραγματοποιείται η μεταφορά των δεδομένων στο επίπεδο δικτύου. και για το λόγο αυτό δεν συγκρατείται η κατάσταση του επιπέδου εφαρμογής. Η εικόνα 4.1 είναι ένα σχηματικό παράδειγμα της δομής αυτής [22,32,34]



Εικόνα 4.1: Δομή 6LoWPAN [23]

### 4.3 Επίπεδο Εφαρμογής

Η αρχιτεκτονική του πρωτοκόλλου 6LoWPAN είναι εκείνη που υποδεικνύει τον τρόπο σύνδεσης των δεδομένων με το πρότυπο 802.15.4 και αποτελεί τη βάση επιλογής των κυριότερων στοιχείων του επιπέδου εφαρμογής. Τα στοιχεία αυτά διαχωρίζονται σε τρία (3) βασικά μέρη :

- Ο τρόπος με τον οποίο πραγματοποιείται η συμπίεση της επικεφαλίδας (overhead). Στην προκειμένη περίπτωση τίθεται η υπόθεση πως γίνεται χρήση όμορης πληροφορίας. Το κομμάτι εκείνο της επικεφαλίδας, δύναται να παρακαμφθεί όταν μια ομάδα πληροφοριών προερχόμενη από το επίπεδο προσαρμογής, έχει την ευκαιρία να λάβει δεδομένα από το επίπεδο ζεύξης, υπό το πρίσμα της υπόθεσης ότι συναντάται μια ήδη γνωστή ομάδα δεδομένων.
- Ο τρόπος με τον οποίο πραγματοποιείται ο τεμαχισμός – κατακερματισμός. Στην περίπτωση αυτή αναλύεται ο τρόπος κατά τον οποίο οι ομάδες δεδομένων του IPv6, υποβάλλονται σε μια διαδικασία διαχωρισμού τους σε πλήθος πλαισίων του επιπέδου ζεύξης. Ο λόγος αυτής της διεργασίας είναι η κάλυψη του μικρότερου μεγέθους της μέγιστης μονάδας μετάδοσης (Maximum Transmission Unit).
- Ο τρόπος βάσει του οποίου πραγματοποιείται η διαδικασία υλοποίησης της μετάβασης στο επόμενο επίπεδο, με σκοπό τη συνεισφορά στην προώθηση των πλαισίων δεδομένων στο επίπεδο προσαρμογής. Με τον τρόπο αυτό δύναται να μεταβιβαστεί η εκάστοτε διεύθυνση στα πιο μακρινά σημεία ενός step μετάδοσης της IP. Μια άλλη σημαντική ορολογία είναι εκείνη της οποίας η εφαρμογή συναντάται στο σύνολο του επιπέδου προσαρμογής και φέρει την ονομασία “key”. Κατά τη διαδικασία αυτή παραβλέπονται ορισμένες από τις επικεφαλίδες του δικτύου, υπό το πρίσμα της εφαρμογής χωρίς τη χρήση όμορου περιεχομένου συμπίεσης.

Για τις τρεις (3) προαναφερθείσες καταστάσεις η μέθοδος της συμπίεσης μπορεί να πραγματοποιηθεί μέσα σε ένα μικρό αριθμό από Bytes. Δίνεται η δυνατότητα της περαιτέρω συμπίεσης άλλων σημείων των επικεφαλίδων σε μερικά και μόνο bits με την προϋπόθεση της παρουσίας ίδιων τιμών στις περιπτώσεις που εμφανίζονται λιγότερο γνωστά για τον προγραμματισμό overheads. Αυτές οι παρόμοιες τιμές κάνουν την εμφάνισή τους στις περιπτώσεις όπου γίνεται παρατεταμένα χρήση ενός υποσυνόλου λειτουργιών.

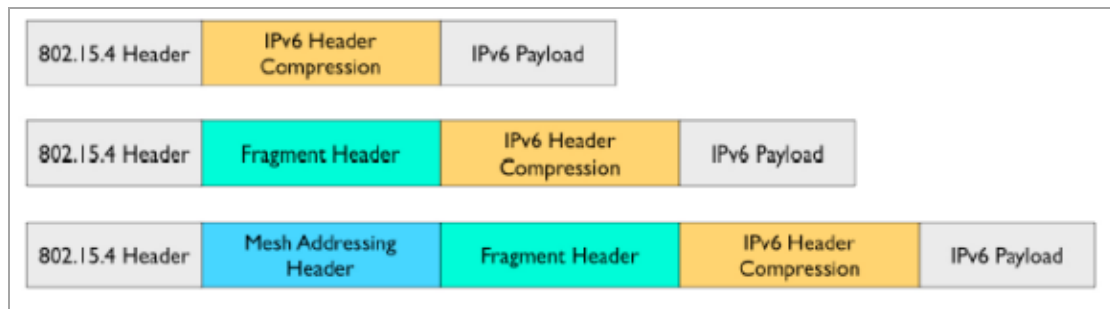
Οι γνωστές μέθοδοι συμπίεσης που αναφέρθηκαν θεωρούνται αξιόπιστες. Αυτό συμβαίνει λόγω του ότι δεν παρακάμπτουν όλες τις προηγούμενες αιτήσεις που έχουν υποβληθεί. Τουναντίον δίνεται ιδιαίτερη σημασία στην εξέλιξη και προώθηση των ροών αυτών, στα πλαίσια κάποιων πολύ στενών περιθωρίων, ενός υποτιθέμενου δικτύου. Κατά τη διάρκεια τη διεργασίας αυτής ωστόσο, υπάρχουν και κάποιες προϋποθέσεις. Μια τέτοια προϋπόθεση είναι η άμεση επικοινωνία μεταξύ του κόμβου συμπίεσης και του κόμβου αποσυμπίεσης, προκειμένου να επιτυγχάνεται άμεσα και

με την απαραίτητη ταχύτητα η αντίστοιχη συμπίεση και αποσυμπίεση. Μέσω αυτών των μεθόδων απλουστεύεται και βελτιώνεται η μετάδοση πληροφοριών μέσω μεγάλου μεγέθους ροών, προϋποθέτοντας όμως στα αρχικό στάδιο οι ομάδες δεδομένων να αποστέλλονται ασυμπίεστες, πράγμα το οποίο τις καθιστά απαγορευτικές για LoWPAN Networks.

Η κίνηση (Traffic), που συναντάται σε διάφορες εφαρμογές των δικτύων αυτών, περιλαμβάνει τις όχι και τόσο συχνά εμφανιζόμενες ειδοποιήσεις, αντί των τεμαχίων πληροφορίας. Όπως έχει αναφερθεί, η μεταφορά αυτή των δεδομένων απαιτείται να πραγματοποιείται σε τακτά χρονικά διαστήματα με την αντίστοιχη συμπίεση και αποσυμπίεση μεταξύ των υπεύθυνων κόμβων. Κάτι τέτοιο οδηγεί αυτομάτως στη μέθοδο της «αναδρομολόγησης», η οποία όμως προκειμένου να καταστεί εφικτή θα πρέπει έχει στη διάθεση της και το σύνολο της πληροφορίας από προηγούμενες ενέργειες. Κάτι τέτοιο είναι χρονοβόρο και επηρεάζει άμεσα τους δείκτες απόδοσης, με το πρωτόκολλο 6LoWPAN να δίνει τη λύση στο ζήτημα αυτό, καθώς κατά τα πρότυπά του, δεν απαιτεί κάποια πληροφορία προερχόμενη από τις προηγούμενες ενέργειες.[23,35,36]

#### 4.3.1 Μορφή Επικεφαλίδας

Μια από τις ενέργειες του προτύπου είναι και η συλλογή όλων των επικεφαλίδων (overheads), ούτως ώστε να απλουστευθεί και να γίνει γρηγορότερα, σε περίπτωση που χρειαστεί, η αναζήτηση στα χαρακτηριστικά του καθενός από αυτά και στα πλεονεκτήματα που μπορεί να προσφέρει, ανάλογα πάντα με τις απαιτήσεις. Παρέχεται η μέθοδος υποστήριξης της αναγνώρισης των τύπων των επικεφαλίδων, μέσω της χρήσης ενός μεγέθους δεδομένων στην αφετηρία του καθενός. Υποστηρίζεται επίσης η μετάφραση των δεδομένων των extensions overheads, τα επονομαζόμενα undeheads, με τη μέθοδο μετάδοσης στα δίκτυα πλέγματος, να πραγματοποιείται μέσω της μεταφοράς των δεδομένων στο δεύτερο επίπεδο. Έτσι με τη συγκέντρωση όλων των επικεφαλίδων επιτυγχάνεται το φιλτράρισμά τους και ακολουθεί η απόρριψη εκείνων οι οποίες στο τέλος κρίνονται ως μη χρήσιμες. Ομοίως στην περίπτωση των δικτύων πλέγματος γίνεται επίσης η απόρριψη ή παράκαμψη των επικεφαλίδων, όταν η μεταφορά της ομάδας των δεδομένων πραγματοποιείται μέσω μια μονής κατεύθυνσης διαδρομής, με αποτέλεσμα να λαμβάνεται η ταύτιση των επικεφαλίδων αφετηρίας και λήξης με εκείνες του επιπέδου ζεύξης. [31,34].



Εικόνα 4.2: Δομή Overhead [23]

### 4.3.2 Επικεφαλίδα Κατακερματισμού

Η Επικεφαλίδα Κατακερματισμού (Overhead Fragment) χρησιμοποιείται, στις περιπτώσεις όπου το χρήσιμο και ο φορτίο περισσεύει, εξαιτίας του όγκου των δεδομένων σε ένα και μόνο κομμάτι του προτύπου 802.15.4. Θα μπορούσε να παρομοιαστεί με την επικεφαλίδα κατακερματισμού ενός άλλου προτύπου, του IEEE1394. Σε αυτό εμπεριέχονται τρία ξεχωριστά χαρακτηριστικά:

- Ο εντοπισμός του πλαισίου
- Το υπόλοιπο πλαισίου
- Ο όγκος του πλαισίου

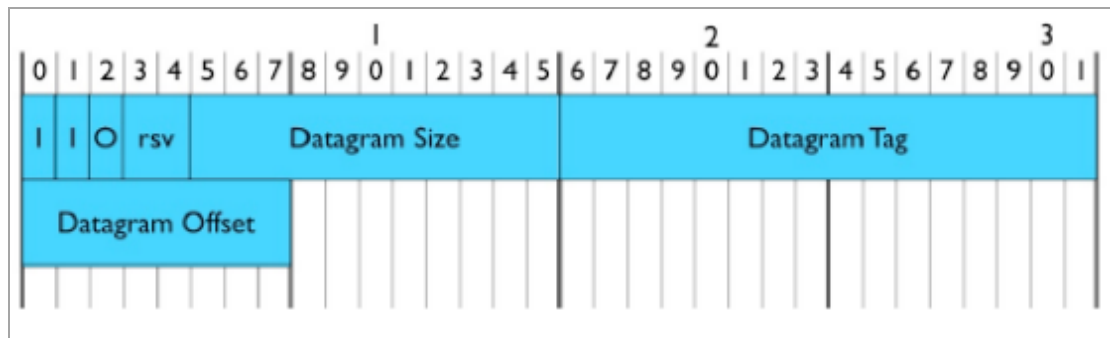
Η ανάλυση του καθενός εκ των τριών (3) αυτών είναι η εξής :

Μέσα από το χαρακτηριστικό του όγκου του πλαισίου, πραγματοποιείται ο ορισμός του αθροιστικού όγκου των δεδομένων που είναι ωφέλιμος. Ενεργοποιείται πριν τη διαδικασία του κατακερματισμού και εμπεριέχεται σε οποιαδήποτε ομάδα δεδομένων με σκοπό τη διευκόλυνση της αναζήτησης της παροδικής μνήμης από την πλευρά του πομπού προς τη πλευρά του δέκτη, όταν οι ομάδες αυτές των δεδομένων έχουν εκτροχιαστεί.

Ο εντοπισμός πλαισίου απαριθμεί και φέρνει στην επιφάνεια το άθροισμα των ομάδων δεδομένων που ανήκουν σε ένα επωφελές πλαίσιο. Χρησιμοποιείται για τη συλλογή των ομάδων δεδομένων που ανήκουν στο ίδιο επωφελές πλαίσιο.

Το υπόλοιπο πλαισίου επιτελεί τη λειτουργία του ορισμού του επωφελούς πλαισίου δεδομένων πριν από τη διαδικασία του κατακερματισμού και υπολογίζεται σε μεγέθη της τάξεως των 8 Byte. Στην περίπτωση που δινόταν η δυνατότητα υπολογισμού δίχως περιορισμούς το υπόλοιπο πλαισίου σε μεγέθη του ενός 1 Byte, τότε θα υπήρχε το αποτέλεσμα των επικεφαλίδων της τάξεως των 11 bit. Στα πλαίσια υπολογισμού της τάξεως των 8 Bytes παρουσιάζονται ανάγκες για οκτώ 8 bit. Στην εικόνα 4.3, γίνεται μια εκτενής παρουσίαση της μορφής της επικεφαλίδας που είναι της τάξης των 2 bit. Από εκείνα που παρουσιάζονται, το 3ο είναι εκείνο που επιτελεί τη λειτουργία της συμπίεσης του υπολοίπου του πλαισίου στην ομάδα δεδομένων και λαμβάνει μονίμως την τιμή 0. Η επικεφαλίδα κατακερματισμού είναι της τάξεως των τεσσάρων 4 Bytes και κάθε αρχική ομάδα δεδομένων της τάξεως των 5 Bytes για κάθε ένα που ακολουθεί. [34,38]





Εικόνα 4.3: Overhead κατακερμαστισμού [23]

### 4.3.3 Επικεφαλίδα Διευθυνσιοδότησης Πλέγματος

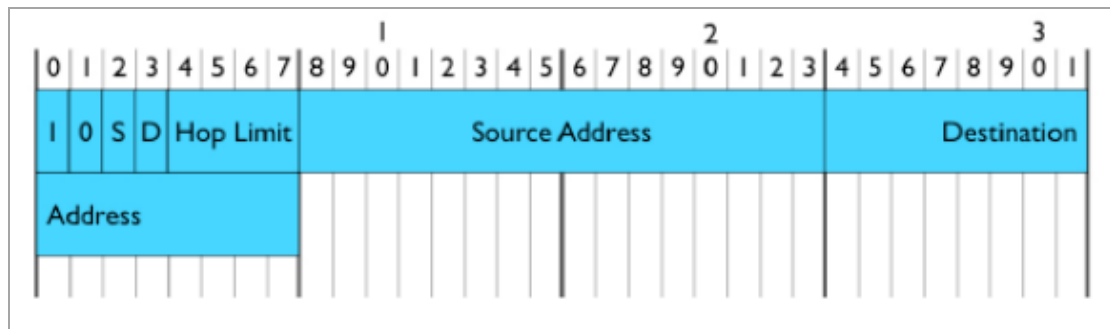
Η επικεφαλίδα αυτή καλείται να καλύψει το κενό της προώθησης των σημαντικών ομάδων δεδομένων με τις ανάλογες μεταβάσεις, ανάμεσα στο πλήθος των διαφόρων κόμβων και της μετάδοσής τους στο δεύτερο επίπεδο.

Εμπεριέχονται τρία (3) διαφορετικά επίπεδα, τα οποία είναι τα εξής :

- Το όριο μεταβάσεων
- Η διεύθυνση της πηγής
- Η διεύθυνση του προορισμού

Όσον αφορά το όριο μεταβάσεων , μπορεί να χαρακτηριστεί ως συναφές με το IPv6. Η εργασία που επιτελεί τη διαδικασία του περιορισμού των μεταφορών από ένα τον έναν κόμβο στον άλλο, υπό το πρίσμα της προώθησης μιας ομάδας δεδομένων. Η ομάδα αυτή των δεδομένων επιδέχεται μείωση της τάξης μεγέθους της 1 μονάδας για κάθε μεταφορά, με αποτέλεσμα κάποια στιγμή να λάβει και την τιμή μηδέν 0. Όταν πλέον λάβει την τελική αυτή τιμή, σημαίνει πως έχει φτάσει στον προορισμό και πλέον έρχεται αντιμέτωπη με την ακύρωση.

Τα επίπεδα της διεύθυνσης πηγής και της διεύθυνσης προορισμού έχουν την αρμοδιότητα να υποδεικνύουν τον πομπό και τον δέκτη κάθε τεμαχίου δεδομένων. Συναντώνται σε 2 εκδόσεις, είτε είναι μικρού είτε μεγάλου μεγέθους. Στην παρακάτω εικόνα 4.4 απεικονίζεται ο τύπος της Επικεφαλίδας Διευθυνσιοδότησης Πλέγματος. Θα πρέπει επίσης να σημειωθεί πως και οι 2 προαναφερθείσες εκδόσεις διευθύνσεων λαμβάνουν τον τύπο του προτύπου 802.15.4.[23,34,37]



Εικόνα 4.4: Overhead διευθυνσιοδότησης [23]

#### 4.3.4 Συμπίεση Επικεφαλίδας – RFC-4944

Η συμπίεση HC1 (Header Compression 1) επιτυγχάνεται μέσω του RFC-4944. Πρόκειται για μια δομή συμπίεσης η οποία δεν απαιτεί την αναγκαιότητα της ύπαρξης πληροφορίας από κάποια προηγούμενη ενέργεια και σε αναβαθμισμένη μορφή, για την υλοποίηση τοπικής σύνδεσης. Έτσι επιτυγχάνεται ο ορισμός του encoded Byte (κωδικοποιημένο Byte), το οποίο προηγείται της συμπιεσμένης επικεφαλίδας και αναλαμβάνει ρόλο στα πλαίσια των επικεφαλίδων των ανώτερων επιπέδων.

Μέσω του προτύπου 6LoWPAN πραγματοποιείται η παράλειψη ορισμένων εκ των πλαισίων, υπό την εικασία των γνωστών παραμέτρων. Στα πλαίσια μιας υπόθεσης, υλοποιείται η συμπίεση ενός 64 bit – προθέματος, για το σύνολο δύο διαδρομών μεταξύ αφετηρίας και κατάληξης μέσα σε μια τάξη μεγέθους του 1 bit, στην περίπτωση της συνύπαρξης σε όμορο δίκτυο. Επιτυγχάνεται επιπροσθέτως η συμπίεση του πλαισίου της επικεφαλίδας που ακολουθεί σε μια τάξη μεγέθους, ανάλογη των 2 bit, όταν από το πλαίσιο δεδομένων χρησιμοποιείται ένα εκ των πρωτοκόλλων:

- UDP
- TCP
- ICMPv6

Μια ακόμα λειτουργία είναι η συμπίεση των traffic class (τάξη κίνησης) και flow label (ετικέτα ροής), σε ένα πλαίσιο δεδομένων, στις περιπτώσεις που λαμβάνονται και στις δύο περιπτώσεις η τιμή 0. Σε κάθε μια από τις συμπιεσμένες αυτές επιλογές, εμπεριέχονται κατοχυρωμένες παράμετροι που οδηγούν στο συμπέρασμα πως τα πλαίσια δεδομένων μεταβιβάζονται σε μια διαδικασία οριοθέτησης υπό την πρόβλεψη πως δεν υπάρχει αντιστοίχιση.

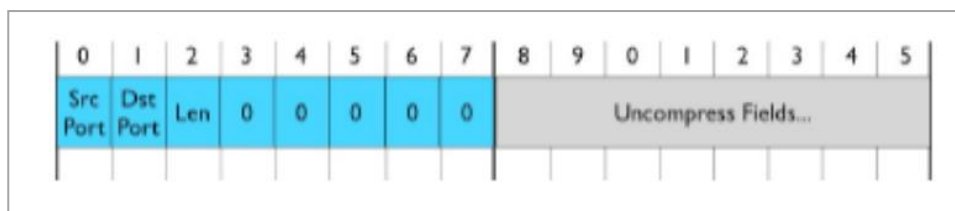
Το πρότυπο αυτό επίσης τείνει να παραβλέπει και άλλα εκ των πλαισίων δεδομένων μετατρέποντας σε χρήσιμες, τις μη χρήσιμες πληροφορίες στο ενδιάμεσο των

επιπέδων. Δύναται επίσης να λάβει την πληροφόρηση σχετικά με το μέγεθος του χρήσιμου όγκου δεδομένων από το πρότυπο 802.15.4 ή και την επικεφαλίδα κατακερματισμού.

Η διαδικασία κωδικοποίησης του HC1 αποτυπώνεται στην εικόνα 4.5. Πιο συγκεκριμένα το αρχικό Byte εκκίνησης, μέσω του οποίου ορίζεται το HC1 περιγράφεται. Με μια ακολουθία της τάξεως του μεγέθους των 8 bit, δηλώνεται ο τρόπος κατά τον οποίο έχει πραγματοποιηθεί η συμπίεση. Για την εκάστοτε διεύθυνση, γίνεται η χρήση 1 bit, προκειμένου να προσδιοριστεί το αν το εισαγωγικό κομμάτι του IPv6, χαρακτηρίζεται ως τοπικό και έχει γίνει η παράβλεψη ενός bit, για να αποδειχθεί αν ο εντοπισμός της διεπαφής, δύναται να έρθει ως αποτέλεσμα από τη διεύθυνση ζεύξης του προτύπου 802.15.4.

Γίνεται επίσης ο διαχωρισμός μεταξύ της κλάσης κίνησης και της ετικέτας ροής, σχετικά με το αν η τιμή τους είναι μηδέν, μέσω του T.F (Traffic Class) bit, προκειμένου να αποφασιστεί η ενδεχόμενη παράλειψή τους.

Όσον αφορά τα bit τα οποία ακολουθούν και υποδεικνύουν την ύπαρξη ενός εκ των τριών προτύπων από αποτυπώθηκαν παραπάνω ή επρόκειτο για μια διαδικασία συμπίεσης ή σειριακής μετάδοσης, αυτά αποκαλούνται ως bit επόμενης κεφαλίδας. Το αν η επικεφαλίδα που ακολουθεί έχει υποστεί τη διαδικασία της συμπίεσης, υποδεικνύεται μέσω του HC2 (Header Compression) bit. Πρέπει να σημειωθεί πως το όριο πλαισίου είναι το μόνο το οποίο δεν υποβάλλεται σε διαδικασία συμπίεσης, αλλά κατά τη μεταφορά του συνεχίζει αυτούσιο.[33,38]



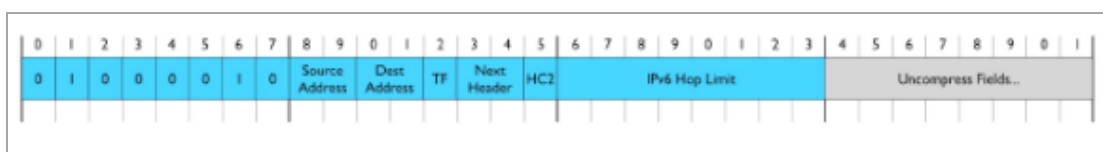
Εικόνα 4.5: HC1 Encoding [23]

### 4.3.5 Συμπίεση Προθέματος UDP (RFC-4944)

Οι αλλαγές του προθέματος UDP σχετικά με την ελάττωση όγκου, πραγματοποιούνται επιτυχώς μέσω της μεθόδου RFC-4944. Πιο συγκεκριμένα, στον καθορισμό του HC2 bit κατά τη διάρκεια της διαδικασίας κωδικοποίησης του αντίστοιχου HC1, αμέσως γίνεται η προσθήκη της τάξης μεγέθους από οκτώ bit. Εκείνα οποία έπονται αυτών της κωδικοποίησης, όπου μέσω των οποίων ορίζεται ο τρόπος σύμφωνα με τον οποίο θα υλοποιηθεί η συμπίεση της επικεφαλίδας UDP.

Υπάρχει ένας συγκεκριμένος αριθμός από θύρες που προσφέρονται από το πρωτόκολλο 6LoWPAN και συμβάλλουν στην αποτελεσματική συμπίεση των UDP θυρών (61616 – 61631).

Σε μια περίπτωση που μια εκ των θυρών αυτών τύχει να συμπίπτει μαζί με μια UDP θύρα, τότε τα αρχικά 12 bit, τότε μπορούν να παρακαμφθούν. Επιπλέον στην περίπτωση που και οι δύο θύρες (source – portal) συμπίσουν στις παραπάνω, το πλαίσιο κάτω από το οποίο καθορίζονται, επιδέχεται συμπίεσης και λαμβάνει την τάξη μεγέθους του 1 Byte. Μέσω του HC2, δίνεται η δυνατότητα για παράβλεψη του μεγέθους του UDP, με την προοπτική ότι δύναται να ληφθεί από το πλαίσιο του ωφέλιμου φορτίου του πρωτοκόλλου IPv6. Στην εικόνα 4.6 που ακολουθεί, αποτυπώνεται η ανωτέρω διαδικασία κωδικοποίησης.[31,34]



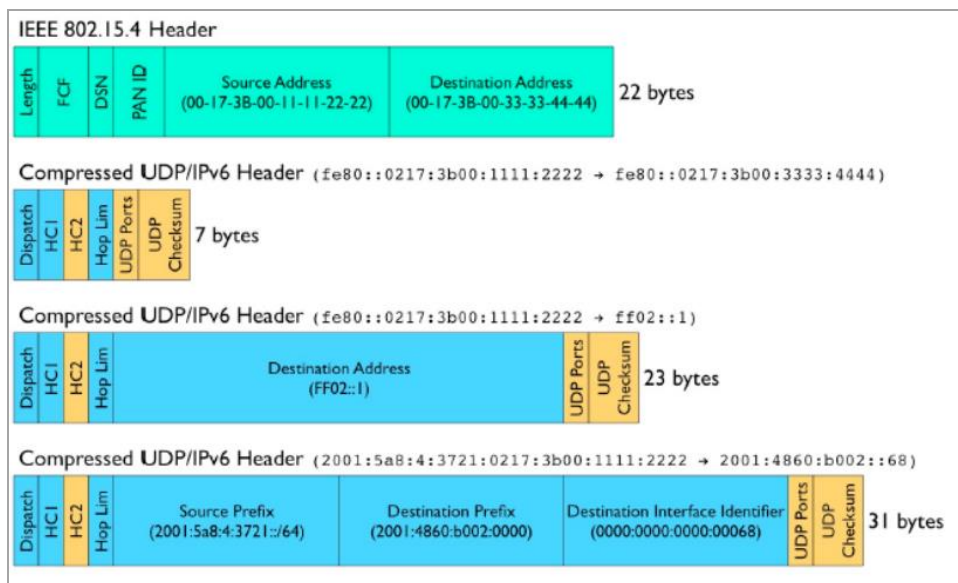
Εικόνα 4.6: Κωδικοποίηση HC2 [23]

### 4.3.6 Παραδείγματα Συμπίεσεων

Στην εικόνα 4.7 παρατίθεται ένα παράδειγμα στα πλαίσια των υπηρεσιών διαφοροποίησης της επικεφαλίδας, με σκοπό το HC1-encoding και HC2-encoding. Όσον αφορά τη διαδικασία συμπίεσης λαμβάνεται το βέλτιστο αποτέλεσμα στις περιπτώσεις όπου υπάρχει σύνδεση με ένα τοπικό δίκτυο, και κατά τη διαδικασία της αποστολής υπάρχει μόνο ένας δέκτης. Στην περίπτωση αυτή δύναται να υλοποιηθεί μια συνολική διαδικασία συμπίεσης αναφορικά με τις επικεφαλίδες UDP, με τις HC1 – HC2 να τα τοποθετούν μαζί σε μια τάξη μεγέθους των 7 bytes.

Το RFC – 4944 δεν επιτελεί χωρίς απώλειες τη διαδικασία συμπίεσης όταν καλείται να λειτουργήσει στα πλαίσια ενός τοπικού δικτύου ή να αποστείλει έναν όγκο δεδομένων με μεγάλο αριθμό δεκτών. Η διαδικασία προβλέπει πως κάθε αρχικό στοιχείο δεδομένων πέραν του τοπικού, είναι αναγκαίο να μεταβιβαστεί με την παρούσα μορφή του. Στην Εικόνα 4.7 που ακολουθεί απεικονίζεται η συμπίεση σε μια τάξη μεγέθους περί των 23 Bytes από τα HC1 – HC2 με σκοπό την επίτευξη του βέλτιστου αποτελέσματος.

Σε κάθε μορφής σύνδεση εκτός του δικτύου LoWPAN το πρόθεμα όπως και η IPv6 διεύθυνση είναι απαραίτητο να μην τροποποιηθούν και να συνεχίσουν χωρίς συμπίεση.[32,36]



Εικόνα 4.7: Compression Examples [23]

### 4.3.7 Βελτιωμένη Συμπίεση Προθέματος - UDP/IPV6

Προκειμένου να επιτευχθεί κατά τον βέλτιστο τρόπο η διαδικασία συμπίεσης, εισερχόμενη σε ένα πλαίσιο επιλογών, το πρότυπο που αναλύεται διαθέτει αυτοματοποιημένες διαδικασίες συμπίεσης – αποσυμπίεσης. Αυτές οι μέθοδοι αποτυπώνονται στο HC πρότυπο. Μέσω αυτού ορίζεται η εκ νέου διαδικασία κωδικοποίηση επικεφαλίδας του IPv6 και το IPHC. Μέσω του προτύπου αυτού, δίνεται η άδεια στις επικεφαλίδες της κλάσης κίνησης και της ετικέτας ροής μεμονωμένα, να υπεισέρχονται σε μια νέα διαδικασία συμπίεσης. [35]

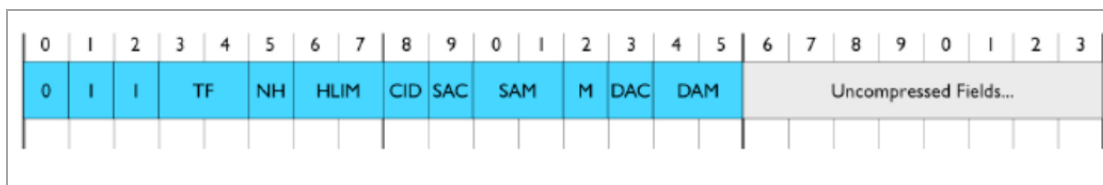
Η αρμοδιότητα του Context, αφορά το σύνολο των κόμβων που παρευρίσκονται μέσα σε ένα δίκτυο το οποίο λειτουργεί στα πλαίσια χαμηλής κατανάλωσης ενέργειας. Ο εντοπισμός του πλαισίου γίνεται από το πρότυπο IPHC με τη χρήση ενός ευρετηρίου της τάξης μεγέθους περί τα 4 bit, δίνοντας έτσι την ευκαιρία για λειτουργία σε έως και 16 ξεχωριστά πλαίσια εντός του δικτύου. Στην περίπτωση που μια IPv6 διεύθυνση είναι ίδια με το ήδη υπάρχον πρόθεμα σε ένα από τα πλαίσια, τότε ενεργοποιείται η διαδικασία συμπίεσης από το πρότυπο IPHC.

Πρέπει να σημειωθεί πως σχετικά με τα frame prefixes (προθέματα πλαισίου) δεν υπάρχει περιορισμός αναφορικά με αυτά εντός ενός LoWPAN. Τουναντίον μπορούν να λάβουν μια όποια ανεξάρτητη τιμή. Έτσι εξάγεται το συμπέρασμα, πως τα πλαίσια αυτά έχουν τη δυνατότητα να μπορούν να προβαίνουν στις απαραίτητες διεργασίες με σκοπό την επεξεργασία των κόμβων του δικτύου, ούτως ώστε να κατορθώσουν τη συμπίεση του frame prefix και στην αφετηρία όπως και στην κατάληξη, ακόμα και στις περιπτώσεις που εκείνα επιχειρούν συνδέσεις και με κόμβους εκτός δικτύου. [31]

Η απεικόνιση μιας βελτιστοποιημένης διαδικασίας συμπίεσης– αποσυμπίεσης παρουσιάζεται στην Εικόνα 4.8. Εδώ παρατηρούνται μέσα από το γενικότερο σύνολο, τα αρχικά 3 bit τα οποία είναι εκείνα που υποδεικνύουν τη μορφή της επικεφαλίδας και τη μορφή λειτουργίας του IPHC. Όπως έχει ήδη αναφερθεί, τα στοιχεία traffic class όπως και η ετικέτα ροής χαρακτηρίζονται από τα TF bit σχετικά με το αν είναι συμπιεσμένα.

Στην περίπτωση των επομένων bit, που βρίσκονται ανάμεσα στον αριθμό 8 και τον αριθμό 15, γίνεται αναφορά στους τρόπους με τους οποίους επιτυγχάνεται η διαδικασία συμπίεσης σχετικά με τις source address (διεύθυνση αποστολέα) και destination address (διεύθυνση προορισμού). Στην περίπτωση όπου το CID (Context Identifier) έχει λάβει μηδενική τιμή τότε ορίζεται ως προεπιλεγμένο συμπιεσμένο πλαίσιο αναφορικά με τις δύο διευθύνσεις. Η συγκεκριμένη διαδικασία επιτελείται κατά τα διαστήματα που και οι δύο παρευρίσκονται εντός του ίδιου δικτύου.

Στον αντίποδα που το CID δεν έχει μηδενική τιμή, αλλά έχει λάβει την τιμή 1, υπάρχει η προσθήκη στην επικεφαλίδα δυο παραπάνω ομάδων, στις οποίες αναφέρεται σε ποιο πλαίσιο γίνεται αναφορά, για τη διεύθυνση πηγής και διεύθυνση προορισμού αντίστοιχα.[32,37].

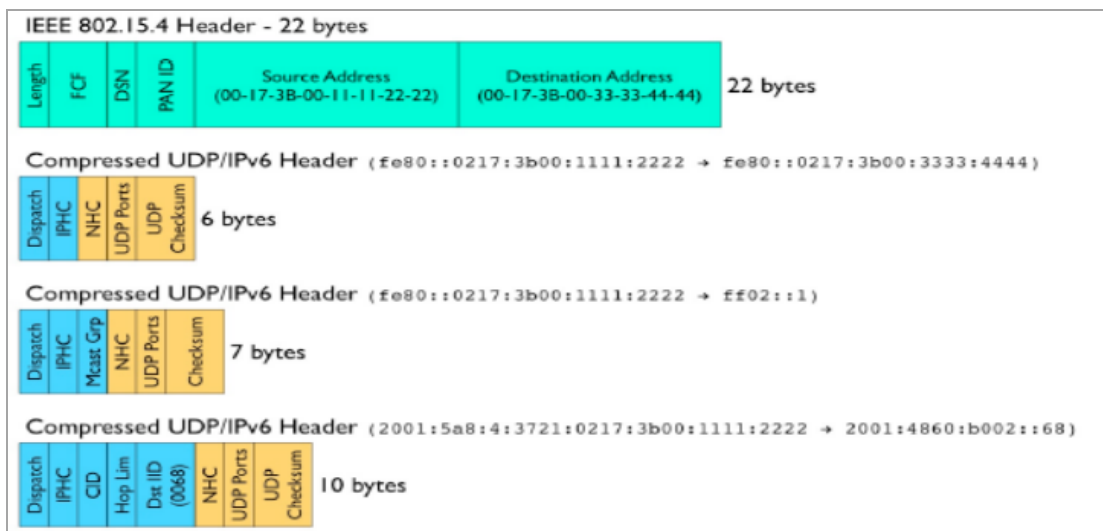


**Εικόνα 4.8: Βελτιστοποιημένη κωδικοποίηση [23]**

#### 4.3.8 Βελτιωμένα Παραδείγματα Συμπίεσης UDP/IPv6

Η Εικόνα 4.9 απεικονίζει τις αλλαγές με βάση τα πρότυπα IPHC (IP Header Compression) και NHC (Next Header Compression). Με την ίδια μεθοδολογία που ακολουθήθηκε για το πρότυπο RFC-4944, η βέλτιστη επιλογή που μπορεί να γίνει, αναφορικά με την απόδοση της διαδικασίας συμπίεσης είναι αυτή κατά την οποία εκτελείται αποστολή με έναν μόνο δέκτη στο τοπικό δίκτυο.

Οι κινήσεις αναβάθμισης που πραγματοποιήθηκαν στο πρότυπο IPHC εμφανίζονται στις περιπτώσεις σύνδεσης με τη μορφή multicast και για συνδέσεις εκτός local network. Στην Εικόνα 4.9 γίνεται επίσης μια αποτύπωση των διεργασιών αυτών με την απεικόνιση των μεγεθών των συμπίεσεων και των επεξεργασιών που γίνονται στα πλαίσια της εκμετάλλευσης και της ορθής χρήσης του προτύπου IPHC. [33].



Εικόνα 4.9: Examples of improved compression [23]

#### 4.4 IPv6/6LoWPAN - Αρχιτεκτονική

Σύμφωνα με το πρότυπο IPv6/6LoWPAN, προσδιορίζεται ο τρόπος με τον οποίο αναπαρίσταται ο κατακερματισμός, η συμπίεση και η προώθηση στο δεύτερο επίπεδο του μοντέλου OSI σε ένα πακέτο δεδομένων με βάση το πρότυπο 802.15.4. Ωστόσο, οι εξαρτήσεις του 6LoWPAN, οι οποίες ορίζονται σε συγκεκριμένες λειτουργίες θεωρούνται ελάχιστες. Δεδομένου αυτού υποστηρίζονται σχεδόν όλα τα πρωτόκολλα σε επίπεδο MAC που χρησιμοποιούνται στο πλαίσιο δεδομένων 802.15.4. Η δομή του πρωτοκόλλου 6LoWPAN δεν εξηγεί το πώς λειτουργίες του πρωτοκόλλου, IPv6, όπως το Neighbor Detection και την Αυτόματη Διαμόρφωση Διεύθυνσης Πηγής (Stateless Address Autoconfiguration) ενσωματώνονται για να διαμορφώσουν το LoWPAN και να επιτύχουν συνοχή με το στρώμα προσαρμογής. [38,40].

##### 4.4.1 IEEE 802.15.4 στην πράξη

Το πρότυπο IEEE 802.15.4 παρουσιάζει αρκετά πρακτικά θέματα που έχουν σημαντικές επεμβάσεις στην αρχιτεκτονική, πέραν του πρωτοκόλλου 6LoWPAN. Τα πρωτόκολλα δικτύου είναι αναγκαίο να ανταποκρίνονται σε απροσδόκητες καταστάσεις που προκύπτουν, μέσω της χρήσης πολλαπλών μεταβάσεων (multi-hops), οι οποίες βελτιώνουν την εμβέλεια και μπορούν έτσι να συνεισφέρουν στην αποφυγή εμποδίων κατά τη μετάδοση των δεδομένων. Ένα LoWPAN δίκτυο συνήθως δεν λειτουργεί όπως ένα απλό δίκτυο εκπομπής. Η ποιότητα σύνδεσης μεταξύ δύο κόμβων είναι συχνά περίπλοκη και μπορεί να μεταβάλλεται με τον χρόνο, εξαιτίας περιβαλλοντικών συνθηκών. Σε αυτό το σημείο, κρίνεται αναγκαία η αναφορά στα σχήματα αναμετάδοσης (hop-by-hop), τα οποία βοηθούν στη βιώσιμη μετάδοση μεταξύ συνδέσεων 802.15.4. Σχετικά με τις συνδέσεις, παρουσιάζεται το γεγονός ότι υποφέρουν από απώλειες. Ωστόσο, μόνο μέσω των σχημάτων αυτών δεν δύναται να

εξασφαλιστεί η αποτελεσματική μετάδοση δεδομένων. Συνήθως, συνδέσεις που θεωρούνται καλές, δηλαδή που έχουν υψηλό ποσοστό επιτυχίας στη μετάδοση δεδομένων, μπορεί να αντιμετωπίζουν περιοδικά προβλήματα που οφείλονται σε αλλαγές στο επίπεδο θορύβου και παρασιτικές παρεμβολές. Αυτά τα χρονικά διαστήματα απωλειών μπορούν να είναι προσωρινά και να επηρεάζουν την αξιοπιστία της σύνδεσης, αλλά συνήθως δεν αποτελούν μόνιμο πρόβλημα, αφού η δρομολόγηση είναι σε θέση να τα αντιμετωπίσει, επιλέγοντας διαφορετικές διαδρομές, κατά τη διάρκεια προώθησης πακέτων. Για την ακρίβεια, η δρομολόγηση λειτουργεί επιλέγοντας δυναμικά τον επόμενο κατάλληλο κόμβο για μετάδοση, λαμβάνοντας υπόψη την ποικιλομορφία του παραλήπτη. Προκειμένου να αντιμετωπιστούν επιτυχώς οι προκλήσεις των συνδέσεων, το επίπεδο δικτύου χρειάζεται περαιτέρω ορατότητα σε μια πιο λεπτομερή συμπεριφορά κάθε σύνδεσης, ώστε να μπορεί να δημιουργήσει και να διατηρήσει αποτελεσματικές δομές δρομολόγησης.

Πολλές εφαρμογές LoWPAN, έρχονται αντιμέτωπες με μεγάλη κινητικότητα συσκευών μέσα στο δίκτυο, προκαλώντας έτσι, αλλαγές στις συνδέσεις τους ανά τακτά χρονικά διαστήματα. Αυτές οι αλλαγές μπορεί να προκύπτουν είτε από την κινητικότητα των συσκευών εντός του δικτύου, είτε από τις διακυμάνσεις των περιβαλλοντικών συνθηκών. Για παράδειγμα, στη παρακολούθηση πακέτων μπορεί να συμπεριληφθεί η μετακίνηση κινητών συσκευών, σε σχέση με έναν αριθμό σταθερών συσκευών. Ωστόσο, δεν είναι βέβαιο ότι θα υπάρχει φορητότητα σε επίπεδο διεύθυνσης IP, διότι οι συσκευές ενδέχεται να παραμένουν στο τοπικό δίκτυο ακόμα και όταν μετακινούνται, παραμένοντας συνδεδεμένες σε αυτό. Παρόλα αυτά, τέτοιες αλλαγές απαιτούν την προσαρμογή της δρομολόγησης σε κάθε νέο πρόβλημα συνδεσιμότητας που προκύπτει.

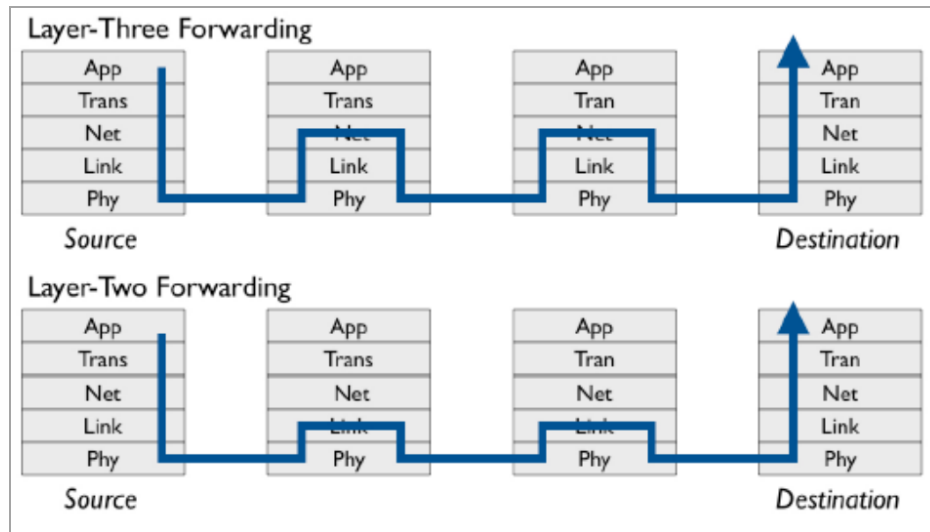
Το πλήθος των εμπορικών εφαρμογών και των βιομηχανικών προτύπων, που βασίζονται στο πρότυπο 802.15.4, καθορίζουν τους μηχανισμούς διαχείρισης ενέργειας κατά τον καθορισμό των πρωτοκόλλων δρομολόγησης. Έτσι, προκειμένου να παραμείνουν χαμηλά τα επίπεδα ενέργειας, πρέπει αρχικά να οριστεί το χρονικό διάστημα κατά το οποίο θα μεταδίδονται πληροφορίες από τους κόμβους. Αυτό απαιτεί συγχρονισμό τόσο του πομπού όσο και του δέκτη, ώστε να επικοινωνούν με συγκεκριμένο τρόπο και για συγκεκριμένο χρονικό διάστημα. Οι πιο συνήθεις πρακτικές, για αυτό συμπεριλαμβάνουν μεθόδους δειγματοληψίας ή προγραμματισμού. [37,38]

#### 4.4.2 Τύπος Δρομολόγησης

Γενικώς υφίστανται δύο βασικά ερωτήματα για την λειτουργία του IPv6 στο LoWPAN, σχετικά με το πως οι παράγοντες επιπέδου ζεύξης ενημερώνουν την δρομολόγηση, όπως και το σε ποιο επίπεδο πραγματοποιείται η προώθηση των πακέτων μέσα στο LoWPAN. Συνήθως, στο επίπεδο δικτύου πραγματοποιείται η δρομολόγηση IP χωρίς όμως να υπάρχει κάποια επιρροή από τις υποκείμενες συνδέσεις όπου υλοποιούνται οι ανεξάρτητες μεταβάσεις. Το 6LoWPAN χαρακτηρίζεται ως στρώμα προσαρμογής ανάμεσα στο επίπεδο ζεύξης και το δίκτυο.



Αντίστοιχη αναπαράσταση πραγματοποιείται μέσω των εικόνων 4.2 και 4.3 . Στην Εικόνα 4.10 αποτυπώνεται η δρομολόγηση στο επίπεδο δικτύου και ζεύξης.



**Εικόνα 4.10: Δρομολόγηση στο επίπεδο δικτύου και ζεύξης [23]**

Στο LoWPAN, ένα δίκτυο τύπου πλέγματος, κατά τη φάση της οργάνωσης, παρατηρείται πως η στοίβα δικτύου δεν πραγματοποιεί καμία δρομολόγηση IP δικτύου. Αντιθέτως παρατηρείται από το στρώμα προσαρμογής η κάλυψη της πλήρους εκπομπής στο φυσικό επίπεδο. Με αυτό τον τρόπο η δρομολόγηση γίνεται με διαφάνεια και οδηγούνται τα πλαίσια καταλλήλως στο εσωτερικό του LoWPAN. Κατά την αντιγραφή της συνολικής εκπομπής (broadcast), η στοίβα δικτύου με την βοήθεια των πρωτοκόλλων Ipv6 παρουσιάζει συμβατότητα μιας και τα IPv6 περιμένουν αντίστοιχη συμπεριφορά κατά την διάρκεια της επικοινωνίας. Για την πραγματοποίηση των τοπολογιών πλέγματος, θα πρέπει να πραγματοποιηθούν πολλαπλές μεταβάσεις, καθώς και σε κάθε κόμβο συνολικά του LoWPAN, θα πρέπει να δοθούν πακέτα από την τοπική ζεύξη multicast. Για να μπορέσει να εκτελεσθεί αποτελεσματικά η πλεγματοποίηση, οι μηχανισμοί σχηματισμού, διατήρησης και διάγνωσης δρομολόγησης IP, είναι αναγκαίο να οριστούν εκ νέου στο επίπεδο ζεύξης.

Υπάρχει επίσης η περίπτωση όπου κάθε κόμβος δύναται να εκτελεί λειτουργία δρομολογητή IP, με την κατάσταση αυτή να περιγράφεται ως δρομολόγηση από ψηλά. Εδώ, παρέχεται υποστήριξη στους μηχανισμούς με τους οποίους επιτελείται προώθηση στο επίπεδο δικτύου, οι οποίοι με τη σειρά τους μπορούν εκτελέσουν επιλογές που καθορίζονται από το IP για το επίπεδο δικτύου, τις λειτουργίες διαχείρισης και διαμόρφωσης.

Δίνεται επίσης η δυνατότητα σε πρωτόκολλα δρομολόγησης IP στην εργασία, με την προοπτική χρήσης μεγαλύτερου αριθμού τεχνολογιών ζεύξης, επιτρέποντας κατά αυτόν τον τρόπο τη βέλτιστη ενσωμάτωση σε πιο πολλά δίκτυα. [39,41].

#### 4.4.3 Διευθυνσιοδότηση -Αυτόματη Διαμόρφωση

Κάθε εξυπηρετητής μέσα από την αυτόματη διαμόρφωση διεύθυνσης πηγής (SAA), έχει τη δυνατότητα να δημιουργήσει μια μοναδική τοπική διεύθυνση IPv6 αποκλειστικής μετάδοσης (unicast), χρησιμοποιώντας είτε τη διεύθυνση IEEE EUI-64 είτε τη διεύθυνση 16-bit ή και τα δύο. Η επικοινωνία εντός του LoWPAN, ακόμη και όταν εμπλέκονται πολλαπλές μεταβάσεις, μπορεί να πραγματοποιηθεί μέσα από μία τοπική διεύθυνση ζεύξης με απαραίτητη προϋπόθεση μια δρομολογημένη διεύθυνση επικοινωνίας εκτός του τοπικού δικτύου. Η τοπική διεύθυνση σύνδεσης επαρκεί μόνο για την επικοινωνία με κοντινούς κόμβους έχοντας απευθείας επικοινωνία με τον πομπό. Στην περίπτωση όπου χρειαστεί μετάδοση πέρα από αυτούς τους κόμβους, απαιτείται η χρήση μιας δρομολογημένης διεύθυνσης.

Για τη βελτιστοποίηση της αποτελεσματικότητας, συνιστάται ο διαχωρισμός όλων των Unicast διευθύνσεων από τις διευθύνσεις ζεύξης του 802.15.4. Λόγω της σύνδεσης ανάμεσα στο 6LoWPAN, της ζεύξης και της επικεφαλίδας IP, υπάρχει η δυνατότητα παράλειψης της IP διεύθυνσης. Εκείνη προκύπτει από την διεύθυνση ζεύξης με αποτέλεσμα τη μη περεταίρω ανάλυση της. Ομοίως, είναι απαραίτητη η διαμόρφωση της διευθυνσιοδότησης διεπαφής, μέσω ενός κοινού προθέματος, έτσι ώστε μέσα από την αυτόματη διαμόρφωση να συμπιεστεί η επικεφαλίδα του 6LoWPAN και ως εκ τούτου να μειωθεί το μήκος της συνολικά. [37,41].

#### 4.4.4 Εύρεση γειτονικών σταθμών

Ένας κόμβος μπορεί να χρησιμοποιήσει τη λειτουργία εύρεσης γειτονικών σταθμών IPv6 (ND), για να βρει τους γειτονικούς σταθμούς, να αποθηκεύσει πληροφορίες προσβασιμότητας, να επεξεργαστεί προεπιλεγμένες διαδρομές και να συμβάλλει στη μεταφορά παραμέτρους διαμόρφωσης. Δεδομένου ότι η λειτουργία ND προορίζεται για τη διευκόλυνση της επικοινωνίας μεταξύ γειτονικών κόμβων, έχει ρυθμιστεί να λειτουργεί μέσω συνδέσεων με το πρωτόκολλο IP. Το ND στέλνει αιτήματα σε όλους τους κοντινούς κόμβους προκειμένου να επιλύσει τις διευθύνσεις και να ανιχνεύσει τους γειτονικούς σταθμούς που δεν είναι προσβάσιμοι.

Εκτός από την εύρεση γειτονικού σταθμού, οι κόμβοι χρησιμοποιούν την λειτουργία ND για την εύρεση διπλότυπης διεύθυνσης (DAD). Όλες οι επικοινωνίες πραγματοποιούνται στο τοπικό επίπεδο ζεύξης. Παρόλα αυτά, η χρήση των προδιαγραφών ND στα δίκτυα LoWPAN παρουσιάζουν ιδιαίτερες απαιτήσεις. Ειδικότερα προκειμένου να εντοπίσει για κάθε κόμβο την διεύθυνση, εκμεταλλεύεται συνεχώς το multicast. Με αυτό τον τρόπο το κόστος καθώς και η κίνηση στο δίκτυο αυξάνονται. Είναι σημαντικό να αναφερθεί πως εάν γίνει χρήση για μικρά δίκτυα είναι φερέγγυο, ενώ εάν χρησιμοποιηθεί σε μεγάλα δίκτυα πλέγματος με πολλαπλές μεταβάσεις (multihops) υπάρχουν μεγάλες χρονικές παρεκκλίσεις. [23,38,42].

#### 4.4.5 Mobile Ad-hoc NETWORK - MANET

Η κατάσταση κάθε κόμβου σε ένα δίκτυο περιορίζεται από τις ελάχιστες καθορισμένες δυνατότητες μνήμης και επικοινωνίας, οι οποίες περιορίζουν τις πληροφορίες δρομολόγησης που μπορούν να κοινοποιηθούν. Αυτοί οι περιορισμοί καθιστούν μη πρακτικό, ο εκάστοτε κόμβος να βασίζεται αποκλειστικά σε πληροφορίες κατάστασης επιπέδου σύνδεσης για πρωτόκολλα δρομολόγησης. Επιπλέον, τα παραδοσιακά διανυσματικά πρωτόκολλα απόστασης που έχουν σχεδιαστεί για ασύρματα κινητά δίκτυα κόμβων (Mobile Ad-hoc NETWORK , MANET), δεν είναι κρίνονται ως κατάλληλα για LoWPAN δίκτυα λόγω των διαφορετικών χαρακτηριστικών φορητότητας των κόμβων. Στην πρώτη περίπτωση, όλοι οι κόμβοι είναι φορητοί, ενώ στην δεύτερη υπάρχουν ένας συνδυασμός φορητών και σταθερών κόμβων. Επομένως, τα πρωτόκολλα MANET βασίζονται στη συνεχή ροή εντός του δικτύου για την ανακάλυψη και τη διατήρηση διαδρομών, κάτι που δεν είναι συμβατό με τους πιο αργούς ρυθμούς μετάδοσης των αντίστοιχων LoWPAN. Τα τελευταία χρησιμοποιούν πρωτόκολλα δρομολόγησης που βασίζονται σε ελλιπείς πληροφορίες δικτύου, αποδεικνύοντας την ικανότητά τους να ανέχονται την μη συνεχή ροή. [34,35]

#### 4.4.6 Ασφάλεια

Η χρήση των διεργασιών ασφαλείας AES-128 που προσφέρονται από το 802.15.4 είναι μια βασική πτυχή για το 6LoWPAN. Στα δίκτυα 6LoWPAN υπάρχει η δυνατότητα να ενσωματώνονται διεργασίες ασφαλείας επιπέδου μεταφοράς. Παρόλα αυτά, ενώ οι διεργασίες αυτές επιπέδου δικτύου όπως το IPsec και το Secure Neighbor Discovery (SND) αποτελούν προϊόν μελέτης, τα δίκτυα 6LoWPAN επί του παρόντος είναι σε ερευνητικό στάδιο. [36,39].

## ΚΕΦΑΛΑΙΟ 5ο : Contiki COOJA

Στην αυτήν την διπλωματική εργασία αναπτύσσεται, μέσω της χρήση του προγράμματος Contiki COOJA, τρία ξεχωριστά σενάρια προσομοίωσης. Αρχικά όμως, θα πρέπει πρώτα να εξεταστεί το περιβάλλον της εφαρμογής στο οποίο καλείται ο εκάστοτε χρήστης να εργαστεί. Πρέπει να δοθεί σημασία, στις λειτουργίες και στις δυνατότητες που προσφέρονται μέσω της εφαρμογής, ούτως ώστε ο μελλοντικός αναγνώστης μέσω της παρούσας εργασίας, να έχει τη δυνατότητα να τη χρησιμοποιήσει σαν οδηγό για μελλοντική έρευνα. Σε κάθε περίπτωση σε αυτό το κεφάλαιο πραγματοποιείται αυτού του είδους τη ανάλυση με την παράθεση ακόμα των πιο γενικών πληροφοριών που αφορούν την εφαρμογή.

### 5.1 Προέλευση

Μέσα στα πλαίσια της ανάπτυξης της τεχνολογίας, το RISE SICS (Swedish Institute of Computer Science, Σουηδικό Ινστιτούτο Επιστήμης Υπολογιστών), πραγματοποίησε μια επιστημονική έρευνα αναφορικά με τα Ασύρματα Δίκτυα Αισθητήρων. Η έρευνα αυτή περιελάμβανε ένα θεωρητικό μέρος το οποίο αποτέλεσε την απαρχή των ερευνών, αλλά και τη δημιουργία ενός μοντέλου, βάσει του οποίου υπολογιζόταν ο χρόνος ζωής του δικτύου πάνω σε πραγματικές συνθήκες, κυρίως μάλιστα πάνω σε βιομηχανικό περιβάλλον.

Στα πλαίσια λοιπόν της αναζήτησης, το Ινστιτούτο δημιούργησε το λειτουργικό σύστημα Contiki. Ένα κατά βάση χαμηλών απαιτήσεων λειτουργικό με επίκεντρο στα ασύρματα δίκτυα των αισθητήρων και πιο συγκεκριμένα, τους κόμβους αυτών.

Ωστόσο υπήρξε μια επέκταση της εφαρμογής αυτής. Αυτή ήταν το COOJA Simulator το οποίο και προοριζόταν ως προσομοίωση του Contiki OS (Operating System). Το γεγονός πως ένα τέτοιο σύστημα πρέπει να έχει στον καμβά των δυνατοτήτων του ένα μεγάλο πλήθος επιλογών για πειραματισμό και ανάπτυξη εικονικών δικτύων, πυρήνα κατά την εξέλιξή του αποτέλεσε η έννοια της επεκτασιμότητας.

Στη κύρια εκδοχή του λειτουργικού, τρέχουν οι γνωστές υλοποιήσεις σχετικά με τα ασύρματα δίκτυα αισθητήρων. Στα πλεονεκτήματά της, είναι η εύκολη και γρήγορη προσθήκη επιπλέον συσκευών εντός αυτών των δικτύων. Επίσης βασικό πλεονέκτημα είναι ο τρόπος συμπεριφοράς του πρωτοκόλλου επικοινωνίας, το οποίο και γίνεται εύκολα αντιληπτό από τον χρήστη κατά τη διάρκεια της προσομοίωσης στο κομμάτι του κώδικα και στο κομμάτι των σταδίων των δοκιμών.[44,45]

### 5.2 Χαρακτηριστικά

Το Contiki χρησιμοποιεί προγραμματισμό πολυδιεργασιών (multitasking programming) και παρέχει ενσωματωμένο Σουίτα πρωτοκόλλου TCP/IP. Ακόμη χρησιμοποιεί τα 10KB της μνήμης RAM και τα 30KB της μνήμης ROM. Μόλις 30KB της μνήμης RAM δεσμεύονται για να εκτελεστεί στο εκάστοτε PC ολόκληρο το σύστημα, μαζί με το γραφικό περιβάλλον του χρήστη (GUI). Γι' αυτό, το Contiki

είναι σχεδιασμένο να τρέχει σε συστήματα τα οποία έχουν περιορισμένη μνήμη, ενέργεια, επεξεργαστική ισχύ και επικοινωνιακό εύρος ζώνης.

Το Contiki παρέχει τριών ειδών μηχανισμούς δικτύωσης:

- το uIP TCP/IP stack, το οποίο παρέχει δικτύωση IPv4,
- το uIPv6 stack το οποίο παρέχει IPv6 δικτύωση
- την στοίβα επικοινωνίας Rime ( Rime communication stack) η οποία είναι ένα σύνολο από χαμηλών απαιτήσεων πρωτόκολλα δικτύωσης, τα οποία έχουν σχεδιαστεί με στόχο την χαμηλή κατανάλωση ενέργειας ασύρματων δικτύων.[46,47]

### 5.3 Κόμβοι

Στα πλαίσια της υλοποίησης της εκάστοτε εργασίας, από το πρόγραμμα παρέχονται κάποιοι προσομοιωμένοι κόμβοι. Κατά τη διάρκεια της προσομοίωσης πραγματοποιείται η αντίστοιχη μεταγλώττιση των κόμβων. Κάποιοι από αυτούς είναι οι: Sky Mote, Z1 Mote, ESB Mote, Exp1120, Cooja Mote, MicaZ, Eth1120, Exp2420, Trxeb2520, CC430, WisMote, Exp1101, Trxeb2520[46]

#### 5.3.1 Sky Mote

Το Sky Mote είναι ο κόμβος εκείνος στον οποίο θα βασιστεί η παρουσίαση των προσομοιώσεων. Είναι δόκιμο επομένως, να πραγματοποιηθεί μια εις βάθος μελέτη του. Αξίζει να σημειωθεί, πως αποτελεί ίσως τον κόμβο πάνω στον οποίο αναπτύσσονται οι περισσότερες εκ των υλοποιήσεων στην κοινότητα των προγραμματιστών.[46]

##### 5.3.1.1 Περιγραφή υλικού

Το Tmote sky είναι ένα ασύρματο σύστημα εξαιρετικά χαμηλής ισχύος ιδανικό για χρήση σε δίκτυα αισθητήρων, εφαρμογές παρακολούθησης, και ταχεία πρότυπα εφαρμογής. Το Tmote sky αξιοποιεί βιομηχανικά πρότυπα όπως το USB και το IEEE 802.15.4 για απρόσκοπτη διαλειτουργικότητα με άλλες συσκευές. Μέσω των αυτού μπορούν να ληφθούν μετρήσεις που αφορούν την υγρασία, τη θερμοκρασία και το φως παρέχοντας διασύνδεση με άλλα περιφερειακά Tmotest. Έτσι μπορεί να δημιουργηθεί πλέον ένα ευρύ φάσμα δικτύου με τις αντίστοιχες εφαρμογές[46,47].

##### 5.3.1.2 Βασικά Χαρακτηριστικά

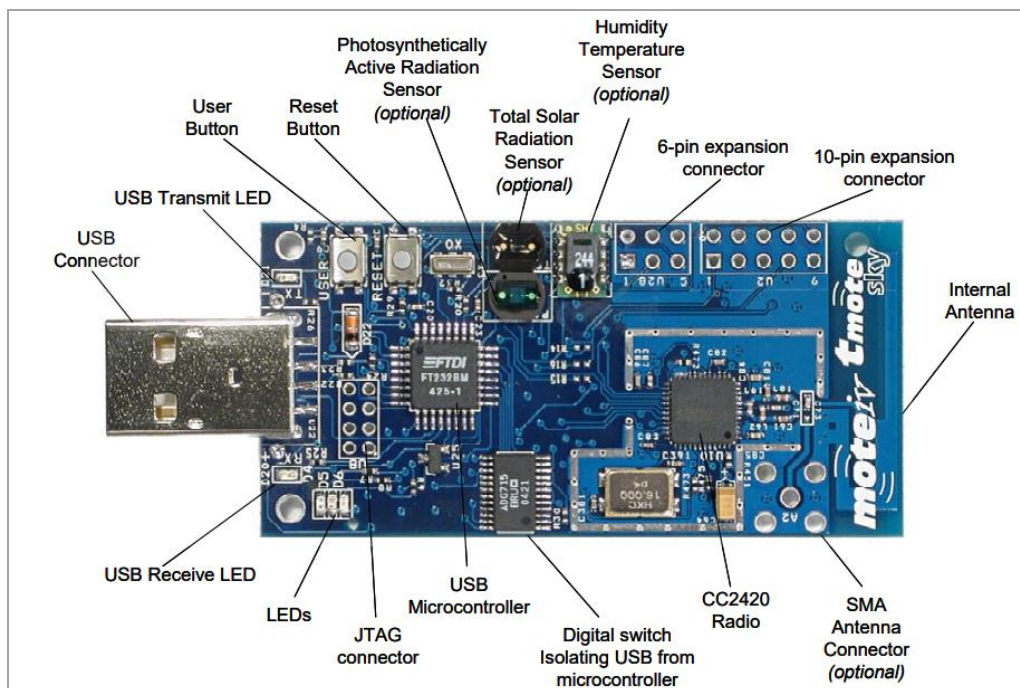
Παρακάτω παρατίθενται τα τεχνικά χαρακτηριστικά του Tmote Sky μέσω του Datasheet του κατασκευαστή:

- 250kbps 2.4GHz IEEE 802.15.4 Chipcon Wireless Transceiver
- Interoperability with other IEEE 802.15.4 devices
- 8MHz Texas Instruments MSP430 microcontroller (10k RAM, 48k Flash)

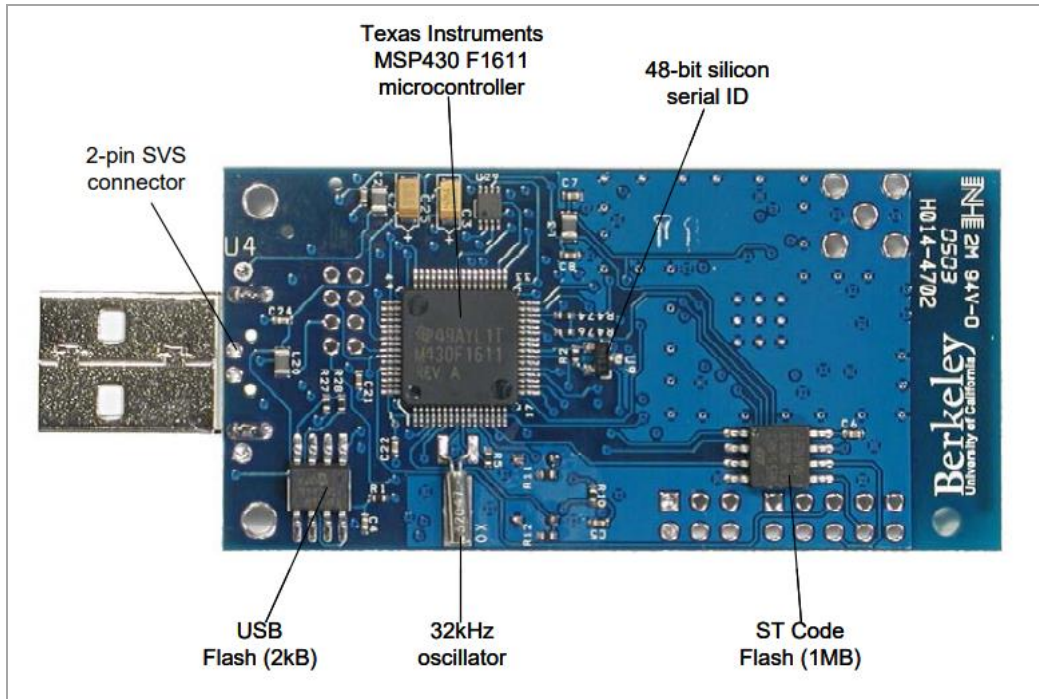
- Integrated ADC, DAC, Supply Voltage Supervisor, and DMA Controller
- Integrated onboard antenna with 50m range indoors / 125m range outdoors
- Integrated Humidity, Temperature, and Light sensors
- Ultra-low current consumption
- Fast wakeup from sleep (<math><6\mu\text{s}</math>)
- Hardware link-layer encryption and authentication
- Programming and data collection via USB
- 16-pin expansion support and optional SMA antenna connector
- TinyOS support : mesh networking and communication implementation [46]

### 5.3.1.3 Περιγραφή του Module

Η μονάδα Tmote Sky είναι ένα mote χαμηλής ισχύος με ενσωματωμένους αισθητήρες, σύστημα ραδιοεκπομπής, κεραία, μικροελεγκτή και δυνατότητες προγραμματισμού. Οι εικόνες 5.1 και 5.2 δείχνουν την αποτύπωση της πλακέτας του mote :



Εικόνα 5.1 : Περιγραφή Tmote (Πάνω πλευρά) [46]



Εικόνα 5.2 : Περιγραφή Tmote (Κάτω πλευρά) [46]

#### 5.3.1.4 Τροφοδοσία

Το Tmote sky μπορεί να τροφοδοτείται από δύο μπαταρίες τύπου AA. Το εύρος λειτουργίας είναι από 2,1 έως 3,6 V DC, ωστόσο η τάση πρέπει να είναι τουλάχιστον 2,7 V κατά τον προγραμματισμό του φλας του μικροελεγκτή ή του εξωτερικού φλας. Εάν η μονάδα Tmote sky συνδεθεί στη θύρα USB για προγραμματισμό ή επικοινωνία, θα λάβει την απαραίτητη τροφοδοσία από τον κεντρικό υπολογιστή η οποία ανέρχεται στα 3V. Εάν το Tmote sky είναι συνέχεια συνδεδεμένο μέσω θύρας USB, τότε δεν απαιτείται η χρήση μπαταρίας. Ο σύνδεσμος επέκτασης 16 ακίδων μπορεί να παρέχει τροφοδοσία στο module. Οποιαδήποτε από τις συνδέσεις ακροδεκτών της μπαταρίας μπορεί επίσης να παρέχει ρεύμα στη μονάδα[46,49].

#### 5.3.1.5 Τυπικές συνθήκες λειτουργίας

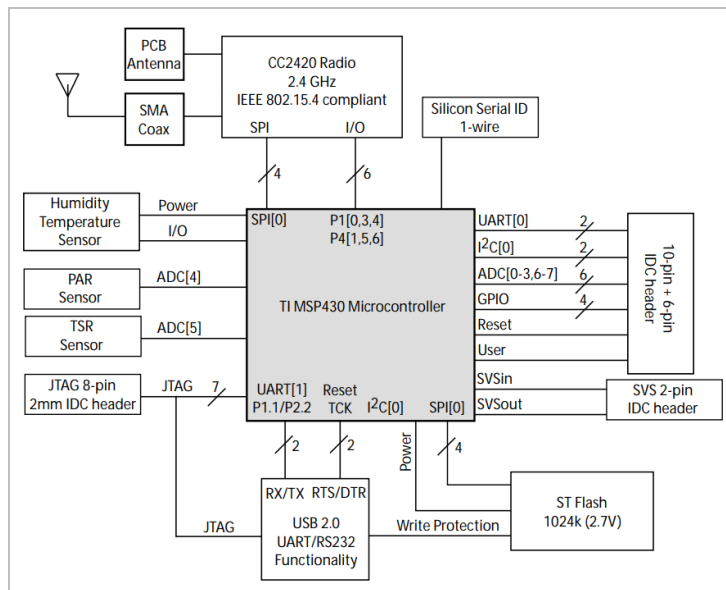
Στον πίνακα 5.1, παρουσιάζονται οι τυπικές συνθήκες λειτουργίας του Tmote Sky όπως ακριβώς αποτυπώνονται στο αντίστοιχο datasheet.

	MIN	NOM	MAX	UNIT
Τάση τροφοδοσίας	2.1		3.6	V
Τάση τροφοδοσίας κατά τον προγραμματισμό της μνήμης flash	2.7		3.6	V
Θερμοκρασία ελεύθερου αέρα λειτουργίας	-40		85	C
Τωρινή κατανάλωση: MCU on, Radio RX		21.8	23	mA
Τωρινή κατανάλωση: MCU on, Radio TX		19.5	21	mA
Τωρινή κατανάλωση: MCU on, Radio off		1800	2400	μA
Τωρινή κατανάλωση: MCU idle, Radio off		54.5	1200	μA
Τωρινή κατανάλωση: MCU stanby		5.1	21	μA

**Πίνακας 5.1: Τυπικές συνθήκες λειτουργίας Tmote Sky [46]**

### 5.3.1.6 Μπλοκ Διάγραμμα Tmote Sky

Το λειτουργικό μπλοκ διάγραμμα (Block Diagram) της μονάδας Tmote Sky, των στοιχείων της και των buses του mote αποτελείται από υλικά όπως η PCB κεραία, το ST φλας (2.7V), τους PAR και TSR αισθητήρες όπως και τους αισθητήρες υγρασίας και θερμοκρασίας. Η εικόνα 5.3 απεικονίζει το διάγραμμα αυτό.



**Εικόνα 5.3: Block diagram Tmote Sky [46]**

### 5.3.2 Z1 Mote - (Zolertia Mote)

Το Z1 Mote είναι ένας εξίσου σημαντικός κόμβος ο οποίος δύναται να χρησιμοποιηθεί μέσω της εφαρμογής Contiki και να μας δώσει μια ευρεία γκάμα επιλογών. Ακολουθεί μια μικρή περιγραφή του mote αυτού.



### 5.3.2.1 Περιγραφή υλικού

Το Z1 Mote της Zolertia είναι μια μονάδα ασύρματου δικτύου αισθητήρων χαμηλής κατανάλωσης (WSN) που χρησιμεύει κυρίως ως πλατφόρμα ανάπτυξης γενικού σκοπού για προγραμματιστές WSN, ερευνητές, λάτρεις και χομπίστες. Είναι συμβατό με τα πρωτόκολλα IEEE 802.15.4 και Zigbee και προορίζεται να βοηθήσει τους προγραμματιστές WSN να δοκιμάσουν και να αναπτύξουν τις δικές τους εφαρμογές και πρωτότυπα με σκοπό την ανάπτυξη και την ευελιξία του υλικού[51].

### 5.3.2.2 Βασικές λειτουργίες

- Περιβαλλοντική παρακολούθηση
- Ανιχνευτές έκτακτης ανάγκης
- Συσκευές ασφάλειας και διάσωσης
- Μακροχρόνια παρακολούθηση χωρίς επίβλεψη
- Μέτρηση Ενέργειας
- Παρακολούθηση Αγροτικών Δραστηριοτήτων [51,52]

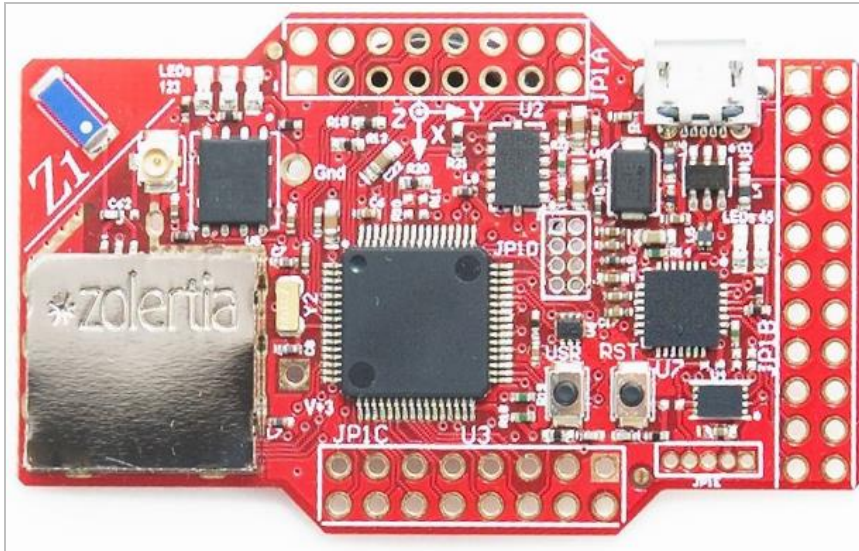
### 5.3.3.3 Βασικά Χαρακτηριστικά

Παρακάτω παρατίθενται τα τεχνικά χαρακτηριστικά του Tmote Sky μέσω του αντίστοιχου Datasheet:

- Industrial-grade temperature range (-40°C-85°C)
- 52-pin expansion connector
- 2nd generation MSP430™ ultra-low power 16-bit MCU 16MHz
- 2.4GHz IEEE 802.15.4, 6LoWPAN compliant and ZigBee™ ready
- 3-Axis, ±2/4/8/16 g digital accelerometer
- Low-power digital temperature sensor with ±0.5°C accuracy (in -25°C ~ 85°C range)
- Optional external antenna via U.FL connector
- Micro-USB connector for power and debugging [51]

### 5.3.3.4 Το module

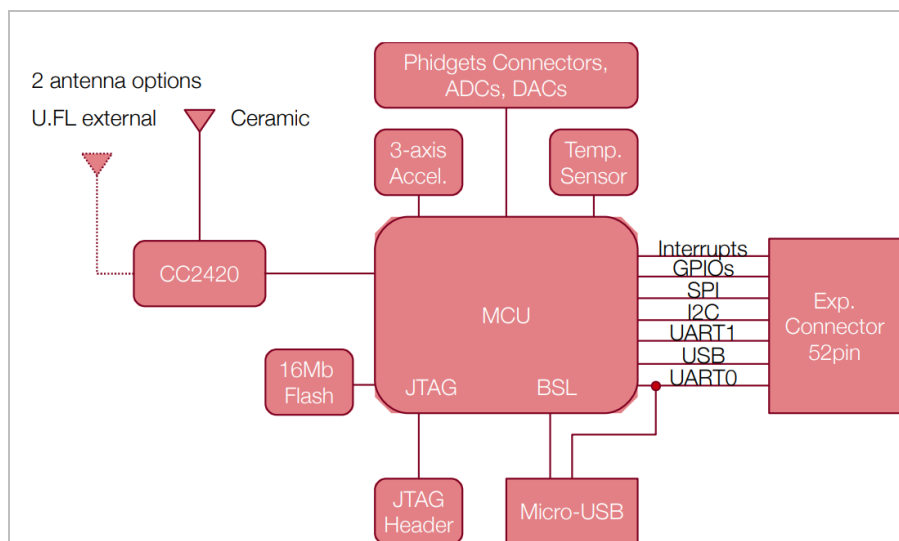
Παρατίθεται εικόνα του module του Z1 Mote με απεικόνιση εξοπλισμού όπως οι διακόπτες, οι timers και των comparators της πλακέτας. :



Εικόνα 5.4: Z1 Mote [51]

### 5.3.3.5 Μπλοκ Διάγραμμα Z1 Mote

Παρατίθεται εικόνα με την αποτύπωση του μπλοκ διαγράμματος του Z1 mote με απεικόνιση του εξοπλισμού του όπως η θύρα USB, η είσοδος τροφοδοσίας, οι αισθητήρας θερμοκρασίας, το 16 MB Flash και λοιπός εξοπλισμός.



Εικόνα 5.5: Z1 Mote – Block Diagram [51]

## 5.4 Threads και Protothreads

Αντίθετα με την πλειοψηφία υπόλοιπων περιπτώσεων συστημάτων τα οποία αναπτύσσονται με βάση τα γεγονότα, το παρόν πρόγραμμα αναπτύσσει την προσέγγιση multithreading. Η διαδικασία αυτή επιτυγχάνεται μέσω μιας βιβλιοθήκης η οποία ανάλογα με τις συνθήκες έχει την δυνατότητα να συνδέεται με προγράμματα που το απαιτούν. Με αυτόν τον τρόπο επιτρέπεται στους κόμβους να λειτουργούν

εφαρμογές, που κανονικά δεν έχουν την δυνατότητα να υποστηρίξουν αναφορικά με συστήματα που πυρήνα τους αποτελούν τα γεγονότα, όπως κρυπτογραφικούς υπολογισμούς. Αυτή η λειτουργία σε άλλες περιπτώσεις έχει μεγάλη χρονική διάρκεια και καταλαμβάνει όλο το σύστημα.

Πέραν των threads, το Contiki παρέχει και την υποστήριξη των Protothreads. Κατά βάση όταν γράφονται κώδικες για συστήματα που είναι καθοδηγούμενα από γεγονότα, είναι βασικό να παρουσιάζονται ως μια συγκεκριμένη κατάσταση μηχανημάτων με ρητά ανεπτυγμένο κώδικα, ο οποίος ενδεχομένως να οδηγήσει σε μια δυσκολία διατήρησης και εμπέδωσής του. Τα Protothreads είναι μια διαδικασία συγκέντρωσης του συνόλου του προγραμματισμού της εφαρμογής, που χρησιμοποιείται στα ανώτερα επίπεδα των εν λόγω συστημάτων και στοχεύουν σε όσο πιο απλές διεργασίες σε συνάρτηση με λειτουργικότητα υψηλών απαιτήσεων. Μέσω των Protothreads, δίνεται η δυνατότητα εκτέλεσης των προγραμμάτων, χωρίς να υπάρχει επιπλέον καταπόνηση λειτουργίας των threads που συναντώνται συνήθως. [53,54]

## 5.5 Διασυνδέσεις COOJA

Οι διασυνδέσεις COOJA είναι ο βασικότερος τρόπος και εκείνος που συναντάται στις περισσότερες των περιπτώσεων αναφορικά με την επεξεργασία και την συνεργασία ανάμεσα στους κόμβους της εκάστοτε προσομοίωσης. Ανεξάρτητα του γεγονότος ότι προσομοιώνουν όλα τα υλικά εξαρτήματα με τον τρόπο που αντιμετωπίζονται οι διασυνδέσεις (interfaces), έχουν τη δυνατότητα να δημιουργήσουν και μια άριστα εναρμονισμένη διαδικασία. Αντί να υπάρχουν διασυνδέσεις αποκλειστικά για τον εξοπλισμό ενός κόμβου, καλύτερη επιλογή είναι το να αποτελούν διασυνδέσεις των χαρακτηριστικών του. Μια εξ αυτών των χρήσεων αποτελεί η τοποθεσία του κόμβου. Αρκετές φορές ο κόμβος δεν γνωρίζει την αληθινή του τοποθεσία και για το λόγο αυτό, τέτοιες διασυνδέσεις ονοματίζονται ως νοητές (virtual). Η προσαρμογή της θέσης της διασύνδεσης, παρέχει τη δυνατότητα ανάπτυξης προσομοίωσης αποτελούμενης από μεταβαλλόμενους κόμβους. Τέτοιου είδους λειτουργίες έχουν σημασία στις περιπτώσεις που υπάρχει η θέληση για προσομοίωση κόμβων, με σκοπό για παράδειγμα την αποτύπωση μιας κύλισης πάνω σε βασικά σημεία στήριξης ενός ρομπότ το οποίο βρίσκεται εν κινήσει.

Οι διασυνδέσεις COOJA συναντώνται τόσο εντός του πυρήνα, όσο και εντός του προσομοιωτή. Εκείνες που δραστηριοποιούνται εντός του προσομοιωτή (simulation interfaces), διαθέτουν εξ ολοκλήρου δικαίωμα πρόσβασης στη μνήμη του κόμβου. Εκείνες οι οποίες δραστηριοποιούνται εντός του πυρήνα (core interfaces) διαθέτουν τη δυνατότητα πρόσβασης στις συναρτήσεις του προγράμματος. Συχνό φαινόμενο είναι μεταξύ των διασυνδέσεων του πυρήνα και του προσομοιωτή να δημιουργούνται τόσο καταστάσεις σχέσεων όσο και εξαρτήσεων. Ένα τέτοιο παράδειγμα είναι ο ραδιοπομπός, ο οποίος πρέπει να αποτελείται από μια διασύνδεση ραδιοσυχνοτήτων. Στην περίπτωση που τα δεδομένα αυτά βρίσκονται σε κατάσταση μεταφοράς ή λήψης, οι διασυνδέσεις επικοινωνούν η μία με την άλλη και όχι απευθείας με το Contiki. Κατά το χρονικό διάστημα που ο κόμβος τίθεται σε λειτουργία, η διασύνδεση του πυρήνα δύναται να προσφέρει δεδομένα στο σύστημα με τρόπο,

όπως και οποιαδήποτε άλλη συσκευή στο δίκτυο. Με τη διαδικασία αποθήκευσης δηλαδή των δεδομένων εντός προσωρινού χώρου και ανακαλώντας τη συνάρτηση `uip_input()`.

Επιπροσθέτως δίνεται η δυνατότητα διαχωρισμού μεταξύ των διασυνδέσεων του προσομοιωτή σε ενεργητικές και παθητικές. Η κατάσταση του κόμβου καθορίζει τη μεθοδολογία με την οποία δύναται να υλοποιηθεί αυτός ο διαχωρισμός. Η μοναδική λειτουργία του κόμβου που δεν έχει τη δυνατότητα να διαχειριστεί, είναι εκείνη κατά την οποία η κατάσταση του μπορεί να είναι είτε ενεργή (active), είτε σε αναμονή (stand-by), είτε κλειστή (shut). Ο απενεργοποιημένος κόμβος συνεχίζει να είναι υπαρκτός, αλλά δεν μπορεί ποτέ να ενεργοποιηθεί και να μεταβάλλει την κατάσταση του. Όταν ο κόμβος ο οποίος είναι σε κατάσταση αναμονής τεθεί σε λειτουργία, οι παθητικές διασυνδέσεις είναι οι μόνες που μπορούν να δραστηριοποιηθούν και η διαδικασία της διέγερσης δεν δίνεται στον πυρήνα. Μια περίπτωση μιας τέτοιας διασύνδεσης η οποία επιβάλλεται να είναι παθητική είναι εκείνη της μπαταρίας. Είναι αναγκαίο να βρίσκεται σε μια ενεργή κατάσταση σε κάθε περίπτωση ακόμα και όταν ο κόμβος είναι σε λειτουργία αναμονής, καθώς ακόμη και τότε έχει ανάγκη την παροχή ενέργειας μέσω της μπαταρίας. Αντίθετα, μια ενεργή διασύνδεση που περιγράφει ένα PIR αισθητήρα ενδέχεται να μην έχει τη δυνατότητα παρατήρησης των αλλαγών του φωτός στην περίπτωση που ο κόμβος είναι στην κατάσταση αναμονής. Εκείνες οι οποίες είναι ενεργές έχουν την δυνατότητα να ενεργοποιούν ένα κόμβο που βρίσκεται σε αναμονή με το να δημιουργούν μια εξωτερική διακοπή.

Κατά την περίπτωση όπου μια διασύνδεση κόμβου επιθυμεί πρόσβαση εντός μιας μεταβλητής, επί της ουσίας υλοποιείται μια αντιστοίχιση μεταξύ του ονόματος της μεταβλητής και της διεύθυνσης μνήμης. Προκειμένου να υπάρχει ένας σχεδιασμός με προοπτική επέκτασης είναι αναγκαίο κάθε διασύνδεση να έχει την ευθύνη των εξαρτήσεών της. Επομένως, όταν μια διασύνδεση υλοποιείται είναι απαραίτητο να εξετάζεται, ώστε το σύνολο των βασικών μεταβλητών και οι υπόλοιπες διασυνδέσεις να είναι διαθέσιμες.

Όλες οι διασυνδέσεις της προσομοίωσης είναι βασικό να διαθέτουν τη δυνατότητα της παρατήρησης. Οποιοδήποτε κομμάτι του προσομοιωτή δύναται να δηλωθεί ως παρατηρητής και παρουσιάζεται όταν το αποφασιστεί. Η διασύνδεση ενός πομπού έχει για παράδειγμα την δυνατότητα να επιλέγει να γνωστοποιεί τους παρατηρητές του, όταν είναι έτοιμος να αποστείλει δεδομένα. Ένας μη μεταβλητός από πομπούς παρατηρητής, είναι ένας ραδιομεταφορέας, ο οποίος επίσης έχει δυνατότητα της μεταφοράς των νέων δεδομένων και της κρίσης αναφορικά με την αποστολή τους, στους υπόλοιπους πομπούς

Ο τρόπος με τον οποίο ενεργεί ο παρατηρητής επιφέρει εντονότερες αλληλεπιδράσεις εντός των επιπέδων του προσομοιωτή, και δεν περιορίζεται εξ ολοκλήρου από τη διασύνδεση της προσομοίωσης. Για την προσομοίωση την οποία παρακολουθεί, για παράδειγμα, ειδοποιεί όλους τους παρατηρητές όταν εκείνη εκκινήσει ή εισέλθει σε κατάσταση αναμονής, όταν δημιουργηθούν καινούργιοι κόμβοι ή στην περίπτωση που προστεθούν ή αφαιρεθούν κόμβοι. Η διαδικασία κατά την οποία επαναλαμβάνεται μια προσομοίωση μπορεί επίσης να τεθεί υπό παρακολούθηση,

ώστε να γνωστοποιήσει στους παρατηρητές της, τότε έχει συμπληρωθεί ένας πλήρης κύκλος.

Μια διασύνδεση προσομοίωσης μπορεί σε κάποιες περιπτώσεις να διαθέτει τη δυνατότητα γραφικής παρουσίασης, σε περιπτώσεις που δίνει πληροφορίες και αλληλοεπιδρά με τον διαχειριστή. Οι γραφικές αναπαραστάσεις δημιουργούνται ως τυπικά Java panels και έχουν την δυνατότητα να αναπαρασταθούν σε επιπρόσθετο σύστημα του COOJA.

Ένας αποκλειστικός τύπος κόμβου λαμβάνει τις διασυνδέσεις όσο διαρκεί η διαδικασία μεταγλώττισης. Ο βέλτιστη μέθοδος είναι η επιλογή των διασυνδέσεων προσομοίωσης που είναι αναγκαίο να υποστηρίζονται και τότε ό,τι προβλέπεται από τον πυρήνα θα προστεθεί από μόνο του χωρίς κάποια επιπλέον δράση [55,56].

## 5.6 Δομές μνήμης

Η μνήμη διαδικασίας απαρτίζεται από ξεχωριστά τμήματα. Κάθε τμήμα εξ αυτών αποτελεί μια γραμμή διευθύνσεων χωρίς την παρουσία κενών. Το σύνολο των δεδομένων ενός τμήματος έχουν κοινή αντιμετώπιση. Τα τμήματα τα οποία υπάρχουν διαφέρουν μεταξύ των υλοποιήσεων, αλλά το κοινό στοιχείο μεταξύ όλων είναι τα απλοποιημένα, όπου είναι κατ’ ελάχιστον τρία.

Τα τμήματα αυτά είναι: αρχικά το κείμενο (text), τα δεδομένα (data) και τα BSS (Basic Service Sets) τμήματα. Το τμήμα κειμένου έχει ως αρχή τη διατήρηση του κώδικα του προγράμματος και των σταθερών του. Στις περισσότερες των περιπτώσεων δεν μεταβάλλεται κατά τη διάρκεια που το πρόγραμμα βρίσκεται σε εξέλιξη. Το τμήμα δεδομένων διατηρεί τις αρχικές τιμές των μεταβλητών και σε αντίθεση με το παραπάνω μπορεί να τροποποιηθεί. Το τμήμα bss έχει παρόμοια χαρακτηριστικά με το τμήμα των δεδομένων, αλλά έχει ως διαφορά το να συγκρατεί μη αρχικοποιημένες μεταβλητές. Το τμήμα bss λειτουργεί κατά τέτοιο τρόπο ώστε να υπάρχει δυνατότητα να εξοικονομείται χώρος σε εκτελέσιμα δυαδικά αρχεία. Με δεδομένο πως όλα τα δεδομένα στο τμήμα έχουν αρχική τιμή μηδέν όταν εκκινείται το πρόγραμμα, αναγκαία είναι η αποθήκευση λοιπόν στο διάδικο αρχείο, του μήκους του τμήματος, και όχι όλων των μηδενικών δεδομένων. Επιπλέον τμήματα μνήμης είναι ο σωρός (heap) και η στοίβα (stack). Το τμήμα σωρού καθιστά ενεργή τη δυναμική μνήμη. Κατά τον χρόνο ανάπτυξης ενός προγράμματος ίσως χρειαστεί η διάθεση καινούργιας μνήμης και η τοποθέτηση με την σειρά της εντός του σωρού. Το τμήμα στοίβας χρησιμοποιείται προκειμένου να επιτευχθεί η διαδικασία αποθήκευσης τοπικών μη στατικών μεταβλητών και παραμέτρων που καλούνται από συνάρτηση. [54,57]

## 5.7 JNI

Η εικονική μηχανή της Java (Java Virtual Machine, JVM) έχει ενσωματωμένη μια διεπαφή (Java Native Interface, JNI), η οποία εξασφαλίζει μια οδό για την ανίχνευση εναλλακτικών μεθόδων σε μια υλοποίηση. Αυτού του είδους ο κώδικας μπορεί να ενεργήσει μέσα στην JVM και διαθέτει την ικανότητα της αλληλεπίδρασης με

εφαρμογές δομημένες πάνω σε διαφορετικές γλώσσες προγραμματισμού όπως η C ή η Assembly. Μέσω της JNI καθίσταται εφικτή η ανάκληση βιβλιοθηκών και APIs (Application Programming Interfaces) που δεν υλοποιούνται στην Java ή ακόμη και για να γίνουν ταχύτερα οι υπολογισμοί με τη χρήση κώδικα Assembly. Με τη μέθοδο `System.loadLibrary` μια βιβλιοθήκη φορτώνεται σε μία κλάση Java και συγκεκριμένες μέθοδοι αντιστοιχίζονται μετά τη φόρτωση μέσω λειτουργιών στη βιβλιοθήκη.

Ο φορτωτής κλάσης ξεκινά να συλλέγει μη χρήσιμα δεδομένα. Εφόσον έχει φορτωθεί μια βιβλιοθήκη, τότε και μόνο τότε θα εκφορτωθεί. Πολλές βιβλιοθήκες δύναται επιδεχθούν τη διαδικασία φόρτωσης από όμοιο φορτωτή κλάσης στην περίπτωση που υπάρχει ξεχωριστή ονομασία. Όταν όμως, φορτωθούν τα ονόματα της συνάρτησης θα είναι δεσμευμένα και οποιαδήποτε βιβλιοθήκη. Επομένως μετ’ έπειτα, είναι αδύνατο να εκτελεστούν οι συναρτήσεις της ίδιας εγγραφής. [58,59]

## 5.8 Πρόσθετα

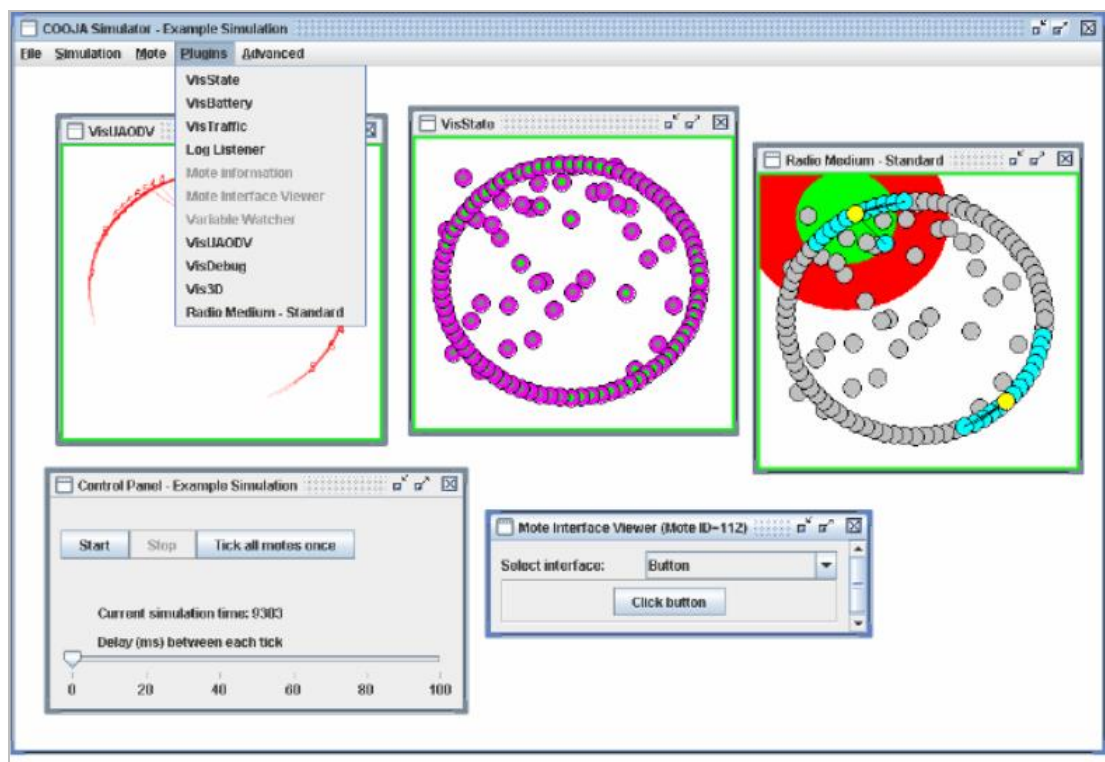
Οι διασυνδέσεις COOJA αποτελούν τον βέλτιστο τρόπο αλληλεπίδρασης με τους προσομοιωμένους κόμβους, ενώ τα πρόσθετα (plugins) είναι ο βέλτιστος τρόπος για την επικοινωνία με μια προσομοίωση. Τα plugins είναι ενσωματωμένα εντός του χρόνου εκτέλεσης (runtime) πριν ακόμα τεθούν σε χρήση και στις περισσότερες των περιπτώσεων κατά τη διαδικασία εκκίνησης του προσομοιωτή. Κατά τη διάρκεια των προσομοιώσεων, ο εκάστοτε χρήστης δύναται να δημιουργεί συγκυρίες διαθέσιμων καταχωρημένων προσθέτων. Χρησιμοποιούνται όπως ένα τυπικό σύνολο επιλογών της Java, και έτσι ο χρήστης μπορεί να υλοποιήσει εξελιγμένα γραφικά διασυνδέσεων κατευθείαν. Τα πρόσθετα διακρίνονται σε τέσσερα διαφορετικά είδη, και προσδιορίζονται με διαφορετικό τρόπο το καθένα.

Το 1<sup>ο</sup> είδος είναι το γραφικό περιβάλλον. Το γραφικό περιβάλλον ή αλλιώς GUI (Graphical User Interface), θα μπορούσε να χαρακτηριστεί ως το ευκολότερο σε σχέση με τα υπόλοιπα, καθώς προαπαιτεί η δημιουργία μόνο ενός λειτουργικού GUI προκειμένου να εισέλθει ως συνθήκη τη στιγμή που ο χρήστης πραγματοποιεί αρχικοποίηση του Plugin. Στην περίπτωση που οι εικονικοί κόμβοι μπορούν να διευθετηθούν με το GUI και στην περίπτωση που υπάρχει σε εξέλιξη κάποιου είδους προσομοίωση, τότε και μόνο τότε μπορούμε να ειπωθεί πως τα σχετικά δεδομένα εξελίσσονται. Για να γίνει περισσότερο αντιληπτή η κατάσταση, υπάρχει η παραδειγματική κατάσταση, όπου το GUI-plugin, μπορεί να επιτελέσει τη λειτουργία test-tun, κατά την οποία γίνεται μια μεταφόρτωση της εκάστοτε προσομοίωσης, πραγματοποιούνται δοκιμές, αποθηκεύεται η πληροφορία και μεταφορτώνεται εκ νέου καινούργια προσομοίωση.

Το 2<sup>ο</sup> και 3<sup>ο</sup> είδος βασίζονται σε μια προσομοίωση, τη Simulation plugin types. Ένα παράδειγμα, που μπορεί να παρατεθεί είναι η ανάδειξη δεδομένων αναφορικά με την τρέχουσα προσομοίωση, καθώς και το σύνολο των προσομοιωμένων κόμβων και τύπων ή η τρέχουσα κατάσταση της προσομοίωσης. Αξίζει να σημειωθεί ότι, είναι χρήσιμη η γνώση της γραφικής παράστασης των σημείων εμφάνισης του συνόλου

αυτού των προσομοιωμένων κόμβων. Ανάμεσα στα δύο αυτά είδη των προσθέτων, το ένα είδος (που ονομάζεται simulation standard type) αυτοδημιουργείται κατά την περίπτωση που μια καινούργια προσομοίωση παρουσιαστεί και το έτερο είδος ξεκινά να αναπτύσσεται προαιρετικά από τον εκάστοτε χρήστη.

Το εναπομείναν είδος προσθέτου, βασίζεται σε ένα προσομοιωμένο κόμβο και η κίνηση του κόμβου είναι αλληλένδετη με το plugin, αφού όταν μετακινείται ο κόμβος, μετακινείται και εκείνο. Ονομάζεται Mote plugin type. Η εξέταση μιας ορισμένης μεταβλητής του κόμβου και η παύση της προσομοίωσης κάθε φορά που αλλάζει η μεταβλητή, είναι ένα τέτοιο παράδειγμα. Επιπλέον παράδειγμα, είναι ένα plugin που εξετάζει το ιστορικό χρήσης ενέργειας, ιδανικό για να ελέγχει την ποσότητα της ενέργειας που απομένει στην μπαταρία ενός κόμβου και να παρουσιάζει αυτά τα δεδομένα γραφικά ως προς το χρονοστιγμιότυπο με διαφορετικούς τύπους κόμβους με αρχικοποιημένα πρόσθετα. Το παράδειγμα αυτό αποτυπώνεται στην εικόνα 5.6.



**Εικόνα 5.6: Αρχικοποιημένα Plugins[59]**

Η πλειοψηφία των ειδών είναι απαραίτητο να φορτώνονται όταν το πρόγραμμα ξεκινά την ανάπτυξή του και να βρίσκονται σε κατάσταση διαθεσιμότητας έως ότου το COOJA τερματιστεί. Το COOJA έχει την ικανότητα να τα αποθηκεύει, καθώς και να τα καταργεί. Αυτό βέβαια είναι σε συνάρτηση με τις συνθήκες της τρέχουσας προσομοίωσης. Αυτά τα πρόσθετα λαμβάνουν την ονομασία δυναμικά (dynamic plugins). Η μόνη διαφορά μεταξύ τους εμφανίζεται στον τρόπο εγγραφής τους. Κάποια από τα τμήματα της προσομοίωσης υποστηρίζουν τα πρόσθετα αυτά, λόγω του ότι ενδέχεται να υπάρξει η ανάγκη καταγραφής ενός εξ αυτών από αυτών. Τα

μέσα, μέσω των οποίων επιτυγχάνεται η μετάδοση, θα μπορούσαν κάλλιστα να αποτελέσουν ένα τέτοιο παράδειγμα. Το μέσο μετάδοσης το οποίο βρίσκεται σε χρήση, αποτελεί το μοναδικό υπεύθυνο τμήμα στο COOJA αναφορικά με τις πληροφορίες που αφορούν τον τρόπο χειρισμού των εκπεμπόμενων δεδομένων. Τον καθορισμό δηλαδή των κόμβων, οι οποίοι θα έχουν την ικανότητα να φτάσουν και να συγκρουστούν με άλλους κόμβους, σφάλματα μετάδοσης των bit κλπ. Η εφαρμογή μιας γενικής ιδεατής διασύνδεσης δεν χαρακτηρίζεται από ιδιαίτερη ευκολία για το σύνολο των πρόσθετων, διότι τα μέσα μετάδοσης ενδέχεται να είναι πολύπλοκα ή εξελιγμένα. Μέσω της παρουσίας δυναμικών προσθέτων, το μέσο μετάδοσης μπορεί να εξετάζει ένα ή περισσότερα από αυτά. Έπειτα ο χρήστης να ενεργοποιήσει ή να τροποποιήσει συγκεκριμένες παραμέτρους του μέσου μετάδοσης. Στην εικόνα 5.6 παρατηρείται ότι το μέσο μετάδοσης του οποίου γίνεται χρήση, καλείται Radio Medium - Standard και καταχωρεί ένα δυναμικό πρόσθετο που φέρει την ίδια ονομασία. Έτσι δίνεται η δυνατότητα στο χρήστη να προβάλλει, να μεταβάλλει τη μετάδοση και να επεξεργαστεί τα όρια. [54,58]

## 5.9 Μέσα Μετάδοσης

Ο προσομοιωτής COOJA διαθέτει τη δυνατότητα πρόσθεσης και τη χρήση ξεχωριστών μέσων διάδοσης, καθώς εντός ενός κεντρικού δικτύου αισθητήρων αυτός διαθέτει κεντρικό ρόλο. Οποιοδήποτε μέσο μετάδοσης υπακούει σε ένα άθροισμα που αποτελείται από καταγεγραμμένους πομπούς. Ο εκάστοτε πομπός διαθέτει μια ξεχωριστή θέση. Ένα μέσο μετάδοσης υπεισέρχεται σε μια διαδικασία αναζήτησης και ανάδειξης της θέσεως αυτής, κατά το χρονικό διάστημα που παραλαμβάνονται νέες πληροφορίες από τον πομπό. Αυτή είναι η περίπτωση κατά την οποία ανατίθεται ο ρόλος του υπολογισμού, σχετικά με τον προσδιορισμό των υπολοίπων πομπών, οι οποία θα πρέπει να έχουν τη δυνατότητα αποκωδικοποίησης των πληροφοριών αυτών. Το παρόν μέσο μετάδοσης είναι το μοναδικό το οποίο διακινεί στο δίκτυο δεδομένα μέσω ασύρματης επικοινωνίας. Μέσω της διαδικασίας αυτής, δύναται να επιτευχθεί η εξέλιξη ενός μέσου μετάδοσης. Προκειμένου να επιτευχθεί η αύξηση αλληλεπίδρασης μεταξύ του χρήστη και του εκάστοτε μέσου μετάδοσης απαραίτητη είναι η καταχώριση δυναμικών COOJA διασυνδέσεων.

Στη βασική έκδοση του COOJA εμφανίζονται δύο εκ των μέσων μετάδοσης, τα οποία με απλό τρόπο αναπτύσσονται και καταχωρούνται κατά ενεργοποίηση του προσομοιωτή. Η λειτουργία του ενός εκ των δύο χαρακτηρίζεται από την αποφυγή δημιουργίας θορύβου και την αδυναμία μεταφοράς πληροφοριών. Το έτερο μέσο αναπτύσσει δύο παραμέτρους που αφορούν η μια το εύρος μετάδοσης και η άλλη το εύρος παρεμβολής. Οι ομάδες των δεδομένων φτάνουν σε μια ασύρματη συσκευή λήψης, με τη μεταφορά τους να επιτυγχάνεται αδιάκοπα. [59,60]



## 5.10 Σύστημα Διαμόρφωσης

Μέσω της Java του προσομοιωτή αναπτύσσεται το σύστημα διαμόρφωσης, στο οποίο περιέχεται και ένα αναντικατάστατο κομμάτι του προσομοιωτή, καθώς και υλοποιημένα πρόσθετα. Ένας χρήστης έχει τη δυνατότητα να προσθέσει μια διαμόρφωση ή να τοποθετήσει μια άλλη στη θέση της.

Ο πυρήνας των ενεργειών του προσομοιωτή εστιάζει στην ανάγνωση των ρυθμιστών, των θέσεων, των διεπαφών, των προσθέτων και των μέσων μετάδοσης τα οποία είναι αναγκαίο να ενσωματωθούν κατά την εκκίνηση. Ένας χρήστης διαθέτει τη δυνατότητα της επιλογής μόνο ανάμεσα στα αυτά τα δεδομένα, κατά το χρονικό διάστημα που ο προσομοιωτής εκκινείται. Ωστόσο μέσω της φόρτωσης ενός διαφορετικού αρχείου διαμόρφωσης, δύναται να τεθούν στη διάθεση του χρήστη μέσω αυτών και καινούργια στοιχεία.

Ο τρόπος με τον οποίο θα μπορούσε να εκμεταλλευτεί το σύστημα διαμόρφωσης μια διασύνδεση μπαταρίας για παράδειγμα, μπορεί να είναι η ανάγνωση της αρχικής του ενέργειας. Κατά τη διαδικασία αυτή μια διασύνδεση ενός LED, δύναται να αναγνώσει το ποσό της ενέργειας το οποίο καθίσταται απαραίτητο κατά την ανάπτυξη των εκάστοτε σταδίων του. Έτσι με μοναδική ενέργεια την μεταβολή των δεδομένων διαμόρφωσης, το προσομοιωμένο υλικό, μπορεί να εμφανίσει σημαντικές αλλαγές, σχετικά με την ταυτότητα των αρχείων διαμόρφωσης τα οποία φορτώνονται. Μια συνήθης χρήση μπορεί να είναι η δημιουργία ξεχωριστών αρχείων διαμόρφωσης, σχετικά με ανάγκη προσομοίωσης του υλικού. Ο διαχωρισμός έγκειται στο χρόνο φόρτωσης των αρχείων αυτών, ο οποίος μπορεί να είναι είτε κατά την εκκίνηση της προσομοίωσης είτε κατά τη δημιουργία του τύπου του κόμβου[54,63].

## 5.11 G.U.I

Η βασική δομή του GUI) διατηρείται όσο το δυνατόν πιο απλή. Στηρίζεται σε ένα παράθυρο επιφάνειας εργασίας, παρέχοντας στον χρήστη την δυνατότητα να δημιουργεί, να φορτώνει, και να αποθηκεύει προσομοιώσεις, χρησιμοποιώντας τα διαθέσιμα μενού και διαλόγους. Επιπλέον, ο χρήστης έχει την δυνατότητα να δημιουργεί νέους κόμβους, να προσαρμόζει ρυθμίσεις και να διαχειρίζεται πρόσθετα που προσαρτώνται.

Κατά τη φόρτωση προσομοίωσης, εμφανίζονται αποθηκευμένες τιμές που ωστόσο, δεν είναι επεξεργάσιμες. Για παράδειγμα, ο χρήστης πρέπει να επανακαθορίσει τους κόμβους πριν προσθέσει νέους, επιτρέποντάς του να προσαρμόζει τις διαδικασίες του προσομοιωτή.

Τα παράθυρα στην επιφάνεια εργασίας λειτουργούν ως πρόσθετα από την αρχή. Χωρίς αυτά, η αλληλεπίδραση περιορίζεται στη δημιουργία και προσθήκη κόμβων. Η βασική αναμενόμενη έκδοση του συστήματος συμπεριλαμβάνει εφέ, τα οποία εστιάζονται στις κυρίως λειτουργίες, όπως η παρουσίαση και η μετακίνηση των κόμβων, η παρακολούθηση της κίνησης των σημάτων, και ο απόλυτος έλεγχος των προσομοιώσεων [58,67].

## 5.12 Ρυθμιστές θέσης και διανομείς IP

Οι ρυθμιστές θέσεων και οι διανομείς IP καταγράφονται και οι δύο ως κανάλια διανομής, υποδηλώνοντας την ευκολία στη διαδικασία δημιουργίας και ένταξης νέων. Οι ελεγκτές θέσεων διαμορφώνουν τις θέσεις των κόμβων και συμπεριλαμβάνονται στην κύρια έκδοση του COOJA. Είτε τοποθετούνται τυχαία, είτε ακολουθούν γραμμική σειρά, είτε τοποθετούνται σε σχήματα όπως «έλλειψη». Οι ελεγκτές μπορεί να εξάγουν θέσεις από εξωτερικά αρχεία ή να τις προσδιορίζουν με βάση συγκεκριμένους αλγορίθμους.

Οι διανομείς IP ορίζουν τις διευθύνσεις IP των κόμβων, είτε με τυχαίο τρόπο, είτε με βάση ένα μοναδικό αναγνωριστικό κόμβου και τη θέση του [58,64].

## 5.13 Διαμόρφωση Περιβάλλοντος Εκκίνησης

Η διαμόρφωση του περιβάλλοντος εκκίνησης εκπορεύεται από τον τομέα Java του προσομοιωτή. Αντικαθιστά τον βασικό κώδικα του προσομοιωτή και ενσωματώνει νέα πρόσθετα προγράμματα. Τα καθορισμένα αρχεία διαβάζονται κατά την εκκίνηση ή κατά τη διαμόρφωση νέων κόμβων. Ο χρήστης έχει τη δυνατότητα να ενσωματώσει νέες ρυθμίσεις ή να αντικαταστήσει τις υπάρχουσες. Η βασική λειτουργία του προσομοιωτή διαβάζει για παράδειγμα ποιοι ρυθμιστές θέσης, διανομείς IP, interfaces, plugins, και μέσα μετάδοσης πρέπει να καταχωρηθούν κατά την εκκίνηση. Ο χρήστης μπορεί να διαλέξει μεταξύ αυτών των στοιχείων κατά την εκκίνηση του προσομοιωτή. Με τη φόρτωση ενός διαφορετικού αρχείου διαμόρφωσης, νέα στοιχεία μπορεί να είναι προσβάσιμα, παραμένοντας ευέλικτα για αλλαγές.

Για παράδειγμα, ο τρόπος μέσω του οποίου ένα σύστημα διαμόρφωσης θα χρησιμοποιούσε μια διεπαφή μπαταρίας, θα ήταν μέσω της ανάγνωσης της αρχικής της ενέργειας. Αντίστοιχα, ένα “led interface” θα μπορούσε να αναγνωρίζει την κατανάλωση ενέργειας κατά τα διάφορα στάδια. Με αυτόν τον τρόπο, οι αλλαγές στα αρχεία διαμόρφωσης μπορούν να οδηγήσουν σε σημαντικές διαφοροποιήσεις στο προσομοιωμένο υλικό, ανάλογα με τα φορτωμένα αρχεία. Ένα συνηθισμένο σενάριο είναι η δημιουργία διαφορετικών αρχείων διαμόρφωσης για διάφορες υλικές πλατφόρμες, είτε κατά την εκκίνηση της προσομοίωσης είτε κατά τη δημιουργία του τύπου του κόμβου. [58,64,67]

## ΚΕΦΑΛΑΙΟ 6ο : Εγκατάσταση – Λειτουργία - Simulation

Στο κεφάλαιο αυτό πραγματοποιείται μια αναλυτική περιγραφή του Contiki, η οποία περιλαμβάνει όλα τα απαραίτητα βήματα, από την λήψη, την εγκατάσταση έως το τελικό αποτέλεσμα που είναι η συλλογή της πληροφορίας. Σκοπός του συγκεκριμένου κεφαλαίου είναι να αποτελέσει έναν πλήρη οδηγό όσον αφορά τις λειτουργίες της εφαρμογής, αλλά και μέσω αυτού να γίνει μια εκτενής παρουσίαση των τριών προσομοιώσεων. Μέσα από ένα σύνολο εικόνων και περιγραφών που παρατίθενται, πραγματοποιείται η πλήρης ανάπτυξη της πληροφορίας, σχετικά με βασικά βήματα που πρέπει να κάνει ένας χρήστης, που εισέρχεται για πρώτη φορά στο περιβάλλον αυτό.

Στα πλαίσια των καινούργιων στοιχείων που εισάγονται μέσω της εργασίας αυτής, είναι οι αλλαγές στους υφιστάμενους κώδικες και η επισήμανση τους με διακριτό τρόπο. Οι προβληματισμοί και τα εμπόδια που παρουσιάστηκαν αποτελούν επίσης μέρος αυτής της διπλωματικής εργασίας, καθώς η επίλυση αυτών ήταν παράγωγο της προσπάθειας αυτής.

### 6.1 Εγκατάσταση Contiki

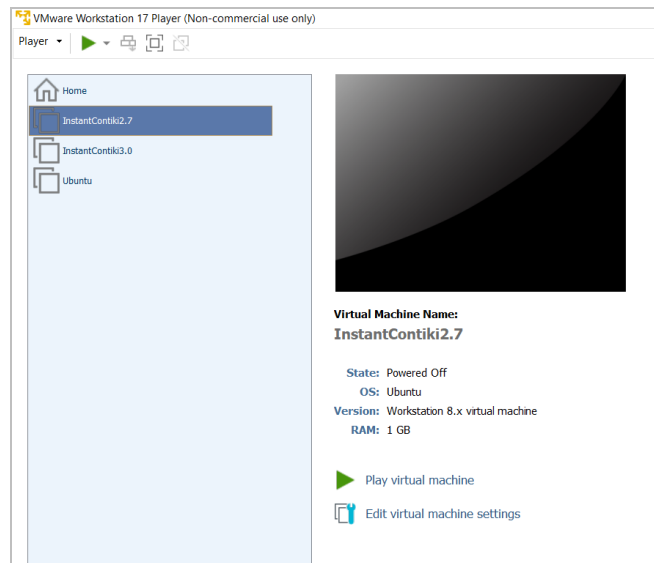
Η εγκατάσταση του προγράμματος χαρακτηρίζεται από ευκολία, ακολουθώντας τα βήματα του οδηγού. Η εφαρμογή μπορεί εύκολα να βρεθεί και να ληφθεί από το διαδίκτυο, με μια απλή αναζήτηση της. Η εικόνα 6.1 αποτυπώνει τις εκδόσεις της εφαρμογής, καθώς και την έκδοση που τελικώς προτιμήθηκε[66].

Στην εργασία αυτή προτιμήθηκε η έκδοση του Contiki 2.7 και όχι η έκδοση 3.0, διότι η νεότερη έκδοση παρουσίασε έντονα προβλήματα στην λειτουργία της. Πιο συγκεκριμένα, η πρόσβαση στο αρχικό περιβάλλον χρήστη δεν ήταν εφικτή ύστερα από την εγκατάστασή της.



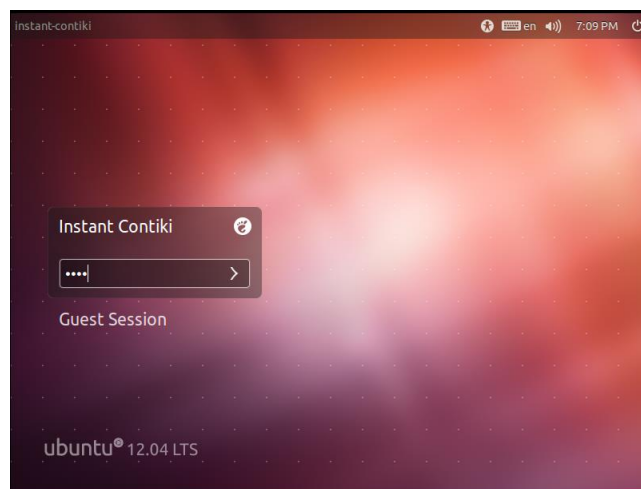
Εικόνα 6.1: Λήψη αρχείου – Επιλογή έκδοσης Contiki 2.7

Η εφαρμογή αυτή μπορεί να εκτελεστεί μόνο σε Linux και πιο συγκεκριμένα στην διανομή Ubuntu. Στην περίπτωση που υπάρχει ηλεκτρονικός υπολογιστής με διαφορετικό λειτουργικό σύστημα (Windows, MAC OS), θα χρειαστεί η χρήση προγράμματος όπου θα παρέχει μια εικονική μηχανή (Virtual Machine, VM) . Μέσω της χρήσης εργαλείων όπως το VMWare Player ή το Virtual Box, μπορεί σε επίπεδο εφαρμογής να εκτελεστεί το λειτουργικό σύστημα το Linux και μέσω αυτού η εφαρμογή Contiki. Κατά τη διάρκεια επιλογής των χαρακτηριστικών που χρειάζεται να έχει το συγκεκριμένο Virtual Machine (επεξεργαστική ισχύς, μνήμη, χώρος στο δίσκο), τα προτεινόμενα χαρακτηριστικά κατά την έναρξη της εφαρμογής κρίνονται ως ικανοποιητικά. Στην περίπτωση που οι απαιτήσεις του εκάστοτε χρήστη είναι μεγαλύτερες, μπορούν να γίνουν οι απαραίτητες ρυθμίσεις, όπως δείχνει η εικόνα 6.2.



**Εικόνα 6.2: Είσοδος στο VMWare Player – Επιλογή Contiki 2.7**

Για την εκτέλεση του προγράμματος προσομοιωτή, χρειάζεται να φορτωθεί η εικονική μηχανή με το λειτουργικό σύστημα όπως παρουσιάζεται μέσω της εικόνας 6.3.



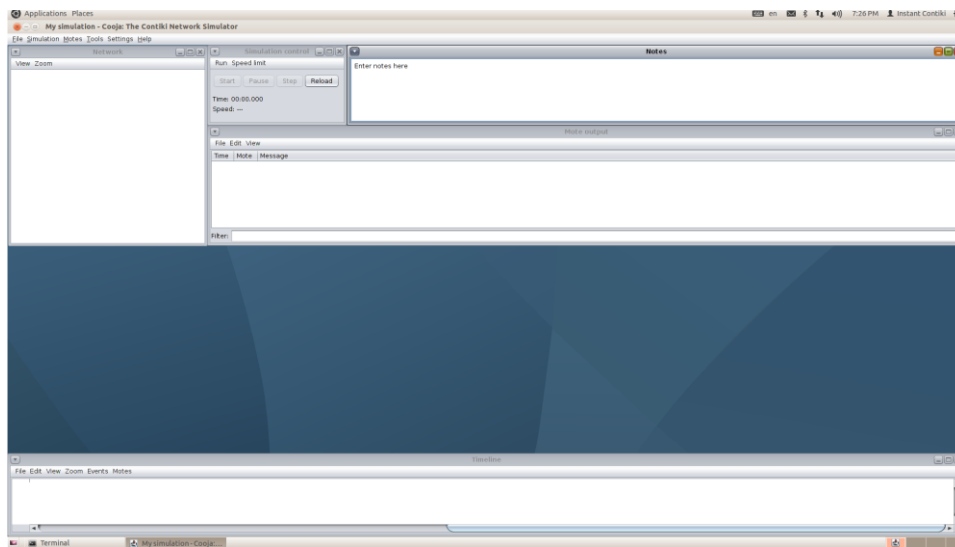
### Εικόνα 6.3: Αρχική οθόνη λειτουργικού

Υπάρχουν δυο επιλογές: Η μια, με την απευθείας εκτέλεση της εφαρμογής από την επιφάνεια εργασίας επιλέγοντας το εικονίδιο Cooja και η δεύτερη με τη χρήση των κατάλληλων εντολών μέσω του τερματικού εντολών (terminal, εικόνα 6.4).

```
user@ubuntu:~/home/user/contiki/tools/cooja$ ant run
```

### Εικόνα 6.4: Εντολή ανοίγματος εφαρμογής μέσω terminal

Δύναται πλέον να πραγματοποιηθεί οποιαδήποτε εργασία αφού δημιουργηθεί πρώτα και δοθεί ονομασία σε ένα νέο Simulation. Περισσότερες πληροφορίες ωστόσο δίνονται παρακάτω και κατά την ανάπτυξη των σεναρίων.



### Εικόνα 6.5: Περιβάλλον υποδοχής νέου Simulation

## 6.2 Σενάριο 1<sup>ο</sup>

Στο πρώτο σενάριο όπως και στα υπόλοιπα η ανάπτυξη της εργασίας επιτυγχάνεται με το Sky Mote.

Με βάση το παράδειγμα το οποίο αφορά το άναμμα των LED μέσω του πατήματος ενός εικονικού κουμπιού (button), θα υλοποιηθεί η πρώτη επεξεργασία. Πιο συγκεκριμένα επηρεάζοντας τον ήδη υπάρχον κώδικα του αντίστοιχου παραδείγματος με νέες εντολές οι οποίες θα παρατεθούν αντιστοίχως θα πραγματοποιηθεί μια αλλαγή στην εντολή. Αναλυτικότερα η αλλαγή αυτή θα περιλαμβάνει το άναμμα του κόκκινου LED ανάμεσα στα τρία, ( Πράσινο – Μπλε – Κόκκινο ) με το πάτημα του button. Εφόσον πλέον αυτό παραμένει αναμμένο με τις όποιες διεργασίες ενδέχεται να τρέχουν στο παρασκήνιο, τη χρονική στιγμή που θα αποφασιστεί το σβήσιμό του με το πάτημα του button, θα παρουσιαστεί στιγμιαία το άναμμα και των τριών (3) LED.

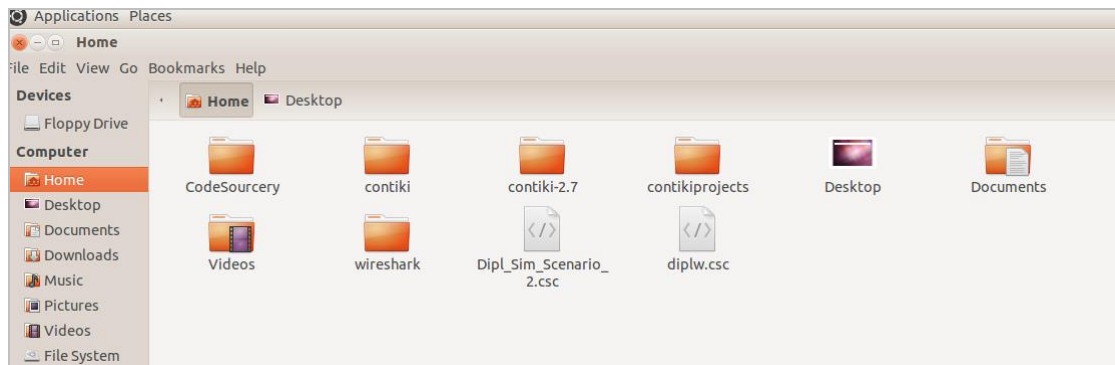
## 6.2.1 Προετοιμασία Σεναρίου 1

Η διεργασία ξεκινά όπως φαίνεται στη φωτογραφία 6.5 στο περιβάλλον υποδοχής της εφαρμογής, αφού έχει επιλεγεί η δημιουργία μιας νέας προσομοίωσης.

## 6.2.2 Αλλαγές

Αρχικά πρέπει να γίνουν οι απαραίτητες ενέργειες στις βιβλιοθήκες για να ληφθεί το επιθυμητό αποτέλεσμα.

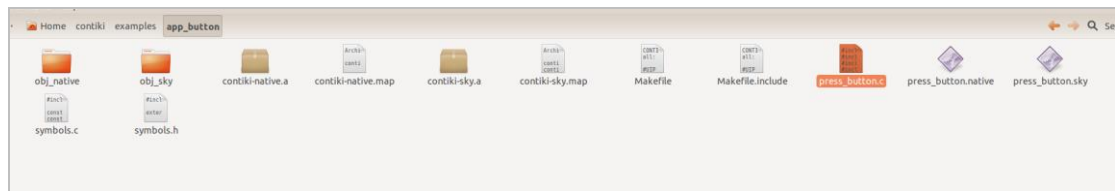
Αρχικά ανοίγοντας τους φακέλους της εφαρμογής γίνεται είσοδος στον φάκελο contiki όπως φαίνεται στην εικόνα 6.6. Ύστερα στον φάκελο examples δημιουργείται ένας νέος φάκελος με όνομα app\_button όπως παρουσιάζεται στην εικόνα 6.7. Εν συνεχεία δημιουργείται ένα text αρχείο press\_button το εμφανίζεται μέσω της εικόνας 6.8. Σε αυτό θα γραφτεί το κείμενο του κώδικα το οποίο θα κληθεί σε μετέπειτα στάδιο.



Εικόνα 6.6: Είσοδος στο φάκελο Home – Επιλογή Contiki



Εικόνα 6.7: Δημιουργία φακέλου app\_button



Εικόνα 6.8: Δημιουργία text αρχείου press\_button

Με την επιστροφή εκ νέου στον φάκελο contiki, γίνεται είσοδος στον φάκελο dev και άνοιγμα του αρχείου button-sensor.h. Μέσα από εκείνον δίνεται η κατεύθυνση για το αρχείο sensors.h το οποίο βρίσκεται στο φάκελο libόπως φαίνεται στην εικόνα 6.9.

```
button-sensor.h x
/*
 * Copyright (c) 2005, Swedish Institute of Computer Science
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the Institute nor the names of its contributors
 * may be used to endorse or promote products derived from this software
 * without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS ``AS IS'' AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 *
 * This file is part of the Contiki operating system.
 */
#ifdef __BUTTON_SENSOR_H__
#define __BUTTON_SENSOR_H__

#include "lib/sensors.h"

extern const struct sensors_sensor button_sensor;

#define BUTTON_SENSOR "Button"
#endif /* __BUTTON_SENSOR_H__ */
```

Εικόνα 6.9: Μέσα στο αρχείο button-sensor.h

Στο αρχείο `sensors.h`, λαμβάνονται όλες οι απαραίτητες πληροφορίες για το `button sensor` και το ποιες είναι οι απαραίτητες ενέργειες προκειμένου να γίνει `activate`. Η εν λόγω διαδικασία αναλύεται μέσω της εικόνας 6.10.

```
#ifndef __SENSORS_H__
#define __SENSORS_H__

#include "contiki.h"

/* some constants for the configure API */
#define SENSORS_HW_INIT 128 /* internal - used only for initialization */
#define SENSORS_ACTIVE 129 /* ACTIVE => 0 -> turn off, 1 -> turn on */
#define SENSORS_READY 130 /* read only */

#define SENSORS_ACTIVATE(sensor) (sensor).configure(SENSORS_ACTIVE, 1)
#define SENSORS_DEACTIVATE(sensor) (sensor).configure(SENSORS_ACTIVE, 0)

#define SENSORS_SENSOR(name, type, value, configure, status) \
const struct sensors_sensor name = { type, value, configure, status }

#define SENSORS_NUM (sizeof(sensors) / sizeof(struct sensors_sensor *))

#define SENSORS(...) \
const struct sensors_sensor *sensors[] = {__VA_ARGS__, NULL}; \
unsigned char sensors_flags[SENSORS_NUM]

struct sensors_sensor {
char * type;
int (* value) (int type);
int (* configure) (int type, int value);
int (* status) (int type);
};

const struct sensors_sensor *sensors_find(const char *type);
const struct sensors_sensor *sensors_next(const struct sensors_sensor *s);
const struct sensors_sensor *sensors_first(void);

void sensors_changed(const struct sensors_sensor *s);

extern process_event_t sensors_event;

PROCESS_NAME(sensors_process);

#endif /* __SENSORS_H__ */
```

**Εικόνα 6.10: Μέσα στο αρχείο `sensors.h`**

Ακολουθεί η σύνταξη του `text` αρχείου `press_button` το οποίο δημιουργήθηκε νωρίτερα.

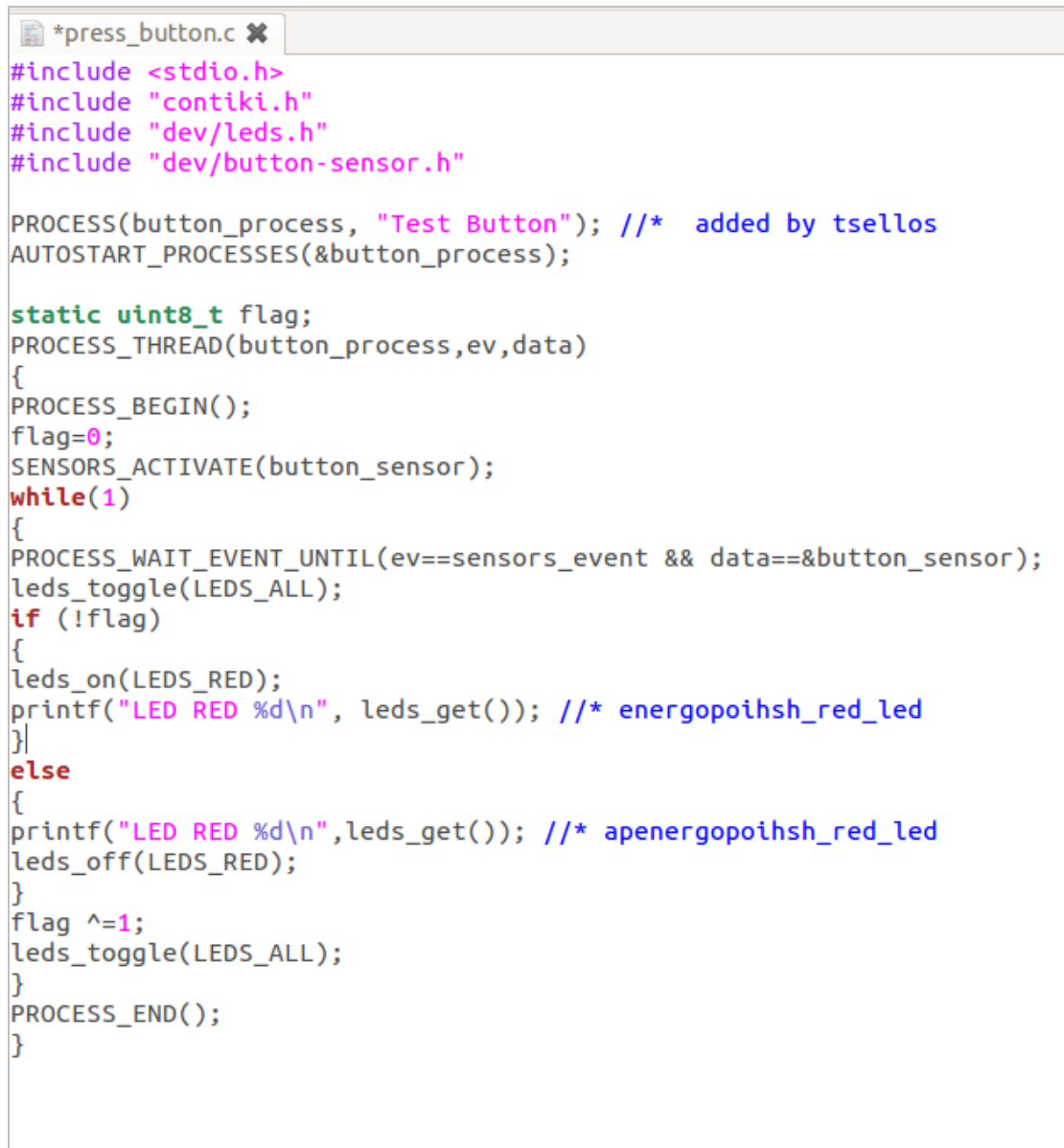
Στο σημείο αυτό είναι αναγκαίο να τονιστεί, αν και αναπτύσσεται, εκτενέστερα σε άλλο κεφάλαιο, πως στην αναζήτηση για τη λειτουργία του Contiki η βιβλιογραφία να μην συνέβαλε σημαντικά στο θεωρητικό μέρος, αλλά όχι και στο πρακτικό. Για το πρακτικό μέρος πραγματοποιήθηκε μια ευρεία αναζήτηση στο διαδίκτυο και βρέθηκε λύση μέσα από διάφορα Forums, καθώς ο Open Source χαρακτήρας της εφαρμογής δίνει τη δυνατότητα για ευρεία συζήτηση ανάμεσα σε ενδιαφερόμενους.

Η βάση για τον κώδικα αυτόν προήλθε από αναζήτηση, με πυλώνα πολλές φορές τη σελίδα GitHub[67]. Χαρακτηριστικό είναι το παράδειγμα πως από την απαρχή της



συγγραφής της εργασίας έως σήμερα, κάποια από τα αναρτημένα αρχεία, πολλές φορές εμφανίστηκαν και εξαφανίστηκαν, με αποτέλεσμα να δυσκολεύεται η αναζήτηση τους.

Στην Εικόνα 6.11 που ακολουθεί μπορεί να γίνει ορατό το κείμενο του κώδικα και τις αλλαγές που έγιναν, κυρίως όσον αφορά την αλλαγή στο κόκκινο χρώμα του LED.



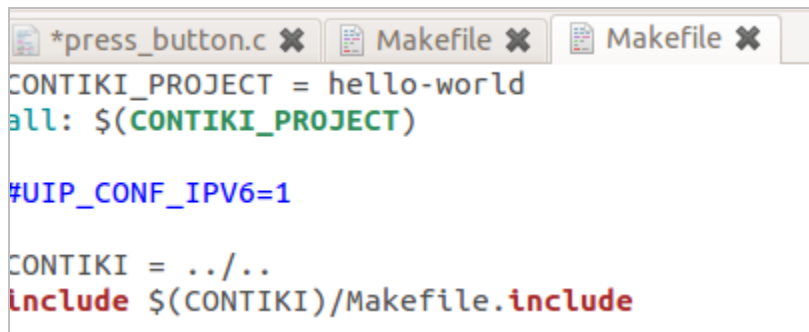
```
*press_button.c ✕
#include <stdio.h>
#include "contiki.h"
#include "dev/leds.h"
#include "dev/button-sensor.h"

PROCESS(button_process, "Test Button"); /* added by tsellos
AUTOSTART_PROCESSES(&button_process);

static uint8_t flag;
PROCESS_THREAD(button_process, ev, data)
{
PROCESS_BEGIN();
flag=0;
SENSORS_ACTIVATE(button_sensor);
while(1)
{
PROCESS_WAIT_EVENT_UNTIL(ev==sensors_event && data==&button_sensor);
leds_toggle(LED_ALL);
if (!flag)
{
leds_on(LED_RED);
printf("LED RED %d\n", leds_get()); /* energopoihsh_red_led
}
else
{
printf("LED RED %d\n", leds_get()); /* apenergopoihsh_red_led
leds_off(LED_RED);
}
flag ^=1;
leds_toggle(LED_ALL);
}
PROCESS_END();
}
```

**Εικόνα 6.11: Κείμενο κώδικα press\_button**

Πραγματοποιείται μια ακόμη επεξεργασία στο αρχείο Makefile, του οποίου ένα πρότυπο μπορεί να ληφθεί από το παράδειγμα Hello World, όπως φαίνεται στην εικόνα 6.12.



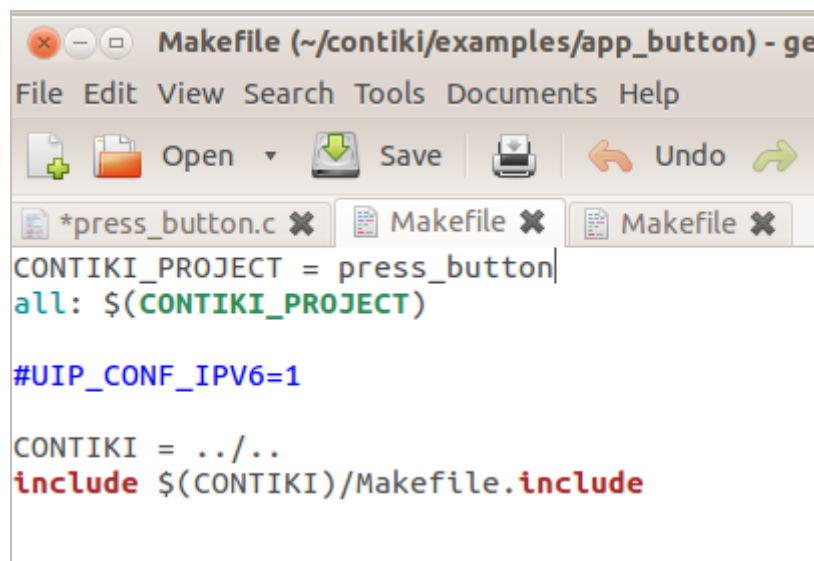
```
*press_button.c x Makefile x Makefile x
CONTIKI_PROJECT = hello-world
all: $(CONTIKI_PROJECT)

#UIP_CONF_IPV6=1

CONTIKI = ../..
include $(CONTIKI)/Makefile.include
```

**Εικόνα 6.12: Πρότυπο Makefile**

Υλοποιώντας την απαραίτητη επεξεργασία στη θέση του hello-world τοποθετείται το όνομα του νέου αρχείου press\_button, όπως παρουσιάζεται στην εικόνα 6.13



```
Makefile (~/.contiki/examples/app_button) - ge
File Edit View Search Tools Documents Help
+ Open Save Undo
*press_button.c x Makefile x Makefile x
CONTIKI_PROJECT = press_button
all: $(CONTIKI_PROJECT)

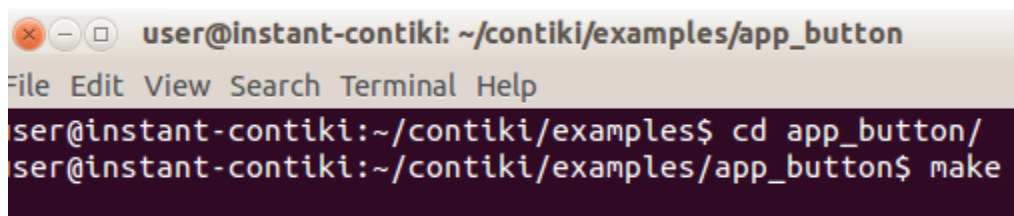
#UIP_CONF_IPV6=1

CONTIKI = ../..
include $(CONTIKI)/Makefile.include
```

**Εικόνα 6.13: Επεξεργασμένο Makefile**

Επόμενο σταθμό αποτελεί το terminal, καθώς πρέπει πλέον να γίνει ενεργοποίηση του καινούργιου σεναρίου.

Πληκτρολογούνται οι εντολές της Εικόνας 6.14.



```
user@instant-contiki: ~/contiki/examples/app_button
File Edit View Search Terminal Help
ser@instant-contiki:~/contiki/examples$ cd app_button/
ser@instant-contiki:~/contiki/examples/app_button$ make
```

**Εικόνα 6.14: Εντολή make στο terminal**

Αφού είναι σίγουρο πως δεν έχει γίνει κάποιο συντακτικό λάθος και δεν παρουσιάζεται κάποιο σφάλμα κατά τη διάρκεια των διεργασιών στο terminal, τότε στον καινούργιο φάκελο που θα εμφανιστούν επιμέρους αρχεία και φάκελοι όπως αποτυπώνεται στην εικόνα 6.15.

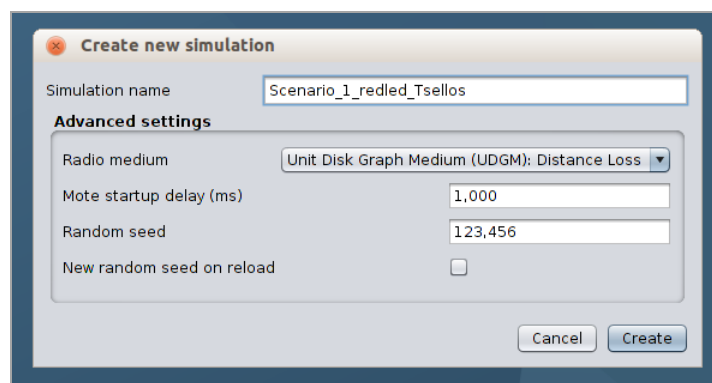


**Εικόνα 6.15: Δημιουργία απαραίτητων αρχείων**

Πλέον είναι όλα έτοιμα για την είσοδο εκ νέου στο περιβάλλον του Contiki για την παρακολούθηση του αποτελέσματος.

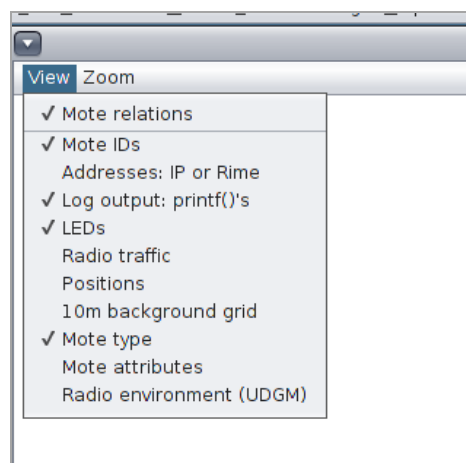
### 6.2.3 Εφαρμογή Σεναρίου

Στην εικόνα 6.16, δημιουργείται μια προσομοίωση με το επιθυμητό όνομα.



**Εικόνα 6.16: Δημιουργία προσομοίωσης**

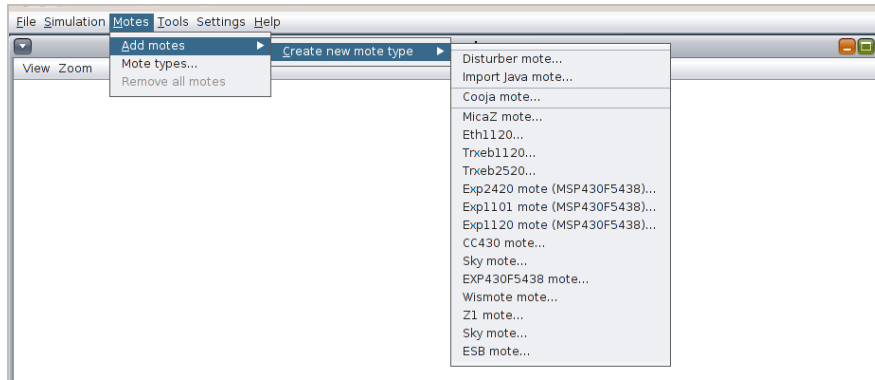
Στην καρτέλα View (η οποία χρησιμοποιείται περισσότερο στη συνέχεια) επιλέγονται επιλογές που παρουσιάζονται στην εικόνα 6.17.



**Εικόνα 6.17: Επιλογές καρτέλας View**

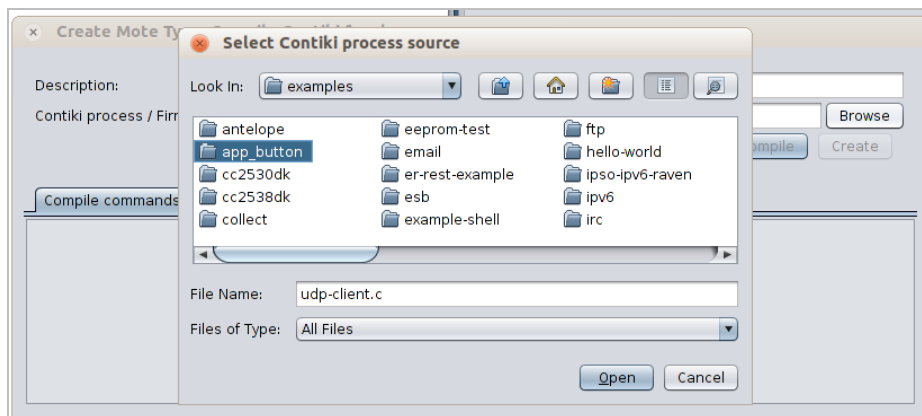
Ακολουθείται η σειρά Motes → Add motes → Create new mote type

Η σειρά αυτή ακολουθείται για οποιοδήποτε mote υπάρξει χρειάζεται να προστεθεί για οποιαδήποτε εργασία. Στην εικόνα 6.18 επιλέγεται εκ νέου τοSky Mote.



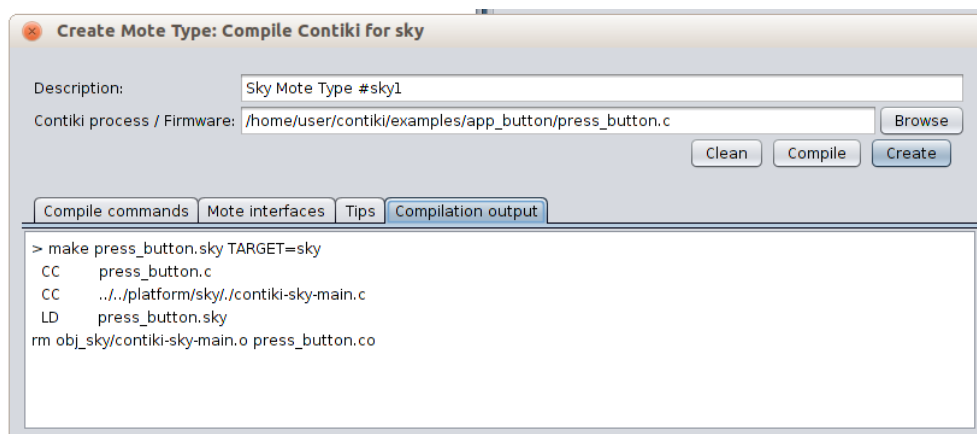
**Εικόνα 6.18: Επιλογή Sky Mote**

Στο παράθυρο που εμφανίζεται επιλέγεται η επιλογή Browse. Στην εικόνα 6.19, κάνοντας αναζήτηση στους φακέλους της εφαρμογής εντοπίζεται και επιλέγεται εκείνος που έχει δημιουργηθεί, δηλαδή ο press\_button.



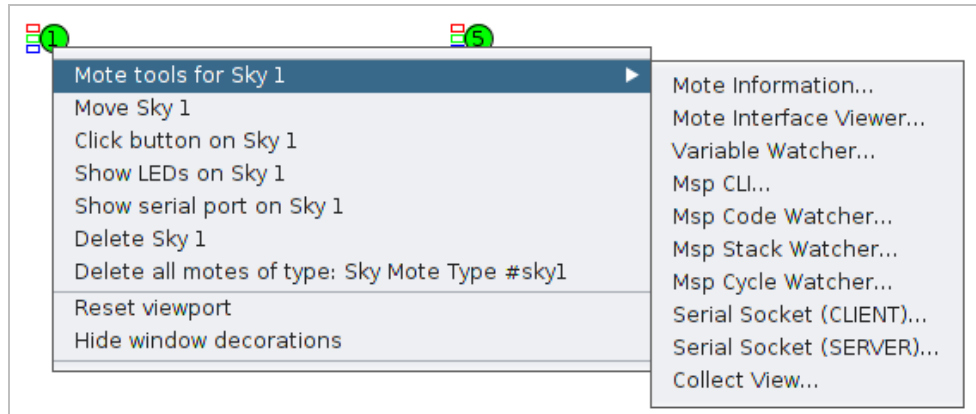
**Εικόνα 6.19: Επιλογή app\_button**

Επιλέγεται το καινούργιο αρχείο press\_button και γίνεται compile. Στην εικόνα 6.20, αφού καταστεί σαφές ότι όλα βγαίνουν καλώς δύναται πλέον να πατηθεί το πλήκτρο create, και να επιλεγεί ο αριθμός των motes τα οποία χρειάζονται να προστεθούν.



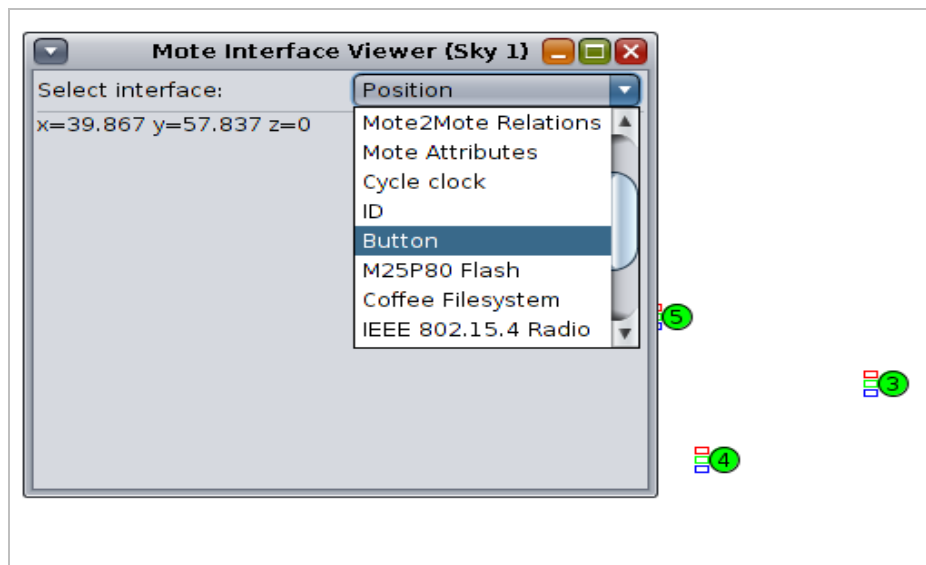
**Εικόνα 6.20: Επιτυχές compile – Create**

Δημιουργούνται πέντε (5) motes, αν και δεν είναι απαραίτητο για το συγκεκριμένο παράδειγμα, και δίνεται συνέχεια με δεξί κλικ σε ένα οποιοδήποτε mote. Επιλέγονται με τη σειρά Mote tools for Sky 1 → Mote Interface Viewer, όπως αποτυπώνεται στην εικόνα 6.21.

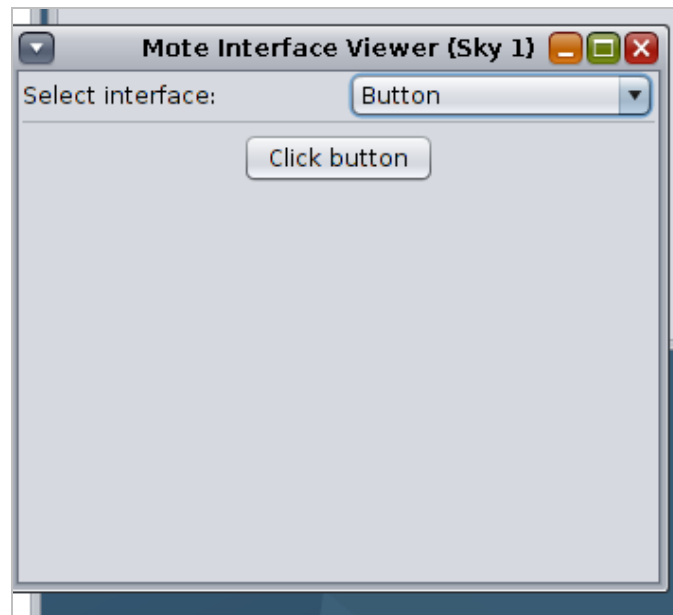


**Εικόνα 6.21: Επιλογή Mote Interface Viewer**

Για να εμφανιστεί το εικονικό κουμπί πατώντας το βέλος γίνεται επιλογή του Button και να εμφανιστεί το αποτέλεσμα. Οι διαδικασίες παρουσιάζονται στις εικόνες 6.22 και 6.23.

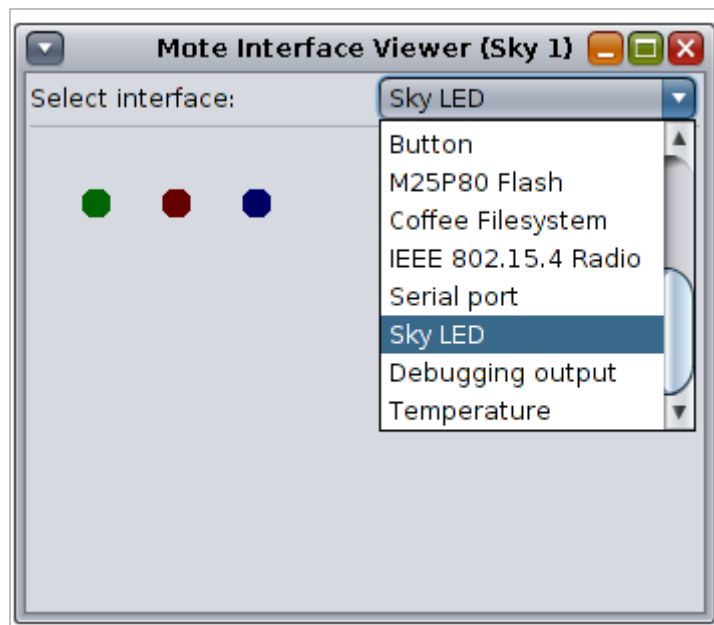


**Εικόνα 6.22: Επιλογή Button**



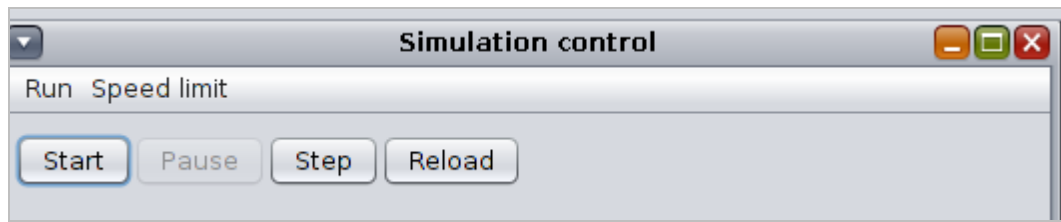
**Εικόνα 6.23: Εμφάνιση παραθύρου Button**

Για να εμφανιστεί το παράθυρο των LED, ακολουθείται η ίδια διαδικασία και πιέζεται αντιστοίχως η επιλογή Sky LED. Στην εικόνα 6.24, παρατηρείται ένα παράθυρο στο οποίο απεικονίζονται τα LED τριών (3) χρωμάτων όπως έχει ήδη ειπωθεί Πράσινο, Κόκκινο και Μπλε.



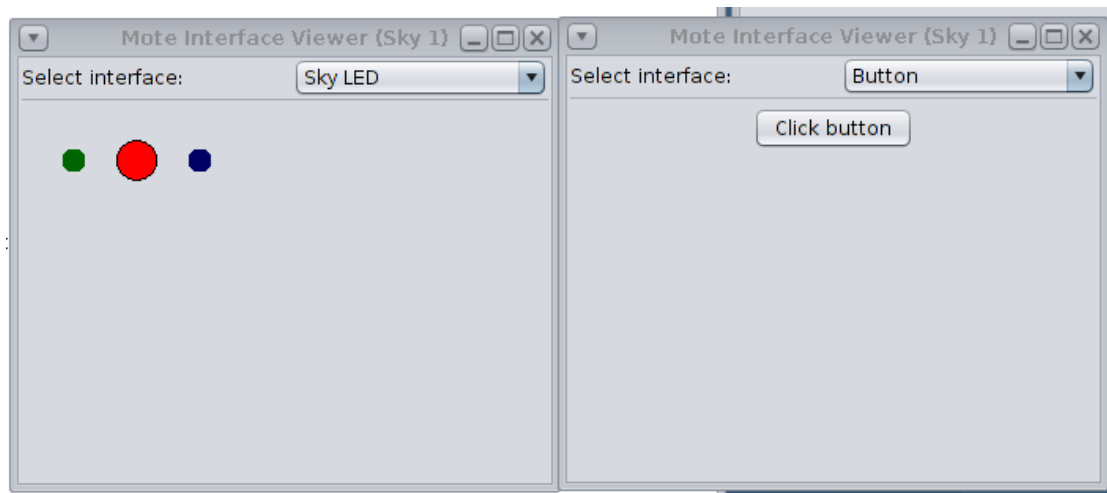
**Εικόνα 6.24: Εμφάνιση παραθύρου Sky LED**

Σύμφωνα με την εικόνα 6.25, για να παρουσιαστεί το αποτέλεσμα πρέπει να επιλεγεί το run από το παράθυρο simulation control.



**Εικόνα 6.25: Εντολή Start**

Πιέζεται το Click Button και παρατηρείται το κόκκινο LED να ανάβει, όπως φαίνεται στην εικόνα 6.26



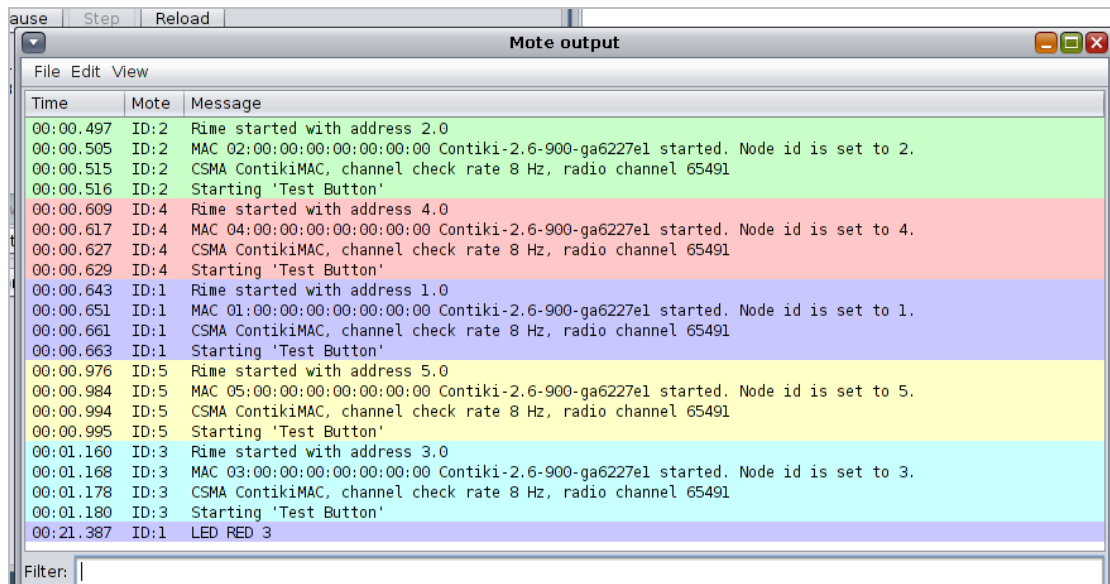
**Εικόνα 6.26: Αναμμένο κόκκινο LED**

Δύναται ωστόσο να παρατηρηθεί το κόκκινο LED να ανάβει και από το παράθυρο Network στο οποίο και έχουν τοποθετηθεί όλα τα motes μέσω της επιλογής LEDS από το μενού View. Η δυνατότητα αυτή παρουσιάζεται στην εικόνα 6.27.



**Εικόνα 6.27: Αναμμένο κόκκινο LED – Παράθυρο Network**

Μπορεί επιπροσθέτως να αντληθεί ένας μεγάλος όγκος πληροφοριών και από το παράθυρο Mote output, οι οποίες μπορούν να μας φανούν ιδιαίτερες χρήσιμες, ανάλογα με την εργασία που καλείται ο εκάστοτε χρήστης να επιτελέσει. Η παρουσίασή τους γίνεται στην εικόνα 6.28.



Time	Mote	Message
00:00.497	ID:2	Rime started with address 2.0
00:00.505	ID:2	MAC 02:00:00:00:00:00 Contiki-2.6-900-ga6227e1 started. Node id is set to 2.
00:00.515	ID:2	CSMA ContikiMAC, channel check rate 8 Hz, radio channel 65491
00:00.516	ID:2	Starting 'Test Button'
00:00.609	ID:4	Rime started with address 4.0
00:00.617	ID:4	MAC 04:00:00:00:00:00 Contiki-2.6-900-ga6227e1 started. Node id is set to 4.
00:00.627	ID:4	CSMA ContikiMAC, channel check rate 8 Hz, radio channel 65491
00:00.629	ID:4	Starting 'Test Button'
00:00.643	ID:1	Rime started with address 1.0
00:00.651	ID:1	MAC 01:00:00:00:00:00 Contiki-2.6-900-ga6227e1 started. Node id is set to 1.
00:00.661	ID:1	CSMA ContikiMAC, channel check rate 8 Hz, radio channel 65491
00:00.663	ID:1	Starting 'Test Button'
00:00.976	ID:5	Rime started with address 5.0
00:00.984	ID:5	MAC 05:00:00:00:00:00 Contiki-2.6-900-ga6227e1 started. Node id is set to 5.
00:00.994	ID:5	CSMA ContikiMAC, channel check rate 8 Hz, radio channel 65491
00:00.995	ID:5	Starting 'Test Button'
00:01.160	ID:3	Rime started with address 3.0
00:01.168	ID:3	MAC 03:00:00:00:00:00 Contiki-2.6-900-ga6227e1 started. Node id is set to 3.
00:01.178	ID:3	CSMA ContikiMAC, channel check rate 8 Hz, radio channel 65491
00:01.180	ID:3	Starting 'Test Button'
00:21.387	ID:1	LED RED 3

Εικόνα 6.28: Παράθυρο Mote output

Εδώ κλείνει η παρουσίαση του πρώτου σεναρίου. Τα συμπεράσματα θα παρατεθούν μαζί με τα υπόλοιπα το κεφάλαιο των Συμπερασμάτων.

### 6.3 Σενάριο 2<sup>ο</sup>

Στο παρόν σενάριο χρησιμοποιούνται οι δυνατότητες που δίνονται από το Sky Mote για να δημιουργηθεί μια εικονική συλλογή πληροφοριών.

#### 6.3.1 Προετοιμασία Σεναρίου 2

Η συλλογή αυτή των δεδομένων, γίνεται χρησιμοποιώντας το παράδειγμα του border.router. Κατά το σενάριο αυτό ορίζεται ένα mote ως border router προκειμένου να επιτευχθεί η επικοινωνία μας μέσω της εφαρμογής με τον έξω κόσμο. Τα υπόλοιπα motes τίθενται ως συσκευές οι οποίες είναι αρμόδιες για αυτή τη συλλογή πληροφοριών.

Ωστόσο τα χαρακτηριστικά των motes αυτών, είναι κατά τέτοιο τρόπο ρυθμισμένα ούτως ώστε σε περίπτωση που δεν πραγματοποιηθεί παρεμβολή, να ληφθούν αποτελέσματα μόνο όσον αφορά τη θερμοκρασία και το φως.

Στο σημείο αυτό γίνεται παρέμβαση. Σύμφωνα με τις διεργασίες που πραγματοποιούνται στα ήδη υπάρχοντα κείμενα των κωδίκων στις βιβλιοθήκες του Contiki, προστίθενται στα αποτελέσματα των motes και εκείνα της Υγρασίας, του Βάρους και της Πίεσης.

Αναφορικά με τη μέτρηση της Υγρασίας, το Tmote Sky φέρεται να το υποστηρίζει στην πραγματικότητα και όπως θα φανεί και στις φωτογραφίες, μέσα στις βιβλιοθήκες, προδιαγράφεται για προσθήκη, ασχέτως του ότι δεν είναι ενεργοποιημένο να χρησιμοποιείται και να παρουσιάζεται στο τελικό αποτέλεσμα.

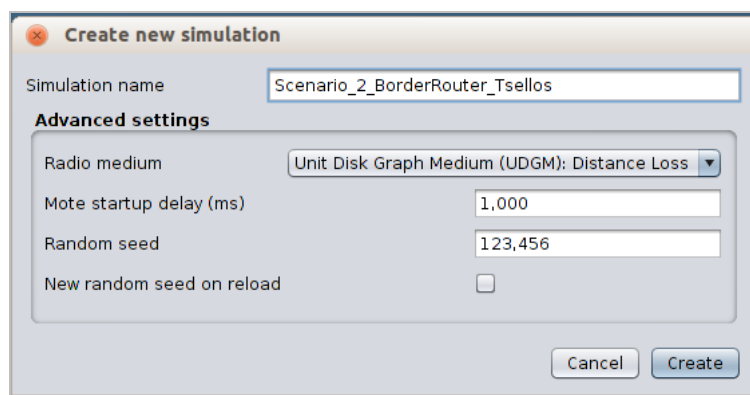


Η μέτρηση του βάρους είναι κάτι το οποίο δεν υποστηρίζεται στην πραγματικότητα από Tmote Sky, ωστόσο στα πλαίσια της έρευνας έγινε η υπόθεση πως δύναται να το υποστηρίξει, προκειμένου να προστεθεί μια ακόμη μέτρηση στο τελικό αποτέλεσμα και να υποδειχθεί η προσθήκη ενός μεγέθους, end to end.

Η μέτρηση της πίεσης επίσης ακολουθεί τη λογική της μέτρησης βάρους, στα πλαίσια μιας “θεωρητικής” προσθήκης για την εξαγωγή εικονικών αποτελεσμάτων.

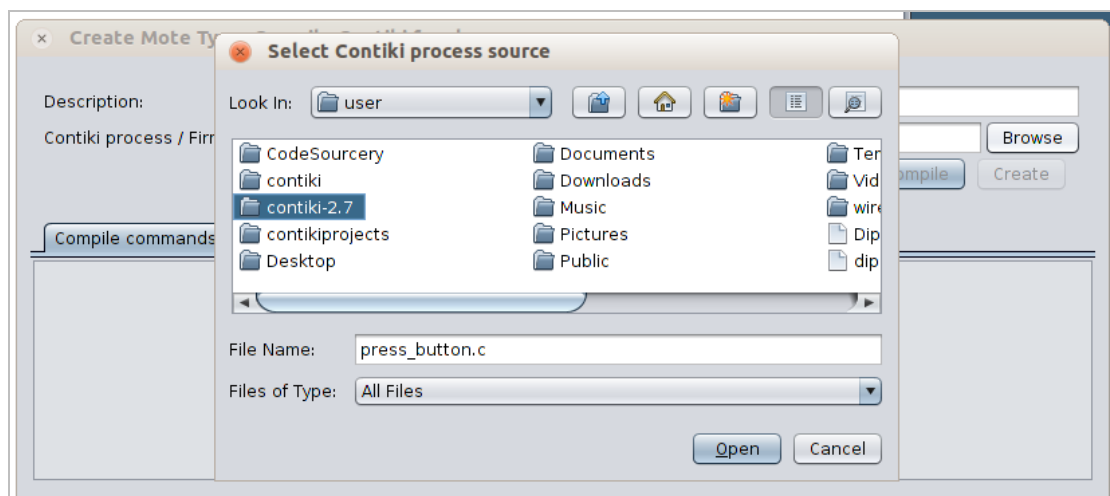
### 6.3.2 Παρούσα κατάσταση

Στην εικόνα 6.29, δημιουργείται μια καινούργια προσομοίωση για τις απαιτήσεις του σεναρίου.

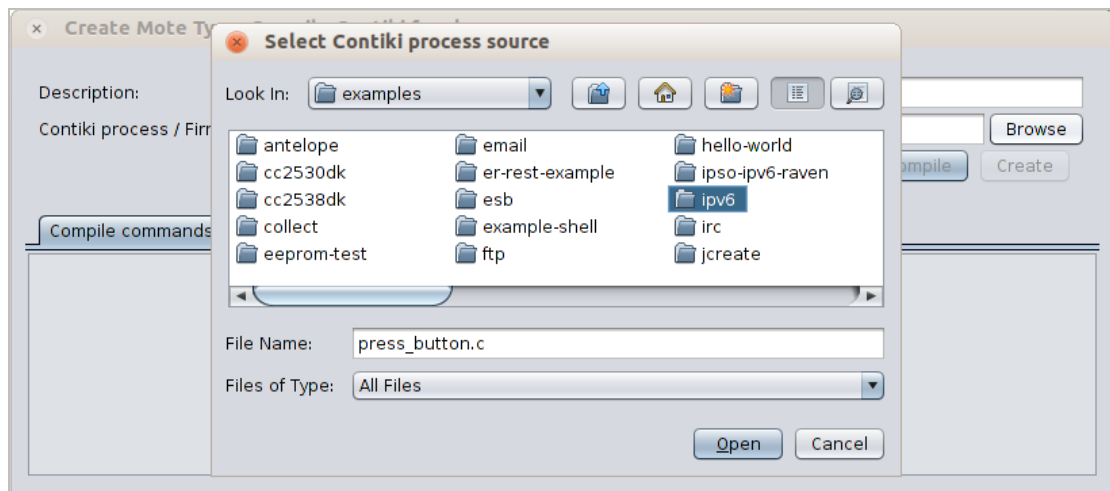


Εικόνα 6.29: Δημιουργία νέας προσομοίωσης

Στην καινούργια αυτή διαδικασία προστίθεται ένας τύπος mote ο οποίος θα είναι ο border-router. Χρειάζεται να δοθεί προσοχή κατά το αναζήτηση για το άνοιγμα του αρχείου, να έχει γίνει έξοδος από τον φάκελο contiki και είσοδος στον αντίστοιχο contiki-2.7, όπως φαίνεται στην εικόνα 6.30, με επιλογή του φακέλου IPv6, μέσα από τον φάκελο examples, όπως επίσης φαίνεται στην εικόνα 6.31.

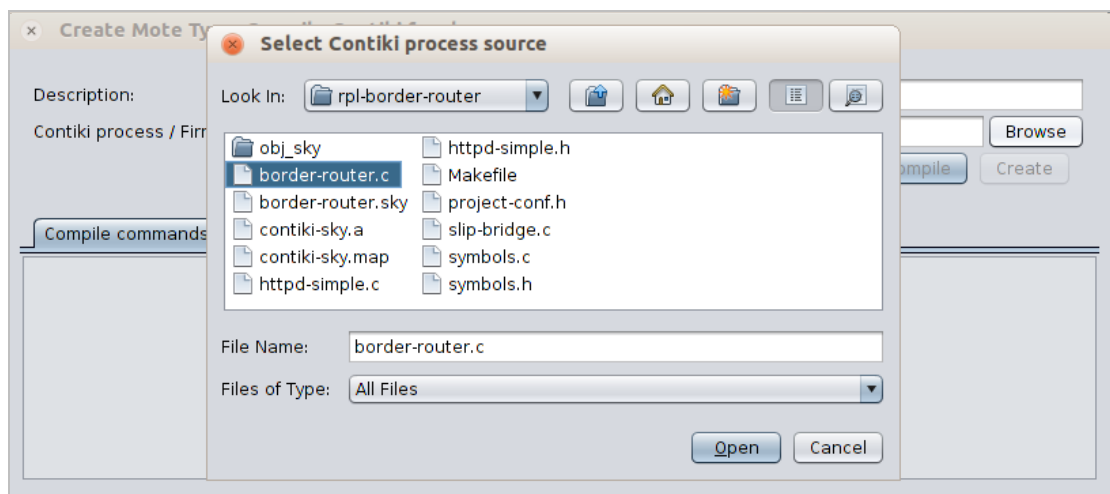


Εικόνα 6.30: Επιλογή Contiki-2.7

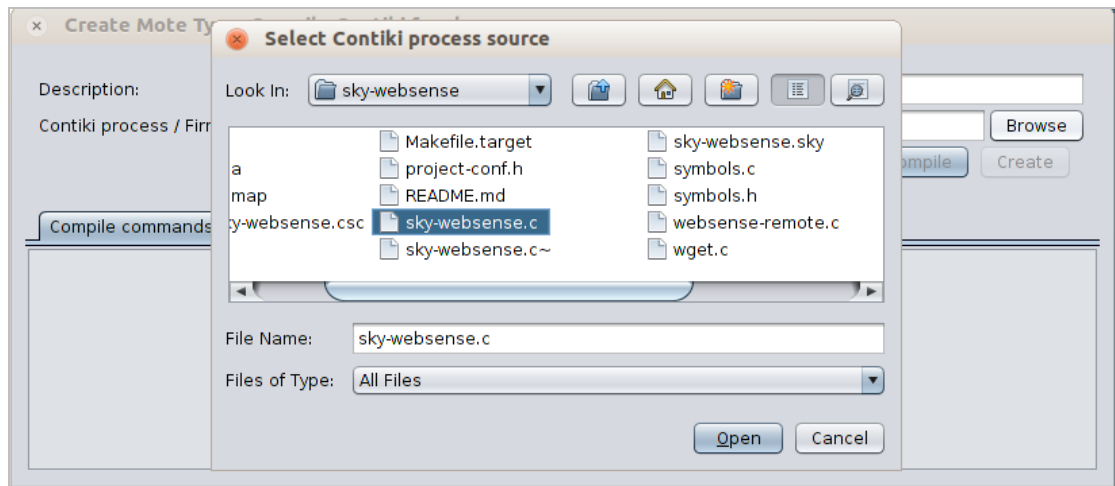


**Εικόνα 6.31: Επιλογή IPv6**

Στην εικόνα 6.32, γίνεται εισαγωγή του moteborder-router, και του mote skywebsense.c από τον φάκελο skywebsense, το οποίο χρησιμεύει στην προβολή των αποτελεσμάτων.

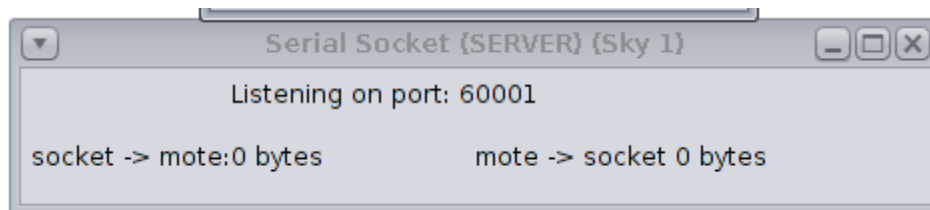


**Εικόνα 3.32: Επιλογή border-router.c**



**Εικόνα 6.33: Επιλογή sky-websense.c**

Στην εικόνα 6.34, για την εκτέλεση της εντολής run αυτό το Simulation, γίνεται αντιληπτό πως δεν έχει γίνει το άνοιγμα του border-router γι’ αυτό και παρατηρείται πως δεν πραγματοποιείται ανταλλαγή δεδομένων.

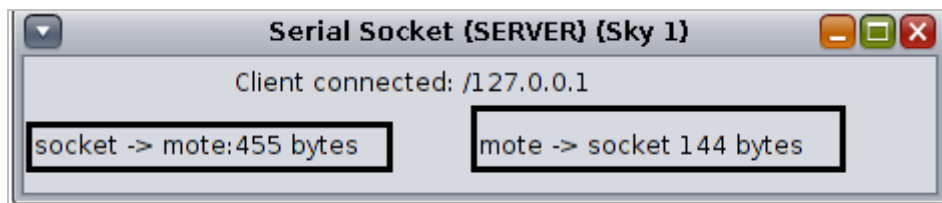


**Εικόνα 6.34: Μη ανταλλαγή δεδομένων**

Ανοίγοντας το terminal πληκτρολογείται η εντολή της εικόνας 6.35 και γίνεται εμφανής πλέον η ανταλλαγή δεδομένων, όπως αποτυπώνεται στην εικόνα 6.36.

```
user@instant-contiki:~/contiki-2.7/examples/ipv6/rpl-border-router$ make connect-router-cooja
```

**Εικόνα 6.35: Εντολή terminal – Έναρξη ανταλλαγής δεδομένων**



**Εικόνα 6.36: Έναρξη ανταλλαγής δεδομένων – Serial Socket (Server)**

Στην εικόνα 6.37, γίνεται λήψη από το terminal των τελευταίων δύο σειρών των IPV6 διευθύνσεων των συσκευών, προκειμένου μέσω ενός browser να παρατηρηθεί τι θα εμφανιστεί.

```
tun0      Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
inet addr:127.0.1.1 P-t-P:127.0.1.1 Mask:255.255.255.255
inet6 addr: fe80::1/64 Scope:Link
inet6 addr: aaaa::1/64 Scope:Global
UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:500
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

*** Address:aaaa::1 => aaaa:0000:0000:0000
Got configuration message of type P
Setting prefix aaaa::
Server IPv6 addresses:
  aaaa::212:7401:1:101
  fe80::212:7401:1:101
```

Εικόνα 6.37: Διευθύνσεις μέσω terminal

Στην εικόνα 6.38, πληκτρολογώντας την πρώτη διεύθυνση παρουσιάζεται ως αποτέλεσμα μια σελίδα στην οποία αποτυπώνονται όλες οι διευθύνσεις των motes . Επιλέγεται μια από τις διευθύνσεις αυτές, με σκοπό την επικέντρωση σε ένα εκ των motes ,με το αποτέλεσμα που λαμβάνεται να είναι οι τιμές του φωτός και της θερμοκρασίας, όπως ήταν άλλωστε και αναμενόμενο. Τα αποτελέσματα εμφανίζονται στην εικόνα 6.39.

```
Neighbors

fe80::212:7409:9:909
fe80::212:7406:6:606
fe80::212:740a:a:a0a
fe80::212:7403:3:303
fe80::212:7407:7:707
fe80::212:740b:b:b0b
fe80::212:7405:5:505
fe80::212:7408:8:808

Routes

aaaa::212:7409:9:909/128 (via fe80::212:7409:9:909) 16711413s
aaaa::212:7406:6:606/128 (via fe80::212:7406:6:606) 16711414s
aaaa::212:740a:a:a0a/128 (via fe80::212:740a:a:a0a) 16711412s
aaaa::212:7403:3:303/128 (via fe80::212:7403:3:303) 16711413s
aaaa::212:7407:7:707/128 (via fe80::212:7407:7:707) 16711412s
aaaa::212:740b:b:b0b/128 (via fe80::212:740b:b:b0b) 16711411s
aaaa::212:7405:5:505/128 (via fe80::212:7405:5:505) 16711413s
aaaa::212:7408:8:808/128 (via fe80::212:7408:8:808) 16711412s
```

Εικόνα 6.38: Διευθύνσεις motes μέσω Mozilla



Εικόνα 6.39: Αποτελέσματα ενός εκ των motes

### 6.3.3 Αλλαγές

Στις αλλαγές που θα χρειαστεί να γίνουν περιλαμβάνεται η επεξεργασία των παρακάτω αρχείων :

- Sky-websense.c
- Sht11.c
- Sht11.h
- Sht11-sensor.c
- Sht11-sensor.h

Με τα τέσσερα (4) τελευταία από αυτά να βρίσκονται στον υποφάκελο de του φακέλου core στο contiki2.7.

Στις εικόνες 6.40 – 6.48 παρατίθενται οι αλλαγές που έγιναν.

- Στο αρχείο Sky-websense.c :

```
#define HISTORY 16
static int temperature[HISTORY];
static int pressure[HISTORY]; /* added tsellos
static int humidity[HISTORY]; /* added tsellos
static int weight[HISTORY]; /* added tsellos
static int light1[HISTORY];
static int sensors_pos;
```

Εικόνα 6.40: 1η επεξεργασία Sky-websense

```
.....*/
static int
get_temp(void)
{
    return ((sht11_sensor.value(SHT11_SENSOR_TEMP) / 10) - 396) / 10;
}
/* added tsellos
static int
get_pressure(void)
{
    return ((sht11_sensor.value(SHT11_SENSOR_PRESSURE) / 10) - 396) / 10;
}
/* added tsellos
static int
get_humidity(void)
{
    return ((sht11_sensor.value(SHT11_SENSOR_HUMIDITY) / 10) - 396) / 10;
}
/* added tsellos
static int
get_weight(void)
{
    return ((sht11_sensor.value(SHT11_SENSOR_WEIGHT) / 10) - 396) / 10;
}
.....*/
```

Εικόνα 6.41: 2η επεξεργασία Sky-websense

```
static
PT_THREAD(send_values(struct httpd_state *s))
{
    PSOCK_BEGIN(&s->sout);

    SEND_STRING(&s->sout, TOP);

    if(strncmp(s->filename, "/index", 6) == 0 ||
        s->filename[1] == '\0') {
        /* Default page: show latest sensor values as text (does not
           require Internet connection to Google for charts). */
        blen = 0;
        ADD("<h1>Current readings</h1>\n"
            "Light: %u<br>"
            "Temperature: %u&deg; C\n",
            "Pressure: %u; <br> \n", /* added tsellos
            "Humidity: %u; <br> %\n", /* added tsellos
            "Weight: %u; <br> kg\n", /* added tsellos
            get_light(), get_temp(), get_pressure(), get_humidity(), get_weight());
        SEND_STRING(&s->sout, buf);
    }
}
```

Εικόνα 6.42: 3η επεξεργασία Sky-websense

```
PROCESS_THREAD(web_sense_process, ev, data)
{
    static struct etimer timer;
    PROCESS_BEGIN();

    sensors_pos = 0;

    etimer_set(&timer, CLOCK_SECOND * 2);
    SENSORS_ACTIVATE(light_sensor);
    SENSORS_ACTIVATE(sht11_sensor);

    while(1) {
        PROCESS_WAIT_EVENT_UNTIL(etimer_expired(&timer));
        etimer_reset(&timer);

        light[sensors_pos] = get_light();
        temperature[sensors_pos] = get_temp();
        pressure[sensors_pos] = get_pressure(); /* added tsellos
        humidity[sensors_pos] = get_humidity(); /* added tsellos
        weight[sensors_pos] = get_weight(); /* added tsellos
        sensors_pos = (sensors_pos + 1) % HISTORY;
    }

    PROCESS_END();
}
/*-----*/
```

Εικόνα 6.43: 4η επεξεργασία Sky-websense

- Στο αρχείο Sht11.c:

```
#ifndef SDA_0
#define SDA_0() (SHT11_PxDIR |= BV(SHT11_ARCH_SDA)) /* SDA Output=0 */
#define SDA_1() (SHT11_PxDIR &= ~BV(SHT11_ARCH_SDA)) /* SDA Input */
#define SDA_IS_1 (SHT11_PxIN & BV(SHT11_ARCH_SDA))

#define SCL_0() (SHT11_PxOUT &= ~BV(SHT11_ARCH_SCL)) /* SCL Output=0 */
#define SCL_1() (SHT11_PxOUT |= BV(SHT11_ARCH_SCL)) /* SCL Output=1 */
#endif

#define STATUS_REG_W 0x06 /* adr command r/w */
#define STATUS_REG_R 0x07 /* 000 0011 0 */
#define MEASURE_TEMP 0x03 /* 000 0001 1 */
#define MEASURE_HUMI 0x05 /* 000 0010 1 */
#define MEASURE_PRESS 0x09 /* 000 0100 1 ADDED BY TSELLOS*/
#define MEASURE_WEIGH 0x09 /* 000 1000 1 ADDED BY TSELLOS*/
#define RESET 0x1e /* 000 1111 0 */

/* This can probably be reduced to 250ns according to data sheet. */
#ifdef delay_400ns
#define delay_400ns() _NOP()
#endif
/*-----*/
```

Εικόνα 6.44: 1η επεξεργασία Sht11.c

```
/*-----*/
/*
 * Call may take up to 210ms.
 */
unsigned int
ht11_humidity(void)

    return scmd(MEASURE_HUMI);

/* ADDED TSELLOS
unsigned int
ht11_pressure(void)

    return scmd(MEASURE_PRESS);

/* ADDED TSELLOS
unsigned int
ht11_weight(void)

    return scmd(MEASURE_WEIGH);
/*-----*/
```

Εικόνα 6.45: 2η επεξεργασία Sht11.c

- Στο αρχείο Sht11.h:

```
unsigned int sht11_temp(void);
unsigned int sht11_humidity(void);
unsigned int sht11_pressure(void); /* added tsellos
unsigned int sht11_weight(void); /* added tsellos
unsigned int sht11_sreg(void);
int sht11_set_sreg(unsigned);
```

Εικόνα 6.46: 1η επεξεργασία Sht11.h

- Στο αρχείο Sht11-sensor.c:

```
/*-----*/
static int
value(int type)
{
    switch(type) {
        /* Photosynthetically Active Radiation. */
        case SHT11_SENSOR_TEMP:
            return sht11_temp();

        /* Total Solar Radiation. */
        case SHT11_SENSOR_HUMIDITY:|
            return sht11_humidity();
        case SHT11_SENSOR_PRESSURE: /* added tsellos
            return sht11_pressure();
        case SHT11_SENSOR_WEIGHT: /* added tsellos
            return sht11_weight();
        case SHT11_SENSOR_BATTERY_INDICATOR:
            return sht11_sreg() & 0x40? 1: 0;
    }

    return 0;
}
/*-----*/
```

Εικόνα 6.47: 1η επεξεργασία Sht11-sensor.c

- Στο αρχείο Sht11-sensor.h:

```
#include "lib/sensors.h"

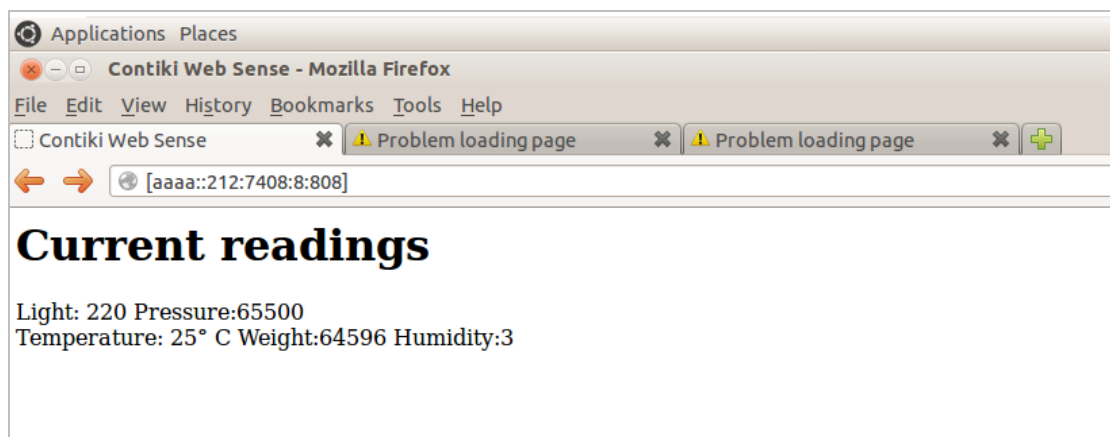
extern const struct sensors_sensor sht11_sensor;

#define SHT11_SENSOR_TEMP           0
#define SHT11_SENSOR_HUMIDITY      1
#define SHT11_SENSOR_BATTERY_INDICATOR 2
#define SHT11_SENSOR_PRESSURE      3
#define SHT11_SENSOR_WEIGHT        4
```

Εικόνα 6.48: 1η επεξεργασία Sht11-sensor.h

### 6.3.4 Εκτέλεση

Με τις απαραίτητες αλλαγές να έχουν ολοκληρωθεί, επαναλαμβάνονται όλα τα προηγούμενα βήματα, με το αντίστοιχο στήσιμο των motes και τις ρυθμίσεις που πρέπει να γίνουν και αφού γίνει σύνδεση σε ένα browser, όπως το Firefox θα παρουσιαστεί το αποτέλεσμα των αλλαγών. Το αποτέλεσμα αποτυπώνεται στην εικόνα 6.49.



Εικόνα 6.49: Τελικό αποτέλεσμα

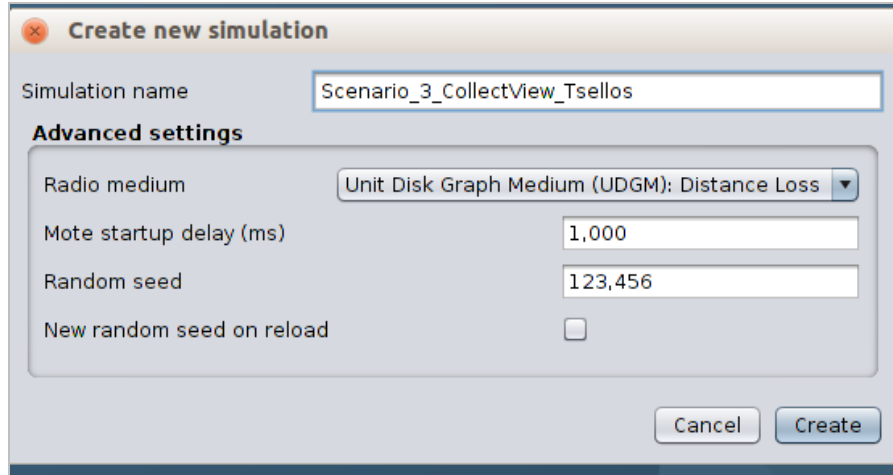
### 6.4 Σενάριο 3<sup>ο</sup>

Στο τελευταίο εκ των τριών σεναρίων το οποίο είναι και εκείνο στο οποίο δεν έχει γίνει καμία παρεμβολή σε κάποιο κείμενο του κώδικα. Στο σενάριο αυτό αξιοποιείται η λειτουργία “collect view” του Contiki προκειμένου να γίνει η συλλογή πληροφοριών για τα motes . Γίνεται επίσης μια ευρύτερη παρουσίαση δυνατοτήτων του προγράμματος που δεν έχουν αναδειχθεί προηγουμένως. Επιπροσθέτως μετά την παρουσίαση των δυνατοτήτων αυτών του προγράμματος, πραγματοποιείται και ένας πειραματισμός με την μεταφορά των σημείων (motes) στην κάτοψη AutoCad ενός γραφείου, με σκοπό να αναδειχθεί η ουσία στο να επικοινωνούν οι συσκευές μεταξύ τους χωρίς να χρειάζονται κάποια άλλου είδους συσκευή να παρεμβάλλεται.



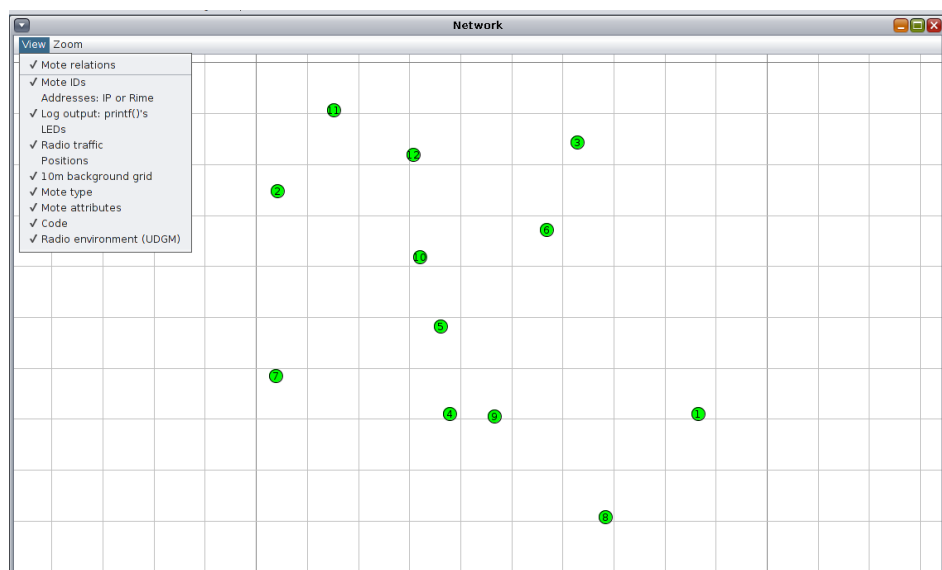
### 6.4.1 Εκτέλεση

Αρχικά, δημιουργείται ένα νέο Simulation, όπως φαίνεται στην εικόνα 6.50. Στη συνέχεια για το σκοπό αυτό, θα πρέπει να προστεθούν τα κατάλληλα motes, τα collect view.

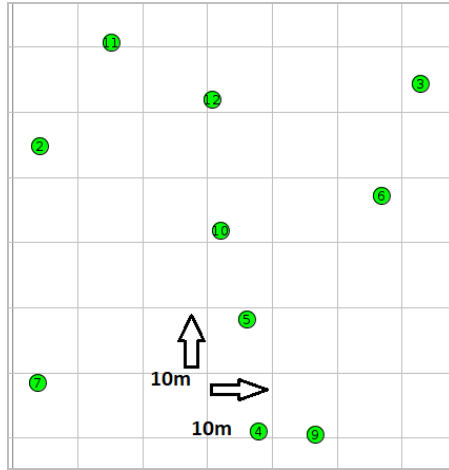


Εικόνα 6.50: Δημιουργία Simulation 3

Πατώντας στο View, δύναται να παρουσιαστούν και να χρησιμοποιηθούν διάφορες επιλογές, όπως αποτυπώνεται στην εικόνα 6.51. Με το Motes IDs εμφανίζεται η ονομασία – ο αριθμός του καθενός από αυτά. Με το Log output: printf()’s παρακολουθούνται τα μηνύματα που αποστέλλονται. Με το radio traffic, το οποίο και είναι ίσως το πιο “εντυπωσιακό” κομμάτι του Contiki παρατηρείται ζωντανά ο τρόπος επικοινωνίας μεταξύ των motes με κατευθυνόμενα βέλη που δείχνουν την πορεία των μηνυμάτων. Με το 10m background grid εμφανίζεται ένα πλέγμα το οποίο υποδεικνύει τις αποστάσεις μεταξύ των motes με το κάθε τετράγωνο να έχει πλευρές 10 μέτρα έκαστος, όπως φαίνεται στην εικόνα 6.52.

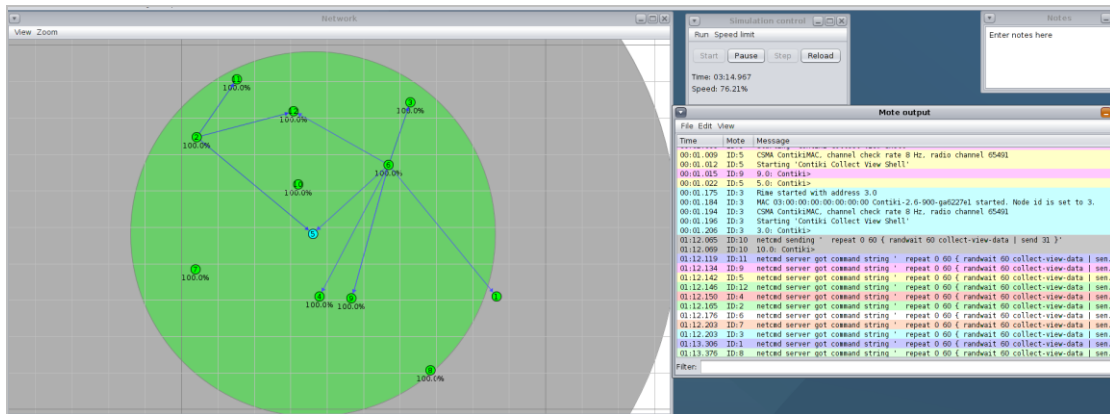


Εικόνα 6.51: Επιλογές View



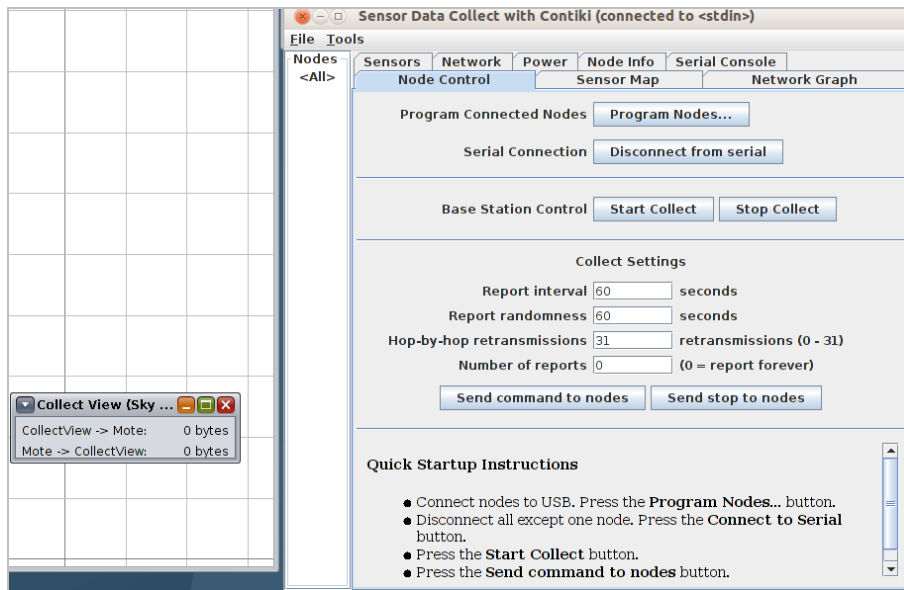
Εικόνα 6.52: 10m background grid

Το mote type είναι μια σημαντική επιλογή επίσης, καθώς στις περιπτώσεις που υπάρχουν motes διαφορετικού τύπου όπως στο Σενάριο 2, εκείνα που διαφέρουν και προσδιορίζονται διαφορετικά έχουν ξεχωριστό χρώμα από τα υπόλοιπα. Τέλος εκείνο που θα χρησιμοποιηθεί τελευταίο, είναι το Radio Environment (UDGM), το οποίο μέσω μια ακτίνας δείχνει το ενεργό πεδίο κάθε mote. Όποιο από τα υπόλοιπα motes βρίσκεται εντός αυτού του πεδίου, δύναται να επικοινωνήσει με αυτό, όπως παρουσιάζεται στην εικόνα 6.53. Σε επόμενο στάδιο θα γίνει αυξομείωση του πεδίου αυτού.



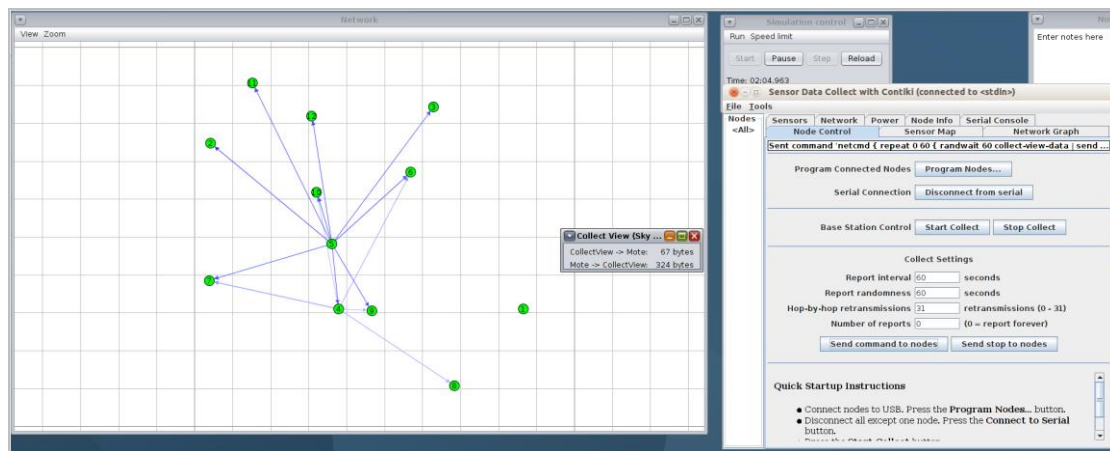
Εικόνα 6.53: Radio Environment (UDGM)

Στο collect vie μέσω του Mote tools for Sky, εμφανίζονται δύο παράθυρα. Όπως εμφανίζεται στην Εικόνα 6.54, στο μικρό παράθυρο collect view φαίνεται πως δεν υπάρχει καμία ανταλλαγή δεδομένων.



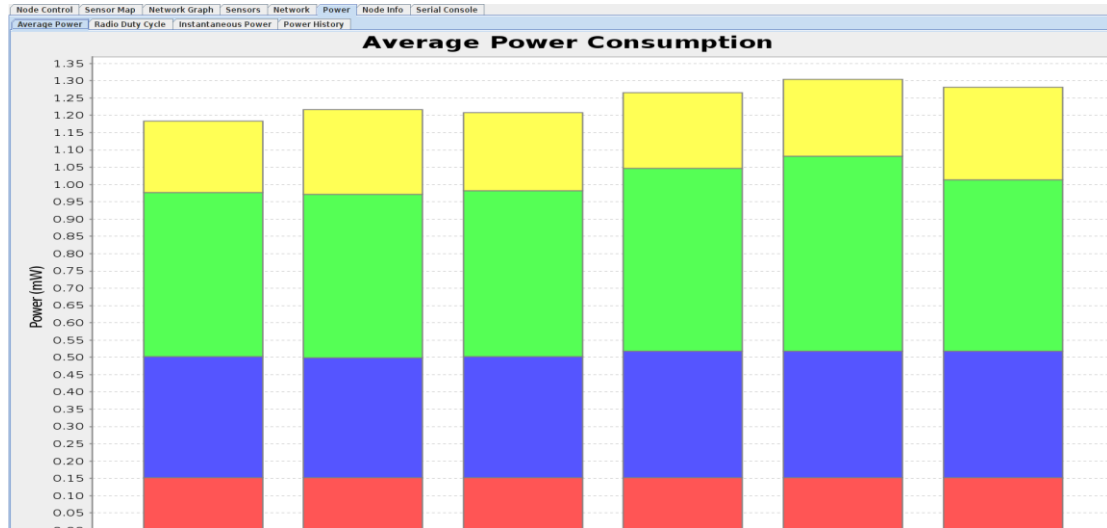
Εικόνα 6.54: Εμφάνιση παραθύρων collect view

Στην εικόνα 6.55, πατώντας run όπως και την επιλογή Send command to nodes παρατηρείται πως στο collect view αρχίζουν να εμφανίζονται τα bytes των δεδομένων.

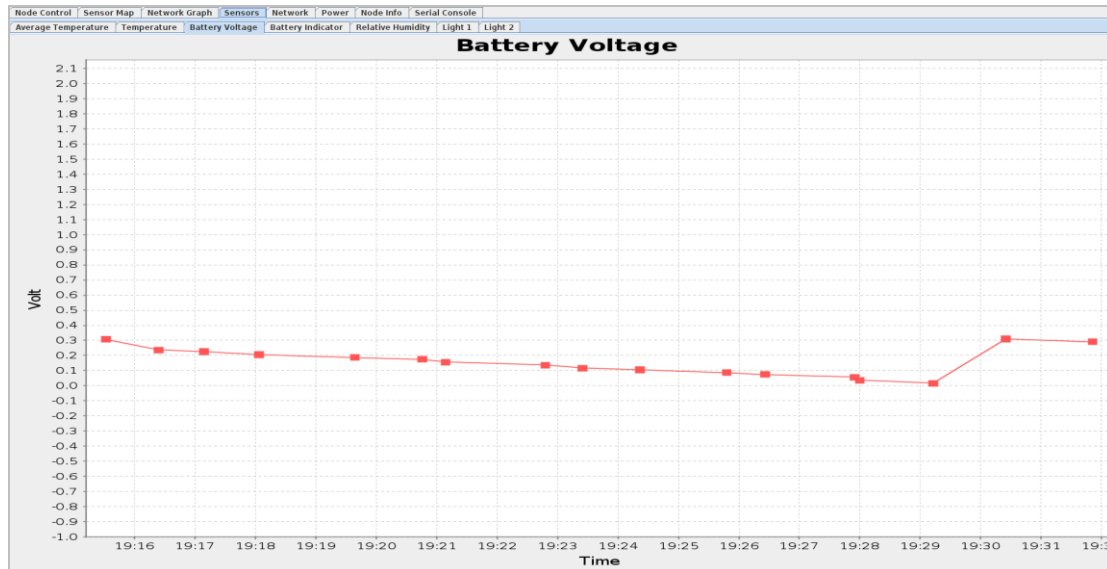


Εικόνα 6.55: Εμφάνιση bytes στο collect view.

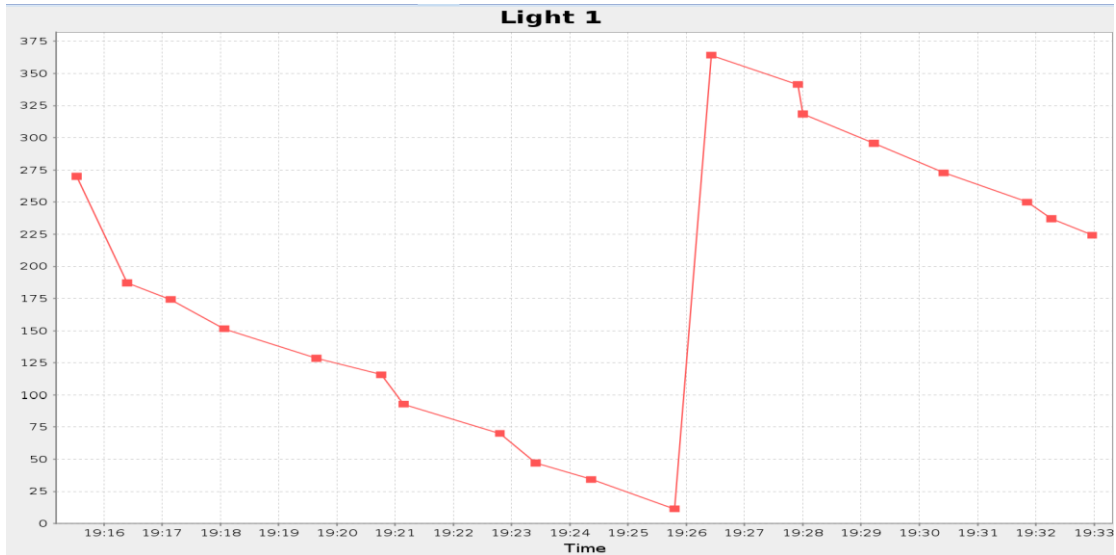
Παρακάτω στις εικόνες 6.56 - 6.61 που ακολουθούν, παρουσιάζονται ενδεικτικά κάποια από τα γραφήματα τα οποία μπορούν να ληφθούν.



Εικόνα 6.56: Average Power Consumption



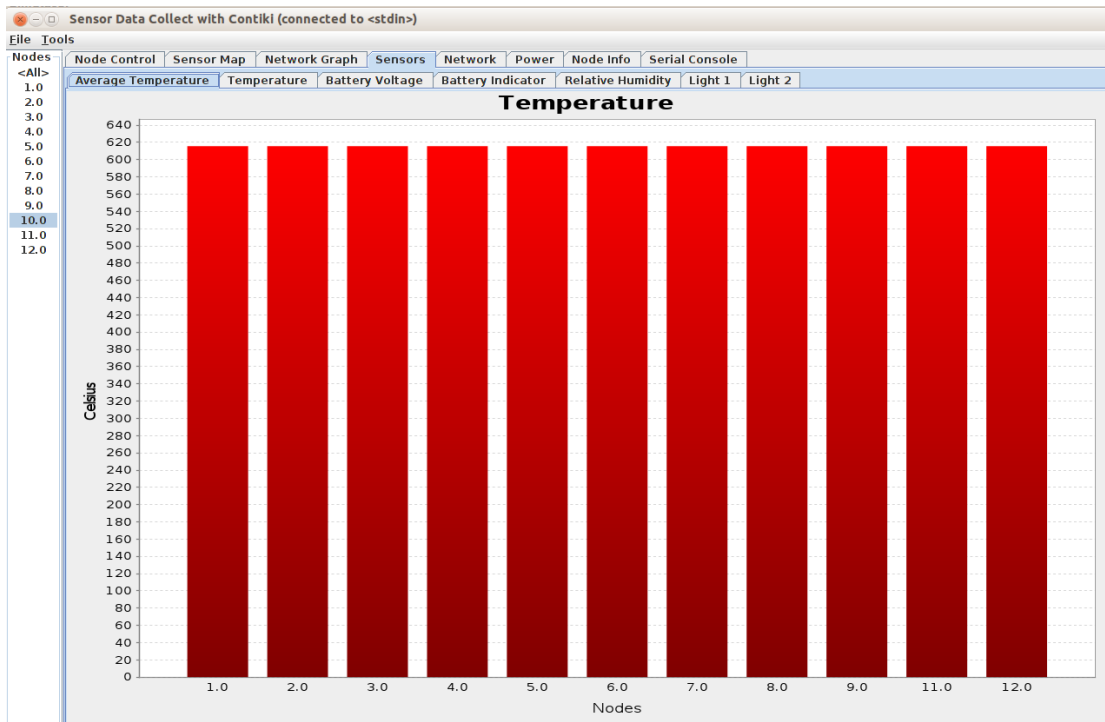
Εικόνα 6.57: Battery Voltage



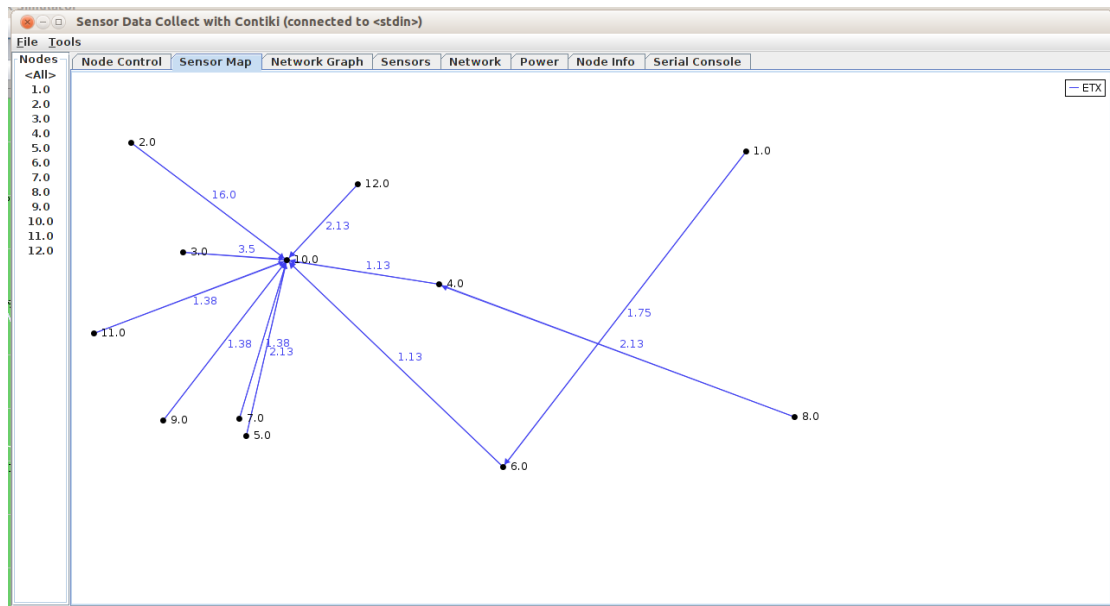
Εικόνα 6.58: Light

Nodes	Node Control	Sensor Map	Network Graph	Sensors	Network	Power	Node Info	Serial Console												
1.0	12	0	0	1.000	48.917	0.917	1	4 min, 45 sec	0	0.364	0.152	0.411	0.055	0.983	2 min...	0.685	0.104	0 min, 54 sec	0 min, 16 sec	1 min, 20 sec
2.0	12	0	0	1.000	8.000	1.000	0	4 min, 17 sec	0	0.357	0.153	0.418	0.050	0.977	2 min...	0.697	0.094	0 min, 54 sec	0 min, 23 sec	1 min, 39 sec
3.0	12	0	0	1.000	18.000	2.250	0	4 min, 32 sec	0	0.368	0.152	0.408	0.043	0.990	2 min...	0.710	0.082	0 min, 54 sec	0 min, 14 sec	1 min, 52 sec
4.0	13	0	0	1.000	9.308	1.163	0	4 min, 30 sec	0	0.365	0.152	0.408	0.051	0.976	2 min...	0.680	0.096	0 min, 52 sec	0 min, 23 sec	1 min, 45 sec
5.0	13	0	0	2.000	16.308	1.000	0	5 min, 00 sec	0	0.374	0.152	0.402	0.096	1.024	2 min...	0.670	0.180	0 min, 06 sec	0 min, 06 sec	1 min, 37 sec
6.0	13	0	0	1.000	17.231	2.154	0	4 min, 40 sec	0	0.360	0.153	0.402	0.056	0.971	2 min...	0.671	0.105	0 min, 56 sec	0 min, 31 sec	1 min, 36 sec
7.0	13	0	0	1.000	8.000	1.000	0	4 min, 43 sec	0	0.385	0.152	0.417	0.131	1.085	2 min...	0.696	0.246	0 min, 52 sec	0 min, 04 sec	1 min, 55 sec
8.0	13	0	0	1.000	8.000	1.000	0	4 min, 52 sec	0	0.365	0.152	0.418	0.056	0.992	2 min...	0.697	0.106	0 min, 53 sec	0 min, 06 sec	1 min, 30 sec
9.0	12	0	0	1.000	18.000	2.250	0	4 min, 12 sec	0	0.353	0.153	0.434	0.035	0.975	2 min...	0.723	0.066	0 min, 53 sec	0 min, 25 sec	1 min, 32 sec
10.0	13	0	0	1.000	17.231	2.154	0	4 min, 32 sec	0	0.366	0.152	0.414	0.094	1.026	2 min...	0.689	0.177	0 min, 53 sec	0 min, 19 sec	1 min, 47 sec
11.0	12	0	0	1.000	50.167	0.948	1	4 min, 16 sec	0	0.365	0.152	0.447	0.070	1.034	2 min...	0.744	0.131	0 min, 54 sec	0 min, 26 sec	1 min, 24 sec
12.0	0	0	0	0.000	0.000	0.000	0		0	0.000	0.000	0.000	0.000	0.000		0.000	0.000			
Avg	12.545	0.000	0.000	1.091	20.015	1.440	0.182	4 min, 34 sec	0.000	0.366	0.152	0.418	0.067	1.003	2 min...	0.697	0.126	0 min, 53 sec	0 min, 16 sec	1 min, 37 sec

Εικόνα 6.59: Node info

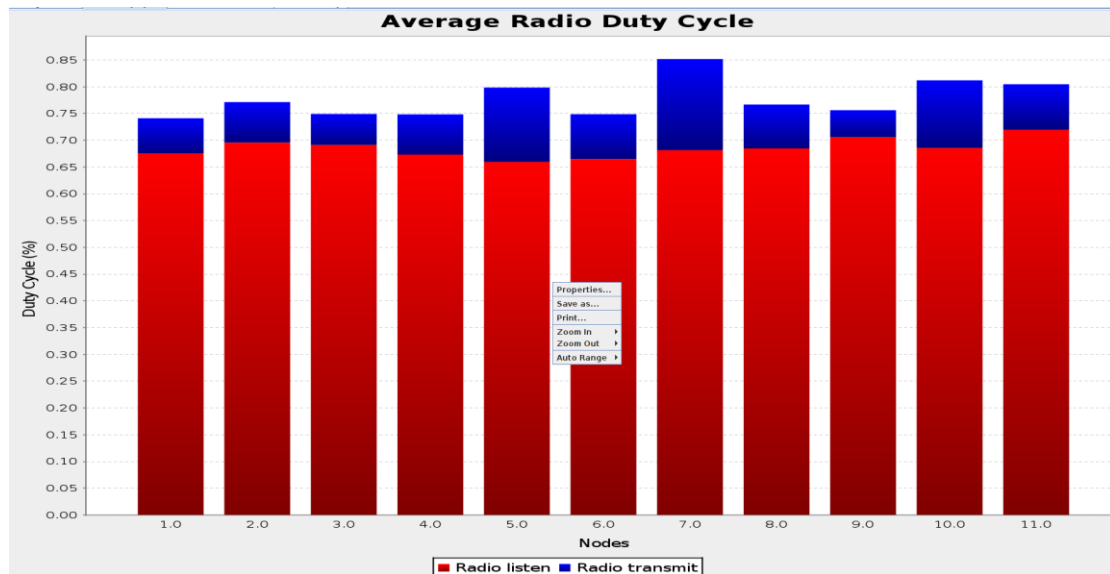


Εικόνα 6.60: Temperature



Εικόνα 6.61:Sensor Map

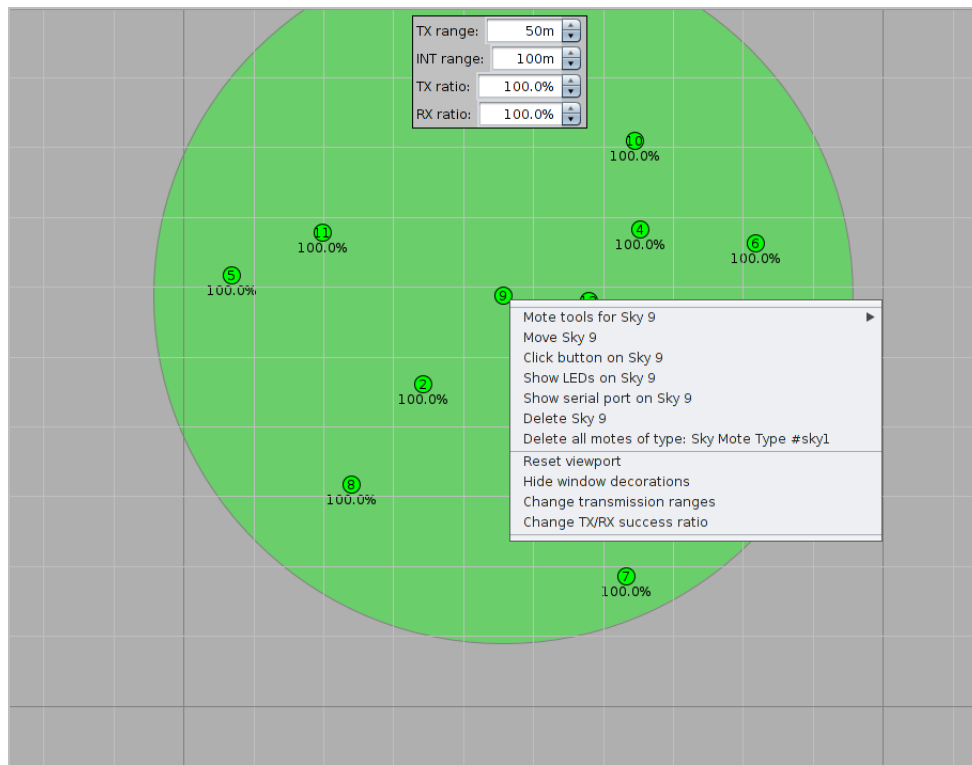
Εδώ θα πρέπει να σημειωθεί πως για τα γραφήματα αυτά υπάρχει η δυνατότητα της επεξεργασίας όπως και να της αποθήκευσης όπως φαίνεται και στην Εικόνα 6.62.



Εικόνα 6.62:Average Radio Duty Cycle – Properties Γραφημάτων

#### 6.4.1 Μια εναλλακτική ματιά

Στο σημείο αυτό θεωρήθηκε αναγκαίο να γίνει μια επεξεργασία της ακτίνας του πεδίου μετάδοσης των nodes. Πηγαίνοντας στην επιλογή Change Transmission ranges, μπορεί να γίνει επεξεργασία το προαναφερθέν πεδίο το οποίο είναι προκαθορισμένο στα 50m, όπως φαίνεται στην εικόνα 6.63.



**Εικόνα 6.63: Καθορισμός Transmission ranges**

Τέλος έγινε η προσπάθεια να μεταφερθεί το περιβάλλον αυτό σε μια κάτοψη AutoCad ούτως ώστε να γίνει παρατήρηση της απόστασης του 10m background grid πάνω σε μια ήδη μετρημένη κάτοψη. Στην εικόνα 6.64, ελήφθη επομένως η παρακάτω κάτοψη.



**Εικόνα 6.64: Κάτοψη Γραφείου**

Στην επόμενη εικόνα 6.65 φαίνεται η κάλυψη που θα είχε ένα mote με ρυθμισμένο το range στα 15m μέτρα.



Εικόνα 6.65: Κάτοψη Γραφείου – Πεδίο Radio Environment (UDGM)

## Συμπεράσματα

Η παρούσα διπλωματική εργασία αποτέλεσε μια ματιά μέσα στον τεράστιο κόσμο που ονομάζεται Internet of Things. Στον κόσμο αυτό η περιήγηση πραγματοποιήθηκε με γνώμονα τη μελέτη των WNS δικτύων και ειδικότερα με βάση το πρωτόκολλο 6LoWPAN. Στο μελέτη αυτή συναντήθηκαν πρωτόκολλα όπως το IEEE 802.15.4 και το Zigbee. Αυτές τα δεδομένα συγκροτούν ένα οδικό χάρτη επιστημονικής περιήγησης είναι μόνο αδιάφορα δεν είναι με την εξέλιξη του πυρήνα της θεματολογίας της εργασίας αυτής.

Μέσα λοιπόν από τη μελέτη των παραπάνω εννοιών εμφανίστηκε στο δρόμο της έρευνας το πρόγραμμα Contiki COOJA. Το Contiki το οποίο αποτέλεσε σημαντικό εργαλείο, προκειμένου να εμφανιστεί στην πράξη η εφαρμογή των προτύπων πάνω στα οποία ξεκίνησε η μελέτη. Μέσα από τις τεράστιες δυνατότητες που διαθέτει, προσφέρει μια ξεχωριστή προσέγγιση στα WNS δίκτυα. Χρησιμοποιώντας τις απεριόριστες επιλογές του οι ιδέες μετατράπηκαν σε προσομοιώσεις και εκείνες με τη σειρά τους σε έναν καμβά, πάνω στον οποίο αναπτύχθηκαν οι διάφορων ειδών επικοινωνίες μεταξύ των συσκευών.

Σαν συμπέρασμα μένει σίγουρα η ποικιλία και η πολυπλοκότητα που είχε η ενασχόληση με τα παραπάνω πρωτόκολλα όπως και με τη συγκεκριμένη εφαρμογή. Μπορεί να ειπωθεί πως κυρίως μέσω του 2<sup>ου</sup> σεναρίου, τίθεται ένα λιθαράκι για την επίλυση ζητημάτων που μπορεί να κληθεί να αντιμετωπίσει ο μελλοντικός αναγνώστης και ανάβει η λυχνία της αναζήτησης αναφορικά με την περαιτέρω ενασχόληση με το αντικείμενο.



Κατά τη πορεία της διπλωματικής αυτής έρευνας αντιμετωπίστηκε ένα πλήθος ζητημάτων. Το κυριότερο όλων ήταν το βασικό θέμα της εργασίας, το οποίο κατά τη διάρκειας της έρευνας άλλαξε άρδην, με αποτέλεσμα την ανάπτυξη του παρόντος θέματος. Η ελάχιστη παροχή επαρκούς βιβλιογραφίας αναφορικά με το πρακτικό κομμάτι της εφαρμογής σχετικά με την υλοποίηση των σεναρίων είχε σαν αποτέλεσμα να αφιερωθεί ένα μεγάλο χρονικό διάστημα στην υλοποίησή τους. Η έλλειψη φακέλων από την νέα έκδοση και κατά συνέπεια απαραίτητων αρχείων από την εφαρμογή, προέτρεψε στον πειραματισμό και σε αλχημείες ανταλλαγής αρχείων μεταξύ διαφόρων εκδόσεων του Contiki, οι οποίες εν τέλει έφεραν θετικό αποτέλεσμα. Η ιδέα επίσης για αποτύπωση του περιβάλλοντος της εφαρμογής σε μια κάτοψη AutoCad, παρουσίασε ένα βαθμό δυσκολίας, όσον αφορά τη διαστασιολόγηση των δύο εφαρμογών. Ο λόγος υλοποίησης της ενέργειας αυτής ήταν για να γίνει μια αποτύπωση των αποστάσεων των motes, πάνω σε μια κάτοψη γραφείου και να υπάρχει κατά αυτόν τον τρόπο μια πιο πρακτική εικόνα της μετάδοσης των δεδομένων μεταξύ των συσκευών. Συμπερασματικά, όλες οι ιδέες που δημιουργήθηκαν, με τον έναν ή τον άλλο τρόπο, η εφαρμογή κατάφερε να τις υποστηρίξει και έτσι στο τέλος ελήφθη και το κόκκινο LED και οι μετρήσεις των μεγεθών, αλλά και τα δεδομένα των motes πάνω στα γραφήματα.

Ελπίζω στο άμεσο μέλλον η εργασία αυτή να αποτελέσει το έναυσμα και τον οδηγό για μελλοντικές επιστημονικές έρευνες και να αποτελέσει τον αρωγό αυτό που θα βοηθήσει τις ιδέες που θα έρθουν να γίνουν πραγματικότητα.

## **Βιβλιογραφία**

[1] Γιαννέλος Σ. (2022). Η εξέλιξη της τεχνολογίας των Μικροψηφίδων. (Διπλωματική Εργασία, Πανεπιστήμιο Δυτικής Αττικής, Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών, 2022). Ανακτήθηκε: 2023

[2] Διονυσόπουλος Γ. (2022). Ασύρματες Τεχνολογίες σε Εφαρμογές IoT. (Μεταπτυχιακή διπλωματική εργασία, Οικονομικό Πανεπιστήμιο Αθηνών, 2022). Ανακτήθηκε 2023, από : [https://mm.aueb.gr/master\\_theses/xylomenos/2022-dionisopoulos.pdf](https://mm.aueb.gr/master_theses/xylomenos/2022-dionisopoulos.pdf)

[3] Τσερεγκούνης Γ. (2021). Ασφαλής διαχείριση IoT συσκευών. (Διπλωματική εργασία, Πανεπιστήμιο Δυτικής Αττικής, 2021). Ανακτήθηκε 2023, από: [https://polynoe.lib.uniwa.gr/xmlui/bitstream/handle/11400/1156/ice\\_cs151056.pdf?sequence=1&isAllowed=y](https://polynoe.lib.uniwa.gr/xmlui/bitstream/handle/11400/1156/ice_cs151056.pdf?sequence=1&isAllowed=y)

[4] Σιμόπουλος Σ. (2022). Εφαρμογές του διαδικτύου των πραγμάτων σε κρίσιμα επικοινωνιακά συστήματα. (Διπλωματική εργασία, Πανεπιστήμιο Δυτικής Αττικής, 2022). Ανακτήθηκε 2023, από :

[https://polynoe.lib.uniwa.gr/xmlui/bitstream/handle/11400/2842/Simopoulos\\_80697720.pdf?sequence=1&isAllowed=y](https://polynoe.lib.uniwa.gr/xmlui/bitstream/handle/11400/2842/Simopoulos_80697720.pdf?sequence=1&isAllowed=y)

[5] Αλχαζίδης Β. (2019). Μελέτη χρήσης νευρωνικών δικτύων για την επεξεργασία χωροχρονικών δεδομένων από δίκτυα αισθητήρων. (Διπλωματική εργασία, Πανεπιστήμιο Αιγαίου, 2019). Ανακτήθηκε 2023, από : <https://hellenicus.lib.aegean.gr/handle/11610/20441>

[6] Τζιαμπάζη Δ. (2016). Υλοποίηση του φυσικού επιπέδου πρωτοκόλλου 802.15.4 στα 2.4GHz. (Διπλωματική εργασία, Πανεπιστήμιο Πατρών, 2016). Ανακτήθηκε 2023, από : <https://nemertes.library.upatras.gr/items/314d2f8b-e321-4cfc-8e03-49a8caa918b8>

[7] Φωκαέως Μ. (2020). Σχεδιασμός και ανάπτυξη χαμηλού κόστους αισθητήρα υγρασίας φυλλώματος και προσαρμογή αυτού σε πλατφόρμα IoT. (Διπλωματική εργασία, Πανεπιστήμιο Πατρών, 2020). Ανακτήθηκε 2023, από : <https://nemertes.library.upatras.gr/items/b9bce429-c218-4a48-b75d-bea4c3dce902>

[8] Ζουμή Ε. (2019). Καταγραφή, ταξινόμηση και διαχείριση ιατρικών δεδομένων με χρήση πλατφόρμας IoT. (Μεταπτυχιακή διατριβή, Πανεπιστήμιο Πειραιώς, 2019). Ανακτήθηκε 2023, από : <https://dione.lib.unipi.gr/xmlui/handle/unipi/12530>

[9] Βασιλείου Σ. Π. (2019). Έξυπνα σπίτια και οδηγός εγκατάστασης της πλατφόρμας ανοιχτού κώδικα Home Assistant. (Διπλωματική εργασία, Πανεπιστήμιο Θεσσαλίας, 2019). Ανακτήθηκε 2023, από :

<https://ir.lib.uth.gr/xmlui/handle/11615/50538;jsessionid=14D00D9217E51C8977BD DA905A000330>

[10] Σεφινά Α. (2016). IEEE 802.15.4 Ανάλυση MAC και Physical Layer Επιπέδου, Απώλειες Δεδομένων Λόγω Περιβάλλοντος και Παρεμβολών Θορύβου Άλλων Τεχνολογιών. (Διπλωματική εργασία, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης, 2017). Ανακτήθηκε 2023, από : <http://ikee.lib.auth.gr/record/291986?ln=el>

[11] Βετουλαδίτης Μ. (2006). Υλοποίηση MAC πρωτοκόλλου 802.15.4 για ασύρματα δίκτυα αισθητήρων. (Μεταπτυχιακή διπλωματική εργασία, Πανεπιστήμιο Πειραιώς, 2007). Ανακτήθηκε 2023, από : <https://dione.lib.unipi.gr/xmlui/handle/unipi/1350>

[12] Vasilios A. Orfanos - Stavros D. Kaminaris - Panagiotis Papageorgas - Dimitrios Piromalis - Dionisis Kandris (2023). A Comprehensive Review of IoT Networking Technologies for Smart Home Automation Applications. ( Article, Department of Electrical and Electronic Engineering, University of West Attica). Ανακτήθηκε: 2023

[13] Vasilios A. Orfanos - Stavros D. Kaminaris - Panagiotis Papageorgas - Dimitrios Piromalis (2019). Trends in home automation systems and protocols. ( Article,

Department of Electrical and Electronic Engineering, University of West Attica).  
Ανακτήθηκε: 2023

[14] Javapoint.com. Προσπελάστηκε :2023 από <https://www.javatpoint.com/multiple-access-protocols>

[15] Thread Group (2020). Thread NetWork Fundamentals. Ανακτήθηκε: 2023

[16] Παρασκευοπούλου Έ. (2020). Διαδίκτυο των πραγμάτων στον κλάδο των επιχειρήσεων. Ανάλυση και επεξεργασία δεδομένων σε επιχειρησιακά δίκτυα και οικοσυστήματα που βασίζονται σε επιχειρησιακά μοντέλα. (Μεταπτυχιακή διπλωματική εργασία, Πανεπιστήμιο Μακεδονίας, 2020). Ανακτήθηκε 2023, από : <https://dspace.lib.uom.gr/handle/2159/24696>

[17] Σιμόπουλος Ε. Ν. (2013). Forest fire detection with wireless sensor networks. (Διπλωματική εργασία, Εθνικό Μετσόβιο Πολυτεχνείο, 2014). Ανακτήθηκε 2023, από : <https://dspace.lib.ntua.gr/xmlui/handle/123456789/38214?show=full>

[18] Σιηττή Χ. (2013). Ασύρματα δίκτυα αισθητήρων με εφαρμογές στον προσομοιωτή Contiki Cooja. (Διπλωματική εργασία, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης, 2013). Ανακτήθηκε 2023, από :

<https://ikee.lib.auth.gr/record/291570?ln=el>

[19] Παπαγεωργίου Α. (2020). Σχεδίαση και υλοποίηση συστήματος ασφαλούς απομακρυσμένου προγραμματισμού ασύρματων αισθητήρων. (Μεταπτυχιακή διπλωματική εργασία, Ελληνικό Ανοικτό Πανεπιστήμιο, 2020). Ανακτήθηκε 2023, από : <https://apothesis.eap.gr/archive/item/78138>

[20] Pixabay (6 Ιουνίου 2023). Internet of Things. Ανακτήθηκε 2024, από : <https://pixabay.com/illustrations/computer-technology-connected-8037838/>

[21] Raja Abdelmoumen (2019). A Review of Link Layer Protocols for Internet of Things. (Article, Higher Institute of Technological Studies of Sousse). Ανακτήθηκε: 2023

[22] Anton Alm (2019). Internet of Things mesh network - Using the Thread networking protocol. ( Article, Faculty of Health, Science and Technology). Ανακτήθηκε: 2023

[23] Πλατοκούκη Γ. – Γκολφινόπουλου Ξ. (2015). Internet of Things & Zigbee Network Απομακρυσμένος έλεγχος θερμοκηπίου. (Διπλωματική εργασία, Εθνικό Μετσόβιο Πολυτεχνείο, 2015). Ανακτήθηκε 2023, από : <https://dspace.lib.ntua.gr/xmlui/bitstream/handle/123456789/40916/%CE%B4%CE%B9%CF%80%CE%BB%CF%89%CE%BC%CE%B1%CF%84%CE%B9%CE%BA%CE%AE%20IIOT%20final.pdf?sequence=1&isAllowed=y>

[24] Amy J.C. Trappey - Charles V. Trappey - Usharani Hareesh Govindarajan - Allen C. Chuang - John J. Sun (2017). A review of essential standards and patent landscapes for the Internet of Things: A key enabler for Industry 4.0. ( Article , Department of Industrial Engineering and Engineering Management, National Tsing Hua University, Hsinchu, Taiwan - Department of Management Science, National Chiao Tung University, Hsinchu, Taiwan). Ανακτήθηκε : 2023

[25] Θεανώ Β. Μαλούτα - Ελένη Ζ. Μίχου (2019). Το Διαδίκτυο των Πραγμάτων (IoT). Συγκριτική παρουσίαση και αξιολόγηση των σημαντικότερων τεχνολογιών υλοποίησης. (Διπλωματική Εργασία, Σχολή Θετικών Επιστημών, Τμήμα Πληροφορικής και Τηλεπικοινωνιών , Εθνικό Και Καποδιστριακό Πανεπιστήμιο Αθηνών). Ανακτήθηκε: 2023

[26] Stefan Marksteiner, Víctor Juan Exposito Jiméneez, Heribert Vallant, Herwig Zeiner (2017). An Overview of Wireless IoT Protocol Security in the Smart Home Domain. (Article, DIGITA - Institute for Information and Communication Technologies - Graz, Austria). Ανακτήθηκε: 2023

[27] Shahid Razaa - Prasant Misra - Zhitao Hea - Thiemo Voigt (2017). Building the Internet of Things with bluetooth smart. (Article, SICS Swedish ICT, Stockholm, Sweden, TATA Consultancy Services Ltd., Bangalore, India Department of Information Technology, Uppsala University, Sweden). Ανακτήθηκε: 2023

[28] Suk Kyu Lee - Mungyu Bae - Hwangnam Kim (2017). Future of IoT Networks: A Survey. (Article, LG Electronics Inc., Seoul, School of Electrical Engineering, Korea University). Ανακτήθηκε: 2023

[29] Guilherme Mussi Toschi - Leonardo Barreto - Campos, Carlos Eduardo Cugnasca (2017). Home Automation Networks: A Survey. (Article, Department of Computer Engineering, University of Sao Paulo, SP, Brazil). Ανακτήθηκε: 2023

[30] Φώτιος Σολδάτος (2017). <Internet of Things>. (Εργασία Εξαμήνου, Πολυτεχνική Σχολή – Πανεπιστήμιο Πατρών). Ανακτήθηκε: 2023

[31] Shadi Al-Sarawi - Mohammed Anbar - Kamal Alieyan - Mahmood Alzubaidi (2017). Internet of Things (IoT) Communication Protocols: Review. (Article, 4National Advanced IPv6 Centre (NAV6), Universiti Sains Malaysia). Ανακτήθηκε: 2023

[32] Gianfranco Cerullo - Giovanni Mazzeo - Gaetano Papale - Bruno Ragucci - Luigi Sgaglione (2017). IoT AND SENSOR NETWORKS SECURITY. (Article, University of Naples “Parthenope”, Naples, Italy). Ανακτήθηκε: 2023

[33] Terence K.L. Hui a - R. Simon Sherratt - Daniel Díaz Sánchez (2017). Major requirements for building Smart Homes in Smart Cities based on Internet of Things technologies. ( Article, School of Systems Engineering, University of Reading, UK, Dep. Ing. Telemática, Universidad Carlos III de Madrid, Spain). Ανακτήθηκε: 2023

[34] Matteo Cerchecci - Francesco Luti - Alessandro Mecocci - Stefano Parrino - Giacomo Peruzzi - Alessandro Pozzebon (2018). A Low Power IoT Sensor Node Architecture for Waste Management Within Smart Cities Context. ( Article , Department of Information Engineering and Mathematical Sciences, University of Siena). Ανακτήθηκε: 2023

[35] Arbia Riahi Sfar - Enrico Natalizio - Yacine Challal - Zied Chtourou (2018). A roadmap for security challenges in the Internet of Things. ( Article , VRIT Lab - Military Academy, Nabeul, Tunisia, Sorbonne Universites, Universite de technologie de Compiegne, CNRS, Laboratoire de Methodes de Conception de Systemes (LMCS), Ecole Nationale Superieure d’Informatique (ESI), Centre de Recherche sur l’Information Scientifique et Technique (CERIST), Algiers). Ανακτήθηκε: 2023

[36] Bogdan-Cosmin Chifor - Ion Bica - Victor-Valeriu Patriciu - Florin Pop (2018). A security authorization scheme for smart home Internet of Things devices. ( Article , Military Technical Academy, University Politehnica of Bucharest, Computer Science Department, Splaiul Independentei, National Institute for Research and Development in Informatics (ICI)). Ανακτήθηκε: 2023

[37] P.P. Ray (2018). A survey on Internet of Things architectures. ( Article , Department of Computer Applications, Sikkim University). Ανακτήθηκε: 2023

[38] Iván Froiz-Míguez -Tiago M. Fernández-Caramés - Paula Fraga-Lamas - Luis Castedo (2018). Design, Implementation and Practical Evaluation of an IoT Home Automation System for Fog Computing Applications Based on MQTT and ZigBee-WiFi Sensor Nodes. (Article, Department of Computer Engineering, Faculty of Computer Science, Universidade da Coruña). Ανακτήθηκε: 2023

[39] Duarte Raposo - André Rodrigues - Soraya Sinche - Jorge Sá Silva - Fernando Boavida (2018). Industrial IoT Monitoring: Technologies and Architecture Proposal. (Article, Centre of Informatics and Systems of the University of Coimbra, Polytechnic Institute of Coimbra, ISCAC, DETRI, Escuela Politécnica Nacional). Ανακτήθηκε: 2023

[40] Wael Ayoub - Abed Samhat - Fabienne Nouvel - Mohamad Mroue - Jean-Christophe Prévotet (2018). Internet of Mobile Things: Overview of LoRaWAN, DASH7, and NB-IoT in LPWANs standards and Supported Mobility. (Article, Institut National des Sciences Appliquees de Rennes, Faculty of Engineering - CRSI, Lebanese University, Hadath Campus). Ανακτήθηκε: 2023

[41] Abhishek D. Pathaka - Jitendra V. Tembhurnea (2018). Internet of Things: A Survey on IoT Protocols. ( Article, St. Vincent Pallotti College of Engineering & Technology). Ανακτήθηκε: 2023

[42] Ishaq Unwala - Zafar Taqvi - Jiang Lu (2018). IoT Security: ZWave and Thread. (Article, Computer Engineering University of Houston Clear Lake). Ανακτήθηκε: 2023

- [43] Keith Nolan - Mark Kelly (2018). IPv6 Convergence for IoT Cyber-Physical Systems. ( Article, Intel Research and Development Ireland Limited). Ανάκτηση: 2023
- [44] Anshuman Dash - Satyajit Pal - Chinmay Hegde (2018). Ransomware Auto-Detection in IoT Devices using Machine Learning. ( Article, Academy for Corporate Excellence, REVA University). Ανακτήθηκε: 2023
- [45] Ishaq Unwala - Zafar Taqvi - Jiang Lu (2018). Thread: An IoT Protocol. (Article, Computer Engineering University of Houston Clear Lake). Ανακτήθηκε 2023
- [46] Moteiv Corporation (3/5/2005). Tmote sky : PRELIMINARY Datasheet. Τεχνικό Φυλλάδιο Tmote Sky. Ανακτήθηκε 2023, από : <http://www.moteiv.com>
- [47] Tara Salman - Raj Jain (2019). A Survey of Protocols and Standards for Internet of Things. (Article, Department of Computer Science and Engineering Washington University). Ανακτήθηκε: 2023
- [48] Kishor Krishnan Nair - Adnan M. Abu-Mahfouz - Samuel Lefophane (2019). Analysis of the Narrow Band Internet of Things (NB-IoT) Technology. (Article, Department of Electrical Engineering, Tshwane University of Technology). Ανακτήθηκε: 2023
- [49] Prachin Bhojar - Parul Sahare - S. B.Dhok - R. B. Deshmukh (2019). Communication Technologies and Security Challenges for Internet of Things: A Comprehensive Review. (Article, Visvesvaraya National Institute of Technology, Indian Institute of Information Technology). Ανακτήθηκε: 2023
- [50] Mardiana binti Mohamad Noor - Wan Haslina Hassan (2019). Current research on Internet of Things (IoT) security: A survey. (Article, Malaysia-Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia). Ανακτήθηκε: 2023
- [51] ZOLERTIA TM (11/3/2010). Z1 Datasheet. Τεχνικό Φυλλάδιο Z1 mote. Ανακτήθηκε 2023, από : [www.zolertia.com](http://www.zolertia.com)
- [52] Daniel Minoli (2019). Positioning of Blockchain Mechanisms in IoT-powered Smart Home Systems: A Gateway-based Approach. ( Article, DVI Communications). Ανακτήθηκε: 2023
- [53] Lilia Tightiz - Hyosik Yang (2020). A Comprehensive Review on IoT Protocols' Features in Smart Grid Communication. ( Article, Department of Computer Engineering, Sejong University). Ανακτήθηκε: 2023
- [54] Georgios Kambourakis - Constantinos Koliass - Dimitrios Geneiatakis - Georgios Karopoulos - Georgios Michail Makrakis - Ioannis Kounelis ( 2020). A State-of-the-Art Review on the Security of Mainstream IoT Wireless PAN Protocol Stacks. ( Article, European Commission, Joint Research Centre (JRC), Faculty of Computer Science, College of Engineering, University of Idaho). Ανακτήθηκε: 2023

[55] B.B. Gupta - Megha Quamara (2020). An overview of Internet of Things (IoT): Architectural aspects, challenges, and protocols. ( Article, National Institute of Technology Kurukshetra, Kurukshetra). Ανακτήθηκε: 2023

[56] Guillermo del Campo - Igor Gomez - Guillermo Cañada - Luca Piovano, Asuncion Santamaria (2020). ( Article, Guidelines and criteria for selecting the optimal low-power wide-area network technology). Ανακτήθηκε: 2023

[57] V.A. Orfanos - S.D. Kaminaris - D. Piromalis - P. Papageorgas (2020). Smart Home Automation in the IoT Era: A Communication Technologies Review. ( Article, Department of Electrical and Electronics Engineering, University of West Attica, Department of Industrial Design & Production, University of West Attica). Ανακτήθηκε: 2023

[58] Κατσιβέλης Ν. (2016). Διερεύνηση των τεχνολογιών και των εφαρμογών Smart Home στο χώρο της υγείας. (Διπλωματική εργασία, Εθνικό Μετσόβιο Πολυτεχνείο, 2016). Ανακτήθηκε 2023, από :

<http://artemis.cslab.ece.ntua.gr:8080/jspui/handle/123456789/13297>

[59] Φατσή Γ. (2020). Ευφυή προγραμματιζόμενα δίκτυα (software defined networks) στο διαδίκτυο των πραγμάτων (internet of things). (Μεταπτυχιακή διπλωματική εργασία, Πανεπιστήμιο Μακεδονίας, 2020). Ανακτήθηκε 2023, από : <https://dspace.lib.uom.gr/handle/2159/24572>

[60] Καλορίτης Νικόλαος (2022). Μελέτη των Προτύπων Επικοινωνίας για την Υποστήριξη Εφαρμογών της Βιομηχανίας 4.0. (Διπλωματική Εργασία, Σχολή Θετικών Επιστημών και Τεχνολογίας, Ελληνικό Ανοικτό Πανεπιστήμιο). Ανακτήθηκε: 2023

[61] Biljana L. Risteska Stojkoska - Kire V. Trivodaliev (2017). A review of Internet of Things for smart home: Challenges and solutions. ( Article , Faculty of Computer Science and Engineering, University “Ss. Cyril and Methodius”, Skopje, Macedonia) Ανακτήθηκε : 2023

[62] Stefano Tennina. Anis Koubâa, Roberta Daidone, Mário Alves, Petr Jurčík, Ricardo Severino, Marco Tiloca, Jan-Hinrich Hauer, Nuno Pereir (2013). IEEE802.15.4 and ZigBee as Enabling Technologies for Low-Power Wireless Systems with Quality-of-Service Constraints. (Springer). Ανακτήθηκε: 2023

[63] M. Chere, T. Ngqondi and M. Bembe (2019). Wireless Low Power Area Networks in the Internet of Things: A Glimpse on 6LoWPAN. (International Conference on Electronics, Information, and Communication (ICEIC), Auckland, New Zealand). Ανακτήθηκε: 2023

[64] Paolo Baronti, Prashant Pillai, Vince W.C. Chook, Stefano Chessa ,AlbertoGotta, Y. Fun Hu. (2006). Wireless sensor networks: A survey on thestate of the art and the

802.15.4 and ZigBee standards. (University of Bradford, Istituto di Scienza e Technologie dell'Informazione, University of Piza). Ανακτήθηκε: 2023

[65] Dieter Uckelmann, Mark Harrison, Florian Michahelles (2011). Architecting theInternet of Things. (Springer). Ανακτήθηκε: 2023

[66] SourceForge. Προσπελάστηκε: 2023 από

<https://sourceforge.net/projects/contiki/>[67] GitHub. Προσπελάστηκε: 2023 από

<https://github.com/>

[68] Γούτσου Κ. (2014). Υλοποίηση τεχνικών για την αποφυγή συμφόρησης σε τοπικά ασύρματα δίκτυα αισθητήρων. (Διπλωματική εργασία, Πανεπιστήμιο Πατρών, 2014). Ανακτήθηκε 2023, από : <https://nemertes.library.upatras.gr/items/23f53a51-6480-403d-9b13-5f9b9cc6e9bb>

[69] Κουρουλέας Γ. (2014). Αυστηρή σχεδίαση εφαρμογών στο διαδίκτυο των αντικειμένων. (Μεταπτυχιακή διπλωματική εργασία, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης, 2014). Ανακτήθηκε 2023, από : <https://ikee.lib.auth.gr/record/136113>

[70] Nicole Angelyn T Lopez, Justin Parado, John Ryan Pasaoa, Joshua Morales (2016). A Comparative Study of Thread Against ZigBee, Z-Wave, Bluetooth, and Wi-Fi as a Home-Automation Networking Protocol. (Asia Pasific College). Ανακτήθηκε: 2023



## ΠΑΡΑΡΤΗΜΑ Α

```
button-sensor.h ✕
/*
 * Copyright (c) 2005, Swedish Institute of Computer Science
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the Institute nor the names of its contributors
 *    may be used to endorse or promote products derived from this software
 *    without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS ``AS IS'' AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 *
 * This file is part of the Contiki operating system.
 */
#ifndef __BUTTON_SENSOR_H__
#define __BUTTON_SENSOR_H__

#include "lib/sensors.h"

extern const struct sensors_sensor button_sensor;

#define BUTTON_SENSOR "Button"
#endif /* __BUTTON_SENSOR_H__ */
```

```
#ifndef __SENSORS_H__
#define __SENSORS_H__

#include "contiki.h"

/* some constants for the configure API */
#define SENSORS_HW_INIT 128 /* internal - used only for initialization */
#define SENSORS_ACTIVE 129 /* ACTIVE => 0 -> turn off, 1 -> turn on */
#define SENSORS_READY 130 /* read only */

#define SENSORS_ACTIVATE(sensor) (sensor).configure(SENSORS_ACTIVE, 1)
#define SENSORS_DEACTIVATE(sensor) (sensor).configure(SENSORS_ACTIVE, 0)

#define SENSORS_SENSOR(name, type, value, configure, status) \
const struct sensors_sensor name = { type, value, configure, status }

#define SENSORS_NUM (sizeof(sensors) / sizeof(struct sensors_sensor *))

#define SENSORS(...) \
const struct sensors_sensor *sensors[] = {__VA_ARGS__, NULL}; \
unsigned char sensors_flags[SENSORS_NUM]

struct sensors_sensor {
    char *      type;
    int        (* value)      (int type);
    int        (* configure) (int type, int value);
    int        (* status)     (int type);
};

const struct sensors_sensor *sensors_find(const char *type);
const struct sensors_sensor *sensors_next(const struct sensors_sensor *s);
const struct sensors_sensor *sensors_first(void);

void sensors_changed(const struct sensors_sensor *s);

extern process_event_t sensors_event;

PROCESS_NAME(sensors_process);

#endif /* __SENSORS_H__ */
```

```
*press_button.c X
#include <stdio.h>
#include "contiki.h"
#include "dev/leds.h"
#include "dev/button-sensor.h"

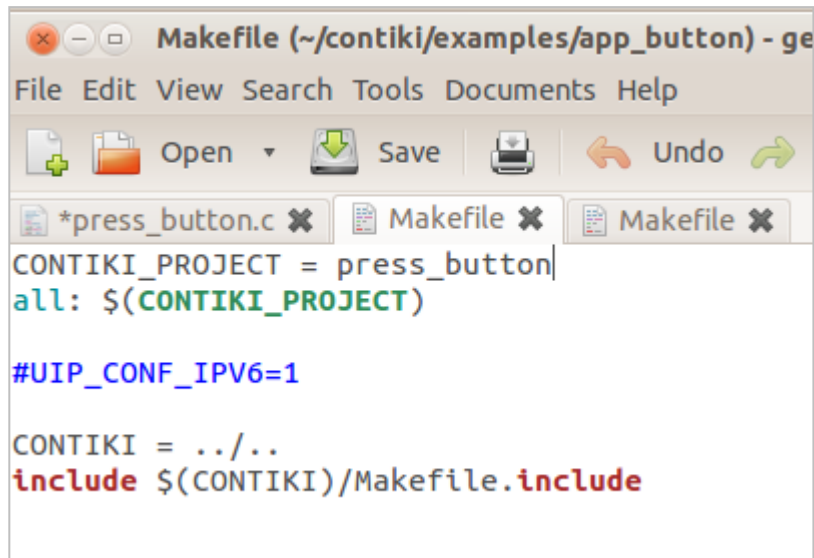
PROCESS(button_process, "Test Button"); /* added by tsellos
AUTOSTART_PROCESSES(&button_process);

static uint8_t flag;
PROCESS_THREAD(button_process, ev, data)
{
PROCESS_BEGIN();
flag=0;
SENSORS_ACTIVATE(button_sensor);
while(1)
{
PROCESS_WAIT_EVENT_UNTIL(ev==sensors_event && data==&button_sensor);
leds_toggle(LED_ALL);
if (!flag)
{
leds_on(LED_RED);
printf("LED RED %d\n", leds_get()); /* energopoihsh_red_led
}
else
{
printf("LED RED %d\n", leds_get()); /* apenergopoihsh_red_led
leds_off(LED_RED);
}
flag ^=1;
leds_toggle(LED_ALL);
}
PROCESS_END();
}
```

```
*press_button.c X Makefile X Makefile X
CONTIKI_PROJECT = hello-world
all: $(CONTIKI_PROJECT)

#UIP_CONF_IPV6=1

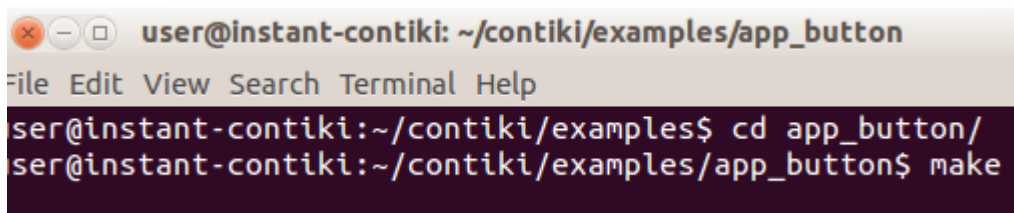
CONTIKI = ../..
include $(CONTIKI)/Makefile.include
```



```
CONTIKI_PROJECT = press_button
all: $(CONTIKI_PROJECT)

#UIP_CONF_IPV6=1

CONTIKI = ../..
include $(CONTIKI)/Makefile.include
```



```
user@instant-contiki: ~/contiki/examples/app_button
ser@instant-contiki:~/contiki/examples$ cd app_button/
ser@instant-contiki:~/contiki/examples/app_button$ make
```

```
#define HISTORY 16
static int temperature[HISTORY];
static int pressure[HISTORY]; /* added tsellos
static int humidity[HISTORY]; /* added tsellos
static int weight[HISTORY]; /* added tsellos
static int light1[HISTORY];
static int sensors_pos;
```

```
/*-----*/
static int
get_temp(void)
{
    return ((sht11_sensor.value(SHT11_SENSOR_TEMP) / 10) - 396) / 10;
}
/* added tsellos
static int
get_pressure(void)
{
    return ((sht11_sensor.value(SHT11_SENSOR_PRESSURE) / 10) - 396) / 10;
}
/* added tsellos|
static int
get_humidity(void)
{
    return ((sht11_sensor.value(SHT11_SENSOR_HUMIDITY) / 10) - 396) / 10;
}
/* added tsellos
static int
get_weight(void)
{
    return ((sht11_sensor.value(SHT11_SENSOR_WEIGHT) / 10) - 396) / 10;
}
/*-----*/
```

```
/*-----*/
static int
get_temp(void)
{
    return ((sht11_sensor.value(SHT11_SENSOR_TEMP) / 10) - 396) / 10;
}
/* added tsellos
static int
get_pressure(void)
{
    return ((sht11_sensor.value(SHT11_SENSOR_PRESSURE) / 10) - 396) / 10;
}
/* added tsellos|
static int
get_humidity(void)
{
    return ((sht11_sensor.value(SHT11_SENSOR_HUMIDITY) / 10) - 396) / 10;
}
/* added tsellos
static int
get_weight(void)
{
    return ((sht11_sensor.value(SHT11_SENSOR_WEIGHT) / 10) - 396) / 10;
}
/*-----*/
```

```
static
PT_THREAD(send_values(struct httpd_state *s))
{
    Psock_BEGIN(&s->sout);

    SEND_STRING(&s->sout, TOP);

    if(strncmp(s->filename, "/index", 6) == 0 ||
        s->filename[1] == '\0') {
        /* Default page: show latest sensor values as text (does not
           require Internet connection to Google for charts). */
        blen = 0;
        ADD("<h1>Current readings</h1>\n"
            "Light: %u<br>"
            "Temperature: %u&deg; C\n",
            "Pressure: %u; <br> \n",    /* added tsellos
            "Humidity: %u; <br> %\n",    /* added tsellos
            "Weight: %u; <br> kg\n",    /* added tsellos
            get_light(), get_temp(), get_pressure(), get_humidity(), get_weight());
        SEND_STRING(&s->sout, buf);
    }
}
```

```
PROCESS_THREAD(web_sense_process, ev, data)
{
    static struct etimer timer;
    PROCESS_BEGIN();

    sensors_pos = 0;

    etimer_set(&timer, CLOCK_SECOND * 2);
    SENSORS_ACTIVATE(light_sensor);
    SENSORS_ACTIVATE(sht11_sensor);

    while(1) {
        PROCESS_WAIT_EVENT_UNTIL(etimer_expired(&timer));
        etimer_reset(&timer);

        light1[sensors_pos] = get_light();
        temperature[sensors_pos] = get_temp();
        pressure[sensors_pos] = get_pressure(); /* added tsellos
        humidity[sensors_pos] = get_humidity(); /* added tsellos
        weight[sensors_pos] = get_weight();    /* added tsellos
        sensors_pos = (sensors_pos + 1) % HISTORY;
    }

    PROCESS_END();
}
/*-----*/
```

```

#ifndef SDA_0
#define SDA_0() (SHT11_PxDIR |= BV(SHT11_ARCH_SDA)) /* SDA Output=0 */
#define SDA_1() (SHT11_PxDIR &= ~BV(SHT11_ARCH_SDA)) /* SDA Input */
#define SDA_IS_1 (SHT11_PxIN & BV(SHT11_ARCH_SDA))

#define SCL_0() (SHT11_PxOUT &= ~BV(SHT11_ARCH_SCL)) /* SCL Output=0 */
#define SCL_1() (SHT11_PxOUT |= BV(SHT11_ARCH_SCL)) /* SCL Output=1 */
#endif

/* adr command r/w */
#define STATUS_REG_W 0x06 /* 000 0011 0 */
#define STATUS_REG_R 0x07 /* 000 0011 1 */
#define MEASURE_TEMP 0x03 /* 000 0001 1 */
#define MEASURE_HUMI 0x05 /* 000 0010 1 */
#define MEASURE_PRESS 0x09 /* 000 0100 1 ADDED BY TSELLOS*/
#define MEASURE_WEIGHT 0x09 /* 000 1000 1 ADDED BY TSELLOS*/
#define RESET 0x1e /* 000 1111 0 */

/* This can probably be reduced to 250ns according to data sheet. */
#ifndef delay_400ns
#define delay_400ns() _NOP()
#endif
/*-----*/

```

```

#ifndef SDA_0
#define SDA_0() (SHT11_PxDIR |= BV(SHT11_ARCH_SDA)) /* SDA Output=0 */
#define SDA_1() (SHT11_PxDIR &= ~BV(SHT11_ARCH_SDA)) /* SDA Input */
#define SDA_IS_1 (SHT11_PxIN & BV(SHT11_ARCH_SDA))

#define SCL_0() (SHT11_PxOUT &= ~BV(SHT11_ARCH_SCL)) /* SCL Output=0 */
#define SCL_1() (SHT11_PxOUT |= BV(SHT11_ARCH_SCL)) /* SCL Output=1 */
#endif

/* adr command r/w */
#define STATUS_REG_W 0x06 /* 000 0011 0 */
#define STATUS_REG_R 0x07 /* 000 0011 1 */
#define MEASURE_TEMP 0x03 /* 000 0001 1 */
#define MEASURE_HUMI 0x05 /* 000 0010 1 */
#define MEASURE_PRESS 0x09 /* 000 0100 1 ADDED BY TSELLOS*/
#define MEASURE_WEIGHT 0x09 /* 000 1000 1 ADDED BY TSELLOS*/
#define RESET 0x1e /* 000 1111 0 */

/* This can probably be reduced to 250ns according to data sheet. */
#ifndef delay_400ns
#define delay_400ns() _NOP()
#endif
/*-----*/

```

```

unsigned int sht11_temp(void);
unsigned int sht11_humidity(void);
unsigned int sht11_pressure(void); /* added tsellos
unsigned int sht11_weight(void); /* added tsellos
unsigned int sht11_sreg(void);
int sht11_set_sreg(unsigned);

```

```

/*-----*/
static int
value(int type)
{
    switch(type) {
        /* Photosynthetically Active Radiation. */
        case SHT11_SENSOR_TEMP:
            return sht11_temp();

        /* Total Solar Radiation. */
        case SHT11_SENSOR_HUMIDITY:|
            return sht11_humidity();
        case SHT11_SENSOR_PRESSURE: /* added tsellos
            return sht11_pressure();
        case SHT11_SENSOR_WEIGHT: /* added tsellos
            return sht11_weight();
        case SHT11_SENSOR_BATTERY_INDICATOR:
            return sht11_sreg() & 0x40? 1: 0;
    }
    return 0;
}
/*-----*/

```

```
#include "lib/sensors.h"

extern const struct sensors_sensor sht11_sensor;

#define SHT11_SENSOR_TEMP           0
#define SHT11_SENSOR_HUMIDITY      1
#define SHT11_SENSOR_BATTERY_INDICATOR 2
#define SHT11_SENSOR_PRESSURE      3
#define SHT11_SENSOR_WEIGHT        4
```