

# **Ανοικτό Πανεπιστήμιο Κύπρου**

**Σχολή Θετικών και Εφαρμοσμένων Επιστημών**

## **Μεταπτυχιακή Διατριβή στα Πληροφοριακά Συστήματα**



**Δημιουργία Ασύρματου Δικτύου Κόμβων Μετρήσεων  
Χαμηλού Κόστους**

**Φίλιππος Γώττας**

**Επιβλέπων Καθηγητής  
Μιχαήλ Βασιλακόπουλος**

**Ιούνιος 2013**

# **Ανοικτό Πανεπιστήμιο Κύπρου**

## **Σχολή Θετικών και Εφαρμοσμένων Επιστημών**

**Δημιουργία Ασύρματου Δικτύου Κόμβων Μετρήσεων  
Χαμηλού κόστους**

**Φίλιππος Γώττας**

**Επιβλέπων Καθηγητής  
Μιχαήλ Βασιλακόπουλος**

Η παρούσα μεταπτυχιακή διατριβή υποβλήθηκε  
προς μερική εκπλήρωση των απαιτήσεων για απόκτηση

μεταπτυχιακού τίτλου σπουδών  
στα Πληροφοριακά Συστήματα

από τη Σχολή Θετικών και Εφαρμοσμένων Επιστημών  
του Ανοικτού Πανεπιστημίου Κύπρου

**Ιούνιος 2013**

## Περίληψη

Τα ασύρματα δίκτυα κόμβων μετρήσεων τα τελευταία χρόνια εμφανίζονται σε περισσότερους τομείς της ανθρώπινης δραστηριότητας. Σε αυτό συνέβαλε η μείωση του κόστους κατασκευής των ολοκληρωμένων κυκλωμάτων (ολοκληρωμένα πολύ υψηλής κλίμακας ολοκλήρωσης - VLSI), η ανάγκη της βιομηχανίας να ελέγχει τις διαδικασίες παραγωγής καθώς και κάθε άλλη δραστηριότητα όπου πρέπει να μετρηθούν φυσικά ή βιομηχανικά μεγέθη να αποθηκευθούν ή να μεταδοθούν για περαιτέρω επεξεργασία.

Η παρούσα μεταπτυχιακή διατριβή ασχολείται με την κατασκευή ενός χαμηλού κόστους ασύρματου δικτύου κόμβων μετρήσεων. Το ασύρματο δίκτυο αποτελείται από έναν κεντρικό κόμβο και πολλούς απομακρυσμένους κόμβους. Ο κεντρικός κόμβος έχει άπειρη ενέργεια, επικοινωνεί με όλους τους απομακρυσμένους κόμβους και είναι υπεύθυνος για την καταχώρηση των μετρήσεων σε βάση δεδομένων. Οι απομακρυσμένοι κόμβοι αποστέλλουν τις μετρήσεις τους σε προγραμματισμένα χρονικά διαστήματα στον κεντρικό κόμβο. Ο χρήστης μέσω Web-based διεπαφής μπορεί να λαμβάνει τις μετρήσεις των κόμβων καθώς και γραφήματα.

Για να υλοποιηθεί το ασύρματο δίκτυο κατασκευάστηκαν ο κεντρικός κόμβος και πέντε απομακρυσμένοι κόμβοι. Όλοι οι κόμβοι αποτελούνται από έναν μικροελεγκτή 8-bit και έναν πομποδέκτη. Το πρόγραμμα του μικροελεγκτή αναπτύχθηκε σε C και περιλαμβάνει ένα «απλό» πρωτόκολλο επικοινωνίας και μια μηχανή καταστάσεων. Τον περισσότερο χρόνο τόσο ο μικροελεγκτής όσο και ο πομποδέκτης βρίσκονται σε κατάσταση «ύπνου» επιτυγχάνοντας ελάχιστη κατανάλωση ενέργειας.

Βασικές παράμετροι για την ανάπτυξη του συστήματος, θεωρήθηκαν: το χαμηλό κόστος, η χαμηλή κατανάλωση ενέργειας, η διαθεσιμότητα των υλικών και η ευκολία κατασκευής.

Η μεταπτυχιακή διατριβή αναδεικνύει όλες τις βασικές έννοιες για την υλοποίηση ενός ασύρματου δικτύου αισθητήρων χαμηλού κόστους. Η απλότητα στο σχεδιασμό ενός πρωτοκόλλου επιπέδου πρόσβασης μέσου παίζει σημαντικό ρόλο σε δίκτυα με μικρή κίνηση και συμβάλει σημαντικά στην εξοικονόμηση ενέργειας που ήταν βασική παράμετρος του συστήματος.

## Summary

During recent years, wireless sensor networks nodes appear in most areas of human activity. This was helped by the reduction of the manufacturing cost of integrated circuits (very large scale integration - VLSI), the need for industry to control production processes and any other activity where measurement of natural or industrial sizes is needed to be stored or transmitted for further processing.

This M.Sc. dissertation deals with the construction of low cost wireless sensor network nodes. The wireless network consists of a central hub and multiple remote nodes. The central node has infinite energy, communicates with all remote nodes and is responsible for recording the measurements in a database. The remote nodes send their measurements, at scheduled intervals, to the central hub. The user through Web-based interface may take measurements of nodes and graphs.

To implement the wireless sensor network, a central node and five remote nodes are constructed. All nodes consist of an 8-bit microcontroller and a transceiver. The microcontroller program has been developed in C and includes a "simple" communication protocol and a state machine. Most of the time, both the microcontroller and the transceiver are in "sleep" state achieving minimum energy consumption.

Key parameters for the development of the system were: low cost, low power consumption, the availability of materials and ease of manufacturing.

The M.Sc. dissertation highlights all the key concepts for the implementation of a wireless sensor network of low cost. The simplicity in the design of a medium access layer protocol plays an important role in networks with little traffic and contributes significantly to energy saving which was a key parameter of the system.

## Ευχαριστίες

Πάνω απ' όλα είμαι ευγνώμων στην οικογένειά μου που με στήριξε και με βοήθησε όλα αυτά τα χρόνια.

Θα ήθελα να ευχαριστήσω τον καθηγητή κ. Μιχαήλ Βασιλακόπουλο, Αναπληρωτή Καθηγητή Συστημάτων Βάσεων Δεδομένων του Πανεπιστημίου Στερεάς Ελλάδας , για την επίβλεψη της παρούσας μεταπτυχιακής διατριβής, την εμπιστοσύνη και την καθοδήγηση που επέδειξε στο πρόσωπό μου.

# Περιεχόμενα

1	Εισαγωγή.....	1
1.1	Ασύρματα Δίκτυα Κόμβων Μετρήσεων (Ασύρματα Δίκτυα Αισθητήρων).....	1
1.2	Δομικά Στοιχεία Ασύρματου Κόμβου Μετρήσεων .....	2
1.3	Εφαρμογές των Ασύρματων Δικτύων Αισθητήρων .....	3
1.4	Ασύρματοι Κόμβοι Μετρήσεων.....	4
1.4.1	Σταθμός Mica 2.....	5
1.4.2	Σταθμός Sun SPOT.....	6
1.4.3	Σταθμός BTnode .....	7
1.4.4	TinyNode.....	8
2	Πρωτόκολλα Επικοινωνίας Ασύρματων Δικτύων Αισθητήρων .....	1
2.1	Πρωτόκολλο Επικοινωνίας.....	1
2.2	Πρωτοκόλλα Επικοινωνίας για Ασύρματα Δίκτυα Αισθητήρων.....	2
2.2.1	Πρωτόκολλο LEACH (Low-Energy Adaptive Clustering Hierarchy).....	2
2.2.2	Πρωτόκολλο Sensor-MAC (S-MAC) .....	3
2.2.3	Πρωτόκολλο Wise-MAC (Wireless Sensor-MAC).....	4
2.3	Συμπεράσματα.....	5
3	Απλό πρωτόκολλο Επικοινωνίας – Λειτουργία Κόμβων .....	6
3.1	Απλό Πρωτόκολλο Επικοινωνίας.....	6
3.2	Υλοποίηση Απλού Πρωτοκόλλου Επικοινωνίας για Ασύρματο Δίκτυο Αισθητήρων .....	7
3.2.1	Πρωτόκολλο LBT (Listen Before Talk) .....	7
3.3	Λειτουργία του κόμβου .....	9
3.3.1	Καταστάσεις λειτουργίας απομακρυσμένου κόμβου (NODE) .....	10
3.3.2	Καταστάσεις λειτουργίας κεντρικού κόμβου (MASTER) .....	13
3.4	FIFO και PACKET HANDLER modes.....	14
3.4.1	Λειτουργία FIFO σε κατάσταση μετάδοσης (TX) .....	18
3.4.2	Λειτουργία FIFO σε κατάσταση λήψης (RX) .....	19
3.4.3	Λειτουργία PH (Automatic Packet Handler) .....	22
3.5	Δομή Πακέτων Ασύρματου Κόμβου.....	27
3.6	Κατανάλωση Ενέργειας .....	29
4	Επιλογή Υλικού Αρχιτεκτονική Κόμβων .....	32
4.1	Επιλογή μικροελεγκτή .....	32
4.2	Συσκευή Προγραμματισμού μικροελεγκτή .....	33
4.3	Επιλογή πομποδέκτη .....	34
4.4	Επικοινωνία Κύριου Κόμβου με Η/Υ .....	37
4.5	Αρχιτεκτονική Κόμβων .....	38
4.5.1	Αρχιτεκτονική απομακρυσμένων Κόμβων.....	38
4.5.2	Αρχιτεκτονική Κεντρικού Κόμβου.....	39
5	Κατασκευή Ασύρματων κόμβων Μετρήσεων.....	41

5.1	Κατασκευή Κόμβων σε Πλακέτα Δοκιμών .....	41
5.1.1	Σύνδεση θύρας προγραμματισμού ISP μικροελεγκτή.....	41
5.1.2	Σύνδεση Μικροελεγκτή Πομποδέκτη .....	43
5.1.3	Σύνδεση Μικροελεγκτή με Ηλεκτρονικό Υπολογιστή .....	45
5.2	Κατασκευή Κόμβων σε Τυπωμένο Κύκλωμα .....	49
5.2.1	Σχεδιασμός Τυπωμένου Κυκλώματος Κόμβων .....	49
6	Επικοινωνία Κεντρικού Κόμβου με τη Βάση Δεδομένων .....	59
6.1	Αποστολή Δεδομένων σε Ηλεκτρονικό Υπολογιστή .....	59
6.1.1	Ερώτημα SQL- Πρόγραμμα Επικοινωνίας .....	61
6.2	Δομή της Βάσης Δεδομένων .....	61
7	Διεπαφή Χρήστη .....	63
7.1	Οθόνη Ασύρματου Δικτύου Μετρήσεων .....	63
7.2	Οθόνη Μετρήσεων .....	67
7.2.1	Γραφική Παράσταση Μετρήσεων.....	68
7.3	Ενημέρωση Κόμβου .....	69
8	Επίλογος.....	71
	Βιβλιογραφία .....	74
A	Ηλεκτρονικές Πηγές Πληροφόρησης.....	A-1
A.1	Δικτυακοί Τόποι (Υλικό).....	A-1
A.2	Δικτυακοί Τόποι (Λογισμικό ) .....	A-2
B	Προγράμματα.....	B-1
B.1	Προγράμματα Μικροελεγκτή.....	B-1
B.2	Προγράμματα Επικοινωνίας RS232 - MySQL.....	B-3
B.3	Προγράμματα Διεπαφής Χρήστη .....	B-4

## Περιεχόμενα Εικόνων

Εικόνα 1.1	Δομικά στοιχεία Ασύρματου Κόμβου Μετρήσεων.....	2
Εικόνα 1.2	Σταθμός Mica 2 .....	5
Εικόνα 1.3	Αρχιτεκτονική Mica2 .....	6
Εικόνα 1.4	Sun Spot Ασύρματος Κόμβος.....	7
Εικόνα 1.5	BTnode .....	7
Εικόνα 1.6	Κόμβος TinyNode .....	8
Εικόνα 2.1	WiseMAC πρωτόκολλο [06].....	4
Εικόνα 3.1	Πρωτόκολλο MAC – Διαδικασία Listen Before Talk.....	8
Εικόνα 3.2	Λειτουργία των κόμβων .....	10
Εικόνα 3.3	Μηχανή καταστάσεων απομακρυσμένου κόμβου.....	11
Εικόνα 3.4	Μηχανή καταστάσεων κεντρικού κόμβου (MASTER).....	13
Εικόνα 3.5	Γενική Δομή Πακέτου .....	14
Εικόνα 3.6	Καταχωρητές τρόπου λειτουργίας και ελέγχου [13].....	15
Εικόνα 3.7	Καταχωρητές –Λειτουργία FIFO, Διακοπή Αποστολή Πακέτου.....	18
Εικόνα 3.8	Καταχωρητής 3Eh, μέγεθος πακέτου .....	19
Εικόνα 3.9	Καταχωρητές 03h και 05h - Διακοπή «έγκυρο πακέτο».....	20
Εικόνα 3.10	Δομή πακέτου με ενεργοποιημένη τη λειτουργία PH .....	21
Εικόνα 3.11	Δομή πακέτου με απενεργοποιημένη τη λειτουργία PH.....	21
Εικόνα 3.12	Καταχωρητής 30h, έλεγχος του Packet Handler.....	22
Εικόνα 3.13	Καταχωρητής 33h και 34h, έλεγχος του Header .....	23
Εικόνα 3.14	Καταχωρητής 32h έλεγχος Header 1.....	25
Εικόνα 3.15	Δομή Πακέτου Απομακρυσμένου Κόμβου .....	27
Εικόνα 3.16	Μορφή Δεδομένων Απομακρυσμένου Κόμβου.....	27
Εικόνα 3.17	Μορφή Δεδομένων πακέτου ACK κύριου κόμβου.....	28
Εικόνα 3.18	Προφίλ κατανάλωσης ενέργειας απομακρυσμένου κόμβου.....	29
Εικόνα 3.19	Κατανάλωση φορτίου (μετάδοση κάθε λεπτό και κάθε 10 λεπτά) .....	31
Εικόνα 4.1	Μικροελεγκτής ATMEGA328P-PU .....	33
Εικόνα 4.2	Προγραμματιστής Μικροελεγκτών ATMEL.....	34
Εικόνα 4.3	Θύρα προγραμματισμού ISP .....	34
Εικόνα 4.4	Τάση χρήσης συχνοτήτων σε καταναλωτικές, βιομηχανικές και εφαρμογές στην αυτοκινητοβιομηχανία [05] .....	36
Εικόνα 4.5	Πομποδέκτης RFM22B-D της HOPE .....	37
Εικόνα 4.6	Μετατροπέας USB-Serial.....	37
Εικόνα 4.7	Ολοκληρωμένο διασύνδεσης TTL-RS232 .....	38
Εικόνα 4.8	Απομακρυσμένοι κόμβοι (NODE) .....	38
Εικόνα 4.9	Αρχιτεκτονική Κεντρικού κόμβου (MASTER) .....	39
Εικόνα 5.1	Θύρα προγραμματισμού ISP 6 ακροδεκτών.....	42
Εικόνα 5.2	Συνδεσμολογία θύρας ISP και μικροελεγκτή ATMEGA328P-PU .....	42
Εικόνα 5.3	RFM22B μετατροπέας.....	43
Εικόνα 5.4	Σύνδεση μετατροπέα πομποδέκτη στην πλακέτα δοκιμών.....	44
Εικόνα 5.5	Πλήρης λειτουργικός κόμβος σε πλακέτα δοκιμών.....	44
Εικόνα 5.6	Μετατροπή λογικών επιπέδων [21] .....	45
Εικόνα 5.7	BS170 MOSFET.....	46
Εικόνα 5.8	Ολοκληρωμένο HIN232 - Κύκλωμα TTL σε RS232 .....	47
Εικόνα 5.9	Ηλεκτρονικό σχέδιο κεντρικού κόμβου .....	48
Εικόνα 5.10	Κεντρικός κόμβος – Επικοινωνία RS232.....	48
Εικόνα 5.11	Αρχικό στάδιο ηλεκτρονικού κυκλώματος .....	49



Εικόνα 5.12 Ηλεκτρονικό σχέδιο απομακρυσμένου κόμβου .....	50
Εικόνα 5.13 Δημιουργία διαδρόμων τυπωμένου κυκλώματος .....	51
Εικόνα 5.14 Μάσκα επάνω όψης τυπωμένου .....	52
Εικόνα 5.15 Μάσκα κάτω όψης τυπωμένου.....	52
Εικόνα 5.16 Μεταφορά της μάσκας (επάνω όψη) στην πλακέτα.....	53
Εικόνα 5.17 Μεταφορά της μάσκας (επάνω όψη) σε φωτοευαίσθητη πλακέτα .....	54
Εικόνα 5.18 Διαδικασία αποχάλκωσης.....	55
Εικόνα 5.19 Τελική μορφή της πλακέτας (αρχική έκδοση υλικού 0.1).....	55
Εικόνα 5.20 Απομακρυσμένος κόμβος (αρχική έκδοση υλικού 0.1) .....	56
Εικόνα 5.21 Απομακρυσμένος Κόμβος (Έκδοση υλικού 0.3) .....	57
Εικόνα 5.22 Απομακρυσμένος Κόμβος 3D (Έκδοση υλικού 0.3).....	57
Εικόνα 5.23 Κεντρικός Κόμβος σε τρισδιάστατη απεικόνιση (Έκδοση υλικού 0.3) .....	58
Εικόνα 6.1 Πρόγραμμα ανάγνωσης σειριακής θύρας .....	60
Εικόνα 6.2 Σχήμα της Βάσης Δεδομένων .....	62
Εικόνα 7.1 Δομή Διεπαφής Χρήστη .....	64
Εικόνα 7.2 Αρχική οθόνη Χρήστη.....	65
Εικόνα 7.3 Είσοδος του Χρήστη στην εφαρμογή .....	65
Εικόνα 7.4 Παράμετροι Διεπαφής.....	66
Εικόνα 7.5 Μετρήσεις κόμβου 20.....	67
Εικόνα 7.6 Γραφική απεικόνιση μετρήσεων κόμβου 20 .....	69
Εικόνα 7.7 Ενημέρωση παραμέτρων κόμβου 20 .....	69

# Κεφάλαιο 1

## Εισαγωγή

### 1.1 Ασύρματα Δίκτυα Κόμβων Μετρήσεων (Ασύρματα Δίκτυα Αισθητήρων)

Τα ασύρματα δίκτυα αισθητήρων (ΑΔΑ, Wireless Sensor Network – WSN) είναι μια νέα σχετικά κατηγορία δικτύων τα οποία αποτελούνται από πολλούς ασύρματους κόμβους οι οποίοι μετρούν τοπικές ή περιβαλλοντικές παραμέτρους και αποστέλλουν τις μετρήσεις τους σε έναν κεντρικό κόμβο. Το περιβάλλον στο οποίο ζούμε και εργαζόμαστε (γραφεία, σπίτια, εργοστάσια ) και οι μηχανές που χρησιμοποιούμε έχουν πληθώρα αισθητήρων. Για να ελέγξουμε μια παράμετρο σε μια διαδικασία παραγωγής θα πρέπει να την μετρήσουμε και ανάλογα να παρέμβουμε . Για να επιτευχθεί ο έλεγχος φυσικών ή βιομηχανικών μεγεθών χρειάζεται να χρησιμοποιηθούν αισθητήρες. Τα τελευταία χρόνια το κόστος κατασκευής, η κατανάλωση ενέργειας και το μέγεθος των αισθητήρων έχει μειωθεί σημαντικά. Πολλές φορές το κόστος εγκατάστασης είναι πολλαπλάσιο του κόστους του αισθητήρα. Για παράδειγμα το κόστος καλωδίωσης για ένα απλό διακόπτη ξεπερνά κατά πολύ το κόστος του ίδιου του διακόπτη. Εάν θέλουμε να μετακινήσουμε τον διακόπτη, το κόστος επανεγκατάστασης είναι ακόμη μεγαλύτερο [01]. Το κόστος πολλαπλασιάζεται όταν έχουμε να κάνουμε με βιομηχανικές

εγκαταστάσεις όπου μια αλλαγή μπορεί να μην είναι εύκολα πραγματοποιήσιμη ή να είναι ασύμφορη .

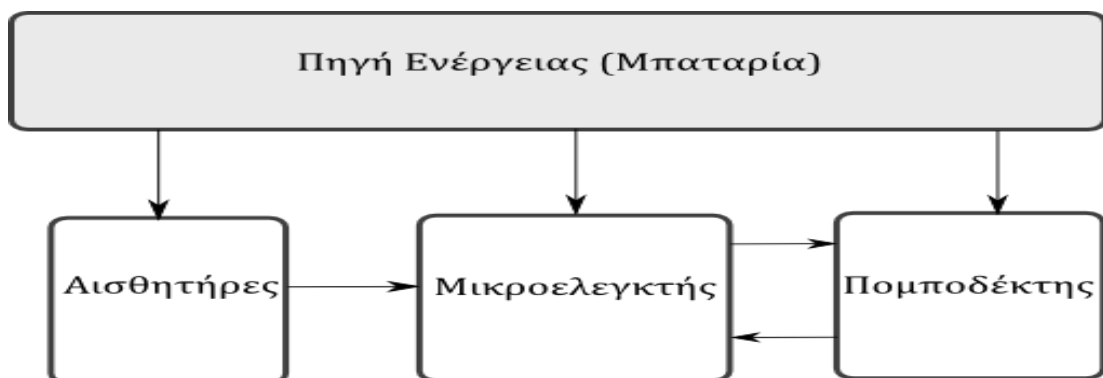
Οι κόμβοι των ΑΔΑ έχουν περιορισμένους υπολογιστικούς και ενεργειακούς πόρους. Οι περιορισμένοι πόροι των κόμβων επιβάλλουν την ανάπτυξη συστημάτων τα οποία θα απαιτούν ελάχιστη ενέργεια ώστε η διάρκεια ωφέλιμης λειτουργίας τους να είναι μεγαλύτερη. Πολλές φορές οι κόμβοι τοποθετούνται σε θέσεις που είναι δύσκολο να προσπελαστούν ή το κόστος αντικατάστασής τους είναι πολλαπλάσιο του κόστους του ίδιου του κόμβου. Με την ανάπτυξη της τεχνολογίας το κόστος των κόμβων έχει μειωθεί σημαντικά με αποτέλεσμα να διευρύνεται το πεδίο εφαρμογής τους[02].

## 1.2 Δομικά Στοιχεία Ασύρματου Κόμβου Μετρήσεων

Ένας ασύρματος κόμβος μετρήσεων αποτελείται από τουλάχιστον τέσσερα βασικά δομικά στοιχεία[03]:

- Μικροελεγκτή
- Πομποδέκτη
- Αισθητήρες μετρήσεων
- Πηγή ενέργειας (μπαταρία)

Στην Εικόνα 1.1 παρουσιάζονται τα βασικά δομικά στοιχεία ενός ασύρματου κόμβου μετρήσεων.



Εικόνα 1.1 Δομικά στοιχεία Ασύρματου Κόμβου Μετρήσεων

Βασικές λειτουργίες που εκτελεί ο μικροελεγκτής είναι: διαβάσει τις μετρήσεις από τους αισθητήρες των μετρούμενων φυσικών μεγεθών και τις προωθεί στον πομποδέκτη, εποπτεύει την συνεργασία του πομποδέκτη με τον κεντρικό κόμβο και υλοποιεί το πρωτόκολλο επικοινωνίας. Ο πομπός αναλαμβάνει να στείλει τα δεδομένα στον κύριο κόμβο και περιμένει να λάβει το σήμα επιβεβαίωσης. Όταν ο μικροελεγκτής λάβει από τον δέκτη το σήμα επιβεβαίωσης (ACK) τότε πηγαίνει σε κατάσταση χαμηλής κατανάλωσης μέχρι τον επόμενο κύκλο.

Σήμερα υπάρχουν διαφορετικές τεχνολογίες διαθέσιμες για την κατασκευή αισθητήρων ακριβείας και χαμηλού κόστους. Μικροηλεκτρομηχανικοί αισθητήρες (MEMS) μπορούν να μετρήσουν δυνάμεις, χημικές συγκεντρώσεις και περιβαλλοντικά μεγέθη όπως υγρασία, θερμοκρασία, ταχύτητα ανέμου, φωτεινότητα κ.λπ. Αυτές οι συσκευές παράγονται μαζικά και έχουν χαμηλό κόστος. Πολλοί αισθητήρες είναι ψηφιακοί και βαθμονομημένοι ώστε να παρέχουν τις μετρήσεις τους σε ψηφιακή μορφή αξιόπιστα χωρίς να χρειάζεται μετατροπή από αναλογικό σε ψηφιακό σήμα από τον μικροελεγκτή. Αυτό επιτρέπει τη χρήση μικρότερων και φθηνότερων μικροελεγκτών που τους απαλλάσσει από διαδικασίες ψηφιοποίησης αναλογικών μεγεθών επιτρέποντάς τους να επιτελούν γρηγορότερα τον έλεγχο των διαδικασιών.

Οι πομποδέκτες επιφορτίζονται με το έργο αποστολής και λήψης πακέτων δεδομένων. Λαμβάνουν από τον μικροελεγκτή τα δεδομένα που πρέπει να εκπέμψουν σε ψηφιακή μορφή, τα μετατρέπουν σε μορφή κατάλληλη για να αποσταλούν στον δέκτη και το αντίστροφο. Υπάρχουν πολλών ειδών πομποδέκτες και η επιλογή τους γίνεται ανάλογα με το είδος της εφαρμογής, την ακτίνα κάλυψης που έχουν, το κόστος και του χώρου στον οποίο θα εγκατασταθεί η εφαρμογή. Πομποδέκτες που χρησιμοποιούνται ευρέως είναι: Sub-GHz (σε συχνότητες κάτω του 1GHz) και 2,4GHz (Wi-Fi, ZigBee) ή υπερύθρων.

### **1.3 Εφαρμογές των Ασύρματων Δικτύων Αισθητήρων**

Τα πεδία εφαρμογής των ΑΔΑ καλύπτουν πολλές επιστημονικές και ερευνητικές περιοχές. Ενδεικτικά αναφέρονται[03] :

- Εφαρμογές Υγείας

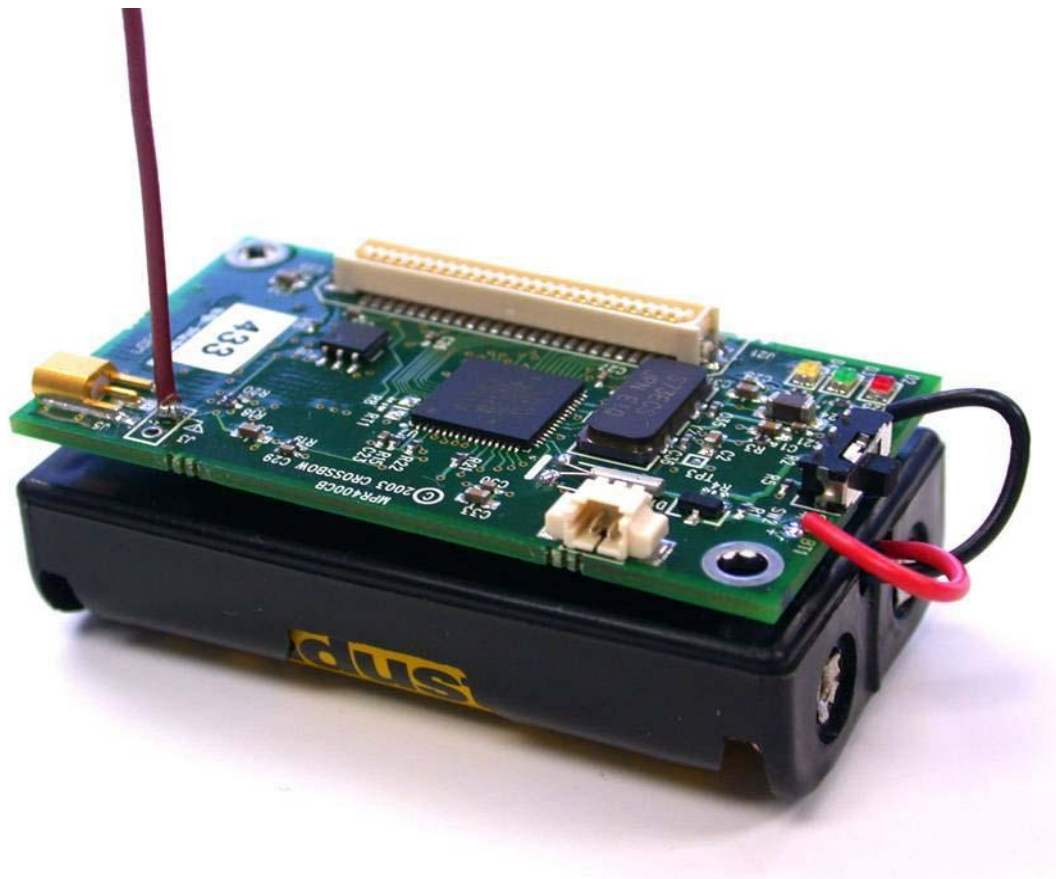
- Παρακολούθηση ασθενών (απομακρυσμένη παρακολούθηση φυσιολογικών τιμών)
- Εντοπισμός γιατρών και ασθενών σε νοσοκομείο
- Βοήθεια ηλικιωμένων
- Χορήγηση φαρμάκων
- Οικιακές εφαρμογές
  - Οικιακός αυτοματισμός
  - Αυτόματη ανάγνωση μετρητών
- Εμπορικές Εφαρμογές
  - Έλεγχος περιβάλλοντος σε βιομηχανίες και γραφεία
  - Έλεγχος αποθεμάτων
  - Παρακολούθηση και εντοπισμός οχημάτων
  - Επιτήρηση κυκλοφορίας
- Στρατιωτικές εφαρμογές
  - Παρακολούθηση φίλιων δυνάμεων
  - Επιτήρηση πεδίου μάχης
  - Ανίχνευση πυρηνικής, βιολογικής και χημικής επίθεσης
- Περιβαλλοντικές εφαρμογές
  - Ανίχνευση δασικών πυρκαγιών
  - Ανίχνευση πλημμυρών
- Αγροτικές Εφαρμογές
  - Γεωργία ακριβείας
  - Παρακολούθηση περιβαλλοντικών μεγεθών
  - Παρακολούθηση στάθμης δεξαμενών

## 1.4 Ασύρματοι Κόμβοι Μετρήσεων

Στο εμπόριο υπάρχουν αρκετοί κόμβοι για τη δημιουργία ασύρματων δικτύων μετρήσεων. Πολλοί κόμβοι υποστηρίζουν λειτουργικά συστήματα ανεπτυγμένα για ασύρματους κόμβους, διαθέτουν αρκετή μνήμη και άλλα χαρακτηριστικά όπως κάρτες επέκτασης, εξωτερική μνήμη, ενσωματωμένο ψηφιακό θερμότερο κλπ. Μερικοί από τους εμπορικούς ασύρματους κόμβους μετρήσεων περιγράφονται παρακάτω [04].

### 1.4.1 Σταθμός Mica 2

Ο ασύρματος κόμβος Mica2 της CrossBow είναι ένας ασύρματος κόμβος χαμηλής κατανάλωσης που στηρίζεται στον μικροελεγκτή ATmega128L της ATMEL και στον πομποδέκτη CC1000 της Texas Instruments [04].

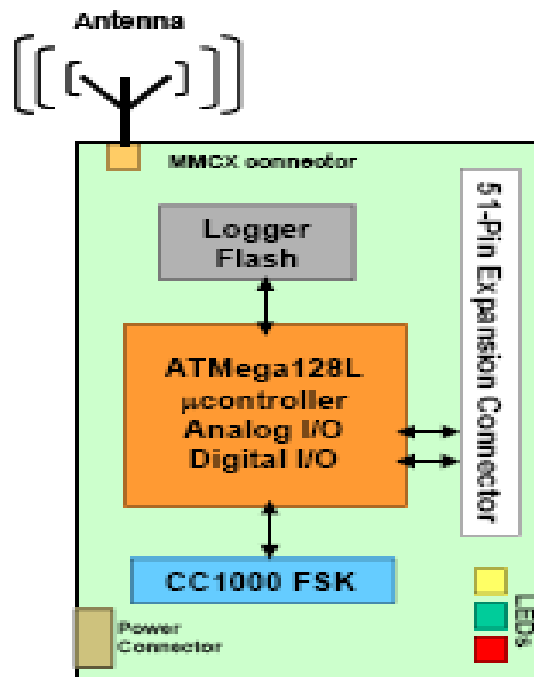


Εικόνα 1.2 Σταθμός Mica 2

Η αρχιτεκτονική του ασύρματου κόμβου mica2 φαίνεται στην Εικόνα 1.3 και τα βασικά χαρακτηριστικά του συνοψίζονται παρακάτω:

- 4 Kbytes RAM
- 128 Kbytes Flash memory
- >25mA(Tx), <15μA(sleep) στα 3,3V
- 8MHz ρολόι συστήματος

- 512 Kbytes Flash μνήμη μετρήσεων
- 4Kbytes EEPROM
- Θύρα επέκτασης (Σύνδεση καρτών επέκτασης, αισθητήριων κ.λ.π)



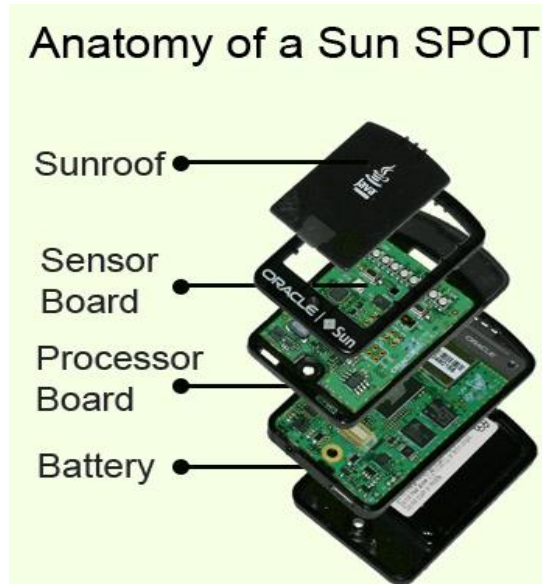
Εικόνα 1.3 Αρχιτεκτονική Mica2

#### 1.4.2 Σταθμός Sun SPOT

Ο κόμβος Sun SPOT είναι ένας κόμβος ο οποίος προγραμματίζεται σε Java. Η βασική έκδοση περιλαμβάνει επιταχυνσιόμετρο, αισθητήρα φωτός και θερμοκρασίας, πομποδέκτη, 2 διακόπτες ελέγχου, 5 ψηφιακές Εισόδους/Εξόδους, 6 αναλογικές εισόδους, 4 ψηφιακές εξόδους, και μια επαναφορτιζόμενη μπαταρία [04]. Τα βασικά χαρακτηριστικά της Rev8 είναι:

- 1MByte RAM
- 8Mbytes Flash

- Μικροελεγκτής AT91SAM9G20
- Πομποδέκτης CC2420 της Texas Instruments



Εικόνα 1.4 Sun Spot Ασύρματος Κόμβος

### 1.4.3 Σταθμός BTnode

Ο κόμβος BTnode είναι ένας ασύρματος κόμβος μετρήσεων αναπτύχθηκε από την ΕΤΗ Zurich βασισμένος σε Ericsson Bluetooth πομποδέκτη και μικροελεγκτή ATmega128  
Εικόνα 1.5.



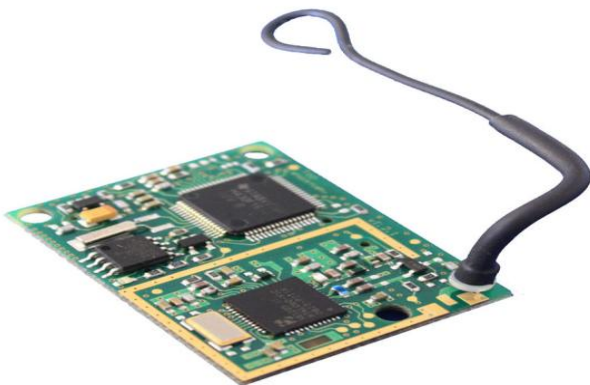
Εικόνα 1.5 BTnode



Εκτός από το πομποδέκτη Bluetooth διαθέτει και ασύρματο πομποδέκτη με το ολοκληρωμένο CC1000 της Texas Instruments καθώς και 4x60KBytes εξωτερικής μνήμης. Τα χαρακτηριστικά του είναι παρόμοια με αυτά του Mica2. Διαθέτει επίσης δύο θύρες επέκτασης των 15 και 40 ακροδεκτών στις οποίες μπορούν να συνδεθούν κάρτες επέκτασης. Είναι συμβατός με το λειτουργικό σύστημα TinyOS και προγραμματίζεται σε C [04].

#### 1.4.4 TinyNode

Ο κόμβος TinyNode είναι βελτιστοποιημένος να λειτουργεί με το λειτουργικό TinyOS και διαθέτει 19 εισόδους -εξόδους εκ των οποίων 6 αναλογικές εισόδους, 2 αναλογικές εξόδους καθώς και σειριακή θύρα. Λειτουργεί με τον μικροελεγκτή TI MSP430 και έχει λιγότερο από 6μsec χρόνο αφύπνισης.



**Εικόνα 1.6** Κόμβος TinyNode

# Κεφάλαιο 2

## Πρωτόκολλα Επικοινωνίας

### Ασύρματων Δικτύων

### Αισθητήρων

#### 2.1 Πρωτόκολλο Επικοινωνίας

Το πρωτόκολλο επικοινωνίας, επιπέδου πρόσβασης μέσου (MAC: Medium Access Control), των ασύρματων κόμβων παίζει σημαντικό ρόλο στην κατασκευή ενός ασύρματου δικτύου. Το πρωτόκολλο επικοινωνίας MAC είναι υπεύθυνο για την πρόσβαση στο κοινό μέσο επικοινωνίας. Έχουν αναπτυχθεί πολλά πρότυπα για ασύρματα δίκτυα, τα πιο δημοφιλή (διαθέσιμα εμπορικά) είναι: οι διάφορες εκδόσεις του προτύπου IEEE 802.11 (802.11b γνωστό ως Wi-Fi), το πρότυπο IEEE 802.16x γνωστό ως WiMax, το πρότυπο IEEE 802.15.1 γνωστό ως Bluetooth. Αυτά τα πρότυπα, επίπεδου πρόσβασης μέσου, σχεδιάστηκαν χωρίς να θέτουν αυστηρούς περιορισμούς στην κατανάλωση ενέργειας και δύσκολα μπορούν να καλύψουν τις ανάγκες ενός ασύρματου δικτύου κόμβων μετρήσεων. Με την ανάπτυξη των ασύρματων δικτύων

αισθητήρων υπάρχει έντονο ερευνητικό ενδιαφέρον για πρωτόκολλα που θα συμβάλουν στην εξοικονόμηση ενέργειας [05].

Για τα δίκτυα sub-GHz υπάρχουν πρότυπα όπως το Zig-Bee (το οποίο είναι και το μόνο που διαθέτει εκδόσεις τόσο για τα δίκτυα των 2,4GHz όσο και για δίκτυα sub-GHz στις συχνότητες των 868MHz και 900MHz), EnOcean, io-homecontrol®, ONE-NET, INSTEON® και Z-Wave [05]. Ενώ τα πρότυπα μας αποδεσμεύουν από τους κατασκευαστές, αυξάνουν ωστόσο το κόστος και το μέγεθος των συστημάτων. Σε ένα ασύρματο κόμβο 2,4GHz Zig-Bee η υλοποίηση του πρωτοκόλλου απαιτεί 128 KB μνήμης σε αντίθεση με ένα κόμβο sub-GHz με ιδιόκτητο πρωτόκολλο που απαιτεί 4 KB [05].

## **2.2 Πρωτόκολλα Επικοινωνίας για Ασύρματα Δίκτυα Αισθητήρων**

Οι ασύρματοι κόμβοι, όπως προαναφέρθηκε, έχουν περιορισμένες δυνατότητες σε υπολογιστική ισχύ και διαθέσιμη ενέργεια. Η λειτουργία τους γίνεται από μπαταρίες και τον περισσότερο χρόνο βρίσκονται σε κατάσταση αδράνειας ('ύπνου') ώστε να επιτευχθεί η ελάχιστη κατανάλωση ενέργειας. Μερικά από τα διαθέσιμα πρωτόκολλα για ΑΔΑ παρουσιάζονται παρακάτω [06].

### **2.2.1 Πρωτόκολλο LEACH (Low-Energy Adaptive Clustering Hierarchy)**

Στο πρωτόκολλο αυτό το δίκτυο αποτελείται από έναν κεντρικό κόμβο και πολλούς απομακρυσμένους. Οι απομακρυσμένοι κόμβοι ομαδοποιούνται σε μικρότερες ομάδες (clusters) και κάθε ομάδα έχει το δικό της κεντρικό κόμβο. Κάθε κόμβος της ομάδας στέλνει τα δεδομένα στον κεντρικό της κόμβο και οι κεντρικοί κόμβοι των ομάδων τα στέλνουν στον κεντρικό κόμβο του δικτύου. Οι κόμβοι κάθε ομάδας αναλαμβάνουν ρόλο κεντρικού κόμβου ομάδας και ο κεντρικός κόμβος κάθε ομάδας μοιράζει σε κάθε κόμβο της ομάδας ένα χρονοδιάγραμμα εκπομπής που είναι γνωστό σε όλους τους κόμβους της ομάδας. Όταν ένας κόμβος δεν εκπέμπει μπορεί να μεταβεί σε κατάσταση χαμηλής ενέργειας. Η εναλλαγή των κόμβων κάθε ομάδας σε κεντρικό κόμβο εξασφαλίζει την εξισορρόπηση της ενέργειας μέσα στην ομάδα. Επίσης η μετάβαση των κόμβων σε κατάσταση αδράνειας συμβάλλει στην εξοικονόμηση ενέργειας. Με τον διαμοιρασμό του χρονοδιαγράμματος στην ομάδα, αποφεύγονται οι συγκρούσεις των

πακέτων μέσα στην ομάδα αλλά δεν αποφεύγονται οι συγκρούσεις όταν οι κεντρικοί κόμβοι των ομάδων στέλνουν δεδομένα στον κεντρικό κόμβο του δικτύου [06].

### **2.2.2 Πρωτόκολλο Sensor-MAC (S-MAC)**

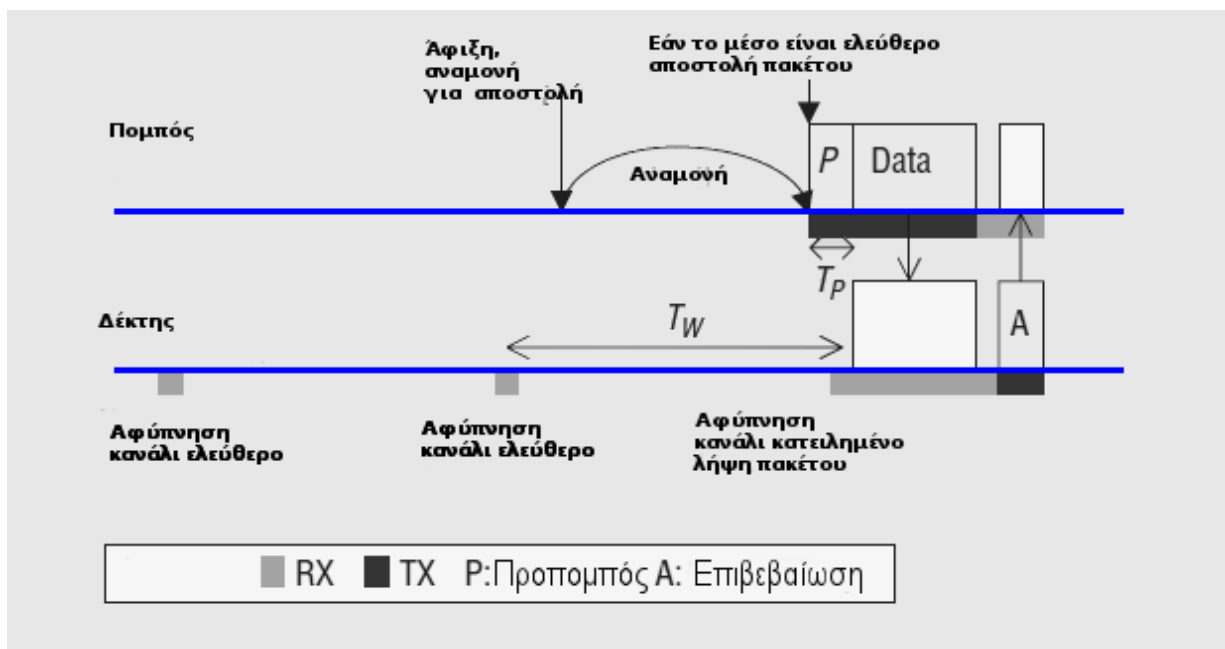
Στο πρωτόκολλο αυτό ο σημαντικότερος παράγοντας είναι η εξοικονόμηση ενέργειας. Κύριοι παράγοντες σπατάλης ενέργειας θεωρούνται: οι συγκρούσεις, η λήψη πακέτων που δεν αφορούν τον κόμβο, η ακρόαση του καναλιού σε κατάσταση αναμονής και η αποστολή πακέτων ελέγχου[07]. Οι κόμβοι μεταβαίνουν περιοδικά από κατάσταση ακρόασης σε ανενεργή κατάσταση κλείνοντας τον πομποδέκτη, επιτυγχάνοντας εξοικονόμηση ενέργειας 50%, και ενεργοποιούν ένα χρονόμετρο για να μεταβούν σε κατάσταση ακρόασης. Κάθε κόμβος μπορεί να επιλέξει ανεξάρτητα το δικό του χρονοδιάγραμμα. Γειτονικοί κόμβοι δημιουργούν εικονικές ομάδες και χρησιμοποιούν κοινό χρονοδιάγραμμα. Εάν ένας κόμβος ανήκει σε δύο ομάδες ακολουθεί το χρονοδιάγραμμα και των δύο ομάδων. Η ενημέρωση των χρονοδιαγραμμάτων μεταξύ των κόμβων γίνεται με αποστολή πακέτων SYNC. Η αποφυγή των συγκρούσεων γίνεται με ακρόαση του καναλιού. Κάθε φορά που ένας κόμβος λαμβάνει ένα πακέτο που δεν τον αφορά αυξάνει μια παράμετρο NAV (Network allocation Vector) και ενεργοποιεί ένα χρονόμετρο γι' αυτή. Τα πακέτα έχουν ένα πεδίο με τη διάρκεια του πακέτου και έτσι ο κόμβος γνωρίζει για πόσο χρόνο πρέπει να μείνει σε κατάσταση αναμονής. Όταν πυροδοτείται το χρονόμετρο, της παραμέτρου NAV, η παράμετρος μειώνεται. Η παράμετρος NAV αποτελεί την εικονική ακρόαση καναλιού. Το μέσο επικοινωνίας θεωρείται διαθέσιμο όταν η φυσική και η εικονική ακρόαση του καναλιού δείχνουν ότι είναι ελεύθερο. Για τη αποφυγή συγκρούσεων χρησιμοποιείται η αποστολή πακέτων RTS/CTS. Όταν οι γειτονικοί κόμβοι του πομπού και του δέκτη λάβουν RTS/CTS πηγαίνουν σε κατάσταση «ύπνου» για το χρόνο που χρειάζεται να μεταδοθεί το πακέτο. Μια σημαντική λειτουργία του S-MAC είναι η κατάτμηση μεγάλων μηνυμάτων σε μικρότερα και η αποστολή τους κατά ριπή. Περιοδική λειτουργία των κόμβων μπορεί να προκαλέσει μεγάλες καθυστερήσεις επειδή οι γειτονικοί κόμβοι έχουν το δικό τους χρονοδιάγραμμα.

Πλεονεκτήματα: η σπατάλη ενέργειας, από παθητική ακρόαση, μειώνεται με την εφαρμογή χρονοδιαγράμματος αφύπνισης.

Μειονεκτήματα: Τα πακέτα Broadcast δεν χρησιμοποιούν πακέτα CTS/RTS και έτσι αυξάνεται η πιθανότητα συγκρούσεων. Το χρονοδιάγραμμα αφύπνισης έχει σταθερούς και προκαθορισμένους χρόνους μειώνοντας έτσι την αποτελεσματικότητα του πρωτοκόλλου σε περίπτωση μεταβλητού φόρτου του δικτύου [06],[07].

### 2.2.3 Πρωτόκολλο Wise-MAC (Wireless Sensor-MAC)

Το πρωτόκολλο αυτό βασίζεται στη δειγματοληψία του προπομπού (preamble). Η τεχνική αυτή ελέγχει περιοδικά το μέσο μετάδοσης για δραστηριότητα. Όλοι οι κόμβοι στο δίκτυο «ακούν» το κανάλι για μικρό σταθερό χρονικό διάστημα βάση σχετικού ανεξάρτητου χρονοδιαγράμματος. Εάν το κανάλι είναι κατειλημμένο ο κόμβος συνεχίζει την ακρόαση μέχρι να λάβει δεδομένα ή να ελευθερωθεί το κανάλι [08]. Κάθε κόμβος γνωρίζει το χρονοδιάγραμμα των γειτόνων του και ενημερώνεται για αυτό από τα πακέτα επιβεβαίωσης. Βασιζόμενος στο χρονοδιάγραμμα του κόμβου λήπτη, ο αποστολέας προγραμματίζει την εκπομπή του ώστε η δειγματοληψία του καναλιού από τον δέκτη να αντιστοιχεί στο μέσο του προπομπού του αποστολέα [11].



Εικόνα 2.1 WiseMAC πρωτόκολλο [06]

Πλεονεκτήματα: Ο δυναμικός χειρισμός του μήκους του προπομπού αποδίδει καλύτερα σε συνθήκες μεταβαλλόμενου φορτίου. Το πρωτόκολλο χειρίζεται τις ολισθήσεις του ρολογιού και έτσι μετριάζεται ο εξωτερικός συγχρονισμός των ρολογιών των κόμβων.

Μειονεκτήματα: Βασικό μειονέκτημα είναι το κατανεμημένο χρονοδιάγραμμα εκπομπής και λήψης. Κατά την ευρυεκπομπή ενός πακέτου, τα πακέτα για γειτονικούς κόμβους που είναι αδρανείς, θα αποθηκευτούν και θα μεταδοθούν πολλές φορές σύμφωνα με το χρονοδιάγραμμα αφύπνισης των γειτονικών κόμβων. Η προαναφερθείσα κατάσταση συμβάλει στην αύξηση κατανάλωσης ενέργειας. Ένα άλλο μειονέκτημα είναι το πρόβλημα του κρυμμένου τερματικού που οδηγεί σε συγκρούσεις όταν ένας αποστολέας εκτός εμβέλειας στέλνει προπομπό σε ένα λήπτη που ήδη λαμβάνει από άλλο αποστολέα [06],[08],[09].

## 2.3 Συμπεράσματα

Η μεγάλη ανάπτυξη των ΑΔΑ έχει πυροδοτήσει το ερευνητικό ενδιαφέρον για πρότυπα τα οποία θα μπορούν να καλύψουν τις ιδιαιτερότητες των ασύρματων δικτύων αισθητήρων. Τα πρότυπα αυτά σχεδιάζονται με πρωταρχικό στόχο τη μείωση της κατανάλωσης της ενέργειας και την ανάγκη σε υπολογιστική ισχύ και μνήμη. Για την εξοικονόμηση ενέργειας χρησιμοποιούνται τεχνικές εναλλασσόμενης ενεργούς και ανενεργούς κατάστασης ή αλγόριθμοι δρομολόγησης ώστε οι κόμβοι να βρίσκονται σε ενεργή κατάσταση όσο το δυνατόν για μικρότερο διάστημα ή να χρησιμοποιούν εναλλακτικά μονοπάτια προς τον κεντρικό κόμβο, με την μικρότερη κατανάλωση ενέργειας.

Αν και υπάρχουν αρκετά νέα πρότυπα ωστόσο δεν υπάρχει ένα καθολικό πρότυπο για την ανάπτυξη ενός ασύρματου δικτύου αισθητήρων. Οι εφαρμογές των ΑΔΑ έχουν πολύ διαφορετικές ανάγκες από εφαρμογή σε εφαρμογή. Το κενό αυτό έρχονται να καλύψουν τα ιδιόκτητα πρότυπα τα οποία διαφοροποιούνται κατάλληλα για να καλύψουν τις ιδιαιτερότητες της κάθε εφαρμογής. Βασικός στόχος των ιδιόκτητων προτύπων είναι το χαμηλό κόστος και η απλότητα λειτουργίας τους. Άλλωστε πληθώρα εφαρμογών μπορούν να λειτουργήσουν με απλά πρωτόκολλα επικοινωνίας αποφεύγοντας την πολυπλοκότητα που εισάγουν τα πρότυπα.

# Κεφάλαιο 3

## Απλό πρωτόκολλο

### Επικοινωνίας – Λειτουργία

### Κόμβων

#### 3.1 Απλό Πρωτόκολλο Επικοινωνίας

Για τις ανάγκες της μεταπτυχιακής διατριβής σχεδιάστηκε ένα απλό πρωτόκολλο επικοινωνίας το οποίο στηρίχτηκε στο ιδιόκτητο πρωτόκολλο της Silicon Labs EZMacPRO [10]. Παραδοχές που έγιναν για την ανάπτυξη του πρωτοκόλλου είναι: .

1. οι ασύρματοι κόμβοι είναι σε τοπολογία αστέρα και επικοινωνούν με τον κεντρικό κόμβο
2. όλοι οι κόμβοι μπορούν να «ακούσουν» όλους τους άλλους κόμβους και έτσι δεν υπάρχει το πρόβλημα του κρυφού τερματικού,
3. οι κόμβοι εκπέμπουν τις μετρήσεις τους σε προκαθορισμένα χρονικά διαστήματα και περιμένουν από τον κεντρικό κόμβο το πακέτο επιβεβαίωσης ACK (acknowledge)

4. εάν δεν λάβουν το πακέτο ACK τότε θεωρούν ότι το πακέτο χάθηκε και μεταβαίνουν στην επόμενη κατάσταση λειτουργίας τους.
5. Ο κεντρικός κόμβος έχει απεριόριστη ενέργεια και είναι πάντα ενεργός.

## **3.2 Υλοποίηση Απλού Πρωτοκόλλου Επικοινωνίας για Ασύρματο Δίκτυο Αισθητήρων**

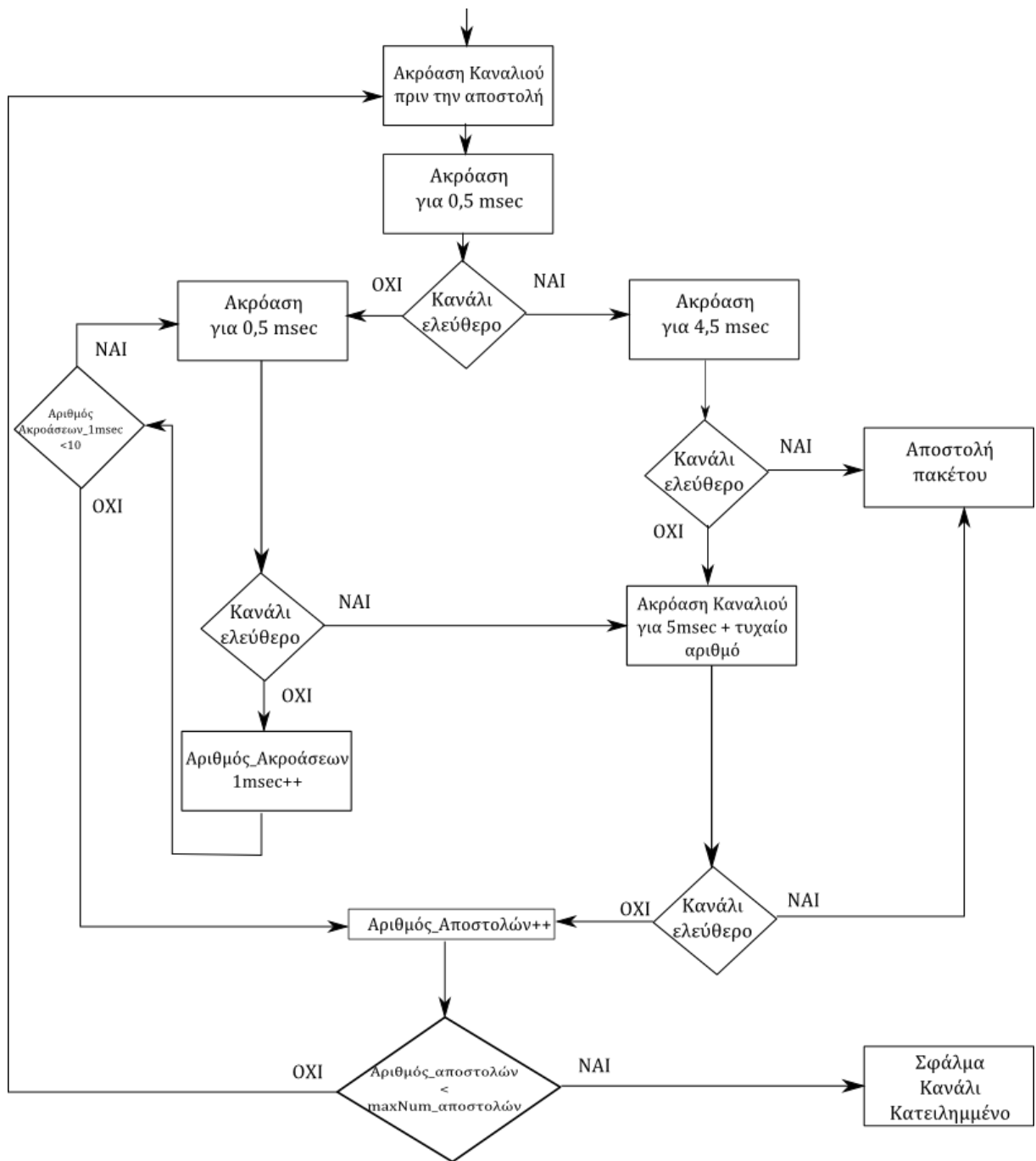
Όπως αναφέρθηκε υπάρχει έντονο ερευνητικό ενδιαφέρον για πρωτόκολλα επιπέδου πρόσβασης μέσου για ασύρματα δίκτυα αισθητήρων. Το πρωτόκολλο, της μεταπτυχιακής διατριβής, σχεδιάστηκε ώστε η κατανάλωση ενέργειας να είναι ελάχιστη. Οι κόμβοι λειτουργούν περιοδικά και στέλνουν τα δεδομένα τους στο κεντρικό κόμβο. Μετά την αποστολή των δεδομένων περιμένουν για ένα μικρό χρονικό διάστημα το πακέτο επιβεβαίωσης (ACK) από τον κεντρικό κόμβο. Αν δεν λάβουν το πακέτο ACK τότε θεωρούν ότι το πακέτο χάθηκε και μεταβαίνουν σε κατάσταση κατανάλωσης χαμηλής ενέργειας ('ύπνου') μέχρι τον επόμενο κύκλο.

### **3.2.1 Πρωτόκολλο LBT (Listen Before Talk)**

Όταν ένας κόμβος θέλει να στείλει ένα πακέτο στον κεντρικό κόμβο τότε ακολουθεί το παρακάτω πλάνο για να ελέγξει εάν το κανάλι είναι ελεύθερο ή όχι Εικόνα 3.1.

Ο έλεγχος του καναλιού γίνεται από τον πομποδέκτη ο οποίος έχει τη δυνατότητα να μετρά την ισχύ του λαμβανόμενου σήματος. Στον καταχωρητή 0x27 του πομποδέκτη, RSSI Threshold for CCI (Receive Signal Strength Indicator Threshold for Clear Channel Indicator), ορίζεται ένα κατώφλι για το λαμβανόμενο σήμα κάτω από το οποίο θεωρείται ότι το κανάλι επικοινωνίας είναι ελεύθερο. Εάν το σήμα που λαμβάνει ο δέκτης είναι μεγαλύτερο από το κατώφλι RSSI τότε ο δέκτης ενεργοποιεί μια διακοπή στον μικροελεγκτή και εκτελείται η ρουτίνα διακοπής [11].





**Εικόνα 3.1** Πρωτόκολλο MAC – Διαδικασία Listen Before Talk

Η παραπάνω διαδικασία αναφέρεται ως LBT (Listen Before Talk) και αποτρέπει την αποστολή πακέτου εάν το κανάλι είναι κατειλημμένο [11]. Η διαδικασία LBT στην Εικόνα 3.1 αναπτύχθηκε σαν μηχανή καταστάσεων και περιγράφεται παρακάτω.

1. Ο κόμβος πηγαίνει σε κατάσταση λήψης και ελέγχει αν το κανάλι επικοινωνίας είναι ελεύθερο. Η ακρόαση του καναλιού διαρκεί 0,5 msec και εάν το κανάλι είναι ελεύθερο ακολουθεί ακρόαση του καναλιού για άλλα 4,5 msec. Εάν το κανάλι είναι ελεύθερο για το

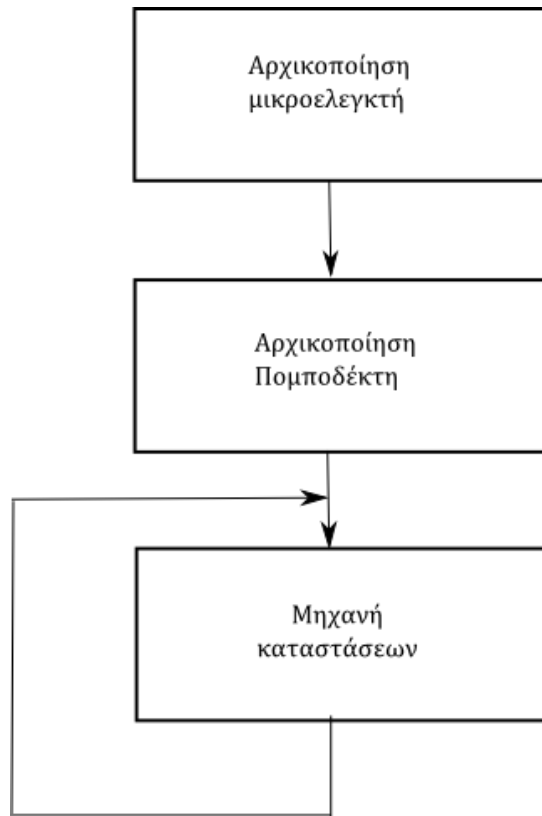
σύνολο των 5 msec , η ισχύς του σήματος δεν υπερβαίνει το κατώφλι RSSI, τότε ο κόμβος εκπέμπει το πακέτο. Εάν το κανάλι δεν είναι ελεύθερο για τα τελευταία 4,5 msec τότε η μηχανή καταστάσεων μεταβαίνει στο βήμα 3.

2. Εάν το κανάλι θεωρηθεί κατειλημμένο στα πρώτα 0,5 msec τότε γίνεται δειγματοληψία της ισχύος του λαμβανόμενου σήματος κάθε 1 msec για 10 msec. Εάν εκπνεύσει ο χρόνος των 10 msec τότε η μηχανή καταστάσεων πηγαίνει στο βήμα 5.
3. Εάν το λαμβανόμενο σήμα είναι χαμηλότερο από το κατώφλι RSSI τότε η μηχανή καταστάσεων παραμένει σε κατάσταση λήψης (ακρόασης) για άλλα 5 + τυχαίο αριθμό msec. Το εύρος της γεννήτριας τυχαίων αριθμών είναι παραμετροποιήσιμο και μπορεί να αλλάξει ανάλογα με τις απαιτήσεις της εφαρμογής.
4. Εάν η ισχύς του λαμβανόμενου σήματος παραμένει κάτω από το κατώφλι, για το διάστημα των 5 + τυχαίος αριθμός msec, τότε ο κόμβος αποστέλλει το πακέτο. Σε αντίθετη περίπτωση η μηχανή καταστάσεων πηγαίνει στο βήμα 5.
5. Εάν ο αριθμός αποστολών είναι μικρότερος από τη σταθερά maxNum\_αποστολών τότε ο σταθμός επαναλαμβάνει τη διαδικασία από το βήμα 1. Εάν ο αριθμός αποστολών είναι μικρότερος τότε η μηχανή καταστάσεων πηγαίνει στο βήμα 6.
6. Εάν ο κόμβος δεν μπορέσει να στείλει το πακέτο, επειδή η συχνότητα ήταν κατειλημμένη, τότε η μηχανή καταστάσεων επιστρέφει κατάσταση λάθους και ο κόμβος μεταβαίνει στην επόμενη κατάσταση.

### 3.3 Λειτουργία του κόμβου

Οι ασύρματοι κόμβοι λειτουργούν σαν μια μηχανή καταστάσεων και έχουν το ίδιο λογισμικό. Κάθε κόμβος χαρακτηρίζεται από μια σταθερά που ορίζει αν ο κόμβος είναι ένας απλός κόμβος (NODE) ή αν είναι ο κεντρικός κόμβος (MASTER). Η λειτουργία των κόμβων γίνεται από μπαταρίες και τον περισσότερο χρόνο βρίσκονται σε κατάσταση χαμηλής κατανάλωσης ενέργειας. Ο κεντρικός κόμβος είναι πάντα ενεργός. Σε κάθε κόμβο NODE, μια σταθερά, ορίζει κάθε πότε θα στέλνει ο κόμβος δεδομένα στον κεντρικό κόμβο. Η διαδικασία εκκίνησης των κόμβων ακολουθεί το διάγραμμα που φαίνεται στην Εικόνα 3.2.

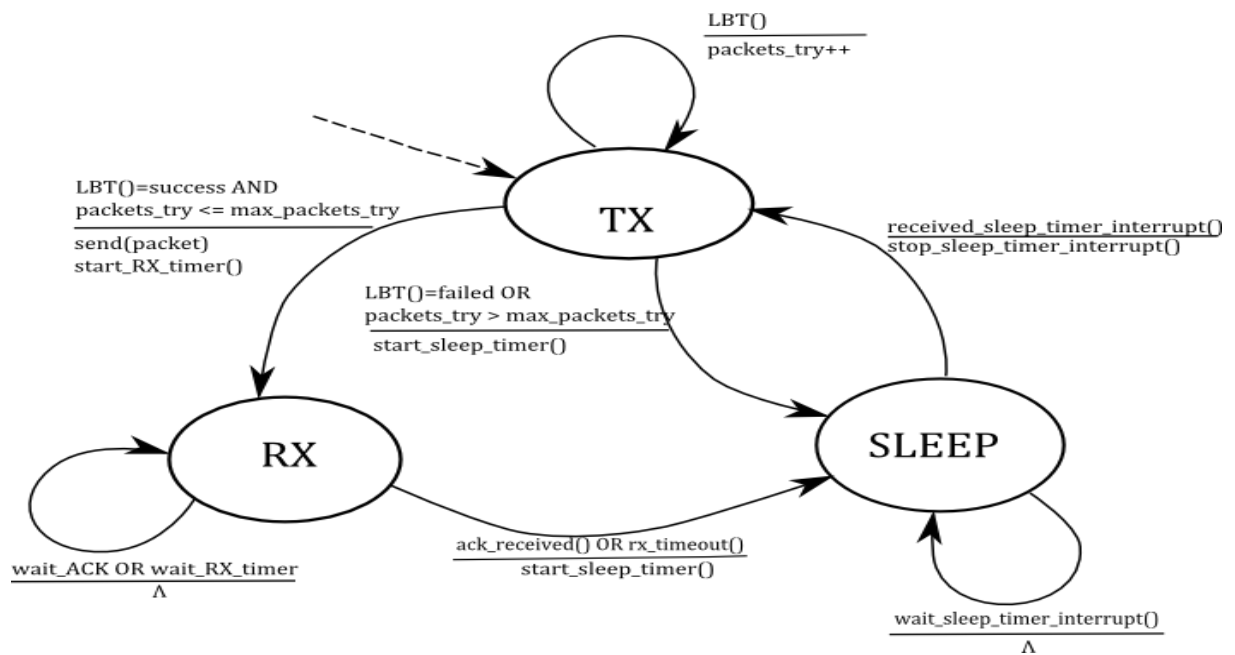
Όταν ξεκινά ο κόμβος, αρχικοποιείται ο μικροελεγκτής, κατόπιν θέτει τις αρχικές τιμές στο ολοκληρωμένο του πομποδέκτη και τέλος μπαίνει σε έναν ατέρμονα βρόγχο όπου υλοποιείται μια μηχανή καταστάσεων για τη λειτουργία του κόμβου.



Εικόνα 3.2 Λειτουργία των κόμβων

### 3.3.1 Καταστάσεις λειτουργίας απομακρυσμένου κόμβου (NODE)

Οι καταστάσεις στις οποίες μπορεί να βρεθεί ένας κόμβος, τύπου NODE, είναι κατάσταση χαμηλής κατανάλωσης ενέργειας (SLEEP), κατάσταση μετάδοσης δεδομένων (TX) και κατάσταση λήψης (RX). Οι παραπάνω λειτουργίες υλοποιούνται από τη μηχανή καταστάσεων που φαίνεται στην Εικόνα 3.3 [12].



Εικόνα 3.3 Μηχανή καταστάσεων απομακρυσμένου κόμβου

Εάν ο κόμβος βρει το κανάλι ελεύθερο και η μεταβλητή `packets_try` (αριθμός επανεκπομπών) δεν ξεπερνά τον μέγιστο αριθμό επανεκπομπών τότε ο κόμβος εκπέμπει το πακέτο, ξεκινά τον μετρητή αναμονής πακέτου επιβεβαίωσης και μεταβαίνει στην κατάσταση RX. Στην κατάσταση RX περιμένει για χρόνο ίσο με 200 msec ή μέχρι να λάβει το πακέτο ACK από τον κύριο κόμβο. Αν λάβει το πακέτο ACK τότε μεταβαίνει σε κατάσταση SLEEP και περιμένει να λήξει το χρονόμετρο ώστε να ξεκινήσει ο επόμενος κύκλος εκπομπής. Εάν δεν λάβει το πακέτο ACK μέσα σε 200msec τότε θεωρεί ότι το πακέτο χάθηκε και μεταβαίνει σε κατάσταση SLEEP. Από την κατάσταση SLEEP μεταβαίνει στην κατάσταση εκπομπής (TX), όταν λήξει το χρονόμετρο, οπότε και ξεκινά ένας επόμενος κύκλος. Από την κατάσταση TX ο κόμβος μπορεί να βρεθεί στην κατάσταση SLEEP εάν δεν μπορέσει να βρει το κανάλι ελεύθερο ή εάν έχει ξεπεραστεί ο μέγιστος αριθμός επανεκπομπών του πακέτου.

### Κατάσταση SLEEP

Όταν ο κόμβος βρίσκεται σε κατάσταση SLEEP τότε ο μικροελεγκτής και ο πομποδέκτης βρίσκονται σε κατάσταση χαμηλής κατανάλωσης ενέργειας («ύπνου»). Στον πομποδέκτη ορίζεται ο χρόνος παραμονής του συστήματος σε κατάσταση «ύπνου». Για το σκοπό αυτό χρησιμοποιείται η δυνατότητα του πομποδέκτη να έχει

ενεργό το χρονόμετρο ενώ βρίσκεται σε αυτή την κατάσταση. Το χρονόμετρο αφύπνισης χρησιμοποιεί, για το χρονισμό του, εσωτερικό RC ταλαντωτή 32,768KHz. Για τον ορισμό του χρόνου αφύπνισης χρησιμοποιούνται οι καταχωρητές 14h-16h του πομποδέκτη. Κατά την εκπνοή του χρόνου αφύπνισης ενεργοποιείται διακοπή που εμφανίζεται στον ακροδέκτη του πομποδέκτη nIRQ. Ο μικροελεγκτής λαμβάνει το σήμα διακοπής, ελέγχει από του καταχωρητές 03h-04h το είδος της διακοπής και ενεργεί αντίστοιχα. Για τον ορισμό του χρόνου αφύπνισης (WUT- Wake Up Time) χρησιμοποιείται η παρακάτω εξίσωση:

$$WUT = \frac{4 * M * 2^R}{32,768} msec$$

όπου R: wtr[4:0] και M: wtm[15:0]. Η μεταβλητή “wtr” είναι ο καταχωρητής 14h και η μεταβλητή “wtm” είναι οι καταχωρητές 15h και 16h. Με βάση την παραπάνω εξίσωση μπορεί να οριστεί χρόνος αφύπνισης από 0-97 ημέρες με βήμα που εξαρτάται από τις τιμές των M και R. Η μέγιστη τιμή του R είναι 0x14 (20<sub>10</sub>) [13].

### **Κατάσταση TX**

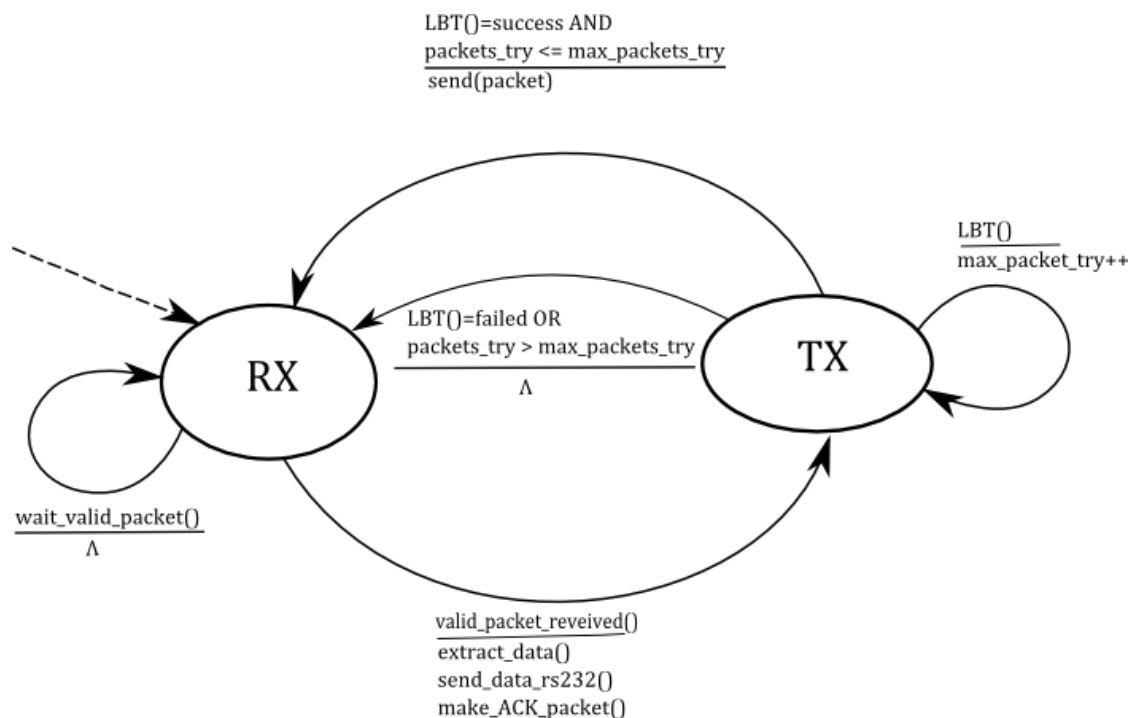
Όταν ο κόμβος είναι στην κατάσταση TX τότε έχει ετοιμάσει το πακέτο δεδομένων προς αποστολή και αφού εκτελέσει επιτυχώς τη διαδικασία LBT, τότε αποστέλλει το πακέτο και μεταβαίνει στην κατάσταση RX. Εάν δεν είναι επιτυχής η διαδικασία LBT τότε ο πομποδέκτης μπορεί να επαναλάβει τη διαδικασία αποστολής του ίδιου πακέτου χωρίς να το ξαναδημιουργήσει επειδή το πακέτο προς αποστολή είναι έτοιμο από την προηγούμενη προσπάθειά του. Στην εφαρμογή, για λόγους απλότητας, επιλέχθηκε να μεταβαίνει σε κατάσταση SLEEP μετά την πρώτη προσπάθεια και να θεωρεί ότι το πακέτο χάθηκε. Κατά τη διαδικασία LBT ο κόμβος προσπαθεί να στείλει το πακέτο μέχρι η μεταβλητή max\_packet\_try να είναι ίση με ένα μέγιστο αριθμό αποστολών που ορίζεται σε μεταβλητή του κόμβου.

## Κατάσταση RX

Μετά την επιτυχημένη αποστολή ενός πακέτου ο κόμβος μεταβαίνει στην κατάσταση RX περιμένοντας το πακέτο επιβεβαίωσης. Σε αυτή την κατάσταση παραμένει για το πολύ 200msec. Εάν δεν λάβει το πακέτο επιβεβαίωσης σε αυτό το χρονικό διάστημα πηγαίνει στην κατάσταση SLEEP και θεωρεί ότι το πακέτο χάθηκε.

### 3.3.2 Καταστάσεις λειτουργίας κεντρικού κόμβου (MASTER)

Μετά την εκκίνηση του κεντρικού κόμβου, ο κόμβος μεταβαίνει σε κατάσταση RX και περιμένει να λάβει πακέτα. Μόλις ό κόμβος λάβει ένα πακέτο το οποίο είναι έγκυρο, τότε μεταβαίνει σε κατάσταση TX και ακολουθεί τη διαδικασία που ακολουθούν οι απομακρυσμένοι κόμβοι ώστε να στείλει το πακέτο επιβεβαίωσης. Αμέσως μετά ο κόμβος μεταβαίνει σε κατάσταση RX περιμένοντας για το επόμενο πακέτο. Όταν λάβει ένα έγκυρο πακέτο, ο κεντρικός κόμβος, το στέλνει στην σειριακή θύρα του υπολογιστή ώστε να εισαχθούν οι μετρήσεις στη βάση δεδομένων. Η παραπάνω διαδικασία περιγράφεται από τη μηχανή καταστάσεων της Εικόνα 3.4.



Εικόνα 3.4 Μηχανή καταστάσεων κεντρικού κόμβου (MASTER)

## Κατάσταση RX

Ο κεντρικός κόμβος είναι πάντα ενεργός και βρίσκεται στη κατάσταση RX τον περισσότερο χρόνο της λειτουργίας του. Σε αυτή την κατάσταση περιμένει να λάβει πακέτα από τους απομακρυσμένους κόμβους. Όταν λάβει ένα έγκυρο πακέτο, εξάγει από το πακέτο τα δεδομένα, στέλνει τα δεδομένα στη σειριακή θύρα του Η/Υ και ετοιμάζει το πακέτο επιβεβαίωσης. Αμέσως μετά μεταβαίνει στην κατάσταση αποστολής και ακολουθεί τη διαδικασία LBT για να στείλει στον απομακρυσμένο κόμβο το πακέτο επιβεβαίωσης. Εάν η διαδικασία LBT είναι επιτυχής και δεν έχει ξεπεραστεί ο μέγιστος αριθμός αποστολών τότε εκπέμπει το πακέτο ACK και επιστρέφει στην κατάσταση RX. Εάν δεν καταφέρει να βρει ελεύθερο το κανάλι ή έχει ξεπεράσει το μέγιστο αριθμό επανεκπομπών τότε επιστρέφει στη κατάσταση RX. Ο κεντρικός κόμβος το μεγαλύτερο χρονικό διάστημα είναι σε κατάσταση RX.

## Κατάσταση TX

Ο κεντρικός κόμβος μεταβαίνει σε αυτή την κατάσταση μόνο για πολύ μικρό χρονικό διάστημα, ώστε να εκπέμψει τα πακέτα επιβεβαίωσης, και αμέσως επιστρέφει σε κατάσταση RX.

## 3.4 FIFO και PACKET HANDLER modes

Η γενική δομή ενός πακέτου που μπορούν να χρησιμοποιήσουν οι ασύρματοι κόμβοι για να επικοινωνήσουν μεταξύ τους έχει την παρακάτω μορφή Εικόνα 3.5.



Εικόνα 3.5 Γενική Δομή Πακέτου

Κάθε κόμβος, απομακρυσμένος ή κεντρικός, λειτουργεί σε κατάσταση FIFO χρησιμοποιώντας την αυτόματη λειτουργία PH(automatic Packet Handler) [14]. Σ' αυτή την κατάσταση τα δεδομένα προς αποστολή ή λήψη βρίσκονται στη μνήμη FIFO και διαβάζονται ή γράφονται από τον μικροελεγκτή μέσω της θύρας SPI του πομποδέκτη. Τα επιπλέον bytes που προσθέτει η λειτουργία PH ενσωματώνονται και αφαιρούνται από τα πακέτα αυτόματα από τη λειτουργία PH.

Η κατάσταση λειτουργίας του πομποδέκτη προσδιορίζεται από τιμές των bits των δύο καταχωρητών 07h και 08h (Operating Mode and Function Control 1 και 2 αντίστοιχα) Εικόνα 3.6. Ενεργοποιώντας τα bits των δύο καταχωρητών ελέγχουμε τη λειτουργία του πομποδέκτη. Περιγραφή των λειτουργιών των δύο καταχωρητών δίνεται στον Πίνακα 3.1 και Πίνακα 3.2 [13].

Register	D7	D6	D5	D4	D3	D2	D1	D0
<b>07h</b> Operating Mode and Function Control 1	swres	enlbd	enwt	x32ksel	txon	rxon	pllon	xton
<b>08h</b> Operating Mode and Function Control 2	antdiv[2:0]			rxmpk	autotx	enldm	ffclrx	ffcltx

**Εικόνα 3.6** Καταχωρητές τρόπου λειτουργίας και ελέγχου [13]

bit	Ονομασία	Περιγραφή
7	swres	Software Reset Bit  Αυτό το bit μπορεί να χρησιμοποιηθεί ώστε να τοποθετήσουμε σε όλους τους καταχωρητές προκαθορισμένες τιμές χωρίς να χρειάζεται να γίνεται εγγραφή σε έναν προς έναν. Η επαναφορά γίνεται με τον καθορισμό swres =1. Το bit αυτό διαγράφεται αυτόματα.
6	enlbd	Enable Low Battery Detect.



bit	Ονομασία	Περιγραφή
		Όταν αυτό το bit είναι 1, το κύκλωμα ανίχνευσης χαμηλής τάσης μπαταρίας και η σύγκριση με το κατώφλι θα ενεργοποιηθεί.
5	enwt	Enable Wake-Up-Timer.  Ενεργοποιείται όταν enwt = 1. Με την ενεργοποίηση του χρονόμετρου αφύπνισης το ολοκληρωμένο στέλνει μια διακοπή, με την εκπνοή του χρόνου, στον ακροδέκτη GPIO σε οποιονδήποτε τρόπο λειτουργίας και αν βρίσκεται.
4	x32ksel	32,768 kHz Crystal Oscillator Select.  0: RC ταλαντωτής  1: 32 kHz κρύσταλλος
3	txon	TX on in Manual Transmit Mode.  Αυτόματος καθαρισμός (bit=0) σε FIFO λειτουργία αμέσως μετά την αποστολή του πακέτου..
2	rxon	RX on in Manual Receiver Mode.  Αυτόματος καθαρισμός (rxon=0) όταν ληφθεί ένα έγκυρο πακέτο και είναι απενεργοποιημένη η λειτουργία Multiple Packets
1	pllon	TUNE Mode (PLL is ON).  Όταν pllon = 1, το PLL θα παραμείνει ενεργοποιημένο και στην κατάσταση ηρεμίας. Αυτό επιτρέπει ταχύτερους χρόνους επιστροφής σε ενεργή κατάσταση με κόστος την αυξημένη κατανάλωσης ενέργειας στην κατάσταση ηρεμίας.
0	xton	READY Mode (Xtal is ON).

**Πίνακας 3.1** Καταχωρητής 07h- Τρόπος Λειτουργίας και ελέγχου 1 [13]

bit	Ονομασία	Περιγραφή
7:5	antdiv[2:0]	Software Reset Bit Το GPIO πρέπει να παραμετροποιηθεί για διαφορική λήψη ώστε να λειτουργήσει σωστά ο αλγόριθμος.
4	rxmpk	RX Multi Packet. Εάν έχει επιλεγεί FIFO λειτουργία και RX-Packet Handling τότε η μνήμη FIFO θα γεμίσει με πολλαπλά πακέτα εάν rxmpk=1 διαφορετικά ο πομποδέκτης θα φύγει αυτόματα από την κατάσταση λήψης μόλις λάβει ένα έγκυρο πακέτο.
3	autotx	Automatic Transmission. Όταν autotx = 1, ο πομποδέκτης θα τεθεί αυτόματα σε λειτουργία μετάδοσης (TX), όταν η μνήμη FIFO είναι σχεδόν πλήρης. Όταν η μνήμη FIFO είναι άδεια, θα επιστρέψει αυτόματα στην κατάσταση αναμονής.
2	enldm	Enable Low Duty Cycle Mode. Αν αυτό το bit είναι 1 τότε ο πομποδέκτης πηγαίνει σε κατάσταση λήψης (RX) τακτικά. Η συχνότητα πρέπει να καθοριστεί στον καταχωρητή Timer Wake-Up Period, ενώ ο ελάχιστος χρόνος ON θα πρέπει να καθοριστεί στον καταχωρητή Low-Duty Mode Duration. Η λειτουργία FIFO θα πρέπει να ενεργοποιηθεί.
1	ffclrx	RX FIFO Reset/Clear. Για τον σβήσιμο της μνήμης RX-FIFO πρέπει να γίνουν δύο συνεχόμενες εγγραφές: ffclrx = 1 ακολουθούμενη από ffclrx = 0.
0	ffcltx	TX FIFO Reset/Clear. Για τον σβήσιμο της μνήμης TX-FIFO πρέπει να γίνουν δύο

bit	Ονομασία	Περιγραφή
		συνεχόμενες εγγραφές: ffclrtx = 1 ακολουθούμενη από ffclrtx = 0.

**Πίνακας 3.2** Καταχωρητής 08h- Τρόπος Λειτουργίας και ελέγχου 2 [13]

### 3.4.1 Λειτουργία FIFO σε κατάσταση μετάδοσης (TX)

Στην κατάσταση αυτή τα δεδομένα προς αποστολή γράφονται στην μνήμη FIFO byte προς byte μέσω της θύρας SPI. Εάν δεν έχει ενεργοποιηθεί η λειτουργία αυτόματου χειρισμού πακέτου (PH: packet Handler) τότε τα δεδομένα που βρίσκονται μόνο στη μνήμη FIFO μεταδίδονται. Αυτό δίνει τη δυνατότητα στον μικροελεγκτή να έχει πλήρη έλεγχο στη δομή του πακέτου. Η μνήμη FIFO μπορεί να προσπελαστεί σε οποιαδήποτε κατάσταση λειτουργίας του πομποδέκτη, ακόμη και όταν είναι σε κατάσταση SLEEP. Το ολοκληρωμένο λειτουργεί σε FIFO ορίζοντας στον καταχωρητή 71h τα bits dtmod[1:0] ίσον με  $(10)_2$  [13]. Ο πομποδέκτης μπορεί να παραμετροποιηθεί ώστε να ενεργοποιεί μια διακοπή στον ακροδέκτη nIRQ όταν γίνουν ίσον με ένα τα bits ipksent και enpk sent των καταχωρητών 03h και 05h αντίστοιχα Εικόνα 3.7. Όταν ενεργοποιηθεί η διακοπή το ολοκληρωμένο εξέρχεται της κατάστασης TX και πηγαίνει στην κατάσταση που ορίζεται από τα υπόλοιπα bits του καταχωρητή 07h [14].

Register	D7	D6	D5	D4	D3	D2	D1	D0
<b>71h</b> Modulation mode Control 2	trclk[1:0]		dtmod[1:0]		eninv	fd8	modtyp[1:0]	
<b>03h</b> Interrupt/Status 1	ifferr	itxffaful	itxffaem	irxffaful	iext	ipksent	ipkvalid	icrerror
<b>05h</b> Interrupt Enable 1	enferr	entxffaful	entxffaem	enrffaful	enext	enpk sent	enpkvalid	enrerror

**Εικόνα 3.7** Καταχωρητές –Λειτουργία FIFO, διακοπή ‘αποστολή πακέτου’

Εάν είναι ενεργοποιημένη η λειτουργία PH, τότε ο αριθμός των bytes που θα διαβαστούν από τη μνήμη FIFO ώστε να συγκροτηθεί το πακέτο προς αποστολή, πρέπει να είναι γραμμένα στον καταχωρητή 3Eh Εικόνα 3.8 [13].

Register	D7	D6	D5	D4	D3	D2	D1	D0
3Eh Packet Length Register	pklen[7:0]							

**Εικόνα 3.8** Καταχωρητής 3Eh, μέγεθος πακέτου

Εάν η λειτουργία PH δεν είναι ενεργοποιημένη, τότε τα bytes που βρίσκονται στη μνήμη FIFO θεωρούνται δεδομένα προς αποστολή. Έτσι τα δεδομένα διαβάζονται από τη μνήμη FIFO μέχρι αυτή να αδειάσει. Αν κατά τη διάρκεια της αποστολής ο μικροελεγκτής συνεχίζει να γεμίζει τη μνήμη FIFO με νέα δεδομένα, τότε θεωρητικά είναι δυνατόν να δημιουργηθεί πακέτο απείρου μήκους. Μετά τη μετάδοση του τελευταίου byte από τη FIFO ο πομποδέκτης δημιουργεί τη διακοπή ipksent και θεωρείται ότι το πακέτο ολοκληρώθηκε. Επιπλέον η μνήμη TX FIFO έχει δύο όρια , άνω και κάτω, τα οποία ορίζονται στους καταχωρητές 7Ch (σχεδόν γεμάτη FIFO) και 7Dh (σχεδόν άδεια FIFO). Αυτά τα όρια μπορούν να χρησιμοποιηθούν ώστε να δημιουργήσουν αντίστοιχες διακοπές όταν το περιεχόμενο της μνήμης FIFO φτάσει τα όρια αυτά (στη μεταπτυχιακή διατριβή δεν χρησιμοποιήθηκαν τα όρια αυτά). Για τη διαγραφή της μνήμης TX-FIFO πρέπει να γίνουν δυο έγγραφές του bit fflrtx στον καταχωρητή 07h. Αυτό το bit δεν μηδενίζεται αυτόματα, επομένως μια έγγραφη του 1 και μια του 0, χρειάζεται ώστε να διαγραφεί η μνήμη.

### 3.4.2 Λειτουργία FIFO σε κατάσταση λήψης (RX)

Όταν το ολοκληρωμένο λειτουργεί σε κατάσταση FIFO τότε τα δεδομένα που ελήφθησαν, διαβάζονται από την μνήμη RX-FIFO. Η διαδικασία ανάγνωσης μπορεί να γίνει και σε κατάσταση λειτουργίας SLEEP.

Στην λειτουργία RX-FIFO μόνο τα bytes που θεωρούνται δεδομένα καταχωρούνται στη μνήμη FIFO [14]. Εάν είναι ενεργοποιημένη και η λειτουργία PH τότε η λειτουργία PH φροντίζει να προσδιορίσει τα bytes των δεδομένων από το πακέτο που ελήφθη. Τα bytes TX Header και CRC απομακρύνονται από το πακέτο και μόνο τα bytes που θεωρούνται δεδομένα αποθηκεύονται στη μνήμη RX-FIFO. Στη μνήμη RX-FIFO δεν αποθηκεύονται ποτέ τα bytes TX Header και CRC. Εάν δεν είναι ενεργοποιημένη η λειτουργία PH τότε τα bytes μετά τη λέξη συγχρονισμού (Sync Word) θεωρούνται δεδομένα και αποθηκεύονται στη μνήμη RX-FIFO. Έτσι πρέπει να προγραμματιστούν, το κατώφλι ανίχνευσης προπομπού καθώς και η λέξη συγχρονισμού ώστε ο δέκτης να γνωρίζει από πού πρέπει να αρχίσει να αποθηκεύει τα bytes των δεδομένων στην RX-FIFO μνήμη, ακόμη και αν δεν έχει ενεργοποιηθεί η λειτουργία PH.

Ο δέκτης μπορεί να προγραμματιστεί ώστε να δημιουργεί μια διακοπή όταν λάβει ένα έγκυρο πακέτο δεδομένων. Η κατάσταση της διακοπής αυτής εμφανίζεται στο bit ipkvalid (D1) του καταχωρητή 03h. Για να ενεργοποιηθεί η διακοπή στον ακροδέκτη nIRQ, πρέπει να γίνει 1 το bit enpkvalid (D1) του καταχωρητή 05h. Εικόνα 3.9. Η διακοπή αυτή είναι διαθέσιμη μόνο όταν η λειτουργία PH ενεργή. Όταν η λειτουργία PH είναι απενεργοποιημένη, ο δέκτης δεν έχει καμία γνώση για το μέγεθος του λαμβανόμενου πακέτου και δεν μπορεί να δημιουργήσει τη διακοπή.

Register	D7	D6	D5	D4	D3	D2	D1	D0
<b>03h</b> Interrupt/Status 1	ifferr	itxffafull	itxffaem	irxffafull	iext	ipksent	ipkvalid	icrerror
<b>05h</b> Interrupt Enable 1	enferr	entxffafull	entxffaem	enrxffafull	enext	enpksent	enpkvalid	enccrerror

**Εικόνα 3.9** Καταχωρητές 03h και 05h - διακοπή 'έγκυρο πακέτο'

Μετά την λήψη ενός έγκυρου πακέτου ο δέκτης εξέρχεται από την κατάσταση RX, μετά τη λήψη της διακοπής ipkvalid, και μεταβαίνει σε μια κατάσταση που προσδιορίζεται από τα υπόλοιπα bits του καταχωρητή 07h.

Όταν σε τρόπο λειτουργίας RX είναι ενεργοποιημένη η δυνατότητα PH όλα τα πεδία της δομής του πακέτου πρέπει να ορισθούν. Η λειτουργικότητα PH επεξεργάζεται και αφαιρεί τα επιπλέον πεδία του πακέτου και αποθηκεύει μόνο τα bytes δεδομένων στη μνήμη RX-FIFO. Μπορεί να παραμετροποιηθεί να χειρίζεται πακέτα σταθερού ή μεταβλητού μήκους με Header ή όχι και CRC ή όχι. Εάν είναι ενεργοποιημένη η λειτουργία PH, ο δέκτης μπορεί αυτόματα να καθορίσει το τέλος του πακέτου επειδή έχει γνώση του αναμενόμενου μήκους. Με τη λήψη ενός έγκυρου πακέτου ο δέκτης μεταπηδά σε κατάσταση που ορίζεται από τη μηχανή καταστάσεων. Η δομή ενός πακέτου με ενεργοποιημένη τη λειτουργία PH φαίνεται στην Εικόνα 3.10.



**Εικόνα 3.10** Δομή πακέτου με ενεργοποιημένη τη λειτουργία PH

Εάν η λειτουργία PH είναι απενεργοποιημένη τα πεδία Preamble και SYNC είναι απαραίτητα στο πακέτο προς λήψη. Όλα τα bytes που ακολουθούν το πεδίο SYNC θεωρούνται bytes δεδομένων και μεταφέρονται στη μνήμη RX-FIFO. Με αυτό τον τρόπο μπορούν να δημιουργηθούν πακέτα με διαφορετικά πεδία και κώδικες ελέγχου από τον μικροελεγκτή. Αυτό αυξάνει τις ανάγκες για υπολογιστική ισχύ στον μικροελεγκτή και επιπλέον οι έλεγχοι πρέπει να γίνονται από τον ίδιο, καθώς ο δέκτης δεν έχει καμία πληροφορία για τη δομή του πακέτου. Σ' αυτόν τον τρόπο λειτουργίας ο δέκτης παραμένει στην κατάσταση RX μέχρι ο μικροελεγκτής του δώσει εντολή να μεταβεί σε άλλη κατάσταση.



**Εικόνα 3.11** Δομή πακέτου με απενεργοποιημένη τη λειτουργία PH

Η διαγραφή της μνήμης RX-FIFO γίνεται με δύο συνεχόμενες εγγραφές του bit `ffclrrx` (D1) στον καταχωρητή 08h πρώτα θέτοντας `ffclrrx=1` και μετά `ffclrrx=0`.

### 3.4.3 Λειτουργία PH (Automatic Packet Handler)

Ο σκοπός της λειτουργίας PH είναι η αυτόματη κατασκευή της βασικής δομής ενός πακέτου σε λειτουργία TX και αποδόμηση του πακέτου σε λειτουργία RX χωρίς τη μεσολάβηση του μικροελεγκτή [14]. Τα βασικά πεδία ενός πακέτου (`preamble`, `Sync Word`, `TX Header`, `Packet Length` και `CRC`) δεν αλλάζουν συχνά και μπορούν να αποθηκευτούν σε καταχωρητές. Προσθέτοντας αυτόματα αυτά τα πεδία στο πακέτο μειώνεται η υπολογιστική ισχύς που χρειάζεται ο μικροελεγκτής, επιτρέποντας έτσι τη χρήση φθηνότερων μικροελεγκτών.

Η λειτουργία PH μπορεί να ενεργοποιηθεί ή να απενεργοποιηθεί και στις δύο λειτουργικές καταστάσεις του πομποδέκτη RX ή TX. Τα δύο bit που ελέγχουν τη λειτουργία PH για τον πομπό ή τον δέκτη είναι `enpactx` (D3) και `enpacrx` (D7) του καταχωρητή 30h αντίστοιχα Εικόνα 3.12.

Register	D7	D6	D5	D4	D3	D2	D1	D0
30h Data Access control	enpacrx	lsbfrst	credonly	skip2ph	enpactx	encrc	crc[1:0]	

Εικόνα 3.12 Καταχωρητής 30h, έλεγχος του Packet Handler

Οι λειτουργικότητες που προσθέτει η ενεργοποίηση του αυτόματου χειριστή πακέτων (PH)[14], περιλαμβάνει:

- ανίχνευση / επικύρωση ποιότητας του Preamble σε κατάσταση λήψης (σήμα `PREAMBLE_VALID`)
- ανίχνευση της λέξης συγχρονισμού (`Sync Word`) σε κατάσταση λήψης (σήμα `Sync OK`)

- ανίχνευση έγκυρων πακέτων (σήμα PKT\_valid)
- ανίχνευση CRC λαθών (σήμα CRC\_error)
- Αποθήκευση του ωφέλιμου φορτίου των δεδομένων στη μνήμη RX-FIFO
- Κατασκευή των πεδίων Preamble και Sync Word στη λειτουργία TX
- Κατασκευή του Header πεδίου εάν έχει ενεργοποιηθεί η λειτουργία PH
- Κατασκευή του πεδίου μήκος πακέτου στη λειτουργία TX
- Κατασκευή / ενσωμάτωση του ωφέλιμου φορτίου από τη μνήμη FIFO στο πακέτο προς αποστολή
- Κατασκευή του πεδίου CRC εάν είναι ενεργοποιημένη η λειτουργία.

### Λειτουργία Packet Handler

Στην κατάσταση λειτουργίας TX (εκπομπής) οι λειτουργίες PH έχουν νόημα εάν έχει ενεργοποιηθεί η λειτουργία FIFO. Τότε το πακέτο δημιουργείται αυτόματα από τη μνήμη FIFO και τα πεδία του PH που έχουν ενεργοποιηθεί [14]. Εάν η δομή του πακέτου κατασκευάζεται από τον μικροελεγκτή η λειτουργία PH δεν έχει νόημα στην κατάσταση TX [14]. Οι καταχωρητές που χρησιμοποιούνται για τον προγραμματισμό των πεδίων της λειτουργίας PH είναι οι καταχωρητές 33h και 34h Εικόνα 3.13 .

Register	D7	D6	D5	D4	D3	D2	D1	D0
33h Header Control 2	skipsyn	hdlen[2:0]			fixpklen	syncnlen[1:0]		prealen[8]
34h Preamble Length	prealen[7:0]							

Εικόνα 3.13 Καταχωρητής 33h και 34h, έλεγχος του Header



## Πεδίο προπομπός (Preamble)

Η λειτουργία PH δημιουργεί αυτόματα το πεδίο του προπομπού με μήκος αυτό που ορίστηκε στο πεδίο `prealen[8:0]` των καταχωρητών 33h-34h. Το πεδίο `prealen[8:0]` ορίζει τον αριθμό των nibbles (4 bit) στον προπομπό. Δεν είναι δυνατόν να έχουμε προπομπό μηδενικού μήκους. Σε περίπτωση που οριστεί μηδέν στο πεδίο `prealen[8:0]` το μήκος του προπομπού θεωρείται 1. Το μέγιστο μήκος του προπομπού μπορεί να ορισθεί στα 255,5 bytes. Εάν χρειάζεται προπομπός μεγαλύτερου μήκους πρέπει να απενεργοποιηθεί η λειτουργία PH και όλη η δομή του πακέτου να ελέγχεται από τον μικροελεγκτή. Το πρώτο ψηφίο του προπομπού είναι πάντα '0' και επομένως η ακολουθία του προπομπού είναι της μορφής '010101..'. Στην κατάσταση λήψης ο δέκτης ψάχνει για πλήθος συνεχόμενων ακολουθιών '0101..', nibbles (4-bit), ίσο με τον αριθμό που έχει οριστεί στο πεδίο `preath[4:0]` του καταχωρητή 35h. Όταν γίνει έγκυρη λήψη του προπομπού ο δέκτης μπορεί να δημιουργήσει μια διακοπή `PREAMBLE_VALID`. Με την έγκυρη λήψη πακέτου το bit `ipreaval` του καταχωρητή 04h γίνεται '1' και η διακοπή δημιουργείται εάν έχει ενεργοποιηθεί το αντίστοιχο bit `enpreaval` του καταχωρητή 06h. Εάν σε χρόνο που έχει οριστεί στα bit `invalid_preamble_threshold[3:0]` του καταχωρητή 60h δεν ληφθεί έγκυρος προπομπός τότε ο δέκτης θέτει το bit `preainval` του καταχωρητή 04h και δημιουργεί διακοπή εάν το αντίστοιχο bit `enpreainval` του καταχωρητή 06h, είναι ίσο με '1' [14].

## Πεδίο Λέξη Συγχρονισμού (Sync Field)

Το μήκος της λέξης ορίζεται στα bit D2-D1, `synclen[1:0]`, του καταχωρητή 33h Εικόνα 3.13. Το μέγεθος είναι από 1-4 bytes και το μήκος δεν μπορεί να είναι μηδενικό. Οι λέξεις συγχρονισμού ορίζονται στους καταχωρητές 36h, 37h, 38h και 39h και μεταδίδονται κατά φθίνουσα σειρά. Έτσι για `synclen=(00)2` μεταδίδεται η λέξη του καταχωρητή 36h (`sync word 3`), για `synclen=(01)2` μεταδίδεται η λέξη του καταχωρητή 36h ακολουθούμενη από τη λέξη του καταχωρητή 37h, για `synclen=(10)2` μεταδίδονται οι λέξεις των καταχωρητών 36h, 37h και 38h και για `synclen=(11)2` μεταδίδονται οι λέξεις των καταχωρητών 36h, 37h, 38h και 39h. Η λέξεις συγχρονισμού δείχνουν στον δέκτη αν το λαμβανόμενο πακέτο τον αφορά ή όχι. Η διαδικασία εντοπισμού της λέξης συγχρονισμού δεν αρχίζει εάν δεν εντοπιστεί ένα bit που να μην ανήκει στην ακολουθία του προπομπού. Επειδή ο δέκτης ψάχνει, μετά τη λήψη του προπομπού, για ακολουθίες bit οι οποίες δεν μοιάζουν με τη μορφή του προπομπού θα πρέπει να ορίζονται λέξεις συγχρονισμού διαφορετικές από την ακολουθία του προπομπού. Θα πρέπει να

αποφεύγονται λέξεις όπως AAh ή 55h, οι οποίες είναι σχεδόν ίδιες με την ακολουθία του προπομπού, γιατί ο δέκτης θα συνεχίζει να ψάχνει για το πεδίο συγχρονισμού. Ο εντοπισμός του πεδίου συγχρονισμού θέτει ίσο με '1' το bit iswdet του καταχωρητή 04h και μπορεί να ενεργοποιήσει μια διακοπή εάν είναι ίσο με '1' το αντίστοιχο bit eniswdet του καταχωρητή 06h. Ο ρόλος της λέξης συγχρονισμού είναι πολύ σημαντικός στη λειτουργία του δέκτη διότι μετά την αναγνώριση της λέξης συγχρονισμού μπορεί να γίνει ο διαχωρισμός των ωφέλιμων δεδομένων από το πακέτο [14].

### Πεδίο TX Header

Τα bit D6, D5 και D4 του καταχωρητή 33h ορίζουν το μήκος της επικεφαλίδας του πακέτου. Η επικεφαλίδα μπορεί να έχει μήκος από 1-4 bytes. Οι τιμές των bytes ορίζονται στους καταχωρητές 3Ah-3Dh. Για τιμή του hdlen=(001) μεταδίδεται η επικεφαλίδα 3 (καταχωρητής 3Ah), για τιμή hdlen=(010) μεταδίδεται η επικεφαλίδα 3 και η επικεφαλίδα 2 (καταχωρητές 3Ah και 3Bh αντίστοιχα) και ούτω καθεξής. Τα bits hdlen[2:0] του καταχωρητή 33h προσδιορίζουν το μήκος της επικεφαλίδας του πακέτου. Όταν ο δέκτης λάβει το πακέτο, τα πεδία της επικεφαλίδας μπορούν να συγκριθούν με συγκεκριμένες τιμές τις ονομαζόμενες τιμές ελέγχου. Ο αριθμός των τιμών ελέγχου που θα συγκριθούν μπορεί να είναι μικρότερος από τον αριθμό των πεδίων της επικεφαλίδας του πακέτου. Τα πεδία της επικεφαλίδας που θα συγκριθούν ορίζονται στα τέσσερα bits hdch[3:0] του καταχωρητή 32h Εικόνα 3.14.

Register	D7	D6	D5	D4	D3	D2	D1	D0
32h Header Control 1	bcen[3:0]				hdch[3:0]			

Εικόνα 3.14 Καταχωρητής 32h έλεγχος Header 1

Επομένως η τιμή hdch[3:0]='0000' ορίζει να μην γίνει σύγκριση στα πεδία της επικεφαλίδας, η τιμή hdch[3:0]='0001' ορίζει να γίνει σύγκριση του πεδίου 0 της επικεφαλίδας, η τιμή hdch[3:0]='0101' ορίζει να γίνει σύγκριση του πεδίου 0 και 2 της επικεφαλίδας κ.λπ. Οι τιμές των πεδίων της επικεφαλίδας που θα γίνει η σύγκριση με

τις ληφθείσες τιμές, πρέπει να προγραμματιστούν στους καταχωρητές 3Fh-42h. Επιπλέον μπορεί να προγραμματιστεί, στους καταχωρητές 43h-46h, να μην γίνει σύγκριση για συγκεκριμένα bit των πεδίων της επικεφαλίδας. Εάν θέλουμε να συγκριθούν όλα τα bits της επικεφαλίδας 3 και τα τέσσερα τελευταία bits της επικεφαλίδας 2 τότε μπορούμε να θέσουμε στον καταχωρητή 43h τιμή 0xFF και στον καταχωρητή 44h τιμή 0x0F. Στο δίκτυο που αναπτύχθηκε τα τρία πρώτα bytes είναι κοινά για όλους τους κόμβους, χαρακτηρίζοντας έτσι ένα δίκτυο, και το τελευταίο byte είναι ο αριθμός του σταθμού. Ο έλεγχος της επικεφαλίδας χρησιμοποιείται για να προσδιορίζει τον προορισμό ενός πακέτου σε ένα δίκτυο με πολλούς κόμβους [14].

### **Πεδίο μήκος πακέτου (Packet length)**

Το μήκος των ωφέλιμων δεδομένων μπορεί να συμπεριληφθεί αυτόματα στο σχηματισμό του πακέτου. Εάν στον καταχωρητή 33h το bit D3 (fixrklen) είναι ίσο με το 1, τότε το μήκος των ωφέλιμων δεδομένων δεν συμπεριλαμβάνεται στο σχηματισμό του πακέτου και ο πομπός και ο δέκτης θεωρούν ότι το μήκος των δεδομένων είναι σταθερό και γνωστό. Εάν είναι ενεργοποιημένη η λειτουργία PH το μήκος των ωφέλιμων δεδομένων πρέπει να ορίζεται στον καταχωρητή 3Eh. Το byte αυτό συμπεριλαμβάνεται στην επικεφαλίδα κατά τον αυτόματο σχηματισμό του πακέτου. Με αυτό τον τρόπο ο δέκτης γνωρίζει τον αριθμό των bytes που αποτελούν τα ωφέλιμα δεδομένα ο οποίος μπορεί να είναι και μεταβλητός. Επειδή το μήκος των ωφέλιμων δεδομένων προσδιορίζεται από ένα byte, το μέγιστο μήκος που μπορούν να έχουν είναι 255 Bytes. Σε περίπτωση που χρειαζόμαστε μεγαλύτερο μήκος ωφέλιμων δεδομένων, πρέπει να απενεργοποιηθεί η λειτουργία PH και όλα να δημιουργούνται από τον μικροελεγκτή, αυξάνοντας την πολυπλοκότητα και την απαιτούμενη υπολογιστική ισχύ.

### **Πεδίο CRC (Cycle redundancy Check)**

Η λειτουργία PH μπορεί να ενεργοποιηθεί ώστε να δημιουργεί CRC κώδικα ανάμεσα στα διάφορα πεδία του TX πακέτου. Για να ενεργοποιηθεί αυτή η λειτουργία πρέπει να γίνει encrc=1 του καταχωρητή 30h. Στον ίδιο καταχωρητή ορίζεται το είδος του πολυωνύμου που θα χρησιμοποιηθεί, για τη δημιουργία των δύο bytes που προστίθενται στο τέλος του πακέτου, καθώς και αν θα υπολογίζεται μόνο στα ωφέλιμα δεδομένα ή σε όλο το πακέτο εκτός προπομπού και λέξης συγχρονισμού [14].

### 3.5 Δομή Πακέτων Ασύρματου Κόμβου

Τα πακέτα τα οποία στέλνει ένας ασύρματος κόμβος έχουν την μορφή της Εικόνα 3.15

<b>Preamble</b> 55 55 55 55 h	<b>Sync Word</b> 2DD4 h	<b>TX Header</b> byte 3: 1 h byte 2: 2 h byte 1 :3 h byte 0 :A/A Κόμβου	<b>Packet Length</b> Υπολογίζεται δυναμικά	<b>DATA</b>	<b>CRC</b>
4 Bytes	2 Bytes	4 Bytes	1Byte		2 Bytes

**Εικόνα 3.15** Δομή Πακέτου Απομακρυσμένου Κόμβου

Ο προπομπός (preamble) έχει οριστεί στα 32bit και περιέχει 4 bytes με τιμές 55h. Η λέξη συγχρονισμού αποτελείται από δύο bytes 2Dh και D4h. Η επικεφαλίδα του πακέτου αποτελείται από 4 bytes. Τα τρία πρώτα bytes είναι κοινά στο δίκτυο και αποτελούν την ταυτότητα του δικτύου. Το τελευταίο byte έχει τον αριθμό του κόμβου (για τον MASTER κόμβο δεκαδικό 254). Το μήκος του πακέτου υπολογίζεται κατά τη δημιουργία των δεδομένων προς αποστολή. Τα δεδομένα έχουν την μορφή της Εικόνα 3.16

<b>Είδος Πακέτου</b> Data : AA h Cmd : 55 h	<b>Αριθμός Κόμβου</b> Tx Header byte 0	<b>Αριθμός πακέτου</b> A/A πακέτου	<b>Θερμοκρασία</b> Διαβάζεται από Καταχωρητή	<b>Τάση Πηγής</b> Διαβάζεται από Καταχωρητή	<b>Επίπεδο ισχύος Πομπού</b> Διαβάζεται από Καταχωρητή
1 Byte	1 Byte	1 Byte	1Byte	1Byte	1 Byte

**Εικόνα 3.16** Μορφή Δεδομένων Απομακρυσμένου Κόμβου

Το πρώτο πεδίο των δεδομένων χαρακτηρίζει το είδος του πακέτου. Αν το byte είναι ίσο με AAh τότε τα υπόλοιπα bytes θεωρούνται δεδομένα ενώ εάν είναι ίσο με 55h τότε τα υπόλοιπα δεδομένα θεωρούνται ως σήμα ελέγχου και ο δέκτης ενεργεί ανάλογα. Με τον τρόπο αυτό δίνεται η δυνατότητα να αποστέλλονται πακέτα ελέγχου στον δέκτη. Η λειτουργία αυτή μας επιτρέπει μαζί με τα πακέτα επιβεβαίωσης, τα οποία θεωρούνται πακέτα ελέγχου, να μπορούμε να στείλουμε εντολές ελέγχου στους απομακρυσμένους

κόμβους. Για να ελαχιστοποιηθεί η κατανάλωση ενέργειας, οι απομακρυσμένοι κόμβοι απενεργοποιούν τον δέκτη τους και τον ενεργοποιούν μετά την αποστολή ενός πακέτου ώστε να λάβουν το πακέτο επιβεβαίωσης. Το πακέτο επιβεβαίωσης μπορεί να περιέχει εντολές προς τον απομακρυσμένο κόμβο, ώστε να είναι δυνατόν να μεταβάλλονται παράμετροί του. Ο αριθμός του κόμβου χαρακτηρίζει τον απομακρυσμένο κόμβο και είναι πεδίο το οποίο έχει αποδοθεί κατά την εγκατάσταση του δικτύου. Ο απομακρυσμένος κόμβος στέλνει αύξοντα αριθμό πακέτου από μηδέν έως 255 και μετά ξεκινά πάλι από το μηδέν. Ακολουθεί ένα byte που είναι η θερμοκρασία του κόμβου. Το επόμενο byte είναι το επίπεδο τάσης της πηγής του κόμβου. Το byte επίπεδο ισχύος πομπού αφορά την ισχύ εκπομπής του πομπού.

Ο κύριος κόμβος του δικτύου αφού λάβει ένα έγκυρο πακέτο τότε δημιουργεί και στέλνει ένα πακέτο επιβεβαίωσης Εικόνα 3.17.

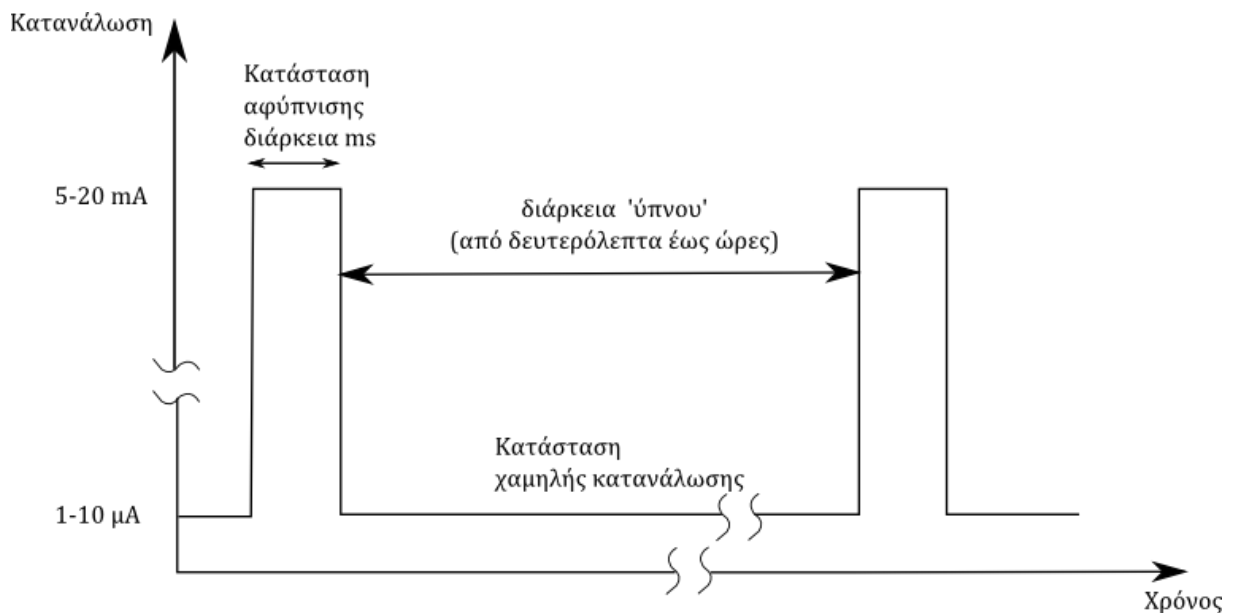
<u>Είδος Πακέτου</u>	<u>Αριθμός Κόμβου</u>	<u>Προορισμός</u>	<u>Επιβεβαίωση A/A</u>
Data : AA h Cmd : 55 h	Tx Header byte 0	Αριθμός απομακρυσμένου σταθμού	Αριθμός ληφθέντος πακέτου
1 Byte	1 Byte	1 Byte	1Byte

**Εικόνα 3.17** Μορφή Δεδομένων πακέτου ACK κύριου κόμβου

Το πακέτο επιβεβαίωσης έχει στο πρώτο byte το είδος του πακέτου (55h: πακέτο ελέγχου), στο δεύτερο τον αριθμό του κόμβου (254: κύριος κόμβος), στο τρίτο byte τον αριθμό του κόμβου προορισμού και τέλος τον αύξοντα αριθμό του πακέτου που έλαβε σωστά. Όταν ο κόμβος προορισμός λάβει το πακέτο επιβεβαίωσης, ελέγχει τον αύξοντα αριθμό του πακέτου με τον αριθμό του πακέτου που περιμένει επιβεβαίωση. Σε περίπτωση συμφωνίας θεωρεί ότι πακέτο έφτασε στον κεντρικό κόμβο με επιτυχία. Σε περίπτωση που δεν λάβει το πακέτο επιβεβαίωσης ή δεν συμφωνεί ο αριθμός τότε θεωρεί ότι το πακέτο χάθηκε και ακολουθεί το χρονοδιάγραμμά του.

### 3.6 Κατανάλωση Ενέργειας

Η κατανάλωση ενέργειας είναι από τους σημαντικότερους παράγοντες που επηρεάζουν της διάρκεια λειτουργίας ενός ασύρματου κόμβου. Τα περισσότερα πρωτόκολλα εστιάζουν την λειτουργίας τους με στόχο την μείωση της κατανάλωσης ώστε οι κόμβοι να λειτουργούν για μεγαλύτερο χρονικό διάστημα. Οι απομακρυσμένοι ασύρματοι κόμβοι λειτουργούν με μπαταρίες και τον περισσότερο χρόνο βρίσκονται σε κατάσταση χαμηλής κατανάλωσης ενέργειας. Η πηγή ενέργειάς τους είναι περιορισμένη και πρέπει να γίνει βέλτιστη χρήση της ώστε ο χρόνος λειτουργίας τους να είναι όσο το δυνατόν μεγαλύτερος. Το προφίλ κατανάλωσης ενέργειας ενός κόμβου φαίνεται στην Εικόνα 3.18.



**Εικόνα 3.18** Προφίλ κατανάλωσης ενέργειας απομακρυσμένου κόμβου

Το μήκος ενός πακέτου συμπεριλαμβάνοντας τα byte της λειτουργίας PH είναι  $L=18$  bytes και η ταχύτητα μετάδοσης είναι 9.6 Kbps. Ο χρόνος που χρειάζεται ένα πακέτο είναι:

$$T_{\text{πακέτου}} = \frac{18 * 8}{9.6} \text{ ms} = \frac{144}{9.6} = 15 \text{ msec}$$

Ο χρόνος μετάβασης από κατάσταση ύπνου σε κατάσταση αφύπνισης είναι της τάξης  $T_{\text{αφύπνισης}}=200 \text{ μsec}=0,2\text{msec}$  και είναι πολύ μικρός σε σχέση με τον  $T_{\text{πακέτου}}$ .

Σε μία ώρα ο χρόνος που ο πομπός λειτουργεί, με μία αποστολή το λεπτό, δίνεται από:

$$T_{active} = \frac{60 * (T_{\text{πακέτου}})}{3600} = \frac{60 * (15 * 10^{-3})}{3600} = \frac{0,9}{3600} h = 0,00025 h$$

Η κατανάλωση του πομποδέκτη για εκπομπή δεδομένων (κατάσταση μετάδοσης) δίνεται [15]:

$$I_{\text{tot-TX}} = I_{\text{RF}} = 30\text{mA}$$

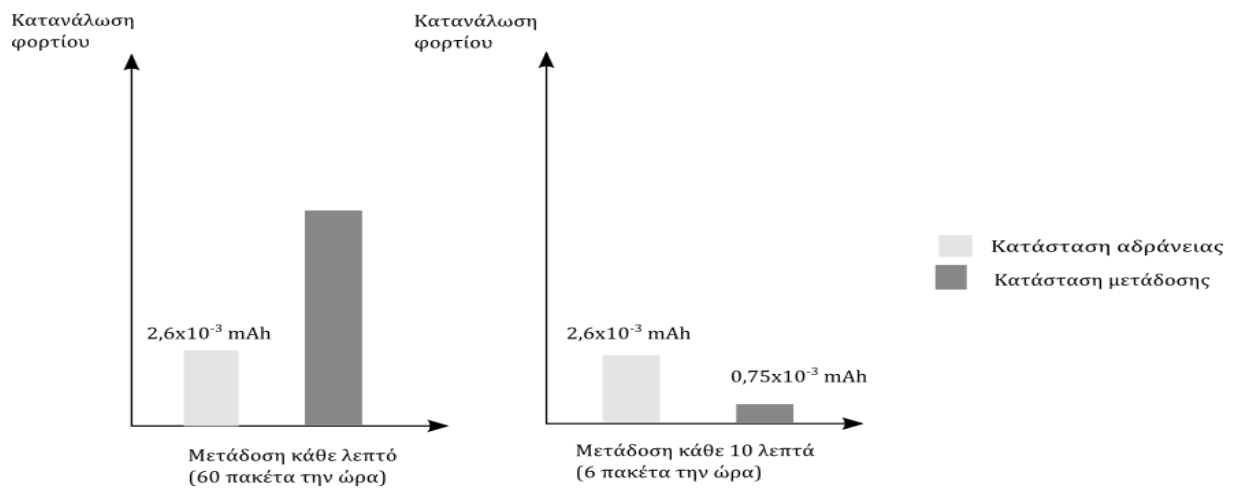
Το φορτίο που καταναλώνεται δίνεται από:

$$Q_{\text{tx}} = T_{\text{active}} * I_{\text{tot-tx}} = 0,00025 h * 30\text{mA} = 7,5 * 10^{-3} \text{mAh}$$

Η κατανάλωση φορτίου από τον μικροελεγκτή, όταν ο κόμβος βρίσκεται σε κατάσταση 'ύπνου' για μία ώρα υπολογίζεται :

$$Q_{\text{sleep}} = T_{\text{sleep}} * I_{\text{MCU\_sleep}} = 1h * 2,6\mu\text{A} = 2,6 * 10^{-3} \text{mAh}$$

Αν οι μεταδόσεις γίνονται ανά 10 λεπτά τότε οι παραπάνω υπολογισμοί θα εκτιμήσουν κατανάλωση φορτίου  $0,75 * 10^{-3} \text{mAh}$  στην κατάσταση μετάδοσης. Στην περίπτωση αυτή η κατανάλωση στην κατάσταση αδράνειας είναι μεγαλύτερη από αυτή της κατάστασης μετάδοσης. Αυτό σημαίνει ότι είναι πολύ σημαντικό να μειωθεί η κατανάλωση στην κατάσταση αδράνειας όπου λειτουργεί μόνο ο μικροελεγκτής.



**Εικόνα 3.19** Κατανάλωση φορτίου (μετάδοση κάθε λεπτό και κάθε 10 λεπτά)



# **Κεφάλαιο 4**

## **Επιλογή Υλικού**

### **Αρχιτεκτονική Κόμβων**

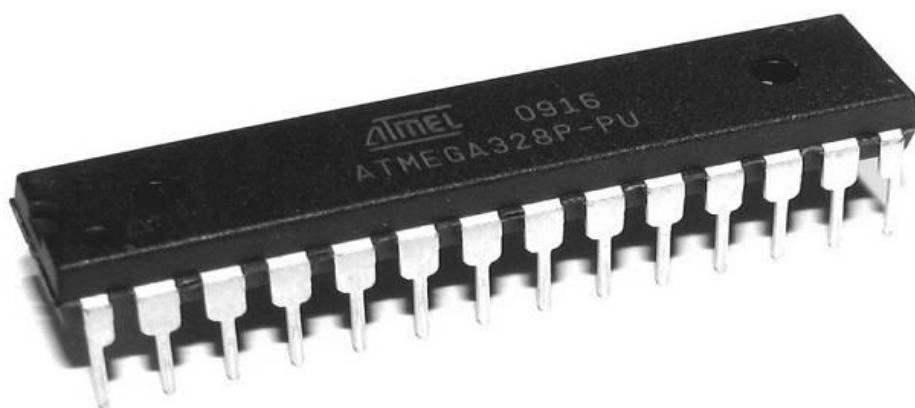
Για την κατασκευή του συστήματος απαιτείται η συνεργασία πομποδέκτη και μικροεπεξεργαστή. Υπάρχουν πολλοί μικροελεγκτές με ενσωματωμένο πομποδέκτη. Οι περισσότεροι έχουν διαθέσιμες ρουτίνες επικοινωνίας με πολλές λειτουργικότητες. Οι μικροελεγκτές της κατηγορίας αυτής είναι σε συσκευασίες ολοκληρωμένων που δεν μπορούν να χρησιμοποιηθούν σε πλακέτες δοκιμών και σχεδόν αδύνατον να ενσωματωθούν σε πλακέτα παρά μόνο από εξειδικευμένα εργαστήρια κατασκευής πλακετών.

#### **4.1 Επιλογή μικροελεγκτή**

Η επιλογή του μικροελεγκτή έγινε με κριτήριο το κόστος, την κατανάλωση, την ευκολία κατασκευής και τη διαθεσιμότητα στην ελληνική αγορά. Ο μικροελεγκτής θα πρέπει να τοποθετείται σε πλακέτα δοκιμών ώστε να γίνονται εύκολα δοκιμές και αλλαγές. Η

επιλογή της συσκευασίας DIP είναι ένας πρώτος περιορισμός. Ο μικροελεγκτής θα πρέπει να είναι προγραμματιζόμενος με εργαλεία που είναι διαθέσιμα στην ελληνική αγορά. Τα εργαλεία προγραμματισμού πρέπει να είναι ελεύθερα και να μπορεί να προγραμματιστεί σε γλώσσα υψηλού επιπέδου. Για τη μείωση του κόστους, θα πρέπει να μπορεί να λειτουργεί και χωρίς εξωτερικά πρόσθετα στοιχεία όπως κρυστάλλους ή ταλαντωτές [16].

Μετά από έρευνα στην ελληνική αγορά επιλέχθηκε ο μικροελεγκτής της ATMEL ATMega328P Εικόνα 4.1 [17]. Επίσης υπάρχει και η εμπειρία του Arduino το οποίο χρησιμοποιεί τον παραπάνω μικροελεγκτή και είναι ευρέως διαθέσιμο.



**Εικόνα 4.1** Μικροελεγκτής ATMEGA328P-PU

Ο μικροελεγκτής ATMEGA328P-PU μπορεί να λειτουργήσει χωρίς εξωτερικό κρύσταλλο με το εσωτερικό ρολόι των 8MHz. Το κόστος του είναι μικρό και διαθέτει τις απαραίτητες θύρες επικοινωνίας και εισόδου/εξόδου.

## 4.2 Συσκευή Προγραμματισμού μικροελεγκτή

Ο προγραμματισμός του μικροελεγκτή γίνεται με τη συσκευή AVRISP mkII Εικόνα 4.2. Ο προγραμματιστής AVRISP mkII προγραμματίζει όλους τους μικροελεγκτές 8-bit της ATMEL που διαθέτουν θύρα ISP (In-System Programming) και συνεργάζεται με το ολοκληρωμένο περιβάλλον προγραμματισμού Atmel Studio το οποίο διατίθεται δωρεάν [18].



**Εικόνα 4.2** Προγραμματιστής Μικροελεγκτών ATMEL.

Ο προγραμματισμός του μικροελεγκτή γίνεται μέσω της θύρας ISP Εικόνα 4.3. Η κόκκινη γραμμή στο καλώδιο δείχνει το pin 1 της θύρας ISP. Η θύρα ISP παρέχει την ευκολία του προγραμματισμού του μικροελεγκτή χωρίς να τον μεταφέρουμε σε εξωτερική συσκευή προγραμματισμού αλλά στο σύστημα όπου είναι τοποθετημένος.



**Εικόνα 4.3** Θύρα προγραμματισμού ISP

### 4.3 Επιλογή πομποδέκτη

Για τις ανάγκες της επικοινωνίας απαραίτητη είναι η ενσωμάτωση πομποδέκτη στο σύστημα. Ο πομποδέκτης πρέπει να έχει θύρα SPI (Serial Programming Interface) ώστε να μπορεί να επικοινωνεί με τον μικροελεγκτή. Η συχνότητα των 2.4MHz χρησιμοποιείται από διάφορους κατασκευαστές ασύρματων κόμβων μετρήσεων αλλά έχουν υψηλό κόστος και δυσκολία εφαρμογής. Λόγω ύπαρξης πολλών συσκευών στη συχνότητα αυτή όπως Wi-Fi και Bluetooth υπάρχει μεγαλύτερη πιθανότητα

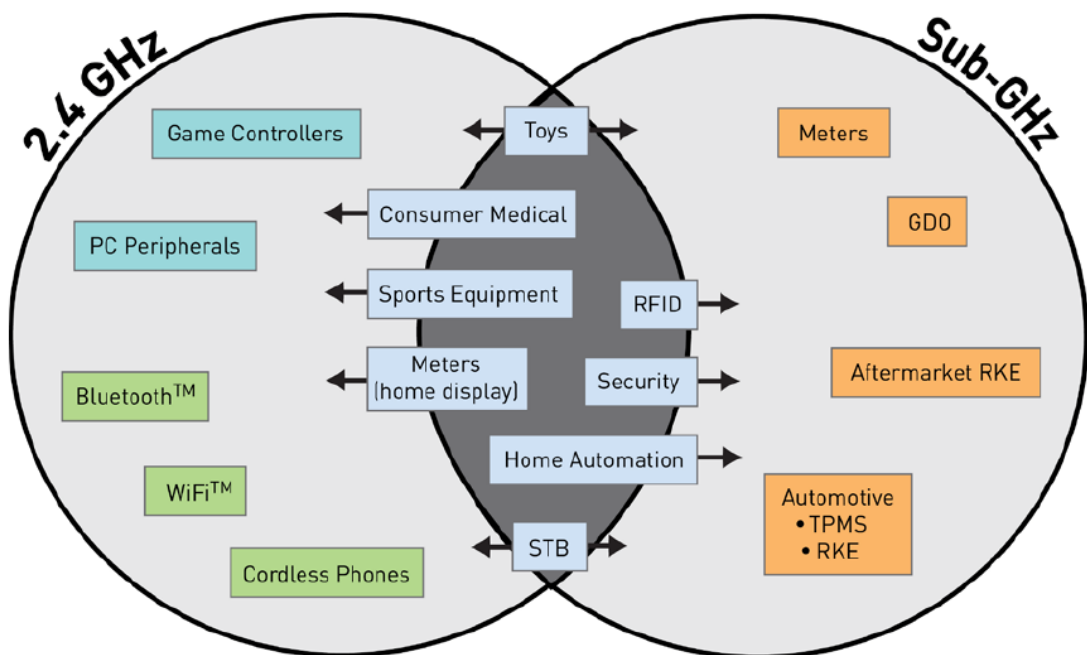
συγκρούσεων. Η επιλογή πομποδέκτη στη συχνότητα sub-GHz, όπως 434MHz ή 868MHz, ήταν προτιμότερη επειδή παρουσιάζουν καλύτερες ιδιότητες μετάδοσης. Για χαμηλές ταχύτητες μεταφοράς δεδομένων, όπως τα ασύρματα δίκτυα κόμβων μετρήσεων, οι sub-GHz πομποδέκτες έχουν μεγαλύτερη εμβέλεια, χαμηλότερη κατανάλωση ενέργειας και χαμηλότερο κόστος ανάπτυξης και λειτουργίας[05]. Οι πομποδέκτες sub-GHz προσφέρουν απλές λύσεις και λόγω της χαμηλής κατανάλωσής τους μπορούν να λειτουργούν, από μπαταρίες, για πολλά χρόνια. Τα βασικότερα πλεονεκτήματά τους, ως προς τα συστήματα των 2,4GHz, είναι [05]:

- Εμβέλεια: Το στενό εύρος ζώνης επιτρέπει τη μετάδοση σε αποστάσεις ενός χιλιομέτρου ή περισσότερο. Αυτό επιτρέπει την επικοινωνία τους με ένα κεντρικό κόμβο χωρίς τη χρήση ενδιάμεσων κόμβων αναμετάδοσης. Οι βασικοί λόγοι της υπεροχής των sub-GHz έναντι των GHz πομποδεκτών όσον αφορά την εμβέλεια είναι:
  - Καθώς τα ραδιοκύματα περνούν μέσα από τοίχους ή άλλα εμπόδια τα ραδιοκύματα εξασθενούν. Το ποσοστό εξασθένησης αυξάνεται σε υψηλότερες συχνότητες, ως εκ τούτου τα ραδιοκύματα των 2,4GHz εξασθενούν γρηγορότερα.
  - Τα GHz ραδιοκύματα εξασθενούν περισσότερο όταν ανακλώνται σε επιφάνειες.
  - Αν και τα ραδιοκύματα ταξιδεύουν σε ευθεία γραμμή, όταν πέφτουν στα άκρα αντικειμένου διαθλώνται. Επειδή όσο μειώνεται η συχνότητα τόσο αυξάνεται η γωνία διάθλασης αυτό επιτρέπει στα ραδιοκύματα sub-GHz να κάμπτονται περισσότερο γύρω από ένα εμπόδιο.
- Χαμηλές παρεμβολές. Οι συχνότητες των 2,4 GHz είναι γεμάτη από συγκρουόμενα σήματα από διάφορες πηγές όπως υπολογιστές και περιφερειακά με Wi-Fi και BlueTooth, κινητά τηλέφωνα και φούρνους μικροκυμάτων. Αυτή η πληθώρα ραδιοκυμάτων 2,4GHz δημιουργεί παρεμβολές. Οι συσκευές sub-GHz χρησιμοποιούνται σε δίκτυα με μικρό κύκλο λειτουργίας μειώνοντας την πιθανότητα συγκρούσεων. Αυτό εξασφαλίζει λιγότερες επαναλήψεις με αποτέλεσμα εξοικονόμηση ενέργειας.

- Χαμηλή κατανάλωση. Το εύρος ζώνης είναι αντιστρόφως ανάλογο της ευαισθησίας του δέκτη. Οι συχνότητες sub-GHz έχουν μικρότερο εύρος ζώνης επιτρέποντας τη λειτουργία σε μικρότερους ρυθμούς μετάδοσης μειώνοντας την κατανάλωση ενέργειας.

Γενικά όλα τα συστήματα ενός πομποδέκτη, σε υψηλές συχνότητες, χρειάζονται περισσότερο ρεύμα για να λειτουργήσουν ώστε να έχουν τις ίδιες επιδόσεις με ένα πομποδέκτη σε χαμηλότερη συχνότητα. Ένα μειονέκτημα των συστημάτων sub\_GHz είναι το μέγεθος της κεραίας. Για ένα σύστημα sub-GHz των 434MHz το μήκος της κεραίας είναι 17,3 εκατοστά. Εάν το μέγεθος της κεραίας είναι πρόβλημα, μπορεί να επιλεγεί η υψηλότερη συχνότητα των 950MHz.

Το παρακάτω σχήμα δείχνει τους τομείς και τις εμπορικές εφαρμογές όπου τα συστήματα sub-GHz και GHz έχουν επικρατήσει [05][05].



**Εικόνα 4.4** Τάση χρήσης συχνοτήτων σε καταναλωτικές, βιομηχανικές και εφαρμογές στην αυτοκινητοβιομηχανία [05]

Στην ελληνική αγορά, αλλά και στο διαδίκτυο, κυκλοφορούν ολοκληρωμένα της εταιρείας HOPE και συγκεκριμένα τα RFM12B και RFM22B τα οποία στηρίζονται σε ολοκληρωμένα της Silicon Labs. Το ολοκληρωμένο RFM22B [15] χρησιμοποιεί το ολοκληρωμένο Si4432 Εικόνα 4.5.



**Εικόνα 4.5** Πομποδέκτης RFM22B-D της HOPE

## 4.4 Επικοινωνία Κύριου Κόμβου με Η/Υ

Η επικοινωνία του κύριου κόμβου (MASTER) με τον ηλεκτρονικό υπολογιστή(Η/Υ), όπου συλλέγονται οι μετρήσεις, γίνεται μέσω της σειριακής θύρας του Η/Υ. Εάν ο Η/Υ δεν διαθέτει σειριακή θύρα (φορητός Η/Υ) χρησιμοποιείται καλώδιο μετατροπής από USB σε Serial Εικόνα 4.6.



**Εικόνα 4.6** Μετατροπής USB-Serial

Για την επικοινωνία του κεντρικού κόμβου με τη σειριακή θύρα του Η/Υ χρησιμοποιήθηκε το ολοκληρωμένο κύκλωμα της Intersil HIN232CP [19] Εικόνα 4.7 [19].



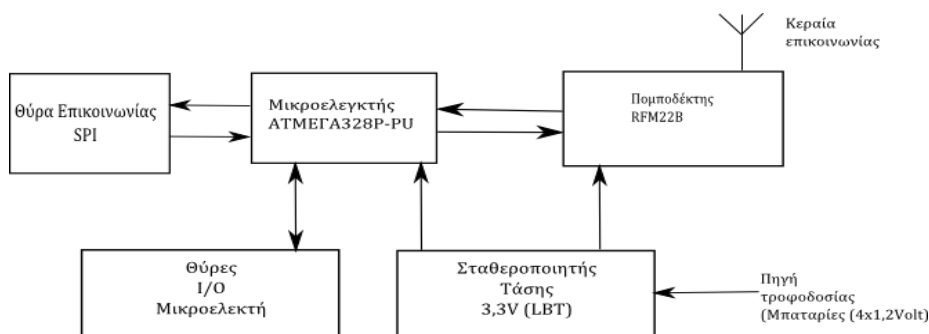
Εικόνα 4.7 Ολοκληρωμένο διασύνδεσης TTL-RS232

## 4.5 Αρχιτεκτονική Κόμβων

Οι κόμβοι του δικτύου είναι δύο τύπων : 1: κεντρικός κόμβος (τύπος: MASTER) και 2: απομακρυσμένοι κόμβοι (τύπος: NODE). Τα βασικά στοιχεία που αποτελούν και του δύο κόμβους είναι: μικροελεγκτής, πομποδέκτης, θύρα προγραμματισμού ISP και τροφοδοσία 3,3 Volt. Επιπλέον ο κύριος κόμβος διαθέτει κύκλωμα προσαρμογής λογικών επιπέδων, ολοκληρωμένο για την σύνδεση με τη σειριακή θύρα RS232, σειριακή θύρα και επιπλέον τροφοδοσία 5 Volt.

### 4.5.1 Αρχιτεκτονική απομακρυσμένων Κόμβων

Οι απομακρυσμένοι κόμβοι αποτελούνται από τα βασικά στοιχεία που προαναφέρθηκαν και φαίνονται στην Εικόνα 4.8.

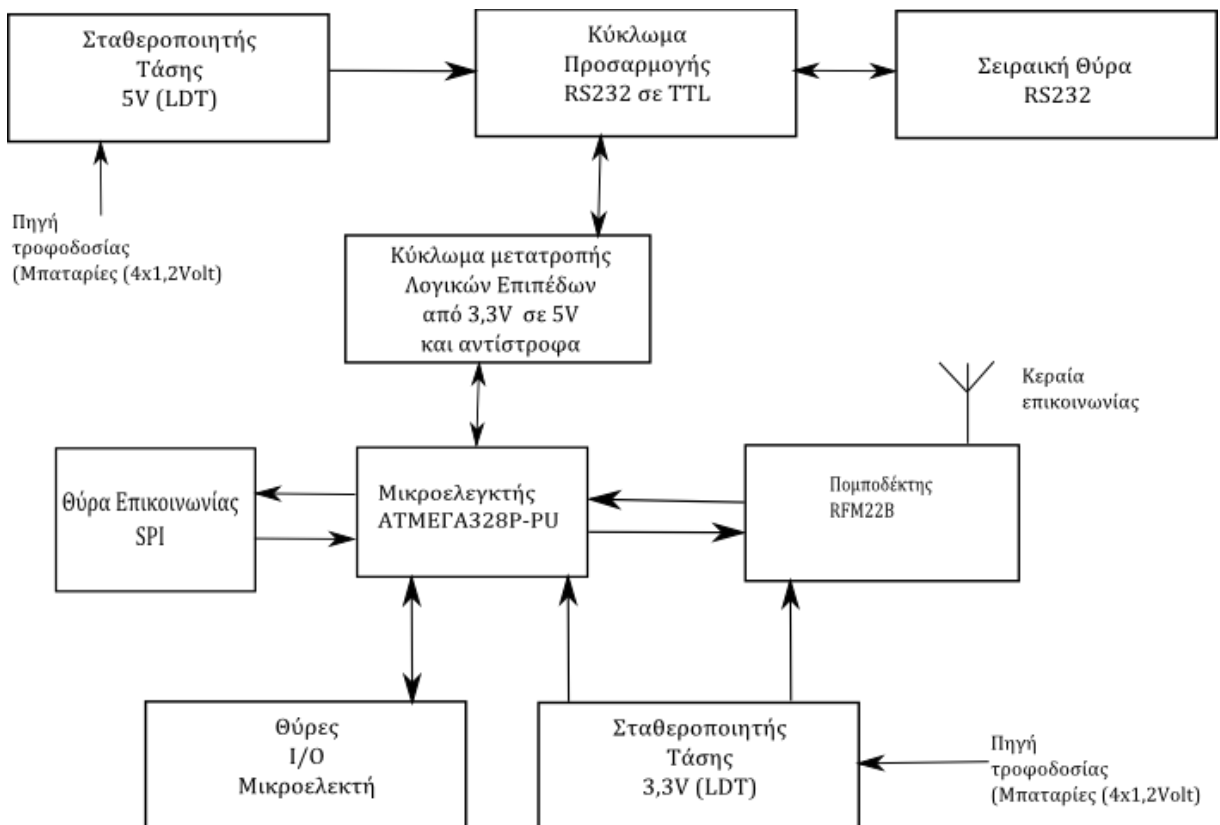


Εικόνα 4.8 Απομακρυσμένοι κόμβοι (NODE)

Δεν υπάρχουν εξωτερικά στοιχεία όπως κρύσταλλοι ή ταλαντωτές και ο χρονισμός του μικροελεγκτή γίνεται από το εσωτερικό του ρολόι. Η παραπάνω αρχιτεκτονική απλουστεύει πολύ την υλοποίηση ενός κόμβου και μειώνει σημαντικά το κόστος. Ο προγραμματισμός του συστήματος μπορεί να γίνει στο σύστημα, μέσω της θύρας ISP, χωρίς να μεταφέρουμε τον μικροελεγκτή σε εξωτερικό προγραμματιστή [16].

#### 4.5.2 Αρχιτεκτονική Κεντρικού Κόμβου

Η αρχιτεκτονική του κεντρικού κόμβου είναι ίδια με την προσθήκη επιπλέον μονάδων για την επικοινωνία του συστήματος με κεντρικό ηλεκτρονικό υπολογιστή Εικόνα 4.9



Εικόνα 4.9 Αρχιτεκτονική Κεντρικού κόμβου (MASTER)

Επειδή η σειριακή θύρα του υπολογιστή έχει διαφορετικά λογικά επίπεδα, χρειάζεται ένα κύκλωμα (ολοκληρωμένο) μετατροπής των επιπέδων αυτών σε επίπεδα συμβατά με το υπόλοιπο κύκλωμα. Ο πομποδέκτης λειτουργεί στα 3,3 Volt για αυτό επιλέχθηκε η



ίδια τάση λειτουργίας και για τον μικροελεγκτή. Το ολοκληρωμένο HIN232, για τη σύνδεση με τη σειριακή θύρα, λειτουργεί στα 5 Volt και επιβάλλεται η χρήση κυκλώματος μετατροπής λογικών επιπέδων από τα 3,3 Volt στα 5 Volt και αντίστροφα ώστε να είναι δυνατή η επικοινωνία μικροελεγκτή και ολοκληρωμένου κυκλώματος HIN232. Επίσης απαιτείται σταθεροποιητής τάσης στα 5 Volt για την τροφοδοσία του ολοκληρωμένου κυκλώματος προσαρμογής [20].

# Κεφάλαιο 5

## Κατασκευή Ασύρματων κόμβων Μετρήσεων

Οι κόμβοι μετρήσεων κατασκευάστηκαν πρώτα σε πλακέτες δοκιμών και κατόπιν σε τυπωμένο κύκλωμα. Όλοι οι κόμβοι είναι ίδιοι εκτός από τον κεντρικό κόμβο ο οποίος περιλαμβάνει ένα ολοκληρωμένο για την επικοινωνία με ηλεκτρονικό υπολογιστή για τη εισαγωγή των μετρήσεων σε βάση δεδομένων.

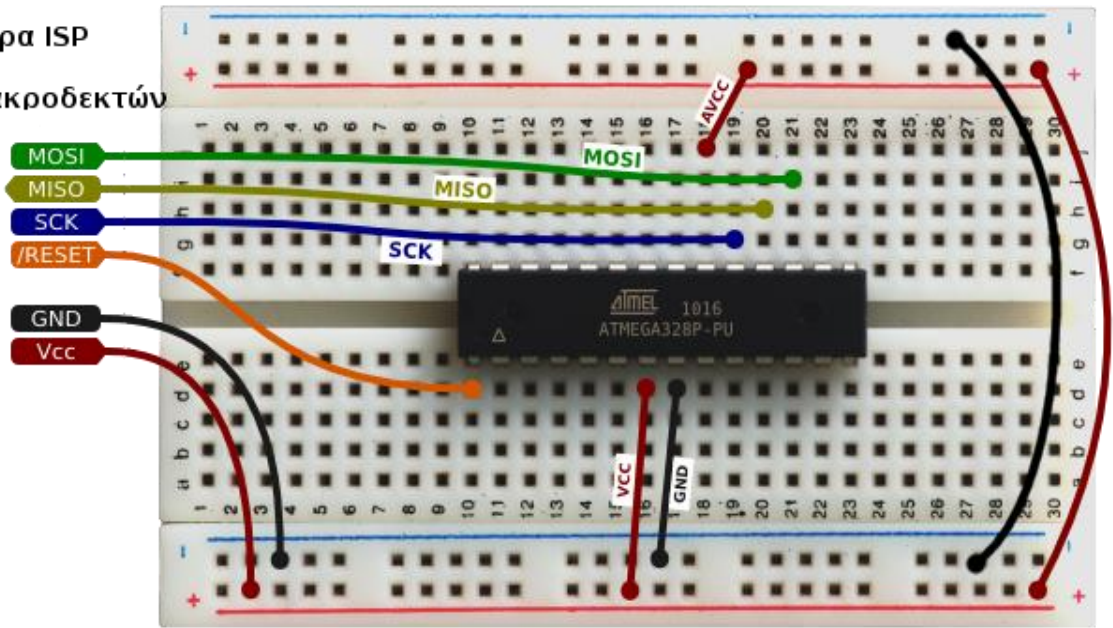
### 5.1 Κατασκευή Κόμβων σε Πλακέτα Δοκιμών

#### 5.1.1 Σύνδεση θύρας προγραμματισμού ISP μικροελεγκτή

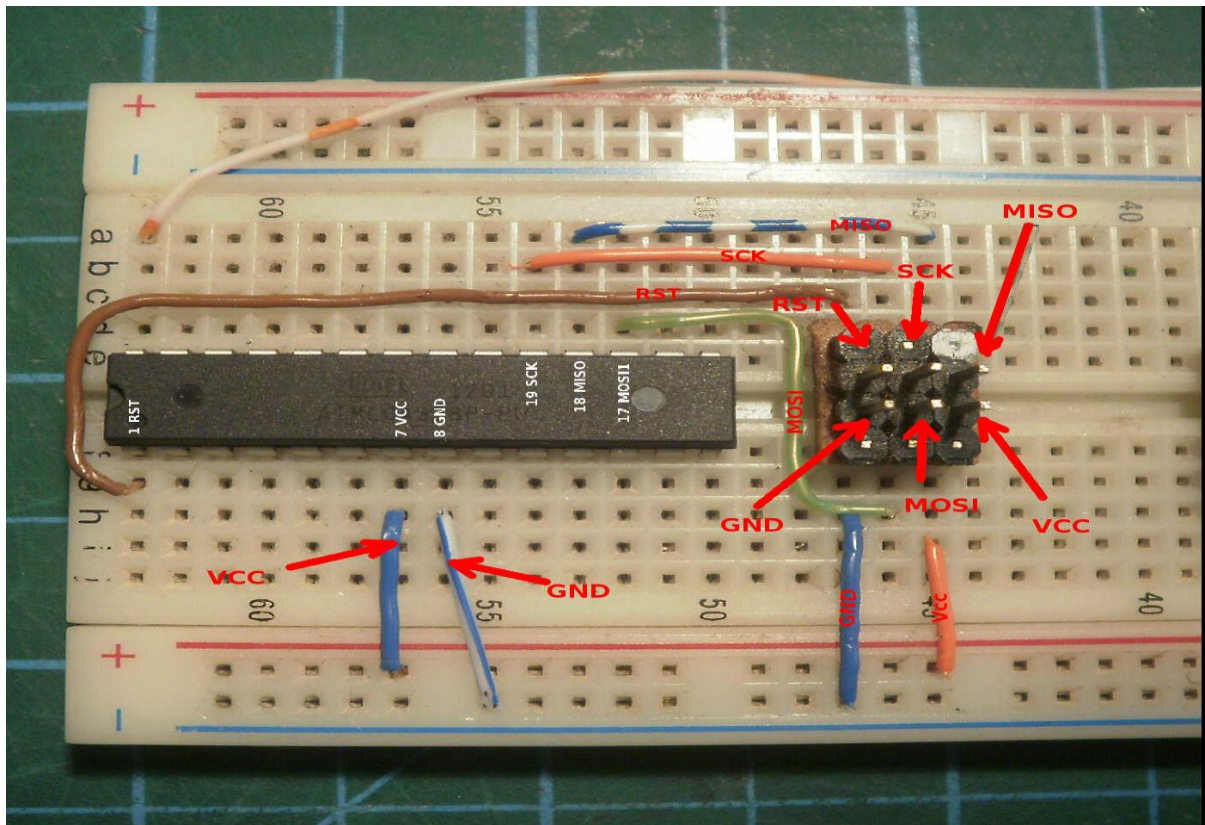
Σε κάθε πλακέτα δοκιμών δημιουργήθηκε η θύρα ISP όπως φαίνεται στην Εικόνα 5.1. Η θύρα ISP μας επιτρέπει να προγραμματίζουμε τον μικροελεγκτή χωρίς να τον μεταφέρουμε σε συσκευή προγραμματισμού.

Θύρα ISP

6 ακροδεκτών



Εικόνα 5.1 Θύρα προγραμματισμού ISP 6 ακροδεκτών

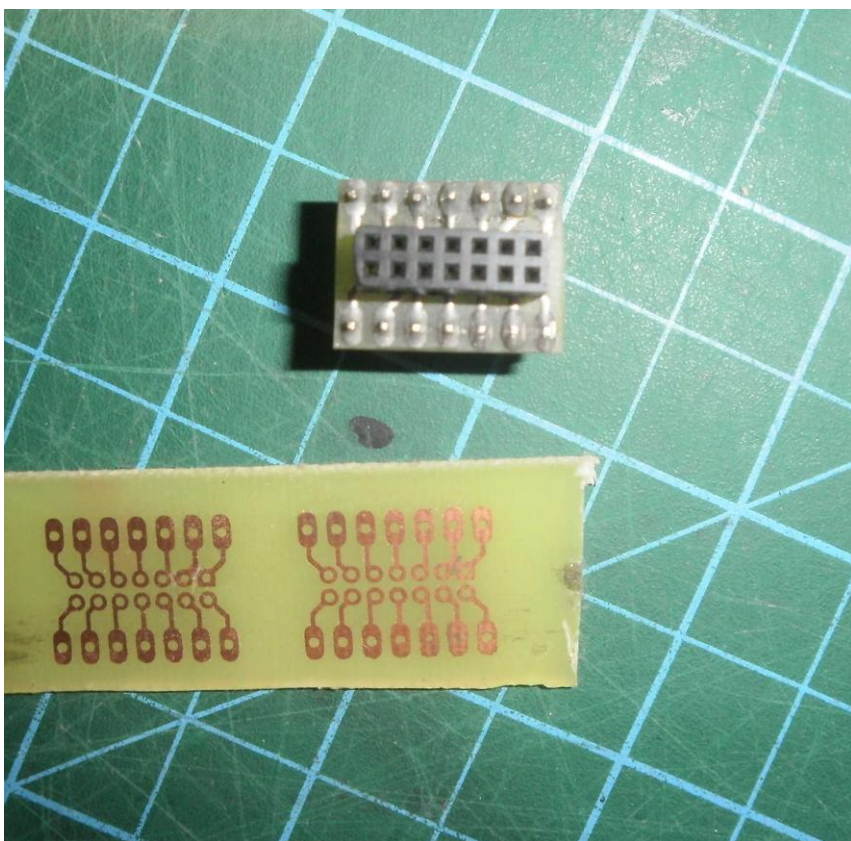


Εικόνα 5.2 Συνδεσμολογία θύρας ISP και μικροελεγκτή ATMEGA328P-PU

Στην πλακέτα δοκιμών της Εικόνα 5.2 φαίνεται η συνδεσμολογία της θύρας ISP και του μικροελεγκτή. Αυτές είναι οι ελάχιστες απαιτήσεις σε υλικό, εκτός της τροφοδοσίας, για να μπορέσουμε να προγραμματίσουμε μέσω της θύρας ISP τον μικροελεγκτή. Ο άσπρος κύκλος στους ακροδέκτες της θύρας δείχνει τον ακροδέκτη νούμερο 1.

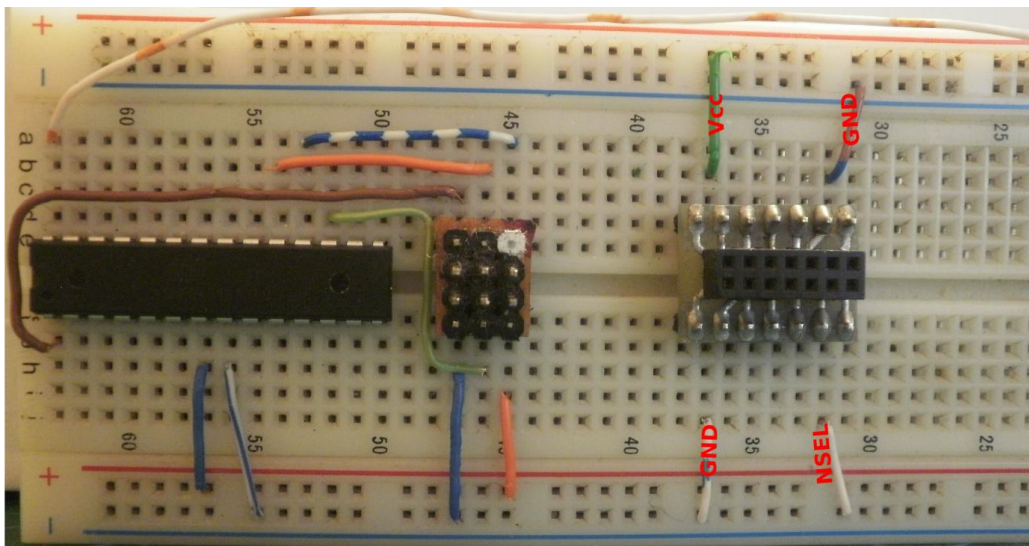
### 5.1.2 Σύνδεση Μικροελεγκτή Πομποδέκτη

Για να συνδεθεί ο πομποδέκτης RFM22B κατασκευάστηκε τυπωμένο κύκλωμα ώστε να μπορεί να προσαρμοστούν οι ακροδέκτες του στην πλακέτα δοκιμών Εικόνα 5.3. Ο πομποδέκτης RFM22B διαθέτει δύο σειρές ακροδεκτών σε απόσταση 2mm ενώ η πλακέτα δοκιμών έχει υποδοχές σε αποστάσεις 2,54mm.



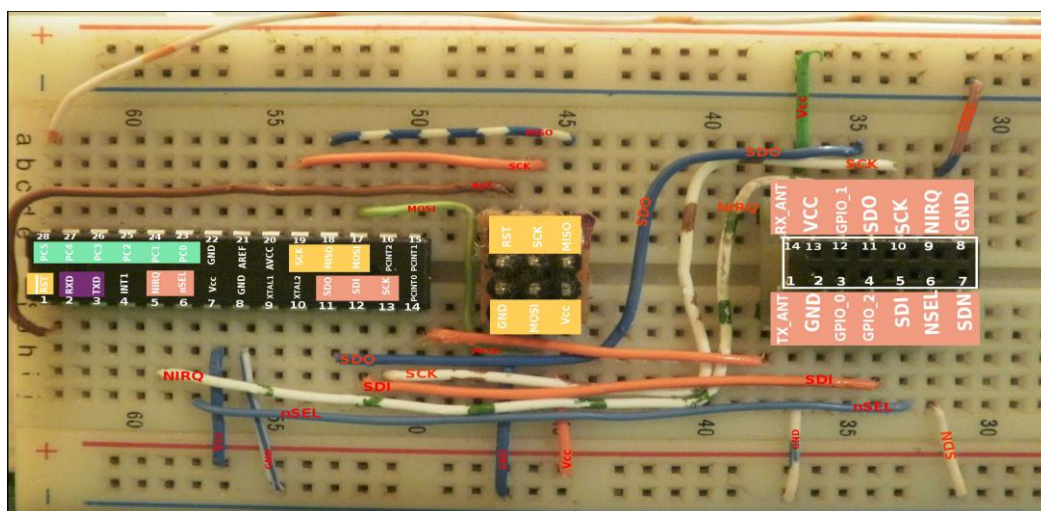
**Εικόνα 5.3** RFM22B μετατροπέας

Ο μετατροπέας του πομποδέκτης τοποθετείται στην πλακέτα δοκιμών όπως φαίνεται στην Εικόνα 5.4. Έχουν γίνει οι συνδέσεις της τροφοδοσίας και το NSEL συνδέεται στη γείωση για να είναι ο πομποδέκτης σε κατάσταση λειτουργίας.



Εικόνα 5.4 Σύνδεση μετατροπέα πομποδέκτη στην πλακέτα δοκιμών

Στην Εικόνα 5.5 φαίνονται όλες οι συνδέσεις του μικροελεγκτή με τη θύρα ISP και τον πομποδέκτη RFM22B. Όπως προκύπτει από την εικόνα υπάρχουν 6 θύρες ελεύθερες του μικροελεγκτή, χρώμα ανοικτό πράσινο (εισόδου / εξόδου ψηφιακές, ή 6 είσοδοι μετατροπής αναλογικού σήματος σε ψηφιακό).



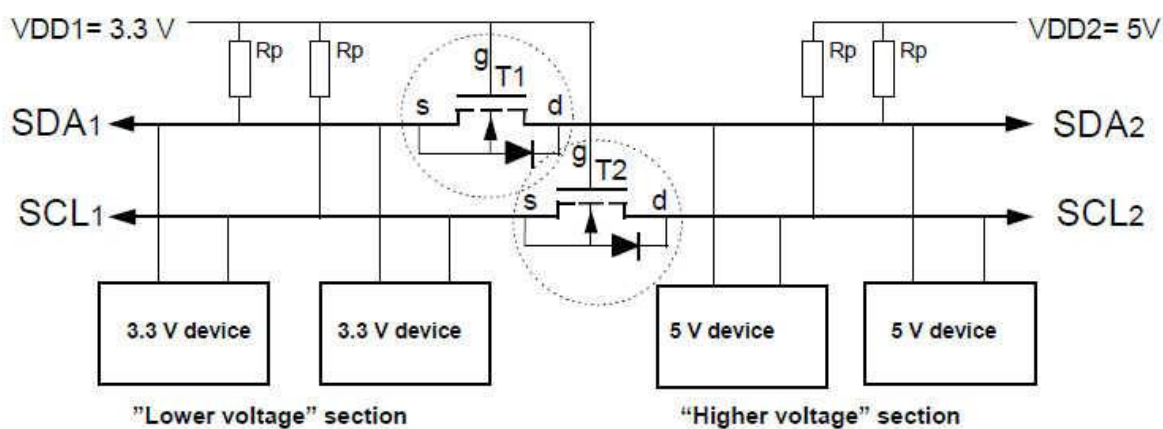
Εικόνα 5.5 Πλήρης λειτουργικός κόμβος σε πλακέτα δοκιμών

### 5.1.3 Σύνδεση Μικροελεγκτή με Ηλεκτρονικό Υπολογιστή

Ο κεντρικός κόμβος συλλέγει ασύρματα τα δεδομένα από τους απομακρυσμένους κόμβους και μέσω της USART που διαθέτει επικοινωνεί με τον Η/Υ ώστε να εισάγει τις μετρήσεις στη βάση δεδομένων. Για την επικοινωνία του μικροελεγκτή με τον Η/Υ χρησιμοποιήθηκε το ολοκληρωμένο HIN232CP [19] ώστε να είναι δυνατή η σύνδεση του κεντρικού κόμβου με τη σειριακή θύρα RS232. Το ολοκληρωμένο HIN232CP είναι ένα κύκλωμα διασύνδεσης που πληροί τις προδιαγραφές EIA RS232E και είναι κατάλληλο για εφαρμογές όπου δεν υπάρχει διαθέσιμη τάση  $\pm 12V$ . Λειτουργεί με  $+5V$  και διαθέτει κυκλώματα για τη δημιουργία τάσεων  $+10V$  και  $-10V$  που χρειάζονται για τη σειριακή θύρα του υπολογιστή. Επειδή ο πομποδέκτης και ο μικροελεγκτής λειτουργούν στα  $3,3V$  χρειάζεται τροφοδοσία  $5V$  για το ολοκληρωμένο HIN232CP. Επιπλέον χρειάζονται κυκλώματα προσαρμογής της τάσης των ακροδεκτών 2 και 3 (RXD και TXD) του μικροελεγκτή ώστε να συνεργάζεται με το κύκλωμα διασύνδεσης χωρίς την ανάγκη σήματος ελέγχου κατεύθυνσης.

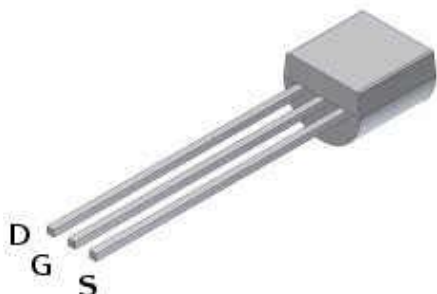
### Λειτουργία Μετατροπέα Λογικών Επιπέδων

Η αμφίδρομη μετατροπή λογικών επιπέδων χρησιμοποιείται για να διασύνδεει δύο συστήματα με διαφορετική τάση λειτουργίας [21]. Για τη διασύνδεση των δύο κυκλωμάτων χρησιμοποιήθηκε η συνδεσμολογία της Εικόνα 5.6 .



Εικόνα 5.6 Μετατροπή λογικών επιπέδων [21]

Για τη δημιουργία του κυκλώματος μετατροπής λογικών επιπέδων χρησιμοποιήθηκαν δυο MOSFET τρανζίστορ BS170 και τέσσερις αντιστάσεις Εικόνα 5.7 [22].



**Εικόνα 5.7** BS170 MOSFET

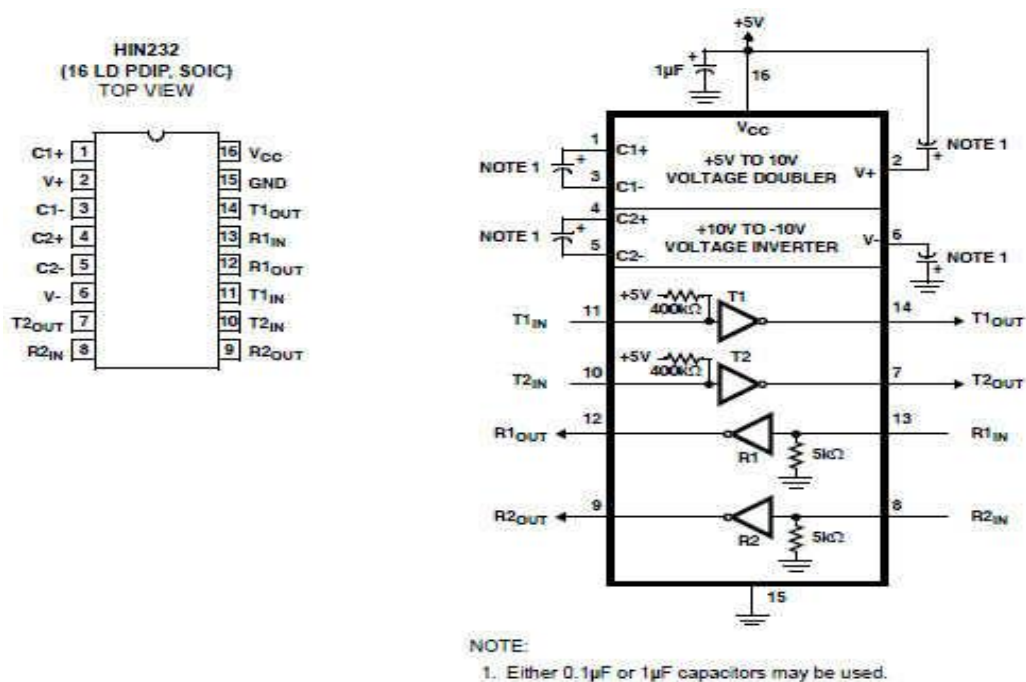
Οι μετατροπείς λογικών επιπέδων είναι ίδιοι για τις γραμμές SDA (Serial Data) και SCL (SerialClock) και αποτελούνται από ένα N-MOSFET. Οι πύλες (g) συνδέονται στη χαμηλή τάση, οι πηγές (s) στις γραμμές SDA και SCL της χαμηλής τάσης και η εκροή στις γραμμές SDA και SCL της υψηλής τάσης. Η λειτουργία του μετατροπέα λογικών επιπέδων συνοψίζεται παρακάτω [22][21].

1. Καμία πλευρά δεν τοποθετεί τις γραμμές σε χαμηλό δυναμικό. Στην πλευρά της χαμηλής τάσης η πηγή και η πύλη βρίσκονται στο ίδιο δυναμικό (3,3V), η τάση  $V_{gs}$  είναι κάτω από το κατώφλι και επομένως το MOSFET δεν άγει. Στην πλευρά του υψηλού δυναμικού το δυναμικό της εκροής, μέσω των αντιστάσεων, βρίσκεται στα 5V. Και οι δύο πλευρές βρίσκονται σε υψηλό δυναμικό αλλά διαφορετικό η κάθε μια.
2. Η πλευρά των 3,3V θέτει το δυναμικό στο χαμηλό επίπεδο. Η πηγή πηγαίνει σε χαμηλό δυναμικό ενώ η πύλη βρίσκεται σε υψηλό. Η τάση  $V_{gs}$  γίνεται μεγαλύτερη από το κατώφλι και το MOSFET άγει. Η πλευρά υψηλού δυναμικού πηγαίνει σε χαμηλό δυναμικό μέσω της αγωγής του MOSFET. Και οι δύο πλευρές βρίσκονται στο ίδιο χαμηλό δυναμικό.
3. Η πλευρά 5V θέτει τη γραμμή σε χαμηλό δυναμικό. Μέσω της διόδου το δυναμικό της  $V_{gs}$  γίνεται μεγαλύτερο από τα κατώφλι και το MOSFET άγει θέτοντας την πλευρά 3,3V σε χαμηλό δυναμικό.

Οι τρεις καταστάσεις δείχνουν ότι τα λογικά επίπεδα μεταφέρονται αμφίδρομα ανεξάρτητα από την πλευρά που οδηγεί το κύκλωμα. Το ελάχιστο δυναμικό που μπορούμε να έχουμε στην πλευρά του χαμηλού δυναμικού εξαρτάται από το δυναμικό  $V_{gs(th)}$  του MOSFET. Το MOSFET BS170 έχει  $V_{gs(th)} = 2,1V$  και επομένως μπορεί να λειτουργήσει με την πλευρά του χαμηλού δυναμικού στα 3,3V [20].

## Λειτουργία Κυκλώματος Διασύνδεσης RS232 και Μικροελεγκτή

Για την επικοινωνία του μικροελεγκτή με τον ηλεκτρονικό υπολογιστή χρειάζονται προσαρμοστικά κυκλώματα επειδή τα επίπεδα τάσης του μικροελεγκτή και της σειριακής θύρας του Η/Υ είναι διαφορετικά. Για τη σύνδεση του μικροελεγκτή με τον ηλεκτρονικό υπολογιστή χρησιμοποιήθηκε το ολοκληρωμένο HIN232CP της Intersil Εικόνα 5.8 [19]. Υπάρχουν παρόμοια ολοκληρωμένα από την MAXIM Integrated, την Texas Instruments και άλλες εταιρείες που είναι συμβατά μεταξύ τους.

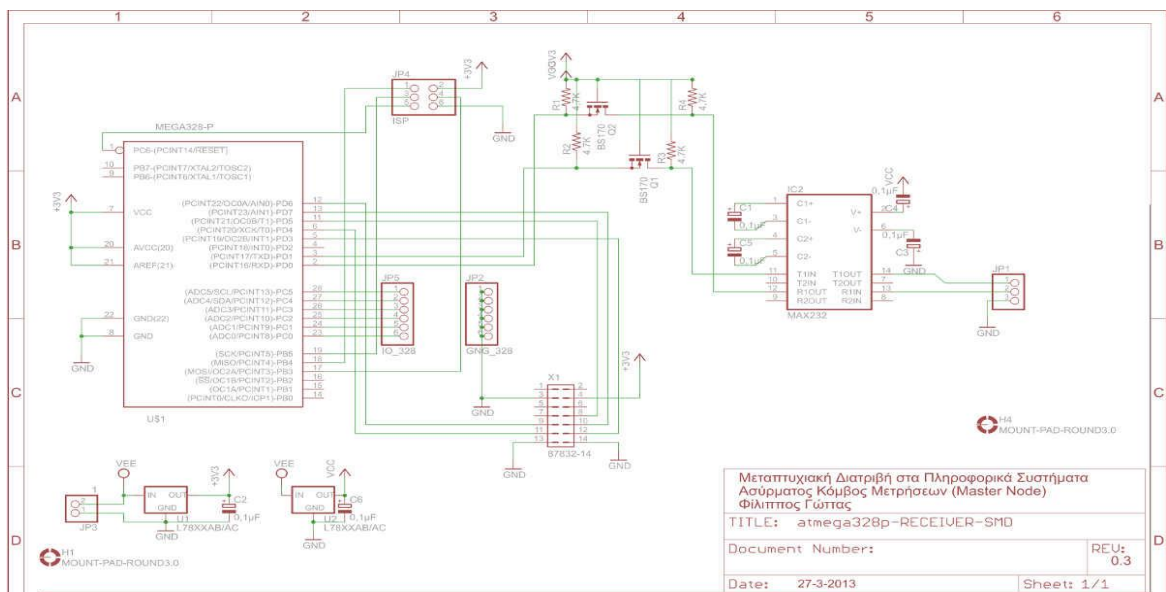


Εικόνα 5.8 Ολοκληρωμένο HIN232 - Κύκλωμα TTL σε RS232

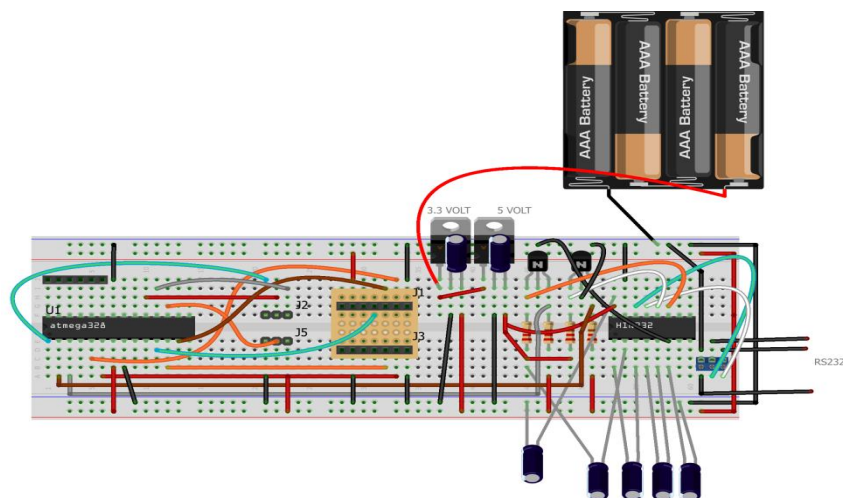
Το παραπάνω ολοκληρωμένο χρειάζεται μόνο μια πηγή τάσης 5V και πέντε πυκνωτές για να λειτουργήσει. Οι τάσεις, απαραίτητες για τη λειτουργία της σειριακής θύρας



RS232, δημιουργούνται από το ολοκληρωμένο με τη βοήθεια των πυκνωτών και τα κυκλώματα πολλαπλασιασμού και αναστροφής που έχει. Οι ταχύτητες που μπορεί να επιτύχει είναι 120Kbps. Ο πομπός του ολοκληρωμένου HIN232, είναι TTL/CMOS συμβατός και μετατρέπει τα σήματα εισόδου σε σήματα εξόδου για τη σειριακή RS232. Αντίστροφα ο δέκτης του HIN232, δέχεται σήματα RS232 και τα μετατρέπει σε σήματα εξόδου TTL μεταξύ 0V και VCC. Το ηλεκτρονικό διάγραμμα του κεντρικού κόμβου φαίνεται στην Εικόνα 5.9 και η υλοποίησή του σε πλακέτα δοκιμών στην Εικόνα 5.10.



Εικόνα 5.9 Ηλεκτρονικό σχέδιο κεντρικού κόμβου



Εικόνα 5.10 Κεντρικός κόμβος – Επικοινωνία RS232

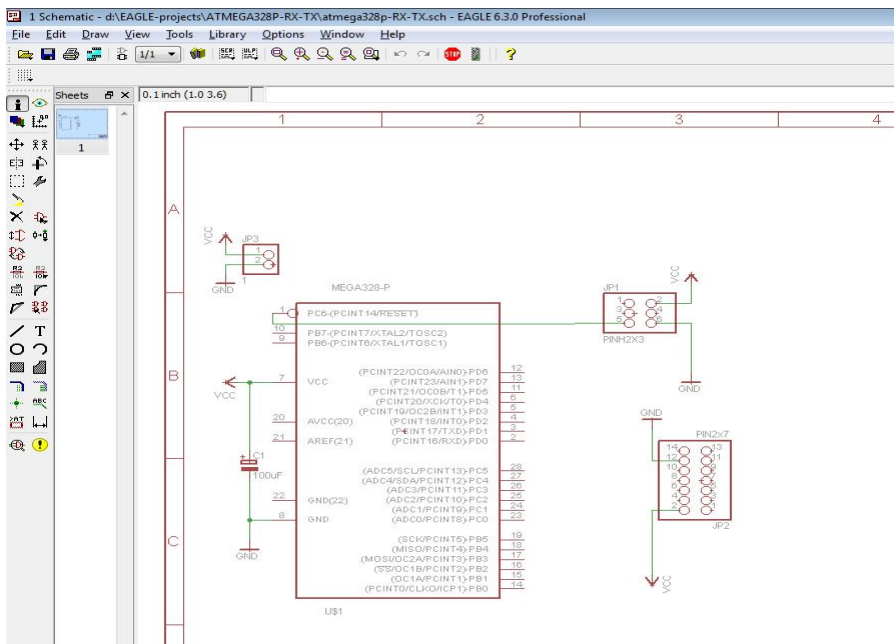
## 5.2 Κατασκευή Κόμβων σε Τυπωμένο Κύκλωμα

Αφού ελέγχθηκε η λειτουργία των κόμβων στις πλακέτες δοκιμών σχεδιάστηκε τυπωμένο κύκλωμα και κατασκευάστηκαν οι κόμβοι σε πλακέτα διπλής όψης. Για την κατασκευή των πλακετών χρειάστηκε να χρησιμοποιηθούν χημικά όπως υδροξείδιο του νατρίου και τριχλωριούχος σίδηρος. Και τα δύο υλικά είναι επικίνδυνα (τοξικά - καυστικά) και η χρήση τους απαιτεί μέτρα προστασίας για την αποφυγή σοβαρών προβλημάτων υγείας. Απαραίτητη είναι η χρήση πλαστικών γαντιών και πλαστικής ποδιάς και η χρήση τους να γίνεται σε εξωτερικό ή πολύ καλά αεριζόμενο χώρο.

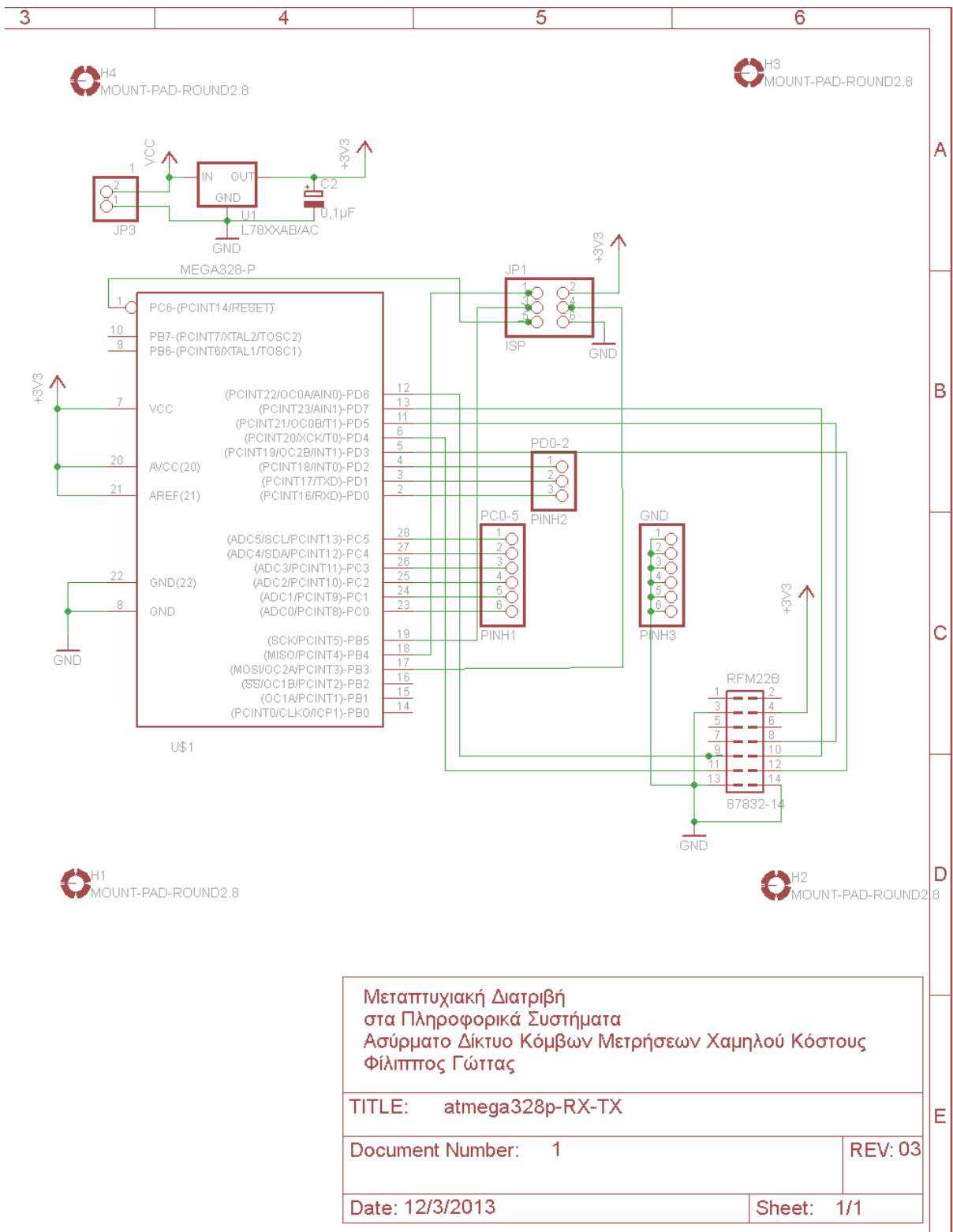
### 5.2.1 Σχεδιασμός Τυπωμένου Κυκλώματος Κόμβων

Για το σχεδιασμό του τυπωμένου κυκλώματος χρησιμοποιήθηκε το λογισμικό EAGLE [23]. Κύριος στόχος ήταν να σχεδιαστεί ένα κύκλωμα μικρών διαστάσεων. Στις βιβλιοθήκες του λογισμικού βρέθηκαν σχεδόν όλα τα ολοκληρωμένα και παθητικά στοιχεία που χρησιμοποιήθηκαν.

Το πρώτο βήμα είναι ο σχεδιασμός του ηλεκτρονικού κυκλώματος του κόμβου. Μετά ακολουθεί η τοποθέτηση των ολοκληρωμένων, των ακροδεκτών και των πηγών τάσης και τέλος γίνονται οι συνδέσεις μεταξύ των επιμέρους ολοκληρωμένων Εικόνα 5.12 .

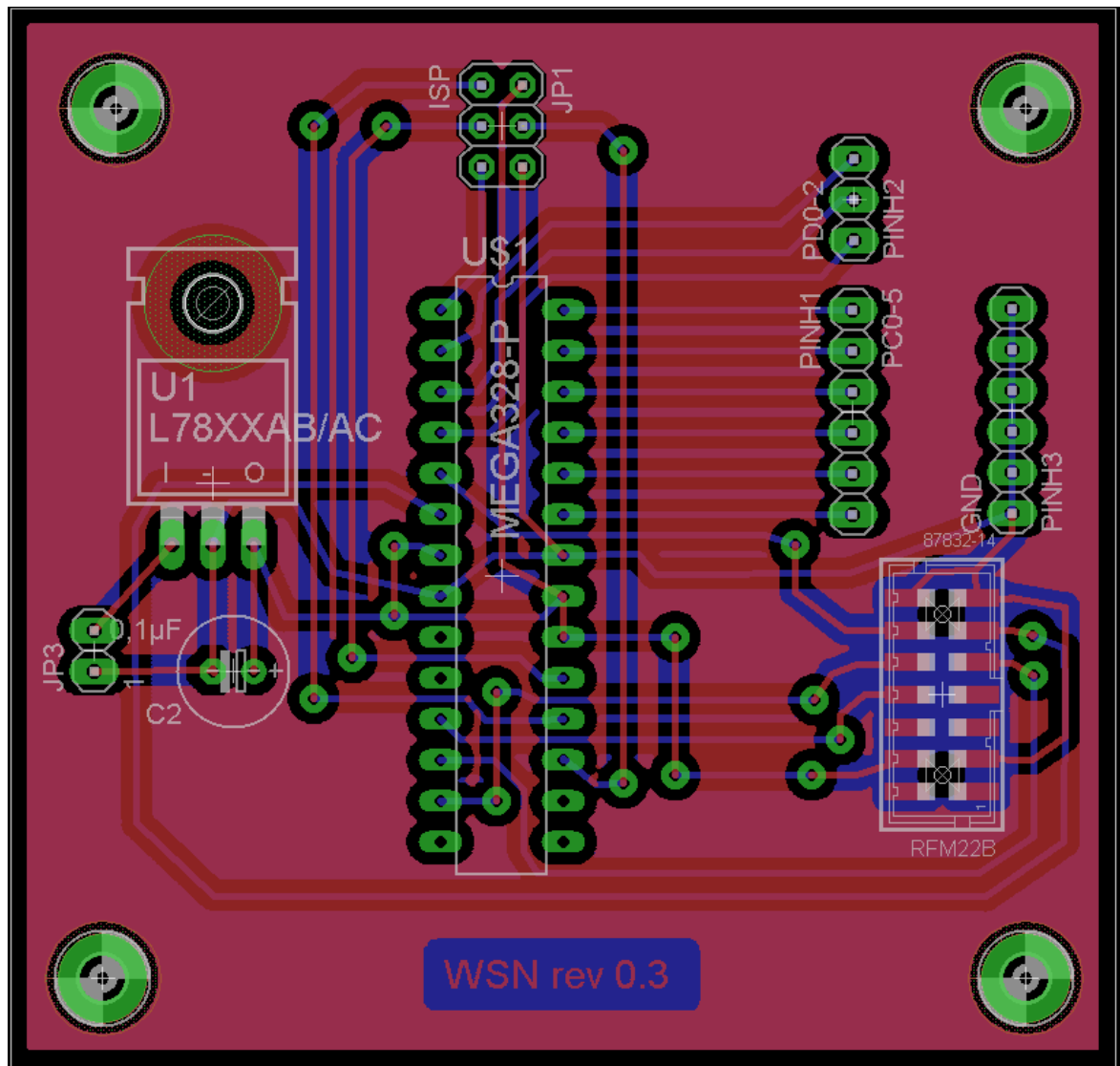


Εικόνα 5.11 Αρχικό στάδιο ηλεκτρονικού κυκλώματος



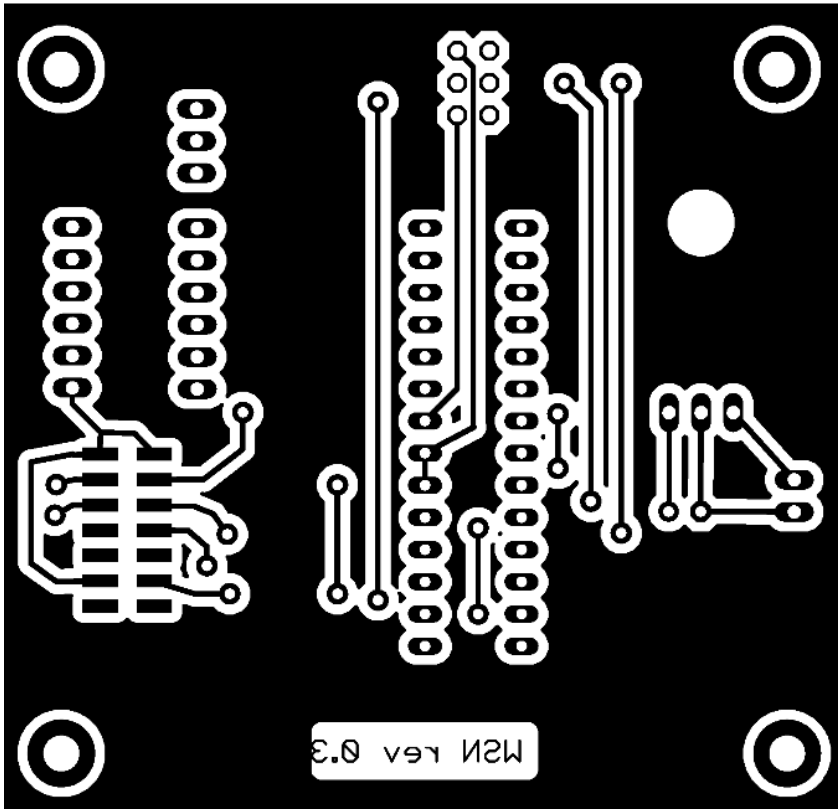
Εικόνα 5.12 Ηλεκτρονικό σχέδιο απομακρυσμένου κόμβου

Στην Εικόνα 5.11 και Εικόνα 5.12 φαίνονται το αρχικό στάδιο της σχεδίασης και το πλήρες ηλεκτρονικό σχέδιο ενός απομακρυσμένου κόμβου. Αφού γίνει ο έλεγχος των συνδέσεων για τυχόν λάθη το επόμενο βήμα είναι η δημιουργία των συνδέσεων του τυπωμένου κυκλώματος Εικόνα 5.13. Μετά τον αρχικό σχεδιασμό έγινε βελτιστοποίηση του συστήματος για εξοικονόμηση χώρου.

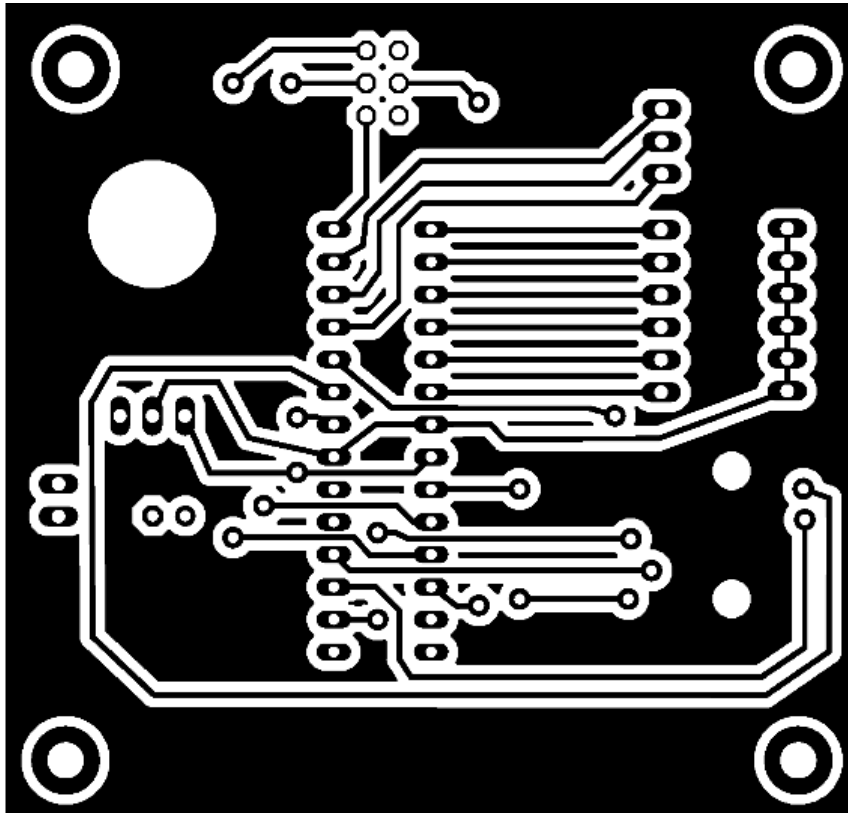


Εικόνα 5.13 Δημιουργία διαδρόμων τυπωμένου κυκλώματος

Για τη δημιουργία του τυπωμένου κυκλώματος χρησιμοποιήθηκε πλακέτα δύο όψεων. Από το διάγραμμα της Εικόνα 5.13 εκτυπώνονται οι μάσκες της επάνω και κάτω όψης του τυπωμένου κυκλώματος Εικόνα 5.14 και Εικόνα 5.15 αντίστοιχα.



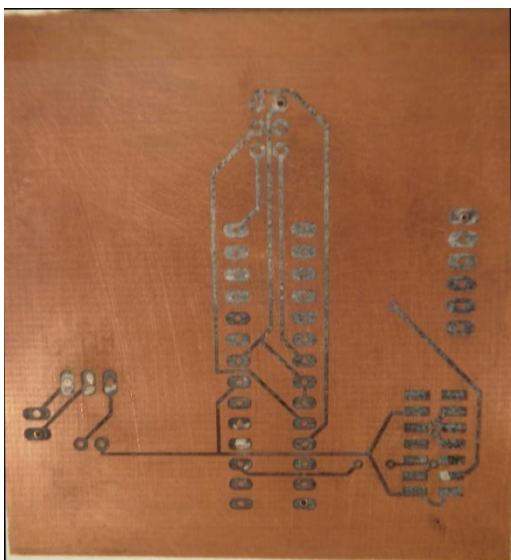
Εικόνα 5.14 Μάσκα επάνω όψης τυπωμένου



Εικόνα 5.15 Μάσκα κάτω όψης τυπωμένου

Οι μάσκες αυτές μεταφέρονται στις αντίστοιχες πλευρές της πλακέτας διπλής όψης. Υπάρχουν διάφοροι τρόποι μεταφοράς της μάσκας σε πλακέτα. Στον αρχικό σχεδιασμό επιλέχθηκε ένας απλός τρόπος ώστε να μπορεί να κατασκευαστεί εύκολα και γρήγορα η πλακέτα. Οι μάσκες εκτυπώθηκαν σε φωτογραφικό χαρτί με μέγιστη ανάλυση και σε εκτυπωτή laser. (Προσοχή!!! Κατά την εκτύπωση υπάρχει περίπτωση να αποκολληθεί το επάνω στρώμα του φωτογραφικού χαρτιού και να καταστρέψει το τύμπανο του εκτυπωτή. Καλό είναι να μην γίνονται πολλές εκτυπώσεις συνεχόμενα ώστε να μην αυξηθεί πολύ η θερμοκρασία στον εκτυπωτή). Χρησιμοποιώντας ηλεκτρικό σίδερο μεταφέρθηκε ο γραφίτης από το φωτογραφικό χαρτί επάνω στην πλακέτα δύο όψεων

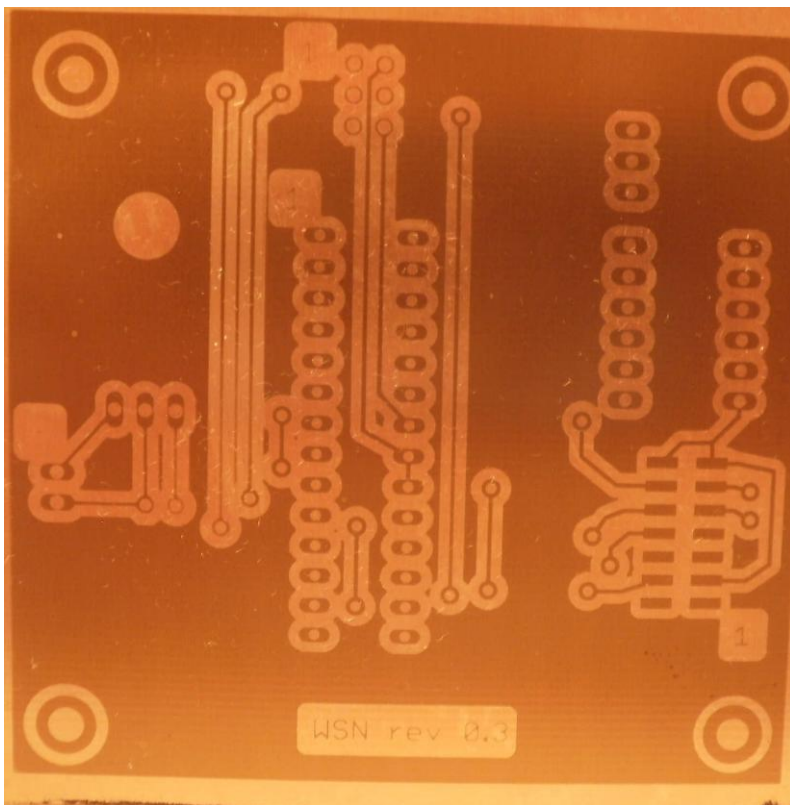
Εικόνα 5.16.



**Εικόνα 5.16** Μεταφορά της μάσκας (επάνω όψη) στην πλακέτα

Στην τελική υλοποίηση των κόμβων, επιλέχθηκε η μέθοδος μεταφοράς της μάσκας επάνω στην πλακέτα με τη χρήση φωτοευαίσθητης στρώσης επάνω στις επιφάνειες του χαλκού. Σε αυτή τη μέθοδο εκτυπώνονται οι μάσκες σε διαφάνειες σε laser ή InkJet εκτυπωτή χρησιμοποιώντας τις κατάλληλες διαφάνειες. Κατόπιν οι διαφάνειες τοποθετούνται επάνω στην φωτοευαίσθητη επιφάνεια και εκτίθενται σε υπεριώδη ακτινοβολία. Ο χρόνος έκθεσης εξαρτάται από την ισχύ και την απόσταση της λάμπας (περίπου 10 λεπτά). Για την έκθεση των φωτοευαίσθητων πλακετών χρησιμοποιήθηκε λάμπα OSRAM ULTRA-VITALUX 300W. Τα σημεία τα οποία δεν καλύπτονται από τη μάσκα αλλοιώνονται από την υπεριώδη ακτινοβολία και με τη χρήση διαλύματος

υδροξειδίου του νατρίου (NaOH), σε αναλογία 10gr σε ένα λίτρο νερού, απομακρύνονται από την επιφάνεια. Στην Εικόνα 5.17 παρουσιάζεται η επάνω όψη μετά την εμφάνισή της σε υδροξείδιο του νατρίου (NaOH).



**Εικόνα 5.17** Μεταφορά της μάσκας (επάνω όψη) σε φωτοευαίσθητη πλακέτα

### **Δημιουργία Τυπωμένου Κυκλώματος**

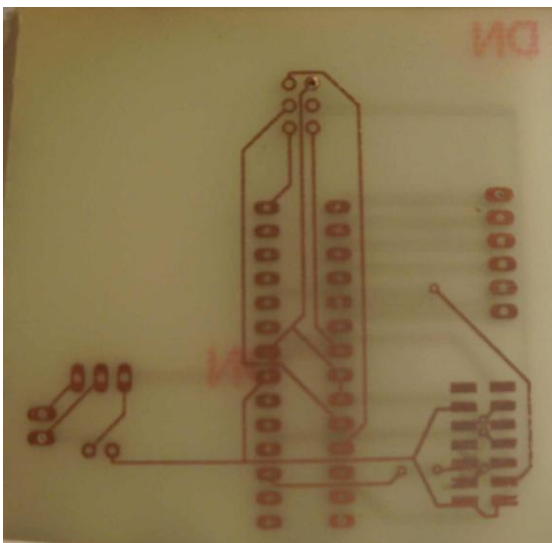
Μετά την μεταφορά της μάσκας επάνω στην πλακέτα ακολουθεί η διαδικασία της αποχάλκωσης. Για την αποχάλκωση χρησιμοποιήθηκε τριχλωριούχος σίδηρος ( $\text{FeCl}_3$ ) διαλυμένος σε ζεστό νερό. ΠΡΟΣΟΧΗ !!! . Ο τριχλωριούχος σίδηρος είναι πολύ τοξικός, εξαιρετικά διαβρωτικός και καυστικός. Πρέπει οπωσδήποτε να χρησιμοποιούμε πλαστικά γάντια. Ο χρόνος που χρειάζεται για να γίνει η αποχάλκωση εξαρτάται από την πυκνότητα του διαλύματος και τη θερμοκρασία. Καλό είναι να ανακινούμε το διάλυμα και να έχουμε μια σταθερή θερμοκρασία. Στην Εικόνα 5.18 φαίνεται η πακέτα σε διάλυμα τριχλωριούχου σιδήρου. Στην επάνω όψη, όπου δεν υπάρχει σχέδιο, ο χαλκός έχει φύγει και εμφανίζονται οι αγωγοί. Η άλλη πλευρά της πλακέτας δεν είναι

έτοιμη και χρειάζεται να την αναστρέψουμε για να επιταχυνθεί η αποχάλκωση. ΠΡΟΣΟΧΗ!!! Εκτός από την υψηλή τοξικότητά του, ο τριχλωριούχος σίδηρος, είναι παρά πολύ δύσκολο να απομακρυνθεί από αντικείμενα (ρούχα ή πορώδη αντικείμενα) με τα οποία θα έρθει σε επαφή, με αποτέλεσμα να τα καταστρέφει.



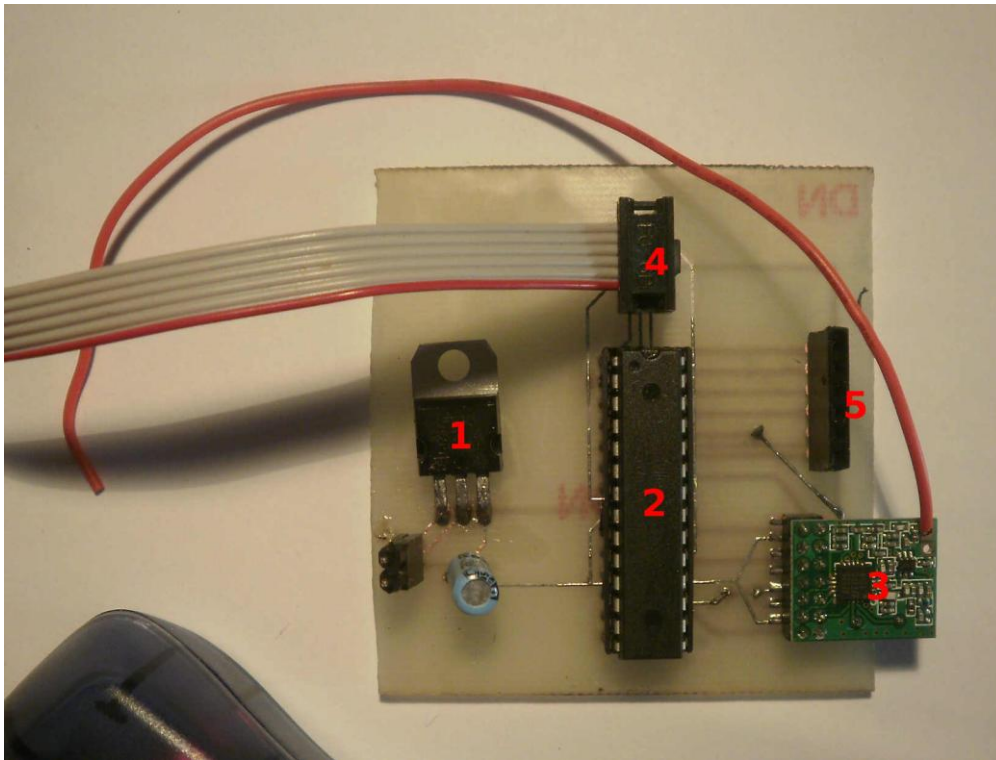
**Εικόνα 5.18** Διαδικασία αποχάλκωσης

Αφού ολοκληρωθεί η αποχάλκωση απομακρύνεται το τόνερ από την πλακέτα με ασετόν. Το αποτέλεσμα φαίνεται στην Εικόνα 5.19. Ακολουθεί το άνοιγμα των σπών και η τοποθέτηση των ολοκληρωμένων.



**Εικόνα 5.19** Τελική μορφή της πλακέτας (αρχική έκδοση υλικού 0.1)



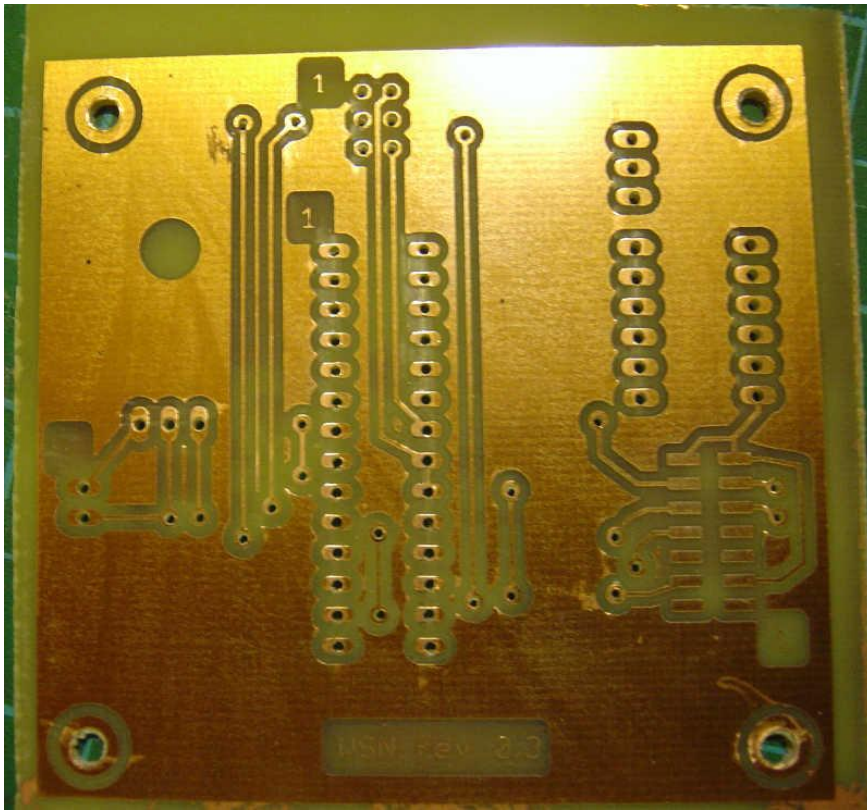


**Εικόνα 5.20** Απομακρυσμένος κόμβος (αρχική έκδοση υλικού 0.1)

Στην Εικόνα 5.20 φαίνεται η μορφή ενός απομακρυσμένου κόμβου (έκδοση υλικού 0.1).  
Ξεχωρίζουν τα ακόλουθα στοιχεία:

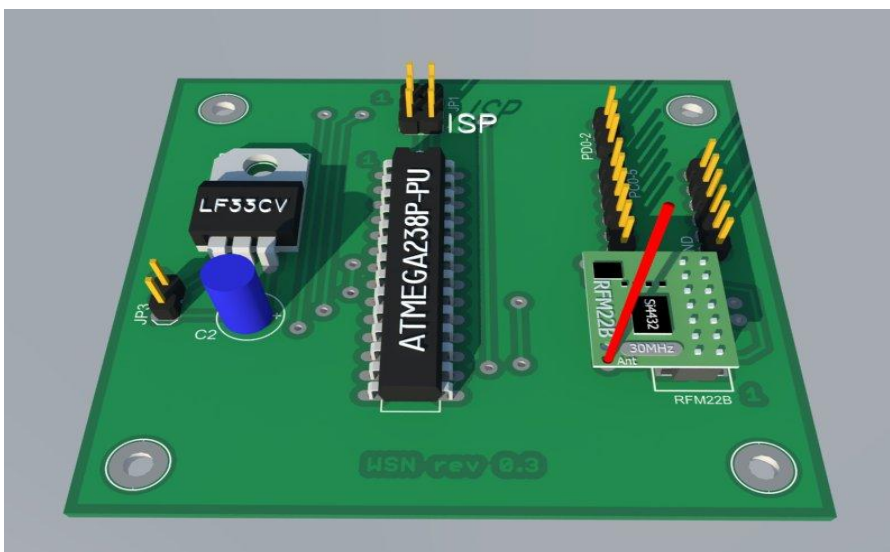
- (1) σταθεροποιητής τάσης 3,3V
- (2) μικροελεγκτής ATMEGA328P
- (3) Πομποδέκτης RFM22B
- (4) Θύρα προγραμματισμού ISP
- (5) Θύρες εισόδου/εξόδου μικροελεγκτή

Το υλικό των κόμβων σχεδιάστηκε αρκετές φορές και η τελευταία έκδοσή του είναι η 0.3. Στην έκδοση 0.3 για τη μεταφορά της μάσκας χρησιμοποιήθηκε φωτοευαίσθητη πλακέτα και υπεριώδης ακτινοβολία. Η εμφάνιση της μάσκας έγινε με υδροξείδιο του νατρίου. Η ποιότητα των πλακετών, της έκδοσης 0.3, είναι προφανώς ανώτερη της αρχικής έκδοσης Εικόνα 5.21.



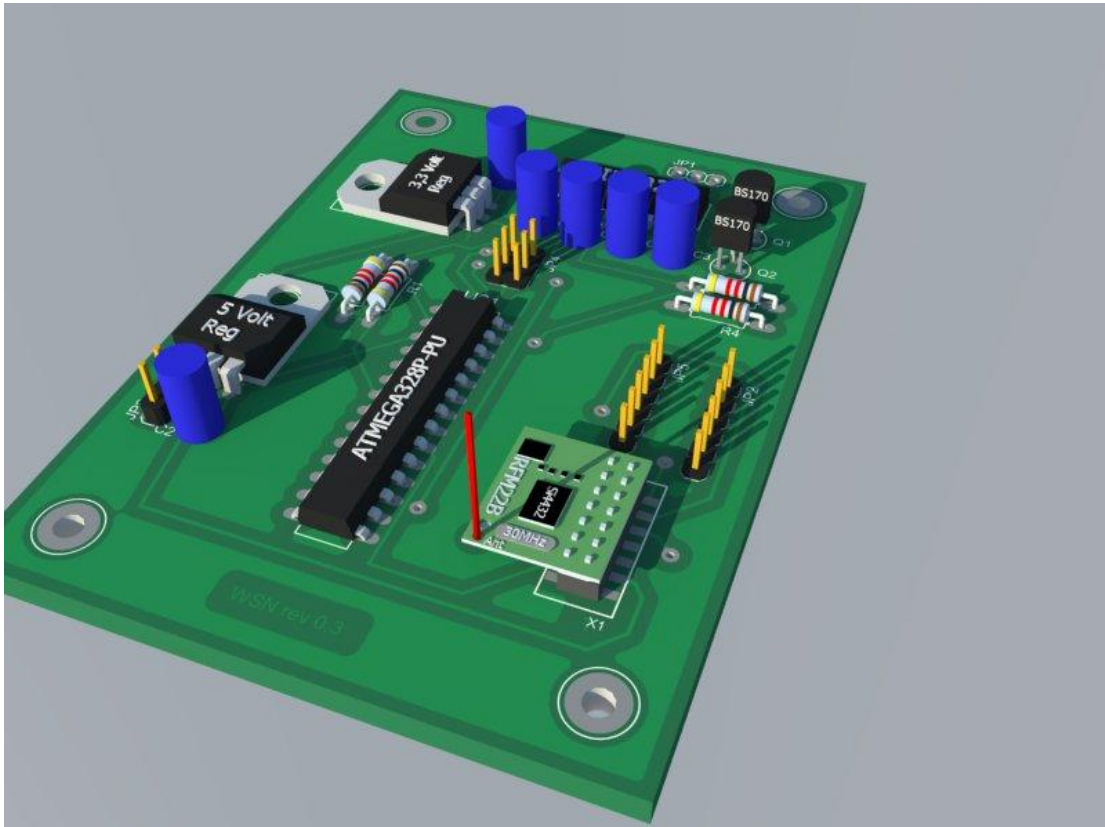
**Εικόνα 5.21** Απομακρυσμένος Κόμβος (Έκδοση υλικού 0.3)

Στην Εικόνα 5.22 φαίνεται ο απομακρυσμένος κόμβος ο οποίος σχεδιάστηκε σε EAGLE. Για τη δημιουργία του μοντέλου χρησιμοποιήθηκε το Google SketchUp [24].



**Εικόνα 5.22** Απομακρυσμένος Κόμβος 3D (Έκδοση υλικού 0.3)

Στην Εικόνα 5.23 φαίνεται ο κεντρικός κόμβος αφού σχεδιάστηκε σε EAGLE και έγινε εξαγωγή του μοντέλου σε Google SketchUp.



**Εικόνα 5.23** Κεντρικός Κόμβος σε τρισδιάστατη απεικόνιση (Έκδοση υλικού 0.3)

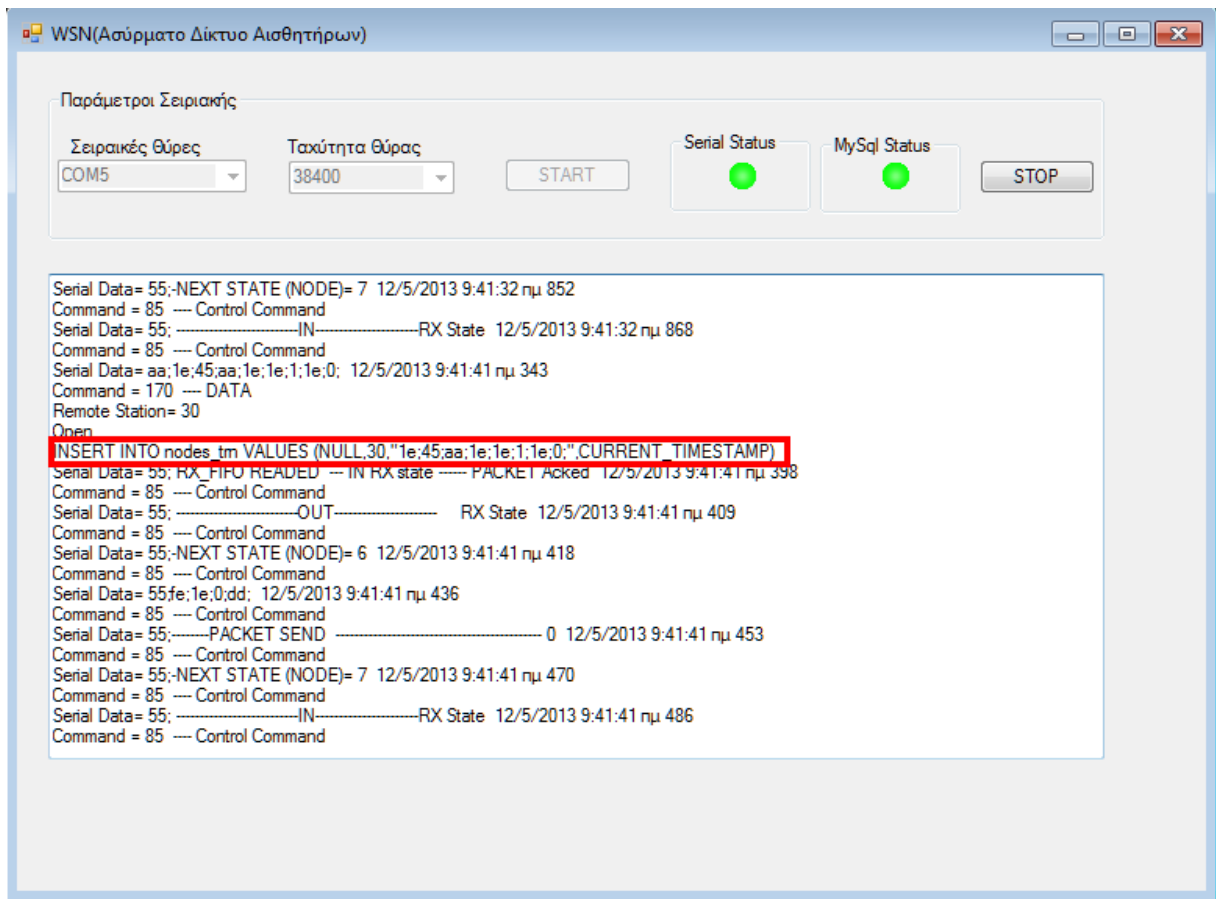
# Κεφάλαιο 6

## Επικοινωνία Κεντρικού Κόμβου με τη Βάση Δεδομένων

Ο κεντρικός κόμβος συλλέγει μετρήσεις από τους απομακρυσμένους κόμβους και επικοινωνεί με κεντρικό υπολογιστή ώστε να εισάγει τις μετρήσεις σε βάση δεδομένων. Για το σκοπό αυτό αναπτύχθηκε το κύκλωμα επικοινωνίας που περιγράφεται στο κεφάλαιο 5.

### 6.1 Αποστολή Δεδομένων σε Ηλεκτρονικό Υπολογιστή

Η σειριακή θύρα του μικροελεγκτή στέλνει συνεχώς δεδομένα στη σειριακή θύρα το Η/Υ. Στον Η/Υ εκτελείται ένα πρόγραμμα το οποίο διαβάζει συνεχώς τη σειριακή θύρα που του έχουμε ορίσει. Το πρόγραμμα είναι γραμμένο C++ (Visual Studio 8 της Microsoft) [76][25]. Η διεπαφή του προγράμματος φαίνεται στην Εικόνα 6.1.



**Εικόνα 6.1** Πρόγραμμα ανάγνωσης σειριακής θύρας

Στο πρόγραμμα ορίζουμε τη σειριακή θύρα που θέλουμε να διαβάσουμε. Η λίστα επιλογών με τις σειριακές θύρες γεμίζει αυτόματα με τις διαθέσιμες θύρες που υπάρχουν στο σύστημα. Η ταχύτητα επικοινωνίας έχει οριστεί στα 38.400 bps. Ορίζοντας αυτές τις παραμέτρους επιλέγουμε το κουμπί “START”. Οι πράσινες ενδείξεις μας πληροφορούν για την κατάσταση των συνδέσεων του προγράμματος με τη σειριακή θύρα και τη βάση δεδομένων MySql. Το πρόγραμμα μόλις δεχθεί δεδομένα στη σειριακή θύρα ενεργοποιεί τη ρουτίνα διακοπής η οποία εκτελεί τις παρακάτω ενέργειες:

- Εμφανίζει τα δεδομένα στην οθόνη, τον αριθμό του κόμβου μαζί με άλλες πληροφορίες (χρήσιμες κατά τη διάρκεια αποσφαλμάτωσης του συστήματος).
- Αποθηκεύει τις μετρήσεις σε αρχείο κειμένου.
- Εισάγει τις μετρήσεις σε βάση δεδομένων MySql. Το ερώτημα που εκτελείται φαίνεται στην Εικόνα 6.1 σε κόκκινο πλαίσιο.

### 6.1.1 Ερώτημα SQL- Πρόγραμμα Επικοινωνίας

Μόλις το πρόγραμμα επικοινωνίας λάβει τα δεδομένα από τη σειριακή θύρα, δημιουργεί ένα ερώτημα INSERT και το στέλνει στη βάση δεδομένων MySql . Τα πεδία του ερωτήματος είναι:

Πεδίο 1	Πεδίο 2	Πεδίο 3	Πεδίο 4
NULL	Αριθμός Κόμβου	String με τις μετρήσεις του κόμβου	TIME_STAMP

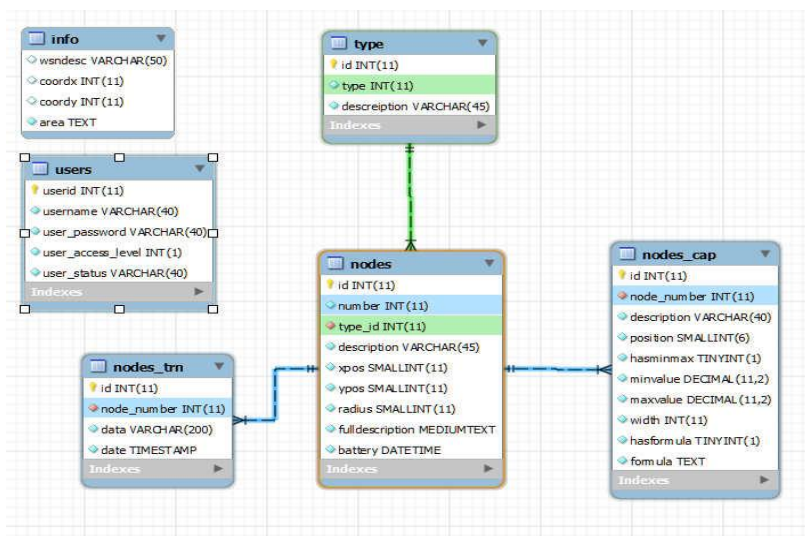
Πίνακας 6.3 Πεδία Ερωτήματος εισαγωγής μετρήσεων σε βάση δεδομένων

Παράδειγμα INSERT: INSERT INTO nodes\_trn

VALUES(NULL,15,"f;b6;aa;1f;1e;17;0;",CURRENT\_TIMESTAMP).

## 6.2 Δομή της Βάσης Δεδομένων

Η βάση δεδομένων δημιουργήθηκε με το MySQL Workbench v.5.2.46 και αποτελείται από τους πίνακες που φαίνονται στην Εικόνα 6.2 .



### Εικόνα 6.2 Σχήμα της Βάσης Δεδομένων

Ο πίνακας info έχει γενικές πληροφορίες για το δίκτυο αισθητήρων. Ο πίνακας users περιέχει τα ονόματα των χρηστών και τους κωδικούς τους καθώς και το επίπεδο πρόσβασης που έχουν στη βάση δεδομένων. Ένας χρήστης μπορεί να έχει δικαίωμα μόνο να προβάλλει τα δεδομένα και να μην μπορεί να αλλάξει τα στοιχεία του κόμβου.

Ο Πίνακας type έχει τους τύπους των κόμβων. Αυτή τη στιγμή δύο τύποι κόμβων υπάρχουν καταχωρημένοι: 1 (MASTER) και 2(NODE).

Ο πίνακας node περιέχει τους κόμβους του συστήματος και αποτελείται από τα πεδία: number (αριθμός κόμβου), type\_id(τύπος κόμβου), description(περιγραφή), xpros(συντεταγμένη X), ypros(συντεταγμένη Y), radius(ακτίνα για τη δημιουργία image map), fulldescription(αναλυτική περιγραφή), battery(ημερομηνία/ώρα αλλαγής μπαταρίας).

Ο πίνακας node\_cap περιέχει τις δυνατότητες που έχει κάποιος κόμβος. Αυτός ο πίνακας χρησιμοποιείται για τη δημιουργία του πίνακα μετρήσεων στη οθόνη των μετρήσεων. Αποτελείται από τα πεδία: node\_number(αριθμός κόμβου), description (περιγραφή που εμφανίζεται στον πίνακα μετρήσεων), position (θέση εμφάνισης στον πίνακα μετρήσεων ) hasminmax (εάν έχει μέγιστο και ελάχιστο), minvalue(ελάχιστη τιμή), maxvalue(μέγιστη τιμή), width(πλάτος στήλης), hasformula(εάν η τιμή προς εμφάνιση υπολογίζεται με τύπο), formula(τύπος υπολογισμού της τιμής προς εμφάνιση). Ο τύπος υπολογισμού έχει τη μορφή:  $rn * 0.5 - 64$ . Όπου rn είναι η τιμή που είναι καταχωρημένη στη βάση δεδομένων. Η τιμή που προκύπτει από τον υπολογισμό του παραπάνω τύπου εμφανίζεται στην οθόνη μετρήσεων. Ο συγκεκριμένος τύπος υπολογίζει τη θερμοκρασία (αν  $rn=182$  τότε η θερμοκρασία είναι:  $182*0,5-64 = 27$ ).

# Κεφάλαιο 7

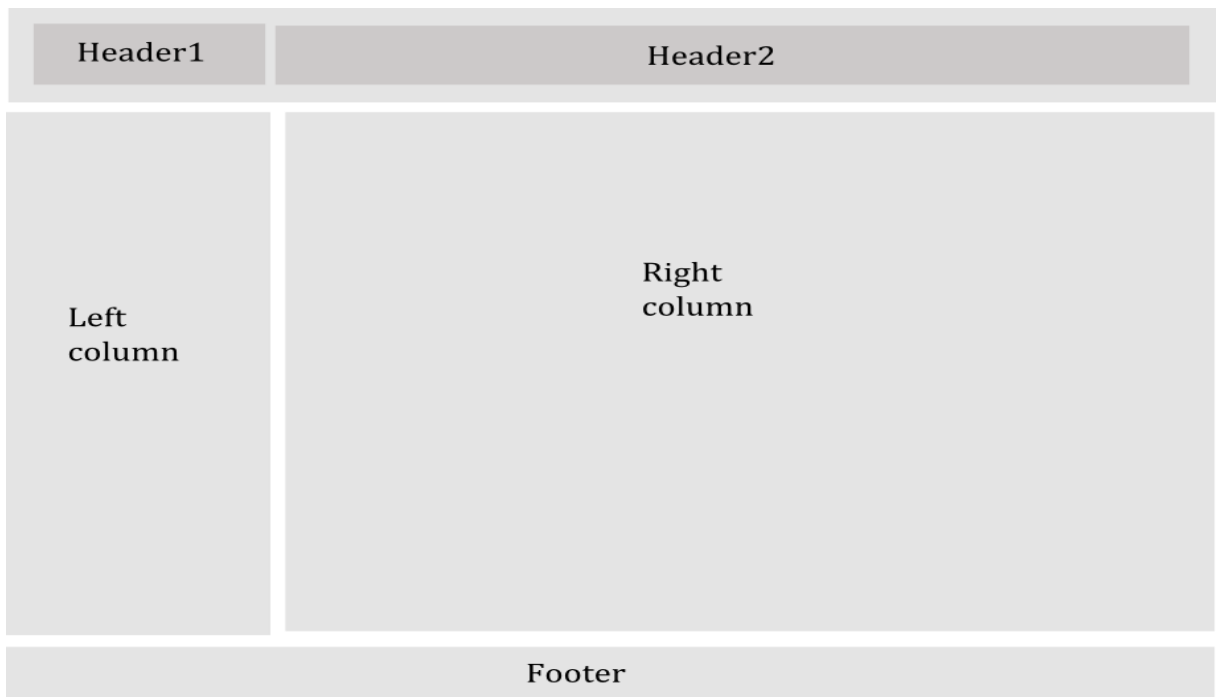
## Διεπαφή Χρήστη

Ο χρήστης μπορεί να δει τις μετρήσεις του ΑΔΑ μέσω μιας διεπαφής στον υπολογιστή όπου καταχωρούνται οι μετρήσεις ή μέσω διαδικτύου αν υπάρχει σύνδεση του ηλεκτρονικού υπολογιστή με το διαδίκτυο. Η διεπαφή έχει αναπτυχθεί σε PHP [26] και χρησιμοποιεί JavaScript [27][28] για να ενημερώνει πεδία της οθόνης. Επίσης υπάρχει η δυνατότητα γραφικής απεικόνισης των μετρήσεων καθώς και γραφική απεικόνιση των κόμβων.

### 7.1 Οθόνη Ασύρματου Δικτύου Μετρήσεων

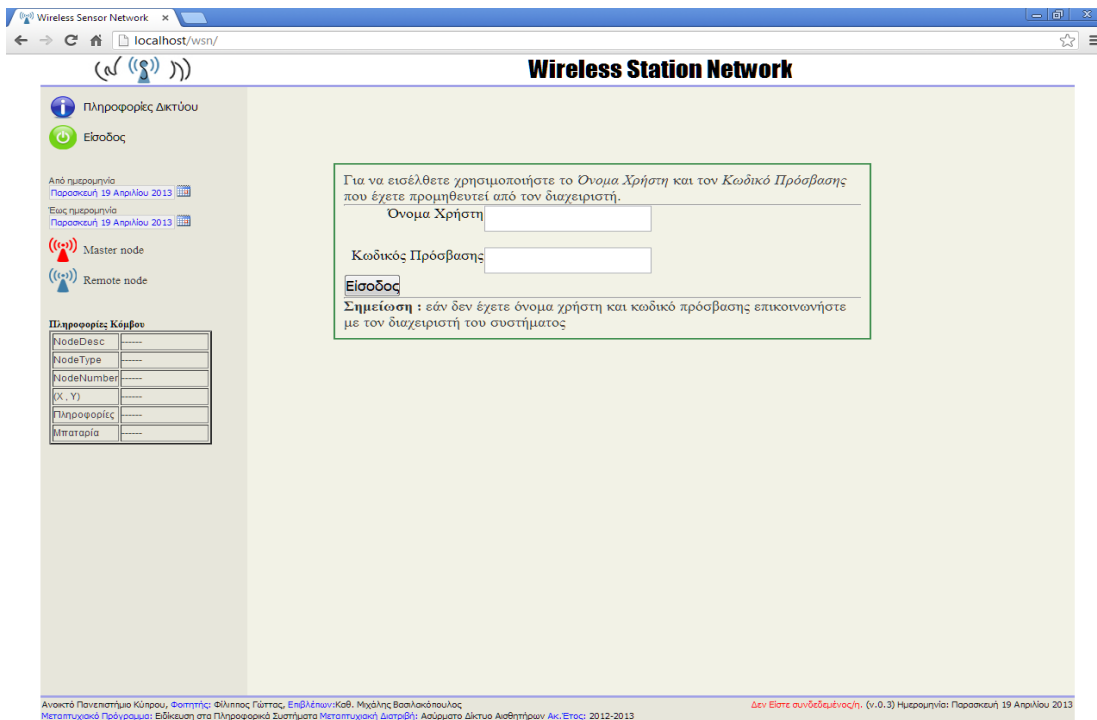
Η διεπαφή χρήστη έχει τη μορφή που φαίνεται στην Εικόνα 7.1. Κάθε σελίδα αποτελείται Header(Κεφαλίδα) + Περιεχόμενο + Footer(υποσέλιδο). Η κεφαλίδα και το υποσέλιδο παραμένουν σταθερά σε κάθε σελίδα της εφαρμογής. Στο header1 εμφανίζεται ένα λογότυπο και στο header2 η περιγραφή του συστήματος. Στο υποσέλιδο εμφανίζονται στοιχεία που αφορούν τη μεταπτυχιακή διατριβή.





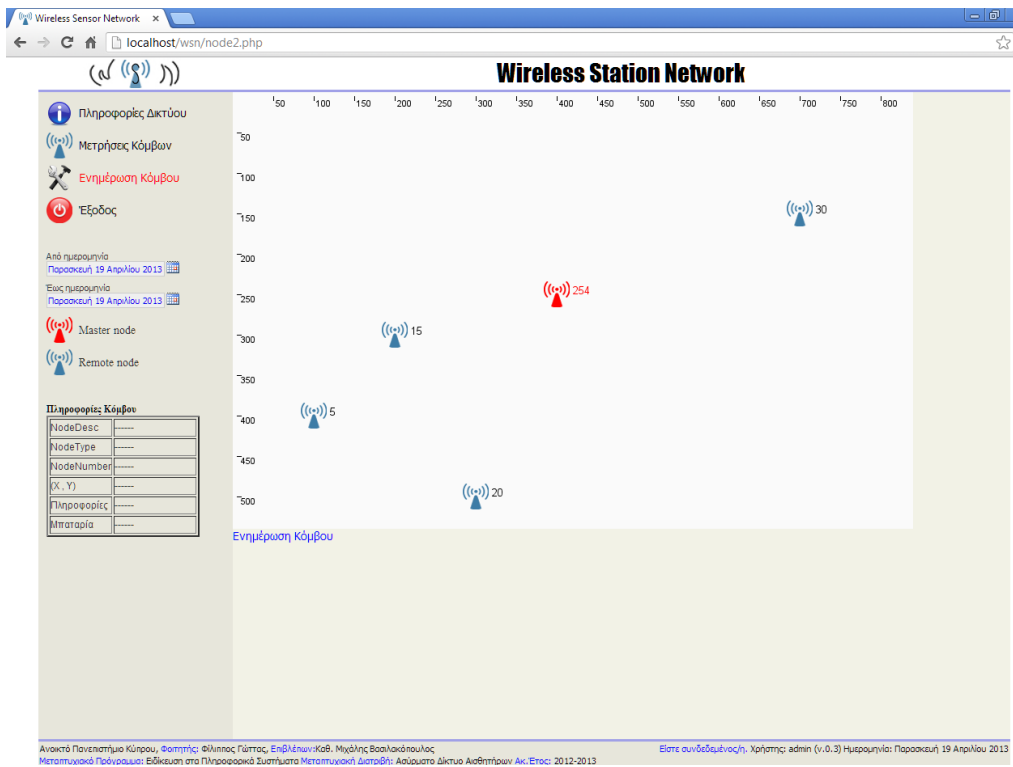
**Εικόνα 7.1** Δομή Διεπαφής Χρήστη

Με την είσοδο του χρήστη στην εφαρμογή εμφανίζεται η οθόνη της Εικόνα 7.2. Ο χρήστης πρέπει να εισάγει όνομα χρήστη και κωδικό πρόσβασης για να εισέλθει. Στην αριστερή πλευρά της οθόνης εμφανίζονται οι επιλογές: Πληροφορίες Δικτύου, «Είσοδος» καθώς και οι επιλογές της ημερομηνίας για τις οποίες θέλουμε να δούμε τις μετρήσεις. Ακολουθεί το υπόμνημα για τους κόμβους και πληροφορίες για τον κόμβο τον οποίο επιλέγουμε με ένα κλικ με το ποντίκι. Στην αρχική οθόνη οι επιλογές «Από Ημερομηνία» και «Έως ημερομηνία» έχουν την τρέχουσα ημερομηνία του συστήματος. Μόλις εισέλθουμε στην εφαρμογή, εμφανίζονται οι πληροφορίες του δικτύου και ανάλογα με τα δικαιώματα που έχει ο χρήστης εμφανίζεται στα αριστερά το αντίστοιχο μενού επιλογών. Η μορφοποίηση της οθόνης γίνεται με αρχείο αλληλουχίας φύλλων στυλ (CSS style) και είναι παραμετροποιήσιμη [29].





Εικόνα 7.2 Αρχική οθόνη Χρήστη

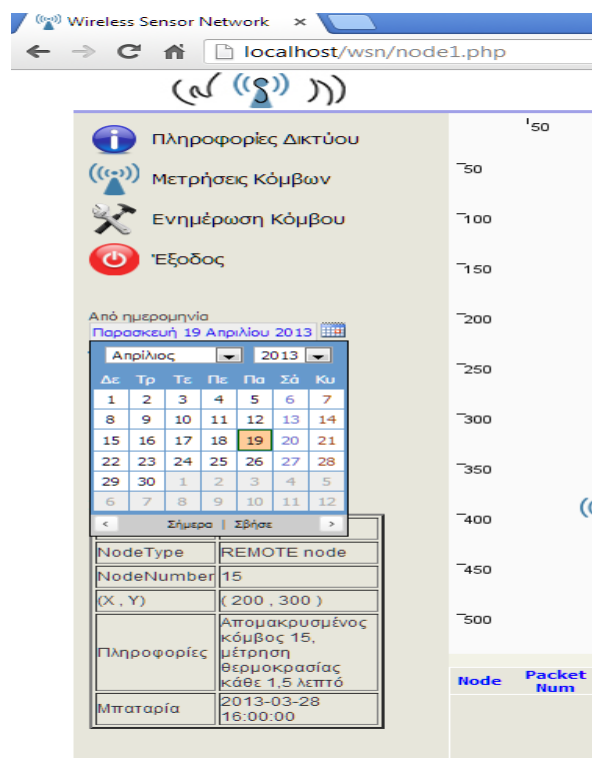
Εάν επιλεγθεί η επιλογή «Μετρήσεις Κόμβων» εμφανίζεται η οθόνη που φαίνεται στην Εικόνα 7.3.



Εικόνα 7.3 Είσοδος του Χρήστη στην εφαρμογή

Στην Εικόνα 7.3 εμφανίζονται οι κόμβοι του ασύρματου δικτύου τοποθετημένοι σε μια περιοχή 800x500 μονάδων μέτρησης. Οι απλοί κόμβοι μετρήσεων (Remote Node) έχουν γαλάζιο χρώμα  και κεντρικός κόμβος (Master Node) έχει κόκκινο χρώμα . Δίπλα σε κάθε κόμβο εμφανίζεται ο αριθμός του κόμβου. Ο κεντρικός κόμβος έχει τον αριθμό 254 και ο αριθμός 255 έχει κρατηθεί ώστε να είναι δυνατόν να υποστηριχθούν μηνύματα broadcast.

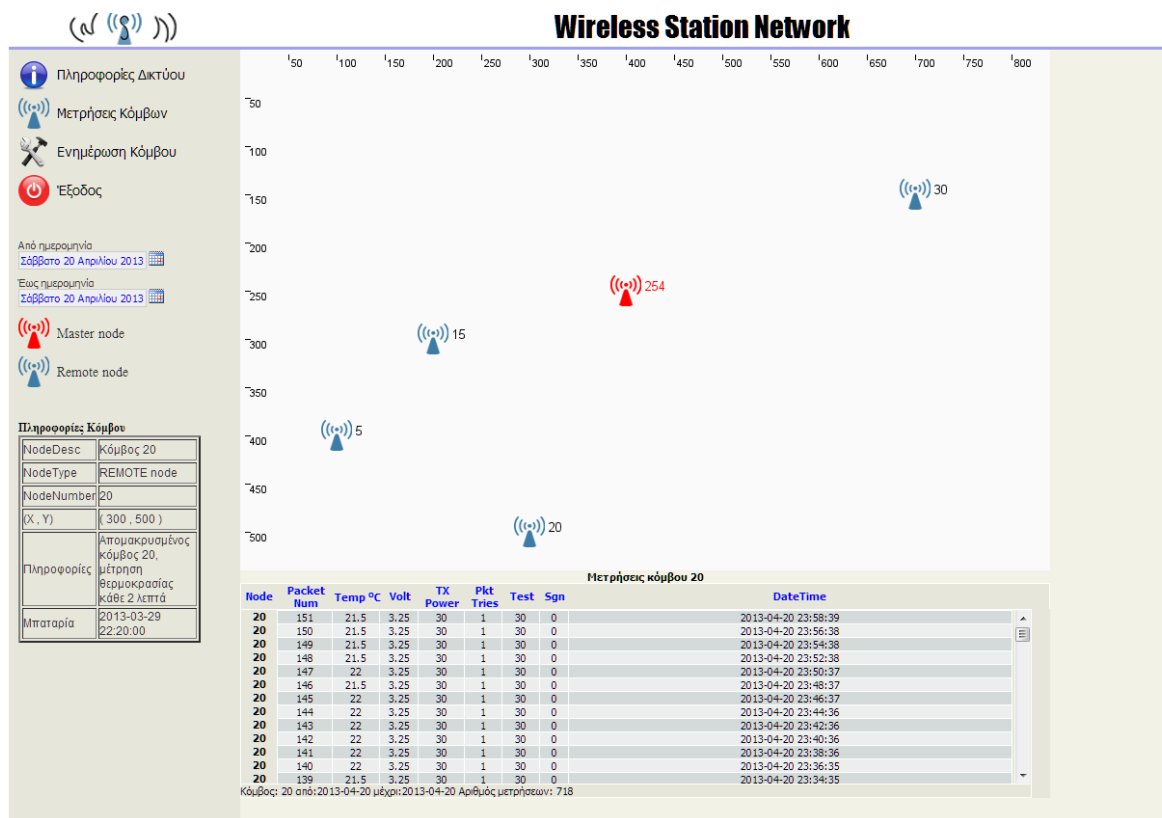
Η επιλογή ημερομηνιών γίνεται από στην αριστερή πλευρά των επιλογών όπου επιλέγουμε τις ημερομηνίες για τις οποίες θέλουμε να δούμε τις μετρήσεις των κόμβων Εικόνα 7.4.



**Εικόνα 7.4** Παράμετροι Διεπαφής

## 7.2 Οθόνη Μετρήσεων

Αφού επιλέξουμε τις ημερομηνίες για τις οποίες θέλουμε να δούμε τις μετρήσεις, κάνοντας ένα κλικ με το ποντίκι επάνω σε κάποιο κόμβο, στο κάτω μέρος της οθόνης εμφανίζονται οι μετρήσεις του συγκεκριμένου κόμβου Εικόνα 7.5. Όλοι οι κόμβοι έχουν αισθητήριο θερμοκρασίας. Στην αριστερή στήλη εμφανίζεται ο αριθμός του κόμβου, ο αριθμός του πακέτου, η θερμοκρασία, η τάση της πηγής, και η ημερομηνία και ώρα της μέτρησης. Στο κάτω μέρος των μετρήσεων εμφανίζονται πληροφορίες για το χρονικό διάστημα των μετρήσεων και το πλήθος των μετρήσεων.



Εικόνα 7.5 Μετρήσεις κόμβου 20

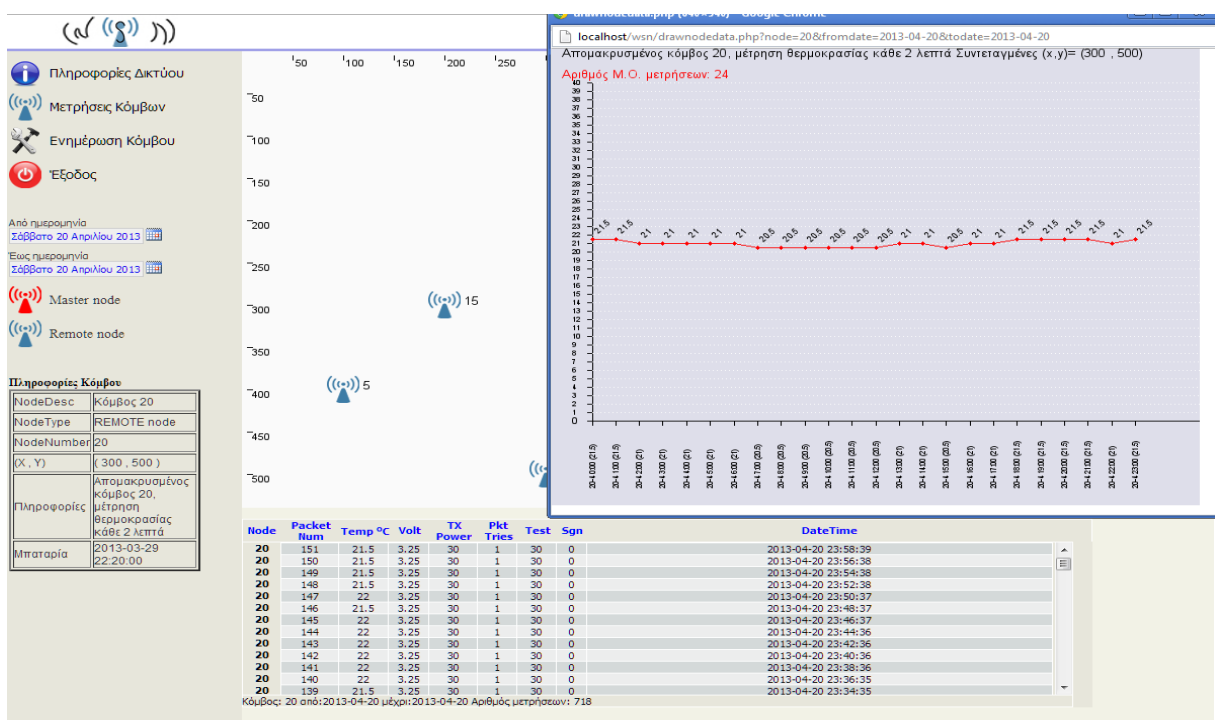
Ο πίνακας των μετρήσεων δημιουργείται δυναμικά και μπορεί να είναι διαφορετικός για κάθε κόμβο μετρήσεων. Η μορφοποίησή του γίνεται με CSS style sheets [30] και ο ορισμός του ύψους, του πλάτους, της στοίχισης, του μεγέθους των γραμμάτων και των χρωμάτων είναι εύκολα παραμετροποιήσιμα. Για την ενημέρωση του πίνακα χρησιμοποιήθηκε η τεχνολογία Ajax [31], ώστε η ενημέρωσή του να γίνεται χωρίς να

φορτώνεται όλη η σελίδα ξανά. Τα κοινά πεδία είναι οι στήλες: Node (Αριθμός Κόμβου), Packet Num (ο α/α του πακέτου) και Date Time(ημερομηνία και ώρα της μέτρησης). Ο πίνακας node\_cap περιέχει τις απαραίτητες πληροφορίες για κάθε στήλη δηλαδή πως θα εμφανιστεί, τι μήκος θα έχει και με ποια σειρά θα εμφανιστεί.

## 7.2.1 Γραφική Παράσταση Μετρήσεων

Κάνοντας διπλό κλικ επάνω στον κόμβο εμφανίζεται παράθυρο με την γραφική παράσταση της θερμοκρασίας του κόμβου Εικόνα 7.6. Στη γραφική παράσταση δεν εμφανίζονται όλες οι μετρήσεις για το χρονικό διάστημα που ορίζουμε αλλά ο μέσος όρος ανά ώρα. Παρατηρούμε, όπως προκύπτει από την Εικόνα 7.6, στη συγκεκριμένη ημερομηνία ότι υπάρχουν 718 μετρήσεις ενώ στο διάγραμμα εμφανίζονται 24 σημεία. Τα σημεία αυτά είναι ο μέσος όρος θερμοκρασίας ανά ώρα. Ο κόμβος 20 μετρά τη θερμοκρασία κάθε 2 λεπτά. Για μία ημέρα, που έχει 24 ώρες X 60 λεπτά/ώρα=1440 λεπτά, θα πρέπει να έχει περίπου  $1440/2 = 720$  μετρήσεις.

Για τη γραφική παράσταση χρησιμοποιήθηκε η βιβλιοθήκη GD της PHP, που δίνει τη δυνατότητα δημιουργίας και χειρισμού εικόνων. Για να μπορεί να εκτελείται σωστά η διεπαφή χρήστη και για να εμφανίζει τις γραφικές απεικονίσεις, θα πρέπει να είναι εγκατεστημένη η βιβλιοθήκη GD στην PHP του διακομιστή.

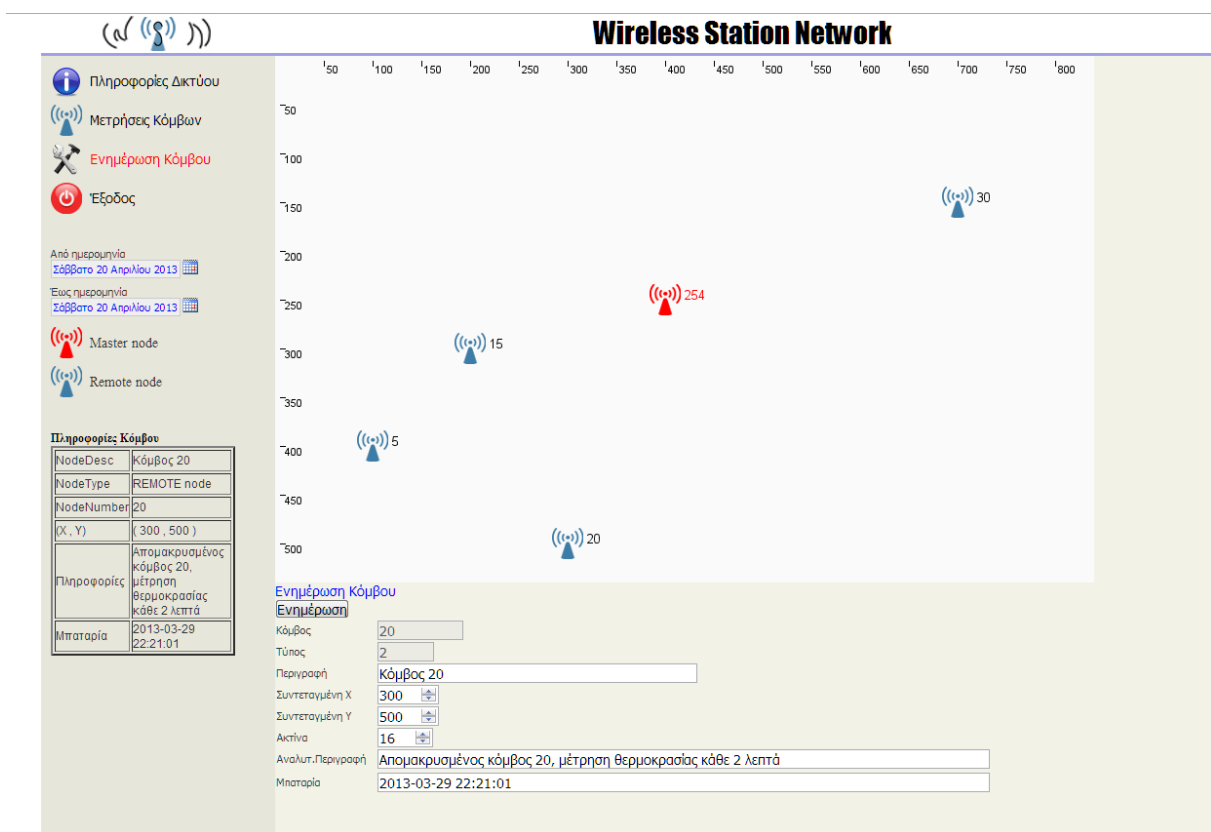


Εικόνα 7.6 Γραφική απεικόνιση μετρήσεων κόμβου 20

Το ελάχιστο και το μέγιστο της γραφικής παράστασης είναι παράμετροι στον πίνακα nodes\_cap και χρησιμοποιούνται για την δημιουργία του άξονα y. Επίσης στον πίνακα nodes\_cap ορίζονται το πλάτος και το ύψος της γραφικής παράστασης.

### 7.3 Ενημέρωση Κόμβου

Από την επιλογή «Ενημέρωση Κόμβου» μπορούμε να αλλάξουμε κάποιες από τις παραμέτρους του κόμβου όπως την Περιγραφή, Συντεταγμένη X, Συντεταγμένη Y, Ακτίνα, αναλυτική Περιγραφή και ημερομηνία και ώρα αλλαγής της μπαταρίας Εικόνα 7.7.



Εικόνα 7.7 Ενημέρωση παραμέτρων κόμβου 20

Στην παραπάνω φόρμα εισαγωγής δεν μπορούμε να αλλάξουμε τον αριθμό του κόμβου καθώς και τον τύπου του. Οι παράμετροι αυτοί ορίζονται κατά το σχεδιασμό του

δικτύου και επηρεάζουν τις εγγραφές των μετρήσεων στη βάση δεδομένων. Ο τύπος του κόμβου είναι χαρακτηριστικό που δεν μπορεί να αλλάξει χωρίς να αλλάξει και το λογισμικό του κόμβου και είναι κρίσιμη παράμετρος του δικτύου. Αλλάζοντας τις συντεταγμένες οι κόμβοι μετακινούνται στο χώρο που ορίζεται από λευκό πλαίσιο των 800x500 μονάδων. Στο πεδίο μπαταρία καταγράφεται η ημερομηνία και η ώρα αλλαγής της μπαταρίας.

# Κεφάλαιο 8

## Επίλογος

Η παρούσα μεταπτυχιακή διατριβή διερεύνησε την υλοποίηση ενός ασύρματου δικτύου αισθητήρων με στόχο την όσο το δυνατόν απλή και οικονομική κατασκευή. Η εύκολη υλοποίηση με συμβατικά υλικά μας δίνει την δυνατότητα δημιουργίας ενός εργαλείου έρευνας και πειραματισμού ευέλικτου με χαμηλό κόστος. Η υλοποίηση του ασύρματου δικτύου αισθητήρων μπορεί εύκολα να πραγματοποιηθεί σε πλακέτες δοκιμών ώστε να γίνονται αλλαγές και προσθήκες χωρίς να χρειάζεται επανασχεδιασμός όλου του συστήματος. Η δυνατότητα αυτή προσφέρει ευελιξία στον σχεδιασμό και ελευθερία στην επιλογή των αισθητήρων που θα συνδέονται στους ασύρματους κόμβους. Το πρωτόκολλο επικοινωνίας που αναπτύχθηκε μπορεί να προγραμματιστεί σε μικροελεγκτές με μικρές απαιτήσεις σε ταχύτητα και μνήμη. Οι μικροελεγκτές αυτοί, έχουν χαμηλό κόστος και μπορούν να χρησιμοποιηθούν εύκολα σε πλακέτες δοκιμών. Στην παρούσα μεταπτυχιακή διατριβή χρησιμοποιήθηκε ο μικροελεγκτής της ATMEL ATMEGA328P, που διατίθεται σε συσκευασία DIP, καθώς και ο πομποδέκτης της HOPE RFM22B βασισμένος στο ολοκληρωμένο Si4432 της Silicon Labs.

Εμπορικοί κόμβοι ασύρματων δικτύων χρησιμοποιούν παρόμοια ολοκληρωμένα με μεγαλύτερη μνήμη και υψηλότερο χρονισμό και υλοποιούν πρωτόκολλα τα οποία



προϋποθέτουν ιδιόκτητες πλατφόρμες ανάπτυξης και κάρτες επέκτασης, ανεβάζοντας έτσι κατά πολύ το κόστος. Επίσης, ο ερευνητής δεσμεύεται από τη συγκεκριμένη τεχνολογία του κατασκευαστή περιορίζοντας τις επιλογές του σε εργαλεία ανάπτυξης και αισθητήρες μέτρησης φυσικών μεγεθών.

Στην μεταπτυχιακή διατριβή δόθηκε μεγάλη προσοχή ώστε να χρησιμοποιηθούν, όσο ήταν δυνατόν, εργαλεία ανάπτυξης ελεύθερου λογισμικού. Το πρόγραμμα των κόμβων είναι γραμμένο σε C και έτσι μπορεί εύκολα να μεταφερθεί σε άλλους μικροελεγκτές που μπορούν να προγραμματιστούν με γλώσσα υψηλού επιπέδου, όπως οι μικροελεγκτές της Microchip. Η βάση δεδομένων είναι η MySQL και η διεπαφή χρήστη είναι γραμμένη σε PHP και εύκολα μεταφέρονται σε άλλες πλατφόρμες. Ο διακομιστής που στηρίζει την εφαρμογή είναι ο Apache.

Η ανάπτυξη ενός συστήματος ασύρματων κόμβων μετρήσεων αγγίζει πάρα πολλούς τεχνολογικούς τομείς. Η επιλογή του υλικού ήταν δύσκολη και η εύρεση των ολοκληρωμένων πολλές φορές επιβράδυνε την ανάπτυξη και τη μελέτη του συστήματος. Σε επίπεδο υλικού χρειάστηκε να συνεργαστούν μέρη με διαφορετικά λογικά επίπεδα και διαφορετικές τεχνολογίες όπως: τα λογικά επίπεδα της σειριακής θύρας του υπολογιστή με τον μικροελεγκτή, ο πομποδέκτης σε συσκευασία των 2mm με τα ολοκληρωμένα σε συσκευασίες των 2,54mm. Η κατασκευή των πλακετών έγινε με διάφορους τρόπους όπως θερμική μεταφορά της μάσκας και μεταφορά με υπεριώδη ακτινοβολία. Ανάλογα με το στάδιο ανάπτυξης του συστήματος επιλέχθηκαν οι προσφορότεροι τρόποι κατασκευής και αυτοί που θα μπορούσαν να υλοποιηθούν με τον υπάρχοντα τεχνικό εξοπλισμό. Σε επίπεδο λογισμικού οι προκλήσεις ήταν πάρα πολλές διότι απαιτήθηκε να αναπτυχθούν πολλά διαφορετικά προγράμματα που έπρεπε να συνεργαστούν για να δώσουν το τελικό αποτέλεσμα. Προγραμματισμός σε C απαιτήθηκε για το λογισμικό επικοινωνίας του μικροελεγκτή και της σειριακής θύρας καθώς και για τον προγραμματισμό του μικροελεγκτή. Ανάπτυξη προγραμμάτων σε PHP και JavaScript χρειάστηκε για την ανάπτυξη της διεπαφής χρήστη και για την επικοινωνία με την βάση δεδομένων. Χρησιμοποιήθηκε η βιβλιοθήκη GD (Graphics Draw) της PHP για τη δημιουργία δυναμικών γραφικών καθώς και για τη δημιουργία γραφημάτων. Για την υποστήριξη της διεπαφής εγκαταστάθηκε η βάση δεδομένων MySQL και ο διακομιστής Apache.

Για την υλοποίηση ενός δικτύου ασύρματου δικτύου κόμβων μετρήσεων θα μπορούσε να χρησιμοποιηθεί και η γνωστή πλατφόρμα “Arduino” η οποία χρησιμοποιεί τον ίδιο επεξεργαστή και μπορεί να προγραμματιστεί σε C [32]. Μια τέτοια επιλογή αυξάνει πολύ το κόστος, επειδή τα επιμέρους στοιχεία όπως πομποδέκτης (ο οποίος έχει το μεγαλύτερο κόστος), μπαταρίες και επιπλέον αισθητήρες είναι κοινά και στις δύο προσεγγίσεις. Το κόστος κατασκευής του ασύρματου δικτύου της μεταπτυχιακής διατριβής είναι τουλάχιστον τέσσερις έως πέντε φορές μικρότερο από την πλατφόρμα “Arduino”.

## Βιβλιογραφία

- [01] L. Doherty, J. Simon, T. Watteyne ‘Wireless Sensor Network Challenges and Solutions’, Microwave Journal, 55, 22-34,2012
- [02] V. Madan, S. Reddy ‘Review of Wireless Sensor Mote Platforms’ VSRD International Journal of Electrical, Electronics & Comm. Engg. Vol. 2 (2), 2012,pp 50-55
- [03] K. Sohraby, D. Minoli, T. Znati ‘Wireless sensor networks: technology, protocols, and applications’, John Wiley & Sons, New Jersey,2007
- [04] A. Garcia-Hernando, J.Fernan, M. Ortega et al ‘Problem Solving for Wireless Sensor Networks’, Springer-Verlag, London, 2008
- [05] Silicon Laboratories Inc ‘Key Priorities for Sub-GHz Wireless Deployment’, διαθέσιμο: <http://www.silabs.com/Support%20Documents/TechnicalDocs/Key-Priorities-for-Sub-GHz-Wireless-Deployments.pdf>
- [06] I.Demirkol, C. Ersoy, and F. Alagöz ‘MAC Protocols for Wireless Sensor Networks: a Survey’, IEEE Communications Magazine, April 2006
- [07] W. Ye, J. Heidemann, D. Estrin ‘An Energy-Efficient MAC protocol for Wireless Sensor Networks’, In Proceedings of the IEEE Infocom, pp. 1567-1576. New York, NY, USA, USC/Information Sciences Institute, IEEE. June, 2002
- [08] A. El-Hoiyi, J.-D. Decotignie, and J. Hernandez ‘Low power MAC protocols for infrastructure wireless sensor networks’, In Proceedings of the Fifth European Wireless Conference, Feb. 2004.
- [09] C. C. Enz, A. El-Hoiydi, J.-D. Decotignie, V. Peiris, ‘WiseNET: An Ultralow-Power Wireless Sensor Network Solution’, IEEE Computer, Volume: 37, Issue: 8, August 2004.
- [10] Silicon Labs, ‘AN550 EZMACPRO® PROGRAMMING GUIDE’, Application Note 550, Rev. 0.1 10/2010, Austin 2010, διαθέσιμο: <http://www.silabs.com/Support Documents/TechnicalDocs/AN550.pdf>

- [11] Silicon Labs 'EZMAC AND EZHOP USER'S GUIDE', Rev. 3.2 6/12, Austin 2012, διαθέσιμο: [http://www.silabs.com/Support\\_Documents/TechnicalDocs/EZMac-EZHOp-UG.pdf](http://www.silabs.com/Support_Documents/TechnicalDocs/EZMac-EZHOp-UG.pdf)
- [12] J. Kurose, K. Ross 'Computer Networking A top Down Approach', Addison-Wesley, 2010
- [13] Silicon Labs, 'AN440 Si4430/31/32 Register Descriptions', Application Note, Rev.0.6 4/2013, Austin, 2013, διαθέσιμο: [http://www.silabs.com/Support\\_Documents/TechnicalDocs/AN440.pdf](http://www.silabs.com/Support_Documents/TechnicalDocs/AN440.pdf)
- [14] Silicon Labs, 'AN537-FIFO & DIRECT MODE, PACKET HANDLER OPERATION For EZRadio Pro', Application Note, Rev02, August 2011, διαθέσιμο: [http://www.silabs.com/Support\\_Documents/TechnicalDocs/AN537.pdf](http://www.silabs.com/Support_Documents/TechnicalDocs/AN537.pdf)
- [15] HOPERF ELECTRONIC, RFM22 Datasheet, διαθέσιμο: <http://www.hoperf.com/rf/module/fsk/RFM22B.htm>
- [16] J. Catsoulis 'Designing Embedded Hardware', O'Reilly, Sebastopol, 2005
- [17] Atmel '8-bit Atmel Microcontroller with 4/8/16/32K Bytes In-System Programmable Flash', Datasheet, San Jose 05/2011, διαθέσιμο: [http://www.atmel.com/Images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P\\_datasheet.pdf](http://www.atmel.com/Images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P_datasheet.pdf)
- [18] Atmel, 'AVR910-In System Programming', Application Note, San Jose, August 2008, διαθέσιμο: <http://www.atmel.com/Images/doc0943.pdf>
- [19] Intersil '+5V Powered RS-232 Transmitters/Receivers', September 2008, διαθέσιμο: <http://www.intersil.com/content/dam/Intersil/documents/fn31/fn3138.pdf>
- [20] K. Arnold 'Embedded Control Hardware Design', LLH Technology Publishing, 2001
- [21] H. Shutte, 'Bi-directional level shifter for I<sup>2</sup>C-bus and other Systems', Application Note AN97055, Philips Semiconductors Systems Laboratory Eindhoven, The Netherlands, August 1997, διαθέσιμο: <http://ics.nxp.com/support/documents/interface/pdf/an97055.pdf>

- [22] Fairchild, 'BS170/ MMBF170 N-Channel Enhancement Mode Field Effect Transistor', Datasheet, March 2010, διαθέσιμο: <http://www.fairchildsemi.com/ds/BS/BS170.pdf>
- [23] CadSoft Computer GmbH, 'Eagle Tutorial Version 6', διαθέσιμο: <http://www.cadsoftusa.com/training/tutorials>
- [24] S. Singh 'Beginning Google SketchUp for 3D Printing', Apress, 2011
- [25] I. Horton 'Beginning Visual C++ 2008', Wiley Publishing, Indianapolis, 2008
- [26] Q. Zervaas 'Practical Web 2.0 Applications with PHP', Apress, Berkeley, 2008
- [27] D.S. Malik, 'Java Programming: From Problem Analysis to Program Design', Course Technology, Boston, 2010
- [28] N. C. Zakas, 'High Performance JavaScript' O'Reilly Media, Sebastopol, 2010
- [29] C. Murphy, N. Persson 'HTML and CSS Web Standards Solutions: A Web Standardistas' Approach', friendsofED, Berkeley, December 2008
- [30] D. Powers 'Getting Started With CSS', friendsofED, Berkeley, 2009
- [31] S.Lauriat 'Advanced Ajax Architecture and Best Practices', Pearson Education, Boston, 2008
- [32] R. Faludi 'Building Wireless Sensor Networks', O'Reilly Media, Sebastopol, 2011

# Παράρτημα Α

## Ηλεκτρονικές Πηγές

### Πληροφόρησης

Στο παράρτημα Α δίνονται δικτυακοί τόποι οι οποίοι χρησιμοποιήθηκαν για άντληση πληροφοριών και περιέχουν πληροφορίες που μπορούν να βοηθήσουν τον αναγνώστη.

#### **A.1 Δικτυακοί Τόποι (Υλικό)**

<http://www.atmel.com/>

<http://www.hoperf.com/>

<http://www.silabs.com/support/pages/document-library.aspx>

<http://www.intersil.com/en.html>

<http://www.maximintegrated.com/>

<http://www.marelectronics.gr>

[http://en.wikipedia.org/wiki/List\\_of\\_wireless\\_sensor\\_nodes](http://en.wikipedia.org/wiki/List_of_wireless_sensor_nodes)

[http://en.wikipedia.org/wiki/Sensor\\_node](http://en.wikipedia.org/wiki/Sensor_node)

<http://www.arcelect.com/rs232.htm>

[http://en.wikibooks.org/wiki/Serial\\_Programming:RS-232\\_Connections](http://en.wikibooks.org/wiki/Serial_Programming:RS-232_Connections)

## **A.2 Δικτυακοί Τύποι (Λογισμικό )**

<http://www.atmel.com/tools/atmelstudio.aspx>

<http://msdn.microsoft.com/en-us/library/52f3sw5c%28v=vs.90%29.aspx>

<http://www.apachefriends.org/en/xampp.html>

<http://httpd.apache.org/>

<http://php.net/>

<http://www.mysql.com/>

<http://java.com/en/>

<http://www.w3schools.com/js/>

<http://php.net/manual/en/book.image.php>

# Παράρτημα Β

## Προγράμματα

Για την ολοκλήρωση της μεταπτυχιακής διατριβής γράφτηκαν προγράμματα σε διάφορες γλώσσες όπως AVR-C, C++, PHP, JavaScript, SQL, HTML. Στις παρακάτω ενότητες γίνεται αναφορά στις συναρτήσεις που χρησιμοποιήθηκαν.

### B.1 Προγράμματα Μικροελεγκτή

Τα προγράμματα για τον μικροελεγκτή γράφηκαν σε AVR-C και χρησιμοποιήθηκε το ολοκληρωμένο περιβάλλον προγραμματισμού ATMEL Studio 6 (Version 6.0.1996-Service Pack 2).

MCU\_init() : Αρχικοποίηση του μικροελεγκτή

USART\_init(): Αρχικοποίηση της σειριακής θύρας του μικροελεγκτή

USART\_putstr(char\* StringPtr,uint8\_t cr) : Στέλνει στη σειριακή θύρα του Υ/Η τα δεδομένα που δείχνει ο δείκτης StringPtr. Εάν το cr=1 στέλνει και αλλαγή γραμμής. Εάν το cr=0 δεν στέλνει αλλαγή γραμμής.

RFM22B\_init(): Αρχικοποιεί τις παραμέτρους του πομποδέκτη.



`enable_battery_threshold(unsigned int bat_threshold)`: Θέτει το κατώφλι του επιπέδου της τάσης της μπαταρίας.

`register_set_bit(unsigned int reg, uint8_t bit)`: Θέτει το bit του καταχωρητή `reg` που ορίζεται από το όρισμα `bit`.

`registers_print(uint8_t from_reg, uint8_t to_reg)`: Στέλνει στη σειριακή θύρα τους καταχωρητές από `from_reg` μέχρι και `to_reg` σε δεκαεξαδική και δυαδική μορφή.

`WriteRegister(unsigned int cmd)`: Γράφει στον καταχωρητή, που ορίζεται από το πρώτο byte του ορίσματος `cmd`, την τιμή του δεύτερου byte του `cmd`.

`ReadRegister(unsigned int cmd)`: Διαβάζει των καταχωρητή `cmd` και επιστρέφει το περιχόμενό του.

`Temperature_Sensor_Init(void)`: Αρχικοποιεί τον αισθητήρα θερμοκρασίας.

`clear_tx_fifo(void)`: Σβήνει τα περιεχόμενα της μνήμης TX FIFO.

`clear_Interrupt(void)`: Μηδενίζει τη σημαία των διακοπών.

`read_RX_FIFO(uint8_t vrs, uint8_t vcr)`: Διαβάζει την μνήμη RX FIFO. Η παράμετρος `vrs` είναι η ισχύς του λαμβανόμενου σήματος και η `vcr` δείχνει εάν έχει συμβεί CRC σφάλμα.

`IDLE_READY_state(unsigned int idle_time)`: Βάζει τον μικροελεκτή σε READY κατάσταση.

`RX_sync(void)`:

`packet_to_send(uint8_t pl, uint8_t packet_command, uint8_t destination)`: Ετοιμάζει τα δεδομένα στη μνήμη TX FIFO. Η παράμετρος `pl` είναι το μήκος του πακέτου, η παράμετρος `packet_command` παίρνει την τιμή 0 για πακέτα δεδομένων ή την τιμή 1 για πακέτα ελέγχου. Η παράμετρος `destination` αφορά το αναγνωριστικό (ID) του σταθμού στο δίκτυο.

`send(void)`: Ξεκινά τη διαδικασία αποστολής των δεδομένων.

`print_ItStatus(void)`: Ρουτίνα για αποσφαλμάτωση του προγράμματος.

`listen_before_talk(void)`: Υλοποιεί τη διαδικασία LBT (Listen Before Talk)

`to_ready_mode(void)`: Τοποθετεί τον πομποδέκτη στην κατάσταση Ready.

`RX_state(void)`: Τοποθετεί τον πομποδέκτη στην κατάσταση RX (λήψης).

`RX_packet_valid_crc_error(void)`: Ενεργοποιεί την διακοπή `packet_valid` (έγκυρο πακέτο) και την διακοπή `CRC_error` (σφάλμα CRC) τοποθετεί τον πομποδέκτη στην κατάσταση RX (λήψης).

to\_rx\_RSSI\_mode(): Ενεργοποιεί την διακοπή RSSI (Receive Signal Strength Indicator) και μεταβαίνει στην κατάσταση RX.

to\_tx\_mode(void): Τοποθετεί τον πομποδέκτη στην κατάσταση TX(μετάδοσης).

led\_on\_of(int times): Αναβοσβήνει ένα led times φορές.

Write0(void): Γράφει 0 στην πόρτα του μικροελεγκτή που αντιστοιχεί στο σήμα SDI, για την υλοποίηση της επικοινωνίας.

Write1(void): Γράφει 1 στην πόρτα του μικροελεγκτή που αντιστοιχεί στο σήμα SDI, για την υλοποίηση της επικοινωνίας.

my\_delay\_ms(long int n): Δημιουργεί μια καθυστέρηση n msec.

RFM\_States(): Υλοποιεί τη μηχανή καταστάσεων του συστήματος.

## **B.2 Προγράμματα Επικοινωνίας RS232 - MySQL**

Για να επικοινωνεί ο μικροελεγκτής με τον ηλεκτρονικό υπολογιστή, ώστε οι μετρήσεις να καταχωρούνται στη βάση δεδομένων, αναπτύχθηκε πρόγραμμα επικοινωνίας σε C++ με το ολοκληρωμένο περιβάλλον ανάπτυξης της Mikrossoft Visula Studio 2008.

findPorts(void): Βρίσκει τις διαθέσιμες σειριακές πόρτες του υπολογιστή.

openSerial\_Click(args..): Ανοίγει τη σειριακή πόρτα για ώστε να είναι δυνατή η επικοινωνία.

closeSerial\_Click(args..): Κλείνει τη σειριακή πόρτα.

Form1\_Load(args..): Εκτελεί ρουτίνες όταν φορτώνουμε τη φόρμα (εκκίνηση του προγράμματος).

serialPort1\_DataReceived(args..): Εκτελεί ρουτίνες όταν λάβει δεδομένα στη σειριακή θύρα. Εισάγει τα δεδομένα στη βάση δεδομένων MySql και στέλνει πληροφορίες στη σειριακή θύρα του υπολογιστή ώστε να εμφανιστούν στην οθόνη.

Form1\_FormClosed(args..): Εκτελεί ρουτίνες όταν κλείνουμε τη φόρμα (πρόγραμμα).

## B.3 Προγράμματα Διεπαφής Χρήστη

Για τη διεπαφή χρήστη αναπτύχθηκαν προγράμματα σε PHP και JavaScript. Η JavaScript χρειάζεται ώστε να ενημερώνει τμήματα της διεπαφής, τρέχει στον υπολογιστή του χρήστη και για να απαντά σε ενέργειες του χρήστη.

Ο φάκελος `wsn` έχει όλα τα αρχεία του δικτυακού τόπου που υποστηρίζει την εφαρμογή. Για την εγκατάσταση της διεπαφής αρκεί η τοποθέτηση του φακέλου `wsn` μέσα στον φάκελο `htdocs` του διακομιστή Apache. Ο φάκελος `calendar` περιέχει τα αρχεία για τη δημιουργία του ημερολογίου. Ο φάκελος `include` περιέχει τα αρχεία της κεφαλίδας, του υποσέλιδου και το αρχείο σύνδεσης με τη βάση δεδομένων. Ο φάκελος `images` περιέχει τα αρχεία των εικόνων που χρησιμοποιούνται στην εφαρμογή. Τα παρακάτω αρχεία εκτελούν τις λειτουργίες που περιγράφονται.

`Index.php`: αρχική σελίδα.

`./include/Header.php`: δημιουργία κεφαλίδας σελίδας και αριστερής στήλης.

`./include/Conn.php`: σύνδεση με τη βάση δεδομένων.

`./include/Footer.php`: δημιουργία υποσέλιδου σελίδας.

`./styles/style.css`: Αρχείο CSS (Cascade Style Sheets - αλληλουχία φύλλων στυλ).

`Login0.php`: αποτελείται από `Header` + περιεχόμενο(`login_form.php`)+`Footer`.

`Login_form.php`: δημιουργία σελίδας εισόδου. Εκτελεί το αρχείο `login_submit.php` και εάν επιτραπεί η είσοδος ανοίγει το αρχείο `node0.php` διαφορετικά το αρχείο `login0.php`.

`Login_submit()`: Αποστολή ερωτήματος εισόδου, έλεγχος δικαιωμάτων χρήστη δημιουργία μεταβλητών συνεδρίας.

`Node0.php`: αποτελείται από `Header` + περιεχόμενο(`node00.php`)+`Footer`.

`Node00.php`: εμφανίζει πληροφορίες για το δίκτυο.

`Node1.php`: αποτελείται από `Header` + περιεχόμενο(`node11.php`)+`Footer`.

Node11.php: Δημιουργεί την εικόνα του δικτύου (WSNet.php), το χάρτη της εικόνας (για αλληλεπίδραση με τους κόμβους) και αντιδρά σε ένα κλικ ή σε διπλό κλικ εκτελώντας Javascripts.

getnodedata-1.php: Εκτελείται όταν πατήσουμε ένα κλικ σε έναν κόμβο. Δημιουργεί, στο κάτω μέρος της οθόνης, έναν πίνακα με τις μετρήσεις του συγκεκριμένου κόμβου.

Drawnodedata.php: Δημιουργεί το γράφημα του μετρούμενου μεγέθους (θερμοκρασία).

Node2.php: αποτελείται από Header + περιεχόμενο(node21.php)+Footer.

Node21.php: Δημιουργεί την εικόνα του δικτύου, το χάρτη της εικόνας (για αλληλεπίδραση με τους κόμβους) και αντιδρά σε ένα κλικ εκτελώντας Javascript το οποίο εμφανίζει, στο κάτω μέρος της οθόνης, τις παραμέτρους του κόμβου(getnodedata-2.php). Με διπλό κλικ εμφανίζει το γράφημα του μετρούμενου μεγέθους (θερμοκρασία).

getnodedata-2.php: Εκτελείται όταν πατήσουμε ένα κλικ σε έναν κόμβο. Εμφανίζει, στο κάτω μέρος της οθόνης, τις παραμέτρους του συγκεκριμένου κόμβου.