



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ ΗΛΕΚΤΡΟΝΙΚΗΣ
ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

**ΑΝΑΠΤΥΞΗ ΣΥΣΤΗΜΑΤΟΣ ΤΑΥΤΟΠΟΙΗΣΗΣ
ΠΡΟΣΩΠΩΝ ΚΑΙ ΑΝΤΙΚΕΙΜΕΝΩΝ ΜΕ ΧΡΗΣΗ
ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΩΝ ΑΝΟΙΚΤΟΥ ΛΟΓΙΣΜΙΚΟΥ
ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ RFID**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΠΑΝΑΓΙΩΤΗΣ ΠΛΕΣΣΑΣ

Επιβλέποντες: Η. ΚΟΥΚΟΥΤΣΗΣ
ΚΑΘΗΓΗΤΗΣ Ε.Μ.Π.

Χ. ΜΑΡΜΑΛΙΔΗΣ
Δρ. – Μηχ. (Γ.Ε.Τ.Ε.Ν.)

Αθήνα, Μάρτιος 2015



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ ΗΛΕΚΤΡΟΝΙΚΗΣ
ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

**ΑΝΑΠΤΥΞΗ ΣΥΣΤΗΜΑΤΟΣ ΤΑΥΤΟΠΟΙΗΣΗΣ
ΠΡΟΣΩΠΩΝ ΚΑΙ ΑΝΤΙΚΕΙΜΕΝΩΝ ΜΕ ΧΡΗΣΗ
ΜΙΚΡΟΠΕΞΕΡΓΑΣΤΩΝ ΑΝΟΙΚΤΟΥ ΛΟΓΙΣΜΙΚΟΥ
ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ RFID**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΠΑΝΑΓΙΩΤΗΣ ΠΛΕΣΣΑΣ

Επιβλέποντες: Η. ΚΟΥΚΟΥΤΣΗΣ
ΚΑΘΗΓΗΤΗΣ Ε.Μ.Π.

Χ. ΜΑΡΜΑΛΙΔΗΣ
Δρ. – Μηχ. (Γ.Ε.Τ.Ε.Ν.)

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 13^η Μαρτίου 2015.

Η. ΚΟΥΚΟΥΤΣΗΣ
ΚΑΘΗΓΗΤΗΣ Ε.Μ.Π.

Κ. ΠΑΠΑΟΔΥΣΣΕΥΣ
ΚΑΘΗΓΗΤΗΣ Ε.Μ.Π.

Γ. ΚΑΜΠΟΥΡΑΚΗΣ
ΚΑΘΗΤΗΤΗΣ Ε.Μ.Π.

Αθήνα, Μάρτιος 2015

.....
ΠΑΝΑΓΙΩΤΗΣ ΠΛΕΣΣΑΣ

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright © ΠΑΝΑΓΙΩΤΗΣ ΠΛΕΣΣΑΣ, ΧΡΗΣΤΟΣ ΜΑΡΜΑΛΙΔΗΣ 2015

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

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

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν το συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

ΠΕΡΙΛΗΨΗ

Τα τελευταία χρόνια έχει εκδηλωθεί τεράστιο ενδιαφέρον για τα συστήματα ραδιοσυχνικής αναγνώρισης, γνωστά και ως RFID, τα οποία αναμένεται να αποτελέσουν την επόμενη τεχνολογία στα συστήματα αυτόματης αναγνώρισης αντικειμένων και ανάκτησης δεδομένων. Το ενδιαφέρον αυτό εκφράζεται τόσο από την ακαδημαϊκή κοινότητα σε ερευνητικό επίπεδο, όσο και από τους εν δυνάμει χρήστες της τεχνολογίας.

Η παρούσα διπλωματική έχει διπλό στόχο. Αφενός μεν να αποτελέσει μια γενική εισαγωγή στις βασικές δομές, στη θεωρία λειτουργίας και στις εφαρμογές των συστημάτων RFID και αφετέρου να παρουσιάσει αναλυτικά την υλοποίηση ενός συγκεκριμένου συστήματος ελεγχόμενης πρόσβασης μέσω του Arduino.

Πιο συγκεκριμένα, στο *Κεφάλαιο 1* γίνεται μια ιστορική αναδρομή τόσο στην δημιουργία του συστήματος Arduino, όσο και στην δημιουργία του RFID ως τεχνολογική εξέλιξη των συστημάτων ραντάρ αλλά και συμπληρωματική τεχνολογία των ευρέως διαδεδομένων γραμμωτών κωδικών. Στην συνέχεια στο *Κεφάλαιο 2* παρουσιάζονται αναλυτικά τα δομικά στοιχεία και εξετάζονται σε λεπτομέρεια τα τεχνικά χαρακτηριστικά του συστήματος RFID.

Στο *Κεφάλαιο 3* περιγράφεται η υλοποίηση και παρουσιάζεται ο τρόπος διασύνδεσης των επιμέρους συστημάτων που έχουν χρησιμοποιηθεί.

Στο *Κεφάλαιο 4* παρουσιάζεται συνολικά το υλικό που χρησιμοποιήθηκε στην υλοποίηση του συστήματος ταυτοποίησης, ενώ στην συνέχεια στο *Κεφάλαιο 5* αναπτύσσεται το λογισμικό με έμφαση στην ικανότητα κάλυψης απαιτήσεων μιας ιδιωτικής εταιρίας, αλλά και στην απλότητα χρήσης του.

Τέλος, στο *Κεφάλαιο 6* περιγράφεται και σχολιάζεται αναλυτικά ο κώδικας που έχει κατασκευαστεί για να υλοποιήσει το συγκεκριμένο σύστημα ταυτοποίησης προσώπων, ενώ ο κώδικας συγκεντρωμένος μπορεί να βρεθεί στο παράρτημα.

Λέξεις-Κλειδιά

Ραδιοσυχνική αναγνώριση, συστήματα αυτόματης αναγνώρισης, αναγνώστες RFID, πομποδέκτες RFID, κεραιές RFID, περιοχή ανάγνωσης συστήματος RFID.

ABSTRACT

The Radio frequency identification systems (RFID) have attracted an enormous interest during the recent years and they are expected to become soon the prevailing technology in the automatic identification and data capture systems. This interest comes from both the academic research community and the rest potential technology stakeholders.

This dissertation has two aims. Firstly, to give a general introduction into the basic structure, the theory of operation and the applications of the RFID systems and secondly, to present in detail the design of a particular access control system.

Chapter 1 makes a throwback in both the creation of Arduino Project but also to the fundamental principles of RFID systems, along with their history as a technological branch of radar systems and as a complementary technology to the currently prevailing barcode technology. In addition, chapter 2 presents analytical components and detail examination for the technical characteristics of RFID system.

Chapter 3 describes the implementation and provides a detailed presentation of how the components used have been interfaced to produce the characteristics of the corresponding application.

Chapter 4 presents the hardware used in the implementation of the access control system, while in Chapter 5 develops the software with emphasis on the ability to satisfy requirements of a private company with the necessary simplicity of use.

Finally, Chapter 6 describes and discusses in detail the code that has been engineered to deliver the access control system. The whole code can be found in the Annex.

KEYWORDS

Radio frequency identification, RFID, RFID readers, RFID tags, transponders, RFID reading range, inductive coupling, capacitive coupling, backscatter coupling, reflecting coupling, anti-collision systems.

Ευχαριστίες

Καταρχάς θα ήθελα να ευχαριστήσω τον καθηγητή μου Ηλία Κουκούτση γιατί με στήριξε και με βοήθησε σε κάθε δυσκολία υποδεικνύοντας λύσεις ώστε να περατώσω την διπλωματική. Επίσης θα ήθελα να ευχαριστήσω τον Δρ.-Μηχ. Χρήστο Μαρμαλίδη, Ειδικό Επιστήμονα του Γραφείου Έρευνας και Τεχνολογικών Εξελίξεων Ναυτικού (Γ.Ε.Τ.Ε.Ν) για τις παρατηρήσεις και τις προτάσεις του που συντέλεσαν στην βελτίωση της διπλωματικής.

Αθήνα, Μάρτιος 2015
Π. Πλέσσας

*Αφιερώνεται με ιδιαίτερη ευγνωμοσύνη στην
οικογένεια μου, για την αμέριστη και συνεχή
υποστήριξη της.*

Περιεχόμενα

ΠΕΡΙΛΗΨΗ.....	5
1. Βασικές έννοιες και ιστορική εξέλιξη.....	13
1.1 Ιστορικό μικροελεγκτή Arduino	13
1.2 Ιστορικό τεχνολογίας RFID.....	14
2. Βασικές έννοιες, δομή & χαρακτηριστικά ενός συστήματος RFID	16
2.1 Βασική δομή ενός συστήματος RFID.....	16
2.2 Αρχή λειτουργίας ενός συστήματος RFID.....	17
2.3 Πομποδέκτες (RFID transponders ή tags).....	18
2.3.1 Κατηγοριοποίηση ανάλογα με τον τρόπο λήψης ισχύος	18
2.3.2 Κατηγοριοποίηση ανάλογα με την δυνατότητα επανεγγραφής τους.....	21
2.3.3 Κατηγοριοποίηση ανάλογα με την κατασκευή τους	22
2.3.4 Κατηγοριοποίηση ανάλογα με την εφαρμογή τους	24
2.3.5 Κύκλωμα πομποδέκτη (RFID Tag Circuit).....	26
2.4 Αναγνώστες (RFID Readers)	29
2.4.1 Οι υπομονάδες ενός RFID αναγνώστη.....	30
2.4.2 Μονάδα υψηλών συχνοτήτων	31
2.4.3 Μονάδα ελέγχου	33
2.4.4 Είδη αναγνωστών	33
2.5 Σύστημα διαχείρισης & επεξεργασίας της πληροφορίας	34
2.6 Ιδιαίτερα χαρακτηριστικά των συχνοτήτων	35
2.7 Περιοχή ανάγνωσης / εγγραφής.....	41
2.7.1 Ορισμός.....	41
2.7.2 Παράγοντες καθορισμού της περιοχής ανάγνωσης.....	41
2.7.3 Επιθυμητό μέγεθος & σχήμα περιοχής ανάγνωσης ή εγγραφής	42
2.7.4 Κατηγοριοποίηση συστημάτων RFID ανάλογα με απόσταση ανάγνωσης.....	43
2.7.5 Ταχύτητα ανάγνωσης δεδομένων	44
2.8 Συστήματα αποφυγής σύγκρουσης πομποδεκτών (Tag Anticollision).....	45

2.9 Συστήματα αποφυγής σύγκρουσης αναγνωστών (Reader Anticollision)	46
2.10 Ασφάλεια στα συστήματα RFID	47
2.11 Πρότυπα και πρωτόκολλα	47
2.11.1 Πρότυπο ISO	47
2.11.2 Το σύστημα EPC.....	48
2.12 Κατηγορίες σύζευξης.....	50
2.12.1 Επαγωγική σύζευξη	50
2.12.2 Ηλεκτρομαγνητική σύζευξη σκέδασης	58
2.12.3 Χωρητική σύζευξη.....	63
2.13 Κωδικοποίηση δεδομένων.....	64
2.14 Διαμόρφωση δεδομένων.....	64
2.15 Πλεονεκτήματα συστημάτων RFID.....	65
3. Υλοποίηση	66
4. Hardware (Υλικό)	68
4.1 Arduino Mega ADK rev3	68
4.1.1 Schematics & Reference Design του Arduino Mega ADK rev3	68
4.1.2 Ακροδέκτες POWER του Arduino Mega ADK rev3.....	70
4.1.3 Ακροδέκτες ANALOG IN του Arduino Mega ADK rev3	71
4.1.4 Ακροδέκτες DIGITAL IN/OUT του Arduino Mega ADK rev3.....	72
4.1.5 Ακροδέκτες Communication του Arduino Mega ADK rev3	74
4.1.6 Τεχνικά χαρακτηριστικά Arduino Mega ADK rev3	74
4.2 Arduino Ethernet Shield rev3.....	75
4.3 Cubieboard A10	77
4.4 ID Innovations ID-20LA RFID Reader	78
4.5 Linksys Ethernet Switch	82
4.6 DC Relay.....	83
4.7 Ηλεκτρική Κλειδαριά (Electric Strike).....	86
5. Software (Λογισμικό).....	88
5.1 Notepad ++.....	88

5.2 Arduino IDE.....	88
5.3 Διακομιστής LAMP.....	91
6. Προγραμματισμός του μικροελεγκτή της Atmel	95
7. Συμπεράσματα - Προτάσεις	122
Βιβλιογραφία	125
Παράρτημα	126
Παραμετροποίηση Arduino Mega.....	126

1. Βασικές έννοιες και ιστορική εξέλιξη

1.1 Ιστορικό μικροελεγκτή Arduino

Το 2005, ένα σχέδιο κίνησε προκειμένου να φτιαχτεί μία συσκευή για τον έλεγχο προγραμμάτων διαδραστικών σχεδίων από μαθητές, η οποία θα ήταν πιο φθηνή από άλλα πρωτότυπα συστήματα διαθέσιμα εκείνη την περίοδο. Οι ιδρυτές Massimo Banzι και David Cueartielles ονόμασαν το σχέδιο από τον Arduino της Ivrea και ξεκίνησαν να παράγουν πλακέτες σε ένα μικρό εργοστάσιο στην Ιβρέα, κωμόπολη της επαρχίας Τορίνο στην περιοχή Πεδεμόντιο της βορειοδυτικής Ιταλίας την ίδια περιοχή στην οποία στεγαζόταν η εταιρία υπολογιστών Olivetti. Το σχέδιο Arduino είναι μία διακλάδωση της πλατφόρμας Wiring για λογισμικό ανοικτού κώδικα και προγραμματίζεται χρησιμοποιώντας μια γλώσσα βασισμένη στο Wiring (σύνταξη και βιβλιοθήκες), παρόμοια με την C++ με απλοποιήσεις και αλλαγές, καθώς και ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE).

Μία πλακέτα Arduino αποτελείται από ένα μικροελεγκτή Atmel AVR (ATmega168/328 στις νεότερες εκδόσεις ή ATmega8 στις παλαιότερες) και συμπληρωματικά εξαρτήματα για την διευκόλυνση του χρήστη στον προγραμματισμό και την ενσωμάτωση του σε άλλα κυκλώματα. Όλες οι πλακέτες περιλαμβάνουν ένα γραμμικό ρυθμιστή τάσης 5V και έναν κρυσταλλικό ταλαντωτή 16MHz (ή κεραμικό αντηχητή σε κάποιες παραλλαγές). Ο μικροελεγκτής είναι από κατασκευής προγραμματισμένος με ένα *bootloader*, έτσι ώστε να μην χρειάζεται εξωτερικός προγραμματιστής.



Σε εννοιολογικό επίπεδο, στην χρήση του Arduino software stack, όλα τα boards προγραμματίζονται με μία RS-232 σειριακή σύνδεση, αλλά ο τρόπος που επιτυγχάνεται αυτό διαφέρει σε κάθε hardware εκδοχή. Οι σειριακές πλάκες Arduino περιέχουν ένα απλό level shifter κύκλωμα για την μετατροπή σημάτων επιπέδου RS-232 και TTL. Τα τωρινά Arduino προγραμματίζονται μέσω USB, αυτό καθιστάτε δυνατό μέσω της εφαρμογής προσαρμοστικών chip USB-to-Serial όπως το FTDI FT232. Κάποιες παραλλαγές, όπως το Arduino mini και το ανεπίσημο Boarduino, χρησιμοποιούν ένα αφαιρούμενο USB-to-Serial καλώδιο ή board, Bluetooth ή άλλες μεθόδους. Τα Diecimila, Duemilanove και το Uno παρέχουν 14 ψηφιακά I/O pins, έξι από τα οποία μπορούν να παράγουν pulse-width διαμορφωμένα σήματα και έξι αναλογικά δεδομένα, ενώ το Mega παρέχει 50 ψηφιακά I/O pins και 16 αναλογικά. Αυτά τα pins βρίσκονται συνήθως στην κορυφή και στις άκρες των boards με female headers 0.1 ιντσών (2,2mm). Επιπλέον διάφορες εφαρμογές με την μορφή

shield plug-in είναι εμπορικώς διαθέσιμα. Στο σύστημα που αναπτύχθηκε χρησιμοποιήθηκε η έκδοση του Arduino Mega ADK rev3 λόγω χαμηλού κόστους αγοράς, πολλών δυνατοτήτων και χαμηλής κατανάλωσης ρεύματος.

1.2 Ιστορικό τεχνολογίας RFID

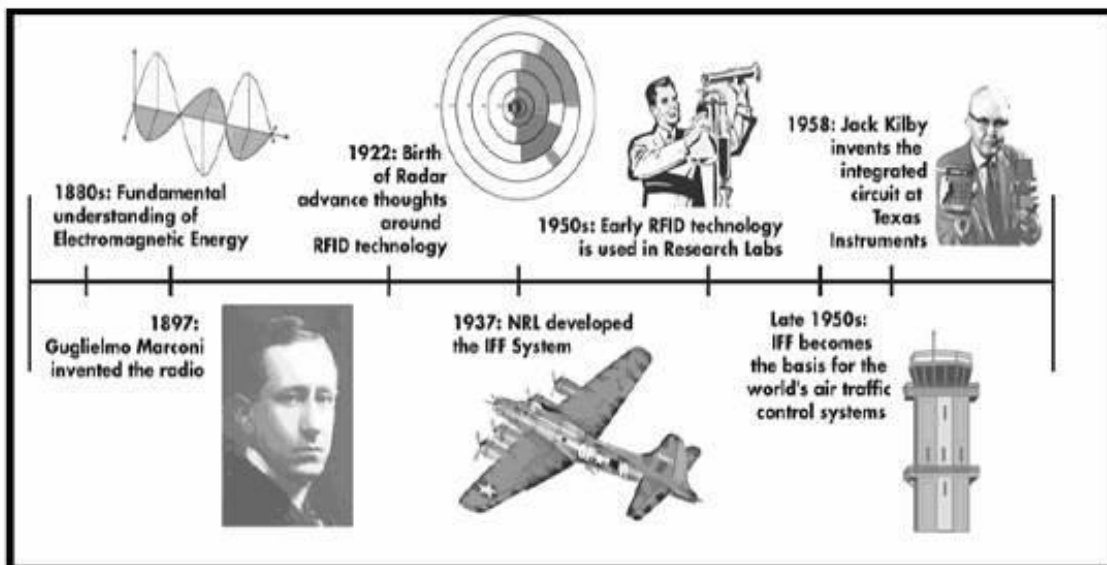
Ο προάγγελος αυτού που σήμερα ονομάζεται τεχνολογία RFID αναπτύχθηκε κατά τη διάρκεια του Β' Παγκοσμίου από το στρατό προκειμένου να αποτρέψει την κατάρριψη αεροπλάνων από τα λεγόμενα «φίλικα πυρά». Σύμφωνα με την καταγεγραμμένη πληροφορία που υπάρχει για τα συστήματα εκείνης της περιόδου (RFID Journal), το τότε στρατιωτικό σύστημα εξέπεμπε ένα ραδιοσυχνικό σήμα σε κάθε αεροπλάνο που πλησίαζε και εάν αυτό ήταν φιλικό, τότε ανάγκαζε τον αναμεταδότη του αεροπλάνου να ανταποκριθεί στο σήμα που δεχόταν. Κατά τις δεκαετίες 1960 και 1970, ξεκίνησε η χρήση πομπών, που εξέπεμπαν ραδιοκύματα και είχαν σκοπό την παρακολούθηση του πολεμικού εξοπλισμού. Στα 1980 και 1990 άρχισε να γίνεται εφικτή η κατασκευή ενός λεπτομερούς σχεδίου του κυκλώματος ενός πομπού, με εξαίρεση την κεραία, σε ένα πολύ μικρό κύκλωμα (chip). Αυτό το τεχνολογικό άλμα σηματοδότησε την άφιξη της ετικέτας ραδιοσυχνικής αναγνώρισης (RFID tag), μιας μικρής, υψηλής συχνότητας, συσκευής αναγνώρισης, η οποία μπορεί να τοποθετηθεί πάνω σε οποιοδήποτε αντικείμενο. Έως τα μέσα της δεκαετίας του 1990, οι RFID ετικέτες έγιναν ευρέως γνωστές χάρη στη χρήση τους σε μια πληθώρα εφαρμογών που εξυπηρετούσαν κυρίως καταναλωτές. Παραδείγματα τέτοιων εφαρμογών είναι η χρήση RFID ετικετών στα αυτοκίνητα, για την αυτόματη πληρωμή στα διόδια, η χρήση καρτών που παίζουν το ρόλο των κλειδιών για την είσοδο σε χώρους ελεγχόμενης πρόσβασης, καθώς και στα συστήματα ανεφοδιασμού καυσίμων με αυτόματη πληρωμή. Το κόστος μιας RFID ετικέτας ανέρχεται, αυτή τη στιγμή, περίπου στο \$1 ενώ η τεχνολογία RFID εδραιώνεται όλο και περισσότερο. Παρ' όλα αυτά, όμως, δεν έχουν καθοριστεί τα πρότυπα εκείνα που θα επιτρέψουν καθολική υιοθέτηση αυτής της τεχνολογίας.

Το 1999 ένα νέο κέντρο, αποκαλούμενο ως Auto-ID Center, ιδρύθηκε στο πανεπιστήμιο MIT, με σκοπό τη δημιουργία των τεχνολογιών, των προτύπων και των συστημάτων που θα οδηγούσαν στην ευρεία υιοθέτηση των RFID ετικετών χαμηλού κόστους. Το κέντρο αυτό πρότεινε μία προσέγγιση προσανατολισμένη στην έννοια του συστήματος. Οι αρχές που πρότεινε υπαγόρευαν τα εξής: διατήρηση της απλής μορφής των ετικετών και εστίαση στο κόστος και την απόδοση, άντληση των δεδομένων από τις ετικέτες με τη βοήθεια συσκευής ανάγνωσης και την εισαγωγή τους στο σύστημα της επιχείρησης, καθώς επίσης και την ευελιξία του λογισμικού του συστήματος, να αποθηκεύει τα επιχειρησιακά δεδομένα και να εκτελεί τις απαραίτητες ενέργειες προκειμένου να εκμεταλλεύεται τα οφέλη των RFID ετικετών. Αυτή η μινιμαλιστική προσέγγιση, που συχνά αναφέρεται ως προσέγγιση του Ηλεκτρονικού Κωδικού Προϊόντος (Electronic Product Code-EPC), ερχόταν σε αντίθεση με τον έως τότε κόσμο των ιδιαίτερα εξειδικευμένων ετικετών, οι οποίες περιείχαν πολύ μεγάλη ενσωματωμένη λειτουργικότητα.



Είναι προφανές σήμερα ότι η απλότητα αυτή που προτείνει η προσέγγιση EPC προσφέρει δύο σπουδαία πλεονεκτήματα. Πρώτον, θέτοντας τη λειτουργικότητα σε ένα συγκεκριμένο επίπεδο, γίνεται εφικτή η εκμετάλλευση ενός κοινού προτύπου από ένα μεγάλο αριθμό εφαρμογών. Δεύτερον, περιορίζοντας το μέγεθος τους σχεδίου κυκλώματος στο εσωτερικό των ετικετών είναι δυνατό να μειωθούν τα κόστη, να βελτιωθεί η ποιότητα λειτουργίας και η απόδοση του συστήματος.

Τα πρότυπα που προτάθηκαν από το Auto-ID Center κέρδισαν την ευρεία υποστήριξη των τελικών χρηστών της τεχνολογίας RFID και οδήγησαν στην ίδρυση ενός νέου μη κερδοσκοπικού φορέα υπό την ονομασία EPC global, ο οποίος συνεχίζει το έργο του Auto-ID Center. Ο φορέας EPC global έχει δημοσιεύσει έναν αριθμό προτύπων υλικού και λογισμικού, τα οποία αναμένεται να οδηγήσουν σε μία νέα εποχή τυποποιημένης, χαμηλού κόστους και ευρέως διαδεδομένης RFID τεχνολογίας (Mark Global, 2004).



Αναμεταδότες ή ετικέτες διαφοροποιούνται βάση της περιοχής συχνοτήτων σε τέσσερις κατηγορίες:

- Χαμηλών Συχνοτήτων (Low Frequency) < 135 KHz
- Υψηλών Συχνοτήτων (High Frequency) 13,56 MHz
- Πολύ Υψηλών Συχνοτήτων (Ultra High Frequency) 860-930 MHz
- Συχνότητες Μικροκυμάτων (Microwaves) 2,45 GHz

2. Βασικές έννοιες, δομή & χαρακτηριστικά ενός συστήματος RFID

2.1 Βασική δομή ενός συστήματος RFID

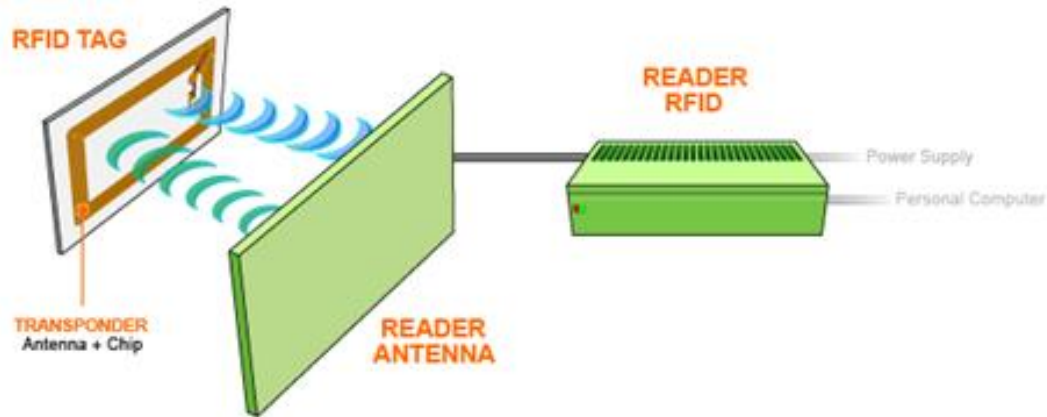
Η τεχνολογία RFID, είναι τα αρχικά του όρου Radio Frequency Identification, η απόδοση του στα ελληνικά ορίζεται ως «ταυτοποίηση μέσω ραδιοσυχνοτήτων» ή «ραδιοσυχνική αναγνώριση».

Τα συστήματα RFID αποτελούν ένα υποσύνολο των Συστημάτων Αυτόματου Προσδιορισμού (Automatic Identification Systems). Ειδικότερα, λειτουργεί ως γενικός όρος των τεχνολογιών που χρησιμοποιούν ραδιοκύματα, για να προσδιορίσουν αυτόματα ανθρώπους ή αντικείμενα και αποτελεί την τεχνολογική εξέλιξη των ραβδωτών κωδίκων (barcode). Τις τελευταίες δεκαετίες, άρχισε να εδραιώνεται η χρήση και εκμετάλλευσή της τεχνολογίας RFID. Αρχικά, σε πειραματικό στάδιο και σε εργαστηριακό επίπεδο, για να φτάσουμε στο σήμερα, όπου γίνεται λόγος για εφαρμογή της τεχνολογίας RFID στην καθημερινή ζωή των ανθρώπων, κυρίως μέσω του εμπορίου. Παράλληλα αναπτύσσεται το ενδεχόμενο της ευρείας εφαρμογής του, με την καθιέρωση προτύπων και την λειτουργία της σε παγκόσμιο επίπεδο.

Τα συστήματα RFID απαρτίζονται από τέσσερα κύρια μέρη τα οποία και θα αναλυθούν στις επόμενες σελίδες.

- ✚ Το πρώτο είναι οι **πομποδέκτες** (transponders ή tags), που συχνά αναφέρονται και ως ετικέτες RFID (RFID tags). Οι ετικέτες RFID είναι μικρά chips που αποτελούνται από ένα ολοκληρωμένο κύκλωμα, το οποίο περιλαμβάνει μνήμη ώστε να αποθηκεύει δεδομένα- πληροφορίες και μία κεραία. Το μέγεθός τους, μπορεί να είναι τόσο μικρό, όσο το μισό ενός κόκκου άμμου (1/3 του χιλιοστού), ανάλογα με το τύπο τις ετικέτας.
- ✚ Το δεύτερο μέρος είναι οι **αναγνώστες** ή αισθητήρες (readers ή interrogators), οι οποίοι ανακτούν τα δεδομένα από τις ετικέτες RFID. Οι αναγνώστες RFID, έχουν ενσωματωμένα μια κεραία και μια μονάδα ελέγχου όπου δίνει την δυνατότητα να εντοπίζει και να ενεργοποιεί τον πομποδέκτη και να διαβάζει τις αποθηκευμένες πληροφορίες. Σε ορισμένες περιπτώσεις, ο αναγνώστης μπορεί και να λειτουργήσει ως συσκευή εγγραφής δεδομένων στον πομποδέκτη. Ο αναγνώστης χαρακτηρίζεται από την περιοχή ή ζώνη ή εύρος ανάγνωσης κι εγγραφής του, μέσα στην οποία μπορεί να επικοινωνήσει με τον πομποδέκτη.
- ✚ Το **σύστημα διαχείρισης κι επεξεργασίας της πληροφορίας** (λογισμικό), την οποία διαβάζει ο αναγνώστης από τον πομποδέκτη
- ✚ Ο **χώρος** μέσα στον οποίο λειτουργεί το σύστημα. Στο χώρο αυτό συνήθως υπάρχουν κι άλλα «αντικείμενα», τα οποία μπορούν να επηρεάσουν τη λειτουργία του συστήματος.

Η λειτουργία των συστημάτων RFID είναι απλή και βασίζεται στη δυναμική και αμφίδρομη επικοινωνία των ετικετών και των αναγνώστων. Όταν οι ετικέτες RFID βρεθούν στην εμβέλεια της κεραίας του αναγνώστη, η μονάδα ελέγχου επικοινωνεί με ραδιοκύματα με την κεραία των ετικετών RFID. Οι ετικέτες RFID ενεργοποιούνται με τη σειρά τους και επιστρέφουν τα αναζητούμενα δεδομένα στους αναγνώστες. Στη συνέχεια παρεμβαίνει ένα ενδιάμεσο λογισμικό, το οποίο κατανοεί τις πληροφορίες, οι οποίες αποστέλλονται από τη μονάδα ελέγχου του αναγνώστη. Ο αναγνώστης τις μεταφέρει στο εκάστοτε πληροφοριακό σύστημα (Εικόνα 1).

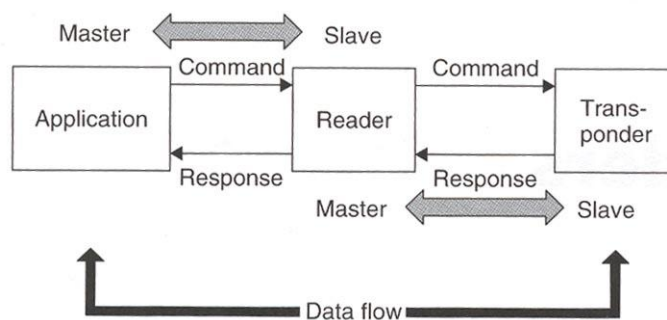


Εικόνα 1: Σύστημα RFID

2.2 Αρχή λειτουργίας ενός συστήματος RFID

Οι λειτουργίες ανάγνωσης κι εγγραφής σε ένα σύστημα RFID πραγματοποιούνται σύμφωνα με την αρχή αφέντη-δούλου (master-slave). Αυτό σημαίνει ότι όλες οι λειτουργίες του αναγνώστη και του πομποδέκτη ελέγχονται από το σύστημα διαχείρισης κι επεξεργασίας της πληροφορίας. Το λογισμικό είναι λοιπόν ο αφέντης, ενώ ο αναγνώστης ως δούλος ενεργοποιείται όταν δεχθεί εντολές ανάγνωσης ή εγγραφής.

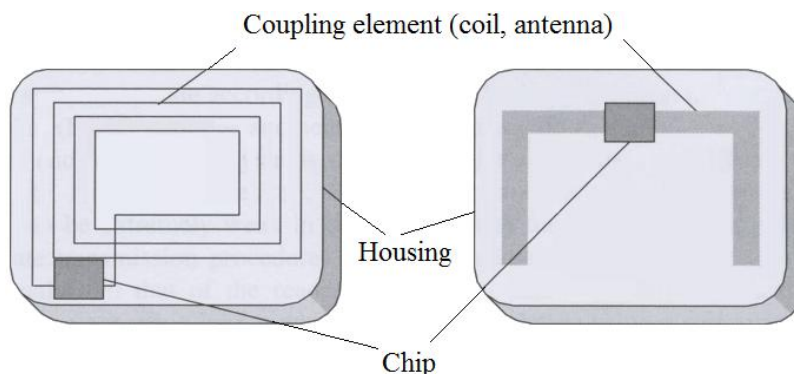
Προκειμένου να εκτελέσει μια τέτοια εντολή, ο αναγνώστης έρχεται σε επικοινωνία με τον πομποδέκτη. Τώρα ο αναγνώστης παίζει το ρόλο του αφέντη σε σχέση με τον πομποδέκτη. Ο πομποδέκτης απαντά μόνο στις εντολές του αναγνώστη και ποτέ δε λειτουργεί ανεξάρτητα. Εξαιρέση αποτελούν οι πομποδέκτες μόνο ανάγνωσης, οι οποίοι μόλις βρεθούν στη ζώνη ανάγνωσης του αναγνώστη, αρχίζουν να εκπέμπουν συνεχώς το σειριακό τους αριθμό. Στην εικόνα 2 φαίνεται η αρχιτεκτονική αφέντη-δούλου ανάμεσα στο σύστημα διαχείρισης της πληροφορίας (λογισμικό), τον αναγνώστη και τον πομποδέκτη.



Εικόνα 2: Η αρχιτεκτονική αφέντη-δούλου

2.3 Πομποδέκτες (RFID transponders ή tags)

Ένας πομποδέκτης αποτελείται από ένα ηλεκτρονικό κύκλωμα (IC, chip) στο οποίο αποθηκεύει τα δεδομένα που μεταφέρει, κι από ένα στοιχείο σύζευξης με τη μορφή συνήθως πηνίου-κεραίας (εικόνα 3).



Εικόνα 3: Διάγραμμα Πομποδέκτη RFID

Οι πομποδέκτες ταξινομούνται σε διάφορες κατηγορίες, ανάλογα με κάποια κριτήρια, όπως τον τρόπο παροχής ισχύος, την ικανότητα επανεγγραφής τους, τη μνήμη τους τα οποία περιγράφονται αναλυτικότερα παρακάτω.

2.3.1 Κατηγοριοποίηση ανάλογα με τον τρόπο λήψης ισχύος

Η λειτουργία των πομποδεκτών, δηλαδή η λειτουργία του ολοκληρωμένου κυκλώματός τους και η μετάδοση δεδομένων μέσω της κεραίας τους προϋποθέτει κάποια μορφή παροχής ισχύος σε αυτά. Σε γενικές γραμμές, μπορούμε να πούμε ότι οι πομποδέκτες ενεργοποιούνται, λαμβάνοντας ενέργεια από το μαγνητικό, ηλεκτρικό ή ηλεκτρομαγνητικό πεδίο που αναπτύσσεται γύρω από την κεραία του αναγνώστη, οπότε έχουμε επαγωγική, χωρητική ή ηλεκτρομαγνητική (μικροκυματική) σύζευξη αναγνώστη και πομποδέκτη αντίστοιχα. Το είδος του πεδίου που αναπτύσσεται εξαρτάται από τα κυκλωματικά στοιχεία της κεραίας του αναγνώστη και από τη συχνότητα λειτουργίας του συστήματος.

Επιπλέον, οι πομποδέκτες μπορούν να περιέχουν ή όχι κάποια πηγή ενέργειας (μπαταρία), οπότε διακρίνονται σε **ενεργητικούς, παθητικούς, ημιενεργητικούς / ημιπαθητικούς** αντίστοιχα.

- Παθητικοί Πομποδέκτες (Passive RFID Tags)

Οι παθητικοί πομποδέκτες δεν έχουν εσωτερική πηγή ενέργειας(μπαταρία), αλλά αντίθετα λαμβάνουν όλη την απαιτούμενη ενέργεια για τη λειτουργία τους και για τη μετάδοση των δεδομένων τους από τα ραδιοσήματα που τους στέλνει ο αναγνώστης μέσω της μονάδας σύζευξής τους. Με άλλα λόγια, ένας παθητικός πομποδέκτης που δε διαθέτει δική του παροχή ισχύος, είναι τελείως αδρανής, όταν βρίσκεται εκτός της ζώνης ανάγνωσης του αναγνώστη του συστήματος κι ενεργοποιείται μονάχα όταν βρεθεί εντός αυτής. Για όσο χρονικό διάστημα ο πομποδέκτης παραμένει μέσα στη ζώνη ανάγνωσης συνεχίζει να λειτουργεί.

Το βασικό *πλεονέκτημα* των παθητικών πομποδεκτών είναι το μικρό τους μέγεθος και κόστος κι αναμένεται να επικρατήσουν στις ανοιχτές εφαρμογές. Σήμερα κατασκευάζονται τσιπ με διαστάσεις μικρότερες του ενός mm, οπότε το τελικό μέγεθος των παθητικών πομποδεκτών εξαρτάται κυρίως από τις διαστάσεις της κεραίας. Επίσης, οι παθητικοί πομποδέκτες θεωρούνται γενικά πιο αξιόπιστοι, διότι αποτελούνται από λιγότερα τμήματα κι έχουν απλούστερο σχεδιασμό. Ακόμη, έχουν πρακτικά απεριόριστο χρόνο ζωής.

Ένα *μειονέκτημά* τους είναι ότι προκειμένου να διατηρηθούν χαμηλά τα κόστη, δε διαθέτουν μεγάλη μνήμη (περίπου 96 bits).

Τέλος, μπορούμε να πούμε ότι γενικά τα συστήματα με παθητικούς πομποδέκτες χαρακτηρίζονται από μικρότερη απόσταση ανάγνωσης από τον αναγνώστη, λόγω της μικρότερης διαθέσιμης σε αυτούς ισχύος. Ένα άλλο χαρακτηριστικό τους είναι η μικρότερη ταχύτητα μετάδοσης δεδομένων, διότι οι παθητικοί πομποδέκτες μεταδίδουν δεδομένα, περιμένοντας να λάβουν πρώτα την απαραίτητη ισχύ από τον αναγνώστη.

- Ενεργητικοί Πομποδέκτες (Active RFID Tags)

Οι ενεργητικοί πομποδέκτες χρησιμοποιούν τόσο την ενέργεια που τους παρέχει ο αναγνώστης, όσο και την ενέργεια από τη μπαταρία τους, προκειμένου να μεταδώσουν τα δεδομένα τους στον αναγνώστη. Η λειτουργία του εσωτερικού τους κυκλώματος τροφοδοτείται αποκλειστικά από την ενέργεια που παρέχει η ενσωματωμένη μπαταρία τους. Ουσιαστικά, η ενέργεια που λαμβάνει ένας ενεργητικός πομποδέκτης από τον αναγνώστη, χρησιμοποιείται κυρίως για την ενεργοποίηση του εσωτερικού του κυκλώματος, ώστε να αρχίσει να μεταδίδει δεδομένα. Λόγω της μεγαλύτερης ποσότητας διαθέσιμης ισχύος, τα RFID συστήματα με ενεργητικούς πομποδέκτες, χαρακτηρίζονται από μεγαλύτερο εύρος ανάγνωσης, που μπορεί να φτάσει μέχρι και τα 30 μέτρα κι από μεγαλύτερες ταχύτητες ανάγνωσης κι εγγραφής. Οι ενεργητικοί πομποδέκτες, εξαιτίας της πολυπλοκότητας τους είναι πολύ *μεγαλύτεροι* σε μέγεθος από τους παθητικούς πομποδέκτες, ενώ λόγω

του μεγάλου κόστους κατασκευής τους, είναι πολύ ακριβότεροι. Επίσης, έχουν ανάγκη από συντήρηση. Όταν η μπαταρία τους αποφορτιστεί, τότε θα πρέπει αυτή να φορτιστεί ή να αντικατασταθεί ή ακόμη σε μερικές περιπτώσεις να αντικατασταθεί και ο ίδιος ο πομποδέκτης. Προκειμένου να αποφευχθεί η συχνή ανάγκη για επαναφόρτιση της μπαταρίας, το ολοκληρωμένο κύκλωμα του πομποδέκτη περνά αυτόματα σε ημιανανεργή κατάσταση (stand-by), όταν ο πομποδέκτης βρεθεί εκτός του εύρους ανάγνωσης του αναγνώστη. Με τον τρόπο αυτό η ισχύς που καταναλώνεται αντιστοιχεί σε λίγα μόνο μA . Το ολοκληρωμένο κύκλωμα ενεργοποιείται μόνο όταν λάβει ένα αρκετά ισχυρό σήμα από τον αναγνώστη, οπότε επανέρχεται στην κανονική λειτουργία. Οι ενεργητικοί πομποδέκτες έχουν αυξημένη λειτουργικότητα σε σχέση με τους παθητικούς, με την έννοια ότι μπορούν να εκτελέσουν κάποιες «έξυπνες» λειτουργίες χωρίς την παρουσία του αναγνώστη, διαθέτοντας εσωτερικούς «έξυπνους» μηχανισμούς (*internal intelligence*).

- *Ημιενεργητικοί/Ημιπαθητικοί Πομποδέκτες*

Σε αυτήν την κατηγορία ανήκουν οι πομποδέκτες που διαθέτουν μπαταρία, αλλά δε χρησιμοποιούν την ισχύ της μπαταρίας τους για τη μετάδοση ραδιοσημάτων στον αναγνώστη παρά μόνο για τη λειτουργία του ολοκληρωμένου κυκλώματός τους και της διατήρησης των δεδομένων τους. Είναι γνωστά ως ημιενεργητικά ή ημιπαθητικά tags. Οι δύο όροι είναι συνώνυμοι και ο όρος που κάθε φορά χρησιμοποιείται εξαρτάται από την επιλογή του κατασκευαστή. Τα ημιενεργητικά tags είναι πολύ πιο συνηθισμένα από τα αμιγώς ενεργητικά tags, τόσο ώστε πολλές φορές στη βιβλιογραφία τα ημιενεργητικά tags αναφέρονται ως ενεργητικά.

Σε αντίθεση με τα αμιγώς παθητικά tags, τα οποία δεν μπορούν να εκτελέσουν καμιά λειτουργία παρά μόνο υπό την παρουσία του αναγνώστη, τα ημιπαθητικά tags έχουν τη δυνατότητα να εκτελέσουν κάποιες τοπικές λειτουργίες, όταν δε βρίσκονται μέσα στην περιοχή ανάγνωσης. Για παράδειγμα, ένα ημιπαθητικό tag μπορεί να κατασκευαστεί με ένα αισθητήρα θερμοκρασίας στο εσωτερικό του, έτσι ώστε να είναι προγραμματισμένο να απομνημονεύει την υψηλότερη και τη χαμηλότερη παρατηρούμενη θερμοκρασία. Όταν το tag διαβάζεται από τον αναγνώστη, τότε αυτό αναφέρει το σειριακό του αριθμό, ό,τι άλλα δεδομένα που έχει τυχόν αποθηκευμένα, τις δύο αυτές ακραίες θερμοκρασίες και την τρέχουσα θερμοκρασία. Ένα τέτοιο tag πάνω σε μία παλέτα κατεψυγμένων ειδών έχει τη δυνατότητα να πιστοποιεί συνεχώς το αν τα προϊόντα μέσα στην παλέτα βρίσκονται συνεχώς μέσα στο εύρος των σωστών θερμοκρασιών κατά τη μεταφορά τους. Η μπαταρία ενός ημιπαθητικού tag είναι πολύ μικρότερη (και φθηνότερη) απ'ότι αυτή ενός ενεργητικού tag, οπότε και τα ίδια τα tags είναι μικρότερα και φθηνότερα.

Η απόσταση ανάγνωσης των ημιπαθητικών πομποδεκτών είναι γενικά αρκετά μεγαλύτερη από αυτή ενός παθητικού πομποδέκτη και μικρότερη ενός ενεργητικού. Σε αντίθεση με τα ενεργητικά tags, τα ημιπαθητικά δεν έχουν ανάγκη συντήρησης, με την έννοια ότι όταν η μπαταρία τους αποφορτιστεί, τότε το tag αχρηστεύεται.

2.3.2 Κατηγοριοποίηση ανάλογα με την δυνατότητα επανεγγραφής τους

Ένα άλλο σημαντικό χαρακτηριστικό των πομποδεκτών είναι το αν επιτρέπουν ή όχι και πόσες φορές την επανεγγραφή δεδομένων στη μνήμη τους.

Διακρίνουμε τρεις βασικές κατηγορίες πομποδεκτών τους πομποδέκτες μόνο ανάγνωσης (Read-only), τους πομποδέκτες μίας εγγραφής-πολλών αναγνώσεων (Write Once-Read Many) γνωστοί και ως WORM και τους πομποδέκτες ανάγνωσης/εγγραφής (Read/Write).

- **Πομποδέκτες μόνο ανάγνωσης (Read-only)**

Όπως υπονοεί το όνομά τους οι πομποδέκτες μόνο ανάγνωσης δεν υποστηρίζουν λειτουργίες εγγραφής. Τα δεδομένα που περιέχουν, συνήθως ένας μοναδικός σειριακός αριθμός αναγνώρισης με ένα ψηφίο ελέγχου, εγγράφονται στη μνήμη τους με χρήση ακτίνων laser κατά το κατασκευαστικό στάδιο του ολοκληρωμένου κυκλώματος (τσιπ). Κατά συνέπεια, δε μπορούν να τροποποιηθούν. Το RFID σύστημα διαθέτει συνήθως βάσεις δεδομένων, όπου συσχετίζεται ο σειριακός αριθμός του πομποδέκτη με το υπό αναγνώριση αντικείμενο. Όταν ένας πομποδέκτης μόνο ανάγνωσης βρεθεί στην περιοχή ανάγνωσης του αναγνώστη, αυτός αρχίζει να μεταδίδει συνεχώς το σειριακό του αριθμό.

- **Πομποδέκτες μίας εγγραφής-πολλών αναγνώσεων (Write Once-Read Many, WORM)**

Οι πομποδέκτες μιας εγγραφής/πολλών αναγνώσεων μπορούν να εγγραφούν, αλλά μόνο μία φορά. Αυτό επιτρέπει στο χρήστη να εγγράψει στον πομποδέκτη το σειριακό αριθμό που ο ίδιος επιθυμεί, συνήθως όταν ο πομποδέκτης ενσωματώνεται στο προϊόν που πρέπει να αναγνωρισθεί.

Οι πομποδέκτες αυτής της κατηγορίας προτιμούνται σε πολλές εφαρμογές, επειδή τους δίνεται η επιλογή ορισμού των σειριακών αριθμών αναγνώρισης αντί των αριθμών που προγραμματίστηκαν από την κατασκευάστρια εταιρεία. Από τη στιγμή που προγραμματιστεί ένας WORM πομποδέκτης, χάνεται βέβαια η δυνατότητα επανεγγραφής του.

- **Πομποδέκτες ανάγνωσης-εγγραφής (Read-Write)**

Οι πομποδέκτες ανάγνωσης/εγγραφής μπορούν να επανεγγραφούν πολλές φορές χωρίς περιορισμό. Τα συγκεκριμένα tags είναι τα πιο ακριβά, αλλά προσφέρουν τη μεγαλύτερη λειτουργικότητα. Επίσης, τυπικά διαθέτουν μεγαλύτερη χωρητικότητα μνήμης από τους άλλους τύπους πομποδεκτών.

Τα δεδομένα που αποθηκεύονται στις ετικέτες αποτελούνται από ένα μοναδικό αναγνωριστικό και μπορούν να περιλαμβάνουν ένα λειτουργικό σύστημα, μία αποθήκη δεδομένων (πητική ή όχι) και έναν ηλεκτρονικό κώδικα προϊόντων (Electronic Product Code - EPC). Το μέγεθος των δεδομένων που μια ετικέτα RFID έχει την δυνατότητα να αποθηκεύσει καθορίζεται από τον εκάστοτε προμηθευτή αλλά και την ίδια την εφαρμογή, με

ανώτερο όριο αποθήκευσης τα 2KB χωρητικότητα αρκετή για να αποθηκευτούν τα απαραίτητα δεδομένα του κάθε αντικειμένου.

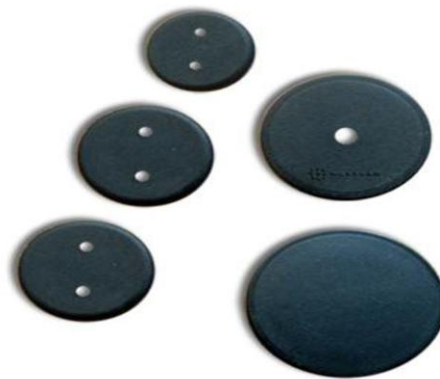
2.3.3 Κατηγοριοποίηση ανάλογα με την κατασκευή τους

Κάθε πομποδέκτης έχει κάποια κατασκευή που ενσωματώνει την κεραία και το ολοκληρωμένο του κύκλωμα, ενώ επιτρέπει την τοποθέτησή του πάνω στο προϊόν που πρέπει να αναγνωριστεί. Διατίθενται σε μια ποικιλία σχημάτων και μεγεθών, προκειμένου να μπορούν να χρησιμοποιηθούν σε πολλές διαφορετικές εφαρμογές.

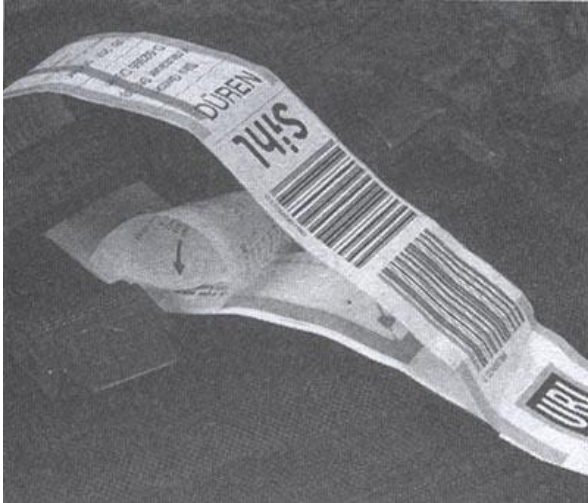


Εικόνα 4: Διάφορες κατασκευές πομποδεκτών

Ένα ευρέως χρησιμοποιούμενο είδος πομποδέκτη είναι ο **δίσκος** (disk) (εικόνα 5), μια στρογγυλή θερμοπλαστικά διαμορφωμένη (ABS) κατασκευή, με διάμετρο που κυμαίνεται από λίγα χιλιοστά μέχρι 10 cm. Αντί της θερμοπλαστικής διαμόρφωσης, μπορεί να χρησιμοποιηθεί εποξειδική ρητίνη, ώστε να επιτευχθεί ευρύτερο εύρος θερμοκρασίας λειτουργίας. Συνήθως, έχει μια οπή για βίδωμα στερέωσης στο κέντρο, ώστε να μπορεί για παράδειγμα να τοποθετηθεί πάνω σε μία παλέτα. Ορισμένοι πομποδέκτες σχήματος δίσκου έχουν μέγεθος ενός κουμπιού πουκαμίσου και σχεδιάζονται για να ραφτούν πάνω σε ένα ένδυμα ή πάνω σε ετικέτα ρούχου και συνήθως αφαιρούνται μετά την αγορά του ενδύματος.



Εικόνα 5: Πομποδέκτες σε μορφή δίσκου



Ένα άλλο πού συχνά χρησιμοποιούμενο είδος πομποδέκτη είναι οι **έξυπνες ετικέτες** (smart labels) που είναι κοινές χάρτινες ή πλαστικές ετικέτες στις οποίες ενσωματώνεται ένας πομποδέκτης RFID τύπου **inlay**. Το inlay έχει τη μορφή ενός πλαστικού αυτοκόλλητου φιλμ επί του οποίου κατασκευάζεται το τσιπ και η κεραία του πομποδέκτη με μεταξοτυπία ή χάραξη. Στην αγορά διατίθενται ειδικοί εκτυπωτές θερμικής μεταφοράς, οι οποίοι τυπώνουν στο χάρτινο ή πλαστικό μέρος της έξυπνης ετικέτας τα barcodes και τις υπόλοιπες πληροφορίες, ενώ παράλληλα προγραμματίζουν τον πομποδέκτη.

Εικόνα 6: Έξυπνη Ετικέτα σε εφαρμογή διαχείρισης αποσκευών στα αεροδρόμια

Οι έξυπνες ετικέτες παρακάμπτον το σοβαρό περιορισμό των RFID της μη διαθεσιμότητας πληροφοριών που μπορούν να διαβαστούν από τον άνθρωπο κι αναμένεται να επικρατήσουν σε εφαρμογές διακίνησης και αποθήκευσης των ομαδικών συσκευασιών των προϊόντων στην εφοδιαστική αλυσίδα. Λειτουργούν στα 13,56 MHz, στη ζώνη 860-930 MHz και στα 2,45GHz.

Μια παραλλαγή των έξυπνων ετικετών είναι οι **έξυπνες κάρτες μη επαφής** (contactless smart cards), που εφαρμόζονται κυρίως στον έλεγχο πρόσβασης (εικόνα 7).



Εικόνα 7: Έξυπνη RFID κάρτα μη επαφής

Ένα άλλο ενδιαφέρον είδος πομποδέκτη είναι οι **γυάλινοι σωλήνες** (glass tubes). Πρόκειται για συσκευές πολύ μικρές με μήκος περίπου 30 mm που έχουν κατασκευασθεί έτσι ώστε να μπορούν να εισαχθούν με ένεση κάτω από το δέρμα ζώου. Στις εφαρμογές αναγνώρισης ζώων χρησιμοποιούνται κι άλλα είδη πομποδεκτών, ανάμεσά τους ετικέτες περιλαίμιου (collar tags), ear tags και κεραμικές ετικέτες (ceramic tags) οι οποίες έχουν κατασκευασθεί έτσι να καταπίνονται από το ζώο και να παραμένουν μόνιμα μέσα του. Άλλοι πομποδέκτες έρχονται σε μορφή πολύ μικρών δαχτυλιδιών που τοποθετούνται στα πόδια περιστεριών σε αγώνες, ώστε να μπορεί να αναγνωριστεί ο ακριβής χρόνος που φθάνει το περιστέρι στη γραμμή τερματισμού.



Εικόνα 8: Glass Tube πομποδέκτης

Καθώς αναπτύσσονται νέες εφαρμογές των συστημάτων RFID, αναπτύσσονται παράλληλα τεχνολογίες κατασκευής εξειδικευμένων πομποδεκτών που καλύπτουν τις αντίστοιχες ανάγκες (π.χ. υψηλές μηχανικές καταπονήσεις, ακραίες κλιματικές συνθήκες). Ένα βήμα για τη σμίκρυνση των πομποδεκτών είναι η ενσωμάτωση του πηνίου μέσα στο ολοκληρωμένο (**coil-on-chip**), η οποία επιτυγχάνεται με μία ειδική μικρογαλβανική διαδικασία που λαμβάνει χώρα σε μία φέτα CMOS διαστάσεων 3mm x 3mm περίπου.

Τέλος, υπάρχουν πομποδέκτες που μπορούν να τοποθετηθούν πάνω στο αντικείμενο με τέτοιο τρόπο, ώστε να υπάρχει κάποιο κενό διάστημα ανάμεσα στην κεραία του πομποδέκτη και στο αντικείμενο. Αυτός ο τρόπος τοποθέτησης προτιμάται στις εφαρμογές αναγνώρισης μεταλλικών αντικειμένων (που ανακλούν τα ραδιοκύματα) κι αντικειμένων με υγρό περιεχόμενο (που απορροφούν τα ραδιοκύματα).

2.3.4 Κατηγοριοποίηση ανάλογα με την εφαρμογή τους

Μια άλλη σημαντική κατηγοριοποίηση που μπορούμε να διακρίνουμε στις ετικέτες RFID σχετίζεται με την εφαρμογή τους. Δεδομένου ότι τα συστήματα RFID έχουν εφαρμογή σε διάφορους τομείς στην καθημερινή ζωή του σύγχρονου ανθρώπου, η κατασκευή των ετικετών RFID αλλάζει ανάλογα με τις εφαρμογές και τις ανάγκες που χρειάζεται κάθε φορά, να καλύψει. Εφαρμογές RFID μπορεί κανείς να συναντήσει στους ακόλουθους τομείς:

✓ **Αδιάκοπη παρακολούθηση της ροής των προϊόντων (Object ID-Tracking)**

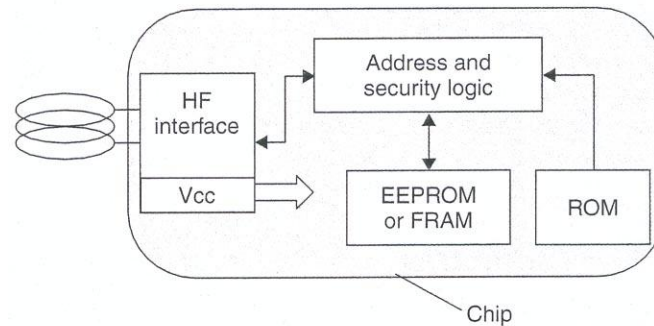
Βασίζονται σε: Ετικέτες / Αυτοκόλλητα, Hard-Tags

Εφαρμογές: Ιατρεία, Νοσοκομεία, Φαρμακευτική βιομηχανία, Διαχείριση λημμάτων, Τράπεζες, Αεροπορικές εταιρείες (παρακολούθηση αποσκευών) κλπ

- ✓ **Παρακολούθηση ζώων (Animal Tracking)**
 Βασίζονται σε: Hard-Tags, Μοσχεύματα
 Εφαρμογές: Γεωργία, Αγροτικές δραστηριότητες, Βιομηχανία τροφίμων
- ✓ **Παρακολούθηση ανθρώπων (Human-Tracking)**
 Βασίζονται σε: Hard-Tags
 Εφαρμογές: Για τον ιατρικό κλάδο, Ιατρεία, Νοσοκομεία, Συστήματα ασφαλείας, Δικαστήρια, Χώρους ψυχαγωγίας και αναψυχής
- ✓ **Παρακολούθηση εγγράφων και βιβλίων (Document / Book-Tracking)**
 Βασίζονται σε: Ετικέτες / Αυτοκόλλητα, RFID-Tags ενσωματωμένα σε χαρτί
 Εφαρμογές: Δημόσιες υπηρεσίες, Κρατικούς φορείς, Μεγάλες εταιρείες, Τράπεζες, Βιβλιοθήκες, Τεχνικά γραφεία
- ✓ **Συστήματα εισιτηρίων για ελέγχους εισόδων / ασύρματη μετάδοση πληροφοριών (Ticketing Solutions)**
 Βασίζονται σε: Χάρτινα εισιτήρια, Πλαστικές κάρτες, Hard-tags
 Εφαρμογές: Διόδια, Μεταφορικά μέσα, Δημόσιους χώρους στάθμευσης, Δημόσιες υπηρεσίες, Εισιτήρια θεαμάτων, Χιονοδρομικά κέντρα
- ✓ **Ηλεκτρονική προστασία προϊόντων, Πρόληψη πειρατείας (Electronic Product Protection)**
 Βασίζονται σε: Ετικέτες / Αυτοκόλλητα, Hard-Tags
 Εφαρμογές: Λιανικό και χονδρικό εμπόριο, Κατασκευαστές
- ✓ **Αυτόματη απογραφή αποθεμάτων (Automatic Inventory)**
 Βασίζονται σε: Ετικέτες / Αυτοκόλλητα, Hard-Tags
 Εφαρμογές: Λιανικό και χονδρικό εμπόριο, Κατασκευαστές
- ✓ **Υπολογισμός και καταγραφή θερμοκρασιών (Temperature Data Collection)**
 Βασίζονται σε: Ετικέτες / Αυτοκόλλητα, Hard-Tags
 Εφαρμογές: Λιανικό/χονδρικό εμπόριο, Φαρμακευτική βιομηχανία, Βιομηχανία τροφίμων
- ✓ **Συστήματα πρόσβασης για εισόδους (Entrance Solutions)**
 Βασίζονται σε: Ετικέτες / Αυτοκόλλητα, Hard-Tags
 Εφαρμογές: Ξενοδοχεία, διαμερίσματα και άλλους χώρους
- ✓ **Συλλογή δεδομένων εξ αποστάσεως (Remote Data Collection)**
 Βασίζονται σε: Αισθητήρες
 Εφαρμογές: Επιστημονικές εφαρμογές (πχ. σεισμολογία), Συστήματα ασφαλείας, Λιανικό/χονδρικό εμπόριο, Φαρμακευτική βιομηχανία, Βιομηχανία τροφίμων

2.3.5 Κύκλωμα πομποδέκτη (RFID Tag Circuit)

Το ολοκληρωμένο κύκλωμα (IC) του πομποδέκτη αποτελείται από τη μονάδα υψηλών συχνοτήτων (HF Interface), μία ή περισσότερες μονάδες μνήμης και το ψηφιακό κύκλωμα (Address and Security Logic).



Εικόνα 9: Block Διάγραμμα ολοκληρωμένου κυκλώματος του πομποδέκτη

Παρακάτω γίνεται επιμέρους ανάλυση των μονάδων του block διαγράμματος (εικόνα 9) ενός RFID tag.

✓ Μονάδα Υψηλών Συχνοτήτων (HF Interface)

Η μονάδα υψηλών συχνοτήτων εκτελεί τις λειτουργίες ενός κλασικού modem (διαμορφωτή- αποδιαμορφωτή).

Αποδιαμορφώνει το διαμορφωμένο σήμα από τον αναγνώστη και μεταφέρει τη ψηφιακή πληροφορία στο ψηφιακό κύκλωμα για περαιτέρω επεξεργασία. Επίσης, διαμορφώνει τα δεδομένα που προορίζονται για τον αναγνώστη μέσω ενός διαμορφωτή φορτίου ή ενός διαμορφωτή ανάκλασης ανάλογα με τον τρόπο μεταφοράς των δεδομένων προς τον αναγνώστη. Επίσης, τροφοδοτεί με ισχύ το υπόλοιπο ολοκληρωμένο κύκλωμα μέσω της ισχύος του σήματος που λαμβάνει η κεραία από τον πομποδέκτη.

✓ Μνήμη

Η μνήμη των εγγράψιμων πομποδεκτών κυμαίνεται από 1 byte μέχρι 64 Kbytes και συχνά χωρίζεται σε τμήματα (blocks). Ένα block μνήμης αποτελείται από ένα προκαθορισμένο αριθμό bytes, που διαβάζονται κι εγγράφονται ως μια ξεχωριστή μονάδα. Τα μεγέθη των blocks μνήμης που χρησιμοποιούνται σήμερα είναι 16 bits, 4 bytes ή 16 bytes. Η δόμηση της μνήμης σε blocks επιτρέπει την επιλεκτική ανάγνωση κι εγγραφή από τον αναγνώστη. Οι εγγράψιμοι πομποδέκτες χρησιμοποιούν μνήμες RAM, EEPROM, FRAM ή SRAM.

Η μνήμη RAM χρησιμοποιείται κυρίως για την προσωρινή αποθήκευση δεδομένων για όσο χρονικό διάστημα δηλαδή ο πομποδέκτης λαμβάνει σήμα από τον αναγνώστη. Ωστόσο, στους ενεργητικούς πομποδέκτες, η RAM σε συνδυασμό με

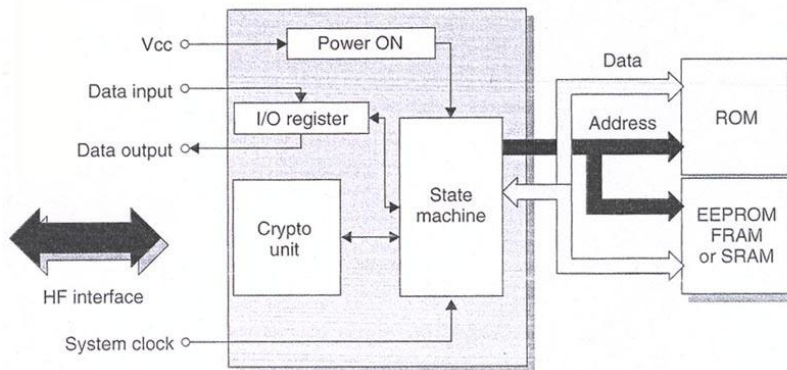
την ενσωματωμένη μπαταρία έχει τη δυνατότητα μακροχρόνιας αποθήκευσης δεδομένων.

Οι μνήμες EEPROM και FRAM έχουν τη δυνατότητα αποθήκευσης δεδομένων για μεγάλες χρονικές περιόδους, μέχρι και δέκα χρόνια περίπου χωρίς παροχή ισχύος (παθητικοί πομποδέκτες), σε αντίθεση με τη SRAM που απαιτεί μόνιμη παροχή ισχύος μέσω μιας βοηθητικής μπαταρίας (ενεργητικοί πομποδέκτες). Η EEPROM και η FRAM χρησιμοποιούνται στα επαγωγικά συζευγμένα συστήματα, ενώ η SRAM στα μικροκυματικά συστήματα.

Η λειτουργία εγγραφής στη μνήμη FRAM, σε αντίθεση με την EEPROM, γίνεται με πολύ μεγάλη ταχύτητα, με τυπικούς χρόνους εγγραφής γύρω στα 0,1 msec. Επίσης, η FRAM υπερτερεί της EEPROM και στην κατανάλωση ισχύος, με αποτέλεσμα η FRAM να θεωρείται πιο κατάλληλη σε σχέση με την EEPROM για τα παθητικά συστήματα RFID.

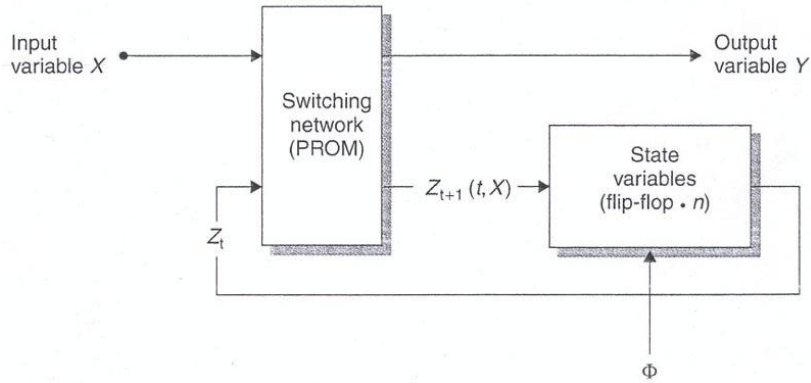
✓ Ψηφιακό Κύκλωμα (*Address and Security Logic*)

Το ψηφιακό κύκλωμα αποτελεί τον πυρήνα του ολοκληρωμένου κυκλώματος. Είναι υπεύθυνο για τον έλεγχο της πρόσβασης στη μνήμη, προκειμένου να εκτελεστούν εντολές επιλεκτικής ανάγνωσης κι εγγραφής. Επίσης, ελέγχει τις λειτουργίες αυθεντικοποίησης, κρυπτογράφησης δεδομένων και διαχείρισης του κρυπτολογικού κλειδιού μέσω μιας πρόσθετης κρυπτολογικής μονάδας δεδομένων.



Εικόνα 10: Το μπλοκ διάγραμμα του ψηφιακού κυκλώματος

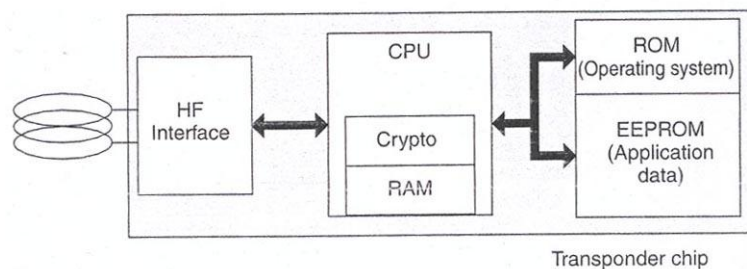
Στα πιο απλά συστήματα, οι παραπάνω λειτουργίες επιτελούνται από μία μηχανή κατάστασης (state machine), ενώ στα προγραμματιζόμενα συστήματα από έναν μικροεπεξεργαστή (microprocessor). Η μηχανή κατάστασης (εικόνα 11) είναι μια διάταξη που χρησιμοποιείται για την εκτέλεση λογικών λειτουργιών κι έχει την ικανότητα αποθήκευσης μεταβλητών καταστάσεων. Η μεταβλητή εξόδου Y εξαρτάται από τη μεταβλητή εισόδου X καθώς και από την προηγούμενη κατάσταση που αντιπροσωπεύεται από τη μεταβλητή κατάσταση των flip flops. Η μηχανή κατάστασης περνά από διαφορετικές καταστάσεις. Η μετάβαση από κατάσταση σε κατάσταση μπορεί να απεικονιστεί σε ένα διάγραμμα καταστάσεων.



Εικόνα 11: Το block διάγραμμα μιας μηχανής κατάστασης

Μια μηχανή κατάστασης μπορεί να επιτελέσει πολύ σύνθετες λειτουργίες, ωστόσο ο προγραμματισμός και η ακολουθία των λειτουργιών και προγραμμάτων της δεν είναι δυνατό να αλλαχθούν διότι έχουν ήδη καθοριστεί κατά την κατασκευή του ολοκληρωμένου κυκλώματος. Το πρόβλημα αυτό μπορεί να αντιμετωπιστεί με τη χρήση μικροεπεξεργαστή. Ένας μικροεπεξεργαστής, όπως ο 8051 ή ο 6805, διαθέτει το δικό του λειτουργικό σύστημα, οπότε οι διαδικασίες των λειτουργιών του μικροεπεξεργαστή αλλάζουν μέσω τροποποίησης του λογισμικού.

Το λειτουργικό κύκλωμα του μικροεπεξεργαστή είναι υπεύθυνο για την εκτέλεση των εντολών του αναγνώστη, τη διαχείριση ενεργειών και την εκτέλεση κρυπτογραφικών αλγορίθμων (π.χ. κρυπτογράφηση, αυθεντικοποίηση). Σε ορισμένα ολοκληρωμένα κυκλώματα, οι κρυπτογραφικές λειτουργίες δεν εκτελούνται από τον μικροεπεξεργαστή, αλλά από ειδικούς μαθηματικούς συνεπεξεργαστές που έχουν τη δυνατότητα γρήγορης εκτέλεσης των υπολογισμών που απαιτούνται στις διαδικασίες κρυπτογράφησης.



Εικόνα 12 Το block διάγραμμα του κυκλώματος ενός πομποδέκτη με μικροεπεξεργαστή

2.4 Αναγνώστες (RFID Readers)

Οι αναγνώστες παρεμβάλλονται ανάμεσα στους πομποδέκτες και το σύστημα διαχείρισης της πληροφορίας.

Συνοπτικά, οι κύριες λειτουργίες του αναγνώστη είναι η ενεργοποίηση των πομποδεκτών, ο έλεγχος της επικοινωνίας μαζί τους, καθώς και η ανταλλαγή δεδομένων ανάμεσα στην εφαρμογή λογισμικού και τον πομποδέκτη. Επίσης, η δημιουργία σύνδεσης και η εκτέλεση ορισμένων διαδικασιών που υπάρχουν σε μερικά RFID συστήματα, όπως η αποφυγή σύγκρουσης (anticollision) και η αυθεντικοποίηση (authentication), ελέγχονται εξολοκλήρου από τον αναγνώστη.

Σε γενικές γραμμές μπορούμε να πούμε ότι οι πομποδέκτες ενεργοποιούνται λαμβάνοντας ενέργεια από το ηλεκτρομαγνητικό πεδίο που αναπτύσσεται γύρω από την κεραία του αναγνώστη, οπότε έχουμε ηλεκτρομαγνητική σύζευξη αναγνώστη και πομποδέκτη αντίστοιχα. Το είδος του πεδίου που αναπτύσσεται εξαρτάται από τα κυκλωματικά στοιχεία της κεραίας του αναγνώστη και από τη συχνότητα λειτουργίας του συστήματος.

Από τη στιγμή που ο πομποδέκτης βρεθεί μέσα στο πεδίο του αναγνώστη, μια εντολή από την εφαρμογή λογισμικού προς τον αναγνώστη αρχικοποιεί μια σειρά διαδοχικών βημάτων επικοινωνίας ανάμεσα στον πομποδέκτη και τον αναγνώστη. Η επικοινωνία πομποδέκτη-αναγνώστη βασίζεται σε συγκεκριμένα πρωτόκολλα επικοινωνίας που καθορίζουν τις εντολές επικοινωνίας μεταξύ τους.

Στο παρακάτω παράδειγμα *εικόνα 13*, μια εντολή ανάγνωσης οδηγεί πρώτα στην ενεργοποίηση του πομποδέκτη, ακολουθούμενη από την εκτέλεση της διαδικασίας αυθεντικοποίησης και τελικά τη μετάδοση των ζητούμενων δεδομένων.

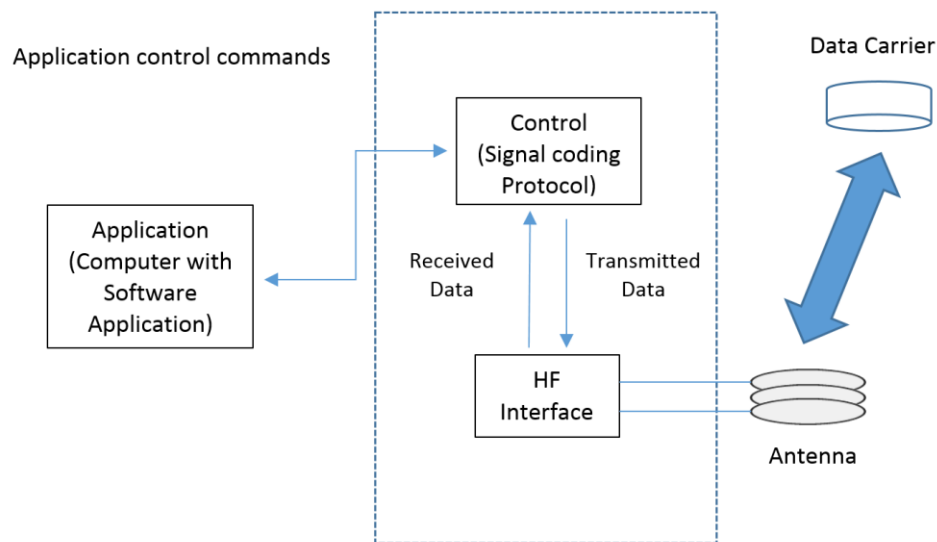
Application ↔ Reader	Reader ↔ Transponder	Comment
➡ Blockread_Address [00]		Read Transponder memory [address]
	➡ Request	Transponder in the field?
	⬅ ATR_SNR[4712]	Transponder operates with Serial Number
	➡ GET_Random	Initiate authentication
	⬅ Random[081514]	
	➡ SEND_Token1	
	⬅ GET_Token2	Authentication successfully completed
	➡ Read_@[00]	Read command [address]
	⬅ Data[9876543210]	Data from Transponder
⬅ Data[9876543210]		Data to application

Εικόνα 13: Παράδειγμα εκτέλεσης μιας εντολής ανάγνωσης

2.4.1 Οι υπομονάδες ενός RFID αναγνώστη

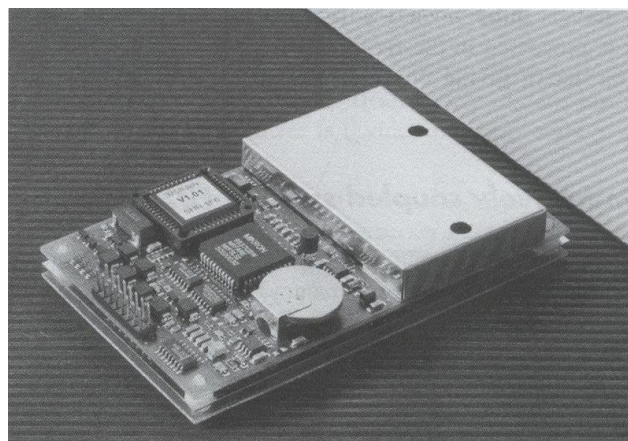
Σε όλα τα RFID συστήματα, οι αναγνώστες έχουν κοινή βασική αρχή λειτουργίας και επομένως παρόμοιο σχεδιασμό.

Αποτελούνται από μία κεραία που λειτουργεί ως πομπός και δέκτης και τον controller, ο οποίος περιλαμβάνει μία μονάδα ελέγχου (control system) και μία μονάδα υψηλών συχνοτήτων (HF interface). Η μεταφορά των δεδομένων από τον αναγνώστη στο σύστημα διαχείρισης της πληροφορίας γίνεται μέσω μιας θύρας (π.χ. RS232 ή RS485), ενώ η κεραία μπορεί να είναι ενσωματωμένη ή όχι στο υπόλοιπο σύστημα. Παρακάτω στην *εικόνα 14* φαίνεται το μπλοκ διάγραμμα ενός αναγνώστη.

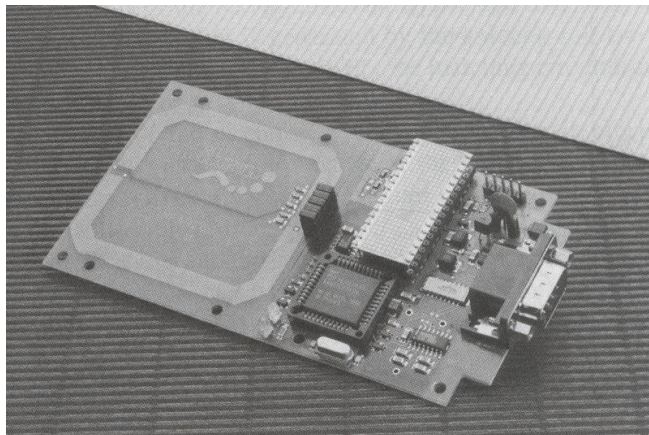


Εικόνα 14: Το μπλοκ διάγραμμα ενός RFID αναγνώστη

Στην *εικόνα 15* στο δεξιό μέρος φαίνεται η μονάδα υψηλών συχνοτήτων, ενώ στο αριστερό η μονάδα ελέγχου.



Εικόνα 15: Παράδειγμα ενός RFID αναγνώστη

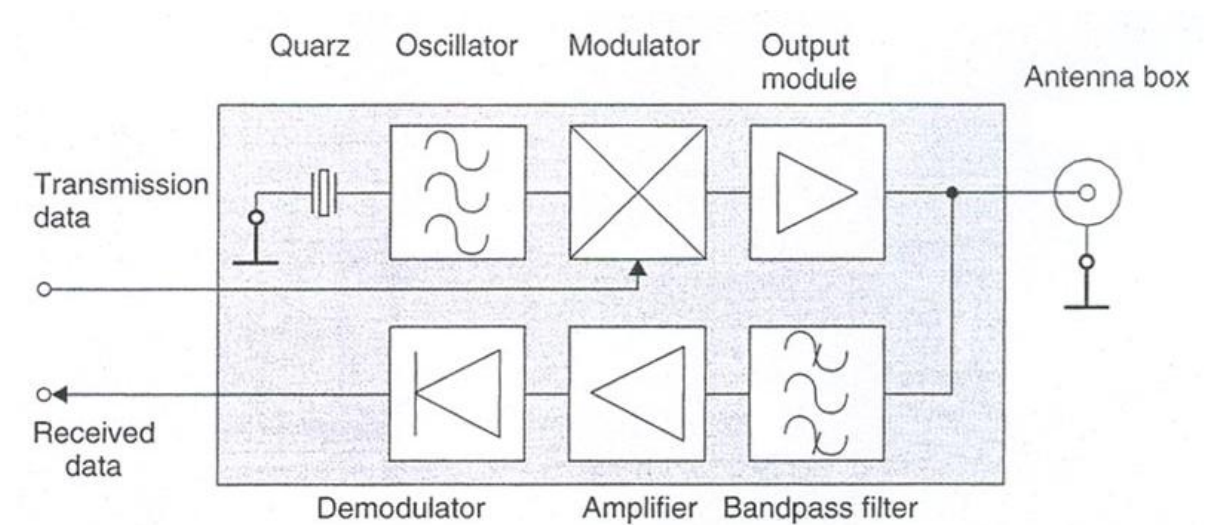


Εικόνα 16: Ένα παράδειγμα αναγνώστη με την κεραία ενσωματωμένη στο υπόλοιπο σύστημα.

2.4.2 Μονάδα υψηλών συχνοτήτων

Η μονάδα υψηλών συχνοτήτων του αναγνώστη παράγει την ισχύ υψηλής συχνότητας μετάδοσης που είναι απαραίτητη για την ενεργοποίηση και τη μετάδοση ισχύος στον πομποδέκτη. Κατά τη λειτουργία της εγγραφής, είναι υπεύθυνη για τη διαμόρφωση του σήματος μετάδοσης δεδομένων προς τον πομποδέκτη. Επιπλέον, λαμβάνει και αποδιαμορφώνει τα σήματα υψηλών συχνοτήτων από τον πομποδέκτη.

Τα δεδομένα που πρόκειται να μεταδοθούν προς τον πομποδέκτη περνούν από διαφορετικά κυκλώματα (κανάλι μετάδοσης) σε σχέση με τα λαμβανόμενα από τον πομποδέκτη (κανάλι λήψης), προκειμένου να υποστούν διαφορετική επεξεργασία. Παρακάτω, θα γίνει ανάλυση των δύο καναλιών σήματος με μεγαλύτερη λεπτομέρεια, λαμβάνοντας υπόψιν τις διαφορές ανάμεσα στα διάφορα είδη συστημάτων RFID.



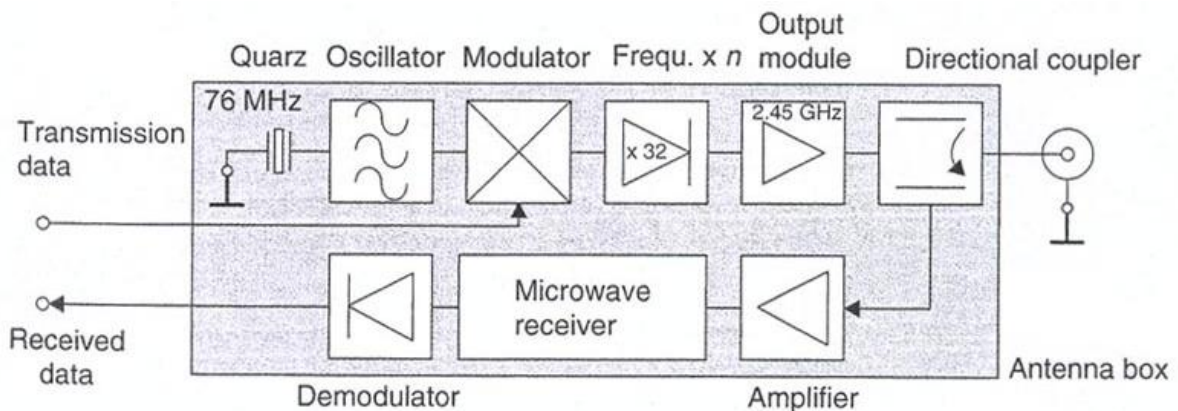
Εικόνα 17: Το μπλοκ διάγραμμα μιας μονάδας υψηλών συχνοτήτων για ένα επαγωγικά συζευγμένο σύστημα RFID.

Στο κανάλι μετάδοσης των επαγωγικών συζευγμένων συστημάτων παράγεται αρχικά ένα σήμα της απαιτούμενης συχνότητας λειτουργίας από ένα σταθερό ταλαντωτή πυριτίου. Υπάρχουν υψηλές απαιτήσεις για τη σταθερότητα της φάσης και του θορύβου πλευρικών ζωνών στο συγκεκριμένο ταλαντωτή, έτσι ώστε να μη χειροτερεύει ο σηματοθορυβικός λόγος του λαμβανόμενου σήματος από τον πομποδέκτη, το οποίο είναι εξαιρετικά αδύναμο. Στη συνέχεια, το σήμα του ταλαντωτή τροφοδοτείται σε μια μονάδα διαμόρφωσης ελεγχόμενη από το σήμα βασικής ζώνης που έχει ήδη δημιουργηθεί από τη μονάδα κωδικοποίησης σήματος στη μονάδα ελέγχου του αναγνώστη. Ανάλογα με τον τύπο του διαμορφωτή, το σήμα του ταλαντωτή υπόκειται σε ASK ή PSK διαμόρφωση. Επίσης και η FSK διαμόρφωση είναι δυνατή, κατά την οποία το σήμα βασικής ζώνης τροφοδοτείται άμεσα σε μια γεννήτρια σύνθεσης συχνότητας. Έπειτα το διαμορφωμένο σήμα ενισχύεται και μεταφέρεται στην κεραία για να μεταδοθεί.

Το πρώτο μέρος του καναλιού λήψης των επαγωγικά συζευγμένων συστημάτων, είναι ένα όσο γίνεται πιο απότομο ζωνοπερατό φίλτρο, το οποίο διαχωρίζει το λαμβανόμενο από τον πομποδέκτη (μέσω της κεραίας) σήμα από το ισχυρό σήμα εκπομπής του αναγνώστη. Η επιλογή του ζωνοπερατού φίλτρου είναι πολύ σημαντική, γιατί ανάλογα με τον τρόπο μεταφοράς των δεδομένων από τον πομποδέκτη στον αναγνώστη, υπάρχει περίπτωση η συχνότητα μετάδοσης να είναι πολύ κοντά στη συχνότητα λήψης.

Μετά το φίλτρο, ακολουθεί η ενίσχυση και η αποδιαμόρφωση του σήματος, το οποίο, όπως θα δούμε στη συνέχεια, αποκωδικοποιείται στη μονάδα ελέγχου και στη συνέχεια μεταφέρεται στο σύστημα διαχείρισης της πληροφορίας.

Στα RFID συστήματα που λειτουργούν με ηλεκτρομαγνητική σύζευξη πομποδέκτη-αναγνώστη, η συχνότητα λειτουργίας είναι πιο υψηλή από αυτή των επαγωγικά συζευγμένων συστημάτων, τυπικά 2.45 GHz (μικροκυματικά συστήματα). Η διαφορά λοιπόν στο κανάλι μετάδοσης των συστημάτων αυτών είναι ότι η συχνότητα λειτουργίας παράγεται μέσω μιας διαδικασίας πολλαπλασιασμού μιας μικρότερης συχνότητας ενός ταλαντωτή. Επίσης, ορισμένα μικροκυματικά συστήματα διαθέτουν ένα κατευθυντικό συζεύκτη για το διαχωρισμό του εκπεμπόμενου από το λαμβανόμενο σήμα.



Εικόνα 18: Το μπλοκ διάγραμμα μιας μονάδας υψηλών συχνοτήτων στα μικροκυματικά συστήματα

2.4.3 Μονάδα ελέγχου

Η μονάδα ελέγχου του αναγνώστη είναι υπεύθυνη για την επικοινωνία του αναγνώστη με το σύστημα διαχείρισης της πληροφορίας και για την εκτέλεση των εντολών που δίνονται από το τελευταίο, οπότε περιέχει μια θύρα RS232 ή RS485. Επίσης, ελέγχει την επικοινωνία με τον πομποδέκτη (αρχή αφέντη-δούλου) μέσω συγκεκριμένων πρωτοκόλλων επικοινωνίας. Τέλος, κωδικοποιεί κι αποκωδικοποιεί τα σήματα δεδομένων. Η κωδικοποίηση των δεδομένων που στέλνονται από τον αναγνώστη στον πομποδέκτη γίνεται με κωδικοποίηση NRZ, σε αντίθεση με την αποστολή δεδομένων από τον πομποδέκτη στον αναγνώστη που γίνεται με κωδικοποίηση Manchester.

Στα πιο σύνθετα συστήματα, η μονάδα ελέγχου εκτελεί επιπρόσθετες λειτουργίες, όπως την εκτέλεση ενός αλγορίθμου αποφυγής σύγκρουσης, την κρυπτογράφηση/αποκρυπτογράφηση των δεδομένων που μεταφέρονται ανάμεσα στον πομποδέκτη και τον αναγνώστη, καθώς και την εκτέλεση της διαδικασίας αυθεντικοποίησης ανάμεσα στον πομποδέκτη και τον αναγνώστη. Αυτές οι σύνθετες λειτουργίες εκτελούνται από ένα μικροεπεξεργαστή που περιέχεται μέσα στη μονάδα ελέγχου. Εξαιρέση αποτελούν οι κρυπτολογικές διαδικασίες, οι οποίες συνήθως εκτελούνται από μία πρόσθετη μονάδα ASIC, έτσι ώστε να μην επιφορτίζεται ο επεξεργαστής με σύνθετους και δύσκολους υπολογισμούς.

2.4.4 Είδη αναγνωστών

Υπάρχουν διάφοροι τύποι αναγνωστών που χρησιμοποιούνται σε διαφορετικές εφαρμογές. Στις βιομηχανικές εφαρμογές, χρησιμοποιούνται αναγνώστες που μπορούν να λειτουργήσουν σε δύσκολα περιβάλλοντα. Για παράδειγμα, υπάρχουν **βιομηχανικοί** αναγνώστες με ειδική προστασία από εκρήξεις, πτώσεις, επαφή με υγρά κ.τ.λ. Επίσης, διατίθενται οι **φορητοί** αναγνώστες, στους οποίους η κεραία είναι ενσωματωμένη στον αναγνώστη. Διαθέτουν μια LCD οθόνη, προκειμένου ο χρήστης να μπορεί να διαβάσει τα δεδομένα του πομποδέκτη κι ένα πληκτρολόγιο μέσω του οποίου ο χρήστης μπορεί να εισάγει δεδομένα και να λειτουργήσει τον αναγνώστη. Οι φορητοί αναγνώστες χρησιμοποιούνται σε εφαρμογές, όπου τα αντικείμενα θα πρέπει να αναγνωριστούν ξεχωριστά και οι χρήστες θα πρέπει να έχουν τη δυνατότητα άμεσης ανάγνωσης των δεδομένων των πομποδεκτών. Οι φορητοί αναγνώστες χρησιμοποιούνται σχεδόν με τον ίδιο τρόπο, όπως οι φορητοί αναγνώστες γραμμωτών κωδικών (barcode scanners), ενώ υπάρχουν αναγνώστες dual function που μπορούν να διαβάσουν τόσο γραμμωτούς κωδικούς, όσο και RFID πομποδέκτες. Οι φορητοί αναγνώστες χρησιμοποιούνται στην αναγνώριση ζώων, ως συσκευές ελέγχου στις δημόσιες μεταφορές, ως τερματικά στις πληρωμές, στη ενεργοποίηση συστημάτων, κ.τ.λ. Διαθέτουν μνήμη με μεγάλη σχετικά χωρητικότητα για την προσωρινή αποθήκευση των δεδομένων, τα οποία στη συνέχεια μεταφέρονται σε κάποιο σύστημα μέσω μιας θύρας RS232.

Ένας αναγνώστης μπορεί γενικά να διαβάσει ή να εγγράψει δεδομένα σε ένα μόνο τύπο πομποδέκτη, συνήθως της ίδιας κατασκευαστικής εταιρείας ή σε πομποδέκτες μίας συγκεκριμένης συχνότητας. Εξαιρέση αποτελούν οι λεγόμενοι **ευέλικτοι** αναγνώστες, οι οποίοι έχουν τη δυνατότητα να επικοινωνήσουν με διάφορους τύπους πομποδεκτών, είτε διαφορετικού κατασκευαστή είτε διαφορετικής συχνότητας, συμβάλλοντας στη συμβατότητα μεταξύ διαφορετικών RFID συστημάτων.

2.5 Σύστημα διαχείρισης & επεξεργασίας της πληροφορίας

Ο αναγνώστης συνδέεται με έναν τοπικό υπολογιστή με κατάλληλο σύστημα διαχείρισης και επεξεργασίας της πληροφορίας.

Το παραπάνω λογισμικό, που περιλαμβάνει μεταξύ άλλων βάσεις δεδομένων με τους σειριακούς αριθμούς των πομποδεκτών του συστήματος, δίνει εντολές ανάγνωσης κι εγγραφής σε έναν ή περισσότερους αναγνώστες και στη συνέχεια λαμβάνει τα αντίστοιχα δεδομένα. Επιπλέον, στον τοπικό υπολογιστή γίνεται η πρώτη φάση επεξεργασίας της τεράστιας σε όγκο πληροφορίας που προκύπτει. Για παράδειγμα, μία από τις πιο χρήσιμες λειτουργίες του τοπικού υπολογιστή είναι να απομακρύνει τα πολλαπλά αντίγραφα ανάγνωσης του ίδιου σειριακού αριθμού, όταν ο συγκεκριμένος σειριακός αριθμός αντιστοιχεί σε ένα αντικείμενο που δεν έχει αλλάξει θέση. Για παράδειγμα, γνωρίζουμε ότι οι πομποδέκτες μόνο ανάγνωσης μόλις βρεθούν στην περιοχή ανάγνωσης του αναγνώστη, μεταδίδουν συνεχώς το σειριακό τους αριθμό το χρονικό διάστημα που λαμβάνουν σήμα από τον αναγνώστη. Το αποτέλεσμα είναι να δημιουργούνται πολλαπλά άχρηστα αντίγραφα ανάγνωσης που θα πρέπει να απομακρυνθούν. Η απομάκρυνση των αντιγράφων ενός σειριακού αριθμού γίνεται με κριτήριο τη χρονική απόσταση μεταξύ τους.

Ακόμη, ο τοπικός υπολογιστής που τοποθετείται συνήθως κοντά στις εγκαταστάσεις του RFID συστήματος, π.χ. σε μια γραμμή παραγωγής, μπορεί να καθορίζει αν θα πρέπει να γίνει κάποια συγκεκριμένη ενέργεια σχετική με ένα αντικείμενο που αναγνωρίστηκε.

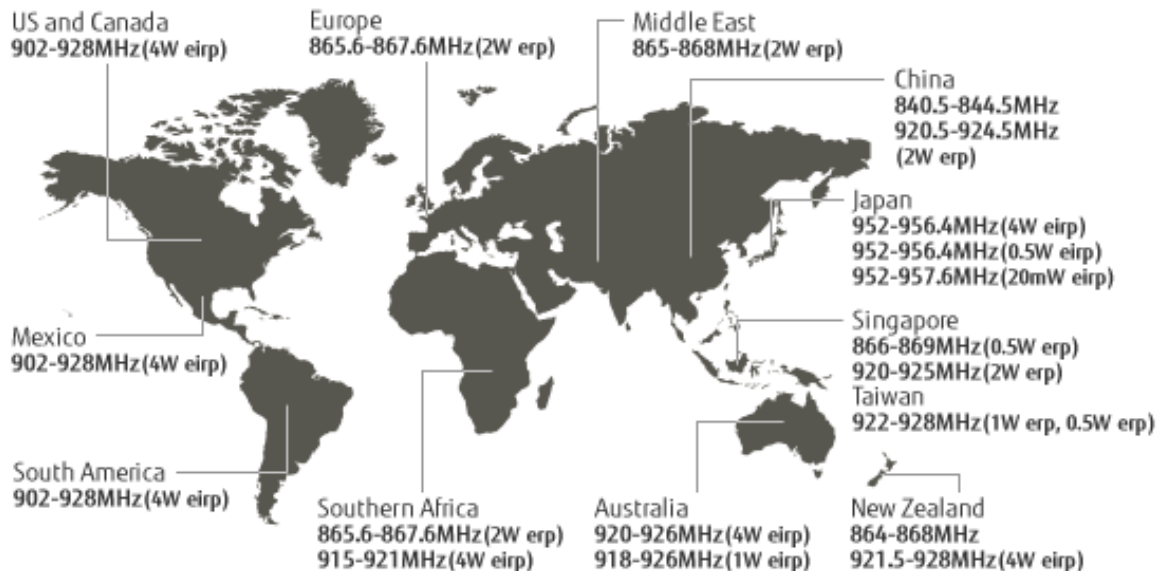
Επίσης, μπορεί να έχει την ικανότητα να ξεχωρίζει γενικότερα τη χρήσιμη από την άχρηστη πληροφορία και να μεταφέρει τη χρήσιμη σε κάποιο άλλο υπολογιστή ή σύστημα για να υποστεί περαιτέρω επεξεργασία. Κατ'αυτόν τον τρόπο, η πληροφορία που προκύπτει μέσω της ανάγνωσης, υπόκειται σε πολυεπίπεδη επεξεργασία μέσα σε ένα ευρύτερο δίκτυο υπολογιστών.

Τόσο ο τοπικός υπολογιστής, όσο και οι υπόλοιποι στο προαναφερθέν δίκτυο μπορούν να λειτουργήσουν κι ως ένα είδος «δρομολογητή», στέλνοντας τα δεδομένα που λαμβάνουν σε διαφορετικά τερματικά ανάλογα με το περιεχόμενό τους, ρυθμίζοντας κατ'αυτόν τον τρόπο τη ροή των δεδομένων του συστήματος.

Τα υψηλά επίπεδα επεξεργασίας συνήθως σχετίζονται με τη λήψη σοβαρών αποφάσεων σε θέματα στρατηγικής, και συνδιαλλαγής με άλλους οργανισμούς. Σε πολλές εφαρμογές, περιλαμβάνουν ειδικά συστήματα, όπως τα Συστήματα Διαχείρισης Επιχειρησιακών Πόρων (ERP), τα Συστήματα Διαχείρισης Αποθηκών (WMS) και τα Συστήματα Διαχείρισης Μεταφορών (TMS). Τα παραπάνω συστήματα λειτουργούν κι ανεξάρτητα από την RFID τεχνολογία. Ωστόσο, με τη ενσωμάτωση του RFID αυξάνεται η χρηστικότητα κι η αποδοτικότητά τους, διότι μπορούν να λάβουν πιο πολλά δεδομένα και μάλιστα σε μεγαλύτερη σε πραγματικό χρόνο και με μικρότερο κόστος.

2.6 Ιδιαίτερα χαρακτηριστικά των συχνοτήτων

Οι αναγνώστες RFID αποτελούνται από μία κεραία, η οποία αναλαμβάνει την επικοινωνία, μέσω ραδιοσυχνοτήτων, με τις ετικέτες. Καθώς και μία μονάδα ελέγχου, που εκτελεί δύο συγκεκριμένα έργα. Πρωτίστως τον καθορισμό των διάφορων ενεργειών (αποστολή/ λήψη σημάτων, ανάγνωση/ εγγραφή ετικετών κ.ά.) ενέργεια που πραγματοποιείται μέσω του ενδιάμεσου λογισμικό και δευτερευόντως την επικοινωνία με το πληροφοριακό σύστημα. Οι αναγνώστες RFID μπορούν να κατηγοριοποιηθούν σε σχέση με τις φυσικές τους διαστάσεις, την εφαρμογή τους και τις τεχνικές ιδιότητες σε "σταθερούς αναγνώστες", "ολοκληρωμένους αναγνώστες", "αναγνώστες χειρός" & σε "ενσωματωμένους αναγνώστες". Ένα σημαντικό κριτήριο για την επιλογή του συστήματος RFID, είναι η συχνότητα στην οποία επικοινωνούν ο αναγνώστης RFID με της ετικέτες. Πρέπει να δοθεί μεγάλη προσοχή για να αποφευχθούν «συγκρούσεις» συχνοτήτων, με άλλα ασύρματα συστήματα. Ο λόγος έγκειται στο γεγονός, ότι το RFID λειτουργεί στην ISM ζώνη συχνοτήτων, στην οποία λειτουργούν επίσης πολλές ιατρικές, βιομηχανικές και επιστημονικές εφαρμογές.



Εικόνα 19: Η εικόνα δείχνει ένα παγκόσμιο χάρτη, με τις συχνότητες λειτουργίας της RFID τεχνολογίας, που λειτουργούν στις χώρες αυτές. Δεν υπάρχει μια παγκόσμια συχνότητα, διότι κάθε χώρα χωρίζει διαφορετικά τις ζώνες συχνοτήτων της. Ορισμένες χρησιμοποιούν τη συχνότητα για διαφορετική εφαρμογή. Κάθε ζώνη συχνοτήτων έχει τη δική της χρήση, άρα και δικαιώματα.

Κάθε ζώνη συχνοτήτων έχει τα δικά της ιδιαίτερα χαρακτηριστικά, τα οποία θα πρέπει να ληφθούν υπόψιν κατά την επιλογή ενός συστήματος RFID. Στην παράγραφο αυτή, συγκρίνουμε τα συστήματα RFID διαφορετικών συχνοτήτων με βάση κάποια συγκεκριμένα χαρακτηριστικά.

- **Μέγιστη Απόσταση Ανάγνωσης**

Η συχνότητα λειτουργίας ενός συστήματος RFID καθορίζει τη μέγιστη απόσταση ανάγνωσης του συστήματος.

Πιο συγκεκριμένα τα συστήματα χαμηλών συχνοτήτων με συχνότητα λειτουργίας περίπου στα 125 KHz χαρακτηρίζονται από μέγιστη απόσταση ανάγνωσης μέχρι 50 cm. Τα συστήματα υψηλών συχνοτήτων με συχνότητα λειτουργίας περίπου στα 13,56 MHz χαρακτηρίζονται από μέγιστη απόσταση ανάγνωσης μέχρι 1 μέτρο. Τα συστήματα πολύ υψηλών συχνοτήτων με συχνότητες λειτουργίας, όπως 868 MHz (στην Ευρώπη) και 2,45 GHz (μικροκυματική συχνότητα) χαρακτηρίζονται από μέγιστη απόσταση ανάγνωσης γύρω στα 1 με 3 μέτρα, ενώ τα UHF συστήματα που λειτουργούν στη ζώνη με κεντρική συχνότητα τα 915 MHz (στις ΗΠΑ) μπορούν να αναγνώσουν δεδομένα σε αποστάσεις των 6 ή περισσότερων μέτρων.

Τα παραπάνω στοιχεία ισχύουν μόνο στα παθητικά συστήματα, διότι με τη χρήση ενεργητικών πομποδεκτών η μέγιστη απόσταση ανάγνωσης των συστημάτων αυξάνει κατά πολύ και μπορεί να φθάσει μέχρι 30 μέτρα.

- **Είδος Σύζευξης**

Τα συστήματα RFID χαμηλών κι υψηλών συχνοτήτων μέχρι 30 MHz λειτουργούν με επαγωγική σύζευξη, ενώ στις πολύ υψηλές και στις μικροκυματικές συχνότητες η σύζευξη γίνεται με χρήση ηλεκτρομαγνητικών πεδίων (ανακλαστική σύζευξη).

- **Ευαισθησία στον Προσανατολισμό**

Τα συστήματα RFID χαμηλών συχνοτήτων (τυπικά κάτω των 500 KHz) δεν είναι ευαίσθητα στον προσανατολισμό μεταξύ αναγνώστη-πομποδέκτη, σε αντίθεση με τα συστήματα υψηλότερων συχνοτήτων.

- **Απορρόφηση και Εξασθένηση**

Στα συστήματα χαμηλών συχνοτήτων, τα αντικείμενα που βρίσκονται στο χώρο (π.χ. μεταλλικές επιφάνειες ή υγρά) επηρεάζουν ελάχιστα τη λειτουργία του συστήματος, με δεδομένο ότι η απορρόφηση κι η εξασθένηση θεωρούνται πρακτικά αμελητέες. Ένα ενδεικτικό παράδειγμα είναι το bolus, ένας πομποδέκτης που τοποθετείται στο στομάχι των ζώων και μπορεί να διαβαστεί από RFID συστήματα συχνότητας κάτω των 135 KHz. Καθώς ανεβαίνει η συχνότητα, θα πρέπει να λαμβάνεται υπόψιν η επίδραση των αντικειμένων που βρίσκονται τόσο στην περιοχή ανάγνωσης όσο και στο ευρύτερο χώρο του συστήματος.

- **Ευαισθησία στο Θόρυβο και στις Ηλεκτρομαγνητικές Παρεμβολές**

Ένα άλλο σημαντικό χαρακτηριστικό είναι η ευαισθησία των συστημάτων στο θόρυβο και στις ηλεκτρομαγνητικές παρεμβολές, όπως αυτές που παράγονται από ηλεκτρικές μηχανές. Σε αντίθεση με τους επαγωγικούς πομποδέκτες, τα μικροκυματικά συστήματα πλεονεκτούν σε αυτόν τον τομέα, με αποτέλεσμα να προτιμούνται στους χώρους βιομηχανικής παραγωγής.

- **Ρυθμός Μετάδοσης Δεδομένων**

Τα συστήματα RFID χαμηλών συχνοτήτων (τυπικά κάτω των 500 KHz) χαρακτηρίζονται γενικά από μικρότερες ταχύτητες μετάδοσης δεδομένων σε σχέση με τα συστήματα υψηλότερων συχνοτήτων.

Συγκεκριμένα, η συχνότητα 13,56 MHz χαρακτηρίζεται από υψηλό ρυθμό μετάδοσης δεδομένων, γύρω στα 106 kbits/sec, όπως και η συχνότητα 27,125 MHz με ρυθμό μετάδοσης δεδομένων 424 kbits/sec (μεγάλο επιτρεπόμενο εύρος ζώνης).

- **Αντοχή σε Ακραίες Περιβαλλοντικές Συνθήκες**

Γνωρίζουμε ότι τα συστήματα HF μπορούν να λειτουργήσουν σε συνθήκες υψηλής υγρασίας και ότι τα μικροκυματικά συστήματα έχουν μεγάλη αντοχή στις υψηλές θερμοκρασίες.

Στον παρακάτω πίνακα 1, μπορεί κανείς να διαβάσει τις εφαρμογές των συστημάτων RFID ανάλογα με τη συχνότητα λειτουργίας τους και την ικανότητα επανεγγραφής τους.

	Συχνότητα 125 KHz Απόσταση ανάγνωσης γύρω στα <u>10 εκατοστά</u>	Συχνότητα 13,56 MHz Απόσταση ανάγνωσης <u>1 μέτρο</u> περίπου	Συχνότητα 900 MHz Απόσταση ανάγνωσης <u>3 μέτρα</u> περίπου
Είδος Πομποδέκτη			
Μόνο Ανάγνωσης	Εμφυτεύματα σε κατοικίδια Κτηνοτροφία Ακίνητοποιητές αυτοκινήτων	Βιβλιοθήκες Κτηνοτροφία	Βιβλιοθήκες Αναγνώριση Άγριων Ζώων, Πρατήρια Βενζίνης
Μίας Εγγραφής- Πολλών Αναγνώσεων	Σημεία Πώλησης, Πετροχημικά, Συσκευές Δανεισμού, Αναγνώριση Επικίνδυνων Υλικών	Ιχνηλασιμότητα στην Εφοδιαστική Αλυσίδα, Σημεία Πώλησης, Έλεγχος Πρόσβασης, Εντοπισμός Δεμάτων, Αναγνώριση Ασθενών, Εισιτήρια Δημόσιας Μεταφοράς, Κάρτες Ασφαλείας, Φιάλες Αίματος, Αναγνώριση Παγίων	Παλέτες και Επαναχρησιμοποιήσιμα Υλικά Συσκευασίας, Έλεγχος Πρόσβασης, Εντοπισμός Αποσκευών, Ασφάλεια Κειμένων, Εντοπισμός κι Έλεγχος Φαρμάκων, Διόδια Οχημάτων, Αναγνώριση Προσωπικού, Ταυτότητες και Διαβατήρια, Αναγνώριση Άγριων Ζώων, Διαχείριση Αποσκευών στα Αεροδρόμια
Ανάγνωσης- Εγγραφής	Σημεία Πώλησης (διαχείριση τιμών & εγγυήσεων) Δοχεία Χημικών	Εντοπισμός Δεμάτων, Σημεία Πώλησης, Βιβλιοθήκες (βιβλία, ταινίες, CD κτλ)	Εντοπισμός Αεροπορικού Φορτίου με συστήματα Αναδρομολόγησης, Βιβλιοθήκες

Πίνακας 1: Αντιστοίχισης RFID εφαρμογών με συχνότητα λειτουργίας και ικανότητα επανεγγραφής.

Παρακάτω αναφέρονται τα πλεονεκτήματα & μειονεκτήματα των περιοχών συχνοτήτων.

Ζώνη συχνοτήτων 100 – 135 KHz

Πλεονεκτήματα

- Χρήση χαμηλού κόστους παθητικών πομποδεκτών.
- Καλή διείσδυση στα μη μεταλλικά αντικείμενα, στο νερό και στους οργανικούς ιστούς.
- Τυποποίηση μέσω του ISO 11784/85.
- Σχετικά αδιαπέραστη σε μεταλλικές παρεμβολές.
- Ζώνη συχνοτήτων παγκοσμίως διαθέσιμη.
- Υψηλή επιτρεπόμενη ισχύς εκπομπής.

Μειονεκτήματα

- Ευρύ φάσμα των διαθέσιμων μορφών κατασκευής αναμεταδότη (υψηλή συνέλιξη της σπειροειδής κεραίας).
- Μικρή χωρητικότητα δεδομένων.
- Αργή μεταφορά δεδομένων.

Συχνότητα 13,56 MHz

Πλεονεκτήματα

- Χρήση χαμηλού κόστους παθητικών πομποδεκτών.
- Τυποποίηση μέσω του ISO 15693, Part 1-3.
- Υψηλή χωρητικότητα δεδομένων.
- Η μέση ταχύτητα μεταφοράς δεδομένων 26 kbit/sec.
- Ζώνη συχνοτήτων παγκοσμίως διαθέσιμη.

Μειονεκτήματα

- Υψηλή εξασθένηση μέσα από μεταλλικά περιβάλλοντα.
- Η απόσταση ανάγνωσης περιορίζεται από τους νομικούς κανονισμούς.
- Μεγαλύτερες αποστάσεις ανάγνωσης, μπορούν να επιτευχθούν με μεγαλύτερες κεραίες.

Συχνότητες 896, 915 MHz

Πλεονεκτήματα

- Μεγάλες αποστάσεις ανάγνωσης.
- Απλός σχεδιασμός της κεραίας.
- Αποδοτικό από πλευράς κόστους.
- Τυποποίηση (EPC - Electronic Product Code).

Μειονεκτήματα

- Αδύναμη η διείσδυση στο νερό και τους οργανικούς ιστούς.

Συχνότητα 2,45 GHz

Πλεονεκτήματα

- Υψηλή ταχύτητα μεταφοράς δεδομένων.
- Μεγάλες αποστάσεις ανάγνωσης.

Μειονεκτήματα

- Μεγάλος μέγεθος.
- Υψηλό κόστος.
- Χρήση μπαταριών στους αναμεταδότες.
- Η διάρκεια ζωής περιορίζεται, λόγω των μπαταριών.
- Δεν υπάρχει τυποποίηση.

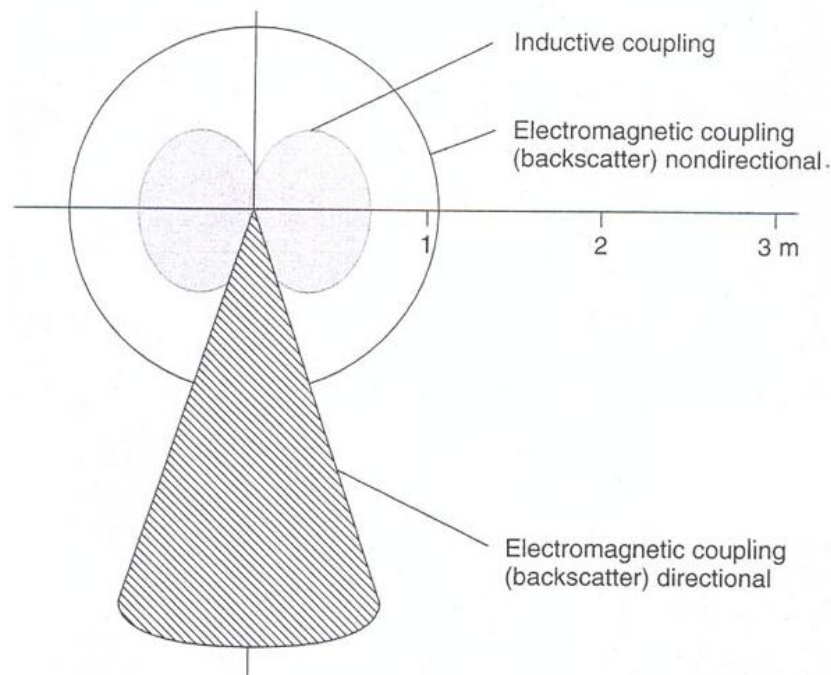
2.7 Περιοχή ανάγνωσης / εγγραφής

2.7.1 Ορισμός

Με τον όρο περιοχή ανάγνωσης ή εγγραφής εννοούμε την περιοχή γύρω από τον αναγνώστη, στην οποία όταν βρεθεί ένας πομποδέκτης, ο αναγνώστης θα μπορέσει να διαβάσει ή να εγγράψει δεδομένα σε αυτόν. Ένα χαρακτηριστικό της περιοχής ανάγνωσης ή εγγραφής είναι η μέγιστη απόσταση ανάγνωσης ή εγγραφής αντίστοιχα ανάμεσα στον πομποδέκτη και τον αναγνώστη. Γενικά, η περιοχή ανάγνωσης ενός συστήματος είναι μεγαλύτερη από την περιοχή εγγραφής, για λόγους που θα γίνουν κατανοητοί στην παρακάτω παράγραφο.

2.7.2 Παράγοντες καθορισμού της περιοχής ανάγνωσης

Η περιοχή ανάγνωσης είναι από τα πιο κρίσιμα λειτουργικά χαρακτηριστικά ενός συστήματος RFID. Εξαρτάται σε μεγάλο βαθμό από τη φέρουσα συχνότητα του αναγνώστη και κατά κανόνα είναι μεγαλύτερη στις υψηλότερες συχνότητες. Επίσης ανάλογα με το είδος της σύζευξης (εικόνα 20) η περιοχή ανάγνωσης είναι λιγότερο κατευθυντική (επαγωγική σύζευξη) ή περισσότερο κατευθυντική (ηλεκτρομαγνητική σύζευξη).



Εικόνα 20: Σύγκριση περιοχών ανάγνωσης RFID ανάλογα με τον τρόπο σύζευξης τους

Επίσης, η περιοχή ανάγνωσης καθορίζεται σε μεγάλο βαθμό από το μέγεθος της ισχύος που μπορεί να μεταδώσει ο αναγνώστης, καθώς και από την ευαισθησία του. Τα παραπάνω εξαρτώνται από τα ηλεκτρονικά κυκλώματα του αναγνώστη, όπως το κύκλωμα ανόρθωσης, το κύκλωμα συντονισμού, ο θόρυβος και από τα χαρακτηριστικά της κεραίας, όπως το

κέρδος, η ενεργός επιφάνεια, ο αριθμός των σπειρών, η πόλωση και η κατευθυντικότητα της. Είναι κατανοητό, ότι όσο πιο κατευθυντική είναι η κεραία του αναγνώστη, τόσο πιο στενή θα είναι και η περιοχή ανάγνωσης ή εγγραφής.

Η περιοχή ανάγνωσης σε ένα σύστημα RFID δεν είναι σταθερή, αλλά εξαρτάται από τον εκάστοτε πομποδέκτη που θα βρεθεί στο πεδίο του αναγνώστη. Συγκεκριμένα, η ελάχιστη απαιτούμενη ισχύς από τον αναγνώστη για την ενεργοποίηση και λειτουργία του ολοκληρωμένου κυκλώματος του πομποδέκτη καθορίζει τη μέγιστη απόσταση ανάγνωσης ή εγγραφής.

Η ελάχιστη απαιτούμενη ισχύς (ευαισθησία) για τη λειτουργία του πομποδέκτη εξαρτάται επίσης από τα ηλεκτρονικά του κυκλώματα, καθώς και από τα χαρακτηριστικά της κεραίας του. Η λειτουργία της εγγραφής απαιτεί μεγαλύτερη κατανάλωση ρεύματος απ'ότι η λειτουργία της ανάγνωσης, οπότε γίνεται κατανοητό για ποιο λόγο η περιοχή εγγραφής ενός συστήματος είναι γενικά μικρότερη από την περιοχή ανάγνωσης του.

Οι απώλειες ισχύος στο ασύρματο κανάλι και στις διάφορες καλωδιώσεις επηρεάζουν επίσης τη μέγιστη απόσταση ανάγνωσης. Επίσης, ο περιβαλλοντικός θόρυβος, οι περιβαλλοντικές συνθήκες (υγρασία και θερμοκρασία), καθώς και η ύπαρξη μετάλλων κι άλλων αντικειμένων στο χώρο επηρεάζουν λιγότερο ή περισσότερο το μέγεθος της περιοχής ανάγνωσης ανάλογα με τη συχνότητα λειτουργίας.

2.7.3 Επιθυμητό μέγεθος & σχήμα περιοχής ανάγνωσης ή εγγραφής

Το επιθυμητό μέγεθος και σχήμα της περιοχής ανάγνωσης ενός συστήματος εξαρτάται από την εκάστοτε εφαρμογή.

Τα κύρια χαρακτηριστικά μιας εφαρμογής που καθορίζουν το μέγεθος και το σχήμα της περιοχής ανάγνωσης ενός συστήματος είναι η ακρίβεια στον καθορισμό της θέσης του πομποδέκτη, η ελάχιστη απόσταση ανάμεσα στους πομποδέκτες, καθώς και η ταχύτητα τους. Σε πολλές εφαρμογές ο πομποδέκτης δεν έχει σταθερή απόσταση από τον αναγνώστη. Στην περίπτωση αυτή η επιθυμητή περιοχή ανάγνωσης του συστήματος θα πρέπει να καλύπτει τη μέγιστη δυνατή απόσταση αναγνώστη-πομποδέκτη. Σε ορισμένα συστήματα, η περιοχή ανάγνωσης πρέπει να σχεδιαστεί έτσι ώστε μόνο ένας πομποδέκτης να βρίσκεται κάθε φορά μέσα σε αυτήν.

Παράδειγμα αποτελούν οι εφαρμογές ασύρματης πληρωμής, όπως με εισιτήρια δημόσιας μεταφοράς (εφαρμόζεται ήδη στη Ν. Κορέα), όπου μόνο ένα εισιτήριο θα πρέπει να μπορεί να διαβαστεί κάθε φορά. Σε αντίθετη περίπτωση, θα διαβάζονταν ταυτόχρονα πάνω από ένα εισιτήρια, με αποτέλεσμα την αδυναμία αντιστοίχισης κάθε εισιτηρίου με τον κάτοχό του. Κατά συνέπεια, στις εφαρμογές ασύρματης πληρωμής, η επιθυμητή περιοχή ανάγνωσης καθορίζεται από την πιθανή απόσταση ανάμεσα σε δύο επιβάτες. Σε αυτή την κατηγορία εφαρμογών, τα μικροκυματικά συστήματα, των οποίων το πεδίο αποτελείται από μία κατευθυντική δέσμη, πλεονεκτούν σε σχέση με τα μη κατευθυντικά πεδία των επαγωγικά συζευγμένων πεδίων. Ωστόσο, υπάρχουν και εφαρμογές που αποσκοπούν στην ταυτόχρονη ανάγνωση όσο γίνεται περισσότερων πομποδεκτών, με επιθυμητή την όσο το δυνατό μικρότερη απόσταση ανάμεσά τους. Στις εφαρμογές αυτές, όσον αφορά το συγκεκριμένο αυτό χαρακτηριστικό, πλεονεκτούν τα συστήματα επαγωγικής σύζευξης.

Η σχετική ταχύτητα ενός πομποδέκτη με αυτή ενός αναγνώστη, σε συνδυασμό με το μέγεθος της περιοχής ανάγνωσης/εγγραφής ενός συστήματος, καθορίζουν το χρονικό διάστημα που βρίσκεται ο πομποδέκτης μέσα στην περιοχή ανάγνωσης του αναγνώστη. Με δεδομένη την ταχύτητα ανάγνωσης ή εγγραφής, το απαιτούμενο εύρος του RFID

συστήματος σχεδιάζεται έτσι ώστε με μέγιστη ταχύτητα του αντικειμένου που φέρει τον πομποδέκτη, το χρονικό διάστημα που βρίσκεται το αντικείμενο μέσα στη ζώνη ανάγνωσης να είναι αρκετό για τη μετάδοση των απαιτούμενων δεδομένων.

2.7.4 Κατηγοριοποίηση συστημάτων RFID ανάλογα με απόσταση ανάγνωσης

Τα RFID συστήματα ανάλογα με το μέγεθος της μέγιστης απόστασης ανάγνωσης πομποδέκτη- αναγνώστη κατηγοριοποιούνται σε συστήματα κοντινής σύζευξης, σε συστήματα απομακρυσμένης σύζευξης και σε συστήματα μεγάλου εύρους.

- *Συστήματα Κοντινής Σύζευξης*

Τα συστήματα κοντινής σύζευξης (close coupling) έχουν σχεδιαστεί για πολύ μικρές αποστάσεις ανάγνωσης ανάμεσα σε 0,1 και 1 cm, οπότε για τη λειτουργία του συστήματος ο πομποδέκτης εισάγεται μέσα στον αναγνώστη ή τοποθετείται πάνω σε μια ειδική επιφάνεια ('touch & go').

Η σύζευξή τους γίνεται μέσω ηλεκτρικών ή μαγνητικών πεδίων, ενώ η συχνότητα λειτουργίας τους θεωρητικά κυμαίνεται σε οποιαδήποτε χρήσιμη συχνότητα μεταξύ DC και 30 MHz και κυρίως μεταξύ 1-10 MHz. Τα συστήματα κοντινής σύζευξης χαρακτηρίζονται από τις μικρότερες απώλειες ισχύος στο ασύρματο κανάλι σε σχέση με τα συστήματα των δύο άλλων κατηγοριών λόγω της πολύ μικρής απόστασης αναγνώστη-πομποδέκτη. Οι πομποδέκτες των συστημάτων αυτών περιέχουν μικροεπεξεργαστές, οι οποίοι απαιτούν ισχύ γύρω στα 10 mW ισχύ.

Τα συστήματα κοντινής σύζευξης χρησιμοποιούνται κυρίως σε εφαρμογές που απαιτούν μεγάλη ασφάλεια, όπως στα συστήματα ελέγχου πρόσβασης.

Σήμερα, η κύρια εφαρμογή πομποδεκτών κοντινής σύζευξης είναι οι έξυπνες κάρτες χωρίς επαφή (contactless smart cards), τα μηχανικά κι ηλεκτρικά χαρακτηριστικά των οποίων περιγράφονται στο πρωτόκολλο ISO 10536.

Πρέπει να τονισθεί ότι ο ρόλος των συστημάτων κοντινής σύζευξης στην αγορά γίνεται ολοένα και λιγότερο σημαντικός.

- *Συστήματα Απομακρυσμένης Σύζευξης*

Τα συστήματα απομακρυσμένης σύζευξης (remote coupling) έχουν μέγιστη απόσταση ανάγνωσης μέχρι 1 m. Οι συχνότητες μετάδοσης που χρησιμοποιούνται είναι αυτές που βρίσκονται κάτω από 135 KHz ή η συχνότητα στα 13.56 MHz. Ορισμένες ειδικές εφαρμογές (όπως η Eurobalise) λειτουργούν στα 27.125 MHz. Το 90% αυτών των συστημάτων λειτουργούν με επαγωγική (μαγνητική) σύζευξη, οπότε λέγονται και επαγωγικά ραδιοσυστήματα (inductive radio systems), ενώ μόνο το 10% λειτουργούν με χωρητική (ηλεκτρική) σύζευξη. Ο Οργανισμός Διεθνών Προδιαγραφών (ISO) έχει δημοσιεύσει μια σειρά προδιαγραφών που καθορίζουν τις τεχνικές παραμέτρους του πομποδέκτη και του αναγνώστη για διάφορες εφαρμογές, όπως τις έξυπνες κάρτες χωρίς επαφή (contactless smart cards), την αναγνώριση ζώων ή και το βιομηχανικό αυτοματισμό.

Επιπλέον, ο ISO διακρίνει τα RFID συστήματα απομακρυσμένης σύζευξης σε proximity coupling συστήματα (ISO 14443) και σε vicinity coupling συστήματα (ISO 15693).

Τα proximity coupling συστήματα έχουν εύρος 4-10 ίντσες περίπου και εφαρμόζονται στις έξυπνες κάρτες χωρίς επαφή. Συχνά χρησιμοποιούνται σε αθλητικά γεγονότα και σε άλλες μεγάλες συγκεντρώσεις που απαιτούν έλεγχο πρόσβασης.

Τα vicinity coupling συστήματα λειτουργούν σε μεγαλύτερες αποστάσεις ανάγνωσης 1 έως 1,5 m και βρίσκουν εφαρμογή στις έξυπνες ετικέτες (smart labels) και στις έξυπνες κάρτες χωρίς επαφή (contactless smart cards). Μια τυπική εφαρμογή είναι οι κάρτες ελεγχόμενης πρόσβασης (κάρτες ασφάλειας ή κάρτες ταυτότητας σε κτίρια γραφείων ή κέντρα δεδομένων). Επίσης, χρησιμοποιούνται στον έλεγχο πρόσβασης σε πολύ μεγάλα δημόσια γεγονότα και στην αναγνώριση των συμμετεχόντων σε αγώνες. Στο μαραθώνιο ένας πομποδέκτης μπορεί να τοποθετηθεί στο παπούτσι των αθλητών, οπότε μπορεί να υπολογιστεί με ακρίβεια ο χρόνος τερματισμού.

- *Συστήματα Μεγάλου Εύρους*

Τα RFID συστήματα με απόσταση ανάγνωσης πολύ μεγαλύτερη του 1 m ονομάζονται μεγάλου εύρους (long-range) συστήματα και λειτουργούν χρησιμοποιώντας ηλεκτρομαγνητικά κύματα στις UHF συχνότητες (μικροκυματικό εύρος). Η μεγάλη πλειοψηφία των συστημάτων αυτών λειτουργούν με ηλεκτρομαγνητική σύζευξη. Επιπλέον, υπάρχουν μεγάλου εύρους συστήματα που χρησιμοποιούν πομποδέκτες ακουστικών επιφανειακών κυμάτων στο μικροκυματικό εύρος. Όλα αυτά τα συστήματα λειτουργούν στις UHF συχνότητες των 868 MHz (στην Ευρώπη) και των 915 MHz (στις Ηνωμένες Πολιτείες), καθώς και στις μικροκυματικές συχνότητες των 2.5 GHz και των 5.8 GHz. Με τη χρήση παθητικών πομποδεκτών μπορούν να επιτευχθούν μέγιστες αποστάσεις ανάγνωσης των 3 μέτρων, ενώ με τη χρήση μπαταρίας μπορούν να επιτευχθούν μέγιστες αποστάσεις ανάγνωσης της τάξεως των 30 μέτρων.

2.7.5 Ταχύτητα ανάγνωσης δεδομένων

Η ταχύτητα ανάγνωσης ενός πομποδέκτη μέσα στην περιοχή ανάγνωσης είναι ένα πολύ σημαντικό λειτουργικό χαρακτηριστικό ενός συστήματος RFID και επηρεάζεται επίσης από πολλούς παράγοντες. Η ταχύτητα ανάγνωσης (ρυθμός αναγνώρισης) εξαρτάται κυρίως από την ταχύτητα μετάδοσης δεδομένων του συστήματος, άρα και από το εύρος ζώνης του συστήματος. Στα επαγωγικά συζευγμένα συστήματα που λειτουργούν στις UHF συχνότητες, οι μεγαλύτεροι ρυθμοί μετάδοσης δεδομένων χαρακτηρίζονται από μικρότερη μέγιστη απόσταση ανάγνωσης.

Ο θόρυβος, τόσο ο ηλεκτρονικός από τα ηλεκτρονικά κυκλώματα του αναγνώστη και του πομποδέκτη, όσο και ο περιβαλλοντικός, οδηγούν σε κακή ποιότητα σήματος κι εσφαλμένες μεταδόσεις, οπότε μειώνουν την ταχύτητα μετάδοσης. Επίσης, η ταχύτητα ανάγνωσης εξαρτάται σε μεγάλο βαθμό από τον αριθμό των πομποδεκτών στο πεδίο του αναγνώστη, το χρησιμοποιούμενο πρωτόκολλο επικοινωνίας και τον αλγόριθμο αποφυγής σύγκρουσης. Ένας επιπλέον παράγοντας που μπορεί να μειώσει το ρυθμό αναγνώρισης ή ανάγνωσης είναι η στιγμιαία απώλεια ισχύος στον πομποδέκτη.

2.8 Συστήματα αποφυγής σύγκρουσης πομποδεκτών (Tag Anticollision)

Σε πολλές εφαρμογές, είναι πιθανό να βρεθούν ταυτόχρονα πολλοί πομποδέκτες στη ζώνη ανάγνωσης ενός αναγνώστη. Στην περίπτωση αυτή, ο αναγνώστης στέλνει σήμα σε όλους να επικοινωνήσουν μαζί του. Προκειμένου να του απαντήσουν, οι πομποδέκτες μεταδίδουν ταυτόχρονα το σειριακό τους αριθμό ή τα υπόλοιπα δεδομένα που τους ζητά ο αναγνώστης. Οι μεταδόσεις παρεμβάλλουν η μία στην άλλη και καμία δε μπορεί να ληφθεί σωστά. Το φαινόμενο αυτό λέγεται σύγκρουση πομποδεκτών κι αντιμετωπίζεται μέσω ειδικών πρωτοκόλλων αποφυγής σύγκρουσης που ενσωματώνονται στους πομποδέκτες RFID και ελέγχονται από τον αναγνώστη του συστήματος.

Τα πρωτόκολλα αποφυγής σύγκρουσης πολλές φορές αναφέρονται κι ως μέθοδοι ή αλγόριθμοι αποφυγής σύγκρουσης και παρουσιάζουν πολλές ομοιότητες με τους αλγόριθμους αποφυγής σύγκρουσης στα δίκτυα. Ωστόσο, λόγω ορισμένων αδυναμιών των πομποδεκτών, οι αλγόριθμοι αποφυγής σύγκρουσης στα RFID συστήματα υστερούν σε σχέση τους αντίστοιχους αλγόριθμους στα δίκτυα. Ένα βασικό πρόβλημα είναι η περιορισμένη διαθέσιμη ισχύς στους πομποδέκτες. Επίσης, οι πομποδέκτες δε μπορούν να λάβουν πληροφορίες κατάστασης, όπως το ακριβές ποσοστό των ήδη αναγνωσμένων πομποδεκτών. Επιπλέον, η ανίχνευση συγκρούσεων στα συστήματα RFID είναι δύσκολη, εξαιτίας της διαφορετικής έντασης των σημάτων απόκρισης κάθε πομποδέκτη. Τέλος, όπως και στα περισσότερα ασύρματα δίκτυα, οι πομποδέκτες δεν μπορούν να «ακούσουν» ο ένας τον άλλο, με την έννοια ότι δε μπορούν να αντιληφθούν τότε μεταδίδει ένας κοντινός τους πομποδέκτης.

Οι αλγόριθμοι αποφυγής σύγκρουσης πομποδεκτών διακρίνονται σε πιθανοτικούς και σε ντετερμινιστικούς, ανάλογα με τον τρόπο απόκρισης των πομποδεκτών (κατά τον αλγόριθμο).

Υπάρχουν πολλές παραλλαγές των πιθανοτικών αλγορίθμων ανάλογα με τη δυνατότητα ελέγχου των πομποδεκτών από τον αναγνώστη, ενώ πολλοί βασίζονται στον αλγόριθμο Aloha γνωστό από τα δίκτυα. Στο συγκεκριμένο αλγόριθμο, οι χρόνοι απόκρισης των πομποδεκτών είναι είτε συνεχείς, είτε σε σχισμές. Για παράδειγμα, το πρωτόκολλο ISO 15693 υποστηρίζει το πρωτόκολλο Aloha με σχισμές.

Στους ντετερμινιστικούς αλγόριθμους, ο αναγνώστης έχει τη δυνατότητα να ενεργοποιήσει τους πομποδέκτες που ο ίδιος επιθυμεί με βάση το σειριακό αριθμό αναγνώρισής τους. Ο απλούστερος ντετερμινιστικός αλγόριθμος είναι ο αλγόριθμος αναζήτησης δυαδικού δέντρου, στον οποίο ο αναγνώστης διασχίζει ένα δέντρο με όλους τους πιθανούς σειριακούς αριθμούς αναγνώρισης της εφαρμογής, με στόχο να αναγνωρίσει τους πομποδέκτες που βρίσκονται στο πεδίο του. Κατά τη διάσχιση του δέντρου, ο αναγνώστης επισκεπτόμενος έναν κόμβο, ενεργοποιεί, δηλαδή εντολή ανάγνωσης στους πομποδέκτες των οποίων ο σειριακός αριθμός είναι παιδί του συγκεκριμένου κόμβου και στη συνέχεια περιμένει αποκρίσεις. Σε περίπτωση απόκρισης, επισκέπτεται στη συνέχεια τα παιδιά του κόμβου, ενώ σε περίπτωση που δε λάβει καμιά απόκριση, μεταβαίνει στα αδέρφια του κόμβου.

Η απόδοση των αλγορίθμων εξαρτάται από τις παρακάτω παραμέτρους:

- την ταχύτητα ανάγνωσης των πομποδεκτών
- το εύρος ζώνης του εκπεμπόμενου από τον αναγνώστη σήματος
- το εύρος ζώνης του εκπεμπόμενου σήματος
- το μέγεθος της πληροφορίας κατάστασης που μπορεί να αποθηκευτεί αξιόπιστα στους πομποδέκτες
- την ανοχή του αλγορίθμου σε διαφορετικά είδη θορύβου στο πεδίο
- το κόστος του πομποδέκτη
- το κόστος του αναγνώστη
- το εύρος ανάγνωσης του συστήματος.

Το επιτρεπόμενο εύρος ζώνης του εκπεμπόμενου σήματος από τον αναγνώστη παίζει καθοριστικό ρόλο στην επιλογή του αλγορίθμου αποφυγής σύγκρουσης σε ένα σύστημα. Ισχύει ότι σε κάθε φέρουσα συχνότητα των συστημάτων RFID, το επιτρεπόμενο εύρος του εκπεμπόμενου σήματος είναι διαφορετικό.

Για παράδειγμα, στα 13.56 MHz το εύρος ζώνης είναι πολύ μικρότερο από αυτό στα 915 MHz. Το αποτέλεσμα είναι ότι στα συστήματα με συχνότητα λειτουργίας των 13.56 MHz χρησιμοποιούνται αλγόριθμοι αποφυγής σύγκρουσης βασισμένοι στο Aloha, σε αντίθεση με τους ντετερμινιστικούς αλγόριθμους που χρησιμοποιούνται στα συστήματα των 915 MHz. Ωστόσο, οι πιο πολλοί αλγόριθμοι πρακτικά συνδυάζουν στοιχεία και από τις δύο κατηγορίες αλγορίθμων.

2.9 Συστήματα αποφυγής σύγκρουσης αναγνωστών (Reader Anticollision)

Σε εφαρμογές με μεγάλη πυκνότητα αναγνωστών στο χώρο, όπως στη διαχείριση της εφοδιαστικής αλυσίδας, μπορούν να προκύψουν προβλήματα παρεμβολής των αναγνωστών μεταξύ τους.

Το πρόβλημα της σύγκρουσης των αναγνωστών επισημάνθηκε για πρώτη φορά από το Auto-id center και λύνεται μέσω δυναμικής παραχώρησης συχνοτήτων σε έναν αριθμό συχνοτήτων στο χώρο. Η παραπάνω διαδικασία γίνεται είτε με κεντρικό είτε με καταναμημένο έλεγχο. Τα προβλήματα σύγκρουσης των αναγνωστών παρουσιάζουν ομοιότητες με τα προβλήματα ανάθεσης συχνοτήτων στα συστήματα κινητής τηλεφωνίας. Ωστόσο, οι λύσεις που εφαρμόζονται στα συστήματα αυτά, δε μπορούν να εφαρμοσθούν αντίστοιχα στα συστήματα RFID, λόγω της περιορισμένης λειτουργικότητας των πομποδεκτών. Πιο συγκεκριμένα, οι πομποδέκτες δε μπορούν να αναγνωρίσουν με ποιον αναγνώστη επικοινωνούν κάθε φορά. Για παράδειγμα, όταν επικοινωνούν δύο αναγνώστες ταυτόχρονα με έναν πομποδέκτη, αυτός δε μπορεί να αντιληφθεί από ποιον δέχεται εντολές κάθε χρονική στιγμή.

Κατά συνέπεια, ένας πομποδέκτης θα πρέπει να επικοινωνεί με ένα μόνο αναγνώστη κάθε φορά, ιδίως στην περίπτωση εκτέλεσης εντολών που πρόκειται να αλλάξουν την κατάσταση του πομποδέκτη. Συστήνεται λοιπόν η επικοινωνία αναγνωστών και πομποδεκτών να είναι σύντομη κι ατομική.

2.10 Ασφάλεια στα συστήματα RFID

Στις σύγχρονες ανοιχτές εφαρμογές των συστημάτων RFID, τίθενται ορισμένα ζητήματα ασφάλειας. Συγκεκριμένα, τα συστήματα RFID θα πρέπει να προστατεύονται από προσπάθειες ανάγνωσης ή τροποποίησης των δεδομένων του πομποδέκτη (rfid tag) από μη εξουσιοδοτημένους αναγνώστες.

Προκειμένου να λυθούν τα ζητήματα ασφάλειας, οι πομποδέκτες και οι αναγνώστες θα πρέπει να «εμπιστεύονται» ο ένας τον άλλο με την έννοια ότι θα πρέπει να γνωρίζουν ο ένας την ταυτότητα του άλλου, πριν την έναρξη της ανταλλαγής δεδομένων μεταξύ τους. Το παραπάνω επιτυγχάνεται μέσω μιας διαδικασίας αμοιβαίας αυθεντικοποίησης της ταυτότητας αναγνώστη & πομποδέκτη μέσω γνωστών πρωτοκόλλων. Συνήθως χρησιμοποιείται ένα μυστικό κρυπτολογικό κλειδί που θα πρέπει να γνωρίζουν και οι δύο (αμοιβαία συμμετρική αυθεντικοποίηση).

Επίσης, η ασύρματη μεταφορά δεδομένων θα πρέπει να προστατευθεί από προσπάθειες υποκλοπής. Υπάρχει περίπτωση ένας μη εξουσιοδοτημένος χρήστης που θα θελήσει να διαβάσει ή να εγγράψει δεδομένα σε έναν αναγνώστη, να προσπαθήσει να υποκλέψει το μυστικό κρυπτολογικό κλειδί και μετά να το χρησιμοποιήσει, προκειμένου να πετύχει το σκοπό του. Οι υποκλοπές αντιμετωπίζονται μέσω ποικίλων διαδικασιών κρυπτογράφησης τόσο των δεδομένων, όσο και του μυστικού κλειδιού. Γνωστή είναι η κρυπτογράφηση δημόσιου κλειδιού. Οι μέθοδοι κρυπτογράφησης κι αυθεντικοποίησης στα συστήματα RFID προέρχονται από τα δίκτυα υπολογιστών, τα οποία θα πρέπει επίσης να προστατεύονται από ενδεχόμενες επιθέσεις.

Το πρόβλημα της ενσωμάτωσης των παραπάνω μεθόδων στα συστήματα RFID είναι η αύξηση του κόστους των πομποδεκτών.

2.11 Πρότυπα και πρωτόκολλα

Μέχρι σήμερα δεν έχουν δημιουργηθεί παγκοσμίως αποδεκτά πρότυπα για όλα τα συστήματα RFID, ενώ υπάρχουν δύο οργανισμοί που έχουν αναλάβει την ευθύνη της ανάπτυξης προτύπων, ο **ISO** και ο **EPCglobal**.

Σε μια προσπάθεια δημιουργίας διεθνώς αποδεκτών προτύπων έχει συσταθεί μια επιτροπή από αντιπροσώπους των δύο αυτών οργανισμών που προσπαθούν να δημιουργήσουν συμβατά πρότυπα για τα ανοιχτά συστήματα της νέας γενιάς.

2.11.1 Πρότυπο ISO

Ο ISO έχει αναπτύξει ένα μεγάλο αριθμό προτύπων για διάφορα συστήματα RFID που προορίζονται για διαφορετικές εφαρμογές. Τα πρότυπα περιγράφουν τα φυσικά χαρακτηριστικά και τις διαστάσεις του εκάστοτε συστήματος, τα πρωτόκολλα επικοινωνίας του, τη συχνότητα λειτουργίας, την παροχή ισχύος, τις μεθόδους αποφυγής σύγκρουσης που τυχόν χρησιμοποιεί, τον τρόπο διαμόρφωσης και κωδικοποίησης των δεδομένων κατά τη μεταφορά τους από τον αναγνώστη στον πομποδέκτη και το αντίθετο, κ.α.

Ανάμεσα στα πρότυπα που έχει αναπτύξει ο ISO για τα συστήματα RFID είναι τα ISO 11784, 11785 και 14223 για τις εφαρμογές αναγνώρισης ζώων, το ISO 10536 για τα συστήματα κοντινής σύζευξης, το ISO 14443 για τα proximity coupling συστήματα, ISO

15693 για τα vicinity coupling συστήματα (έξυπνες ετικέτες) και το ISO18000 για εφαρμογές αναγνώρισης και διαχείρισης αντικειμένων.

2.11.2 Το σύστημα EPC

Το σύστημα EPC αναπτύχθηκε από το Auto-id Center στο Πανεπιστήμιο του MIT με τη χορηγία πολλών βιομηχανιών και κατασκευαστών συστημάτων RFID. Η ευθύνη για τη διαχείριση του συστήματος EPC μετατέθηκε το 2003 στον οργανισμό EPCglobal, Inc, ο οποίος ανήκει στο UCC και στο EAN International.

Το πιο σημαντικό χαρακτηριστικό του συστήματος EPC είναι ότι προσφέρει τη δυνατότητα σύνδεσης αντικειμένων με το Internet μέσω του Electronic Product Code. Το Electronic Product Code αποθηκεύεται στον πομποδέκτη και ταυτοποιεί μοναδικά το υπό αναγνώριση αντικείμενο. Το EPC παρέχει τη δυνατότητα αναγνώρισης διαφορετικών αντικειμένων του ίδιου τύπου. Ωστόσο, οποιαδήποτε άλλη πληροφορία αφορά το αντικείμενο, αποθηκεύεται στο δίκτυο κι όχι στον πομποδέκτη, με το EPC να αποτελεί ένα είδος δείκτη (pointer) σε αυτήν την πληροφορία. Το σύστημα που προκύπτει, ελαχιστοποιεί τη λειτουργικότητα του πομποδέκτη, «μεταφέροντας» την στο δίκτυο.

Τα παραπάνω επιτυγχάνονται μέσω της υπηρεσίας ONS, η οποία λειτουργεί ως υπηρεσία ηλεκτρονικού καταλόγου, αντιστοιχώντας κάθε κωδικό EPC με μια διεύθυνση IP, όπου μπορεί να γραφεί ή να διαβαστεί πληροφορία για το σχετικό αντικείμενο. Η πληροφορία καταχωρείται σε γλώσσα PML (Physical Markup Language).

Η λειτουργία του συστήματος EPC βασίζεται και στους λεγόμενους Savant Servers.

Το Savant είναι λογισμικό υπεύθυνο για τη διαχείριση κι επεξεργασία του μεγάλου όγκου δεδομένων που παράγεται από τα συστήματα RFID. Επιτρέπει τη δημιουργία ενός καταναμημένου δικτύου, με τον κάθε Savant Server να λειτουργεί ως μια πύλη για το επόμενο υψηλότερο επίπεδο στην ιεραρχία Savant, απομονώνοντας αποτελεσματικά το υποδίκτυο του αναγνώστη. Τόσο το ONS, όσο και το Savant δεν είναι απαραίτητο να ενσωματωθούν σε μια εφαρμογή που χρησιμοποιεί την τεχνολογία EPC. Ωστόσο, η ενσωμάτωσή τους σε ένα σύστημα EPC μειώνει τις ενεργειακές απαιτήσεις του πομποδέκτη κι επιτρέπει τη μείωση της μνήμης του, άρα και του κόστους του. Επιπλέον, επιτρέπει τη μείωση του εύρους ζώνης που απαιτείται για την ασύρματη μεταφορά των δεδομένων και κάνει το σύστημα πιο ανθεκτικό, προσφέροντας τη δυνατότητα απομόνωσης των προβληματικών τμημάτων του συστήματος.

Υπάρχουν πολλές προδιαγραφές EPC που διαφοροποιούνται ανάλογα με την τάξη (class) και την έκδοση (version) τους. Ακολουθεί μια σύντομη ανάλυση των προδιαγραφών EPC:

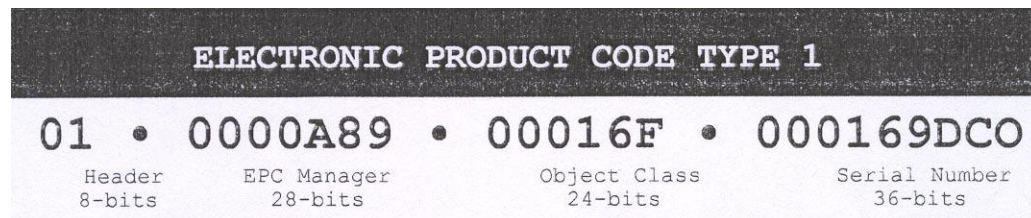
- Οι πομποδέκτες **τάξης 0** χρησιμοποιούνται μόνο για ανάγνωση με μνήμη των 64 ή 96 bits και λειτουργούν στη UHF ζώνη 868-930 MHz. Υπάρχουν και οι πομποδέκτες τάξης 0+ που προσφέρουν τη δυνατότητα επανεγγραψιμότητας.
- Οι **UHF πομποδέκτες τάξης 1** είναι τύπου WORM, λειτουργούν στη ζώνη 868-930 MHz κι έχουν μνήμη των 96 bits.
- Οι προδιαγραφές **EPC τάξης 2** συστημάτων UHF αναπτύχθηκαν αρχικά, για να ξεπεραστούν προβλήματα παρεμβολών και μη διαθεσιμότητας συχνοτήτων σε παγκόσμιο επίπεδο που χαρακτήριζαν το αρχικό πρωτόκολλο. Οι πομποδέκτες δεύτερης γενιάς έχουν μνήμη 96-256 bits. Το Υπουργείο Αμύνης των ΗΠΑ (DoD) και η Wal-Mart προωθούν τη χρήση της τεχνολογίας EPC δεύτερης γενιάς. Η ζώνη

λειτουργίας για το UHF σύστημα δεύτερης γενιάς είναι 860-960 MHz και είναι συμβατή με τα πρότυπα ISO 18000-6.

Όπως έχει αναφερθεί, τα συστήματα RFID ανά τον κόσμο λειτουργούν σε διαφορετικές συχνότητες μέσα στη UHF ζώνη (π.χ. 902-928 MHz στις ΗΠΑ, 865.6-875.1 MHz στην Ευρώπη (ETSI), 950-956MHz στην Ιαπωνία κτλ.) ανάλογα με τους τοπικούς (εθνικούς) κανονισμούς μετάδοσης. Οι UHF πομποδέκτες δεύτερης γενιάς αποκρίνονται σε όλα τα σωστά κωδικοποιημένα σήματα μέσα στη ζώνη των 860-960 MHz, με αποτέλεσμα να δημιουργούνται συστήματα τα οποία να είναι πλέον διεθνώς συμβατά.

- Οι **HF πομποδέκτες τάξης 1** έχουν μνήμη των 96 bit, μπορούν να επανεγγραφούν μόνο για μια φορά (WORM) και λειτουργούν στα 13,56 MHz.
- Υπάρχουν πομποδέκτες τάξης 3, 4 και 5 που χρησιμοποιούνται σε συστήματα RFID μεγαλύτερης λειτουργικότητας (π.χ. με ενσωματωμένους αισθητήρες).

Ο κωδικός EPC αποτελείται από μία επικεφαλίδα και τρία επιπλέον τμήματα δεδομένων. Στην επικεφαλίδα αποθηκεύεται η τάξη (class) του EPC, στο δεύτερο τμήμα ο κατασκευαστής του υπό αναγνώριση αντικειμένου και στο τρίτο τμήμα ο ακριβής τύπος του προϊόντος. Το τέταρτο τμήμα δεδομένων είναι ο μοναδικός σειριακός αριθμός του αντικειμένου. Ακολουθεί ένα παράδειγμα κωδικού EPC των 96 bits σε δεκαεξαδική μορφή.



Εικόνα 21: Παράδειγμα κωδικού EPC των 96 bits.

Ο παραπάνω κωδικός έχει πολύ μεγάλες δυνατότητες αποθήκευσης πληροφοριών. Συγκεκριμένα, μπορεί να κωδικοποιήσει 68 δισεκατομμύρια σειριακούς αριθμούς για πάνω από 16 εκατομμύρια τύπους προϊόντων για 268 εκατομμύρια διαφορετικές επιχειρήσεις.

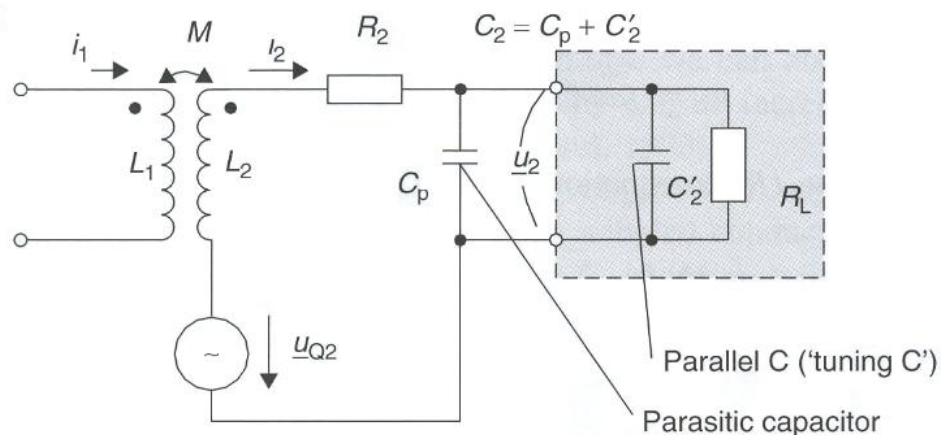
2.12 Κατηγορίες σύζευξης

Η μεγάλη πλειοψηφία των συστημάτων RFID με συχνότητα λειτουργίας κάτω των 30 MHz λειτουργούν σύμφωνα με την αρχή της **επαγωγικής σύζευξης**. Αντίθετα, στα συστήματα RFID με συχνότητα λειτουργίας άνω των 100 MHz, ο πομποδέκτης και ο αναγνώστης επικοινωνούν μέσω **ηλεκτρομαγνητικών κυμάτων**, ενώ στις εφαρμογές κοντινής σύζευξης χρησιμοποιούνται συστήματα **χωρητικής σύζευξης**.

2.12.1 Επαγωγική σύζευξη

Σε ένα επαγωγικά συζευγμένο σύστημα RFID οι κεραίες τόσο του αναγνώστη, όσο και του πομποδέκτη έχουν τη μορφή κατάλληλα διαμορφωμένων σπειρών, δηλαδή είναι πηνία. Οι επαγωγικά συζευγμένοι πομποδέκτες είναι παθητικοί όπου η απαιτούμενη ενέργεια για τη λειτουργία τους παρέχεται από τον αναγνώστη. Τα μήκη κύματος που αντιστοιχούν στις συχνότητες λειτουργίας των επαγωγικών συστημάτων είναι πολλές φορές μεγαλύτερα από την απόσταση ανάμεσα στην κεραία του αναγνώστη και τον πομποδέκτη, άρα έχουμε σύζευξη στο κοντινό πεδίο και το ηλεκτρομαγνητικό πεδίο που δημιουργείται μπορεί να θεωρηθεί ως ένα απλό μαγνητικό εναλλασσόμενο πεδίο.

Η επαγωγική σύζευξη των δύο κεραίων (πηνίων) θυμίζει τη διάταξη και τη σύζευξη που λαμβάνει χώρα σε ένα μετασχηματιστή, αν και στα συστήματα RFID η σύζευξη ανάμεσα στα δύο ελίγματα είναι αρκετά πιο ασθενής. Συγκεκριμένα, τα επαγωγικά συζευγμένα κυκλώματα λειτουργούν με συντελεστές σύζευξης k της τάξης του 0,01. Όταν το πηνίο του πομποδέκτη βρεθεί στο μαγνητικό πεδίο του πηνίου του αναγνώστη, τότε διαπερνάται από μαγνητική ροή. Η επαγόμενη ισχύς μπορεί να ενεργοποιήσει το κύκλωμα του πομποδέκτη, εφόσον η τιμή της διατηρείται πάνω από κάποιο ελάχιστο όριο. Η ισχύς που επάγεται στον πομποδέκτη εξαρτάται από πολλούς παράγοντες. Συγκεκριμένα, είναι ανάλογη της συχνότητας λειτουργίας f , του αριθμού των σπειρών N , της ενεργού επιφάνειας A του πηνίου του tag, ενώ εξαρτάται και από τη σχετική γωνία και την απόσταση ανάμεσα στα δύο πηνία. Επίσης, η παρουσία μετάλλων κοντά στην περιοχή των πηνίων επηρεάζει σε μεγάλο βαθμό αρνητικά το συντελεστή σύζευξης. Στην **εικόνα 22** φαίνεται το ισοδύναμο κύκλωμα για επαγωγικά συζευγμένα συστήματα.

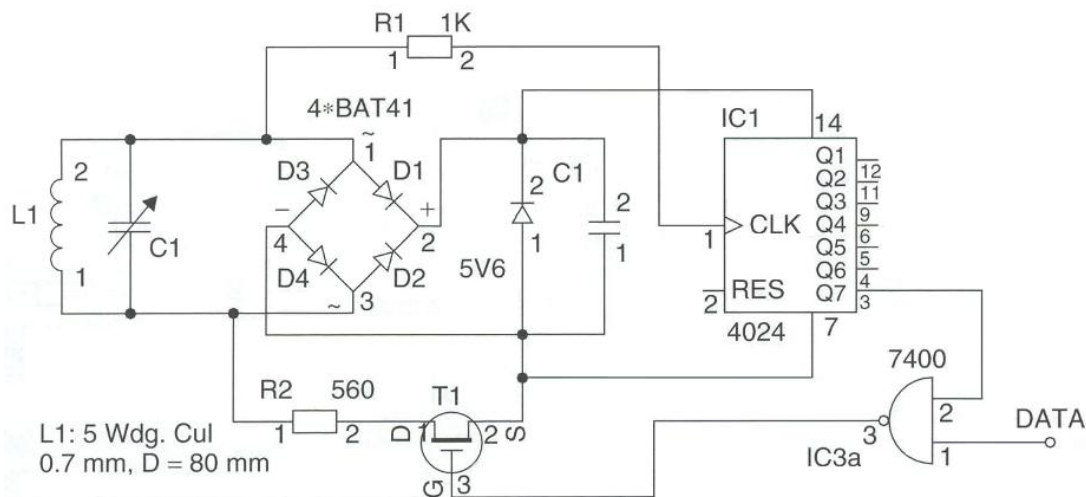


Εικόνα 22: Διάγραμμα ισοδύναμου κυκλώματος αναγνώστη και πομποδέκτη σε μαγνητικά συζευγμένα κυκλώματα.

Το πηνίο L_1 αντιπροσωπεύει την κεραία μετάδοσης του αναγνώστη, το L_2 την κεραία του πομποδέκτη, το R_2 την αντίσταση πηνίου της κεραίας του πομποδέκτη, ενώ το R_L είναι η αντίσταση εισόδου του ολοκληρωμένου κυκλώματος του πομποδέκτη. Ο πυκνωτής C_2 δημιουργείται από την παράλληλη σύνδεση ενός πυκνωτή C'_2 και μιας παρασιτικής χωρητικότητας C_p που υπάρχει στο κύκλωμα του πομποδέκτη. Στους πομποδέκτες που λειτουργούν σε συχνότητες κάτω των 135 KHz, το πηνίο L_2 συνδέεται συνήθως παράλληλα με έναν πυκνωτή μεγέθους $C'_2=20\sim 220$ pF, προκειμένου να επιτευχθεί η επιθυμητή συχνότητα συντονισμού.

Αντίθετα, σε συχνότητες όπως τις 13.56 MHz και 27.125 MHz η απαιτούμενη χωρητικότητα είναι συνήθως πολύ μικρή και παρέχεται συνήθως από την εσωτερική χωρητικότητα του ίδιου του πομποδέκτη μαζί με την παρασιτική χωρητικότητα του πηνίου. Το πηνίο L_2 μαζί με τον πυκνωτή C_2 δημιουργούν ένα **παράλληλο κύκλωμα συντονισμού** με συχνότητα ίση με τη συχνότητα λειτουργίας του συστήματος. Στην πράξη, δεν είναι πάντα εφικτό να ταυτιστεί απόλυτα η συχνότητα του πομποδέκτη με τη συχνότητα μετάδοσης του αναγνώστη.

Ο **αποσυντονισμός** αυτός μπορεί να οφείλεται σε περιβαλλοντικούς λόγους, όπως στην υψηλή θερμοκρασία και υγρασία, στην παρουσία μετάλλων, καθώς και σε ανοχές κατά την κατασκευή του πομποδέκτη. Ορισμένοι κατασκευαστές ημιαγωγών ενσωματώνουν επιπρόσθετους πυκνωτές εξομάλυνσης στο πηνίο του πομποδέκτη για την ελαχιστοποίηση του φαινομένου αυτού. Σε μερικά συστήματα, η τιμή της C_2 είναι μεταβαλλόμενη για λόγους μεταφοράς δεδομένων, όποτε προκαλείται αποσυντονισμός. Επίσης, σε ορισμένα συστήματα αποφυγής σύγκρουσης, η συχνότητα λειτουργίας του πομποδέκτη είναι σκόπιμα μεγαλύτερη από τη συχνότητα του αναγνώστη (σκόπιμος αποσυντονισμός). Η τάση u_2 στα άκρα της αντίστασης R_L είναι η τάση που χρησιμοποιείται για την τροφοδοσία του ολοκληρωμένου κυκλώματος των παθητικών πομποδεκτών. Αρχικά, η τάση ανορθώνεται μέσω μιας γέφυρας χαμηλών απωλειών και στη συνέχεια εξομαλύνεται. Ένα απλό τέτοιο κύκλωμα φαίνεται στην **εικόνα 23**. Στους ενεργητικούς πομποδέκτες η τάση u_2 χρησιμεύει απλά ως σήμα για την ενεργοποίησή του ολοκληρωμένου κυκλώματος.



Εικόνα 23: Κύκλωμα ανορθωτή γέφυρας

Από την ανάλυση του κυκλώματος της εικόνας 22 προκύπτει:

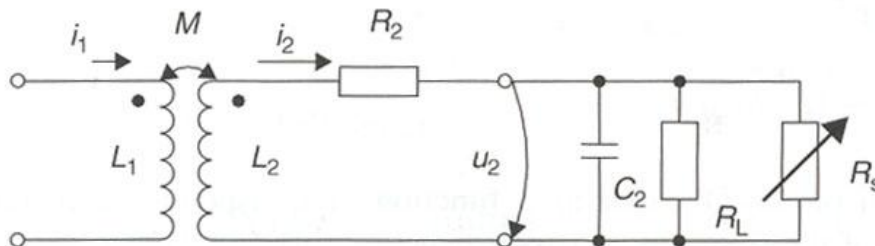
$$u_2 = \frac{j \cdot \omega \cdot k \cdot \sqrt{L_1 L_2} \cdot i_1}{1 + (j\omega L_2 + R_2) \cdot \left(\frac{1}{R_L} + j\omega C_2\right)}$$

Για τον παράγοντα ποιότητας Q του συντονισμένου κυκλώματος ισχύει:

$$Q = \frac{1}{R_2 \sqrt{\frac{C_2}{L_2} + \frac{1}{R_L} \sqrt{\frac{L_2}{C_2}}}} = \frac{1}{\frac{R_2}{\omega L_2} + \frac{\omega L_2}{R_L}}$$

Παρατηρούμε ότι όταν $R_2 \rightarrow \infty$ και $R_L \rightarrow 0$, τότε η τάση u_2 τείνει στο μηδέν. Αντίθετα, όταν $R_2 \rightarrow 0$ και $R_L \rightarrow \infty$ (άρα έχουμε ελάχιστη κατανάλωση ισχύος από το chip), η επαγόμενη τάση u_2 παίρνει υψηλές τιμές. Για κάθε ζεύγος παραμέτρων (R_2 , R_L), υπάρχει μια τιμή της L_2 για την οποία μεγιστοποιείται ο παράγοντας Q και κατά συνέπεια η τάση u_2 . Η ιδιότητα αυτή είναι πολύ χρήσιμη και πρέπει να λαμβάνεται υπόψιν κατά την κατασκευή του πομποδέκτη, προκειμένου να βελτιστοποιείται η απόσταση ανάγνωσης των επαγωγικά συζευγμένων RFID συστημάτων.

Η λειτουργία του ολοκληρωμένου κυκλώματος απαιτεί μια σταθερή τάση λειτουργίας της τάξης των 3-5 V μετά την ανόρθωση. Ωστόσο, στο ισοδύναμο κύκλωμα της εικόνας 22 υπάρχει η πιθανότητα πολύ μεγάλης αύξησης της u_2 μέχρι και πάνω από 100 V, σε περίπτωση αύξησης του συντελεστή σύζευξης ή μείωσης της αντίστασης φορτίου. Προκειμένου να ρυθμίζεται η τάση ανεξάρτητα από τους παραπάνω παράγοντες και να παραμένει σταθερή, προστίθεται παράλληλα με την R_L μια ρυθμιστική αντίσταση παράκαμψης R_s , η τιμή της οποίας εξαρτάται από την τάση. Το αντίστοιχο ισοδύναμο κύκλωμα φαίνεται στην εικόνα 24.



Εικόνα 24: Ρύθμιση της τάσης σε έναν πομποδέκτη με τη χρήση ενός ρυθμιστή παράκαμψης

Η R_S πρέπει να μπορεί να παίρνει ένα αρκετά μεγάλο εύρος τιμών (αρκετές δυνάμεις του δέκα) προκειμένου η επαγόμενη τάση να είναι σταθερή κι ανεξάρτητη του συντελεστή σύζευξης. Για την υλοποίηση της R_S συνήθως χρησιμοποιούνται διατάξεις διόδων zener. Θα ήταν χρήσιμο να υπολογιστεί η ελάχιστη ένταση μαγνητικού πεδίου H_{\min} (στη θέση του πομποδέκτη) που είναι ικανή να επάγει στον πομποδέκτη την ελάχιστη απαιτούμενη για τη λειτουργία του τάση. Η τελευταία διαφοροποιείται ανάλογα με το είδος του πομποδέκτη (παθητικός ή ενεργητικός) και τη λειτουργία της ανάγνωσης ή της εγγραφής. Η τάση u_{Q2} (εικόνα 22) που επάγεται στο πηνίο του πομποδέκτη μπορεί να υπολογιστεί από την παρακάτω σχέση:

$$u_{Q2} = \mu_o \cdot A \cdot N \cdot \omega \cdot H$$

όπου H είναι η ένταση του ημιτονοειδούς μαγνητικού πεδίου, ω είναι η γωνιακή συχνότητα, N ο αριθμός των σπειρών του πηνίου του πομποδέκτη L_2 και A η ενεργός επιφάνειά του.

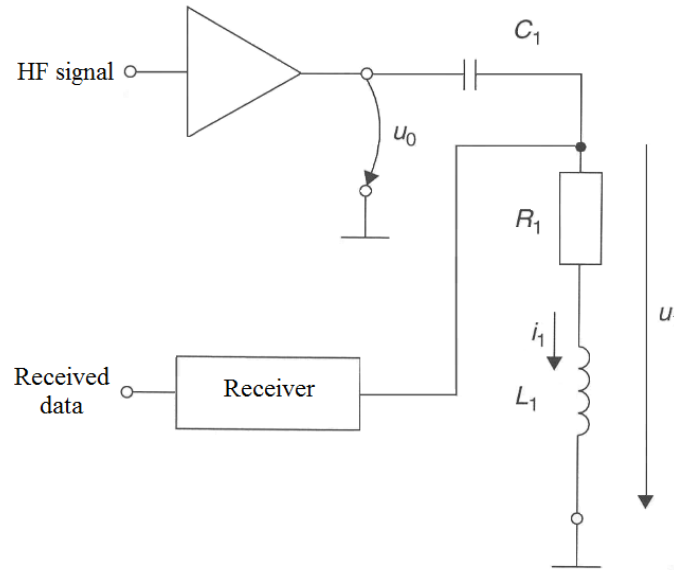
Χρησιμοποιώντας τις παραπάνω εξισώσεις βρίσκουμε την ακόλουθη έκφραση για το H .

$$H = \frac{u_2 \cdot \sqrt{\left(\frac{\omega L_2}{R_L} + \omega R_2 C_2\right)^2 + \left(1 - \omega^2 L_2 C_2 + \frac{R_2}{R_L}\right)^2}}{\omega \cdot \mu_o \cdot A \cdot N}$$

Παρατηρούμε ότι η ένταση του μαγνητικού πεδίου H εξαρτάται από τη συχνότητα ω , την ενεργό επιφάνεια της κεραίας A , τον αριθμό των σπειρών N του πηνίου, την τάση u_2 και την αντίσταση εισόδου R_2 . Είναι φανερό ότι στην περίπτωση συντονισμού η H παίρνει τη χαμηλότερη τιμή της. Μια απόκλιση της συχνότητας συντονισμού του πομποδέκτη από τη συχνότητα μετάδοσης του αναγνώστη, οδηγεί σε μεγαλύτερη απαιτούμενη ένταση μαγνητικού πεδίου κι επομένως σε μικρότερη απόσταση ανάγνωσης.

Η ελάχιστη απαιτούμενη ένταση μαγνητικού πεδίου σχετίζεται με την ευαισθησία του πομποδέκτη και καθορίζει την πιο μακρινή απόσταση που μπορεί να τοποθετηθεί ο πομποδέκτης από τον αναγνώστη. Η απόσταση αυτή λέγεται **μέγιστη απόσταση ενεργοποίησης** και δε συμπίπτει πάντοτε με τη μέγιστη απόσταση ανάγνωσης του συστήματος, η οποία μπορεί να είναι μικρότερη. Κάτι τέτοιο εξαρτάται από το αν μπορούν να ανιχνευθούν και να διαβαστούν τα δεδομένα που στέλνονται από τον πομποδέκτη στον αναγνώστη στη μέγιστη απόσταση ενεργοποίησης.

Για τη μελέτη του κυκλώματος του αναγνώστη, χρησιμοποιούμε το ισοδύναμο κύκλωμα της εικόνας 25. Το πηνίο L_1 είναι το στοιχείο που δημιουργεί το μεταβαλλόμενο μαγνητικό πεδίο. Η αντίσταση R_1 αντιπροσωπεύει τις ωμικές απώλειες του πηνίου L_1 . Ο πυκνωτής C_1 χρησιμοποιείται για τη δημιουργία ενός εν σειρά κυκλώματος συντονισμού συχνότητας συντονισμού $f_{\text{συντ}}$ ίση με τη συχνότητα μετάδοσης f_{Tx} , που μεγιστοποιεί το ρεύμα που διαρρέει το πηνίο L_1 . Το μέγιστο ρεύμα ισούται με u_0/R_1 , επειδή η εμπέδηση του κυκλώματος για $f=f_{\text{συντ}}$ ισούται με R_1 , εφόσον οι εμπέδησεις του L_1 και του C_1 είναι ίσες κι αντίθετες και αλληλοαναιρούνται.



Εικόνα 25: Ισοδύναμο κύκλωμα αναγνώστη στα επαγωγικά συζευγμένα συστήματα

Παρά το γεγονός ότι η τάση πηγής u_0 είναι λίγα μόνο volts, οι ξεχωριστές τιμές των τάσεων στο L_1 και C_1 μπορούν να φθάσουν στα επίπεδα των εκατοντάδων volts. Για το λόγο αυτό, τα στοιχεία του κυκλώματος θα πρέπει να αντέχουν σε υψηλές τάσεις.

Η ενεργοποίηση του κυκλώματος του πομποδέκτη έχει σαν αποτέλεσμα την προς τα πίσω επαγωγή ενός ρεύματος στην κεραία του αναγνώστη. Μπορούμε λοιπόν να θεωρήσουμε ότι σε σειρά με την R_1 και την L_1 παρεμβάλλεται μια εμπέδηση Z'_T που αντιπροσωπεύει τη σύζευξη από τον πομποδέκτη προς τον αναγνώστη.

Η σχετική ανάλυση δίνει την σχέση:

$$Z'_T = \frac{\omega^2 \kappa^2 L_1 L_2}{R_2 + j\omega L_2 + \frac{R_L}{1 + j\omega R_L C_2}}$$

Όπως θα δούμε παρακάτω, η Z'_T χρησιμοποιείται για τη μοντελοποίηση της μεταφοράς των δεδομένων από τον πομποδέκτη προς τον αναγνώστη.

Η μεταφορά δεδομένων από τον πομποδέκτη στον αναγνώστη στα επαγωγικά συζευγμένα συστήματα γίνεται μέσω των ακόλουθων διαδικασιών:

- **Διαμόρφωση Φορτίου (Load Modulation)**

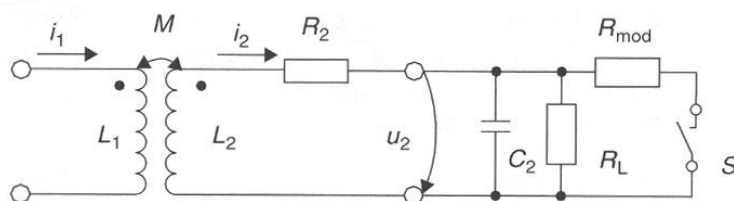
Η διαμόρφωση φορτίου είναι μία από τις πιο συνηθισμένες διαδικασίες μεταφοράς δεδομένων από τον πομποδέκτη στον αναγνώστη.

Γίνεται μέσω της μεταβολής κάποιων κυκλωματικών παραμέτρων σε χρονισμό με τη ροή των δεδομένων, οπότε μεταβάλλεται η εμπέδηση Z'_T που ορίσαμε στην προηγούμενη παράγραφο και διαμορφώνεται κατά πλάτος ή κατά φάση η επαγόμενη τάση στα άκρα του πηνίου της κεραίας του αναγνώστη. Η ανάκτηση των δεδομένων γίνεται μέσω μιας διαδικασίας αποδιαμόρφωσης στον αναγνώστη, αφού προηγηθεί μια διαδικασία ενίσχυσης της επαγόμενης τάσης.

Στη διαμόρφωση φορτίου, η συχνότητα λειτουργίας του πομποδέκτη συμπίπτει με τη συχνότητα μετάδοσης του αναγνώστη.

Στα επαγωγικά συζευγμένα συστήματα, οι κυκλωματικές παράμετροι που μπορούν να μεταβληθούν είναι η αντίσταση φορτίου R_L (ωμική διαμόρφωση φορτίου) και η παράλληλη χωρητικότητα C_2 (χωρητική διαμόρφωση φορτίου).

Στην **ωμική διαμόρφωση φορτίου** (εικόνα 26) προσθαφαιρείται μια παράλληλη αντίσταση φορτίου R_{mod} στο ολοκληρωμένο μνήμης του πομποδέκτη σε χρονισμό με τη ροή των δεδομένων. Η παράλληλη σύνδεση της R_{mod} μειώνει την ολική συνολική αντίσταση, οπότε και τον παράγοντα ποιότητας Q , καθώς και την Z'_T . Η τελευταία μεταβάλλεται μόνο ως προς το πλάτος, ενώ η φάση της παραμένει σταθερή. Συνεπώς, η τάση u_1 στα άκρα του πηνίου (εικόνα 26) διαμορφώνεται κατά πλάτος.



Εικόνα 26: Κύκλωμα ωμικής διαμόρφωσης φορτίου

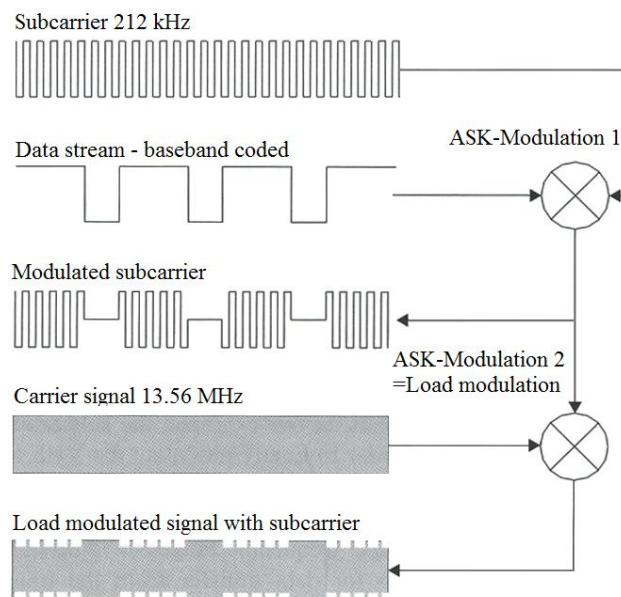
Στη **χωρητική διαμόρφωση φορτίου**, προσθαφαιρείται μια πρόσθετη χωρητικότητα φορτίου C_{mod} σε χρονισμό με τη ροή των δεδομένων. Το αποτέλεσμα είναι η μεταβολή της συχνότητας συντονισμού του πομποδέκτη ανάμεσα σε δύο τιμές, οπότε έχουμε μεταβολή του πλάτους και της φάσης της Z'_T .

- **Διαμόρφωση Φορτίου με Υποφέρον (Load Modulation with Subcarrier)**

Εξαιτίας της ασθενούς σύζευξης ανάμεσα στις κεραίες του πομποδέκτη και του αναγνώστη, το σήμα του πομποδέκτη προς τον δέκτη του αναγνώστη είναι εξαιρετικά αδύναμο σε σχέση με το σήμα που μεταδίδει ο ίδιος ο αναγνώστης προς

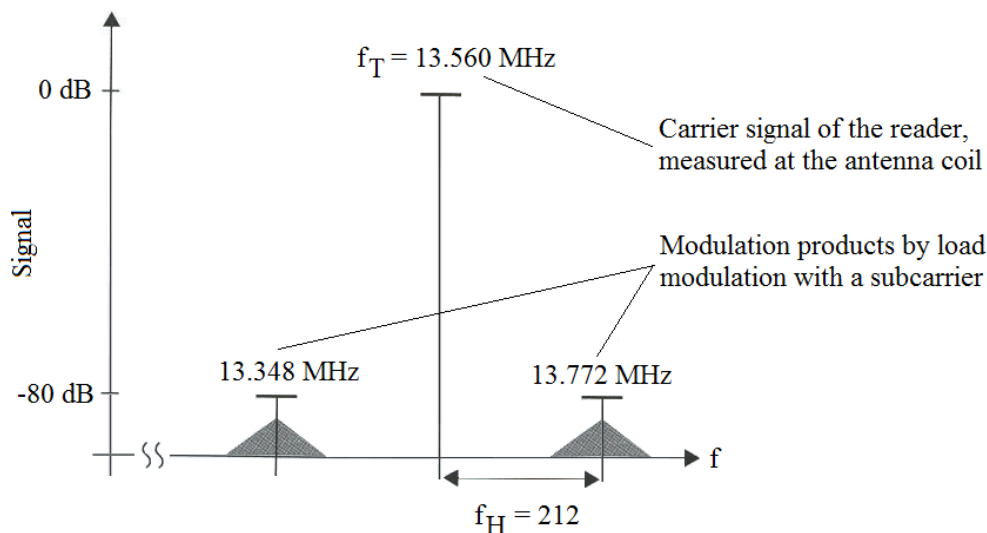
τον πομποδέκτη ή τους πομποδέκτες. Πρακτικά, σ' ένα σύστημα των 13,56 MHz η τάση εξόδου της κεραίας του αναγνώστη είναι περίπου στα 100V (υψηλή τάση λόγω συντονισμού), ενώ το χρήσιμο σήμα που αναμένεται από τον αναγνώστη είναι της τάξης των 10 mV (-80 dB). Το ασθενές αυτό σήμα μπορεί να ανιχνευθεί μόνο μέσω πολύπλοκων κυκλωμάτων που αποδιαμορφώνουν τις πλευρικές ζώνες που παράγονται κατά τη διαμόρφωση πλάτους.

Μια καλύτερη μέθοδος είναι η χρήση υποφέροντος που απαιτεί πολυεπίπεδη διαμόρφωση. Η απλή διαμόρφωση φορτίου γίνεται με τη προσθαφαίρεση μιας αντίστασης φορτίου σε χρονισμό με ένα κωδικοποιημένο σήμα βασικής ζώνης. Αντίθετα, στη διαμόρφωση φορτίου με υποφέρον το κωδικοποιημένο σήμα βασικής ζώνης διαμορφώνει πρώτα ένα υποφέρον χαμηλής συχνότητας χρησιμοποιώντας ASK, FSK ή PSK διαμόρφωση. Η συχνότητα του υποφέροντος προκύπτει συνήθως από τη δυαδική διαίρεση της συχνότητας λειτουργίας. Για τα RFID συστήματα των 13,56MHz συνήθως χρησιμοποιούνται οι ακόλουθες συχνότητες υποφέροντος: 847kHz (=13,56MHz÷16), 424kHz(=13,56MHz÷32) και 212kHz (=13,56MHz÷64). Στη συνέχεια, το διαμορφωμένο σήμα του υποφέροντος χρησιμοποιείται για το χρονισμό της προσθαφαίρεσης της αντίστασης φορτίου και τη μεταφορά των δεδομένων (εικόνα 27).



Εικόνα 27: Διαδικασία διαμόρφωσης φορτίου με διαμορφωμένο κατά πλάτος (ASK)

Το μεγάλο πλεονέκτημα της χρήσης διαμόρφωσης υποφέροντος είναι το φάσμα συχνοτήτων που προκύπτει. Η διαμόρφωση φορτίου με υποφέρον δημιουργεί δύο φασματικές γραμμές σε μια απόσταση $\pm f_H$, ίση με τη συχνότητα μετάδοσης του υποφέροντος γύρω από τη συχνότητα μετάδοσης του αναγνώστη f_T . Οι γραμμές αυτές εύκολα ανιχνεύονται με τη προϋπόθεση ότι η f_H είναι μικρότερη της f_T . Η πληροφορία τώρα μεταδίδεται στις δύο πλευρικές ζώνες των δύο φασματικών γραμμών, ανάλογα με το είδος της διαμόρφωσης. Στην εικόνα 28 εκτός από τα παραπάνω, φαίνεται και η διαφορά ανάμεσα στο σήμα του φέροντος του αναγνώστη συχνότητας f_T και τις λαμβανόμενες πλευρικές ζώνες διαμόρφωσης.



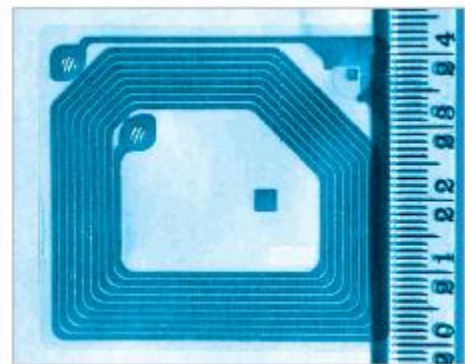
Εικόνα 28: Τα παράγωγα της διαμόρφωσης φορτίου με υποφέρον

Οι πλευρικές ζώνες διαμόρφωσης κατά τη διαμόρφωση φορτίου με υποφέρον μπορούν να διαχωριστούν από το πολύ πιο ισχυρό σήμα του αναγνώστη με ζωνοπερατό φίλτράρισμα σε μία από τις δύο συχνότητες $f_T \pm f_H$. Δεν έχει σημασία ποια από τις δύο συχνότητες θα χρησιμοποιηθεί, γιατί η πληροφορία περιέχεται και στις δύο πλευρικές ζώνες. Μετά την εξομάλυνσή του, το σήμα του υποφέροντος αποδιαμορφώνεται εύκολα. Εξαιτίας του μεγάλου απαιτούμενου εύρους ζώνης για τη μετάδοση του υποφέροντος, αυτή η διαδικασία μπορεί να γίνει μόνο στα εύρη συχνοτήτων ISM, όπως 6.78 MHz, 13.56 MHz και στα 27.125 MHz. Η διαμόρφωση φορτίου με υποφέρον διακρίνεται σε ωμική και χωρητική σε αντιστοιχία με την απλή διαμόρφωση φορτίου.

Ένα τυπικό είδος κεραίας αναγνώστη στα επαγωγικά συζευγμένα συστήματα είναι η κεραία-βρόχος μικρών διαστάσεων σχετικά με τα αντίστοιχα μήκη κύματος.

Η κεραία των πομποδεκτών (π.χ. στα inlays *εικόνα 29*) είναι ένα επίπεδο πηνίο, ο αριθμός των σπειρών του οποίου εξαρτάται από τη συχνότητα λειτουργίας του συστήματος. Γενικά, όσο αυξάνει η συχνότητα f , μειώνεται η απαιτούμενη αυτεπαγωγή του πηνίου, οπότε μειώνεται και ο αριθμός των σπειρών (στα 135 KHz απαιτούνται 100-1000 σπείρες, ενώ στα 13.56 MHz 3-10 σπείρες). Αυτό οφείλεται στο γεγονός ότι η επαγόμενη τάση στον πομποδέκτη είναι ανάλογη της συχνότητας f .

Οι φυσικές διαστάσεις του πηνίου επηρεάζουν την ενεργό του επιφάνεια, άρα και την ευαισθησία του συστήματος. Σε δύσκολες συζεύξεις (π.χ. πολλά μεταλλικά αντικείμενα στο χώρο) μπορούν να χρησιμοποιηθούν πηνία με πυρήνα από σιδηρομαγνητικό υλικό για αύξηση της επαγωγής.



Εικόνα 29: Ένα inlay

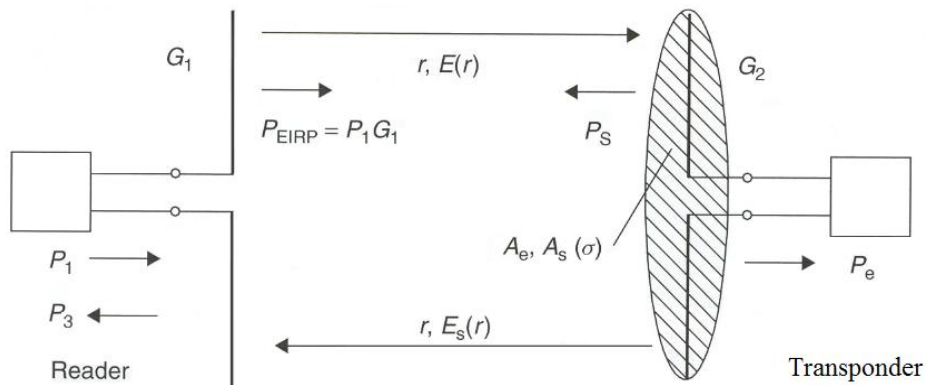
2.12.2 Ηλεκτρομαγνητική σύζευξη σκέδασης

Η λειτουργία των συστημάτων RFID που λειτουργούν στις UHF συχνότητες 868 MHz (Ευρώπη) και 915 MHz (ΗΠΑ) και στις μικροκυματικές συχνότητες 2.5 GHz και 5.8 GHz βασίζεται στη σκέδαση/ανάκλαση των ηλεκτρομαγνητικών κυμάτων. Τα μήκη κύματος που αντιστοιχούν στις συχνότητες λειτουργίας αυτών των συστημάτων είναι σχετικά μικρά, οπότε οι πομποδέκτες θεωρούνται πάντοτε ότι βρίσκονται στο μακρινό πεδίο της κεραίας του αναγνώστη.

Ένας μικροκυματικός πομποδέκτης, βρισκόμενος στο μακρινό πεδίο του αναγνώστη, θα δεχθεί ισχύ από τον τελευταίο με τη μορφή ηλεκτρομαγνητικών κυμάτων, την οποία ανάλογα με το αν είναι παθητικός ή ενεργητικός χρησιμοποιεί για την τροφοδότηση του ολοκληρωμένου του και τη μετάδοση δεδομένων στον αναγνώστη ή απλά για την ενεργοποίηση του κυκλώματός του μέσω ενός κυκλώματος ανίχνευσης. Χρησιμοποιώντας τη σύγχρονη τεχνολογία ημιαγωγών, τα παραγόμενα τσιπς των παθητικών πομποδεκτών μπορούν να λειτουργήσουν με κατανάλωση ισχύος όχι μεγαλύτερης των 5 μW . Η απαιτούμενη εισερχόμενη ισχύς για έναν ενεργητικό πομποδέκτη είναι σαφώς πολύ μικρότερη, με αποτέλεσμα μεγαλύτερες αποστάσεις ανάγνωσης.

Στα συστήματα RFID ηλεκτρομαγνητικής σύζευξης, η μετάδοση δεδομένων από τον πομποδέκτη στον αναγνώστη, όπως υποδεικνύει και το όνομά τους, βασίζεται στην ανάκλαση/σκέδαση ηλεκτρομαγνητικών κυμάτων. Οι ανακλαστικές ιδιότητες των αντικειμένων και στη συγκεκριμένη περίπτωση των κεραιών αυξάνουν γενικά με την αύξηση της συχνότητας, οπότε προκύπτει ένας επιπλέον λόγος που η ηλεκτρομαγνητική ανακλαστική σύζευξη εφαρμόζεται στα συστήματα υψηλών συχνοτήτων. Επίσης, προκειμένου οι πομποδέκτες να λειτουργούν ως καλοί σκεδαστές, θα πρέπει οι κεραίες τους να είναι μικρών διαστάσεων, συγκρίσιμων με το μικρό μήκος κύματος.

Η αρχή λειτουργίας των συστημάτων ανακλαστικής σύζευξης βασίζεται στην τεχνολογία των ραντάρ. Ένα απλοποιημένο μοντέλο ενός συστήματος ανακλαστικής σύζευξης φαίνεται στην *εικόνα 30*. Ο αναγνώστης μεταδίδει προς την κατεύθυνση του πομποδέκτη ένα ηλεκτρομαγνητικό κύμα με $\text{EIRP} = P_1 \cdot G_1$, όπου G_1 το κέρδος της κεραίας του προς την κατεύθυνση του πομποδέκτη. Μέρος της ισχύος που φθάνει στον πομποδέκτη απορροφάται από τον ίδιο τον πομποδέκτη και το υπόλοιπο ανακλάται προς τα πίσω. Τελικά, στην κεραία του αναγνώστη φθάνει η ισχύς P_3 .

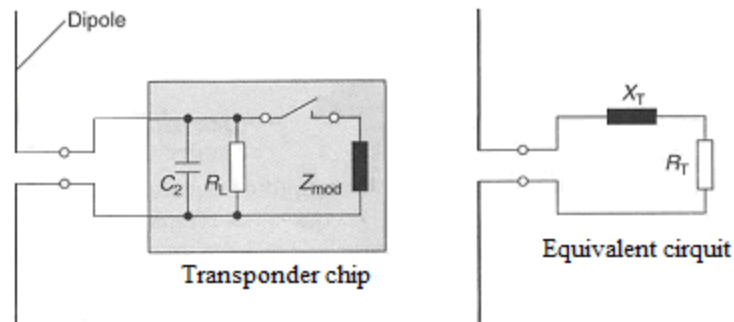


Εικόνα 30: Ένα απλοποιημένο μοντέλο ενός συστήματος ανακλαστικής σύζευξης

Γνωρίζουμε ότι η μέγιστη ισχύς P_e που μπορεί να λάβει η κεραία του πομποδέκτη, με δεδομένη τη σωστή πόλωσή της σε σχέση την πόλωση του αναγνώστη, είναι ανάλογη της πυκνότητας ισχύος S του επίπεδου κύματος από τον αναγνώστη στη θέση του πομποδέκτη και της ενεργούς της επιφάνειας A_e . Ισχύει:

$$P_e = A_e \cdot S$$

Αποτέλεσμα είναι η επαγωγή μιας τάσης στην κεραία του πομποδέκτη, το ισοδύναμο κύκλωμα του οποίου φαίνεται στην *εικόνα 31*, όπου $Z_T = R_T + X_T$ είναι η εμπέδηση εισόδου του ολοκληρωμένου κυκλώματος του πομποδέκτη.



Εικόνα 31: Ισοδύναμα κυκλώματα ενός πομποδέκτη με κεραία-δίπολο στην πιο απλοποιημένη (δεξιά) και στη πιο σύνθετη μορφή (αριστερά).

Θεωρώντας ότι έχουμε προσαρμογή μεταξύ της κεραίας του πομποδέκτη και του ολοκληρωμένου κυκλώματός του, ισχύει:

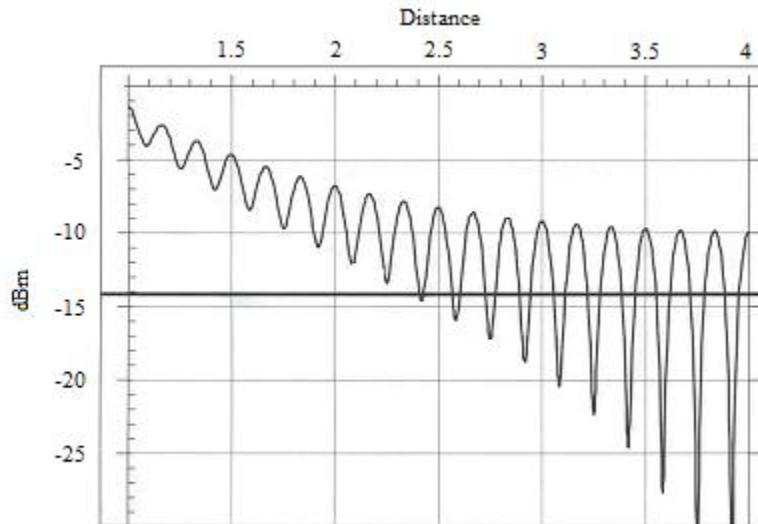
$$P_e = \frac{\lambda_o^2}{4\pi} \cdot G_2 \cdot S = u_o \cdot i_o$$

όπου u_o και i_o η επαγόμενη τάση και ρεύμα αντίστοιχα στα άκρα της κεραίας. Το ολοκληρωμένο κύκλωμα του πομποδέκτη ενεργοποιείται από την ισχύ P_e , μέσω ενσωματωμένων ανορθωτικών κυκλωμάτων που χρησιμοποιούν διόδους Schottky με ιδιαίτερος χαμηλό κατώφλι τάσης.

Η απαιτούμενη ισχύς στα άκρα της κεραίας του πομποδέκτη για τη λειτουργία του τσιπ είναι $P_e = 50 \mu W$. Αν η ισχύς εκπομπής του αναγνώστη είναι $0.5 W$ EIRP, τότε η απώλεια ελεύθερου χώρου δε θα πρέπει να υπερβαίνει τα $50 dB$, προκειμένου να ληφθεί αρκετή ισχύς στον πομποδέκτη για τη λειτουργία του τσιπ.

Η διάδοση των ηλεκτρομαγνητικών κυμάτων επηρεάζεται ως γνωστόν και από διάφορα αντικείμενα που βρίσκονται στο χώρο (**πολυδιαδρομική διάδοση**). Το φαινόμενο αυτό επηρεάζει πολύ την ένταση του πεδίου E γύρω από τον αναγνώστη, δημιουργώντας ζώνες ενίσχυσης (υπέρθυση κυμάτων σύμφωνης φάσης) ή εξασθένησης (υπέρθυση κυμάτων αντίθετης φάσης) του πεδίου και παρουσιάζεται συχνά σε βιομηχανικά περιβάλλοντα, όπου υπάρχουν μεγάλα μεταλλικά αντικείμενα, όπως μηχανές, μεταλλικοί σωλήνες κι άλλα.

Στην *εικόνα 32* απεικονίζεται η ένταση πεδίου E σε συνάρτηση με την απόσταση από τον αναγνώστη σε περιοχή πολλών ανακλάσεων.



Εικόνα 32: Διάγραμμα της έντασης πεδίου E σε συνάρτηση με την απόσταση από την κεραία του αναγνώστη σε περιβάλλον πολλών ανακλάσεων

Η ισχύς P_s (εικόνα 30) που ανακλάται (σκεδάζεται) από τον πομποδέκτη προς τον αναγνώστη είναι ανάλογη της **ενεργούς επιφάνειας σκέδασης A_s της κεραίας** του πομποδέκτη. Οι κεραίες των πομποδεκτών στα συστήματα RFID ανακλαστικής σύζευξης (backscatter) έχουν μικρές διαστάσεις συγκρίσιμες με το μήκος κύματος, ώστε να έχουν ικανοποιητικές δυνατότητες σκέδασης προς όλες τις κατευθύνσεις.

Η τιμή της A_s εξαρτάται από την τιμή της Z_T . Συγκεκριμένα, παίρνει τη μέγιστη τιμή της $A_s=4A_e$ για $Z_T=0$ και την ελάχιστη τιμή της $A_s=0$ για $Z_T \rightarrow \infty$. Συνεπώς, η ενεργός επιφάνεια σκέδασης μπορεί να πάρει οποιαδήποτε τιμή, ανάμεσα στις δύο ακραίες τιμές, ανάλογα με την τιμή της Z_T , ιδιότητα που χρησιμοποιείται στη μετάδοση δεδομένων, όπως θα δούμε σε επόμενη παράγραφο.

Στην περίπτωση μιας προσαρμοσμένης κεραίας η A_s ισούται με την ενεργό διατομή λήψης A_e της κεραίας ($\sigma = A_s = A_e$). Αυτό σημαίνει ότι η μισή ισχύς που λαμβάνεται από το ηλεκτρομαγνητικό πεδίο παρέχεται στην αντίσταση τερματισμού R_T και η υπόλοιπη μισή ανακλάται στο χώρο από την κεραία.

Από τη θεωρία των ραντάρ, έχουμε την εξής σχέση για την P_3 :

$$P_3 = \frac{P_1 \cdot G_1^2 \cdot \lambda_o^4 \cdot G_2^2}{(4\pi r)^4}$$

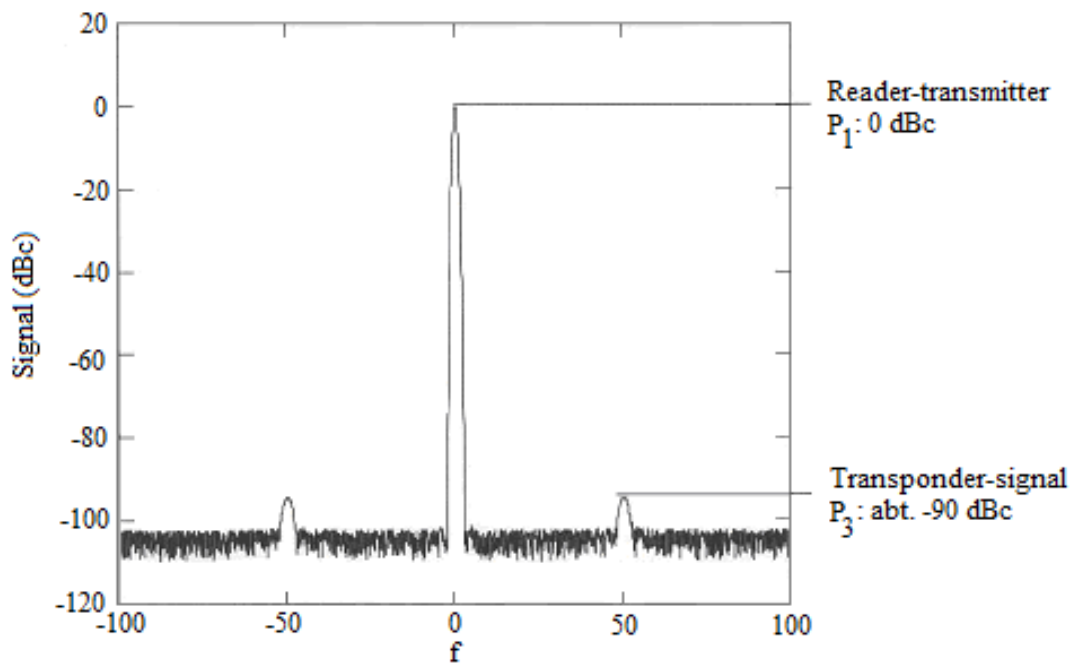
Γνωρίζοντας την **ευαισθησία του αναγνώστη**, την ελάχιστη δηλαδή απαιτούμενη ισχύ P_3 (P_{3min}) που θα πρέπει να λάβει ο αναγνώστης, προκειμένου να μπορεί να διαβάσει τα δεδομένα που του στέλνει ο πομποδέκτης, είναι δυνατό να υπολογιστεί η μέγιστη απόσταση ανάγνωσης του εκάστοτε συστήματος. Άλλωστε, για τη σωστή λειτουργία ενός RFID συστήματος, για να γίνει ανάγνωση δηλαδή ή εγγραφή δεδομένων δεν αρκεί απλά η τροφοδότηση του πομποδέκτη με αρκετή ισχύ για την ενεργοποίησή του, αλλά και το σήμα P_3 που θα φθάσει στον αναγνώστη θα πρέπει να είναι αρκετά ισχυρό, ώστε να διαβαστούν σωστά τα δεδομένα.

Λύνοντας την προηγούμενη σχέση ως προς r προκύπτει:

$$r_{\max} = \frac{\lambda_o}{4\pi} \cdot 4 \sqrt{\frac{P_1 \cdot G_1^2 \cdot G_2^2}{P_{3\min}}}$$

Η ευαισθησία του αναγνώστη καθορίζεται από διάφορους παράγοντες, όπως το επίπεδο του θορύβου στην κεραία και στο πρώτο στάδιο της εισόδου του δέκτη του αναγνώστη. Επίσης, επηρεάζεται δυσμενώς κι από τα υπόλοιπα σήματα που φθάνουν στο δέκτη του αναγνώστη από άλλες πηγές (π.χ. ανακλώμενα σήματα από αντικείμενα στο χώρο του αναγνώστη). Επιπλέον, πολύ θόρυβο παράγει και ο ενσωματωμένος στους αναγνώστες πομπός που μεταδίδει συνεχώς ενέργεια για την ενεργοποίηση του πομποδέκτη, ως αποτέλεσμα κυρίως του θορύβου φάσης του ταλαντωτή που εμπεριέχει. Το αποτέλεσμα είναι η δραστηκή μείωση της ευαισθησίας του δέκτη του αναγνώστη.

Για το λόγο αυτό, έχει καθιερωθεί ένας άτυπος κανόνας που υπαγορεύει ότι το σήμα του πομποδέκτη δε θα πρέπει να είναι μικρότερο από 100dB σε σχέση με το σήμα του φέροντος του πομπού του αναγνώστη (εικόνα 33).



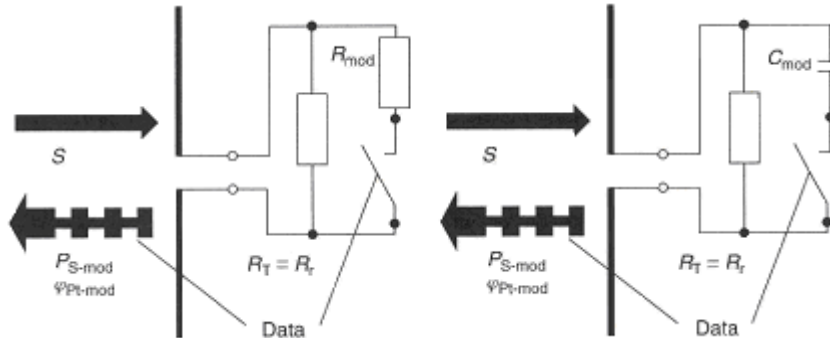
Εικόνα 33: Το φέρον σήμα εκπομπής του αναγνώστη και οι πλευρικές ζώνες διαμόρφωσης που λαμβάνονται από το δέκτη του αναγνώστη

Θα πρέπει να ληφθεί υπόψιν ότι για την αποδιαμόρφωση παίζει ρόλο μόνο το τμήμα της P_3 που αντιστοιχεί στη μία πλευρική ζώνη διαμόρφωσης. Ενδεικτικά, αναφέρουμε ότι η χρήσιμη ισχύς στην περίπτωση ASK διαμόρφωσης είναι το 25% της P_3 .

Γενικά, τα συστήματα RFID ηλεκτρομαγνητικής σκέδασης έχουν μεγαλύτερη μέγιστη απόσταση ανάγνωσης κατά μία τάξη μεγέθους συγκρινόμενα με τα συστήματα επαγωγικής σύζευξης. Στα συστήματα ανακλαστικής σύζευξης, η μετάδοση δεδομένων από τον πομποδέκτη στον αναγνώστη γίνεται μέσω της μεθόδου της διαμορφωμένης ανάκλασης

(modulated backscatter). Ο πομποδέκτης θα πρέπει να μεταδώσει στον αναγνώστη τα δεδομένα που του ζητά ο τελευταίος, μέσω της ισχύος P_3 .

Η μέθοδος αυτή στηρίζεται στην εξάρτηση της επιφανειακής σκέδασης $\sigma = AS$ από την εμπέδηση Z_T . Συγκεκριμένα, η Z_T του πομποδέκτη μεταβάλλεται ανάμεσα σε δύο τιμές σε χρονοισμό με τη ροή των δεδομένων που πρέπει να μεταδοθούν προσθαφαιρώντας ένα επιπρόσθετο φορτίο Z_{mod} . Το φορτίο Z_{mod} μπορεί να είναι είτε μία αντίσταση R_{mod} , οπότε πρόκειται για ωμική διαμόρφωση είτε ένας πυκνωτής C_{mod} , οπότε πρόκειται για χωρητική διαμόρφωση (εικόνα 34).

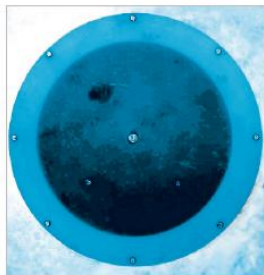


Εικόνα 34: Κυκλώματα ωμικής και χωρητικής διαμόρφωσης σκέδασης

Η διαμόρφωση της εμπέδησης Z_T οδηγεί όχι μόνο στη διαμόρφωση πλάτους, αλλά και στη διαμόρφωση φάσης της επιφανειακής σκέδασης, οπότε και της ανακλώμενης ισχύος P_S . Η σχετική μεταβολή του πλάτους και της φάσης της ενεργούς διατομής σ δίνεται από την παρακάτω σχέση.

$$\Delta\sigma = \frac{\lambda_o^2 \cdot G_2 \cdot \Delta Z_{mod}}{4 \cdot \pi \cdot R_r}$$

Ο συνδυασμός διαμόρφωσης πλάτους και φάσης πρέπει να λαμβάνεται υπόψιν κατά την κατασκευή των μικροκυματικών αναγνώστων. Η πιο ευρέως χρησιμοποιούμενη κεραία αναγνώστη σε αυτά τα συστήματα είναι η κατευθυντική επίπεδη κεραία (patch antenna).

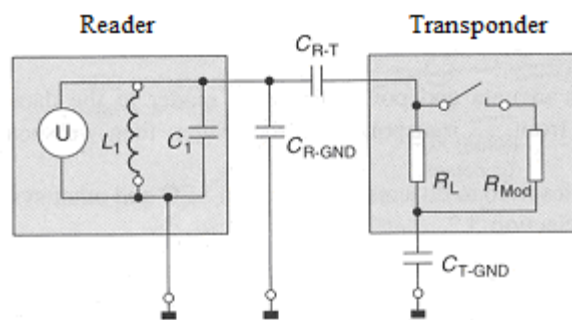


Οι UHF και οι μικροκυματικοί πομποδέκτες μπορούν να χρησιμοποιήσουν πολλά είδη κεραιών, τόσο αυτές που χρησιμοποιούνται στα επαγωγικά όσο και στα χωρητικά συζευγμένα συστήματα, αρκεί οι διαστάσεις τους να είναι περίπου ίσες με το μισό μήκος κύματος της συχνότητας λειτουργίας (περίπου 16 cm στα 915 MHz).

Εικόνα 35: Μία επίπεδη κεραία (patch antenna)

2.12.3 Χωρητική σύζευξη

Η σύζευξη στα ηλεκτρικά (ή αλλιώς χωρητικά) συζευγμένα συστήματα RFID γίνεται μέσω ενός ηλεκτρικού πεδίου που αναπτύσσεται στο κοντινό πεδίο του αναγνώστη. Πρόκειται για συστήματα με συχνότητα λειτουργίας κάτω των 30 MHz που βρίσκουν εφαρμογές κυρίως στα συστήματα κοντινής σύζευξης. Η κεραία του αναγνώστη είναι μια μεγάλη, ηλεκτρικά αγώγιμη επιφάνεια (ηλεκτρόδιο). Όταν μια τάση υψηλής συχνότητας εφαρμοστεί στο ηλεκτρόδιο, τότε ένα υψίσυχο ηλεκτρικό πεδίο δημιουργείται ανάμεσα στο ηλεκτρόδιο και τη γη (γείωση). Ο αναγνώστης διαθέτει ένα συντονισμένο κύκλωμα που αποτελείται από ένα πηνίο συνδεδεμένο παράλληλα με έναν εσωτερικό πυκνωτή και μια ενεργή χωρητικότητα ανάμεσα στο ηλεκτρόδιο και τη γη (εικόνα 36). Η συχνότητα συντονισμού του συντονισμένου κυκλώματος συμπίπτει με τη συχνότητα μετάδοσης του αναγνώστη.



Εικόνα 36: Διάγραμμα ισοδύναμου χωρητικά συζευγμένου κυκλώματος συστήματος RFID

Η κεραία του πομποδέκτη αποτελείται από δύο ηλεκτρόδια πάνω σε ένα επίπεδο. Όταν ο πομποδέκτης τοποθετηθεί μέσα στο ηλεκτρικό πεδίο ενός αναγνώστη, τότε αναπτύσσεται ηλεκτρική τάση ανάμεσα στα δύο ηλεκτρόδια του πομποδέκτη, η οποία χρησιμοποιείται για την παροχή ισχύος στο τσιπ μνήμης του πομποδέκτη. Η αντίσταση R_L αντιπροσωπεύει την αντίσταση εισόδου του πομποδέκτη, η C_{R-T} η χωρητικότητα ανάμεσα στην κεραία του αναγνώστη και την κεραία του πομποδέκτη και η C_{T-GND} η χωρητικότητα που δημιουργείται, όταν ακουμπήσει κανείς τα ηλεκτρόδια του πομποδέκτη (εικόνα 36).

Τα ρεύματα που ρέουν στην επιφάνεια των ηλεκτροδίων του πομποδέκτη είναι πολύ μικρά, οπότε δεν επιβάλλονται περιορισμοί στην αγωγιμότητα του υλικού των ηλεκτροδίων. Πέρα από τις συνηθισμένες μεταλλικές επιφάνειες, τα ηλεκτρόδια κατασκευάζονται επίσης από αγώγιμα χρώματα ή από επικάλυψη γραφίτη.

Η μεταφορά δεδομένων από τον πομποδέκτη στον αναγνώστη γίνεται με διαμόρφωση φορτίου. Μεταβάλλοντας μία κυκλωματική παράμετρο του πομποδέκτη, μεταβάλλεται το ηλεκτρικό πεδίο του αναγνώστη.

Συγκεκριμένα, όταν ένας ηλεκτρικά συζευγμένος πομποδέκτης τοποθετηθεί στη ζώνη ανάγνωσης του αναγνώστη, η αντίσταση εισόδου R_L του πομποδέκτη προκαλεί μια μικρή εξασθένηση στο συντονισμένο κύκλωμα του αναγνώστη. Με την προσθαφαίρεση μιας αντίστασης R_{mod} σε παράλληλο κύκλωμα με την R_L (εικόνα 36) σε χρονισμό με τα δεδομένα που πρέπει να μεταδοθούν, η προαναφερθείσα εξασθένηση εναλλάσσεται ανάμεσα σε δύο τιμές. Το αποτέλεσμα είναι η διαμόρφωση πλάτους της τάσης στα άκρα του αναγνώστη και η μετάδοση των δεδομένων.

Οι πιο συχνά χρησιμοποιούμενες κεραίες αναγνώστων των χωρητικά συζευγμένων συστημάτων είναι το μονόπολο πάνω από επίπεδη επιφάνεια, η meander-line κεραία και η ελικοειδής κεραία, ενώ ένα τυπικό είδος κεραίας ευαίσθητης στα ηλεκτρικά πεδία που χρησιμοποιείται στους πομποδέκτες των χωρητικά συζευγμένων συστημάτων είναι η κεραία bow-tie.

Τα συστήματα χωρητικής σύζευξης χρησιμοποιούνται μόνο όταν η απόσταση του αναγνώστη και του πομποδέκτη είναι μέχρι λίγα, γιατί επηρεάζονται πολύ από μεταλλικά αντικείμενα στο χώρο. Για παράδειγμα, τέτοια εφαρμογή είναι οι κάρτες πρόσβασης.

2.13 Κωδικοποίηση δεδομένων

Στα συστήματα RFID, η επιλογή της τεχνικής κωδικοποίησης γίνεται με βάση κάποια κριτήρια. Η τεχνική κωδικοποίησης δεν πρέπει να περιορίζει σε μεγάλο βαθμό την ισχύ που λαμβάνει ο πομποδέκτης, ούτε να καταλαμβάνει μεγάλο εύρος ζώνης κυρίως κατά την επικοινωνία του αναγνώστη προς τον πομποδέκτη, ενώ πρέπει να επιτρέπει την ανίχνευση συγκρούσεων.

Τα πιο πολλά συστήματα RFID χρησιμοποιούν κωδικοποίηση PPM ή PWM κατά την επικοινωνία του αναγνώστη προς τον πομποδέκτη και Manchester ή NRZ κατά την αντίθετη επικοινωνία.

2.14 Διαμόρφωση δεδομένων

Κατά τη μεταφορά δεδομένων από τον αναγνώστη στον πομποδέκτη χρησιμοποιούνται όλες οι γνωστές διαδικασίες ψηφιακής διαμόρφωσης (ASK, FSK, PSK) με προτιμητέα την ASK διαμόρφωση, λόγω της εύκολης αποδιαμόρφωσης της.

Κατά τη μεταφορά δεδομένων από τον πομποδέκτη στον αναγνώστη, στα επαγωγικά συζευγμένα συστήματα προτιμάται η ASK διαμόρφωση κατά τη διαμόρφωση φορτίου (με ή χωρίς υποφέρον), ενώ στα μικροκυματικά συστήματα η διαμορφωμένη ανάκλαση γίνεται κυρίως με PSK διαμόρφωση.

2.15 Πλεονεκτήματα συστημάτων RFID

Τα σημαντικά **πλεονεκτήματα** που προσφέρει το RFID μπορούν να συνοψιστούν στα παρακάτω:

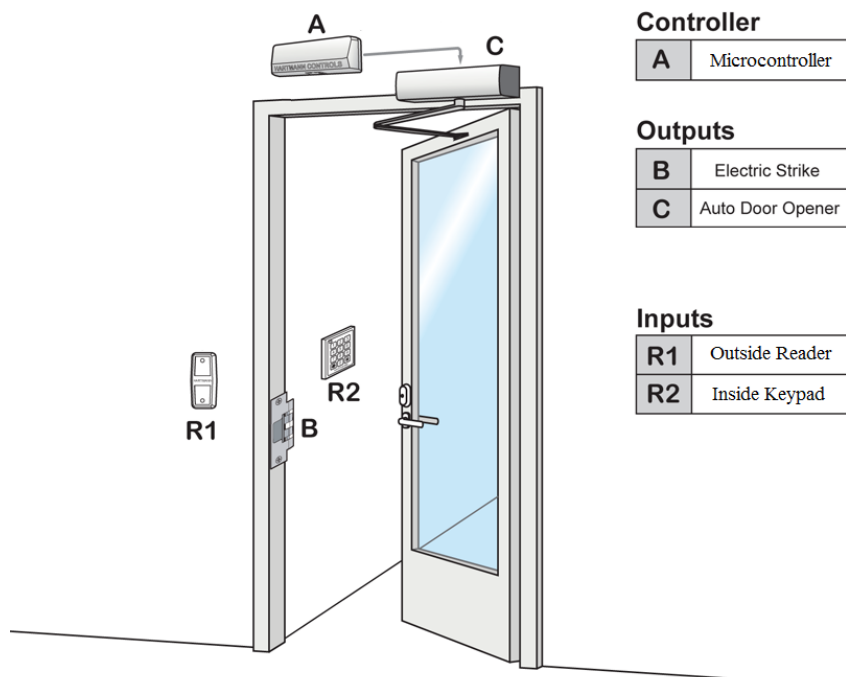
- Η **αναγνώριση μπορεί να γίνει από απόσταση** μιας και υπάρχουν RFID tags που είναι σε θέση παίρνοντας ενέργεια από κάποια πηγή που συνήθως είναι μπαταρία να στείλουν τις πληροφορίες στον δέκτη και να αυξάνουν την εμβέλεια. Δεν χρειάζεται φυσική και οπτική επαφή των RFID tags με τον αναγνώστη, πράγμα το οποίο επιταχύνει ουσιαστικά διαδικασίες σε εφαρμογές.
- Σε αντίθεση με τα Bar Code, τα RFID είναι **ανθεκτικά στη χρήση και τις ακραίες θερμοκρασίες**.
- Υπάρχει η **δυνατότητα αποθήκευσης περισσότερων δεδομένων** σε σχέση με τα Bar Code. Η ποσότητα εξαρτάται από τον προμηθευτή και την εφαρμογή, αλλά τυπικά δεν υπερβαίνει τα 2KB δεδομένων στον τομέα των "contactless memories".
- Χάρη στην απουσία επαφής ανάμεσα στον φορέα δεδομένων και στον αναγνώστη (όχι όπως σε αναγνώστες που εισέρχεται ή υπάρχει οπτική επαφή της κάρτας) ελαχιστοποιείται η επιβάρυνση μηχανικής μορφής ή ρύπων, με αποτέλεσμα τη **μεγαλύτερη διάρκεια λειτουργίας και λιγότερες συντηρήσεις**, οι οποίες τελικά μεταφράζονται σε μια απλή αλλαγή μπαταρίας.
- Φορείς δεδομένων με τσιπ RFID αναγνωρίζονται γρήγορα και αξιόπιστα και παρέχουν ασφαλή μετάδοση δεδομένων. Συνοδευτικά, είναι εφικτή η **παράλληλη ανάγνωση πολλών φορέων ταυτόχρονα**, χωρίς απώλειες πληροφοριών και βλάβες του συστήματος (anticollision).
- Τα RFID tags μπορούν να μην είναι ορατά από το ανθρώπινο μάτι και **υπάρχουν σε διαφορετικές φόρμες**: hard tags, πλαστικές κάρτες, χάρτινα εισιτήρια, ετικέτες, αυτοκόλλητα, μπρελόκ, περικάρπια και πολλούς άλλους τύπους - πάντα προσαρμοσμένα στις απαιτήσεις της εφαρμογής!
- Η τεχνολογία RFID παρέχει μέσω δικαιωμάτων πρόσβασης και κρυπτογραφίας ένα **υψηλότατο επίπεδο ασφάλειας και προστασίας δεδομένων**.
- **Δυνατότητα προγραμματισμού εξ' αποστάσεως**.
- Μπορούν να μην είναι ορατά στο ανθρώπινο μάτι τα RFID tags μιας και για την αναγνώριση τους δεν χρειάζεται οπτικό μέσο.
- Επιπρόσθετες λειτουργίες πχ παρακολούθηση και καταγραφή της θερμοκρασίας

3. Υλοποίηση

Σε ένα κοινόχρηστο χώρο με πολλές αίθουσες, όπως αυτός για τον οποίο απευθύνεται η συγκεκριμένη υλοποίηση έχει πρόσβαση αρκετό προσωπικό, κάποιοι είναι γνωστοί αλλά αρκετοί είναι άγνωστοι. Επομένως, ο έλεγχος πρόσβασης αποτελεί παράγοντα ασφάλειας και ελέγχου για το χώρο αλλά και για τους εργαζόμενους.

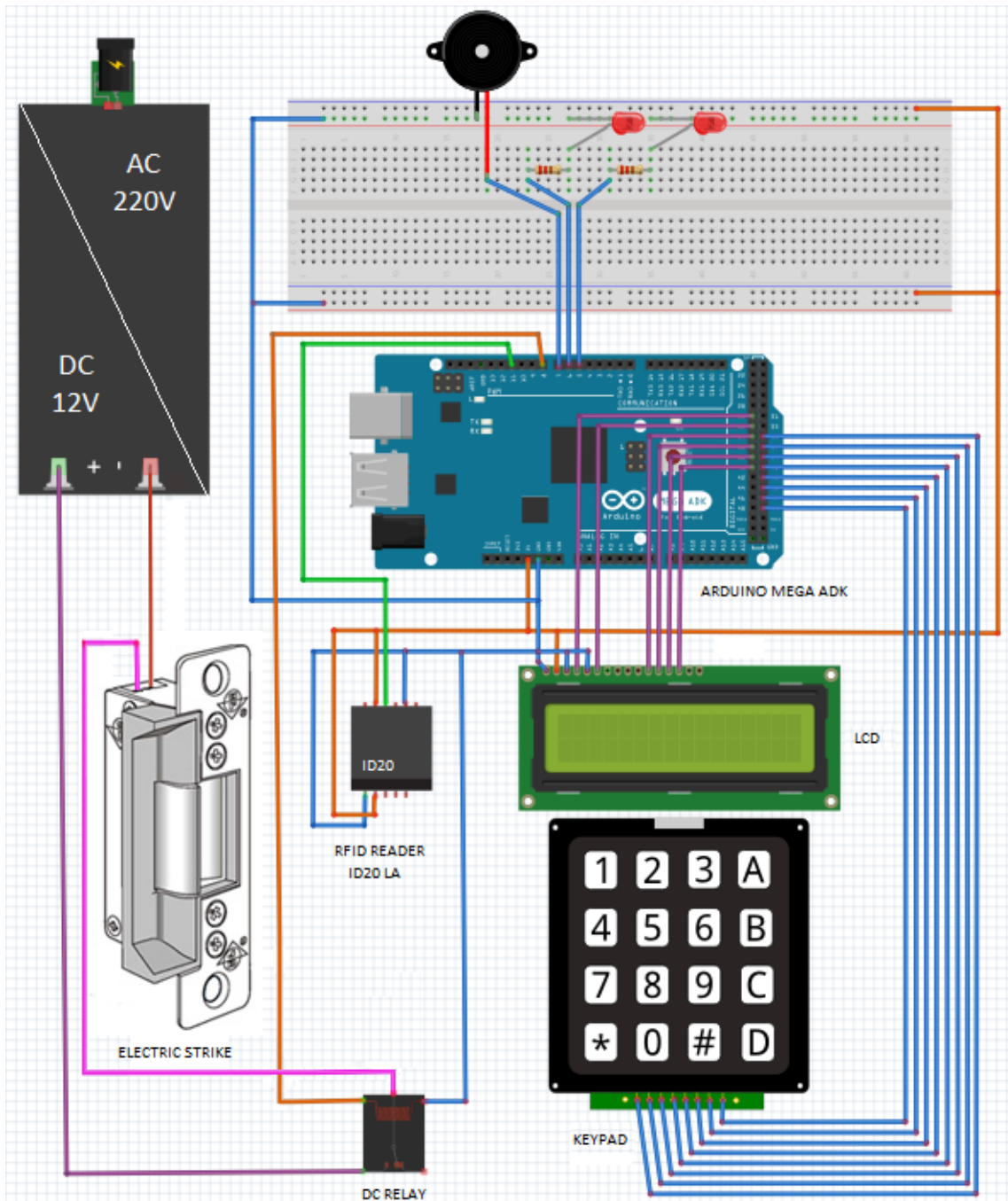
Για το λόγο αυτό, αρχικά σχεδιάστηκε και στη συνέχεια υλοποιήθηκε, ένα σύστημα το οποίο επιτρέπει την είσοδο σε ένα εργαζόμενο μόνο εάν έχει στην κατοχή του, το «κλειδί» για αυτό τον χώρο. Στην περίπτωση μας το κλειδί για την είσοδο δεν είναι ένα κοινό κλειδί πόρτας, αλλά είναι είτε μια κάρτα-κλειδί, είτε ένα κωδικός πρόσβασης. Σε κάθε αίθουσα εγκαθίσταται ένας μικροελεγκτής arduino mega, ο οποίος συνδέεται μέσω ενός Ethernet shield σε ένα τοπικό δίκτυο με ένα πληροφοριακό σύστημα που εκτελείται σε ένα διακομιστή που εδώ είναι το Cubieboard με λειτουργικό σύστημα Cubian.

Ο μικροελεγκτής διαβάζει το κλειδί εισόδου κάθε εργαζομένου (RFID tag) και επικοινωνεί με το πληροφοριακό σύστημα, το οποίο θα εγκρίνει ή θα απορρίψει την πρόσβαση. Στο πληροφοριακό σύστημα υπάρχει μια κεντρική κονσόλα διαχείρισης, που υλοποιείται μέσω ενός ιστοχώρου, ο οποίος ύστερα από κατάλληλη αυθεντικοποίηση, παρέχει όλες τις διαχειριστικές λειτουργίες. Από την στιγμή που επιβεβαιωθεί το κλειδί εισόδου του εργαζομένου ενεργοποιείται για δεδομένο χρονικό διάστημα η ηλεκτρική κλειδαριά όπου επιτρέπει στον εργαζόμενο την είσοδο στον ελεγχόμενο χώρο. Για την έξοδο του από αυτόν υπάρχει ένα keypad με οθόνη lcd όπου ο χρήστης πρέπει να εισάγει έναν τετραψήφιο κωδικό κοινό για όλους όσους βρίσκονται στην ελεγχόμενη περιοχή και έπειτα από έγκυρη εισαγωγή ενεργοποιείται ξανά η ηλεκτρική κλειδαριά (electric strike) που επιτρέπει στον εργαζόμενο την έξοδο του από τον χώρο αυτό όπως φαίνεται παρακάτω (εικόνα 37).



Εικόνα 37: Εγκατάσταση μονάδων σε σημείο ελεγχόμενης πρόσβασης

Στην εικόνα 38 παρουσιάζεται ο τρόπος διασύνδεσης του συστήματος όπως έχει υλοποιηθεί για το κομμάτι που αφορά την τοπική πρόσβαση και εμφανίζεται στην προηγούμενη εικόνα. Το σύστημα θα τροφοδοτηθεί με 220V για την ηλεκτρική κλειδαριά, ενώ ο μικροελεγκτής θα τροφοδοτηθεί μέσω Ethernet από ένα power over Ethernet switch από όπου θα γίνει μεταφορά και των data στον Cubieboard server.



Εικόνα 38: Τρόπος διασύνδεσης του συστήματος που έχει υλοποιηθεί

4. Hardware (Υλικό)

Η υλοποίηση του συστήματος ελέγχου πρόσβασης βασίστηκε στο μικροελεγκτή Arduino Mega και στήθηκε γύρω από αυτό. Η αρχιτεκτονική Arduino επιτρέπει τη σύνδεση διαφόρων εξαρτημάτων (αισθητήρων, συσκευών κτλ.) τα οποία με τον κατάλληλο προγραμματισμό επιτελούν τον αντίστοιχο στόχο.

Ο μικροελεγκτής ελέγχει και διαχειρίζεται ένα σύνολο υλικών-εξαρτημάτων:

- Ethernet Shield
- RFID Reader
- Keypad
- Ηλεκτρική κλειδαριά (Electric Strike)

Ουσιαστικά το Arduino δέχεται πληροφορίες, αλληλοεπιδρώντας με το περιβάλλον, μέσω των υλικών και των εξαρτημάτων που είναι συνδεδεμένα σε αυτό (RFID Reader κλπ). Στη συνέχεια στέλνει αυτά τα δεδομένα στο server (Cubieboard) μέσω Ethernet lan και αφού λάβει την απάντηση δρα αντίστοιχα.

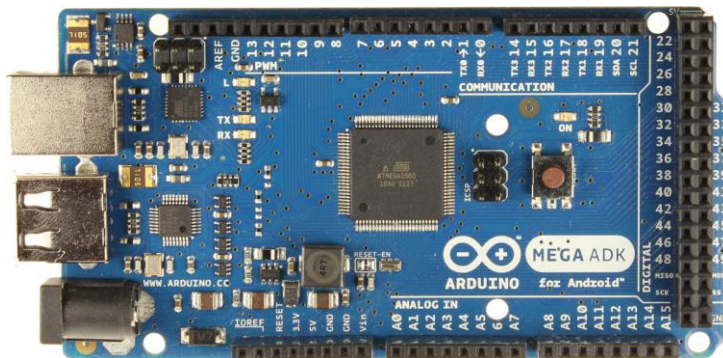
Παρακάτω θα ακολουθήσει μια περιγραφή του hardware που έχει χρησιμοποιηθεί στην υλοποίηση της διπλωματικής.

4.1 Arduino Mega ADK rev3

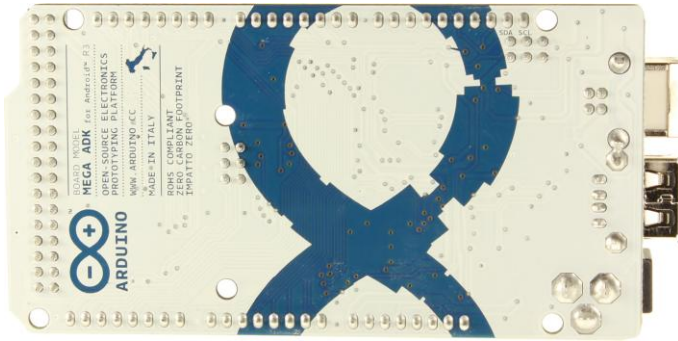
Στην ενότητα αυτή γίνεται αναφορά στο υλικό του Arduino Mega ADK rev3 με περιγραφή όλων των ακροδεκτών I/O του board καθώς και των τεχνικών χαρακτηριστικών του.

4.1.1 Schematics & Reference Design του Arduino Mega ADK rev3

Στην *εικόνα 39α & β* απεικονίζονται τα Schematics του Arduino Mega ADK rev3.

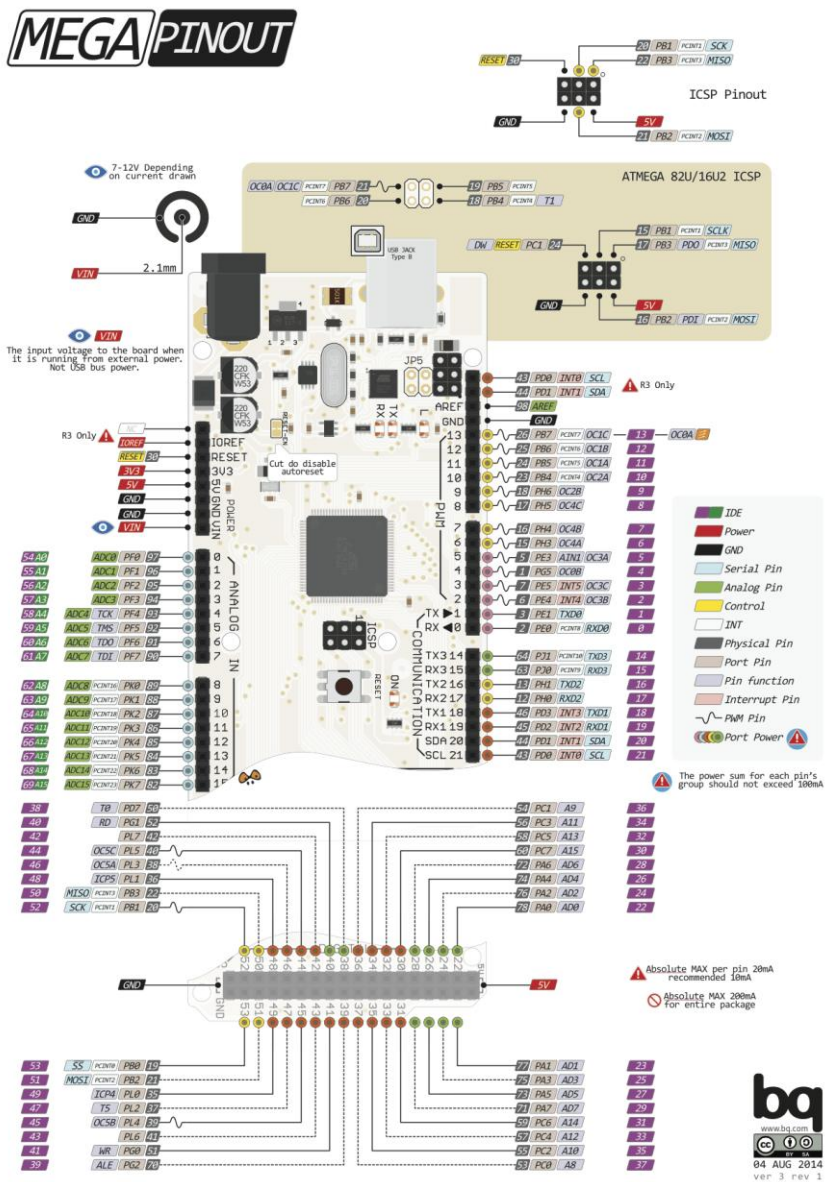


Εικόνα 39α: Front Schematics of Arduino Mega ADK rev3



Εικόνα 39β: Back Schematics of Arduino Mega ADK rev3

Έπειτα στην εικόνα 40 απεικονίζεται το Reference Design του Arduino Mega rev3.



Στο τελευταίο revision (**R3**) του Arduino Mega έγιναν οι παρακάτω τροποποιήσεις:

- ✓ Αντικαταστάθηκε ο ATmega8U2 με τον ATmega16U2 για το USB-to-Serial Converter.
- ✓ Έγινε προσθήκη των SDA και SCL pins για TWI επικοινωνία και τοποθέτηση δίπλα στο AREF pin.
- ✓ Επιπλέον προστέθηκε το IOREF όπου επιτρέπει στα onboard shields να προσαρμόζονται στην τάση που παρέχεται από το βασικό board, αλλά και ένα ακόμα pin το οποίο δεν χρησιμοποιείται αυτή την στιγμή αλλά θα ενσωματωθεί μελλοντικά.
- ✓ Έγινε βελτίωση του κυκλώματος στο κουμπί του RESET.

4.1.2 Ακροδέκτες POWER του Arduino Mega ADK rev3

Το Arduino Mega ADK μπορεί να τροφοδοτηθεί από τον υπολογιστή μέσω της σύνδεσης USB, ή από εξωτερική τροφοδοσία που παρέχεται μέσω μιας υποδοχής φισ των 2.1mm (θετικός πόλος στο κέντρο) και βρίσκεται στην κάτω-αριστερή γωνία του Arduino.

Αν χρησιμοποιηθεί μπαταρία πρέπει να συνδεθεί στα pin headers Vin και Gnd του POWER. Επιπλέον πρέπει να αναφερθεί ότι το Mega είναι και USB Host πάνω στο οποίο μπορούν να συνδεθούν κινητά τηλέφωνα android οπότε θεωρείται βέλτιστο σε αυτή την περίπτωση χρησιμοποιηθεί εξωτερικό τροφοδοτικό τουλάχιστον 1.5A .

Για να μην υπάρχουν προβλήματα, η εξωτερική τροφοδοσία πρέπει να είναι από **7 ως 12V** και μπορεί να προέρχεται από ένα κοινό μετασχηματιστή του εμπορίου, από μπαταρίες ή οποιαδήποτε άλλη πηγή DC.

- ✚ Αν τροφοδοτηθεί με λιγότερα από 7V τότε το 5V pin μπορεί να έχει λιγότερα από πέντε volts και το board ίσως να παρουσιάσει αστάθειες.
- ✚ Αν τροφοδοτηθεί με περισσότερα από 12V ο voltage regulator μπορεί να υπερθερμανθεί και να καταστρέψει το board.

Η *εικόνα 41* παρουσιάζει εισόδους / εξόδους των ακροδεκτών **POWER** του Arduino Mega.



Εικόνα 41: Power pins of Arduino Mega ADK rev3

Οι ακροδέκτες POWER είναι οι ακόλουθοι:

- **VIN Pin:** Η τάση εισόδου της πλακέτας όταν χρησιμοποιούμε εξωτερική πηγή ενέργειας. Η τροφοδοσία τάσης γίνεται μέσω του ακροδέκτη Vin και Gnd. Αν όμως έχουμε ήδη συνδεδεμένη εξωτερική τροφοδοσία μέσω του φις των 2.1mm (DC power jack 7-12V) μπορούμε να χρησιμοποιήσουμε το VIN pin για να τροφοδοτήσουμε άλλα εξαρτήματα με την πλήρη τάση της εξωτερικής τροφοδοσίας (7-12V), πριν αυτή περάσει από τον ρυθμιστή τάσης όπως γίνεται με το pin των 5V.
- **GND Pin:** Είσοδοι γείωσης.
- **5V Pin:** Η τάση αυτή προέρχεται άμεσα από την θύρα USB (5V), ή από την εξωτερική τροφοδοσία μέσω του φις των 2.1mm (7-12V) ή από το Vin pin του board (7-12V) αφού περάσει από ένα ρυθμιστή τάσης για να την «φέρει» στα 5V.

ΠΡΟΣΟΧΗ: Αν τροφοδοτηθεί απευθείας μέσω των pin 5V ή 3.3V pins γίνεται προσπέραση του regulator και μπορεί να καταστραφεί το board.

- **3.3V Pin:** Η τάση αυτή παράγεται από το ολοκληρωμένο FTDI μέσω του on-board regulator. Το μέγιστο ρεύμα που μπορεί να αντλήσει είναι 50mA.
- **Reset Pin:** Αν δοθεί LOW σε αυτό το pin γίνεται reset ο microcontroller. Τυπικά το pin αυτό χρησιμοποιείται όταν προσθέτουμε διάφορα shields και κρύβουν το reset button που ήδη υπάρχει στο board.
- **IOREF Pin:** Αυτό το pin παρέχει στο board το voltage reference με το οποίο λειτουργεί ο microcontroller.

4.1.3 Ακροδέκτες ANALOG IN του Arduino Mega ADK rev3

Στην κάτω πλευρά του Arduino με τη σήμανση ANALOG IN όπως φαίνεται και στην εικόνα 3 υπάρχει μια ακόμη σειρά από 16 pin αριθμημένα από το 0 ως 15. Η τάση αναφοράς (Reference Voltage) για τις αναλογικές εισόδους μπορεί να ρυθμιστεί με μια εντολή αφού τροφοδοτήσουμε εξωτερικά με αυτή την τάση το pin με τη σήμανση AREF που βρίσκεται στην απέναντι πλευρά της πλακέτας.

Η εικόνα 42 παρουσιάζει τα αναλογικά pins των ακροδεκτών ANALOG IN του Arduino Mega ADK rev3.



Εικόνα 42: Analog In pins of Arduino Mega ADK rev3

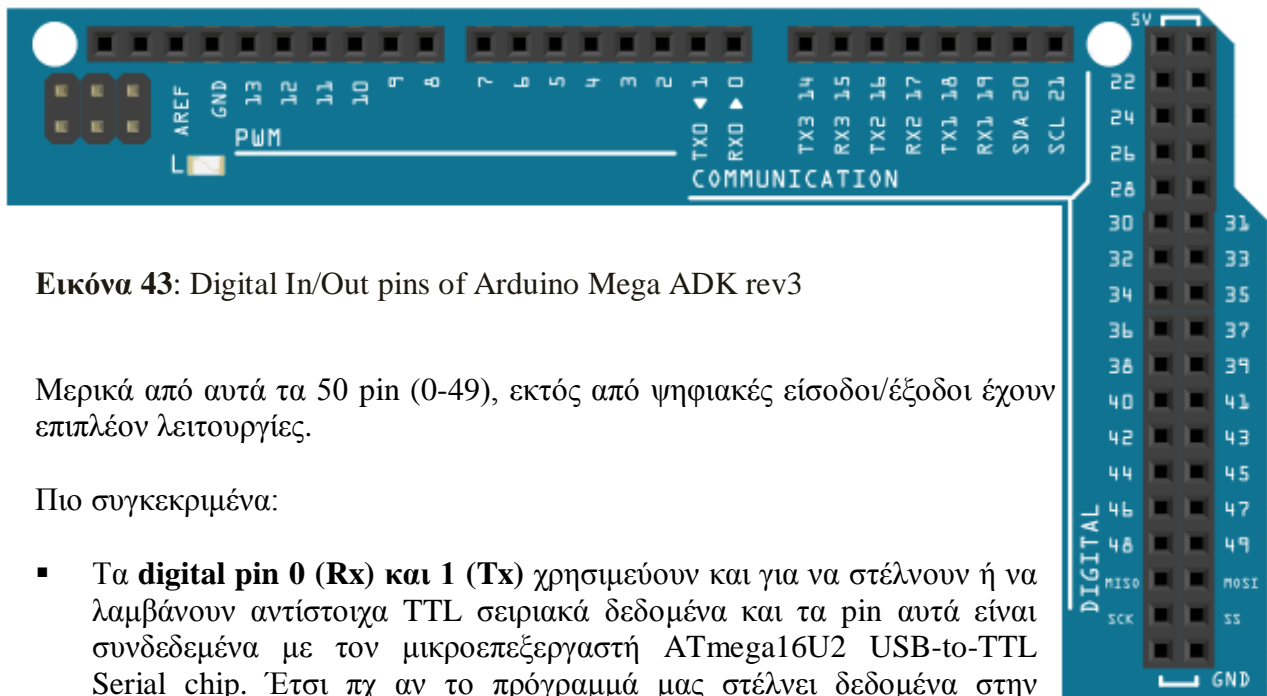
4.1.4 Ακροδέκτες DIGITAL IN/OUT του Arduino Mega ADK rev3

Στην πάνω και δεξιά πλευρά του Arduino βρίσκονται 50 θηλυκά digital pin αριθμημένα από 0-49 όπου μπορούν να λειτουργήσουν ως ψηφιακές εισοδοί & έξοδοι χρησιμοποιώντας τις [pinMode\(\)](#), [digitalWrite\(\)](#) και [digitalRead\(\)](#) functions. Λειτουργούν στα 5V και καθένα pin μπορεί να παρέχει ή να δεχτεί το πολύ 40mA.

Ως ψηφιακή έξοδος το κάθε pin από αυτά μπορεί να τεθεί μέσω κατάλληλου προγραμματισμού σε κατάσταση HIGH ή LOW, οπότε το Arduino θα ξέρει αν πρέπει να διοχετεύσει ή όχι ρεύμα στο συγκεκριμένο pin. Με αυτόν τον τρόπο μπορούμε πχ να ανάψουμε ή να σβήσουμε ένα LED συνδεδεμένο στο αντίστοιχο pin.

Αν πάλι ρυθμίσουμε ένα από αυτά τα pin ως ψηφιακή είσοδο μπορούμε με την κατάλληλη εντολή να διαβάσουμε την κατάσταση του (HIGH ή LOW) ανάλογα με το αν η εξωτερική συσκευή που έχει συνδεθεί σε αυτό το pin διοχετεύει ή όχι ρεύμα στο pin, οπότε με αυτόν τον τρόπο μπορούμε πχ να «διαβάσουμε» την κατάσταση ενός διακόπτη.

Οι *εικόνα 43* παρουσιάζει τα ψηφιακά pins των ακροδεκτών **DIGITAL IN/OUT** του Arduino Mega ADK rev3.



Εικόνα 43: Digital In/Out pins of Arduino Mega ADK rev3

Μερικά από αυτά τα 50 pin (0-49), εκτός από ψηφιακές εισοδοί/έξοδοι έχουν επιπλέον λειτουργίες.

Πιο συγκεκριμένα:

- Τα **digital pin 0 (Rx)** και **1 (Tx)** χρησιμεύουν και για να στέλνουν ή να λαμβάνουν αντίστοιχα TTL σειριακά δεδομένα και τα pin αυτά είναι συνδεδεμένα με τον μικροεπεξεργαστή ATmega16U2 USB-to-TTL Serial chip. Έτσι πχ αν το πρόγραμμά μας στέλνει δεδομένα στην σειριακή αυτά προωθούνται στην θύρα USB (μέσω του ελεγκτή Serial-Over-USB), αλλά & στο pin 0 (Rx) για να τα διαβάσει ενδεχομένως μια άλλη συσκευή (πχ δεύτερο Arduino στο δικό του pin1).

ΠΡΟΣΟΧΗ: Αν στο πρόγραμμά μας ενεργοποιήσουμε το serial interface αυτό σημαίνει ότι αυτόματα χάνουμε 2 ψηφιακές εισόδους/εξόδους. Επιπλέον TTL σειριακά δεδομένα μπορούν να στέλνουν ή να λαμβάνουν τα **digital pin 19 (Rx)** και **18 (Tx)** ως Serial 1, τα **digital pin 17 (Rx)** και **16 (Tx)** ως Serial 2, τα **digital pin 15 (Rx)** και **14 (Tx)** ως Serial 3. Used to receive (RX) and transmit (TX) TTL serial data.

- Τα **digital pin 2 και 3** μπορούν να ρυθμιστούν και ως εξωτερικά interrupt (interrupt 0 και 1 αντίστοιχα), ώστε να λειτουργούν αποκλειστικά ως ψηφιακές εισοδοι στις οποίες όταν συμβαίνουν συγκεκριμένες αλλαγές η κανονική ροή του προγράμματος σταματάει άμεσα και εκτελείται μια συγκεκριμένη συνάρτηση. Την ίδια χρήση έχουν και τα **digital pin 18** (interrupt 5), **digital pin 19** (interrupt 4), **digital pin 20** (interrupt 3) και **digital pin 21** (interrupt 2). Τα εξωτερικά interrupt είναι ιδιαίτερα χρήσιμα σε εφαρμογές που απαιτούν συγχρονισμό μεγάλης ακρίβειας και μπορεί να ρυθμιστούν για να προκαλέσουν την διακοπή μιας χαμηλής τιμής, μια άνοδο ή την πτώση ακμής. Για περισσότερες λεπτομέρειες στην [attachInterrupt\(\)](#) function.
- Τα **digital pin 2 έως 13 και 44 έως 46** μπορούν να λειτουργήσουν και ως ψευδο-αναλογικές έξοδοι με το σύστημα PWM (Pulse Width Modulation), δηλαδή το ίδιο σύστημα που διαθέτουν οι μητρικές των υπολογιστών για να ελέγχουν τις ταχύτητες των ανεμιστήρων. Έτσι έχουμε 8bit PWM output μέσω της [analogWrite\(\)](#) function και μπορούμε πχ να συνδέσουμε ένα LED σε κάποιο από αυτά τα pin και να ελέγξουμε πλήρως την φωτεινότητά του με ανάλυση 8bit (256 καταστάσεις από 0-σβηστό ως 255-πλήρως αναμμένο) αντί να έχουμε απλά την δυνατότητα αναμμένο-σβηστό που παρέχουν οι υπόλοιπες ψηφιακές έξοδοι. Το PWM δεν είναι πραγματικά αναλογικό σύστημα και θέτοντας στην έξοδο την τιμή 127, δεν σημαίνει ότι η έξοδος θα δίνει 2.5V αντί της κανονικής τιμής των 5V, αλλά ότι θα δίνει ένα παλμό που θα εναλλάσσεται με μεγάλη συχνότητα και για ίσους χρόνους μεταξύ των τιμών 0 και 5V.
- Τα **digital pin 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS)** υποστηρίζουν SPI communication χρησιμοποιώντας την [SPI library](#).
- Στο **digital pin 13** υπάρχει ένα built-in LED συνδεδεμένο όπου όταν το pin βρίσκεται σε τιμή HIGH το ενσωματωμένο LED είναι αναμμένο, ενώ όταν το pin είναι σε LOW το LED είναι σβηστό.
- **AREF pin:** Το Arduino Mega έχει 16 αναλογικές εισόδους A0 έως A15 όπου από default μετράνε GND έως 5Volts. Από αυτό το pin είναι δυνατόν να αλλαχτεί αυτό το Reference Voltage για τις αναλογικές εισόδους μέσω της [analogReference\(\)](#) function.
- **I2C:** Τα **digital pin 20 (SDA) και 21 (SCL)** έχουν προστεθεί για υποστήριξη TWI (Two-Wire Interface) communication χρησιμοποιώντας την [Wire library](#). Στο Arduino Mega ADK rev3 board η βιβλιοθήκη αυτή σου επιτρέπει να επικοινωνείς με I2C / TWI devices μέσω των pins SDA (data line), SCL (clock line) και GND.
- **USB Host MAX3421E** επικοινωνεί με το Arduino μέσω του SPI bus χρησιμοποιώντας τα digital pins 7 (RST), 50 (MISO), 51 (MOSI), 52 (SCK). Είναι σημαντικό να MHN προγραμματιστεί το digital pin 7 ως είσοδο ή έξοδο, διότι χρησιμοποιείται για την επικοινωνία με το MAX3421E.

4.1.5 Ακροδέκτες Communication του Arduino Mega ADK rev3

Το Arduino Mega ADK έχει ένα μεγάλο αριθμό pins για επικοινωνία με έναν υπολογιστή, άλλο Arduino ή μικροελεγκτή. Ο ATmega2560 παρέχει 4 hardware UARTs για TTL (5V) σειριακή επικοινωνία. Ο ATmega8U2 χρησιμοποιεί μια από αυτές πάνω από USB και παρέχει ένα virtual com port στον υπολογιστή (τα windows machines χρειάζονται ένα .inf αρχείο, αλλά τα OSX και Linux machines αναγνωρίζουν το board σαν COM port αυτόματα). Το Arduino software 1.0.6 παρέχει στον χρήστη ένα serial monitor που επιτρέπει δεδομένα να σταλούν από και προς το board.

Τα Rx και Tx LEDs του board αναβοσβήνουν όταν δεδομένα μεταδίδονται μέσω των ATmega8U2/16U2 chip και USB σύνδεσης στον υπολογιστή, αλλά όχι για την σειριακή επικοινωνία μέσω των pins 0 και 1. Η SoftwareSerial library επιτρέπει την σειριακή επικοινωνία όλων των digital pins του Mega. Ο ATmega2560 υποστηρίζει TWI και SPI επικοινωνία, ενώ περιέχει την Wire library για να απλουστεύσει την χρησιμοποίηση του TWI bus όπως και την SPI library αντίστοιχα. Το USB host interface που δίνεται από το MAX3421E IC επιτρέπει στο Arduino Mega ADK να συνδεθεί και να αλληλοεπιδράσει με οποιαδήποτε συσκευή έχει USB port. Για παράδειγμα, σου επιτρέπει να αλληλεπιδράσεις με πολλές κατηγορίες κινητών τηλεφώνων, να ελέγχεις Canon κάμερες, να διασυνδέεις πληκτρολόγια/mouse ή ακόμα και game controllers όπως του Wii και του PS4.

4.1.6 Τεχνικά χαρακτηριστικά Arduino Mega ADK rev3

Ο μικροεπεξεργαστής ATmega2560 που βρίσκεται στο Arduino Mega rev3 έχει τρεις ομάδες μνήμης. Διαθέτει flash memory στην οποία αποθηκεύονται τα Arduino sketch, SRAM (static random access memory) στην οποία δημιουργείται το sketch και χρησιμοποιεί τις μεταβλητές όταν τρέχει, και EEPROM η οποία χρησιμοποιείται από τους προγραμματιστές για την αποθήκευση μακροχρόνιων πληροφοριών. Πιο συγκεκριμένα:

- **256KB μνήμης Flash:** 8KB χρησιμοποιούνται από το firmware (bootloader) του Arduino που έχει εγκαταστήσει ήδη ο κατασκευαστής του. Το firmware είναι αναγκαίο για την εγκατάσταση προγραμμάτων στο μικροελεγκτή μέσω της θύρας USB. Τα υπόλοιπα 248KB της μνήμης Flash χρησιμοποιούνται για την αποθήκευση αυτών ακριβώς των προγραμμάτων, αφού πρώτα μεταγλωττιστούν στον υπολογιστή. Η μνήμη Flash, δε χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή επανεκκίνησης.
- **8KB μνήμης SRAM:** Η ωφέλιμη μνήμη που μπορούν να χρησιμοποιήσουν τα προγράμματα για να αποθηκεύουν μεταβλητές, πίνακες κ.λπ. Η μνήμη χάνει τα δεδομένα της όταν η παροχή ρεύματος στο Arduino σταματήσει ή πατηθεί το κουμπί επανεκκίνησης.
- **4KB μνήμης EEPROM:** Μπορεί να χρησιμοποιηθεί για εγγραφή ή ανάγνωση δεδομένων από τα προγράμματα. Σε αντίθεση με την SRAM, δε χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή επανεκκίνησης.

Ο πίνακας 2 παρουσιάζει αναλυτικότερα τα τεχνικά χαρακτηριστικά του Arduino Mega ADK rev3.

Τεχνικά χαρακτηριστικά Arduino Mega ADK Rev3	
Microcontroller:	ATmega2560
Operating Voltage:	5V
Input Voltage (recommended):	9V
Input Voltage (limits):	7-18V
Digital I/O Pins:	54 (14 provide PWM output)
Analog Input Pins:	16
DC Current per I/O Pin:	40 mA
DC Current for 3.3V Pin:	50 mA
Flash Memory:	256 KB (8kb used by bootloader)
SRAM:	8 KB
EEPROM:	4 KB
Clock Speed:	16 MHz

Πίνακας 2: Τεχνικά χαρακτηριστικά Arduino Mega ADK Rev3

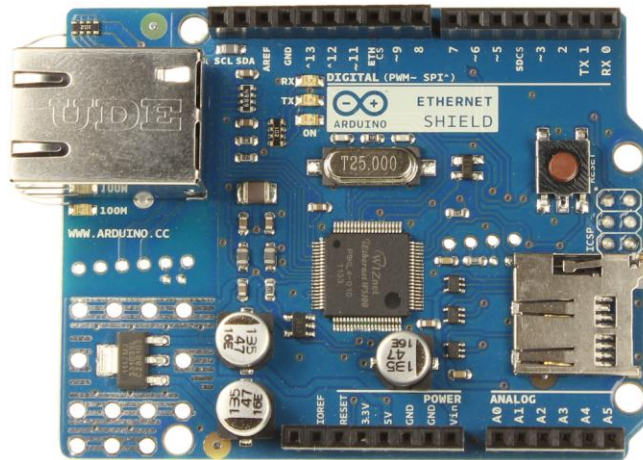
4.2 Arduino Ethernet Shield rev3

Η έννοια του shield στην αρχιτεκτονική Arduino είναι η ενσωμάτωση επιπλέον υλικού (hardware) στον μικροελεγκτή που του προσδίδει μια νέα ιδιότητα, κυρίως σε θέματα επικοινωνίας. Με την προσαρμογή του υλικού αυτού η επικοινωνία από σειριακή (μέσω usb) μετατρέπεται σε αυτή παρέχει το νέο shield και στην δική μας περίπτωση σε Ethernet.

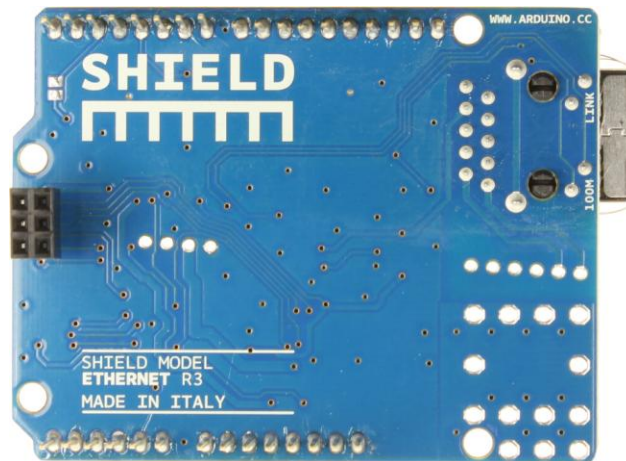
Το Arduino Ethernet Shield επιτρέπει στο Arduino board να συνδέεται στο Internet. Βασίζεται στο Wiznet W5100 Ethernet chip το οποίο παρέχει ένα ip stack κατάλληλο τόσο για μεταφορά TCP όσο και UDP πακέτων ενώ επιτρέπει μέχρι 4 ταυτόχρονα συνδέσεις socket χρησιμοποιώντας την Ethernet library. Το Ethernet Shield έχει ένα clip RJ45 με ενσωματωμένο ένα line transformer και δυνατότητα για power over Ethernet τροφοδοσία. Επιπλέον πάνω στο shield υπάρχει ένα micro-SD card slot το οποίο μπορεί να χρησιμοποιηθεί για να αποθηκεύσει αρχεία που θα χρησιμοποιούνται στο δίκτυο και το shield είναι συμβατό τόσο με το Uno board όσο και με το Mega board μέσω της Ethernet library. Πάνω στο shield υπάρχει ένα reset controller για να σιγουρέψει ότι το W5100 Ethernet module έχει γίνει κατάλληλα reset στο power-up. Προηγούμενα revisions του shield δεν ήταν συμβατά με το Mega και χρειαζόνταν να κάνεις χειροκίνητα reset σε κάθε power-up.

Το Arduino Mega επικοινωνεί με το W5100 και την SD card χρησιμοποιώντας το SPI bus μέσω του ICSP header στα pins 50, 51, 52. Επιπλέον τα pin 4 και 10 χρησιμοποιούνται για την επιλογή του W5100 / SD card αντίστοιχα και δεν μπορούν να χρησιμοποιηθούν για διαφορετική I/O χρήση. Ακόμα στο Mega το pin 53 πρέπει πάντα να δηλώνεται ως OUTPUT, διότι διαφορετικά το SPI bus δεν θα δουλέψει σωστά.

Στην εικόνα 44α & β απεικονίζονται τα Schematics του Arduino Ethernet Shield rev3.



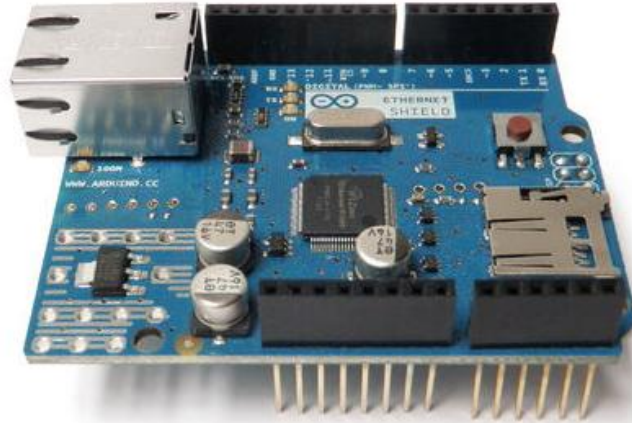
Εικόνα 44α: Front Schematics of Arduino Ethernet Shield rev3



Εικόνα 44β: Back Schematics of Arduino Ethernet Shield rev3

Το shield περιλαμβάνει έναν αριθμό από LEDs:

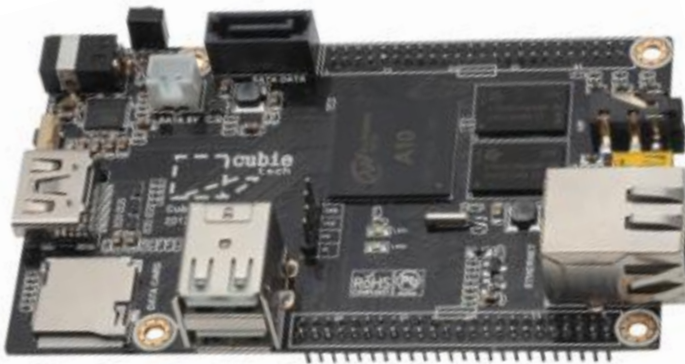
- **PWR:** Μας πληροφορεί ότι το board και το shield έχουν ρεύμα
- **LINK:** Μας πληροφορεί ότι έχει ανιχνευτεί δίκτυο δεδομένων και αναβοσβήνει όταν το shield στέλνει/λαμβάνει δεδομένα
- **FULLD:** Μας πληροφορεί ότι το δίκτυο δεδομένων είναι Full Duplex
- **100M:** Μας πληροφορεί ότι έχει ανιχνευτεί δίκτυο 100Mb/s (σε αντίθεση με αυτά των 10Mb/s)
- **Rx:** Αναβοσβήνει όταν το shield λαμβάνει δεδομένα
- **Tx:** Αναβοσβήνει όταν το shield στέλνει δεδομένα
- **COLL:** Αναβοσβήνει όταν στο δίκτυο ανιχνεύονται collisions.



Εικόνα 45: Arduino Ethernet Shield rev3

4.3 Cubieboard A10

Μετά το Raspberry Pi έχει ακολουθήσει μια επανάσταση στον χώρο των Development Boards και των mini υπολογιστών που αναπτύσσεται συνεχώς. Μέσα σε όλα αυτά τα boards, το Cubieboard με την χαμηλή τιμή σε σχέση με χαρακτηριστικά του, εξελίσσεται παγκοσμίως στο αντίπαλο δέος του Raspberry Pi 2.



Εικόνα 46: Cubieboard A10

Το Cubieboard βλέπε *εικόνα 46* είναι ένα νέο mini pc όπως και το Raspberry Pi 2 το οποίο χρησιμοποιεί το διαδεδομένο socket Allwinner A10 με επεξεργαστή (CPU) ARM Cortex A8 1Ghz και κάρτα γραφικών (GPU) Mali400 που την συναντάμε στο Rikomagic MK802III.

Έρχεται προ-εγκατεστημένο με το λειτουργικό Android στην εσωτερική μνήμη (4GB) που διαθέτει και μπορεί πολύ εύκολα να εγκατασταθεί Linux σε μια εξωτερική Micro SD κάρτα ή ακόμα και σε έναν σκληρό δίσκο συνδέοντας το στην SATA υποδοχή του Cubieboard.

Επίσης διαθέτει μνήμη RAM 1 GB DDR3 , υποδοχή για υπέρυθρες (IR), USB, Ethernet, HDMI, 96 pins και πολλά άλλα, όλα αυτά σε μέγεθος που δεν ξεπερνά τα 10cm x 6cm x 2cm.

Τεχνικά χαρακτηριστικά Cubieboard

Κατασκευαστής:	CubieTech
Επεξεργαστής:	ARM Cortex-A8 @ 1Ghz
SoC:	Allwinner A10
Κάρτα Γραφικών:	ARM Mali-400
Μνήμη RAM:	DRAM 1GB DDR3 @ 480MHz
Ενσωματωμένη μνήμη:	4GB NAND
Λειτουργικό σύστημα:	Android, Linux
Τροφοδοσία:	Mini-USB ή DC
Έξοδος ήχου:	3.5mm jack, HDMI
Δίκτυο:	10/100 RJ45
Υποδοχές αποθήκευσης:	MicroSDHC, SATA, USB
Άλλες υποδοχές:	Infrared, 2x USB
Διαστάσεις:	10cm x 6cm x 2cm

Πίνακας 3: Τεχνικά χαρακτηριστικά Cubieboard

Επιπλέον υπάρχουν 2*48pins External Headers τα οποία περιλαμβάνουν υποστήριξη για i2c (twi), spi, rgb/lvds, csi/ts, fm-in, adc, cvbs, vga, spdif-out, touch-panel.

4.4 ID Innovations ID-20LA RFID Reader

Το **RFID** είναι τα αρχικά του όρου **Radio Frequency Identification**, η απόδοση του στα ελληνικά ορίζεται ως «ταυτοποίηση μέσω ραδιοσυχνοτήτων». Τα συστήματα RFID αποτελούν ένα υποσύνολο των Συστημάτων Αυτόματου Προσδιορισμού (Automatic Identification Systems). Ειδικότερα λειτουργεί ως γενικός όρος των τεχνολογιών που χρησιμοποιούν ραδιοκύματα για να προσδιορίσουν αυτόματα ανθρώπους ή αντικείμενα και αποτελεί την τεχνολογική εξέλιξη των ραβδωτών κωδίκων (barcode).

Η τεχνολογία RFID είναι γνωστή εδώ και 50 χρόνια. Χρησιμοποιήθηκε για πρώτη φορά από την πολεμική αεροπορία της Αγγλίας κατά τη διάρκεια του Β' Παγκοσμίου, για την αναγνώριση και τη διάκριση των εχθρικών από τα φιλικά αεροπλάνα. Κατά τη διάρκεια των επόμενων δεκαετιών, άρχισε να εδραιώνεται η χρήση και εκμετάλλευσή της. Αρχικά, σε πειραματικό στάδιο και σε εργαστηριακό επίπεδο, για να φτάσουμε στο σήμερα, όπου γίνεται λόγος για εφαρμογή της τεχνολογίας RFID στην καθημερινή ζωή των ανθρώπων, κυρίως μέσω του εμπορίου.



Παράλληλα αναπτύσσεται το ενδεχόμενο της ευρείας εφαρμογής του, με την καθιέρωση προτύπων και την λειτουργία της σε παγκόσμιο επίπεδο.

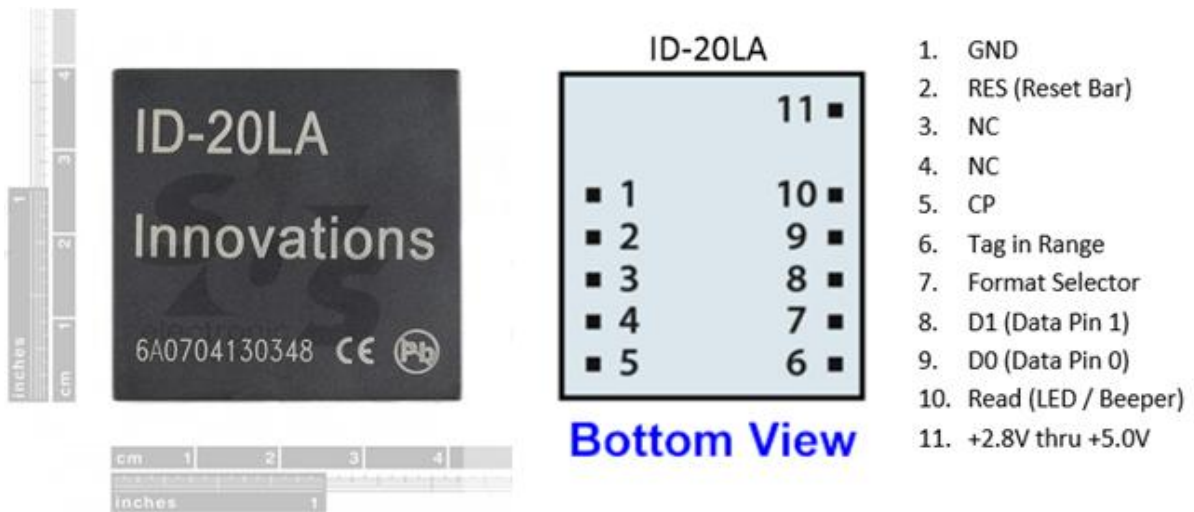
Τα συστήματα RFID απαρτίζονται από δύο κύρια μέρη.

- ✚ Το πρώτο είναι οι πομποδέκτες (transponders) που συχνά αναφέρονται και ως ετικέτες RFID (RFID tags). Οι ετικέτες RFID είναι μικρά chips που αποτελούνται από ένα ολοκληρωμένο κύκλωμα, το οποίο περιλαμβάνει μνήμη ώστε να αποθηκεύει δεδομένα- πληροφορίες, και μία κεραία. Το μέγεθός τους μπορεί να είναι τόσο μικρό όσο το μισό ενός κόκκου άμμου (1/3 του χιλιοστού), ανάλογα με το τύπο τις ετικέτας.
- ✚ Το δεύτερο μέρος είναι οι αναγνώστες ή αισθητήρες (readers), οι οποίοι ανακτούν τα δεδομένα από τις ετικέτες RFID. Οι αναγνώστες RFID έχουν ενσωματωμένα μια κεραία και μια μονάδα ελέγχου.

Η λειτουργία των συστημάτων RFID είναι απλή και βασίζεται στη δυναμική και αμφίδρομη επικοινωνία των ετικετών και των αναγνωστών. Όταν οι ετικέτες RFID βρεθούν στην εμβέλεια της κεραίας του αναγνώστη, η μονάδα ελέγχου επικοινωνεί με ραδιοκύματα με την κεραία των ετικετών RFID. Οι ετικέτες RFID ενεργοποιούνται με τη σειρά τους και επιστρέφουν τα αναζητούμενα δεδομένα στους αναγνώστες.



Στη συνέχεια παρεμβαίνει ένα ενδιάμεσο λογισμικό, το οποίο κατανοεί τις πληροφορίες, οι οποίες αποστέλλονται από τη μονάδα ελέγχου του αναγνώστη και πράττει ανάλογα. Το RFID (αναγνώριση μέσω ραδιοσυχνοτήτων) είναι λοιπόν η χρήση ηλεκτρομαγνητικών πεδίων για τον σκοπό της αναγνώρισης και του εντοπισμού ετικετών (tags) που είναι ενσωματωμένα σε αντικείμενα. Αυτό λοιπόν είναι ο ID-20LA, ένας πολύ απλός στην χρήση RFID reader από την ID Innovations με ενσωματωμένη κεραία και 2mm απόσταση ανάμεσα στα pin (εικόνα 47).



Εικόνα 47: Διαστάσεις και pinout του ID-20LA

Παρακάτω ακολουθεί ο πίνακας τα τεχνικά χαρακτηριστικά του ID-20LA.

Parameter	ID-20LA
Frequency	125 kHz nominal
Card Format	EM 4001 or compatible
Read Range ID3	Up to 30 using suitable antenna using ID-Innovations clamshell card @5v
Read Range ID13	Up to 12cm using ISO card, up to 18cm using ID-Innovations clamshell card @5v
Read Range ID23	Up to 18cm using ISO card, up to 25cm using ID-Innovations clamshell card @5v
Encoding	Manchester 64-bit, modulus 64
Power Requirement	+2.8 VDC thru +5 VDC @ 45mA ID-20LA
RF I/O Output Current	+/- 200mA PKPK

Πίνακας 4: Τεχνικά χαρακτηριστικά ID20LA RFID Reader

Έπειτα στον πίνακα 5 ακολουθεί επεξήγηση του ID-20LA pinout

Pin #	Description	ASCII	Magnet Emulation	Wiegand26
Pin 1	Zero Volts	GND 0V	GND 0V	GND 0V
Pin 2	Strap to Pin11	Reset Bar	Reset Bar	Reset Bar
Pin 3	To External Antenna ID-2LA only	Antenna	Antenna	Antenna
Pin 4	To External Antenna ID-2LA only	Antenna	Antenna	Antenna
Pin 5	Card Present	No function	Card Present*	No function
Pin 6	Tag in Range (Future)	Tag in Range	Tag in Range	Tag in Range
Pin 7	Format Selector (+/-)	Strap to GND	Strap to Pin 10	Strap to +5V
Pin 8	Data 1	CMOS	Clock*	One Output*
Pin 9	Data 0	TTL Data (inverted)	Data*	Zero Output*
Pin 10	3.1 kHz Logic	Beeper / LED	Beeper / LED	Beeper / LED
Pin 11	DC Voltage Supply	+2.8 thru 5V	+2.8V thru 5V	+2.8V thru 5V

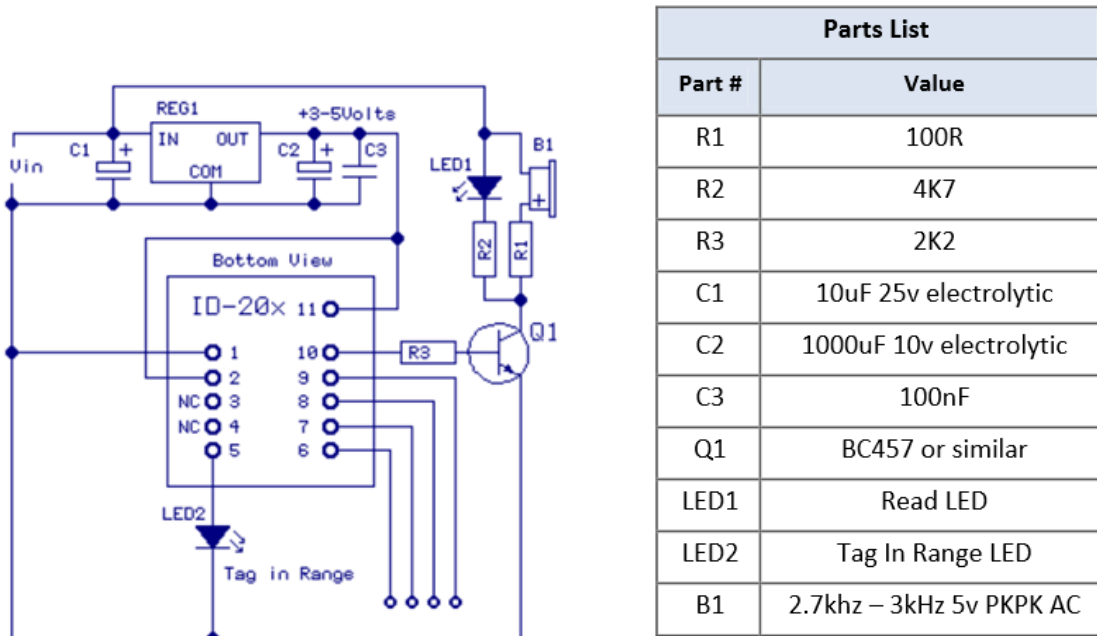
Πίνακας 5: ID20LA RFID Reader Pinout

Αναλυτικότερα το pinout ακολουθεί παρακάτω.

- **Pin1** είναι το GND.
- **Pin2** είναι το Reset και πρέπει να είναι ΠΑΝΤΑ συνδεδεμένο με το pin11.
- **Pin3** είναι συνδεδεμένο μόνο στον ID-2LA και δίνει την δυνατότητα να συνδέσεις εξωτερική κεραία 1.337mH. Στους ID-12LA / 20LA modules το pin3 πρέπει να μείνει κενό.
- **Pin4** είναι συνδεδεμένο μόνο στον ID-2LA και δίνει την δυνατότητα να συνδέσεις εξωτερική κεραία 1.337mH. Στους ID-12LA / 20LA modules το pin3 πρέπει να μείνει κενό.
- **Pin5** χρησιμοποιείται ως ένδειξη 'Card Present' όταν η έξοδος έχει ρυθμιστεί σε Magnetic Emulation.
- **Pin6** χρησιμοποιείται ως 'Tag in Range' ένδειξη. Όταν ένα tag βρίσκεται σε κοντινή απόσταση το pin πάει στην VDD τάση, αλλιώς στο GND. Η έξοδος του Pin6 έχει μια εσωτερική αντίσταση 3K3 & μπορεί να χρησιμοποιηθεί για να οδηγήσει απευθείας ένα LED.
- **Pin7** είναι το format selector. Ο format selector επιλέγει το format ανάλογα με το τι είναι συνδεδεμένο και επιλέγεται στο switch on και δεν μπορεί να αλλαχτεί αργότερα.

- **Pin8:** Όταν το output format είναι ρυθμισμένο σε Magnetic Emulation το pin8 χρησιμοποιείται για έξοδο ‘Clock’.
- **Pin9** μπορεί να χρησιμοποιηθεί για διασύνδεση με UART.
- **Pin10** έχει ένα beeper όπου όταν είναι ενεργό το pin10 φέρνει μια έξοδο τετραγωνικού παλμού των 3.3KHz.
- **Pin11** είναι η τροφοδοσία DC +2.8V έως 5V.

Το κυκλωματικό διάγραμμα του ID20LA ακολουθεί στην *εικόνα 48*.



Εικόνα 48: Circuit Diagram for ID-20LA

4.5 Linksys Ethernet Switch

Για την διασύνδεση όλων (Cubieboard server, arduino Ethernet microcontroller, pc του administrator για έλεγχο του συστήματος κλπ) χρησιμοποιήθηκε το παρακάτω Ethernet switch με λειτουργία Power Over Ethernet (*Εικόνα 49*). Επιπλέον έχει 5 πόρτες gigabit από τις οποίες χρησιμοποιήσαμε τις 3.



Εικόνα 49: Linksys switch

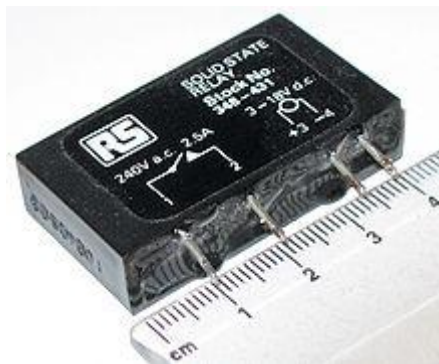
4.6 DC Relay

Ο **ηλεκτρονόμος, ρελέ (relay)** ή **ρελές** είναι ένας ηλεκτρικός διακόπτης που ανοίγει και κλείνει ένα ηλεκτρικό κύκλωμα κάτω από τον έλεγχο ενός άλλου ηλεκτρικού κυκλώματος. Στην αρχική μορφή του, ένας ηλεκτρομαγνήτης ενεργοποιούσε το διακόπτη, με το άνοιγμα ή κλείσιμο μιας ή περισσότερων επαφών. Εφευρέθηκε από τον Τζόζεφ Χένρυ το 1835. Επειδή ένας ηλεκτρονόμος είναι ικανός να ελέγχει ένα κύκλωμα εξόδου υψηλότερης ισχύος από το κύκλωμα εισόδου, μπορεί να θεωρηθεί μια μορφή ηλεκτρικού ενισχυτή.

Κάθε επαφή ενός ηλεκτρονόμου μπορεί να είναι *Κανονικά-Ανοικτή (Normally Open, NO)*, *Κανονικά-Κλειστή' (Normally Closed, NC)* ή *μεταγωγικός (change-over)*, ανάλογα με τον τύπο της.

- Μια επαφή **Κανονικά-Ανοικτή** συνδέει το κύκλωμα όταν ο ηλεκτρονόμος ενεργοποιείται· το κύκλωμα αποσυνδέεται όταν ο ηλεκτρονόμος είναι ανενεργός. Μια τέτοια επαφή καλείται επίσης *Επαφή Μορφής Α* ή *επαφή "make"*. Η επαφή μορφής Α είναι ιδανική για εφαρμογές που απαιτούν την ενεργοποίηση μιας πηγής υψηλής τάσης από απόσταση.
- Μια επαφή **Κανονικά-Κλειστή** αποσυνδέει το κύκλωμα όταν ο ηλεκτρονόμος ενεργοποιείται· το κύκλωμα συνδέεται όταν ο ηλεκτρονόμος είναι ανενεργός. Μια τέτοια επαφή καλείται επίσης *Επαφή Μορφής Β* ή *επαφή "break"*. Η επαφή μορφής Β είναι ιδανική για εφαρμογές που απαιτούν το κύκλωμα να παραμένει κλειστό (ενεργό) μέχρι ο ηλεκτρονόμος να ενεργοποιηθεί.
- Μια επαφή **Μεταγωγική** μπορεί να ελέγχει δύο κυκλώματα. Ισοδυναμεί με μια επαφή κανονικά-ανοικτή και μια επαφή κανονικά-κλειστή που έχουν ένα κοινό ακροδέκτη. Μια τέτοια επαφή καλείται επίσης *Επαφή Μορφής C*.

Συνήθως ένας ηλεκτρονόμος αποτελείται από περισσότερες από μία ελεγχόμενες επαφές. Οι επαφές χωρίζονται σε κύριες και βοηθητικές. Οι κύριες διαρρέονται συχνά από ισχυρότερα ρεύματα και έτσι είναι αυτές που διακόπτουν το κύριο κύκλωμα και συνήθως είναι **Κανονικά-Ανοικτές**. Οι βοηθητικές έχουν όπως υπονοεί και το όνομά τους επικουρικό χαρακτήρα και ο ρόλος τους είναι να βοηθούν στον έλεγχο των αυτοματισμών (που είναι ο κύριος τομέας χρήσης των ηλεκτρονόμων). Για παράδειγμα βοηθούν στην ενεργοποίηση/απενεργοποίηση βοηθητικών κυκλωμάτων όπως ενδεικτικές λυχνίες.



Εικόνα 50: Ηλεκτρονόμος στερεάς κατάστασης χωρίς κινούμενα μέρη.

Όταν ηλεκτρικό ρεύμα διαρρέει το πηνίο του ηλεκτρονόμου, το παραγόμενο μαγνητικό πεδίο έλκει έναν οπλισμό που είναι μηχανικά συνδεδεμένος σε μια κινούμενη επαφή. Έτσι, η κινούμενη επαφή είτε συνδέεται με μια σταθερή επαφή είτε αποσυνδέεται από τη σταθερή επαφή. Μόλις το ηλεκτρικό ρεύμα στο πηνίο διακοπεί, ο οπλισμός επιστρέφει στη θέση ηρεμίας του εξαιτίας μιας δύναμης επαναφοράς, που είναι ίση με το ήμισυ της μαγνητικής. Η δύναμη επαναφοράς παρέχεται συνήθως από ένα ελατήριο, αλλά και η βαρύτητα χρησιμοποιείται συχνά σε βιομηχανικούς εκκινητές μηχανών. Η μεταβολή της μαγνητικής ροής στο πηνίο γεννά ένα ηλεκτρικό ρεύμα, το λεγόμενο "επαγωγικό", που έχει αντίθετη φορά από εκείνο που παρέχεται στο πηνίο. Για τη λειτουργία του πηνίου και τη μετακίνηση των επαφών απαιτείται σχετικά μεγάλη ένταση ηλεκτρικού ρεύματος, αλλά - μόλις ο οπλισμός κλείσει - το ηλεκτρικό ρεύμα που απαιτείται για να κρατήσει τον οπλισμό κλειστό είναι ένα μικρό κλάσμα του αρχικού, τυπικά το $1/10$. Οι ηλεκτρονόμοι κατασκευάζονται για να λειτουργούν γρήγορα. Σε μια εφαρμογή χαμηλής τάσης, αυτό γίνεται για τη μείωση του θορύβου. Σε μια εφαρμογή υψηλής τάσης ή υψηλής έντασης ρεύματος, αυτό γίνεται για τη μείωση των σπινθηρισμών (ηλεκτρικών εκφορτίσεων μορφής τόξου).

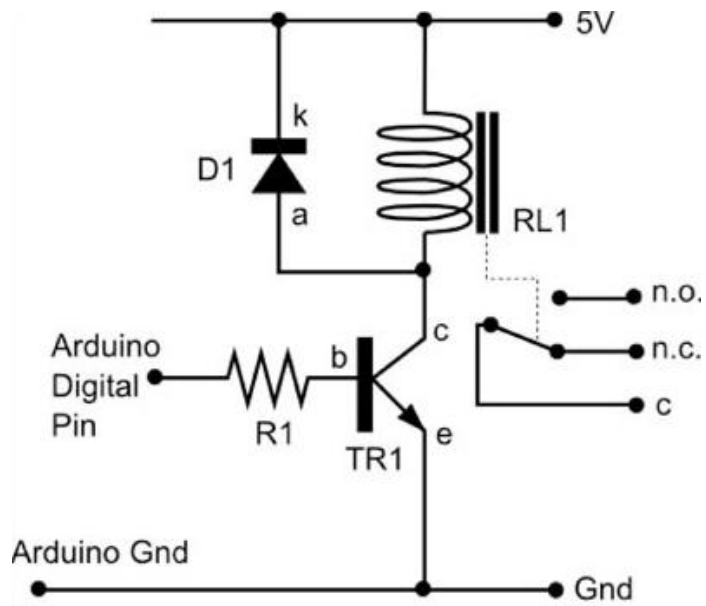
Εάν το πηνίο διεγείρεται με συνεχές (DC) ρεύμα, ανεξάρτητα από το ηλεκτρικό ρεύμα που ρέει διαμέσου των επαφών, μια δίοδος μπαίνει συνήθως παράλληλα με το πηνίο. Όταν το πηνίο διεγείρεται, αποκαθίσταται ένα μαγνητικό πεδίο. Όταν το πηνίο αποδιεγείρεται, το καταρρέον μαγνητικό πεδίο δημιουργεί μια αιχμή ηλεκτρικού ρεύματος που θα μπορούσε να βλάψει το υπόλοιπο κύκλωμα. Αν το πηνίο διεγείρεται με εναλλασσόμενο (AC) ρεύμα, ένα μικρό χάλκινο δαχτυλίδι πτυχώνεται στο άκρο του σωληνοειδούς πηνίου. Το AC μηδενίζεται 100 φορές το δευτερόλεπτο και σε κάθε χρονική στιγμή μηδενισμού δεν υπάρχει καμιά μαγνητική δύναμη που να συγκρατεί τις επαφές κλειστές. Το μικρό χάλκινο δαχτυλίδι παρέχει ένα μικρό ρεύμα εκτός φάσεως που καλείται *shadow pole* (σκιάδης πόλος). Το άθροισμα του εναλλασσόμενου ρεύματος και του *shadow pole* εξασφαλίζει τη συγκράτηση του οπλισμού στη θέση εμπλοκής σε όλες τις χρονικές στιγμές.

Για την συγκεκριμένη εφαρμογή με διέγερση 5V DC επιλέχθηκε έτοιμο 2 Relay Module όπως εμφανίζεται στην παρακάτω εικόνα 51 και όχι ένας απλός DC ρελέ για καλύτερη προστασία του μικροελεγκτή arduino.



Εικόνα 51: 2 Relay Module

Η εναλλακτική θα ήταν να κατασκευαστεί ένα κύκλωμα όπου το ρελέ να οδηγείται από ένα διπολικό τρανζίστορ όπως φαίνεται παρακάτω στην *εικόνα 52*.



Εικόνα 52: Κύκλωμα οδήγησης ρελέ

4.7 Ηλεκτρική Κλειδαριά (Electric Strike)

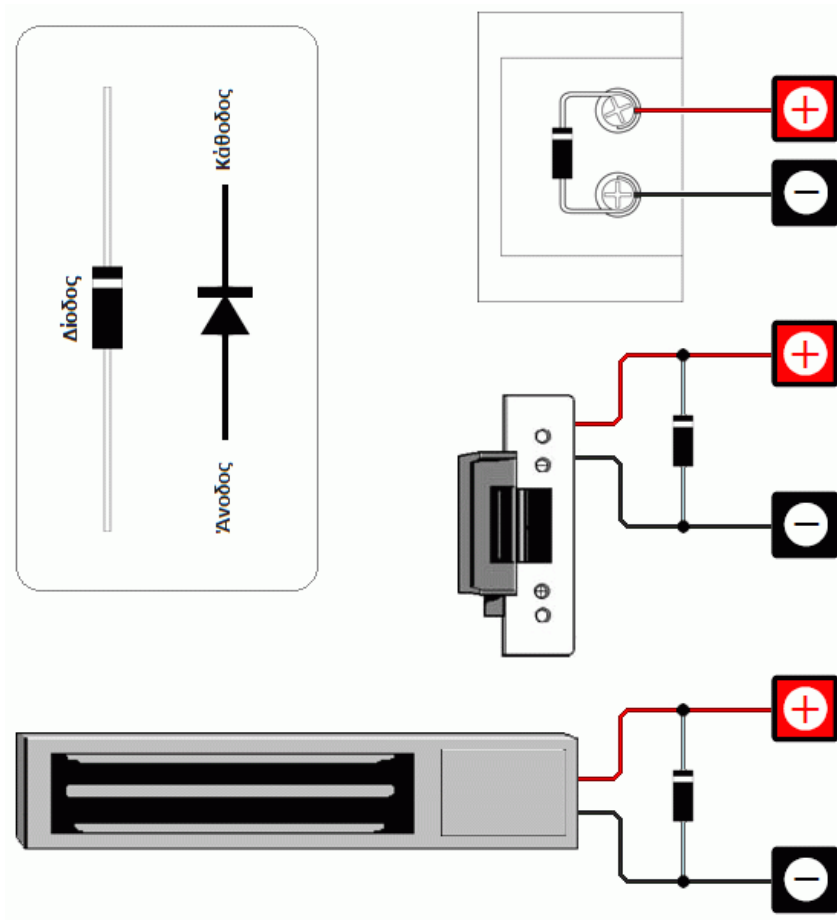
Τα ηλεκτρικά αντικρίσματα (**ηλεκτρικό κυπρί**) είναι ηλεκτρικές κλειδαριές τις οποίες συναντάμε συνήθως στις εισόδους οικοδομών, σε γραφεία, ιατρεία κτλ.

Στην συγκεκριμένη υλοποίηση όταν ο μικροελεγκτής δώσει την εντολή στον ρελέ να οπλίσει τότε περνάει ρεύμα χαμηλής συνεχής τάσης **12V** στον ηλεκτρομαγνήτη που απασφαλίζει την **ηλεκτρική κλειδαριά** ώστε να επιτραπεί η είσοδος, όπως εμφανίζεται στην *εικόνα 53*.



Εικόνα 53: Ηλεκτρική κλειδαριά

Η ηλεκτρική κλειδαριά εμφανίζεται αναλυτικότερα παρακάτω στην εικόνα 54.



Εικόνα 54: Ηλεκτρική Κλειδαριά

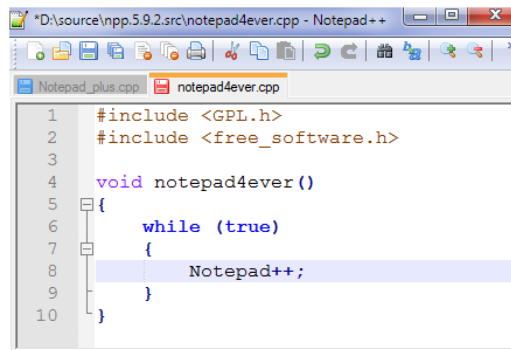
5. Software (Λογισμικό)

Στην ενότητα αυτή γίνεται αναφορά στην πλατφόρμα ανάπτυξης λογισμικού που χρησιμοποιήθηκε.

- ✚ Για τον server του συστήματος ελέγχου πρόσβασης χρησιμοποιήθηκε το Cubieboard στο οποίο εγκαταστάθηκε το λειτουργικό σύστημα Cubian X1 (Debian based) για ARM επεξεργαστές.
- ✚ Για το προγραμματισμό και την κατασκευή του ιστοχώρου έγινε εγκατάσταση του τοπικού διακομιστή LAMP και phpMyAdmin όλα ανοιχτού λογισμικού και μέσω του Packet Management System της συγκεκριμένης διανομής.
- ✚ Για το προγραμματισμό του Arduino Mega χρησιμοποιήθηκε το Arduino IDE και για την συγγραφή του κώδικα το Notepad++. Αντίστοιχα για την επεξεργασία εικόνων το GIMP, ενώ για την κατασκευή των απεικονιζομένων εικόνων των κυκλωμάτων που χρησιμοποιήθηκαν το Fritzing.

5.1 Notepad ++

Πρόκειται για πρόγραμμα ανοιχτού λογισμικού (εικόνα 55), όπου χρησιμοποιείται για την ανάπτυξη script σε αρκετές γλώσσες προγραμματισμού πχ php, html, mysql, css κλπ.



```
*D:\source\hpp.5.9.2.src\notepad4ever.cpp - Notepad++
Notepad_plus.cpp notepad4ever.cpp
1 #include <GPL.h>
2 #include <free_software.h>
3
4 void notepad4ever()
5 {
6     while (true)
7     {
8         Notepad++;
9     }
10 }
```







Εικόνα 55: Notepad ++

5.2 Arduino IDE

Το ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) του Arduino είναι μία εφαρμογή γραμμένη σε Java, που λειτουργεί σε πολλές πλατφόρμες, και προέρχεται από το IDE για τη γλώσσα προγραμματισμού Processing και το σχέδιο Wiring. Έχει σχεδιαστεί για να εισαγάγει τον προγραμματισμό σε όσους δεν είναι ιδιαίτερα εξοικειωμένοι με την ανάπτυξη λογισμικού. Περιλαμβάνει ένα πρόγραμμα επεξεργασίας κώδικα με χαρακτηριστικά όπως είναι η επισήμανση σύνταξης και ο συνδυασμός αγκύλων και είναι επίσης σε θέση να μεταγλωττίζει και να φορτώνει προγράμματα στην πλακέτα με ένα μόνο κλικ. Δεν υπάρχει συνήθως καμία ανάγκη να επεξεργαστούμε αρχεία make ή να τρέξουμε προγράμματα σε ένα περιβάλλον

γραμμής εντολών. Ένα πρόγραμμα ή κώδικας που γράφτηκε για Arduino ονομάζεται σκίτσο (sketch). Τα Arduino προγράμματα είναι γραμμένα σε C ή C++.

Στον πίνακα 6 παρουσιάζονται τα εργαλεία του περιβάλλοντος ανάπτυξης υπό μορφή κουμπιών.

Εργαλείο	Περιγραφή
 Verify	Ελέγχει για συντακτικά λάθη στον κώδικα.
 Upload	Μεταγλωττίζει τον κώδικα και τον φορτώνει στο Arduino. Αν δεν είναι συντακτικά σωστός δεν μπορεί να γίνει η φόρτωση.
 New	Δημιουργεί ένα νέο sketch.
 Open	Παραθέτει ένα menu με όλα τα sketch. Ενεργοποιώντας ένα από αυτά, θα ανοίξει αυτόματα στο τρέχον παράθυρο.
 Save	Αποθηκεύει ένα sketch.
 Serial Monitor	Ανοίγει την σειριακή οθόνη και μέσω αυτής παρακολουθείται η ανταλλαγή δεδομένων που γίνεται στην σειριακή θύρα.

Πίνακας 6: Εργαλεία περιβάλλοντος ανάπτυξης

Το Arduino IDE έρχεται με μια βιβλιοθήκη λογισμικού που ονομάζεται "Wiring" από το πρωτότυπο σχέδιο Wiring γεγονός που καθιστά πολλές κοινές λειτουργίες εισόδου/εξόδου πολύ πιο εύκολες.

Συνδέεται με το hardware μέρος του Arduino για να φορτώσει προγράμματα και να επικοινωνεί μαζί τους.

Ο χρήστης για να μπορέσει να φτιάξει ένα πρόγραμμα κυκλικής εκτέλεσης αρκεί μόνο να ορίσει δύο λειτουργίες:

- **setup()**: μία συνάρτηση που τρέχει μία φορά στην αρχή του προγράμματος η οποία αρχικοποιεί τις ρυθμίσεις
- **loop()**: μία συνάρτηση η οποία καλείται συνέχεια μέχρι η πλακέτα να απενεργοποιηθεί

Ένα τυπικό πρώτο πρόγραμμα για έναν μικροελεγκτή αναβοσβήνει απλά ένα LED. Στο περιβάλλον του Arduino, ο χρήστης μπορεί να γράψει ένα πρόγραμμα σαν αυτό:

```
#define LED_PIN 13

void setup () {
  pinMode (LED_PIN, OUTPUT); // enable pin 13 for digital output
}

void loop () {
  digitalWrite (LED_PIN, HIGH); // turn on the LED
  delay (1000); // wait one second (1000 milliseconds)
  digitalWrite (LED_PIN, LOW); // turn off the LED
  delay (1000); // wait one second
}
```



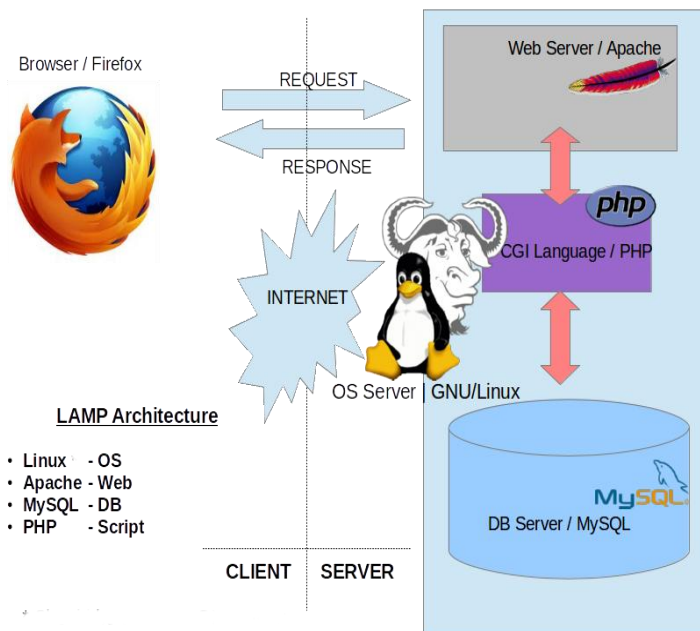
Εικόνα 56: Στιγμιότυπο λογισμικού Arduino

Είναι ένα χαρακτηριστικό των περισσότερων πλακετών Arduino ότι έχουν ένα LED και μία αντίσταση φορτίου που συνδέονται μεταξύ του pin 13 και GND, ένα βολικό χαρακτηριστικό για πολλά απλά τεστ.

Ο προηγούμενος κώδικας δεν θα αναγνωριστεί από ένα κανονικό μεταγλωττιστή C++ ως έγκυρο πρόγραμμα και ο χρήστης πρέπει να κάνει κλικ στο κουμπί "Upload to I/O board" του IDE και ένα αντίγραφο του κώδικα θα γραφτεί σε ένα προσωρινό αρχείο με ένα παραπάνω include στην κορυφή και μία πολύ απλή συνάρτηση main() στο τέλος, ώστε να φτιαχτεί πλέον ένα έγκυρο C++ πρόγραμμα.

Το IDE του Arduino χρησιμοποιεί το GNU toolchain και το AVR Libc για να μεταγλωττίζει προγράμματα και το avrdude για να φορτώνει προγράμματα στην πλακέτα. Δεδομένου ότι η πλατφόρμα Arduino χρησιμοποιεί Atmel μικροελεγκτές, το περιβάλλον ανάπτυξης της Atmel, το AVR Studio ή το νεότερη έκδοση του Atmel Studio, μπορεί επίσης να χρησιμοποιηθεί για την ανάπτυξη λογισμικού για το Arduino.

5.3 Διακομιστής LAMP



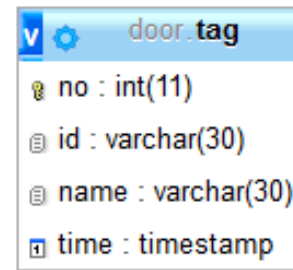
Ο τοπικός διακομιστής LAMP χρησιμοποιήθηκε για την ανάπτυξη και την διασύνδεση με την βάση ενώ πρόκειται για ένα σύνολο προγραμμάτων ανοικτού λογισμικού που λειτουργεί ως διακομιστής διαδικτύου. Το LAMP (Linux, Apache, MySQL, PHP) είναι ένα σύνολο από open source τεχνολογίες όπου μας παρέχουν περιβάλλον για Web Development με Linux (Cubian) ως λειτουργικό σύστημα, Apache για Web Server, MySQL για σύστημα διαχείρισης βάσης δεδομένων και PHP ως scripting γλώσσα για τις εφαρμογές μας. Επιπλέον χρησιμοποιήθηκε το phpMyAdmin για απομακρυσμένη διαχείριση της MySQL.

Εικόνα 57: LAMP

Η **MySQL** είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων που έλαβε το όνομά της από την κόρη του Μόντυ Βιντένιους, τη Μάι (αγγλ. My). Το πρόγραμμα τρέχει έναν εξυπηρετητή (server) παρέχοντας πρόσβαση πολλών χρηστών σε ένα σύνολο βάσεων δεδομένων. Ο κωδικός του εγχειρήματος είναι διαθέσιμος μέσω της GNU General Public License, καθώς και μέσω ορισμένων ιδιοκτητών συμφωνιών. Ανήκει και χρηματοδοτείται από μία και μοναδική κερδοσκοπική εταιρία, τη σουηδική MySQL AB, η οποία σήμερα ανήκει στην Oracle. Η MySQL είναι δημοφιλής βάση δεδομένων για διαδικτυακά προγράμματα και ιστοσελίδες. Χρησιμοποιείται σε κάποιες από τις πιο διαδεδομένες διαδικτυακές υπηρεσίες, όπως το Flickr, YouTube, Wikipedia, Google, Facebook.

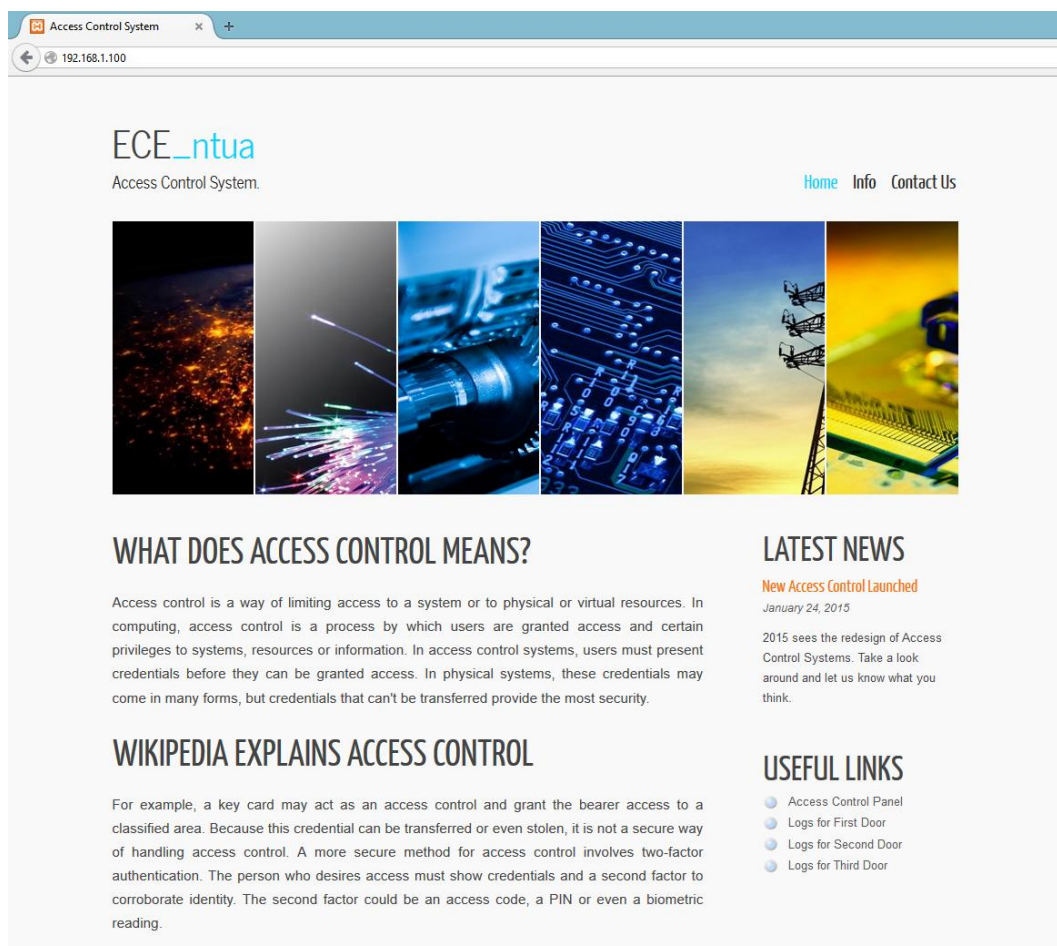
Η MySQL είναι μια πολύ γρήγορη σε απόδοση και ισχυρή σε δυνατότητες υλοποίηση ενός συστήματος διαχείρισης σχεσιακών βάσεων δεδομένων. Τα δεδομένα αποθηκεύονται σε αντικείμενα βάσης δεδομένων, τα οποία ονομάζεται πίνακες. Ένας πίνακας, είναι μια συλλογή από σχετικές καταχωρήσεις δεδομένων και αποτελείται από στήλες (columns) και γραμμές (rows). Οι βάσεις δεδομένων είναι χρήσιμες για την αποθήκευση πληροφοριών σε κατηγορίες. Η διαχείριση των δεδομένων στις βάσεις, γίνονται μέσω των SQL εντολών. Η πιο σημαντική ίσως κατηγορία εντολών που χρησιμοποιείται είναι τα queries. Ένα query είναι ένα ερώτημα ή ένα αίτημα. Με τη MySQL επιτρέπεται η διερεύνηση σε μια βάση δεδομένων και η επιστροφή ζητηθέντων πληροφοριών.

Για την συγκεκριμένη λοιπόν υλοποίηση δημιουργήθηκε μια βάση δεδομένων **door** ένα table **tag** και 4 column: **no** (ένας αύξων αριθμός auto increment), **id** (εδώ είναι αποθηκευμένα τα rfid tags που έχουν πρόσβαση), **name** (εδώ είναι αποθηκευμένα τα ονόματα που αντιστοιχούν στα rfid της στήλης id), **time** (η ημερομηνία που γίνεται η εισαγωγή ενός χρήστη στην βάση) όπως αυτά φαίνονται στην *Εικόνα 58*.



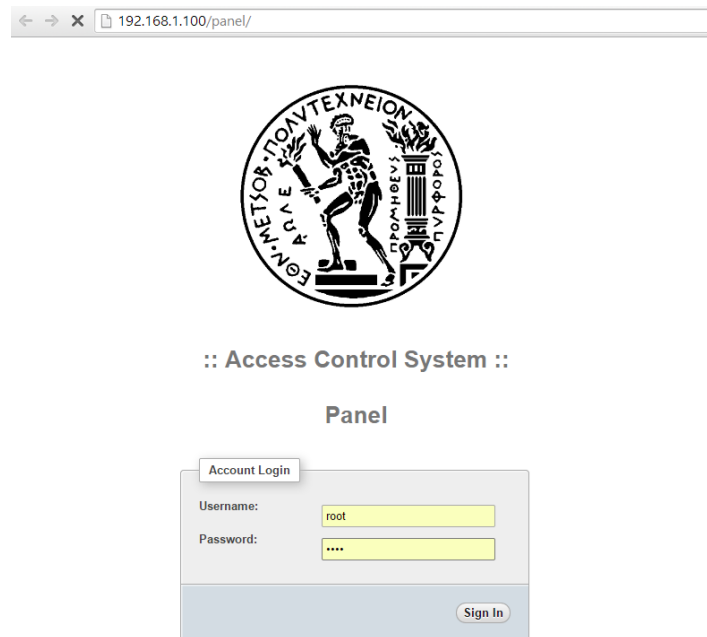
Εικόνα 58: MySQL

Για την ευκολότερη πρόσβαση στο σύστημα διαχείρισης της βάσης δεδομένων και των access logs της κάθε πόρτα στήθηκε σε html το παρακάτω front panel (*εικόνα 59*). Ο χρήστης πληκτρολογώντας την ip του Cubieboard Server (192.168.1.100) μπαίνει στην κεντρική σελίδα όπου μπορεί να διαβάσει αρκετές χρήσιμες πληροφορίες για το σύστημα αλλά και από τα δεξιά στην κατηγορία USEFUL LINKS να αποκτήσει πρόσβαση είτε στο σύστημα διαχείρισης της βάσης δεδομένων (Access Control Panel) όπου μπορεί να δει, να προσθέσει ή να αφαιρέσει την πρόσβαση σε χρήστες είτε να δει τα logs από όλα τα σημεία ελεγχόμενης πρόσβασης (πχ Logs for First Door κλπ).



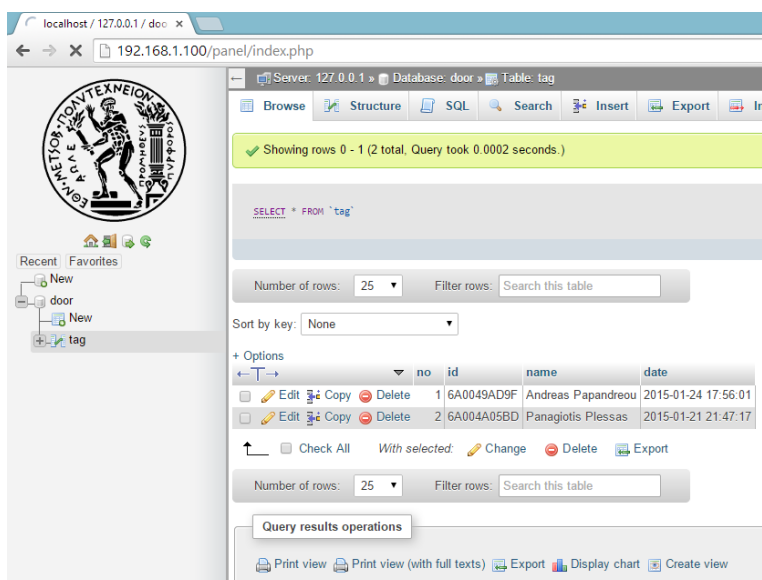
Εικόνα 59: Front Panel του Access Control System

Επιλέγοντας λοιπόν ο χρήστης από τα δεξιά στο μενού των Useful Links το Access Control Panel η σελίδα τον μεταφέρει σε ένα νέο περιβάλλον όπου βάζοντας τα κατάλληλα στοιχεία Username και Password μπορεί να αποκτήσει πρόσβαση στην βάση δεδομένων (Εικόνα 60).



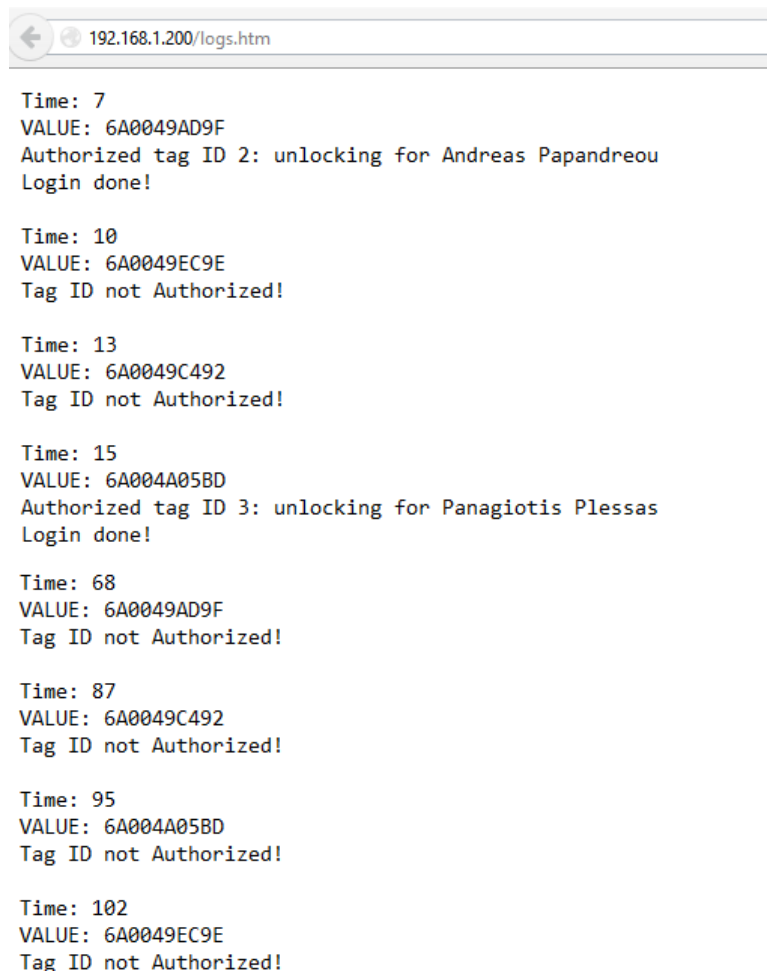
Εικόνα 60: Outside MySQL Panel

Έπειτα στην Εικόνα 61 μπορεί κάποιος να κάνει περιήγηση στην βάση δεδομένων και πολύ εύκολα να δει ποιοι χρήστες έχουν πρόσβαση στην βάση και να τους τροποποιήσει. Όλες οι αλλαγές στην βάση δεδομένων είναι πραγματικού χρόνου και έχουν **ΑΜΕΣΗ** εφαρμογή στην δυνατότητα πρόσβασης.



Εικόνα 61: Inside MySQL Panel

Επιπλέον το arduino mega, όπου βρίσκεται εγκατεστημένο σε κάθε σημείο ελεγχόμενης πρόσβασης, εκτός από client έχει ρυθμιστεί να δουλεύει αυτόνομα και ως web server (http) στην μοναδική ip που του έχει δοθεί πχ 192.168.1.200 και το default port 80. Η λειτουργία αυτή του δόθηκε ώστε μέσω κατάλληλου προγραμματισμού να μπορεί κάποιος απομακρυσμένα (λόγου χάρι ο διαχειριστής του συστήματος) να δει τα logs με όσους έχουν χρησιμοποιήσει το συγκεκριμένο σημείο πρόσβασης. Όσον αφορά την αποτύπωση του χρόνου στην συγκεκριμένη υλοποίηση για συντομία χρησιμοποιήθηκε απλά ένας timer που ξεκινάει να μετράει το χρόνο σε δευτερόλεπτα από την στιγμή που κάνει boot το arduino. Σε κανονική υλοποίηση θα πρέπει να αντικατασταθεί ο software timer με αντίστοιχο κύκλωμα RTC (Real Time Clock) όπου θα ρυθμιστεί ημερομηνία και ώρα για να καταγράφεται έπειτα στα logs. Πρόσβαση στα LOGS του κάθε σημείου ελεγχόμενης πρόσβασης μπορεί να έχει ο διαχειριστής από το κεντρικό front panel (βλέπε *Εικόνα 59*) δεξιά στα USEFUL LINKS αφού επιλέξει πχ Logs for First Door όπου τον κάνει redirect στον http server που έχει σηκωθεί στο arduino mega του συγκεκριμένου σημείου ελεγχόμενης πρόσβασης. Έπειτα παρακάτω (βλέπε *Εικόνα 62*) εμφανίζονται τα LOGS όπως καταγράφηκαν κατά την διάρκεια λειτουργικών δοκιμών του συστήματος.



```
Time: 7
VALUE: 6A0049AD9F
Authorized tag ID 2: unlocking for Andreas Papandreou
Login done!

Time: 10
VALUE: 6A0049EC9E
Tag ID not Authorized!

Time: 13
VALUE: 6A0049C492
Tag ID not Authorized!

Time: 15
VALUE: 6A004A05BD
Authorized tag ID 3: unlocking for Panagiotis Plessas
Login done!

Time: 68
VALUE: 6A0049AD9F
Tag ID not Authorized!

Time: 87
VALUE: 6A0049C492
Tag ID not Authorized!

Time: 95
VALUE: 6A004A05BD
Tag ID not Authorized!

Time: 102
VALUE: 6A0049EC9E
Tag ID not Authorized!
```

Εικόνα 62: LOGS του σημείου ελεγχόμενης πρόσβασης που έχει επιλεγεί

6. Προγραμματισμός του μικροελεγκτή της Atmel

Στις επόμενες σελίδες θα ακολουθήσει το κυρίως κομμάτι της διπλωματικής που αποτελεί την υλοποίηση του κώδικα που θα γίνει compile στον μικροελεγκτή της Atmel, καθώς και ο λεπτομερέστατος σχολιασμός του. Ο κώδικας βρίσκεται ολοκληρωμένος στο παράρτημα.

Στις επόμενες σελίδες λοιπόν θα γίνει επεξήγηση γραμμή προς γραμμή του κώδικα ο οποίος έχει γίνει upload στο Arduino Mega, καθώς και οτιδήποτε κρίνεται αναγκαίο να επεξηγηθεί περισσότερο σχετικά με την υλοποίηση αυτού.

```
1 #include <SoftwareSerial.h>
```

Το hardware του Arduino έχει ενσωματωμένη υποστήριξη για σειριακή επικοινωνία διαμέσου των pins 0 και 1 που όμως επίσης χρησιμοποιούνται και για την επικοινωνία του Arduino με τον υπολογιστή μέσω USB σύνδεσης. Για αυτό το σκοπό δημιουργήθηκε η βιβλιοθήκη SoftwareSerial ώστε να επιτρέψει η σειριακή επικοινωνία να γίνεται σε άλλα ψηφιακά pins του Arduino χρησιμοποιώντας software για να αντιγράψει την λειτουργία αυτή εξού και το όνομα Software Serial. Επιπλέον είναι δυνατόν να έχεις πολλαπλά software serial ports με ταχύτητες έως 115200 bps. Όσον αφορά το Mega οι περιορισμοί της βιβλιοθήκης επιτρέπουν change interrupts μόνο στα παρακάτω pins: RX 10, 11, 12, 13, 14, 15, 50, 51, 52, 53, A8 (62), A9 (63), A10 (64), A11 (65), A12 (66), A13 (67), A14 (68), A15 (69).

```
2 #include <Ethernet.h>
```

Αυτή η βιβλιοθήκη επιτρέπει στο Arduino board μέσω του Ethernet Shield να αποκτήσει δικτυακές δυνατότητες. Μπορεί να λειτουργήσει σαν server δεχόμενος εισερχόμενες συνδέσεις ή ως client δημιουργώντας εξερχόμενες. Η βιβλιοθήκη υποστηρίζει μέχρι 4 ταυτόχρονες συνδέσεις (εισερχόμενες ή εξερχόμενες ή συνδυασμός αυτών). Το Arduino επικοινωνεί με το Ethernet Shield μέσω του SPI bus. Αυτό στο Arduino Mega γίνεται μέσω των pins 50, 51, 52 και του pin 10 ως Slave Select (SS). Επιπλέον το hardware SS pin 53 πρέπει στο κώδικα μας να είναι δηλωμένο πάντα ως OUTPUT διαφορετικά δεν θα μπορεί να δουλέψει σωστά το SPI interface.

```
3 #include <SPI.h>
```

Η βιβλιοθήκη SPI επιτρέπει την επικοινωνία του Arduino ως master συσκευή με άλλες SPI συσκευές. Το Serial Peripheral Interface (SPI) είναι ένα σύγχρονο πρωτόκολλο σειριακών δεδομένων το οποίο χρησιμοποιείται για την επικοινωνία μεταξύ μικροελεγκτών και άλλων περιφερειακών συσκευών σε μικρές αποστάσεις. Σε μια σύνδεση SPI πάντα υπάρχει μια master συσκευή (συνήθως ένας μικροελεγκτής) που ελέγχει περιφερειακές συσκευές.

Τυπικά υπάρχουν 3 γραμμές κοινές σε όλες τις συσκευές της SPI σύνδεσης:

- **MISO** (Master In Slave Out) - Η γραμμή Slave για αποστολή δεδομένων στον master,

- **MOSI** (Master Out Slave In) – Η γραμμή Master για αποστολή δεδομένων στις περιφερειακές συσκευές,
- **SCK** (Serial Clock) – Ο παλμός ρολογιού για το συγχρονισμό της μετάδοσης δεδομένων από την master συσκευή,

Και άλλη μια γραμμή συγκεκριμένη για κάθε μια συσκευή:

- **SS** (Slave Select) – Το pin σε κάθε συσκευή το οποίο χρησιμοποιεί ο master ώστε να ενεργοποιεί ή να απενεργοποιεί τις συσκευές αυτές.

Όταν σε μια συσκευή το Slave Select (SS) pin είναι low αυτή η συσκευή επικοινωνεί με την master συσκευή, ενώ όταν είναι high την αγνοεί. Αυτό σου επιτρέπει να έχεις πολλαπλές SPI συσκευές μοιραζόμενες τις ίδιες MISO, MOSI και SLK γραμμές.

```
4 #include <SD.h>
```

Η βιβλιοθήκη SD επιτρέπει την ανάγνωση και εγγραφή σε SD κάρτες όπως αυτή που υπάρχει στο Ethernet Shield. Η βιβλιοθήκη υποστηρίζει FAT16 και FAT32 file systems και SD ή SDHC κάρτες. Από την version 1.0 και μετά η βιβλιοθήκη υποστηρίζει το άνοιγμα πολλαπλών αρχείων μαζί. Η επικοινωνία μεταξύ του μικροεπεξεργαστή και της SD κάρτας χρησιμοποιεί το SPI bus το οποίο χρειάζεται τα pins 50, 51, 52 στο Mega. Επιπλέον πρέπει να χρησιμοποιηθεί ένα ακόμα pin για την επιλογή της SD κάρτας. Αυτό θα μπορούσε να είναι είτε το hardware SS pin 53 ή ένα οποιοδήποτε άλλο ορισμένο στην SD.begin(). Όπως και να έχει είτε επιλέξουμε να χρησιμοποιήσουμε ή όχι το SS pin 53 πρέπει να οριστεί ως OUTPUT στον κώδικα μας διαφορετικά η βιβλιοθήκη SD δεν θα δουλέψει σωστά.

```
5 #include <Keypad.h>
```

Η βιβλιοθήκη Keypad χρησιμοποιείται για keypads στο Arduino με σχήμα πίνακα πχ 3x3, 3x4, 4x4 κλπ. Δεν χρειάζονται εξωτερικές αντιστάσεις ή διόδους διότι η βιβλιοθήκη χρησιμοποιεί εσωτερικές pull-up αντιστάσεις και όλα τα χρησιμοποιήσιμα column pins έχουν high-impedance.

```
6 #include <Wire.h>
```

Η βιβλιοθήκη Wire επιτρέπει την επικοινωνία με I2C / TWI συσκευές. Στα Arduino boards που έχουν το rev3 layout το pin 20 SDA (data line) και το pin 21 SCL (clock line) βρίσκονται δίπλα στο AREF pin.

```
7 #include <sha1.h>
```

Για να υλοποιηθεί παρακάτω η διασύνδεση με την εξωτερική βάση δεδομένων χρησιμοποιήσαμε το έναν MySQL Connector ο οποίος χρειάζεται μια βιβλιοθήκη κρυπτογράφησης (sha1.h) κατασκευασμένη από τον Peter Knight.


```
8 #include <stdlib.h>
```

Η βιβλιοθήκη περιέχει αρκετές γενικής φύσεως συναρτήσεις, όπως δυναμική διαχείριση μνήμης, δημιουργία τυχαίων αριθμών, επικοινωνία με το περιβάλλον, ταξινόμηση, μετατροπή, αναζήτηση κλπ.

```
9 #include <mysql.h>
```

Η βιβλιοθήκη που απαιτείται για την διασύνδεση με την βάση MySQL.

```
10 #include <LiquidCrystal.h>
```

Η βιβλιοθήκη επιτρέπει στο Arduino να ελέγχει Liquid Crystal displays (LCDs) βασισμένες στο Hitachi HD44780 (ή σε συμβατό) chipset, το οποίο βρίσκουμε στις περισσότερες text-based LCDs. Η βιβλιοθήκη δουλεύει είτε σε 4 ή 8-bit mode (χρησιμοποιώντας δηλαδή 4 ή 8 γραμμές δεδομένων εκτός από τις RS, Enable, RW γραμμές ελέγχου).

```
11 #define greenPin 5
12 #define redPin 6
13 #define speakerPin 7
14 #define lockPin 8
15 #define rxpin 11
16 #define txpin 255
17 #define unlockLength 500
```

Η #define είναι χρήσιμη στην C διότι επιτρέπει στον προγραμματιστή να δώσει ένα όνομα σε μια constant value πριν το πρόγραμμα γίνει compiled. Οι #define constants στο Arduino δεν πίνουν χώρο στον τσιπ που αντιστοιχεί για την μνήμη του προγράμματος. Την ώρα του compile θα γίνει αντικατάσταση όπου υπάρχει αναφορά σε αυτές τις μεταβλητές με ότι έχει οριστεί στο #define.

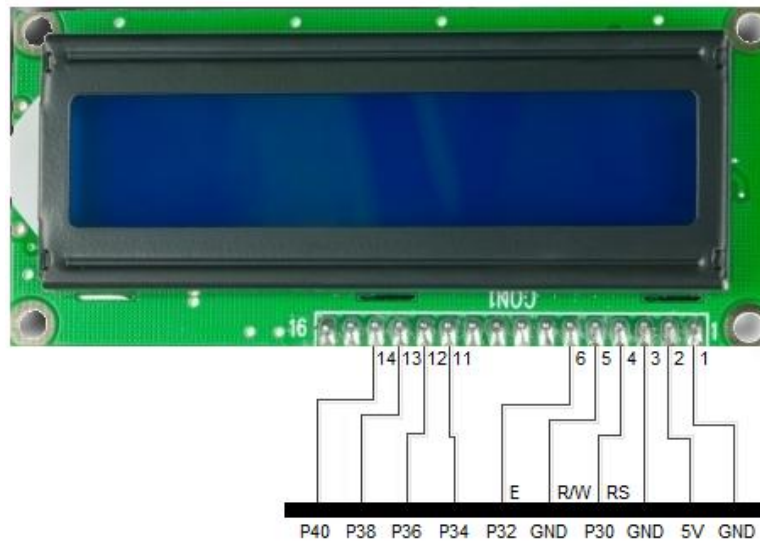
Εδώ έχουμε το greenPin στο ψηφιακό pin 5 του Arduino, το redPin στο pin 6, το speakerPin στο pin 7, το lockPin στο pin 8, το rxpin στο pin 11, το txpin στο 255 (πρακτικά δεν χρησιμοποιείται) και το unlockLength στα 500 msec (μισό δευτερόλεπτο).

```
35 LiquidCrystal lcd(30, 32, 34, 36, 38, 40);
```

Δημιουργείται μια μεταβλητή lcd τύπου LiquidCrystal. Ο έλεγχος της LCD μπορεί να γίνει με 4 ή με 8 γραμμές δεδομένων. Στην συγκεκριμένη υλοποίηση χρησιμοποιήσαμε 4bit mode (4 γραμμές δεδομένων) στα ψηφιακά pin 34, 36, 38, 40.

Η σύνταξη έχει την μορφή **LiquidCrystal(rs, enable, d34, d36, d38, d40)**, όπου rs είναι το νούμερο του Arduino pin το οποίο είναι διασυνδεδεμένο με το RS pin του LCD και enable το νούμερο του Arduino pin το οποίο είναι διασυνδεδεμένο με το enable pin του LCD.

Παρακάτω στην *Εικόνα 63* ακολουθεί ο τρόπος διασύνδεσης της LCD με Arduino Mega.



Εικόνα 63: Οθόνη LCD και διασύνδεση με το Arduino Mega

Έπειτα στον *Πίνακα 7* γίνεται μια αντιστοιχία των pins της LCD με τα digital pins του Arduino.

LCD Pinout	Arduino Pinout
Pin 1	GND
Pin 2	5V
Pin 3	GND
Pin 4 (RS)	Digital Pin 30
Pin 5 (R/W)	GND
Pin 6 (E)	Digital Pin 32
Pin 11	Digital Pin 34
Pin 12	Digital Pin 36
Pin 13	Digital Pin 38
Pin 14	Digital Pin 40

Πίνακας 7: LCD – Arduino pinout

```
37 SoftwareSerial rfidreader = SoftwareSerial( rxpin, txpin );
```

Χρησιμοποιούμε προς δικιά μας ευκολία την βιβλιοθήκη SoftwareSerial library (που πλέον είναι κομμάτι του Arduino IDE), διότι διαφορετικά εάν χρησιμοποιούσαμε τα στάνταρ Rx/Tx pins θα έπρεπε να τα ξεσυνδέουμε όποτε θα έπρεπε να ανεβάσουμε κώδικα στο Arduino και έπειτα να τα ξανασυνδέουμε πάλι. Χρησιμοποιώντας την βιβλιοθήκη SoftwareSerial μπορούμε πλέον να χρησιμοποιούμε όποιο pin θέλουμε. Έπειτα δημιουργούμε ένα instance ενός SoftwareSerial object το οποίο ονομάζουμε rfidReader και του δίνουμε τα rxpin / txpin που έχουμε ορίσει στο #define. Παρόλο που χρειάζεται να ορίσουμε και τα 2 pin πρακτικά

αυτό που θα χρησιμοποιήσουμε είναι το rxpin του Arduino, δηλαδή το Tx του RFID Reader χωρίς να επιστρέφει τίποτα πίσω στον Reader.

```
39 byte mac_addr[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
```

Ορίζω το mac address που θα έχει το Arduino και το οποίο πρέπει να είναι μοναδικό μέσα στο lan.

```
40 IPAddress ip(192, 168, 1, 200);
```

Ορίζω το ip address που θα έχει το Arduino και το οποίο πρέπει να είναι μοναδικό μέσα στο lan.

```
41 IPAddress server_addr(192, 168, 1, 100);
```

Ορίζω την ip address που θα έχει ο Cubieboard Server και η οποία πρέπει να είναι μοναδική μέσα στο lan.

```
42 EthernetServer server(80);
```

Έπειτα δημιουργούμε ένα instance του EthernetServer object μαζί με το port number της συσκευής. Ο server στο Arduino θα ακούει για εισερχόμενες συνδέσεις στο port 80, δηλαδή για καλύτερη κατανόηση αν προσπαθούσαμε να το παρομοιάσουμε θα λέγαμε πως το MAC address είναι η διεύθυνση ενός κτιρίου και το port number ο αριθμός του διαμερίσματος στο κτίριο αυτό.

```
43 char user[] = "root";  
44 char password[] = "my4cc3ss";
```

Εισάγουμε το username και το password του χρήστη της MySQL που θα έχει ελεύθερη πρόσβαση στην συγκεκριμένη βάση που θα συνδέεται το Arduino. Στην περίπτωση που θέλουμε να συνδέσουμε περισσότερα Arduino θα ήταν καλό να δημιουργήσουμε περισσότερους χρήστες με τον καθένα να έχει πρόσβαση στην δικιά του βάση.

```
46 Connector my_conn;
```

Η διασύνδεση με την βάση δεδομένων MySQL υλοποιείται μέσω ενός database connector που ονομάζεται Connector/Arduino και στον οποίο γίνεται υλοποίηση του MySQL client communication protocol μέσα σε μια βιβλιοθήκη φτιαγμένη για την πλατφόρμα του Arduino. Για να μπορέσουμε να καλέσουμε την βιβλιοθήκη του Connector θα πρέπει να έχουμε ένα Ethernet Shield οπότε και την βιβλιοθήκη Ethernet η οποία με την σειρά της απαιτεί την βιβλιοθήκη SPI.

Επιπλέον ο Connector χρειάζεται και την βιβλιοθήκη SHA1, άρα για ένα ελάχιστο sketch με δυνατότητα διασύνδεσης σε εξωτερική βάση χρειαζόμαστε τις παρακάτω βιβλιοθήκες:

- #include <Ethernet.h>
- #include <SPI.h>
- #include <sha1.h>
- #include <mysql.h>

```
48 // To store incoming serial data
49 int incomingByte = 0;
50
51 File myFile;
52 File webFile;
53 unsigned long time;
54 unsigned long seconds;
```

Εδώ ορίζουμε μια μεταβλητή για να κρατάει τα εισερχόμενα data όπως διαβάζονται από την σειριακή πόρτα. Επιπλέον δημιουργώ δυο file instances το myFile για να γράφω τα logs του access control στην SD κάρτα και webFile για να τα διαβάζω απομακρυσμένα μέσω του web server. Ακόμα ορίζω και τις μεταβλητές time και seconds τις οποίες θα χρησιμοποιήσω αργότερα για την δημιουργία ενός counter.

```
56 /* Keypad */
57
58 const byte ROWS = 4;    // four rows
```

Ορίζω τον αριθμό των σειρών που θα έχει το keypad.

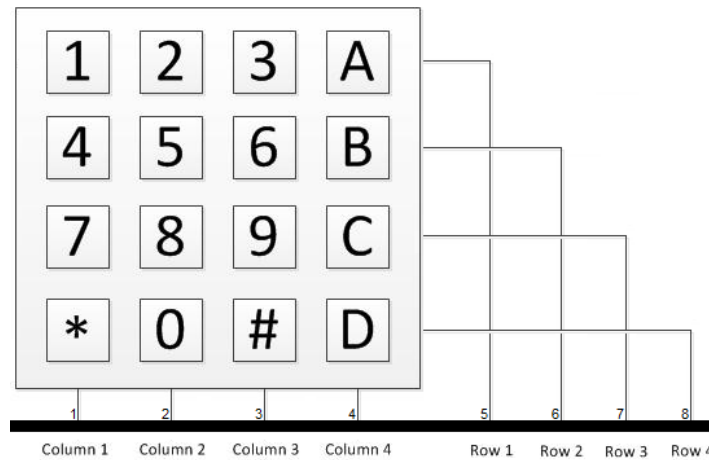
```
59 const byte COLS = 4;    // four columns
```

Ορίζω τον αριθμό των γραμμών που θα έχει το keypad.

```
61 char keys[ROWS][COLS] =
62 {
63   {'1','2','3','A'},
64   {'4','5','6','B'},
65   {'7','8','9','C'},
66   {'*','0','#','D'}
67 };
```

Ορίζω έναν πίνακα keys με περιεχόμενο τις σειρές και τις γραμμές όπως ακριβώς εμφανίζονται στο keypad.

Πιο συγκεκριμένα το keypad που χρησιμοποιήσαμε έχει την παρακάτω μορφή:



Έπειτα γίνεται μια αντιστοιχία των pins του keypad με τα digital pins του Arduino.

Keypad Pinout	Arduino Pinout
Column 1 (Pin 1)	Digital Pin 35
Column 2 (Pin 2)	Digital Pin 37
Column 3 (Pin 3)	Digital Pin 39
Column 4 (Pin 4)	Digital Pin 41
Row 1 (Pin 5)	Digital Pin 43
Row 2 (Pin 6)	Digital Pin 45
Row 3 (Pin 7)	Digital Pin 47
Row 4 (Pin 8)	Digital Pin 49

```
69 byte rowPins[ROWS] = { 43, 45, 47, 49 }; // define row pinouts of the keypad
```

Ορίζω τα digital pins του Arduino που αντιστοιχούν στα rows 1 έως 4 του keypad.

```
70 byte colPins[COLS] = { 35, 37, 39, 41 }; // define column pinouts of the keypad
```

Ορίζω τα digital pins του Arduino που αντιστοιχούν στα columns 1 έως 4 του keypad.

```
72 Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );
```

Ξεκινάει το instance της Keypad class.

```
74 char PIN[4]={'2','5','A','2'}; // secret code
```

Στον πίνακα PIN αποθηκεύω τον κωδικό που θα χρειάζεται να πληκτρολογήσει κάποιος ώστε να έχει πρόσβαση σε έναν ελεγχόμενο χώρο.

```
75 char attempt[4]={'0','0','0','0'}; // used for comparison
```

Στον πίνακα attempt κάνω μια αρχικοποίηση και αργότερα στην κατάλληλη συνάρτηση αφού έχει γίνει κάποια εισαγωγή κωδικού θα τον καλέσω για μια σύγκριση με τον PIN.

```
77 int z=0;
```

Κάνω μια αρχικοποίηση στην μεταβλητή z που θα την χρειαστώ σε παρακάτω συνάρτηση.

```
79 void setup()
80 {
81     Serial.println(" ");
82     Serial.begin(9600);
```

Στην void setup () περιγράφονται όσα θέλουμε να τρέξουν μια μόνο φορά. Ξεκινώντας λοιπόν θα εκκινήσει η σειριακή επικοινωνία αφήνοντας το baud rate default στα 9600bps.

```
83     rfidreader.begin(9600);
```

Ακολουθεί η εκκίνηση της επικοινωνίας με το rfid reader id201a με το baud rate στα 9600bps.

```
85     // columns, rows of a 16x2 LCD
86     lcd.begin(16,2);
```

Έπειτα γίνεται η εκκίνηση του interface της LCD οθόνης και δηλώνονται οι διαστάσεις (σε στήλες και σειρές) του display.

```
87     // start with a blank screen
88     lcd.clear();
```

Ξεκινάει ο καθαρισμός της οθόνης με τοποθέτηση του κέρσορα στην πάνω αριστερή γωνία.

```
90     Ethernet.begin(mac_addr, ip);
```

Γίνεται εκκίνηση της επικοινωνίας Ethernet ρυθμίζοντας την IP και το MAC address της συσκευής.

```
91     server.begin();
```

Έπειτα λέμε στον server να ξεκινήσει να ακούει για εισερχόμενες συνδέσεις χρησιμοποιώντας την εντολή begin().

```
93     Serial.println("RFID Reader Initialized");
94     Serial.println("Ethernet Interface Initialized");
```

Τυπώνω στο serial monitor τα παραπάνω μηνύματα.

```

96     if (!SD.begin(4))
97     {
98         Serial.println("SD Card Initialization failed!");
99     }
100    Serial.println("SD Card Initialized");
101    Serial.println(" ");

```

Η εντολή `SD.begin` εκκινεί την SD κάρτα και την βιβλιοθήκη SD και επιστρέφει `true` αν έχει γίνει επιτυχημένα η εκκίνηση ή `false` εάν όχι. Εφόσον ο έλεγχος με την `if !SD.begin` (δηλαδή NOT `SD.begin`) δείξει ότι η διαδικασία εκκίνησης έχει αποτύχει, τότε ενημερώνεται σχετικά ο χρήστης στο serial monitor με αντίστοιχο μήνυμα.

```

103    my_conn.mysql_connect(server_addr, 3306, user, password);

```

Έπειτα με την `mysql_connect()` δημιουργώ μια σύνδεση στην εξωτερική βάση δεδομένων χρησιμοποιώντας την `ip` του server, το `port`, το όνομα του χρήστη που έχει πρόσβαση και τον κωδικό του.

```

105    pinMode (speakerPin, OUTPUT);

```

Το `speakerPin` δηλώνεται ως έξοδος του Arduino.

```

106    pinMode (greenPin, OUTPUT);
107    pinMode (redPin, OUTPUT);
108    pinMode (lockPin, OUTPUT);

```

Ρυθμίζουμε τα `greenPin`, `redPin`, `lockPin` ως εξόδους του Arduino.

```

110    /* the hardware SS pin (53 on the Mega) must be left as an output
111    or the SD library functions will not work. */
112
113    pinMode (53, OUTPUT);

```

Το `pin 53` πρέπει να δηλωθεί ως έξοδος διότι στο Mega είναι το hardware SS pin, ώστε να λειτουργεί σωστά η βιβλιοθήκη SD.

```

115    digitalWrite(greenPin, LOW);
116    digitalWrite(redPin, LOW);
117    digitalWrite(lockPin, LOW);

```

Θέτουμε τα παραπάνω pins σε `LOW (0V)`, ώστε πχ το `greenPin/redPin` να είναι σβηστά κατά την εκκίνηση και το `lockPin` να μην διεγείρει με `5V` τον DC ρελέ.

```

119     Serial.println(" ");
120     Serial.print("Welcome User!");
121     Serial.println(" ");
122     Serial.print("For Login please use your RFID card.");
123     Serial.println(" ");
124     Serial.print("For Logout please Enter your PIN.");
125     Serial.println(" ");
126     Serial.println(" ");
127 }

```

Εμφανίζονται τα παραπάνω μηνύματα στο serial monitor του Arduino.

```

130 void loop()
131 {
132     byte i = 0;
133     byte val = 0;
134     byte checksum = 0;
135     byte bytesRead = 0;
136     byte tempByte = 0;

```

Ακολουθεί μια αρχικοποίηση των μεταβλητών που θα χρησιμοποιηθούν στην loop(). Οι περισσότερες από αυτές εξηγούνται από μόνες τους και όσες όχι θα αναλυθούν περισσότερο παρακάτω.

```

138     // "Unique" tags are only 5 bytes but we need an extra byte for the checksum
139     byte tagBytes[6];
140     char tagValue[10];

```

Στην συγκεκριμένη υλοποίηση χρησιμοποιήσαμε το ID-20 RFID reader module όπου δουλεύει με μοναδικά rfid tags και ένα 5-byte identification code, οπότε φυσιολογικά θα χρειαζόμασταν η **tagBytes** να ήταν ένας πίνακας 5 στοιχείων. Εντούτοις, έχουμε ορίσει ότι θα είναι 6 στοιχείων, διότι μετά το διάβασμα του tag id το sketch υπολογίζει το checksum value ώστε να σιγουρέψει ότι δεν έχουν γίνει λάθη διαβάσματος και το checksum έπειτα προσθέτετε στο τέλος ως έκτο στοιχείο.

Όσον αφορά την **tagValue** είναι ένας πίνακας από χαρακτήρες, και χρειάζεται να έχει μέγεθος 10 στοιχείων διότι από την στιγμή που ένα 5-byte tag value μετατραπεί σε ένα ASCII string τότε επεκτείνεται σε 10 χαρακτήρες. Έτσι λοιπόν όπως θα διαπιστώσουμε και κατά την διάρκεια ελέγχου των tags καθένα από αυτά έχει μέγεθος 10 characters.

```

142     time = millis();
143     seconds = (time/1000);

```

Η time επιστρέφει τον αριθμό σε milliseconds από την στιγμή που το Arduino board ξεκίνησε να τρέχει το συγκεκριμένο πρόγραμμα. Έπειτα γίνεται μετατροπή του σε δευτερόλεπτα μέσω της seconds. Ο αριθμός αυτός θα γίνει overflow (και θα μηδενιστεί εκ νέου) αφού περάσουν 50 ημέρες από την ημέρα που ξεκίνησε να μετράει.


```
145 EthernetClient client = server.available();           // try to get client
```

Χρειάζεται να ακούμε για εισερχόμενους clients πχ αιτήσεις για σελίδες που είναι φορτωμένες στο Arduino κλπ. Για να γίνει αυτό πρέπει να δημιουργήσουμε ένα instance τύπου EthernetClient και να το χρησιμοποιήσουμε για να ελέγχουμε αν υπάρχουν διαθέσιμα δεδομένα για διάβασμα.

```
147 readKeypad();
```

Καλούμε την συνάρτηση readKeypad().

```
149 // set cursor to column 0, row 0 (the first row)
150 lcd.setCursor(0,0);
```

Τοποθετώ τον κέρσορα της lcd στην στήλη 0 - σειρά 0 (δηλ. πάνω αριστερά).

```
151 lcd.print("Plz Enter Pin ..");
152 // set cursor to column 0, row 1
153 lcd.setCursor(0,1);
154 lcd.print("Welcome User!");
```

Τυπώνω ένα μήνυμα στην lcd, τοποθετώ τον κέρσορα της lcd στην στήλη 0 - σειρά 1 και τυπώνω πάλι ένα μήνυμα.

```
156 if (client)
157 {
158     // got client?
159     boolean currentLineIsBlank = true;
160     while (client.connected())
161     {
162         if (client.available())
163         {
164             // client data available to read
165             {
166                 char c = client.read();
167                 // read 1 byte (character) from client
```

Εφόσον υπάρχουν clients και δεδομένα να διαβαστούν τότε διάβασε 1 byte (έναν δηλαδή χαρακτήρα του client) και αποθήκευσε τον στην μεταβλητή c τύπου char.

```

164 // last line of client request is blank and ends with \n
165 // respond to client only after last line received
166 if (c == '\n' && currentLineIsBlank)
167 {
168     // send web page
169     webFile = SD.open("logs.htm"); // open web page file
170     if (webFile)
171     {
172         while(webFile.available())
173         {
174             client.write(webFile.read()); // send web page to client
175         }
176         webFile.close();
177     }
178     break;
179 }

```

Από την στιγμή που γίνει λήψη της τελευταίας γραμμής του client κενή και τελειώνει με \n, τότε το sketch ανοίγει το αρχείο logs.htm που βρίσκεται στην SD κάρτα, διαβάζει το περιεχόμενο του και έπειτα το κλείνει.

```

181 // every line of text received from the client ends with \r\n
182 if (c == '\n')
183 {
184     // last character on line of received text
185     // starting new line with next character read
186     currentLineIsBlank = true;
187 }
188 else if (c != '\r')
189 {
190     // a text character was received from client
191     currentLineIsBlank = false;
192 }

```

Παρακάτω ακολουθούν οι ενέργειες στην περίπτωση που ο τελευταίος χαρακτήρας που έχει διαβάσει ο c είναι ο \n ή \r .

```

193     } // end if (client.available())
194 } // end while (client.connected())
195 delay(1); // give the web browser time to receive the data
196 client.stop(); // close the connection
197 } // end if (client)

```

Τέλος της client.available() και κλείσιμο της σύνδεσης με τον client.

```

200 // Read the RFID reader by checking for header
201 if((val = rfidreader.read()) == 2)
202 {

```

Έπειτα γίνεται έλεγχος αν υπάρχουν εισερχόμενα δεδομένα από την σειριακή πόρτα του RFID reader. Αν ναι και ο πρώτος χαρακτήρας είναι ο ASCII character 2, που είναι ο transmission start code, τότε ξέρουμε ότι ένα ID string είναι έτοιμο να μεταδοθεί και

ξεκινάμε να διαβάζουμε τα ψηφία αποθηκεύοντας τα στην μεταβλητή val. Οπότε, το sketch στην αρχή διαβάζει ένα byte από το rfid virtual serial device και ελέγχει εάν είναι header value (δηλ. ASCII character 2).

```
203     bytesRead = 0;
204
205     // Read 10 digit code + 2 digit checksum
206     while (bytesRead < 12)
207     {
208         if(rfidreader.available() > 0)
209         {
210             //store value in val
211             val = rfidreader.read();
```

Γίνεται reset το bytesRead counter (δηλ. μηδέν) και μπαίνει σε ένα loop ώστε να συνεχίσει να δέχεται τιμές από το RFID reader μέχρι να δει συνολικά 12 χαρακτήρες, δηλαδή το 10-character tag code και επιπλέον το 2-character checksum.

```
213         // Append the first 10 bytes (0 to 9) to the raw tag value
214         if (bytesRead < 10)
215         {
216             tagValue[bytesRead] = val;
217         }
```

Αποθηκεύουμε προς δικιά μας χρήση αργότερα στο πρόγραμμα το tag ID σε δυο διαφορετικά formats. Το πρώτο είναι μια σειρά από raw values αποθηκευμένες σε έναν πίνακα χαρακτήρων, όπου βρίσκεται μόνο το tag ID. Δεν θέλουμε να περιέχει το checksum, αλλά μόνο τους πρώτους 10 χαρακτήρες σε στοιχεία μέσα στο πίνακα tagValue που ξεκινούν από 0 έως 9.

```
219         // Check if there is STX, ETX, CR or LF before the 10 digit reading is complete
220         if((val == 0x0D)|| (val == 0x0A)|| (val == 0x03)|| (val == 0x02))
221         {
222             // Stop reading
223             break;
224         }
```

Στην περίπτωση που το σύστημα βρεθεί σε μια κατάσταση όπου πχ ο ID20 RFID reader έχει στείλει ένα κομμάτι μόνο από το tag value ο κώδικας ελέγχει κάθε τιμή για να διαπιστώσει εάν είναι header ή stop byte. Εάν δει κάποια από αυτά όσο το πρόγραμμα είναι μέσα στο loop() υποθέτει ότι κάτι δεν έχει πάει καλά και καλεί την break για να βγει έξω από την loop() και να ξεκινήσει ξανά. Έτσι λοιπόν με αυτό τον τρόπο το πρόγραμμα μπορεί να επανέλθει από τέτοιες καταστάσεις όπως όταν διαβάσει κομμάτι του tag.

```

226 // Convert ascii hex to integer hex value
227 if ((val >= '0') && (val <= '9'))
228 {
229     val = val - '0';
230 }
231 else if ((val >= 'A') && (val <= 'F'))
232 {
233     val = 10 + val - 'A';
234 }

```

Το δεύτερο format απαιτεί μερική επεξεργασία και αποθήκευση κάθε τιμής σαν ξεχωριστό στοιχείο σε έναν πίνακα από bytes, ώστε το πρόγραμμα να κάνει μερικές μετατροπές ASCII/Hex σε κάθε τιμή.

```

236 // Every two hex-digits, add a byte to the code
237 // So if odd number merge 2 4bit digits into one (8 bit) byte
238 if (bytesRead & 1 == 1)
239 {

```

Ένα byte είναι ένα σύνολο από 8 bits και μπορεί να κρατήσει μια τιμή από 0 έως decimal 255, ενώ ένα hexadecimal digit είναι ένα 4-bit value (μισό byte) από 0 έως F. Για αυτό μπορούμε να αποθηκεύσουμε δυο hex digits σε κάθε byte σαν ένα ζευγάρι από 4-bit values. Η δομή του κώδικα μοιάζει να είναι λίγο ανάποδο, δηλαδή το πρώτο κομμάτι του κώδικα έχει να κάνει με το δεύτερο ζευγάρι των hex digits, ενώ το δεύτερο κομμάτι του κώδικα έχει να κάνει με το πρώτο ζευγάρι.

Ο κώδικας πρώτα ελέγχει εάν έχει να κάνει με το δεύτερο ζευγάρι των hex digit και αν είναι θετικό τότε η παραπάνω σύγκριση είναι true.

```

240 // Make space for this hex-digit by shifting the previous digit 4 bits to the left
241 tagBytes[bytesRead >> 1] = (val | (tempByte << 4));

```

Αυτό το κομμάτι του κώδικα έχει να κάνει με το δεύτερο ζευγάρι των hex digits. Κάνει χώρο για τα ψηφία που υπάρχουν ήδη στο byte μέσω μιας αριστερής ολίσθησης 4 bits αφήνοντας έτσι χώρο να τοποθετηθούν και τα υπόλοιπα 4 bits του byte χρησιμοποιώντας την τιμή του bytesRead.

```

243 // If we're at the checksum byte,
244 if (bytesRead >> 1 != 5)
245 {
246     // Calculate the checksum... (XOR)
247     checksum ^= tagBytes[bytesRead >> 1];
248 };

```

Έπειτα ελέγχει αν έχει φτάσει στο checksum byte και αν ναι τότε υπολογίζει το checksum χρησιμοποιώντας μια XOR operation.

```

249     }
250     else
251     {
252         // Store the first hex digit first
253         tempByte = val;
254     };

```

Εδώ έχουμε να κάνουμε με το πρώτο ζευγάρι των hex digits τοποθετώντας την τιμή απευθείας στην μεταβλητή. Έπειτα, η τιμή αυτή στο επόμενο loop σύμφωνα με το προηγούμενο κομμάτι του κώδικα που αναφέραμε γίνεται ολίσθηση 4 bits στα αριστερά.

```

255     // Ready to read next digit
256     bytesRead++;
257 }
258

```

Το sketch έπειτα αυξάνει τον μετρητή ο οποίος παρακολουθεί πόσα bytes έχουν διαβαστεί και πάει πάλι στην αρχή για να ελέγξει άμα έχει τελειώσει το διάβασμα όλων των ψηφίων.

```

260     // Send the result to the host connected via USB
261     if (bytesRead == 12)
262     // 12 digit read is complete
263     {

```

Το sketch ελέγχει εάν 12 bytes έχουν διαβαστεί, δηλαδή ότι έχει τελειώσει επιτυχώς ένα πλήρες διάβασμα ενός tag και έπειτα προχωράει στην σύγκριση του tag που έχει μόλις διαβαστεί με αυτά που επιτρέπονται.

```

264     // We have reached the end of all 12 digits and read is complete, so add a null terminator
265     tagValue[10] = '\0';

```

Ο πίνακας χαρακτήρων tagValue περιέχει μια σειρά από 10 χαρακτήρες στα στοιχεία 0 έως 9 του πίνακα, αλλά πρέπει να είμαστε σίγουροι τότε θα τερματίσει το sketch ώστε να το αντιμετωπίσει ως string όταν φτάσει στο τέλος. Γιαυτό λοιπόν τοποθετούμε στο εντέκατο στοιχείο του πίνακα (δηλ. tagValue[10]) το χαρακτήρα null (\0).

```

266 Serial.print("Tag read: ");
267 for (i=0; i<5; i++)
268 {
269     // Add a leading 0 to pad out values below 16
270     if (tagBytes[i] < 16)
271     {
272         Serial.print("0");
273     }
274     Serial.print(tagBytes[i], HEX);
275 }
276 Serial.println();
277 Serial.print("Checksum: ");
278 Serial.print(tagBytes[5], HEX);
279 Serial.println(tagBytes[5] == checksum ? " -- passed." : " -- error.");

```

Επιπλέον τυπώνεται το checksum το οποίο έχει υπολογιστεί για το κάθε tag που έχει διαβαστεί και γίνεται σύγκριση του με την τιμή που έχει προμηθεύσει το ίδιο το tag reader (τελευταία 2 ψηφία από τα 12 που διαβάζονται κάθε φορά), ώστε να διαπιστωθεί εάν ταιριάζουν και το οποίο επίσης τυπώνεται στο serial monitor.

```

281 // Show the raw tag value
282 Serial.print("VALUE: ");
283 Serial.println(tagValue);

```

Για να υπάρχει feedback κατά την διάρκεια δοκιμών εγγραφής του κώδικα έχει επιλεγεί να τυπώνονται κάποια μηνύματα στο serial monitor του Arduino IDE ώστε να μπορεί κάποιος να δει την πορεία εκτέλεσης του κώδικα. Εδώ λοιπόν γίνεται print στο serial monitor η τιμή του tagValue που έχει διαβαστεί.

```

285 // Search the tag database for this particular tag
286 int tagId = findTag( tagValue );

```

Καλείται η συνάρτηση findTag() με είσοδο το tagValue που είναι η τιμή του tag και τελικά αυτό που θα επιστρέψει η συνάρτηση θα είναι μορφής ακεραίου & το οποίο θα αποθηκευτεί στο tagId.

```

288 // Only fire the strike plate if this tag was found in the database
289 if( tagId > 0 )
290 {

```

Εάν αυτό που έχει επιστρέψει η findTag είναι μεγαλύτερο του μηδέν σημαίνει ότι έχει βρεθεί το tag στην βάση και το sketch κάνει τα παρακάτω.

```

291 Serial.print("Time: ");
292 Serial.println(seconds);
293 Serial.print("Authorized tag ID");
294 Serial.println(" ");
295 Serial.print("Login done!");
296 Serial.println(" ");

```

Τυπώνει στο serial monitor τον χρόνο σε second μαζί με τα σχετικά μηνύματα.

```

298 // unlock the door
299 unlock();

```

Καλεί την συνάρτηση unlock().

```

301 // open and write tagValue to log.txt
302 myFile = SD.open("logs.htm", FILE_WRITE);
303 if (myFile)
304 {
305     myFile.print("Time: ");
306     myFile.println(seconds);
307     myFile.print("VALUE: ");
308     myFile.println(tagValue);
309     myFile.print("Authorized tag ID");
310     myFile.println(" ");
311     myFile.println("Login done!");
312     myFile.println(" ");
313     myFile.close();
314 }

```

Έπειτα στο αρχείο logs.htm που βρίσκεται στην SD κάρτα γίνεται εγγραφή του χρόνου, του tag value και μερικών μηνυμάτων ακόμα και στο τέλος κλείνει το αρχείο.

```

315 }
316 else
317 {
318     Serial.print("Time: ");
319     Serial.println(seconds);
320     Serial.print("Tag ID not Authorized!");
321     Serial.println();

```

Σε διαφορετική περίπτωση (δηλαδή το tagId = 0) τυπώνει στο serial monitor τον χρόνο μαζί με το σχετικό μήνυμα.

```

323 // door stays locked
324 lock();

```

Και καλεί την συνάρτηση lock().

```

325
326 // open and write tagValue to log.txt
327 myFile = SD.open("logs.htm", FILE_WRITE);
328 if (myFile)
329 {
330     myFile.print("Time: ");
331     myFile.println(seconds);
332     myFile.print("VALUE: ");
333     myFile.println(tagValue);
334     myFile.print("Tag ID not Authorized!");
335     myFile.println(" ");
336     myFile.println(" ");
337     myFile.close();
338 }
339
340 }
341 // Blank separator line in output
342 Serial.println();
343 myFile.println();
344 }
345 bytesRead = 0;
346 }
347 }

```

Έπειτα όπως και πριν ανοίγει το αρχείο logs.htm που βρίσκεται στην SD κάρτα και γίνεται εγγραφή του χρόνου, του tag value και μερικών μηνυμάτων και στο τέλος κλείνει το αρχείο. Στην συνέχεια μηδενίζει το bytesRead και επιστρέφει στην αρχή του loop().

```

350 /* Search for a specific tag in the database */
351 int findTag( char tagValue[10] )
352 {

```

Η συνάρτηση findTag δέχεται σαν είσοδο το 10 χαρακτήρων tag id (tagValue[10]) και αναλαμβάνει να επιστρέψει στην loop() **1** εάν έχει βρεθεί στην εξωτερική βάση δεδομένων το tagValue ή **0** στην περίπτωση που δεν έχει βρεθεί.

```

353     char QUERY_POP[64];
354     sprintf(QUERY_POP,"SELECT id FROM door.tag WHERE id = '%s'",tagValue);
355     int rows = do_query(QUERY_POP);

```

Έπειτα καλείται η συνάρτηση do_query() με είσοδο το QUERY_POP και αυτό που θα επιστρέψει θα είναι πάλι ένας ακέραιος ο οποίος θα αποθηκευτεί στο rows. Το QUERY_POP θα διαβάσει το column **id** της βάσης **door** του table **tag** και θα ψάξει να δει εάν το tagValue βρίσκεται μέσα στην στήλη αυτή.


```

356     if (rows>0)
357     {
358         Serial.println("We found a match!");
359         return(1);
360     }
361     else
362         Serial.println("No match.");
363         return(0);
364 }

```

Μετά την αναζήτηση στην βάση εάν ο ακέραιος που έχει επιστραφεί και αποθηκευτεί στο rows είναι >0 τότε τυπώνει ένα μήνυμα στο serial monitor και επιστρέφει στην findTag() την τιμή **1** διαφορετικά επιστρέφει το αντίστοιχο μήνυμα και την τιμή **0**.

```

367 int do_query(const char *q)
368 {
369     // pointer to column values
370     column_names *c;
371
372     // pointer to row values
373     row_values *r;

```

Η συνάρτηση do_query() δέχεται σαν είσοδο έναν δείκτη q στο QUERY_POP και ξεκινάει την διαδικασία ελέγχου της βάσης δεδομένων για το tagValue δημιουργώντας δυο δείκτες στις τιμές των column και row αντίστοιχα.

```

375     // First, execute query. If it returns a value pointer,
376     // we have a result set to process. If not, we exit.
377     if (!my_conn.cmd_query(q))
378     {
379         return 0;
380     }

```

Έπειτα τρέχει η cmd_query() ώστε να ξεκινήσει το query και εάν επιστρέψει ο δείκτης μια τιμή έχουμε αποτέλεσμα διαφορετικά επιστρέφει 0.

```

382 // Next, we read the column names and display them.
383 //
384 // NOTICE: You must *always* read the column names even if
385 //          you do not use them. This is so the connector can
386 //          read the data out of the buffer. Row data follows the
387 //          column data and thus must be read first.
388
389 c = my_conn.get_columns();
390 for (int i = 0; i < c->num_fields; i++)
391 {
392     Serial.print(c->fields[i]->name);
393     if (i < c->num_fields - 1)
394     {
395         Serial.print(",");
396     }
397 }
398 Serial.println();

```

Στο κομμάτι αυτό της `do_query()` διαβάζονται πρώτα τα columns και έπειτα τα rows της βάσης.

```

400 // We use the get_next_row() iterator and read rows printing
401 // the values returned until the get_next_row() returns NULL.
402
403 int num_cols = c->num_fields;
404 int rows = 0;
405 do
406 {
407     r = my_conn.get_next_row();
408     if (r)
409     {
410         rows++;
411         for (int i = 0; i < num_cols; i++)
412         {
413             Serial.print(r->values[i]);
414             if (i < num_cols - 1)
415             {
416                 Serial.print(", ");
417             }
418         }
419         Serial.println();

```

Διαβάζονται τα περιεχόμενα της κάθε γραμμής έως ότου η `get_next_row()` επιστρέψει null που σημαίνει ότι έχουν διαβαστεί όλες οι γραμμές της στήλης που μας ενδιαφέρει. Επιπλέον το `rows` μετράει σε πόσες σειρές βρέθηκε η εγγραφή που μας ενδιαφέρει.

```

421     // We free the row read to free the memory allocated for it.
422     my_conn.free_row_buffer();

```

Το query έχει τελειώσει και απελευθερώνονται οι πόροι του συστήματος.

```

423     }
424     } while (r);
425     Serial.print(rows);
426     Serial.println(" rows in result.");

```

Εκτυπώνει στο serial monitor τον αριθμό των εγγραφών που βρέθηκαν.

```

428     // Finally, we are done so we free the column buffers
429     my_conn.free_columns_buffer();

```

Το query έχει τελειώσει και απελευθερώνονται οι πόροι του συστήματος.

```

430     return rows;
431 }

```

Επιστρέφει πίσω στην findTag() τον αριθμό των rows που βρήκε στην βάση με το συγκεκριμένο tagValue. Αν είναι 0 σημαίνει δεν βρήκε κανένα, αν είναι 1 σημαίνει βρήκε μια εγγραφή και εάν είναι παραπάνω από 1 σημαίνει βρέθηκαν πολλαπλές εγγραφές.

```

435 /* If an authorised tag was found in the database unlock the door */
436 void unlock()
437 {
438     tone(speakerPin, 1000, 500);
439     digitalWrite(greenPin, HIGH);
440     delay(1000);
441     digitalWrite(greenPin, LOW);
442     digitalWrite(lockPin, HIGH);
443     delay(unlockLength);
444     digitalWrite(lockPin, LOW);
445 }

```

Η συνάρτηση unlock() καλείται στην περίπτωση που ανιχνευτεί ένα tag το οποίο είναι καταχωρημένο στην βάση δεδομένων. Σε αυτή την περίπτωση έχει ρυθμιστεί να ακούγεται ένας ήχος από το speaker καλώντας την tone(pin, frequency, duration) όπου το pin είναι αυτό που έχει συνδεθεί στο speaker του Arduino, η ακουστική συχνότητα είναι αυτή που έχει επιλεγεί δηλ. 1khz και η διάρκεια είναι μισό δευτερόλεπτο. Το πράσινο λαμπάκι παίρνει τροφοδοσία ανάβει για ένα δευτερόλεπτο και έπειτα κλείνει. Ακόμα το lockPin που είναι η διέγερση του DC ρελέ της κλειδαριάς γίνεται HIGH οπότε κλείνει το κύκλωμα τροφοδοσίας με 12V του electric strike και μπορεί κάποιος να ανοίξει την πόρτα. Η διάρκεια που το lockPin είναι High καθορίζεται από το unlockLength το οποίο ορίζεται με την σειρά του στην αρχή του προγράμματος και μετά το τέλος του χρόνου αυτού το lockPin γίνεται πάλι

LOW οπότε ανοίγει το κύκλωμα τροφοδοσίας του electric strike και πλέον δεν μπορεί κάποιος να ανοίξει την πόρτα.

```
448 /* If an unauthorised tag was found do the following */
449 void lock()
450 {
451     tone(speakerPin, 250, 250);
452     delay(250);
453     tone(speakerPin, 150, 250);
454     digitalWrite(redPin, HIGH);
455     delay(1000);
456     digitalWrite(redPin, LOW);
457 }
```

Η συνάρτηση lock() καλείται στην περίπτωση που ανιχνευτεί ένα tag το οποίο δεν είναι καταχωρημένο στην βάση δεδομένων. Σε αυτή την περίπτωση έχει ρυθμιστεί να ακούγεται ένας ήχος από το speaker καλώντας την tone(pin, frequency, duration) όπου το pin είναι αυτό που έχει συνδεθεί στο speaker του Arduino και η ακουστική συχνότητα είναι αυτή που έχει επιλεγεί δηλ. 250hz με την διάρκεια να είναι 250ms. Μετά από αυτό το delay αλλάζει η συχνότητα του speaker από 250 σε 150hz για άλλα 250ms και το κόκκινο λαμπάκι παίρνει πλέον τροφοδοσία ανάβοντας για ένα δευτερόλεπτο έως ότου το pin γίνει LOW όπου και κλείνει.

```
460 void readKeypad()
461 {
462     char key = keypad.getKey();
```

Η συνάρτηση readKeypad() καλείται να διαβάσει το κωδικό που πληκτρολογούμε στο keypad μέσω της getKey() και ο οποίος αποθηκεύεται στην key.

```
463     if (key != NO_KEY)
464     {
465         attempt[z]=key;
466         z++;
```

Στην περίπτωση λοιπόν που έχει πατηθεί κάποιο κουμπί (δηλαδή key διάφορο του NO_KEY) βάλτο το key σε ένα πίνακα attempt[z] και αύξησε το z κατά ένα. Ο attempt[] συνεχίζει να γεμίζει με τα κουμπιά που πατιούνται μέχρι να γίνει μια από τις παρακάτω δυο ενέργειες, δηλαδή πατηθεί το κουμπί reset (*) ή το κουμπί enter (#).

```
467         switch(key)
468         {
469             case '*':
470                 z=0;
471                 break;
```

Στην περίπτωση που πατηθεί το * (δηλ. Reset pin) μηδένισε το z και βγες.

```
472         case '#':
473             z=0;
474             delay(100);
475             checkPIN();
476             break;
477     }
478 }
479 }
```

Στην περίπτωση που πατηθεί το # (δηλ. Enter pin) μηδενίζει το z, περιμένει 100ms και τρέχει την συνάρτηση checkPIN() για να γίνει έλεγχος του pin και έπειτα βγαίνει.

```
482 // Pin Checked
483 void checkPIN()
484 {
485     int correct=0;
486
487     for (int i = 0; i < 4 ; i++ )
488     {
489         if (attempt[i]==PIN[i])
490         {
491             correct++;
492         }
493     }
```

Η συνάρτηση checkPIN() καλείται για να κάνει έλεγχο του pin που έχει πληκτρολογήσει ο χρήστης στο keypad. Ο μήκος του κωδικού έχει ρυθμιστεί στην αρχή του κώδικα να είναι 4 χαρακτήρων, οπότε και στην συνάρτηση ο έλεγχος θα γίνεται για 0 έως 4 (δηλ. 4 χαρακτήρες). Ο κώδικας λοιπόν της checkPIN() αυτό που κάνει είναι να ελέγχει αν ο κάθε χαρακτήρας που έχει πληκτρολογηθεί στην attempt[i] είναι ίδιος με αυτόν που έχει οριστεί στην PIN[i], οπότε κάθε φορά που βρίσκει έναν συνεχόμενο αριθμό ίδιο ανεβάζει τον μετρητή correct κατά ένα.

```
495     if (correct==4)
496     {
497         correctPIN();
498     }
```

Έπειτα αν ο μετρητής connect συνεχόμενων συγκρίσεων της attempt[i] με την PIN[i] γίνει 4 σημαίνει πως έχει επιβεβαιωθεί το pin και ακολουθεί η εκτέλεση της συνάρτησης correctPIN().

```

500     else
501     {
502         incorrectPIN();
503     }

```

Σε διαφορετική περίπτωση τρέχει την συνάρτηση incorrectPIN().

```

505     for (int j=0; j<4; j++)
506     {
507         attempt[j]='0';
508     }
509 }

```

Ταυτόχρονα μηδενίζει τον πίνακα attempt[] για την επόμενη σύγκριση.

```

512 // Do this if correct PIN entered
513 void correctPIN()
514 {
515     // start with a blank screen
516     lcd.clear();
517     // set cursor to column 0, row 0 (the first row)
518     lcd.setCursor(0,0);
519     lcd.print("* Logout Done! *");
520     // set cursor to column 0, row 1
521     lcd.setCursor(0,1);
522     lcd.print("* Correct PIN *");

```

Η συνάρτηση correctPIN() τρέχει όταν ο μετρητής correct πάρει την τιμή 4 οπότε ένα σωστό pin έχει ανιχνευτεί. Στην περίπτωση αυτή καθαρίζει την απεικόνιση της οθόνης LCD, τοποθετεί τον κέρσορα της LCD στο αντίστοιχο column και row τυπώνοντας τα κατάλληλα μηνύματα στην LCD.

```

524     Serial.print("Time: ");
525     Serial.println(seconds);
526     Serial.print("* Correct PIN *");
527     Serial.println(" ");
528     Serial.print("* Logout done! *");
529     Serial.println(" ");
530     Serial.println(" ");

```

Έπειτα τυπώνει στο serial monitor το χρόνο από τον μετρητή του συστήματος ο οποίος τρέχει από τότε που ξεκινάει το Arduino, καθώς και κάποια επιμέρους σχετικά μηνύματα.

```

532     myFile = SD.open("logs.htm", FILE_WRITE);
533     if (myFile)
534     {
535         myFile.print("Time: ");
536         myFile.println(seconds);
537         myFile.print("* Correct PIN *");
538         myFile.println(" ");
539         myFile.print("* Logout done! *");
540         myFile.println(" ");
541         myFile.println(" ");
542         myFile.close();
543     }

```

Παρακάτω για να υπάρχουν και logs της κάθε απόπειρας για πρόσβαση ανοίγει το αρχείο logs.htm στην SD κάρτα και τυπώνει εκεί ότι είχε τυπωθεί προηγουμένως στο serial monitor και αφού τελειώσει την εγγραφή το κλείνει.

```

545     tone(speakerPin, 1000, 500);
546     digitalWrite(greenPin, HIGH);
547     delay(1000);
548     digitalWrite(greenPin, LOW);
549     digitalWrite(lockPin, HIGH);
550     delay(unlockLength);
551     digitalWrite(lockPin, LOW);

```

Έπειτα έχει ρυθμιστεί να ακούγεται ένας ήχος από το speaker καλώντας την tone(pin, frequency, duration) όπου το pin είναι αυτό που έχει συνδεθεί στο speaker του Arduino, η ακουστική συχνότητα είναι αυτή που έχει επιλεγεί δηλ. 1khz και η διάρκεια είναι μισό δευτερόλεπτο. Το πράσινο λαμπάκι παίρνει τροφοδοσία ανάβει για ένα δευτερόλεπτο και έπειτα κλείνει. Ακόμα το lockPin που είναι η διέγερση του DC ρελέ της κλειδαριάς γίνεται HIGH οπότε κλείνει το κύκλωμα τροφοδοσίας με 12V του electric strike και μπορεί κάποιος να ανοίξει την πόρτα. Η διάρκεια που το lockPin είναι High καθορίζεται από το unlockLength το οποίο ορίζεται με την σειρά του στην αρχή του προγράμματος και μετά το τέλος του χρόνου αυτού το lockPin γίνεται πάλι LOW οπότε ανοίγει το κύκλωμα τροφοδοσίας του electric strike και πλέον δεν μπορεί κάποιος να ανοίξει την πόρτα.

```

553     // start with a blank screen
554     lcd.clear();
555 }

```

Καθαρίζει την LCD οθόνη από μηνύματα.

```

558 // Do this if incorrect PIN entered
559 void incorrectPIN()
560 {
561     // start with a blank screen
562     lcd.clear();
563     // set cursor to column 0, row 0 (the first row)
564     lcd.setCursor(0,0);
565     lcd.print("* Try again *");
566     // set cursor to column 0, row 1
567     lcd.setCursor(0,1);
568     lcd.print("* Invalid PIN *");

```

Η συνάρτηση `incorrectPIN()` τρέχει όταν ο μετρητής `correct` είναι για διαφόρους λόγους μικρότερος του 4, άρα και το σωστό pin δεν έχει ανιχνευτεί. Στην περίπτωση αυτή καθαρίζει την απεικόνιση της οθόνης LCD, τοποθετεί τον κέρσορα της LCD στο αντίστοιχο column και row τυπώνοντας ταυτόχρονα τα κατάλληλα μηνύματα στην LCD.

```

570     Serial.print("Time: ");
571     Serial.println(seconds);
572     Serial.print("* Incorrect PIN *");
573     Serial.println(" ");
574     Serial.print("* Try again *");
575     Serial.println(" ");
576     Serial.println(" ");

```

Έπειτα τυπώνει στο serial monitor το χρόνο από τον μετρητή του συστήματος ο οποίος τρέχει από τότε που ξεκινάει το Arduino, καθώς και κάποια επιμέρους σχετικά μηνύματα.

```

578     myFile = SD.open("logs.htm", FILE_WRITE);
579     if (myFile)
580     {
581         myFile.print("Time: ");
582         myFile.println(seconds);
583         myFile.print("* Incorrect PIN *");
584         myFile.println(" ");
585         myFile.print("* Try again *");
586         myFile.println(" ");
587         myFile.println(" ");
588         myFile.close();
589     }

```

Παρακάτω για να υπάρχουν και logs της κάθε απόπειρας για πρόσβαση ανοίγει το αρχείο `logs.htm` στην SD κάρτα και τυπώνει εκεί ότι είχε τυπωθεί προηγουμένως στο serial monitor και αφού τελειώσει την εγγραφή το κλείνει.


```
591     tone(speakerPin, 250, 250);
592     delay(250);
593     tone(speakerPin, 150, 250);
594     digitalWrite(redPin, HIGH);
595     delay(1000);
596     digitalWrite(redPin, LOW);
```

Έπειτα έχει ρυθμιστεί να ακούγεται ένας ήχος από το speaker καλώντας την `tone(pin, frequency, duration)` όπου το `pin` είναι αυτό που έχει συνδεθεί στο speaker του Arduino και η ακουστική συχνότητα είναι αυτή που έχει επιλεγεί δηλ. 250hz με την διάρκεια να είναι 250ms. Μετά από αυτό το `delay` αλλάζει η συχνότητα του speaker από 250 σε 150hz για άλλα 250ms και το κόκκινο λαμπάκι παίρνει πλέον τροφοδοσία ανάβοντας για ένα δευτερόλεπτο έως ότου το `pin` γίνει LOW όπου και κλείνει.

```
598     // start with a blank screen
599     lcd.clear();
600 }
```

Καθαρίζει την LCD οθόνη από μηνύματα.

7. Συμπεράσματα – Προτάσεις

Στην παρούσα διπλωματική εργασία, μελετήσαμε τις απαιτήσεις του πρωτοκόλλου RFID και ακολούθως συνδυάσαμε την τεχνολογία RFID κάνοντας χρήση μικροεπεξεργαστή ανοικτού λογισμικού (Arduino) που έχει τα ακόλουθα πλεονεκτήματα:

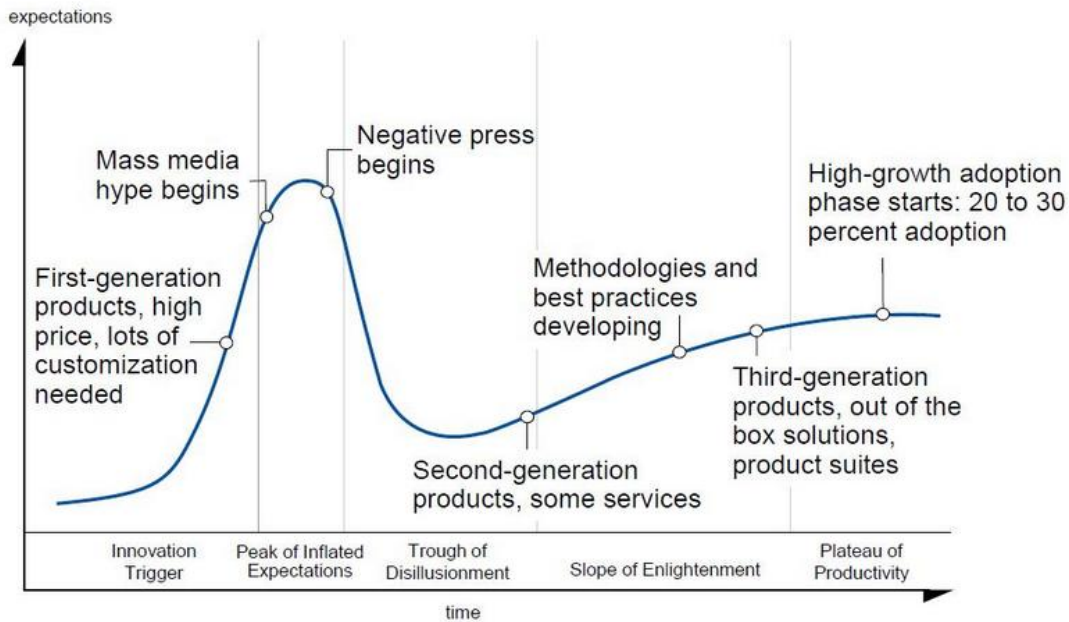
- Είναι ανεξαρτήτου πλατφόρμας (cross – platform)
- Έχει ανοικτού κώδικα λογισμικό
- Απλό περιβάλλον ανάπτυξης για τους νέους μη εξοικειωμένους χρήστες
- Έχει χαμηλό κόστος
- Εύκολη διασύνδεση μέσω USB για τον προγραμματισμό του μικροεπεξεργαστή
- Δυνατότητα τροφοδοσίας μέσω USB ή ακόμα και με POE με κατάλληλο module
- Έχει πολλαπλές αναλογικές και ψηφιακές εισόδους / εξόδους
- Πολύ μεγάλη online κοινότητα και ενεργή συνεισφορά των χρηστών μέσα από forum

Ακολούθως σχεδιάσαμε και υλοποιήσαμε ένα ολοκληρωμένο σύστημα πιστοποίησης ταυτότητας. Επίσης, δημιουργήσαμε μια πλατφόρμα έξυπνης διαχείρισης του συστήματος στην οποία τηρούνται αρχεία με τα ονόματα των χρηστών που έχουν πρόσβαση στο σύστημα και τα tag id των καρτών που τους αντιστοιχούν, καθώς και καταγραφή χρόνων παραμονής των χρηστών στην ελεγχόμενη ζώνη. Ο Διαχειριστής του συστήματος έχει εύκολη πρόσβαση στην πλατφόρμα και δύναται άμεσα να ανακτήσει δεδομένα και να πραγματοποιήσει απαιτούμενες αλλαγές και τροποποιήσεις στο σύστημα πρόσβασης.

Το υλοποιημένο ολοκληρωμένο σύστημα πιστοποίησης ταυτότητας πολύ χαμηλού κόστους εγκαταστάθηκε και δοκιμάστηκε στο Γραφείο Έρευνας και Τεχνολογικών Εξελίξεων Ναυτικού (ΓΕΤΕΝ) και απέδωσε άριστα.

Μελλοντικά το σύστημα δύναται να επεκταθεί προσθέτοντας ένα Real Time Clock ώστε να γίνεται ακριβής καταγραφή χρόνων εισόδου και εξόδου χρηστών, αλλά και συγχρονισμός της ώρας στην περίπτωση που γίνει επέκταση του συστήματος σε παραπάνω από ένα σημεία ελέγχου πρόσβασης. Επιπλέον, σε κάθε σημείο ελεγχόμενης πρόσβασης κρίνεται αναγκαία η προσθήκη κάμερας ασφαλείας πάνω από το μέρος που έχει εγκατασταθεί ο RFID Reader για διασφάλιση της ακεραιότητας του συστήματος από εξωτερικές ενέργειες. Ακόμα εσωτερικά μπορεί να αντικατασταθεί το Keypad με ένα Fingerprint Reader για καλύτερη διασφάλιση της ιδιωτικότητας των χρηστών του συστήματος. Τέλος, κρίνεται αναγκαία η χρήση Online UPS για την αδιάλειπτη λειτουργία του συστήματος σε περιπτώσεις διακοπών ρεύματος.

Το σύστημα Arduino κάνοντας χρήση της τεχνολογίας RFID μπορεί να επεκταθεί περαιτέρω και σε άλλους επιχειρηματικούς τομείς, όπως την μέτρηση εμπορευμάτων από μεγάλη απόσταση ή την καταγραφή των προϊόντων στα καροτσάκια των πελατών σε σουπερμάρκετ. Το RFID όπως και όλα τα νέα τεχνολογικά θαύματα ακολουθεί το γνωστό Hype Cycle όπου τώρα στην Αμερική βρίσκεται αρκετά κάτω από την κορυφή του με την Ευρώπη να ακολουθεί και να βρίσκεται ακόμα στην κορυφή.



Εικόνα 64: The Hype Cycle of RFID

Παρ' όλο που η τεχνολογία RFID εμφανίζει αρκετά (προς το παρόν) μειονεκτήματα, θεωρείται ένα πραγματικά αποδοτικό σύστημα. Οι επιχειρήσεις όμως θα πρέπει να επιστήσουν την προσοχή τους σε δυο κυρίως σημεία. Το πρώτο είναι ότι μέσω της χρήσης του RFID σε επίπεδο συστήματος μπορούν να αποκομίσουν σημαντικά οφέλη, ωστόσο προκειμένου να καταφέρουν κάτι τέτοιο πρέπει να προωθηθεί ταχύτατα η στενή συνεργασία μεταξύ επιχειρήσεων – partners του supply chain. Το δεύτερο σημείο το οποίο πρέπει να προσέξουν είναι η αξιολόγηση και η παρακολούθηση της εξέλιξης του RFID, διότι η τεχνολογία του RFID όπως διαφημίζεται σήμερα και ακολουθώντας το Hype Cycle δεν θα συνεχίσει να αποδίδει για πολύ καιρό ακόμη. Επιπλέον η τεχνολογία RFID παρότι παρέχει πολλές χρήσιμες εφαρμογές δεν είναι απόλυτα ώριμη σήμερα ως προς την ασφάλεια. Υπάρχουν πολλές δυνατότητες για την βελτίωση των συστημάτων αυτών οι οποίες όμως αυξάνουν αρκετά το κόστος του RFID tag. Ο κίνδυνος διαρροής εμπιστευτικών πληροφοριών ή κλωνοποίησης είναι υπαρκτός και κρίνεται ιδιαίτερα σημαντικός πχ στα δεδομένα των βιομετρικών διαβατηρίων κλπ.

Όσον αφορά την ελληνική επιχειρηματική σκηνή, θα πρέπει να δοθεί ιδιαίτερο βάρος στην σφαιρική συλλογή επιτυχημένων παραδειγμάτων και πρακτικών επιχειρήσεων από το διεθνή χώρο που θα οδηγήσουν αναπόφευκτα στην ομαλή υιοθέτηση της τεχνολογίας RFID αρχικά από τις μεγάλες ελληνικές επιχειρήσεις και στην συνέχεια στην σταδιακή εξάπλωσή της και στις υπόλοιπες. Στο νέο επιχειρηματικό τοπίο, οι βασικές λειτουργικές δομές των εταιριών οφείλουν να προσαρμοστούν και να εξοπλιστούν κατάλληλα ώστε να λειτουργούν αποτελεσματικά και να υποστηρίζουν την επιχείρηση του σήμερα.

Η ταυτοποίηση με ραδιοσυχνότητες είναι μια ακόμη τεχνολογία, ενταγμένη στην ραγδαία τεχνολογική ανάπτυξη που διέπει την εποχή μας η οποία από την μια πλευρά είναι

απαραίτητη για την οικονομική πρόοδο και την δημόσια ασφάλεια, ενώ από την άλλη εγκυμονεί σοβαρούς κινδύνους για την ιδιωτικότητα και τα προσωπικά δεδομένα απειλώντας το αναφαίρετο δικαίωμα κάθε ατόμου στον πληροφοριακό του αυτοκαθορισμό αυξάνοντας τις δυνατότητες εποπτείας και χειραγώγησης του τόσο ως πολίτη αλλά και ως καταναλωτή. Ο τρόπος χρήσης της νέας τεχνολογίας επιβάλλεται να τεθεί μέσα σε ένα ειδικό νομικό πλαίσιο, με βασικό γνώμονα ότι «τα συστήματα επεξεργασίας» (πχ προσωπικών δεδομένων) υπηρετούν πάνω από όλα τον άνθρωπο. Υπάρχουν πολύ σοβαρά θέματα που αφορούν την προστασία της ιδιωτικότητας (privacy), τα οποία και θα πρέπει να ληφθούν σοβαρά υπόψη κατά την χρήση της τεχνολογίας από τους Νομικούς και τις Αρχές Προστασίας Προσωπικών Δεδομένων (Data Protection Authorities). Οι Αρχές Προστασίας Προσωπικών Δεδομένων των διαφόρων χωρών θα πρέπει να μελετήσουν καλά τις επιπτώσεις από την χρήση της τεχνολογίας RFID και να δημιουργήσουν πλαίσιο αρχών που θα έχει σαν γνώμονα την προστασία της ιδιωτικότητας, των προσωπικών δεδομένων και των πολιτικών ελευθεριών όπου θα ρυθμίζει τις εφαρμογές της τεχνολογίας αυτής και θα συμβάλλει παράλληλα στην περαιτέρω ανάπτυξή της.

Αυτό θα αποτελέσει και το κλειδί της επιτυχίας των συστημάτων RFID.

Βιβλιογραφία

- **Tanenbaum, Andrew S.**, «Δίκτυα Υπολογιστών», Κλειδάριθμος, 2012
- **J.F. Kurose, K.W. Ross**, «Δικτύωση Υπολογιστών», Α. Γκιούρδα, 2008
- **Charles Bell** - Beginning Sensor Networks with Arduino, APRESS, 2013
- **Luke Welling** – PHP and MySQL Web Development, 2009
- **Daniel M. Dobkin** – The RF in RFID Passive UHF, ELSEVIER, 2008
- **www.ietf.org** - The Internet Engineering Task Force-Internet Drafts και RCF pages
- **www.google.com** - Η πιο γνωστή μηχανή αναζήτησης
- **Wikipedia** - The free encyclopedia

Παράρτημα

Παραμετροποίηση Arduino Mega

```
1 #include <SoftwareSerial.h>
2 #include <Ethernet.h>
3 #include <SPI.h>
4 #include <SD.h>
5 #include <Keypad.h>
6 #include <Wire.h>
7 #include <sha1.h>
8 #include <stdlib.h>
9 #include <mysql.h>
10 #include <LiquidCrystal.h>
11 #define greenPin 5
12 #define redPin 6
13 #define speakerPin 7
14 #define lockPin 8
15 #define rxpin 11
16 #define txpin 255
17 #define unlockLength 500
18
19 /* LCD Connections
20
21 LCD - ARDUINO ADK PINS:
22 1->GND
23 2->5V
24 3->GND
25 4(RS)->P30
26 5(R/W)->GND
27 6(E)->P32
28 11->P34
29 12->P36
30 13->P38
31 14->P40
32 LiquidCrystal lcd(rs, enable, d4, d5, d6, d7);
33 */
34
35 LiquidCrystal lcd(30, 32, 34, 36, 38, 40);
36
37 SoftwareSerial rfidreader = SoftwareSerial( rxpin, txpin );
38
39 byte mac_addr[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
40 IPAddress ip(192, 168, 1, 200);
41 IPAddress server_addr(192, 168, 1, 100);
42 EthernetServer server(80);
43 char user[] = "root";
44 char password[] = "my4cc3ss";
```

```

45
46 Connector my_conn;
47
48 // To store incoming serial data
49 int incomingByte = 0;
50
51 File myFile;
52 File webFile;
53 unsigned long time;
54 unsigned long seconds;
55
56 /* Keypad */
57
58 const byte ROWS = 4;    // four rows
59 const byte COLS = 4;    // four columns
60
61 char keys[ROWS][COLS] =
62 {
63   {'1','2','3','A'},
64   {'4','5','6','B'},
65   {'7','8','9','C'},
66   {'*','0','#','D'}
67 };
68
69 byte rowPins[ROWS] = { 43, 45, 47, 49 };    // define row pinouts of the keypad
70 byte colPins[COLS] = { 35, 37, 39, 41 };    // define column pinouts of the keypad
71
72 Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );
73
74 char PIN[4]={'2','5','A','2'};    // secret code
75 char attempt[4]={'0','0','0','0'};    // used for comparison
76
77 int z=0;
78
79 void setup()
80 {
81   Serial.println(" ");
82   Serial.begin(9600);
83   rfidreader.begin(9600);
84
85   // columns, rows of a 16x2 LCD
86   lcd.begin(16,2);
87   // start with a blank screen
88   lcd.clear();
89
90   Ethernet.begin(mac_addr, ip);
91   server.begin();
92
93   Serial.println("RFID Reader Initialized");
94   Serial.println("Ethernet Interface Initialized");
95
96   if (!SD.begin(4))
97   {
98     Serial.println("SD Card Initialization failed!");
99   }
100  Serial.println("SD Card Initialized");
101  Serial.println(" ");
102
103  my_conn.mysql_connect(server_addr, 3306, user, password);
104
105  pinMode (speakerPin, OUTPUT);

```

```

106     pinMode (greenPin, OUTPUT);
107     pinMode (redPin, OUTPUT);
108     pinMode (lockPin, OUTPUT);
109
110     /* the hardware SS pin (53 on the Mega) must be left as an output
111     or the SD library functions will not work. */
112
113     pinMode (53, OUTPUT);
114
115     digitalWrite(greenPin, LOW);
116     digitalWrite(redPin, LOW);
117     digitalWrite(lockPin, LOW);
118
119     Serial.println(" ");
120     Serial.print("Welcome User!");
121     Serial.println(" ");
122     Serial.print("For Login please use your RFID card.");
123     Serial.println(" ");
124     Serial.print("For Logout please Enter your PIN.");
125     Serial.println(" ");
126     Serial.println(" ");
127 }
128
129
130 void loop()
131 {
132     byte i = 0;
133     byte val = 0;
134     byte checksum = 0;
135     byte bytesRead = 0;
136     byte tempByte = 0;
137
138     // "Unique" tags are only 5 bytes but we need an extra byte for the checksum
139     byte tagBytes[6];
140     char tagValue[10];
141
142     time = millis();
143     seconds = (time/1000);
144
145     EthernetClient client = server.available();           // try to get client
146
147     readKeypad();
148
149     // set cursor to column 0, row 0 (the first row)
150     lcd.setCursor(0,0);
151     lcd.print("Plz Enter Pin ..");
152     // set cursor to column 0, row 1
153     lcd.setCursor(0,1);
154     lcd.print("Welcome User!");
155
156     if (client)
157     {
158         // got client?
159         boolean currentLineIsBlank = true;
160         while (client.connected())
161         {
162             if (client.available())           // client data available to read
163             {
164                 char c = client.read();       // read 1 byte (character) from client
165                 // last line of client request is blank and ends with \n
166                 // respond to client only after last line received
167                 if (c == '\n' && currentLineIsBlank)

```



```

167     {
168         // send web page
169         webFile = SD.open("logs.htm");           // open web page file
170         if (webFile)
171         {
172             while(webFile.available())
173             {
174                 client.write(webFile.read()); // send web page to client
175             }
176             webFile.close();
177         }
178         break;
179     }
180
181     // every line of text received from the client ends with \r\n
182     if (c == '\n')
183     {
184         // last character on line of received text
185         // starting new line with next character read
186         currentLineIsBlank = true;
187     }
188     else if (c != '\r')
189     {
190         // a text character was received from client
191         currentLineIsBlank = false;
192     }
193     } // end if (client.available())
194 } // end while (client.connected())
195 delay(1); // give the web browser time to receive the data
196 client.stop(); // close the connection
197 } // end if (client)
198
199
200 // Read the RFID reader by checking for header
201 if((val = rfidreader.read()) == 2)
202 {
203     bytesRead = 0;
204
205     // Read 10 digit code + 2 digit checksum
206     while (bytesRead < 12)
207     {
208         if(rfidreader.available() > 0)
209         {
210             //store value in val
211             val = rfidreader.read();
212
213             // Append the first 10 bytes (0 to 9) to the raw tag value
214             if (bytesRead < 10)
215             {
216                 tagValue[bytesRead] = val;
217             }
218
219             // Check if this is a if STX, ETX, CR or LF before the 10 digit reading is complete
220             if((val == 0x0D)|| (val == 0x0A)|| (val == 0x03)|| (val == 0x02))
221             {
222                 // Stop reading
223                 break;
224             }
225
226             // Convert ascii hex to integer hex value
227             if ((val >= '0') && (val <= '9'))

```

```

228     {
229         val = val - '0';
230     }
231     else if ((val >= 'A') && (val <= 'F'))
232     {
233         val = 10 + val - 'A';
234     }
235
236     // Every two hex-digits, add a byte to the code
237     // So if odd number merge 2 4bit digits into one (8 bit) byte
238     if (bytesRead & 1 == 1)
239     {
240         // Make space for this hex-digit by shifting the previous digit 4 bits to the left
241         tagBytes[bytesRead >> 1] = (val | (tempByte << 4));
242
243         // If we're at the checksum byte,
244         if (bytesRead >> 1 != 5)
245         {
246             // Calculate the checksum... (XOR)
247             checksum ^= tagBytes[bytesRead >> 1];
248         };
249     }
250     else
251     {
252         // Store the first hex digit first
253         tempByte = val;
254     };
255     // Ready to read next digit
256     bytesRead++;
257 }
258 }
259
260 // Send the result to the host connected via USB
261 if (bytesRead == 12)
262 // 12 digit read is complete
263 {
264     // We have reached the end of all 12 digits and read is complete, so add a null terminator
265     tagValue[10] = '\0';
266     Serial.print("Tag read: ");
267     for (i=0; i<5; i++)
268     {
269         // Add a leading 0 to pad out values below 16
270         if (tagBytes[i] < 16)
271         {
272             Serial.print("0");
273         }
274         Serial.print(tagBytes[i], HEX);
275     }
276     Serial.println();
277     Serial.print("Checksum: ");
278     Serial.print(tagBytes[5], HEX);
279     Serial.println(tagBytes[5] == checksum ? " -- passed." : " -- error.");
280
281     // Show the raw tag value
282     Serial.print("VALUE: ");
283     Serial.println(tagValue);
284
285     // Search the tag database for this particular tag
286     int tagId = findTag( tagValue );
287
288     // Only fire the strike plate if this tag was found in the database

```

```

289     if( tagId > 0 )
290     {
291         Serial.print("Time: ");
292         Serial.println(seconds);
293         Serial.print("Authorized tag ID");
294         Serial.println(" ");
295         Serial.print("Login done!");
296         Serial.println(" ");
297
298         // unlock the door
299         unlock();
300
301         // open and write tagValue to log.txt
302         myFile = SD.open("logs.htm", FILE_WRITE);
303         if (myFile)
304         {
305             myFile.print("Time: ");
306             myFile.println(seconds);
307             myFile.print("VALUE: ");
308             myFile.println(tagValue);
309             myFile.print("Authorized tag ID");
310             myFile.println(" ");
311             myFile.println("Login done!");
312             myFile.println(" ");
313             myFile.close();
314         }
315     }
316     else
317     {
318         Serial.print("Time: ");
319         Serial.println(seconds);
320         Serial.print("Tag ID not Authorized!");
321         Serial.println();
322
323         // door stays locked
324         lock();
325
326         // open and write tagValue to log.txt
327         myFile = SD.open("logs.htm", FILE_WRITE);
328         if (myFile)
329         {
330             myFile.print("Time: ");
331             myFile.println(seconds);
332             myFile.print("VALUE: ");
333             myFile.println(tagValue);
334             myFile.print("Tag ID not Authorized!");
335             myFile.println(" ");
336             myFile.println(" ");
337             myFile.close();
338         }
339     }
340
341     // Blank separator line in output
342     Serial.println();
343     myFile.println();
344 }
345 bytesRead = 0;
346 }
347 }
348
349

```

```

350 /* Search for a specific tag in the database */
351 int findTag( char tagValue[10] )
352 {
353     char QUERY_POP[64];
354     sprintf(QUERY_POP,"SELECT id FROM door.tag WHERE id = '%s'",tagValue);
355     int rows = do_query(QUERY_POP);
356     if (rows>0)
357     {
358         Serial.println("We found a match!");
359         return(1);
360     }
361     else
362     {
363         Serial.println("No match.");
364         return(0);
365     }
366 }
367 int do_query(const char *q)
368 {
369     // pointer to column values
370     column_names *c;
371
372     // pointer to row values
373     row_values *r;
374
375     // First, execute query. If it returns a value pointer,
376     // we have a result set to process. If not, we exit.
377     if (!my_conn.cmd_query(q))
378     {
379         return 0;
380     }
381
382     // Next, we read the column names and display them.
383     //
384     // NOTICE: You must *always* read the column names even if
385     //           you do not use them. This is so the connector can
386     //           read the data out of the buffer. Row data follows the
387     //           column data and thus must be read first.
388
389     c = my_conn.get_columns();
390     for (int i = 0; i < c->num_fields; i++)
391     {
392         Serial.print(c->fields[i]->name);
393         if (i < c->num_fields - 1)
394         {
395             Serial.print(",");
396         }
397     }
398     Serial.println();
399
400     // We use the get_next_row() iterator and read rows printing
401     // the values returned until the get_next_row() returns NULL.
402
403     int num_cols = c->num_fields;
404     int rows = 0;
405     do
406     {
407         r = my_conn.get_next_row();
408         if (r)
409         {
410             rows++;

```

```

411         for (int i = 0; i < num_cols; i++)
412         {
413             Serial.print(r->values[i]);
414             if (i < num_cols - 1)
415             {
416                 Serial.print(", ");
417             }
418         }
419         Serial.println();
420
421         // We free the row read to free the memory allocated for it.
422         my_conn.free_row_buffer();
423     }
424 } while (r);
425 Serial.print(rows);
426 Serial.println(" rows in result.");
427
428 // Finally, we are done so we free the column buffers
429 my_conn.free_columns_buffer();
430 return rows;
431 }
432
433
434
435 /* If an authorised tag was found in the database unlock the door */
436 void unlock()
437 {
438     tone(speakerPin, 1000, 500);
439     digitalWrite(greenPin, HIGH);
440     delay(1000);
441     digitalWrite(greenPin, LOW);
442     digitalWrite(lockPin, HIGH);
443     delay(unlockLength);
444     digitalWrite(lockPin, LOW);
445 }
446
447
448 /* If an unauthorised tag was found do the following */
449 void lock()
450 {
451     tone(speakerPin, 250, 250);
452     delay(250);
453     tone(speakerPin, 150, 250);
454     digitalWrite(redPin, HIGH);
455     delay(1000);
456     digitalWrite(redPin, LOW);
457 }
458
459
460 void readKeypad()
461 {
462     char key = keypad.getKey();
463     if (key != NO_KEY)
464     {
465         attempt[z]=key;
466         z++;
467         switch(key)
468         {
469             case '*':
470                 z=0;
471                 break;

```

```

472         case '#':
473             z=0;
474             delay(100);
475             checkPIN();
476             break;
477         }
478     }
479 }
480
481
482 // Pin Checked
483 void checkPIN()
484 {
485     int correct=0;
486
487     for (int i = 0; i < 4 ; i++ )
488     {
489         if (attempt[i]==PIN[i])
490         {
491             correct++;
492         }
493     }
494
495     if (correct==4)
496     {
497         correctPIN();
498     }
499
500     else
501     {
502         incorrectPIN();
503     }
504
505     for (int j=0; j<4; j++)
506     {
507         attempt[j]='0';
508     }
509 }
510
511
512 // Do this if correct PIN entered
513 void correctPIN()
514 {
515     // start with a blank screen
516     lcd.clear();
517     // set cursor to column 0, row 0 (the first row)
518     lcd.setCursor(0,0);
519     lcd.print("* Logout Done! *");
520     // set cursor to column 0, row 1
521     lcd.setCursor(0,1);
522     lcd.print("* Correct PIN *");
523
524     Serial.print("Time: ");
525     Serial.println(seconds);
526     Serial.print("* Correct PIN *");
527     Serial.println(" ");
528     Serial.print("* Logout done! *");
529     Serial.println(" ");
530     Serial.println(" ");
531
532     myFile = SD.open("logs.htm", FILE_WRITE);

```

```

533     if (myFile)
534     {
535         myFile.print("Time: ");
536         myFile.println(seconds);
537         myFile.print("* Correct PIN *");
538         myFile.println(" ");
539         myFile.print("* Logout done! *");
540         myFile.println(" ");
541         myFile.println(" ");
542         myFile.close();
543     }
544
545     tone(speakerPin, 1000, 500);
546     digitalWrite(greenPin, HIGH);
547     delay(1000);
548     digitalWrite(greenPin, LOW);
549     digitalWrite(lockPin, HIGH);
550     delay(unlockLength);
551     digitalWrite(lockPin, LOW);
552
553     // start with a blank screen
554     lcd.clear();
555 }
556
557
558 // Do this if incorrect PIN entered
559 void incorrectPIN()
560 {
561     // start with a blank screen
562     lcd.clear();
563     // set cursor to column 0, row 0 (the first row)
564     lcd.setCursor(0,0);
565     lcd.print("* Try again *");
566     // set cursor to column 0, row 1
567     lcd.setCursor(0,1);
568     lcd.print("* Invalid PIN *");
569
570     Serial.print("Time: ");
571     Serial.println(seconds);
572     Serial.print("* Incorrect PIN *");
573     Serial.println(" ");
574     Serial.print("* Try again *");
575     Serial.println(" ");
576     Serial.println(" ");
577
578     myFile = SD.open("logs.htm", FILE_WRITE);
579     if (myFile)
580     {
581         myFile.print("Time: ");
582         myFile.println(seconds);
583         myFile.print("* Incorrect PIN *");
584         myFile.println(" ");
585         myFile.print("* Try again *");
586         myFile.println(" ");
587         myFile.println(" ");
588         myFile.close();
589     }
590
591     tone(speakerPin, 250, 250);
592     delay(250);
593     tone(speakerPin, 150, 250);

```

```
594     digitalWrite(redPin, HIGH);
595     delay(1000);
596     digitalWrite(redPin, LOW);
597
598     // start with a blank screen
599     lcd.clear();
600 }
```