

Πανεπιστήμιο Θεσσαλίας



ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ ΔΙΚΤΥΩΝ

**Ανάπτυξη πρωτοκόλλου για συσκευές RFID Readers
σύμφωνα με την έκδοση του προτύπου της EPCglobal:**

Reader Protocol Standard Version 1.1

του

Μπότη Ιωάννη

Ευχαριστίες

Καταρχήν, θα ήθελα να ευχαριστήσω τους γονείς μου για την υποστήριξη τους όλα αυτά τα χρόνια των σπουδών μου και ιδιαίτερα τους ευχαριστήσω για την υπομονή και την βοήθεια τους όλα αυτά τα χρόνια. Επιπλέον, ένα μεγάλο ευχαριστώ στους φίλους και συμφοιτητές μου για την αλληλοϋποστήριξη και τα αξέχαστα χρόνια που περάσαμε.

Θα ήθελα επίσης να ευχαριστήσω τον καθηγητή κ. Γεώργιο Σταμούλη, επιβλέποντα της διπλωματικής μου, για τη στήριξη και τη βοήθεια που μου παρείχε αυτά τα πέντε χρόνια των σπουδών μου.

Περιεχόμενα

Εισαγωγή	6
1. Ιστορία	7
1.1 Ανάγκη για σήμανση στα αγαθά	7
1.2 Barcode, απλό και εύχρηστο	8
1.3 Ταυτοποίηση μέσω ραδιοσυχνοτήτων, RFID	9
1.3.1 Οι πρώτες προσπάθειες	9
1.3.2 Οι πρώτες πατέντες	10
1.3.3 Εθνικό Εργαστήριο στο Los Alamos των ΗΠΑ	10
1.3.4 Εφαρμογές RFID σε ευρεία κλίμακα	11
1.3.5 Το πρότυπο EPC	11
2. Βασικά μέρη ενός συστήματος RFID	13
2.1 Εισαγωγή	13
2.2 Tags	14
2.3 Antennas ή Barcode scanners	14
2.4 Reader Device	14
2.5 Host	14
3. Ανάγκη για πρότυπα	15
3.1 Χωρίς κανόνες	15
3.2 Τα πρώτα πρότυπα	15
3.3 Οργανισμοί πιστοποίησης και προτύπων	16
3.3.1 International Standards Organization ISO	16
3.3.2 EPCglobal	16
3.4 Κατηγορίες προτύπων	17
3.4.1 Air Interface	17
3.4.2 Data content&encoding	17
3.4.3 Testing of RFID systems	18
3.4.4 Interoperability	18
3.4 Μοντέλο EPCglobal	19
4. Reader Protocol	20
4.1 Εισαγωγή	20
4.2 Επίπεδα Πρωτοκόλλου	20
4.3 Κανάλια Επικοινωνίας	21
5. Reader Layer – Γενική ιδέα	21
5.1 Read Subsystem	23
5.1.1 Sources	23
5.1.2 ReadTrigger	23
5.1.3 Data Acquisition Stage	23
5.1.4 Read Filter Stage	24
5.2 Event Subsystem	24
5.2.1 Smoothing/Event Generation Stage	24
5.3 Output Subsystem	26
5.3.1 Data Selector Stage	26
5.3.2 Report Buffer Stage	26
5.3.3 Notification Trigger	26
5.4 Communication Subsystem – Message/Transport Binding Stage	26

6. Object Model	27
6.1 ReaderDevice Object	28
6.2 ReadPoints	28
6.3 Sources	28
6.4 Triggers	29
6.5 TagSelectors	29
6.5.1 Πολλαπλοί TagSelectors	29
6.6 Channels	29
6.7 DataSelectors	30
6.8 TagFields	30
6.9 ReadReport	30
6.10 ReportBuffer	30
6.11 IOPort	30
6.12 Σύνοψη	30
7. Υλοποίηση σε Java	31
7.1 Περιβάλλον και πρόγραμμα ανάπτυξης	31
7.2 Βιβλιοθήκες που χρησιμοποιήθηκαν	31
7.3 Υλοποίηση προγράμματος Client στο Host	31
7.4 Άλλες υλοποιήσεις ανοικτού κώδικα	31
8. Δίκτυο EPCglobal	32
8.1 Παραγωγή ενός προϊόντος	32
8.2 Αλυσίδα διάθεσης προϊόντος	32
8.3 Πώληση προϊόντος	33
8.4 Πλεονεκτήματα σε αντιδιαστολή με το barcode	33
9. Εκτέλεση σεναρίου λειτουργίας	34
9.1 ReaderDevice, CommandChannel και ReadPoints	34
9.2 Sources	34
9.3 TagField	35
9.4 TagSelectors	35
9.5 DataSelector	36
9.6 NotificationChannels	36
9.7 Triggers	36
9.8 Διάγραμμα αντικειμένων σεναρίου	37
9.9 Έναρξη σεναρίου – 1η φάση	38
9.10 Αποστολή events στον Host – 2η φάση	40
9.11 Ροή πληροφορίας	41
10. Messaging / Transport Binding(MTB)	42
10.1 Messaging Layer	42
10.1.1 Text	42
10.1.2 XML	42
10.2 Transport Layer	43
10.2.1 Serial	44
10.2.2 Tcp	44
10.2.3 Http	44

<i>10.3 Σενάριο εντολής</i>	45
<i>Γλωσσάριο</i>	47
<i>Πηγές</i>	48

Εισαγωγή

Ο τομέας των RFID έχει μπει για τα καλά στη ζωή μας τα τελευταία χρόνια. Όλο και περισσότερες εταιρίες επιλέγουν να επενδύσουν σε εφαρμογές και λύσεις που μπορεί να προσφέρει η νέα αυτή τεχνολογία. Ήδη σε πολλές πτυχές της καθημερινής ζωής μας χρησιμοποιούμε συστήματα RFID χωρίς να το γνωρίζουμε.

Στο παρών έγγραφο αρχικά θα γίνει μια σύντομη περιγραφή της τεχνολογίας των RFID και της ιστορίας της, η περιγραφή του προτύπου του πρωτοκόλλου και της υλοποίησης που έχει αναπτυχθεί. Θα περιγραφούν οι ανάγκες που οδήγησαν στην ανάπτυξη του πρωτοκόλλου, πως μπορεί να χρησιμοποιηθεί και να εφαρμοστεί από εταιρίες κατασκευής συσκευών, από προγραμματιστές εφαρμογών για τις συσκευές αυτές και από ερευνητές. Τέλος θα παρουσιαστεί ένα σενάριο λειτουργίας του πρωτοκόλλου.

Ο όρος RFID (Radio Frequency Identification) αντιστοιχεί στη ταυτοποίηση μέσω ραδιοσυχνοτήτων σε ελεύθερη μετάφραση. Αρχικός σκοπός είναι να περιγραφεί το γενικό μοντέλο της ιδέας αυτής και να παρουσιαστούν κάποιες από τις εφαρμογές ώστε να γίνει κατανοητή η χρήση αυτής της τεχνολογίας. Βασικός σκοπός είναι να περιγραφεί διεξοδικά το πρωτόκολλο, να παρουσιαστεί η υλοποίηση που έχει αναπτυχθεί και το μοντέλο που καλύπτει.

Το πρωτόκολλο που υλοποιήθηκε έχει εκδοθεί από τον οργανισμό EPCglobal (ακρωνύμιο του Electronic Product Code global, παγκόσμιος Ηλεκτρονικός Κώδικας Προϊόντος σε μετάφραση) του οποίου σκοπός είναι η ανάπτυξη και η προώθηση καθολικών πρωτοκόλλων για τα προϊόντα της τεχνολογίας RFID, από τις ετικέτες ταυτοποίησης (από δω και πέρα θα αναφερόμαστε σ' αυτές ως tags) και τις συσκευές ανάγνωσης/εγγραφής (Reader Device) μέχρι πληροφοριακά συστήματα βάσεων δεδομένων στο διαδίκτυο με πληροφορίες για καθένα κωδικό tag(EPCIS).

Στην παρούσα εργασία υλοποιήθηκε το πρωτόκολλο ανάγνωσης για συσκευές Reader Device, το οποίο αποτελεί ένα πρότυπο διασύνδεσης που προσδιορίζει τις αλληλεπιδράσεις μεταξύ μιας συσκευής ανάγνωσης/εγγραφής ετικετών και των λογισμικών εφαρμογής. Ο στόχος είναι να οριστεί μια ανοιχτή και επεκτάσιμη διασύνδεση που κατασκευαστές Reader Device θα μπορούν να υλοποιούν, υποστηρίζοντας περισσότερες λειτουργίες με ένα πρότυπο (κοινό-καθολικό) τρόπο, χωρίς όμως να υπαγορεύεται η εφαρμογή ούτε να αποκλείεται η επέκταση και η καινοτομία της συσκευής από τον κατασκευαστή. Όπως θα παρουσιαστεί εκτενώς στη συνέχεια της εργασίας, το πρότυπο του πρωτοκόλλου είναι αντικειμενοστραφές, δηλαδή αποτελείται από αντικείμενα που αντιστοιχούν σε φυσικές και λογικές οντότητες της συσκευής. Κάθε αντικείμενο διαθέτει τις δικές του μεταβλητές και συναρτήσεις που πρέπει ή μπορεί να υποστηρίζει.

Η υλοποιούσα έκδοση του προτύπου είναι η 1.1(Reader Protocol Standard Version 1.1), η οποία έχει επικυρωθεί και έχει τεθεί στη διάθεση του κοινού τον Αύγουστο του 2006. Αξιοσημείωτα χαρακτηριστικά του περιλαμβάνουν εντολές για να διαβάσει, να γράφει και να καταστρέφει ετικέτες, την πρόσβαση σε «Μνήμη χρήστη» ετικέτας, καθώς και πληροφορίες ταυτότητας, εκτεταμένες ρυθμίσεις που αφορούν τις εντολές, πλούσιες επιλογές αναφοράς, ασύγχρονη ειδοποίηση, πολλαπλά Πρωτόκολλα Μεταφοράς Μηνυμάτων(Message Transport Bindings MTB) με επικοινωνία μέσω serial, TCP είτε HTTP, και XML ή μορφή μηνυμάτων κειμένου(Text), και μηχανισμούς επεκτασιμότητας.

Ας αναφερθεί εδώ ότι δεν υπάρχει έκδοση 1.0. Ένα πολύ πρώιμο σχέδιο της εργασίας-σε-εξέλιξη κυκλοφόρησε στο διαδίκτυο με το παρατσούκλι "1.0", αλλά το έγγραφο ήταν ένα πολύ πρώιμο αρχείο, που δεν περιλάμβανε τα περισσότερα από τα σημεία του σχεδιασμού, ακόμη και την εποχή εκείνη. Η ομάδα εργασίας επέλεξε να αλλάξει τον αριθμό έκδοσης σε 1.1 για να αποφευχθεί η σύγχυση με αυτό το πρώιμο έγγραφο, και να αντικατοπτρίζει τις σημαντικές προόδους από την προηγούμενη εργασία.

1. Ιστορία

1.1 Ανάγκη για σήμανση στα αγαθά.

Οι βασικές ανάγκες στις οποίες κλήθηκε να απαντήσει η τεχνολογία RFID (ακρωνύμιο του Radio Frequency Identification) είναι εκείνες της ταυτοποίησης και κατηγοροποίησης προϊόντων και αγαθών. Από πολύ παλιά, από την αρχαιότητα με την απαρχή του εμπορίου, χρησιμοποιήθηκαν διάφοροι τρόποι για την επίτευξη των παραπάνω στόχων. Η πλέον διαδεδομένη μέθοδος στον χώρο του εμπορίου είναι η τεχνική της ετικέτας (Label) προϊόντων για να διαπιστώσουμε το είδος και τα χαρακτηριστικά που μας ενδιαφέρουν ενός αγαθού.



Σχήμα 1.1 Ταυτοποίηση προϊόντων στην αρχαιότητα

Με την είσοδο στην εποχή των υπολογιστών, δημιουργήθηκε η ανάγκη για αυτόματη ταυτοποίηση προϊόντων από υπολογιστικά συστήματα. Για μια ακόμη φορά χρησιμοποιήθηκε η μέθοδος της ετικέτας. Έπρεπε όμως να δημιουργηθεί μία ετικέτα που να μπορεί να αναγνωρίζεται από τα υπάρχοντα υπολογιστικά συστήματα γρήγορα και με το ελάχιστο δυνατό κόστος. Διάφορες εταιρίες είχαν αναπτύξει το δικό τους σύστημα αυτόματης ταυτοποίησης που περιοριζόταν στη δική τους αλυσίδα παραγωγής και διακίνησης προϊόντων. Από απλές ετικέτες που περιείχαν τίποτα περισσότερο από τον αριθμό ή το όνομα του προϊόντος, μέχρι ετικέτες που περιείχαν πρόσθετες πληροφορίες όπως χρόνο παραγωγής, βάρος ή οτιδήποτε χρήσιμο, η μορφή και η τεχνολογία των ετικετών και των υπολογιστικών συστημάτων που χρησιμοποιούνταν ποίκιλε ανάλογα με τις ανάγκες της κάθε εταιρίας.

Τα παραπάνω όμως δημιούργησαν μια σειρά ερωτημάτων και προβλημάτων που έπρεπε να αντιμετωπιστούν. Η βασική πρόκληση ήταν η ανάπτυξη ενός εύκολου στη χρήση, αξιόπιστου και καθολικού τρόπου σήμανσης των αγαθών. Έτσι το ερώτημα που προέκυψε ήταν αν υπάρχει κάποιο τέτοιου είδους σύστημα, το οποίο να μπορεί να χρησιμοποιηθεί σε μια σειρά από διαφορετικές εφαρμογές και ταυτόχρονα να είναι απλό και αξιόπιστο στη χρήση του. Το πρώτο σύστημα που υιοθετήθηκε σε καθολική κλίμακα και έγινε προάγγελος των RFID συστημάτων ήταν το barcode. Ας πάρουμε όμως τα γεγονότα με τη σειρά.

1.2 Barcode, απλό και εύχρηστο.

Στην Αμερική ακριβώς μετά το τέλος του 2ου παγκοσμίου πολέμου είχε αρχίσει να συντελείται μια ραγδαία άνοδος στους τομείς της παραγωγικότητας, της κατανάλωσης και του εμπορίου. Ο αριθμός των προϊόντων και των υπηρεσιών που προσφέρονταν στους καταναλωτές ήταν ο υψηλότερος στην ανθρώπινη ιστορία και σε συνδυασμό με τη ραγδαία ανάπτυξη των ηλεκτρονικών συστημάτων δημιούργησε την ανάγκη για ένα αυτόματο αξιόπιστο σύστημα ταυτοποίησης. Σημαντική προσπάθεια για να δώσει απάντηση στην παραπάνω ανάγκη ήταν η εφεύρεση του Barcode ή γραμμωτού κώδικα από έναν πτυχιούχο φοιτητή του Πανεπιστημίου της Φιλαδέλφειας των ΗΠΑ μετά από την παράκληση του προέδρου της τοπικής αγοράς προϊόντων διατροφής προκειμένου να καλύψει τις ανάγκες των εκεί σούπερ μάρκετ για γρηγορότερη διακίνηση ("πέραςμα") των αγαθών από τα ταμεία. Στην προ barcode εποχή, οι ταμίες ήταν αναγκασμένοι να πληκτρολογούν στην ταμειακή μηχανή την τιμή του κάθε προϊόντος που επρόκειτο να αγοραστεί από τον πελάτη, με αποτέλεσμα τη δημιουργία ουρών, σφαλμάτων και τη δαπάνη πολύτιμου χρόνου. Μετά από μια σειρά ερευνών και δοκιμών σε καταστήματα, άρχισαν να γίνονται εμφανή τα πλεονεκτήματα της νέας τεχνολογίας τα οποία διαπιστώθηκε ότι ήταν περισσότερα από ότι είχε προβλεφθεί.

Έτσι όλο και περισσότερες εταιρίες και καταστήματα άρχισαν να χρησιμοποιούν το νέο σύστημα, μαρκάροντας τα προϊόντα τους με την νέα "ετικέτα". Ο τρόπος αναπαράστασης του γραμμωτού κώδικα αντιστοιχούσε σε ένα αριθμό που διάβαζε αυτόματα με οπτικό δέκτη και παρείχε το είδος του προϊόντος, την εταιρία παραγωγής ή άλλες χρήσιμες πληροφορίες, όπως την τιμή του. Κάθε λευκή και μαύρη γραμμή "μεταφράζεται" σε σε δυαδικό ψηφίο, 0 ή 1. Το πρότυπο που υιοθετήθηκε ήταν του 12-ψηφίου καθολικού γραμμωτού κώδικα, του ίδιου ακριβώς γνωστού έως σήμερα barcode και για να γίνει καθολικό, σε κάθε είδος προϊόντος αντιστοιχούσε ένας μοναδικός κωδικός (Universal Product Code (UPC)) ενώ η ετικέτα του barcode συνήθως "έμπαινε" στη γραμμή παραγωγής του προϊόντος και χρησιμοποιούνταν στις ταμιακές μηχανές και αργότερα σε όλη την αλυσίδα διάθεσής του. Από τρόφιμα και ρούχα μέχρι οπτικά συστήματα όλα άρχισαν να "μαρκάρονται" με barcode διευκολύνοντας τους τομείς της παραγωγής, διάθεσης και πώλησης των εταιριών και των καταστημάτων. Μέχρι τις αρχές της δεκαετίας του '90 το νέο πρότυπο σύστημα ταυτοποίησης ή κάποια παραλλαγή του χρησιμοποιούνταν ήδη από σχεδόν όλες τις επιχειρήσεις στις ΗΠΑ και είχε επεκταθεί σε πολλές άλλες χώρες του κόσμου.

Το 1974 δημιουργήθηκαν ο οργανισμός Uniform Code Council (Συμβούλιο Καθολικού Κώδικα), που είναι υπεύθυνος για την προώθηση του πρότυπου γραμμωτού κώδικα UPC, και ο αντίστοιχος ευρωπαϊκός οργανισμός International EAN, υπεύθυνος για τον γραμμωτό κώδικα στην Ευρώπη, οποίος ανέπτυξε τα δικά του πρότυπα για την ευρωπαϊκή ήπειρο.



Σχήμα 1.2 Γραμμωτός κώδικας UPC

1.3 Ταυτοποίηση μέσω ραδιοσυχνοτήτων, RFID

Παρότι το σύστημα Barcode αποδείχθηκε ιδιαίτερα αξιόπιστο και απλό στη χρήση, είτε σαν UPC είτε σαν άλλες μορφές στις οποίες μπορούσες να αποθηκεύεις περισσότερα στοιχεία, άρχισε να γίνεται εμφανές ότι με την πρόοδο των Η/Υ υπήρχε ανάγκη για κάτι που θα έδινε τη δυνατότητα για περισσότερες εφαρμογές. Ενώ η τεχνολογία της ταυτοποίησης μέσω ραδιοσυχνοτήτων(RFID) είναι όσο παλιά όσο αυτή του Barcode, μόλις πρόσφατα έγινε δυνατό η χρήση της σε ευρεία κλίμακα στην παραγωγή και το εμπόριο. Οι πρώτες εφαρμογές στο εμπόριο ήταν για ηλεκτρονικά συστήματα παρακολούθησης και πρόσβασης προσωπικού.

Σταθμός στην δημιουργία θεωρητικού υποβάθρου για την ανάπτυξη αυτής της τεχνολογίας είναι η εργασία “Communication by Means of Reflected Power”,σε μετάφραση “Επικοινωνία διαμέσου ανακλώμενης ενέργειας” του ερευνητή Harry Stockman, που εκδόθηκε το 1948. Στην εργασία αυτή παρουσιάζει ένα νέο τρόπο επικοινωνίας όπου για την αποστολή πληροφορίας περιγράφονται διάφορα μοντέλα, με χρήση ραδιοκυμάτων, φωτός ή ήχου, και συνδέεται με τη λειτουργία των ραντάρ, από όπου δανείζεται αρχές και θεωρία για να γίνει πιο εύκολα κατανοητή. Παρότι στην εργασία γινόταν αναφορά για πιθανές εφαρμογές του νέου μοντέλου επικοινωνίας, ήταν πολύ νωρίς ακόμα για εφαρμογές σε μεγάλη κλίμακα.

1.3.1 Οι πρώτες προσπάθειες

Οι πρώτες δύο αξιοσημείωτες εφαρμογές ήταν το σύστημα αναγνώρισης φίλου ή εχθρού, Identification friend or foe(IFF), που χρησιμοποιήθηκε στα αεροσκάφη ήδη από το 2ο παγκόσμιο πόλεμο και έδωσε τη δυνατότητα να αναγνωρίζονται τα φιλικά από τα εχθρικά αεροσκάφη και η ηλεκτρονική παρακολούθηση αντικειμένου,electronic article surveillance(EAS) στα μέσα της δεκαετίας του '60 που χρησιμοποιήθηκε από καταστήματα πώλησης για την αντιμετώπιση της κλοπής των εμπορευμάτων. Το μεν πρώτο(IFF) ήταν ένα αρκετά πολύπλοκο σύστημα επικοινωνίας με κρυπτογράφηση δεδομένων ενώ το δεύτερο δεν ήταν τίποτε περισσότερο από ετικέτες 1 δυαδικού ψηφίου, που σήμαινε αν το προϊόν είχε αγοραστεί ή όχι. Και τα δύο συστήματα συνεχίζουν να χρησιμοποιούνται μέχρι σήμερα.



Σχήμα 1.3 Αριστερά ένας κόμβος EAS,στη μέση μία ετικέτα(tag) EAS, δεξιά ένα σύστημα IFF

Άλλες εφαρμογές που παρουσιάστηκαν στα μέσα της δεκαετίας του 1970 ήταν συστήματα ελέγχου της πρόσβασης προσωπικού,στα οποία κάθε εργαζόμενος προμηθευόταν μία κάρτα με ένα tag ενσωματωμένο σ' αυτή και η πρόσβαση σε χώρους, αντικείμενα ή πόρους γινόταν μέσω ενός συστήματος αυθεντικοποίησης, και συστήματα παρακολούθησης ζώων, στα οποία είχε προσδεθεί ένα tag και η αναγνώρισή τους μπορούσε να γίνει αυτόματα και από απόσταση.

Παρά την όποια πρόοδο και τις όποιες εφαρμογές είχαν παρουσιαστεί και με την φθηνή απλούστερη και πιο κατανοητή μέθοδο του Barcode να έχει πλέον διαδοθεί, η μόνη ευρέως διαδεδομένη εμπορική χρήση RFID μέχρι την αρχή του 1980 ήταν το EAS. Αν και το τελευταίο με μεγάλη δυσκολία θα μπορούσε να χαρακτηριστεί ως σύστημα RFID,καθώς δεν παρείχε αναγνώριση αντικειμένου,αλλά μόνο μια σήμανση για το αν είχε πληρωθεί.

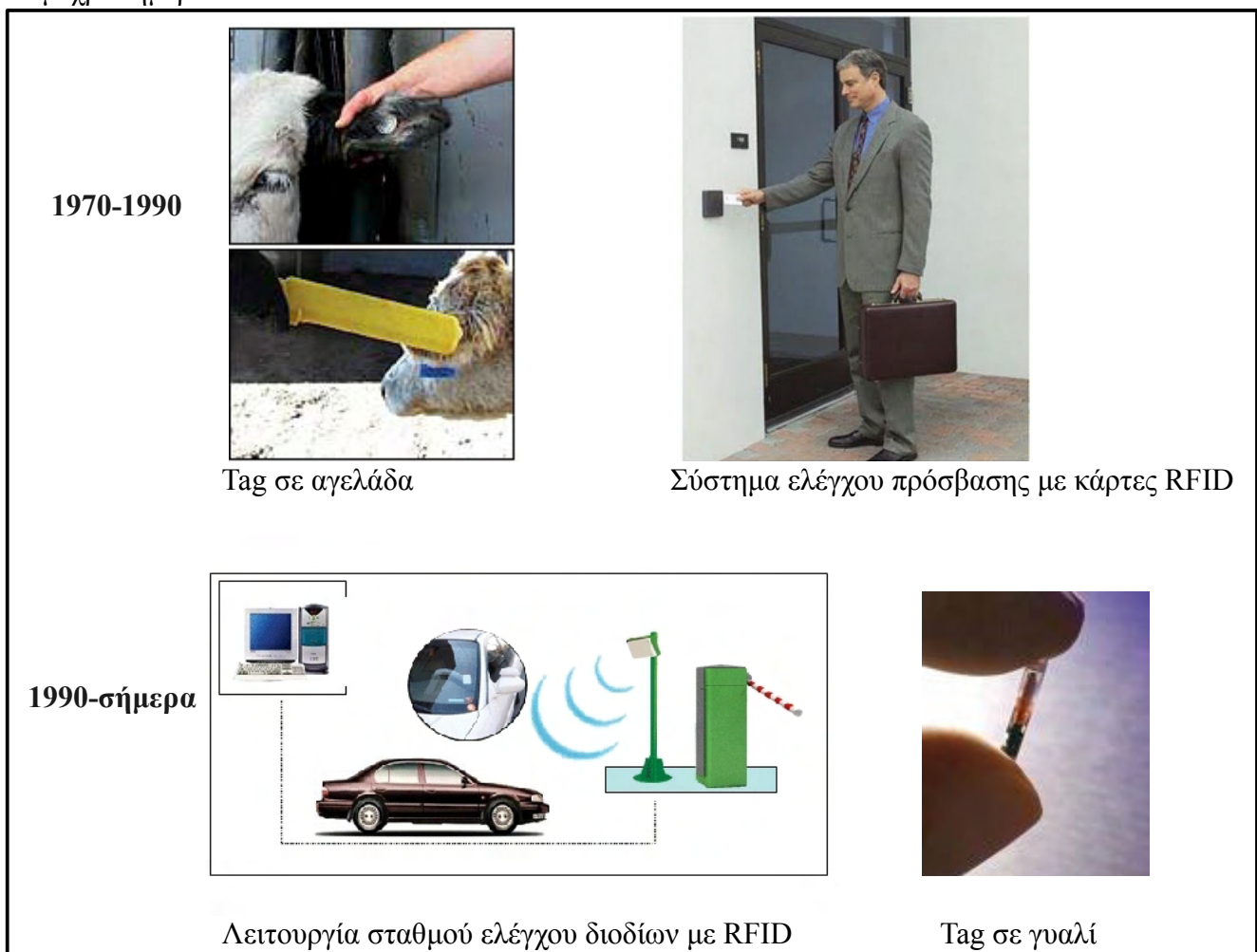
1.3.2 Οι πρώτες πατέντες

Ο ερευνητής Mario W. Cardullo έλαβε την πρώτη πατέντα στις ΗΠΑ για ένα tag με επανεγγράψιμη μνήμη στις 23 Ιανουαρίου 1973, ενώ τον ίδιο χρόνο ο επιχειρηματίας Charles Walton από την Καλιφόρνια των ΗΠΑ έλαβε μία άλλη πατέντα για ένα σύστημα πρόσβασης με κάρτες που άνοιγε μια πόρτα χωρίς κλειδί, όταν η συσκευή ανάγνωσης της πόρτας λάμβανε ένα έγκυρο αναγνωριστικό αριθμό που ήταν αποθηκευμένος μέσα στην κάρτα.

1.3.3 Εθνικό Εργαστήριο στο Los Alamos των ΗΠΑ

Σημαντική συνεισφορά στην ανάπτυξη της τεχνολογίας προέρχεται από το εθνικό εργαστήριο στο Los Alamos στις ΗΠΑ. Το 1970 το Υπουργείο Ενέργειας των ΗΠΑ απηύθυνε στο εργαστήριο το αίτημα για την ανάπτυξη ενός αξιόπιστου συστήματος παρακολούθησης των ραδιενεργών αποβλήτων. Μία ομάδα επιστημόνων του εργαστηρίου κατέληξε στη δημιουργία ενός συστήματος RFID, όπου ένα tag, που τοποθετούνταν σε κάθε φορητό μεταφοράς και λειτουργούσε σαν αναμεταδότης, αποκρινόταν στις συσκευές ανάγνωσης στις εισόδους τέτοιων εγκαταστάσεων. Αργότερα το ίδιο ακριβώς σύστημα θα χρησιμοποιούνταν σε σταθμούς διοδίων.

Ταυτόχρονα, μετά από αίτημα του Υπουργείου Γεωργίας των ΗΠΑ, το εργαστήριο ανέπτυξε ένα σύστημα RFID για την παρακολούθηση αγελάδων. Κάθε αγελάδα μαρκάρωνταν με ένα tag που περιείχε το μοναδικό αναγνωριστικό της και με τη χρήση Η/Υ ήταν δυνατό να κρατούνται δεδομένα για την κατάσταση της υγείας της και της ζωής της, μειώνοντας τον ανθρώπινο παράγοντα και ως εκ τούτου τα λάθη. Αργότερα, εταιρίες ανέπτυξαν ένα σύστημα με μικρότερα tag τοποθετημένα σε γυαλί που εμφυτεύονταν κάτω από το δέρμα των ζώων, το οποίο χρησιμοποιείται μέχρι σήμερα.



Σχήμα 1.4 Πρώτες εφαρμογές συστημάτων RFID

1.3.4 Εφαρμογές RFID σε ευρεία κλίμακα

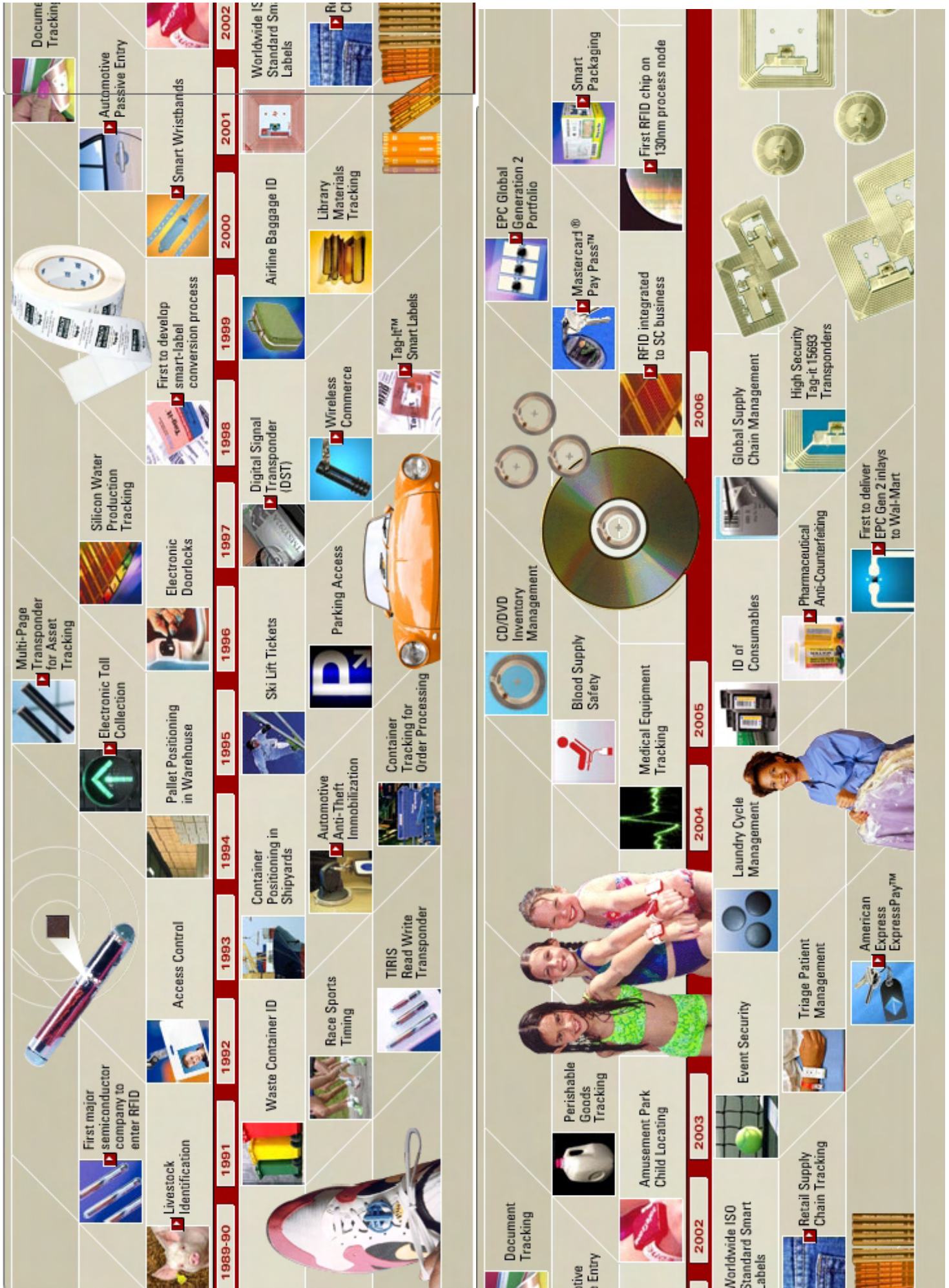
Η βελτίωση και η ανάπτυξη συστημάτων RFID γνώρισε μεγάλη πρόοδο κατά την δεκαετία του '80 με πολλά ερευνητικά προγράμματα σε όλο τον κόσμο για την μαζική εφαρμογή τρόπων αξιοποίησης της νέας τεχνολογίας. Βασικό ρόλο έπαιξε η ραγδαία πλέον ανάπτυξη των Η/Υ και η μαζική χρήση τους, η μείωση του μεγέθους και του κόστους των tag, της ενέργειας που χρειάζεται για την λειτουργία των αναγνώστων και αυξήθηκε η απόσταση από την οποία γίνονται αντιληπτά. Έτσι κατέστη δυνατό η ανάπτυξη πλήρως αυτοματοποιημένων συστημάτων αναγνώρισης πολύ πιο εξελιγμένων και με πολλές εφαρμογές κατά την επόμενη δεκαετία.

Η αρχή έγινε με το αυτόματο σύστημα RFID για τη συλλογή διοδίων που εγκαταστάθηκε το 1987 στη Νορβηγία και το αντίστοιχο που εγκαταστάθηκε στις ΗΠΑ το 1989. Η τεχνολογία RFID είχε βρει μία θέση στην αγορά με τα ηλεκτρονικά συστήματα διοδίων και νέες εταιρίες επένδυναν καθημερινά στην έρευνα για την νέα τεχνολογία. Όλο και περισσότερα αυτοκίνητα προμηθεύονταν κάρτες RFID tag εξασφαλίζοντας τη γρήγορη και απλή διέλευση από τους σταθμούς διοδίων. Πλέον όλες οι πολιτείες των ΗΠΑ, αλλά και πολλές άλλες χώρες, συμπεριλαμβανομένου και της Ελλάδος, έχουν επενδύσει σε σταθμούς διοδίων με RFID. Το σύστημα εφαρμόστηκε επίσης σε ιδιωτικά πάρκινγκ και εγκαταστάσεις όπου ήταν αναγκαίος ο έλεγχος της πρόσβασης των οχημάτων.

Τη δεκαετία του '90 πολλές εταιρίες προσέφεραν τις δικές τους λύσεις με βάση τα όλο και εξελισσόμενα συστήματα RFID. Σίγα αλλά σταθερά τα RFID άρχισαν να μπαίνουν για τα καλά στη ζωή μας και να κατακτούν τη θέση στην αγορά. Καινούριες εφαρμογές, πολλές φορές στη θέση του γνωστού και διαδεδομένου Barcode, από το γνωστό σύστημα προστασίας κατά της κλοπής του αυτοκινήτου, καλούμενο immobilizer, από την εγκατάσταση tag σε βιβλία βιβλιοθηκών μέχρι τον έλεγχο της τροφοδοσίας φαρμάκων και μονάδων αίματος. Ιδιαίτερα ευφάνταστες λύσεις, μερικές από τις οποίες παρουσιάζονται στην επόμενη σελίδα.

1.3.5 Το πρότυπο EPC

Στις αρχές του '90 οι μηχανικοί της IBM ανέπτυξαν ένα σύστημα RFID πολύ υψηλών συχνοτήτων (ultra-high frequency UHF), το οποίο προσέφερε λειτουργία από μεγαλύτερη απόσταση και ταχύτερη μεταφορά δεδομένων. Το 1999 ο Uniform Code Council, ο αντίστοιχος ευρωπαϊκός οργανισμός International EAN και οι εταιρίες PROCTER & GAMBLE και Gillette προσέφεραν χρηματοδότηση για την ίδρυση του Auto-ID Center στο Πολυτεχνείο της Μασαχουσέτης. Δύο καθηγητές εκεί ερευνούσαν την δυνατότητα της τοποθέτησης χαμηλού κόστους RFID tags σε όλα τα προϊόντα του κόσμου για την παρακολούθησή τους σε όλη την αλυσίδα διάθεσής τους. Η ιδέα τους ήταν να τοποθετηθεί μόνο ένας αριθμός σε μια απλή και φθηνή tag με στόχο το μικρό κόστος. Τα δεδομένα τα οποία συνδέονται με τον αριθμό του προϊόντος θα υπάρχουν σε μία βάση δεδομένων η οποία θα είναι προσπελάσιμη μέσω του διαδικτύου. Κάθε προϊόν θα είχε έναν μοναδικό αριθμό στη βάση δεδομένων ενώ στο Barcode κάθε είδος προϊόντος είχε μοναδικό αριθμό. Πλέον οι επιχειρήσεις μπορούσαν να παρακολουθήσουν τα εμπορεύματά τους και αυτά που είχαν αγοράσει μέσω του διαδικτύου. Μεταξύ 1999 και 2003 το Auto-ID Center κέρδισε την υποστήριξη πάνω από 100 μεγάλων εταιριών, του Υπουργείου Άμυνας των ΗΠΑ και πολλών σημαντικών πωλητών RFID. Ανέπτυξε δύο πρωτόκολλα διεπαφής μέσω αέρα (air interface protocols), Class 1 και Class 0, το αριθμητικό σχήμα του Ηλεκτρονικού Κώδικα Προϊόντος, Electronic Product Code (EPC), και την αρχιτεκτονική ενός δικτύου για την αναζήτηση δεδομένων που συνδέονται με κάποιο RFID tag στο διαδίκτυο. Η τεχνολογία αυτή αδειοδοτήθηκε στο Uniform Code Council το 2003 και το τελευταίο σε κοινοπραξία με τον International EAN δημιούργησαν το EPCglobal, ένα καινούριο οργανισμό υπεύθυνο για την προώθηση και την εμπορική ανάπτυξη της τεχνολογίας EPC. Το Auto-ID Center έκλεισε και έδωσε τη θέση του στα Auto-ID Labs, εργαστήρια στο Πολυτεχνείο που συνεχίζουν την έρευνα για τεχνολογίες RFID και EPC. Το 2005 ο οργανισμός Uniform Code Council και ο International EAN σχημάτισαν μια κοινοπραξία από την οποία προέκυψε ο Global Standards 1, GS1, ένας παγκόσμιος πλέον οργανισμός υπεύθυνος για την ανάπτυξη προτύπων όχι μόνο για το Barcode, αλλά και μια σειρά άλλων λύσεων για την διαχείριση της αλυσίδας παραγωγής και διάθεσης.



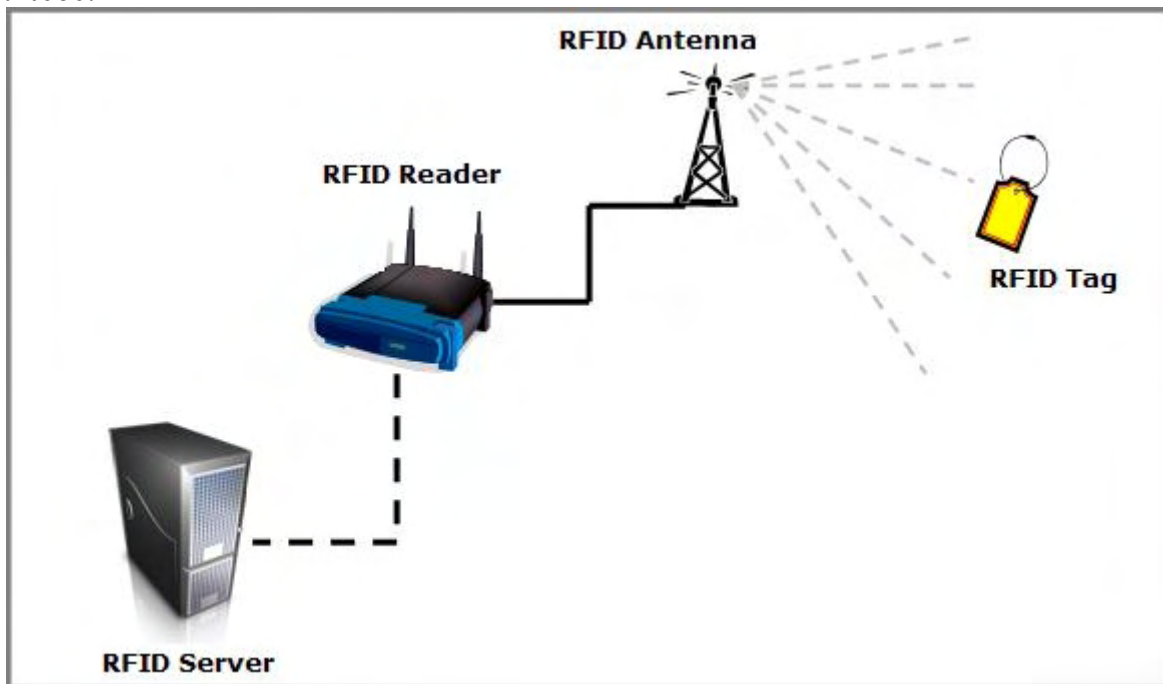
Σχήμα 1.5 Χρονολόγιο εφαρμογών RFID

2. Βασικά μέρη ενός συστήματος RFID

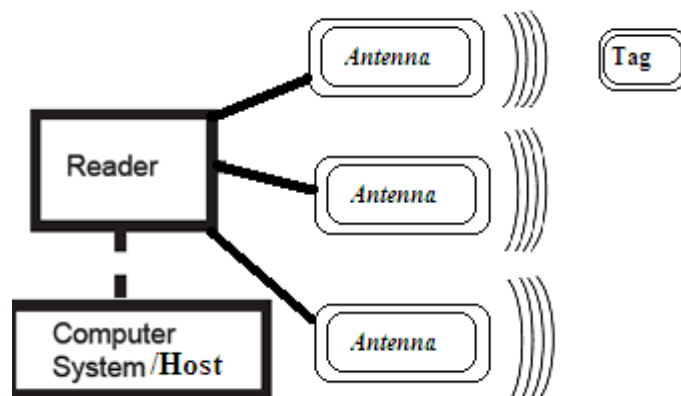
2.1 Εισαγωγή

Ανάλογα με τις ανάγκες που έχουν κλιθεί να καλύψουν, τα συστήματα RFID ποικίλουν από απλές κεραίες που διαβάζουν το αναγνωριστικό ενός tag μέχρι συστήματα πολλαπλών αισθητήρων που συλλέγουν μια σειρά από δεδομένα τα οποία βρίσκονται στη μνήμη ενός tag. Όσο αναφορά τον προάγγελο του RFID, το Barcode, το μόνο που χρειαζόταν ήταν μια συσκευή ανάγνωσης που “διάβαζε” τον γραμμωτό κώδικα, μία ετικέτα κάθε φορά και για την εκτύπωση δεν χρειαζόταν τίποτα περισσότερο από έναν εκτυπωτή. Το μόνο που είχε να κάνει έπειτα ήταν χρησιμοποιώντας μία βάση δεδομένων να βρει τις πληροφορίες που υπήρχαν σ’ αυτή για τον αριθμό που είχε “διαβάσει”. Τα πράγματα όμως δεν είναι τόσο απλά για συστήματα RFID και οι προκλήσεις είναι μεγαλύτερες. Τα tags πρέπει να είναι σε θέση να ανιχνεύονται από απόσταση και συχνά οι εφαρμογές πρέπει να μπορούν να χρησιμοποιούν πολλαπλούς αισθητήρες ταυτόχρονα.

Τα βασικά μέρη του μοντέλου που επικράτησε ήταν μια συσκευή ανάγνωσης (Reader Device) με ένα αριθμό αισθητήρων (κεραίων, Antenna, ή οτιδήποτε άλλο μπορεί να αλληλεπιδράσει με tag), tags τα οποία φέρουν τις προς ανίχνευση πληροφορίες και ο Η/Υ (Host) που διαχειρίζεται τη συσκευή και ενδεχομένως διαθέτει και τη βάση δεδομένων ή αυτή είναι προσπελάσιμη μέσω του διαδικτύου.



Σχήμα 2.1 Βασικά μέρη ενός συστήματος RFID



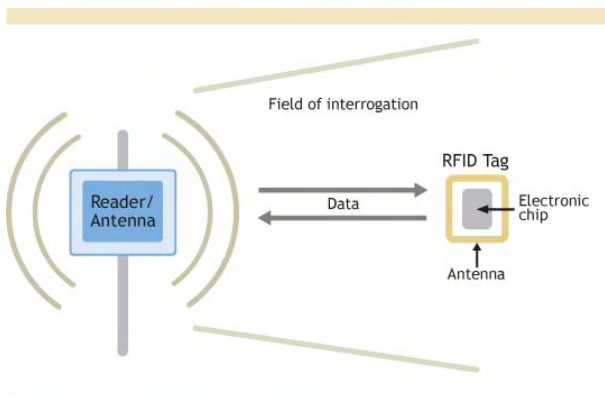
Σχήμα 2.2 Το σχήμα του βασικού μοντέλου ενός συστήματος RFID

2.2 Tags

Tag σημαίνει ετικέτα στα ελληνικά, αλλά αυτός ο όρος δεν αρκεί για να περιγράψει τις ιδιότητες, τις λειτουργίες και τις λύσεις που προσφέρονται μέσω τέτοιων συστημάτων. Θα μπορούσε να χαρακτηριστεί ως κλειδί, ως πιστωτική κάρτα ή οτιδήποτε άλλο ανάλογο προς τη χρήση του. Ο όρος tag περιλαμβάνει τα Barcode tag, δηλαδή ετικέτες γραμμωτού κώδικα, και RFID tag, ένα ηλεκτρικό κύκλωμα με μία κεραία και μνήμη κι ενδεχομένως μία μπαταρία. Η πολυπλοκότητα του RFID tag είναι ανάλογη των πληροφοριών που αποθηκεύουμε(μνήμη), της απόστασης που θέλουμε να γίνεται αντιληπτή και της λειτουργικότητας που θέλουμε να προσφέρουμε, πχ κρυπτογράφηση δεδομένων. Στο πρότυπο EPC κάθε tag έχει αποθηκευμένο τουλάχιστον ένα κωδικό EPC(χωρίς να αποκλείεται η αποθήκευση περισσότερων πληροφοριών), που χρησιμοποιείται για την εύρεση πληροφοριών για το προϊόν μέσω μιας βάσης δεδομένων στο internet.

2.3 Antennas ή Barcode scanners

Για την ανάγνωση των tags χρησιμοποιούμε Barcode scanners για Barcode tag και Antennas, κεραίες, για RFID tags. Πρόκειται για συσκευές που αναλαμβάνουν τη λήψη των πληροφοριών του tag σε φυσικό επίπεδο και υποστηρίζουν ένα πολύ περιορισμένο σύνολο εντολών. Οι συγκεκριμένες συσκευές θα αναφέρονται με τον ReadPoints.



Σχήμα 2.3 Antenna και RFID tag



Εικόνα 2.4 Barcode scanner και tag

2.4 Reader Device

Πρόκειται για τις συσκευές ανάγνωσης/εγγραφής στις οποίες συνδέονται ένας αριθμός από ReadPoints και προσφέρουν μια σταθερή λειτουργία ανάγνωσης/εγγραφής και μια διεπαφή με ένα μεγαλύτερο σύνολο εντολών. Καθιστούν έτσι πιο εύκολο τον προγραμματισμό, μειώνουν την επικοινωνία και προσφέρουν την απεμπλοκή του Η/Υ(Host) από τη διατήρηση της καλής λειτουργίας του συστήματος. Ουσιαστικά προσφέρουν στον προγραμματιστή ένα πιο υψηλό επίπεδο πρωτοκόλλου λειτουργίας και αναλαμβάνουν την εκτέλεση των εντολών και λειτουργιών “τοπικά” στη Reader Device. Οι συσκευές αυτές μπορούν να προσφέρουν αποδοτικές τεχνικές φιλτραρίσματος και ειδοποίησης και λειτουργία μέσω δικτύου.

2.5 Host

Ως Host θα αναφέρεται από δω και πέρα ο Η/Υ που χρησιμοποιεί ο χρήστης για την επικοινωνία με τη Reader Device. Μία Reader Device μπορεί να επικοινωνεί με περισσότερους από ένα Hosts. Στο Host ο χρήστης επικοινωνεί μέσω ενός ενδιάμεσου λογισμικού με τη συσκευή και ενδεχομένως με περισσότερες συσκευές ReaderDevice και ελέγχει την λειτουργία της συλλέγοντας τα δεδομένα που χρειάζεται. Συχνά η Reader Device και ο Host έχουν κατασκευαστεί σε ένα ενιαίο σύστημα.

3. Ανάγκη για πρότυπα

3.1 Χωρίς κανόνες

Μέχρι πριν από μερικά χρόνια πολλοί προμηθευτές και αγοραστές που επένδυναν σε τεχνολογίες και λύσεις RFID σημείωναν ότι δεν υπάρχουν πρότυπα για την νέα τεχνολογία ή ότι οι κατασκευαστές τέτοιων συστημάτων δεν έχουν ακόμη εφαρμόσει κάποιο πρότυπο στα συστήματα που προσφέρουν, είτε για tags ή Antenna ή Reader Device. Ο κάθε κατασκευαστής κατασκεύαζε τα δικά του RFID εξαρτήματα, υλοποιούσε το δικό του πρωτόκολλο επικοινωνίας και χρησιμοποιούσε το δικό του λογισμικό επεξεργασίας.

Όταν κάποια εταιρία ήθελε να χρησιμοποιήσει ένα σύστημα RFID έπρεπε να αξιολογήσει και να επιλέξει τυχόν παρόμοιες λύσεις που είχαν ήδη υλοποιηθεί και αν ζητούσε κάτι νέο ως προς τις υπάρχουσες λύσεις, ο κατασκευαστής ίσως έπρεπε να δημιουργήσει ένα καινούριο σύστημα από την αρχή χωρίς να ακολουθήσει κανένα κανόνα για τον τρόπο που θα το κατασκεύαζε. Έτσι ήταν δύσκολο να επιτευχθεί συμβατότητα μεταξύ των εξαρτημάτων διαφορετικών εταιριών. Επιπλέον η εταιρία ήταν δεσμευμένη πλέον στο συγκεκριμένο κατασκευαστή για προμήθεια tags, ανταλλακτικών, περαιτέρω αναβάθμιση και σέρβις. Αυτό είχε σαν αποτέλεσμα αυξημένο κόστος εγκατάστασης, λειτουργίας και αναβάθμισης. Δεν δινόταν στους προγραμματιστές μια ενιαία πλατφόρμα εργασίας, συγκεκριμένα πρωτόκολλα για την ανάπτυξη των εφαρμογών τους. Από την άλλη, αν κάτι πήγαινε στραβά σε κάποιο μέρος του συστήματος, είτε ήταν κάποιο ελάττωμα στο υλικό είτε στο λογισμικό, συνήθως δεν υπήρχε άλλη επιλογή από την αλλαγή όλου του συστήματος.

Ένα ακόμη πρόβλημα που υπήρχε ήταν ότι οι αγοραστές τέτοιων συστημάτων δεν είχαν κάποιο τρόπο να γνωρίζουν αν τα εξαρτήματα και το λογισμικό μιας εταιρίας πληρούσε τις ανάγκες και της προϋποθέσεις που επιθυμούσαν. Δεν υπήρχε κάποιος τρόπος πιστοποίησης της τεχνολογίας που επρόκειτο να προμηθευτούν.

Όλα τα παραπάνω οδήγησαν στη ανάγκη δημιουργίας προτύπων και οργανισμών υπεύθυνων για την ανάπτυξη και την προώθησή τους.

3.2 Τα πρώτα πρότυπα

Δεν είναι τυχαίο πως οι πρώτες προσπάθειες για την ανάπτυξη προτύπων αφορούσαν την πρώτη μαζική εφαρμογή της τεχνολογίας RFID, την αυτόματη συλλογή διοδίων. Όπως παρουσιάστηκε παραπάνω, στη Νορβηγία αρχικά και λίγο αργότερα σε διάφορες πολιτείες των ΗΠΑ εφαρμόστηκε το νέο σύστημα. Ελλείπει όμως κάποιου προτύπου κάθε εταιρία παρουσίαζε τη δική της λύση και κάθε εταιρία που είχε επιλεγεί για την υλοποίηση του συστήματος RFID των διοδίων κάθε πολιτείας ή ακόμα χειρότερα ενός συγκεκριμένου σχεδίου κάποιας οδού παρείχε τις δικές της κάρτες RFID, κατάλληλες μόνο για τη δική της υλοποίηση. Αυτό έγινε ποιο εμφανές όταν το σύστημα που εγκαταστάθηκε στο σύστημα διοδίων της πολιτείας του Kansas, βασισμένο στο πρότυπο Title 21, παρείχε υποστήριξη και στο αντίστοιχο σύστημα της γειτονικής πολιτείας της Oklahoma. Η πολιτεία της Georgia θα ακολουθούσε, αναβαθμίζοντας τον εξοπλισμό της ώστε να υποστηρίζει το Title 21. Στην πραγματικότητα, αυτές ήταν οι πρώτες εφαρμογές που μπορούσαν να υποστηρίξουν πολλαπλά πρωτόκολλα.

Το 1990 επτά οργανισμοί διοδίων των ΗΠΑ ίδρυσαν την E-Z Pass Interagency Group (IAG), μια κοινοπραξία για την ανάπτυξη ενός ηλεκτρονικού συστήματος συλλογής διοδίων συμβατό σε πολλές διαφορετικές πολιτείες. Το σύστημα αυτό είναι το μοντέλο για τη χρήση μίας μόνο ετικέτας και ενός μόνο λογαριασμό χρέωσης ανά όχημα για την πρόσβαση σε αυτοκινητοδρόμους των διαφόρων αρχών των διοδίων. Ταυτόχρονα ένα πανευρωπαϊκό πρότυπο ήταν αναγκαίο για εφαρμογές διοδίων στην Ευρώπη, και πολλές εταιρείες εργάζονταν στην ανάπτυξη του προτύπου CEN για τα ηλεκτρονικά διόδια.

Τα πρότυπα που δημιουργήθηκαν βρήκαν εφαρμογή σε μια σειρά από άλλες λύσεις όπως σε πάρκινγκ. Έτσι με το ίδιο tag μπορούσε κάποιος να περάσει από διόδια και για να χρησιμοποιήσει πάρκινγκ. Όμως τα πρότυπα αφορούσαν μόνο ένα σύστημα χρέωσης ενός λογαριασμού, με βάση τον αριθμό του tag, που λειτουργεί σε συγκεκριμένο εύρος συχνοτήτων.

3.3 Οργανισμοί πιστοποίησης και προτύπων.

Υπάρχουν διάφοροι οργανισμοί πιστοποίησης και έκδοσης προτύπων που ασχολούνται με την τεχνολογία των RFID. Οι κυριότεροι είναι:

α) ο International Standards Organization ISO, διεθνής οργανισμός προτύπων

β) ο EPCglobal, ο οργανισμός που διαδέχθηκε το Auto ID Center και ασχολείται με την ανάπτυξη προτύπων για τα RFID.

Άλλοι οργανισμοί που ασχολούνται με την δημιουργία προτύπων για τα RFID είναι ο European Telecommunications Standards Institute (ETSI), ευρωπαϊκός οργανισμός τηλεπικοινωνιακών προτύπων και ο Federal Communications Commission (FCC).

Όπως σε κάθε αναδυόμενη αγορά, ο πρόωρος ορισμός προτύπων είναι μια δύσκολη εργασία, που πολλές φορές μπορεί να προκαλέσει σύγχυση. Καθώς οι τεχνολογίες μπαίνουν στην αγορά, διαφορετικές εταιρείες εφαρμόζουν αυτές τις τεχνολογίες με πολλές διαφορετικές μεθόδους και μορφοποιήσεις δεδομένων.

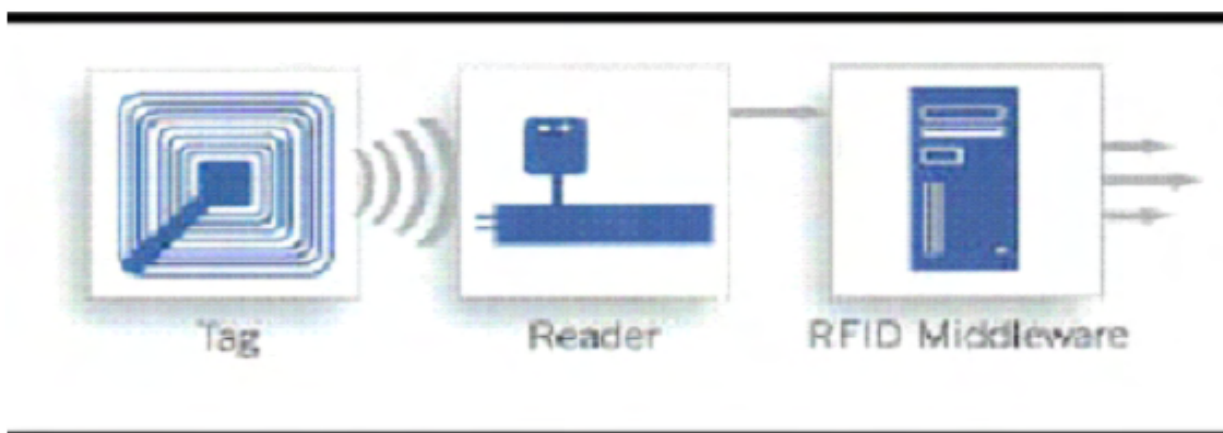
3.3.1 International Standards Organization ISO

Ο ISO είναι μια διεθνή κοινοπραξία 150 χωρών με στόχο την ανάπτυξη προτύπων και την παροχή πιστοποιήσεων σε όλους τους τομείς της ανθρώπινης δραστηριότητας. Έχει εκδώσει μια σειρά προτύπων, άλλα αφορούν συγκεκριμένες εφαρμογές, όπως την σήμανση βοοειδών και ζώων ή τις έξυπνες κάρτες και άλλα αφορούν πιο γενικά πρωτόκολλα επικοινωνίας, πχ που περιγράφουν την επικοινωνία σε συγκεκριμένο εύρος συχνοτήτων.

3.3.2 EPCglobal

Ο EPCglobal έχει ως συγκεκριμένο σκοπό την ανάπτυξη προτύπων για ένα σύστημα που τελικά θα επιτρέψει την αποκλειστική ταυτοποίηση σχεδόν κάθε αγαθού που κατασκευάζεται στον πλανήτη, μαζί με ένα σύστημα πληροφοριών(EPCIS) που θα μπορεί να ανακτήσει όλη την ιστορία του, από τη στιγμή της κατασκευής του. Τέτοιες πληροφορίες θα μπορούσαν να περιλαμβάνουν, για παράδειγμα την ημερομηνία και τον τόπο κατασκευής του προϊόντος, τον αριθμό παρτίδας καθώς και την ιστορία διακίνησης του. Οι εφαρμογές ενός τέτοιου παγκόσμιου συστήματος παραγωγής, και εμπορίας είναι ιλιγγιώδης. Ήδη έχει αρχίσει η δημιουργία ενός τέτοιου συστήματος με την ραγδαία ανάπτυξη του διαδικτύου και της ιδέας ενός διαδικτύου των πραγμάτων(Internet of things).

Ο EPCglobal έχει εκδώσει διάφορα πρωτόκολλα διαφορετικών επιπέδων και διαφόρων λειτουργιών για την υλοποίηση ενός τέτοιου συστήματος, που θα παρουσιαστούν συνοπτικά παρακάτω. Παρότι η προσπάθεια του οργανισμού είναι στην ανάπτυξη ενός χαμηλού κόστους πρότυπου συστήματος για τον παραπάνω στόχο, τα πρότυπα που έχει αναπτύξει μπορούν να καλύψουν μια σειρά από πολλές άλλες εφαρμογές.



Σχήμα 3.1 Βασικό μοντέλο EPCglobal

3.4 Κατηγορίες προτύπων

Οι κατηγορίες των προτύπων για τα RFID περιλαμβάνουν:

- α) τα λεγόμενα **Air Interface** πρότυπα
- β) τη μορφοποίηση των δεδομένων, **data content&encoding**
- γ) την πιστοποίηση συστημάτων, **testing of RFID systems**
- δ) την διαλειτουργικότητα, **interoperability**, μεταξύ εφαρμογών και συστημάτων RFID

Με βάση τους παραπάνω τρόπους διαχωρίζονται τα πρότυπα όσο αναφορά την γενική λειτουργία τους. Θα μπορούσαν να διαχωριστούν με βάση μια συγκεκριμένη εφαρμογή την οποία καλύπτουν, όπως πρωτόκολλα για την σήμανση βοοειδών(ISO 11784 για μορφοποίηση δεδομένων και ISO 11785 για το Air Interface) ή τις έξυπνες κάρτες.

3.4.1 Air Interface

Τα πρότυπα Air Interface καλύπτουν την επικοινωνία σε φυσικό επίπεδο και παρέχουν ένα χαμηλού επιπέδου πρωτόκολλο επικοινωνίας μεταξύ tag και μιας Antenna. Τα βασικά πρότυπα αυτής της κατηγορίας είναι:

- ISO 18000-1 – Γενικές παράμετροι μιας Air Interface διεπαφής για παγκόσμια αποδεκτές συχνότητες.
- ISO 18000-2 – για συχνότητες χαμηλότερα των 135 kHz
- ISO 18000-3 – για λειτουργία σε 13.56 Mhz
- ISO 18000-4 – για λειτουργία σε 2.45 Ghz
- ISO 18000-5: για λειτουργία σε 5.8 GHz
- ISO 18000-6: για λειτουργία μεταξύ 860 MHz και 930 MHz
- ISO 18000-7: για λειτουργία σε 433.92 MHz

τα οποία έχει εκδώσει ο ISO και αντίστοιχα

- Class 0 που περιγράφει ένα μόνο για ανάγνωση tag προγραμματισμένο τη στιγμή της κατασκευής του
- Class 1 που περιγράφει ένα απλό μόνο για ανάγνωση tag που μπορεί να προγραμματιστεί μία φορά
- EPC Generation 2 , η Air Interface επικοινωνία για δεύτερης γενιάς εφαρμογές

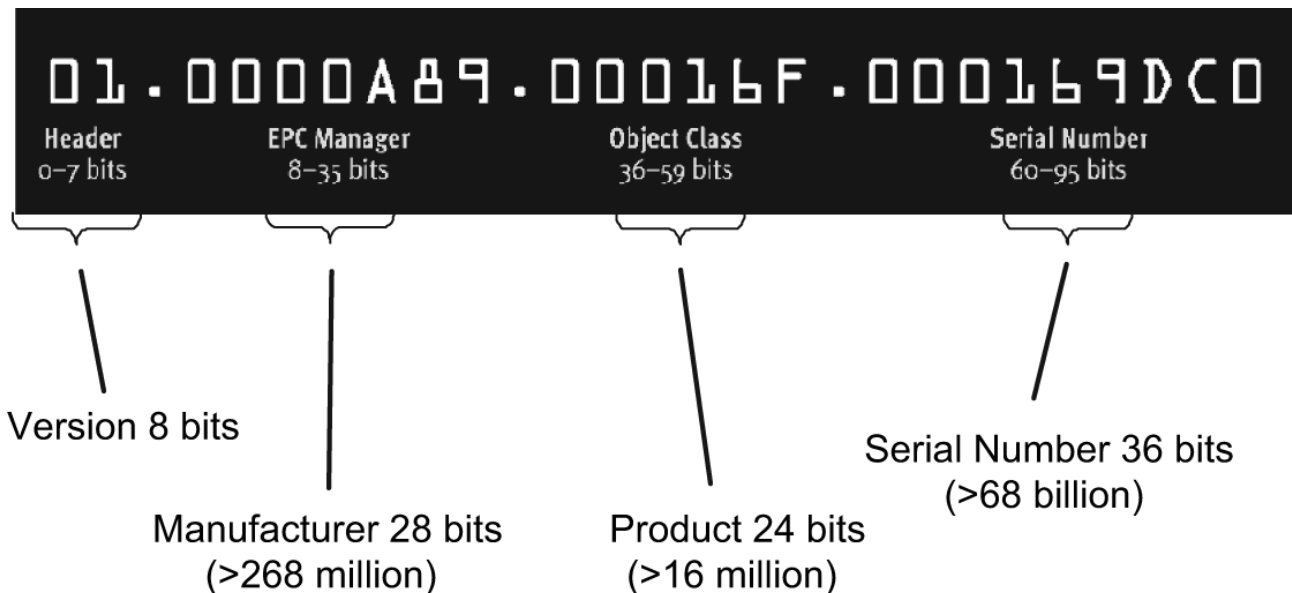
την οποία έχει εκδώσει ο EPCglobal.

Μιας και τα Class0,Class1,Gen2 λειτουργούν πρακτικά στο φάσμα 860 MHz και 930 MHz UHF, θα μπορούσε να υπάρξει συμφωνία για την προσαρμογή τους και του ISO 18000-6 ώστε να επιτευχθεί διαλειτουργικότητα, όμως τελικά κάτι τέτοιο έχει αποτύχει.

3.4.2 Data content&encoding

Τα πρότυπα αυτού του είδους καλύπτουν την αναπαράσταση και τη μορφή των δεδομένων και των πληροφοριών σε tag και Reader Device καθώς και την μορφοποίηση της επικοινωνίας μεταξύ Reader Device και Host. Συνήθως πρόκειται για πρότυπα που προορίζονται για συγκεκριμένες εφαρμογές, όπως σήμανση βοοειδών ή κάρτες διόδων.

Ο EPCglobal έχει εκδώσει το πρότυπο EPC Tag Data Standard για τη μορφοποίηση των δεδομένων στο tag και το EPC Tag Data Translation (TDT) Standard, το αντίστοιχα πρότυπο για την ερμηνεία των δεδομένων του tag στα Reader Device. Το πιο σημαντικό είναι στα δύο αυτά πρότυπα είναι ο αριθμός EPC και η ερμηνεία του. Έχουν μέχρι στιγμής δημιουργηθεί 8 πρότυπα για EPC κώδικες(περισσότερα στο <http://www.epcglobalinc.org/standards/tds/>), με ποιο σημαντικό τον GS1 General Identifier,GID, έναν 96-ψήφιο δυαδικό κώδικα.



Εικόνα 3.2 Μορφή κωδικού EPC/General Identifier EPC

3.4.3 Testing of RFID systems

Πρόκειται για πρότυπα για τη δοκιμή της λειτουργίας και της μέτρησης των επιδόσεων του εξοπλισμού RFID σε συγκεκριμένες επιδόσεις. Η ISO έχει εκδώσει δύο οδηγίες για RFID, ISO 18047 και ISO 18046, ενώ η EPCglobal για κάθε μέρος του συστήματος που αναπτύσσει, έχει εκδώσει οδηγίες τις αντίστοιχες απαιτήσεις.

3.4.4 Interoperability

Πολλές φορές τα πρότυπα που ορίζονται θέλουμε να μπορούν να υλοποιηθούν με μια σειρά διαφορετικών συστημάτων, που χρησιμοποιούνται για να υλοποιούν τον ίδιο σκοπό. Γι αυτό ορίζονται πιο υψηλού επιπέδου πρωτόκολλα και πρότυπα, τα οποία να υλοποιούν μια σειρά από λειτουργίες χωρίς να ορίζεται ο τρόπος που αυτές επιτυγχάνονται.

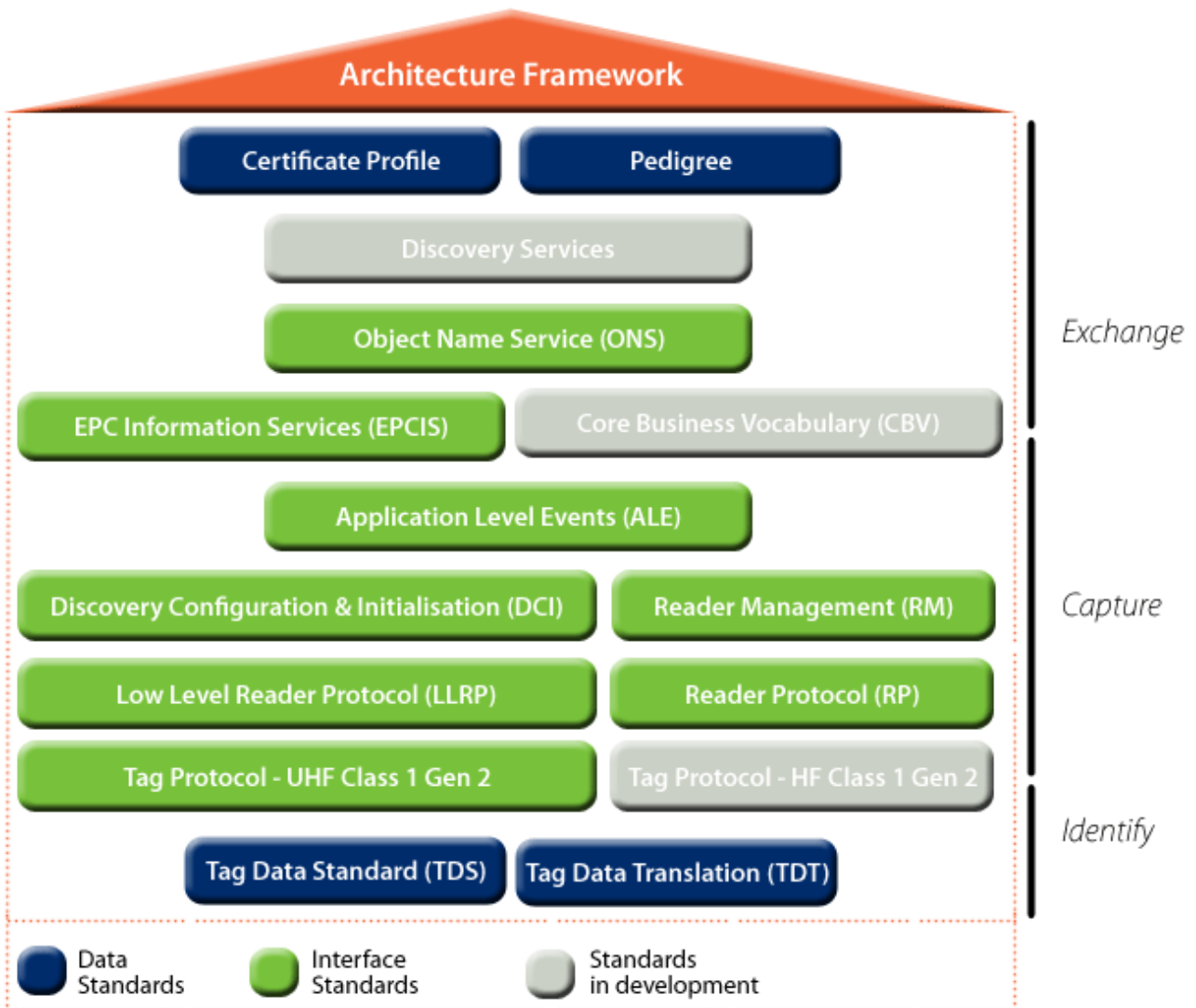
Στο EPCglobal, το πρωτόκολλο της Reader Device προσφέρει τη δυνατότητα της ανάπτυξης ενός συνόλου εντολών και λειτουργιών χωρίς να θέτει περιορισμούς στο σύστημα που θα χρησιμοποιηθεί. Η επιλογή των ReadPoints, του υλικού και του λογισμικού της Reader Device είναι ελεύθερη. Προσφέρεται επίσης μια σειρά τρόπων επικοινωνίας με το Host, serial, tcp ή http.



Σχήμα 3.3 Μοντέλο συστήματος RFID και πρωτόκολλα επικοινωνίας ReaderDevice

3.5 Μοντέλο EPCglobal

Όπως έχει ήδη παρουσιαστεί παραπάνω ο EPCglobal έχει ένα μοναδικό σκοπό, τη σήμανση όλων των αγαθών που παράγονται στον πλανήτη και την συλλογή πληροφοριών για αυτά μέσω του διαδικτύου. Η αρχιτεκτονική του μοντέλου που έχει αναπτύξει φαίνεται στο παρακάτω σχήμα:



Σχήμα 3.4 Αρχιτεκτονική μοντέλου EPCglobal

Έχει 3 βασικά επίπεδα, Identify (Αναγνώριση), Capture (Συλλογή), Exchange (Ανταλλαγή). Το επίπεδο Identify καθορίζει τη μορφή των δεδομένων στο tag και την ερμηνεία τους. Το Capture καθορίζει κανόνες για την συλλογή και επεξεργασία των κωδικών και των δεδομένων που υπάρχουν στα tag. Το Exchange καθορίζει πρότυπα ανταλλαγής πληροφορίας για τα προϊόντα μέσω του διαδικτύου και πιστοποίησης των προϊόντων.

Στη παρούσα εργασία υλοποιήθηκε το επίπεδο του Reader Protocol, που ορίζει ένα πρότυπο για τις συσκευές Reader Device.

4 Reader Protocol

4.1 Εισαγωγή

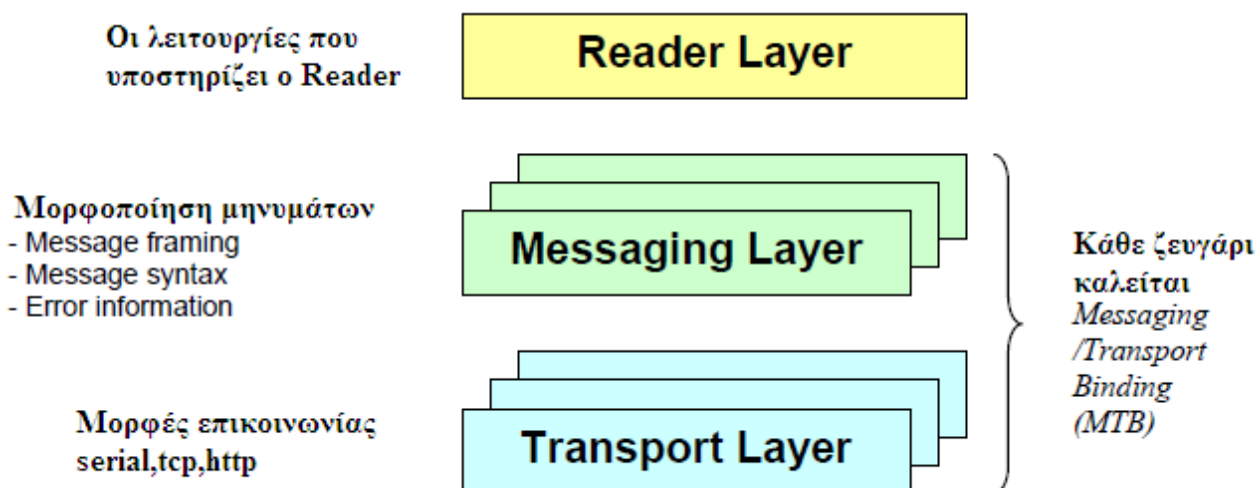
Το Reader Protocol όπως αναφέρθηκε παραπάνω πρόκειται για το πρότυπο που ανέπτυξε ο EPCglobal για το πρωτόκολλο επικοινωνίας μιας συσκευής Reader Device. Το Reader Protocol ορίζει την αλληλεπίδραση μεταξύ μιας Reader Device και του λογισμικού εφαρμογής του Host. Το Reader Protocol προσδιορίζει μόνο την επικοινωνία και τις λειτουργίες μεταξύ Reader Device και Host. Από δω και πέρα η Reader Device θα αναφέρεται εναλλακτικά και ως Reader και αντίστοιχα το λογισμικό επικοινωνίας ως Host.

Η μορφή των δεδομένων στο tag και η αλληλεπίδραση μεταξύ tag και Reader Device είναι εκτός του αντικείμενου αυτού του προτύπου. Ο στόχος είναι η επίτευξη πολυμορφισμού, δηλαδή να απομονωθεί ο Host από τη γνώση των λεπτομερειών του πως η Reader Device και τα tags αλληλεπιδρούν. Οι Reader Devices μπορούν να χρησιμοποιούν μια σειρά από διαφορετικά πρωτόκολλα Air Interface για να αλληλεπιδρούν με τα tag, ακόμη και να διαβάζουν Barcodes.

Προσφέρεται έτσι στους προγραμματιστές μια διεπαφή προγραμματισμού και ελέγχου μέσω των συσκευών Reader Device, χωρίς να χρειάζεται να ασχοληθεί ο προγραμματιστής με χαμηλού επιπέδου επικοινωνία. Επιπλέον εξασφαλίζεται η ελευθερία στους αγοραστές τέτοιων συσκευών να χρησιμοποιήσουν μια σειρά από διαφορετικές συσκευές εφόσον επιτελούν τις ίδιες λειτουργίες.

4.2 Επίπεδα Πρωτοκόλλου

Το πρωτόκολλο του Reader διακρίνεται σε 3 διαφορετικά επίπεδα:



Σχήμα 4.1 Επίπεδα του Reader Protocol

Reader Layer: Το επίπεδο αυτό είναι η καρδιά του πρωτοκόλλου, προσδιορίζοντας τις εντολές που υποστηρίζει ένας Reader και την ερμηνεία τους.

Messaging Layer: Το επίπεδο αυτό προσδιορίζει τον τρόπο που τα μηνύματα και οι εντολές που ορίζονται στο Reader Layer μορφοποιούνται για αποστολή στα κανάλια επικοινωνίας.

Transport Layer: Το επίπεδο αυτό αντιστοιχεί στους τρόπους επικοινωνίας σε φυσικό επίπεδο που προσφέρει το λειτουργικό σύστημα του Reader.

Κάθε συνδυασμός Messaging και Transport Layers καλείται Messaging/Transport Binding (MTB). Διαφορετικά MTBs παρέχουν διαφορετικές μορφές επικοινωνίας.

4.3 Κανάλια Επικοινωνίας

Η επικοινωνία μεταξύ Reader και Host γίνεται με τη μορφή καναλιών επικοινωνίας, message channel. Υπάρχουν 2 τύποι καναλιών:

- *Control Channel – Κανάλι ελέγχου.* Το κανάλι ελέγχου μεταφέρει τις αιτήσεις του Host στο Reader και τις απαντήσεις στις αιτήσεις αυτές. Όλα τα μηνύματα που ανταλλάσσονται σ' αυτό το κανάλι ακολουθούν το μοντέλο αίτηση/απάντηση(request/response). Ουσιαστικά σ' αυτό το κανάλι εκδίδονται οι εντολές του Reader και στέλνονται οι απαντήσεις.
- *Notification Channel – Κανάλι ειδοποίησης.* Το κανάλι ειδοποίησης στέλνει μηνύματα που εκδίδονται ασύγχρονα από τον Reader στο Host. Τα μηνύματα σ' αυτό το κανάλι στέλνονται μόνο από τον Reader. Ουσιαστικά πρόκειται για αναφορές ανάγνωσης tags, οι οποίες στέλνονται στο Host, ανάλογα με τις συνθήκες που ο ίδιος έχει ορίσει για το κάθε πότε θα σταλούν.

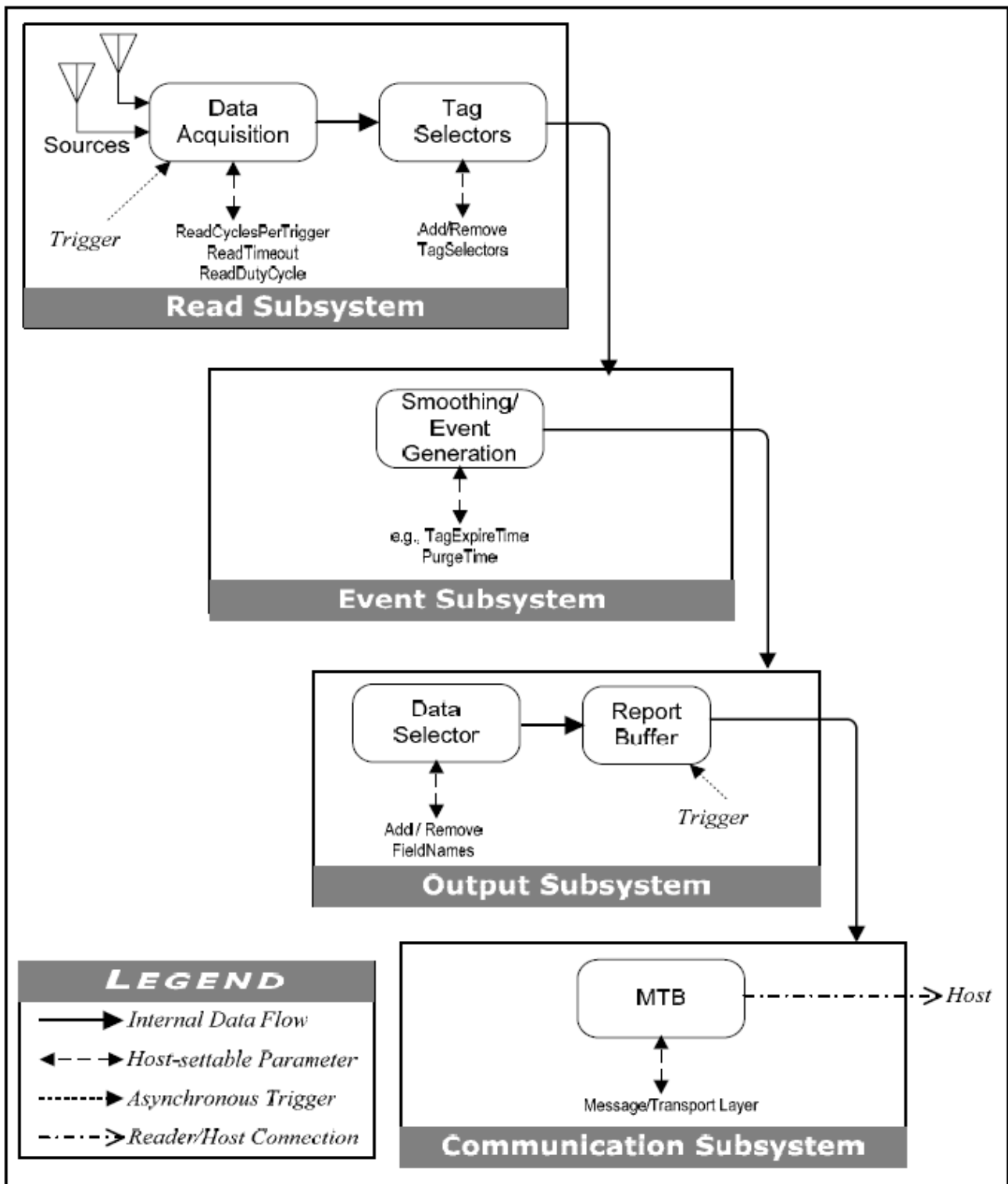
Δύο τύποι καναλιών ορίζονται ώστε οι αναφορές ανάγνωσης να μπορούν να σταλθούν σε διαφορετικό Host από αυτόν που δίνει εντολές. Μπορούν να οριστούν ένα ή περισσότερα κανάλια ελέγχου και ένα ή περισσότερα κανάλια ειδοποίησης. Κάθε κανάλι μπορεί να χρησιμοποιεί το δικό του MTB.

5 Reader Layer – Γενική ιδέα

Το Reader Layer παρέχει ένα κοινό τρόπο για τους Host να αποκτούν πρόσβαση και να ελέγχουν τους συμβατούς Readers που κατασκευάζονται από διαφορετικούς κατασκευαστές. Διαφορετικά μοντέλα Readers μπορεί να διαφέρουν πολύ στην λειτουργικότητα που προσφέρουν, από απλούς Readers που το μόνο που κάνουν είναι να αναφέρουν ποια tags βρίσκονται εντός του πεδίου τους μέχρι πολύπλοκους έξυπνους Readers που παρέχουν φιλτράρισμα tag και γεγονότων ανάγνωσης και άλλες λειτουργίες. Το Reader Protocol ορίζει ένα συγκεκριμένο σύνολο από λειτουργίες που υλοποιούνται συνήθως και παρέχει ένα πρότυπο τρόπο για την πρόσβαση και τον έλεγχο των λειτουργιών αυτών. Δεν χρειάζεται όμως όλοι οι Readers να υλοποιούν όλες τις λειτουργίες που ορίζονται εδώ, παρά μόνο όσες είναι υποχρεωτικές. Όμως όσες λειτουργίες επιπλέον, από αυτές που ορίζει το πρωτόκολλο, μπορούν να υποστηριχθούν, η υλοποίηση τους γίνεται με τον τρόπο που ορίζεται εδώ.

Ένας Reader για παράδειγμα μπορεί να υποστηρίζει τη χρήση απεριόριστων TagSelectors, δηλαδή φίλτρων για tags, ενώ ένας άλλος μόνο ένα ή ένας άλλος κανένα. Το Reader Protocol παρέχει εντολές για την ανακάλυψη των δυνατοτήτων ενός Reader, τις οποίες όλοι οι συμβατοί Readers πρέπει να υλοποιούν.

Στην επόμενη σελίδα παρουσιάζεται ένα διάγραμμα ροής πληροφορίας από τα ReadPoints, επεξεργασίας στο Reader και παράδοσης στο Host. Είναι ένα γενικό μοντέλο που δεν παρουσιάζει όμως όλη την λειτουργικότητα που μπορεί να έχει ο Reader. Η σειρά εφαρμογής των σταδίων επεξεργασίας των δεδομένων δεν είναι υποχρεωτική ούτε προκαθορισμένη και κάθε Reader μπορεί να υλοποιεί με διαφορετική σειρά τις λειτουργίες που παρουσιάζονται.



Σχήμα 5.1 Γενική ροή πληροφορίας και στάδια ανάγνωσης. Πηγή: EPCglobal RP Standard v1.1

Το μοντέλο του διαγράμματος όμως δεν είναι αρκετό για να περιγραφεί το Reader Protocol γι αυτό θα παρουσιαστεί ένα μοντέλο αντικειμένων(Object Model).

5.1 Read Subsystem

Σ' αυτό το στάδιο γίνεται η ανάγνωση των δεδομένων από τα ReadPoints ή Sources όπως παρουσιάζονται εδώ. Υπάρχουν επίσης Triggers, που ελέγχουν πότε θα γίνει η ανάγνωση, και TagSelectors, για το φιλτράρισμα των tags. Το Reader Protocol μπορεί να παρέχει τρόπους χειρισμού της διαδικασίας ανάγνωσης. Κάθε ατομικό διάστημα ανάγνωσης μίας Source καλείται read cycle (κύκλος ανάγνωσης).

5.1.1 Sources

Οι Sources “διαβάζουν” tags και παρουσιάζουν τα δεδομένα τους στο στάδιο Data Acquisition του Read Subsystem. Sources μπορεί να είναι οι κεραιές ή Barcode scanners, ακριβώς όπως τα ReadPoints που έχουν παρουσιαστεί. Στο αντικειμενοστραφές μοντέλο η έννοια των Sources αλλάζει όπως θα δούμε και διαχωρίζεται από αυτήν των ReadPoints.

5.1.2 ReadTrigger

Το ReadTrigger παρέχει τη δυνατότητα της εκκίνησης ανάγνωσης χωρίς αίτηση του Host. Ένα ReadTrigger προκαλεί την ανάγνωση δεδομένων από τις Sources.

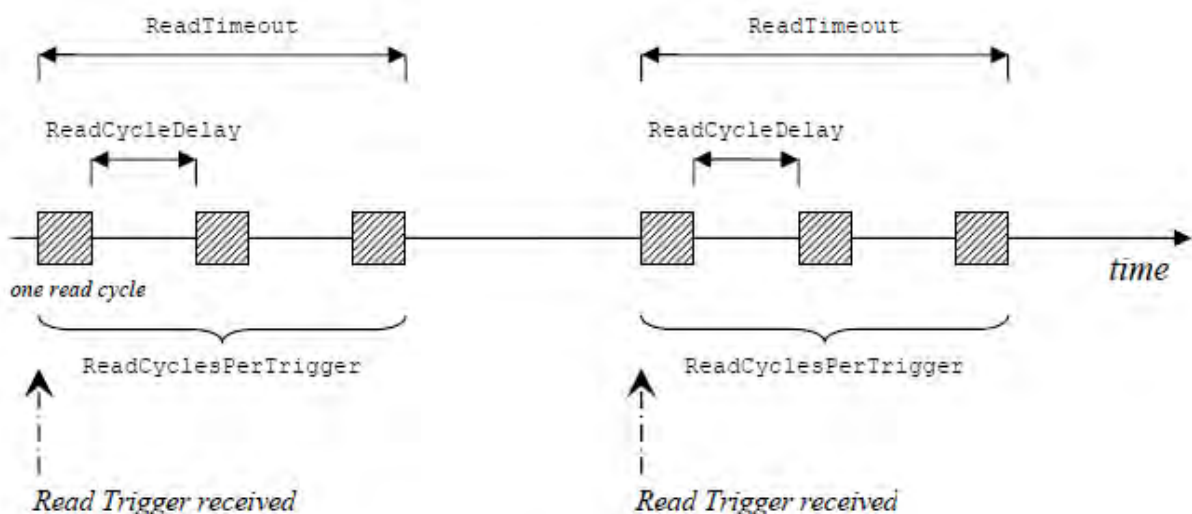
5.1.3 Data Acquisition Stage

Το στάδιο Data Acquisition αντιστοιχεί στη λογική της συλλογής των δεδομένων από τις Sources. Κάθε Source έχει τη δική της ατομική μονάδα συλλογής δεδομένων, το read cycle. Κάθε φορά που “συμβαίνει” ένα ReadTrigger, μια σειρά από read cycles αρχίζει.

Οι παρακάτω παράμετροι ορίζουν την παραπάνω διαδικασία:

- ReadCyclesPerTrigger: ο αριθμός των read cycles που θα εκτελεστούν αν δεν περάσει ο χρόνος ανάγνωσης ReadTimeout.
- ReadTimeout: το χρονικό όριο για να εκτελεστούν τα read cycles. Εάν οριστεί 0 τότε δεν υπάρχει χρονικό όριο.
- ReadMaxDutyCycle: το μέγιστο % όριο ενεργού χρόνου.

$$\text{ReadMaxDutyCycle} = \frac{\text{ReadCycleDuration}}{(\text{ReadCycleDuration} + \text{ReadCycleDelay})}$$



Σχήμα 5.2 Data Acquisition Stage

5.1.4 Read Filter Stage

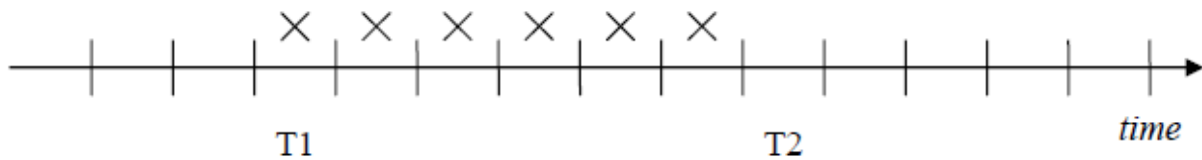
Το στάδιο αυτό αντιστοιχεί στη λογική της αφαίρεσης ορισμένων tags ανάλογα με τα ID τους. Το Reader Protocol παρέχει μία bitwise μέθοδο φιλτραρίσματος, που θα παρουσιαστεί αργότερα.

5.2 Event Subsystem

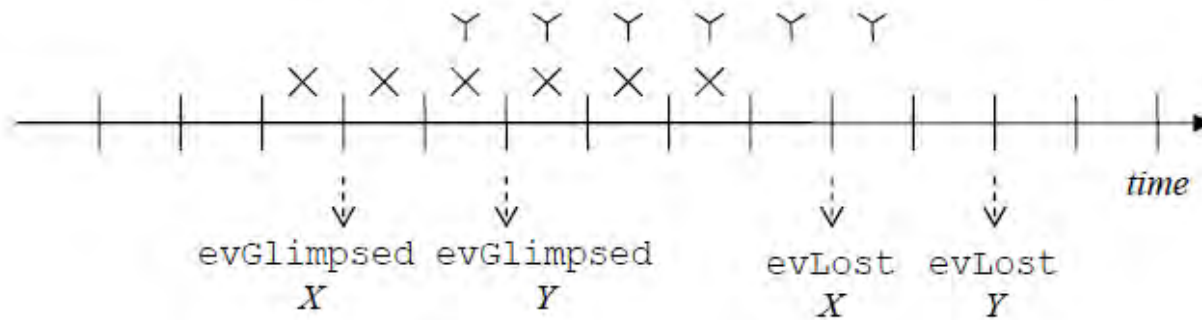
Το στάδιο αυτό εκτελεί event smoothing και generation. Παράγουν δηλαδή μια νέα σειρά γεγονότων με βάση την εμφάνιση κάθε tag στη σειρά των read cycles. Όταν ένα tag εμφανίζεται στο πεδίο μιας συγκεκριμένης Source, το Read Subsystem περιλαμβάνει το tag στην έξοδο του στο τέλος κάθε read cycle. Όταν ένα tag εμφανίζεται για πολλαπλούς read cycles τότε αυτό παράγει ένα σημαντικό αριθμό δεδομένων. Το Event Generation stage μειώνει τον αριθμό αυτό παράγοντας ένα event μόνο όταν κάτι ενδιαφέρον συμβαίνει, πχ όταν ένα tag πρωτοεμφανίζεται και όταν χάνεται.

5.2.1 Smoothing/Event Generation Stage

Ακόμα και όταν υπάρχει φιλτράρισμα των tag, τα δεδομένα που παράγονται είναι πολύ περισσότερα από όσα θα ήθελε μια εφαρμογή να επεξεργαστεί. Ο σκοπός του Event Subsystem είναι να μειώσει το μέγεθος των δεδομένων με το να ειδοποιεί για tags μόνο όταν εμφανίζονται ή βγαίνουν από τη θέα της Source.

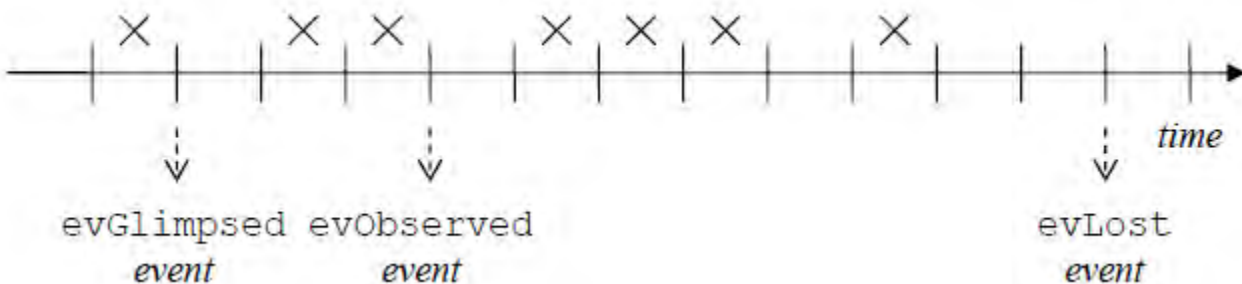


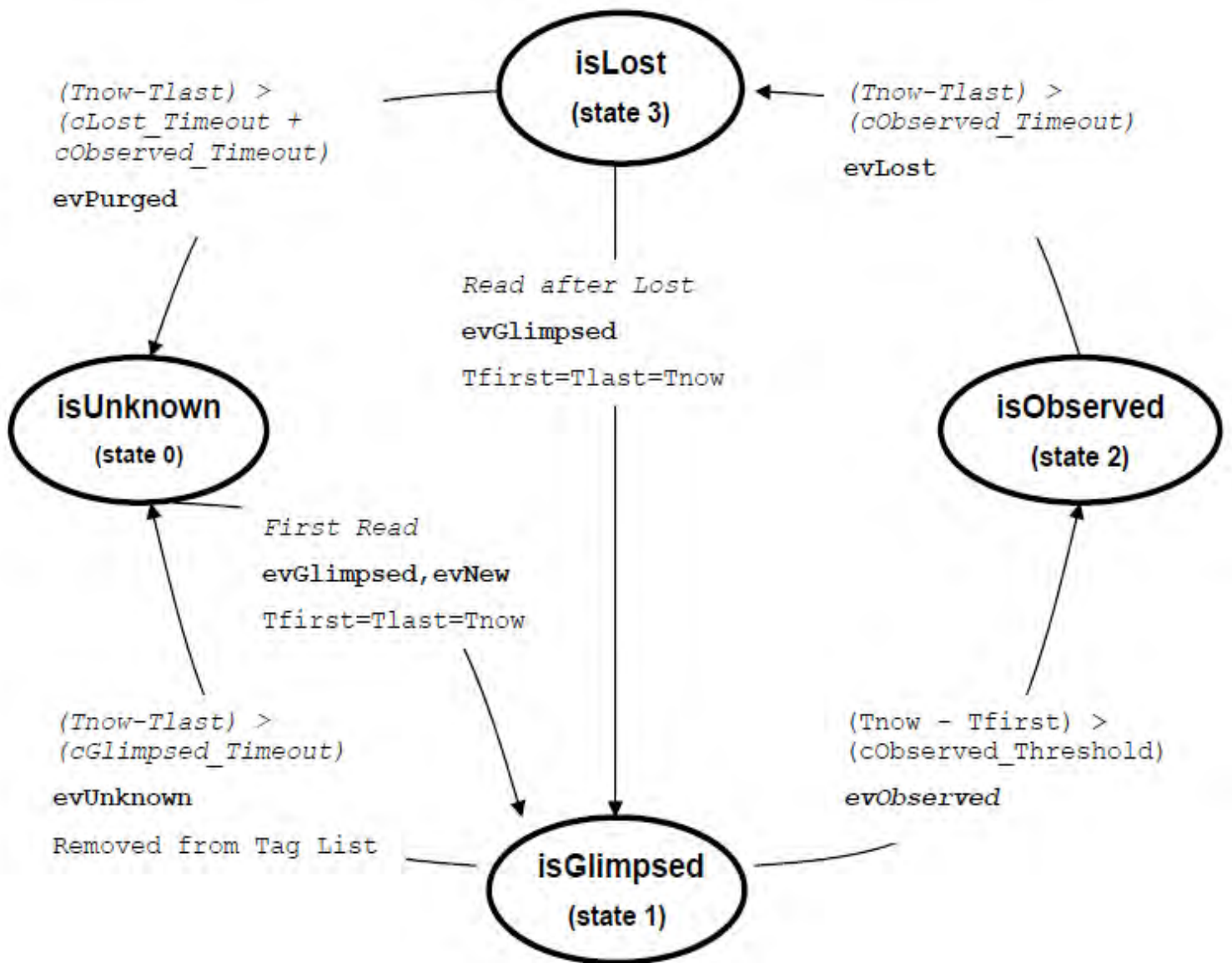
Το Smoothing και Event Generation Stage είναι υπεύθυνο για τη μείωση του αριθμού των δεδομένων παράγοντας “events” όταν η κατάσταση ενός tag αλλάζει.



4. Από 12 γεγονότα που θα είχαμε για κάθε read cycle που εμφανίζονται τα tag X και Y έχουμε

Έτσι ο Reader πρέπει να κρατά πληροφορίες κατάστασης για κάθε tag κάθε Source ξεχωριστά. Επειδή συχνά κάποιο tag μπορεί να μη γίνει αισθητό ακόμα κι αν είναι εντός του πεδίου μίας Source, η παραγωγή των events γίνεται με βάση μεταβλητές που ορίζουν την κατάσταση κάθε tag. Στο παρακάτω σχήμα φαίνεται μια τυπική σειρά read cycles ενός tag.





Σχήμα 5.3 Διάγραμμα καταστάσεων για κάθε tag που γίνεται αισθητό σε κάθε Source και παραγωγή events.

Το παραπάνω διάγραμμα παρουσιάζει τις σχέσεις μετάβασης από τη μία κατάσταση ενός tag σε άλλη με βάση τους χρόνους T_{now} , T_{first} και T_{last} και τις μεταβλητές $c_{Glimpsed_Timeout}$, $c_{Observed_Timeout}$, $c_{Observed_Threshold}$ και $c_{Lost_Timeout}$.

5.3 Output Subsystem

Το στάδιο αυτό καθορίζει ποια δεδομένα θα σταλούν και διαχειρίζεται τη σειρά τους προς παράδοση. Η παράδοση των δεδομένων μπορεί να γίνει σύγχρονα μετά από αίτηση του Host είτε ασύγχρονα μετά από την ενεργοποίηση ενός notification trigger(trigger ειδοποίησης).

5.3.1 Data Selector Stage

Ο Host μπορεί να ρυθμίσει συγκεκριμένα τα περιεχόμενα των αναφορών που θα του παραδοθούν σε αυτό το στάδιο. Ένας Data Selector περιέχει μια σειρά από πεδία που ο Host ενδιαφέρεται. Η επιλογή των δεδομένων γίνεται σε 2 επίπεδα:

- Φιλτράροντας events και
- Προσδιορίζοντας τα πεδία δεδομένων που θα σταλούν

Ένας Data Selector ορίζει μόνο για ποια πεδία ενδιαφέρεται ο Host όχι πως γίνεται αυτή η επιλογή.

5.3.2 Report Buffer Stage

Τα δεδομένα που παράγονται από το Smoothing/Event generation Stage κρατούνται σε ένα Report Buffer.

5.3.3 Notification Trigger

Η αποστολή των δεδομένων στο Host όπως είπαμε μπορεί να γίνει με 2 τρόπους, είτε μετά από αίτηση του Host είτε ασύγχρονα μετά από την ενεργοποίηση κάποιου notification trigger.

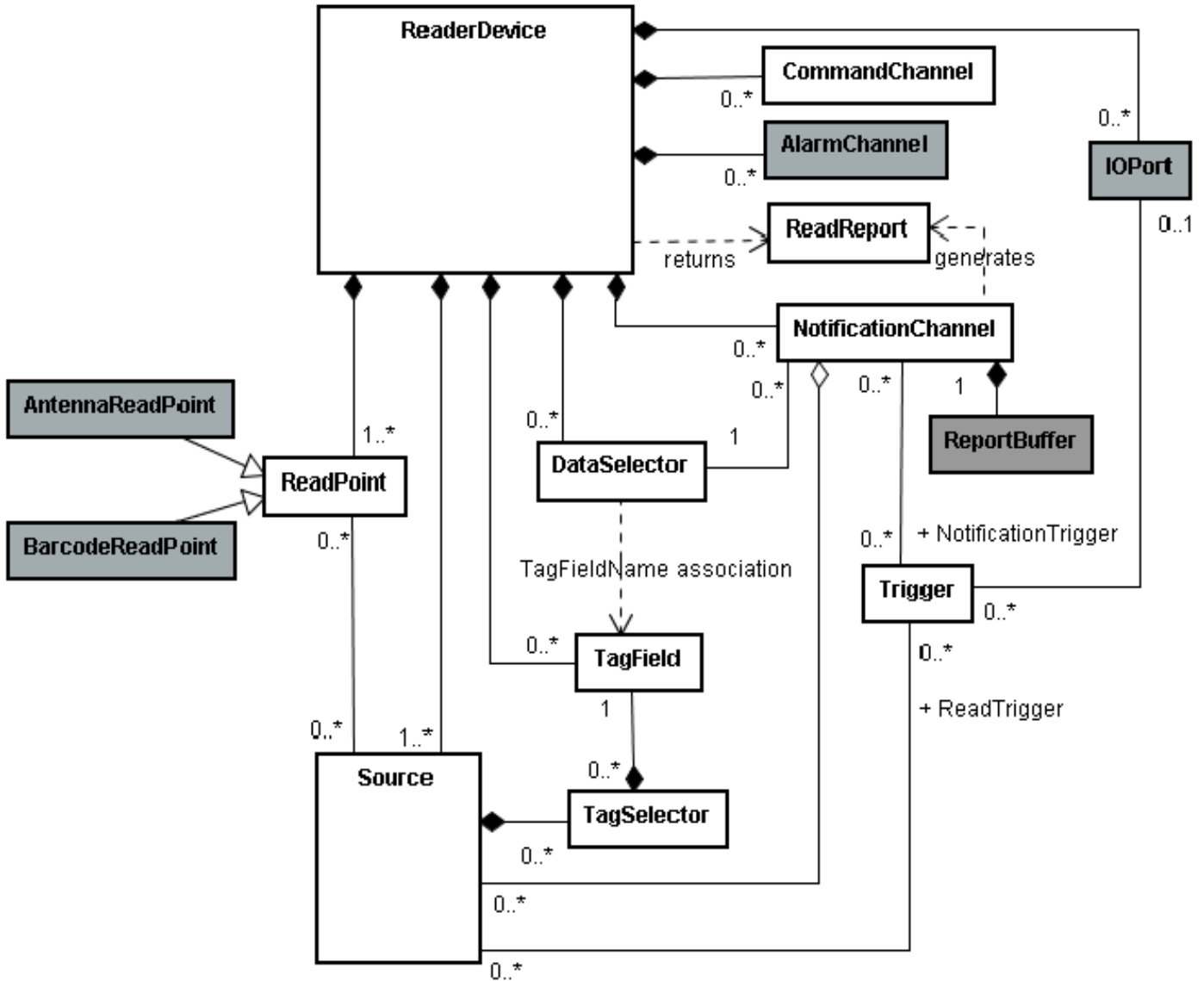
5.4 Communication Subsystem – Message/Transport Binding Stage

Σ' αυτό το στάδιο γίνεται η επικοινωνία μέσω των καναλιών που έχουν οριστεί με βάση ένα συγκεκριμένο MTB για καθένα από αυτά. Η μορφή των μηνυμάτων και των εντολών που ανταλλάσσονται μεταξύ Reader και Host είναι είτε μορφή xml είτε μορφή κειμένου(text). Οι τρόποι σύνδεσης με τη συσκευή μπορεί να είναι είτε serial, tcp ή http.

Για την πρόσβαση σε κάθε αντικείμενο ενός τύπου χρησιμοποιείται ο τύπος του αντικειμένου και το όνομα του και για την κλήση κάποιας συνάρτησης χρησιμοποιείται το όνομα της συνάρτησης.

6 Object Model

Το γενικό διάγραμμα ροής πληροφορίας που παρουσιάστηκε δεν αρκεί για να περιγράψει πλήρως την πολυπλοκότητα ενός Reader. Χρειαζόμαστε ένα μοντέλο αντικειμένων για την πλήρη περιγραφή του πρωτοκόλλου του Reader. Αυτό το γενικό μοντέλο αντικειμένων φαίνεται στο παρακάτω σχήμα.



Σχήμα 6.1 Object Model του Reader Protocol

Το παραπάνω μοντέλο περιέχει τις σχέσεις μεταξύ των αντικειμένων, δηλαδή ποια και πόσα αντικείμενα μπορούν να υπάρξουν και τις αναφορές που πρέπει να χρησιμοποιεί κάθε αντικείμενο για τα άλλα. Στη συνέχεια γίνεται η περιγραφή του σκοπού και των ιδιοτήτων κάθε αντικειμένου.

6.1 ReaderDevice Object

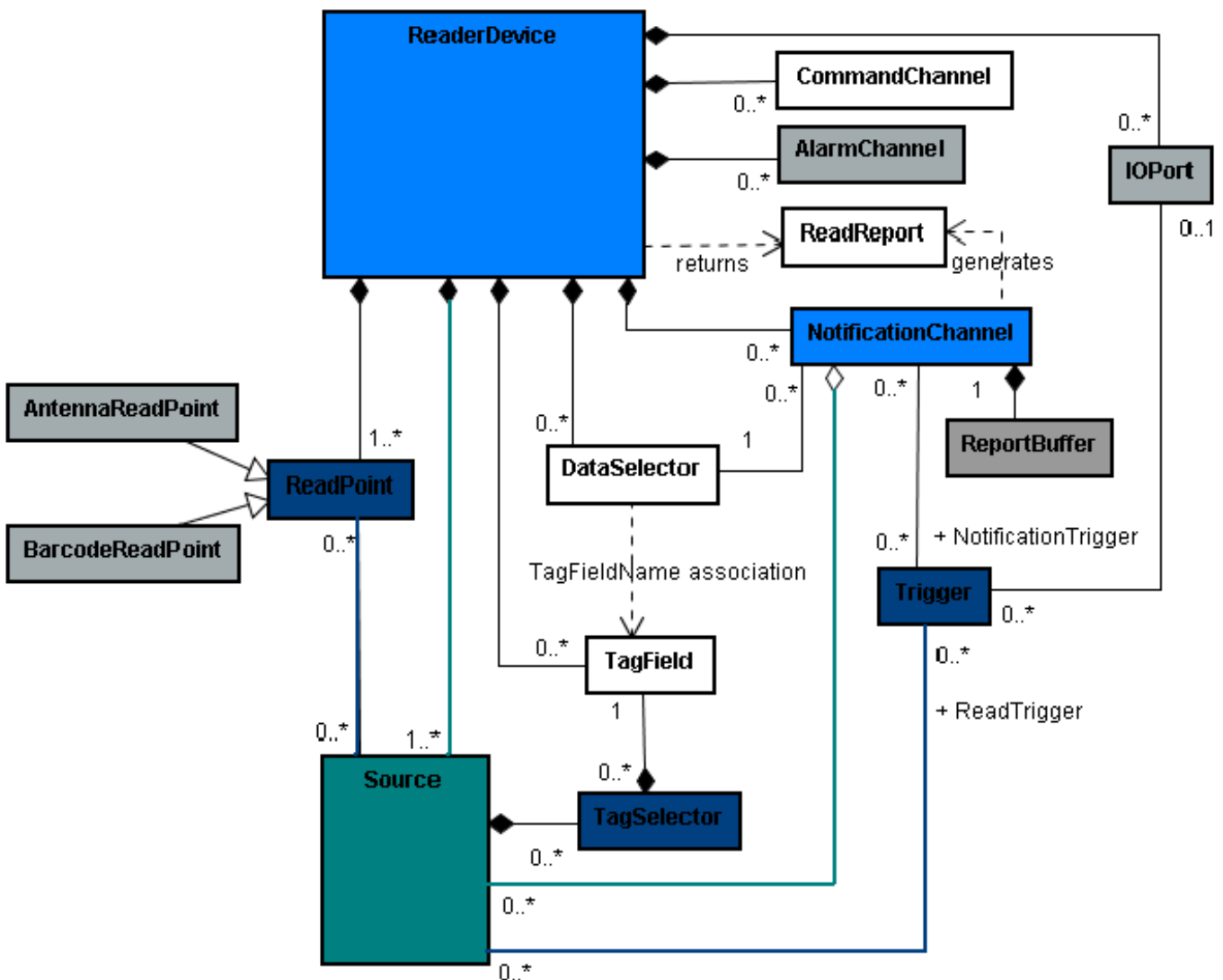
Το αντικείμενο ReaderDevice είναι μοναδικό και αναπαριστά τη συσκευή του Reader. Είναι το βασικό αντικείμενο που περιέχει(κρατάει αναφορές) όλα σχεδόν τα υπόλοιπα αντικείμενα του Reader Object Model συμπεριλαμβανομένου ενός προκαθορισμένου CommandChannel. Επίσης περιέχει διάφορα χαρακτηριστικά που χρησιμοποιούνται για τη διαχείριση του Reader Protocol.

6.2 ReadPoints

Πρόκειται για αντικείμενα που αναπαριστούν φυσικές οντότητες, δηλαδή φυσικούς αισθητήρες RFID ή Barcode scanners. Ο αριθμός και το είδος τους είναι προκαθορισμένο από τον κατασκευαστή.

6.3 Sources

Ένα αντικείμενο τύπου Source αναπαριστά μια λογική οντότητα στην οποία περιέχεται ένα σύνολο από ReadPoints.



Σχήμα 6.2 Σχεσιακό μοντέλο του αντικειμένου Source.

Μπλε σκούρο: ReadPoints, TagSelectors, Triggers περιέχονται σε μια Source
Γαλάζιο: ReaderDevice, NotificationChannels μπορεί να περιέχουν Sources

Αντικείμενα τύπου Source περιέχονται από το ReaderDevice και από αντικείμενα τύπου NotificationChannel. Μία Source περιέχει κανένα, ένα ή περισσότερα ReadPoints, προαιρετικό αριθμό TagSelectors και Triggers (ReadTriggers). Χρησιμοποιούμε το αντικείμενο Source ως μια λογική οντότητα πιο υψηλού επιπέδου από τα ReadPoints και για την ομαδοποίηση τους ταυτόχρονα.

6.4 Triggers

Αντικείμενα Trigger περιέχονται είτε από μία Source, όπου δρουν ως ReadTriggers ξεκινώντας μια διαδικασία ανάγνωσης των δεδομένων των ReadPoints της Source και μεταφορά τους σε ReportBuffers, είτε από ένα NotificationChannel, όπου δρουν ως NotificationTriggers ξεκινώντας μια διαδικασία αποστολής αναφοράς ανάγνωσης των δεδομένων, που υπάρχουν στο ReportBuffer αυτού του NotificationChannel.

Η χρησιμοποίηση των Triggers είναι προαιρετική στο Reader Protocol.

6.5 TagSelectors

Ένα αντικείμενο TagSelector περιέχεται από μία ή περισσότερες Source. Πρόκειται για αντικείμενα φιλτραρίσματος των tags που έχουν εντοπιστεί από τα ReadPoints, με βάση δεδομένα που υπάρχουν στο tag.

Ένας TagSelector προσδιορίζεται από 2 δεκαεξαδικά αλφαριθμητικά, μία filter τιμή και μία mask τιμή. Στη mask τιμή όλα τα bit που η τιμή της θέσης τους μας ενδιαφέρει είναι 1. Στη filter τιμή, ορίζεται η επιθυμητή τιμή αυτών των θέσεων που μας ενδιαφέρουν.

Παράδειγμα:

filter mask M = 1C (00011100 σε δυαδικό, μας ενδιαφέρουν οι θέσεις 4,5,6)

filter value V = 10 (00010000 σε δυαδικό, οι τιμές των θέσεων 4,5,6 πρέπει να ταιριάζουν)

tag ID data A = 55 (01010101)

Ταιριάζουν οι τιμές στις θέσεις 4,6 όχι όμως της 5.

6.5.1 Πολλαπλοί TagSelectors

Πολλαπλοί TagSelectors μπορούν να οριστούν σε μια Source. Καθένας από αυτούς μπορεί να είναι είτε inclusive(ενδιαφερόμαστε μόνο για tag που ταιριάζουν), είτε exclusive(ενδιαφερόμαστε για tag που δεν ταιριάζουν). Αν έχουμε πολλαπλούς TagSelectors τότε ενδιαφερόμαστε για κάποιο tag αν:

- το tag ταιριάζει με τουλάχιστον ένα inclusive TagSelector(εφόσον υπάρχουν) και
- δεν ταιριάζει με κανένα από τα exclusive TagSelectors

6.6 Channels

Υπάρχουν 2 τύποι Channel, τα CommandChannels υπεύθυνα για την αποστολή των εντολών του Host στο Reader και των απαντήσεων του δευτέρου στον πρώτο, και τα NotificationChannels υπεύθυνα για την αποστολή των αναφορών ανάγνωσης της συσκευής. Κάθε Channel χρησιμοποιεί ένα συγκεκριμένο MTB για την επικοινωνία. Τουλάχιστον ένα CommandChannel πρέπει να είναι προκαθορισμένο από τον κατασκευαστή και να “ακούει” σε μια συγκεκριμένη διεύθυνση.

6.7 DataSelectors

Πρόκειται για αντικείμενα που καθορίζουν τα δεδομένα που θα περιέχονται σε μια αναφορά ανάγνωσης και επίσης παρέχουν μέθοδο φιλτραρίσματος των event που μας ενδιαφέρουν. Περιέχονται είτε από NotificationChannels, ορίζοντας τα πεδία και τις πληροφορίες των tags που θα σταλθούν στα κανάλια αυτά είτε περνάνε ως παράμετροι σε εντολές ανάγνωσης για τον ίδιο λόγο.

6.8 TagFields

Ένα TagField αναφέρεται σε συγκεκριμένη περιοχή στη μνήμη ενός tag είτε χρησιμοποιώντας αριθμούς θέσης, (από ποιο σημείο byte μέχρι που μας ενδιαφέρει) είτε αφαιρετική προσέγγιση χρησιμοποιώντας ένα όνομα για την περιοχή αυτή(αυτό γίνεται στα tags Gen2 όπου το tagID υπάρχει στην περιοχή μνήμης “TAGID”).

6.9 ReadReport

Μία ReadReport αναπαριστά μια αναφορά ανάγνωσης των tags που έχουν γίνει αισθητά. Μία ReadReport μπορεί να σταλεί από ένα NotificationChannel ή ως απάντηση σε μια εντολή ανάγνωσης.

6.10 ReportBuffer

Κάθε NotificationChannel διαθέτει ένα ReportBuffer στον οποίο κρατούνται τα tags που έχουν γίνει αισθητά από τις Sources που ανήκουν σ' αυτό. Με βάση τα περιεχόμενα του ReportBuffer παράγεται μία ReadReport, για την αποστολή της στο Host.

6.11 IOPort

Το IOPort αντιπροσωπεύει τη φυσική οντότητα μιας θύρας εισόδου/εξόδου, στην οποία μπορούν να σταλούν ή να ληφθούν δεδομένα. Όμως δεν είναι αναγκαίο να αντιστοιχούν σε μια φυσική θύρα I/O της συσκευής, αλλά μπορούν να αντιστοιχούν σε αντικείμενα που προσομοιάζουν την λειτουργία μιας θύρας I/O.

6.12 Σύνοψη

Παραπάνω έγινε μια σύντομη περιγραφή των ιδιοτήτων του αντικειμενοστραφούς μοντέλου του Reader Protocol. Ένας συμβατός Reader καλείται να παρέχει υποστήριξη για το παραπάνω μοντέλο. Λόγω του γεγονότος ότι είναι αντικειμενοστραφές το μοντέλο του πρωτοκόλλου η υλοποίησή του είναι προτιμότερο να γίνει σε μια αντικειμενοστραφής γλώσσα προγραμματισμού. Γι αυτό το λόγο έχει επιλεγεί η Java. Επιπλέον δεν χρειάζεται προγραμματισμός σε φυσικό επίπεδο, κάτι που διευκολύνει την υλοποίηση.

Σε κάθε αντικείμενο ενός τύπου έχει δοθεί ένα μοναδικό όνομα, το οποίο χρησιμοποιείται για τον προσδιορισμό του. Δύο αντικείμενα του ίδιου τύπου δεν μπορούν να έχουν το ίδιο όνομα.

7. Υλοποίηση σε Java

Για την υλοποίηση του προτύπου επιλέχθηκε η γλώσσα προγραμματισμού της Java κυρίως λόγω του αντικειμενοστραφούς χαρακτήρα της. Είναι πιο απλό να ορίσουμε τα αντικείμενα ως κλάσεις της Java και τις συναρτήσεις κάθε αντικειμένου ως μεθόδους του.

7.1 Περιβάλλον και πρόγραμμα ανάπτυξης

Χρησιμοποιήθηκε το πρόγραμμα JCreator της Xinox Software, έκδοση 4.5 (www.jcreator.com) σε περιβάλλον Windows.

Επίσης χρησιμοποιήθηκε το πρόγραμμα Eclipse SDK του Eclipse Foundation, έκδοση 3.5.1 (<http://www.eclipse.org>) σε περιβάλλον Linux.

7.2 Βιβλιοθήκες που χρησιμοποιήθηκαν

- Sun Java Development Kit 1.6, η πρότυπη βιβλιοθήκη της Java
- Dom4j (org.dom4j) για δημιουργία και επεξεργασία δεδομένων σε xml μορφή
- JSR 80: Java USB API για τον έλεγχο συσκευών με διασύνδεση usb.

7.3 Υλοποίηση προγράμματος Client στο Host

Επιπλέον με την υλοποίηση του προγράμματος, υλοποιήθηκε και το αντίστοιχο πρόγραμμα Client στον Host. Ο σκοπός του λογισμικού Client είναι να παρέχει την ψευδαίσθηση της χρήσης αντικειμένων και εκτέλεσης των εντολών του Reader Protocol τοπικά στον Host και να αποκρύψει το περιβάλλον επικοινωνίας. Ακόμα παρέχεται η διαχείριση πολλαπλών συσκευών ReaderDevice. Έτσι οι προγραμματιστές μπορούν να αναπτύξουν προγράμματα με τη χρήση της Client διεπαφής λογισμικού.

7.4 Άλλες υλοποιήσεις ανοικτού κώδικα

Το 2001 το Auto-ID Lab και ο όμιλος Κατανεμημένων συστημάτων του ETH Zurich (ομοσπονδιακό Ινστιτούτο Τεχνολογιών της Ελβετίας) δημιούργησαν ένα ερευνητικό πρόγραμμα με στόχο την υλοποίηση τεχνολογιών RFID. Στις αρχικές συνεισφορές του προγράμματος περιλαμβάνονται η γλώσσα PML Core, μια γλώσσα σήμανσης, η οποία δημοσιεύτηκε το 2003 από τον EPCglobal, και το RFIDStack, ένα middleware, που δημιουργεί ένα Interface αφαιρώντας τις ιδιαιτερότητες διαφορετικών συσκευών ανάγνωσης (Reader Devices).

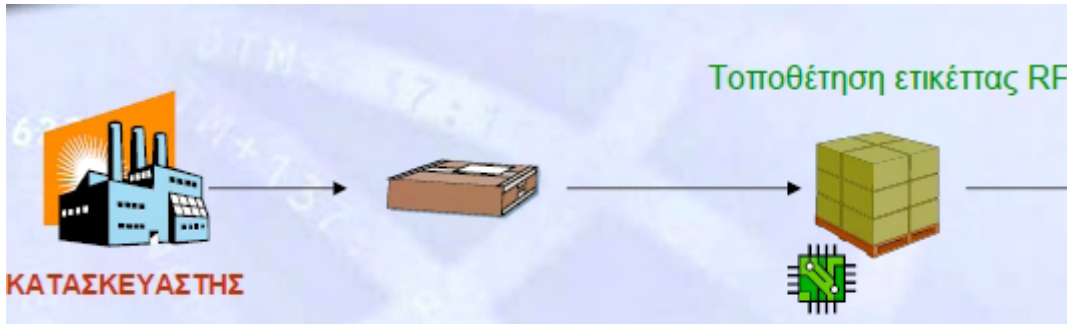
Το 2005 αποφασίστηκε η ανάπτυξη πακέτων λογισμικού για RFID και συγκεκριμένα η υλοποίηση των πρωτοκόλλων του EPCglobal, όπως το Reader Protocol. Το πρόγραμμα αρχικά ονομάστηκε Accada, αλλά αργότερα η ονομασία άλλαξε σε Fosstrak.

Το Fosstrak Reader Project περιλαμβάνει την υλοποίηση λογισμικού του πρωτοκόλλου Reader Protocol Standard v1.1 σε Java, το οποίο διαθέτει 3 μέρη. Το Reader RP/RM Core αφορά την υλοποίηση λογισμικού στη ReaderDevice, το Reader RP Proxy αφορά την υλοποίηση λογισμικού στην πλευρά του Client και το Reader RP Client είναι ένα γραφικό περιβάλλον εκτέλεσης εντολών και επικοινωνίας με ReaderDevices.

8. Δίκτυο EPCglobal

Το δίκτυο EPCglobal όπως αναφέρθηκε παραπάνω έχει ως στόχο την σήμανση με RFID tags όλων των προϊόντων του κόσμου. Ήδη πολλές εταιρίες έχουν εισάγει την νέα τεχνολογία στις αλυσίδες παραγωγής και διάθεσής τους. Ας κάνουμε όμως μια μικρή ανάλυση του πως ακριβώς δουλεύει και ποια στάδια περιλαμβάνει για ένα προϊόν.

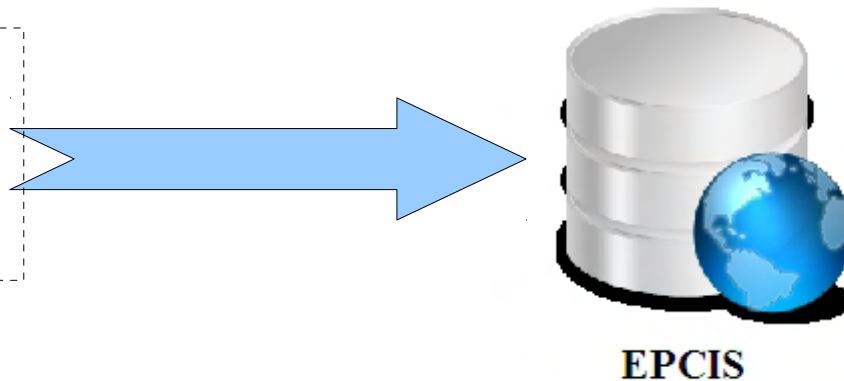
8.1 Παραγωγή ενός προϊόντος



Εικόνα 8.1 Παραγωγή και σήμανση με tag

Κατά την παραγωγή και συσκευασία ενός προϊόντος τοποθετείται ένα RFID tag με μοναδικό κωδικό EPC. Έπειτα γίνεται εισαγωγή πληροφοριών γύρω από το προϊόν στη βάση δεδομένων EPCIS (EPC Information System).

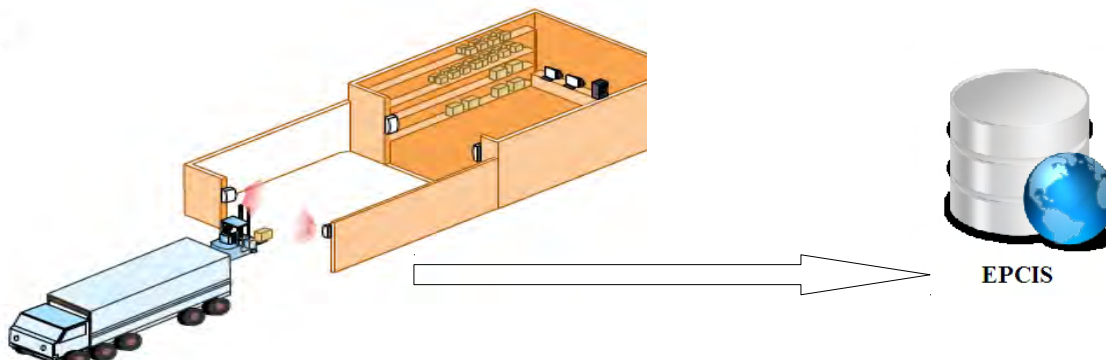
Πληροφορίες για το προϊόν όπως πχ ημερομηνία παραγωγής εταιρία παραγωγής κ.ά. Σύνδεση των πληροφοριών με τον κωδικό EPC του tag.



Εικόνα 8.2 EPCIS βάση δεδομένων

8.2 Αλυσίδα διάθεσης προϊόντος

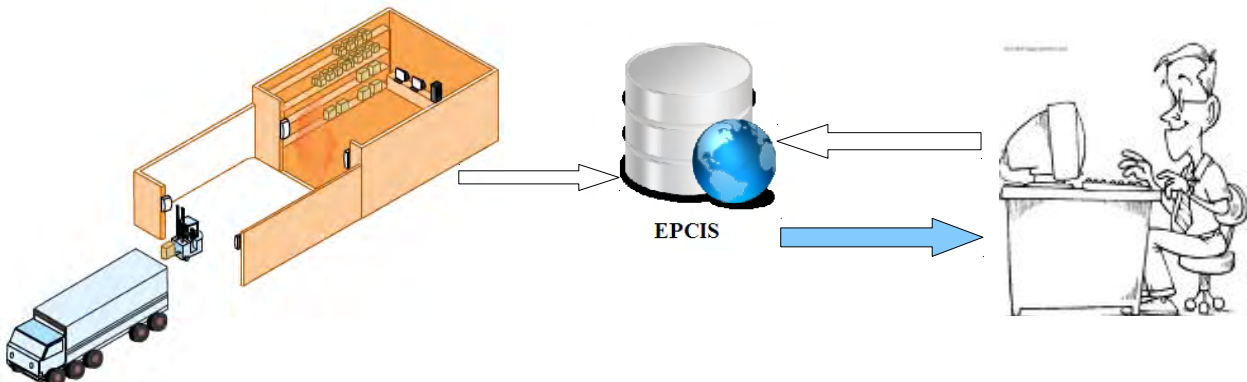
Από όταν φεύγει το προϊόν από τις αποθήκες του εργοστασίου ξεκινά η λειτουργία του δικτύου EPCglobal. Αισθητήρες (ReadPoints) στις εισόδους των αποθηκών λαμβάνουν τους κωδικούς των προϊόντων που μπαίνουν.



Εικόνα 8.3 Εντοπισμός προϊόντος και προσπέλαση πληροφοριών στη EPCIS

8.3 Πώληση προϊόντος

Όταν το προϊόν πωληθεί, μέχρι να φτάσει στα χέρια του αγοραστή, ο τελευταίος μπορεί να γνωρίζει που βρίσκεται ανά πάσα στιγμή εφόσον όλα τα σημεία διανομής διαθέτουν συστήματα αισθητήρων. Η θέση του προϊόντος ανανεώνεται στη βάση δεδομένων EPCIS και ο αγοραστής “βλέπει” τη θέση του και το χρόνο αποστολής στη EPCIS.



Εικόνα 8.4 Ο αγοραστής λαμβάνει πληροφορία για το προϊόν του.

8.4 Πλεονεκτήματα σε αντιδιαστολή με το Barcode

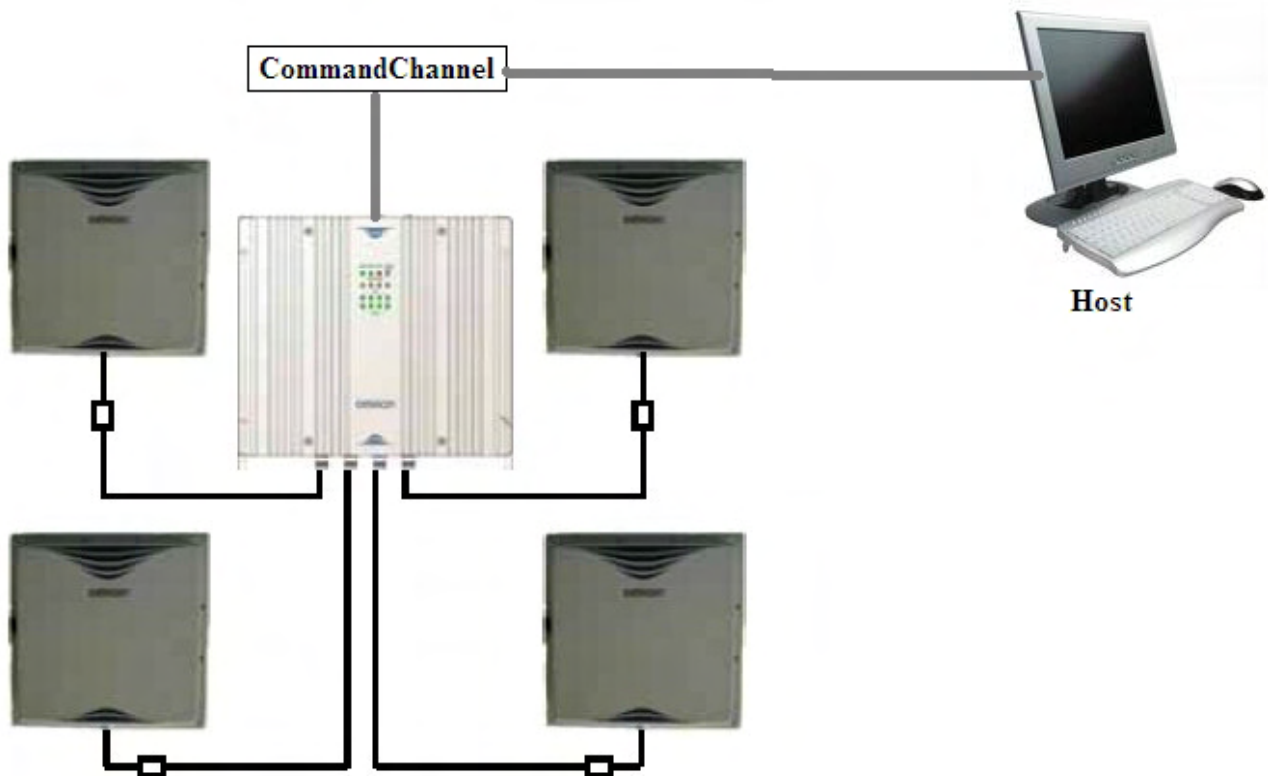
Με τα RFID tags, έχουμε ένα πιο αξιόπιστο τρόπο ανάγνωσης καθώς είναι ανθεκτικότερα στο χρόνο, η ανάγνωση τους γίνεται από μεγαλύτερη απόσταση και χωρίς να βρίσκονται σε οπτική επαφή και με λιγότερα λάθη. Ο ανθρώπινος παράγοντας μειώνεται ακόμα περισσότερο και επιτυγχάνεται μεγαλύτερη αυτοματοποίηση της αλυσίδας διάθεσης. Κάθε προϊόν έχει μοναδικό κωδικό με RFID tag ενώ στο Barcode κάθε είδος προϊόντος έχει μοναδικό αριθμό.

9. Εκτέλεση σεναρίου λειτουργίας

Στη συνέχεια θα παρουσιαστεί ένα σενάριο λειτουργίας της συσκευής ReaderDevice με στόχο να γίνει κατανοητό το Object Model που παρουσιάστηκε στην 6η ενότητα. Θα δημιουργηθούν κάποια αντικείμενα του μοντέλου και θα εξηγηθεί η λειτουργία τους.

9.1 Reader Device, CommandChannel και ReadPoints

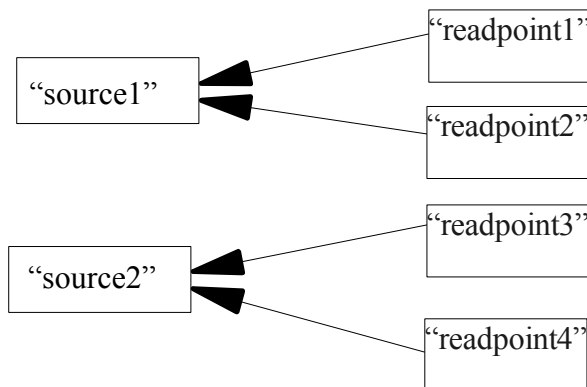
Θεωρούμε ότι η συσκευή ReaderDevice δέχεται εντολές σε ένα CommandChannel και διαθέτει 4 ReadPoints για αυτό το σενάριο. Έστω ότι τα ReadPoints έχουν τα εξής ονόματα: readpoint1, readpoint2, readpoint3, readpoint4.



Σχήμα 9.1 ReaderDevice με 4 ReadPoints

9.2 Sources

Δημιουργούμε δύο Sources με ονόματα source1 και source2. Προσθέτουμε σε κάθε Source από 2 ReadPoints.



Σχήμα 9.2 Sources

9.3 TagField

Δημιουργούμε ένα TagField με όνομα tagfield1 και χαρακτηριστικά

- tagFieldName = “tagId”, που αναφέρεται στη μνήμη του tag με αναγνωριστικό “tagId” που δεν είναι άλλο από τον κωδικό EPC.
- Offset = 0, δηλαδή τα δεδομένα που ενδιαφερόμαστε αρχίζουν από το bit στη θέση 0(από την αρχή του EPC).
- Length = 36, μας ενδιαφέρουν τα 36 bits από τη θέση του Offset και μετά.

Για συγκεκριμενοποιήσουμε το σενάριο μας παραπέρα, θεωρούμε ότι ενδιαφερόμαστε για EPCs τύπου GID-96, στον οποίο τα πρώτα 8 bits προσδιορίζουν τον τύπο του EPC, ενώ τα 28 επόμενα την εταιρία ή τον οργανισμό που κατασκευάζει το συγκεκριμένο προϊόν.

Έτσι με αυτό πλέον το TagField προσδιορίζουμε τα δεδομένα που μας ενδιαφέρουν στα tags.

	Header	General Manager Number	Object Class	Serial Number
GID-96	8	28	24	36
	0011 0101			

Πίνακας 9.3 GID-96, αποτελείται από 3 πεδία δεδομένων(General Manager Number, Object Class, Serial Number) και το πεδίο Header(00110101), που προσδιορίζει τον τύπο αυτό

9.4 TagSelectors

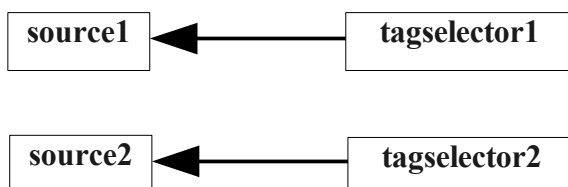
Δημιουργούμε δύο TagSelectors με ονόματα tagselector1 και tagselector2 και αρχικές τιμές:

Όνομα	TagField	Value	Mask	InclusiveFlag
tagselector1	tagfield1	“35AC36021”	“FFFFFFFF”	true
tagselector2	tagfield1	“35B6112E6”	“FFFFFFFF”	true

Πίνακας 9.4 TagSelectors

Οι δύο αυτοί TagSelectors φιλτράρουν τα tags με βάση τα δεδομένα στη θέση που ορίζει το TagField και τιμή της θέσης αυτής “35AC36021”. Η τιμή mask “FFFFFFFF” όπως εξηγείται στην ενότητα 6.5 προσδιορίζει ότι όλα τα bits της Value θα χρησιμοποιηθούν στο φίλτρο. Έτσι τα 2 πρώτα ψηφία “35” σημαίνουν ότι μας ενδιαφέρουν EPCs τύπου GID-96 και τα υπόλοιπα 7 σημαίνουν την εταιρία που μας ενδιαφέρει. Το InclusiveFlag θεωρεί ότι και οι δύο είναι inclusive(περισσότερα στην 6.5).

Θέτουμε τον tagselector1 στη source1 και τον tagselector2 στη source2.



Σχήμα 9.5 TagSelectors

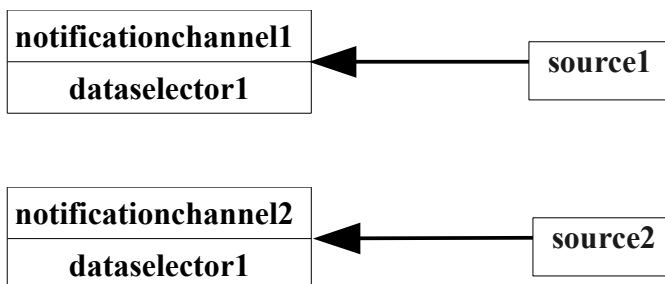
9.5 DataSelector

Δημιουργούμε ένα DataSelector με όνομα `dataselector1`. Προσθέτουμε τα εξής χαρακτηριστικά:

- `EventFilters`: `isObserved`, `isLost`, δηλαδή μας ενδιαφέρουν τα γεγονότα αυτά μόνο και φιλτράρονται τα υπόλοιπα.
- `FieldNames`: `tagID`, `eventType`, `eventTimeUTC`, `sourceName`, δηλαδή μόνο αυτές οι πληροφορίες μας ενδιαφέρουν για κάθε event.

9.6 NotificationChannels

Δημιουργούμε δύο NotificationChannels με ονόματα `notificationchannel1` και `notificationchannel2`. Στο πρώτο προσθέτουμε την `source1` και στο δεύτερο την `source2`. Σαν DataSelector των NotificationChannels θέτουμε τον `dataselector1`.



Σχήμα 9.6 NotificationChannels

9.7 Triggers

Δημιουργούμε δύο Triggers με ονόματα `trigger1` και `trigger2` και χαρακτηριστικά:

Trigger	Τύπος(TriggerType)	Value
<code>trigger1</code>	<code>timer</code>	<code>ms=1000</code>
<code>trigger2</code>	<code>timer</code>	<code>ms=60000</code>

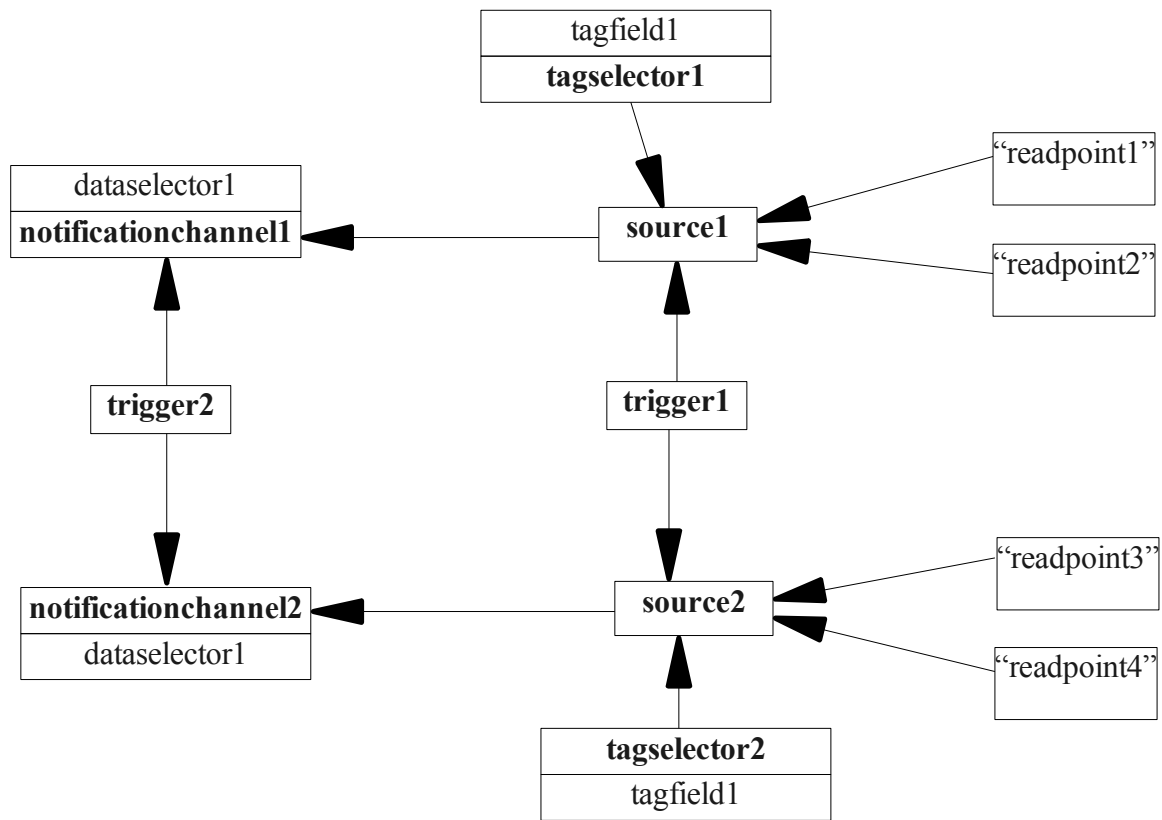
Πίνακας 9.7 Triggers

Ο τύπος `timer` σημαίνει ότι ένα Trigger θα “εκτελείται” κάθε Value ms. Την εκτέλεση του Trigger θα την καλούμε `fire`. Το `trigger1` προστίθεται στις δύο Sources και το `trigger2` στα NotificationChannels.



Σχήμα 9.8 Triggers

9.8 Διάγραμμα Αντικειμένων Σεναρίου

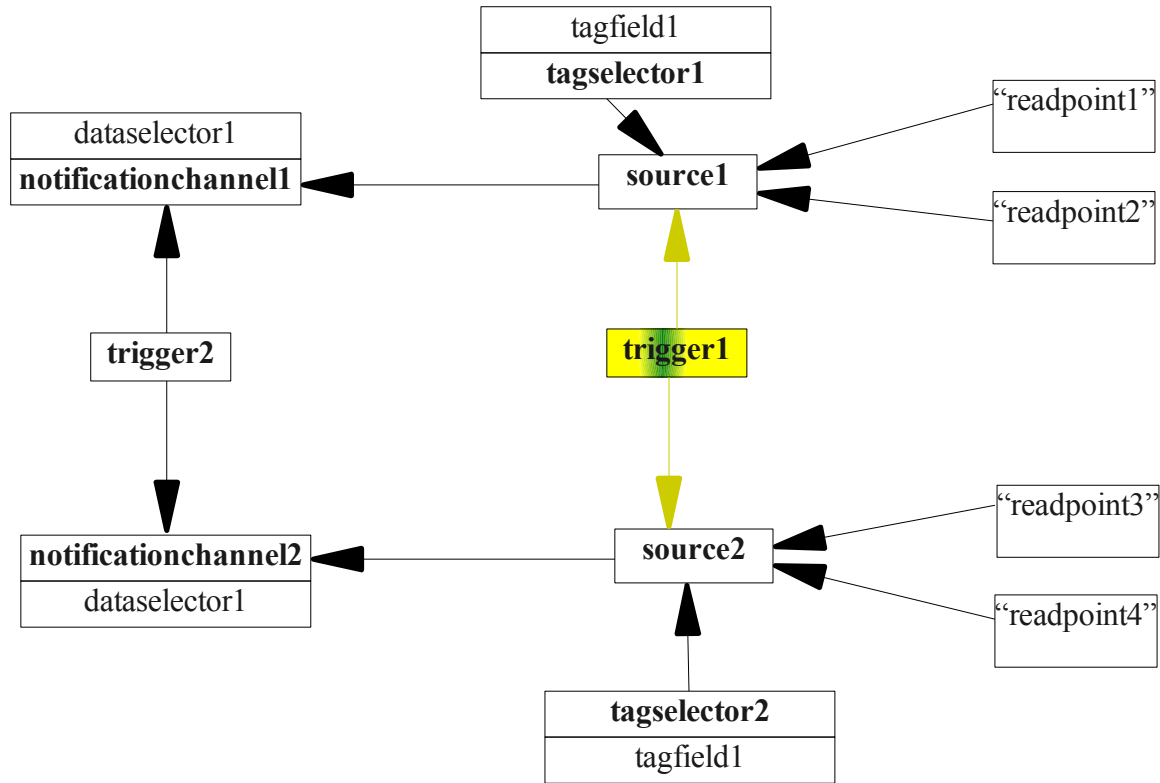


Σχήμα 9.9 Διάγραμμα αντικειμένων σεναρίου

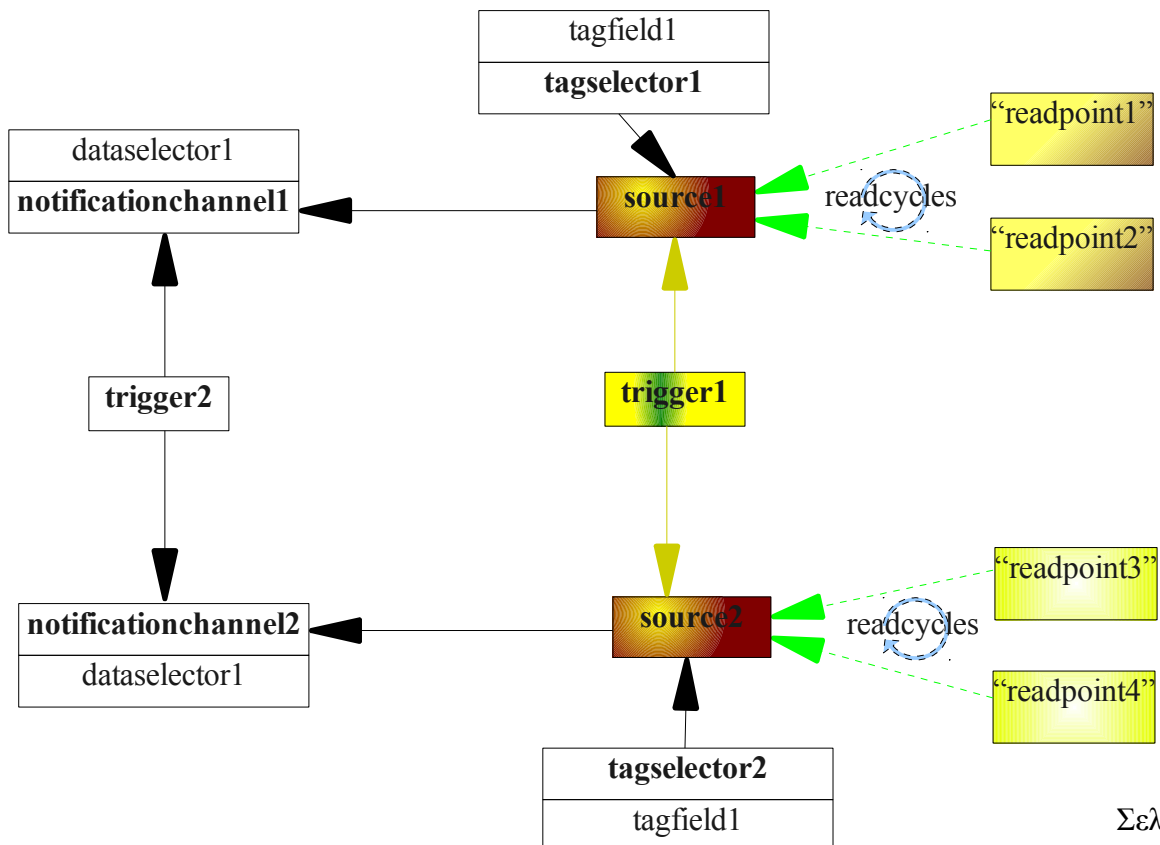
9.9 Έναρξη σεναρίου – 1η φάση

Αφού ολοκληρώσει ο Host την κλήση των εντολών στη ReaderDevice για την κατασκευή των παραπάνω αντικειμένων και του προσδιορισμού των σχέσεων μεταξύ τους, μόλις τα trigger προστεθούν σε Sources ή NotificationChannels ενεργοποιούνται και ξεκινά η εκτέλεση του σεναρίου. Έτσι ακολουθούν τα εξής στάδια:

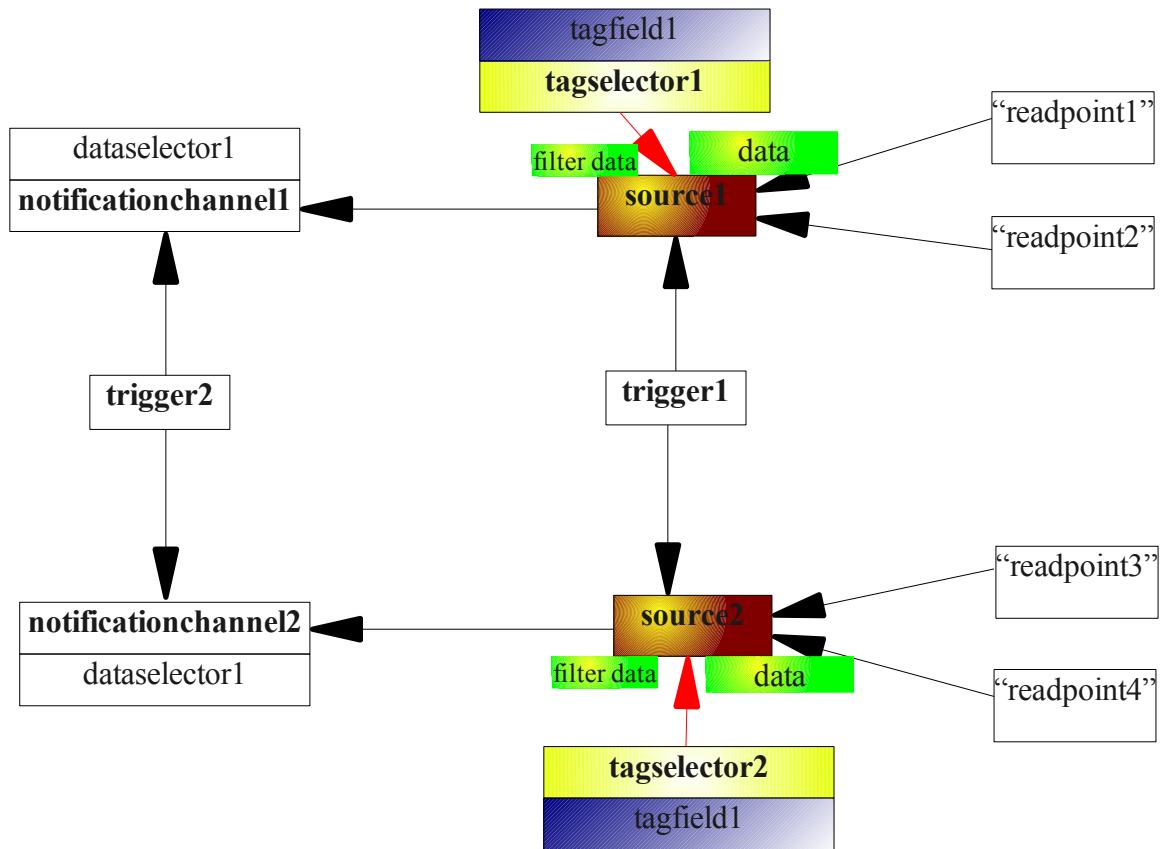
- Μετά από 1000ms το trigger1 κάνει fire και προκαλεί την ανάγνωση δεδομένων από τα ReadPoints των Sources.



Σχήμα 9.10 Το trigger1 κάνει fire(επάνω) και αμέσως προκαλεί την ανάγνωση δεδομένων(κάτω)



Όταν το trigger κάνει fire, αρχίζει η εκτέλεση μιας σειράς από readcycles στις δύο Sources. Σε κάθε readcycle γίνεται ανάγνωση από τα ReadPoints που έχει η Source. Τα διαδοχικά readcycle οδηγούν στην παραγωγή των events των tags που ανιχνεύονται(με τον τρόπο που αναφέρεται στο σχήμα 5.3). Ταυτόχρονα γίνεται φιλτράρισμα των tags με βάση τους TagSelectors κάθε Source. Το φίλτρο του TagSelector εφαρμόζεται στην περιοχή που ορίζει το περιεχόμενο σ' αυτόν TagField.

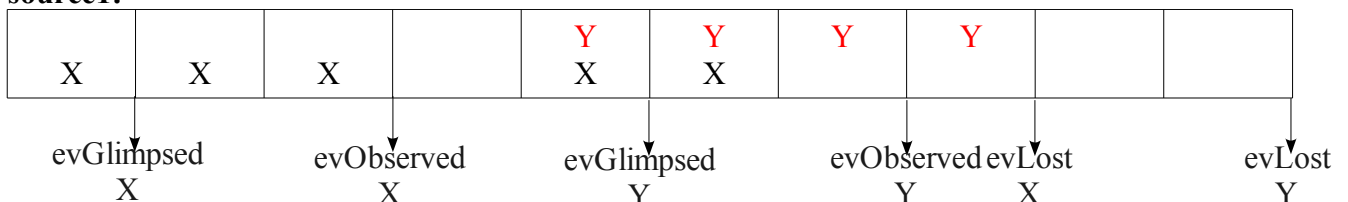


Σχήμα 9.11 Φιλτράροντας τα events με βάση τα TagSelectors κάθε Source.

Όταν ολοκληρωθεί ο αριθμός των readcycles, οι πληροφορίες από τα tags που έχουν απομείνει αποθηκεύονται στο ReportBuffer των NotificationChannels που οι Sources ανήκουν.

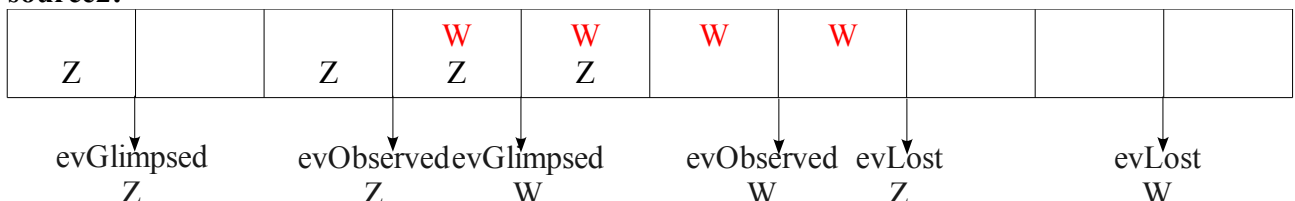
Ας υποθέσουμε ότι τα είδη των events που παράγει η ReaderDevice είναι evGlimped, evObserved και evLost και ότι έχουμε τα παρακάτω διαγράμματα events για τις δύο Sources σε ένα σύνολο 10 readcycles:

source1:



όπου X = 35AC36021012345678901234567 και Y = 35BE36029012345678901234567

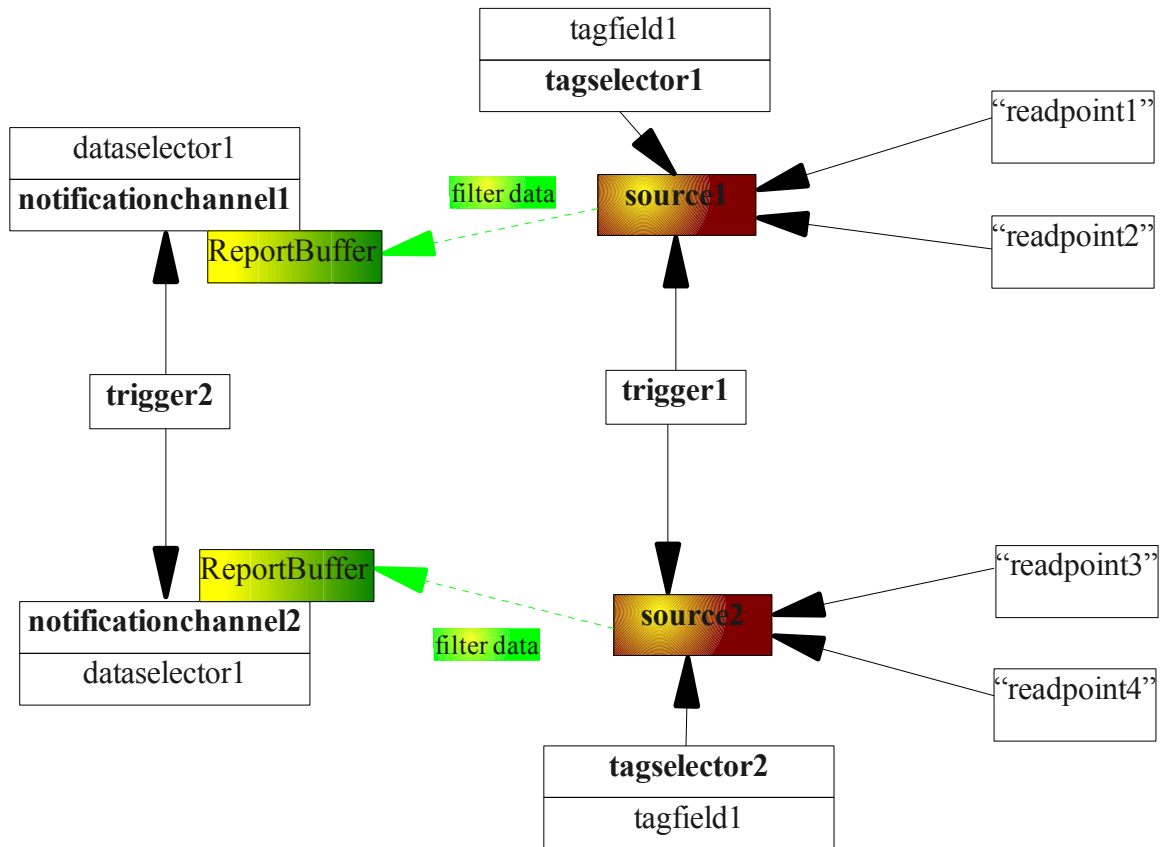
source2:



όπου Z = 35B6112E6012345678901234567 και W = 354E40C90012345678901234567

Σχήμα 9.12 Διάγραμμα events σεναρίου

Τα tags Y και W απορρίπτονται με βάση τους tagselector1 και tagselector2 αντίστοιχα.

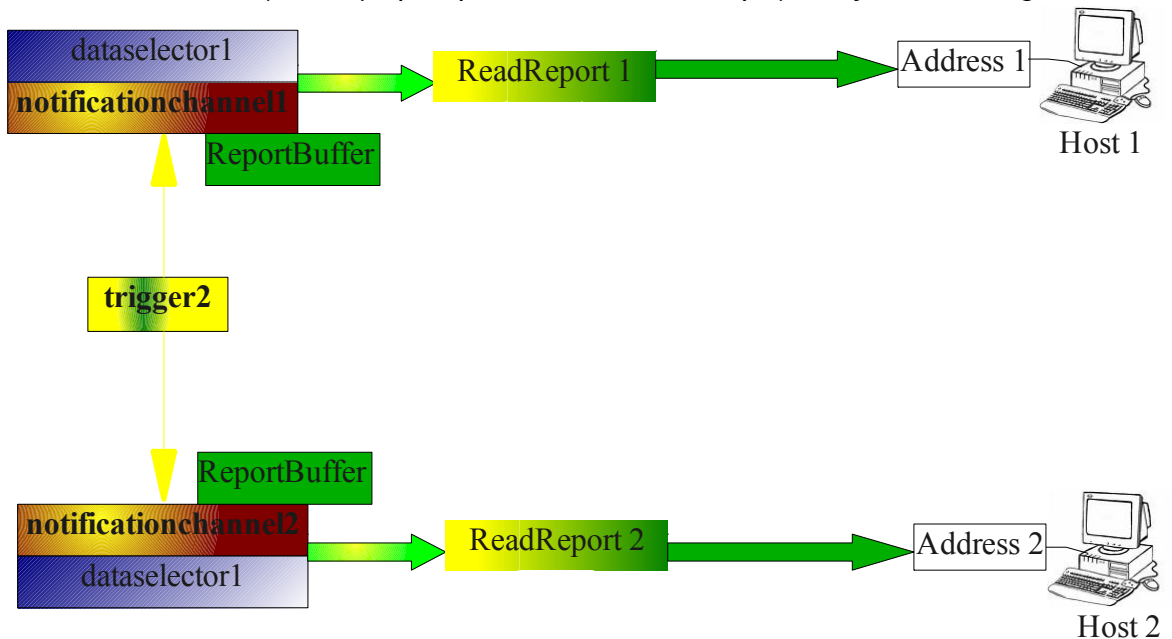


Σχήμα 9.13 Αποθήκευση δεδομένων στα ReportBuffer των NotificationChannels

Κάθε 1000ms το trigger1 κάνει fire και επαναλαμβάνεται η ίδια διαδικασία. Έτσι συσσωρεύονται συνεχώς νέα δεδομένα στο ReportBuffer.

9.10 Αποστολή events στον Host – 2η φάση

Η 2η φάση αρχίζει μόλις το trigger2 κάνει fire μετά από 60000ms και προκαλεί την αποστολή των δεδομένων στις διευθύνσεις των NotificationChannels. Η επιλογή των πληροφοριών για τα events που θα σταλούν γίνεται με βάση τον dataselector1, παράγοντας ένα ReadReport.



Σχήμα 9.14 Αποστολή events στους Hosts μετά από fire του trigger2

Από τα events που υποθέσαμε στην 1η φάση επιλέγονται να σταλούν τα evObserved και evLost με πληροφορίες κάθε event: {tagID, eventType, eventTimeUTC, sourceName} σύμφωνα με τα χαρακτηριστικά του dataselector1 που έχουμε ορίσει στα NotificationChannels.

notificationchannel1:

ReportBuffer: tag X events: {evGlimpsed, evObserved, evLost}.

Μετά την εφαρμογή του dataselector1 παράγεται το παρακάτω ReadReport:

tagID	eventType	eventTimeUTC	sourceName
35AC36021012345678 901234567	evObserved	2009-11- 10T20:15:05.123	mysource1
35AC36021012345678 901234567	evLost	2009-11- 10T20:15:05.127	mysource1

Πίνακας 9.15 ReadReport του notificationchannel1

notificationchannel2:

ReportBuffer: tag Z events: {evGlimpsed, evObserved, evLost}.

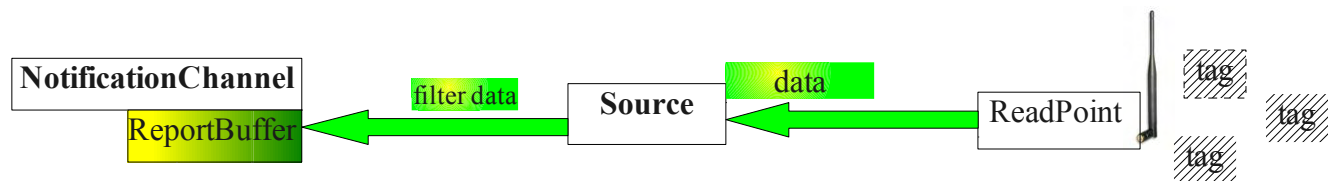
Μετά την εφαρμογή του dataselector1 παράγεται το παρακάτω ReadReport:

tagID	eventType	eventTimeUTC	sourceName
35B6112E6012345678 901234567	evObserved	2009-11- 10T20:15:05.143	mysource2
35B6112E6012345678 901234567	evLost	2009-11- 10T20:15:05.147	mysource2

Πίνακας 9.16 ReadReport του notificationchannel2

9.11 Ροή πληροφορίας

Οι φάσεις του σεναρίου επαναλαμβάνονται κάθε φορά που τα Triggers κάνουν fire. Στο σενάριο μας, στην 1η φάση η ροή της πληροφορίας είναι:



Σχήμα 9.17 Ροή πληροφορίας όταν κάνει fire ένα ReadTrigger

ενώ στη 2η είναι:



Σχήμα 9.18 Ροή πληροφορίας όταν κάνει fire ένα NotificationTrigger

10. Messaging / Transport Binding(MTB)

Το επίπεδο Reader Layer, που παρουσιάστηκε στην ενότητα 5, περιγράφει τις λειτουργίες που πρέπει ή μπορεί να υποστηρίζει ένας Reader σύμφωνα με το μοντέλο αντικειμένων που παρουσιάστηκε στην ενότητα 6. Μέσω της client διεπαφής Host έχει την ψευδαίσθηση ότι διαθέτει τα αντικείμενα τοπικά και εκτελεί τοπικά τις εντολές. Όμως υπάρχουν δύο επίπεδα επικοινωνίας με τα οποία επιτυγχάνεται η επικοινωνία μεταξύ Reader και ενός ή περισσότερων Host.

10.1 Messaging Layer

Όπως έχει ήδη αναφερθεί στην ενότητα 4.2, στο επίπεδο αυτό προσδιορίζεται ο τρόπος που τα μηνύματα και οι εντολές που ορίζονται στο Reader Layer μορφοποιούνται για αποστολή στα κανάλια επικοινωνίας.

Υπάρχουν δύο μορφές μηνυμάτων που υποστηρίζει το πρωτόκολλο, μορφή XML και μορφή Text. Για την αναφορά σε ένα αντικείμενο του μοντέλου της ενότητας 6, χρησιμοποιείται το είδος και το όνομα ενός αντικειμένου. Παρακάτω θα παρουσιαστούν οι δύο αυτές μορφοποιήσεις, με σύντομο τρόπο και χωρίς να περιγράφονται πλήρως. Περισσότερες λεπτομέρειες μπορούν να αναζητηθούν στο έγγραφο του πρωτοκόλλου στο διαδικτυακό τόπο του EPCglobal.

10.1.1 Text

Η μορφή της εντολής που στέλνει ο Host είναι:

- **[Τύπος αντικειμένου]#[Όνομα αντικειμένου].[Όνομα εντολής] [παράμετροι] LF**
πχ.

→ Source#source1.getAllReadPoints LF

→ SRC#source1.aRP {readpoint1, readpoint2, readpoint3} LF

στο 2ο παράδειγμα το SRC αποτελεί συντόμευση του Source και το aRP της εντολής addReadPoints. LF είναι ο χαρακτήρας τέλους γραμμής.

Η μορφή της απάντησης είναι:

- OK LF

[return value] LF

ή σε περιπτώσεις λάθους

- ERR,[error-code],[error-description] LF

πχ.

→ OK LF

→ OK LF

readpoint2 LF

readpoint3 LF

→ ERR,000E,ERROR_READPOINT_NOT_FOUND

10.1.2 XML

Η μορφή της εντολής που στέλνει ο Host είναι:

- **<command>**

<id>1234</id>

<targetName>Όνομα αντικειμένου</targetName>

<Τύπος αντικειμένου>

<Όνομα εντολής>

<parameter>value</parameter>

.....

</Όνομα εντολής>

</Τύπος αντικειμένου>

</command>

πχ.

```
→ <command>
  <id>1234</id>
  <targetName>source1</targetName>
  <source>
    <getAllTagSelectors>
    </getAllTagSelectors>
  </source>
</command>
```

Η μορφή της απάντησης είναι:

```
➤ <reply>
  <id>1234</id>
  <resultCode>0</resultCode>
  <Τύπος αντικειμένου>
    <Όνομα εντολής>
      <returnValue>value</returnValue>
    </Όνομα εντολής>
  </Τύπος αντικειμένου>
</reply>
```

πχ.

```
→ <reply>
  <id>1234</id>
  <resultCode>0</resultCode>
  <source>
    <getAllTagSelectors>
      <returnValue>
        <list>
          <value>tagSelector1</value>
          <value>tagSelector2</value>
        </list>
      </returnValue>
    </getAllTagSelectors>
  </source>
</reply>
```

Για τον έλεγχο της σύνταξης των XML μηνυμάτων έχει εκδοθεί ένα XML schema το οποίο περιέχεται στο έγγραφο του πρωτοκόλλου.

10.2 Transport Layer

Αφού έχει παραχθεί η μορφή των μηνυμάτων ακολουθεί η αποστολή του στο Transport Layer. Ανάλογα με τον τρόπο διασύνδεσης, υποστηρίζονται κάποιοι μηχανισμοί για την επιτυχή επικοινωνία μεταξύ Host και Reader.

Όπως έχει αναφερθεί το πρωτόκολλο περιγράφει 3 τρόπους διασύνδεσης μεταξύ Reader και Host, serial, tcp ή http. Παρακάτω γίνεται μια συνοπτική περιγραφή τους.

Και για τους 3, υποστηρίζεται προαιρετικός μηχανισμός επιβεβαίωσης των μηνυμάτων που ανταλλάσσονται. Επίσης και στους 3, προστίθεται υποχρεωτικά ως πεδίο του Header το μέγεθος του μηνύματος που θα αποσταλεί.

10.2.1 Serial

Αφού η εντολή που έχει μορφοποιηθεί σε XML ή Text μορφή, πριν την αποστολή της στο serial κανάλι προστίθεται ένα πεδίο Header και ένα Trailer. Η μορφή που προκύπτει είναι:

<Header><msg content><Trailer>

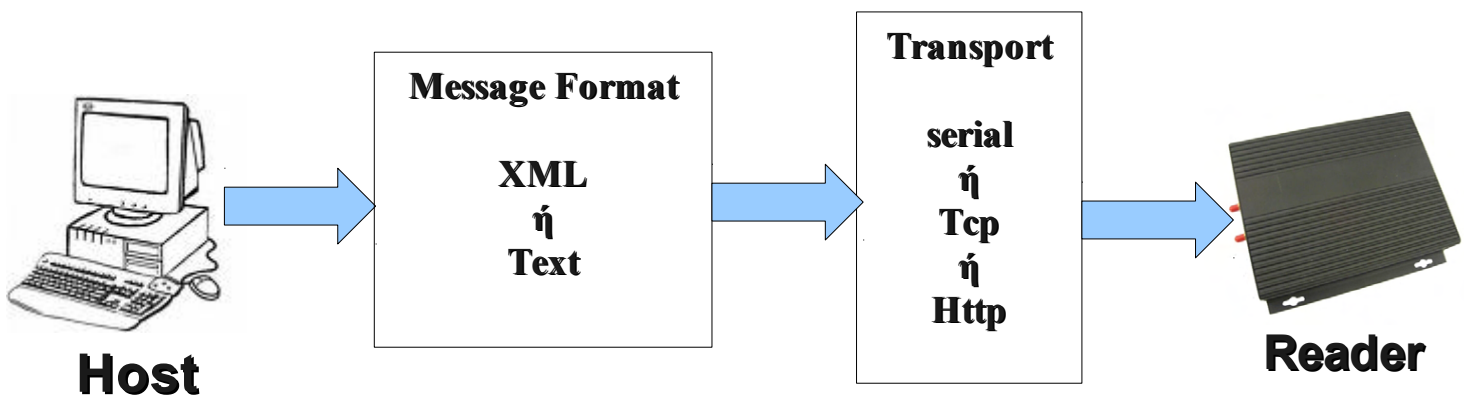
Πληροφορίες για τα περιεχόμενα των Header και Trailer βρίσκονται στο έγγραφο του πρωτοκόλλου.

10.2.2 Tcp

Στην tcp μορφή προστίθεται ένα πεδίο Header. Πληροφορίες για τα περιεχόμενα του Header βρίσκονται στο έγγραφο του πρωτοκόλλου.

10.2.3 Http

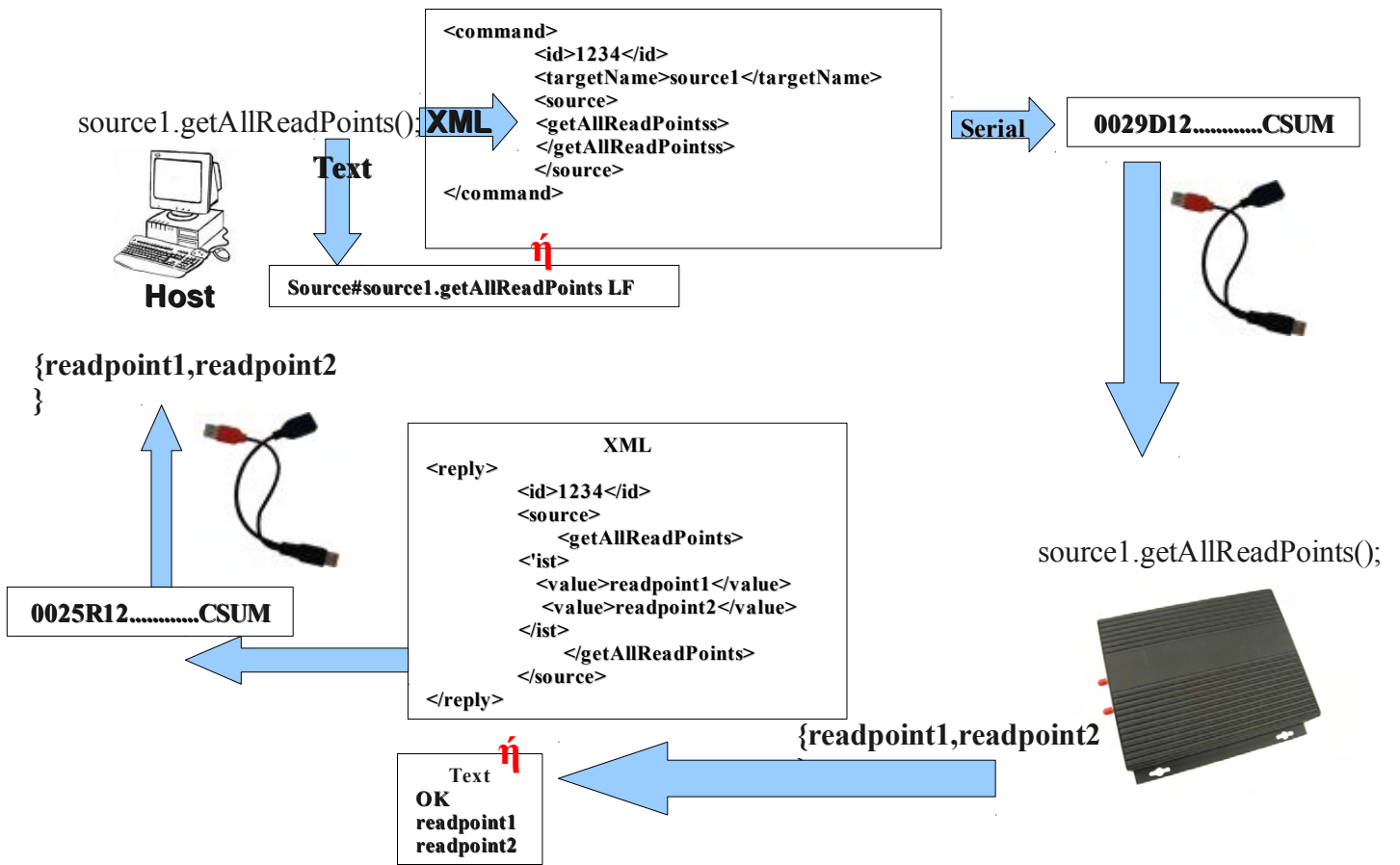
Όταν η επικοινωνία γίνεται μέσω του πρωτοκόλλου http, χρησιμοποιούνται ένας αριθμός Header, άλλα υποχρεωτικά κι άλλα προαιρετικά. Πληροφορίες για τα Header βρίσκονται στο έγγραφο του πρωτοκόλλου.



Σχήμα 10.1 Messaging / Transport Binding

10.3 Σενάριο εντολής

Στο παρακάτω διάγραμμα παρουσιάζεται το σενάριο της εντολής `getAllReadPoints` μίας Source με όνομα `source1` με serial Transport Layer. Ενδεικτικά παρουσιάζονται και οι δύο μορφές Messaging Layer, στην πραγματικότητα μόνο μία χρησιμοποιείται για κάθε Channel.



Σχήμα 10.2 Σενάριο της εντολής `getAllReadPoints` μίας Source.

Γλωσσάριο

RFID: Radio Frequency Identification , ταυτοποίηση μέσω ραδιοσυχνοτήτων

tag: ετικέτα ταυτοποίησης

Reader Device: Συσκευή ανάγνωσης/εγγραφής ετικετών ταυτοποίησης (tag).

EPC: Electronic Product Code, Ηλεκτρονικός Κωδικός Προϊόντος. Πρόκειται για τον τρόπο αναπαράστασης του κωδικού των EPC tags που αντιστοιχεί σε κάποιο εμπόρευμα.

EPCglobal: παγκόσμιος οργανισμός για την προώθηση και την εμπορική ανάπτυξη της τεχνολογίας EPC.

Barcode: γραμμωτός κώδικας, διαδοχικές άσπρες και μαύρες γραμμές που αντιστοιχούν στα λογικά 0 και 1.

UPC: Universal Product Code, καθολικός κωδικός προϊόντος, 12-ψήφια μορφή γραμμωτού κώδικα που χρησιμοποιείται για τη σήμανση προϊόντων (χρησιμοποιείται κυρίως σε ΗΠΑ και Καναδά)

EAN: European Article Number, Ευρωπαϊκός Αριθμός Προϊόντος, το αντίστοιχο πρότυπο του UPC για την Ευρώπη

Uniform Code Council: Συμβούλιο Καθολικού Κώδικα, μη κερδοσκοπικός οργανισμός υπεύθυνος για την γραμμωτό κώδικα UPC

International EAN: μη κερδοσκοπικός οργανισμός υπεύθυνος για τον γραμμωτό κώδικα στην Ευρώπη

EPCIS: EPC Information System, σύστημα βάσεων δεδομένων προσπελάσιμο μέσω του διαδικτύου για την παροχή πληροφοριών για προϊόντα που έχουν σημειωθεί με κάποιο EPC κωδικό.

IFF: Identification friend or foe, σύστημα ταυτοποίησης εχθρικών αεροσκαφών μέσω ραδιοσυχνοτήτων που αναπτύχθηκε στο 2ο παγκόσμιο πόλεμο.

EAS: Electronic article surveillance, αντικλεπτικό σύστημα που αναπτύχθηκε τη δεκαετία του 70.

Πηγές

- **Οργανισμός EPCglobal**, ιστοσελίδα: www.epcglobalinc.org , έγγραφο πρωτοκόλλου Reader Protocol Standard Version 1.1 <http://www.epcglobalinc.org/standards/rp>
- **Wikipedia**, Online εγκυκλοπαίδεια ,
1) [RFID](#)
- **Jeremy Landt, Shrouds of Time the history of RFID**, An AIM Publication, 2001.
- **RFID Journal** <http://www.RFIDjournal.com>, articles:
1) [The History of RFID Technology](#),
2) [A Summary of RFID Standards](#)
- **RFID Consultation** <http://www.RFIDconsultation.eu>, άρθρα:
1) [Interoperability article](#)
- **Fosstrak: Open Source RFID Software Platform** , άρθρα:
1) [RFID Application Development With the Accada Middleware Platform](#)
παρουσιάσεις:
1) [Fosstrak, Free and Open-Source Software for Track and Trace](#)
2) [How to Prepare for EPC Data Cost-Effectively](#), Christian Floerkemeier – MIT Auto-ID Lab
- **η-Επιχειρείν** , [πληροφορίες για τα RFID](#)