



ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ ΚΑΙ ΔΙΟΙΚΗΣΗΣ

Διεπαφή τεχνολογίας NFC με αυτόματα
συστήματα.

Interfacing NFC to automation
systems.

Γρηγόριος Σιδηράς

Επιβλέπων καθηγητής: Κιζήρογλου Μιχαήλ

Abstract

The objective of this final year project is to develop an NFC interface for relay control, suitable for automation systems. In chapter 2, reference is made to the technologies of RFID and NFC. Also, the capabilities of each technology, their common characteristics and the differences between them are mentioned. In addition, the operation of the MFRC522 chip was studied, as well as the function of the reader, in which it is integrated. A description of the operation of the PN532 chip and a comparison with the MFRC522 are also presented. In addition, NDEF messages were studied and in the last section of this chapter, the functionality of four different types of NFC reader is analyzed. Chapter 3 contains the RFID project and is based on reader RC522. This project was created for the practical application and study of RFID technology. Chapter 4 features an NFC project, based on the PN532 V3 reader. This was created to study NFC technology practically and to understand the sending and reading of NDEF messages. Also, in chapter 5, the comparison of the 2 projects was made and the conclusions from this comparison were reported. Finally, in chapter 6, preliminary work on NFC activation and control using a mobile phone is presented. This goal was not finally realized in the present dissertation. Nevertheless, in the same chapter, two solutions are proposed, through which this goal could potentially be realized successfully.

Περίληψη

Ο στόχος της διπλωματικής ήταν, η ανάπτυξη μιας διεπαφής τεχνολογίας NFC, κατάλληλη για συστήματα αυτοματισμού. Στο κεφάλαιο 2, γίνεται αναφορά στις τεχνολογίες του RFID και του NFC. Επίσης, αναφέρονται οι δυνατότητες που έχει η κάθε τεχνολογία, τα κοινά χαρακτηριστικά τους και οι διαφορές μεταξύ τους. Επιπλέον, έγινε μελέτη της λειτουργίας του chip MFRC522, καθώς και της λειτουργίας της συσκευής ανάγνωσης, όπου είναι ενσωματωμένο. Επίσης αναφέρθηκε, η λειτουργία του chip PN532 και έγινε σύγκριση με το MFRC522. Επιπλέον, έγινε μελέτη των NDEF μηνυμάτων και στην τελευταία ενότητα αυτού του κεφαλαίου, παρουσιάζονται τέσσερις NFC συσκευές ανάγνωσης, όπου γίνεται ανάλυση των λειτουργιών τους. Στο κεφάλαιο 3 παρουσιάζεται η βασική πειραματική μελέτη της διπλωματικής εργασίας που βασίζεται στην συσκευή ανάγνωσης RC522. Το project αυτό δημιουργήθηκε, για την πρακτική εφαρμογή και μελέτη της RFID τεχνολογίας. Στο κεφάλαιο 4 βρίσκεται ένα NFC project, το οποίο βασίζεται στον συσκευή ανάγνωσης PN532 V3. Αυτό δημιουργήθηκε, για την μελέτη της NFC τεχνολογίας πρακτικά και την κατανόηση της αποστολής και του διαβάσματος των NDEF μηνυμάτων. Επίσης, στο κεφάλαιο 5, έγινε η σύγκριση των 2 project και αναφέρθηκαν τα συμπεράσματα από την σύγκριση αυτή. Τέλος, στο κεφάλαιο 6, γίνεται αναφορά σε μια προσπάθεια που έγινε, η οποία ήταν: μέσω του NFC του κινητού μας να στείλουμε ένα NDEF μήνυμα σε μια συσκευή ανάγνωσης. Ο στόχος αυτός τελικά δεν πραγματοποιήθηκε στην παρούσα πτυχιακή εργασία, όμως σε αυτό το κεφάλαιο προτείνονται 2 λύσεις, μέσω των οποίων μπορεί να πραγματοποιηθεί αυτός ο στόχος με επιτυχία.

Ευχαριστίες

Ευχαριστώ θερμά, τον επιβλέποντα καθηγητή κ. Κιζήρογλου για την εμπιστοσύνη στην ανάθεση του θέματος και για την πολύτιμη βοήθεια του στην εκπόνηση της πτυχιακής, καθώς και την οικογένεια μου για την στήριξη τους.

Πίνακας περιεχομένων

Abstract	i
Περίληψη	ii
Ευχαριστίες	iii
Πίνακας περιεχομένων.....	iv
Επεξήγηση ακρωνύμων.....	vi
1 Εισαγωγή και στόχοι.....	1
1.1 Εισαγωγή.....	1
1.2 Στόχοι	1
2 Τεχνολογία RFID και NFC	2
2.1 Η λειτουργία και η σχέση μεταξύ τους	2
2.1.1 NFC	2
2.1.2 RFID	4
2.1.3 Αρχιτεκτονική του NFC και η σχέση με το RFID	6
2.1.4 NDEF.....	8
2.2 ΛΕΙΤΟΥΡΓΙΑ ΤΩΝ 2 CHIP	13
2.2.1 Η συσκευή ανάγνωσης RC522 και το chip του MFRC522	13
2.2.2 PN532	17
2.3 Διαφορές των 2 chip.....	19
2.4 NFC συσκευές ανάγνωσης.....	20
2.4.1 PN532 V3	21
2.4.2 PN532 V4	23
2.4.3 Adafruit PN532 RFID/NFC Breakout και Shield	24
2.4.4 NFC Shield V2.0	27
3 RFID σύστημα	29
3.1 Σχεδίαση του κυκλώματος	29
3.1.1 Περιγραφή των υλικών.....	29
3.1.2 Κόστος κατασκευής.....	33
3.2 Λειτουργία RFID συστήματος	34
3.3 ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ RFID ΣΥΣΤΗΜΑΤΟΣ.....	37
3.3.1 MIT app Inventor.....	37

3.3.2 Κώδικας Arduino IDE	42
4 NFC σύστημα.....	53
4.1 Δομή βιβλιοθήκης Arduino	53
4.2 Σχεδίαση του κυκλώματος	56
4.2.1 Περιγραφή των υλικών.....	56
4.2.2 Κόστος υλικών.....	57
4.3 Λειτουργία NFC συστήματος	58
4.4 ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ NFC ΣΥΣΤΗΜΑΤΟΣ.....	64
5 Σύγκριση των 2 συστημάτων.....	72
5.1 Πλεονεκτήματα-μειονεκτήματα των συστημάτων.....	72
5.2 Συμπεράσματα.....	73
6 Συμπεράσματα και προτάσεις λύσης	74
7.Βιβλιογραφία	77

Επεξήγηση ακρωνύμων

NFC	Near Field Communication
RFID	Radio Frequency Identification
NDEF	NFC Data Exchange Format
ISO	International Organization for Standardization
P2P	Peer-to-Peer
LLCP	Logical Link Control Protocol
SNEP	Simple NDEF Exchange Protocol
TNF	Type Name Format
RTD	Record Type Definition
URN	Uniform Resource Name
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
SPI	Serial Peripheral Interface
I2C	Inter-integrated-circuit
UART	Universal Asynchronous Reception and Transmission

1 Εισαγωγή και στόχοι

1.1 Εισαγωγή

Η τεχνολογία RFID, είναι μια τεχνολογία αναγνώρισης, η οποία χρησιμοποιείται για έλεγχο πρόσβασης, αλλά και στην βιομηχανία. Με την τεχνολογία αυτή, τα προϊόντα έχοντας ένα chip μπορούν, να διαβάζονται από μια συσκευή ανάγνωσης και με αυτό τον τρόπο, για παράδειγμα, να ξέρουμε ακριβώς πόσα προϊόντα υπάρχουν στην αποθήκη ή πόσα προϊόντα φορτώθηκαν σε ένα φορτηγό. Από την άλλη μεριά, έχουμε την τεχνολογία NFC, η οποία βασίζεται πάνω στο RFID, όμως έχοντας κάποιες διαφορές. Η τεχνολογία αυτή χρησιμοποιείται για συναλλαγές μέσω του κινητού μας τηλεφώνου, ή με οποιαδήποτε άλλη συσκευή που διαθέτει NFC. Υπάρχουν και άλλες δυνατότητες που έχουμε με το NFC, όπως να ξεκλειδώσουμε μια κλειδαριά, για παράδειγμα του αυτοκινήτου, παρ όλα αυτά, περισσότερο χρησιμοποιείται για πληρωμές προς το παρόν και είναι μια αρκετά υποσχόμενη τεχνολογία για το μέλλον.

1.2 Στόχοι

Οι στόχοι της συγκεκριμένης πτυχιακής εργασίας ήταν: η μελέτη της τεχνολογίας RFID και NFC και η κατανόηση αυτών, μέσω 2 πρακτικών εφαρμογών. Επίσης, ένας ακόμη στόχος ήταν το διάβασμα ενός NDEF μηνύματος από το κινητό τηλέφωνο.

2 Τεχνολογία RFID και NFC

ΣΚΟΠΟΣ

Ο σκοπός του 2^{ου} κεφαλαίου, είναι η εισαγωγή στις έννοιες του RFID και του NFC, όπως επίσης και η σχέση που έχουν μεταξύ τους, καθώς και οι διαφορές που υπάρχουν ανάμεσα τους. Επιπλέον, θα γίνει αναφορά στο RFID chip MFRC522 αλλά και στην συσκευή ανάγνωσης που είναι ενσωματωμένο αυτό το chip, όπως επίσης και στο NFC chip PN532, το οποίο είναι ενσωματωμένο σε αρκετές NFC συσκευές ανάγνωσης, 4 από τις οποίες θα αναφερθούν σε αυτό το κεφάλαιο και είναι ο PN532 η έκδοση V3 αλλά και η V4, ο PN532 NFC/RFID Breakout και Shield και η NFC/RFID συσκευή ανάγνωσης, NFC Shield V2.0.

2.1 Η λειτουργία και η σχέση μεταξύ τους

2.1.1 NFC

Το NFC είναι μια τεχνολογία που βασίζεται στο RFID και λειτουργεί στα 13.56 MHz, ωστόσο με το NFC μπορούμε να έχουμε μια πιο σύνθετη ανταλλαγή δεδομένων σε σχέση με το RFID. Επίσης, μια NFC συσκευή ανάγνωσης μπορεί να διαβάζει και να γράφει RFID ετικέτες ή κάρτες, όπως και επίσης να επικοινωνεί με συσκευές που διαθέτουν NFC. Είναι χαμηλής εμβέλειας και για να επιτευχτεί η επικοινωνία μεταξύ 2 συσκευών NFC, χρειάζεται να έρθουν σε επαφή και να βρίσκονται σε απόσταση μικρότερη των 5_cm, έτσι με αυτόν τον τρόπο θα έχουν μια σύντομη επικοινωνία και θα ανταλλάξουν μεταξύ τους δεδομένα κατευθείαν, χωρίς να χρειάζεται να έχουν σύζευξη μεταξύ τους όπως γίνεται με το Bluetooth. Όμως, το NFC είναι τεχνολογία χαμηλότερης ταχύτητας σε σχέση με το Bluetooth και το WIFI, οπότε εάν θέλουμε ανταλλαγή δεδομένων μεταξύ των συσκευών μας σε υψηλότερες ταχύτητες, θα χρησιμοποιήσουμε αυτές τις τεχνολογίες και όχι το NFC [1].

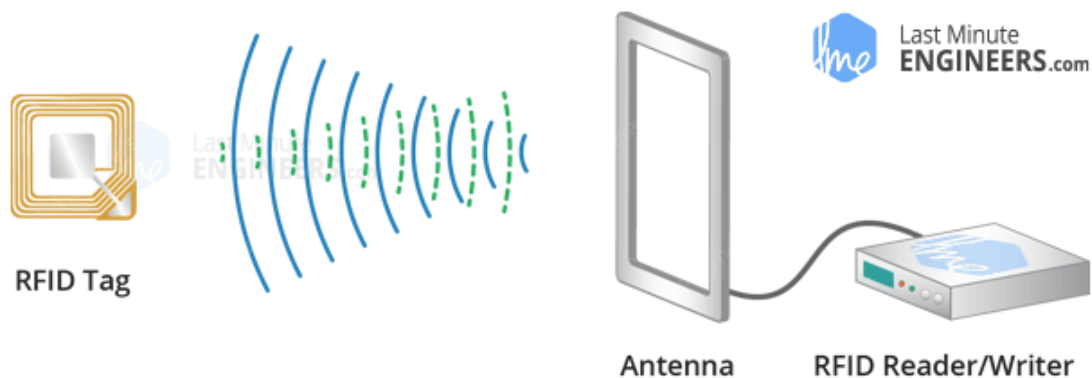
Οι συσκευές NFC διαθέτουν 6 λειτουργίες:

- **Reader / Writer mode:** Η συσκευή μέσω αυτής της λειτουργίας, μπορεί να γράφει και να διαβάζει δεδομένα ανέπαφα από ετικέτες μέσω διαφόρων εφαρμογών, τις οποίες μπορούμε να εγκαταστήσουμε στην συσκευή μας.
- **Peer-to-Peer mode:** Δύο συσκευές NFC έρχονται σε πολύ κοντινή απόσταση μεταξύ τους για την ανταλλαγή δεδομένων. Αυτή η λειτουργία, χρησιμοποιείται για την εύκολη ανταλλαγή των δεδομένων μεταξύ των 2 συσκευών.
- **Card Emulation mode:** Μια συσκευή NFC(πχ. κινητό) λειτουργεί σαν μια ανέπαφη κάρτα, ώστε να μπορεί να επικοινωνεί με μια συσκευή ανάγνωσης(NFC συσκευή ανάγνωσης). Δυο παραδείγματα αυτής της λειτουργίας είναι: η εξομοίωση ανέπαφων τραπεζικών καρτών, τις οποίες χρησιμοποιούμε για τις αγορές και η εξομοίωση ανέπαφων εισιτηρίων για τα μέσα μαζικής μεταφοράς.
- **Secure Element based Card Emulation:** Σε αυτή την λειτουργία, η εξομοίωση κάρτας ελέγχεται από ένα ασφαλές στοιχείο (secure element) μέσα στην NFC συσκευή. Το ασφαλές αυτό στοιχείο μπορεί να είναι, είτε ένα chip ασφαλείας ή μια SIM με δυνατότητα NFC. Στις 2 αυτές περιπτώσεις, τα δεδομένα που θα λαμβάνει η συσκευή από την NFC συσκευή ανάγνωσης, θα προωθούνται στο secure element.Συνεπώς, αυτή η λειτουργία μας παρέχει την δυνατότητα για συναλλαγές υψηλού επίπεδου ασφαλείας.
- **Host Card Emulation:** Σε αυτή την λειτουργία έχουμε πάλι εξομοίωση κάρτας, όμως θα πρέπει στην συσκευή μας να έχουμε εγκατεστημένη μια εφαρμογή HCE. Ο NFC ελεγκτής(controller) του κινητού μας, κάθε δεδομένο που δέχεται το μεταβιβάζει σε αυτήν την εφαρμογή, η οποία μπορεί να επικοινωνεί με μια NFC συσκευή ανάγνωσης με το NFC API.

- **Wireless Charging mode:** Με την λειτουργία αυτή μπορούμε να έχουμε μεταφορά ισχύος έως και 1 W από την συσκευή μας(έξυπνο κινητό τηλέφωνο) σε μια άλλη μικρότερη συσκευή, η οποία χρειάζεται για να λειτουργήσει μικρή παροχή ρεύματος, δηλαδή μπορούμε να φορτίσουμε τέτοιου είδους συσκευές. Μια τέτοια συσκευή μπορεί να είναι τα ασύρματα ακουστικά, αλλά και άλλες παρόμοιες συσκευές [Error! Reference source not found.](#).

2.1.2 RFID

Το RFID είναι μια τεχνολογία η οποία λειτουργεί στα 13.56 MHz και έχει σχεδιαστεί για αναγνώριση και όχι για επικοινωνία. Η τεχνολογία αυτή, αποτελείται από μια RFID συσκευή ανάγνωσης, η οποία δημιουργεί ένα ηλεκτρομαγνητικό πεδίο και μπορεί να διαβάσει παθητικές ή ενεργές ετικέτες, έχει δηλαδή 2 τρόπους επικοινωνίας(**Σχήμα 2.1**).



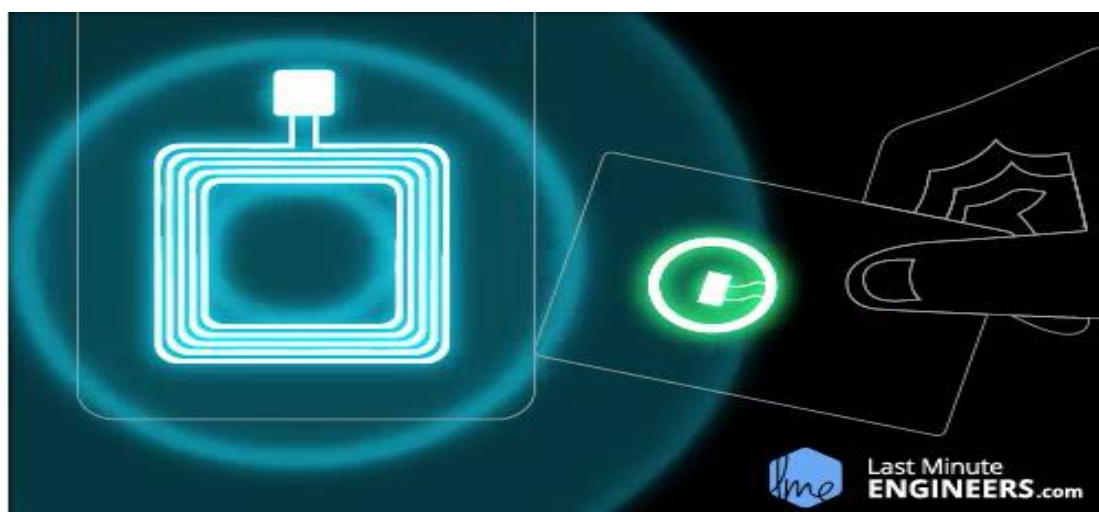
Σχήμα 2.1 Η δημιουργία ηλεκτρομαγνητικού πεδίου από μια RFID συσκευή ανάγνωσης και η επικοινωνία με μια παθητική ετικέτα. Πηγή:

<https://lastminuteengineers.com/how-rfid-works-rc522-arduino-tutorial/> **Error!**

Reference source not found..

Επιπλέον, μπορεί και να γράφει δεδομένα σε αυτές εάν διαθέτει αυτήν την δυνατότητα. Στην περίπτωση των παθητικών ετικετών, οι ετικέτες αυτές δεν διαθέτουν κάποια πηγή ενεργείας(μπαταρία),και έτσι για να λειτουργήσουν θα πρέπει να έρθουν αρκετά κοντά στην συσκευή ανάγνωσης και να βρίσκονται εντός του ηλεκτρομαγνητικού πεδίου που δημιουργεί [1].

Οι ετικέτες αυτές διαθέτουν ένα μικροσίπ, στο οποίο αποθηκεύονται δεδομένα. Επίσης έχει μια κεραία, η οποία λαμβάνει και στέλνει σήμα. Μόλις η παθητική ετικέτα πλησιάσει την συσκευή ανάγνωσης, τότε το σήμα της συσκευής ανάγνωσης λαμβάνεται από την κεραία της ετικέτας και μετατρέπεται σε ηλεκτρικό ρεύμα και έπειτα τροφοδοτείται το chip της(Σχήμα 2.2).



Σχήμα 2.2 Ενεργοποίηση του chip της παθητικής ετικέτας. Πηγή: <https://lastminuteengineers.com/how-rfid-works-rc522-arduino-tutorial/> [3].

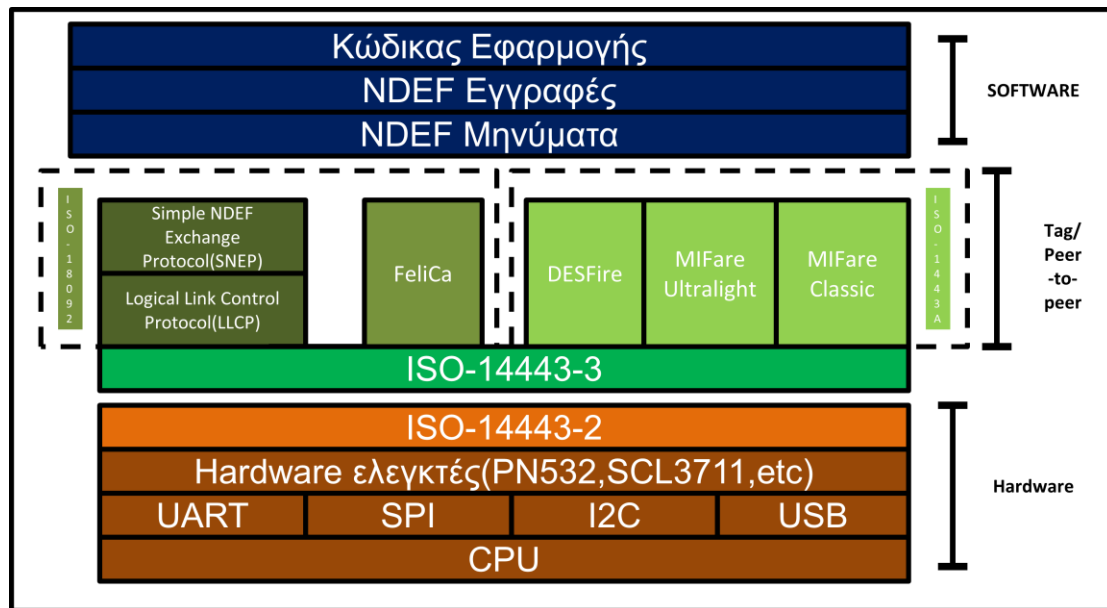
Στην συνέχεια, το chip ανταποκρίνεται με τα δεδομένα τα οποία έχει αποθηκευμένα και έπειτα αυτά στέλνονται μέσω της κεραίας της πίσω στην συσκευή ανάγνωσης. Η συσκευή ανάγνωσης με την σειρά της, εντοπίζει και ερμηνεύει αυτά τα δεδομένα και τα στέλνει σε έναν μικροελεγκτή ή υπολογιστή για περαιτέρω ανάλυση **Error!**

Reference source not found.. Όσον αφορά την ενεργή επικοινωνία(Peer-to-Peer),εδώ έχουμε 2 συσκευές οι οποίες είναι ενεργές, δηλαδή διαθέτουν μια πηγή ενέργειας(μπαταρία). Όπου κάθε συσκευή δημιουργεί εναλλάξ το δικό της ηλεκτρομαγνητικό πεδίο. Για να επιτευχθεί η επικοινωνία,

θα πρέπει κάθε φορά η μια συσκευή να είναι εκκινητής και η άλλη στόχος. Αν για παράδειγμα, η συσκευή_1 είναι εκκινητής και θέλει να στείλει δεδομένα στην συσκευή_2, τότε θα ενεργοποιήσει(συσκευή_1) το ηλεκτρομαγνητικό της πεδίο και η συσκευή_2 θα το έχει απενεργοποιημένο. Μόλις η συσκευή_1 πλησιάσει την συσκευή_2 θα της στείλει τα δεδομένα που θέλει. Όταν η συσκευή_2 θα χρειαστεί να απαντήσει, τότε θα ενεργοποιήσει το δικό της ηλεκτρομαγνητικό πεδίο και η συσκευή_1 αφού έχει απενεργοποιημένο το ηλεκτρομαγνητικό της πεδίο, τότε η συσκευή_2 θα της στείλει δεδομένα. Με λίγα λόγια έχουμε εναλλαγή των 2 συσκευών, από εκκινητή σε στόχο, κάθε φορά που θέλει να στείλει η μια στην άλλη δεδομένα. Ενώ στην παθητική επικοινωνία η συσκευή ανάγνωσης είναι ο εκκινητής και η ετικέτα ο στόχος. Επίσης, τα δεδομένα ανταλλάσσονται μέσω ενός ηλεκτρομαγνητικού πεδίου κάθε φορά, με την μια συσκευή να ενεργοποιεί και την άλλη να απενεργοποιεί το ηλεκτρομαγνητικό της πεδίο [4].

2.1.3 Αρχιτεκτονική του NFC και η σχέση με το RFID

Το NFC και το RFID συχνά συγχέονται, όμως δεν είναι το ίδιο πράγμα. Το NFC είναι μια επέκταση του RFID. Το NFC αποτελείται από 3 επίπεδα(Σχήμα 2.3), το χαμηλότερο επίπεδο είναι το φυσικό επίπεδο, το οποίο αποτελείται από την CPU και βασίζεται στο πρωτόκολλο ISO-14443-2 του RFID, το οποίο λειτουργεί στα 13.56 MHz. Μέσω αυτού του επιπέδου, επιτυγχάνεται η επικοινωνία με τις άλλες NFC συσκευές. Επίσης τα μέρη του hardware, επικοινωνούν με τον κύριο επεξεργαστή της συσκευής, χρησιμοποιώντας ένα από τα σειριακά πρωτοκολλά(UART,SPI,I2C).



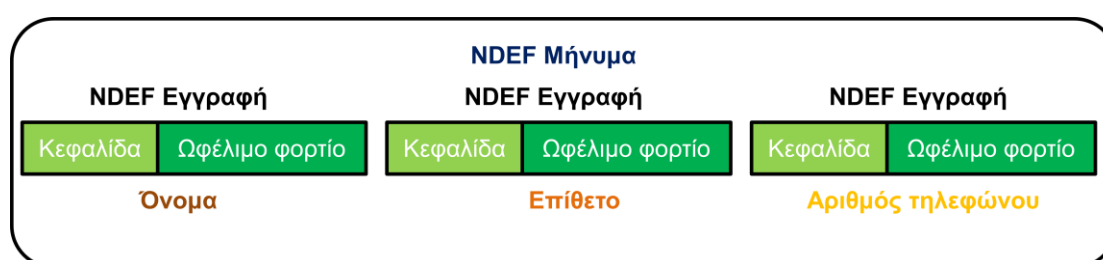
Σχήμα 2.3 Η δομή του NFC.

Στο μεσαίο επίπεδο, το κάτω μέρος αποτελείται από το πρωτόκολλο ISO-14443-3, όπου εκεί γίνεται η πακετοποίηση (τακτοποίηση) των δεδομένων. Πάνω από το ISO-14443-3, υπάρχουν πρωτόκολλα τα οποία βασίζονται στο RFID και είναι υπεύθυνα για την μεταφορά των δεδομένων. Στα δεξιά, υπάρχει το πρωτόκολλο του RFID το ISO-14443A, το οποίο είναι υπεύθυνο για την εγγραφή και το διάβασμα NFC ετικετών. Οι ετικέτες Mifare Classic, Mifare Ultralight και DESFire, είναι συμβατές με το ISO-14443A. Επίσης στα αριστερά, υπάρχει το πρωτόκολλο ISO-18092, βάση του οποίου μπορούμε να επιτύχουμε P2P (Peer-to-Peer) επικοινωνία, οι RFID κάρτες Sony FeliCa, βασίζονται σε αυτό το πρωτόκολλο. Τέλος, έχουμε το υψηλό επίπεδο, όπου εκεί έχουμε 2 NDEF επίπεδα, τα οποία είναι υπεύθυνα για την διαμόρφωση των δεδομένων και πάνω από αυτά βρίσκεται ο κώδικας της εφαρμογής που χρησιμοποιούμε. Η **1^η διαφορά** μεταξύ NFC και RFID, είναι ο τρόπος επικοινωνίας P2P. Στο NFC και συγκεκριμένα στο μεσαίο επίπεδο, υπάρχουν στο πρωτόκολλο ISO-18092, 2 μέρη, το ένα μέρος είναι το πρωτόκολλο LLCP (Logical Link Control Protocol), και το πρωτόκολλο ανταλλαγής NDEF, το SNEP (Simple NDEF Exchange Protocol), τα οποία διαχειρίζονται ανταλλαγές P2P. Η **2^η διαφορά**, είναι το NFC Data Exchange Format (NDEF), που βρίσκεται στην κορυφή αυτών των πρωτοκόλλων ελέγχου. Το επίπεδο NDEF, ορίζει μια σύντομη ανταλλαγή δεδομένων σε μηνύματα. Μέσα στα μηνύματα αυτά, υπάρχει το περιεχόμενο τους που ορίζεται ως NDEF εγγραφές (NDEF records). Το NDEF είναι αυτό που κάνει την εφαρμογή μας ικανή:

να ανταλλάξει δεδομένα με μια ετικέτα, διαβάζοντας ή γράφοντας σε αυτή, να έχουμε P2P επικοινωνία με μια άλλη συσκευή ή να λειτουργεί η συσκευή μας σαν κάρτα (Card Emulation). Επιπλέον το NDEF, μπορεί να χρησιμοποιηθεί για σύντομες ανταλλαγές δεδομένων από μια ετικέτα σε μια συσκευή αλλά και από μια συσκευή σε μια άλλη συσκευή. Αυτό μας δείχνει ότι το NFC, είναι κάτι περισσότερο από μια απλή τεχνολογία αναγνώρισης. Γενικά υπάρχουν τέσσερις τύποι ετικετών NFC, οι οποίοι είναι βασισμένοι στα RFID πρωτόκολλα, 3 από τους τύπους των ετικετών βασίζονται στο ISO-14443A και ο 4^ο τύπος βασίζεται στο ISO-18092. Αυτό καθιστά τις ετικέτες NFC τουλάχιστον εν μέρει συμβατές με πολλά RFID συστήματα(συσκευές ανάγνωσης), οι οποίες υποστηρίζουν και επικοινωνούν με Mifare και FeliCa ετικέτες. Παρόλο που αυτά τα RFID συστήματα δεν υποστηρίζουν NDEF, μπορούν ακόμα να υποστηρίζουν τις ετικέτες NFC που είναι συμβατές με αυτές, δηλαδή να τις διαβάζουν ή να γράφουν σε αυτές. Μια RFID συσκευή ανάγνωσης που μπορεί να διαβάσει ετικέτες Mifare Ultralight, θα μπορέσει να διαβάσει το UID, όπως για παράδειγμα, μιας ετικέτας NFC type 2, όμως δεν θα μπορέσει να διαβάσει δεδομένα NDEF, που είναι κωδικοποιημένα στην NFC ετικέτα [1].

2.1.4 NDEF

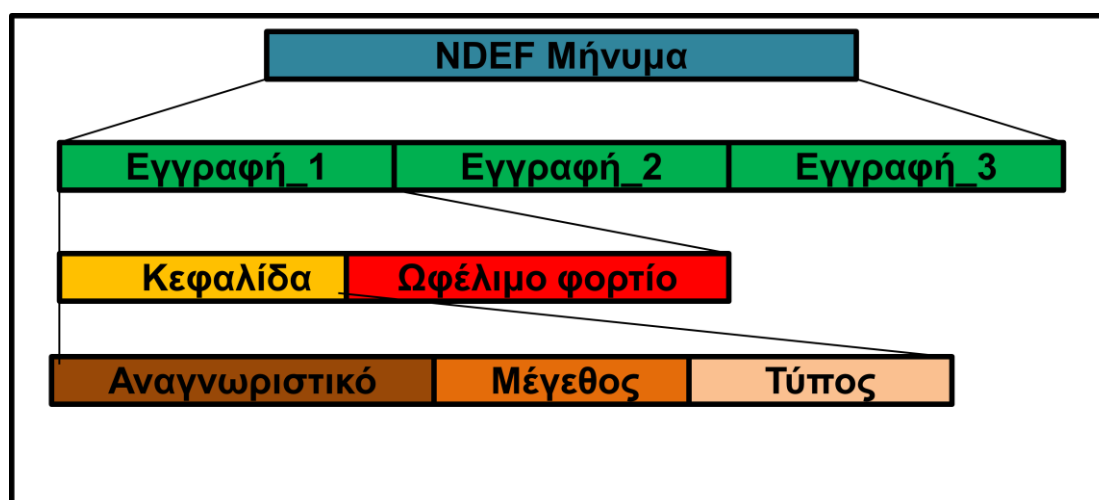
Είναι μια δυαδική μορφή δομημένη σε μηνύματα, κάθε NDEF μήνυμα αποτελείται από 1 ή περισσότερες εγγραφές(records)(**Σχήμα 2.4**).



Σχήμα 2.4 Η δομή ενός NDEF μηνύματος.

Κάθε ανταλλαγή δεδομένων που γίνεται, στέλνεται ή λαμβάνεται από μια συσκευή ένα μόνο NDEF μήνυμα κάθε φορά, επίσης σε μια ετικέτα μπορούμε να αποθηκεύσουμε ένα NDEF μήνυμα. Κάθε εγγραφή(record) αποτελείται από μια κεφαλίδα(header), η οποία περιέχει πληροφορίες για την συγκεκριμένη

εγγραφή(record), όπως τον τύπο της εγγραφής (record type) και το μέγεθος της (record length). Επιπλέον διαθέτει, το ωφέλιμο φορτίο(payload), στο οποίο υπάρχει το περιεχόμενο του NDEF μηνύματος. Άρα με λίγα λόγια θα λέγαμε, ότι ένα μήνυμα NDEF είναι μια μικρή παράγραφος που αφορά ένα συγκεκριμένο θέμα, όπου τα records είναι οι προτάσεις της παραγράφου, μέσω των οποίων λαμβάνουμε τις πληροφορίες για το συγκεκριμένο θέμα και μπορούμε να στείλουμε ένα τέτοιο μήνυμα από μια συσκευή σε μια άλλη ή από μια συσκευή σε μια ετικέτα, όταν αυτές οι συσκευές πλησιάσουν αρκετά κοντά. Όσον αφορά την κάθε **NDEF εγγραφή(record)**, που υπάρχει σε κάθε μήνυμα, αυτή αποτελείται από το ωφέλιμο φορτίο(payload) των δεδομένων, το οποίο μπορεί να είναι ένας από τους διαφορετικούς τύπους δεδομένων. Επιπλέον, διαθέτει κάποια τμήματα, τα οποία περιγράφουν πως ερμηνεύεται το payload. Επίσης, η κάθε εγγραφή περιέχει και έναν Header, ο οποίος περιγράφει την συγκεκριμένη εγγραφή και την θέση που έχει στο NDEF μήνυμα η εγγραφή αυτή. Επιπλέον, έχει τον τύπο της εγγραφής και το ID.

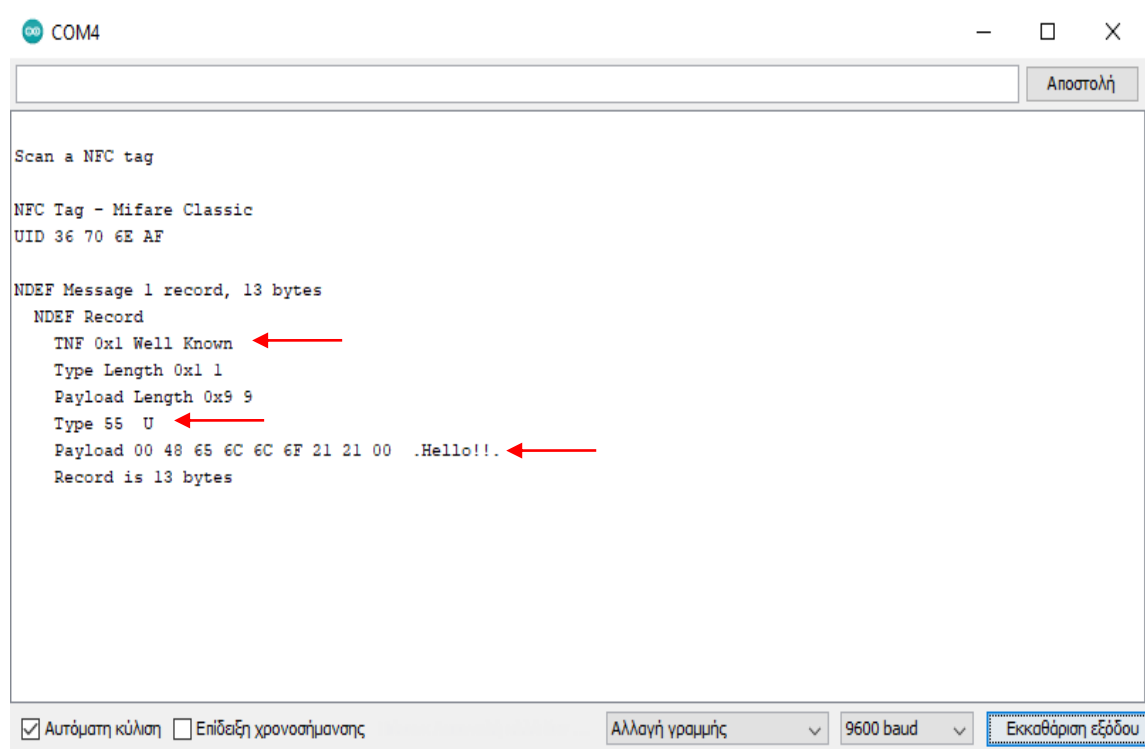


Σχήμα 2.5 Η δομή μιας NDEF εγγραφής.

Ποιο συγκεκριμένα και συνοψίζοντας την δομή ενός NDEF record(**Σχήμα 2.5**). Αποτελείται από το TNF, payload type και payload identifier, τα οποία βρίσκονται στον header, επιπλέον διαθέτει και το payload. Το TNF(Type Name Format), είναι αυτό το οποίο μας λέει πως ερμηνεύεται το payload type. Το payload type είναι ένας ειδικός τύπος NFC, για παράδειγμα MIME media-type ή URI που μας λέει πώς ερμηνεύεται το payload. Έχουμε 8 τύπους TNF: 0)Empty 1)Well-known 2)MIME media-type 3)Absolute URI 4)External 5)Unknown 6)Unchanged 7) Reserved. Τα

πιο συχνά είναι τα TNF 01(Well-known), TNF 02(MIME media-type) και τα TNF 04(External),τα οποία είναι για τα Android κινητά. Το RTD(Record Type Definition) περιγράφει τύπους Well-known και ορίζει τους κανόνες για την δημιουργία τύπων External. Με λίγα λόγια το TNF, ορίζει το τι τύπος θα είναι η εγγραφή, δηλαδή το τι είδους(κατηγορία) είναι αυτή η εγγραφή. Για παράδειγμα, εάν η εγγραφή είναι Well-known, η κατηγορία αυτή έχει 4 τύπους, άρα η εγγραφή που ανήκει σε αυτήν την κατηγορία, θα έχει έναν από τους 4 αυτούς τύπους. Επίσης, το payload type μας δείχνει το τι τύπος είναι το payload, το οποίο λέγεται και record type. Το payload identifier είναι προαιρετικό και επιτρέπει την συσχέτιση πολλών payloads μεταξύ τους. Υπάρχουν μερικοί Well-known τύποι NDEF εγγράφων. Τα Text records, είναι εγγραφές που μπορούμε να στείλουμε οποιαδήποτε συμβολοσειρά θέλουμε σαν μήνυμα. Επίσης υπάρχουν οι εγγραφές URI. Αυτές οι εγγραφές περιέχουν μια διεύθυνση από το δίκτυο. Ο 3^{ος} τύπος εγγραφών λέγεται Smart Posters, αυτού του τύπου την εγγραφή την χρησιμοποιούμε εάν θέλουμε στο μήνυμα που θα στείλουμε να συμπεριλαμβάνονται αρκετές πληροφορίες. Με λίγα λόγια μπορεί να περιέχει ένα σύνδεσμο URI, και όχι μόνο αυτό, ενώ μπορεί να έχει και ένα μήνυμα(text) με το θέμα του μηνύματος. Γενικά, μπορούμε με αυτού του τύπου την εγγραφή, μόλις διαβάσουμε μια ετικέτα να γίνουν πολλές ενέργειες, όπως να ανοίξει μια σελίδα στο διαδίκτυο, να λάβουμε ένα SMS ή κάποιο email, όλα αυτά μπορεί να γίνουν, όμως αυτά εξαρτώνται από το περιεχόμενο του συγκεκριμένου μηνύματος. Ο 4^{ος} τύπος είναι ο Signatures, μας παρέχει πληροφορίες για την προέλευση που έχουν τα δεδομένα τα οποία βρίσκονται εντός μιας εγγραφής NDEF. Για παράδειγμα μια εγγραφή(record) που έχει TNF 01 (Well-known),θα έχει ως τύπο εγγραφής(Record type) το γράμμα “T” σαν μήνυμα, εάν το Payload είναι ένα κείμενο(Text), ενώ το “U” στην περίπτωση URI(**Σχήμα 2.6**) και “SP” εάν το payload είναι smart poster. Όσον αφορά τους όρους URN,URI και URL. Το URI ή Uniform Resource Identifier, είναι μια συμβολοσειρά χαρακτήρων που προσδιορίζει μια διεύθυνση ιστού, δηλαδή θα δούμε σαν μήνυμα, μόλις για παράδειγμα διαβάσουμε μια ετικέτα, την διεύθυνση μιας ιστοσελίδας, την οποία πρέπει να επισκεφτούμε για να βρούμε κάποιες πληροφορίες για κάτι που μας ενδιαφέρει. Το URN ή Uniform Resource Name είναι το όνομα του URI, δηλαδή της σελίδας που πρέπει να επισκεφτούμε. Και το URL ή Uniform Resource Locator είναι ο σύνδεσμος, ο οποίος θα μας παραπέμψει στην ιστοσελίδα, όπου θα βρούμε τις πληροφορίες. Για παράδειγμα, ας υποθέσουμε ότι

θέλουμε να παραγγείλουμε παπούτσια από μια ιστοσελίδα. Το URN είναι το όνομα του μαγαζιού, για παράδειγμα ‘‘Shoes shop’’, το URI, είναι η διεύθυνση του μαγαζιού Shoes_shop.gr, ενώ το URL, θα είναι ο σύνδεσμος (http://www.Shoes_shop.gr) που ακριβώς θα μας οδηγήσει στην ιστοσελίδα για να αγοράσουμε τα παπούτσια.



Σχήμα 2.6 NDEF μήνυμα κατηγορίας Well known τύπου URI, όπου το περιεχόμενο του μηνύματος(payload) είναι το Hello!!.

Τα NDEF μηνύματα μπορούν να περιέχουν πολλά payload types, παρακάτω έχουμε ένα NDEF μήνυμα, με 3 records, όπου τα 2 είναι τύπου Text και το ένα τύπου URI. (Σχήμα 2.7).

```
COM4
NFC Tag - Mifare Classic
UID 89 7D A5 03

NDEF Message 3 records, 68 bytes
NDEF Record
TNF 0x1 Well Known
Type Length 0x1 1
Payload Length 0x12 18
Type 54 T
Payload 02 65 6E 48 65 6C 6C 6F 2C 20 41 72 64 75 69 6E 6F 21 .enHello, Arduino!
Record is 22 bytes
NDEF Record
TNF 0x1 Well Known
Type Length 0x1 1
Payload Length 0x12 18
Type 55 U
Payload 00 68 74 74 70 3A 2F 2F 61 72 64 75 69 6E 6F 2E 63 63 .http://arduino.cc
Record is 22 bytes
NDEF Record
TNF 0x1 Well Known
Type Length 0x1 1
Payload Length 0x14 20
Type 54 T
Payload 02 65 6E 47 6F 6F 64 62 79 65 2C 20 41 72 64 75 69 6E 6F 21 .enGoodbye, Arduino!
Record is 24 bytes
```

Σχήμα 2.7 NDEF μήνυμα με 3 records.

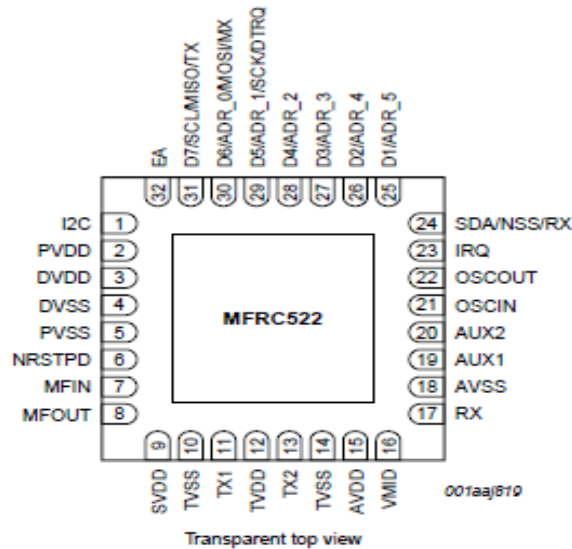
Τέλος, το **payload** είναι το περιεχόμενο του μηνύματος. Που θα στείλει(Σχήμα 2.6), για παράδειγμα μια συσκευή σε μια ετικέτα και το περιεχόμενο αυτό μπορεί να είναι κρυπτογραφημένο, μπορεί να είναι ένα απλό κείμενο(Text), μπορεί να είναι απλά μια λέξη η οτιδήποτε άλλο θέλουμε. Από την εφαρμογή, μέσω της οποίας στέλνουμε NDEF μηνύματα αλλά και μέσω της εφαρμογής που λαμβάνει τα NDEF μηνύματα, εξαρτάται το τι θα είναι και πως θα διαμορφώνεται το payload [1].

2.2 ΛΕΙΤΟΥΡΓΙΑ ΤΩΝ 2 CHIP

2.2.1 Η συσκευή ανάγνωσης RC522 και το chip του MFRC522

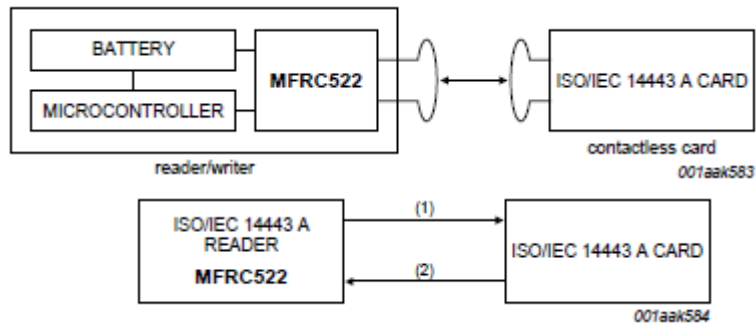
Το MFRC522(Σχήμα 2.8) είναι ένα chip το οποίο είναι ενσωματωμένο στην RFID συσκευή ανάγνωσης RC522 και είναι της εταιρίας NXP. Τα γενικά χαρακτηριστικά που διαθέτει είναι ότι λειτουργεί στα 13.56 MHz και έτσι μπορεί να διαβάσει ή να γράφει δεδομένα ανέπαφα σε κάρτες ή ετικέτες αρκεί να είναι MIFARE / ISO/IEC 14443A ή NTAG. Υποστηρίζει όλες τις κάρτες MIFARE.Μερικές από τις κάρτες που υποστηρίζει είναι: MIFARE Mini, MIFARE 4K, MIFARE Ultralight, MIFARE DESFire EV1 and MIFARE Plus. Η απόσταση που μπορεί να βρίσκεται η κάρτα, ώστε να μπορέσει να επικοινωνήσει με την συσκευή ανάγνωσης, είναι μέχρι τα 5 εκατοστά, αλλά αυτή η απόσταση εξαρτάται από το μέγεθος της κεραίας της συσκευή ανάγνωσης και τον συντονισμό της. Υποστηρίζει 3 πρωτόκολλα επικοινωνίας, το SPI, I2C και UART, μέσω των οποίων μπορεί να επικοινωνεί με διάφορους μικροελεγκτές και να τους στέλνει τα δεδομένα τα οποία έχει λάβει από μια κάρτα. Επίσης, μπορεί να κάνει Hard reset με χαμηλή λειτουργία ισχύος, έχει την δυνατότητα για power down από το software mode, έχει προγραμματιζόμενο timer, διαθέτει εσωτερικό ταλαντωτή για σύνδεση με 27.12 MHz κρύσταλλο, CRC συνεπεξεργάστη, προγραμματιζόμενα I/O pins, εσωτερικό self-test και οι τιμές των τάσεων που μπορεί να λειτουργεί είναι από 2.5V μέχρι 3.3V. Ένα ακόμα ενδιαφέρον χαρακτηριστικό είναι ότι είναι αρκετά ανθεκτικός στις επιδράσεις της θερμοκρασίας, όπου οι αντοχές του κυμαίνονται από -20 έως 80 βαθμούς κελσίου.

Όσον αφορά την λειτουργία του, έχει 1 πομπό, ο οποίος χειρίζεται την κεραία, ώστε να μπορούμε να διαβάσουμε και να γράφουμε δεδομένα σε MIFARE κάρτες ή αναμεταδότες χωρίς επιπλέον ενεργό κύκλωμα. Επίσης έχει 1 δέκτη, ο οποίος αποδιαμορφώνει και κωδικοποιεί τα σήματα που λαμβάνονται από τις κάρτες και τους αναμεταδότες.



Σχήμα 2.8 Αντιστοίχιση ακροδεκτών του ολοκληρωμένου κυκλώματος MFRC522. Πηγή: <https://www.nxp.com/docs/en/datasheet/MFRC522.pdf> [5].

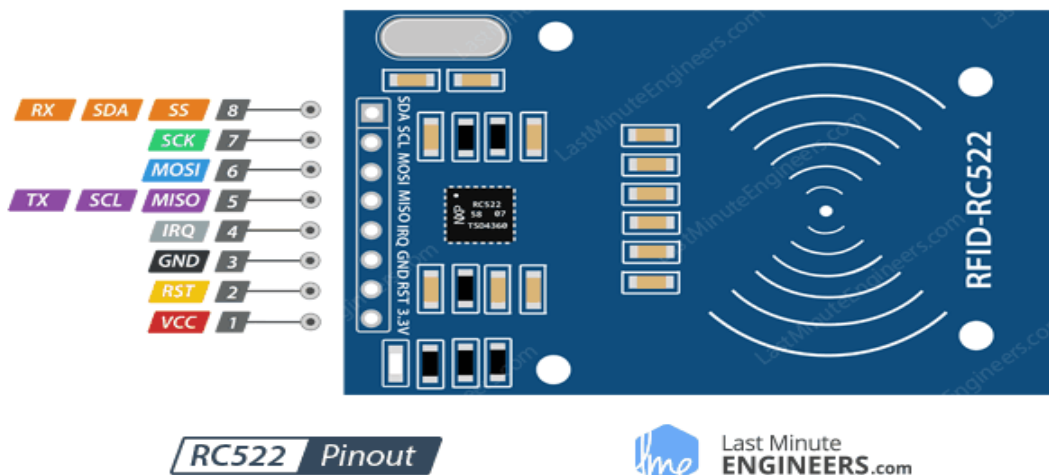
Η συσκευή ανάγνωσης στέλνει ένα σήμα, το οποίο είναι σε ψηφιακή μορφή με διαμόρφωση ASK. Το σήμα αυτό είναι ψηφιακό με χρήση ενός ημιτονοειδούς φέροντος. Επίσης, το σήμα αυτό μπορεί να έχει 4 ταχυτητες, 106, 212, 424, 848 kBd κωδικοποιημένο κατά Miller. Με ποια ταχύτητα θα σταλθεί εξαρτάται από το μέγεθος των bits, όσο μεγαλύτερο είναι το μέγεθος τόσο πιο αργά θα σταλθεί το σήμα. Από την άλλη πλευρά, η κάρτα απαντάει με ένα επίσης ψηφιακό σήμα, το οποίο αν στέλνεται με ταχύτητα 106 kBd, τότε είναι κωδικοποιημένο κατά Manchester, ενώ στις άλλες ταχύτητες στέλνεται ένα ψηφιακό σήμα με κωδικοποίηση BPSK (Σχήμα 2.9).



Σχήμα 2.9 Η επικοινωνία μεταξύ του RC522 και μιας κάρτας.

Πηγή: <https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf> [5].

Στην παρούσα πτυχιακή, έγινε χρήση της RFID συσκευής ανάγνωσης RC522(Σχήμα 2.10), και χρησιμοποιήθηκε το πρωτόκολλο επικοινωνίας SPI,για το οποίο θα γίνει αναφορά παρακάτω.



Σχήμα 2.10 Η συσκευή ανάγνωσης RC522

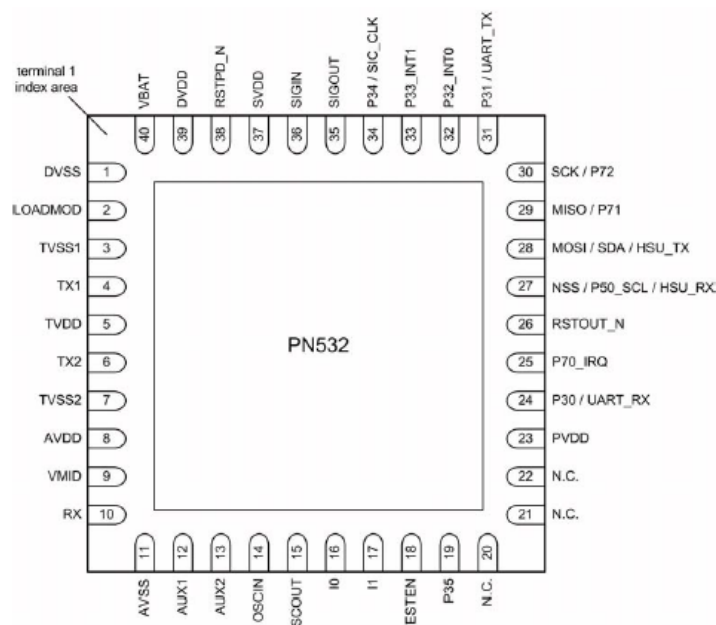
Πηγή:<https://lastminuteengineers.com/how-rfid-works-rc522-arduino-tutorial/> [3].

Το SPI είναι μια διεπαφή που υποστηρίζει την υψηλή ταχύτητα επικοινωνίας. Όταν έχουμε SPI, ανάμεσα στον RC522 και σε ένα μικροελεγκτή(Arduino), έχουμε σειριακή επικοινωνία σε υψηλή ταχύτητα μέχρι 10Mbit/s. Ο RC522 λειτουργεί σαν δέκτης(slave), κατά την διάρκεια της SPI επικοινωνίας και λαμβάνει δεδομένα για να ρυθμίσει τις λειτουργίες των καταχωρητών του. Κάθε καταχωρητής είναι του ενός

Byte, εκτός από τον CRC ο οποίος είναι των 2 Byte. Κάθε καταχωρητής έχει 8 bits και το κάθε bit από αυτά έχει τον σκοπό του και την ανάλογη συμπεριφορά. Τα bits τα οποία έχουν την ίδια συμπεριφορά, δηλαδή κάνουν τις ίδιες ενέργειες, τοποθετούνται σε κοινούς καταχωρητές. Έχουμε 6 κατηγορίες bits βάση της συμπεριφοράς τους: 1)R/W bits, αυτά τα bits μπορούν να γραφούν και να διαβαστούν από έναν μικροελεγκτή 2) Dynamic bits 3)Read only 4) Write only 5)reserved 6)RFT. Ο RC522 έχει 31 καταχωρητές και 13 test registers, σύνολο 44 καταχωρητές. Ο RC522 διαθέτει 8 Pins. Το **VCC** για να τροφοδοτείται η συσκευή ανάγνωσης στα 3.3V. Το **GND** για την γείωση, επίσης μέσω της γραμμής **MOSI**, στέλνονται τα δεδομένα από τον μικροελεγκτή, για παράδειγμα, από τον Arduino(master) στον RC522(slave). Μέσω της γραμμής **MISO**, στέλνονται δεδομένα από τον RC522 στον Arduino. Το pin **SCK** είναι αυτό που δέχεται τους παλμούς ρολογιού που παρέχονται από τον Arduino(master) **Error! Reference source not found.** Το pin **SDA** λειτουργεί ως είσοδος σήματος. Το **RST** είναι μια είσοδος για επαναφορά (reset) και απενεργοποίηση (power down). Όταν είναι σε κατάσταση LOW, τότε η συσκευή ανάγνωσης μπαίνει σε λειτουργία απενεργοποίησης. Το **IRQ**, είναι μια ακίδα διακοπής που ειδοποιεί τον μικροελεγκτή όταν μια ετικέτα βρίσκεται κοντά **Error! Reference source not found.**

2.2.2 PN532

Το PN532 είναι ένα NFC chip, της εταιρίας NXP(Σχήμα 2.11) και είναι ενσωματωμένο σε διάφορες NFC συσκευές ανάγνωσης, μερικές από τις οποίες θα δούμε παρακάτω σε αυτό το κεφάλαιο.



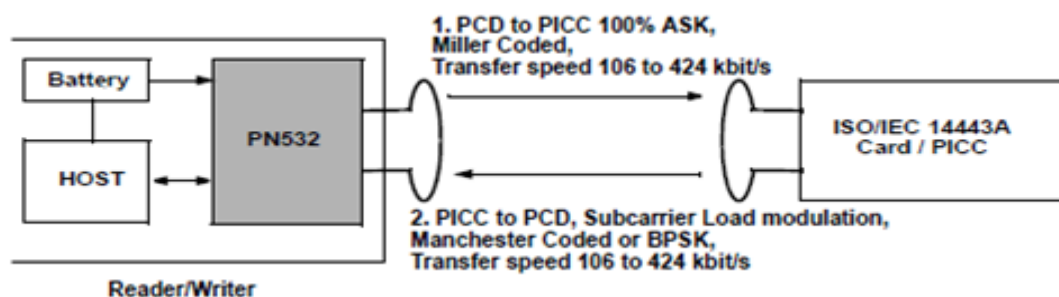
Σχήμα 2.11 Αντιστοίχιση ακροδεκτών του ολοκληρωμένου κυκλώματος PN532.

Πηγή: https://www.nxp.com/docs/en/nxp/datasheets/PN532_C1.pdf [6].

Ο PN532 είναι μια μονάδα πομποδέκτη η οποία βασίζεται στον πυρήνα του μικροελεγκτή 80C51, με την οποία μπορούμε να έχουμε ασύρματη επικοινωνία στα 13.56 MHz. Οι λειτουργίες που διαθέτει είναι 6. Η 1^η λειτουργία, από αυτές είναι η ISO/IEC 14443A/MIFARE Reader/Writer, με την οποία μας δίνει την δυνατότητα να μπορούμε να διαβάζουμε και να γράφουμε δεδομένα σε αυτές τις κάρτες. Διαθέτει αποδιαμορφωτή και αποκωδικοποιητή για σήματα από τις συγκεκριμένες κάρτες και τους αναμεταδότες. Επίσης, χειρίζεται την πλήρη πλαισίωση και τον εντοπισμό σφαλμάτων αυτών των καρτών. Η 2^η λειτουργία, είναι η FeliCa Reader/Writer, με την οποία μπορούμε να κάνουμε ακριβώς τις ίδιες ενέργειες με την προηγούμενη λειτουργία, αυτή την φορά όμως σε κάρτες FeliCa. Το PN532 μπορεί να αποδιαμορφώσει και να αποκωδικοποιήσει κωδικοποιημένα σήματα από αυτές τις κάρτες. Επιπλέον, χειρίζεται τον εντοπισμό σφαλμάτων τους και υποστηρίζει

ανέπαφη επικοινωνία χρησιμοποιώντας υψηλές ταχύτητες μεταφοράς έως και 424 kbps και προς τις δύο κατευθύνσεις. Σε αυτές τις 2 λειτουργίες, μπορούμε να έχουμε επικοινωνία με αυτές τις κάρτες μέχρι τα 5 εκατοστά, εξαρτάται όμως τι μέγεθος έχει η κεραία αλλά και ο συντονισμός της. Η 3^η λειτουργία, είναι η ISO/IEC 14443B Συσκευή Reader/Writer. Η 4^η λειτουργία, είναι η ISO/IEC 14443A/MIFARE Card MIFARE Classic 1K και MIFARE Classic 4K card emulation mode, αυτή είναι μια λειτουργία εξομοίωσης κάρτας. Η 5^η λειτουργία, είναι η FeliCa Card Emulation. Στην λειτουργία εξομοίωσης, ο PN532 μπορεί να λαμβάνει δεδομένα μέχρι και τα 10 εκατοστά, ανάλογα βέβαια και με το μέγεθος της κεραίας. Το PN532 σε αυτή την λειτουργία έχει την δυνατότητα να δουλέψει σαν μια κάρτα MIFARE ή FeliCa και να στείλει μια απάντηση, όπως θα το έκανε μια από αυτές τις κάρτες. Η 6^η λειτουργία είναι ISO/IEC 18092 και ECMA 340 NFCIP-1(peer to peer), σε αυτήν την λειτουργία μπορούμε να έχουμε παθητική και ενεργή επικοινωνία μεταξύ μια συσκευής που είναι ο εκκινητής και μιας ακόμη συσκευής που είναι ο στόχος, οι 2 αυτές συσκευές είναι 2 συσκευές ανάγνωσης(ίδιες), που έχουν το chip PN532. Οι ταχύτητες μεταφοράς των δεδομένων σε αυτήν την λειτουργία είναι έως και 424 kbps και η απόσταση λειτουργίας είναι τα 5 cm ανάλογα την κεραία. Όσο αφορά τις παραπάνω λειτουργίες, όποια και εάν χρησιμοποιήσουμε, οι ταχύτητες που ανταλλάσσονται τα δεδομένα είναι 106,212 και 424 kbps. Επίσης ο PN532 υποστηρίζει 3 πρωτόκολλα επικοινωνίας, το SPI, I2C και UART(HSU), διαθέτει 2 λειτουργίες χαμηλής ισχύος, την Hard-Power-Down (1μΑ) και Soft-Power-Down(22 μΑ). Το εύρος λειτουργίας του, κυμαίνεται από 2.7 έως 5.5V και έχει υψηλές αντοχές στις επιδράσεις της θερμοκρασίας, οι όποιες κυμαίνονται από -30 έως 85 βαθμούς κελσίου. Επιπλέον, αποτελείται από ειδικές θύρες IO για έλεγχο εξωτερικής συσκευής, ταλαντωτή κρυστάλλου και προγραμματιζόμενους χρονοδιακόπτες. Στην 1^η λειτουργία(Σχήμα 2.12), έχουμε μια NFC συσκευή ανάγνωσης, η οποία περιμένει μια κάρτα να εισέλθει εντός του ηλεκτρομαγνητικού πεδίου που δημιουργεί, ώστε να της στείλει ή να λάβει από αυτή κάποια δεδομένα. Αρχικά, ο συσκευή ανάγνωσης μόλις η κάρτα είναι εντός του πεδίου στέλνει ένα σήμα το οποίο είναι ψηφιακό με διαμόρφωση ASK. Το σήμα αυτό μπορεί να έχει 3 ταχύτητες 106 kbps, 212 kbps και 424 kbps, επίσης είναι κωδικοποιημένο κατά Miller. Αφού σταλθεί το σήμα στην κάρτα, αυτή θα απαντήσει με ένα επίσης ψηφιακό σήμα, το οποίο εάν έχει ταχύτητα 106 kbps είναι κωδικοποιημένο κατά Manchester και στις περιπτώσεις των άλλων 2 ταχυτήτων

έχουμε κωδικοποίηση BPSK. Όσο πιο μεγάλο είναι το μέγεθος των bits, τόσο πιο αργά στέλνονται τα δεδομένα. Όσον αφορά την λειτουργία κάρτας(card emulation),λειτουργεί ακριβώς όπως και 1^η λειτουργία.



Σχήμα 2.12 Η επικοινωνία μεταξύ μιας συσκευής ανάγνωσης με το chip PN532 και μιας κάρτας. Πηγή: https://www.nxp.com/docs/en/nxp/datasheets/PN532_C1.pdf [6].

Στο κεφάλαιο 4, όπου βρίσκεται το NFC project, εκεί έγινε χρήση του πρωτόκολλου επικοινωνίας I2C, για αυτό τον λόγο θα γίνει αναφορά σε αυτό. Τα κύρια χαρακτηριστικά αυτού του πρωτοκόλλου είναι η υποστήριξη Master/slave I2C. Επίσης διαθέτει 2 λειτουργίες, την κανονική (100kHz SCLK) και την γρήγορη(400 kHz SCLK), ενώ έχει λειτουργία αφύπνισης του PN532. Το I2C μπορεί να λειτουργήσει στις 4 αυτές λειτουργίες: 1)Master πομπός 2)Master δέκτης 3)slave δέκτης 4)slave πομπός. Τα δεδομένα μεταφέρονται από τον πομπό του Master στον δέκτη του slave. Ως συμπέρασμα μπορούμε να πούμε ότι, το I2C μπορεί να λειτουργεί ως Master ή ως slave, όμως στο PN532 το I2C ρυθμίζεται ως slave, επειδή ο μικροελεγκτής, όπως ο Arduino είναι ο master. Επιπλέον, το block αφύπνισης του PN532 μπορεί να χρησιμοποιηθεί μόνο όταν το I2C είναι σε slave mode [6].

2.3 Διαφορές των 2 chip

Ο MFRC522 και ο PN532 έχουν αρκετά κοινά στοιχεία μεταξύ τους, όπου ένα από αυτά, είναι ότι και τα 2 κατασκευάζονται από την ίδια εταιρία. Ένα άλλο, είναι ότι και τα 2 δουλεύουν στην ίδια συχνότητα, στα 13.56 MHz. Λειτουργούν ακριβώς με τον ίδιο τρόπο, όπως για παράδειγμα η λειτουργία για το διάβασμα των καρτών MIFARE/ISO IEC 14443A,την διαθέτουν και οι 2, και λειτουργεί αμφοτέρως με τον ίδιο τρόπο. Η μονή διαφορά που έχουν σε αυτή την λειτουργία είναι ότι ο MFRC522

έχει την δυνατότητα να στέλνει δεδομένα με ταχύτητα μέχρι 848 kbps, ενώ ο PN532 μέχρι 424 kbps, άρα ο MFRC522 έχει την δυνατότητα να στείλει πιο γρήγορα δεδομένα από τον PN532. Από εκεί και πέρα, ο PN532 είναι πιο εμπλουτισμένος όσον αφορά τις λειτουργίες του, διότι διαθέτει 6 λειτουργίες έναντι μιας που έχει ο MFRC522, η οποία είναι το διάβασμα και η εγγραφή καρτών MIFARE και NTAG. Πιο συγκεκριμένα, η συσκευή ανάγνωσης, η οποία δουλεύει με το chip MFRC522, είναι ο RC522. Το hardware του RC522 δεν υποστηρίζει: 1) Peer-to-peer (ISO/IEC 18092) επικοινωνία 2)Επικοινωνία με έξυπνα κινητά τηλέφωνα 3)Εξομοίωση κάρτας(card emulation). Επίσης, το λογισμικό του RC522 δεν υποστηρίζει κάποιες κάρτες MIFARE, τις MIFARE DESFire, MIFARE DESFire EV1/EV2, δεν διαθέτει λειτουργία μειωμένης ισχύς και από τον κώδικα που μέχρι στιγμής υπάρχει στο διαδίκτυο, δεν υποστηρίζει τα πρωτόκολλα I2C και UART. Αυτό έχει ως αποτέλεσμα να είναι μονόδρομος η χρήση της SPI επικοινωνίας μεταξύ του συσκευής ανάγνωσης και του μικροελεγκτή [7]. Από την άλλη πλευρά, ο PN532 και γενικά οι συσκευές ανάγνωσης που διαθέτουν αυτό το chip υποστηρίζουν και τα 3 πρωτόκολλα επικοινωνίας με επιτυχία. Επιπλέον, ο PN532 έχει λίγη μεγαλύτερη αντοχή στις επιδράσεις της θερμοκρασίας και μπορούμε να το τροφοδοτήσουμε με μέγιστη τάση τα 5.5 V έναντι των 3.3V που τροφοδοτούμε τον RC522. Συμπερασματικά, μπορούμε να πούμε ότι το chip MFRC522 το οποίο υπάρχει στον RFID συσκευή ανάγνωσης RC522 υστερεί αρκετά σε σχέση με τον PN532. Όμως, ανάλογα με την εφαρμογή την οποία έχουμε, θα πρέπει να επιλέξουμε και τον κατάλληλο συσκευή ανάγνωσης που έχει ενσωματωμένο πάνω του ένα από αυτά τα 2 chip.

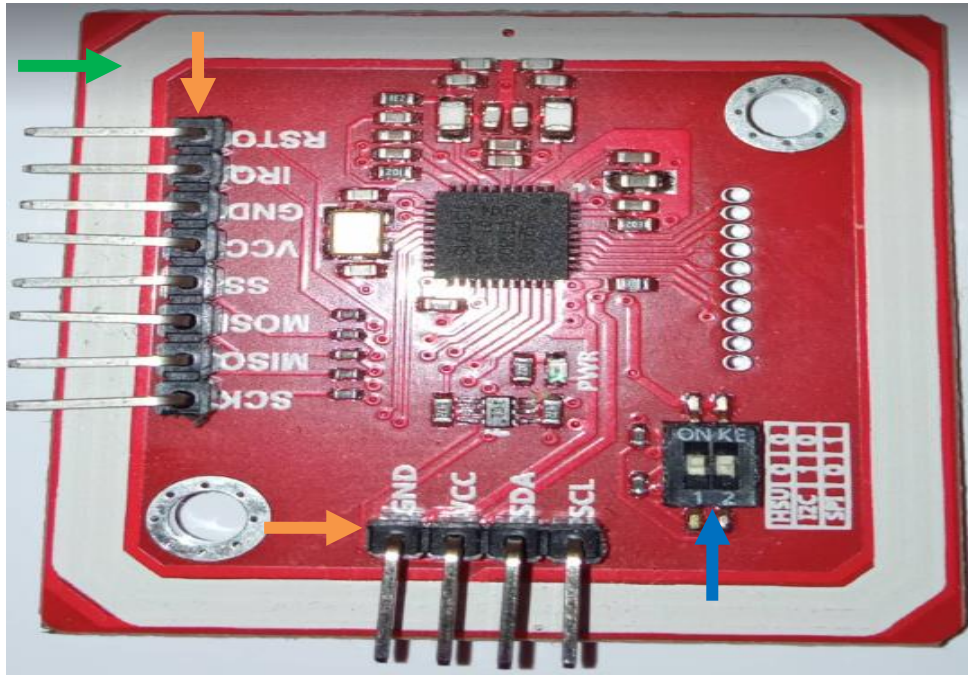
2.4 NFC συσκευές ανάγνωσης

Το PN532 είναι ένα chip το οποίο είναι ενσωματωμένο σε πολλές συσκευές ανάγνωσης, τις οποίες μπορούμε να χρησιμοποιήσουμε σε διαφορές εφαρμογές που αφορούν το NFC, το RFID, να διαβάσουμε NDEF μηνύματα από διαφορές κάρτες όπως MIFARE και γενικά να δημιουργήσουμε εφαρμογές πάνω στο κομμάτι της ασύρματης επικοινωνίας. Κάθε συσκευή ανάγνωσης που θα αναφερθεί παρακάτω σε αυτήν την ενότητα έχει τις δίκες του δυνατότητες και αυτό εξαρτάται από το Hardware και το Software που διαθέτει η κάθε συσκευή ανάγνωσης.

2.4.1 PN532 V3

Η πρώτη συσκευή ανάγνωσης στην οποία θα αναφερθούμε είναι ο PN532 V3. Είναι μια μονάδα που υποστηρίζει το NFC αλλά και το RFID. Τα γενικά χαρακτηριστικά του είναι: **1)**Υποστηρίζει, 3 πρωτοκόλλα επικοινωνίας: το SPI,I2C και UART. Όταν χρησιμοποιήσουμε την SPI επικοινωνία πρέπει να τροφοδοτήσουμε την συσκευή ανάγνωσης με 3.3V, ενώ στα άλλα 2 πρωτόκολλα με 5V. **2)** Μπορεί να λειτουργήσει, ως μια RFID συσκευή ανάγνωσης και να διαβάζει και να γράφει δεδομένα σε κάρτες MIFARE 1k ή 4k κάρτες Ultralight και DesFire. Επίσης, υποστηρίζει κάρτες που είναι βασισμένες στο πρωτόκολλο ISO/IEC 14443-4, Innovision Jewel και FeliCa κάρτες,**3)** Διαθέτει μια εσωτερική κεραία PCB, μέσω της οποίας μπορεί να επικοινωνεί με κάρτες στα 5 με 7 cm. **4)** Έχει την δυνατότητα να δουλεύει σαν μια κάρτα 14443-A ή ως μια εικονική κάρτα, δηλαδή έχει την δυνατότητα προσομοίωσης μιας κάρτα. **5)** Επιπλέον, είναι συμβατό με το Arduino και πλακέτες όπως το UNO,το Mega,το Leonardo και το Due. **6)**Υποστηρίζει την δυνατότητα επικοινωνίας με το NFC των κινητών τηλεφώνων. Εξωτερικά ο PN532(**Σχήμα 2.13**), αποτελείται από την κεραία(**πράσινο βελάκι**), η οποία και δημιουργεί το ηλεκτρομαγνητικό πεδίο για να μπορέσει να επικοινωνήσει η συσκευή ανάγνωσης με οποιανδήποτε συσκευή ή κάρτα. Επίσης, έχει τις ακίδες(**πορτοκαλί βελάκι**) τις οποίες ανάλογα με ποιο πρωτόκολλο επικοινωνίας χρησιμοποιούμε επιλέγουμε τις κατάλληλες.

Για το I2C, επιλέγουμε τις 4 ακίδες(VCC,GND,SDA,SCL), ενώ για τα άλλα 2 πρωτόκολλα επιλέγουμε τις άλλες ακίδες(**Σχήμα 2.13** αριστερά). Επίσης, έχει ένα διακόπτη(**μπλε βελάκι**) 2 θέσεων, το οποίο το ρυθμίζουμε και επιλεγούμε μέσω αυτού, το πρωτόκολλο επικοινωνίας που θέλουμε. Εάν θέλουμε UART, το αφήνουμε όπως είναι, δηλαδή και οι 2 θέσεις του διακόπτη να είναι στην θέση 0, δηλαδή προς τα κάτω. Εάν θέλουμε I2C θέτουμε την θέση 1 προς τα πάνω, ενώ για SPI την θέση 2 προς τα πάνω [8].

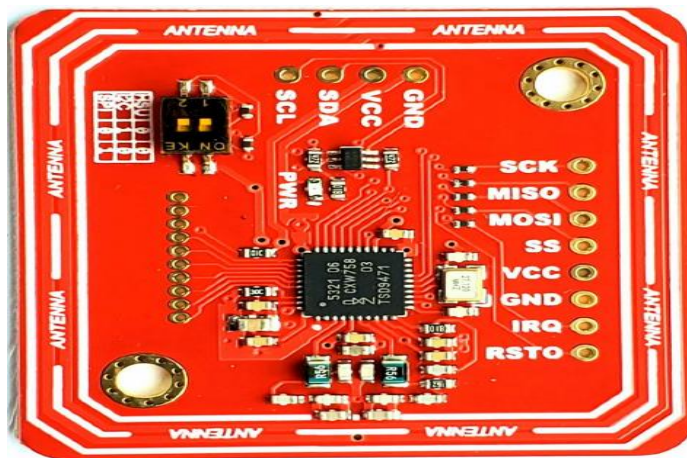


Σχήμα 2.13 Φωτογραφία της πλακέτας PN532 V3.

Όσον αφορά τις λειτουργίες του, ανάλογα με το ποιά θέλουμε να χρησιμοποιήσουμε θα πρέπει να κατεβάσουμε από το διαδίκτυο την κατάλληλη βιβλιοθήκη. Για παράδειγμα, για διάβασμα καρτών MIFARE θα πρέπει να κατεβάσουμε την βιβλιοθήκη με όνομα **PN532** [9]. Εάν θέλουμε να έχουμε μια P2P επικοινωνία ανάμεσα σε 2 συσκευές ανάγνωσης PN532 V3, θα πρέπει να κατεβάσουμε άλλη βιβλιοθήκη σύμφωνα με το manual. Για αυτή την λειτουργία, η μια συσκευή ανάγνωσης θα παίζει το ρόλο του εκκινητή και ο άλλος του στόχου. Επιπλέον θα χρειαστούμε 2 πλακέτες Arduino και θα πρέπει να είμαστε ιδιαίτερα προσεκτικοί ώστε να μην βάλουμε στον ίδιο φάκελο την βιβλιοθήκη, με την βιβλιοθήκη του **PN532** ώστε να μην έχουμε προβλήματα στο πρόγραμμα μας. Στην περίπτωση που θέλουμε να επικοινωνήσουμε με ένα κινητό τηλέφωνο, τότε θα πρέπει να κατεβάσουμε την βιβλιοθήκη `NFC_Module_DEV`, η βιβλιοθήκη αυτή λειτουργεί μόνο σε UART επικοινωνία. Εάν θέλουμε να δουλέψουμε με SPI θα πρέπει να εγκαταστήσουμε την `NFC_Shield_DEV`, όμως η βιβλιοθήκη αυτή έχει γραφτεί για την συσκευή ανάγνωσης Seed studio. Επομένως για τον PN532 V3 πρέπει να χρησιμοποιηθεί η βιβλιοθήκη `NFC_Module_DEV`. Ωστόσο, αν θέλουμε να χρησιμοποιήσουμε αυτήν την λειτουργία, θα πρέπει να ξέρουμε ότι δεν λειτουργεί με το Arduino UNO, αυτή την δυνατότητα μας την προσφέρουν το Arduino Mega ή Leonardo. Ο λόγος είναι, επειδή η βιβλιοθήκη που πρέπει να χρησιμοποιήσουμε

δουλεύει μόνο με την επικοινωνία UART, η οποία βιβλιοθήκη χρειάζεται τουλάχιστον μια σειριακή θύρα. Ο Arduino UNO έχει μόνο μια και την χρησιμοποιεί για να επικοινωνεί με τον H/Y, συνεπώς δεν μας παρέχει καμία ελεύθερη σειριακή θύρα, ενώ ο Arduino Leonardo μας παρέχει μια ελεύθερη θύρα και ο Arduino Mega, ο οποίος έχει 4 σειριακές θύρες, την μια την δεσμεύει για την επικοινωνία με τον H/Y και μας παρέχει 3 ελεύθερες για να τις χρησιμοποιήσουμε [8].

2.4.2 PN532 V4



Σχήμα 2.14 Φωτογραφία της πλακέτας PN532 V4.

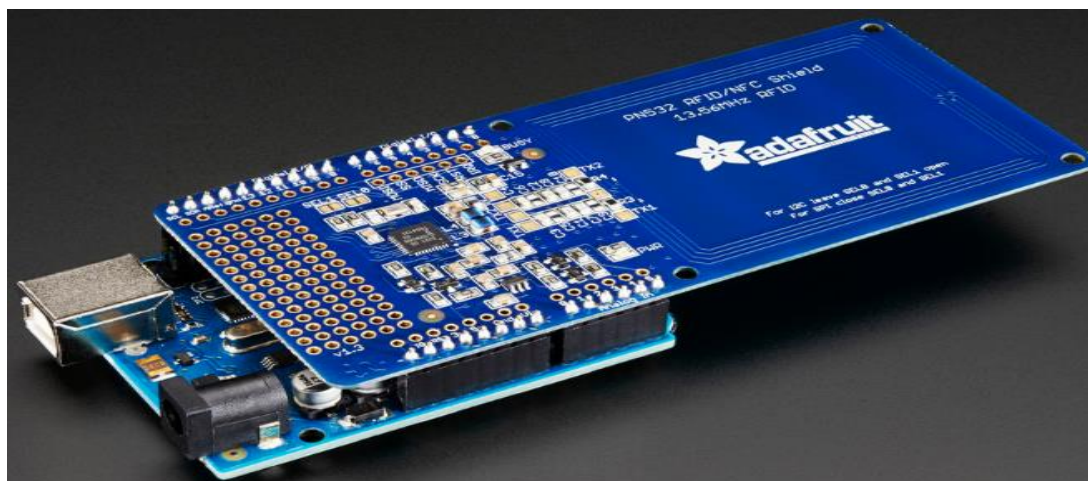
Πηγή: <https://www.elechouse.com/product/pn532-nfc-rfid-module-v4/> [10].

Η έκδοση V4(Σχήμα 2.14), έχει τα ίδια ακριβώς χαρακτηριστικά με την προηγούμενη έκδοση, όπως επίσης και οι λειτουργίες είναι οι ίδιες. Η μόνη διαφορά που υπάρχει με την προηγούμενη έκδοση, είναι στο ότι το ηλεκτρομαγνητικό πεδίο που δημιουργεί έχει μεγαλύτερο εύρος, μπορεί να διαβάσει κάρτες MIFARE/ISO 14443-A σε απόσταση από 6 έως 8 cm σε αντίθεση με την V3, η οποία διαβάσει στα 5 με 6 cm. Επιπλέον, έχει βελτίωση στο διάβασμα καρτών ISO14443-B, τις οποίες διαβάσει από τα 2 έως τα 4 cm ενώ η V3 από τα 1 έως 2cm. Άλλη μια διαφορά με την προηγούμενη έκδοση, είναι η τιμή αγοράς, όπου η V4 κάνει περίπου 35\$ ενώ η έκδοση V3 περίπου 20\$ [10].

2.4.3 Adafruit PN532 RFID/NFC Breakout και Shield

Η εταιρία Adafruit διαθέτει 2 πλακέτες PN532 RFID/NFC, η μια είναι το Breakout και το άλλο λέγεται Shield. Οι 2 αυτές πλακέτες, δουλεύουν στην συχνότητα των 13.56 MHz. Επίσης, έχουν και οι 2 ενσωματωμένο το chip PN532 και μπορούμε να τις χρησιμοποιήσουμε, σε εφαρμογές που έχουν σχέση με το RFID αλλά και με το NFC. Το Shield είναι αυτό που εφαρμόζεται πάνω στο Arduino(Σχήμα 2.15) και το breakout είναι εξωτερικό και το συνδέουμε με το Arduino με καλώδια(Σχήμα 2.16). Αμφότερες οι πλακέτες, χρησιμοποιούν την ίδια βιβλιοθήκη, ώστε να μπορούν να λειτουργούν με τα πρωτόκολλα επικοινωνίας SPI και I2C. Το κόστος των πλακετών αυτών είναι στα 40\$.

Adafruit PN532 NFC/RFID Controller Shield



Σχήμα 2.15 Adafruit PN532 NFC/RFID Controller Shield.

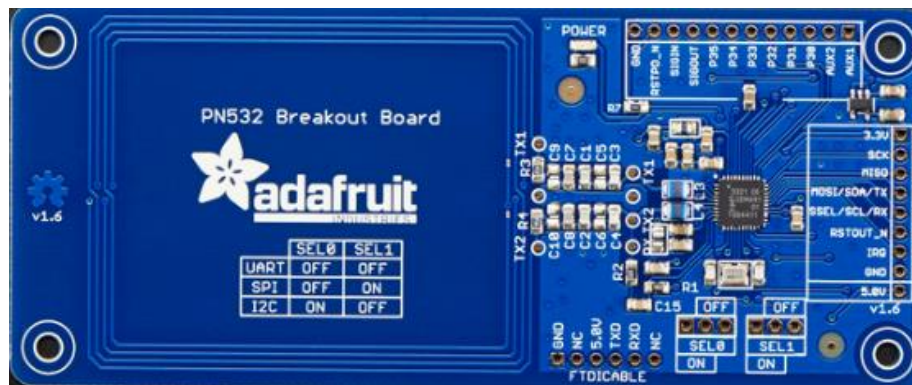
Πηγή: <https://www.adafruit.com/product/789> [11].

Την συγκεκριμένη πλακέτα, μπορούμε να την συνδέσουμε στο Arduino απευθείας, χωρίς την χρήση καλωδίων, όπως γίνεται στην περίπτωση του breakout. Ενώ το PN532, έχει πολλές δυνατότητες που αναφέρθηκαν στην ενότητα 2.2.2, παρόλα αυτά η βιβλιοθήκη που χρησιμοποιούμε για αυτήν την πλακέτα, υποστηρίζει αυτήν την στιγμή μόνο το διάβασμα και την εγγραφή ετικετών και καρτών. Μπορεί δηλαδή, να δουλέψει με ετικέτες NFC/RFID τύπου 1 έως 4. Όμως, δεν μας παρέχει την δυνατότητα, επικοινωνίας με κινητά τηλέφωνα που διαθέτουν NFC, αλλά ούτε προσομοίωση κάρτας(card emulation). Όσον αφορά την κεραία του, είναι

σχεδιασμένη ώστε να μπορεί να επικοινωνεί με κάρτες μέχρι τα 10 cm. Επίσης, η βιβλιοθήκη υποστηρίζει 2 πρωτόκολλα επικοινωνίας το SPI και το I2C. Η προκαθορισμένη (default) κατάσταση για το Shield, είναι η I2C. Εδώ χρησιμοποιούνται 3 pins, η ψηφιακή ακίδα 2 για διακοπές και οι αναλογικές ακίδες 4 και 5. Στην περίπτωση που θέλουμε SPI επικοινωνία, πρέπει να χρησιμοποιήσουμε 4 ακίδες που θέλουμε εμείς, αρκεί να βάλουμε 2 βραχυκυκλωτήρες στο πάνω μέρος του PCB. Επίσης, οι μικροελεγκτές που είναι συμβατοί με αυτό το Shield είναι, ο Arduino UNO, Diecimilla, Duemilanove, Mega R3. Εάν έχουμε I2C, μπορούμε να έχουμε Mega R2 αλλά και παλιότερες εκδόσεις [10].

Η βιβλιοθήκη: <https://github.com/adafruit/Adafruit-PN532/>

PN532 NFC/RFID controller breakout board – v1.6

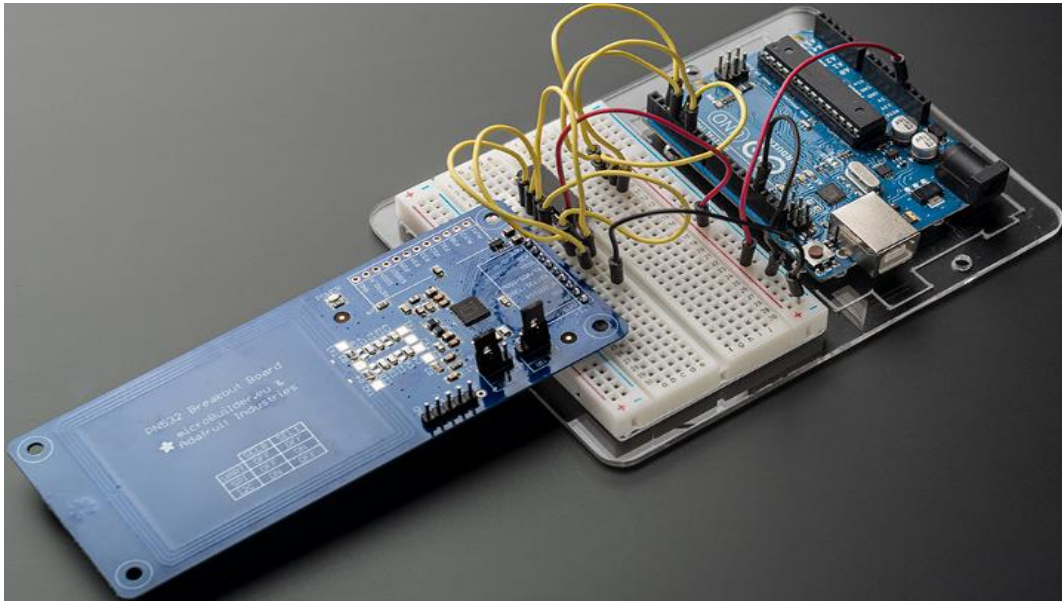


Σχήμα 2.16 PN532 NFC/RFID controller breakout board - v1.6

Πηγή: <https://www.adafruit.com/product/364> [20].

Το breakout δουλεύει και έχει τις ίδιες δυνατότητες και λειτουργίες με το Adafruit shield, η διαφορά ανάμεσα τους, είναι στο ότι πρέπει να χρησιμοποιήσουμε καλώδια για να το συνδέσουμε με το Arduino(Σχήμα 2.17) και επίσης έχει σχεδιαστεί να λειτουργεί στα 3.3V. Επομένως, επειδή οι ακίδες του Arduino δουλεύουν και παρέχουν τάση των 5V, για αυτό τον λόγο θα χρησιμοποιήσουμε το chip 4050, ώστε να έχουμε πτώση τάσης στα 3.3V, που δουλεύει η πλακέτα. Όσο αφορά την επιλογή του πρωτοκόλλου επικοινωνίας, θα χρειαστούμε 2 σειρές των τριών ακίδων για να τις

κολλήσουμε στα SEL0 και SEL1 και ανάλογα με ποιον τρόπο θα συνδέσουμε τους βραχυκυκλωτήρες, θα έχουμε ενεργοποίηση του κατάλληλου πρωτοκόλλου [12].



Σχήμα 2.17 Η συνδεσμολογία μεταξύ του Adafruit Breakout με το Arduino UNO μέσω του chip 4050. Πηγή: <https://www.adafruit.com/product/364> [20].

Παρακάτω επισυνάπτονται οι συνδέσεις μεταξύ του Arduino με το chip 4050 και με την πλακέτα Breakout:

Breakout:

3.3V -> 3.3V Arduino

GND -> GND Arduino

MISO -> Arduino(PIN 5)

SCK -> 4050(PIN 10)

SCL -> 4050(PIN 15)

MOSI -> 4050(PIN 12)

Chip 4050:

PIN 1 -> 3.3V Arduino

PIN 8 -> GND Arduino

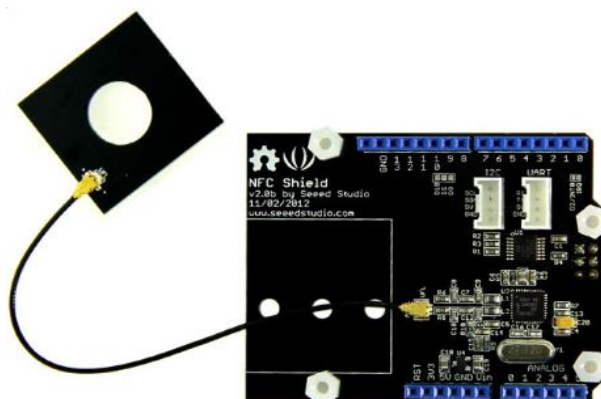
Arduino:

PIN2(ψηφιακή) -> 4050(PIN 9)

PIN3(ψηφιακή) -> 4050(PIN 11)

PIN4(ψηφιακή) -> 4050(PIN 14)

2.4.4 NFC Shield V2.0



Σχήμα 2.19 NFC Shield V2.0

Πηγή: <https://www.robotpark.com/NFC-Shield-V2-0-En> [21].

Είναι μια πλακέτα η οποία έχει ενσωματωμένο το chip PN532, είναι της εταιρίας Seeed studio και μπορούμε μέσω αυτής της πλακέτας να επιτύχουμε ασύρματη επικοινωνία στα 13.56 MHz, δηλαδή μπορούμε να γράψουμε και να διαβάσουμε μηνύματα από μια ετικέτα αλλά και μπορούμε να έχουμε P2P επικοινωνία, ανάμεσα στο NFC Shield V2.0 και ενός έξυπνου κινητού τηλεφώνου. Επίσης, διαθέτει και άλλες δυνατότητες, υποστηρίζει τα 3 πρωτόκολλα επικοινωνίας SPI,I2C και UART. Όσον αφορά το SPI, εδώ χρησιμοποιεί την κεφαλίδα ICSP, για αυτό τον λόγο η πλακέτα δουλεύει όσον αφορά το Arduino με τις πλακέτες,UNO,Mega και Leonardo. Επιπλέον, δουλεύει στα 5V και με ρεύμα 100mA, δημιουργεί ένα ηλεκτρομαγνητικό πεδίο όπου, μπορεί να έρθει σε επαφή με μια συσκευή ή κάρτα μέχρι τα 5 cm. Οι κάρτες με τις οποίες επικοινωνεί, πρέπει να υποστηρίζουν τα πρωτόκολλα ISO 14443A και B. Αυτή η πλακέτα, μπορεί να έχει μόνο μια κεραία. Η κεραία της δεν είναι όπως στις προηγούμενες συσκευές ανάγνωσης, που αναφέρθηκαν παραπάνω, συνδέεται με την πλακέτα με ένα καλώδιο. Το καλώδιο αυτό είναι 12 cm,εάν θέλουμε να χρησιμοποιήσουμε μεγαλύτερο καλώδιο, μπορούμε να βάλουμε ένα μέχρι

τα 15 cm. Όσον αφορά το κόστος της, είναι η πιο ακριβή πλακέτα από αυτές που αναφέρθηκαν σε αυτό το κεφαλαίο, με το κόστος της να ανέρχεται περίπου στα 60 \$ [13]. Τέλος, μπορεί να υποστηρίξει και να επικοινωνήσει με αρκετά μοντέλα έξυπνων κινητών τηλεφώνων της εταιρίας LG, της εταιρίας SAMSUNG και επίσης υποστηρίζει και το Oneplus 6, όμως όσα κινητά αναφέρονται στην ιστοσελίδα του github δεν σημαίνει ότι υποστηρίζει μόνο αυτά, θα πρέπει οποιοσδήποτε θέλει να χρησιμοποιήσει αυτήν την συσκευή ανάγνωσης για κάποια εργασία με το κινητό του τηλέφωνο, να δοκιμάσει με την δική του συσκευή [14].

3 RFID σύστημα

ΣΚΟΠΟΣ

Στο κεφάλαιο 3 γίνεται αναφορά σε ένα RFID σύστημα, το οποίο δημιουργήθηκε με σκοπό να διαβάζουμε τα ID καρτών MIFARE με την χρήση μιας RFID συσκευής ανάγνωσης τον RC522 και σε συνδυασμό με μια εφαρμογή στο κινητό να ξεκλειδώνει μια κλειδαριά. Θα γίνει αναφορά στο σχεδιασμό του συστήματος, στο κόστος των υλικών που χρησιμοποιήθηκαν, στην λειτουργία του συστήματος, στον κώδικα της εφαρμογής και τέλος στον κώδικα που γράφτηκε στο προγραμματιστικό περιβάλλον Arduino IDE.

3.1 Σχεδίαση του κυκλώματος

3.1.1 Περιγραφή των υλικών

Αρχικά, για την υλοποίηση της εφαρμογής χρησιμοποιήθηκε ο μικροελεγκτής Arduino UNO, ο οποίος προγραμματίστηκε στο προγραμματιστικό περιβάλλον Arduino IDE. Επίσης, χρησιμοποιήθηκαν 2 LED ένα πράσινο και ένα κόκκινο, ένα buzzer, ένας σερβοκινητήρας SG90, η πλακέτα HC-05 και ένα κινητό τηλέφωνο οπότε έχουμε εγκατεστημένη την εφαρμογή, η οποία σχεδιάστηκε και προγραμματίστηκε στο προγραμματιστικό περιβάλλον MIT App Inventor.

- **Ποιο αναλυτικά τα υλικά που χρησιμοποιήθηκαν :**

- **Breadboard:** Πάνω στο οποίο έγινε η συναρμολόγηση του κυκλώματος.
- **Arduino Uno:** Αποτελεί τον μικροελεγκτή του συστήματος, μέσω του οποίου θα γίνεται ο έλεγχος των διαφόρων εξαρτημάτων του συστήματος, βάση του προγράμματος, το οποίο γράφτηκε στο Arduino IDE και που είναι αποθηκευμένο στην μνήμη του.



- **RFID συσκευή ανάγνωσης (RC 522):** Αποτελεί την συσκευή ανάγνωσης του συστήματος και χρησιμοποιείται για το διάβασμα των ID, από κάρτες MIFARE.



- **Jumpers wires(MEN-TO-MEN):** Χρησιμοποιήθηκαν για την σύνδεση των ψηφιακών ακίδων του Arduino με τα εξαρτήματα του συστήματος.
- **2 Αντιστάσεις 330 Ω:** Οι αντιστάσεις αυτές, τοποθετήθηκαν μεταξύ των ψηφιακών ακίδων του Arduino και των ανόδων των LED.

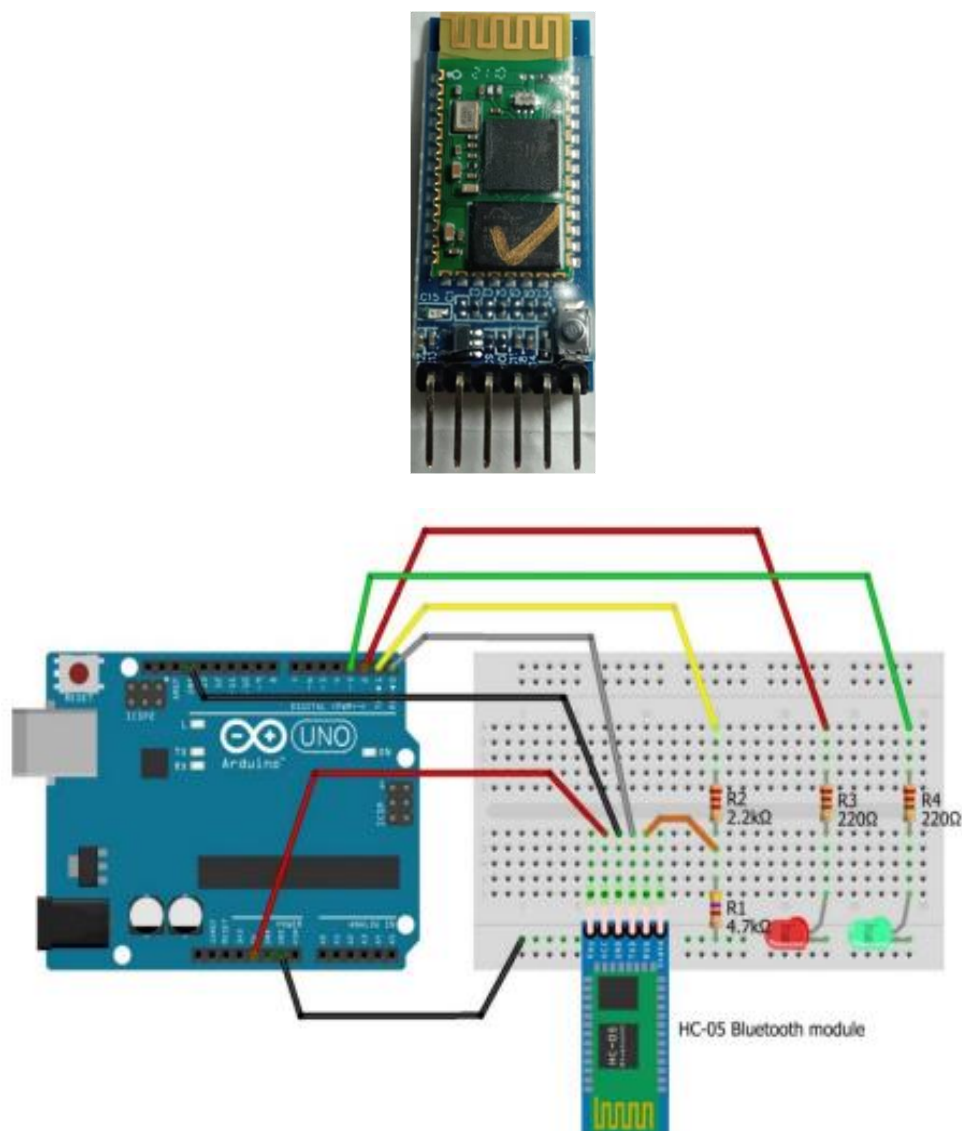
- **2 LED(Πράσινο και κόκκινο):** Χρησιμοποιήθηκαν για να έχουμε οπτική ειδοποίηση στην περίπτωση που σταλούν έγκυρα ή άκυρα δεδομένα.
- **Buzzer:** Έγινε χρήση ενός Buzzer για ηχητική ειδοποίηση, όπου στην περίπτωση που σταλούν δεδομένα που είναι έγκυρα ακούγεται για μια φορά ένας συνεχόμενος ήχος μεγάλης συχνότητας, ενώ όταν τα δεδομένα δεν είναι έγκυρα ακούγεται 3 φορές ένας ήχος χαμηλής συχνότητας.



- **Servo Motor (SG 90):** Ο σερβοκινητήρας, επειδή δεν χρησιμοποιήθηκε ηλεκτρομαγνητική κλειδαριά, θα παίζει το ρολό της κλειδαριάς.



- **HC-05:** Το HC-05 είναι μια πλακέτα ασύρματης επικοινωνίας και έχει ένα πομπό (TX) και ένα δέκτη (RX). Ο πομπός του συνδέεται απευθείας με το PIN 0 του Arduino, ενώ ο δέκτης του HC-05 συνδέεται με το PIN 1 του Arduino όχι όμως απευθείας. Επειδή, ο δέκτης του HC-05 δουλεύει στα 3.3 V και ο πομπός του Arduino στέλνει 5 V δημιουργήθηκε ένας διαιρέτης τάσης με μια αντίσταση 2.2 kΩ και άλλη μια 4.7 kΩ, ώστε να μην υπάρξει καταστροφή του δέκτη [15].



Σχήμα 3.1 Συνδεσμολογία ανάμεσα στο Arduino UNO και το HC-05. Πηγή: <https://www.vodafonegenerationnext.gr/data/lessons/31/downloads/Internet%20of%20Things%202-Presentation.pdf> [22].

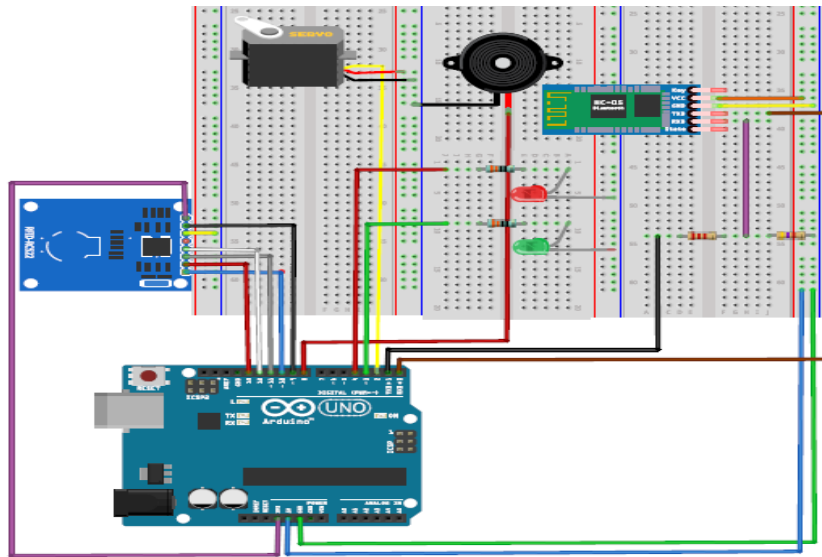
- Αντίσταση 2.2 kΩ
- Αντίσταση 4.7 kΩ
- **Κινητό τηλέφωνο:** Χρησιμοποιήθηκε για την εγκατάσταση της εφαρμογής, μέσω της οποίας γίνεται το ξεκλείδωμα της κλειδαριάς.

3.1.2 Κόστος κατασκευής

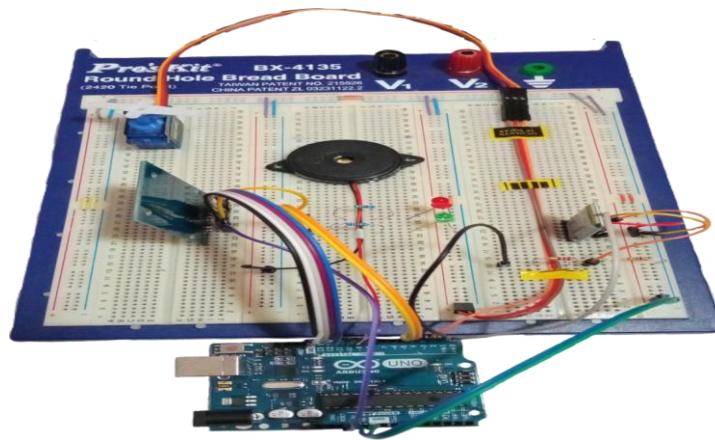
	Υλικά	Κόστος
	Arduino UNO	7 €
	RC522	5 €
	LED-Αντιστάσεις- Καλώδια	3 €
	Buzzer	2 €
	HC-05	7 €
	Servomotor	4 €
Σύνολο		28 €

Σχήμα 3.2 Τα υλικά της RFID εφαρμογής και το κόστος τους.

3.2 Λειτουργία RFID συστήματος

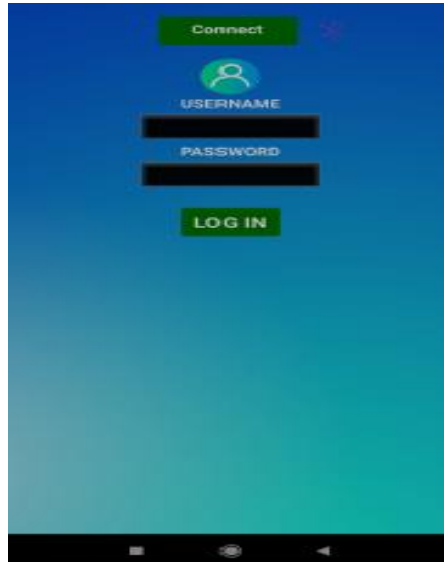


Σχήμα 3.3 Η σχεδίαση του κυκλώματος στο Fritzing.



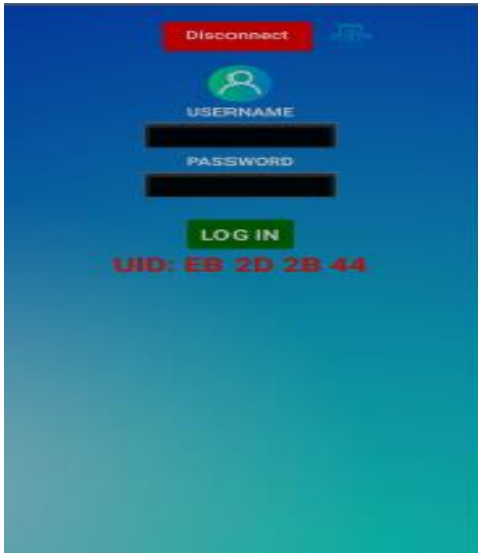
Σχήμα 3.4 Η υλοποίηση του κυκλώματος στο Breadboard.

Αρχικά, θα πρέπει να ενεργοποιήσουμε το Bluetooth του κινητού μας και να ανοίξουμε την εφαρμογή, όπου από εκεί θα συνδέσουμε το κινητό μας μέσω Bluetooth με το HC-05, ώστε να επιτύχουμε την επικοινωνία του κινητού με το Arduino για την ανταλλαγή δεδομένων, αυτό θα γίνει πατώντας “Connect” μέσω της εφαρμογής(Σχήμα 3.5) .

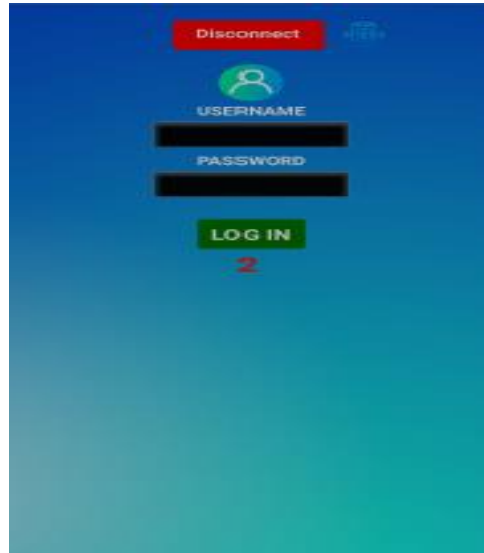


Σχήμα 3.5 Η όψη της εφαρμογής στο κινητό.

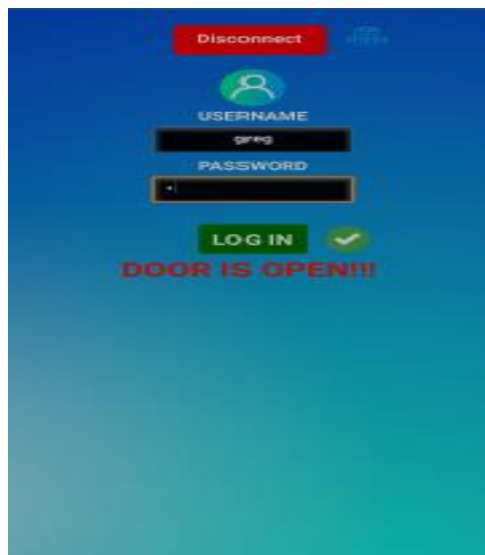
Όταν μια κάρτα διαβαστεί, θα δούμε το ID της (**Σχήμα 3.6**) και εάν είναι έγκυρη τότε θα δούμε σαν μήνυμα στην εφαρμογή μας το μήνυμα “ACCESS”.



Σχήμα 3.6 Η εμφάνιση του UID της κάρτας που διαβάστηκε, στην εφαρμογή.



Σχήμα 3.7 Η αντίστροφη μέτρηση που γίνεται για την καταχώρηση των στοιχείων.



Σχήμα 3.8 Η εμφάνιση μηνύματος μόλις η πόρτα ανοίξει.

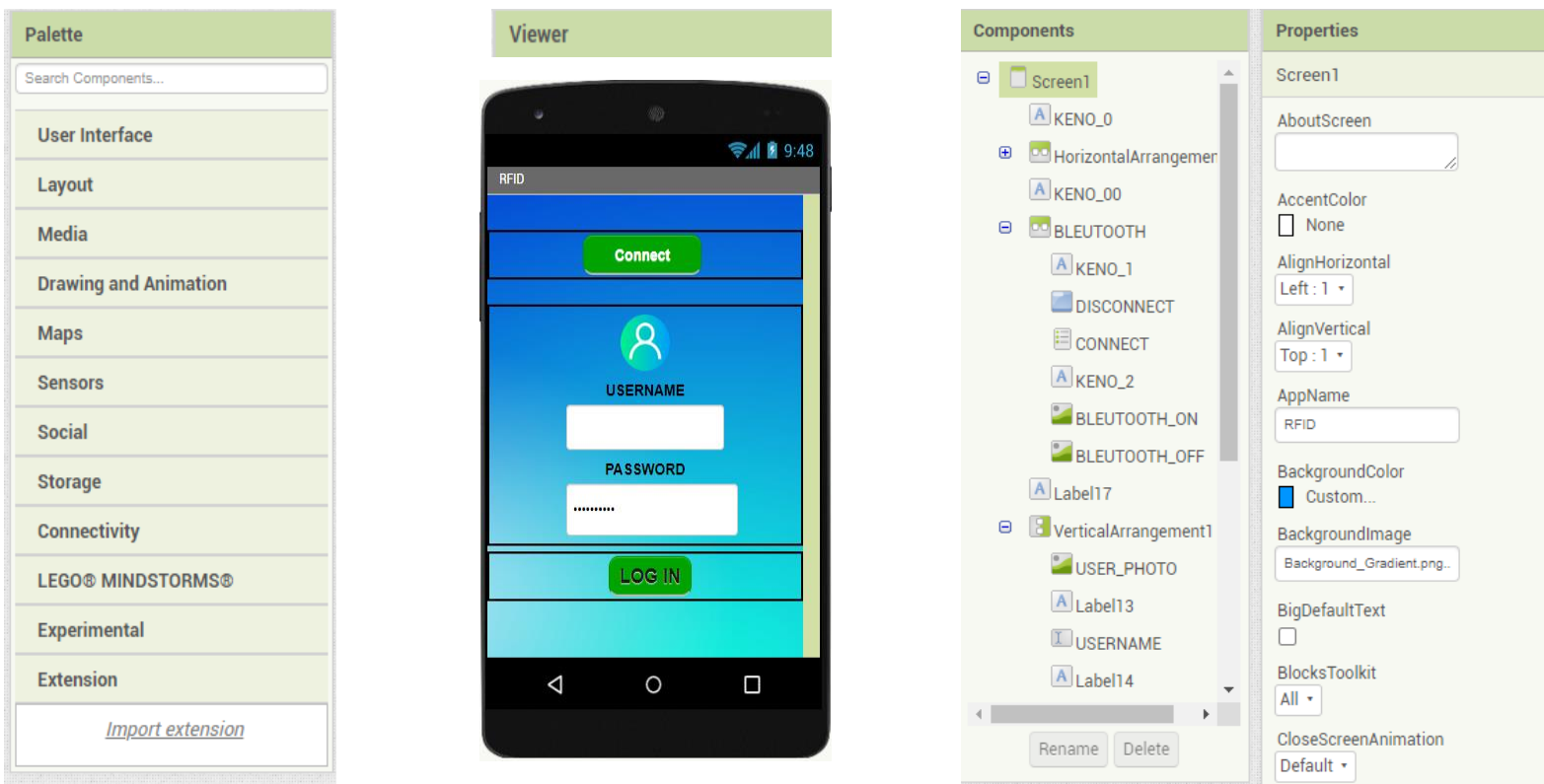
Θα δοθεί χρόνος 5 δευτερόλεπτων ώστε να καταχωρήσουμε τα στοιχεία μας(Σχήμα 3.7), τα οποία είναι το όνομα χρηστή και ένας κωδικός, και έπειτα πατώντας LOG IN να ανοίξει η κλειδαριά. Εάν δεν προλάβουμε να καταχωρήσουμε τα στοιχεία μας σε αυτό το χρονικό διάστημα, θα πρέπει να ξανά διαβαστεί η κάρτα. Αφού η κάρτα είναι έγκυρη και έχουμε γράψει σωστά τα στοιχεία μας, τότε θα δούμε το μήνυμα **“DOOR IS OPEN”** στην εφαρμογή (Σχήμα 3.8), επίσης το πράσινο led θα αναβοσβήσει 3 φορές, ώστε να έχουμε οπτική ειδοποίηση. Το Buzzer θα ηχήσει και ο σερβοκινητήρας που παίζει το ρόλο της κλειδαριάς θα πάει από την αρχική του κατάσταση, η οποία είναι οι 0 μοίρες, στις 180 μοίρες, θα παραμείνει εκεί για 5 δευτερόλεπτα ώστε να προλάβουμε να “ανοίξουμε” την πόρτα και έπειτα θα επιστρέψει στην αρχική του κατάσταση, βλέποντας στην εφαρμογή το μήνυμα **“DOOR IS CLOSED ”**. Στην περίπτωση της άκυρης κάρτας, απλά θα δούμε το ID της στην εφαρμογή μας, το κόκκινο led θα αναβοσβήσει 3 φορές και το buzzer θα ηχήσει ώστε να έχουμε ηχητική ειδοποίηση και έπειτα θα δούμε το μήνυμα **“REJECTION”** στην οθόνη του κινητού μας.

3.3 ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ RFID ΣΥΣΤΗΜΑΤΟΣ

3.3.1 MIT app Inventor

Το MIT App Inventor είναι μια πλατφόρμα, στην οποία μπορούμε να προγραμματίσουμε και να σχεδιάσουμε την δική μας εφαρμογή για το κινητό μας τηλέφωνο. Αποτελείται από το σχεδιαστικό περιβάλλον, όπου εκεί από τα εργαλεία τα οποία μας παρέχει αυτή η πλατφόρμα, μπορούμε να διαμορφώσουμε την όψη, που θα έχει η εφαρμογή μας στο κινητό όταν θα την ανοίξουμε. Επίσης αποτελείται, από το προγραμματιστικό περιβάλλον, όπου εκεί ανάλογα με τις προσθήκες που κάναμε στο σχεδιαστικό μέρος, προγραμματίζουμε κατάλληλα την εφαρμογή μας ανάλογα με το τι ενέργειες θέλουμε να πραγματοποιεί.

• ΣΧΕΔΙΑΣΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ



Σχήμα 3.9 Το σχεδιαστικό περιβάλλον του MIT App Inventor και η όψη της εφαρμογής. Πηγή: <https://appinventor.mit.edu/> [16].

Στο σχεδιαστικό περιβάλλον έχουμε την περιοχή **pallette**, η οποία αποτελείται από διαφορές κατηγορίες από τις οποίες παίρνουμε τα αντικείμενα τα οποία θα τα χρησιμοποιήσουμε για την οθόνη, όπως για παράδειγμα, εικόνες που θέλουμε να εμφανίζονται για να μας ειδοποιήσουν για κάτι, ψηφιακά μπουτόν ώστε να μπορούμε για παράδειγμα, πατώντας το μπουτόν να ανάψουμε ένα Led και οποιοδήποτε άλλο αντικείμενο χρειαζόμαστε για την οθόνη. Στο μενού **Viewer** βλέπουμε, το πώς θα είναι η εφαρμογή μας μόλις την ανοίξουμε στο κινητό. Στην συνέχεια στο **components**, παρατηρούμε τα αντικείμενα που προσθέσαμε από την περιοχή pallette και από το **properties**, μπορούμε να προσθέσουμε background στην εφαρμογή και να αλλάζουμε μέγεθος ή χρώμα στα αντικείμενα που έχουμε τοποθετήσει [16].

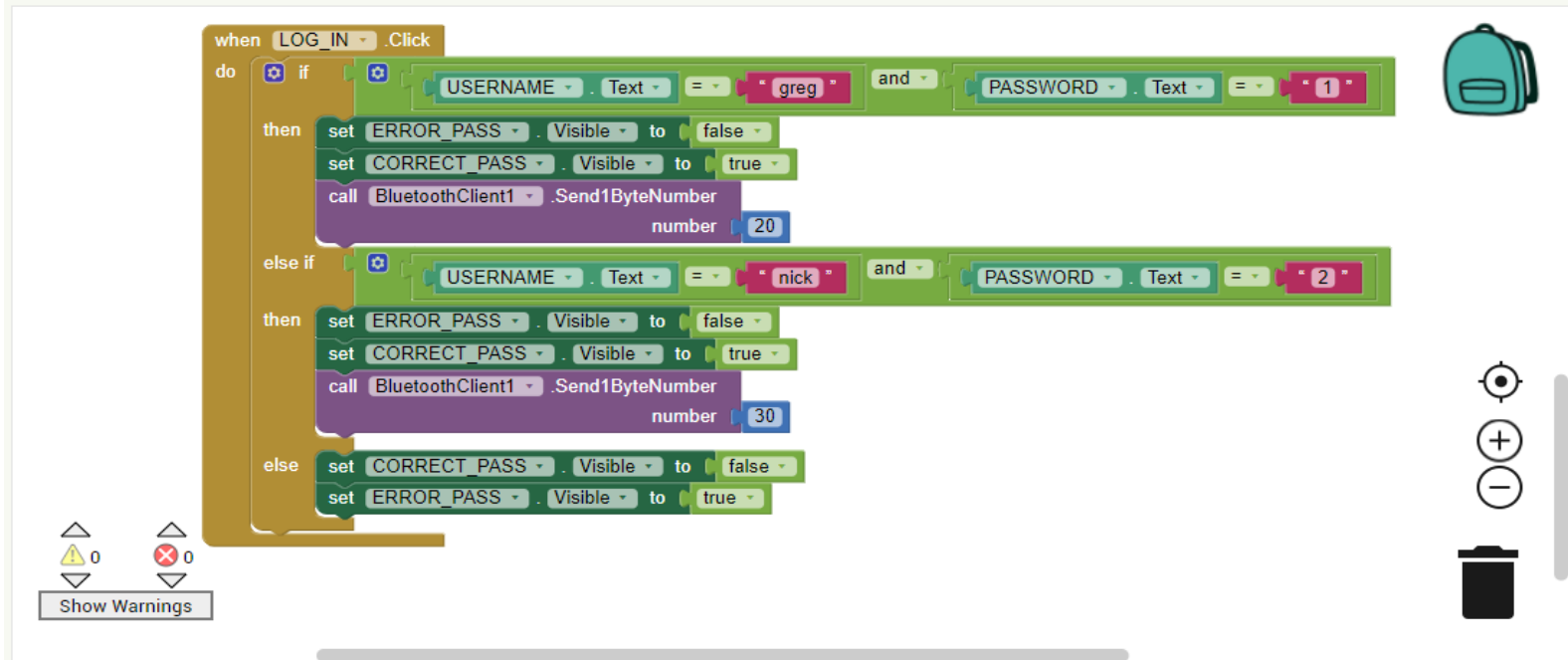
• ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ

The image displays the MIT App Inventor interface. On the left is the 'Blocks' palette with categories: Built-in, Control, Logic, Math, Text, Lists, Dictionaries, Colors, Variables, and Procedures. Below it are project components: Screen1, KENO_0, HorizontalArrangement, KENO_00, BLEUTOOTH, KENO_1, and DISCONNECT. The main workspace contains the following code blocks:

- when CONNECT .BeforePicking**
 - do set CONNECT . Elements to BluetoothClient1 . AddressesAndNames
- when CONNECT .AfterPicking**
 - do if call BluetoothClient1 . Connect address CONNECT . Selection
 - then set CONNECT . Elements to BluetoothClient1 . AddressesAndNames
 - if BluetoothClient1 . IsConnected
 - then set BLEUTOOTH_ON . Visible to true
 - set BLEUTOOTH_OFF . Visible to false
 - set CONNECT . Visible to false
 - set DISCONNECT . Visible to true
- when DISCONNECT .Click**
 - do call BluetoothClient1 . Disconnect
 - set CONNECT . Visible to true
 - set DISCONNECT . Visible to false
 - set BLEUTOOTH_OFF . Visible to true
 - set BLEUTOOTH_ON . Visible to false
- when Clock1 .Timer**
 - do if BluetoothClient1 . IsConnected and call BluetoothClient1 . BytesAvailableToReceive > 0
 - then set ΟΘΟΝΗ_ΜΗΝΥΜΑΤΩΝ . Text to call BluetoothClient1 . ReceiveText
 - set ΟΘΟΝΗ_ΜΗΝΥΜΑΤΩΝ . Visible to true
 - set ΟΘΟΝΗ_ΜΗΝΥΜΑΤΩΝ . TextColor to red

Σχήμα 3.10 Το προγραμματιστικό περιβάλλον του MIT App Inventor και ο κώδικας της εφαρμογής (Μέρος 1^ο).

Πηγή: <https://appinventor.mit.edu/> [16].



Σχήμα 3.11 Το προγραμματιστικό περιβάλλον του MIT App Inventor και ο κώδικας της εφαρμογής (Μέρος 2^ο).
 Πηγή: <https://appinventor.mit.edu/> [16].

Στο προγραμματιστικό περιβάλλον από το μενού **Blocks**(Σχήμα 3.10), χρησιμοποιούμε τα εργαλεία από τα οποία θα δημιουργήσουμε το πρόγραμμα μας, το οποίο είναι σε μορφή πάζλ. Αρχικά στο πρόγραμμα μας, έχουμε 5 blocks. Το πρώτο είναι το **CONNECT.BeforePicking** (Σχήμα 3.10), το οποίο το χρησιμοποιούμε ώστε να μπορέσουμε να συνδέσουμε το Bluetooth με την εφαρμογή. Το επόμενο μπλοκ είναι το **DISCONNECT.Click**, το οποίο δημιουργήθηκε ώστε στην περίπτωση που δεν έχουμε συνδεθεί με το Bluetooth:

- 1) Να είναι εμφανές το μπουτόν “**CONNECT**”, ώστε να μπορούμε να το πατήσουμε και να συνδεθούμε με το Bluetooth.
- 2) Να είναι κρυμμένο το μπουτόν “**DISCONNECT**”.
- 3) Να είναι εμφανείς η εικόνα “**BLUETOOTH OFF**”, που μας δείχνει ότι είμαστε αποσυνδεδεμένοι, και κρυμμένη η εικόνα “**BLUETOOTH ON**”.

Στην συνέχεια, έχουμε το block **CONNECT.AfterPicking**, το οποίο θα τεθεί σε λειτουργία μετά το πάτημα του “**CONNECT**”, όπου:

- 1) Θα εμφανιστεί το μπουτόν “**DISCONNECT**”, ώστε όποτε θέλουμε να αποσυνδεθούμε από το Bluetooth, να το κάνουμε πατώντας το μπουτόν αυτό.

2) Θα έχουμε εμφάνιση της εικόνας(**BLUETOOTH ON**) που στο προηγούμενο block(**DISCONNECT.Click**) παραπάνω είχε εξαφανιστεί.

Επίσης, έχουμε το block **Clock1.Timer**, ώστε να μπορεί η εφαρμογή να λαμβάνει και να εμφανίζει τα μηνύματα που έχουν γραφτεί με την εντολή **Serial Println()** στο Arduino IDE, με κόκκινο χρώμα στην οθόνη του κινητού μας, τα μηνύματα αυτά εμφανίζονται επίσης και στην σειριακή οθόνη.

Τέλος, έχουμε το block **LOG_IN.Click** (**Σχήμα 3.11**) όπου εδώ έχουμε 2 περιπτώσεις. Στην μια περίπτωση έχουμε ως **USERNAME** το όνομα “greg” και ως **PASSWORD** τον αριθμό ”1” και στην άλλη περίπτωση το όνομα “nick” με κωδικό το “2”. Σε κάθε περίπτωση, όταν γράψουμε σωστά αυτά τα στοιχεία και πατήσουμε LOG IN θα μας εμφανιστεί μια εικόνα με ένα πράσινο “Τικ”(Σχήμα 3.8) ή ένα κόκκινο “X”, εάν τα στοιχεία μας είναι λανθασμένα. Εάν πάρουμε για παράδειγμα, την πρώτη περίπτωση και αφού γράψουμε σωστά τα στοιχεία μας όταν θα μας ζητηθούν, τότε, όταν θα πατήσουμε LOG IN, θα σταλθεί σαν δεδομένο στο Arduino ο αριθμός “20”ή το “30” στην δεύτερη περίπτωση. Από κει και πέρα το Arduino θα αναλάβει να επεξεργαστεί κάθε δεδομένο που του στέλνουμε, ώστε να ξεκλειδώσει η κλειδαριά και να δούμε στην οθόνη μας το αντίστοιχο μήνυμα(Σχήμα 3.8).

3.3.2 Κώδικας Arduino IDE

- ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ

1) Αρχικά, δηλώθηκαν οι βιβλιοθήκες που ήταν απαραίτητες για την λειτουργία του προγράμματος. Οι οποίες είναι η **SPI.h**, διότι έχουμε SPI επικοινωνία μεταξύ του Arduino και του HC-05, η **MFRC522.h** για να έχουμε στην διάθεση μας τις λειτουργίες της συσκευής ανάγνωσης και την βιβλιοθήκη **Servo.h** για να μπορέσει να λειτουργήσει ο σερβοκινητήρας.

```
//BIBΛΙΟΘΗΚΕΣ
```

```
( #include <SPI.h>  
#include <MFRC522.h>  
#include <Servo.h> )
```

2) Κάνουμε τις απαραίτητες αρχικοποιήσεις για όλες τις μεταβλητές, ώστε να είναι ορατές σε όλο το πρόγραμμα και οι οποίες χρειάζονται για την λειτουργία του προγράμματος.

```
(char Data=0; //Αρχικοποίηση μεταβλητής, στην οποία θα καταχωρούνται τα  
δεδομένα που έρχονται από την εφαρμογή του κινητού. )
```

```
//ID CARDS
```

- Δηλώθηκαν τα ID 2 καρτών, και στην περίπτωση που θα διαβαστεί ένα από τα δυο, μόνο τότε θα έχουμε την δυνατότητα να συμπληρώσουμε τα στοιχεία μας, απλώς όταν διαβαστεί μια κάρτα θα πρέπει να σταλούν από την εφαρμογή και τα κατάλληλα στοιχεία για την κάρτα που διαβάστηκε, ώστε να ξεκλειδώσει η κλειδαριά.

```
( String card_1="36 70 6E AF";  
String card_2="35 84 D0 1B"; )
```

- Έχουμε 2 LED, τα οποία είναι συνδεδεμένα με τα PIN 3 και 4 του Arduino. Επίσης, γίνεται δήλωση και των υπόλοιπων ακίδων του Arduino που θα χρησιμοποιηθούν.

```
( //LED  
  
#define led_green 3  
#define led_red 4  
  
  
  
//BUZZER  
  
int buzzer=8;  
  
  
//SERVO  
  
Servo servo ;  
#define servo_pin 2  
  
  
//RFID  
  
#define SS_PIN 10  
#define RST_PIN 9  
  
MFRC522 mfrc522(SS_PIN,RST_PIN); )
```

3) Στην συνάρτηση **setup**, αρχικά ενεργοποιούμε την σειριακή επικοινωνία στα 9600 bps και ορίζουμε τα pins του Arduino τα οποία χρειάστηκαν για το project, ως εξόδους. Γενικά, η default κατάσταση των ακίδων του Arduino είναι προσαρμοσμένοι ως είσοδοι, στην δική μας περίπτωση δεν είχαμε κάποια ακίδα να είναι είσοδος παρ' όλα αυτά, γενικά καλό είναι, να τα γραφούμε για να γνωρίζουμε τι είναι η κάθε ακίδα.

```
( void setup()
{
  //RFID
  Serial.begin(9600);
  SPI.begin();
  mfrc522.PCD_Init(); )
```

```
//SERVO
```

- Ο σερβοκινητήρας αποτελείται από 3 καλώδια, ένα καφέ το οποίο το συνδέουμε στην γείωση, ένα κόκκινο που συνδέεται στα 5V και ένα πορτοκαλί το οποίο το συνδέουμε στο PIN 2 του Arduino, το οποίο καθορίσαμε στο πρόγραμμα μας με την εντολή **servo.attach**. Επίσης, με την εντολή **servo.write** καθορίζουμε τις μοίρες, που θα βρίσκεται ο σερβοκινητήρας μόλις τον θέσουμε σε λειτουργία [17].

```
( servo.attach(servo_pin);
servo.write(0); )
```

- Με την εντολή **pinMode**, γίνεται καθορισμός μιας ψηφιακής ακίδας του Arduino σε είσοδο ή έξοδο, στο πρόγραμμα αυτό οι ακίδες καθορίστηκαν ως έξοδοι.

```

( //ΕΞΟΛΟΙ LED

pinMode (led_red, OUTPUT);

pinMode (led_green, OUTPUT);

//ΕΞΟΛΟΣ BUZZER

pinMode(buzzer, OUTPUT);

}

//ΤΕΛΟΣ void setup() )

```

4) Στην συνάρτηση **loop** υπάρχει το κυρίως πρόγραμμα και γίνονται οι κλήσεις των συναρτήσεων του προγράμματος.

```

( void loop()

{

Data=0; // Μηδενίζουμε την μεταβλητή, ώστε μόλις ξεκινήσει η εκτέλεση της
loop, να διαγράφεται το προηγούμενο δεδομένο που είχε.

if( ! mfrc522.PICC_IsNewCardPresent())

{

return;

}

if( ! mfrc522.PICC_ReadCardSerial())

{

return;

```

```

}

Serial.print("ID:");

String door="";

for(byte i=0; i<mfr522.uid.size; i++)
{
  Serial.print(mfr522.uid.uidByte[i] < 0x10 ? "0:" " ");
  Serial.print(mfr522.uid.uidByte[i],HEX);
  door.concat(String(mfr522.uid.uidByte[i] < 0x10 ? "0:" " "));
  door.concat(String(mfr522.uid.uidByte[i],HEX));
}

door.toUpperCase(); )

```

- 5) Στην περίπτωση διαβάσματος μιας κάρτας, εάν μια από τις δυο συνθήκες είναι αληθής τότε, θα μπει μέσα στον βρόχο όπου γίνεται κλήση της συνάρτησης pass και εάν έχει σταλθεί σαν δεδομένο στο Arduino για παράδειγμα ο αριθμός 20, από την εφαρμογή μέσω Bluetooth, δηλαδή έβαλε ο χρήστης το όνομα ->greg και τον κωδικό ->1 και ενώ έχει διαβαστεί νωρίτερα η card_1, τότε θα ξεκλειδώσει η κλειδαριά(Servo_Motor).

```

( //KAPTA_1
if(door.substring(1)==card_1)
{
  pass(20);
  // USERNAME: greg
  // PASSWORD:1

```

```

}

//KAPTA_2
else if(door.substring(1)==card_2)
{
    pass(30);
    // USERNAME: nick
    // PASSWORD:2
} )

```

- Περίπτωση άκυρης κάρτας, της οποίας το ID δεν έχει συμπεριληφθεί στο πρόγραμμα.

```

( else
{
    delay(2000);
    Serial.println();

    Serial.println("REJECTION");
    delay(1000);

    led(led_red,3,250);
    buzz(3,500,250);
    delay(1000);

}
} )

```

6) Για κάθε κομμάτι του κώδικα που αφορά μια λειτουργία του κυκλώματος μας, όπως για παράδειγμα το άναμμα ή σβήσιμο των LED, δημιουργήθηκαν συναρτήσεις για καλύτερη κατανομή του προγράμματος, αποφυγή επαναλαμβανόμενων εντολών και ευκολότερη ανίχνευση σφαλμάτων.

- Η συνάρτηση **led** είναι τύπου void, δηλαδή δεν επιστρέφει καμία τιμή στο κυρίως πρόγραμμα. Αποτελείται από 3 παραμέτρους οι οποίοι είναι τύπου Int(ακέραιοι). Η 1^η παράμετρος, δέχεται ως όρισμα το όνομα του Led, το οποίο θέλουμε να αναβοσβήνει. Η 2^η παράμετρος, έχει ως όρισμα τις φορές που θέλουμε να αναβοσβήσει το led και στην 3^η παράμετρο, βάζουμε ως όρισμα την καθυστέρηση που θέλουμε να έχουμε, από την στιγμή που θα ανάψει το led μέχρι που να σβήσει και το αντίστροφο.

```
( //ΣΥΝΑΡΤΗΣΗ LED
```

```
void led(int ledpin, int number, int del)
```

```
{
```

```
for(int i=0; i<number; i++)
```

```
{
```

```
digitalWrite(ledpin, HIGH);
```

```
delay(del);
```

```
digitalWrite(ledpin, LOW);
```

```
delay(del);
```

```
}
```

```
} )
```

- Η συνάρτηση **buzz** είναι τύπου void και έχει 3 ακέραιες παραμέτρους. Στην 1^η παράμετρο, βάζουμε το πόσες φορές θέλουμε να ηχήσει το buzzer. Στην 2^η παράμετρο, βάζουμε την συχνότητα του ήχου που θέλουμε να παραχθεί από το buzzer και στην 3^η παράμετρο, βάζουμε την καθυστέρηση ανάμεσα στις ηχητικές ειδοποιήσεις του buzzer. Στην περίπτωση που μια κάρτα είναι έγκυρη, ο ήχος που βγάζει το buzzer έχει μεγαλύτερη διάρκεια και μεγαλύτερη συχνότητα, ενώ όταν η κάρτα είναι άκυρη έχουμε έναν επαναλαμβανόμενο ήχο μικρότερης συχνότητας.

```
( //ΣΥΝΑΡΤΗΣΗ BUZZER
void buzz(int number, int freq, int del)
{
for(int i=0; i<number; i++)
{
tone(buzzer, freq);
delay(del);
noTone(buzzer);
delay(del);
}
} )
```

- Η συνάρτηση αυτή είναι τύπου void και έχει μια ακέραιη παράμετρο, της οποίας το όρισμα είναι η τιμή που έρχεται από την εφαρμογή του κινητού.

```
( //ΣΥΝΑΡΤΗΣΗ ΕΓΚΥΡΗΣ ΚΑΡΤΑΣ
void pass(int rec )
{
delay(2000);
Serial.println(); )
```


- Αφού το πρόγραμμα μπει μέσα στην συνάρτηση, βλέπουμε στο κινητό μας αλλά και στην σειριακή οθόνη το μήνυμα **ACCESS**.

```
( Serial.println("ACCESS ");
```

```
delay(2000); )
```

```
//BUFFER ΕΙΣΕΡΧΟΜΕΝΩΝ ΔΕΔΟΜΕΝΩΝ
```

- Εδώ γίνεται έλεγχος, εάν έχει έρθει κάποιο δεδομένο από την εφαρμογή.

```
( while(Serial.available())
```

```
Serial.read(); )
```

- Στην περίπτωση που η κάρτα είναι έγκυρη, θα μας ζητηθεί να βάλουμε τα στοιχεία μας για την συγκεκριμένη κάρτα και να πατήσουμε Log In ώστε να σταλούν στο Arduino, επίσης μας δίνεται ο χρόνος των 5s για την καταχώρηση των στοιχείων, εάν δεν προλάβουμε σε αυτό το χρόνο να γράψουμε τα στοιχεία μας ή βάλουμε λάθος στοιχεία τότε θα δούμε τα μηνύματα **“TRY AGAIN”, “PUT YOUR CARD”**.

```
( if(Data==false) // Εάν η κάρτα είναι έγκυρη, αλλά δεν έχει ληφθεί δεδομένο τότε..
```

```
{
```

```
Serial.println("Put your password");
```

```
Serial.println();
```

```
delay(2000);
```

```
for(int i=5; i>=0; i--) //Περίμενε 5s
```

```
{
```

```
Serial.println(i);
```

```
delay(1000);
```

```
}
```

```
//Serial_ available
```

```
if(Serial.available()) //Διαβάζω εάν έχουν έρθει δεδομένα.
```

```
{
```

```
Data=Serial.read(); //Αφού ήρθαν, θα διαβαστούν στην μεταβλητή Data. )
```

- Εδώ γίνεται ο έλεγχος του δεδομένου που έχει ληφθεί, εάν όλα έχουν γίνει σωστά τότε θα ανάψει το πράσινο led, θα ηχήσει το buzzer και ο σερβοκινητήρας θα πάει στις 180 μοίρες, θα παραμείνει εκεί 5 s, και έπειτα από το πέρας αυτού του χρόνου η κλειδαριά θα κλειδώσει, δηλαδή ο σερβοκινητήρας θα επιστρέψει στις 0 μοίρες και θα δούμε το μήνυμα “**DOOR IS CLOSE**”.

```
( if(Data==rec) //Εάν το δεδομένο είναι...
```

```
{
```

```
led(led_green,3,150);
```

```
buzz(1,800,1500);
```

```

//SERVO_DOOR

servo.write(180); //Ξεκλειδώνει η κλειδαριά.

Serial.println("DOOR IS OPEN!!!");

delay(1000);

for(int i=5; i>=0; i--) //Περιμένει 5s
{
Serial.println(i);

delay(1000);

}

servo.write(0); //Κλειδώνει η κλειδαριά.

delay(1000);

Serial.println("DOOR IS CLOSED!!!");

}

}

else

{

//ΜΗΝΥΜΑΤΑ

Serial.println("TRY AGAIN");

delay(1000);

Serial.println("PUT YOUR CARD");

}

//ΤΕΛΟΣ

} )

```

4 NFC σύστημα

ΣΚΟΠΟΣ

Στο κεφάλαιο 4, γίνεται αναφορά σε ένα NFC σύστημα, το οποίο δημιουργήθηκε με σκοπό να διαβάζουμε NDEF μηνύματα από MIFARE κάρτες ή tags με την χρήση μιας NFC συσκευής ανάγνωσης τον PN532, όπου όταν διαβαστεί το σωστό μήνυμα θα ξεκλειδώνεται μια κλειδαριά. Θα γίνει μια γενική αναφορά στην δομή, δηλαδή στο τι περιέχει μια βιβλιοθήκη, την οποία μπορούμε να χρησιμοποιήσουμε στα προγράμματα μας στο Arduino IDE. Επιπλέον θα δούμε τον σχεδιασμό του κυκλώματος, το κόστος των υλικών που χρησιμοποιήθηκαν. Επίσης, θα περιγραφεί η λειτουργία του συστήματος και θα γίνει επεξήγηση του κώδικα που γράφτηκε στο Arduino IDE.

4.1 Δομή βιβλιοθήκης Arduino

Παρακάτω σε αυτήν την ενότητα, θα γίνει μια γενική αναφορά στην δομή μιας βιβλιοθήκης Arduino. Η γλώσσα που προγραμματίζεται ο Arduino είναι η C++, η οποία είναι μια επέκταση της γλώσσας C, με την μονή διαφορά που έχουν μεταξύ τους, να είναι ότι στην C++ έχουμε τον αντικειμενοστραφή προγραμματισμό (object orientation). Διαθέτοντας την δυνατότητα αυτή η C++, μας δίνει την ευκαιρία να συγγράψουμε διάφορες βιβλιοθήκες. Ο αντικειμενοστραφής προγραμματισμός, χρησιμοποιεί τις κλάσεις, οι οποίες αποτελούνται από τις μεταβλητές μέλη και από τις μεθόδους. Οι μέθοδοι είναι συναρτήσεις που υπάρχουν μέσα στην κλάση. Οι συναρτήσεις αυτές μπορεί να είναι δημόσιες (public) ή ιδιωτικές (private). Εάν είναι δημόσιες, μπορούν να τις καλέσουν και οι υπόλοιπες κλάσεις που βρίσκονται μέσα στην βιβλιοθήκη, ενώ εάν είναι ιδιωτικές, μπορούν να τις καλέσουν μόνο οι συναρτήσεις που βρίσκονται εντός αυτής της κλάσης. Κάθε κλάση αποτελείται από 2 αρχεία, το αρχείο κεφαλίδας (Header file), το οποίο έχει κατάληξη .h και το αρχείο υλοποίησης (implementation file), το οποίο έχει κατάληξη .cpp. Για παράδειγμα, το

NdefMessage.h είναι το αρχείο κεφαλίδας της κλάσης NdefMessage και επίσης διαθέτει και το αρχείο υλοποίησης .cpp(Σχήμα 4.1).

Όνομα	Ημερομηνία τροποποι...	Τύπος	Μέγεθος
examples	13/9/2018 3:10 πμ	Φάκελος αρχείων	
tests	13/9/2018 3:10 πμ	Φάκελος αρχείων	
Due	13/9/2018 3:10 πμ	Header file	1 KB
keywords	13/9/2018 3:10 πμ	Έγγραφο κειμένου	2 KB
LICENSE	13/9/2018 3:10 πμ	Έγγραφο κειμένου	2 KB
MifareClassic	13/9/2018 3:10 πμ	C++ source file	14 KB
MifareClassic	13/9/2018 3:10 πμ	Header file	1 KB
MifareUltralight	13/9/2018 3:10 πμ	C++ source file	8 KB
MifareUltralight	13/9/2018 3:10 πμ	Header file	1 KB
Ndef	13/9/2018 3:10 πμ	C++ source file	2 KB
Ndef	13/9/2018 3:10 πμ	Header file	1 KB
NdefMessage	13/9/2018 3:10 πμ	C++ source file	7 KB
NdefMessage	13/9/2018 3:10 πμ	Header file	2 KB
NdefRecord	13/9/2018 3:10 πμ	C++ source file	8 KB
NdefRecord	13/9/2018 3:10 πμ	Header file	2 KB
NfcAdapter	13/9/2018 3:10 πμ	C++ source file	5 KB
NfcAdapter	13/9/2018 3:10 πμ	Header file	2 KB
NfcDriver	13/9/2018 3:10 πμ	Header file	1 KB
NfcTag	13/9/2018 3:10 πμ	C++ source file	3 KB
NfcTag	13/9/2018 3:10 πμ	Header file	2 KB
README.md	13/9/2018 3:10 πμ	Αρχείο MD	6 KB

Σχήμα 4.1 Αρχείο κεφαλίδας(.h) και υλοποίησης(.cpp) της κλάσης NdefMessage.

Το αρχείο κεφαλίδας(.h) λέει στο Sketch, τι πρέπει να ξέρει για να μπορέσει να χρησιμοποιήσει την συγκεκριμένη βιβλιοθήκη. Γενικά το αρχείο .cpp, περιέχει τον κώδικα για την κάθε κλάση και υλοποιεί τις λειτουργίες που οριστήκαν στον Header. Ενώ το αρχείο .h, μας δείχνει τι έχει μέσα η συγκεκριμένη κλάση, δηλαδή από τι συναρτήσεις αποτελείται, και είναι αυτό το οποίο καλούμε στο Arduino IDE, και με αυτό τον τρόπο εντάσσουμε στο πρόγραμμα μας, τον κώδικα που έχει γράψει για την συγκεκριμένη βιβλιοθήκη. Για παράδειγμα, εντός του αρχείο κεφαλίδας της κλάσης NdefMessage, η πρώτη μέθοδος(Σχήμα 4.2) είναι η NdefMessage(void), παρατηρούμε ότι έχει το ίδιο όνομα με την κλάση, η μέθοδος αυτή ονομάζεται constructor και μπορούμε να την χρησιμοποιήσουμε, ώστε να δημιουργήσουμε ένα αντικείμενο μέσα στο πρόγραμμα μας(sketch) όπως θα δούμε παρακάτω.

```
Start here  NdefMessage.cpp  NdefMessage.h  X
4  #include <Ndef.h>
5  #include <NdefRecord.h>
6
7  #define MAX_NDEF_RECORDS 4
8
9  class NdefMessage
10 {
11     public:
12         NdefMessage(void); ←
13         NdefMessage(const byte *data, const int numBytes);
14         NdefMessage(const NdefMessage& rhs);
15         ~NdefMessage();
16         NdefMessage& operator=(const NdefMessage& rhs);
17
18         int getEncodedSize(); // need so we can pass array to encode
19         void encode(byte *data);
20
21         boolean addRecord(NdefRecord& record);
22         void addMimeMediaRecord(String mimeType, String payload);
23         void addMimeMediaRecord(String mimeType, byte *payload, int payloadLength);
24         void addTextRecord(String text);
25         void addTextRecord(String text, String encoding);
26         void addUriRecord(String uri);
27         void addEmptyRecord();
```

Σχήμα 4.2 Μέθοδος δόμησης στην κλάση NdefMessage.

Ένα ακόμη αρχείο που υπάρχει σε μια βιβλιοθήκη είναι το keywords.txt(Σχήμα 4.3). Είναι ένας πίνακας με 2 στήλες. Η αριστερά στήλη είναι η λέξη κλειδί και η δεξιά στήλη μας δείχνει τον τύπο της λέξης. Τα ονόματα των κλάσεων πρέπει να είναι KEYWORD1, ενώ των μεθόδων KEYWORD2. Το αρχείο αυτό χρησιμοποιείται για να χρωματίζει τις λέξεις στα προγράμματα. Επιπλέον, έχουμε το αρχείο examples(Σχήμα 4.3), ο φάκελος αυτός περιέχει τα παραδείγματα της συγκεκριμένης βιβλιοθήκης, κάθε παράδειγμα είναι και ένα Sketch στο Arduino IDE, γενικά όσο πιο σύνθετη είναι μια βιβλιοθήκη τόσο και περισσότερα Sketch διαθέτει [18].

Όνομα	Ημερομηνία τροποποι...	Τύπος	Μέγεθος
examples	13/9/2018 3:10 πμ	Φάκελος αρχείων	
tests	13/9/2018 3:10 πμ	Φάκελος αρχείων	
Due	13/9/2018 3:10 πμ	Header file	1 KB
keywords	13/9/2018 3:10 πμ	Έγγραφο κειμένου	2 KB
LICENSE	13/9/2018 3:10 πμ	Έγγραφο κειμένου	2 KB
MifareClassic	13/9/2018 3:10 πμ	C++ source file	14 KB
MifareClassic	13/9/2018 3:10 πμ	Header file	1 KB
MifareUltralight	13/9/2018 3:10 πμ	C++ source file	8 KB
MifareUltralight	13/9/2018 3:10 πμ	Header file	1 KB
Ndef	13/9/2018 3:10 πμ	C++ source file	2 KB
Ndef	13/9/2018 3:10 πμ	Header file	1 KB
NdefMessage	13/9/2018 3:10 πμ	C++ source file	7 KB
NdefMessage	13/9/2018 3:10 πμ	Header file	2 KB
NdefRecord	13/9/2018 3:10 πμ	C++ source file	8 KB
NdefRecord	13/9/2018 3:10 πμ	Header file	2 KB
NfcAdapter	13/9/2018 3:10 πμ	C++ source file	5 KB
NfcAdapter	13/9/2018 3:10 πμ	Header file	2 KB
NfcDriver	13/9/2018 3:10 πμ	Header file	1 KB
NfcTag	13/9/2018 3:10 πμ	C++ source file	3 KB
NfcTag	13/9/2018 3:10 πμ	Header file	2 KB
README.md	13/9/2018 3:10 πμ	Αρχείο MD	6 KB

Σχήμα 4.3 Δομή της βιβλιοθήκης NDEF.

4.2 Σχεδίαση του κυκλώματος

4.2.1 Περιγραφή των υλικών

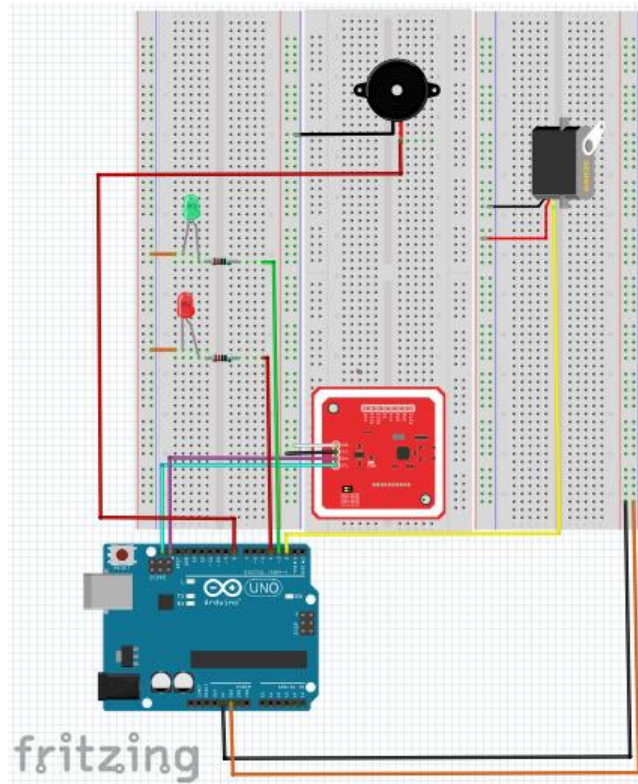
Τα υλικά που χρησιμοποιήθηκαν για το συγκεκριμένο project, ήταν η NFC συσκευή ανάγνωσης PN532, για τον οποίο γίνεται αναφορά στο κεφάλαιο 2(ενότητα 2.4.1). Επίσης, χρησιμοποιήθηκαν 2 LED(πράσινο και κόκκινο), αντιστάσεις 330 Ω οι οποίες συνδέθηκαν στις ανόδους των 2 LED, ένα buzzer για ηχητική ειδοποίηση και ένας σερβοκινητήρας ο SG90, ο οποίος όπως και στο RFID project παίζει τον ρολό της κλειδαριάς, για όλα τα παραπάνω υλικά έγινε αναφορά στο κεφάλαιο 3(ενότητα 3.1.1) και για αυτό τον λόγο δεν θα επαναληφθεί η αναφορά στα υλικά αυτά σε αυτό το κεφάλαιο.

4.2.2 Κόστος υλικών

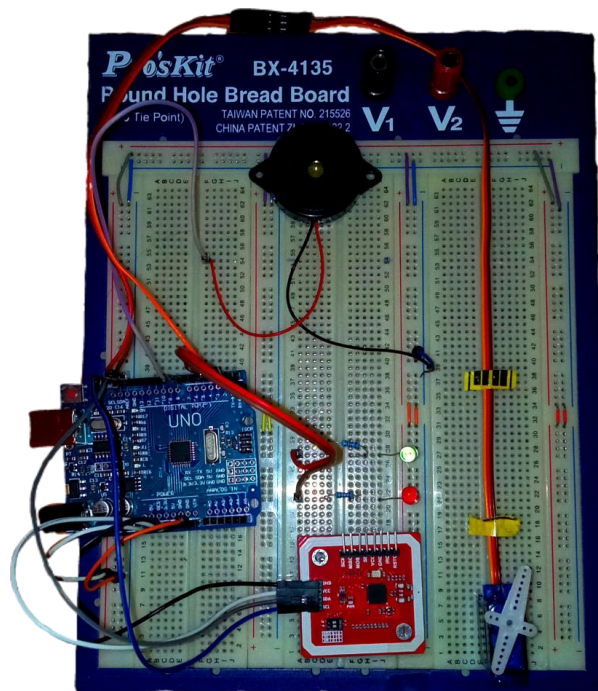
	Υλικά	Κόστος
	Arduino UNO (κλώνος)	7 €
	PN532	20 €
	LED-Αντιστάσεις- Καλώδια	3 €
	Buzzer	2€
	Servomotor	4€
Σύνολο		36 €

Σχήμα 4.4 Πίνακας υλικών και το κόστος αυτών για την NFC εφαρμογή.

4.3 Λειτουργία NFC συστήματος

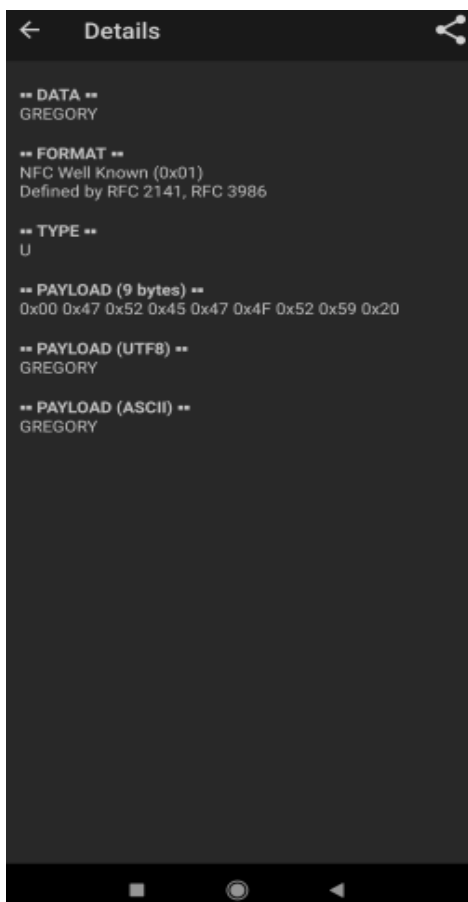


Σχήμα 4.5 Η σχεδίαση του κυκλώματος στο Fritzing.

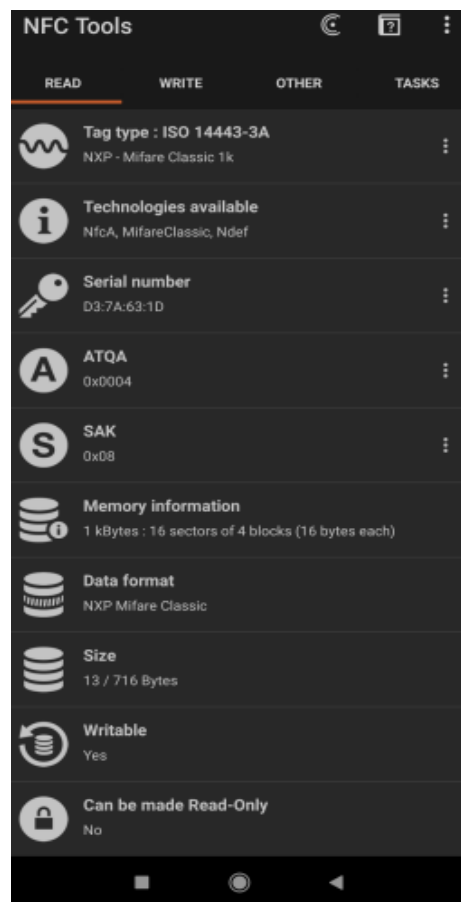


Σχήμα 4.6 Η υλοποίηση του κυκλώματος στο Breadboard.

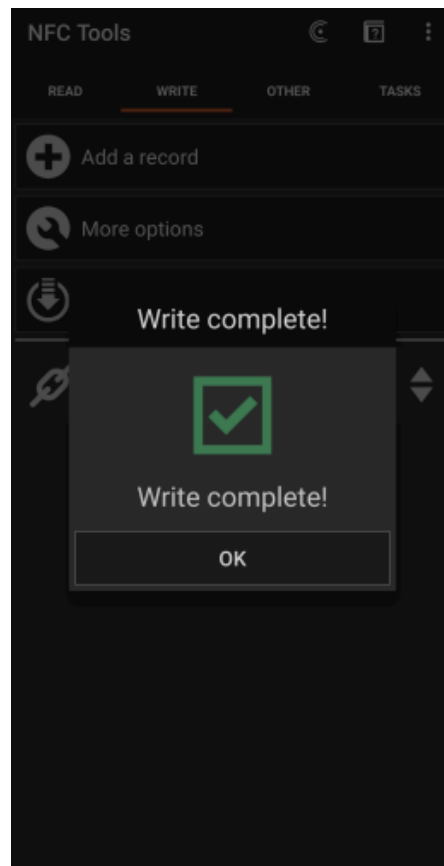
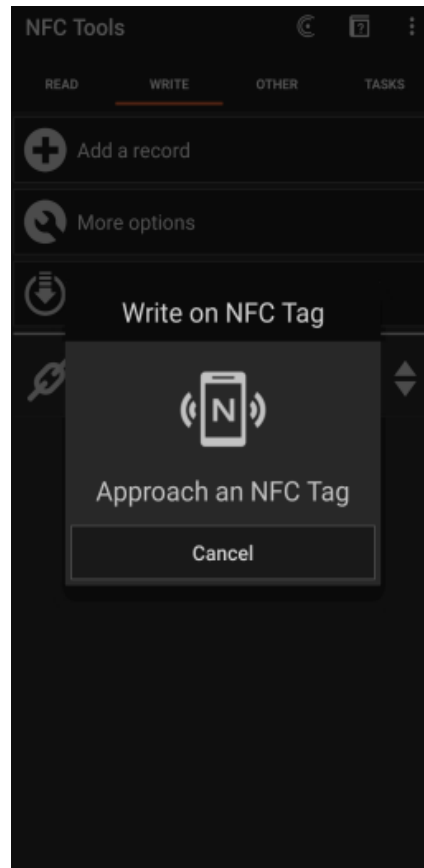
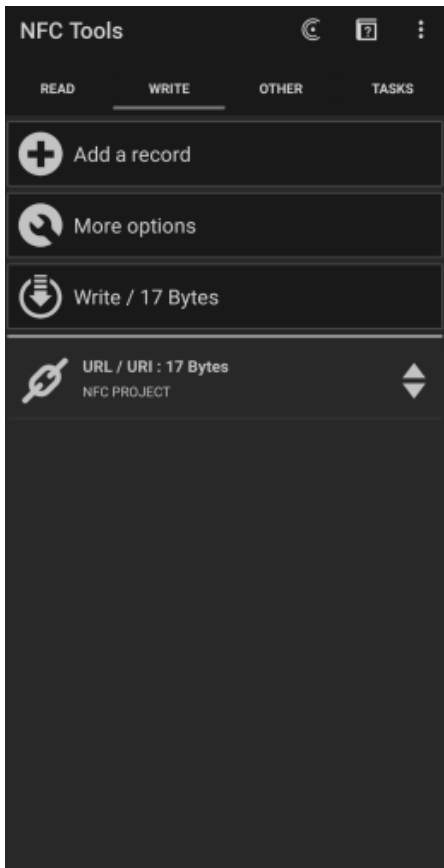
Αφού έχουμε έτοιμο το κύκλωμα και συνδέσουμε το Arduino με τον υπολογιστή, αυτό που χρειάζεται επιπλέον να κάνουμε, είναι να ανοίξουμε την εφαρμογή NFC tools, την οποία θα πρέπει να έχουμε εγκαταστήσει στο κινητό μας τηλέφωνο. Η εφαρμογή αυτή είναι της εταιρίας WAKDEV και μέσω αυτής μπορούμε να διαβάσουμε κάρτες ή ετικέτες, ώστε να βλέπουμε το ID, τον τύπο της κάρτας που διαβάζεται ή κάποιο NDEF μήνυμα που έχει. Επίσης, μπορεί και μας εμφανίζει και λεπτομερές για το συγκεκριμένο μήνυμα(Σχήμα 4.7), αλλά μπορούμε να δούμε και άλλες πληροφορίες που πιθανόν μας ενδιαφέρουν από μια συγκεκριμένη κάρτα(Σχήμα 4.8).Η συγκεκριμένη εφαρμογή, μας δίνει πολλές δυνατότητες, όμως για αυτό το project, χρησιμοποιήθηκε ώστε να γράφονται NDEF μηνύματα στις ετικέτες(Σχήμα 4.9) ,τα μηνύματα αυτά μπορεί να έχουν οποιόν τύπο επιθυμούμε, όμως στην εργασία αυτή, τα NDEF μηνύματα που χρησιμοποιήθηκαν, ήταν τύπου URI.



Σχήμα 4.7 Πληροφορίες για το NDEF μήνυμα που έχει εγγραφεί σε μια κάρτα.



Σχήμα 4.8 Διάβασμα μια κάρτας με το κινητό τηλέφωνο μέσω της εφαρμογής NFC Tools.



Σχήμα 4.9 Διαδικασία εγγραφής NDEF μηνύματος σε μια κάρτα.

Αφού γράψουμε το μήνυμα που θέλουμε σε μια ετικέτα μέσω της εφαρμογής, σειρά έχει η ετικέτα αυτή να διαβαστεί από την NFC συσκευή ανάγνωσης. Μόλις διαβαστεί η ετικέτα(Σχήμα 4.10),θα δούμε αμέσως στην σειριακή οθόνη, το TNF,τον τύπο του μηνύματος, καθώς και το payload, δηλαδή το περιεχόμενο του μηνύματος αυτού. Εάν το NDEF μήνυμα που έχει, ταυτίζεται με αυτό που έχει γραφτεί στο Arduino IDE,τότε θα ηχήσει το buzzer 1 φορά, θα ανάψει το πράσινο LED και θα ξεκλειδωθεί η κλειδαριά(Σερβοκινητήρας), θα παραμείνει ανοιχτή για 5 δευτερόλεπτα και έπειτα θα ξανά κλειδώσει. Όλες τις ενέργειες αυτές, θα τις δούμε στην σειριακή οθόνη σαν μηνύματα(Σχήμα 4.10). Στην περίπτωση όμως που το NDEF μήνυμα που διαβάστηκε από την συσκευή ανάγνωσης, δεν είναι ενσωματωμένο στον κώδικα τότε δεν θα ξεκλειδωθεί η κλειδαριά, θα ανάψει το κόκκινο LED, θα ηχήσει το buzzer 3 φορές και θα δούμε το μήνυμα “TRY AGAIN”.

```
Mifare Classic
UID: D3 7A 63 1D

We have an NDEF Message with : 1 NDEF Record
NDEF Record: 1
  TNF: 1
  Type: U
Payload (as String): GREGORY
WELCOME!!!
DOOR IS OPEN!!!
5
4
3
2
1
0
DOOR IS CLOSE!!!
```

```
Mifare Classic
UID: A9 48 D8 A2

We have an NDEF Message with : 1 NDEF Record
NDEF Record: 1
  TNF: 1
  Type: U
Payload (as String): NFC PROJECT
TRY AGAIN!!!
```

Σχήμα 4.10 Η 1^η περίπτωση(πάνω) είναι όταν το NDEF μήνυμα είναι έγκυρο και η 2^η περίπτωση(κάτω), όταν είναι άκυρο.

Στο παραπάνω σχήμα(**Σχήμα 4.10**), παρατηρούμε στο TNF τον αριθμό 1, και στον τύπο(type) το γράμμα U, αυτό σημαίνει ότι και στις 2 περιπτώσεις, ότι διαβάστηκε ένα μήνυμα το οποίο έχει μια εγγραφή(record), η οποία ανήκει στην κατηγορία του TNF 0x01 Well-known και είναι τύπου URI. Το payload στην πρώτη περίπτωση είναι το όνομα “GREGORY”, το οποίο εμφανίζεται ως συμβολοσειρά, και στην δεύτερη το όνομα “NFC PROJECT”.

Μια ακόμη ενδιαφέρον περίπτωση(Σχήμα 4.11), είναι όταν στην ετικέτα έχουμε γράψει NDEF μήνυμα με παραπάνω από μια εγγραφή(record). Έγινε δοκιμή σε μια ετικέτα και γράφτηκε 1 NDEF μήνυμα με 3 Records. Τα 2 ήταν τύπου Text και το ένα τύπου URI. Ο κώδικας ο οποίος γράφτηκε και θα αναλυθεί παρακάτω είναι έτσι ώστε, να δέχεται κάποιο NDEF μήνυμα με record τύπου URI. Παρατηρήθηκε ότι, ελέγχονται όλα τα records ένα προς ένα, και σε οποιαδήποτε θέση και εάν βρίσκεται η εγγραφή(record) URI που γράφτηκε, δηλαδή είτε είναι πρώτη είτε τρίτη, θα δούμε ότι μόλις γίνει ο έλεγχος, θα ξεκλειδώσει η κλειδαριά, οπότε δεν έχει σημασία η θέση που έχει το κάθε record μέσα σε ένα NDEF μήνυμα.

```
COM4

Mifare Classic
UID: 89 7D A5 03

We have an NDEF Message with : 3 NDEF Record
NDEF Record: 1
  TNF: 1
  Type: T
  Payload (as String): GenRFID project
  TRY AGAIN!!!
-----
NDEF Record: 2
  TNF: 1
  Type: U
  Payload (as String): GREGORY
  WELCOME!!!
  DOOR IS OPEN!!!
  5
  4
  3
  2
  1
  0
  DOOR IS CLOSE!!!
-----
NDEF Record: 3
  TNF: 1
  Type: T
  Payload (as String): GenNFC project
  TRY AGAIN!!!
```

Σχήμα 4.11 NDEF μήνυμα με 3 records.

4.4 ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ NFC ΣΥΣΤΗΜΑΤΟΣ

- **ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ**

- 1) Αρχικά δηλώθηκαν οι βιβλιοθήκες οι οποίες είναι απαραίτητες για την λειτουργία της NFC συσκευής ανάγνωσης PN532 σε λειτουργία I2C(οι 3 πρώτες), έχουμε μια βιβλιοθήκη ώστε να διαβάσουμε NDEF μηνύματα[19] και άλλη μια για τον σερβοκινητήρα.

```
( #include <Wire.h>
#include <PN532_I2C.h>
#include <PN532.h>
#include <NfcAdapter.h>
#include <Servo.h> )
```

- Εδώ ως σχόλια αναφέρονται οι 4 κλάσεις της βιβλιοθήκης NDEF που χρησιμοποιήθηκαν στον κώδικα. Η βιβλιοθήκη αυτή έχει 7 κλάσεις. Παρακάτω, κάθε αντικείμενο που δημιουργείται και γενικά ότι περιέχεται σε μια από αυτές τις κλάσεις, σαν σχόλιο θα αναφέρεται δίπλα ο αριθμός, για να γνωρίζουμε σε ποια κλάση ανήκει κάθε δεδομένο στον κώδικα.

```
//NfcAdapter είναι η κλάση (1)
//NfcTag είναι η κλάση (2)
//NdefMessage είναι η κλάση (3)
//To NdefRecord είναι η κλάση (4)
```

- 2) Παρακάτω ορίζονται σε ποιες ακίδες του Arduino θα συνδεθούν τα led,buzzer και ο σερβοκινητήρας.

```
(
PN532_I2C pn532_i2c(Wire);
NfcAdapter nfc = NfcAdapter(pn532_i2c);
```

```
//LED
#define led_green 3
```

```

#define led_red 4
//BUZZER
int buzzer=8;

//SERVO
Servo servo;
#define servo_pin 2 )

```

3) Στην συνάρτηση **setup**, αρχικά ενεργοποιούμε την σειριακή επικοινωνία στα 9600 bps, γράφουμε ένα μήνυμα το οποίο θα εμφανιστεί στην σειριακή οθόνη ώστε να καταλάβουμε ότι το πρόγραμμα έχει εγκατασταθεί στην μνήμη του Arduino. Επίσης, ορίζουμε τα pins του Arduino τα οποία χρειάστηκαν για το project, ως εξόδους. Για περισσότερες λεπτομέρειες όσον αφορά τα pins τα οποία χρησιμοποιήθηκαν, υπάρχουν στο κεφάλαιο 3(ενότητα 3.3.2), όπου εκεί αναφέρονται κάποιες επιπλέον πληροφορίες.

```

(
void setup()
{

Serial.begin(9600);
Serial.println("Scan a NFC tag");
nfc.begin();//(1)

//SERVO
servo.attach(servo_pin);
servo.write(0);

//ΕΞΟΔΟΙ LED
pinMode(led_red, OUTPUT);
pinMode(led_green, OUTPUT);

//ΕΞΟΔΟΣ BUZZER
pinMode(buzzer, OUTPUT);

```



```
}
```

```
void loop()
```

```
{
```

```
    if (nfc.tagPresent()) //Γίνεται έλεγχος, εάν υπάρχει κάποια ετικέτα για να  
    διαβαστεί. (1)
```

```
    {
```

```
    )
```

- 4) Αφού έχει γίνει έλεγχος και υπάρχει μια ετικέτα για διάβασμα, τότε δημιουργούμε ένα αντικείμενο της κλάσης NfcTag, με όνομα tag, και εκεί καταχωρούνται τα δεδομένα τα οποία διαβάστηκαν από την ετικέτα αυτή.

```
( NfcTag tag = nfc.read(); // (2) )
```

- Αφού διαβαστεί η ετικέτα θα δούμε τον τύπο και το UID της. Όπως βλέπουμε γίνεται χρήση του αντικειμένου tag και βάζουμε τελεία(.), ώστε να έχουμε πρόσβαση στις συναρτήσεις/μέλη της συγκεκριμένης κλάσης, δηλαδή με αυτό τον τρόπο καλούμε το συγκεκριμένο κομμάτι του κώδικα από την κλάση.

```
( Serial.println(tag.getTagType()); //TYPE (2)
```

```
Serial.print("UID: ");
```

```
Serial.println(tag.getUidString()); //UID (2)
```

```
Serial.println(); )
```

5) Γίνεται έλεγχος εάν υπάρχει NDEF μήνυμα στην ετικέτα.

```
( if (tag.hasNdefMessage()) // (2) )  
  {
```

- Το **message** είναι το αντικείμενο της κλάσης NdefMessage, μπορούμε να βάλουμε οποίο όνομα θέλουμε, όταν θέλουμε να χρησιμοποιήσουμε και να καλέσουμε αυτό το αντικείμενο βάζουμε τελεία.

```
( NdefMessage message = tag.getNdefMessage(); //(2)
```

```
  Serial.print("We have an NDEF Message with : ");
```

```
  Serial.print(message.getRecordCount()); //(3) Εδώ θα δούμε τον αριθμό  
των records που έχει η κάρτα που διαβάστηκε.
```

```
  Serial.println(" NDEF Record "); )
```

- Γίνεται δήλωση της μεταβλητής recordCount, όπου καταχωρείται η τιμή των records.

```
( int recordCount;  
  recordCount= message.getRecordCount(); (3) )
```

6) Η **for** αυτή δημιουργήθηκε, για την περίπτωση που έχουμε περισσότερα από ένα record στο μήνυμα και με αυτό τον τρόπο να μπορούμε να βλέπουμε τα χαρακτηριστικά για το κάθε record.

```
( for (int i = 0; i < recordCount; i++) //(3)
{
    Serial.println("NDEF Record: "); //Ξανά βλέπουμε τον αριθμό των
records.

    Serial.println(i+1); // Με αυτό τον τρόπο βλέπουμε ποιο record
διαβάζεται. )
```

7) Δημιουργήθηκε ένα αντικείμενο της κλάσης Ndefrecord, όπου εκεί για κάθε record θα καταχωρείτε το TNF, ο τύπος και το payload του μηνύματος.

```
( NdefRecord record = message.getRecord(i); //(3,4)

    Serial.println(" TNF: ");
    Serial.println(record.getTnf()); //(4)

    Serial.println(" Type: ");
    Serial.println(record.getType()); //(4)

    int payloadLength;
    payloadLength = record.getPayloadLength(); //(4) Δήλωση της
μεταβλητής payloadLength, όπου καταχωρείται το μέγεθος του payload.

    byte payload[payloadLength]; //(4)
    record.getPayload(payload); //(4) )
```

8) Παρακάτω, κάθε byte που έχει το payload του μηνύματος, θα το δούμε ως συμβολοσειρά.

```
( String payloadAsString = "";

    for (int c = 0; c < payloadLength; c++) //(4)
    {
        payloadAsString += (char)payload[c];
```

```

}
Serial.print("Payload (as String): ");
Serial.println(payloadAsString);

```

- Δημιουργήθηκε μια συμβολοσειρά με όνομα `strin`, και έχει ως περιεχόμενο το όνομα "GREGORY", η τελεία που υπάρχει είναι διότι, όταν γράφεται από την εφαρμογή στην ετικέτα ένα μήνυμα τύπου URI έχει μπροστά μια τελεία η οποία δεν φαίνεται.

```

( String strin=" .GREGORY"; )

```

- 9) Γίνεται η σύγκριση του μηνύματος που διαβάστηκε και συγκρίνεται με την συμβολοσειρά `strin`, εάν υπάρχει ισότητα τότε θα ξεκλειδώσει η κλειδαριά, διαφορετικά θα πρέπει να διαβαστεί η σωστή ετικέτα με το σωστό NDEF μήνυμα.

```

( if( payloadAsString.equalsIgnoreCase(strin))
{
  led(led_green,3,150);
  buzz(1,800,1500);

  Serial.println(" WELCOME!!! ");
  servo.write(180); //Ξεκλειδώνει η κλειδαριά.
  Serial.println("DOOR IS OPEN!!!");
  delay(1000);

  for(int i=5; i>=0; i--) //Περιμένει 5s
  {
    Serial.println(i);
    delay(1000);
  }

  servo.write(0); //Κλειδώνει η κλειδαριά.
  delay(1000);
  Serial.println("DOOR IS CLOSE!!!");

}

```

```

else
{
  led(led_red,3,250);
  buzz(3,500,250);
  delay(1000);
  Serial.println(" TRY AGAIN!!! ");
}

} // ΤΕΛΟΣ For recordcount

} // ΤΕΛΟΣ tag.hasNDEFMessage

} // ΤΕΛΟΣ nfc.tag present

delay(3000);

} // ΤΕΛΟΣ void loop )

```

10) Παρακάτω είναι και οι συναρτήσεις που χρησιμοποιήθηκαν, των οποίων είχε γίνει χρήση και στο RFID project(ενότητα 3.3.2), για αυτό τον λόγο δεν θα ξανά γίνει αναφορά σε αυτές στο κεφάλαιο αυτό.

```

( //ΣΥΝΑΡΤΗΣΗ LED
void led(int ledpin,int number,int del)
{
  for(int i=0; i<number; i++)
  {

    digitalWrite(ledpin, HIGH);
    delay(del);
    digitalWrite(ledpin, LOW);
    delay(del);
  }
}

```

```
    }  
  }  
  
  //ΣΥΝΑΡΤΗΣΗ BUZZER  
  void buzz(int number,int freq, int del)  
  {  
    for(int i=0; i<number; i++)  
    {  
      tone(buzzer, freq);  
      delay(del);  
      noTone(buzzer);  
      delay(del);  
    }  
  } )
```

5 Σύγκριση των 2 συστημάτων

ΣΚΟΠΟΣ

Στο κεφάλαιο 5, θα γίνει αναφορά στα 2 συστήματα, RFID και NFC, τα οποία αναλύθηκαν παραπάνω. Θα αναφερθούν τα μειονεκτήματα και τα πλεονεκτήματα του κάθε συστήματος και τα συμπεράσματα που προκύπτουν μετά από την ενασχόληση με αυτά τα συστήματα.

5.1 Πλεονεκτήματα-μειονεκτήματα των συστημάτων

	RFID project	NFC project
Διάβασμα UID MIFARE καρτών και ετικετών.	ΝΑΙ	ΝΑΙ
Διάβασμα NDEF μηνυμάτων μέσω καρτών.	ΟΧΙ	ΝΑΙ
Διάβασμα NDEF μηνυμάτων μέσω κινητού τηλεφώνου.	ΟΧΙ	ΙΣΩΣ

Σχήμα 5.1 Πίνακας πλεονεκτημάτων και μειονεκτημάτων των 2 συστημάτων.

5.2 Συμπεράσματα

Στα κεφάλαια 3 και 4 αναλύθηκαν 2 συστήματα, ένα RFID και ένα NFC αντίστοιχα. Κάθε project δημιουργήθηκε για τον δικό του σκοπό. Με το RFID project, μπορούμε να διαβάζουμε κάρτες και ετικέτες MIFARE, δηλαδή μπορούμε να διαβάζουμε το ID αυτών των καρτών, κάτι το οποίο μπορούμε να κάνουμε και με το NFC project, όποτε συμπεραίνουμε ότι είναι μια λειτουργία που την διαθέτουν και τα 2. Στην περίπτωση όμως, που θέλουμε με τις κάρτες ή τις ετικέτες που διαθέτουμε, να στείλουμε ένα NDEF μήνυμα ώστε να το διαβάσει η εκάστοτε συσκευή ανάγνωσης, δεν μπορούμε να χρησιμοποιήσουμε το RFID και συγκριμένα την συσκευή ανάγνωσης RC522. Έγινε δοκιμή με μια βιβλιοθήκη NDEF από το github(<https://github.com/aroller/NDEF-MFRC522>), η οποία είναι βασισμένη και γραμμένη για αυτήν την συσκευή ανάγνωσης, όμως παρά τις προσπάθειες που έγιναν δεν επετεύχθη ο στόχος, ο οποίος ήταν, να σταλθεί ένα NDEF μήνυμα και να διαβαστεί από τον RC522. Αυτός ο στόχος όμως με τον PN532, επετεύχθη, διότι σαν συσκευή ανάγνωσης έχει καλύτερο Hardware και με τις βιβλιοθήκες που έχουν γραφτεί για αυτήν την συσκευή ανάγνωσης, την καθιστούν ικανή να διαβάζει NDEF μηνύματα από κάρτες και ετικέτες. Και για τα 2 συστήματα χρησιμοποιήθηκαν τα ίδια υλικά, εξαίρεση αποτελεί όμως το RFID, όπου εκεί έγινε χρήση και της πλακέτας HC-05, η οποία μας δίνει την δυνατότητα να έχουμε επικοινωνία ανάμεσα στο κινητό τηλέφωνο και στο Arduino μέσω Bluetooth, και με αυτό τον τρόπο να έχουμε την δυνατότητα να ελέγχουμε το κύκλωμα μέσω της εφαρμογής. Στο NFC project, θα μπορούσε να είχε γίνει το ίδιο, παρ' όλ' αυτά δεν έγινε και έμεινε η εργασία αυτή στις δόκιμες που αφορούσαν τα NDEF μηνύματα. Όμως είναι ένα κύκλωμα, το οποίο έχει την δυνατότητα να υποστηρίξει μια προσθήκη όπως αυτή με το HC-05. Άρα ως συμπέρασμα θα λέγαμε ότι, και τα 2 συστήματα είναι εξίσου αξιόπιστα, απλά το NFC project μας δίνει την ευκαιρία για να δημιουργήσουμε περισσότερες εφαρμογές. Ανάλογα βεβαίως με την εφαρμογή την οποία θέλουμε να κάνουμε, θα πρέπει να επιλέξουμε και το κατάλληλο σύστημα, αν για παράδειγμα, θέλουμε να δημιουργήσουμε ένα project, όπου θα έχουμε ένα σύστημα και με το διάβασμα μιας κάρτας να εισερχόμαστε σε ένα προσωπικό μας χώρο, τότε το RFID σύστημα θα ήταν το ιδανικό, αφού είναι και ποιο οικονομικό σε σχέση με το NFC project, το οποίο είναι περίπου 10 € ακριβότερο. Όμως εάν θέλουμε για παράδειγμα, να διαβάζουμε

NDEF μηνύματα μέσω καρτών και με αυτόν τον τρόπο να ξεκλειδώνει μια πόρτα ενός προσωπικού χώρου μας, τότε θα πρέπει να χρησιμοποιήσουμε το NFC project.

6 Συμπεράσματα και προτάσεις λύσης

ΣΚΟΠΟΣ

Στο κεφάλαιο 6, θα γίνει αναφορά στην προσπάθεια που έγινε, ώστε να μπορέσουμε να διαβάσουμε ένα NDEF μήνυμα από το NFC του κινητού, και οι πιθανές λύσεις που προτείνονται για την υλοποίηση του στόχου αυτού.

Μια αρκετά ενδιαφέρουσα εργασία θα ήταν, να μπορούμε μέσω του κινητού μας να στέλνουμε ένα NDEF μήνυμα, και με αυτόν τον τρόπο να ξεκλειδωθεί μια πόρτα. Όπως για παράδειγμα, πληρώνουμε με το κινητό μας και το χρησιμοποιούμε ως ένα κινητό πορτοφόλι, να μπορούμε με αυτό να ανοίγουμε διαφορές πόρτες και να μην χρειάζεται να κουβαλάμε μαζί μας πολλά κλειδιά. Αυτή η προσπάθεια έγινε σε αυτήν την πτυχιακή εργασία, όμως τελικά ήταν ένας στόχος, ο οποίος τελικά δεν πραγματοποιήθηκε. Αρχικά έγινε δοκιμή στο RFID project, όπου έγινε προσπάθεια να διαβάσουμε το κινητό με αυτό τον συσκευή ανάγνωσης, το αποτέλεσμα ήταν ότι κάθε φορά που διάβαζε το chip του κινητού, βλέπαμε κάθε φορά διαφορετικό ID(Σχήμα 6.1), το οποίο δεν είχε καμία σχέση με το προηγούμενο ID που είχε διαβαστεί. Χρησιμοποιήθηκε μια βιβλιοθήκη NDEF από το github(<https://github.com/aroller/NDEF-MFRC522>), όμως ούτε με αυτή την προσθήκη άλλαξε κάτι. Έπειτα, δοκιμάστηκε να γίνει το κινητό κάρτα με διάφορες εφαρμογές από το Play store, με κανένα όμως αποτέλεσμα. Η επομένη δοκιμή έγινε στο NFC project, όπου ο συσκευή ανάγνωσης αυτός όπως αναφέρει το manual, έχει την δυνατότητα να διαβάσει το NFC του κινητού. Έγιναν προσπάθειες σε όλες τις σειριακές επικοινωνίες, SPI,I2C και UART, παρ' όλ' αυτά, είχαμε το ίδιο αποτέλεσμα με τον RC522, ότι δηλαδή είχαμε κάθε φορά διαφορετικό ID. Μέσα στο manual, αναφέρεται ότι με την χρήση του Arduino Mega ή Leonardo και με την εγκατάσταση των καταλλήλων βιβλιοθηκών που προτείνονται εκεί, μπορεί να δουλέψει η συσκευή

ανάγνωσης και να διαβάσει το NFC chip του κινητού, επειδή όμως είχαμε στην διάθεση μας, μόνο το Arduino UNO, δεν λειτούργησε με αυτό, κάτι το οποίο αναφέρει και το manual του PN532 V3. Δεν έγινε αγορά άλλης πλακέτας Arduino και οι δόκιμες σταμάτησαν σε αυτό το σημείο. Όμως με την προσθήκη κάποιου εκ των 2 πλακετών(Mega, Leonardo) και με την προσθήκη των βιβλιοθηκών, υπάρχουν αρκετές πιθανότητες να διαβαστεί το NFC του κινητού μας τηλεφώνου(**1^η λύση**). Μια άλλη λύση που προτείνεται και που φαίνεται και η πιο πιθανή και ικανή να λειτουργήσει, είναι η προσθήκη στο κύκλωμα μας της NFC συσκευής ανάγνωσης Seeed Studio V2.0(**Ενότητα 2.4.4**). Αυτή η συσκευή ανάγνωσης διαθέτει την λειτουργία P2P, η οποία μας επιτρέπει να επικοινωνήσουμε με κινητά τηλέφωνα και με την χρήση των κατάλληλων βιβλιοθηκών, οι όποιες υπάρχουν στο github για την συγκεκριμένη συσκευή ανάγνωσης, θα μπορούσαμε να στέλνουμε μηνύματα από το κινητό μας στην συσκευή ανάγνωσης, και με αυτόν τον τρόπο θα καταφέρουμε να δημιουργήσουμε πολύ ενδιαφέροντα projects(**2^η λύση**). Η λειτουργία P2P, αναφέρεται και στο manual του PN532 V3, όμως είναι η λειτουργία που μας δίνει την δυνατότητα: μια συσκευή ανάγνωσης PN532 V3 να μπορεί να επικοινωνεί με μια άλλη PN532 V3 συσκευή ανάγνωσης.

```
Scan a NFC tag

Tag is not NDEF formatted.
Mifare Classic
UID: 08 29 38 C5

Scan a NFC tag

Tag is not NDEF formatted.
Mifare Classic
UID: 08 29 2C FB

Scan a NFC tag

Tag is not NDEF formatted.
Mifare Classic
UID: 08 A0 3F F3
```

Σχήμα 6.1 Διάβασμα του NFC chip του κινητού, βλέποντας σε κάθε διάβασμα διαφορετικό ID. Από τις πειραματικές δοκιμές αυτής της διπλωματικής εργασίας.

7.Βιβλιογραφία

- [1] Igoe, T., Coleman, D., & Jepson, B. (2014). Beginning NFC: near field communication with Arduino, Android, and Phonegap.
- [2] NFC FORUM. (2022). NFC FORUM. Retrieved from <https://nfc-forum.org/learn/nfc-technology/>
- [3] Engineers, L. M. (2022). Last Minute Engineers. Ανάκτηση από <https://lastminuteengineers.com/how-rfid-works-rc522-arduino-tutorial/>
- [4] ada, I. (2022). Adafruit. Retrieved from <https://learn.adafruit.com/adafruit-pn532-rfid-nfc/about-nfc>
- [5] NXP. (2022). Retrieved from <https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf>
- [6] NXP. (2022). Retrieved from https://www.nxp.com/docs/en/nxp/datasheets/PN532_C1.pdf.
- [7] Balboa, M. A. (2022). github. Ανάκτηση από <https://github.com/miguelbalboa/rfid>
- [8] Wilson. (2022). elechouse. Ανάκτηση από https://www.elechouse.com/elechouse/images/product/PN532_module_V3/PN532_%20Manual_V3.pdf
- [9] github. (2022). Ανάκτηση από <https://github.com/elechouse/PN532>

- [10] Wilson. (2022). elechouse. Ανάκτηση από <https://www.elechouse.com/product/pn532-nfc-rfid-module-v4/>
- [11] Adafruit. (2022). Ανάκτηση από <https://www.adafruit.com/product/789#technical-details>
- [12] ada, I. (2022). adafruit. Ανάκτηση από <https://learn.adafruit.com/adafruit-pn532-rfid-nfc/overview>
- [13] Studio, S. (2022). Ανάκτηση από https://wiki.seedstudio.com/NFC_Shield_V2.0/
- [14] github. (2022). github. Ανάκτηση από <https://github.com/Seed-Studio/PN532/wiki/List-of-devices-supported>
- [15] Nedelkovski, D. (2022). howtomechatronics. Ανάκτηση από <https://howtomechatronics.com/tutorials/arduino/arduino-and-hc-05-bluetooth-module-tutorial/>
- [16] INVENTOR, M. A. (2022). MIT APP INVENTOR. Ανάκτηση από <https://appinventor.mit.edu/>
- [17] Νούσης, Β. (2019). Ανάκτηση από <http://3gym-igoum.thesp.sch.gr/images/Arduino/book.pdf>
- [18] Monk, S. (2016). Programming Arduino: getting started with sketches. McGraw-Hill Education.
- [19] Coleman, D. (2022). github. Ανάκτηση από <https://github.com/don/NDEF>

- [20] Adafruit. (2022). Ανάκτηση από <https://www.adafruit.com/product/364>
- [21] Seeed Studio. (2022). robotpark. Retrieved from <https://www.robotpark.com/NFC-Shield-V2-0-En>
- [22] Μαλαμίδης, Ι. (2022). vodafonegenerationnext.gr. Retrieved from <https://www.vodafonegenerationnext.gr/data/lessons/31/downloads/Internet%20of%20Things%202-Presentation.pdf>

