

**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ  
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ**

**ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ Τ.Ε**

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ 1487**

**ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΑΝΑΠΤΥΞΗ ΗΛΕΚΤΡΙΚΟΥ  
ΣΥΣΤΗΜΑΤΟΣ ΕΛΕΓΧΟΥ ΠΡΟΣΒΑΣΗΣ**

**ΓΙΑΝΝΟΠΟΥΛΟΣ ΑΛΕΞΑΝΔΡΟΣ (5004)**

**ΡΑΠΤΗΣ ΠΑΝΑΓΙΩΤΗΣ (5003)**

**ΕΠΟΠΤΕΥΩΝ ΚΑΘΗΓΗΤΗΣ: ΜΠΙΣΔΟΥΝΗΣ ΛΑΜΠΡΟΣ**

**ΠΑΤΡΑ 2015**

( κενή σελίδα )

## Πρόλογος

Η παρούσα πτυχιακή εργασία πραγματοποιήθηκε στο Τεχνολογικό Εκπαιδευτικό Ίδρυμα Δυτικής Ελλάδας, στο τμήμα Ηλεκτρολόγων Μηχανικών Τ.Ε.. Στόχος αυτής της πτυχιακής εργασίας είναι η σχεδίαση, υλοποίηση και κατασκευή συστήματος ελέγχου με τη χρήση μικροελεγκτή και βασική μέθοδο ελέγχου ταυτότητας τη χρήση αλφαριθμητικού πληκτρολογίου.

Σε αυτό το σημείο θα θέλαμε να ευχαριστήσουμε τον εποπτεύοντα καθηγητή κ. Μπισδούνη Λάμπρο, τον τεχνικό εργαστηρίων κ. Κατσαϊτή Ανδρέα και τον κ. Ντόκα Ιωάννη, για την αμέριστη βοήθεια και συμπαράσταση καθ' όλη τη διάρκεια της παρούσας πτυχιακής εργασίας.

Ιδιαίτερα θα θέλαμε να ευχαριστήσουμε τις οικογένειες μας για την στήριξη και συμπαράσταση τους καθ' όλη την διάρκεια των σπουδών μας.

( κενή σελίδα )

## Περίληψη

Η ανθρώπινη ζωή και τα ανθρώπινα αγαθά, είναι συνυφασμένα και απόλυτα διασυνδεδεμένα με την έννοια της ασφάλειας. Λέγοντας ασφάλεια, αναφερόμαστε στο βαθμό της αντίστασης και της προστασίας οποιουδήποτε πολύτιμου και ευαίσθητου πόρου ή συστήματος, από εξωτερικές απειλές.

Για τη θωράκιση της ασφάλειας, χρησιμοποιούνται μηχανισμοί οι οποίοι δημιουργούν επίπεδα προστασίας ανάμεσα στους προστατευόμενους πόρους ή συστήματα και τις εξωτερικές απειλές. Οι μηχανισμοί αυτοί ονομάζονται συστήματα ελέγχου, τα οποία χρησιμοποιούνται για την εξασφάλιση της ομαλής λειτουργίας των προστατευόμενων στοιχείων.

Σκοπός της παρούσας πτυχιακής εργασίας είναι ο σχεδιασμός και υλοποίηση ενός συστήματος ελέγχου με τη χρήση μικροελεγκτή και ηλεκτρολογίου. Συγκεκριμένα, η ανάπτυξη μιας κλειδαριάς, η ενεργοποίηση και η απενεργοποίηση της οποίας ελέγχεται μέσω της πλατφόρμας Arduino, θα παρουσιαστεί.

Το σύστημα ελέγχου που θα αναπτύξουμε σε αυτήν την εργασία θα επιτρέπει μόνο σε εξουσιοδοτημένα άτομα να έχουν πρόσβαση στο φυλασσόμενο χώρο, καθώς θα απαιτείται η εισαγωγή κωδικού ασφάλειας μέσω ηλεκτρολογίου ώστε να τους επιτραπεί η είσοδος.

( κενή σελίδα )

# Περιεχόμενα

1	Εισαγωγή.....	1
1.1	Δομή εργασίας.....	1
2	Ασφάλεια και Ηλεκτρονική Κλειδαριά.....	3
2.1	Φυσική ασφάλεια.....	3
2.1.1	Έλεγχος πρόσβασης.....	3
2.2	Ηλεκτρονική Κλειδαριά.....	6
2.2.1	Λειτουργία.....	6
2.2.2	Είδη ηλεκτρονικής κλειδαριάς.....	6
2.2.3	Μέθοδοι ελέγχου ταυτότητας.....	9
3	Arduino.....	13
3.1	Σύντομη Περιγραφή.....	13
3.2	Ιστορικά Στοιχεία.....	13
3.3	Υλικό.....	14
3.3.1	Διαθέσιμες πλακέτες.....	15
3.4	Λογισμικό.....	19
3.5	Ανάπτυξη.....	20
3.6	Arduino Mega 2560.....	21
3.6.1	Γενική Περιγραφή.....	21
3.6.2	Σχεδίαση.....	22
3.6.3	Τροφοδοσία.....	24
3.6.4	Μνήμη.....	25
3.6.5	Είσοδος και Έξοδος.....	25
3.6.6	Επικοινωνία.....	26
3.6.7	Προγραμματισμός.....	27
3.6.8	Αυτόματη Επαναφορά.....	27
3.6.9	Προστασία από Υπερένταση.....	28
3.6.10	Φυσικά Χαρακτηριστικά και Συμβατότητα.....	28
4	Παραγωγή Συστήματος.....	29
4.1	Προδιαγραφές Συστήματος.....	29
4.1.1	Τύποι Χρηστών.....	30
4.1.2	Περιορισμοί σχεδίασης και υλοποίησης.....	30

4.1.3	Λειτουργία συστήματος .....	30
4.1.4	Άλλες απαιτήσεις .....	35
4.1.5	Συγκέντρωση προδιαγραφών .....	36
4.2	Σχεδίαση .....	39
4.2.1	Διαγράμματα ροής.....	39
4.2.2	Σχεδίαση συνδέσεων .....	42
4.3	Υλοποίηση.....	44
4.3.1	Σταθερές.....	44
4.3.2	Μέθοδος setup.....	45
4.3.3	Μέθοδος loop().....	46
4.3.4	Μέθοδος keypadEvent .....	47
4.3.5	Μέθοδος clearEnteredPassword.....	48
4.3.6	Μέθοδος εμφάνισης μηνύματος.....	48
4.3.7	Μέθοδος αναπαραγωγής ήχου .....	48
4.3.8	Μέθοδος checkPassword.....	49
4.3.9	Μέθοδος inDoorOpened.....	50
4.3.10	Μέθοδος setNewPassword() .....	51
4.4	Έλεγχος.....	52
4.4.1	Έλεγχος προδιαγραφών.....	52
4.4.2	Έλεγχος λειτουργίας.....	54
4.4.3	Έλεγχος δομής.....	56
4.5	Κατασκευή.....	58
5	Σύνοψη και Μελλοντική Δουλειά .....	65
5.1	Μελλοντική Δουλειά .....	65
6	Βιβλιογραφία.....	67



## Λίστα Εικόνων

<b>Εικόνα 1:</b> Μαγνητική κλειδαριά.....	6
<b>Εικόνα 2:</b> Ηλεκτρονικό κυπρί.....	7
<b>Εικόνα 3:</b> Κυλινδρικές κλειδαριές ηλεκτρικού χωνευτή.....	8
<b>Εικόνα 4:</b> Σύστημα ελέγχου εξόδου κινδύνου.....	8
<b>Εικόνα 5:</b> Κλειδαριά με αριθμητικούς κώδικες.....	9
<b>Εικόνα 6:</b> Διακριτικό ασφαλείας .....	10
<b>Εικόνα 7:</b> Κλειδαριά βιομετρικού μηχανισμού .....	10
<b>Εικόνα 8:</b> Κλειδαριά με τεχνολογία RFID .....	11
<b>Εικόνα 9:</b> Το ολοκληρωμένο περιβάλλον ανάπτυξης του Arduino .....	19
<b>Εικόνα 10:</b> Η πλακέτα Arduino Mega 2560 .....	21
<b>Εικόνα 11:</b> Σχηματικά διαγράμματα Arduino Mega 2560 .....	22
<b>Εικόνα 12:</b> Πλήρης απεικόνιση των ακίδων του μικροελεγκτή Atmega2560 .....	23
<b>Εικόνα 13:</b> Διάγραμμα ροής για την εισαγωγή κωδικού.....	40
<b>Εικόνα 14:</b> Διάγραμμα ροής για την αλλαγή κωδικού .....	41
<b>Εικόνα 15:</b> Ηλεκτρικό διάγραμμα συστήματος.....	43

( κενή σελίδα )

## Λίστα Τμημάτων Κώδικα

<b>Κώδικας 1:</b> Παράδειγμα κώδικα στο ολοκληρωμένο περιβάλλον ανάπτυξης του Arduino	20
<b>Κώδικας 2:</b> Ορισμός πληκτρολογίου .....	44
<b>Κώδικας 3:</b> Ορισμός ελληνικών χαρακτήρων .....	44
<b>Κώδικας 4:</b> Άλλες σταθερές .....	45
<b>Κώδικας 5:</b> Μέθοδος setup .....	46
<b>Κώδικας 6:</b> Μέθοδος loop .....	47
<b>Κώδικας 7:</b> Μέθοδος keypadEvent.....	48
<b>Κώδικας 8:</b> Μέθοδος clearEnteredPassword .....	48
<b>Κώδικας 9:</b> Μέθοδος printEnterPasswordPrompt .....	48
<b>Κώδικας 10:</b> Μέθοδος playBellSound.....	49
<b>Κώδικας 11:</b> Μέθοδος checkPassword.....	50
<b>Κώδικας 12:</b> Μέθοδος inDoorOpened.....	51
<b>Κώδικας 13:</b> Μέθοδος setNewPassword .....	51

( κενή σελίδα )

## Λίστα Πινάκων

<b>Πίνακας 1:</b> Χαρακτηριστικά Arduino Mega 2560 .....	24
<b>Πίνακας 2:</b> Κοστολόγιο .....	38
<b>Πίνακας 3:</b> Έλεγχος προδιαγραφών .....	54

( κενή σελίδα )

# 1 Εισαγωγή

Ένα από τα σημαντικότερα και πιο πολύπλοκα μοντέλα για την αρμονική αλληλοσύνδεση των ανθρώπινων αξιών και αγαθών είναι η ασφάλεια. Στο πλαίσιο της παρούσας πτυχιακής εργασίας, η ασφάλεια αναφέρεται ως ο βαθμός αντίστασης και προστασίας πολύτιμων και ευαίσθητων συστημάτων και δεδομένων για τον άνθρωπο, από εξωτερικές απειλές.

Προκειμένου να προστατευθούν οι διάφοροι πόροι επιτυχώς, έχουν αναπτυχθεί διάφοροι μηχανισμοί ασφάλειας ανάμεσα σε αυτούς και τις εξωτερικές απειλές, δημιουργώντας πολλαπλά επίπεδα προστασίας. Οι μηχανισμοί αυτοί είναι γνωστοί ως συστήματα ελέγχου και χρησιμοποιούνται ώστε να εξασφαλίζεται η ομαλή και ασφαλή λειτουργία των προστατευόμενων πόρων.

Στα πλαίσια της παρούσας πτυχιακής εργασίας, θα σχεδιασθεί και θα υλοποιηθεί ένα πλήρες σύστημα ελέγχου εισόδου – εξόδου, με χρήση κυκλώματος μικροελεγκτή και πληκτρολογίου. Συγκεκριμένα θα αναπτυχθεί ένα σύστημα ηλεκτρικής κλειδαριάς πόρτας, της οποίας η ενεργοποίηση και απενεργοποίηση ελέγχεται μέσω της διάσημης πλατφόρμας εργαλείων Arduino.

Το σύστημα ελέγχου που θα αναπτύξουμε σε αυτήν την εργασία θα επιτρέπει μόνο σε εξουσιοδοτημένα άτομα να έχουν πρόσβαση στο φυλασσόμενο χώρο, καθώς θα απαιτείται η εισαγωγή κωδικού ασφάλειας μέσω πληκτρολογίου ώστε να τους επιτραπεί η είσοδος.

## 1.1 Δομή εργασίας

Για την καλύτερη πλοήγηση και ανάγνωση του εγγράφου, το υπόλοιπο μέρος της παρούσας διπλωματικής εργασίας δομείται σε πέντε κεφάλαια. Στο επόμενο κεφάλαιο, αναλύεται η θεμελιώδης έννοια της ασφάλειας, και η σημασία της στην ανθρώπινη ζωή και καθημερινότητα. Στη συνέχεια, γίνεται εκτενής αναφοράς στα διάφορα είδη ηλεκτρονικών κλειδαριών, τα πλεονεκτήματα και μειονεκτήματά τους, τη λειτουργία τους και τους μηχανισμούς ελέγχου που χρησιμοποιούν.

Στο τρίτο κεφάλαιο, παρουσιάζεται το Arduino, μια πλατφόρμα εργαλείων για την ανάπτυξη και δημιουργία υλικού και λογισμικού ανοικτού κώδικα, τα οποία χρησιμοποιούνται για την υλοποίηση και κατασκευή ψηφιακών συσκευών και διαδραστικών συστημάτων.

Συγκεκριμένα, πραγματοποιείται μια σύντομη περιγραφή της λειτουργίας και της ιστορίας του Arduino, ενώ αναλύεται με λεπτομέρεια το υλικό, το λογισμικό και η διαδικασία ανάπτυξης συστημάτων που απαιτείται από την πλατφόρμα, εστιάζοντας στο χρησιμοποιούμενο μοντέλο Arduino της παρούσας πτυχιακής εργασίας, δηλαδή το μοντέλο Arduino Mega 2560 (rev. 3).

Στο τέταρτο κεφάλαιο, αναλύεται ο κύκλος ζωής ενός συστήματος και ορίζονται τα διάφορα στάδια του. Συγκεκριμένα, για τη δημιουργία της ηλεκτρονικής κλειδαριάς, βασιζόμαστε στα εξής στάδια: προδιαγραφές συστήματος, σχεδίαση, υλοποίηση, έλεγχος και κατασκευή

του συστήματος. Στα στάδια αυτά, πραγματοποιείται ανάλυση και ορισμός των απαιτήσεων και εξαγωγή των προδιαγραφών του συστήματος, σχεδιάζεται το σύστημα με μεγάλη λεπτομέρεια και ακρίβεια, συγγράφεται ο απαραίτητος κώδικας, προγραμματίζεται και ενεργοποιείται το σύστημα, ελέγχεται η λειτουργία και η συμπεριφορά του συστήματος και κατασκευάζεται ως φυσική υπόσταση, αντίστοιχα.

Τέλος, στο πέμπτο κεφάλαιο παρουσιάζεται μια περίληψη του συνολικού έργου, καθώς και τα μελλοντικά βήματα για την επέκταση του έργου. Στο έκτο, και τελευταίο, κεφάλαιο παρουσιάζεται η βιβλιογραφία που χρησιμοποιήθηκε για τη συγγραφή της παρούσας πτυχιακής εργασίας.



## 2 Ασφάλεια και Ηλεκτρονική Κλειδαριά

### 2.1 Φυσική ασφάλεια

Η φυσική ασφάλεια αφορά μέτρα ασφαλείας που έχουν σχεδιαστεί ώστε να εμποδίζεται η μη εξουσιοδοτημένη είσοδος σε εγκαταστάσεις, εξοπλισμό και πόρους, και για να προστατεύεται το προσωπικό και η περιουσία από βλάβες ή ζημιές. Η φυσική ασφάλεια περιλαμβάνει τη χρήση πολλαπλών αλληλεξαρτώμενων συστημάτων που μπορεί να περιέχουν κάμερες, φρουρούς, προστατευτικές μπάρες, κλειδαριές, πρωτόκολλα ελέγχου πρόσβασης και πολλές άλλες τεχνικές.

Τα συστήματα φυσικής ασφαλείας έχουν γενικά δύο βασικούς σκοπούς:

- Να αποτρέψουν πιθανούς εισβολείς,
- Να ενεργοποιήσουν τις κατάλληλες αντιδράσεις για κάθε περιστατικό.

Για την εξισορρόπηση των μεθόδων ασφαλείας έναντι των κινδύνων είναι υπεύθυνοι οι σχεδιαστές, οι αρχιτέκτονες και οι αναλυτές ασφαλείας και κατά τη διαδικασία θα πρέπει να λαμβάνουν υπόψιν τους το κόστος καθορισμού, ανάπτυξης, ελέγχου, υλοποίησης, χρήσης, διαχείρισης και συντήρησης των μεθόδων ελέγχου, καθώς επίσης και ευρύτερα θέματα όπως η αισθητική, τα ανθρώπινα δικαιώματα, η υγεία και η ασφάλεια και τα περιορισμοί που τυχόν να προέρχονται από τα τοπικά ήθη και έθιμα. Για παράδειγμα τα μέτρα ασφαλείας της φυσικής πρόσβασης είναι κατάλληλα για μια φυλακή υψίστης ασφαλείας ή για μια στρατιωτική μονάδα αλλά είναι ακατάλληλα για ένα γραφείο, ή ένα σπίτι, παρόλο που οι γενικές αρχές είναι ίδιες.

#### 2.1.1 Έλεγχος πρόσβασης

Διάφορες μέθοδοι ελέγχου πρόσβασης χρησιμοποιούνται για την παρακολούθηση και τον έλεγχο της κυκλοφορίας σε συγκεκριμένα σημεία και περιοχές ασφαλών εγκαταστάσεων.

##### 2.1.1.1 Είδη συστημάτων ελέγχου πρόσβασης

Τα τρία είδη συστημάτων ελέγχου πρόσβασης είναι:

###### 2.1.1.1.1 Συστήματα μηχανικού ελέγχου πρόσβασης

Τα συστήματα μηχανικού ελέγχου πρόσβασης περιλαμβάνουν περιστρεφόμενες πόρτες, πύλες, πόρτες και κλειδαριές. Ο έλεγχος με κλειδί αποτελεί πρόβλημα όταν πρόκειται για μεγάλο αριθμό χρηστών. Η διαχείριση των κλειδιών γίνεται δύσκολη πολύ γρήγορα, και συχνά επιβάλλει μεθόδους ηλεκτρονικού ελέγχου πρόσβασης.

###### 2.1.1.1.2 Συστήματα ηλεκτρονικού ελέγχου πρόσβασης

Για μεγάλο αριθμό χρηστών ενδείκνυται η χρήση ηλεκτρονικών συστημάτων ελέγχου πρόσβασης καθώς είναι δυνατό να καθοριστούν οι προσβάσιμες μέρες και ώρες ανά χρήση καθώς και ατομικά σημεία πρόσβασης. Για παράδειγμα τα ατομικά δικαιώματα πρόσβασης ενός χρήστη μπορεί να του επιτρέπουν την πρόσβαση μεταξύ 7 το πρωί και 7 το βράδυ από Δευτέρα έως Παρασκευή και λήγουν σε 90 μέρες. Αυτά τα συστήματα ελέγχου πρόσβασης

συνήθως συνδυάζονται με περιστρεφόμενες πόρτες για τον έλεγχο εισόδου σε κτήρια ώστε να αποτρέπεται η μη εξουσιοδοτημένη πρόσβαση.

#### 2.1.1.1.3 Συστήματα αναγνώρισης και πολιτικές πρόσβασης

Ένα άλλο είδος ελέγχου πρόσβασης περιλαμβάνει τη χρήση πολιτικών, διεργασιών και διαδικασιών πρόσβασης ώστε να γίνεται διαχείριση της πρόσβασης σε απαγορευμένες περιοχές. Ένα παράδειγμα αποτελεί η χρήση προσωπικού ασφαλείας, το οποίο πραγματοποιεί ελέγχους ταυτότητας σε προκαθορισμένα σημεία εισόδου. Αυτή η μορφή ελέγχου πρόσβασης συνήθως συμπληρώνεται από τα συστήματα ελέγχου πρόσβασης που αναφέρθηκαν παραπάνω, ή από απλές συσκευές, όπως είναι οι φυσικοί ελεγκτές.

#### 2.1.1.2 Λειτουργία συστημάτων ελέγχου πρόσβασης

Όταν ένα διαπιστευτήριο παρουσιάζεται σε έναν αναγνώστη, ο αναγνώστης στέλνει τις πληροφορίες διαπίστευσης, συνήθως έναν αριθμό, σε έναν πίνακα ελέγχου που δεν είναι παρά ένας αξιόπιστος επεξεργαστής. Ο πίνακας ελέγχου συγκρίνει τον αριθμό διαπίστευσης με μια λίστα ελέγχου πρόσβασης, δέχεται ή απορρίπτει την αίτηση, και στέλνει ένα αρχείο κίνησης σε μια βάση δεδομένων. Όταν απορρίπτεται η πρόσβαση με βάση τη λίστα ελέγχου, η πόρτα παραμένει κλειστή.

Σε περίπτωση που η διαπίστευση ταιριάζει με ένα αντικείμενο της λίστας ελέγχου, τότε ο πίνακας ελέγχου ενεργοποιεί έναν ηλεκτρονόμο, ο οποίος με τη σειρά του ξεκλειδώνει την πόρτα. Ο πίνακας ελέγχου αγνοεί το σήμα ανοιχτής πόρτας ώστε να μην ενεργοποιηθεί ο συναγερμός. Συχνά ο αναγνώστης παρέχει κάποιο είδος ανάδρασης, όπως ένα κόκκινο LED ανάβει σε περίπτωση που απορριφθεί το αίτημα ή ένα πράσινο LED ανάβει σε περίπτωση που γίνει αποδεκτό.

Στην παραπάνω διαδικασία υπάρχει ένας μοναδικός παράγοντας ταυτοποίησης. Η λίστα πρόσβασης μπορεί εύκολα να παρακαμφθεί καθώς το διαπιστευτήριο μπορεί εύκολα να αποκαλυφθεί σε άλλους χρήστες, οι οποίοι είναι πιθανό να μην έχουν τα ίδια δικαιώματα πρόσβασης. Για να εμποδιστεί αυτό, μπορούν να χρησιμοποιηθούν δύο μέθοδοι ταυτοποίησης.

Σε μια διαδικασία όπου χρησιμοποιούνται δύο μέθοδοι ταυτοποίησης, απαιτείται ένα διαπιστευτήριο και ένας επιπλέον παράγοντας ώστε να γίνει η αποδεκτή η αίτηση για παραχώρηση πρόσβασης. Αυτός ο δεύτερος παράγοντας μπορεί να είναι ένα PIN, ένα βιομετρικό, ένα δεύτερο διαπιστευτήριο, ή η παρέμβαση ενός διαχειριστή.

Υπάρχουν τρία είδη παραγόντων που μπορούν να χρησιμοποιηθούν για να γίνει ταυτοποίηση των χρηστών:

- Κάτι που γνωρίζει ο χρήστης, όπως για παράδειγμα ένας αλφαριθμητικός κωδικός (PIN)
- Κάτι που έχει ο χρήστης, όπως για παράδειγμα μια έξυπνη κάρτα
- Κάτι που χαρακτηρίζει το χρήστη μοναδικά, όπως για παράδειγμα το δαχτυλικό αποτύπωμα

Ένα διαπιστευτήριο λοιπόν μπορεί να είναι ένα φυσικό αντικείμενο, κάτι που γνωρίζει ο χρήστης, ή ένα μοναδικό χαρακτηριστικό του χρήστη, το οποίο του δίνει τη δυνατότητα πρόσβασης σε μια εγκατάσταση. Ένα τυπικό διαπιστευτήριο είναι συνήθως μια κάρτα ή ένα ειδικό κλειδί, ή ακόμα και το έξυπνο κινητό τηλέφωνο ενός χρήστη.

Υπάρχουν διάφορες τεχνολογίες υλοποίησης της κάρτας όπως είναι η μαγνητική ταινία, το barcode, οι κάρτες περάσματος, και γενικότερα οι κάρτες που απαιτούν επαφή και εκείνες που δεν απαιτούν επαφή. Οι βιομετρικές τεχνολογίες συμπεριλαμβάνουν τα αποτυπώματα, την αναγνώριση προσώπου, της αναγνώριση της ίριδας, την αναγνώριση φωνής, και των γεωμετρικών χαρακτηριστικών του χρήστη. Κάποιες νέες τεχνολογίες όπως είναι η επικοινωνία σε κοντινό πεδίο (NFC) ή η τεχνολογία Bluetooth, επιτρέπουν την επικοινωνία με τον αναγνώστη του συστήματος ελέγχου πρόσβασης από απόσταση.

### 2.1.1.3 Στοιχεία ενός συστήματος ελέγχου πρόσβασης

Ένα σημείο ελέγχου πρόσβασης, το οποίο μπορεί να είναι μια πόρτα, μια μπάρα, ένα ασανσέρ ή ένα φυσικό φράγμα, όπου η παραχώρηση πρόσβασης ελέγχεται ηλεκτρονικά. Το πιο συνηθισμένο σημείο πρόσβασης είναι η πόρτα.

Μια πόρτα ηλεκτρονικού ελέγχου πρόσβασης μπορεί να περιέχει αρκετά στοιχεία. Στην πιο απλή της μορφή, υπάρχει μόνο μια ηλεκτρονική κλειδαριά, την οποία ξεκλειδώνει ο διαχειριστής μέσω ενός διακόπτη. Για να αυτοματοποιηθεί η διαδικασία, ο διαχειριστής αντικαθίσταται από έναν αναγνώστη, ο οποίος μπορεί να είναι ένα αριθμητικό πληκτρολόγιο στο οποίο εισάγεται ένας κωδικός, ή ένας αναγνώστης καρτών, ή ένας αναγνώστης βιομετρικών χαρακτηριστικών.

Οι αναγνώστες συνήθως δεν αποφασίζουν εάν θα παραχωρήσουν πρόσβαση, αλλά στέλνουν έναν αριθμό στον πίνακα ελέγχου και εκεί ελέγχεται εάν ο αριθμός είναι καταχωρημένος στη λίστα πρόσβασης. Για την παρακολούθηση της θέσης της πόρτας μπορεί να χρησιμοποιηθεί ένας μαγνητικός διακόπτης. Γενικά πραγματοποιείται έλεγχος μόνο κατά την είσοδο, ενώ κατά την έξοδο δεν είναι απαραίτητο να γίνει έλεγχος.

Σε περιπτώσεις όπου ελέγχεται και η έξοδος ελέγχεται, ένας δεύτερος αναγνώστης χρησιμοποιείται στην άλλη πλευρά της πόρτας. Σε περιπτώσεις όπου δεν ελέγχεται η έξοδος χρησιμοποιείται μια συσκευή η οποία ονομάζεται REX (request to exit). Ο χρήστης αρκεί να πατήσει το διακόπτη για να ξεκλειδώσει η πόρτα, ή η πόρτα ανοίγει αυτόματα όταν ανιχνευτεί κίνηση.

Σε κάθε περίπτωση ο συναγερμός απενεργοποιείται προσωρινά όσο η πόρτα παραμένει ανοιχτή. Το γεγονός ότι δεν απαιτείται ηλεκτρονικό ξεκλείδωμα της κλειδαριάς για την έξοδο αποτελεί ένα σημαντικό χαρακτηριστικό ασφάλειας.

## 2.2 Ηλεκτρονική Κλειδαριά

Η ηλεκτρονική κλειδαριά (ή ηλεκτρική κλειδαριά) είναι μια συσκευή κλειδώματος η οποία λειτουργεί με ηλεκτρικό ρεύμα. Οι ηλεκτρονικές κλειδαριές χρησιμοποιούνται είτε σαν αυτόνομες συσκευές είτε σε συνδυασμό με ένα σύστημα ελέγχου πρόσβασης.

Στη δεύτερη περίπτωση παρουσιάζουν κάποια πλεονεκτήματα όπως είναι για παράδειγμα ο έλεγχος κλειδιών, όπου είναι δυνατό να προστεθούν ή να αφαιρεθούν κλειδιά χωρίς να απαιτείται επαναρύθμιση του κυλίνδρου της κλειδαριάς, έλεγχος ορθής πρόσβασης, όπου ο χρόνος και η τοποθεσία αποτελούν παράγοντες ελέγχου, καθώς και καταγραφή κινήσεων, όπου καταγράφεται η δραστηριότητα του σημείου ελέγχου.

### 2.2.1 Λειτουργία

Οι ηλεκτρονικές κλειδαριές χρησιμοποιούν μαγνήτες, ηλεκτρομαγνητικές βαλβίδες, ή κινητήρες για να κινήσουν την κλειδαριά είτε παρέχοντας ή αποκόπτοντας ισχύ. Η λειτουργία της κλειδαριάς μπορεί να είναι τόσο απλή όσο η χρήση ενός διακόπτη, όπως για παράδειγμα το σύστημα ενδοεπικοινωνίας σε πολυκατοικίες που επιτρέπει την απελευθέρωση της πόρτας, ή τόσο περίπλοκη όσο ένα βιομετρικό σύστημα ελέγχου πρόσβασης. Υπάρχουν δύο βασικά είδη κλειδαριάς: ο μηχανισμός πρόληψης και ο μηχανισμός λειτουργίας.

### 2.2.2 Είδη ηλεκτρονικής κλειδαριάς

Υπάρχουν διάφορα είδη ηλεκτρονικής κλειδαριάς, τα οποία παρουσιάζουν ορισμένα πλεονεκτήματα και μειονεκτήματα και χρησιμοποιούνται ανάλογα με το απαιτούμενο επίπεδο ασφάλειας καθώς και την επιθυμητή απόκριση της κλειδαριάς σε περίπτωση αποτυχίας. Παρακάτω παρατίθενται συνοπτικά πέντε είδη ηλεκτρικής κλειδαριάς.

#### 2.2.2.1 Μαγνητικές κλειδαριές

Το πιο σημαντικό είδος ηλεκτρονικής κλειδαριάς είναι η μαγνητική κλειδαριά. Ένας μεγάλος ηλεκτρομαγνήτης είναι τοποθετημένος στο πλαίσιο της πόρτας και ο αντίστοιχος οπλισμός είναι τοποθετημένος στην πόρτα. Όταν ο μαγνήτης τροφοδοτείται η πόρτα είναι κλειδωμένη, ο οπλισμός έλκεται δυνατά από το μαγνήτη. Η εγκατάσταση της μαγνητικής κλειδαριάς είναι εύκολη και είναι πολύ ανθεκτικές σε επιθέσεις.



Εικόνα 1: Μαγνητική κλειδαριά

Ένα μειονέκτημά τους είναι ότι σε περίπτωση που δε γίνει σωστή εγκατάσταση ή δε συντηρηθούν σωστά μπορεί να πέσουν πάνω στον άνθρωπο, και επιπλέον απαιτείται κλείδωμα και κατά την είσοδο και κατά την έξοδο. Αυτό έχει προκαλέσει την επιβολή αυστηρών κωδίκων σχετικά με τη χρήση των μαγνητικών κλειδαριών και την πρακτική του ελέγχου της πρόσβασης σε γενικές γραμμές από τους υπεύθυνους πυρασφάλειας.

Ένα άλλο μειονέκτημα της μαγνητικής κλειδαριάς είναι η μικρή χρονοκαθυστέρηση στην απελευθέρωση της πόρτας μετά το ξεκλείδωμα καθώς το μαγνητικό πεδίο δεν αδρανοποιείται στιγμιαία γεγονός που μπορεί να οδηγήσει το χρήστη να πέσει πάνω στην πόρτα.

Ολοκληρώνοντας, οι μαγνητικές κλειδαριές όταν αποτυγχάνουν ξεκλειδώνονται, αν δηλαδή διακοπεί η παροχή ενέργειας ξεκλειδώνουν γεγονός που μπορεί να αποτελεί πρόβλημα όταν η ασφάλεια αποτελεί πρωταρχικό μέλημα. Επιπλέον, οι διακοπές ρεύματος θα μπορούσαν να επηρεάσουν τις μαγνητικές κλειδαριές που έχουν εγκατασταθεί σε πόρτες πυροπροστασίας, οι οποίες απαιτείται να παραμένουν ασφαλισμένες ανά πάσα στιγμή. Επομένως, μια μαγνητική κλειδαριά δεν πληροί τις προϋποθέσεις θυρών πυρασφάλειας.

#### 2.2.2.2 Ηλεκτρονικό κυπρί

Το ηλεκτρονικό κυπρί αντικαθιστά το μηχανικό κυπρί που είναι τοποθετημένο στο πλαίσιο της πόρτας δέχεται το μάνταλο. Ένα ηλεκτρονικό κυπρί μπορεί να απαιτεί απλή εγκατάσταση όταν αντικαθιστά ένα μηχανικό κυπρί, αν και τα ηλεκτρονικά κυπρί κάποιες φορές απαιτούν εκτεταμένη τροποποίηση στο πλαίσιο της πόρτας. Αποτυχία ενός ηλεκτρικού κυπρί μπορεί είτε να σημαίνει ξεκλείδωμα της πόρτας ή κλείδωμα της πόρτας, ανάλογα με την απαιτούμενη λειτουργία. Για παράδειγμα εάν πρόκειται για πόρτα πυρασφάλειας αυτή θα πρέπει να παραμένει κλειδωμένη σε περίπτωση αποτυχίας. Τα ηλεκτρικά κυπρί είναι πιο εύκολο να παραβιαστούν σε σχέση με τις μαγνητικές κλειδαριές καθώς συνήθως υπάρχει ένα κενό μεταξύ του κυπρί και του μαντάλου της πόρτας. Το κενό αυτό συχνά καλύπτεται με ένα προστατευτικό μαντάλου.



Εικόνα 2: Ηλεκτρονικό κυπρί

### 2.2.2.3 Κυλινδρικές κλειδαριές ηλεκτρικού χωνευτή

Οι κλειδαριές αυτές αντικαθιστούν τις μηχανικές κλειδαριές και το μόνο που απαιτείται είναι να ανοιχτεί μια επιπλέον τρύπα στην πόρτα για τα καλώδια. Επίσης είναι απαραίτητο να χρησιμοποιηθεί ένας μεντεσές μεταφοράς ενέργειας προκειμένου με μεταβίβαστεί η ενέργεια από το πλαίσιο της πόρτας, στην πόρτα. Οι κυλινδρικές κλειδαριές ηλεκτρονικού χωνευτή επιτρέπουν τη μη μηχανική έξοδο και η αποτυχία τους μπορεί είτε να σημαίνει κλείδωμα της πόρτας είτε ξεκλείδωμα, ανάλογα με την επιθυμητή λειτουργία.



Εικόνα 3: Κυλινδρικές κλειδαριές ηλεκτρικού χωνευτή

### 2.2.2.4 Συστήματα ελέγχου εξόδου κινδύνου

Τα συστήματα αυτά χρησιμοποιούνται σε εφαρμογές εξόδων κινδύνου. Αρκεί ο χρήστης να σπρώξει τη μπάρα για να ανοίξει η πόρτα, γεγονός που καθιστά αυτό το είδος κλειδαριάς την πιο εύκολη μη μηχανική λύση για έξοδο. Τα συστήματα ελέγχου εξόδου κινδύνου μπορούν είτε να ξεκλειδώνουν όταν αποτυγχάνουν, είτε να παραμένουν κλειδωμένα ανάλογα με τις απαιτήσεις. Ένα μειονέκτημά τους είναι η πολυπλοκότητά τους, η οποία απαιτεί τη σωστή εγκατάσταση και συντήρηση προκειμένου να λειτουργούν σωστά.



Εικόνα 4: Σύστημα ελέγχου εξόδου κινδύνου

### 2.2.2.5 Κλειδαριές με κινητήρα

Οι κλειδαριές αυτές χρησιμοποιούνται κυρίως στην Ευρώπη και διαθέτουν δύο καταστάσεις λειτουργίας: λειτουργία ημέρας όπου μόνο το μάνταλο λειτουργεί ηλεκτρονικά, και λειτουργία νύχτας όπου λειτουργεί ηλεκτρονικά και η ασφαλέστερη κλειδαριά.

### 2.2.3 Μέθοδοι ελέγχου ταυτότητας

Οι ηλεκτρονικές κλειδαριές προσφέρουν μια ποικιλία μεθόδων ελέγχου ταυτότητας. Παρακάτω θα αναφερθούν οι τέσσερις πιο συνηθισμένες μέθοδοι.

#### 2.2.3.1 Αριθμητικοί κώδικες

Ο επικρατέστερος ίσως τρόπος ελέγχου ταυτότητας στις ηλεκτρονικές κλειδαριές είναι ίσως ο αριθμητικός κώδικας όπου απαιτείται εισαγωγή του σωστού κωδικού προκειμένου να απενεργοποιηθεί η κλειδαριά και να ανοίξει η πόρτα. Αυτού του τύπου οι κλειδαριές συνήθως παρέχουν ένα πληκτρολόγιο, και κάποιο είδος ηχητικής απόκρισης σε κάθε πάτημα πλήκτρου. Οι συνδυασμοί έχουν συνήθως μήκος από 4 έως 6 ψηφία.



Εικόνα 5: Κλειδαριά με αριθμητικούς κώδικες

### 2.2.3.2 Διακριτικό ασφαλείας

Ένας άλλος τρόπος ελέγχου της ταυτότητας είναι η απαίτηση για σάρωση ή πέρασμα ενός διακριτικού ασφαλείας όπως για παράδειγμα μιας έξυπνης κάρτας. Κάποιες κλειδαριές έχουν τη δυνατότητα πρόσβασης αποθηκευμένα διαπιστευτήρια σε PDA χρησιμοποιώντας μεθόδους πρόσβασης δεδομένων με υπέρυθρες.



Εικόνα 6: Διακριτικό ασφαλείας

### 2.2.3.3 Βιομετρικά

Τα βιομετρικά κερδίζουν όλο και περισσότερο την αγορά ως ένα αναγνωρισμένο μέσω θετικής ταυτοποίησης και αυξάνεται συνεχώς η χρήση τους σε συστήματα ασφαλείας. Κάποιες νέες ηλεκτρονικές κλειδαριές εκμεταλλεύονται τεχνολογίες όπως η σάρωση δακτυλικού αποτυπώματος, η σάρωση της ίριδας του ματιού και φωνητική αναγνώριση για την ταυτοποίηση των χρηστών.



Εικόνα 7: Κλειδαριά βιομετρικού μηχανισμού



#### 2.2.3.4 RFID

Αναγνώριση ραδιοσυχνότητας ή RFID καλείται η χρήση ενός αντικειμένου (συνήθως γνωστό ως RFID ετικέτα) το οποίο ενσωματώνεται σε κάποιο προϊόν, ζώο ή άνθρωπο με σκοπό την αναγνώρισή του και τον εντοπισμό του με χρήση ραδιοκυμάτων. Κάποιες ετικέτες είναι δυνατό να διαβάζονται από αρκετά μέτρα μακριά και πέρα από το ορατό πεδίο του λήπτη. Αυτή η τεχνολογία χρησιμοποιείται στις μοντέρνες ηλεκτρονικές κλειδαριές.



Εικόνα 8: Κλειδαριά με τεχνολογία RFID

( κενή σελίδα )

## 3 Arduino

Για τη δημιουργία της ηλεκτρονικής κλειδαριάς θα χρησιμοποιηθεί μικροελεγκτής από την Arduino. Σε αυτό το κεφάλαιο γίνεται μια αναφορά στο Arduino καθώς και στο μικροελεγκτή που θα χρησιμοποιηθεί για την υλοποίηση του συστήματος.

### 3.1 Σύντομη Περιγραφή

Το Arduino είναι ένας πάροχος υλικού και λογισμικού ανοικτού κώδικα, ο οποίος σχεδιάζει και αναπτύσσει εργαλεία τα οποία χρησιμοποιούνται για την υλοποίηση και κατασκευή ψηφιακών συσκευών και διαδραστικών συστημάτων.

Οι πλακέτες του Arduino παρέχονται είτε ως προσυναρμολογημένες είτε ως σύνολα τα οποία μπορεί να κατασκευάσει ο ενδιαφερόμενος από την αρχή, γνωστά και ως φτιάξε-το-μόνος ή do-it-yourself σύνολα. Και στις δυο περιπτώσεις οι πληροφορίες για τη σχεδίαση υλικού είναι διαθέσιμες.

Η ανάπτυξη της πλατφόρμας Arduino βασίζεται σε συστήματα μικροελεγκτών τα οποία κατασκευάζονται κυρίως από την εταιρία SmartProjects, χρησιμοποιώντας μικροελεγκτές και επεξεργαστές της εταιρίας Atmel (Arduino, 2014). Τα συστήματα αυτά παρέχουν σύνολα ψηφιακών και αναλογικών ακίδων εισόδου – εξόδου (I/O pins), τα οποία μπορούν να συνδεθούν σε διάφορες πλακέτες επέκτασης και άλλα κυκλώματα.

Οι πλακέτες παρέχουν σειριακές διεπαφές επικοινωνίας, συμπεριλαμβανομένου του προτύπου USB σε κάποια μοντέλα, ώστε να εξυπηρετούν την εύκολη και γρήγορη φόρτωση προγραμμάτων από προσωπικούς υπολογιστές. Για τον προγραμματισμό των μικροελεγκτών, η πλατφόρμα Arduino παρέχει ένα ολοκληρωμένο περιβάλλον ανάπτυξης, το οποίο υποστηρίζει τη συγγραφή κώδικα σε C και C++.

Το πρώτο σύστημα Arduino αναπτύχθηκε το 2005, ενώ σήμερα η χρήση του έχει εξαπλωθεί σε όλο τον κόσμο με ραγδαίους ρυθμούς ανάπτυξης, καθώς εκτιμάται ότι περισσότερες από εκατό χιλιάδες επίσημες πλακέτες χρησιμοποιούνται σε διάφορα έργα (Cuartielles, 2013).

### 3.2 Ιστορικά Στοιχεία

Η ανάπτυξη της πλατφόρμας Arduino ξεκίνησε πριν δέκα χρόνια από φοιτητές του Ινστιτούτου Διαδραστικού Σχεδιασμού στην πόλη Ivrea της Ιταλίας. Στόχος των δημιουργών του Arduino ήταν να αναπτύξουν μια πλατφόρμα υλικού και λογισμικού χαμηλού κόστους και υψηλής απόδοσης, η οποία θα ήταν διαθέσιμη στην κοινότητα ανοικτού κώδικα (Kushner, 2011).

Η αρχική ομάδα ανάπτυξης του Arduino ήταν οι Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino και David Mellis. Το όνομα Arduino προέκυψε από ένα μπαρ της

πόλης Ivrea στο οποίο συνήθιζαν να συναντιούνται τα μέλη της ομάδας ανάπτυξης (Kushner, 2011).

### 3.3 Υλικό

Κάθε πλακέτα Arduino αποτελείται από ένα AVR μικροελεγκτή της Atmel, των 8-, 16- ή 32-μπιτ με διάφορα συμπληρωματικά στοιχεία, ώστε να διευκολύνεται ο προγραμματισμός και η ενσωμάτωση του συστήματος σε πιο περίπλοκα κυκλώματα.

Μια από τις σημαντικότερες πτυχές του Arduino είναι οι υποδοχείς σύνδεσης, που επιτρέπουν τη διασύνδεση της πλακέτας με διάφορα σύνολα επέκτασης. Κάποια από τα σύνολα επέκτασης επικοινωνούν με την πλακέτα του Arduino απευθείας μέσω των ακίδων, ενώ άλλα είναι προσβάσιμα μέσω σειριακού διαύλου.

Όσον αφορά τα μικροστοιχεία που χρησιμοποιούνται από το Arduino, αυτά αποτελούν κυρίως τη σειρά megaAVR. Ωστόσο, έχουν χρησιμοποιηθεί κι άλλοι επεξεργαστές, κυρίως από συστήματα συμβατά με το Arduino.

Οι περισσότερες πλακέτες περιλαμβάνουν ένα γραμμωτό ρυθμιστή 5V και έναν κρυσταλλικό ταλαντωτή 16MHz. Επίσης, ο μικροελεγκτής της πλακέτας είναι προ-προγραμματισμένος με ένα μικρό πρόγραμμα φόρτωσης εκκίνησης, ώστε να απλοποιείται η διαδικασία φόρτωσης προγραμμάτων στη μνήμη.

Οι περισσότερες από τις ακίδες εισόδου – εξόδου της πλακέτας του Arduino προορίζονται για χρήση από άλλα κυκλώματα. Για παράδειγμα, οι πλακέτες Diecimila, Duemilanove, και Uno παρέχουν δεκατέσσερις ψηφιακές ακίδες εισόδου – εξόδου, έξι από τις οποίες μπορούν να παράγουν σήματα διαμόρφωσης εύρους παλμού, καθώς και έξι αναλογικές ακίδες εισόδου, οι οποίες επίσης μπορούν να χρησιμοποιηθούν ως ψηφιακές ακίδες εισόδου – εξόδου.

Στην αγορά πλέον υπάρχουν πολλές πλακέτες που είναι συμβατές με το Arduino. Κάποιες από αυτές είναι λειτουργικά ισοδύναμες με το Arduino και μπορούν να χρησιμοποιηθούν ως εναλλακτική λύση. Πολλές από αυτές βελτιώνουν τη λειτουργία του βασικού Arduino προσθέτοντας οδηγούς εξόδου, και κυρίως χρησιμοποιούνται για εκπαιδευτικούς σκοπούς.

Άλλες πλακέτες είναι ηλεκτρικά ισοδύναμες αλλά διαφοροποιούν τον παράγοντα μορφής (form factor), ενώ παράλληλα διατηρούν συνήθως τη συμβατότητα με τα σύνολα επέκτασης. Τέλος, κάποια εναλλακτικά συστήματα χρησιμοποιούν τελείως διαφορετικούς επεξεργαστές, χωρίς να εξασφαλίζεται πλήρως η συμβατότητα.

### 3.3.1 Διαθέσιμες πλακέτες

Παρακάτω σημειώνονται οι διαθέσιμες πλακέτες από το Arduino, μαζί με τα κυριότερα χαρακτηριστικά τους.

- **Arduino Zero ή Arduino Zero Pro**  
Επεξεργαστής: ATSAM21G18A (Cortex-M0+)  
Συχνότητα: 48 MHz  
Μέγεθος: 68.6 mm × 53.3 mm  
Διεπαφή ξενιστή: USB / Native & EDBG Debug  
Τάση: 3.3 V  
Μνήμη flash: 256 KB  
Μνήμη EEPROM: από 0 ως 16 KB  
Μνήμη SRAM: 32 KB  
Ακίδες εισόδου – εξόδου: 14  
Ακίδες αναλογικής εισόδου: 6
- **Arduino Due**  
Επεξεργαστής: ATSAM3X8E (Cortex-M3)  
Συχνότητα: 84 MHz  
Μέγεθος: 101.6 mm × 53.3 mm  
Διεπαφή ξενιστή: USB / 16U2 + native host  
Τάση: 3.3 V  
Μνήμη flash: 512 KB  
Μνήμη EEPROM: 0  
Μνήμη SRAM: 96 KB  
Ακίδες εισόδου – εξόδου: 54  
Ακίδες αναλογικής εισόδου: 12
- **Arduino Yún**  
Επεξεργαστής: Atmega32U4 ή Atheros AR9331  
Συχνότητα: 16 MHz ή 400 MHz  
Μέγεθος: 68.6 mm × 53.3 mm  
Διεπαφή ξενιστή: USB  
Τάση: 5 V  
Μνήμη flash: 32 KB ή 16 MB  
Μνήμη EEPROM: 1 KB ή 0 KB  
Μνήμη SRAM: 2.5 KB ή 64 MB  
Ακίδες εισόδου – εξόδου: 14  
Ακίδες αναλογικής εισόδου: 12
- **Arduino Leonardo**  
Επεξεργαστής: Atmega32U4  
Συχνότητα: 16 MHz  
Μέγεθος: 68.6 mm × 53.3 mm  
Διεπαφή ξενιστή: USB / 32U4

Τάση: 5 V

Μνήμη flash: 32 KB

Μνήμη EEPROM: 1 KB

Μνήμη SRAM: 2.5 KB

Ακίδες εισόδου – εξόδου: 20

Ακίδες αναλογικής εισόδου: 12

- **Arduino Uno**

Επεξεργαστής: ATmega328P

Συχνότητα: 16 MHz

Μέγεθος: 68.6 mm × 53.3 mm

Διεπαφή ξενιστή: USB / 8U2 / 16U2

Τάση: 5 V

Μνήμη flash: 32 KB

Μνήμη EEPROM: 1 KB

Μνήμη SRAM: 2 KB

Ακίδες εισόδου – εξόδου: 14

Ακίδες αναλογικής εισόδου: 6

- **Arduino Mega2560**

Επεξεργαστής: ATmega2560

Συχνότητα: 16 MHz

Μέγεθος: 101.6 mm × 53.3 mm

Διεπαφή ξενιστή: USB / 8U2 / 16U2

Τάση: 5 V

Μνήμη flash: 256 KB

Μνήμη EEPROM: 4 KB

Μνήμη SRAM: 8 KB

Ακίδες εισόδου – εξόδου: 54

Ακίδες αναλογικής εισόδου: 16

- **Arduino Ethernet**

Επεξεργαστής: ATmega328

Συχνότητα: 16 MHz

Μέγεθος: 68.6 mm × 53.3 mm

Διεπαφή ξενιστή: Ethernet Serial interface / Wiznet Ethernet

Τάση: 5 V

Μνήμη flash: 32 KB

Μνήμη EEPROM: 1 KB

Μνήμη SRAM: 2 KB

Ακίδες εισόδου – εξόδου: 14

Ακίδες αναλογικής εισόδου: 6

- **Arduino Fio**

Επεξεργαστής: ATmega328P

Συχνότητα: 8 MHz

Μέγεθος: 66.0 mm × 27.9 mm

Διεπαφή ξενιστή: XBee Serial

Τάση: 3.3 V

Μνήμη flash: 32 KB

Μνήμη EEPROM: 1 KB

Μνήμη SRAM: 2 KB

Ακίδες εισόδου – εξόδου: 14

Ακίδες αναλογικής εισόδου: 8

- **Arduino Nano**

Επεξεργαστής: ATmega328

Συχνότητα: 16 MHz

Μέγεθος: 43.18 mm × 18.54 mm

Διεπαφή ξενιστή: USB / FTDI FT232R

Τάση: 5 V

Μνήμη flash: 16/32 KB

Μνήμη EEPROM: 0.5/1 KB

Μνήμη SRAM: 1/2 KB

Ακίδες εισόδου – εξόδου: 14

Ακίδες αναλογικής εισόδου: 8

- **LilyPad Arduino**

Επεξεργαστής: ATmega168V ή ATmega328V

Συχνότητα: 8 MHz

Μέγεθος: φορετό σύστημα / διάμετρος 51 mm

Τάση: 2.7-5.5 V

Μνήμη flash: 16 KB

Μνήμη EEPROM: 0.5 KB

Μνήμη SRAM: 1 KB

Ακίδες εισόδου – εξόδου: 14

Ακίδες αναλογικής εισόδου: 6

- **Arduino Pro**

Επεξεργαστής: ATmega168 ή ATmega328

Συχνότητα: 16 MHz

Μέγεθος: 68.6 mm × 53.3 mm

Διεπαφή ξενιστή: UART Serial, I2C(TWI), SPI / FTDI

Τάση: 5 V ή 3.3 V

Μνήμη flash: 16/32 KB

Μνήμη EEPROM: 0.5/1 KB

Μνήμη SRAM: 1/2 KB

Ακίδες εισόδου – εξόδου: 14

Ακίδες αναλογικής εισόδου: 6

- **Arduino Mega ADK**

Επεξεργαστής: ATmega2560

Συχνότητα: 16 MHz  
Μέγεθος: 101.6 mm × 53.3 mm  
Διεπαφή ξενιστή: 8U2 / MAX3421E USB Host  
Τάση: 5 V  
Μνήμη flash: 256 KB  
Μνήμη EEPROM: 4 KB  
Μνήμη SRAM: 8 KB  
Ακίδες εισόδου – εξόδου: 54  
Ακίδες αναλογικής εισόδου: 16

▪ **Arduino Esplora**

Επεξεργαστής: Atmega32U4  
Συχνότητα: 16 MHz  
Μέγεθος: 165.1 mm × 61.0 mm  
Διεπαφή ξενιστή: 32U4  
Τάση: 5 V  
Μνήμη flash: 32 KB  
Μνήμη EEPROM: 1 KB  
Μνήμη SRAM: 2.5 KB

▪ **Arduino Micro**

Επεξεργαστής: ATmega32U4  
Συχνότητα: 16 MHz  
Μέγεθος: 17.8 mm × 48.3 mm  
Τάση: 5 V  
Μνήμη flash: 32 KB  
Μνήμη EEPROM: 1 KB  
Μνήμη SRAM: 2.5 KB  
Ακίδες εισόδου – εξόδου: 20  
Ακίδες αναλογικής εισόδου: 12

▪ **Arduino (Pro) Mini**

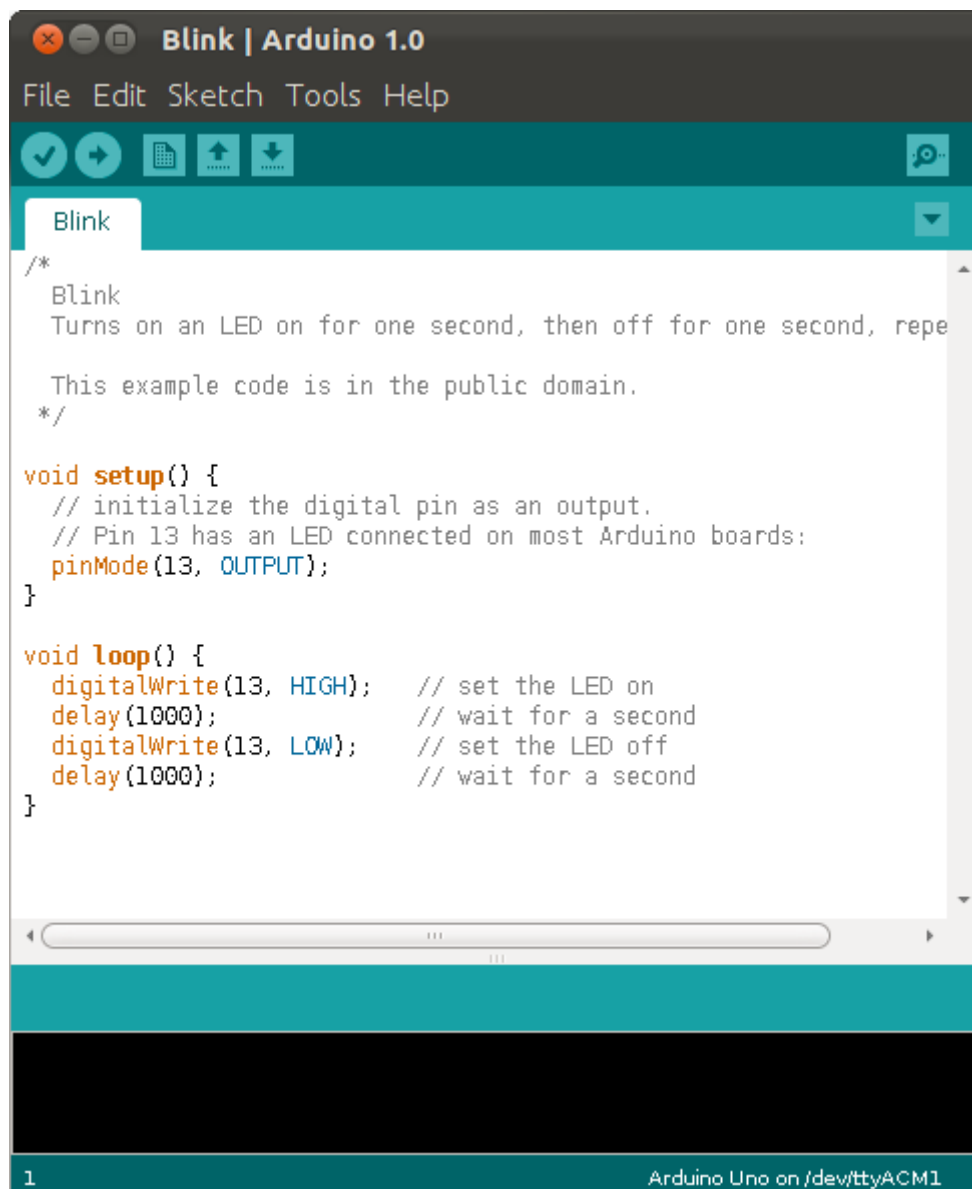
Επεξεργαστής: ATmega168 (Pro: ATmega328)  
Συχνότητα: 8 MHz (3.3 V μοντέλο) ή 16 MHz (5 V μοντέλο)  
Μέγεθος: 17.8 mm × 33.0 mm  
Τάση: 5 V or 3.3 V  
Μνήμη flash: 16 KB  
Μνήμη EEPROM: 0.5 KB  
Μνήμη SRAM: 1 KB  
Ακίδες εισόδου – εξόδου: 14  
Ακίδες αναλογικής εισόδου: 6



### 3.4 Λογισμικό

Το ολοκληρωμένο περιβάλλον ανάπτυξης (integrated development environment – IDE) που παρέχεται από το Arduino είναι μια εφαρμογή, ανεξάρτητης πλατφόρμας, γραμμένη σε γλώσσα Java.

Το περιβάλλον ανάπτυξης είναι σχεδιασμένο έτσι ώστε να εισάγει τους σχεδιαστές υλικού στον κόσμο του προγραμματισμού ομαλά καθώς και να βοηθήσει τους αρχάριους προγραμματιστές. Παρέχει έναν επεξεργαστή κώδικα (code editor) με πλούσια χαρακτηριστικά όπως επισήμανση και χρωματισμός της σύνταξης, έλεγχος αρχής και τέλους παρενθέσεων, αυτόματη δημιουργία εσοχών και επιπέδων, εύκολη και γρήγορη φόρτωση του κώδικα στην πλακέτα, κλπ.



Εικόνα 9: Το ολοκληρωμένο περιβάλλον ανάπτυξης του Arduino

Η γλώσσα που χρησιμοποιείται για την ανάπτυξη προγραμμάτων Arduino είναι η C ή η C++. Το ολοκληρωμένο περιβάλλον παρέχει διάφορες βιβλιοθήκες στο χρήστη ώστε να εισάγει πρόσθετες λειτουργίες. Επιπρόσθετες βιβλιοθήκες είναι διαθέσιμες από την κοινότητα του Arduino και είναι ελεύθερα διαθέσιμες μέσω του διαδικτυακού τόπου του Arduino: [www.arduino.cc](http://www.arduino.cc).

Προκειμένου να είναι εκτελέσιμο ένα πρόγραμμα, απαιτούνται μόνο δυο μέθοδοι:

- `setup()`: η μέθοδος αυτή τρέχει μόνο μια φορά, στην αρχή του προγράμματος, και χρησιμοποιείται για την αρχικοποίηση των ρυθμίσεων.
- `loop()`: η μέθοδος αυτή τρέχει συνεχώς μέχρι η πλακέτα να απενεργοποιηθεί.

Στη συνέχεια παρουσιάζουμε ένα τυπικό παράδειγμα κώδικα και πως αυτό εμφανίζεται στο ολοκληρωμένο περιβάλλον ανάπτυξης του Arduino.

```
#define LED_PIN 13

void setup() {
  pinMode(LED_PIN, OUTPUT);
}

void loop() {
  digitalWrite(LED_PIN, HIGH);
  delay(1000);
  digitalWrite(LED_PIN, LOW);
  delay(1000);
}
```

Κώδικας 1: Παράδειγμα κώδικα στο ολοκληρωμένο περιβάλλον ανάπτυξης του Arduino

Οι περισσότερες πλακέτες Arduino περιλαμβάνουν μια δίοδο φωτοεκπομπής LED και μια αντίσταση συνδεδεμένη με την ακίδα 13 και τη γη, προσφέροντας έτσι ένα γρήγορο και βολικό τρόπο απλού ελέγχου λειτουργίας.

Όπως έχει αναφερθεί η πλατφόρμα Arduino χρησιμοποιεί μικροελεγκτές της Atmel, επομένως το λογισμικό μπορεί να αναπτυχθεί και μέσω του ολοκληρωμένου περιβάλλοντος ανάπτυξης που παρέχει η Atmel, το AVR Studio ή το Atmel Studio.

### 3.5 Ανάπτυξη

Όπως έχει ήδη ειπωθεί, το Arduino παρέχει υλικό ανοιχτού κώδικα. Αυτό σημαίνει πως τα σχέδια που καθορίζουν το υλικό είναι ελεύθερα και ανοιχτά διαθέσιμα στο κοινό μέσω του επίσημου διαδικτυακού τόπου, υπό την άδεια Creative Commons Attribution Share-Alike 2.5. Αντίστοιχα, και ο πηγαίος κώδικας του ολοκληρωμένου περιβάλλοντος ανάπτυξης είναι ανοιχτά διαθέσιμος, υπό την άδεια GNU General Public License.

Παρά το γεγονός πως το λογισμικό και το υλικό είναι ελεύθερα διαθέσιμα, η ομάδα ανάπτυξης του Arduino έχει απαιτήσει ο τίτλος “Arduino” να χρησιμοποιείται αποκλειστικά για το επίσημο προϊόν, και να μην χρησιμοποιείται χωρίς την άδεια της.

## 3.6 Arduino Mega 2560

Για την κατασκευή της ηλεκτρικής κλειδαριάς χρησιμοποιήθηκε η πλακέτα Arduino Mega 2560, και συγκεκριμένα η τρίτη έκδοση. Σε αυτήν την ενότητα συνεπώς αναλύεται η πλακέτα Mega 2560.



Εικόνα 10: Η πλακέτα Arduino Mega 2560

### 3.6.1 Γενική Περιγραφή

Το Arduino Mega 2560 είναι μια πλακέτα μικροελεγκτή βασισμένο στον επεξεργαστή ATmega256. Έχει 54 ακίδες ψηφιακής εισόδου – εξόδου, 16 ακίδες αναλογικής εισόδου, 4 σειριακές θύρες UART, έναν κρυσταλλικό ταλαντωτή 16 MHz, σύνδεση USB, υποδοχή εξωτερικής τροφοδοσίας, κεφαλίδα ICSP και ένα κουμπί επαναφοράς, παρέχοντας έτσι μια μεγάλη γκάμα χαρακτηριστικών για την καλύτερη αξιοποίηση του μικροελεγκτή.

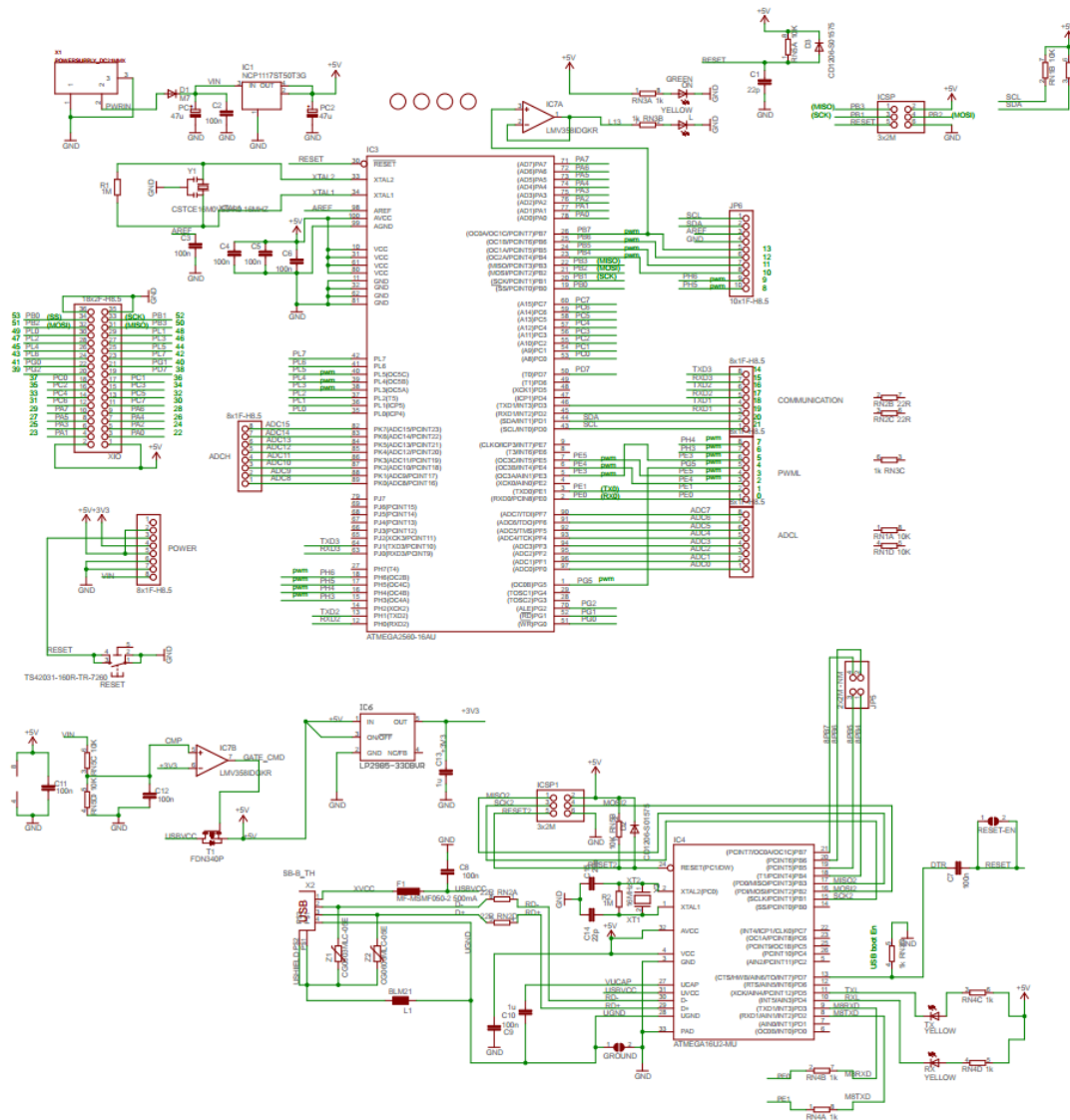
Η πλακέτα Mega 2560 είναι ουσιαστικά ένα εξελιγμένο μοντέλο της κλασικής πλακέτας Mega, και είναι συμβατή με τα περισσότερα σύνολα επέκτασης σχεδιασμένα για το Arduino Duemilavone ή Diecimila. Ωστόσο, παρουσιάζει και σημαντικές διαφορές με τις προηγούμενες εκδόσεις της πλακέτας. Η ουσιαστικότερη διαφορά είναι η χρήση του ATmega16U2 ως μετατροπέας από USB σε σειριακή επικοινωνία.

Η τρίτη έκδοση της πλακέτας Arduino Mega 2560, η οποία είναι και αυτή που χρησιμοποιήσαμε για την υλοποίηση της ηλεκτρικής κλειδαριάς, έχει τα εξής χαρακτηριστικά:

- Διάταξη ακίδων: έχουν προστεθεί οι ακίδες SDA και SCL δίπλα στην ακίδα AREF, έχουν προστεθεί δυο άλλες ακίδες δίπλα στην ακίδα επαναφοράς. Μελλοντικά, τα σύνολα επέκτασης θα είναι συμβατά και με την πλακέτα AVR που λειτουργεί στα 5 Volts αλλά και με το Arduino Due, που λειτουργεί στα 3.3 Volts. Συνεπώς, κάποιες από τις νέες ακίδες θα χρησιμοποιηθούν στο κοντινό μέλλον.
- Ισχυρότερο κύκλωμα επαναφοράς.
- Ο μικροελεγκτής ATmega16U2 έχει αντικαταστήσει τον ATmega8U2.

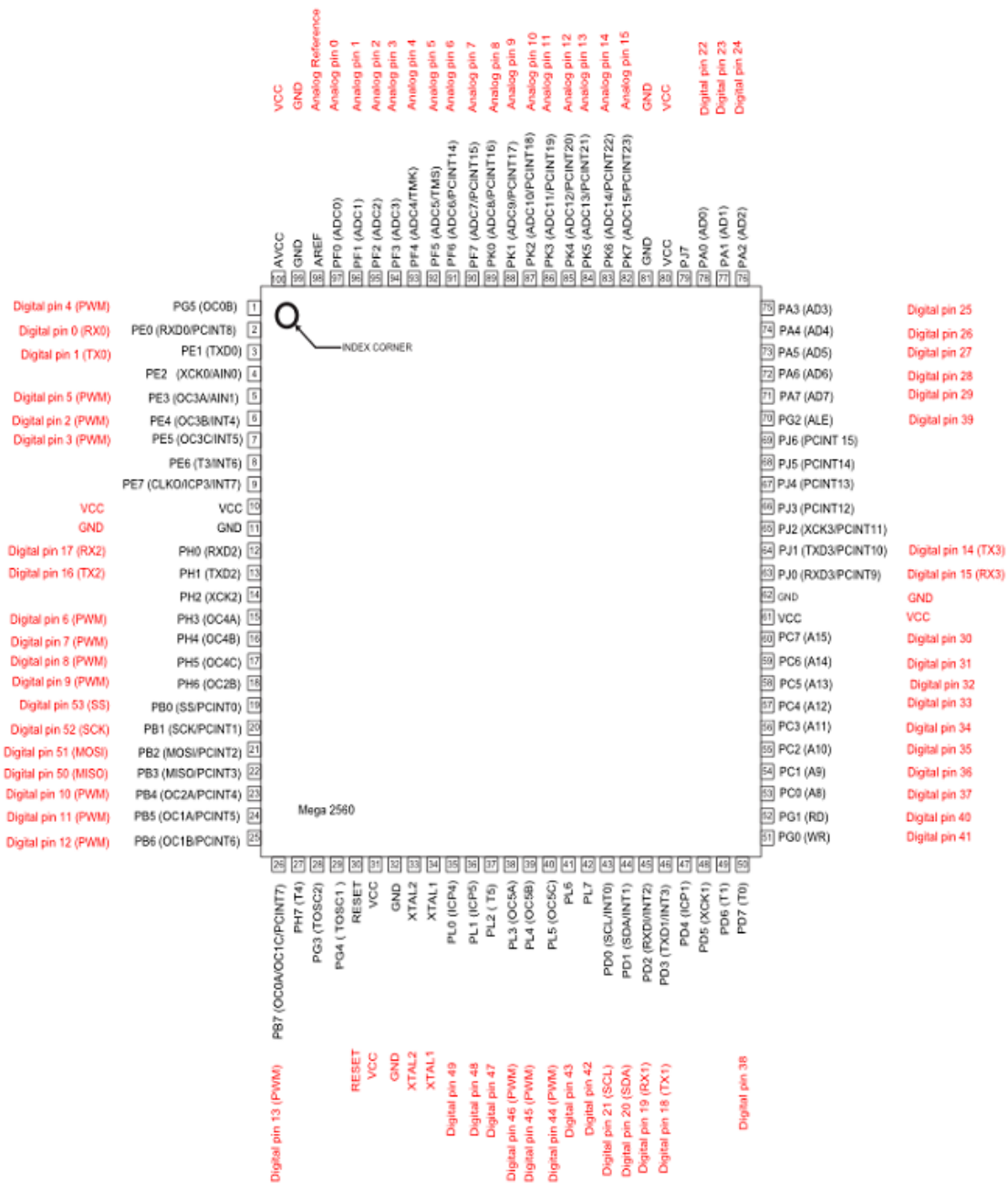
### 3.6.2 Σχεδίαση

Τα σχηματικά διαγράμματα της πλακέτας Arduino Mega 2560 παρουσιάζονται παρακάτω.



Εικόνα 11: Σχηματικά διαγράμματα Arduino Mega 2560

Η πλήρης απεικόνιση των ακίδων φαίνεται στην επόμενη εικόνα.



Εικόνα 12: Πλήρης απεικόνιση των ακίδων του μικροελεγκτή Atmega2560

Συνοπτικά, τα χαρακτηριστικά του Arduino Mega 2560 παρουσιάζονται στον επόμενο πίνακα.

Κατηγορία	Χαρακτηριστικό
Μικροελεγκτής	ATmega2560
Τάση λειτουργίας	5V
Τάση εισόδου (η οποία συνίσταται)	7-12V
Τάση εισόδου (οριακές τιμές)	6-20V
Ακίδες ψηφιακής εισόδου – εξόδου	54
Ακίδες αναλογικής εισόδου	16
Ρεύμα DC ανά ακίδα εισόδου – εξόδου	40 mA
Ρεύμα DC για την ακίδα 3.3V	50 mA
Μνήμη τύπου flash	256 KB, τα 8 KB εκ των οποίων χρησιμοποιούνται για τον κώδικα εκκίνησης.
Μνήμη τύπου SRAM	8 KB
Μνήμη τύπου EEPROM	4 KB
Συχνότητα	16 MHz

Πίνακας 1: Χαρακτηριστικά Arduino Mega 2560

### 3.6.3 Τροφοδοσία

Η πλακέτα Arduino Mega τροφοδοτείται είτε μέσω της σύνδεσης USB ή μέσω εξωτερικής πηγής τροφοδοσίας. Η πηγή της τροφοδοσίας επιλέγεται αυτόματα.

Η εξωτερική τροφοδοσία μπορεί να προέρχεται είτε από μετασχηματιστή AC σε DC, ή από μπαταρία. Ο μετασχηματιστής χρησιμοποιείται με τη σύνδεση ενός βύσματος διαμέτρου 2.1mm στην υποδοχή τροφοδοσίας της πλακέτας. Όσον αφορά την μπαταρία, τα καλώδια της συνδέονται με τις ακίδες **Gnd** και **Vin**.

Η εξωτερική τροφοδοσία πρέπει να κυμαίνεται από 6 Volts ως 20 Volts, ώστε να λειτουργήσει ομαλά η πλακέτα. Ωστόσο, αν η τροφοδοσία είναι μικρότερη από 7 Volts, η ακίδα 5V θα παρέχει λιγότερα από 5 Volts, καθιστώντας την πλακέτα ασταθή. Από την άλλη πλευρά, η χρήση τροφοδοσίας μεγαλύτερης των 12 Volts, μπορεί να οδηγήσει στην υπερθέρμανση του γραμμικού ρυθμιστή και επομένως στην καταστροφή της πλακέτας. Συνεπώς, το εύρος εξωτερικής τροφοδοσίας που συνίσταται κυμαίνεται από 7 Volts ως 12 Volts.

Οι ακίδες τροφοδοσίας είναι οι ακόλουθες:

- **VIN:** υποδεικνύει την εισερχόμενη τάση στην πλακέτα του Arduino, όταν χρησιμοποιείται εξωτερική πηγή τροφοδοσίας (σε αντίθεση με την τάση των 5 Volts που εξασφαλίζεται με τη χρήση USB σύνδεσης). Μέσω αυτής της ακίδας είναι δυνατή η παροχή επιπρόσθετης τάσης.
- **5V:** Αυτή η ακίδα λειτουργεί ως ρυθμιστής τάσης 5V στην πλακέτα. Η πλακέτα μπορεί να τροφοδοτηθεί είτε από την DC υποδοχή τροφοδοσίας (7 – 12 Volts), ή μέσω της σύνδεσης USB (5 Volts), ή μέσω της ακίδας VIN της πλακέτας (7 – 12 Volts). Η παροχή τάσης μέσω των ακίδων 5V ή 3.3V παρακάμπτει το ρυθμιστή, και μπορεί να καταστρέψει την πλακέτα. Συνεπώς, η χρήση αυτής της μεθόδου τροφοδοσίας δε συνίσταται.
- **3V3:** υποδεικνύει την τροφοδοσία 3.3 Volts, η οποία δημιουργείται από το ρυθμιστή της πλακέτας. Το μέγιστο ρεύμα που μπορεί να απορροφηθεί είναι 50 mA.
- **GND:** Οι ακίδες της γείωσης.
- **IOREF:** Αυτή η ακίδα παρέχει την τάση αναφοράς στην οποία λειτουργεί ο μικροελεγκτής. Ένα σωστά ρυθμισμένο σύνολο επέκτασης μπορεί να διαβάσει την τάση της ακίδας IOREF και να επιλέξει την κατάλληλη πηγή τροφοδοσίας.

### 3.6.4 Μνήμη

Ο επεξεργαστής ATmega2560 παρέχει 256 KB μνήμης τύπου flash, για την αποθήκευση του κώδικα, τα 8 KB εκ των οποίων χρησιμοποιούνται για τον κώδικα εκκίνησης. Επίσης, παρέχει 8 KB μνήμης τύπου SRAM και 4 KB μνήμης τύπου EEPROM (η ανάγνωση και η εγγραφή της οποίας πραγματοποιείται με χρήση της βιβλιοθήκης EEPROM).

### 3.6.5 Είσοδος και Έξοδος

Κάθε μια από τις 54 ακίδες της πλακέτας μπορεί να χρησιμοποιηθεί είτε ως είσοδος ή ως έξοδος, με κατάλληλη χρήση των μεθόδων `pinMode()`, `digitalWrite()` και `digitalRead()`.

Οι είσοδοι – έξοδοι της πλακέτας λειτουργούν στα 5 Volts. Κάθε ακίδα παρέχει ή δέχεται ρεύμα μέγιστης τιμής 40 mA, και έχει μια εσωτερική αντίσταση των 20-50 kOhms.

Επιπλέον, κάποιες από τις ακίδες έχουν ειδικές μεθόδους:

- **Serial: 0 (RX)** και **1 (TX)** / **Serial 1: 19 (RX)** και **18 (TX)** / **Serial 2: 17 (RX)** και **16 (TX)** / **Serial 3: 15 (RX)** και **14 (TX)**. Χρησιμοποιούνται για να δεχτούν (RX) ή να μεταδώσουν (TX) σειριακά δεδομένα TTL. Οι ακίδες 0 και 1 είναι επίσης συνδεδεμένες στις αντίστοιχες ακίδες του ATmega16U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), και 21 (interrupt 2)**. Οι ακίδες αυτές ρυθμίζονται έτσι ώστε να πυροδοτούν διακοπές σε μια συγκεκριμένη τιμή, κατά τη διάρκεια αλλαγής της τιμής, κλπ. Η μέθοδος `attachInterrupt()` χρησιμοποιείται εκτενώς για την εκμετάλλευση των διακοπών.

- **PWM: 2 – 13 και 44 – 46.** Οι ακίδες αυτές παρέχουν έξοδο PWM των 8-μπιτ μέσω της μεθόδου `analogWrite()`.
- **SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS).** Οι ακίδες αυτές υποστηρίζουν την SPI επικοινωνία με χρήση της βιβλιοθήκης SPI. Οι ακίδες SPI ορίζονται επίσης στην κεφαλίδα ICSP, η οποία είναι συμβατή με τα μοντέλα Uno, Duemilanove και Diecimila.
- **LED: 13.** Υπάρχει μια ενσωματωμένη δίοδος φωτοεκπομπής (LED), συνδεδεμένη στην ακίδα 13. Όταν η τιμή της ακίδας είναι `HIGH`, η δίοδος είναι ενεργή, ενώ όταν η τιμή είναι `LOW`, η δίοδος είναι ανενεργή.
- **TWI: 20 (SDA) και 21 (SCL).** Παρέχουν υποστήριξη της TWI επικοινωνίας, μέσω της βιβλιοθήκης `Wire`.

Όπως έχει ήδη αναφερθεί, η πλακέτα Arduino Mega 2560 δα διαθέτει 16 ακίδες αναλογικής εισόδου, κάθε μια από τις οποίες ορίζεται από συνολικά 10 μπιτ, δηλαδή  $2^{10}=1024$  διαφορετικές τιμές. Από προεπιλογή, οι τιμές ορίζονται ως οι μετρήσεις από τη γείωση ως τα 5 Volts. Ωστόσο, είναι δυνατή η αλλαγή του εύρους μετρήσεων, τόσο το πάνω όριο όσο και το κάτω όριο. Αυτό επιτυγχάνεται μέσω της ακίδας AREF και της μεθόδου `analogReference()`.

Κάποιες άλλες ακίδες που είναι διαθέσιμες στην πλακέτα είναι οι εξής:

- **AREF:** Υποδηλώνει την τάση αναφορά για τις αναλογικές εισόδους και χρησιμοποιείται με τη μέθοδο `analogReference()`.
- **Reset:** Υποδηλώνει την επαναφορά του συστήματος.

### 3.6.6 Επικοινωνία

Η πλακέτα Arduino Mega 2560 διαθέτει μια σειρά από τρόπους επικοινωνίας με υπολογιστή, με άλλες πλακέτες Arduino, ή άλλους μικροελεγκτές. Ο ελεγκτής ATmega2560 παρέχει τέσσερις πόρτες UART για σειριακή επικοινωνία TTL (5V). Ένας μικροελεγκτής ATmega16U2 στην πλακέτα διοχετεύει μια από αυτές μέσω USB σύνδεσης και παρέχει μια εικονική θύρα COM για το λογισμικό στον υπολογιστή (τα συστήματα με λογισμικό Windows χρειάζονται ένα `.inf` αρχείο, ενώ τα συστήματα με λογισμικό OSX και Linux αναγνωρίζουν την πλακέτα σαν μια θύρα COM αυτόματα).

Το λογισμικό της πλακέτας Arduino περιλαμβάνει μια σειριακή οθόνη η οποία επιτρέπει την αποστολή απλών δεδομένων κειμένου να αποστέλλονται από και προς την πλακέτα. Οι δίοδοι φωτοεκπομπής (LED) RX και TX που βρίσκονται στην πλακέτα αναβοσβήνουν όταν μεταδίδονται δεδομένα μέσω του ολοκληρωμένου ATmega16U2 και της USB σύνδεσης στον υπολογιστή (αλλά όχι στην περίπτωση σειριακής επικοινωνίας στις ακίδες 0 και 1).

Η βιβλιοθήκη `SoftwareSerial` επιτρέπει την σειριακή επικοινωνία σε οποιαδήποτε ψηφιακή ακίδα της πλακέτας Mega2560. Η πλακέτα ATmega2560 υποστηρίζει επιπλέον επικοινωνία TWI και SPI. Το λογισμικό της πλακέτας Arduino περιλαμβάνει μια



βιβλιοθήκη [Wire](#) για την απλοποίηση της χρήσης της διαύλου TWI. Για επικοινωνία SPI, χρησιμοποιείται η βιβλιοθήκη [SPI](#).

### 3.6.7 Προγραμματισμός

Ο προγραμματισμός της πλακέτας Arduino Mega μπορεί να πραγματοποιηθεί μέσω του ολοκληρωμένου περιβάλλοντος ανάπτυξης που παρέχει το Arduino, όπως έχει ήδη συζητηθεί. Ο επίσημος διαδικτυακός τόπος του Arduino παρέχει πληθώρα βοηθητικών οδηγιών και τεκμηρίωσης για καλύτερη κατανόηση του λογισμικού.

Στο μικροελεγκτή ATmega2560 έχει ήδη φορτωθεί ένας κώδικας εκκίνησης, ο οποίος επιτρέπει στον χρήστη να ανεβάσει και να τρέξει κάποιον κώδικα στην πλακέτα, χωρίς να απαιτείται η χρήση εξωτερικού προγραμματιστή λογισμικού. Ωστόσο, είναι δυνατή η παράκαμψη του κώδικα εκκίνησης και ο προγραμματισμός του μικροελεγκτή μέσω των ICSP κεφαλίδων.

Όσον αφορά το μικροελεγκτή ATmega16U2, ο πηγαίος κώδικας υλικολογισμικού είναι αποθηκευμένος και διαθέσιμο στον επίσημο διαδικτυακό τόπο του Arduino. Ο μικροελεγκτής ATmega16U2 είναι προ-φορτωμένος με κώδικα εκκίνησης DFU, το οποίο ενεργοποιείται μέσω μια αντίστασης η οποία βρίσκεται μεταξύ του 16U2 HWB και της γης. Με χρήση της κατάλληλης τεχνικής, π.χ. λογισμικό FLIP από την Atmel ή χρήση των ISP κεφαλίδων, πραγματοποιείται η φόρτωση νέου υλικολογισμικού.

### 3.6.8 Αυτόματη Επαναφορά

Η πλακέτα Arduino Mega 2560 έχει σχεδιαστεί έτσι ώστε να είναι δυνατή η επαναφορά της μέσω λογισμικού που τρέχει σε ένα συνδεδεμένο υπολογιστή. Μια γραμμή ελέγχου ροής (DTR) του μικροελεγκτή ATmega8U2 είναι συνδεδεμένη στη γραμμή επαναφοράς της πλακέτας ATmega2560 μέσω ενός πυκνωτή 100 nF. Όταν αυτή η γραμμή οδηγείται χαμηλά, η γραμμή επαναφοράς πέφτει για αρκετό χρονικό διάστημα ώστε να γίνει επαναφορά του ολοκληρωμένου. Το λογισμικό της πλακέτας Arduino χρησιμοποιεί αυτή τη δυνατότητα για να επιτρέπει στο χρήστη να ανεβάζει τον κώδικά του πατώντας απλά το αντίστοιχο κουμπί στο ολοκληρωμένο περιβάλλον ανάπτυξης. Αυτό σημαίνει ότι ο κώδικας εκκίνησης μπορεί να έχει μικρότερο χρονικό όριο, καθώς η μείωση της γραμμής ελέγχου ροής (DTR) μπορεί να είναι καλά συντονισμένη με την έναρξη της φόρτωσης.

Ωστόσο, αυτή η ρύθμιση έχει διάφορες συνέπειες. Όταν η πλακέτα Mega 2560 είναι συνδεδεμένη είτε με ένα υπολογιστή που τρέχει λογισμικό Mac OSX ή Linux, τότε γίνεται επαναφορά κάθε φορά που πραγματοποιείται σύνδεση με το λογισμικό μέσω USB. Για περίπου μισό δευτερόλεπτο, ο κώδικας εκκίνησης τρέχει στην πλακέτα Mega 2560. Ενώ η πλακέτα είναι προγραμματισμένη να αγνοεί κακά διαμορφωμένα δεδομένα (δηλαδή οτιδήποτε εκτός της φόρτωσης νέου κώδικα), θα παρεμποδίσει τα πρώτα byte δεδομένων που θα αποσταλούν στην πλακέτα αφού αποκατασταθεί η σύνδεση.

Σε περίπτωση που ο κώδικας που τρέχει στην πλακέτα λάβει μια ρύθμιση ή άλλα δεδομένα όταν ξεκινήσει, τότε θα πρέπει το λογισμικό με το οποίο επικοινωνεί να περιμένει ένα δευτερόλεπτο αφού γίνει αποκατάσταση της σύνδεσης πριν στείλει αυτά τα δεδομένα.

Η πλακέτα Mega2560 περιέχει ένα ίχνος, το οποίο μπορεί να αποκοπεί για να απενεργοποιηθεί η αυτόματη επαναφορά. Οι επαφές σε κάθε πλευρά του ίχνους μπορούν να συγκολληθούν για να ενεργοποιηθεί ξανά. Είναι επίσης δυνατό να απενεργοποιηθεί η αυτόματη επαναφορά συνδέοντας μια αντίσταση 110 Ohm στη γραμμή επαναφοράς.

### 3.6.9 Προστασία από Υπερένταση

Η πλακέτα Arduino Mega 2560 παρέχει μια επαναφερόμενη ασφάλεια, η οποία προστατεύει τις θύρες USB του υπολογιστή από βραχυκυκλώματα και υπερεντάσεις. Αν και οι περισσότεροι υπολογιστές διαθέτουν εσωτερική προστασία για τέτοια φαινόμενα, η ασφάλεια της πλακέτα παρέχει ένα επιπλέον επίπεδο προστασίας. Σε περίπτωση που εφαρμοσθούν περισσότερα από 500 mA στη θύρα USB, η ασφάλεια θα διακόψει αυτόματα τη σύνδεση μέχρι το φαινόμενο που ενεργοποίησε την ασφάλεια να πάψει να υφίσταται.

### 3.6.10 Φυσικά Χαρακτηριστικά και Συμβατότητα

Το μέγιστο μήκος της πλακέτας Mega 2560 είναι 101.6 mm, και το μέγιστο πλάτος είναι 50.8 mm. Ωστόσο, η υποδοχή σύνδεσης USB και η υποδοχή τροφοδοσίας επεκτείνουν λίγο το μήκος της πλακέτας. Η πλακέτα παρέχει τρεις τρύπες, στις οποίες μπορούν να τοποθετηθούν βίδες, ώστε να μπορεί η πλακέτα να προσαρμοστεί σε οποιαδήποτε επιφάνεια.

Σχετικά με τη συμβατότητα, η πλακέτα Mega 2560 έχει σχεδιασθεί ώστε να είναι συμβατή με τα περισσότερα σύνολα επέκτασης για τα μοντέλα Uno, Diecimila και Duemilanove. Πιο συγκεκριμένα, οι ακίδες ψηφιακής εισόδου – εξόδου 0 ως 13, οι ακίδες AREF και GND, οι ακίδες αναλογικής εισόδου 0 ως 5, η κεφαλίδα παροχής, η κεφαλίδα ICSP βρίσκονται σε ισοδύναμες θέσεις. Επιπλέον, η κύρια θύρα UART βρίσκεται στις ίδιες ακίδες (0 και 1).

## 4 Παραγωγή Συστήματος

Ο κύκλος ζωής ενός συστήματος, είτε αυτό είναι λογισμικό, υλικό ή συνδυασμός των δυο, ορίζεται από διάφορα στάδια, τα οποία είναι σαφώς ορισμένα. Η διάκριση των σταδίων έχει ως σκοπό στην παραγωγή συστημάτων υψηλής ποιότητας, τα οποία ανταποκρίνονται πλήρως στις ανάγκες των χρηστών τους. Αυτό το μοντέλο παραγωγής είναι γνωστό ως SDLC (Systems Development Life Cycle) και αποτελεί βάση για διάφορες μεθοδολογίες ανάπτυξης (Pendharkara, et al., 2008; Avison & Shah, 1997).

Τα στάδια του μοντέλου SDLC, στα οποία θα βασιστούμε για τη δημιουργία της ηλεκτρονικής κλειδαριάς, είναι τα εξής:

- **Προδιαγραφές συστήματος:** κατά το στάδιο αυτό πραγματοποιείται ανάλυση και ορισμός των απαιτήσεων και εξαγωγή των προδιαγραφών του συστήματος.
- **Σχεδίαση:** έχοντας ως βάση τις σαφώς ορισμένες προδιαγραφές του συστήματος, έπεται το στάδιο της σχεδίασης, όπου το σύστημα σχεδιάζεται με μεγάλη λεπτομέρεια και ακρίβεια.
- **Υλοποίηση:** κατά το στάδιο της υλοποίησης συγγράφεται ο απαραίτητος κώδικας, προγραμματίζεται και ενεργοποιείται το σύστημα.
- **Έλεγχος:** κατά το στάδιο του ελέγχου, ελέγχεται τόσο η λειτουργία και η συμπεριφορά του συστήματος, όσο και το αν πληροί τις προδιαγραφές. Μετά το στάδιο ελέγχου, πραγματοποιούνται αλλαγές, εφόσον απαιτούνται, και το σύστημα επανελέγχεται.
- **Κατασκευή:** Εφόσον το σύστημα μας έχει ελεγχθεί σε βάθος, ακολουθεί το στάδιο της κατασκευής της ηλεκτρονικής κλειδαριάς, όπου είναι και το τελικό στάδιο στο πλαίσιο αυτής της πτυχιακής εργασίας.

### 4.1 Προδιαγραφές Συστήματος

Πριν τη σχεδίαση ενός συστήματος, ανεξαρτήτως της φύσης του, θα πρέπει τόσο οι στόχοι όσο και οι ανάγκες που καλύπτει το σύστημα να είναι σαφώς ορισμένα. Το στάδιο αυτό καλείται ανάλυση και προσδιορισμός των απαιτήσεων και των προδιαγραφών του συστήματος (Pohl, 2010).

Όπως είναι εύκολα κατανοητό, η ανάλυση των απαιτήσεων είναι ένα από τα κρίσιμα στάδια κατά τη διαδικασία δημιουργίας ενός συστήματος, και κατά ένα μεγάλο ποσοστό, ορίζει την τελική επιτυχία του συστήματος. Προκειμένου η διαδικασία σχεδίασης και υλοποίησης ενός συστήματος να είναι επιτυχής, θα πρέπει οι απαιτήσεις και οι προδιαγραφές να είναι καλά τεκμηριωμένες, μετρήσιμες, επαληθεύσιμες, ανιχνεύσιμες και σαφώς ορισμένες (Wohlin, et al., 2012).

Οι απαιτήσεις ενός συστήματος κατηγοριοποιούνται ανάλογα με τον σκοπό που εξυπηρετούν, π.χ. λειτουργική συμπεριφορά του συστήματος, τύποι χρηστών, κλπ. (Wieggers & Beatty, 2013). Στο υπόλοιπο μέρος αυτής της ενότητας πραγματοποιείται η ανάλυση των απαιτήσεων και η εξαγωγή των προδιαγραφών του συστήματος της ηλεκτρικής κλειδαριάς.

### 4.1.1 Τύποι Χρηστών

Όσοι έχουν φυσική πρόσβαση στο σύστημα της ηλεκτρονικής κλειδαριάς είναι και χρήστες του συστήματος. Ωστόσο, το ξεκλείδωμα της ηλεκτρονικής κλειδαριάς πραγματοποιείται μόνο από διαπιστευμένους χρήστες καθώς και από άτομα που βρίσκονται στην εσωτερική πλευρά του χώρου. Ως διαπιστευμένοι χρήστες θεωρούνται όσοι εισάγουν το σωστό κωδικό, ο οποίος ενεργοποιεί το ξεκλείδωμα της κλειδαριάς.

### 4.1.2 Περιορισμοί σχεδίασης και υλοποίησης

Όσον αφορά το λογισμικό, οι περιορισμοί έγκεινται στο ολοκληρωμένο περιβάλλον ανάπτυξης και τη γλώσσα προγραμματισμού. Εφόσον ο προγραμματισμός γίνεται σε πλακέτα Arduino το ολοκληρωμένο περιβάλλον ανάπτυξης θα πρέπει να είναι κάποιο από τα συμβατά. Αυτό που επιλέξαμε για αυτήν την πτυχιακή εργασία είναι το περιβάλλον που παρέχεται από τον επίσημο διαδικτυακό τόπο του Arduino. Αντίστοιχα, η γλώσσα προγραμματισμού περιορίζεται από το Arduino, και συνεπώς η επιλεγμένη γλώσσα είναι η C/C++.

Όσον αφορά το υλικό, οι περιορισμοί αφορούν την πλακέτα Arduino Mega 2560, τις δυνατότητές της, τα χαρακτηριστικά της και τα συμβατά στοιχεία επέκτασης και ελέγχου, όπως αυτά ορίζονται από την επίσημη τεκμηρίωση.

Εκτός από το λογισμικό και το υλικό, περιορισμοί μπορούν να εντοπισθούν και στην κατασκευή της κλειδαριάς. Συγκεκριμένα, η εφαρμογή της κλειδαριάς θα μπορούσε να γίνει σε οποιοδήποτε χώρο όπου είναι δυνατή η είσοδος ή έξοδος. Ωστόσο, στα πλαίσια αυτής της πτυχιακής εργασίας, θα κατασκευασθεί μια πόρτα μικρής κλίμακας, ώστε να ευνοείται η φορητότητα και ο έλεγχος, ενώ παράλληλα εξομοιώνει τη λειτουργία μιας πραγματικής πόρτας.

### 4.1.3 Λειτουργία συστήματος

Η λειτουργία του συστήματος αφορά τη συνολική συμπεριφορά του συστήματος και συνοψίζεται στις επόμενες υπο-ενότητες.

Για την ανάλυση των λειτουργικών απαιτήσεων θα χρησιμοποιήσουμε το μοντέλο της Θεωρίας Δραστηριοτήτων, το οποίο θα μας βοηθήσει να κατανοήσουμε σε βάθος τις ενέργειες που απαιτούνται, τόσο από τον χρήστη όσο και από το σύστημα, κατά τη διάρκεια μιας δραστηριότητας.

Επίσης, θα αντιληφθούμε ποιοι είναι οι στόχοι των χρηστών κατά τη διάρκεια μιας δραστηριότητας, τι κάνουν για να τους επιτύχουν, για ποιο λόγο εκτελούν συγκεκριμένες ενέργειες, και τι σημαίνουν αυτές οι ενέργειες για τους χρήστες (Kaptelinin & Nardi, 2012). Συνεπώς, η Θεωρία Δραστηριοτήτων είναι το κατάλληλο πλαίσιο, το οποίο θα χρησιμοποιήσουμε ώστε να αναλύσουμε σε βάθος και με μεγάλη λεπτομέρεια τις λειτουργικές απαιτήσεις του συστήματος της ηλεκτρονικής κλειδαριάς.

#### 4.1.3.1 Είσοδος χρήστη

Σε περίπτωση που ένας χρήστης θέλει να εισέλθει στο χώρο που ασφαλίζεται με την ηλεκτρονική κλειδαριά, θα πρέπει να του δοθεί η κατάλληλη εξουσιοδότηση ώστε να του επιτραπεί η είσοδος.

Υπάρχουν διάφοροι τρόποι ελέγχου και ταυτοποίησης του χρήστη, όπως έχει ήδη συζητηθεί, ωστόσο, για την παρούσα πτυχιακή εργασία, ο έλεγχος του διαπιστευμένου χρήστη γίνεται μέσω της πληκτρολόγησης κωδικού.

Συνεπώς, η διαδικασία ελευθέρωσης της κλειδαριάς από ένα διαπιστευμένο χρήστη είναι η εξής:

Ερέθισμα	Ο χρήστης βρίσκεται μπροστά στην κλειδαριά.
Αντίδραση	Του ζητείται η εισαγωγή κωδικού, μέσω μιας οθόνης.
Ερέθισμα	Πληκτρολογεί τον κωδικό, και ύστερα ζητά την ελευθέρωση της κλειδαριάς.
Αντίδραση	Το σύστημα ελέγχει αν ο κωδικός είναι ίδιος με τον αποθηκευμένο κωδικό. Μετά την επιβεβαίωση του κωδικού, η κλειδαριά ελευθερώνεται, ενώ παράλληλα ο χρήστης ενημερώνεται μέσω μηνύματος στην οθόνη για την επιτυχή εισαγωγή του κωδικού. Παράλληλα μια ηχητική και μια οπτική ένδειξη ενεργοποιούνται ώστε καταστήσουν σαφές στο χρήστη ότι του επιτρέπεται η είσοδος.

Σε περίπτωση εισαγωγής λανθασμένου κωδικού, η αντίδραση του συστήματος αλλάζει. Πιο συγκεκριμένα, η κλειδαριά παραμένει ασφαλισμένη, ενώ παράλληλα κατάλληλη ηχητική και οπτική ένδειξη ενεργοποιείται ώστε να ενημερώσουν το χρήστη για την εισαγωγή εσφαλμένου κωδικού. Ο χρήστης ενημερώνεται επίσης μέσω μηνύματος στην οθόνη.

Η οπτική ένδειξη για την ενημέρωση του χρήστη θα μπορούσε μια δίοδος φωτοεκπομπής (LED) κόκκινου ή πράσινου χρώματος, ώστε να συνδέει άμεσα και μηχανικά ο χρήστης την ενεργεία του ως εσφαλμένη/ ανεπιτυχή ή ορθή/ επιτυχημένη, αντίστοιχα. Η ηχητική ένδειξη θα μπορούσε να είναι συγκεκριμένος τόνος, ο οποίος θα διαφέρει ανάλογα με το αποτέλεσμα της ενέργειας του χρήστη, όπως ακριβώς συμβαίνει και με την οπτική ένδειξη.

Σε αυτό το σημείο θεωρούμε πως πρέπει η ασφάλεια του συστήματος να θωρακισθεί με ένα επιπλέον επίπεδο προστασίας. Πιο συγκεκριμένα, ο χρήστης ο οποίος εισάγει λανθασμένο κωδικό πολλαπλές φορές, θα πρέπει να αποθαρρύνεται. Συνεπώς, ο μηχανισμός εισαγωγής κωδικού απενεργοποιείται μετά από τρεις εσφαλμένες προσπάθειες, και ενεργοποιείται αυτόματα μετά την παρέλευση συγκεκριμένου χρονικού διαστήματος, π.χ. τρία λεπτά.

Επίσης, για την μεγαλύτερη ασφάλεια των χρηστών, θα πρέπει να μην εμφανίζεται ο εισαγόμενος κωδικός στην οθόνη, αλλά ο χρήστης θα πρέπει να μπορεί να βλέπει ότι πληκτρολογεί κάποιο αλφαριθμητικό στοιχείο, ώστε να διατηρηθεί η ευχρηστία του συστήματος. Συνεπώς, η χρήση ειδικών χαρακτήρων ενδείκνυται κατά την εισαγωγή του κωδικού.

#### 4.1.3.2 Ενεργοποίηση/απενεργοποίηση οθόνης

Η οθόνη θα πρέπει να μένει ενεργή ή ανενεργή, ανάλογα με το αν είναι παρών κάποιος χρήστης ή όχι, αντίστοιχα. Με αυτόν τον τρόπο θα εξοικονομείται ενέργεια, καθιστώντας το σύστημα πιο αποδοτικό. Συνεπώς, η ύπαρξη ενός μηχανισμού ανίχνευσης κίνησης είναι απαραίτητη.

#### 4.1.3.3 Ρύθμιση φωτεινότητας οθόνης

Ειδικός μηχανισμός ρύθμισης της φωτεινότητας της οθόνης θα πρέπει να παρέχεται στον χρήστη, ώστε να μπορεί να διαμορφώσει κατάλληλα τη φωτεινότητα σύμφωνα με την προσωπική του προτίμηση ή τις διάφορες καιρικές συνθήκες, π.χ. έντονα ηλιόλουστη μέρα ή σκοτάδι.

#### 4.1.3.4 Ελευθέρωση κλειδαριάς από το εσωτερικό

Ο χρήστης ο οποίος βρίσκεται στον εσωτερικό χώρο, θα πρέπει να μπορεί να ελευθερώσει την κλειδαριά και να ανοίξει την πόρτα χωρίς να χρειάζεται να περάσει κάποιον έλεγχο. Συνεπώς, δεν απαιτείται η ύπαρξη ηλεκτρολογίου στο εσωτερικό μέρος. Αντί αυτού, ένα απλό κουμπί είναι αρκετό.

Να σημειωθεί ότι ο μηχανισμός ελευθέρωσης της κλειδαριάς από το εσωτερικό, παρακάμπτει και επαναφέρει στην αρχική κατάσταση το μηχανισμό εισαγωγής κωδικού. Η ροή είναι η εξής:

Ερέθισμα	Ο χρήστης πατάει το κουμπί απελευθέρωσης από το εσωτερικό.
Αντίδραση	Η κλειδαριά ελευθερώνεται και η πόρτα ανοίγει. Κατάλληλη οπτική και ηχητική ένδειξη συνοδεύει την ενέργεια αυτή, ενώ ο μηχανισμός εισαγωγής κωδικού επανέρχεται στην αρχική κατάσταση.

#### 4.1.3.5 Κουδούνι

Ο χρήστης που βρίσκεται στο εξωτερικό μέρος θα πρέπει να έχει τη δυνατότητα να ειδοποιεί τους παρευρισκόμενους στον εσωτερικό χώρο, ώστε να του επιτρέπεται η είσοδος, ακόμα κι αν δεν είναι διαπιστευμένο μέλος του συστήματος. Η ειδοποίηση αυτή γίνεται με το πάτημα ενός κουμπιού, και την αναπαραγωγή ήχου από το σύστημα, εξομοιώνοντας τη λειτουργία του κουδουνιού.

Ερέθισμα	Ο χρήστης πατάει το κουμπί ειδοποίησης (κουδούνι)
Αντίδραση	Το σύστημα αναπαράγει ηχητικό σήμα για όσο χρόνο ο χρήστης κρατά πατημένο το κουμπί, εξομοιώνοντας τη λειτουργία του κουδουνιού.

#### 4.1.3.6 Ειδοποίηση ανοικτής πόρτας

Σε περίπτωση που η πόρτα είναι ανοικτή, οι χρήστες του συστήματος θα πρέπει να ειδοποιούνται, ώστε να παρέχει το σύστημα μεγαλύτερη ασφάλεια. Για το λόγο αυτό, τόσο οπτική όσο και ηχητική ένδειξη θα πρέπει να ενεργοποιούνται, και να παραμένουν ενεργές όσο διάστημα παραμένει ανοικτή η πόρτα. Συνεπώς, είναι απαραίτητη η ύπαρξη ενός μηχανισμού ανίχνευσης της κατάστασης της πόρτας (ανοικτή ή κλειστή), ο οποίος θα βασίζεται σε μαγνητική διάταξη.

#### 4.1.3.7 Ενεργοποίηση/Απενεργοποίηση ήχου

Σύμφωνα με αυτά που περιγράφηκαν πιο πάνω, είναι απαραίτητη η αναπαραγωγή συγκεκριμένου χρόνου όταν η πόρτα παραμένει ανοικτή, ώστε να ειδοποιούνται οι χρήστες του συστήματος. Κάτι τέτοιο όμως μπορεί να μην είναι επιθυμητό από τους χρήστες για διάφορους λόγους, π.χ. μαζική προσέλευση ατόμων στον εσωτερικό χώρο, όταν και η παρατεταμένη αναπαραγωγή μπορεί να προκαλέσει εκνευρισμό. Συνεπώς, ο χρήστης στο εσωτερικό του χώρου θα πρέπει να μπορεί να ενεργοποιεί ή να απενεργοποιεί την αναπαραγωγή του ήχου.

Ερέθισμα	Ο χρήστης του εσωτερικού χώρου θέτει το διακόπτη σε θέση 1 (ή 0) και η πόρτα είναι ανοικτή.
Αντίδραση	Το ηχητικό σήμα που συνοδεύει την κατάσταση ανοικτής πόρτας αναπαράγεται (ή δεν αναπαράγεται).

#### 4.1.3.8 Μηχανισμός ελευθέρωσης πόρτας

Η ελευθέρωση της πόρτας θα πραγματοποιείται μέσω ηλεκτρικού κυπρί, το οποίο θα ενεργοποιείται μέσα από το σύστημα της ηλεκτρονικής κλειδαριάς, όταν ο χρήστης εισάγει το σωστό κωδικό εισόδου ή όταν πατηθεί το κουμπί εισόδου από το εσωτερικό.

#### 4.1.3.9 Διόρθωση εισαγόμενου κωδικού

Κατά τη διάρκεια εισαγωγής κωδικού, είναι πιθανό ο χρήστης να κάνει λάθος καθώς πληκτρολογεί κάποιο αλφαριθμητικό στοιχείο. Συνεπώς, το σύστημα θα πρέπει να διαθέτει μηχανισμό διόρθωσης κωδικού, όπου ο μέχρι τότε εισαγόμενος κωδικός θα διαγράφεται, επιτρέποντας στο χρήστη να ξεκινήσει τη διαδικασία εισαγωγής κωδικού από την αρχή.

Ερέθισμα	Ο χρήστης πατά κάποιο πλήκτρο κατά λάθος, και στη συνέχεια πατά το ειδικό κουμπί διόρθωσης κωδικού.
Αντίδραση	Το σύστημα διαγράφει τον κωδικό που είχε εισάγει ως τότε ο χρήστης. Ο χρήστης ενημερώνεται για αυτό μέσω μηνύματος στην οθόνη καθώς και μέσω αναπαραγωγής ηχητικού σήματος.

#### 4.1.3.10 Αλλαγή κωδικού

Η ύπαρξη μηχανισμού αλλαγής κωδικού είναι απαραίτητη ώστε ο χρήστης να είναι σε θέση να χρησιμοποιεί έναν προσωποποιημένο κωδικό, αντί για τον προεπιλεγμένο. Η ροή αλλαγής κωδικού είναι η εξής:

Ερέθισμα	Ο χρήστης επιθυμεί να αλλάξει τον κωδικό, και πατά το αντίστοιχο πλήκτρο.
Αντίδραση	Το σύστημα ζητά από τον χρήστη να πληκτρολογήσει τον παλιό κωδικό, ώστε να επαληθεύσει το επίπεδο διαπίστευσης του χρήστη.
Ερέθισμα	Ο χρήστης πληκτρολογεί τον παλιό κωδικό, και πατά το πλήκτρο εισαγωγής.
Αντίδραση	Το σύστημα ελέγχει τον κωδικό: <ul style="list-style-type: none"><li>- Αν είναι σωστός, μεταφέρει το χρήστη στο επίπεδο στάδιο, ζητώντας από τον χρήστη να εισάγει το νέο κωδικό (περιγράφεται στην αμέσως επόμενη γραμμή).</li><li>- Αν είναι λανθασμένος, ειδοποιεί τον χρήστη και τον επαναφέρει στην αρχική οθόνη και κατάσταση.</li></ul>
Ερέθισμα	Ο χρήστης πληκτρολογεί το νέο κωδικό, και πατά το πλήκτρο εισαγωγής.
Αντίδραση	Το σύστημα κάνει την αλλαγή των κωδικών και επιστρέφει στην αρχική κατάσταση και οθόνη, ενημερώνοντας κατάλληλα τον χρήστη.



#### 4.1.3.11 Προεπιλεγμένη κατάσταση

Η προεπιλεγμένη κατάσταση του συστήματος είναι ενεργή μόνο την πρώτη φορά λειτουργίας του συστήματος και περιλαμβάνει την χρήση του κωδικού «1234», την ενεργοποίηση της κόκκινης φωτεινής ένδειξης (σε περίπτωση που ο μηχανισμός ανίχνευσης κατάστασης της πόρτας είναι σε θέση LOW) και την μη-αναπαραγωγή ήχου (σε περίπτωση που ο μηχανισμός ανίχνευσης κατάστασης της πόρτας είναι σε θέση LOW).

#### 4.1.4 Άλλες απαιτήσεις

Σε αυτήν την ενότητα παρουσιάζονται απαιτήσεις που δεν ομαδοποιούνται σε καμιά από τις προηγούμενες ενότητες.

##### 4.1.4.1 Απόδοση

Όσον αφορά την απόδοση του συστήματος της ηλεκτρονικής κλειδαριάς, σημαντικός παράγοντας είναι η απόκριση του συστήματος. Έχοντας ως δεδομένο το γεγονός πως μόνο ένας χρήστης μπορεί να εξυπηρετηθεί κάθε φορά από το σύστημά μας, ο χρόνος απόκρισης θα πρέπει να είναι μικρότερος του ενός δευτερολέπτου, ώστε να μην προκαλείται σύγχυση και εκνευρισμός στο χρήστη. Σε περίπτωση που το χρονικό διάστημα της απόκρισης είναι μεγαλύτερο, ο χρήστης θα πρέπει ενημερώνεται από το σύστημα, π.χ. κατάλληλο μήνυμα στην οθόνη (Nielsen, 1994).

##### 4.1.4.2 Ευχρηστία

Η ευχρηστία ενός συστήματος ορίζεται ως ο βαθμός ευκολίας και ευχαρίστησης χρήσης του διαδραστικού συστήματος, και θα πρέπει να λαμβάνεται υπόψη κατά τη σχεδίασης.

Υπάρχουν διάφορες οδηγίες σχεδίασης ευχρηστίας, ωστόσο οι πιο γνωστές, οι αποτελούν και βάση για άλλες, είναι δέκα οδηγίες όπως ορίστηκαν από τον Nielsen (1994). Συνοπτικά, είναι οι εξής:

- Το σύστημα πρέπει να κρατά το χρήστη ενήμερο σχετικά με τη λειτουργία του.
- Τα μηνύματα προς το χρήστη θα πρέπει να είναι απλά και κατανοητά, στη γλώσσα του χρήστη.
- Ο χρήστης θα πρέπει να είναι σε θέση να καταλαβαίνει και να μαθαίνει από τα λάθη του, κατά τη διάρκεια αλληλεπίδρασης με το σύστημα.
- Στο σύστημα θα πρέπει να επικρατεί συνοχή, π.χ. ίδιες λειτουργίες για ίδιες ενέργειες.
- Τα μηνύματα σφάλματος θα πρέπει να είναι απλά και να μην αποθαρρύνουν το χρήστη.
- Το φορτίο μνήμης του χρήστη θα πρέπει να είναι ελάχιστο κατά τη διάρκεια της αλληλεπίδρασης.
- Το σύστημα θα πρέπει να παρέχει εργαλεία και λειτουργίες τόσο για άπειρους όσο και για έμπειρους χρήστες.
- Ο συνολικός σχεδιασμός θα πρέπει να είναι μινιμαλιστικός και αισθητικός.

- Θα πρέπει να παρέχεται βοήθεια προς το χρήστη σε οποιοδήποτε στάδιο της αλληλεπίδρασης.

#### 4.1.4.3 Εκτίμηση κόστους

Μια πολύ σημαντική παράμετρος για την ολοκλήρωση του έργου που παρουσιάζεται σε αυτήν την πτυχιακή εργασία, είναι η εκτίμηση του κόστους. Αυτή πραγματοποιείται λαμβάνοντας υπόψη όλα τα υλικά που θα χρειαστούμε, καθώς και εργατοώρες τρίτων, εφόσον χρειάζονται. Στα πλαίσια αυτής της πτυχιακής εργασίας, τα έξοδα αφορούν αποκλειστικά την αγορά υλικών και αποτυπώνονται στο κοστολόγιο, το οποίο βρίσκεται στην επόμενη ενότητα.

#### 4.1.5 Συγκέντρωση προδιαγραφών

Στην παρακάτω λίστα εμφανίζονται οι τελικές προδιαγραφές του συστήματος, όπως αυτές προέκυψαν από την ανάλυση των απαιτήσεων:

- ΠΡ 1. Χρήστες του συστήματος είναι όλοι όσοι έχουν φυσική πρόσβαση σε αυτό.
- ΠΡ 2. Διαπιστευμένοι χρήστες του συστήματος είναι όσοι έχουν το σωστό κωδικό εισόδου.
- ΠΡ 3. Η γλώσσα προγραμματισμού είναι η C / C++.
- ΠΡ 4. Το ολοκληρωμένο περιβάλλον ανάπτυξης είναι το Arduino Software (IDE) 1.6.5.
- ΠΡ 5. Η τελική κατασκευή θα είναι μια πόρτα μικρής κλίμακας, εξομοιώνοντας τη συμπεριφορά μιας πραγματικής πόρτας.
- ΠΡ 6. Ύπαρξη πληκτρολογίου για την εισαγωγή κωδικού, ώστε να ελέγχεται ο χρήστης.
- ΠΡ 7. Αποστολή κωδικού στο σύστημα με το πάτημα ενός πλήκτρου, π.χ. \*.
- ΠΡ 8. Ύπαρξη οθόνης, ώστε να ενημερώνεται ο χρήστης για το αποτέλεσμα των ενεργειών του κατά τη διάρκεια της αλληλεπίδρασής του με το σύστημα.
- ΠΡ 9. Ύπαρξη διόδου φωτοεκπομπής (LED) χρώματος κόκκινου, ώστε να ενημερώνεται ο χρήστης για το ανεπιτυχές ή εσφαλμένο αποτέλεσμα των ενεργειών του.
- ΠΡ 10. Ύπαρξη διόδου φωτοεκπομπής (LED) χρώματος πράσινου, ώστε να ενημερώνεται ο χρήστης για το επιτυχημένο ή ορθό αποτέλεσμα των ενεργειών του.
- ΠΡ 11. Κατά την πληκτρολόγηση του κωδικού, οι εισαγόμενοι χαρακτήρες εμφανίζονται ομοιόμορφα, με κάποιον ειδικό χαρακτήρα, π.χ. \*, έτσι ώστε να προστατεύεται η πραγματική τους τιμή.
- ΠΡ 12. Μετά την τρίτη συνεχόμενη εισαγωγή λανθασμένου κωδικού, ο μηχανισμός εισαγωγής κωδικού απενεργοποιείται για τρία λεπτά, και κατάλληλο μήνυμα εμφανίζεται στην οθόνη.
- ΠΡ 13. Ύπαρξη μηχανισμού ανίχνευσης κίνησης.

- ΠΡ 14. Η φωτεινότητα της οθόνης θα πρέπει να μπορεί να ρυθμίζεται από το χρήστη μέσω ειδικού μηχανισμού.
- ΠΡ 15. Η κλειδαριά θα πρέπει να ελευθερώνεται άμεσα από το εσωτερικό μέρος, με το πάτημα κατάλληλου κουμπιού.
- ΠΡ 16. Μετά την ελευθέρωση της κλειδαριάς, ο μηχανισμός εισαγωγής κωδικού επαναφέρεται στην αρχική κατάσταση. Ουσιαστικά, μηδενίζονται οι αποτυχημένες ή εσφαλμένες προσπάθειες εισαγωγής κωδικού.
- ΠΡ 17. Ύπαρξη κουμπιού ως κουδούνι.
- ΠΡ 18. Με το πάτημα το κουμπιού κουδουνιού, αναπαράγεται κατάλληλος ήχος ειδοποίησης από το σύστημα.
- ΠΡ 19. Ύπαρξη μηχανισμού ανίχνευσης κατάστασης πόρτας (ανοικτή ή κλειστή) με χρήση μαγνητικής διάταξης.
- ΠΡ 20. Ενεργοποίηση οπτικής και ηχητικής ένδειξης για όσο διάστημα η πόρτα παραμένει ανοικτή.
- ΠΡ 21. Η ελευθέρωση της πόρτας γίνεται μέσω ηλεκτρικού κυπρί.
- ΠΡ 22. Ύπαρξη μηχανισμού διόρθωσης κωδικού, όπου το σύστημα θα διαγράφει τον κωδικό που είχε πληκτρολογηθεί ως τότε.
- ΠΡ 23. Ενεργοποίηση του μηχανισμού διόρθωσης κωδικού με το πάτημα ενός πλήκτρου, π.χ. #.
- ΠΡ 24. Ύπαρξη μηχανισμού αλλαγής κωδικού, όπου το σύστημα πρώτα θα επαληθεύει τον παλιό κωδικό και ύστερα θα ζητά την εισαγωγή νέου.
- ΠΡ 25. Ενεργοποίηση του μηχανισμού αλλαγής κωδικού με το πάτημα ενός πλήκτρου, π.χ. A.
- ΠΡ 26. Ύπαρξη προεπιλεγμένης κατάστασης:
- αρχικός κωδικός πρόσβασης: 1234
  - ενεργή κόκκινη φωτεινή ένδειξη (εφόσον η πόρτα είναι κλειστή)
  - μη αναπαραγωγή ήχου (εφόσον η πόρτα είναι κλειστή)
- ΠΡ 27. Γρήγορη απόκριση συστήματος (χρόνος απόκρισης μικρότερος του ενός δευτερολέπτου).
- ΠΡ 28. Σε περίπτωση απόκρισης με χρόνο μεγαλύτερο του ενός δευτερολέπτου, κατάλληλο μήνυμα θα πρέπει να εμφανίζεται στο χρήστη.
- ΠΡ 29. Το σύστημα θα πρέπει να διέπουν οι αξίες της ευχρηστίας, όπως αυτές ορίζονται από τον Nielsen (1994).

#### 4.1.5.1 Κοστολόγιο

Το κόστος των επιμέρους υλικών του έργου μας συνοψίζεται στον ακόλουθο πίνακα:

Αντικείμενο	Ποσότητα	Κόστος μονάδας	Συνολικό κόστος
Arduino Mega 2560	1	30,00 €	30,00 €
Κουμπί (button)	2	0,35 €	0,70 €
Υποδοχή led	2	0,30 €	0,60 €
Δίοδος led	2	0,30 €	0,60 €
Αντίσταση (resistor)	8	0,01 €	0,08 €
Πυκνωτής	1	0,02 €	0,02 €
Δίοδος	2	0,02 €	0,04 €
Ρελέ	1	0,75 €	0,75 €
Οθόνη LCD δυο γραμμών	1	6,50 €	6,50 €
Μαγνητική διάταξη πόρτας	1	0,80 €	0,80 €
Πληκτρολόγιο 4x4	1	1,65 €	1,65 €
Καλώδιο / σύρμα	1	2,00 €	2,00 €
Κυπρί	1	12,00 €	12,00 €
Κλειδαριά	1	7,00 €	7,00 €
Διακόπτης δυο καταστάσεων	1	0,50 €	0,50 €
Κλέμα	6	0,20 €	0,20 €
Είδη ξυλείας	1	10,00 €	10,00 €
Φύλλο plexiglass	1	1,50 €	1,50 €
Ποτενσιόμετρο	1	0,80 €	0,80 €
Ηχείο μικρό	1	1,25 €	1,25 €
Ηχείο μεγάλο	1	2,50 €	2,50 €
Καλώδιο USB	1	1,50 €	1,50 €
Καλώδιο τροφοδοσίας	1	6,50 €	6,50 €
<b>ΣΥΝΟΛΟ</b>			<b>87,50 €</b>

Πίνακας 2: Κοστολόγιο

## 4.2 Σχεδίαση

Η σχεδίαση αποτελεί το δεύτερο στάδιο της δημιουργίας του συστήματος και συμπεριλαμβάνει τα διαγράμματα ροής, τη σχεδίαση του ηλεκτρικού κυκλώματος και τις βασικές μεθόδους του κώδικα.

### 4.2.1 Διαγράμματα ροής

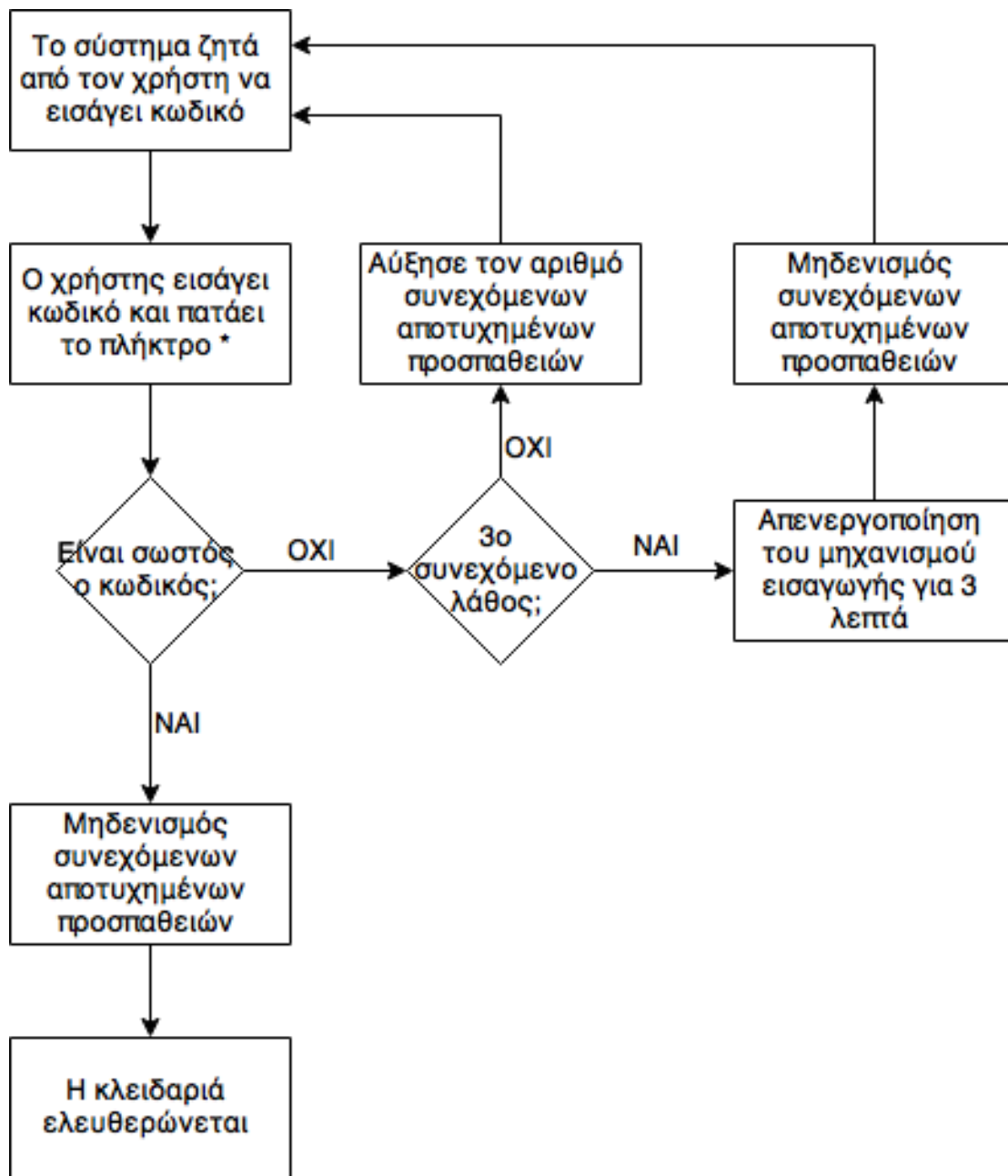
Τα διαγράμματα ροής χρησιμεύουν στη διαγραμματική απεικόνιση ενός αλγορίθμου ή μιας διαδικασίας, απεικονίζοντας τα βήματα ως κουτιά διαφόρων ειδών, τα οποία συνδέονται με κατευθυνόμενα βέλη που υποδεικνύουν τη σειρά των βημάτων. Αυτή η διαγραμματική αναπαράσταση αποτελεί ένα μοντέλο λύσης σε ένα δεδομένο πρόβλημα. Τα διαγράμματα ροής χρησιμοποιούνται για την ανάλυση, τη σχεδίαση, την τεκμηρίωση ή τη διαχείριση μιας διαδικασίας ή ενός προγράμματος σε διάφορους τομείς.

Βοηθούν στην οπτικοποίηση του προβλήματος και συμβάλλουν στην κατανόηση της διαδικασίας. Επιπλέον επιτρέπουν την ανίχνευση ελαττωματικών σημείων, σημείων συμφόρησης, και άλλων λιγότερο εμφανή χαρακτηριστικών. Υπάρχουν διάφορα είδη διαγραμμάτων ροής, και κάθε είδος περιλαμβάνει ένα δικό του σύνολο από κουτιά και συμβάσεις αναπαράστασης. Τα δύο πιο κοινά είδη κουτιών είναι:

- Ένα βήμα επεξεργασίας, που συνήθως καλείται δραστηριότητα και αναπαρίσταται με ένα τετράγωνο κουτί
- Ένα βήμα απόφασης, που συνήθως αναπαρίσταται με ένα κουτί ρόμβου.

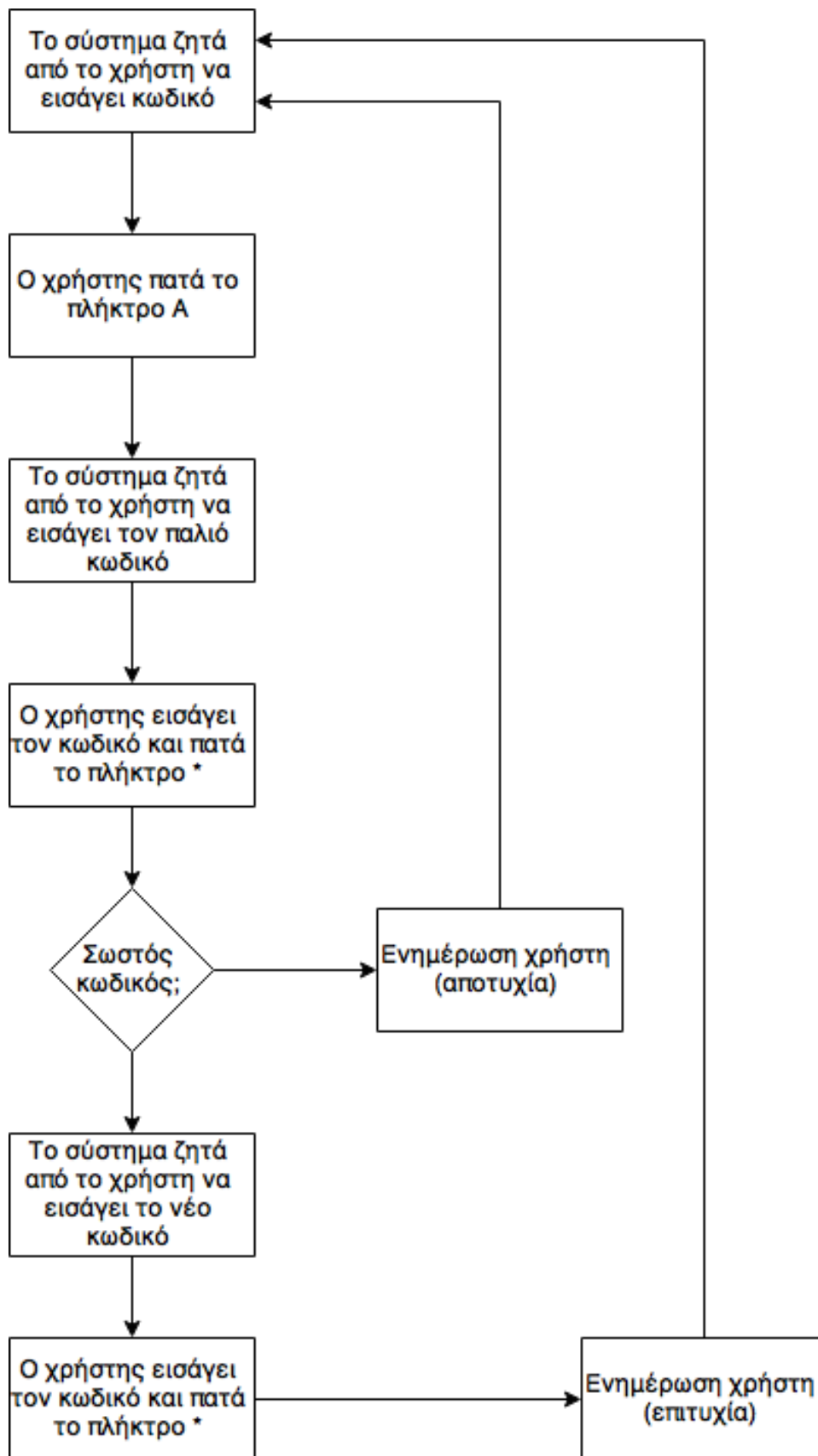
Στην παρούσα πτυχιακή εργασία θα χρησιμοποιηθούν διαγράμματα ροής για να σχεδιαστούν οι πιο σημαντικές και περίπλοκες λειτουργίες του συστήματος: η εισαγωγή κωδικού και η αλλαγή κωδικού.

#### 4.2.1.1 Εισαγωγή κωδικού



Εικόνα 13: Διάγραμμα ροής για την εισαγωγή κωδικού

#### 4.2.1.2 Αλλαγή κωδικού



Εικόνα 14: Διάγραμμα ροής για την αλλαγή κωδικού

#### 4.2.2 Σχεδίαση συνδέσεων

Για τη σχεδίαση ηλεκτρικών κυκλωμάτων είτε αναλογικών, ή ψηφιακών, είναι απαραίτητο να υπολογιστούν οι τάσεις και τα ρεύματα σε όλα τα σημεία του κυκλώματος. Η διαδικασία μπορεί να πραγματοποιηθεί με το χέρι αλλά προκειμένου να αποφευχθούν λάθη, καθώς η πολυπλοκότητα των υπολογισμών αυξάνεται με την αύξηση της πολυπλοκότητας του συστήματος, χρησιμοποιείται ειδικό λογισμικό προσομοίωσης κυκλωμάτων το οποίο ελέγχει την ακεραιότητα της σχεδίασης του ηλεκτρικού κυκλώματος και προβλέπει τη συμπεριφορά του συστήματος.

Προκειμένου να γίνει ανάλυση του συστήματος, ώστε να αποφευχθούν τυχόν λάθη και καθυστερήσεις κατά την κατασκευή του, χρησιμοποιήθηκε λογισμικό προσομοίωσης κυκλωμάτων και προέκυψε το σχήμα της επόμενης σελίδας.





## 4.3 Υλοποίηση

Στο στάδιο της υλοποίησης εμπίπτουν όλες εκείνες οι διαδικασίες που αφορούν την ανάπτυξη του συστήματος και για την παρούσα εργασία η υλοποίηση εστιάζει στον κώδικα σε γλώσσα C++. Το μέρος της υλικής υλοποίησης θα αναλυθεί παρακάτω, στην ενότητα της κατασκευής.

### 4.3.1 Σταθερές

Σε πρώτη φάση ορίζονται οι σταθερές οι οποίες έχουν καθολική ισχύ σε όλο το πρόγραμμά μας. Οι σταθερές είναι ποσότητες που έχουν γνωστή αρχική τιμή και δεν αλλάζουν κατά την εκτέλεση του προγράμματος, και ο ορισμός τους συμβάλλει στη βελτιστοποίηση και την άμεση εποπτεία του κώδικα.

Αρχικά ορίζεται το πληκτρολόγιο 4x4, καθώς και οι ακίδες στις οποίες θα συνδεθεί το πληκτρολόγιο. Τέλος, γίνεται η αντιστοίχιση του πληκτρολογίου.

```
const byte ROWS = 4;
const byte COLS = 4;
char keys[COLS][ROWS] = {
    {'1', '2', '3', 'A'},
    {'4', '5', '6', 'B'},
    {'7', '8', '9', 'C'},
    {'*', '0', '#', 'D'}
};

byte rowPins[ROWS] = { 49, 47, 45, 43 };
byte colPins[COLS] = { 41, 39, 37, 35 };
Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );
```

Κώδικας 2: Ορισμός πληκτρολογίου

Τα μηνύματα που θα εμφανίζονται στην οθόνη θα είναι στα ελληνικά, ώστε να εξυπηρετούν την καλύτερη κατανόηση από τους χρήστες, και για αυτό το λόγο θα πρέπει να ορισθούν όσοι ελληνικοί χαρακτήρες (κεφαλαία) δεν βρίσκονται σε αντιστοιχία με τους λατινικούς χαρακτήρες. Οι χαρακτήρες αυτοί είναι μόλις οκτώ και πρόκειται για τους: Γ, Δ, Λ, Ξ, Π, Φ, Ψ και Ω. Ο ορισμός των ελληνικών χαρακτήρων είναι ο εξής:

```
byte FI[8] = { B01110, B10101, B10101, B10101, B01110, B00100, B00100, B00000 };
byte PSI[8] = { B10101, B10101, B10101, B01110, B00100, B00100, B00100, B00000 };
byte OMEGA[8] = { B01110, B10001, B10001, B10001, B01110, B00000, B11111, B00000 };
byte GAMMA[8] = { B11111, B10000, B10000, B10000, B10000, B10000, B10000, B00000 };
byte DELTA[8] = { B00100, B01010, B10001, B10001, B10001, B10001, B11111, B00000 };
byte LAMDA[8] = { B00100, B01010, B10001, B10001, B10001, B10001, B10001, B00000 };
byte KSI[8] = { B11111, B00000, B00000, B01110, B00000, B00000, B11111, B00000 };
byte PEE[8] = { B11111, B10001, B10001, B10001, B10001, B10001, B10001, B00000 };
```

Κώδικας 3: Ορισμός ελληνικών χαρακτήρων

Κάποιες άλλες σταθερές που θα χρειαστούμε αφορούν άλλα χαρακτηριστικά του συστήματος, όπως η πράσινη και κόκκινη οπτική ένδειξη led, τα ηχεία, η μαγνητική διάταξη της πόρτας, το κουδούνι, οι διάφοροι τόνοι, κλπ.

```
const int speakerPin = 31;
const int note = 6000;
const int note1 = 120;
const int note2 = 999;
const int note3 = 4555;
const int doorSwitchPin = 26;
const int buttonPin = 7;
int buttonState = 0;
const int hallPin = 2;
int photocellPin = 0;
int photocellReading;
int lcdBackLight = 9; // το φως της οθόνης
int hallState = 0;
int redLed = 12; // το κόκκινο φωτάκι
int greenLed = 10; // το πράσινο φωτάκι
const int bellPin = 27; // Κουδούνι
const int stateEnterPassword = 0;
const int stateConfirmOldPassword = 1;
const int stateChangePassword = 2;
int stateKeyboard = stateEnterPassword;
const int maxFailedAttempts = 3;
```

Κώδικας 4: Άλλες σταθερές

### 4.3.2 Μέθοδος setup

Η μέθοδος `setup()` εκτελείται μόνο μια φορά, κατά την εκκίνηση του προγράμματος. Αποτελεί ουσιαστικά τη μέθοδο αρχικοποίησης διάφορων καταστάσεων και μεταβλητών για το πρόγραμμά μας, όπως η αντιστοίχιση των ελληνικών χαρακτήρων, ο ορισμός των ακίδων ως είσοδοι ή έξοδοι, η αρχικοποίηση των φωτεινών ενδείξεων, η εμφάνιση του αρχικού μηνύματος στην οθόνη, κλπ.

```
void setup()
{
    // Ορισμός ελληνικών γραμμάτων
    lcd.createChar(0, FI);
    lcd.createChar(1, PSI);
    lcd.createChar(2, OMEGA);
    lcd.createChar(3, GAMMA);
    lcd.createChar(4, DELTA);
    lcd.createChar(5, LAMDA);
    lcd.createChar(6, KSI);
    lcd.createChar(7, PEE);

    // Pins που είναι είσοδοι
    pinMode(hallPin, INPUT); // Μαγνήτης πόρτας
    pinMode(doorSwitchPin, INPUT);
    pinMode(bellPin, INPUT); // Κουδούνι

    // Pins που είναι έξοδοι
    pinMode(lcdBackLight, OUTPUT); // Οθόνη
    pinMode(redLed, OUTPUT); // Κόκκινο φως
    pinMode(greenLed, OUTPUT); // Πράσινο φως

    lcd.begin(16, 2);

    printEnterPasswordPrompt(); //Εμφάνιση μηνύματος "ΔΩΣΕ ΚΩΔΙΚΟ"

    digitalWrite(redLed, HIGH);
```

```

digitalWrite(greenLed, LOW);
keypad.addEventListener(keypadEvent); //Προσθήκη listener για το
πληκτρολόγιο

buttonState = digitalRead(buttonPin);
digitalWrite(lcdBackLight, HIGH);
timer.SetTimer(0,0,20);
stateKeyboard = stateEnterPassword;
}

```

Κώδικας 5: Μέθοδος setup

### 4.3.3 Μέθοδος loop()

Η μέθοδος `loop()` εκτελείται επαναληπτικά καθ' όλη τη διάρκεια της λειτουργίας του συστήματος. Συνεπώς, πρόκειται για την ίσως πιο σημαντική και κρίσιμη μέθοδο, καθώς όλοι οι έλεγχοι πραγματοποιούνται σε αυτήν. Συγκεκριμένα ελέγχεται αν υπάρχει ανίχνευση κίνησης, ώστε να ενεργοποιηθεί η οθόνη, ελέγχεται η κατάσταση πληκτρολογίου, δηλαδή ο χρήστης εισάγει έναν κωδικό ώστε να ελευθερώσει την κλειδαριά ή χρησιμοποιεί το πληκτρολόγιο ώστε να αλλάξει τον κωδικό. Ελέγχεται, επίσης, η ενεργοποίηση ή απενεργοποίηση του ήχου, μέσω του διακόπτη δυο θέσεων στο εσωτερικό μέρος, καθώς και η κατάσταση της μαγνητικής διάταξης, ώστε να γνωρίζουμε αν η πόρτα είναι ανοικτή ή κλειστή (οι μαγνήτες ακουμπούν ο ένας το άλλον).

```

Void loop()
{
    kinisi();

    if ( stateKeyboard == stateEnterPassword ) {
printEnterPasswordPrompt(); }

    keypad.getKey();

    if (blnBlockState){
        timer.Timer();
        lcd.setCursor(0,1);
        lcd.print(timer.ShowMinutes());
        lcd.print(":");

        if (timer.ShowSeconds() < 10) { lcd.print(0);
lcd.print(timer.ShowSeconds()); }
        else lcd.print(timer.ShowSeconds());

        //Αν μηδενίσει ο μετρητής γύρνα στην αρχική συνθήκη,
        if ( (timer.ShowSeconds() == 0) && (timer.ShowMinutes() == 0) ) {
            lcd.clear();
            blnBlockState = false;
        }
        } // /end-of-else

    if ( digitalRead(doorSwitchPin) == LOW ) soundIsActive = false;
    else soundIsActive = true;

    buttonState = digitalRead(buttonPin); // Ανάγνωση της κατάστασης του
εσωτερικού κουμπιού πόρτας
    if ( buttonState == HIGH ) { // Αν πατηθεί το εσωτερικό κουμπί,
        inDoorOpened(); // η πόρτα ανοίγει
    } // end-of-if

    hallState = digitalRead(hallPin); // Ανάγνωση της κατάστασης του
μαγνήτη

```

```

if (hallState == HIGH){ // Αν οι μαγνήτες δεν ενώνονται,
  inDoorOpened(); // η πόρτα είναι ανοιχτή
} // end-of-if

int bellState = digitalRead(bellPin); // Ανάγνωση της κατάστασης του
μαγνήτη
if (bellState == HIGH){ // Όταν το κουδούνι πατηθεί,
  playBellSound(); // ο ήχος του κουδουνιού
αναπαράγεται
} // end-of-if
}

```

Κώδικας 6: Μέθοδος loop

#### 4.3.4 Μέθοδος keypadEvent

Μέσω της μεθόδου `keypadEvent` (`KeypadEvent keyPress`), το πρόγραμμα είναι σε θέση να αναγνωρίζει το πλήκτρο που πάτησε ο χρήστης και να μεταφέρει τη ροή σε άλλο σημείο του κώδικα.

```

void keypadEvent(KeypadEvent keyPress)
{
  if(keypad.getState() == PRESSED)
  {
    currentPasswordLength++;

    switch (keyPress)
    {
      case '*':
        if ( (stateKeyboard == stateEnterPassword) || (stateKeyboard ==
stateConfirmOldPassword) ) { stateKeyboard = checkPassword(stateKeyboard);
}
        else if ( stateKeyboard == stateChangePassword ) { stateKeyboard
= setNewPassword(); }
        password.reset();
        break;
      case '#':
        clearEnteredPassword();
        break;
      case 'A':
        password.reset();
        if (stateKeyboard == stateEnterPassword) confirmOldPassword();
        break;
      case 'B':
        break;
      case 'C':
        break;
      case 'D':
        break;

      default:
        password.append(keyPress); // Δημιουργία

        if (stateKeyboard == stateChangePassword) strNewPassword +=
keyPress;

        if(ilkdongu)
        { lcd.setCursor(0,1);
          lcd.print(" ");
        }
        ilkdongu = 0;
        lcd.setCursor(k,1);
        lcd.print('*'); // Εμφάνιση *
        tone(speakerPin, note, 100); // Αναπαραγωγή ήχου
        delay(200); // Καθυστέρηση 200 ms
        k++;
    }
  }
}

```

```
}//end-of-switch
}// end of if
```

Κώδικας 7: Μέθοδος keypadEvent

#### 4.3.5 Μέθοδος clearEnteredPassword

Η μέθοδος `clearEnteredPassword()` ενεργοποιείται όταν ο χρήστης πατήσει το πλήκτρο #, ώστε να διορθώσει τον κωδικό που είχε πληκτρολογήσει ως τότε. Συγκεκριμένα, σβήνεται ο κωδικός, αναπαράγεται ο κατάλληλος ήχος, εμφανίζεται μήνυμα στην οθόνη για ενημέρωση του χρήστη, έπεται μια μικρή καθυστέρηση, ώστε να προλάβει ο χρήστης να διαβάσει το μήνυμα, και ύστερα ακολουθεί ο καθαρισμός της οθόνης και η εμφάνιση του μηνύματος της προηγούμενης κατάστασης.

```
void clearEnteredPassword() {
    password.reset(); // Σβήσιμο του κωδικού
    tone(speakerPin, note2, 500); // Αναπαραγωγή ήχου
    printClearPasswordPrompt(); // Εμφάνιση μηνύματος
    delay(2000); // Καθυστέρηση 2 δευτερολέπτων
    lcd.clear(); // Καθαρισμός οθόνης
}
```

Κώδικας 8: Μέθοδος clearEnteredPassword

#### 4.3.6 Μέθοδος εμφάνισης μηνύματος

Οι μέθοδοι αυτές χρησιμοποιούνται για να οπτικοποιήσουν ένα μήνυμα και ακολουθούν το ίδιο μοτίβο. Πιο συγκεκριμένα, αρχικά ορίζεται η θέση του μηνύματος, π.χ. 1<sup>η</sup> σειρά, 4<sup>η</sup> στήλη. Ύστερα, αναγράφεται το μήνυμα. Αν απαιτείται αλλαγή σειράς, ορίζεται εκ νέου η θέση του υπο-μέρους του μηνύματος. Για την καλύτερη κατανόηση της λειτουργίας των μεθόδων αυτών, παρατίθεται παρακάτω ο κώδικας για την εμφάνιση του μηνύματος «ΔΩΣΕ ΚΩΔΙΚΟ».

```
void printEnterPasswordPrompt()
{
    lcd.setCursor(0,0); //Τοποθέτηση μηνύματος στη θέση [0,0] - 1η γραμμή
    lcd.write("* "); //Εμφάνιση "*"
    lcd.write(byte(4));lcd.write(byte(2));lcd.write(246); lcd.write("E");
    //Εμφάνιση "ΔΩΣΕ"
    lcd.write(" K"
);lcd.write(byte(2));lcd.write(byte(4));lcd.write("I");lcd.write("K");lcd.write(
"O:"); //Εμφάνιση " ΚΩΔΙΚΟ:"
}
```

Κώδικας 9: Μέθοδος printEnterPasswordPrompt

#### 4.3.7 Μέθοδος αναπαραγωγής ήχου

Κάτι αντίστοιχο συμβαίνει και με τις μεθόδους αναπαραγωγής ήχου. Πιο συγκεκριμένα, ένας ορισμένος τόνος (σταθερή ποσότητα) αναπαράγεται για ένα συγκεκριμένο χρονικό διάστημα, εφόσον η αναπαραγωγή ήχων είναι ενεργοποιημένη, μέσω του διακόπτη δυο θέσεων στο εσωτερικό μέρος. Για την καλύτερη κατανόηση της λειτουργία των μεθόδων αυτών, παρατίθεται ο κώδικας για την αναπαραγωγή του ήχου κουδουνιού.

```
void playBellSound()
{
    if (soundIsActive){
        for (int i=0; i<10000; i++){
```

```

    tone(speakerPin, i, 200);
    delay(100);
  }
}
}

```

Κώδικας 10: Μέθοδος playBellSound

#### 4.3.8 Μέθοδος checkPassword

Μια άλλη σημαντική μέθοδος για τη λειτουργία του συστήματός μας είναι η `checkPassword(int stateKeyboard)` η οποία ουσιαστικά ελέγχει την ορθότητα του κωδικού πρόσβασης. Πιο συγκεκριμένα, αν ο κωδικός πρόσβασης είναι σωστός, καλείται η μέθοδος `inDoorOpened()`, ενώ ο αποθηκευμένος, στη μνήμη, κωδικός διαγράφεται. Αν ο κωδικός είναι λανθασμένος, το πρόγραμμα ελέγχει τον αριθμό των συνεχόμενων λανθασμένων προσπαθειών. Αν αυτός ο αριθμός είναι μικρότερος από το τρία, τότε ο χρήστης ειδοποιείται για τον εσφαλμένο κωδικό και το πρόγραμμα επιστρέφει στην αρχική κατάσταση εισαγωγής κωδικού πρόσβασης. Σε περίπτωση όμως που οι συνεχόμενες λανθασμένες προσπάθειες είναι τρεις, ο μηχανισμός εισαγωγής κωδικού απενεργοποιείται για τρία λεπτά, με τη χρήση αντίστροφου μετρητή, ενημερώνοντας παράλληλα το χρήστης.

```

int checkPassword(int stateKeyboard){

    k=6;
    maxPasswordLength = 4;
    currentPasswordLength = 0;

    if ( ( stateKeyboard == stateEnterPassword ) && ( password.evaluate() ) )
        // Αν ο κωδικός είναι σωστός,
    {
        inDoorOpened();
        // άνοιξε την πόρτα και
        password.reset();
        // σβήσε τον κωδικό
    } // end-of-if
    else if ( ( stateKeyboard == stateConfirmOldPassword ) && (
password.evaluate() ) ) // Αν ο χρήστης βρίσκεται στην κατάσταση
επιβεβαίωσης παλιού κωδικού
    {
        // ΚΑΙ ο κωδικός είναι σωστός,
        printEnterNewPasswordMessage();
        // Εμφάνιση μηνύματος νέου κωδικού
        password.reset();
        // Σβήσιμο κωδικού
        tone(speakerPin, 500, 2800);
        // Αναπαραγωγή ήχου
        stateKeyboard = stateChangePassword;
        // Αλλαγή κατάστασης
    }
    else if ( (stateKeyboard == stateConfirmOldPassword) && (
!password.evaluate() ) ) // Αν ο χρήστης βρίσκεται στην κατάσταση
επιβεβαίωσης παλιού κωδικού
    {
        // ΚΑΙ ο κωδικός ΔΕΝ είναι σωστός,
        stateKeyboard = stateEnterPassword;
        // η κατάσταση αλλάζει στην βασική
        password.reset();
        // ο κωδικός μηδενίζεται
        printWrongPasswordMessage();
        // εμφάνιση μηνύματος λάθους
        playWrongPasswordSound();
        // αναπαραγωγή ήχου λάθους
    }
}

```

```

    lcd.clear();
    // καθαρισμός οθόνης
}
else if( ( stateKeyboard == stateEnterPassword ) && ( intFailedAttempts
< maxFailedAttempts ) )
{
    printWrongPasswordMessage(); // Εμφάνιση μηνύματος λάθους

    digitalWrite(10,LOW);
    delay(200);
    digitalWrite(10, HIGH);
    digitalWrite(10, HIGH);
    delay(200);
    digitalWrite(10,LOW);
    delay(200);
    digitalWrite(10, HIGH);
    delay(500);

    lcd.clear(); // Καθαρισμός οθόνης
    printEnterPasswordPrompt(); // Εμφάνιση μηνύματος νέου κωδικού
    intFailedAttempts++; // Αύξηση των λανθασμένων προσπαθειών
} // end-of-else-if
else if( ( stateKeyboard == stateEnterPassword ) && ( intFailedAttempts
== maxFailedAttempts ) )
{
    // 10 seconds
    timer.StartTimer();

    blnBlockState = true; // Αποκλεισμός χρήστη από την εισαγωγή
    // νέου κωδικού
    printTryNewPasswordMessage(); // Εμφάνιση μηνύματος ενημέρωσης προς
    // το χρήστη
    intFailedAttempts = 0; // Μηδενισμός των αποτυχημένων
    // προσπαθειών πρόβασης
}

return stateKeyboard;
}

```

Κώδικας 11: Μέθοδος checkPassword

#### 4.3.9 Μέθοδος inDoorOpened

Η μέθοδος `inDoorOpened()` καλείται κάθε φορά που η πόρτα είναι ανοικτή. Κατά την εκτέλεσή της, η οθόνη είναι ενεργή και εμφανίζει το μήνυμα «ΠΟΡΤΑ ΑΝΟΙΧΤΗ».

Ταυτόχρονα, η πράσινη φωτεινή ένδειξη είναι ενεργή, ενώ η κόκκινη ανενεργή.

Παράλληλα, αναπαράγεται συγκεκριμένος ήχος και μηδενίζονται οι συνεχόμενες αποτυχημένες προσπάθειες εισαγωγής κωδικού πρόσβασης.

```

void inDoorOpened(){
    digitalWrite(lcdBackLight, HIGH); // Η οθόνη είναι ενεργή
    printOpenDoorMessage(); // Εμφανίζεται το μήνυμα
    "ΠΟΡΤΑ ΑΝΟΙΧΤΗ"

    digitalWrite(greenLed,HIGH); // Το πράσινο led ανάβει
    digitalWrite(redLed,LOW); // Το κόκκινο led σβήνει
}

```



```

    if (soundIsActive){
        tone(speakerPin, note3, 1500); // Αναπαραγωγή ήχου note3 για
1500 ms
        delay(1500); // Καθυστέρηση 1500 ms
    }

    intFailedAttempts = 0; // Μηδενισμός των αποτυχημένων
προσπαθειών
}

```

Κώδικας 12: Μέθοδος inDoorOpened

#### 4.3.10 Μέθοδος setNewPassword()

Η μέθοδος `setNewPassword()` καλείται κάθε φορά που ο χρήστης επιθυμεί να αλλάξει τον κωδικό πρόσβασης. Ως νέος κωδικός πρόσβασης ορίζεται ο τελευταίος συνδυασμός αλφαριθμητικών στοιχείων που πληκτρολόγησε ο χρήστης και αποθηκεύεται στη μνήμη του συστήματός μας. Ταυτόχρονα, ο χρήστης ενημερώνεται με κατάλληλο μήνυμα, στη συνέχεια η οθόνη καθαρίζεται, και τέλος, το σύστημα επιστρέφει στην αρχική οθόνη.

```

int setNewPassword(){
    // Μετατροπή string σε πίνακα char
    char charNewPassword[1024];
    strncpy(charNewPassword, strNewPassword.c_str(),
sizeof(charNewPassword));
    charNewPassword[sizeof(charNewPassword) - 1] = 0;

    password.set(charNewPassword); // Ως κωδικός ορίζεται ο συνδυασμός που
πληκτρολόγησε ο χρήστης
    password.reset(); // Ο κωδικός μηδενίζεται
    strNewPassword = ""; // Η συμβολοσειρά που κρατάει το νέο
κωδικό μηδενίζεται επίσης
    lcd.clear(); // Καθαρισμός οθόνης
    return stateEnterPassword; // Επιστροφή επόμενης κατάστασης
}

```

Κώδικας 13: Μέθοδος setNewPassword





## 4.4 Έλεγχος

Το τελικό στάδιο, πριν την κατασκευή του συστήματος της ηλεκτρικής κλειδαριάς, είναι το στάδιο του ελέγχου. Κατά τη διάρκεια της δοκιμής, λοιπόν, το σύστημά μας λειτουργεί υπό ελεγχόμενες συνθήκες, ώστε να επαληθεύσουμε ότι πληροί τις τελικές προδιαγραφές και να εντοπίσουμε λάθη και αστοχίες, ώστε να τα διορθώσουμε πριν το στάδιο της κατασκευής (Myers, et al., 2011).

Παρά το γεγονός πως είναι αδύνατη η εύρεση όλων των πιθανών δυσλειτουργιών ενός συστήματος, όσο διεξοδικός κι αν είναι ο έλεγχος, το στάδιο του ελέγχου αυξάνει την αξιοπιστία του συστήματος, και θα πρέπει σε κάθε περίπτωση να πραγματοποιείται ενδελεχώς (Malaiya, et al., 1994). Τα βασικά είδη ελέγχου είναι ο έλεγχος λειτουργίας και ο έλεγχος δομής, και θα αναλυθούν στις επόμενες υπο-ενότητες.

### 4.4.1 Έλεγχος προδιαγραφών

Στον επόμενο πίνακα παρουσιάζονται οι αρχικές μας προδιαγραφές και ο έλεγχος για το αν έχουν καλυφθεί ή όχι.

Α/Α	Περιγραφή	Κάλυψη
ΠΡ 1.	Χρήστες του συστήματος είναι όλοι όσοι έχουν φυσική πρόσβαση σε αυτό.	
ΠΡ 2.	Διαπιστευμένοι χρήστες του συστήματος είναι όσοι έχουν το σωστό κωδικό εισόδου.	
ΠΡ 3.	Η γλώσσα προγραμματισμού είναι η C / C++.	
ΠΡ 4.	Το ολοκληρωμένο περιβάλλον ανάπτυξης είναι το Arduino Software (IDE) 1.6.5.	
ΠΡ 5.	Η τελική κατασκευή θα είναι μια πόρτα μικρής κλίμακας, εξομοιώνοντας τη συμπεριφορά μιας πραγματικής πόρτας.	
ΠΡ 6.	Ύπαρξη πληκτρολογίου για την εισαγωγή κωδικού, ώστε να ελέγχεται ο χρήστης.	
ΠΡ 7.	Αποστολή κωδικού στο σύστημα με το πάτημα ενός πλήκτρου, π.χ. *.	
ΠΡ 8.	Ύπαρξη οθόνης, ώστε να ενημερώνεται ο χρήστης για το αποτέλεσμα των ενεργειών του κατά τη διάρκεια της αλληλεπίδρασής του με το σύστημα.	
ΠΡ 9.	Ύπαρξη διόδου φωτοεκπομπής (LED) χρώματος κόκκινου, ώστε να ενημερώνεται ο χρήστης για το ανεπιτυχές ή εσφαλμένο αποτέλεσμα των ενεργειών του.	

<b>ΠΡ 10.</b>	Ύπαρξη διόδου φωτοεκπομπής (LED) χρώματος πράσινου, ώστε να ενημερώνεται ο χρήστης για το επιτυχημένο ή ορθό αποτέλεσμα των ενεργειών του.	
<b>ΠΡ 11.</b>	Κατά την πληκτρολόγηση του κωδικού, οι εισαγόμενοι χαρακτήρες εμφανίζονται ομοιόμορφα, με κάποιον ειδικό χαρακτήρα, π.χ. *, έτσι ώστε να προστατεύεται η πραγματική τους τιμή.	
<b>ΠΡ 12.</b>	Μετά την τρίτη συνεχόμενη εισαγωγή λανθασμένου κωδικού, ο μηχανισμός εισαγωγής κωδικού απενεργοποιείται για τρία λεπτά, και κατάλληλο μήνυμα εμφανίζεται στην οθόνη.	
<b>ΠΡ 13.</b>	Ύπαρξη μηχανισμού ανίχνευσης κίνησης.	
<b>ΠΡ 14.</b>	Η φωτεινότητα της οθόνης θα πρέπει να μπορεί να ρυθμίζεται από το χρήστη μέσω ειδικού μηχανισμού.	
<b>ΠΡ 15.</b>	Η κλειδαριά θα πρέπει να ελευθερώνεται άμεσα από το εσωτερικό μέρος, με το πάτημα κατάλληλου κουμπιού.	
<b>ΠΡ 16.</b>	Μετά την ελευθέρωση της κλειδαριάς, ο μηχανισμός εισαγωγής κωδικού επαναφέρεται στην αρχική κατάσταση. Ουσιαστικά, μηδενίζονται οι αποτυχημένες ή εσφαλμένες προσπάθειες εισαγωγής κωδικού.	
<b>ΠΡ 17.</b>	Ύπαρξη κουμπιού ως κουδούνι.	
<b>ΠΡ 18.</b>	Με το πάτημα το κουμπιού κουδουνιού, αναπαράγεται κατάλληλος ήχος ειδοποίησης από το σύστημα.	
<b>ΠΡ 19.</b>	Ύπαρξη μηχανισμού ανίχνευσης κατάστασης πόρτας (ανοικτή ή κλειστή) με χρήση μαγνητικής διάταξης.	
<b>ΠΡ 20.</b>	Ενεργοποίηση οπτικής και ηχητικής ένδειξης για όσο διάστημα η πόρτα παραμένει ανοικτή.	
<b>ΠΡ 21.</b>	Η ελευθέρωση της πόρτας γίνεται μέσω ηλεκτρικού κυπρί.	
<b>ΠΡ 22.</b>	Ύπαρξη μηχανισμού διόρθωσης κωδικού, όπου το σύστημα θα διαγράφει τον κωδικό που είχε πληκτρολογηθεί ως τότε.	
<b>ΠΡ 23.</b>	Ενεργοποίηση του μηχανισμού διόρθωσης κωδικού με το πάτημα ενός πλήκτρου, π.χ. #.	
<b>ΠΡ 24.</b>	Ύπαρξη μηχανισμού αλλαγής κωδικού, όπου το σύστημα πρώτα θα επαληθεύει τον παλιό κωδικό και ύστερα θα ζητά την εισαγωγή νέου.	
<b>ΠΡ 25.</b>	Ενεργοποίηση του μηχανισμού αλλαγής κωδικού με το πάτημα ενός πλήκτρου, π.χ. Α.	

<b>ΠΡ 26.</b>	Ύπαρξη προεπιλεγμένης κατάστασης: 1) αρχικός κωδικός πρόσβασης: 1234 2) ενεργή κόκκινη φωτεινή ένδειξη (εφόσον η πόρτα είναι κλειστή) 3) μη αναπαραγωγή ήχου (εφόσον η πόρτα είναι κλειστή)	
<b>ΠΡ 27.</b>	Γρήγορη απόκριση συστήματος (χρόνος απόκρισης μικρότερος του ενός δευτερολέπτου).	
<b>ΠΡ 28.</b>	Σε περίπτωση απόκρισης με χρόνο μεγαλύτερο του ενός δευτερολέπτου, κατάλληλο μήνυμα θα πρέπει να εμφανίζεται στο χρήστη.	
<b>ΠΡ 29.</b>	ΠΡ 29. Το σύστημα θα πρέπει να διέπουν οι αξίες της ευχρηστίας, όπως αυτές ορίζονται από τον Nielsen (1994).	

Πίνακας 3: Έλεγχος προδιαγραφών



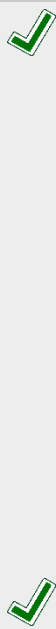
#### 4.4.2 Έλεγχος λειτουργίας




Ο έλεγχος λειτουργίας συστήματος είναι μια τεχνική που χρησιμοποιείται ώστε να γίνει δοκιμή του συστήματος τόσο ως προς τη λειτουργία του όσο και ως προς τη συμπεριφορά του. Η μέθοδος αυτή δεν απαιτεί γνώση του εσωτερικού του συστήματος ή του πηγαίου κώδικα. Προκειμένου να διεξαχθεί έλεγχος λειτουργίας του συστήματος αρκεί να είναι γνωστή η είσοδος και η έξοδος του συστήματος προς εξέταση, γεγονός που θα επιτρέψει την αξιολόγηση της ορθής λειτουργίας με βάση τις προδιαγραφές του συστήματος.

Για τη διεξαγωγή του ελέγχου λειτουργίας δημιουργήθηκαν τα πλάνα δοκιμής. Στους επόμενους πίνακες παρουσιάζονται ενδεικτικά μια σειρά από ελέγχους λειτουργίας. Σε κάθε πλάνο δοκιμής καθορίζονται οι έγκυρες και μη έγκυρες τιμές εισόδου, και για κάθε μια από αυτές καταγράφεται η αναμενόμενη έξοδος σύμφωνα με τις προδιαγραφές του συστήματος και η πραγματική έξοδος. Όταν αυτές δεν ταυτίζονται τότε υπάρχουν σφάλματα στο σύστημα, το οποίο θα πρέπει να διορθωθούν και στη συνέχεια να γίνει ένας νέος έλεγχος λειτουργίας.

##### 4.4.2.1 Αλληλεπίδραση με πληκτρολόγιο

Χαρακτηριστικά εισόδου	Έγκυρη κλάση δεδομένων	Άκυρη κλάση δεδομένων
<b>Πλήκτρο του πληκτρολογίου 4x4</b>	Οποιοσδήποτε αριθμός, τα σύμβολα * και #, και ο χαρακτήρας A	Οι χαρακτήρες B, C, D

Χαρακτηριστικό υπό έλεγχο	Έλεγχοι	Αναμενόμενο αποτέλεσμα	Πραγματικό αποτέλεσμα
<b>Πάτημα κάποιου πλήκτρου αριθμού</b>	0 1 2 3 4 5 6 7 8 9	Ένας αστερίσκος (*) εμφανίζεται αντί για την πραγματική τιμή στην οθόνη. Το σύστημα ωστόσο αποθηκεύει τη σωστή τιμή. Μετά το πάτημα του πλήκτρου, συγκεκριμένος τόνος ήχου αναπαράγεται.	
<b>Πάτημα του πλήκτρου #</b>	#	Ό,τι έχει πληκτρολογηθεί από το χρήστη σβήνεται τόσο από την οθόνη όσο και από την προσωρινή μνήμη του συστήματος. Παράλληλα, κατάλληλος ήχος αναπαράγεται.	
<b>Πάτημα του πλήκτρου *</b>	* 1. είσοδος νέου κωδικού  2α. αλλαγή κωδικού (1 <sup>η</sup> φάση)	1. Κατά την εισαγωγή νέου κωδικού, και το πάτημα του πλήκτρου *, το σύστημα ελέγχει αν ο κωδικός είναι σωστός. Αν είναι σωστός, η κλειδαριά ελευθερώνεται. Αλλιώς, ενημερώνει το χρήστη και επιστρέφει στην αρχική οθόνη.  2α. Κατά την πρώτη φάση αλλαγής κωδικού, και το πάτημα του πλήκτρου *, ο κωδικός ελέγχεται κι αν είναι σωστός, το σύστημα μεταφέρει το χρήστη στην επόμενη φάση αλλαγής κωδικού. Αλλιώς, ενημερώνει το	

		χρήστη και επιστρέφει στην αρχική οθόνη.	
	2β. αλλαγή κωδικού (2 <sup>η</sup> φάση)	2β. Κατά τη δεύτερη φάση αλλαγής κωδικού, και το πάτημα του πλήκτρου *, ο νέος κωδικός ορίζεται στο σύστημα, το οποίο ενημερώνει το χρήστη και επιστρέφει στην αρχική οθόνη.	
<b>Πάτημα του πλήκτρου A</b>	A	Το σύστημα μπαίνει σε διαδικασία αλλαγής κωδικού, και ο χρήστης ενημερώνεται κατάλληλα. Σε περίπτωση που το σύστημα βρίσκεται ήδη σε κατάσταση αλλαγής κωδικού, το πάτημα του πλήκτρου A, δεν φέρει αλλαγές.	
<b>Πάτημα οποιουδήποτε άλλου πλήκτρου</b>	B C D	Το πάτημα άλλων πλήκτρων δεν οδηγεί το σύστημά μας σε κάποια αλλαγή.	

#### 4.4.3 Έλεγχος δομής

Ο έλεγχος δομής συστήματος είναι τεχνική που χρησιμοποιείται ώστε να γίνει εξέταση των εσωτερικών δομών του συστήματος και απαιτείται γνώση τόσο των στοιχείων του συστήματος όσο και του πηγαίου κώδικα. Προκειμένου να διεξαχθεί αυτός ο έλεγχος επιλέγονται συγκεκριμένες τιμές εισόδου έτσι ώστε να εξεταστούν συγκεκριμένοι κόμβοι του συστήματος ή συγκεκριμένες διακλαδώσεις του κώδικα.

Όπως και στην περίπτωση του ελέγχου λειτουργίας συστήματος, έτσι και στην περίπτωση ελέγχου δομής συστήματος θα παρουσιαστεί ένα μικρό μέρος του ελέγχου. Είναι απαραίτητο να καθοριστούν οι συνθήκες, οι οποίες θα οδηγήσουν στην εξέταση ή στην αποφυγή ενός συγκεκριμένου κόμβου ή ενός συγκεκριμένου τμήματος του κώδικα, το μονοπάτι το οποίο εξετάζεται, καθώς και το αναμενόμενο και το πραγματικό αποτέλεσμα.

### Υπό εξέταση μέθοδος

```
void inDoorOpened() {
    digitalWrite(lcdBackLight, HIGH);
    printOpenDoorMessage();
    digitalWrite(greenLed, HIGH);
    digitalWrite(redLed, LOW);
    if (soundIsActive) {
        tone(speakerPin, note3, 1500);
        delay(1500);
    }

    intFailedAttempts = 0;
}
```

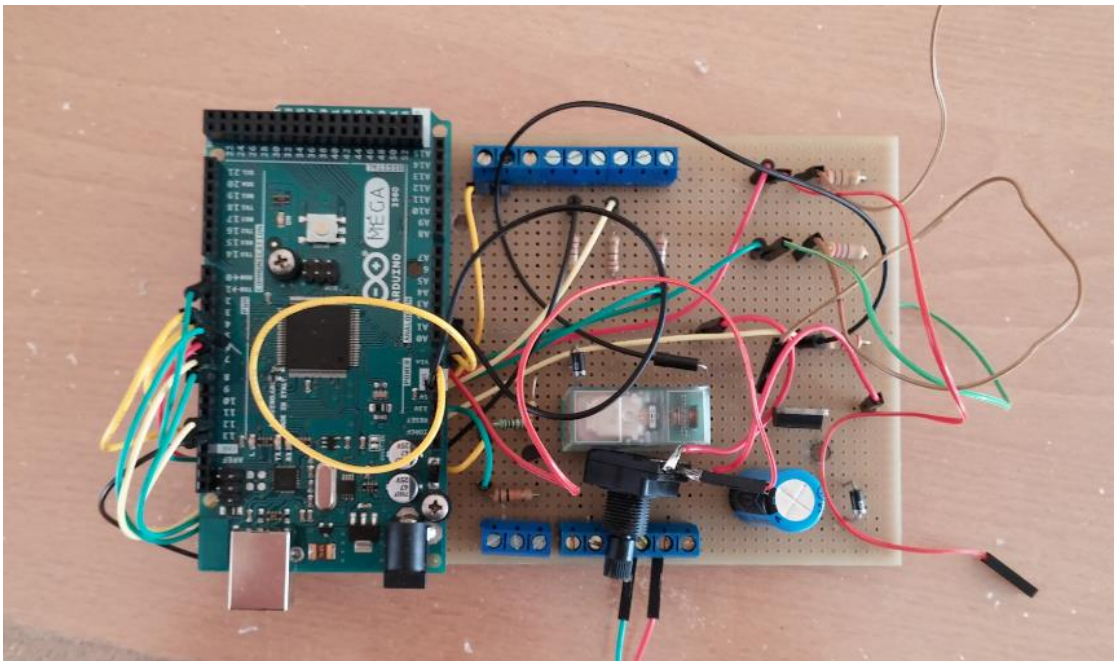
Υπό έλεγχος κώδικας	Συνθήκες για να επιτευχθεί η εκτέλεση του κώδικα	Συνθήκες για να αποφευχθεί η εκτέλεση του κώδικα
Κώδικας πριν το IF	-	-
Κώδικας μέσα στο IF	soundIsActive = true	soundIsActive = false
Κώδικας μετά το IF	-	-

Υπό έλεγχο μονοπάτι	Έλεγχος	Αναμενόμενο αποτέλεσμα	Πραγματικό αποτέλεσμα
Κώδικας πριν το IF	Ορίζεται η κατάσταση του συστήματος, όταν η πόρτα ανοιχθεί	- ενεργοποιείται η οθόνη - εμφανίζεται κατάλληλο μήνυμα - ανάβει η πράσινη φωτεινή ένδειξη - σβήνει η κόκκινη φωτεινή ένδειξη	
Κώδικας πριν το IF + μέσα στο IF	Μαζί με τον ορισμό της κατάστασης, αναπαράγεται και ήχος	Σε συνέχεια των παραπάνω, αναπαράγεται ο τόνος tone3 για 1500 ms.	
Κώδικας πριν το IF + μέσα στο IF + μετά το IF	Μαζί με τον ορισμό της κατάστασης, της αναπαραγωγής του ήχου, μηδενίζονται οι αποτυχημένες προσπάθειες εισαγωγής κωδικού.	Σε συνέχεια των παραπάνω, μηδενίζονται οι συνεχόμενες προσπάθειες εισαγωγής λανθασμένου κωδικού πρόσβασης.	
Κώδικας πριν το IF + μετά το IF	Μαζί με τον ορισμό της	- ενεργοποιείται η οθόνη	

	<p>κατάστασης, μηδενίζονται οι αποτυχημένες προσπάθειες εισαγωγής κωδικού.</p>	<ul style="list-style-type: none"> <li>- εμφανίζεται κατάλληλο μήνυμα</li> <li>- ανάβει η πράσινη φωτεινή ένδειξη</li> <li>- σβήνει η κόκκινη φωτεινή ένδειξη</li> <li>- μηδενίζονται οι συνεχόμενες προσπάθειες εισαγωγής λανθασμένου κωδικού πρόσβασης</li> </ul>	
--	--------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

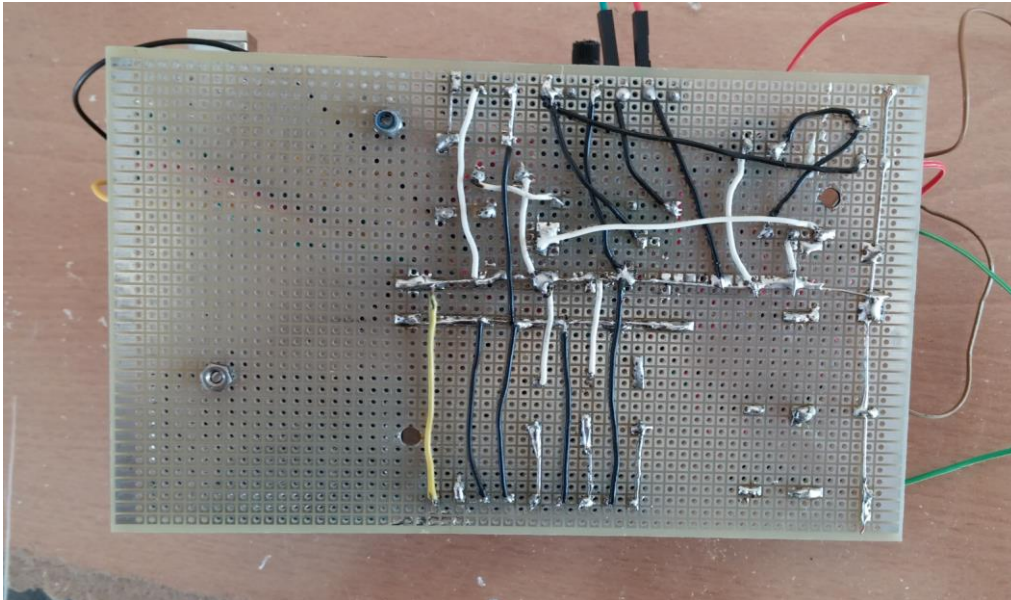
#### 4.5 Κατασκευή

Στην φωτογραφία αυτή βλέπουμε την τοποθέτηση των υλικών που χρησιμοποιήσαμε, πάνω στη πλακέτα PCB





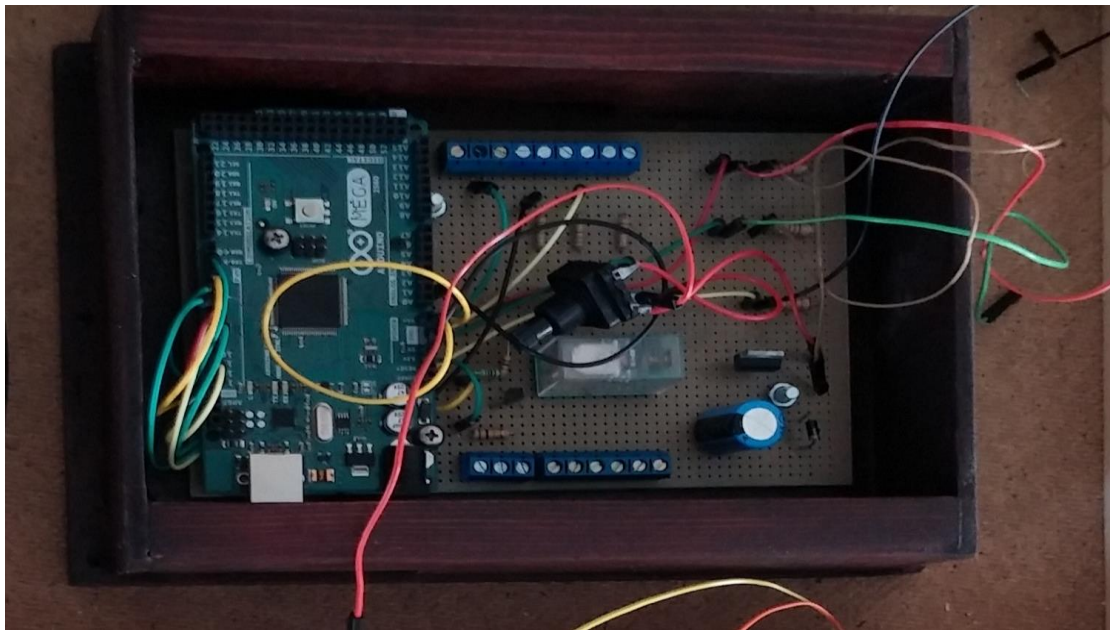
Στην παρακάτω φωτογραφία βλέπουμε την πίσω όψη της πλακέτας PCB. Φαίνονται οι ηλεκτροκολλήσεις των υλικών που χρησιμοποιήσαμε. Χαρακτηριστικό είναι οι δυο ευθείες γραμμές κόλλησης στο κέντρο και στα δεξιά της πλακέτας, μια για τη γείωση και μια για την τάση των 5 Volts, καθιστώντας το σύστημά μας πιο αποδοτικό ώστε να διευκολύνει τις συνδέσεις.



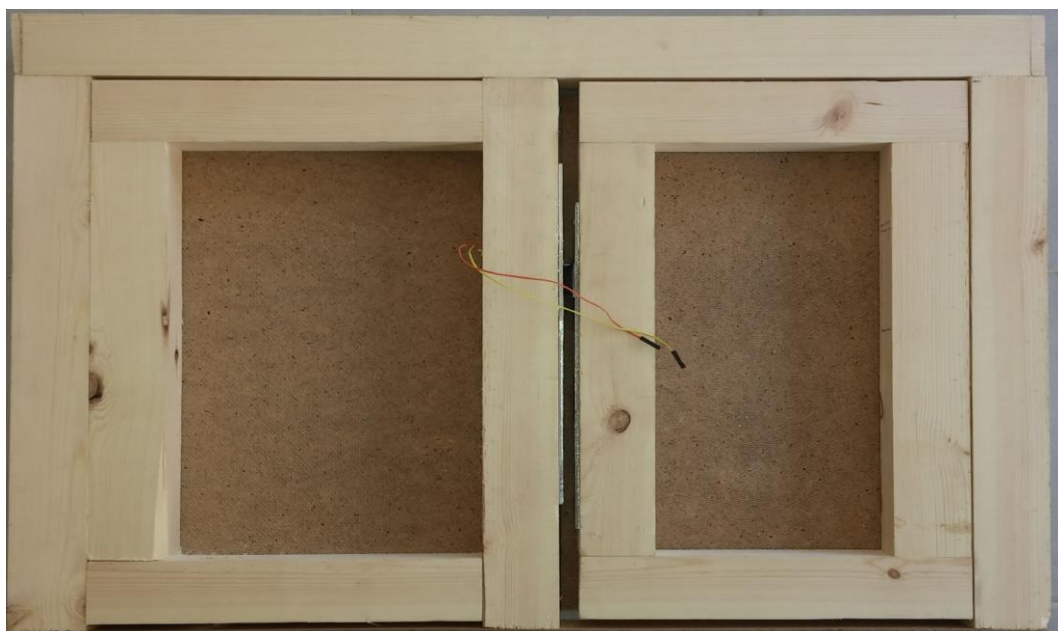
Στη συνέχεια, λαμβάνοντας υπόψη τις διαστάσεις της πλακέτας, δημιουργήσαμε μια ξύλινη κατασκευή, ώστε να την τοποθετήσουμε στο εσωτερικό της.



Η ένταξη της πλακέτας και του κυκλώματός μας στο εσωτερικό της ξύλινης κατασκευής, παρουσιάζεται στην επόμενη εικόνα.



Στην επόμενη φωτογραφία βλέπουμε τη ξύλινη κατασκευή της οποίας το δεξί μέρος θα αποτελέσει το σταθερό μέρος, προσομοιώνοντας τον τοίχο, ώστε να τοποθετήσουμε σε αυτό το κουτί που περιέχει την πλακέτα μας. Το αριστερό μέρος θα αποτελέσει το κινητό μέρος της κατασκευής, εξομοιώνοντας την πόρτα. Επίσης, διακρίνονται τα καλώδια σύνδεσης που φέρει το ηλεκτρικό κυβρί.



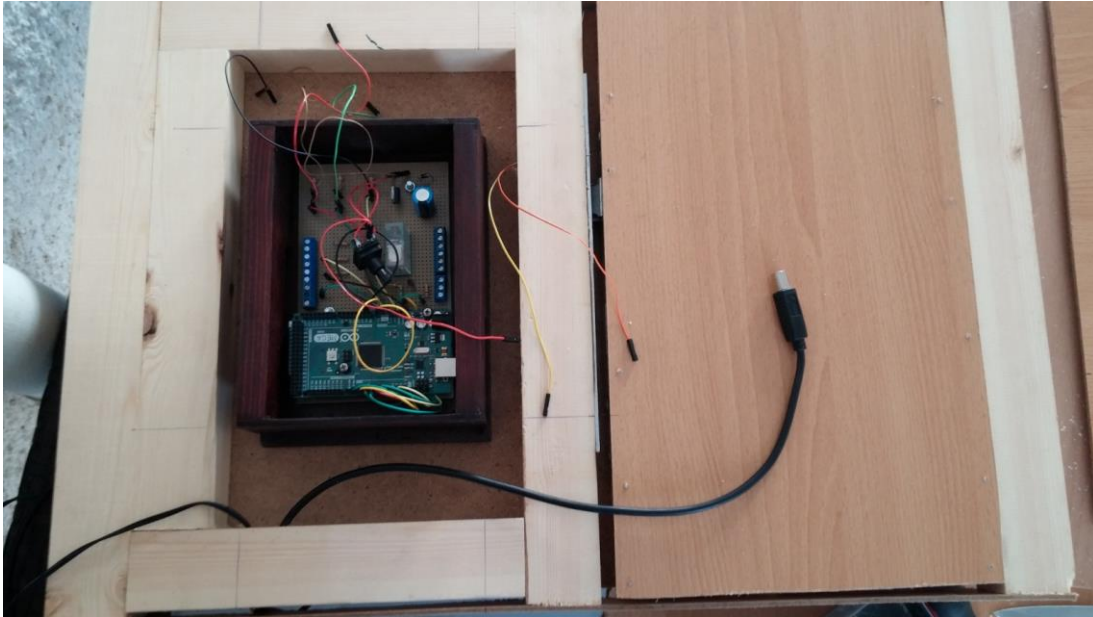
Στην επόμενη φωτογραφία βλέπουμε το ηλεκτρικό κυβρί, στο πάνω μέρος, και την κλειδαριά, στο κάτω μέρος. Το ηλεκτρικό κυβρί και η κλειδαριά βρίσκονται τοποθετημένα αντικρυστά, στο σταθερό και το κινητό μέρος αντίστοιχα.



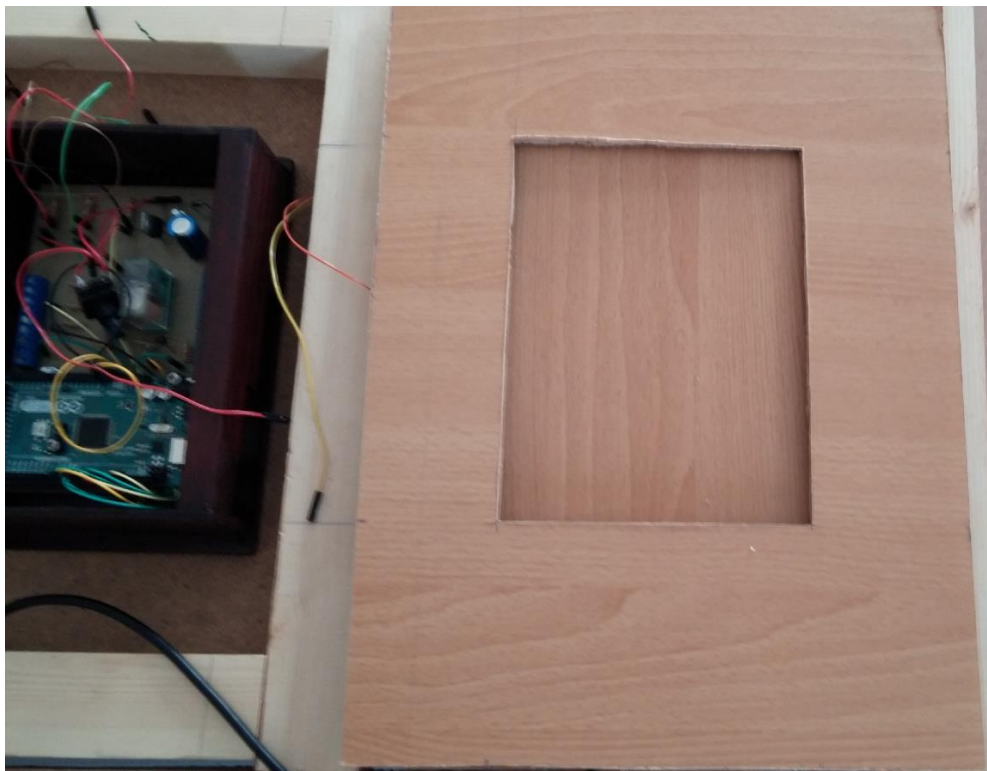
Στην επόμενη φωτογραφία βλέπουμε τα καλώδια τροφοδοσίας του Arduino, καθώς και τα καλώδια τροφοδοσίας ενεργοποίησης του ρελέ, ώστε να λειτουργήσει το ηλεκτρικό κυβρί.



Στην επόμενη φωτογραφία βλέπουμε την τοποθέτηση του κουτιού που περιέχει το Arduino και την πλακέτα μας στο εσωτερικό μέρος της ξύλινης κατασκευής.



Στην επόμενη φωτογραφία το ξύλινο φύλλο που θα επενδύσει το μπροστινό μέρος του σταθερού τμήματος της ξύλινης κατασκευής.



Στη συνέχεια αφού συνδέσαμε την οθόνη LCD και το πληκτρολόγιο με τις κατάλληλες ακίδες του Arduino, ενισχύσαμε το κουτί με φύλλο plexiglass, κατάλληλα διαμορφωμένο ώστε να επιτραπεί η έξοδος της οθόνης LCD, του πληκτρολογίου, των δυο φωτεινών ενδείξεων led, ενός ποτενσιόμετρου (ρυθμιστής φωτεινότητας χαρακτήρων οθόνης) και ενός κουμπιού (κουδούνι). Τα καλώδια που διακρίνονται περιφερειακά του κουτιού, είναι οι συνδέσεις που αναφέραμε στην πρώτη φωτογραφία.



Στην επόμενη φωτογραφία βλέπουμε τη μπροστινή όψη της κατασκευής (εξωτερικό μέρος), όπου εκτός από το κουτί διακρίνεται και ο αισθητήρας ανίχνευσης κίνησης.



Στην επόμενη φωτογραφία βλέπουμε την πίσω όψη της κατασκευής. Συγκεκριμένα, εμφανίζονται η μαγνητική διάταξη ελέγχου κατάστασης πόρτας, το εσωτερικό κουμπί απελευθέρωσης κλειδαριάς και ο διακόπτης δυο καταστάσεων, για την ενεργοποίηση/απενεργοποίηση του ήχου.



Τέλος, στην επόμενη φωτογραφία βλέπουμε την τελική μας κατασκευή σε κατάσταση αρχικής λειτουργίας.



## 5 Σύνοψη και Μελλοντική Δουλειά

Στην παρούσα πτυχιακή εργασία παρουσιάστηκε η δημιουργία ενός ψηφιακού συστήματος ελέγχου με χρήση κλειδαριάς και ηλεκτρολογίου. Όλα τα στάδια της δημιουργίας παρουσιάστηκαν εκτενώς, ξεκινώντας με την ανάλυση και ορισμό των απαιτήσεων του έργου και την εξαγωγή των τελικών προδιαγραφών του συστήματος ελέγχου.

Η δημιουργία του συστήματος ελέγχου συνεχίστηκε με την ακριβή και λεπτομερή φάση της σχεδίασης. Ακολούθησε το στάδιο της υλοποίησης, όπου συγγράφηκε και τεκμηριώθηκε ο απαραίτητος κώδικας σε γλώσσα C++, ώστε να προγραμματιστεί και ενεργοποιηθεί το σύστημα, καλύπτοντας τις δηλωθείσες προδιαγραφές.

Οι προδιαγραφές ελέγχθηκαν κατά το στάδιο ελέγχου, ώστε να εξασφαλισθεί η κάλυψη τους από το σύστημα ελέγχου. Κατά το στάδιο αυτό, ελέγχθηκε επίσης η συνολική λειτουργία και συμπεριφορά του συστήματος. Τέλος, το σύστημα πήρε φυσική υπόσταση κατά το στάδιο της κατασκευής.

Η ενασχόληση σε ένα σχετικά μικρό αλλά αρκετά πολύπλοκο έργο ήταν μια άκρως ενδιαφέρουσα εμπειρία, καθώς η ομάδα ενασχολήθηκε με όλους τα στάδια δημιουργίας ενός προϊόντος, από την αρχική ιδέα ως την τελική κατασκευή. Επίσης, πέρα από το θεωρητικό, τεχνικό και δημιουργικό κομμάτι του έργου, η ομάδα υλοποίησης εκμεταλλευομένη τη φύση του έργου, ανέπτυξε και βελτίωσε κι άλλες περιμετρικές ιδιότητες και δεξιότητες, όπως η ομαδικότητα, η συνεργασία, η παραγωγή έργου με περιορισμό χρόνου και πόρων, ο κατάλληλος επιμερισμός έργου, κλπ.

Κοιτώντας πίσω τη διαδικασία δημιουργίας του συστήματος ελέγχου, ασφαλώς υπάρχουν σημεία τα οποία θα μπορούσαν να διευθετηθούν και χειριστούν με αποδοτικότερο τρόπο, είτε εξασφαλίζοντας περισσότερο χρόνο και πόρους, είτε έχοντας μεγαλύτερη εμπειρία σε έργα αυτού του τύπου. Συνολικά, και έχοντας ως δεδομένες τους παραπάνω περιορισμούς, η διαδικασία σχεδίασης και υλοποίησης ενός συστήματος ελέγχου, βασισμένου σε ηλεκτρονική κλειδαριά με τη χρήση ηλεκτρολογίου, κρίνεται πετυχημένη, καθώς όλα τα μέρη που συμμετείχαν σε αυτήν, αποκόμισαν κέρδη, είτε αυτά αφορούν τεχνική κατάρτιση, εμπειρία, συμμετοχή σε ομαδικά έργα, κλπ.

### 5.1 Μελλοντική Δουλειά

Όσον αφορά τη μελλοντική δουλειά για την επέκταση του συστήματος ελέγχου, αρκετά πράγματα θα μπορούσαν να υλοποιηθούν. Καταρχήν, θα πρέπει το παρόν σύστημα ηλεκτρονικής κλειδαριάς να εξετασθεί και να ελεγχθεί σε μεγαλύτερο βάθος και υψηλή λεπτομέρεια, ώστε να αυξηθεί η αξιοπιστία του συστήματος.

Η επέκταση του συστήματος της ηλεκτρονικής κλειδαριάς μπορεί να σχετίζεται είτε με επιπλέον χαρακτηριστικά και λειτουργίες του παρόντος, ή με την αλλαγή των βασικών μεθόδων ελέγχου. Το Arduino είναι μια πλατφόρμα με υψηλό βαθμό επεκτασιμότητας, οπότε οι δυνατότητες διεύρυνσης είναι απεριόριστες.

Όσον αφορά τα χαρακτηριστικά και τις λειτουργίες του συστήματος, θα μπορούσε να προστεθεί ένας μηχανισμός επαναφοράς συστήματος σε αρχική (εργοστασιακή) κατάσταση. Ο μηχανισμός αυτός είναι χρήσιμος όταν κανένα από τα διαπιστευμένα μέλη δε θυμάται τον κωδικό πρόσβασης, ή όταν το σύστημα έχει αστοχήσει.

Επιπλέον, θα μπορούσαν οι διαπιστευμένοι χρήστες να αντιστοιχούν σε ξεχωριστές οντότητες, όπου η κάθε μια από αυτές θα έχει μοναδικό όνομα και κωδικό πρόσβασης. Αυτός ο μηχανισμός αυξάνει το επίπεδο ασφάλειας που παρέχει το σύστημα, αλλά ταυτόχρονα επηρεάζει την ευχρηστία και την πολυπλοκότητα του συστήματος.

Επιπρόσθετα, θα μπορούσαν οι διάφορες ρυθμίσεις του συστήματος, όπως το ανώτατο όριο συνεχόμενων αποτυχημένων προσπαθειών κι ο χρόνος αποκλεισμού από την εισαγωγή κωδικού, να είναι πλήρως παραμετροποιήσιμες από τους διαπιστευμένους χρήστες. Με αυτόν τον τρόπο, το σύστημα είναι πλήρως ελεγχόμενο από τον χρήστη, δίνοντάς του την ελευθερία να το παραμετροποιήσει σύμφωνα με τις δικές του, προσωπικές, προτιμήσεις και ανάγκες.

Τέλος, όσον αφορά την επέκταση των βασικών μεθόδων ελέγχου, το σύστημα θα μπορούσε να χρησιμοποιεί προηγμένους τρόπους ελέγχου, όπως βιομετρικά χαρακτηριστικά. Ωστόσο, κάτι τέτοιο θα σήμαινε την πιθανή αναβάθμιση του μικροελεγκτή και τη χρήση διάφορων εργαλείων επέκτασης. Χάρη στη φύση της πλατφόρμας του Arduino, κάτι τέτοιο είναι αρκετά απλό στην υλοποίηση, ενώ η συγγραφή κατάλληλου κώδικα δεν παρουσιάζει ιδιαίτερες δυσκολίες, καθώς έχουν αναπτυχθεί κατάλληλες βιβλιοθήκες από την παγκόσμια κοινότητα του Arduino. Ωστόσο, εγείρονται θέματα ασφάλεια και διαφύλαξης των προσωπικών στοιχείων των χρηστών, οπότε χρειάζεται διερεύνηση και προσεκτική προσέγγιση σε τέτοιους μηχανισμούς, ώστε να διασφαλίζονται οι προσωπικές πληροφορίες των χρηστών.



## 6 Βιβλιογραφία

Arduino, 2014. *Arduino - Frequently Asked Questions*. [Ηλεκτρονικό]

Available at: <http://www.arduino.cc/en/Main/FAQ>

[Πρόσβαση 10 Ιούνιος 2015].

Avison, D. & Shah, H., 1997. *The information systems development life cycle: A first course in information systems*. 1η επιμ. Νέα Υόρκη: McGraw-Hill.

Banzi, M. & Shiloh, M., 2014. *Make: Getting Started with Arduino: The Open Source Electronics Prototyping Platform*. 3η επιμ. Sebastopol : Maker Media.

Blum, J., 2013. *Exploring Arduino: Tools and Techniques for Engineering Wizardry*. 1η επιμ. Νέα Υόρκη: Wiley & Sons.

Cuartielles, D., 2013. *Arduino FAQ – With David Cuartielles*. [Ηλεκτρονικό]

Available at: <http://medea.mah.se/2013/04/arduino-faq/>

[Πρόσβαση 12 Ιούνιος 2015].

Gartner, K. & Uyeda, A., 1990. *Combination lock with an additional security lock*. United States of America, Ευρεσιτεχνία Αρ. 4,904,984.

Kaptelinin, V. & Nardi, B., 2012. Activity Theory in HCI: Fundamentals and Reflections. *Synthesis Lectures on Human-Centered Informatics*, 5(1), pp. 1-105.

Konicek, J. & Little, K., 1997. *Security, ID Systems and Locks: The Book on Electronic Access Control*. 1η επιμ. Burlington: Elsevier.

Kushner, D., 2011. The Making of Arduino. *IEEE Spectrum* , Τόμος 26.

Malaiya, Y. και συν., 1994. *The Relationship Between Test Coverage and Reliability*, Fort Collins: Department of Computer Science, Colorado State University.

Myers, G., Sandler, C. & Badgett, T., 2011. *The Art of Software Testing*. 3η επιμ. Νέα Υόρκη: John Wiley & Sons.

Nielsen, J., 1994. *Usability Inspection Methods*. 1η επιμ. Νέα Υόρκη: John Wiley & Sons.

Norman, T., 2011. *Electronic Access Control*. 1η επιμ. Λονδίνο: Butterworth-Heinemann.

Pendharkara, P., Rodgerb, J. & Subramaniana, G., 2008. An empirical study of the Cobb–Douglas production function properties of software development effort. *Information and Software Technology*, 50(12), p. 1181–1188.

Pohl, K., 2010. *Requirements Engineering: Fundamentals, Principles, and Techniques*. 1η επιμ. Βερολίνο: Springer.

Simon, M., 2011. *Programming Arduino: Getting Started With Sketches*. 1η επιμ. Νέα Υόρκη: Tab Books.

Strack, G., Ornatska, M., Pita, M. & Katz, E., 2008. Biocomputing Security System: Concatenated Enzyme-Based Logic Gates Operating as a Biomolecular Keypad Lock. *Journal of the American Chemical Society*, 130(13), p. 4234–4235.

Wiegers, K. & Beatty, J., 2013. *Software Requirements*. 3η επιμ. Redmond: Microsoft Press.

Wohlin, C. και συν., 2012. *Experimentation in Software Engineering*. 1η επιμ. Βερολίνο: Springer.