

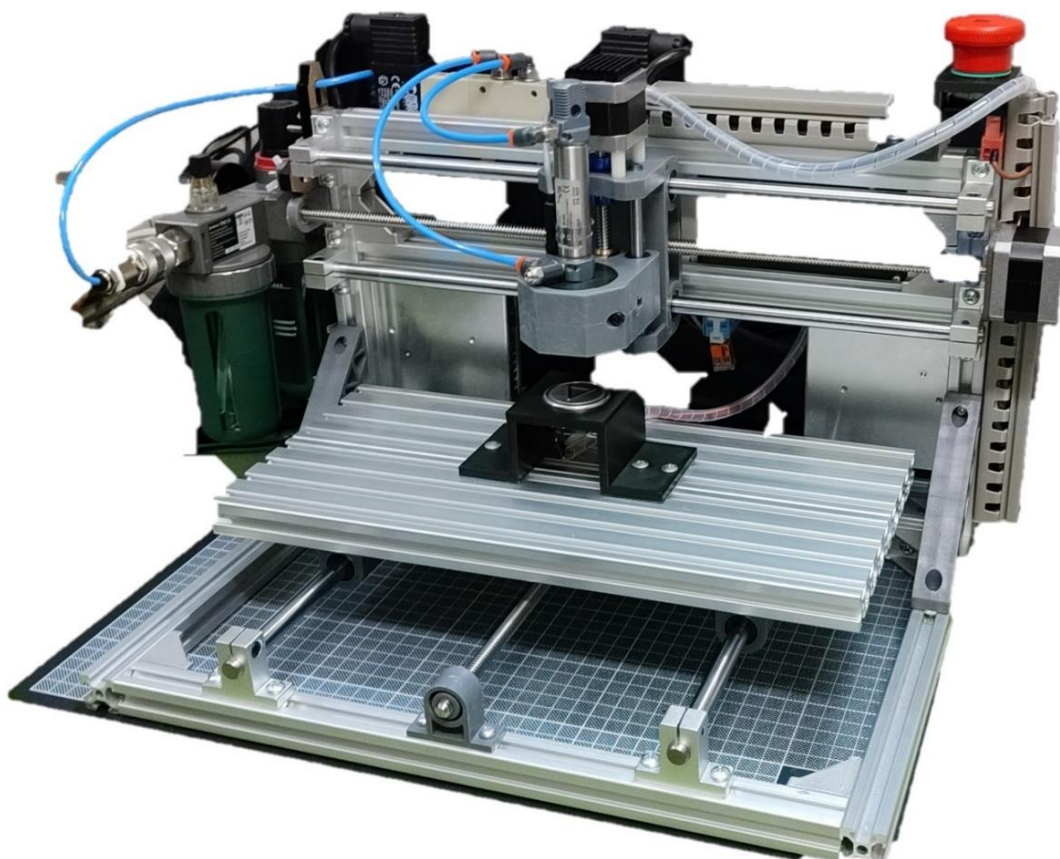
Θέμα διπλωματικής εργασίας :

ΜΗΧΑΝΙΣΜΟΣ ΕΛΕΓΧΟΥ ΔΙΑΡΚΕΙΑΣ ΖΩΗΣ ΚΟΜΒΙΩΝ

BUTTON LIFETIME CONTROL MECHANISM

Όνοματεπώνυμο φοιτητή : Ζωηρός Πέτρος

Επιβλέπων καθηγητής : Τσαγκάρης Απόστολος



Εξεταστική επιτροπή:

Τσαγκάρης Απόστολος

Ξανθός Στέλιος

Μπεχτσής Δημήτριος

Τμήμα Μηχανικών Παραγωγής και Διοίκησης
ΠΜΣ ΕΦΑΡΜΟΣΜΕΝΑ ΣΥΣΤΗΜΑΤΑ ΑΥΤΟΜΑΤΟΠΟΙΗΣΗΣ

Θεσσαλονίκη 2022

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Η παρούσα Διπλωματική Εργασία και τα συμπεράσματά της, σε οποιαδήποτε μορφή, αποτελούν συνιδιοκτησία του Τμήματος Μηχανικών Παραγωγής και Διοίκησης του Διεθνούς Πανεπιστημίου Ελλάδος και του φοιτητή. Οι προαναφερόμενοι διατηρούν το δικαίωμα ανεξάρτητης χρήσης και αναπαραγωγής (τμηματικά ή συνολικά) για διδακτικούς και ερευνητικούς σκοπούς. Σε κάθε περίπτωση πρέπει να αναφέρεται ο τίτλος, ο συγγραφέας, ο επιβλέπων και το τμήμα του ΔιΠαΕ.

Η έγκριση της παρούσας Διπλωματικής Εργασίας από το Τμήμα Μηχανικών Παραγωγής και Διοίκησης δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα εκ μέρους του Τμήματος.

Ο υπογεγραμμένος δηλώνω υπεύθυνα ότι η παρούσα Διπλωματική Εργασία είναι εξ' ολοκλήρου δικό μου έργο και συγγράφηκε ειδικά για τις απαιτήσεις του προγράμματος σπουδών του Τμήματος Μηχανικών Παραγωγής και Διοίκησης.

Δηλώνω υπεύθυνα ότι κατά τη συγγραφή ακολούθησα την πρόπουσα ακαδημαϊκή δεοντολογία αποφυγής λογοκλοπής και έχω αποφύγει οποιαδήποτε ενέργεια που συνιστά παράπτωμα λογοκλοπής.

Πέτρος Ζωηρός

28/09/2022

(Όνομα, Υπογραφή, Ημερομηνία)

Περιεχόμενα

Περιεχόμενα.....	3
Ευχαριστίες.....	5
Περίληψη.....	6
Εισαγωγή.....	8
Κεφάλαιο 1 ^ο : Βιβλιογραφική ανασκόπηση.....	10
• 1 ^η βιβλιογραφική ανασκόπηση.....	11
• 2 ^η βιβλιογραφική ανασκόπηση.....	14
• Ανάλυση πνευματικών συστημάτων.....	16
Κεφάλαιο 2 ^ο : Ποιο ήταν οι στόχοι.....	18
• Μηχανισμός ελέγχου διάρκειας ζωής κομβίου.....	18
• Επίτευξη στόχου.....	20
• Τι είναι το κομβίο ;.....	21
Κεφάλαιο 3 ^ο : Τι χρησιμοποιήθηκε.....	23
• Περιγραφή υλικών.....	23
• Περιγραφή διαδικασίας κατασκευής.....	36
• Περιγραφή διαδικασίας κατασκευής 3D εκτυπωμένων υλικών.....	40
Κεφάλαιο 4 ^ο : Περιγραφή προγράμματος, κατανόηση αυτοματισμού.....	46
• Περιγραφή προγράμματος για την κατανόηση του αυτοματισμού.....	46
• Ηλεκτρολογικό σχέδιο συνδεσμολογίας σε Eplan.....	48
Κεφάλαιο 5 ^ο : Ανάλυση προγράμματος C.....	52
• Ανάλυση προγράμματος C.....	52
• Ανάλυση κώδικα C.....	53
Κεφάλαιο 6 ^ο : Ανάλυση προγράμματος C#.....	63
• Ανάλυση προγράμματος C#.....	63

• Ανάλυση κώδικα C#.....	64
Κεφάλαιο 7° : Ανάλυση προγράμματος SQL.....	72
• Ανάλυση προγράμματος SQL.....	72
• Ανάλυση κώδικα SQL.....	73
Κεφάλαιο 8° : Δοκιμές.....	74
• Ανάλυση δοκιμών στρόγγυλων κομβίων.....	74
• Κομβίο Α.....	74
• Κομβίο Β.....	76
• Κομβίο Γ.....	77
• Ανάλυση δοκιμών τετράγωνων κομβίων.....	78
• Κομβίο Δ.....	78
• Κομβίο Ε.....	79
• Κομβίο ΣΤ.....	80
• Συγκεντρωτική ανάλυση αποτελεσμάτων.....	81
Συμπεράσματα και προτάσεις για την πιθανή εξέλιξη του συστήματος.....	82
Βιβλιογραφία – Αναφορές.....	85
Παραρτήματα :.....	86
• Datasheets.....	87

Ευχαριστίες

Θα ήθελα ιδιαίτερα να ευχαριστήσω τον επιβλέπων καθηγητή μου κ. Απόστολο Τσαγκάρη, που μέσα από την εμπιστοσύνη που έδειξε στο πρόσωπο μου κατάφερα να επιτύχω τον στόχο μου, αβίαστα και προπαντός ευχάριστα.

Επίσης ευχαριστώ τα μέλη της τριμελούς επιτροπής για την αξιολόγηση της παρούσης διπλωματικής εργασίας.

Ευχαριστώ του φίλους μου κ. Στέλιο Αγγελίδη και κ. Παναγιώτη Σιδηρόπουλο για την παραχώρηση των αεροσυμπιεστών τους έτσι ώστε να πραγματοποιηθούν και να ολοκληρωθούν οι συνεχείς και χρονοβόρες δοκιμές.

Τέλος ένα μεγάλο ευχαριστώ για την υποστήριξη και υπομονή που υπέδειξαν όλα τα μέλη της οικογένειας μου καθ' όλο το διάστημα εκπόνησης της διπλωματικής.

Περίληψη

Η παρούσα διπλωματική εργασία πραγματεύεται τον έλεγχο της διάρκειας ζωής των κομβίων.

Τα κομβία των μεγαλύτερων κατασκευαστικών κολοσσών έχουν πιστοποιήσεις άνω του ενός εκατομμύριο πατημάτων όμως η πράξη αποδεικνύει ότι το πραγματικό νούμερο είναι πάντα υποπολλαπλάσιο του.

Η ειδοποιός διαφορά του μηχανισμού ελέγχου που έχει σχεδιαστεί και αναπτυχθεί είναι ότι, το πνευματικό έμβολο που θα “πατά” και θα ενεργοποιεί το κομβίο δεν θα πιέζει κάθε φορά σταθερά σε ένα σημείο όπως πραγματοποιείται σε αντίστοιχα συστήματα, αλλά θα το κάνει μετακινούμενο συνεχώς στους άξονες x και y έτσι ώστε να επιτευχθεί ανομοιόμορφη κατανομή των πιέσεων που ασκούνται κάθε φορά.

Με την μέθοδο αυτή αυξάνεται το ποσοστό αστοχίας τους στην πράξη μετά από μόλις μερικά χιλιάδες πατήματα, αντικατοπτρίζοντας την πραγματικότητα και όχι μια απλή εργαστηριακή δοκιμή.

Η κατασκευή του μηχανισμού έχει επιτευχθεί με την χρήση CNC, ενώ για το πάτημα του κομβίου επιλέχθηκε ένα πνευματικό έμβολο. Η κίνηση του εμβόλου στους άξονες x, y με τα πραγματοποιείται με την βοήθεια των stepper motor του CNC.

Ο έλεγχος του ολοκληρωμένου μηχανισμού επιτυγχάνεται με ειδικά κατασκευασμένο πρόγραμμα σε C#, όπως επίσης υπάρχει καταγραφή και οπτικοποίηση των αποτελεσμάτων με την χρήση ενός SQL server.

Βασικός πυλώνας της υλοποίησης αυτού του project είναι το να κρατηθεί σε πολύ χαμηλά επίπεδα το κόστος ανάπτυξης αλλά και συντήρησης για να ανταποκρίνεται στην σημερινή πραγματικότητα της συνεχούς αύξησης τιμών.

Λέξεις κλειδιά : διάρκεια ζωής, κομβίο, πάτημα, ποσοστό αστοχίας, CNC, πνευματικό έμβολο, C, C#, SQL, χαμηλό κόστος ανάπτυξης.

Abstract

This master thesis deals with the lifetime of buttons.

The buttons of the biggest manufacturing giants have certifications for more than one million presses, but the practice proves that the real number is always an aliquot part of it.

The noticeable difference of this control mechanism that has been designed and developed is that the pneumatic piston that will “press” and activate the button will not press only at one point each time as it does in respective systems, but will make it constantly moving on the axles x and y so as to achieve an uneven distribution of the pressures exerted each time.

This method increases their failure rate in practice after just a few thousand clicks, reflecting reality rather than a simple laboratory test.

The construction of the mechanism has been achieved using a CNC, while a pneumatic piston was selected for the push of the button. The movement of the piston on the x, y axes with them is carried out with the help of CNC stepper motors.

The control of the complete mechanism is achieved with a specially built program in C #, as well as recording and visualizing the results using a SQL server.

The main pillar of the implementation of this project is to keep the costs of development and maintenance at very low levels to meet the current reality of continuous price increases.

Key words : lifetime, button, press, failure rate, CNC, pneumatic piston, C, C#, SQL, low development cost.

Εισαγωγή

Στους σύγχρονους ανελκυστήρες το design και η αισθητική έχουν εισχωρήσει στην κουλτούρα και έχουν μεταμορφώσει τα παλιά μουντά κονσερβοκούτια κάθετης μετακίνησης σε σύγχρονα και ευχάριστα μέσα μεταφοράς με υψηλές ταχύτητες και βέλτιστους χρόνους εξυπηρέτησης.

Η εξέλιξη αυτή συμπαρέσυρε όλα τα εξαρτήματα που περιλαμβάνει ένας ανελκυστήρας.

Έτσι λοιπόν μαζί με όλα τα άλλα και τα κομβία πλέον απέκτησαν σύγχρονα χαρακτηριστικά και κατασκευάζονται σε πολλά σχήματα, μεγέθη, χρώματα και υλικά.

Αυτό έχει σαν αποτέλεσμα την αναγκαστική χρήση νέων τεχνολογιών με υλικά που δεν αντέχουν στο πέρασμα του χρόνου και έχουν πολύ μικρές ανοχές και αντοχές. Όπως επίσης έχουν συρρικνωθεί σε μέγεθος, κάτι το οποίο δυσκολεύει ακόμη περισσότερο τον σκοπό τους.

Με όλα τα παραπάνω γεννήθηκε η ανάγκη ενός τρόπου δοκιμής και καταγραφής όλων των δεδομένων, για τον υπολογισμό της διάρκειας ζωής των διακοπών των κομβίων.

Στην υπάρχουσα συσκευή δοκιμής ζωής των κομβίων, χρησιμοποιείται μια συσκευή τροφοδοσίας αέρα για να οδηγήσει τη συσκευή επαφής, ένα πνευματικό έμβολο δηλαδή σε παλινδρομική κίνηση, προσομοιώνοντας το πάτημα ενός ανθρώπινου δακτύλου.

Γενικότερα οι συσκευές δοκιμής διάρκειας ζωής των κομβίων έχουν μια περίπλοκη δομή και μπορεί να έχουν μια διάρκεια ζωής λιγότερο από 10.000 πατήματα, στις 50.000 δοκιμές θα έχουν φτάσει στο όριο τους και για πάνω από 100.000 πατήματα οι μηχανές της δοκιμής διάρκειας ζωής θα παρουσιάσουν φαινόμενα κόπωσης και τέτοια κόπωση θα προκαλέσει φθορά και καταστροφή τους.

Η έξοδος της επαφής τους δεν είναι ομοιόμορφη, γεγονός που προκαλεί βλάβη στον διακόπτη του κομβίου και δεν μπορεί να ανιχνεύσει με ακρίβεια την πραγματική διάρκεια ζωής της επαφής του κομβίου, επηρεάζοντας έτσι τις τεχνικές προδιαγραφές του.

Ο μηχανισμός οδηγεί το πνευματικό έμβολο σε κάθετη κίνηση και η διαφορά μετατόπισης που δημιουργείται από τον αέρα μετατρέπεται σε πίεση προς τα κάτω του

μηχανισμού πίεσης, η δύναμη πίεσης που εφαρμόζεται είναι ομοιόμορφη και έτσι μπορεί να αντέξει για πάνω από 1 εκατομμύριο φορές.

Μετά το πάτημα, η πίεση διατηρείται ομοιόμορφα και έτσι ο διακόπτης του κομβίου δεν καταστρέφεται, ενώ ο αριθμός των πιέσεων του πνευματικού εμβόλου καταγράφεται από τον μετρητή για να μετρηθεί με ακρίβεια η πραγματική ζωή του διακόπτη του κομβίου.

Το πειραματικό μοντέλο μας έχει τα πλεονεκτήματα της απλής δομής, της βολικής λειτουργίας και του χαμηλού κόστους κατασκευής και συντήρησης.

Κεφάλαιο 1^ο : Βιβλιογραφική ανασκόπηση

Κάθε ερευνητική εργασία θα πρέπει να αξιοποιεί την υπάρχουσα επιστημονική γνώση αλλά και τα ερευνητικά δεδομένα του χώρου και σε καμία περίπτωση δεν θα πρέπει να ξεκινά από το μηδέν.

Μέσω της συστηματικής μελέτης των ερευνών που έχουν διεξαχθεί στο συγκεκριμένο πεδίο και διαμέσου της βιβλιογραφικής ανασκόπησης θα πραγματοποιήσουμε μια παρουσίαση της σύνθεσης των βιβλιογραφικών πηγών που μελετήθηκαν.

Θα ακολουθήσει η κριτική αποτίμησή των δύο ερευνών μέσω του σχολιασμού έτσι ώστε να αποκτηθεί μια σαφή εικόνα για το θέμα που διερευνούμε και να αναδειχθεί το όποιο ερευνητικό έλλειμμα του πεδίου, έτσι ώστε να μπορέσουμε να αναπτύξουμε λύσεις.

Στο συγκεκριμένο ερευνητικό πεδίο οι προσεγγίσεις του προβλήματος σε θεωρητικό επίπεδο είναι ελάχιστες, για το λόγο ως πρώτη βιβλιογραφική ανασκόπηση επιλέχθηκε μια σύγχρονη και εμπειριστατωμένη μελέτη που αφορά την εφαρμογή της μεθόδου της επιταχυνόμενης δοκιμής της διάρκειας ζωής των μικροδιακοπών.

Η δεύτερη βιβλιογραφική ανασκόπηση αφορά μια ευρεσιτεχνία, δηλαδή μια πατέντα, η οποία είναι από τις Ηνωμένες Πολιτείες και περιγράφει την κατασκευή ενός συστήματος ελέγχου κομβίων στην γενικότερη του μορφή με μια πληθώρα διαφορετικών προσεγγίσεων.

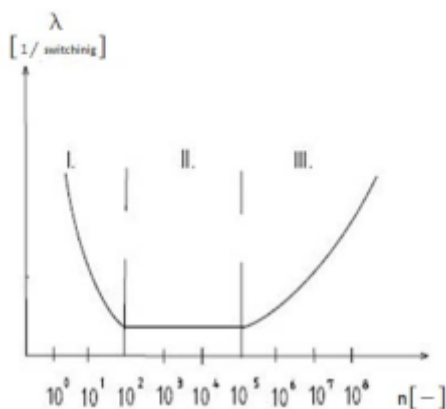
1^η βιβλιογραφική ανασκόπηση :

Η παρούσα βιβλιογραφική ανασκόπηση εξετάζει μια θεωρητική προσέγγιση ανάλυσης μιας μεθόδου επιταχυνόμενης δοκιμής της διάρκειας ζωής των μικροδιακοπών. Θα επικεντρωθούμε και θα εντυπώσουμε στα σημεία που δίνουν αξία στην δική μας έρευνα.

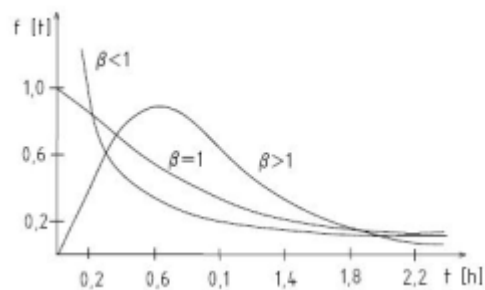
Οι Vivien Sipkás, Gabriella Bognár αναφέρουν στην δημοσίευση τους *The Application of Accelerated Life Testing Method for Micro Switches* (2018) :

Η ανάλυση στοχεύει να λάβει υπόψη πολλούς παράγοντες που επηρεάζουν, και επιχειρεί να παράσχει την στατιστική διασφάλιση της αξιοπιστίας αλλά και να δώσει στατιστικά αξιόπιστα δεδομένα διάρκειας ζωής σε σύντομο χρονικό διάστημα.

Για το επιτύχει αυτό χρησιμοποιεί την κατανομή Weibull όπου εφαρμόζεται για τη διερεύνηση του ποσοστού αστοχίας στην καμπύλη διάρκειας ζωής τύπου (δηλαδή σχήματος) μανιέρας του προϊόντος. Έτσι καθορίζεται και η διανομή Weibull ικανή να μοντελοποιήσει τη λειτουργία ρυθμού κινδύνου σε σχήμα μανιέρας.



Εικόνα καμπύλη bathtub



Εικόνα κατανομής Weibull

Η σημασία αυτής της κατανομής έγκειται στην ικανότητά της να μοντελοποιεί μονοτονικά καθώς και μη μονοτονικά ποσοστά αστοχίας, τα οποία είναι αρκετά κοινά σε προβλήματα διάρκειας ζωής και αξιοπιστίας.

Έχουν συλλεχθεί οι πιο συνηθισμένες αστοχίες των μικροδιακοπών, οι οποίες είναι απαραίτητες για τη δοκιμή και την κατανόηση του σχεδιασμού του εξοπλισμού δοκιμής.

Οι μικροδιακόπτες ανοίγουν και κλείνουν τις επαφές του ηλεκτρικού κυκλώματος κατά τη λειτουργία του. Κατά τη χρήση τους οι διακόπτες πρέπει να εκτελέσουν ένα μεγάλο αριθμό από διακοπές και κύκλους κλεισίματος, έτσι είναι η ζωή τους επηρεάζεται και καθορίζεται από την «υλική μετανάστευση» και φθορά των επιφανειών επαφής του.

Όμως τα τελευταία χρόνια, οι διακόπτες έχουν υποστεί εκτεταμένη σμίκρυνση.

Επίσης η αξιοπιστία των διακοπών έχει αυξηθεί πολύ, υιοθετώντας πρωτόγνωρες υπηρεσίες μεγάλης διάρκειας ζωής των ενεργών και των παθητικών εξαρτημάτων.

Ένας διακόπτης πρέπει να παρέχει έως και 10^6 ανοιγοκλεισίματα χωρίς σφάλματα.

Όμως κατά την λειτουργία του ένα ηλεκτρικό τόξο μπορεί να σχηματιστεί ανοίγοντας ή διακόπτοντας το κύκλωμα, οπότε και αυτό επηρεάζει το μέγεθος, τη χρηστικότητα αλλά και τη διάρκεια ζωής των επαφών του.

Οι μικροδιακόπτες έχουν έναν αριθμό από δυσλειτουργίες, μερικές από τις οποίες παρουσιάζονται ως τυπικές αποτυχίες.

Ένα από τα πιο κοινά προβλήματα ενός μικροδιακόπτη είναι η παραμόρφωση λόγω υψηλής θερμοκρασίας. Υπάρχουν διάφοροι λόγοι για αυτό το πρόβλημα, παραδείγματος χάρη υπερθερμαίνεται λόγω υπερφόρτωσης ή υπερεναλλαγής.

Όταν το κύκλωμα είναι κλειστό και διακόπτεται, δημιουργείται όπως αναφέρθηκε προηγουμένως ένα ηλεκτρικό τόξο. Η μετανάστευση των υλικών εσωτερικά της επαφής είναι η συνέπεια αυτού του φαινομένου, λόγω της παραγωγής θερμότητας αλλά και της αύξησης της προσωρινής ηλεκτρικής αντίστασης των υλικών. Προκύπτει επίσης η υπερθέρμανση των προ συναρμολογημένων εξαρτημάτων όπως τα ελατήρια επαναφοράς.

Μια περαιτέρω κατανομή δείχνει ότι όταν το ρεύμα που ρέει μέσα στο διακόπτη είναι μεγαλύτερο από την επιτρεπόμενη τιμή ή σε περίπτωση βραχυκυκλώματος, το φύλλο ελατηρίου μπορεί να παραμορφωθεί χάνοντας την ελαστικότητά του και παύοντας την αρχική του λειτουργία. Επομένως, το μηχανήμα δεν μπορεί να απενεργοποιηθεί ή να παραμείνει αναμμένο γιατί μπορεί να προκληθεί απώλεια υλικού με την στατική εκκένωση μέσω των επαφών.

Επίσης εξαιτίας ηλεκτρικών, θερμικών και μηχανικών καταπονήσεων, η επαφή μπορεί να σπάσει σε έναν ορισμένο αριθμό διακοπών λόγω της σύσφιξης που δημιουργείται.

Ένα άλλο τυπικό πρόβλημα αστοχίας είναι η φθορά του κουμπιού του μικροδιακόπτη, το οποίο θα μπορούσε να είναι το αποτέλεσμα της κακής ποιότητας κατασκευής του.

Μετά τη μελέτη του μαθηματικού μοντέλου, τις επιταχυνόμενες δοκιμές και τη διανομή Weibull, το επόμενο έργο της έρευνας σύμφωνα με τους συντάκτες της έρευνας είναι ο σχεδιασμός και η σύνταξη ενός πάγκου εργασίας κατάλληλου για δοκιμή μικροδιακοπών, όπου στη συνέχεια θα οριστούν οι παράμετροι μέτρησης, ακολουθούμενοι από δοκιμές, αξιολόγηση και συλλογή των δεδομένων μετά από κάθε δοκιμή.

Ο περαιτέρω στόχος είναι η παροχή λεπτομερούς αριθμητικής ανάλυσης της επιταχυνόμενης διάρκειας ζωής στη βάση των πειραματικών δεδομένων που έχουν συλλεχθεί.

Το ερευνητικό έλλειμα στην συγκεκριμένη περίπτωση είναι προφανές και έχει ήδη δηλωθεί από τον ερευνητή. Καθώς αναφέρει ότι σε επόμενο έργο της έρευνας θα σχεδιαστεί και θα κατασκευαστεί ένα παρόμοιο σύστημα ή μηχανισμός για τον έλεγχο των μικροδιακοπών.

Όμως ο ερευνητής μη έχοντας εμπειρία στο χώρο αλλά και καμία άλλη ανάδραση από πειραματικά δεδομένα, πιθανόν να μην μπορέσει να καταφέρει να σχεδιάσει μηχανισμό παρόμοιο με την προσέγγιση της παρούσης διπλωματικής εργασίας.

2^η βιβλιογραφική ανασκόπηση :

Η παρούσα βιβλιογραφική ανασκόπηση εξετάζει την ανάγκη για ένα χαμηλού κόστους και αξιόπιστο σύστημα για τη δοκιμή της λειτουργικότητας των κουμπιών.

Οι Adrien και Baïma, αναφέρουν στην δημοσίευσή τους *Push-button testing system* (2013) :

Πεδίο της εφεύρεσης

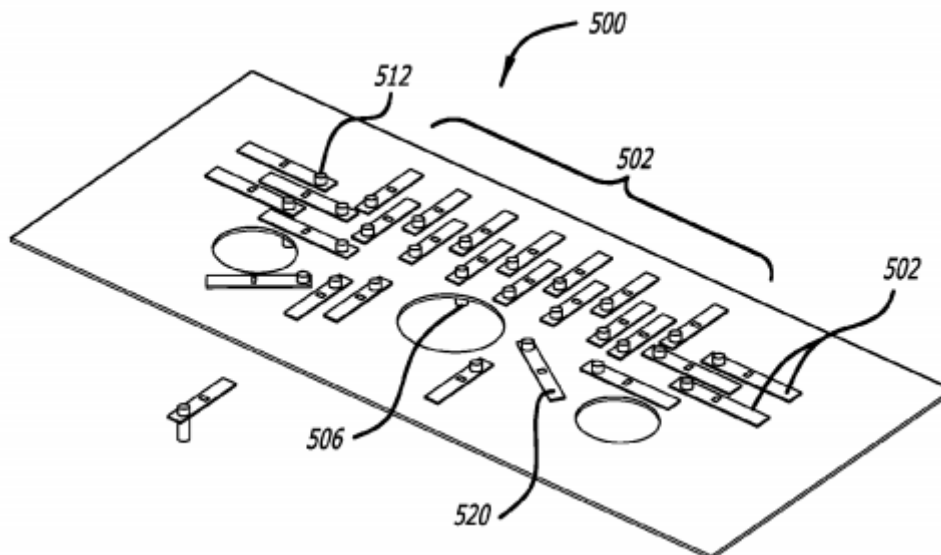
Παρέχεται ένα σύστημα για τη δοκιμή ενός διακόπτη με μπουτόν. Το σύστημα για τη δοκιμή ενός διακόπτη με μπουτόν περιλαμβάνει μια συσκευή δοκιμής διακόπτη.

Η συσκευή δοκιμής διακόπτη έχει μια εύκαμπτη γλωττίδα προσαρτημένη σε ένα μέλος ώθησης στο άκρο της εύκαμπτης γλωττίδας.

Ένας αισθητήρας είναι προσαρτημένος στην εύκαμπτη γλωττίδα. Ο αισθητήρας παράγει ένα σήμα που αλλάζει σε σχέση με μια παραμόρφωση της εύκαμπτης γλωττίδας. Ένα σύστημα συλλογής δεδομένων συνδέεται με τη συσκευή δοκιμής διακόπτη και λαμβάνει σήματα από τον αισθητήρα.

Είναι ένα πολύπλοκο και ασαφές σύστημα για τη δοκιμή πολλών διακοπών με κουμπιά.

Αυτή η εφεύρεση αναφέρεται γενικά σε συστήματα δοκιμών και ειδικότερα σε συστήματα για τη δοκιμή εξαρτημάτων με κουμπιά.



Εικόνα Α

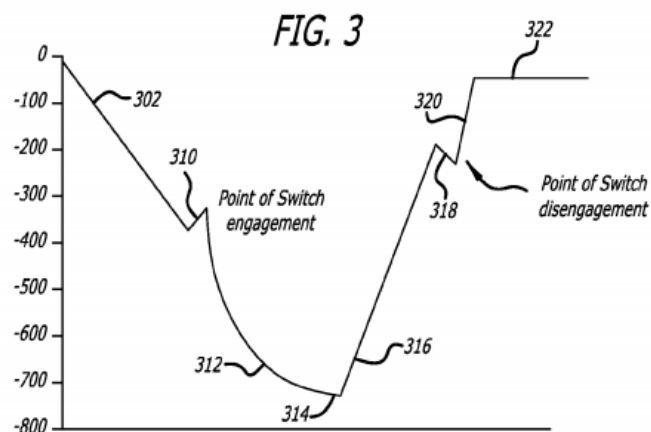
Το γράφημα στο ΣΧ. Το 3 επιτρέπει την γρήγορη οπτική ανάλυση είτε επιτυχίας είτε αποτυχίας του κουμπιού. Εάν το γράφημα περιλαμβάνει τα σημεία εμπλοκής και απεμπλοκής (στα 310 και 318, αντίστοιχα στο Σχ. 3), το κουμπί μπορεί να θεωρηθεί ότι λειτουργεί.

Εάν λείπει κάποιο σημείο από το γράφημα, το δίνει αντίθετη κατεύθυνση (προς τα πάνω). Η δύναμη f_1 αναγκάζει την εύκαμπτη γλωττίδα 102 να κινείται προς την καθοδική κατεύθυνση όταν η δύναμη εφαρμόζεται στην κατεύθυνση προς τα κάτω (στο 220). Η εύκαμπτη γλωττίδα 102 κινείται επίσης προς τα πάνω όταν η κατεύθυνση της δύναμης f_1 αλλάζει για να πάει προς τα πάνω (στο 220).

Καθώς η εύκαμπτη γλωττίδα 102 κινείται, κάμπτεται, γεγονός που προκαλεί μια αλλαγή στην αντίσταση στην ευαίσθητη στην παραμόρφωση αντίσταση 104 που μπορεί να είναι ουσιαστικά ανάλογη με την ποσότητα δύναμης που επιβάλλεται στην εύκαμπτη γλωττίδα 102.

Το ερευνητικό έλλειμμα στην συγκεκριμένη περίπτωση είναι ότι η εφεύρεση είναι πολύ γενική και απλά προσπαθεί να καλύψει όλες στις πιθανές επεκτάσεις ενός μηχανισμού ελέγχου, έτσι ώστε σε περίπτωση που επιθυμεί κάποιος άλλος να σχεδιάσει και να πατεντάρει κάτι αντίστοιχο να μην μπορεί ή εάν προσπαθήσει να κατασκευάσει αντίστοιχο προϊόν προς πώληση να πρέπει πληρώσει τον κάτοχο της πατέντας για να την χρησιμοποιήσει στην κατασκευή του.

Επιπλέον οι εφευρέτες δείχνουν να μην έχουν εμπειρία στον χώρο και είναι πιθανόν να μην μπορέσουν να σχεδιάσουν στην πράξη μηχανισμό παρόμοιο με την προσέγγιση της παρούσης διπλωματικής εργασίας.



Εικόνα Β

Ανάλυση πνευματικών συστημάτων :

Σύμφωνα με τους Σκαρπέτης, Μ., & Κουμπουλής, Φ. (2016) : τα πνευματικά συστήματα αυτομάτου ελέγχου χρησιμοποιούνται κυρίως σε βιομηχανικά περιβάλλοντα όπου απαιτείται ακριβής και γρήγορος έλεγχος.

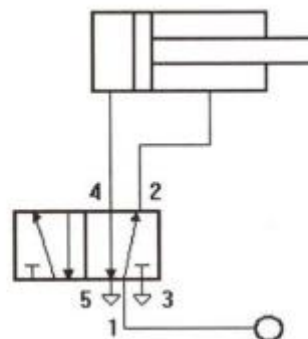
Με τα πνευματικά συστήματα ελέγχου μπορούν να υλοποιηθούν όλες οι συναρτήσεις δυαδικού ελέγχου καθώς επίσης μπορούν να υλοποιηθούν και να σχεδιαστούν τα περισσότερα κυκλώματα αυτομάτου ελέγχου που βασίζονται στην ψηφιακή δυαδική λογική.

Τα βασικά πλεονεκτήματα των πνευματικών συστημάτων είναι οι υψηλές ταχύτητες απόκρισης, η ικανοποιητική δυνατότητα συμπίεσης του αέρα, το μικρό κόστος αποθήκευσης του αέρα σε δεξαμενές και το μικρό κόστος κατασκευής των συστημάτων ελέγχου και ρύθμισης.

Για την περαιτέρω κατανόηση των πνευματικών κυκλωμάτων απαιτείτε η γνώση του τρόπου ελέγχου των πνευματικών κυλίνδρων.

Έλεγχος κυλίνδρου διπλής ενέργειας

Στο ακόλουθο διάγραμμα απεικονίζετε ένα έμβολο διπλής ενέργειας που ελέγχεται με βαλβίδα 5/2. Με τη χρήση της συγκεκριμένης βαλβίδας δίνεται η δυνατότητα στον αέρα να εκτονώνεται μέσω της βαλβίδας σε ξεχωριστές εξόδους κατά την έκταση και την σύμπτυξη.

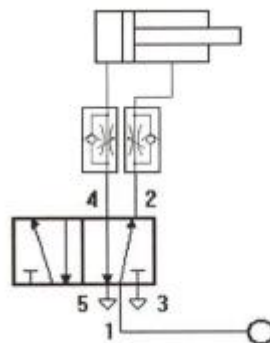


Σχήμα βαλβίδας 5/2

Έλεγχος ταχύτητας κυλίνδρου διπλής ενέργειας

Η ρύθμιση της ταχύτητας ενός κυλίνδρου διπλής ενέργειας γίνεται με την τοποθέτηση δύο ρυθμιζόμενων στραγγαλιστικών βαλβίδων που φέρουν ανεπίστροφες βαλβίδες αντίστοιχα για την κάθε ροή με την ίδια φορά. Όταν η βαλβίδα βρίσκεται στην αρχική της θέση, ο αέρας ρέει από την παροχή 1 στην έξοδο 4 και πριν οδηγηθεί στον κύλινδρο περνά από την στραγγαλιστική βαλβίδα ενώ ο αέρας από τον θάλαμο σύμπτυξης περνά από την ανεπίστροφη βαλβίδα και μέσω του βαλβίδας ελέγχου από το 2 στο 3 εκτονώνετε στην ατμόσφαιρα.

Με την αλλαγή της θέσης της βαλβίδας ελέγχου ο συμπιεσμένος αέρας περνά από το 1 και 2 της βαλβίδας στον κύλινδρο μέσω της στραγγαλιστικής βαλβίδας στον θάλαμο εκτόνωσης ενώ ο αέρας του θαλάμου σύμπτυξης ρέει μέσω της ανεπίστροφης βαλβίδας της γραμμής εκτόνωσης και μέσω του 4 και 5 ελευθερώνετε στην ατμόσφαιρα.



Σχήμα στραγγαλιστικές βαλβίδες

Έλεγχος της δύναμης πνευματικών κυλίνδρων

Η δύναμη που ασκεί κατά την έκταση ή την σύμπτυξη ένας πνευματικός κύλινδρος εξαρτάται από την πίεση του συμπιεσμένου αέρα και από τη διάμετρο του κυλίνδρου. Όταν όμως θέλουμε το έμβολο να έχει διαφορετική δύναμη ή να ρυθμίζετε όποτε απαιτείτε τότε χρησιμοποιείται ρυθμιστής πίεσης ο οποίος τοποθετείται μεταξύ της πόρτας Α της βαλβίδας κίνησης και της πόρτας εισαγωγής του εμβόλου.

Κεφάλαιο 2^ο : Ποιοι ήταν οι στόχοι

Μηχανισμός ελέγχου διάρκειας ζωής κομβίου :

Ακολουθεί μια αναλυτική περιγραφή του μηχανισμού δοκιμής ζωής των κομβίων.

Κύριο μέρος της συσκευής, η βάση και ο μηχανισμός αλλαγής θέσης χτυπήματος είναι ένα CNC το οποίο έχει τροποποιηθεί κατάλληλα έτσι ώστε να επιτύχουμε τον στόχο μας.

Τα stepper motor του CNC, οδηγούν τον μηχανισμό στους άξονες X & Y αλλά και στον Z για την βαθμονόμηση του συστήματος.

Χρησιμοποιείται μια συσκευή τροφοδοσίας αέρα μέσω ενός ρυθμιστή πίεσης που είναι ταυτόχρονα και παγίδα νερού αλλά και λιπαντήρας.

Η οδήγηση της συσκευής επαφής πραγματοποιείται μέσω μια βαλβίδας 5/2 και ενός πνευματικού εμβόλου το οποίο τίθεται σε παλινδρομική κίνηση, προσομοιώνοντας το πάτημα ενός ανθρώπινου δακτύλου.

Πιο αναλυτικά, ο μηχανισμός οδηγεί μέσω της βαλβίδας το πνευματικό έμβολο σε κάθετη κίνηση και η διαφορά μετατόπισης που δημιουργείται από τον αέρα μετατρέπεται σε πίεση προς τα κάτω του μηχανισμού, ενώ η δύναμη πίεσης που εφαρμόζεται είναι ομοιόμορφη.

Με αυτό τον τρόπο το πνευματικό έμβολο μπορεί να αντέξει να εκτείνεται και να συμπύσσεται για περισσότερες από 1 εκατομμύριο φορές. Επιπλέον μετά το πάτημα, δηλαδή την έκταση του εμβόλου, η πίεση διατηρείται ομοιόμορφα και έτσι ο διακόπτης του κομβίου δεν καταστρέφεται.

Ο αριθμός των πιέσεων (εκτάσεων) του πνευματικού εμβόλου καταγράφεται από ένα ψηφιακό μετρητή του προγράμματος για να μετρηθεί με ακρίβεια η πραγματική ζωή του διακόπτη του κομβίου, η πληροφορία μεταφέρεται μέσω ενός μαγνητικού διακόπτη και ενός ρελέ.

Υπάρχει και η ανάδραση από το κομβίο ότι πατήθηκε μέσω του ίδιου του διακόπτη του κομβίου, μέσω ενός δεύτερου ψηφιακού μετρητή του προγράμματος.

Για την ασφαλή χρήση του μηχανισμού τοποθετήθηκε σε εμφανή και εύκολα προσβάσιμο σημείο στην κορυφή του, ένας διακόπτης έκτακτης ανάγκης (μανιτάρι στοπ) ο οποίος διακόπτει την τάση εισόδου στα τροφοδοτικά.

Η τροφοδοσία επιτυγχάνεται με την χρήση 2 σταθεροποιημένων τροφοδοτικών στα 12Vdc για το Arduino & τους stepper motors και στα 24Vdc για τα πηνία της βαλβίδας 5/2 αλλά και του κυκλώματος ελέγχου εκτάσεων του πνευματικού εμβόλου.

Όλα τα παραπάνω οδηγούνται και καταγράφονται από ένα Arduino το οποίο είναι τοποθετημένο σε ειδικό shield για την οδήγηση των stepper motors, της βαλβίδας 5/2 που οδηγεί το πνευματικό έμβολο, του μηχανισμού καταγραφής των κινήσεων του εμβόλου και της καταγραφής των πατημάτων του ίδιου του δοκιμίου.

Το Arduino ελέγχεται μέσω της σειριακής θύρας (USB) του υπολογιστή με tailor made πρόγραμμα υλοποιημένο σε C#, ενώ η καταγραφή των εκτάσεων του εμβόλου και των πατημάτων του κομβίου μεταφέρεται σε μια βάση δεδομένων SQL μετά από κάθε κύκλο.

Αναλυτικότερα μέσω του προγράμματος των windows συνδεόμαστε με καλώδιο USB στο Arduino. Ενεργοποιούμε την σύνδεση μέσω επιλογής θύρας COM.

Έχουμε την δυνατότητα να επιλέξουμε μεταξύ χειροκίνητου και αυτόματου.

Στην επιλογή του χειροκίνητου ενεργοποιούμε ξεχωριστά τους βηματικούς κινητήρες προς όλες τις κατευθύνσεις με γρήγορη και συνεχή κίνηση ή με βήματα των οποίων την διάρκεια μπορούμε να επιλέξουμε. Επίσης ενεργοποιούμε την μόνιμη έκταση ή σύμπτυξη του εμβόλου. Όλα τα παραπάνω μας βοηθούν στο να βαθμονομήσουμε σωστά την συσκευή και να ξεκινήσουμε τον αυτόματο έλεγχο.

Στο αυτόματο, επιλέγουμε την εκκίνηση ή την παύση της δοκιμής όπως επίσης και τους πόσους κύκλους δοκιμών θα πραγματοποιήσουμε.

Επιπλέον υπάρχει η δυνατότητα να επιλέξουμε σε ποια απόσταση από το κέντρο θα πραγματοποιηθούν τα υπόλοιπα από τα 4 σημεία χτυπήματος.

Καθορίζουμε τα επιθυμητά χτυπήματα ανά θέση και φυσικά υπάρχει live οπτικοποίηση των εκτάσεων του εμβόλου και των επιτυχών πατημάτων του κομβίου.

Στόχος επίτευξης :

Τι ήθελα να επιτύχω με τον συγκεκριμένο μηχανισμό δοκιμής και γιατί υπήρχε η ανάγκη ενός τρόπου δοκιμής αλλά και καταγραφής όλων των δεδομένων, για τον υπολογισμό της διάρκειας ζωής των διακοπών των κομβίων ;

Γενικότερα όλες οι συσκευές δοκιμής διάρκειας ζωής έχουν μια περίπλοκη δομή και συνήθως έχουν μια διάρκεια ζωής λιγότερο από 10.000 δοκιμές, ενώ στις 50.000 δοκιμές θα έχουν ήδη φτάσει στο όριο τους και ξεπεράσουν τα 100.000 πατήματα θα παρουσιαστούν έντονα φαινόμενα κόπωσης και αυτή η κόπωση θα προκαλέσει την φθορά και την καταστροφή τους.

Επιπλέον η έξοδος της επαφής τους δεν είναι πάντα ομοιόμορφη, γεγονός που προκαλεί βλάβη στον διακόπτη του κομβίου ενώ δεν μπορούν να ανιχνεύσουν με ακρίβεια την πραγματική διάρκεια ζωής της επαφής του κομβίου, επηρεάζοντας έτσι τις τεχνικές προδιαγραφές του.

Το μοντέλο μας έχει τα πλεονεκτήματα της απλής δομής, της βολικής λειτουργίας και του χαμηλού κόστους κατασκευής και συντήρησης.

Ειδοποιός διαφορά αποτελεί το γεγονός ότι όλοι οι μηχανισμοί πραγματοποιούν μια και μόνο κάθετη κίνηση στο ίδιο πάντα σημείο. Όμως στην πραγματικότητα αυτό δεν συμβαίνει ποτέ, καθώς τα κομβία δεν τα πατάμε ποτέ στο ίδιο σημείο.

Το μοντέλο που αναπτύχθηκε εφαρμόζει μια διαφορετική προσέγγιση στο θέμα, γιατί το σημείο πατήματος πάνω στο προς δοκιμή κομβίο αλλάζει σε έως και 5 διαφορετικές θέσεις.

Το δεύτερο πολύ αξιόλογο σημείο επίτευξης στο μηχανισμό είναι ότι καταγράφει όχι μόνο ότι δόθηκε το σήμα για έλεγχο, αλλά και ότι ο μηχανισμός χτυπήματος λειτούργησε μέσω της ανάδρασης από τον μαγνητικό αισθητήρα, όπως επίσης και ότι το κομβίο προς δοκιμή πατήθηκε και ενεργοποιήθηκε.

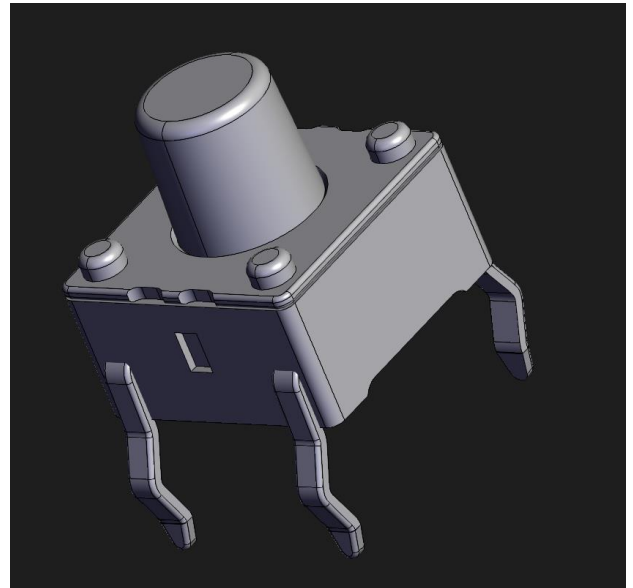
Έτσι οι καταγραφές αποτυπώνουν την πλήρη πραγματικότητα και δεν θεωρούνται μόνο πλέον εργαστηριακές μετρήσεις αλλά πραγματικά δεδομένα.

Τι είναι το κομβίο ;

Ποιο συγκεκριμένα τι είναι ο διακόπτης του κομβίου ;

Ο διακόπτης στα αγγλικά «Push Button» είναι ένας τύπος διακόπτη που αποτελείται από έναν απλό ηλεκτρικό μηχανισμό ή μηχανισμό διακόπτη αέρα για να ενεργοποιηθεί ή να απενεργοποιηθεί κάτι.

Το κομβίο είναι κατασκευασμένο συνήθως από ισχυρά και ανθεκτικά υλικά όπως μέταλλο ή πλαστικό. Οι διακόπτες των κομβίων παρέχονται σε διάφορα μεγέθη και σχήματα.



Εικόνα διακόπτη κομβίου

Ανάλογα με τον τύπο, θα μπορούσαν να ονομαστούν στιγμιαίας λειτουργίας ή λειτουργίας μανδάλωσης. Οι διακόπτες κομβίων χρησιμοποιούνται σε όλες τις βιομηχανικές και τις ιατρικές εφαρμογές και είναι πλέον μέρος της καθημερινής μας ζωής.

Σε ότι αφορά τις χρήσεις τους στον βιομηχανικό τομέα, τα κομβία είναι συχνά μέρος ενός μεγαλύτερου συστήματος και συνδέονται συνήθως μέσω μηχανικής σύνδεσης.

Αυτό σημαίνει ότι όταν πατηθεί ένα κομβίο μπορεί να προκαλέσει την ενεργοποίηση μιας λειτουργίας.

Ακολουθούν μερικά παραδείγματα διακοπών κομβίου καθώς αυτοί παρέχονται σε ευρύ φάσμα και σε διαφορετικές κατασκευές για διαφορετικές χρήσεις, όπως :

- Arcades: έχουν συνήθως πολύ έντονα χρώματα για να παροτρύνουν τους ανθρώπους να παίζουν
- Μηχανήματα ακινητοποίησης: συχνά γύρω από του βιομηχανικού τύπου κατασκευές, υπάρχει ένα κομβίο διακοπής έκτακτης ανάγκης, πολλές φορές τα κομβία αυτά τοποθετούνται στον τοίχο
- Διακόπτες επαναφοράς: αυτοί είναι συνήθως μικροί και απαιτούν το πάτημα ενός εργαλείου για να αποφευχθεί η ακούσια λειτουργία τους
- Κομβία αριθμομηχανής ή πληκτρολογίου: μια αριθμομηχανή χειρός έχει πολλούς μικρούς διακόπτες κομβίων, όπως αντίστοιχα και το ένα πληκτρολόγιο υπολογιστή
- Κομβία ασανσέρ – απαραίτητο στοιχείο λειτουργίας τους, καθώς χωρίς αυτά θα ήταν αδύνατη η κλήση και η αποστολή του θαλάμου.

Κεφάλαιο 3^ο : Τι χρησιμοποιήθηκε

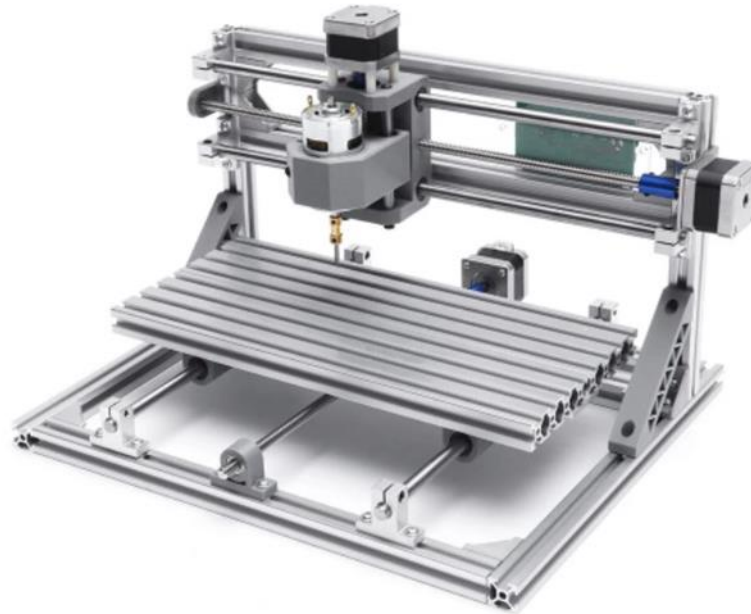
Περιγραφή υλικών :

Τα υλικά που χρησιμοποιήθηκαν επιλέχθηκαν προσεκτικά έτσι ώστε η συγκεκριμένη κατασκευή να έχει αξιοπιστία, ακρίβεια, αντοχή και χαμηλό κόστος κτήσης και συντήρησης.

Η παρακάτω λίστα μας δίνει επακριβώς στο κόστος κτήσης των υλικών που όλος τυχαίος κρατήθηκε κάτω από το ψυχολογικό φράγμα των 500€. Επίσης υπολογίζεται ότι απαιτούνται περίπου 9 εργατοώρες για την συναρμολόγηση και καλωδίωση του μηχανισμού.

Description	€
CNC 3018	121
Pneumatic valve 5/2 SOV 25 SOB 00	101,22
Mini-cylinder 16 AF 0025	68,65
Coils MW Φ8 24Vdc 2W	22,64
Magnetic switch	25,56
Connectors – piping's	33,04
Air pressure adjuster	25
Adjusters piping	15
CNC shield V4	7
Arduino Nano	7
3 x A4988 drivers	7,5
relay board 5Vdc	5
Relay 24Vdc	5
Emergency stop	5
Heat sinks	4,5
Cabling – trunks	9
Power supply 24Vdc	18,5
Power supply 12Vdc	18,5
Total :	499,11 €

Το δημοφιλές κινεζικό CNC router 3^{ov} αξόνων, αποτελεί την ποιο αξιόπιστη βάση για την εφαρμογή. Είναι πλήρες, στιβαρό και έρχεται με καλωδίωση για τους βηματικούς κινητήρες αλλά και με την πλακέτα για τον έλεγχο τους κάτι που βοηθά στην ευκολότερη διαχείριση του, ενώ οι μηχανισμοί είναι απλοί και μπορούν να βρεθούν εύκολα ανταλλακτικά άμεσα και γρήγορα.



Εικόνα CNC 3018

Οι βηματικοί κινητήρες που παρέχονται με το CNC 3018 είναι της σειράς 42HS, οι οποίοι δεν απαιτούν μεγάλο ρεύμα, ενώ έχουν πολύ καλή ακρίβεια θέσης.

42HS Series Hybrid Stepping Motors

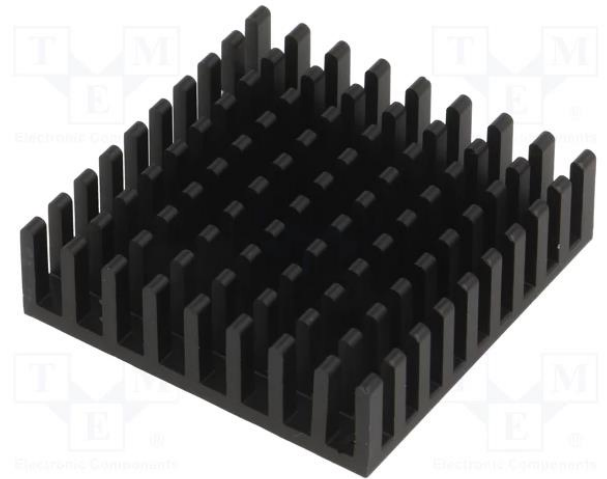


General Specifications

Step Angle Degree	1.8°
Step Angle Accuracy	±5%(full step, no load)
Temperature Rise	80°CMax
Ambient Temperature	-10°C — +50°C
Insulation Resistance	100MΩmin.500VDC
Dielectric Strength	500VAC for one minute
Shaft Radial Play	0.06 Max.(450g-load)
Shaft Axial Play	0.08 max.(450g-load)

Εικόνα βηματικών κινητήρων

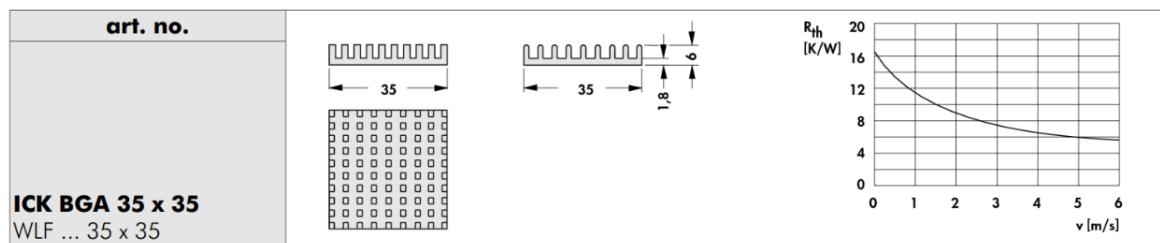
Όπως όμως είναι γνωστό, οι βηματικοί κινητήρες αναπτύσσουν υψηλές θερμοκρασίες ακόμη και όταν είναι σε στάση. Κατά τις δοκιμές παρατηρήθηκε ότι υπήρχε μεγαλύτερη αύξηση της θερμοκρασίας τους από το αναμενόμενο, άρα και κόπωση τους με αποτέλεσμα να χάνεται σταδιακά η ακρίβεια θέσης τους.



Εικόνα ψήκτρα αλουμινίου

Μετά από μια μικρή έρευνα για την καλύτερη ψύξη τους επιλέχθηκαν ψήκτρες αλουμινίου όπου τοποθετήθηκαν σε συνδυασμό θερμοαγωγίμης πάστας και μικρής ποσότητας κόλλας με πολύ ικανοποιητικά αποτελέσματα.

Πλέον υπάρχει μόνιμη πτώση της θερμοκρασίας και αύξηση της ακρίβειας θέσης τους.



Εικόνα χαρακτηριστικών της ψήκτρας αλουμινίου

Το τροφοδοτικό των 24V/5A που υπήρχε μαζί με το CNC αντικαταστάθηκε λόγω υψηλής κυμάτωσης στην έξοδο του με ένα MW LRS-75-24 και ένα MW LRS-75-12, έτσι ώστε να παρέχετε στο σύστημα ικανοποιητική ισχύς εφόσον χρειαστεί π.χ. σε πιθανό upgrade των βηματικών κινητήρων.



Εικόνα τροφοδοτικού MW LRS-75-24

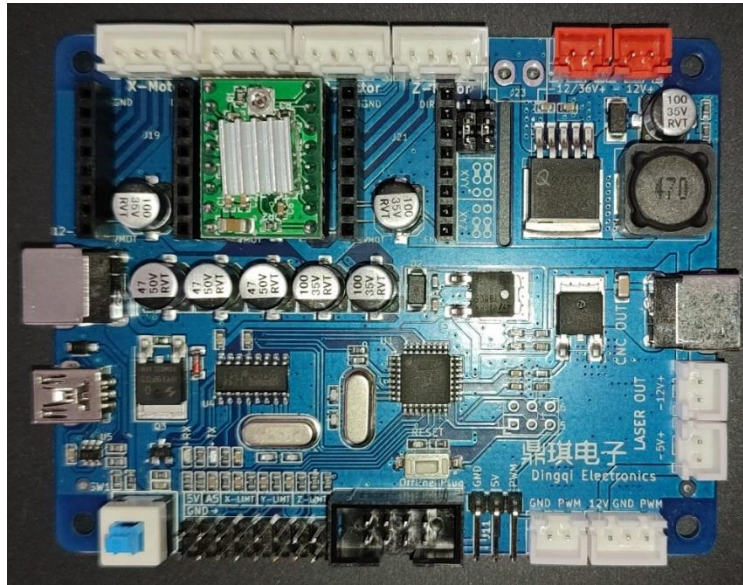
Η πλακέτα οδήγησης των βηματικών κινητήρων που παραλήφθηκε με το CNC παρόλο που μπορούσε να δεχθεί το SW μέσω της USB, όπως δηλαδή ανεβάζεις ένα πρόγραμμα στο Arduino.

Αντικαταστάθηκε όμως με την πλακέτα CNC shield V4, λόγω καλύτερης διαχείρισης κόστους σε περίπτωση βλάβης καθώς όλα τα υλικά της είναι διακριτά και επομένως υπάρχουν άμεσα διαθέσιμα στην αγορά ως ανταλλακτικά.

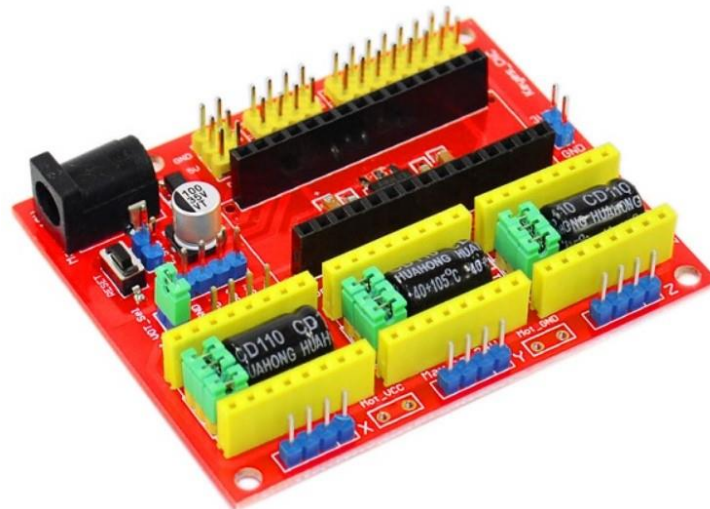
Η CNC shield V4 μπορεί να δεχθεί τους 3 drivers (A4988) των stepper motors όπως επίσης και ένα Arduino Nano.

Η τροφοδοσία της πλακέτας είναι στα 12Vdc, ενώ υπάρχουν και εξόδοι των 5Vdc.

Έχει εξόδους για limit switches για laser κλπ., είναι επομένως μια πλήρης συσκευή για χρήση σε μικρής κλίμακας CNC.



CNC 3018 shield



CNC shield V4

Το Arduino Nano του οποίου οι είσοδοι και έξοδοι ειδικά για την χρήση σε CNC router αναγράφονται αναλυτικά στο παρακάτω σχήμα.

Είναι ένα από τα βασικά, λιτά μοντέλα Arduino του οποίου τα I/O επαρκούν για την χρήση την οποία το προορίζουμε.

Οι έξοδοι D2 έως D8, παρέχουν τις εντολές στους drivers A4988 των βηματικών κινητήρων.

Ενώ οι έξοδοι D9 και D12 παρέχουν τις εντολές στα πηνία της πνευματικής βαλβίδας 5/2 και τέλος οι D11 & D10 είναι οι είσοδοι για τον μαγνητικό αισθητήρα του εμβόλου και την έξοδο του κομβίου αντίστοιχα.

Η είσοδος του USB είναι η πόρτα επικοινωνίας μας με τον ηλεκτρονικό υπολογιστή, ο οποίος φέρει το πρόγραμμα οδήγησης του μηχανισμού.

Δεν μπορούμε να παραλείψουμε το μηδαμινό κόστος κτήσης του.

Pinout Ref		Arduino Nano V 3.0 GRBL Pinout		Pinout Ref	
D13	Spindle Direction	D13		D12	Limit Z-Axis
3V3	Not Used	3V3		D11	Variable spindle PWM
VREF	Not Used	VREF		D10	Limit X-Axis
A0	Reset/ Abort	A0		D9	Limit Y-Axis
A1	Feed Hold	A1		D8	Stepper Enable/Disable
A2	Cycle Star/ Resume	A2		D7	Direction Z Axis
A3	Coolant Enable	A3		D6	Direction Y Axis
A4	(Not Used/ Reserve)	A4		D5	Direction X Axis
A5	Probe	A5		D4	Step Pulse Z Axis
A6	Not Used	A6		D3	Step Pulse Y Axis
A7	Not Used	A7		D2	Step Pulse X Axis
		5V		GND	
		RST		RST	
		GND		RX1	
		VIN		TX1	

Arduino Nano V3.0

Ο driver των βηματικών κινητήρων μπορεί να παρέχει έως και 35V έξοδο, ενώ το αντίστοιχο ρεύμα εξόδου με την χρήση ψήκτρας αγγίζει τα 2 Ampere.

Για να λειτουργήσει σε full step mode απαιτεί μόνιμα γειωμένα τα MS1, MS2 & MS3.

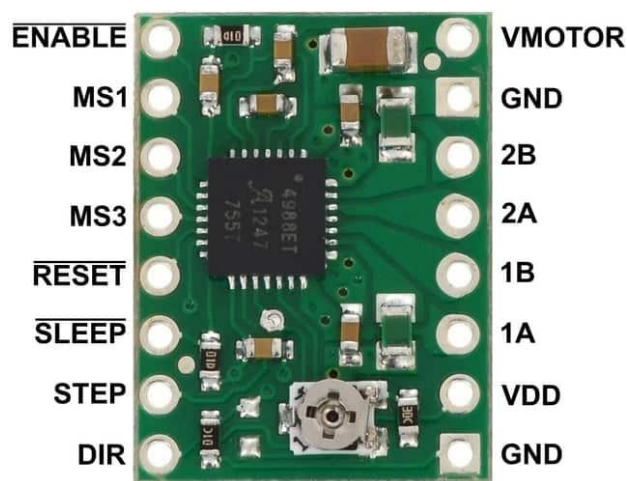
Το ENABLE συνδέεται στην έξοδο D8 του Arduino ενώ τα RESET & SLEEP είναι συνδεδεμένα μεταξύ τους.

Οι εντολές για κίνηση και κατεύθυνση δίνονται στις εισόδους STEP & DIR αντίστοιχα.

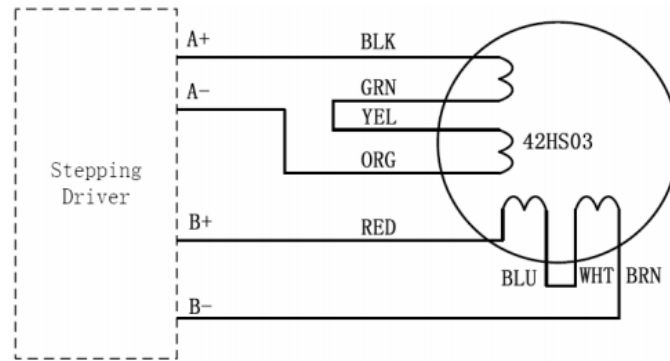
Στο VDD δίνουμε μόνιμα τάση 5V, ενώ στο VMOTOR παρέχουμε την τάση που θέλουμε να τροφοδοτήσουμε τους βηματικούς κινητήρες.

Στην συγκεκριμένη περίπτωση δίνουμε 12V το όριο που μπορεί να δώσει το CNC shield V4 που χρησιμοποιούμε.

Οι έξοδοι 1A, 1B, 2A, 2B συνδέονται στους αντίστοιχους ακροδέκτες του bipolar stepper motor.



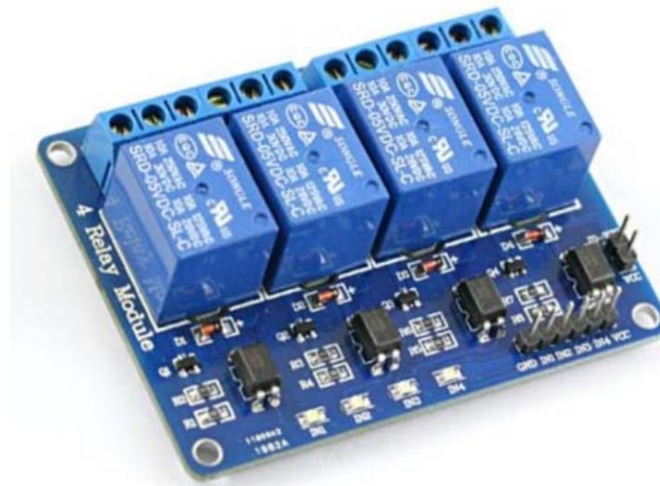
A4988 stepper motor driver



Bipolar stepper motor

Για την οδήγηση από το Arduino, των πηνίων της βαλβίδας 5/2 των οποίων η τάση ενεργοποίησης είναι 24V, απαιτήθηκε η εμπλοκή ρελέ 5V, καθώς οι I/O του δεν μπορούν να διαχειριστούν μεγαλύτερη τάση.

Επιλέχθηκε η παρακάτω πλακέτα η οποία διαθέτει 4 ρελέ αντί για τα δύο που απαιτούνται. Ο λόγος είναι απλός, σε περίπτωση που έχουμε αστοχία κάποιου ρελέ, πολύ γρήγορα να γίνει η αντικατάσταση του με άλλο με την μετακίνηση ενός καλωδίου.



4 channel Relay board

Η καρδιά του μηχανισμού ελέγχου «χτυπά» στην κυριολεξία με την βοήθεια ενός πνευματικού εμβόλου τύπου Mini-cylinder 16 AF 0025 της Metalwork®. Η διαδρομή του είναι

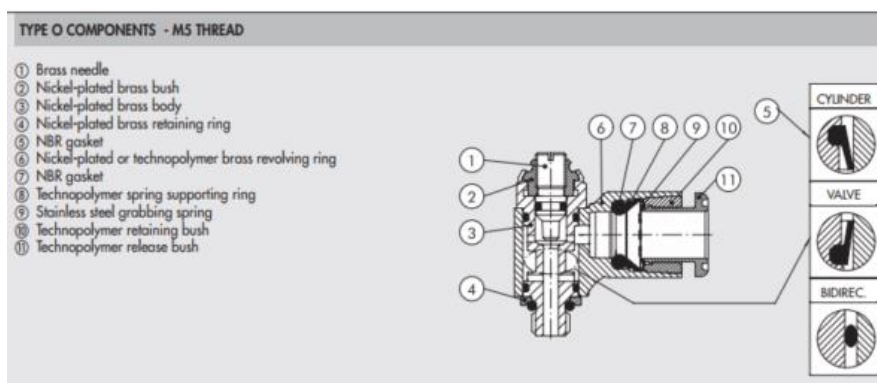
25mm η διάμετρος του είναι 16mm ενώ οι τσιμούχες του είναι τύπου NBR, οι οποίες εξασφαλίζουν πάνω από 1 εκατομμύριο απροβλημάτιστα χτυπήματα.

Η επιλογή του μεγέθους και της διαδρομής του έγινε με γνώμονα την ταχύτητα οπότε επιλέχθηκε μικρή διαδρομή και την μικρή δύναμη που απαιτείται καθώς μεγαλύτερη δύναμη θα επηρέαζε αρνητικά την διάρκεια ζωής του κομβίου. Επίσης χρησιμοποιείται ρυθμιστής πίεσης περεταίρω μείωση της δύναμης ο οποίος θα αναλυθεί αμέσως μετά.



Εικόνα Micro piston 16 AF 0025

Ο ρυθμιστής ροής τύπου MRF compact O, είναι ένας σύγχρονος ρυθμιστής πίεσης όπου η ρύθμιση της ροής γίνεται εύκολα με αν μικρό κατσαβίδι. Η ροή είναι αδιάλειπτη και συνεχής με αυτού του είδους το απαραβίαστο καπάκι.



Εικόνα FLOW MICROREGULATOR SERIES MRF COMPACT O

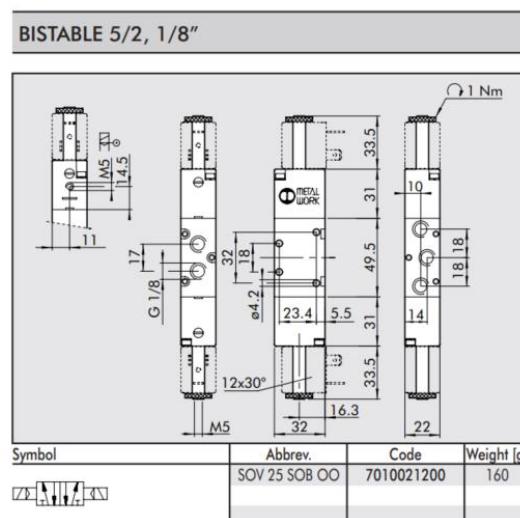
Ένας μαγνητικός διακόπτης τύπου Reed switch με NO επαφή και τάση λειτουργίας 5-240V AC/DC τοποθετήθηκε πάνω στο πνευματικό έμβολο σε σημείο όπου κατά την πλήρη έκταση του ενεργοποιείται. Φέρει πάνω του και ενδεικτικό led σε κόκκινο χρώμα για μια ποιο άμεση οπτική ένδειξη ότι δουλεύει σωστά.

Με αυτό το όργανο μέτρησης έχουμε την δυνατότητα να καταμετρούμε την κάθε κίνηση του εμβόλου, κάτι που θα μας βοηθήσει για να έχουμε μια ποιο πλήρη καταγραφή των κινήσεων του μηχανισμού ελέγχου.



Εικόνα μαγνητικού διακόπτη KT-50R

Η πνευματική βαλβίδα που επιλέχθηκε είναι τύπου 5/2 της Metalwork® με τα αντίστοιχα πηνία τάσης 24V όπου λειτουργεί χωρίς ελατήριο επαναφοράς, για να μπορούμε να ελέγχουμε πλήρως την έκταση και την σύμπτυξη του εμβόλου.



Εικόνα A Pneumatic valve 5/2 SOV 25 SOB 00



Εικόνα B Pneumatic valve 5/2 SOV 25 SOB 00

Το φίλτρο αέρα απομάκρυνσης νερού με ρυθμιστική βαλβίδα και λίπανση που επιλέχθηκε είναι το PDWE 8 B2 της Parkside.

Η συσσώρευση υδρατμών είναι καταστροφική για το έμβολο αλλά και την βαλβίδα και η χρήση μιας νεροπαγίδας είναι απαραίτητη. Εξίσου απαραίτητη είναι και η ρυθμιστική βαλβίδα καθώς η πίεση του του αέρα πρέπει να ελέγχεται και να διατηρείται στα επιθυμητά επίπεδα για την σωστή και πλήρη έκταση του εμβόλου. Η λίπανση δεν απαιτείται για τον συγκεκριμένο τύπο του εμβόλου, αλλά μπορεί να χρησιμοποιηθεί εφόσον χρειαστεί με ένα άλλο τύπο εμβόλου.



Εικόνα PDWE 8 B2

Το μικρορελέ της σειράς 40 από την Finder® τοποθετήθηκε στο κύκλωμα ελέγχου του μαγνητικού διακόπτη και οδηγείτε με 24V για να μπορεί να επικοινωνεί με το Arduino με την χρήση των 5V.

40.52



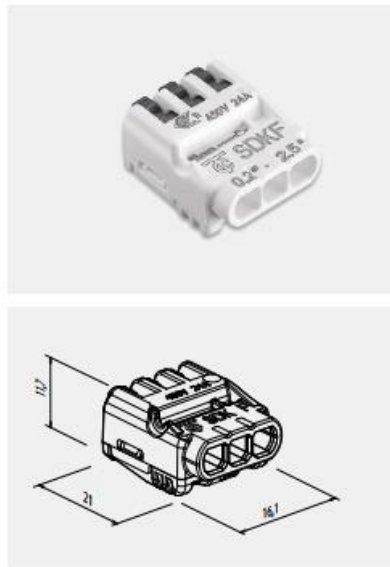
Εικόνα μικρορελέ 40.52.9.024.000

Το μανιτάρι στοπ της GIOVENZANA® μαζί με μια NC επαφή τοποθετήθηκε στην είσοδο του κυκλώματος παροχής ισχύος στα τροφοδοτικά MW, για την άμεση διακοπή ισχύος στο σύστημα, σε άμεσα προσβάσιμο και ορατό σημείο στο πάνω μέρος του μηχανισμού, σε ειδική βάση.



Εικόνα push pull emergency stop

Οι ταχυσύνδεσμοι 3^{ων} θέσεων της SDKF μας δίνουν την δυνατότητα ασφαλούς και μόνιμης σύνδεσης μεταξύ των αγωγών, χωρίς διακοπές.



Εικόνα SDKF_Electroterminal

Ο αεροσυμπιεστής που χρησιμοποιήθηκε για την αδιάλειπτη παροχή αέρα στον μηχανισμό είναι της οικονομικής σειράς από την Einhell με δυνατότητα παροχής 150lt/min στα 4 bar που απαιτούνται για την ορθή λειτουργία του.

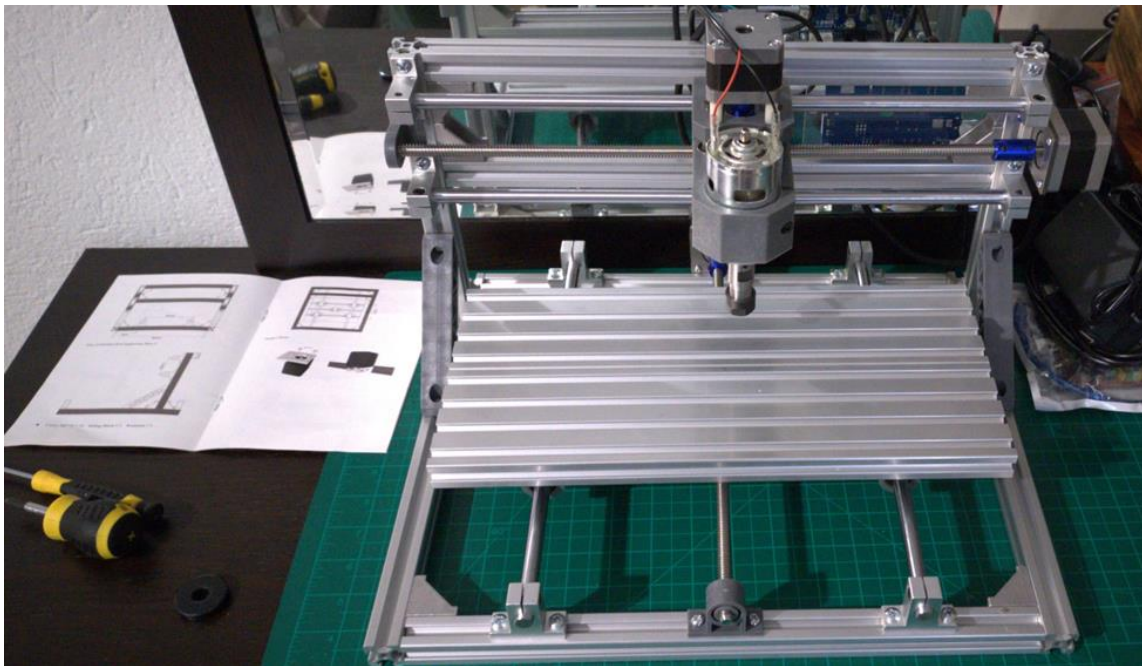


Εικόνα Einhell BT-AC 270/50 2.5hp/50lt

Περιγραφή διαδικασίας κατασκευής :

Η διαδικασία κατασκευής του όλου εγχειρήματος περιλάμβανε πέντε στάδια, ενώ η συνολική διάρκεια ανέρχεται στους 4 μήνες.

1. Στο πρώτο στάδιο κατασκευάστηκε και λειτούργησε το CNC 3018, για να κατανοηθεί ο τρόπος λειτουργίας του και να εκτιμηθούν οι δυνατότητες του. Αλλάχθηκε η main board του CNC με το CNC shield V4 και προγραμματίστηκε για να βρεθούν τα όρια του συστήματος.



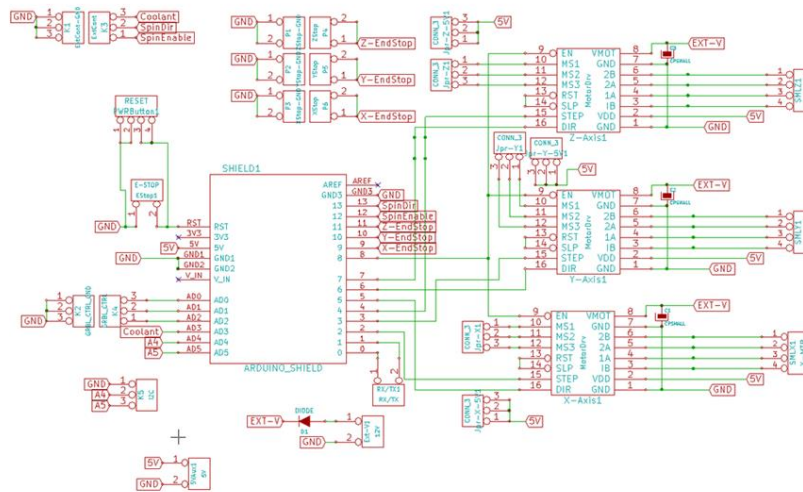
Εικόνα συναρμολόγηση του CNC 3018

2. Στο δεύτερο στάδιο επιλέχθηκαν τα εξαρτήματα που απαρτίζουν το μέρος του μηχανισμού που κινείται με τον αέρα, δηλαδή το έμβολο, η βαλβίδα 5/2 τα πηνία, οι σύνδεσμοί ρυθμιστές πίεσης, ρυθμιστές ροής κλπ.

3. Στο τρίτο στάδιο και το ποιο απαιτητικό σε συντονισμό κινήσεων αλλά και χρονοβόρο, καθώς δεν υπάρχει άλλη τέτοιου τύπου κατασκευή.

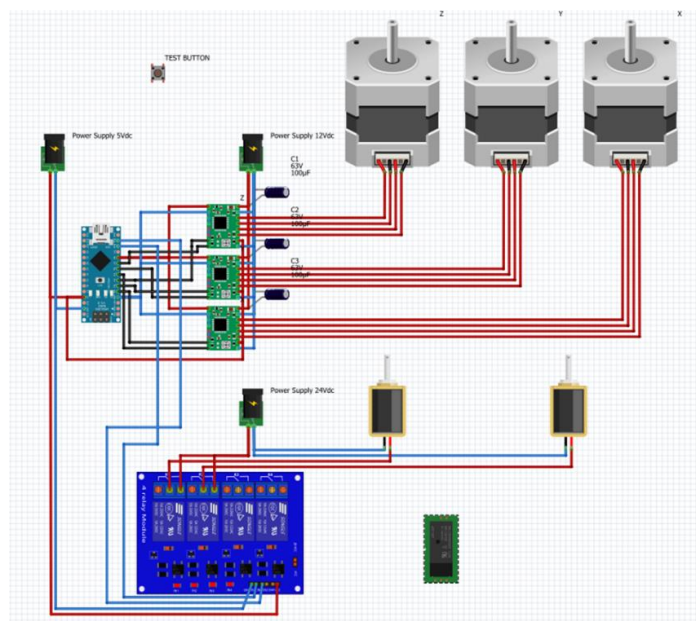
Μελετήθηκαν τα κυκλωματικά διαγράμματα του CNC shield V4, επιλέχθηκε ο τρόπος λειτουργίας του και δημιουργήθηκε το ηλεκτρολογικό σχέδιο.

Επιλέχθηκαν τα ρελέ και κατασκευάστηκαν όλα τα επιμέρους εξαρτήματα που δεν υπήρχαν διαθέσιμα στο εμπόριο και θα αναλυθούν αργότερα, με την χρήση 3D εκτυπωτή.



Εικόνα κυκλωματικό διάγραμμα του CNC shield V4

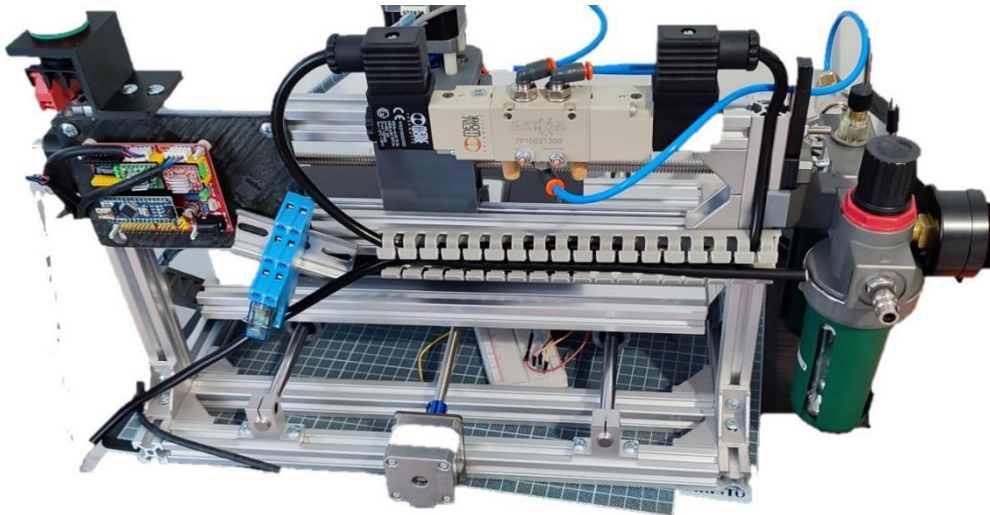
4. Στο τέταρτο στάδιο δημιουργήθηκαν τα επιμέρους κομμάτια του κώδικα για το Arduino αλλά και στην C# για το έλεγχο του Arduino και παράλληλα ξεκίνησαν οι δοκιμές απόδοσης του πνευματικού εμβόλου και του συντονισμού των κινήσεων των βηματικών κινητήρων. Τέλος έγινε η ηλεκτρική σύνδεση όλων των εξαρτημάτων.



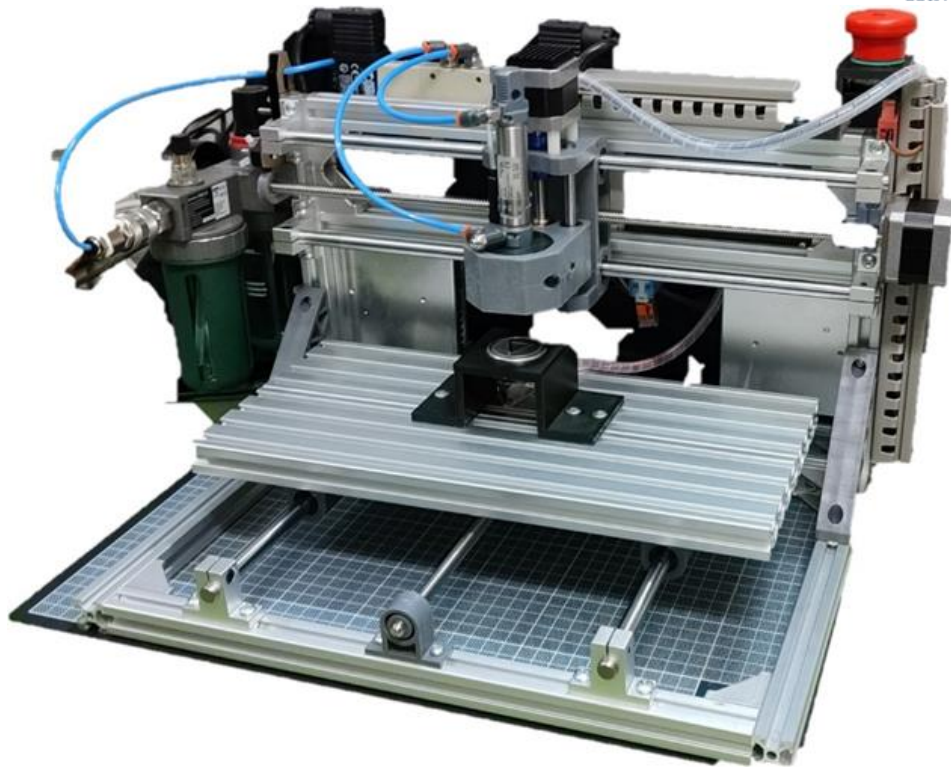
Εικόνα σχεδιασμός και οπτικοποίηση στο Fritzing®

5. Ενώ στο πέμπτο στάδιο πραγματοποιήθηκαν οι δοκιμές, οι οποίες περιλάμβαναν :

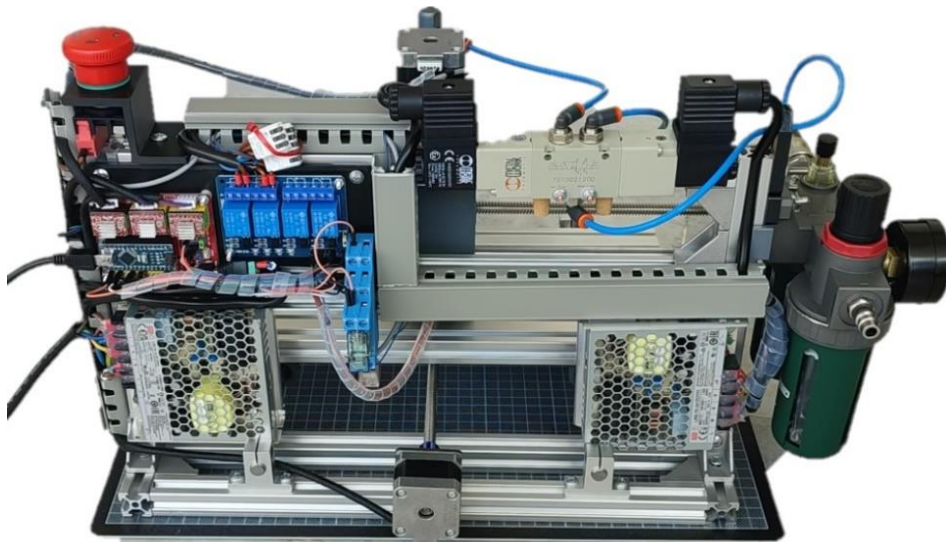
- Τις ρυθμίσεις της ταχύτητας λειτουργίας του εμβόλου
- την επιλογή του ρυθμού χτυπήματος
- αλλαγές μικρούλικών
- τον επανασχεδιασμό των βάσεων για τα κομβία
- την ολοκλήρωση του προγραμματισμού σε Arduino, C# & σύνδεση με τον SQL server



Εικόνα πίσω όψη του μηχανισμού κατά την διάρκεια της κατασκευής



Εικόνα πλάγια όψη του μηχανισμού



Εικόνα πίσω όψη του μηχανισμού

Περιγραφή διαδικασίας κατασκευής 3D εκτυπωμένων υλικών :

Για να σχεδιαστούν όλα τα “tailor made” απάρτια του μηχανισμού επιλέχθηκε η online πλατφόρμα Tinkercad της Autodesk®.

Η πλατφόρμα δίνει την δυνατότητα να κατασκευάζεις πολύ γρήγορα και εύκολα ότι μπορείς να φανταστείς και διαθέτει πληθώρα έτοιμων λύσεων αλλά και βάση δεδομένων όπου οι χρήστες της μπορούν να «ανεβάσουν» τις δημιουργίες τους.

Ενώ παράλληλα μπορεί να τα εξάγει σε διάφορες μορφές αρχείων για 3D printing ή Laser cutting. Επομένως εξήχθησαν σε μορφή stl προς εκτύπωση.

Τα απάρτια εκτυπώθηκαν σε ένα «σπιτικό» 3D εκτυπωτή της Creality®, τον Ender 3 V2.

Ο συγκεκριμένος 3D εκτυπωτής είναι ένας από τους πιο αξιόπιστους αλλά και διαδεδομένους καθώς το κόστος αγοράς του είναι χαμηλό.



Εικόνα Creality Ender 3 V2

Ο εκτυπωτής συνήθως εκτυπώνει με το κλασικό υλικό νήματος το PLA (Polylactic Acid) όπως όλοι οι φθηνοί εκτυπωτές. Όμως τα εξαρτήματα που έπρεπε να εκτυπωθούν θέλαμε ταυτόχρονα να έχουν αντοχή για να ανταπεξέλθουν κάτω από δύσκολες συνθήκες χρήσης τους σε βιομηχανικό περιβάλλον. Επιλέχθηκε να αναβαθμιστεί με το σύστημα της Creality® CR touch έτσι ώστε να μπορεί να κάνει αυτόματο μηδενισμό του άξονα Z.

Αυτό είναι απαραίτητο καθώς κατασκευάστηκε ένα κάλυμμα από ξύλο για να εγκιβωτιστεί ο εκτυπωτής. Επίσης αλλάχθηκε ο extruder & το PTFE tube για να υποστηρίξει μεγαλύτερες θερμοκρασίες λειτουργίας.

Όλα τα παραπάνω χρειάζονται για να μπορεί πλέον αυτός ο απλός 3D εκτυπωτής να εκτυπώνει με νήμα από ABS (Acrylonitrile Butadiene Styrene) !

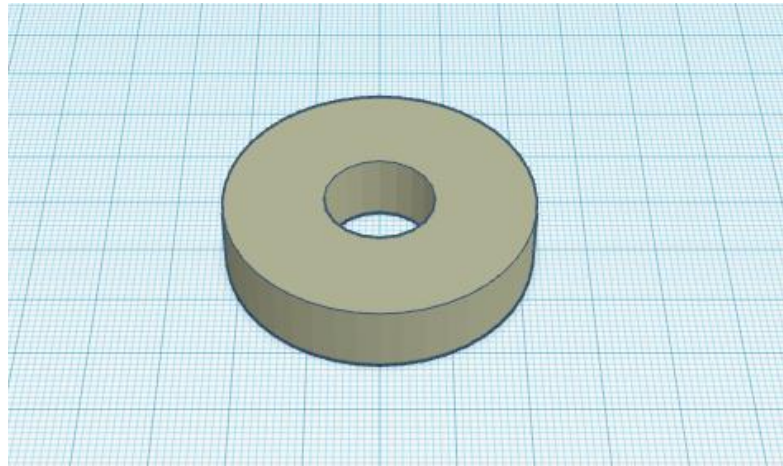
Το ακρυλονιτρίλιο έχει υψηλή αντοχή, θερμική σταθερότητα αλλά και χημική σταθερότητα, ενώ το βουταδιένιο έχει σκληρότητα και αντοχή στην κρούση. Τέλος το στυρένιο έχει εύκολη επεξεργασία, υψηλή στιλπνότητα και υψηλή αντοχή.



Εικόνα του «καλύμματος» για τον Creality Ender 3 V2

Το πρώτο κομμάτι που κατασκευάστηκε ήταν ο δακτύλιος συγκράτησης του πνευματικού εμβόλου στην βάση στήριξης για το spindle motor του CNC.

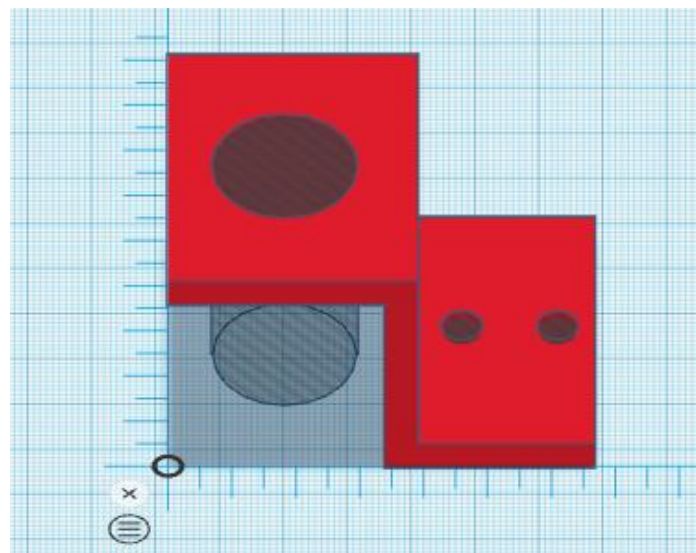
Η διαφορά με το ίδιο εξάρτημα κατασκευασμένο από PLA γίνεται αμέσως εμφανής, καθώς το εξάρτημα από PLA ράγισε μόλις στα 2' μετά την εφαρμογή του.



Εικόνα δακτύλιος συγκράτησης του πνευματικού εμβόλου

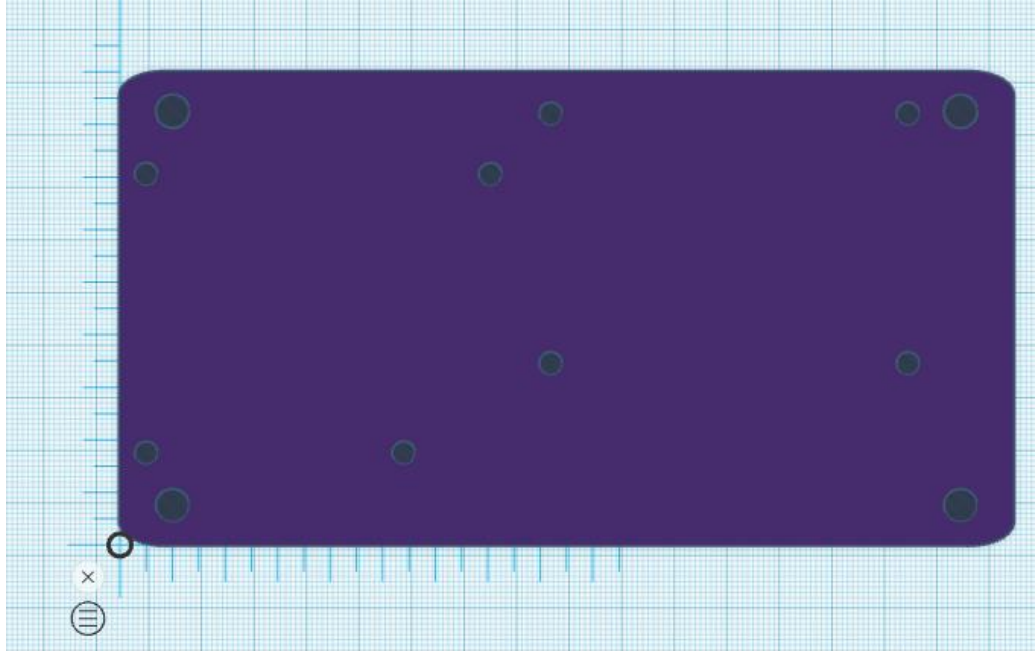
Το δεύτερο εξάρτημα που παράχθηκε ήταν η βάση στήριξης του emergency stop.

Επίσης σε σχέση με το ίδιο εξάρτημα αλλά κατασκευασμένο από PLA, διαπιστώθηκε ότι πλέον μπορείς να σηκώσεις όλο το βάρος της κατασκευής από την βάση χωρίς να σπάσει !

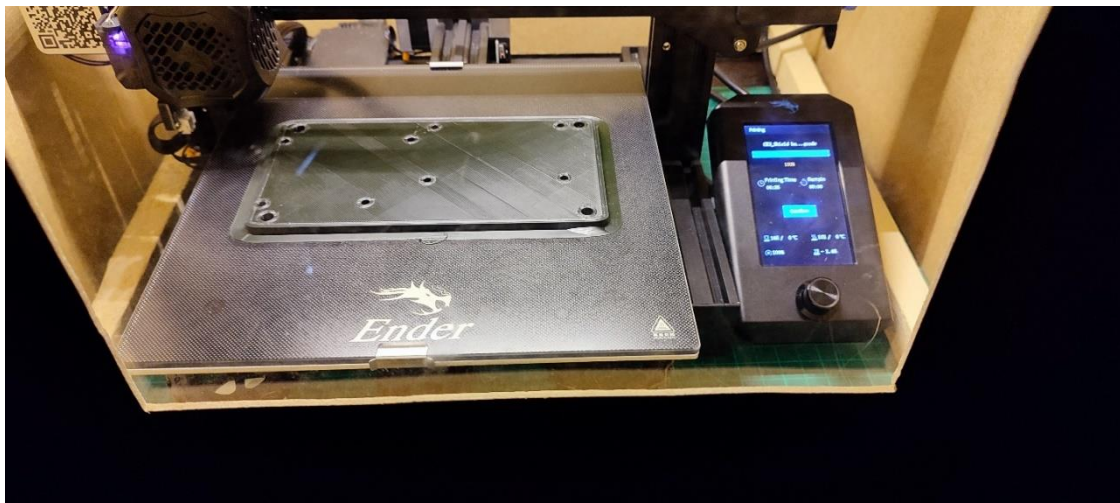


Εικόνα βάση στήριξης του emergency stop

Το 3^ο εξάρτημα σχεδιάστηκε και εκτυπώθηκε 2 φορές καθώς η διάταξη των υλικών απαιτούσε δοκιμή και κάποιες αλλαγές. Συγκρατεί τις πλακέτες του shield, των ρελέ για τα πηνία της βαλβίδας και του ρελέ για τον μαγνητικό αισθητήρα.

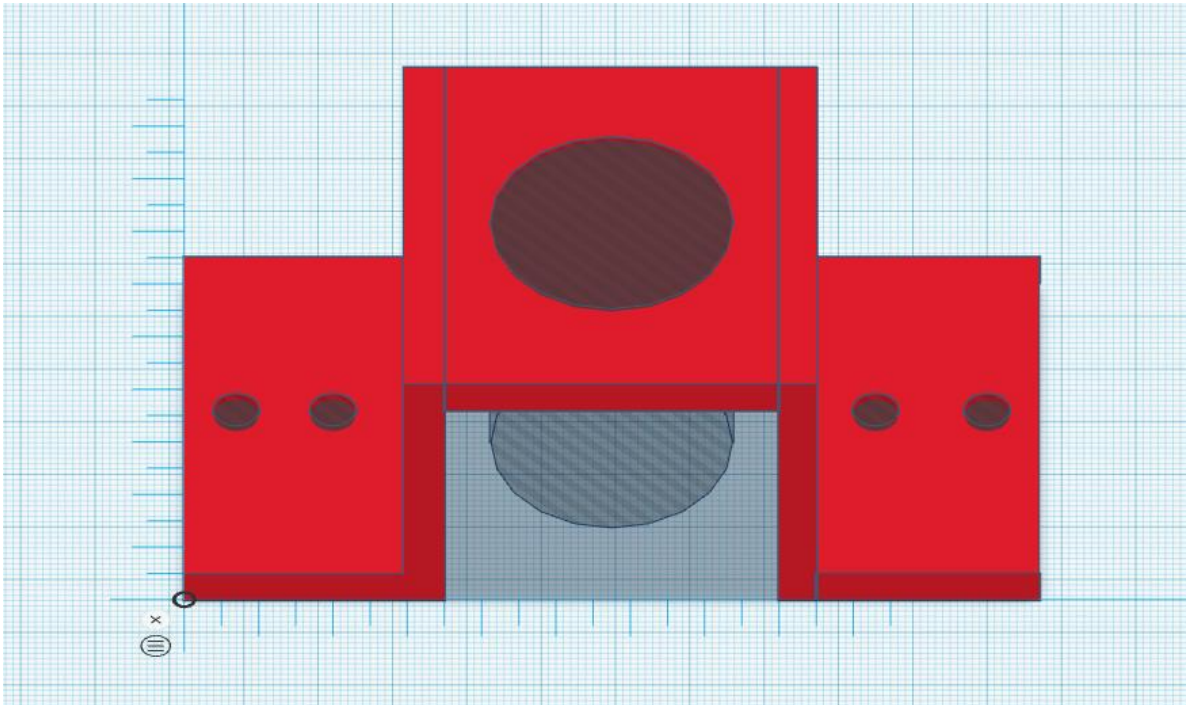


Εικόνα βάση στήριξης πλακετών

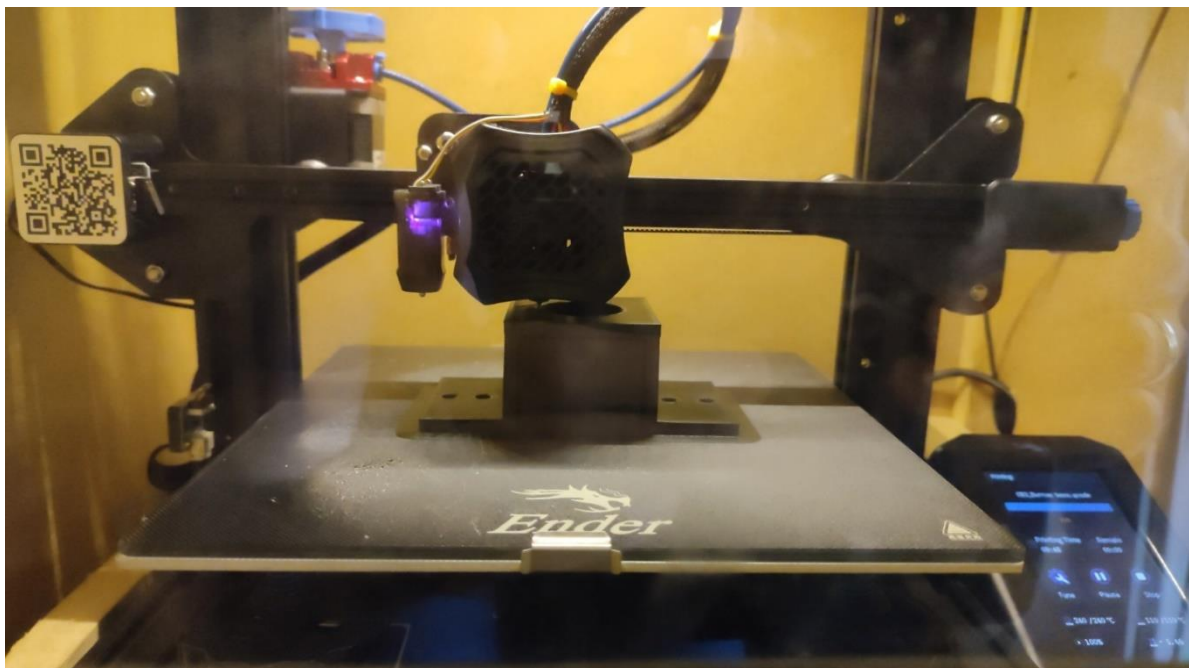


Εικόνα 3D εκτύπωση της βάσης στήριξης πλακετών

Το 4^ο εξάρτημα σχεδιάστηκε και εκτυπώθηκε για να συγκρατεί τα στρόγγυλα κομβία της δοκιμής. Η επιλογή του υλικού αποδείχθηκε εξαιρετική καθώς άντεξε και αντέχει όλα τα χτυπήματα των δοκιμών !

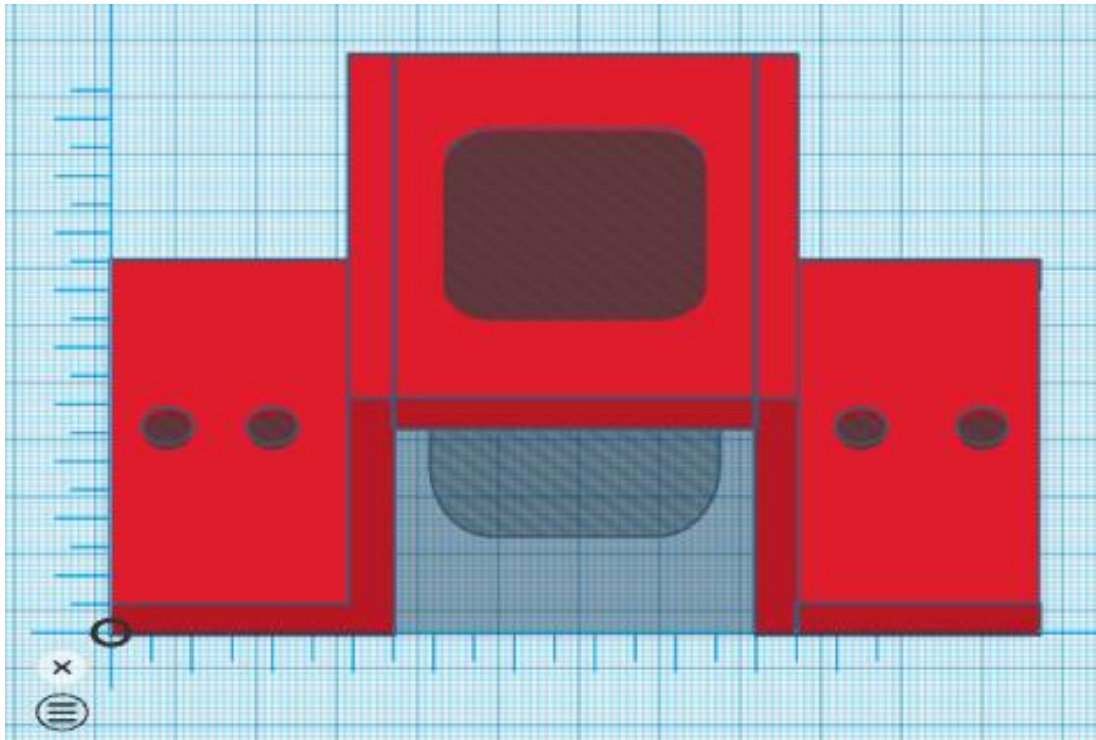


Εικόνα σχέδιο της βάσης στήριξης στρόγγυλου κομβίου

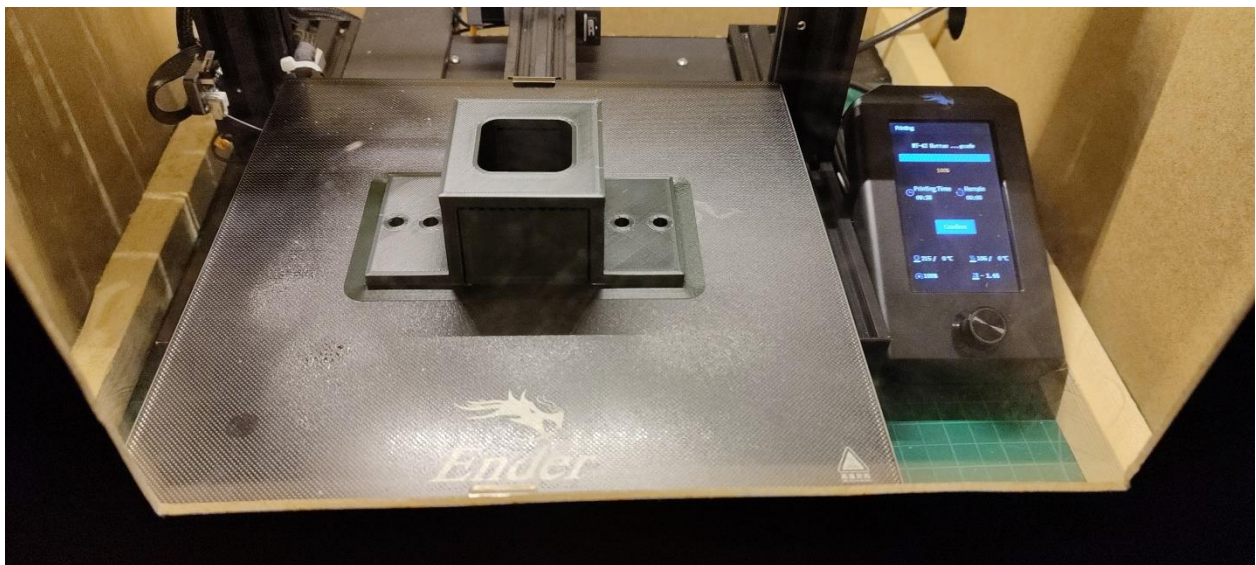


Εικόνα 3D εκτύπωση της βάσης στήριξης στρόγγυλου κομβίου

Το 5^ο εξάρτημα σχεδιάστηκε και εκτυπώθηκε για να συγκρατεί τα τετράγωνα κομβία της δοκιμής.



Εικόνα σχέδιο της βάσης στήριξης τετράγωνου κομβίου



Εικόνα 3D εκτύπωση της βάσης στήριξης τετράγωνου κομβίου

Κεφάλαιο 4^ο :

Περιγραφή προγράμματος για την κατανόηση του αυτοματισμού :

Δημιουργήθηκε με γνώμονα την ευκολία στον χειρισμό από τον μέσο χειριστή CNC, έτσι ώστε τα αποτελέσματα των δοκιμών να είναι άμεσα διαθέσιμα προς μελέτη και εξαγωγή συμπερασμάτων ανά πάσα στιγμή, χωρίς χρονοβόρες διαδικασίες.

Επιλέχθηκε επίσης να μην γίνει «αυτόνομο» δηλαδή ο έλεγχος να είναι μόνο από το Arduino αλλά να συνδυαστεί με επιπλέον πρόγραμμα γραμμένο σε C# για χρήση σε σύνδεση με H/Y και περιβάλλον Windows, έτσι ώστε να υπάρχει καλύτερη οπτικοποίηση, με βέλτιστο και άμεσο χειρισμό όλων των παραμέτρων.

Το πρόγραμμα στην ουσία ελέγχει την κίνηση των 3^{ων} βηματικών κινητήρων του CNC αλλά και την έκταση ή την σύμπτυξη του πνευματικού εμβόλου. Ο συνδυασμός όμως των κινήσεων μαζί με την πλήρης παραμετροποίηση τους είναι που συνθέτουν την μοναδικότητα του συγκεκριμένου προγράμματος.

Αναλυτικότερα, έχουμε την δυνατότητα να επιλέξουμε μεταξύ χειροκίνητου χειρισμού[2] και αυτόματου χειρισμού[5], σε διακριτές για τον χειριστή περιοχές.

Επιπλέον υπάρχει αλληλομανδάλωση των επιλογών ώστε να μην υπάρχει περίπτωση να συνδυαστούν εκούσιες ή ακούσιες επιλογές από την πλευρά του χειριστή του μηχανισμού.

Στην επιλογή του χειροκίνητου, ενεργοποιούμε μέσω κομβίων συνεχούς πίεσης «κλικ», τον καθένα ξεχωριστά από τους τρεις βηματικούς κινητήρες προς όλες τις κατευθύνσεις τους είτε με γρήγορη και συνεχή κίνηση [3], είτε με βήματα των οποίων την διάρκεια μπορούμε να επιλέξουμε [4].

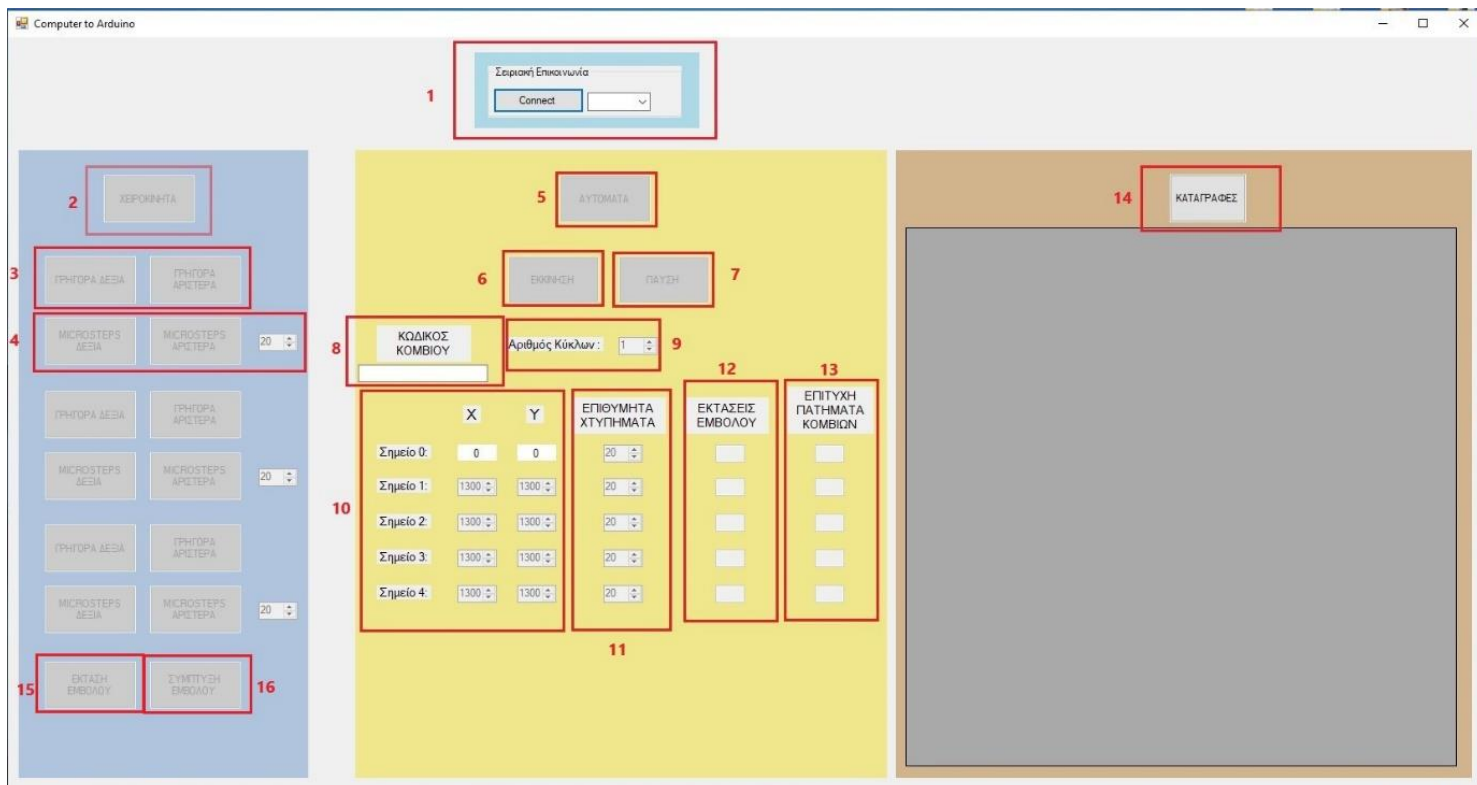
Επίσης με αντίστοιχα κομβία ενεργοποιούμε την μόνιμη έκταση [15] ή σύμπτυξη [16] του εμβόλου. Όλα τα παραπάνω μας βοηθούν στο να βαθμονομήσουμε σωστά την συσκευή έτσι ώστε να ξεκινήσουμε τον αυτοματισμό για τις συνεχείς δοκιμές.

Στην επιλογή του αυτόματου [5], έχουμε την δυνατότητα εκκίνησης [6] ή παύσης [7] της δοκιμής όπως επίσης και την επιλογή του αριθμού των κύκλων δοκιμών [9] που θα πραγματοποιήσουμε. Επιπλέον υπάρχει η δυνατότητα να επιλέξουμε σε ποια απόσταση από το κέντρο θα πραγματοποιηθούν τα υπόλοιπα από τα 4 σημεία χτυπήματος [10]. Επίσης μπορούμε να εισάγουμε την περιγραφή του κομβίου για να ξεχωρίζουμε τις καταγραφές [8].

Καθορίζουμε με ξεχωριστές επιλογές τα επιθυμητά χτυπήματα ανά θέση και φυσικά υπάρχει Live οπτικοποίηση των εκτάσεων του εμβόλου [12] και των επιτυχών πατημάτων του κομβίου [13].

Μετά από κάθε ολοκληρωμένη σειρά χτυπημάτων σε μια θέση, το πλήθος των εκτάσεων του εμβόλου και το πλήθος των επιτυχών πατημάτων του κομβίου, συλλέγονται και μεταφέρονται την βάση της SQL, όπου πατώντας το κομβίο καταγραφές μπορούμε να τις δούμε και στο πρόγραμμα [14].

Με αυτό τον τρόπο κατακερματίζουμε την δοκιμή σε μικρότερα κομμάτια και εξασφαλίζουμε ότι δεν θα χαθούν πολύτιμες πληροφορίες αναγκαίες για την σωστή εξαγωγή των αποτελεσμάτων.



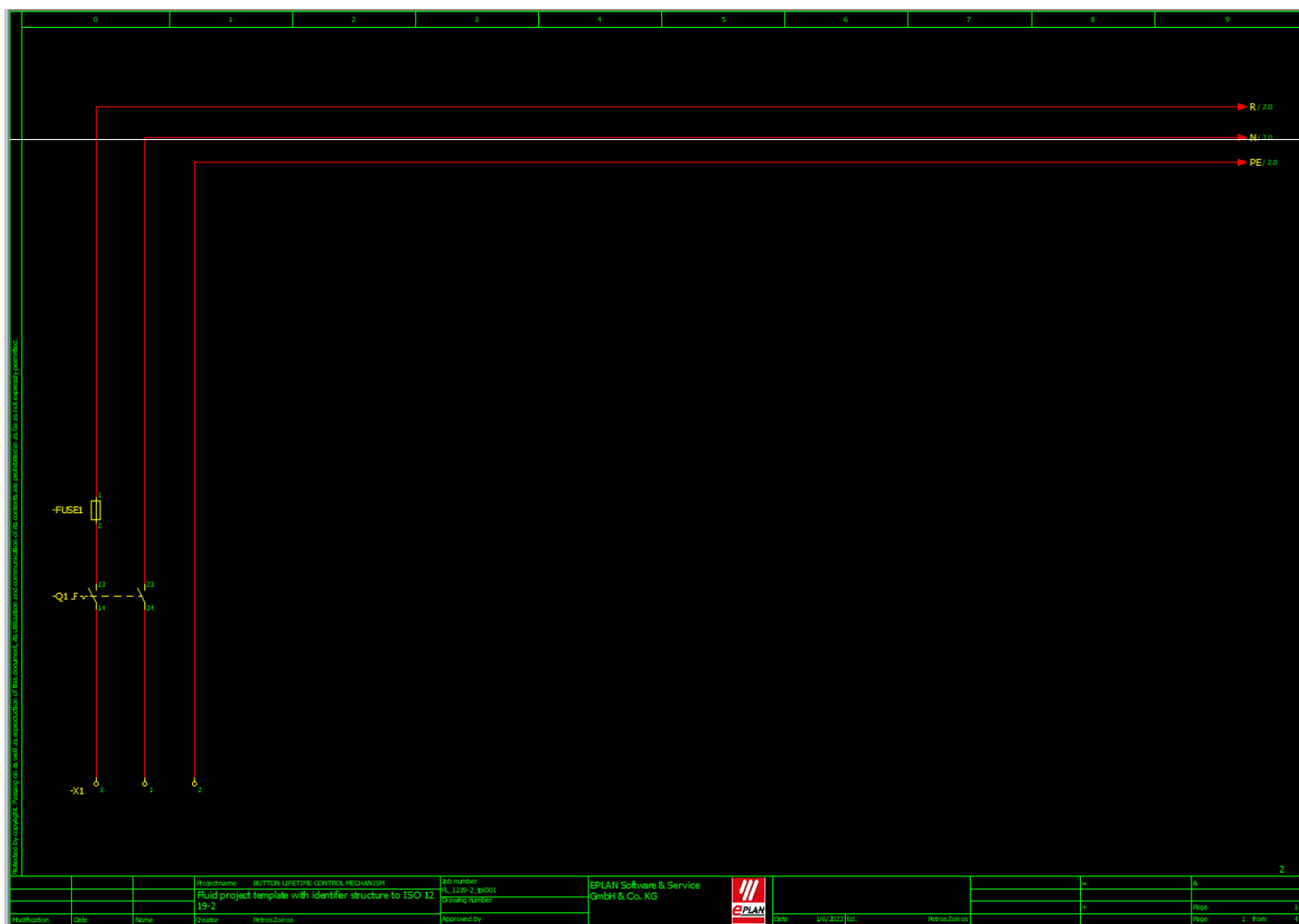
Εικόνα προγράμματος

Ηλεκτρολογικό σχέδιο συνδεσμολογίας σε EPLAN :

Απαραίτητο στοιχείο για την οποιαδήποτε επισκευή αλλά και για την όποια μελλοντική αναβάθμιση αποτελεί η αποτύπωση των συνδέσεων των εξαρτημάτων σε ένα πλήρες ηλεκτρολογικό σχέδιο του μηχανισμού.

Καθώς και ο μηχανισμός είναι μέρος της παρούσας διπλωματική εργασίας, είναι δεδομένο ότι απαιτείται η ορθή αποτύπωση όλων των στοιχείων της με σύγχρονα μέσα.

Επομένως για το ηλεκτρολογικό σχέδιο χρησιμοποιήθηκε το πλέον σύγχρονο εργαλείο των ηλεκτρολόγων μηχανικών το EPLAN®, στην μορφή Education 2.9 SP1.



Εικόνα ηλεκτρολογικό σχέδιο σε Eplan σελίδα 1^η

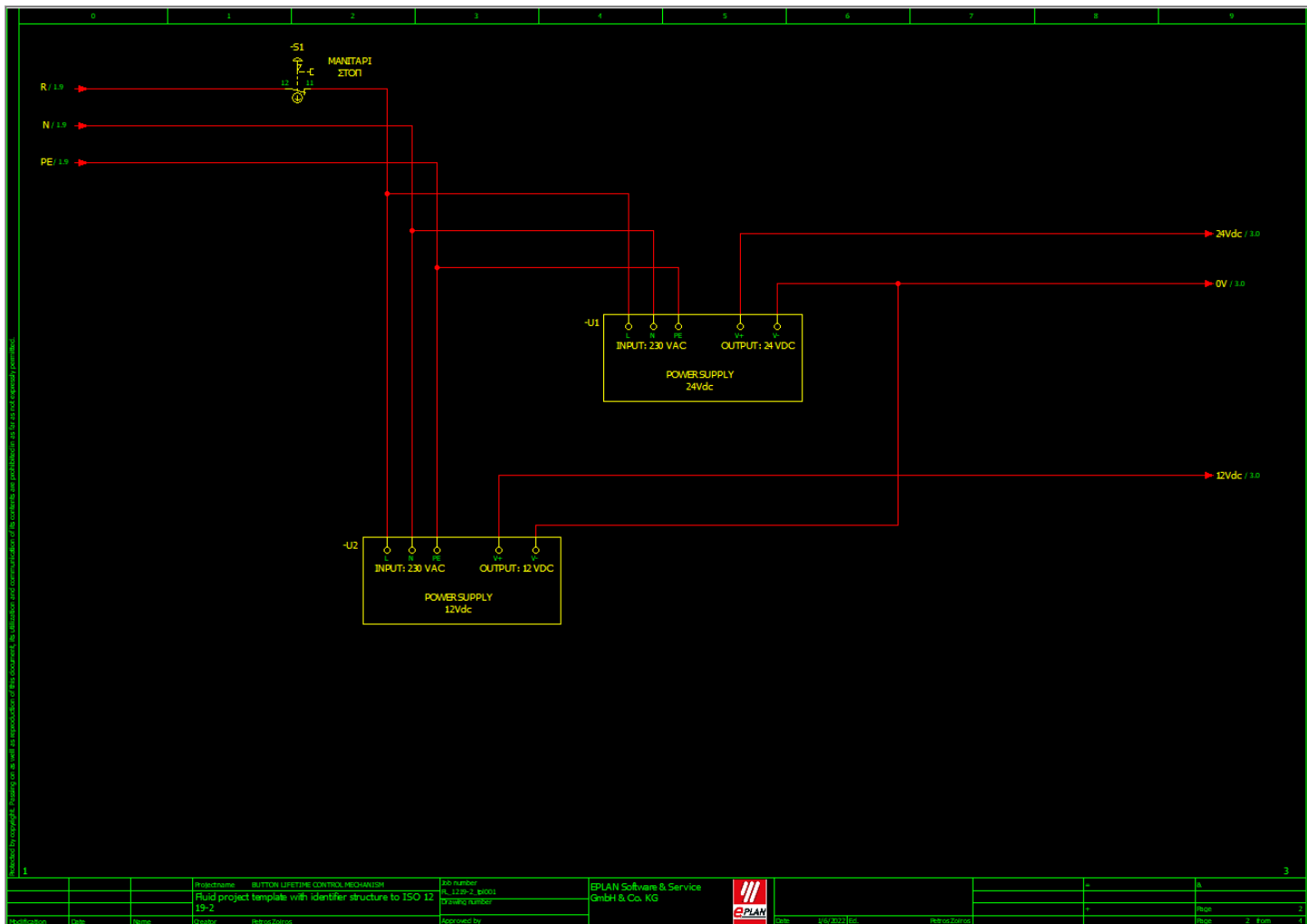
Η αποτύπωση περιλαμβάνει 4 σελίδες όπου περιέχουν:

Στην 1^η σελίδα την αποτύπωση :

- του μονοφασικού διακόπτη
- της ασφάλειας

Στην 2^η σελίδα την αποτύπωση :

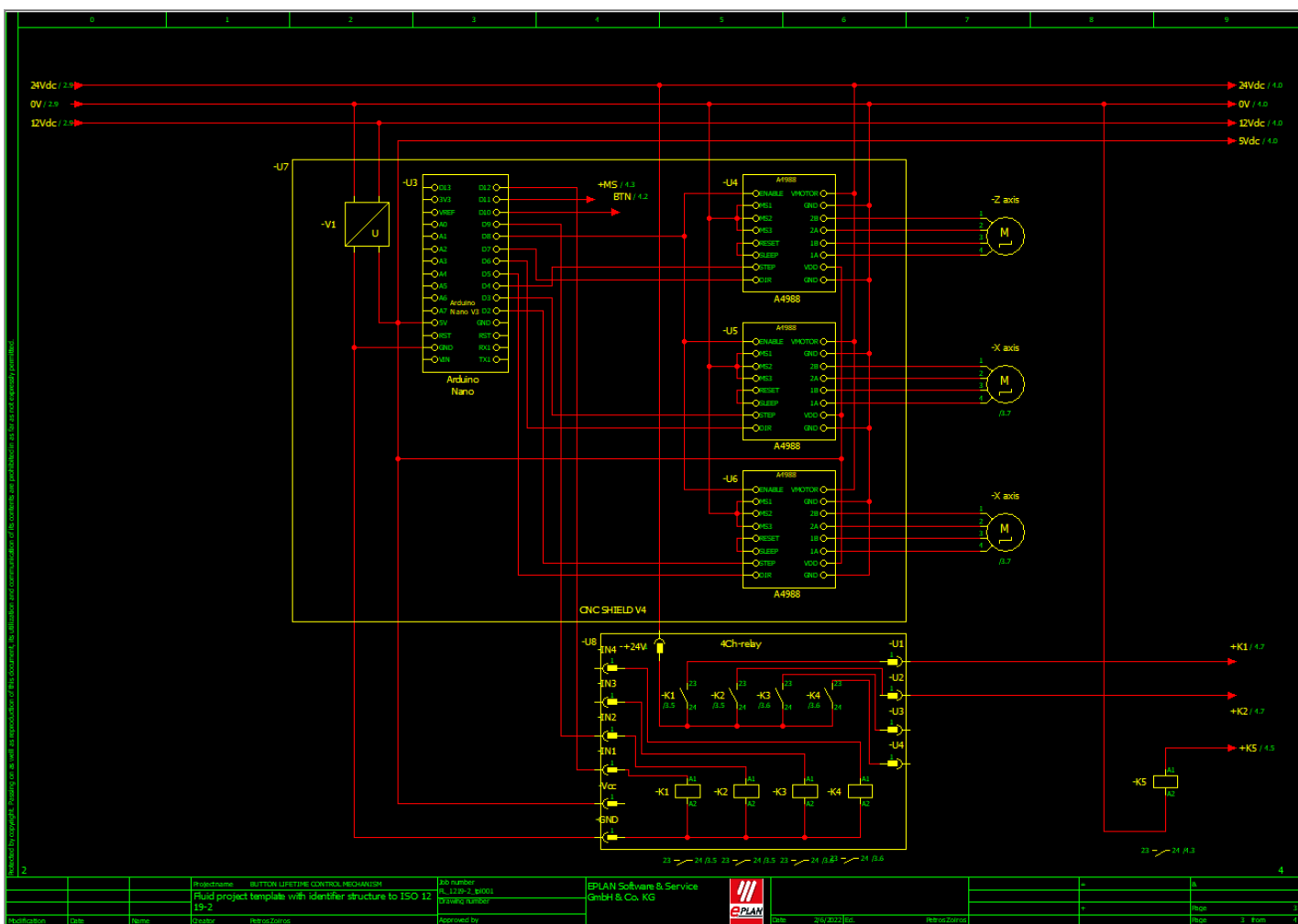
- του διακόπτη έκτακτης ανάγκης (μανιτάρι στοπ)
- του τροφοδοτικού 12Vdc
- του τροφοδοτικού 24Vdc.



Εικόνα ηλεκτρολογικό σχέδιο σε Eplan σελίδα 2^η

Στην 3^η σελίδα την αποτύπωση :

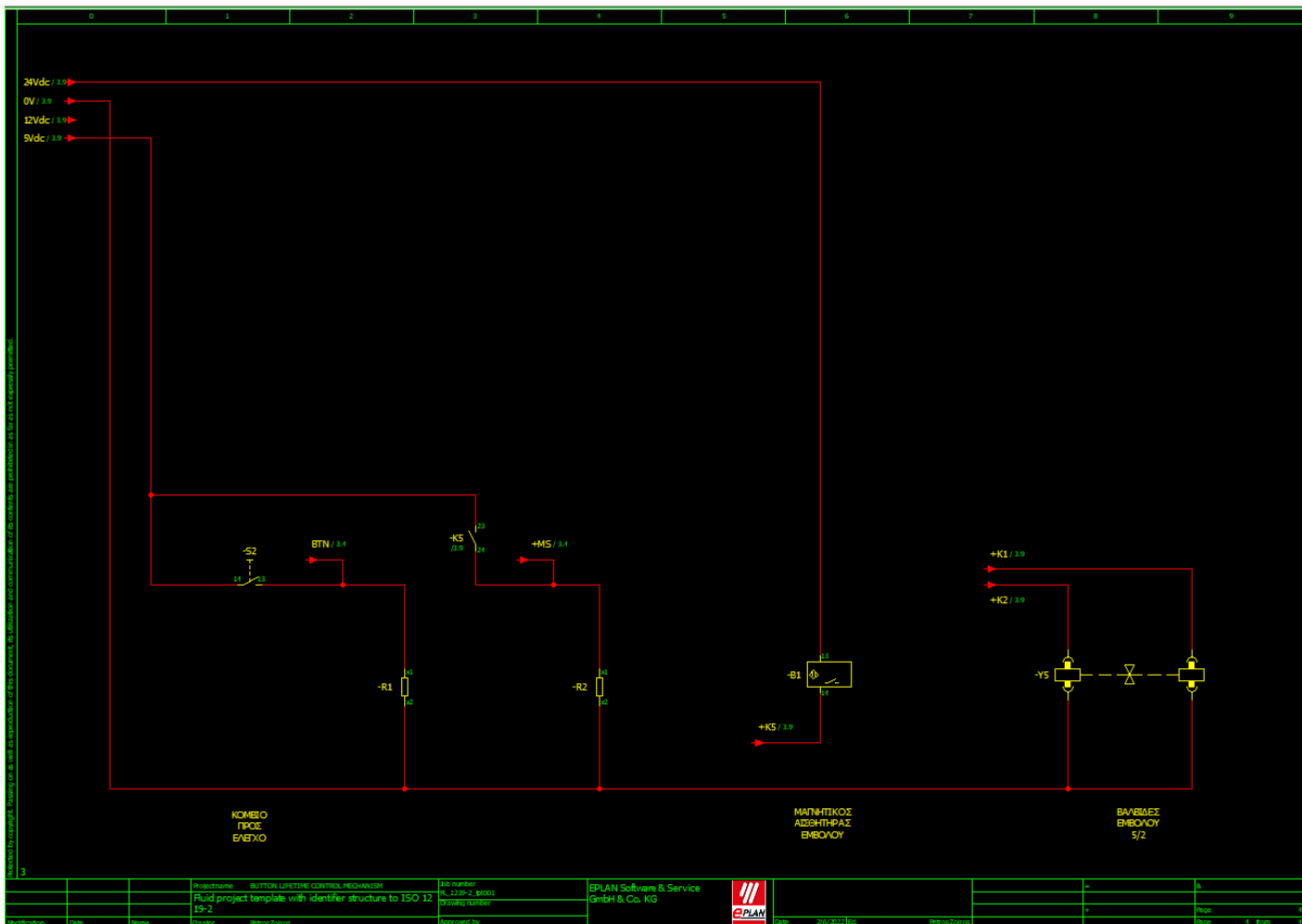
- του CNC shield που περιλαμβάνει το Arduino Nano
- τους drivers A4988 για τους βηματικούς κινητήρες
- τους 3 βηματικούς κινητήρες
- το κύκλωμα της πλακέτας των 4^{ων} ρελέ
- το ρελέ K5 που ενεργοποιείται μέσω του μαγνητικό αισθητήρα που είναι προσαρμοσμένος στο πνευματικό έμβολο.



Εικόνα ηλεκτρολογικό σχέδιο σε Eplan σελίδα 3^η

Τέλος στην 4^η σελίδα την αποτύπωση :

- του κομβίου προς έλεγχο και του κυκλώματος οδήγησης του
- του μαγνητικού αισθητήρα
- του αντίστοιχου κυκλώματος οδήγησης το ρελέ Κ5
- την αποτύπωση των πηνίων της βαλβίδας 5/2



Εικόνα ηλεκτρολογικό σχέδιο σε Eplan σελίδα 4^η

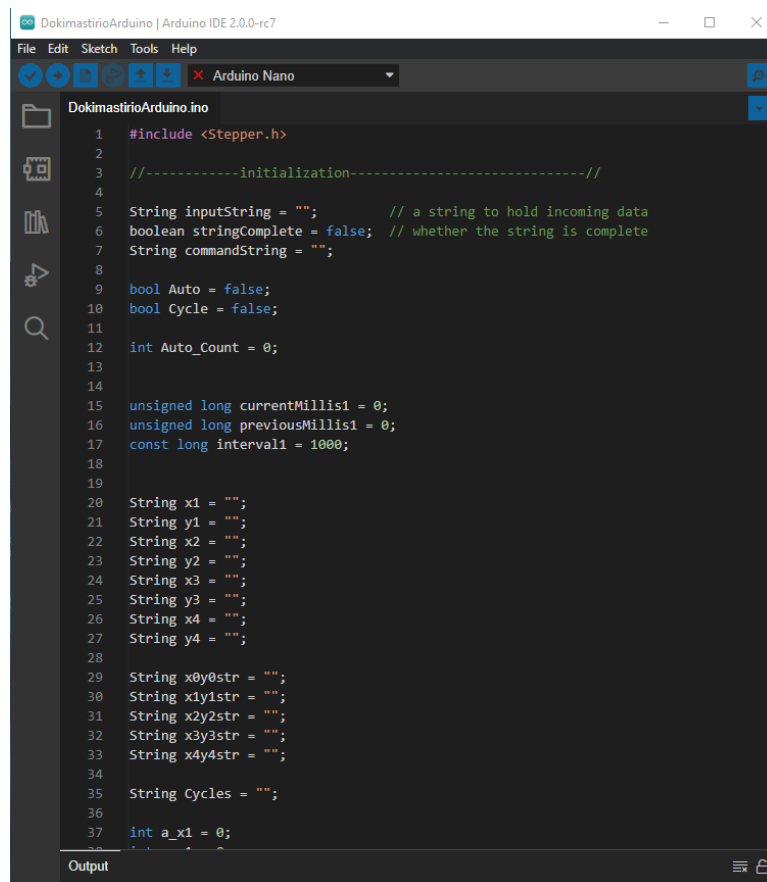
Κεφάλαιο 5^ο :

Ανάλυση προγράμματος C :

Η καρδιά του μηχανισμού είναι ένα Arduino Nano, το οποίο προγραμματίζεται με το Arduino IDE, ένα πρόγραμμα που μπορείς να γράφεις τις εντολές σε γλώσσα C. Το οποίο στην έκδοση 2.0.0-rc7 που χρησιμοποιήθηκε έχει πλέον πολλές δυνατότητες όπως debug κλπ.

Στην συγκεκριμένη εφαρμογή, για να μπορέσει να διαχειριστεί τους βηματικούς κινητήρες θα πρέπει να εισάγουμε την βιβλιοθήκη Stepper.h όπου κατά αυτό τον τρόπο θα μας είναι πιο εύκολο να τους προγραμματίσουμε χωρίς να γράψουμε περίπλοκο κώδικα.

Ακολουθεί η ανάλυση των επιμέρους στοιχείων του κώδικα, καθώς έχει πολλά κομμάτια που συνδέουν το πρόγραμμα του Arduino με το πρόγραμμα από την C#.



```
DokimastirioArduino.ino
1  #include <Stepper.h>
2
3  //-----initialization-----//
4
5  String inputString = ""; // a string to hold incoming data
6  boolean stringComplete = false; // whether the string is complete
7  String commandString = "";
8
9  bool Auto = false;
10 bool Cycle = false;
11
12 int Auto_Count = 0;
13
14
15 unsigned long currentMillis1 = 0;
16 unsigned long previousMillis1 = 0;
17 const long interval1 = 1000;
18
19
20 String x1 = "";
21 String y1 = "";
22 String x2 = "";
23 String y2 = "";
24 String x3 = "";
25 String y3 = "";
26 String x4 = "";
27 String y4 = "";
28
29 String x0y0str = "";
30 String x1y1str = "";
31 String x2y2str = "";
32 String x3y3str = "";
33 String x4y4str = "";
34
35 String Cycles = "";
36
37 int a_x1 = 0;
```

Εικόνα πρόγραμμα Arduino IDE 2.0.0 rc7

Ανάλυση κώδικα C :

```
#include <Stepper.h>
```

Εισαγωγή της βιβλιοθήκης stepper

```
//-----initialization-----//  
  
String inputString = "";           // a string to hold incoming data  
boolean stringComplete = false;    // whether the string is complete  
String commandString = "";  
  
bool Auto = false;  
bool Cycle = false;  
  
int Auto_Count = 0;  
  
unsigned long currentMillis1 = 0;  
unsigned long previousMillis1 = 0;  
const long interval1 = 1000;  
  
String x1 = "";  
String y1 = "";  
String x2 = "";  
String y2 = "";  
String x3 = "";  
String y3 = "";  
String x4 = "";  
String y4 = "";  
  
String x0y0str = "";  
String x1y1str = "";  
String x2y2str = "";  
String x3y3str = "";  
String x4y4str = "";  
  
String Cycles = "";  
  
int a_x1 = 0;  
int a_y1 = 0;  
int a_x2 = 0;  
int a_y2 = 0;  
int a_x3 = 0;  
int a_y3 = 0;  
int a_x4 = 0;  
int a_y4 = 0;
```

```
int a_x0y0str = 0;
int a_x1y1str = 0;
int a_x2y2str = 0;
int a_x3y3str = 0;
int a_x4y4str = 0;

int a_x0y0curstr = 0;
int a_x1y1curstr = 0;
int a_x2y2curstr = 0;
int a_x3y3curstr = 0;
int a_x4y4curstr = 0;

int a_Cycles = 0;

int a_CyclesCur = 0;

int x0y0sensor = 0;
int x1y1sensor = 0;
int x2y2sensor = 0;
int x3y3sensor = 0;
int x4y4sensor = 0;

int x0y0btn = 0;
int x1y1btn = 0;
int x2y2btn = 0;
int x3y3btn = 0;
int x4y4btn = 0;

int sensorState = 0;
int lastSensorState = 0;

int buttonState = 0;
int lastButtonState = 0;

int Time1 = 50;
int Time2 = 500;
int Time3 = 650; // Piston time on or off

boolean isConnected = false;
```

Αρχικοποιήσεις

```
//----inputs
const int Piston_Opened = 11; // magnetic sensor input
bool Piston_State; // magnetic sensor status

const int Button = 10; // button input
```

Δήλωση εισόδων για τον μαγνητικό αισθητήρα και το κομβίο

```
//----outputs  
const int Open_Piston = 12;  
const int Close_Piston = 9;
```

Δήλωση εξόδων για το άνοιγμα και κλείσιμο του πνευματικού εμβόλου

```
//---- Stepper motors  
  
int stepsPerRevolution = 800; // total steps per revolution  
Stepper stepperX = Stepper(stepsPerRevolution, 2, 5); // X = pins 2,5  
Stepper stepperY = Stepper(stepsPerRevolution, 3, 6); // Y = pins 3,6  
Stepper stepperZ = Stepper(stepsPerRevolution, 4, 7); // Z = pins 4,7
```

Δήλωση εξόδων για την διαχείριση των βηματικών κινητήρων

```
//----  
  
bool XMFR = false;  
bool XMFL = false;  
  
bool YMFR = false;  
bool YMFL = false;  
  
bool ZMFR = false;  
bool ZMFL = false;
```

Αρχικοποίηση κομβίων

```
//-----initialization setup-----//  
  
void setup() {  
  Serial.begin(9600);  
  
  stepperX.setSpeed(400); //RPM  
  stepperY.setSpeed(400); //RPM  
  stepperZ.setSpeed(400); //RPM  
  
  pinMode(Open_Piston, OUTPUT);  
  pinMode(Close_Piston, OUTPUT);  
  
  pinMode(Piston_Opened, INPUT);  
  pinMode(Button, INPUT);  
  
  digitalWrite(Close_Piston, LOW);  
  digitalWrite(Open_Piston, HIGH);  
  delay(1000);  
  digitalWrite(Close_Piston, HIGH);  
  
  Serial.begin(9600);  
}
```

Στο setup καθορίζουμε την επικοινωνία με τον υπολογιστή το baud rate πάντα στα 9600 και ορίζουμε την ταχύτητα των βηματικών κινητήρων αλλά και τις καταστάσεις των pins δηλαδή εάν θα είναι είσοδοι ή έξοδοι.

```
//-----LOOP-----//  
  
void loop() {  
  
    ///-----command cotrol from serial port -----  
    ---//  
  
    if (stringComplete) {  
        stringComplete = false;  
  
        if (inputString.length() > 0) {  
            commandString = inputString.substring(1, 5);  
        }  
  
        if (commandString.equals("AUTO")) {  
            Auto = true;  
        }  
  
        if (commandString.equals("MANU")) {  
            Auto = false;  
        }  
  
        if (commandString.equals("STAR")) {  
            Cycle = true;  
  
            x1 = inputString.substring(5, 9);  
            y1 = inputString.substring(9, 13);  
            x2 = inputString.substring(13, 17);  
            y2 = inputString.substring(17, 21);  
            x3 = inputString.substring(21, 25);  
            y3 = inputString.substring(25, 29);  
            x4 = inputString.substring(29, 33);  
            y4 = inputString.substring(33, 37);  
  
            a_x1 = x1.toInt();  
            a_y1 = y1.toInt();  
            a_x2 = x2.toInt();  
            a_y2 = y2.toInt();  
            a_x3 = x3.toInt();  
            a_y3 = y3.toInt();  
            a_x4 = x4.toInt();  
            a_y4 = y4.toInt();  
        }  
    }  
}
```



```
x0y0str = inputString.substring(37, 40);  
x1y1str = inputString.substring(40, 43);  
x2y2str = inputString.substring(43, 46);  
x3y3str = inputString.substring(46, 49);  
x4y4str = inputString.substring(49, 52);  
  
a_x0y0str = x0y0str.toInt();  
a_x1y1str = x1y1str.toInt();  
a_x2y2str = x2y2str.toInt();  
a_x3y3str = x3y3str.toInt();  
a_x4y4str = x4y4str.toInt();  
  
Cycles = inputString.substring(52, 55);  
  
a_Cycles = Cycles.toInt();  
}  
  
if (commandString.equals("STOP")) {  
    Cycle = false;  
}
```

Δήλωση εντολών από την σειριακή

```
//-----manual operation -----//  
  
if (Auto == false) {  
  
    //----- X axis manual operation -----//  
    if (commandString.equals("XMFR")) {  
        XMFR = true;  
    } else {  
        XMFR = false;  
    }  
  
    if (commandString.equals("XMFL")) {  
        XMFL = true;  
    } else {  
        XMFL = false;  
    }  
  
    if (commandString.equals("XMMR")) {
```

```
String steps = inputString.substring(5, 7);
stepperX.setSpeed(100);
stepperX.step(-steps.toInt());
}

if (commandString.equals("XMML")) {
String steps = inputString.substring(5, 7);
stepperX.setSpeed(100);
stepperX.step(steps.toInt());
}
```

Χειροκίνητη λειτουργία άξονα X, το ίδιο ισχύει και για του Y,Z

```
//----- pneumatic piston - manual -----/////

if (commandString.equals("EKTS")) {
digitalWrite(Open_Piston, LOW);
digitalWrite(Close_Piston, HIGH);
delay(500);
digitalWrite(Open_Piston, HIGH);
}

if (commandString.equals("SIMP")) {
digitalWrite(Open_Piston, HIGH);
digitalWrite(Close_Piston, LOW);
delay(500);
digitalWrite(Close_Piston, HIGH);
}
}

inputString = "";
}
```

Χειροκίνητη έκταση και σύμπτυξη πνευματικού εμβόλου

```
//-----execution of orders-----//

//----- X axis manual -----//

if (XMFR == true) {
stepperX.setSpeed(100);
stepperX.step(-2);
}

if (XMFL == true) {
stepperX.setSpeed(100);
stepperX.step(2);
}
```

Χειροκίνητη λειτουργία άξονα X, το ίδιο ισχύει και για του Y,Z

```
//-----automatic operation -----  
//  
  
if (Auto == true) {  
  
    if (Cycle == true) {  
  
        stepperX.setSpeed(100);  
        stepperY.setSpeed(100);  
  
        //----- move to 0 point -----//  
  
        if (Auto_Count == 0) {  
  
            if (a_x0y0curstr <= a_x0y0str - 1) {  
  
                //----- open command to piston ----//  
                digitalWrite(Open_Piston, LOW);  
                digitalWrite(Close_Piston, HIGH);  
                delay(Time3);  
  
                //----- inputs read ----//  
  
                sensorState = digitalRead(Piston_Opened);  
                buttonState = digitalRead(Button);  
                delay(50);  
  
                //----- inputs status check ----//  
  
                // -- piston magnetic sensor --//  
                if (sensorState != lastSensorState) {  
                    if (sensorState == HIGH) {  
                        x0y0sensor = x0y0sensor + 1;  
                    }  
                    delay(50);  
                }  
  
                // -- button --//  
                if (buttonState != lastButtonState) {  
                    if (buttonState == HIGH) {  
                        x0y0btn = x0y0btn + 1;  
                    }  
                }  
            }  
        }  
    }  
}
```

```

        delay(50);
    }

    // -- serial print -- //

    Serial.println('0' + String(x0y0sensor));
    Serial.println('1' + String(x0y0btn));
    delay(50);

    lastSensorState = sensorState;
    lastButtonState = buttonState;

    //----- close command to piston ----//

    digitalWrite(Open_Piston, HIGH);
    digitalWrite(Close_Piston, LOW);
    delay(Time3);

    //----- inputs read ----//

    sensorState = digitalRead(Piston_Opened);
    buttonState = digitalRead(Button);
    delay(100);
    lastSensorState = sensorState;
    lastButtonState = buttonState;

    a_x0y0curstr = a_x0y0curstr + 1;
}

if (a_x0y0curstr >= a_x0y0str) {
    digitalWrite(Close_Piston, HIGH);
    delay(500);
}
}
}

```

Αυτόματη λειτουργία, μετακίνηση στο σημείο 0 με εντολή ανοίγματος και κλεισίματος εμβόλου, όπως επίσης με ανάγνωση εισόδων

```

//----- routine called when available data are in serial port -----
//-----//

void serialEvent() {
    while (Serial.available()) {
        // get the new byte:
        char inChar = (char)Serial.read();

        // add it to the inputString:

```

```
inputString += inChar;

// if the incoming character is a newline, set a flag
// so the main loop can do something about it:
if (inChar == '\n') {
    stringComplete = true;
}
}
}

if (Auto_Count == 0 && a_x0y0curstr >= a_x0y0str) {
    Serial.println('A');
    Auto_Count = Auto_Count + 1;
    a_x0y0curstr = 0;
    goto label1;
}

if (Auto_Count >= 1 && Auto_Count <= 2) {
    Auto_Count = Auto_Count + 1;
    goto label1;
}

if (Auto_Count == 3 && a_x1y1curstr >= a_x1y1str) {
    Serial.println('B');
    Auto_Count = Auto_Count + 1;
    a_x1y1curstr = 0;
    goto label1;
}

if (Auto_Count >= 4 && Auto_Count <= 7) {
    Auto_Count = Auto_Count + 1;
    goto label1;
}

if (Auto_Count == 8 && a_x2y2curstr >= a_x2y2str) {
    Serial.println('C');
    Auto_Count = Auto_Count + 1;
    a_x2y2curstr = 0;
    goto label1;
}

if (Auto_Count >= 9 && Auto_Count <= 12) {
    Auto_Count = Auto_Count + 1;
    goto label1;
}

if (Auto_Count == 13 && a_x3y3curstr >= a_x3y3str) {
    Serial.println('D');
```

```
    Auto_Count = Auto_Count + 1;
    a_x3y3curstr = 0;
    goto label1;
}

if (Auto_Count >= 14 && Auto_Count <= 17) {
    Auto_Count = Auto_Count + 1;
    goto label1;
}

if (Auto_Count == 18 && a_x4y4curstr >= a_x4y4str) {
    Serial.println('E');
    Auto_Count = Auto_Count + 1;
    a_x4y4curstr = 0;
    goto label1;
}

if (Auto_Count >= 19) {
    Auto_Count = Auto_Count + 1;
    goto label1;
}

label1:
    delay(1);
}

if (Cycle == false) {
    Auto_Count = 0;
    a_CyclesCur = 0;
    digitalWrite(Close_Piston, HIGH);
    delay(500);

    x0y0sensor = 0;
    x1y1sensor = 0;
    x2y2sensor = 0;
    x3y3sensor = 0;
    x4y4sensor = 0;

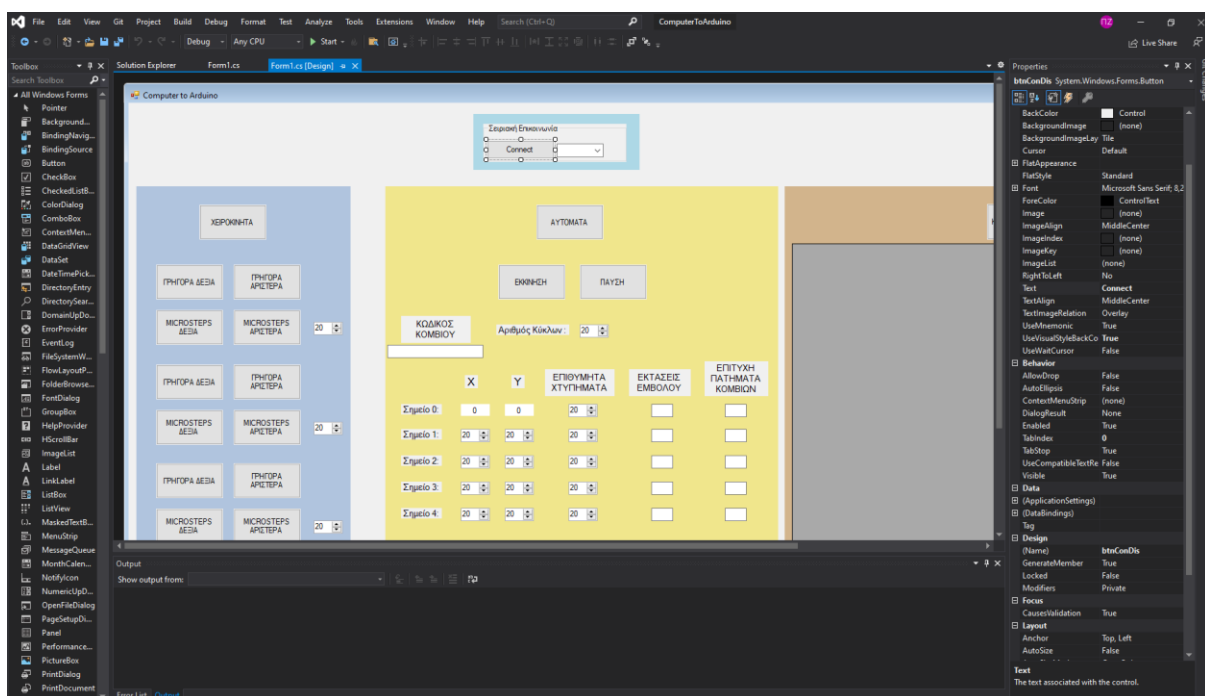
    x0y0btn = 0;
    x1y1btn = 0;
    x2y2btn = 0;
    x3y3btn = 0;
    x4y4btn = 0;
}
}
}
```

Κεφάλαιο 6^ο :

Ανάλυση προγράμματος C# :

Στο αντίστοιχο πρόγραμμα γραμμένο στην C# δηλώνουμε τις μεταβλητές, επικοινωνούμε μέσω της σειριακής με το Arduino και μέσω των κομβίων ελέγχουμε τον μηχανισμό.

Υπάρχουν αλληλομανδαλώσεις για να μην υπάρχει περίπτωση να πατηθούν κομβία κατά λάθος. Τέλος υπάρχει η επικοινωνία με την SQL για εγγραφή και ανάγνωση των δεδομένων.



Εικόνα προγράμματος C#

Ανάλυση κώδικα C# :

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.IO.Ports;
using System.Data.SqlClient;

namespace ComputerToArduino
{
    public partial class Form1 : Form
    {
        bool isConnected = false;
        String[] ports;
        SerialPort port;

        bool a_XMFR = false;
        bool a_XMFL = false;
        bool a_XMMR = false;
        bool a_XMML = false;

        bool a_YMFR = false;
        bool a_YMFL = false;
        bool a_YMMR = false;
        bool a_YMML = false;

        bool a_ZMFR = false;
        bool a_ZMFL = false;
        bool a_ZMMR = false;
        bool a_ZMML = false;

        //bool a_ektasi = false;
        //bool a_siptixi = false;

        delegate void serialCalback(string val);
        //private string ReceivedData;
```

Δήλωση μεταβλητών

```
public Form1()
{
    InitializeComponent();
    getAvailableComPorts();

    x1.Value = 1300;
    y1.Value = 1300;
    x2.Value = 1300;
    y2.Value = 1300;
    x3.Value = 1300;
    y3.Value = 1300;
    x4.Value = 1300;
    y4.Value = 1300;

    x0y0str.Value = 20;
    x1y1str.Value = 20;
    x2y2str.Value = 20;
```



```
x3y3str.Value = 20;  
x4y4str.Value = 20;  
  
Cycles.Value = 1;
```

Αρχικοποίηση μεταβλητών

```
foreach (string port in ports)  
{  
    cbxPorts.Items.Add(port);  
    Console.WriteLine(port);  
    if (ports[0] != null)  
    {  
        cbxPorts.SelectedItem = ports[0];  
    }  
}
```

Εύρεση διαθέσιμων σειριακών θυρών

```
//----- connect disconnect ARDUINO -----  
-//  
  
private void btnConDis_Click(object sender, EventArgs e)  
{  
    if (!isConnected)  
    {  
        connectToArduino();  
        //---- connection --- //  
        // btnConDis.Enabled = true;  
        // cbxPorts.Enabled = true;  
  
        // --- Manual ---//  
        btnManual.Enabled = true;  
        btnXMFR.Enabled = false;  
        btnXMFL.Enabled = false;  
        btnXMMR.Enabled = false;  
        btnXMML.Enabled = false;  
        btnYMFR.Enabled = false;  
        btnYMFL.Enabled = false;  
        btnYMMR.Enabled = false;  
        btnYMML.Enabled = false;  
        btnZMFR.Enabled = false;  
        btnZMFL.Enabled = false;  
        btnZMMR.Enabled = false;  
        btnZMML.Enabled = false;  
        btnStayOpen.Enabled = false;  
        btnStayClose.Enabled = false;  
        numXmicro.Enabled = false;  
        numYmicro.Enabled = false;  
        numZmicro.Enabled = false;
```

```
// --- Auto---//
    btnAuto.Enabled = true;
    btnStart.Enabled = false;
    btnStop.Enabled = false;

    Cycles.Enabled = false;
    x1.Enabled = false;
    y1.Enabled = false;
    x2.Enabled = false;
    y2.Enabled = false;
    x3.Enabled = false;
    y3.Enabled = false;
    x4.Enabled = false;
    y4.Enabled = false;
    x0y0str.Enabled = false;
    x1y1str.Enabled = false;
    x2y2str.Enabled = false;
    x3y3str.Enabled = false;
    x4y4str.Enabled = false;
    x0y0fedt.Enabled = false;
    x1y1fedt.Enabled = false;
    x2y2fedt.Enabled = false;
    x3y3fedt.Enabled = false;
    x4y4fedt.Enabled = false;
    x0y0suct.Enabled = false;
    x1y1suct.Enabled = false;
    x2y2suct.Enabled = false;
    x3y3suct.Enabled = false;
    x4y4suct.Enabled = false;

}
else
{
    disconnectFromArduino();

//---- connection --- //
    // btnConDis.Enabled = true;
    // cbxPorts.Enabled = true;

    // --- Manual ---//
    btnManual.Enabled = false;
    btnXMFR.Enabled = false;
    btnXMFL.Enabled = false;
    btnXMMR.Enabled = false;
    btnXMML.Enabled = false;
    btnYMFR.Enabled = false;
    btnYMFL.Enabled = false;
    btnYMMR.Enabled = false;
    btnYMML.Enabled = false;
    btnZMFR.Enabled = false;
    btnZMFL.Enabled = false;
    btnZMMR.Enabled = false;
    btnZMML.Enabled = false;
    btnStayOpen.Enabled = false;
    btnStayClose.Enabled = false;
    numXmicro.Enabled = false;
    numYmicro.Enabled = false;
    numZmicro.Enabled = false;

```

```
// --- Auto---//  
btnAuto.Enabled = false;  
btnStart.Enabled = false;  
btnStop.Enabled = false;  
  
Cycles.Enabled = false;  
x1.Enabled = false;  
y1.Enabled = false;  
x2.Enabled = false;  
y2.Enabled = false;  
x3.Enabled = false;  
y3.Enabled = false;  
x4.Enabled = false;  
y4.Enabled = false;  
x0y0str.Enabled = false;  
x1y1str.Enabled = false;  
x2y2str.Enabled = false;  
x3y3str.Enabled = false;  
x4y4str.Enabled = false;  
x0y0fedt.Enabled = false;  
x1y1fedt.Enabled = false;  
x2y2fedt.Enabled = false;  
x3y3fedt.Enabled = false;  
x4y4fedt.Enabled = false;  
x0y0suct.Enabled = false;  
x1y1suct.Enabled = false;  
x2y2suct.Enabled = false;  
x3y3suct.Enabled = false;  
x4y4suct.Enabled = false;  
}  
  
}
```

Σύνδεση ή αποσύνδεση με το Arduino ενεργοποίηση και απενεργοποίηση κομβίων στο πρόγραμμα

```
void getAvailableComPorts()  
{  
    ports = SerialPort.GetPortNames();  
}  
  
private void connectToArduino()  
{  
    isConnected = true;  
    string selectedPort = cbxPorts.GetItemText(cbxPorts.SelectedItem);  
    port = new SerialPort(selectedPort, 9600, Parity.None, 8, StopBits.One);  
    port.Open();  
    btnConDis.Text = "Disconnect";  
  
    port.DataReceived += port_DataReceived;  
}
```

Σύνδεση με την διαθέσιμη σειριακή θύρα

```
private void disconnectFromArduino()  
{  
    isConnected = false;  
    port.Close();  
}
```

```
btnConDis.Text = "Connect";
}
```

Αποσύνδεση με την διαθέσιμη σειριακή θύρα

```
//----- connect disconnect ARDUINO -----
-//

//-----X axis manual -----//

private void btnXMFR_MouseDown(object sender, MouseEventArgs e)
{
    if (a_XMFR == false)
    {
        port.Write("#XMFR\n");
        a_XMFR = true;
    }
}

private void btnXMFR_MouseUp(object sender, MouseEventArgs e)
{
    port.Write("#XOFF\n");
    a_XMFR = false;
}

private void btnXMFL_MouseDown(object sender, MouseEventArgs e)
{
    if (a_XMFL == false)
    {
        port.Write("#XMFL\n");
        a_XMFL = true;
    }
}

private void btnXMFL_MouseUp(object sender, MouseEventArgs e)
{
    port.Write("#XOFF\n");
    a_XMFL = false;
}

private void btnXMMR_MouseDown(object sender, MouseEventArgs e)
{
    if (a_XMMR == false)
    {
        port.Write("#XMMR" + Convert.ToString(numXmicro.Value) + "\n");
        a_XMMR = true;
    }
}

private void btnXMMR_MouseUp(object sender, MouseEventArgs e)
{
    port.Write("#XOFF\n");
    a_XMMR = false;
}

private void btnXMML_MouseDown(object sender, MouseEventArgs e)
{
    if (a_XMML == false)
    {
```

```

        port.Write("#XMML" + Convert.ToString(numXmicro.Value) + "\n");
        a_XMML = true;
    }
}

private void btnXMML_MouseUp(object sender, MouseEventArgs e)
{
    port.Write("#XOFF\n");
    a_XMML = false;
}

```

Διασύνδεση κομβίων που έχουν να κάνουν με την κίνηση του άξονα X με την σειριακή

```

//-----piston manual -----//
private void btnStayOpen_MouseDown(object sender, MouseEventArgs e)
{
    port.Write("#EKTS\n");
}

private void btnStayClose_MouseDown(object sender, MouseEventArgs e)
{
    port.Write("#SIMP\n");
}

private void btnManual_Click(object sender, EventArgs e)
{
    port.Write("#MANU\n");
}

```

Διασύνδεση κομβίων που έχουν να κάνουν με το πνευματικό έμβολο με την σειριακή

```

private void btnStart_Click(object sender, EventArgs e)
{
    int vx1 = Convert.ToInt32(x1.Value);
    int vy1 = Convert.ToInt32(y1.Value);
    int vx2 = Convert.ToInt32(x2.Value);
    int vy2 = Convert.ToInt32(y2.Value);
    int vx3 = Convert.ToInt32(x3.Value);
    int vy3 = Convert.ToInt32(y3.Value);
    int vx4 = Convert.ToInt32(x4.Value);
    int vy4 = Convert.ToInt32(y4.Value);

    int vx0y0str = Convert.ToInt32(x0y0str.Value);
    int vx1y1str = Convert.ToInt32(x1y1str.Value);
    int vx2y2str = Convert.ToInt32(x2y2str.Value);
    int vx3y3str = Convert.ToInt32(x3y3str.Value);
    int vx4y4str = Convert.ToInt32(x4y4str.Value);

    int vCycles = Convert.ToInt32(Cycles.Value);

    port.Write("#STAR" + vx1.ToString("D4") + vy1.ToString("D4") +
vx2.ToString("D4") + vy2.ToString("D4") + vx3.ToString("D4") + vy3.ToString("D4") +

```

```
vx4.ToString("D4") + vy4.ToString("D4") + vx0y0str.ToString("D3") +  
vx1y1str.ToString("D3") + vx2y2str.ToString("D3") + vx3y3str.ToString("D3") +  
vx4y4str.ToString("D3") + vCycles.ToString("D3") + "\n");
```

Διασύνδεση κομβίου START με τις τιμές από όλα τα πεδία με την σειριακή

```
private void btnStop_Click(object sender, EventArgs e)  
{  
    port.Write("#STOP\n");  
}
```

Διασύνδεση κομβίου STOP με την σειριακή

```
private void port_DataReceived(object sender, SerialDataReceivedEventArgs e)  
{  
    string incomString = port.ReadLine();  
    char[] b = new char[incomString.Length];  
    b = incomString.ToCharArray();  
  
    if (b[0] == '0')  
    {  
        string a = incomString.Substring(1);  
        setText0(a);  
    }  
  
    if (b[0] == '1')  
    {  
        string a = incomString.Substring(1);  
        setText1(a);  
    }  
  
    if (b[0] == '2')  
    {  
        string a = incomString.Substring(1);  
        setText2(a);  
    }  
  
    if (b[0] == '3')  
    {  
        string a = incomString.Substring(1);  
        setText3(a);  
    }  
  
    if (b[0] == '4')  
    {  
        string a = incomString.Substring(1);  
        setText4(a);  
    }  
  
    if (b[0] == '5')  
    {  
        string a = incomString.Substring(1);  
        setText5(a);  
    }  
}
```

```
if (b[0] == '6')
{
    string a = incomString.Substring(1);
    setText6(a);
}

if (b[0] == '7')
{
    string a = incomString.Substring(1);
    setText7(a);
}

if (b[0] == '8')
{
    string a = incomString.Substring(1);
    setText8(a);
}

if (b[0] == '9')
```

Διασύνδεση σειριακής (Arduino) με το πρόγραμμα, μας στέλνει τις τιμές του

```
if (b[0] == 'A')
{
    writeDB(btncode.Text, "0", x0y0fedt.Text, x0y0suct.Text);
}

if (b[0] == 'B')
{
    writeDB(btncode.Text, "1", x1y1fedt.Text, x1y1suct.Text);
}

if (b[0] == 'C')
{
    writeDB(btncode.Text, "2", x2y2fedt.Text, x2y2suct.Text);
}

if (b[0] == 'D')
{
    writeDB(btncode.Text, "3", x3y3fedt.Text, x3y3suct.Text);
}

if (b[0] == 'E')
{
    writeDB(btncode.Text, "4", x4y4fedt.Text, x4y4suct.Text);
}

}
```

Διασύνδεση σειριακής (Arduino) με το πρόγραμμα, μας στέλνει τις τιμές του και την εντολή στην SQL να γράψει τις τιμές

Κεφάλαιο 7^ο :

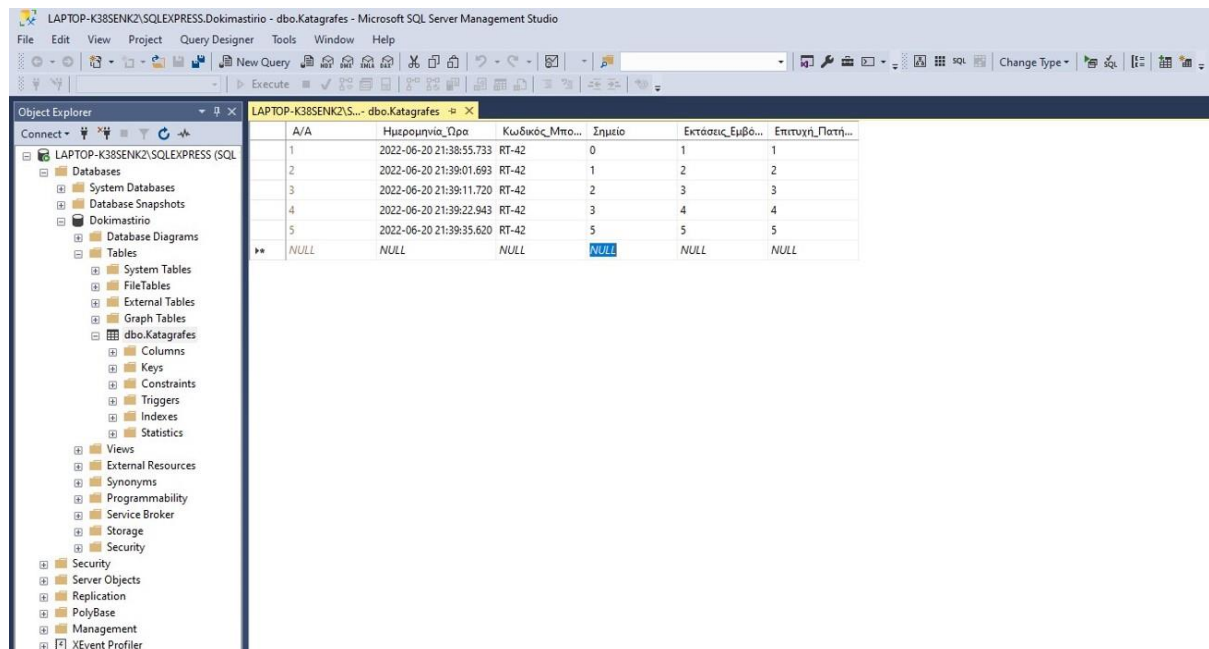
Ανάλυση προγράμματος SQL :

Η γλώσσα SQL ακρωνύμιο της Structured Query Language, σχεδιάστηκε για τη διαχείριση δεδομένων, σε ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (Relational Database Management System, RDBMS).

Για να γίνει η σύνδεση με τη βάση δεδομένων πρέπει πρώτα να δημιουργηθεί το `connectionString` που περιέχει πληροφορίες για τη βάση.

Επίσης για κάθε ενέργεια που πραγματοποιείται στη βάση (read - write), χρησιμοποιούνται SQL commands, δηλαδή εντολές της γλώσσας SQL η οποία αφορά αποκλειστικά τις βάσεις δεδομένων.

Για να υλοποιηθούν όλα τα παραπάνω έχει γραφεί ο κώδικας που παρατίθεται στην παρακάτω ανάλυση.



A/A	Ημερομηνία_Όρα	Κωδικός_Μπο...	Σημείο	Εκτάσεις_Εμβό...	Επιτυχη_Πατή...
1	2022-06-20 21:38:55.733	RT-42	0	1	1
2	2022-06-20 21:39:01.693	RT-42	1	2	2
3	2022-06-20 21:39:11.720	RT-42	2	3	3
4	2022-06-20 21:39:22.943	RT-42	3	4	4
5	2022-06-20 21:39:35.620	RT-42	5	5	5
***	NULL	NULL	NULL	NULL	NULL

Εικόνα βάσης δεδομένων SQL

Ανάλυση κώδικα SQL :

```
private void writeDB(string a,string b,string c,string d)
{
    string fc1 = a; //button code
    string fc2 = b; //point of impact
    string fc3 = c; //piston open
    string fc4 = d; //success strikes

    string connectionString = "Data Source=LAPTOP-K38SENK2\\SQLEXPRESS;Initial
Catalog=Dokimastirio;Integrated Security=True";

    string sqlQuery1 = " INSERT INTO Katagrafes
(Κωδικός_Μπουτόν, Σημείο, Εκτάσεις_Εμβόλου, Επιτυχή_Πατήματα) VALUES
(@fc1,@fc2,@fc3,@fc4) ";

    SqlConnection con = new SqlConnection(connectionString);

    con.Open();

    SqlCommand sc1 = new SqlCommand(sqlQuery1, con);

    sc1.Parameters.AddWithValue("@fc1", fc1);
    sc1.Parameters.AddWithValue("@fc2", fc2);
    sc1.Parameters.AddWithValue("@fc3", fc3);
    sc1.Parameters.AddWithValue("@fc4", fc4);

    sc1.ExecuteNonQuery();

    con.Close();
}
```

Διασύνδεση που γράφει στην SQL

```
private void btnshow_Click(object sender, EventArgs e)
{
    string connectionString = "Data Source=LAPTOP-
K38SENK2\\SQLEXPRESS;Initial Catalog=Dokimastirio;Integrated Security=True";
    SqlConnection con = new SqlConnection(connectionString);
    con.Open();
    SqlDataAdapter sqlDa = new SqlDataAdapter("SELECT
Ημερομηνία_Ώρα,Κωδικός_Μπουτόν, Σημείο, Εκτάσεις_Εμβόλου, Επιτυχή_Πατήματα FROM
Katagrafes ORDER BY Ημερομηνία_Ώρα ASC", con);
    DataTable dtbl = new DataTable();

    dataGridView1.DefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter;
    dataGridView1.ColumnHeadersDefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleCenter;

    dataGridView1.DataSource = dtbl;

    sqlDa.Fill(dtbl);

    con.Close();
}
```

Διασύνδεση που διαβάζει από την SQL

Κεφάλαιο 8^ο : Δοκιμές

Ανάλυση δοκιμίων στρόγγυλων κομβίων :

Κομβίο Α :

Το πρώτο στρόγγυλο κομβίο που δοκιμάστηκε είναι παραγωγής ενός γνωστού γερμανικού οίκου με παρουσία στο χώρο για πολλές δεκαετίες. Το εν λόγω κομβίο είναι μοντέλο με αντιβανδαλικές προδιαγραφές κλάσης 1 βάση του προτύπου EN_81-71 και κατασκευάζεται περίπου 15 συναπτά έτη.

Λόγω της απειρίας στον χειρισμό του μηχανισμού και υπό τον φόβο αστοχίας σε οποιοδήποτε εξάρτημα, η δοκιμή διεξήχθη με αρκετές διακοπές και με πολύωρη παρατήρηση.

Εφόσον βεβαιώθηκε ότι η δοκιμή συνεχίζεται απρόσκοπτα, η παρατήρηση συνεχίστηκε με την χρήση Webcam και μέσω του προγράμματος απομακρυσμένης διαχείρισης Any desk.

Παρατηρήθηκαν σημάδια κόπωσης του μηχανισμού στις πρώτες ώρες λειτουργίας του, ποιο συγκεκριμένα η θερμοκρασία λειτουργίας των βηματικών κινητήρων ανέβηκε σε δυσθεώρητα ύψη και επιλέχθηκε η χρήση ψήκτρας αλουμινίου με εξαιρετικά αποτελέσματα.

Το δοκίμιο εξεταζόταν με αφαίρεση από την βάση δοκιμής κάθε περίπου 50000 χτυπήματα για να ελεγχθεί μέσω αφής για αλλαγή στην ανάδραση του μηχανισμού επαναφοράς στην αρχική θέση και συγκρινόμενο με ένα ακόμη κομβίο αναφοράς ίδιου τύπου.

Τις πρώτες τέσσερις ημέρες στα περίπου 200000 πατήματα η συμπεριφορά του κομβίου ήταν καθόλα άψογη και πανομοιότυπη με το κομβίο αναφοράς.

Μετά από την 5^η ημέρα δοκιμών δηλαδή στα περίπου 245000 πατήματα, η καταπόνηση ήταν εμφανής με κύριο χαρακτηριστικό την ασάφεια θέσης κατά την επαφή με το κινούμενο μέρος του και με ελαφρά ασάφεια κατά την επαναφορά μετά το πάτημα του.

Στην 9^η ημέρα της δοκιμής και μετά από περίπου 439000 πατήματα ο διακόπτης του κομβίου σταμάτησε να λειτουργεί. Η επαναφορά μετά το πάτημα του παρέμεινε με ελαφρά ασάφεια που είχε αποκτήσει.



Εικόνα πρώτη δοκιμή στρόγγυλου κομβίου

Ο παρακάτω πίνακας δοκιμών απεικονίζει αναλυτικά ανά κύκλο ημέρας το πλήθος των εκτάσεων του εμβόλου συγκριτικά με τα επιτευχθέντα πατήματα του κομβίου Α.



Αναλυτικός πίνακας δοκιμών ανά ημέρα κομβίου Α

Κομβίο Β :

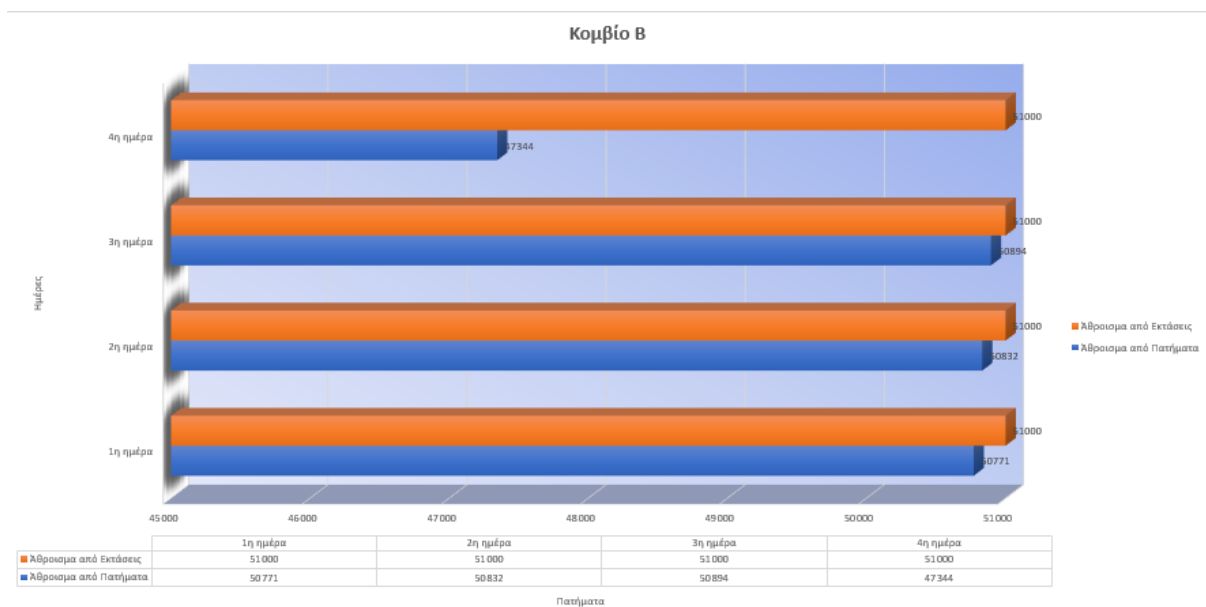
Το δεύτερο στρόγγυλο κομβίο που δοκιμάστηκε είναι παραγωγής ενός οίκου εκτός Ευρώπης με μικρή εμπειρία στο χώρο της κατασκευής κομβίων αλλά και ως μοντέλο παραγωγής έχει μόλις 2 έτη στην αγορά.

Το δοκίμιο εξεταζόταν με αφαίρεση από την βάση δοκιμής κάθε περίπου 51000 χτυπήματα για να ελεγχθεί μέσω αφής για αλλαγή στην ανάδραση του μηχανισμού επαναφοράς στην αρχική θέση και συγκρινόμενο με ένα ακόμη κομβίο αναφοράς ίδιου τύπου.

Τις πρώτες τρεις ημέρες στα περίπου 150000 πατήματα η συμπεριφορά του κομβίου ήταν καθόλα άψογη και πανομοιότυπη με το κομβίο αναφοράς.

Την 4η ημέρα δοκιμών δηλαδή στα περίπου 200000 πατήματα, ο διακόπτης του κομβίου σταμάτησε να λειτουργεί. Η επαναφορά μετά το πάτημα του ήταν υποδειγματική και πανομοιότυπη με το κομβίο αναφοράς.

Ο παρακάτω πίνακας δοκιμών απεικονίζει αναλυτικά ανά κύκλο ημέρας το πλήθος των εκτάσεων του εμβόλου συγκριτικά με τα επιτευχθέντα πατήματα του κομβίου Β.



Αναλυτικός πίνακας δοκιμών ανά ημέρα κομβίου Β

Κομβίο Γ :

Το τρίτο στρόγγυλο κομβίο που δοκιμάστηκε είναι παραγωγής ενός οίκου εντός Ευρώπης με μεγάλη εμπειρία στο χώρο της κατασκευής κομβίων με το συγκεκριμένο μοντέλο παραγωγής να μετρά αρκετά χρόνια στην αγορά.

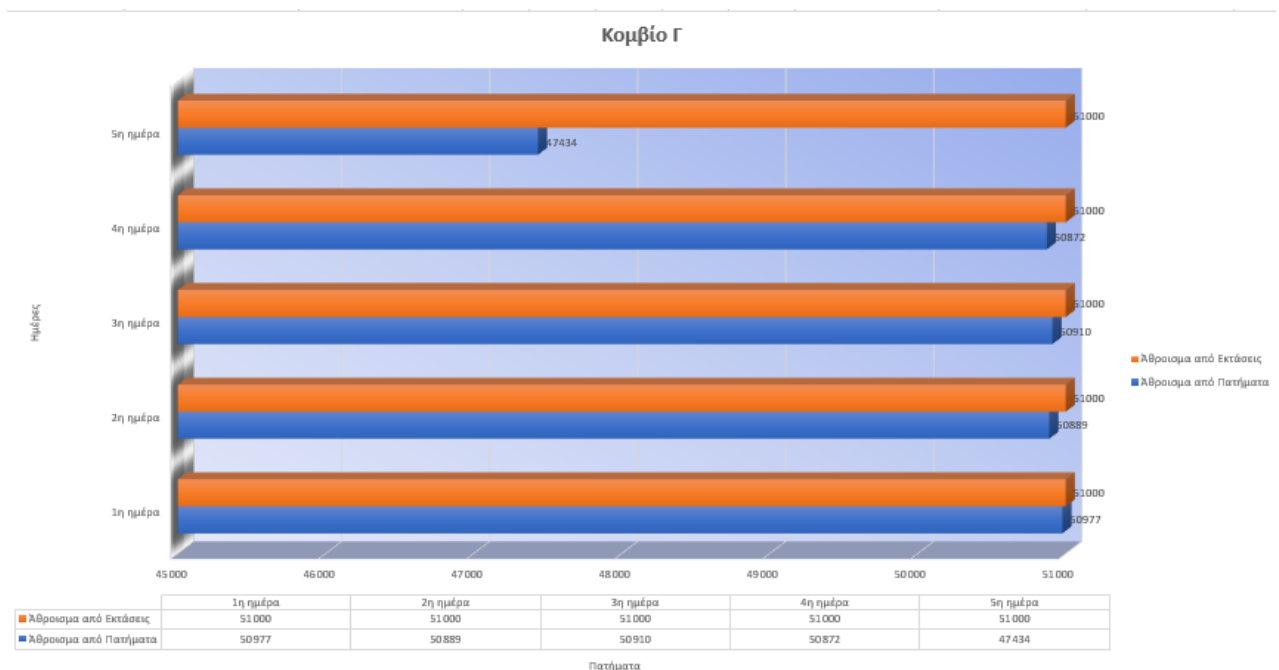
Το δοκίμιο εξεταζόταν με αφαίρεση από την βάση δοκιμής κάθε περίπου 51000 χτυπήματα για να ελεγχθεί μέσω αφής για αλλαγή στην ανάδραση του μηχανισμού επαναφοράς στην αρχική θέση και συγκρινόμενο με ένα ακόμη κομβίο αναφοράς ίδιου τύπου.

Τις πρώτες δύο ημέρες στα περίπου 100000 πατήματα η συμπεριφορά του κομβίου ήταν καθόλα άψογη και πανομοιότυπη με το κομβίο αναφοράς.

Την 3η ημέρα δοκιμών δηλαδή στα περίπου 150000 πατήματα, παρατηρήθηκε ασάφεια θέσης κατά την επαφή με το κινούμενο μέρος του και μια ελαφρά ασάφεια κατά την επαναφορά μετά το πάτημα του.

Την 4η ημέρα δοκιμών δηλαδή στα περίπου 250000 πατήματα, ο διακόπτης του κομβίου σταμάτησε να λειτουργεί, ενώ η επαναφορά μετά το πάτημα του παραμένει με την ελαφρά ασάφεια που είχε αποκτήσει.

Ο παρακάτω πίνακας δοκιμών απεικονίζει αναλυτικά ανά κύκλο ημέρας το πλήθος των εκτάσεων του εμβόλου συγκριτικά με τα επιτευχθέντα πατήματα του κομβίου Γ.



Αναλυτικός πίνακας δοκιμών ανά ημέρα κομβίου Γ

Ανάλυση δοκιμών τετράγωνων κομβίων :

Κομβίο Δ :

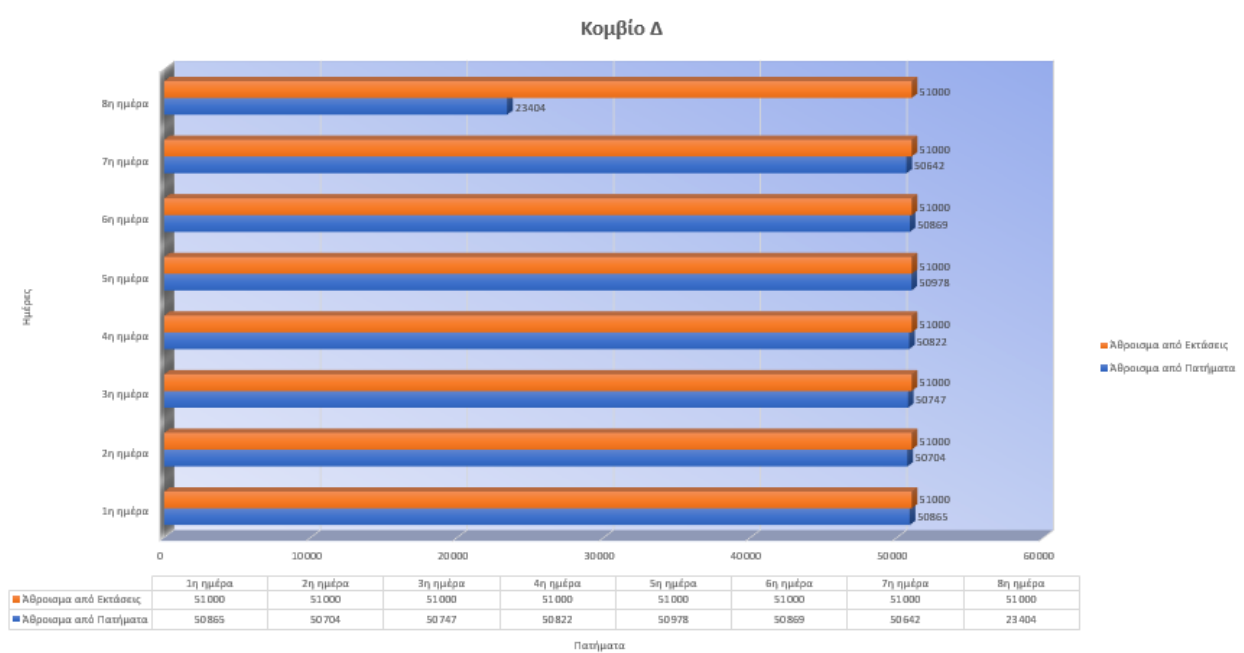
Το πρώτο τετράγωνο κομβίο που δοκιμάστηκε είναι παραγωγής του ίδιου γνωστού γερμανικού οίκου με το πρώτο στρόγγυλο που δοκιμάστηκε. Το εν λόγω κομβίο είναι μοντέλο με παρουσία στην αγορά για τουλάχιστον 25 συναπτά έτη.

Το δοκίμιο εξεταζόταν με αφαίρεση από την βάση δοκιμής κάθε περίπου 51000 χτυπήματα για να ελεγχθεί μέσω αφής για αλλαγή στην ανάδραση του μηχανισμού επαναφοράς στην αρχική θέση και συγκρινόμενο με ένα ακόμη κομβίο αναφοράς ίδιου τύπου.

Μετά τις πρώτες τρεις ημέρες στα περίπου 150000 πατήματα η συμπεριφορά του κομβίου η καταπόνηση ήταν εμφανής με κύριο χαρακτηριστικό την ασάφεια θέσης κατά την επαφή με το κινούμενο μέρος του και με ελαφρά ασάφεια κατά την επαναφορά μετά το πάτημα του.

Στην 4η ημέρα της δοκιμής και μετά από περίπου 200000 πατήματα η ασάφεια θέσης κατά την επαφή με το κινούμενο μέρος του αλλά και η επαναφορά μετά το πάτημα του παραμένει με την ελαφρά ασάφεια που είχε αποκτήσει.

Ο παρακάτω πίνακας δοκιμών απεικονίζει αναλυτικά ανά κύκλο ημέρας το πλήθος των εκτάσεων του εμβόλου συγκριτικά με τα επιτευχθέντα πατήματα του κομβίου Δ.



Αναλυτικός πίνακας δοκιμών ανά ημέρα κομβίου Δ

Κομβίο Ε :

Το δεύτερο τετράγωνο κομβίο που δοκιμάστηκε είναι παραγωγής γνωστού γερμανικού οίκου. Το εν λόγω κομβίο είναι μοντέλο με παρουσία στην αγορά για τουλάχιστον 14 συναπτά έτη.

Το δοκίμιο εξεταζόταν με αφαίρεση από την βάση δοκιμής κάθε περίπου 51000 χτυπήματα για να ελεγχθεί μέσω αφής για αλλαγή στην ανάδραση του μηχανισμού επαναφοράς στην αρχική θέση και συγκρινόμενο με ένα ακόμη κομβίο αναφοράς ίδιου τύπου.

Μετά τις πρώτες τρεις ημέρες στα περίπου 150000 πατήματα στην συμπεριφορά του κομβίου παρατηρήθηκε ασάφεια θέσης κατά την επαφή με το κινούμενο μέρος του και μια ελαφρά ασάφεια κατά την επαναφορά μετά το πάτημα του.

Στην 6^η ημέρα της δοκιμής και μετά από περίπου 300000 πατήματα η ασάφεια θέσης κατά την επαφή με το κινούμενο μέρος του αλλά και η επαναφορά μετά το πάτημα του παραμένει σταθερή.

Στην 7^η ημέρα της δοκιμής και μετά από περίπου 350000 πατήματα ο διακόπτης του κομβίου σταμάτησε να λειτουργεί. Η επαναφορά μετά το πάτημα του απέκτησε μεγαλύτερη ασάφεια.

Ο παρακάτω πίνακας δοκιμών απεικονίζει αναλυτικά ανά κύκλο ημέρας το πλήθος των εκτάσεων του εμβόλου συγκριτικά με τα επιτευχθέντα πατήματα του κομβίου Ε.



Αναλυτικός πίνακας δοκιμών ανά ημέρα κομβίου Ε

Κομβίο ΣΤ :

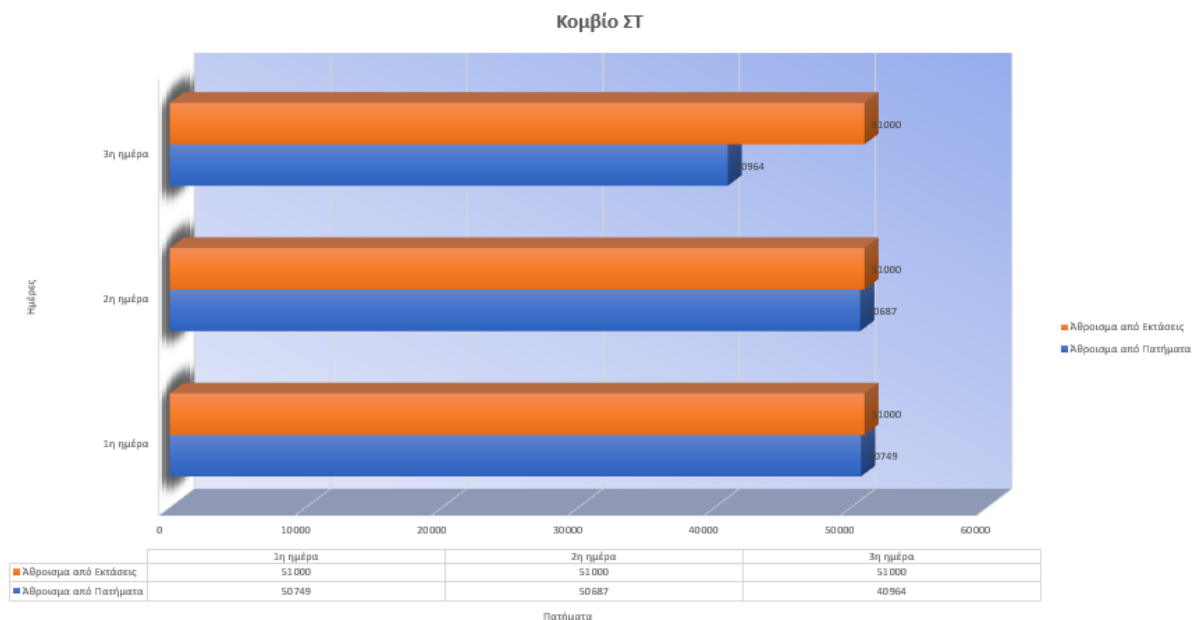
Το τρίτο τετράγωνο κομβίο που δοκιμάστηκε είναι παραγωγής ενός άγνωστου Κινεζικού οίκου. Είναι μοντέλο με παρουσία στην αγορά για τουλάχιστον 10 έτη.

Το δοκίμιο εξεταζόταν με αφαίρεση από την βάση δοκιμής κάθε περίπου 51000 χτυπήματα για να ελεγχθεί μέσω αφής για αλλαγή στην ανάδραση του μηχανισμού επαναφοράς στην αρχική θέση και συγκρινόμενο με ένα ακόμη κομβίο αναφοράς ίδιου τύπου.

Μετά τις πρώτες δύο ημέρες στα περίπου 100000 πατήματα στην συμπεριφορά του κομβίου παρατηρήθηκε έντονη ασάφεια θέσης κατά την επαφή με το κινούμενο μέρος του και μια ελαφρά ασάφεια κατά την επαναφορά μετά το πάτημα του.

Στην 3^η ημέρα της δοκιμής και μετά από περίπου 142000 πατήματα ο διακόπτης του κομβίου σταμάτησε να λειτουργεί. Η επαναφορά μετά το πάτημα του απέκτησε μεγαλύτερη ασάφεια.

Ο παρακάτω πίνακας δοκιμών απεικονίζει αναλυτικά ανά κύκλο ημέρας το πλήθος των εκτάσεων του εμβόλου συγκριτικά με τα επιτευχθέντα πατήματα του κομβίου ΣΤ.



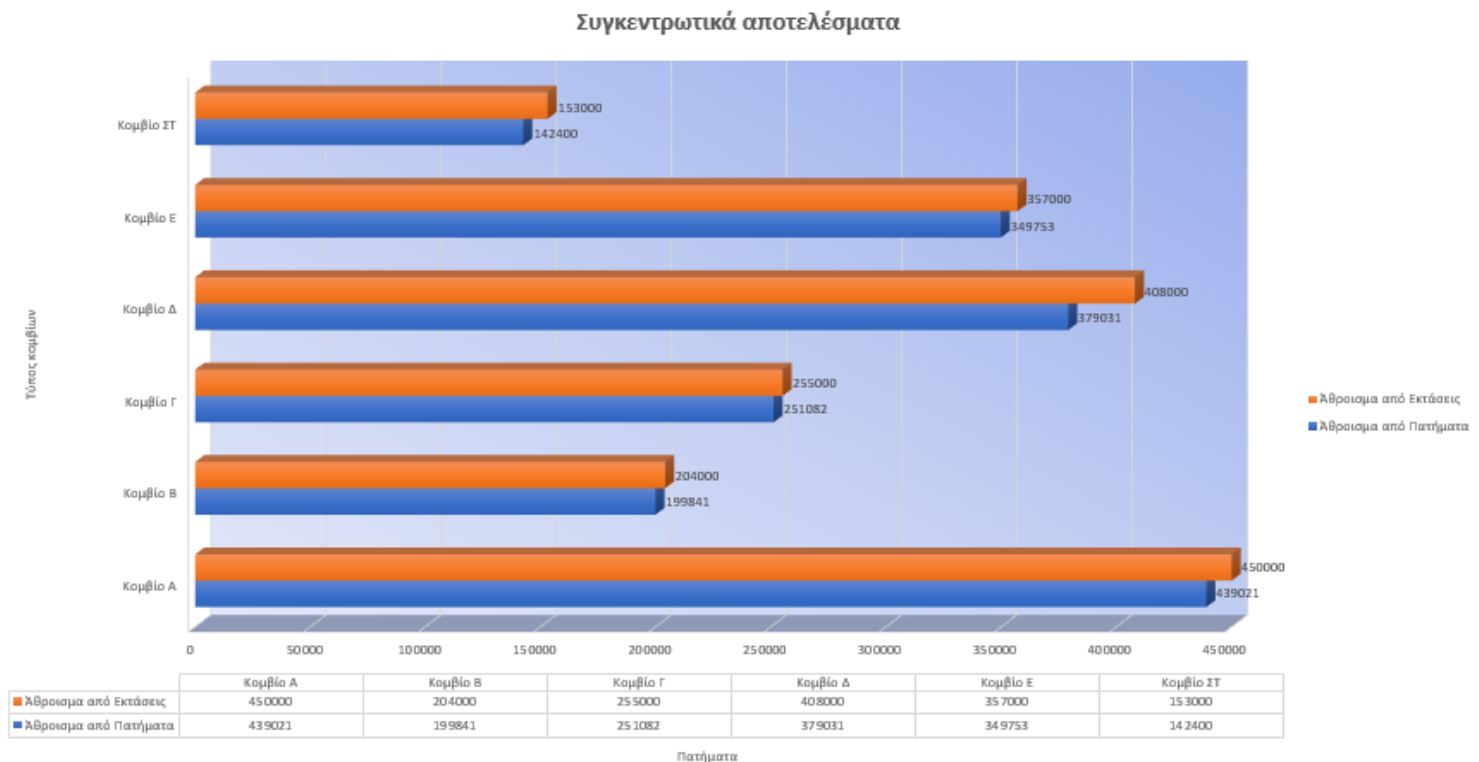
Αναλυτικός πίνακας δοκιμών ανά ημέρα κομβίου ΣΤ

Συγκεντρωτική ανάλυση αποτελεσμάτων :

Μετά από δοκιμή συνολικά έξι κομβίων, τριών στρόγγυλων και τριών τετράγωνων, δηλαδή των δύο επικρατέστερων ανά τον κόσμο σχημάτων κομβίων, μπορούμε να συμπεράνουμε με ασφάλεια ότι σχετικά με την ποιότητα των υλικών αλλά και ταυτόχρονα την ποιότητα κατασκευής δίνεται εμφανή προβάδισμα στα κομβία κατασκευασμένα από οίκους της κεντρικής Ευρώπης.

Φυσικά στον αντίποδα όλων αυτών έχουμε το κόστος κτήσης που είναι συνυφασμένο με την συνολική διάρκεια ζωής ενός κομβίου.

Ο παρακάτω πίνακας απεικονίζει συνολικά το πλήθος των εκτάσεων του εμβόλου συγκριτικά με τα επιτευχθέντα πατήματα ξεχωριστά για το κάθε κομβίο (Α έως ΣΤ).



Συγκεντρωτικός αναλυτικός πίνακας δοκιμών ανά κομβίο (Α έως ΣΤ)

Συμπεράσματα και προτάσεις για την πιθανή εξέλιξη του συστήματος :

Η παρούσα ερευνητική εργασία στον χώρο των μηχανισμών δοκιμής, θέλησε να εξετάσει το πεδίο δοκιμών της διάρκειας ζωής των διακοπών για χρήση σε κομβία, καθώς όλες οι συσκευές δοκιμής έχουν μια διάρκεια ζωής μικρότερη των 10.000 - 50.000 δοκιμών ενώ εάν ξεπεράσουν τις 100.000 δοκιμές θα παρουσιαστούν έντονα φαινόμενα κόπωσης και θα καταστραφούν.

Το μοντέλο που κατασκευάστηκε έχει τα πλεονεκτήματα της απλής δομής και του χαμηλού κόστους κατασκευής και συντήρησης και εφαρμόζει μια διαφορετική προσέγγιση καθώς το σημείο πατήματος πάνω στο προς δοκιμή κομβίο αλλάζει σε έως και 5 διαφορετικές θέσεις.

Η επίτευξη των στόχων που τέθηκαν, ολοκληρώθηκε, καθώς ο μηχανισμός είναι σε θέση να καταγράφει ότι το πνευματικό έμβολο χτυπήματος λειτούργησε (μέσω της ανάδρασης από τον μαγνητικό αισθητήρα), όπως επίσης και ότι το κομβίο προς δοκιμή πατήθηκε και ενεργοποιήθηκε.

Οι καταγραφές του μηχανισμού αποτυπώνουν την πραγματικότητα, δηλαδή ότι τα κομβία δεν προσεγγίζουν το προσδόκιμο ζωής των μικροδιακοπών αλλά εμφανίζουν σημάδια κόπωσης και εντέλει καταστροφής τους σε πολύ μικρότερα μεγέθη του αγγίζουν το 40-60% του προσδόκιμου ζωής.

Πρέπει να σημειωθεί ότι δεν υπήρξε πρότερη εμπειρία σε τέτοιου είδους δοκιμές και τις περισσότερες φορές χρειάστηκε να αυτοσχεδιάσουμε για να επιτύχουμε τον στόχο μας.

Επιπλέον ο μηχανισμός της διπλωματικής εργασίας δεν κατασκευάστηκε σε επαγγελματικό χώρο αλλά σε οικία, όπου όταν ολοκληρώθηκε και ξεκίνησαν οι δοκιμές, αντικειμενικά προκλήθηκαν πολλαπλά προβλήματα. Έπρεπε οι δοκιμές να είναι συνεχείς δηλαδή επί 24ώρου βάσεως.

Τροχοπέδη στο όλο εγχείρημα ο εκκωφαντικός θόρυβος που παράγεται από τον αεροσυμπιεστή, με αποτέλεσμα η κατασκευή να μεταφερθεί σε διαφορετικό χώρο (υπόγειο) και να «στηθεί» ένα δίκτυο παρακολούθησης με την βοήθεια κάμερας και χρήσης προγράμματος απομακρυσμένης διαχείρισης υπολογιστή (Any Desk) .

Θα μπορούσαμε να επεκτείνουμε την χρήση του συγκεκριμένου μηχανισμού για την δοκιμή διακοπών όχι πλέον κομβίων αλλά από τελείως διαφορετική κατηγορία διακοπών, παραδείγματος χάρη διακόπτες on/off τύπου toggle switch.

Είναι γεγονός ότι ο μηχανισμός που κατασκευάστηκε απέδειξε περίτρανα ότι αποτελεί ένα εξαιρετικών δυνατοτήτων σύστημα, καθώς τα αποτελέσματα των δοκιμών που πραγματοποιήθηκαν απέδειξαν την γενικότερη τάση στο πεδίο των σύγχρονων κομβίων που διατίθενται.

Η χρησιμότητα ενός τέτοιου συστήματος θεωρείται δεδομένη, για να μπορούν να αναδειχθούν πιθανές πρώιμες αστοχίες και να αποφευχθούν κατασκευαστικά λάθη των προϊόντων, όπου θα δημιουργούσαν αρνητικό αντίκτυπο στις πωλήσεις τους.

Σίγουρα βέβαια παντού μπορούν να υπάρξουν βελτιώσεις και αυτό καλούμαστε να απαντήσουμε στο παρακάτω ερώτημα :

Τι θα μπορούσε να αλλάξει για να βελτιωθεί ο υπάρχων μηχανισμός δοκιμής στο μέλλον ;

Το γεγονός ότι η κατασκευή του μηχανισμού πραγματοποιήθηκε από το μηδέν, χωρίς να υπάρχει κάποιο πρότυπο να ακολουθηθεί, έγινε μια προσπάθεια να είναι όσο πιο πλήρης και αυτόνομη γίνεται, αλλά και με ταυτόχρονο στόχο το κόστος να κρατηθεί σε λογικά για την εποχή επίπεδα.

Ως πρώτη πρόταση για την βελτίωση του συστήματος θα μπορούσε να είναι η αλλαγή των ρελέ με SSR (solid stare relays). Τα ρελέ αυτό λόγω έλλειψής κινούμενων μερών επιτυγχάνουν πολλά περισσότερα ανοιγοκλεισίματα αλλά και πιο αθόρυβα ταυτόχρονα.

Το υψηλό κόστος τους αλλά και η δυσκολία εύρεσης αντικατάστασης σε περίπτωση αστοχίας είναι δύο παράγοντες που απέτρεψαν την επιλογή τους αρχικά.

Μια δεύτερη πρόταση για την βελτίωση του συστήματος θα μπορούσε να είναι η προσθήκη ενός αεροθυλάκιου έτσι ώστε να υπάρχει ισορροπημένη και συνεχής ροή αέρα χωρίς να έχουμε την μείωση πίεσης του αέρα έως ότου ενεργοποιηθεί ο αεροσυμπιεστής.

Επιπλέον βελτιώνεται και οι ποιότητα των χτυπημάτων καθώς αυτά θα είναι πάντα με την ίδια πίεση και δεν θα απαιτείται συχνά η εποπτεία η επανάληψη βαθμονόμησης του συστήματος.

Βέβαια και εδώ το κόστος μιας τέτοιας επέμβασης είναι πολύ μεγάλο αλλά εάν το σύστημα εργάζεται συνεχώς πιθανώς σε μελλοντική εξέλιξη να συμπεριληφθεί.

Μια τρίτη πρόταση για την βελτίωση του συστήματος θα μπορούσε να είναι η προσθήκη ενός online UPS, για να διασφαλιστεί ότι σε περίπτωση διακοπής τροφοδοσίας κατά την δοκιμή θα συνεχιστεί η λειτουργία του μηχανισμού αλλά και για την απομόνωση των διάφορων παρεμβολών από το δίκτυο παροχής ηλεκτρικού ρεύματος ειδικά στο βιομηχανικό περιβάλλον. Φυσικά και σε αυτή την πρόταση το κόστος είναι ιδιαίτερα υψηλό οπότε αποτελεί μια ακόμη μελλοντική ενέργεια αναβάθμισης.

Βιβλιογραφία - Αναφορές

Vivien Sipkás, Gabriella Bognár. (2018) *The Application of Accelerated Life Testing Method for Micro Switches*. *International Journal of Instrumentation and Measurement*, **3**, 1-5
[online]

Available from : <https://www.iasas.org/iasas/home/caijim/the-application-of-accelerated-life-testing-method-for-micro-switches> (Accessed 13 June 2022)

Adrien Baima, (2013) *Push-button testing system* [online]

Available from : https://www.researchgate.net/publication/302625228_Push-button_testing_system (Accessed 13 June 2022)

Σκαρπέτης, Μ., & Κουμπουλής, Φ. (2016). *Αυτόματος Έλεγχος Υδραυλικών και Πνευματικών Συστημάτων* [online]

Available from : <https://repository.kallipos.gr/handle/11419/6262> (Accessed 9 May 2022)

Παραρτήματα :

Ευρεσιτεχνίες σχετικά με τον έλεγχο κομβίων :

<https://worldwide.espacenet.com/patent/search/family/060752948/publication/CN207114140U?q=pn%3DCN207114140U>

<https://worldwide.espacenet.com/patent/search/family/052067657/publication/CN204027814U?q=pn%3DCN204027814U>

<https://worldwide.espacenet.com/patent/search/family/050177440/publication/CN203465087U?q=pn%3DCN203465087U>

<https://worldwide.espacenet.com/patent/search/family/076901665/publication/CN213779483U?q=pn%3DCN213779483U>

<https://worldwide.espacenet.com/patent/search/family/050874814/publication/CN203643579U?q=pn%3DCN203643579U>

<https://worldwide.espacenet.com/patent/search/family/068093310/publication/CN209470840U?q=pn%3DCN209470840U>

<https://worldwide.espacenet.com/patent/search/family/054745690/publication/CN204855151U?q=button%20switch%20test%20mechanism>

<https://worldwide.espacenet.com/patent/search/family/050333530/publication/CN203502560U?q=pn%3DCN203502560U>

<https://worldwide.espacenet.com/patent/search/family/065218288/publication/CN208477080U?q=pn%3DCN208477080U>

<https://worldwide.espacenet.com/patent/search/family/061737963/publication/CN207181014U?q=pn%3DCN207181014U>

<https://worldwide.espacenet.com/patent/search/family/046599751/publication/CN102650562A?q=pn%3DCN102650562A>

Datasheet :

Διαθέσιμα εφόσον ζητηθούν