

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 501

Έμπειρος-Ασαφής έλεγχος βιολογικού αντιδραστήρα

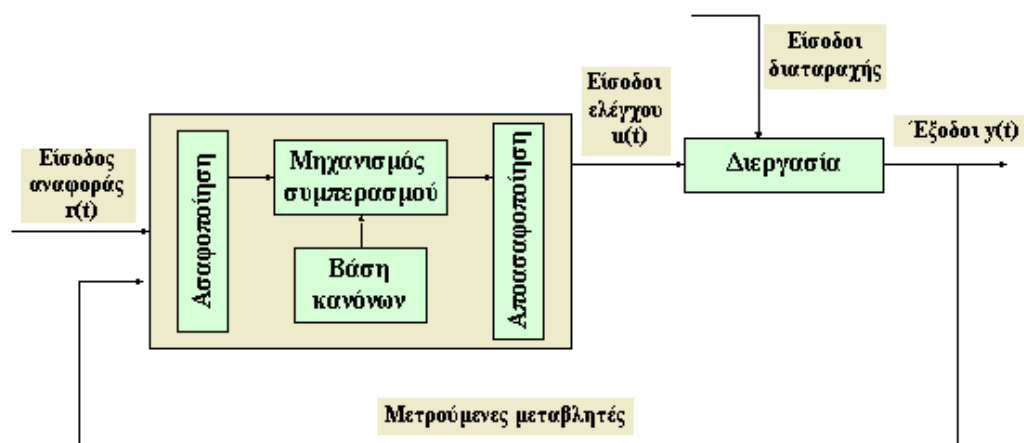
**Εργαστήριο Γενικής Ηλεκτροτεχνίας
Τμήμα Ηλεκτρολόγων Μηχανικών & Τεχνολογίας Υπολογιστών
Τομέας Συστημάτων & Αυτομάτου Ελέγχου**

Πανεπιστήμιο Πατρών , 2011

Έμπειρος-Ασαφής έλεγχος βιολογικού αντιδραστήρα

Συχνά είναι πολύ δύσκολο να μοντελοποιήσουμε τα σύνθετα συστήματα του πραγματικού κόσμου, ειδικά όταν υπάρχει ανάγκη εφαρμογής σε βιομηχανικό περιβάλλον. Ακόμη και αν μπορεί να αναπτυχθεί ένα σχετικά ακριβές μοντέλο ενός δυναμικού συστήματος, είναι συχνά πολύ σύνθετο για να χρησιμοποιηθεί στην ανάπτυξη ενός ελεγκτή. Σ' αυτές τις περιπτώσεις χρησιμοποιείται ο ασαφής έλεγχος, που παρέχει μία "επίσημη" μεθοδολογία για την αναπαράσταση, το χειρισμό και την εφαρμογή της ανθρώπινης γνώσης γύρω από τον έλεγχο ενός συστήματος.

Το σχηματικό διάγραμμα ενός ασαφούς ελεγκτή είναι το ακόλουθο:



Αρχιτεκτονική ασαφούς ελεγκτή

Όπως φαίνεται στο σχήμα, ο ασαφής ελεγκτής αποτελείται από 4 επιμέρους τμήματα:

1. Τη Βάση Κανόνων
2. Το Μηχανισμό Εξαγωγής Συμπεράσματος
3. Τη Διαδικασία Ασαφοποίησης
4. Τη Διαδικασία Από-Ασαφοποίησης

Δηλαδή, ο ασαφής ελεγκτής μοιάζει μ' έναν τεχνητό "λήπτη" αποφάσεων, που λειτουργεί σε σύστημα κλειστού βρόχου, σε πραγματικό χρόνο (real-time). Συγκεντρώνει τις εξόδους $y(t)$ και με βάση τις εισόδους αναφοράς $r(t)$ αποφασίζει ποια πρέπει να είναι η $u(t)$ ώστε να επιτευχθούν οι επιθυμητοί στόχοι. Θα μπορούσαμε ακόμη να προσθέσουμε ότι, ενώ η γλώσσα του συμβατικού ελέγχου είναι οι συνήθεις διαφορικές εξισώσεις, η γλώσσα του ασαφούς ελέγχου είναι οι κανόνες.

Ασαφής Λογική - Ασαφή Σύνολα

Η θεωρία των ασαφών συνόλων διαφέρει από την παραδοσιακή θεωρία των Boolean (δίτιμων) συνόλων ως προς το γεγονός ότι επιτρέπει τη "μερική" συμμετοχή. Η κλασική θεωρία των Boolean συνόλων είναι δίτιμη, από την άποψη ότι ένα μέλος είτε ανήκει στο σύνολο είτε όχι (δύο καταστάσεις που απεικονίζονται με 1 και 0 αντίστοιχα). Η θεωρία των ασαφών συνόλων επιτρέπει τη "εν μέρει" συμμετοχή ή ένα βαθμό συμμετοχής, που μπορεί να πάρει οποιαδήποτε τιμή μεταξύ του 0 και 1. Η ασαφής λογική είναι επομένως μία μέθοδος χειρισμού μεταβλητών οι οποίες δεν έχουν καθορισμένη αριθμητική τιμή, αλλά προσδιορίζονται με λεκτικούς όρους. Αν η ασαφής λογική συνδυασθεί με εμπειρικούς κανόνες είναι δυνατόν να μιμηθεί την ανθρώπινη συλλογιστική και να χρησιμοποιηθεί στον έλεγχο διεργασιών (ασαφή-έμπειρα συστήματα).

Ένας λεκτικός όρος μπορεί να οριστεί ποσοτικά από μιας μορφής ασαφές σύνολο που λέγεται *συνάρτηση συμμετοχής*. Η *συνάρτηση συμμετοχής* ορίζει βαθμούς συμμετοχής με βάση μια ιδιότητα, όπως η θερμοκρασία ή η πίεση. Έχοντας ορίσει τις συναρτήσεις συμμετοχής για τις εισόδους και εξόδους ελεγκτών ή έμπειρων συστημάτων, μπορούμε στη συνέχεια να ορίσουμε μία βάση κανόνων με υποθετικούς κανόνες της μορφής IF-THEN. Μία τέτοια βάση κανόνων και οι αντίστοιχες συναρτήσεις συμμετοχής χρησιμοποιούνται για την ανάλυση των εισόδων των ελεγκτών και τον καθορισμό των εξόδων τους με τη διαδικασία της εξαγωγής συμπεράσματος της ασαφούς λογικής.

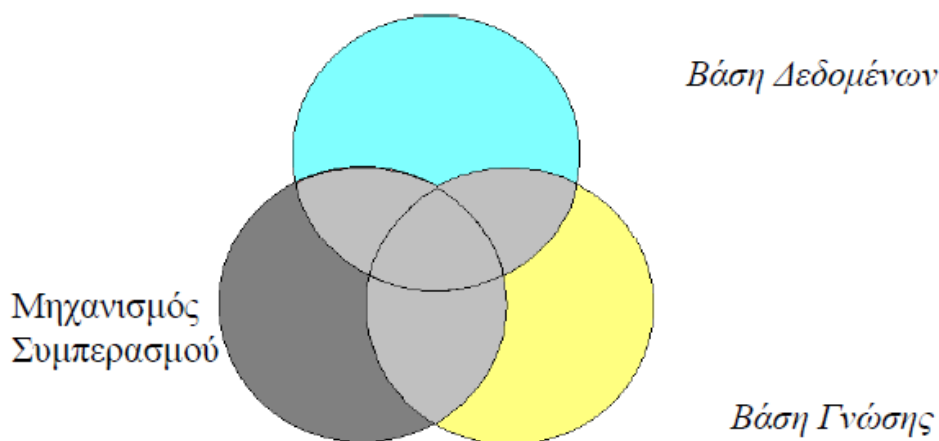
Ορίζοντας έναν τέτοιο ασαφή ελεγκτή, ο έλεγχος διεργασιών μπορεί να εκτελεστεί γρήγορα και εύκολα. Πολλά τέτοια συστήματα είναι δύσκολο ή αδύνατο να μοντελοποιηθούν μαθηματικά, κάτι το οποίο απαιτείται για το σχεδιασμό των περισσότερων παραδοσιακών αλγορίθμων ελέγχου. Επιπρόσθετα, πολλές διεργασίες που, είτε μπορούν είτε όχι να μοντελοποιηθούν μαθηματικά, είναι εξαιρετικά σύνθετες ή μη γραμμικές ώστε να ελεγχθούν με παραδοσιακές στρατηγικές. Παρόλ' αυτά, αν μια στρατηγική ελέγχου μπορεί να περιγραφεί ποιοτικά από έναν ειδικό, η ασαφής λογική μπορεί να χρησιμοποιηθεί για τον έλεγχο μιας διεργασίας που ο άνθρωπος μπορεί χειροκίνητα να ελέγξει με τις γνώσεις που έχει αποκτήσει από την πείρα του. Οι γλωσσικοί κανόνες ελέγχου που ο άνθρωπος ειδικός μπορεί να περιγράψει με διαισθητικό και γενικό τρόπο, μπορούν απ' ευθείας να μεταφραστούν σε μια βάση κανόνων για τον ελεγκτή ασαφούς λογικής.

Βασικές έννοιες Ασαφής Λογικής και Ελέγχου

Ο Ευφυής Έλεγχος (Intelligent Control) αντιμετωπίζει το πρόβλημα του ελέγχου διαδικασιών με τρόπο που διαφέρει ριζικά από τον συμβατικό έλεγχο. Ο Ευφυής Έλεγχος βασίζεται στη γνώση και την εμπειρία του ανθρώπου-χειριστή και δεν απαιτεί εξειδικευμένη γνώση της ελεγχόμενης διαδικασίας. Μπορούμε, συνεπώς, να θεωρήσουμε τον Ευφυή Έλεγχο ως μια από τις πρώτες τεχνικές ελέγχου χωρίς πρότυπο. Τα Ευφυή Συστήματα κατατάσσονται ανάλογα με την ικανότητά τους να συμπεραίνουν τη δράση ελέγχου από εμπειρικές υποδείξεις και χωρίς αναφορά σε μικροσκοπικά πρότυπα της ελεγχόμενης διαδικασίας. Αυτή είναι και η βασική διαφορά τους από τα συμβατικά συστήματα.

Η ανάπτυξη ενός Ασαφούς Ελεγκτή, που ανήκει στα Ευφυή Συστήματα, βασίζεται σε μεθόδους αναπαράστασης της γνώσης και μηχανισμούς ανεύρεσης αποφάσεων Ασαφούς Λογικής. Οι κανόνες, που είναι κατάλληλα κωδικοποιημένοι, καταχωρούνται σε μια **βάση γνώσης** και τα δεδομένα της διαδικασίας (μετρήσεις των μεταβλητών, παράμετροι, συντελεστές αισθητήρων και ενεργοποιητών της ελεγχόμενης διαδικασίας) σε μια **Βάση Δεδομένων Πραγματικού Χρόνου** (Real Time Data Base).

Ο πυρήνας ενός Ασαφούς Ελεγκτή είναι ο μηχανισμός συμπερασμού, ικανός να συμπεραίνει και να υποστηρίζει τις αποφάσεις που εξάγει.



Σχεδιάγραμμα Λειτουργίας Ασαφούς Ελεγκτή

Σε αυτό το σημείο κρίνεται απαραίτητο να κάνουμε μια περιληπτική αναφορά στις βασικές έννοιες της ασαφούς λογικής και την σύνδεση της με τον έλεγχο.

Ασαφή Σύνολα και Συνάρτηση Συμμετοχής

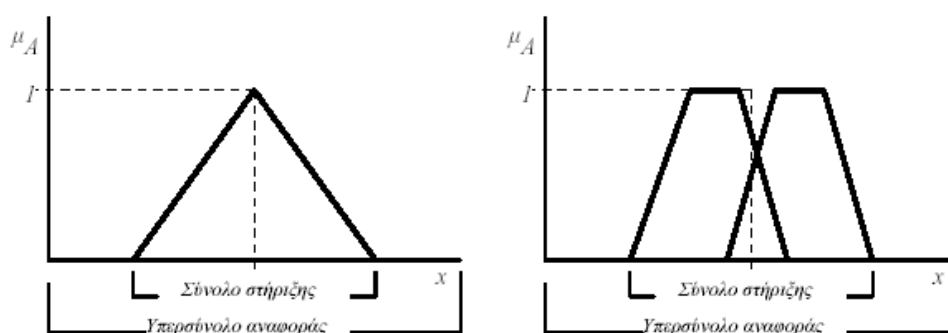
Στην κλασσική θεωρία συνόλων, κάθε στοιχείο μπορεί να ανήκει ή να μην ανήκει σε ένα σύνολο. Η συμμετοχή, λοιπόν, ενός στοιχείο είναι περιορισμένη σε δύο καταστάσεις-τιμές (ναι/όχι, 1/0, αλήθεια/ψέμα). Σύνολα που περιέχουν

στοιχεία με δυαδικές συναρτήσεις συμμετοχής τους ονομάζονται μη-ασαφή σύνολα. Τα ασαφή σύνολα, από την άλλη, επιτρέπουν στις συναρτήσεις συμμετοχής να πάρουν και ενδιάμεσες τιμές και γι αυτό θεωρείται ότι εξομοιώνουν την ανθρώπινη σχέση. Η ασαφής λογική (fuzzy logic) είναι ένα υπερόνολο της κλασσικής λογικής, η οποία έχει επεκταθεί έτσι ώστε να μπορεί να χειριστεί τιμές αληθείας μεταξύ του «απολύτως αληθούς» και του «απολύτως ψεύδους». Οι άνθρωποι χρησιμοποιούν την λογική της σταδιακής μετάβασης μεταξύ δύο καταστάσεων και όχι τόσο της δυαδικής.

Ένα αντικείμενο, λοιπόν, μπορεί να ανήκει **μερικώς** σε ένα ασαφές σύνολο. Ο βαθμός συμμετοχής στο σύνολο ονομάζεται **συνάρτηση συμμετοχής** (membership function $f(x)$) και έχουμε τις εξής περιπτώσεις:

- $f(x)=0$ το αντικείμενο δεν ανήκει στο σύνολο
- $f(x)=1$ είναι σίγουρα μέλος του συνόλου
- Οι υπόλοιπες τιμές για την $f(x)$ δείχνουν το βαθμό συμμετοχής

Η ασάφεια μπορεί να εισαχθεί στη θεωρία των συνόλων, αν γενικευτεί η χαρακτηριστική συνάρτηση για να λαμβάνει άπειρο αριθμό τιμών στο διάστημα $[0,1]$.



Παραδείγματα τριγωνικού και τραπεζοειδών ασαφών συνόλων

Ασαφές Σύνολο (fuzzy set) A είναι ένα σύνολο διατεταγμένων ζευγών $(x, \mu_A(x))$ όπου x ανήκει στο X και $\mu_A(x)$ ανήκει στο $[0,1]$.

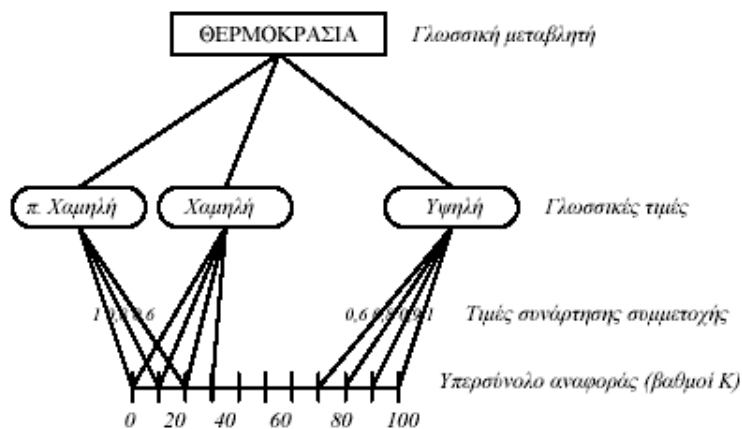
- Το σύνολο X αποτελεί ένα ευρύτερο σύνολο αναφοράς που περιλαμβάνει όλα τα αντικείμενα στα οποία μπορεί να γίνει αναφορά.
- Η τιμή $\mu_A(x)$ λέγεται βαθμός αληθείας (degree of truth), συμβολίζει το βαθμό της συγγένειας του x στο A και παίρνει τιμές στο διάστημα $[0,1]$.
- Η συνάρτηση $\mu_A(x)$ ονομάζεται συνάρτηση συμμετοχής ή συγγένειας (membership function) και στην πράξη μπορεί να προέρχεται από:

- a. Υποκειμενικές εκτιμήσεις
- b. Προκαθορισμένες και απλοποιημένες μορφές
- c. Συχνότητες εμφανίσεων και πιθανότητες
- d. Φυσικές μετρήσεις
- e. Διαδικασίες μάθησης και προσαρμογής (συνήθως με νευρωνικά δίκτυα)

Η ασαφής θεωρία συνόλων μεταπίπτει στην αντίστοιχη κλασσική, όταν οι δυνατές τιμές της συνάρτησης συμμετοχής είναι μόνο 0 ή 1.

Λεκτικές Μεταβλητές

Οι τιμές μιας ασαφούς μεταβλητής (fuzzy variable) μπορούν να θεωρηθούν ετικέτες (labels) ασαφών συνόλων. Έτσι, για παράδειγμα, η ΘΕΡΜΟΚΡΑΣΙΑ σε κάποιο σημείο μιας διαδικασίας μπορεί να θεωρηθεί ως ασαφής μεταβλητή που μπορεί να πάρει λεκτικές τιμές Κανονική, Χαμηλή, Μέση, Υψηλή και πολύ_Υψηλή. Τα ασαφή σύνολα και οι λεκτικές μεταβλητές χρησιμοποιούνται για να εξηγήσουν ποσοτικά έννοιες της φυσικής γλώσσας, τα οποία μπορούμε στη συνέχεια να χειριστούμε. Οι λεκτικές μεταβλητές χρησιμοποιούνται στους ευρεστικούς κανόνες. Οι τιμές της μεταβλητής ΘΕΡΜΟΚΡΑΣΙΑ του παραδείγματος μπορούν έτσι να εκφραστούν ως *Υψηλή*, *όχι_Υψηλή*, *σχετικά_Υψηλή*, *όχι_πολύ_Υψηλή*, *πάρα_πολύ_Υψηλή*, *αρκετά_Υψηλή* κ.ά. δηλαδή με προτάσεις αποτελούμενες από την ετικέτα *Υψηλή*, την άρνηση *όχι*, τα συνδυαστικά και άλλα καθώς και τα περιγράμματα *πολύ*, *σχετικά*, *αρκετά* κ.ά. Κατά τον τρόπο αυτό η μεταβλητή ΘΕΡΜΟΚΡΑΣΙΑ είναι μια **λεκτική μεταβλητή**.



Σχεδιάγραμμα Λεκτικής Μεταβλητής-Θερμοκρασία

Μηχανισμός Συμπερασμού και Λεκτικοί Κανόνες

Κάθε ασαφής ελεγκτής αποτελείται από το *μηχανισμό συμπερασμού ή συμπερασμάτων (inference engine)* με τον οποίο υπολογίζονται οι έξοδοι του ελεγκτή. Ο μηχανισμός συμπερασμού ουσιαστικά μας παρέχει την ικανότητα συλλογισμού. Αντίθετα με τα συμβατικά έμπειρα συστήματα, στο μηχανισμό συμπερασμού ενός ασαφούς ελεγκτή διεξάγεται εξονυχιστική ανίχνευση όλων των κανόνων στη βάση γνώσης με σκοπό να υπολογιστεί ο βαθμό συμμετοχής ή εκπλήρωσης κάθε κανόνα. Ο έλεγχος διαδικασιών με ασαφείς ελεγκτές προϋποθέτει την ύπαρξη ενός **σύνολου λεκτικών κανόνων** που περιγράφει τις πράξεις ενός εμπειρογνώμονα χειριστή. Οι κανόνες αυτοί είναι όμοιοι με αυτούς που εκπαιδεύονται οι χειριστές και στη συνέχεια εφαρμόζουν στην πράξη. Το σύνολο των λεκτικών κανόνων αποθηκεύεται στη *βάση γνώσης (rule base)* του ελεγκτή. Φυσικό είναι να μην είναι γνωστοί όλοι οι κανόνες που είναι απαραίτητοι να αντιμετωπίσουν όλες τις καταστάσεις της ελεγχόμενης διαδικασίας. Έτσι η γνώση για τον έλεγχο μιας διαδικασίας συνήθως καθορίζεται από ένα σύνολο κανόνων της μορφής 'ΑΝ (αίτιο) ΤΟΤΕ (συμπέρασμα)' ή $AN\ A\ TOTE\ B$ ο οποίος ορίζεται ως τον κλασικός προτασιακός λογισμός (*propositional calculus*) που μπορούμε να εκφράσουμε συμβολικά ως: $A \rightarrow B$

Η συνηθέστερη μορφή είναι $AN...και...και...TOTE....$ δηλαδή για την εκπλήρωση του AN χρειάζονται περισσότερα από ένα αίτια.

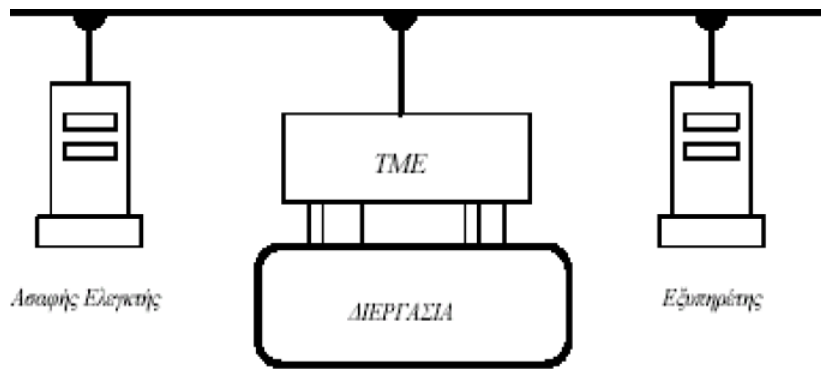
Έτσι η ασαφής λογική χρησιμοποιείται σε πολύπλοκες βιομηχανικές διαδικασίες όπου οι χειριστές του συστήματος κάνουν ενέργειες ελέγχου με βάση την δική τους εμπειρία και χρησιμοποιώντας τους δικούς τους εμπειρικούς κανόνες σε μορφή Αν/Τότε (π.χ. Αν η θερμοκρασία είναι ψήλη και η πίεση είναι μέτρια, Τότε πρέπει να ανοίξω την βαλβίδα στο μισό).

Στην ασαφή λογική υπάρχουν δύο κύριες κατηγορίες κανόνων συνεπαγωγής. Η πρώτη είναι η κατηγορία *Generalised Modus Ponens* (ή *GMP*) που συσχετίζεται άμεσα με τους μηχανισμούς πρόσθιας συνεπαγωγής με δεδομένα (*forward data-driven inference*), που βρίσκει εφαρμογή σε όλους τους ασαφείς ελεγκτές. Η δεύτερη είναι η κατηγορία *Generalised Modus Tollens* (ή *GMT*) που σχετίζεται στενά με τους μηχανισμούς οπισθόδρομου συμπερασμού με στόχο (*backwards goal-driven*) που συνηθίζεται στα έμπειρα συστήματα. Αντίθετα με τον κανόνα *GMP*, ο σκοπός του μηχανισμού συμπερασμού εδώ είναι η ανεύρεση των αιτιών που έχουν ένα δεδομένο αποτέλεσμα. Οι κυριότερες συνεπαγωγές είναι η Boole, η Lukasiewicz, η Zadeh, η Mamdani και η Larsen, όπου οι δύο τελευταίες είναι οι επικρατέστερες πλέον στα ασαφή συστήματα και χρησιμοποιούνται για τον υπολογισμό των εξόδων του ασαφούς συστήματος. Πιο συγκεκριμένα έχουμε τους κανόνες: 1) μέγιστου-ελάχιστου (max-min) Mamdani και 2) μέγιστου γινομένου (max-product) Larsen οι οποίοι χρησιμοποιούν κάποιες απλές

συναρτήσεις εύρεσης μέγιστου(max) και ελαχίστου(min) μεταξύ δύο ή και περισσότερων αριθμών. Οι δύο αυτοί κανόνες και κυρίως ο Larsen(αυτόν χρησιμοποιεί και το λογισμικό μας FUZZY CONTROL ++V6.0) καθώς και κάποιες απλές πράξεις πινάκων συνθέτουν την λειτουργία του μηχανισμού συμπερασμού και την ασαφοποίηση των εισόδων που περιλαμβάνει πράξεις μεταξύ των τιμών των συναρτήσεων συμμετοχής τους. Δεν θα αναλύσουμε περισσότερο την διαδικασία αυτή γιατί είναι μεν απλή στην λογική αλλά απαιτεί μεγάλο αριθμό αριθμητικών υπολογισμών και πράξεις πινάκων, ωστόσο αν κάποιος ενδιαφέρεται μπορεί να την βρει εύκολα στις περισσότερες βιβλιογραφίες που σχετίζονται με την ασαφή λογική.

Θα αναφερθούμε ακόμα στην από-ασαφοποίηση, δηλαδή στον υπολογισμό των εξόδων. Η από-ασαφοποίηση λαμβάνει χώρα αφού πρώτα έχει γίνει η ασαφοποίηση των εισόδων και έχει λειτουργήσει ο μηχανισμός συμπερασμού, δηλαδή έχει ενεργοποιηθεί ένας ή και περισσότεροι κανόνες AN...και...και...ΤΟΤΕ... και έχει πάρει μια απόφαση για κάποια έξοδο, αφορά δηλαδή το κομμάτι ΤΟΤΕ.. του λεκτικού κανόνα και στον προσδιορισμό της ακριβής τιμής της εξόδου. Υπάρχουν τέσσερις διαδεδομένες τεχνικές από-ασαφοποίησης και είναι οι εξής: α) αποασαφοποίηση μέγιστου (Maximum defuzzifier), β) αποασαφοποίηση μέσου όρου των μέγιστων (Mean of Maxima defuzzifier) - MOM , γ) αποασαφοποίηση κεντρώου (Centre of Area - defuzzifier) και δ) αποασαφοποίηση κέντρου βάρους (Centre of Gravity - COG). Θα σταθούμε στην τελευταία τεχνική, από-ασαφοποίηση κέντρου βάρους καθώς αυτήν χρησιμοποιεί το λογισμικό μας FUZZY CONTROL++ V6.0

Στις σύγχρονες βιομηχανίες διεργασιών η διασύνδεση της φυσικής διαδικασίας με τον ευφυή ελεγκτή συνήθως γίνεται διαμέσου **Τοπικών Μονάδων Ελέγχου (TME)** στο ήδη υπάρχον Σύστημα Εποπτικού Ελέγχου και Συλλογής Πληροφοριών. Οι TME περιλαμβάνουν υλικό για την αναλογική προς ψηφιακή (Α/Ψ) μετατροπή των διαφόρων σημάτων της διαδικασίας με σάρωση χιλιάδων σημάτων ανά δευτερόλεπτο, καθώς και λογισμικό για την οδήγηση των ενεργοποιητών. Οι TME επίσης κατέχουν οδηγό (*driver*) για τη μετάδοση των σημάτων προς και από τον εξυπηρετητή (*server*) μέσω τοπικού δικτύου, ενσύρματης ή ασύρματης επικοινωνίας.



Αρχιτεκτονική Ασαφούς Ελεγκτή.

Η πιο συνήθης μορφή των TME είναι οι ευρέως διαδεδομένοι προγραμματιζόμενοι ελεγκτές (PLC), που εκτός από τις διαδικασίες εκκίνησης, μανδάλωσης και προστασίας του εξοπλισμού, έχουν την ικανότητα συλλογής, μετάδοσης αναλογικών και ψηφιακών σημάτων καθώς και υπολογισμών μεγάλου πλήθους αλγορίθμων ελέγχου.

Ασαφοποίηση των δεδομένων

Γνωρίζοντας τις μεταβλητές εισόδου και εξόδου, μπορούμε να κάνουμε ασαφοποίηση αυτών των δεδομένων. Γι' αυτό το σκοπό, απαιτείται για κάθε μεταβλητή ο καθορισμός των ακόλουθων στοιχείων:

- ενός υπερσυνόλου αναφοράς, δηλαδή ενός συνόλου ορισμού αυτών των μεταβλητών, το οποίο θα λαμβάνει μία ελάχιστη και μία μέγιστη τιμή
- των συναρτήσεων συμμετοχής, ως προς το πλήθος, το σχήμα, το διάστημα και το γλωσσικό χαρακτηρισμό τους.



Επειδή δεν είναι δυνατή η ύπαρξη Συστήματος Βιολογικού Καθαρισμού σ' ένα εργαστήριο και προκειμένου το PLC να λειτουργεί με πραγματικές εισόδους - εξόδους, οι τελευταίες εξομοιώνονται με τη βοήθεια του πάνελ χειρισμού - ρύθμισης που φαίνεται στη φωτογραφία. Πιο συγκεκριμένα, με τη βοήθεια του ταμπλό αυτού μπορούμε να αλλάζουμε την τιμή οποιασδήποτε εισόδου και να παρακολουθούμε την αντίδραση του ελεγκτή, όπως θα κάναμε αν οι μεταβολές των εισόδων ήταν πραγματικές.

Είσοδοι και Έξοδοι του συστήματος του Βιολογικού Αντιδραστήρα

Ο βιολογικός αντιδραστήρας έχει έξι εισόδους και τρεις εξόδους, τις οποίες έπρεπε να ελέγξουμε, όπως ορίζονται στους πίνακες 1 και 2.

ΤΑΞΗ ΕΙΣΟΔΟΥ	ΣΥΜΒΟΛΙΣΜΟΣ	ΠΕΡΙΓΡΑΦΗ	ΜΟΝΑΔΑ ΜΕΤΡΗΣΗΣ
1 ^η	D(BOD)	Αντιμετώπιση οργανικού φορτίου(BOD _{ΕΙΣΟΔΟΥ} -BOD _{ΕΞΟΔΟΥ})	mg/l
2 ^η	MLSS	Συγκέντρωση των αιωρούμενων στερεών στο ανάμεικτο υγρό	mg/l
3 ^η	TEMP	Θερμοκρασία στο εσωτερικό της τάφρου	°C
4 ^η	DO	Συγκέντρωση του διαλυμένου οξυγόνου στην αερόβια ζώνη	mg/l
5 ^η	N-NH ₃	Συγκέντρωση της αμμωνίας στην έξοδο της αερόβιας ζώνης	mg/l
6 ^η	N-NO ₃	Συγκέντρωση των νιτρικών στην έξοδο της αερόβιας ζώνης	mg/l

Πίνακας 1 Καθορισμός των εισόδων του ασαφή ελεγκτή

ΤΑΞΗ ΕΞΟΔΟΥ	ΣΥΜΒΟΛΙΣΜΟΣ	ΠΕΡΙΓΡΑΦΗ
1 ^η	PROxygen	Παρεχόμενο οξυγόνο στην αερόβια ζώνη
2 ^η	Rsludge	Ανακυκλοφορία λάσπης

3 ^η	Rm1	Ανακυκλοφορία ανάμεικτου υγρού
----------------	-----	--------------------------------------

Πίνακας 2 Καθορισμός των εξόδων του ασαφή ελεγκτή

Είσοδοι συστήματος

Όπως ξέρουμε το σύστημα μας έχει έξι εισόδους και τρεις εξόδους. Οι έξι μεταβλητές των εισόδων σε ένα πραγματικό βιολογικό αντιδραστήρα είναι έξοδοι κάποιων αισθητήρων οι οποίοι έχουν τοποθετηθεί μέσα στην διαδικασία. Συνεπώς οι έξοδοι αυτών των αισθητήρων θα είναι αναλογικά ηλεκτρικά σήματα είτε τάσης, είτε ρεύματος. Οι περιοχές τιμών αυτών των αναλογικών τάσεων ή ρευμάτων που εξέρχονται από τους αισθητήρες είναι τυποποιημένες. Έτσι, για ρεύματα έχουμε μια περιοχή τιμών από 4-20 mA ενώ για τις τάσεις έχουμε , 0..10V, -10..10V, -1..1V, -50mV..50mV, -500mV..500mV κ.α. Οποιαδήποτε λοιπόν τιμή της φυσικής ποσότητας π.χ της θερμοκρασίας σε °C, αντιστοιχεί σε κάποια τιμή τάσης ή ρεύματος. Αυτά τα αναλογικά σήματα των αισθητήρων οδηγούνται σε μια αναλογική κάρτα εισόδου του PLC. Αυτό που αξίζει να σημειώσουμε στην παρούσα φάση είναι ότι κάθε μεταβλητή έχει ένα υπερσύνολο αναφοράς, π.χ. για την θερμοκρασία-μεταβλητή TEMP ορίζουμε ένα υπερσύνολο αναφοράς 10...30°C. Ο αισθητήρας όμως, δίνει μια τάση ως έξοδο η οποία αντιστοιχεί σε μια τιμή της φυσικής μεταβλητής π.χ. 12°C για -60mV. Άρα το υπερσύνολο αναφοράς που έχουμε ορίσει για την φυσική μεταβλητή πρέπει να αντιστοιχεί στο σύνολο τάσεων που δίνει ως έξοδο ο αισθητήρας. Δηλαδή το υπερσύνολο 10...30°C αντιστοιχεί στα -50mV...+50mV που δίνει ο αισθητήρας ως έξοδο.

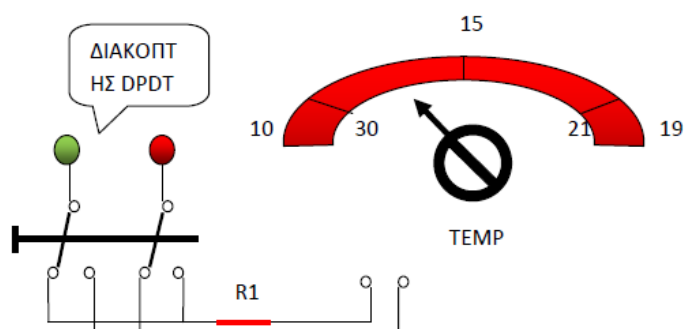
Στο εργαστήριο έχει κατασκευαστεί μια διάταξη η οποία αναπαριστά την λειτουργία των αισθητήρων. Αυτή η διάταξη είναι μια συνδεσμολογία η οποία περιλαμβάνει το ταμπλό εισόδων-εξόδων και ένα τροφοδοτικό. Αναλυτικότερα η διάταξη αυτή μας δίνει για τις πρώτες τέσσερις μεταβλητές (D(BOD), MLSS, TEMP, DO) τάσεις -50mV..+50mV ενώ για τις δύο τελευταίες -1V..+1V. Η μεταβολή του σήματος εισόδου τόσο από τα -50mV..+50mV όσο και από τα -1V..+1V επιτυγχάνεται με ροοστάτες που βρίσκονται πάνω στο ταμπλό. Έτσι, με αυτό τον τρόπο ο χρήστης, ρυθμίζοντας έξι ροοστάτες στους οποίους προσαρμόσαμε κλίμακες ένδειξης των φυσικών ποσοτήτων των μεταβλητών, αναλαμβάνει τον ρόλο του συστήματος και μπορεί να εισάγει στον ασαφή ελεγκτή οποιαδήποτε συνδυασμό των εισόδων. Φυσικά δεν επιτυγχάνουμε μια δυναμική συμπεριφορά αφού οι μεταβλητές εισόδου εξαρτώνται η μια από την άλλη, αλλά με αυτόν τον τρόπο μπορούμε να έχουμε ένα πιλοτικό οδηγό ο οποίος μας βοηθάει να εκμάθουμε σε σύντομο χρονικό διάστημα τις ενέργειες

ελέγχου που πρέπει να γίνονται όταν το σύστημα εισέλθει σε μια συγκεκριμένη κατάσταση.

Διακόπτης DPDT

Όπως παρατηρούμε έχουμε μια αλλαγή του προσήμου της τάσης. Για να είναι συνεχής η μεταβολή της τάσης το ταμπλό είναι εφοδιασμένο με διακόπτη DPDT με την βοήθεια του οποίου μπορούμε να έχουμε μια αλλαγή του προσήμου της τάσης που εφαρμόζεται στο ροοστάτη. Αρκεί να περιγράψουμε μια είσοδο π.χ. την θερμοκρασία TEMP για να κατανοήσουμε τον τρόπο λειτουργίας του διακόπτη καθώς και το πώς επιτυγχάνεται η συνεχής μεταβολή. Ας το δούμε πιο αναλυτικά:

Όπως γνωρίζουμε το υπερσύνολο αναφοράς της TEMP είναι 10..30°C. Έτσι σύμφωνα με το πεδίο μεταβολής της τάσης έχουμε 10°C=-50mV ενώ 30°C=+50mV. Συνεπώς η τιμή τάσης ίση με 0mV θα αντιστοιχεί σε 20°C στην φυσική ποσότητα. Άρα το σύνολο 10..20°C αντιστοιχεί σε μια μεταβολή της τάσης από -50mV..0mV ενώ το σύνολο 20..30°C αντιστοιχεί σε μεταβολή της τάσης από 0mV...+50mV. Μια πλήρης περιστροφή του ροοστάτη μπορεί να υλοποιήσει ένα από τα δύο αυτά πεδία τιμών της τάσης. Για να υλοποιηθεί το δεύτερο θα πρέπει να έχουμε μια αλλαγή προσήμου της τάσης. Αυτή την επιτυγχάνουμε με τον διακόπτη DPDT(δύο πόλων δύο δρόμων). Ο τρόπος δράσης του διακόπτη καθώς και η σύνδεση του στο ροοστάτη φαίνεται στο σχήμα παρακάτω.



Κύκλωμα υλοποίησης της μεταβλητής TEMP

Όπως βλέπουμε όταν ο διακόπτης είναι στην πάνω θέση, όπως στο σχήμα, τότε παίρνουμε το σύνολο 0...50mV περιστρέφοντας τον ροοστάτη από την θέση τέρμα δεξιά μέχρι τέρμα αριστερά, αυτή η μεταβολή αντιστοιχεί σε 21..30°C της θερμοκρασίας. Αν γυρίσουμε τον διακόπτη DPDT στην κάτω θέση και ενώ ο ροοστάτης βρίσκεται στην θέση τέρμα αριστερά, έχουμε αρνητικό πρόσημο της τάσης και παίρνουμε το σύνολο -50mV..0V και περιστρέφοντας τον διακόπτη από

τα αριστερά προς τα δεξιά η θερμοκρασία μεταβάλλεται 10..19°C. Πρέπει να αναφέρουμε ότι η κλίμακα για κάθε θέση του ροοστάτη είναι γραμμική επειδή ο ροοστάτης είναι γραμμικός ενώ η ένδειξη μας παρέχει την τιμή της φυσικής ποσότητας. Έπειτα, η τάση τροφοδοσίας για όλες τις εισόδους είναι 2.3V και παρέχεται μέσω τροφοδοτικού ρυθμισμένο σε αυτήν την τάση. Από τα 2.3V η τάση φτάνει στα -50mV..+50mV και στα -1V..+1V με την σύνδεση αντιστάσεων σε σειρά όπως φαίνεται και στο σχήμα. Δεδομένου ότι η συνολική αντίσταση κάθε ροοστάτη είναι 1KΩ υπάρχει μια σε σειρά αντίσταση $R1=39K\Omega$ για καθεμία από τις πρώτες τέσσερις εισόδους των $\pm 50mV$ και για τις άλλες δύο των $\pm 1V$ μια αντίσταση $R2=1.5V$.

Εξοδοι συστήματος

Για τις τρεις εξόδους τώρα ακολουθεί η ανάποδη διαδικασία. Ο ελεγκτής (PLC) υπολογίζει την τιμή της κάθε εξόδου με βάση τις εισόδους και με τρόπο που θα αναλύσουμε στην συνέχεια. Όπως ξέρουμε η ενεργοποίηση και ρύθμιση ορισμένων μεταβλητών μιας διεργασίας γίνεται με τους ενεργοποιητές. Με τους ενεργοποιητές παρέχουμε κάποιες διορθωτικές τιμές σε ορισμένες μεταβλητές της διεργασίας που θέλουμε να ελέγξουμε. Η έξοδος αυτών των ενεργοποιητών μπορεί να είναι είτε το κλείσιμο μιας βαλβίδας είτε και ένα ηλεκτρικό σήμα που θα προκαλέσει μια μηχανική κίνηση. Η είσοδος τους όμως όταν αυτοί είναι συνδεδεμένοι σε PLC είναι ένα ηλεκτρικό σήμα τάσης ή ρεύματος. Συνεπώς η έξοδος ενός ελεγκτή θα είναι σήμα τάσης ή ρεύματος το οποίο στην συνέχεια θα οδηγήσει κάποιον ενεργοποιητή στην υπό έλεγχο διεργασία.

Όπως και για τις εισόδους, έτσι και για τις εξόδους ορίζουμε το υπερσύνολο αναφοράς τους 0...100% και για την υλοποίηση τους το πεδίο της τάσης εξόδου από την αναλογική κάρτα εξόδου(θα αναφερθούμε αναλυτικότερα σ' αυτήν στην συνέχεια) του PLC είναι 0..10V. Για την παρατήρηση των εξόδων, στο ταμπλό έχουν προστεθεί τρία βολτόμετρα των οποίων η κλίμακα ένδειξης των τιμών τους βαθμονομήθηκε είτε ως ποσοστό παροχής οξυγόνου είτε ως ποσοστό ανακυκλοφορίας λάσπης είτε ως ποσοστό ανάμεικτου υγρού για κάθε μια έξοδο αντίστοιχα. Κατά αυτό τον τρόπο ο χρήστης μπορεί να παρατηρήσει απευθείας και με φυσικό τρόπο τις ενέργειες ελέγχου που συμπεραίνονται από τον ελεγκτή. Στην εικόνα παρακάτω παρουσιάζεται το ταμπλό εξομοίωσης εισόδων-εξόδων.



Ταμπλό εξομοίωσης

A. Εκτέλεση και πειραματισμός με Ασαφή Ελεγκτή μέσω PLC

1. Επιλέγουμε το "ΠΡΟΓΡΑΜΜΑ ΑΣΑΦΟΥΣ FUZZY CONTROL V6".
2. Επιλέγουμε "Targetsystem", "Selection" και πατάμε OK στο εικονίδιο που εμφανίζεται (S7-4K).
3. Πατάμε x στο εικονίδιο που εμφανίζεται.
4. Επιλέγουμε "Connect" και εμφανίζονται τα στοιχεία του targetsystem.
ΠΡΟΣΟΧΗ δεν αλλάζουμε τίποτα και πατάμε OK.
5. Εμφανίζεται μήνυμα "A Connection with the DB "DB30" has been successfully made" Πατάμε OK.
6. Για απεικόνιση των εισόδων και εξόδων επιλέγουμε Test→Curver Plotter και Διαλέγουμε "Start Recording" (πράσινο βελάκι).
ΠΡΟΣΟΧΗ→Να μην γίνει η επιλογή ΠΟΤΕ του 3-D Graphic Representation.
7. Για την παρατήρηση των κανόνων ενεργοποίησης επιλέγουμε: Test→Rule Activity (αφού έχει ενεργοποιηθεί το Curve plotter. Μπορείτε πια να παρακολουθήσετε την ενεργοποίηση των κανόνων για κάθε έξοδο.

Μετρήσεις:

Εφαρμόζουμε στο πάνελ χειρισμού τις παρακάτω εισόδους και μετράμε τις αντίστοιχες εξόδους συμπληρώνοντας τους πίνακες των ερωτημάτων. Επίσης σημειώνουμε τους αριθμούς των κανόνων που ικανοποιούνται σε κάθε περίπτωση.

Σημείωση: Ο διακόπτης κάθε εισόδου στην αρχή πρέπει να είναι στην κόκκινη θέση (όλοι δεξιά). Οι τιμές των εισόδων και των εξόδων μετρώνται στην οθόνη εποπτείας του υπολογιστή και όχι στις ενδείξεις του πάνελ χειρισμού.

1.

	BO D	MLS S	TEM P	D O	N_N H₃	N_N O₃	Κανόνε ς	Q1 Prov_o x	Q2 R_sludg e	Q3 R_m l
	25	3500	20	1.7 5	250	1.5				

Ποια η τιμή του κανόνα που ικανοποιείται για κάθε έξοδο? Ποια είναι η τιμή της κάθε εξόδου? Τι παρατηρείτε? Εξηγήστε.

2.

	BO D	MLS S	TEM P	D O	N_N H₃	N_N O₃	Κανόνε ς	Q1 Prov_o x	Q2 R_sludg e	Q3 R_m l
	45	3500	20	1.7 5	250	1.5				

Ποια η τιμή του κανόνα που ικανοποιείται για κάθε έξοδο? Ποια είναι η τιμή της κάθε εξόδου? Τι παρατηρείτε? Εξηγήστε.

3.

	BO D	MLS S	TEM P	D O	N_N H₃	N_N O₃	Κανόνε ς	Q1 Prov_o x	Q2 R_sludg e	Q3 R_m l
	35	3500	20	1.7 5	250	1.5				

Σε ποιο σημείο ικανοποιείται ο κάθε κανόνας και τι συμπεραίνετε συγκρίνοντας με τις τιμές των εξόδων? Πόσο είναι το άθροισμα των % των κανόνων για κάθε έξοδο?

4.

	BO D	MLS S	TEM P	D O	N_N H₃	N_N O₃	Κανόνε ς	Q1 Prov_o x	Q2 R_sludg e	Q3 R_m l
	25	3500	20	1.7 5	250	2.158				

Πόσο είναι το άθροισμα των % των κανόνων για κάθε έξοδο? Παρατηρήστε την έξοδο R_ml, τι τιμή έχει ο κάθε κανόνας ξεχωριστά και τελικά ποια είναι η τιμή της εξόδου? Είναι λογικό? Εξηγήστε.

5.

	BO D	MLS S	TEM P	D O	N_N H₃	N_N O₃	Κανόνε ς	Q1 Prov_o x	Q2 R_sludg e	Q3 R_m l
	38	3500	21.15	1.75	313.72	1.84				

Σχολιάστε την τιμή των εξόδων σε κάθε περίπτωση. (Παρατηρήστε επίσης ότι το άθροισμα των ποσοστών είναι >100%).

6.

	BO D	MLS S	TEM P	D O	N_N H₃	N_N O₃	Κανόνε ς	Q1 Prov_o x	Q2 R_sludg e	Q3 R_m l
	55	5870	31	0.33	525	3.15				

Πόσοι κανόνες ικανοποιούνται? Είναι λογικό αυτό? Εξηγήστε.

7. Βρείτε συνδυασμούς των εισόδων έτσι ώστε να ικανοποιούνται 3 κανόνες και καταγράψτε τις τιμές των εξόδων καθώς και τον αριθμό των κανόνων που ικανοποιούνται.

	BO D	MLS S	TEM P	D O	N_N H₃	N_N O₃	Κανόνε ς	Q1 Prov_o x	Q2 R_sludg e	Q3 R_m l

Είναι δυνατόν για μια έξοδο να ικανοποιούνται π.χ 4 κανόνες και για άλλη έξοδο 3 κανόνες? Εξηγήστε.

8. Ορίστηκαν οι παρακάτω εισοδοι και έξοδοι για εποπτεία του ελεγκτή

BOD	TEMP	N_NO₃	R_sludge	R_ml
------------	-------------	-------------------------	-----------------	-------------

Μεταβάλλετε τις εισόδους (3 πρώτες μεταβλητές του πίνακα) και παρατηρήστε γραφικά τις μεταβολές των εξόδων. Σχολιάστε. (Αν επιθυμείτε κρατήστε την γραφική για την αναφορά σας). Παρατηρήστε την αλλαγή στους κανόνες που ικανοποιούνται κάθε φορά κατά την μεταβολή των εισόδων.

B. Υλοποίηση Ασαφούς Ελεγκτή σε περιβάλλον Fuzzy Control

Να υλοποιήσετε έναν ελεγκτή, δίνοντάς του το όνομα της ομάδας σας, για μια φυσική διεργασία της επιλογής σας. Καθορίστε τις εισόδους και τις εξόδους με βάση τα ακόλουθα χαρακτηριστικά.

- Τρεις Εισόδους και μια Έξοδο (Ορίστε ονόματα που υποδεικνύουν το εκάστοτε μέγεθος)
- Τρεις συναρτήσεις συμμετοχής για κάθε μεταβλητή εισόδου- εξόδου.
- Ικανοποιητικό αριθμό κανόνων με σκοπό την επίτευξη του εκάστοτε επιθυμητού στόχου που εσείς έχετε θέσει.
-

Οδηγίες διεξαγωγής της άσκησης

1. Επιλέγουμε "File → New " δίνοντας όνομα και αριθμό εισόδων και εξόδων του ελεγκτή που θα δημιουργήσουμε.
2. Ορισμός των ασαφών συνόλων των μεταβλητών εισόδου.
 - a. Επιλέγουμε την πρώτη είσοδο για να εμφανιστεί η οθόνη με τις ιδιότητες της.
 - b. Επιλέγουμε "Add" για τον ορισμό του αριθμού των συναρτήσεων συμμετοχής της επιλεγμένης μεταβλητής εισόδου.
 - c. Καθορίζουμε τα όρια του ασαφούς συνόλου και των συναρτήσεων συμμετοχής.
3. Εργαζόμαστε ανάλογα για τον ορισμό των ασαφών συνόλων των μεταβλητών εξόδου.
4. Επιλέγουμε "Add " για την προσθήκη νέων κανόνων.
5. Ορίζουμε σήματα εισόδου για την εξομοίωση του συστήματός μας επιλέγοντας "Test → Curve Plotter".
6. Επιλέγοντας "Go" εμφανίζεται η οθόνη εποπτείας του ασαφούς ελεγκτή μας.
7. Για την εποπτεία των κανόνων που ενεργοποιούνται επιλέγουμε "Test → Rule Activity".
8. Αποθηκεύουμε τον ελεγκτή μας επιλέγοντας "File → Save".

Ζητήματα

- Περιγράψτε την φυσική διεργασία που σκεφτήκατε και σχολιάστε την ικανοποιητική ή όχι λειτουργία του.
- Παρατηρήσατε ότι οι δυνατές συναρτήσεις συμμετοχής για την έξοδο είναι συναρτήσεις γραμμής (κρουστικές). Αυτό συνεπάγεται ότι η έξοδος

παίρνει αποκλειστικά τις τιμές αυτές? Απαντήστε βασισμένη στο γράφημα που λάβατε στο ερώτημα 5) και στην συνέχεια εξηγήστε γιατί συμβαίνει αυτό? Πότε είναι δυνατόν η έξοδος να παίρνει μόνο αυτές τις τρεις τιμές.

Υλοποίηση Ασαφούς Ελεγκτή για τον έλεγχο θερμοκρασίας δωματίου σε περιβάλλον MATLAB.

Να υλοποιήσετε ένα ασαφή ελεγκτή για τον έλεγχο της θερμοκρασίας ενός δωματίου με χρήση της εργαλειοθήκης Fuzzy του λογισμικού Matlab (στο Command Window πληκτρολογήστε `>>fuzzy`). Σκοπός του ελεγκτή είναι ο έλεγχος του κλιματιστικού του δωματίου καθώς και ενός παραθύρου με στόχο την επίτευξη της επιθυμητής θερμοκρασίας των 25°.

Οι μεταβλητές και παράμετροι του συστήματος είναι:

- Θερμοκρασία δωματίου: $th \in [0,50]$
- Επιθυμητή θερμοκρασία: $thd = 25^\circ$
- Εξωτερική θερμοκρασία: $thex \in [0,50]$
- Κλιματισμός: $kl \in [-1,1]$ (οι αρνητικές τιμές υποδηλώνουν ψύξη, ενώ οι θετικές θέρμανση).
- Παράθυρο: $w \in [0,1]$ (0: κλειστό, 1: τελείως ανοιχτό).

Ο ελεγκτής πρέπει να έχει δύο εισόδους με 3-5 συναρτήσεις συμμετοχής η κάθε μία:

- Είσοδος 1: $(th - thd) \in [-25,25]$, διαφορά θερμοκρασίας από επιθυμητή.
- Είσοδος 2: $(thex - th) \in [-50,50]$, διαφορά εξωτερικής θερμοκρασίας από εσωτερική.

και δύο εξόδους με 3-5 συναρτήσεις συμμετοχής η κάθε μία:

- Έξοδος 1: Κλιματισμός: $kl \in [-1,1]$
- Έξοδος 2: Παράθυρο: $w \in [0,1]$

Αφού υλοποιήσετε τον ελεγκτή αποθηκεύστε τον με το όνομα της ομάδας σας και κατάληξη `fis`. Στη συνέχεια ανοίξτε με τον Matlab Editor το αρχείο `ask501.m`. Στην γραμμή 24 του προγράμματος αντικαταστήστε το `controller_name` με το όνομα του αρχείου που μόλις αποθηκεύσατε. Τρέξτε το αρχείο `ask501.m`, καταγράψτε αποκρίσεις και σχολιάστε.

ΑΣΚΗΣΗ 502

**ΠΑΡΑΔΕΙΓΜΑΤΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ
ΤΟΥ ΛΟΓΙΚΟΥ ΕΛΕΓΚΤΗ S7-200**

ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ LAD και STL

Προγραμματισμός του ελεγκτή S7-200 (σε γλώσσα LAD ή STL) με το λογισμικό STEP7-Micro/WIN

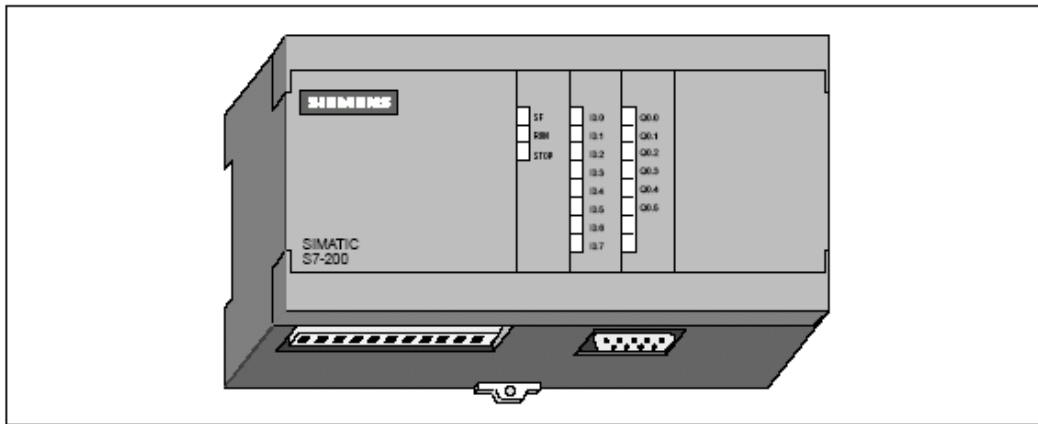
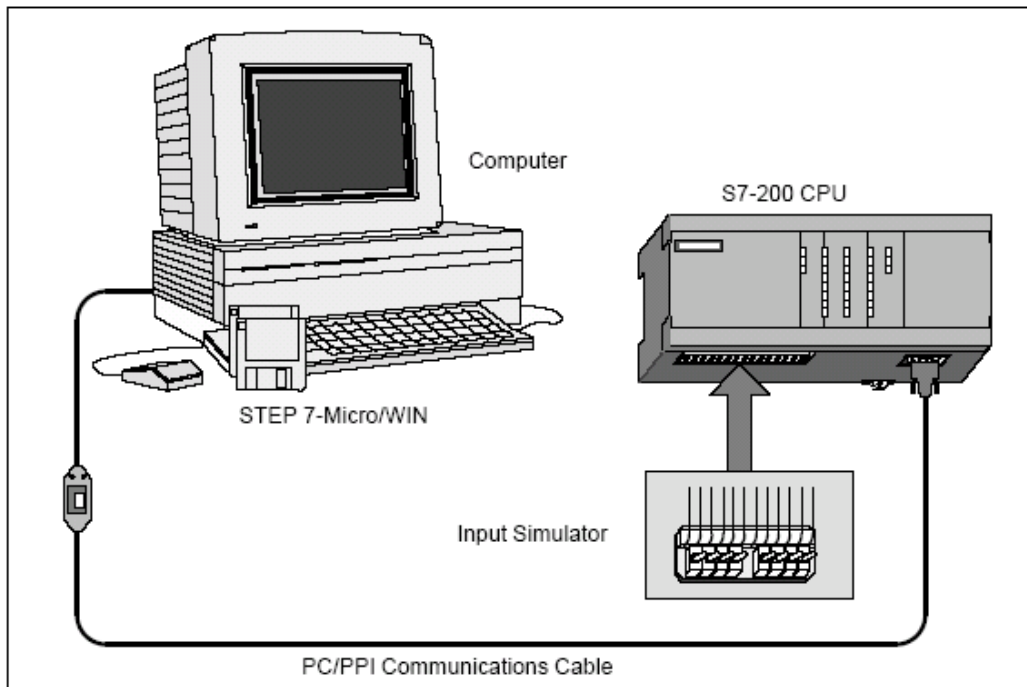


Figure 1-1 S7-200 Micro PLC

Figure 1-2 shows the basic S7-200 Micro PLC system, which includes an S7-200 CPU module, a personal computer, STEP 7-Micro/WIN programming software, and a communications cable.

In order to use a personal computer (PC), you must have one of the following sets of equipment:

- A PC/PPI cable



Changing the Operating Mode with STEP 7-Micro/WIN 32

As shown in Figure 4-13, you can use STEP 7-Micro/WIN 32 to change the operating mode of the CPU. To enable the software to change the operating mode, you must set the mode switch on the CPU to either TERM or RUN.

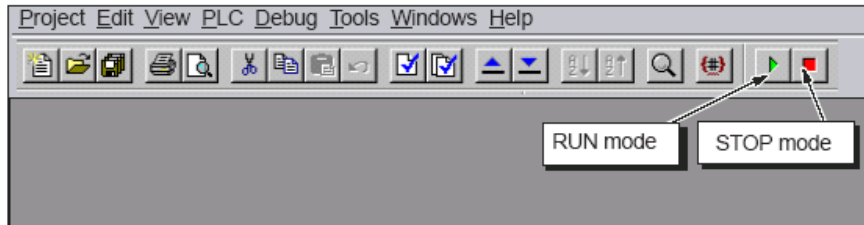


Figure 4-13 Using STEP 7-Micro/WIN 32 to Change the Operating Mode of the CPU

Using Single/Multiple Scans to Monitor Your Program

You can specify that the CPU execute your program for a limited number of scans (from 1 scan to 65,535 scans). By selecting the number of scans for the CPU to run, you can monitor the program as it changes the process variables. Use the menu command **Debug > Multiple Scans** to specify the number of scans to be executed. Figure 4-15 shows the dialog box for entering the number of scans for the CPU to execute.

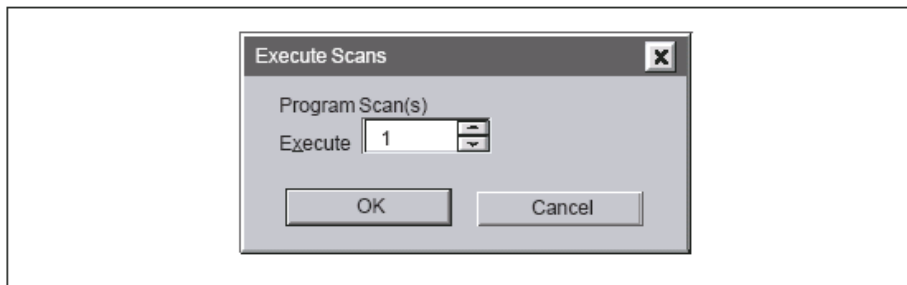


Figure 4-15 Executing Your Program for a Specific Number of Scans

6.3 Concepts of the S7-200 Programming Languages

The S7-200 CPU (and STEP 7-Micro/WIN) supports the following programming languages:

- Statement list (STL) is a set of mnemonic instructions that represent functions of the CPU.
- Ladder logic (LAD) is a graphical language that resembles the electrical relay diagrams for the equipment.

STEP 7-Micro/WIN also provides two representations for displaying the addresses and the programming instructions in the program: international and SIMATIC. Both the international and SIMATIC representations refer to the same S7-200 instruction set. There is a direct correspondence between the international and the SIMATIC representation; both representations have the same functionality.

Understanding the Basic Elements of Ladder Logic

When you write a program in ladder, you create and arrange the graphical components to form a network of logic. As shown in Figure 6-3, the following types of elements are available for creating your program:

- Contacts: each of these elements represents a switch through which power can flow when a switch is closed.
- Coils: each of these elements represents a relay that is energized by power flowing to that relay.
- Boxes: each of these elements represents a function that is executed when power flows to the box.
- Networks: each of these elements forms a complete circuit. Power flows from the left power rail through the closed contacts to energize the coils or boxes.

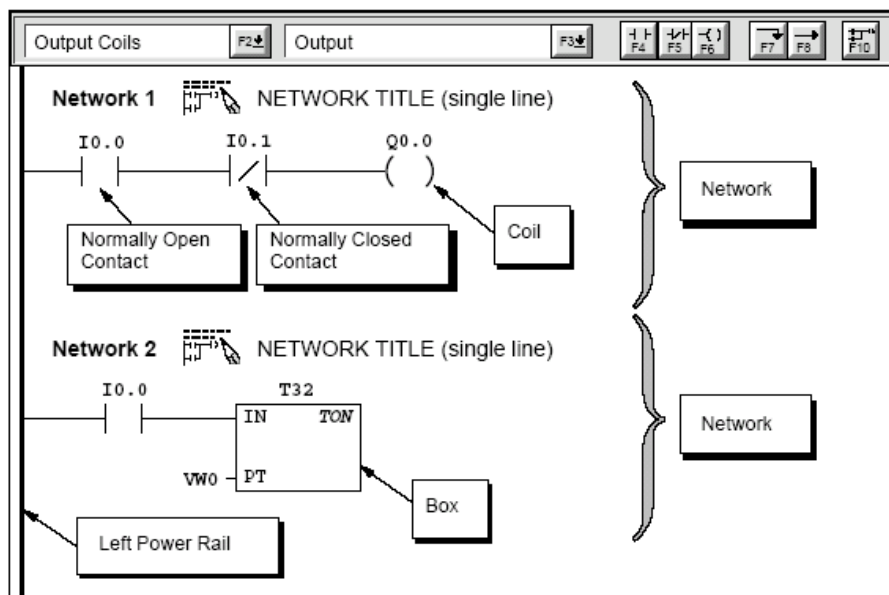


Figure 6-3 Basic Elements of Ladder Logic

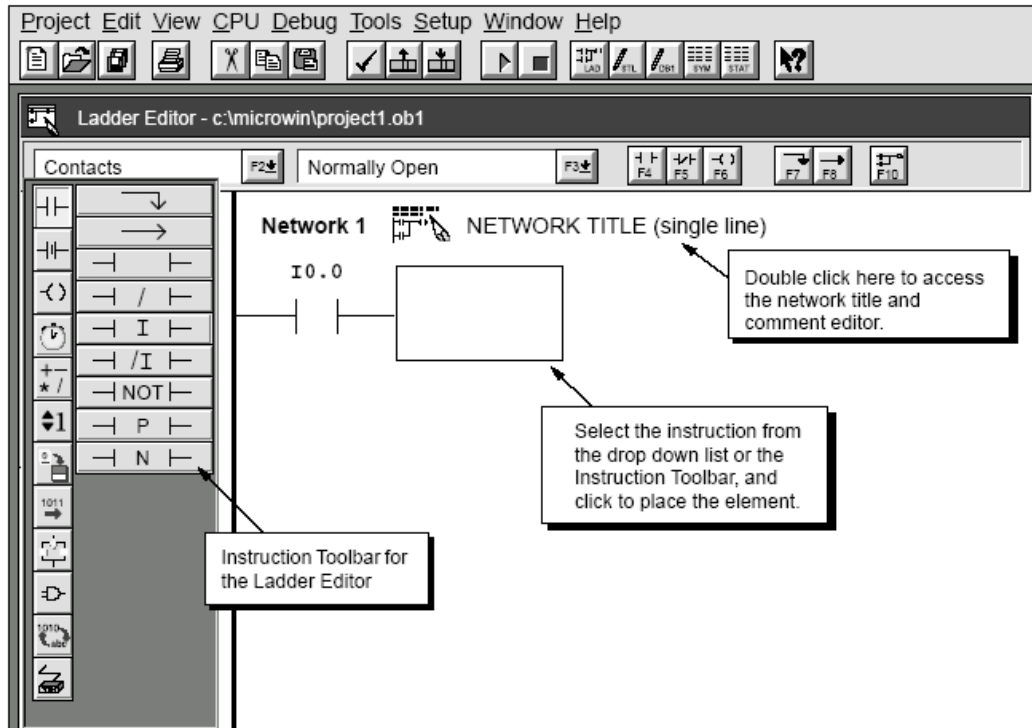


Figure 3-21 Ladder Editor Window

Understanding the Statement List Instructions

Statement list (STL) is a programming language in which each statement in your program includes an instruction that uses a mnemonic abbreviation to represent a function of the CPU. You combine these instructions into a program to produce the control logic for your application.

Figure 6-4 shows the basic elements of a statement list program.

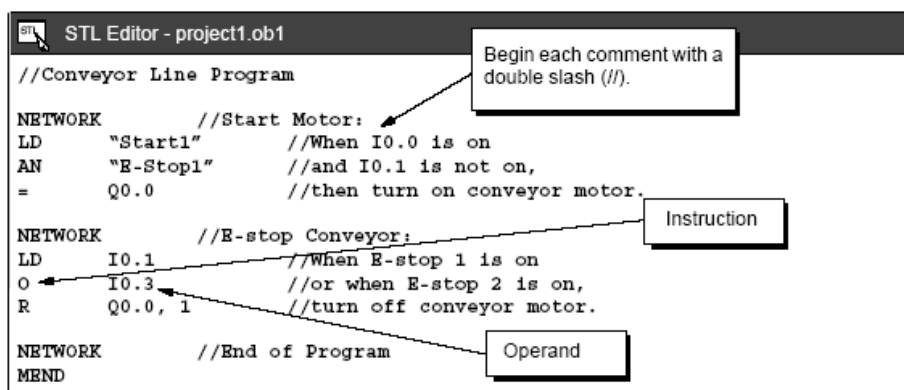


Figure 6-4 STL Editor Window with Sample Program

The STL instructions use a logic stack in the CPU for solving your control logic. As shown in Figure 6-5, this logic stack is nine bits deep by one bit wide. Most of the STL instructions work either with the first bit or with the first and the second bits of the logic stack. New values can be "pushed" (or added) onto the stack; when the top two bits of the stack are combined, the stack is "popped" (reduced by one bit).

While most STL instructions only read the values in the logic stack, many STL instructions also modify the values stored in the logic stack. Figure 6-5 shows examples of how three instructions use the logic stack.

Downloading Your Program

After completing your program, you can download the project to the CPU. To download your program, select the menu command **Project ► Download...** or click the Download button in the main window. 

The Download dialog box that appears allows you to specify the project components that you want to download, as shown in Figure 3-23.

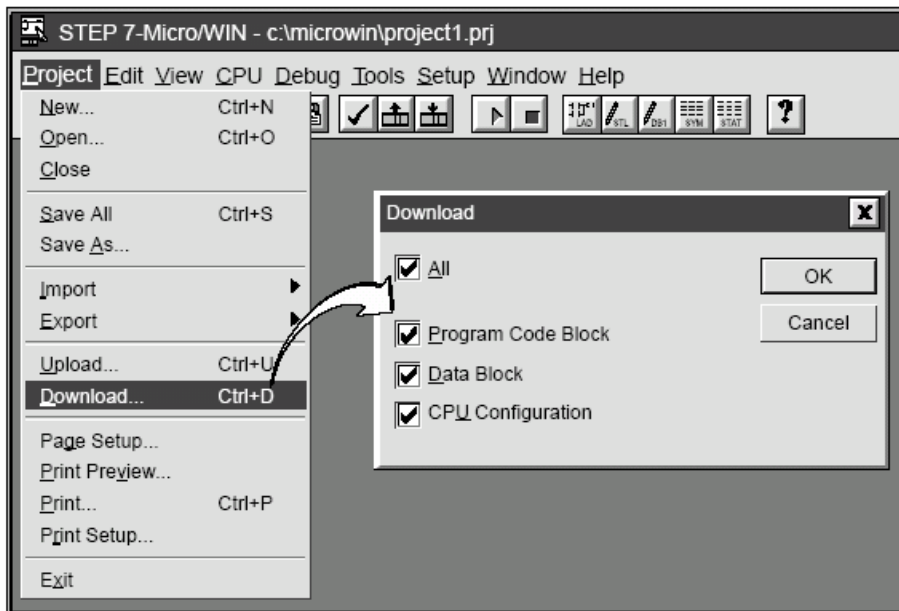


Figure 3-23 Downloading Project Components to the CPU

- Program Code Block (OB1) contains your program logic to be executed by the CPU.
- Data Block (DB1) contains the initialization values to be used by your program.
- CPU Configuration (CFG) contains the system setup information, which includes communication parameters, retentive ranges, input filter selections, password, and output table definitions.

Click the "OK" button or press ENTER to confirm your choices and to execute the download operation.

Viewing a Program in Ladder Logic or Statement List

You can view a program in either ladder or STL by selecting the menu command **View ▶ STL** or **View ▶ Ladder**, as shown in Figure 3-24.

When you change the view from STL to ladder and back again to STL, you may notice changes in the presentation of the STL program, such as:

- Instructions and addresses are changed from lower case to upper case.
- Spaces between instructions and addresses are replaced with tabs.

You can accomplish the same formatting of the STL instructions by selecting the menu command **CPU ▶ Compile** while the STL Editor is active.

Note

Certain combinations of statement list instructions cannot be converted successfully to ladder view. In that case, the message "Illegal Network" marks the section of code that cannot be represented in ladder.

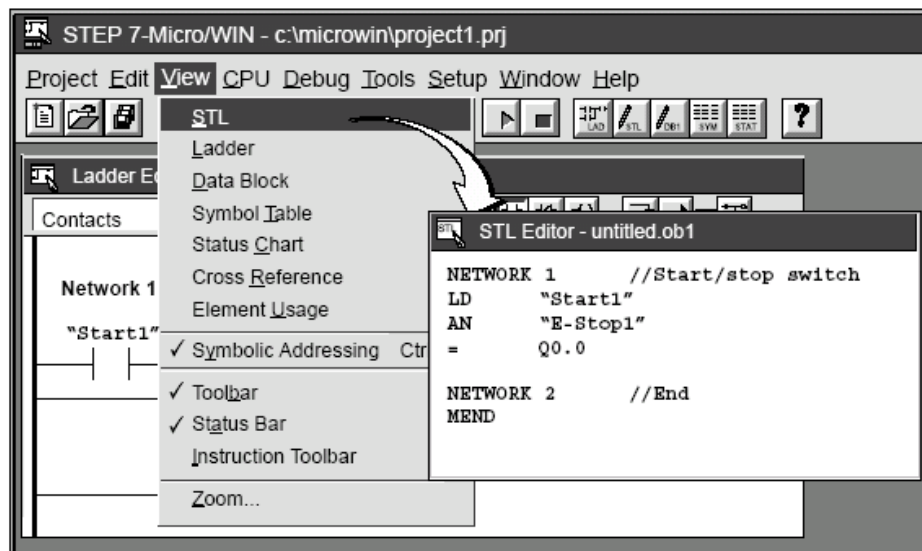


Figure 3-24 Changing the Program View from Ladder to Statement List

3.9 Using Symbolic Addressing

The Symbol Table allows you to give symbolic names to inputs, outputs, and internal memory locations. See Figure 3-27. You can use the symbols that you have assigned to these addresses in the Ladder Editor, STL Editor, and Status Chart of STEP 7-Micro/WIN. The Data Block Editor does not support the use of symbolic names.

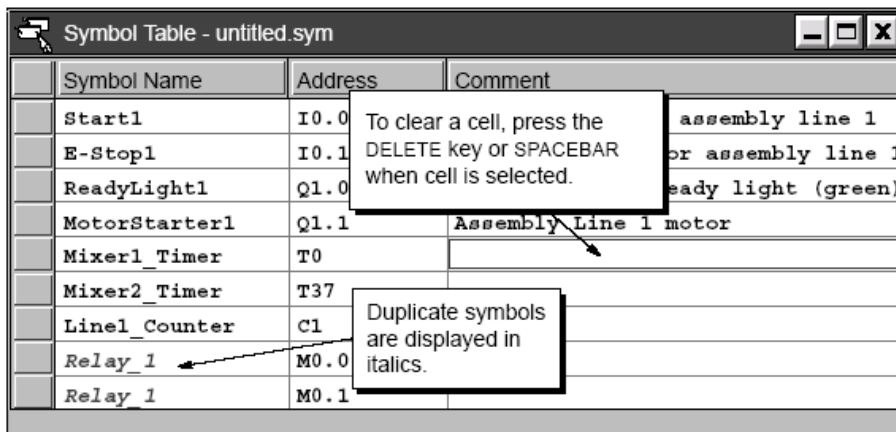
Guidelines for Entering Symbolic Addresses

The first column of the Symbol Table is used to select a row. The other columns are for the symbol name, address, and comment. For each row, you assign a symbolic name to the absolute address of a discrete input, output, memory location, special memory bit, or other element. A comment for each assigned symbol is optional. Follow these guidelines when creating a Symbol Table:

- You can enter symbol names and absolute addresses in any order.
- You can use up to 23 characters in the Symbol Name field.
- You can define up to 1,000 symbols.
- The Symbol Table is case-sensitive: for example, "Pump1" is considered a different symbol from "pump1".
- All leading and trailing spaces are removed from the symbol name by the Symbol Table Editor. All adjacent internal spaces are converted to a single underscore. For example, "Motor starter 2" changes to "Motor_starter_2".
- If you have duplicate symbol names and/or addresses, they are marked with blue italics by the Symbol Table Editor. These duplicate names are not compiled, and they are not recognized outside the symbol table. Overlapping addresses are not flagged as duplicates; for example, VB0 and VW0 overlap in memory but are not flagged as duplicates.

Starting the Symbol Table Editor

The Symbol Table Editor appears by default as a minimized window icon at the bottom of the main window. To access the Symbol Table, double-click the icon, or click the Restore or Maximize button on the icon (in Windows 95).



	Symbol Name	Address	Comment
	Start1	I0.0	assembly line 1
	E-Stop1	I0.1	or assembly line 1
	ReadyLight1	Q1.0	ready light (green)
	MotorStarter1	Q1.1	Assembly Line 1 motor
	Mixer1_Timer	T0	
	Mixer2_Timer	T37	
	Line1_Counter	C1	
	<i>Relay_1</i>	M0.0	
	<i>Relay_1</i>	M0.1	

Figure 3-27 Example of a Symbol Table

Displaying the Status of the Program in Ladder Logic

You can monitor the status of the ladder program by using STEP 7-Micro/WIN 32. STEP 7-Micro/WIN 32 must be displaying ladder logic. Ladder status displays the status of all instruction operand values. See Figure 4-17. All status is based upon the value of these elements that are read at the end of a PLC scan cycle. STEP 7-Micro/WIN 32 acquires the values for the status display across multiple PLC scan cycles and then updates the ladder status screen display. Consequently, the ladder status display does not reflect the actual status of each ladder element of the time of execution.

To open the LAD status window, select the status icon from the toolbar (Figure 4-17).

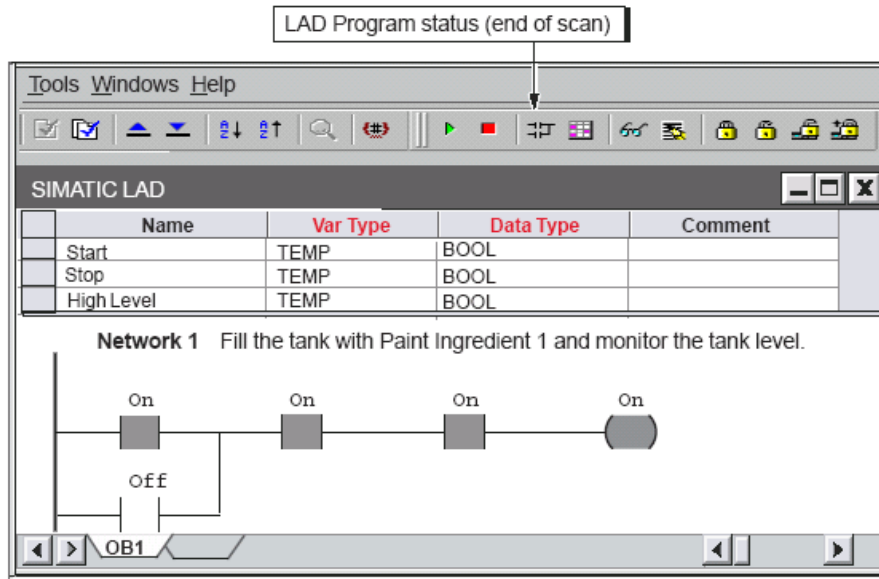


Figure 4-17 Displaying the Status of a Program in Ladder Logic

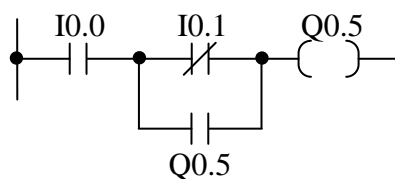
ΕΦΑΡΜΟΓΕΣ-ΠΑΡΑΔΕΙΓΜΑΤΑ

Ο προγραμματιζόμενος λογικός ελεγκτής **S7-200** διαθέτει τις ακόλουθες εισόδους/εξόδους:

Είσοδοι: I0.0, I0.1, I0.2, I0.3, I0.4, I0.5, I0.6, I0.7

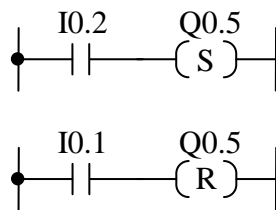
Έξοδοι: Q0.0, Q0.1, Q0.2, Q0.3, Q0.4, Q0.5

1. Προγραμματισμός μιας αυτοσυγκράτησης



Υλοποιήστε το πιο πάνω πρόγραμμα LAD καθώς και όλα τα πιο κάτω προγράμματα και δοκιμάστε τη λειτουργία τους.

2. “Ρελέ” μνήμης (με μανδάλωση)

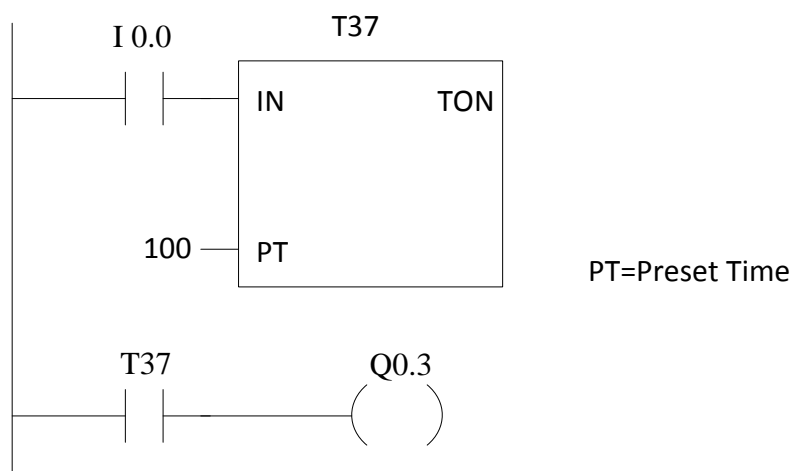


Ενεργοποιώντας στιγμιαία την είσοδο I0.2, η έξοδος Q 0.5 ενεργοποιείται και παραμένει ενεργοποιημένη ανεξάρτητα από την κατάσταση της εισόδου I0.2.

Ενεργοποιώντας στιγμιαία την είσοδο I0.1 η έξοδος απενεργοποιείται.

Αν χρησιμοποιήσουμε την ίδια είσοδο στους δύο κλάδους του προγράμματος τότε ενεργοποιώντας στιγμιαία την είσοδο αυτή, η έξοδος θα είναι συνεχώς απενεργοποιημένη. Αυτό γίνεται γιατί το τελικό αποτέλεσμα εξαρτάται από το ποια εντολή εκτελείται τελευταία.

3. Χρονικά στοιχεία (Χρονοδιακόπτες)

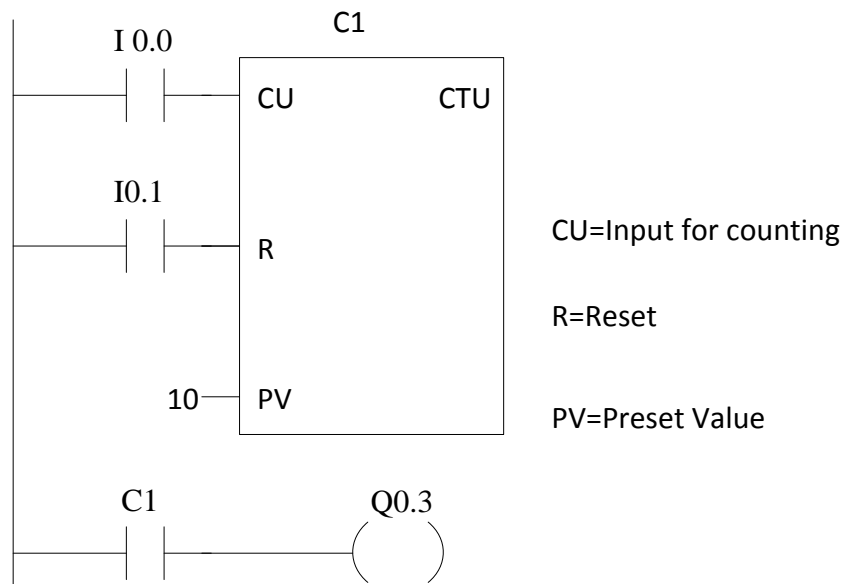


Ενεργοποιώντας την είσοδο I 0.0 το χρονικό T37 αρχίζει να μετράει το χρόνο που ορίζουμε στην είσοδο PT. Μετά από 10 sec η έξοδος Q0.3 ενεργοποιείται.

Εξηγείστε με τη βοήθεια (*) του “HELP Timer On-Delay” γιατί επελέγη ο Timer T37 και γιατί τέθηκε χρόνος 100 μονάδες στην είσοδο PT.

(*) Στο Φάκελο “Instructions” → στον υποφάκελο “Timers” → στο στοιχείο “TON” → δεξί κλικ → πατάμε HELP.

4.Απαριθμητές



Ενεργοποιώντας και απενεργοποιώντας την είσοδο I0.0 10 φορές, η έξοδος Q0.3 ενεργοποιείται.

Υλοποιήστε ένα ανάλογο πρόγραμμα για απαρίθμηση του ίδιου απαριθμητή προς τα πάνω από την είσοδο I0.0, απαρίθμηση προς τα κάτω από την είσοδο I0.1 και Reset με την είσοδο I0.2.

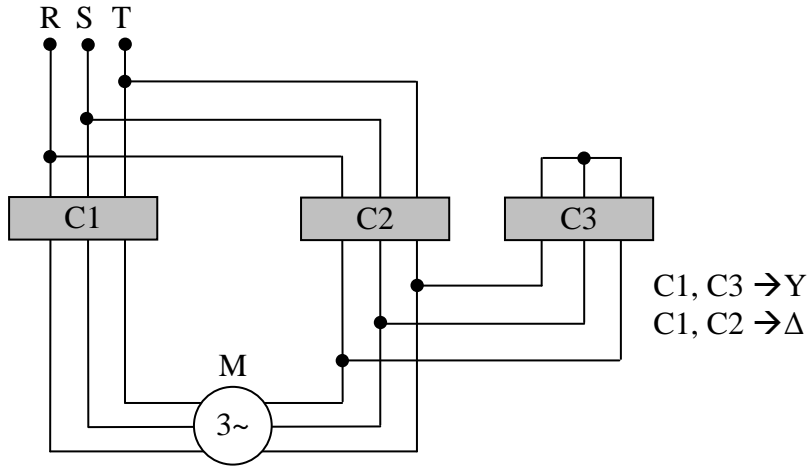
5. Λειτουργία μικροκινητήρα με δύο φορές περιστροφής

Στις εξόδους Q0.0 και Q0.1 είναι συνδεδεμένος ένας μικρός κινητήρας για εκπαιδευτικούς σκοπούς. Όταν ενεργοποιηθεί η έξοδος Q0.0 ο κινητήρας περιστρέφεται δεξιά. Όταν ενεργοποιηθεί η έξοδος Q0.1 (και ταυτόχρονα είναι ενεργοποιημένη η Q0.0) περιστρέφεται αριστερά.

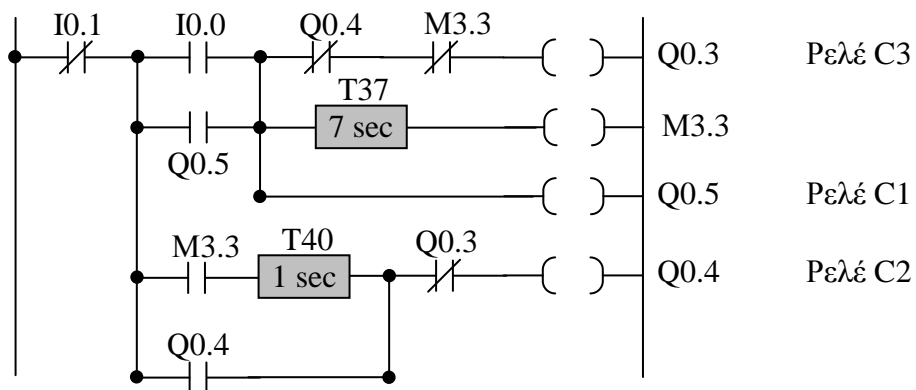
- 1) Να προγραμματισθεί ο ΠΛΕ έτσι ώστε ο κινητήρας να περιστρέφεται αριστερά ή δεξιά ανάλογα με την ενεργοποιούμενη είσοδο I0.0 ή I0.1 αντίστοιχα.
- 2) Να προγραμματισθεί ο ΠΛΕ έτσι ώστε ο κινητήρας να αλλάζει φορά περιστροφής αυτόματα κάθε 10 sec άπαξ και ενεργοποιηθεί η είσοδος I0.0.
- 3) Το πρόγραμμα (2) να μετατραπεί σε γλώσσα STL και κατόπιν σε «συμβολικό» εισάγοντας όποιες αντίστοιχες συμβολικές μεταβλητές θέλετε για τις εισόδους, εξόδους, χρονιστές κλπ.

6. Εκκίνηση – Λειτουργία κινητήρα κατά Υ/Δ

Κύκλωμα ισχύος



Πρόγραμμα



Υλοποιήστε το πιο πάνω πρόγραμμα LAD και κατεβάστε το στο PLC.

ΠΡΟΣΟΧΗ: Στο πιο πάνω πρόγραμμα οι χρονιστές εμφανίζονται συμβολικά. Ο προγραμματισμός τους πρέπει να γίνει σύμφωνα με τη λογική που εφαρμόσατε πιο πάνω στο (3).

Οι διακόπτες START, STOP αντιστοιχούν στις εισόδους I0.0 και I0.1 αντίστοιχα.

ΠΡΟΣΟΧΗ: Να γίνει έλεγχος για τυχόν σφάλματα στον προγραμματισμό και στο κύκλωμα ισχύος. Στη συνέχεια θέστε σε λειτουργία τον κινητήρα και παρατηρήστε τις χρονικές αλλαγές του κινητήρα από Υ σε Δ.

ΡΟΜΠΟΤΙΚΟΣ ΒΡΑΧΙΟΝΑΣ ΚΑΤΑΝΑ 400

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 503



ΧΕΙΡΟΚΙΝΗΤΟΣ ΕΛΕΓΧΟΣ

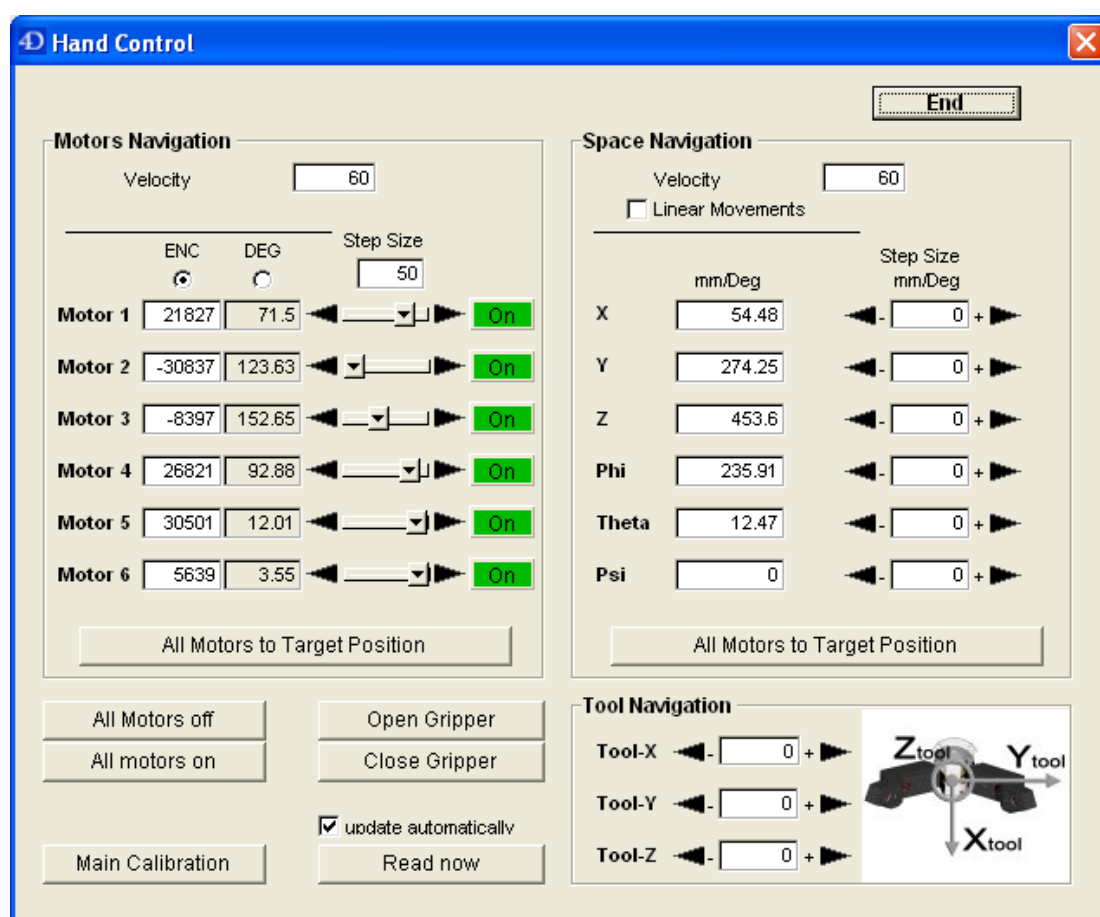
Με το χειροκίνητο έλεγχο μπορείτε να λειτουργήσετε το ρομπότ πολύ εύκολα. Για να κινήσετε το ρομπότ μπορείτε να επιλέξετε μία εκ των τριών καταστάσεων πλοήγησης που είναι η «πλοήγηση κινητήρων», η «πλοήγηση στο χώρο» και η «πλοήγηση εργαλείου».

Με τη χρήση της πλοήγησης κινητήρων κάθε άξονας του ρομπότ ελέγχεται ανεξάρτητα.

Με την πλοήγηση στο χώρο μπορείτε να κινήσετε το ρομπότ με βάση το σύστημα συντεταγμένων.

Με την πλοήγηση εργαλείου μπορείτε να κινήσετε το ρομπότ με βάση την επιθυμητή θέση του κεντρικού σημείου αναφοράς του εργαλείου.

Όλες οι δυνατότητες κίνησης του ρομπότ με χειροκίνητο έλεγχο φαίνονται στο μενού που ακολουθεί.



1. Πλοήγηση Κινητήρων

Με κλικ πάνω στα τόξα που υπάρχουν δίπλα σε κάθε κινητήρα ρυθμίζετε τον αριθμό βημάτων που θα κινηθεί ο αντίστοιχος κινητήρας αριστερά/δεξιά (ή πάνω/κάτω), σύμφωνα με το μέγεθος βημάτων (step size) και την κατεύθυνση που έχετε επιλέξει. (ENC=Encoder, DEG=Degree). Οι θέσεις προορισμού των κινητήρων μπορούν επίσης να εισαχθούν και απ' ευθείας στο αντίστοιχο πεδίο τιμών.

Πιέστε **All motors to target position** για να κινηθεί το ρομπότ στη θέση που εισάγατε.
Να είστε προσεκτικοί κάθε φορά που κινείται το ρομπότ.

Υποδείξεις:

- Αν επιλέξετε “DEG”, οι αριθμητικές τιμές που θα εισάγετε θα αναφέρονται στις γωνίες των κινητήρων.
- Επιλέξτε μέγεθος βημάτων για τον Κωδικοποιητή (ENC) μεταξύ 50 και 5000.
- Να εξετάζετε με προσοχή αν το ρομπότ ΚΑΤΑΝΑ μπορεί πραγματικά να κινηθεί στη θέση που του ορίζετε να κινηθεί (π.χ. ύπαρξη εμποδίων ή του τραπέζιου έδρασης).

2. Πλοήγηση στο χώρο

Η κίνηση του ρομπότ μπορεί να ελεγχθεί με βάση το τρισδιάστατο καρτεσιανό σύστημα συντεταγμένων. Η αρχή του συστήματος συντεταγμένων είναι στο μέσον της 2^{ης} άρθρωσης, όπως φαίνεται στο σχήμα, όπου ο άξονας Z συμπίπτει με το «νήμα της στάθμης».



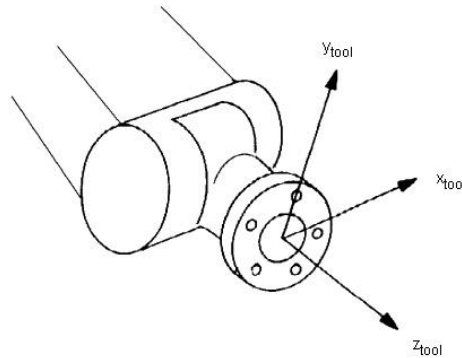
Σύστημα συντεταγμένων βάσης

Για να ελέγξετε πλήρως την κίνηση του ρομπότ θα πρέπει να ορίσετε τις τιμές των τριών μεταβλητών **X**, **Y**, **Z** καθώς και των γωνιών **Phi**, **Theta** και **Psi**. Οι τελευταίες καθορίζουν τον προσανατολισμό της λαβής στο χώρο όπως φαίνεται στο σχήμα αλλά και στο μενού

του χειροκίνητου ελέγχου. Οι μονάδες είναι αντίστοιχα mm και degrees. Στα πεδία τιμών της περιοχής Space Navigation ορίστε την επιθυμητή μετακίνηση και περιστροφή.

Πιέστε **All motors to target position** για να κινηθεί το ρομπότ στη θέση που ορίσατε.

Αν επιλέξετε “Linear Movements” το ρομπότ θα κινηθεί στην επιθυμητή θέση με γραμμική κίνηση.



Σύστημα συντεταγμένων εργαλείου

3. Πλοήγηση Εργαλείου

Με εισαγωγή αριθμητικών τιμών στα πεδία τιμών της περιοχής “Tool Navigation” το ρομπότ θα κινηθεί σε σχέση με το σύστημα συντεταγμένων του εργαλείου(λαβίδας). Ο άξονας Z είναι κατά μήκος του εργαλείου και ο άξονας Y κατά μήκος των δακτύλων.

Για τον ορισμό της ταχύτητας και την επιλογή γραμμική/μη γραμμική κίνηση χρησιμοποιούνται τα πεδία του Space Navigation.

4. Πρόσθετες Εντολές Χειρισμού

All Motors Off = Θέτει εκτός λειτουργίας όλους τους κινητήρες. Πατήστε αυτό το μπουτόν αν θέλετε να κινήσετε το ρομπότ με τα χέρια σας.

Προσοχή: Κρατήστε το ρομπότ με τα χέρια σας πριν «κλείσετε» τους κινητήρες γιατί αν βρίσκεται σε έκταση θα πέσει στο τραπέζι έδρασης.

All Motors On = Θέτει σε λειτουργία όλους τους κινητήρες. Το ρομπότ παραμένει στη θέση που έχει πριν πατήσετε το **All Motors On**.

Open Gripper = Ανοίγει τη λαβίδα

Close Gripper = Κλείνει τη λαβίδα

Main Calibration = Πραγματοποιεί την κύρια βαθμονόμηση-αρχικοποίηση του ρομπότ. Κάθε φορά που ανοίγουμε το ρομπότ για να λειτουργήσει (Power On) οι κινητήρες του πρέπει να αρχικοποιηθούν. Αυτό γίνεται πατώντας

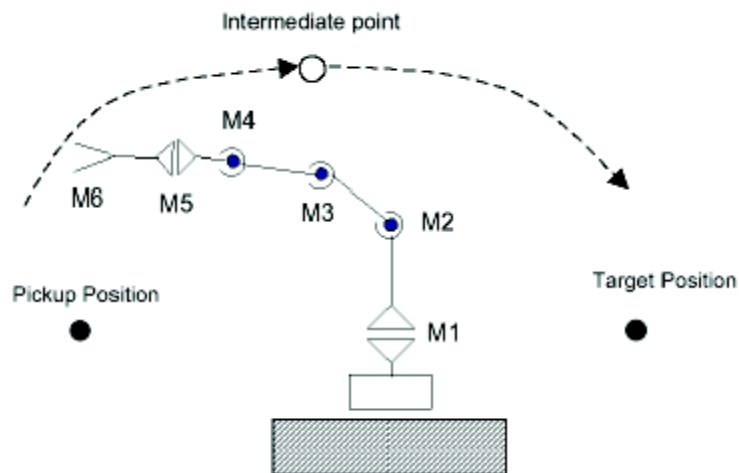
Main Calibration. Κύρια Αρχικοποίηση σημαίνει ότι οι κινητήρες περιστρέφονται αργά προς τα αντίστοιχα μηχανικά τέρματα (stops) αυτών και στη θέση αυτή οι κωδικοποιητές τίθενται στις προεπιλεγμένες τιμές.

Αρχικοποίηση πρέπει να γίνεται και κάθε φορά που το ρομπότ συγκρούεται με ένα εμπόδιο.

Read now = Ενημέρωση του υπολογιστή για τη θέση των κινητήρων ή τη θέση του ρομπότ στο χώρο με διάβασμα από τη μνήμη του ρομπότ.

Δημιουργία εφαρμογής «Πιάσε-Θέσε» (Pick & Place) με χρήση του Wizard

Σε αυτή την εφαρμογή, για να δημιουργήσουμε ένα παράδειγμα «Πιάσε-Θέσε» ένα αντικείμενο, χρειάζεται να ορίσουμε δύο σημεία. Το ένα σημείο είναι η θέση του αντικειμένου το οποίο θέλουμε να πιάσει το ρομπότ και το δεύτερο σημείο είναι η θέση στην οποία θέλουμε να το τοποθετήσει. Αν υποθέσουμε ότι υπάρχει ένα εμπόδιο μεταξύ των δύο σημείων και θέλουμε το ρομπότ να το αποφύγει, τότε πρέπει να ορισθεί και ένα ενδιάμεσο σημείο όπως φαίνεται στο σχήμα που ακολουθεί.

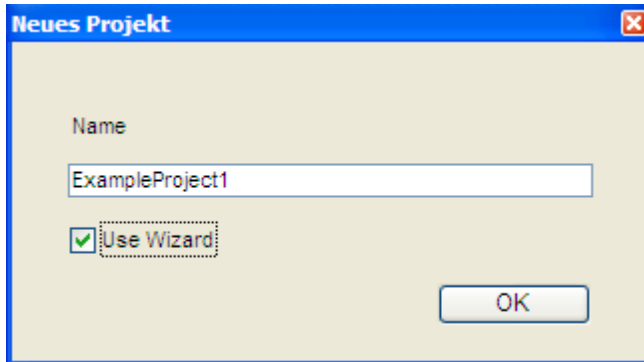


Γραφική αναπαράσταση του έργου «Πιάσε-Θέσε» του ρομπότ.

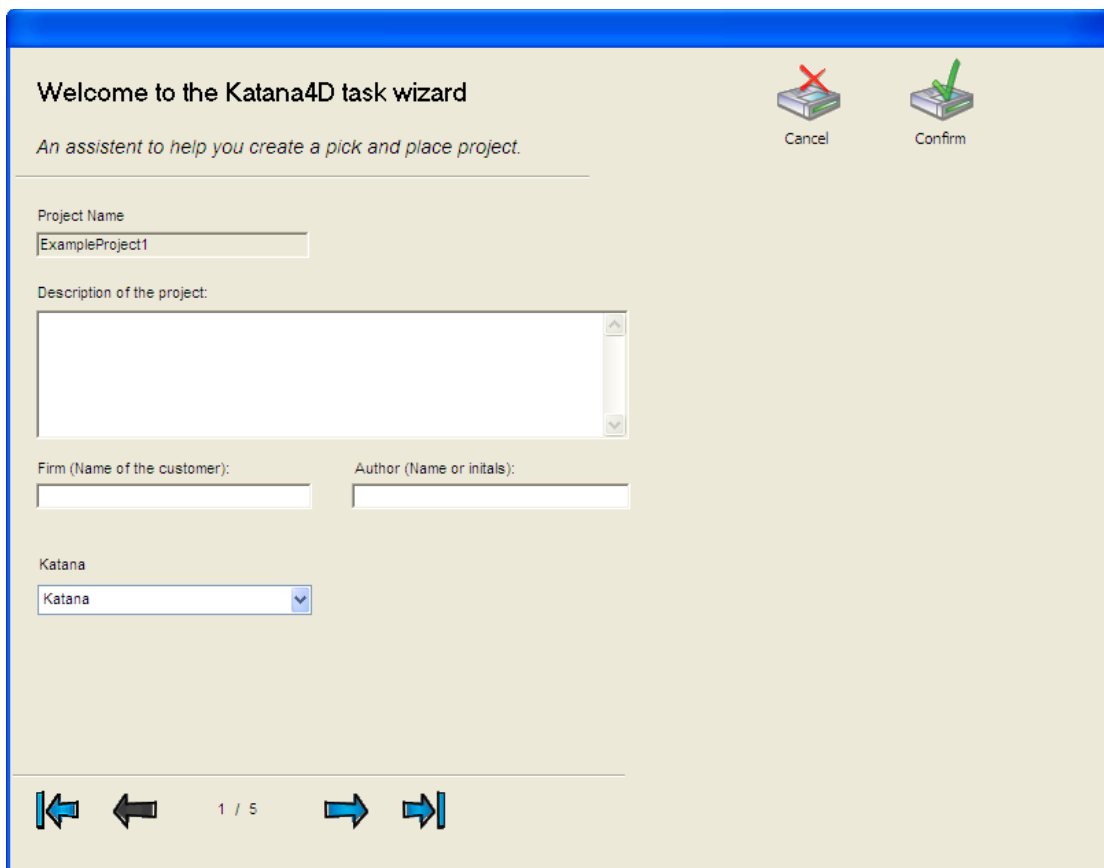
Βήμα 1: Προετοιμασία

Τοποθετήστε το ρομπότ σε μία ενδιάμεση θέση έτσι ώστε να υπάρχει ικανός ενεργός χώρος σε κάθε πλευρά του ρομπότ. Τοποθετήστε το αντικείμενο στο σημείο από το οποίο θα το πιάσει το ρομπότ και αποφασίστε για το σημείο στο οποίο θέλετε να το αφήσει.

- Επιλέξτε Login/Logout και εισαχθείτε σαν Administrator.
- Πατήστε New Project στην ετικέτα Projects & Programs.
- Στο πεδίο κειμένου Name, γράψτε ExampleProject1.



- Ενεργοποιείτε την επιλογή Use Wizard και πατήστε OK.
- Σε αυτό το σημείο εμφανίζεται η αρχική μορφή του Wizard.



Βήμα 2: Αρχικά στοιχεία του Wizard

Το Wizard αποτελείται από διάφορες σελίδες στις οποίες μεταβαίνουμε με τη βοήθεια των μπλε τόξων. Το Wizard περιέχει εξηγήσεις και καθοδηγεί τον χρήστη βήμα-προς-βήμα για όλες τις ενέργειες που πρέπει να γίνουν προκειμένου να δημιουργηθεί ένα εκτελέσιμο πρόγραμμα που θα υλοποιεί την επιθυμητή κίνηση.

- Γράψτε αν θέλετε μια σύντομη περιγραφή του έργου στο αντίστοιχο πεδίο κειμένου όπως π.χ. «Κίνηση τριών σημείων για pick&place αντικειμένου».

- Στα επόμενα δύο πεδία κειμένου εισάγονται ατομικά στοιχεία.
- Στο επόμενο πεδίο επιλέξτε ΚΑΤΑΝΑ
- Πηγαίνετε στην επόμενη σελίδα

Βήμα 3: Πιάσιμο αντικειμένου

Η δεύτερη σελίδα, που φαίνεται πιο κάτω, ορίζει που και πως θα γίνει η παραλαβή του αντικειμένου από το ρομπότ.

- Πατήστε **All Motors Off**.

Προσοχή: Κρατήστε το ρομπότ με τα χέρια σας πριν «κλείσετε» τους κινητήρες γιατί αν βρίσκεται σε έκταση θα πέσει στο τραπέζι έδρασης.


- Κινήστε το ρομπότ με τα χέρια σας μέχρι το σημείο που βρίσκεται το αντικείμενο.
- Πατήστε **All Motors On** και αφήστε ελεύθερο το ρομπότ.
- Αν είναι αναγκαίο διορθώστε τη θέση του ρομπότ με το *Χειροκίνητο Έλεγχο* (βλ. αντίστοιχη παράγραφο).
- Πατήστε **Teach** για να καταλάβει το ρομπότ το σημείο στο οποίο θα πιάσει το αντικείμενο. Οι συντεταγμένες X, Y, Z εμφανίζονται στα αντίστοιχα πεδία.

Το σημείο λήψης του αντικειμένου έχει τώρα αποθηκευτεί.

- Στα επόμενα δύο πεδία (στο κάτω μέρος της σελίδας 2) εισάγονται δύο αριθμοί που εκφράζουν μήκος σε mm και συγκεκριμένα,


α) Ο πρώτος εκφράζει το ύψος επάνω από το αντικείμενο προς το οποίο θα κινηθεί το ρομπότ πριν κινηθεί απ' ευθείας για να πιάσει το αντικείμενο.

β) Ο δεύτερος εκφράζει το ύψος επάνω από το αντικείμενο προς το οποίο θα κινηθεί το ρομπότ αφού έχει πιάσει το αντικείμενο, πριν κινηθεί προς το ενδιαμέσο σημείο. Οι προεπιλεγμένες τιμές είναι 20 mm. **Μην αλλάξετε τις τιμές αυτές.**



- Πατήστε  για να πάτε στην επόμενη σελίδα.

Βήμα 4: Εναπόθεση αντικειμένου

Η διαδικασία για τον ορισμό του σημείου εναπόθεσης του αντικειμένου είναι ακριβώς η ίδια με αυτή για τη λήψη του αντικειμένου (βήμα 3).

- Επαναλάβετε όλες τις ενέργειες του βήματος 3 για τη σελίδα 3 “Target Position of the Object”, που φαίνεται πιο κάτω.
- Πατήστε πάλι  για να πάτε στην επόμενη σελίδα.


Target position of the object

 Cancel
 Confirm

Multiple Position





Please move the robot by hand to the target position. Then correct the position with "Hand Control". Teach the pick up point at the end with "Teach".

x	y	z	
-100	100	100	Teach



How much should be the position above the target position? mm

How much over the grab point should the robot moves after the object positioning? mm



3 / 5



Βήμα 5: Ενδιάμεσα σημεία

- Το παράθυρο που ακολουθεί μας επιτρέπει να ορίσουμε τα ενδιάμεσα σημεία μέσω των οποίων θα κινηθεί το ρομπότ.

Waypoints

Cancel Confirm

Chose the type of the movement between the intermediate points or between the pickup and the target positions

Linear Movements

no linear movement

Are there points between pickup and target position?



Create new intermediate Point

Hand Control

All Motors off All motors on

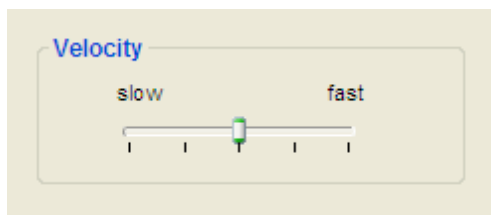
Name	Order :	X :	Y :	Z :	taget wa	return wa	<input type="checkbox"/> Lin
------	---------	-----	-----	-----	----------	-----------	------------------------------

4 / 5

- Επιλέξτε τη δυνατότητα “No Linear Movements” αν δεν είναι ήδη επιλεγμένη.
- Επιλέξτε επίσης την επόμενη δυνατότητα που ορίζει ότι υπάρχουν ενδιάμεσα σημεία.
- Πριν προχωρήσετε στην επόμενη ενέργεια, αποφασίστε τώρα αν το ενδιάμεσο σημείο θα ακολουθείται κατά την προς το στόχο κίνηση του ρομπότ ή/και κατά την κίνηση επιστροφής.
- Πατήστε **Create new intermediate point**
- Με την προηγούμενη ενέργεια εμφανίζεται μία γραμμή στον πίνακα στην οποία θα αποθηκευτεί το ενδιάμεσο σημείο.
- Το προεπιλεγμένο όνομα του σημείου είναι Z1. Με κλικ στο όνομα μπορείτε να το αλλάξετε.
- Για να ορίσετε το ενδιάμεσο σημείο πατήστε **All Motors Off**, κινήστε το ρομπότ στην επιθυμητή ενδιάμεση θέση και πατήστε **All Motors On**.
- Κάντε κλικ στη γραμμή του πίνακα. Θα δείτε τρία σύμβολα στη δεξιά πλευρά του πίνακα. Πατήστε  για να αποθηκευτεί το ενδιάμεσο σημείο.
- Πατήστε  για να πάτε στην τελευταία σελίδα.

Βήμα 6: Ρύθμιση ταχύτητας

- Στην τελευταία σελίδα ρυθμίστε την ταχύτητα κίνησης του ρομπότ στην περιοχή αργά.

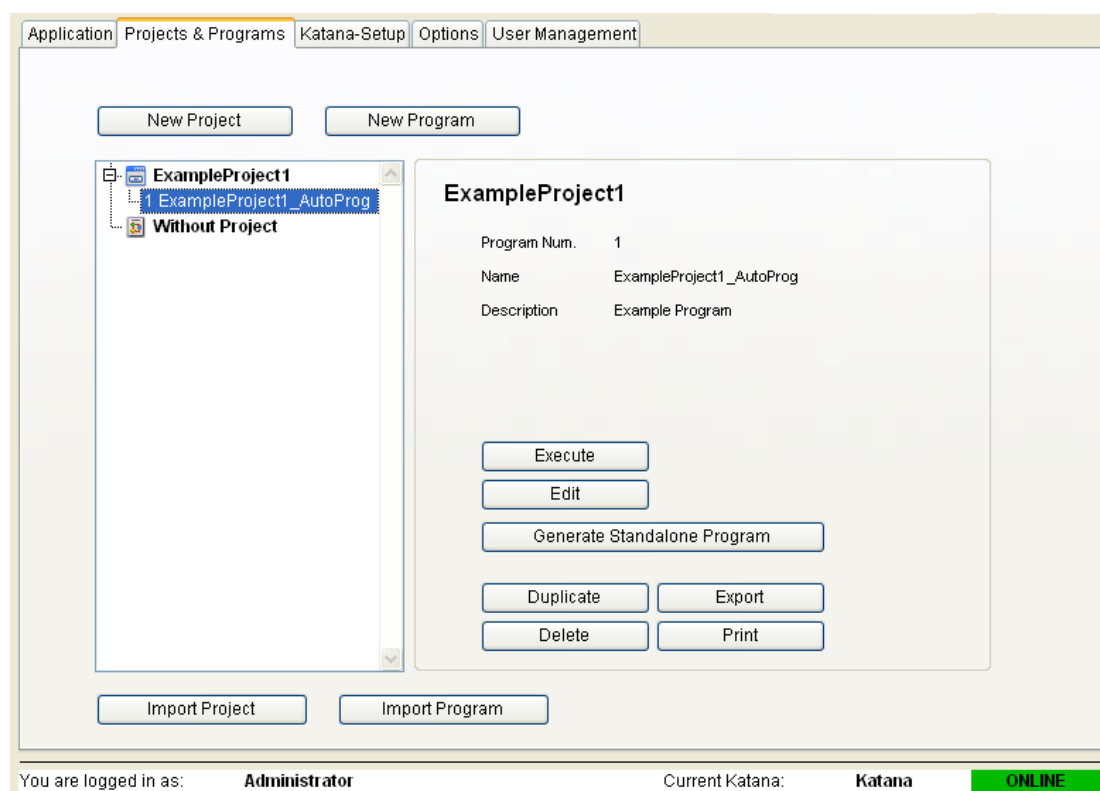


- Πατήστε **Confirm**. Τότε το Wizard κλείνει και δημιουργείται το πρόγραμμα της εφαρμογής.

Βήμα 7: Τρέξιμο του προγράμματος

Η δημιουργία της εφαρμογής «Πιάσε-Θέσε» έχει ολοκληρωθεί. Το Wizard έχει τώρα προσθέσει ένα project με το όνομα ExampleProject1 και ένα πρόγραμμα με το όνομα ExampleProject1_AutoProg στην ετικέτα Projects & Programs. Το project περιέχει το «πλέγμα» των σημείων και το πρόγραμμα τον κώδικα των κινήσεων.

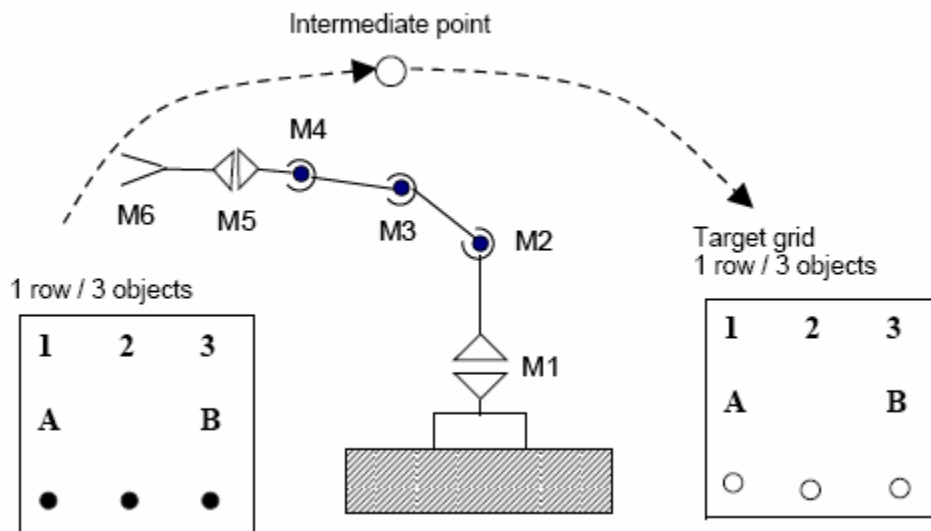
- Το πρόγραμμα εκτελείται πατώντας **Execute**.



- Αν θέσετε αριθμό επαναλήψεων 1, μόνο ένα αντικείμενο θα μετακινηθεί. Με αριθμό 3 ή μεγαλύτερο θα μετακινηθούν 3 ή ισάριθμα αντικείμενα. Με αριθμό 0 το πρόγραμμα θα εκτελείται επαναληπτικά μέχρι να πατηθεί είτε το πλήκτρο Shift είτε κάνετε κλικ στο **End**.

Δημιουργία εφαρμογής «Πιάσε-Θέσε» (Pick & Place) πολλαπλών αντικειμένων από διαφορετικές θέσεις με χρήση του Wizard

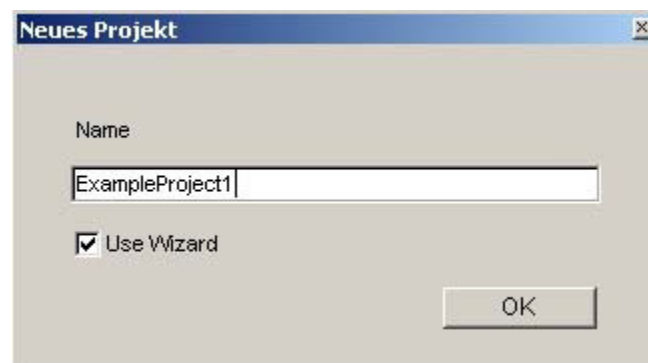
Στο παράδειγμα αυτό θα δημιουργήσετε δύο “πλέγματα” σημείων εκ των οποίων το ένα αντιστοιχεί στα σημεία παραλαβής των αντικειμένων από το ρομπότ, και το δεύτερο στα σημεία εναπόθεσης των αντικειμένων. Κάθε πλέγμα χαρακτηρίζεται από το πλήθος των γραμμών και από το πλήθος των αντικειμένων ανά γραμμή. Στο παράδειγμα θα ορίσετε μία γραμμή σε κάθε πλέγμα και τρία αντικείμενα ανά γραμμή, όπως φαίνεται στο επόμενο σχήμα.



Γραφική αναπαράσταση του έργου «Πιάσε-Θέσε» πολλαπλών σημείων.

Τοποθετήστε το ρομπότ σε μία ενδιάμεση θέση έτσι ώστε να υπάρχει ικανός ενεργός χώρος σε κάθε πλευρά του ρομπότ. Τοποθετήστε τα αντικείμενα στα ακραία σημεία A και B από τα οποία θα τα πιάσει το ρομπότ και αποφασίστε για τα σημεία στα οποία θέλετε να τα αφήσει.



- Πατήστε **New Project** στην ετικέτα Projects & Programs.
- Στο πεδίο κειμένου Name, γράψτε ExampleProject1.



- Ενεργοποιείτε την επιλογή Use Wizard και πατήστε OK.
- Σε αυτό το σημείο εμφανίζεται η αρχική μορφή του Wizard.

Welcome to the Katana4D task wizard

An assistant to help you create a pick and place project.

 Cancel
  Confirm

Project Name

Description of the project:





Firm (Name of the customer): Author (Name or initials):

Katana

Gripper

open (Enc)

close (Enc)



1 / 5



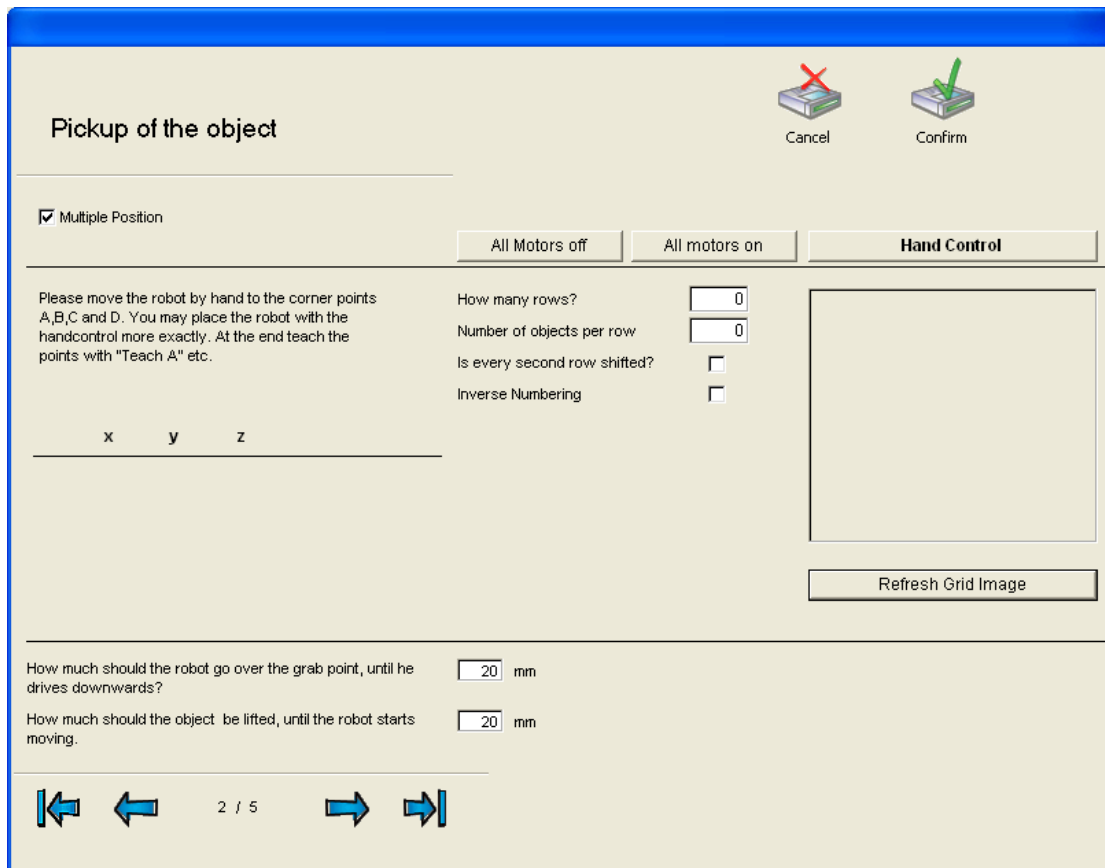
Επαναλάβετε ότι κάνατε και στο προηγούμενο παράδειγμα σχετικά με τα πεδία του πιο πάνω παραθύρου.

- Στην περιοχή **“Gripper”** του παραθύρου μπορείτε να ορίσετε τις τιμές κωδικοποιητή της λαβίδας για ανοικτή και κλειστή κατάσταση αυτής.
- Πηγαίνετε στην επόμενη σελίδα

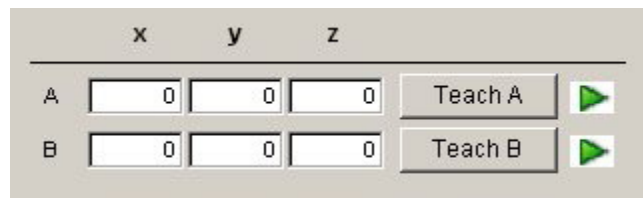
Βήμα 1: Πιάσιμο αντικειμένων

Η δεύτερη σελίδα, που φαίνεται πιο κάτω, ορίζει που και πως θα γίνει η παραλαβή των αντικειμένων από το ρομπότ.

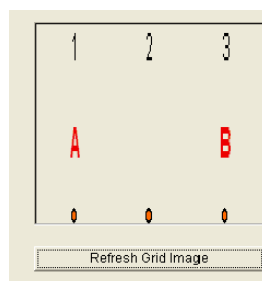
- Επειδή το ρομπότ θα πιάσει περισσότερα του ενός αντικείμενα, ενεργοποιήστε τη επιλογή **“Multiple Position”**.
- Τότε θα εμφανισθεί το παράθυρο που ακολουθεί με τα πρόσθετα απαιτούμενα στοιχεία.



- Γράψτε τον αριθμό 1 στο πεδίο **How many rows?**, τον αριθμό 3 στο πεδίο **Number of objects per row** και πατήστε Enter.
- Τότε θα δείτε τα ακραία σημεία A και B ως ακολούθως,



- Πατήστε **Refresh grid Image** στη δεξιά πλευρά του παραθύρου.
- Τότε θα δείτε τα σημεία του πλέγματος (1X3) που ορίσατε



- Για να αποθηκεύσετε το πλέγμα των τριών σημείων θα πρέπει πρώτα να “διδάξετε” στο ρομπότ με χειροκίνητο τρόπο τα ακραία σημεία A και B, όπως και

στο προηγούμενο παράδειγμα. Τότε αυτό υπολογίζει αυτόματα τα ενδιάμεσα σημεία (για το παράδειγμα ένα σημείο).

- Πατήστε **All Motors Off**.

Προσοχή: Κρατήστε το ρομπότ με τα χέρια σας πριν «κλείσετε» τους κινητήρες γιατί αν βρίσκεται σε έκταση θα πέσει στο τραπέζι έδρασης.

- Κλείστε τη λαβίδα τόσο ώστε το ρομπότ να πιάνει σταθερά το αντικείμενο. Για να το πετύχετε αυτό ενεργοποιήστε το **Hand Control** και κλείστε τη λαβίδα πατώντας **Close Gripper** επαναληπτικά. Κλείστε το παράθυρο **Hand Control** πατώντας **End**.
- Κινήστε το ρομπότ με τα χέρια σας μέχρι το πρώτο σημείο (A) όπου θα βρίσκεται το αντικείμενο, διατηρώντας το άνοιγμα της λαβίδας αμετάβλητο.
- Πατήστε **All Motors On** και αφήστε ελεύθερο το ρομπότ.
- Αν είναι αναγκαίο διορθώστε τη θέση του ρομπότ με το *Χειροκίνητο Έλεγχο* (βλ. αντίστοιχη παράγραφο).
- Πατήστε **Teach A** για να καταλάβει το ρομπότ το πρώτο σημείο στο οποίο θα πιάσει αντικείμενο. Οι συντεταγμένες X, Y, Z εμφανίζονται στα αντίστοιχα πεδία.
- Επαναλάβετε όλες τις προηγούμενες ενέργειες (από την ενέργεια Πατήστε **All Motors Off**) για το σημείο B.
- Τότε θα παρατηρήσετε τις συντεταγμένες των ακραίων σημείων A και B στα αντίστοιχα πεδία.

	x	y	z		
A	50.3	91.3	-20.2	Teach A	▶
B	70.2	89.8	-19.8	Teach B	▶

- Τα σημεία του πρώτου πλέγματος έχουν τώρα αποθηκευτεί.
- Πηγαίνετε στην επόμενη σελίδα.

Βήμα 2: Εναπόθεση αντικειμένων

- Η διδασκαλία του πλέγματος σημείων στα οποία θα τοποθετηθούν τα αντικείμενα είναι ακριβώς η ίδια όπως στο 1^ο βήμα.
- Η μόνη διαφορά είναι ότι τώρα η λαβίδα πρέπει να ανοίξει έτσι ώστε το ρομπότ να εναποθέσει τα αντικείμενα με ομαλό τρόπο.
- Επαναλάβετε όλες τις ενέργειες που κάνατε στο 1^ο βήμα «Πιάσιμο Αντικειμένων», εκτός από την ενέργεια που αντιστοιχεί στο άνοιγμα της λαβίδας.
- Το άνοιγμα της λαβίδας θα γίνει πάλι μέσω του **Hand Control**.

Target position of the object

Cancel
 Confirm

Multiple Position

All Motors off
All motors on
Hand Control

Please move the robot by hand to the corner points A,B,C and D. You may place the robot with the handcontrol more exactly. At the end teach the points with "Teach A" etc.

	x	y	z	
A	<input type="text" value="342.34"/>	<input type="text" value="-274.75"/>	<input type="text" value="-434.7"/>	Teach A
B	<input type="text" value="-577.6"/>	<input type="text" value="-236.59"/>	<input type="text" value="-434.7"/>	Teach B

How many rows?

Number of objects per row

Is every second row shifted?

Inverse Numbering

How much should be the position above the target position? mm

How much over the grab point should the robot moves after the object positioning? mm

- Πηγαίνετε στην επόμενη σελίδα.

Βήμα 3: Ενδιάμεσο σημείο – Ρύθμιση ταχύτητας - Πρόγραμμα

Επαναλάβετε τις ίδιες ακριβώς ενέργειες που κάνατε και στο 1^ο παράδειγμα προκειμένου να αποθηκεύσετε το ενδιάμεσο σημείο, να ρυθμίσετε την ταχύτητα και να δημιουργήσετε ή να τρέξετε το πρόγραμμα εφαρμογής.

- Αν θέσετε αριθμό επαναλήψεων 1, μόνο ένα αντικείμενο θα μετακινηθεί. Αν θέσετε αριθμό επαναλήψεων 3 θα μετακινηθούν όλα, και τα 3, αντικείμενα. Με αριθμό 0 το πρόγραμμα θα εκτελείται επαναληπτικά μέχρι είτε να πατηθεί το πλήκτρο Shift είτε να κάνετε κλικ στο **End**.



- 1 Ethernet 1 and 2
- 2 USB Device
- 3 2 x USB Host
- 4 Industrial Power Supply
- 5 Power Supply
- 6 Digital I/O, Sensors/Actuators, Soft Stop, UPS

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 504

ΔΙΑΤΑΞΗ ΠΑΡΑΓΩΓΗΣ, ΜΕΤΡΗΣΗΣ ΚΑΙ ΟΠΤΙΚΟΠΟΙΗΜΕΝΗΣ ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ ΦΥΣΙΚΩΝ ΜΕΓΕΘΩΝ

Εργαστήριο Γενικής Ηλεκτροτεχνίας
Τμήμα Ηλεκτρολόγων Μηχανικών & Τεχνολογίας Υπολογιστών
Τομέας Συστημάτων & Αυτομάτου Ελέγχου

Πανεπιστήμιο Πατρών

ΠΕΡΙΕΧΟΜΕΝΑ

- Τα μετρούμενα φυσικά μεγέθη
- Η εργαστηριακή διάταξη
- Ο προγραμματιζόμενος Λογικός Ελεγκτής
 - Τρόπος λειτουργίας του ΠΛΕ
 - Λειτουργικό Σύστημα
- Προγραμματισμός του PLC
 - Γλώσσες προγραμματισμού
- Οπτικοποιημένη παρακολούθηση
- Επικοινωνία διάταξης PLC
- Διεξαγωγή της άσκησης

Φυσικά μεγέθη διεργασιών

Σκοπός της εργαστηριακής διάταξης είναι η παραγωγή, μέτρηση και παρακολούθηση των ακολούθων έξι φυσικών μεγεθών: θερμοκρασία, πίεση, δύναμη-βάρος, ταχύτητα αέρα, γωνιακή κλίση και επιτάχυνση.

Τα φυσικά μεγέθη αφού παραχθούν ανιχνεύονται από αισθητήρες των οποίων οι έξοδοι οδηγούνται ως αναλογικά σήματα εισόδου στον Προγραμματιζόμενο Λογικό Ελεγκτή (PLC) όπου καταχωρούνται ως ψηφιακά δεδομένα. Οι τιμές των φυσικών μεγεθών μετρώνται στις ακόλουθες κλίμακες:

- Δύναμη-Βάρος: 0-100 Kg και 0.100-100 Kg
- Πίεση: 0-10 bar
- Γωνιακή Κλίση: $\pm 90^\circ$
- Επιτάχυνση: ± 5 g
- Ταχύτητα αέρα: 0-20 m/sec
- Θερμοκρασία: 0-100 °C

Οι αισθητήρες μέτρησης των φυσικών μεγεθών τροφοδοτούνται με τις εξής τάσεις:

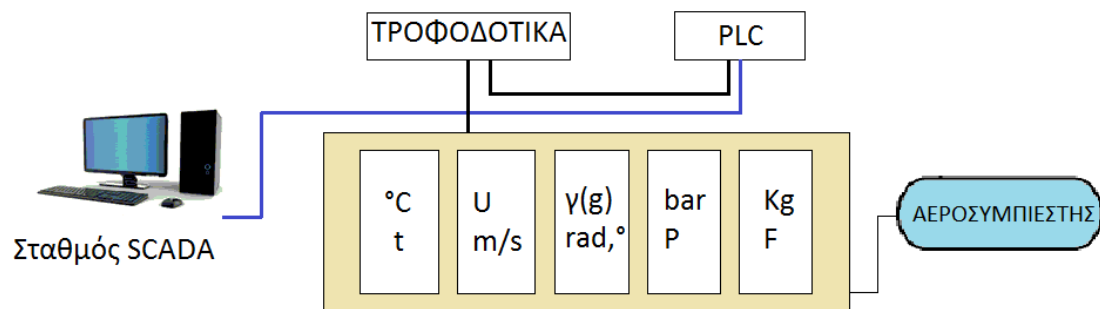
- Αισθητήρες πίεσης και ταχύτητας αέρα: 24V
- Επιταχυνσιόμετρο: 5V
- Γωνιόμετρο: 24V
- Θερμοστοιχείο: 7V
- Αισθητήρες βάρους-δύναμης: 10V

Κάθε αισθητήρας δίνει ένα εύρος τάσης στην έξοδο του που χρησιμοποιείται ως είσοδος στις αναλογικές κάρτες του προγραμματιζόμενου ελεγκτή. Το εύρος τάσης εξόδου κάθε αισθητήρα είναι:

- Αισθητήρες πίεσης: Για πίεση 0-10 bar, έξοδος 0-10V
- Αισθητήρας βάρους-δύναμης: Για βάρος 0-100Kg, έξοδος 0-10V
Για βάρος 0.02-10Kg, έξοδος 0-5.45V
- Αισθητήρας ταχύτητας αέρα: 0-20 m/sec, έξοδος 0-2V
- Θερμοστοιχείο: Για θερμοκρασία 0-100 °C, έξοδος 0-50mV
- Γωνιόμετρο: Για κλίση $\pm 90^\circ$, έξοδος 0.5-4.5V
- Επιταχυνσιόμετρο: Για επιτάχυνση ± 5 g, έξοδος 1.6-3.6V

Περιγραφή της εργαστηριακής διάταξης

Η διάταξη περιλαμβάνει το επιτραπέζιο ταμπλό στα πέντε ομοιόμορφα τμήματα του οποίου παράγονται και ανιχνεύονται τα φυσικά μεγέθη, τον προγραμματιζόμενο λογικό ελεγκτή, τον ηλεκτρονικό υπολογιστή που αποτελεί τον σταθμό SCADA, τα τροφοδοτικά ισχύος και τον αεροσυμπιεστή. Ακολουθεί το σχηματικό διάγραμμα της διάταξης:



Περιγραφή των συσκευών παραγωγής των φυσικών μεγεθών

Θερμοκρασία: Στο αντίστοιχο τμήμα υπάρχει ένα μπλοκ αλουμινίου εντός του οποίου είναι τοποθετημένα:

- Ένας θερμαντήρας ονομαστικής τάσης λειτουργίας 220V που θερμαίνει το μπλοκ αλουμινίου.
- Ένα θερμοστοιχείο μέτρησης της θερμοκρασίας του μπλοκ αλουμινίου.
- Ένας ανεμιστήρας ονομαστικής τάσης λειτουργίας 24V, για την επιτάχυνση της πτώσης της θερμοκρασίας του μπλοκ αλουμινίου.

Ακόμη, υπάρχουν δύο διακόπτες, ένας τύπου ON-OFF για την γενική τροφοδοσία του θερμαντήρα και ένας επιλογικός δύο θέσεων για την επιλογή «Λειτουργία με έλεγχο από το PLC» ή «Χειροκίνητη Λειτουργία», μία ενδεικτική λυχνία που ανάβει όταν ο θερμαντήρας είναι υπό τάση και ο μετατροπέας αναλογικού σήματος 4-20mA / 0-25A, 220V.

Επεξήγηση: Ο θερμαντήρας αποτελεί ένα ηλεκτρικό φορτίο μικρής ισχύος υπό τάση 220V. Η αναλογική έξοδος ενός PLC είναι πάντα ένα μεταβαλλόμενο σήμα χαμηλού επιπέδου ισχύος. Στην εφαρμογή αυτή είναι 4-20mA. Υπάρχει συνεπώς μία ασυμβατότητα μεταξύ του αναλογικού σήματος εξόδου και του φορτίου. Την ασυμβατότητα αυτή γεφυρώνει ο αναλογικός ρυθμιστής που δέχεται ως είσοδο 4-20mA από το PLC και παράγει έξοδο 0-25A/ 220V, την οποία εφαρμόζει στο θερμαντήρα.

Ο έλεγχος για την επίτευξη μιας επιθυμητής θερμοκρασίας του μπλοκ αλουμινίου θα είναι δύο ειδών με αντίστοιχο προγραμματισμό του PLC, δηλαδή:

- Συνεχής έλεγχος τύπου PID
- Διακριτός έλεγχος τύπου ON-OFF

Ταχύτητα αέρα: Στο αντίστοιχο τμήμα υπάρχει ένας ανεμιστήρας 220V, σταθερής ταχύτητας περιστροφής, που δημιουργεί το ρεύμα αέρος, ο αισθητήρας ανίχνευσης της ταχύτητας αέρα, ένας επιλογικός διακόπτης δύο θέσεων για την επιλογή της κατάστασης λειτουργίας του ανεμιστήρα «Λειτουργία με έλεγχο από το PLC» ή «Χειροκίνητη Λειτουργία» και μία ενδεικτική λυχνία λειτουργίας.

Ο αισθητήρας δύναται να ολισθαίνει μπροστά στο ρεύμα αέρος που δημιουργεί ο ανεμιστήρας, με αποτέλεσμα μετατοπίζοντας τον μπροστά στο σταθερό ρεύμα αέρος να έχουμε διαφορετικές ταχύτητες και επομένως διαφορετικές μετρήσεις.

Ο έλεγχος για την διατήρηση σταθερής ταχύτητας αέρα θα είναι διακριτός τύπου ON-OFF.

Επιτάχυνση-γωνιακή κλίση: Το επιταχυνσιόμετρο (έχει την μορφή και το μέγεθος ενός τρανζίστορ) είναι αναρτημένο στο άκρο ενός πακτωμένου ελατηρίου, έτσι ώστε να είναι δυνατή η εφαρμογή (με το χέρι) κρουστικών διεγέρσεων στο επιταχυνσιόμετρο και να δημιουργείται μετρήσιμη επιτάχυνση. Με ανάλογο τρόπο γίνεται και η δημιουργία και μέτρηση της γωνιακής κλίσης.

Στο ίδιο τμήμα υπάρχει και ένας περιστρεφόμενος μηχανισμός στον οποίο είναι αναρτημένο ένα κλισιόμετρο δύο αξόνων. Περιστρέφοντας τον μηχανισμό λαμβάνουμε μετρήσεις ως προς τον χ και ψ άξονα.

Πίεση: Η πίεση δημιουργείται με τη βοήθεια ενός αεροσυμπιεστή. Το κύκλωμα πεπιεσμένου αέρα περιλαμβάνει μια βαλβίδα ελέγχου ροής, μια βαλβίδα εκτόνωσης, ένα φίλτρο αέρα και ένα μανόμετρο για την οπτική ένδειξη της πίεσης.

Δύναμη: Περιλαμβάνει μια δυναμοκυψέλη, όπως καλείται ο αισθητήρας βάρους ή δύναμης, και ένα αυτόνομο μετρητή της δύναμης, ο οποίος δίνει και το αναλογικό σήμα εξόδου που οδηγείται στο PLC. Η δύναμη ασκείται με τη βοήθεια ενός ελατηρίου και του ενσωματωμένου κοχλία για την επέκταση αυτού.

Επίσης, περιλαμβάνει ένα αισθητήρα δύναμης που λαμβάνει μετρήσεις όταν εφαρμοστεί σε αυτόν δύναμη με το δάκτυλο του χεριού.

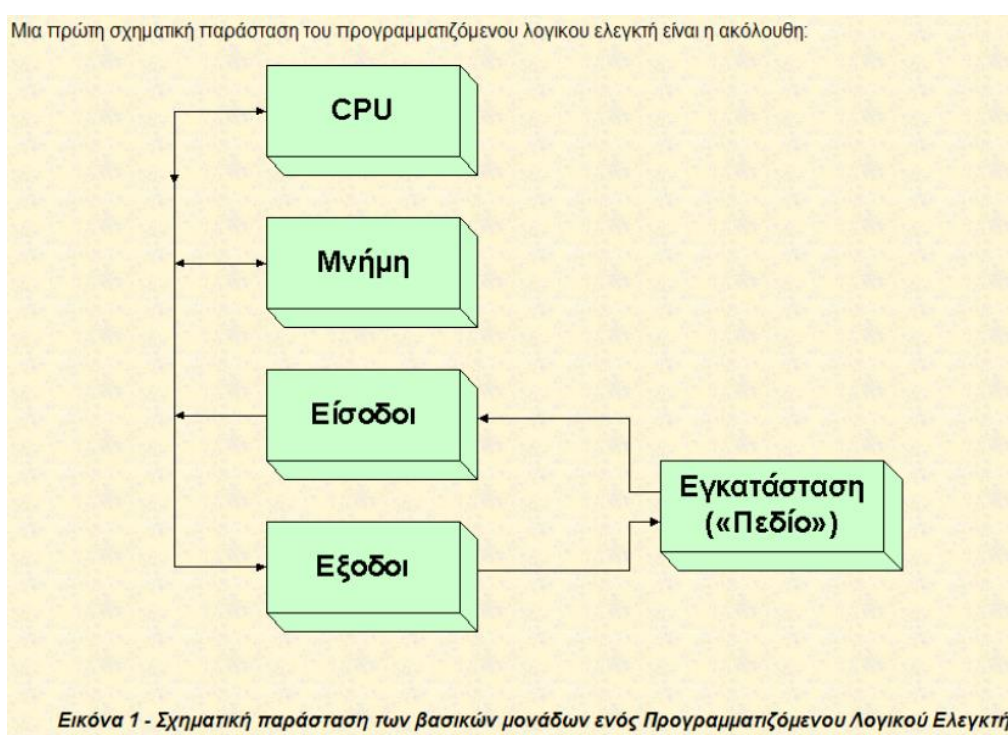
Προγραμματιζόμενος Λογικός Ελεγκτής

Ο έλεγχος της εργαστηριακής διάταξης πραγματοποιείται με τη βοήθεια ενός προγραμματιζόμενου λογικού ελεγκτή και ειδικότερα αυτού της σειράς SLC 500, της Allen-Bradley με επεξεργαστή SLC 5/03. Γι' αυτό κρίνεται σκόπιμη μια σύντομη ανασκόπηση των σημαντικότερων στοιχείων που χαρακτηρίζουν ένα PLC (Programmable Logic Controller).

Βασική Δομή του PLC

Ο Προγραμματιζόμενος Λογικός Ελεγκτής (ΠΛΕ) είναι ένας ειδικής κατασκευής βιομηχανικός υπολογιστής που χρησιμοποιείται κυρίως σε βιομηχανικό περιβάλλον για να επιτυγχάνεται η αυτόματη λειτουργία των παραγωγικών διαδικασιών. Όπως φαίνεται στο σχήμα που ακολουθεί, αποτελείται από μία κεντρική μονάδα επεξεργασίας (CPU), τη μνήμη, τις κάρτες εισόδων και τις κάρτες εξόδων. Μέσω των τελευταίων, ο ΠΛΕ επικοινωνεί με το «πεδίο» των αισθητήρων, των ενεργοποιητών και γενικά των μηχανών (εγκατάσταση). Ο ΠΛΕ λειτουργεί βάσει ενός προγράμματος που είναι αποθηκευμένο στη μνήμη του. Το πρόγραμμα αυτό είναι ένα σύνολο οδηγιών (εντολών) που καθορίζουν τον τρόπο ή τη «λογική» λειτουργίας των μηχανών. Το πρόγραμμα αυτό αναπτύσσεται από το χρήστη. Η λειτουργία του ΠΛΕ ως υπολογιστικής μηχανής καθορίζεται από ένα άλλο πρόγραμμα (Λειτουργικό Σύστημα), που επίσης είναι αποθηκευμένο στη μνήμη του και αναπτύσσεται από τον κατασκευαστή του ΠΛΕ.

Ένας ΠΛΕ χρησιμοποιείται οπουδήποτε υπάρχει η έννοια του αυτοματισμού και οι βασικότεροι λόγοι που τον έχουν καταστήσει τόσο διαδεδομένο είναι η ευελιξία και η αξιοπιστία που παρέχει, σε συνδυασμό με την εύκολη συναρμολόγηση και το



σχετικά μικρό κόστος.

Η παραπάνω σχηματική παράσταση, σε συνδυασμό με το ακόλουθο σχήμα, μπορούν να δημιουργήσουν μία πρώτη άποψη σχετικά με τη δομή και τη λειτουργία του προγραμματιζόμενου λογικού ελεγκτή.

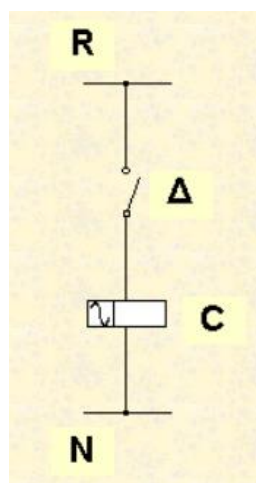


Εικόνα 2 - Δομή ενός προγραμματιζόμενου λογικού ελεγκτή. Σε μία βάση στήριξης τοποθετούνται οι μονάδες που απαιτεί η κάθε συγκεκριμένη εφαρμογή. Δηλαδή, πέραν των μονάδων τροφοδοσίας και CPU, ο ΠΛΕ συντίθεται από μονάδες που επιλέγει ο χρήστης.

Τρόπος λειτουργίας του ΠΛΕ-Λειτουργικό σύστημα

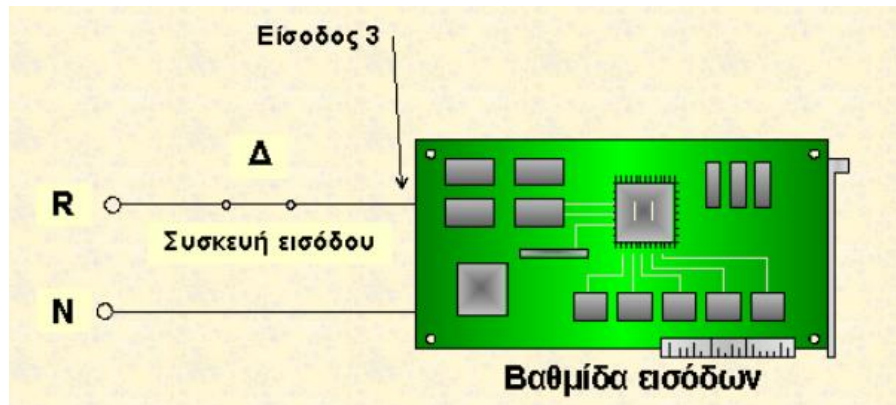
Η CPU και η μνήμη ενός ΠΛΕ εκτελούν τα καθήκοντα τα οποία συνήθως εκτελούνται σε ένα κοινό υπολογιστή, δηλαδή ανάκτηση και επεξεργασία των δεδομένων εισόδου που προωθούνται σε ειδικές περιφερειακές μονάδες εξόδου. Αντίθετα, διαφορετικός είναι ο ρόλος των καρτών εισόδων/εξόδων. Ένας υπολογιστής συνδέεται με ποικίλα περιφερειακά (πληκτρολόγιο, ποντίκι, εκτυπωτής, LAN κτλ.) βάση πρωτοκόλλων επικοινωνίας που ανταλλάσσουν σήματα ψηφιακού τύπου πάντοτε με τα ίδια ηλεκτρικά χαρακτηριστικά. Στην συγκεκριμένη διάταξη η σύνδεση υπολογιστή-ΠΛΕ επιτυγχάνεται με τη χρήση του πρωτοκόλλου RS232.

Για την καλύτερη κατανόηση του τρόπου λειτουργίας των ΠΛΕ, ας θεωρήσουμε το παράδειγμα του απλού κυκλώματος:

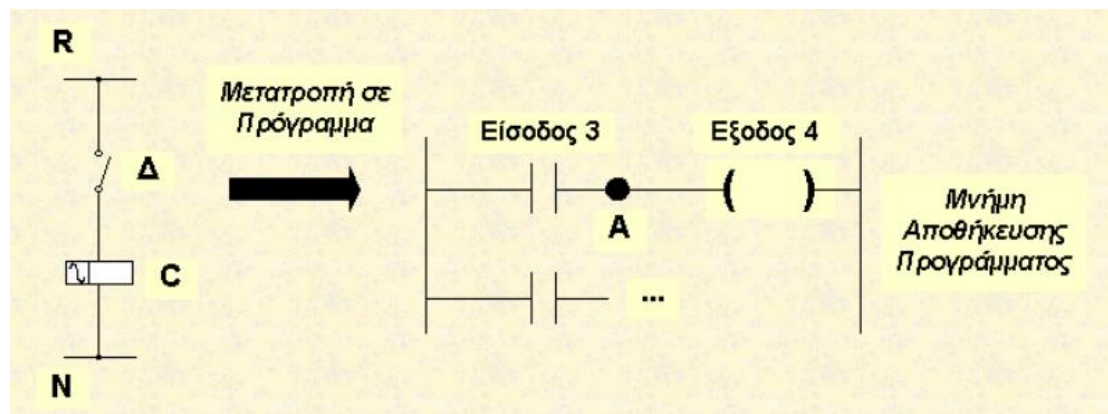


Ο διακόπτης Δ είναι εξ' αρχής κλειστός. Θέτουμε τον ΠΛΕ σε κατάσταση λειτουργίας (RUN) για να παρακολουθήσουμε τα βήματα που πραγματοποιεί ο επεξεργαστής

βάση του λειτουργικού συστήματος. Η βαθμίδα εισόδων, οδηγούμενη από τον μικροεπεξεργαστή, διαβάζει όλες τις εισόδους, άρα και την είσοδο 3.



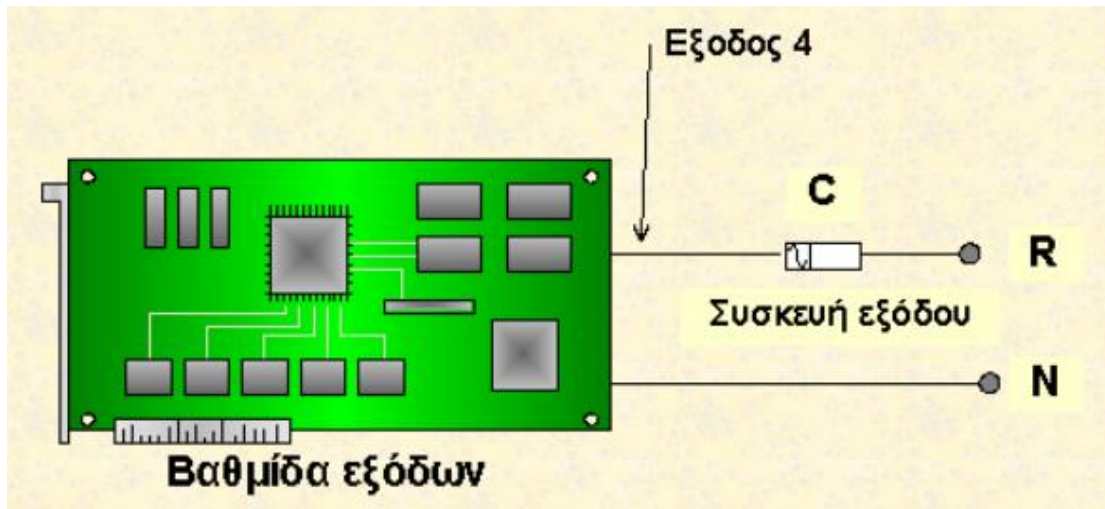
Αυτό σημαίνει ότι ανιχνεύει αν σε κάθε είσοδο υπάρχει τάση ή όχι. Επειδή ο διακόπτης Δ είναι κλειστός, στην είσοδο 3 εφαρμόζεται τάση. Η τάση αυτή υποβιβάζεται και μετατρέπεται από τη βαθμίδα εισόδων σε ασθενές TTL σήμα. Το γεγονός της ύπαρξης του TTL σήματος αποθηκεύεται σαν «λογικό 1» στη μνήμη απεικόνισης εισόδων στη θέση που αντιστοιχεί στην είσοδο 3. Στις εισόδους που δεν εφαρμόζεται τάση αποθηκεύεται αντίστοιχα «λογικό 0». Αφού αποθηκευτούν οι καταστάσεις όλων των εισόδων, ο μικροεπεξεργαστής ξεκινά την εκτέλεση του προγράμματος.



Η εκτέλεση της εντολής του πρώτου κλάδου “ -||- είσοδος 3” σημαίνει εξ’ ορισμού της αντίστοιχης γλώσσας προγραμματισμού, ότι το σημείο A είναι σε λογικό 1, αν η είσοδος 3 είναι ενεργοποιημένη και σε λογικό 0 αν είναι απενεργοποιημένη. Για να εκτελέσει συνεπώς ο επεξεργαστής την παραπάνω εντολή χρειάζεται να ξέρει ποια είναι η κατάσταση της εισόδου 3. Στη συνέχεια ο μικροεπεξεργαστής εκτελεί την επόμενη εντολή “ -()- είσοδος 4”. Αυτή σημαίνει, επίσης εξ’ ορισμού, ότι αν το σημείο A είναι σε λογικό 1, τότε η έξοδος 4 πρέπει να ενεργοποιηθεί, ενώ αν είναι σε λογικό 0, τότε πρέπει να απενεργοποιηθεί.

Η ενεργοποίηση ή μη μιας εξόδου ως αποτέλεσμα εκτέλεσης εντολής του προγράμματος σημαίνει καταχώρηση λογικού 1 ή 0 αντίστοιχα στη θέση μνήμης της

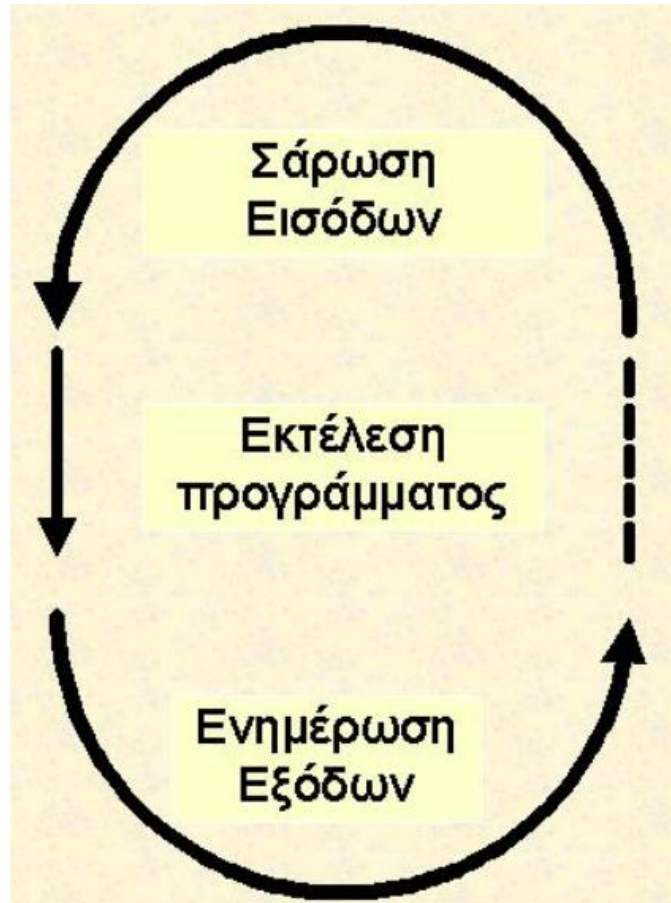
εξόδου 4. Σύμφωνα με τις συνθήκες του παραδείγματος, η έξοδος 4 πρέπει να ενεργοποιηθεί και να καταχωρηθεί «λογικό 1» στην αντίστοιχη θέση μνήμης.



Κατά τον ίδιο τρόπο εκτελούνται όλοι οι υπόλοιποι κλάδοι του προγράμματος όταν υπάρχουν. Όταν εκτελεστεί ολόκληρο το πρόγραμμα, τότε στη μνήμη απεικόνισης εξόδων έχει καταχωρηθεί η κατάσταση 0 ή 1 όλων των εξόδων. Στη συνέχεια, ο μικροεπεξεργαστής μεταφέρει το περιεχόμενο της μνήμης απεικόνισης εξόδων στη βαθμίδα εξόδων. Άρα στην έξοδο 4 της βαθμίδας εξόδων μεταφέρεται «λογικό 1», δηλαδή ένα TTL σήμα που εντός της βαθμίδας μετατρέπεται σε σήμα ελέγχου ενός διακοπτικού στοιχείου, μέσω του οποίου ενεργοποιείται η συσκευή εξόδου, δηλαδή το ρελέ C. Στη συνέχεια, ο μικροεπεξεργαστής επαναλαμβάνει την ανάγνωση των εισόδων, εκτελεί εκ νέου το πρόγραμμα, ενημερώνει πάλι τις εξόδους κ.ο.κ. Αυτή η συνεχής κυκλική λειτουργία των ΠΛΕ είναι γνωστή σαν **σάρωση**. Ο χρόνος σάρωσης ενός ΠΛΕ εκφράζει την ταχύτητα λειτουργίας του και πρέπει να είναι της τάξης των μερικών ms ή μικρότερος.

Λειτουργικό σύστημα-Εκκίνηση

Όπως είδαμε και στον τρόπο λειτουργίας, ο ΠΛΕ χαρακτηρίζεται από ένα κυκλικό τρόπο λειτουργίας, ο οποίος μπορεί σχηματικά να αποδοθεί ως εξής:



Όσον αφορά την εκκίνηση του ελεγκτή, οι σπουδαιότερες ενέργειες που κάνει το λειτουργικό σύστημα ενός PLC κατά την εκκίνηση λειτουργίας του είναι:

- Σβήνεται η μνήμη απεικόνισης εισόδων-εξόδων
- Όλες οι εισοδοί παίρνουν σήμα «0»
- Καταχωρείται το πόσες μονάδες εισόδων-εξόδων είναι τοποθετημένες και σε ποια θέση
- Γίνεται έλεγχος της μνήμης
- Δημιουργείται και αποταμιεύεται η λίστα των “δικτυωμάτων” από τα οποία αποτελείται το πρόγραμμα του χρήστη

Προγραμματισμός του PLC

Ο προγραμματισμός ενός ΠΛΕ αποτελεί τη διαδικασία εισαγωγής σ' αυτόν των οδηγιών βάσει των οποίων θα επιτυγχάνεται η αυτόματη λειτουργία ενός ελεγχόμενου συστήματος. Οι οδηγίες αυτές δεν είναι παρά «εντολές» προγράμματος, γραμμένες βάσει μιας γλώσσας προγραμματισμού που είναι κατανοητή και φιλική στο χρήστη. Οι εντολές αυτές μεταφράζονται σε γλώσσα μηχανής που είναι κατανοητή στο μικροεπεξεργαστή του ΠΛΕ και εκτελούνται στη σειρά η μία μετά την άλλη. Ο ΠΛΕ μπορεί να ελέγχει μόνο την συντακτική και λογική ορθότητα των εντολών του χρήστη, αλλά όχι την ορθότητα λειτουργίας του ελεγχόμενου συστήματος. Ο ΠΛΕ εκτελεί πιστά την κάθε εντολή του προγράμματος. Επομένως, το πρόγραμμα του χρήστη είναι αυτό που καθορίζει στον ΠΛΕ όλες τις ενέργειες που πρέπει να κάνει και την σειρά με την οποία πρέπει να τις κάνει, ώστε να επιτυγχάνεται η επιθυμητή αυτόματη λειτουργία.

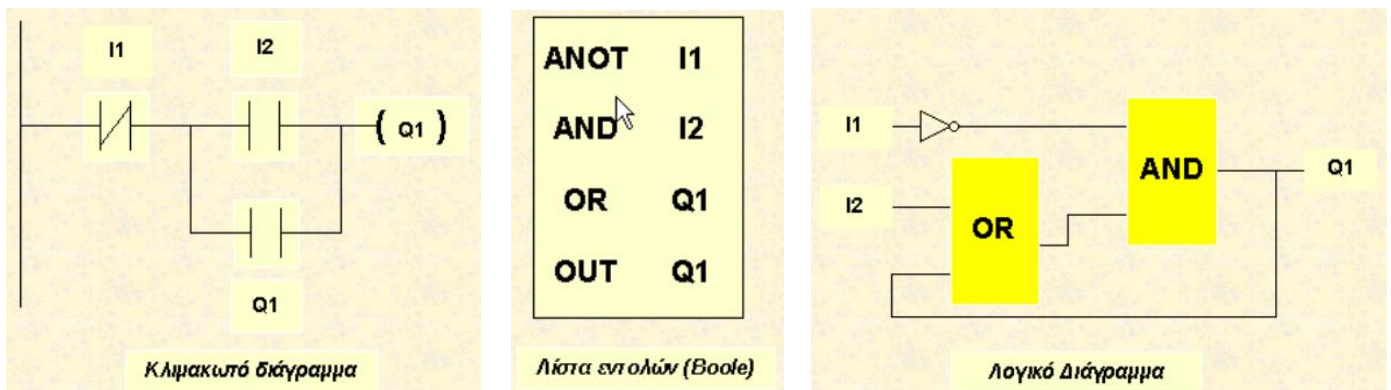
ΠΡΟΣΟΧΗ!

Ο προγραμματισμός ενός επεξεργαστή δεν μπορεί να τον κάνει να λύσει ένα πρόβλημα, αν ο ίδιος ο προγραμματιστής δεν ξέρει πώς να λύσει το συγκεκριμένο πρόβλημα.

Γλώσσες Προγραμματισμού PLC

Οι ΠΛΕ δεν προγραμματίζονται με μια διεθνώς τυποποιημένη γλώσσα προγραμματισμού. Αντίθετα, υπάρχουν διάφορες μορφές γλωσσών προγραμματισμού, ποικίλουν από εταιρία σε εταιρία, φέρουν διάφορα ονόματα ακόμη κι αν είναι παρόμοιες και τέλος υπάρχει ασυμβατότητα μεταξύ παρόμοιων γλωσσών προγραμματισμού που έχουν αναπτυχθεί από διαφορετικούς κατασκευαστές. Παρά την έλλειψη τυποποίησης στο θέμα του προγραμματισμού, μπορούμε να διακρίνουμε τρεις βασικές γλώσσες προγραμματισμού που είναι: α) η γλώσσα κλιμακωτού διαγράμματος, β) η γλώσσα Boolean, γ) η γλώσσα λογικών στοιχείων. Η πιο δημοφιλής γλώσσα από αυτές είναι η γλώσσα κλιμακωτού διαγράμματος (γλώσσα επαφών), αφού προσομοιάζει πολύ με το κλασικό κύκλωμα αυτοματισμού. Αυτός ήταν άλλωστε ο λόγος για την υιοθέτηση της από όλες σχεδόν τις κατασκευάστριες εταιρίες, αφού κατ' αυτόν τον τρόπο ήταν πιο

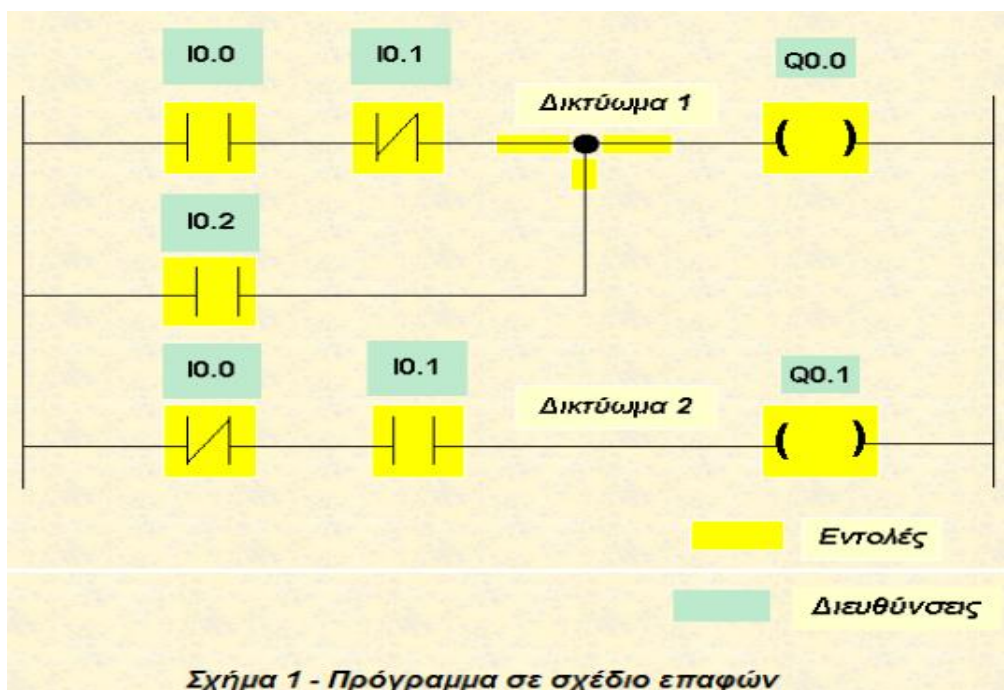
εύκολη η διάδοση και υιοθέτηση των ΠΛΕ από τους παλαιούς μηχανικούς που δεν ήταν εξοικειωμένοι με τον προγραμματισμό. Στο επόμενο σχήμα φαίνεται η γενική μορφή των τριών γλωσσών προγραμματισμού.



Το συμβατικό κύκλωμα αυτοματισμού για start-stop λειτουργία ενός κινητήρα έχει γραφεί ως πρόγραμμα για έναν ΠΛΕ και στις τρεις συμβατικές γλώσσες προγραμματισμού. I1 και I2 είναι οι εισοδοι του ελεγκτή, ενώ Q1 είναι η έξοδος αυτού, η οποία ενεργοποιεί το ρελέ ισχύος το οποίο με τη σειρά του τροφοδοτεί τον κινητήρα με ισχύ.

❖ Γλώσσα επαφών (Ladder Diagram Programming – LAD)

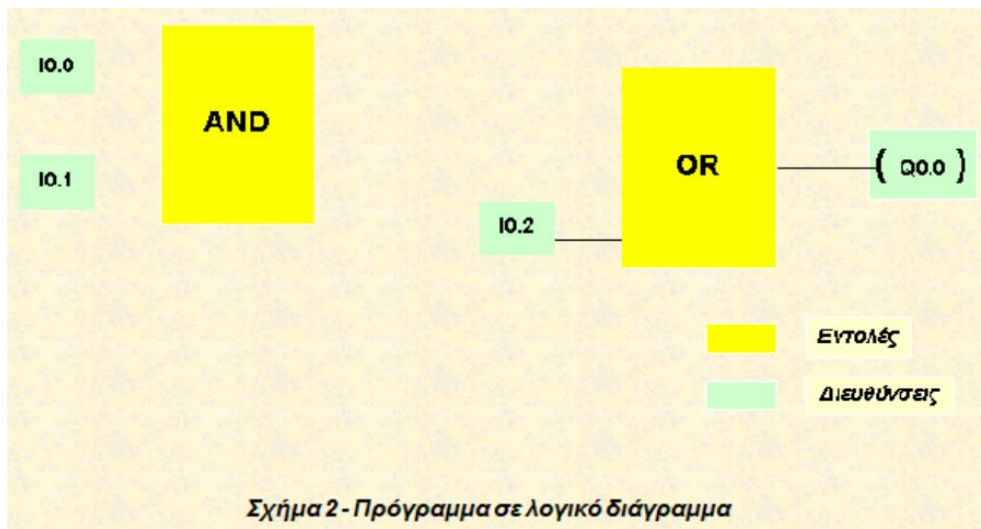
Η γλώσσα επαφών LAD είναι μια γραφική γλώσσα που συναντάται σε όλους του μικρού και μεσαίου μεγέθους ΠΛΕ. Το πρόγραμμα απαρτίζεται από δικτύωματα (networks) κάθε ένα από τα οποία περιγράφεται από μία λογική εξίσωση Boole. Στο σχήμα 1 φαίνεται ένα πρόγραμμα σε γλώσσα επαφών. Κάθε δικτύωμα περιλαμβάνει τις εντολές και τις αντίστοιχες διευθύνσεις. Κάθε στοιχείο σε κίτρινο φόντο αποτελεί την εντολή, ενώ σε πράσινο φόντο φαίνεται η αντίστοιχη διεύθυνση. Η γλώσσα επαφών περιέχει και σύνθετες εντολές, όπως η ενεργοποίηση ενός χρονοδιακόπτη ή η κλήση μιας υπορουτίνας.



Σχήμα 1 - Πρόγραμμα σε σχέδιο επαφών

❖ **Γλώσσα Λογικών Στοιχείων/ Λογικό Διάγραμμα(Function Block Diagram Programming-FBD)**

Πρόκειται επίσης για μια γραφική γλώσσα με ανάλογα χαρακτηριστικά όπως αυτά της LAD. Η μόνη τους διαφορά έγκειται στην υλοποίηση του προγράμματος. Οι εντολές έχουν τη μορφή λογικών στοιχείων της ψηφιακής λογικής και υλοποιούν τις βασικές λειτουργίες των λογικών πυλών AND, OR και NOT. Η γλώσσα λογικού διαγράμματος περιέχει και αυτή όλες τις απαιτούμενες σύνθετες εντολές που συνίστανται από ένα γραφικό στοιχείο και ένα κωδικόγραμμα που εκφράζει τη λειτουργία.



❖ **Γλώσσα Boole ή Λίστας εντολών (Statement List Programming – STL)**

Η γλώσσα Boole δεν μοιάζει με τις άλλες δύο. Οι εντολές της γλώσσας αυτής υλοποιούνται αποκλειστικά με αλφαβητικά σύμβολα και συμπληρώνονται από τις διευθύνσεις, όπως και στις υπόλοιπες γλώσσες προγραμματισμού. Το πρόγραμμα χωρίζεται σε τμήματα (αντίστοιχα των δικτυωμάτων) και είναι δυνατόν να περιέχει υπορουτίνες.

Εντολές	Διευθύνσεις
LD	I0.1
AN	I0.1
O	I0.2
=	Q0.0

Σχήμα 3 - Πρόγραμμα σε λίστα εντολών

Οπτικοποιημένη Παρακολούθηση

SIMATIC WINCC FLEXIBLE 2008

Για τον σχεδιασμό του συστήματος SCADA, δηλαδή του εποπτικού ελέγχου, της συλλογής δεδομένων και της διεπαφής ανθρώπου-μηχανής (HMI) χρησιμοποιήθηκε το πρόγραμμα Wincc Flexible της Siemens. Το πρόγραμμα αυτό προσφέρει δυνατότητες όπως η απεικόνιση των δεδομένων μέσω γραφικών παραστάσεων και αναπαραστάσεων, αλλά και έλεγχο των διεργασιών του αυτοματισμού μέσα από εντολές που μπορεί να δώσει ο χειριστής, οι οποίες αποστέλλονται στο PLC το οποίο αντιδρά ανάλογα.

Ακολουθεί μία συνοπτική παρουσίαση των menu που διαθέτει το Wincc Flexible για τον σχεδιασμό ενός συστήματος SCADA.

❖ **Project View**

Το Project View αποτελεί το βασικό menu κατά τον σχεδιασμό ενός Project. Σε αυτό εμφανίζονται με δομή δέντρου όλα τα διαθέσιμα στοιχεία για την δημιουργία και διαμόρφωση ενός Project. Στοιχεία όπως:

- **Screens**
Οθόνες που εισάγει ο σχεδιαστής (τα παράθυρα του προγράμματος) ανάλογα με τις ανάγκες του συστήματος.
- **Template**
Οποιοδήποτε αντικείμενο εισαχθεί στο template είναι ορατό σε όλες τις οθόνες του προγράμματος.
- **Connections**
Εδώ ορίζεται η σύνδεση του συστήματος SCADA με το PLC.
Επιλέγεται η HMI συσκευή (ο υπολογιστής δηλαδή ή το panel), το δίκτυο επικοινωνίας και τέλος η συσκευή PLC.
- **Tags**
Εδώ ορίζονται όλες οι μεταβλητές (tags) του συστήματος. Επιλέγεται το είδος (integer, floating, κτλ), το όνομα, και αν είναι εσωτερικές μεταβλητές (αφορούν δηλαδή μόνο το SCADA) ή αν συνδέονται με το PLC.
- **Alarm Management**
Εδώ ορίζεται το μήνυμα που θα εμφανιστεί στην οθόνη στην περίπτωση ενεργοποίησης ενός alarm, η μεταβλητή και η τιμή ενεργοποίησης.
- **Alarms**
Υποδεικνύουν την αλλαγή κατάστασης (discrete alarm) ή την παραβίαση των ορίων μίας μεταβλητής (analog alarm).

❖ Work Area

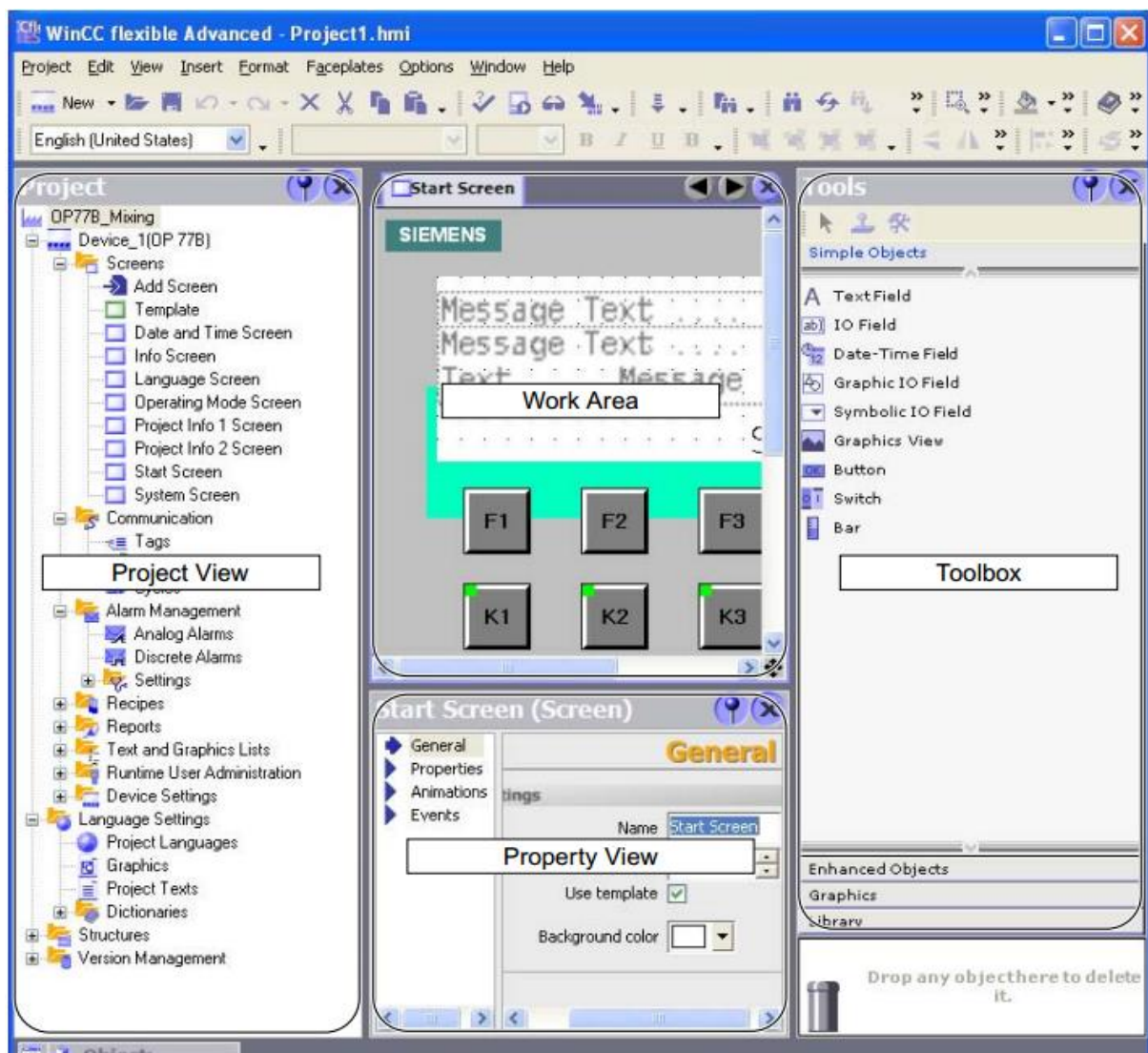
Στην «περιοχή εργασίας» γίνεται η επεξεργασία του Project. Στην περιοχή αυτή βλέπουμε και μπορούμε να επεξεργαστούμε το περιεχόμενο του Project View.

❖ Property View

Χρησιμοποιείται για την επεξεργασία ιδιοτήτων των αντικειμένων. Ανάλογα με το αντικείμενο που επιλέγουμε το περιεχόμενο του property view αλλάζει. Οι βασικότερες όμως ιδιότητες των αντικειμένων είναι το χρώμα, οι διαστάσεις, η τοποθεσία, η μορφοποίηση του κειμένου, η εκτέλεση εντολών αλλά και η κίνηση ή η ορατότητα ενός αντικειμένου ανάλογα με την τιμή κάποιας μεταβλητής.

❖ Toolbox

Περιλαμβάνει αντικείμενα όπως εικόνες, πεδία εμφάνισης και εισαγωγής τιμών και κειμένου, κουμπιά κτλ, τα οποία μπορούμε να σύρουμε στην περιοχή εργασίας και να τα επεξεργαστούμε αλλάζοντας τις ιδιότητες τους από το menu Property View.



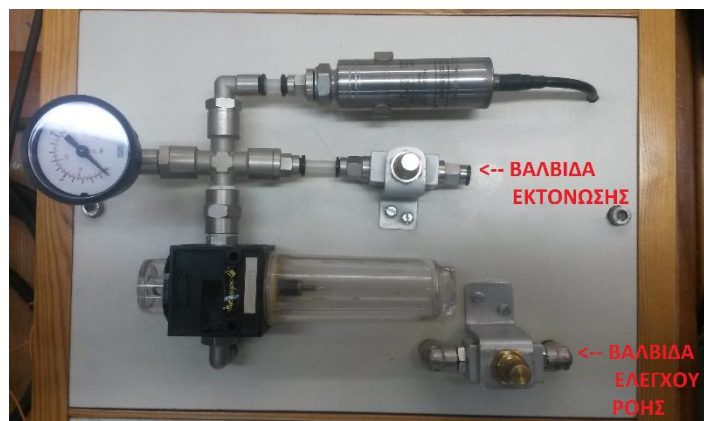
Διεξαγωγή της άσκησης

Ελέγξτε ότι είναι υπό τάση και λειτουργούν κατά σειρά:

- Ο προγραμματιζόμενος λογικός ελεγκτής AB-SLC 500
 - Τα τροφοδοτικά της διάταξης
 - Ο Η/Υ
1. Τρέξτε το αρχείο askisi.Device_1, που αποτελεί το runtime αρχείο του προγράμματος SCADA.
 2. **Παράθυρο βάρους:** Ενεργοποιήστε το αντίστοιχο παράθυρο.
 - Προκαλέστε μεταβολές στις μετρήσεις, γυρνώντας τον κοχλία της δυναμοκυψέλης και εφαρμόζοντας πίεση με το δάκτυλο στον αισθητήρα δύναμης. Παρατηρήστε τις αντίστοιχες εικονικές μετρήσεις και αποθηκεύστε τις πιο χαρακτηριστικές εξ' αυτών με (Printscreen)



3. **Παράθυρο πίεσης:** Ενεργοποιήστε το αντίστοιχο παράθυρο.
 - Ανοίξτε την βαλβίδα ελέγχου ροής και έπειτα ανοιγοκλείστε την βαλβίδα εκτόνωσης ώστε να προκαλέσετε μεταβολές στην πίεση και στις μετρήσεις. Παρατηρήστε τις αντίστοιχες εικονικές μετρήσεις και αποθηκεύστε τις πιο χαρακτηριστικές εξ' αυτών με (Printscreen)



4. **Παράθυρο κλίσης:** Ενεργοποιήστε το αντίστοιχο παράθυρο.
- Προκαλέστε σταθερές γωνιακές κλίσεις και στα δύο όργανα. Παρατηρήστε τις αντίστοιχες εικονικές μετρήσεις.
5. **Παράθυρο επιτάχυνσης:** Ενεργοποιήστε το αντίστοιχο παράθυρο.
- Προκαλέστε κρουστικές επιταχύνσεις και παρατηρήστε τις αντίστοιχες εικονικές μετρήσεις.



6. **Παράθυρο ταχύτητας αέρα:** Ενεργοποιήστε το αντίστοιχο παράθυρο. Ο επιλογικός διακόπτης (ΔΕ) που υπάρχει πάνω στο ταμπλό της διάταξης έχει δύο θέσεις, τη «Χ», όπου ο ανεμιστήρας λειτουργεί μόνιμα, και την «Α», όπου η λειτουργία του εξαρτάται από το PLC (ON-OFF έλεγχος).
- Με τον ΔΕ στη θέση «Χ» ο ανεμιστήρας λειτουργεί μόνιμα και συνεπώς μπορούμε να υποθέσουμε ότι δημιουργεί ένα σταθερό ρεύμα αέρος. Μετακινήστε τον αισθητήρα μπροστά από το σταθερό ρεύμα αέρος, ώστε να μετρά διαφορετικές τιμές ταχύτητας του αέρα. Παρατηρήστε τις αντίστοιχες εικονικές μετρήσεις και αποθηκεύστε με (Printscreen)
 - Με τον ΔΕ στην θέση «Α» ενεργοποιήστε το διακριτό έλεγχο (ON-OFF) από τον εικονικό διακόπτη και θέστε στα εικονικά όργανα επιλογής τιμής ΑΝΩ και ΚΑΤΩ ορίου την ίδια τιμή. Παρατηρήστε τον τρόπο λειτουργίας του ανεμιστήρα και τις αντίστοιχες εικονικές μετρήσεις και αποθηκεύστε με (Printscreen)
 - Επαναλάβετε το προηγούμενο βήμα θέτοντας διαφορετικά ΑΝΩ και ΚΑΤΩ όρια.



7. **Παράθυρο θερμοκρασίας:** Ενεργοποιήστε το αντίστοιχο παράθυρο. Για να λειτουργήσει ο θερμοαντήρας, πρέπει ο γενικός διακόπτης Δ. ON-OFF που βρίσκεται πάνω στο ταμπλό θερμοκρασίας της διάταξης να είναι στη «θέση ON». Ο επιλογικός διακόπτης που επίσης βρίσκεται πάνω στο ταμπλό θερμοκρασίας έχει δύο θέσεις με αντίστοιχες ενδείξεις «X» και «A». Ο θερμοαντήρας λειτουργεί συνεχώς και ανεξάρτητα από τον ΠΛΕ, όταν ο ΔΕ είναι στη θέση «X».

- Με τον ΔΕ στην θέση «Α», θέστε στο εικονικό όργανο επιλογής τιμής Setpoint την τιμή 40°C, και τις τιμές Kc=1000, Ki=800, Kd=10. Ενεργοποιήστε τον έλεγχο PID. Παρατηρήστε τις αντίστοιχες εικονικές μετρήσεις και αποθηκεύστε αυτές με (Printscreen)
- Θέστε στο εικονικό όργανο επιλογής τιμής Setpoint την τιμή 45°C, και τις τιμές Kc=1000, Ki=800, Kd=0. Ενεργοποιήστε το AUTO-FAN. Παρατηρήστε και αποθηκεύστε την απόκριση.
- Απενεργοποιήστε τον PID έλεγχο. Θέστε στο εικονικό όργανο επιλογής τιμής Temperature την τιμή 50°C, και ενεργοποιήστε τον ON-OFF έλεγχο. Παρατηρήστε και αποθηκεύστε την απόκριση.
- Απενεργοποιήστε τον ON-OFF έλεγχο και το AUTO-FAN και κλείστε το runtime (EXIT).

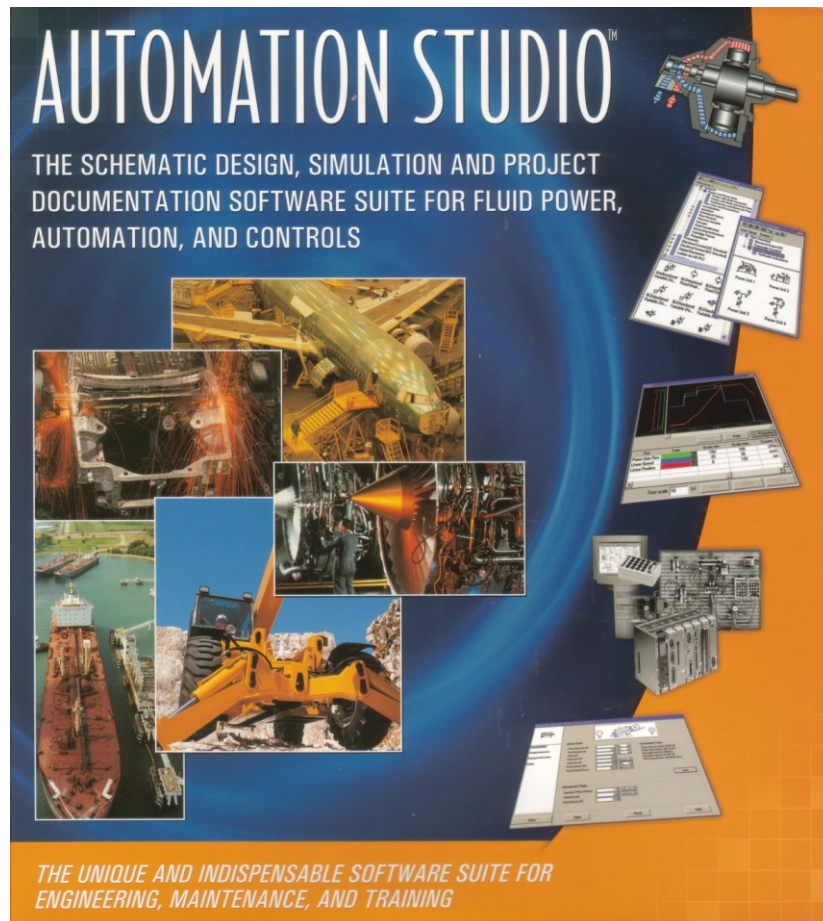


8. **Παράθυρο προγραμματισμού:** : Τρέξτε το αρχείο askisi. Μεταβείτε στο screen ΠΑΡΑΘΥΡΟ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ.

- Με τη βοήθεια του toolbox δημιουργήστε ένα αντικείμενο (πχ τετράγωνο), και αλλάξτε τις ιδιότητες του (Property View) ώστε να είναι ορατό (Animations→Visibility) όταν η είσοδος I:1/1 (tag: TEST_TAG) είναι ενεργοποιημένη.
- Χρησιμοποιείτε τον αισθητήρα ταχύτητας αέρα, είσοδος I:4.0 (tag: AIR_INPUT) για να μετακινήσετε το αντικείμενο προς μία κατεύθυνση (π.χ. Horizontal movement). Χρησιμοποιήστε εύρος εισόδου: 0-1200.

ΑΣΚΗΣΗ 505

Εξάσκηση στο λογισμικό αυτοματισμού “Automation Studio”



Εργαστήριο Γενικής Ηλεκτροτεχνίας
Τμήμα Ηλεκτρολόγων Μηχανικών & Τεχνολογίας Υπολογιστών
Τομέας Συστημάτων & Αυτομάτου Ελέγχου

Πανεπιστήμιο Πατρών , 2006

ΑΝΤΙΚΕΙΜΕΝΟ

Το αντικείμενο της άσκησης είναι η εξοικείωση των φοιτητών με το λογισμικό **Automation Studio** με τη βοήθεια του οποίου μπορούμε,

1. Να συνθέσουμε διατάξεις υδραυλικού, πνευματικού και ηλεκτροπνευματικού αυτοματισμού
2. Να εξομοιώσουμε τη λειτουργία τους με πραγματική δυναμική απεικόνιση (κίνηση) στην οθόνη του υπολογιστή
3. Να συμπληρώσουμε τη μελέτη-σχεδίαση ενός συστήματος αυτοματισμού με παραγωγή φύλλων τεκμηρίωσης π.χ. διαστασιολόγια, διαγράμματα συρμάτωσης κλπ., και
4. Να κάνουμε χρήση διαφόρων άλλων “εργαλείων” ή λειτουργικών συναρτήσεων που μας προσφέρει όπως για παράδειγμα να προγραμματίσουμε ένα σύστημα αυτοματισμού σε γλώσσα SFC (Sequential Function Chart).

Κατά τη διεξαγωγή της εργαστηριακής άσκησης εμείς θα περιοριστούμε στη σχεδίαση-εξομοίωση ηλεκτροπνευματικών διατάξεων.

ΕΝΕΡΓΕΙΕΣ ΠΡΟΕΤΟΙΜΑΣΙΑΣ ΓΙΑ ΤΗΝ ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ

1. Συνδεθείτε μέσω του διαδικτύου με την ιστοσελίδα της Automation Studio (Εικ.1) στη διεύθυνση www.automationstudio.com

Εικ.1

AUTOMATION STUDIO
Because today, it's not just about schematic design but how you can easily troubleshoot, validate and modelize complete technology projects

EDUCATIONAL VERSION

See a General overview of **AUTOMATION STUDIO...** download movie samples !

PROFESSIONAL EDITION
Schematic Design, Simulation and Project Documentation for Fluid Power, Automation, and Controls
The Schematic design, simulation, and project documentation software suite for fluid power, automation, and controls. Automation Studio is intended to be used by engineers and corporate trainers in a wide variety of related fields. If you are involved in the design, maintenance, and training of industrial or mobile applications, discover why Automation Studio will become an indispensable part of your workflow and help your organization increase both quality and productivity.

EDUCATIONAL EDITION
The Worldwide Standard for Teaching Fluid Power and Automation Technologies
Automation Studio allows educators to quickly draw circuits and perform demonstrations. By using schematics, help files, cut-away animations, and data monitoring, it becomes an indispensable tool that helps explaining equipment and controls operation from the component up to the system level. Not only can the student learn by observing the states of switches and components, but also by observing the color schemes of symbols and using the variable plotting functions.

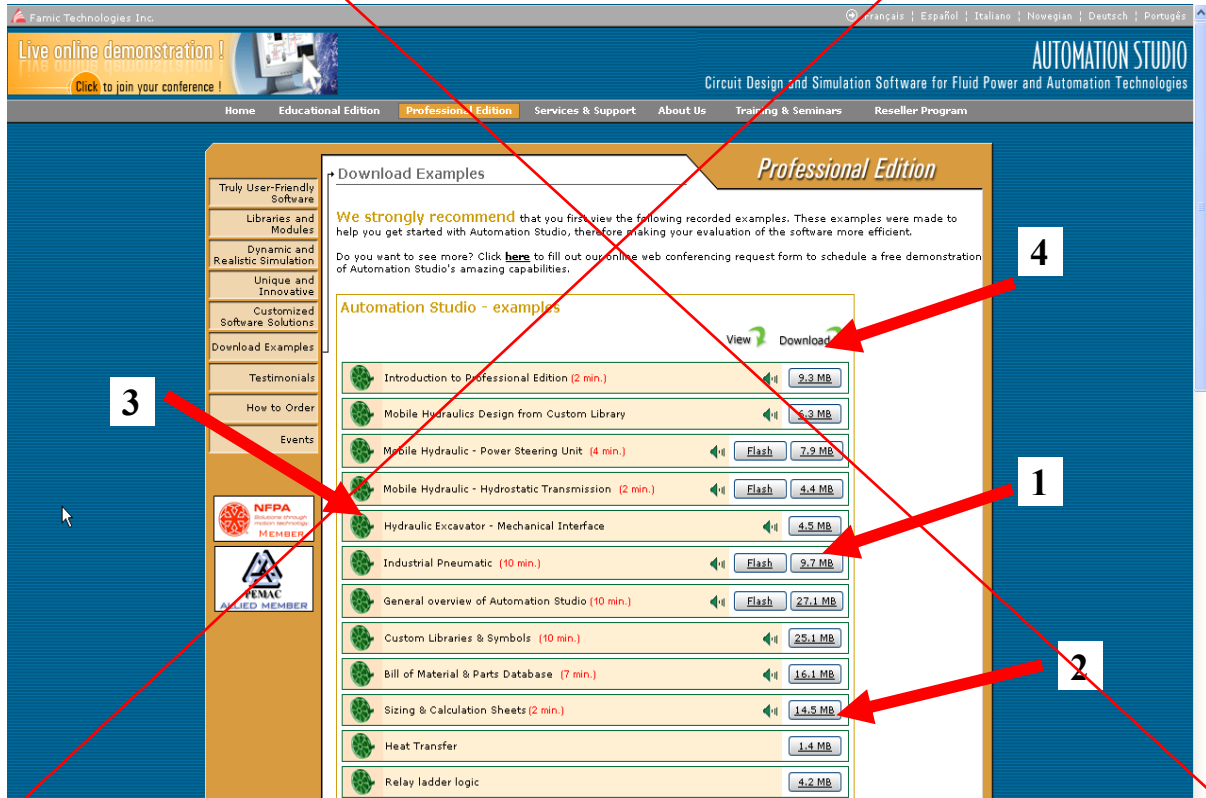
What's New
- AQPPDT
- APEMIQ
- Worlddidac
- APEEMQ
- NTPN

Quick Menu
- [Download Movie Samples](#)
- [Download Brochures](#)
- [Service Releases](#)
- [On-Line Technical Support](#)
- [Request a Demo](#)

Careers
- [Job opportunities](#)

Υπάρχουν στο youtube

2. Επιλέξτε το “κατέβασμα παραδειγμάτων” όπως φαίνεται στην Εικ.1, **Download Movie Samples**
3. Κατεβάστε στον υπολογιστή σας και στη συνέχεια τρέξτε (αρχαία .exe) κατά σειρά τα ακόλουθα παραδείγματα όπως φαίνονται στην Εικ.2.



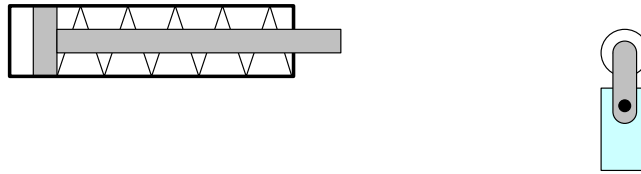
Εικ.2

1. **General overview of Automation Studio** (Automation_Studio_General_Overview.zip)
2. **Relay Ladder Logic** (HYD-ELEC.zip)
3. **Industrial Pneumatics** (Movie-pro-4.zip)
4. **Introduction to Professional Edition** (Movie-pro-1.zip)

4. Κατεβάστε οποιοδήποτε άλλο παράδειγμα νομίζετε ότι θα σας βοηθήσει να προετοιμαστείτε καλύτερα για τη διεξαγωγή της άσκησης.

ΔΙΕΞΑΓΩΓΗ ΑΣΚΗΣΗΣ – ΕΦΑΡΜΟΓΕΣ ΠΟΥ ΘΑ ΑΝΑΠΤΥΞΕΤΕ

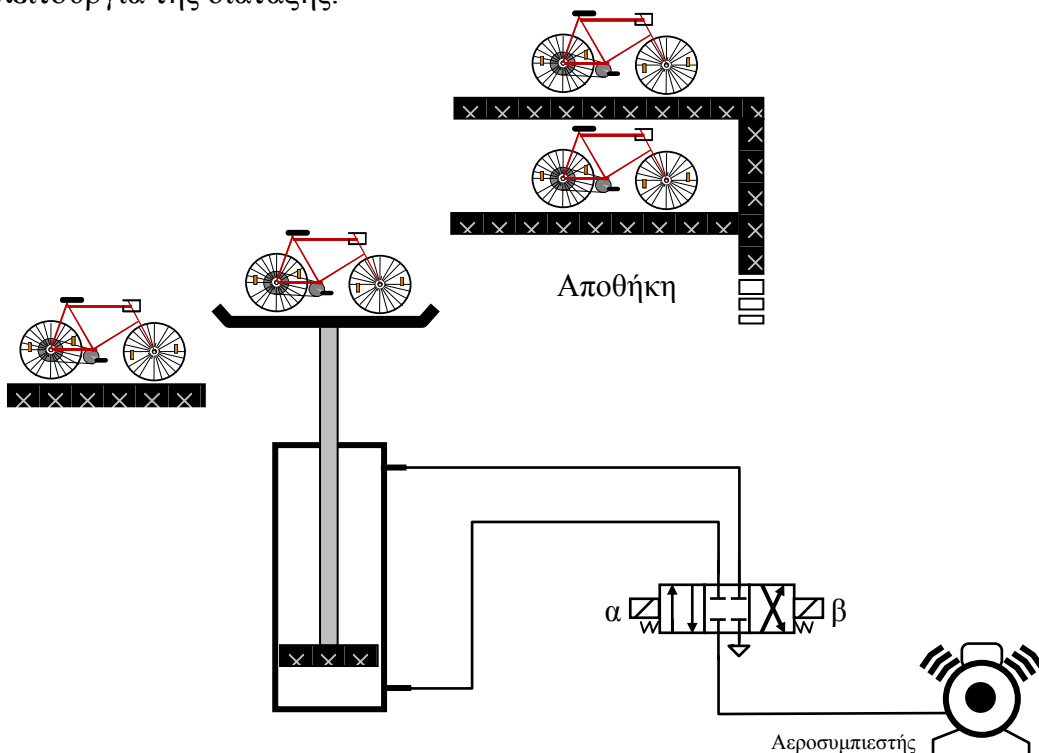
1. Σχεδιάστε τη διάταξη που φαίνεται στο σχήμα όπου το έμβολο εκτείνεται με σήμα START, φθάνει αυτό τέρμα της διαδρομής του, ενεργοποιεί τον οριακό διακόπτη και επιστρέφει αυτόματα. Εξομοιώστε τη λειτουργία της διάταξης.



Σημ.: Προετοιμάστε το κύκλωμα πεπιεσμένου αέρα καθώς και το κύκλωμα αυτοματισμού από το σπίτι, ώστε να έχετε αρκετό χρόνο για την υλοποίηση και εξομοίωση της διάταξης.

Να σωθεί το αρχείο (project) και όλες οι εικόνες-σχέδια σε δικό σας μέσο ώστε να υποβληθούν μαζί με την έκθεση της άσκησης.

2. Σχεδιάστε τη διάταξη που φαίνεται στο σχήμα όπου το έμβολο ανεβαίνει όσο πατάμε το μπουτόν ΠΑΝΩ και κατεβαίνει όσο πατάμε το μπουτόν ΚΑΤΩ. Το έμβολο πρέπει να μπορεί να σταματά σε όποια θέση είναι επιθυμητή. Εξομοιώστε τη λειτουργία της διάταξης.



(α) πνευματικό κύκλωμα

Σημ.: Προετοιμάστε το κύκλωμα αυτοματισμού σε γλώσσα Ladder από το σπίτι, ώστε να έχετε αρκετό χρόνο για την υλοποίηση και εξομοίωση της διάταξης.

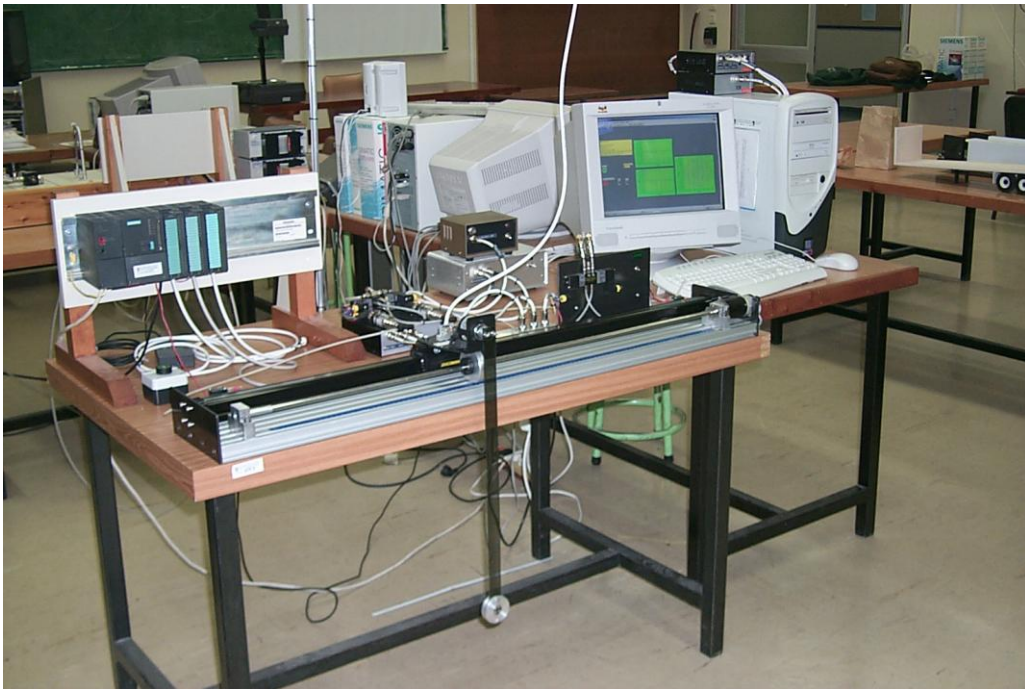
Να σωθεί το αρχείο (project) και όλες οι εικόνες-σχέδια σε δικό σας μέσο ώστε να υποβληθούν μαζί με την έκθεση της άσκησης.

3. Σχεδιάστε όποια διάταξη νομίζετε που να είναι συνθετότερη των προηγούμενων και εξομοιώστε τη λειτουργία της.

Να σωθεί το αρχείο (project) και όλες οι εικόνες-σχέδια σε δικό σας μέσο ώστε να υποβληθούν μαζί με την έκθεση της άσκησης.

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 506

ΕΛΕΓΧΟΣ ΘΕΣΗΣ ΚΑΙ ΤΑΛΑΝΤΩΣΕΩΝ ΕΚΚΡΕΜΟΥΣ ΦΟΡΤΙΟΥ ΣΕ ΔΙΑΜΗΚΗ ΦΟΡΕΑ-ΓΕΡΑΝΟ ΠΕΠΙΕΣΜΕΝΟΥ ΑΕΡΑ



Εργαστήριο Γενικής Ηλεκτροτεχνίας
Τμήμα Ηλεκτρολόγων Μηχανικών & Τεχνολογίας Υπολογιστών
Τομέας Συστημάτων & Αυτομάτου Ελέγχου

Πανεπιστήμιο Πατρών, 2010

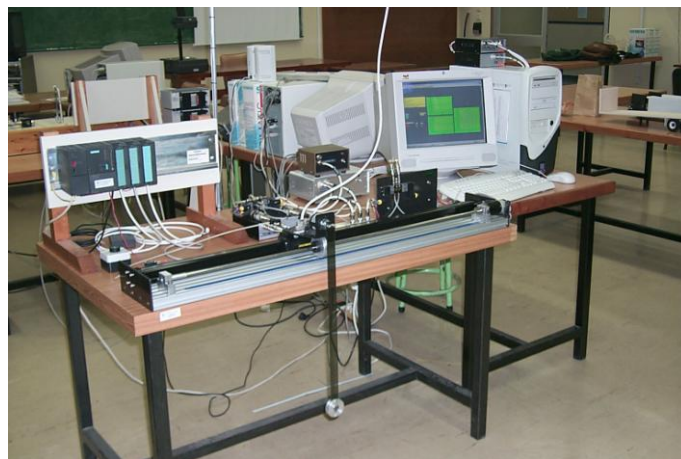
1 Στόχος της Άσκησης – Περιγραφή της Εργαστηριακής Διάταξης

Στην παρούσα ενότητα γίνεται μία σύντομη περιγραφή των στοιχείων από τα οποία αποτελείται η πειραματική διάταξη που ευρίσκεται στο χώρο του Εργαστηρίου Γενικής Ηλεκτροτεχνίας του Πανεπιστημίου Πατρών. Στόχος είναι να προσομοιωθεί η λειτουργία ενός συστήματος γερανογέφυρας για μεταφορά εκκρεμών φορτίων (Εικόνα 1), ώστε να δοκιμαστεί σε αυτό η αποτελεσματικότητα διαφόρων ελεγκτών.



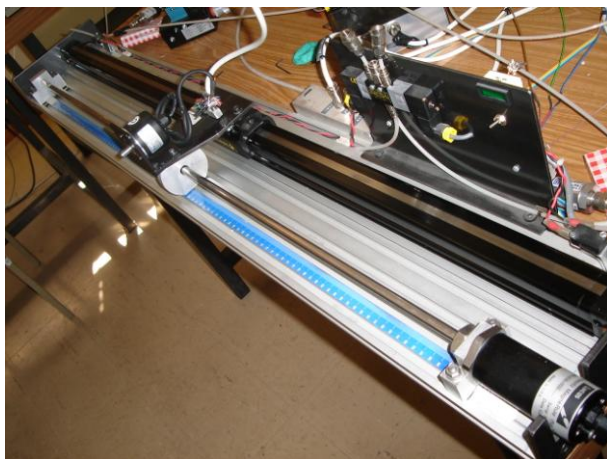
Εικόνα 1. Σύστημα γερανογέφυρας για τη μεταφορά φορτίων

Στην άσκηση αυτή, η διάταξη που χρησιμοποιείται είναι ένα εκκρεμές φορτίο αναρτώμενο από έναν φορέα, ο οποίος τίθεται σε κίνηση μέσω ενός ολοκληρωμένου πνευματικού συστήματος ελεγχόμενο από ηλεκτροβαλβίδες (Εικόνα 2).



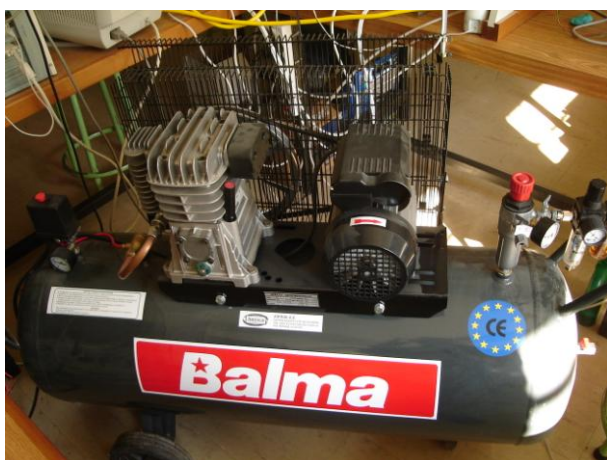
Εικόνα 2. Πειραματική διάταξη

Τα βασικά στοιχεία από τα οποία απαρτίζεται είναι ένας πνευματικός κύλινδρος διπλής δράσης (Εικόνα 3), όπου δύναται να παρέχεται αέρας και στους δύο θαλάμους, καθώς και ένα εκκρεμές φορτίο που αναρτάται από το φορέα ο οποίος ολισθαίνει επί του κυλίνδρου.



Εικόνα 3. Πνευματικός κύλινδρος διπλής δράσης

Η παρεχόμενη πίεση στο σύστημα δίνεται από έναν αεροσυμπιεστή στο χώρο του εργαστηρίου (Εικόνα 4), μέσω των σωλήνων διασύνδεσης.



Εικόνα 4. Αεροσυμπιεστής

Η είσοδος του αέρα μέσα στον κύλινδρο προκαλεί την μετατόπιση του πιστονιού, το οποίο στον συγκεκριμένο πνευματικό κύλινδρο συνδέεται μέσω κατάλληλου μηχανισμού με έναν φορέα που ολισθαίνει επί του εξωτερικού μέρους, διατηρώντας τη στεγανότητα των θαλάμων με το εξωτερικό περιβάλλον (Εικόνα 5).



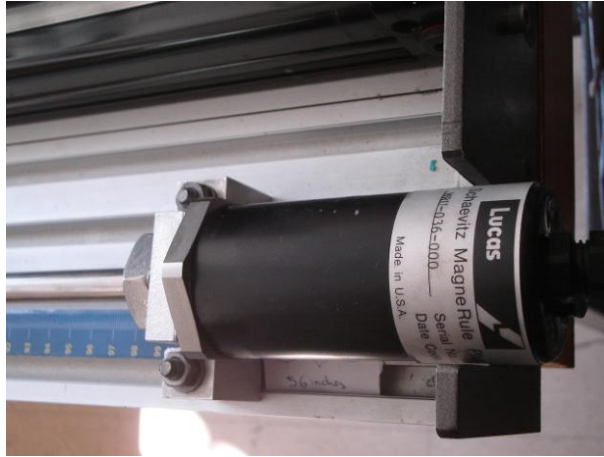
Εικόνα 5. Ολισθαίνων φορέας επί του κυλίνδρου

Ο φορέας έχει ενωθεί με έναν αισθητήρα θέσης μέσω μιας μεταλλικής πλάκας, πάνω στην οποία έχει προσαρτηθεί το εκκρεμές, αναρτώμενο από έναν άξονα, που μπορεί να περιστρέφεται ελεύθερα, διαμέσου μίας ελαφριάς αλουμινένιας ράβδου (Εικόνα 6).



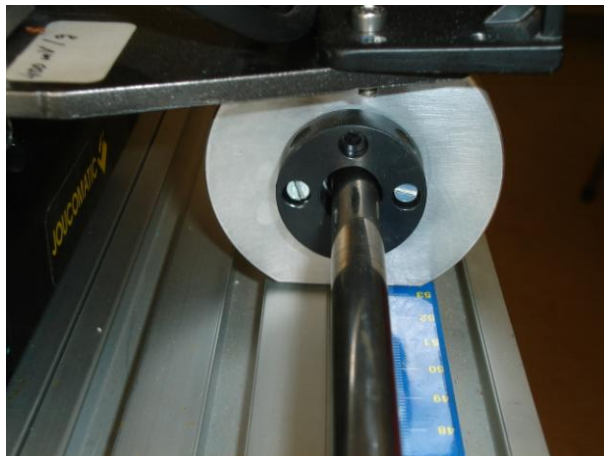
Εικόνα 6. Εκκρεμές φορτίο

Για τον έλεγχο του συστήματος έχουν προσαρτηθεί σε αυτό αισθητήρες σε κατάλληλα σημεία, ώστε να μετρούνται τα φυσικά μεγέθη που μας ενδιαφέρουν. Παράλληλα στον κύλινδρο, κατά την φορά κίνησης του εξωτερικού φορέα, έχει τοποθετηθεί ένας αισθητήρας θέσης (Εικόνα 7), ο οποίος μετράει την απόσταση ενός μαγνητικού δαχτυλιδιού από ένα προκαθορισμένο σημείο που βρίσκεται στο ένα άκρο του κυλίνδρου.



Εικόνα 7. Αισθητήρας θέσης φορέα

Συνδέοντας έτσι τον φορέα με το μαγνητικό αυτό δαχτυλίδι (Εικόνα 8), μέσω της μεταλλικής πλάκας που προαναφέρθηκε, μπορούμε να έχουμε συνεχή μέτρηση της θέσης του φορέα.



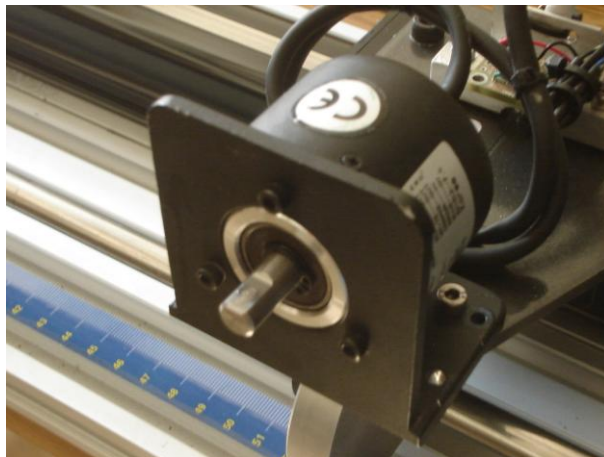
Εικόνα 8. Σύστημα διασύνδεσης μαγνητικού δαχτυλιδιού με φορέα κυλίνδρου

Επιπλέον, πάνω στον τελευταίο έχει προσαρτηθεί και ένα επιταχυνσιόμετρο (Εικόνα 9), ώστε να μετράται και η επιτάχυνσή του.

Τέλος, πάνω στην πλάκα που συνδέει το δαχτυλίδι με τον φορέα έχει στηριχθεί ένας αυξητικού τύπου encoder (Εικόνα 10) για την μέτρηση της γωνίας εκτροπής του εκκρεμούς, το οποίο αναρτάται από τον άξονα/άρθρωση του αισθητήρα αυτού.



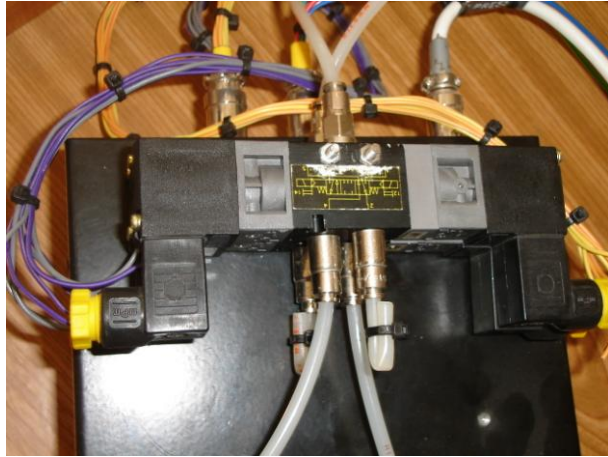
Εικόνα 9. Επιταχυνσιόμετρο



Εικόνα 10. Αυξητικού τύπου encoder

Ως ενεργοποιητές χρησιμοποιούνται δύο ζεύγη ηλεκτροβαλβίδων. Αρχικά, τα πειράματα διεξάγονται με δύο ψηφιακές (on-off) βαλβίδες 5/3 (Εικόνα 11). Η χρήση δύο αντί μίας βαλβίδας μας δίνει την επιπλέον δυνατότητα να παρέχουμε αέρα από τον αεροσυμπιεστή και στους δύο θαλάμους του κυλίνδρου ταυτόχρονα, κάνοντας όμως την μελέτη του συστήματος πιο περίπλοκη.

Στην συνέχεια, οι βαλβίδες αυτές αντικαθίστανται από δύο αναλογικές (PPC: Proportional Pressure Control), όπου καθεμία από αυτές δέχεται ως είσοδο ένα ηλεκτρικό σήμα που αντιπροσωπεύει την επιθυμητή πίεση στον εκάστοτε θάλαμο, και μέσω ενός ενσωματωμένου συστήματος κλειστού βρόχου οδηγεί την πίεση στην έξοδό της στο επιθυμητό επίπεδο (Εικόνα 12).



Εικόνα 11. On-off βαλβίδες 5/3



Εικόνα 12. Αναλογική PPC βαλβίδα

Τα σήματα των αισθητήρων (έξοδοι του συστήματος) συνδέονται σε μια PCI κάρτα του υπολογιστή από τον οποίο ελέγχεται η διάταξη της άσκησης. Οι ελεγκτές υλοποιούνται στο προγραμματιστικό περιβάλλον Labview της εταιρείας NI (www.ni.com), όπου σε πραγματικό χρόνο υπολογίζεται και αποστέλλεται στο σύστημα η εκάστοτε είσοδος (έξοδος του ελεγκτή).

2 Έλεγχος μέσω on-off Βαλβίδων

2.1 Λειτουργία των On-off Βαλβίδων

Όπως αναφέρθηκε στο σύστημα έχουν τοποθετηθεί 2 ηλεκτροβαλβίδες on-off μέσω των οποίων μπορεί να γίνει έλεγχος της πίεσης των θαλάμων και έμμεσα της κατάστασης του συστήματος. Η κάθε βαλβίδα ελέγχεται μέσω των δύο πηνίων της, το ρεύμα των οποίων παίρνει διακριτές τιμές. Δεδομένου ότι η κάθε βαλβίδα αναλαμβάνει την ρύθμιση της πίεσης ενός εκ των δύο θαλάμων του κυλίνδρου, μπορεί να θεωρηθεί ότι καθορίζει αν θα εισαχθεί αέρας στο θάλαμο, αν θα εξαερωθεί ο θάλαμος ή αν θα διατηρηθεί ο υπάρχων αέρας μέσα σε αυτόν.

Η δύναμη που εφαρμόζεται πάνω στον φορέα είναι ανάλογη της διαφοράς της πίεσης των δύο θαλάμων. Δεδομένου ότι τα ρεύματα των πηνίων των βαλβίδων είναι ψηφιακά σήματα, ή πίεση κάθε θαλάμου μπορεί να είναι P_{max} ή P_{atm} , όπου P_{max} η πίεση του αεροσυμπιεστή (όταν η βαλβίδα παρέχει αέρα στο θάλαμο) και P_{atm} η ατμοσφαιρική πίεση (όταν η βαλβίδα ελευθερώνει τον υπάρχοντα αέρα του θαλάμου στον περιβάλλον). Ενδιάμεσες καταστάσεις δεν δύναται να επιτευχθούν/ελεγχθούν, επειδή οι καταστάσεις των βαλβίδων είναι διακριτές. Συνεπώς η εφαρμοζόμενη δύναμη στο φορέα δύναται να πάρει τις τιμές: $(P_{max}-P_{atm})A$, $-(P_{max}-P_{atm})A$ ή 0 , όπου A η διατομή του κυλίνδρου. Συμπεραίνουμε λοιπόν ότι μπορεί να γίνει η εξής σύμβαση όσο αφορά την είσοδο στο σύστημα:

- $u = -1$: μετακίνηση προς τα αριστερά
- $u = 0$: ακινητοποίηση του φορέα
- $u = 1$: μετακίνηση προς τα δεξιά

όπου για κάθε μία από τις παραπάνω τιμές του u , τα ρεύματα των πηνίων καθορίζονται μονοσήμαντα. Περαιτέρω λεπτομέρειες δεν αποτελούν αντικείμενο της παρούσας άσκησης.

Όσο αφορά την περίπτωση $u=0$ (το οποίο συνεπάγεται ακινητοποίηση) υπάρχουν 3 τρόποι για να επιτευχθεί αυτό: (α) ταυτόχρονη παροχή αέρα και

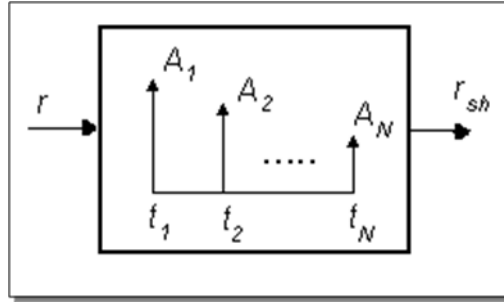
στους δύο θαλάμους, (β) ταυτόχρονη εξαέρωση και των δύο θαλάμων και (γ) εγκλωβισμός του υπάρχοντος αέρα και στους δύο θαλάμους. Στα πλαίσια της άσκησης χρησιμοποιείται ο δεύτερος τρόπος προς αποφυγή τυχών διαρροών αέρα μεταξύ των θαλάμων (περίπτωση γ) ή κραδασμών (περίπτωση α). Αυτός είναι και ο λόγος που το σύστημα ελέγχεται από δύο βαλβίδες, και όχι από μία.

2.2 Έλεγχος του Συστήματος με Προαντισταθμιστή Input Shaper

Στην πράξη υπάρχουν πολλά φυσικά συστήματα κατά τη λειτουργία των οποίων κυριαρχούν φαινόμενα ταλαντώσεων μακράς διάρκειας. Η συμπεριφορά αυτή στα περισσότερα από αυτά είναι ανεπιθύμητη καθώς δυσχεραίνουν την επιθυμητή λειτουργία τους ή προκαλούν σχετικά νωρίς φθορές στα μηχανικά μέρη τους.

Διάφοροι κλειστού βρόχου ελεγκτές μπορούν να σχεδιαστούν για την αντιμετώπιση των υποαποσβεννύμενων ταλαντώσεων που παρατηρούνται με καλά αποτελέσματα. Δυστυχώς τις περισσότερες φορές, λόγω της πολυπλοκότητας των πραγματικών συστημάτων, είτε οι σχεδιαζόμενοι ελεγκτές είναι πολύ μεγάλης τάξεως ή άλλες φορές δεν είναι δυνατή η μέτρηση κάποιων απαραίτητων για έλεγχο φυσικών μεγεθών, με συνέπεια να υπάρχει ανάγκη για ένα ανοιχτού βρόχου ελεγκτή (προαντισταθμιστής). Όπως είναι γνωστό όμως, οι περισσότεροι ανοιχτού βρόχου ελεγκτές δεν έχουν καλά αποτελέσματα.

Η input shaping τεχνική είναι μία πολύ καλή λύση σε τέτοιες περιπτώσεις, αφού πρακτικά είναι ένα φίλτρο που διαμορφώνει κατάλληλα την είσοδο στο σύστημα (συνήθως το σήμα αναφοράς αφού ο βρόχος δεν κλείνει), ώστε να καταστέλλονται οι ταλαντώσεις στην έξοδο. Ο shaper ουσιαστικά είναι ένας συρμός N κρουστικών με t_j και A_j οι χρονικές στιγμές και τα πλάτη των κρουστικών αυτών αντίστοιχα, όπου $j = 1, \dots, N$. Η σχηματική αναπαράστασή του φαίνεται στην Εικόνα 13, όπου το σήμα r_{sh} προκύπτει από τη συνέλιξη του αρχικού σήματος r με τον input shaper.



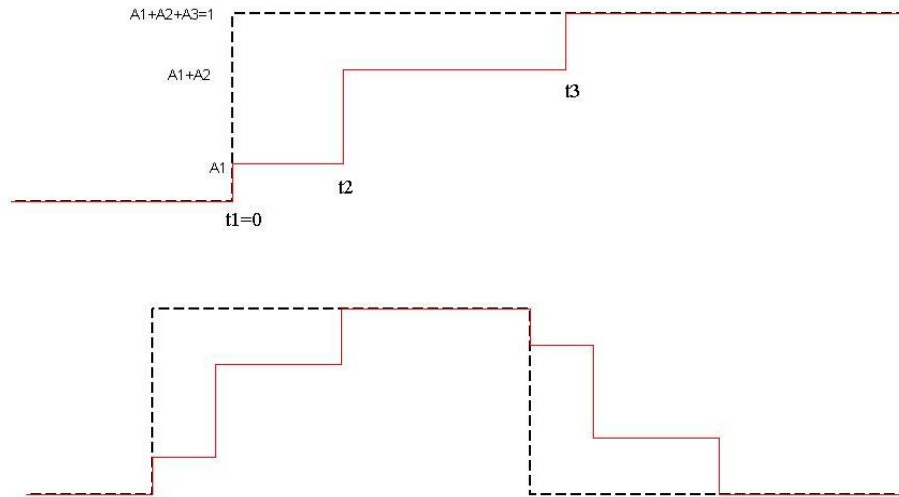
Εικόνα 13. Σχηματική αναπαράσταση του input shaper

Γνωρίζοντας ότι η συνέλιξη ενός σήματος με μία κρουστική πλάτους A τη χρονική στιγμή T αναπαρίσταται στο πεδίο του χρόνου ως χρονοκαθυστέρηση του σήματος αυτού κατά T μονάδες και παράλληλα ενίσχυση κατά έναν συντελεστή A , εύκολα μπορεί να εξαχθεί η σχέση που συνδέει τα σήματα r και r_{sh} ως:

$$r_{sh}(t) = A_1 r(t-t_1) + A_2 r(t-t_2) + \dots + A_N r(t-t_N) = \sum_{j=1}^N A_j r(t-t_j) .$$

Ένας shaper είναι ένας ελεγκτής ανοιχτού βρόχου και έχει ως σκοπό να διαμορφώσει κατάλληλα το σήμα εισόδου στο σύστημα ώστε η έξοδος να φτάσει στο επιθυμητό σημείο το γρηγορότερο δυνατόν χωρίς ταλαντώσεις. Όπως φαίνεται, οι παράμετροι του ελεγκτή που πρέπει να προσδιοριστούν είναι οι t_j και A_j , για $j=1, \dots, N$, για έναν πεπερασμένο αριθμό κρουστικών N . Το μεγάλο πλεονέκτημα του ελεγκτή αυτού είναι το γεγονός ότι δεν απαιτείται η πλήρης γνώση του μοντέλου του συστήματος για τον προσδιορισμό των παραμέτρων αυτών, αλλά αποκλειστικά και μόνο τα χαρακτηριστικά της ταλαντωτικής συμπεριφοράς αυτού: ο συντελεστής απόσβεσης (ζ) και η φυσική ιδιοσυχνότητα (ω) της ταλάντωσης.

Στην Εικόνα 14 φαίνεται σχηματικά ο τρόπος με τον οποίο διαμορφώνεται από τον input shaper μια βηματική και μια παλμική είσοδος αντίστοιχα.

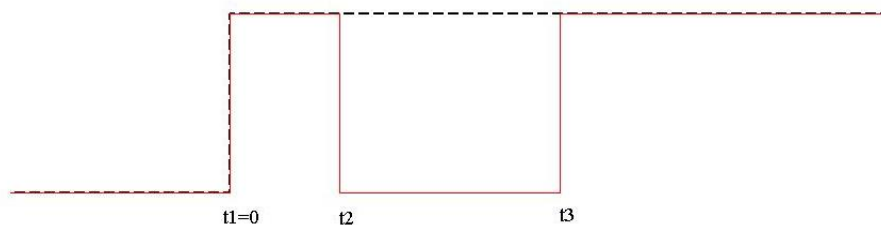


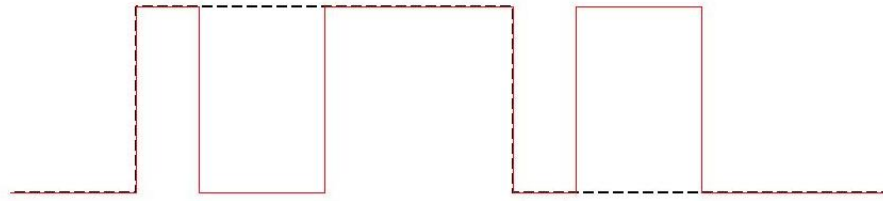
Εικόνα 14. Διαμόρφωση ενός βηματικού (επάνω γράφημα) και παλμικού (κάτω γράφημα) σήματος μέσω ενός input shaper τριών κρουστικών (Περίπτωση Αναλογικής Εισόδου)

Υπάρχουν όμως περιπτώσεις συστημάτων (όπως αυτό της άσκησης στην περίπτωση των on-off βαλβίδων) όπου η είσοδος δεν μπορεί να λάβει οποιαδήποτε τιμή, αλλά μία εκ των $u \in -u_s, 0, u_s$, $u_s \in \mathfrak{R}_+^*$.

Στην περίπτωση αυτή ο shaper, που στη βιβλιογραφία συναντάται ως UM (Unity-Magnitude), πρέπει να είναι ένας συρμός με περιττό αριθμό κρουστικών μεγαλύτερο ίσο του τρία, και μάλιστα τα πλάτη των κρουστικών αυτών να ικανοποιούν τη σχέση $A_j = -1^{j+1}$.

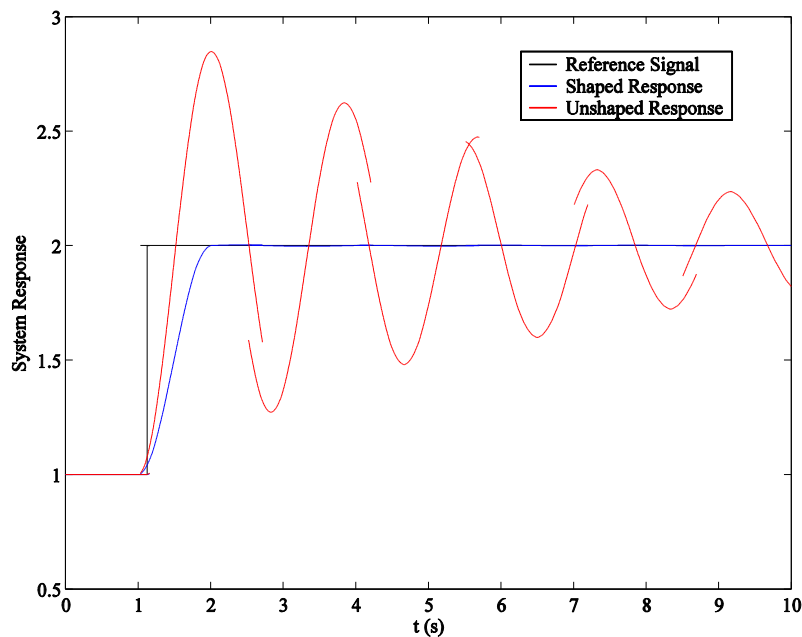
Στην Εικόνα 15 φαίνεται σχηματικά ο τρόπος με τον οποίο διαμορφώνεται από τον input shaper μια βηματική και μια παλμική είσοδος αντίστοιχα, στην περίπτωση που η είσοδος θεωρείται ότι παίρνει διακεκριμένες τιμές.





Εικόνα 15. Διαμόρφωση ενός βηματικού (πάνω γράφημα) και παλμικού (κάτω γράφημα) σήματος μέσω ενός *input shaper* τριών κρουστικών (Περίπτωση Ψηφιακής Εισόδου)

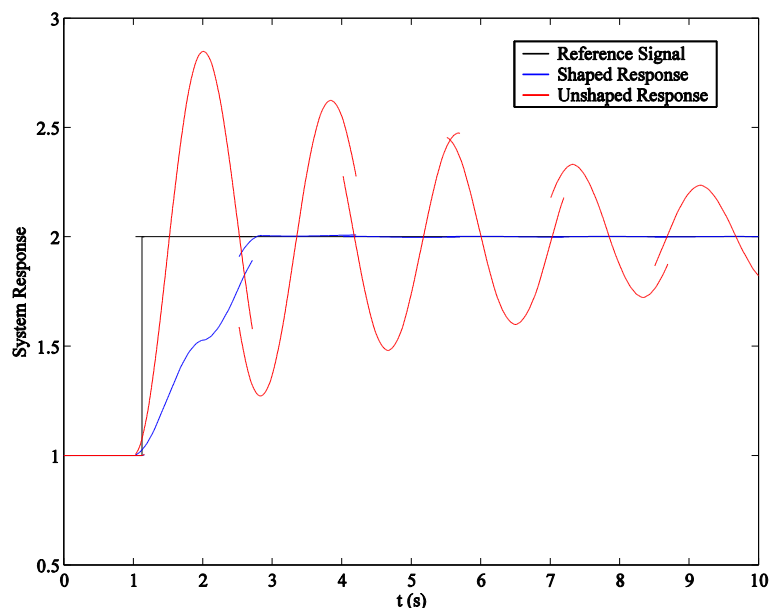
Για να γίνει κατανοητός ο τρόπος με τον οποίο καταστέλλονται οι ταλαντώσεις, παρουσιάζονται στο Διάγραμμα 1 οι αποκρίσεις ενός δευτεροβάθμιου συστήματος με και χωρίς τη χρήση του *shaper*, στην περίπτωση αναλογικής εισόδου. Όπως μπορεί να παρατηρηθεί, η έξοδος φτάνει στην επιθυμητή θέση χωρίς ταλαντώσεις σε χρόνο ίση με την ημιπερίοδο ταλάντωσής του.



Διάγραμμα 1. Απόκριση γραμμικού συστήματος ελεγχόμενο με *shaper* 2 κρουστικών

Τα αποτελέσματα προσομοίωσης του ίδιου συστήματος με αυτό της παραπάνω περίπτωσης για την ίδια είσοδο, ελεγχόμενο όμως τώρα από έναν *shaper* 3 κρουστικών, παρουσιάζονται στο Διάγραμμα 2. Παρατηρούμε ότι η έξοδο πάλι φτάνει στην επιθυμητή τιμή χωρίς ταλαντώσεις, με τη διαφορά ότι

φτάνει σε χρόνο διπλάσιο από όταν είχε χρησιμοποιηθεί ένας shaper 2 κρουστικών.



Διάγραμμα 2. Απόκριση γραμμικού συστήματος ελεγχόμενο με shaper 3 κρουστικών

Παρατηρείται ότι όσο αυξάνεται το μήκος του shaper, N , τόσο αυξάνει ο χρόνος που απαιτείται για να φτάσει το σύστημα στη μόνιμη κατάσταση χωρίς ταλαντώσεις, t_N . Εύλογα, λοιπόν, μπορεί κάποιος να αναρωτηθεί τι νόημα έχει ένας shaper με άνω των δύο κρουστικών (που είναι και ο ελάχιστος αριθμός). Η απάντηση βρίσκεται στο γεγονός ότι στην παραπάνω ανάλυση έχει γίνει σιωπηρά η υπόθεση ότι η φυσική ιδιοσυχνότητα ω , καθώς και ο συντελεστής απόσβεσης ζ του υπό έλεγχο συστήματος είναι γνωστά με απόλυτη ακρίβεια. Εδώ είναι το σημείο όπου υπεισέρχεται το ζήτημα της ευαισθησίας του ελεγκτή σε αβεβαιότητα των παραμέτρων του συστήματος. Πιο συγκεκριμένα, όσο πιο πολλές είναι οι κρουστικές του συρμού του shaper, τόσο λιγότερο ευαίσθητο είναι το σύστημα σε αποκλίσεις των εκτιμώμενων ω και ζ από τις πραγματικές.

2.3 Έλεγχος του Συστήματος μέσω on-off Ελεγκτή

Ο πιο συνήθης χρησιμοποιούμενος ελεγκτής για συστήματα που δέχονται ψηφιακή είσοδο ως γνωστόν είναι ο on-off. Το υπό έλεγχο σύστημα δυστυχώς απαιτεί τον έλεγχο δύο μεγεθών (θέση φορέα και γωνία εκτροπής εκκρεμούς) μέσω μιας μόνο εισόδου ελέγχου. Επιλέγοντας σαν υπό έλεγχο μεταβλητή την

θέση του φορέα (και εμμέσως του φορτίου) ο on-off ελεγκτής εξετάζει την διαφορά της επιθυμητής θέσης (που δίνεται από τον χρήστη) από την πραγματική και καθορίζει την είσοδο στο σύστημα μέσω του προσήμου της διαφοράς αυτής. Όπως αναφέρθηκε προηγουμένως, έχοντας υπολογίσει την τιμή του u , μπορούν να καθοριστούν μονοσήμαντα τα ρεύματα των βαλβίδων που στην συνέχεια αποστέλλονται στο πραγματικό σύστημα.

3 Έλεγχος Συστήματος μέσω Αναλογικών Βαλβίδων

Για το σύστημα της εργαστηριακής διάταξης πρόκειται να ελεγχθούν ταυτοχρόνως η θέση του φορέα και η γωνία εκτροπής του εκκρεμούς από την κατακόρυφο με χρήση μερικής ανατροφοδότησης κατάστασης. Δεδομένου του γεγονότος ότι χρησιμοποιούνται αναλογικές βαλβίδες, πλέον η είσοδος δεν παίρνει μόνο διακριτές τιμές (όπως στην περίπτωση των on-off βαλβίδων) αλλά μπορεί να γίνει αναλογικός έλεγχος της πίεσης του κάθε θαλάμου του κυλίνδρου. Αυτό έχει ως συνέπεια να μπορούμε να επέμβουμε στη διαφορική πίεση αυτών και συνεπώς να ελέγξουμε τη δύναμη που ασκείται πάνω στο φορέα (η οποία όπως έχει προαναφερθεί είναι ανάλογη της διαφορικής πίεσης).

Μπορεί λοιπόν να θεωρηθεί ότι η είσοδος στο σύστημα είναι η διαφορική πίεση μεταξύ των δύο θαλάμων του κυλίνδρου. Το διάνυσμα κατάστασης αποτελείται από (α) τη θέση του φορέα x , (β) την ταχύτητα του φορέα \dot{x} , (γ) την γωνία εκτροπής του εκκρεμούς από την κατακόρυφο θ και (δ) την γωνιακή ταχύτητα του εκκρεμούς $\dot{\theta}$.

Στον παρόν πείραμα επιλέγουμε να ανατροφοδοτήσουμε (1) τη θέση του φορέα και (2) τη γωνιακή ταχύτητα του εκκρεμούς φορτίου. Έτσι, ο νόμος ελέγχου ανάγεται να είναι της μορφής $\Delta P = K_x x^{des} - x + K_{\dot{\theta}} \dot{\theta}$, όπου ΔP είναι η είσοδος του συστήματος ($P_1 - P_2$), x^{des} είναι η επιθυμητή θέση του φορέα επί του κυλίνδρου και $K_x, K_{\dot{\theta}}$ είναι οι παράμετροι του ελεγκτή. Προφανώς $\dot{\theta}^{des} = 0$, γι' αυτό και παραλείπεται από το νόμο ελέγχου. Ο λόγος που δεν επιλέγουμε να ανατροφοδοτήσουμε τη γωνία του εκκρεμούς (την οποία διαβάζουμε από τον encoder) είναι ο εξής:

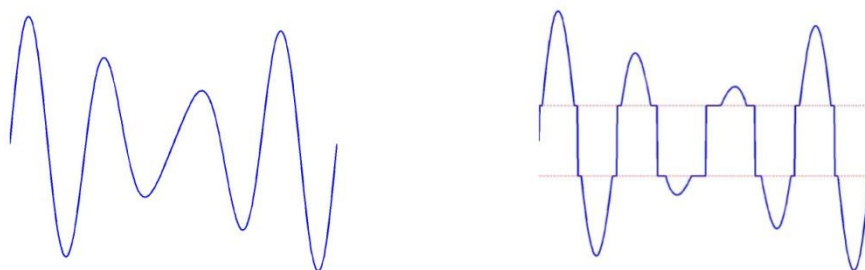
Επειδή ο encoder (αισθητήρας γωνίας εκτροπής) είναι αυξητικού τύπου τυγχάνει να εισάγει σφάλματα μέτρησης τα οποία συνεχώς συσσωρεύονται κατά τη διάρκεια του πειράματος. Αυτό συνεπάγεται ότι στην περίπτωση που το εκκρεμές ισορροπεί στην πραγματικότητα, ο encoder ενδέχεται να δείχνει ότι ισορροπεί στις 10 μοίρες, ως παράδειγμα. Αυτό θα έχει ως κατάληξη, ο νόμος

ελέγχου να είναι διάφορος του μηδενός ακόμα και όταν το σύστημα βρίσκεται στην πραγματικότητα στην ισορροπία.

Η γωνιακή ταχύτητα του εκκρεμούς όμως δεν είναι μετρήσιμη στην παρούσα διάταξη. Μπορούμε όμως να την εκτιμήσουμε παραγωγίζοντας αριθμητικά το σήμα της γωνίας που λαμβάνουμε από τον encoder.

Όπως έχει αναφερθεί, ο φορέας που ολισθαίνει επί του κυλίνδρου υπόκειται σε ισχυρά μη-γραμμικές τριβές. Το κύριο πρόβλημα στον έλεγχο είναι η στατική τριβή, όπου αν η διαφορική πίεση στους θαλάμους είναι μικρότερη από κάποια τιμή (1.1bar στην συγκεκριμένη εργαστηριακή διάταξη), τότε ο φορέας δεν ξεκινάει την κίνηση, εκτός εάν κάποια στιγμή ξεπεραστεί η τιμή αυτή, και δεδομένου ότι δεν θα ξανασταματήσει η κίνησή του.

Για να αποφευχθεί η ακινητοποίηση του φορέα σε κάποια άλλη θέση εκτός αυτής που θέλουμε, εισάγουμε έναν μη γραμμικό όρο που κρατάει την είσοδο πάντα μεγαλύτερη (σε απόλυτη τιμή) της στατικής τριβής, με αποτέλεσμα να υπάρχει πάντα κίνηση. Πρακτικά, είναι προτιμότερο το σύστημα να κάνει «μικρομανούβρες» γύρω από την επιθυμητή κατάσταση, παρά να «κλειδώνει» σε κάποια άλλη θέση του φορέα εκτός της επιθυμητής. Η επίδραση ενός τέτοιου μη γραμμικού όρου φαίνεται γραφικά στην Εικόνα 16.



Εικόνα 16. Μετασχηματισμός του σήματος ελέγχου έτσι ώστε ο φορέας να βρίσκεται πάντα σε κίνηση

4 Εκτέλεση της Άσκησης

4.1 Έλεγχος Συστήματος μέσω Ψηφιακών (on-off) Βαλβίδων.

4.1.1 Λειτουργία σε Ανοικτό Βρόχο

Το πρώτο βήμα σε κάθε σύστημα υπό έλεγχο είναι να εξεταστεί ή συμπεριφορά του ώστε να εξακριβωθεί αν πράγματι έχει νόημα να υλοποιηθεί κάποιος έλεγχος για αυτό. Συνήθως το κίνητρο είναι οι ανεπιθύμητες συμπεριφορές του συστήματος όταν λειτουργεί σε χειροκίνητο έλεγχο.



Εικόνα 17. Open Loop

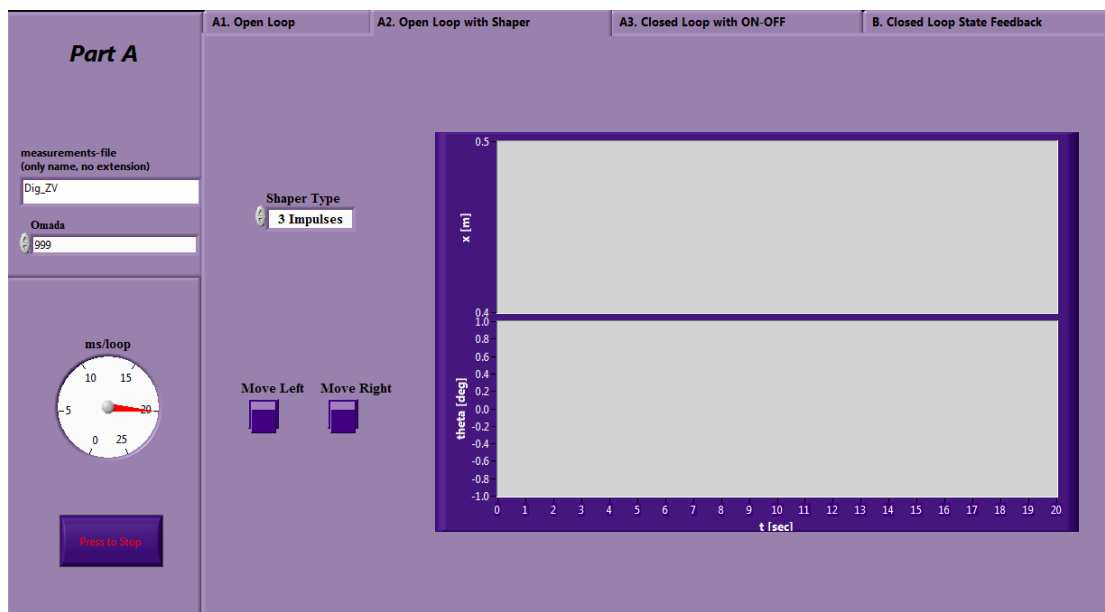
Να πραγματοποιήσετε μια συνεχής χειροκίνητη μανούβρα του φορέα και να καταγραφούν τα δεδομένα σε αρχείο. **Μεγάλη προσοχή η κίνηση να γίνει από αριστερά προς τα δεξιά ώστε να αποφευχθούν τυχόν καταστροφές στην διάταξη.**

Να σχολιαστεί η όποια συμπεριφορά του συστήματος και να τονιστεί αν υπάρχει ανάγκη ελέγχου. Αν ναι, ποια φυσικά μεγέθη χρίζουν ελέγχου; Οι στήλες του αρχείου μετρήσεων αντιστοιχούν στα παρακάτω μεγέθη.

$t(sec)$	$x(m)$	$\theta(deg)$
----------	--------	---------------

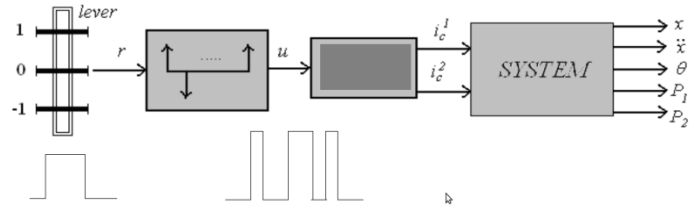
4.1.2 Έλεγχος Ανοικτού Βρόχου μέσω Input Shaper

Στο προηγούμενο ερώτημα (4.1.1) ορίσαμε την κίνηση του φορέα χειροκίνητα ενώ παράλληλα δεν υπήρχε κανένας έλεγχος της γωνίας εκτροπής του εκκρεμούς. Στο στάδιο αυτό, χωρίς να κλείσει ο βρόχος, θα εισάγουμε έναν προαντισταθμιστή (input shaper) ο οποίος θα έχει ως σκοπό να τροποποιεί σε πραγματικό χρόνο το σήμα του χειριστή ώστε, όταν ο φορέας φτάσει στη θέση που εμείς του ορίζουμε (χειροκίνητα), η ταλάντωση αυτού να είναι μηδενική (όσο είναι δυνατόν να επιτευχθεί).



Εικόνα 18. Open Loop with Shaper

Επειδή η είσοδος στο σύστημα μπορεί να θεωρηθεί ότι παίρνει διακριτές τιμές (-1:κίνηση προς τα αριστερά, 0:ακινητοποίηση, +1: κίνηση προς τα δεξιά), θα χρησιμοποιήσουμε έναν unity magnitude shaper (UM), έτσι ώστε το μετασχηματισμένο σήμα να παίρνει κι αυτό τιμές στο σύνολο αυτό. Οι παράμετροι του shaper είναι τα χαρακτηριστικά της ταλάντωσης του εκκρεμούς (ω , ζ), τα οποία έχουν εκτιμηθεί εκ των προτέρων, ενώ δεν χρειάζεται ανάδραση κανενός φυσικού μεγέθους, εφόσον ο έλεγχος είναι ανοικτού βρόχου.



Εικόνα 19. Δομικό διάγραμμα συστήματος ελέγχου

Να πραγματοποιηθεί μία μανούβρα του φορέα από τα αριστερά προς τα δεξιά χειροκίνητα (μέσω των 2 κουμπιών του νί) για κάθε τύπο shaper (3 και 5 κρουστικές αντίστοιχα). Να απεικονιστούν η θέση του φορέα και η γωνία του εκκρεμούς συναρτήσει του χρόνου και να σχολιαστεί η επίδραση του input shaper:

I. Στην εύρος της ταλάντωσης συγκριτικά με το ερώτημα 4.1.1. Γιατί πιστεύετε ότι δεν επιτυγχάνεται απόλυτα μηδενική ταλάντωση;

II. Να γίνει σύγκριση της επίδρασης του shaper 3 και 5 κρουστικών όσο αφορά την ταλάντωση του εκκρεμούς. Πόσες υπό-μανούβρες απαιτούνται για να γίνει η μετάβαση του φορέα στην επιθυμητή θέση; Σχολιάστε.

III. Με τον τρόπο αυτό επιτύχατε απόλυτο έλεγχο της θέσης; Σχολιάστε.

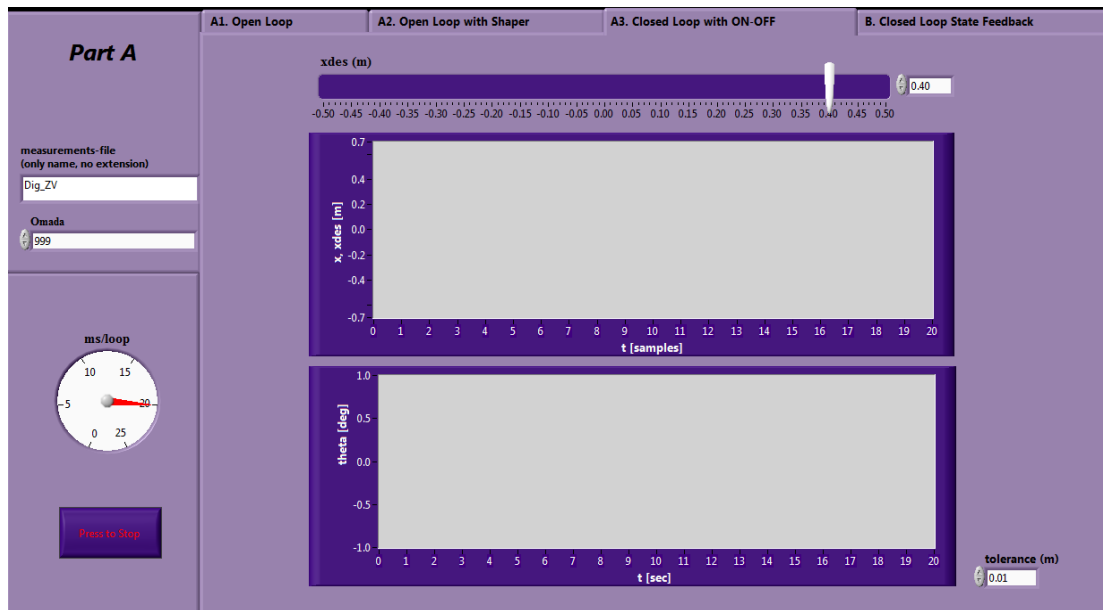
Σημειώνεται ότι τα μεγέθη που αντιστοιχούν στις στήλες του αρχείου φαίνονται παρακάτω.

$t(sec)$	$x(m)$	$\theta(deg)$
----------	--------	---------------

4.1.3 Έλεγχος Κλειστού Βρόχου μέσω on-off Ελεγκτή

Μέχρι τώρα ο έλεγχος του συστήματος ήταν ανοικτού βρόχου. Στη φάση αυτή θα πειραματιστούμε με έλεγχο κλειστού βρόχου. Το πρόβλημα στην περίπτωση αυτή έγκειται στο γεγονός ότι η είσοδος παίρνει διακριτές τιμές, $\{-1,0,1\}$, με συνέπεια κλασικοί ελεγκτές (P, PID) να μην μπορούν να εφαρμοστούν (Γιατί). Ο πιο διαδεδομένος ελεγκτής για αυτές τις περιπτώσεις είναι ο on-off.

Επειδή όμως και το σύστημα μας έχει δύο καταστάσεις προς έλεγχο (θέση φορέα, γωνία εκκρεμούς) αλλά μία είσοδο, θα πρέπει να επιλέξουμε μία εκ των δύο αυτών για ρύθμιση. Για ένα γερανοφόρο σύστημα είναι πιο σημαντική η μεταφορά του φορτίου σε κάποια επιθυμητή θέση με κάποιο ποσό ταλάντωσης, παρά η ισορροπία του σε κάποια άλλη θέση. Για αυτό το λόγο, ο on-off έλεγχος εφαρμόζεται στη θέση του φορέα x .



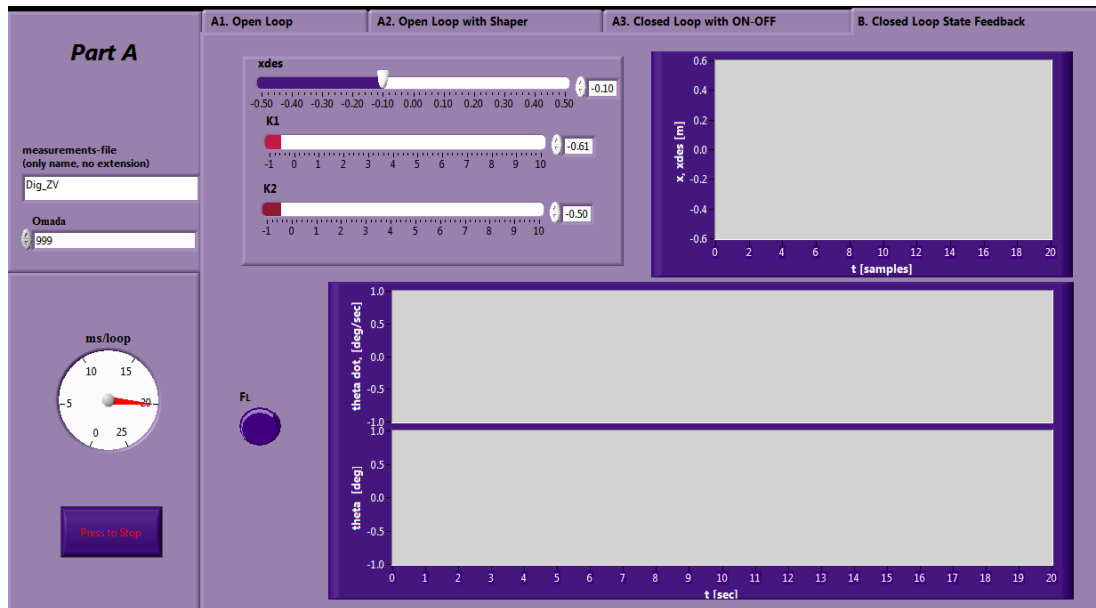
Εικόνα 20. Closed Loop with ON-OFF

Να ληφθεί μία μέτρηση για διάφορες μεταβάσεις του φορέα επί του κυλίνδρου και να σχολιαστεί η συμπεριφορά του συστήματος. Τι παρατηρείτε όσον αφορά τη γωνία εκτροπής του εκκρεμούς; Σχολιάστε. Σχολιάστε επίσης την επίδραση της ανεκτικότητας σφάλματος (για τη θέση του φορέα) σε ότι αφορά το ποσό ταλάντωσης του εκκρεμούς και της συμπεριφοράς του φορέα. Τα μεγέθη που αντιστοιχούν στις στήλες του αρχείου φαίνονται παρακάτω.

$t(sec)$	$xdes(m)$	$x(m)$	$\theta(deg)$
----------	-----------	--------	---------------

4.2 Έλεγχος Συστήματος μέσω Αναλογικών (PPC) Βαλβίδων.

Για το σύστημα της εργαστηριακής διάταξης πρόκειται να ελεγχθούν ταυτοχρόνως η θέση του φορέα και η γωνία εκτροπής του εκκρεμούς από την κατακόρυφο με χρήση (μερικής) ανατροφοδότησης κατάστασης.



Εικόνα 21. Closed Loop State Feedback

I. Απενεργοποιώντας το μη γραμμικό όρο για την αντιστάθμιση των τριβών ορίστε τα κέρδη στο διάστημα $[0.5,1]$. Επιλέξτε μια επιθυμητή θέση του φορέα. Τι παρατηρείτε;

II. Σε καθένα από τα παρακάτω ζητήματα θα πρέπει να αποθηκεύσετε τις μετρήσεις σε διαφορετικό αρχείο. Η κάθε στήλη στο αρχείο αντιστοιχεί στο φυσικό μέγεθος που φαίνεται παρακάτω:

$t(sec)$	$x^{des}(m)$	$x(m)$	$\theta(deg)$	$\theta dot(deg/sec)$
----------	--------------	--------	---------------	-----------------------

Να παρουσιαστούν και να σχολιαστούν αντίστοιχα τα γραφήματα της θέσης και της γωνιακής ταχύτητας του εκκρεμούς. Σχολιάστε τον τρόπο με τον οποίο το σύστημα συγκλίνει στη επιθυμητή κατάσταση.

K_x	$[0.5,1]$	$[8,10]$	$[5,8]$	$[-1,-0.5]$	$[5,8]$
K_θ	$[8,10]$	$[0.5,1]$	$[5,8]$	$[5,8]$	$[-1,-0.5]$

Τι συμπεραίνετε όσο αφορά τη συσχέτιση που έχουν τα κέρδη του ελεγκτή με τη βαρύτητα που δίνει ο ελεγκτής για την εξάλειψη του κάθε σφάλματος; Για τις 2 τελευταίες περιπτώσεις σχολιάστε τον τρόπο με τον οποίο ωθείται το σύστημα στην αστάθεια.