

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Έλεγχος Μπάρων σε Διαστάυρωση
Τρένων-Αυτοκινήτων με τη χρήση PLC



ΕΠΙΜΕΛΕΙΑ

**ΚΑΠΛΑΝΙΔΗΣ ΝΙΚΟΛΑΟΣ
ΑΛΕΞΑΝΔΡΟΥ ΒΑΣΙΛΗΣ**

ΕΙΣΗΓΗΤΗΣ

ΦΡΑΓΚΙΑΔΑΚΗΣ ΝΙΚΟΛΑΟΣ

ΠΕΡΙΕΧΟΜΕΝΑ

Σκοπός της άσκησης	σελ.1
Τι είναι τα P.L.C.	σελ.1
Που τα χρησιμοποιούμε	σελ.2
Κλασσικός αυτοματισμός	σελ.3
Ιστορική αναδρομή	σελ.4
Πλεονεκτήματα χρησιμοποίησης του P.L.C	σελ.5
Στάδια εργασίας	σελ.6
Οικογένεια SIMATIC S7	σελ.7
Τα βασικά μέρη του P.L.C	σελ.8-9
Εισόδοι P.L.C	σελ.10-11
Εξόδοι P.L.C	σελ.12
Μνήμη	σελ.13
CPU	σελ.14
FIRMWARE	σελ.15
Λειτουργία συστήματος P.L.C	σελ.16-17
Τι σημαίνει προγραμματισμός του P.L.C	σελ.18-20
STL	σελ.20
Βασικά στοιχεία για την δομή ενός προγράμματος	σελ.21
Οργάνωση του προγράμματος	σελ.21
Λογικές εντολές	σελ.22-30
Βήματα μεταφοράς προγράμματος	σελ.31-35
Παραδείγματα προγραμματισμού	σελ.36
Έλεγχος μπαρών	σελ.37-39
LED	σελ.40
Κύκλωμα τροφοδοσίας	σελ.41
Μακέτα εργασίας	σελ.42
Κώδικας του P.L.C	σελ.44-60
Επίλογος	σελ.61

The purpose

The purpose of this final project is to help to avoid accidents and become safer the crossroad between trains and cars. For that, we use circuits and P.L.C device.

The P.L.C

The P.L.C (Programmable Logic Controllers) are micro calculator systems which the right programming of logic equations solve automations problems.

The P.L.C's are electronics devices in which we can simulate a table of automation. For ex. There are inputs and outputs which are connected with elements of a system and of course an algorithm which adjusts a specific combination of inputs, which produce a solution in the outputs. The specific characteristic of P.L.C is that the rules, which adjust the behavior of outputs, are not stable like in a classic table of automation, but we can change them easily by changing the program without to change the hardware of the system. As for the parts of P.L.C, consist of the CPU which contents the logic of automation and reads the situation of inputs cards, enable the outputs cards, according to the program which is stored in the memory. The CPU is a combination with input card, adjusts the situation of a switch. Also the output card adjusts relay and pass through it, enable the system.

So, the system works as: The CPU reads the inputs, as for example every input and if we have a potential storage the logic 1 in the memory. Furthermore the program tests the inputs and decides the logics of outputs in which there is storage in the appropriate part of output memory. Finally, one part of the memory transfers in the output card and enable the relay. So we have a specific situation which repeats all the time.

ΕΙΣΑΓΩΓΗ

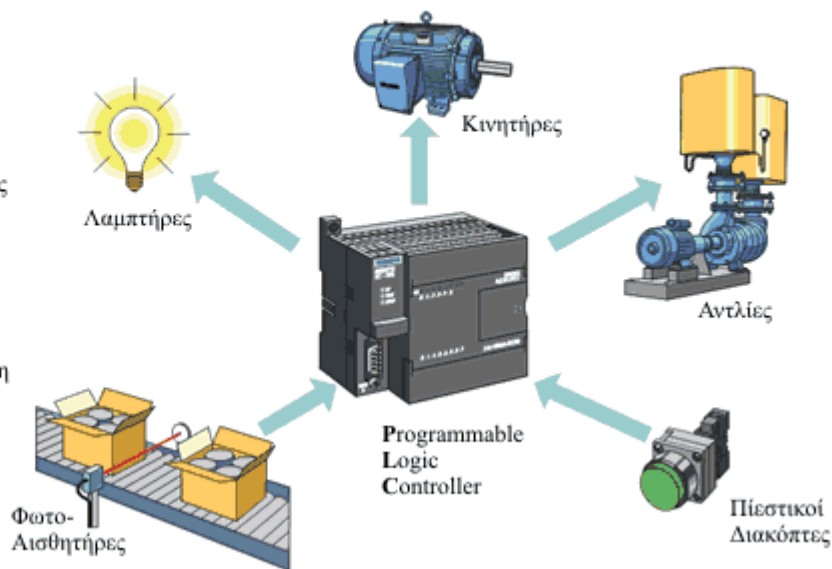
ΣΚΟΠΟΣ ΤΗΣ ΑΣΚΗΣΗΣ

Ο σκοπός αυτής της κατασκευής είναι να βοηθήσει στο να αποφεύγονται τα ατυχήματα σε σιδηροδρομικές διαβάσεις και να γίνεται ασφαλέστερη η διέλευσή τους με την χρήση κάποιων κυκλωμάτων και με τη βοήθεια των PLC.

ΤΙ ΕΙΝΑΙ ΤΑ PLC

PLCs (Programmable Logic Controllers)

Οι Προγραμματιζόμενοι Λογικοί Ελεγκτές, (PLC), ανήκουν στην οικογένεια των υπολογιστών. Χρησιμοποιούνται σε εμπορικές και βιομηχανικές εφαρμογές. Τα PLC's δρουν ως ελεγκτές μηχανών και διαδικασιών. Επιτηρούν τις εισόδους, λαμβάνουν αποφάσεις και ελέγχουν τις εξόδους τους με σκοπό την αυτοματοποίηση μηχανών και επεξεργασιών



Τα PLC (Programmable Logic Controllers) είναι μικρουπολογιστικά συστήματα που με κατάλληλο προγραμματισμό λογικών εξισώσεων επιλύουν προβλήματα αυτοματοποίησης.

Το PLC είναι μια ηλεκτρονική διάταξη η οποία από την άποψη της λειτουργίας θα μπορούσε να προσομοιωθεί με ένα πίνακα αυτοματισμού. Έχει δηλαδή εισόδους και εξόδους που συνδέονται με τα στοιχεία μια εγκατάστασης και βέβαια ένα αλγόριθμο που καθορίζει ότι κάποιος

συνδυασμός εισόδων παράγει ένα αποτέλεσμα στις εξόδους. Το ιδιαίτερο χαρακτηριστικό του PLC είναι ότι οι κανόνες που καθορίζουν τη συμπεριφορά των εξόδων δεν είναι σταθεροί όπως σε ένα κλασικό πίνακα αυτοματισμού αλλά μπορούν να μεταβάλλονται με επέμβαση στο πρόγραμμα που PLC χωρίς καμιά επέμβαση στο Hardware του συστήματος.

Σε ότι αφορά το υλικό τα PLC αποτελούνται από τη CPU η οποία περιέχει τη λογική του αυτοματισμού και η οποία διαβάζει την κατάσταση των καρτών εισόδου, ενεργοποιεί τις κάρτες εξόδου σύμφωνα με το πρόγραμμα που έχουμε υποθηκεύσει στη μνήμη του. Η CPU με τη βοήθεια της κάρτας εισόδου γνωρίζει κάθε στιγμή την κατάσταση ενός διακόπτη. Επιπλέον με τη βοήθεια της κάρτας εξόδου οπλίζει ένα ρελέ και μέσω αυτού ενεργοποιεί μια διάταξη.

Έτσι τώρα το σύνολο του συστήματος λειτουργεί ως εξής:

Αρχικά η CPU διαβάζει τις εισόδους, δηλαδή παρατηρεί κάθε είσοδο, και αν σε αυτή έχει εμφανιστεί η τάση καταχωρεί ένα λογικό 1 σε μια περιοχή της μνήμης του που είναι ειδική γι' αυτό το σκοπό. Στη συνέχεια εκτελείται το πρόγραμμα όπου εξετάζεται η τιμή των εισόδων και αποφασίζετε η τιμή των εξόδων η οποία καταχωρείται σε μια αντίστοιχη περιοχή μνήμης εξόδου.

Τέλος μια περιοχή μνήμης εξόδου μεταφέρεται στη κάρτα εξόδου και διεγείρει με τη σειρά της το ρελέ. Η διαδικασία αυτή επαναλαμβάνεται από την αρχή συνεχώς.

ΠΟΥ ΤΑ ΧΡΗΣΙΜΟΠΟΙΟΥΜΕ

Ασανσέρ , διωλιστήρια , καράβια , υδροηλεκτρικά φράγματα , συστήματα γεννητριών , ανεμογεννήτριες, βιολογικοί καθαρισμοί , αντλιοστάσια , φανάρια σε διασταυρώσεις δρόμων , κυλιόμενες σκάλες, τούνελ κυκλοφορίας αυτοκινήτων, «έξυπνα» σπίτια, συναγερμοί, γραμμές παραγωγής στην βιομηχανία, αυτόματες μηχανές συσκευασίας – εμφιάλωσης, γκαραζόπορτες, κυλιόμενες διαφημιστικές πινακίδες είναι μόνο λίγες από τις εφαρμογές που χρησιμοποιούνται τα PLC.

Τα τελευταία 20 χρόνια στην Ελλάδα χρησιμοποιούνται σε πάρα πολλές εφαρμογές τείνοντας να αντικαταστήσουν τον κλασικό αυτοματισμό.

Καλύπτουν λοιπόν ένα τεράστιο φάσμα εφαρμογών και για αυτό το λόγο πολλοί μηχανικοί από διάφορους κλάδους έχουν στραφεί στην ενασχόλησή με αυτά.

Τα τελευταία χρόνια βέβαια έχει αναπτυχθεί ένας αρκετά κερδοφόρος κλάδος που ονομάζεται <<τεχνολογία των έξυπνων σπιτιών>>. Αυτή η τεχνολογία αποτελεί στην ουσία ένα υποσύνολο των δυνατοτήτων των PLC όπως τα γνωρίζουμε στην κλασσική μορφή τους και οι μηχανικοί των PLC που ασχολούνται με αυτήν , προσαρμόζονται πολύ πιο εύκολα.

ΚΛΑΣΣΙΚΟΣ ΑΥΤΟΜΑΤΙΣΜΟΣ

Πριν την εμφάνιση των PLC, οι αυτοματισμοί κατά 100% μπορούσαν να πραγματοποιηθούν με κατάλληλη διασυρμάτωση ηλεκτρομηχανολογικών στοιχείων όπως ρελέ , χρονικά , βοηθητικές επαφές κ.τ.λ.. Αυτό συνήθως ονομάζεται έλεγχος μέσω καλωδίων (hard-wired control).

Για να πραγματοποιηθεί ένας κλασσικός αυτοματισμός πρέπει να σχεδιαστεί το κυκλωμά του , να οριστούν τα υλικά και να γίνει η εγκατάσταση. Κάθε λογική αυτοματισμού με λογικές AND , OR , XOR χρονικά , μετρητές κ.τ.λ πρέπει να συρματωθούν.

Αν γίνει ένα λάθος κατά την εγκατάσταση της λογικής του αυτοματισμού τότε τα καλώδια πρέπει να βγούν και να επανατοποθετηθούν σωστά. Μία αλλαγή λοιπόν σημαίνει είτε πρόσθεση είτε αφαίρεση υλικών καθώς και νέα συρμάτωση. Δηλαδή χαμένος χρόνος , κόπος και χρήμα.

ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ

Τα PLC's έκαναν την εμφάνισή τους στο τέλος της δεκαετίας του 1960 για τις ανάγκες αυτοματοποίησης της αμερικανικής βιομηχανίας αυτοκινήτων και η εφαρμογή τους τείνει να αντικαταστήσει πλήρως τον κλασικό αυτοματισμό , ο οποίος χρησιμοποιεί υλικά ηλεκτρομηχανικής τεχνολογίας. Από εκείνη την εποχή και μέχρι σήμερα έχουν αναπτυχθεί τόσο έτσι ώστε να αποτελούν αναπόσπαστο κομμάτι κάθε μορφής βιομηχανικού αυτοματισμού και όχι μόνο.

Βασικό στοιχείο του PLC είναι ο μικροεπεξεργαστής ο οποίος έχει την μορφή ολοκληρωμένου κυκλώματος και συνδυάζει τα πλεονεκτήματα του μικρού μεγέθους , του εύκολου προγραμματισμού , της υψηλής αξιοπιστίας και του χαμηλού κόστους.Ο μικροεπεξεργαστής μπορεί με τον κατάλληλο κάθε φορά προγραμματισμό να συμπεριφέρεται διαφορετικά και να εκτελεί μία ποικιλία λειτουργιών σύμφωνα με τις απαιτήσεις του προβλήματος. Σε αυτήν την ιδιότητα , δηλαδή ότι μπορεί να κάθε φορά να προγραμματίζεται διαφορετικά , οφείλει και την ονομασία του : “Programmable”.

Ο βασικός λόγος της ανάπτυξης μίας τέτοιας συσκευής ήταν το πολύ μεγάλο κόστος κατασκευής και συντήρησης των πολύπλοκων μονάδων αυτοματισμού που αποτελούνταν από μηχανολογικό και ηλεκτρικό εξοπλισμό όπως : ηλεκτρονόμοι (ρελέ) , βοηθητικές επαφές , χρονικά κ.τ.λ. Αυτές οι διατάξεις παρουσίαζαν συχνές βλάβες με αποτέλεσμα το συχνό σταμάτημα των μηχανών για τον εντοπισμό και την αντιμετώπιση της βλάβης. Εντοπισμός επίπονος και χρονοβόρος με δυσμενείς οικονομικές επιπτώσεις στις επιχειρήσεις.

Πρώτη η εταιρεία Betford πρότεινε μία διάταξη για την αντιμετώπιση των προβλημάτων που προαναφέρθηκαν.Η συσκευή ονομάστηκε **MODular DIgital CONtroller (MODICON)** και πολύ γρήγορα διατέθηκε στο εμπόριο με το όνομα MODICON 084. Λίγα χρόνια αργότερα το 1973 κάνει την εμφάνισή του το πρωτόκολλο επικοινωνίας Modbus της MODICON το οποίο δίνει την δυνατότητα στα PLC's να επικοινωνήσουν μεταξύ τους με σκοπό την ανταλλαγή δεδομένων.

Τέλος την δεκαετία του 1990 το πρότυπο **IEC 1131-3** καθορίζει τόσο τον αριθμό και την ονομασία των γλωσσών προγραμματισμού , όσο και τα εσωτερικά τους στοιχεία (σύμβολα εντολές κ.τ.λ).

ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΧΡΗΣΙΜΟΠΟΙΗΣΗΣ ΤΟΥ PLC

Συγκριτικά με το κλασσικό αυτοματισμό τα πλεονεκτήματα του αυτοματισμού με PLC είναι πολλά:

- ⌚ Τα PLC ως ηλεκτρικές συσκευές καταλαμβάνουν πολύ μικρότερο χώρο σε σχέση με τα υλικά του κλασσικού αυτοματισμού, καταναλώνουν δε πολύ λιγότερη ενέργεια από αυτά.
- ⌚ Οι αλλαγές γίνονται πολύ πιο εύκολα και γρήγορα.
- ⌚ Τα PLC's έχουν εσωτερικά διαγνωστικά και λειτουργίες δίνοντας πληροφορίες για την κατάσταση τους
- ⌚ Τα διαγνωστικά μπορούν να είναι διαθέσιμα και απομακρυσμένα μέσω δικτύου.
- ⌚ Οι εφαρμογές μπορούν να αρχειοθετηθούν και να εκτυπωθούν.
- ⌚ Οι εφαρμογές μπορούν να αντιγραφούν γρηγορότερα και πιο οικονομικά.
- ⌚ Η λειτουργία του αυτοματισμού μπορεί ν' αλλάξει πιο εύκολα σε οποιοδήποτε στάδιο θελήσουμε (μελέτη, κατασκευή).

Αν έπρεπε να δώσουμε το κύριο πλεονεκτημα-χαρακτηριστικό των PLC's, θα λέγαμε ότι είναι μία διάταξη , η οποία μπορεί κάθε φορά να επαναπρογραμματίζεται , με σκοπό να εκτελεί και μία διαφορετική εργασία , ανάλογα με τις απαιτήσεις που έχουμε να αντιμετωπίσουμε , σε αντίθεση με τον κλασσικό αυτοματισμό, ο οποίος κάθε φορά σχεδιάζεται για να μπορεί να εκτελεί ένα συγκεκριμένο ρόλο, χωρίς την δυνατότητα αλλαγής με γρήγορο και οικονομικό τρόπο.

ΣΤΑΔΙΑ ΕΡΓΑΣΙΑΣ

Τεχνική περιγραφή: καταγράφονται οι απαιτήσεις του πελάτη που έχει σήμερα και πιθανές απαιτήσεις που θα εμφανιστούν στο μέλλον έτσι ώστε να γίνει η κατασκευή με προδιαγραφές να μπορεί να βελτιωθεί στο μέλλον σύμφωνα με τις απαιτήσεις που θα εμφανιστούν. Βελτιώσεις μπορεί να γίνουν στο τομέα hardware και software ή μόνο στο τομέα software.

Είδος PLC: επιλογή το PLC που θα χρησιμοποιηθεί με χαμηλό κόστος και σύμφωνα με τα κριτήρια από την οικονομοτεχνική μελέτη που έγινε.

Εκπόνηση σχεδίου: γίνεται η κατασκευή του πίνακα όπου θα τοποθετηθεί το PLC.

Προγραμματισμός: γίνεται η υλοποίηση των προδιαγραφών και μια "μερική" δοκιμή.

Θέση σε λειτουργία: γίνεται τοποθέτηση του PLC με τον πίνακα, ακολούθως γίνονται οι συνδέσεις με ότι εξωτερικό κύκλωμα υπάρχει και μετά γίνεται ο προγραμματισμός. Τέλος γίνεται και η ολική δοκιμή της κατασκευής.

Φάκελος έργου: ο φάκελος αυτός περιέχει τα τελικά διορθωμένα σχέδια, και το πρόγραμμα μαζί με τα επεξηγηματικά σχόλια.

ΟΙΚΟΓΕΝΕΙΑ SIMATIC S7

S7-200

Χρησιμοποιείτε για εφαρμογές με μικρές απαιτήσεις σε όγκο προγράμματος και αριθμό σημάτων και εντολών.

Σαν πλεονεκτήματα έχουμε: ταχύτητα, ευελιξία, αποκρίσεις σε πραγματικό χρόνο, δικτύωση, επεκτάσιμα, παράλληλο Bus, δυνατότητα επεξεργασίας αναλογικών σημάτων, μικρές διαστάσεις, βρόγχους ελέγχου με PID, δυνατότητα προγραμματισμού σε όλες τις γλώσσες.

S7-300

Για μεσαίας κλίμακας εφαρμογές στις οποίες συγκαταλέγονται και οι περισσότερες εφαρμογές στη Ελλάδα.

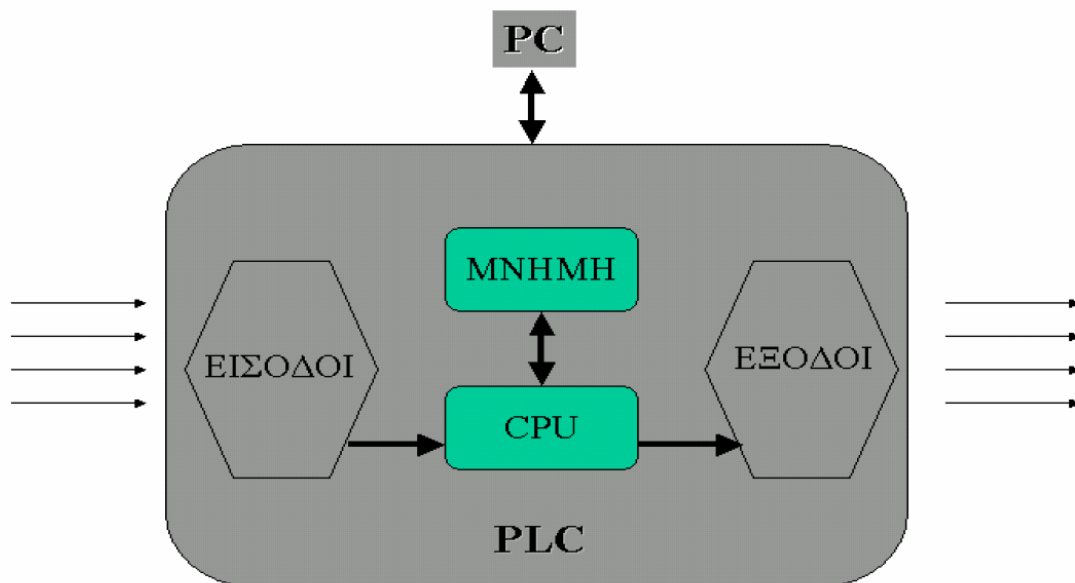
S7-400

Η πλέον ισχυρή σειρά για εφαρμογές υψηλών απαιτήσεων σε αριθμό σημάτων, χρόνο επεξεργασίας και μέγεθος προγράμματος.

ΤΑ ΒΑΣΙΚΑ ΜΕΡΗ ΤΟΥ PLC

Ένα PLC (Σχ.1) μοιάζει πολύ με ένα Η/Υ και αποτελείται από τέσσερα βασικά μέρη:

- **ΕΙΣΟΔΟΙ** : όπου λαμβάνονται από το εξωτερικό περιβάλλον
- **ΕΞΟΔΟΙ** : όπου δρούν σε συσκευές , λάμπες κ.τ.λ
- **ΜΝΗΜΗ** : όπου αποθηκεύεται το πρόγραμμα
- **CPU** : ο επεξεργαστής , ο οποίος <<διαβάζει>> τη λογική κατάσταση των εισόδων και στην συνέχεια θέτει σε λογική κατάσταση '1' ή '0' τις εξόδους, σε συνάρτηση με τις εντολές του προγράμματος.

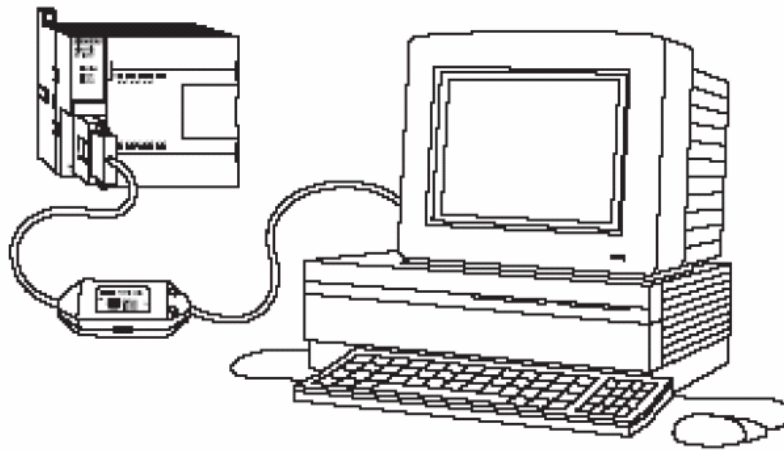


Σχ.1

Το PLC (Σχ.2) συνεργάζεται με έναν προσωπικό υπολογιστή (PC) ή ειδική συσκευή προγραμματισμού καθώς και ειδικό καλώδιο προγραμματισμού – επικοινωνίας προκειμένου να:

- Δημιουργήσει και να μεταφέρει το πρόγραμμα του χρήστη στη μνήμη του PLC.
- Παρακολουθεί την σωστή εκτέλεση του προγράμματος.

- *Εντοπίσει βλάβες στα στοιχεία εισόδου και εξόδου.*



Σχ.2

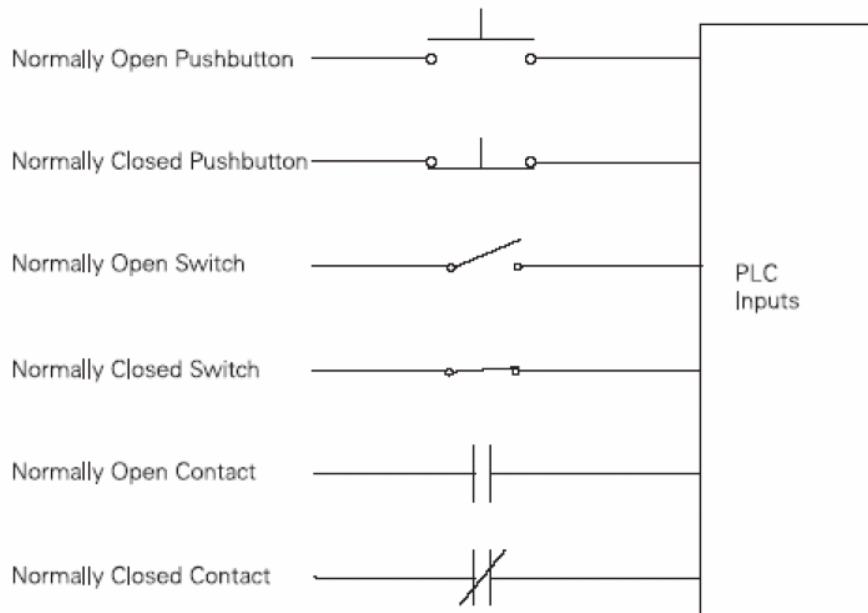
ΕΙΣΟΔΟΙ PLC

Σε έναν Η/Υ το πληκτρολόγιο , το scanner , το ποντίκι κ.α αποτελούν μονάδες εισόδου. Στο PLC οι εισόδοι χωρίζονται σε δύο κατηγορίες:

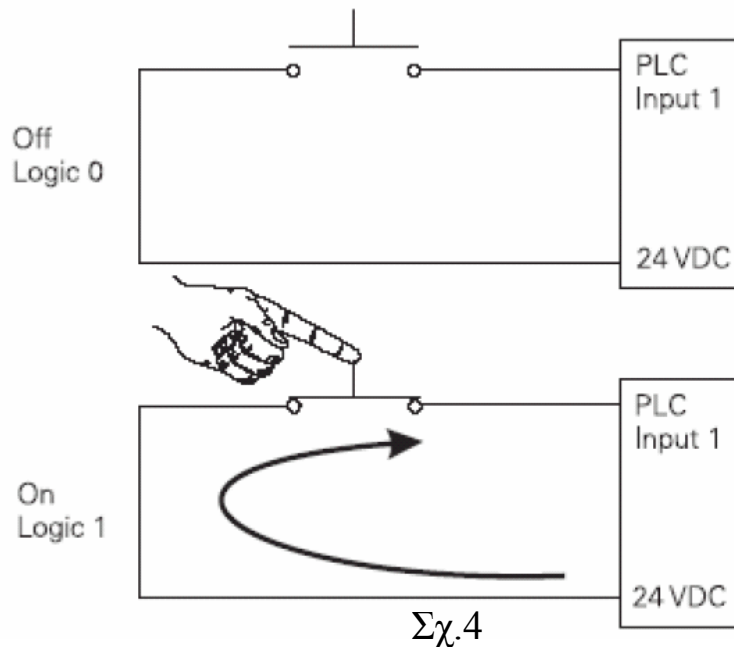
- Ψηφιακές εισόδοι
- Αναλογικές εισόδοι

Ψηφιακές εισόδοι (Digital Input)

Ψηφιακή είσοδος καλείται η είσοδος που μπορεί να βρεθεί σε συνθήκες είτε ON (1) είτε OFF (0) . Κουμπιά, διακόπτες , τερματικοί διακόπτες , φωτοκύτταρα , επαγωγικοί διακόπτες είναι παραδείγματα από ψηφιακούς αισθητήρες που είναι συνδεδεμένοι στις ψηφιακές εισόδους του PLC. Σε συνθήκη ON μία ψηφιακή είσοδος βρίσκεται σε λογικό 1 (logic 1) η λογικό υψηλό (logic high). Σε συνθήκη OFF η ψηφιακή είσοδος βρίσκεται σε λογικό 0 (logic 0) η λογικό χαμηλό (logic low). (Σχ.3)



Σχ.3



Ένα κανονικά ανοιχτό κουμπί (Normally Open (NO) pushbutton) χρησιμοποιείται στο παράδειγμα του σχήματος 4. Η μία πλευρά του κουμπιού είναι συνδεδεμένη με την πρώτη είσοδο του PLC (Input1). Η άλλη πλευρά είναι συνδεδεμένη με την εσωτερική τροφοδοσία 24VDC που διαθέτει το PLC. Η κατάσταση OFF ισχύει όταν δεν πατήσουμε το κουμπί. Η κατάσταση ON εκτελείται όταν πατήσουμε το κουμπί.

Όμως για να καλυφθούν και άλλες ανάγκες των ποικίλων εφαρμογών οι είσοδοι μπορούν να έχουν και άλλες τάσεις όπως : 220VAC , 48VDC , 110VAC. Ο λόγος ότι οι είσοδοι έχουν συνήθως τάση 24VDC είναι ότι , κατά την εγκατάσταση ή συντήρηση των καλωδίων, δεν θα κινδυνέψει άνθρωπος από ηλεκτροπληξία σε περίπτωση που τροφοδοτηθούν οι είσοδοι με 24VDC , γιατί είναι γνωστό πως η τάση αυτή είναι ασφαλής και δεν σκοτώνει.

Αναλογικές εισόδους (Analog Inputs)

Αναλογική είσοδος είναι ένα συνεχόμενα μεταβαλλόμενο μέγεθος τάσης ή ρεύματος. Τυπικές αναλογικές εισόδους που βρίσκουμε στα PLC είναι: 0mA έως 20mA , 4mA έως 20mA , 0Volts έως 10Volts.

Για παράδειγμα σε μια δεξαμενή το ύψος της οποίας μπορεί να μετρηθεί με την βοήθεια ενός μεταδότη ύψους (level transimeter) ο οποίος μετατρέπει το ύψος της δεξαμενής σε αναλογικό σήμα π.χ 0-10 V. Αυτό σημαίνει όταν η δεξαμενή είναι αδεια τότε έχουμε 0V. Όταν είναι γεμάτη 10 V , ενώ όταν είναι στην μέση 5 V . Η τιμή θα μειώνεται η θα αυξάνεται ανάλογα με το ύψος της δεξαμενής.Κάθε φορά λοιπόν γνωρίζουμε το ύψος και εκτελούμε ανάλογα στο προγράμμα μας.

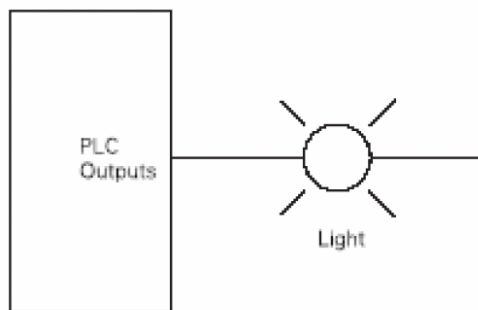
ΕΞΟΔΟΙ

Στο PLC οι εξόδοι χωρίζονται σε δύο κατηγορίες

- Ψηφιακές εξόδοι
- Αναλογικές εξόδοι

Ψηφιακές εξόδοι (Digital Outputs)

Στην ψηφιακή λογική γνωρίζουμε ότι ένας συνδιασμός ψηφιακών σημάτων έχουν ένα αποτέλεσμα εξόδου '0' ή '1'. Και στο PLC λοιπόν ένας λογικός συνδιασμός ηλεκτρικών ψηφιακών εισόδων με καταλληλο προγραμματισμό έχει ένα ψηφιακό αποτέλεσμα '0' η '1'. Παράδειγμα στο σχήμα 6 βλέπουμε μία λάμπα να γίνεται ON η OFF μέσω του PLC.



Σχ.6

Αναλογικές εξόδοι (Analog Outputs)

Όπως διακρίναμε το είδος της αναλογικής εισόδου με τον ίδιο τρόπο μπορούμε να αντιληφθούμε και την αναλογική έξοδο. Είναι λοιπόν ένα συνεχόμενο μεταβαλλόμενο μέγεθος τάσης – ρεύματος. Τυπικές αναλογικές εξόδοι που βρίσκουμε στα PLC είναι: 0mA έως 20mA , 4mA έως 20mA , 0Volt έως 10Volts. Για παράδειγμα μία αναλογική έξοδος μπορεί να επιδράσει σε έναν αναλογικό μετρητή , ή μία αναλογική πνευματική ηλεκτροβάνα. Μπορούμε πίο εύκολα βέβαια να καταλάβουμε την αναλογική έξοδο σαν μία μεταβαλλόμενη αντίσταση κατά την οποία μπορούμε να αυξομειώσουμε την ένταση φωτισμού σε μία λάμπα, την ταχύτητα ενός ανεμιστήρα κ.τ.λ.

MNHMH

Η μνήμη του PLC χωρίζεται σε δύο μέρη.

- **Μνήμη RAM (προσωρινή)**
- **Μνήμη EEPROM (μόνιμη επανεγράψιμη)**

Και δύο μνήμες μετριώνται σε bytes όπως ακριβώς και στα PC.

RAM

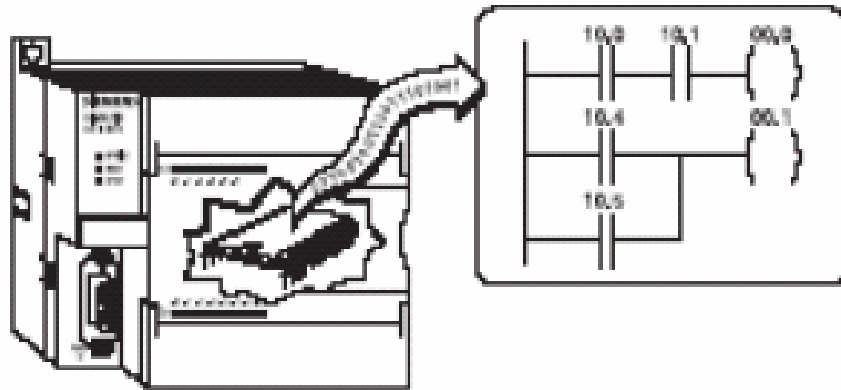
Όπως και σε ένα PC έτσι και στο PLC, κάθε πρόγραμμα που εκτελείται από την CPU, εκτελείται με την βοήθεια της μνήμης όπου το πρόγραμμα και τα δεδομένα (DATA) αποθηκεύονται εκεί προσωρινά. Το πρόγραμμα δεν χάνεται άμεσα με την μη τροφοδοσία του PLC, γιατί συνήθως υπάρχει: είτε ειδική μπαταρία λιθίου είτε ειδικός πυκνωτής που αποθηκεύουν ενέργεια να συντηρούν τα δεδομένα για κάποιο χρονικό διάστημα. Με την λήξη της ενέργειας και της μη τροφοδοσίας του PLC το πρόγραμμα και τα δεδομένα χάνονται.

EEPROM

Η EEPROM είναι μία μνήμη BACKUP σε ένα ειδικό ολοκληρωμένο όπου το πρόγραμμα που είναι αποθηκευμένο εκεί δεν χάνεται ποτέ. Αν χαθεί από την RAM για οποιοδήποτε λόγο, τότε μπορούμε με μία διαδικασία να ανακτήσουμε το πρόγραμμα από την EEPROM στην RAM.

ΠΡΟΣΟΧΗ : Μόνο ένα πρόγραμμα μπορεί να αποθηκευτεί στην RAM η στην EEPROM. Αν θελήσουμε να αλλάξουμε πρόγραμμα θα πρέπει να γνωρίζουμε πως το παλιό, αν δεν το σώσουμε, θα χαθεί όταν αρχίζουμε να <<κατεβάζουμε>> (download) το καινούργιο στην μνήμη.

CPU(Control Process Unit)



Σε αυτή αποθηκεύετε και εκτελείτε κυκλικά το πρόγραμμα του χρηστή. Με βάση τις τιμές που διαβάζονται από τις εισόδους καθώς και τις από πριν υποθηκευμένες άλλες τιμές, παράγονται οι αποφάσεις που θα εκτελεστούν για να υλοποιηθεί ο αυτοματισμός της εγκατάστασης.

Εσωτερικά στοιχεία:

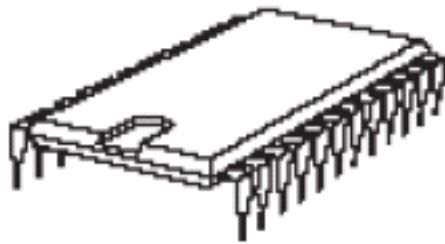
1. **Ο μικροεπεξεργαστής**, ο οποίος εκτελεί το πρόγραμμα που έχει μέσα στη μνήμη του και ελέγχει τη σωστή λειτουργία όλων των μονάδων που είναι συνδεδεμένες σε αυτόν.
2. **Η μνήμη**, η οποία λογικά χωρίζεται σε διάφορες περιοχές εκ των οποίων οι σημαντικότερες είναι:
 - **Μνήμη του χρηστή**, όπου αποθηκεύεται το πρόγραμμα που εμείς έχουμε γράψει για τον αυτοματισμό της εγκατάστασης.
 - **Μνήμη για το λειτουργικό σύστημα**, όπου τρέχει το πρόγραμμα για την λειτουργία του ίδιου του PLC.
 - **Μνήμη για χρονικά απεριθμητές, βοηθητικά.**
 - **Μνήμη απεικόνισης της περιφέρειας**, όπου καταχωρείται η κατάσταση των σημάτων εισόδου και εξόδου, το τι γίνεται δηλαδή εκτός PLC.

3. Κάθε κεντρική μονάδα έχει κατ'ελάχιστο:
- Διακοπή με κλειδί RUN-P/RUN/STOP/MRES.
 - Ενδεικτικά LED για την κατάσταση της CPU.
 - Θέση για σύνδεση συσκευής προγραμματισμού ή συσκευών καταγραφής – απεικόνισης.
 - Θέση για τοποθέτηση εξωτερικής μνήμης.

Όταν ο διακόπτης είναι στη θέση RUN-P είναι δυνατό να τροποποιήσουμε το πρόγραμμα της CPU. Στη θέση RUN μπορούμε μόνο να διαβάσουμε πληροφορίες από την CPU και την χρησιμοποιούμε μόνο για λογούς ασφαλείας.

Στη θέση STOP το πρόγραμμα που υπάρχει στη CPU δεν εκτελείτε ενώ στη θέση MRES μπορούμε να σβήσουμε το πρόγραμμα που υπάρχει στη CPU.

FIRMWARE

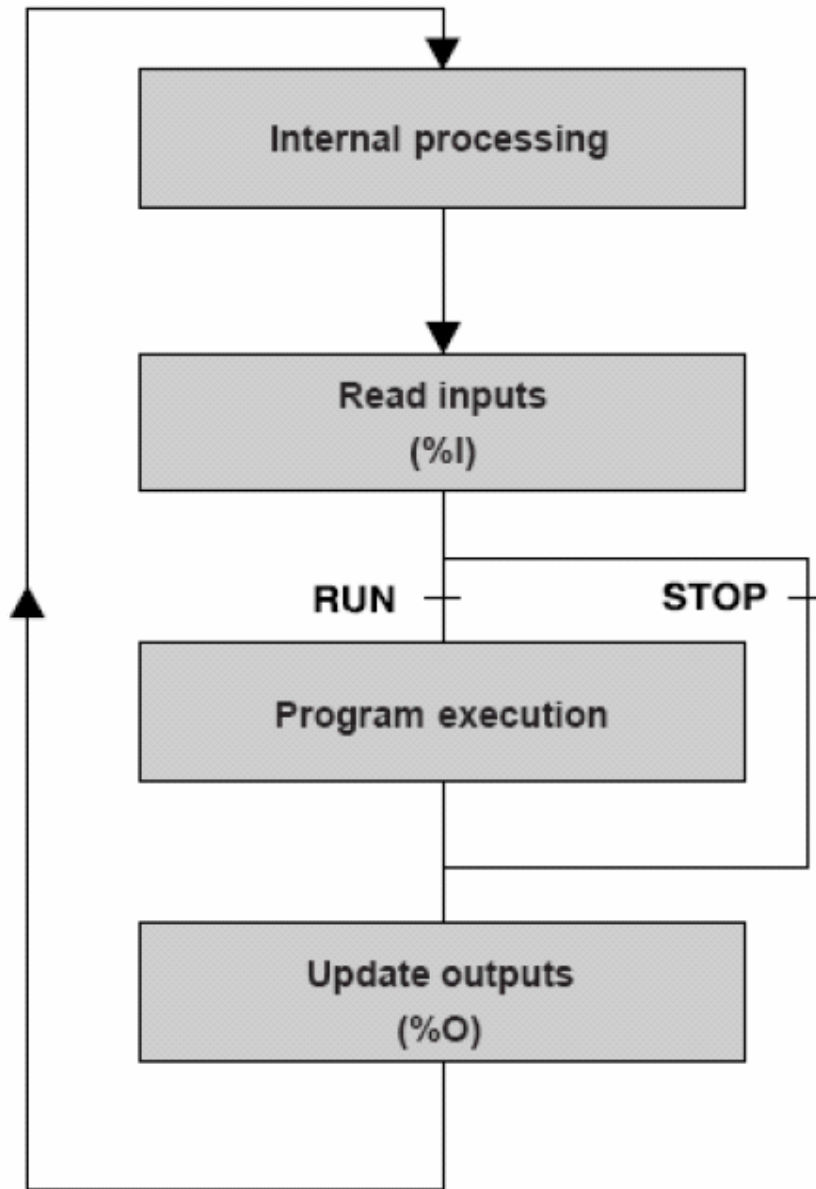


Το λειτουργικό σύστημα (Firmware) είναι ένα ειδικό software αποθηκευμένο σε μία EPROM μέσα στο PLC και παραδίδεται σαν κομμάτι του hardware. Το λειτουργικό σύστημα δίνει στο PLC τις βασικές του λειτουργίες όπως και σε ένα PC.

Είναι εφικτό να υπάρχουν αναβαθμίσεις στο λειτουργικό τις οποίες μπορούμε να φορτώσουμε μέσω ειδικών software που μας παρέχει η εκάστοτε εταιρεία.

ΛΕΙΤΟΥΡΓΙΑ ΣΥΣΤΗΜΑΤΟΣ PLC

- Διαδικασία κυκλικής επεξεργασίας



Στο παραπάνω σχήμα βλέπουμε πώς λειτουργεί το σύστημα ενός PLC. Αρχικά η CPU εκτελεί κάποιες εσωτερικές λειτουργίες (internal processing). Στην συνέχεια «διαβάζει» όλες τις εισόδους σειριακά ξεκινώντας από την πρώτη κατά σειρά και μετά μία προς μία. Παρατηρεί την κάθε είσοδο, και αν σε αυτή έχει εμφανισθεί τάση καταχωρεί λογικό '1', αν όχι καταχωρεί λογικό '0' σε μια περιοχή της μνήμης, που προβλέπεται ειδικά γι' αυτό τον σκοπό και ονομάζεται **καταχωρητής εισόδων**. Η περιοχή αυτή περιέχει κάθε στιγμή την κατάσταση των εισόδων και λειτουργεί σαν ενδιάμεσος σταθμός ανάμεσα στον "έξω κόσμο" και την CPU.

Στην συνέχεια εάν επιτρέψουμε στο PLC την εκτέλεση του προγράμματος με την εντολή RUN εκτελείται το πρόγραμμα (program execution) . Δηλαδή αποφασίζεται μέσω των λογικών εξισώσεων του προγράμματος του χρήστη και των τιμών των εισόδων η τιμή των εξόδων.

Το αποτέλεσμα των εξόδων καταχωρείται σειριακά ξεκινώντας από την πρώτη και μετά μία προς μία στην περιοχή μνήμης που καλείται **καταχωρητής εξόδων**. Η CPU τότε ενημερώνει και τις πραγματικές εξόδους (update outputs) όπου και καταχωρεί το λογικό 0 η 1.

Τέλος εάν λήξει η εκτέλεση του προγράμματος με την εντολή STOP ,τότε απλά ενημερώνονται οι έξοδοι με τιμές που δεν έχουν σχέση με το αποτέλεσμα του προγράμματος αλλά μόνο με εξαναγκασμένη τιμή 0 η 1 που μπορεί να δώσει ο χρήστης στη έξοδο μέσω του περιβάλλοντος προγραμματισμού, ανεξάρτητα από το πρόγραμμα.

Η διαδικασία αυτή επαναλαμβάνεται από την αρχή και διαρκώς .Δηλαδή ξαναδιαβάζεται η είσοδος που μπορεί τώρα να έχει διαφορετική τιμή-εκτελείται το πρόγραμμα και τέλος ενημερώνονται οι έξοδοι.

Σε αυτό εδώ το σημείο θα πρέπει να τονίσουμε μερικά λεπτά σημεία όπου γίνονται παρανοήσεις από μηχανικούς που ασχολούνται με αμιγώς μικροπολογιστικά συστήματα που προγραμματίζονται με γλώσσες χαμηλού επιπέδου.

- 1) Η CPU ακούραστα και για πάντα , εφ'όσον το PLC βρίσκεται σε κατάσταση RUN ,εκτελεί τον κύκλο λειτουργίας του όπως τον περιγράψαμε πλήρως.
- 2) Τίποτα δεν εκτελείται ταυτόχρονα. Όλα εκτελούνται σειριακά αφού έχουμε μία CPU για όλες τις δουλειές. Οι γρήγοροι χρόνοι που εκτελούνται οι διαδικασίες μας δίνουν την εντύπωση πως όλα γίνονται ταυτόχρονα, αλλά αυτό είναι μια ψευδαίσθηση.
- 3) Περίπτωση τερματισμού της κυκλικής λειτουργίας του προγράμματος υπάρχει εφ'όσον ο χρήστης δώσει τέτοια εντολή από το πρόγραμμα

ΤΙ ΣΗΜΑΙΝΕΙ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ PLC

Προγραμματισμός του PLC σημαίνει να δημιουργήσουμε μία σειρά από λογικές εντολές οι οποίες λύνουν ένα πρόβλημα συστήματος αυτοματισμού.

Σαν μικροπολογιστικό σύστημα κάθε PLC έχει μία συγκεκριμένη γλώσσα μηχανής (γλώσσα με εντολές γραμμένες σε μορφή άμεσα κατανοητή από τον επεξεργαστή του PLC), σύμφωνα με την αρχιτεκτονική του hardware. Είναι δυνατόν θεωρητικά να προγραμματίσουμε ένα PLC γράφοντας εντολές σε γλώσσα μηχανής.

Κάτι τέτοιο όμως θα έκανε τα PLC να προγραμματίζονται με επίπονο τρόπο και μόνο από ανθρώπους με βαθιά γνώση στην δομή και την λειτουργία των διαφόρων επεξεργαστών.

Ο σκοπός των PLC είναι να χρησιμοποιούνται από μεγάλες ομάδες ανθρώπων με τρόπο όσο το δυνατόν πιά εύκολο και φιλικό.

Για το σκοπό αυτό , οι κατασκευαστές των PLC πρότειναν διάφορες γλώσσες προγραμματισμού που μπορούν να χρησιμοποιηθούν από ανθρώπους που σχετίζονται με τον έλεγχο συστημάτων.

Η επιλογή της γλώσσας προγραμματισμού εξαρτάται από την εμπειρία και την γνώση του χρήστη σε ψηφιακά ηλεκτρονικά , σε υπολογιστές , σε συστήματα αυτοματισμού που λειτουργούν με κλασσικό τρόπο και φυσικά εξαρτάται από την φύση του προβλήματος που έχουμε να αντιμετωπίσουμε.

Το μεγάλο πρόβλημα με τις γλώσσες προγραμματισμού είναι ότι η κάθε κατασκευαστική εταιρεία παρά το γεγονός ότι χρησιμοποιεί μία ή δύο ή περισσότερες από τις γνωστές γλώσσες προγραμματισμού , στην πραγματικότητα υπάρχουν από μικρές μέχρι τεράστιες διαφορές από εταιρία σε εταιρία , με αποτέλεσμα ο χρήστης να αντιμετωπίζει σοβαρό πρόβλημα όταν θέλει να κάνει χρήση ενός PLC διαφορετικής εταιρείας.

Την λύση σε αυτό το μεγάλο και σοβαρό πρόβλημα ήρθε να αντιμετωπίσει τις αρχές της δεκαετίας του 1990 το πρότυπο IEC 1131-3 που καθορίζει τόσο τον αριθμό και την ονομασία των γλωσσών προγραμματισμού, όσο και τα εσωτερικά του στοιχεία (σύμβολα , εντολές κ.τ.λ).

Βέβαια τελικά και για λόγους ανταγωνισμού λίγες εταιρείες συμμορφώθηκαν με το πρότυπο και εξακολουθούν να διαφοροποιούνται από αυτό. Οι γλώσσες προγραμματισμού μπορούν να ταξινομηθούν σε γραφικές και μη γραφικές ανάλογα με το είδος των στοιχείων που χρησιμοποιούν:

Γλώσσες προγραμματισμού σύμφωνα με το πρότυπο IEC 1131-3			
Γλώσσες προγραμματισμού	Σύντμηση ονόματος	Είδος Γλώσσας	Απόδοση στην Ελληνική γλώσσα
Instruction List	IL-STL	Μη γραφική	Λίστα εντολών
Ladder Diagram	LD	Γραφική	Σχέδιο επαφών
Function Block Diagram	FBD	Γραφική	Λογικό διάγραμμα
Structure text	ST	Μη γραφική	Δομημένου κειμένου
Sequential Function Chart	SFC	Γραφική	Διάγραμμα ροής

Όταν θέλουμε να διευθηνσιοδοτήσουμε τις εισόδους και τις εξόδους του PLC σε ένα πρόγραμμα, τότε ακολουθούμε τους παρακάτω συμβολισμούς που είναι σύμφωνοι με το πρότυπο IEC 1131-3.

I 0.0

I = Είσοδος (Input)

0 = Είναι η πρώτη ομάδα εισόδων που συνήθως βρίσκεται στην ίδια πλακέτα που είναι και η CPU.

. = Διαχωρισμός ομάδας εισόδων από την είσοδο.

0 = Η πρώτη είσοδος

Q 0.0

Q = Εξοδος (Output)

0 = Είναι η πρώτη ομάδα εξόδων που συνήθως βρίσκεται στην ίδια πλακέτα που είναι και η CPU.

. = Διαχωρισμός ομάδας εξόδων από την έξοδο.

0 = Η πρώτη έξοδο

Το S7 προσφέρει πολλούς τύπους από εντολές που μας επιτρέπουν να ανταπεξέρθουμε σε μια μεγάλη ποικιλία από αυτοματισμούς. Υπάρχουν 2 βασικά σεντ εντολών διαθέσιμα σε μια S7 CPU:

- SIMATIC
- IEC

Επίσης το λογισμικό του υπολογιστή μας STEP 7-Micro/Win 32 παρέχει διαφορετικές επιλογές για Editors που μας επιτρέπουν να δημιουργήσουμε προγράμματα ελέγχου με αυτές τις εντολές. Για παράδειγμα μπορεί να προτιμήσουμε να φτιάξουμε προγράμματα σε ένα περιβάλλον με σχήματα(LAD-FBD) ενώ μπορούμε να δημιουργήσουμε προγράμματα με τον γνωστό μας τρόπο, δηλαδή σε μορφή (STL).

Έχουμε δυο βασικές επιλογές για την δημιουργία των προγραμμάτων μας:

- Το σεντ εντολών που θέλουμε να χρησιμοποιήσουμε(SIMATIC ή IEC).
- Το τύπο του Editor που θέλουμε να χρησιμοποιήσουμε(STL,LAD,FBD)

STATEMENT LIST EDITOR(STL)

Ο STL μας επιτρέπει να δημιουργήσουμε προγράμματα ελέγχου με την είσοδο του στις εντολές μνήμης. Μας επιτρέπει να δημιουργήσουμε προγράμματα που δεν μπορούμε να δημιουργήσουμε με LAD και FBD. Αυτό συμβαίνει επειδή η STL προγραμματίζει την CPU στη θεμελίωση γλωσσά της ενώ οι άλλοι Editors πρέπει να προσαρμόσουν τα σχήματα τους ώστε να γίνουν τα διαγράμματα τους σωστά.

Η CPU εκτελεί κάθε εντολή με τη σειρά που υπαγορεύεται στο πρόγραμμα μας από την αρχή προς το τέλος και ξαναρχίζει από την αρχή.

Π.χ.

LD I0.0 /φόρτωσε την είσοδο I0.0
LD I0.1 /φόρτωσε την είσοδο I0.1
O I0.0 /είτε όταν κλείσει ο διακόπτης I0.0
O I0.1 /είτε όταν κλείσει ο διακόπτης I0.1
= Q0.0 /ενεργοποιεί την έξοδο Q0.0
END /τέλος προγράμματος.

ΒΑΣΙΚΑ ΣΤΟΙΧΕΙΑ ΓΙΑ ΤΗΝ ΔΟΜΗ ΕΝΟΣ ΠΡΟΓΡΑΜΜΑΤΟΣ

Η S7 CPU εκτελεί το πρόγραμμα συνεχώς για να ελέγξει μια λειτουργία που του έχουμε αναθέσει. Δημιουργούμε το πρόγραμμα αυτό με το STEP 7-Micro/Win 32 και να το φορτώνουμε στη CPU. Από το κυρίως πρόγραμμα μπορούμε να καλέσουμε διαφορετικές υπορουτίνες ή Interrupts ρουτίνες.

ΟΡΓΑΝΩΣΗ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ

Προγράμματα για μια S7 CPU κατασκευάζονται από τρία βασικά στοιχεία, το βασικό πρόγραμμα, τις υπορουτίνες και τις Interrupts ρουτίνες.

Ένα πρόγραμμα σε S7 δομείται στα ακόλουθα στοιχεία οργάνωσης:

- **Το κυρίως πρόγραμμα.** Το κυρίως σώμα του προγράμματος είναι εκεί που τοποθετούμε τις εντολές που ελέγχουν την εφαρμογή μας. Οι εντολές του κυρίως προγράμματος εκτελούνται διαδοχικά μια φορά σε κάθε σάρωση της CPU.
- **Interrupts ρουτίνες.** Αυτά τα προαιρετικά στοιχεία του προγράμματος μας εκτελούνται σε κάθε εμφάνιση ενός περιστατικού διακόπτη.
- **Υπορουτίνες.** Αυτά τα προαιρετικά στοιχεία του προγράμματος μας εκτελούνται μόνο όταν εκτελούνται μόνο όταν καλούνται από κυρίως προγράμματα η μια Interrupt ρουτίνα.

ΛΟΓΙΚΕΣ ΕΝΤΟΛΕΣ

ΕΝΤΟΛΗ AND (A)

Η εντολή αυτή ρωτάει **αν και** στη είσοδο έχουμε λογικό "1". Αν ισχύει αυτή η συνθήκη ενεργοποιείται η έξοδος.

Π.χ.

```
LD  I0.0
A   I0.0
=   Q0.0
END
```

ΕΝΤΟΛΗ AND NOT (AN)

Η εντολή αυτή ρωτάει **αν και** στη είσοδο έχουμε λογικό "0". Αν ισχύει αυτή η συνθήκη ενεργοποιείται η έξοδος.

Π.χ.

```
LDN I0.0
AN  I0.0
=   Q0.0
END
```

ΕΝΤΟΛΗ OR (O)

Η εντολή αυτή ρωτάει **είτε** στη είσοδο έχουμε λογικό "1". Αν ισχύει αυτή η συνθήκη ενεργοποιείται η έξοδος.

Π.χ.

```
LD  I0.0
O   I0.0
=   Q0.0
END
```

ΕΝΤΟΛΗ OR NOT (ON)

Η εντολή αυτή ρωτάει **είτε** στη είσοδο έχουμε λογικό "0". Αν ισχύει αυτή η συνθήκη ενεργοποιείται η έξοδος.

Π.χ.

```
LDN I0.0
ON I0.0
= Q0.0
END
```

ΕΝΤΟΛΗ NOT

Η εντολή αυτή κάνει αλλαγή στη κατάσταση από λογικό "1" σε λογικό "0" και αντίστροφα.

Π.χ.

```
LD I0.0
A I0.0
NOT
= Q0.0
END
```

ΕΝΤΟΛΗ ΑΥΤΟΣΥΓΚΡΑΤΗΣΗΣ SET (S)

Η εντολή αυτή ενεργοποιεί μια έξοδο στιγμιαία με λογικό "1". Και μένει S μέχρι να την απενεργοποιήσουμε.

Συνταξη της εντολης:

S bit,N

Π.χ.

```
LD I0.0
S Q0.0,1
END
```

ΕΝΤΟΛΗ ΑΥΤΟΣΥΓΚΡΑΤΗΣΗΣ RESET (R)

Η εντολή αυτή απενεργοποιεί μια έξοδο στιγμιαία με λογικό "0" που ήταν ενεργοποιημένη. Και μένει R μέχρι να την ενεργοποιήσουμε ξανά.

Συνταξη της εντολης:

R bit,N

Π.χ.

```
LD I0.0  
R Q0.0,1  
END
```

ON DELAY TIMER (TON)

Αυτό το χρονικό ενεργοποιείται εφόσον συμβεί μεταβολή από το λογικό "0" σε λογικό "1". Τα αποτελέσματα στη έξοδο θα φανούν όταν περάσει ο χρόνος με την προϋπόθεση ότι δεν έχει συμβεί άλλη μεταβολή.

Τύπος χρονικού	Τιμες σε ms	max σε sec	Αριθμος Timer
TON	1 ms	32,767 sec	T32,T96
	10 ms	327,67 sec	T33-T36,T97-T100
	100 ms	3276,7 sec	T37-T63,T101-T255

Ο Timer TON συντάσσεται ως εξής: TON Txxx, PT

```
LD I0.0  
TON T33,+2
```


OFF DELAY TIMER (TOF)

Αυτό το χρονικό απενεργοποιείται εφόσον συμβεί μεταβολή από το λογικό "1" σε λογικό "0". Τα αποτελέσματα στη έξοδο θα φανούν όταν περάσει ο χρόνος με την προϋπόθεση ότι δεν έχει συμβεί άλλη μεταβολή. Με την επόμενη μεταβολή από "0" σε "1" μηδενίζεται και ξεκινάει η ίδια διαδικασία.

Τύπος χρονικού	Τιμες σε ms	max σε sec	Αριθμος Timer
TOF	1 ms	32,767 sec	T32,T96
	10 ms	327,67 sec	T33-T36,T97-T100
	100 ms	3276,7 sec	T37-T63,T101-T255

Ο Timer TOF συντάσσεται ως εξής: TOF Txxx, PT

LD I0.0

TOF T37,+200

ON DELAY TIMER ME ΣΥΓΚΡΑΤΗΣΗ (TONR)

Αυτό το χρονικό ενεργοποιείται εφόσον συμβεί μεταβολή από το λογικό "0" σε λογικό "1". Για όση ώρα έχουμε κατάσταση "1" το χρονικό θα μετράει ωσότου φτάσει τη τελική τιμή που του έχουμε δώσει μέσα στο πρόγραμμα. Τα αποτελέσματα στη έξοδο θα φανούν όταν περάσει ο χρόνος με την προϋπόθεση ότι δεν έχει συμβεί άλλη μεταβολή.

Τύπος χρονικού	Τιμες σε ms	max σε sec	Αριθμος Timer
TONR	1 ms	32,767 sec	T0,T64
	10 ms	327,67 sec	T1-T4,T65-T68
	100 ms	3276,7 sec	T5-T31,T69-T95

Ο Timer TONR συντάσσεται ως εξής: TONR Txxx, PT

LD I0.0

TONR T33,+2

ΑΥΞΗΣΗ ΠΕΡΙΕΧΟΜΕΝΟΥ ΑΠΑΡΗΘΜΗΤΗ (CTU)

Η εντολή CTU μετράει προς τα πάνω ως τη μέγιστη τιμή που έχουμε ορίσει στη είσοδο που χρησιμοποιούμε για τον απαριθμητή. Το περιεχόμενο του απαριθμητή αυξάνεται κατά 1 κάθε φορά που έχουμε μεταβολή από το "0" σε "1".

Ο απαριθμητής CTU συντάσσεται ως εξής: CTU Cxxx, PV

```
LD I0.0
CTU C1, +3
LD C1
= Q0.0
END
```

ΜΕΙΩΣΗ ΠΕΡΙΕΧΟΜΕΝΟΥ ΑΠΑΡΗΘΜΗΤΗ (CTD)

Η εντολή CTD μετράει προς τα κάτω ως τη προκαθορισμένη τιμή που έχουμε ορίσει στη είσοδο που χρησιμοποιούμε για τον απαριθμητή. Το περιεχόμενο του απαριθμητή μειώνεται κατά 1 κάθε φορά που έχουμε μεταβολή από το "0" σε "1".

Ο απαριθμητής CTD συντάσσεται ως εξής: CTD Cxxx, PV

```
LD I0.0
CTD C34, +5
LD C34
= Q0.0
END
```

ΑΥΞΗΣΗ & ΜΕΙΩΣΗ ΠΕΡΙΕΧΟΜΕΝΟΥ ΑΠΑΡΙΘΜΗΤΗ (CTUD)

Η εντολή CTUD είναι ένας συνδυασμός των παραπάνω εντολών και με αυτή τη δυνατότητα να αυξήσουμε και να μειώσουμε το περιεχόμενο ενός απαρithμητή. Σε αυτό χρειαζόμαστε δυο διακόπτες οπού ο ένας μειώνει και ο άλλος αυξάνει το περιεχόμενο του απαρithμητή.

Ο απαρithμητής CTUD συντάσσεται ως εξής: CTUD Cxxx, PV

```
LD I0.0
LD I0.1
CTU C110, +5
LD C110
= Q0.0
END
```

ΣΥΓΚΡΙΣΗ ΑΚΕΡΑΙΩΝ (INTEGER)

- Η εντολή ισότητας (INT1=INT2) συντάσσεται ως εξής:

LDW= INT1,INT2

- Η εντολή διαφόρου από (INT1<>INT2) συντάσσεται ως εξής:

LDW<> INT1,INT2

- Η εντολή μικρότερου από (INT1<INT2) συντάσσεται ως εξής:

LDW< INT1,INT2

- Η εντολή μεγαλύτερου από (INT1>INT2) συντάσσεται ως εξής:

LDW> INT1,INT2

- Η εντολή μεγαλύτερου ή ίσου από (INT1>=INT2) συντάσσεται ως εξής:

LDW>= INT1,INT2

- Η εντολή μικρότερου ή ίσου από (INT1<=INT2) συντάσσεται ως εξής:

LDW<= INT1,INT2

ΣΥΓΚΡΙΣΗ ΠΡΑΓΜΑΤΙΚΩΝ (REAL)

- Η εντολή ισότητας ($INT1=INT2$) συντάσσεται ως εξής:

LDR= INT1,INT2

- Η εντολή διαφόρου από ($INT1\langle>INT2$) συντάσσεται ως εξής:

LDR<> INT1,INT2

- Η εντολή μικρότερου από ($INT1<INT2$) συντάσσεται ως εξής:

LDR< INT1,INT2

- Η εντολή μεγαλύτερου από ($INT1>INT2$) συντάσσεται ως εξής:

LDR> INT1,INT2

- Η εντολή μεγαλύτερου ή ίσου από ($INT1>=INT2$) συντάσσεται ως εξής:

LDR>= INT1,INT2

- Η εντολή μικρότερου ή ίσου από ($INT1>=INT2$) συντάσσεται ως εξής:

LDR<= INT1,INT2

ΣΥΓΚΡΙΣΗ BYTE

- Η εντολή ισότητας ($INT1=INT2$) συντάσσεται ως εξής:

LDB= INT1,INT2

- Η εντολή διαφόρου από ($INT1\langle>INT2$) συντάσσεται ως εξής:

LDB<> INT1,INT2

- Η εντολή μικρότερου από ($INT1<INT2$) συντάσσεται ως εξής:

LDB< INT1,INT2

- Η εντολή μεγαλύτερου από ($INT1 > INT2$) συντάσσεται ως εξής:

LDB> INT1,INT2

- Η εντολή μεγαλύτερου ή ίσου από ($INT1 \geq INT2$) συντάσσεται ως εξής:

LDB>= INT1,INT2

- Η εντολή μικρότερου ή ίσου από ($INT1 \leq INT2$) συντάσσεται ως εξής:

LDB<= INT1,INT2

ΕΝΤΟΛΗ MOVE BYTE

Η εντολή MOVE BYTE μεταφέρει το byte εισόδου στο byte εξόδου χωρίς να επηρεάζει το byte εισόδου από τη μεταφορά στη έξοδο.

Η εντολή MOVE BYTE συντάσσεται ως εξής:

MOVEB IN,OUT

ΕΝΤΟΛΗ MOVE WORD

Η εντολή MOVE WORD μεταφέρει το word εισόδου στο word εξόδου χωρίς να επηρεάζει το word εισόδου από τη μεταφορά στη έξοδο.

Η εντολή MOVE WORD συντάσσεται ως εξής:

MOVEW IN,OUT

ΕΝΤΟΛΗ MOVE DOUBLE WORD

Η εντολή MOVE DOUBLE WORD μεταφέρει τη DW εισόδου στη DW εξόδου χωρίς να επηρεάζει το DW εισόδου από τη μεταφορά στη έξοδο.

Η εντολή MOVE DOUBLE WORD συντάσσεται ως εξής:

MOVED IN,OUT

ΕΝΤΟΛΗ MOVE REAL

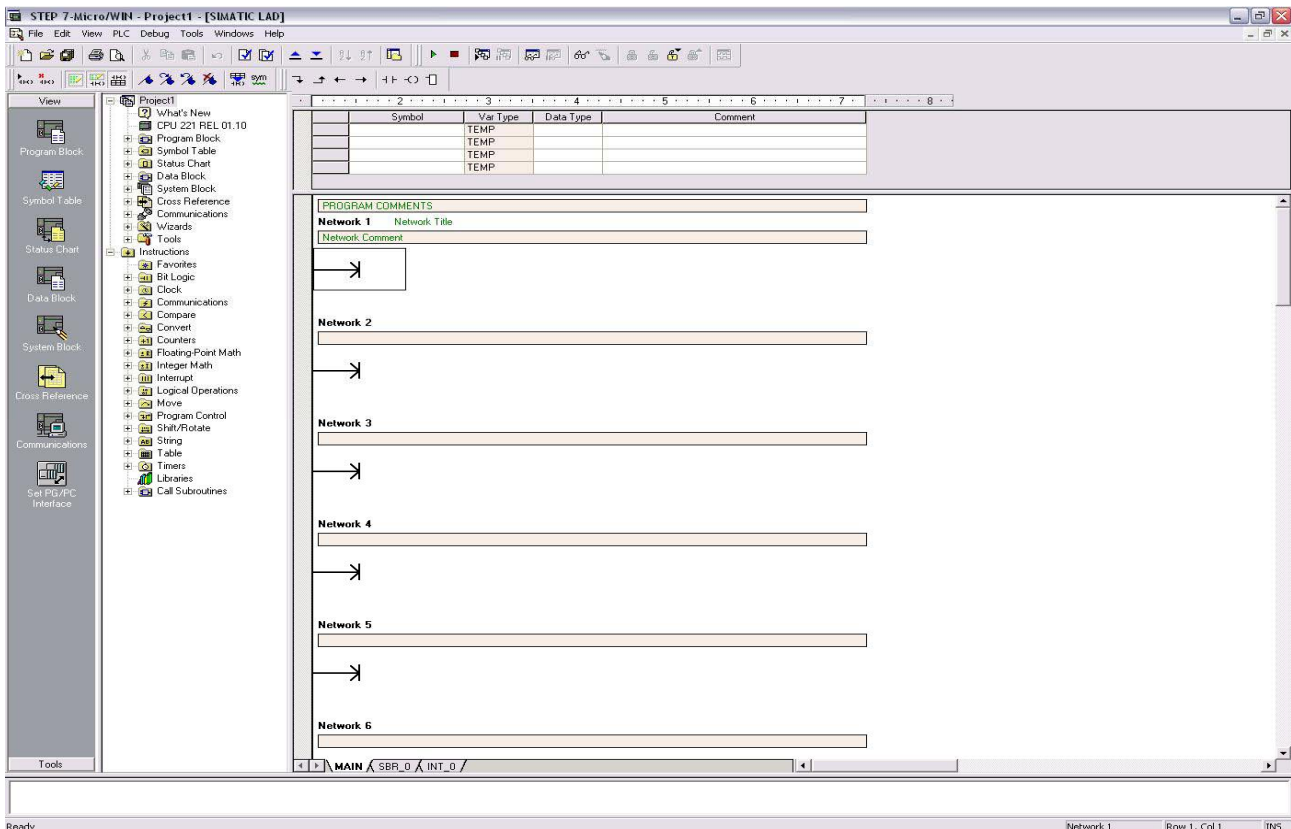
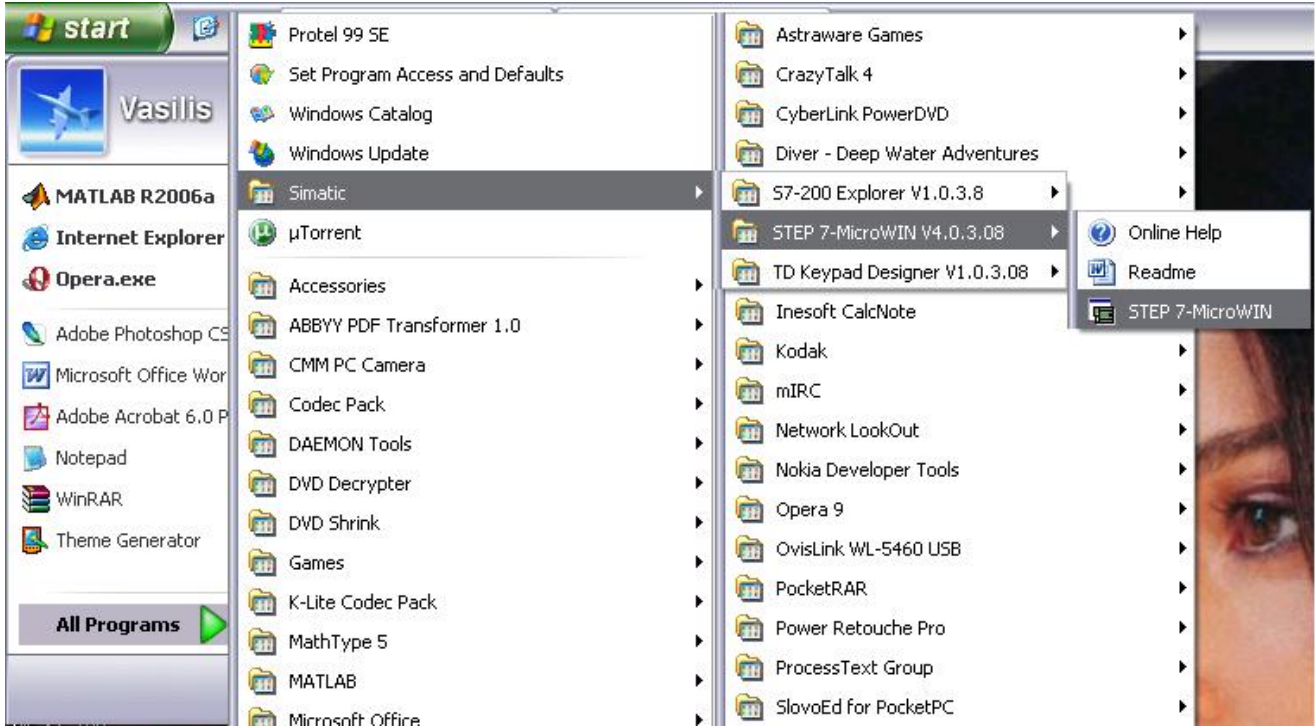
Η εντολή MOVE REAL μεταφέρει τον πραγματικό αριθμό της εισόδου στη έξοδο χωρίς να επηρεάζει το περιεχόμενο της εισόδου από τη μεταφορά στη έξοδο.

Η εντολή MOVE BYTE REAL συντάσσεται ως εξής:

MOVER IN,OUT

ΠΑΡΑΚΑΤΩ ΕΙΝΑΙ ΤΑ ΒΗΜΑΤΑ ΓΙΑ ΤΗ ΜΕΤΑΦΟΡΑ ΠΡΟΓΡΑΜΜΑΤΟΣ ΑΠΟ ΤΟ ΠΡΟΓΡΑΜΜΑ STEP-7 ΣΤΟ PLC

Ανοίγουμε το πρόγραμμα Step 7-MicroWin



Από το μενού View επιλεγούμε STL

The screenshot shows the SIMATIC Manager interface with the 'View' menu open. The 'STL' option is selected, and the 'View' menu is expanded to show various options. The main workspace displays the STL code editor for 'Network 1'.

View Menu Options:

- STL
- Ladder
- FBD
- Component
- Symbolic Addressing Ctrl+Y
- Symbol Table
- Symbol Information Table Ctrl+T
- POU Comments
- Network Comments
- Toolbars
- Frame
- Bookmarks
- Properties...

STL Code Editor Content:

Symbol	Var Type	Data Type	Comment
	TEMP		
	TEMP		
	TEMP		
	TEMP		

PROGRAM COMMENTS

Network 1 Network Title

Network Comment

Network 2

Network 3

Network 4

Network 5

Network 6

MAIN SBR_0 INT_0 /

Views STL Code Editor Network 1 Row 1, Col 1 INS

Τώρα μπορούμε να γράψουμε το κώδικα σε μορφή STL

The screenshot displays the SIMATIC Manager software interface for editing STL code. The main window is titled "STEP 7-MicroWIN - Project1 - [SIMATIC STL]". The interface includes a menu bar (File, Edit, View, PLC, Debug, Tools, Windows, Help), a toolbar, and a project tree on the left. The project tree shows a hierarchy starting with "Project1", followed by "What's New", "CPU 221 REL 01.10", and various blocks like "Program Block", "Symbol Table", "Status Chart", "Data Block", "System Block", "Cross Reference", "Communications", "Wizards", "Tools", "Instructions", "Favorites", "BR Logic", "Clock", "Communications", "Compare", "Convert", "Counters", "Floating-Point Math", "Integer Math", "Interrupt", "Logical Operations", "Move", "Program Control", "Shift/Rotate", "String", "Table", "Timers", "Libraries", and "Call Subroutines".

The main editing area is divided into two sections. The top section is a table with the following columns: Symbol, Var Type, Data Type, and Comment. The table contains four rows, each with "TEMP" in the Var Type column. The bottom section is a large text area for writing STL code, currently showing "PROGRAM COMMENTS" and a list of "Network 1" through "Network 9", each with a "Network Title" and "Network Comment" field.

At the bottom of the window, the status bar shows "Ready" and "Network 1 Ln 1, Col 1".

Αφού γράψουμε τον κώδικα είμαστε έτοιμοι να τον μεταφέρουμε στο PLC

The screenshot displays the STEP 7-Micro/WIN software interface. The main window shows a project titled "Project1" with a variable table and a ladder logic network.

Symbol	Var Type	Data Type	Comment
	TEMP		
	TEMP		
	TEMP		
	TEMP		

The ladder logic network is titled "Network 1" and contains the following code:

```
Network 1 Network Title  
Network Comment  
ID I0.0  
A I0.0  
= Q0.0  
END
```

The status bar at the bottom indicates "Ready" and "Network: 1 Ln 4, Col 4".

Πατάμε το Download και μας εμφανίζει την παρακάτω εικόνα. Πατάμε download αφού βάλουμε το PLC στο STOP

The screenshot shows the STEP 7-Micro/WIN software interface. A 'Download' dialog box is open, displaying a warning message: "The project PLC type (CPU 221 REL 01.10) does not match the remote PLC type (CPU 224 REL 01.11). Click the Change Project button to set the project PLC type to match the remote PLC." The dialog box also includes a 'Remote Address' field set to '2' and a 'CPU 224 REL 01.11' label. Below the warning, there are buttons for 'Options', 'Change Project', 'Download', and 'Cancel'. The 'Options' section is checked, showing 'Program Block', 'Data Block', and 'System Block' to be downloaded to the PLC. The background shows a ladder logic network with various components like timers and switches.

Symbol	Var Type	Data Type	Comment
	TEMP		
	TEMP		
	TEMP		
	TEMP		

PROGRAM COMMENTS

Network 1 Network Title

Network 1 Download

PPI Connection
Use the Options button to select blocks to download.

Remote Address: 2 CPU 224 REL 01.11

Warning: The project PLC type (CPU 221 REL 01.10) does not match the remote PLC type (CPU 224 REL 01.11). Click the Change Project button to set the project PLC type to match the remote PLC.

Options:

- Program Block To: PLC
- Data Block To: PLC
- System Block To: PLC
- Recipes
- Data Log Configurations

Close dialog on success
 Prompt on RUN to STOP

Network 4

```

ID  I0.3
A   I0.3
R   Q0.3, 1
R   Q0.5, 1
S   Q0.2, 1
S   Q1.0, 1
TON T35, +1500
LD  T35
S   Q0.4, 1
    
```

Compiling System Block...
Compiled Block with 0 errors, 0 warnings

Ready Network 2 Ln 3, Col 17 INS

ΠΑΡΑΔΕΙΓΜΑΤΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

Παράδειγμα 1

Έστω ότι έχουμε δυο διακόπτες σε σειρά S1 και S2 συνδεδεμένους σε σειρά και θέλουμε να ανάβει μια λάμπα L1 όταν κλείσει ένας από τους δυο διακόπτες.

Λύση:

```
LD I0.0
LD I0.1
O I0.0
O I0.1
= Q0.0
END
```

Παράδειγμα 2

Πατώντας ένα μπουτόν S1 δεν πρέπει να παρατηρείται καμιά μεταβολή στη έξοδο K1. Με το άνοιγμα θα πρέπει μετά από 5sec να διεγείρεται η έξοδος K1 και να παραμένει ανοικτή για 10sec.

Λύση:

```
LD I0.0
A I0.0
TOF T37, +50
LD T37
NOT
S Q0.0 , 1
LD Q0.0
A Q0.0
TON T38, +100
LD T38
R Q0.0 , 1
END
```

Παράδειγμα 3

Να φτιαχτεί ένας Counter που να μετρά προς τα πάνω και όταν φτάσει την τιμή 6 να κλείνει μια έξοδο. (Η έξοδος πρέπει να είναι ανοικτή με το ξεκίνημα του προγράμματος).

Λύση:

```
LD I0.0
LD I0.1
S Q0.0 , 1
CTU C1, +6
LD C1
R Q0.0 , 1
END
```

Παράδειγμα 4

Να φτιαχτεί ένας Counter που να μετρά προς τα κάτω και όταν μηδενίζεται να κλείνει μια έξοδο. (Η έξοδος πρέπει να είναι ανοικτή με το ξεκίνημα του προγράμματος).

Λύση:

```
LD I0.0
LD I0.1
S Q0.0 , 1
CTD C1, +5
LD C1
R Q0.0 , 1
END
```

Παράδειγμα 5

Σε μια διαδικασία ελέγχου θέλουμε να γίνονται τα εξής: όταν περάσουν 5 αυτοκίνητα από την είσοδο του πάρκινγκ να ανάβει μια λάμπα η οποία να ανάβει για 3sec. Η διαδικασία να επαναλαμβάνεται συνέχεια.

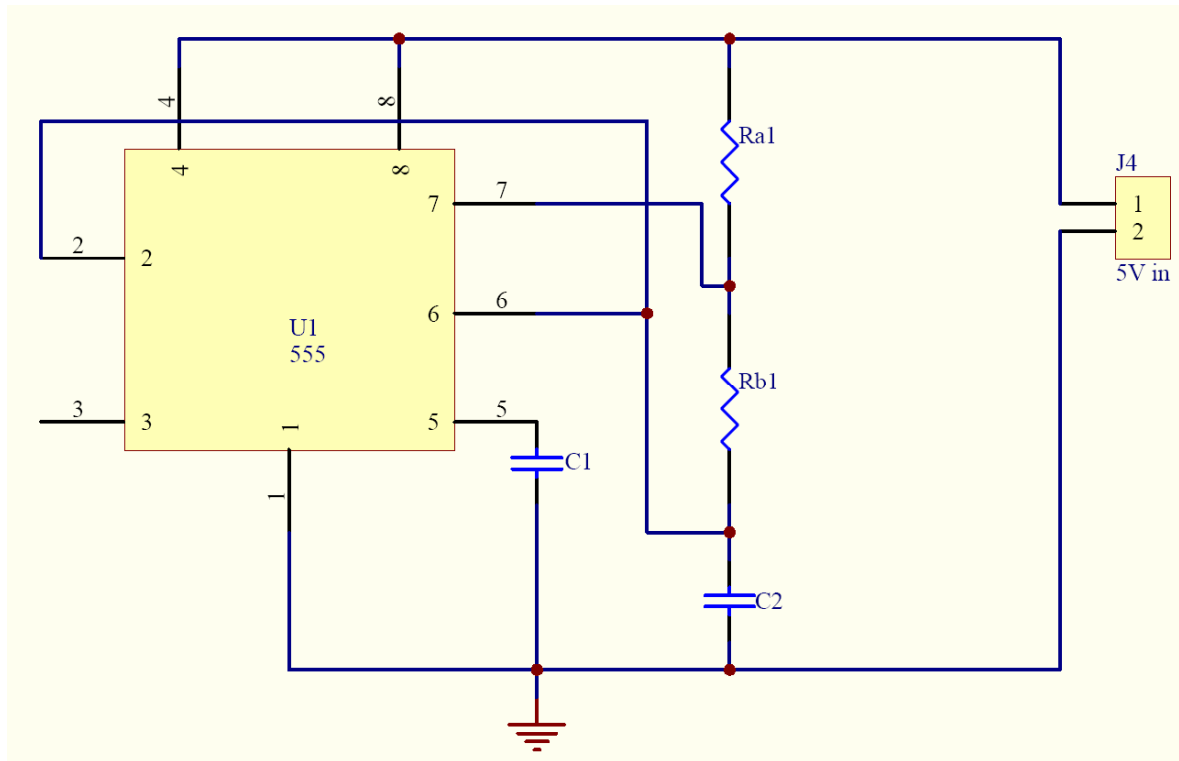
Λύση:

```
LD I0.0
LD I0.1
CTU C34, +5
LD C34
S Q0.0 , 1
TON T37, +30
LD T37
R Q0.0 , 1
END
```


ΕΛΕΓΧΟΣ ΜΠΑΡΩΝ

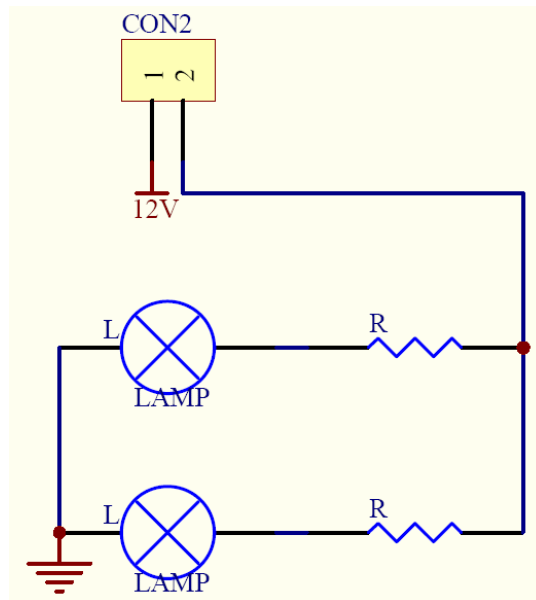
Για να ελέγχουμε τις μπάρες χρησιμοποιήσαμε Σέρβο-κινητήρες και το απαραίτητο κύκλωμα οδήγησης τους.

Το κύκλωμα οδήγησης τους αποτελείται από ένα LM555 και με τη ρύθμιση των σωστών αντιστάσεων ελέγχουμε τους παλμούς που δίνουμε στο κινητήρα για να δουλέψει. Παρακάτω είναι το κύκλωμα οδήγησης του Σέρβο-κινητήρα.



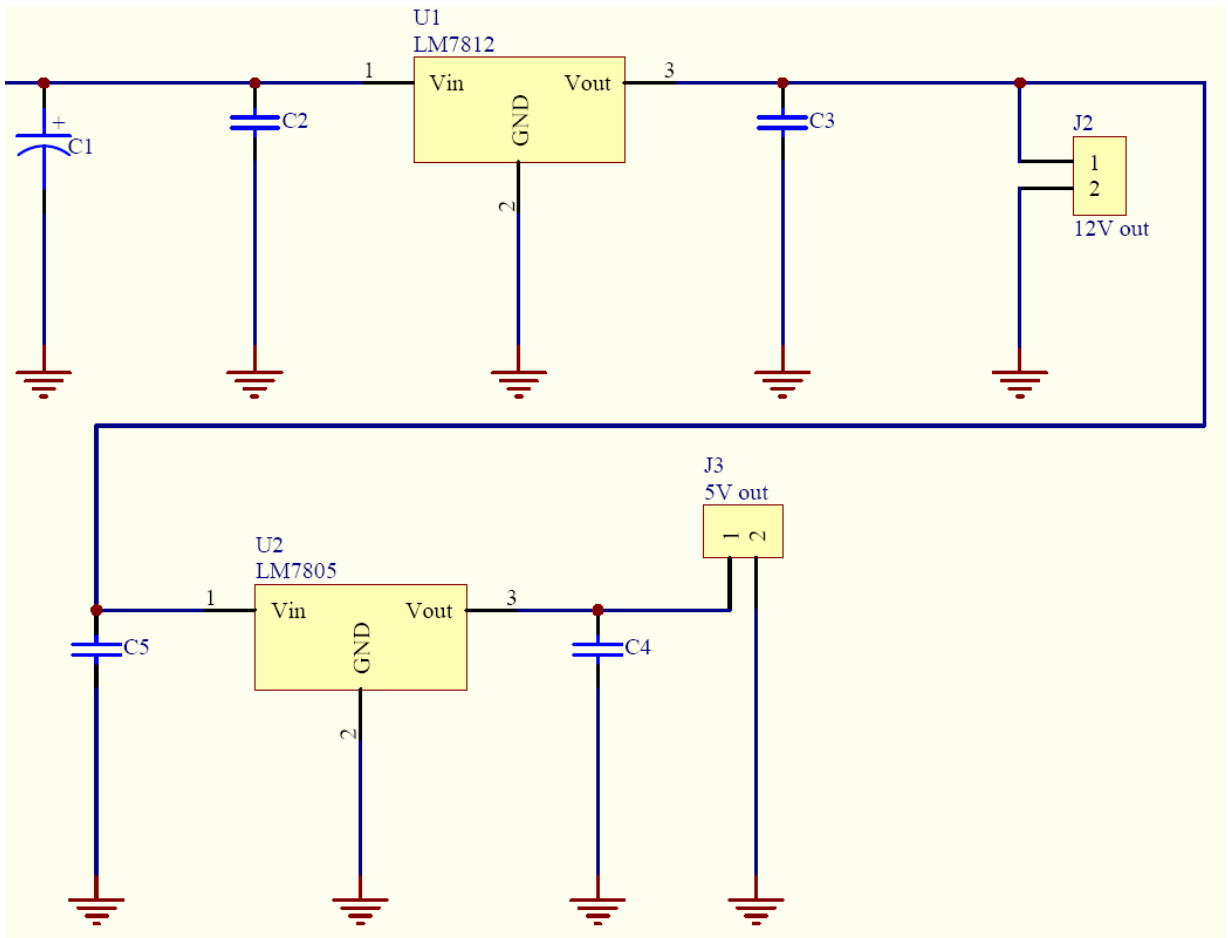
LED

Τα λαμπάκια (LED) και το BUZZER τα τροφοδοτήσαμε από μια πηγή DC 12V. Τα LED τα οδηγήσαμε με μια αντίσταση του 1KΩ.



ΚΥΚΛΩΜΑ ΤΡΟΦΟΔΟΣΙΑΣ

Είναι ένα τροφοδοτικό τάσης εξόδου 12V και 5V. Τα 12V τα χρησιμοποιήσαμε για να τροφοδοτήσουμε τα λαμπάκια και το Buzzer. Τα 5V τα χρησιμοποιήσαμε για να τροφοδοτήσουμε το κύκλωμα οδήγησης των Σέρβο-κινητήρων και για να τροφοδοτήσουμε και τους Σέρβο-κινητήρες. Την σταθεροποίηση την κάναμε με τα ολοκληρωμένα LM7812 και LM7805.



ΜΑΚΕΤΑ ΕΡΓΑΣΙΑΣ

Το κουτί της μακέτας αποτελείται από:

- 1) 2 κομμάτια ξύλου κόντρα-πλακέ διαστάσεων 80cm x 60cm.
- 2) 2 κομμάτια ξύλου μελαμίνης διαστάσεων 10cm x 60cm.
- 3) 2 κομμάτια ξύλου μελαμίνης διαστάσεων 10cm x 80cm.
- 4) 2 κομμάτια ξύλου μελαμίνης διαστάσεων 5cm x 25cm .
- 5) 10 LED κόκκινου χρώματος.
- 6) 4 LED πορτοκαλί χρώματος.
- 7) 2 LED πράσινου χρώματος.
- 8) 6 μαγνητικοί διακόπτες.
- 9) 12 m καλώδιο μαύρου χρώματος.
- 10) 12 m καλώδιο κόκκινου χρώματος.
- 11) 10 m καλώδιο κίτρινου χρώματος.
- 12) 10 m καλώδιο μπλέ χρώματος.
- 13) 10 m καλώδιο πράσινου χρώματος.
- 14) 10 m καλώδιο καφέ χρώματος.
- 15) 1 διακόπτης button.
- 16) 2 διακόπτες δύο θέσεων.

Για την ένωση και την σταθεροποίηση των κομματιών μεταξύ τους χρησιμοποιήθηκαν γωνίες τεσσάρων βιδών για καλύτερο αποτέλεσμα και επίσης το κάτω μέρος σταθεροποιήθηκε με την βοήθεια καρφιών.

Το πάνω μέρος της μακέτας αποτελείται από:

Στο πάνω μέρος χρησιμοποιήσαμε το δεύτερο κομμάτι του κόντρα-πλακέ και το καλύψαμε με τσόχα στις διαστάσεις του ξύλου 80cm x 60cm. Οι σιδηροδρομικές γραμμές όπως και τα φαναράκια αποτελούνται από ξύλο Balsa που είναι το πλέον κατάλληλο για μακέτες και γενικά σε ότι έχει να κάνει με μοντελισμό. Είναι το πιο εύχριστο και το πιο φθινό υλικό της αγοράς. Η ασφάλτος αποτελείται από δύο κομμάτια ξύλου μελαμίνης 5cm x 25cm. Το υλικό που χρησιμοποιήθηκε είναι μπουτόχαρτο το οποίο ουσιαστικά είναι ένα ειδικό υαλόχαρτο. Ανάμεσα στις ράγες του τρένου υπάρχει ένα κομμάτι ξύλου με διαστάσεις 2cm x 6cm το οποίο λειτουργεί ως διακόπτης καθώς υπάρχει στο κάτω μέρος υπάρχει ένας διακόπτης ο οποίος στην κανονική του κατάσταση off χάρη στο ελατήριο το οποίο υπάρχει ανάμεσα στο ξύλο και σε ένα δεύτερο κομμάτι που βρίσκεται από κάτω. Όταν το ελατήριο πιέζεται τότε ο διακόπτης έρχεται στην θέση on. Οι μπάρες έχουν τοποθετηθεί πάνω σε δύο σερβοκινητήρες. Επίσης πάνω στην μακέτα υπάρχουν μαγνητικοί διακόπτες οι οποίοι όταν περάσει το τρένο από μπροστά τους ο μαγνήτης που υπάρχει στο δεξί και το αριστερό μέρος τους ενεργοποιεί και έρχονται σε κατάσταση on και έτσι εκτελείται η ανάλογη λειτουργία. Στην μακέτα επίσης έχουμε

τοποθετήσει δύο διακόπτες ON-OFF ο ένας για την έναρξη της λειτουργίας και ο δεύτερος για την αρχικοποίηση της λειτουργίας του P.L.C.

Τροφοδοσία μακέτας :

Για την τροφοδοσία των πλακετών που βρίσκονται στο κάτω μέρος της μακέτας κατασκευάσαμε ένα τροφοδοτικό με εξόδους στα 12V και στα 5V.

ΚΩΔΙΚΑΣ του PLC

LD	I0.0	%Με αυτό το κομμάτι προγραμματίσαμε
A	I0.0	%τις τέσσερις πορτοκαλί LED να
AN	T38	% αναβοσβήνουν.
TON	T37, +20	
LD	T37	
=	Q0.1	
A	T37	
TON	T38, +20	
<hr/>		
LD	I0.4	%Με αυτό το κομμάτι προγραμματίσαμε
A	I0.4	%το PLC να ανάβει κόκκινο
R	Q0.3, 1	% όταν έρχεται τρένο από δεξιά.
S	Q0.2, 1	
S	Q0.6, 1	
<hr/>		
LD	I0.3	%Με αυτό το κομμάτι προγραμματίσαμε
A	I0.3	%το PLC να ανάβει κόκκινο
R	Q0.3, 1	%όταν έρχεται τρένο από αριστερά.
S	Q0.2, 1	
S	Q1.0, 1	
<hr/>		
LD	I0.1	%Με αυτό το κομμάτι προγραμματίσαμε
A	I0.1	%το PLC να ανάβει πράσινο όταν
R	Q0.2, 1	%περάσει το τρένο από δεξιά.
R	Q0.6, 1	
S	Q0.3, 1	
<hr/>		
LD	I0.2	%Με αυτό το κομμάτι προγραμματίσαμε
A	I0.2	%το PLC να ανάβει πράσινο όταν
R	Q0.2, 1	%περάσει το τρένο από αριστερά.
R	Q1.0, 1	
S	Q0.3, 1	
<hr/>		
LD	I0.5	%Γίνεται έλεγχος αν κόλλησε κάποιο
A	I0.5	%αμάξι πάνω στη διαβάσει να
TON	T100, +1000	%ειδοποιήσει τα τρένα να σταματήσουν
LD	T100	%ειδοποιεί και την υπεύθυνη
S	Q0.0, 1	%υπηρεσία για το πρόβλημα.
S	Q1.1, 1	

LD	I0.6	%Γίνεται έλεγχος αν έρχονται τρένα και
LD	I0.7	%από τις δυο πλευρές και τα ειδοποιεί
A	I0.6	%να σταματήσουν.Ειδοποιεί
A	I0.7	%επίσης και την υπεύθυνη
S	Q0.0, 1	%υπηρεσία για το πρόβλημα.
S	Q1.1, 1	

LD	I1.2	%Γίνεται RESET στις λάμπες που
A	I1.2	%ειδοποιούν τα τρένα να
R	Q0.0, 1	% σταματήσουν και τον δεκτή της
R	Q1.1, 1	% υπηρεσίας ότι το πρόβλημα
		% διορθώθηκε.

LD	I0.0	
A	I0.0	
TOF	T37, +30	%Ιανουάριος
LD	T37	
AN	T61	
TOF	T62, +12	%μετρητής για 9 ώρες
LD	T62	
TOF	T60, +3600	% μετρητής για 1 ώρα
LD	T60	
=	Q0.0	
TOF	T63, +10	% μετρητής για 13 ώρες
LD	T63	
TOF	T61, +3600	% μετρητής για 1 ώρα
LD	T61	

TOF	T37, +30	%Φεβρουάριος
LD	T37	
AN	T61	
TOF	T62, +12	%μετρητής για 9 ώρες
LD	T62	
TOF	T60, +3600	% μετρητής για 1 ώρα
LD	T60	
=	Q0.0	
TOF	T63, +10	% μετρητής για 13 ώρες
LD	T63	
TOF	T61, +3600	% μετρητής για 1 ώρα
LD	T61	

TOF T37, +30 %Μάρτης
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Απρίλης
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Μάης
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Ιούνιος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Ιούλιος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Αύγουστος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Σεπτέμβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Οκτώβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Νοέμβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30	%Δεκέμβριος
LD T37	
AN T61	
TOF T62, +12	%μετρητής για 9 ώρες
LD T62	
TOF T60, +3600	% μετρητής για 1 ώρα
LD T60	
= Q0.0	
TOF T63, +10	% μετρητής για 13 ώρες
LD T63	
TOF T61, +3600	% μετρητής για 1 ώρα
LD T61	

LD I0.0	
A I0.0	
TOF T37, +30	%Ιανουάριος
LD T37	
AN T61	
TOF T62, +12	%μετρητής για 9 ώρες
LD T62	
TOF T60, +3600	% μετρητής για 1 ώρα
LD T60	
= Q0.0	
TOF T63, +10	% μετρητής για 13 ώρες
LD T63	
TOF T61, +3600	% μετρητής για 1 ώρα
LD T61	

TOF T37, +30	%Φεβρουάριος
LD T37	
AN T61	
TOF T62, +12	%μετρητής για 9 ώρες
LD T62	
TOF T60, +3600	% μετρητής για 1 ώρα
LD T60	
= Q0.0	
TOF T63, +10	% μετρητής για 13 ώρες
LD T63	
TOF T61, +3600	% μετρητής για 1 ώρα
LD T61	

TOF T37, +30 %Μάρτης
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Απρίλης
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Μάης
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Ιούνιος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Ιούλιος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Αύγουστος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Σεπτέμβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Οκτώβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Νοέμβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Δεκέμβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

LD I0.0
A I0.0
TOF T37, +30 %Ιανουάριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Φεβρουάριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Μάρτης
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Απρίλης
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Μάης
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Ιούνιος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Ιούλιος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Αύγουστος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Σεπτέμβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Οκτώβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Νοέμβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Δεκέμβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

LD I0.0
A I0.0
TOF T37, +30 %Ιανουάριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +28 %Φεβρουάριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Μάρτης
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Απρίλης
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Μάης
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Ιούνιος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Ιούλιος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Αύγουστος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Σεπτέμβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Οκτώβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30 %Νοέμβριος
LD T37
AN T61
TOF T62, +12 %μετρητής για 9 ώρες
LD T62
TOF T60, +3600 % μετρητής για 1 ώρα
LD T60
= Q0.0
TOF T63, +10 % μετρητής για 13 ώρες
LD T63
TOF T61, +3600 % μετρητής για 1 ώρα
LD T61

TOF T37, +30	%Δεκέμβριος
LD T37	
AN T61	
TOF T62, +12	%μετρητής για 9 ώρες
LD T62	
TOF T60, +3600	% μετρητής για 1 ώρα
LD T60	
= Q0.0	
TOF T63, +10	% μετρητής για 13 ώρες
LD T63	
TOF T61, +3600	% μετρητής για 1 ώρα
LD T61	

END

ΕΠΙΛΟΓΟΣ

Τα P.L.C είναι η πλέον ανερχόμενη δύναμη στον αυτοματισμό τα τελευταία 20 χρόνια και συγκρινόμενα με άλλα παρόμοια σε δυνατότητες συστήματα σύγχρονα όπως οι επεξεργαστές DS ο κώδικας σε μορφή STL είναι ο πλέον απλός και ο πλέον διαδεδομένος σε όσους ασχολούνται επαγγελματικά με αυτοματισμούς. Μπορεί άλλη μια πτυχιακή εργασία να έφτασε στο τέλος της όμως ο στόχος του θέματος αυτού δεν τελειώνει εδώ. Όπως εμείς οι ίδιοι βοηθηθήκαμε με τις αρκετά καινότεμες ιδέες από άλλους συναδέλφους και από διάφορα site και διάφορες εργασίες πολλών και αξιόλογων καθηγητών από όλη την Ελλάδα όσον αφορά την δομή του P.L.C και άλλα κομμάτια μέχρι τον κώδικα έτσι και η προσωμοίωση του κώδικα στην μακέτα φέρνει αρκετά κοντά τους φοιτητές στις πλέον νέες τεχνολογίες ξεφεύγοντας από τον κλασσικό αυτοματισμό. Το να ασχολείται κάποιος με ένα θέμα και να εμβαθίνει σε αυτό τον αναγκάζει να ξεφύγει από την επιφανειακή ενασχόληση και να δει τα πράγματα όπως είναι πραγματικά και όχι όπως θέλουμε να νομίζουμε.

ΒΙΒΛΙΟΓΡΑΦΙΑ

Αυτοματισμός με χρήση PLC
Μπερέτας, Ιωάννης

1) McIntyre – Losee, Βιομηχανικοί αυτοματισμοί κινητήρων,
Εκδόσεις Τζιόλα, 2000

2) Collins – Lane, Προγραμματιζόμενοι ελεγκτές, Ένας πρακτικός
οδηγός, Εκδόσεις Τζιόλα, 2000

3) Petruzella, Προγραμματιζόμενοι λογικοί ελεγκτές (P.L.C.),
Εκδόσεις Τζιόλα, 2000

4) Κρανά, Βιομηχανικοί αυτοματισμοί & προγραμματιζόμενοι
λογικοί ελεγκτές (PLC), Εκδόσεις Ιων, 2000

5) Κοτζαμπάση Μ., Αυτοματισμοί (βιβλίο και CDROM), Εκδόσεις
Ιων, 2000

6) Πανταζή Ν., Προγραμματιζόμενοι λογικοί ελεγκτές, Εκδόσεις Ιων,
2000

Χρήσιμες διευθύνσεις στο internet

<http://www.metadosi-ischios.gr/article.php?ID=69>

<http://www.algosystems.gr/new/gr/default.php?id=5&mid=2&pid=1>
algosystems

<http://www.barn.org/FILES/historyofplc.html>

http://www.isa.org/Content/ContentGroups/News/2006/February24/History_of_the_PLC.htm

<http://www.plcs.net/chapters/history2.htm>