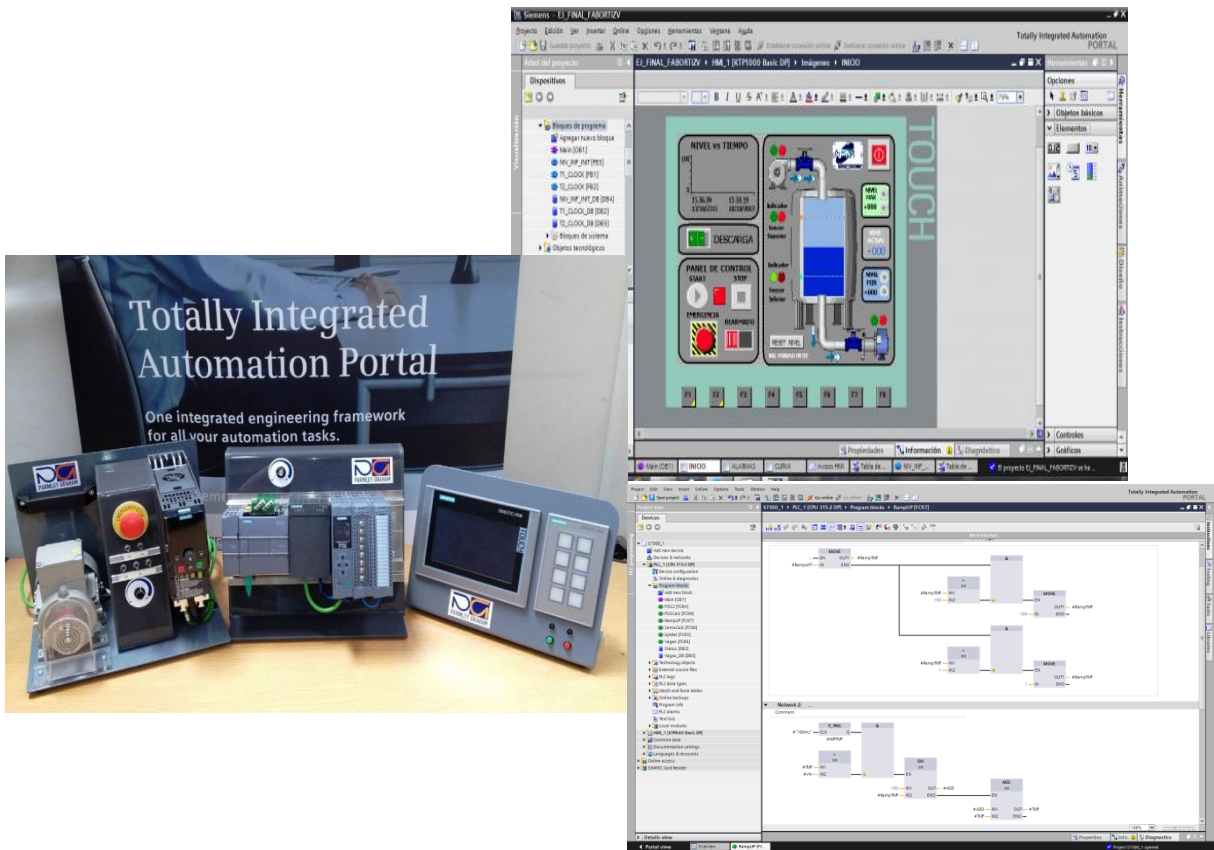


ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ 1429



**Προσομοίωση σταθμού παστερίωσης
γάλακτος που ελέγχεται από PLC μέσω του
προγράμματος TIA PORTAL.**

ΣΠΟΥΔΑΣΤΗΣ ΝΙΚΟΛΟΠΟΥΛΟΣ ΧΡΗΣΤΟΣ (5964)

ΕΠΟΠΤΕΥΟΝ ΚΑΘΗΓΗΤΗΣ:ΚΑΛΑΝΤΖΟΠΟΥΛΟΣ ΑΘΑΝΑΣΙΟΣ

Πρόλογος

Σκοπός της παρούσας πτυχιακής εργασίας είναι η γνωριμία και η εξοικείωση με το λογισμικό πακέτο TIA PORTAL της Siemens το οποίο παρέχει τα απαιτούμενα εργαλεία για την ανάπτυξη απλών ή και σύνθετων αυτοματισμών με PLCs (Programmable Logic Controllers – Προγραμματιζόμενοι Λογικοί Ελεγκτές). Υλοποιήθηκε και προσομοιώθηκε μια εφαρμογή για την λειτουργία ενός σταθμού παστερίωσης γάλακτος. Ο προγραμματισμός του PLC πραγματοποιήθηκε στην γλώσσα προγραμματισμού ladder ενώ για τον έλεγχο της εφαρμογής αναπτύχθηκε ένα HMI (Human Machine Interface – Διεπαφή Ανθρώπου Μηχανής) αξιοποιώντας τα εργαλεία STEP 7, PLC-SIM και WinCC που περιλαμβάνονται στο λογισμικό πακέτο TIA PORTAL της Siemens.

Αρχικά στο πρώτο κεφάλαιο της παρούσας πτυχιακής εργασίας, γίνεται μια εισαγωγή στους προγραμματιζόμενους λογικούς ελεγκτές και στα βασικά χαρακτηριστικά τους. Επίσης γίνεται αναφορά στις γλώσσες προγραμματισμού και στη δομή τους, στην αρχή λειτουργίας καθώς και στα HMI. Στο δεύτερο κεφάλαιο, παρουσιάζεται το λογισμικό πακέτο Tia Portal της Siemens, οι δυνατότητες του καθώς και η εξομοίωση εφαρμογής για καλύτερη κατανόηση. Στο τρίτο κεφάλαιο γίνεται εισαγωγή στους ελεγκτές συστημάτων αυτόματου ελέγχου. Επίσης παρουσιάζεται η χρήση και λειτουργία του ελεγκτή που χρησιμοποιήθηκε στη εφαρμογή. Στο τέταρτο κεφάλαιο περιγράφεται η υλοποίηση της εφαρμογής για την λειτουργία ενός σταθμού παστερίωσης γάλακτος. Σε αυτό το κεφάλαιό παρουσιάζεται επίσης ο σχεδιασμός της οθόνης του HMI μέσα από την οποία ο χρήστης έχει την δυνατότητα να ελέγχει την λειτουργία της παραπάνω εφαρμογής. Τέλος, στο πέμπτο κεφάλαιο παρουσιάζεται η λειτουργία της εφαρμογής από την μεριά του χρήστη καθώς και τα διάφορα σενάρια ασφαλείας που υποστηρίζονται.

Στο σημείο αυτό θα ήθελα να ευχαριστήσω το διδακτικό προσωπικό του Τμήματος Ηλεκτρολόγων Μηχανικών ΤΕ του ΤΕΙ Δυτικής Ελλάδας για τις γνώσεις και τις εμπειρίες που μου παρέχουν κατά την διάρκεια της φοίτησης μου. Επίσης θα ήθελα να ευχαριστήσω τον εισηγητή μου κ. Καλαντζόπουλο Αθανάσιο για την διαρκή υποστήριξη του για την υλοποίηση και ολοκλήρωση της παρούσας πτυχιακής εργασίας. Επίσης τους συμφοιτητές μου για τα όμορφα φοιτητικά χρόνια αλλά και για την δημιουργία της εργασίας. Τέλος θα ήθελα να ευχαριστήσω την οικογένεια μου για την διαρκή ψυχολογική και υλική υποστήριξη που μου παρέχουν όλη την διάρκεια των σπουδών μου.

Περίληψη

Σκοπός της πτυχιακής εργασίας είναι η παρουσίαση των βασικών δυνατοτήτων των επιμέρους προγραμμάτων που περιλαμβάνει το λογισμικό πακέτο TIA PORTAL V12 της Siemens. Μέσω των παραπάνω προγραμμάτων, ο χρήστης έχει την δυνατότητα να γνωρίσει τα PLCs (Programmable Logic Controllers – Προγραμματιζόμενους Λογικούς Ελεγκτές) και να εξοικειωθεί με τον προγραμματισμό τους. Επίσης, έχει την δυνατότητα να εξετάσει την ορθή λειτουργία του προγράμματος που ανέπτυξε αξιοποιώντας την εφαρμογή PLCSim η οποία προσομοιώνει την λειτουργία ενός πραγματικού PLC. Επιπρόσθετα, δίνεται η δυνατότητα δημιουργίας προγραμμάτων γραφικής απεικόνισης που αποκαλούνται HMI (Human Machine Interface – Διεπαφή Ανθρώπου Μηχανής) για την εποπτεία και τον έλεγχο μιας εφαρμογής που εκτελείται από το PLC.

Με σκοπό να παρουσιαστεί ένα τμήμα των δυνατοτήτων που προσφέρει το λογισμικό πακέτο TIA PORTAL αρχικά θα περιγραφεί η ανάπτυξη του προγράμματος ενός PLC που ελέγχει ένα σταθμό παστερίωσης γάλακτος. Ο έλεγχος της συγκεκριμένης εφαρμογής θα γίνεται αποκλειστικά μέσω ενός HMI που θα αναπτυχθεί με το WinCC της Siemens. Ο έλεγχος της λειτουργίας τόσο του προγράμματος του PLC όσο και του HMI θα πραγματοποιηθεί αξιοποιώντας τα εργαλεία προσομοίωσης που διαθέτει το TIA Portal.

Στην παραπάνω εφαρμογή το πρόγραμμα του PLC αναλαμβάνει την λειτουργία μιας μεταφορικής ταινίας στην οποία τοποθετούνται τα μπουκάλια γάλακτος και τον έλεγχο του φούρνου παστερίωσης. Ένα από τα σημαντικότερα προβλήματα που πρέπει να αντιμετωπιστούν στην συγκεκριμένη εφαρμογή είναι ο έλεγχος της θερμοκρασίας του φούρνου παστερίωσης μέσω ενός PID ελεγκτή. Επίσης στο πρόγραμμα που θα υλοποιηθεί θα πρέπει να γίνεται έλεγχος για τυχόν σφάλματα που θα προκύπτουν κατά την λειτουργία της εφαρμογής.

Περιεχόμενα

| | |
|--|----------|
| Πρόλογος | iii |
| Περίληψη | v |
| Περιεχόμενα..... | vii |
| Κεφάλαιο 1. Εισαγωγή στους λογικούς προγραμματιζόμενους ελεγκτές..... | 1 |
| 1.1 Ιστορική Αναδρομή | 1 |
| 1.2 Πλεονεκτήματα των PLCs | 2 |
| 1.3 Εφαρμογές των PLCs | 3 |
| 1.4 Ανάλυση των τμημάτων του PLC | 4 |
| 1.4.1 Δομή του PLC | 4 |
| 1.4.2 Πλαίσιο τοποθέτησης μονάδων | 5 |
| 1.4.3 Μονάδα τροφοδοσίας..... | 5 |
| 1.4.4 Κεντρική μονάδα επεξεργασίας | 6 |
| 1.4.5 Μονάδες εισόδων-εξόδων | 6 |
| 1.4.6 Κάρτες ψηφιακών εισόδων..... | 7 |
| 1.4.7 Κάρτες ψηφιακών εξόδων | 7 |
| 1.4.8 Κάρτες αναλογικών εισόδων | 7 |
| 1.4.9 Κάρτες αναλογικών εξόδων..... | 8 |
| 1.5 Εσωτερική δομή του PLC..... | 8 |
| 1.5.1 Μνήμη RAM | 9 |
| 1.5.2 Μνήμη EEPROM..... | 9 |
| 1.5.3 SD κάρτα | 9 |
| 1.5.4 Μνήμη ROM..... | 10 |
| 1.6 Γλώσσες προγραμματισμού..... | 10 |
| 1.6.1 Ladder..... | 10 |
| 1.6.2 Function Block Diagram | 11 |
| 1.6.3 Statement List | 11 |
| 1.7 Δομή προγράμματος..... | 12 |
| 1.7.1 Μπλοκ χρήση | 12 |
| 1.7.2 Μπλοκ συστήματος..... | 13 |
| 1.7.3 Στάνταρτ μπλοκ | 13 |
| 1.7.4 Δομή μπλοκ | 13 |
| 1.8 Μορφές προγραμματισμού | 14 |
| 1.9 Αρχή λειτουργίας ενός PLC..... | 15 |
| 1.10 Τρόποι επικοινωνίας των PLCs | 16 |
| 1.10.1 Profibus..... | 17 |
| 1.10.2 Modbus RTU..... | 17 |
| 1.10.3 Profinet | 17 |
| 1.10.4 Industrial Ethernet..... | 18 |
| 1.10.5 AS-Interface | 19 |
| 1.11 Τηλεχειρισμός και εποπτικός έλεγχος | 19 |
| 1.12 Κατηγορίες HMI και SCADA..... | 20 |
| 1.12.1 Push Button Panels..... | 20 |
| 1.12.2 Operator interfaces | 20 |
| 1.12.3 Text displays | 21 |

| | |
|---|-----------|
| 1.12.4 Operator Panels..... | 21 |
| 1.12.5 Touch Panels..... | 22 |
| Κεφάλαιο 2. Το λογισμικό πακέτο Tia Portal..... | 23 |
| 2.1 Εισαγωγή..... | 23 |
| 2.2 Δυνατότητες του Tia Portal..... | 24 |
| 2.3 Εισαγωγή στο πρόγραμμα Tia Portal..... | 25 |
| 2.4 Ανάπτυξη του προγράμματος του PLC..... | 28 |
| 2.5 Ανάπτυξη του HMI..... | 31 |
| Κεφάλαιο 3. Εισαγωγή στον ελεγκτή PID..... | 35 |
| 3.1 Ελεγκτές συστημάτων αυτόματου ελέγχου..... | 35 |
| 3.2 Αναλογικός (P) ελεγκτής..... | 35 |
| 3.3 Ολοκληρωτικός (I) ελεγκτής..... | 36 |
| 3.4 Διαφορικός (D) ελεγκτής..... | 37 |
| 3.5 Αναλογικός-ολοκληρωτικός (PI) ελεγκτής..... | 38 |
| 3.6 Αναλογικός-διαφορικός (PD) ελεγκτής..... | 39 |
| 3.7 Αναλογικός-ολοκληρωτικός-διαφορικός(PID) ελεγκτής..... | 40 |
| 3.8 Ελεγκτής PID Continuous Controller(CONT_C)..... | 41 |
| Κεφάλαιο 4. Περιγραφή του σταθμού παστερίωσης γάλακτος..... | 47 |
| 4.1 Περιγραφή σεναρίων..... | 47 |
| 4.1.1 Κανονική λειτουργία χωρίς σφάλματα..... | 47 |
| 4.1.2 Λειτουργία με σφάλμα στη ρύθμιση θερμοκρασίας εντός ορίων..... | 47 |
| 4.1.3 Λειτουργία με σφάλμα στη θερμοκρασία παστερίωσης κάτω από ένα όριο..... | 47 |
| 4.1.4 Λειτουργία με σφάλμα στη μεταφορική ταινία..... | 48 |
| 4.2 Υλοποίηση προγράμματος..... | 48 |
| 4.3 Υλοποίηση του HMI..... | 62 |
| 4.3.1 Ορισμός μπουτόν με ενδεικτική λυχνία..... | 63 |
| 4.3.2 Επιλογή μπουτόν..... | 65 |
| 4.3.3 Επιλογή φωτεινού σηματοδότη..... | 66 |
| 4.3.4 Παρουσίαση αριθμών..... | 67 |
| 4.3.5 Παρουσίαση πλαισίου σφαλμάτων..... | 68 |
| 4.3.6 Κίνηση μπουκαλιού..... | 69 |
| Κεφάλαιο 5. Αποτελέσματα - συμπεράσματα..... | 73 |
| 5.1 Λειτουργία της διαδικασίας του σταθμού παστερίωσης γάλακτος..... | 73 |
| 5.1.1 Σενάριο 1- Λειτουργία χωρίς σφάλματα..... | 73 |
| 5.1.2 Σενάριο 2-Σφάλμα 1, Θερμοκρασία εκτός ορίων..... | 76 |
| 5.1.3 Σενάριο 3-Σφάλμα 2, Πτώση της θερμοκρασίας κάτω από ένα όριο..... | 79 |
| 5.1.4 Σενάριο 4-Σφάλμα 3, Σφάλμα στην μεταφορική ταινία..... | 80 |
| 5.2 Συμπεράσματα – προτάσεις..... | 81 |
| Βιβλιογραφία..... | 83 |

Κεφάλαιο 1

Εισαγωγή στους λογικούς προγραμματιζόμενους ελεγκτές

1.1 Ιστορική αναδρομή

Η εξέλιξη των αυτοματισμών, όπως είναι φυσικό, ακολούθησε την πορεία της εξέλιξης της τεχνολογίας. Οι πρώτοι αυτοματισμοί ήταν καθαρά μηχανικοί, δηλαδή όλοι οι έλεγχοι καθοριζόταν από την κίνηση γραναζιών και μοχλών. Το μεγάλο άλμα στους αυτοματισμούς έγινε με τη χρήση του ηλεκτρισμού όπου το κύριο εξάρτημα ηλεκτρολογικών αυτοματισμών είναι ο ηλεκτρονόμος (relay).

Μετά τον δεύτερο παγκόσμιο πόλεμο αρχίζει η ηλεκτρονική εποχή. Ήδη από τις αρχές του 20ου αιώνα έχουμε τις πρώτες ηλεκτρονικές συσκευές, το ραδιόφωνο και, αργότερα, την τηλεόραση, τους ασυρμάτους και τα ραντάρ. Η ανακάλυψη του τρανζίστορ το 1950 ήταν η αρχή της ηλεκτρονικής επανάστασης των ημιαγωγών. Το θαυματουργό αυτό στοιχείο αντικατέστησε την ακριβή, ογκώδη και ενεργοβόρα ηλεκτρονική λυχνία και έκανε τις ηλεκτρονικές συσκευές μικρότερες, εύκολες στην κατασκευή και μείωσε το κόστος τους.

Το 1945 κατασκευάστηκε ο πρώτος ηλεκτρονικός υπολογιστής, ο ENIAC, ο οποίος χρησιμοποιούσε λυχνίες. Ο ENIAC δεν θύμιζε σε τίποτα τους σημερινούς υπολογιστές, ήταν ένα ολόκληρο εργοστάσιο το οποίο έλυνε μαθηματικές εξισώσεις. Μετά το 1950 και με την χρήση των τρανζίστορ έχουμε τους πρώτους πραγματικούς υπολογιστές, οι οποίοι χρησιμοποιούνται κυρίως στο θέμα της μηχανογράφησης, δηλαδή στην αποθήκευση και διαχείριση μεγάλων αρχείων δεδομένων.

Από την δεκαετία του '60 ήδη οι μηχανικοί άρχισαν να σκέφτονται τρόπους για να αξιοποιήσουν τις δυνατότητες των υπολογιστών στη βιομηχανία. Από τις πρώτες εφαρμογές των υπολογιστών στην βιομηχανία ήταν οι αυτόματες εργαλειομηχανές (τόρνοι, φρέζες κ.λπ), οι οποίες μέχρι τότε χρησιμοποιούσαν κυρίως μηχανολογικούς και λιγότερο ηλεκτρολογικούς αυτοματισμούς. Η επιτυχημένη αυτή εφαρμογή οδήγησε τους μηχανικούς να αρχίσουν να σκέφτονται την αντικατάσταση όλων των αυτοματισμών ενός εργοστασίου από ένα υπολογιστή. Μέχρι όμως την δεκαετία του '80 αυτό ήταν αδύνατο, διότι ο υπολογιστής ήταν μια πανάκριβη και δύσκολη στην χρήση της συσκευή.

Η επανάσταση της πληροφορικής ξεκινά το 1975 με την κατασκευή του πρώτου μικροϋπολογιστή. Πολλά από όλα όσα σήμερα θεωρούμε αυτονόητα δημιουργήθηκαν μετά το 1980. Η τεχνολογία άλλαξε πορεία, αλλάζοντας πορεία σε όλους τους τομείς της καθημερινής ζωής. Ο μικροϋπολογιστής αναπτύχθηκε παντού, σε οποιαδήποτε τομέα, σε οποιαδήποτε εφαρμογή.

Η βιομηχανία μέχρι και τη δεκαετία του '80 μπορούμε να πούμε ότι χρησιμοποιούσε ελάχιστα ηλεκτρονικά. Το 90% και πλέον των αυτοματισμών καταλάμβαναν οι αυτοματισμοί με ηλεκτρονόμους. Τα ηλεκτρονικά χρησιμοποιούνταν κυρίως για κάποιες "ευφυείς" εφαρμογές και οι πλακέτες αυτές τοποθετούνταν μέσα στους πίνακες των ηλεκτρονόμων.

Στις αρχές της δεκαετίας του '80 οι εταιρίες παραγωγής ηλεκτρολογικού υλικού εμφανίζουν στους μηχανικούς και τεχνικούς της βιομηχανίας ένα νέο προϊόν αυτοματισμού, το οποίο ονόμασαν PLC (Programmable Logic Controller - Προγραμματιζόμενος Λογικός Ελεγκτής).

Το PLC δεν είναι τίποτα άλλο παρά ένας μικροϋπολογιστής, κατάλληλα προσαρμοσμένος ώστε να χρησιμοποιείται για τη λειτουργία αυτοματισμών. Η χρήση του προορίζονταν για αντικαταστήσουν τον κλασικό πίνακα αυτοματισμού με τους ηλεκτρονόμους. Όπως γίνεται εύκολα κατανοητό μιλάμε για μια τεράστια αλλαγή στον τρόπο που μέχρι τότε δούλευε η βιομηχανία, δηλαδή έπρεπε να περάσει κατευθείαν από τους

ηλεκτρονόμους στους υπολογιστές. Προσάρμοσαν τον τρόπο χρήσης του PLC στον τρόπο που δούλευε τότε η βιομηχανία, δηλαδή, έντεχνα απέφυγαν να χρησιμοποιήσουν όρους που θα προκαλούσαν αρνητική εντύπωση στο τεχνικό κατεστημένο της βιομηχανίας, όπως για παράδειγμα υπολογιστής, προγραμματισμός κλπ. Επίσης το όνομα του νέου προϊόντος απέφυγαν να το χρησιμοποιήσουν ολοκληρωμένο και προτιμούσαν να αναφέρουν τη συσκευή ως PLC. Επίσης προσπάθησαν να μην αλλάξουν τον μέχρι τότε τρόπο εργασίας στον τομέα των αυτοματισμών σχετικά με τον σχεδιασμό ενός βιομηχανικού αυτοματισμού. Οι εταιρίες κατασκευής PLC έδειξαν ένα νέο τρόπο σχεδιασμού του αυτοματισμού στους μηχανικούς, όπου ουσιαστικά τους μάθαιναν προγραμματισμό. Οι πρώτες γλώσσες προγραμματισμού αντέγραφαν μέσω μιας ειδικής συσκευής προγραμματισμού με πλήκτρα το ηλεκτρολογικό σχέδιο του εκάστοτε αυτοματισμού.

Με τον τρόπο αυτό η είσοδος του PLC στην βιομηχανία υπήρξε επιτυχής και ομαλή. Σήμερα, ο κλασικός αυτοματισμός με ηλεκτρονόμους τείνει να εκλείψει. Όλες οι καινούριες εγκαταστάσεις χρησιμοποιούν PLCs με αποτέλεσμα μετά από λίγα χρόνια ελάχιστες εγκαταστάσεις θα χρησιμοποιούν πίνακες κλασικού αυτοματισμού.

Σήμερα, τα PLC έχουν εξελιχτεί πάρα πολύ σε σχέση με τα πρώτα μοντέλα της δεκαετίας του '80. Σε αυτό συνέβαλλε και το γεγονός ότι το προσωπικό της βιομηχανίας έχει εκπαιδευτεί κατάλληλα στον χειρισμό και προγραμματισμό των PLCs. Επομένως, ένας ηλεκτρολόγος θα πρέπει να διαθέτει στοιχειώδεις γνώσεις σχετικά με τα ηλεκτρονικά και τις βασικές αρχές των υπολογιστών διαφορετικά θα είναι πολύ δύσκολο να διαβάσει και να κατανοήσει το εγχειρίδιο ενός PLC.

Η χρήση των PLC σε σχέση με τον κλασικό αυτοματισμό έχει οικονομικό όφελος πρώτιστα στις εταιρίες που παράγουν τα είδη αυτοματισμού. Σε μια εταιρεία παραγωγής ηλεκτρολογικού εξοπλισμού, η παραγωγή ενός μεγάλου αριθμού βοηθητικών ηλεκτρονόμων, χρονικών και απαριθμητών θα ήταν μεγάλο. Επιπρόσθετα, η ψηφιοποίηση σε όλους τους τομείς (και όχι μόνο στον τομέα των αυτοματισμών) οδηγεί σε σημαντική μείωση του κόστους παραγωγής των αντίστοιχων συσκευών.

Τέλος, είναι γεγονός ότι, ενώ σε όλους τους τομείς της παραγωγής η εξελικτική πορεία πέρασε από τις ηλεκτρικές συσκευές, στις ηλεκτρονικές με λυχνίες, μετά στις ηλεκτρονικές με ημιαγωγούς (τρανζίστορ) και κατέληξε στις συσκευές με μικροϋπολογιστές (ψηφιακές συσκευές), στον τομέα των αυτοματισμών η μετάβαση πραγματοποιήθηκε σχεδόν κατευθείαν από τους ηλεκτρολογικούς αυτοματισμούς στους αυτοματισμούς με PLCs. Η πλέον πιθανή αιτία για το γεγονός αυτό είναι οι ραγδαίες εξελίξεις στον τομέα της ηλεκτρονικής σε σχέση με τον τομέα της βιομηχανίας που εξελίσσεται με σημαντικά πιο αργούς ρυθμούς.

1.2 Πλεονεκτήματα των PLCs

Συγκριτικά με τον κλασικό αυτοματισμό τα πλεονεκτήματα των PLCs είναι πάρα πολλά. Ενδεικτικά αναφέρονται τα πιο σημαντικά από αυτά:

- Στο στάδιο της μελέτης δεν υπάρχει το πρόβλημα του αν επαρκούν οι επαφές των ηλεκτρονόμων, των χρονικών ή των εξωτερικών τερματικών.
- Η λειτουργία του αυτοματισμού, μπορεί να αλλάξει πολύ εύκολα σε οποιοδήποτε στάδιο (μελέτη, κατασκευή, θέση σε λειτουργία ή αργότερα)
- Ο εντοπισμός των βλαβών διευκολύνεται, γιατί για κάθε εξωτερική εντολή υπάρχει αντίστοιχο LED. Επίσης, η ροή του αυτοματισμού μπορεί να παρακολουθηθεί άνετα, με τη βοήθεια μιας συσκευής προγραμματισμού.
- Ο αυτοματισμός παραδίδεται συντομότερα σε λειτουργία, γιατί ο προγραμματισμός του PLC μπορεί να γίνει παράλληλα με την τοποθέτηση και συρμάτωση.

- Δεν υπάρχει το γνωστό πρόβλημα των "μη ενημερωμένων" σχεδίων του πίνακα αυτοματισμού μετά από λίγο καιρό. Το PLC έχει πάντα αποθηκευμένο στην μνήμη του το τελευταίο πρόγραμμα, το οποίο μπορεί να ανακτηθεί με μια συσκευή προγραμματισμού.
- Υπάρχει σημαντική οικονομία στο χώρο, τη συντήρηση και το κόστος.
- Η τοποθέτηση μπορεί να γίνει χωρίς κίνδυνο και μέσα σε πεδία ισχύος.
- Ένα PLC μπορεί να συνδεθεί με περιφερειακές μονάδες για την επιτήρηση και τον έλεγχο της εγκατάστασης (οθόνη, εκτυπωτής, πληκτρολόγιο). Με τον τρόπο αυτό καταργείται το κλασικό μιμητικό διάγραμμα και ο πίνακας χειρισμών. Σε πολλές περιπτώσεις το PLC διασυνδέεται με κάποιο ηλεκτρονικό υπολογιστή με σκοπό την μεταφορά δεδομένων.
- Η γλώσσα προγραμματισμού είναι προσαρμοσμένη στο βιομηχανικό αυτοματισμό. Συνεπώς είναι προσιτή στο προσωπικό που μέχρι σήμερα συντηρούσε τους κλασσικούς πίνακες αυτοματισμού.
- Έχουν την δυνατότητα επέκτασης, καταργώντας με αυτό τον τρόπο τα συγκεκριμένα όρια στον αριθμό, εισόδων και εξόδων ενός συστήματος κλασσικού αυτοματισμού.
- Τέλος, μπορούν να συνδεθούν με συστήματα SCADA, δίνοντας μια εμποπτική εικόνα λειτουργίας του αυτοματισμού, αλλά και επιτρέποντας την τηλεμετρία και τον τηλεχειρισμό του συστήματος αυτοματισμού.

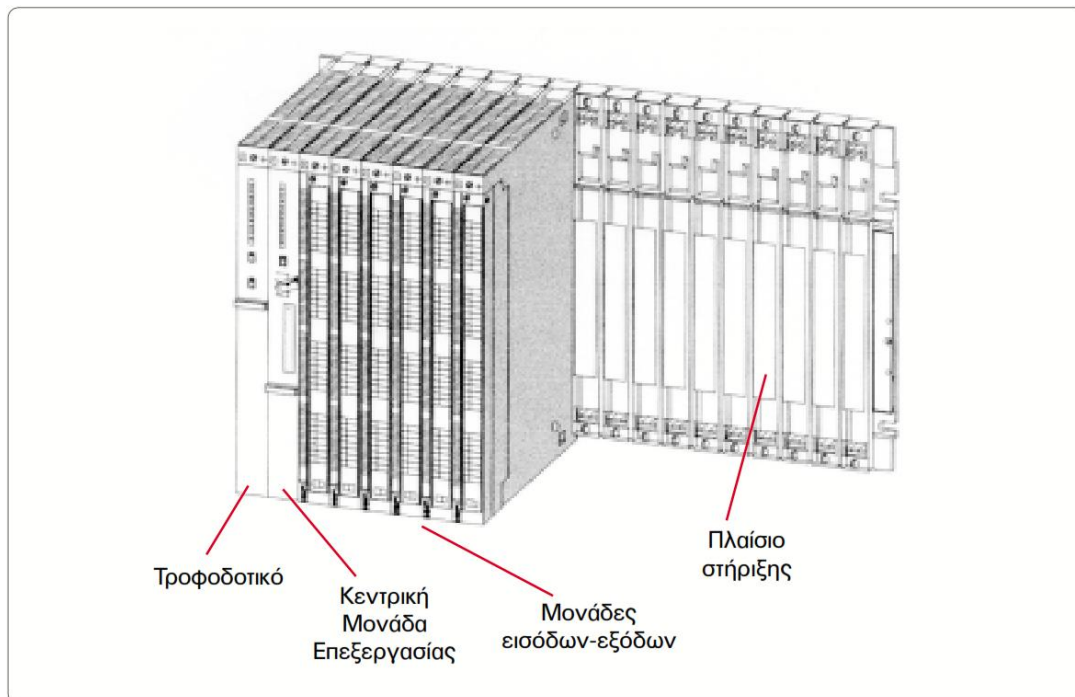
1.3 Εφαρμογές των PLCs

Σε όλες τις εγκαταστάσεις, προτείνεται η χρήση των PLCs, διότι ο έλεγχος μια εγκατάστασης παύει σιγά-σιγά να υφίσταται με τον κλασσικό αυτοματισμό. Οι δυνατότητες που μας παρέχουν τα σημερινά PLCs είναι απεριόριστες και ταυτόχρονα καταπληκτικές. Στην συνέχεια παρουσιάζονται μερικές πιο σύνθετες καταστάσεις που ενδείκνυται η χρήση των PLCs.

- Μέσω ενός πίνακα ελέγχου, μπορεί κανείς να ελέγχει διάφορες λειτουργίες και να δίνει τιμές σε μεταβλητές για τον τρόπο λειτουργίας του φωτισμού, την ρύθμιση της θερμοκρασίας, την ώρα απενεργοποίησης του εξωτερικού φωτισμού κτλ.
- Με συνδυασμό αισθητήρων φωτεινότητας και ανιχνευτή κίνησης, είναι εφικτός ο αποτελεσματικός έλεγχος του φωτισμού ενός χώρου. Φυσικά, οι ίδιοι ανιχνευτές κίνησης μπορούν να χρησιμοποιηθούν και για την λειτουργία του συναγερμού.
- Δυνατότητα οπτικοποίησης και ελέγχου της εγκατάστασης μέσω ενός υπολογιστή. Οι λειτουργίες μίας εγκατάστασης μπορούν να παρακολουθούνται από το μόνιτορ ενός υπολογιστή, ενώ με την βοήθεια του πληκτρολογίου και του ποντικιού μπορεί ο χρήστης να επεμβαίνει στον τρόπο λειτουργίας της εγκατάστασης.
- Σήματα από συναγερμό ή ανιχνευτή φωτιάς, ή άλλων ανεπιθύμητων συμβάντων, μπορούν να στέλνονται μέσω modem σε συγκεκριμένο τηλεφωνικό νούμερο, ή να εκτυπώνονται σε ένα εκτυπωτή, ή να απεικονίζονται σε πίνακα ελέγχου ή να ενεργοποιούν ανάλογα συστήματα πχ. πυρασφάλεια.

1.4 Ανάλυση των τμημάτων του PLC

Το PLC είναι μία ηλεκτρονική διάταξη η οποία από την άποψη της λειτουργίας θα μπορούσε να προσομοιωθεί με έναν πίνακα αυτοματισμού. Το PLC διαθέτει μια σειρά εισόδων και εξόδων που συνδέονται με τα στοιχεία μιας εγκατάστασης και εκτελεί ένα πρόγραμμα που καθορίζει την λειτουργία των εξόδων λαμβάνοντας υπόψη την κατάσταση των εισόδων (π.χ. η ενεργοποίηση ενός τερματικού διακόπτη σταματά τον κινητήρα μιας μεταφορικής ταινίας). Οι ομοιότητες όμως σταματούν εδώ αφού το ιδιαίτερο πλεονέκτημα των PLC είναι ότι οι μεταβλητές που καθορίζουν την συμπεριφορά των εξόδων δεν είναι σταθερή όπως σε ένα κλασικό πίνακα αυτοματισμού αλλά μπορεί να διαφοροποιηθεί με επέμβαση στο πρόγραμμα που εκτελεί το PLC χωρίς καμία επέμβαση στο υλικό (hardware) του συστήματος (Σχήμα 1.1).



Σχήμα 1.1: Δομή ενός PLC

1.4.1 Δομή του PLC

Στην αγορά υπάρχουν εκατοντάδες μοντέλα PLC κατασκευασμένα από πλήθος διαφορετικών εταιρειών. Γενικά, σε ένα PLC μπορούμε να διακρίνουμε τα παρακάτω μέρη:

- Την CPU (Central Processing Unit- Κεντρική Μονάδα Επεξεργασίας)
- Τη μονάδα τροφοδοσίας
- Τις μονάδες I/O (Input/Output –Είσοδοι/Εξοδοι)

Η CPU, η μονάδα τροφοδοσίας και οι μονάδες I/O αποτελούν την κύρια μονάδα αυτοματισμού, δηλαδή το κύριο μέρος του PLC. Σε πολλά μοντέλα, κυρίως στα μικρά μοντέλα των εταιριών, οι τρεις παραπάνω μονάδες βρίσκονται ενσωματωμένες σε μια συσκευή. Εκτός από την κεντρική μονάδα αυτοματισμού, σε ένα PLC είναι ακόμη απαραίτητα:

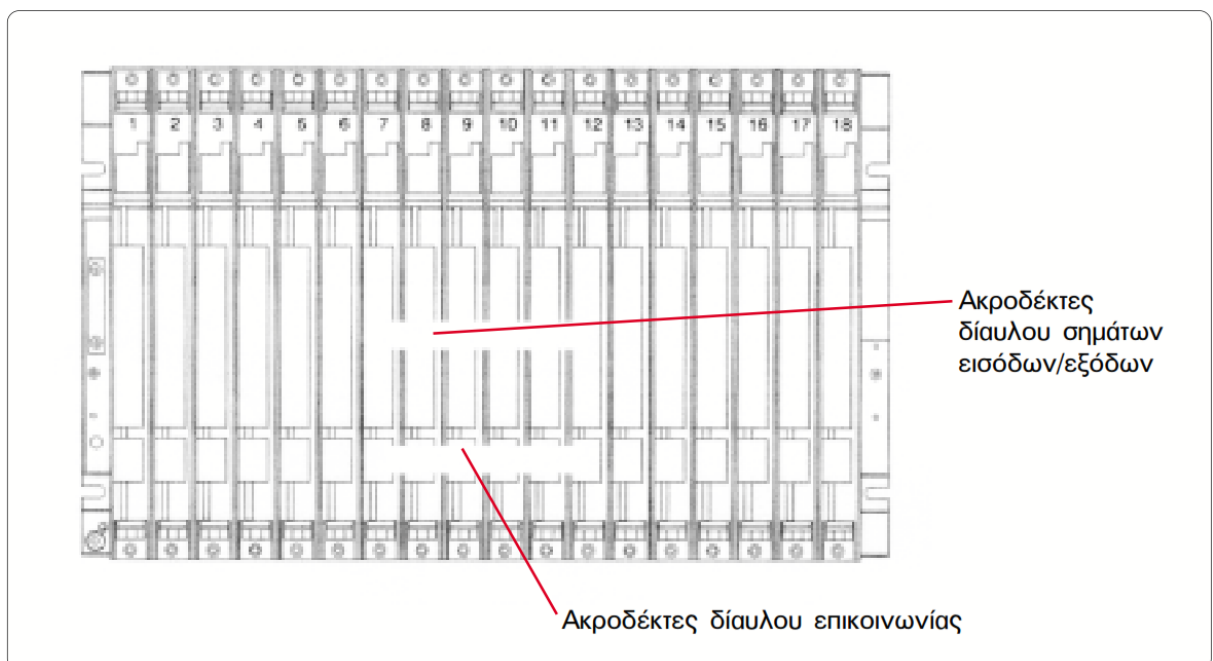
- Το πλαίσιο (rack) για την τοποθέτηση των μονάδων και των τυχόν επεκτάσεων τους.
- Η συσκευή προγραμματισμού (programmer) για τον προγραμματισμό του PLC.

Η συσκευή προγραμματισμού είναι μια συσκευή τελείως ξεχωριστή από την μονάδα αυτοματισμού. Χρησιμοποιείται για την ανάπτυξη και την φόρτωση (load) του προγράμματος στο PLC και την παρακολούθηση της λειτουργίας του αυτοματισμού μέσα από την οθόνη, που διαθέτει. Με ένα μόνο προγραμματιστή μπορούμε να χειριζόμαστε όλες τις μονάδες των PLCs μιας αυτοματοποιημένης εγκατάστασης. Σε πολλές περιπτώσεις σήμερα η συσκευή προγραμματισμού έχει αντικατασταθεί με ένα ηλεκτρονικό υπολογιστή στον οποίο έχει εγκατασταθεί το απαραίτητο λογισμικό.

1.4.2 Πλαίσιο τοποθέτησης μονάδων

Οι μονάδες ενός μεγάλου PLC τοποθετούνται σε ένα κεντρικό πλαίσιο (Σχήμα 1.1). Στο πλαίσιο αυτό είναι ενσωματωμένο ένα σύστημα αγωγών (σύστημα ζυγών) μέσω των οποίων επικοινωνούν οι διάφορες μονάδες με την CPU.

Αν οι θέσεις του κεντρικού πλαισίου, που διατίθεται, δεν επαρκούν για να τοποθετηθούν οι μονάδες I/O που απαιτούνται σε μια συγκεκριμένη εφαρμογή, τότε χρησιμοποιούνται ένα ή περισσότερα πλαίσια επέκτασης για την τοποθέτηση των πρόσθετων μονάδων. Κάθε πλαίσιο επέκτασης διασυνδέεται με το κεντρικό πλαίσιο ή με άλλα πλαίσια επέκτασης μέσω μιας μονάδας διασύνδεσης. Κάθε εταιρεία και το κάθε μοντέλο έχει το δικό του σύστημα πλαισίου. Στο Σχήμα 1.2 παρουσιάζεται ένα πλαίσιο στήριξης PLC



Σχήμα 1.2: Πλαίσιο στήριξης μονάδων PLC

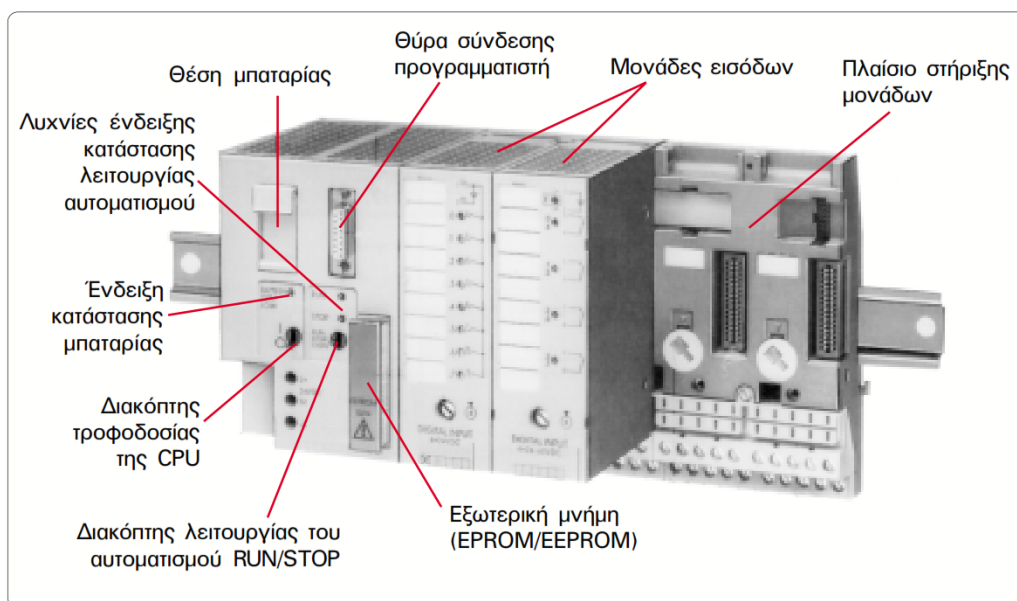
1.4.3 Μονάδα τροφοδοσίας

Η μονάδα τροφοδοσίας ενός PLC έχει σκοπό να δημιουργήσει από την τάση του δικτύου τροφοδοσίας τις απαραίτητες εσωτερικές τάσεις, που απαιτούνται για την τροφοδοσία των ηλεκτρονικών στοιχείων (τρανζίστορ, ολοκληρωμένα κυκλώματα κ.λπ.) του PLC. Οι τυπικές εσωτερικές τάσεις των PLC είναι συνήθως: DC 5 V, DC 9 V, DC 24 V.

Σε ορισμένα μοντέλα PLC, όταν το PLC δεν τροφοδοτείται από το δίκτυο, η μονάδα τροφοδοσίας διατηρεί το περιεχόμενο της μνήμης του PLC με την βοήθεια μιας μπαταρίας που διαθέτει. Σε άλλα μοντέλα PLC η παραπάνω μπαταρία βρίσκεται στην CPU. Σε πιο σύγχρονα μοντέλα η CPU είναι εφοδιασμένη με μια SD (Secured Digital) κάρτα μνήμης στην οποία αποθηκεύονται το πρόγραμμα του χρήστη και τα πλέον κρίσιμα δεδομένα της εφαρμογής.

1.4.4 Κεντρική μονάδα επεξεργασίας

Η CPU είναι η βασική μονάδα του PLC, η οποία είναι υπεύθυνη για τη λειτουργία του αυτοματισμού (Σχήμα 1.3). Επίσης στην CPU διακρίνουμε τον μικροεπεξεργαστή και τη μνήμη. Ο μικροεπεξεργαστής είναι ένα ολοκληρωμένο κύκλωμα, το οποίο ελέγχει και κατευθύνει όλες τις εργασίες της CPU.



Σχήμα 1.3:Κεντρική μονάδα επεξεργασίας και μονάδες εισόδων ενός PLC

1.4.5 Μονάδες εισόδων-εξόδων

Οι μονάδες I/O ή κάρτες I/O αποτελούν τις μονάδες επικοινωνίας της κεντρικής μονάδας με τον έξω κόσμο (Σχήμα 1.3). Στις κάρτες εισόδων συνδέονται στοιχεία που δίνουν τις απαραίτητες πληροφορίες στο PLC όπως αισθητήρες, διακόπτες, μπουτόν κλπ. Επίσης στις κάρτες εξόδων συνδέονται στοιχεία που αποτελούν τους τελικούς αποδέκτες όπως ηλεκτρονόμοι ισχύος, ηλεκτρομαγνητικές βαλβίδες, ενδεικτικές λυχνίες κλπ.

Η CPU μπορεί να δεχτεί ψηφιακά σήματα εισόδου και εξόδου χαμηλής τάσης και πολύ μικρού ρεύματος. Η τάση που δέχεται η CPU είναι συνήθως 0 Volt για το λογικό '0' και 5 Volt για το λογικό '1'. Το ρεύμα εισόδου καθώς και το ρεύμα εξόδου είναι της τάξης των mA. Οι κάρτες εισόδων και εξόδων αναλαμβάνουν να προσαρμόσουν τα σήματα εισόδου και εξόδου, που προέρχονται από τα ηλεκτρολογικά στοιχεία του αυτοματισμού, σε σήματα που μπορεί να δεχτεί και να επεξεργαστεί η CPU. Η προσαρμογή αυτή γίνεται με τη χρήση ηλεκτρονικών ισχύος, όπως τρανζίστορ ισχύος, θυρίστορ και τριάς, είτε ακόμη με τη χρήση κατάλληλων μικρό-ηλεκτρονόμων.

Κάθε PLC καταλήγει πάντα σε ακροδέκτες εισόδων στους οποίους καταλήγουν αγωγοί που έρχονται από αισθητήρια (τερματικούς διακόπτες, πιεζοστάτες κλπ.), διακόπτες, μπουτόν κλπ. και σε ακροδέκτες εξόδων όπου καταλήγουν οι αγωγοί που τροφοδοτούν τα πηνία των ηλεκτρονόμων ισχύος, τις ηλεκτρομαγνητικές βαλβίδες, τις ενδεικτικές λυχνίες κλπ.

1.4.6 Κάρτες ψηφιακών εισόδων

Ο ρόλος τους είναι να μεταφέρουν την εικόνα της εγκατάστασης στη CPU, όπως για παράδειγμα ότι ενεργοποιήθηκε ένας τερματικός διακόπτης ή ότι πατήθηκε από τον χειριστή ένα μπουτόν. Η πληροφορία αυτή μεταφέρεται ηλεκτρικά στον ακροδέκτη της κάρτας ψηφιακών εισόδων, ψηφιοποιείται και αποθηκεύεται στη μνήμη απεικόνισης εισόδων της CPU.

Υπάρχουν κάρτες ψηφιακών εισόδων του ενός Byte, των 2 Bytes και των 4 Bytes. Γνωρίζοντας ότι ένα Byte αποτελείται από 8 bits και ότι κάθε bit ισοδυναμεί με μία ψηφιακή είσοδο, έχουμε κάρτες με 8, 16 και 32 εισόδους αντίστοιχα. Επίσης οι κάρτες εισόδων εκτός από το μέγεθος της χωρητικότητας τους διακρίνονται, σε συνεχούς ρεύματος (DC) με τάση 24V και εναλλασσομένου ρεύματος (AC) με τάση 230V. Μια σημαντική παράμετρος της κάρτας ψηφιακών εισόδων είναι η καθυστέρηση που εισαγάγει στη μεταφορά του σήματος προς την CPU. Μια τυπική τιμή για τα PLC της Siemens είναι της τάξης των 1.2-25 msec.

Τέλος, οι κάρτες ψηφιακών εισόδων διαθέτουν γαλβανική απομόνωση, δηλαδή η τάση εισόδου δεν οδηγείται απευθείας τα κυκλώματα της CPU αλλά μέσω ενός οπτοζεύκτη (optocoupler). Το σημαντικό πλεονέκτημα που προσφέρει η γαλβανική απομόνωση συνίσταται στην ασφάλεια των κυκλωμάτων της CPU έναντι τυχόν υπερτάσεων ή εσφαλμένων τροφοδοτήσεων στην κάρτα ψηφιακών εισόδων.

1.4.7 Κάρτες ψηφιακών εξόδων

Ο ρόλος τους είναι να μετατρέπουν τις αποφάσεις που πήρε η CPU σε εντολές προς την εγκατάσταση. Οι αποφάσεις αυτές βρίσκονται καταχωρημένες στη μνήμη απεικόνισης εξόδων της CPU και μετατρέπονται σε ηλεκτρικά σήματα από τις κάρτες ψηφιακών εξόδων. Οι κάρτες ψηφιακών εξόδων λειτουργούν ως διακόπτες, στους οποίους δίνουμε εμείς τάση και όταν κλείσει ο διακόπτης, η τάση διοχετεύεται προς το υπόλοιπο κύκλωμα. Η τάση λειτουργίας των ψηφιακών εξόδων είναι του μεγέθους 24VDC και 230VAC. Κατά αντιστοιχία με τις κάρτες ψηφιακών εισόδων έχουμε κάρτες ψηφιακών εξόδων των 1, 2 ή 4 Bytes ή των 8, 16, 32 εξόδων αντίστοιχα. Στην περίπτωση των ψηφιακών καρτών εξόδου όταν μια έξοδος είναι ενεργοποιημένη εμφανίζει τάση 24 VDC, ενώ όταν είναι ανενεργή τάση 0 VDC. Τέλος, οι κάρτες ψηφιακών εξόδων διαθέτουν γαλβανική απομόνωση για προστασία της CPU.

1.4.8 Κάρτες αναλογικών εισόδων

Επιτελούν την ίδια λειτουργία με τις κάρτες ψηφιακών εισόδων, με τη μόνη διαφορά ότι αυτές μπορούν να διαβάζουν συνεχείς τιμές τάσης ή ρεύματος. Δηλαδή μια κάρτα αναλογικών εισόδων "διαβάζει" την τιμή του αναλογικού σήματος, εύρους 0-10 V για παράδειγμα, και τη μετατρέπει σε ένα ακέραιο αριθμό. Προφανώς για να γίνει η μετατροπή αυτή πρέπει να υπάρχει ενσωματωμένη στην κάρτα αναλογικών εισόδων μια μονάδα A/D (αναλογικό σε ψηφιακό).

Οι συνεχείς τιμές που αντιλαμβάνονται οι κάρτες αναλογικών εισόδων είναι είτε τιμές συνεχούς τάσης εύρους 0-10 V είτε τιμές συνεχούς ρεύματος εύρους 0-20 mA ή 4-20 mA.

Γίνεται φανερό ότι πριν την κάρτα αναλογικών εισόδων πρέπει να υπάρχει κατάλληλος αισθητήρας ο οποίος θα μετατρέπει το προς μέτρηση φυσικό μέγεθος σε μια από τις παραπάνω αποδεκτές τιμές τάσης ή ρεύματος. Τέλος, κάθε αναλογική είσοδος της κάρτας ονομάζεται κανάλι (channel) και μπορεί να μετρήσει είτε τάση είτε ρεύμα. Για τον σκοπό αυτό εκτός από την κατάλληλη συνδεσμολογία θα πρέπει να πραγματοποιηθεί και η αντίστοιχη δήλωση στο περιβάλλον προγραμματισμού του PLC. Έτσι λοιπόν υπάρχουν κάρτες αναλογικών εισόδων με 2,4 ή 8 κανάλια. Ανάλογα με τον A/D μετατροπέα που διαθέτουν, οι κάρτες αναλογικών εισόδων έχουν ακρίβεια 12 ή 14 bits. Σε κάθε περίπτωση το αποτέλεσμα της μετατροπής καταλαμβάνει 2 Bytes στην μνήμη απεικόνισης εισόδων της CPU.

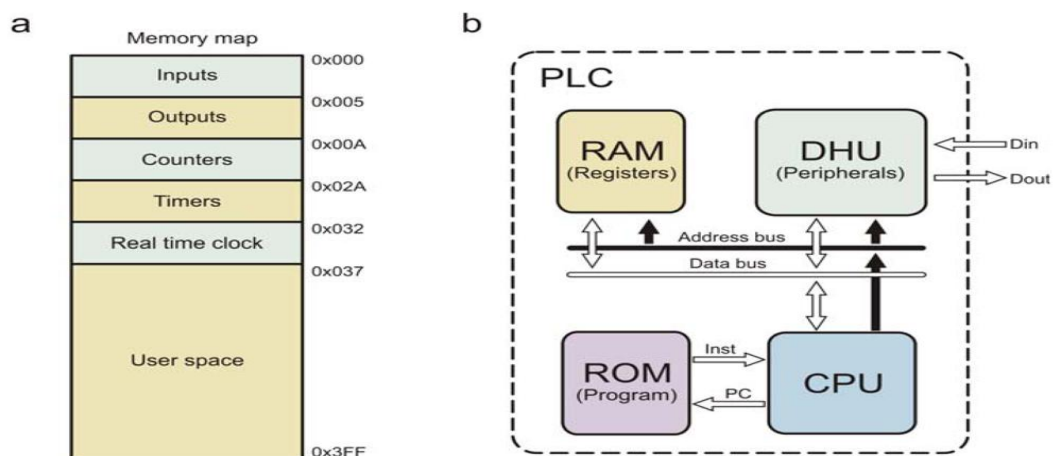
1.4.9 Κάρτες αναλογικών εξόδων

Κατά αναλογία με τις κάρτες ψηφιακών εξόδων, ο συγκεκριμένος τύπος καρτών επιτρέπει την ενεργοποίηση ή τη ρύθμιση διαφόρων διατάξεων με βάση την κατάσταση των εισόδων αλλά και το πρόγραμμα που εκτελείται στο PLC.

Η πιο σημαντική διαφορά των καρτών αναλογικών εξόδων από τις αντίστοιχες ψηφιακές, είναι ότι η έξοδος δεν είναι ένα διακριτό σήμα με δύο δυνατές τιμές, αλλά ένα συνεχές σήμα τάσης (0-10 V) ή ρεύματος (0-20 mA ή 4-20 mA). Η λειτουργία των καρτών αυτών είναι αντίστοιχη με αυτή των αναλογικών εισόδων, με τη διαφορά ότι αυτές φέρουν μετατροπέα D/A (ψηφιακό σε αναλογικό), αφού σ' αυτές η ψηφιακή πληροφορία (με ακρίβεια 12 ή 14 bits) όπως αυτή προκύπτει από την επεξεργασία του προγράμματος του PLC, μετατρέπεται σε ένα αναλογικό σήμα τάσης ή ρεύματος. Και στην περίπτωση των καρτών αναλογικών εξόδων υπάρχει γαλβανική απομόνωση με χρήση οπτοζευκτών (optocoupler).

1.5 Εσωτερική δομή του PLC

Σ' ένα σύστημα αυτοματισμού, η μνήμη της CPU είναι από τα βασικά χαρακτηριστικά του. Καθώς η εξέλιξη των υπολογιστικών συστημάτων είναι γρήγορη, έχουν βελτιωθεί οι αποδόσεις και το επίπεδο αντοχής τους. Επίσης η μνήμη είναι από τις βασικές συνιστώσες στη λειτουργία του PLC λόγω των απαιτητικών εφαρμογών με σύνθετους αυτοματισμούς, επικοινωνίες, τηλεέλεγχο και internet. Τέλος η μνήμη της CPU διακρίνεται σε μνήμη RAM, ROM και EPROM.



Σχήμα 1.4: Η εσωτερική δομή ενός Προγραμματιζόμενου λογικού ελεγκτή

1.5.1 Μνήμη RAM

Η μνήμη RAM (Random Access Memory – Μνήμη Τυχαίας Προσπέλασης) χρησιμοποιείται από το λειτουργικό σύστημα της CPU για την συνεχή ανάγνωση και εγγραφή δεδομένων. Η μνήμη RAM στις περισσότερες περιπτώσεις για την καλύτερη διαχείρισή της από το λειτουργικό σύστημα οργανώνεται συνήθως στις παρακάτω περιοχές:

- Περιοχή μνήμης όπου αποθηκεύονται οι καταστάσεις των εισόδων και εξόδων. Η περιοχή αυτή ονομάζεται για τις εισόδους "εικόνα εισόδων" και για τις εξόδους "εικόνα εξόδων".
- Περιοχή μνήμης όπου αποθηκεύονται οι ενδιάμεσες πληροφορίες, που αφορούν τη λειτουργία του αυτοματισμού.
- Περιοχή μνήμης των χρονικών.
- Περιοχή των απαριθμητών.
- Περιοχή μνήμης όπου αποθηκεύονται τα προγράμματα του χρήστη, δηλαδή τα προγράμματα που λειτουργούν ένα συγκεκριμένο αυτοματισμό.

Επειδή η μνήμη RAM χάνει το περιεχόμενο της μόλις σταματήσει η τροφοδοσία της σε πολλές περιπτώσεις μένει μόνιμα υπό τάση μέσω μιας εφεδρικής μπαταρίας. Με τον τρόπο αυτό διασφαλίζεται ότι το πρόγραμμα του χρήστη όπως και μια σειρά από κρίσιμα δεδομένα παραμένουν αναλλοίωτα ακόμα και όταν χαθεί η τάση τροφοδοσίας.

1.5.2 Μνήμη EEPROM

Σε πολλές περιπτώσεις στα PLCs δεν χρησιμοποιείται μόνο ο παραπάνω τρόπος, "της πάντα τροφοδοτούμενης μνήμης RAM", για να διατηρείται το πρόγραμμα του χρήστη. Ένας πιο ασφαλής τρόπος είναι η χρήση της μνήμης EEPROM (Electrically Erasable Programmable Read Only Memory), η οποία προγραμματίζεται και σβήνει με ηλεκτρικά σήματα.

Πρόκειται για μνήμη που δεν χάνει το περιεχόμενο της όταν μείνει χωρίς τροφοδοσία. Ο χρήστης έχει την δυνατότητα μέσω ενός κατάλληλου προγραμματιστή να διαμορφώνει το περιεχόμενο της EEPROM. Σε πολλά PLCs η EEPROM χρησιμοποιείται σαν "κασέτα" ώστε να είναι εύκολη η αλλαγή του τρόπου λειτουργίας του αυτοματισμού από ένα απλό χειριστή. Σε αυτή την περίπτωση το εναλλακτικό πρόγραμμα είναι αποθηκευμένο στην EEPROM και ο χειριστής απλά αντικαθιστά την "κασέτα" της EEPROM στο PLC, όταν επιθυμεί την αλλαγή του προγράμματος λειτουργίας του αυτοματισμού.

1.5.3 SD κάρτα

Σε πιο σύγχρονα PLCs η CPU είναι εφοδιασμένη με μια κατάλληλα διαμορφωμένη SD κάρτα. Σε αυτή την SD κάρτα συνήθως αποθηκεύεται το πρόγραμμα του χρήστη και μια σειρά από κρίσιμα δεδομένα για την λειτουργία της εφαρμογής. Το πλεονέκτημα της αξιοποίησης της SD κάρτας είναι ότι δεν χάνει τα δεδομένα της ακόμα και όταν διακοπεί η τροφοδοσία της CPU. Επιπλέον, υποστηρίζει μεγαλύτερες ταχύτητες ανάγνωσης και εγγραφής σε σχέση με την EEPROM και δεν απαιτείται για την εγγραφή των δεδομένων κάποια ειδική συσκευή προγραμματισμού. Για τον λόγο αυτό οι CPUs που διαθέτουν SD κάρτα αφενός δεν χρειάζονται εφεδρική μπαταρία και αφετέρου δεν διαθέτουν υποδοχή για την σύνδεση "κασετών" EEPROM.

1.5.4 Μνήμη ROM

Στη μνήμη ROM ο κατασκευαστής του PLC αποθηκεύει το λειτουργικό σύστημα του PLC, δηλαδή τις οδηγίες για όλες τις βασικές λειτουργίες που είναι απαραίτητες για να δουλέψει το PLC.

1.6 Γλώσσες προγραμματισμού

Οι γλώσσες προγραμματισμού αξιοποιούνται για την ανάπτυξη του προγράμματος του χρήστη που είναι υπεύθυνο για την υλοποίηση του επιθυμητού αυτοματισμού. Ο χρήστης έχει την δυνατότητα να επιλέξει την γλώσσα προγραμματισμού που επιθυμεί μέσα από ένα σύνολο διαφορετικών γλωσσών προγραμματισμού σύμφωνα με τις ιδιαίτερες ανάγκες ή προτιμήσεις του. Οι περισσότερες από τις διαθέσιμες γλώσσες προγραμματισμού είναι κοινές σε όλα τα PLCs ανεξάρτητα από την κατασκευάστρια εταιρία.

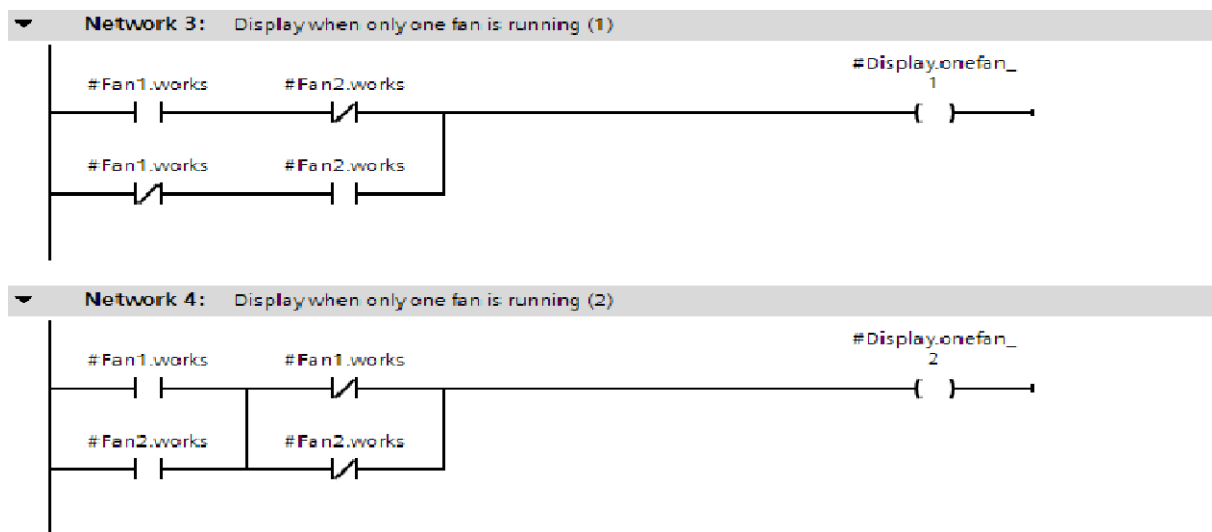
Η Siemens για τον προγραμματισμό των PLCs της προσφέρει το περιβάλλον προγραμματισμού STEP 7 το οποίο υποστηρίζει τις γλώσσες προγραμματισμού LAD (Ladder Logic Diagram), FBD (Function Block Diagram), STL (Statement List), SCL (Structured Control Language) και GRAPH. Οι πιο κοινές γλώσσες προγραμματισμού στη βιομηχανία είναι Ladder, FBD και STL. Στο Σχήμα 1.5 παρουσιάζονται οι υποστηριζόμενες γλώσσες προγραμματισμού από τα λογισμικά πακέτα SIMATIC Manager και TIA Portal της Siemens.

| The programming languages of STEP 7 | |
|--|--|
| STEP 7 V5.5 Platform SIMATIC Manager | STEP 7 V11 Platform TIA Portal |
| LAD Ladder logic | LAD Ladder logic |
| FBD Function Block Diagram | FBD Function Block Diagram |
| STL Statement List | STL Statement List |
| Option packages | SCL Structured Control Language |
| S7-SCL Structured Control Language | GRAPH Sequence control |
| S7-GRAPH Sequence control | |

Σχήμα 1.5: Γλώσσες προγραμματισμού του STEP 7

1.6.1 Ladder

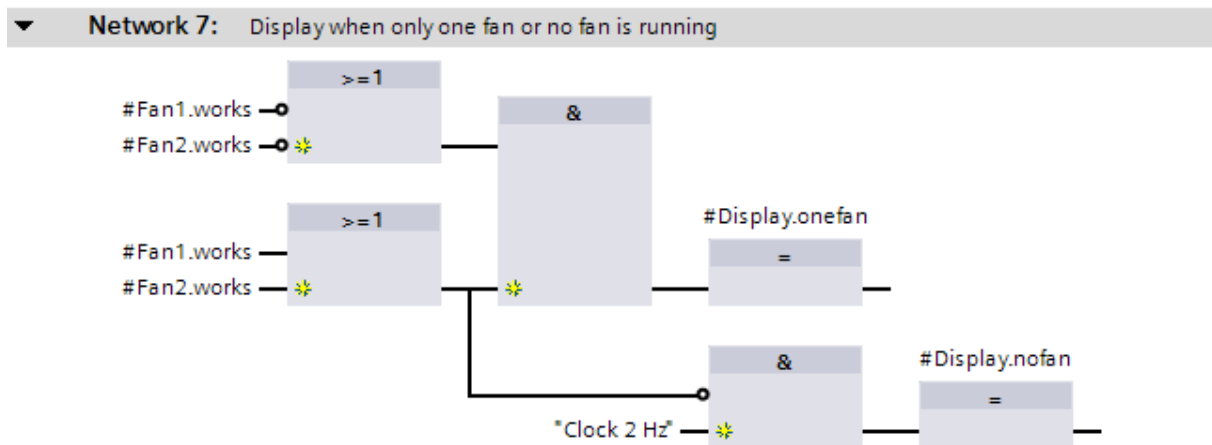
Η Ladder είναι μια γλώσσα προγραμματισμού που αποτελείται κυρίως από γραφικά στοιχεία. Η σύνταξη των εντολών μοιάζει με το διάγραμμα κυκλώματος κλασικού αυτοματισμού κι επιτρέπει την παρακολούθηση της ροής του σήματος από τις επαφές, τις εξόδους, τα χρονικά, τους μετρητές κλπ. Ένα μπλοκ που έχει προγραμματιστεί με γλώσσα ladder χωρίζεται σε τμήματα που αναφέρονται ως "networks". Κάθε "network" αποτελείται τουλάχιστον από μια διαδικασία-λειτουργία ή κάποιο block λειτουργίας. Το Σχήμα 1.6 δείχνει τη δομή ενός μπλοκ με προγραμματισμό ladder σε STEP 7 μέσω του TIA Portal. Μπορεί να δοθεί τίτλος για το συγκεκριμένο block, κάποιο σχόλιο "comment" της διαδικασίας και της λειτουργίας που ακολουθεί.



Σχήμα 1.6: Γλώσσα προγραμματισμού Ladder

1.6.2 Function Block Diagram

Η FBD (Function Block Diagram) είναι γλώσσα προγραμματισμού με γραφικά στοιχεία, παρόμοια της γλώσσας ladder. Οι εντολές εδώ αναπαριστούνται με “κουτιά” και ο χρήστης μπορεί να παρακολουθεί τη ροή του σήματος. Μπορεί να δοθεί τίτλος για το συγκεκριμένο block, κάποιο σχόλιο “comment” της διαδικασίας και της λειτουργίας που ακολουθεί.



Σχήμα 1.7: Γλώσσα προγραμματισμού FBD

1.6.3 Statement List

Η STL (statement list) αποτελείται από εντολές που γράφονται ανά γραμμή. Οι λογικές πράξεις (A, AN, O, ON κλπ) επεξεργάζονται τα σήματα και βγάζουν κάποιο λογικό αποτέλεσμα. Επίσης μπορεί να δοθεί τίτλος για το συγκεκριμένο block, κάποιο σχόλιο “comment” της διαδικασίας και της λειτουργίας που ακολουθεί.

| Network 1: | | | |
|------------------|-----------|--------------------------|-----------|
| Comment | | | |
| | | | |
| | | | RLO Value |
| 1 | A | "Motor_1_Enabled" | 1 1 |
| 2 | AN | "Motor_1_EmergencyStop" | 0 1 |
| 3 | JC | n_OK | 1 |
| 4 | = | "Motor_1_Start" | 1 1 |
| 5 | AN | "Motor_1_SpeedOK" | 0 1 |
| 6 | AN | "Motor_1_BreakesEnabled" | 0 0 |
| 7 | = | "Motor_1_Stop" | 0 0 |
| 8 | JU | End | |
| 9 | n_OK: SET | | |
| 10 | AN | "Motor_1_BreakesEnabled" | |
| 11 | = | "Motor_1_Stop" | |
| 12 | End: NOP | 0 | 0 |

Σχήμα 1.8:Γλώσσα προγραμματισμού STL

1.7 Δομή προγράμματος

Το πρόγραμμα του χρήστη μπορεί να χωριστεί σε πολλά τμήματα ώστε να γίνεται ευκολότερα κατανοητό στον χρήστη και να επιταχύνεται η διαδικασία της διόρθωσης. Κάθε τμήμα του προγράμματος πρέπει να χαρακτηρίζεται από ένα τεχνολογικό ή λειτουργικό πλαίσιο και ονομάζεται μπλοκ. Ανάλογα με την χρήση και τις ιδιότητες τα μπλοκ χωρίζονται στα παρακάτω είδη:

- Μπλοκ χρήστη: Τα μπλοκ αυτά περιέχουν το πρόγραμμα και τα δεδομένα του χρήστη.
- Μπλοκ συστήματος: Τα μπλοκ αυτά αποτελούν στοιχεία του λειτουργικού συστήματος.
- Στάνταρτ μπλοκ: Τα μπλοκ αυτά είναι έτοιμα μπλοκ που παρέχονται από την Siemens.

1.7.1 Μπλοκ χρήστη

Τα μεγάλα και περίπλοκα προγράμματα «δομούνται» σε μπλοκ τα οποία εν μέρη είναι απαραίτητα. Ο χρήστης ανάλογα με την εφαρμογή έχει την δυνατότητα να επιλέξει διαφορετικά είδη μπλοκ.

- OB (Organization Block – Μπλοκ οργάνωσης).

Τα OBs αποτελούν την διεπαφή μεταξύ του λειτουργικού συστήματος και του προγράμματος του χρήστη. Οι CPUs καλούν τα αντίστοιχα OBs όταν λαμβάνουν χώρα συγκεκριμένα γεγονότα, π.χ. στην περίπτωση διακοπής. Τα OBs έχουν συγκεκριμένους αριθμούς βασισμένους στο είδος των γεγονότων, τα οποία καλούνται να χειριστούν. Το κυρίως πρόγραμμα του χρήστη βρίσκεται στο OB1.

- FB (Function Block – Μπλοκ Συνάρτησης).

Τα FBs αποτελούν μέρος του προγράμματος του χρήστη και οι κλήσεις τους μπορούν να προγραμματιστούν μέσω των παραμέτρων του μπλοκ. Κάθε FB διαθέτει την δική του μνήμη μεταβλητών η οποία περιλαμβάνεται σε ένα DB (Data Block – Μπλοκ Δεδομένων) το οποίο ονομάζεται instance (στιγμιότυπο) DB. Κατά την κλήση ενός FB ως μεμονωμένο instance το συγκεκριμένο FB συνδέεται με ένα διαφορετικό instance DB που έχει την ίδια δομή δεδομένων άλλα διαφορετικές τιμές.

- FCs (Function – Συνάρτηση).

Τα FCs αναλαμβάνουν την υλοποίηση λειτουργιών που χρησιμοποιούνται επαναλαμβανόμενα στο πρόγραμμα του χρήστη. Για τον λόγο αυτό τα FCs μπορούν να παραμετροποιηθούν. Τα FCs δεν διαθέτουν δικό τους DB με αποτέλεσμα να μην αποθηκεύουν δεδομένα. Στις περισσότερες περιπτώσεις τα FCs επιστρέφουν τα αποτελέσματα τους στο μπλοκ που τα κάλεσε.

- DB (Data Block – Μπλοκ Δεδομένων)

Τα DBs περιέχουν τα δεδομένα του προγράμματός του χρήστη. Ένα DB δημιουργείται από τον χρήστη είτε ως global (γενικό) DB, είτε ως instance DB είτε ως type (τύπος) DB. Στα global DBs ο χρήστης προγραμματίζει απευθείας το τύπο και το περιεχόμενο των μεταβλητών. Στα instance DBs, ο προγραμματισμός των σχετιζόμενων με αυτά FBs καθορίζει το είδος και το περιεχόμενο των δεδομένων των αντίστοιχων DBs. Τα type DBs περιέχουν την δομή ενός τύπου δεδομένων που έχει αναπτυχθεί από τον χρήστη και αξιοποιείται σε άλλα DBs ή στο πρόγραμμα του χρήστη.

Ο αριθμός των OBs και οι αντίστοιχοι αριθμοί τους καθορίζονται από το λειτουργικό σύστημα της CPU. Οι αριθμοί όλων των άλλων ειδών μπλοκ καθορίζονται από τον χρήστη και θα πρέπει να βρίσκονται μέσα στο επιτρεπόμενο εύρος τιμών. Ο χρήστης έχει την δυνατότητα να ορίζει ένα συμβολικό όνομα για κάθε μπλοκ και στη συνέχεια να αναφέρεται σε αυτό αξιοποιώντας το συμβολικό όνομα.

1.7.2 Μπλοκ συστήματος

Τα μπλοκ συστήματος αποτελούν στοιχεία του λειτουργικού συστήματος και χωρίζονται σε SFCs (System Functions – Συναρτήσεις Συστήματος), SFBs (System Function Blocks – Μπλοκ Συνάρτησης Συστήματος) και SDBs (System Data Blocks – Μπλοκ Δεδομένων Συστήματος). Τα μπλοκ συστήματος παρέχουν πρόσβαση στον χρήστη σε έναν αριθμό από σημαντικές λειτουργίες του λειτουργικού συστήματος όπως ο καθορισμός του εσωτερικού ρολογιού της CPU, ή οι διάφορες συναρτήσεις επικοινωνίας.

Ο χρήστης έχει την δυνατότητα να καλέσει τα SFCs και τα SFBs αλλά δεν μπορεί να τροποποιήσει ή να επαναπρογραμματίσει το κώδικα τους. Τα μπλοκ συστήματος δεν καταλαμβάνουν από μόνα τους χώρο στην μνήμη του χρήστη. Μόνο οι κλήσεις των μπλοκ συστήματος και τα instance DBs των SFBs αποθηκεύονται στην μνήμη του χρήστη.

1.7.3 Στάνταρτ μπλοκ

Εκτός από τα FCs και τα FBs που μπορεί να αναπτύξει ο χρήστης από μόνος του, έχει την δυνατότητα να καλέσει και τα στάνταρτ μπλοκ που έχουν αναπτυχθεί από την Siemens. Τα στάνταρτ μπλοκ είναι διαθέσιμα μέσω του λογισμικού STEP 7 είτε ως εκτεταμένες εντολές είτε ως βιβλιοθήκες. Η συμπεριφορά των στάνταρτ μπλοκ είναι ίδια με την συμπεριφορά των μπλοκ του χρήστη και καταλαμβάνουν χώρο στην μνήμη του χρήστη. Επίσης τα στάνταρτ μπλοκ έχουν τον δικό τους αριθμό και μοιράζονται το εύρος της αρίθμησης με τα μπλοκ του χρήστη. Επομένως όταν κληθεί ένα στάνταρτ μπλοκ ο αριθμός αυτός δεν μπορεί να χρησιμοποιηθεί από κάποιο μπλοκ χρήστη.

1.7.4 Δομή μπλοκ

Συνήθως τα μπλοκ αποτελούνται από τρία μέρη:

- Τον αριθμό του μπλοκ που περιέχει τις ιδιότητες του μπλοκ, όπως το όνομα του κλπ.

- Το μέρος των δηλώσεων όπου δηλώνονται οι τοπικές μεταβλητές του μπλοκ.
- Το μέρος του προγράμματος όπου περιέχονται οι εντολές του προγράμματος .

Ένα DB αποτελείται από τα παρακάτω μέρη:

- Την επικεφαλίδα του μπλοκ που περιέχει τις ιδιότητες του μπλοκ.
- Το μέρος των δηλώσεων που περιέχει τις διευθύνσεις των δεδομένων με τους αντίστοιχους τύπους δεδομένων.
- Το μέρος της αρχικοποίησης, στο οποίο μπορούν να οριστούν οι αρχικές τιμές για μεμονωμένες διευθύνσεις δεδομένων.

1.8 Μορφές προγραμματισμού

Για την ευκολότερη ανάπτυξη και ανάλυση ενός περίπλοκου αυτοματισμού θα πρέπει να χωριστεί η εφαρμογή σε μικρότερα μέρη σύμφωνα με τις εκάστοτε ανάγκες. Στην συνέχεια θα πρέπει να διαμορφωθούν τα επιμέρους μέρη καθορίζοντας τις λειτουργίες που θα υλοποιούν και καθορίζοντας τα εσωτερικά σήματα προς τα επιμέρους μέρη της εφαρμογής. Η παραπάνω λογική μπορεί να ακολουθηθεί και κατά την ανάπτυξη του προγράμματος, με αποτέλεσμα η δομή του προγράμματος να ανταποκρίνεται στον διαχωρισμό της εφαρμογής.

Ένα σημαντικό πλεονέκτημα της παραπάνω μεθοδολογίας είναι ότι ένα τέτοιο πρόγραμμα μπορεί να αναπτυχθεί ευκολότερα και να υλοποιηθεί από περισσότερα από ένα άτομα. Επίσης, χωρίζοντας το πρόγραμμα σε μέρη είναι πιο εύκολη και πιο αποτελεσματική η αποσφαλμάτωση του.

Σύμφωνα με την μεθοδολογία που ακολουθείται για την υλοποίηση μιας εφαρμογής ακολουθούνται συνήθως οι παρακάτω προγραμματισμοί:

- Γραμμικός προγραμματισμός.

Σε αυτή την περίπτωση όλο το κυρίως πρόγραμμα του χρήστη είναι γραμμένο στο OB1. Η τμηματοποίηση του προγράμματος πραγματοποιείται με τον χωρισμό του προγράμματος σε networks όπου σε κάθε network υλοποιείται μια συγκεκριμένη διαδικασία.

- Τμηματοποιημένος προγραμματισμός

Ο τμηματοποιημένος προγραμματισμός βασίζεται στον γραμμικό προγραμματισμό με την διαφορά ότι το κυρίως πρόγραμμα που υλοποιείται στο OB1 χωρίζεται σε επιμέρους μπλοκ. Η εκτέλεση των επιμέρους μπλοκ πραγματοποιείται με την σειρά που καλούνται στο OB1. Με τον τρόπο αυτό αφενός το πρόγραμμα είναι πιο ευανάγνωστο και αφετέρου απλοποιείται η διαδικασία της αποσφαλμάτωσης

- Δομημένος προγραμματισμός

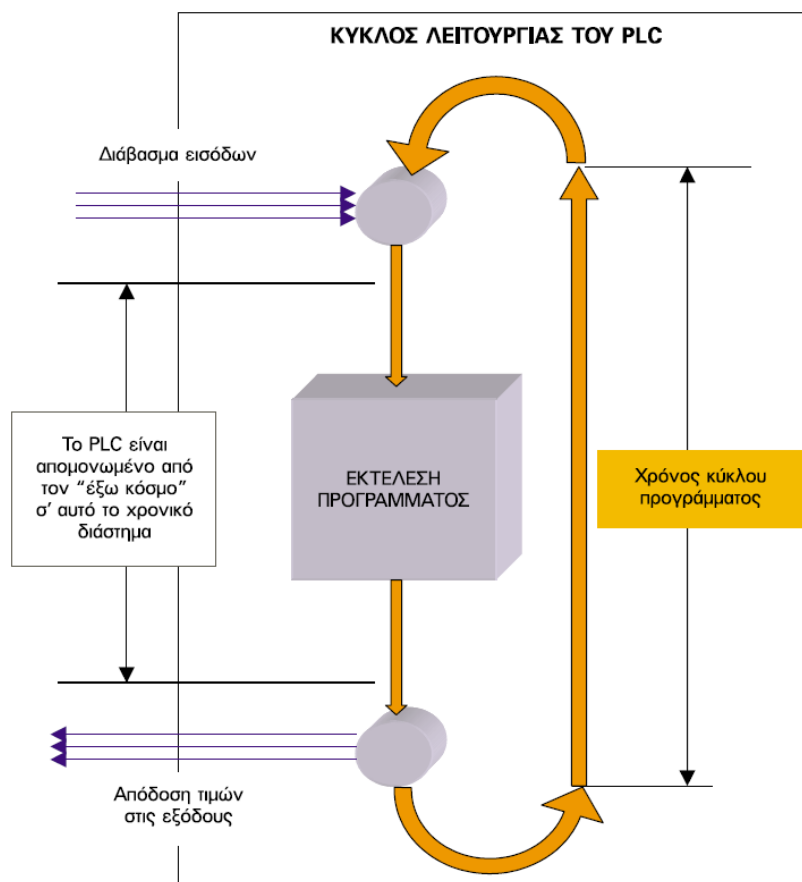
Ο δομημένος προγραμματισμός χρησιμοποιείται για την υλοποίηση απαιτητικών και πολύπλοκων εφαρμογών. Το κυρίως πρόγραμμα που περιέχει το OB1 καλεί μια σειρά από μπλοκ που έχει αναπτύξει ο χρήστης. Κάθε ένα από τα μπλοκ του χρήστη μπορεί να καλεί και επιμέρους μπλοκ. Επίσης όταν υλοποιούνται διαδικασίες οι οποίες επαναλαμβάνονται χρησιμοποιούνται διαφορετικά στιγμιότυπα των FBs ή παραμετροποιημένα FCs που έχουν αναπτυχθεί από τον χρήστη. Τα μπλοκ που χρησιμοποιούνται σε ένα δομημένο πρόγραμμα θα πρέπει να οποία ανταλλάσσουν όσο το δυνατόν λιγότερες πληροφορίες με τα άλλα μπλοκ ώστε να απλοποιείται η επικοινωνία μεταξύ των μπλοκ. Τέλος, η οργάνωση του κυρίως προγράμματος καθορίζει την σειρά με την οποία η CPU θα εκτελέσει τα μπλοκ που έχει δημιουργήσει ο χρήστης σύμφωνα με τις ανάγκες και τις ιδιαιτερότητες της εφαρμογής.

1.9 Αρχή λειτουργίας ενός PLC

Σε ένα PLC που βρίσκεται σε κατάσταση εκτέλεσης του αυτοματισμού (RUN), η CPU ακολουθεί την λειτουργία που παρουσιάζεται στο Σχήμα 1.9. Η λειτουργία της CPU συνοψίζεται στα παρακάτω βήματα:

- Βήμα 1

Στην αρχή η CPU "διαβάζει" της εισόδους. Αυτό σημαίνει ότι για κάθε είσοδο ελέγχει αν έχει "υψηλή" τάση (λογικό '1') ή "χαμηλή" τάση (λογικό '0'). Η τιμή '0' ή '1' για κάθε είσοδο αποθηκεύεται σε μια ειδική περιοχή της μνήμης η οποία ονομάζεται εικόνα εισόδων. Η εικόνα εισόδων ουσιαστικά μπορεί να παρομοιαστεί με ένα πίνακα, στον οποίο η CPU σημειώνει τις τιμές των εισόδων.



Σχήμα 1.9: Κύκλος λειτουργίας PLC

- Βήμα 2:

Στη συνέχεια η CPU χρησιμοποιώντας σαν δεδομένα τις τιμές των εισόδων, που διάβασε, εκτελεί τις εντολές του προγράμματος. Το πρόγραμμα αυτό στην ουσία περιέχει μια σειρά από λογικές πράξεις. Η εκτέλεση του προγράμματος θα δώσει τα αποτελέσματα για τις εξόδους. Τα αποτελέσματα αυτά αποθηκεύονται στην ειδική περιοχή της μνήμης που ονομάζεται εικόνα εξόδων. Όπως η εικόνα εισόδων, έτσι και η εικόνα εξόδων περιέχει την τιμή ('0' ή '1') για κάθε έξοδο.

- Βήμα 3:

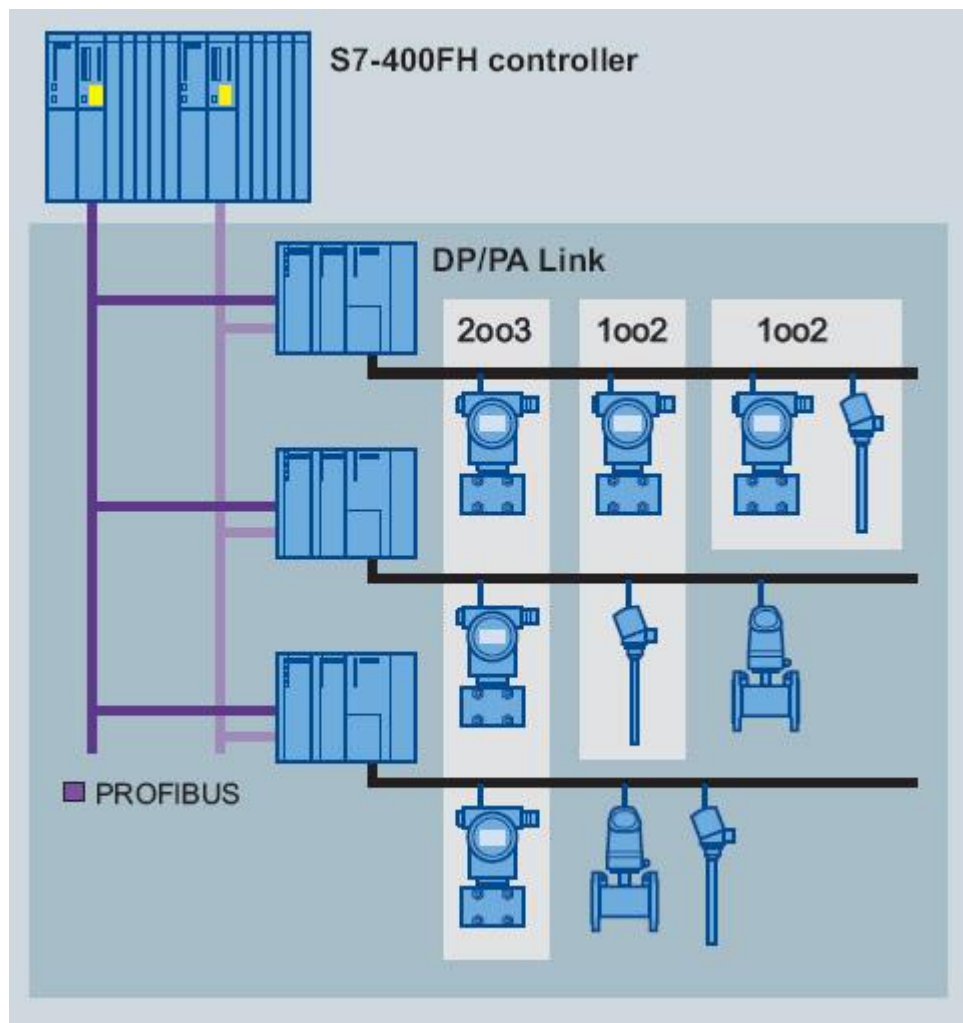
Στη συνέχεια η CPU θέτει τις τιμές της εικόνας εξόδων στις εξόδους. Αυτό σημαίνει ότι θα δοθεί "υψηλή" τάση σε όποια έξοδο έχει '1' και χαμηλή τάση σε όποια έξοδο έχει '0'. Με τη συμπλήρωση του 3ου βήματος συμπληρώνεται ένας πλήρης κύκλος λειτουργίας και η διαδικασία αρχίζει από την αρχή.

Ο κύκλος λειτουργίας εκτελείται συνεχώς όσο το PLC βρίσκεται σε κατάσταση RUN. Ο χρόνος που χρειάζεται για να εκτελέσει το PLC ένα πλήρη κύκλο λειτουργίας ονομάζεται χρόνος κύκλου και εξαρτάται από την ταχύτητα του επεξεργαστή του PLC, αλλά και από τον αριθμό και το είδος των εντολών του προγράμματος. Δηλαδή στο ίδιο PLC για ένα μεγαλύτερο πρόγραμμα έχουμε μεγαλύτερο χρόνο κύκλου.

Ο χρόνος κύκλου αποτελεί ένα μέτρο σύγκρισης μεταξύ των PLC. Για να μπορούν να συγκριθούν τα PLC ως προς την ταχύτητα εκτέλεσης ενός προγράμματος, ορίζουμε τον μέσο χρόνο κύκλου, σαν το χρόνο κύκλου ενός προγράμματος που περιλαμβάνει 1 KByte δυαδικές εντολές. Πάντως στη χειρότερη περίπτωση και σε ένα αργό PLC, ο χρόνος κύκλου δεν ξεπερνά τις μερικές εκατοντάδες millisecond.

1.10 Τρόποι επικοινωνίας των PLCs

Στις σύγχρονες βιομηχανικές εγκαταστάσεις χρησιμοποιούνται δίκτυα επικοινωνίας, που είναι γνωστά ως “Fieldbus”, και συχνά αποτελούν μέρος της υλοποίησης εφαρμογών αυτοματισμού και ελέγχου σύνθετων ηλεκτροκινητήριων συστημάτων. Τα δίκτυα αυτά σχεδιάζονται με ειδικές προδιαγραφές που τα διαφοροποιούν από τα υπόλοιπα δίκτυα υπολογιστών και τα καθιστούν ικανά να λειτουργούν σε βιομηχανικά περιβάλλοντα κάτω από πολύ δυσμενείς συνθήκες.



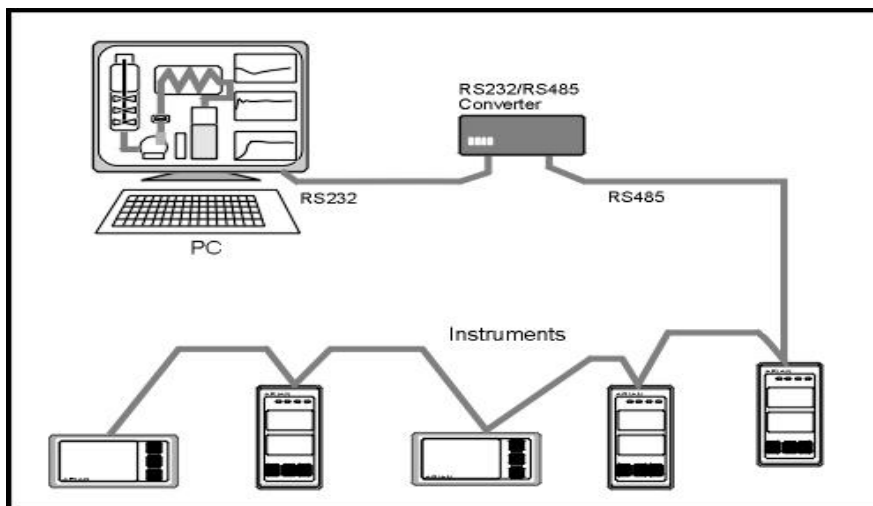
Σχήμα 1.10: Τύπος δικτύου Profibus

1.10.1 Profibus

Το **Profibus** είναι σήμερα το πιο βελτιστοποιημένο βιομηχανικό δίκτυο αφού παρέχει μετάδοση μεσαίας ποσότητας δεδομένων μεταξύ σταθμών. Η αρχή λειτουργίας του βασίζεται στο token passing, δηλαδή όλοι οι σταθμοί λαμβάνουν και στέλνουν πληροφορίες κυκλικά. Έτσι, είναι με μεγάλη διαφορά το πιο διαδεδομένο δίκτυο fieldbus σε χρήση. Η έκδοση Profibus DP είναι προσανατολισμένη στην επικοινωνία, στην κυκλική μετάδοση μικρής ποσότητας δεδομένων σε υψηλές ταχύτητες σε μεγάλες αποστάσεις μεταξύ αυτοματισμού και συστημάτων ελέγχου και παρέχει πρόβλεψη για καταναμημένες μονάδες εισόδου-εξόδου.

1.10.2 Modbus RTU

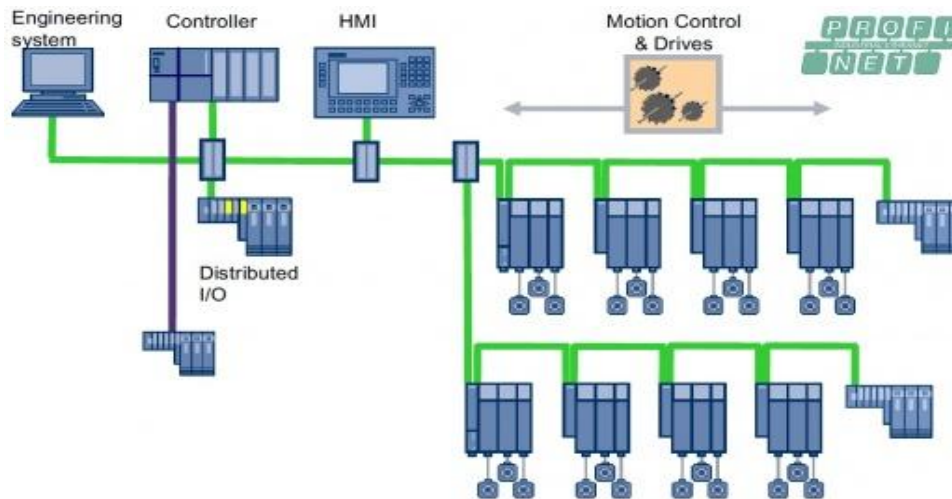
Το Modbus RTU είναι ένα παλιό αλλά πολύ δημοφιλές δίκτυο RS485 διαύλου. Το Modbus RTU είναι βασισμένο στην αρχιτεκτονική «Master/Slave» και τρέχει με ταχύτητες έως 115kbps. Αναπτύχθηκε για επικοινωνίες μεταξύ PLC, αλλά χρησιμοποιείται ευρέως και για τον προγραμματισμό παραμέτρων σε ηλεκτροκινητήρια συστήματα. Χρησιμοποιείται σήμερα σε όλο το φάσμα των βιομηχανιών παγκοσμίως λόγω της απλότητάς του και του εξαιρετικά χαμηλού κόστους υλοποίησης.



Σχήμα 1.11: Τύπος δικτύου Modbus RTU

1.10.3 Profinet

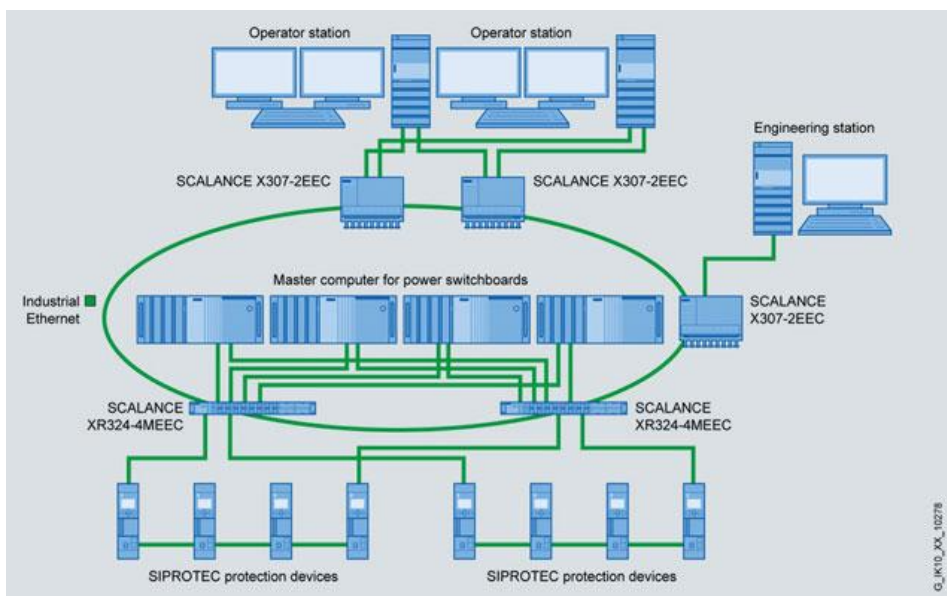
Το Profinet είναι ένα πρότυπο βιομηχανικού Ethernet που διαθέτει ένα υποδίκτυο για επικοινωνίες fieldbus μέσα στο χώρο του εργοστασίου, καθώς και ένα υποδίκτυο για επικοινωνία με το δίκτυο στο χώρο των γραφείων. Το Profinet μπορεί να διαχειριστεί ταυτόχρονα κοινά πακέτα Ethernet και πακέτα σε πραγματικό χρόνο μέσα σε ένα μέγιστο χρόνο διεκπεραίωσης. Η επικοινωνία ενός δικτύου Profinet κλιμακώνεται σε τρία βασικά επίπεδα. Το επίπεδο των συσκευών, που χρησιμοποιεί TCP/IP και πετυχαίνει κυκλικούς χρόνους της τάξης των 100ms. Αυτό το επίπεδο επιλέγεται συνήθως στην επικοινωνία μεταξύ ελεγκτών. Το δεύτερο επίπεδο είναι αυτό του πραγματικού χρόνου που λειτουργεί με χρόνους κύκλου της τάξης των 10ms και χρησιμοποιείται κυρίως για την επικοινωνία μονάδων I/O. Το τελευταίο επίπεδο είναι το ισόχρονο επίπεδο, που επιτυγχάνει χρόνους επικοινωνίας μικρότερους του 1ms. Και τα τρία επίπεδα μπορούν να συνυπάρχουν σε ένα κοινό δίαυλο μαζί με ένα κοινό δίκτυο πληροφορίας Ethernet.



Σχήμα 1.12: Τύπος δικτύου Profinet

1.10.4 Industrial Ethernet

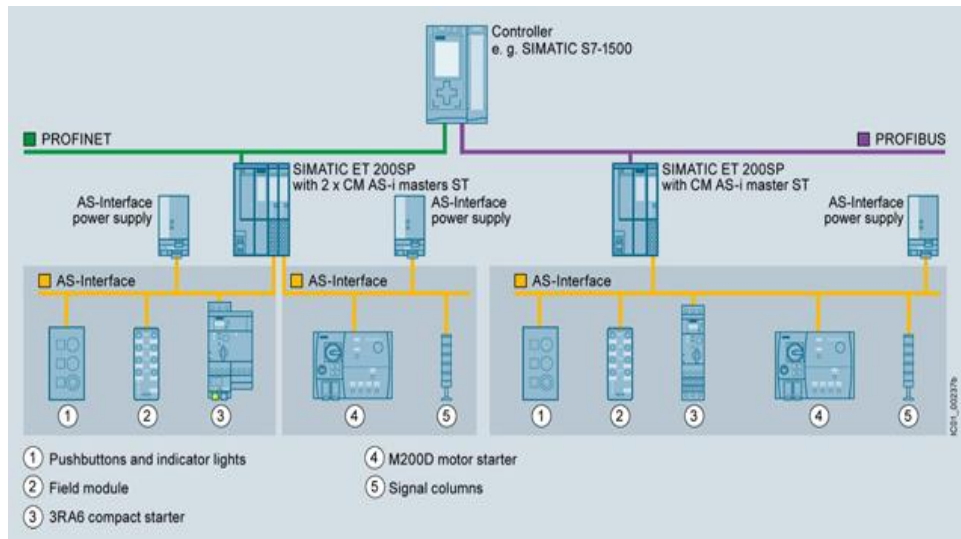
Το **Industrial Ethernet** παρουσιάζει πολλά πλεονεκτήματα, μεταξύ των οποίων συγκαταλέγονται και η διασυνδεσιμότητα σε πολλά άλλα δίκτυα, η σύνδεση με το Ίντερνετ, η ασφαλής ασύρματη μετάδοση και ο πολύ φθηνός εξοπλισμός διασύνδεσης. Σε πολλά προβλήματα του παρελθόντος μπορεί να δοθεί λύση από το Ethernet, το οποίο ωστόσο είναι ένα δίκτυο που σχεδιάστηκε για τη δικτύωση πληροφοριών, πράγμα που το καθιστά ακατάλληλο για ακριβή έλεγχο πραγματικού χρόνου σε εφαρμογές, όπως ο έλεγχος ταχύτητας και θέσης σε ηλεκτροκινητήρια συστήματα. Το Ethernet δεν μπορεί να προσφέρει στη βιομηχανία αυτό το καθολικό αποτέλεσμα συνδεσιμότητας που έχει δώσει στην πληροφορική, αλλά διατηρεί σπουδαία πλεονεκτήματα που είναι χρήσιμα και στις βιομηχανικές εφαρμογές όπως η εύκολη πρόσβαση σε μηχανές από διαδικτυακά interfaces, η αποστολή ηλεκτρονικού ταχυδρομείου, η παραγωγή πληροφορίας χρονικής σήμανσης και η αυτόματη διευθυνσιοδότηση των διάφορων συσκευών.



Σχήμα 1.13: Τύπος δικτύου Industrial Ethernet

1.10.5 AS-interface

Είναι ένα δίκτυο για τη διασύνδεση ψηφιακών αισθητηρίων. Ο όγκος των πληροφοριών είναι περιορισμένος. Η αρχή λειτουργίας του βασίζεται στη Master-Slave λογική, όπου ο κεντρικός σταθμός (Master) επικοινωνεί κυκλικά με όλους τους σταθμούς Slave που υπόκεινται σε αυτόν. Οι αποστάσεις που καλύπτονται φθάνουν τα 300 μέτρα με ταχύτητα μετάδοσης 165Kbps. Ο μέγιστος αριθμός σταθμών που αποτελούν ένα δίκτυο είναι 32.



Σχήμα 1.14: Τύπος δικτύου AS-I

1.11 Τηλεχειρισμός και εποπτικός έλεγχος

Τα συστήματα HMI (Human Machine Interface) και SCADA (Supervisory Control And Data acquisition) είναι συστήματα για την απομακρυσμένη παρακολούθηση και τον έλεγχο μιας εγκατάστασης ή εφαρμογής. Μπορούν να διαχειρίζονται τα σήματα της λειτουργίας αλλά και σήματα που παράγει η εφαρμογή. Έχουν την δυνατότητα αυτά τα σήματα ή πληροφορίες να αποστέλλονται σε κάποιο κεντρικό σύστημα ώστε να το ενημερώνουν για την κατάσταση της εφαρμογής. Επίσης μπορούν να δέχονται πληροφορίες από πολλές λειτουργίες και εφαρμογές σε μικρές και μεγάλες αποστάσεις.

Η εταιρεία Siemens παρέχει το λογισμικό πακέτο WinCC όπου ο χρήστης μπορεί να ρυθμίζει τη μορφή των γραφικών στοιχείων ή να σχεδιάσει ένα μιμικό διάγραμμα της εφαρμογής. Οι αναπαραστάσεις γίνονται με διάφορα διαγράμματα και μπορούν να αποτελούνται από σχηματικά σύμβολα όπου αντιπροσωπεύουν τα στοιχεία της διαδικασίας, από ψηφιακές φωτογραφίες του τεχνολογικού εξοπλισμού και σε κάποιες περιπτώσεις από αντικείμενα που έχουν την δυνατότητα να κινούνται. Τα σύμβολα ή τα γραφικά στοιχεία μπορεί να έχουν στατική λειτουργία ή να περιλαμβάνουν κάποια κίνηση.

Ένα ακόμα σημαντικό χαρακτηριστικό αυτών των συστημάτων είναι η καταγραφή και παρουσίαση πιθανών σφαλμάτων της εφαρμογής. Ο χρήστης έχει την δυνατότητα να μπορεί να ορίζει τα σφάλματα-Alarms, για την διάγνωση των σφαλμάτων αλλά και την σωστή λειτουργία της εγκατάστασης.

1.12 Κατηγορίες HMI και SCADA

Σε αυτά τα συστήματα γίνεται μια διαβάθμιση των συσκευών αυτών προσαρμοσμένη στις ανάγκες των εκάστοτε εφαρμογών. Οι βασικές κατηγορίες των συσκευών αυτών είναι οι εξής:

1.12.1 Push Button Panels

Τα Push Button Panels (PP) (Σχήμα 1.15) αποτελούνται από buttons και ενδεικτικές λυχνίες LED δύο χρωμάτων. Η σύνδεση της συσκευής γίνεται μέσω κάποιου δικτύου με το PLC και μπορούν να πραγματοποιηθούν ρυθμίσεις μέσω του κατάλληλου μενού. Επίσης εμφανίζονται καταστάσεις του συστήματος καθώς και μηνύματα σφαλμάτων σε απλό κείμενο.



Σχήμα 1.15: Πάνελ τύπου Push Button

1.12.2 Operator interfaces

Οι συσκευές αυτές μας δίνουν στον χρήστη τη δυνατότητα να παρατηρεί στην οθόνη τους τις πληροφορίες που σχετίζονται με την λειτουργία της εφαρμογής σε μορφή κειμένου ή σε μορφή γραφικών ενώ μέσω των ενσωματωμένων πλήκτρων μπορεί να πραγματοποιεί μια σειρά από χειρισμούς. Υπάρχουν τρεις βασικές κατηγορίες: Text Displays(TD), Operator Panels(OP) και Touch Panels(TP).



Σχήμα 1.16:Οθόνη τύπου Text Display

1.12.3 Text displays

Τα Text Displays (TD) (Σχήμα 1.16) αποτελούνται από οθόνη κειμένου και πλήκτρα πλοήγησης. Επικοινωνούν με όλα τα PLC της Siemens μέσω θύρας επικοινωνίας καθώς και με PLC άλλων κατασκευαστών. Η οθόνη τους μπορεί να απεικονίζει κείμενο σε 8 ή σε 4 γραμμές, ανάλογα με το μέγεθος των χαρακτήρων που έχει επιλεγεί. Η φωτεινότητα της οθόνης τους μπορεί να ρυθμιστεί από τον χρήστη. Στις περισσότερες περιπτώσεις τα Text Displays διαθέτουν και ένα ενσωματωμένο ρολόι πραγματικού χρόνου.

1.12.4 Operator Panels

Αυτή η κατηγορία των συσκευών χωρίζεται σε δύο ομάδες (Σχήμα 1.170, στα μικρά Operator Panels (OP) που διαθέτουν μια οθόνη κειμένου και πλήκτρα πλοήγησης και στα μεγάλα OP που διαθέτουν μια οθόνη γραφικών. Μπορούν να απεικονίζουν μηνύματα με κείμενο, μεταβλητές τιμές, να διαχειρίζονται μηνύματα βλάβης, να προσδίδουν ώρα και ημερομηνία σε κάθε συμβάν. Επίσης διαθέτουν αριθμητικό πληκτρολόγιο μέσω του οποίου ο χρήστης μπορεί να καθορίσει διάφορες μεταβλητές στο PLC. Διαθέτουν επίσης κάποια πλήκτρα ειδικών λειτουργιών (function keys) που συνήθως συνδέονται με την εκτέλεση κάποιων διαδικασιών της παραγωγής.

Τα μεγάλα συστήματα έχουν σχεδιαστεί ώστε να παρουσιάζουν σε γραφική μορφή την εγκατάσταση. Οι διάφορες παράμετροι της εγκατάστασης μπορούν να απεικονίζονται ως απλές τιμές, ιστογράμματα ή καμπύλες. Οι εικόνες στη οθόνη μπορούν να αποτελούνται από στατικά και δυναμικά μέρη. Τέλος τα μηνύματα σφάλματος μπορούν να απεικονίζονται σε ειδική οθόνη με διαφορετικό χρώμα για πρόσφατα σφάλματα ή σφάλματα που έχουν επικυρωθεί συνοδευόμενα με ώρα και ημερομηνία του συμβάντος.



Σχήμα 1.17:Οθόνη τύπου Operator Panel

1.12.5 Touch Panels

Η τεχνολογία των συσκευών αφής είναι μια επέκταση της κλασικής έκδοσης των συσκευών ελέγχου και τηλεχειρισμού. Ο χρήστης μέσω της οθόνης μπορεί να ενεργοποιεί μια προγραμματισμένη εργασία καθώς και να παρατηρεί σε ποιο στάδιο είναι μια εφαρμογή και τι λειτουργίες πραγματοποιεί. Τέλος με τις κατάλληλες ρυθμίσεις υπάρχει δυνατότητα εμφάνισης κάποιου μηνύματος ή σφάλματος.



Σχήμα 1.18:Οθόνη τύπου Touch Panel

Κεφάλαιο 2

Το λογισμικό πακέτο TIA Portal

2.1 Εισαγωγή

Το τμήμα Βιομηχανικού Αυτοματισμού της Siemens εισήγαγε το πρώτο ολοκληρωμένο λογισμικό πακέτο για την ανάπτυξη βιομηχανικών αυτοματισμών που καλύπτει όλες τις εργασίες αυτοματισμού και ονομάζεται TIA Portal (Totally Integrated Automation Portal). Αυτό το νέο λογισμικό πακέτο επιτρέπει στους χρήστες να δημιουργήσουν και να αναπτύξουν συστήματα αυτοματισμού γρήγορα και λειτουργικά. Το TIA Portal είναι σχεδιασμένο με γνώμονα την υψηλή απόδοση και την φιλικότητα προς το χρήστη ώστε να είναι κατάλληλο τόσο για αρχάριους όσο και έμπειρους χρήστες.

Η Siemens στοχεύει να εφαρμόσει το Tia Portal ως βάση όλων των μελλοντικών λογισμικών που σχετίζονται με την διαμόρφωση, τον προγραμματισμό και την ανάπτυξη εφαρμογών αυτοματισμού. Αυτό το πακέτο θα περιλαμβάνει το λογισμικό Simatic Step 7 για την ανάπτυξη εφαρμογών για τα Simatic PLC και το λογισμικό Simatic WinCC για την ανάπτυξη HMI εφαρμογών που εκτελούνται είτε σε κάποιο ηλεκτρονικό υπολογιστή είτε σε κάποια Simatic HMI συσκευή.

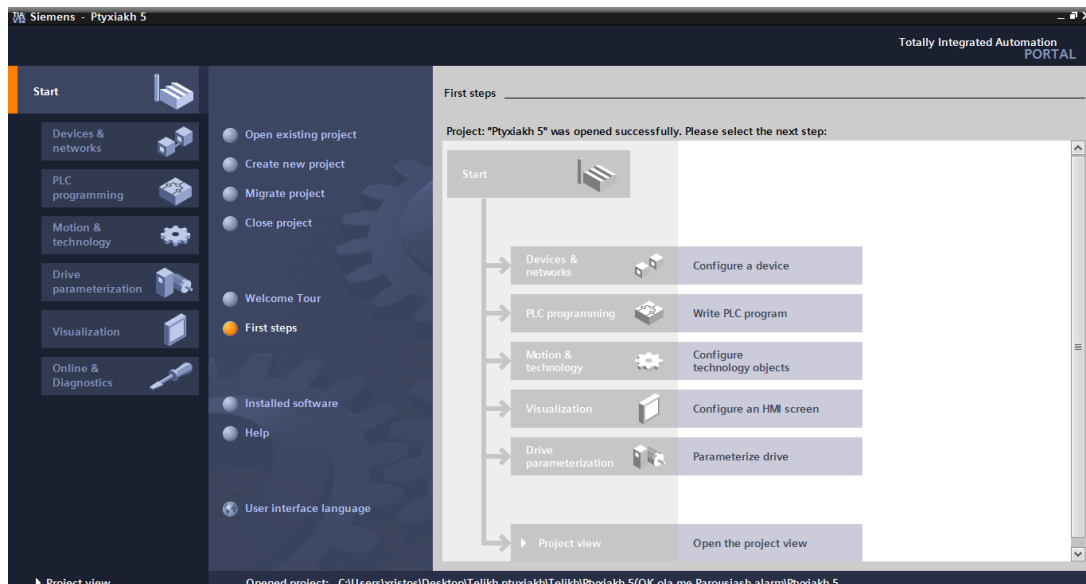
Ως αναπόσπαστο κομμάτι όλων των πακέτων λογισμικού της Siemens, το λογισμικό πακέτο Tia Portal παρέχει μια σειρά από προηγμένες υπηρεσίες που σχετίζονται με την διαμόρφωση των διασυνδέσεων μεταξύ των επιμέρους συσκευών εξασφαλίζοντας στον χρήστη ένα ενιαίο περιβάλλον για την ανάπτυξη και αποσφαλμάτωση του συστήματος. Για παράδειγμα, όλες οι συσκευές και τα δίκτυα σε ένα σύστημα αυτοματισμού μπορούν να ρυθμιστούν από μια συσκευή. Επομένως τυπικές δυνατότητες που είναι στην διάθεση του χρήστη μέσω του TIA Portal είναι η πλοήγηση στο project και τις βιβλιοθήκες, η διαχείριση δεδομένων, η αποθήκευση και διαμόρφωση του project, οι διαθέσιμοι διαγνωστικοί έλεγχοι και τέλος οι online λειτουργίες.

Επιπλέον, όλα τα δεδομένα για τις παραμέτρους του PLC, όπως block και labels ή μηνύματα είναι απαραίτητο να οριστούν μόνο μία φορά, εξοικονομώντας σημαντικό χρόνο και κόστος για τον αυτοματισμό. Ο σχεδιασμός του Tia Portal βασίζεται σε προηγμένες αρχιτεκτονικές λογισμικού και στη κεντρική διαχείριση δεδομένων. Παρέχει απόλυτη συνάφεια των δεδομένων αποφεύγοντας σφάλματα που μπορεί να προκύψουν κατά την εισαγωγή δεδομένων. Ο χρήστης μπορεί εύκολα να εντοπίσει στοιχεία και μπλοκ προγράμματος σε όλο τον αυτοματισμό χρησιμοποιώντας ένα ενιαίο περιβάλλον διαχείρισης του συνολικού project. Επίσης μειώνεται σημαντικά ο χρόνος για να αντιμετωπιστεί και να διορθωθεί ένα πρόβλημα στον αυτοματισμό.

Σημαντική δυνατότητα του νέου λογισμικού είναι ότι όλα τα PLC της Siemens και τα συστήματα HMI μπορούν να διαμορφωθούν στο ίδιο περιβάλλον. Αυτό μειώνει σημαντικά τον χρόνο για την διασύνδεση και διαμόρφωση της επικοινωνίας μεταξύ των PLC και των συσκευών HMI. Το Tia Portal μπορεί να υποστηρίξει Simatic PLC που ανήκουν στις οικογένειες S7-1200, S7-300 και S7-400 καθώς και PC based σύστημα αυτοματισμού που ονομάζονται Simatic WinAC. Η υποστήριξη αυτού του ευρέως φάσματος των PLC επιτρέπει στο Simatic Step 7 την βέλτιστη αξιοποίηση των παρεχόμενων δυνατοτήτων και την κάλυψη των απαιτήσεων που έχει ένας σύγχρονο σύστημα αυτοματισμού.

2.2 Δυνατότητες του Tia Portal

Μετά την αρχική εκκίνηση, εμφανίζεται το μενού εκκίνησης (Start Portal). Το μενού εκκίνησης καθιστά διαθέσιμες όλες τις λειτουργίες και τα εργαλεία που απαιτούνται για την αντίστοιχη σειρά εργασιών μέσω του Portal view. Τα πεδία εφαρμογής, καθώς και το εύρος των λειτουργιών και των εργαλείων εξαρτάται από τις εγκατεστημένες εφαρμογές. Η έναρξη του STEP 7 επιτρέπει την επιλογή των ακόλουθων portals.(Σχήμα 2.1).



Σχήμα 2.1: Παρουσίαση Portal: Τα πρώτα βήματα ανοίγοντας νέα εργασία

Στο Devices & networks, ο χρήστης μπορεί να διαμορφώσει το υλικό του PLC, π.χ. να επιλέξει τις μονάδες του PLC, την θέση τους, και να ορίσει τις ιδιότητές τους. Στη περίπτωση που πολλές συσκευές είναι δικτυωμένες, ο χρήστης μπορεί να καθορίσει τις συνδέσεις μεταξύ των επιμέρους συσκευών.

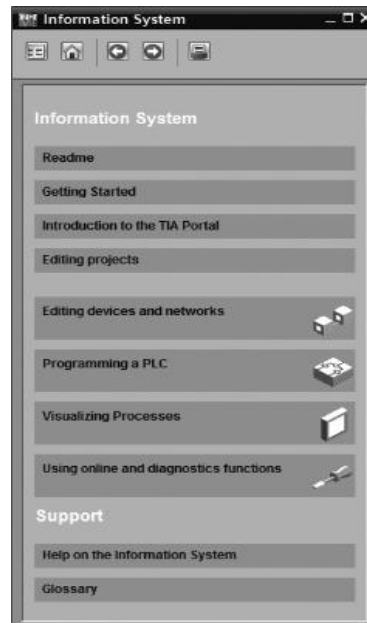
Το Tia portal στο κομμάτι του προγραμματισμού του PLC περιέχει όλα τα εργαλεία που απαιτούνται για τον προγραμματισμό του.

Στην επιλογή Visualization, ο χρήστης δημιουργεί τον έλεγχο και την διεπαφή παρακολούθησης για HMI συσκευές. Εδώ μπορεί να διαμορφώσει τις οθόνες της διαδικασίας, τα στοιχεία ελέγχου, και τα αντίστοιχα μηνύματα.

Χρησιμοποιώντας το Online & Diagnostics, ο χρήστης έχει την δυνατότητα να διασυνδέσει τη συσκευή προγραμματισμού με ένα PLC. Μέσω αυτής της επιλογής ο χρήστης έχει την δυνατότητα να φορτώσει στο PLC το πρόγραμμα που ανέπτυξε με σκοπό την υλοποίηση του αυτοματισμού. Επίσης παρέχονται στον χρήστη δυνατότητες και εργαλεία για τον γρήγορο εντοπισμό τυχόν σφαλμάτων.

Πρόσθετες λειτουργίες που περιλαμβάνονται στο μενού εκκίνησης είναι οι εξής: δημιουργία νέου έργου project (Create new project), άνοιγμα υπάρχον project (Open existing project), και μεταφορά project από άλλη έκδοση του TIA Portal (Migrate project). Η επιλογή Welcome Tour και First Step παρέχουν μια εισαγωγή στο STEP 7. Το εγκατεστημένο λογισμικό παρέχει μια επισκόπηση των περαιτέρω SIMATIC εφαρμογών που είναι διαθέσιμες στη συσκευή προγραμματισμού. Επίσης παρέχεται στο χρήστη η δυνατότητα να αξιοποιήσει την ενσωματωμένη βοήθεια του TIA Portal. Για την ενεργοποίηση της βοήθειας (Σχήμα 2.2) ο χρήστης θα πρέπει να επιλέξει με δεξί κλικ την επιλογή Help στο Portal ή να επιλέξει Help→Show Help στο κύριο μενού της επιλογής Project VIEW.

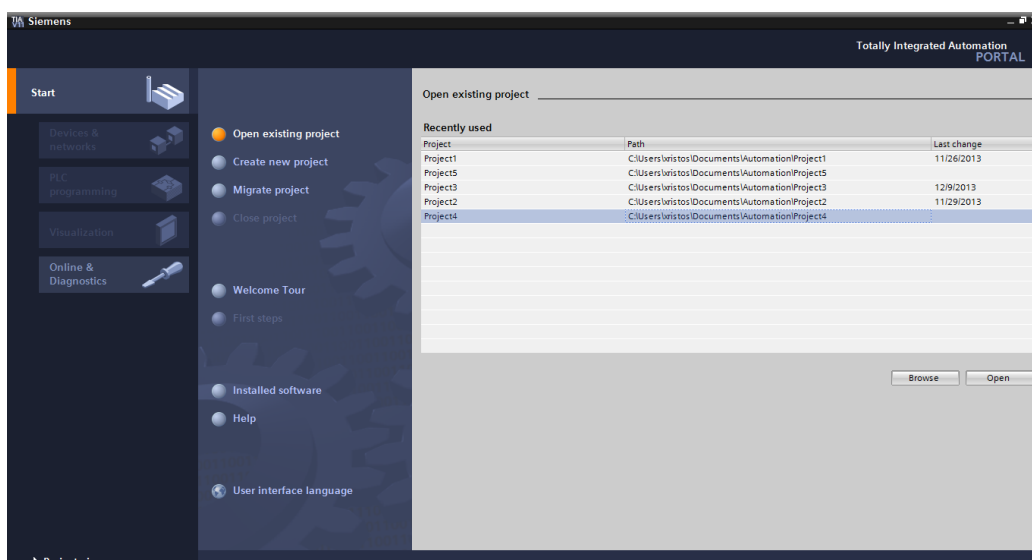
Η ηλεκτρονική βοήθεια είναι κατά προσέγγιση διαιρεμένη σύμφωνα με την επεξεργασία του project: διαμόρφωση (configuration), παραμετροποίηση (parameterization), δικτύωση των συσκευών, δόμηση και προγραμματισμός του προγράμματος, οπτικοποίηση διαδικασιών. Η επιλογή Readme παρέχει γενικές πληροφορίες για το STEP 7, καθώς επίσης περαιτέρω πληροφορίες οι οποίες δεν μπορούν να συμπεριληφθούν στην ηλεκτρονική βοήθεια.



Σχήμα 2.2: Αρχική σελίδα των πληροφοριών συστήματος

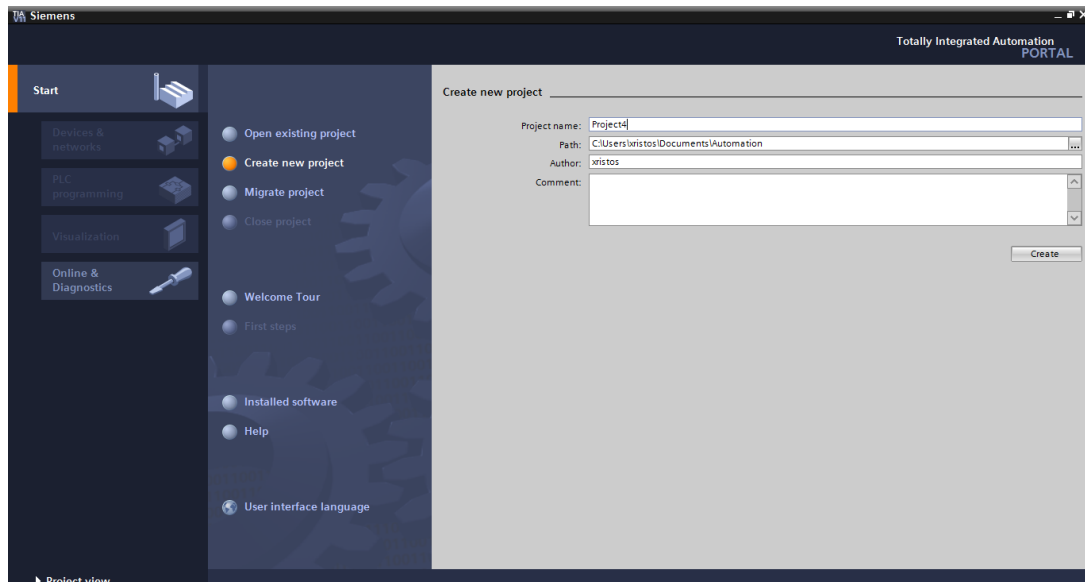
2.3 Εισαγωγή στο πρόγραμμα Tia Portal

Με το άνοιγμα του προγράμματος ο χρήστης οδηγείται στην επιφάνεια εργασίας του Tia Portal. Εκεί του παρέχονται κάποιες πρώτες δυνατότητες και λειτουργίες που μπορεί να εκτελέσει (Σχήμα 2.3).

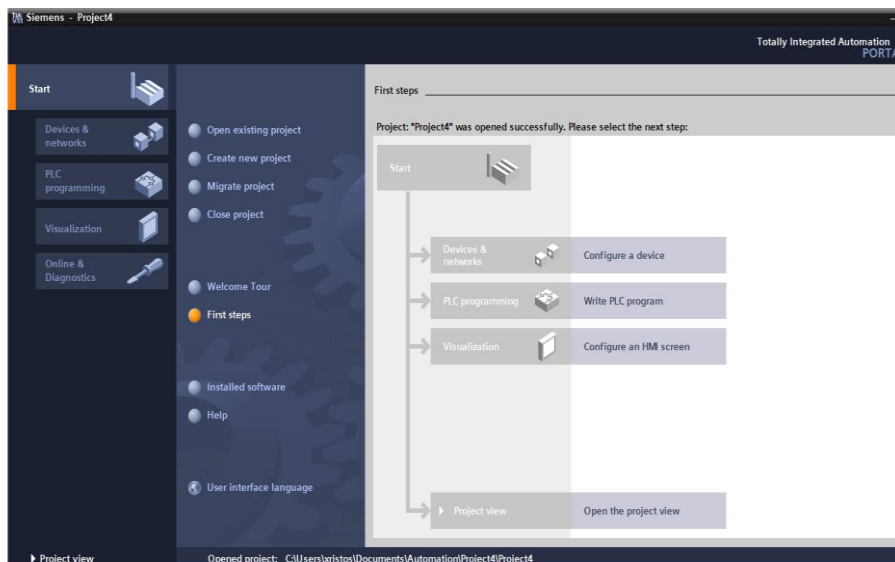


Σχήμα 2.3: Επιφάνεια εργασίας του προγράμματος

Σε αυτό το αρχικό στάδιο δίνεται η δυνατότητα να δημιουργηθεί ένα νέο project ή να γίνει η επιλογή ενός υπάρχων (Σχήμα 2.4). Στη δημιουργία ενός νέου project ο χρήστης μπορεί να δώσει στοιχεία της εργασίας που θα εκτελέσει όπως την ονομασία του project, την επιλογή του φακέλου όπου θα αποθηκευτεί το project, το όνομα του δημιουργού καθώς επίσης πληροφορίες σχετικά με το αντικείμενο που θα ασχολείται (Σχήμα 2.4).

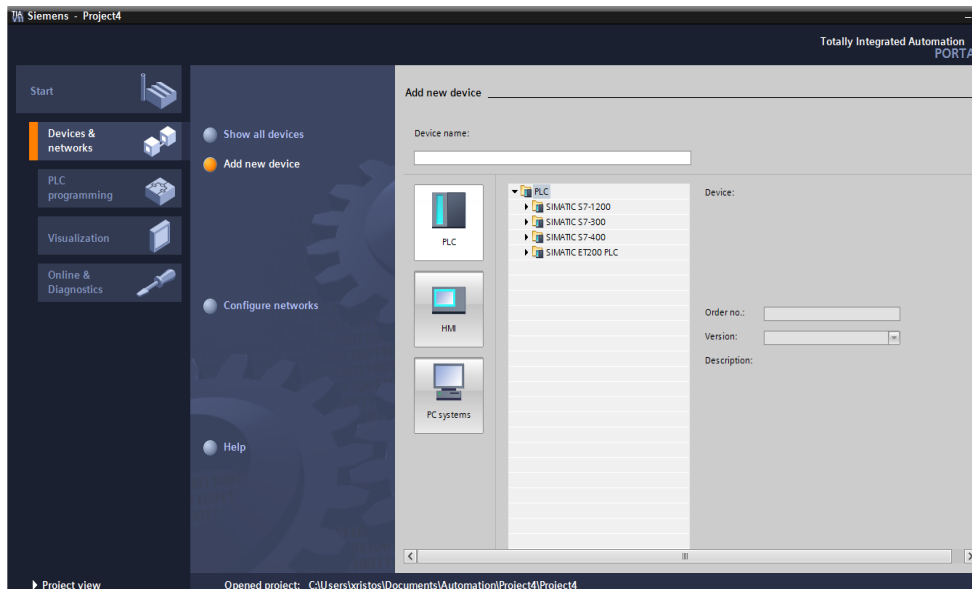


Σχήμα 2.4: Δημιουργία ενός καινούριου project



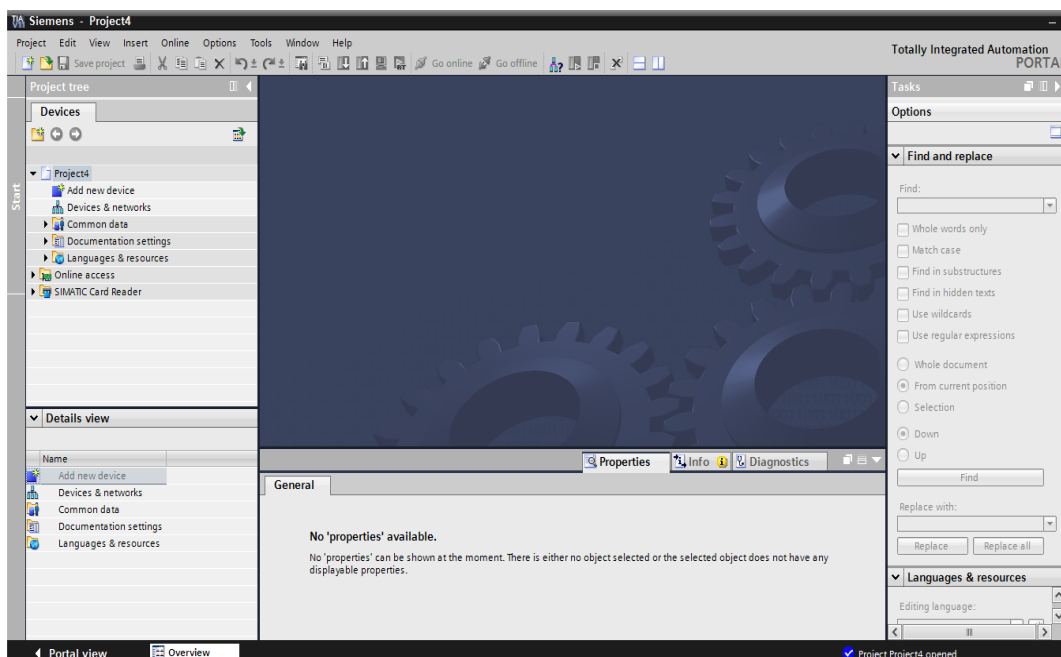
Σχήμα 2.5: Δυνατότητες προγράμματος

Εφόσον έχει γίνει επιλογή δημιουργίας project ή συνέχεια κάποιου άλλου υπάρχοντος ο χρήστης μπορεί να ξεκινήσει την επεξεργασία του. Στο Σχήμα 2.5 παρουσιάζονται οι λειτουργίες που μπορεί να επιλέξει όπως να προσθέσει μια συσκευή (Configure a device), να γράψει πρόγραμμα σε γλώσσα του PLC (Write PLC program), να κάνει ρύθμιση οθόνης HMI (Configure an HMI screen) και επίσης να έχει πρόσβαση σε ολόκληρο το project (Open the project view). Στην επιλογή για την πρόσθεση συσκευής οι επιλογές που ακολουθούν είναι PLC, HMI και PC systems (Σχήμα 2.6).



Σχήμα 2.6: Επιλογές για την πρόσθεση συσκευής στο project

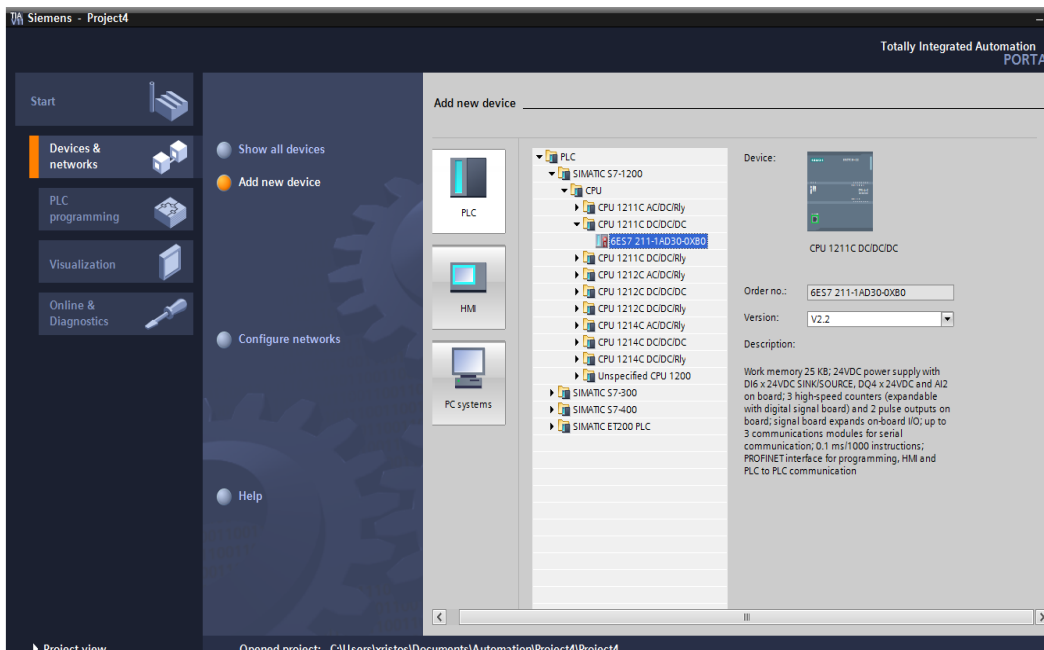
Μέσω της επιλογής Open the project view ο χρήστης μεταφέρεται στο περιβάλλον για την επεξεργασία του project (Σχήμα 2.7). Εκεί περιλαμβάνονται τα πλαίσια Project tree, Work area, tool window και property view. Στο Project tree παρουσιάζεται το όνομα του project, οι συσκευές που έχουν επιλεγεί και η διασύνδεση μεταξύ των επιλεγμένων συσκευών. Στην περιοχή Work area πραγματοποιούνται οι κύριες εργασίες του project ανάλογα με τις επιλογές του χρήστη. Στο πλαίσιο Tool window βρίσκονται όλα τα απαραίτητα εργαλεία για την υλοποίηση του project καθώς περιλαμβάνει τις βιβλιοθήκες με τα απαραίτητα στοιχεία. Τέλος στο Property View παρουσιάζονται πληροφορίες που σχετίζονται με την εκτέλεση του προγράμματος, τα πιθανά σφάλματα που παρουσιάστηκαν κατά την εκτέλεση και επίσης εκεί μπορούν να πραγματοποιηθούν ορισμένες ρυθμίσεις οι οποίες σχετίζονται με τις λειτουργίες των επιλεγμένων συσκευών.



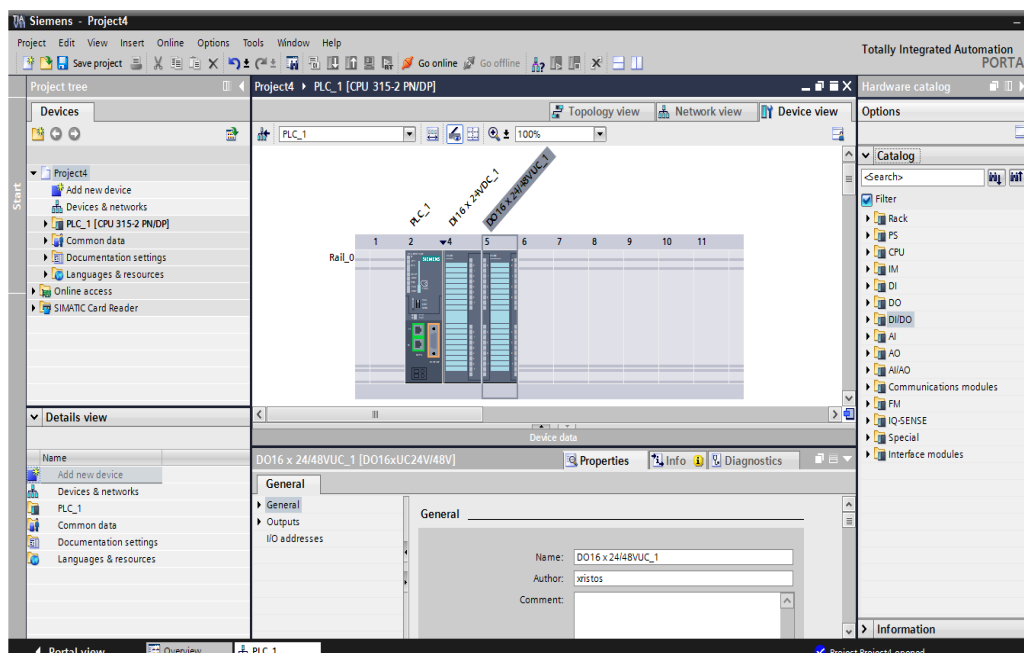
Σχήμα 2.7: Επιφάνεια επεξεργασίας του project

2.4 Ανάπτυξη του προγράμματος του PLC

Για την κατασκευή ενός προγράμματος για PLC πρέπει να έχει προηγηθεί η δημιουργία ενός project ή να χρησιμοποιηθεί κάποιο υπάρχον. Η επιλογή του PLC εξαρτάται από το είδος της εφαρμογής που θέλει να υλοποιήσει ο χρήστης. Στην επιφάνεια όπου παρουσιάζονται όλες οι επιλογές του TIA Portal (Σχήμα 2.5) αρχικά θα πρέπει να επιλεγεί η επιλογή Configure a device ώστε ο χρήστης να επιλέξει την επιθυμητή συσκευή (Σχήμα 2.8).



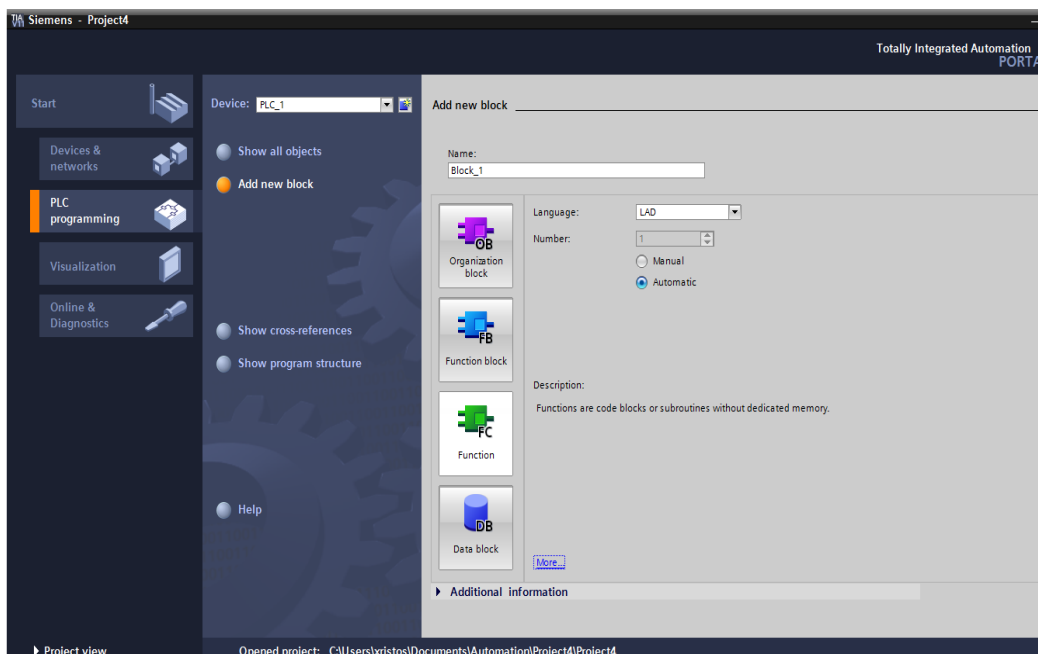
Σχήμα 2.8: Επιλογή PLC



Σχήμα 2.9: Επεξεργασία PLC του project

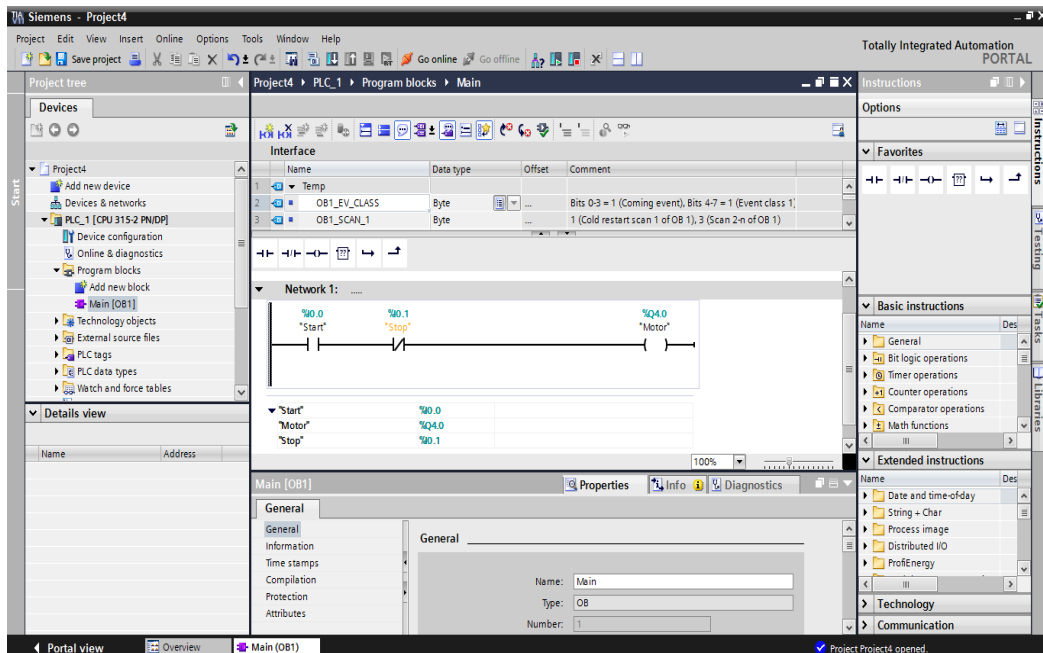
Για την διαμόρφωση του PLC ο χρήστης μέσα από τις διάφορες κατηγορίες που υπάρχουν θα πρέπει να επιλέξει την CPU που επιθυμεί σύμφωνα με τις ανάγκες της εφαρμογής. Όταν ο χρήστης επιλέξει μια CPU εμφανίζονται στο δεξιό μέρος του TIA Portal η σχηματική της παράσταση, ο κωδικός της, η έκδοση της και τα τεχνικά χαρακτηριστικά της επιλεγμένης CPU. Στην συνέχεια ο χρήστης μεταφέρεται στην επιφάνεια επεξεργασίας του project. Στο Hardware Catalog υπάρχουν οι διαθέσιμες μονάδες που μπορεί να αξιοποιήσει ο χρήστης ώστε να ολοκληρώσει την διαμόρφωση του PLC (Σχήμα 2.9).

Μετά από την ολοκλήρωση της διαμόρφωσης του PLC δίνεται η δυνατότητα στον χρήστη να αναπτύξει το πρόγραμμα που θα υλοποιεί την λειτουργία του αυτοματισμού που επιθυμεί ο χρήστης. Για τον σκοπό αυτό ο χρήστης μπορεί να επιστρέψει στο Portal View και επιλέξει την επιλογή PLC programming. Με την επιλογή Add new block ανοίγει ένα παράθυρο (Σχήμα 2.10) όπου εκεί γίνεται η επιλογή του είδους του block και της γλώσσας που θα χρησιμοποιηθεί για την ανάπτυξη του κώδικα που θα περιέχει το συγκεκριμένο block. Τα είδη γλώσσας που μπορούν να χρησιμοποιηθούν για τον προγραμματισμό είναι LAD, STL, FBD, SCL. Επίσης υπάρχουν διαφορετικά είδη block τα οποία μπορούν να αξιοποιηθούν ανάλογα με το πόσο σύνθετες ή απλές είναι οι λειτουργίες που θέλει να υλοποιήσει ο χρήστης.



Σχήμα 2.10: Επιλογή κατάλληλου block

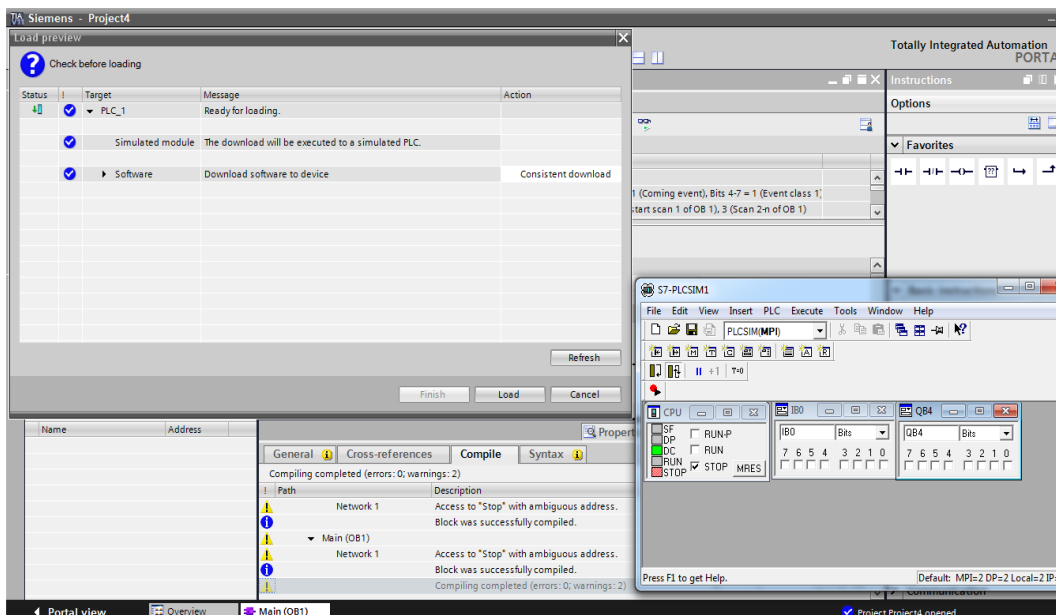
Με την επιλογή του απαραίτητου block γίνεται μετάβαση στην επιφάνεια επεξεργασίας και ο χρήστης μπορεί να αρχίσει την υλοποίηση του προγράμματος (Σχήμα 2.11) στην Work Area. Το πρόγραμμα του επιλεγμένου block χωρίζεται σε Network τα οποία περιέχουν τμήματα του κώδικα. Για κάθε η προεπιλεγμένη γλώσσα προγραμματισμού είναι η LAD. Σε κάθε εντολή που εισάγεται στο Network ο χρήστης θα πρέπει να ορίσει την απόλυτη διεύθυνση της εισόδου ή εξόδου που επιθυμεί. Επίσης ο χρήστης έχει την δυνατότητα να χρησιμοποιεί συμβολικά ονόματα για κάθε είσοδο ή έξοδο με σκοπό να κάνει το κώδικα του περισσότερο κατανοητό. Ο ορισμός του συμβολικού ονόματος μπορεί να πραγματοποιηθεί με δεξί κλικ και επιλέγοντας rename tag.



Σχήμα 2.11: Υλοποίηση προγράμματος

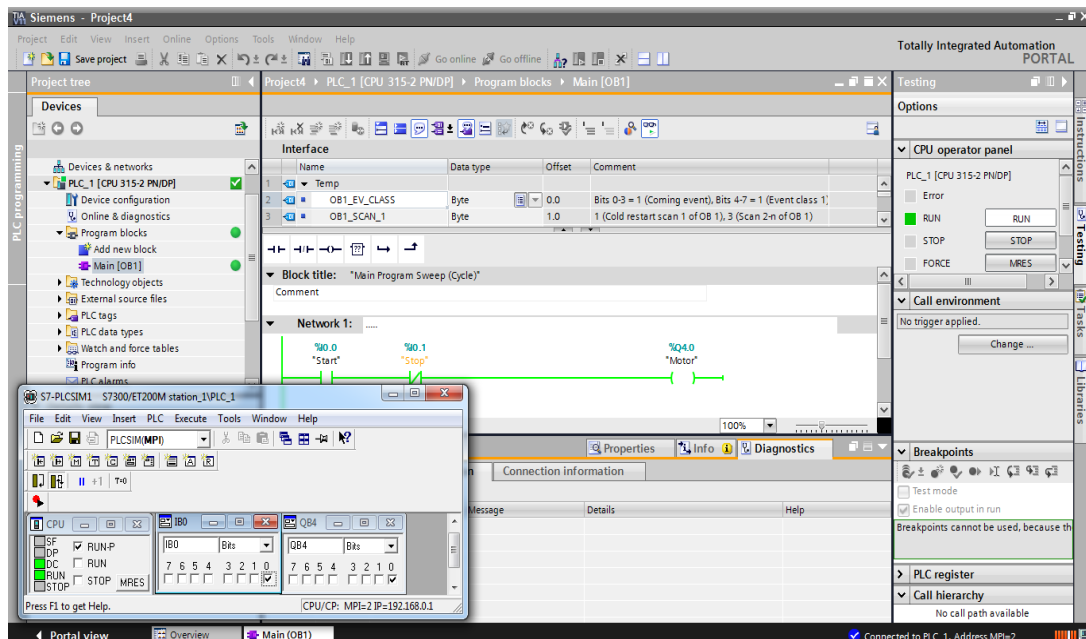
Ο χρήστης αφού έχει ολοκληρώσει το πρόγραμμα μπορεί να δει πως λειτουργεί δηλαδή τι συμβαίνει με τις εισόδους και τις εξόδους του PLC αξιοποιώντας τον εξομοιωτή του PLC που ονομάζεται PLCSIM. Αυτή η διαδικασία πραγματοποιείται επιλέγοντας από την γραμμή εργαλείων Online>Simulation>Start.

Πατώντας Start Simulation ανοίγει το παράθυρο του PLCSIM και το TIA Portal ξεκινά τον έλεγχο για τυχόν λάθη στο πρόγραμμα του χρήστη. Εφόσον δεν υπάρχουν λάθη στο πρόγραμμα ενεργοποιείται η επιλογή Load ώστε να φορτωθεί το πρόγραμμα στην εικονική CPU που εκτελείται στο παράθυρο του PLCSIM. (Σχήμα 2.12).



Σχήμα 2.12: Προετοιμασία του προγράμματος για εξομοίωση.

Αφού φορτωθεί με επιτυχία το πρόγραμμα στην εικονική CPU ο χρήστης θα πρέπει να θέσει σε λειτουργία την εικονική CPU μέσα από το παράθυρο του PLCSIM (Σχήμα 2.13) επιλέγοντας την επιλογή RUN ή την επιλογή RUN-P. Στην συνέχεια ο χρήστης έχει την δυνατότητα να καθορίζει την κατάσταση των εισόδων που επιθυμεί και να παρατηρεί τις εξόδους του εικονικού PLC ώστε να επιβεβαιώσει την λειτουργία της εφαρμογής του. Επιπρόσθετα μέσω του TIA Portal και της επιλογής Monitoring On/Off ο χρήστης μπορεί να παρατηρήσει με γραφικό τρόπο την λειτουργία του προγράμματος.

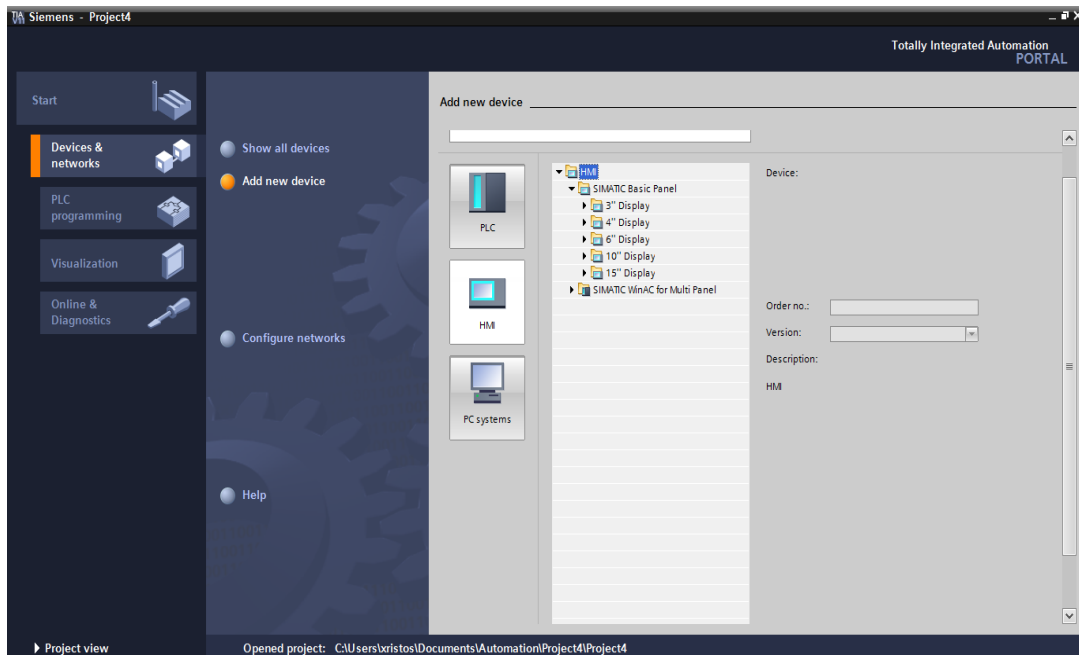


Σχήμα 2.13: Εξομοίωση του προγράμματος

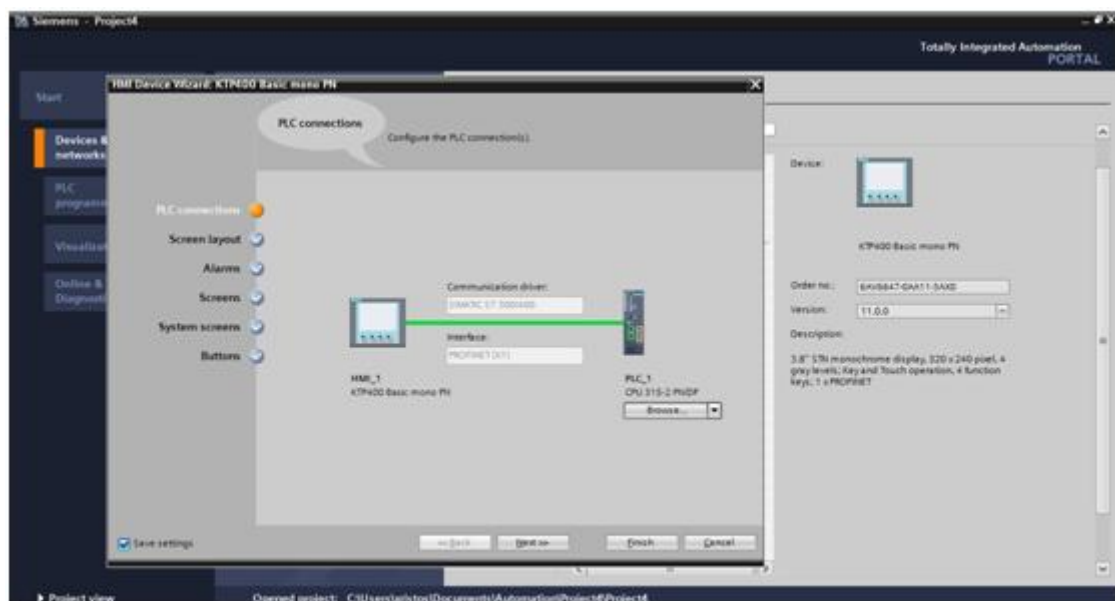
2.5 Ανάπτυξη του HMI

Μια ακόμα δυνατότητα που δίνει στον χρήστη το TIA PORTAL είναι η γραφική απεικόνιση της λειτουργίας της εφαρμογής. Η διαδικασία αυτή ονομάζεται οπτικοποίηση (Visualization). Με την δημιουργία ενός project και PLC προγράμματος γίνεται δυνατή η δημιουργία μιας γραφικής απεικόνισης.

Στην επιφάνεια επιλογών του TIA PORTAL (Σχήμα 2.5) ο χρήστης θα πρέπει να επιλέξει την επιλογή Configure an HMI screen ώστε να επιλεγεί η συσκευή η οποία θα αναλάβει την εκτέλεση του HMI. Στην συνέχεια ο χρήστης μεταφέρεται στην οθόνη για την επιλογή της HMI οθόνης που θα αναλάβει την οπτικοποίηση της εφαρμογής (Σχήμα 2.14). Σ' αυτό το σημείο θα πρέπει να σημειωθεί ότι η οθόνη που θα επιλεγεί θα πρέπει να υποστηρίζει τουλάχιστον ένα κοινό πρωτόκολλο επικοινωνίας με την CPU του PLC που επέλεξε ο χρήστης νωρίτερα. Αφού επιλεγεί η κατάλληλη οθόνη σύμφωνα με τις απαιτήσεις και τους περιορισμούς της εφαρμογής ο χρήστης θα πρέπει να πραγματοποιήσει μια σειρά από ρυθμίσεις όπως ο τρόπος επικοινωνίας μεταξύ της HMI οθόνης και της CPU. (Σχήμα 2.15).

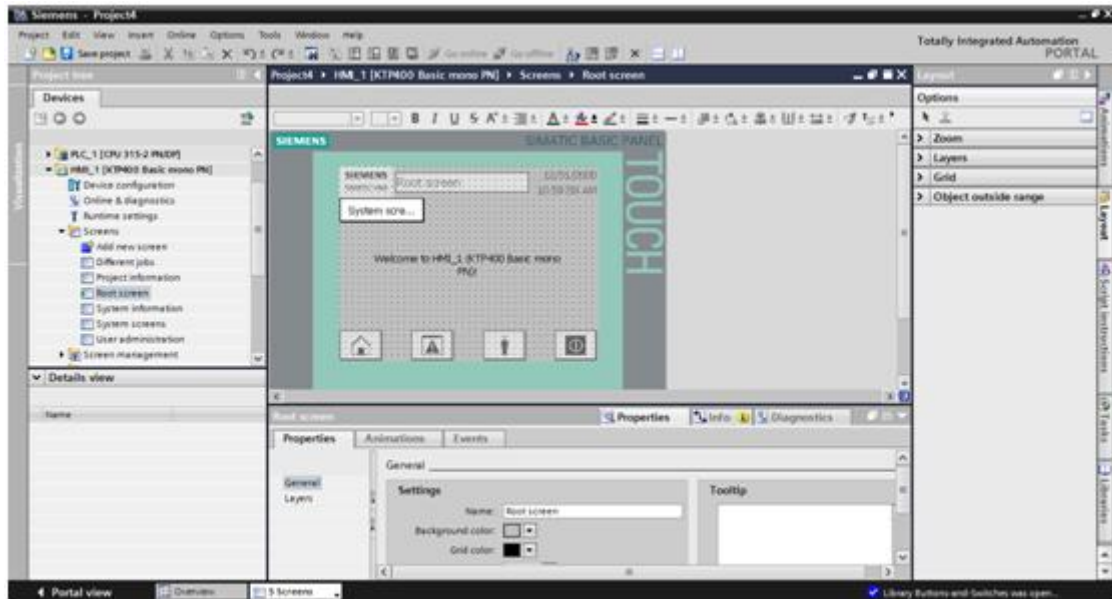


Σχήμα 2.14: Επιλογή της HMI οθόνης



Σχήμα 2.15: Ρυθμίσεις HMI οθόνης

Ο χρήστης αφού έχει κάνει τις απαραίτητες ρυθμίσεις ανάλογα με τις απαιτήσεις του επιλέγει Finish και μεταβαίνει αυτόματα στην επιφάνεια επεξεργασίας (Σχήμα 2.16). Ο κύριος σχεδιασμός του HMI θα πραγματοποιηθεί στην οθόνη Root screen. Όμως ο χρήστης έχει την δυνατότητα να προσθέσει στο HMI και δευτερεύουσες οθόνες ανάλογα με τις ανάγκες της εφαρμογής. Μέσω της επιλογής Toolbox εμφανίζεται ένα πλαίσιο το οποίο περιέχει όλα τα απαραίτητα αντικείμενα που διαθέτει το TIA PORTAL και μπορούν να χρησιμοποιηθούν από τον χρήστη για την υλοποίηση του HMI. Επίσης δίνεται στον χρήστη η δυνατότητα να εισάγει και να χρησιμοποιήσει αντικείμενα που έχει σχεδιάσει ο ίδιος.



Σχήμα 2.17: Επιφάνεια επεξεργασίας οθόνης

Κάθε φορά που τοποθετείται ένα αντικείμενο στην οθόνη του HMI ο χρήστης θα πρέπει μέσω του πεδίου Property View να δώσει την επιθυμητή ονομασία και να ορίσει την απόλυτη διεύθυνση του tag του PLC που θα αλληλεπιδρά με το συγκεκριμένο αντικείμενο ώστε το PLC να εκτελεί την αντίστοιχη λειτουργία. Αφού τοποθετηθούν όλα τα απαραίτητα αντικείμενα και διασυνδεθούν με τα αντίστοιχα tag του PLC θα πρέπει να γίνει ο έλεγχος της λειτουργίας του HMI και της διασύνδεσης με το PLC. Αρχικά θα πρέπει το εικονικό PLC εφόσον πρόκειται για εξομοίωση, να είναι σε λειτουργία και να εκτελεί το αντίστοιχο πρόγραμμα. Στην συνέχεια ο χρήστης ανοίγει την HMI οθόνη και επιλέγει Start Simulation. Με τον τρόπο αυτό ενεργοποιείται μια εικονική HMI οθόνη η οποία επικοινωνεί με το εικονικό PLC. Σε αυτό το σημείο ο χρήστης αξιοποιώντας τις δυνατότητες του PLCSIM μπορεί να επιβεβαιώνει την λειτουργία του HMI.

Κεφάλαιο 3

Εισαγωγή στον ελεγκτή PID

3.1 Ελεγκτές συστημάτων αυτόματου ελέγχου

Στα συστήματα αυτόματου ελέγχου χρησιμοποιούνται συνδεσμολογίες ελεγκτών οι οποίοι αποσκοπούν στη βελτίωση των χαρακτηριστικών του συστήματος. Με τη χρήση ελεγκτών επιτυγχάνονται καλύτερες επιδόσεις του συστήματος δηλαδή να γίνει το σύστημα πιο γρήγορο, πιο ακριβές, πιο ευσταθές και να μην επηρεάζεται από τυχόν διαταραχές.

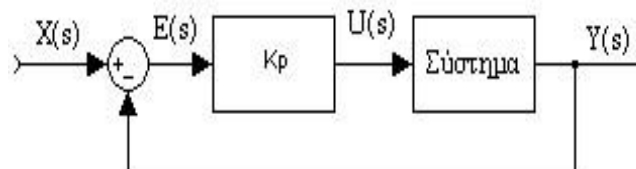
Οι ελεγκτές είναι τα κύρια στοιχεία που ελέγχουν και καθορίζουν τον τρόπο λειτουργίας των διάφορων αυτόματων συστημάτων ανάλογα με τις παραμέτρους που τους ορίζονται, έτσι ώστε η μεταβλητή που παίρνει στην έξοδο τους να είναι στην επιθυμητή τιμή που έχει οριστεί. Η κύρια δουλειά του κάθε ελεγκτή είναι να διατηρεί την έξοδο στην επιθυμητή τιμή ανεξαρτήτως από της όποιες ανωμαλίες προκύπτουν στην διαδικασία. Αν και είναι συχνά πολύ δύσκολο να επιτευχθεί αυτό, τα αποτελέσματα μπορούν να βελτιωθούν με την ανατροφοδότηση της εξόδου της διαδικασίας. Ο ελεγκτής αποτελείται από έναν ανιχνευτή λάθους και μια μονάδα τρόπου ελέγχου.

Ο ανιχνευτής λάθους υπολογίζει το λάθος αφαιρώντας την μετρούμενη μεταβλητή (cm) από το επιθυμητό σημείο (setpoint). Η μονάδα ελέγχου χρησιμοποιεί το σήμα λάθους για να παραγάγει τη δράση ελέγχου (v). Ένα από τα σημαντικότερα χαρακτηριστικά ενός ελεγκτή είναι ο τρόπος που χρησιμοποιεί το λάθος για να διαμορφώσει τη δράση ελέγχου. Έτσι λοιπόν, ο ελεγκτής μπορεί να εφαρμοστεί σε πνευματικά κυκλώματα, αναλογικά ηλεκτρονικά κυκλώματα, ή ψηφιακά ηλεκτρονικά κυκλώματα.

Οι πνευματικοί ελεγκτές χρησιμοποιούν ένα πνευματικό αντίτιμο του λειτουργικού ενισχυτή για να παραγάγουν τη δράση ελέγχου. Οι ηλεκτρονικοί αναλογικοί ελεγκτές χρησιμοποιούν ένα ανθεκτικό κύκλωμα για να υπολογίσουν το λάθος και έναν λειτουργικό ενισχυτή για να παραγάγουν τη δράση ελέγχου. Οι ψηφιακοί ελεγκτές χρησιμοποιούν έναν μικροεπεξεργαστή και έναν αλγόριθμο ελέγχου για να παραγάγουν τη δράση ελέγχου.

3.2 Αναλογικός (P) ελεγκτής

Ο αναλογικός ελεγκτής (Proportional Control) δίνει στην έξοδο του σήμα ανάλογο του σφάλματος που δέχεται στην είσοδο και στην ουσία πρόκειται για μια διάταξη ενίσχυσης του σήματος σφάλματος με κέρδος K_p . Ο αναλογικός ελεγκτής έχει την μορφή που παρουσιάζεται στο παρακάτω σχήμα.



Σχήμα 3.1: Ο αναλογικός ελεγκτής

Είναι φανερό από το σχήμα ότι η συνάρτηση μεταφοράς του ελεγκτή είναι:

$$P(s) = K_p \quad (3.1)$$

Ο αναλογικός ελεγκτής αυξάνει την ταχύτητα απόκρισης του συστήματος αλλά πιθανόν να δημιουργεί σφάλμα μόνιμης κατάστασης αναλόγως τον τύπο του συστήματος, ενώ για μηδενικό σφάλμα όπως είναι λογικό θα έχει και μηδενική έξοδο.

Σε περίπτωση που το σύστημα έχει συνάρτηση μεταφοράς

$$G_C(s) = \frac{1}{s+A} \quad (3.2)$$

τότε η συνάρτηση μεταφοράς του συστήματος κλειστού βρόχου με P ελεγκτή είναι:

$$G_C(s) = \frac{\frac{K_p}{s+A}}{1+\frac{K_p}{s+A}} \leftrightarrow G_C(s) = \frac{K_p}{s+A+K_p} \quad (3.3)$$

Επομένως αν διεγείραμε το σύστημα με σήμα εισόδου την μοναδιαία βηματική συνάρτηση θα βρίσκαμε το σφάλμα μόνιμης κατάστασης από τη σχέση:

$$e_{ss} = 1 - \lim_{s \rightarrow 0} \left(s * G_C(s) * \frac{1}{s} \right) = 1 - \lim_{s \rightarrow 0} \left(\frac{K_p}{s+A+K_p} \right) \rightarrow e_{ss} = 1 - \frac{K_p}{A+K_p} \quad (3.4)$$

Συμπεραίνουμε ότι το σφάλμα μόνιμης κατάστασης μειώνεται όσο αυξάνουμε το κέρδος του αναλογικού ελεγκτή, όμως στη πράξη δεν γίνεται να αυξήσουμε απεριόριστα το κέρδος.

Αν όμως το σύστημα είναι σε συνάρτηση μεταφοράς

$$G(s) = \frac{1}{s^2+s*A+B} \quad (3.5)$$

τότε η συνάρτηση μεταφοράς του συστήματος κλειστού βρόχου με P ελεγκτή είναι

$$G_C(s) = \frac{\frac{K_p}{s^2+s*A+B}}{1+\frac{K_p}{s^2+s*A+B}} \rightarrow G_C(s) = \frac{K_p}{s^2+s*A+B+K_p} \quad (3.6)$$

Ομοίως αν διεγείραμε το σύστημα με σήμα εισόδου μοναδιαία βηματική συνάρτηση θα βρίσκαμε το σφάλμα μόνιμης κατάστασης από τη σχέση:

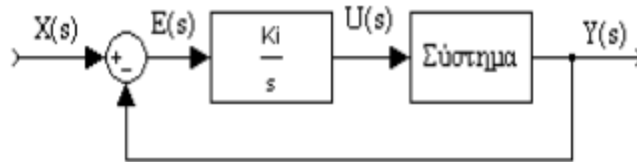
$$e_{ss} = 1 - \lim_{s \rightarrow 0} \left(s * G_C(s) * \frac{1}{s} \right) = 1 - \lim_{s \rightarrow 0} \left(\frac{K_p}{s^2+s*A+B+K_p} \right) \rightarrow e_{ss} = 1 - \frac{K_p}{B+K_p} \quad (3.7)$$

Άρα η χρήση P ελεγκτή μειώνει το σφάλμα όσο αυξάνεται το κέρδος του ελεγκτή και αυξάνεται και η φυσική συχνότητα $\omega_n = \sqrt{B + K_p}$ ενώ χωρίς τον ελεγκτή $\omega_n = \sqrt{B}$ του συστήματος κάτι που προσδίδει στο σύστημα πιο γρήγορη απόκριση. Στο σύστημα με αναλογικό ελεγκτή το A ισούται με $A=2*\zeta*\omega_n$ ομοίως με το σύστημα χωρίς ελεγκτή και παραμένει σταθερό με τη διαφορά όμως ότι αυξάνεται η φυσική συχνότητα του συστήματος ενώ μειώνεται ο συντελεστής απόσβεσης ζ του συστήματος κάτι που έχει ως αποτέλεσμα την αύξηση της υπέρυψωσης στη βηματική απόκριση του συστήματος.

Ένας αναλογικός ελεγκτής αντιδρά άμεσα στην ύπαρξη σφάλματος επιτυγχάνει μείωση του σφάλματος μόνιμης κατάστασης όμως αδυνατεί να το εξαλείψει τελείως αφού για μέγιστη μείωση του σφάλματος χρειάζονται μη ρεαλιστικές υψηλές τιμές κέρδους, αυξάνει την ταχύτητα απόκρισης του συστήματος αλλά δημιουργεί σημαντική υπέρυψωση και ταλαντώσεις με την αύξηση του κέρδους K_p

3.3 Ολοκληρωτικός (I) ελεγκτής

Ο ολοκληρωτικός ελεγκτής (Integral Control) πήρε αυτή την ονομασία επειδή δίνει στην έξοδο του σήμα ανάλογο του ολοκληρώματος του σφάλματος που δέχεται στην είσοδο. Ο ολοκληρωτικός ελεγκτής περιέχει και μια παράμετρο την K_i η οποία έχει μονάδα μέτρησης το sec^{-1} και ονομάζεται συντελεστής ολοκλήρωσης. Ο ολοκληρωτικός ελεγκτής έχει την εξής μορφή:



Σχήμα 3.2: Ο ολοκληρωτικός ελεγκτής

Ο ολοκληρωτικός ελεγκτής εισάγει στο σύστημα ένα πόλο και είναι φανερό από το σχήμα ότι η συνάρτηση μεταφοράς του ελεγκτή είναι:

$$I(s) = \frac{K_i}{s} \quad (3.8)$$

Ο ολοκληρωτικός ελεγκτής για βηματική διέγερση απαιτεί κάποιο χρονικό διάστημα για να πάρει την τιμή της εισόδου δηλαδή η έξοδος του έχει μορφή αναρρίχησης και όταν το σφάλμα μηδενίζεται η έξοδος του ελεγκτή παραμένει σταθερή.

Αν όμως το σύστημα είναι σε συνάρτηση μεταφοράς

$$G(s) = \frac{1}{s+A} \quad (3.9)$$

τότε η συνάρτηση μεταφοράς του συστήματος κλειστού βρόχου με I ελεγκτή είναι:

$$G_C(s) = \frac{\frac{K_i}{s*(s+A)}}{1 + \frac{K_i}{s*(s+A)}} \rightarrow G_C(s) = \frac{K_i}{s*(s+A)+K_i} \rightarrow G_C(s) = \frac{K_i}{s^2+A*s+K_i} \quad (3.10)$$

Επομένως αν διεγείραμε το σύστημα με σήμα εισόδου μοναδιαία βηματική θα βρίσκαμε το σφάλμα μόνιμης κατάστασης από τη σχέση:

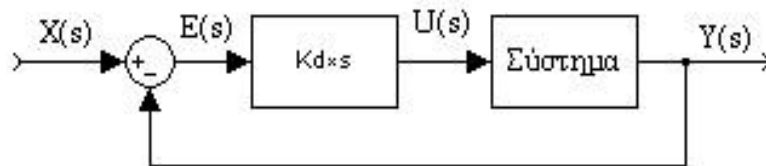
$$e_{ss} = 1 - \lim_{s \rightarrow 0} \left(s * G_C(s) * \frac{1}{s} \right) = 1 - \lim_{s \rightarrow 0} \left(\frac{K_i}{s^2+A*s+K_i} \right) \rightarrow e_{ss} = 1 - \frac{K_i}{K_i} \rightarrow e_{ss} = 0 \quad (3.11)$$

Άρα το σφάλμα μόνιμης κατάστασης μηδενίζεται με τη χρήση ολοκληρωτικού ελεγκτή.

Παρατηρείτε όμως το σύστημα με τη χρήση του ολοκληρωτικού ελεγκτή από το ένα σύστημα μετατράπηκε σε άλλο σύστημα, όπου η τιμή του συντελεστή ολοκλήρωσης είναι ίδια με ένας φυσικής συχνότητας του συστήματος. Όσο αυξάνεται ο συντελεστής ολοκλήρωσης αυξάνεται η φυσική συχνότητα του συστήματος άρα έχουμε ταχύτερη απόκριση του συστήματος και μειώνεται ο συντελεστής απόσβεσης, αφού το γινόμενο $2*\zeta*\omega_n$, παραμένει σταθερό και ίσο με A , με συνέπεια την αύξηση της υπερύψωσης στη βηματική απόκριση του συστήματος. Ένας ολοκληρωτικός ελεγκτής εξαλείφει το σφάλμα μόνιμης κατάστασης για σταθερή είσοδο αναφοράς και επιπλέον αντιμετωπίζει με επιτυχία αποκλίσεις από πιθανό διαταραχές του συστήματος.

3.4 Διαφορικός (D) ελεγκτής

Ο διαφορικός ελεγκτής (Differential Control) όταν το σήμα του σφάλματος στην είσοδο του έχει τη μορφή βηματικής διέγερσης τότε η έξοδος του ελεγκτή είναι η κρουστική συνάρτηση με θεωρητικά άπειρο πλάτος για $t=0$. Όταν το σφάλμα είναι σταθερό τότε η έξοδος του διαφορικού ελεγκτή είναι μηδέν.



Σχήμα 3.3: Ο διαφορικός ελεγκτής

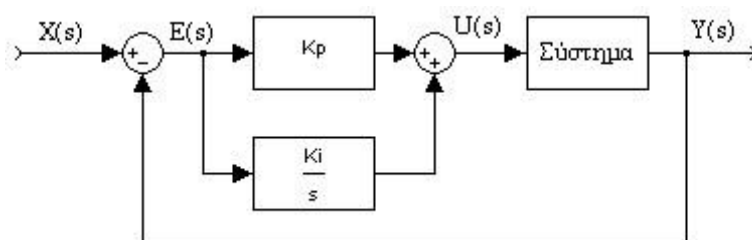
Ο διαφορικός ελεγκτής εισάγει στο σύστημα ένα μηδενικό και η συνάρτηση μεταφοράς του ελεγκτή είναι:

$$D(s) = K_d * s \quad (3.12)$$

Ο διαφορικός ελεγκτής περιορίζει το σφάλμα στη μόνιμη κατάσταση κατά το στάδιο της μεταβατικής απόκρισης των συστημάτων όμως στη πράξη δεν χρησιμοποιείται ποτέ αποκλειστικά διαφορικός ελεγκτής.

3.5 Αναλογικός– ολοκληρωτικός (PI) ελεγκτής

Ο αναλογικός – ολοκληρωτικός ελεγκτής συνδυάζει αθροιστικά τον αναλογικό και τον ολοκληρωτικό έλεγχο. Η συνδεσμολογία του αναλογικού – ολοκληρωτικού ελέγχου είναι η συνδεσμολογία του παρακάτω σχήματος.



Σχήμα 3.4: Ο αναλογικός – ολοκληρωτικός ελεγκτής

Ο αναλογικός – ολοκληρωτικός ελεγκτής εισάγει ένα πόλο στο μηδέν και ένα μηδενικό στο $-\frac{K_i}{K_p}$ συμπέρασμα που προκύπτει από τη συνάρτηση μεταφοράς του ελεγκτή.

$$PI(s) = K_p + \frac{K_i}{s} = \frac{s \cdot K_p + K_i}{s} \rightarrow PI(s) = K_p * \frac{s + \frac{K_i}{K_p}}{s} \quad (3.13)$$

Η χρήση του όρου ολοκλήρωσης αποσκοπεί στην εξάλειψη του σφάλματος στη μόνιμη κατάσταση και μεταβάλλει το DC-κέρδος του συστήματος ενώ ο αναλογικός όρος αυξάνει τη ταχύτητα απόκρισης και βελτιώνει την ευστάθεια του συστήματος.

Αν το σύστημα που ελέγχουμε είναι πρώτης τάξης με συνάρτηση μεταφοράς

$$G(s) = \frac{1}{s+A} \quad (3.14)$$

τότε η συνάρτηση μεταφοράς του συστήματος κλειστού βρόχου με PI ελεγκτή είναι:

$$G_{C(s)} = \frac{\frac{s * K_p + K_i}{s} * \frac{1}{s + A}}{1 + \frac{s * K_p + K_i}{s} * \frac{1}{s + A}} \rightarrow G_{C(s)} = \frac{\frac{s * K_p + K_i}{s * (s + A)}}{\frac{s * (s + A) + s * K_p + K_i}{s * (s + A)}} \rightarrow$$

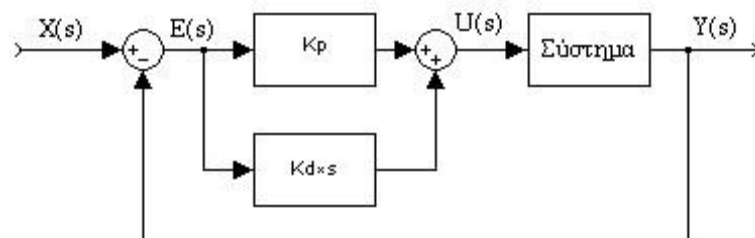
$$\rightarrow G_{C(s)} = \frac{s * K_p + K_i}{s^2 + A * s + s * K_p + K_i} \rightarrow G_{C(s)} = \frac{s * K_p + K_i}{s^2 + (A + K_p) * s + K_i} \quad (3.15)$$

Το σύστημα αλλάζει και γίνεται με $\omega_n = K_i$ και $2 * \zeta * \omega_n = A + K_p$.

Άρα όταν χρησιμοποιούμε τον PI ελεγκτή με την αύξηση του ολοκληρωματικού όρου K_i αυξάνεται η φυσική συχνότητα ω_n του συστήματος και με την αύξηση και με την αύξηση του αναλογικού συντελεστή K_p αυξάνεται το γινόμενο $2 * \zeta * \omega_n$, αλλά χωρίς να μεταβάλλεται η ω_n οπότε αυξάνεται ο συντελεστής απόσβεσης του συστήματος με αποτέλεσμα τη μείωση της υπερύψωσης. Η ακριβής επίδραση του PI ελεγκτή στην μεταβατική απόκριση του συστήματος εξαρτάται από το μηδενισμό $s = -\frac{K_i}{K_p}$ που εισάγει ο ελεγκτής στο σύστημα. Τέλος με την εισαγωγή του όρου ολοκλήρωσης το σύστημα για βηματική είσοδο παρουσιάζει μηδενικό σφάλμα μόνιμης κατάστασης.

3.6 Αναλογικός– διαφορικός (PD) ελεγκτής

Ο αναλογικός – διαφορικός ελεγκτής συνδυάζει αθροιστικά τον αναλογικό και το διαφορικό έλεγχο. Η συνδεσμολογία του αναλογικού– διαφορικού ελέγχου είναι η συνδεσμολογία του παρακάτω σχήματος.



Σχήμα 3.5: Ο αναλογικός – διαφορικός ελεγκτής

Ο αναλογικός – διαφορικός ελεγκτής εισάγει ένα πόλο στο $\frac{K_p}{K_d}$ συμπέρασμα που προκύπτει από τη συνάρτηση μεταφοράς του ελεγκτή:

$$PD_{(s)} = K_p + s * K_d \rightarrow PD_{(s)} = s + \frac{K_p}{K_d} \quad (3.16)$$

Ο PD ελεγκτής αυξάνει την απόσβεση του συστήματος και περιορίζει την υπερύψωση της, δεν μειώνει άμεσα το σφάλμα μόνιμης κατάστασης. Η χρήση του διαφορικού όρου προκαλεί την αυξημένη απόσβεση του συστήματος που επιτρέπει την αύξηση του αναλογικού κέρδους που έχει ως αποτέλεσμα τη βελτίωση του σφάλματος μόνιμης κατάστασης και την αύξηση της ταχύτητας απόκρισης του συστήματος. Στην πράξη ο PD έλεγχος εξαιτίας του διαφορικού όρου ενισχύει το θόρυβο ο οποίος μπορεί να εμφανιστεί στην απόκριση του συστήματος. Είναι πιθανό να δημιουργηθούν προβλήματα στην περίπτωση που υπάρχουν απότομες αλλαγές στο σήμα αναφοράς γιατί έχουν ως συνέπεια ο διαφορικός όρος να παράγει έξοδο κρουστικής μορφής.

Αν το σύστημα είναι σε συνάρτηση μεταφοράς

$$G(s) = \frac{1}{s^2 + A*s + B} \quad (3.16)$$

τότε η συνάρτηση μεταφοράς του συστήματος κλειστού βρόχου με PD ελεγκτή είναι:

$$\begin{aligned} G_C(s) &= \frac{(K_p + s * K_d) * \frac{1}{s^2 + A * s + B}}{1 + (K_p + s * K_d) * \frac{1}{s^2 + A * s + B}} \rightarrow \\ &\rightarrow G_C(s) = \frac{\frac{(K_p + s * K_d)}{s^2 + A * s + B}}{\frac{s^2 + A * s + B + K_p + s * K_d}{s^2 + A * s + B}} \rightarrow \\ &\rightarrow G_C(s) = \frac{K_p + s * K_d}{s^2 + (A + K_d) * s + (B + K_p)} \end{aligned} \quad (3.17)$$

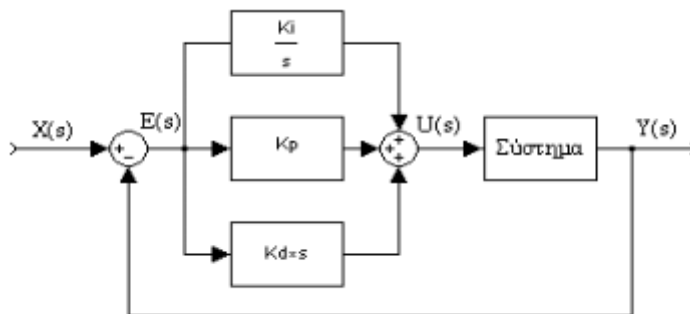
Το νέο σύστημα έχει τα εξής χαρακτηριστικά:

$$\omega_n = B + K_p \text{ και } 2 * \zeta * \omega_n = A + K_d \quad (3.18)$$

Άρα η αύξηση του αναλογικού κέρδους K_d οδηγεί σε αύξηση της φυσικής συχνότητας του συστήματος επομένως και σε ταχύτερη απόκριση του συστήματος. Η αύξηση του διαφορικού συντελεστή K_d οδηγεί σε αύξηση του γινομένου $2 * \zeta * \omega_n$ με αμετάβλητη τη φυσική συχνότητα άρα αυξάνεται ο συντελεστής απόσβεσης του συστήματος που οδηγεί στη μείωση της υπερύψωσης της απόκρισης του συστήματος. Τέλος το σφάλμα μόνιμης κατάστασης μειώνεται όσο αυξάνεται το αναλογικό κέρδος ενώ δεν επηρεάζεται από τη μεταβολή του διαφορικού συντελεστή.

3.7 Αναλογικός – ολοκληρωτικός – διαφορικός (PID) ελεγκτής

Ο πιο διαδεδομένος ελεγκτής είναι ο αναλογικός – ολοκληρωτικός – διαφορικός ελεγκτής (PID) και οφείλεται κατά ένα μέρος στην πολύ καλή συμπεριφορά του έναντι μιας αρκετά ευρείας κλίμακας λειτουργικών συνθηκών και κατά ένα άλλο μέρος στη σχετική του απλότητα η οποία επιτρέπει έναν εύκολο και αξιόπιστο χειρισμό. Ο PID έλεγχος συνδυάζει αθροιστικά την επίδραση των τριών βασικών όρων ελέγχου. Η συνδεσμολογία του PID ελέγχου παρουσιάζεται στο παρακάτω σχήμα.



Σχήμα 3.6: Ο αναλογικός – ολοκληρωτικός – διαφορικός ελεγκτής

Ο PID ελεγκτής εισάγει στο υπό έλεγχο σύστημα ανοιχτού βρόχου ένα πόλο στο μηδέν και δύο μηδενικά. Η συνάρτηση μεταφοράς του ελεγκτή είναι η εξής:

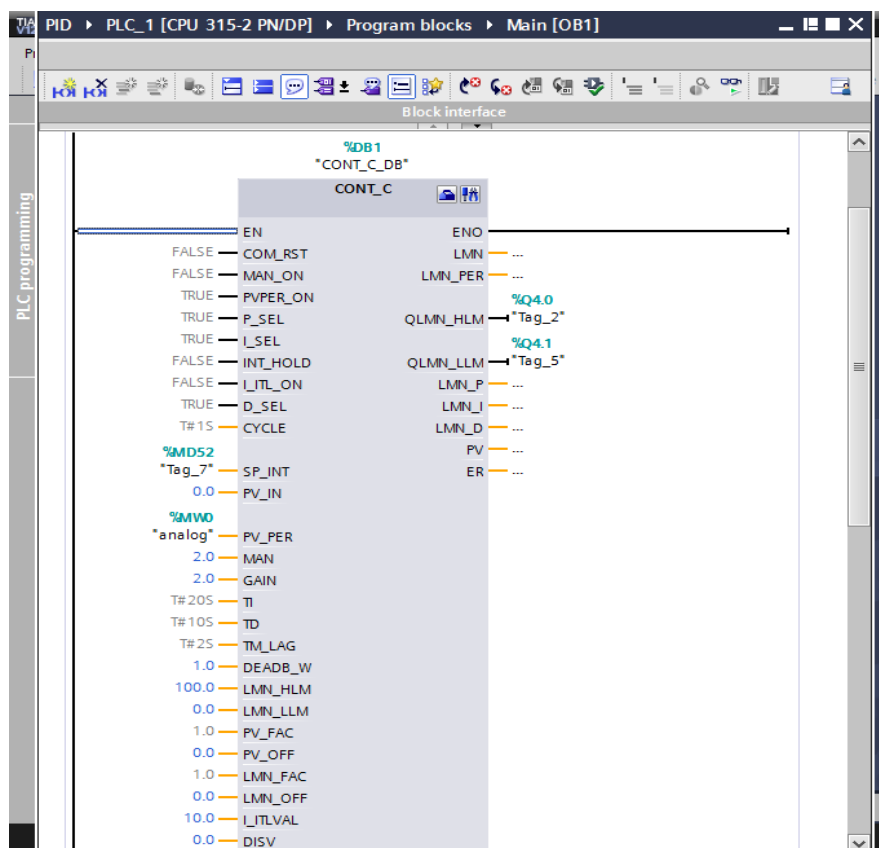
$$PID(s) = K_p + s * K_d + \frac{K_i}{s} \rightarrow PID(s) = \frac{K_d * s^2 + K_p * s + K_i}{s} \quad (3.19)$$

Πολλές διεργασίες βιομηχανικού τύπου ελέγχονται με την βοήθεια μονάδων ελεγκτών PID. Ο PID ελεγκτής έχει την καλύτερη συνολικά απόδοση καθώς το σύστημα εμφανίζει ταχεία απόκριση, μηδενική υπερύψωση και μηδενικό σφάλμα μόνιμης κατάστασης.

Για να χρησιμοποιηθεί ένας τέτοιου είδους ελεγκτής θα πρέπει να σχεδιαστεί με κατάλληλη επιλογή των τριών παραμέτρων του αναλογικού κέρδους, του ολοκληρωτικού συντελεστή και του διαφορικού συντελεστή έτσι ώστε το σύστημα κλειστού βρόχου να έχει τα πλεονεκτήματα όλων των ελεγκτών που συνδυάζονται στον PID ελεγκτή. Στη ουσία το πιο δύσκολο στη σχεδίαση ενός PID ελεγκτή είναι να δοθούν οι κατάλληλες τιμές στα K_p , K_d και K_i έτσι ώστε το ένα μην επιδρά αρνητικά στη λειτουργία του άλλου.

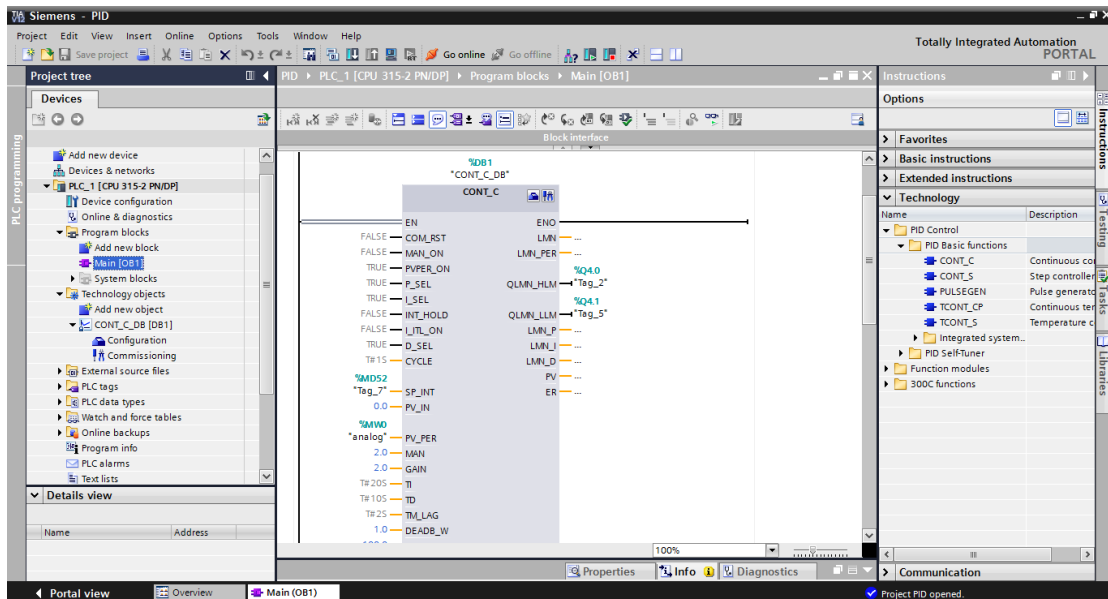
3.8 Ελεγκτής PID Continuous Controller(CONT_C)

Για τις ανάγκες της εφαρμογής αλλά και την παρουσίαση των δυνατοτήτων των PLC στο κομμάτι όπου πραγματοποιείται η παστερίωση του γάλακτος θα γίνει χρήση ελεγκτή PID για τον έλεγχο της θερμοκρασίας στον φούρνο. Η επιλογή του ελεγκτή έγινε με βάση τις απαιτήσεις της εφαρμογής όπου ο χρήστης θα πρέπει να έχει την δυνατότητα για συνεχή έλεγχο της θερμοκρασίας σε κάθε φάση. Επιλέχθηκε ο ελεγκτής CONT_C όπου ο χρήστης θα ελέγχει τις εισόδους του PID δίνοντας του διάφορες τιμές στο Set Point καθώς επίσης και στις υπόλοιπες μεταβλητές του PID ανάλογα με τις απαιτήσεις της εφαρμογής (Σχήμα 3.7).



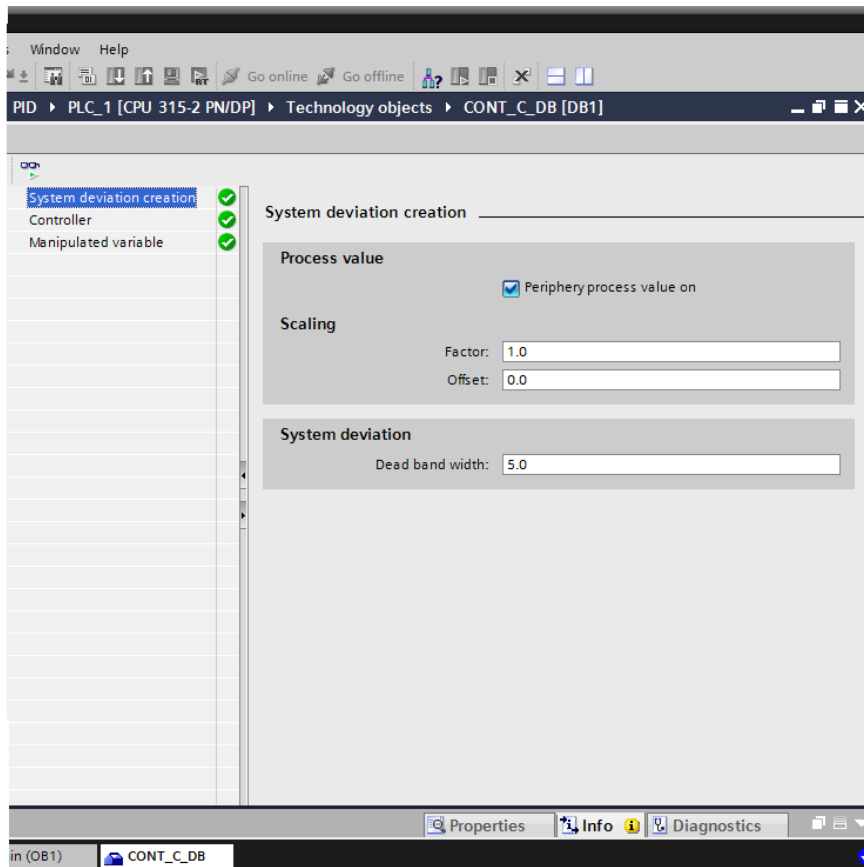
Σχήμα 3.7: Ελεγκτής PID CONT_C

Στην παραπάνω εικόνα παρουσιάζεται η μορφή του PID στο πρόγραμμα. και φαίνονται όλες οι εισόδους που μπορεί ο χρήστης να ρυθμίσει. Επίσης το πρόγραμμα διαθέτει εγχειρίδιο με την λειτουργία της κάθε εισόδου και τον τρόπο ρύθμισης της.

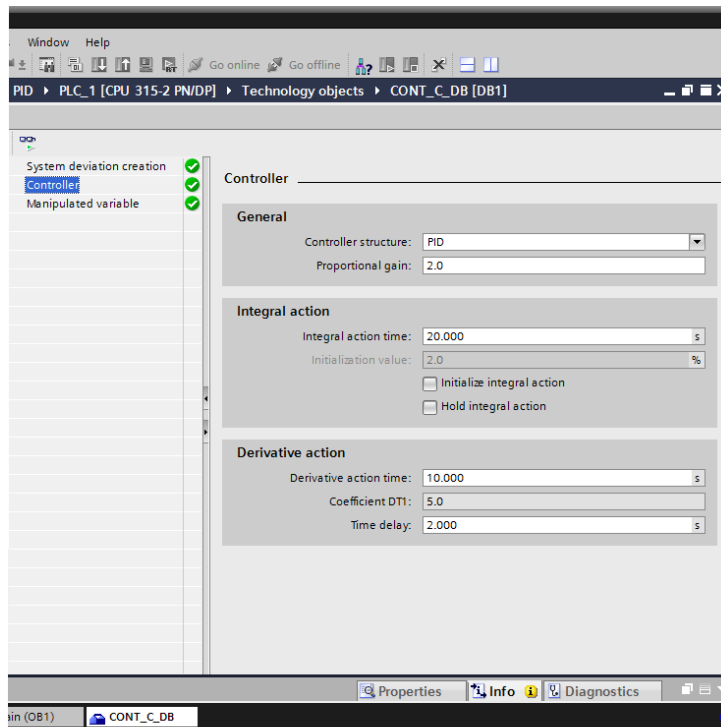


Σχήμα 3.8: Επιφάνεια εργασίας PID

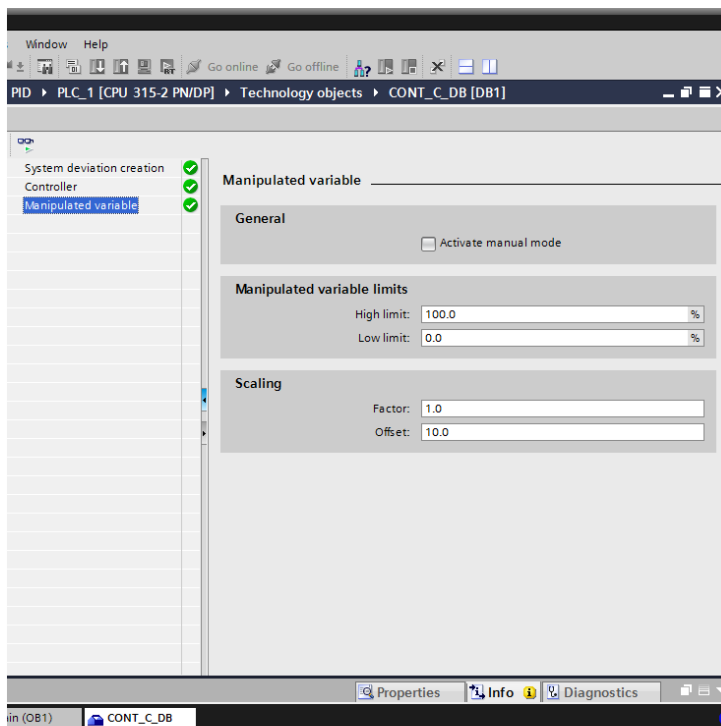
Για την λειτουργία του ελεγκτή ο χρήστης πρέπει να ορίσει Set Point στην εντολή SP_INT με την μορφή "MD" και την είσοδο του PID στην εντολή PV_PER με μορφή "MW". Στη πλευρά όπου βρίσκεται Project Tree φαίνονται οι επιλογές που έχει ο ελεγκτής (Σχήμα 3.8). Στην επιλογή Configuration ο χρήστης κάνει γενικές ρυθμίσεις όπως τον ορισμό του ελεγκτή καθώς και τον τρόπο χειρισμού (Σχήματα 3.9 – 3.11).



Σχήμα 3.9: Ρύθμιση ελεγκτή 1

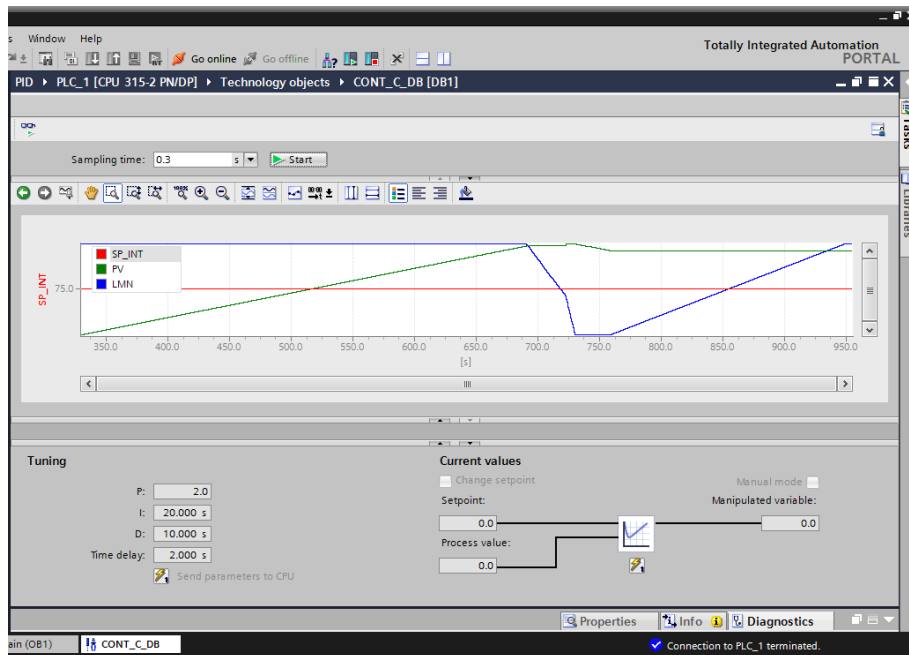


Σχήμα 3.10: Ρύθμιση ελεγκτή 2



Σχήμα 3.11: Ρύθμιση ελεγκτή 3

Αφού γίνουν οι παραπάνω ρυθμίσεις ο χρήστης θα πρέπει να μεταβεί στην επιλογή Commissioning ώστε να δει την λειτουργία και την συμπεριφορά του ελεγκτή σε διάγραμμα (Σχήματα 3.9 - 3.11). Για να ξεκινήσει η διαδικασία θα πρέπει να γίνει επιλογή του Start Simulation του OB1 και έπειτα η επιλογή του Monitor All στο Commissioning.



Σχήμα 3.12: Συμπεριφορά ελεγκτή

Στην παραπάνω γραφική απεικόνιση παρουσιάζεται η συμπεριφορά του ελεγκτή (Σχήμα 3.12). Για την παρουσίαση έχουν οριστεί χρώματα για την κατανόηση του κάθε μεγέθους. Με κόκκινο αναπαρίστανται το Set Point, με πράσινο χρώμα αναπαρίστανται το PV που είναι η είσοδος του ελεγκτή και με μπλε χρώμα το LMN που είναι η έξοδος του ελεγκτή. Η γραφική απεικόνιση είναι από μια διαδικασία που έχει οριστεί Set Point:75, MAN:1, GAIN:1, DEAD_W:1. Αυτές οι τιμές είναι οι πιο κρίσιμες όπου επηρεάζουν την έξοδο του PID. Τα όρια της διαδικασίας είναι από 0 έως 100. Παρατηρώντας το διάγραμμα η τιμή της εισόδου αυξάνεται σταδιακά ώστε να παρουσιαστούν καλύτερα οι μεταβολές της εξόδου. Καθώς η είσοδος αυξάνεται και φθάσει σε κάποιο όριο που έχει ορίσει ο χρήστης πάνω από το Set Point τότε ξεκινά η έξοδος του ελεγκτή να μεταβάλλεται. Όσο η τιμή της εισόδου είναι σε αυτό το όριο, η έξοδος του ελεγκτή μειώνεται μέχρι να φθάσει στην ελάχιστη τιμή που έχει ορίσει ο χρήστης. Αντίθετα όταν η τιμή της εισόδου πέσει κάτω από ένα όριο του Set Point τότε αρχίζει να αυξάνεται η τιμή της εξόδου μέχρι την μέγιστη τιμή που έχει οριστεί.

Για την παραμετροποίηση του PID μέσα από το πρόγραμμα του χρήστη ο ελεγκτής CONT_C διαθέτει μια σειρά από εισόδους και εξόδους που περιγράφονται στους πίνακες 3.1 και 3.2.

| Είσοδος | Τύπος Δεδομ. | Προκαθ. Τιμή | Description |
|----------|--------------|--------------|--|
| COM_RST | BOOL | FALSE | Ενεργοποίηση της διαδικασίας αρχικοποίησης κάθε φορά που ενεργοποιείται η είσοδος Restart |
| MAN_ON | BOOL | TRUE | Αν η είσοδος "Enable manual mode" είναι ενεργοποιημένη ο βρόχος ελέγχου διακόπτεται και μια τιμή που δίνει ο χρήστης ορίζεται ή ως τιμή χειρισμού. |
| PVPER_ON | BOOL | TRUE | Αν η τιμή της διαδικασίας είναι να διαβαστεί από τις I/O, η είσοδος PV_PER πρέπει να διασυνδεθεί με τις I/O και η είσοδος "Enable process value I/Os" θα πρέπει να ενεργοποιηθεί |
| P_SEL | BOOL | TRUE | Οι ενέργειες του PID μπορούν να ενεργοποιηθούν και να απενεργοποιηθούν ξεχωριστά στον αλγόριθμο PID. Η P-action ενεργοποιείται όταν η είσοδος "Enable P-action" έχει ενεργοποιηθεί |

Πίνακας 3.1: Οι εισοδοί του ελεγκτή CONT_C

| Είσοδος | Τύπος Δεδομ. | Προκαθ. Τιμή | Description |
|----------|--------------|--------------|---|
| I_SEL | BOOL | TRUE | Οι ενέργειες του PID μπορούν να ενεργοποιηθούν και να απενεργοποιηθούν ξεχωριστά στον αλγόριθμο PID. Η I-action ενεργοποιείται όταν η είσοδος "Enable I-action" έχει ενεργοποιηθεί |
| INT_HOLD | BOOL | FALSE | Η έξοδος της ενέργειας ολοκλήρωσης μπορεί να παγώσει. Για αυτή την λειτουργία η είσοδος " I-action hold" πρέπει να ενεργοποιηθεί |
| I_ITL_ON | BOOL | FALSE | Η έξοδος του ενέργειας ολοκλήρωσης μπορεί να οριστεί στη I_ITLVAL είσοδο. Για αυτή την λειτουργία η είσοδος "Set I- action" πρέπει να ενεργοποιηθεί |
| D_SEL | BOOL | FALSE | Οι ενέργειες του PID μπορούν να ενεργοποιηθούν και να απενεργοποιηθούν ξεχωριστά στον αλγόριθμο PID. Η D-action ενεργοποιείται όταν η είσοδος "Enable D-action" έχει ενεργοποιηθεί |
| CYCLE | TIME | T#1s | Ο χρόνος μεταξύ των κλήσεων των block πρέπει να είναι σταθερός . Η είσοδος "Sampling time" προσδιορίζει το χρονικό διάστημα μεταξύ των κλήσεων των block . CYCLE >=1ms |
| SP_INT | REAL | 0.0 | Η είσοδος "Internal setpoint " χρησιμοποιείται για να καθορίσει μια επιθυμητή τιμή. Επιτρεπτές τιμές είναι -100 έως 100 % ή μια φυσική μεταβλητή |
| PV_IN | REAL | 0.0 | Στη "Process value input " μπορεί να οριστούν παράμετροι μιας commissioning τιμής ή μπορεί να διασυνδεθεί η τιμή μιας εξωτερικής διαδικασίας ως αριθμός κινητής υποδιαστολής. Επιτρεπτές τιμές είναι -100 έως 100 % ή μια φυσική μεταβλητή 1) |
| PV_PER | WORD | W#16#0000 | Η τιμή της διαδικασίας σε μορφή I/O είναι διασυνδεδεμένη με τον ελεγκτή στην είσοδο "Process value I/O". |
| MAN | REAL | 0.0 | Η είσοδος "Manual value" χρησιμοποιείται για να οριστεί μία χειροκίνητη τιμή χρησιμοποιώντας τις λειτουργίες της διεπαφής χειριστή Επιτρεπτές τιμές είναι -100 έως 100 % ή μια φυσική μεταβλητή 2) . |
| GAIN | REAL | 2.0 | Η είσοδος "Proportional gain" καθορίζει την ενίσχυση ελεγκτή. |
| TI | TIME | T#20s | Η είσοδος "Integration time" καθορίζει το χρόνο απόκρισης της ενέργειας ολοκλήρωσης. TI >=CYCLE |
| TD | TIME | T#10s | Η είσοδος "Derivative action time" καθορίζει το χρόνο απόκρισης της ενέργειας παραγωγίσισης. TD >= CYCLE |
| TIME_LAG | TIME | T#2s | Χρόνος υστέρησης της D-action. Ο αλγόριθμος της D – action περιέχει μια καθυστέρηση για τις παραμέτρους τις οποίες μπορεί να ανατεθούν στην είσοδο "Time lag of the D-action". TM_LAG >= CYCLE/2 |
| DEADB_W | REAL | 0.0 | Μια νεκρή ζώνη εφαρμόζεται στην απόκλιση του συστήματος. Η είσοδος "Dead band width" καθορίζει το μέγεθος της νεκρής ζώνης . DEADB_W >= 0.0 (%) ή μια φυσική τιμή . |
| LMN_HLM | REAL | 100.0 | Η χειριζόμενη τιμή είναι μεταξύ ενός υψηλού και ενός χαμηλού ορίου. Η είσοδος "High limit of manipulated value" καθορίζει το υψηλό όριο. Οι επιτρεπτές τιμές της LMN_HLM είναι μέχρι το 100% ή μια φυσική τιμή. |
| LMN_LLM | REAL | 0.0 | Η χειριζόμενη τιμή είναι μεταξύ ενός υψηλού και ενός χαμηλού ορίου. Η είσοδος "Low limit of manipulated value" καθορίζει το χαμηλό όριο. Επιτρεπτές τιμές είναι |

Πίνακας 3.1: Οι είσοδοι του ελεγκτή CONT_C (Συνέχεια)

| Είσοδος | Τύπος Δεδομ. | Προκαθ. Τιμή | Description |
|----------|--------------|--------------|--|
| | | | της LMN_LLM κυμαίνονται από -100 έως LMN_HLM (%) ή μια φυσική τιμή . |
| PV_FAC | REAL | 1.0 | Η είσοδος "Process value factor" πολλαπλασιάζεται με την Process Value. Η είσοδος αυτή χρησιμοποιείται για να διαμορφωθεί η κατάλληλη κλίμακα τιμών της Process Value. |
| PV_OFF | REAL | 0.0 | Η είσοδος "Process value offset" προστίθεται στην Process value . Η είσοδος αυτή χρησιμοποιείται για να διαμορφωθεί η κατάλληλη κλίμακα τιμών της Process Value. |
| LMN_FAC | REAL | 1.0 | Η είσοδος "Manipulated value factor" πολλαπλασιάζεται με την manipulated value. Η είσοδος αυτή χρησιμοποιείται για να διαμορφωθεί η κατάλληλη κλίμακα τιμών της manipulated value. |
| LMN_OFF | REAL | 0.0 | Η είσοδος "Manipulated value offset" προστίθεται στην manipulated value. Η είσοδος αυτή χρησιμοποιείται για να διαμορφωθεί η κατάλληλη κλίμακα τιμών της manipulated value. |
| I_ITLVAL | REAL | 0.0 | Η έξοδος της διαδικασίας ολοκλήρωσης μπορεί να ρυθμιστεί στην είσοδο I_ITL_ON. Η τιμή αρχικοποίησης εφαρμόζεται στην είσοδο "Initialization value of the I-action". Επιτρεπόμενες τιμές είναι -100,0 με 100,0 (%) ή μια φυσική μεταβλητή |
| DISV | REAL | 0.0 | Για feedforward έλεγχο, η μεταβλητή διαταραχής είναι διασυνδεδεμένη με την είσοδο "Disturbance variable". Επιτρεπόμενες τιμές είναι -100,0 με 100,0 (%) ή μια φυσική μεταβλητή. |

Πίνακας 3.1: Οι εισοδοί του ελεγκτή CONT_C (Συνέχεια)

| Έξοδος | Τύπος Δεδομ. | Προκαθ. Τιμή | Description |
|----------|--------------|--------------|--|
| LMN | REAL | 0.0 | Το αποτέλεσμα της Manipulated value είναι έξοδος σε μορφή αριθμών κινητής υποδιαστολής |
| LMN_PER | WORD | W#16#0000 | Η manipulated value σε μορφή I/O είναι διασυνδεδεμένη στην είσοδο "Manipulated value I/O" με τον ελεγκτή . |
| QLMN_HLM | BOOL | FALSE | Η manipulated value είναι πάντα περιορισμένη σε ένα υψηλό όριο και χαμηλό όριο . Η έξοδος «High limit of manipulated value reached» δηλώνει ότι το υψηλό όριο έχει επιτευχθεί. |
| QLMN_LLM | BOOL | FALSE | Η manipulated value είναι πάντα περιορισμένη σε ένα υψηλό όριο και χαμηλό όριο . Η έξοδος «Low limit of manipulated value reached» δηλώνει ότι το χαμηλό όριο έχει επιτευχθεί |
| LMN_P | REAL | 0.0 | Η έξοδος "P-action " περιέχει το αποτέλεσμα της διαδικασίας proportional στην manipulated value |
| LMN_I | REAL | 0.0 | Η έξοδος "I-action" περιέχει το αποτέλεσμα της διαδικασίας integral στην manipulated value |
| LMN_D | REAL | 0.0 | Η έξοδος "D-action" περιέχει το αποτέλεσμα της διαδικασίας differential στην manipulated value |
| PV | REAL | 0.0 | Η έξοδος "Process Value" περιέχει το τελικό αποτέλεσμα της διαδικασίας. |
| ER | REAL | 0.0 | Η έξοδος "Error signal" περιέχει την απόκλιση του συστήματος |

Πίνακας 3.2: Οι έξοδοι του ελεγκτή CONT_C

Κεφάλαιο 4

Περιγραφή του σταθμού παστερίωσης γάλακτος

4.1 Περιγραφή σεναρίων

Η εφαρμογή για την παστερίωση γάλακτος θα περιλαμβάνει τέσσερις βασικές λειτουργίες. Η πρώτη λειτουργία είναι η κανονική διαδικασία χωρίς σφάλματα. Η δεύτερη λειτουργία περιλαμβάνει το πρώτο σφάλμα που αντιστοιχεί στην πρώτη φάση της διαδικασίας και αναφέρεται στην ρύθμιση της θερμοκρασίας εντός ορίων. Στη συνέχεια η τρίτη λειτουργία είναι το δεύτερο σφάλμα όπου αντιστοιχεί στην δεύτερη φάση, της παστερίωσης, όταν η θερμοκρασία πέσει κάτω από ένα συγκεκριμένο όριο. Τέλος, η τέταρτη λειτουργία αντιστοιχεί σε σφάλμα της μεταφορικής ταινίας στην πρώτη και στην δεύτερη φάση της όλης διαδικασίας.

4.1.1 Κανονική λειτουργία χωρίς σφάλματα

Με την πίεση του κουμπιού Start το προϊόν (μπουκάλι) θα ξεκινάει από την αρχή της μεταφορικής ταινίας και θα έχει πορεία προς το αισθητήριο LS_1. Όταν φθάσει σε εκείνο το σημείο θα ενεργοποιείται ο φούρνος και θα πρέπει να ρυθμιστεί η κατάλληλη θερμοκρασία για τον φούρνο που θα γίνει η παστερίωση. Εφόσον ο φούρνος πάρει την κατάλληλη θερμοκρασία το προϊόν παίρνει εντολή να ξεκινήσει με κατεύθυνση προς τον αισθητήριο LS_2. Μόλις φθάσει στο αισθητήριο LS_2 το προϊόν σταματάει και αρχίζει η παστερίωση. Με την ολοκλήρωση της παστερίωσης μετά από ορισμένο χρόνο το προϊόν παίρνει εντολή να ξεκινήσει με κατεύθυνση το αισθητήριο LS_3. Μόλις το προϊόν φθάσει στο αισθητήριο LS_3 έχει ολοκληρωθεί η διαδικασία. Τέλος, ο χρήστης θα μπορεί να τερματίσει το πρόγραμμα με την πίεση του κουμπιού Stop ή να δώσει εντολή με την πίεση ενός κουμπιού για νέα διαδικασία.

4.1.2 Λειτουργία με σφάλμα στη ρύθμιση της θερμοκρασίας εντός ορίων

Ο χειριστής πατάει το κουμπί Start, το προϊόν παίρνει εντολή να ξεκινήσει και από την αρχή της μεταφορικής ταινίας κατευθύνεται προς το αισθητήριο LS_1. Όταν θα φθάσει στο αισθητήριο LS_1 ενεργοποιείται ο φούρνος και παράλληλα θα πρέπει να ρυθμιστεί η τιμή της θερμοκρασίας εντός ορίων και εντός κάποιου χρόνου. Στην περίπτωση που δεν ικανοποιηθούν οι προηγούμενες απαιτήσεις τότε ενεργοποιείται το Alarm 1 και εμφανίζεται μήνυμα με κείμενο που ενημερώνει για το τύπο του σφάλματος. Το επόμενο βήμα είναι να γίνει η αναγνώριση του σφάλματος από τον χειριστή. Μόλις γίνει η αναγνώριση με την πίεση κατάλληλου κουμπιού το προϊόν παίρνει εντολή και αρχίζει να κατευθύνεται στο τέλος της μεταφορικής ταινίας προς το αισθητήριο LS_3. Μόλις φθάσει στο αισθητήριο LS_3 ο χειριστής μπορεί είτε να τερματίσει την διαδικασία με Stop είτε να πατήσει το κουμπί για νέα διαδικασία.

4.1.3 Λειτουργία με σφάλμα στη θερμοκρασία παστερίωσης κάτω από ένα όριο

Ο χειριστής πατώντας το κουμπί Start, δίνει στο προϊόν εντολή να ξεκινήσει και από την αρχή της μεταφορικής ταινίας κατευθύνεται προς το αισθητήριο LS_1. Όταν θα φθάσει στο αισθητήριο LS_1 γίνεται ενεργοποίηση του φούρνου και παράλληλα ρυθμίζεται κατάλληλα η τιμή της θερμοκρασίας παστερίωσης. Εφόσον έχουν ικανοποιηθεί οι προηγούμενες απαιτήσεις, το προϊόν έχει πάρει εντολή για να κατευθύνεται προς το αισθητήριο LS_2 που βρίσκεται στο σημείο του φούρνου. Στο αισθητήριο LS_2 το προϊόν σταματάει και αρχίζει η διαδικασία της παστερίωσης. Σ αυτή την φάση αν η θερμοκρασία πέσει κάτω από ένα όριο για συγκεκριμένο χρόνο και δεν επανέλθει άμεσα τότε ενεργοποιείται το Alarm 2 και

εμφανίζεται μήνυμα με κείμενο που ενημερώνει για το τύπο του σφάλματος. Το επόμενο βήμα είναι να γίνει η αναγνώριση του σφάλματος από τον χειριστή. Μόλις γίνει η αναγνώριση με την πίεση κατάλληλου κουμπιού το προϊόν παίρνει εντολή και αρχίζει να κατευθύνεται στο τέλος της μεταφορικής ταινίας προς το αισθητήριο LS_3. Με τον τερματισμό του προϊόντος στο αισθητήριο LS_3 ο χειριστής μπορεί είτε να τερματίσει την διαδικασία με Stop είτε να πατήσει το κουμπί για νέα διαδικασία.

4.1.4 Λειτουργία με σφάλμα στη μεταφορική ταινία

Ο χειριστής πατώντας το κουμπί Start, δίνει στο προϊόν εντολή να ξεκινήσει και από την αρχή της μεταφορικής ταινίας κατευθύνεται προς το αισθητήριο LS_1. Στα μέσα της διαδρομής παρουσιάζεται σφάλμα και το προϊόν σταματάει να κινείται. Σ' αυτή την φάση ενεργοποιείται το Alarm 3 και εμφανίζεται μήνυμα με κείμενο που ενημερώνει για το τύπο του σφάλματος. Το επόμενο βήμα είναι να γίνει η αναγνώριση του σφάλματος από τον χειριστή που για αυτό το σφάλμα είναι η πίεση του κουμπιού Stop που τερματίζει όλες τις διαδικασίες.

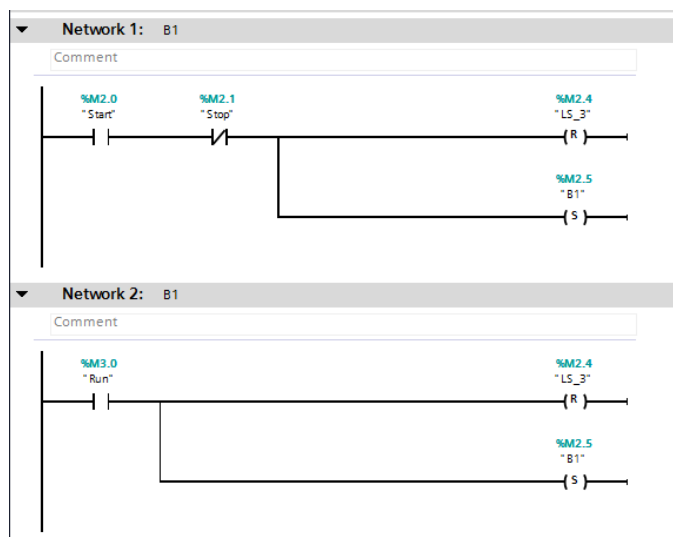
Επίσης σφάλμα στη μεταφορική ταινία έχει οριστεί και στην διαδρομή για την παστερίωση από το αισθητήριο LS_1 προς το LS_2. Ο χειριστής πατώντας το κουμπί Start, δίνει στο προϊόν εντολή να ξεκινήσει και από την αρχή της μεταφορικής ταινίας κατευθύνεται προς το αισθητήριο LS_1. Όταν θα φθάσει στο αισθητήριο LS_1 γίνεται ενεργοποίηση του φούρνου και παράλληλα ρυθμίζεται κατάλληλα η τιμή της θερμοκρασίας παστερίωσης. Εφόσον έχουν ικανοποιηθεί οι προηγούμενες απαιτήσεις, το προϊόν έχει πάρει εντολή για να κατευθύνεται προς το αισθητήριο LS_2 που βρίσκεται στο σημείο του φούρνου. Κατά την διαδρομή αυτή το προϊόν σταματάει να κινείται και ενεργοποιείται το Alarm 3 και εμφανίζεται μήνυμα με κείμενο που ενημερώνει για τον τύπο του σφάλματος. Τέλος, γίνεται αναγνώριση του σφάλματος από τον χειριστή που για αυτή την περίπτωση σφάλματος είναι η πίεση του κουμπιού Stop και τερματίζει όλες τις διαδικασίες.

4.2 Υλοποίηση προγράμματος

Στον κώδικα για την υλοποίηση του προγράμματος στο PLC έγινε χρήση της γλώσσας προγραμματισμού ladder. Για την υλοποίηση του προγράμματος χρησιμοποιήθηκαν εντολές τύπου set (S), reset (R), μπλοκ διαγράμματα και χρονικά. Επίσης με την χρήση των παραπάνω εντολών βελτιώθηκε η παρουσίαση των γραφικών στοιχείων στην οθόνη HMI. Τα ονόματα των εντολών που έχουν οριστεί σχετίζονται με την κίνηση του μπουκαλιού, με τα σφάλματα καθώς και με την ενεργοποίηση, απενεργοποίηση λειτουργιών.

Με B1 έχει οριστεί η κίνηση του μπουκαλιού από την αρχή της μεταφορικής ταινίας έως το αισθητήριο LS_1. Αντίστοιχα, B2 η κίνηση του μπουκαλιού από το αισθητήριο LS_1 έως LS_2 και B3 η κίνηση από το αισθητήριο LS_2 έως LS_3.

Λόγω ότι το PLC διαβάζει κυκλικά το πρόγραμμα και υπάρχει περίπτωση κάποια εντολή να μην έχει απενεργοποιηθεί, κάθε φορά γίνεται απενεργοποίηση όλων των λειτουργιών. Τέλος, με τον τερματισμό της διαδικασίας της παστερίωσης ή των σφαλμάτων ενεργοποιείται η εντολή "Run" όπου κάθε φορά που ενεργοποιείται από τον χρήστη γίνεται νέα αρχή της διαδικασίας παστερίωσης.



Σχήμα 4.1: Αρχή προγράμματος, κίνηση μπουκαλιού, ενεργοποίηση αισθητηρίου LS_1

Network 1

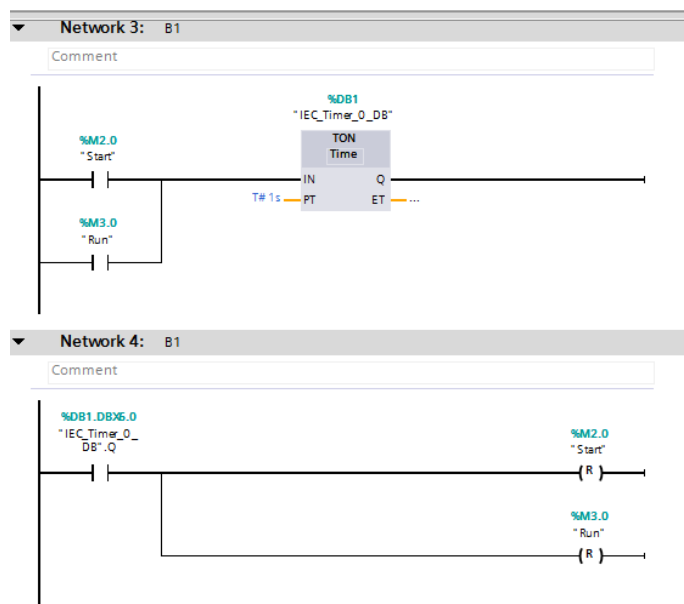
Με την ενεργοποίηση του start (S) γίνεται η αρχή της διαδικασίας καθώς και η απενεργοποίηση του αισθητηρίου LS_3 (R) (Σχήμα 4.1).

Network 2

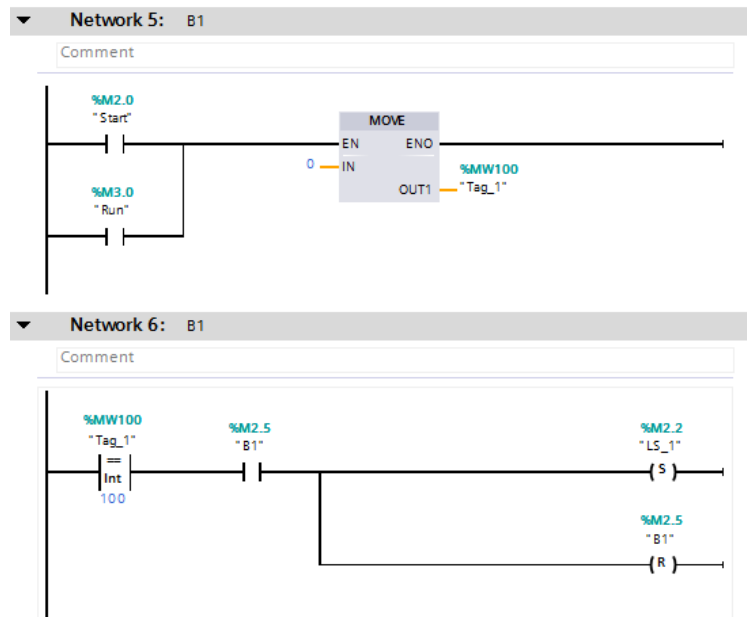
Όταν έχει τελειώσει η διαδικασία της παστερίωσης ή των σφαλμάτων με την ενεργοποίηση της εντολής run ξεκινά εκ νέου η διαδικασία με ενεργοποίηση της διαδρομής B1 και απενεργοποίηση του LS_3 (Σχήμα 4.1).

Network 3 και 4

Λόγω ότι πρέπει μετά από κάθε ενεργοποίηση του start(S), να απενεργοποιείται(R), έχει χρησιμοποιηθεί ένα χρονικό τύπου delay on με σκοπό όταν περάσει ο χρόνος που έχει οριστεί να πραγματοποιείται αυτή η λειτουργία. (Σχήμα 4.2)



Σχήμα 4.2: Συνέχεια προηγούμενου προγράμματος με χρονοκαθυστέρηση



Σχήμα 4.3: Εντολή “MOVE” για τον μηδενισμό memory word

Network 5

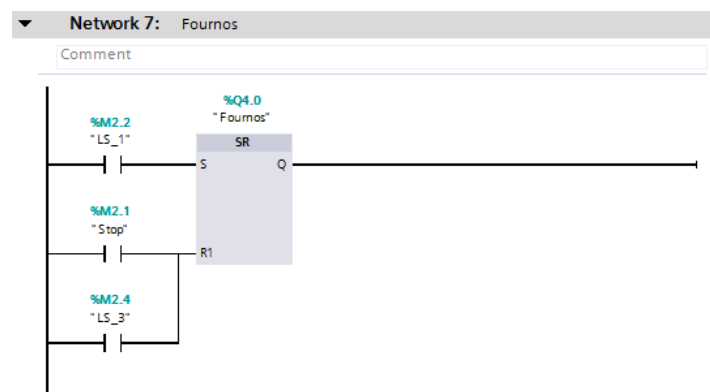
Για την πραγματοποίηση σωστής κίνησης του μπουκαλιού στην HMI οθόνη, δηλαδή να ξεκινάει το μπουκάλι από την αρχή της μεταφορικής ταινίας (σημείο μηδέν) και να καταλήγει στο αισθητήριο LS_1(100), χρησιμοποιήθηκε μπλοκ διάγραμμα τύπου MOVE που πραγματοποιεί αυτή την διαδικασία. Η απόσταση αυτή έχει οριστεί με διεύθυνση MW100 όπου με το μπλοκ MOVE και την ενεργοποίηση του Start ή του Run κάθε φορά η διαδικασία ξεκινά από το 0 (Σχήμα 4.3).

Network 6

Όταν η έξοδος του μπλοκ πάρει την τιμή 100 τότε ενεργοποιείται το αισθητήριο LS_1(S) και απενεργοποιείται η κίνηση B1(Σχήμα 4.3).

Network 7

Σ' αυτό το σημείο έχει χρησιμοποιηθεί μπλοκ με λειτουργία set-reset. Όταν το μπουκάλι φθάσει στο αισθητήριο LS_1 ενεργοποιείται ο φούρνος. Αντίστοιχα ο φούρνος απενεργοποιείται με την ενεργοποίηση του stop ή του αισθητηρίου LS_3 (Σχήμα 4.4).



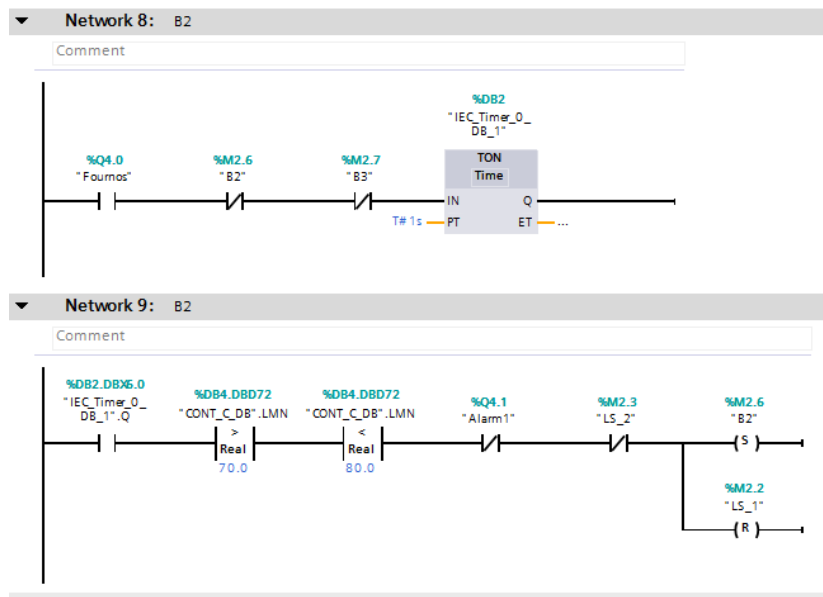
Σχήμα 4.4: Εντολή Set-Reset για την ενεργοποίηση του φούρνου

Network 8

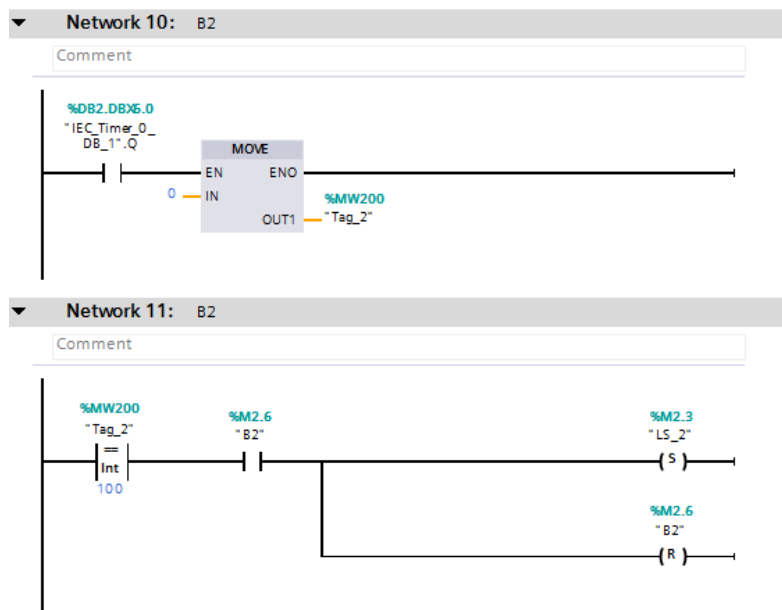
Το χρονικό τύπου delay on που έχει χρησιμοποιηθεί έχει σκοπό να δημιουργήσει μια μικρή καθυστέρηση για την μετάβαση της πρώτης φάσης στην δεύτερη φάση της διαδικασίας. Η χρήση του χρονικού είναι γιατί υπάρχει πρόβλημα στην απόκριση του συστήματος (Σχήμα 4.5).

Network 9

Για την ενεργοποίηση του μπουκαλιού στη δεύτερη φάση, δηλαδή το ξεκίνημα του μπουκαλιού από το LS_1 και να καταλήξει στο LS_2, έγινε χρήση δύο εντολών σύγκρισης. Όταν η θερμοκρασία έχει πάρει τιμή που είναι εντός ορίων, τότε ξεκινά η κίνηση του μπουκαλιού και απενεργοποιείται το αισθητήριο LS_1 (Σχήμα 4.5)



Σχήμα 4.5: Αρχή κίνησης μπουκαλιού στη 2^η φάση, ενεργοποίηση αισθητηρίου LS_2



Σχήμα 4.6: Εντολή MOVE για τον μηδενισμό memory word

Network 10

Για την πραγματοποίηση σωστής κίνησης του μπουκαλιού στην HMI οθόνη, δηλαδή να ξεκινάει το μπουκάλι από το σημείο LS_1(μηδέν) και να καταλήγει στο αισθητήριο LS_2(100), χρησιμοποιήθηκε μπλοκ διάγραμμα τύπου MOVE που πραγματοποιεί αυτή την διαδικασία. Η απόσταση αυτή έχει οριστεί με διεύθυνση MW200 όπου με το μπλοκ MOVE και την ενεργοποιήσει του start ή του run κάθε φορά η διαδικασία ξεκινά από το 0 (Σχήμα 4.6)

Network 11

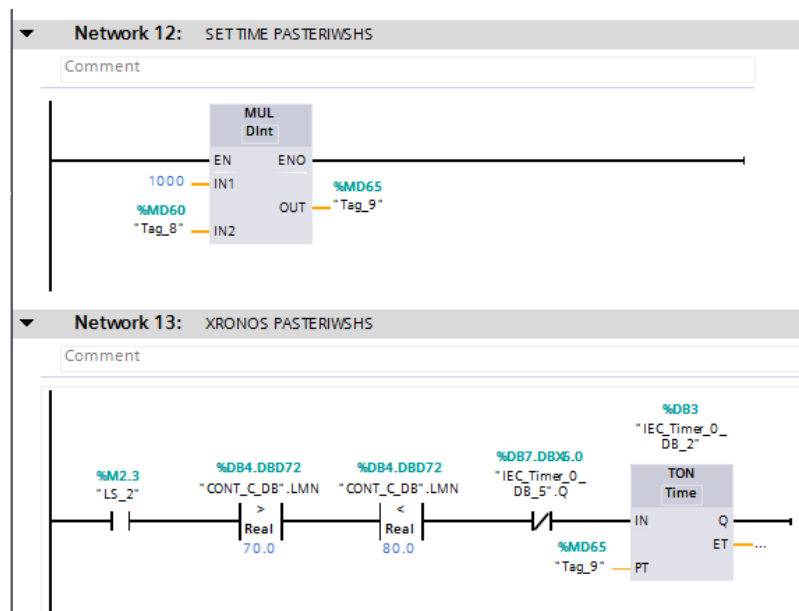
Όταν η έξοδος του μπλοκ πάρει την τιμή 100 τότε ενεργοποιείται το αισθητήριο LS_2 (S) και απενεργοποιείται η κίνηση B2 (Σχήμα 4.6).

Network 12

Για να μπορεί ο χρήστης να ορίζει χρόνο από την οθόνη HMI, χρόνο παστερίωσης στην παρούσα στιγμή, χρησιμοποιήθηκε μπλοκ διάγραμμα τύπου MUL που πραγματοποιεί πολλαπλασιασμό ώστε να ορίζεται ο χρόνος σε δευτερόλεπτα. Στην είσοδο IN1 ορίζεται το 1000 ενώ στην είσοδο IN2 ορίζεται ο επιθυμητός χρόνος από την HMI οθόνη με μια διεύθυνση τύπου MD60. Στο OUT ορίζεται μια διεύθυνση τύπου MD65 όπου θα καταγράφεται το αποτέλεσμα του πολλαπλασιασμού (Σχήμα 4.7).

Network 13

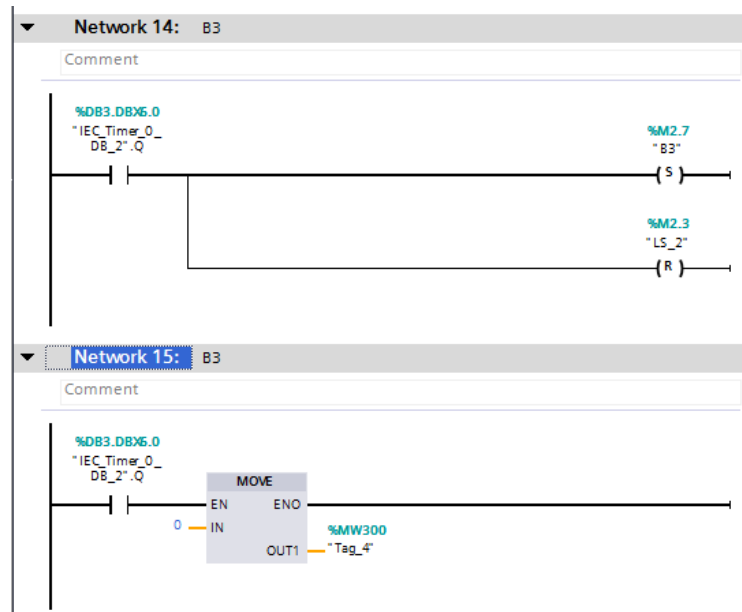
Για την ενεργοποίηση του χρόνου παστερίωσης στη δεύτερη φάση, έγινε χρήση δύο εντολών σύγκρισης. Όταν η θερμοκρασία έχει πάρει τιμή που είναι εντός ορίων και το μπουκάλι έχει φθάσει στο αισθητήριο LS_2 τότε ξεκινά ο χρόνος της παστερίωσης (Σχήμα 4.7).



Σχήμα 4.7: Πράξη πολλαπλασιασμού για Set Time σε μονάδα χρόνου σε second

Network 14

Με το πέρας του χρόνου της παστερίωσης, το LS_2 απενεργοποιείται και το μπουκάλι ξεκινά για την τρίτη φάση της διαδικασίας με κατάληξη στο LS_3 όπου εκεί θα τερματίσει κάθε λειτουργία (Σχήμα 4.8).



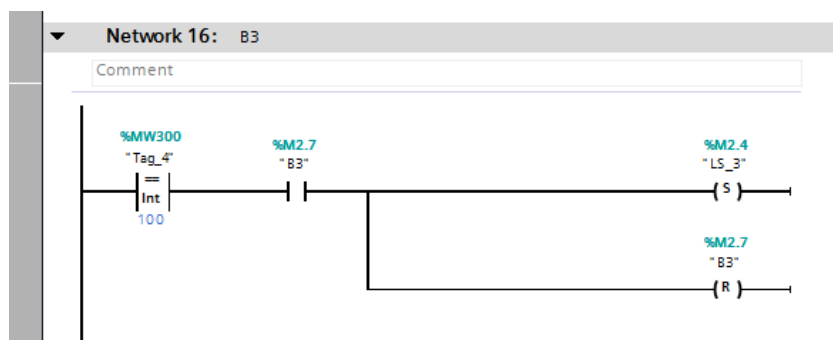
Σχήμα 4.8: Αρχή κίνησης μπουκαλιού στη 3^η φάση και ενεργοποίηση αισθητηρίου LS_3

Network 15

Για την πραγματοποίηση σωστής κίνησης του μπουκαλιού στην HMI οθόνη, δηλαδή να ξεκινάει το μπουκάλι από το σημείο LS_2(μηδέν) και να καταλήγει στο αισθητήριο LS_3(100), χρησιμοποιήθηκε μπλοκ διάγραμμα τύπου MOVE που πραγματοποιεί αυτή την διαδικασία. Η απόσταση αυτή έχει οριστεί με διεύθυνση MW300 όπου με το μπλοκ MOVE και την ενεργοποίηση του Start ή του Run κάθε φορά η διαδικασία ξεκινά από το 0 (Σχήμα 4.8).

Network 16

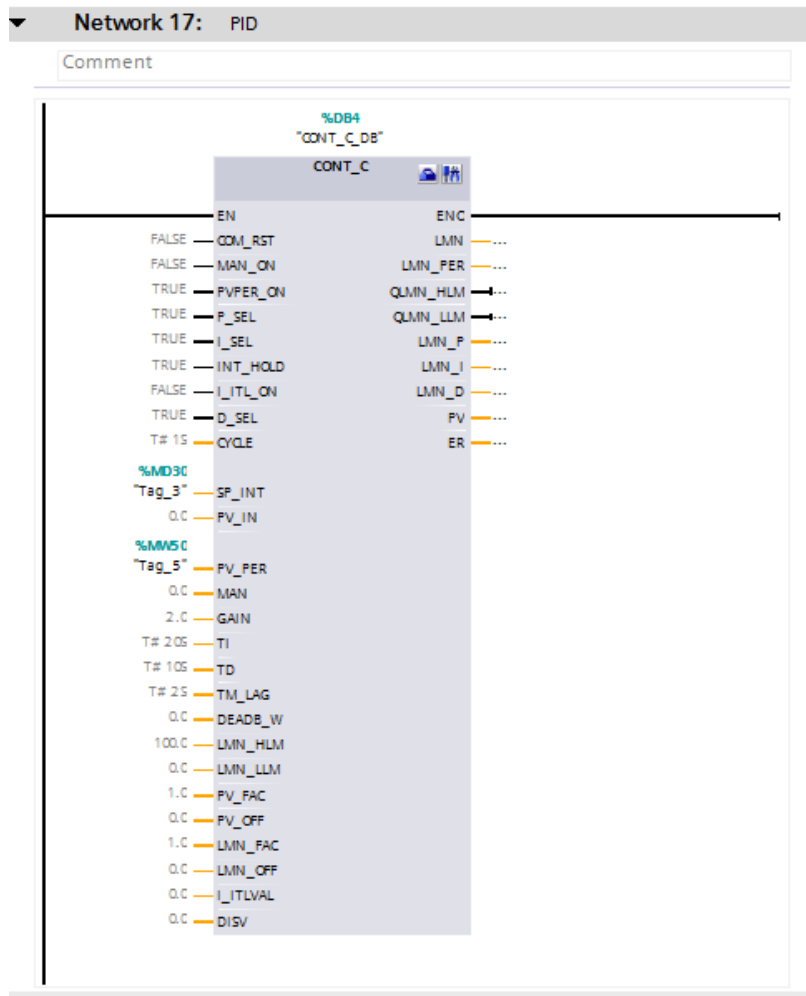
Όταν η έξοδος του μπλοκ πάρει την τιμή 100 τότε ενεργοποιείται το αισθητήριο LS_3(S) και απενεργοποιείται η κίνηση B3 (Σχήμα 4.9)



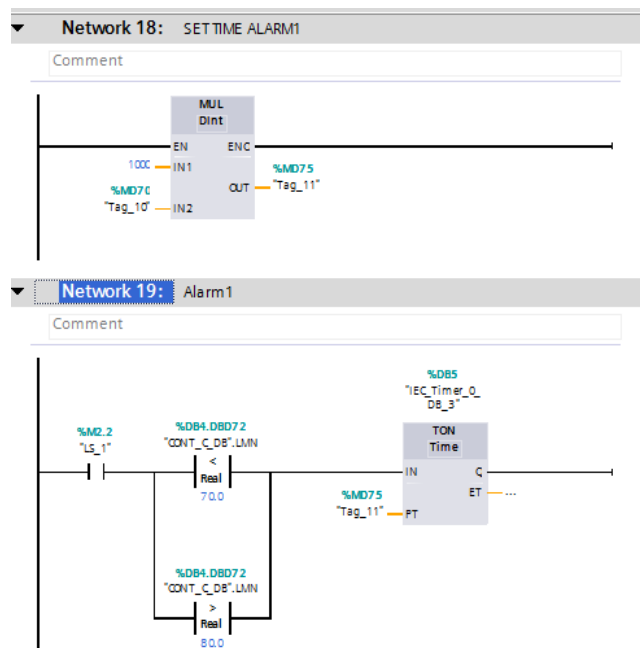
Σχήμα 4.9: Τέλος 3^{ης} φάσης και προγράμματος

Network 17

Σε αυτό το network (Σχήμα 4.10) παρουσιάζονται οι ρυθμίσεις του PID Ελεγκτή που αναλαμβάνει το έλεγχο της θερμοκρασίας.



Σχήμα 4.10: Ρύθμιση του PID ελεγκτή



Σχήμα 4.11: Πράξη πολλαπλασιασμού για Set Time σε μονάδα χρόνου σε second και συνθήκες για ενεργοποίηση χρονικού σφάλματος 1^ο.

Network 18

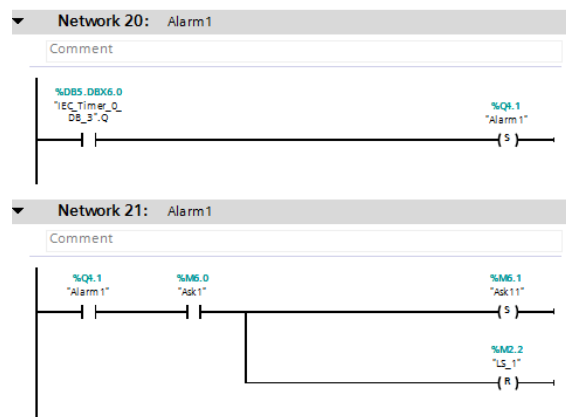
Για να μπορεί ο χρήστης να ορίζει χρόνο από την οθόνη HMI, χρόνο για alarm 1 στην παρούσα στιγμή, χρησιμοποιήθηκε μπλοκ διάγραμμα τύπου MUL που πραγματοποιεί πολλαπλασιασμό ώστε να ορίζεται ο χρόνος σε δευτερόλεπτα. Στην είσοδο IN1 ορίζεται το 1000 ενώ στην είσοδο IN2 ορίζεται ο επιθυμητός χρόνος από την HMI οθόνη με μια διεύθυνση τύπου MD70. Στο OUT ορίζεται μια διεύθυνση τύπου MD75 όπου θα καταγράφεται το αποτέλεσμα του πολλαπλασιασμού (Σχήμα 4.11).

Network 19

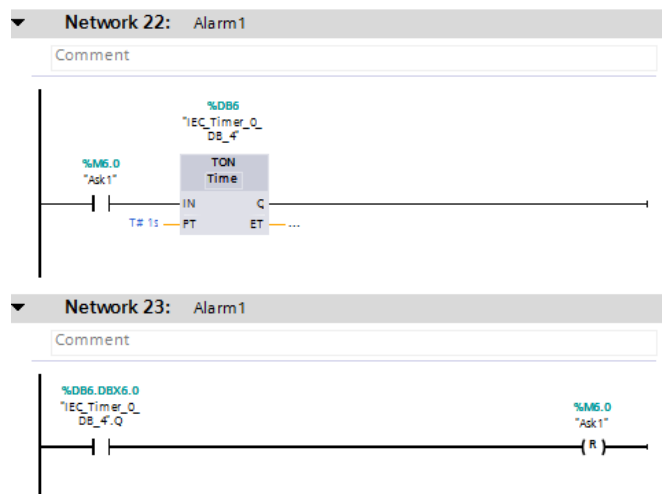
Η ενεργοποίηση του alarm 1 γίνεται σε συνδυασμό δύο εντολών σύγκρισης και εφόσον είναι ενεργοποιημένο το αισθητήριο LS_1. Όσο η τιμή της θερμοκρασίας είναι εκτός ορίων ο χρόνος μεταβάλλεται μέχρι να ενεργοποιηθεί η έξοδος του χρονικού (Σχήμα 4.11).

Network 20 και 21

Με την ενεργοποίηση της εξόδου του χρονικού ,ενεργοποιείται το alarm 1. Με την ενεργοποίηση του alarm 1 καθώς και την ενεργοποίηση της αναγνώρισης του σφάλματος (Ask1), απενεργοποιείται το αισθητήριο LS_1 και ξεκινάει η κίνηση του μπουκαλιού από το αισθητήριο LS_1 προς το LS_3 (Σχήμα 4.12)



Σχήμα 4.12: Ενεργοποίηση του 1^{ου} Alarm μετά από ορισμένο χρόνο



Σχήμα 4.13: Συνέχεια προγράμματος με επιβεβαίωση σφάλματος “Ask1”

Network 22

Χρονικό delay on με σκοπό να δημιουργήσει μικρή καθυστέρηση (Σχήμα 4.13).

Network 23

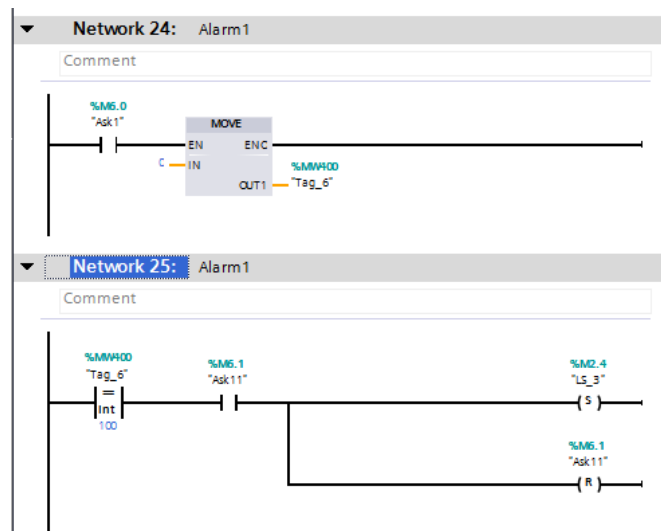
Με την ενεργοποίηση της εξόδου του χρονικού απενεργοποιείται η αναγνώριση του σφάλματος (Ask1) (Σχήμα 4.13).

Network 24

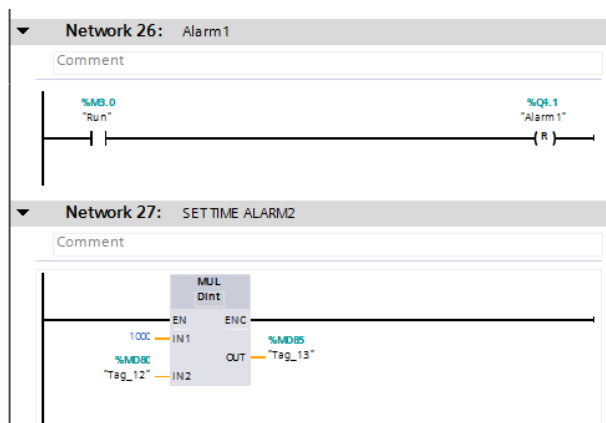
Για την πραγματοποίηση σωστής κίνησης του μπουκαλιού στην HMI οθόνη, δηλαδή να ξεκινάει το μπουκάλι από το σημείο LS_1(μηδέν) και να καταλήγει στο αισθητήριο LS_3(100), χρησιμοποιήθηκε μπλοκ διάγραμμα τύπου MOVE που πραγματοποιεί αυτή την διαδικασία. Η απόσταση αυτή έχει οριστεί με διεύθυνση MW400 όπου με την ενεργοποίηση του alarm 1 κάθε φορά η διαδικασία ξεκινά από το 0 (Σχήμα 4.14).

Network 25

Όταν η έξοδος του μπλοκ πάρει την τιμή 100 τότε ενεργοποιείται το αισθητήριο LS_3(S) και απενεργοποιείται η κίνηση Ask11 (Σχήμα 4.14).



Σχήμα 4.14:Εντολή MOVE για τον μηδενισμό memory word



Σχήμα 4.15:Τέλος σφάλματος με εντολή “Run”

Network 26

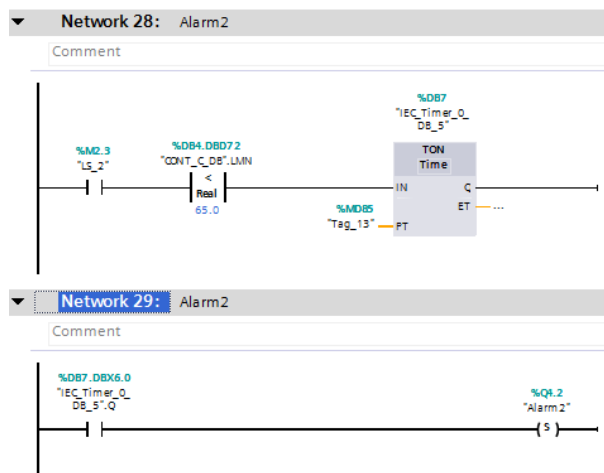
Με την ενεργοποίηση του run απενεργοποιείται το alarm 1 (Σχήμα 4.15).

Network 27

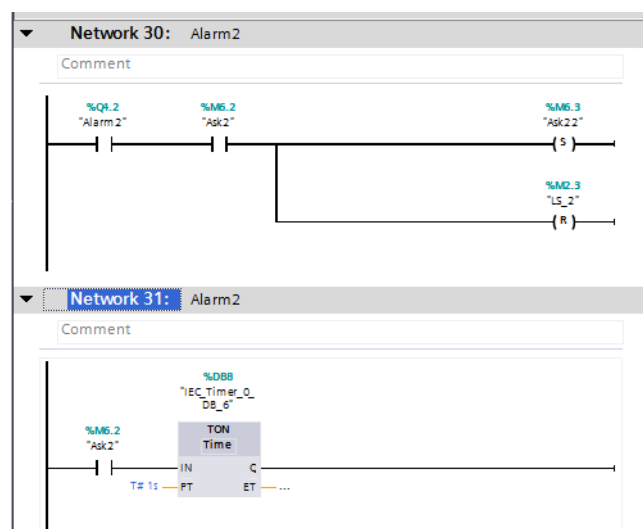
Για να μπορεί ο χρήστης να ορίζει χρόνο από την οθόνη HMI, χρόνο για alarm 2 στην παρούσα στιγμή, χρησιμοποιήθηκε μπλοκ διάγραμμα τύπου MUL που πραγματοποιεί πολλαπλασιασμό ώστε να ορίζεται ο χρόνος σε δευτερόλεπτα. Στην είσοδο IN1 ορίζεται το 1000 ενώ στην είσοδο IN2 ορίζεται ο επιθυμητός χρόνος από την HMI οθόνη με μια διεύθυνση τύπου MD80. Στο OUT ορίζεται μια διεύθυνση τύπου MD85 όπου θα καταγράφεται το αποτέλεσμα του πολλαπλασιασμού (Σχήμα 4.15).

Network 28 και 29

Το χρονικό τύπου delay on σε συνδυασμό με μια εντολή σύγκρισης και με ενεργοποιημένο το αισθητήριο LS_2, ξεκινά να μεταβάλλει τον χρόνο. Με την ενεργοποίηση της εξόδου του χρονικού ενεργοποιείτε το alarm 2 (Σχήμα 4.16).



Σχήμα 4.16: Ενεργοποίηση χρονικού 2^{ου} σφάλματος με συνθήκες



Σχήμα 4.17: Ενεργοποίηση 2^{ου} σφάλματος και αναγνώριση σφάλματος

Network 30

Με την ενεργοποίηση του alarm 2 καθώς και την ενεργοποίηση της αναγνώρισης του σφάλματος (Ask2) απενεργοποιείται το αισθητήριο LS_2 και ξεκινάει η κίνηση του μπουκαλιού από το αισθητήριο LS_2 προς το LS_3 (Σχήμα 4.17).

Network 31

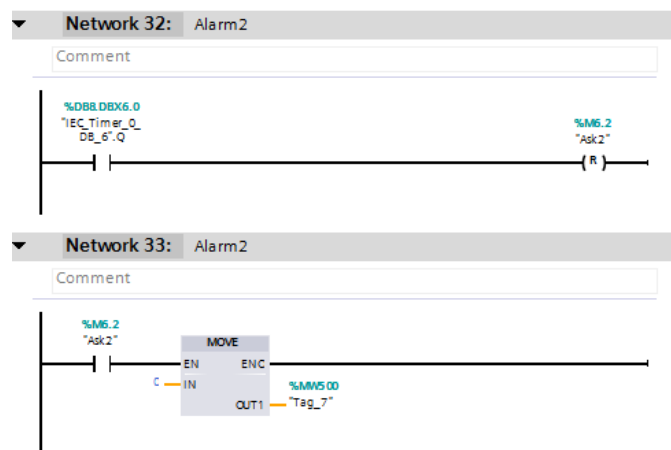
Χρονικό delay οπ με σκοπό να δημιουργήσει μικρή καθυστέρηση (Σχήμα 4.17).

Network 32

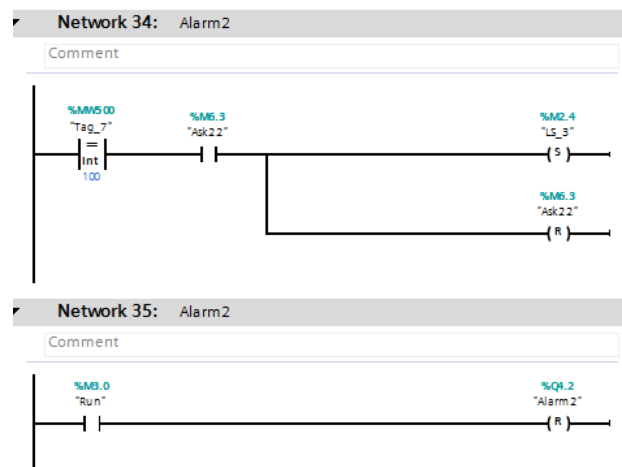
Με την ενεργοποίηση της εξόδου του χρονικού απενεργοποιείται η αναγνώριση του σφάλματος (Ask2) (Σχήμα 4.18).

Network 33

Για την πραγματοποίηση σωστής κίνησης του μπουκαλιού στην HMI οθόνη, δηλαδή να ξεκινάει το μπουκάλι από το σημείο LS_2(μηδέν) και να καταλήγει στο αισθητήριο LS_3(100), χρησιμοποιήθηκε μπλοκ διάγραμμα τύπου MOVE που πραγματοποιεί αυτή την διαδικασία. Η απόσταση αυτή έχει οριστεί με διεύθυνση MW500 όπου με την ενεργοποίηση του alarm 2 κάθε φορά η διαδικασία ξεκινά από το 0 (Σχήμα 4.18).



Σχήμα 4.18: Απενεργοποίηση της αναγνώρισης σφάλματος και μηδενισμό memory word



Σχήμα 4.19:Ενεργοποίηση 2^{ου} σφάλματος με τις συνθήκες

Network 34 και 35

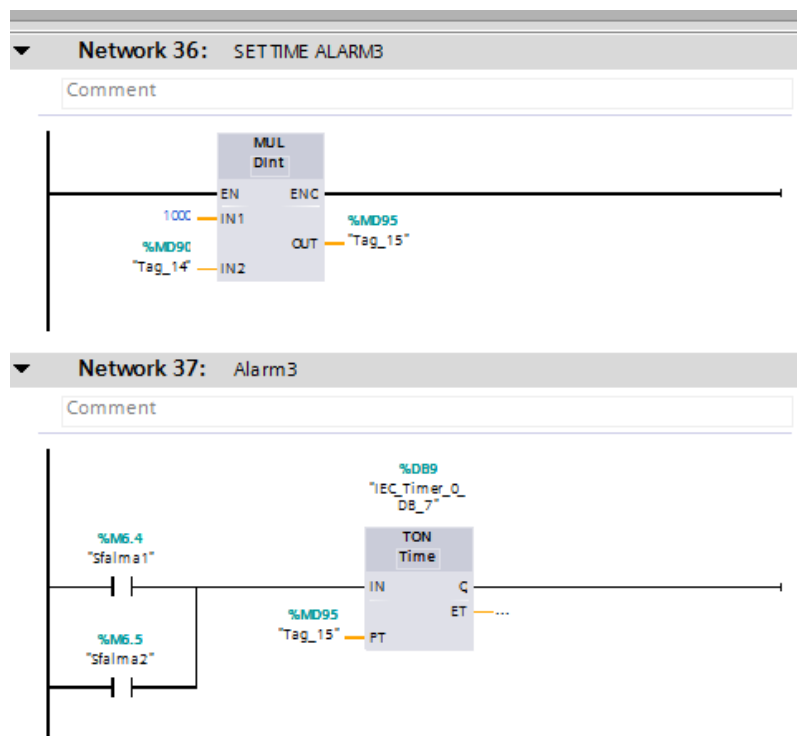
Όταν η έξοδος του μπλοκ πάρει την τιμή 100 τότε ενεργοποιείται το αισθητήριο LS_3(S) και απενεργοποιείται η κίνηση Ask22 (Σχήμα 4.19). Ενώ με την ενεργοποίηση του run απενεργοποιείται το alarm 2.

Network 36

Για να μπορεί ο χρήστης να ορίζει χρόνο από την οθόνη HMI, χρόνο για alarm 1 στην παρούσα στιγμή, χρησιμοποιήθηκε μπλοκ διάγραμμα τύπου MUL που πραγματοποιεί πολλαπλασιασμό για να ορίζεται ο χρόνος αμέσως σε δευτερόλεπτα. Στην είσοδο IN1 ορίζεται το 1000 ενώ στην είσοδο IN2 ορίζεται ο επιθυμητός χρόνος από την HMI οθόνη με μια διεύθυνση τύπου MD90. Στο OUT ορίζεται μια διεύθυνση τύπου MD95 όπου θα καταγράφεται το αποτέλεσμα του πολλαπλασιασμού (Σχήμα 4.20).

Network 37

Αν ενεργοποιηθεί το Sfalma 1 ή το Sfalma 2 ο χρόνος του χρονικού αρχίζει να μεταβάλλεται (Σχήμα 4.20).



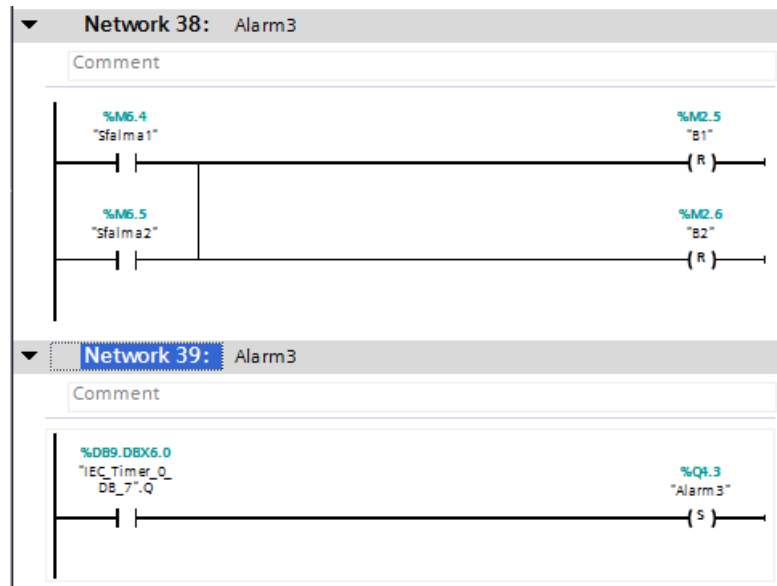
Σχήμα 4.20: Πράξη πολλαπλασιασμού για Set Time σε μονάδα χρόνου σε second και συνθήκες για ενεργοποίηση χρονικού σφάλματος 2.

Network 38

Με την ενεργοποίηση του Sfalma 1 και Sfalma 2 διακόπτονται οι κινήσεις μπουκαλιών B1 και B2 (Σχήμα 4.21).

Network 39

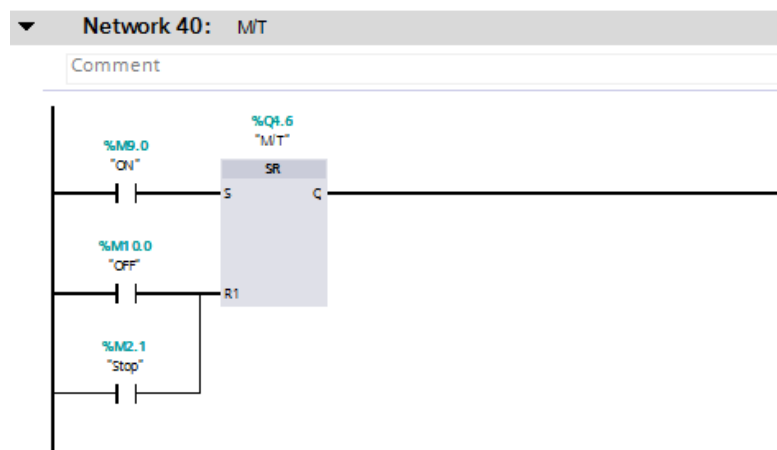
Με την ενεργοποίηση της εξόδου του χρονικού ενεργοποιείται το alarm 3 (Σχήμα 4.21).



Σχήμα 4.21: Ορισμός του Sfalma 1 και Sfalma 2

Network 40

Είναι ένα μπλοκ διάγραμμα με εντολή set-reset της λειτουργίας της μεταφορικής ταινίας (Σχήμα 4.22).



Σχήμα 4.22: Εντολή Set-Reset για ενεργοποίηση της μεταφορικής ταινίας

Network 41

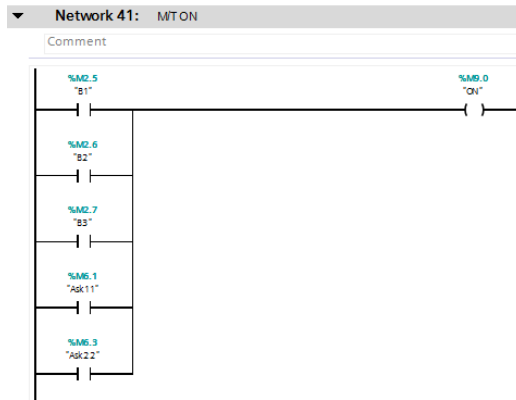
Περιλαμβάνει όσες διαδικασίες απαιτούν την κίνηση της μεταφορικής ταινίας (Σχήμα 4.23).

Network 42

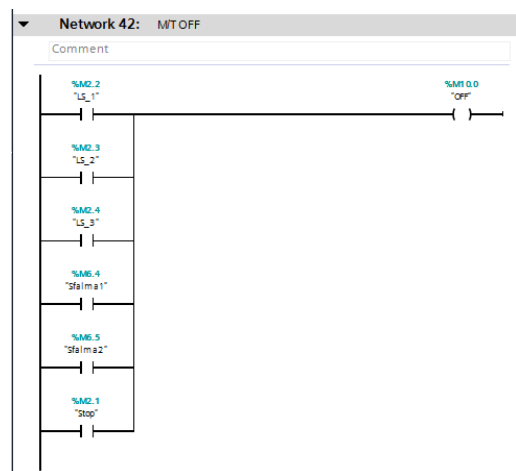
Περιλαμβάνει όσες διαδικασίες απαιτούν η μεταφορική ταινία να είναι σταματημένη (Σχήμα 4.24).

Network 43

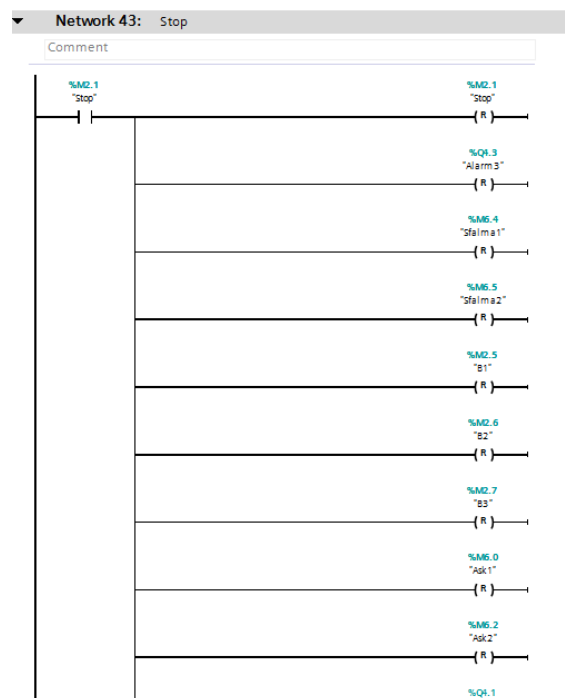
Όταν ενεργοποιηθεί το stop κάνει όλες τις μεταβλητές reset (Σχήματα 4.25 και 4.26)..



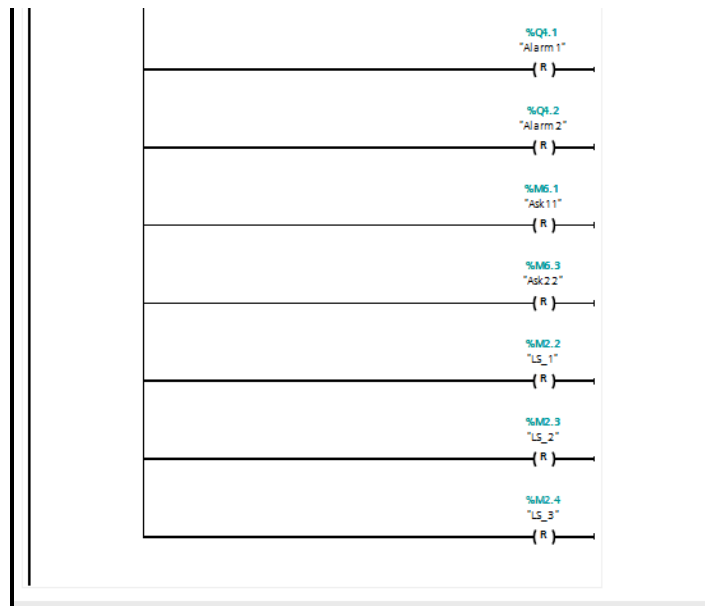
Σχήμα 4.23: Συνθήκες για την κατάσταση της M/T σε “ON”



Σχήμα 4.24: Συνθήκες για την κατάσταση της M/T σε “OFF”



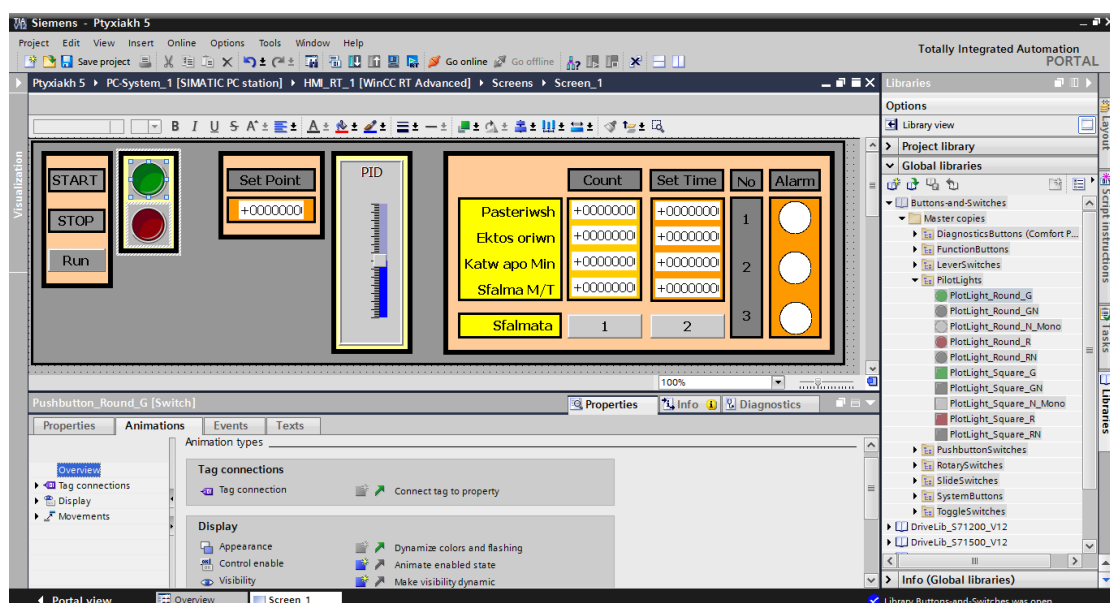
Σχήμα 4.25: Απενεργοποίηση όλων των λειτουργιών με εντολή “STOP”



Σχήμα 4.26: Απενεργοποίηση όλων των λειτουργιών με εντολή “STOP” (Συνέχεια)

4.3 Υλοποίηση του HMI

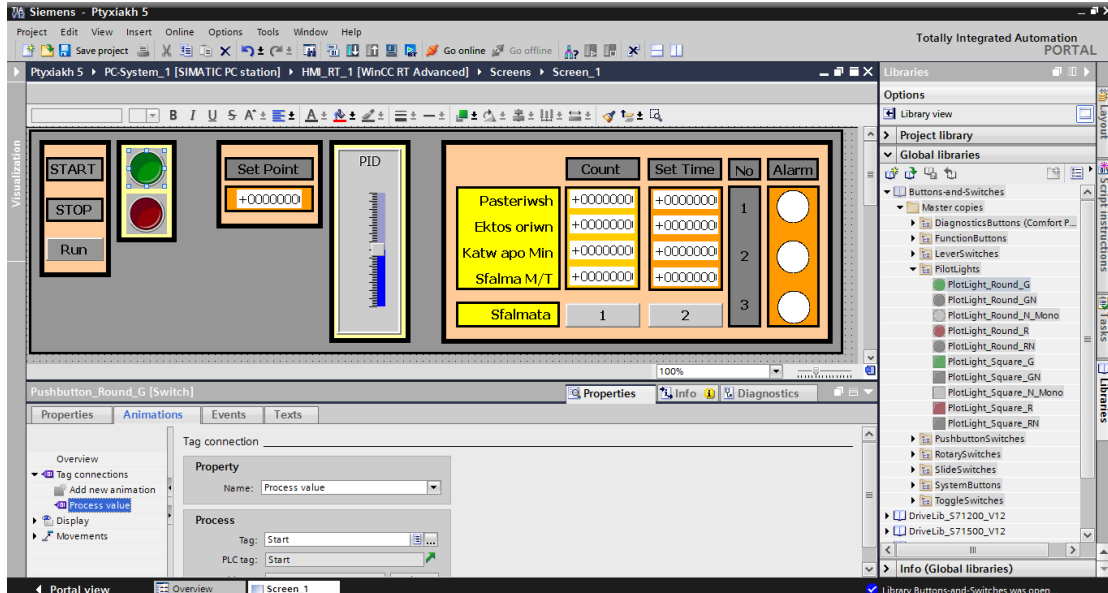
Για την υλοποίηση της γραφικής απεικόνισης χρησιμοποιήθηκαν γραφικά στοιχεία και εργαλεία από τις βιβλιοθήκες που διαθέτει το πρόγραμμα. Σημαντική λειτουργία της εφαρμογής είναι ότι μέσω κάποιων ρυθμίσεων ο χρήστης έχει την δυνατότητα να αποδώσει κάποια κίνηση σε γραφικά στοιχεία σε οποιαδήποτε κατεύθυνση επιθυμεί. Επιπλέον η σήμανση LED ή φωτεινών σηματοδοτών είναι εφικτή καθώς και η χρήση μπουτον ή διακοπτών. Η κίνηση ενός στοιχείου ή μιας ένδειξης σηματοδοτή έχει άμεση σχέση με το πρόγραμμα της εφαρμογής καθώς η γραφική απεικόνιση αποδίδει λειτουργίες ή αποτελέσματα του προγράμματος.



Σχήμα 4.27:Επιλογή σηματοδοτών, μπουτον

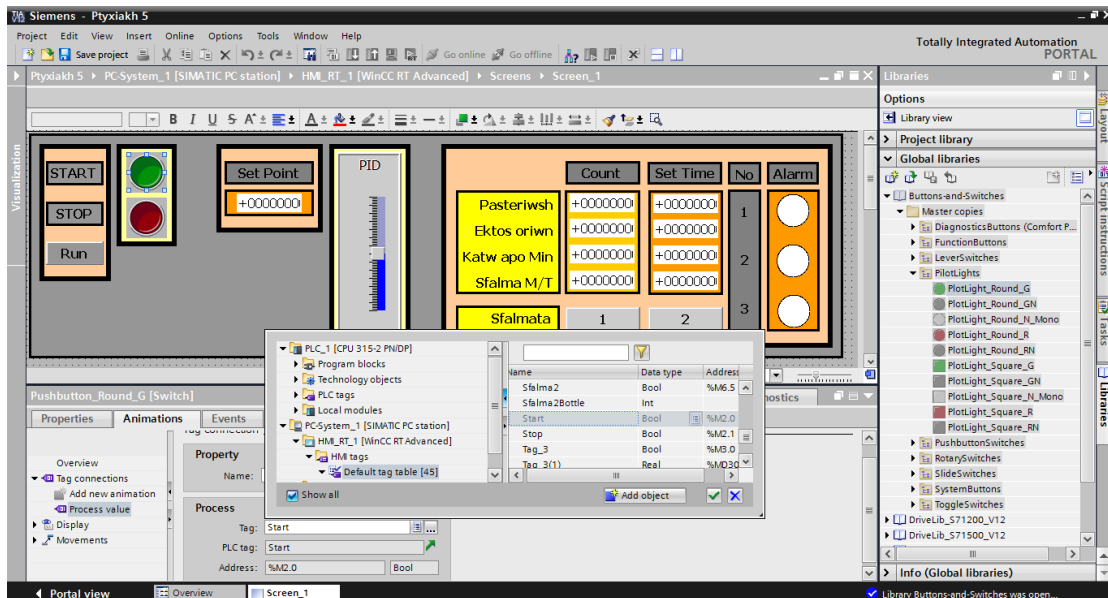
4.3.1 Ορισμός μπουτόν με ενδεικτική λυχνία

Στις βιβλιοθήκες του προγράμματος υπάρχει κατάλογος με όνομα Global Libraries (γενικές βιβλιοθήκες) όπου διαθέτει κατηγορία με μπουτόν, διακόπτες καθώς και φωτεινούς σηματοδότες τύπου LED. Ο χρήστης κάνει επιλογή του επιθυμητού μπουτόν-σηματοδότη (Σχήμα 4.27). Με κλικ πάνω στο στοιχείο αναδύεται παράθυρο με τις ρυθμίσεις που μπορούν να πραγματοποιηθούν(Σχήμα 4.28).



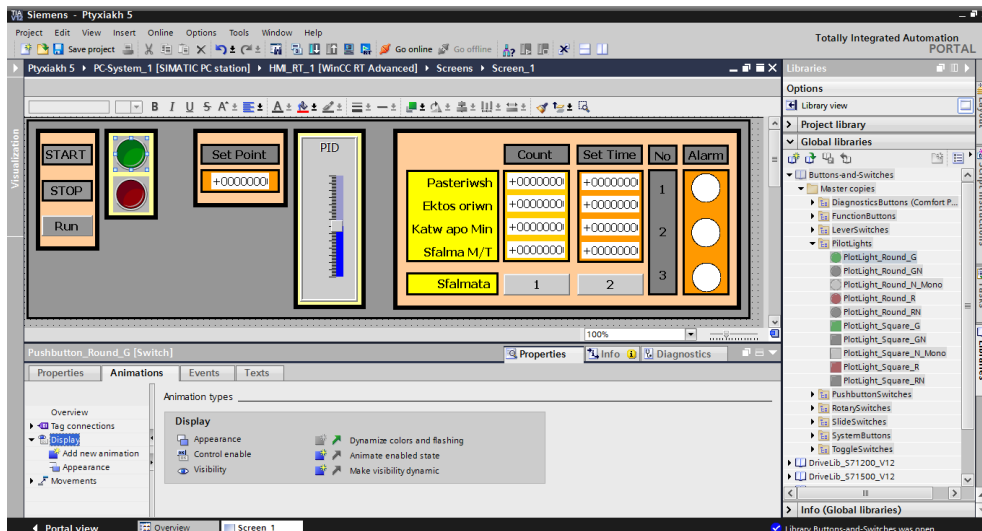
Σχήμα 4.28: Ρύθμιση για ενεργοποίηση μπουτον και σηματοδοτών

Στην κατηγορία Animations στο πλαίσιο Property, επιλέγουμε Process Value και στο πλαίσιο Process την διεύθυνση της λειτουργίας που έχει οριστεί στο πρόγραμμα (Σχήματα 4.28 και 4.29).

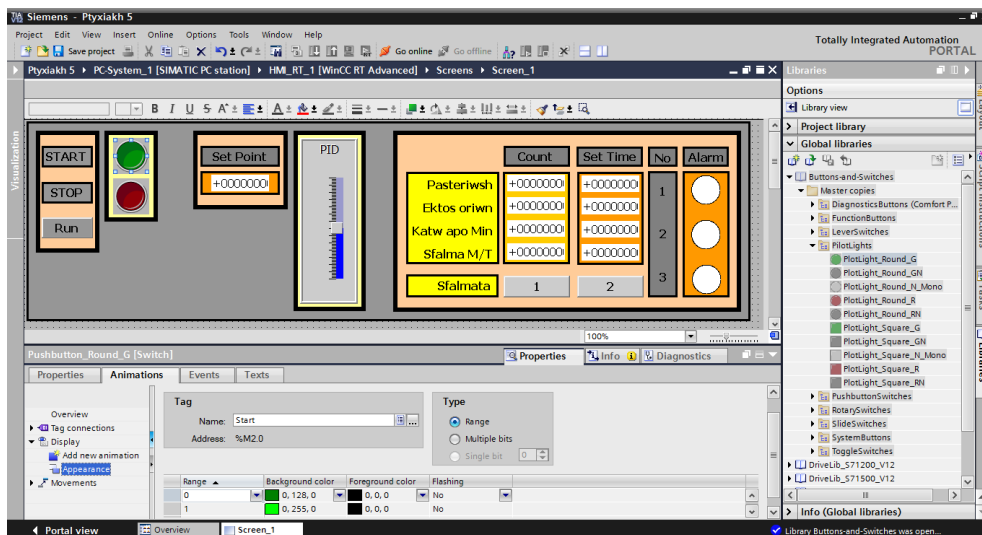


Σχήμα 4.29: Ρύθμιση για ενεργοποίηση μπουτον και σηματοδοτών (Συνέχεια)

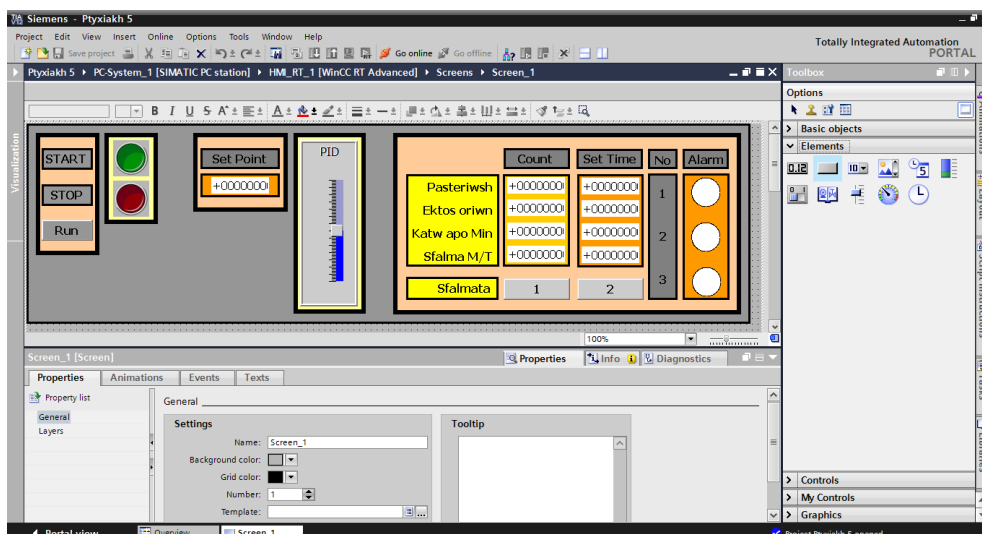
Επίσης είναι δυνατή η επιλογή του χρώματος όταν είναι σε λειτουργία με την επιλογή Display>Appearance και ρυθμίζεται Dynamize colors and flashing(Σχήματα 4.30 και 4.31).



Σχήμα 4.30: Ορισμός εμφάνισης σηματοδότη με χρώμα



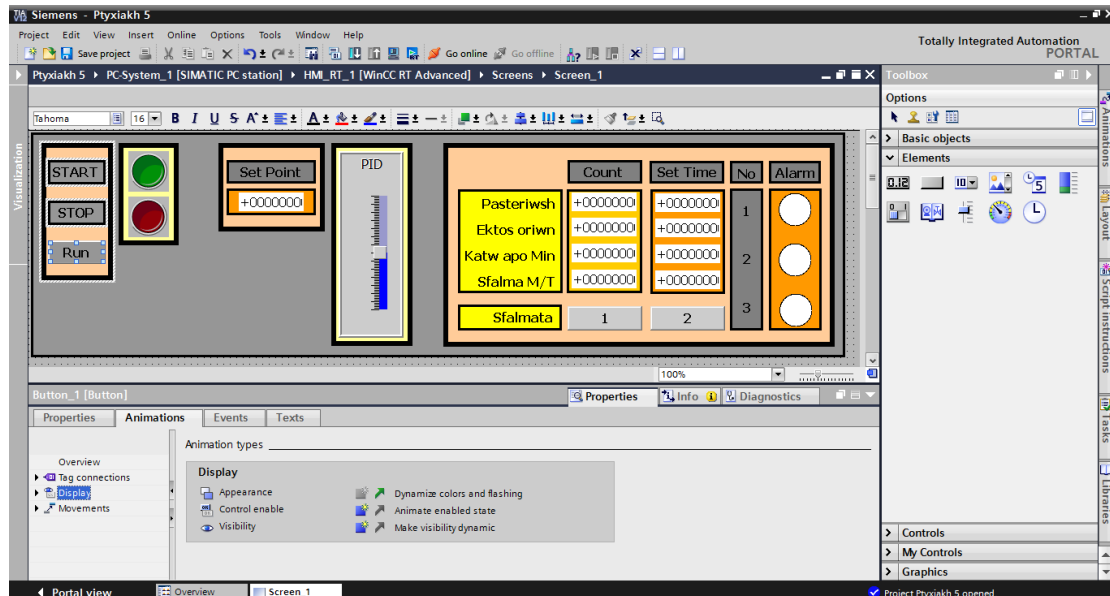
Σχήμα 4.31: Ορισμός εμφάνισης σηματοδότη με χρώμα (Συνέχεια)



Σχήμα 4.32: Επιλογή μπουτον από κατηγορία “Elements”

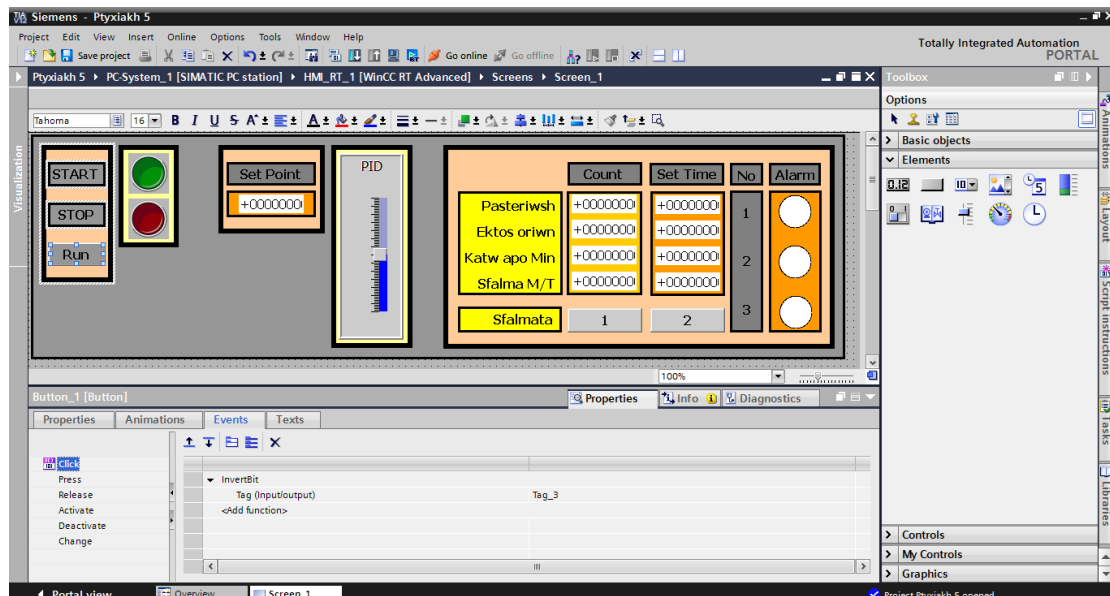
4.3.2 Επιλογή μπουτόν

Η εργαλειοθήκη (Tool Box) του προγράμματος στην κατηγορία Elements διαθέτει buttons(Σχήμα 4.32) απλού τύπου. Με την επιλογή του μπουτον, υπάρχει η δυνατότητα να δοθεί όνομα για να γνωρίζουμε σε τι λειτουργία αντιστοιχεί όπως Start, Stop. Στη περίπτωση της εφαρμογής δίνεται όνομα Run (Σχήμα 4.33). Επίσης με κλικ πάνω στο μπουτον αναδύεται παράθυρο με τις ρυθμίσεις που μπορούν να πραγματοποιηθούν.



Σχήμα 4.33: Επιλογή μπουτον από κατηγορία “Elements” (Συνέχεια)

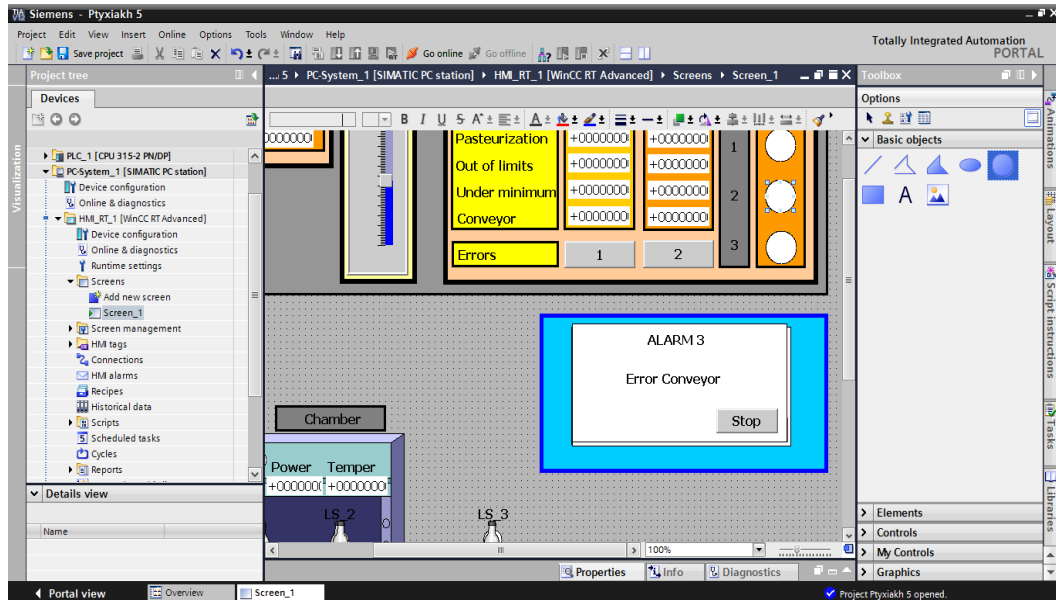
Στο αναδυόμενο παράθυρο στην κατηγορία Events και στην επιλογή Click, επιλέγουμε InvertBit καθώς και την διεύθυνση όπου αντιστοιχεί η λειτουργία που θέλουμε να εκτελούμε πατώντας αυτό(Σχήμα 4.34).



Σχήμα 4.34: Ορισμός εκτέλεσης λειτουργίας με την ενεργοποίηση του μπουτον

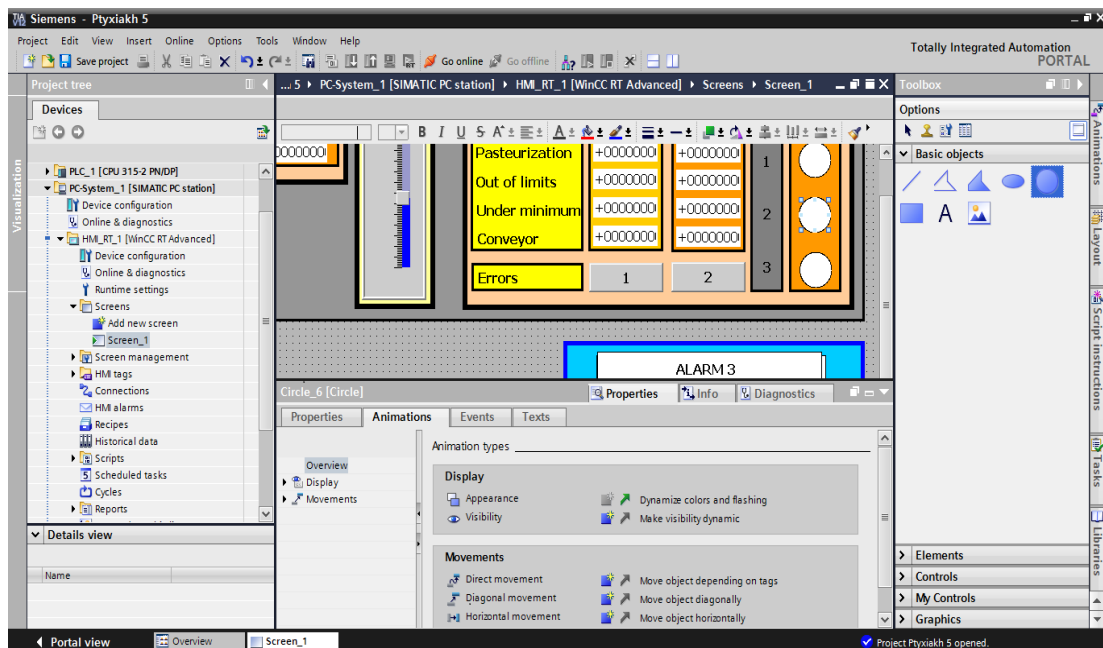
4.3.3 Επιλογή φωτεινού σηματοδότη

Στην εργαλειοθήκη (Tool Box) του προγράμματος γίνεται επιλογή Basic objects>Circle. Με αυτό το εργαλείο ο χρήστης μπορεί να σχηματίσει κύκλους. Στην παρούσα περίπτωση σχηματίζεται ενδεικτική λυχνία για προειδοποίηση του Alarm (Σχήμα 4.35).



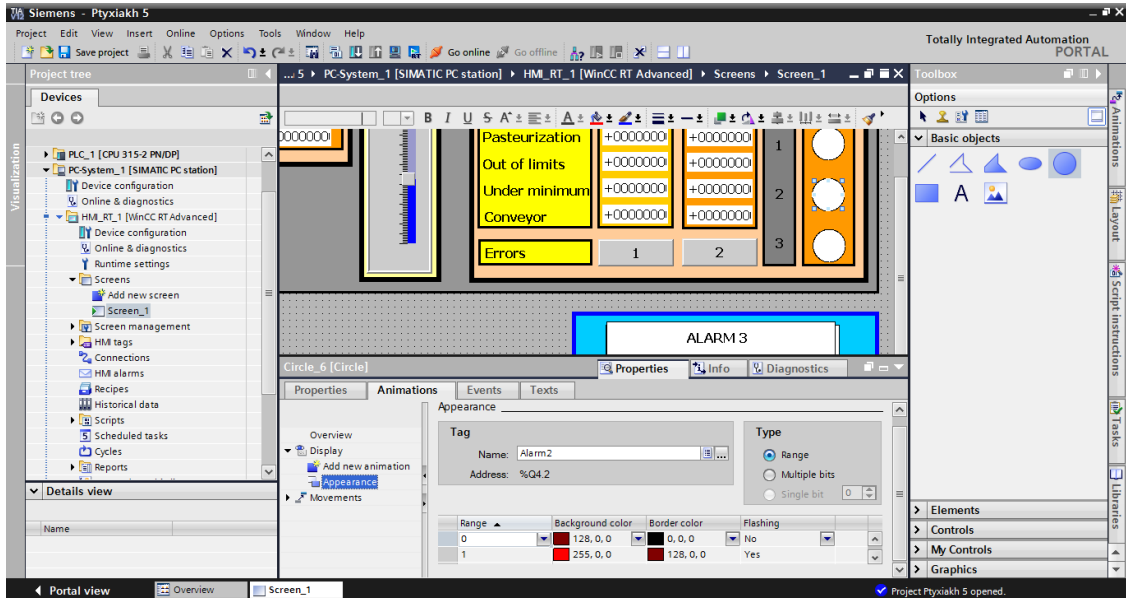
Σχήμα 4.35: Επιλογή εργαλείου για κύκλους

Αφού έχει σχηματιστεί στις κατάλληλες διαστάσεις, με την επιλογή του στοιχείου αναδύεται παράθυρο με τις ρυθμίσεις του(Σχήμα 4.36).



Σχήμα 4.36: Επιλογή για την ρύθμιση Tag και χρωμάτων λειτουργίας

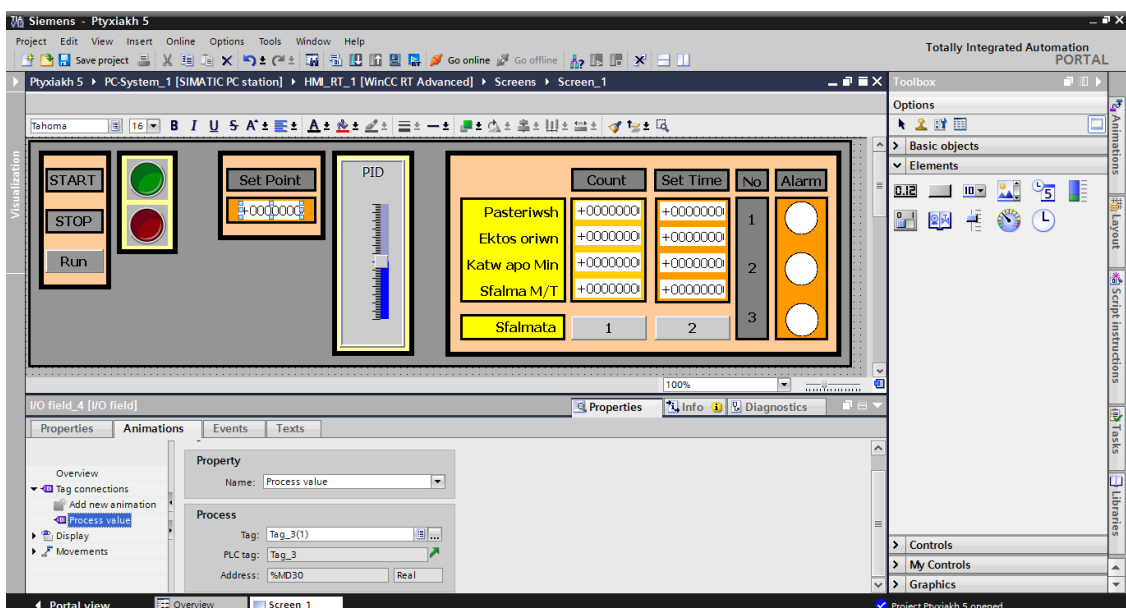
Οι επιλογές στο αναδυόμενο παράθυρο είναι οι εξής Animations>Display> Appearance όπου ρυθμίζεται Dynamize colors and flashing. Σε αυτό το σημείο ορίζεται το Tag που θα συνδέεται η γραφική απεικόνιση με το πρόγραμμα του PLC καθώς και τους χρωματισμούς που θα παίρνει κατά την ενεργοποίηση ή απενεργοποίηση του.



Σχήμα 4.37: Ρυθμίσεις Tag και ορισμός χρωμάτων λειτουργίας

4.3.4 Παρουσίαση αριθμών

Στην εργαλειοθήκη (Tool Box) του προγράμματος γίνεται επιλογή I/O field. Με την επιλογή αυτή δίνεται η δυνατότητα παρουσίασης τιμών ή εκχώρηση κάποιου τιμής σε κάποιο χρονικό, μετρητή κλπ(σχήμα 4.49).

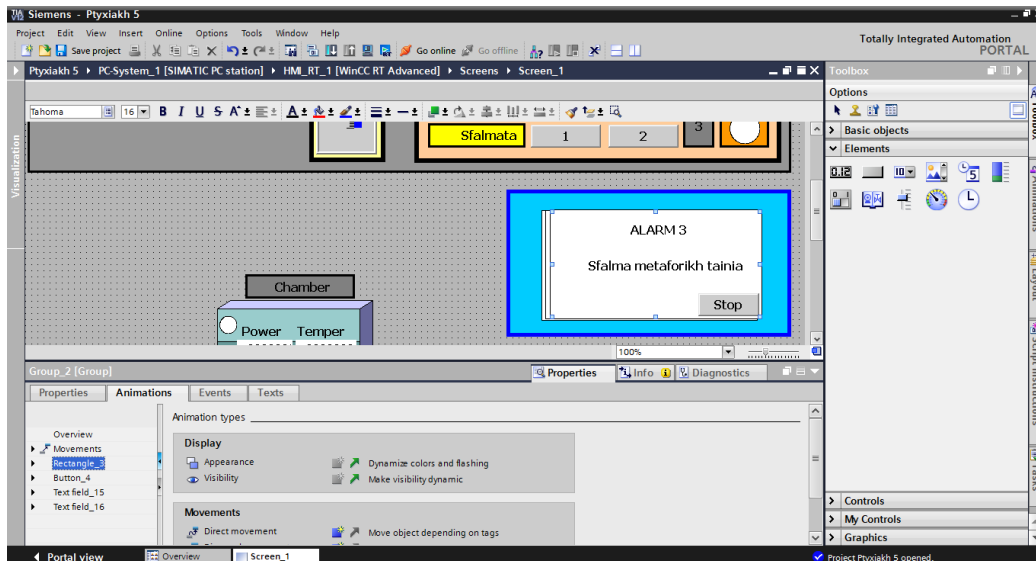


Σχήμα 4.38: Επιλογή I/O Field για καταχώρηση ή εμφάνιση αριθμών

Με κλικ πάνω στο στοιχείο αναδύεται παράθυρο με τις ρυθμίσεις που επιθυμούμε να πραγματοποιήσουμε. Στο σχήμα 4.38 γίνεται επιλογή του tag 3 με διεύθυνση MD30, διότι έχει οριστεί από τον χρήστη στο πρόγραμμα να καταχωρείτε το Set Point. σε αυτό το πεδίο(field).

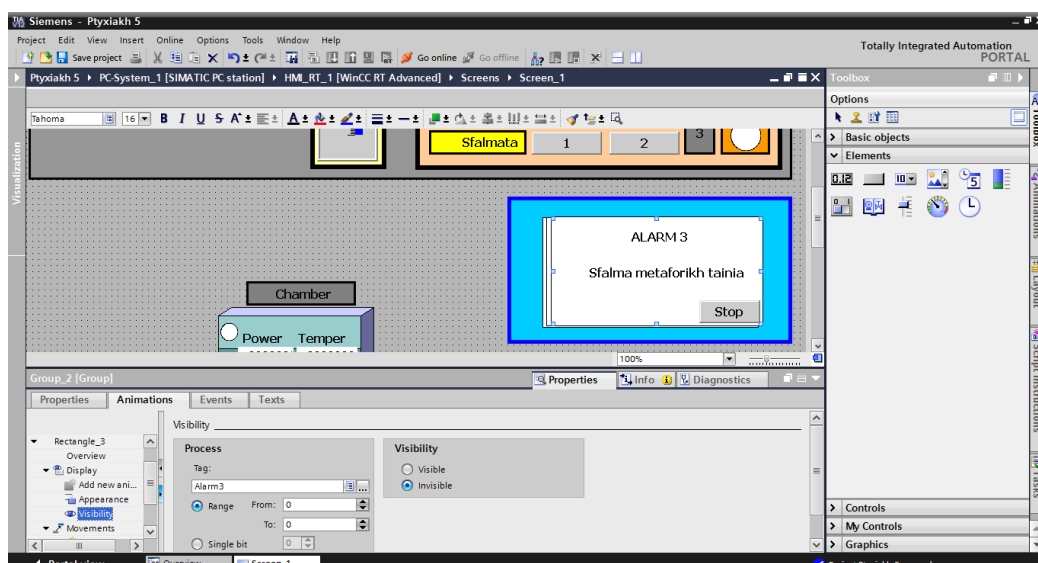
4.3.5 Παρουσίαση πλαισίου σφαλμάτων

Στην εργαλειοθήκη (Tool Box) του προγράμματος στην κατηγορία Basic Objects γίνεται επιλογή του εικονιδίου Rectangle. Με αυτό το εργαλείο ο χρήστης μπορεί να δημιουργήσει πλαίσιο στις διαστάσεις που επιθυμεί (Σχήμα 4.39).



Σχήμα 4.39:Επιλογή πλαισίου

Έχει την δυνατότητα να γράφει κάποιο κείμενο εσωτερικά του πλαισίου. Στο αναδυόμενο παράθυρο στην κατηγορία Animations, στην επιλογή Appearance, ορίζεται πως θα εμφανίζεται.

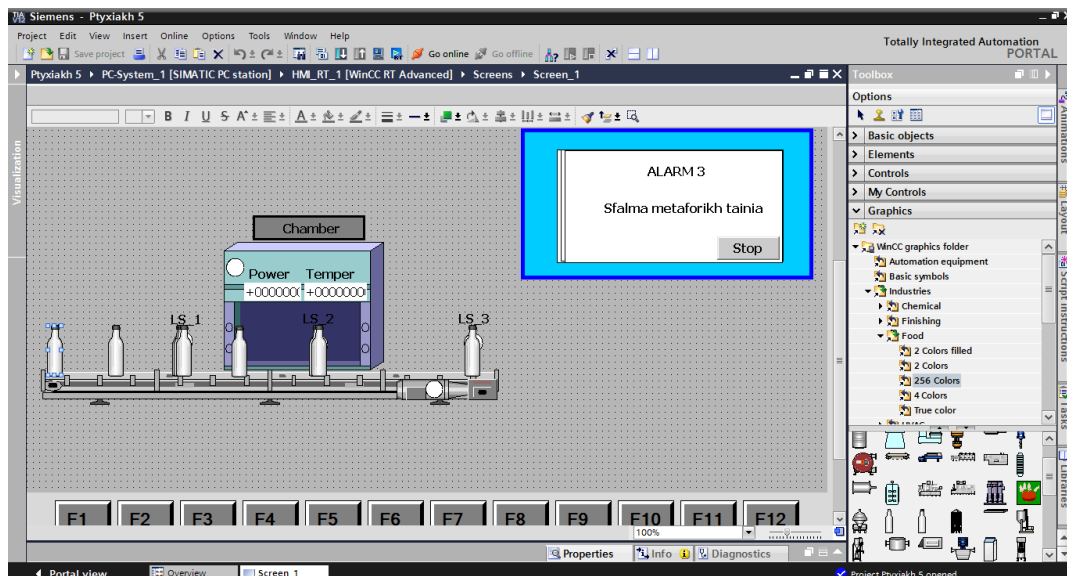


Σχήμα 4.40:Ορισμός λειτουργιών του πλαισίου

Επιπλέον στην επιλογή Visibility (Σχήμα 4.40) ρυθμίζεται πότε θα εμφανίζεται το πλαίσιο που έχουμε δημιουργήσει επιλέγοντας bit "0" ή "1". Στο πλαίσιο Process, επιλέγουμε την διεύθυνση της λειτουργίας που έχει οριστεί στο πρόγραμμα και στο πλαίσιο Visibility ορίζεται Visible ή Invisible, πότε θα γίνεται διακριτό το πλαίσιο.

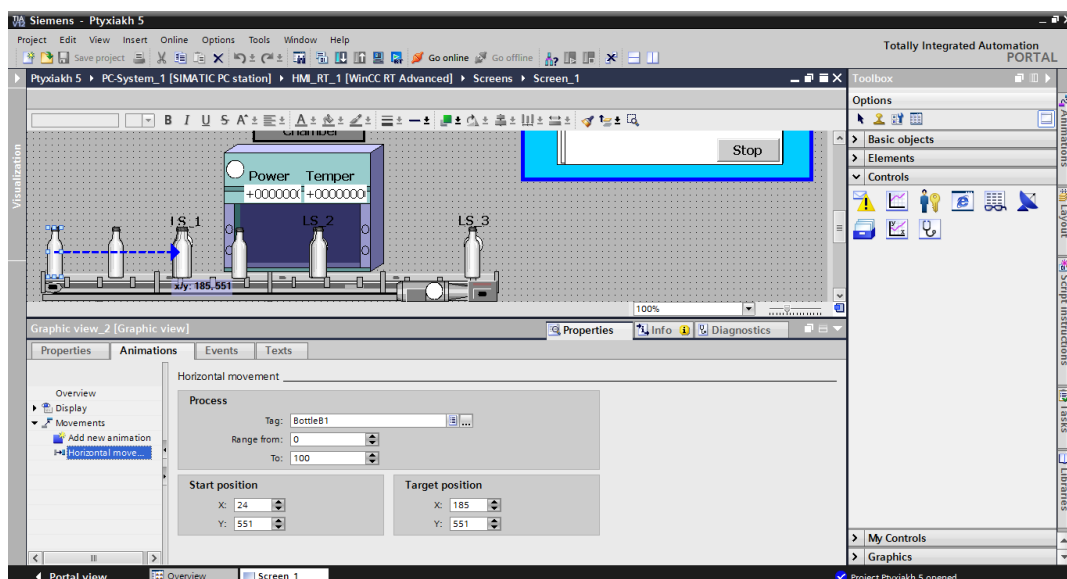
4.3.6 Κίνηση μπουκαλιού

Η κατηγορία Graphics της εργαλειοθήκης του προγράμματος υπάρχει μεγάλο εύρος γραφικών στοιχείων για την πραγματοποίηση της εφαρμογής. Στο σχήμα 4.41 γίνεται επιλογή του σχετικού μπουκαλιού.



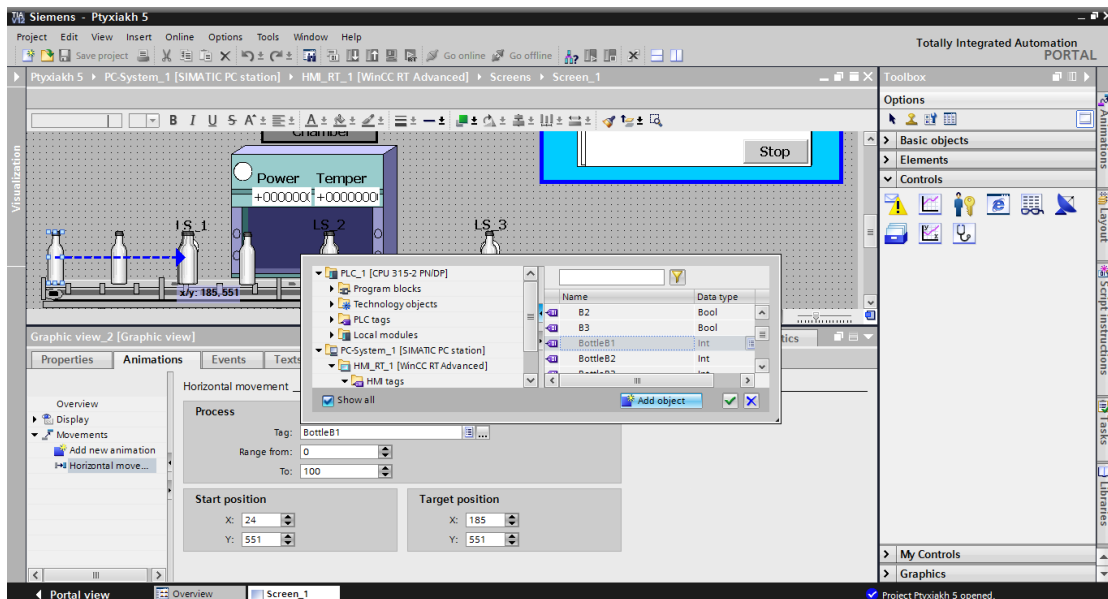
Σχήμα 4.41: Επιλογή μπουκαλιού

Με κλικ στο μπουκάλι στην επιφάνεια εργασίας αναδύεται παράθυρο ώστε να γίνουν οι απαραίτητες ρυθμίσεις. Στην κατηγορία Animations, στην επιλογή Horizontal Movements ρυθμίζονται τα όρια που θα κινείται το μπουκάλι (Σχήμα 4.42).

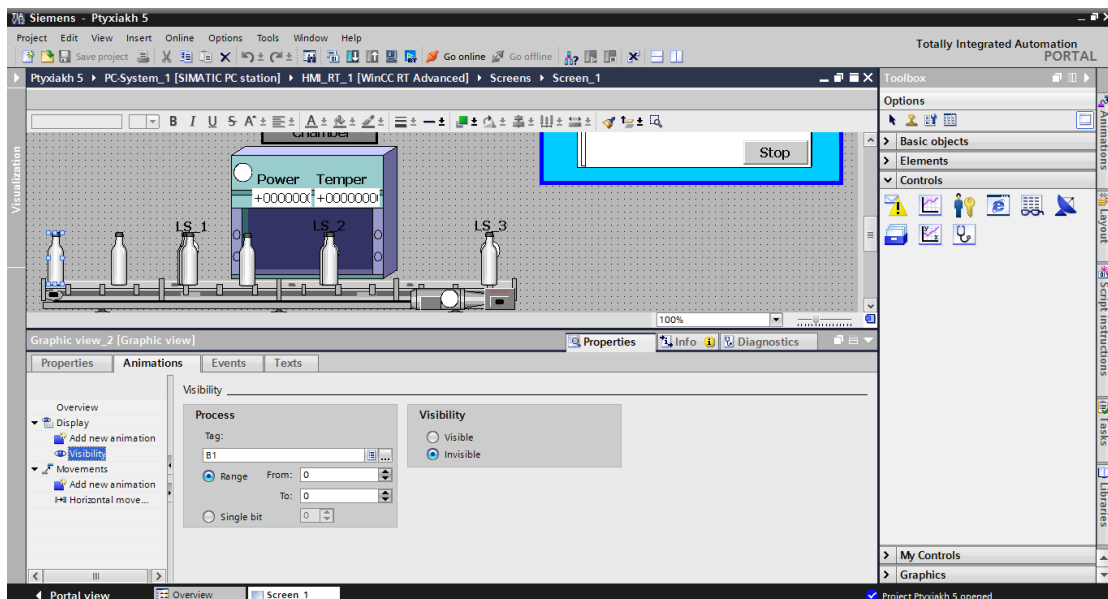


Σχήμα 4.42: Ορισμός κίνησης μπουκαλιού

Η κίνηση του μπουκαλιού έχει άμεση επικοινωνία με το πρόγραμμα του PLC, για αυτό πρέπει να δοθεί διεύθυνση στα γραφικό στοιχείο. Στο πλαίσιο Process, στο κενό διάστημα της επιλογής Tag γίνεται δημιουργία HMI tag για την κίνηση του μπουκαλιού (Σχήμα 4.43).



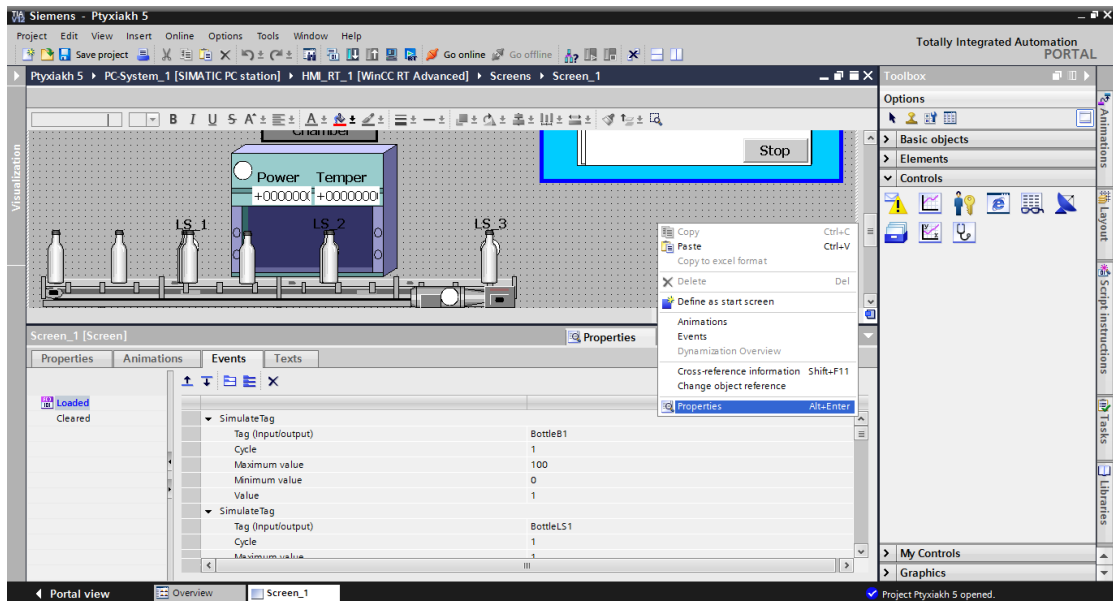
Σχήμα 4.43: Ορισμός κίνησης μπουκαλιού (Συνέχεια)



Σχήμα 4.44: Ορισμός κίνησης μπουκαλιού (Συνέχεια)

Επίσης στην επιλογή Visibility ορίζεται τότε θα εμφανίζεται το μπουκάλι με την κατάλληλη διεύθυνση(Σχήμα 4.44).

Έπειτα στην επιφάνεια εργασίας, πατάμε δεξί κλικ ώστε να αναδυθεί παράθυρο με διάφορες επιλογές. Γίνεται επιλογή Properties και στο αναδυόμενο παράθυρο επιλέγουμε Events. Στην επιλογή Loaded ορίζεται επιλογή SimulateTag και επιλέγεται το όνομα του HMI TAG που φτιάχτηκε νωρίτερα (Σχήμα 4.45). Έτσι σε κάθε ενεργοποίηση αυτού του tag θα έχει ως αποτέλεσμα την εμφάνιση του μπουκαλιού.



Σχήμα 4.45 : Ορισμός κίνησης μπουκαλιού (Συνέχεια)

Κεφάλαιο 5

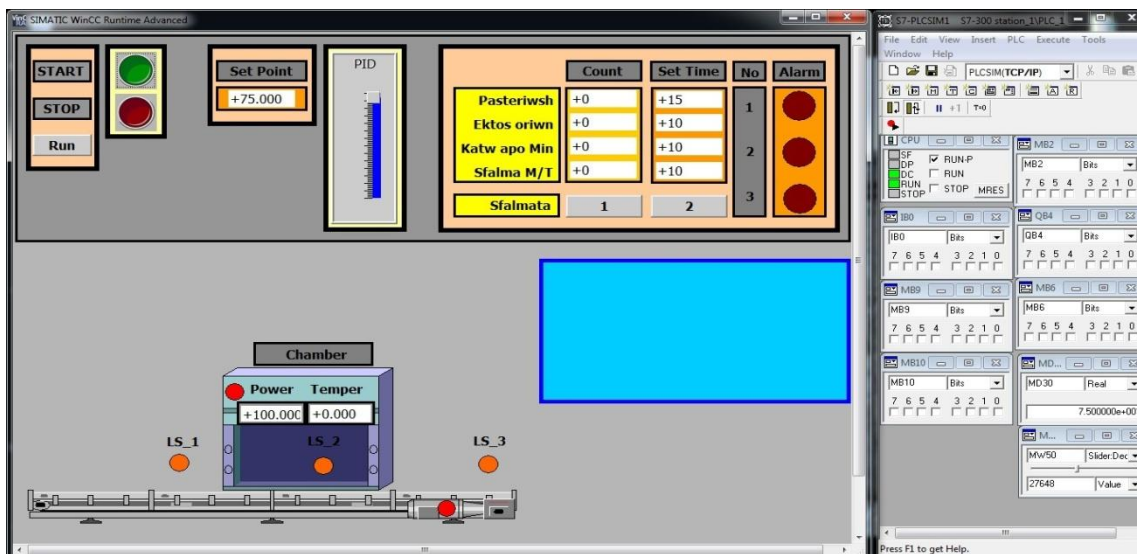
Αποτελέσματα – συμπεράσματα

5.1 Λειτουργία της διαδικασίας του σταθμού παστερίωσης γάλακτος

Στο σταθμό παστερίωσης γάλακτος που υλοποιήθηκε υποστηρίζονται μια σειρά σεναρίων που περιλαμβάνουν την κανονική λειτουργία της εφαρμογής καθώς και κάποια σφάλματα που μπορεί να προκύψουν. Τα σενάρια σφαλμάτων που εξετάζονται σχετίζονται με το αν η θερμοκρασία είναι εκτός ορίων, την πτώση της θερμοκρασίας κάτω από ένα όριο και την λειτουργία της μεταφορικής ταινίας. Μέσα από τα υποστηριζόμενα σενάρια σφαλμάτων εξετάζεται η αποτελεσματικότητα της εφαρμογής στις κρίσιμες καταστάσεις αλλά και η πορεία του γάλακτος για την ολοκλήρωση της διαδικασίας παστερίωσης.

5.1.1 Σενάριο 1 – Λειτουργία χωρίς σφάλματα

Εφόσον έχει γίνει η επιλογή του project και ο χρήστης έχει μεταβεί στην οθόνη human machine interface(HMI) θα πρέπει να κάνει πρώτα κάποιες αρχικές ρυθμίσεις. Αυτές οι ρυθμίσεις είναι ο ορισμός του set point του ελεγκτή, ο χρόνος της παστερίωσης καθώς επίσης και οι χρόνοι που θα ενεργοποιούνται τα σφάλματα (Σχήμα 5.1).

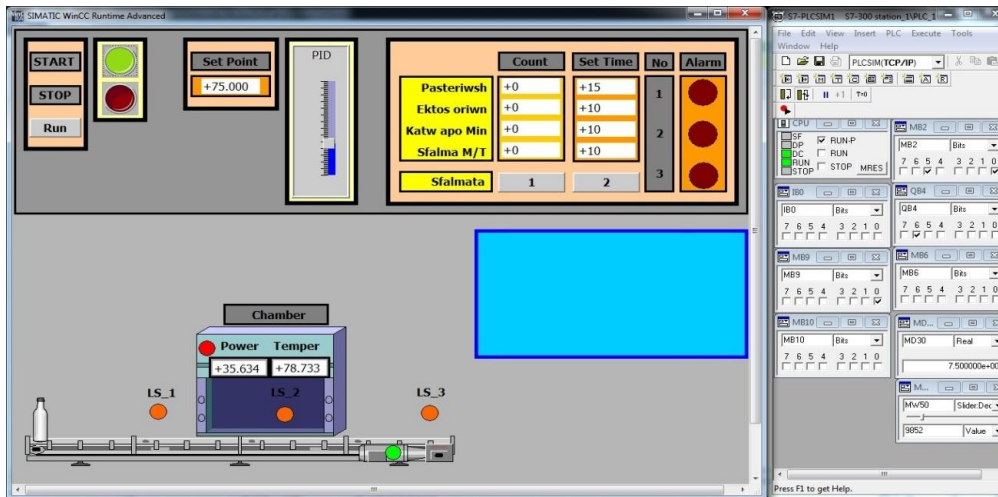


Σχήμα 5.1: Οθόνη HMI Controller και ρυθμίσεις εφαρμογής

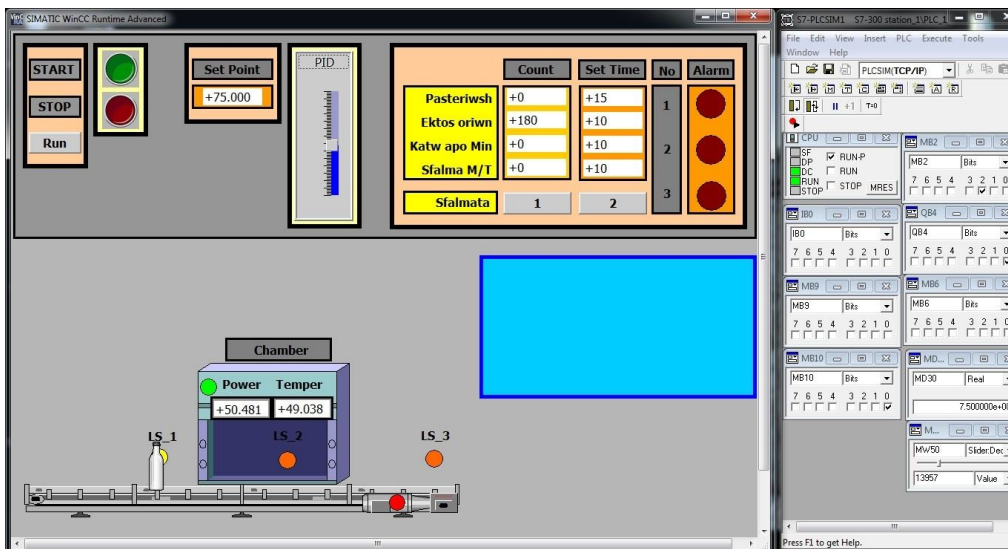
Η εκκίνηση της εφαρμογής είναι έτοιμη για λειτουργία μετά από τις κατάλληλες ρυθμίσεις. Για την εκκίνηση της εφαρμογής ο χρήστης θα πρέπει να κάνει χρήση του πράσινου κουμπιού Start (Σχήμα 5.2).

Τότε στην αρχή της μεταφορικής ταινίας (θέση 0) θα εμφανιστεί το μπουκάλι το οποίο θα κινηθεί προς την θέση LS_1, που είναι τοποθετημένος ο αισθητήρας αναγνώρισης κίνησης (Σχήμα 5.3).

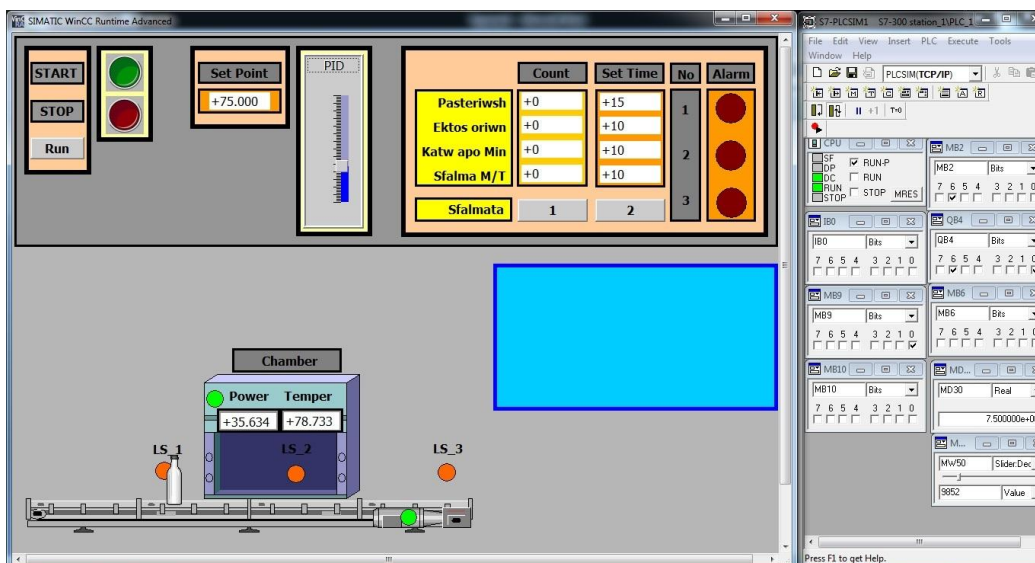
Σε εκείνο το σημείο, δηλαδή όταν το μπουκάλι βρίσκεται στον LS_1, είναι η πρώτη φάση της εφαρμογής. Ο φούρνος θα ενεργοποιηθεί και θα γίνει ρύθμιση της θερμοκρασίας ώστε το προϊόν να μεταβεί στην επόμενη φάση της εφαρμογής (Σχήμα 5.4)



Σχήμα 5.2: Εκκίνηση μπουκαλιού

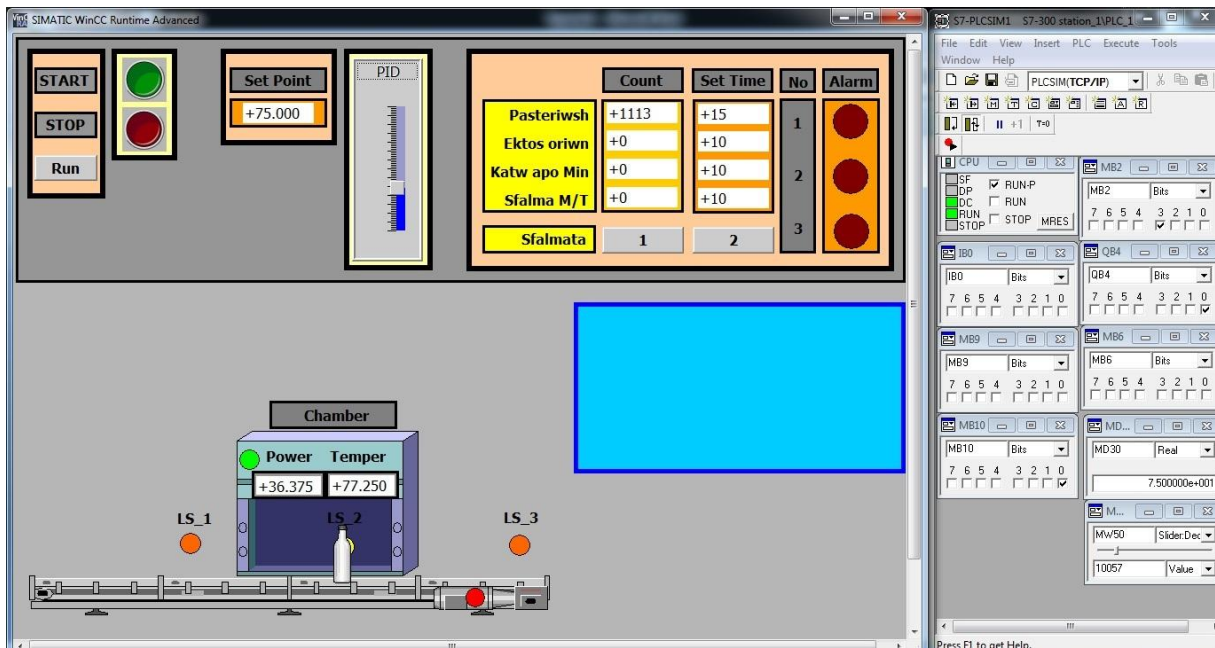


Σχήμα 5.3: Προϊόν στην θέση LS_1



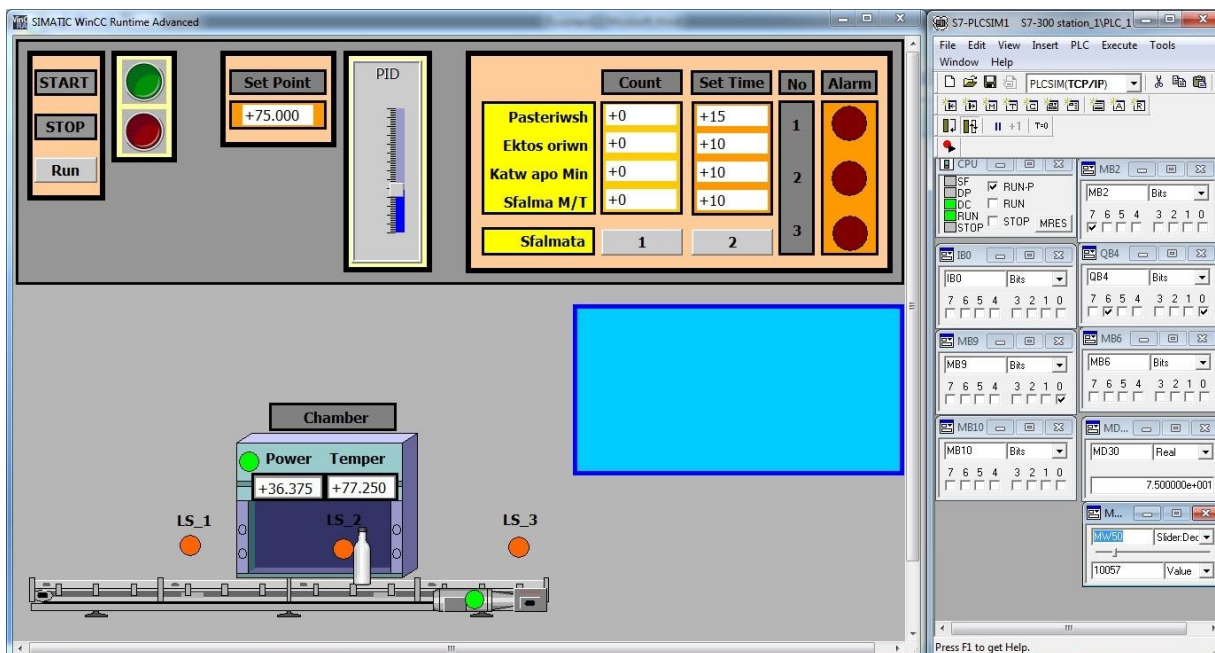
Σχήμα 5.4: Εκκίνηση μπουκαλιού για δεύτερη φάση

Μετά από το πέρας της πρώτης φάσης το προϊόν κατευθύνεται με πορεία προς την δεύτερη φάση, όπου είναι η παστερίωση του γάλακτος. Η διαδικασία της παστερίωσης ξεκινά όταν το προϊόν βρίσκεται στην θέση του LS_2 (Σχήμα 5.5).



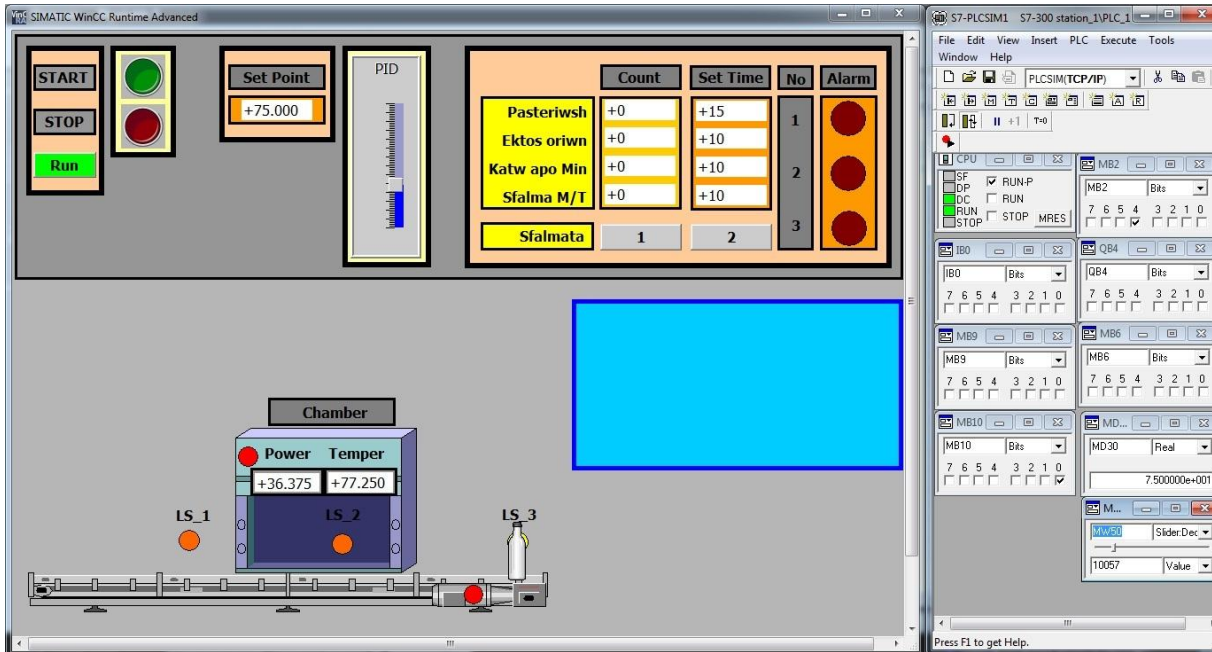
Σχήμα 5.5: Προϊόν στην θέση LS_2

Με την ολοκλήρωση του χρόνου της παστερίωσης και την ολοκλήρωση της δεύτερης φάσης το προϊόν είναι έτοιμο να μεταβεί στην τελική φάση της εφαρμογής (Σχήμα 5.6).



Σχήμα 5.6: Τέλος δεύτερης φάσης

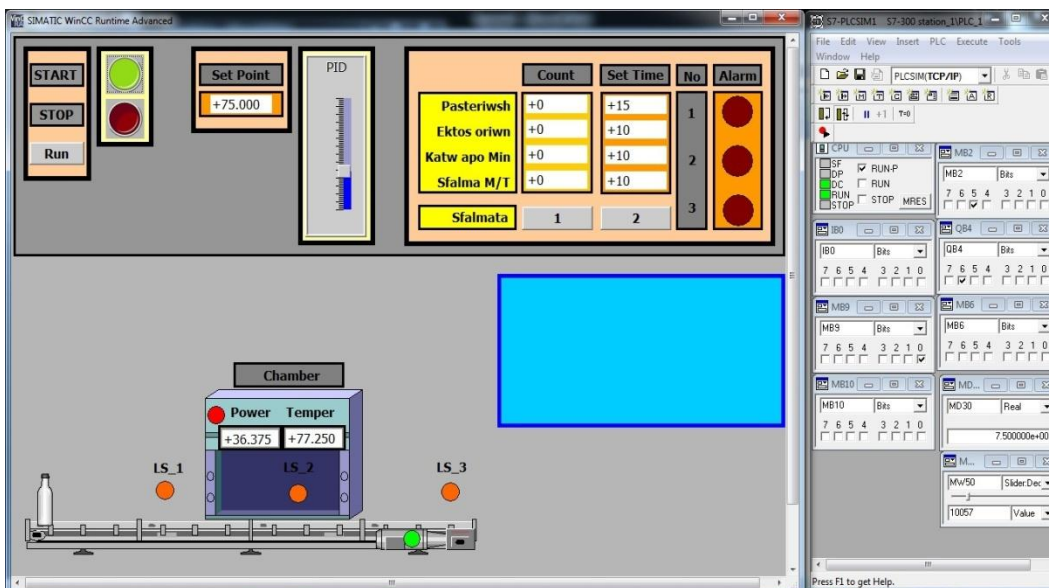
Η τελική διαδρομή που διαγράφει το προϊόν είναι από τον αισθητήρα LS_2 προς τον αισθητήρα LS_3. Όταν το προϊόν φθάσει στον αισθητήρα LS_3 ο φούρνος παστερίωσης απενεργοποιείται και στη συνέχεια η διαδικασία ολοκληρώνεται. Για την κατανόηση της ολοκλήρωσης της διαδικασίας έχει οριστεί από τον χρήστη για το πότε θα ξεκινάει ξανά η διαδικασία με την ενεργοποίηση του μπουτον Run (Σχήμα 5.7).



Σχήμα 5.7: Τέλος διαδικασίας παστερίωσης

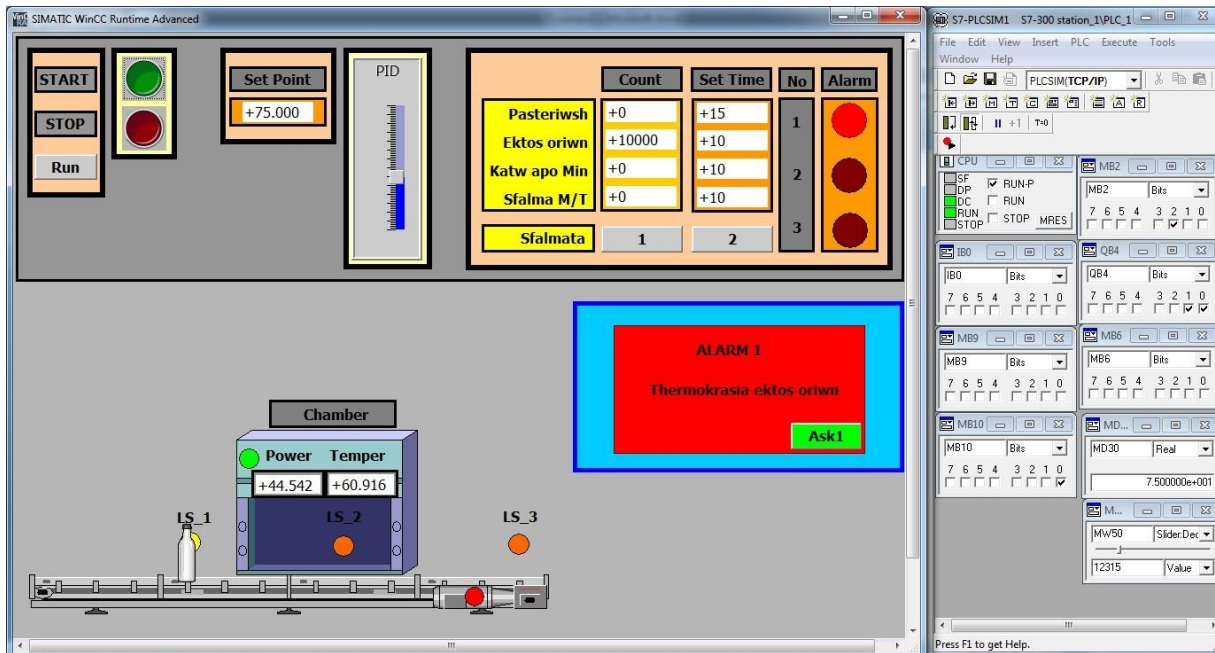
5.1.2 Σενάριο 2 – Σφάλμα 1, Θερμοκρασία εκτός ορίων

Το πρώτο σφάλμα αναφέρεται σε βλάβη στην διαδικασία της πρώτης φάσης. Με την εκκίνηση της εφαρμογής το προϊόν έχει αρχίσει να διαγράφει πορεία προς τον αισθητήρα LS_2 (Σχήμα 5.8)

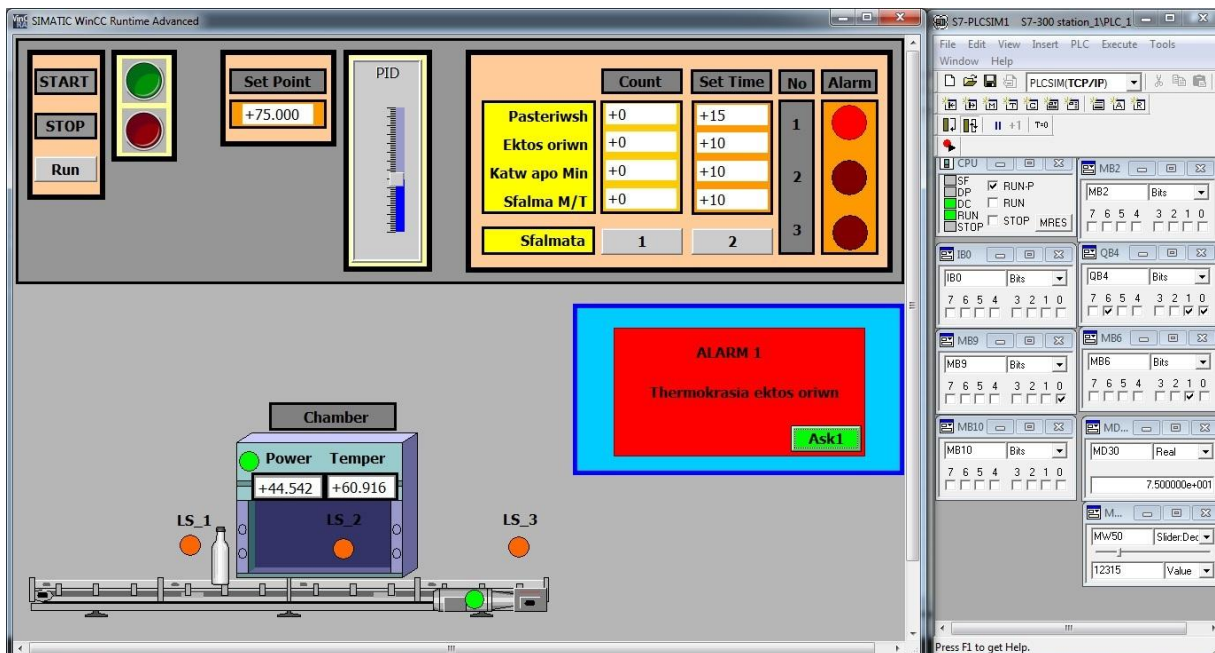


Σχήμα 5.8: Εκκίνηση μπουκαλιού

Όταν το προϊόν φθάσει στο αισθητήρα LS_2 τότε εντός ενός ορισμένου χρόνου θα πρέπει να έχει ρυθμιστεί η θερμοκρασία του φούρνου ώστε το προϊόν να μεταβεί στην επόμενη φάση. Στην περίπτωση που δεν ρυθμιστεί η θερμοκρασία σε κατάλληλο χρόνο δημιουργείται σφάλμα και ενεργοποιείται alarm-1. Με την ενεργοποίηση του alarm-1 αναδύεται ένα παράθυρο στην οθόνη HMI, όπου περιγράφει το είδος του σφάλματος καθώς και τον αριθμό του σφάλματος(Σχήμα 5.9)

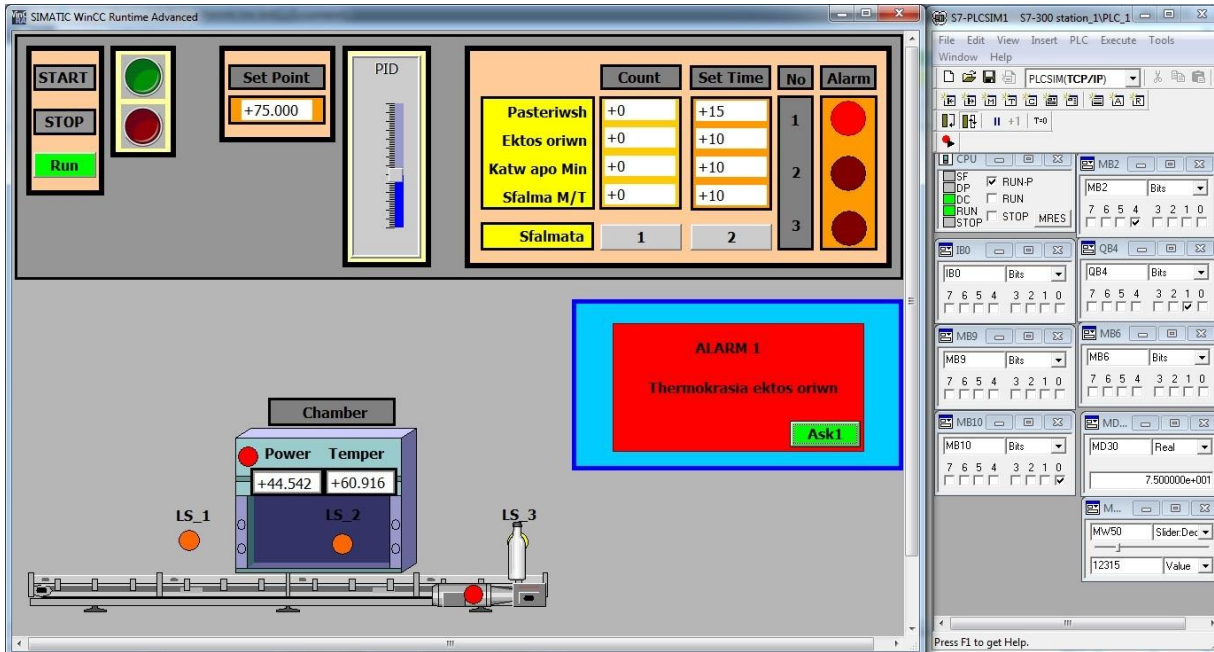


Σχήμα 5.9: Παρουσίαση Alarm-1

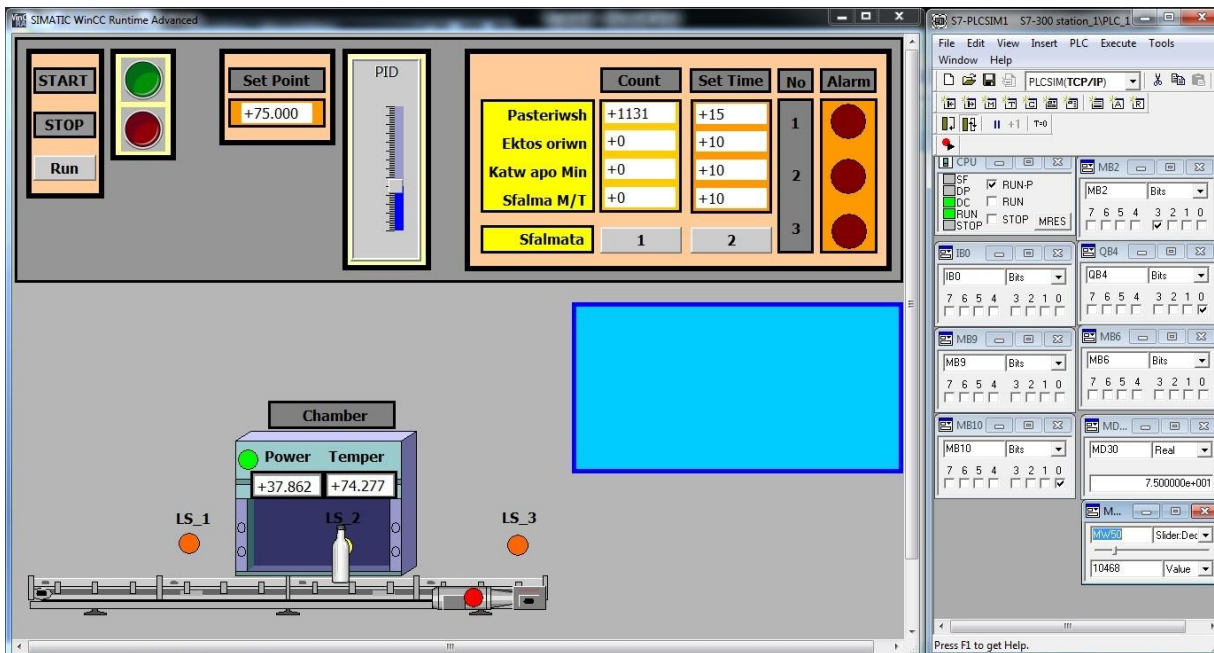


Σχήμα 5.10: Ενεργοποίηση του Ask1

Επίσης στο αναδυόμενο παράθυρο υπάρχει εντολή με όνομα Ask1, όπου ο χρήστης αναγνωρίζει το σφάλμα και με την ενεργοποίησή του συνεχίζει τη διαδικασία με το εσφαλμένο προϊόν προς τον τερματισμό της διαδικασίας παρακάμπτοντας όλες τις άλλες φάσεις (Σχήμα 5.10). Μετά από μικρό χρονικό διάστημα το εσφαλμένο προϊόν φθάνει στον αισθητήρα LS_3. Ο χρήστης σ αυτή την φάση μπορεί να τερματίσει την διαδικασία πατώντας STOP ή να ενεργοποιήσει τον διακόπτη Run ώστε να ξεκινήσει νέα διαδικασία (Σχήμα 5.11).



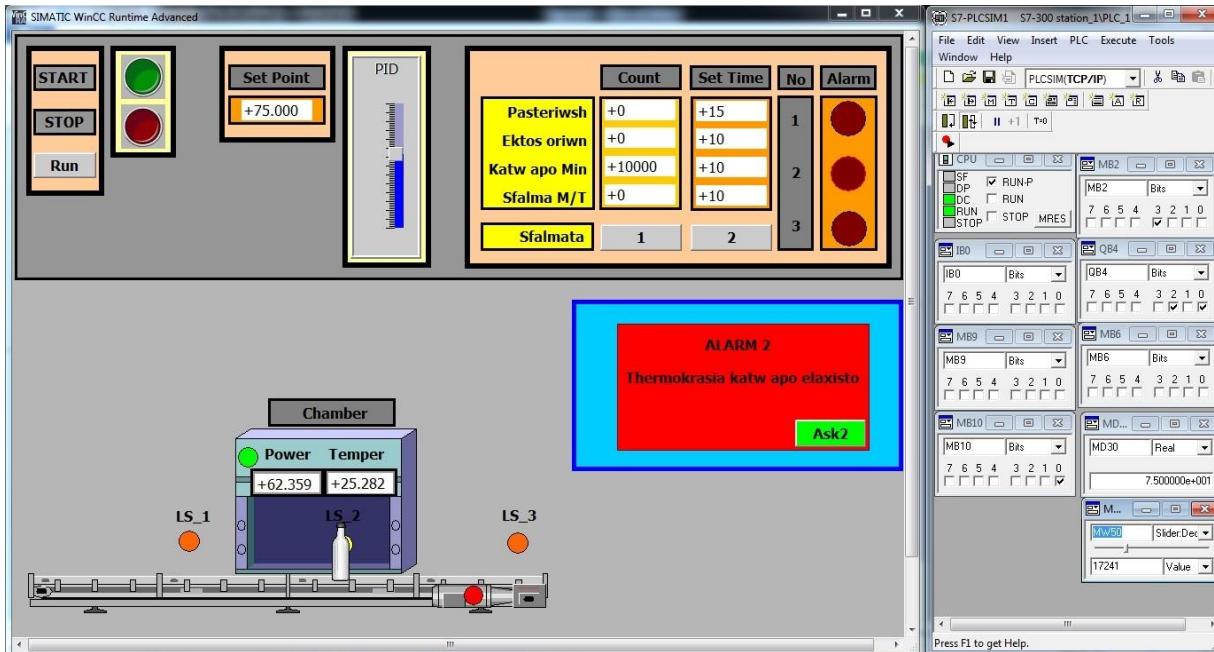
Σχήμα 5.11: Τέλος διαδικασίας



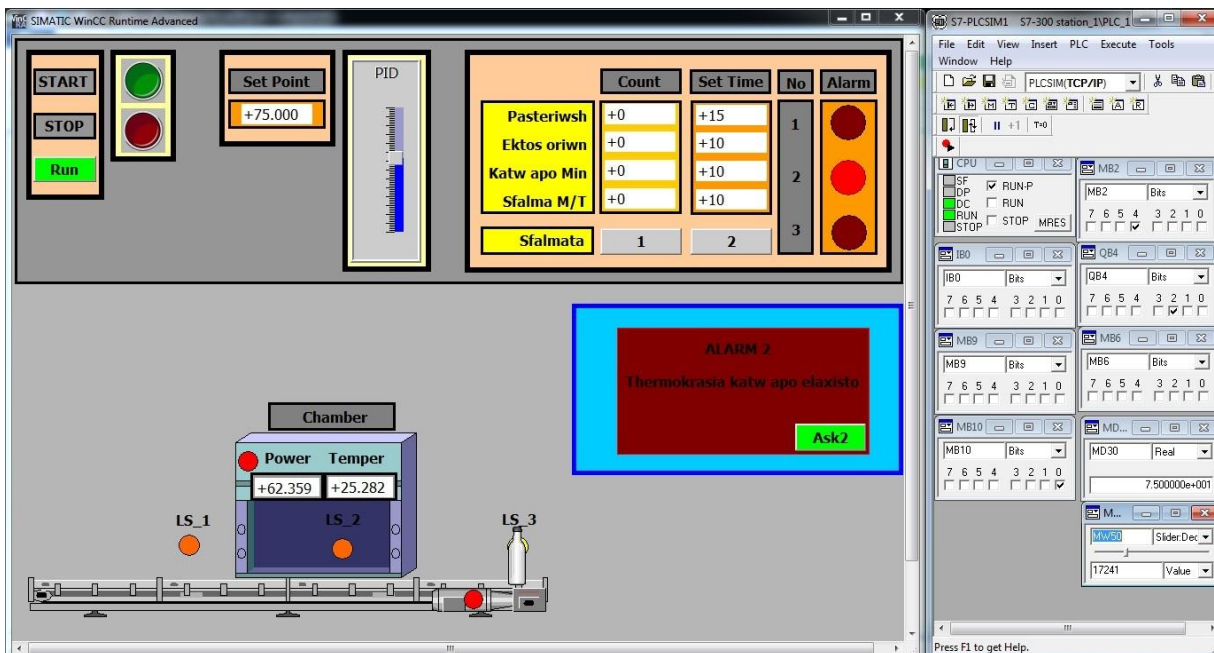
Σχήμα 5.12: Διαδικασία παστερίωσης

5.1.3 Σενάριο 3 – Σφάλμα 2, Πτώση της θερμοκρασίας κάτω από ένα όριο

Το δεύτερο σφάλμα αναφέρεται στη δεύτερη φάση της εφαρμογής, στο σημείο όπου πραγματοποιείται η παστερίωση. Το προϊόν διαγράφει την πορεία από τον αισθητήρα LS_1 προς τον αισθητήρα LS_2. Η διαδικασία της παστερίωσης ξεκινά μόλις το προϊόν φθάσει στον αισθητήρα LS_2 (Σχήμα 5.12). Κατά την διάρκεια της παστερίωσης η θερμοκρασία μειώνεται, μόλις πέσει κάτω από ένα όριο και μετά από ορισμένο χρόνο τότε ενεργοποιείται Alarm-2 (Σχήμα 5.13). Στο αναδυόμενο παράθυρο αναφέρεται το είδος του σφάλματος και ο αριθμός του σφάλματος.



Σχήμα 5.13: Παρουσίαση Alarm-2



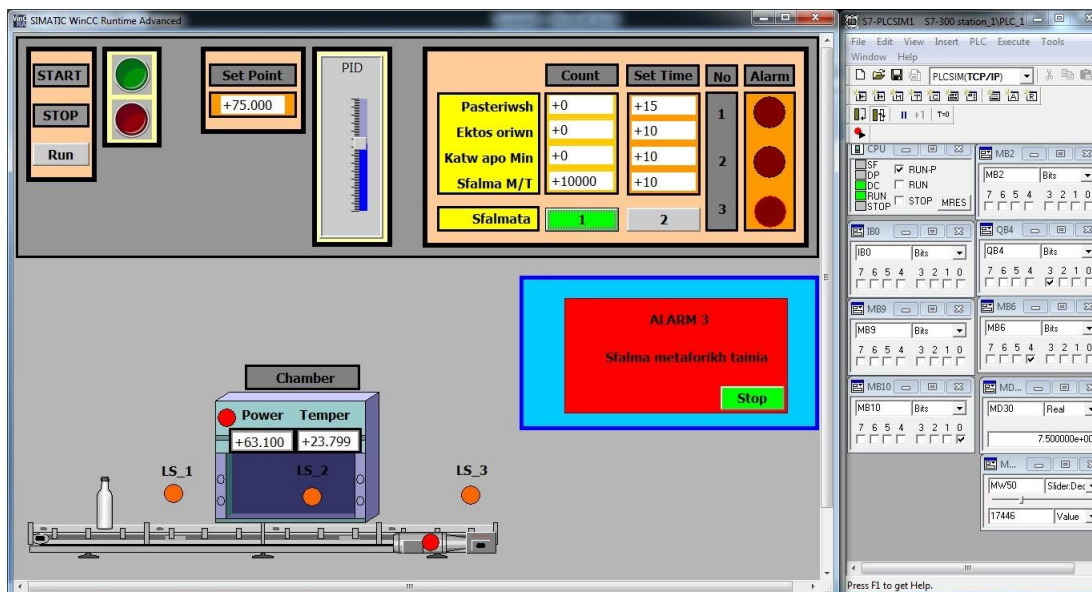
Σχήμα 5.14: Τέλος διαδικασίας

Επίσης στο αναδυόμενο παράθυρο η εντολή με όνομα Ask2, είναι για την αναγνώριση του σφάλματος αλλά δίνει την δυνατότητα στον χρήστη να την ενεργοποιήσει ώστε να συνεχίσει τη διαδικασία με το εσφαλμένο προϊόν προς τον τερματισμό της διαδικασίας (Σχήμα 5.13).

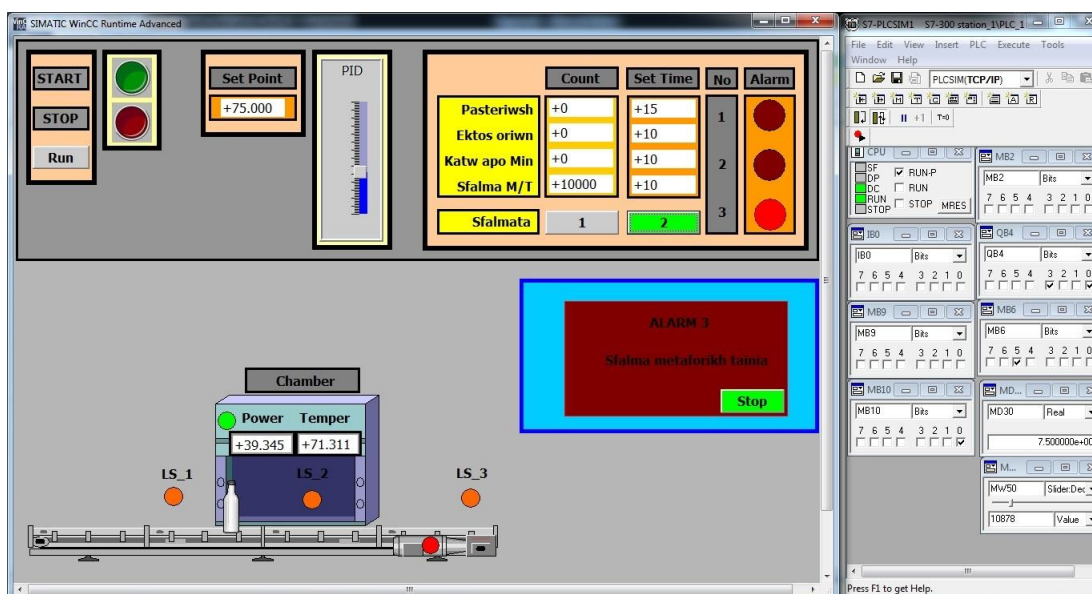
Μετά από μικρό χρονικό διάστημα το εσφαλμένο προϊόν φθάνει στον αισθητήρα LS_3. Ο χρήστης σ αυτή την φάση μπορεί να τερματίσει την διαδικασία πατώντας STOP ή να ενεργοποιήσει τον button Run ώστε να ξεκινήσει νέα διαδικασία (Σχήμα 5.14).

5.1.4 Σενάριο 4 – Σφάλμα 3, Σφάλμα στην μεταφορική ταινία

Το τρίτο σφάλμα αναφέρεται στην μεταφορική ταινία, όπου δημιουργείται βλάβη με αποτέλεσμα να σταματά κάθε διαδικασία που πραγματοποιείται. Έχουν οριστεί δύο σφάλματα στην μεταφορική ταινία, το πρώτο βρίσκεται στη φάση 1 ενώ το δεύτερο βρίσκεται στη φάση 2 (Σχήματα 5.15 και 5.16).



Σχήμα 5.15: Παρουσίασης σφάλματος Alarm-3



Σχήμα 5.16: Παρουσίαση σφάλματος Alarm-3

Με την ενεργοποίησή του σφάλματος, στο αναδυόμενο παράθυρο η εντολή με όνομα Stop δίνει την δυνατότητα στον χρήστη να τερματίζει την εφαρμογή.

5.2 Συμπεράσματα – προτάσεις

Στα πλαίσια της παρούσας πτυχιακής εργασίας και με σκοπό την εξοικείωση με το λογισμικό πακέτο TIA PORTAL και την ανάπτυξη εφαρμογών αυτοματισμού με PLCs αναπτύχθηκε η εφαρμογή για την λειτουργία ενός σταθμού παστερίωσης γάλακτος. Μέσα από αυτή την διαδικασία αποκτήθηκαν γνώσεις που σχετίζονται με το προγραμματισμό των PLCs σε γλώσσα ladder και την ανάπτυξη ενός HMI.

Ο προγραμματισμός της παραπάνω εφαρμογής με χρήση της γλώσσας προγραμματισμού ladder είχε ως αποτέλεσμα την δημιουργία ενός προγράμματος μεγάλου σε έκταση που σε πολλές περιπτώσεις μπορεί να θεωρηθεί δυσανάγνωστο. Ο συνδυασμός και μιας άλλης γλώσσας προγραμματισμού όπως η STL θα βελτίωνε την δομή του προγράμματος και θα μείωνε την έκταση του. Επίσης η ανάπτυξη ορισμένων μπλοκ συναρτήσεων που θα αναλάμβαναν την υλοποίηση συγκεκριμένων διεργασιών θα είχε ως αποτέλεσμα τόσο την καλύτερη δόμηση της εφαρμογής όσο και την ευκολότερη αποσφαλμάτωση της εφαρμογής.

Για την βελτίωση των γραφικών στοιχείων στην οθόνη HMI θα μπορούσε να χρησιμοποιηθεί πρόγραμμα σχεδίασης γραφικών όπως το Photoshop. Έτσι τα γραφικά στοιχεία θα ήταν πιο ρεαλιστικά και περισσότερο φιλικά προς τον χρήστη. Επίσης η δημιουργία περισσότερων οθονών θα διαχώριζε και θα διευκόλυνε τις λειτουργίες για τον χρήστη. Για παράδειγμα για τον ορισμό του χρόνου των σφαλμάτων αλλά και του Set point στον ελεγκτή θα μπορούσε να δημιουργηθεί μια οθόνη ρυθμίσεων. Επίσης θα μπορούσε να δημιουργηθεί μια οθόνη σφαλμάτων στην οποία θα καταγράφονταν τα σφάλματα που προέκυψαν κατά την λειτουργία της εφαρμογής. Με τον τρόπο αυτό ο χρήστης θα είχε την δυνατότητα να παρατηρεί το ιστορικό των σφαλμάτων γεγονός που θα βοηθούσε στην βελτίωση της λειτουργίας μια πραγματικής εφαρμογής αυτοματισμού.

Τέλος, σε ένα πραγματικό σύστημα αυτοματισμού η καταμέτρηση των παραγόμενων προϊόντων ή και των εσφαλμένων προϊόντος θα επέτρεπε στον χειριστή να έχει μια ολοκληρωμένη εικόνα της παραγωγικής διαδικασίας και να προτείνει στους σχεδιαστές του συστήματος ορισμένες βελτιώσεις.

Βιβλιογραφία

- [1] Totally Integrated Automation Portal (TIA Portal) automation software, <http://www.controleng.com/single-article/ec-totally-integrated-automation-portal-tia-portal-automation-software/a5b0104425f02c9235b5d1155d9c5c2d.html> (Ανάκτηση 17/11/11).
- [2] H. Berger, Automating with SIMATIC S7-300 inside TIA Portal, Publicis Publishing, 2nd ed., 2014.
- [3] H. Berger, Automating with SIMATIC S7-400 inside TIA Portal, Publicis Publishing, 2013.
- [4] H. Berger, Automating with SIMATIC S7-1500 inside TIA Portal, Publicis Publishing, 2014.
- [5] Λιγνός Ιωάννης, Μπουσλής Πέτρος, Πολίτης Γεώργιος, Χαμηλοθώρης Γεώργιος Αυτοματισμοί & Συστήματα αυτόματου ελέγχου Β΄ τεύχος, Αθήνα 2002.
- [6] SIEMENS, Simatic TIA Portal Step 7 Basic V10.5, Getting Started, 2009.
- [7] Ανάπτυξη συστημάτων κεντρικού ελέγχου εγκαταστάσεων, <http://auto.teipir.gr/sites/default/files/plc.pdf>, (Ανάκτηση 17/12/11).
- [8] MITSUBISHI ELECTRIC, MELSEC FX Family, Programmable Logic Controllers, Beginners_Manual, 2007.
- [9] Προγραμματιζόμενοι Λογικοί Ελεγκτές (PLC), <https://eclass.duth.gr/modules/document/file.php/TME134/%CE%A0%CE%A1%CE%9F%CE%93%CE%A1%CE%91%CE%9C%CE%9C%CE%91%CE%A4%CE%99%CE%96%CE%9F%CE%9C%CE%95%CE%9D%CE%9F%CE%99%20%CE%9B%CE%9F%CE%93%CE%99%CE%9A%CE%9F%CE%99%20%CE%95%CE%9B%CE%95%CE%93%CE%9A%CE%A4%CE%95%CE%A3%20PLC.pdf>, (Ανάκτηση 17/12/13).
- [10] Σ. Ρουμπής, Αυτοματισμούς με προγραμματιζόμενους ελεγκτές, ΣΗΜΕΝΣ ΑΕ, Ιούνιος 1992.
- [11] SIEMENS, SIMATIC HMI, WinCC V12, Comfort Panels, Runtime Advanced – Advanced Configuration Options, Getting Started, May, 2013.
- [12] SIEMENS, SIMATIC HMI, WinCC V12, Runtime Professional, Getting Started, May, 2013.
- [13] Secomea, Siemens PLC and SIMATIC STEP 7 / TIA_Portal, Application Note, V4.5, January 2014.
- [14] SIEMENS, SIMATIC HMI, HMI devices, Basic Panels, Operating Instructions April 2012.
- [15] SIEMENS, SIMATIC HMI Panels, Efficient to a new level, brochure, https://www.automation.siemens.com/salesmaterial-as/brochure/en/brochure_panels_en.pdf, (Ανάκτηση 20/2/2016)
- [16] Ν. Μαραντίδης, Αυτοματισμός με Simatic S7, ΣΗΜΕΝΣ ΑΕ, 2000.
- [17] C. T. Jones, Step 7 in 7 Steps, 1st ed. 2006.

- [18]** J. R. Hackworth and F. D. Hackworth, Programmable Logic Controllers: Programming Methods and Applications, Pearson, 2003.
- [19]** W. Bolton, Programmable Logic Controllers, 5th ed., Newnes, 2009.
- [20]** K-H John and M. Tiegelkamp, IEC 61131-3: Programming Industrial Automation Systems, 2nd ed., Springer, 2010.
- [21]** PID controller, https://en.wikipedia.org/wiki/PID_controller, (Ανάκτηση 14/1/2016)
- [22]** SCADA, <https://en.wikipedia.org/wiki/SCADA>, (Ανάκτηση 20/1/2016).
- [23]** Programmable Logic Controllers, https://en.wikipedia.org/wiki/Programmable_logic_controller, (Ανάκτηση 25/2/2016)