

**ΑΤΕΙ ΠΕΙΡΑΙΑ**  
**ΤΜΗΜΑ ΑΥΤΟΜΑΤΙΣΜΟΥ**

Βιομηχανικός έλεγχος-Συγκριτική μελέτη  
μεταξύ κλασσικής μεθόδου βιομηχανικού  
ελέγχου (PLC) και συστήματος βασισμένο  
σε πρακτορικό σύστημα.

---

Industrial control-A comparison study of  
agent based and classic control approaches  
(PLC)

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

**ΑΝΑΓΝΩΣΤΟΠΟΥΛΟΣ ΑΘΑΝΑΣΙΟΣ 36820**

Επιβλέποντες Καθηγητές:  
Δρ. Αλαφοδήμος Κωνσταντίνος  
Δρ. Νικολάου Γρηγόριος

Πειραιάς, Ιούλιος 2013

## Ευχαριστίες

Με την ολοκλήρωση της παρούσας πτυχιακής εργασίας, ένα μεγάλο κεφάλαιο κλείνει. Ένα κεφάλαιο που μου έδωσε πολύ σημαντικά εφόδια για να με βοηθήσει στην συνέχεια της πορείας μου. Αισθάνομαι ότι οφείλω, στο σημείο αυτό, να ευχαριστήσω τα άτομα που με βοήθησαν για την ολοκλήρωση αυτής της εργασίας.

Θα ήθελα να ευχαριστήσω θερμά τους κ. Νικολάου Γρηγόρη και κ. Αλαφοδήμο Κωνσταντίνο, Καθηγητές του τμήματος Αυτοματισμού του Ανώτατου Τεχνολογικού Εκπαιδευτικού Ιδρύματος Πειραιά για την επίβλεψη της εργασίας και τις πολύτιμες συμβουλές τους καθώς και την υπομονή τους και εμπιστοσύνη που έδειξαν στο πρόσωπο μου.

Επίσης θα ήθελα να ευχαριστήσω θερμά τους κ. Niels Lohse, κ. Pedro Ferreira και κ. Στέφανο Δολτσίνη, για την διάθεση του εργαστηρίου αυτοματισμού του Πανεπιστημίου του Νότινγχαμ μαζί με τον σύγχρονο εξοπλισμό που περιλάμβανε αυτό, καθώς και για την υποστήριξη που μου προσέφεραν σε όλη την διάρκεια της ετοιμασίας αυτής της εργασίας.

Τέλος θα ήθελα να ευχαριστήσω και να αφιερώσω αυτή την εργασία στα άτομα που πάντα είναι κοντά μου, με στηρίζουν και πιστεύουν σε μένα.

## ΠΕΡΙΕΧΟΜΕΝΑ

Λίστα εικόνων.....	4
Λίστα συντομογραφιών .....	6
Λίστα δημοσιεύσεων.....	7
1. Εισαγωγή .....	8
2. Θεωρητικό υπόβαθρο .....	9
2.1 Ελεγκτές.....	9
2.1.1 Ιστορική αναδρομή .....	9
2.1.2 Περιγραφή και λειτουργία του PLC.....	11
2.1.3 Πλεονεκτήματα .....	13
2.1.4 Δυνατότητες και βελτιώσεις .....	14
2.1.5 Επικοινωνία και διασύνδεση .....	14
2.1.6 Γλώσσες προγραμματισμού .....	16
2.1.7 Χρόνος επεξεργασίας .....	17
2.1.8 Επικοινωνία με τον χρήστη .....	17
2.2 Θεωρητικό υπόβαθρο πρακτορικού συστήματος .....	19
3. Παρουσίαση πειραματικού εξοπλισμού-(Hardware σταθμών) .....	21
3.1 Το σύστημα .....	21
3.1.1 1 <sup>ος</sup> σταθμός (διαφορετικά χρώματα) .....	22
3.1.2 2 <sup>ος</sup> σταθμός (σταθμός παραγωγής).....	23
3.1.3 3 <sup>ος</sup> σταθμός (σταθμός ελέγχου ύψους).....	26
3.1.4 4 <sup>ος</sup> σταθμός (σταθμός τοποθέτησης καλύμματος και ετικέτας).....	28
3.1.5 5 <sup>ος</sup> σταθμός (σταθμός αποστολής προϊόντων) .....	31
3.2 Περιφερειακά-Βοηθητικά μέρη .....	32
3.2.1 Φώτα .....	33
3.2.2 Κουμπιά.....	34
3.3 Beckhoff.....	35

3.3.1	Controllers .....	35
3.3.2	Modules .....	36
3.3.3	Twincat Software .....	38
4.	Λειτουργία σταθμών .....	42
4.1	Σχεδίαση και εφαρμογή κώδικα ελέγχου .....	42
4.1.1	1ος σταθμός .....	42
4.1.2	2ος σταθμός .....	49
4.1.3	3ος σταθμός .....	57
4.1.4	4ος σταθμός .....	63
4.1.5	5ος σταθμός .....	70
5.	Σχεδίαση και εφαρμογή σε πρακτορικό σύστημα .....	76
5.1.	Ορισμός προβλήματος και αιτιολόγηση επιλογής της χρήσης πρακτόρων .....	76
5.2.	Διαμόρφωση σταθμού με πρακτορικό σύστημα .....	77
5.3.	Λογισμικό διαχείρισης πρακτορικού συστήματος .....	77
5.4.	Η αρχιτεκτονική της επικοινωνίας του πρακτορικού συστήματος με το PLC .....	79
5.5.	Σενάρια πειραμάτων .....	83
6.	Αποτελέσματα και συμπεράσματα .....	85
6.1.	Συμπεράσματα .....	88
	Βιβλιογραφία .....	89

## Λίστα εικόνων

Εικόνα 1 – ο εφευρέτης των PLC .....	10
Εικόνα 2 - PLC Allen Bradley .....	14
Εικόνα 3 - Τα μέρη ενός PLC .....	15
Εικόνα 4 – Γλώσσα Ladder .....	16
Εικόνα 5 – οθόνη ελέγχου PLC .....	18
Εικόνα 6 - Τύποι πρακτόρων .....	20
Εικόνα 7 – απεικόνιση κάποιων σταθμών του συστήματος .....	21
Εικόνα 8 – 1 <sup>ος</sup> σταθμός .....	22
Εικόνα 9 – αναγνώστης Barcode .....	23
Εικόνα 10 – 2 <sup>ος</sup> σταθμός .....	24
Εικόνα 11 - ζυγαριά .....	25
Εικόνα 12 – υλικό για γέμιση κουτιών .....	26
Εικόνα 13 - 3 <sup>ος</sup> σταθμός .....	27
Εικόνα 14 – ιμάντες μεταφοράς .....	27
Εικόνα 15 – αναλογικός μετρητής ύψους .....	28
Εικόνα 16 4 <sup>ος</sup> σταθμός .....	29
Εικόνα 17 – εκτυπωτής ετικετών RS-232 .....	30
Εικόνα 18 - 5 <sup>ος</sup> σταθμός .....	31
Εικόνα 19 – πλατφόρμα αποστολής .....	32
Εικόνα 20 – διακόπτες ελέγχου .....	34
Εικόνα 21 – Ελεγκτής Beckhoff CX-1030 .....	35
Εικόνα 22 – EL1008 .....	36
Εικόνα 23 – EL2008 .....	37
Εικόνα 24 – EL3102 .....	38
Εικόνα 25 – System manager .....	39
Εικόνα 26 – PLC control .....	40
Εικόνα 27 – PLC control 2 .....	41
Εικόνα 28 – 1 <sup>ος</sup> σταθμός (1/4) .....	42
Εικόνα 29 - 1 <sup>ος</sup> σταθμός (2/4) .....	43
Εικόνα 30 - 1 <sup>ος</sup> σταθμός (3/4) .....	44
Εικόνα 31 - 1 <sup>ος</sup> σταθμός (4/4) .....	45
Εικόνα 32 – διαδικασία για άδεια κουτιά (αριστερά) – διαδικασία για κόκκινο φως (δεξιά) .....	46
Εικόνα 33 – Στοπ ασφαλείας (αριστερά) – Επιστροφή σε αρχική θέση (δεξιά) .....	47
Εικόνα 34 - 2 <sup>ος</sup> σταθμός (1/5) .....	49
Εικόνα 35 - 2 <sup>ος</sup> σταθμός (2/5) .....	50
Εικόνα 36 - 2 <sup>ος</sup> σταθμός (3/5) .....	51
Εικόνα 37 - 2 <sup>ος</sup> σταθμός (4/5) .....	52
Εικόνα 38 - 2 <sup>ος</sup> σταθμός (5/5) .....	53
Εικόνα 39 - διαδικασία για άδεια κουτιά (αριστερά) – διαδικασία για κόκκινο φως (δεξιά) .....	54
Εικόνα 40 - Στοπ ασφαλείας (αριστερά) – Επιστροφή σε αρχική θέση (δεξιά) .....	55
Εικόνα 41 - 3 <sup>ος</sup> σταθμός (1/2) .....	57
Εικόνα 42 - 3 <sup>ος</sup> σταθμός (2/2) .....	58
Εικόνα 43 - διαδικασία για κόκκινο φως .....	59
Εικόνα 44 - διαδικασία για ουρά στον επόμενο σταθμό .....	60
Εικόνα 45 – Στοπ ασφαλείας (αριστερά) – Επιστροφή σε αρχική θέση (δεξιά) .....	61
Εικόνα 46 - 4 <sup>ος</sup> σταθμός (1/4) .....	63
Εικόνα 47 - 4 <sup>ος</sup> σταθμός (2/4) .....	64
Εικόνα 48 - 4 <sup>ος</sup> σταθμός (3/4) .....	65

Εικόνα 49 - 4 <sup>ος</sup> σταθμός (4/4).....	66
Εικόνα 50 - διαδικασία για άδεια καπάκια (αριστερά) – διαδικασία για κόκκινο φως (δεξιά).....	67
Εικόνα 51 – Στοπ ασφαλείας (αριστερά) – Επιστροφή σε αρχική θέση (δεξιά) .....	68
Εικόνα 52 - διαδικασία για ουρά στον επόμενο σταθμό .....	68
Εικόνα 53 - 5 <sup>ος</sup> σταθμός (1/2).....	70
Εικόνα 54 - 5 <sup>ος</sup> σταθμός (2/2).....	71
Εικόνα 55 - διαδικασία για κόκκινο φως.....	72
Εικόνα 56 - διαδικασία για ουρά στον επόμενο σταθμό .....	73
Εικόνα 57 – Στοπ ασφαλείας (αριστερά) – Επιστροφή σε αρχική θέση (δεξιά) .....	74
Εικόνα 58 – Διαδικασία παραγωγής .....	77
Εικόνα 59 - NetBeans.....	78
Εικόνα 60 – Υβριδική αρχιτεκτονική.....	80
Εικόνα 61 – Περιπτώσεις διαπραγματεύσεων μεταξύ πρακτόρων .....	81
Εικόνα 62 - Πρωτόκολλο επικοινωνίας .....	82
Εικόνα 63 - καθυστερήσεις στην διαδικασία λήψης απόφασης.....	82
Εικόνα 64 - Πλήρης κύκλος προϊόντος χωρίς καθυστερήσεις.....	85
Εικόνα 65 - Χρόνος διαπραγματεύσεων χωρίς καθυστερήσεις (no delays) .....	86
Εικόνα 66 - Χρόνος διαπραγματεύσεων χωρίς καθυστερήσεις (20ms delays αριστερά, 50ms δεξιά)..	86
Εικόνα 67 - Χρόνος διαπραγματεύσεων χωρίς καθυστερήσεις (100ms delays).....	87

## Λίστα συντομογραφιών

**PLC** - Programmable logic controller

**FBD** - Function Block Diagram

**LD** - Ladder Diagram

**ST** - Structured Text

**IL** - Instruction List

**SFC** - Sequential Function Chart

**JADE** - Java Agent Development Framework

**ADS** - Automation Device Specification

**PA** - Product Agent

**RA** - Resource Agent

## Λίστα δημοσιεύσεων

- Ferreira P., Doltsinis S., Anagnostopoulos A., Pascoa F., Lohse N., **"A Performance evaluation of Industrial Agents - A Benchmark against Programmable Logic Controllers"**, IECON 2013, Vienna 10-13/11/2013 - The 39th Annual Conference of the IEEE Industrial Electronics Society, **Accepted**



## 1. Εισαγωγή

Σε αυτή την πτυχιακή παρουσιάζεται ο έλεγχος των παραγωγικών διεργασιών μέσω συστήματος βασισμένο σε σύστημα πρακτόρων και συγκρίνεται με την κλασσική PLC προσέγγιση. Στην συνέχεια παρουσιάζεται μία συγκριτική μελέτη μεταξύ των προαναφερθέντων προσεγγίσεων με σκοπό να εντοπιστούν τα πλεονεκτήματα και τα μειονεκτήματα τους όπως και οι δυνατότητες τους.

Η εργασία αποτελείται από τέσσερα κύρια μέρη. Στο πρώτο μέρος υπάρχει το θεωρητικό υπόβαθρο της τεχνολογίας των ελεγκτών PLC καθώς των πρακτορικών συστημάτων. Στο δεύτερο μέρος γίνεται η παρουσίαση των σταθμών που αποτελούν το σύστημα SMC HAS-200 το οποίο είναι μία γραμμή παραγωγής μικρής κλίμακας η οποία χρησιμοποιεί ηλεκτροπνευματική τεχνολογία. Σε αυτό το μέρος γίνεται ανάλυση όλων των επιμέρους μερών και συσκευών του συστήματος καθώς επίσης και των περιφερειακών και βοηθητικών συσκευών του. Στην συνέχεια αναφέρεται ο κώδικας που έχει γραφεί για την λειτουργία των σταθμών και όλες οι λειτουργίες του. Στο τρίτο μέρος αναλύεται η σχεδίαση και η εφαρμογή του κώδικα στο πρακτορικό σύστημα, η λογική που χρησιμοποιείται, η επικοινωνία μεταξύ του PLC και του πρακτορικού συστήματος στην υβριδική αυτή προσέγγιση καθώς και τα σενάρια των πειραμάτων που εκτελέστηκαν. Στο τέταρτο μέρος βλέπουμε την ανάλυση των αποτελεσμάτων που πήραμε, την σύγκριση μεταξύ των δύο τεχνολογιών καθώς επίσης και τα συμπεράσματα στα οποία οδηγούμαστε.

## 2. Θεωρητικό υπόβαθρο

### 2.1 Ελεγκτές

Κάθε σύστημα ή μηχανή έχει τουλάχιστον έναν ελεγκτή. Αναλόγως το είδος της τεχνολογίας που χρησιμοποιείται, οι ελεγκτές μπορούν να χωριστούν σε κατηγορίες πνευματικών, υδραυλικών, ηλεκτρικών ή ηλεκτρονικών ελεγκτών. Συχνά χρησιμοποιείται και συνδυασμός αυτών των τεχνολογιών. Επίσης υπάρχει διαχωρισμός ανάμεσα στους ελεγκτές που είναι προγραμματισμένοι εκ των προτέρων και σε αυτούς που προγραμματίζονται ξανά και ξανά από τον χρήστη. Οι πρώτοι χρησιμοποιούνται σε εφαρμογές που ο χρήστης δεν έχει την δυνατότητα ή την γνώση να τους προγραμματίσει όπως σε πλυντήρια, ψηφιακές μηχανές ή αυτοκίνητα.

Στην περίπτωση όμως που ο χρήστης έχει την δυνατότητα ή και την ανάγκη να κάνει αλλαγές στο ίδιο το πρόγραμμα ή σε μετρητές τότε χρησιμοποιείται το είδος ελεγκτή όπου το πρόγραμμα γράφεται στη μνήμη του και είναι διαθέσιμο για επεξεργασία. Ένα τέτοιο είδος ελεγκτή είναι και τα PLC όπου δίνουν την δυνατότητα στον χρήστη για οποιεσδήποτε αλλαγές.

#### 2.1.1 Ιστορική αναδρομή

Ήδη από τη δεκαετία του '60 στην Ευρώπη άρχισε η μετάβαση στα συστήματα με ψηφιακά ηλεκτρονικά. Αυτό δεν άλλαξε μόνο τον τρόπο σκέψης των κατασκευαστών αλλά και τη δομή και τον τρόπο λειτουργίας εγκαταστάσεων και μηχανών. Υπήρξαν όμως αρνητικά σημεία αφού απαιτήθηκε η γνώση υψηλής ηλεκτρονικής για τη σωστότερη εγκατάσταση και συντήρησή τους.

Οι πρώτοι προγραμματιζόμενοι λογικοί ελεγκτές (PLC – Programmable Logic Controllers) στην αρχή της δεκαετίας του '70 χρησιμοποιήθηκαν κυρίως για την αντικατάσταση των ρελέ.

Το πρώτο PLC αναπτύχθηκε από μια ομάδα μηχανικών της General Motors το 1968 [Εικόνα 1], όταν η εταιρία έψαχνε μια εναλλακτική για να αντικαταστήσει τα μέχρι τότε πολύπλοκα συστήματα ελέγχου που αποτελούνταν από ηλεκτρικά ρελέ.

Το καινούργιο σύστημα ελέγχου έπρεπε να πληροί κάποιες βασικές προϋποθέσεις όπως ο απλός προγραμματισμός αλλά και να αλλάζει το πρόγραμμα χωρίς να χρειάζεται να γίνεται παρέμβαση στις καλωδιώσεις του συστήματος.

Επίσης έπρεπε να είναι μικρότερο σε μέγεθος, φθηνότερο και πιο αξιόπιστο σε σχέση με τα υπάρχοντα συστήματα με ηλεκτρικά ρελέ. Ακόμα έπρεπε να είναι απλό στην δομή και λειτουργία του και να έχει χαμηλό κόστος συντήρησης.



Εικόνα 1 – ο εφευρέτης των PLC

Η συνεχής ανάπτυξη οδήγησε σε ένα σύστημα το οποίο επέτρεπε την απλή σύνδεση δυαδικών σημάτων. Οι προϋποθέσεις για το πώς αυτά τα σήματα έπρεπε να συνδεθούν προσδιορίζονταν στο πρόγραμμα ελέγχου. Με το νέο σύστημα, ήταν πλέον εφικτό για πρώτη φορά να σχεδιαστούν σήματα σε μία οθόνη και να αποθηκευτούν σε μνήμες ηλεκτρονικού τύπου.

Από τότε έχουν περάσει τέσσερις δεκαετίες κατά τις οποίες έχει γίνει τεράστια πρόοδος στην ανάπτυξη των ηλεκτρονικών εξαρτημάτων και συνεπώς και των PLC. Το εύρος των λειτουργιών έχει μεγαλώσει αισθητά. Πριν αρκετά χρόνια η οπτικοποίηση των διαδικασιών, η αναλογική επεξεργασία ή ακόμα και η χρήση του PLC ως ελεγκτή ήταν ένα άπιαστο όνειρο. Πλέον όμως όλες αυτές οι λειτουργίες και ακόμα περισσότερες, αποτελούν ένα βασικό κομμάτι των περισσότερων PLC.

Η μεγάλη απαίτηση για μείωση του κύκλου παραγωγής άρχισε στη αρχή της δεκαετίας του '80 .Η τεχνολογία γινόταν γρηγορότερη και αναπτυσσόταν συνεχώς , παράλληλα με τις απαιτήσεις του χρήστη .Όπως σε όλους τους τομείς έτσι κι εδώ , η επικοινωνία και η πληροφορία έγιναν η σημαντικότερη βάση για αποδοτική παραγωγή .Οι νέες συσκευές επεξεργάζονται πλέον δεδομένα και ανταλλάσσουν πληροφορίες μεταξύ τους με υπερκείμενους υπολογιστές.

Οι διαδικασίες παραγωγής γίνονται πιο σύνθετες , οι νεκροί χρόνοι στην παραγωγή μειώνονται συνεχώς , οι απαιτήσεις για αυξημένη ποιότητα αυξάνονται . Αλλάζει και ο ρόλος του ανθρώπου στην παραγωγική διαδικασία - τώρα σχεδιάζει , κατασκευάζει ,προγραμματίζει επιτηρεί και επισκευάζει. Το φάσμα της εργασίας μεταφέρεται από τη μυϊκή στην πνευματική.

Κι ενώ η τεχνολογία προχωρά, φθάνουμε στη δεκαετία του '90 όπου τεχνολογικά έγινε μεγάλο άλμα (συσκευές μικρότερες, φθηνότερες, με σημαντικά αυξημένες δυνατότητες συγκριτικά με αυτές της προηγούμενης δεκαετίας) αλλά παράλληλα αυξήθηκε δυσανάλογα το κόστος εκπόνησης προγραμμάτων και της θέσης σε λειτουργία των εγκαταστάσεων.

Οι κατασκευαστές ρίχνουν πλέον σημαντικό βάρος στο λογισμικό όπου παρέχονται έτοιμες λύσεις για τομείς του αυτοματισμού με τη βοήθεια βιβλιοθηκών, εκμεταλλεύονται την πρόοδο των ηλεκτρονικών υπολογιστών και χρησιμοποιούν την εξέλιξη στο λειτουργικό τους σύστημα (τεχνολογία Windows) για να μειώσουν τους χρόνους στον προγραμματισμό των PLC (σχόλια προγράμματος, αντιγραφή τμημάτων προγράμματος από ένα πρόγραμμα σε ένα άλλο κλπ).

Εμφανίζονται νέες γλώσσες προγραμματισμού για τεχνολόγους σε γραφική μορφή, όπου ο χρήστης μέσω βιβλιοθηκών κι έχοντας γνώση μόνο της παραγωγικής διαδικασίας "συνθέτει" τον αυτοματισμό του. Υποστηρίζεται τέλος και η εξέλιξη στις γλώσσες προγραμματισμού των ηλεκτρονικών υπολογιστών (Pascal, C++) για χρήστες που είναι εξοικειωμένοι σε τέτοια περιβάλλοντα.

Τέλος ιδιαίτερη έμφαση δίνεται πλέον στη δικτύωση – ασύρματη ή ενσύρματη για τον προγραμματισμό / επιτήρηση εξ αποστάσεως μέσω ειδικών συσκευών επικοινωνίας και λογισμικού για ηλεκτρονικό υπολογιστή (SCADA) καθώς και στις επικοινωνίες Internet.

### 2.1.2 Περιγραφή και λειτουργία του PLC

Το PLC είναι μία ηλεκτρονική διάταξη η οποία από την άποψη της λειτουργίας θα μπορούσε να προσομοιωθεί με έναν πίνακα αυτοματισμού. Έχει δηλαδή εισόδους και εξόδους που συνδέονται με τα στοιχεία μιας εγκατάστασης και βέβαια έναν αλγόριθμο που καθαρίζει ότι κάποιος συνδυασμός εισόδων παράγει ένα αποτέλεσμα στις εξόδους (π.χ. η ενεργοποίηση ενός τερματικού διακόπτη σταματά τον κινητήρα μιας μεταφορικής ταινίας) . Οι ομοιότητες όμως σταματούν εδώ μια και το ιδιαίτερο χαρακτηριστικό των PLC είναι σταθεροί και "συρματωμένοι" όπως σε ένα κλασσικό πίνακα αυτοματισμού αλλά μπορούν να μεταβάλλονται με την επέμβαση στο πρόγραμμα του PLC χωρίς καμία επέμβαση στο Hardware του συστήματος. Δηλαδή η λογική της λειτουργίας που ενσωματώνεται στο PLC μέσω του προγραμματισμού του είναι μεταβαλλόμενη.

Έτσι σε ότι αφορά το υλικό όλα τα PLC αποτελούνται από CPU η οποία περιέχει τη λογική του αυτοματισμού και η οποία αφού διαβάσει την κατάσταση των καρτών εισόδου (input modules) ενεργοποιεί τις κάρτες εξόδου (output modules) σύμφωνα με τους κανόνες (πρόγραμμα) που έχουμε αποθηκεύσει στη μνήμη του.

Βέβαια το σύστημα συμπληρώνεται από το τροφοδοτικό και πιθανόν από διατάξεις ενδείξεων και χειρισμών (operator panel ,operator display).

Η CPU με τη βοήθεια της κάρτας εισόδου γνωρίζει κάθε στιγμή την κατάσταση ενός διακόπτη εάν δηλαδή είναι διεγερμένος ή όχι .Επιπλέον με τη βοήθεια της κάρτας εξόδου οπλίζει ένα ρελέ και μέσω αυτού ενεργοποιεί μια διάταξη κίνησης, φωτισμού κτλ.

Αυτό που απομένει είναι η "λογική" δηλαδή πότε πρέπει να οπλίσει το ρελέ.

Αυτή η λογική είναι το πρόγραμμα του PLC που συντάσσεται σε συγκεκριμένη γλώσσα με τη βοήθεια ειδικού λογισμικού προγραμματισμού, και αποθηκεύεται στη μνήμη του PLC.

Έτσι τώρα το σύνολο του συστήματος λειτουργεί ως εξής:

Αρχικά η κεντρική μονάδα επεξεργασίας διαβάζει τις εισόδους , δηλαδή παρατηρεί την κάθε είσοδο , και αν σε αυτή έχει εμφανισθεί τάση (που σημαίνει ότι έχει κλείσει ο διακόπτης ) καταχωρεί ένα λογικό 1 σε μία περιοχή της μνήμης του που είναι ειδική για αυτό τον σκοπό (Input Image). Η περιοχή αυτή περιέχει σε κάθε στιγμή την κατάσταση των εισόδων και λειτουργεί σαν ενδιάμεσος σταθμός ανάμεσα στον "έξω κόσμο" και τη CPU.

Στη συνέχεια εκτελείται το πρόγραμμα δηλαδή εξετάζεται η τιμή των εισόδων και αποφασίζεται η τιμή της εξόδου η οποία και καταχωρείται σε μία αντίστοιχη περιοχή μνήμης εξόδου (Output Image).

Τέλος η περιοχή της μνήμης εξόδου μεταφέρεται στην κάρτα εξόδου και διεγείρει με την σειρά της ρελέ.

Η διαδικασία αυτή επαναλαμβάνεται από την αρχή και διαρκώς δηλαδή ξανά – διαβάζεται η είσοδος που μπορεί τώρα να έχει διαφορετική τιμή κλπ. Η διαδικασία αυτή λέγεται κυκλική επεξεργασία στο PLC.

Είναι ιδιαίτερα σημαντικό εδώ να τονίσουμε ότι η πληροφορία για την κατάσταση της εισόδου αποκτάται μόνο στην αρχή του κύκλου και η κατάσταση της εισόδου κατά το χρόνο εκτέλεσης του προγράμματος θεωρείται σταθερή (πράγμα που βεβαίως μπορεί να μην συμβαίνει), όμως ο κύκλος του PLC είναι τόσο σύντομος (τυπικά μερικά msec ) που ακόμα και αν αλλάξει κατάσταση η είσοδος , η CPU θα το αντιληφθεί στον αμέσως επόμενο κύκλο (π.χ. μετά από 3 ms) και θα δράσει ανάλογα με καθυστέρηση μόνο χιλιοστών του δευτερολέπτου.

Φυσικά για ιδιαίτερα κρίσιμες εισόδους υπάρχουν τεχνικές που επιτρέπουν την ακαριαία πληροφόρηση και δράση της κεντρικής μονάδας επεξεργασίας (Event driven interrupt).

Εδώ θα πρέπει επίσης να υπογραμμίσουμε ,όπως εξάλλου είδαμε και πιο πάνω, ότι το αποτέλεσμα του αυτοματισμού (το πότε θα διεγερθεί η έξοδος) το καθορίζει το πρόγραμμα και όχι οι καλωδιώσεις.

Θα μπορούσαμε διατηρώντας τις ίδιες ακριβώς καλωδιώσεις και αλλάζοντας μόνο το πρόγραμμα να κάνουμε το σύστημα να συμπεριφέρεται εντελώς διαφορετικά. Αυτή είναι και η μεγάλη διαφορά του PLC από οποιοδήποτε άλλο σύστημα αυτοματισμού που καθορίζει και το όνομα του δηλαδή προγραμματιζόμενος λογικός ελεγκτής.

### 2.1.3 Πλεονεκτήματα

Συγκριτικά με το κλασσικό αυτοματισμό τα πλεονεκτήματα του προγραμματισμένου με PLC είναι πάρα πολλά .Ενδεικτικά μπορούμε να αναφέρουμε:

- Είναι συσκευές γενικής χρήσεως – δεν είναι κατασκευασμένοι για ένα συγκεκριμένο είδος παραγωγής.
- Δεν ενδιαφέρει ο συνολικός αριθμός των επαφών, χρονικών, απαριθμητών κλπ που θα χρησιμοποιηθούν μιας και αποτελούν στοιχεία μνήμης της CPU και όχι φυσικές οντότητες.
- Η λειτουργία του αυτοματισμού μπορεί να αλλάξει σε οποιοδήποτε στάδιο θελήσουμε (μελέτη, κατασκευή, θέση σε λειτουργία ή αργότερα) χωρίς επέμβαση στο υλικό.
- Εύκολος οπτικός εντοπισμός με μία ματιά ,της λειτουργίας ή μη στοιχείων της εγκατάστασης με τη βοήθεια των LED που υπάρχουν σε όλες τις κάρτες εισόδου/εξόδου. Με τη βοήθεια συσκευής προγραμματισμού μπορεί να παρακολουθεί και η ροή εκτέλεσης του προγράμματος.
- Η κατασκευή του πίνακα που θα τοποθετηθεί το PLC γίνεται παράλληλη με τον προγραμματισμό του, πράγμα το οποίο οδηγεί στη συντομότερη παράδοση του αυτοματισμού.
- Πολύ συχνό είναι το φαινόμενο ο τεχνικός να κληθεί να επισκευάσει μια βλάβη και να δει έκπληκτος ότι άλλα υπάρχουν στα σχέδια και άλλα βλέπει αυτός στην εγκατάσταση. Το πρόβλημα αυτό δεν υπάρχει στα PLC αφού πάντα υπάρχει μόνο ένα "σχέδιο " αποθηκευμένο – το τελευταίο πρόγραμμα που του έχουμε περάσει. Εάν θέλουμε να έχουμε περισσότερα προγράμματα, αυτό είναι δυνατό με τη χρήση δισκετών.
- Τα PLC ως ηλεκτρονικές συσκευές καταλαμβάνουν πολύ μικρότερο χώρο στον πίνακα σε σχέση με τα υλικά του κλασσικού αυτοματισμού, καταναλώνουν δε πολύ λιγότερη ενέργεια από αυτά.
- Τοποθετούνται άφοβα και σε πεδία ισχύος – ο κατασκευαστής δίνει οδηγίες γι αυτές τις περιπτώσεις οι οποίες πρέπει να τηρούνται .
- Οι γλώσσες προγραμματισμού καλύπτουν όλο το φάσμα των ανθρώπων που καλούνται να ασχοληθούν με την τεχνολογία αυτή –Υπάρχει γλώσσα γι ανθρώπους με γνώση στο συμβατικό αυτοματισμό (Ladder), γλώσσες για όσους έχουν υπόβαθρο σε υπολογιστές καθώς και γλώσσες εξειδικευμένες για διάφορες τεχνολογίες.
- Τέλος, σαν ψηφιακές συσκευές σήμερα πια μας δίνουν τη δυνατότητα να συνδέσουμε επάνω τους οθόνες, εκτυπωτές, πληκτρολόγια και τους

πίνακες χειρισμών. Εύκολα είναι επίσης και η διασύνδεση μεταξύ τους για ανταλλαγή πληροφοριών, ο τηλεχειρισμός και η τηλεποπτεία, ο εξ αποστάσεως προγραμματισμός τους και η σύνδεσή τους στο Internet.

#### 2.1.4 Δυνατότητες και βελτιώσεις

Αντιθέτως με τους υπολογιστές γενικής χρήσης, τα PLC [Εικόνα 2] πλέον είναι σχεδιασμένα ούτως ώστε να έχουν πολλαπλές εισόδους και εξόδους, να αντέχουν σε ακραίες θερμοκρασίες, είτε ζέστη είτε κρύο, να μην έχουν ευαισθησία σε ηλεκτρικό θόρυβο και να έχουν αντίσταση στις δονήσεις και σε συγκρούσεις.

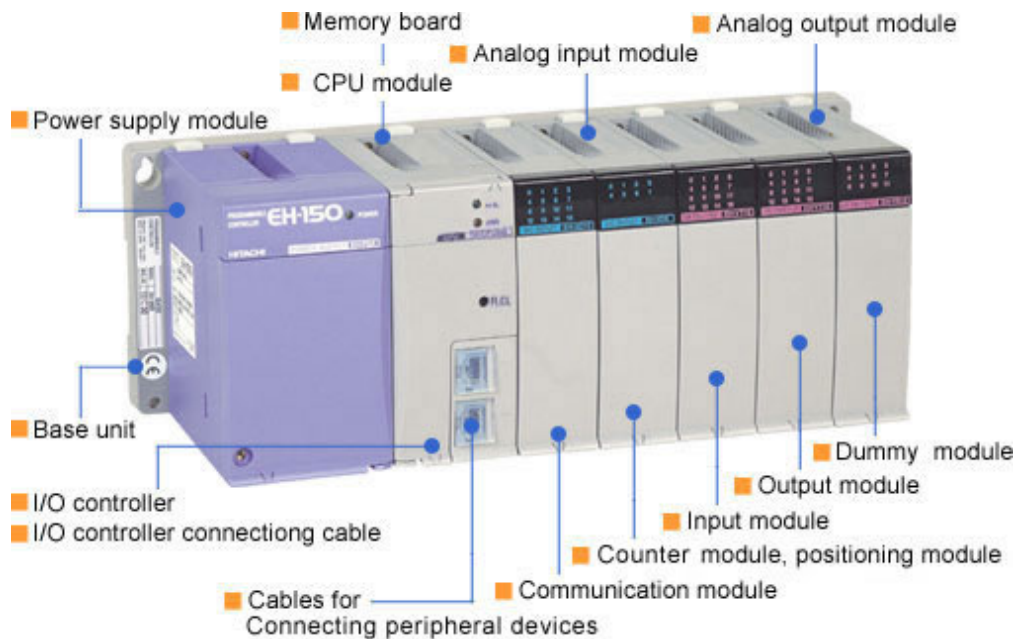
Οι απαιτήσεις που ζητούνται από τα PLC συνεχίζουν να αυξάνονται με ραγδαίους ρυθμούς ταυτόχρονα με την εξάπλωσή της χρήσης τους στην βιομηχανία. Προστέθηκαν οι δυνατότητες οπτικοποίησης της κατάστασης του προγράμματος και του σημείου που βρίσκεται και την προβολή σε κάποια οθόνη και τον καλύτερο έλεγχο, όπως την δυνατότητα να παρεμβαίνεις σε συγκεκριμένες περιπτώσεις ή την δυνατότητα να κλειδώνεις το πρόγραμμα από ανεπιθύμητους χρήστες. Έπειτα δημιουργήθηκε η ανάγκη να ελέγχουμε και να ενώνουμε ανεξάρτητα μεταξύ τους συστήματα και έτσι δημιουργήθηκαν υπολογιστές οι οποίοι χρησιμοποιούσαν εντολές υψηλού επιπέδου και διαχειριζόντουσαν περισσότερα από ένα PLC ταυτόχρονα.



Εικόνα 2 - PLC Allen Bradley

#### 2.1.5 Επικοινωνία και διασύνδεση

Τα PLC έχουν ενσωματωμένες πόρτες επικοινωνίας, με πιο συνηθισμένες την σειριακή RS-232 και την Ethernet. Από πλευράς επικοινωνιακών πρωτοκόλλων, χρησιμοποιούνται πιο πολύ τα Modbus, BACnet ή DF1. Τα πιο σύγχρονα PLC επικοινωνούν μέσω δικτύου με άλλα συστήματα, όπως υπολογιστές που χρησιμοποιούν σύστημα SCADA (Supervisory Control And Data Acquisition) ή με web browser.



Εικόνα 3 - Τα μέρη ενός PLC

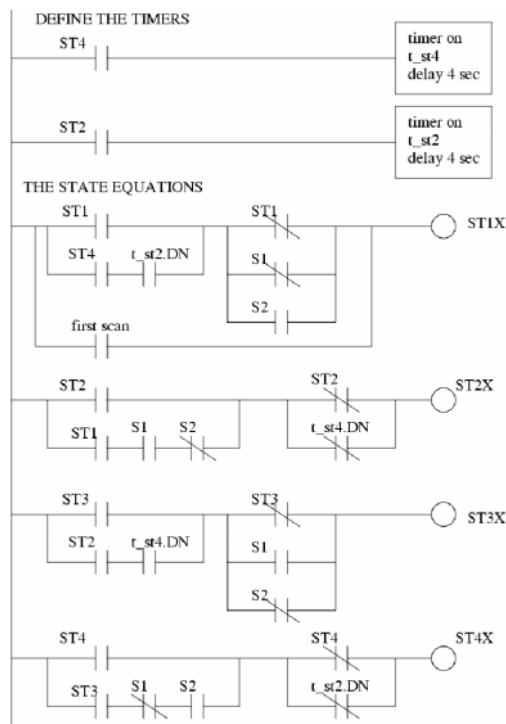
Τα PLC τα οποία χρησιμοποιούνται σε μεγάλου μεγέθους συστήματα που περιλαμβάνουν μεγάλο αριθμό εισόδων και εξόδων έχουν την δυνατότητα για άμεση επικοινωνία με άλλους ελεγκτές (peer-to-peer (P2P)). Αυτό επιτρέπει σε ξεχωριστά μέρη σύνθετων διεργασιών να έχουν τον έλεγχο ενώ επιτρέπουν στα υποσυστήματα να συντονίζονται μέσω του καναλιού επικοινωνίας.

Αυτά τα κανάλια επικοινωνίας χρησιμοποιούνται πολλές φορές για συσκευές επικοινωνίας με τον χρήστη όπως πληκτρολόγια και άλλα.

Τα PLC χρησιμοποιούν τις θύρες εισόδου/εξόδου [Εικόνα 3] που διαθέτουν για να επικοινωνήσουν με αισθητήρια και μηχανισμούς κίνησης. Τα PLC διαβάζουν σήματα από διακόπτες, μεταβλητές από αναλογικές πηγές, όπως θερμοκρασία και πίεση, και την θέση πολύπλοκων συστημάτων εντοπισμού θέσης. Κάποια χρησιμοποιούν και οπτικά αισθητήρια. Από την πλευρά των μηχανισμών κίνησης, τα PLC χρησιμοποιούν ηλεκτρικούς κινητήρες, πνευματικά ή υδραυλικά έμβολα, μαγνητικά ρελέ, πηνία ή αναλογικές εξόδους. Οι εισοδοί-εξοδοί είναι είτε ενσωματωμένοι στο PLC είτε είναι εξωτερικοί και συνδέονται με το PLC μέσω κάποιου δικτύου στο οποίο είναι συνδεδεμένο το PLC.



## 2.1.6 Γλώσσες προγραμματισμού



Εικόνα 4 – Γλώσσα Ladder

Τα πρώτα PLC, λόγω του ότι είχαν σχεδιαστεί για να αντικαταστήσουν τα συστήματα της λογικής των ηλεκτρικών ρελέ, έπρεπε να προγραμματιστούν με μία λογική και με μια γλώσσα προγραμματισμού που να θυμίζει την παλιά λογική. Έτσι δημιουργήθηκε η γλώσσα 'ladder' [Εικόνα 4] η οποία θυμίζει κατά πολύ το διάγραμμα της λογικής των ηλεκτρικών ρελέ. Με την επιλογή και δημιουργία αυτού του προγράμματος, μειώθηκαν οι απαιτήσεις για εκπαίδευση των ήδη υπαρχόντων τεχνικών. Τα πρώτα PLC επίσης δεν είχαν τερματικούς σταθμούς για τον προγραμματισμό τους οι οποίοι να ήταν ικανοί για γραφική παρουσίαση της λογικής του προγράμματος και έτσι παρουσιάζονταν ως μία σειρά από λογικές εκφράσεις σε μία έκδοση της άλγεβρας του Bool.

Με την εξέλιξη των τερματικών σταθμών όμως, έγινε πιο σύνηθες η χρησιμοποίηση της γλώσσας ladder για τους προαναφερθέντες λόγους και γιατί ήταν σε μορφή που έμοιαζε με τους ηλεκτρομηχανικούς πίνακες ελέγχου.

Τα σύγχρονα PLC μπορούν να προγραμματιστούν σε ένα μεγάλο φάσμα γλωσσών, από την λογική ladder μέχρι και συγκεκριμένες διαλέκτους της BASIC και της C. Πιο συγκεκριμένα, γλώσσες όπως function block diagram (FBD), ladder diagram (LD), structured text (ST παρόμοια με την γλώσσα προγραμματισμού Pascal), instruction list (IL παρόμοια με την γλώσσα προγραμματισμού assembly) και sequential function chart (SFC). Μια άλλη μέθοδος είναι η λογική των καταστάσεων, όπου είναι μια πολύ υψηλού επιπέδου γλώσσα προγραμματισμού η οποία σχεδιάστηκε για να προγραμματίζονται τα PLC βάση των διαγραμμάτων state transition diagrams.

### 2.1.7 Χρόνος επεξεργασίας

Η βασική δουλειά των PLC περιλαμβάνει την σύνδεση των σημάτων εισόδου με ένα συγκεκριμένο πρόγραμμα και αν η κατάσταση τους είναι 'αληθής' να αλλάξουν την αντίστοιχη έξοδο. Μία είσοδος και μία έξοδος μπορεί να έχουν μόνο δύο καταστάσεις την 'αληθή' δηλαδή '1' και την 'μη αληθή' δηλαδή '0'. Πλέον τα PLC έχουν πολλές περισσότερες λειτουργίες όπως μετρητές χρόνου και επαναλήψεων, ρυθμίσεις σε θέματα μνήμης μέχρι και την δυνατότητα πραγματοποίησης μαθηματικών πράξεων. Αλλά η λογική των εναλλαγών των καταστάσεων των εισόδων και των εξόδων, παραμένει πάντα.

Ένα πρόγραμμα PLC συνήθως εκτελείται επαναλαμβανόμενα όση ώρα τρέχει το σύστημα. Η κατάσταση των εισόδων καταγράφεται σε ένα μέρος της μνήμης το οποίο είναι προσβάσιμο από τον επεξεργαστή. Το πρόγραμμα τότε τρέχει από την πρώτη βαθμίδα μέχρι την τελευταία, αλλά παίρνει κάποιον χρόνο στον επεξεργαστή του PLC για να αξιολογήσει όλες τις βαθμίδες και να αναβαθμίσει τον πίνακα των εισόδων/εξόδων με την κατάσταση των εξόδων. Αυτή η διαδικασία μπορεί να πάρει λίγα milliseconds για ένα μικρό πρόγραμμα ή έναν γρήγορο επεξεργαστή αλλά τα παλιότερα PLC για να τρέξουν και να εκτελέσουν επιτυχώς ένα πολύ μεγάλο πρόγραμμα χρειαζόντουσαν πολύ περισσότερο χρόνο (ίσως και 100ms).

Όσο τα PLC βελτιώνονται, έχουν αναπτυχθεί διάφοροι μέθοδοι για να μειωθεί ο χρόνος επεξεργασίας, όπως η χρησιμοποίηση υπορουτίνων. Με αυτό τον τρόπο απλοποιείται ο προγραμματισμός και βοηθάει στην μείωση του χρόνου επεξεργασίας του προγράμματος. Για παράδειγμα, τα κομμάτια που χρησιμοποιούνται μόνο στην αρχή και στο τέλος της λειτουργίας της μηχανής μπορούν να παραληφθούν από το κυρίως πρόγραμμα και να χρησιμοποιούν την ισχύ του επεξεργαστή μόνο όταν χρειάζεται πραγματικά. Επίσης χρησιμοποιούνται μονάδες εισόδου/εξόδου όπως μετρητές χρόνου ή γενικοί μετρητές, σε περιπτώσεις που ο χρόνος είναι πολύ δύσκολο να υπολογιστεί αξιόπιστα, όπως στις περιπτώσεις που μετράμε παλμούς. Έτσι σε πιο αργά PLC μπορούμε να μετρήσουμε αξιόπιστα τέτοιες τιμές, χρησιμοποιώντας εξωτερικές μονάδες που δεν επηρεάζονται από το χρόνο που παίρνει στο σύστημα να επεξεργαστεί το πρόγραμμα.

### 2.1.8 Επικοινωνία με τον χρήστη

Τα PLC χρειάζεται να επιδρούν με τον άνθρωπο για περιπτώσεις βελτιστοποίησης και προσαρμογής, αναφοράς σφαλμάτων, ή απλό καθημερινό έλεγχο. Γι αυτό τον λόγο χρησιμοποιούμε ένα γραφικό μέσω επικοινωνίας με τον χρήστη. Σε ένα απλό τέτοιο σύστημα μπορεί να χρησιμοποιεί απλά κουμπιά και λαμπάκια για να επικοινωνεί με τον χρήστη ενώ σε λίγο πιο προχωρημένα συστήματα έχουμε οθόνες που παρουσιάζουν κείμενα όπως και οθόνες επαφής με γραφική απεικόνιση [Εικόνα 5].

Στα πιο προχωρημένα PLC έχουμε την χρήση ειδικών προγραμμάτων εγκατεστημένα σε κάποιον υπολογιστή τα οποία χρησιμοποιούνται για τον προγραμματισμό και την παρακολούθηση των διαδικασιών. Οι υπολογιστές αυτοί και συνεπώς τα προγράμματα αυτά, επικοινωνούν με τα PLC μέσω κάποιων από

τους κλασσικούς τρόπους επικοινωνίας, όπως για παράδειγμα μέσω δικτύου με την χρήση του πρωτοκόλλου Ethernet.



Εικόνα 5 – οθόνη ελέγχου PLC

Τα προγράμματα συνήθως γράφονται σε μια ειδική εφαρμογή στον υπολογιστή και μετά μεταφέρονται στο PLC μέσω κάποιου απ τους τρόπους επικοινωνίας και αποθηκεύονται στο PLC είτε σε κάποια μνήμη τύπου RAM που υποστηρίζεται από κάποια μπαταρία είτε σε κάποια μνήμη τύπου flash.

Ενώ η γενική ιδέα των PLC είναι κοινή σε όλους τους κατασκευαστές, διαφορές στην οργάνωση των εισόδων/εξόδων, στην διαμόρφωση της μνήμης και σε άλλες βασικές λειτουργίες δεν επιτρέπει σε PLC διαφορετικών εταιριών να επικοινωνήσουν μεταξύ τους. Πολλές φορές ακόμα και PLC της ίδιας εταιρίας αλλά διαφορετικά μοντέλα δεν διαθέτουν αυτή την δυνατότητα.

## 2.2 Θεωρητικό υπόβαθρο πρακτορικού συστήματος

Πράκτορας λογισμικού είναι ένα πρόγραμμα υπολογιστή το οποίο δρα για λογαριασμό του χρήστη σε σχέση με άλλες μεταβλητές του συστήματος στο οποίο ανήκει. Η ιδέα των πρακτόρων άρχισε να καλλιεργείται στα μέσα της δεκαετίας του 1950 αλλά η πιο συγκεκριμένη έρευνα χωρίζεται σε δύο περιόδους, η πρώτη στα μέσα του 1970 και η δεύτερη στις αρχές του 1990. Τα συστήματα των πρακτόρων λογισμικού είναι ουσιαστικά η εξέλιξη των πολυπρακτορικών συστημάτων (Multi-Agent Systems) τα οποία εξελίχθηκαν από διάφορα συστήματα τεχνητής νοημοσύνης.

Δεν υπάρχει μόνο ένας ορισμός όμως για του πράκτορες λογισμικού. Με την εξέλιξη της τεχνολογίας και την ανάπτυξη της έρευνας σε αυτά τα κομμάτια διάφοροι ορισμοί κάνουν την εμφάνιση τους με τους κυριότερους να είναι:

«Πράκτορας είναι οτιδήποτε μπορεί να θεωρηθεί ότι αντιλαμβάνεται το περιβάλλον του μέσω αισθητηρίων και αντιδρά πάνω σε αυτό το περιβάλλον μέσω τελεστών»

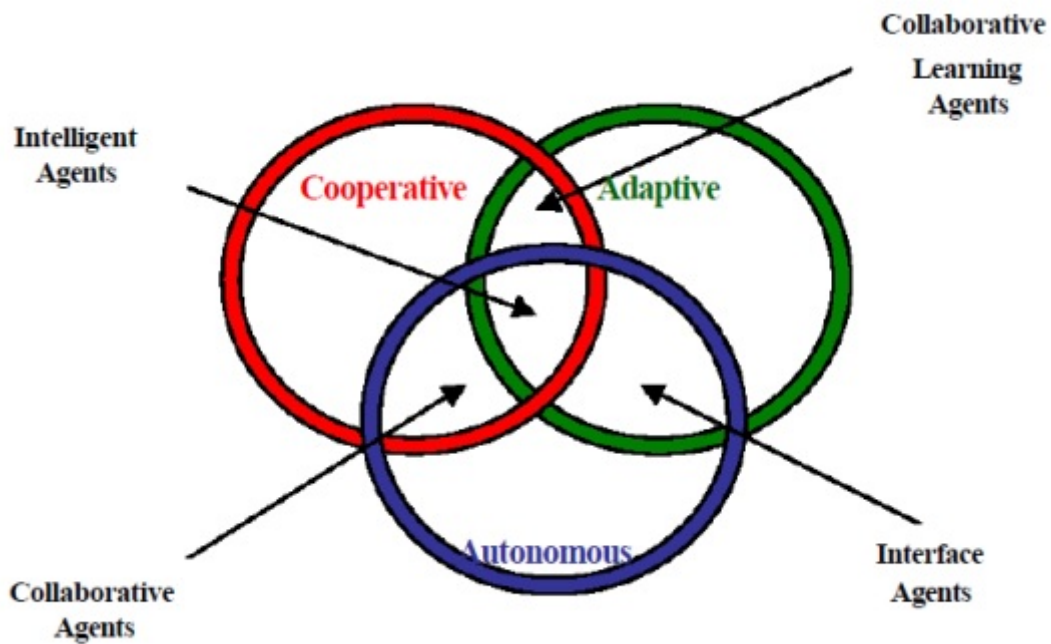
«Αυτόνομοι πράκτορες είναι υπολογιστικά συστήματα τα οποία κατοικούν μέσα σε πολύπλοκα δυναμικά περιβάλλοντα, διαισθάνονται και πράττουν αυτόνομα σε αυτό το περιβάλλον και κάνοντας το αυτό πραγματοποιούν ένα σύνολο στόχων ή καθηκόντων για τα οποία είναι σχεδιασμένοι.»

«Ένας αυτόνομος πράκτορας είναι ένα σύστημα που βρίσκεται μέσα και αποτελεί μέρος ενός περιβάλλοντος το οποίο αισθάνεται αυτό το περιβάλλον και επιδρά σε αυτό, σε βάθος χρόνου, με στόχο την επιδίωξη του δικού του προγραμματισμού ούτως ώστε να επηρεάσει το τι θα αισθάνεται στο μέλλον.»

Βάση και των παραπάνω ορισμών, οδηγούμαστε στο συμπέρασμα ότι οι πράκτορες διαφέρουν από τα προγράμματα λογισμικού. Πιο συγκεκριμένα, είναι τέσσερα σημεία τα οποία δείχνουν αυτή την διαφορά. Αρχικά οι πράκτορες έχουν αλληλεπίδραση με το περιβάλλον τους, που σημαίνει ότι χρησιμοποιούν αισθητήρια από τα οποία παίρνουν πληροφορίες για την κατάσταση του περιβάλλοντος και επίσης δρουν ανάλογα με το αποτέλεσμα που θέλουν να πετύχουν και αλλάζουν το περιβάλλον. Ακόμα είναι η αυτονομία, όπου κάθε πράκτορας μπορεί να δράσει ανεξάρτητα από άλλους πράκτορες και δεν χρειάζεται να είναι διασυνδεδεμένος με ένα ολόκληρο σύστημα. Με αυτό τον τρόπο μπορεί και να τοποθετηθεί και σε διαφορετικό σύστημα από το αρχικό χωρίς πρόβλημα. Επίσης είναι η επιμονή στην εκπλήρωση του έργου που έχουν αναλάβει καθώς προσπαθούν πάντοτε να βρουν την λύση και να επιτύχουν τους στόχους που έχουν θεσπίσει και αυτό συνδέεται με το τέταρτο σημείο που είναι το γνώρισμα που έχουν να επικεντρώνονται πάντα στον στόχο τους. Με αυτό τον τρόπο οι πράκτορες δουλεύουν έχοντας έναν και μόνο σκοπό, χωρίς να αναλώνονται σε άλλες ενέργειες οι οποίες πιθανών να τους καταναλώνουν πόρους, όπως επίσης και χρόνο.

Οι πράκτορες λογισμικού, χωρίζονται σε τέσσερις κατηγορίες οι οποίες προκύπτουν από συνδυασμό των τριών βασικών χαρακτηριστικών των πρακτόρων, τα οποία είναι η συνεργασία (cooperative), η προσαρμοστικότητα (adaptive) και η

αυτονομία (autonomous) [Εικόνα 6]. Έτσι προκύπτουν οι πράκτορες συνεργασίας (collaborative agents), οι οποίοι αποτελούν τους πράκτορες που έχουν ως χαρακτηριστικά την αυτονομία και την συνεργασία. Ακόμα είναι οι πράκτορες όπου μαθαίνουν συνεργασία (collaborative learning agents), όπου προκύπτουν από τον συνδυασμό των χαρακτηριστικών της συνεργασίας και της προσαρμοστικότητας. Συνέχεια έχουν οι πράκτορες διασύνδεσης (interface agents) όπου τα χαρακτηριστικά τους είναι η προσαρμοστικότητα και η αυτονομία και τέλος οι νοήμονες πράκτορες (intelligent agents) όπου κάνουν συνδυασμό των τριών προαναφερθέντων χαρακτηριστικών, συνεργασία, προσαρμοστικότητα και αυτονομία.



Εικόνα 6 - Τύποι πρακτόρων

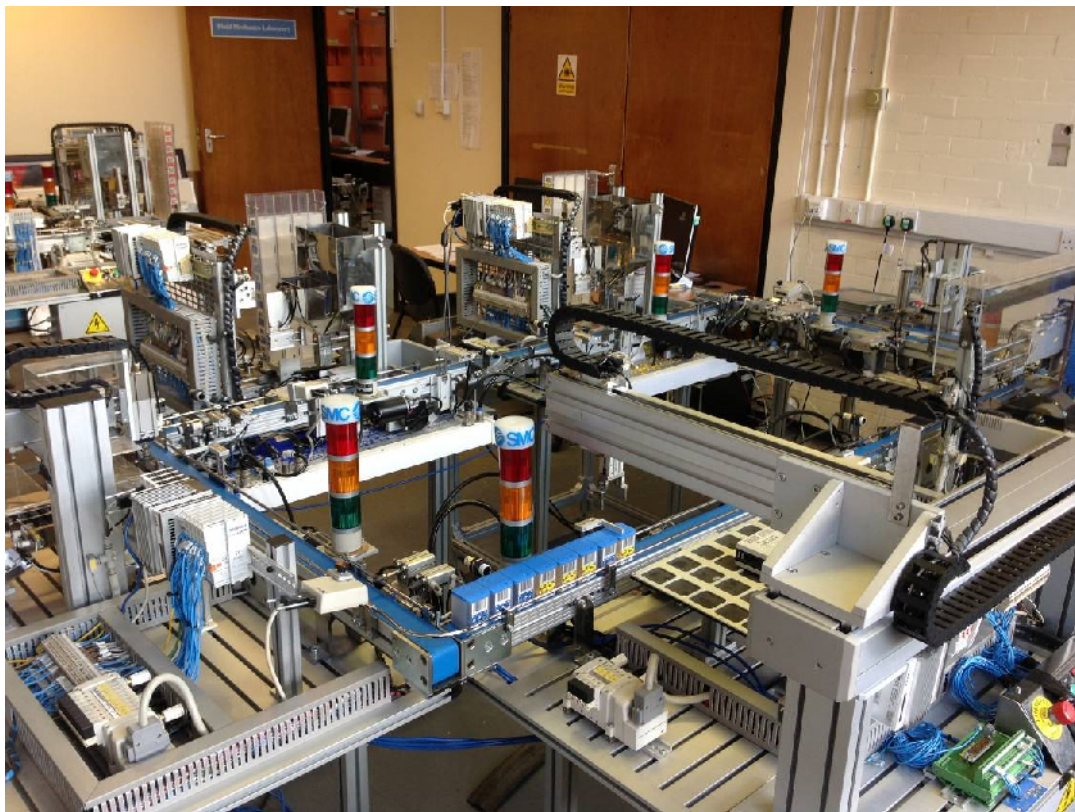
### 3. Παρουσίαση πειραματικού εξοπλισμού-(Hardware σταθμών)

Σε αυτό το κεφάλαιο γίνεται η παρουσίαση του εξοπλισμού που χρησιμοποιήθηκε για την δημιουργία αυτής της πτυχιακής εργασίας. Πιο συγκεκριμένα, αναλύεται ο κάθε σταθμός και τα μέρη από τα οποία αποτελείται, η λειτουργία τους και οι δυνατότητες τους. Ακόμα, γίνεται αναφορά στα βοηθητικά συστήματα του κάθε σταθμού όπως φώτα και κουμπιά.

Τέλος γίνεται ανάλυση στους ελεγκτές που χρησιμοποιούνται στους συγκεκριμένους σταθμούς, οι δυνατότητες τους και τα χαρακτηριστικά τους και επίσης γίνεται αναφορά στο λειτουργικό που χρησιμοποιείται για την παραμετροποίηση του συστήματος και τον προγραμματισμό αυτών των ελεγκτών.

#### 3.1 Το σύστημα

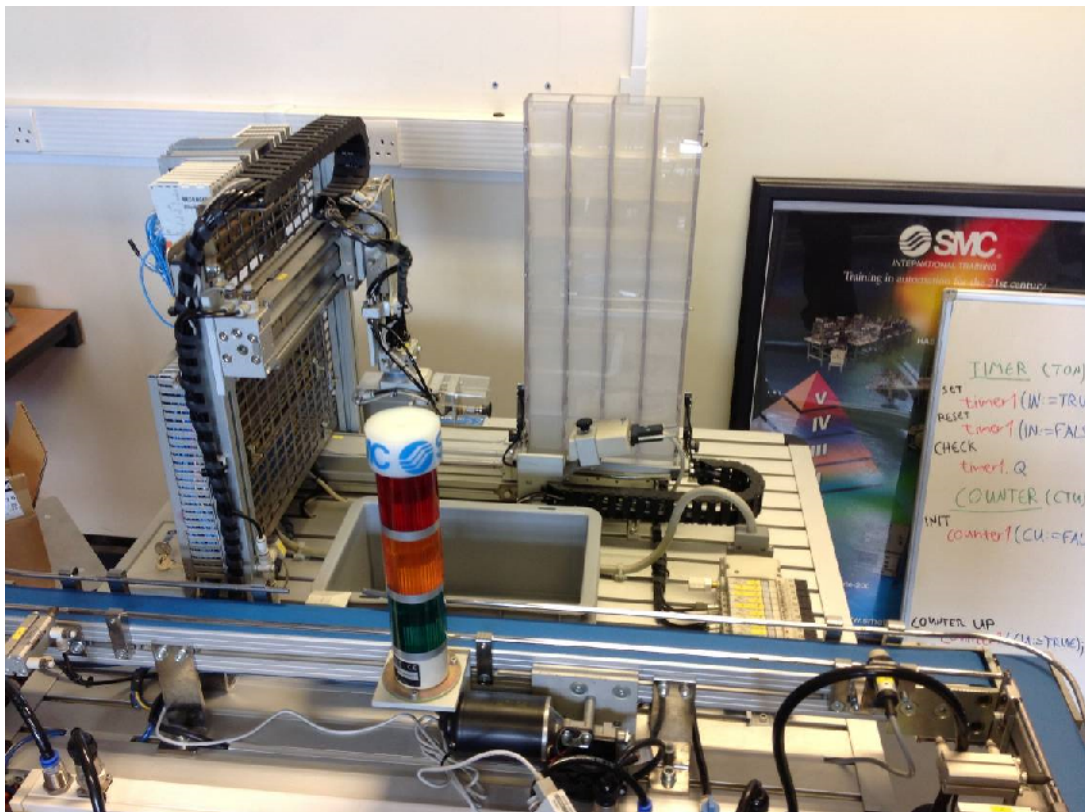
Το σύστημα στο οποίο ανήκουν οι σταθμοί που χρησιμοποιήθηκαν για την δημιουργία της πτυχιακής αυτής εργασίας είναι το HAS-200 της εταιρίας SMC [Εικόνα 7]. Πρόκειται για ένα σύστημα το οποίο έχει σχεδιαστεί βασισμένο στα πρότυπα μιας αυτοματοποιημένης γραμμής παραγωγής αλλά σε μικρότερη κλίμακα. Αποτελείται από συνδυασμό διαφόρων τεχνολογιών αυτοματισμού με την χρήση ηλεκτροπνευματικών βαλβίδων και βοηθάει στην καλύτερη κατανόηση της έννοιας του αυτοματισμού και των συστημάτων παραγωγής για εκπαιδευτικούς και ερευνητικούς σκοπούς.



Εικόνα 7 – απεικόνιση κάποιων σταθμών του συστήματος

### 3.1.1 1<sup>ος</sup> σταθμός (διαφορετικά χρώματα)

Ο πρώτος σταθμός χρησιμοποιείται για να τροφοδοτήσει την γραμμή παραγωγής με άδεια κουτιά με πολύχρωμη ετικέτα [Εικόνα 8]. Τα κουτιά αποθηκεύονται το ένα πάνω στο άλλο σε τέσσερις στήλες και κάθε μία αποτελείται από εννιά κουτιά. Για να χρησιμοποιηθούν τα αποθηκευμένα κουτιά υπάρχουν δύο πνευματικές βαλβίδες οι οποίες μετακινούν τις στήλες ούτως ώστε να έρθουν στην σωστή θέση. Στην συνέχεια χρησιμοποιείται ένας πνευματικός κύλινδρος ο οποίος σπρώχνει το κουτί που βρίσκεται στον πάτο της στήλης η οποία βρίσκεται μπροστά του και το τοποθετεί στην προβλεπόμενη θέση μπροστά από έναν αναγνώστη barcode. Οι στήλες έχουν δύο φωτοαισθητήρια για να ανιχνεύουν τον αριθμό των σειρών από κουτιά που υπάρχουν διαθέσιμα. Ο πρώτος φωτοαισθητήρας βρίσκεται στην προτελευταία σειρά και ενεργοποιείται όταν δεν υπάρχει κανένα κουτί στην συγκεκριμένη σειρά αλλά μόνο στην τελευταία (το πολύ τέσσερα κουτιά διαθέσιμα). Ο δεύτερος φωτοαισθητήρας βρίσκεται στην τελευταία σειρά και ενεργοποιείται όταν όλες οι στήλες αδειάσουν από κουτιά.



Εικόνα 8 – 1<sup>ος</sup> σταθμός

Αφού το κουτί τοποθετηθεί μπροστά από τον αναγνώστη barcode, ελέγχεται το barcode του για να διαπιστωθεί αν ανήκει στα κουτιά που πρέπει να τοποθετηθούν στην γραμμή παραγωγής ή αν χρειάζεται να πάει στον κάδο με τα κουτιά που δεν χρησιμοποιούνται. Ο αναγνώστης barcode επικοινωνεί με το PLC μέσω σειριακής θύρας RS-232 [Εικόνα 9].



Εικόνα 9 – αναγνώστης Barcode

Στην συνέχεια χρησιμοποιούνται διάφοροι μηχανισμοί κίνησης ούτως ώστε να μετακινηθεί το κουτί από τον αναγνώστη barcode στον ιμάντα μεταφοράς της γραμμής παραγωγής ή στον κάδο. Υπάρχουν μηχανισμοί κίνησης όπου μεταφέρουν τα υπόλοιπα μέρη του μηχανισμού ευθύγραμμα ή κατακόρυφα προκειμένου να έρθουν στην κατάλληλη θέση για την συνέχεια της διαδικασίας. Επίσης υπάρχουν μηχανισμοί οι οποίοι πραγματοποιούν κυκλικές κινήσεις για να περιστρέψουν τα υπόλοιπα κομμάτια στην κατάλληλη θέση. Ακόμα υπάρχει μηχανισμός ο οποίος είναι υπεύθυνος για να κρατάει το κουτί και ο οποίος αποτελείται από μία βεντούζα αέρα η οποία συγκρατεί το κουτί πάνω του.

Όλοι οι προαναφερθέντες μηχανισμοί κίνησης έχουν αισθητήρες οι οποίοι χρησιμοποιούνται για να ενημερώσουν το σύστημα για την ακριβή θέση του κάθε ενός ανά πάσα στιγμή.

Στο σταθμό υπάρχουν επίσης τρία φώτα στα χρώματα του πράσινου, πορτοκαλί και κόκκινου τα οποία ανάβουν αναλόγως με την κατάσταση της διαδικασίας.

Τελευταίο κομμάτι του σταθμού είναι ο ιμάντας μεταφοράς. Τροφοδοτείται από έναν κινητήρα συνεχούς ρεύματος (DC) και περιέχει και επιπλέον μία βαλβίδα και έναν αισθητήρα για τον έλεγχο της ροής των κουτιών. Στην αρχή του ιμάντα υπάρχει ένας πνευματικός κύλινδρος τον οποίο ενεργοποιούμε όταν υπάρχει ροή απ τον προηγούμενο σταθμό και δεν θέλουμε να εμπλακεί με την ροή του σταθμού μας. Ο κύλινδρος αυτός λειτουργεί σαν πόρτα η οποία συγκρατεί όλα τα κουτιά που έρχονται από τον προηγούμενο σταθμό. Στο τέλος του ιμάντα υπάρχει ένας φωτοηλεκτρικός αισθητήρας ο οποίος ανιχνεύει αν υπάρχει ουρά στον επόμενο σταθμό και δίνει σήμα στο PLC για να αποφασίσει για τον ρυθμό που θα επεξεργάζεται τα κουτιά.

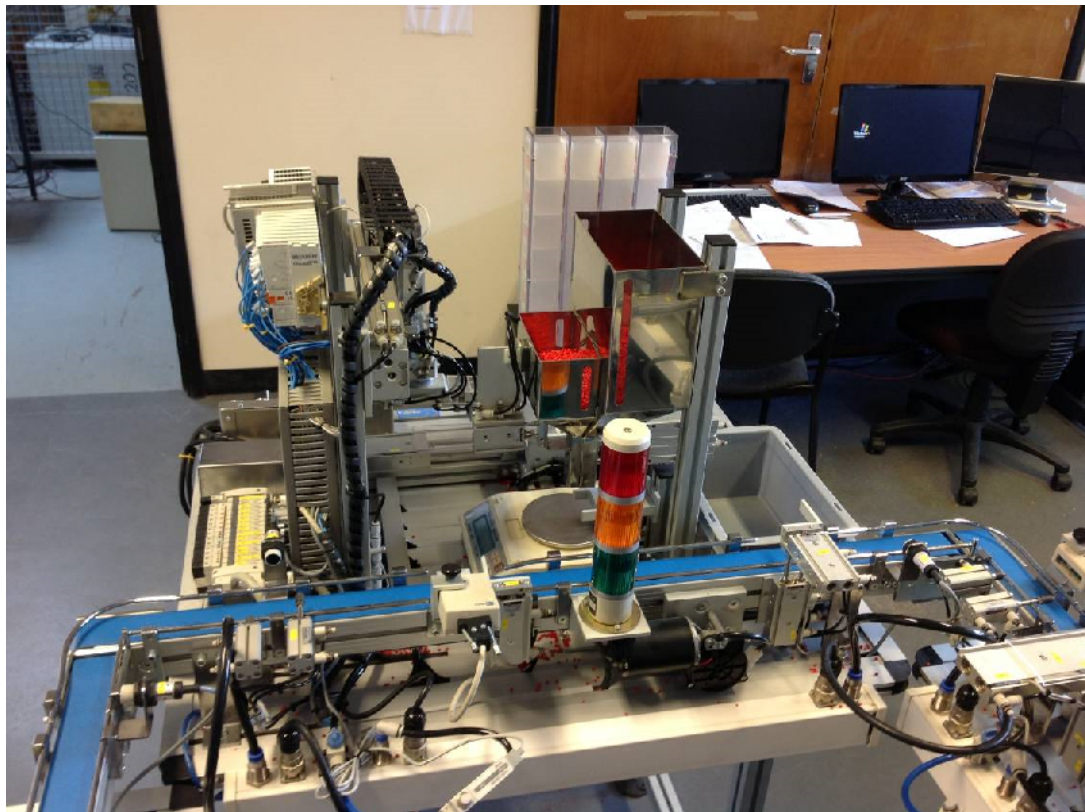
Όλα τα μέρη, βαλβίδες και αισθητήρια του σταθμού ελέγχονται από τα βιομηχανικά PC της εταιρίας Beckhoff (μοντέλο CX-1030) για τα οποία θα γίνει αναφορά με περισσότερες λεπτομέρειες σε επόμενο κεφάλαιο.

### 3.1.2 2<sup>ος</sup> σταθμός (σταθμός παραγωγής)

Ο δεύτερος σταθμός χρησιμοποιείται για να γεμίσει τα κουτιά τα οποία έχουν κόκκινη ετικέτα, με συγκεκριμένη ποσότητα υλικού [Εικόνα 10]. Τα κουτιά αποθηκεύονται το ένα πάνω στο άλλο σε τέσσερις στήλες και κάθε μία αποτελείται από 9 κουτιά. Για να χρησιμοποιηθούν τα αποθηκευμένα κουτιά υπάρχουν δύο



πνευματικοί κύλινδροι οι οποίοι μετακινούν τις στήλες ούτως ώστε να έρθουν στην σωστή θέση. Μετά χρησιμοποιείται ένας πνευματικός κύλινδρος ο οποίος σπρώχνει το κουτί που βρίσκεται στον πάτο της στήλης η οποία βρίσκεται μπροστά του και το τοποθετεί στην προβλεπόμενη πλατφόρμα για να το παραλάβει ο μηχανισμός που χρησιμοποιείται για την μετακίνηση των κουτιών. Οι στήλες έχουν δύο φωτοαισθητήρια για να ανιχνεύουν τον αριθμό των σειρών από κουτιά που υπάρχουν διαθέσιμα. Ο πρώτος φωτοαισθητήρας βρίσκεται στην προτελευταία σειρά και ενεργοποιείται όταν υπάρχουν κουτιά διαθέσιμα μόνο στην τελευταία σειρά (το πολύ τέσσερα κουτιά διαθέσιμα). Ο δεύτερος φωτοαισθητήρας βρίσκεται στην τελευταία σειρά και ενεργοποιείται όταν όλες οι στήλες αδειάσουν από κουτιά.



Εικόνα 10 – 2<sup>ος</sup> σταθμός

Μόλις το κουτί έρθει στην πλατφόρμα χρησιμοποιείται ο πνευματικός μηχανισμός ο οποίος μετακινεί τους μηχανισμούς κίνησης στο ύψος του κουτιού και σπρώχνει μέσω ενός πνευματικού κυλίνδρου μία βεντούζα αέρα η οποία θα συγκρατήσει το κουτί. Στην συνέχεια θα μεταφέρει το κουτί μπροστά από έναν αναγνώστη barcode για έλεγχο. Ο αναγνώστης βρίσκεται μπροστά στον ιμάντα μεταφοράς όπου και τοποθετείται το κουτί. Το κουτί συγκρατείται από ένα μηχανισμό που αποτελείται από μία πνευματική βαλβίδα και έναν άξονα που υπάρχει κάθετα στον ιμάντα. Στην συνέχεια ελέγχεται το barcode του για να διαπιστωθεί αν ανήκει στα κουτιά που πρέπει να γεμίσουν με συγκεκριμένη ποσότητα υλικού ή αν χρειάζεται να πάει στον κάδο με τα κουτιά που δεν χρησιμοποιούνται. Στην δεύτερη περίπτωση ο άξονας που συγκρατεί το κουτί μετακινείται και το κουτί συνεχίζει την πορεία του μέχρι το τέλος της γραμμής όπου

και συναντά μία άλλη πνευματική βαλβίδα που μπαίνει σε λειτουργία για να το πετάξει στον κάδο. Κατά την διάρκεια της διαδικασίας αυτής ο αναγνώστης barcode επικοινωνεί με το PLC μέσω σειριακής θύρας RS-232.

Στην συνέχεια χρησιμοποιούνται διάφοροι μηχανισμοί κίνησης ούτως ώστε να μετακινηθεί το κουτί από τον αναγνώστη barcode στην ζυγαριά που βρίσκεται κάτω από την καταπακτή που πέφτει το υλικό και από εκεί πίσω στον ιμάντα μεταφοράς. Υπάρχουν μηχανισμοί κίνησης όπου μεταφέρουν τα υπόλοιπα μέρη του μηχανισμού ευθύγραμμα ή κατακόρυφα προκειμένου να έρθουν στην κατάλληλη θέση για την συνέχεια της διαδικασίας. Επίσης υπάρχουν μηχανισμοί οι οποίοι πραγματοποιούν κυκλικές κινήσεις για να περιστρέψουν τα υπόλοιπα κομμάτια στην κατάλληλη θέση. Ακόμα υπάρχει μηχανισμός ο οποίος είναι υπεύθυνος για να κρατάει το κουτί και ο οποίος αποτελείται από μία βεντούζα αέρα η οποία συγκρατεί το κουτί πάνω του.

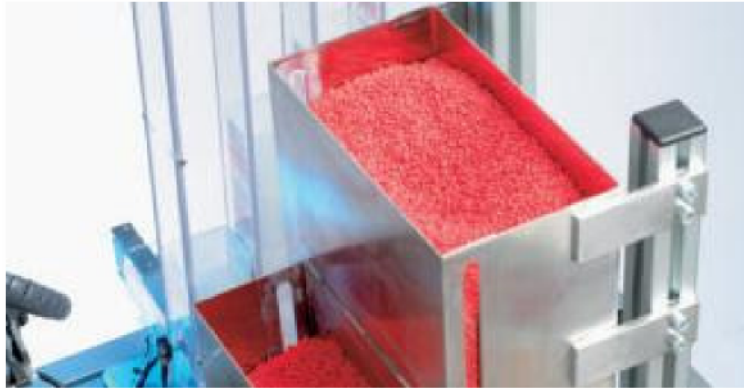
Όλοι οι προαναφερθέντες μηχανισμοί κίνησης χρησιμοποιούν αισθητήρια για να μεταφέρουν πληροφορίες προς το σύστημα για την ακριβή θέση του κάθε ενός ανά πάσα στιγμή.

Η ζυγαριά [Εικόνα 11] χρησιμοποιείται για να μετρήσει την ποσότητα του υλικού που υπάρχει μέσα στο κουτί. Η επικοινωνία με το PLC γίνεται μέσω σειριακής θύρας RS-232. Διαθέτει επίσης και μία οθόνη LCD για απευθείας έλεγχο των τιμών.



Εικόνα 11 - ζυγαριά

Πάνω από την ζυγαριά βρίσκεται τοποθετημένο το δοχείο [Εικόνα 12] που περιέχει το υλικό με το οποίο γεμίζουν τα κουτιά. Δύο πνευματικές βαλβίδες ελέγχουν δύο καταπακτές, μία με μεγάλη ροή και μία με μικρή, από τις οποίες περνάει το υλικό μέσα στο κουτί. Το δοχείο περιέχει και δύο φωτοαισθητήρια με τα οποία ενημερωνόμαστε για τον όγκο του υλικού που απομένει μέσα στο δοχείο.



Εικόνα 12 – υλικό για γέμιση κουτιών

Στο σταθμό υπάρχουν επίσης τρία φώτα με χρώμα πράσινο, πορτοκαλί και κόκκινο τα οποία ανάβουν με συγκεκριμένη σειρά αναλόγως με την κατάσταση της διαδικασίας.

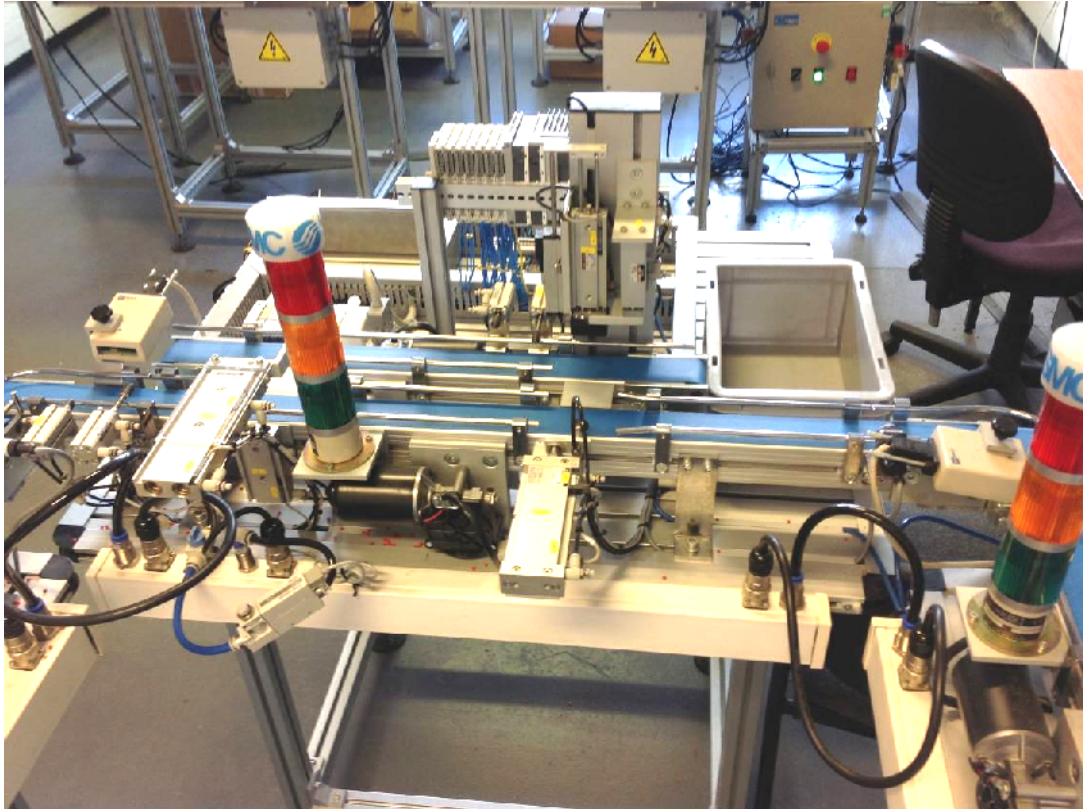
Τελευταίο κομμάτι του σταθμού είναι ο ιμάντας μεταφοράς. Τροφοδοτείται από έναν κινητήρα συνεχούς ρεύματος (DC) και περιέχει και επιπλέον μία βαλβίδα και έναν αισθητήρα για τον έλεγχο της ροής των κουτιών. Στην αρχή του ιμάντα υπάρχει ένας πνευματικός κύλινδρος τον οποίο ενεργοποιούμε όταν υπάρχει ροή απ τον προηγούμενο σταθμό και δεν θέλουμε να εμπλακεί με την ροή του σταθμού μας. Ο κύλινδρος αυτός λειτουργεί σαν πόρτα η οποία συγκρατεί όλα τα κουτιά που έρχονται από τον προηγούμενο σταθμό. Στο τέλος του ιμάντα υπάρχει ένας φωτοηλεκτρικός αισθητήρας ο οποίος ανιχνεύει αν υπάρχει ουρά στον επόμενο σταθμό και δίνει σήμα στο PLC για να αποφασίσει για τον ρυθμό που θα επεξεργάζεται τα κουτιά.

Όλα τα μέρη, βαλβίδες και αισθητήρια του σταθμού ελέγχονται από τα βιομηχανικά PC της εταιρίας Beckhoff (μοντέλο CX-1030) για τα οποία θα γίνει αναφορά με περισσότερες λεπτομέρειες σε επόμενο κεφάλαιο.

### 3.1.3 3<sup>ος</sup> σταθμός (σταθμός ελέγχου ύψους)

Ο τρίτος σταθμός είναι υπεύθυνος για τον έλεγχο του ύψους του υλικού σε κάθε κουτί [Εικόνα 13].

Στην αρχή του ιμάντα μεταφοράς του τρίτου σταθμού υπάρχουν δύο έμβολα που ελέγχονται από μία πνευματική βαλβίδα. Τα έμβολα αυτά χρησιμοποιούνται για να σταματάνε τα κουτιά που έρχονται από τον προηγούμενο σταθμό και να συγκρατούνε το πρώτο κουτί της ουράς μπροστά από τον αναγνώστη barcode. Σε εκείνο το σημείο ελέγχεται το barcode του κουτιού και αποφασίζεται αν θα συνεχίσει την πορεία του στους ιμάντες μεταφοράς για να ελεγχθεί το ύψος του υλικού ή αν θα προχωρήσει προς τον επόμενο σταθμό.



Εικόνα 13 - 3<sup>ος</sup> σταθμός

Εφόσον το κουτί είναι σωστό, απελευθερώνεται από τα έμβολα μπροστά από τον αναγνώστη barcode και σταματάει στο επόμενο έμβολο πάνω στον ιμάντα μεταφοράς όπου μέσω μίας πνευματικής βαλβίδας σπρώχνεται στον παράλληλο ιμάντα μεταφοράς για να συνεχίσει την πορεία του προς το έμβολο που θα κάνει τον έλεγχο του ύψους της στάθμης του υλικού.



Εικόνα 14 – ιμάντες μεταφοράς

Στον βοηθητικό ιμάντα μεταφοράς [Εικόνα 14] υπάρχουν επίσης δύο έμβολα που ελέγχονται από πνευματικές βαλβίδες και είναι υπεύθυνα για να παίρνουν κάθε φορά το πρώτο κουτί της ουράς και να προμηθεύουν με αυτό το έμβολο έλεγχο του ύψους της στάθμης. Το έμβολο αυτό αποτελείται από μία πνευματική βαλβίδα η οποία όταν ενεργοποιείται το μετακινεί κάθετα και από ένα αναλογικό

αισθητήριο το οποίο αναλόγως σε ποιο ύψος σταματήσει το έμβολο, παίρνει την αντίστοιχη τιμή και ενημερώνει το PLC [Εικόνα 15]. Αν η τιμή είναι μέσα στα αποδεκτά όρια τότε ένα έμβολο, το οποίο φέρει στην άκρη του μια βεντούζα αέρα, επεκτείνεται για να συγκρατήσει το κουτί το οποίο απελευθερώνεται από το έμβολο μέτρησης στάθμης και το τραβάει ξανά στον κύριο ιμάντα μεταφοράς. Σε περίπτωση που το κουτί δεν έχει την σωστή στάθμη, τότε απελευθερώνεται από το έμβολο μέτρησης στάθμης και φτάνει στο τέλος του βοηθητικού ιμάντα όπου και πέφτει σε έναν κάδο.



Εικόνα 15 – αναλογικός μετρητής ύψους

Στο σταθμό υπάρχουν επίσης τρία φώτα στα χρώματα του πράσινου, πορτοκαλί και κόκκινου τα οποία ανάβουν αναλόγως με την κατάσταση της διαδικασίας.

Οι ιμάντες μεταφοράς τροφοδοτούνται από έναν DC κινητήρα και περιέχει και ένα επιπλέον εξάρτημα για τον έλεγχο της ροής των κουτιών. Στο τέλος του κύριου ιμάντα υπάρχει ένας φωτοηλεκτρικός αισθητήρας ο οποίος ανιχνεύει αν υπάρχει ουρά στον επόμενο σταθμό και δίνει σήμα στο PLC.

Όλα τα μέρη, βαλβίδες και αισθητήρια του σταθμού ελέγχονται από τα βιομηχανικά PC της εταιρίας Beckhoff (μοντέλο CX-1030) για τα οποία θα γίνει αναφορά με περισσότερες λεπτομέρειες σε επόμενο κεφάλαιο.

#### **3.1.4 4<sup>ος</sup> σταθμός (σταθμός τοποθέτησης καλύμματος και ετικέτας)**

Ο τέταρτος σταθμός είναι υπεύθυνος για την τοποθέτηση καλύμματος στα κουτιά που είναι γεμάτα με προϊόν καθώς και την εκτύπωση αλλά και επικόλληση ετικέτας με συγκεκριμένα στοιχεία πάνω στο καπάκι [Εικόνα 16].

Στην αρχή του ιμάντα μεταφοράς του τέταρτου σταθμού υπάρχουν δύο έμβολα που ελέγχονται από μία πνευματική βαλβίδα. Τα έμβολα αυτά χρησιμοποιούνται για να συγκρατούν τα κουτιά που έρχονται από τον προηγούμενο σταθμό και να τοποθετούν το πρώτο κουτί της ουράς μπροστά από τον αναγνώστη barcode. Σε εκείνο το σημείο ελέγχεται το barcode του κουτιού και αποφασίζεται αν θα γίνει επεξεργασία του κουτιού από τον σταθμό, δηλαδή αν θα τοποθετηθεί καπάκι και ετικέτα ή αν θα προχωρήσει προς τον επόμενο σταθμό.



Εικόνα 16 4<sup>ος</sup> σταθμός

Στην συνέχεια υπάρχει ένα έμβολο που κινείται κάθετα στον ιμάντα μεταφοράς και σταματάει το κουτί στην μέση του ιμάντα. Εκεί ενεργοποιείται ένα ακόμα πνευματικό έμβολο το οποίο σπρώχνει το κουτί σε μία πλατφόρμα στο πλάι του ιμάντα μεταφοράς.

Την ίδια στιγμή ενεργοποιείται και ο μηχανισμός που τροφοδοτεί τον σταθμό με καπάκια. Πιο συγκεκριμένα, τα καπάκια συγκρατούνται σε δύο πλαστικές στήλες, το ένα πάνω από το άλλο. Στο κάτω μέρος της κάθε στήλης υπάρχει ένα άνοιγμα που επιτρέπει σε ένα μόνο καπάκι να βγει από την στήλη. Ένα πνευματικό έμβολο που έχει στην άκρη του μία μεταλλική λεπίδα βρίσκεται από την άλλη άκρη των στηλών και προεκτείνεται με αποτέλεσμα να βγάλει το καπάκι από τον πάτο της στήλης και να το εναποθέσει στην πλατφόρμα που βρίσκεται μπροστά από τις στήλες. Τότε ενεργοποιείται ένας συνδυασμός πνευματικών βαλβίδων και εμβόλων που βρίσκεται πάνω από τον σταθμό και κινείται παράλληλα με τον ιμάντα μεταφοράς. Για την μετακίνηση του καπακιού το οποίο βρίσκεται έτοιμο στην πλατφόρμα χρησιμοποιείται ένα έμβολο το οποίο κινείται κάθετα και στην άκρη του έχει μία βεντούζα αέρα η οποία ενεργοποιείται και συγκρατεί το καπάκι. Το έμβολο ξανά κινείται προς τα πάνω και μετά ολόκληρο το σύστημα κινείται προς το πλάι για να βρεθεί το καπάκι πάνω από το κουτί και να κουμπώσει.

Στην άλλη άκρη του σταθμού βρίσκεται ένας εκτυπωτής ετικετών ο οποίος επικοινωνεί μέσω της σειριακής θύρας RS-232 με τον ελεγκτή [Εικόνα 17]. Η εκτύπωση γίνεται όταν τοποθετηθεί το καπάκι στο κουτί. Την ετικέτα την παίρνει ένα δεύτερο σύστημα με έμβολο που έχει στην άκρη του τρεις μικρές βεντούζες αέρα για να συγκρατούν την ετικέτα χωρίς να την χαλάσουν. Το δεύτερο αυτό σύστημα βρίσκεται στο σύστημα που υπάρχει πάνω από τον σταθμό και κινείται παράλληλα με τον ιμάντα μεταφοράς.



Εικόνα 17 – εκτυπωτής ετικετών RS-232

Το επόμενο κομμάτι που υπάρχει στον σταθμό είναι το έμβολο που κινείται κάθετα στον ιμάντα μεταφοράς και έχει στην άκρη του μία βεντούζα αέρα. Το τρίτο αυτό σύστημα με βεντούζα χρησιμοποιείται για να επαναφέρει το κουτί που είχε σπρώξει από τον ιμάντα στην πλατφόρμα, και πάλι πίσω στον ιμάντα μεταφοράς.

Στο σταθμό υπάρχουν επίσης τρία φώτα με χρώμα πράσινο, πορτοκαλί και κόκκινο τα οποία ανάβουν με συγκεκριμένη σειρά αναλόγως με την κατάσταση της διαδικασίας.

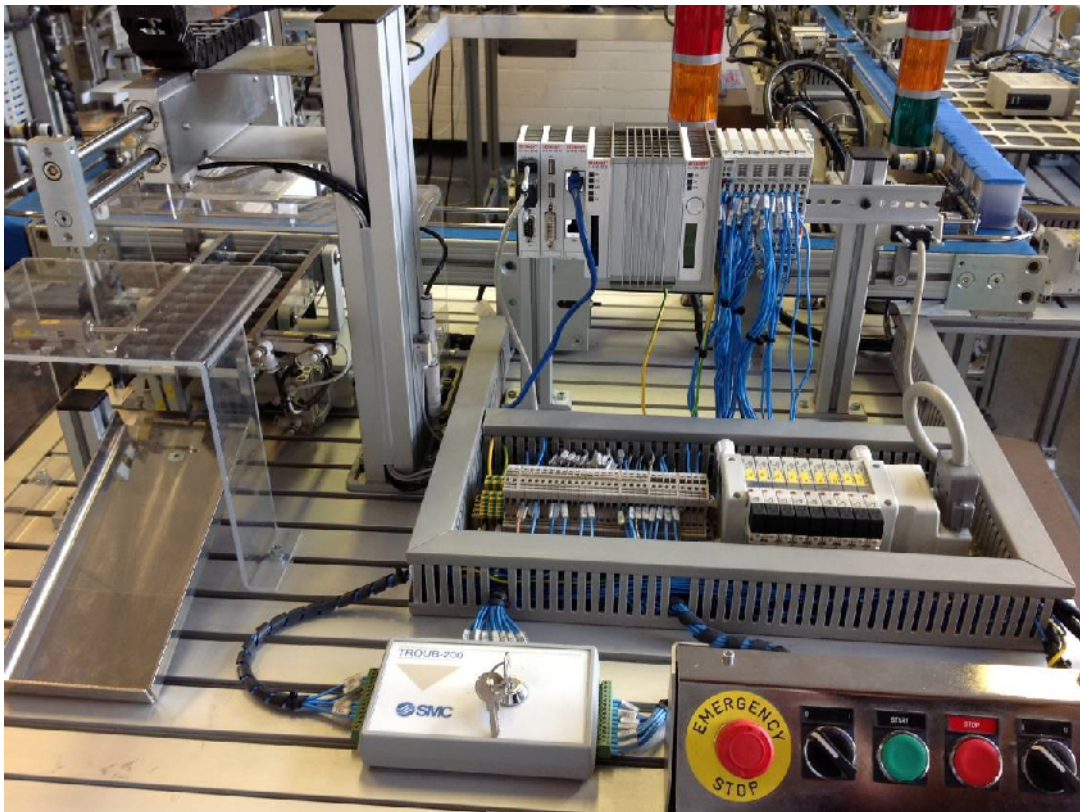
Τελευταίο κομμάτι του σταθμού είναι ο ιμάντας μεταφοράς. Τροφοδοτείται από έναν κινητήρα συνεχούς ρεύματος (DC). Στο τέλος του ιμάντα υπάρχει ένας φωτοηλεκτρικός αισθητήρας ο οποίος ανιχνεύει αν υπάρχει ουρά στον επόμενο σταθμό και δίνει σήμα στο PLC για να αποφασίσει για τον ρυθμό που θα επεξεργάζεται τα κουτιά.

Όλα τα μέρη, βαλβίδες και αισθητήρια του σταθμού ελέγχονται από τα βιομηχανικά PC της εταιρίας Beckhoff (μοντέλο CX-1030) για τα οποία θα γίνει αναφορά με περισσότερες λεπτομέρειες σε επόμενο κεφάλαιο.

### 3.1.5 5<sup>ος</sup> σταθμός (σταθμός αποστολής προϊόντων)

Ο πέμπτος σταθμός είναι υπεύθυνος για την αποστολή των κουτιών που είναι τελείως έτοιμα [Εικόνα 18].

Στην αρχή του ιμάντα μεταφοράς του πέμπτου σταθμού υπάρχουν δύο έμβολα που ελέγχονται από μία πνευματική βαλβίδα. Η χρήση των εμβόλων αυτών γίνεται για να συγκρατούν τα κουτιά που έρχονται από τον προηγούμενο σταθμό και να σταματάνε το πρώτο κουτί ακριβώς μπροστά από τον αναγνώστη barcode. Σε εκείνο το σημείο ελέγχεται το barcode του κουτιού και αποφασίζεται αν θα γίνει επεξεργασία του κουτιού από τον σταθμό, δηλαδή αν θα προχωρήσει σε αποστολή ή αν θα προχωρήσει προς τον επόμενο σταθμό.



Εικόνα 18 - 5<sup>ος</sup> σταθμός

Όταν το κουτί ελευθερωθεί από τα έμβολα που λειτουργούν σαν πόρτες, προχωράει μέσω του ιμάντα μεταφοράς προς την συνέχεια του σταθμού. Λίγο πριν το τέλος του σταθμού υπάρχει ένα πνευματικό έμβολο κάθετο στον σταθμό το οποίο σταματάει το κουτί ακριβώς μπροστά από την πλατφόρμα φόρτωσης. Σε εκείνο το σημείο προεκτείνεται στον οριζόντιο άξονα το πνευματικό έμβολο που βρίσκεται από πάνω της και στην συνέχεια προεκτείνει ένα μικρότερο έμβολο στον κάθετο άξονα το οποίο στο τέλος του έχει μία βεντούζα αέρα. Με αυτό τον τρόπο συγκρατεί το κουτί και το τοποθετεί πάνω στην πλατφόρμα φόρτωσης.





Εικόνα 19 – πλατφόρμα αποστολής

Η πλατφόρμα [Εικόνα 19] αποκτά κλίση με την βοήθεια δύο εμβόλων που βρίσκονται στην άκρη της ούτως ώστε να μετακινούνται τα κουτιά στο τέλος της και επίσης έχει δύο μικρά πνευματικά έμβολα στην άκρη που λειτουργούν σαν πόρτες και ανοίγουν μόνο την στιγμή που η πλατφόρμα είναι γεμάτη και έτοιμη να αποστείλει όλα τα κουτιά.

Στο σταθμό υπάρχουν ακόμα τρία φώτα χρώματος πράσινου, πορτοκαλί και κόκκινου τα οποία ενεργοποιούνται με συγκεκριμένη σειρά αναλόγως με την κατάσταση της διαδικασίας.

Τελευταίο κομμάτι του σταθμού είναι ο ιμάντας μεταφοράς. Τροφοδοτείται από έναν κινητήρα συνεχούς ρεύματος (DC). Στο τέλος του ιμάντα υπάρχει ένας φωτοηλεκτρικός αισθητήρας ο οποίος ανιχνεύει αν υπάρχει ουρά στον επόμενο σταθμό και δίνει σήμα στο PLC για να αποφασίσει για τον ρυθμό που θα επεξεργάζεται τα κουτιά σε περίπτωση που κάποιο κουτί χρειάζεται απλά να περάσει από τον σταθμό και να μην αποσταλθεί.

Όλα τα μέρη, βαλβίδες και αισθητήρια του σταθμού ελέγχονται από τα βιομηχανικά PC της εταιρίας Beckhoff (μοντέλο CX-1030) για τα οποία θα γίνει αναφορά με περισσότερες λεπτομέρειες σε επόμενο κεφάλαιο.

### **3.2 Περιφερειακά-Βοηθητικά μέρη**

Σε αυτό το υποκεφάλαιο θα γίνει αναφορά και ανάλυση των λειτουργιών των βοηθητικών μερών του σταθμού. Πιο συγκεκριμένα, είναι δύο κομμάτια του σταθμού που χρησιμοποιούνται για την ομαλή λειτουργία της παραγωγικής διαδικασίας και είναι τα φώτα και τα κουμπιά του σταθμού.

### 3.2.1 Φώτα

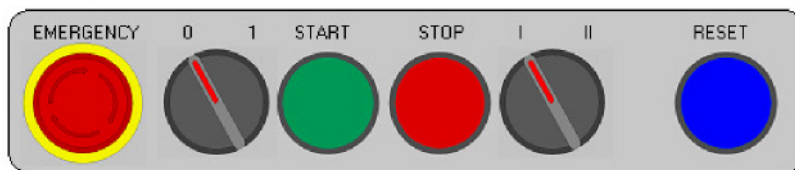
Στο σταθμό υπάρχουν τρία φώτα στα χρώματα του πράσινου, πορτοκαλί και κόκκινου. Τα φώτα αυτά ανάβουν αναλόγως με την κατάσταση της διαδικασίας. Αναλυτικά παρακάτω είναι ένας πίνακας με το χρώμα των φώτων που είναι αναμμένα ή αναβοσβήνουν σε κάθε περίπτωση και για κάθε έναν από τους προαναφερθέντες σταθμούς.

Πίνακας 1

Σταθμός	Πράσινο σταθερά αναμμένο	Κόκκινο που αναβοσβήνει	Κόκκινο σταθερά αναμμένο	Κόκκινο-Πορτοκαλί που αναβοσβήνει	Πορτοκαλί σταθερά αναμμένο
1	Σταθμός έτοιμος/ κανονική λειτουργία	Απόρριψη κουτιού/ Emergency Stop / πέντε λάθος κουτιά	Δεν υπάρχει κουτί	Δεν υπάρχουν κουτιά στις στήλες	Χαμηλό απόθεμα κουτιών στις στήλες
2	Σταθμός έτοιμος/ κανονική λειτουργία	Απόρριψη κουτιού/Emergency Stop / πέντε λάθος κουτιά	Απόρριψη κουτιού	Δεν υπάρχουν κουτιά στις στήλες	Χαμηλό απόθεμα κουτιών στις στήλες
3	Σταθμός έτοιμος/ κανονική λειτουργία	Απόρριψη κουτιού/Emergency Stop / πέντε λάθος κουτιά	Πάτημα start πάνω από 5 φορές	Ουρά κουτιών στον επόμενο σταθμό	
4	Σταθμός έτοιμος/ κανονική λειτουργία	Απόρριψη κουτιού/Emergency Stop / Δεν έφυγε το κουτί από το σταθμό	Πάτημα start πάνω από 3 φορές	Ουρά κουτιών στον επόμενο σταθμό / Δεν υπάρχουν διαθέσιμα καπάκια	
5	Σταθμός έτοιμος/ κανονική λειτουργία	Emergency Stop	Πάτημα start πάνω από 3 φορές	Ουρά κουτιών στον επόμενο σταθμό	

### 3.2.2 Κουμπιά

Ο σταθμός περιέχει μία κονσόλα με έξι κουμπιά [Εικόνα 20].



Εικόνα 20 – διακόπτες ελέγχου

**Διακόπτης Off/On (0/1)** – Αυτός ο διακόπτης δίνει ρεύμα στον σταθμό. Όταν ο διακόπτης είναι στη θέση 1, οι μάντες μεταφοράς του κάθε σταθμού δουλεύουν κανονικά, το PLC τροφοδοτείται με ρεύμα και είναι αναμμένο τουλάχιστον ένα από τα φώτα αναλόγως με την κατάσταση του σταθμού.

**Start** – Αυτό το κουμπί χρησιμοποιείται για να ξεκινήσει την διαδικασία του σταθμού και να ελέγξει τον τρόπο λειτουργίας του. Αυτό εξαρτάται από τον αριθμό των φορών που θα πατηθεί το κουμπί.

Πίνακας 2

Αριθμός πιέσεων του κουμπιού	Σταθμοί				
	1	2	3	4	5
1 φορά	Έναρξη διαδικασίας	Έναρξη διαδικασίας και γέμισμα με 15g προϊόντος	Έναρξη διαδικασίας και γέμισμα με 15g προϊόντος	Έναρξη διαδικασίας	Έναρξη διαδικασίας
2 φορές		Έναρξη διαδικασίας και γέμισμα με 30g προϊόντος	Έναρξη διαδικασίας και γέμισμα με 30g προϊόντος		
3 φορές		Έναρξη διαδικασίας και γέμισμα με 45g προϊόντος	Έναρξη διαδικασίας και γέμισμα με 45g προϊόντος		
4 φορές	Απελευθέρωση κουτιού για να διασχίσει τον σταθμό	Απελευθέρωση κουτιού για να διασχίσει τον σταθμό	Απελευθέρωση κουτιού για να διασχίσει τον σταθμό	Απελευθέρωση κουτιού για να διασχίσει τον σταθμό	Απελευθέρωση κουτιού για να διασχίσει τον σταθμό

**Emergency Stop** – Πατώντας αυτό το κουμπί σταματάει αμέσως η λειτουργία του σταθμού. Εμποδίζει οποιοδήποτε άλλο κουμπί να πατηθεί για λόγους ασφαλείας και προκειμένου να επανέλθει ο σταθμός στην κανονική λειτουργία του πρέπει να απελευθερωθεί το emergency stop και να πατηθούν ταυτόχρονα το κουμπί stop και reset. Στην συνέχεια όλοι οι μηχανισμοί του σταθμού επανέρχονται στην αρχική τους θέση και λειτουργία και το σύστημα

περιμένει και πάλι να πατηθεί το κουμπί start για να ξεκινήσει την επόμενη διαδικασία του.

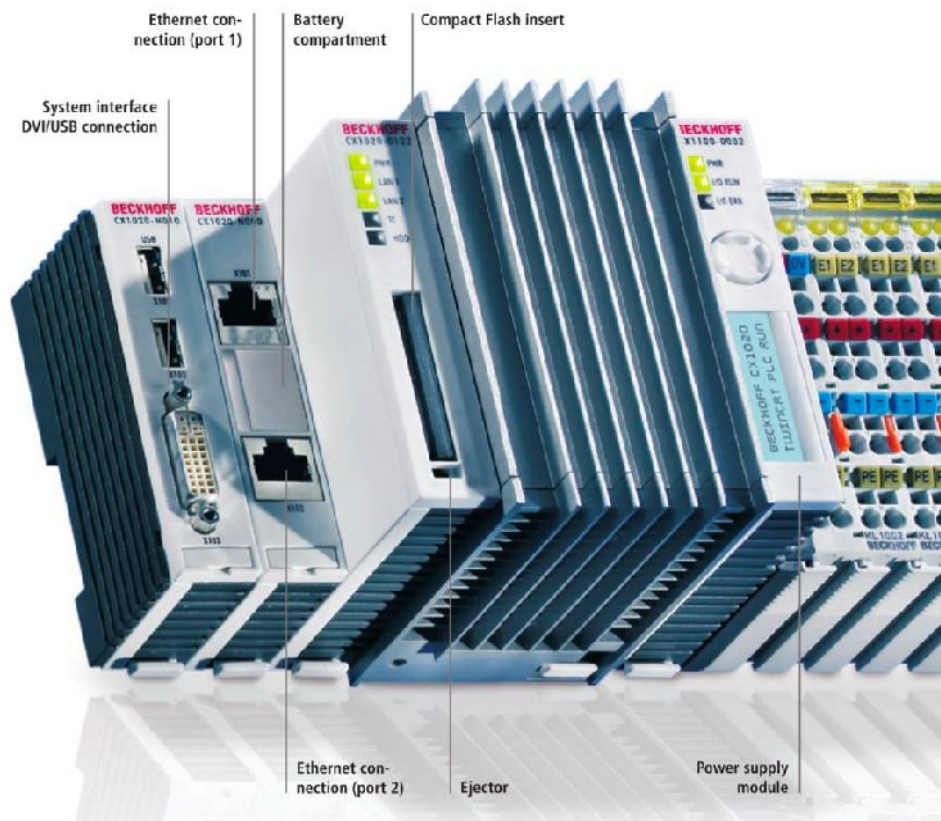
**Reset** – Στους σταθμούς 2 και 3 μετά από πέντε λανθασμένα κουτιά ο σταθμός σταματάει την λειτουργία του και περιμένει να πατηθεί το κουμπί reset για να επανέλθει σε λειτουργία αναμονής για το κουμπί start. Αυτό συμβαίνει για να επιστήσει την προσοχή του χρήστη για τον αριθμό των λάθος κουτιών.

### 3.3 Beckhoff

Beckhoff είναι η επωνυμία της εταιρίας από την οποία προμηθευτήκαμε τους ελεγκτές και τα υπόλοιπα εξαρτήματα για την επικοινωνία των ελεγκτών με τα υπόλοιπα μέρη του σταθμού καθώς επίσης και το λογισμικό για την εγκατάσταση, παραμετροποίηση και προγραμματισμό αυτών των ελεγκτών. Στην συνέχεια θα γίνει ανάλυση όλων αυτών των εξαρτημάτων καθώς και μια παρουσίαση του λογισμικού που χρησιμοποιήθηκε.

#### 3.3.1 Controllers

Για τον προγραμματισμό και έλεγχο των σταθμών χρησιμοποιούνται οι ελεγκτές CX-1030 της εταιρίας Beckhoff [Εικόνα 21]. Πρόκειται για PLC τα οποία είναι βιομηχανικού τύπου υπολογιστές με λειτουργικό embedded windows XP. Έχουν επεξεργαστή 1.8 GHz Intel® Pentium® M και μνήμη RAM 256MB. Διαθέτουν θύρα για κάρτα τύπου Compact Flash στην οποία υπάρχει εγκατεστημένο το λειτουργικό σύστημα.



Εικόνα 21 – Ελεγκτής Beckhoff CX-1030

Έχουν επίσης δύο θύρες δικτύου Ethernet 10/100 Mbit/s οι οποίες χρησιμοποιούνται για την επικοινωνία με υπολογιστή, για δημιουργία δικτύου μεταξύ τους ή σύνδεση με άλλες εξωτερικές συσκευές ελέγχου. Ακόμα έχουν έξοδο DVI για σύνδεση με εξωτερική οθόνη για να είναι δυνατός ο έλεγχος του PLC χωρίς την ανάγκη ύπαρξης εξωτερικού υπολογιστή καθώς και δύο θύρες USB που χρησιμοποιούνται για σύνδεση περιφερειακών συσκευών εισόδου-εξόδου όπως πληκτρολόγιο και ποντίκι ή οθόνη αφής. Τέλος έχουν δύο θύρες σειριακής επικοινωνίας RS-232 για σύνδεση αισθητηρίων και συσκευών που επικοινωνούν σειριακά με το PLC, όπως αναγνώστης barcode, ζυγαριά ή εκτυπωτής.

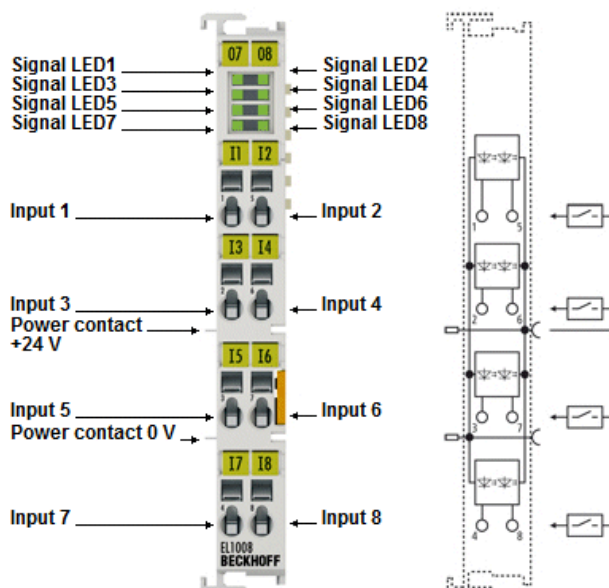
Οι ελεγκτές αυτοί, βάση και των χαρακτηριστικών που μόλις αναφέρθηκαν, αποτελούν την πιο εξελιγμένη τεχνολογία όσον αφορά τα PLC. Πιο συγκεκριμένα, προσφέρουν καλύτερους χρόνους απόκρισης που φτάνουν μέχρι και τα 50  $\mu$ s, έχουν μεγαλύτερες δυνατότητες αποθήκευσης σε σχέση με τα συμβατικά PLC. Ακόμα, μπορούν να δουλεύουν και να προγραμματίζονται ανεξάρτητοι και χωρίς να είναι υποχρεωτική η χρήση υπολογιστή ο οποίος να είναι συνδεδεμένος πάνω τους και έχουν πολύ περισσότερες δυνατότητες διασύνδεσης με περιφερειακές συσκευές.

### 3.3.2 Modules

Οι ελεγκτές που έχουμε στην διάθεση μας υποστηρίζουν έναν μεγάλο αριθμό modules που χρησιμοποιούνται για σύνδεση inputs και outputs ή και εξωτερικών συσκευών. Στο σύστημα μας κάνουμε χρήση των ακόλουθων modules.

- EL1008 [Εικόνα 22]

Το συγκεκριμένο module χρησιμοποιείται για την είσοδο των ψηφιακών σημάτων (Digital input terminal) από τους διάφορους αισθητήρες προς τον ελεγκτή.

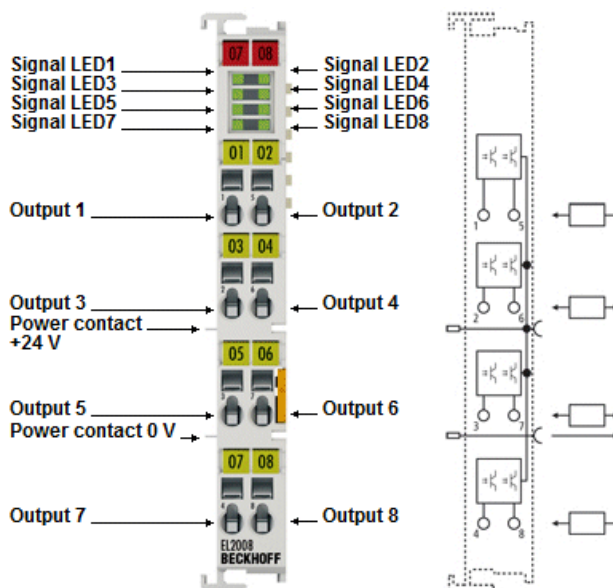


Εικόνα 22 – EL1008

Αποτελείται από οκτώ εισόδους και σε κάθε μία αντιστοιχεί ένα λαμπάκι για την ένδειξη της κατάστασης της. Εκτός από τις οκτώ εισόδους και τα αντίστοιχα λαμπάκια, το module φέρει επίσης και δύο ακροδέκτες από τους οποίους ο ένας τροφοδοτείται με τάση από τον ελεγκτή ή το προηγούμενο module και ο άλλος είναι η γείωση. Από την άλλη μεριά του module υπάρχουν αντίστοιχα άλλοι δύο ακροδέκτες για την σύνδεση με το επόμενο module. Η τάση που χρειάζεται για να λειτουργήσει είναι τα 24V συνεχούς ρεύματος.

- EL2008 [Εικόνα 23]

Το συγκεκριμένο module χρησιμοποιείται για την έξοδο των ψηφιακών σημάτων (Digital output terminal) από τον ελεγκτή προς τους διάφορους μηχανισμούς κίνησης. Αποτελείται από οκτώ εξόδους και σε κάθε μία αντιστοιχεί ένα λαμπάκι για την ένδειξη της κατάστασης της. Εκτός από τις οκτώ εξόδους και τα αντίστοιχα λαμπάκια, το module φέρει επίσης και δύο ακροδέκτες από τους οποίους ο ένας τροφοδοτείται με τάση από τον ελεγκτή ή το προηγούμενο module και ο άλλος είναι η γείωση. Από την άλλη μεριά του module υπάρχουν αντίστοιχα άλλοι δύο ακροδέκτες για την σύνδεση με το επόμενο module. Η τάση που χρειάζεται για να λειτουργήσει είναι τα 24V συνεχούς ρεύματος.



Εικόνα 23 – EL2008

- EL3102 [Εικόνα 24]

Το συγκεκριμένο module χρησιμοποιείται για την είσοδο των αναλογικών σημάτων (Analog input terminal) από τα διάφορα αναλογικά αισθητήρια προς τον ελεγκτή. Αποτελείται από δύο εισόδους και κάθε μία έχει από δύο αναμονές για την τάση και το σήμα από το αισθητήριο. Το module φέρει επίσης και δύο ακροδέκτες από τους οποίους ο ένας τροφοδοτείται με τάση από τον ελεγκτή ή το προηγούμενο module και ο άλλος είναι η γείωση. Από

την άλλη μεριά του module υπάρχουν αντίστοιχα άλλοι δύο ακροδέκτες για την σύνδεση με το επόμενο module. Η τάση που χρειάζεται για να λειτουργήσει είναι τα 24V συνεχούς ρεύματος.



Εικόνα 24 – EL3102

### 3.3.3 Twincat Software

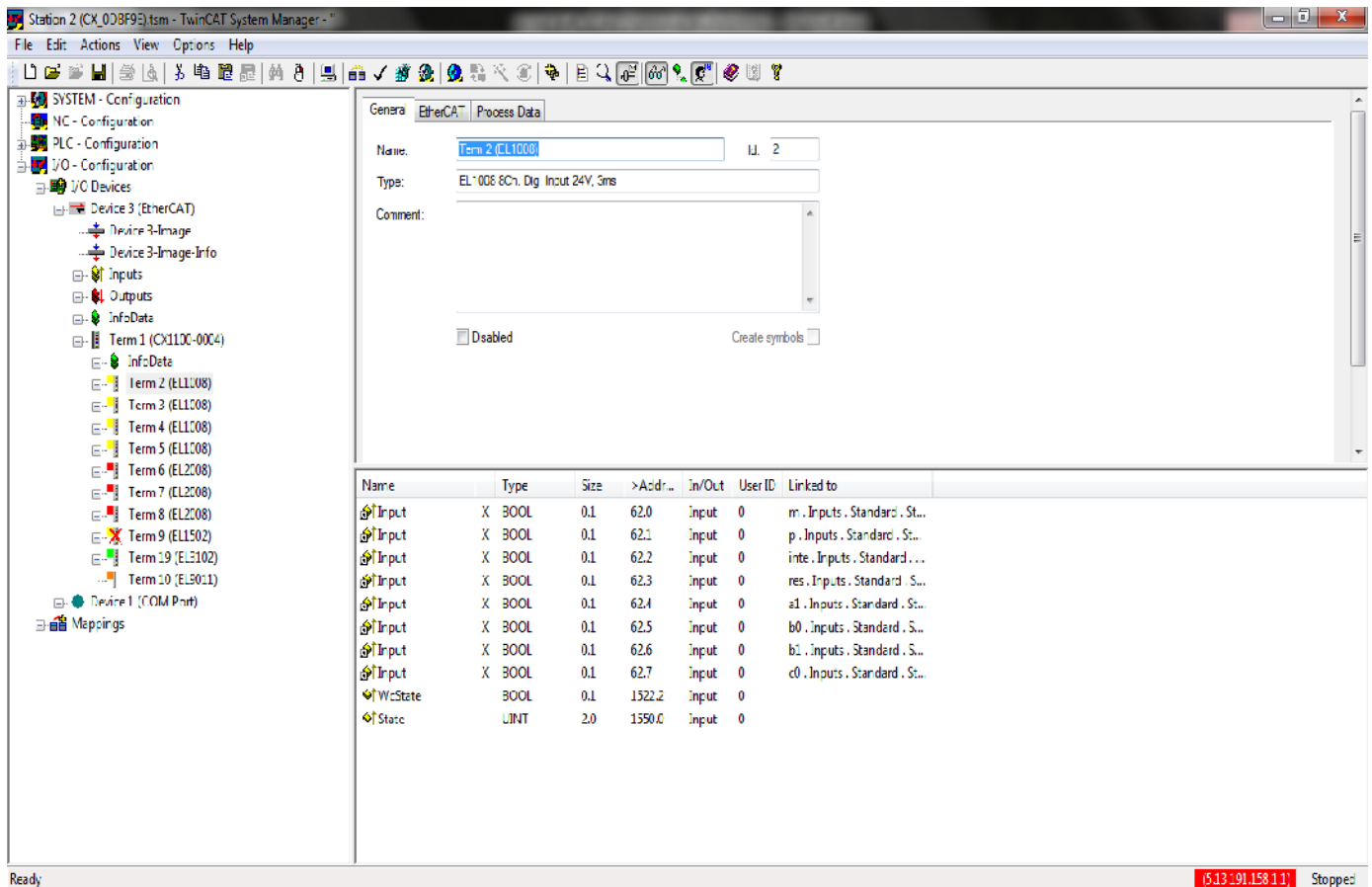
Έτσι ονομάζεται το πρόγραμμα της εταιρίας Beckhoff το οποίο χρησιμοποιείται για την επικοινωνία του υπολογιστή με το PLC και τα επιμέρους modules του, την παραμετροποίηση του συστήματος καθώς και για τον προγραμματισμό του. Αποτελείται από δύο εφαρμογές, το System Manager και το PLC Control.

#### 3.2.3.1 System Manager

Η πρώτη είναι η εφαρμογή που αναγνωρίζει το hardware που υπάρχει συνδεδεμένο με τον ελεγκτή [Εικόνα 25]. Η επικοινωνία ανάμεσα στον υπολογιστή και στον ελεγκτή-PLC γίνεται μέσω δικτύου ethernet. Πιο συγκεκριμένα, κάθε ελεγκτής έχει μοναδική διεύθυνση IP και έτσι με την σύνδεση όλων των ελεγκτών και του υπολογιστή ή υπολογιστών που τους διαχειρίζονται, στο ίδιο δίκτυο, αποκτούμε πρόσβαση σε κάθε ελεγκτή.

Χρησιμοποιώντας αυτή την εφαρμογή έχουμε την δυνατότητα να αναγνωρίσουμε όλα τα modules που είναι συνδεδεμένα με τον ελεγκτή καθώς και τις δυνατότητες αλλά και τα τεχνικά χαρακτηριστικά τους. Με αυτόν τον τρόπο αποκτούμε πλήρη πρόσβαση στον ελεγκτή και εν συνέχεια σε όλα τα αισθητήρια και μηχανισμούς που είναι συνδεδεμένα πάνω του. Μπορούμε δηλαδή ανά πάσα στιγμή να αναγνωρίζουμε την κατάσταση του κάθε αισθητηρίου και τις τυχόν μεταβολές αυτής. Ακόμα κάνουμε την σύνδεση των μεταβλητών του προγράμματος μας με συγκεκριμένα inputs και outputs των modules του ελεγκτή και μπορούμε να

αντιστοιχήσουμε την τιμή των αισθητηρίων με τις αντίστοιχες μεταβλητές του προγράμματος μας.



Εικόνα 25 – System manager

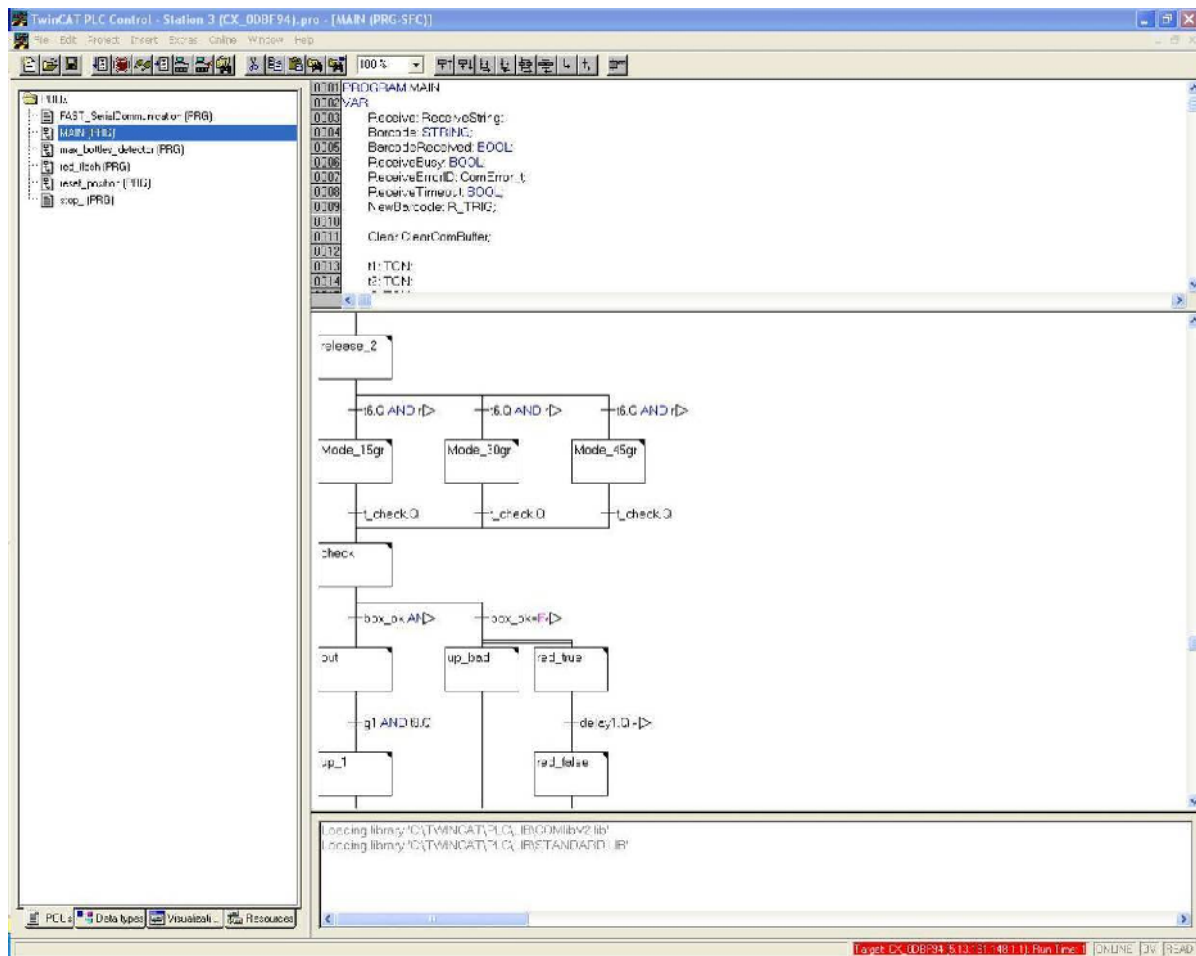
### 3.2.3.2 PLC Control

Η δεύτερη εφαρμογή ονομάζεται PLC control [Εικόνα 26]. Αποτελεί το εργαλείο προγραμματισμού του ελεγκτή/PLC. Από την στιγμή που το πρόγραμμα System Manager συνδεθεί με τον ελεγκτή που θέλουμε να προγραμματίσουμε, μας δίνεται η επιλογή στην συνέχεια να συνδέσουμε και την εφαρμογή PLC Control με τον αντίστοιχο ελεγκτή.

Οι γλώσσες προγραμματισμού PLC που μπορούν να χρησιμοποιηθούν για τον προγραμματισμό των ελεγκτών αυτών είναι όλες οι βασικές γλώσσες, αλλά αυτές που χρησιμοποιούμε στην συγκεκριμένη εργασία είναι οι SFC (Sequential Function Chart) και ST (Structured Text).

Μερικές παραπάνω λεπτομέρειες που είναι καλό να γνωρίζουμε για αυτές τις γλώσσες είναι ότι αρχικά η SFC είναι μία γραφική γλώσσα προγραμματισμού. Αποτελείται από “κουτιά”, τα λεγόμενα “steps” τα οποία συνδέονται με συγκεκριμένες ενέργειες. Μέσα σε κάθε κουτί είναι γραμμένο ένα κομμάτι κώδικα σε γλώσσα ST, την οποία θα αναλύσουμε αμέσως μετά, και είναι ο κώδικας που εκτελείται όταν το κουτί αυτό είναι ενεργό. Από κουτί σε κουτί υπάρχουν οι συνθήκες που πρέπει να είναι αληθείς ούτως ώστε να προχωρήσει το πρόγραμμα.





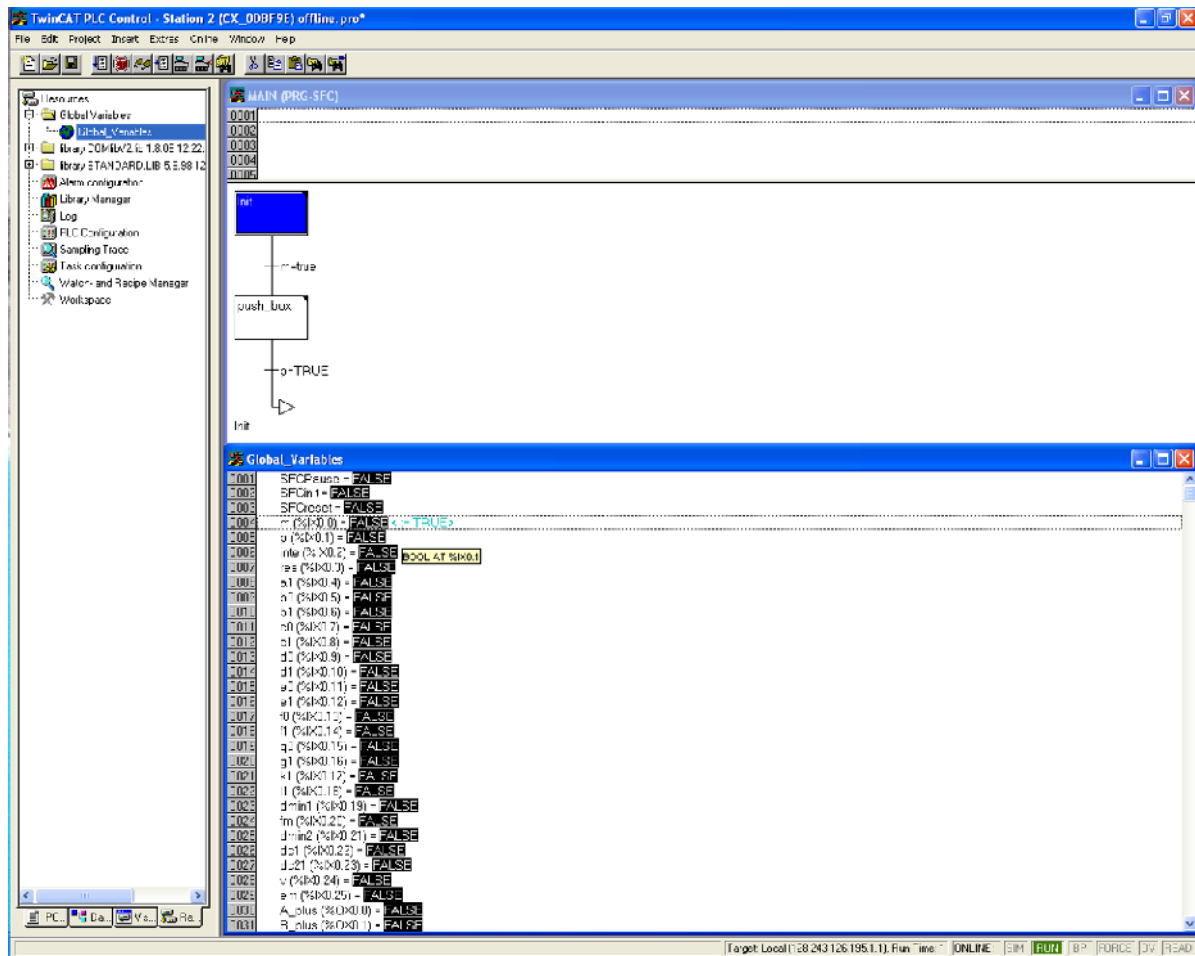
Εικόνα 26 – PLC control

Η γλώσσα προγραμματισμού ST είναι μία γλώσσα υψηλού επιπέδου που χρησιμοποιείται για τον προγραμματισμό PLC. Βασίζεται στην δομή των block και συντακτικά μοιάζει με την Pascal γιατί έχει βασιστεί σε αυτή. Κάνει χρήση όλων των πολύπλοκων δομών, όπως Repeat-Until, While-Do, If-Then-Else, Case και διαφόρων άλλων.

Όσον αφορά την συγκεκριμένη εφαρμογή, αρχικά δηλώνουμε τις μεταβλητές που θα χρησιμοποιήσουμε στο πρόγραμμα μας ούτως ώστε να μπορέσουμε να τις συνδέσουμε με τα inputs και outputs του συστήματος μέσω του system manager. Για τις μεταβλητές πρέπει να δηλωθεί συγκεκριμένα ο τύπος τους, η μορφή τους και το μέγεθος τους. Στην συνέχεια προγραμματίζουμε τον ελεγκτή μας χρησιμοποιώντας όλες τις ιδιότητες των παραπάνω γλωσσών και βάση της λογικής που θέλουμε να εκτελεί ο κάθε σταθμός. Έχουμε την δυνατότητα να κάνουμε και χρήση μετρητών χρόνου (timers), μετρητών (counters), καθώς επίσης και διάφορες άλλες υπορουτίνες που μας προσφέρει το λογισμικό.

Ολοκληρώνοντας τον προγραμματισμό αλλά και ανά πάσα στιγμή, έχουμε την δυνατότητα να “κατεβάσουμε” το πρόγραμμα στον ελεγκτή και να ελέγξουμε κατά πόσο λειτουργεί [Εικόνα 27]. Έχουμε επίσης την δυνατότητα να έχουμε την λίστα με όλες τις μεταβλητές του συστήματος που έχουμε δηλώσει και την ώρα που

τρέχει το πρόγραμμα μπορούμε να παρακολουθούμε τις πραγματικές τους τιμές και καταστάσεις.



Εικόνα 27 – PLC control 2

## 4. Λειτουργία σταθμών

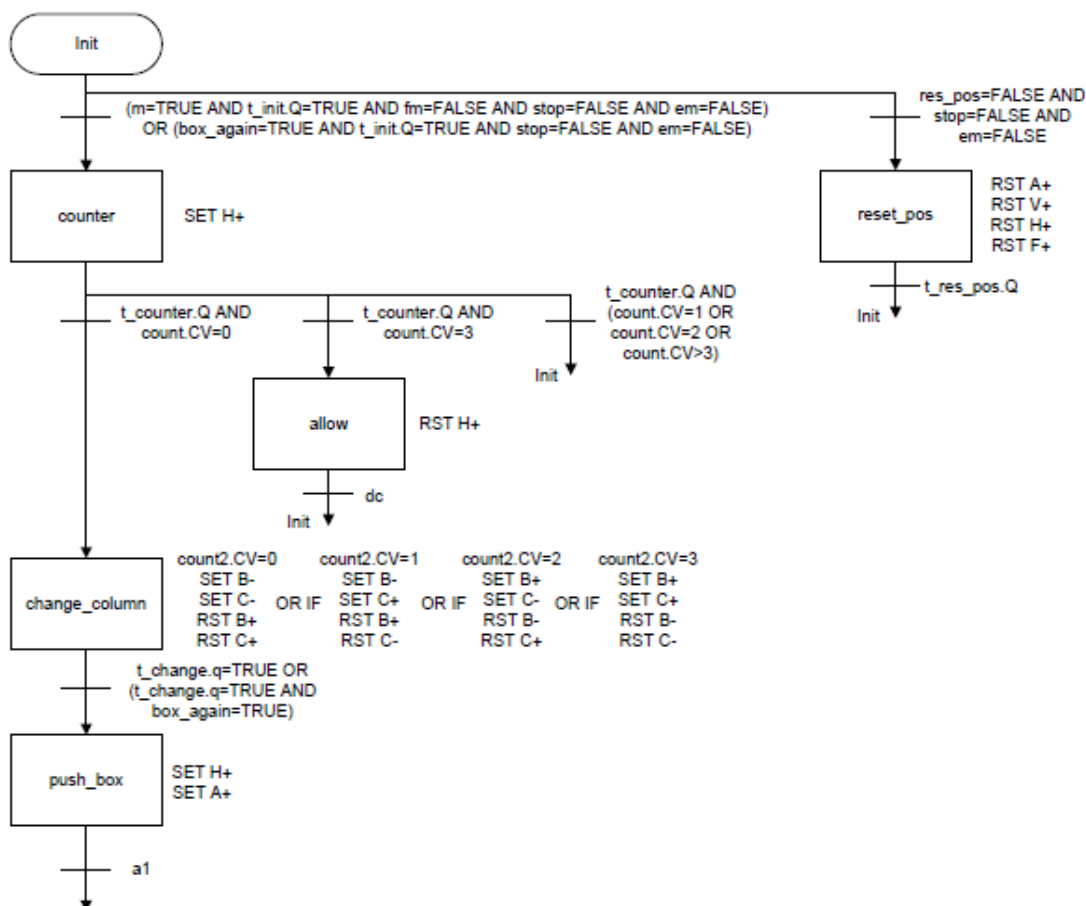
Σε αυτό το κεφάλαιο γίνεται παρουσίαση του κώδικα που γράφτηκε για τον έλεγχο των σταθμών. Ο κώδικας είναι γραμμένος σε γλώσσα προγραμματισμού PLC, και πιο συγκεκριμένα, SFC (Sequential Function Chart) και ST (Structured Text).

Αναφέρονται όλες οι διαδικασίες που περιλαμβάνονται στον σταθμό, καθώς επίσης και οι διαδικασίες άμεσης ανάγκης και ασφαλείας που ακολουθούνται στους ακόλουθους σταθμούς.

### 4.1 Σχεδίαση και εφαρμογή κώδικα ελέγχου

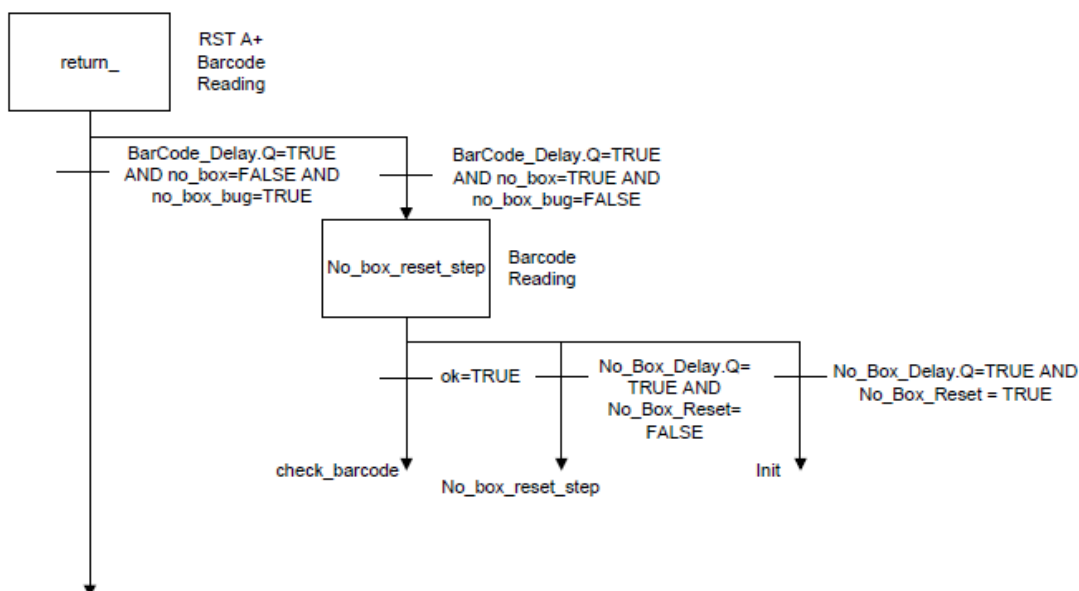
#### 4.1.1 1ος σταθμός

Ο πρώτος σταθμός είναι υπεύθυνος για να τροφοδοτεί την γραμμή παραγωγής με κουτιά συγκεκριμένου τύπου (με πολύχρωμη ετικέτα). Αναλυτικά η λειτουργία του σταθμού είναι η εξής [Εικόνα 28 - Εικόνα 33].



Εικόνα 28 – 1<sup>ος</sup> σταθμός (1/4)

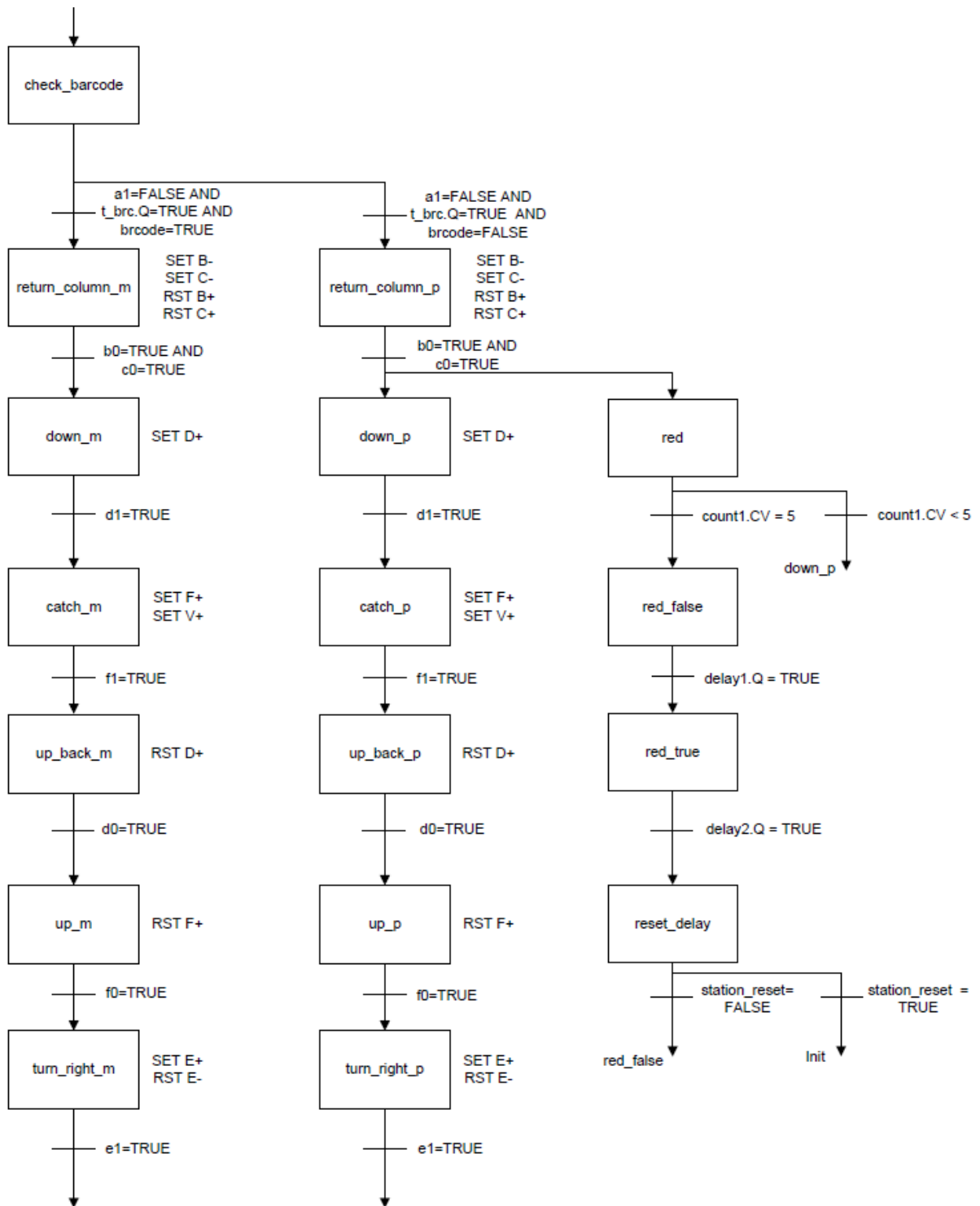
Πατώντας το κουμπί start ο σταθμός ξεκινά την λειτουργία του και σπρώχνει ένα κουτί από την πρώτη στήλη, στην πλατφόρμα μπροστά από τον αναγνώστη barcode. Στην συνέχεια διαβάζεται το barcode και αποφασίζεται βάση μίας λίστας με τους σωστούς αριθμούς, αν είναι επιθυμητό ή όχι. Σε όλη την διάρκεια υπάρχει αναμμένο το πράσινο φως στο φανάρι.



Εικόνα 29 - 1<sup>ος</sup> σταθμός (2/4)

Εάν το κουτί έχει λάθος barcode τότε αναβοσβήνει το κόκκινο φως στο φανάρι και το κουτί μεταφέρεται στον κάδο με τα άχρηστα κουτιά. Σε αυτή την περίπτωση δεν χρειάζεται να ξαναπατηθεί το κουμπί start και ο σταθμός πηγαίνει κατευθείαν και σπρώχνει ένα κουτί από την επόμενη στήλη. Η διαδικασία αυτή συνεχίζεται μέχρι να βρεθεί ένα σωστό κουτί ή πέντε λάθος. Στην περίπτωση που βρεθεί σωστό κουτί μεταφέρεται στον μάντα μεταφοράς και καθ' όλη την διάρκεια αυτή παραμένει αναμμένο το πράσινο φως στο φανάρι.

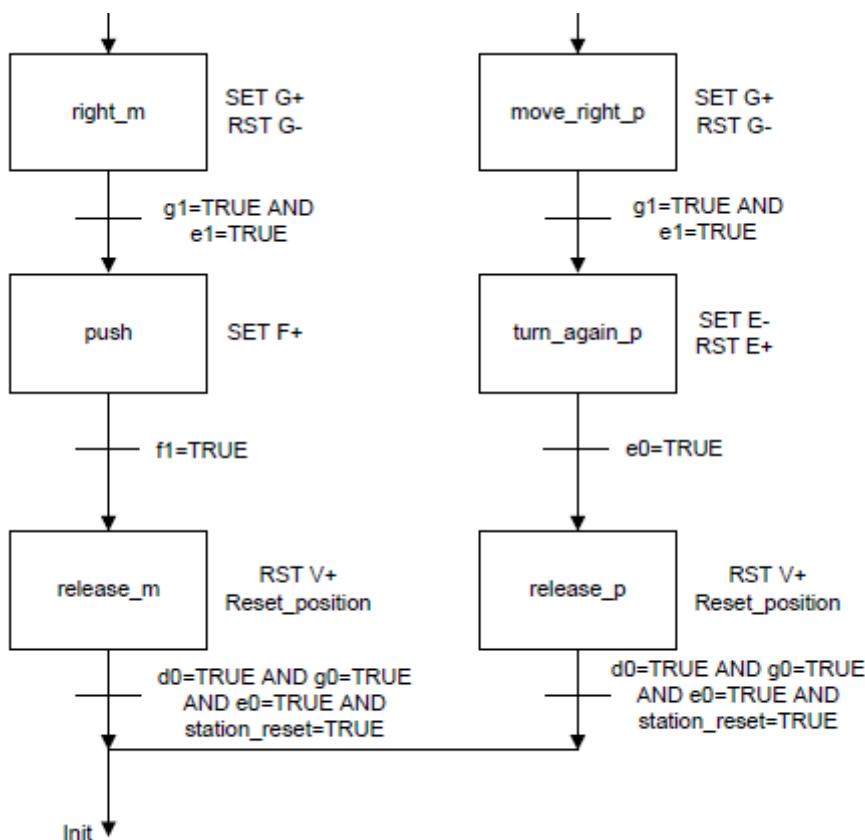
Στην περίπτωση που αφού ο σταθμός σπρώξει ένα κουτί μπροστά από τον αναγνώστη barcode, αυτός δεν διαβάσει τίποτα, ανάβει κόκκινο φως στο φανάρι και παραμένει ακίνητο. Εδώ υπάρχουν δύο εξηγήσεις και συνεπώς δύο εναλλακτικές. Η πρώτη είναι ότι απλά δεν έχει μπει σωστά το κουτί και γι αυτό δεν διαβάστηκε σωστά το barcode οπότε μπορούμε απλά χειροκίνητα να το βάλουμε καλύτερα στην θέση του ούτως ώστε να το διαβάσει ο αναγνώστης και να συνεχίσει το πρόγραμμα κανονικά την λειτουργία του. Η δεύτερη είναι ότι δεν υπάρχει καθόλου κουτί ή είναι κάποιο αντικείμενο που έχει μπει κατά λάθος. Τότε πατώντας το κουμπί reset ο σταθμός προχωράει στην επόμενη στήλη για να τροφοδοτήσει τον σταθμό με το επόμενο κουτί.



Εικόνα 30 - 1<sup>ος</sup> σταθμός (3/4)

Κάθε φορά που εντοπίζεται ένα λάθος κουτί αυξάνεται η μεταβλητή “count1.CV” του συστήματος. Όταν ο σταθμός εντοπίσει το πέμπτο λάθος κουτί τότε προχωράει στην διαδικασία όπου πετάει το κουτί στον κάδο με τα άχρηστα κουτιά αλλά μετά δεν συνεχίζει αυτόματα στην επόμενη στήλη. Αντί αυτού σταματάει και περιμένει αναβοσβήνοντας το κόκκινο φως στο φανάρι. Ο λόγος είναι για να κάνει γνωστό στον χρήστη ότι έχουν υπάρξει ήδη πέντε λάθος κουτιά στις στήλες. Ο

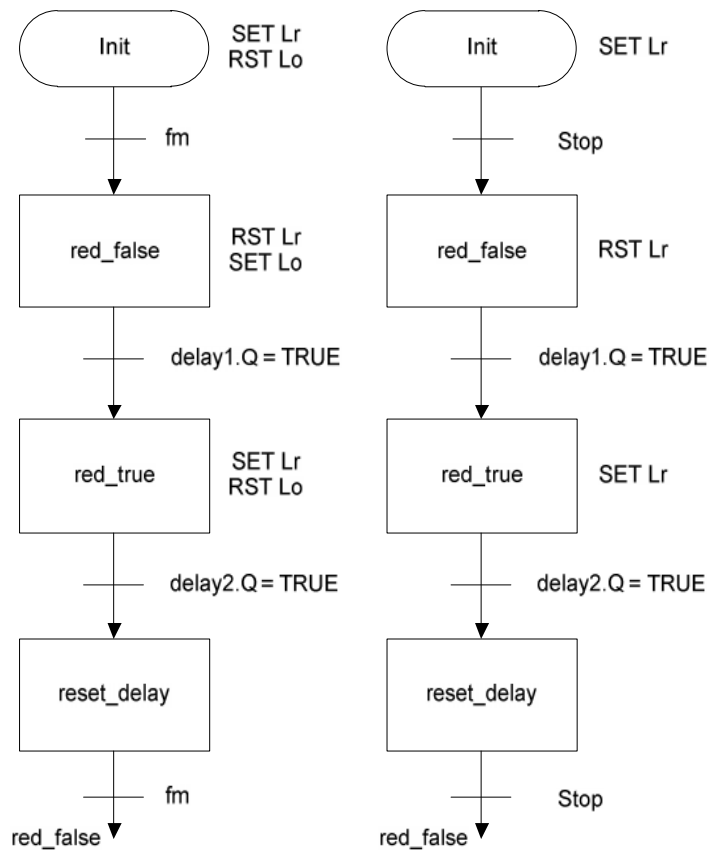
χρήστης μπορεί να πατήσει ανά πάσα στιγμή το reset και ο σταθμός θα μηδενίσει τον μετρητή και θα συνεχίσει τροφοδοτώντας ένα κουτί από την πρώτη στήλη.



Εικόνα 31 - 1<sup>ος</sup> σταθμός (4/4)

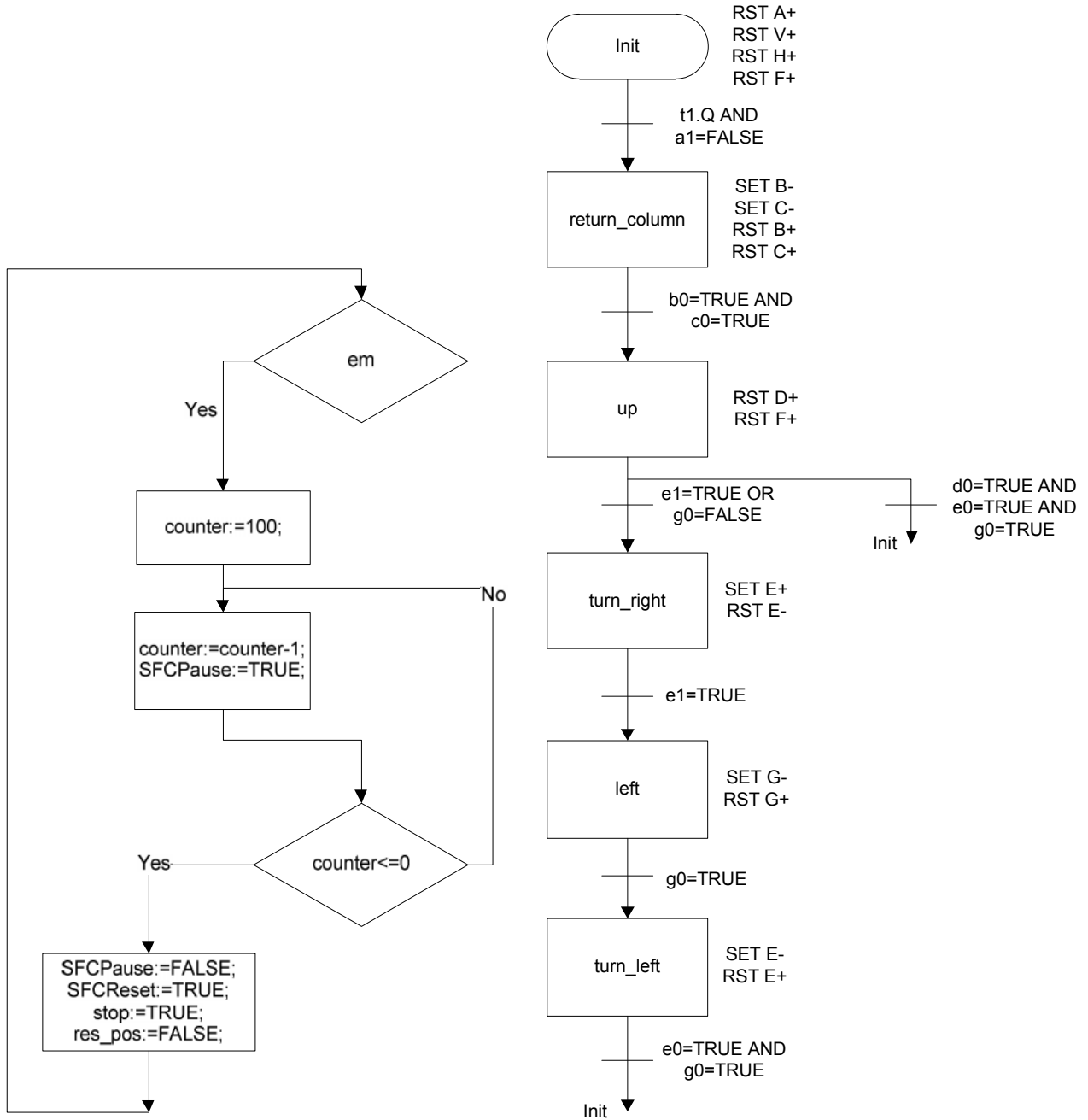
Όταν έχει μείνει μόνο μία σειρά από κουτιά στις στήλες, τότε ανάβει το πορτοκαλί φως στο φανάρι στην θέση του πράσινου, σε όλες τις λειτουργίες του σταθμού. Όταν και το τελευταίο κουτί φύγει από τις στήλες, γίνεται κανονικά η διαδικασία που προβλέπεται για το συγκεκριμένο κουτί (μεταφορά στον ιμάντα ή απόρριψη στον κάδο). Στην συνέχεια όμως, αναβοσβήνουν εναλλάξ το πορτοκαλί με το κόκκινο φως στο φανάρι και τα κουμπιά δεν ανταποκρίνονται σε καμία ενέργεια του χρήστη μέχρι να μπει τουλάχιστον ένα κουτί στις στήλες.

Επίσης όταν μαζευτούν πολλά κουτιά στον ιμάντα μεταφοράς και σχηματίσουν ουρά στον επόμενο σταθμό, ενεργοποιείται ένα αισθητήριο που βρίσκεται στο τέλος του ιμάντα μεταφοράς. Με αυτό το αισθητήριο το PLC αναγνωρίζει αυτή την ουρά και σταματά την διαδικασία μέχρι να μετακινηθούν κάποια κουτιά, να μειωθεί η ουρά και ως συνέπεια να απενεργοποιηθεί το αισθητήριο.



Εικόνα 32 – διαδικασία για άδεια κουτιά (αριστερά) – διαδικασία για κόκκινο φως (δεξιά)

Η τελευταία διαδικασία που υπάρχει στον σταθμό ενεργοποιείται με το κουμπί emergency stop. Πατώντας αυτό το κουμπί ο σταθμός σταματά άμεσα οποιαδήποτε διαδικασία και όλα τα μέρη του μένουν ακίνητα. Μας δίνεται έτσι η δυνατότητα να κόψουμε την τροφοδοσία του αέρα στο πνευματικό μας σύστημα και να ξεμπλοκάρουμε μία πιθανή εμπλοκή του σταθμού.



Εικόνα 33 – Στοπ ασφαλείας (αριστερά) – Επιστροφή σε αρχική θέση (δεξιά)

Αφού επαναφέρουμε το κουμπί emergency stop στην κανονική του θέση και επαναφέρουμε επίσης την τροφοδοσία του αέρα μπορούμε να πατήσουμε ταυτόχρονα τα κουμπιά stop και reset για να γίνει reset ο σταθμός και να επανέλθει σε κατάσταση αναμονής. Στην περίπτωση που είναι ενεργοποιημένη η λειτουργία του emergency stop αναβοσβήνει το κόκκινο φως στο φανάρι.



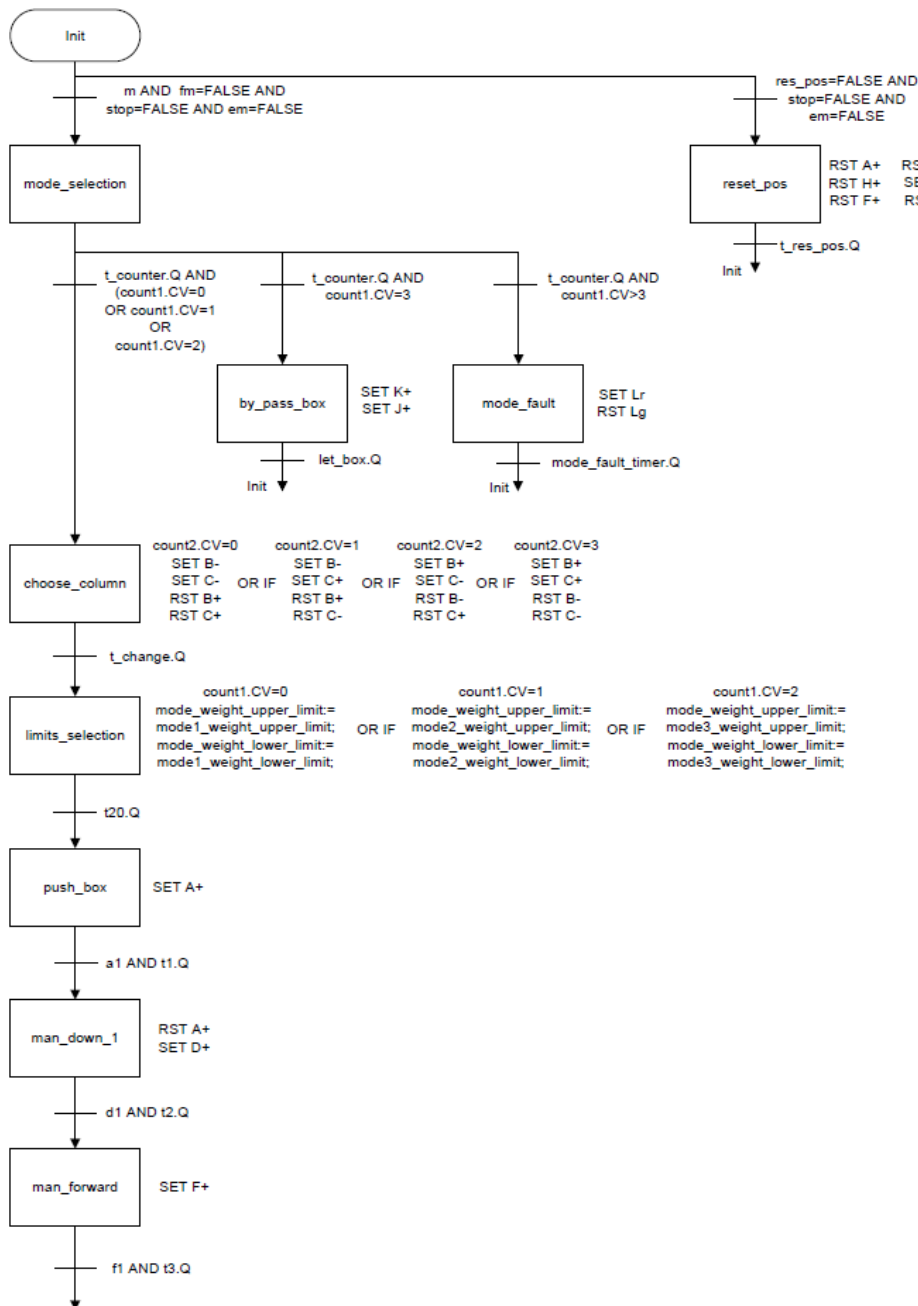
Στην συνέχεια ακολουθεί η λίστα των ενεργειών των inputs και outputs του σταθμού καθώς και τα σύμβολα που αντιστοιχούν σε κάθε αισθητήριο, πνευματικό κύλινδρο ή βαλβίδα. Στην τελευταία στήλη βλέπουμε τον τύπο αυτών των συσκευών.

	ACTION	SIMBOL	TYPE
INPUTS	Start push button	m	N.O.
	Stop push button	p	N.O.
	Integrated selector	int	Comnutator
	Reset push button	r	N.O.
	Bottle feeder forward	a1	Reed magnetic switch
	Positioner_1 cylinder backward	b0	Reed magnetic switch
	Positioner_1 cylinder forward	b1	Reed magnetic switch
	Positioner_2 cylinder backward	c0	Reed magnetic switch
	Positioner_2 cylinder forward	c1	Reed magnetic switch
	Vertical manipulator upward	d0	Reed magnetic switch
	Vertical manipulator downward	d1	Reed magnetic switch
	Rotary actuator backward	e0	Reed magnetic switch
	Rotary actuator forward	e1	Reed magnetic switch
	Horizontal manipulator backward	f0	Reed magnetic switch
	Horizontal manipulator forward	f1	Reed magnetic switch
	Moving manipulator backward	g0	Reed magnetic switch
	Moving manipulator forward	g1	Reed magnetic switch
	Minimun bottles detector	dmin	Photo-electric sensor PNP
	No bottles detector	fm	Photo-electric sensor PNP
	Maximun queue detector	dc	Proximity switch PNP
	Vacuum	v	Pressure sensor PNP
Emergency stop	em	N.O.	
OUTPUTS	Bottle Feeder cylinder forwards	A+	Single solenoid valve
	Positioner 1 cylinder forwards	B+	Double solenoid valve
	Positioner_1 cylinder backwards	B-	
	Positioner_2 cylinder forwards	C+	Double solenoid valve
	Positioner_2 cylinder backwards	C-	
	Vertical manipulator downwards	D+	Single solenoid valve
	Rotary actuator forwards	E+	Double solenoid valve
	Rotary actuator backwards	E-	
	Horizontal manipulator forwards	F+	Single solenoid valve
	Moving manipulator forwards	G+	Double solenoid valve
	Moving manipulator backwards	G-	
	Bottle stopper cylinder_1 forwards	H+	Single solenoid valve
	Vacuum	V+	Single solenoid valve
	Green light	Lg	Neon light
	Orange light	Lo	Neon light
Red light	Lr	Neon light	

### 4.1.2 2ος σταθμός

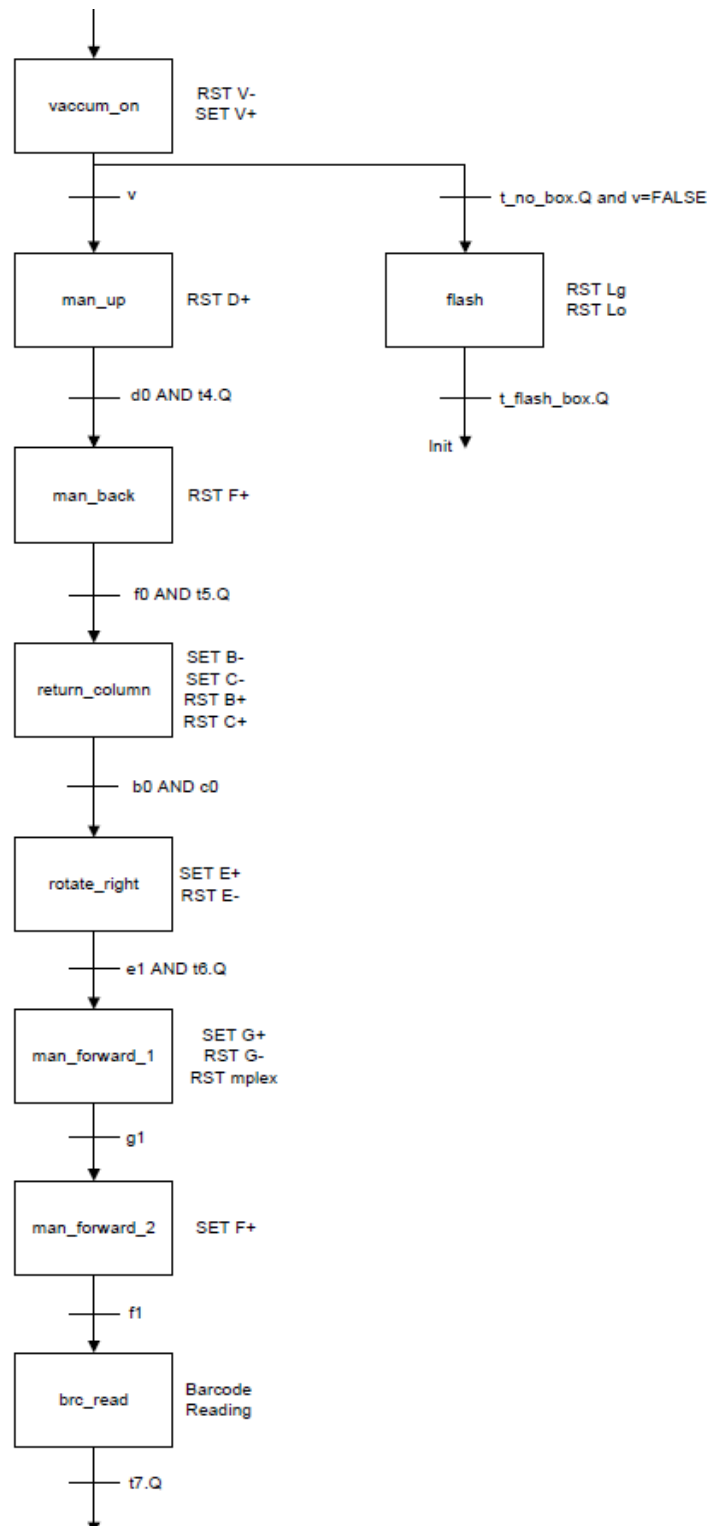
Ο δεύτερος σταθμός είναι υπεύθυνος για να γεμίζει τα κουτιά που έχουν κόκκινο barcode με συγκεκριμένη ποσότητα υλικού.

Το κουμπί start έχει 4 διαφορετικές λειτουργίες. Πατώντας το τέσσερις φορές σε διάστημα δυόμιση δευτερολέπτων επιτρέπει σε οποιοδήποτε κουτί έρχεται από τον προηγούμενο σταθμό να διασχίσει τον ιμάντα μεταφοράς του σταθμού και να πάει στον επόμενο σταθμό. Οι λειτουργίες για ένα, δύο και τρία πατήματα του κουμπιού start αντιστοιχούν στην λειτουργία του σταθμού για να γεμίσει το κουτί με 15, 30 και 45 γραμμάρια αντίστοιχα [Εικόνα 34 - Εικόνα 40].



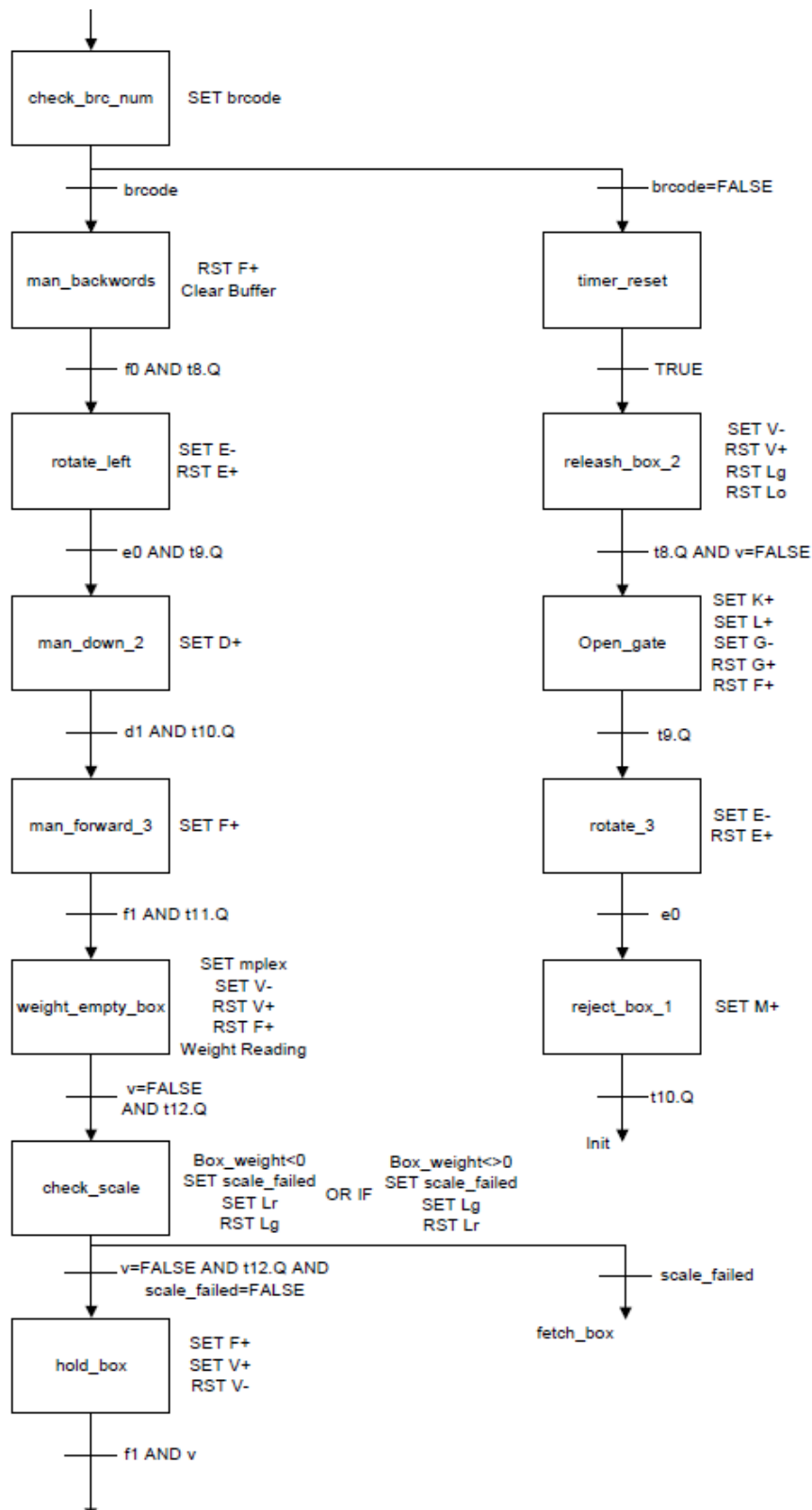
Εικόνα 34 - 2<sup>ος</sup> σταθμός (1/5)

Με το που πατηθεί το κουμπί start και το κουτί βγει από την πρώτη στήλη, οι ηλεκτροπνευματικοί κύλινδροι του σταθμού το μεταφέρουν στον αναγνώστη barcode που βρίσκεται στον ιμάντα μεταφοράς. Εκεί ελέγχεται το barcode του και συγκρίνεται με τους αριθμούς που βρίσκονται σε έναν πίνακα για να αποφασιστεί αν είναι αποδεκτό ή όχι.

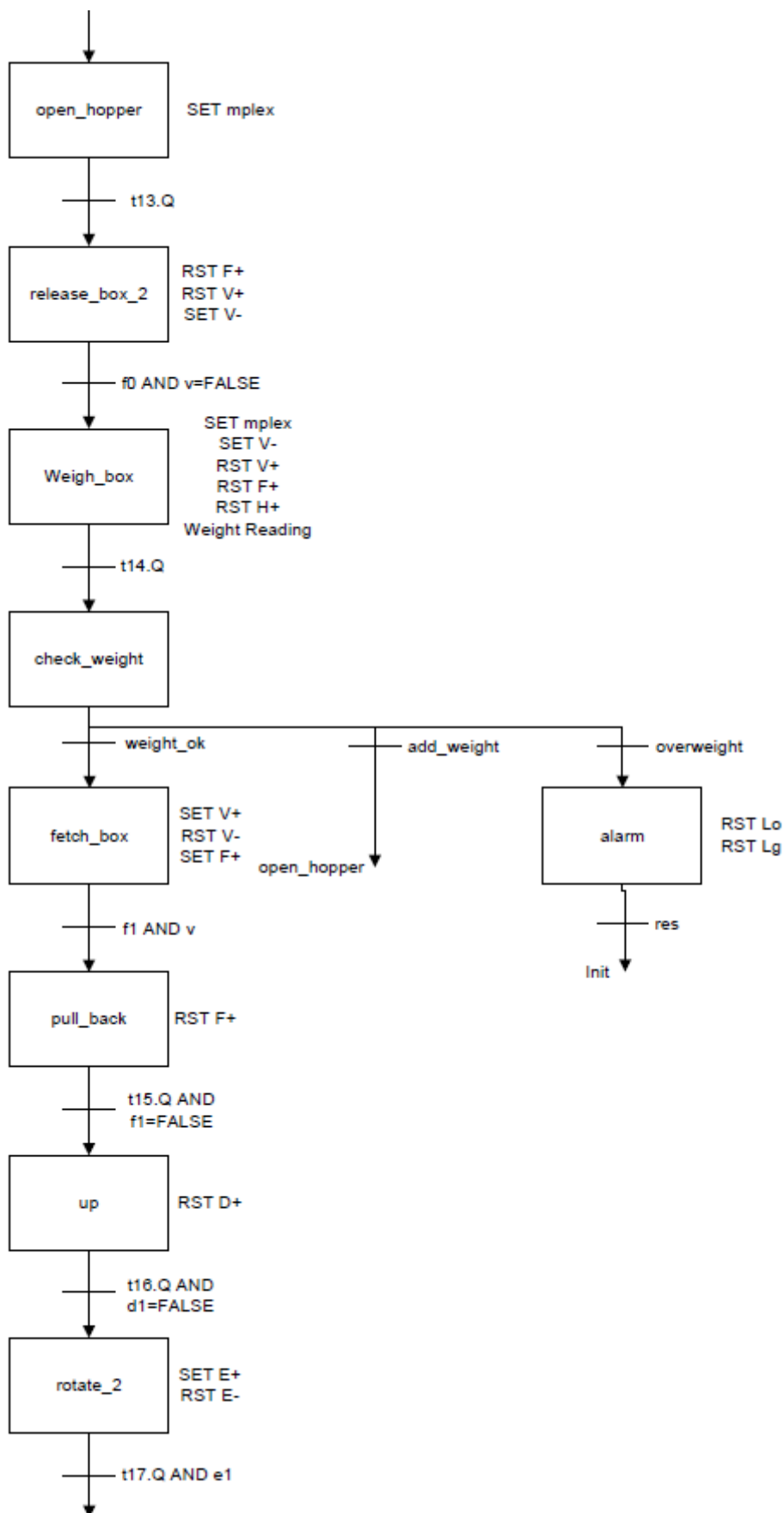


Εικόνα 35 - 2<sup>ος</sup> σταθμός (2/5)

Σε περίπτωση που δεν είναι αποδεκτό τότε το κουτί αφήνεται πάνω στον μάντα και λίγο πριν το τέλος του λειτουργεί ένας μηχανισμός απόρριψης ο οποίος το βγάξει από τον μάντα και το πετάει στο καλάθι των αχρήστων.

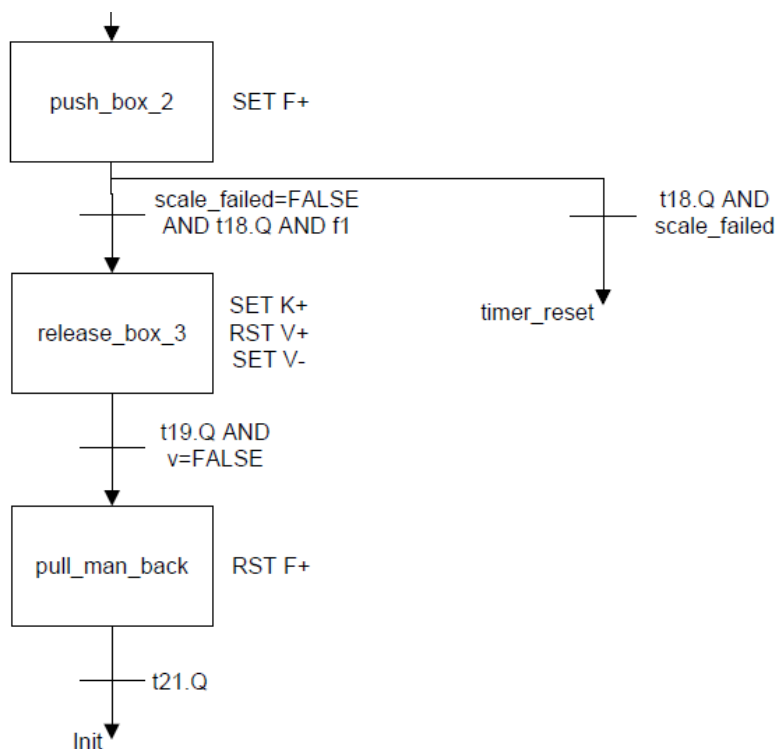


Εικόνα 36 - 2<sup>ος</sup> σταθμός (3/5)



Εικόνα 37 - 2<sup>ος</sup> σταθμός (4/5)

Από την αρχή της διαδικασίας είναι αναμμένο το πράσινο φως στο φανάρι και στην περίπτωση που το κουτί δεν είναι αποδεκτό, σβήνει το πράσινο και αναβοσβήνει το κόκκινο. Στην δεύτερη περίπτωση όπου το κουτί είναι αποδεκτό, ο μηχανισμός του σταθμού το μεταφέρει στην ζυγαριά η οποία βρίσκεται κάτω από το δοχείο που περιέχει το υλικό. Η ζυγαριά αρχικά μετράει το καθαρό βάρος του κουτιού και στην συνέχεια ανοίγει την μεγάλη καταπακτή για να ρίξει μία ποσότητα υλικού. Ξανά μετράει το βάρος του κουτιού με την ποσότητα υλικού που έχει πέσει αυτή την φορά και υπολογίζει το καθαρό βάρος του υλικού που βρίσκεται στο κουτί μέσω αφαίρεσης από την προηγούμενη μέτρηση. Στην συνέχεια αναλόγως ποιο επιθυμητό όριο έχουμε θέσει με τα πατήματα του κουμπιού start και πόσο κοντά βρίσκεται σε αυτό, αποφασίζει το PLC αν θα ανοίξει την μικρή ή μεγάλη καταπακτή και κάθε φορά μετράει την ποσότητα που έχει εκείνη την στιγμή μέσα. Στην περίπτωση που πέσει περισσότερο υλικό από την επιθυμητή ποσότητα, ο μηχανισμός θα πάρει το κουτί, θα το τοποθετήσει στον ιμάντα μεταφοράς και θα τον πετάξει στο κάδο των αχρήστων χρησιμοποιώντας τον μηχανισμό απόρριψης. Στην αντίθετη περίπτωση που το κουτί περιέχει την κατάλληλη ποσότητα υλικού, τότε ο μηχανισμός του σταθμού θα το πάρει και θα το μεταφέρει στον ιμάντα μεταφοράς για να συνεχίσει την πορεία του στον επόμενο σταθμό.

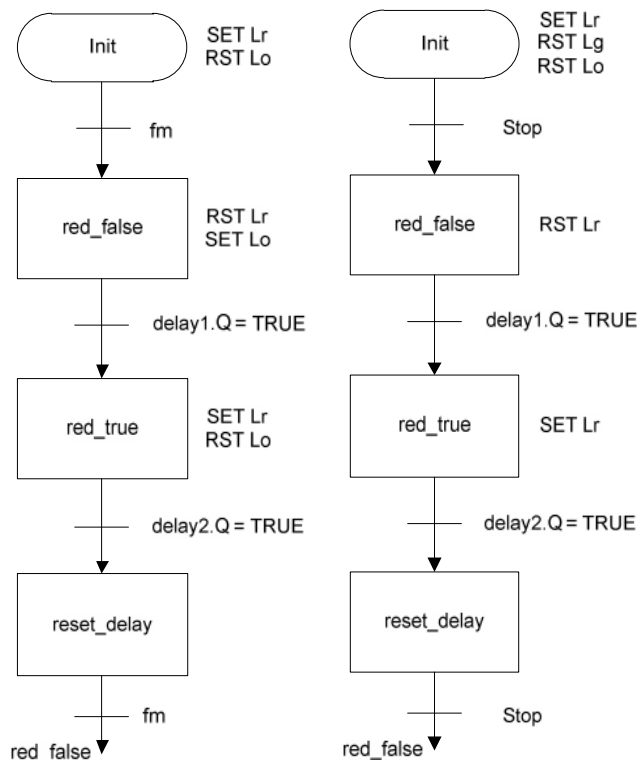


Εικόνα 38 - 2<sup>ος</sup> σταθμός (5/5)

Κάθε φορά που εντοπίζεται ένα λάθος κουτί αυξάνεται μία μεταβλητή του συστήματος. Όταν ο σταθμός εντοπίσει το πέμπτο λάθος κουτί τότε προχωράει κανονικά στην διαδικασία απόρριψης όπου πετάει το κουτί στον κάδο με τα άχρηστα κουτιά. Στην περίπτωση αυτή όμως δεν συνεχίζει αυτόματα στην επόμενη στήλη αλλά σταματάει και περιμένει αναβοσβήνοντας το κόκκινο φως στο φανάρι. Αυτό συμβαίνει για να κάνει γνωστό στον χρήστη ότι έχουν υπάρξει ήδη πέντε λάθος κουτιά στις στήλες. Ο χρήστης μπορεί να πατήσει ανά πάσα στιγμή το reset

και ο σταθμός θα μηδενίσει τον μετρητή και θα συνεχίσει τροφοδοτώντας ένα κουτί από την πρώτη στήλη.

Όταν έχει μείνει μόνο μία σειρά από κουτιά στις στήλες, τότε ανάβει το πορτοκαλί φως στο φανάρι στην θέση του πράσινου, σε όλες τις λειτουργίες του σταθμού. Όταν και το τελευταίο κουτί φύγει από τις στήλες, γίνεται κανονικά η διαδικασία που προβλέπεται για το συγκεκριμένο κουτί (μεταφορά στον ιμάντα ή απόρριψη στον κάδο). Στην συνέχεια όμως, αναβοσβήνουν εναλλάξ το πορτοκαλί με το κόκκινο φως στο φανάρι και τα κουμπιά τίθενται εκτός λειτουργίας μέχρι να μπει τουλάχιστον ένα κουτί στις στήλες.



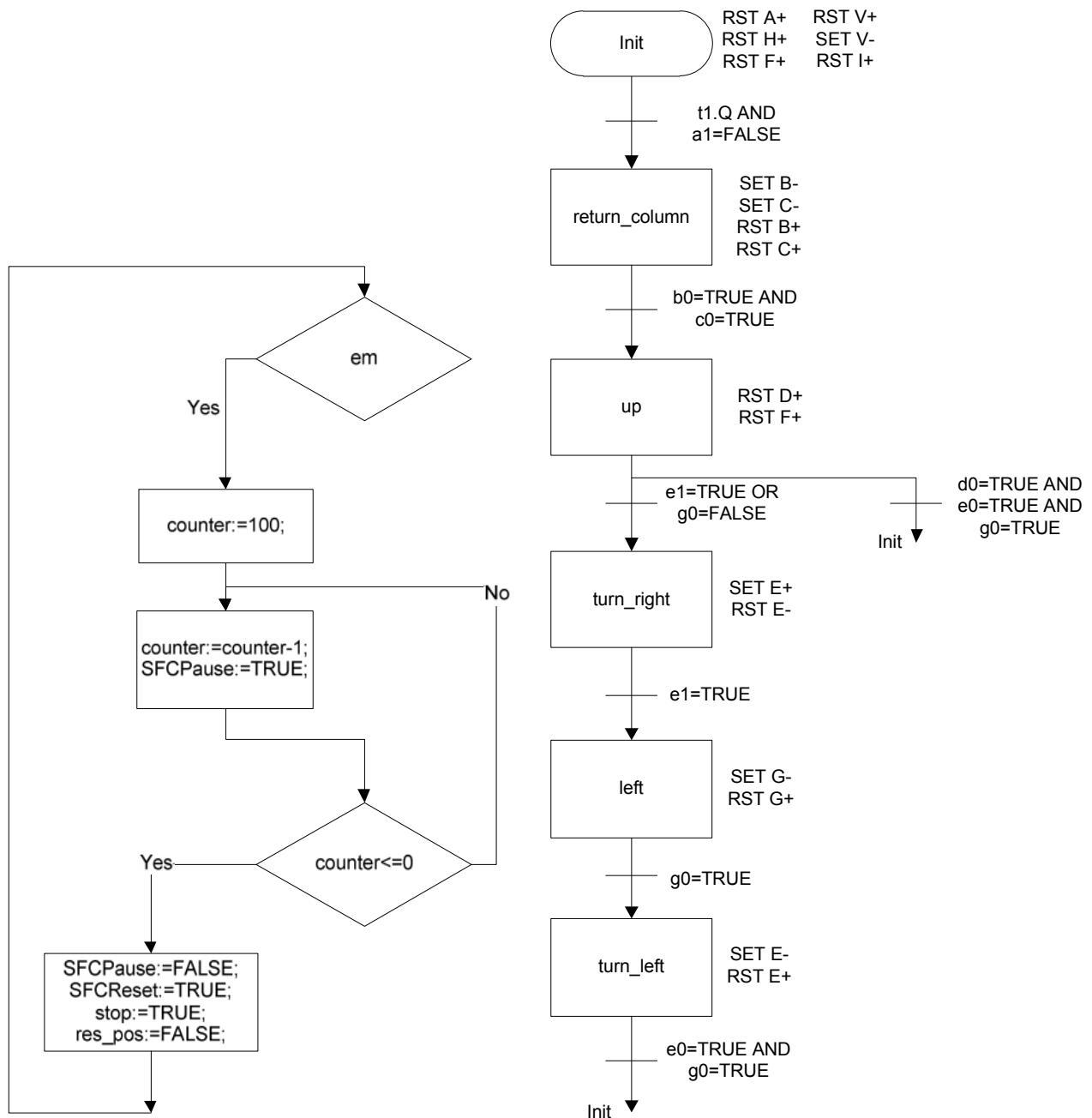
Εικόνα 39 - διαδικασία για άδεια κουτιά (αριστερά) – διαδικασία για κόκκινο φως (δεξιά)

Επίσης όταν μαζευτούν πολλά κουτιά στον ιμάντα μεταφοράς και σχηματίσουν ουρά στον επόμενο σταθμό, ενεργοποιείται ένα αισθητήριο που βρίσκεται στο τέλος του ιμάντα μεταφοράς. Με αυτό το αισθητήριο το PLC αναγνωρίζει αυτή την ουρά και σταματά την διαδικασία μέχρι να μετακινηθούν κάποια κουτιά, να μειωθεί η ουρά και ως συνέπεια να απενεργοποιηθεί το αισθητήριο.

Η τελευταία διαδικασία που υπάρχει στον σταθμό ενεργοποιείται με το κουμπί emergency stop. Πατώντας αυτό το κουμπί ο σταθμός σταματά άμεσα οποιαδήποτε διαδικασία και όλα τα μέρη του μένουν ακίνητα. Μας δίνεται έτσι η δυνατότητα να κόψουμε την τροφοδοσία του αέρα στο πνευματικό μας σύστημα και να ξεμπλοκάρουμε μία πιθανή εμπλοκή του σταθμού.

Αφού επαναφέρουμε το κουμπί emergency stop στην κανονική του θέση και επαναφέρουμε επίσης την τροφοδοσία του αέρα μπορούμε να πατήσουμε

ταυτόχρονα τα κουμπιά stop και reset για να γίνει reset ο σταθμός και επανέλθει σε κατάσταση αναμονής. Στην περίπτωση που είναι ενεργοποιημένη η λειτουργία του emergency stop αναβοσβήνει το κόκκινο φως στο φανάρι.



Εικόνα 40 - Στοπ ασφαλείας (αριστερά) – Επιστροφή σε αρχική θέση (δεξιά)

Στην συνέχεια ακολουθεί η λίστα των ενεργειών των inputs και outputs του σταθμού καθώς και τα σύμβολα που αντιστοιχούν σε κάθε αισθητήριο, πνευματικό

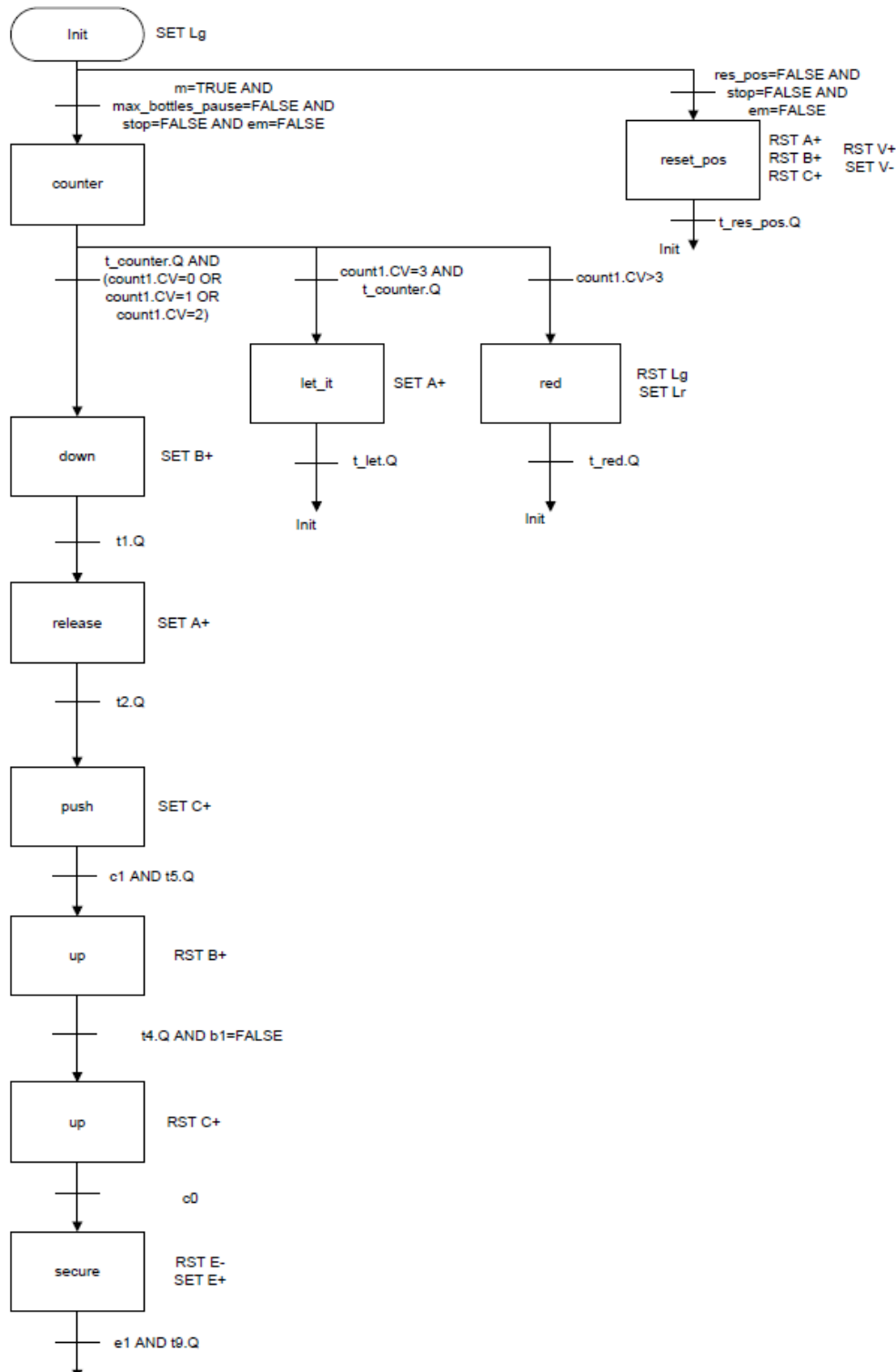


κύλινδρο ή βαλβίδα. Στην τελευταία στήλη βλέπουμε τον τύπο αυτών των συσκευών.

	ACTION	SIMBOL	TYPE
INPUTS	Start push button	m	N.O.
	Stop push button	p	N.O.
	Integrated selector	int	Comnutator
	Reset push button	r	N.O.
	Bottle feeder forward	a1	Reed magnetic switch
	Positioner_1 cylinder backward	b0	Reed magnetic switch
	Positioner_1 cylinder forward	b1	Reed magnetic switch
	Positioner_2 cylinder backward	c0	Reed magnetic switch
	Positioner_2 cylinder forward	c1	Reed magnetic switch
	Vertical manipulator upward	d0	Reed magnetic switch
	Vertical manipulator downward	d1	Reed magnetic switch
	Rotary actuator backward	e0	Reed magnetic switch
	Rotary actuator forward	e1	Reed magnetic switch
	Horizontal manipulator backward	f0	Reed magnetic switch
	Horizontal manipulator forward	f1	Reed magnetic switch
	Moving manipulator backward	g0	Reed magnetic switch
	Moving manipulator forward	g1	Reed magnetic switch
	Bottle stopper elevator upward	k1	Reed magnetic switch
	Rejection cylinder elevator downward	l1	Reed magnetic switch
	Minimun bottles detector	dmin1	Photo-electric sensor PNP
	No bottles detector	fm	Photo-electric sensor PNP
	Minimun material	dmin2	Photo-electric reflex sensor PNP
	Bottle in line	dc1	Proximity switch PNP
	Maximun queue detector	dc2	Proximity switch PNP
	Vacuum in cups	v	Pressure sensor PNP
	Emergency Stop	em	N.O.
OUTPUTS	Bottle Feeder cylinder forwards	A+	Single solenoid valve
	Positioner_1 cylinder forwards	B+	Double solenoid valve
	Positioner_1 cylinder backwards	B-	
	Positioner_2 cylinder forwards	C+	Double solenoid valve
	Positioner_2 cylinder backwards	C-	
	Vertical manipulator downwards	D+	Single solenoid valve
	Rotary actuator forwards	E+	Double solenoid valve
	Rotary actuator backwards	E-	
	Horizontal manipulator forwards	F+	Single solenoid valve
	Moving manipulator forwards	G+	Double solenoid valve
	Moving manipulator backwards	G-	
	Feeding great hopper open	H+	Single solenoid valve
	Feeding small hopper open	I+	Single solenoid valve
	Bottle stopper cylinders forwards	J+	Single solenoid valve
	Bottle stopper elevator upwards	K+	Single solenoid valve
	Rejection cylinder elevator downwards	L+	Single solenoid valve
	Rejection cylinder forwards	M+	Single solenoid valve
	Vacuum in cups	V+	Double solenoid valve
	Stop vacuum in cups	V-	
	Bar Code Reader / Weigh scale	mplex	Relay
Green light	Lg	Neon light	
Orange light	Lo	Neon light	
Red light	Lr	Neon light	

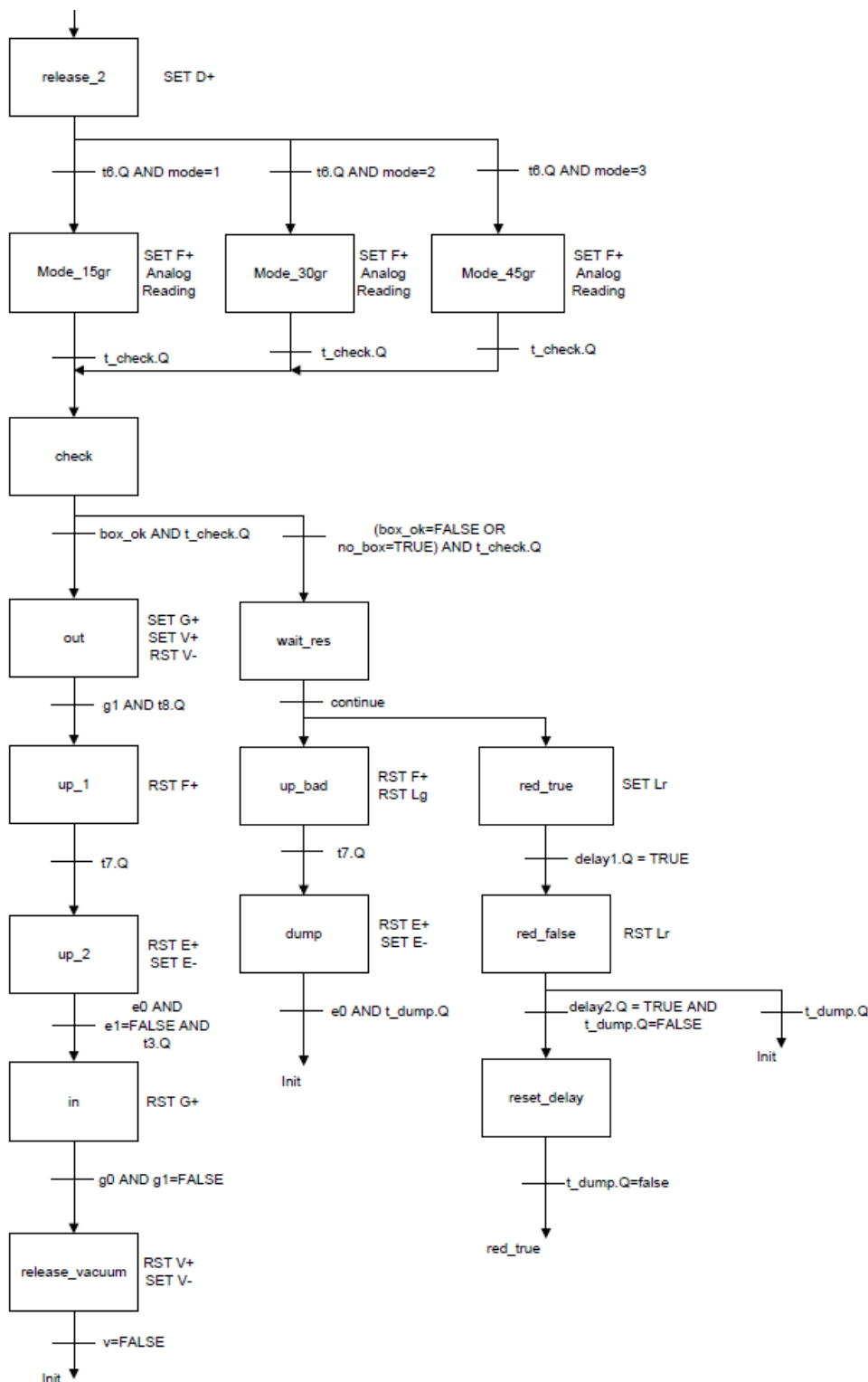
### 4.1.3 3ος σταθμός

Ο τρίτος σταθμός είναι υπεύθυνος για να ελέγχει το ύψος του υλικού που υπάρχει μέσα στα κουτιά. Για τον λόγο αυτό χρησιμοποιείται ένα ηλεκτροπνευματικό έμβολο που δίνει στο PLC σε αναλογική μορφή την τιμή που αντιστοιχεί στο ύψος του περιεχομένου του κουτιού [Εικόνα 41 - Εικόνα 45].



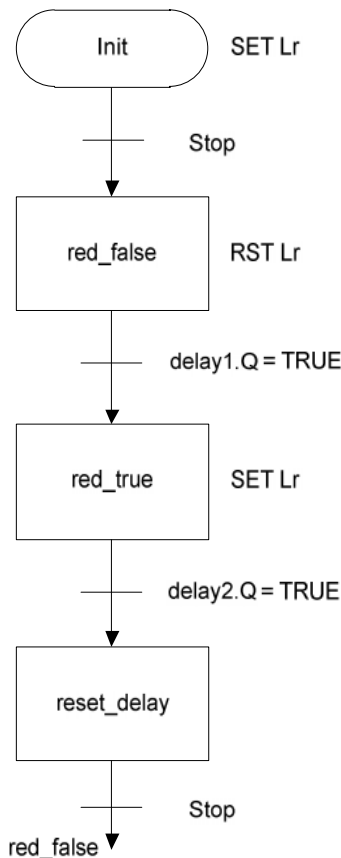
Εικόνα 41 - 3<sup>ος</sup> σταθμός (1/2)

Το κουμπί start έχει και σε αυτόν τον σταθμό τέσσερις διαφορετικές λειτουργίες. Πατώντας το τέσσερις φορές σε διάστημα δυόμιση δευτερολέπτων επιτρέπει στο κουτί που είναι πρώτο ακριβώς πριν από τον αναγνώστη barcode και έρχεται από τον προηγούμενο σταθμό να διασχίσει τον ιμάντα μεταφοράς και να πάει στον επόμενο σταθμό.



Εικόνα 42 - 3<sup>ος</sup> σταθμός (2/2)

Οι λειτουργίες για ένα, δύο και τρία πατήματα αντιστοιχούν στην λειτουργία του σταθμού για να ελέγξει το κουτί αν το ύψος του αντιστοιχεί σε γέμισμα με 15, 30 και 45 γραμμάρια αντίστοιχα. Αν το κουμπί πατηθεί πέντε ή περισσότερες φορές, ανάβει το κόκκινο φως του φαναριού για 1,5 δευτερόλεπτο και μετά περιμένει ξανά να πατηθεί ο σωστός αριθμός επαναλήψεων του κουμπιού start.



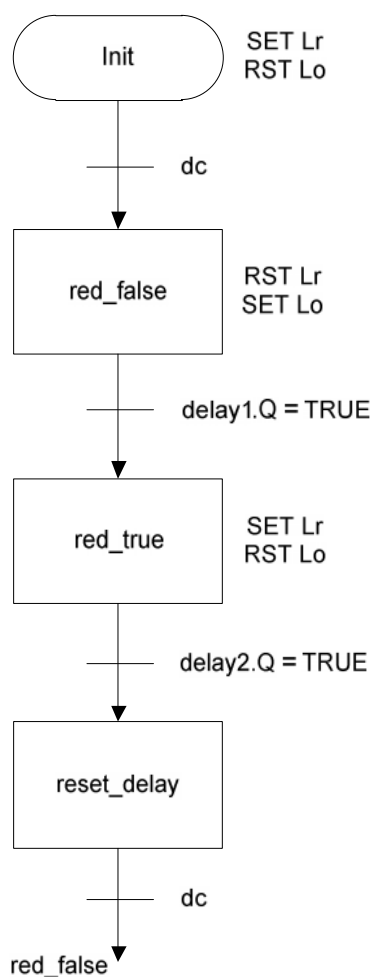
Εικόνα 43 - διαδικασία για κόκκινο φως

Όταν το κουμπί start πατηθεί μία, δύο ή τρεις φορές, τότε ξεκινά η διαδικασία ελέγχου. Αρχικά λειτουργούν δύο διαδοχικές βαλβίδες στην αρχή του ιμάντα μεταφοράς και συγκρατούν το πρώτο κουτί της ουράς μπροστά από τον αναγνώστη barcode. Αφού διαβαστεί το barcode του κουτιού απελευθερώνεται το κουτί και προχωράει λίγο παρακάτω στον ιμάντα μεταφοράς όπου ένα πνευματικό έμβολο θα το σπρώξει στον παράλληλο ιμάντα. Εκεί θα προχωρήσει μέχρι το σημείο που βρίσκονται δύο διαδοχικές βαλβίδες όπου επιτρέπουν να περάσει κάθε φορά μόνο ένα κουτί από την ουρά. Αφού περάσει από αυτές τις βαλβίδες, θα σταματήσει στον μηχανισμό που αποτελείται από ένα κατακόρυφο πνευματικό έμβολο και το οποίο θα ελέγξει το ύψος του προϊόντος στο κουτί. Σε όλη αυτή την διάρκεια είναι αναμμένο το πράσινο φως στο φανάρι. Εάν το κουτί είναι άδειο ή δεν έχει το σωστό ύψος υλικού τότε το κουτί ελευθερώνεται και πέφτει στο τέλος του βοηθητικού ιμάντα μεταφοράς όπου και υπάρχει κάδος αχρήστων. Ταυτόχρονα αναβοσβήνει το κόκκινο φως στο φανάρι για δύο δευτερόλεπτα. Στην αντίθετη περίπτωση που το κουτί έχει το σωστό ύψος υλικού, προεκτείνεται ένα έμβολο το οποίο στην άκρη του έχει μία βεντούζα αέρα η οποία συγκρατεί το κουτί. Το έμβολο επιστρέφει στην θέση του και ταυτόχρονα μεταφέρει το κουτί στον βασικό ιμάντα

μεταφοράς όπου και το απελευθερώνει. Έτσι το κουτί συνεχίζει την πορεία του προς τον επόμενο σταθμό και το σύστημα περιμένει να ξαναπατηθεί το κουμπί start.

Στην περίπτωση που υπάρξουν πέντε προβληματικά κουτιά, το σύστημα αντί να περιμένει να πατηθεί το κουμπί start, αναβοσβήνει το κόκκινο φως του φαναριού προς ενημέρωση του χρήστη και περιμένει να πατηθεί το κουμπί reset προκειμένου να γίνει μηδενισμός του μετρητή που χρησιμοποιείται για να μετράει τα λανθασμένα κουτιά και να μπει το σύστημα ξανά στην αναμονή για το κουμπί start.

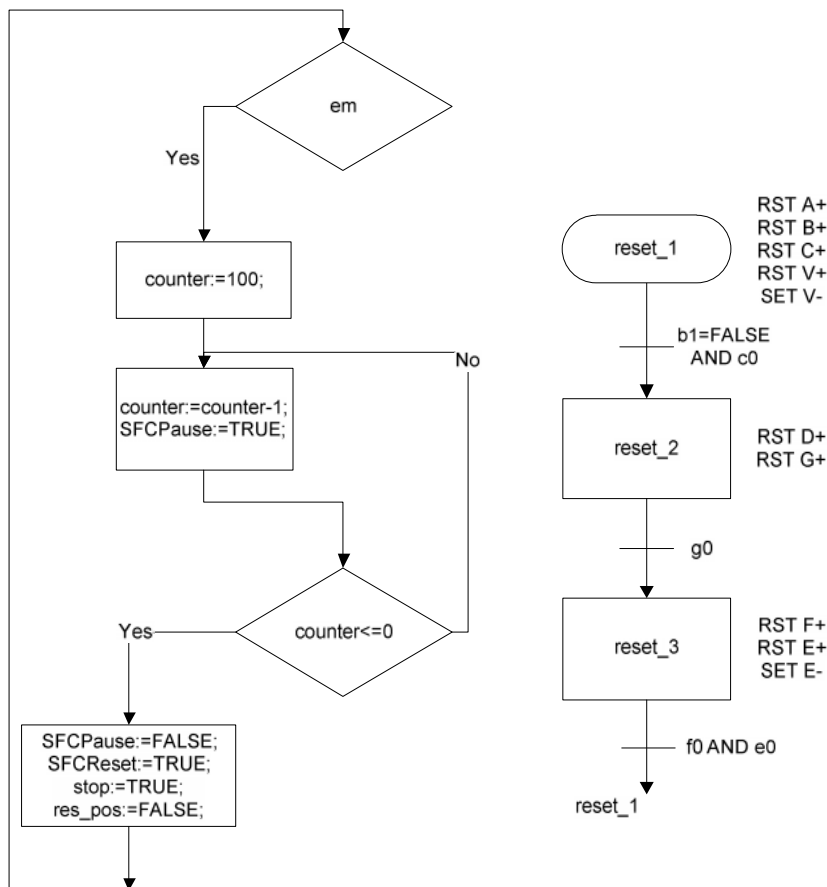
Ακόμα, όταν μαζευτούν πολλά κουτιά στον ιμάντα μεταφοράς και σχηματίσουν ουρά στον επόμενο σταθμό, τότε ενεργοποιείται ένα αισθητήριο που δίνει σήμα στο PLC και σταματά την διαδικασία μέχρι να μετακινηθούν μερικά κουτιά και να μειωθεί η ουρά. Ταυτόχρονα αναβοσβήνουν εναλλάξ τα κόκκινο και πορτοκαλί φώτα στο φανάρι.



Εικόνα 44 - διαδικασία για ουρά στον επόμενο σταθμό

Η τελευταία διαδικασία που υπάρχει στον σταθμό ενεργοποιείται με το κουμπί emergency stop. Πατώντας αυτό το κουμπί ο σταθμός σταματά άμεσα οποιαδήποτε διαδικασία και όλα τα μέρη του μένουν ακίνητα. Μας δίνεται έτσι η

δυνατότητα να κόψουμε την τροφοδοσία του αέρα στο πνευματικό μας σύστημα και να ξεμπλοκάρουμε μία πιθανή εμπλοκή του σταθμού. Αφού επαναφέρουμε το κουμπί emergency stop στην κανονική του θέση και επαναφέρουμε επίσης την τροφοδοσία του αέρα μπορούμε να πατήσουμε ταυτόχρονα τα κουμπιά stop και reset για να γίνει reset ο σταθμός και επανέλθει σε κατάσταση αναμονής. Στην περίπτωση που είναι ενεργοποιημένη η λειτουργία του emergency stop αναβοσβήνει το κόκκινο φως στο φανάρι.



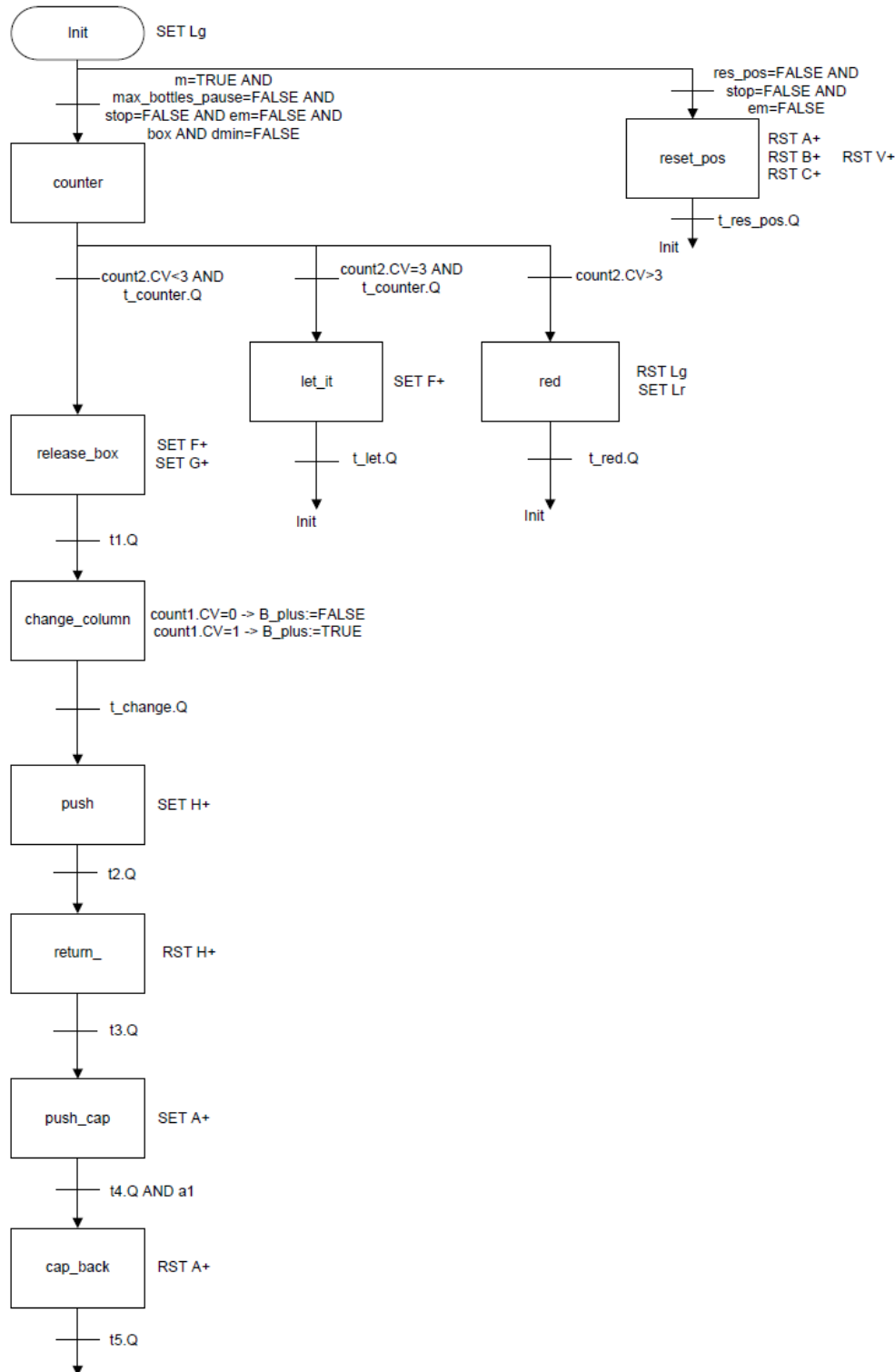
Εικόνα 45 – Στοπ ασφαλείας (αριστερά) – Επιστροφή σε αρχική θέση (δεξιά)

Στην συνέχεια ακολουθεί η λίστα των ενεργειών των inputs και outputs του σταθμού καθώς και τα σύμβολα που αντιστοιχούν σε κάθε αισθητήριο, πνευματικό κύλινδρο ή βαλβίδα. Στην τελευταία στήλη βλέπουμε τον τύπο αυτών των συσκευών.

	ACTION	SIMBOL	TYPE
INPUTS	Start push button	m	N.O.
	Stop push button	p	N.O.
	Integrated selector	int	Comnutator
	Reset push button	r	N.O.
	Line1 2 pusher elevator downward	b1	Reed magnetic switch
	Pusher cylinder line1 2 backward	c0	Reed magnetic switch
	Pusher cylinder line1 2 forward	c1	Reed magnetic switch
	Bottle holder cylinder upward	e0	Reed magnetic switch
	Bottle holder cylinder downward	e1	Reed magnetic switch
	Checker upward	f0	Reed magnetic switch
	Pusher cylinder line2 1 backward	g0	Reed magnetic switch
	Pusher cylinder line2 1 forward	g1	Reed magnetic switch
	Maximun queue detector	dc	Proximity switch PNP
	Vacuum in cups	v	Pressure sensor PNP
	Emergency Stop	em	N.A.
	Analog reading	cA	Linear potentiometer
OUTPUTS	Bottle stoppers line1 forwards	A+	Single solenoid valve
	Line1 2 pusher elevator downwards	B+	Single solenoid valve
	Pusher cylinder line1 2 forwards	C+	Single solenoid valve
	Bottle stoppers line2 forwards	D+	Single solenoid valve
	Bottle holder cylinder downwards	E+	Double solenoid valve
	Bottle holder cylinder upwards	E-	
	Checker downwards	F+	Single solenoid valve
	Pusher cylinder line2 1 forwards	G+	Single solenoid valve
	Vacuum in cups	V+	Double solenoid valve
	Stop vacuum in cups	V-	
	Green light	Lg	Neon light
	Orange light	Lo	Neon light
	Red light	Lr	Neon light

#### 4.1.4 4ος σταθμός

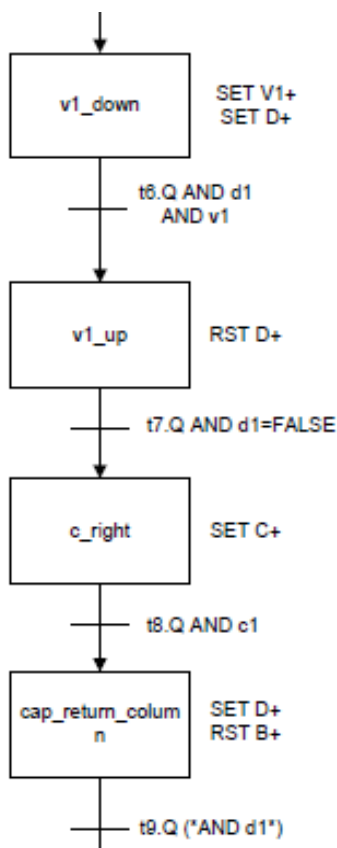
Ο τέταρτος σταθμός είναι υπεύθυνος για την τοποθέτηση καπακιού στα κουτιά που έρχονται στο σταθμό και είναι γεμάτα με υλικό καθώς επίσης και την εκτύπωση ετικέτας με συγκεκριμένα στοιχεία και την επικόλληση της πάνω στο καπάκι [Εικόνα 46 - Εικόνα 52].



Εικόνα 46 - 4<sup>ος</sup> σταθμός (1/4)



Το κουμπί start έχει τρεις διαφορετικές ρυθμίσεις. Όταν πατηθεί 3 φορές, ανοίγουν τα έμβολα που συγκρατούν το πρώτο κουτί και το απελευθερώνουν ούτως ώστε να διασχίσει τον σταθμό και να περάσει στον επόμενο. Όταν το κουμπί πατηθεί περισσότερες από τρεις φορές τότε ανάβει το κόκκινο φως για 2 δευτερόλεπτα και στην συνέχεια ο σταθμός επανέρχεται σε κατάσταση αναμονής και πάλι.



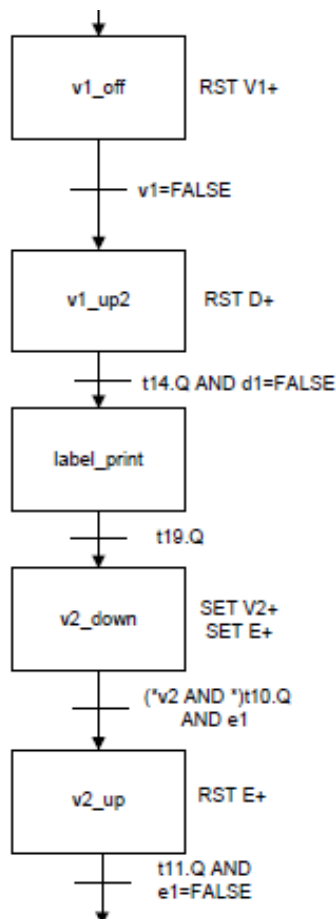
Εικόνα 47 - 4<sup>ος</sup> σταθμός (2/4)

Για να πατηθεί το κουμπί start θα πρέπει ο αναγνώστης barcode να έχει διαβάσει ένα barcode και ως συνέπεια να αναγνωρίσει ότι υπάρχει κουτί μπροστά από την είσοδο του σταθμού. Ο αναγνώστης δηλαδή λειτουργεί και ως αισθητήριο για να αναγνωρίζει την ύπαρξη κουτιού.

Στην κανονική λειτουργία του σταθμού, το κουμπί start πατιέται μία φορά. Το πράσινο φως είναι συνεχώς αναμμένο για να δείξει την σωστή λειτουργία του σταθμού, εκτός αν συμβαίνει κάτι διαφορετικό οπότε και ανάβουν άλλα φώτα. Αφού πατηθεί το start ανοίγουν τα έμβολα που συγκρατούν το κουτί μπροστά από τον αναγνώστη barcode και προχωράει στην συνέχεια του ιμάντα μεταφοράς. Στην μέση περίπου του σταθμού έχει κατέβει ένα έμβολο όπου σταματάει το κουτί και ένα δεύτερο έμβολο το σπρώχνει έξω από τον ιμάντα μεταφοράς και πάνω στην πλατφόρμα όπου θα συνεχιστεί η διαδικασία.

Στην συνέχεια της διαδικασίας, ο τροφοδότης καπακιών παίρνει θέση εναλλάξ στην πρώτη ή δεύτερη στήλη για να τροφοδοτήσει τον σταθμό με καπάκι, μέσω του εμβόλου που σπρώχνει το τελευταίο καπάκι της στήλης στην πλατφόρμα

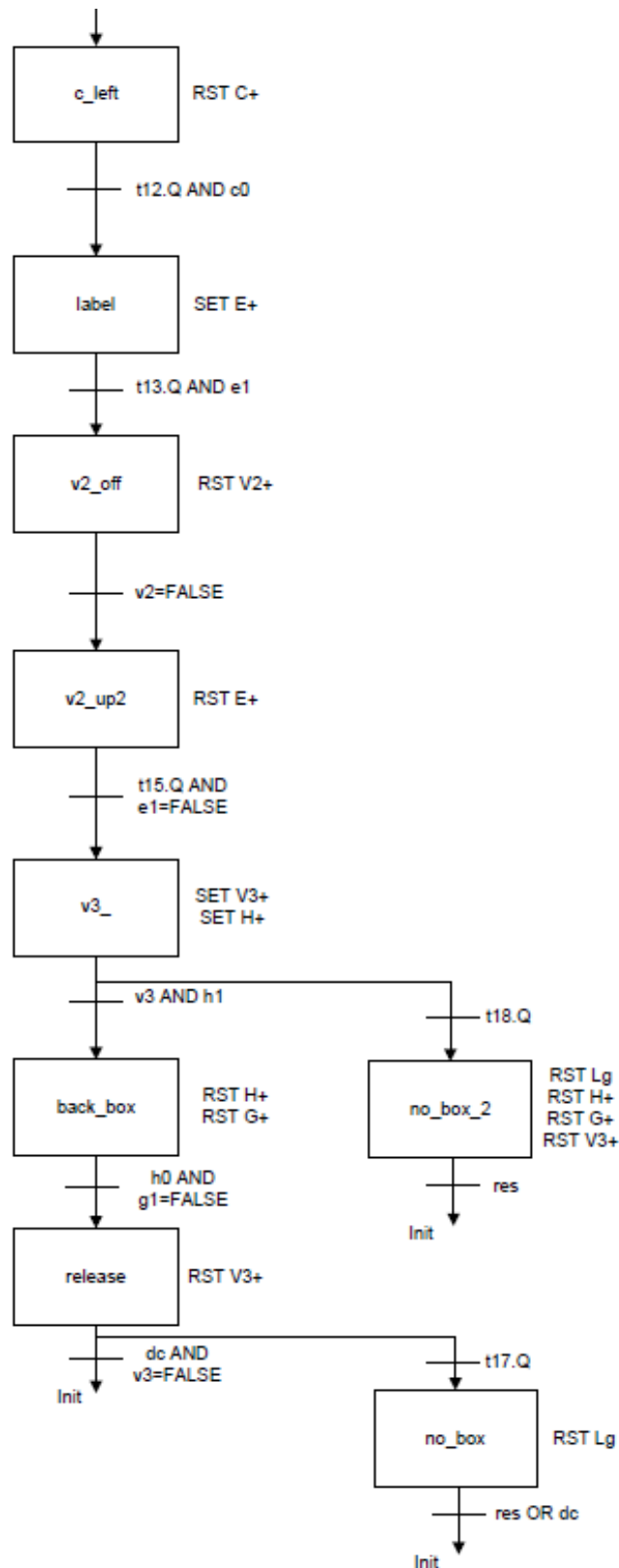
μπροστά του. Από εκεί κινείται κάθετα ένας μηχανισμό που έχει στην άκρη του βεντούζα αέρα, παίρνει το καπάκι και με συνδυασμό οριζόντιων και κάθετων κινήσεων τοποθετεί το καπάκι στο κουτί που βρίσκεται στην διπλανή πλατφόρμα. Αν δεν υπάρχουν διαθέσιμα καπάκια τότε ο σταθμός αναβοσβήνει εναλλάξ το κόκκινο και το πορτοκαλί φως και δεν υπάρχει η δυνατότητα να προχωρήσει η διαδικασία μέχρι να ξανά μπει τουλάχιστον ένα καπάκι. Ο έλεγχος αυτός γίνεται στην αρχή της διαδικασίας πριν καν πατηθεί το κουμπί start.



Εικόνα 48 - 4<sup>ος</sup> σταθμός (3/4)

Από την άλλη μεριά της πλατφόρμας που βρίσκεται το κουτί υπάρχει ο εκτυπωτής των ετικετών. Την ώρα που κουμπώνεται το καπάκι στο κουτί ο εκτυπωτής εκτυπώνει την ετικέτα και με ένα αντίστοιχο μηχανισμό με αυτόν που χρησιμοποιείται για την μεταφορά του καπακιού, μεταφέρεται και επικολλάται και η ετικέτα.

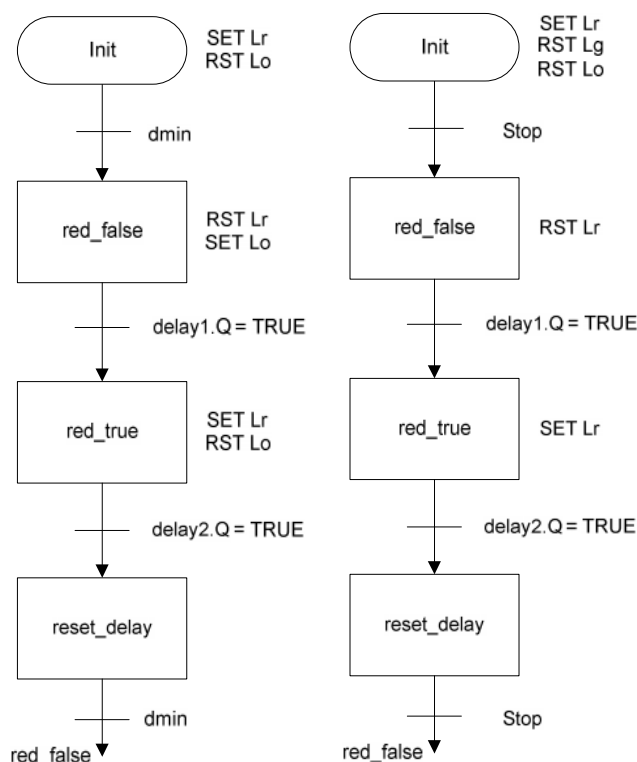
Στην συνέχεια, το έμβολο που έσπρωξε το κουτί από τον ιμάντα μεταφοράς και το τοποθέτησε στην πλατφόρμα, αυτή την φορά προεκτείνεται και χρησιμοποιώντας την βεντούζα αέρα στην άκρη του, επαναφέρει το κουτί στον ιμάντα μεταφοράς. Σε περίπτωση που περάσουν δύομιση δευτερόλεπτα και η βεντούζα αέρα δεν έχει συγκρατήσει κάποιο κουτί αναβοσβήνει το κόκκινο φως και περιμένει να πατηθεί το κουμπί reset για να επανέλθει το πρόγραμμα. Αυτό συμβαίνει για την περίπτωση που για κάποιο λόγο το κουτί δεν υπάρχει πλέον στην πλατφόρμα.



Εικόνα 49 - 4<sup>ος</sup> σταθμός (4/4)

Στην περίπτωση που η βεντούζα αέρα επαναφέρει το κουτί στον μάντα μεταφοράς αλλά αυτό μετά από τρεισήμισι δευτερόλεπτα δεν περάσει από το αισθητήριο που βρίσκεται στο τέλος του μάντα μεταφοράς, αναβοσβήνει και πάλι

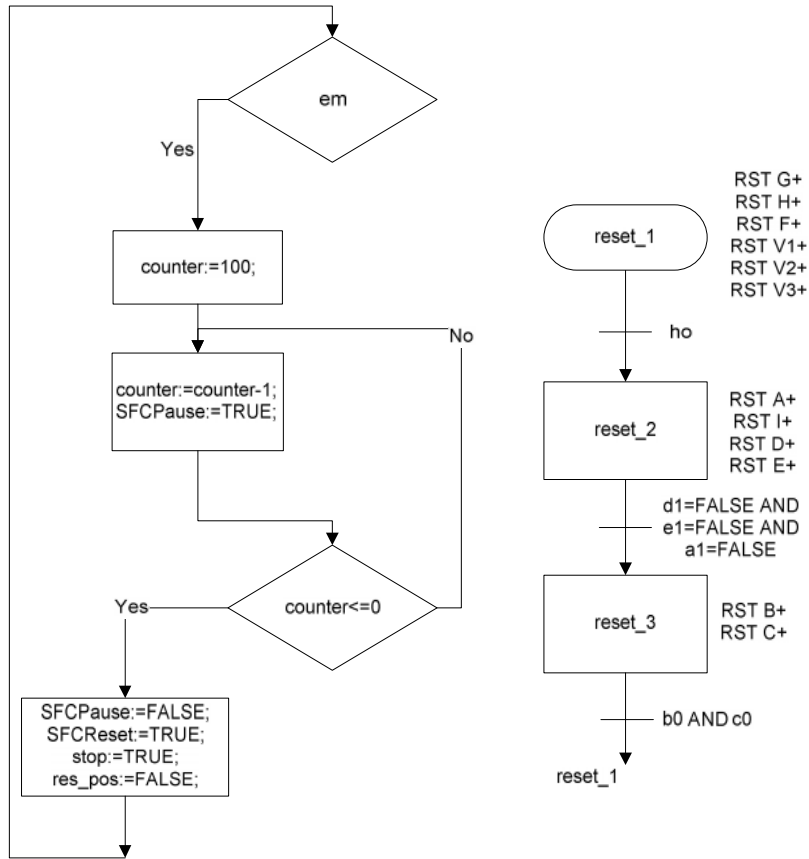
το κόκκινο φως και περιμένει το κουμπί reset για να επανέλθει το πρόγραμμα στην κατάσταση αναμονής. Αυτό συμβαίνει ούτως ώστε να ενημερωθεί ο χρήστης ότι κάτι δεν συμβαίνει σωστά στην παραγωγική διαδικασία.



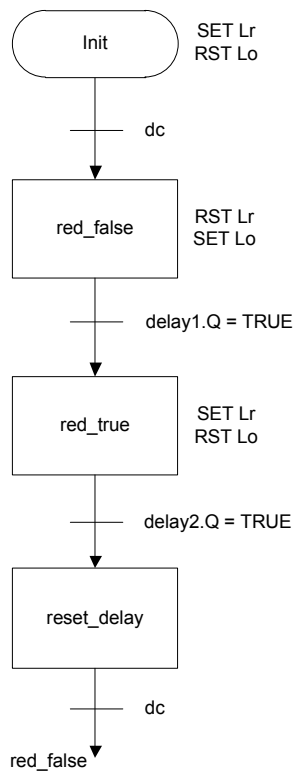
Εικόνα 50 - διαδικασία για άδεια καπάκια (αριστερά) – διαδικασία για κόκκινο φως (δεξιά)

Επίσης όταν μαζευτούν πολλά κουτιά στον ιμάντα μεταφοράς και σχηματίσουν ουρά στον επόμενο σταθμό, ενεργοποιείται ένα αισθητήριο που βρίσκεται στο τέλος του ιμάντα μεταφοράς. Με αυτό το αισθητήριο το PLC αναγνωρίζει αυτή την ουρά και σταματά την διαδικασία μέχρι να μετακινηθούν κάποια κουτιά, να μειωθεί η ουρά και ως συνέπεια να απενεργοποιηθεί το αισθητήριο.

Η τελευταία διαδικασία που υπάρχει στον σταθμό ενεργοποιείται με το κουμπί emergency stop. Πατώντας αυτό το κουμπί ο σταθμός σταματά άμεσα οποιαδήποτε διαδικασία και όλα τα μέρη του μένουν ακίνητα. Μας δίνεται έτσι η δυνατότητα να κόψουμε την τροφοδοσία του αέρα στο πνευματικό μας σύστημα και να ξεμπλοκάρουμε μία πιθανή εμπλοκή του σταθμού. Αφού επαναφέρουμε το κουμπί emergency stop στην κανονική του θέση και επαναφέρουμε επίσης την τροφοδοσία του αέρα μπορούμε να πατήσουμε ταυτόχρονα τα κουμπιά stop και reset για να γίνει reset ο σταθμός και επανέλθει σε κατάσταση αναμονής. Στην περίπτωση που είναι ενεργοποιημένη η λειτουργία του emergency stop αναβοσβήνει το κόκκινο φως στο φανάρι.



Εικόνα 51 – Στοπ ασφαλείας (αριστερά) – Επιστροφή σε αρχική θέση (δεξιά)



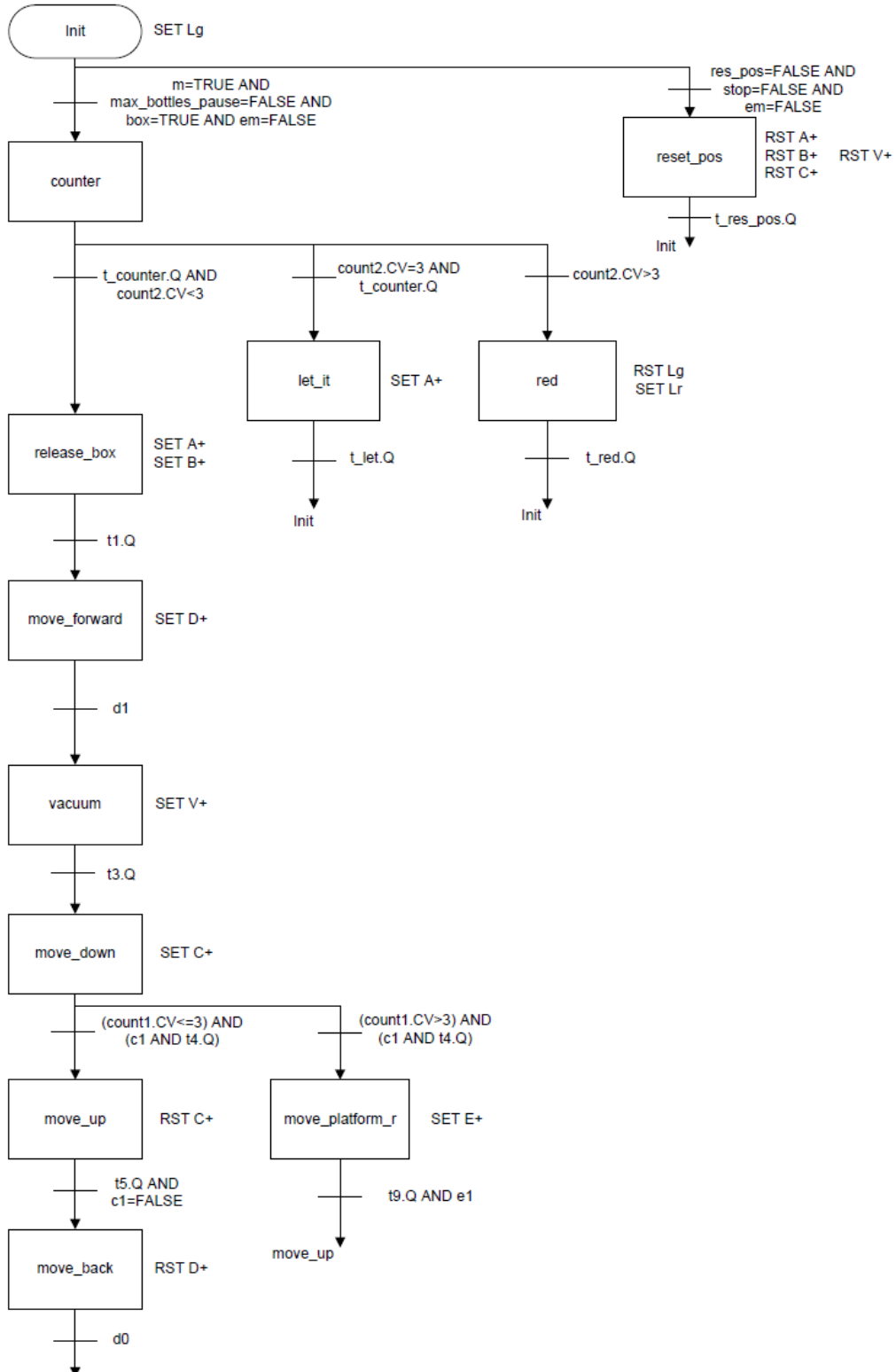
Εικόνα 52 - διαδικασία για ουρά στον επόμενο σταθμό

Στην συνέχεια ακολουθεί η λίστα των ενεργειών των inputs και outputs του σταθμού καθώς και τα σύμβολα που αντιστοιχούν σε κάθε αισθητήριο, πνευματικό κύλινδρο ή βαλβίδα. Στην τελευταία στήλη βλέπουμε τον τύπο αυτών των συσκευών.

	ACTION	SIMBOL	TYPE
INPUTS	Start push button	m	N.O.
	Stop push button	p	N.O.
	Integrated selector	int	Comnutator
	Reset push button	r	N.O.
	Cover feeder forward	a1	Reed magnetic switch
	Positioner cylinder backward	b0	Reed magnetic switch
	Positioner cylinder forward	b1	Reed magnetic switch
	Moving manipulator backward	c0	Reed magnetic switch
	Moving manipulator forward	c1	Reed magnetic switch
	Vertical cover insertion manipulator downward	d1	Reed magnetic switch
	Vertical sticker insertion manipulator downward	e1	Reed magnetic switch
	Line pusher elevator downward	g1	Reed magnetic switch
	Pusher cylinder backward	h0	Reed magnetic switch
	Pusher cylinder forward	h1	Reed magnetic switch
	Vacuum in cover	v1	Pressure sensor PNP
	Vacuum in sticker	v2	Pressure sensor PNP
	Vacuum in bottle	v3	Pressure sensor PNP
	Minimun covers detector	dmin	Photo-electric sensor PNP
Maximun queue detector	dc	Proximity switch PNP	
Emergency Stop	em	N.O.	
OUTPUTS	Cover Feeder cylinder forwards	A+	Single solenoid valve
	Positioner cylinder forwards	B+	Single solenoid valve
	Moving manipulator forwards	C+	Single solenoid valve
	Vertical cover insertion manipulator downwards	D+	Single solenoid valve
	Vertical sticker insertion manipulator downwards	E+	Single solenoid valve
	Bottle stoppers cylinders forwards	F+	Single solenoid valve
	Line pusher elevator downwards	G+	Single solenoid valve
	Pusher cylinder forwards	H+	Single solenoid valve
	Rejection cylinder forwards	I+	Single solenoid valve
	Vacuum in cover	V1+	Single solenoid valve
	Vacuum in sticker	V2+	Single solenoid valve
	Vacuum in bottle	V3+	Single solenoid valve
	Green light	Lg	Neon light
	Orange light	Lo	Neon light
	Red light	Lr	Neon light

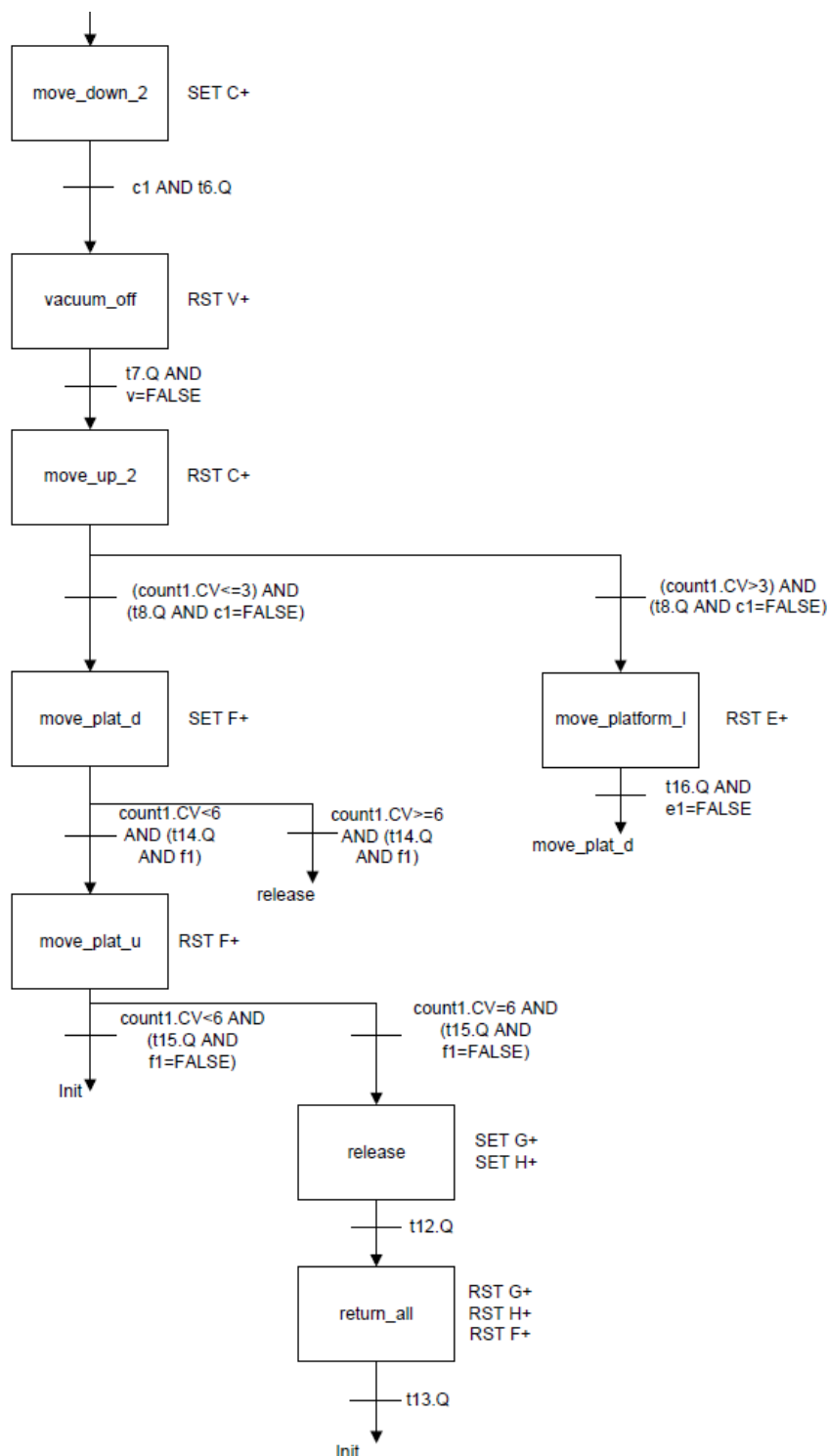
### 4.1.5 5ος σταθμός

Ο πέμπτος σταθμός είναι υπεύθυνος για την παραλαβή των κουτιών που είναι έτοιμα από τους προηγούμενους σταθμούς και να βγάζουν από τον σταθμό για να είναι έτοιμα για αποστολή [Εικόνα 53 - Εικόνα 57].



Εικόνα 53 - 5<sup>ος</sup> σταθμός (1/2)

Το κουμπί start έχει τρεις διαφορετικές ρυθμίσεις. Όταν πατηθεί 3 φορές, ανοίγουν τα έμβολα που συγκρατούν το πρώτο κουτί και το απελευθερώνουν ούτως ώστε να διασχίσει τον σταθμό και να περάσει στον επόμενο. Όταν το κουμπί πατηθεί περισσότερες από τρεις φορές τότε ανάβει το κόκκινο φως για 2 δευτερόλεπτα και στην συνέχεια ο σταθμός επανέρχεται σε κατάσταση αναμονής και πάλι.

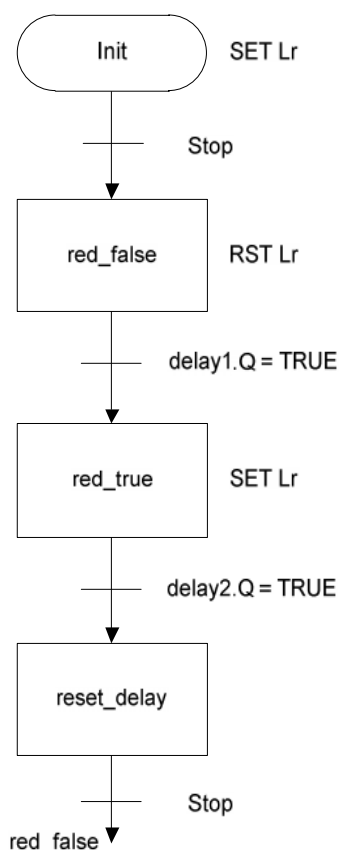


Εικόνα 54 - 5<sup>ος</sup> σταθμός (2/2)



Για να πατηθεί το κουμπί start θα πρέπει ο αναγνώστης barcode να έχει διαβάσει ένα barcode και ως συνέπεια να αναγνωρίσει ότι υπάρχει κουτί μπροστά από την είσοδο του σταθμού. Ο αναγνώστης δηλαδή λειτουργεί και ως αισθητήριο για να αναγνωρίζει την ύπαρξη κουτιού.

Σε όλη την διάρκεια της αναμονής καθώς και της σωστής λειτουργίας του σταθμού, είναι αναμμένο το πράσινο φως στο φανάρι. Σε οποιαδήποτε άλλη περίπτωση ανάβει συνδυασμός ή και ξεχωριστά το κόκκινο και πορτοκαλί φανάρι.

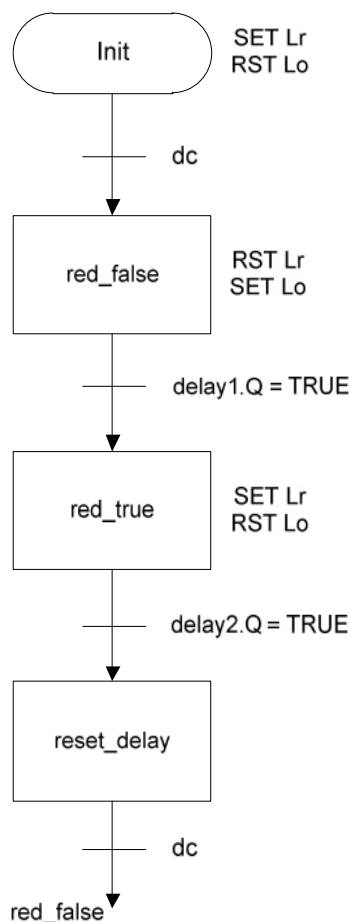


Εικόνα 55 - διαδικασία για κόκκινο φως

Αφού το κουμπί start πατηθεί μία φορά, τα έμβολα που συγκρατούν το κουτί μπροστά από τον αναγνώστη barcode και λειτουργούν σαν πόρτα, απελευθερώνουν το κουτί ούτως ώστε να προχωρήσει στην συνέχεια του ιμάντα μεταφοράς. Λίγο πριν το τέλος του ιμάντα, υπάρχει ένα έμβολο που σταματάει το κουτί μπροστά από την πλατφόρμα αποστολής. Τότε ένας μηχανισμός που αποτελείται από έμβολα που κινούνται οριζόντια και κάθετα κινούνται για να πιάσουν το κουτί και να το τοποθετήσουν πάνω στην πλατφόρμα αποστολής. Η μετακίνηση του κουτιού γίνεται μέσω μίας βεντούζας αέρα που βρίσκεται στο άκρο του ενός εμβόλου.

Από την στιγμή που το κουτί τοποθετηθεί πάνω στην πλατφόρμα, αυτή αποκτά κλίση για να μετακινηθεί το κουτί στην άκρη της πλατφόρμας και μετά επανέρχεται και πάλι σε οριζόντια θέση. Η διαδικασία επαναλαμβάνεται μέχρις ότου μπουν έξι κουτιά στην πρώτη γραμμή της πλατφόρμας και μετά η πλατφόρμα

κινείται κάθε φορά στο πλάι για να γεμίσει και την δεύτερη γραμμή με επίσης έξι κουτιά.



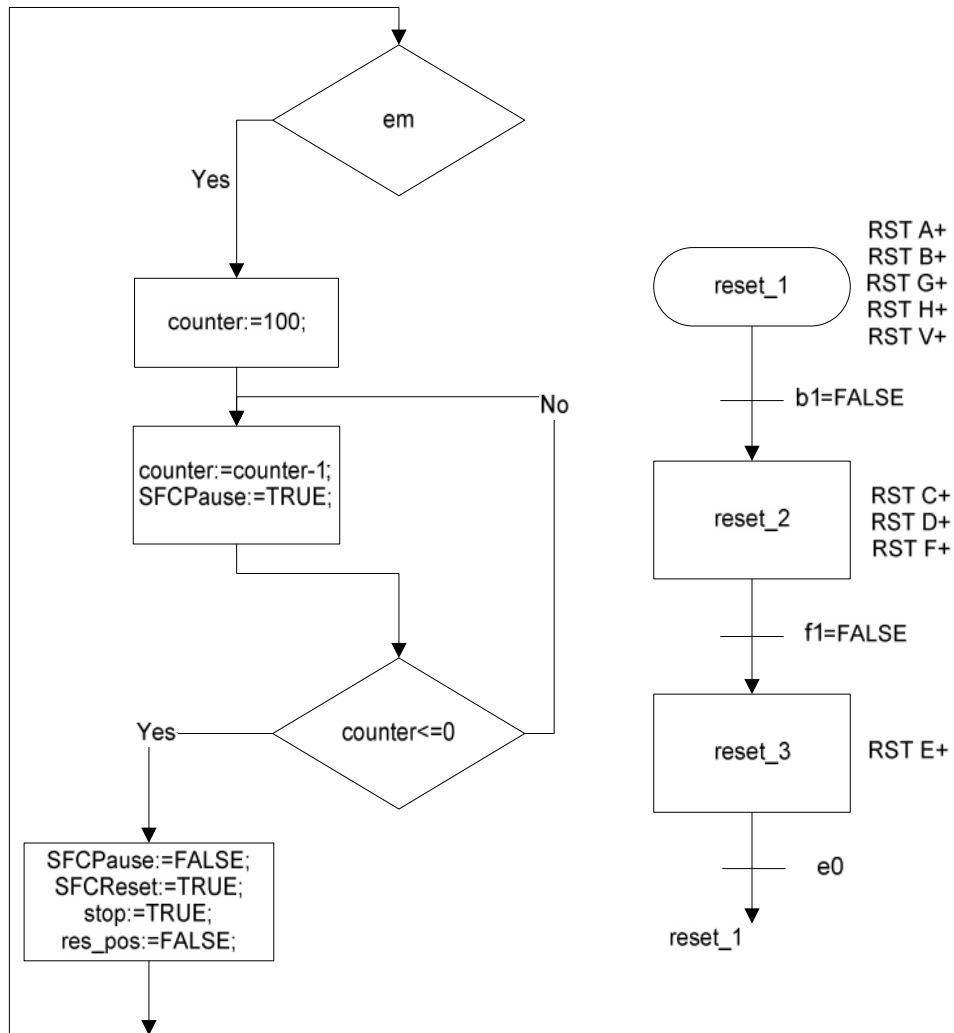
Εικόνα 56 - διαδικασία για ουρά στον επόμενο σταθμό

Όταν η πλατφόρμα αποστολής έχει γεμίσει με δώδεκα κουτιά, ανοίγουν δύο έμβολα που βρίσκονται στην άκρη της και λειτουργούν ως πόρτες για να συγκρατούν τα κουτιά. Με αυτό τον τρόπο τα κουτιά φεύγουν από τον σταθμό και είναι έτοιμα για αποστολή. Μετά από αυτό ο σταθμός είναι ξανά έτοιμος να ετοιμάσει μία καινούργια δωδεκάδα κουτιών για αποστολή.

Επίσης όταν μαζευτούν πολλά κουτιά στον ιμάντα μεταφοράς και σχηματίσουν ουρά στον επόμενο σταθμό, ενεργοποιείται ένα αισθητήριο που βρίσκεται στο τέλος του ιμάντα μεταφοράς. Με αυτό το αισθητήριο το PLC αναγνωρίζει αυτή την ουρά και σταματά την διαδικασία μέχρι να μετακινηθούν κάποια κουτιά, να μειωθεί η ουρά και ως συνέπεια να απενεργοποιηθεί το αισθητήριο.

Η τελευταία διαδικασία που υπάρχει στον σταθμό ενεργοποιείται με το κουμπί emergency stop. Πατώντας αυτό το κουμπί ο σταθμός σταματά άμεσα οποιαδήποτε διαδικασία και όλα τα μέρη του μένουν ακίνητα. Μας δίνεται έτσι η δυνατότητα να κόψουμε την τροφοδοσία του αέρα στο πνευματικό μας σύστημα και να ξεμπλοκάρουμε μία πιθανή εμπλοκή του σταθμού. Αφού επαναφέρουμε το κουμπί emergency stop στην κανονική του θέση και επαναφέρουμε επίσης την

τροφοδοσία του αέρα μπορούμε να πατήσουμε ταυτόχρονα τα κουμπιά stop και reset για να γίνει reset ο σταθμός και επανέλθει σε κατάσταση αναμονής. Στην περίπτωση που είναι ενεργοποιημένη η λειτουργία του emergency stop αναβοσβήνει το κόκκινο φως στο φανάρι.



Εικόνα 57 – Στοπ ασφαλείας (αριστερά) – Επιστροφή σε αρχική θέση (δεξιά)

Στην συνέχεια ακολουθεί η λίστα των ενεργειών των inputs και outputs του σταθμού καθώς και τα σύμβολα που αντιστοιχούν σε κάθε αισθητήριο, πνευματικό κύλινδρο ή βαλβίδα. Στην τελευταία στήλη βλέπουμε τον τύπο αυτών των συσκευών.

	ACTION	SIMBOL	TYPE
INPUTS	Start push button	m	N.O.
	Stop push button	p	N.O.
	Integrated selector	int	Comnutator
	Reset push button	r	N.O.
	Line stopper elevator downward	b1	Reed magnetic switch
	Vertical manipulator downward	c1	Reed magnetic switch
	Horizontal manipulator backward	d0	Reed magnetic switch
	Horizontal manipulator forward	d1	Reed magnetic switch
	Platform positioner cylinder backward	e0	Reed magnetic switch
	Platform positioner cylinder forward	e1	Reed magnetic switch
	Platform elevator cylinder downward	f1	Reed magnetic switch
	Bay_1 stopper cylinder backward	g1	Reed magnetic switch
	Bay_2 stopper cylinder backward	h1	Reed magnetic switch
	Vacuum in cups	v	Pressure sensor PNP
	Maximun queue detector	dc	Proximity switch PNP
Emergency Stop	em	N.O.	
OUTPUTS	Bottle stoppers cylinders forwards	A+	Single solenoid valve
	Line stopper elevator downwards	B+	Single solenoid valve
	Vertical manipulator downwards	C+	Single solenoid valve
	Horizontal manipulator forwards	D+	Single solenoid valve
	Platform positioner cylinder forwards	E+	Single solenoid valve
	Platform elevator cylinder downwards	F+	Single solenoid valve
	Lot_1 stopper cylinder backwards	G+	Single solenoid valve
	Lot_2 stopper cylinder backwards	H+	Single solenoid valve
	Vacuum in cups	V+	Single solenoid valve
	Green light	Lg	Neon light
	Orange light	Lo	Neon light
	Red light	Lr	Neon light

## 5. Σχεδίαση και εφαρμογή σε πρακτορικό σύστημα

Σε αυτό το κεφάλαιο θα περιγραφεί και θα αναλυθεί η λειτουργία του κώδικα που χρησιμοποιούμε στο πρακτορικό σύστημα. Είναι ένας συνδυασμός των γλωσσών PLC, SFC και ST με την γλώσσα Java μέσω της πλατφόρμας JADE. Παρουσιάζεται δηλαδή μία υβριδική λύση που συνδυάζει τις δύο τεχνολογίες για καλύτερο αποτέλεσμα. Για τον προγραμματισμό του πρακτορικού συστήματος και την διενέργεια των πειραμάτων γίνεται χρήση του σταθμού παραγωγής (2<sup>ος</sup> σταθμός).

### 5.1. Ορισμός προβλήματος και αιτιολόγηση επιλογής της χρήσης πρακτόρων

Η χρήση των PLC για τον έλεγχο των γραμμών παραγωγής είναι καθιερωμένη πλέον στην βιομηχανία. Αυτό συμβαίνει γιατί υπάρχουν πολλά πλεονεκτήματα αλλά και το ότι τα PLC είναι σχεδιασμένα να πετυχαίνουν ανταποκρίσεις σε πραγματικό χρόνο με εγγυημένο κύκλο σάρωσης που μερικές φορές φτάνει στα μόλις 50μsec. Παρ' όλες όμως τις προσπάθειες των τελευταίων χρόνων για βελτίωση, τα PLC έχουν πολλούς περιορισμούς όσον αφορά την προσαρμοστικότητα τους σε ένα σύστημα. Η αρχιτεκτονική του λογισμικού τους βασίζεται σε τερματικά εισόδων/εξόδων τα οποία δεν συνδέονται απαραίτητως σε συσκευές και μηχανισμούς. Αυτό σημαίνει ότι δεν υπάρχει άμεση σύνδεση μεταξύ των μηχανισμών των σταθμών και της λογικής του κώδικα, κάτι το οποίο απαιτείται σε συστήματα "plug and produce". Το χαρακτηριστικό των συστημάτων αυτών είναι ότι μπορούν να προσαρμόζονται τις εναλλαγές των απαιτήσεων του συστήματος. Πιο συγκεκριμένα, με την τεχνολογία "plug and produce" το σύστημα αντιδρά άμεσα σε αλλαγές του περιβάλλοντος και προσαρμόζεται σε αυτές με έναν ευέλικτο τρόπο. Έτσι δεν χρειάζεται η αλλαγή όλου του συστήματος από την αρχή, παρά μόνο η αυτόματη προσαρμογή των επιμέρους σημείων. Ως συνέπεια, κάθε εξάρτημα που υπάρχει στο σύστημα έχει ενσωματωμένο έναν πράκτορα λογισμικού ο οποίος αντιλαμβάνεται όλες τις αλλαγές του περιβάλλοντος και τις απαιτήσεις που ζητούνται και προσαρμόζεται ανάλογα. Ενσωματώνοντας λοιπόν τους πράκτορες κατευθείαν μέσα σε μηχανισμούς αυτοματισμών είναι μία προσέγγιση ούτως ώστε να ξεπεράσουμε αυτούς τους περιορισμούς.

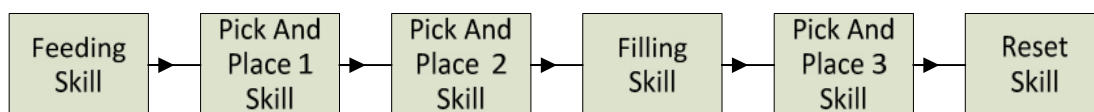
Χρησιμοποιώντας την προσέγγιση των πρακτόρων που μας παρέχει μία αντικειμενοστραφή αρχιτεκτονική, μας δίνει πλεονεκτήματα όπως η προσαρμοστικότητα, μεταφορά μεθόδων, επεκτασιμότητα και άλλα. Παρ' όλο που τα πλεονεκτήματα των πρακτορικών συστημάτων έχουν παρουσιαστεί σε πολλά συστήματα παραγωγής, η βιομηχανία δεν τα έχει ακόμα αποδεχτεί αρκετά. Αυτό συμβαίνει για πολλούς λόγους. Το να υιοθετηθεί αυτή η εναλλακτική θα σήμαινε τεράστιες αλλαγές στην βιομηχανία η οποία δεν είναι διατεθειμένη να αλλάξει συστήματα τα οποία δουλεύουν σωστά. Αντιθέτως προτιμάει να κάνει αλλαγές βήμα-βήμα τις οποίες να υποστηρίζουν τα PLC. Ακόμα τα περισσότερα συστήματα πρακτόρων χρησιμοποιούν πλατφόρμες βασισμένες σε Java γλώσσα όπου είναι γνωστά τα θέματα που έχει σε ανταποκρίσεις πραγματικού χρόνου. Η γλώσσα Java είναι μία γλώσσα υψηλού επιπέδου η οποία έχει ενσωματωμένα εργαλεία για προγραμματισμό που χρειάζεται ανταποκρίσεις πραγματικού ή και μη πραγματικού

χρόνου. Επίσης παρέχει ένα περιβάλλον εργασίας το οποίο είναι πολύ πιο παραγωγικό από πλατφόρμες άλλων γλωσσών. Έχει πολύ μεγάλο αριθμό από βιβλιοθήκες που παρέχονται από ανεξάρτητους προγραμματιστές και έχει μία πολύ μεγάλη κοινότητα που υποστηρίζει αυτή την γλώσσα που αποτελείται ακόμα και από μεγάλες εταιρίες όπως IBM, SUN και Oracle. Τέλος, είναι πολύ πιο ασφαλής σαν γλώσσα συγκρινόμενη με γλώσσες προγραμματισμού χαμηλού επιπέδου. Αυτό συμβαίνει λόγω των δυνατοτήτων που έχει η Java για λειτουργίες εντοπισμού λαθών κατά την διάρκεια του προγραμματισμού.

Η λύση που χρησιμοποιείται στην συγκεκριμένη εργασία προτείνει τον συνδυασμό των δύο τεχνολογιών, όπως αναφέρθηκε και νωρίτερα, με σκοπό να δημιουργηθεί ένα υβριδικό σύστημα το οποίο να συνδυάζει την άριστη απόδοση των PLC και την προσαρμοστικότητα των πρακτορικών συστημάτων. Έτσι, μπορούμε να ξεχωρίσουμε την επικοινωνία των πρακτόρων και την εκτέλεση των διαδικασιών από το PLC για ακόμα καλύτερη διαχείριση του συστήματος.

## 5.2. Διαμόρφωση σταθμού με πρακτορικό σύστημα

Ο σκοπός του σταθμού στον οποίο εφαρμόζεται αυτή η λύση είναι να πάρει ένα κουτί, να του προσθέσει συγκεκριμένο βάρος υλικού (στην προκειμένη περίπτωση 15 γραμμάρια) και να το απελευθερώσει πάνω στον ιμάντα μεταφοράς. Για να γίνει αυτή η παραγωγική διαδικασία, χρησιμοποιεί ένα σύνολο διαδικασιών (skills) που εκτελούνται σε συγκεκριμένη σειρά [Εικόνα 58]. Αρχικά τροφοδοτείται η πλατφόρμα του σταθμού με ένα κουτί (feeding skill). Στην συνέχεια μεταφέρεται μέσω των μηχανισμών των πνευματικών κυλίνδρων μπροστά από τον αναγνώστη barcode, πάνω στον ιμάντα μεταφοράς (pick and place 1 skill). Μετά την ανάγνωση μεταφέρεται πάνω στην ζυγαριά (pick and place 2 skill) όπου ξεκινάει το γέμισμα του κουτιού με προϊόν (filling skill). Στην συνέχεια το κουτί μεταφέρεται και πάλι στον ιμάντα μεταφοράς όπου απελευθερώνεται (pick and place 3 skill). Στο τέλος όλο το σύστημα επανέρχεται στην αρχική θέση του και σε κατάσταση αναμονής κάνοντας χρήση της διαδικασίας επαναφοράς (reset skill).



Εικόνα 58 – Διαδικασία παραγωγής

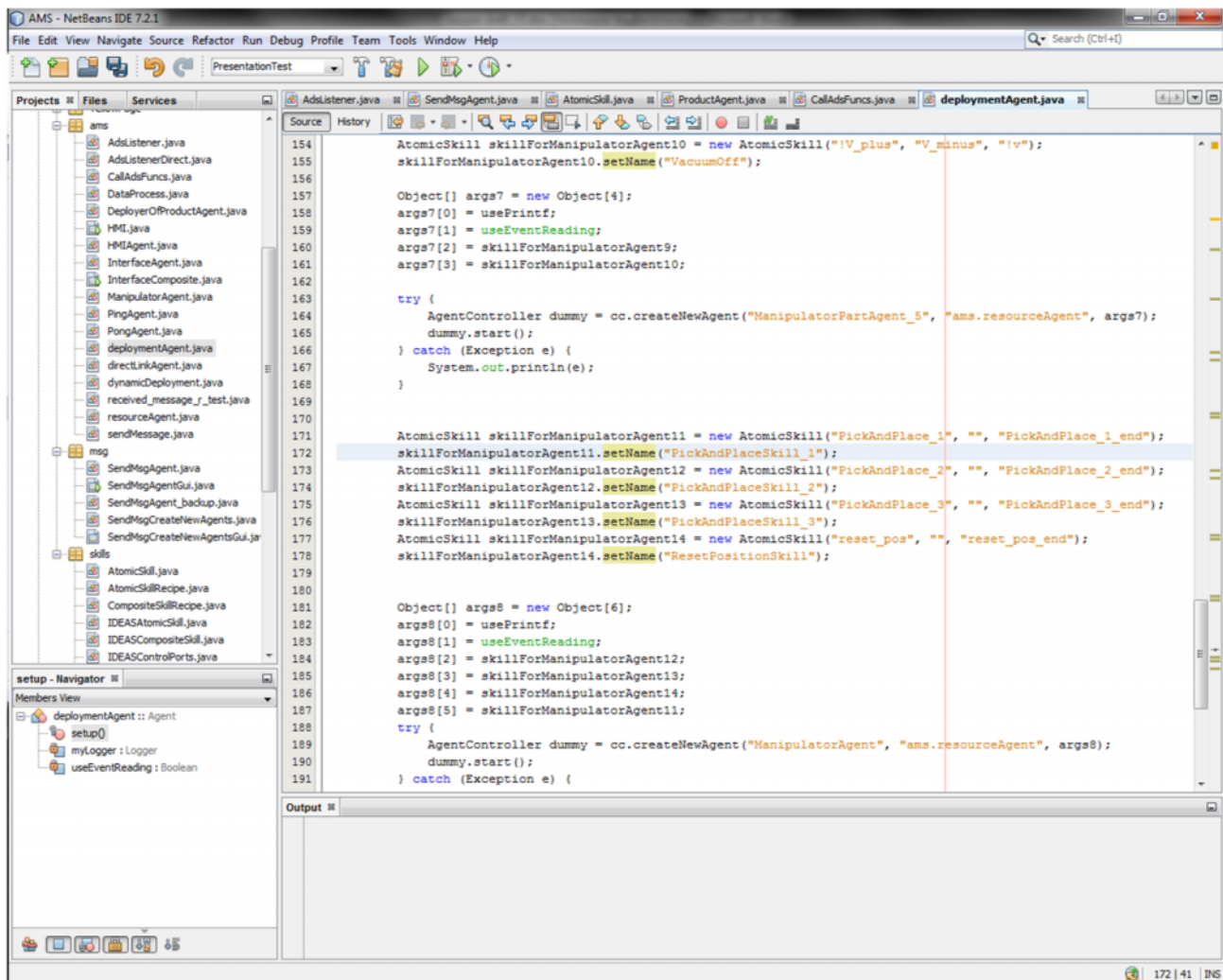
Ο κώδικας που υπάρχει πλέον στο PLC αποτελείται από υπορουτίνες γραμμένες σε γλώσσες SFC και ST που αντιστοιχούν σε κάθε διαδικασία (skill) του πρακτορικού συστήματος και καλείται κάθε μία ή και ένα σύνολο αυτών για κάθε διαδικασία που θέλουμε να εκτελεστεί.

## 5.3. Λογισμικό διαχείρισης πρακτορικού συστήματος

Από την μεριά του κώδικα του πρακτορικού συστήματος, χρησιμοποιείται ένα διαφορετικό λογισμικό για να γίνει ο προγραμματισμός όλων των πρακτόρων. Συγκεκριμένα χρησιμοποιείται η πλατφόρμα NetBeans και ο προγραμματισμός γίνεται σε γλώσσα Java [Εικόνα 59]. Η πλατφόρμα αυτή είναι ένα ολοκληρωμένο

περιβάλλον ανάπτυξης κυρίως για γλώσσα Java αλλά και για άλλες γλώσσες όπως HTML 5, PHP και C/C++. Είναι δωρεάν πλατφόρμα της εταιρίας Oracle η οποία έχει μία τεράστια κοινότητα που την υποστηρίζει. Έχει λειτουργίες οι οποίες επιτρέπουν στον προγραμματιστή να επεξεργάζεται τον κώδικά του γρήγορα και έξυπνα, να ελέγχει για προβλήματα πολύ αποτελεσματικά, να διαχειρίζεται εύκολα και ουσιαστικά τα project του και όλα αυτά σε ένα περιβάλλον χρήσης που βελτιώνεται και ανανεώνεται συνεχώς.

Η επικοινωνία μεταξύ του PLC λογισμικού TwinCat και του NetBeans και ως συνέπεια η επικοινωνία των πρακτόρων με το PLC, γίνεται μέσω της βιβλιοθήκης της εταιρίας Beckhoff, TwinCat Java Communication η οποία χρησιμοποιεί την τεχνολογία ADS (Automation Device Specification). Η τεχνολογία αυτή προσφέρει άμεση επικοινωνία της γλώσσας Java με τις εισόδους και εξόδους του PLC. Επίσης προσφέρει μεθόδους για άμεση εποπτεία και ενημέρωση σε περίπτωση αλλαγής κάποιων μεταβλητών χωρίς την ανάγκη για κάποιο πρόσθετο λογικό επίπεδο το οποίο θα παρακολουθεί συνεχώς για αλλαγές και θα μειώνει σημαντικά την απόδοση του συστήματος.



Εικόνα 59 - NetBeans

#### **5.4. Η αρχιτεκτονική της επικοινωνίας του πρακτορικού συστήματος με το PLC**

Για την πραγματοποίηση των πειραμάτων που ακολουθούν, υπήρχε η ανάγκη να χρησιμοποιηθεί ένα μοντέλο πρακτόρων το οποίο θα είναι απλό, χωρίς περιττές επικοινωνίες και λειτουργίες, αλλά θα πραγματοποιεί την διαδικασία με επιτυχία.

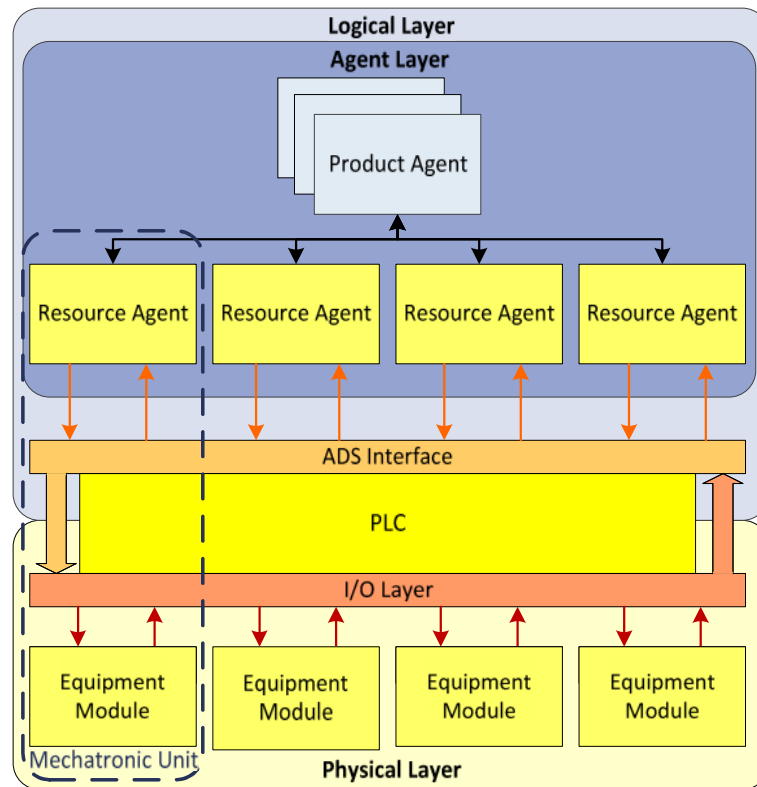
Η χρήση μεγαλύτερου αριθμού και διαφορετικών τύπων πρακτόρων μας δίνει την δυνατότητα να έχουμε πράκτορες οι οποίοι αυτοματοποιούν πολύ περισσότερο το σύστημα μας, παρέχοντας δυνατότητες όπως αυτόματη προσαρμογή της διαδικασίας σε περίπτωση αλλαγής του εξοπλισμού ή την παροχή της δυνατότητας διαχείρισης της ουράς των προϊόντων που χρειάζεται να φτιαχτούν βάση προτεραιότητας, ανάγκης άμεσης ολοκλήρωσης και άλλα. Ακόμα μπορούν να χρησιμοποιηθούν πράκτορες για περαιτέρω και πιο λεπτομερή ανάπτυξη της επικοινωνίας μεταξύ των υπολοίπων πρακτόρων και ολόκληρου του συστήματος με τον χρήστη.

Το μοντέλο που προτείνεται στην συγκεκριμένη πτυχιακή εργασία αποτελείται από δύο τύπους πρακτόρων [Εικόνα 60]. Οι λόγοι για την δημιουργία του βασικού αυτού μοντέλου, όπως προαναφέρθηκε είναι η ανάγκη για ένα μοντέλο που να μπορεί να λειτουργήσει με την χρήση του βασικού αριθμού πρακτόρων και να αποδώσει όσο το δυνατόν καλύτερα. Ο αριθμός των πρακτόρων έχει την δυνατότητα να αυξηθεί ανά πάσα στιγμή αν χρειάζεται να αλλάξουμε το ακόλουθο μοντέλο. Έτσι έχουμε δημιουργήσει τον Product Agent (PA) και τον Resource Agent (RA). Ο στόχος του PA είναι να ερμηνεύει την ροή των προϊόντων, να κατανοεί τις απαιτήσεις του συστήματος και να αναγνωρίζει ποιος RA μπορεί να ανταπεξέλθει στις συγκεκριμένες απαιτήσεις. Αυτό συμβαίνει μέσω διαπραγμάτευσης με τους RA για το ποιος ακριβώς μπορεί να πραγματοποιήσει την διαδικασία. Οι RA είναι παθητικοί πράκτορες οι οποίοι επικοινωνούν μέσω της τεχνολογίας ADS με το PLC κώδικα ούτως ώστε να πραγματοποιήσουν τα skills τα οποία έχουν διαπραγματευθεί να εκτελέσουν. Στην συνέχεια βλέπουμε στην Εικόνα 60 την απεικόνιση του μοντέλου της επικοινωνίας ολόκληρου του συστήματος.

Η επικοινωνία μεταξύ των πρακτόρων παίζει τον πιο σημαντικό ρόλο στο κομμάτι της διαφοράς της ταχύτητας εκτέλεσης του κώδικα των πρακτόρων με του κώδικα PLC. Οι δύο αυτοί κώδικες, ο καθарός PLC κώδικας και ο υβριδικός, αποτελούνται από τις ίδιες ακριβώς εντολές και ο λόγος που η επικοινωνία παίζει τόσο σημαντικό ρόλο είναι ότι η διαφορά τους υφίστανται στην εκτέλεση της λογικής σειράς που καλείται κάθε διαδικασία (skill). Αυτό συμβαίνει γιατί όταν τρέχουμε το πρόγραμμα από κώδικα PLC, η λογική σειρά που καλείται η κάθε διαδικασία (skill) επιλέγεται από την γλώσσα SFC και τον τρόπο με τον οποίο είναι γραμμένος ο κώδικας. Έτσι ο χρόνος που χρειάζεται να παρθεί η απόφαση για να ακολουθηθεί ένα συγκεκριμένο κομμάτι κώδικα ή υπορουτίνας κατά την διάρκεια εκτέλεσης του προγράμματος είναι ο μικρότερος δυνατός και αντιστοιχεί συνήθως στον χρόνο που κάνει ένας κύκλος PLC. Ακόμα, η επικοινωνία του κώδικα PLC και των μεταβλητών του, με τις βαλβίδες και τους αισθητήρες, γίνεται απευθείας από τις εισόδους εξόδους του PLC και δεν παρεμβαίνει η τεχνολογία ADS. Κατά την



διάρκεια της εκτέλεσης όμως του κώδικα των πρακτόρων, έχουμε τον παραπάνω χρόνο που χρειάζεται να επικοινωνήσουν οι πράκτορες μεταξύ τους και με το σύστημα για να εκτελεστεί η κάθε διαδικασία.



Εικόνα 60 – Υβριδική αρχιτεκτονική

Πιο συγκεκριμένα για την εφαρμογή της τεχνολογίας ADS στα τρέχοντα πειράματα, βλέπουμε ότι ο RA είναι ο πράκτορας που διενεργεί την επικοινωνία ανάμεσα στον κώδικα PLC και τον κώδικα των πρακτόρων. Κάθε διαδικασία στον κώδικα PLC έχει μια συγκεκριμένη μεταβλητή που πρέπει να αλλάξει για να ξεκινήσει η εκτέλεση της και μία ακόμα που αλλάζει από το ίδιο το PLC όταν ολοκληρωθεί η εκτέλεση της. Ο πράκτορας αυτός αλλάζει τιμή στην μεταβλητή εκκίνησης που υπεύθυνη για την εκτέλεση της κάθε διαδικασίας. Την αλλαγή αυτή την εντοπίζει άμεσα η τεχνολογία ADS και την μεταφέρει στο PLC και στο πρόγραμμα του. Αυτό με την σειρά του εκτελεί την αντίστοιχη διαδικασία. Στην ανάποδη περίπτωση όπου μία υπορουτίνα του PLC ολοκληρώσει την διαδικασία της, αλλάζει την τιμή της αντίστοιχης μεταβλητής τέλους, και μέσω των εισόδων του PLC ενημερώνεται το πρόγραμμα και στην συνέχεια η τεχνολογία ADS που μεταφέρει την πληροφορία αυτή στον κώδικα των πρακτόρων.

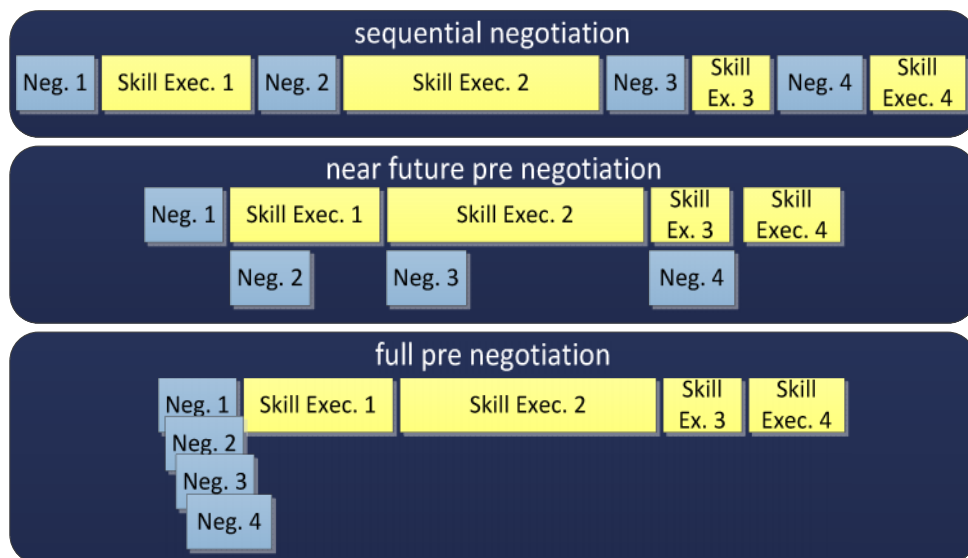
Για τον παραπάνω λόγω θα εκτελέσουμε τρεις διαφορετικές διαδικασίες διαπραγμάτευσης και θα συγκρίνουμε τα αποτελέσματα που παίρνουμε όσον αφορά την ταχύτητα τους [Εικόνα 61].

Στην πρώτη περίπτωση (Sequential Negotiation) που είναι και η πιο κλασσική, οι πράκτορες διαπραγματεύονται την επόμενη διαδικασία, εφόσον έχουν ολοκληρώσει την προηγούμενη. Με αυτό τον τρόπο έχουμε μεγαλύτερη

κατανάλωση χρόνου αλλά το σύστημα μπορεί να προσαρμοστεί σε οποιοσδήποτε αλλαγές συμβούν κατά την διάρκεια της εκτέλεσης ολόκληρης της διαδικασίας γιατί προκειμένου να αποφασίσει ποια διαδικασία θα εκτελέσει μετά περιμένει να ολοκληρωθεί η προηγούμενη και ελέγχει τι είναι πραγματικά διαθέσιμο προς εκτέλεση την στιγμή που χρειάζεται.

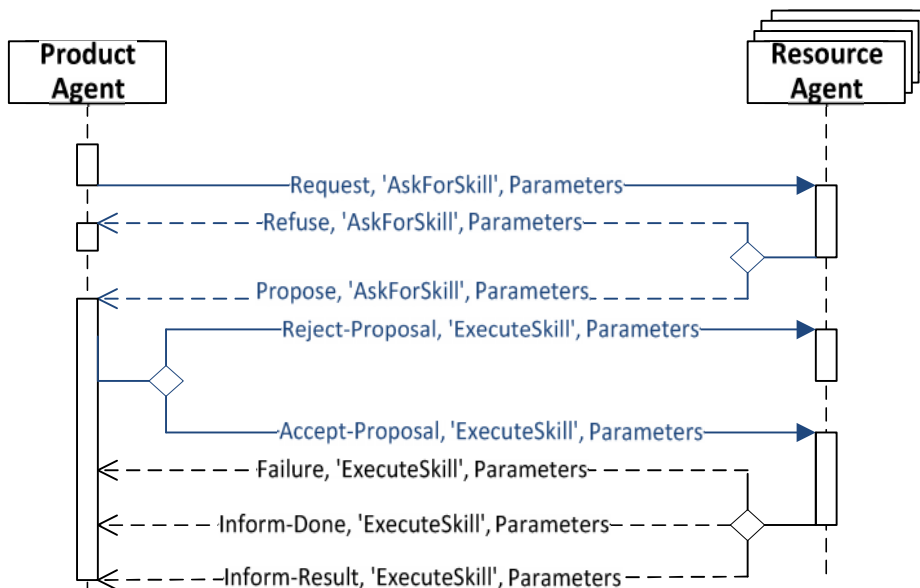
Η επόμενη περίπτωση διαπραγμάτευσης είναι η near future pre negotiation όπου εδώ γίνεται η διαπραγμάτευση για την επόμενη διαδικασία την ώρα που ξεκινάει η προηγούμενη. Αυτό βελτιώνει τον συνολικό χρόνο για κρατά μια ισορροπία όσον αφορά την προσαρμογή του συστήματος σε αλλαγές.

Η τελευταία περίπτωση ονομάζεται full pre negotiation και είναι η περίπτωση όπου όλες οι διαπραγματεύσεις θα γίνουν πριν ξεκινήσει το σύστημα τις διαδικασίες του. Αυτό θα μας δώσει την ταχύτερη διαδικασία ειδικά στην περίπτωση που οι χρόνοι εκτέλεσης μειώνονται αλλά μας παρέχει πολύ λιγότερη ευελιξία στο θέμα της προσαρμογής του συστήματος σε αλλαγές. Αυτό συμβαίνει γιατί από την στιγμή που τελειώνουν οι διαπραγματεύσεις, το σύστημα ξεκινά να εκτελεί την μία διαδικασία μετά την άλλη, αλλά από την στιγμή που ο μηχανισμός διαπραγμάτευσης δεν ξαναχρησιμοποιείται μέχρι να τελειώσει και η τελευταία διαδικασία, οποιαδήποτε διαδικασία δεν είναι διαθέσιμη για κάποιο λόγο, θα μας καθυστερήσει το σύνολο της διαδικασίας.



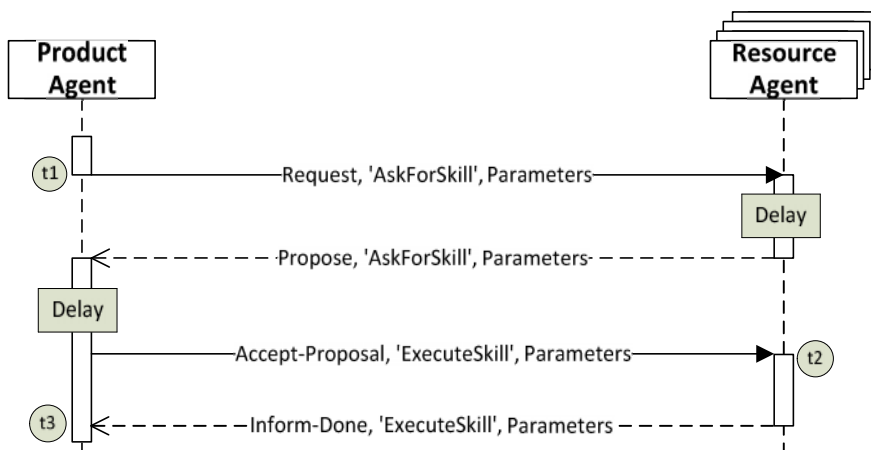
Εικόνα 61 – Περιπτώσεις διαπραγματεύσεων μεταξύ πρακτόρων

Για να συνειδητοποιήσουμε καλύτερα την αλληλεπίδραση των πρακτόρων στο σύστημα, θα πρέπει το πρωτόκολλο επικοινωνίας που θα χρησιμοποιήσουμε να είναι ξεκάθαρο. Το πιο απλό πρωτόκολλο επικοινωνίας και αυτό το οποίο χρησιμοποιούμε είναι το FIPA Contract Net Protocol [Εικόνα 62 - Πρωτόκολλο επικοινωνίας]. Ένα βασικό σημείο που αυτό το πρωτόκολλο μας βοηθάει να επιτύχουμε, είναι να μειώσουμε όσο το δυνατόν περισσότερο το χρόνο διαπραγμάτευσης, χρησιμοποιώντας μηνύματα τα οποία είναι όσο πιο απλά και άμεσα γίνεται. Με αυτό τον τρόπο, σε περίπτωση επιτυχίας χρειάζεται η ανταλλαγή τριών μηνυμάτων με το τρίτο να είναι η εντολή για εκτέλεση της εντολής.



Εικόνα 62 - Πρωτόκολλο επικοινωνίας

Λόγω του μικρού αριθμού πρακτόρων στα πειράματα που αναλύονται στην συνέχεια, η επικοινωνία μεταξύ του RA και του PA αναμένεται να είναι πολύ μικρή. Σε χρήση υπό πραγματικές συνθήκες η επικοινωνία θα είναι πιο πολύπλοκη και λόγω αυτού καθώς επίσης και για να επιτύχουμε πιο ρεαλιστικά αποτελέσματα, προστέθηκε στην διαδικασία καθυστέρηση (delay) που μιμείται την διαδικασία λήψης απόφασης των πρακτόρων [Εικόνα 63]. Πιο συγκεκριμένα, οι καθυστερήσεις τοποθετούνται στον RA και στον PA αφού ληφθεί η αίτηση για την διαδικασία (skill) και προτού σταλθεί το μήνυμα αποδοχής αντίστοιχα. Για καλύτερη ανάλυση του κατά πόσο επηρεάζουν οι καθυστερήσεις την αντίδραση του συστήματος, τα πειράματα γίνονται αρχικά χωρίς καθυστερήσεις και μετά με καθυστερήσεις 20ms, 50ms και 100ms.



Εικόνα 63 - καθυστερήσεις στην διαδικασία λήψης απόφασης

Οι RA αναπτύσσονται στην εκκίνηση του συστήματος αναλόγως με τις μονάδες των εξαρτημάτων που υπάρχουν στο σύστημα. Κάποιοι από αυτούς είναι υπεύθυνοι για την εκτέλεση μίας μόνο διαδικασίας ενώ άλλοι εκτελούν

περισσότερες διαδικασίες. Όταν ένα προϊόν ζητείτε να παραχθεί, ξεκινάει ο RA και ξεκινά την διαπραγμάτευση για την εκτέλεση των διαδικασιών.

### 5.5. Σενάρια πειραμάτων

Για να επαληθευτούν οι προαναφερθείσες προσεγγίσεις, επιλέχτηκαν τρία διαφορετικά σενάρια για να εκτελεστούν.

Το πρώτο σενάριο χρησιμοποιεί αποκλειστικά τον κώδικα PLC που αναπτύχθηκε για αυτή την συγκριτική μελέτη. Πιο συγκεκριμένα, ο κώδικας που είναι γραμμένος στις γλώσσες προγραμματισμού για PLC, SFC και ST, φορτώνεται στον ελεγκτή του σταθμού και από εκεί εκτελεί την προγραμματισμένη διεργασία. Υπάρχουν μετρητές χρόνου που μετράνε τον χρόνο που χρειάζεται ο ελεγκτής και ο σταθμός για να ολοκληρώσουν όλη την διεργασία καθώς και επιμέρους διεργασίες. Οι μετρητές αυτοί αποτελούν μέρος του λογισμικού του PLC και χρησιμοποιούνται μέσω των βιβλιοθηκών που παρέχονται από το TwinCat. Για να διασφαλιστεί η αξιοπιστία των μετρητών αυτών, μετρήθηκαν διαδικασίες για τις οποίες γνωρίζαμε εκ των προτέρων τον χρόνο που χρειάζονται. Οι μετρητές αυτοί ξεκινάνε να μετράνε την στιγμή που ο RA αλλάζει τιμή στην μεταβλητή που είναι υπεύθυνη για να ξεκινήσει η αντίστοιχη διαδικασία και σταματάνε όταν η μεταβλητή τέλους αλλάξει από το σύστημα PLC.

Κατά την διάρκεια του δεύτερου σεναρίου, υπάρχει κώδικας που περιέχει agent σε γλώσσα Java η οποία τρέχει στον υπολογιστή. Οι agent εκτελούνται μέσω της πλατφόρμας JADE και χρησιμοποιούν την επεξεργαστική ισχύ του υπολογιστή αλλά ενεργοποιούν τις διαδικασίες του σταθμού, εκτελώντας κομμάτια κώδικα PLC. Οι μετρητές χρόνου βρίσκονται στο PLC και ταυτόχρονα στον κώδικα Java και μετράνε τον χρόνο της κάθε επιμέρους διεργασίας ξεχωριστά καθώς και τον συνολικό χρόνο που χρειάζεται για να γίνει ολοκληρωμένη η διεργασία. Οι μετρητές υπάρχουν και στις δύο πλατφόρμες για επιβεβαίωση των σωστών αποτελεσμάτων. Κατά την διάρκεια αυτού του σεναρίου, χρησιμοποιούνται οι τρεις διαφορετικές περιπτώσεις επικοινωνίας και διαπραγμάτευσης μεταξύ των πρακτόρων που αναφέρθηκαν στην προηγούμενη ενότητα.

Στο τρίτο σενάριο είναι και πάλι η υβριδική προσέγγιση ανάμεσα στον κώδικα PLC και την Java αλλά αυτή την φορά δεν θα υπάρχουν οι πράκτορες για να πραγματοποιούν την διαπραγμάτευση. Οι υπορουτίνες εκτελούνται με συγκεκριμένη σειρά κατευθείαν από τον Java κώδικα και με αυτό τον τρόπο απαλείφουμε τελείως τον χρόνο των διαπραγματεύσεων και μπορούμε να αξιολογήσουμε καλύτερα την ενσωμάτωση του Java κώδικα στο πρόγραμμα μας. Έτσι χρησιμοποιούμε μόνο την τεχνολογία ADS η οποία μας παρέχει την επικοινωνία μεταξύ των πρακτόρων και του PLC.

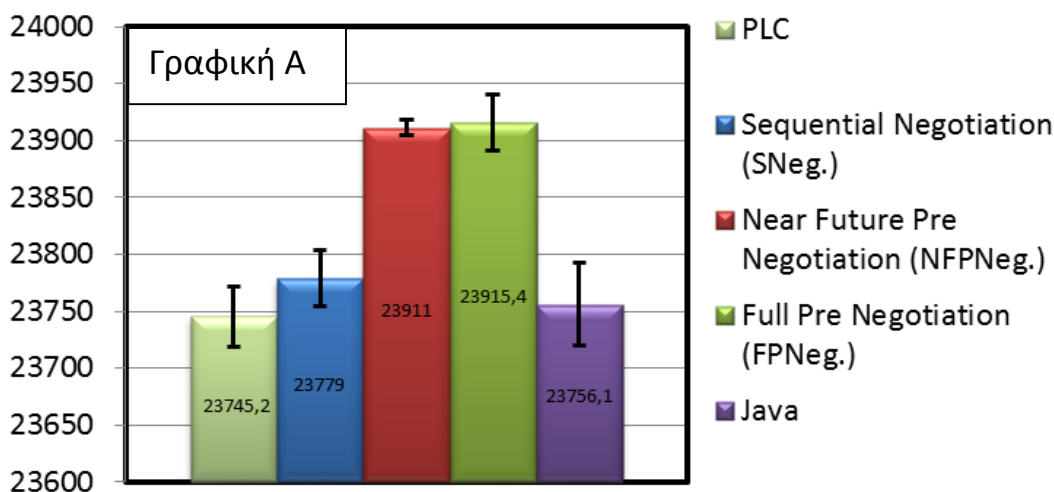
Για την διασφάλιση της αξιοπιστίας των πειραμάτων, κάθε πείραμα πραγματοποιείται δέκα φορές ούτως ώστε να είναι δυνατή η χρησιμοποίηση του μέσου όρου των πειραμάτων. Για να καταλήξουμε σε αυτό το σύνολο πειραμάτων, πραγματοποιήθηκε και ένας μεγάλος αριθμός πειραμάτων που δεν αναφέρονται σε αυτή την εργασία. Τα πειράματα αυτά βελτιώνονταν προοδευτικά βάση των αποτελεσμάτων που λαμβάναμε και των απαραίτητων αλλαγών που γινόντουσαν.

Αλλαγές όπως βελτίωση στον κώδικα του PLC και απαλοιφή των μη απαραίτητων εντολών, βελτίωση του κώδικα των πρακτόρων για σωστή και με άμεση ανταπόκριση χρήση της τεχνολογίας ADS καθώς επίσης αλλαγές στον εξοπλισμό για την επιβεβαίωση ότι υπάρχουν πάντα σταθερές συνθήκες διεξαγωγής των πειραμάτων όπως για παράδειγμα σταθερή πίεση αέρα στα πνευματικά μέρη του συστήματος.

## 6. Αποτελέσματα και συμπεράσματα

Σε αυτό το κεφάλαιο θα παρουσιαστούν τα αποτελέσματα των προαναφερθέντων σεναρίων των πειραμάτων. Όλες οι τιμές των αποτελεσμάτων που θα παρουσιαστούν στην συνέχεια προέρχονται από τον μέσο όρο των τιμών για δέκα αλληπάλληλα πειράματα.

Στο πρώτο σενάριο των πειραμάτων, βλέπουμε τον χρόνο που χρειάζεται το σύστημα για να ολοκληρώσει έναν πλήρη κύκλο με χρήση αποκλειστικά του κώδικα PLC. Ο χρόνος αυτός είναι 23745,2ms και όπως βλέπουμε και από την γραφική Α [Εικόνα 64], η οποία μας δείχνει τον συνολικό χρόνο ολόκληρης της διαδικασίας, είναι ο πιο γρήγορος χρόνος που παίρνουμε. Κάτι τέτοιο είναι απολύτως αναμενόμενο από την στιγμή που έχουμε καθαρό κώδικα PLC, χωρίς την παρεμβολή κώδικα πρακτόρων και το σύστημα εκτελεί μια διαδικασία για την οποία είναι σχεδιασμένο να έχει την μέγιστη δυνατή απόδοση. Αυτό είναι σημαντικό γιατί εδραιώνει μία βάση σύγκρισης για την υβριδική προσέγγιση.

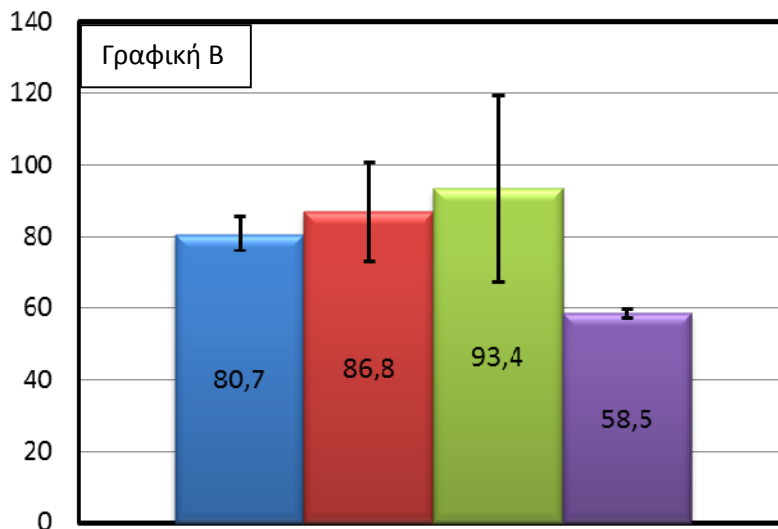


Εικόνα 64 - Πλήρης κύκλος προϊόντος χωρίς καθυστερήσεις

Στο δεύτερο σενάριο χρησιμοποιούμε τον πλήρη υβριδικό κώδικα ανάμεσα στο PLC και τους πράκτορες. Στις γραφικές παραστάσεις Β,Γ,Δ,Ε [Εικόνα 66-Εικόνα 67] συγκρίνουμε τον πρόσθετο χρόνο που χρειάζεται για τις διαπραγματεύσεις το υβριδικό σύστημα. Αυτή η τιμή προκύπτει από την άθροιση των μετρητών χρόνου του PLC των επιμέρους διαδικασιών και την αφαίρεση αυτού του χρόνου από την τιμή του μετρητή του συνολικού χρόνου της διαδικασίας.

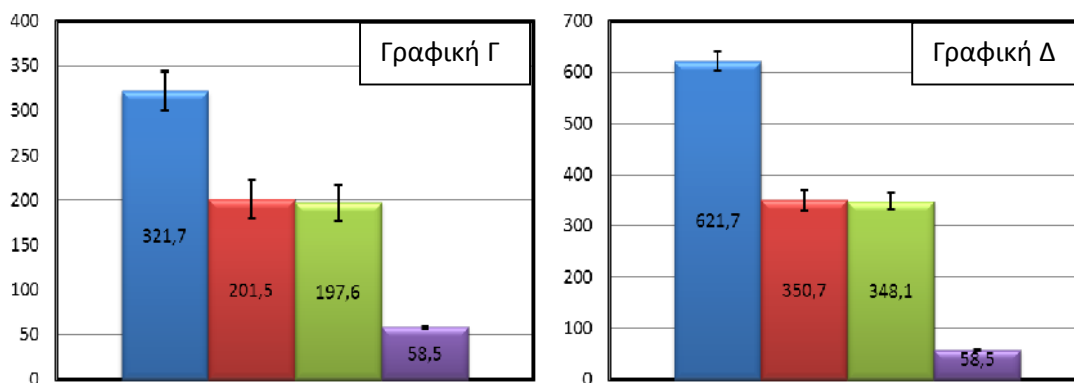
Πιο αναλυτικά, στην γραφική Β [Εικόνα 65] συγκρίνοντας την πιο απλή διαδικασία διαπραγμάτευσης (no delays) μπορεί κάποιος να διακρίνει ότι η μεγαλύτερη τιμή του πρόσθετου χρόνου που χρειάζεται για τις διαπραγματεύσεις είναι 93,4ms που σημαίνει ότι είναι 15,57ms κατά μέσο όρο για κάθε μία από τις επιμέρους διαδικασίες της παραγωγής. Παρ' όλο που αυτά τα αποτελέσματα είναι μέρος πειράματος, μας δείχνουν ότι η υβριδική προσέγγιση που ακολουθήθηκε έχει λογικά αποτελέσματα. Η διαφορά επίσης από τον κώδικα Java χωρίς διαπραγματεύσεις με την περίπτωση που χρειαζόμαστε τον περισσότερο χρόνο για

διαπραγματεύσεις είναι μόλις 34,9ms, δηλαδή 5,8ms για κάθε επιμέρους διαδικασία.



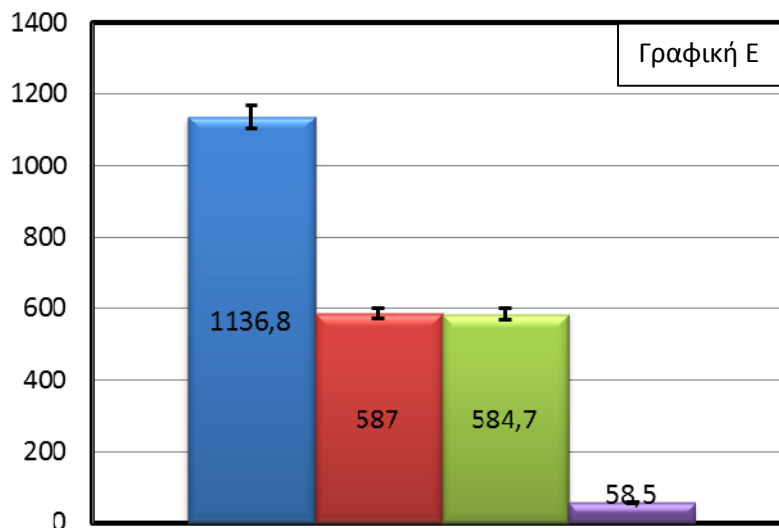
Εικόνα 65 - Χρόνος διαπραγματεύσεων χωρίς καθυστερήσεις (no delays)

Ένα αποτέλεσμα που είναι αναπάντεχο είναι το γεγονός ότι χωρίς καθυστερήσεις στις διαπραγματεύσεις (no delays), η καλύτερη απόδοση επιτυγχάνεται με την διαπραγμάτευση sequential negotiation. Αν όμως αυτό αναλυθεί πιο λεπτομερώς θα καταλάβουμε ότι μπορεί να δικαιολογηθεί από τον επιπλέον χρόνο που χρειάζεται ο επεξεργαστής για να δημιουργηθούν πολλαπλά νήματα ταυτόχρονα στο σύστημα. Αυτό δείχνει ότι υπάρχει ένα σημαντικό κόστος χρόνου στο να δημιουργούμε πολλαπλά νήματα στο σύστημα οπότε πρέπει να το συμπεριλάβουμε στις παραμέτρους που υπολογίζουμε για να επιλέξουμε την καλύτερη περίπτωση διαπραγματεύσεων. Αν αναλύσουμε τα αποτελέσματα για τις περιπτώσεις που έχουμε καθυστερήσεις στην επικοινωνία, οι οποίες μιμούνται ένα σύστημα με πολύ πιο πολύπλοκη διαδικασία αποφάσεων, θα δούμε ότι έχουμε αντιστροφή στο θέμα της επιλογής της καλύτερης στρατηγικής για διαπραγματεύσεις. Αυτό σημαίνει ότι υπάρχει ένα σημείο καμπής όπου κάποιος μπορεί να αλλάξει προσέγγιση αναλόγως τον παραπάνω χρόνο που χρειάζεται για να παρθούν οι αποφάσεις.



Εικόνα 66 - Χρόνος διαπραγματεύσεων χωρίς καθυστερήσεις (20ms delays αριστερά, 50ms δεξιά)

Η σύγκριση, στην συνέχεια, μεταξύ της near future pre negotiation και της full pre negotiation μας δείχνει ότι υπάρχει ελάχιστη διαφορά ανάμεσα στις δύο περιπτώσεις. Αυτό δεν είναι καθόλου αναπάντεχο από την στιγμή που οι επιμέρους διαδικασίες παίρνουν σημαντικά περισσότερο χρόνο να ολοκληρωθούν σε σχέση με την διάρκεια της διαδικασίας διαπραγμάτευσης. Ουσιαστικά, η διαπραγμάτευση full pre negotiation έχει κάποιο ουσιαστικό κέρδος όταν οι διαδικασίες είναι εξαιρετικά γρήγορες και η διαδικασία και η λογική της διαπραγμάτευσης είναι πολύ πολύπλοκη.



Εικόνα 67 - Χρόνος διαπραγματεύσεων χωρίς καθυστερήσεις (100ms delays)

Στο τρίτο σενάριο έχουμε την πιο απλή υβριδική μορφή κώδικα. Συγκεκριμένα, ενώ έχουμε τον κώδικα Java να επικοινωνεί με το PLC, δεν υπάρχει ο χρόνος της διαπραγμάτευσης παρά μόνο ο χρόνος που χρειάζεται ο κώδικας Java να επικοινωνήσει με το PLC και να ζητήσει την εκτέλεση συγκεκριμένης σειράς διαδικασιών. Ο χρόνος που βλέπουμε ότι χρειάζεται το σύστημα για να εκτελέσει έναν πλήρη κύκλο προϊόντος, δηλαδή την πλήρη διαδικασία, είναι 23756,1ms. Αυτό μας δίνει μία διαφορά 10,9ms από τον χρόνο που χρειάζεται ο καθαρός κώδικας PLC. Η διαφορά αυτή αναλογικά στις 6 διαφορετικές διαδικασίες που εκτελέστηκαν σε αυτόν τον χρόνο δίνει 1,8ms περισσότερο χρόνο απ' ότι ο καθαρός κώδικας PLC σε κάθε μία διαδικασία. Αυτό που είναι επίσης ενδιαφέρον, είναι ότι σε αυτή την προσέγγιση βλέπουμε μία πολύ καλή αλληλεπίδραση μεταξύ του κώδικα PLC και μιας γλώσσας προγραμματισμού υψηλού επιπέδου η οποία μας παρέχει πολύ θετικά αποτελέσματα για την περαιτέρω αξιοποίηση του υβριδικού αυτού συστήματος.



## 6.1. Συμπεράσματα

Σε αυτή την πτυχιακή εργασία παρουσιάστηκε η υβριδική προσέγγιση για την λειτουργία ενός σταθμού παραγωγής, ανάμεσα στον κώδικα PLC και κώδικα πρακτόρων μέσω της γλώσσας προγραμματισμού Java.

Η συγκεκριμένη προσέγγιση μας δείχνει ότι είναι δυνατόν να πετύχουμε καλές αποδόσεις χρησιμοποιώντας πρακτορικά συστήματα μέσα σε συστήματα παραγωγής.

Πιο συγκεκριμένα, συγκρίνοντας τον χρόνο που κάνει ο κώδικας PLC να εκτελέσει την προκαθορισμένη διαδικασία, εδραιώνουμε μία βάση σύγκρισης για τα υπόλοιπα πειράματα. Στην συνέχεια βλέπουμε ότι η διαφορά που έχει σε θέμα χρόνου ο κώδικας PLC με αυτόν του υβριδικού συστήματος είναι μερικά μόλις ms τα οποία δεν είναι καθόλου ανασταλτικά για την περαιτέρω χρησιμοποίηση αυτής της τεχνολογίας. Τα πλεονεκτήματα που μας παρέχει η υβριδική αυτή προσέγγιση υπερκαλύπτουν την απώλεια χρόνου. Χρησιμοποιώντας την τεχνολογία των πρακτόρων, έχουμε πολύ μεγαλύτερο έλεγχο του συστήματος μας. Μπορούμε να προσαρμόσουμε τις διαδικασίες που θα εκτελεστούν γρήγορα και άμεσα χωρίς να χρειαστεί να κάνουμε δραματικές αλλαγές στον κώδικα μας.

Διακρίνουμε επίσης πολύ άμεση και γρήγορη επικοινωνία μέσω της τεχνολογίας ADS ανάμεσα στους πράκτορες και το PLC το οποίο επίσης είναι ένα πολύ θετικό στοιχείο. Ουσιαστικά δηλαδή, οι χρόνοι που χρειάζονται για επικοινωνία και διασύνδεση δεν κάνουν μεγάλη διαφορά στην απόδοση του συστήματος.

Αναλυτικότερα για τις περιπτώσεις των διαπραγματεύσεων, βλέπουμε βάση των αποτελεσμάτων ότι η προσέγγιση sequential negotiation είναι η καλύτερη για πολύ απλές διαδικασίες διαπραγματεύσεων ενώ για πιο σύνθετες διαδικασίες που περιλαμβάνουν σύνθετη λογική διαπραγματεύσεων και πολύ γρήγορο χρόνο διαδικασιών είναι καλύτερες οι άλλες δύο διαπραγματεύσεις. Πρέπει ακόμα να αναφερθεί ότι ουσιαστικά, η πολυπλοκότητα της διαδικασίας διαπραγμάτευσης είναι αυτή που κάνει την διαφορά σε θέμα χρόνου, όσον αφορά την προσέγγιση των πρακτόρων.

Η προσωπική μου άποψη, μετά από αρκετούς μήνες απασχόλησης με το κομμάτι των PLC και των πρακτόρων καθώς και τον σχεδιασμό και την εφαρμογή του υβριδικού συστήματος είναι ότι η συγκεκριμένη μελέτη που γίνεται σε αυτή την πτυχιακή εργασία μπορεί να αναπτυχθεί περαιτέρω και να ενσωματωθεί κάποια στιγμή στην βιομηχανία. Είναι αρκετά τα πλεονεκτήματα και μία πολύ καλή εναλλακτική για τον εκσυγχρονισμό των βιομηχανιών.

## Βιβλιογραφία

1. Ferreira, P., *An Agent-based Self-Configuration Methodology for Modular Assembly Systems*. Department of Mechanical, Materials and Manufacturing Engineering, The University of Nottingham, Nottingham, 2011.
2. FESTO, *Programmable Logic Controllers Basic Level*. 2002.
3. Instruments, T., *Programmable Logic Control*. 2012.
4. Mauro Onori, N.L., Jose Barata, Christoph Hanisch, *The IDEAS project: plug & produce at shop-floor level*. Assembly Automation.
5. Mihaela MUNTEAN, C.T., *Mobile Agents Systems in Wireless Applications*. Journal of Mobile, Embedded and Distributed Systems, 2009. **vol. 1, no. 1**.
6. Pedro Ferreira, N.L., *Configuration Model for Evolvable Assembly Systems*. presented at the 4th CIRP Conference On Assembly Technologies And Systems Ann Arbor, Michigan, USA, 2012.
7. Pedro Ferreira, S.D., Athanasios Anagnostopoulos, Flavio Pascoa, Niels Lohse, *A Performance evaluation of Industrial Agents*. IECON 2013, 2013.
8. SIEMENS, *Αυτοματισμός με SIMATIC S7*.
9. Beckhoff, *Beckhoff Documentation*. 2012:  
<http://www.beckhoff.com/english.asp?download/documentation.htm?id=70997977100312>.