

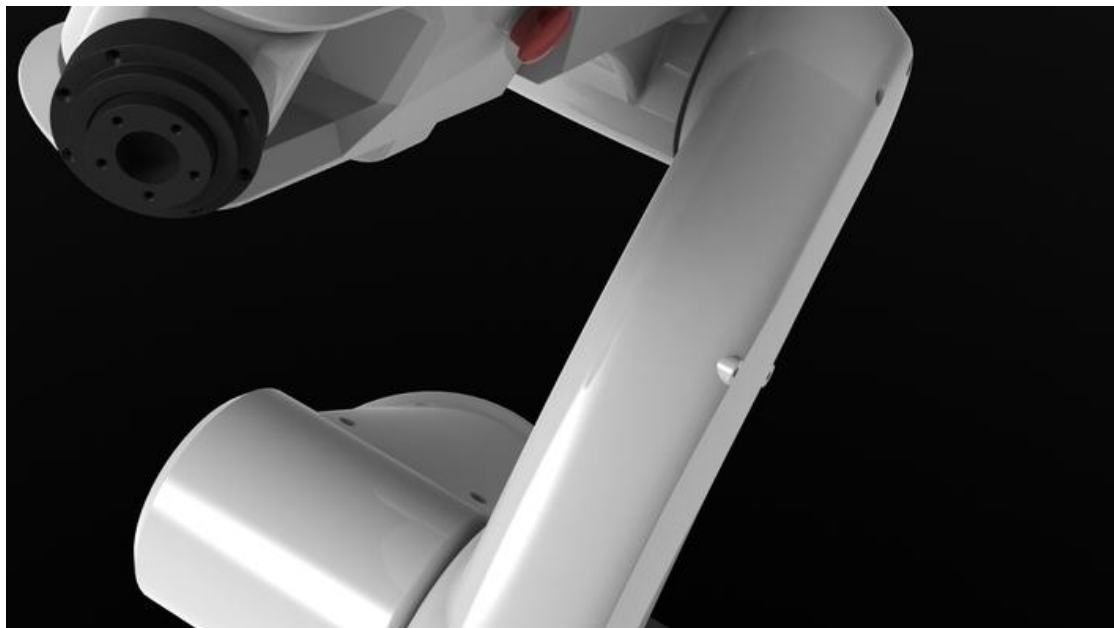
ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ Τ.Ε.

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Αριθμός 1473

ΤΙΤΛΟΣ ΤΗΣ ΠΤΥΧΙΑΚΗΣ

**Εικονικός Έλεγχος και Λειτουργία (Virtual
Commissioning) Βιομηχανικού Ρομποτικού
Βραχίονα μέσω PLC Εξομοιωτή**



ΣΠΟΥΔΑΣΤΗΣ: ΤΟΥΛΑΤΟΣ ΓΡΗΓΟΡΙΟΣ (5429)

**ΕΙΣΗΓΗΤΕΣ: ΚΑΛΑΝΤΖΟΠΟΥΛΟΣ ΑΘΑΝΑΣΙΟΣ
ΑΠΟΣΤΟΛΟΠΟΥΛΟΣ ΑΝΔΡΕΑΣ**

ΠΑΤΡΑ 2015

Πρόλογος

Σκοπός της παρούσας πτυχιακής εργασίας είναι ο εικονικός έλεγχος και λειτουργία (Virtual Commissioning) ενός βιομηχανικού ρομποτικού βραχίονα μέσω του αντίστοιχου τρισδιάστατου μοντέλου. Ο έλεγχος του εικονικού ρομποτικού βραχίονα επιτυγχάνεται μέσω PLC (Programmable Logic Controller), του οποίου η λειτουργία εξομοιώνεται με το λογισμικό S7-PLCSIM στην περίπτωση της μεθόδου SIL (Software In the Loop). Στην μέθοδο HIL (Hardware In the Loop) ο έλεγχος επιτυγχάνεται με πραγματικό PLC.

Αρχικά στο πρώτο κεφάλαιο γίνεται μια εισαγωγή στην διαδικασία VC (Virtual Commissioning). Όπου εξηγείται η διαφορά της με τη διαδικασία commissioning και παρουσιάζονται τα οφέλη, η δομή και τα είδη αυτής. Στο δεύτερο κεφάλαιο παρουσιάζεται ο βιομηχανικός ρομποτικός βραχίονας KR 5 sixx R850 της εταιρίας KUKA, ο οποίος μέσω του τρισδιάστατου μοντέλου του στο SolidWorks αποτελεί το σύστημα που εφαρμόζεται η διαδικασία VC. Επίσης, παρουσιάζονται και εξηγούνται οι τροποποιήσεις που χρειάστηκαν να γίνουν στο τρισδιάστατο μοντέλο για τις ανάγκες της παρούσας πτυχιακής εργασίας. Στη συνέχεια το τρίτο κεφάλαιο αναφέρεται στον έλεγχο του εικονικού ρομποτικού βραχίονα, που επιτυγχάνεται μέσω PLC. Παρουσιάζεται και εξηγείται ο κώδικας του PLC και το HMI (Human Machine Interface) για τον εποπτικό έλεγχο και την συλλογή δεδομένων της εφαρμογής. Το τέταρτο κεφάλαιο περιέχει τον τρόπο που εφαρμόζεται ο έλεγχος από το PLC στον εικονικό ρομποτικό βραχίονα χρησιμοποιώντας το LabVIEW. Παρουσιάζεται και εξηγείται ο γραφικός κώδικας του LabVIEW που προσομοιώνει τις λειτουργίες του υλικού που είναι υπεύθυνο για την κίνηση του ρομποτικού βραχίονα, καθώς και η παραμετροποίηση του OPC Server που είναι υπεύθυνος για την επικοινωνία του PLC με το LabVIEW. Τέλος, στο πέμπτο κεφάλαιο πραγματοποιείται η εκκίνηση της διαδικασίας VC και παρουσιάζονται τα αποτελέσματα και τα συμπεράσματα που προέκυψαν.

Σε αυτό το σημείο θα ήθελα να εκφράσω την εκτίμηση μου στον εισηγητή μου κ. Καλαντζόπουλο Αθανάσιο που μου έδωσε τη δυνατότητα να ασχοληθώ με ένα τόσο ενδιαφέρον θέμα. Θα ήθελα να τον ευχαριστήσω θερμά για την βοήθεια, το ενδιαφέρον αλλά και τον χρόνο που διέθεσε για την υλοποίηση της παρούσας πτυχιακής εργασίας. Επίσης, θα ήθελα να ευχαριστήσω τον εισηγητή μου κ. Αποστολόπουλο Ανδρέα για την πολύτιμη βοήθεια του. Στη συνέχεια θα ήθελα να ευχαριστήσω τους καθηγητές του τμήματος Ηλεκτρολόγων Μηχανικών Τ.Ε. για τις γνώσεις που μου μετέδωσαν όλα αυτά τα χρόνια.

Τέλος, ένα μεγάλο ευχαριστώ οφείλω στους γονείς μου και στα αγαπημένα μου πρόσωπα που ήταν πάντα δίπλα μου, και με βοήθησαν να υλοποιήσω τα όνειρά μου στηρίζοντάς με σε όλη τη διάρκεια των σπουδών μου.

Περίληψη

Ο σκοπός της παρούσας πτυχιακής εργασίας είναι ο εικονικός έλεγχος και λειτουργία ενός συστήματος μέσω του αντίστοιχου τρισδιάστατου μοντέλου. Αυτή η διαδικασία ονομάζεται virtual commissioning και επιτρέπει την επιβεβαίωση της μελέτης και του σχεδιασμού ενός συστήματος. Μέσω της παραπάνω διαδικασίας είναι εφικτή η ανάπτυξη και επικύρωση του κώδικα που εκτελεί ο εκάστοτε ελεγκτής. Επίσης επιτρέπει την μείωση του χρόνου ανάπτυξης ενός συστήματος με αποτέλεσμα την μείωση του συνολικού κόστους. Η διαδικασία του virtual commissioning μπορεί να αξιοποιηθεί για την επίδειξη των δυνατοτήτων του συστήματος και για την εκπαίδευση του προσωπικού.

Στην παρούσα πτυχιακή εργασία πραγματοποιείται ο εικονικός έλεγχος και λειτουργία ενός βιομηχανικού ρομποτικού βραχίονα μέσω ενός τρισδιάστατου μοντέλου που αναπαριστά τον ρομποτικού βραχίονα KR 5 sixx R850 της KUKA. Η εικονική λειτουργία και συμπεριφορά του συγκεκριμένου μοντέλου πραγματοποιείται μέσω του λογισμικού SolidWorks της Dassault Systemes.

Ο έλεγχος του ρομποτικού βραχίονα επιτυγχάνεται μέσω ενός PLC (Programmable Logic Controller) του οποίου η λειτουργία εξομοιώνεται με το λογισμικό SIMATIC S7-PLCSIM. Η υλοποίηση του κώδικα που εκτελεί το PLC πραγματοποιήθηκε μέσω του λογισμικού SIMATIC STEP 7. Ο εποπτικός έλεγχος του συστήματος από τον χρήστη πραγματοποιείται από ένα SCADA (Supervisory Control And Data Acquisition) περιβάλλον που υλοποιήθηκε με το λογισμικό SIMATIC WinCC. Τα λογισμικά πακέτα που αξιοποιήθηκαν για την ανάπτυξη της PLC εφαρμογής περιλαμβάνονται στην σουίτα TIA Portal της SIEMENS.

Η επικοινωνία των παραπάνω λογισμικών πραγματοποιήθηκε μέσω του λογισμικού LabVIEW της National Instruments. Επίσης το LabVIEW αξιοποιήθηκε και για την προσομοίωση των λειτουργιών του υλικού (hardware) που είναι υπεύθυνο για την λειτουργία του ρομποτικού βραχίονα.

Περιεχόμενα

Πρόλογος	iii
Περίληψη	v
Περιεχόμενα	vii
Κατάλογος ακρωνυμίων.....	ix
ΚΕΦΑΛΑΙΟ 1 - Εισαγωγή.....	1
1.1 Η διαδικασία του commissioning.....	1
1.2 Η διαδικασία και τα οφέλη του Virtual Commissioning	2
1.3 Η δομή και τα είδη του Virtual Commissioning	4
ΚΕΦΑΛΑΙΟ 2 - Ο εικονικός βραχίονας KUKA KR5 sixx R850	5
2.1 Εισαγωγή.....	5
2.2 Οι ρομποτικοί βραχίονες	6
2.3 Κατηγορίες ρομποτικών βραχιόνων.....	7
2.4 Ο βραχίονας KR 5 sixx R850.....	10
2.5 Το λογισμικό πακέτο SolidWorks.....	13
2.6 Υλοποίηση του εικονικού ρομποτικού βραχίονα.....	16
2.6.1 Τροποποίηση μοντέλου ρομποτικού βραχίονα	17
2.6.2 Τροποποίηση μοντέλου αρπάγης	21
2.6.3 Συναρμολόγηση	22
2.6.4 Εισαγωγή κινητήρων.....	26
ΚΕΦΑΛΑΙΟ 3 - Ο έλεγχος του εικονικού ρομποτικού βραχίονα	31
3.1 Εισαγωγή.....	31
3.2 Ο προγραμματιζόμενος λογικός ελεγκτής	31
3.2.1 Η δομή του SIMATIC S7-300 PLC	31
3.2.2 Αρχή λειτουργίας PLC	33
3.2.3 Γλώσσες προγραμματισμού PLC.....	34
3.2.4 Τύποι διαθέσιμων block	35
3.2.5 Δομή προγράμματος.....	36
3.3 Τα συστήματα SCADA	37
3.3.1 Τα συστήματα HMI.....	38
3.4 Το HMI της εφαρμογής.....	39
3.5 Ο PLC κώδικας της εφαρμογής	43
3.5.1 Block_auto (FC3).....	45
3.5.2 Block_routine_1 (FC4)	45
3.5.3 Block_move (FC1).....	49
3.5.4 Block_transport_position (FC9).....	50
3.5.5 Block_auto_default_parameters (FC10)	50
3.5.6 Block_manual (FC2)	51

3.5.7	Block_manual_parameters (FC11)	52
3.5.8	Block_reset (FC5)	52
3.5.9	Block_execute (FC6)	53
ΚΕΦΑΛΑΙΟ 4 - Η LabVIEW εφαρμογή για τον έλεγχο του τρισδιάστατου μοντέλου .		55
4.1	Το λογισμικό LabVIEW	55
4.1.1	Το SoftMotion Module	56
4.1.2	Το Datalogging and Supervisory Control Module	56
4.2	Η παραμετροποίηση του OPC Server	56
4.3	Το LabVIEW Project	64
4.4	Το πρόγραμμα του LabVIEW Project	68
ΚΕΦΑΛΑΙΟ 5 - Αποτελέσματα - Συμπεράσματα		73
5.1	Εισαγωγή	73
5.2	Ο έλεγχος της εφαρμογής με την μέθοδο SIL	73
5.2.1	Το λογισμικό NetToPLCSim	73
5.2.2	Η γενική δομή της εφαρμογής	74
5.2.3	Προετοιμασία του υπολογιστή που αντιπροσωπεύει το PLC	74
5.2.4	Προετοιμασία του δεύτερου υπολογιστή	77
5.2.5	Αποτελέσματα	80
5.3	Ο έλεγχος της εφαρμογής με την μέθοδο HIL	84
5.3.1	Προετοιμασία του υπολογιστή 3 και του PLC	85
5.3.2	Αποτελέσματα	86
5.4	Συμπεράσματα	89
Βιβλιογραφία		91

Κατάλογος ακρωνυμίων

2D	Two Dimensional	Δισδιάστατο
3D	Three Dimensional	Τρισδιάστατο
ALU	Arithmetic and Logic Unit	Αριθμητική και Λογική Μονάδα
CAD	Computer Aided Design	Σχεδίαση με τη Βοήθεια Υπολογιστή
CPU	Central Processing Unit	Κεντρική Μονάδα Επεξεργασίας
DB	Data Block	Μπλοκ Δεδομένων
DSC	Datalogging and Supervisory Control	Εποπτικός Έλεγχος και Καταγραφή Δεδομένων
FB	Function Block	Μπλοκ Συνάρτησης
FBD	Function Block Diagram	Μπλοκ Διάγραμμα Συνάρτησης
FC	Function	Συνάρτηση
G	Graphical Language	Γραφική Γλώσσα
HIL	Hardware In the Loop	Υλικό Στο Βρόχο
HMI	Human Machine Interface	Διεπαφή Ανθρώπου Μηχανής
LabVIEW	Laboratory Virtual Instrument Engineering Workbench	Εργαστηριακός Πάγκος Εικονικών Οργάνων
LAD	LADder logic	λογική Σκάλας
MMC	Micro Memory Card	Κάρτα Μνήμης Micro
OLE	Object Linking and Embedding	Σύνδεση και Ενσωμάτωση Αντικειμένων
OPC	OLE for Process Control	OLE για Έλεγχο Διεργασιών
PAC	Programmable Automation Controller	Προγραμματιζόμενος Ελεγκτής Αυτοματισμού
PII	Process Image Input	Μνήμη Απεικόνισης Εισόδων
PIQ	Process Image Output	Μνήμη Απεικόνισης Εξόδων
PLC	Programmable Logic Controller	Προγραμματιζόμενος Λογικός Ελεγκτής
PS	Power Supply	Μονάδα Τροφοδοσίας
RAM	Random Access Memory	Μνήμη Τυχαίας Προσπέλασης
RTU	Remote Terminal Unit	Απομακρυσμένη Τερματική Μονάδα
SCADA	Supervisory Control And Data Acquisition	Εποπτικός Έλεγχος και Συλλογή Δεδομένων
SDB	System Data Block	Μπλοκ Δεδομένων Συστήματος
SFB	System Function Block	Μπλοκ Συνάρτησης Συστήματος
SFC	System Function	Συνάρτηση Συστήματος
SIL	Software In the Loop	Λογισμικό Στον Βρόχο
STL	STatement List	Λίστα Εντολών
TCP/IP	Transmission Control Protocol/Internet Protocol	Πρωτόκολλο Ελέγχου Μετάδοσης / Πρωτόκολλο Διαδικτύου
VC	Virtual Commissioning	Εικονικός Έλεγχος και Λειτουργία
VI	Virtual Instrument	Εικονικό Όργανο

VPL	Visual Programming Language	Οπτική Γλώσσα Προγραμματισμού
NI	National Instruments	
OB	Organization Block	Μπλοκ Οργάνωσης

ΚΕΦΑΛΑΙΟ 1

Εισαγωγή

Σήμερα ο σχεδιασμός των προϊόντων και των συστημάτων παραγωγής λαμβάνει χώρα σε ένα βιομηχανικό περιβάλλον που χαρακτηρίζεται από σημαντικούς περιορισμούς του κόστους, συντόμευση του κύκλου ζωής του προϊόντος και στρατηγικές για την ταχεία διάθεση του στην αγορά. Για αυτούς τους λόγους, το χρονοδιάγραμμα παραγωγής μειώνεται σταδιακά ενώ οι απαιτήσεις για την ακρίβεια και την ποιότητα του σχεδιασμού αυξάνονται. Δεδομένου ότι ο κύκλος ζωής των προϊόντων μειώνεται σε μία συνεχώς μεταβαλλόμενη αγορά, τα σύγχρονα συστήματα παραγωγής θα πρέπει να μπορούν να προσαρμοστούν αποτελεσματικά σε ένα ευρύ φάσμα απαιτήσεων. Μια από τις κύριες προκλήσεις που αντιμετωπίζουν τα συστήματα αυτά είναι το κόστος με γνώμονα την ζήτηση για πιο γρήγορες και ασφαλείς ramp up διεργασίες. Ο όρος ramp up χρησιμοποιείται στα οικονομικά και τις επιχειρήσεις για να περιγράψει μια αύξηση στην παραγωγή εν όψει της αναμενόμενης αύξησης ζήτησης του προϊόντος. Εναλλακτικά, περιγράφει την περίοδο μεταξύ της ανάπτυξης των προϊόντων και της μέγιστης χρησιμοποίησης της παραγωγικής ικανότητας, η οποία χαρακτηρίζεται από το προϊόν, τον πειραματισμό και τις βελτιώσεις. Ωστόσο, ο συνεχώς αυξανόμενος αριθμός των ramp up διεργασιών, σχετίζεται με την ενισχυμένη καινοτομία, την αύξηση της εισαγωγής νέων προϊόντων στην αγορά και τις παραλλαγές αυτών.

Στην προσπάθεια να αντιμετωπιστούν τα παραπάνω προβλήματα, τις τελευταίες δεκαετίες αναπτύσσεται η ψηφιακή προσομοίωση της διαδικασίας της παραγωγής και των προϊόντων στη βιομηχανία και έχει αναδειχθεί ως μέσο για την επικύρωσή τους πριν από την κατασκευή τους. Η ανάπτυξη ξεκίνησε με την εισαγωγή του 2D (Two Dimensional) σχεδιασμού στον υπολογιστή. Λόγω των τεχνολογικών εξελίξεων στον τομέα της πληροφορικής μπορούν να επεξεργαστούν ψηφιακά πιο πολύπλοκες εργασίες σχετικά με τη διαδικασία ανάπτυξης του προϊόντος. Αυτό βασίζεται στο Digital Factory (Ψηφιακό Εργοστάσιο), σύμφωνα με το οποίο χρησιμοποιούνται από κοινού τεχνολογίες προσομοίωσης και συστήματα διαχείρισης δεδομένων για την βελτιστοποίηση της κατασκευής πριν από την έναρξη της παραγωγής, καθώς και για την στήριξη των ramp up διεργασιών. Το Digital Factory ορίστηκε ως γενικός όρος για ένα εκτεταμένο δίκτυο ψηφιακών μοντέλων και μεθόδων, συμπεριλαμβανομένης της προσομοίωσης και της 3D (Three Dimensional) απεικόνισης. Σκοπός του είναι ο ολοκληρωμένος σχεδιασμός, η υλοποίηση, ο έλεγχος και η συνεχής βελτίωση όλων των σημαντικών διεργασιών του εργοστασίου και των πόρων που σχετίζονται με το προϊόν.

1.1 Η διαδικασία του commissioning

Ένα από τα τμήματα του Digital Factory είναι το VC (Virtual Commissioning). Όπως είναι γνωστό, στους κλάδους της μηχανικής η διαδικασία του commissioning ενός (αυτοματοποιημένου) συστήματος είναι ένα σημαντικό στάδιο της ανάπτυξης του. Η οποία διασφαλίζει ότι το σύστημα και τα μέρη που το αποτελούν έχουν σχεδιαστεί, κατασκευαστεί, εγκατασταθεί και λειτουργούν σύμφωνα με τις ανάγκες και τις απαιτήσεις του χρήστη. Στην πράξη περιλαμβάνει την ολοκληρωμένη εφαρμογή ενός συνόλου διαδικασιών για τον έλεγχο και την δοκιμή κάθε συνιστώσας του έργου, από επιμέρους λειτουργίες μέχρι πολύπλοκα υποσυστήματα. Αυτή η διαδικασία μπορεί να εφαρμοστεί όχι μόνο σε νέα έργα αλλά και σε

συστήματα που υπόκεινται σε επέκταση ή αναβάθμιση. Ωστόσο, αυτή η φάση απαιτεί χρόνο και κόστος κάτι το οποίο μπορεί να βελτιωθεί με το VC.

1.2 Η διαδικασία και τα οφέλη του Virtual Commissioning

Το VC είναι η διαδικασία του εικονικού έλεγχου και λειτουργίας ενός συστήματος μέσω προσομοίωσης χρησιμοποιώντας το τρισδιάστατο μοντέλο του. Η διαφορά του με το commissioning είναι ότι δεν απαιτεί την ολοκλήρωση του υπό μελέτη συστήματος, αλλά γίνεται παράλληλα με την ανάπτυξή του. Αυτό επιτρέπει την επιβεβαίωση της μελέτης και του σχεδιασμού καθ' όλη την διάρκεια ανάπτυξης του συστήματος.

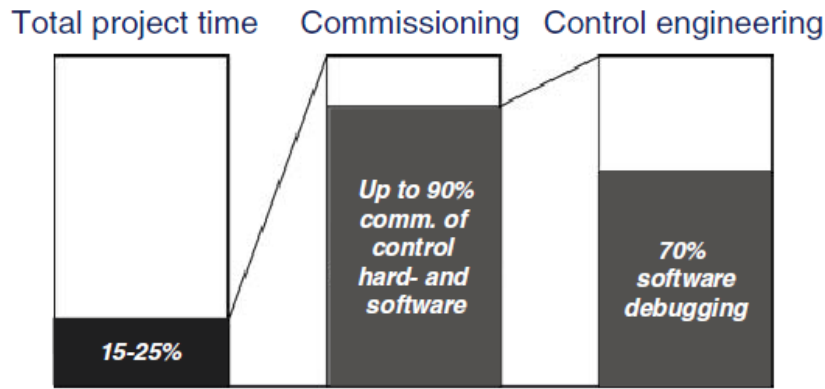
Πολλά από τα σφάλματα που εντοπίζονται κατά τη διάρκεια του commissioning προκύπτουν στις προηγούμενες φάσεις. Αυτό είναι ένα μεγάλο πρόβλημα μιας και ο εντοπισμός τους γίνεται αργά και το κόστος διόρθωσής τους μπορεί να είναι υψηλό, αλλά και η προσπάθεια που θα πρέπει να καταβληθεί πολύ μεγάλη (Σχήμα 1.1). Όταν ένα σφάλμα προέρχεται από προηγούμενη φάση, η επιβάρυνση και η επιπλέον εργασία στις φάσεις που ακολουθούν δεν μπορούν να αποφευχθούν. Για παράδειγμα, εάν οι διαστάσεις ενός εξαρτήματος ρομποτικού βραχίονα έχουν υπολογιστεί εσφαλμένα επηρεάζοντας την επιθυμητή λειτουργία του χρήστη και αυτό δεν έχει εντοπιστεί μέχρι την φάση του commissioning, τότε η διόρθωση αυτού του σφάλματος απαιτεί μεγάλη προσπάθεια με επιπλέον κόστος. Γιατί εκτός από την διόρθωση του συγκεκριμένου εξαρτήματος θα πρέπει να γίνουν τροποποιήσεις σε όσα τμήματα και υποσυστήματα σχετίζονται με αυτό ώστε να επιτευχθεί η επιθυμητή λειτουργία του ρομποτικού βραχίονα.

Με το VC υπάρχει η δυνατότητα να γίνεται ο έλεγχος και η δοκιμή παράλληλα με την εξέλιξη του έργου, άρα και τον άμεσο εντοπισμό και διόρθωση των όποιων σφαλμάτων μπορεί να προκύψουν. Επομένως, επιτυγχάνεται η μείωση του κόστους ανάπτυξης του συστήματος.



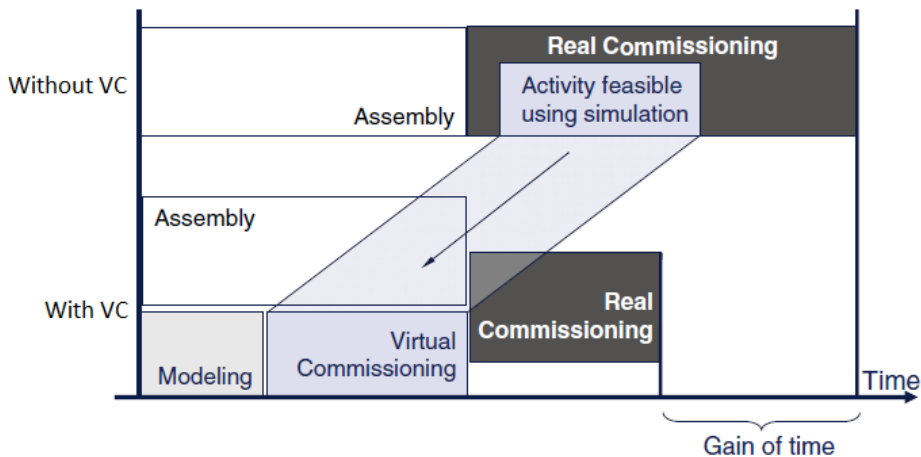
Σχήμα 1.1: Κόστος διόρθωσης σφαλμάτων, ανάλογα με την χρονική στιγμή εντοπισμού τους.

Μια έρευνα για την Γερμανική Ένωση Κατασκευαστών Εργαλειομηχανών (VDW-Verein Deutscher Werkzeugmaschinenfabriken) έδειξε ότι η διαδικασία του commissioning καταλαμβάνει έως και το ένα τέταρτο της συνολικής διάρκειας του έργου (Σχήμα 1.2), εκ του οποίου το 90% χρησιμοποιείται για δραστηριότητες που σχετίζονται με το ηλεκτρολογικό κομμάτι και τα συστήματα ελέγχου π.χ. PLC (Programmable Logic Controller). Εντός του τελευταίου ποσοστού το 70% του χρόνου καταναλώνεται για τον εντοπισμό και την διόρθωση σφαλμάτων (debugging) του κώδικα προγραμματισμού.



Σχήμα 1.2: Ποσοστό του χρόνου που καταναλώνεται σε ένα έργο για debugging.

Η ανάπτυξη, η δοκιμή και η διόρθωση σφαλμάτων του κώδικα του PLC είναι συνήθως χρονοβόρα και μπορεί να προκαλέσει δυσλειτουργία ή σφάλματα στο πραγματικό σύστημα. Δεδομένου ότι η διαδικασία του commissioning διεξάγεται κάτω από ακραία πίεση χρόνου τότε μία ενδεχόμενη καθυστέρηση του έργου υπολογίζεται σαν οικονομική ζημία. Η ανάπτυξη και επικύρωση του κώδικα που εκτελεί ο εκάστοτε ελεγκτής είναι εφικτή σε προηγούμενες φάσεις του έργου αξιοποιώντας την διαδικασία του VC χωρίς να είναι απαραίτητη η χρησιμοποίηση του πραγματικού συστήματος. Το γεγονός αυτό έχει ως αποτέλεσμα την μείωση της χρονικής διάρκειας της διαδικασίας του commissioning, άρα και την μείωση του χρόνου ανάπτυξης του έργου (Σχήμα 1.3).



Σχήμα 1.3: Χρονικό όφελος με VC.

Τέλος, ένα ακόμα σημαντικό όφελος του VC είναι ότι οι χρήστες έχουν τη δυνατότητα να εκπαιδεύονται πάνω στο εικονικό σύστημα με ασφάλεια, συνεχώς και παράλληλα με την ανάπτυξη του. Αυτό έχει ως αποτέλεσμα με την ολοκλήρωση του έργου να μην απαιτείται επιπλέον χρόνος εκπαίδευσης για τους χρήστες καθώς και μείωση του ρίσκου για ενδεχόμενες βλάβες (σωματικές, υλικές). Επομένως, για ακόμη μια φορά υπάρχει μείωση του κόστους.

1.3 Η δομή και τα είδη του Virtual Commissioning

Για τον σχεδιασμό και την ανάπτυξη ενός VC Project απαιτούνται πακέτα λογισμικών που να καλύπτουν τα τρία διακριτά αλλά αλληλένδετα τμήματα του Project:

- Τη μηχανική σχεδίαση, συμπεριλαμβανομένων των ενεργοποιητών (actuators), των αισθητήρων (sensors) που σχετίζονται με την συμπεριφορά του συστήματος (λειτουργικό μοντέλο).
- Τον έλεγχο του συστήματος, συμπεριλαμβανομένου του ελεγκτή και των σημάτων εισόδων/εξόδων.
- Την διασύνδεση μεταξύ των αισθητήρων, των ενεργοποιητών και του ελεγκτή.

Για τον έλεγχο του λειτουργικού μοντέλου ενός VC Project υπάρχουν δύο μέθοδοι προσέγγισης. Η μέθοδος SIL (Software In the Loop), όπου ο έλεγχος του λειτουργικού μοντέλου επιτυγχάνεται από εικονικό ελεγκτή (λογισμικό εξομοίωσης ελεγκτή), και η μέθοδος HIL (Hardware In the Loop) κατά την οποία ελέγχεται από πραγματικό ελεγκτή. Το κύριο πλεονέκτημα της μεθόδου SIL είναι ότι δεν απαιτείται υλικό (hardware) για την εφαρμογή της διαδικασίας VC. Για τον λόγο αυτό η εφαρμογή της διαδικασίας VC μπορεί να γίνει οπουδήποτε, αρκεί να υπάρχει τουλάχιστον ένας υπολογιστής. Το μειονέκτημα της μεθόδου SIL είναι η χαμηλή διαθεσιμότητα ενημερωμένων λογισμικών εξομοίωσης ελεγκτή. Το πλεονέκτημα της μεθόδου HIL όπου χρησιμοποιείται πραγματικός ελεγκτής είναι η δυνατότητα εκτέλεσης πολύπλοκων σεναρίων ελέγχου και αυτοματισμού σε πραγματικό χρόνο και υπό πραγματικές συνθήκες ελέγχου. Το μειονέκτημα είναι ότι με την μέθοδο αυτή η διαδικασία VC μπορεί να εφαρμοστεί μόνο στο χώρο εργασίας μιας και απαιτείται το αντίστοιχο υλικό (hardware).

ΚΕΦΑΛΑΙΟ 2

Ο εικονικός βραχίονας KUKA KR5 sixx R850

2.1 Εισαγωγή

Η λέξη ρομπότ προέρχεται από την Τσέχικη λέξη *robota* που σημαίνει εργασία, η οποία επινοήθηκε από τον Τσέχο συγγραφέα και ζωγράφο Josef Čapek. Καθιερώθηκε ως όρος με την σημερινή του έννοια το 1920 από τον αδερφό του και θεατρικό συγγραφέα Karel Čapek στο έργο του R.U.R. (Rossum's Universal Robots), όπου σατιρίζει την εξάρτηση της κοινωνίας από τους μηχανικούς εργάτες (ρομπότ) της τεχνολογικής εξέλιξης και που τελικά εξοντώνουν τους δημιουργούς τους.

Η Ρομποτική είναι εκείνος ο κλάδος της επιστήμης της μηχανικής που ασχολείται με τη σύλληψη, το σχεδιασμό, την κατασκευή και τη λειτουργία ρομπότ. Τα ρομπότ είναι μηχανές, η χρήση των οποίων αποσκοπεί στην αντικατάσταση του ανθρώπου στην εκτέλεση έργου. Ένα ρομπότ μπορεί να δράσει κάτω από τον απευθείας έλεγχο ενός ανθρώπου ή αυτόνομα κάτω από τον έλεγχο ενός προγραμματιζόμενου ελεγκτή.

Σύμφωνα με το Robot Institute of America, ο όρος ρομπότ αναφέρεται σε ένα μηχανισμό που έχει σχεδιαστεί ώστε μέσω προγραμματιζόμενων κινήσεων να μεταφέρει υλικά, αντικείμενα, εργαλεία ή εξειδικευμένες συσκευές με σκοπό την εκτέλεση ενός συνόλου εργασιών.

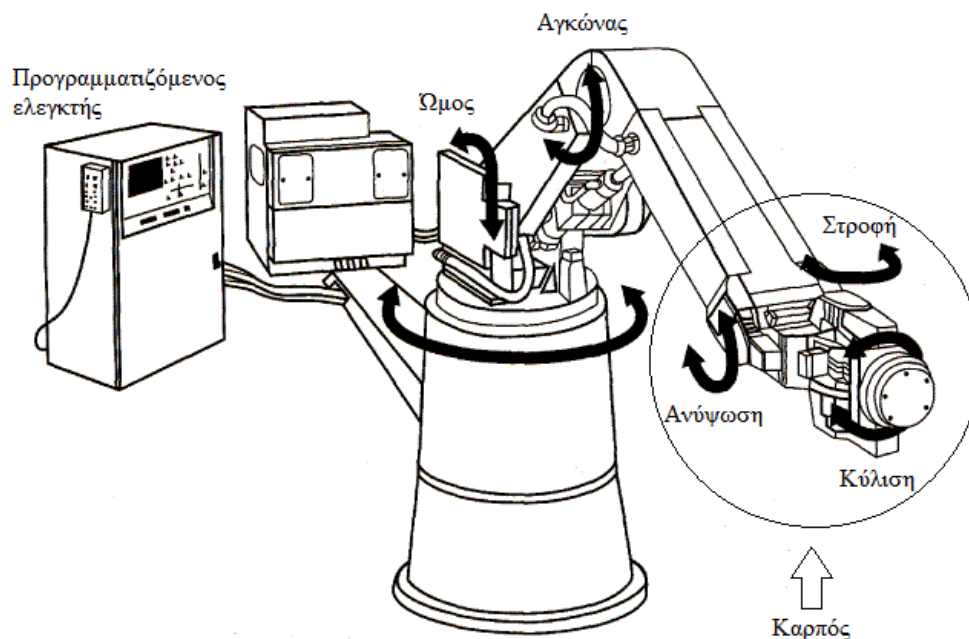
Ένας τέτοιος μηχανισμός συνήθως αποτελείται από τα παρακάτω υποσυστήματα:

- Το μηχανολογικό υποσύστημα που είναι υπεύθυνο για την εκτέλεση του έργου. Το υποσύστημα αυτό αποτελείται από μηχανισμούς που επιτρέπουν στο ρομπότ να κινείται, όπως αρθρώσεις, σύστημα μετάδοσης κίνησης, ενεργοποιητές-κινητήρες, οδηγούς κλπ.
- Το υποσύστημα αίσθησης που είναι υπεύθυνο για την συγκέντρωση πληροφοριών οι οποίες σχετίζονται τόσο με την κατάσταση του ρομπότ όσο και με την κατάσταση του περιβάλλοντος. Στο υποσύστημα αίσθησης περιλαμβάνονται όργανα μετρήσεων, αισθητήρες, ηλεκτρονικά στοιχεία κλπ.
- Το υποσύστημα ελέγχου συνδυάζει κατάλληλα την αίσθηση με τη δράση, έτσι ώστε το ρομπότ να λειτουργεί αποτελεσματικά και με τον επιθυμητό τρόπο. Ο ελεγκτής που διαθέτει το συγκεκριμένο υποσύστημα, αναλαμβάνει την επίβλεψη και τον συντονισμό ολόκληρου του συστήματος που αποτελεί το ρομπότ.

Τα ρομπότ μπορούν να χρησιμοποιηθούν ώστε να εκτελούν εργασίες που είναι δύσκολες ή/και επικίνδυνες για τον άνθρωπο. Σε πολλές περιπτώσεις, αξιοποιούνται για την μείωση του κόστους και του χρόνου παραγωγής ενός προϊόντος. Για τον λόγο αυτό τα ρομπότ χρησιμοποιούνται κατά κύριο λόγο στην βιομηχανία. Ένα ακόμα σημαντικό πλεονέκτημα των βιομηχανικών ρομπότ είναι η ευελιξία τους. Το γεγονός αυτό επιτρέπει στα βιομηχανικά ρομπότ να προσαρμόζονται στις εκάστοτε απαιτήσεις επιτρέποντας την παραγωγή διαφορετικών προϊόντων από την ίδια γραμμή παραγωγής. Επίσης, με κατάλληλο επαναπρογραμματισμό έχουν την δυνατότητα να ανταπεξέρχονται στις μικρές ή μεγάλες μεταβολές του παραγόμενου προϊόντος σύμφωνα με τις ανάγκες της αγοράς. Εκτός από την βιομηχανία τα ρομπότ έχουν πολλές εφαρμογές στην ιατρική, σε διαστημικά προγράμματα, σε στρατιωτικά προγράμματα κλπ.

2.2 Οι ρομποτικοί βραχίονες

Ο ρομποτικός βραχίονας (Σχήμα 2.1) είναι ένας ηλεκτρο-μηχανικός μηχανισμός που μιμείται τις κινήσεις του ανθρώπινου χεριού. Αποτελείται από τους συνδέσμους (links) και τις αρθρώσεις (joints). Σε αντιστοιχία με το ανθρώπινο χέρι, οι σύνδεσμοι αντιστοιχούν στα οστά (κλείδα-ωμοπλάτη, βραχιόνιο, ωλένη-κερκίδα) και οι αρθρώσεις στις ανθρώπινες αρθρώσεις που ενώνουν τα οστά (άρθρωση ώμου, αγκώνα, καρπού). Μια από τις πλέον σημαντικές αρθρώσεις είναι η άρθρωση του καρπού η οποία αποτελείται από ένα σύνολο μηχανικών αρθρώσεων και επιτρέπει την περιστροφή γύρω από τρεις άξονες, τον διαμήκη, που έχει τη διεύθυνση των δακτύλων (κίνηση κύλισης-roll), τον εγκάρσιο, που βρίσκεται στο επίπεδο της παλάμης (κίνηση ανύψωσης-pitch) και είναι κάθετος ως προς τα δάκτυλα, και τέλος τον κάθετο άξονα, που είναι κάθετος ως προς τους άλλους δύο (κίνηση στροφής-yaw).



Σχήμα 2.1: Τυπική μορφή ρομποτικού βραχίονα (Cincinnati Milacron T3).

Για την υλοποίηση λοιπόν των αρθρώσεων ενός ρομποτικού βραχίονα αξιοποιούνται διάφορα είδη μηχανικών αρθρώσεων. Τα βασικότερα είδη μηχανικών αρθρώσεων είναι οι περιστροφικές, οι πρισματικές ή γραμμικές και οι σφαιρικές. Οι περιστροφικές αρθρώσεις υλοποιούν σχετική περιστροφική κίνηση μεταξύ δύο διαδοχικών συνδέσμων, οι πρισματικές αρθρώσεις υλοποιούν σχετική μεταφορική κίνηση μεταξύ δύο διαδοχικών συνδέσμων και οι σφαιρικές αρθρώσεις, σφαιρική περιστροφική κίνηση μεταξύ δύο διαδοχικών συνδέσμων.

Αυτό που χαρακτηρίζει κυρίως έναν ρομποτικό βραχίονα είναι η ελευθερία κινήσεων που έχει, δηλαδή οι βαθμοί ελευθερίας (Degrees Of Freedom-DOF). Για παράδειγμα ένας βραχίονας που κινείται σε ένα επίπεδο έχει λιγότερες δυνατότητες από έναν που κινείται στον τρισδιάστατο χώρο. Συνήθως στο τέλος του καρπού του βραχίονα τοποθετείται ένα εργαλείο με το οποίο θα εκτελεστεί μια εργασία (τελικό στοιχείο δράσης-end effector). Όταν αυτό μπορεί να τοποθετηθεί σε οποιοδήποτε σημείο του χώρου και με οποιαδήποτε διεύθυνση, τότε λέμε ότι έχει έξι βαθμούς ελευθερίας, τρεις για την θέση του στο χώρο (XYZ) και τρεις για τον προσανατολισμό του εργαλείου. Όταν κινείται στο επίπεδο έχει τρεις βαθμούς ελευθερίας, δύο για το επίπεδο (XY) και μια για τον προσανατολισμό του.

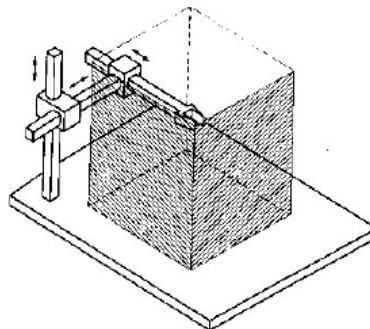
Από τα πιο σημαντικά μεγέθη ενός βιομηχανικού ρομποτικού βραχίονα είναι το ωφέλιμο φορτίο, η επαναληψιμότητα, η ακρίβεια και ο χώρος εργασίας. Πιο συγκεκριμένα τα παραπάνω μεγέθη αναφέρονται στα εξής:

- **Ωφέλιμο φορτίο:** είναι το βάρος που μπορεί να μεταφέρει το άκρο του βραχίονα. Ως σημείο εφαρμογής του βάρους θεωρείται το τελείωμα της άρθρωσης του καρπού. Το προδιαγραφόμενο αυτό φορτίο δεν είναι σταθερό και εξαρτάται από την ταχύτητα με την οποία πρόκειται να κινηθεί η άρθρωση του καρπού.
- **Επαναληψιμότητα:** εκφράζει τη δυνατότητα του βραχίονα να γυρίσει στο ίδιο σημείο μετά από αρκετές επαναλήψεις και δίνεται ως εύρος μέσα στο οποίο ο βραχίονας θα τερματίσει την κίνηση. Η απόκλιση οφείλεται στο ότι κατά τη λειτουργία του είναι δυνατό να χάσει λίγο από τη μέτρηση της θέσης με αποτέλεσμα να μη μπορεί να επιστρέψει στη συγκεκριμένη θέση μετά από ορισμένους κύκλους λειτουργίας.
- **Ακρίβεια:** είναι η ικανότητα του ρομποτικού βραχίονα να πηγαίνει ακριβώς στην προδιαγεγραμμένη θέση. Η ακρίβεια εξαρτάται κυρίως από τη διακριτότητα των εξαρτημάτων ελέγχου, τη μηχανολογική σύνδεση των μελών του και το ελάχιστο επιτρεπόμενο σφάλμα που επιβάλλει η ευστάθεια της λειτουργίας μηχανισμών κίνησης. Η ακρίβεια επηρεάζεται από το είδος και το μέγεθος του εκάστοτε φορτίου, σε αντίθεση με την επαναληψιμότητα, για αυτό και ορισμένοι κατασκευαστές προδιαγράφουν μόνο την τελευταία.
- **Χώρος εργασίας:** ως χώρος εργασίας ορίζεται ο τρισδιάστατος χώρος τον οποίο μπορεί να σαρώσει η άκρη του ρομποτικού βραχίονα. Το μέγεθος και η γεωμετρική μορφή του χώρου αυτού εξαρτώνται από την κατασκευαστική δομή του βραχίονα.

2.3 Κατηγορίες ρομποτικών βραχιόνων

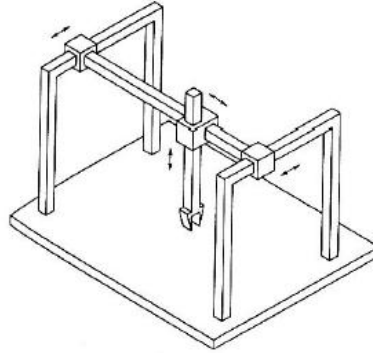
Ο τύπος και η διαδοχή των αρθρώσεων επιτρέπει την ταξινόμησή των ρομποτικών βραχιόνων σε διάφορες κατηγορίες, οι οποίες αναφέρονται παρακάτω. Για την κατηγοριοποίηση λαμβάνονται υπόψη οι τρεις πρώτες αρθρώσεις του βραχίονα και κατά συνέπεια εξαιρούνται οι αρθρώσεις του καρπού.

- **Καρτεσιανοί βραχίονες:** η καρτεσιανή γεωμετρία υλοποιείται με τρεις διαδοχικές πρισματικές αρθρώσεις. Οι άξονες των αρθρώσεων αυτών είναι ανά δύο κάθετοι μεταξύ τους (Σχήμα 2.2). Η καρτεσιανή δομή παρέχει μεγάλη δυσκαμψία και σταθερή ακρίβεια σε ολόκληρο το χώρο εργασίας που είναι ένα παραλληλεπίπεδο. Βασικό μειονέκτημα της κατασκευής είναι η μειωμένη επιδεξιότητα κίνησης, λόγω της πρισματικής φύσης των αρθρώσεων.



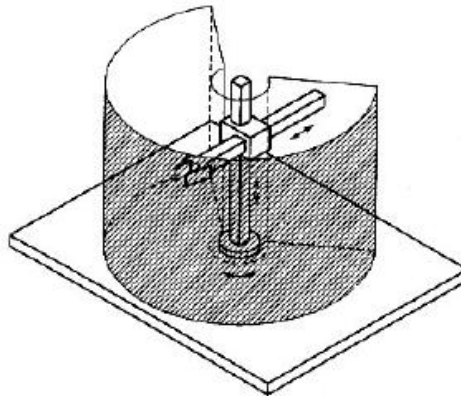
Σχήμα 2.2: Καρτεσιανός βραχίονας.

- **Βραχίονες Gantry:** οι βραχίονες Gantry είναι στην ουσία καρτεσιανοί, διαφέρουν όμως από τους τελευταίους στον τρόπο προσέγγισης του αντικειμένου (Σχήμα 2.3). Ο βραχίονας Gantry προσεγγίζει το αντικείμενο από πάνω, ενώ ένας κλασικός καρτεσιανός βραχίονας προσεγγίζει το αντικείμενο από το πλάι. Άμεσες συνέπειες της διαφοροποίησης αυτής είναι η αύξηση του χώρου εργασίας και της δυσκαμψίας, καθώς επίσης και η δυνατότητα χειρισμού μεγάλων και βαριών αντικειμένων.



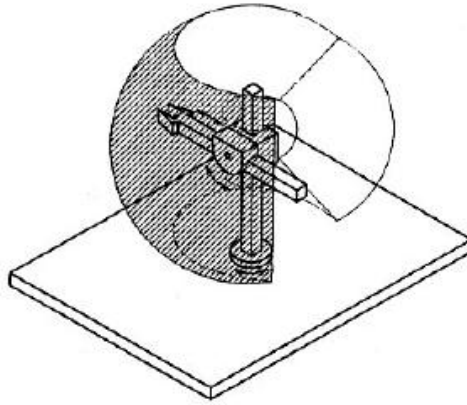
Σχήμα 2.3: Βραχίονας Gantry.

- **Κυλινδρικοί βραχίονες:** στους κυλινδρικούς βραχίονες η πρώτη πρισματική άρθρωση της καρτεσιανής δομής έχει αντικατασταθεί από μία περιστροφική άρθρωση (Σχήμα 2.4). Οι συγκεκριμένοι βραχίονες χαρακτηρίζονται από καλή δυσκαμψία, όμως η ακρίβεια της θέσης του καρπού μειώνεται καθώς η οριζόντια μετατόπιση αυξάνεται. Ο χώρος εργασίας στην περίπτωση αυτή είναι τμήμα κυλίνδρου. Σημαντικό μειονέκτημα της συγκεκριμένης γεωμετρίας είναι το ότι ο βραχίονας εισέρχεται στο χώρο εργασίας και τον περιορίζει.



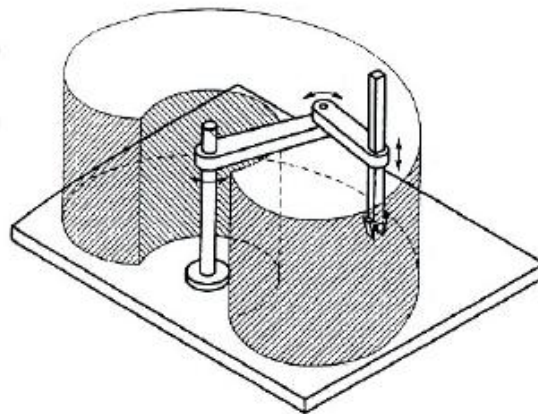
Σχήμα 2.4: Κυλινδρικός βραχίονας.

- **Σφαιρικοί βραχίονες:** στους βραχίονες αυτούς αντικαθίσταται πλέον και η δεύτερη πρισματική άρθρωση της καρτεσιανής δομής με περιστροφική (Σχήμα 2.5). Η μηχανολογική πολυπλοκότητα αυξάνει, ενώ η δυσκαμψία μειώνεται. Επιπλέον, η ακρίβεια του καρπού μειώνεται με την αύξηση της ακτινικής απόστασης. Ο χώρος εργασίας είναι τμήμα σφαίρας και περιέχει ένα μέρος της βάσης με άμεση συνέπεια τη δυνατότητα χειρισμού αντικειμένων που βρίσκονται στο έδαφος.



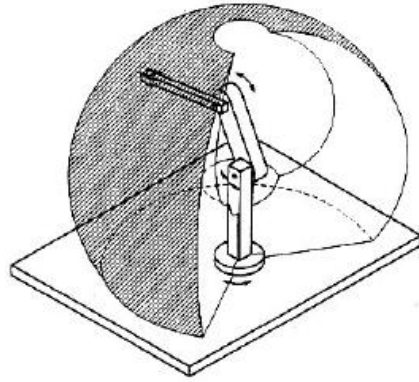
Σχήμα 2.5: Σφαιρικός βραχίονας.

- Βραχίονες SCARA:** η γεωμετρία SCARA (Selective Compliance Assembly Robot Arm) είναι ειδική και περιλαμβάνει δυο περιστροφικές και μια πρισματική άρθρωση τοποθετημένες κατά τέτοιο τρόπο ώστε οι άξονες κίνησης να είναι παράλληλοι μεταξύ τους (Σχήμα 2.6). Η συγκεκριμένη γεωμετρία παρέχει μεγάλη δυσκαμψία σε κατακόρυφη φόρτιση και ελαστικότητα σε οριζόντια. Η ακρίβεια τοποθέτησης του καρπού μειώνεται με την αύξηση της απόστασής του από τον άξονα της πρώτης άρθρωσης.



Σχήμα 2.6: Βραχίονας SCARA.

- Ανθρωπομορφικοί βραχίονες:** η ανθρωπομορφική γεωμετρία υλοποιείται με τρεις διαδοχικές περιστροφικές αρθρώσεις. Ειδικότερα, ο άξονας περιστροφής της πρώτης άρθρωσης είναι κατακόρυφος και κάθετος στους άξονες περιστροφής των επόμενων δύο αρθρώσεων, οι οποίοι είναι παράλληλοι μεταξύ τους (Σχήμα 2.7). Η συγκεκριμένη δομή παρέχει τη μεγαλύτερη επιδεξιότητα από όλες τις προηγούμενες, καθώς όλες οι αρθρώσεις είναι περιστροφικές. Ωστόσο, η ακρίβεια του καρπού δεν είναι σταθερή εντός του χώρου εργασίας που έχει τη μορφή σφαίρας.



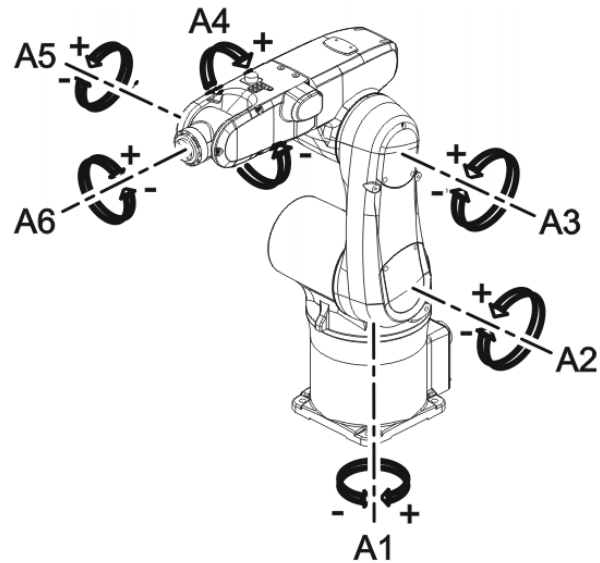
Σχήμα 2.7: Ανθρωπομορφικός βραχίονας.

2.4 Ο βραχίονας KR 5 sixx R850

Ο βιομηχανικός ρομποτικός βραχίονας KR 5 sixx R850 της Γερμανικής εταιρείας KUKA είναι ένας βραχίονας με έξι περιστροφικές αρθρώσεις και έξι βαθμούς ελευθερίας (Σχήμα 2.8). Το βάρος του είναι περίπου 29 kg, σε πλήρη έκταση και χωρίς κάποιο εξάρτημα προσαρμοσμένο, φτάνει σε μήκος τα 855 mm και σε ύψος τα 1115 mm. Μπορεί να χειριστεί αντικείμενα μέγιστου βάρους 5 kg με μεγάλη ευελιξία και επαναληψιμότητα $\pm 0,03$ mm. Στον Πίνακα 2.1 φαίνονται τα τεχνικά χαρακτηριστικά του ρομποτικού βραχίονα.



(α)



(β)

Σχήμα 2.8: (α) Ο ρομποτικός βραχίονας KUKA KR 5 sixx R850, και (β) οι άξονες του ρομποτικού βραχίονα.

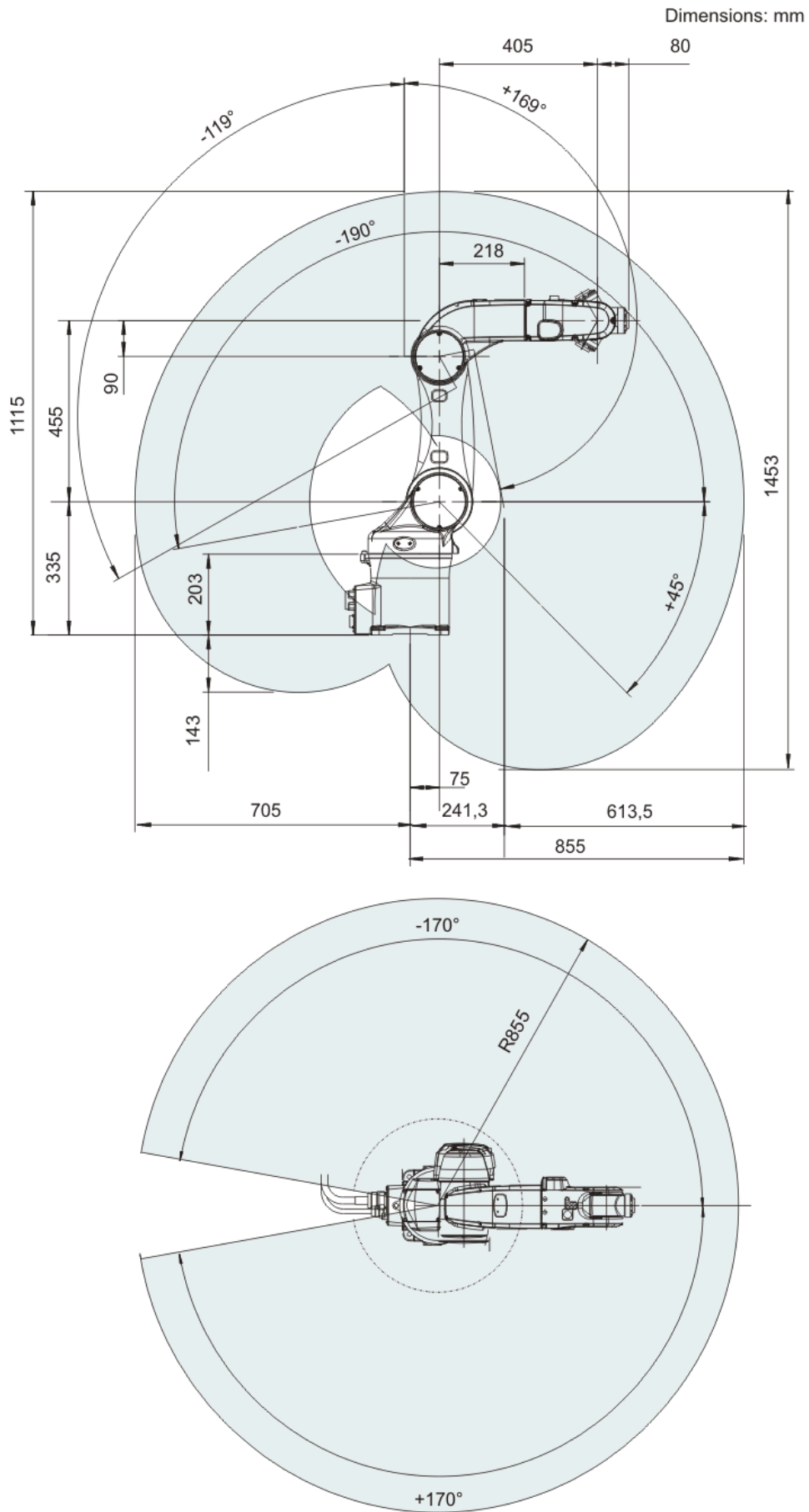
Τύπος	KR 5 sixx R850
Άξονες	6
Όγκος του χώρου εργασίας	2,3 m ³
Επαναληψιμότητα (ISO 9283)	±0,03 mm
Βάρος (περίπου)	29 kg
Ωφέλιμο φορτίο	5 kg
Βαθμός προστασίας ρομπότ	IP 40
Βαθμός προστασίας καρπού	IP 65
Επίπεδο θορύβου	< 75 dB (A) έξω από τον χώρο εργασίας

Πίνακας 2.1: Τεχνικά χαρακτηριστικά ρομποτικού βραχίονα.

Οι αρθρώσεις του έχουν ένα όριο περιστροφής, το οποίο είναι διαφορετικό για την κάθε μία. Επίσης, υπάρχουν περιορισμοί στην γωνιακή ταχύτητα που μπορεί να επιτύχει η κάθε άρθρωση. Όπως είναι φυσικό οι αρθρώσεις του καρπού είναι πιο γρήγορες από αυτές του υπόλοιπου βραχίονα, γιατί είναι πιο μικρές και έχουν μικρότερο φορτίο. Ο Πίνακας 2.2 περιέχει τα όρια περιστροφής και την μέγιστη γωνιακή ταχύτητα για την κάθε άρθρωση. Στο Σχήμα 2.9 φαίνεται ο χώρος εργασίας του συγκεκριμένου ρομποτικού βραχίονα.

Άξονας	Εύρος κίνησης (όριο περιστροφής)	Γωνιακή ταχύτητα με ωφέλιμο φορτίο 5 kg
1	±170°	250°/s
2	+45° έως -190°	250°/s
3	+165° έως -119°	250°/s
4	±190°	410°/s
5	±120°	410°/s
6	±358°	660°/s

Πίνακας 2.2: Όρια περιστροφής και μέγιστη γωνιακή ταχύτητα για κάθε άρθρωση.



Σχήμα 2.9: Χώρος εργασίας ρομποτικού βραχίονα.

2.5 Το λογισμικό πακέτο SolidWorks

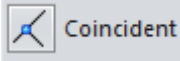
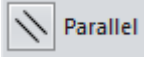
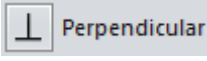
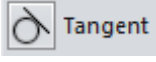
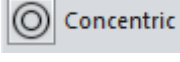

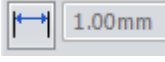
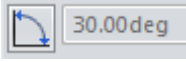
Για τις ανάγκες τις παρούσας πτυχιακής εργασίας επιλέχτηκε η αξιοποίηση του ρομποτικού βραχίονα KUKA KR 5 sixx R850 μέσω ενός κατάλληλα διαμορφωμένου τρισδιάστατου μοντέλου. Το τρισδιάστατο μοντέλο του KUKA KR 5 sixx R850 έχει υλοποιηθεί με το λογισμικό πακέτο SolidWorks. Το συγκεκριμένο λογισμικό πακέτο είναι επίσης υπεύθυνο για την απεικόνιση του τρισδιάστατου μοντέλου και την προσομοίωση τόσο της λειτουργίας όσο και της συμπεριφοράς του επιλεγμένου ρομποτικού βραχίονα.

Το λογισμικό πακέτο SolidWorks είναι ένα ολοκληρωμένο σχεδιαστικό πακέτο το οποίο βρίσκεται ευρεία εφαρμογή στην σύγχρονη βιομηχανία. Συγκεκριμένα το SolidWorks αποτελεί ένα πακέτο 3D CAD (Computer-Aided Design) μοντελοποίησης, το οποίο έχει αναπτυχθεί από την εταιρία Dassault Systèmes SolidWorks Corp. Το SolidWorks χρησιμοποιείται σήμερα από περισσότερους από 1,3 εκ. μηχανικούς και σχεδιαστές και περισσότερες από 130.000 εταιρείες ανά τον κόσμο. Ιστορικά το SolidWorks εμφανίστηκε στη διεθνή αγορά το Δεκέμβριο του 1993 από τον Jon Hirschtick, ο οποίος με έδρα το Concord, Μασαχουσέτη, ΗΠΑ προσέλαβε μία ομάδα μηχανικών για την δημιουργία μιας εταιρείας, η οποία θα ανέπτυξε ένα 3D CAD λογισμικό το οποίο θα ήταν εύκολο στη χρήση, προσιτό και ικανό να λειτουργήσει σε έναν συμβατικό προσωπικό υπολογιστή. Το 1997 η Dassault Systèmes, εταιρία γνωστή στο χώρο των λογισμικών μοντελοποίησης για το CATIA CAD λογισμικό της, απέκτησε την εταιρία και σήμερα κατέχει το 100% των μετοχών της.

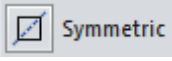
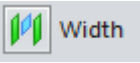
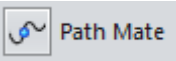
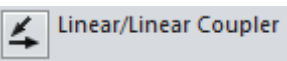
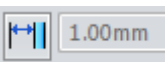
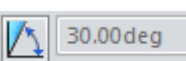
Επίσης, στα πλαίσια της παρούσας πτυχιακής εργασίας χρησιμοποιείται ένα από τα πολλά Add-Ins (πρόσθετα) του σχεδιαστικού πακέτου, το SolidWorks Motion, το οποίο αποτελεί ένα εικονικό εργαλείο πρωτοτυποποίησης. Το SolidWorks Motion παρέχει δυνατότητες προσομοίωσης της κίνησης ενός μηχανισμού, ελέγχοντας με τον τρόπο αυτόν την επιθυμητή λειτουργία του. Παρέχει ολοκληρωμένες ποσοτικές πληροφορίες σχετικά με την κινηματική, συμπεριλαμβανομένης της θέσης, της ταχύτητας και της επιτάχυνσης. Καθώς και πληροφορίες για την δυναμική, συμπεριλαμβανομένων των αντιδράσεων των αρθρώσεων, τις δυνάμεις αδράνειας και τις απαιτήσεις ισχύος όλων των τμημάτων ενός κινούμενου μηχανισμού.

Το SolidWorks μπορεί να αξιοποιηθεί για την 3D σχεδίαση ενός ενιαίου εξαρτήματος (part), την 2D σχεδίαση (drawing) ενός εξαρτήματος ή ενός συναρμολογημένου συγκροτήματος εξαρτημάτων, και για την 3D συναρμολόγηση (assembly) εξαρτημάτων και/ή άλλων συναρμολογημένων συγκροτημάτων.

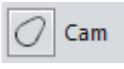
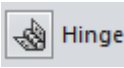

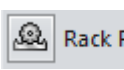
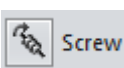

Τα assemblies (συναρμολογημένα συγκροτήματα εξαρτημάτων) είναι πολύπλοκες κατασκευές που αποτελούνται από πολλά επιμέρους στοιχεία, τα οποία μπορεί να είναι εξαρτήματα και/ή άλλα συναρμολογημένα συγκροτήματα, που ονομάζονται subassemblies (υποσυγκροτήματα). Για να γίνει μια συναρμολόγηση χρειάζονται τα mates. Τα mates δημιουργούν γεωμετρικές σχέσεις μεταξύ των εξαρτημάτων και/ή των υποσυγκροτημάτων σε μία συναρμολόγηση. Επίσης, με την προσθήκη mates, ορίζονται οι επιθυμητές κινήσεις γραμμικές ή περιστροφικές με τα αντίστοιχα όρια και τις κατευθύνσεις τους μεταξύ των εξαρτημάτων. Τα είδη των mates χωρίζονται σε τρεις κατηγορίες, τα standard mates (Πίνακας 2.3), τα advanced mates (Πίνακας 2.4) και τα mechanical mates (Πίνακας 2.5).

Σύμβολο	Περιγραφή
 Coincident	Τοποθετεί τα δύο επιλεγμένα στοιχεία (π.χ. επίπεδα, επιφάνειες, ακμές, σημεία από εξαρτήματα) στο ίδιο επίπεδο.
 Parallel	Διατηρεί τα δύο επιλεγμένα στοιχεία παράλληλα μεταξύ τους.
 Perpendicular	Τοποθετεί τα δύο επιλεγμένα στοιχεία σε γωνία 90° μεταξύ τους.
 Tangent	Τοποθετεί ένα κυκλικό, κυλινδρικό, σφαιρικό, κωνικό στοιχείο να εφάπτεται με την επίπεδη επιφάνεια ενός άλλου στοιχείου.
 Concentric	Ομόκεντρη σύνδεση μεταξύ δύο κυκλικών, κυλινδρικών, σφαιρικών, κωνικών στοιχείων.
 Lock	Κλειδώνει δύο στοιχεία στην ίδια θέση και προσανατολισμό.
 1.00mm	Distance: Καθορισμός απόστασης μεταξύ δύο στοιχείων.
 30.00deg	Angle: Τοποθετεί τα επιλεγμένα στοιχεία υπό μια επιθυμητή γωνία μεταξύ τους.

Πίνακας 2.3: Standard mates.

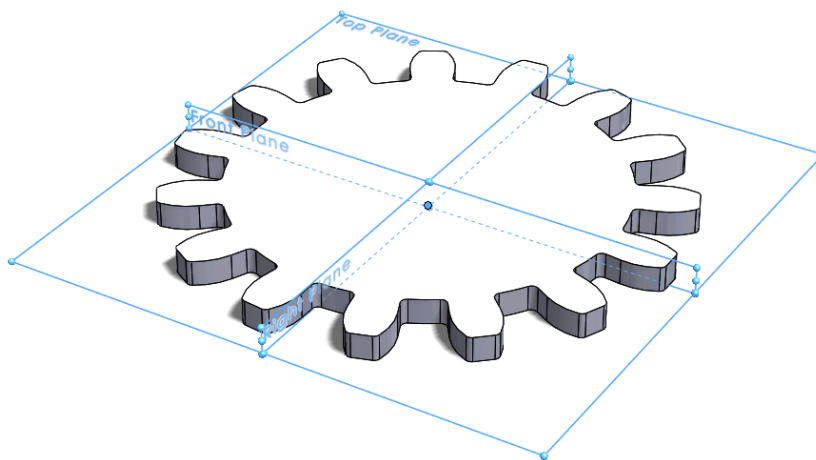
Σύμβολο	Περιγραφή
 Symmetric	Αναγκάζει δύο στοιχεία να γίνουν συμμετρικά σε σχέση με ένα επίπεδο ή μια επιφάνεια.
 Width	Τοποθετεί ένα εξάρτημα στο κέντρο και σε ίση απόσταση ανάμεσα από δύο παράλληλα μεταξύ τους στοιχεία.
 Path Mate	Ορίζει την διαδρομή που θα ακολουθεί ένα σημείο του επιλεγμένου εξαρτήματος.
 Linear/Linear Coupler	Δημιουργεί μια γραμμική σχέση μεταξύ των κινήσεων δύο στοιχείων.
 1.00mm	Distance: Καθορισμός εύρους ελάχιστης, μέγιστης απόστασης μεταξύ δύο στοιχείων καθώς και αρχικής θέσης
 30.00deg	Angle: Καθορισμός εύρους ελάχιστης, μέγιστης γωνίας μεταξύ δύο στοιχείων καθώς και αρχικής θέσης.

Πίνακας 2.4: Advanced mates.

Σύμβολο	Περιγραφή
 Cam	Σύνδεση ενός εξαρτήματος με την επιφάνεια ενός έκκεντρου εξαρτήματος που περιστρέφεται.
 Hinge	Σύνδεση τύπου μεντεσέ μεταξύ δύο στοιχείων που περιορίζει την μετακίνησή τους σε μια γωνία.
 Gear	Σύνδεση μεταξύ γραναζιών ή κυλινδρικών επιφανιών που επιτρέπει την περιστροφή του ενός ως προς το άλλο.
 Rack Pinion	Παρόμοιο με το mate τύπου γραναζιού, εκτός από το ότι η κίνηση του ενός στοιχείου είναι γραμμική και όχι περιστροφική.
 Screw	Σύνδεση τύπου βίδας μεταξύ δύο στοιχείων όπου το ένα στοιχείο είναι σταθερό.
 Universal Joint	Ελεύθερη άρθρωση (σύνδεσμος Cardan).

Πίνακας 2.5: Mechanical mates.

Το SolidWorks ορίζει σε κάθε εξάρτημα τρία επίπεδα (planes) ως προεπιλογές (Σχήμα 2.10). Το μπροστά επίπεδο (front plane), το πλαϊνό επίπεδο (right plane) και το πάνω επίπεδο (top plane). Τα επίπεδα χρησιμοποιούνται για την σχεδίαση εξαρτημάτων και την δημιουργία γεωμετρίας σε διάφορες λειτουργίες όπως την λειτουργία εισαγωγής mates.



Σχήμα 2.10: Επίπεδα εξαρτήματος.

Ένα σχέδιο (εξάρτημα, συγκρότημα ή απλό 2D σχέδιο) στο SolidWorks μπορεί να έχει τρεις καταστάσεις under defined, fully defined ή over defined οι οποίες περιγράφονται ως εξής:

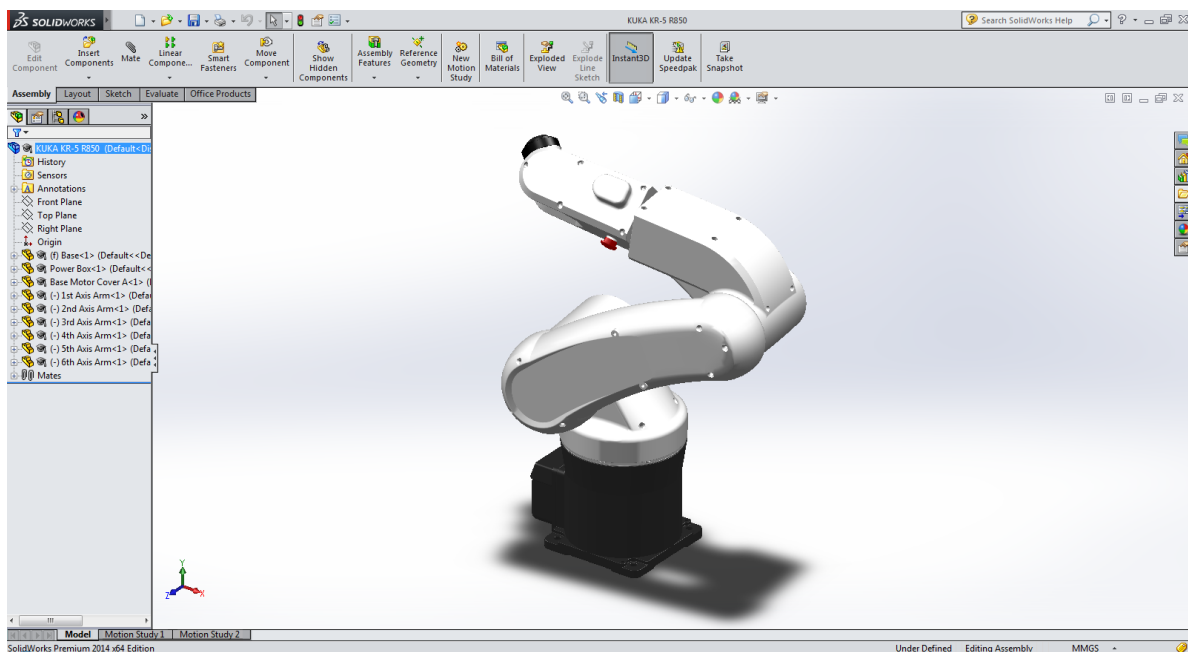
- **Under defined:** ελλιπής καθορισμός γεωμετρίας, το SolidWorks χρειάζεται περισσότερες πληροφορίες για τον έλεγχο της γεωμετρίας. Αυτή η κατάσταση συνήθως προκύπτει όταν υπάρχουν διαστάσεις που δεν έχουν καθοριστεί και ελλιπείς γεωμετρικές σχέσεις (mates). Αυτό μπορεί να οδηγήσει σε λάθος υπολογισμούς και αποτελέσματα και δυσλειτουργία.

- **Fully defined:** πλήρως καθορισμένη γεωμετρία, το SolidWorks έχει όλες τις πληροφορίες που χρειάζεται σε σχέση με την γεωμετρία.
- **Over defined:** το SolidWorks έχει πάρα πολλές πληροφορίες σε βαθμό που μπερδεύεται. Προκύπτει από εισαγωγή πληροφοριών που δεν χρειάζονται γιατί υπάρχουν ήδη ή γιατί έχουν προκύψει από άλλους γεωμετρικούς καθορισμούς και από την εισαγωγή αντικρουόμενων mates. Σε αυτή την περίπτωση το σχέδιο δεν μπορεί να αξιοποιηθεί εάν δεν διορθωθεί η κατάσταση.

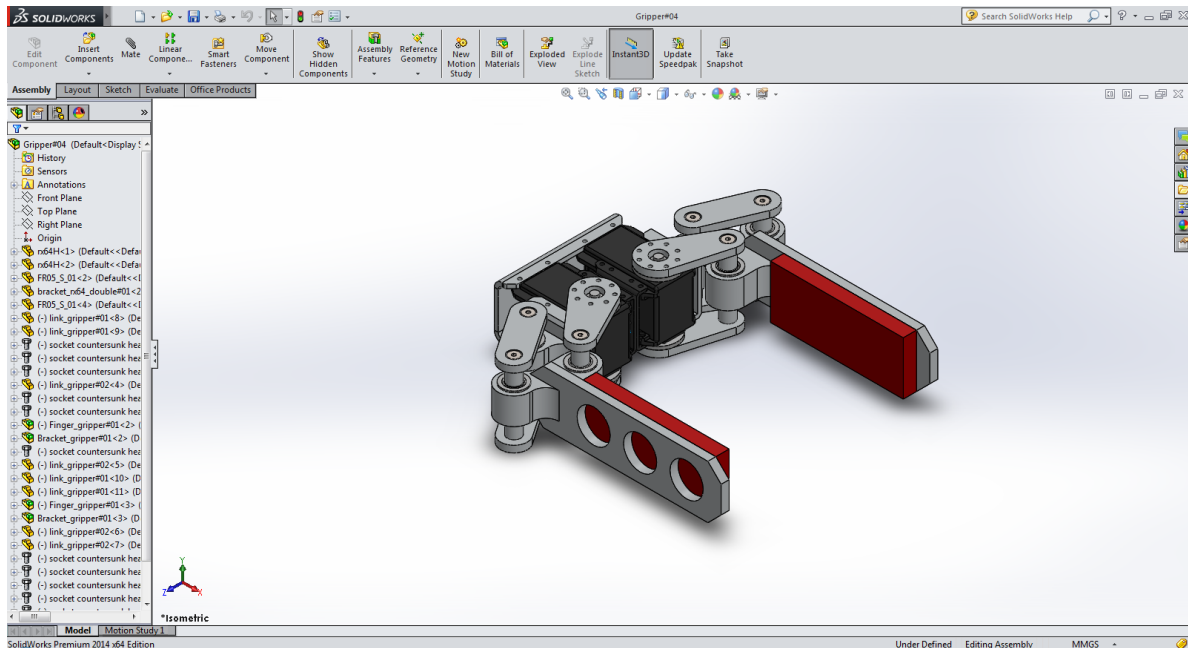
2.6 Υλοποίηση του εικονικού ρομποτικού βραχίονα

Ο εικονικός ρομποτικός βραχίονας είναι αποτέλεσμα μιας συναρμολόγησης. Αποτελείται από το τρισδιάστατο μοντέλο του βραχίονα KUKA KR 5 sixx R850 (Σχήμα 2.11) και το τρισδιάστατο μοντέλο μιας αρπάγης (Σχήμα 2.12), η οποία αποτελεί το τελικό στοιχείο δράσης του εικονικού ρομποτικού βραχίονα.

Τα δύο παραπάνω τρισδιάστατα μοντέλα έγιναν λήψη από την ιστοσελίδα GrabCAD (<https://grabcad.com>), η οποία είναι μια ιστοσελίδα διαμοιρασμού αρχείων CAD. Το τρισδιάστατο μοντέλο του βραχίονα έχει υλοποιηθεί από τον χρήστη Sugi Tjiu (<https://grabcad.com/library/kuka-kr-5-r850>) και αυτό της αρπάγης από τον χρήστη Amir H. Soltanzadeh (<https://grabcad.com/library/4-bar-linkage-gripper-with-dynamixel-rx-64-1>).



Σχήμα 2.11: Τρισδιάστατο μοντέλο KUKA KR 5 sixx R850.



Σχήμα 2.12: Τρισδιάστατο μοντέλο αρπάγης.

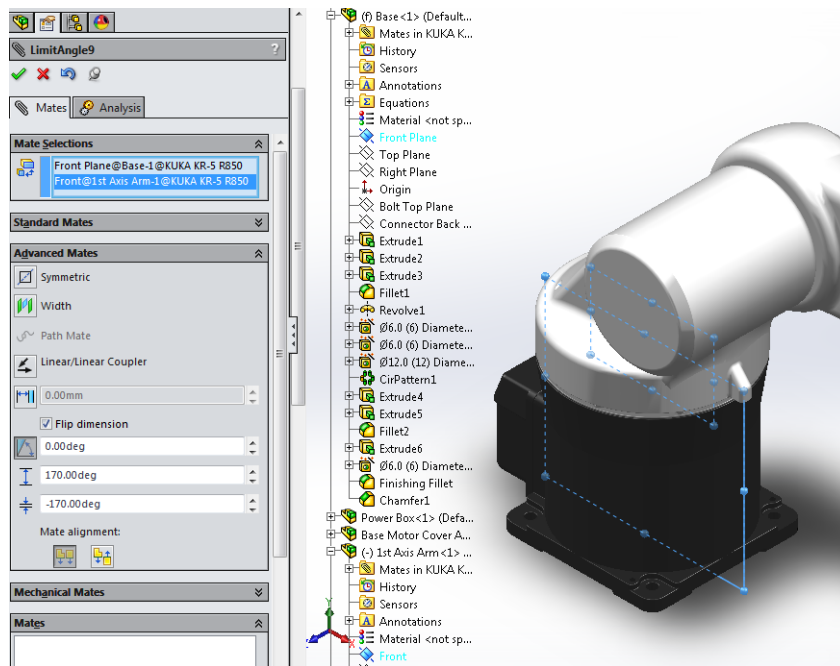
Ωστόσο, πριν από την τελική συναρμολόγηση, τα δύο παραπάνω τρισδιάστατα μοντέλα χρειάστηκε να τροποποιηθούν ως προς τις γεωμετρικές σχέσεις μεταξύ των εξαρτημάτων τους έτσι, ώστε να λειτουργούν σύμφωνα με τις ανάγκες και τις απαιτήσεις της παρούσας πτυχιακής εργασίας. Μετά από προσθαφαίρεση mates καθορίστηκαν πλήρως γεωμετρικά και ορίστηκαν οι επιθυμητές κινήσεις με τα αντίστοιχα όρια και κατευθύνσεις.

2.6.1 Τροποποίηση μοντέλου ρομποτικού βραχίονα

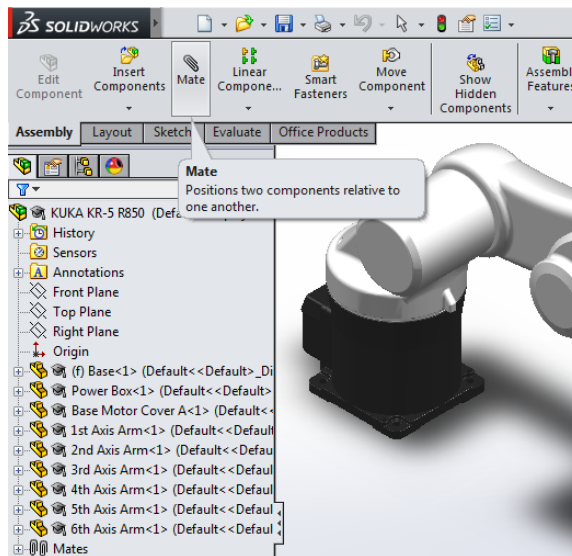
Αναλυτικότερα στο μοντέλο του ρομποτικού βραχίονα προστέθηκαν επτά νέα mates, εκ των οποίων πέντε για τον καθορισμό των κινήσεων και δύο για την βελτίωση της σταθερότητας του μοντέλου. Για τον καθορισμό των κινήσεων χρησιμοποιήθηκαν mates γωνίας με διαμόρφωση εύρους ελάχιστης και μέγιστης γωνίας από την κατηγορία advanced mates.

Ο καθορισμός της κίνησης του πρώτου άξονα (Σχήμα 2.13) πραγματοποιήθηκε με τα παρακάτω βήματα:

- Εισαγωγή στην λειτουργία και το μενού των mates, πατώντας στην λειτουργία mate της καρτέλας assembly (Σχήμα 2.14).
- Από το design tree επιλέγονται το μπροστά επίπεδο της βάσης και το μπροστά επίπεδο του πρώτου συνδέσμου, ούτως ώστε ο πρώτος σύνδεσμος να περιστρέφεται ως προς τη βάση.
- Επιλογή του mate γωνίας με διαμόρφωση εύρους ελάχιστης και μέγιστης γωνίας από την κατηγορία advanced mates.
- Διαμόρφωση εύρους με ελάχιστη γωνία -170° και μέγιστη 170° σύμφωνα με το όριο περιστροφής του πρώτου άξονα από τα τεχνικά χαρακτηριστικά του KUKA KR 5 sixx R850.
- Καθορισμός γωνίας αρχικής θέσης οι 0° .

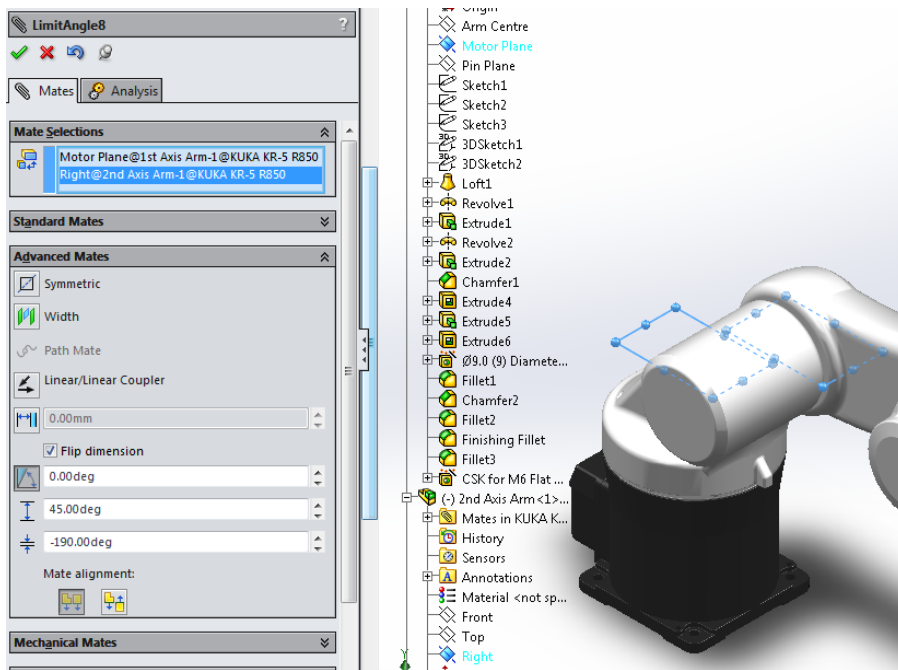


Σχήμα 2.13: Καθορισμός κίνησης πρώτου άξονα.



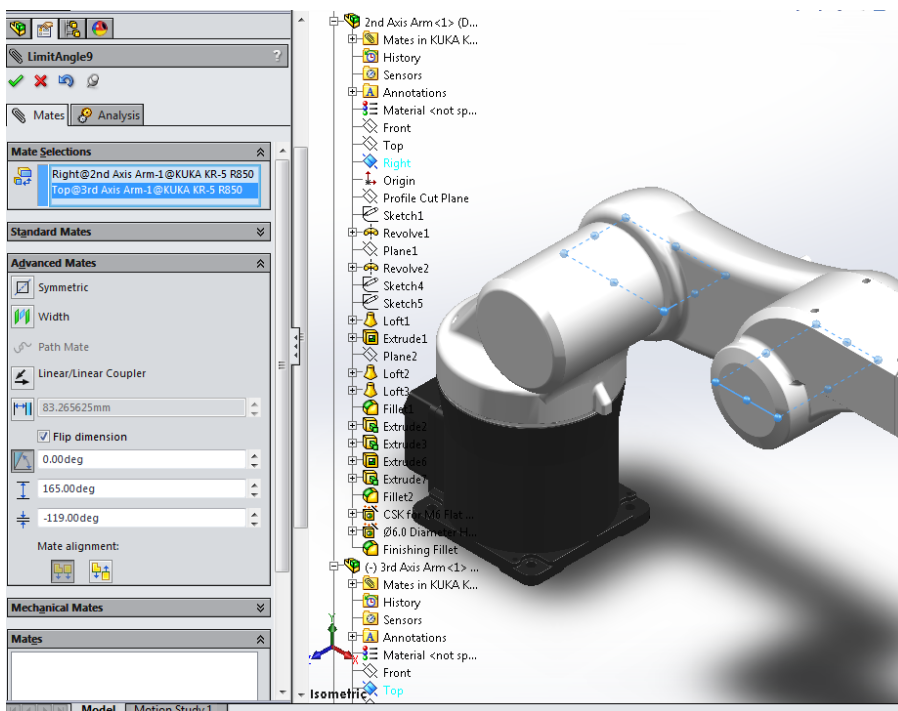
Σχήμα 2.14: Εισαγωγή στην λειτουργία mate.

Ομοίως και για τους υπόλοιπους άξονες επιλέγοντας τα αντίστοιχα επίπεδα και όρια περιστροφής. Για τον δεύτερο άξονα επιλέχτηκε το επίπεδο του πλαισίου του κινητήρα που βρίσκεται στον πρώτο σύνδεσμο και το πλαινό επίπεδο του δεύτερου συνδέσμου, με μέγιστη γωνία 45° , ελάχιστη -190° και αρχική θέση τις 0° (Σχήμα 2.15). Οι 0° έχουν καθοριστεί σαν αρχική θέση για όλους τους άξονες. Με την επιλογή flip dimension ορίζεται ποια πλευρά περιστροφής λαμβάνεται ως θετική και με την επιλογή mate alignment γίνεται αντιστροφή ευθυγράμμισης, για παράδειγμα σαν 0° λαμβάνονται υπόψη οι 180° και σαν 180° οι 0° .

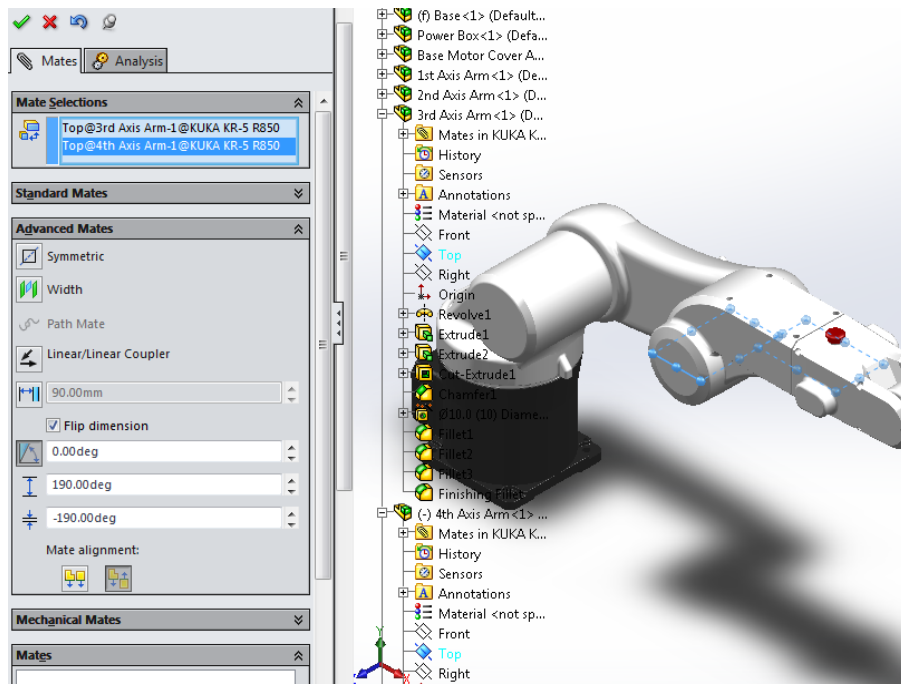


Σχήμα 2.15: Καθορισμός κίνησης δεύτερου άξονα.

Για τον τρίτο άξονα επιλέχτηκε το πλαϊνό επίπεδο του δεύτερου συνδέσμου και το πάνω επίπεδο του τρίτου συνδέσμου, με ελάχιστη γωνία τις -119° και μέγιστη γωνία τις 45° (Σχήμα 2.16). Για τον τέταρτο άξονα το πάνω επίπεδο του τρίτου συνδέσμου και το πάνω επίπεδο του τέταρτου συνδέσμου που ανήκει στον καρπό, με όριο τις $\pm 190^\circ$ (Σχήμα 2.17).

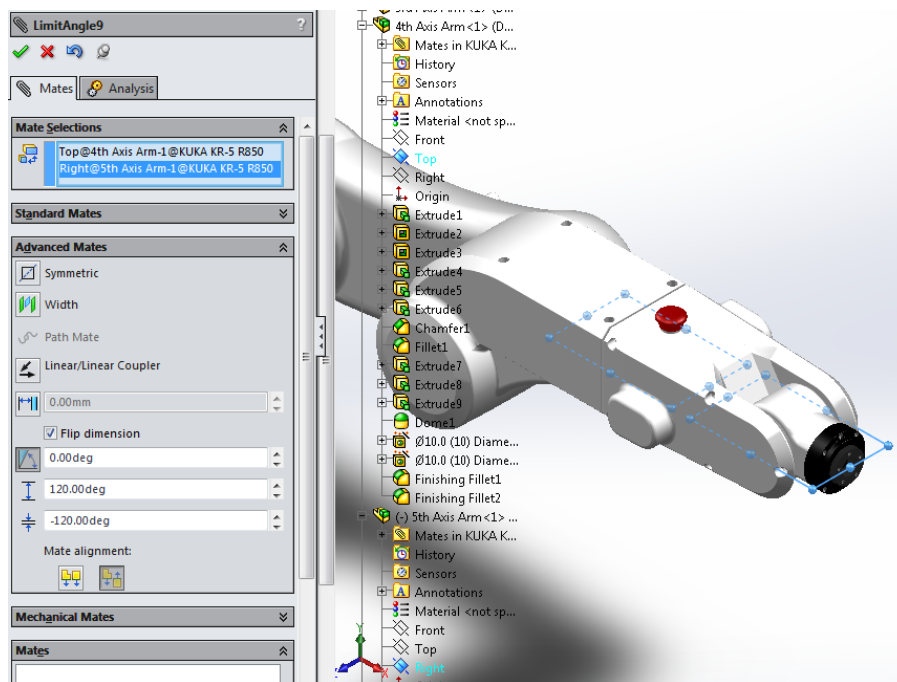


Σχήμα 2.16: Καθορισμός κίνησης τρίτου άξονα.



Σχήμα 2.17: Καθορισμός κίνησης τέταρτου άξονα.

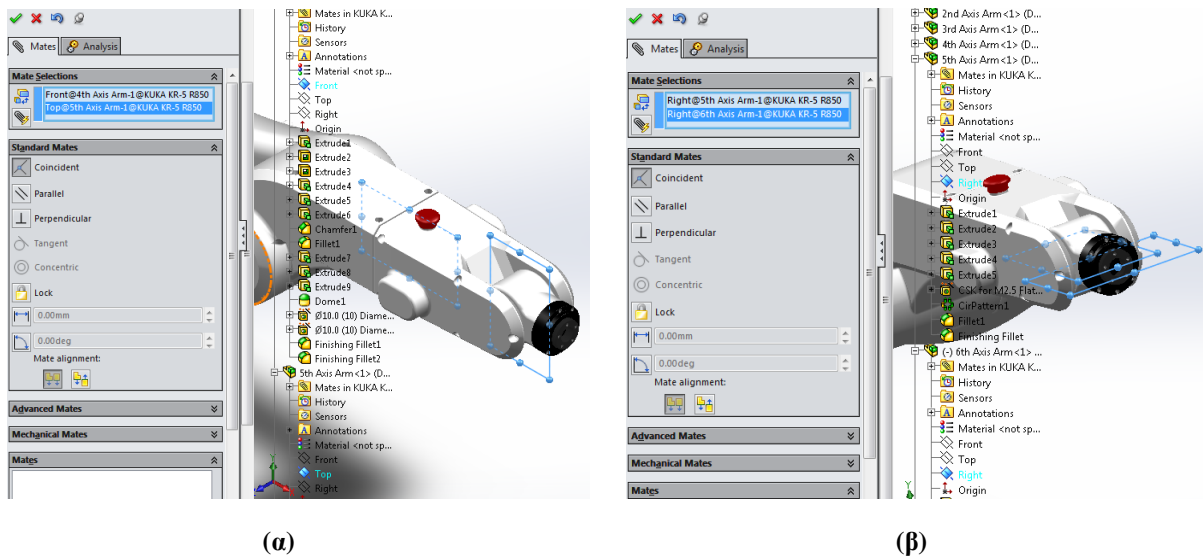
Και τέλος για τον καθορισμό της κίνησης του πέμπτου άξονα επιλέχτηκε το πάνω επίπεδο του τέταρτου συνδέσμου και το πλαϊνό επίπεδο του πέμπτου συνδέσμου που επίσης ανήκει στον καρπό, με εύρος περιστροφής τις $\pm 120^\circ$ (Σχήμα 2.18).



Σχήμα 2.18: Καθορισμός κίνησης πέμπτου άξονα.

Τα mates που προστέθηκαν για την βελτίωση της σταθερότητας του μοντέλου είναι δύο coincident mates, ένα μεταξύ του μπροστά επίπεδου του τέταρτου συνδέσμου και του πλαϊνού επιπέδου του πέμπτου συνδέσμου (Σχήμα 2.19α) και ένα μεταξύ του πλαϊνού επιπέδου του πέμπτου συνδέσμου και του πλαϊνού επιπέδου του άκρου του καρπού (Σχήμα

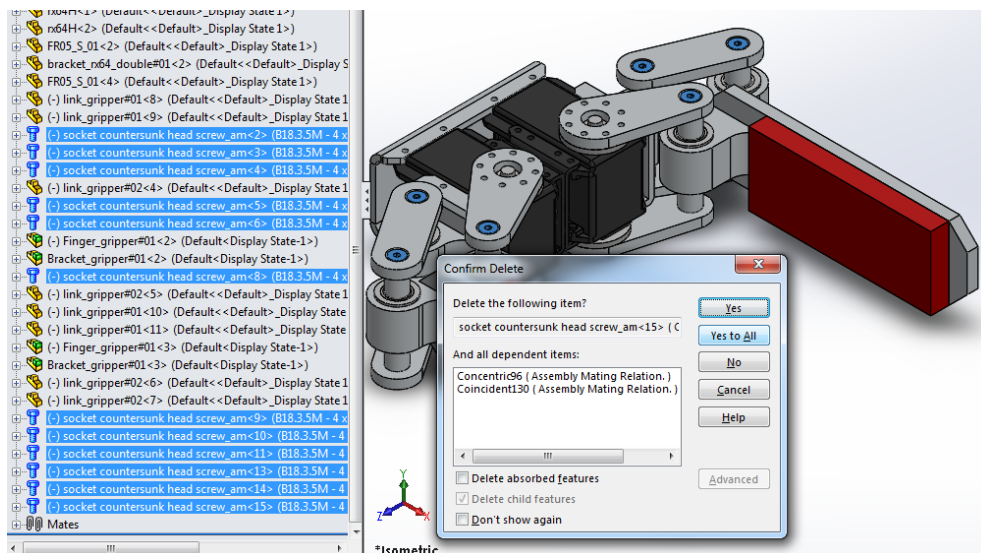
2.19β). Έπειτα από την προσθήκη όλων των παραπάνω mates η κατάσταση του μοντέλου από under defined άλλαξε σε fully defined.



Σχήμα 2.19: (α) Προσθήκη coincident mate σταθερότητας 1, (β) προσθήκη coincident mate σταθερότητας 2.

2.6.2 Τροποποίηση μοντέλου αρπάγης

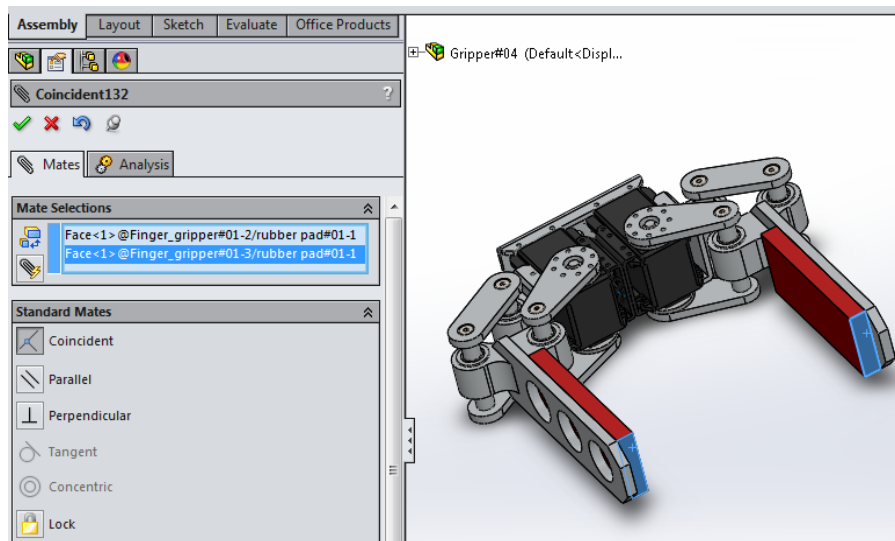
Όσον αφορά το μοντέλο της αρπάγης, καταργήθηκαν όλες οι βίδες των συνδέσμων και όλα τα mates που σχετίζονται με αυτές καθώς δημιουργούσαν πρόβλημα στο να αλλάξει η κατάσταση του μοντέλου από under defined σε fully defined. Ούτως ή άλλως η χρησιμοποίησή τους ήταν καθαρά λόγω ρεαλιστικής απεικόνισης μιας και οι συνδέσεις δημιουργούνται μέσω των γεωμετρικών σχέσεων μεταξύ των εξαρτημάτων. Η κατάργηση έγινε επιλέγοντας τες από το design tree και πατώντας delete μέσω δεξί κλικ (Σχήμα 2.20).



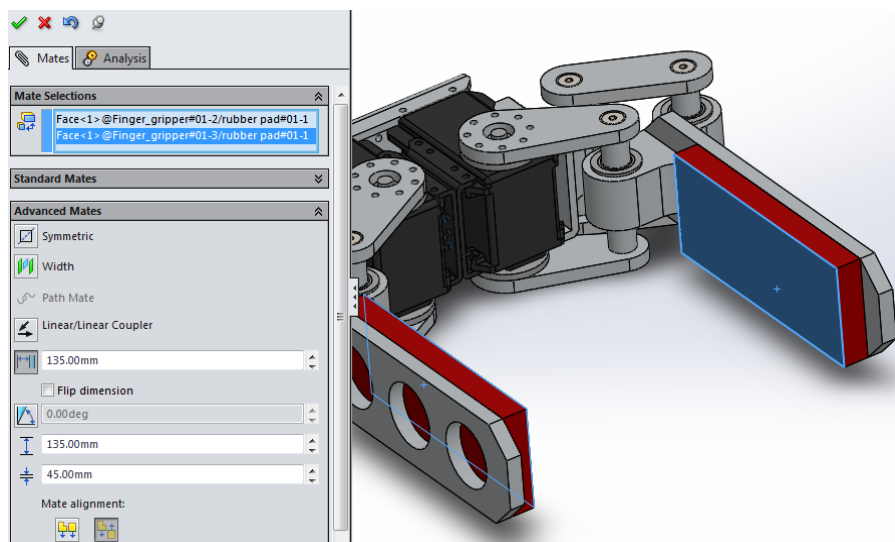
Σχήμα 2.20: Κατάργηση των βιδών και των σχετικών mates από την αρπάγη.

Επίσης, προστέθηκαν κάποια mates που καθόρισαν την κίνηση της αρπάγης. Ένα coincident mate μεταξύ των επιφανειών που φαίνονται Σχήμα 2.21 ώστε να διατηρεί τα δύο άκρα στο ίδιο επίπεδο κατά το άνοιξε κλείσε. Και ένα advanced mate απόστασης με

καθορισμό ελάχιστης απόστασης τα 45 mm, μέγιστης τα 135 mm και αρχική θέση τα 135 mm μεταξύ των εσωτερικών επιφανειών των δύο άκρων της αρπάγης (Σχήμα 2.22).



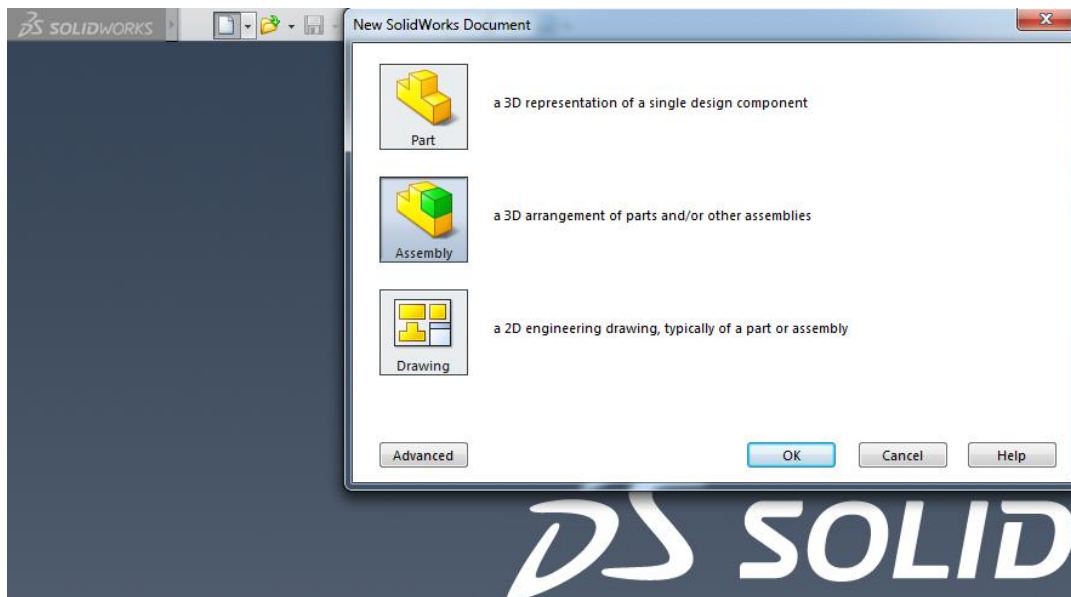
Σχήμα 2.21: Προσθήκη coincident mate.



Σχήμα 2.22: Προσθήκη advanced mate απόστασης.

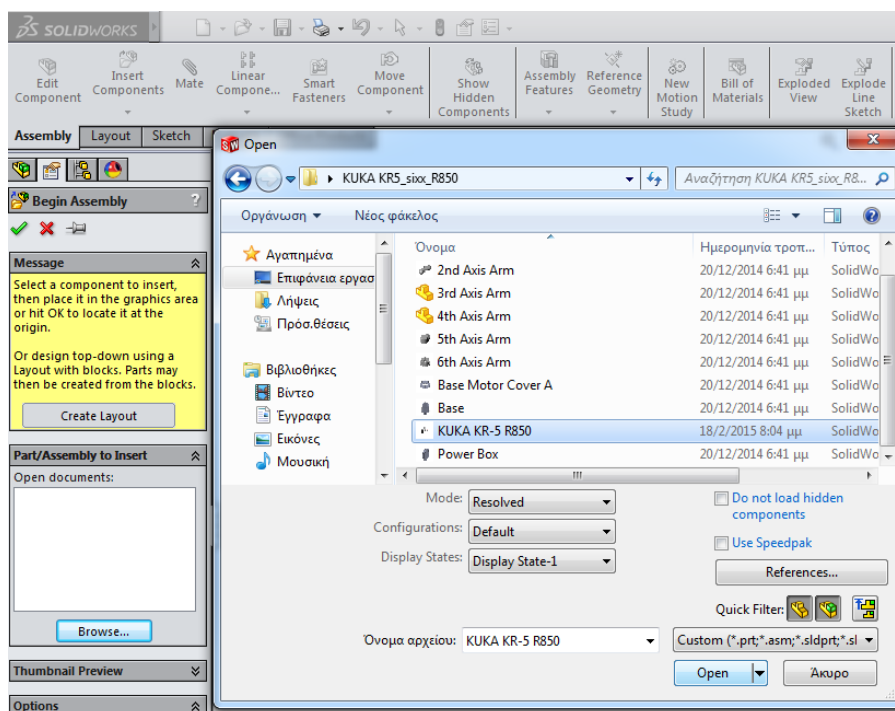
2.6.3 Συναρμολόγηση

Η διαδικασία της συναρμολόγησης ξεκινάει ανοίγοντας το SolidWorks, κάνοντας κλικ στο New (δημιουργία νέου έγγραφου) και επιλέγοντας την λειτουργία assembly (συναρμολόγηση) όπως παρουσιάζεται στο Σχήμα 2.23.

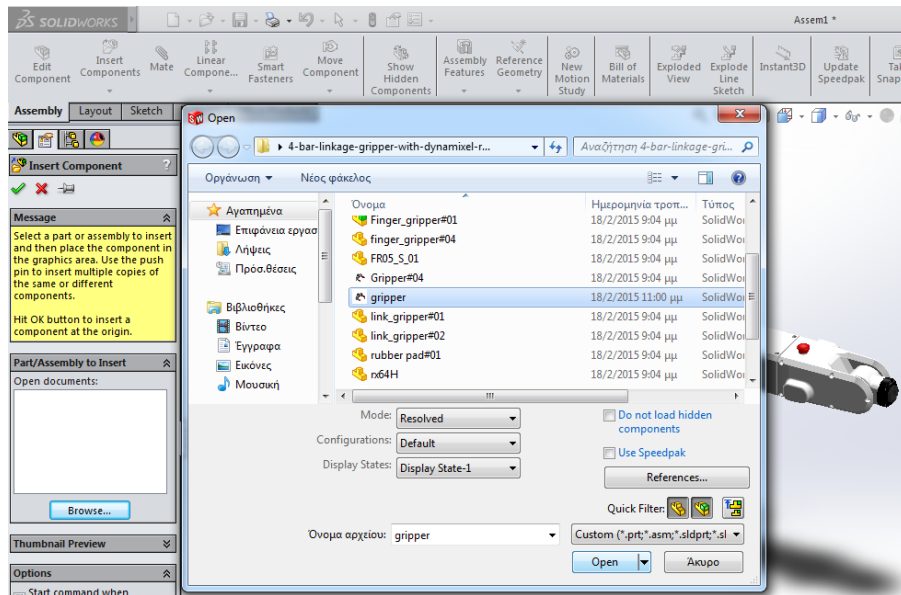


Σχήμα 2.23: Εισαγωγή στην λειτουργία συναρμολόγησης.

Πρώτα εισάγεται το τρισδιάστατο μοντέλο του ρομποτικού βραχίονα, λόγω της βάσης του που είναι πακτωμένη ώστε να δημιουργηθεί το σημείο αναφοράς της συναρμολόγησης (Σχήμα 2.24). Στη συνέχεια πατώντας στην επιλογή insert components προστίθεται και το τρισδιάστατο μοντέλο της αρπάγης (Σχήμα 2.25).

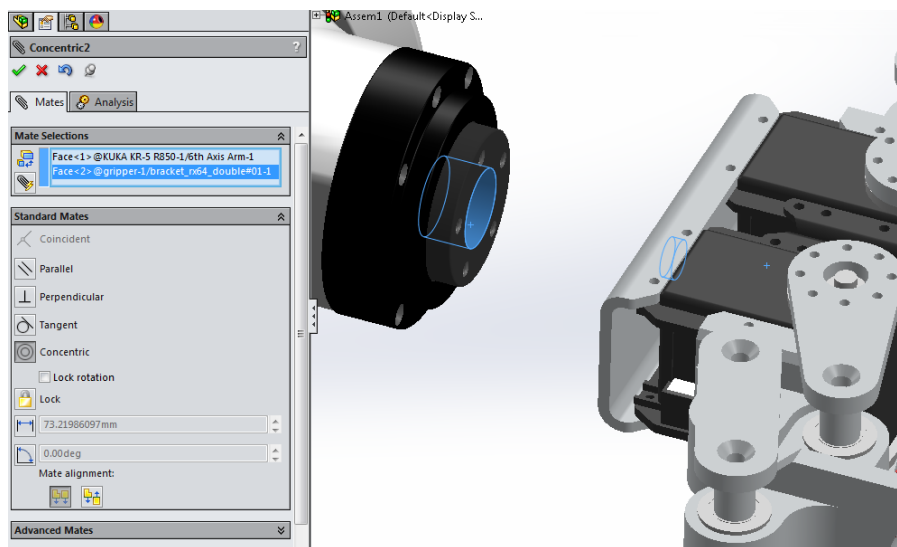


Σχήμα 2.24: Εισαγωγή τρισδιάστατου μοντέλου ρομποτικού βραχίονα.

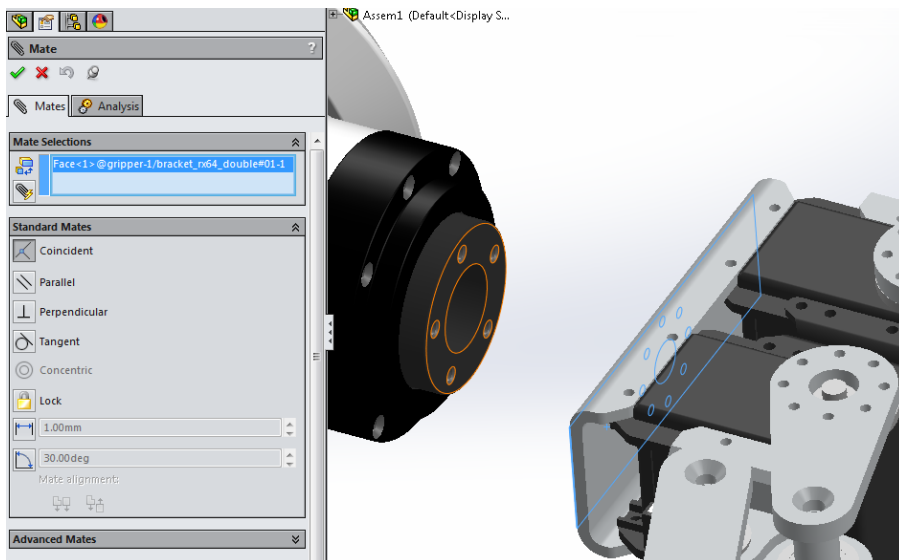


Σχήμα 2.25: Εισαγωγή τρισδιάστατου μοντέλου αρπάγης.

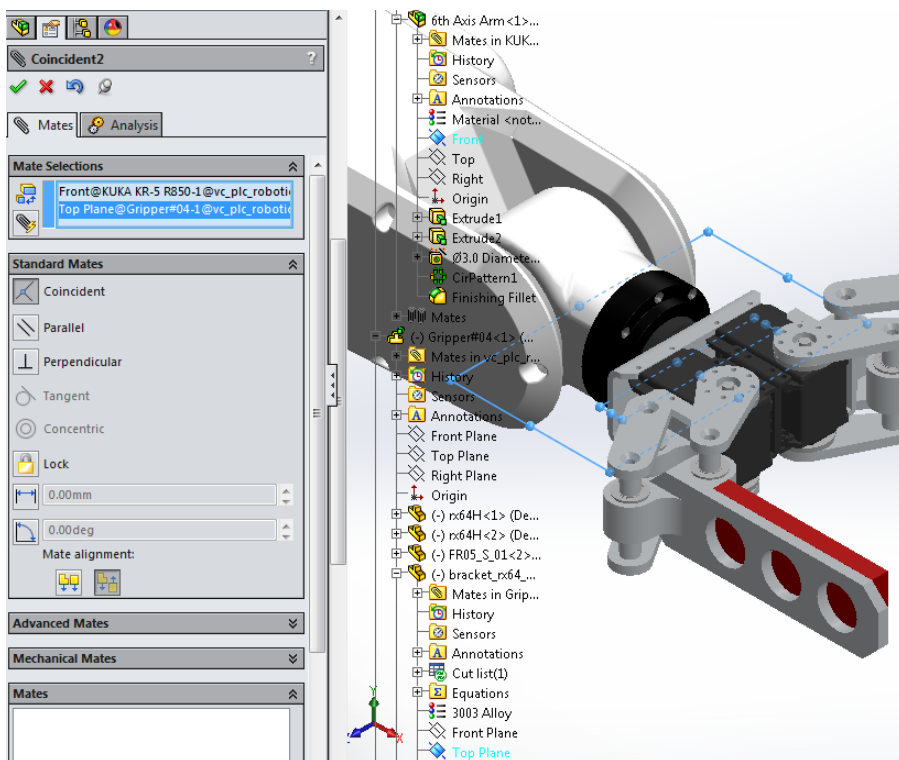
Για την συναρμολόγηση χρησιμοποιήθηκε ένα concentric mate μεταξύ της εσωτερικής κυλινδρικής επιφάνειας του άκρου του καρπού και της εσωτερικής κυλινδρικής επιφάνειας της βάσης της αρπάγης (Σχήμα 2.26). Ένα coincident mate που ενώνει την μπροστά επιφάνεια του άκρου του καρπού με την πίσω επιφάνεια της βάσης της αρπάγης (Σχήμα 2.27). Και ένα coincident mate για σταθερότητα μεταξύ του μπροστά επίπεδου του άκρου του καρπού με το πάνω επίπεδο της βάσης της αρπάγης (Σχήμα 2.28).



Σχήμα 2.26: Προσθήκη concentric mate.

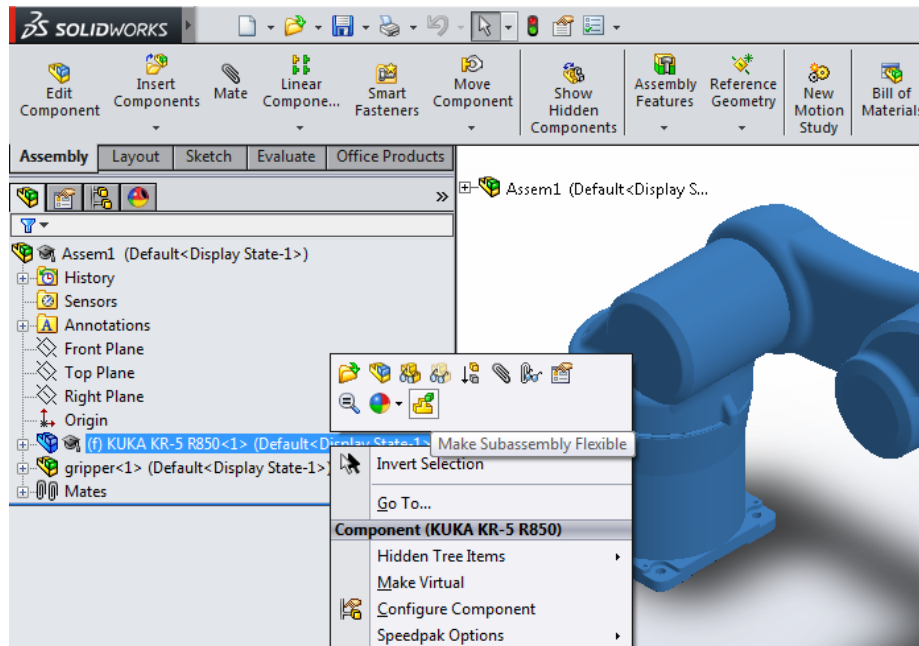


Σχήμα 2.27: Προσθήκη coincident mate.

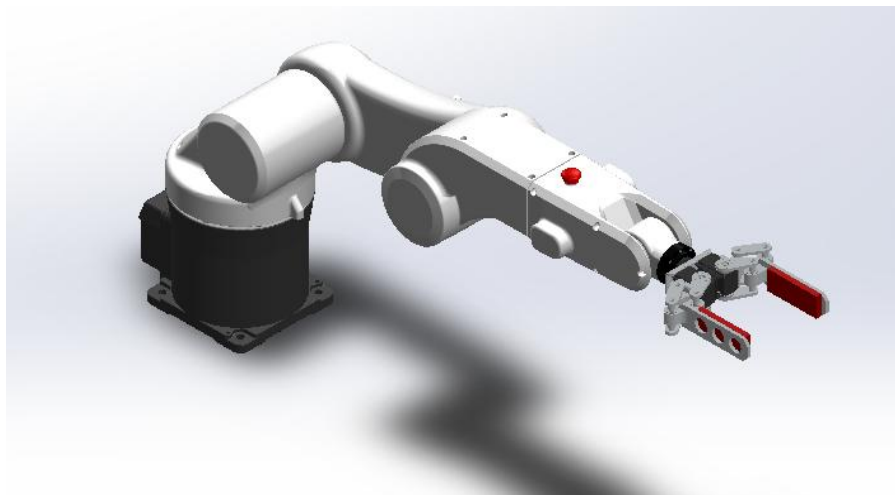


Σχήμα 2.28: Προσθήκη coincident mate.

Το SolidWorks από προεπιλογή αλλάζει την κατάσταση των subassemblies προς συναρμολόγηση κατά την εισαγωγή τους, από flexible (ευέλικτο) σε rigid (άκαμπτο) προς διευκόλυνση της συναρμολόγησης. Το τελευταίο βήμα μετά την συναρμολόγηση είναι να ξαναγίνουν ευέλικτα τα subassemblies από τα οποία αποτελείται το τελικό assembly. Αυτό γίνεται από το design tree πατώντας δεξί κλικ πάνω στα subassemblies και πατώντας στην επιλογή make subassembly flexible (Σχήμα 2.29). Με την ολοκλήρωση των παραπάνω διαδικασιών είναι διαθέσιμο το τελικό τρισδιάστατο μοντέλο του εικονικού ρομποτικού βραχίονα (Σχήμα 2.30).



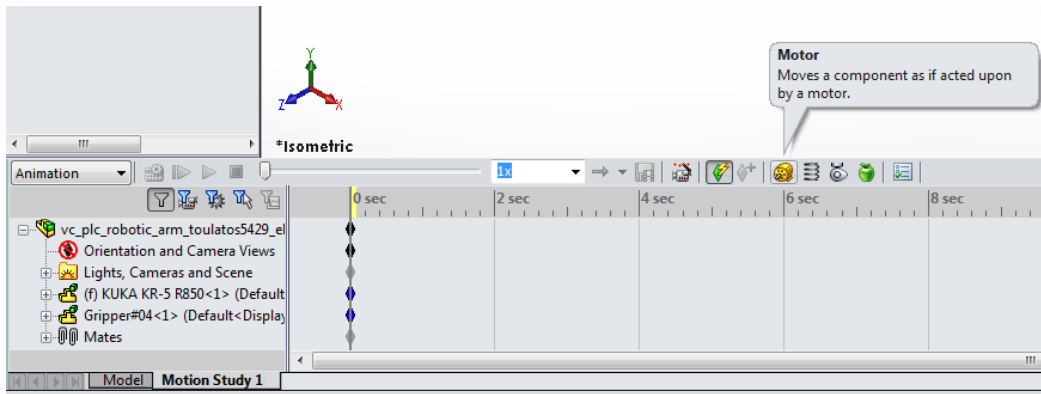
Σχήμα 2.29: Μετατροπή subassembly από rigid σε flexible.



Σχήμα 2.30: Τελικό τρισδιάστατο μοντέλο εικονικού ρομποτικού βραχίονα.

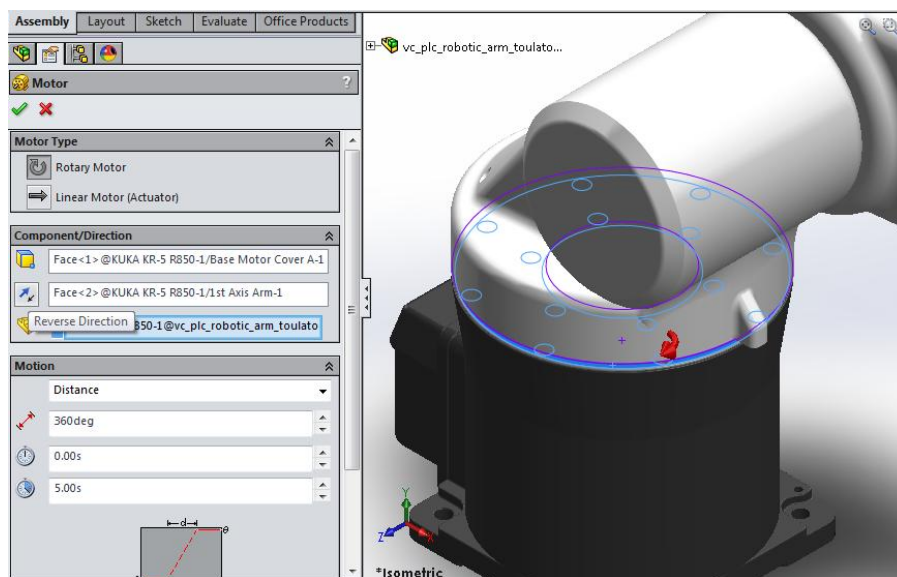
2.6.4 Εισαγωγή κινητήρων

Για την αξιοποίηση των κινήσεων που καθορίστηκαν παραπάνω, αλλά και του ίδιου του εικονικού ρομποτικού βραχίονα κατά την διάρκεια της διαδικασίας VC, πρέπει να εισαχθεί η λειτουργία κινητήρα σε κάθε άξονα έτσι ώστε να μπορεί να γίνει ο έλεγχος της κίνησης. Η διαδικασία είναι παρόμοια με αυτή του καθορισμού των κινήσεων. Αρχικά επιλέγεται η καρτέλα motion study που βρίσκεται κάτω αριστερά στην βασική οθόνη και έπειτα επιλέγεται η λειτουργία motor (Σχήμα 2.31).

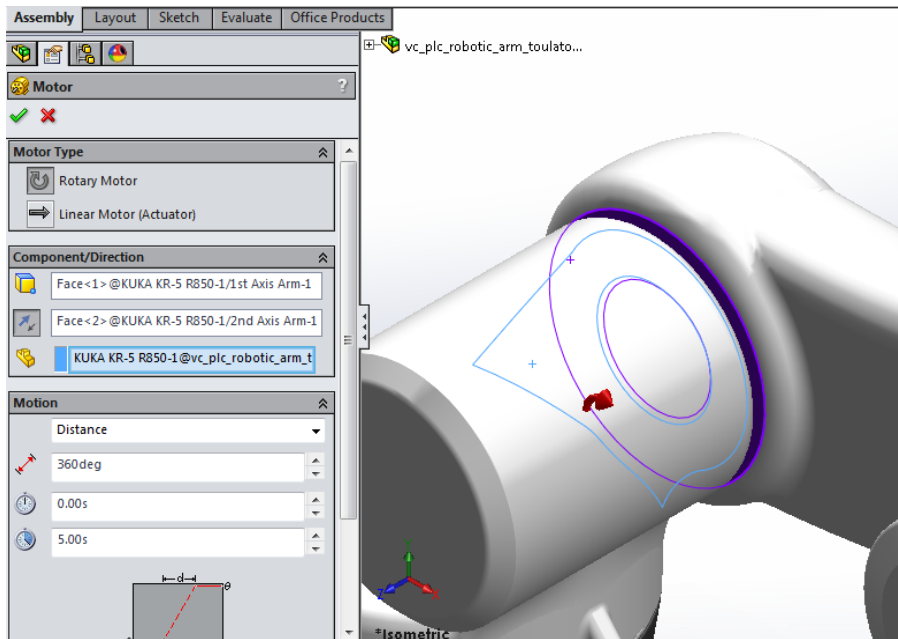


Σχήμα 2.31: Εισαγωγή στην λειτουργία κινητήρα.

Μετά επιλέγονται οι επιφάνειες των εξαρτημάτων που συμμετέχουν στην κίνηση. Στην παρούσα περίπτωση για τον πρώτο άξονα, η πάνω κυκλική επιφάνεια της βάσης και η κάτω κυκλική επιφάνεια του πρώτου συνδέσμου, έτσι ώστε ο πρώτος σύνδεσμος να περιστρέφεται ως προς τη βάση. Από τις διαθέσιμες επιλογές που σχετίζονται με τον τύπο κινητήρα (motor type) επιλέγεται ο περιστροφικός (rotary motor), όπως και σε όλες τις άλλες κινήσεις πλην της αρπάγης όπου τοποθετείται ο γραμμικός κινητήρας (linear motor). Στις επιλογές κίνησης (motion) επιλέγεται απόσταση (distance) με μετατόπιση 360° (Σχήμα 2.32). Οι ρυθμίσεις αυτές παραμένουν ίδιες και για τους υπόλοιπους άξονες, εκτός από την μετατόπιση της αρπάγης η οποία είναι τα 100 mm. Ομοίως και για τους υπόλοιπους άξονες επιλέγοντας τις αντίστοιχες επιφάνειες. Για τον δεύτερο άξονα επιλέχθηκαν οι δύο απέναντι κυκλικές επιφάνειες μεταξύ του πρώτου συνδέσμου και του δεύτερου συνδέσμου (Σχήμα 2.33).

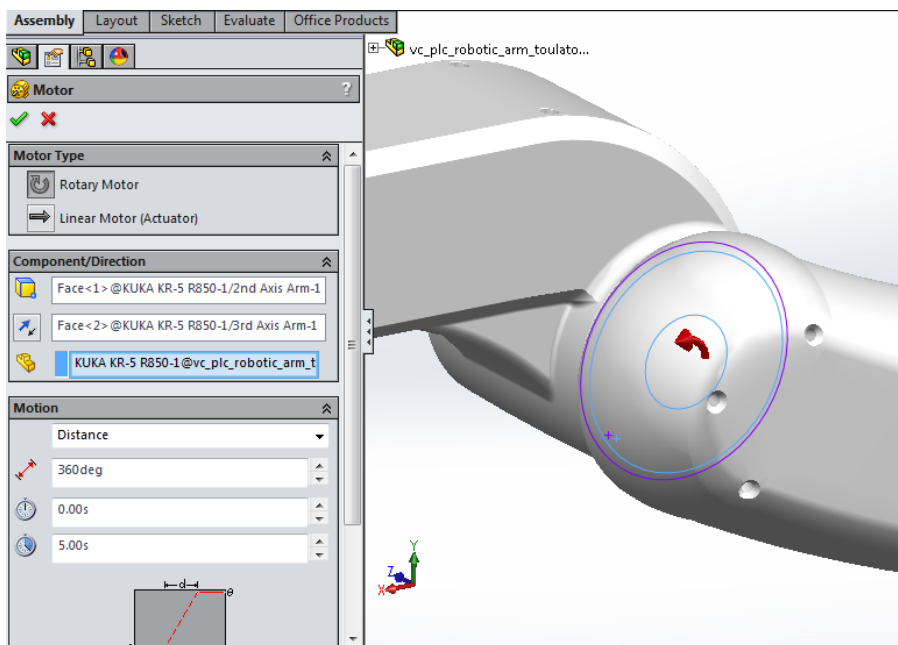


Σχήμα 2.32: Εισαγωγή κινητήρα πρώτου άξονα.

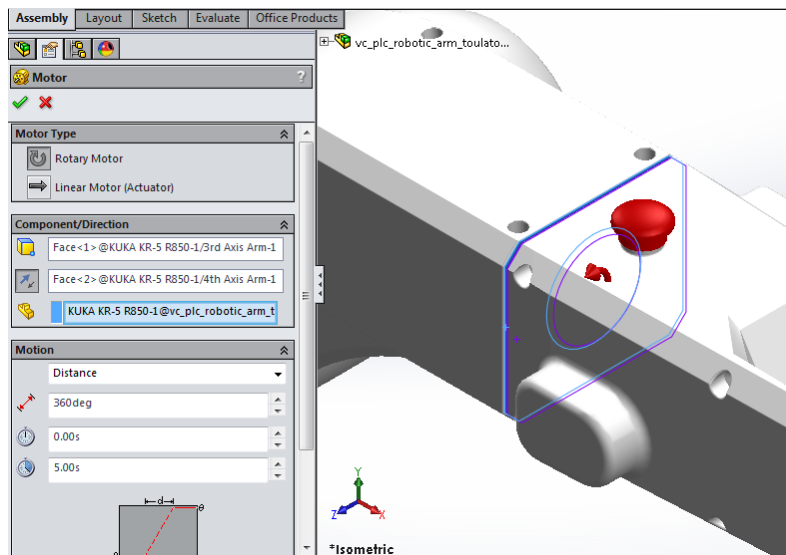


Σχήμα 2.33: Εισαγωγή κινητήρα δεύτερου άξονα.

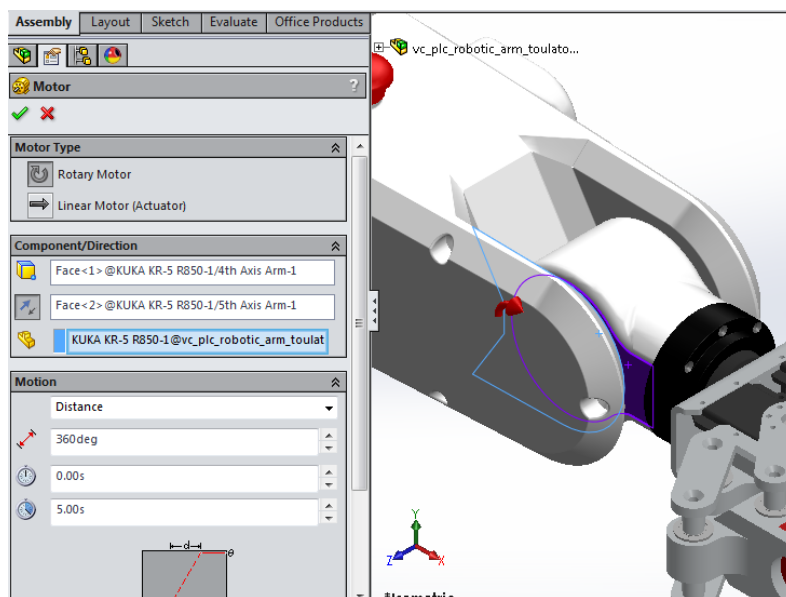
Για τον τρίτο άξονα επιλέχθηκαν οι απέναντι κυκλικές επιφάνειες του δεύτερου συνδέσμου και του τρίτου συνδέσμου (Σχήμα 2.34). Για τον τέταρτο άξονα επιλέχθηκαν οι απέναντι επιφάνειες μεταξύ του τρίτου και του τέταρτου συνδέσμου (Σχήμα 2.35). Για τον πέμπτο άξονα επιλέχθηκαν οι απέναντι επιφάνειες που φαίνονται στο Σχήμα 2.36 και τέλος, για την αρπάγη επιλέχθηκαν οι δύο εσωτερικές επιφάνειες των άκρων της (Σχήμα 2.37).



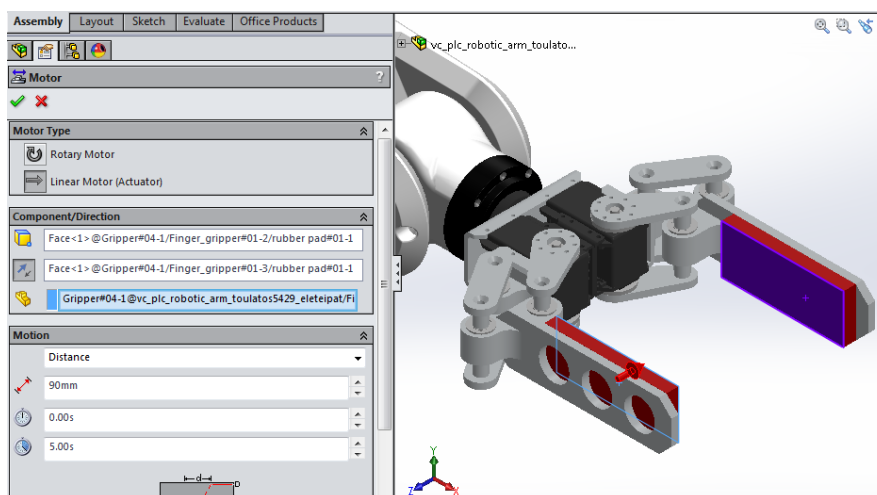
Σχήμα 2.34: Εισαγωγή κινητήρα τρίτου άξονα.



Σχήμα 2.35: Εισαγωγή κινητήρα τέταρτου άξονα.



Σχήμα 2.36: Εισαγωγή κινητήρα πέμπτου άξονα.



Σχήμα 2.37: Εισαγωγή κινητήρα αρπάγης.

ΚΕΦΑΛΑΙΟ 3

Ο έλεγχος του εικονικού ρομποτικού βραχίονα

3.1 Εισαγωγή

Ο έλεγχος του εικονικού ρομποτικού βραχίονα στην παρούσα πτυχιακή εργασία επιτυγχάνεται μέσω PLC, του οποίου η λειτουργία εξομοιώνεται με το λογισμικό S7-PLCSIM στην περίπτωση της μεθόδου SIL. Στην μέθοδο HIL ο έλεγχος επιτυγχάνεται με πραγματικό PLC. Ο κώδικας που εκτελεί το PLC είναι ο ίδιος και στις δύο μεθόδους και ο προγραμματισμός πραγματοποιήθηκε μέσω του λογισμικού STEP 7. Το λογισμικό STEP 7 χρησιμοποιείται για την διαμόρφωση (υλικού, επικοινωνίας κλπ), την παραμετροποίηση και τον προγραμματισμό των SIMATIC PLCs.

Η διαμόρφωση του HMI (Human Machine Interface) για τον εποπτικό έλεγχο και την συλλογή δεδομένων της εφαρμογής πραγματοποιήθηκε με το λογισμικό WinCC. Το λογισμικό WinCC χρησιμοποιείται για την ανάπτυξη και υλοποίηση συστημάτων SCADA (Supervisory Control And Data Acquisition) καθώς και την διαμόρφωση των αντίστοιχων HMIs. Επίσης, μέσω του WinCC μπορούν να υλοποιηθούν συστήματα SCADA που εκτελούνται είτε σε μια συσκευή απεικόνισης είτε σε κάποιο προσωπικό υπολογιστή. Όλα τα παραπάνω λογισμικά περιλαμβάνονται στην σουίτα TIA Portal της SIEMENS.

3.2 Ο προγραμματιζόμενος λογικός ελεγκτής

Το PLC είναι ένας βιομηχανικός ελεγκτής που βασίζεται σε μικροεπεξεργαστή. Χρησιμοποιεί μια προγραμματιζόμενη μνήμη για την αποθήκευση εντολών ώστε να επιτελούνται διάφορες λειτουργίες όπως λογικές πράξεις, αριθμητικές πράξεις, χρονικά και απαριθμητές. Επίσης, επιτρέπει τον έλεγχο διαφόρων βιομηχανικών παραγωγικών διαδικασιών αξιοποιώντας τις απαραίτητες μονάδες εισόδων/εξόδων που διακρίνονται σε αναλογικές και ψηφιακές.

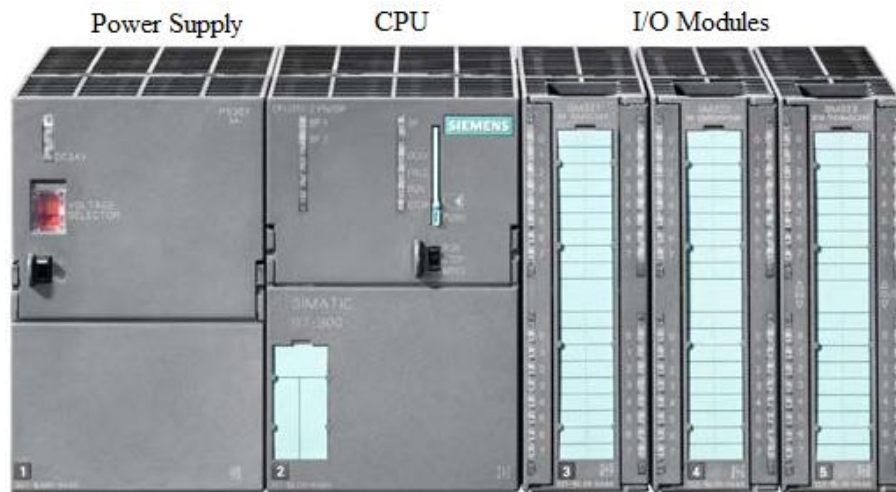
3.2.1 Η δομή του SIMATIC S7-300 PLC

Τα βασικά μέρη από τα οποία αποτελείται ένα τυπικό σύστημα PLC είναι το PS (Power Supply), η CPU (Central Processing Unit), και οι μονάδες εισόδων/εξόδων (I/O Modules) όπως φαίνεται και στο Σχήμα 3.1.

Η CPU αποτελεί τον εγκέφαλο του PLC, ελέγχει και εκτελεί όλες τις λειτουργίες του PLC. Είναι η βασική μονάδα που είναι υπεύθυνη για την λειτουργία του εκάστοτε αυτοματισμού. Με βάση τις τιμές που διαβάζει από τις εισόδους ενεργοποιεί τις κατάλληλες εξόδους σύμφωνα με το πρόγραμμα που εκτελεί και είναι αποθηκευμένο στη μνήμη της CPU.

Εσωτερικά ο μικροεπεξεργαστής της CPU ενός PLC αποτελείται από την ALU (Arithmetic and Logic Unit), η οποία είναι υπεύθυνη για την εκτέλεση αριθμητικών και λογικών πράξεων. Τους καταχωρητές, που χρησιμοποιούνται για προσωρινή αποθήκευση δεδομένων που αφορούν την εκτέλεση του προγράμματος. Και μια μονάδα ελέγχου που ελέγχει και συγχρονίζει όλες τις μεταφορές και την επεξεργασία των δεδομένων. Επίσης, η CPU ενός PLC διαθέτει μια εσωτερική μνήμη η οποία χωρίζεται σε τρεις γενικές κατηγορίες,

τη μνήμη φόρτωσης (load memory), τη μνήμη εργασίας (work memory) και τη μνήμη συστήματος (system memory).



Σχήμα 3.1: Η δομή του SIMATIC S7-300 PLC.

Στη μνήμη φόρτωσης αποθηκεύεται το πρόγραμμα του χρήστη συμπεριλαμβανομένων των δεδομένων διαμόρφωσης του συστήματος. Το πρόγραμμα του χρήστη μεταφέρεται πάντα από την συσκευή προγραμματισμού αρχικά στην μνήμη φόρτωσης και στη συνέχεια από εκεί στην μνήμη εργασίας. Η μνήμη φόρτωσης σε πολλές περιπτώσεις πλέον βρίσκεται στην κάρτα μνήμης (MMC) της CPU, πράγμα που σημαίνει πως είναι απαραίτητη για την λειτουργία της CPU. Η κάρτα μνήμης έχει την δυνατότητα διαγραφής, επανεγγραφής και διατήρησης των δεδομένων χωρίς τροφοδοσία.

Στην μνήμη εργασίας μεταφέρεται από την μνήμη φόρτωσης το πρόγραμμα του χρήστη και εκτελείται. Η μνήμη εργασίας βρίσκεται στη μνήμη RAM (Random Access Memory) υψηλής ταχύτητας, που είναι ενσωματωμένη στη CPU.

Στη μνήμη συστήματος που είναι και αυτή ενσωματωμένη στη CPU περιέχονται ομαδοποιημένες οι μεταβλητές που χρησιμοποιούνται στο πρόγραμμα που εκτελείται κάθε φορά. Οι περιοχές (ομάδες) που χωρίζεται η μνήμη συστήματος είναι:

- PII (Process Image Input). Αποθηκεύονται οι τιμές των εισόδων που διαβάζει η CPU στην αρχή κάθε κύκλου λειτουργίας από τις διαθέσιμες μονάδες εισόδων.
- PIQ (Process Image Output). Κατά τη διάρκεια της εκτέλεσής του, το πρόγραμμα υπολογίζει τις τιμές εξόδου και τις τοποθετεί σε αυτή την περιοχή. Στο τέλος του κύκλου λειτουργίας, η CPU στέλνει τις τιμές αυτές στις διαθέσιμες μονάδες εξόδου.
- M (Memory bit – Μνήμη βοηθητικών). Αποθηκεύονται βοηθητικά δεδομένα και ενδιάμεσα αποτελέσματα του προγράμματος που εκτελείται.
- T (Timers – Χρονικά). Αποθηκεύονται τα δεδομένα των χρονικών που χρησιμοποιούνται από το πρόγραμμα που εκτελείται.
- C (Counters – Απαριθμητές). Αποθηκεύονται τα δεδομένα των απαριθμητών που χρησιμοποιούνται από το πρόγραμμα που εκτελείται.
- L (Temporary Local Data – Μνήμη Προσωρινών Τοπικών Δεδομένων). Αποθηκεύονται προσωρινά δεδομένα που σχετίζονται με την εκτέλεση ενός μπλοκ κώδικα.

Η μονάδα τροφοδοσίας μετατρέπει την τάση του δικτύου σε τάση 24V DC για την τροφοδοσία της CPU καθώς και ορισμένων από τις διαθέσιμες μονάδες. Η έξοδος της μονάδας τροφοδοσίας είναι σταθεροποιημένη και προστατευμένη τόσο από ανοιχτοκυκλώματα όσο και από βραχυκυκλώματα.

Οι μονάδες εισόδων/εξόδων αποτελούν τις μονάδες επικοινωνίας της CPU με τον έξω κόσμο, δηλαδή με τα αισθητήρια και τους ενεργοποιητές. Οι μονάδες εισόδων/εξόδων αναλαμβάνουν να προσαρμόσουν τα σήματα εισόδου και εξόδου που υπάρχουν στον αυτοματισμό, σε σήματα που μπορεί να δεχτεί η CPU, τόσο από άποψη τάσεων όσο και από άποψη ρευμάτων. Η προσαρμογή αυτή γίνεται με τη χρήση ηλεκτρονικών στοιχείων. Οι μονάδες εισόδων/εξόδων χωρίζονται σε ψηφιακές και αναλογικές.

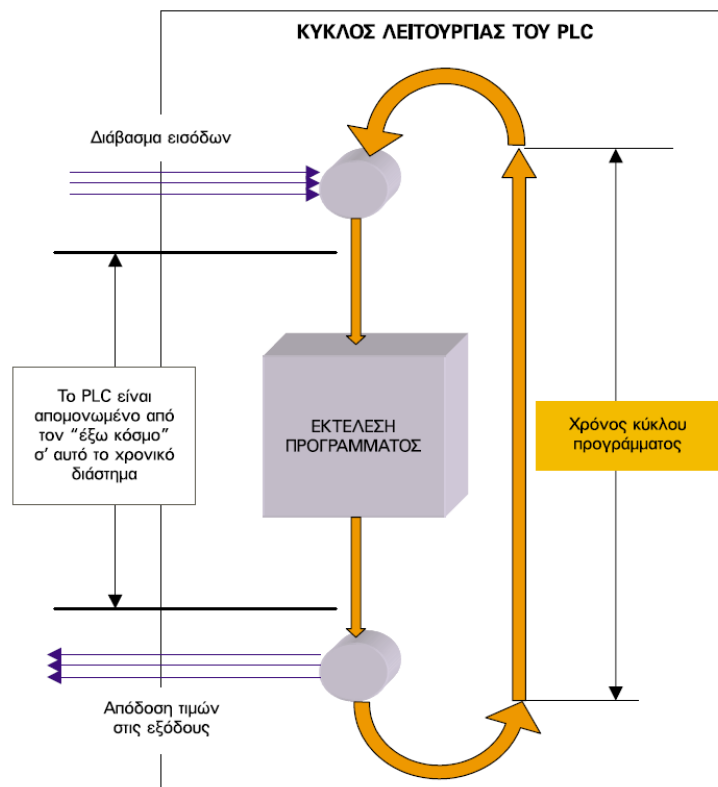
3.2.2 Αρχή λειτουργίας PLC

Τα βήματα που ακολουθεί ένα PLC όταν βρίσκεται σε κατάσταση λειτουργίας (RUN) είναι τα εξής:

Βήμα 1ο. Στην αρχή η CPU διαβάζει τις μονάδες εισόδου. Αυτό σημαίνει ότι για κάθε είσοδο (περίπτωση ψηφιακών εισόδων) ελέγχει αν έχει υψηλή τάση (λογικό '1') ή χαμηλή τάση (λογικό '0'). Η τιμή '0' ή '1' για κάθε είσοδο αποθηκεύεται στην PII.

Βήμα 2ο. Στη συνέχεια η CPU χρησιμοποιώντας σαν δεδομένα τις τιμές των εισόδων, που αποθηκεύτηκαν στην PII, εκτελεί τις εντολές του προγράμματος που υλοποιεί τον επιθυμητό αυτοματισμό. Η εκτέλεση του προγράμματος έχει ως αποτελέσματα τον υπολογισμό των τιμών των εξόδων οι οποίες αποθηκεύονται στην PIQ.

Βήμα 3ο. Στην συνέχεια η CPU αποδίδει τις τιμές που αποθηκεύτηκαν κατά την εκτέλεση του προγράμματος στην PIQ στις μονάδες εξόδου. Αυτό σημαίνει ότι στην περίπτωση των



Σχήμα 3.2: Κύκλος λειτουργίας PLC.

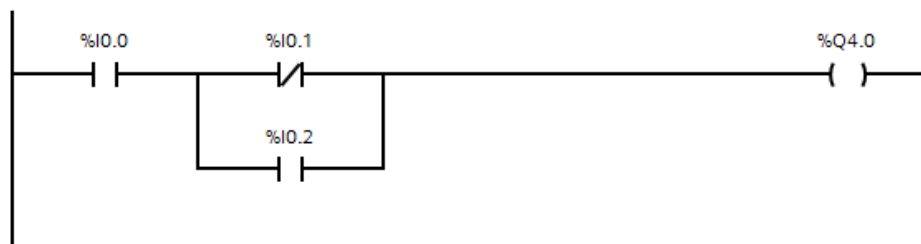
ψηφιακών εξόδων θα δοθεί υψηλή τάση σε όποια έξοδο έχει '1' και θα δοθεί χαμηλή τάση σε όποια έξοδο έχει '0'.

Με την συμπλήρωση του 3ου βήματος συμπληρώνεται ένας πλήρης κύκλος λειτουργίας (Σχήμα 3.2) και η διαδικασία αρχίζει από την αρχή. Ο κύκλος λειτουργίας εκτελείται συνεχώς όσο το PLC βρίσκεται σε κατάσταση RUN.

3.2.3 Γλώσσες προγραμματισμού PLC

Ο προγραμματισμός των PLCs πραγματοποιείται με διάφορες γλώσσες προγραμματισμού. Οι γλώσσες προγραμματισμού που έχουν επικρατήσει διεθνώς είναι η γλώσσα LAD (Ladder logic), η γλώσσα STL (SStatement List) και η γλώσσα FBD (Function Block Diagram).

Η γλώσσα LAD (Σχήμα 3.3) ή γλώσσα ηλεκτρολογικών γραφικών στην ουσία επιτρέπει τη μεταφορά του ηλεκτρολογικού σχεδίου στο PLC. Με τη γλώσσα αυτή η εργασία σχεδιασμού του αυτοματισμού γίνεται εύκολα και γρήγορα. Η γλώσσα LAD χρησιμοποιεί όχι την Ευρωπαϊκή τυποποίηση στο σχεδιασμό των ηλεκτρικών επαφών, αλλά την Αμερικάνικη. Αυτό ίσως οφείλεται στο γεγονός ότι τα πρώτα PLC αναπτύχθηκαν στην Αμερική. Όμως στη συνέχεια αυτός ο τρόπος σχεδιασμού θεωρήθηκε εύχρηστος και έτσι διατηρήθηκε και από τις Ευρωπαϊκές εταιρίες, με αποτέλεσμα σήμερα να είναι ο πλέον διαδεδομένος.



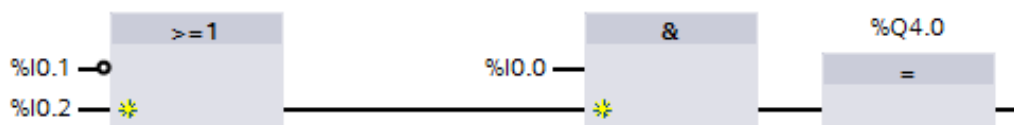
Σχήμα 3.3: Η Γλώσσα LAD.

Η γλώσσα STL ή γλώσσα λογικών εντολών (Σχήμα 3.4) αναπτύχθηκε σχεδόν ταυτόχρονα με τη γλώσσα LAD. Η γλώσσα αυτή δημιουργεί λίστα προγράμματος με εντολές, οι οποίες αντιστοιχούν στις λογικές πύλες (AND, OR, NOT κλπ). Στην αρχή η γλώσσα STL περιοριζόταν μόνο στις βασικές λογικές εντολές. Σήμερα η γλώσσα αυτή έχει εξελιχθεί πάρα πολύ και συναντά κανείς σε αυτή στοιχεία από τις γλώσσες των υπολογιστών και κυρίως της γλώσσας Assembly.

```
A      %I0.0
A (
ON     %I0.1
O      %I0.2
)
=      %Q4.0
```

Σχήμα 3.4: Η Γλώσσα STL.

Η γλώσσα FBD ή λογικού διαγράμματος (Σχήμα 3.5) είναι επίσης γραφική, αλλά αντί του ηλεκτρολογικού σχεδίου του αυτοματισμού χρησιμοποιεί το αντίστοιχο λογικό κύκλωμα. Η γλώσσα αυτή είναι νεότερη και δεν χρησιμοποιείται από όλες τις εταιρείες.



Σχήμα 3.5: Η Γλώσσα FBD.

3.2.4 Τύποι διαθέσιμων block

Το πρόγραμμα του χρήστη μπορεί να χωριστεί σε επιμέρους τμήματα που ονομάζονται block (μπλοκ). Κάθε μπλοκ αποτελεί ένα μέρος του προγράμματος του χρήστη που καθορίζεται από την λειτουργία του, τη δομή του και την εφαρμογή του.

Μπλοκ χρήστη

Για την υλοποίηση μεγάλων και περίπλοκων προγραμμάτων ο χρήστης μπορεί να διαλέξει μεταξύ των διάφορων μπλοκ χρήστη, ανάλογα με την εφαρμογή:

- Μπλοκ οργάνωσης, OB (Organization Block).

Τα μπλοκ οργάνωσης ή OBs έχουν τον ρόλο του διαμεσολαβητή μεταξύ του λειτουργικού συστήματος και του προγράμματος του χρήστη. Κατά την εκδήλωση κάποιων ειδικών γεγονότων (π.χ. κάποιο σφάλμα) το λειτουργικό σύστημα της CPU ανάλογα το γεγονός καλεί το αντίστοιχο OB. Το σημαντικότερο από όλα τα OBs είναι το OB1, το οποίο η CPU καλεί αυτόματα και το εκτελεί κυκλικά. Μέσα σε αυτό το μπλοκ βρίσκεται το πρόγραμμα του χρήστη. Ένα επίσης σημαντικό μπλοκ είναι το OB100 που εκτελείται μόνο μια φορά όταν τροφοδοτείται με τάση το σύστημα. Τα OBs έχουν συγκεκριμένες προτεραιότητες εκτέλεσης που αναπαρίστανται με ένα αριθμό με τιμές από 0 ως 29. Αν ένα μπλοκ έχει μεγαλύτερη προτεραιότητα από κάποιο άλλο, τότε μπορεί να το διακόψει και να εκτελεστεί το ίδιο. Για παράδειγμα το OB1 που έχει προτεραιότητα 1, μπορεί να διακοπεί από όλα τα άλλα OBs με μεγαλύτερη προτεραιότητα.

- Συνάρτηση, FC (Function).

Οι συναρτήσεις ή FCs είναι μπλοκ κώδικα χωρίς μνήμη, τα οποία προγραμματίζονται από τον χρήστη. Χρησιμοποιούνται για υπολογισμό κάποιας συνάρτησης ή για την υλοποίηση μιας συγκεκριμένης εργασίας. Τα FCs είναι παραμετροποιήσιμα και για τον λόγο αυτό χρησιμοποιούνται επίσης στον προγραμματισμό επαναλαμβανόμενων λειτουργιών αυτοματισμού.

- Μπλοκ συνάρτησης, FB (Function Block).

Τα μπλοκ συναρτήσεων ή FBs είναι μπλοκ κώδικα με μνήμη, τα οποία προγραμματίζονται από τον χρήστη. Σε κάθε FB διατίθεται ένα instance DB σαν δική του μνήμη όπου αποθηκεύονται τα δεδομένα του. Τα FBs είναι παραμετροποιήσιμα όπως και τα FCs, επομένως και αυτά χρησιμοποιούνται σε περιπτώσεις επαναλαμβανόμενων λειτουργιών αυτοματισμού.

- Μπλοκ δεδομένων, DB (Data Block).

Τα μπλοκ δεδομένων ή DBs δεν περιέχουν κώδικα, αλλά περιέχουν δεδομένα του προγράμματος του χρήστη. Υπάρχουν δύο τρόποι χρησιμοποίησης των μπλοκ δεδομένων ως global DB και ως instance DB. Ένα global DB είναι ένα ελεύθερο μπλοκ δεδομένων στο πρόγραμμα του χρήστη για κοινή χρήση, ενώ αντίθετα ένα instance DB διατίθεται σε ένα συγκεκριμένο FB.

Μπλοκ συστήματος (System Block)

Τα μπλοκ συστήματος αποτελούν μέρος του λειτουργικού συστήματος και περιέχουν προγράμματα ή δεδομένα. Τα μπλοκ συστήματος χωρίζονται σε SFCs (System Functions), SFBs (System Function Blocks) και SDBs (System Data Blocks). Τα μπλοκ συστήματος παρέχουν στον χρήστη πρόσβαση σε σημαντικές λειτουργίες του συστήματος, όπως είναι η διαχείριση του εσωτερικού ρολογιού της CPU, οι διάφορες λειτουργίες επικοινωνίας κλπ. Ο χρήστης μπορεί να καλέσει τα SFCs και τα SFBs αλλά δεν μπορεί να τα τροποποιήσει. Τα μπλοκ αυτά δεν δεσμεύουν χώρο από στην μνήμη χρήστη που βρίσκεται στην μνήμη συστήματος. Στην μνήμη χρήστη αποθηκεύονται μόνο τα μπλοκ συστήματος και τα αντίστοιχα DBs που καλούνται από το πρόγραμμα του χρήστη. Τα SDBs περιέχουν χρήσιμες πληροφορίες και δεδομένα, π.χ. για την διαμόρφωση του συστήματος αυτοματισμού, η για την παραμετροποίηση των μονάδων του PLC συστήματος κλπ.

Στάνταρτ μπλοκ (Standard Block)

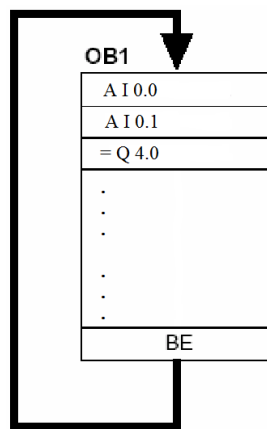
Τα Standard Block είναι έτοιμα μπλοκ που ενσωματώνουν διάφορες λειτουργίες, συναρτήσεις και προσφέρουν στον χρήστη έτοιμες λύσεις ώστε να μην χρειαστεί να τα υλοποιήσει μόνος του. Τα Standard Blocks συμπεριφέρονται σαν τα μπλοκ χρήστη και απαιτούν χώρο στη μνήμη χρήστη.

3.2.5 Δομή προγράμματος

Υπάρχουν δυο τρόποι προγραμματισμού των PLCs, ο γραμμικός προγραμματισμός και ο δομημένος προγραμματισμός.

- **Γραμμικός προγραμματισμός (Linear Programming)**

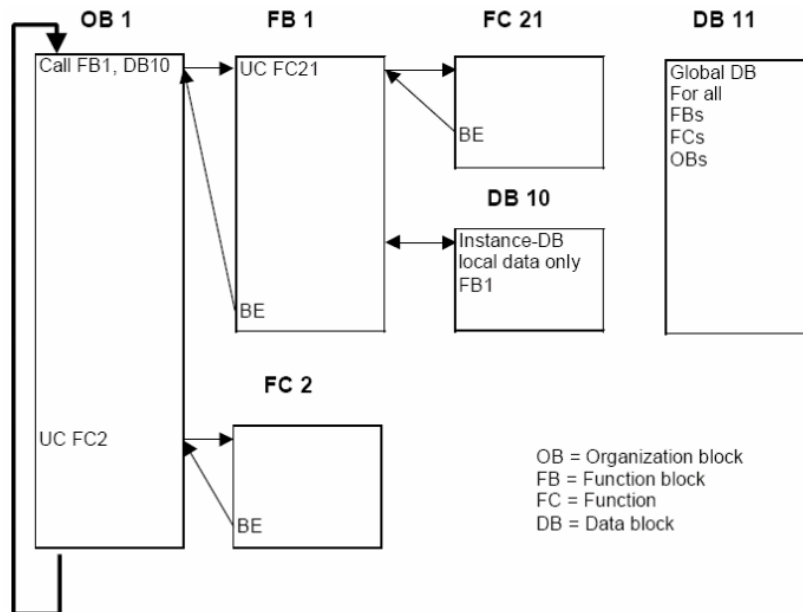
Στο γραμμικό προγραμματισμό (Σχήμα 3.6) όλο το πρόγραμμα του χρήστη βρίσκεται στο μπλοκ οργάνωσης OB1. Η CPU επεξεργάζεται τις εντολές την μια μετά την άλλη μέχρι το τέλος του μπλοκ και ξαναρχίζει η ίδια διαδικασία πάλι από την αρχή. Ο γραμμικός προγραμματισμός είναι απλός, εύκολος και γρήγορος. Παρόλα αυτά είναι δύσχρηστος για μεγάλα και πολύπλοκα προγράμματα αφού αφενός αυξάνονται οι πιθανότητες προγραμματιστικών σφαλμάτων και αφετέρου καθίσταται δύσκολη η ανίχνευση των παραπάνω σφαλμάτων. Για τους παραπάνω λόγους ο γραμμικός προγραμματισμός χρησιμοποιείται κυρίως στην υλοποίηση μικρών εφαρμογών.



Σχήμα 3.6: Παράδειγμα γραμμικού προγραμματισμού.

- **Δομημένος προγραμματισμός (Structured Programming)**

Ο δομημένος προγραμματισμός (Σχήμα 3.7) χρησιμοποιείται σε μεγάλες και περίπλοκες εφαρμογές. Το πρόγραμμα χωρίζεται σε τμήματα που υλοποιούνται είτε σε παραμετροποιήσιμα μπλοκ κώδικα (FCs ή FBs) τα οποία αναλαμβάνουν την υλοποίηση επαναλαμβανόμενων διεργασιών δίνοντας κάθε φορά τις αντίστοιχες παραμέτρους, είτε σε μη παραμετροποιήσιμα μπλοκ κώδικα (FCs ή FBs) που υλοποιούν συγκεκριμένες εργασίες. Με αυτό τον τρόπο είναι εφικτή η ανάπτυξη πολύπλοκων προγραμμάτων, τα οποία αποτελούνται από επιμέρους μπλοκ κώδικα που είναι απλούστερα και ευανάγνωστα.



Σχήμα 3.7: Παράδειγμα δομημένου προγραμματισμού.

3.3 Τα συστήματα SCADA

Τα συστήματα SCADA αποτελούν εφαρμογή της βιομηχανικής πληροφορικής για την εποπτεία της παραγωγής. Με χρήση των συστημάτων SCADA επιτυγχάνεται η online παρακολούθηση, μέσω μονάδων PLCs, και η συνεχής καταγραφή σε ηλεκτρονικούς υπολογιστές όλων των κρίσιμων παραμέτρων της παραγωγικής διαδικασίας για την επίτευξη εποπτείας σε πραγματικό χρόνο.

Ένα σύστημα SCADA περιλαμβάνει την απεικόνιση σε μιμητικά διαγράμματα όλων των διεργασιών παραγωγής, ενδείξεις των τιμών των μετρούμενων μεγεθών, διαρκή συλλογή και αποθήκευση δεδομένων σε Η/Υ, γνωστοποίηση σφαλμάτων κ.α. Επίσης, σημαντική είναι η δυνατότητα παρακολούθησης του συστήματος μέσω κατάλληλα διαμορφωμένων ιστοσελίδων.

Οι κύριες λειτουργίες ενός συστήματος SCADA είναι οι ακόλουθες:

- Συλλογή δεδομένων από τα PLCs και τις RTUs (Remote Terminal Unit). Όλα τα επιθυμητά σήματα μεταδίδονται προς το σύστημα SCADA μέσω του δικτύου βιομηχανικού αυτοματισμού.
- Αποθήκευση των πληροφοριών στη βάση δεδομένων και αναπαράστασή τους μέσω γραφημάτων. Οι επιλεγμένες πληροφορίες αναπαρίστανται είτε αυτούσιες είτε έπειτα από κατάλληλη επεξεργασία.

- Ανάλυση δεδομένων και ειδοποίηση του προσωπικού σε περιπτώσεις σφάλματος. Όταν τα δεδομένα πάρουν τιμές μη κανονικές το σύστημα SCADA ειδοποιεί με οπτική ή ακουστική σήμανση τους χειριστές, ώστε να αποφευχθούν δυσάρεστες επιπτώσεις.
- Έλεγχος κλειστού βρόχου διεργασιών. Υπάρχει η δυνατότητα εφαρμογής τεχνικών ελέγχου, αυτόματες ή χειροκίνητες.
- Γραφική απεικόνιση των τμημάτων της διεργασίας σε μιμητικά διαγράμματα και παρουσιάσεις των δεδομένων σε ενεργά πεδία. Τα μιμητικά διαγράμματα απεικονίζουν ρεαλιστικά τμήματα της διεργασίας με στόχο την ευκολότερη εποπτεία και την κατανόηση των δεδομένων από τους χειριστές του συστήματος.
- Καταγραφή όλων των συμβάντων κανονικών ή μη για την δημιουργία ιστορικού αρχείου. Σε κάθε βιομηχανία υπάρχει καταγραφή όλων των κρίσιμων παραμέτρων. Παλιότερα γινόταν με χειρόγραφο καταγραφή, ενώ σήμερα την ευθύνη αυτή έχει αναλάβει η βάση δεδομένων του συστήματος SCADA.
- Υποστήριξη διπλού υπολογιστικού συστήματος με αυτόματη εναλλαγή, αν αυτό κρίνεται σκόπιμο βάση της υπό έλεγχο διεργασίας. Σε διεργασίες υψηλής επικινδυνότητας πρέπει να ελαχιστοποιηθεί όσο το δυνατόν περισσότερο η εμφάνιση σφάλματος λόγω βλάβης του εξοπλισμού. Για τον λόγο αυτό τα συστήματα SCADA υποστηρίζουν δεύτερο υπολογιστικό σύστημα που αναλαμβάνει σε περίπτωση σφάλματος.
- Μεταφορά δεδομένων σε άλλα τμήματα του κεντρικού συστήματος πληροφόρησης και διαχείρισης.
- Έλεγχος της πρόσβασης χειριστών στα διάφορα υποσυστήματα του συστήματος SCADA.
- Ειδικές εφαρμογές λογισμικού όπως εκτέλεση κώδικα C++ ή ανάπτυξη ευφώνων συστημάτων.

3.3.1 Τα συστήματα HMI

Τα συστήματα HMI αποτελούν το μέρος της λειτουργίας των συστημάτων SCADA που αλληλεπιδρά με τον τελικό χειριστή. Με τα συστήματα HMI πραγματοποιείται η διασύνδεση του χειριστή με την ελεγχόμενη διεργασία μέσω των PLCs. Επομένως, το PLC αποτελεί την μονάδα που ελέγχει την διεργασία και το σύστημα HMI αποτελεί την μονάδα που διασυνδέει το PLC με τον χειριστή.

Με το HMI υπάρχει η δυνατότητα πραγματοποίησης πολλών ενεργειών, μεταξύ των οποίων είναι:

- Απεικόνιση – εποπτεία της διεργασίας.
- Έλεγχος και διαχείριση της διεργασίας από το χειριστή.
- Απεικόνιση προειδοποιητικών μηνυμάτων – σφαλμάτων.
- Καταγραφή και δημιουργία αρχείου δεδομένων και σφαλμάτων.
- Εκτύπωση των αρχείων δεδομένων και σφαλμάτων.

3.4 Το HMI της εφαρμογής

Στην παρούσα πτυχιακή εργασία η συσκευή που χρησιμοποιείται ως HMI είναι ο ηλεκτρονικός υπολογιστής. Το HMI που δημιουργήθηκε για τον έλεγχο της εφαρμογής αποτελείται από τρεις οθόνες. Την αρχική οθόνη (home_screen) από την οποία ο χρήστης επιλέγει λειτουργία (auto ή manual). Την οθόνη της αυτόματης λειτουργίας (auto_screen) και την οθόνη της χειροκίνητης λειτουργίας (manual_screen).

Οι οθόνες του HMI και τα αντικείμενα (μπουτόν, κουμπιά, διακόπτες, φωτεινές ενδείξεις κλπ.) τους έχουν διαμορφωθεί έτσι, ώστε ο χρήστης να μπορεί να ελέγχει τον εικονικό ρομποτικό βραχίονα ελέγχοντας τις λειτουργίες του κώδικα του PLC. Το παραπάνω επιτυγχάνεται συνδέοντας τα αντικείμενα των οθονών του HMI με τα tags των μεταβλητών του κώδικα του PLC που χρειάζονται.

Τα tags είναι οι συμβολικές διευθύνσεις (ονόματα) των μεταβλητών του κώδικα του PLC. Οι μεταβλητές δημιουργούνται από τον χρήστη κατά την σύνταξη του κώδικα. Κάθε μεταβλητή αποτελείται από:

- Την συμβολική διεύθυνση. Ο χρήστης μπορεί να αλλάξει τα προεπιλεγμένα ονόματα των μεταβλητών με ονόματα σχετικά με την λειτουργία τους. Για παράδειγμα στην παρούσα πτυχιακή εργασία η μεταβλητή με συμβολική διεύθυνση reset και απόλυτη διεύθυνση M 1.0 ενεργοποιεί την λειτουργία reset που επαναφέρει τον εικονικό ρομποτικό βραχίονα στην αρχική του θέση. Με αυτόν τον τρόπο το πρόγραμμα του PLC γίνεται ευανάγνωστο επιτρέποντας ταυτόχρονα την ευκολότερη σύνταξη ή τροποποίηση του προγράμματος. Επίσης, πραγματοποιείται ευκολότερα η διασύνδεση με τα αντίστοιχα αντικείμενα του HMI.
- Την απόλυτη διεύθυνση. Η οποία αποτελείται από τον τελεστή που αντιπροσωπεύει την περιοχή μνήμης και την διεύθυνση, π.χ. I 0.1 (το bit 1 του byte 0 των εισόδων), Q 4.2 (το bit 2 του byte 4 των εξόδων), M 10.3 (το bit 3 του byte 10 της βοηθητικής μνήμης), QB 8 (το byte 8 των εξόδων), MW100 (η word 100 της βοηθητικής μνήμης), T1 (το χρονικό 1), C2 (ο απαριθμητής 2) κλπ.
- Και από τον τύπο δεδομένων, π.χ. BOOL (1 bit, '0' ή '1'), BYTE (8 bits, 16#00 έως 16#FF), WORD (16 bits, 16#0000 έως 16#FFFF), INT (16 bits, -32768 έως +32767), REAL (32 bits, $\pm 1.18 \times 10^{-38}$ έως $\pm 3.40 \times 10^{38}$), S5TIME (16 bits, S5T#0ms έως S5TIME#2h46m30s).

Στον Πίνακα 3.1 φαίνονται τα PLC tags που χρησιμοποιήθηκαν για την διασύνδεση των μεταβλητών του PLC με τα αντίστοιχα αντικείμενα των οθονών του HMI της συγκεκριμένης εφαρμογής.

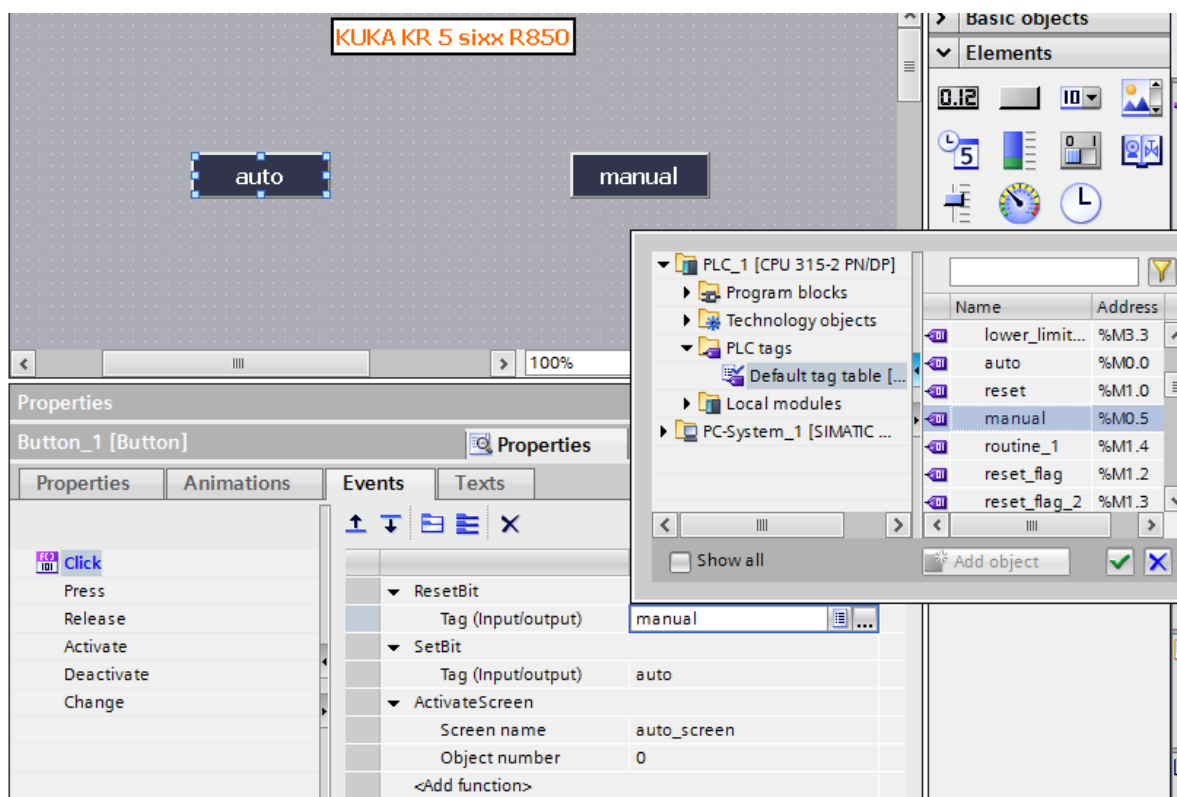
Name	PLC tag	Data Type	Address
acceleration_1	acceleration_1	Int	%MW110
acceleration_2	acceleration_2	Int	%MW210
acceleration_3	acceleration_3	Int	%MW310
acceleration_4	acceleration_4	Int	%MW410
acceleration_5	acceleration_5	Int	%MW510
acceleration_6	acceleration_6	Int	%MW610
auto	auto	Bool	%M0.0
continue	continue	Bool	%M1.7
current_position_1	current_position_1	Int	%MW100
current_position_2	current_position_2	Int	%MW200
current_position_3	current_position_3	Int	%MW300
current_position_4	current_position_4	Int	%MW400
current_position_5	current_position_5	Int	%MW500
current_position_6	current_position_6	Int	%MW600
demo	demo	Bool	%M5.3
done	done	Bool	%M0.2
done_indicator	done_indicator	Bool	%M5.2
execute	execute	Bool	%M0.3
execute_flag	execute_flag	Bool	%M5.0
grab_object_sensor	grab_object_sensor	Bool	%M0.4
lower_limit_1	lower_limit_1	Bool	%M2.1
lower_limit_2	lower_limit_2	Bool	%M2.3
lower_limit_3	lower_limit_3	Bool	%M2.5
lower_limit_4	lower_limit_4	Bool	%M2.7
lower_limit_5	lower_limit_5	Bool	%M3.1
lower_limit_6	lower_limit_6	Bool	%M3.3
manual	manual	Bool	%M0.5
manual_acc_1	manual_acc_1	Int	%MW125
manual_acc_2	manual_acc_2	Int	%MW225
manual_acc_3	manual_acc_3	Int	%MW325
manual_acc_4	manual_acc_4	Int	%MW425
manual_acc_5	manual_acc_5	Int	%MW525
manual_acc_6	manual_acc_6	Int	%MW625
manual_execute	manual_execute	Bool	%M0.6
manual_position_1	manual_position_1	Int	%MW15
manual_position_2	manual_position_2	Int	%MW25
manual_position_3	manual_position_3	Int	%MW35
manual_position_4	manual_position_4	Int	%MW45
manual_position_5	manual_position_5	Int	%MW55
manual_position_6	manual_position_6	Int	%MW65
manual_smo_1	manual_smo_1	Real	%MD1005
manual_smo_2	manual_smo_2	Real	%MD1015
manual_smo_3	manual_smo_3	Real	%MD1025
manual_smo_4	manual_smo_4	Real	%MD1035

Πίνακας 3.1: Τα tags για την διασύνδεση του HMI με το PLC.

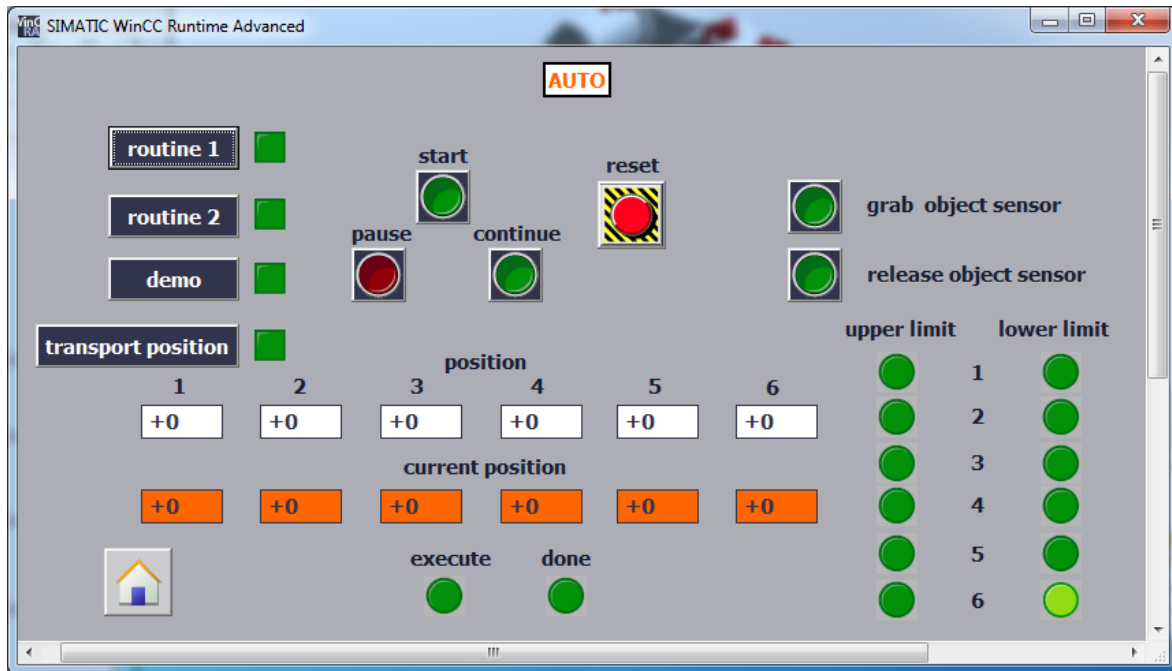
Name	PLC tag	Data Type	Address
manual_smo_5	manual_smo_5	Real	%MD1045
manual_smo_6	manual_smo_6	Real	%MD1055
manual_vel_1	manual_vel_1	Int	%MW120
manual_vel_2	manual_vel_2	Int	%MW220
manual_vel_3	manual_vel_3	Int	%MW320
manual_vel_4	manual_vel_4	Int	%MW420
manual_vel_5	manual_vel_5	Int	%MW520
manual_vel_6	manual_vel_6	Int	%MW620
parameter_switch_1	parameter_switch_1	Bool	%M6.7
parameter_switch_2	parameter_switch_2	Bool	%M7.0
parameter_switch_3	parameter_switch_3	Bool	%M7.1
parameter_switch_4	parameter_switch_4	Bool	%M7.2
parameter_switch_5	parameter_switch_5	Bool	%M7.3
parameter_switch_6	parameter_switch_6	Bool	%M7.4
position_1	position_1	Int	%MW10
position_2	position_2	Int	%MW20
position_3	position_3	Int	%MW30
position_4	position_4	Int	%MW40
position_5	position_5	Int	%MW50
position_6	position_6	Int	%MW60
release_object_sensor	release_object_sensor	Bool	%M0.7
reset	reset	Bool	%M1.0
routine_1	routine_1	Bool	%M1.4
routine_2	routine_2	Bool	%M4.7
smoothing_1	smoothing_1	Real	%MD1000
smoothing_2	smoothing_2	Real	%MD1010
smoothing_3	smoothing_3	Real	%MD1020
smoothing_4	smoothing_4	Real	%MD1030
smoothing_5	smoothing_5	Real	%MD1040
smoothing_6	smoothing_6	Real	%MD1050
start	start	Bool	%M1.5
stop_flag	stop_flag	Bool	%M3.4
transport_position	transport_position	Bool	%M6.6
upper_limit_1	upper_limit_1	Bool	%M2.0
upper_limit_2	upper_limit_2	Bool	%M2.2
upper_limit_3	upper_limit_3	Bool	%M2.4
upper_limit_4	upper_limit_4	Bool	%M2.6
upper_limit_5	upper_limit_5	Bool	%M3.0
upper_limit_6	upper_limit_6	Bool	%M3.2
velocity_1	velocity_1	Int	%MW105
velocity_2	velocity_2	Int	%MW205
velocity_3	velocity_3	Int	%MW305
velocity_4	velocity_4	Int	%MW405
velocity_5	velocity_5	Int	%MW505
velocity_6	velocity_6	Int	%MW605

Πίνακας 3.1: Τα tags για την διασύνδεση του HMI με το PLC (Συνέχεια).

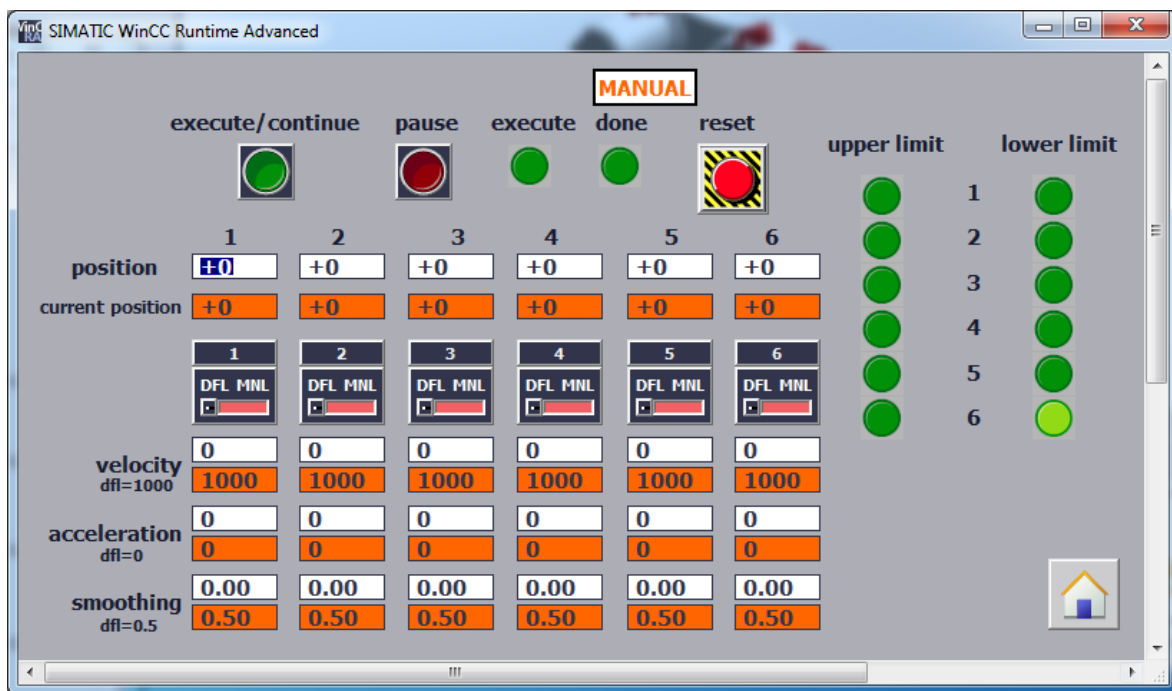
Για την υλοποίηση της αρχικής οθόνης (Σχήμα 3.8) του HMI από την οποία ο χρήστης επιλέγει λειτουργία (auto ή manual) χρησιμοποιήθηκαν δυο κουμπιά γενικής χρήσης από τα βασικά στοιχεία της εργαλειοθήκης του WinCC, ένα για την εισαγωγή στην αυτόματη λειτουργία και ένα για την εισαγωγή στην χειροκίνητη λειτουργία. Η εργαλειοθήκη του WinCC εκτός από τα βασικά στοιχεία περιλαμβάνει βιβλιοθήκες με πλήθος ειδικών στοιχείων για κάθε εφαρμογή, επίσης παρέχει στον χρήστη την δυνατότητα σχεδίασης νέων στοιχείων καθώς και την εισαγωγή ή δημιουργία γραφικών κίνησης (animation) στις HMI οθόνες. Έπειτα ακολούθησε η διαμόρφωση του κάθε στοιχείου ως προς τις λειτουργίες που θα εκτελούνται κατά το πάτημα. Από την καρτέλα events κάθε στοιχείου ο χρήστης εισάγει τις επιθυμητές λειτουργίες και τις συνδέει με τα tags των αντίστοιχων μεταβλητών του PLC κώδικα. Για παράδειγμα το κουμπί auto έχει διαμορφωθεί έτσι, ώστε κάνοντας κλικ πάνω του η μεταβλητή με συμβολική διεύθυνση auto να αποκτά λογικό '1', η μεταβλητή με συμβολική διεύθυνση manual να αποκτά λογικό '0' και να πραγματοποιείται η μετάβαση στην οθόνη της αυτόματης λειτουργίας (auto_screen). Με παρόμοιο τρόπο υλοποιήθηκαν οι οθόνες auto_screen (Σχήμα 3.9) και manual_screen (Σχήμα 3.10).



Σχήμα 3.8: Η διαμόρφωση της αρχικής οθόνης (home_screen).



Σχήμα 3.9: Η οθόνη αυτόματης λειτουργίας (auto_screen).



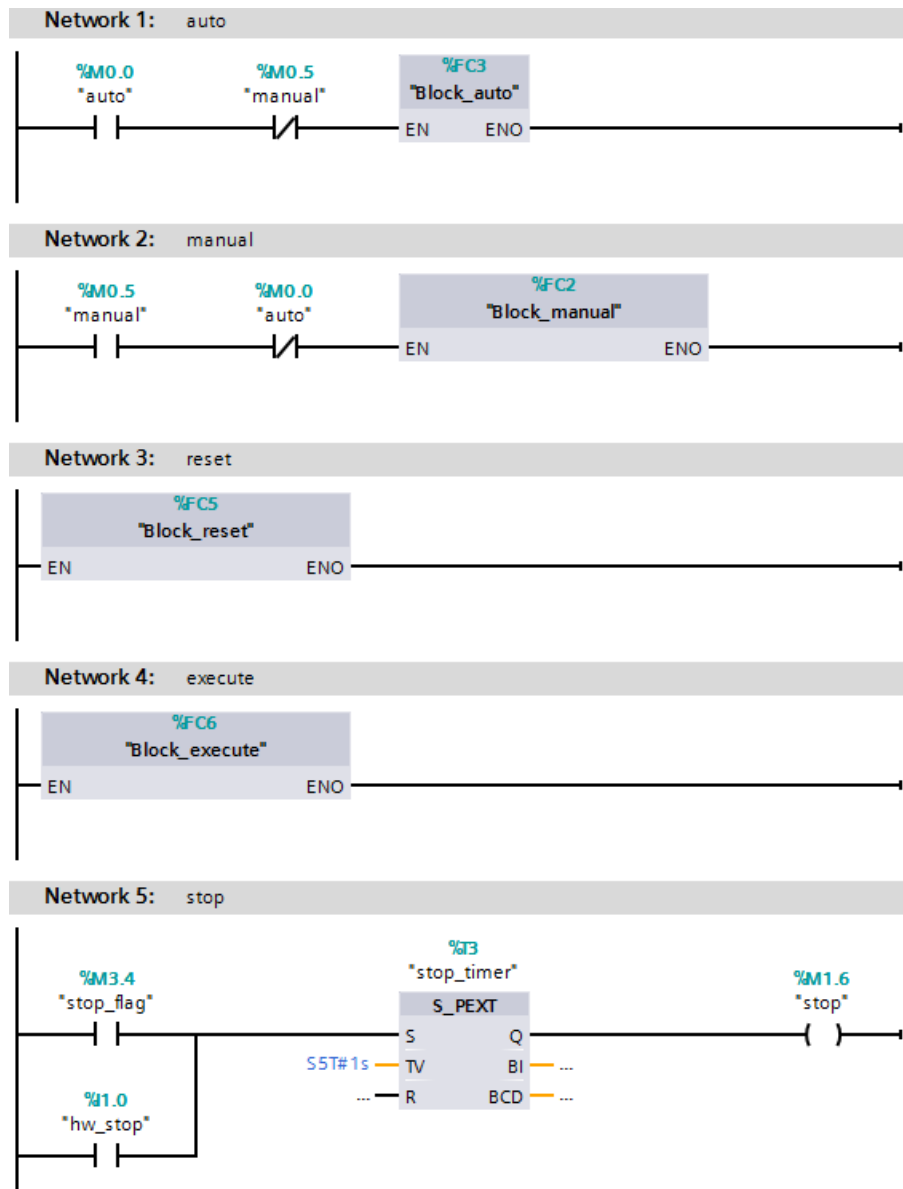
Σχήμα 3.10: Η οθόνη χειροκίνητης λειτουργίας (manual_screen).

3.5 Ο PLC κώδικας της εφαρμογής

Το πρόγραμμα του PLC που δημιουργήθηκε για τον έλεγχο του εικονικού ρομποτικού βραχίονα υποστηρίζει δύο λειτουργίες την αυτόματη (auto) και την χειροκίνητη (manual). Η λειτουργία auto επιτρέπει στον χρήστη να επιλέξει μια από τις τέσσερις υποστηριζόμενες διαδικασίες που περιλαμβάνουν δύο ρουτίνες εργασίας (routine 1 και routine 2), μια ρουτίνα επίδειξης (demo) και την λειτουργία θέσης μεταφοράς (transport position). Η λειτουργία

manual επιτρέπει τον χειροκίνητο έλεγχο του ρομποτικού βραχίονα μέσω των γωνιακών θέσεων των αρθρώσεων και τη θέση της αρπάγης. Επίσης, επιτρέπει την παραμετροποίηση των κινήσεων τόσο των αρθρώσεων όσο και της αρπάγης, ελέγχοντας την ταχύτητα, την επιτάχυνση και την εξομάλυνση.

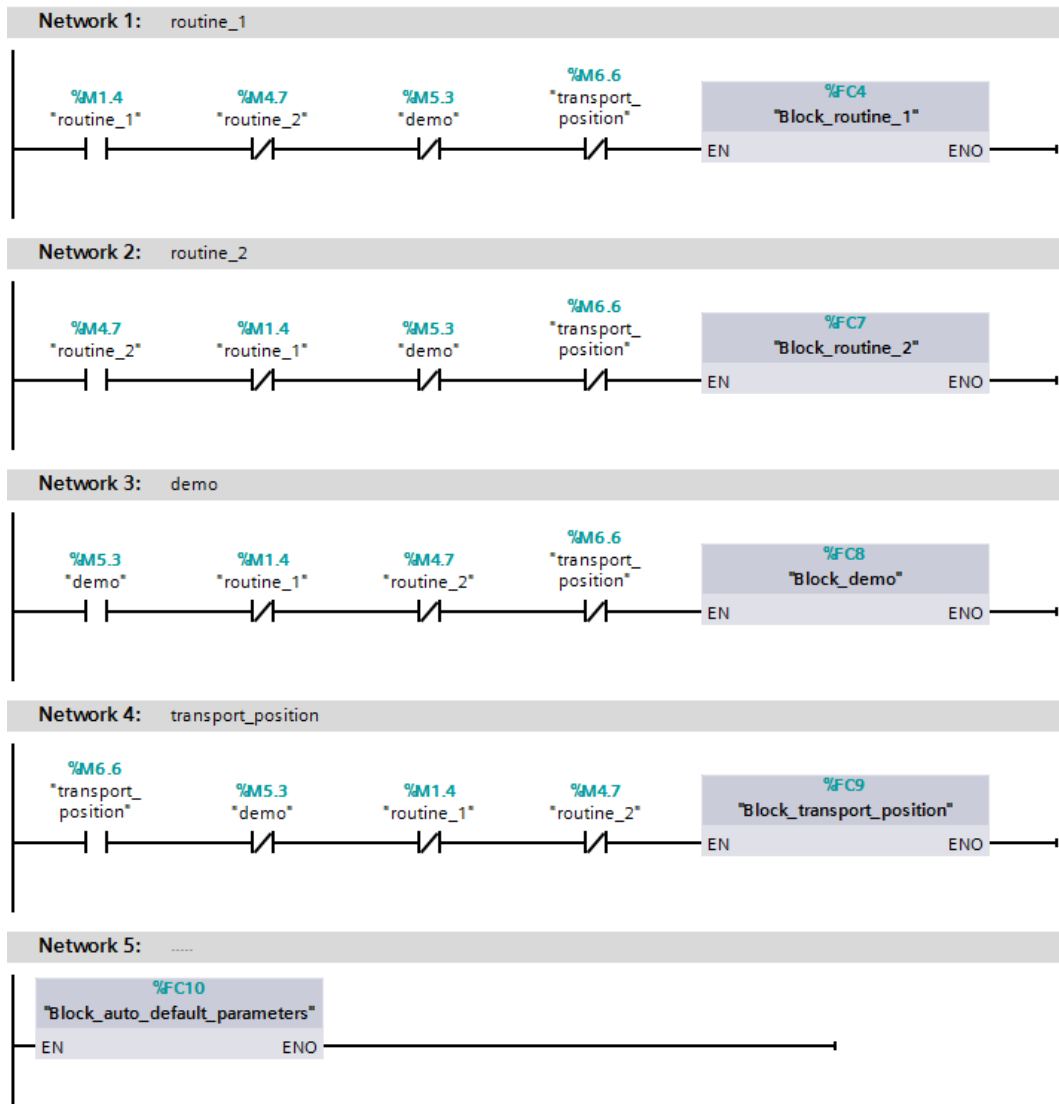
Η υλοποίηση του κώδικα βασίστηκε στον δομημένο προγραμματισμό. Το πρόγραμμα χωρίστηκε σε FCs όπου κάθε ένα από αυτά υλοποιεί μια συγκεκριμένη εργασία, καθώς και σε παραμετροποιήσιμα FCs που ενσωματώνουν λειτουργίες οι οποίες επαναλαμβάνονται στο πρόγραμμα. Στο Πρόγραμμα 3.1 φαίνεται το OB1 που περιέχει το πρόγραμμα του PLC.



Πρόγραμμα 3.1: Το πρόγραμμα του OB1.

3.5.1 Block_auto (FC3)

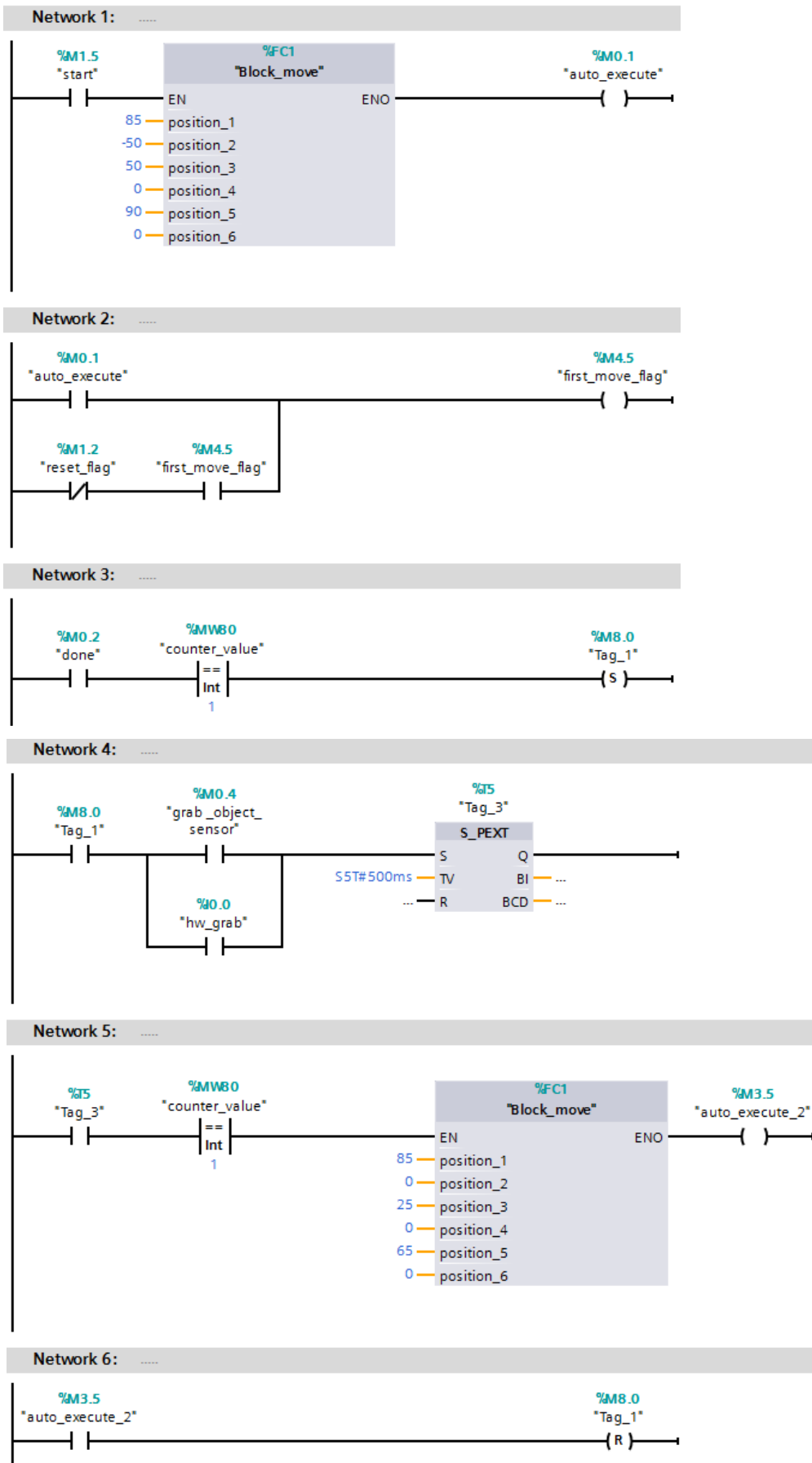
Το Block_auto (FC3) που παρουσιάζεται στο Πρόγραμμα 3.2 περιέχει τον κώδικα της αυτόματης λειτουργίας, ο οποίος χωρίζεται σε πέντε FCs.



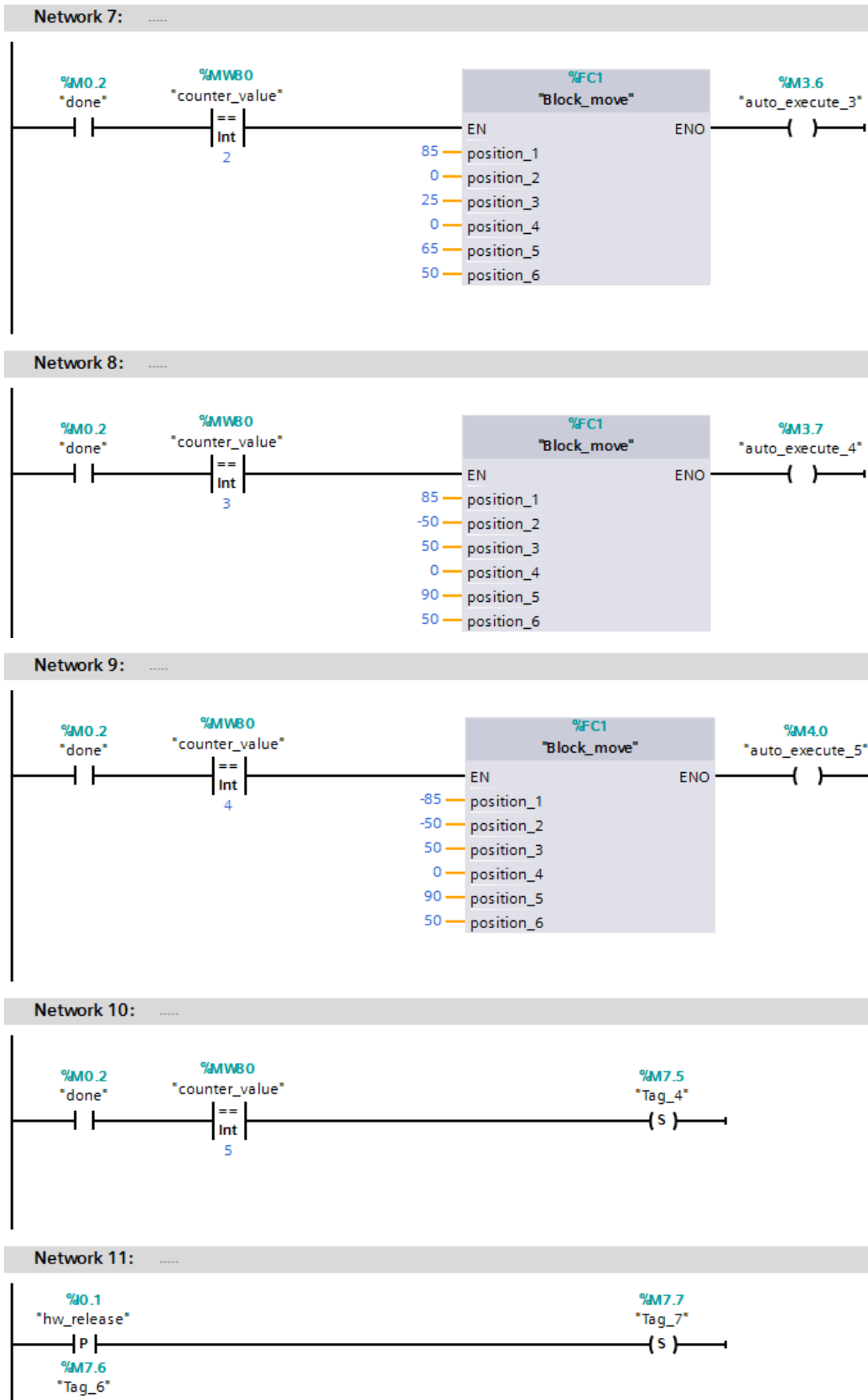
Πρόγραμμα 3.2: Το πρόγραμμα του Block_auto (FC3).

3.5.2 Block_routine_1 (FC4)

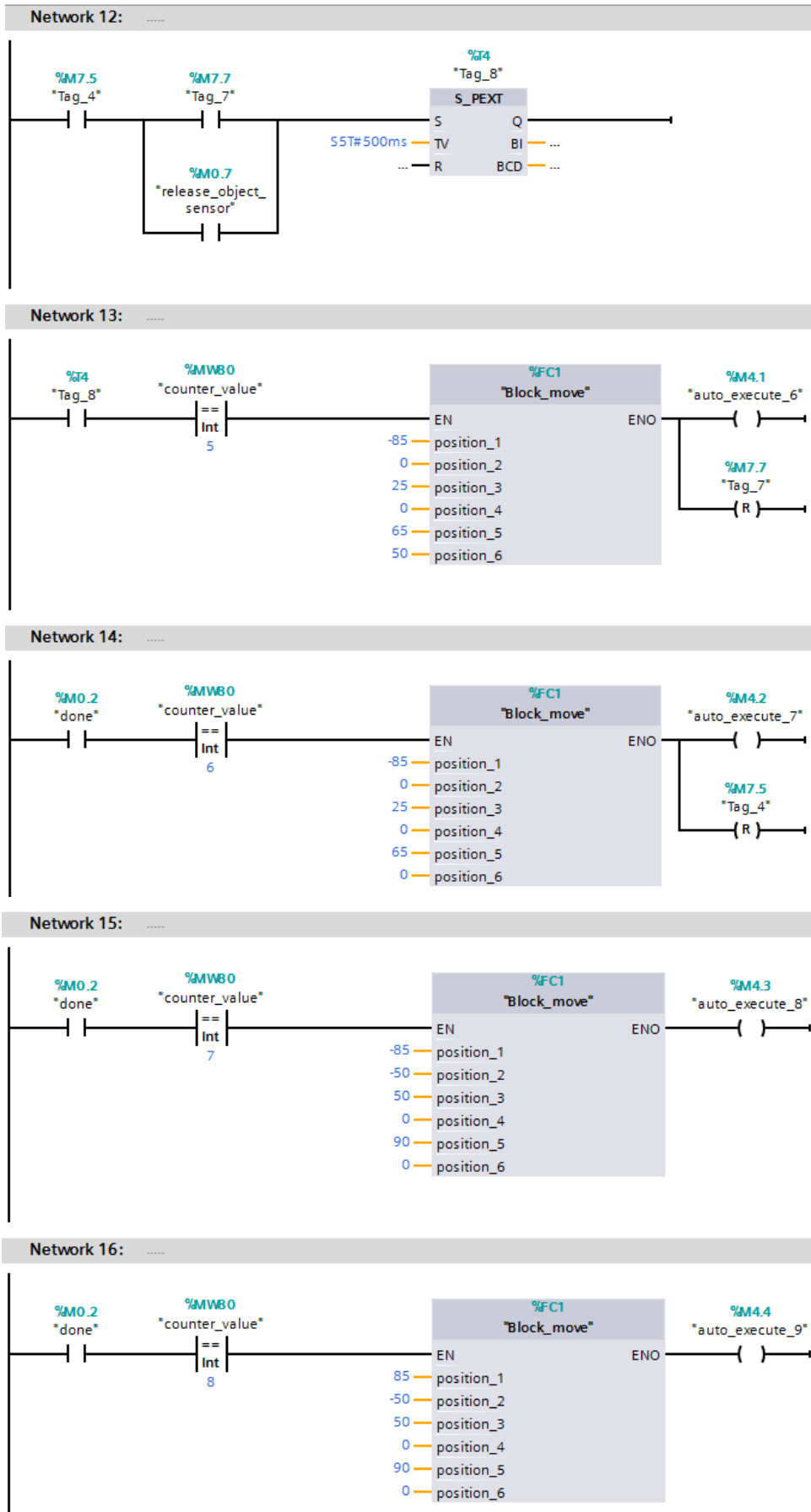
Στο Block_routine_1 (FC4) υπάρχει ο κώδικας της ρουτίνας εργασίας 1 που φαίνεται στο Πρόγραμμα 3.3. Ο κώδικας της ρουτίνας εργασίας 2 και της ρουτίνας επίδειξης είναι παρόμοιος με αυτόν της ρουτίνας εργασίας 1, με σημαντικότερη διαφορά τις διαφορετικές τιμές των γωνιακών θέσεων των αρθρώσεων και της θέσης της αρπάγης σε κάθε κίνηση.



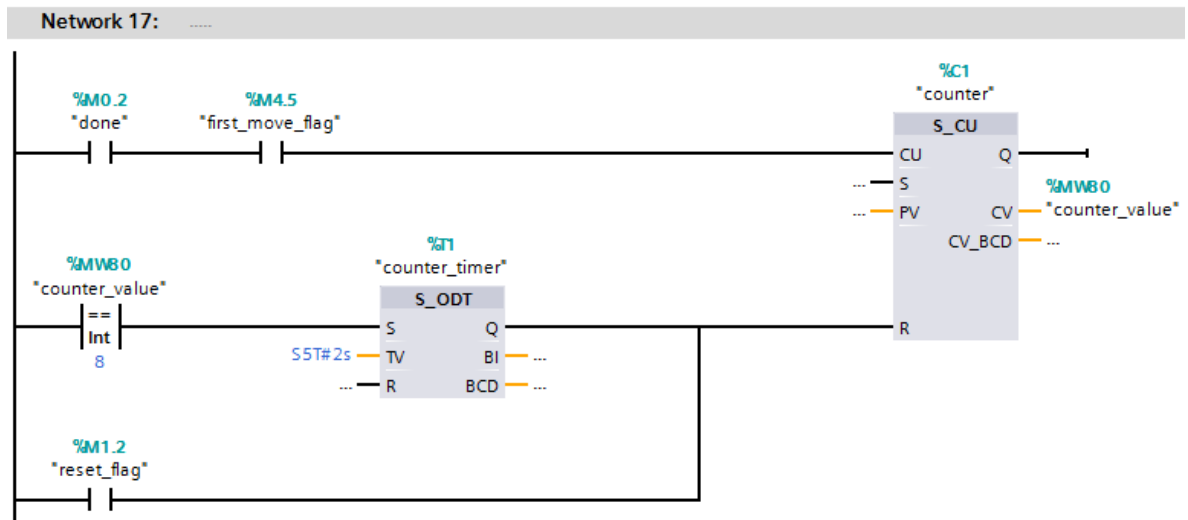
Πρόγραμμα 3.3: Το πρόγραμμα του Block_routine_1 (FC4).



Πρόγραμμα 3.3: Το πρόγραμμα του Block_routine_1 (FC4) (Συνέχεια).



Πρόγραμμα 3.3: Το πρόγραμμα του Block_routine_1 (FC4) (Συνέχεια).



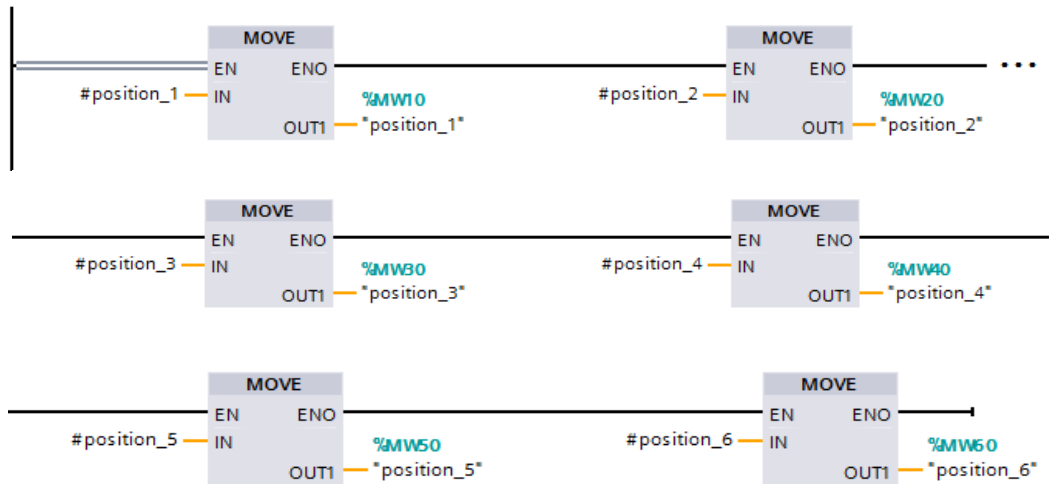
Πρόγραμμα 3.3: Το πρόγραμμα του Block_routine_1 (FC4) (Συνέχεια).

Πατώντας το κουμπί auto στην αρχική οθόνη του HMI πραγματοποιείται η μετάβαση στην οθόνη της αυτόματης λειτουργίας και ενεργοποιείται το Block_auto (FC3) από το OB1. Επιλέγοντας την ρουτίνα εργασίας 1 (routine 1) από την οθόνη της αυτόματης λειτουργίας ενεργοποιείται το Block_routine_1 (FC4).

Πατώντας start εκτελείται ο κώδικας της ρουτίνας 1. Φορτώνονται μέσω του Block_move (FC1) οι τιμές των γωνιακών θέσεων των αρθρώσεων και της θέσης της αρπάγης που αποτελούν την πρώτη κίνηση του εικονικού ρομποτικού βραχίονα και στέλνεται το σήμα execute ώστε να εκτελεστεί η κίνηση. Όταν η κίνηση ολοκληρωθεί το PLC θα λάβει το σήμα επιβεβαίωσης done, το οποίο θα αυξήσει το περιεχόμενο του μετρητή κατά ένα (από 0 θα γίνει 1) και παράλληλα θα εκτελέσει την επόμενη κίνηση, που έχει σαν προϋπόθεση το περιεχόμενο του μετρητή να είναι ίσον με το 1. Όταν ολοκληρωθεί και η δεύτερη κίνηση πράγμα που σημαίνει ότι το PLC θα λάβει πάλι το σήμα επιβεβαίωσης done, το περιεχόμενο του μετρητή αυξάνεται και γίνεται 2 παράλληλα εκτελείται και η τρίτη κίνηση και ούτω καθεξής. Μετά την εκτέλεση και της τελευταία κίνησης, η οποία έχει τις ίδιες τιμές με την πρώτη ο μετρητής μηδενίζεται και ο βρόχος ολοκληρώνεται. Έπειτα ξαναρχίζει η ίδια διαδικασία από το network 3 αυτή την φορά. Πριν από το network 5 και το network 13 όπου ο εικονικός ρομποτικός βραχίονας πλησιάζει για να πιάσει το υποτιθέμενο αντικείμενο και για να αφήσει το υποτιθέμενο αντικείμενο αντίστοιχα, υπάρχει κώδικας με τον οποίο σε συνδυασμό με τα μπουτόν grab object sensor και release object sensor του HMI προσομοιώνεται η λειτουργία των υποτιθέμενων αισθητήρων.

3.5.3 Block_move (FC1)

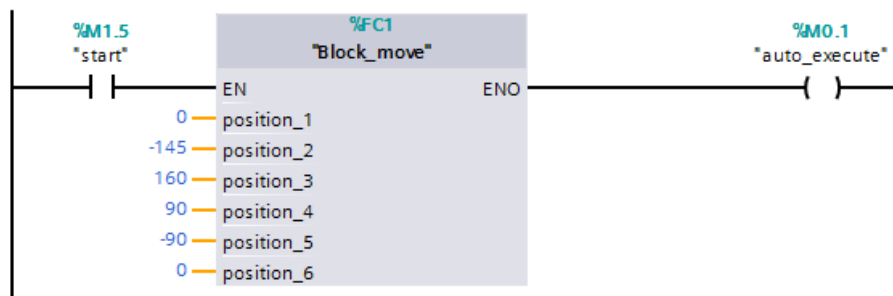
Το Block_move (FC1) που παρουσιάζεται στο Πρόγραμμα 3.4 είναι ένα παραμετροποιήσιμο μπλοκ που κατασκευάστηκε για να ενσωματώσει μια λειτουργία που επαναλαμβάνεται συχνά στο πρόγραμμα. Αυτή η λειτουργία είναι η μεταφορά των τιμών των γωνιακών θέσεων των αρθρώσεων και της θέσης της αρπάγης στα αντίστοιχα memory word (MW) έτσι ώστε να εκτελείται κάθε φορά η επιθυμητή κίνηση. Εξωτερικά αποτελείται από έξι εισόδους στις οποίες κάθε φορά εισάγονται οι επιθυμητές τιμές, και εσωτερικά από έξι εντολές move που μεταφέρουν τις τιμές των εισόδων στα αντίστοιχα MW.



Πρόγραμμα 3.4: Το πρόγραμμα του Block_move (FC1).

3.5.4 Block_transport_position (FC9)

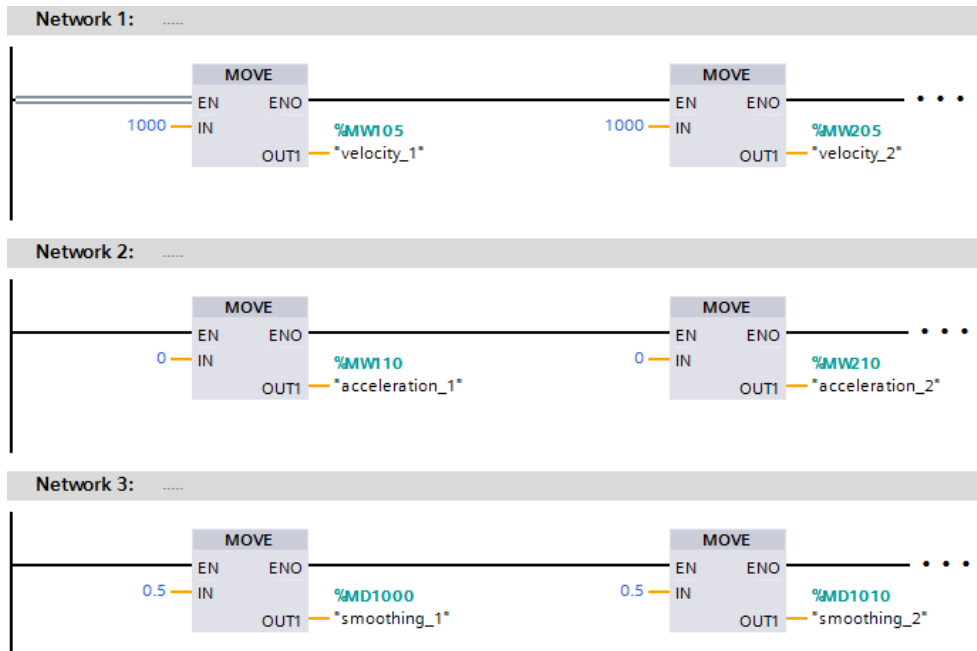
Στο Block_auto (FC3) εκτός από τις δυο ρουτίνες εργασίας και την ρουτίνα επίδειξης υπάρχει και το Block_transport_position (FC9) που παρουσιάζεται στο Πρόγραμμα 3.5. Αυτό περιέχει τον κώδικα για την λειτουργία της θέσης μεταφοράς του εικονικού ρομποτικού βραχίονα. Επιλέγοντας την λειτουργία transport position από την οθόνη της αυτόματης λειτουργίας του HMI και πατώντας start οι αρθρώσεις πάνε στις προεπιλεγμένες από τον κατασκευαστή θέσεις, ούτως ώστε ο ρομποτικός βραχίονας να μαζέψει για να είναι ευκολότερη η μεταφορά του.



Πρόγραμμα 3.5: Το πρόγραμμα του Block_transport_position (FC9).

3.5.5 Block_auto_default_parameters (FC10)

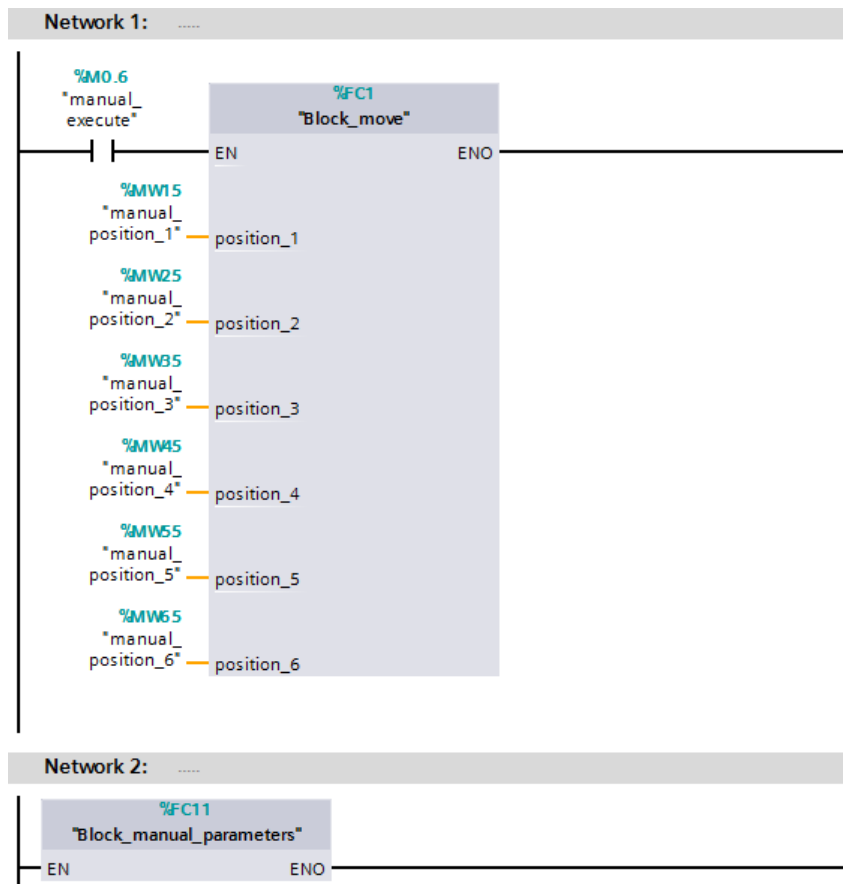
Το Block_auto_default_parameters (FC10) που παρουσιάζεται στο Πρόγραμμα 3.6 ενεργοποιείται μαζί με το Block_auto (FC3) και μεταφέρει τις προεπιλεγμένες τιμές της ταχύτητας, της επιτάχυνσης και της εξομάλυνσης κάθε κινητήρα στα αντίστοιχα MW και MD (memory double word).



Πρόγραμμα 3.6: Το πρόγραμμα του Block_auto_default_parameters (FC10).

3.5.6 Block_manual (FC2)

Το Block_manual (FC2) περιέχει τον κώδικα της χειροκίνητης λειτουργίας (Πρόγραμμα 3.7).



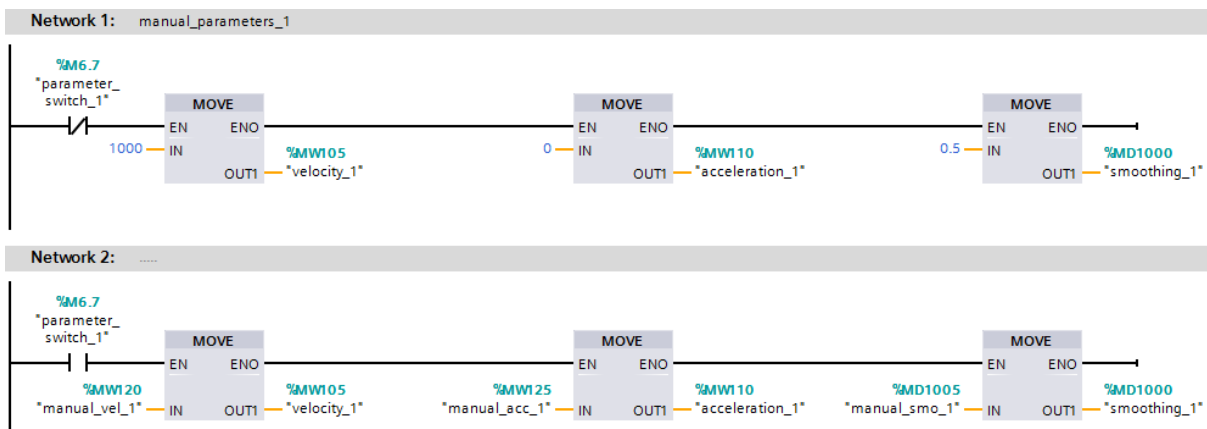
Πρόγραμμα 3.7: Το πρόγραμμα του Block_manual (FC2).

Πατώντας το κουμπί manual στην αρχική οθόνη του HMI πραγματοποιείται η μετάβαση στην οθόνη της χειροκίνητης λειτουργίας και ενεργοποιείται το Block_manual (FC2) από το OB1.

Το Block_manual (FC2) εσωτερικά καλεί το Block_move (FC1) και το Block_manual_parameters (FC11). Το Block_move (FC1) μεταφέρει τις τιμές των θέσεων που εισάγονται από τον χρήστη στα άσπρα κελιά της οθόνης χειροκίνητης λειτουργίας στα αντίστοιχα MW ούτως ώστε πατώντας το κουμπί execute από την οθόνη να επιτευχθεί ο επιθυμητός χειροκίνητος έλεγχος.

3.5.7 Block_manual_parameters (FC11)

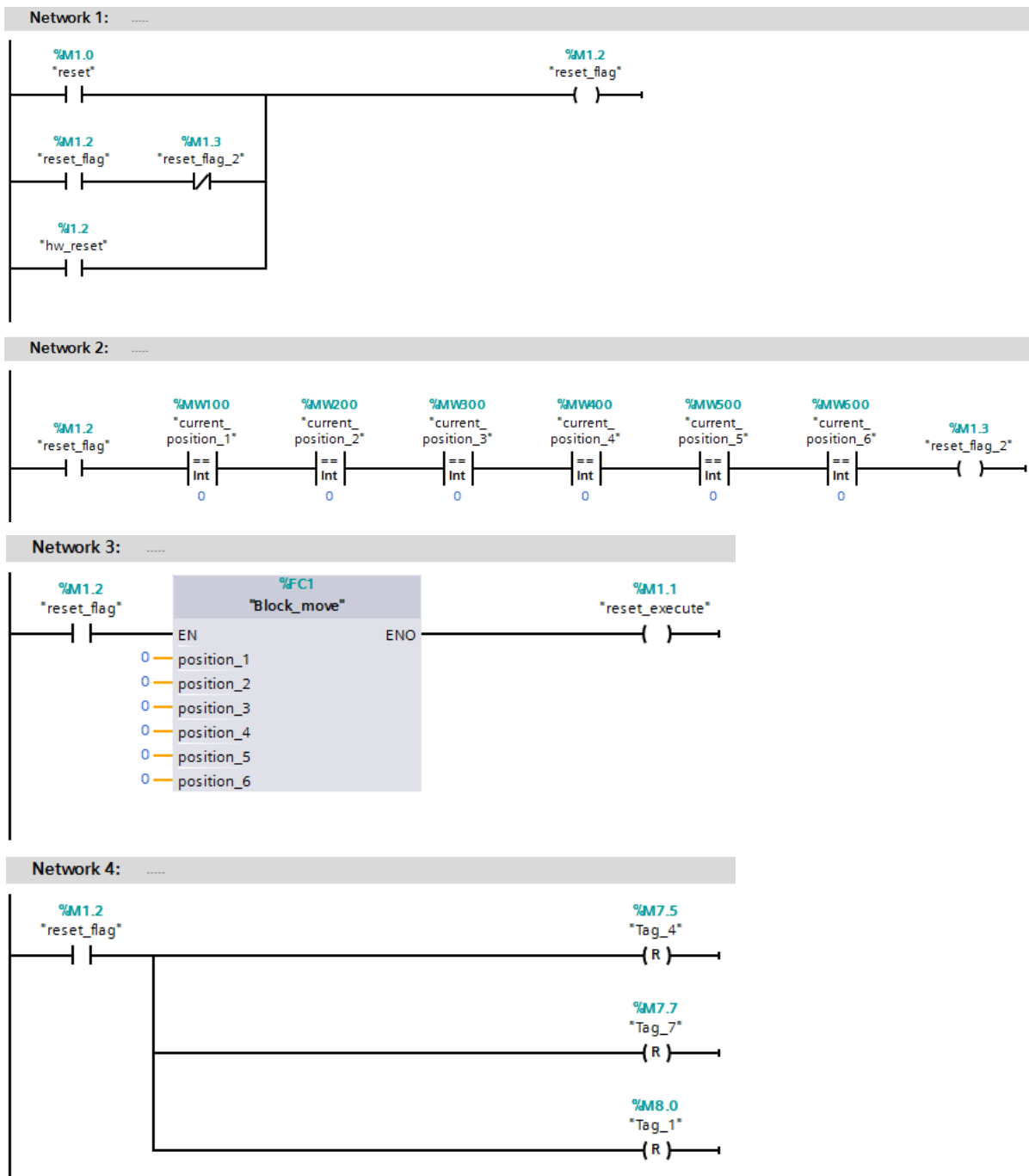
Όταν κάποιος διακόπτης παραμετροποίησης ενός κινητήρα στην οθόνη χειροκίνητης λειτουργίας είναι στη θέση DFL (default) το Block_manual_parameters (FC11) που παρουσιάζεται στο Πρόγραμμα 3.8 αναλαμβάνει να μεταφέρει στα αντίστοιχα MW και MD τις προεπιλεγμένες τιμές για ταχύτητα, επιτάχυνση και εξομάλυνση. Όταν ο διακόπτης είναι στη θέση MNL (manual) μεταφέρει τις τιμές που δίνει ο χρήστης στα άσπρα κελιά.



Πρόγραμμα 3.8: Το πρόγραμμα του Block_manual_parameters (FC11).

3.5.8 Block_reset (FC5)

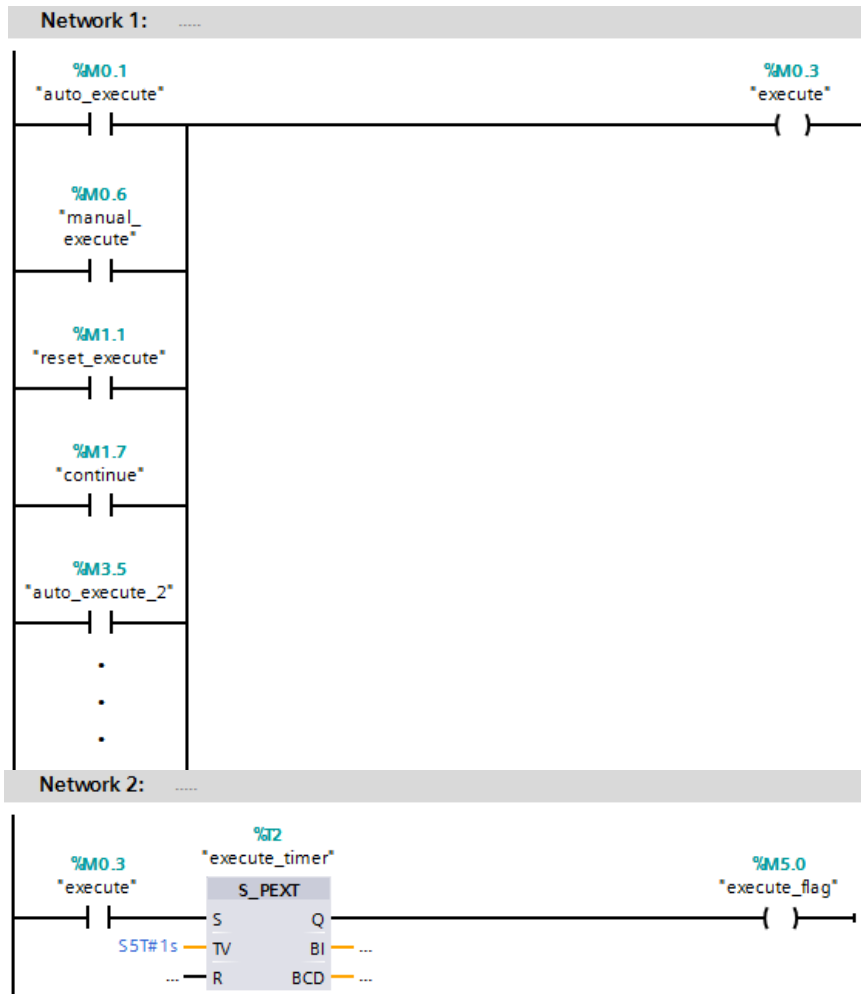
Το Block_reset (FC5) περιέχει τον κώδικα της λειτουργίας reset (Πρόγραμμα 3.9). Πατώντας το μπουτόν reset από την αντίστοιχη οθόνη του HMI είτε στην αυτόματη λειτουργία είτε στην χειροκίνητη πραγματοποιείται η διακοπή οποιασδήποτε λειτουργίας και η επαναφορά του εικονικού ρομποτικού βραχίονα στην αρχική θέση. Δηλαδή στις μηδέν μοίρες όλες οι αρθρώσεις και στα μηδέν χιλιοστά η αρπάγη.



Πρόγραμμα 3.9: Το πρόγραμμα του Block_reset (FC5).

3.5.9 Block_execute (FC6)

Στο Block_execute (FC6) που παρουσιάζεται στο Πρόγραμμα 3.10, συγκεντρώνονται όλα τα σήματα execute του προγράμματος και έπειτα από μια λογική OR προκύπτει το πραγματικό σήμα execute, το οποίο δίνει την εντολή εκτέλεσης των κινήσεων. Μέσω του Network 2 κάθε φορά που ενεργοποιείται το σήμα execute ενεργοποιείται και η ένδειξη του execute στο HMI.



Πρόγραμμα 3.10: Το πρόγραμμα του Block_execute (FC6).

Τέλος, υπάρχει και η δυνατότητα για σταμάτημα του εικονικού ρομποτικού βραχίονα είτε στην αυτόματη είτε στην χειροκίνητη λειτουργία ενώ κινείται. Αυτό γίνεται πατώντας το μπουτόν stop από την αντίστοιχη οθόνη και έπειτα για να συνεχίσει την κίνηση το μπουτόν continue. Η παραπάνω δυνατότητα δίνεται από το Network 5 του OB1.

ΚΕΦΑΛΑΙΟ 4

Η LabVIEW εφαρμογή για τον έλεγχο του τρισδιάστατου μοντέλου

4.1 Το λογισμικό LabVIEW

Το LabVIEW (Laboratory Virtual Instrument Engineering Workbench), είναι ένα γραφικό περιβάλλον προγραμματισμού που αναπτύχθηκε από την NI (National Instruments). Η γλώσσα προγραμματισμού που χρησιμοποιεί ονομάζεται G (Graphical Language) και είναι ένα είδος γλώσσας VPL (Visual Programming Language). Δίνει την δυνατότητα στον χρήστη να προγραμματίζει, όχι με βάση την γραφή εντολών αλλά με τον χειρισμό γραφικών εικονιδίων.

Το LabVIEW αρχικά αναπτύχθηκε από την εταιρία Apple Macintosh το 1986, όπου χρησιμοποιήθηκε κυρίως για την συλλογή δεδομένων, τον έλεγχο οργάνων και την ανάπτυξη βιομηχανικών εφαρμογών σε διάφορα λειτουργικά συστήματα. Όπως το Microsoft Windows, διάφορες εκδόσεις UNIX, Linux και Mac OS X.

Τα προγράμματα που δημιουργούνται στο LabVIEW ονομάζονται VIs (Virtual Instruments). Κάθε VI έχει δύο μέρη, το block diagram και το front panel. Το front panel είναι το περιβάλλον που χειρίζεται ο χρήστης χρησιμοποιώντας ελεγκτές (controls) και ενδείκτες (indicators). Μέσω των controls ο χρήστης έχει την δυνατότητα να μεταβάλλει τις παραμέτρους του προγράμματος, ενώ μέσω των indicators να παρατηρεί τα αποτελέσματα του προγράμματος. Το block diagram είναι το περιβάλλον στο οποίο συντάσσεται ο γραφικός κώδικας. Όλα τα controls και τα indicators παρουσιάζονται στο block diagram ως τερματικά εικονίδια (terminals). Το περιβάλλον του block diagram περιέχει τις απαιτούμενες δομές και συναρτήσεις που απαιτούνται για την δημιουργία του κώδικα μιας εφαρμογής. Μέσω του κώδικα επιτυγχάνεται η επεξεργασία των αρχικών δεδομένων, που περιέχουν τα controls του front panel, ενώ στην συνέχεια παρουσιάζονται τα τελικά δεδομένα σε κάποια indicators.

Στις συνηθισμένες γλώσσες προγραμματισμού που βασίζονται σε γραμμές εντολών (text based), η σειρά των εντολών καθορίζει την εκτέλεση του προγράμματος. Αντίθετα, το LabVIEW χρησιμοποιεί τον προγραμματισμό ροής δεδομένων (dataflow programming) στον οποίο η ροή των δεδομένων καθορίζει την σειρά που θα εκτελεστούν οι συναρτήσεις και τα SubVIs που περιέχει ένα block diagram. Στον προγραμματισμό ροής δεδομένων που χρησιμοποιεί το LabVIEW βασίζονται δυο από τα μεγαλύτερα πλεονεκτήματα του: η γρήγορη αποσφαλμάτωση (debugging) και η παράλληλη εκτέλεση. Επίσης, ένα μεγάλο πλεονέκτημα του LabVIEW είναι ο μικρός χρόνος ανάπτυξης εφαρμογών σε σχέση με τις συνηθισμένες γλώσσες προγραμματισμού. Αυτό οφείλεται κυρίως στην εύκολη αποσφαλμάτωση και στο σύνολο των συναρτήσεων – SubVIs που διαθέτει.

Επίσης, ο γραφικός κώδικας του LabVIEW, διευκολύνει την κατανόηση και την ανάπτυξη ενός προγράμματος. Η μεγάλη ποικιλία των έτοιμων παραδειγμάτων που παρέχει το LabVIEW συμβάλει σημαντικά στην εξοικείωση ενός νέου χρήστη με την ανάπτυξη διαφόρων εφαρμογών. Επιπλέον, είναι δυνατή η δημιουργία VIs, τα οποία θα έχουν την δυνατότητα να επικοινωνούν μεταξύ τους τόσο σε τοπικό επίπεδο όσο και απομακρυσμένα μέσω διαδικτύου.

Η NI προσφέρει ένα μεγάλο αριθμό από Add-Ons (πρόσθετα) για το LabVIEW. Τα Add-Ons χωρίζονται σε modules και toolkits. Έχουν σχεδιαστεί για να επεκτείνουν το περιβάλλον

ανάπτυξης του LabVIEW παρέχοντας εργαλεία, συναρτήσεις και λειτουργίες για την υλοποίηση ειδικών εφαρμογών. Στα πλαίσια της παρούσας πτυχιακής εργασίας χρησιμοποιούνται, το SoftMotion Module και το Datalogging and Supervisory Control Module, τα οποία αναλύονται παρακάτω.

4.1.1 Το SoftMotion Module

Το SoftMotion Module προσφέρει στο LabVIEW την γραφική ανάπτυξη εφαρμογών ελέγχου κίνησης. Παρέχει VIs, συναρτήσεις και λειτουργίες που βοηθούν στην κατασκευή ελεγκτών κίνησης που λειτουργούν σε συνδυασμό με πραγματικό υλικό (hardware) αλλά και με εικονικό υλικό. Το LabVIEW και το SoftMotion Module επιτρέπουν την εικονική πρωτοτυποποίηση για εφαρμογές κίνησης μέσω της δυνατότητας σύνδεσης με το 3D CAD πακέτο SolidWorks και κάνοντας χρήση ενός τρισδιάστατου μοντέλου. Στην παρούσα πτυχιακή εργασία το SoftMotion Module χρησιμοποιείται για την προσομοίωση στο LabVIEW της λειτουργίας του υλικού (hardware) που είναι υπεύθυνο για την κίνηση του ρομποτικού βραχίονα, όπως των οδηγών (drives), των περιστροφικών κωδικοποιητών (rotary encoders) κλπ. Χρησιμοποιείται επίσης για την σύνδεση του LabVIEW με το SolidWorks.

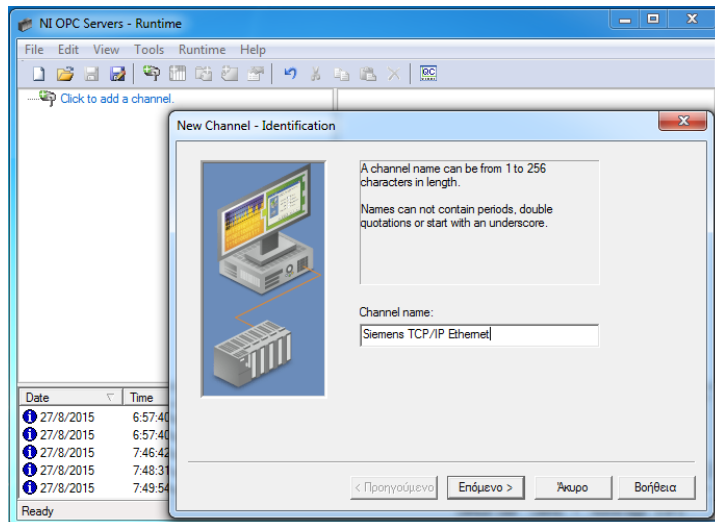
4.1.2 Το Datalogging and Supervisory Control Module

Το DSC (Datalogging and Supervisory Control) Module επεκτείνει τον γραφικό προγραμματισμό του LabVIEW για την ανάπτυξη εφαρμογών SCADA για τον εποπτικό έλεγχο και την συλλογή δεδομένων. Περιλαμβάνει εργαλεία για την επικοινωνία με PLCs και PACs (Programmable Automation Controller), την καταγραφή δεδομένων σε βάσεις δεδομένων, την διαχείριση συναγερμών και γεγονότων και την δημιουργία HMIs

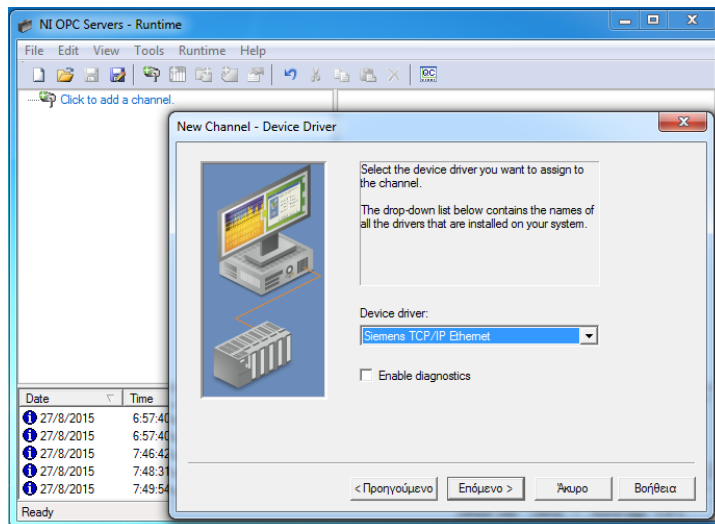
Το OPC (OLE - Object Linking and Embedding for Process Control) είναι ένα πρότυπο που αποτελείται από μια σειρά προδιαγραφών. Καθορίζει και επιτρέπει την επικοινωνία και την μεταφορά δεδομένων σε πραγματικό χρόνο μεταξύ υλικού και λογισμικού διαφορετικών κατασκευαστών. Το πρότυπο OPC βασίζεται στην κλασική αρχιτεκτονική Client/Server. Ουσιαστικά ο OPC server συλλέγει τα δεδομένα από τα διάφορα PLCs και τα μεταδίδει σε διάφορες εφαρμογές που συμπεριφέρονται ως clients. Ο OPC Server είναι ένα εργαλείο που περιλαμβάνεται στο DSC Module και ενσωματώνει την παραπάνω λειτουργία. Στην παρούσα πτυχιακή εργασία χρησιμοποιείται για την επικοινωνία και την μεταφορά των δεδομένων από το PLC προς το LabVIEW.

4.2 Η παραμετροποίηση του OPC Server

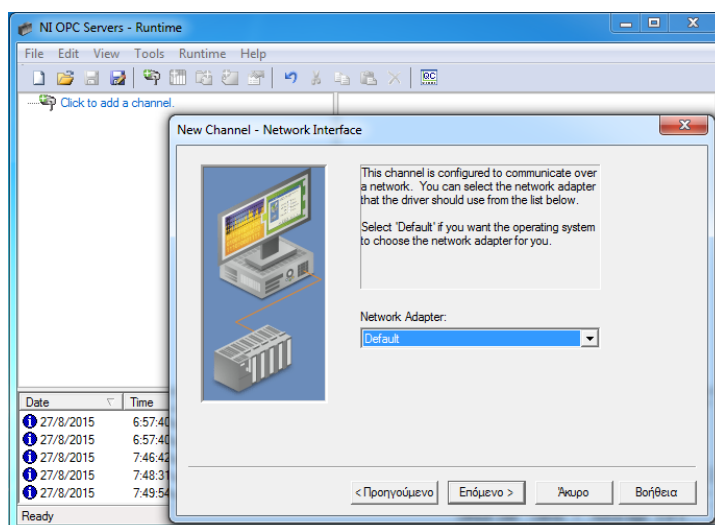
Η παραμετροποίηση του OPC Server αρχίζει με τη δημιουργία του καναλιού επικοινωνίας (Σχήμα 4.1 έως 4.6), ανοίγοντας τον OPC Server και πατώντας στην ένδειξη click to add a channel.



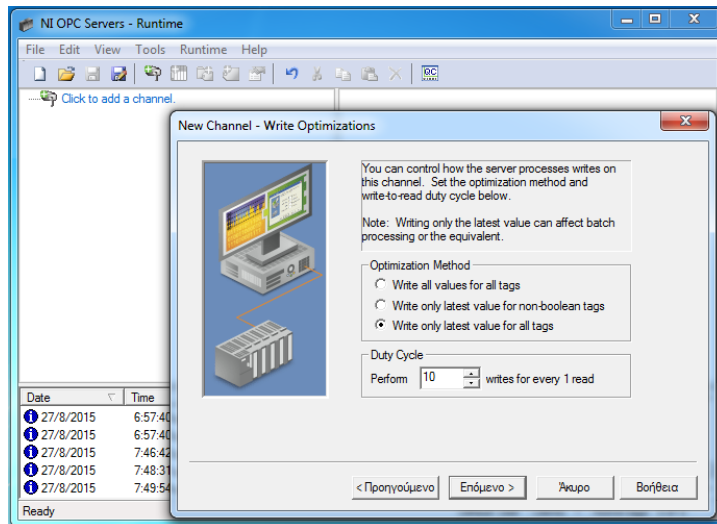
Σχήμα 4.1: Ονομασία καναλιού.



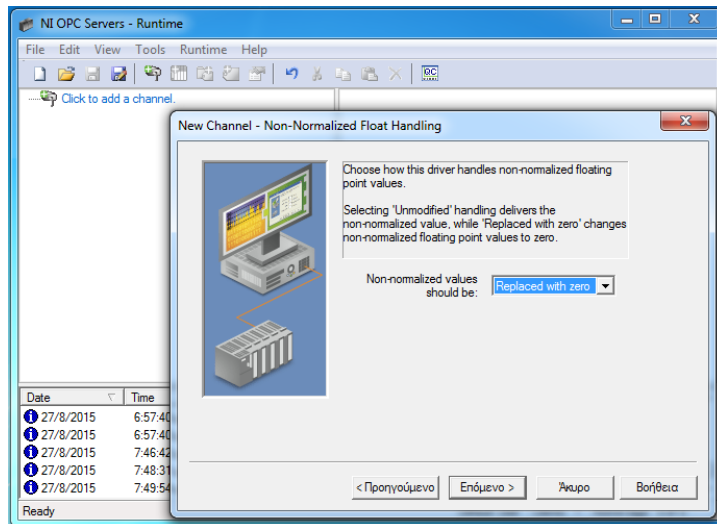
Σχήμα 4.2: Επιλογή του τύπου επικοινωνίας.



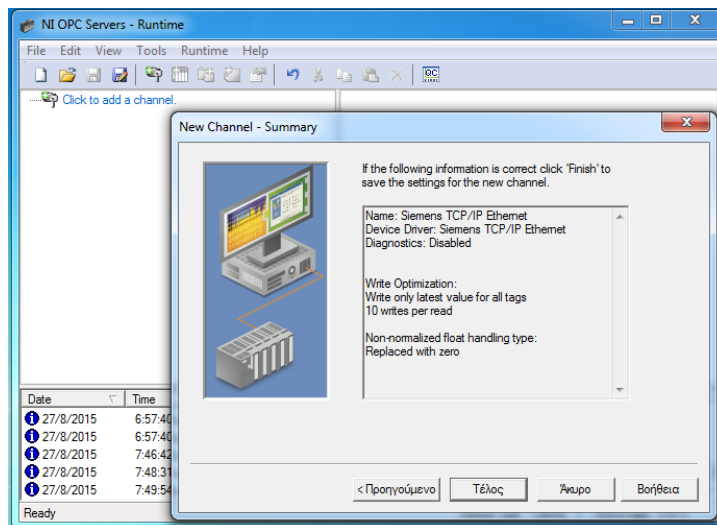
Σχήμα 4.3: Επιλογή του Network Adapter.



Σχήμα 4.4: Επιλογή βελτιστοποίησης.

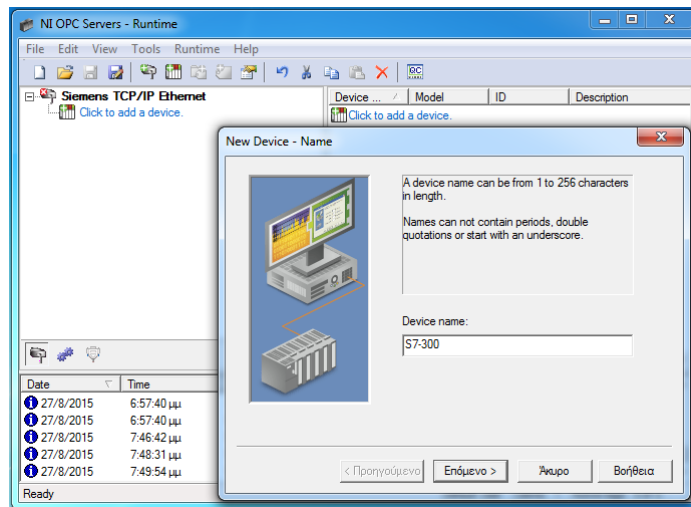


Σχήμα 4.5: Διαχείριση μη κανονικών τιμών.

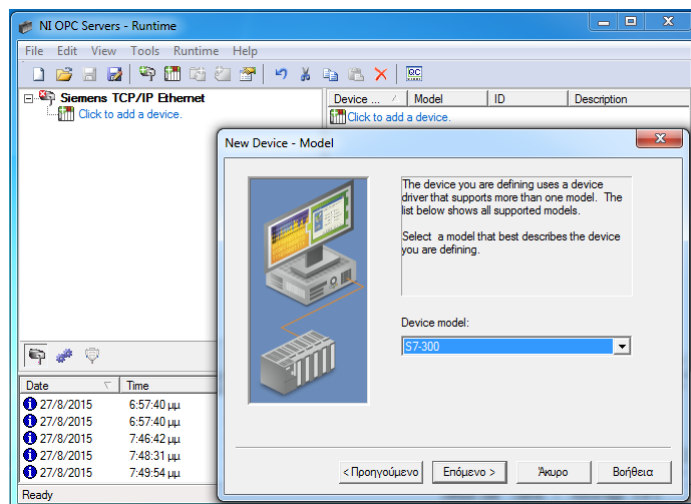


Σχήμα 4.6: Αποθήκευση του καναλιού που δημιουργήθηκε.

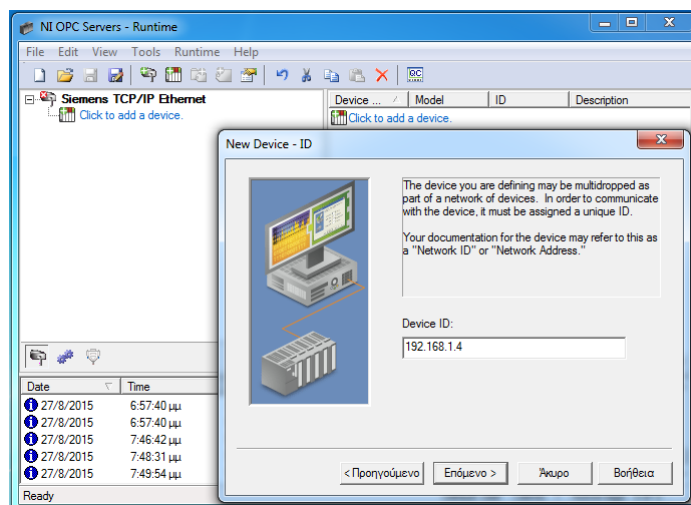
Στη συνέχεια προστίθεται στο κανάλι επικοινωνίας η συσκευή με την οποία θα γίνει η επικοινωνία (Σχήμα 4.7 έως 4.16), πατώντας στην ένδειξη click to add device.



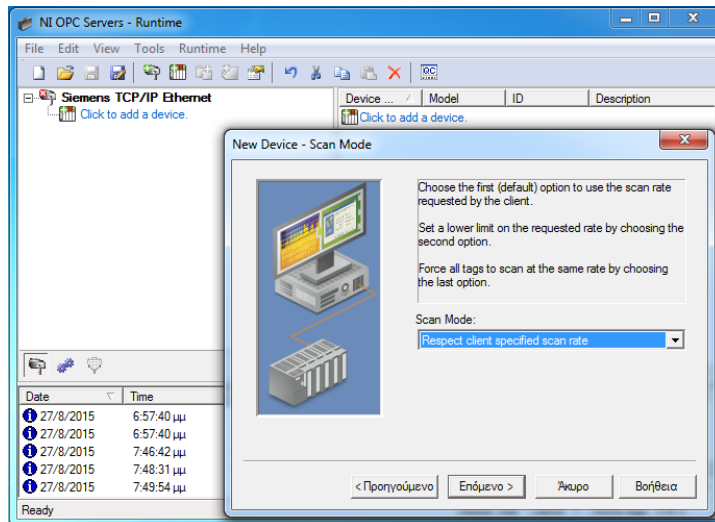
Σχήμα 4.7: Ονομασία συσκευής επικοινωνίας.



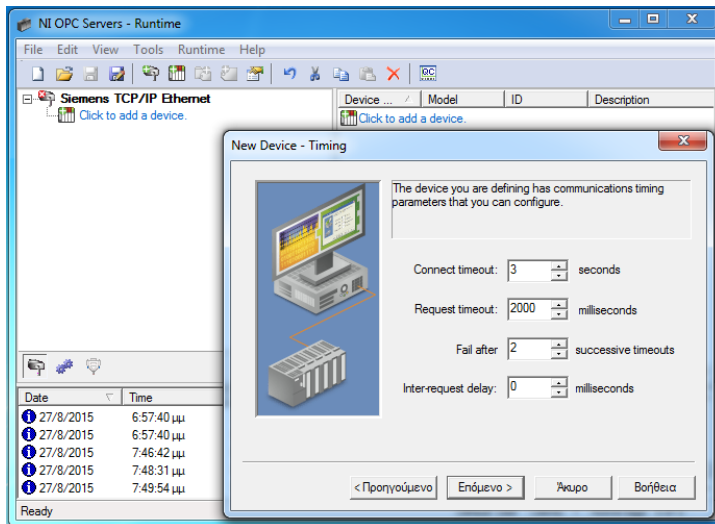
Σχήμα 4.8: Επιλογή σειράς συσκευής επικοινωνίας.



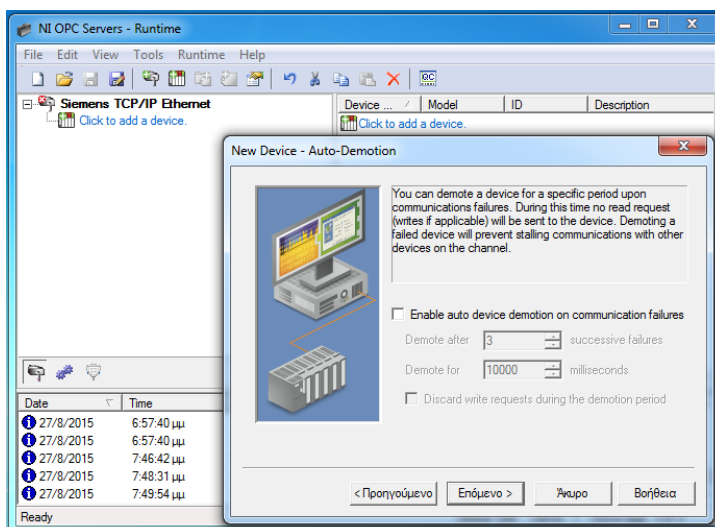
Σχήμα 4.9: IP διεύθυνση συσκευής.



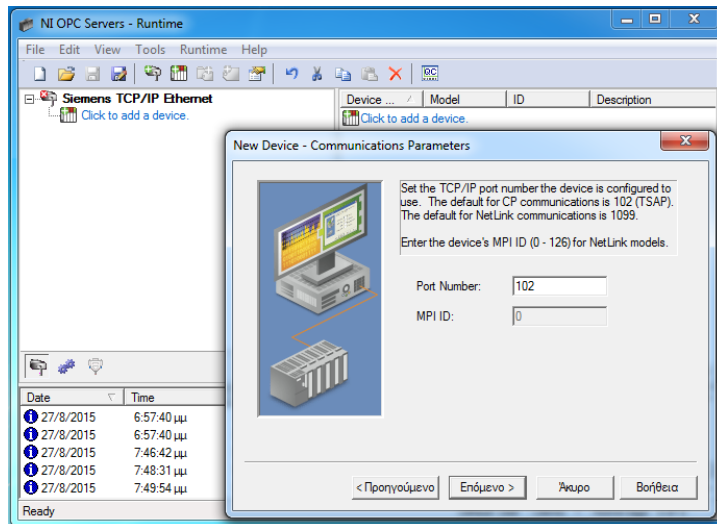
Σχήμα 4.10: Επιλογή παραμέτρων σάρωσης.



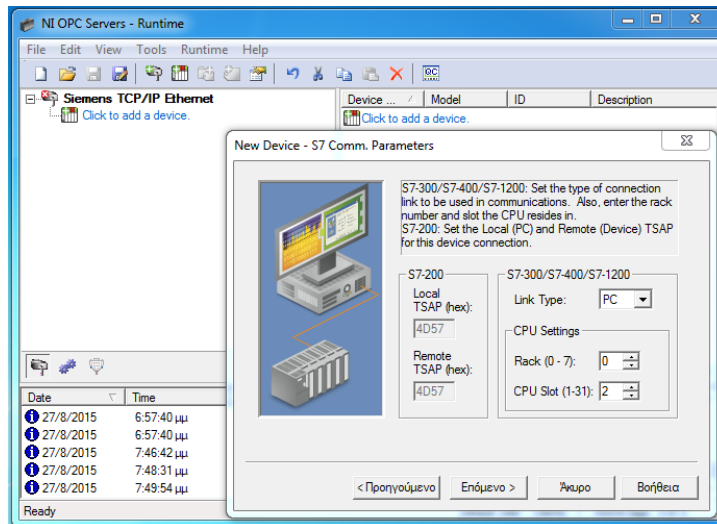
Σχήμα 4.11: Παράμετροι που καθορίζουν τους χρόνους επικοινωνίας.



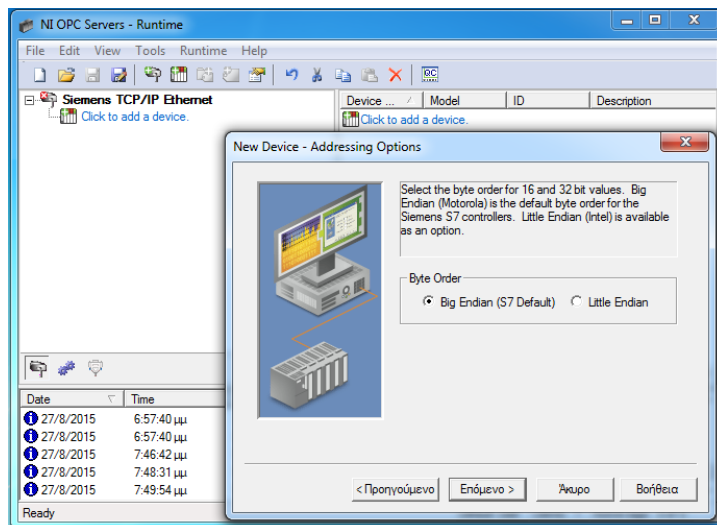
Σχήμα 4.12: Επιλογή υποβιβασμού συσκευής σε περίπτωση αποτυχίας της επικοινωνίας.



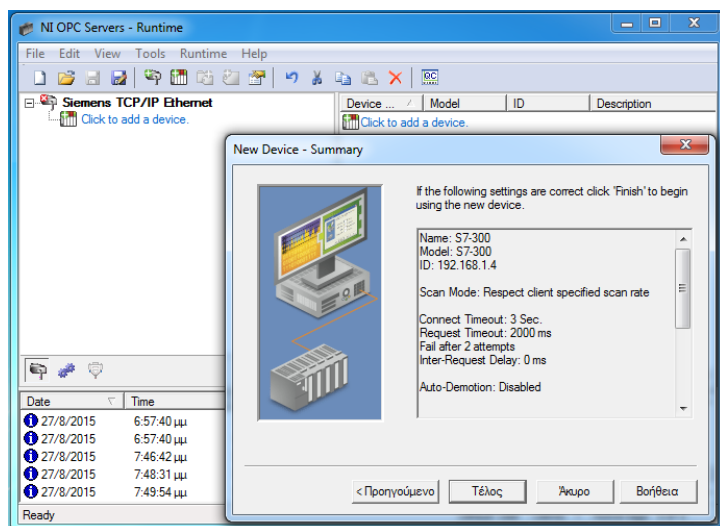
Σχήμα 4.13: Επιλογή θύρας επικοινωνίας.



Σχήμα 4.14: Επιλογή τύπου επικοινωνίας.



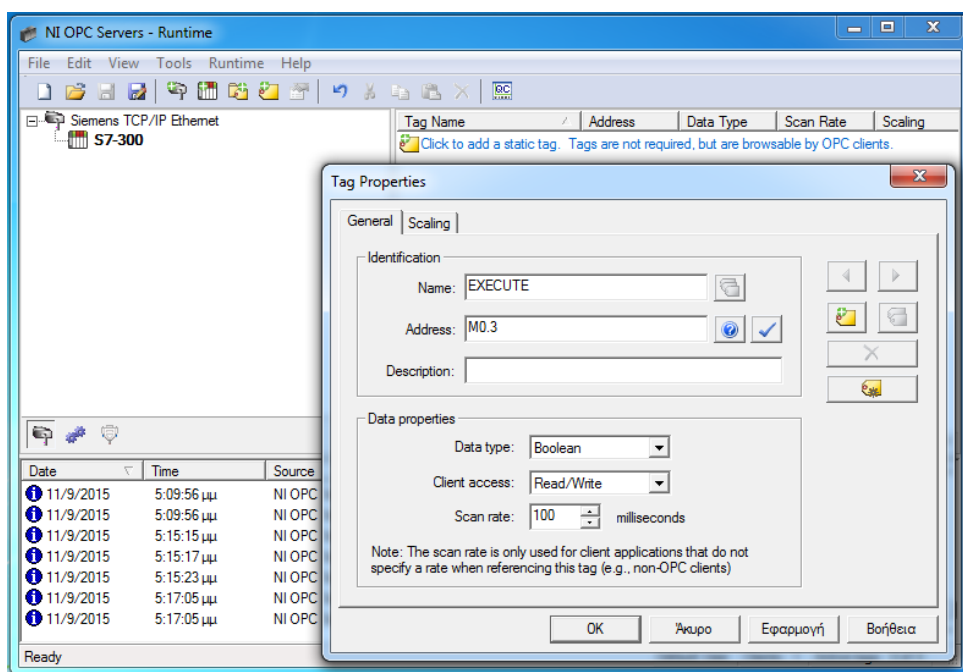
Σχήμα 4.15: Επιλογή της σειράς των Bytes.



Σχήμα 4.16: Τέλος ρυθμίσεων.

Το τελευταίο βήμα για την ολοκλήρωση της παραμετροποίησης του OPC Server είναι η δημιουργία των απαιτούμενων tags, που αντιστοιχούν στα tags του PLC και είναι απαραίτητα για τον έλεγχο του εικονικού ρομποτικού βραχίονα μέσω του αντίστοιχου VI. Ένα tag του OPC Server αντιπροσωπεύει ένα tag του κώδικα του PLC και φέρει πάντα την κατάσταση ή το περιεχόμενο της συγκεκριμένης μεταβλητής του κώδικα του PLC. Η δημιουργία των επιθυμητών tags στον OPC Server ολοκληρώνει την διαδικασία παραμετροποίησης του OPC Server.

Η δημιουργία ενός tag του OPC Server πραγματοποιείται επιλέγοντας Click to add a static tag και δηλώνοντας το όνομα του tag, την απόλυτη διεύθυνση του tag στο PLC και τον τύπο δεδομένων. Για παράδειγμα στο Σχήμα 4.17 φαίνεται η δημιουργία του tag του OPC Server με όνομα EXECUTE που αντιστοιχεί στο tag του PLC με απόλυτη διεύθυνση M 0.3 και συμβολική διεύθυνση execute. Στον Πίνακα 4.1 φαίνονται τα OPC Server tags που δημιουργήθηκαν και η αντιστοιχία τους με τα PLC tags.

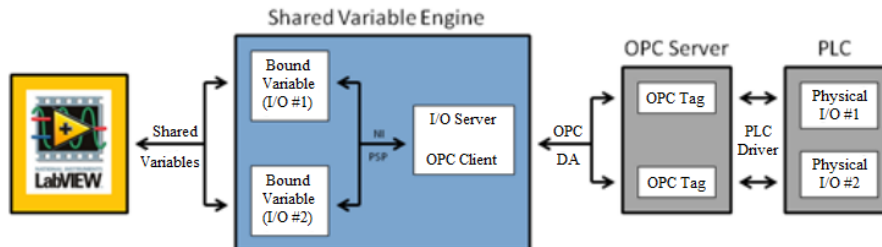


Σχήμα 4.17: Δημιουργία ενός tag.

OPC Server tag	PLC tag	Data Type	Address
acceleration_1	acceleration_1	Int	%MW110
acceleration_2	acceleration_2	Int	%MW210
acceleration_3	acceleration_3	Int	%MW310
acceleration_4	acceleration_4	Int	%MW410
acceleration_5	acceleration_5	Int	%MW510
acceleration_6	acceleration_6	Int	%MW610
current_position_1	current_position_1	Int	%MW100
current_position_2	current_position_2	Int	%MW200
current_position_3	current_position_3	Int	%MW300
current_position_4	current_position_4	Int	%MW400
current_position_5	current_position_5	Int	%MW500
current_position_6	current_position_6	Int	%MW600
done	done	Bool	%M0.2
done_indicator	done_indicator	Bool	%M5.2
execute	execute	Bool	%M0.3
lower_limit_1	lower_limit_1	Bool	%M2.1
lower_limit_2	lower_limit_2	Bool	%M2.3
lower_limit_3	lower_limit_3	Bool	%M2.5
lower_limit_4	lower_limit_4	Bool	%M2.7
lower_limit_5	lower_limit_5	Bool	%M3.1
lower_limit_6	lower_limit_6	Bool	%M3.3
position_1	position_1	Int	%MW10
position_2	position_2	Int	%MW20
position_3	position_3	Int	%MW30
position_4	position_4	Int	%MW40
position_5	position_5	Int	%MW50
position_6	position_6	Int	%MW60
smoothing_1	smoothing_1	Real	%MD1000
smoothing_2	smoothing_2	Real	%MD1010
smoothing_3	smoothing_3	Real	%MD1020
smoothing_4	smoothing_4	Real	%MD1030
smoothing_5	smoothing_5	Real	%MD1040
smoothing_6	smoothing_6	Real	%MD1050
stop	stop	Bool	%M1.6
upper_limit_1	upper_limit_1	Bool	%M2.0
upper_limit_2	upper_limit_2	Bool	%M2.2
upper_limit_3	upper_limit_3	Bool	%M2.4
upper_limit_4	upper_limit_4	Bool	%M2.6
upper_limit_5	upper_limit_5	Bool	%M3.0
upper_limit_6	upper_limit_6	Bool	%M3.2
velocity_1	velocity_1	Int	%MW105
velocity_2	velocity_2	Int	%MW205
velocity_3	velocity_3	Int	%MW305
velocity_4	velocity_4	Int	%MW405
velocity_5	velocity_5	Int	%MW505
velocity_6	velocity_6	Int	%MW605

Πίνακας 4.1: OPC Server tags.

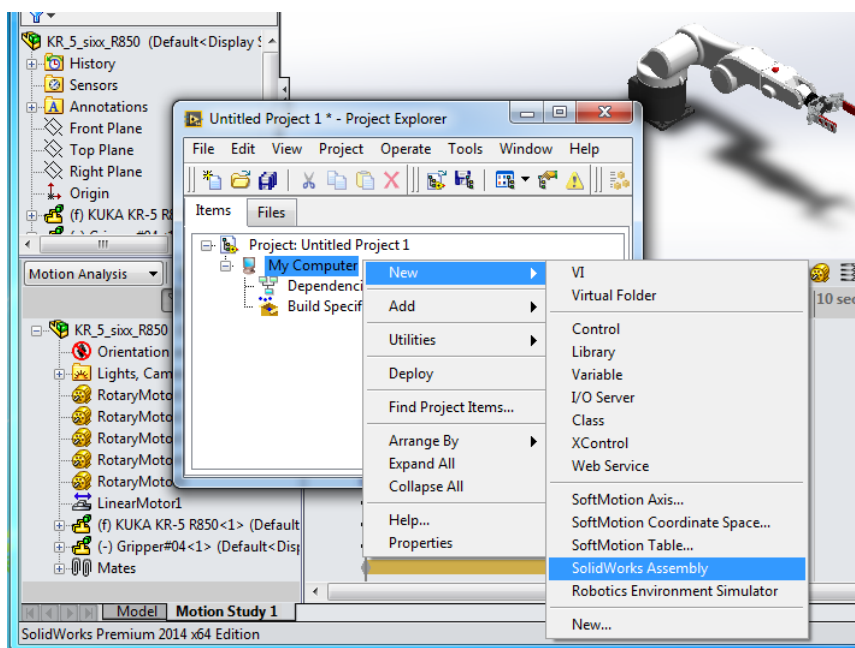
Η επικοινωνία και η μεταφορά δεδομένων από το PLC προς το LabVIEW και το αντίστροφο αναλυτικά γίνεται ως εξής: Το περιεχόμενο των tags του κώδικα του PLC μεταφέρεται μέσω του επιλεγμένου προγράμματος οδήγησης στα αντίστοιχα tags του OPC Server. Τα OPC Server tags συνδέονται με τον server εισόδων/εξόδων του OPC Client μέσω του πρότυπου OPC DA. Έπειτα η Shared Variable Engine του LabVIEW δεσμεύει τα OPC Server tags με μεταβλητές μέσω του πρωτοκόλλου NI PSP. Χρησιμοποιώντας τις παραπάνω μεταβλητές ο γραφικός κώδικας του LabVIEW μπορεί να διαβάσει ή να γράψει δεδομένα στα OPC Server tags και κατά επέκταση στα PLC tags (Σχήμα 4.18).



Σχήμα 4.18: Επικοινωνία PLC με LabVIEW.

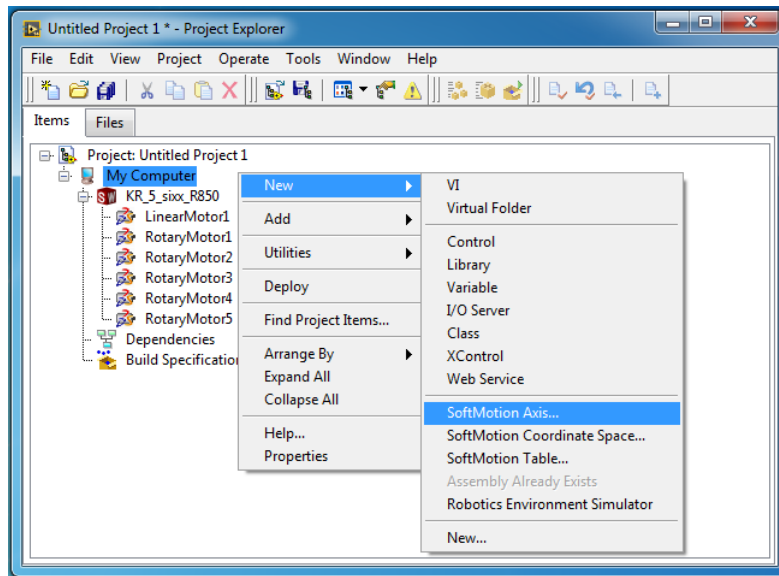
4.3 Το LabVIEW Project

Για να είναι εφικτή η επικοινωνία του LabVIEW με το SolidWorks θα πρέπει να εισαχθεί το assembly του τρισδιάστατου μοντέλου του ρομποτικού βραχίονα στο Project του LabVIEW. Η εισαγωγή του assembly πραγματοποιείται ανοίγοντας πρώτα το αρχείο του τρισδιάστατου μοντέλου στο SolidWorks και έπειτα πατώντας στο Project tree του LabVIEW δεξί κλικ στο My Computer → New → SolidWorks Assembly (Σχήμα 4.19). Στην συνέχεια μέσω του παράθυρου διαλόγου που εμφανίζεται, επιλέγεται η διαδρομή του αρχείου του assembly το οποίο προστίθεται τελικά στο Project του LabVIEW.



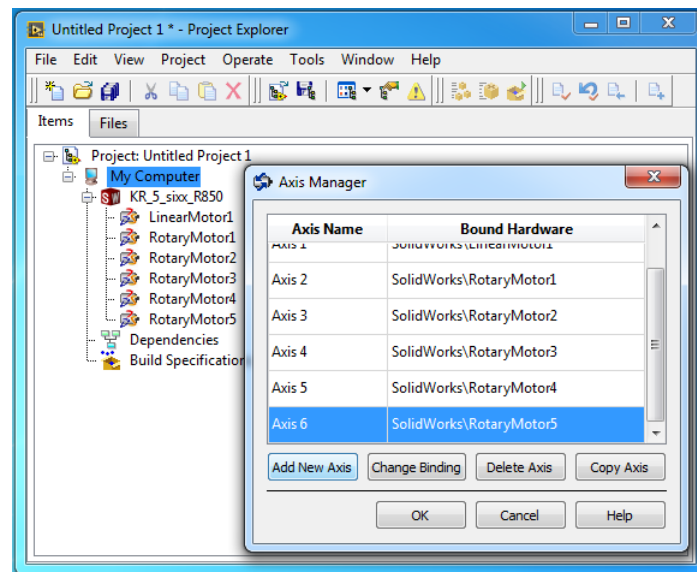
Σχήμα 4.19: Εισαγωγή SolidWorks assembly στο LabVIEW.

Όταν ολοκληρωθεί η εισαγωγή του τρισδιάστατου μοντέλου σειρά έχει η εισαγωγή των αξόνων-κινητήρων του εικονικού ρομποτικού βραχίονα στο LabVIEW Project όπως παρουσιάζεται στο Σχήμα 4.20.



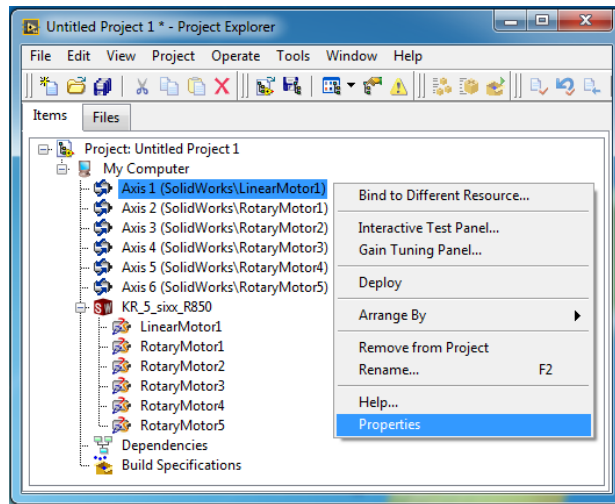
Σχήμα 4.20: Εισαγωγή αξόνων-κινητήρων.

Η εισαγωγή των αξόνων-κινητήρων ολοκληρώνεται πατώντας στην επιλογή Add New Axis έως ότου εισαχθούν όλοι οι άξονες-κινητήρες και μετά στην επιλογή OK (Σχήμα 4.21).

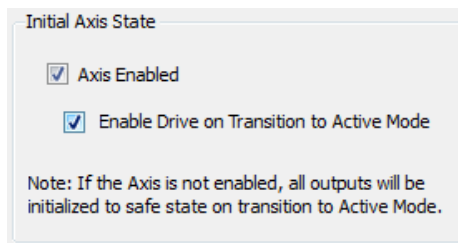


Σχήμα 4.21: Εισαγωγή αξόνων-κινητήρων.

Το επόμενο βήμα είναι η ενεργοποίηση όλων των αξόνων-κινητήρων πατώντας δεξί κλικ σε κάθε άξονα-κινητήρα → properties (Σχήμα 4.22) και η ενεργοποίηση της επιλογής Enable Drive on Transition to Active Mode (Σχήμα 4.23).



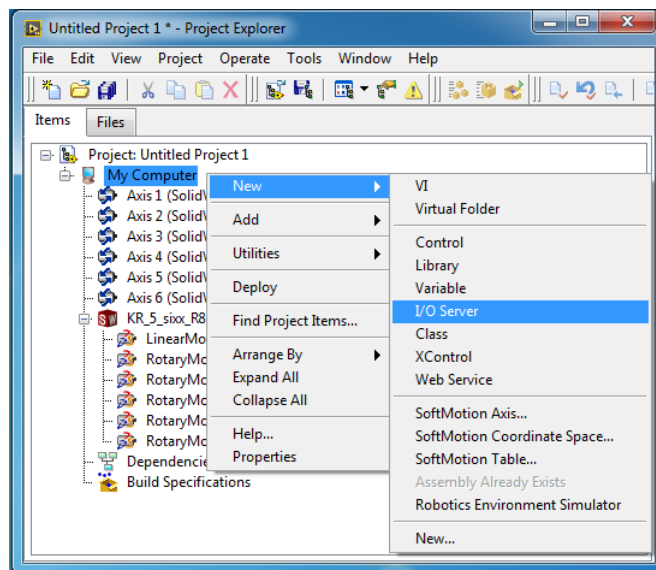
Σχήμα 4.22: Ενεργοποίηση αξόνων-κινητήρων.



Σχήμα 4.23: Ενεργοποίηση αξόνων-κινητήρων.

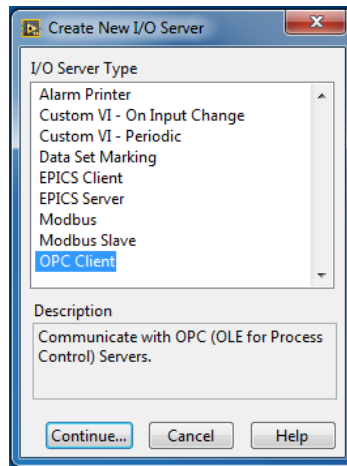
Τέλος, εισάγονται στο LabVIEW Project τα tags του OPC Server που δημιουργήθηκαν τα οποία θα χρησιμοποιηθούν σαν μεταβλητές στον γραφικό κώδικα. Η εισαγωγή πραγματοποιείται με τα παρακάτω βήματα:

- Δημιουργία του I/O Server πατώντας στο Project tree του LabVIEW δεξί κλικ στο My Computer → New → I/O Server (Σχήμα 4.24).



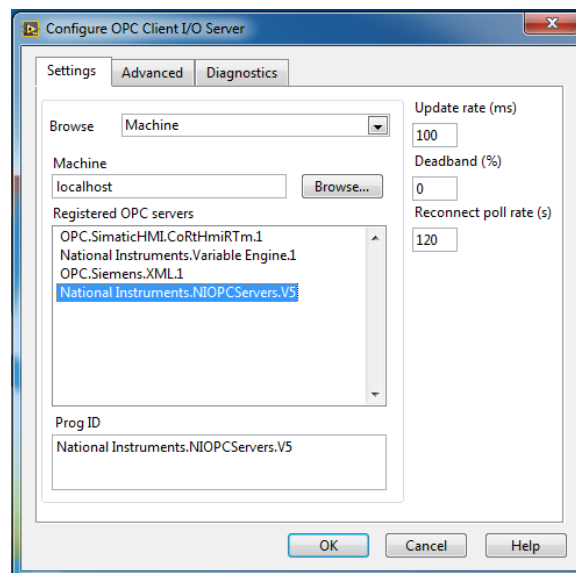
Σχήμα 4.24: Δημιουργία I/O Server.

- Στο παράθυρο που θα εμφανιστεί, επιλογή I/O Server τύπου OPC Client και κλικ στο Continue (Σχήμα 4.25).

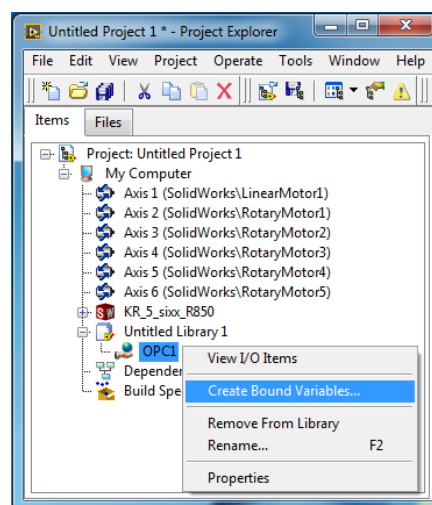


Σχήμα 4.25: Επιλογή τύπου I/O Server.

- Στο παράθυρο διαμόρφωσης, επιλογή του National Instruments.NIOPCServers.V5, ρύθμιση του Update rate στα 100 ms και κλικ στο OK (Σχήμα 4.26).

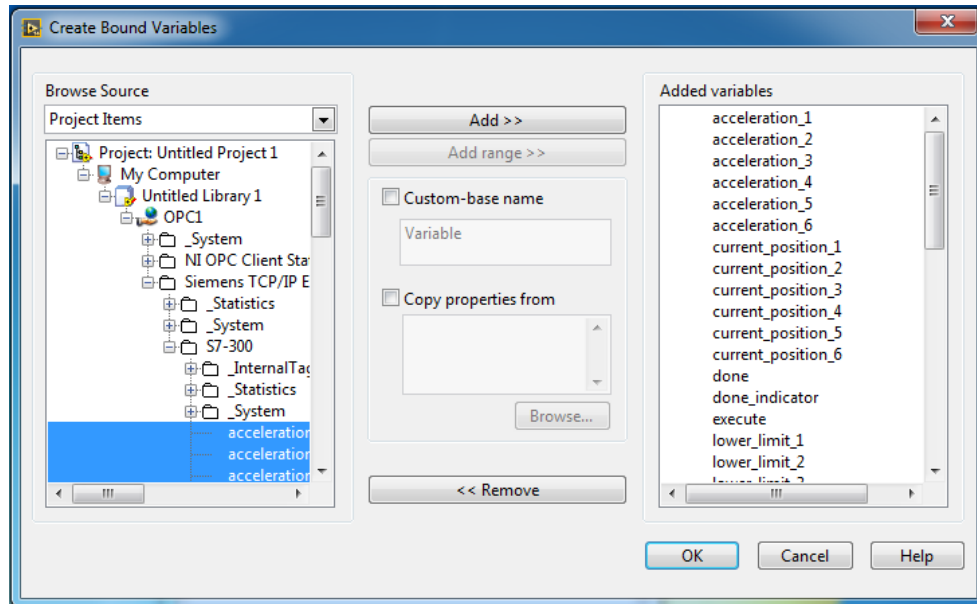


Σχήμα 4.26: Διαμόρφωση OPC Client I/O Server.



Σχήμα 4.27: Δημιουργία μεταβλητών.

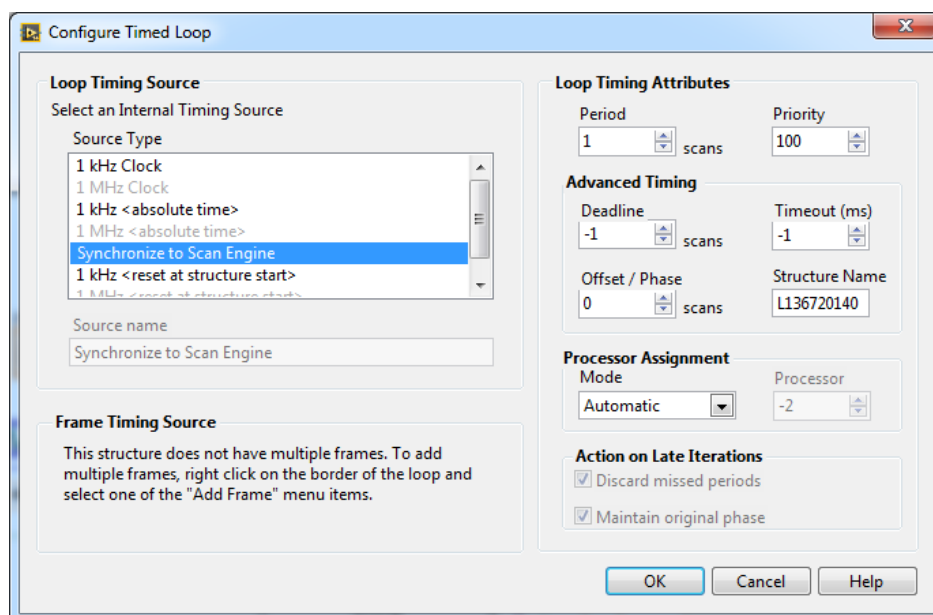
- Δεξί κλικ στο OPC1 εντός του Untitled Library 1 που δημιουργήθηκε από τα παραπάνω βήματα στο LabVIEW Project και επιλογή του Create Bound Variables (Σχήμα 4.27).
- Το τελευταίο βήμα είναι η επιλογή των tags που δημιουργήθηκαν νωρίτερα και είναι φορτωμένα στον NI OPC Server και η εισαγωγή τους στο LabVIEW Project σαν μεταβλητές πατώντας στο Add >> (Σχήμα 4.28).



Σχήμα 4.28: Εισαγωγή tags σαν μεταβλητές.

4.4 Το πρόγραμμα του LabVIEW Project

Η ανάπτυξη του γραφικού κώδικα στο LabVIEW μέσω του οποίου επιτυγχάνεται ο έλεγχος του τρισδιάστατου μοντέλου του ρομποτικού βραχίονα στο SolidWorks από το PLC πραγματοποιήθηκε μέσα σε μία Timed Loop με τις ρυθμίσεις που φαίνονται στο Σχήμα 4.29.



Σχήμα 4.29: Διαμόρφωση Timed Loop.

Το Block Diagram (Σχήμα 4.30) αποτελείται από ένα παραμετροποιήσιμο τμήμα κώδικα (Σχήμα 4.31) που επαναλαμβάνεται έξι φορές και αναλαμβάνει τον έλεγχο του κινητήρα κάθε άρθρωσης, εισάγοντας κάθε φορά τις αντίστοιχες μεταβλητές εισόδου και εξόδου. Οι μεταβλητές εισόδου μεταφέρουν δεδομένα από το PLC στον γραφικό κώδικα του LabVIEW μέσω των tags και οι μεταβλητές εξόδου από το LabVIEW στο PLC μέσω των tags.

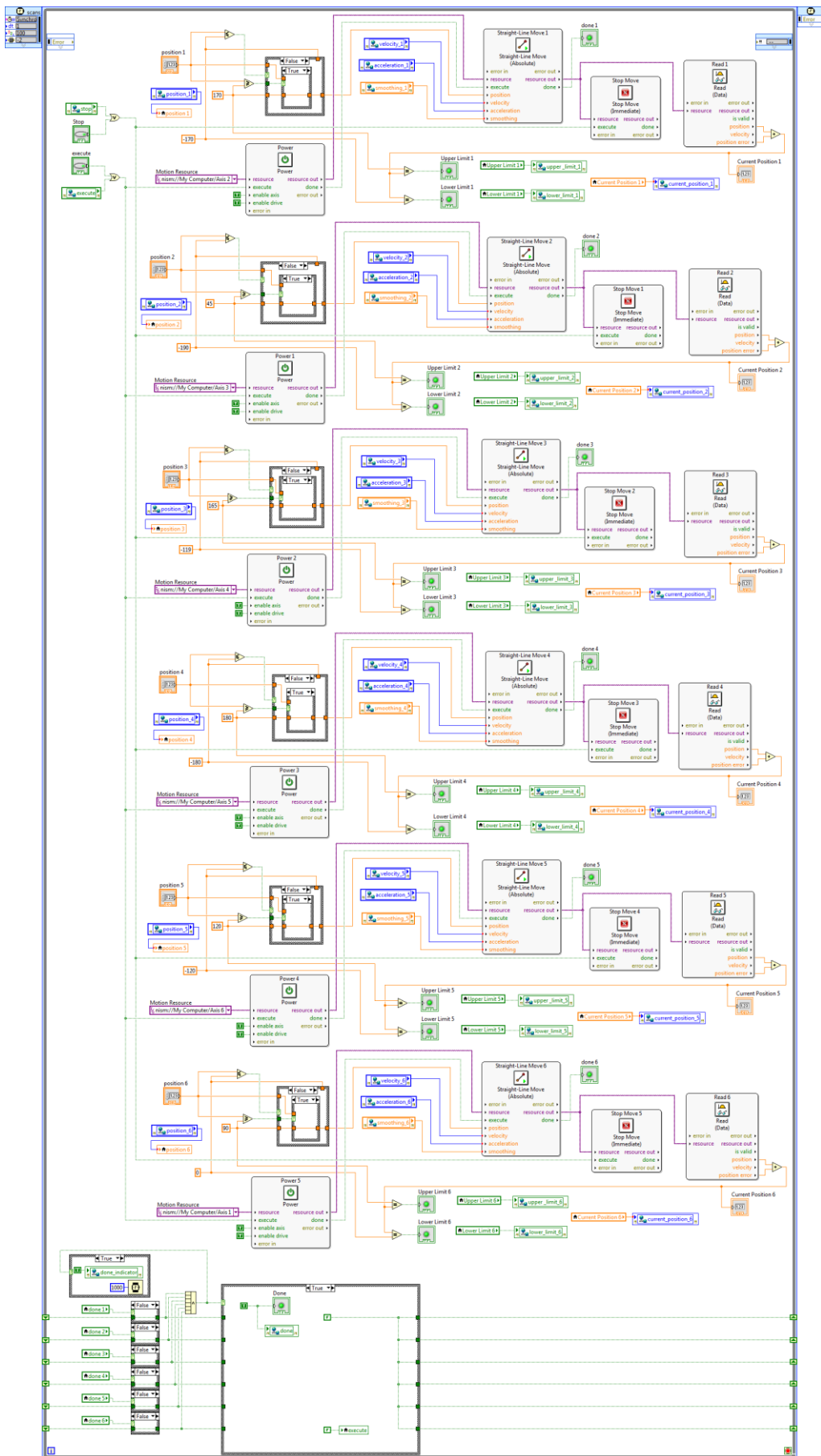
Είσοδο στον γραφικό κώδικα του LabVIEW αποτελούν οι εξής μεταβλητές:

- position_1, position_2, ..., position_6. Περιέχουν τις επιθυμητές τιμές των γωνιακών θέσεων των αρθρώσεων και της θέσης της αρπάγης.
- velocity_1, velocity_2, ..., velocity_6. Περιέχουν την τιμή της ταχύτητας κάθε κινητήρα.
- acceleration_1, acceleration_2, ..., acceleration_6. Περιέχουν την τιμή της επιτάχυνσης κάθε κινητήρα.
- smoothing_1, smoothing_2, ..., smoothing_6. Περιέχουν την τιμή της εξομάλυνσης κάθε κινητήρα.
- execute. Σχετίζεται με την εντολή για την εκτέλεση των κινήσεων.
- stop. Σχετίζεται με την εντολή για το σταμάτημα όλων των κινήσεων.

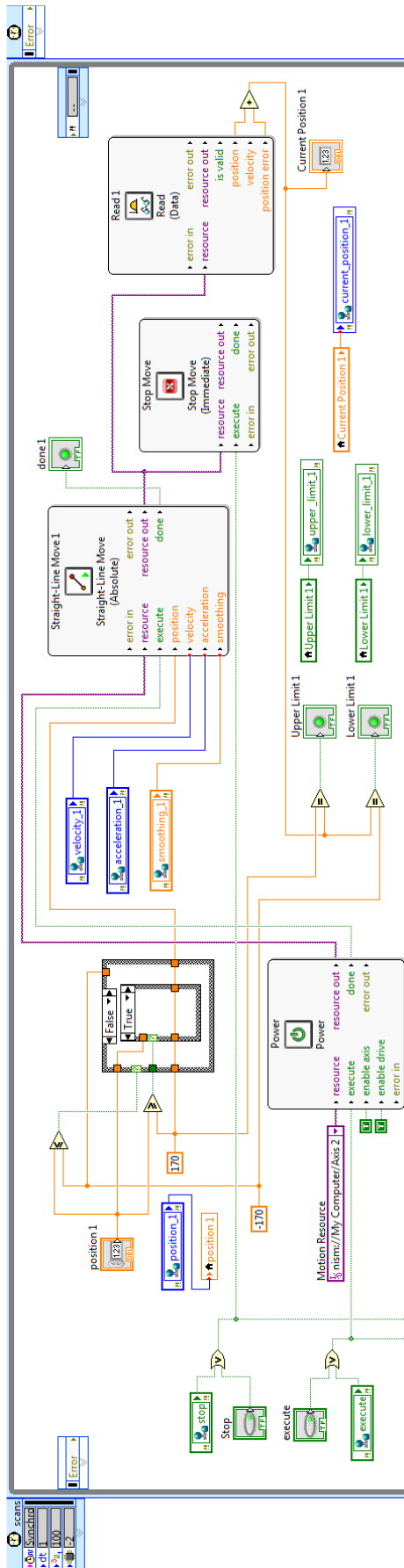
Έξοδο από τον γραφικό κώδικα του LabVIEW αποτελούν οι εξής μεταβλητές:

- current_position_1, current_position_2, ..., current_position_6. Περιέχουν την τρέχουσα θέση κάθε άρθρωσης.
- upper_limit_1, upper_limit_2, ..., upper_limit_6. Περιέχουν την πληροφορία ότι κάποια άρθρωση έχει φτάσει στο ανώτατο όριο.
- lower_limit_1, lower_limit_2, ..., lower_limit_6. Περιέχουν την πληροφορία ότι κάποια άρθρωση έχει φτάσει στο κατώτατο όριο.
- done. Περιέχει την πληροφορία ότι ολοκληρώθηκαν όλες οι κινήσεις.

Το LabVIEW αρχικά δέχεται τις εντολές από το PLC που περιγράφουν τις κινήσεις που θα εκτελεστούν. Στην συνέχεια εκτελούνται οι κινήσεις των αρθρώσεων και της αρπάγης και τέλος ενημερώνεται το PLC ότι οι παραπάνω κινήσεις έχουν ολοκληρωθεί. Αναλυτικότερα όταν ξεκινάει μια ρουτίνα εργασίας από την αυτόματη λειτουργία του εικονικού ρομποτικού βραχίονα στο HMI, το PLC στέλνει το πρώτο σεντ κινήσεων (τιμές των γωνιακών θέσεων των αρθρώσεων και της θέσης της αρπάγης) μέσω των tags στις μεταβλητές εισόδου position_1, position_2, ..., position_6 στο LabVIEW. Αμέσως μετά το PLC στέλνει την εντολή execute και μέσω του αντίστοιχου tag ενεργοποιείται η μεταβλητή εισόδου execute στο LabVIEW και εκτελούνται οι επιθυμητές κινήσεις. Κάθε κίνηση που ολοκληρώνεται στο LabVIEW ενεργοποιεί την αντίστοιχη ένδειξη επιβεβαίωσης done 1, done 2, ..., done 6 τα σήματα αυτών των ενδείξεων συγκεντρώνονται στο τέλος του Block Diagram και συνθέτουν μέσω κατάλληλης λογικής την μεταβλητή εξόδου done. Όταν ολοκληρωθούν όλες οι κινήσεις το σήμα done στέλνεται μέσω του αντίστοιχου tag στο PLC και το LabVIEW αναμένει να λάβει από το PLC τις πληροφορίες που σχετίζονται με το επόμενο σεντ κινήσεων. Η διαδικασία που περιγράφηκε παραπάνω είναι η ίδια ανεξάρτητα αν ο χρήστης μέσω του HMI του συστήματος έχει επιλέξει την αυτόματη ή την χειροκίνητη λειτουργία.



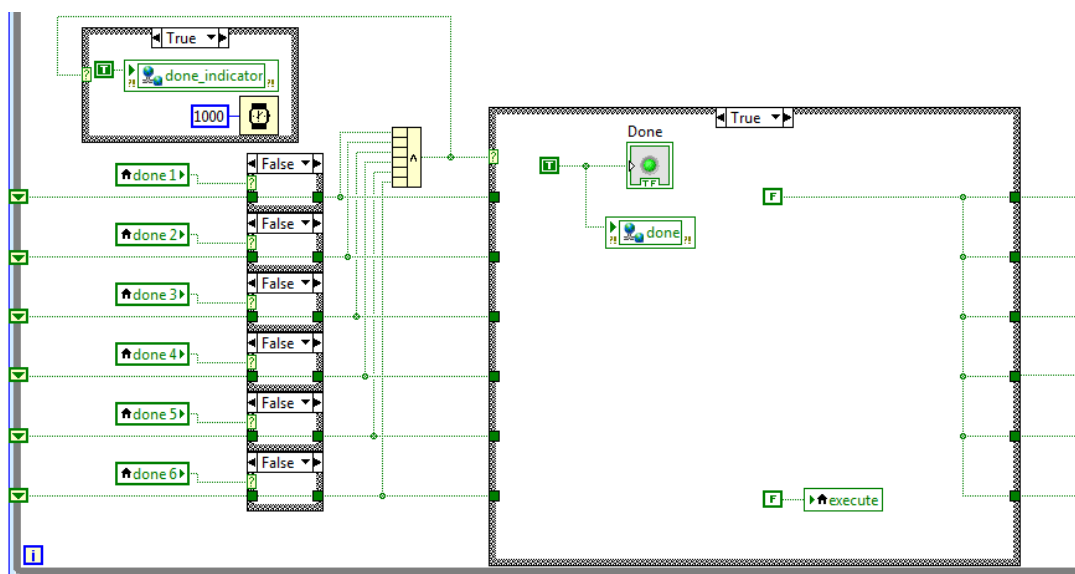
Σχήμα 4.30: Block Diagram.



Σχήμα 4.31: Παραμετροποιήσιμο τμήμα κώδικα.

Για τον έλεγχο του κινητήρα της πρώτης άρθρωσης το παραμετροποιήσιμο τμήμα κώδικα (Σχήμα 4.31) συνδέεται με τον κινητήρα 1 (Axis 2 SolidWorks\RotaryMotor1) και με τις αντίστοιχες μεταβλητές εισόδου/εξόδου π.χ. position_1, velocity_1, current_position_1, upper_limit_1 κλπ. Για τον έλεγχο του ανώτατου και κατώτατου ορίου της εκάστοτε άρθρωσης εισάγονται κάθε φορά τα αντίστοιχα όρια. Στην περίπτωση της πρώτης άρθρωσης έχουν εισαχθεί οι 170° που είναι το ανώτατο όριο και οι -170° που είναι το κατώτατο. Όταν δίνεται τιμή για την γωνιακή θέση της πρώτης άρθρωσης η οποία είναι $\geq 170^\circ$ ή $\leq -170^\circ$ τότε η πρώτη άρθρωση θα κινηθεί έως τις 170° ή τις -170° αντίστοιχα. Ο υπολογισμός των ενδείξεων που σχετίζονται με το ανώτατο και το κατώτατο όριο της κάθε άρθρωσης πραγματοποιείται δυναμικά λαμβάνοντας υπόψη τα προκαθορισμένα όρια και την τρέχουσα θέση της αντίστοιχης άρθρωσης που προσδιορίζεται με το VI "Read". Η κίνηση της άρθρωσης πραγματοποιείται με το VI "Straight-Line Move", το οποίο έχει παραμετροποιηθεί σύμφωνα με την Absolute μέθοδο. Η επιλεγμένη μέθοδος προσομοιώνει την λειτουργία των περιστροφικών κωδικοποιητών (rotary encoder) απόλυτης (absolute) θέσης. Συνεπώς, η κίνηση που εκτελεί η άρθρωση πραγματοποιείται σε σχέση με το σημείο αναφοράς που αντιστοιχεί σε γωνία 0°. Τέλος, η κίνηση μπορεί να διακοπεί αμέσως αν δοθεί εντολή stop στο VI "Stop Move".

Στο τέλος του Block Diagram βρίσκεται η λειτουργία που δημιουργεί τον παλμό του σήματος done (Σχήμα 4.32). Ο παλμός αυτός στέλνεται στο PLC μέσω της μεταβλητής εξόδου done όταν ολοκληρωθούν όλες οι κινήσεις του εικονικού ρομποτικού βραχίονα. Κάθε φορά που ολοκληρώνεται η κίνηση μιας άρθρωσης ενεργοποιείται η αντίστοιχη ένδειξη (done 1, done 2, ..., done 6). Οι καταστάσεις των ενδείξεων αποθηκεύονται προσωρινά μέσω των Shift Registers. Όταν ολοκληρωθούν όλες οι κινήσεις δημιουργείται ο παλμός done που αποστέλλεται στο PLC καθώς και ο παλμός done που αποστέλλεται στο HMI. Τέλος, το σήμα execute και οι επιμέρους ενδείξεις ολοκλήρωσης (done 1, done 2, ..., done 6) παίρνουν την τιμή false.



Σχήμα 4.32: Λειτουργία δημιουργίας των παλμών done.

ΚΕΦΑΛΑΙΟ 5

Αποτελέσματα - Συμπεράσματα

5.1 Εισαγωγή

Στο κεφάλαιο αυτό παρουσιάζεται η διαδικασία του εικονικού ελέγχου και λειτουργίας του βιομηχανικού ρομποτικού βραχίονα KR 5 sixx R850, από το στήσιμο των λογισμικών που συνθέτουν την παραπάνω διαδικασία έως τα αποτελέσματα και τα συμπεράσματα.

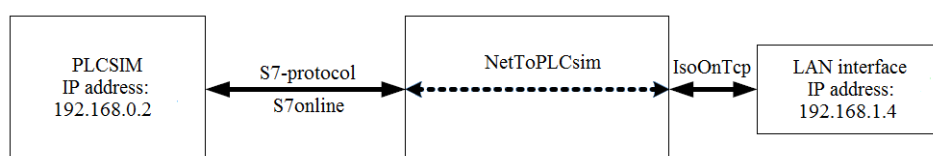
Όπως είχε αναφερθεί στο κεφάλαιο 1, υπάρχουν δυο βασικές μέθοδοι προσέγγισης της διαδικασίας VC όσον αφορά τον έλεγχο του εικονικού συστήματος. Στην παρούσα πτυχιακή εργασία η διαδικασία VC υλοποιείται και με τις δυο μεθόδους. Τη μέθοδο SIL, όπου ο έλεγχος του εικονικού ρομποτικού βραχίονα επιτυγχάνεται με χρήση του PLC εξομοιωτή. Και τη μέθοδο HIL, σύμφωνα με την οποία χρησιμοποιείται το πραγματικό PLC για τον έλεγχο του εικονικού ρομποτικού βραχίονα.

5.2 Ο έλεγχος της εφαρμογής με την μέθοδο SIL

Στη μέθοδο SIL ο έλεγχος του εικονικού ρομποτικού βραχίονα επιτυγχάνεται μέσω PLC, του οποίου η λειτουργία εξομοιώνεται με το λογισμικό S7-PLCSIM που εμπεριέχεται στην σουίτα TIA PORTAL. Το λογισμικό S7-PLCSIM αποτελεί έναν εξομοιωτή PLC που επιτρέπει στον χρήστη να κατεβάσει, να τρέξει και να δοκιμάσει το πρόγραμμά του στον υπολογιστή, χωρίς να είναι απαραίτητη η χρήση ενός πραγματικού PLC. Ο συγκεκριμένος PLC εξομοιωτής δεν έχει τις δυνατότητες διασύνδεσης που έχει ένα πραγματικό PLC. Πράγμα που σημαίνει ότι στην παρούσα πτυχιακή εργασία δεν μπορεί να συνδεθεί με το LabVIEW μέσω του OPC Server ώστε να αξιοποιηθεί για τον έλεγχο του εικονικού ρομποτικού βραχίονα. Λύση σε αυτό το πρόβλημα έρχεται να δώσει το λογισμικό NetToPLCSim.

5.2.1 Το λογισμικό NetToPLCSim

Το NetToPLCSim είναι διαθέσιμο μέσω της ιστοσελίδας <http://nettoplcsim.sourceforge.net/index.html>. Είναι ένα λογισμικό ανοικτού κώδικα που επεκτείνει το λογισμικό εξομοίωσης S7-PLCSIM με μια διεπαφή δικτύου, επιτρέποντας την επικοινωνία με άλλα λογισμικά μέσω του πρωτοκόλλου TCP/IP (Transmission Control Protocol/Internet Protocol). Το NetToPLCSim επικοινωνεί με το S7-PLCSIM μέσω της διεπαφής S7online, η οποία χρησιμοποιεί το S7-Protocol και αξιοποιείται από το λογισμικό STEP 7 για την επικοινωνία με το S7-PLCSIM. Στη συνέχεια προωθεί τα δεδομένα του S7-PLCSIM στο δίκτυο μέσω της TCP θύρας 102 χρησιμοποιώντας το πρωτόκολλο IsoOnTCP και την φυσική IP διεύθυνση του υπολογιστή που εκτελεί το S7-PLCSIM (Σχήμα 5.1).

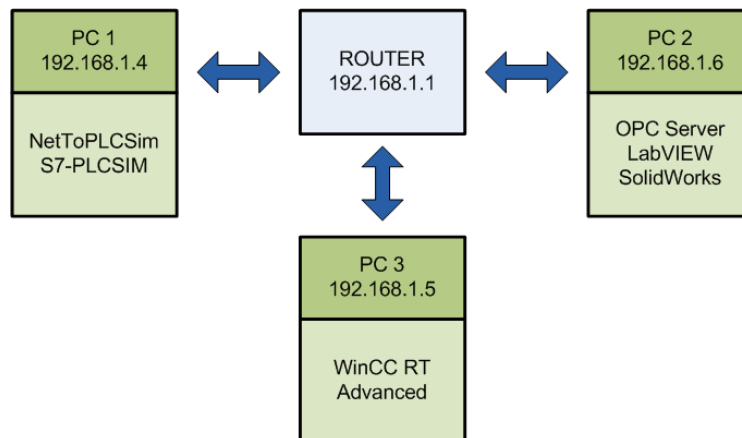


Σχήμα 5.1: Η λειτουργία του λογισμικού NetToPLCSim.

Στην ουσία ο υπολογιστής στον οποίο τρέχουν τα λογισμικά PLCSIM και NetToPLCSim αντιπροσωπεύει ένα PLC. Οπότε για την επίτευξη επικοινωνίας με κάποιο λογισμικό SCADA, OPC κλπ. αρκεί κατά την διαμόρφωση της επικοινωνίας η εισαγωγή της IP διεύθυνσης του υπολογιστή που αντιπροσωπεύει το PLC, όπως ακριβώς και στην περίπτωση ενός πραγματικού PLC.

5.2.2 Η γενική δομή της εφαρμογής

Για την υλοποίηση του VC Project της παρούσας πτυχιακής εργασίας με την μέθοδο SIL χρησιμοποιούνται τρεις υπολογιστές (Σχήμα 5.2). Ο πρώτος υπολογιστής είναι αυτός που αντιπροσωπεύει το PLC και έχει IP διεύθυνση 192.168.1.4. Σε αυτόν τον υπολογιστή εκτελούνται τα λογισμικά S7-PLCSIM και NetToPLCSim. Στον δεύτερο υπολογιστή με IP διεύθυνση 192.168.1.6 εκτελούνται τα λογισμικά NI OPC Servers, LabVIEW και SolidWorks. Η επικοινωνία μεταξύ των δυο υπολογιστών επιτυγχάνεται μέσω του OPC Server. Ο OPC Server βλέπει τον πρώτο υπολογιστή ως PLC, γιατί κατά την παραμετροποίηση του (κεφάλαιο 4) ορίστηκε ως IP διεύθυνση συσκευής η IP διεύθυνση 192.168.1.4. Στον τρίτο υπολογιστή εκτελείται το WinCC Runtime Advanced (HMI).

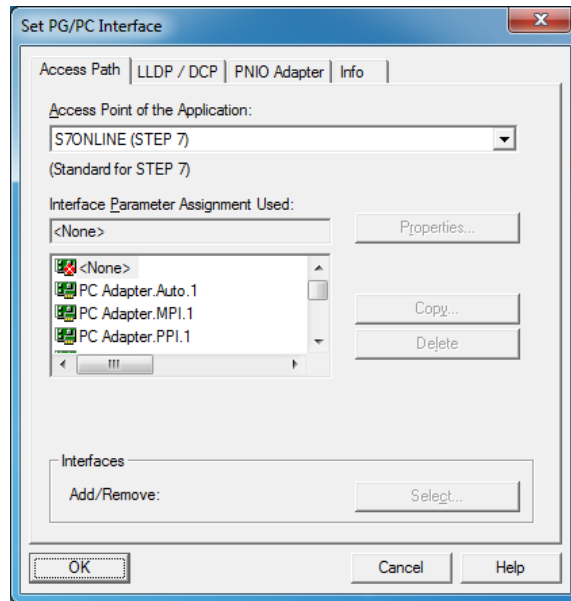


Σχήμα 5.2: Η γενική δομή της εφαρμογής.

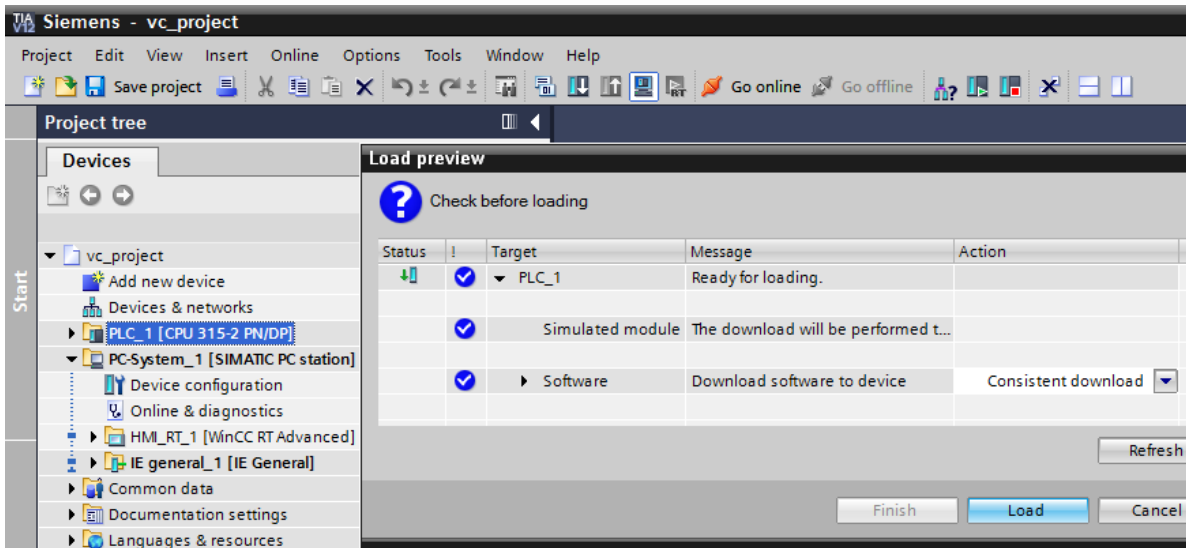
Στο εργαστήριο η υλοποίηση του VC Project πραγματοποιήθηκε και με την χρήση ενός υπολογιστή, όπου όλα τα λογισμικά εκτελούνται σε αυτόν. Ωστόσο, υιοθετήθηκε και επιλέχθηκε να παρουσιαστεί η υλοποίηση του VC Project αξιοποιώντας δυο υπολογιστές. Συνεπώς, ο πρώτος υπολογιστής παίζει τον ρόλο του PLC και ταυτόχρονα παρέχει στον χρήστη την δυνατότητα ελέγχου μέσω του HMI. Ο δεύτερος υπολογιστής αναλαμβάνει τον ρόλο του υπό έλεγχο συστήματος.

5.2.3 Προετοιμασία του υπολογιστή που αντιπροσωπεύει το PLC

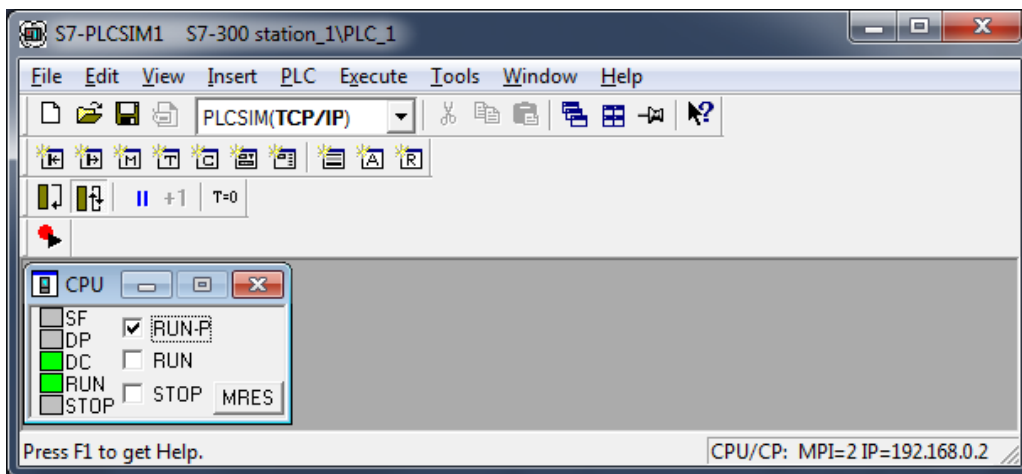
Αρχικά διαμορφώνεται η διεπαφή PG/PC από τον πίνακα ελέγχου όπως φαίνεται στο Σχήμα 5.3. Στην συνέχεια αφού ανοίξει το TIA Portal φορτώνεται το αρχείο που περιέχει το συνολικό Project της εφαρμογής το οποίο περιλαμβάνει τον κώδικα του PLC, την διαμόρφωση του PLC και την διαμόρφωση του HMI. Επιλέγοντας το όνομα του PLC που είναι PLC_1 ενεργοποιείται το S7-PLCSIM μέσω της επιλογής Start simulation (Σχήμα 5.4). Σε αυτή την περίπτωση αφού ανοίξει το εικονικό PLC φορτώνεται ο κώδικας. Ο χρήστης θα πρέπει να θέσει σε κατάσταση RUN ή RUN-P το εικονικό PLC όπως φαίνεται στο Σχήμα 5.5.



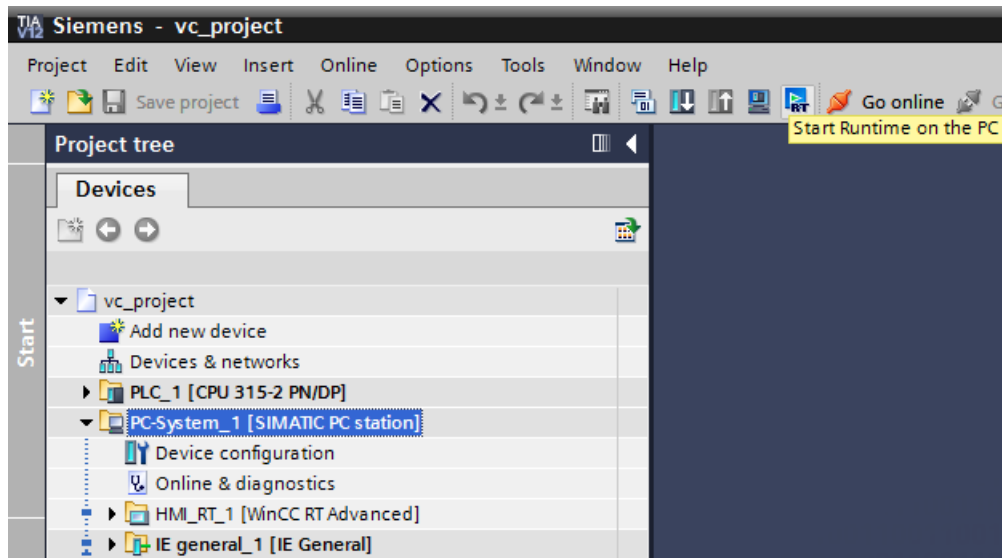
Σχήμα 5.3: Διαμόρφωση διεπαφής PG/PC.



Σχήμα 5.4: Ενεργοποίηση του S7-PLCSIM και φόρτωση του PLC κώδικα.



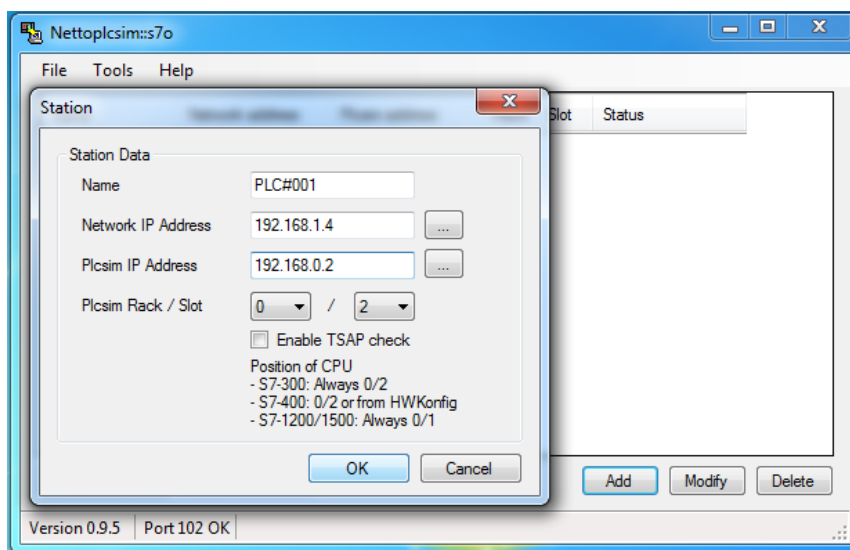
Σχήμα 5.5: Το PLCSIM.



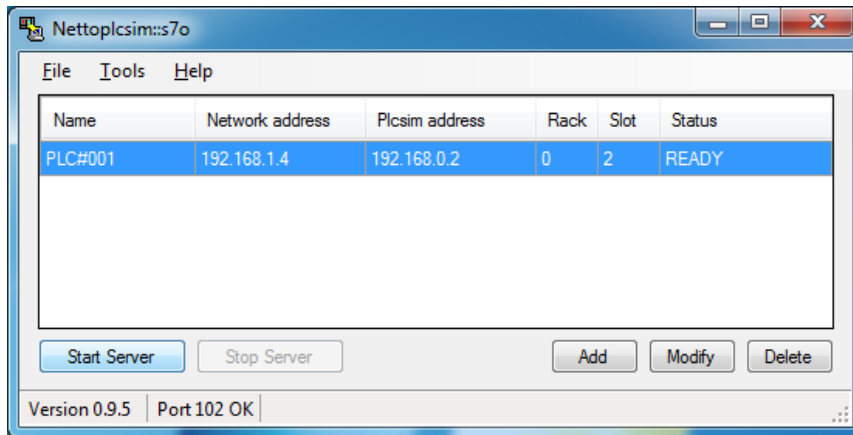
Σχήμα 5.6: Εκκίνηση του HMI.

Για την εκκίνηση του HMI θα πρέπει να επιλεγεί αρχικά το όνομα του HMI (PC-System_1) μέσω του Project tree του TIA Portal, όπως παρουσιάζεται στο Σχήμα 5.6. Το HMI ενεργοποιείται μέσω της επιλογής Start Runtime on the PC. Σε αυτό το σημείο θα πρέπει να επισημανθεί ότι δεν πρόκειται για εξομοίωση του HMI, αλλά για κανονική εκτέλεση αξιοποιώντας το WinCC RT Advance.

Κατά την εκκίνηση του NetToPLCSim θα πρέπει να απενεργοποιηθεί η υπηρεσία των Windows που ονομάζεται SIMATIC IEPG Help και τρέχει στο παρασκήνιο. Το γεγονός αυτό οφείλεται στο ότι η συγκεκριμένη υπηρεσία δεσμεύει την TCP θύρα 102 που χρειάζεται το NetToPLCSim. Για να είναι εφικτή η απενεργοποίηση της συγκεκριμένης υπηρεσίας θα πρέπει ο χρήστης να έχει δικαιώματα διαχειριστή των Windows. Για την διαμόρφωση του NetToPLCSim απαιτείται η δημιουργία ενός σταθμού. Μέσω της επιλογής Add εμφανίζεται το παράθυρο διαμόρφωσης του σταθμού, όπου στο πεδίο Network IP Address εισάγεται η IP διεύθυνση του υπολογιστή που αντιπροσωπεύει το PLC και στο πεδίο Plcsim IP Address εισάγεται η διεύθυνση του PLCSIM (Σχήμα 5.7). Αφού ολοκληρωθεί η διαμόρφωση του σταθμού θα πρέπει να πραγματοποιηθεί η εκκίνηση του συγκεκριμένου σταθμού μέσω της επιλογής Start Server (Σχήμα 5.8).



Σχήμα 5.7: Διαμόρφωση NetToPLCSim.

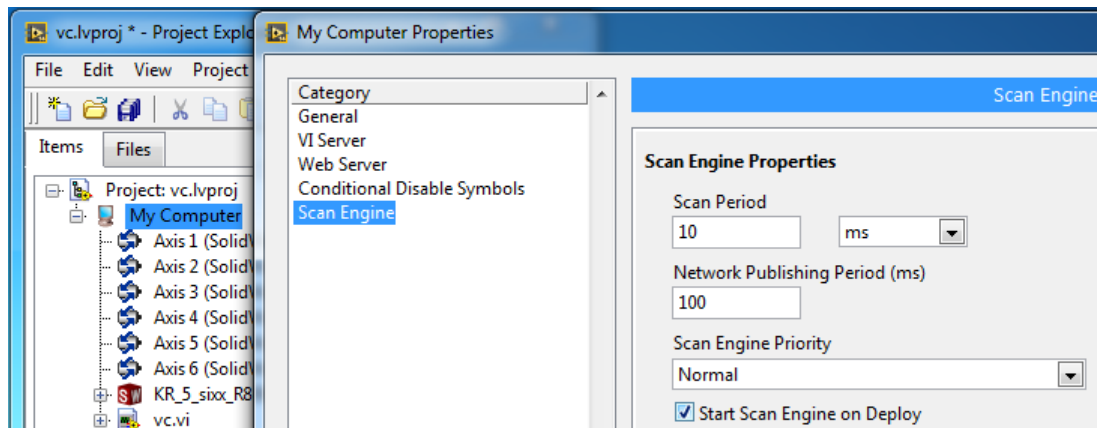


Σχήμα 5.8: Εκκίνηση NetToPLCSim.

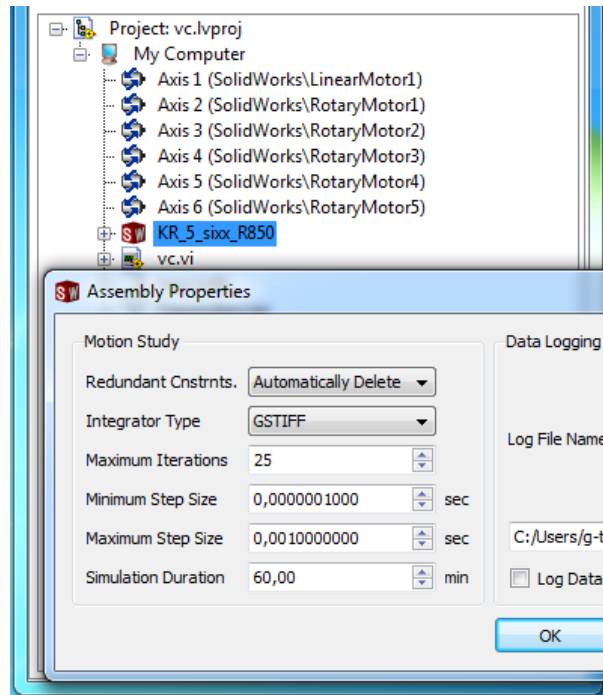
5.2.4 Προετοιμασία του δεύτερου υπολογιστή

Στον δεύτερο υπολογιστή αρχικά πραγματοποιείται το άνοιγμα του LabVIEW Project. Κατά την πρώτη εκτέλεση θα πρέπει να πραγματοποιηθούν οι παρακάτω ρυθμίσεις, οι οποίες είναι απαραίτητες για την σωστή επικοινωνία και λειτουργία της προσομοίωσης:

- Ενεργοποίηση της λειτουργίας Start Scan Engine on Deploy πατώντας από το Project tree δεξί κλικ στο My Computer → Properties (Σχήμα 5.9).
- Ρύθμιση του Motion Study Maximum Step Size από τα 0,01 sec στα 0,001 sec (Σχήμα 5.10).

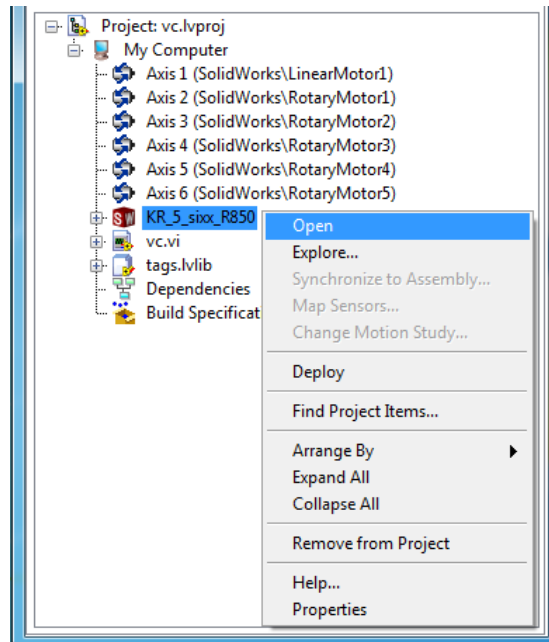


Σχήμα 5.9: Ενεργοποίηση λειτουργίας Start Scan Engine on Deploy.



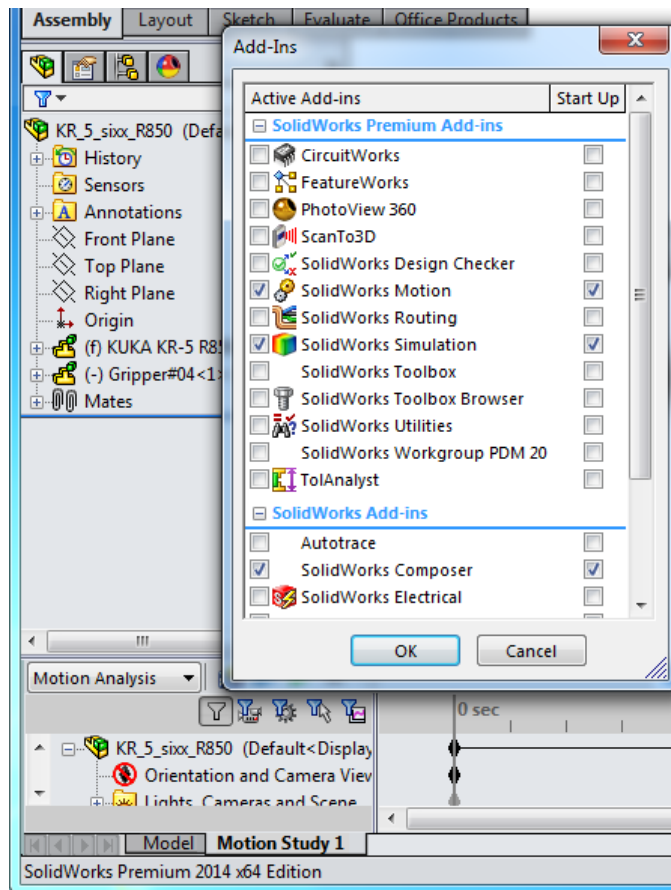
Σχήμα 5.10: Διαμόρφωση Maximum Step Size.

Στην συνέχεια ακολουθεί το άνοιγμα του τρισδιάστατου μοντέλου του βιομηχανικού ρομποτικού βραχίονα στο SolidWorks. Το άνοιγμα του assembly μπορεί να πραγματοποιηθεί και από το LabVIEW πατώντας στο Project tree δεξί κλικ στο SolidWorks assembly → Open (Σχήμα 5.11).



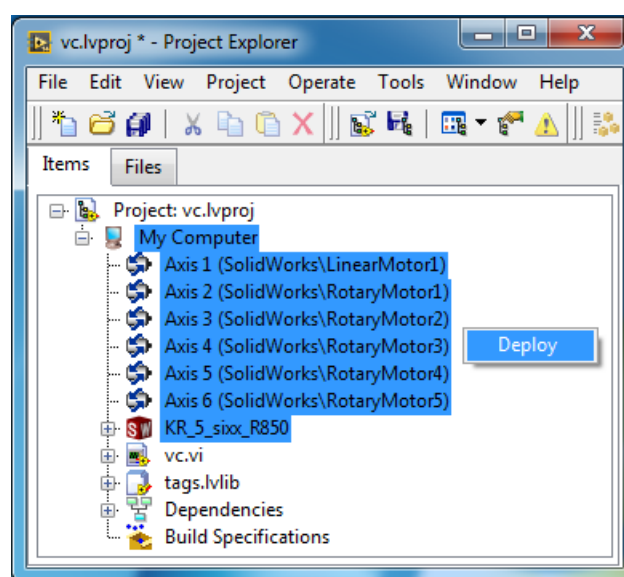
Σχήμα 5.11: Άνοιγμα SolidWorks assembly.

Όταν ολοκληρωθεί το άνοιγμα του τρισδιάστατου μοντέλου στο SolidWorks, ελέγχεται από τα Options → Add-Ins αν τα πρόσθετα SolidWorks Motion και SolidWorks Simulation, τα οποία είναι απαραίτητα για την προσομοίωση, είναι ενεργοποιημένα (Σχήμα 5.12). Μετά επιλέγεται η καρτέλα Motion Study και ο τύπος Motion Analysis στο Type of Study.

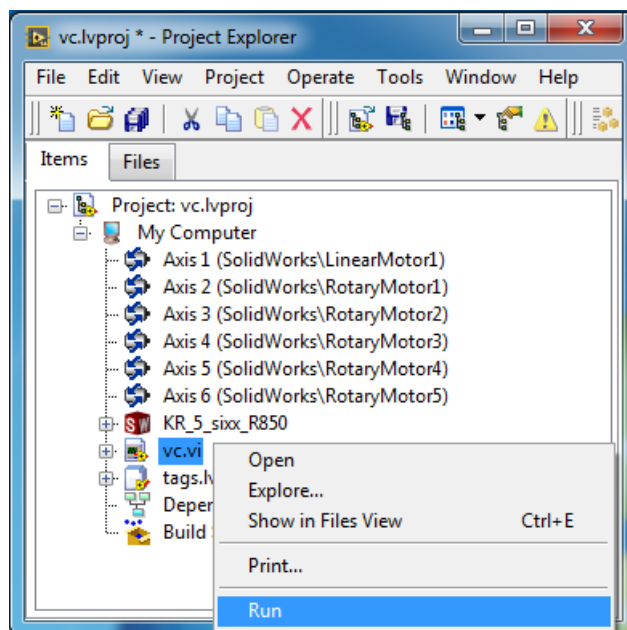


Σχήμα 5.12: Ενεργοποίηση των απαραίτητων Add-Ins.

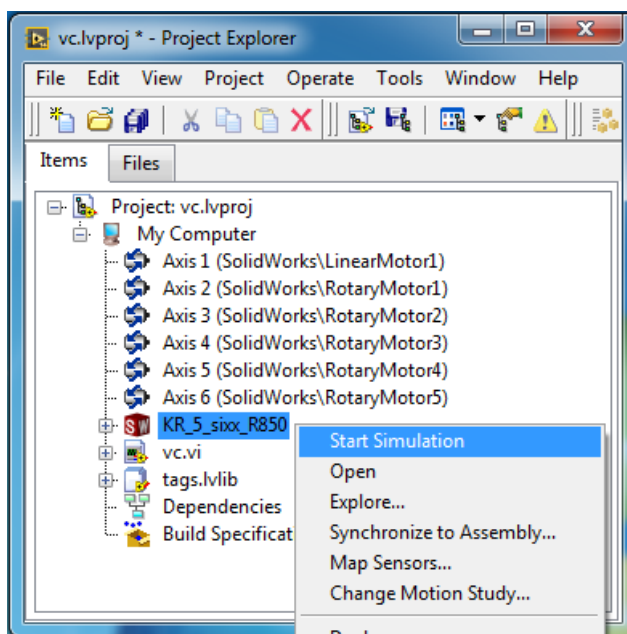
Για την έναρξη της προσομοίωσης αρχικά επιλέγονται στο Project tree του LabVIEW όλοι οι άξονες-κινητήρες, το SolidWorks assembly του τρισδιάστατου μοντέλου και το My Computer και επιλέγεται με δεξί κλικ η διαδικασία Deploy (Σχήμα 5.13). Έπειτα επιλέγεται η εκκίνηση του VI με δεξί κλικ → Run (Σχήμα 5.14), και τέλος πραγματοποιείται η εκκίνηση της προσομοίωσης με δεξί κλικ στο SolidWorks assembly και επιλέγοντας Start Simulation (Σχήμα 5.15).



Σχήμα 5.13: Διαδικασία του Deploy.



Σχήμα 5.14: Εκτέλεση του VI.

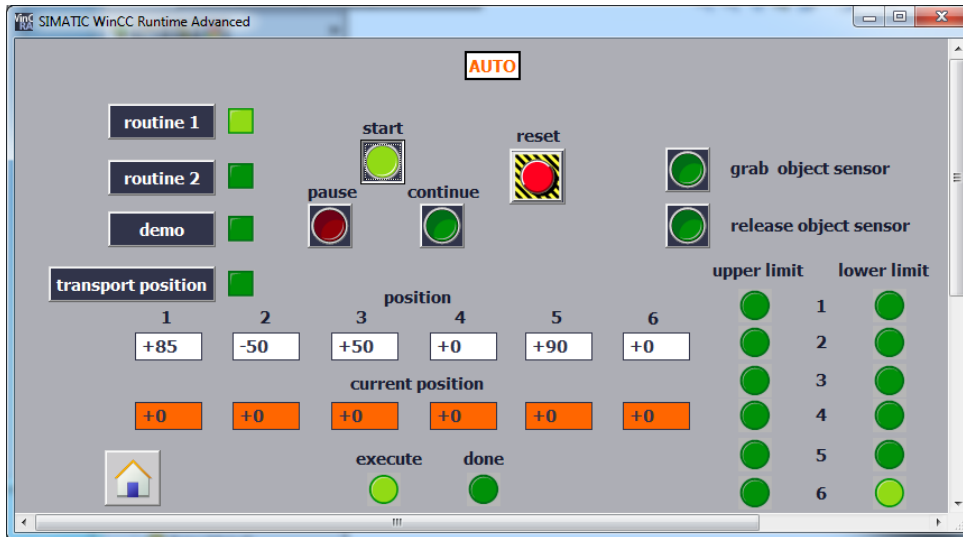


Σχήμα 5.15: Εκκίνηση της προσομοίωσης.

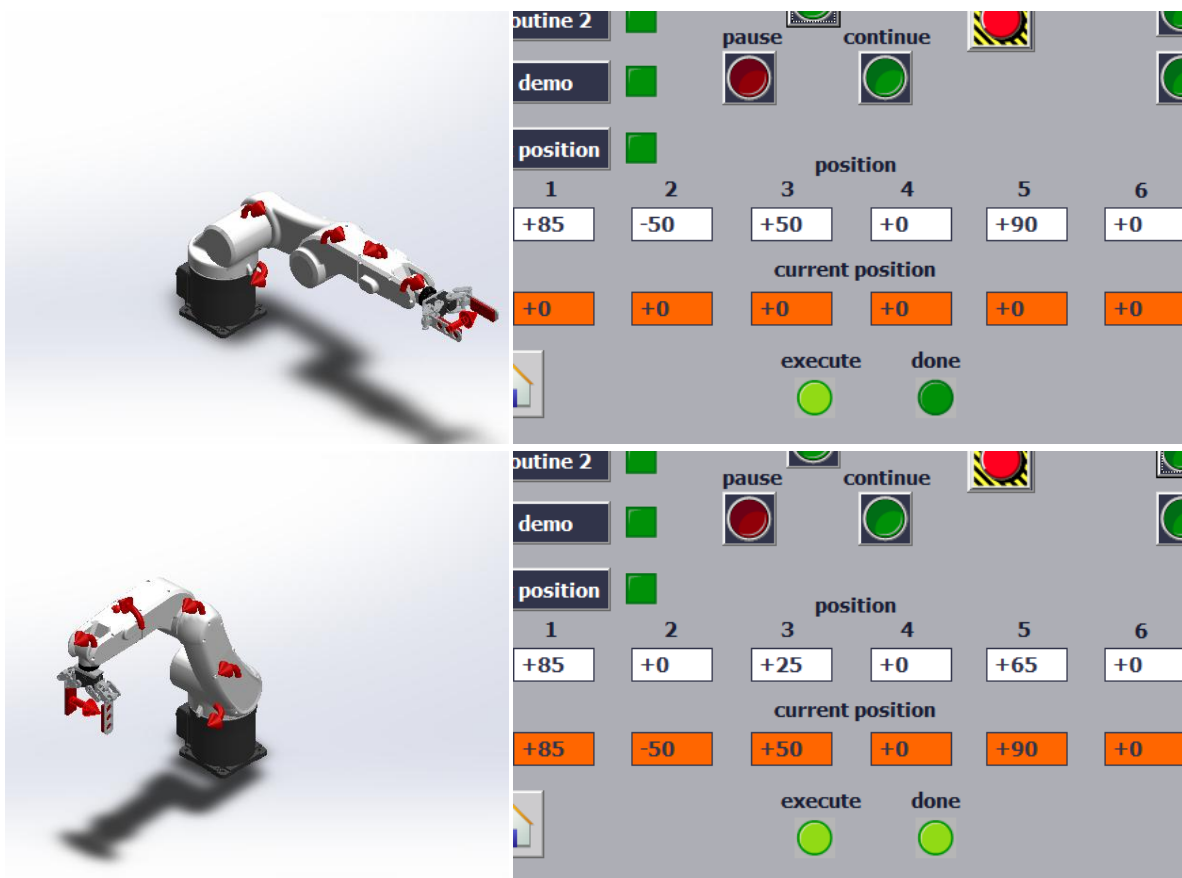
5.2.5 Αποτελέσματα

Με την εκκίνηση της προσομοίωσης ο εικονικός ρομποτικός βραχίονας είναι σε θέση να δεχτεί εντολές από τον χρήστη μέσω του HMI, και να εκτελέσει οποιαδήποτε κίνηση ή διαδικασία του ζητηθεί. Στα πλαίσια των αποτελεσμάτων επιλέχθηκε να παρουσιαστεί η εκτέλεση από τον εικονικό ρομποτικό βραχίονα της ρουτίνας εργασίας 1 από την αυτόματη λειτουργία. Η εκκίνηση της συγκεκριμένης ρουτίνας πραγματοποιείται επιλέγοντάς την από την οθόνη της αυτόματης λειτουργίας του HMI και πατώντας το μπουτόν start (Σχήμα 5.16). Στα Σχήματα 5.17 έως 5.19 παρουσιάζεται ένας πλήρης κύκλος από την εκτέλεση της ρουτίνας. Αριστερά, διακρίνεται η θέση του εικονικού ρομποτικού βραχίονα κατά την

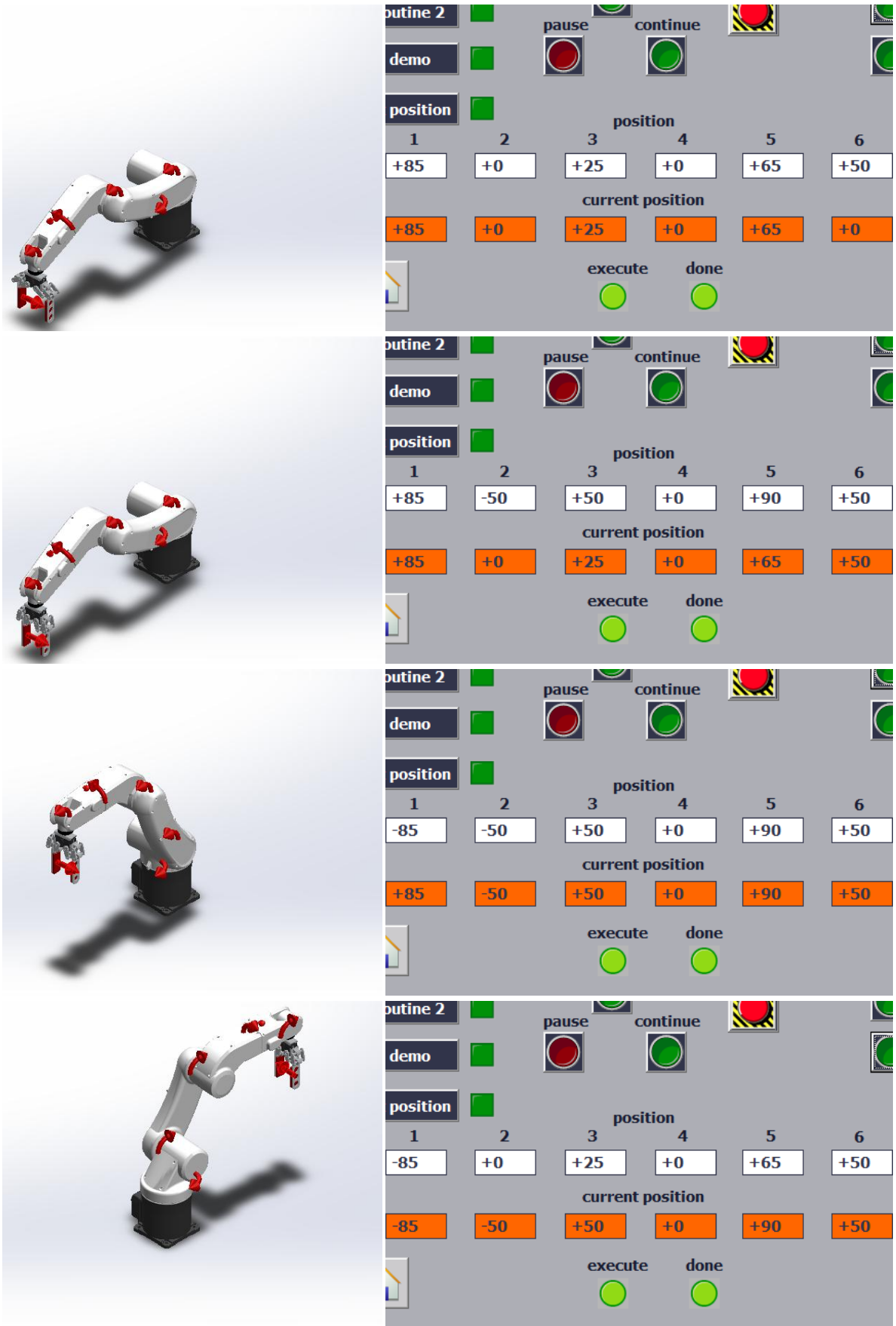
εκτέλεση της ρουτίνας. Δεξιά, στα πορτοκαλί πεδία του HMI διακρίνονται οι τρέχουσες γωνιακές θέσεις των αρθρώσεων του βραχίονα σε μοίρες και η τρέχουσα θέση της αρπάγης σε εκατοστά. Στα άσπρα πεδία εμφανίζονται οι τιμές των γωνιακών θέσεων των αρθρώσεων και της θέσης της αρπάγης που φορτώνονται από τον κώδικα του PLC για να συνθέσουν την επόμενη κίνηση του βραχίονα.



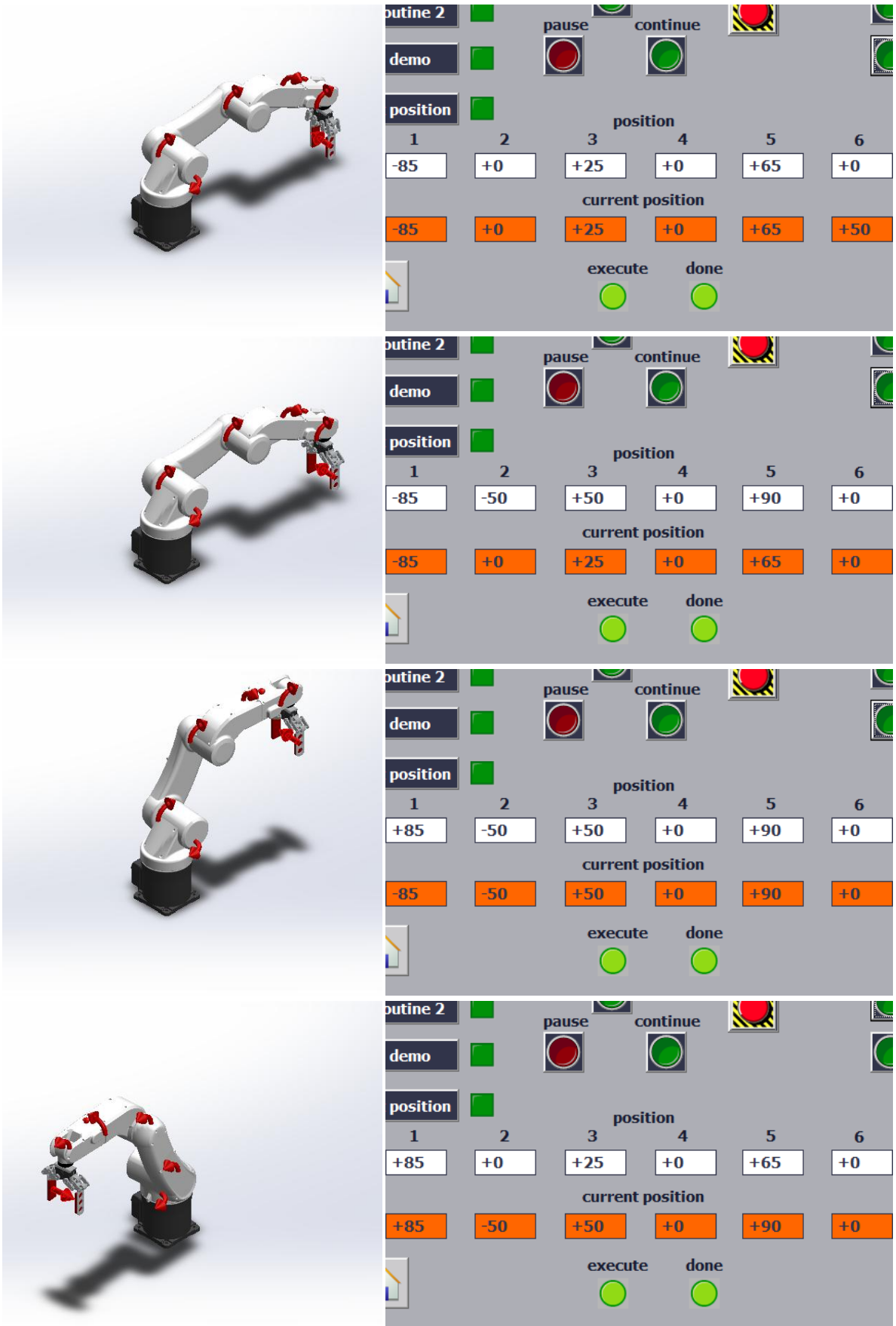
Σχήμα 5.16: Εκκίνηση ρουτίνας εργασίας 1.



Σχήμα 5.17: Εκτέλεση ρουτίνας εργασίας 1.



Σχήμα 5.18: Εκτέλεση ρουτίνας εργασίας 1 (Συνέχεια).

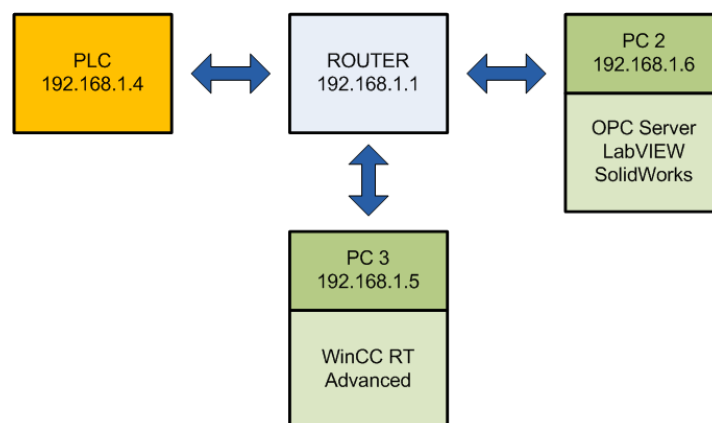


Σχήμα 5.19: Εκτέλεση ρουτίνας εργασίας 1 (Συνέχεια).

5.3 Ο έλεγχος της εφαρμογής με την μέθοδο HIL

Στην μέθοδο HIL ο έλεγχος του εικονικού ρομποτικού βραχίονα επιτυγχάνεται μέσω ενός πραγματικού SIMATIC PLC της σειράς S7-300 της εταιρίας SIEMENS. Τα μέρη που αποτελούν το συγκεκριμένο σύστημα PLC είναι η CPU 315-2 PN/DP, το τροφοδοτικό PS 307 των 5A και η μονάδα ψηφιακών εισόδων SM 321 DI16 × 24VDC. Η ύπαρξη της μονάδα ψηφιακών εισόδων αποσκοπεί στην χρήση πραγματικών διακοπών που αντιστοιχούν στα βασικά μπουτόν του HMI (start, execute/continue, pause, continue, reset) και των μπουτόν που προσομοιώνουν την λειτουργία των υποτιθέμενων αισθητήρων (grab object sensor και release object sensor). Με αυτόν τον τρόπο ο χρήστης έχει την δυνατότητα να επιλέξει να δώσει τις βασικές εντολές είτε μέσω του HMI είτε κάνοντας χρήση των πραγματικών διακοπών.

Για την υλοποίηση του VC Project της παρούσας πτυχιακής με την μέθοδο HIL χρησιμοποιούνται δυο υπολογιστές και το παραπάνω PLC (Σχήμα 5.20 και 5.21). Κατά την διαμόρφωση του PLC μέσω του STEP 7, ορίστηκε σαν IP διεύθυνση του PLC η IP διεύθυνση 192.168.1.4. Ούτως ώστε να είναι εφικτή η επίτευξη επικοινωνίας με τον OPC Server. Ο υπολογιστής 2 με IP διεύθυνση 192.168.1.6 αναλαμβάνει τον ρόλο του υπό έλεγχο συστήματος. Δηλαδή σε αυτόν εκτελούνται τα λογισμικά NI OPC Servers, LabVIEW και SolidWorks, όπως και στην περίπτωση του ελέγχου με την μέθοδο SIL. Στον υπολογιστή 3 με IP διεύθυνση 192.168.1.5 εκτελείται το WinCC Runtime Advanced (HMI) παρέχοντας στον χρήστη την δυνατότητα ελέγχου του εικονικού ρομποτικού βραχίονα μέσω του HMI.



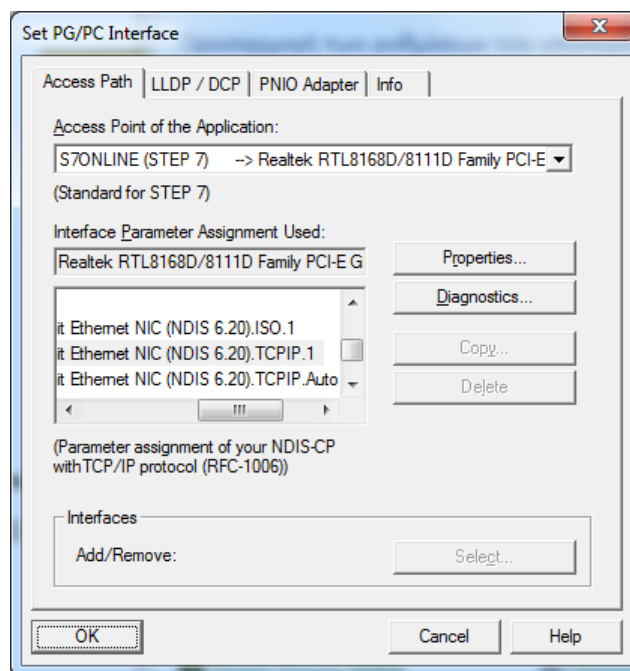
Σχήμα 5.20: Η δομή της εφαρμογής με την μέθοδο HIL.



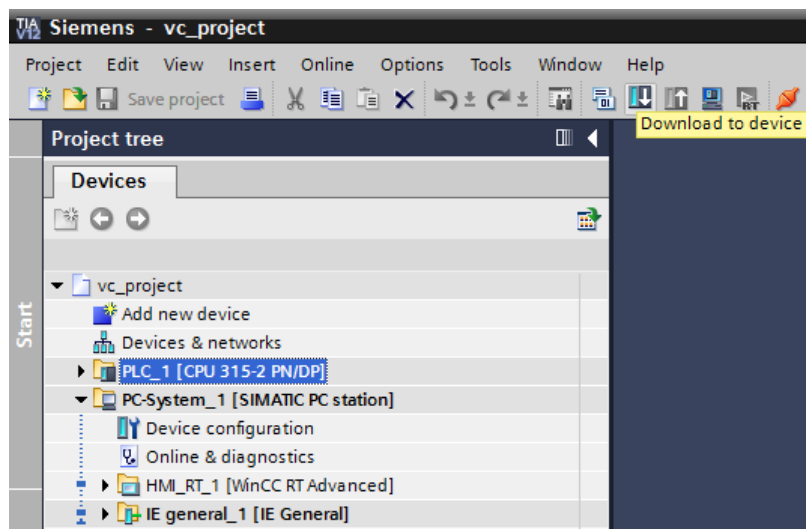
Σχήμα 5.21: Η δομή της εφαρμογής με την μέθοδο HIL.

5.3.1 Προετοιμασία του υπολογιστή 3 και του PLC

Αρχικά διαμορφώνεται η διεπαφή PG/PC από τον πίνακα ελέγχου όπως φαίνεται στο Σχήμα 5.22. Μετά ο χρήστης θα πρέπει να ενεργοποιήσει το PLC γυρνώντας τον διακόπτη του τροφοδοτικού στη θέση ON και να θέσει το PLC σε κατάσταση RUN, γυρνώντας τον διακόπτη της CPU στη θέση RUN. Στην συνέχεια αφού ανοίξει το TIA Portal φορτώνεται το αρχείο που περιέχει το συνολικό Project της εφαρμογής που περιλαμβάνει τον κώδικα του PLC, την διαμόρφωση του PLC και την διαμόρφωση του HMI. Επιλέγοντας μέσω του Project tree του TIA Portal το όνομα του PLC που είναι PLC_1 και πατώντας στην επιλογή Download to device, φορτώνεται ο κώδικας στο PLC (Σχήμα 5.23). Αμέσως μετά ενεργοποιείται το HMI επιλέγοντας το PC-System_1 και πατώντας στην επιλογή Start Runtime on the PC (Σχήμα 5.6). Η προετοιμασία του υπολογιστή 2 που αναλαμβάνει τον ρόλο του υπό έλεγχο συστήματος είναι ακριβώς ίδια, όπως και στην περίπτωση του ελέγχου της εφαρμογής με την μέθοδο SIL.



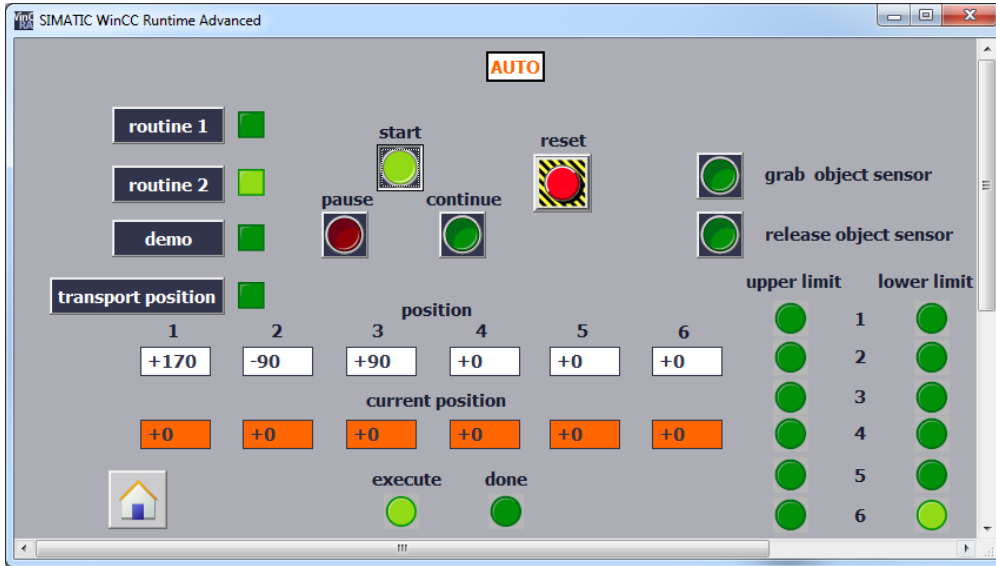
Σχήμα 5.22: Διαμόρφωση της διεπαφής PG/PC.



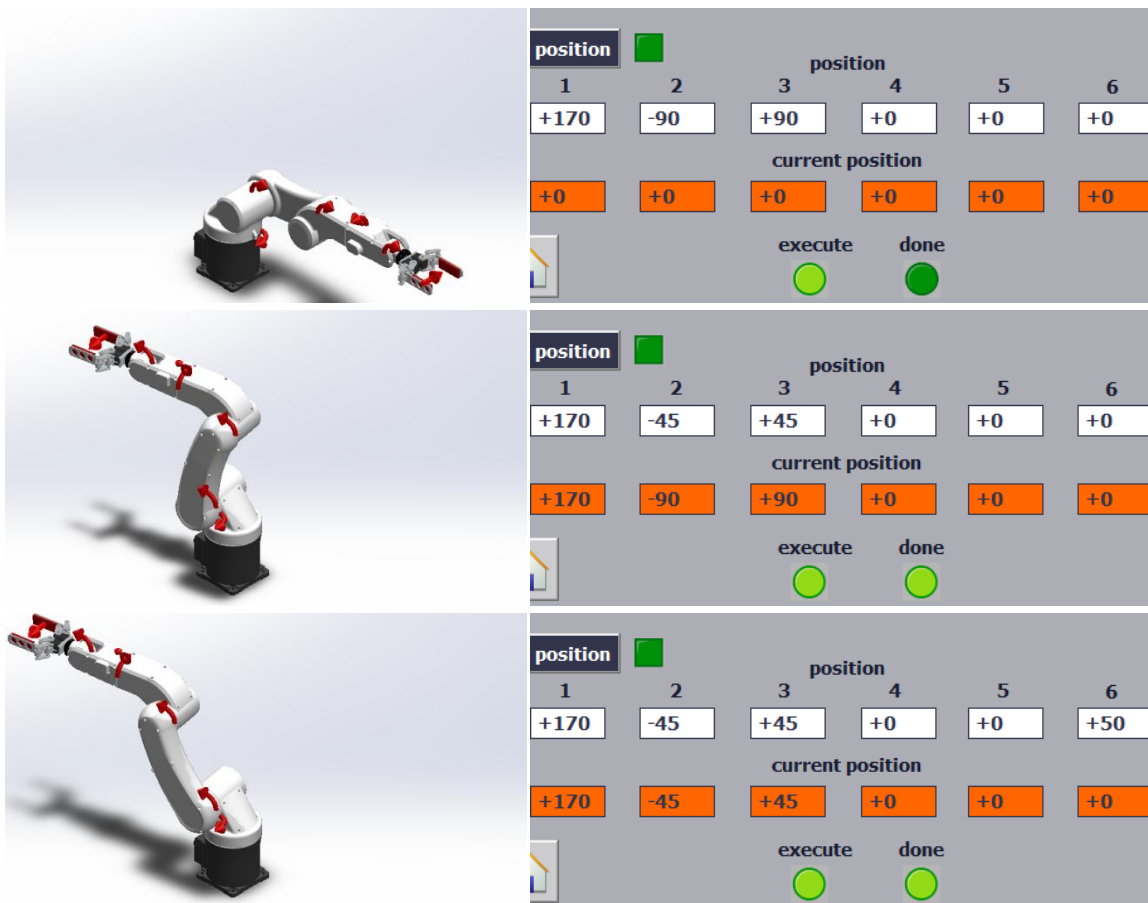
Σχήμα 5.23: Φόρτωση του κώδικα στο PLC.

5.3.2 Αποτελέσματα

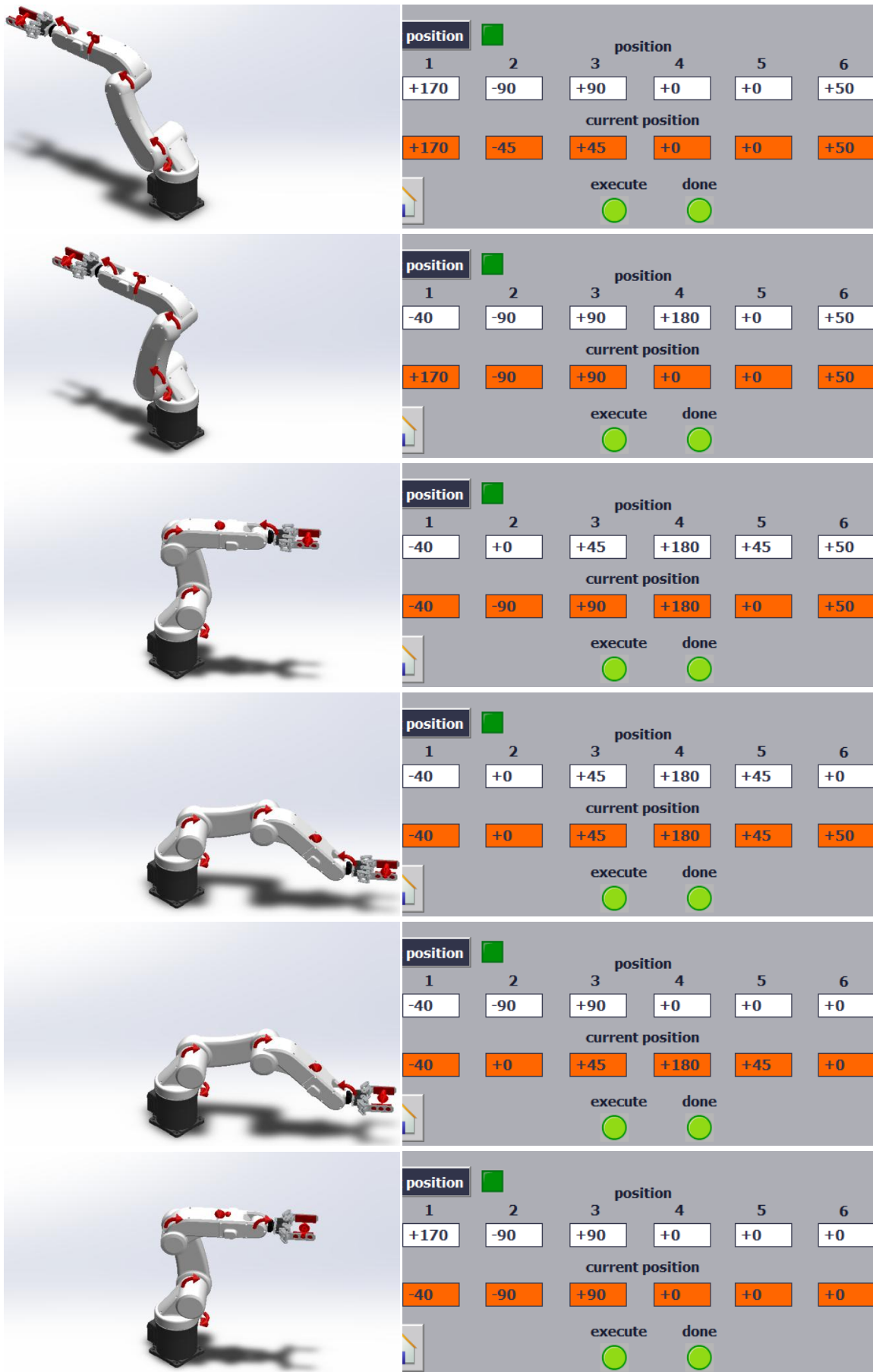
Στα πλαίσια των αποτελεσμάτων αρχικά παρουσιάζεται η εκτέλεση από τον εικονικό ρομποτικό βραχίονα της ρουτίνας εργασίας 2, από την αυτόματη λειτουργία. Στο Σχήμα 5.24 φαίνεται η εκκίνηση της ρουτίνας από την οθόνη της αυτόματης λειτουργίας του HMI. Στα Σχήματα 5.25 έως 5.27 παρουσιάζεται με την ίδια λογική που παρουσιάστηκε η ρουτίνα εργασίας 1, η εκτέλεση της ρουτίνας εργασίας 2.



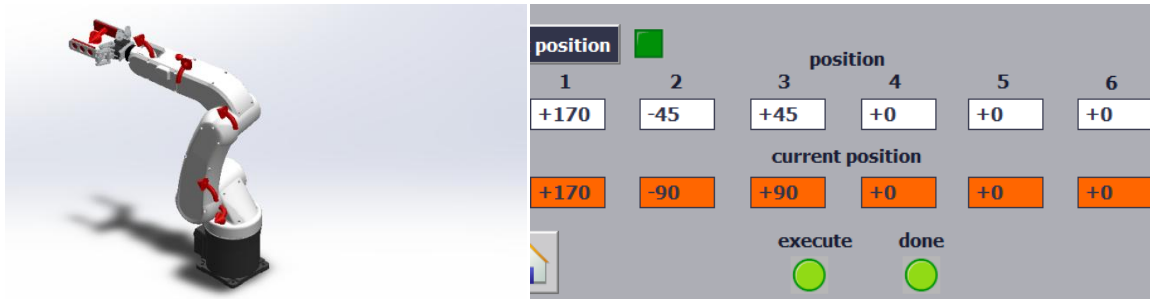
Σχήμα 5.24: Εκκίνηση ρουτίνας εργασίας 2.



Σχήμα 5.25: Εκτέλεση ρουτίνας εργασίας 2.

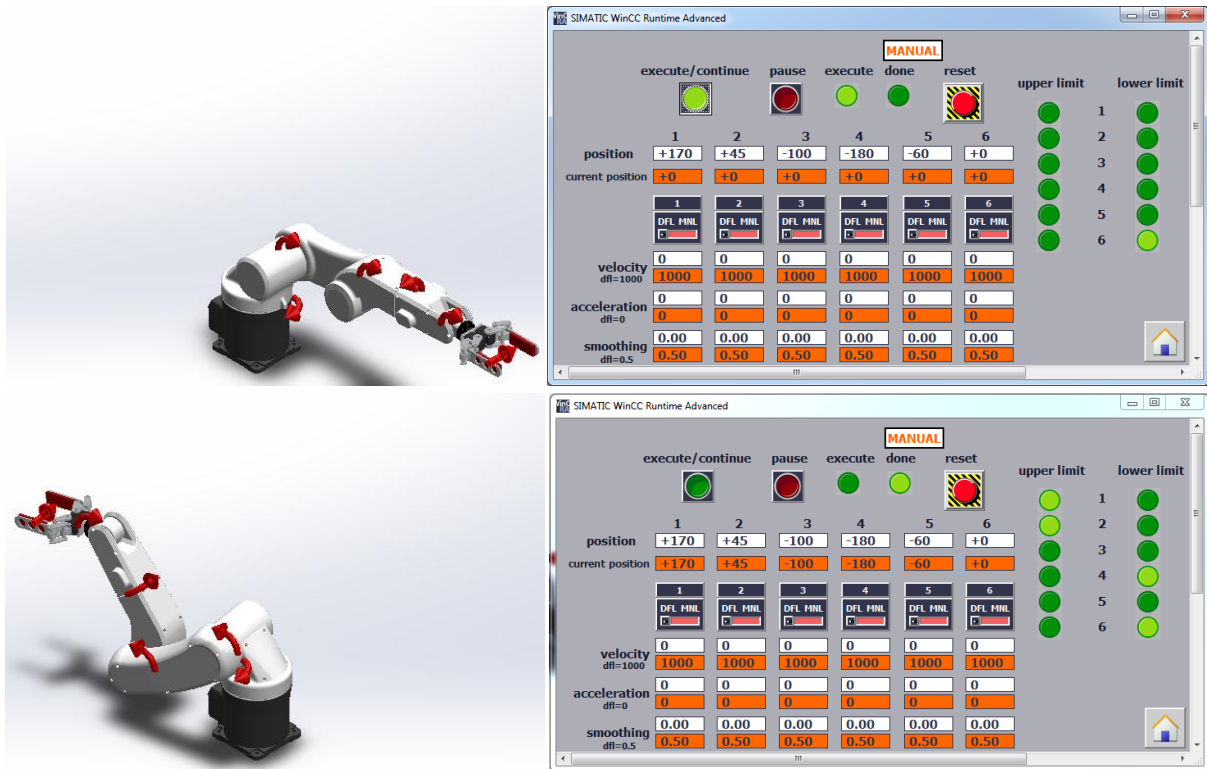


Σχήμα 5.26: Εκτέλεση ρουτίνας εργασίας 2 (Συνέχεια).

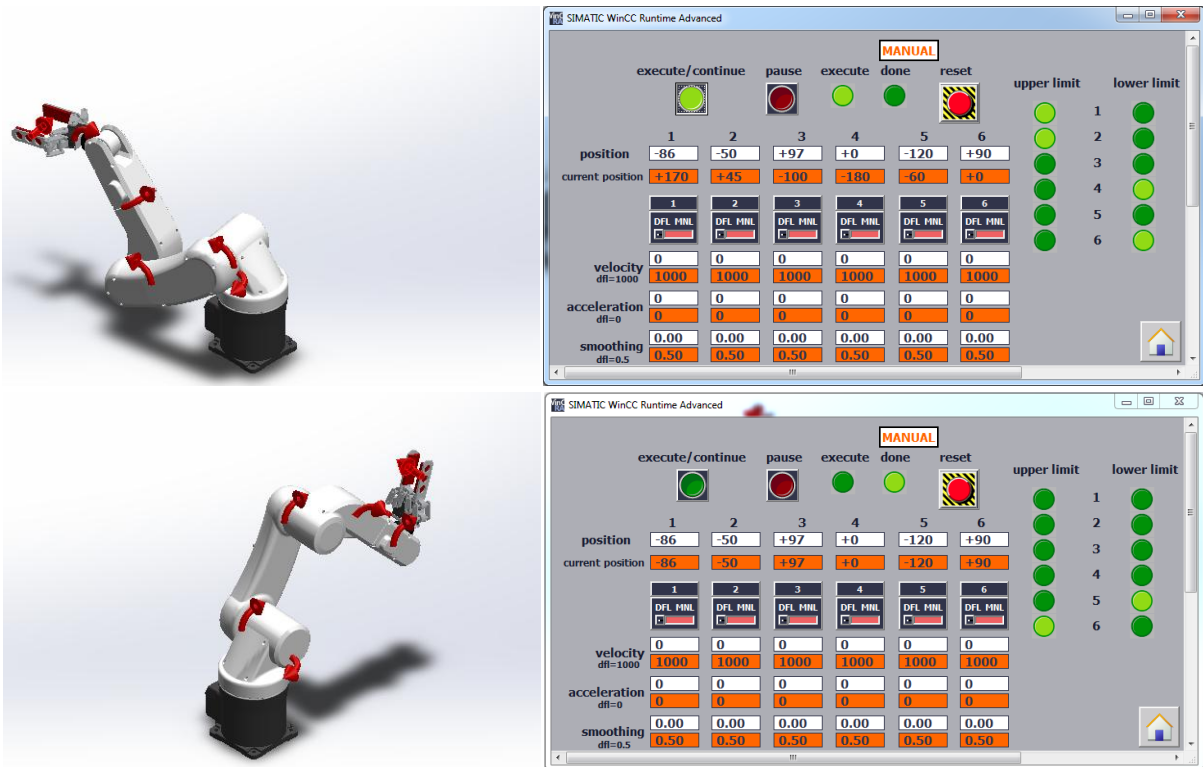


Σχήμα 5.27: Εκτέλεση ρουτίνας εργασίας 2 (Συνέχεια).

Στη συνέχεια ακολουθούν δυο παραδείγματα χειροκίνητου ελέγχου από την χειροκίνητη λειτουργία (Σχήματα 5.28 και 5.29). Από την οθόνη της χειροκίνητης λειτουργίας ο χρήστης εισάγει στα άσπρα πεδία τις επιθυμητές γωνιακές θέσεις των αρθρώσεων σε μοίρες και την επιθυμητή θέση της αρπάγης σε εκατοστά. Έπειτα πατώντας execute/continue ο εικονικός ρομποτικός βραχίονας εκτελεί την συγκεκριμένη κίνηση.



Σχήμα 5.28: Παράδειγμα χειροκίνητου ελέγχου 1.



Σχήμα 5.29: Παράδειγμα χειροκίνητου ελέγχου 2.

5.4 Συμπεράσματα

Στην παρούσα πτυχιακή εργασία παρουσιάστηκε η εφαρμογή της διαδικασίας VC στον βιομηχανικό ρομποτικό βραχίονα KUKA KR 5 sixx R850. Ο εικονικός έλεγχος και λειτουργία του βιομηχανικού ρομποτικού βραχίονα υλοποιείται μέσω ενός προσαρμοσμένου, μη ενιαίου περιβάλλοντος αξιοποιώντας τα λογισμικά LabVIEW, SolidWorks και της σουίτας TIA Portal.

Η αξιοποίηση της διαδικασίας VC κατά την ανάπτυξη ενός υπό μελέτη συστήματος επιτρέπει τον άμεσο εντοπισμό και διόρθωση των όποιων σφαλμάτων μπορεί να προκύψουν. Επίσης επιτρέπει την ανάπτυξη και επικύρωση του κώδικα που εκτελεί ο εκάστοτε ελεγκτής. Επομένως, επιτυγχάνεται η μείωση της χρονική διάρκεια της διαδικασίας του commissioning, άρα και η μείωση του χρόνου ανάπτυξης του συστήματος. Επίσης επιτυγχάνεται η μείωση του κόστους ανάπτυξης του συστήματος.

Το συγκεκριμένο Project εστιάζει στα βασικά τμήματα της διαδικασίας VC. Μελλοντικά θα μπορούσε να επεκταθεί με την προσθήκη της λειτουργίας ανίχνευσης σύγκρουσης (collision detection) στο τρισδιάστατο μοντέλο και την εισαγωγή κινηματικών εξισώσεων. Η λειτουργία ανίχνευσης σύγκρουσης εντοπίζει συγκρούσεις κατά την κίνηση ή περιστροφή μεταξύ των μερών που αποτελούν το τρισδιάστατο μοντέλο. Οι κινηματικές εξισώσεις επιτρέπουν στον εικονικό ρομποτικό βραχίονα την εκτέλεση εργασιών που απαιτούν λεπτομέρεια. Επίσης, θα μπορούσαν να προστεθούν και άλλα συστήματα συνθέτοντας έτσι μια ολοκληρωμένη γραμμή παραγωγής.

Τέλος, από την υλοποίηση αυτού του Project αποκτήθηκαν θεωρητικές γνώσεις όσον αφορά την ρομποτική, τους βιομηχανικούς αυτοματισμούς και τα βιομηχανικά συστήματα παραγωγής. Καθώς επίσης και εμπειρία σχετικά με τα λογισμικά LabVIEW, SolidWorks και την σουίτα TIA Portal.

Βιβλιογραφία

- [1] S. Makris, G. Michalos, and G. Chryssolouris, “Virtual Commissioning of an Assembly Cell with Cooperating Robots”, *Advances in Decision Sciences*, vol. 2012, Article ID 428060, pp. 1-11, 2012, doi: 10.1155/2012/428060.
- [2] G. Reinhart and G. Wunsch, “Economic application of virtual commissioning to mechatronic production systems”, *Production Engineering*, vol. 1, no. 4, pp. 371-379, December 2007, doi: 10.1007/s11740-007-0066-0.
- [3] Zheng Liu, Nico Suchold and Christian Diedrich, “Virtual Commissioning of Automated Systems”, *Automation*, Chapter 7, 2012, doi: 10.5772/45730.
- [4] ΕΙΣΑΓΩΓΙΚΕΣ ΣΗΜΕΙΩΣΕΙΣ ΡΟΜΠΟΤΙΚΗΣ, http://courseware.mech.ntua.gr/ml23419/robotics_pdf/intro.pdf, τελευταία επίσκεψη 4/7/15.
- [5] Ζήκος Νικόλαος, Διπλωματική Εργασία, “Εικονικός Ρομποτικός Βραχίονας”, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης, 2005.
- [6] Καραστεργίου Βασιλική και Τσιλομήτρου Ουρανία, Διπλωματική Εργασία, “Ανάπτυξη εφαρμογών σε όλα τα δυνατά περιβάλλοντα προγραμματισμού του ρομπότ Katana-Neuronics”, Πανεπιστήμιο Πάτρας, 2010.
- [7] KUKA KR 5 sixx R850, http://www.kuka-robotics.com/res/sps/e6c77545-9030-49b1-93f5-4d17c92173aa_Spez_KR_5_sixx_CR_en.pdf, τελευταία επίσκεψη 21/1/15.
- [8] SolidWorks Web Help, <http://help.solidworks.com/>, τελευταία επίσκεψη 16/7/15.
- [9] KUKA KR-5 R850 by Sugi Tjiu, <https://grabcad.com/library/kuka-kr-5-r850>, τελευταία επίσκεψη 13/1/15.
- [10] 4-Bar Linkage Gripper by Amir H. Soltanzadeh, <https://grabcad.com/library/4-bar-linkage-gripper-with-dynamixel-rx-64-1>, τελευταία επίσκεψη 28/1/15.
- [11] Hans Berger, *Automating with SIMATIC S7-300 inside TIA Portal*, 2nd Edition, Publicis, 2014.
- [12] Hans Berger, *Automating with SIMATIC*, 5th Edition, Publicis, 2013.
- [13] Hans Berger, *Automating with STEP 7 in STL and SCL*, 4th Edition, Publicis, 2007.
- [14] ΝΤΟΝΤΟΣ ΚΩΝΣΤΑΝΤΙΝΟΣ, Πτυχιακή Εργασία, “Ανάπτυξη Γραφικού Περιβάλλοντος Βασισμένο στο LabVIEW για την Προσομοίωση Ελεγχόμενων Ανορθωτικών Γεφυρών”, Τ.Ε.Ι. Δυτικής Ελλάδας, 2014.
- [15] Introduction to NI SoftMotion for SolidWorks, <http://www.ni.com/tutorial/9566/en/>, τελευταία επίσκεψη 11/10/15.
- [16] Getting Started with the NI SoftMotion Development Module for LabVIEW, <http://www.ni.com/pdf/manuals/371331d.pdf>, τελευταία επίσκεψη 11/10/15.
- [17] Introduction to OPC, <http://www.ni.com/white-paper/7451/en/>, τελευταία επίσκεψη 9/10/15.
- [18] Connect LabVIEW to Any PLC Using OPC, <http://www.ni.com/tutorial/7450/en/>, τελευταία επίσκεψη 9/10/15.

[19] NetToPLCSim A Network Interface to PLCSim, <http://nettoplcsim.sourceforge.net/index.html>, τελευταία επίσκεψη 2/11/15.