



TecnoCampus
Escola Superior
Politécnica

Centre adscrit a la



Universitat
Pompeu Fabra
Barcelona

Enginyeria electrònica i automàtica

Automatització i control de cèl·lula de fabricació flexible

Memòria

Joel Madueño Carmona
PONENT: Joan Triadó Aymerich

TARDOR/PRIMAVERA 2022/2023



TecnoCampus
Mataró-Maresme

Resum

El projecte tracte d'actualitzar el software de la cèl·lula de fabricació flexible situada al laboratori 4 de control. Es verifica la lògica del procés amb la biblioteca stateflow de Matlab. Una vegada verificat, el PLC Rockwell es programat en el llenguatge ST provinent del C, que proporciona una simplificació i millor estructura del software. El robot ABB IRB120 es programat amb el software RAPID i el robot Fanuc LR Mate 200d 4S amb Karel que ens permet agrupar de les diferent estacions. Una vegada finalitzada la programació es realitzaran les proves d'operació OQ per verificar el seu correcte funcionament.

Resumen

El proyecto trata de actualizar el software de la célula de fabricación flexible situada en el laboratorio 4 de control. Se verifica la lógica del proceso con la biblioteca stateflow de Matlab. Una vez verificado, el PLC Rockwell es programado con el lenguaje ST procedente del C, que proporciona una mayor simplificación i estructura del software. El robot ABB IRB120 es programado con el software RAPID i el robot Fanuc LR Mate 200d 4S con Karel, que nos permite agrupar las diferentes estaciones. Una vez finalizada la programación se realizarán las pruebas de operación OQ para verificar su correcto funcionamiento.

Abstract

The project aims to update the software of the flexible manufacturing cell located in control laboratory 4. The process logic is verified with the Matlab stateflow library. Once verified, the Rockwell PLC is programmed with the ST language from C, which provides greater simplification and structure of the software. The ABB IRB120 robot is programmed with the RAPID software and the Fanuc LR Mate 200d 4S robot with Karel, which allows us to group the different stations. Once the programming is finished, the OQ operation tests will be carried out to verify its correct operation.

Índex.

Índex de figures	V
Índex de Taules	IX
Glossari de termes.	X
Capítol I. Objectius.....	1
1.1.Objecte del projecte.	1
1.2.Revisió d’antecedents i necessitats d’informació	2
1.3.Abast del projecte.	11
1.4.Objectius i especificacions tècniques.	13
Capítol II. Viabilitat del Projecte	14
2.1.Alternatives de solució i viabilitat tècnica.....	14
2.1.1.Estat de l’art	17
2.1.1.1. Grafcet i estructura lògica fonamental	17
2.1.1.2. Maquina d’estats de Matlab	21
2.1.1.3. Rs-Logix 5000 i Programació ST.....	24
2.1.1.4. Robotstudio	26
2.1.1.5. Roboguide	30
2.2.Viabilitat Mediambiental	33
2.3.Viabilitat econòmica i pressupost.....	34
Capítol III. Desenvolupament de la solució.....	36
3.1 Disseny i construcció del GRAFCETS	37
3.2 Modelització i simulació de maquina d’estats.....	37
3.2.1 Programació de Maquina d’estats.	37
3.2.2 Simulació de Maquina d’estats	51
3.2.3. Programació d’autòmat Rockwell	56
3.2.3.1. Configuració de hardware i comunicació.....	56
3.2.3.2. Declaració de E/S.....	60
3.2.3.3. Programació de PLC en ST.....	63
3.2.4 Programació robot ABB IRB 120.....	80
3.3.5. Programació robot Fanuc LR MATE 200id 4S	85
Capítol IV. Planificació del projecte	95
4.1.Planificació.	95

Capítol V. Justificació de producte i Riscos	99
5.1.Pla de contingència	99
5.1.1Anàlisi de riscos	99
5.1.2.Pla d'actuació	102
5.2.Perspectiva de gènere	105
Capítol VI. Conclusions	106
Capítol VII. Referencias.....	108

Índex de figures

Índex.	III
Il·lustració 1. Primera estació, subministradora de peces font: (Gutiérrez, 2021).....	3
Il·lustració 2. Segona estació, detecció tipus de peça font: (Gutiérrez, 2021).....	4
Il·lustració 3. Tercera estació, palet transportador de peces font: (Gutiérrez, 2021).....	4
Il·lustració 4. Quarta estació, Magatzem de peces font: (Gutiérrez, 2021)	5
Il·lustració 5. Robot ABB IRB 120 font: (Gutiérrez, 2021)	5
Il·lustració 6. Robot FANUC LR Mate 200Id/4S (font: Pròpia)	7
Il·lustració 7. Primera subestació, arribada i sortida de peces font: (Peñalver, Primavera 2021)	9
Il·lustració 8. Segona subestació, Peces rebutjades font: (Peñalver, Primavera 2021).....	9
Il·lustració 9. Tercera subestació, Foradat de peça font: (Peñalver, Primavera 2021)	10
Il·lustració 10. Quarta subestació, verificació de foradat font: (Peñalver, Primavera 2021)....	11
Il·lustració 11. Cel·lula de fabricació flexible amb 2 robots i estació de foradat font: (Peñalver, Primavera 2021)	11
Il·lustració 12. Panelview Plus Standard 9W font: (Peñalver, Primavera 2021)	12
Il·lustració 13. Estructura de un Grafcet (font: Pròpia)	18
Il·lustració 14. Estructura de Convergència Y (font: Pròpia)	19
Il·lustració 15. Estructura de Convergència OR (font: Pròpia).....	19
Il·lustració 16. Estructura de Divergència Y (font: Pròpia)	20
Il·lustració 17. Estructura de Divergència OR (font: Pròpia)	20
Il·lustració 18. Estructura de salts (font: Pròpia)	21
Il·lustració 19.. Estructura gràfica del CHART (font: Pròpia).....	22
Il·lustració 20. Editor Gràfic amb funcions de programa (font: Pròpia).....	22
Il·lustració 21. Declaració de variables (font: Pròpia)	23
Il·lustració 22. Programa general (font: Pròpia)	23
Il·lustració 23. Programa principal (font: Pròpia).....	25
Il·lustració 24. Funció JSR (font: Pròpia)	25
Il·lustració 25.. Estructura ST(font: Pròpia).....	25
Il·lustració 26. Interfície gràfica del Robot ABB (font: Pròpia)	26
Il·lustració 27. Panel de control de Robot Studio (font: Pròpia).....	27
Il·lustració 28. Biblioteca de robots ABB (font: Pròpia)	27
Il·lustració 29. Simulació de Robot Studio (font: Pròpia)	28

Il·lustració 30. Configuració de simulació (font: Pròpia)	28
Il·lustració 31. Pestanya de Ràpid Robot Studio (font: Pròpia)	29
Il·lustració 32. Declaració de variables de programa(font: Pròpia)	29
Il·lustració 33. Programa principal (font: Pròpia)	30
Il·lustració 34. Interfície gràfica del robot FANUC (font: Pròpia)	31
Il·lustració 35.Simulació Gràfica de Flex pental Roboguide (font: Pròpia)	32
Il·lustració 36..Bloc de reposició maquina d'estats (font: Pròpia)	38
Il·lustració 37.Programació d'estats del bloc reposició (font: Pròpia)	39
Il·lustració 38. Bloc estació1 subministrament maquina d'estats (font: Pròpia)	40
Il·lustració 39.Programació d'estats del bloc d'estació1 subministrament (font: Pròpia)	41
Il·lustració 40..Bloc estació2 detecció tipus de peça maquina d'estats (font: Pròpia)	42
Il·lustració 41.Programació d'estats del bloc d'estació2 detecció tipus de peça (font: Pròpia)	43
Il·lustració 42.Bloc estació3 de palet a foradat maquina d'estats (font: Pròpia)	44
Il·lustració 43.Programació d'estats del bloc d'estació3 de palet fins a foradat (font: Pròpia)	45
Il·lustració 44.Bloc estació3 foradat maquina d'estats (font: Pròpia)	46
Il·lustració 45.Programació d'estats del bloc d'estació3 foradat part1 (font: Pròpia)	47
Il·lustració 46.Programació d'estats del bloc d'estació3 foradat part2 (font: Pròpia)	47
Il·lustració 47.1.Bloc estació3 de plat a palet maquina d'estats (font: Pròpia)	48
Il·lustració 46.2.Programació d'estats de blocs estació magatzem maquina d'estats (font: Pròpia)	49
Il·lustració 48.Bloc estació4 emmagatzema peça maquina d'estats (font: Pròpia)	50
Il·lustració 49.Programació d'estats del bloc d'estació4 emmagatzema peça (font: Pròpia)....	51
Il·lustració 49.Scada de simulació gràfic (font: Pròpia)	52
Il·lustració 50.Panel de control de simulació (font: Pròpia)	53
Il·lustració 51.Simulació dels actuadors gràfics (font: Pròpia)	54
Il·lustració 52.Simulació dels sensors (font: Pròpia)	55
Il·lustració 53. Configuració tarja de comunicació ethernet (font: Pròpia)	56
Il·lustració 54.Configuració del mòdul d'entrades1(font: Pròpia)	57
Il·lustració 55.Configuració del mòdul d'entrades2 (font: Pròpia)	57
Il·lustració 56.Configuració del mòdul d'entrades3 (font: Pròpia)	58
Il·lustració 57.Configuració del mòdul de sortides1(font: Pròpia)	58
Il·lustració 58.Configuració del mòdul de sortides2 (font: Pròpia)	59
Il·lustració 59.Configuració del mòdul de sortides3 (font: Pròpia)	59

Il·lustració 60.Estació de subministrament part1 (font: Pròpia)	63
Il·lustració 61.Estació de subministrament part2 (font: Pròpia)	64
Il·lustració 62.Moviment del robot ABB de estació de subministrament a estació de detecció tipus de peça (font: Pròpia)	64
Il·lustració 63. Estació de subministrament (font :Pròpia)	65
Il·lustració 64.Moviment del robot ABB de estació de detecció tipus de peça a palet.....	65
Il·lustració 65..Moviment del robot ABB de estació de detecció tipus de peça a contenidor .	65
Il·lustració 66.Moviment del robot ABB de estació de detecció tipus de peça a palet segona peça (font: Pròpia).....	66
Il·lustració 67.Moviment del robot ABB de estació de detecció tipus de peça a palet segona peça part2 (font: Pròpia).....	67
Il·lustració 68.Moviment del robot ABB de estació de detecció tipus de peça a cubeta peces descartades(font: Pròpia).....	68
Il·lustració 69.Estació moviment de palet cinta fins a estació foradat (font: Pròpia)	68
Il·lustració 70.Moviment del robot FANUC de palet a estació de foradat (font: Pròpia.....	69
Il·lustració 71.Estació de foradat gira taula (font: Pròpia).....	70
Il·lustració 72.Estació de foradat subjecta peça i forada (font: Pròpia).....	71
Il·lustració 73.Estació de foradat gira taula fins zona de verificació (font: Pròpia)	72
Il·lustració 74.Estació de foradat, verifica el forat (font: Pròpia)	73
Il·lustració 75.Moviment de robot Fanuc de estació de foradat a palet correcte foradat.....	73
Il·lustració 76.Moviment del robot FANUC de palet a estació de foradat segona peça (font: Pròpia)	74
Il·lustració 77.Estació de foradat gira taula segona peça (font :Pròpia)	75
Il·lustració 78.Estació de foradat subjecta peça i forada segona peça (font: Pròpia).....	76
Il·lustració 79.Estació de foradat gira taula fins zona de verificació segona peça(font: Pròpia)	76
Il·lustració 80.Estació de foradat, verifica el forat segona peça (font: Pròpia).....	77
Il·lustració 81.Moviment del robot FANUC de estació de foradat a palet segona peça y es mou palet fins a el destí desitjat(font: Pròpia).....	78
Il·lustració 82.Moviment de robot ABB de palet a estació de magatzem (font: Pròpia).....	79
Il·lustració 83.Moviment de robot ABB de palet a estació de magatzem segona peça(font: Pròpia)	79
Il·lustració 84.Funció principal de programa Robot Studio (font: Pròpia)	80

Il·lustració 85.Funció posició inici de programa Robot Studio (font: Pròpia).....	81
Il·lustració 86.Funció de subministrament Robot Studio (font: Pròpia).....	81
Il·lustració 87.Funció de detecció peces en palet Robot Studio (font: Pròpia).....	82
Il·lustració 88.Funció de subministrament segona peça Robot Studio (font: Pròpia)	82
Il·lustració 810.Funció de magatzem Robot Studio (font: Pròpia)	83
Il·lustració 90.Funció de contenidor Robot Studio (font: Pròpia)	84
Il·lustració 91.Funció principal de robot FANUC roboguide (font: Pròpia)	85
Il·lustració 92.Funció preparat de robot FANUC roboguide (font: Pròpia)	86
Il·lustració 93.Funció de palet a plat de robot FANUC roboguide (font: Pròpia)	87
Il·lustració 94.Funció palet a plat part2 de robot FANUC roboguide (font: Pròpia).....	88
Il·lustració 95.Funció palet a plat part3 de robot FANUC roboguide (font: Pròpia).....	89
Il·lustració 96.Funció palet a plat part4 de robot FANUC roboguide (font: Pròpia).....	90
Il·lustració 97.Funció plat a palet part1 de robot FANUC roboguide (font: Pròpia).....	91
Il·lustració 98.Funció plat a palet part2 de robot FANUC roboguide (font: Pròpia).....	92
Il·lustració 99.Funció plat a palet part3 de robot FANUC roboguide (font: Pròpia).....	93
Il·lustració 100.Funció contenidor de robot FANUC roboguide (font: Pròpia)	94
Il·lustració 101. Diagrama de gannt del projecte	95

Índex de Taules

Taula 1.Preu dels recursos.....	34
Taula 2.Capitol1 de pressupost, Realització del projecte.....	34
Taula 3.Capitol3 de pressupost, Amortitzacions.....	35
Taula 4..Pressupost total.....	35
Taula 5.Asignació d'entrades digitals de la tarja 1	60
Taula 6.Asignació d'entrades digitals de la tarja 2	60
Taula 7.Asignació d'entrades digitals de la tarja 3	61
Taula 8..Asignació de sortides digitals de la tarja 1	61
Taula 9.Asignació de sortides digitals de la tarja 2.....	62
Taula 10.Asignació de sortides digitals de la tarja 3.....	62
Taula 11.Correspondència entre Entrades del PLC i Sortides FANUC.....	85
Taula 12.Correspondència entre Sortides del PLC i Entradess FANU.....	85
Taula 13.Planificació del projecte	95
Taula 14.Freqüència de punts de control	97
Taula 15.Avaluació gravetat de la incidència	100
Taula 16.Avaluació freqüència de la incidència	100
Taula 17.Facilitat de detecció de la incidència	100
Taula 18.Identificació de riscos i avaluació	101

Glossari de termes.

Matlab	Software de programació i càlcul numèric
Rockwell	Autòmat programable
ST	Llenguatge de programació d'autòmats provinent del C
Ladder	Llenguatge de programació d'autòmats amb blocs i contactes
Rapid	Software de programació de robots ABB
Karel	Software de programació de robots Fanuc
FMS	Sistema de fabricació flexible
Just-in-Time	Filosofia que treballa per la reducció de stock al mínim
IQ	Qualificació de la instal·lació
OQ	Qualificació de l'operació
Datasheet	Fitxa tècnica
PLC	Autòmat programable
Stateflow	Llibreria de matlab per la creació de la maquina d'estats
Payload	Pes que suporta el braç robòtic
Armload	Pes que pot agafar o desplaçar el braç robòtic
HMI	Human-machine Interface
Grafcet	Diagrama funcional d'evolució del procés
E/S	Entrades i sortides
Dill	Dilluns
Dim	Dimarts
Dimec	Dimecres
Dij	Dijous
Div	Divendres
Master /Slave	Model de comunicació o control asimètric
Grafcet	Diagrama de Flux

CHART	Eina de modelat i simulació
JSR	Funció que entra a una subrutina.
Flex Pental	Comandament de Robots
Roboguide	Software per robot FANUC
FANUC	Fabricant de Robots
Backup	Copia de seguretat
Free	Lliure
CC	Quadre de comandament

Capítol I. Objectius

1.1.Objecte del projecte.

L'objecte del projecte és el control i automatització de una cèl·lula de fabricació flexible, enfocada en millorar el rendiment i la facilitació del futur manteniment.

La cèl·lula de fabricació flexible consta d'una versatilitat per tal de poder modificar-se en base al fabricació desitjada.

La verificació del procés de fabricació es realitza mitjançant matlab, s'utilitza una maquina d'estats que controla el temps i la viabilitat del procés. Amb la modelització de matlab , es pot ajustar el temps en base al coll d'ampolla de la línia.

El control es realitza amb un autòmat programable , el qual controla els sensors i actuadors empleats, a mes de les entrades i sortides dels robots utilitzades. L'autòmat es programat en el llenguatge ST, això facilita el seguiment per tal de futures modificacions o manteniments, degut a que disminueix les línies de codi en comparació amb la programació Ladder.

Es programaran 2 robots que s'utilitzaran per interconnectar les diferents estacions de treball, un es ABB que es programa amb el software Rapid i l'altre Fanuc que es programa amb el llenguatge Karel.

Finalment una vegada acabat el projecte i la seva programació ,es procedirà a realitzar les probes OQ, que verifica el compliment de les correctes operacions.

1.2.Revisió d'antecedents i necessitats d'informació

La indústria es un sector que esta tenint un creixement molt elevat , degut a la competència en el termini de la fabricació i la quantitat de productes a fabricar .Això porta a haver de buscar altres alternatives, com és l'automatització per tal de poder accelerar les tasques dels processos.

El mon de l'automatització ve donat per una gran diversitat de processos a realitzar, amb la capacitat d'utilitzar diferents eines per controlar-ho ,com podria ser diferents sensors, la utilització de robots o de noves tecnologies, com la visió artificial i les xarxes neuronals .

Una de les eines interessant a utilitzar son cèl·lules de fabricació flexible , poden servir des de processos simples i repetitius , fins a processos mes precisos i llargs en base als sensors i actuadors dels que esta format.

Les cèl·lules de fabricació flexibles son un grup d'estacions de treball interconnectades per medi d'un sistema de transport automatitzat, controlades per un autòmat programable en el que van connectat els sensors i actuadors. Això porta a poder optimitzar els lots de fabricació, es treballa amb una filosofia de Just-in-Time ja que nomes es fabrica lo necessari, de tal forma que elimina tot tipus de costos de stock. Per altre banda te un gran cost inicial en el software , a mes necessita una reestructuració completa de la producció per les diferents peces que s'utilitzen. (Vasco)

La cèl·lula de fabricació flexible en qüestió, amb la qual treballarem ha passat per un seguit d'evolucions al llarg del temps.

Fent una recerca de quines son les persones que han anat passant per la cèl·lula i les millores que han aplicat sobre aquesta.

La cèl·lula constava de 4 estacions que s'han de comunicar entre elles seqüencialment, per tal de poder fer la fabricació desitjada.

La seqüència que ha de complir cada estació es la següent:

Primera estació

La primera estació es la sortida de les peces, on hi ha un subministrador de peces que deixa caure les peces fins a un carril, on es arrastra per un cilindre pneumàtic fins arribar al final de cursa que detecta que la peça esta al lloc desitjat un cop en lloc la peça haurà d'anar cap a la segona estació



Il·lustració 1. Primera estació, subministradora de peces font: (Gutiérrez, 2021)

Segona estació

La segona estació consta d'un acoblament per les peces amb uns sensor òptics per tal de diferenciar les peces vermelles i metàl·liques. Un cop detecta que la peça es correcte segons la demanda de producció la peça ha d'anar cap a la tercera estació.



Il·lustració 2. Segona estació, detecció tipus de peça font: (Gutiérrez, 2021)

Tercera estació

En la tercera estació trobem un palet per poder desplaçar les peces , esta situat en els carrils de la cinta transportadora, consta d'un pistó que no deixa passar les peces , que actua mitjançant una electrovàlvula que s'activa quan es vol deixar passar el palet, per detectar la posició del palet s'utilitza un sensor inductiu . Una vegada arriba la peça la cinta s'activa i la electrovàlvula fa baixar el pistó. El palet es va movent per tota la cinta fins arribar a la quarta estació.



Il·lustració 3. Tercera estació, palet transportador de peces font: (Gutiérrez, 2021)

Quarta estació

El palet detecta que arriba a l'última estació quan arriba al sensor inductiu i no pot avançar més per el pistó. Trobem un magatzem de peces que consta de 25 espais per emmagatzemar les peces per finalitzar la nostra comanda.



Il·lustració 4. Quarta estació, Magatzem de peces font: (Gutiérrez, 2021)

Principalment va començar per (Graells, Primavera 2011), el qual va fer la implementació del robot ABB IRB 120 sobre la cèl·lula de fabricació, abans d'aquesta implementació la cèl·lula de fabricació no podia estar totalment automatitzada ja que necessitava a una persona per tal de transportar la peça des de una estació fins a un altre.



Il·lustració 5. Robot ABB IRB 120 font: (Gutiérrez, 2021)

El robot té un pes de 25 kg, consta d'un payload de 3-4kg en el qual es te en compte tot el pes que subjecta el robot contant els cables pneumàtic, l'eina situada a l'eix de la ma etc...

Consta d'un armload de 0,3kg, que això ja ens es mes que suficient degut a que no supera el pes de les peces a moure.

Els estudis que es van haver de fer per verificar prèviament la implementació del robot a la cèl·lula van ser els següents:

En primer lloc la utilització del software Robotstudio, el qual consta d'una programació del robot offline que ens permet poder treballar amb el robot sense la necessitat d'atura la línia de fabricació. Amb la utilització d'aquest software aconseguim una reducció de riscos a l'hora de la posada en marxa del robot.

Es va buscar l'optimització de trajectòries per tal de que el robot no fes moviments estranys que poguessin posar en risc el seu funcionament, de tal forma que ens permet que el robot faci moviments mes lineals de forma senzilla i eficient.

Uns dels punt mes importants que es van tenir en compte ja que era necessari va ser l'estudi de detecció de col·lisió, per tal de verificar si les trajectòries eren adients.

Es van implementar les llibreries de CAD i la creació de les peces amb sòlidworks, per tal de poder simular tot el procés de forma virtual ja que robotstudio no constava de tots aquests elements.

De tal forma que amb la utilització de tots aquests recursos , es pot recrear tot el sistema automatitzat.

En definitiva, amb la implementació del robot ABB es va poder automatitzar la línia de fabricació sense la necessitat de cap persona per moure les peces d'estació, de tal forma que la línia es autònoma.

Seguidament una vegada implementat el robot ABB a la línia .

Segons recerca realitzada la nova aplicació que es realitza la vaig trobar en (Alvarado, Primavera 2020) . Que va decidir ampliar la cèl·lula de fabricació amb una nova estació, per tal de què fos l'encarregada de realitzar el foradament de les peces.

Per poder resoldre aquesta nova aplicació es va haver d'implementar un nou robot que en aquest cas es va decidir per un Fanuc de la marca LR Mate 200Id/4S.

(Alvarado, Primavera 2020), no va ser la persona que va implementar el robot Fanuc ni la ampliació de la nova estació, però sí es el treball de la recerca més proper en el qual ja estava generada aquesta modificació



Il·lustració 6. Robot FANUC LR Mate 200Id/4S (font: Pròpia)

El robot FANUC té un pes de 25kg, consta d'un Payload de 4kg amb 6 eixos lliures podent arribar a distàncies de 550mm.

La programació integrada d'aquest robot es va realitzar amb el touchpad del robot programat amb llenguatge d'alt nivell . Això ens permet simular el moviment del robot per poder controlar les trajectòries lineals i simples, evitant així possibles col·lisions amb elements.

La funcionalitat d'aquest robot en la línia de fabricació consta en agafar les peces que es van desplaçant per el palet, creant un punt d'aturada amb un pistó que consta de un sensor inductiu

per detectar el palet de tal forma que quan el sensor detecta que ha arribat el palet i s'atura la cinta, seguidament

el robot es dirigeix a realitzar la trajectòria programada fins a arribar a l'estació de foradat.

Conjuntament amb la implementació del Robot FANUC s'implementa una altre estació, que esta dividida en diferents subestacions, Foradat, Verificació ,arribada i sortida de peces i la de rebuig quan es realitza malament el foradat.

El robot FANUC s'implementa per poder enviar les peces automàticament a la nova estació implementada .

El funcionament que ha de realitzar la nova estació i les subestacions es el següent:

Estació de foradat

La estació de foradat s'implementa entre l'estació 3 i l'estació 4.

Consta de una taula circular, amb 4 subestacions per col·locar la peça, amb un motor que realitza moviments de 90° per anar desplaçant la peça per les diferents subestacions.

Primera subestació

La primera subestació es la mes pròxima al robot, té la funció de arribada i sortida de peces, quan el robot deixa la peça la taula gira 180° per portar-la cap a la tercera subestació que es

l'encarregada de foradar, quan es detecta que la peça es foradada correctament torna un altre vegada a aquesta subestació.

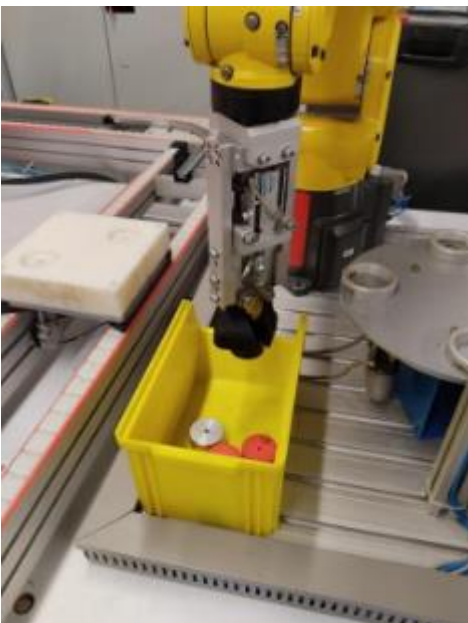


Il·lustració 7. Primera subestació, arribada i sortida de peces font: (Peñalver, Primavera 2021)

Segona subestació

La segona subestació està desplaçada 90° respecte la primera, és l'encarregada de les peces de rebuig, quan es realitza el procés de foradat i es detecta que la peça no ha sigut ben foradada en la subestació quatre, el motor gira 180° per portar-la a aquesta subestació.

Una vegada en aquesta subestació el robot tirarà la peça a la caixa de rebuig.



Il·lustració 8. Segona subestació, Peces rebutjades font: (Peñalver, Primavera 2021)

Tercera subestació

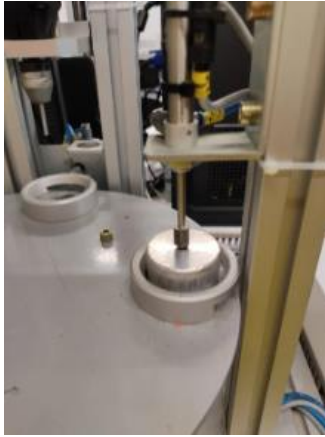
La tercera subestació esta desplaçada 180° respecte la primera, es l'encarregada de simular el foradat de la peça, mitjançant un motor el trepant baixa fins arribar al fons del forat, una vegada finalitzat torna a pujar fent girar el motor en sentit contrari, seguidament gira 90° per anar a la quarta subestació.



Il·lustració 9.Tercera subestació, Foradat de peça font: (Peñalver, Primavera 2021)

Quarta subestació

La quarta subestació esta desplaçada 270° respecte la primera, es l'encarregada de verificar si la peça ha sigut correctament foradada . Baixa un cilindre fins arribar al punt inferior del forat, si no detecta res perquè ha estat ben foradada tornarà a pujar i farà que la taula giri 90° per desplaçar la peça a la primera subestació, en altre cas si baixa el cilindre i no pot baixar fins el final del seu recorregut esperarà un temps i tornarà a pujar, farà girar la taula 180° per portar la peça mal foradada a la segona subestació.



Il·lustració 10. Quarta subestació, verificació de foradat font: (Peñalver, Primavera 2021)

Una vegada finalitzada l'estació de foradat, el robot FANUC agafarà les peces ben foradades i les col·locarà al palet, seguidament la cita s'activarà fent que es mogui el palet fins l'estació quatre, que es l'última per emmagatzemar les peces.



Il·lustració 11. Cel·lula de fabricació flexible amb 2 robots i estació de foradat font: (Peñalver, Primavera 2021)

Finalment l'última modificació realitzada a la cèl·lula segons la recerca realitzada la va fer (Peñalver, Primavera 2021), que va realitzar la implementació d'un Hmi per tenir el control dels actuadors i sensors de la cèl·lula en temps real i poder controlar-la de forma remota.

La qual consta d'un seguit de pantalles per tal de poder tenir el control exhaustiu en tot moment amb el seu scada pertinent.

La interfície gràfica escollida va ser Panelview Plus 7 Standard 9W.



Il·lustració 12. Panelview Plus Standard 9W font: (Peñalver, Primavera 2021)

Aquests dispositiu es compatible amb els controladors ControlLogix i CompactLogix.

Les seves característiques principals són:

- Comunicació Ethernet amb un únic port
- Voltatge d'alimentació 24VDC
- Sistema operatiu Windows CE amb accés a escriptori
- Capacitat de 25 pantalles i 200 alarmes.
- Mida de 9'' amb una amplitud de 800x480
- Port USB i amb implementació de targeta SD
- Capacitat de 512MB

El sistema esta desenvolupat en la creació de diferents pantalles, on realitzen les funcions de control i supervisió del sistema.

Aquestes són les següents pantalles i els seus funcionaments:

Pantalla d'inici

Es la pantalla principal on es troba el logotip de la universitat i dona accés a les diferents pantalles.

Pantalla automàtic

Aquesta pantalla te la funció de poder controlar el procés automàtic desitjat de forma remota.

Pantalla manual

La pantalla manual te la capacitat de poder activar tots els actuadors de forma manual sense seguir un procés, es molt útil per la comprovació del bon funcionament dels actuadors.

Pantalla de configuració

Aquí troben on es realitza la parametrització de la fabricació , segons els tipus i quantitat de peces que volem fabricar.

Pantalla d'alarmes

En la pantalla d'alarmes es realitza un històric d'alarmes , que ens serveix per controlar els avisos o averies que han succeït al llarg del temps

Nivell d'usuari

Aquesta pantalla protegida per contrasenya, serveix per el control d'accés segons el tipus de personal en cada pantalla.

Tot aquest seguit se pantalles ens permet tenir un seguiment i control exhaustiu en tot moment de la cèl·lula de fabricació

1.3.Abast del projecte.

En aquest projecte es realitza la programació de l'autòmat en llenguatge ST per tal de que la cèl·lula de fabricació flexible pugui fer el seu procés desitjat, però no es faran modificacions del software una vegada realitzada la OQ, això es considera com costos apart. Si en algun moment es realitza alguna modificació del software per altre personal no ens fem càrrec de futurs problemes o averies.

Es proporcionarà la programació dels robots de la cèl·lula de fabricació, amb els punts de posició establerts en base a d'instal·lació de la cèl·lula de fabricació flexible . En qualsevol cas si els robots fossin manipulats per altre personal o funcionant amb un mal us, no ens fem càrrec de futurs problemes o averies.

Nomes ens encarreguem del software necessari, no muntarem cap element mecànic ni elèctric de la instal·lació.

1.4.Objectius i especificacions tècniques.

Els objectius a complir del projecte son :

- Verificació prèvia del procés.

-Acceleració del procés

-Simplificació del software

-Sistema totalment automatitzat

Especificacions tècniques necessàries.

1. Simular amb la llibreria Stateflow de Matlab l'aplicació del sistema de cèl·lula flexible situada al laboratori 4 de control
2. Conversió de la simulació feta amb la llibreria Stateflow a un llenguatge C
3. Programar el PLC Rockwell en llenguatge ST que es provinent del C
4. Traslladar el codi C del punt 2 als robots a llenguatge RAPID (Per a l'IRB120 d'ABB)i a llenguatge KAREL(Per al Fanuc LR Mate 200d 4S)
5. Aconseguir els punts necessaris dels robots del punt 4 per complir la seqüència.

Capítol II. Viabilitat del Projecte

2.1. Alternatives de solució i viabilitat tècnica.

Modelització de la lògica i programació d'una cèl·lula de fabricació flexible.

Les cèl·lules de fabricació flexible son molt versàtils degut al seu sistema modular amb possibilitat de futures modificacions.

Treballem en base a un marc establert per les normatives UNE ISO per tal de fer complir la normativa en base als elements utilitzats en la cèl·lula de fabricació flexible.

UNE-EN ISO 11354-1:2012 : Tecnologies avançades d'automatització i les seves aplicacions .Requisits per establir la interoperabilitat entre processos de fabricació en les empreses. (Normalización)

UNE-EN IEC 61131-9:2022 : Autòmats programables. Interfície digital de comunicacions punt a punt per sensors i actuadors petits (Ratificada per la associació espanyola de normalització en setembre de 2022) (Normalización)

Amb els marcs legals establerts procedim a determinar la solució possible tècnicament.

Solució 1:

Es vol aconseguir el control i programació total de la cèl·lula de fabricació. Aquesta consta de la programació d'un autòmat Rockwell que fa la funció de master i dos robots, un ABB IRB 120 i un Fanuc LR Mate 200d 4S. Aquests dos robots tenen les seves respectives entrades i sortides interconnectades al PLC de tal forma que es pot saber l'estat dels robots i del PLC en tot moment.

Prèviament per la programació de l'autòmat s'ha de fer un estudi del funcionament desitjat que es vol aconseguir amb la cèl·lula , conjuntament amb els sensors i actuadors que es tenen per tal de poder controlar-ho.

Una vegada realitzat el estudi , es transfereix el tot el coneixement en diagrames de flux (grafcet), per tal de tenir una estructura i planificació de tot el funcionament de la cèl·lula.

Els grafkets consten de tres grans elements , etapes, accions i transicions . Es comença en una etapa inicial en la qual tots els actuadors i sensors estan en l'estat de repòs desitjat , quan

succeeix la transició pertinent passa a la següent etapa, en la qual es realitza les accions assignades i així fins finalitzar el bucle desitjat.

El Plc Rockwell es programarà amb el software Rs-Logix 5000 ,amb el llenguatge de programació Ladder , mitjançant una programació amb diagrames de contactes. Aquesta forma de programació té un gran avantatge ja que es molt gràfica però genera moltes línies de codi, de tal forma que per programes curts i seqüencials es un llenguatge òptim per utilitzar.

Seguidament per la programació del robot ABB IRB 1200, s'utilitza el software Robot Studio, en el qual es genera un espai virtual 3D amb el que podem simular el nostre braç robòtic.

Principalment per la programació del robot es necessari agafar els punts de treball manualment amb la touchpad del robot, ja que no tenim un escenari virtual amb les distàncies exactes de la cèl·lula.

A continuació una vegada obtingut els punts del robot s'extrauen del touchpad i s'implementen en el software per tal de crear un programa en Rapid. Per la programació de Rapid s'utilitzen condicionals o bucles per fer els moviments adients segons l'estat en el qual es troba la cèl·lula.

Finalment per la programació del robot Fanuc LR Mate 200d 4S, s'utilitza la touchpad per tal de poder obtenir els punts en els quals volem que treballi el nostre robot. Una vegada obtinguts els punts s'utilitza la mateixa touchpad per fer un programa en llenguatge C++ , per tal d'aconseguir la seqüència desitjada de la nostra cèl·lula.

Solució 2:

Es vol aconseguir el control i programació total de la cèl·lula de fabricació, utilitzant una forma de programació simplificada , per tal manera que en futures modificacions o averies es solucioni en el mínim temps possible.

Consta de la programació d'un autòmat Rockwell que fa la funció de master i dos robots, un ABB IRB 120 i un Fanuc LR Mate 200d 4S. Aquests dos robots tenen les seves respectives entrades i sortides interconnectades al PLC de tal forma que es pot saber l'estat dels robots i del PLC en tot moment.

Prèviament per la programació de l'autòmat s'ha de fer un estudi del funcionament desitjat que es vol aconseguir amb la cèl·lula , conjuntament amb els sensors i actuadors que es tenen per tal de poder controlar-ho.

Una vegada realitzat el estudi , es transfereix el tot el coneixement en diagrames de flux (grafcet), per tal de tenir una estructura i planificació de tot el funcionament de la cèl·lula.

Els grafkets consten de tres grans elements , etapes, accions i transicions . Es comença en una etapa inicial en la qual tots els actuadors i sensors estan en l'estat de repòs desitjat , quan succeeix la transició pertinent passa a la següent etapa, en la qual es realitza les accions assignades i així fins finalitzar el bucle desitjat.

Una vegada realitzat els grafkets , es verifica la lògica de programa amb Matlab , dintre de simulink la funció de simulació de Matlab es realitzarà un stateflow de tal forma que es pot verificar si la lògica de programa desitjada es possible realitzar-la tècnicament i verificar el seu funcionament. Gracies a la visualització gràfica que obtenim podem comprovar si es possible o no i en cas de que no fos possible fer les modificacions adients.

El Plc Rockwell es programarà amb el software Rs-Logix 5000 ,amb el llenguatge de programació ST ,es un llenguatge de programació d'alt nivell semblant al C.Conjuntament amb aquest llenguatge de programació es generaran subrutines, per tal de separar el programa principal en petits programes, de tal forma que s'optimitza la cerca de les averies.

Seguidament per la programació del robot ABB IRB 1200, s'utilitza el software Robot Studio, en el qual es genera un espai virtual 3D amb el que podem simular el nostre braç robòtic.

Principalment per la programació del robot es necessari agafar els punts de treball manualment amb la touchpad del robot, ja que no tenim un escenari virtual amb les distancies exactes de la cèl·lula.

A continuació una vegada obtingut els punts del robot s'extrauen del touchpad i s'implementen en el software per tal de crear un programa en Rapid. Per la programació de Rapid s'utilitzen condicionals o bucles per fer els moviments adients segons l'estat en el qual es troba la cèl·lula. Els diferents moviments que ha de fer el robot es dividiran en subrutines , per tal de centralitzar el problema en futures averies o modificacions del programa

Finalment per la programació del robot Fanuc LR Mate 200d 4S, s'utilitza la touchpad per tal de poder obtenir els punts en els quals volem que treballi el nostre robot. Una vegada obtinguts els punts s'utilitza la mateixa touchpad per fer un programa en llenguatge C++ , per tal d'aconseguir la seqüència desitjada de la nostra cèl·lula. Els diferents moviments que ha de fer el robot es dividiran en subprogrames , per tal de centralitzar el problema en futures averies o

modificacions del programa. Des de el programa principal es farà una trucada al subprograma desitjat per tal de que executi el moviment.

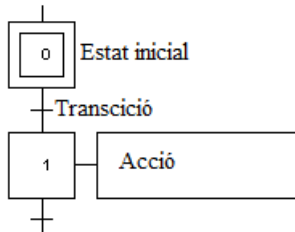
Fent una comparativa de les dues solucions, es molt mes adient la solució 2 degut a que tota la programació esta feta en llenguatge d'alt nivell , això ens proporciona una major estructura del programa i una simplificació del codi de programació. Conjuntament amb la utilització de subrutines i subprogrames per tal d'agrupar les diferents estacions o accions en altres programes, això ens ajuda a centralitzar la cerca de futures averies o modificacions de la cèl·lula.

2.1.1.Estat de l'art

2.1.1.1. Grafcet i estructura lògica fonamental

Per inicialitzar el nostre projecte principalment utilitzarem els Grafkets que es un model gràfic que s'utilitza per transferir el comportament d'un sistema d'automatitzat o de control.

El Grafcet consta de tres elements principals : Les etapes que venen marcades per quadrats amb una numeració seqüencial, les accions que son marcades per rectangles on especifica els actuadors que s'accionaran y les transicions que son les condicions que s'han de complir per que el sistema passi d'una etapa a la següent .



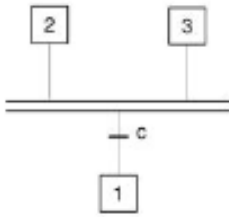
Il·lustració 13. Estructura de un Grafcet (font: Pròpia)

Com podem observar el estat inicial de la cèl·lula bé marcat per un doble quadrat que fa referencia a l'estat de repòs de la cèl·lula.

Per la realització del grafcet tenim diferents estructures lògiques fonamentals d'us com la convergència , divergència o el bus d'un sistema i es poden utilitzar segons la necessitat mes adient.

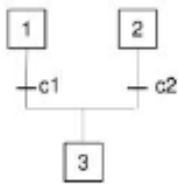
La convergència serveix per indicar que varies transicions condueixen a la mateixa etapa , fa referencia que des de diferents camins indiferentment es pot arribar al mateix destí.

Hi ha dos tipus de convergència la convergència Y i la OR , la diferencia que hi ha entre aquestes dues es que la convergència Y utilitza un bus de connexió, llavors es com fer un compliment lògic d'una porta AND es a dir que s'ha de complir que els camins possibles per activar la següent etapa compleixin amb la seva condició.



Il·lustració 14. Estructura de Convergència Y (font: Pròpia)

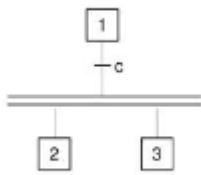
En canvi la convergència OR tal i com diu el seu nom fa un compliment de porta lògica OR es a dir indistintament des de qualsevol camí que activi la següent etapa si es compleix la condició de transició s'inicialitza



Il·lustració 15. Estructura de Convergència OR (font: Pròpia)

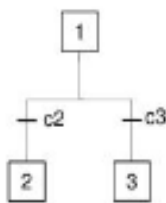
Per altre banda la divergència s'utilitza per indicar que des de una mateixa etapa es pot bifurcar en diferents etapes, s'utilitza normalment per fer dos processos indiferentment del seu compliment.

Hi ha dos tipus de divergència la divergència Y i la OR, la diferència que hi ha entre aquestes dues divergències es que la Y parteix des de una etapa i amb una transició comú inicialitza la bifurcació del sistema a l'hora , es un sistema molt interesant per realitzar dos processos a l'hora.



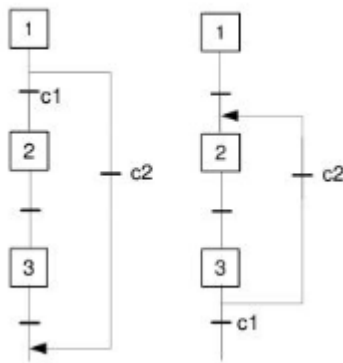
Il·lustració 16. Estructura de Divergència Y (font: Pròpia)

En canvi la divergència OR inicialitza diferents processos des de una etapa comú amb condicions diferents.



Il·lustració 17. Estructura de Divergència OR (font: Pròpia)

Finalment en els graficats tenim els salts , que s'utilitzen per realitzar bucles de programes, comptadors o per emetre alguna part del programa depenent de l'estat del sistema.



Il·lustració 18. Estructura de salts (font: Pròpia)

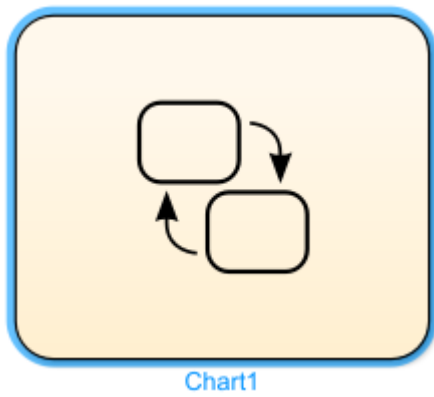
Tal i com podem observar en la il·lustració 18 hi ha 2 tipus de salts els salts cap a endavant y els salts cap a enrere, de normal els salt cap a endavant s'utilitzen per evitar passar per alguna etapa si el sistema ja compleix les condicions marcades. Mentre que els salts cap a enrere de normal es fan servir per realitzar bucles de programes o repeticions del procés tantes vegades com sigui necessari.

2.1.1.2. Maquina d'estats de Matlab

Per realitzar la maquina d'estats s'utilitza el software Matlab amb l'aplicació de l'eina stateflow, ja que es una eina de modelat i simulació que s'utilitza per el disseny de sistemes de control basats en estats, de tal forma que ofereix una notació gràfica basada en diagrames d'estat i transicions.

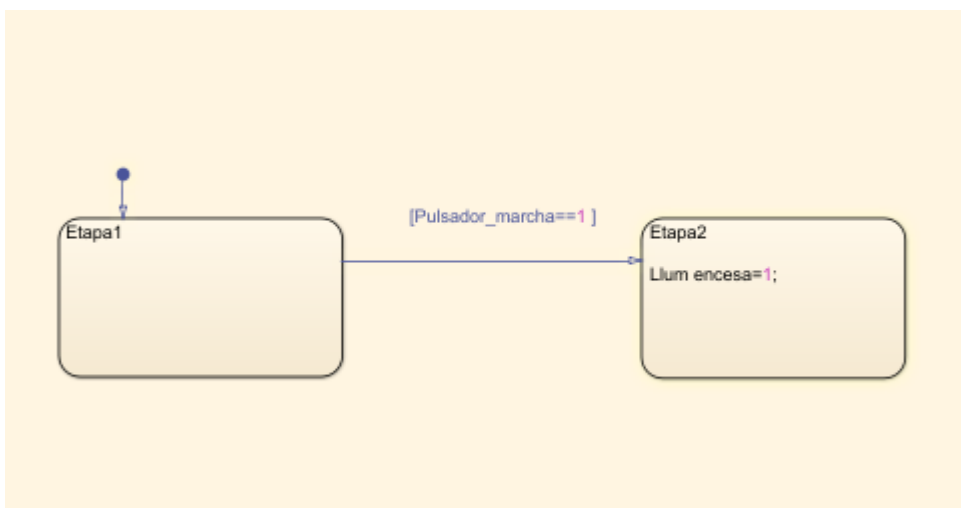
Entre de altres funcions el stateflow ens permet transferir els graficets prèviament generats i ho transforma directament a codi C, a mes ens serveix per modelar i simular prèviament el sistema de control que tenim sense necessitar la maqueta físicament.

Per optar a la generació de stateflow utilitzarem l'eina CHART , ja que al seu interior té una agrupació d'eines que necessitem per la nostra maquina d'estats.



Il·lustració 19.. Estructura gràfica del CHART (font: Pròpia)

- Conte un editor gràfic, que permet plasmar intuïtivament tota la estructura del nostre sistema.
- Nodes de tasques , que s'utilitzen per modelitzar les tasques habituals del sistema
- Estats: S'utilitza per marcar l'inici i final de una tasca
- Accions: Compliments i actuadors que se inicialitzen en cada estat
- Condicions: S'aplica per restringir el procés segons el compliment que es desitgi
- Generació automàtica de diferents tipus de llenguatge entre d'altres C/C++ codi , HDL codi y PLC codi



Il·lustració 20. Editor Gràfic amb funcions de programa (font: Pròpia)

Com es pot observar en la il·lustració 20 es pot veure una estructura gràfica, on els requadres fa la funció de estats, etapes , entremetjts dels dos requadres hi ha una fletxa que fa la funció de transició on la condició s'ha de expressar en llenguatge C. Finalment dintre del requadre de Etapa2 es pot veure la inicialització de una acció que es troba dintre de l'estat.

Una vegada realitzat el programa en dintre del CHART , es pot obtenir el programa automàticament en llenguatge de PLC expressat en ST,

```

22 FUNCTION_BLOCK prova
23 VAR_INPUT
24     ssMethodType: SINT;
25     Pulsador_marcha: LREAL;
26 END_VAR
27 VAR_OUTPUT
28     Llum_encesa: LREAL;
29 END_VAR
30 VAR
31     is_active_c8_prova: USINT;
32     is_c8_prova: USINT;
33 END_VAR
34 CASE ssMethodType OF
35     SS_INITIALIZE:
36         (* SystemInitialize for Chart: '<Root>/prova' incorporates:
37          * Outport: '<Root>/Llum_encesa' *)
38         is_active_c8_prova := 0;
39         is_c8_prova := prova_IN_NO_ACTIVE_CHILD;
40         Llum_encesa := 0.0;
41     SS_STEP:
42         (* Chart: '<Root>/prova' incorporates:
43          * Outport: '<Root>/Llum_encesa' *)
44         (* Gateway: prova *)
45         (* During: prova *)

```

Il·lustració 21. Declaració de variables (font: Pròpia)

```

46     IF is_active_c8_prova = 0 THEN
47         (* Entry: prova *)
48         is_active_c8_prova := 1;
49         (* Entry Internal: prova *)
50         (* Transition: '<S1>:19' *)
51         is_c8_prova := prova_IN_Etapa1;
52     ELSE
53         CASE is_c8_prova OF
54             prova_IN_Etapa1:
55                 (* During 'Etapa1': '<S1>:2' *)
56                 (* '<S1>:5:1' sf_internal_predicateOutput = Pulsador_marcha==1; *)
57                 IF Pulsador_marcha = 1.0 THEN
58                     (* Transition: '<S1>:5' *)
59                     is_c8_prova := prova_IN_Etapa2;
60                     (* Entry 'Etapa2': '<S1>:8' *)
61                     (* '<S1>:8:3' Llum_encesa=1; *)
62                     Llum_encesa := 1.0;
63                 END_IF;
64             ELSE
65                 (* During 'Etapa2': '<S1>:8' *)
66                 (* '<S1>:8:3' Llum_encesa=1; *)
67                 Llum_encesa := 1.0;
68             END_CASE;
69         END_IF;
70         (* End of Chart: '<Root>/prova' *)
71     END CASE;

```

Il·lustració 22. Programa general (font: Pròpia)

2.1.1.3. Rs-Logix 5000 i Programació ST

El Rs-Logix 5000 es un software de programació que s'utilitza per programar una varietat de PLC's de la sèrie logix.

Proporciona un conjunt d'eines de programació, edició i depuració de programa per desenvolupar la lògica de control i automatització de processos. Entre les quals consta de:

1. Editor de lògica (Ladder): Es una eina de programació gràfica que permet generar contactes i bobines entre d'altres per tal de controlar la lògica del sistema.
2. Editor de text (ST) : Es una eina de programació basada en una codificació de text que permet que l'usuari editi els programes en llenguatge estructurat , semblant al C.
3. Eina de simulació : Consta d'una simulació de totes les variables , sensors i actuadors del sistema a controlar.
4. Depurador: Permet que l'usuari depuri el programa en temps real, de tal forma que observa el comportament del controlador mentre executa el programa.

En aquest programa es pot simular la CPU, i totes les targetes de E/S del nostre PLC, per tal de generar una comunicació directa amb el PLC i poder Carregar el programa realitzat o tenir un control en temps real dels estats del PLC.

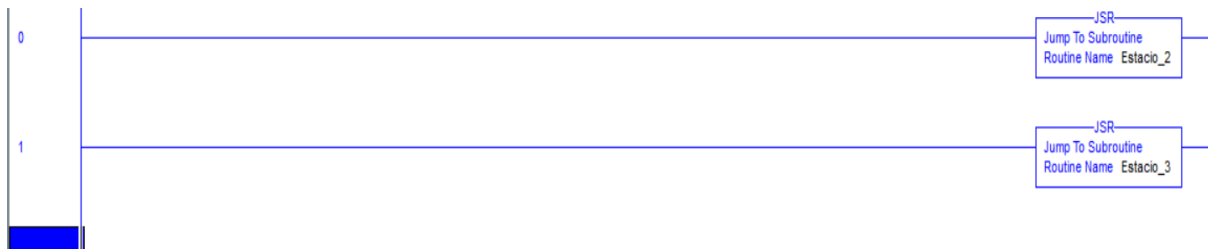
La programació del PLC en ST, es basa en un llenguatge estructurat. En general, la programació en ST consisteix en crear una sèrie de blocs o subrutines , en la que cadascuna s'encarrega de fer una tasca especifica.

Els blocs de codi ST es denominen funcions i consten d'una sèrie de declaracions i comandaments.

1. La declaració de variables: es defineix les variables del programa amb números sencers, reals o booleans.
2. Estructures de control: S'utilitzen per controlar les diferents parts de programa, com bucles utilitzant la funció while o condicionals utilitzant la funció IF,THEN.
3. Funcions i procediments: Es defineixen les diferents funcions o subrutines i son trucades des de el programa principal per fer tasques especificues

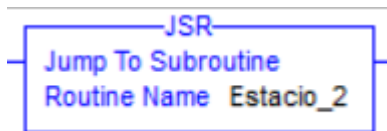
En definitiva la programació ST es una forma molt flexible i capaç de programar un PLC, ja que permet l'estructuració de programes molt detallats i complexos amb la utilització de una

estructura lògica i fàcil d'entendre, a més el llenguatge ST es un tipus de llenguatge estàndard en els PLC significa que no depèn del fabricant del PLC .



Il·lustració 23. Programa principal (font: Pròpia)

Com podem observar en la il·lustració 23 es troba el programa principal on s'utilitza la trucada a diferents blocs o estats de programa mitjançant la funció JSR (Jump Subrutina),



Il·lustració 24.Funció JSR (font: Pròpia)

En la il·lustració 24 podem observar la funció JSR, on s'ha de posar el nom de la rutina a la qual vols accedir, ja pot ser en Ladder o en ST.

```

IF Sensor_Barrera THEN

    etapa6:= 0;
    etapa7 := 1;
    Output_ABB_Subministrament := 0;

END_IF;

IF Sensor_Detecta_Peca_E2 AND etapa7 THEN

    etapa7 := 0;
    etapa8 := 1;
    Output_ABB_Detecta_Palet_E3 := 1;

END_IF;

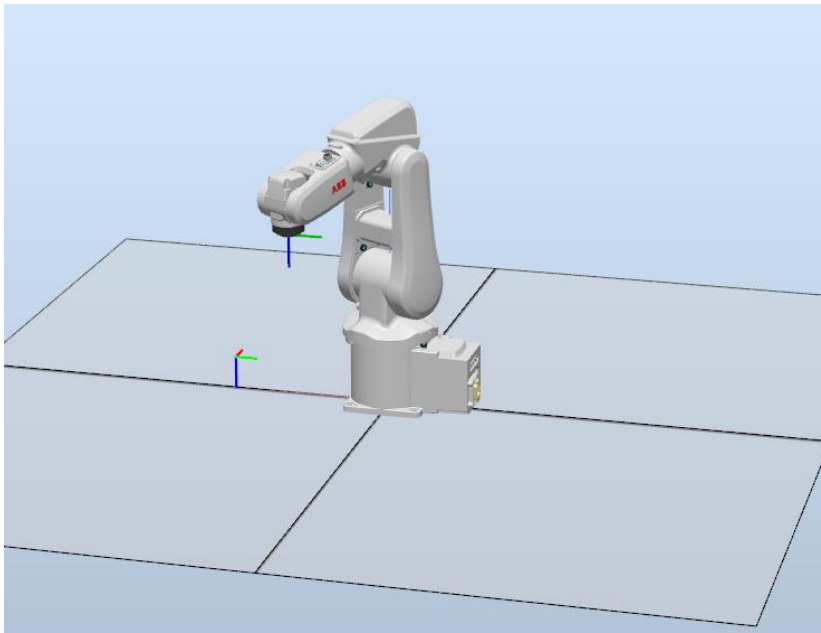
```

Il·lustració 25..Estructura ST(font: Pròpia)

Per la estructura del ST , s'utilitza condicionals i bucles entre d'altres, s'ha de tenir en comte que per assignar un valor o un canvi d'estat a qualsevol actuador o variable dintre del text estructurat ha d'anar acompanyat de '=' per tal de poder assignar-li el valor.

2.1.1.4. Robotstudio

Robot Studio es un software de simulació y programació de robots ABB. Aquest software s'utilitza per el disseny , programació i simulació de sistemes d'automatització basats en robots, facilita el seguiment visual del procés abans d'implementar-ho al mon real.



Il·lustració 26. Interfície gràfica del Robot ABB (font: Pròpia)

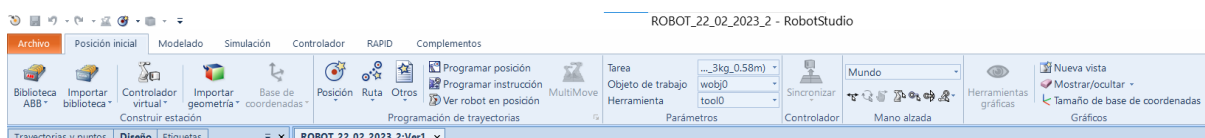
Aquest software genera una gran varietat de beneficis entre d'ells , acceleració del procés de programació, millora de la qualitat del programa, menys costos i major seguretat.

Mitjançant la simulació en temps real de robot Studio permet als programadors de robots una acceleració en el procés de programació degut a que te la facilitat de fer una programació en text estructurat denominat RAPID, a mes de generar una flex pendal virtual per tal de poder simular el procés desitjat abans d'implementar-ho.

Els usuaris del robot amb aquest software, tenen el benefici de detectar possibles problemes de trajectòries o de futurs col·lisions mitjançant la simulació això genera una major qualitat i eficiència de l'automatització ja que es generen moviments mes lineals i dintre dels eixos disponibles del robot.

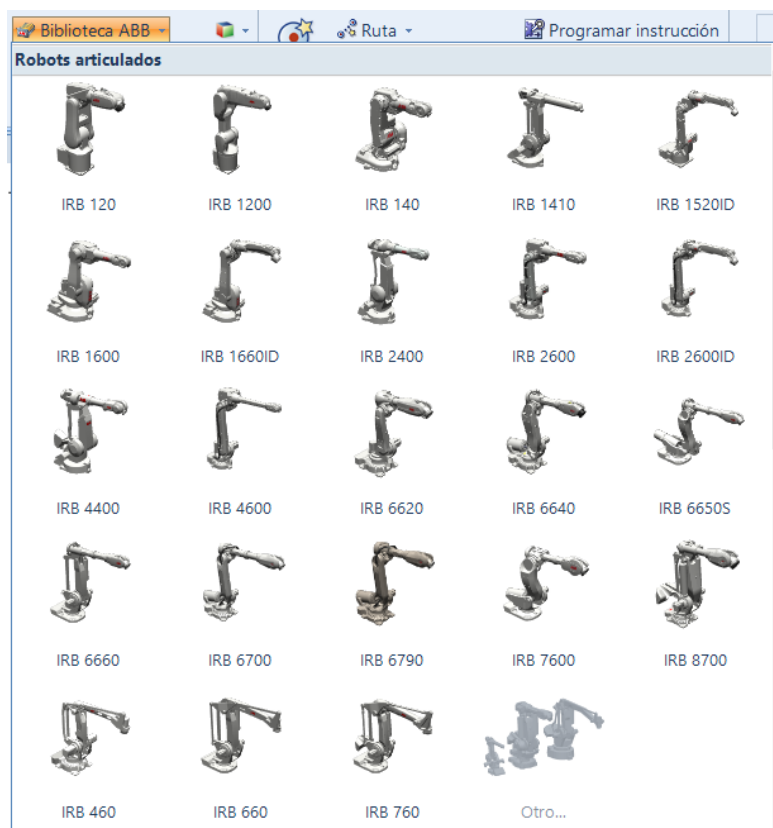
Al reduir el temps de programació i verificar el correcte funcionament del programa això genera un major benefici econòmic.

Finalment la agrupació de tots aquets beneficis genera una major seguretat, degut a que en permet avaluar i millorar la solució d'automatització conjuntament amb l'eficiència i la qualitat del sistema.



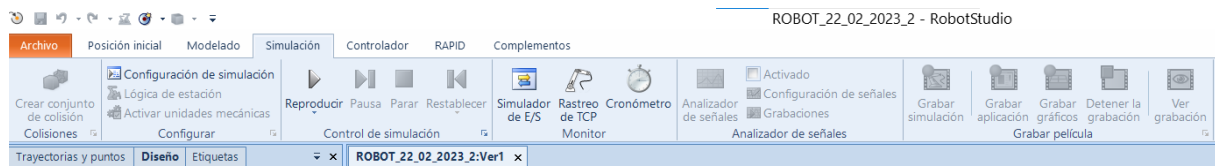
Il·lustració 27.Panel de control de Robot Studio (font: Pròpia)

Aquest programa consta de diferents eines de treball, en la posició inicial , es la ubicació on es pot simular l'estació del robot , escollint el tipus de robot que es vol utilitzar, el tipus de d'eina que es vol implementar al robot i el tipus de moviment que es vol realitzar



Il·lustració 28.Biblioteca de robots ABB (font: Pròpia)

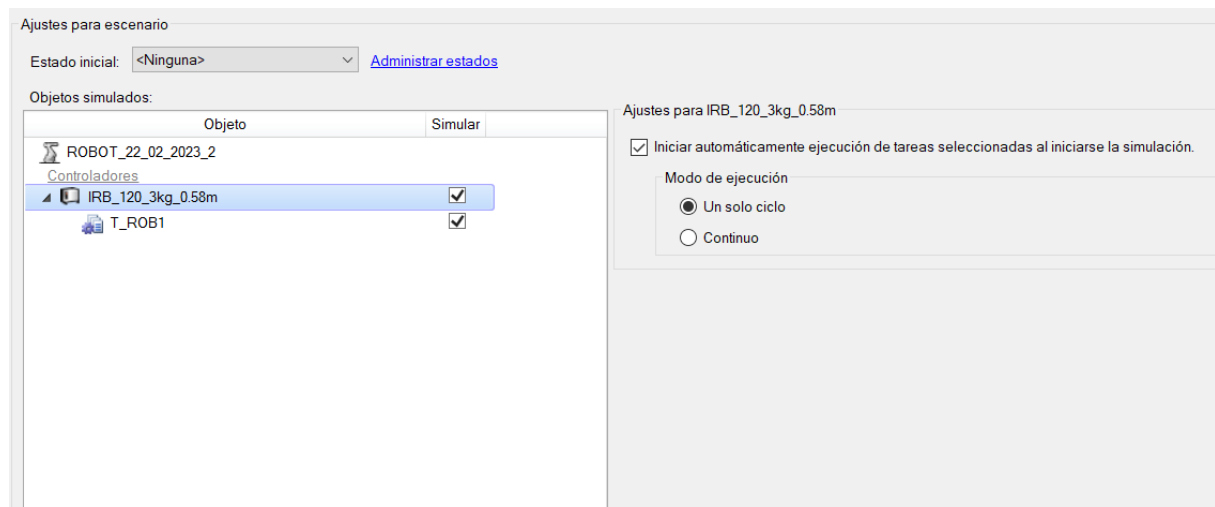
Tal i com podem observar en la il·lustració 28, es pot simular una gran varietat de robots ABB segons las necessitats de l'automatització



Il·lustració 29. Simulació de Robot Studio (font: Pròpia)

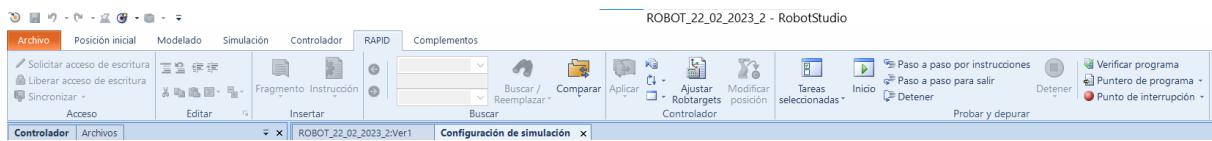
En la il·lustració 29 podem veure que es la pestanya de simulació, on es pot verificar l'estat del programa i del seu funcionament , conjuntament amb la simulació de les entrades i les sortides que es troben connectades al robot.

Es pot configurar per tal de que es el procés de simulació sigui continu o de un simple cicle, tenint en compte els diferents controladors que es poden tenir segons el tipus d'automatització.



Il·lustració 30. Configuració de simulació (font: Pròpia)

Finalment trobem la pestanya de programació de Rapid.



Il·lustració 31. Pestanya de Ràpid Robot Studio (font: Pròpia)

En la il·lustració 31 es pot veure la gran capacitat d'eines hi ha per poder realitzar la programació i el control del sistema, on consta d'entre d'altres en la configuració i declaració de les entrades i sortides del programa, la depuració i control exhaustiu de línia per línia del programa amb l'ajuda de la depuració de pas per pas.

La programació de ràpid consta d'una estructura molt metòdica que es la següent:

```

1  MODULE Module1
2  ▣  CONST robtarget P1:=[300,200,150],[0,-0.707106781,0.707106781,0],[0,0,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09];
3  VAR NUM P1_X:=0;
4
5  VAR NUM P1_Y:=0;
6  VAR NUM i:=0;
7  VAR NUM c:=0;
8  ▣  CONST robtarget INICIO:=[146.64,283.61,470.14],[0,-0.707106781,0.707106781,0],[0,0,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09];
a  !*****

```

Il·lustració 32. Declaració de variables de programa(font: Pròpia)

Principalment en la part superior del programa es declaren les variables que s'utilitzaran, podent ser constants de tipus sencer o una assignació d'una posició.

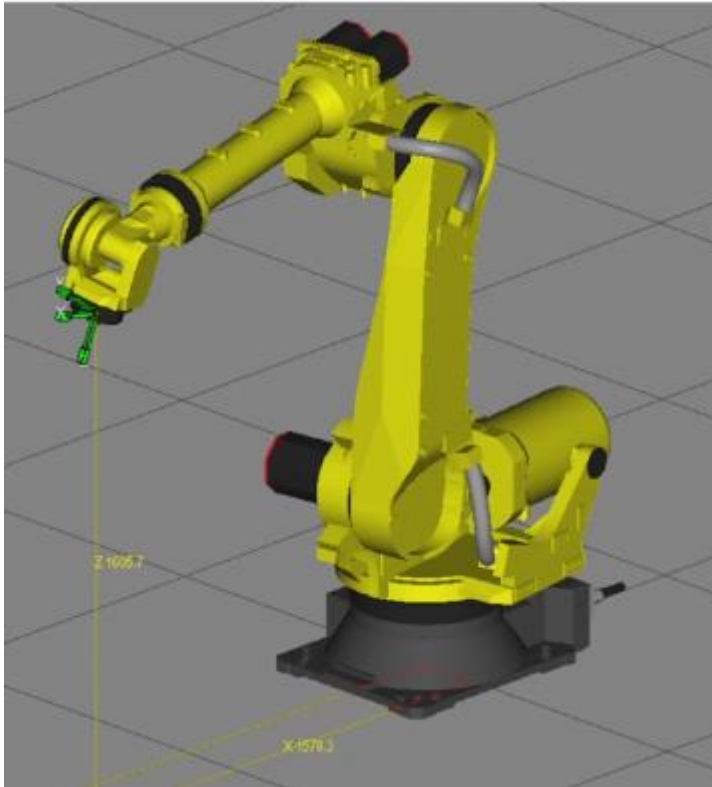
```
29  PROC main()
30
31
32
33  IF DI10_1 =1 THEN
34      Path_INICIO;
35
36
37  ENDIF
38
39
40
41  IF DI10_2=1 THEN
42      SUBMINISTRADOR;
43
44
45
46
47  ENDIF
48
49  IF DI10_4=1 THEN !PEÇA CORRECTA DI10.4
50
51      DETECTA_PALET_E3;
52
53
54  ELSE
55      !IPONER CODIGO DE IR A CONTAINER
56
```

Il·lustració 33. Programa principal (font: Pròpia)

Seguidament es troba el programa principal on s'executarà tot el procés programat , realitzant trucades a altres subrutines o executant el programa seqüencialment.

2.1.1.5. Roboguide

Roboguide es un software de programació per a robots FANUC. Aquest software s'utilitza per el disseny, programació i simulació de sistemes d'automatització basats en robots, facilitant a l'usuari una visualització gràfica i comprovació abans d'implementar-ho al mon real.



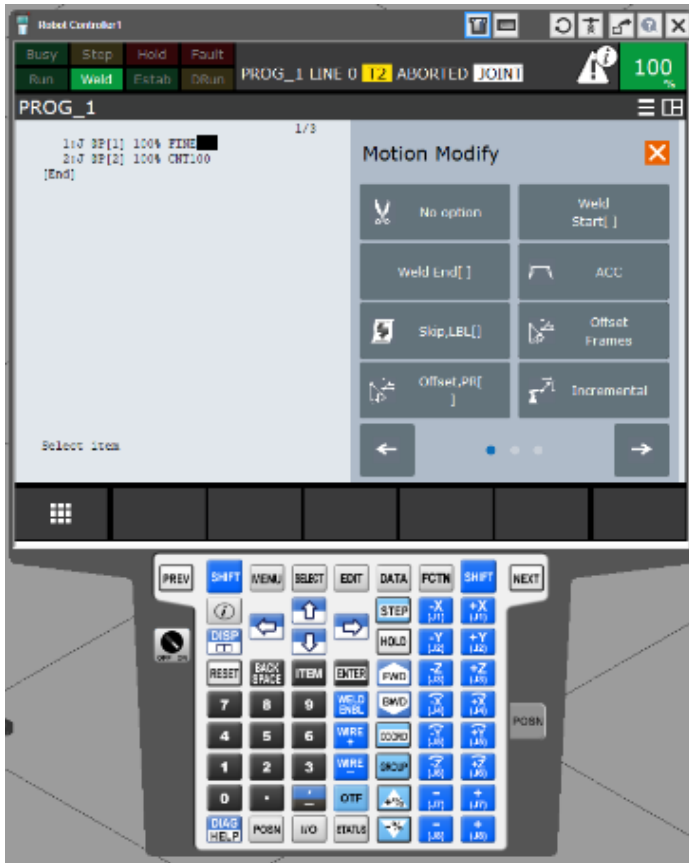
Il·lustració 34. Interfície gràfica del robot FANUC (font: Pròpia)

Roboguide ofereix molt beneficis entre d'ells són:

- Acceleració del procés de programació: Genera una simulació en temps real de tal forma que permet als programadors reduir el temps necessari per posar apunt la solució d'automatització.
- Millora la qualitat del programa: Al tenir la capacitat de simular tot el procés ajuda a evitar futurs problemes amb les trajectòries de robot o amb possibles col·lisions , generant així moviments més lineals i productius.
- Millor aprenentatge: Amb l'eina d'entrenament que facilita ajuda als operadors de robots a millorar el seu aprenentatge i control exhaustiu dels robots de forma més ràpida

Finalment amb l'acceleració del procés i la millora de qualitat genera un benefici econòmic i de seguretat, degut a tot el procés ha estat prèviament simulat evitant així futurs problemes, millorant així l'eficiència i la qualitat del procés.

La programació d'aquest robot es realitza mitjançant una simulació de la flex pental, en la qual es facilita la implementació de condicionals, de bucles de programa o de trucada a subrutines entre d'altres.



Il·lustració 35.Simulació Gràfica de Flex pental Roboguide (font: Pròpia)

2.2.Viabilitat Mediambiental

El nostre projecte en general no té un gran impacte mediambiental a tenir en compte , degut a que esta enfocat en una maqueta docent, no fabricarà cap producte. Es reflexa en el document mediambiental situat a l'annex

Com a molt tindriem que tenir en compte el consum elèctric que causaria , que ve donat per el rang horari d'utilització

2.3.Viabilitat econòmica i pressupost

El projecte a l'estar destinat únicament a estudi docent no generarà cap benefici econòmic ni producció de peces .

De tal forma que la seva rendibilitat pura, és l'aprenentatge i la utilització productiva de la cèl·lula.

Igualment es realitza un pressupost , per tal de veure el cost que constaria la programació i el estudi de la cèl·lula.

S'estableix un preu estàndard de 20€/h que s'aplica proporcionalment a totes les tasques del projecte.

Nom del recurs	Tipus	Taxa estàndard	Taxa hores extra
Joel	Treball	20,00 €/hora	27,00 €/hora

Taula 1.Preu dels recursos

Capítol 1: Realització del projecte			
Codi	Nom de la tasca	Duració	Cost
1.1	Avantprojecte	80 hores	1.600€
1.2	Modelització amb Matlab	32 hores	640€
1.3	Programació d'Autòmat Roswell	85 hores	1.700 €
1.4	Programació Robot ABB IRB120	32 hores	640€
1.5	Programació Robot Fanuc LR Mate 200d 4S	32 hores	640€
1.6	Programació de scada en Hmi	40 hores	800€
1.7	Probes de qualitat d'operació	80 hores	1.600€
1.8	Memòria del Projecte	100 hores	2.000€
Total:			10.260€
(marge condicional del 30%) Total:			13.3386€

Taula 2.CapitolI de pressupost, Realització del projecte

En el nostre projecte no tenim cap cost de material ja que tot el nostre treball es programació amb la utilització de softwares, per tant el capítol 2 resulta un cost de 0€

Capítol 3: Amortitzacions				
Codi	Amortitzacions	Quantitat	Cost unitari	Cost total
3.1	Ordinador	1	687,14€	687,14€
3.2	Llicència Standard de matlab	1	860€	860€
3.3	Desplaçament per reunions	320km= 32l	1,588€/l	50,82€
Total:				1597,96€

Taula 3.Capitol3 de pressupost, Amortitzacions

Pressupost total

Pressupost	
Capítols	Cost total
Total Capítol 1	13.338€
Total Capítol 2	0€
Total Capítol 3	1597,96€
Total:	22.813,95€
IVA 21€:	4790,93€
Total Pressupost:	18.072,51€

Taula 4..Pressupost total

Capítol III. Desenvolupament de la solució

La solució aplicada per el desenvolupament desitjat ,segons les especificacions prefixades es la següent:

- Principalment, s'ha dissenyat un procés a realitzar, el qual consta del control de la cèl·lula de fabricació flexible situada al lab4 del Tecnocampus.
- On es controla 4 diferents estacions , estació de subministrament, detecció de tipus de peça , de foradat y de magatzem.
Per tal de enllaça totes les estacions s'utilitza 2 robots, un robot ABB per les estacions de subministrament ,detecció tipus de peça , magatzem y el robot FANUC per controlar l'estació de foradat.
- Es vol fer pacs de 2 peces metàl·liques en cada cycle amb un forat en el seu centre.
Per tal de traslladar-les s'utilitza un palet amb 2 forats on encaixen les peces , que es mogut per unes cintes mecàniques.
Dintres de l'estació de detecció de peces , es descarten les peces de plàstic a la cubeta y les metàl·liques es porten cap al palet .
Per tal de verificar si el forat s'ha realitzat correctament s'utilitza l'estació de foradat , si el forat no s'ha fet correctament la peça es descartada y si s'ha fet correctament es torna a posar en el palet.
Finalment actuen les cintes fins arribar al lloc desitjat y el robot ABB emmagatzema les peces.
- Es fa una modelització amb matlab ,per controlar el sistema de control que posteriorment volem realitzar, per aquest procés simularem totes les variables per tal de comprovar que la lògica de procés que es correcte.
- Una vegada verificada la lògica de procés amb Matlab, es trasllada aquesta informació al software Rslogix5000 amb codi de programació ST, al RobotStudio per controlar el robot ABB realitzat amb funcions y al RoboGuide per controlar el robot FANUC realitzat amb funcions.

3.1 Disseny i construcció del GRAFCETS

Per l'inici del desenvolupament de la solució més òptima, es comença amb la utilització dels previs coneixements extrets en els antecedents, per tal de dividir en programa en subrutines segons les estacions que es troben a la cèl·lula.

Els graficets es divideixen en graficets del PLC i graficets per cadascun dels robots

Graficets del PLC

Graficets de robot ABB

Graficets de robot FANUC

Una vegada finalitzats els Graficets, es posa en marxa la modelització de la màquina d'estats, en la qual s'utilitzaran els coneixements marcats en els graficets del PLC, de tal forma que es pot comprovar tècnicament si és possible la implementació desitjada.

La implementació dels graficets es pot veure representats en les il·lustracions (37,39,41,43,44,45,46.2 i 49)

3.2 Modelització i simulació de màquina d'estats

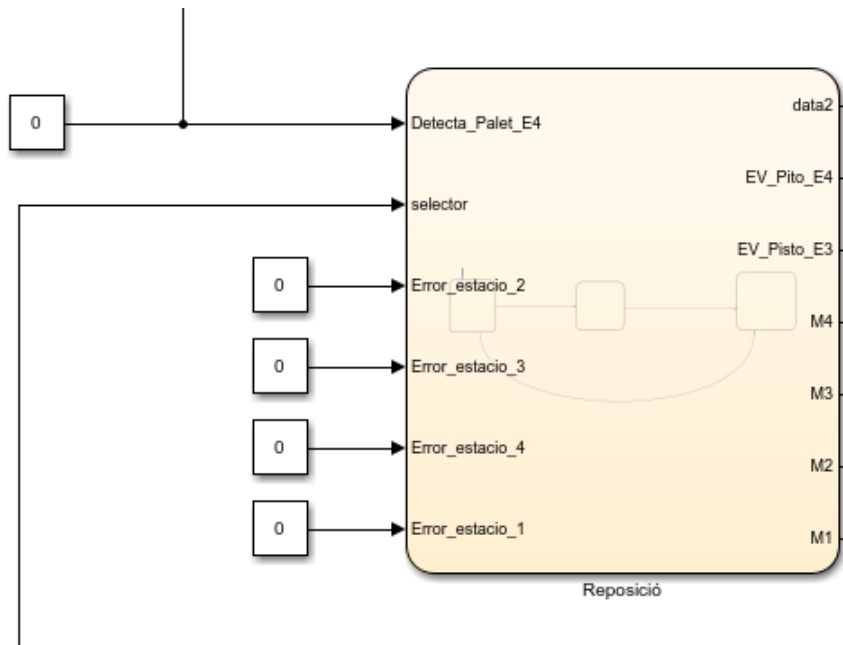
3.2.1 Programació de Màquina d'estats.

Per la programació de la màquina d'estats s'utilitza un tipus de programació gràfica amb la utilització de portes lògiques i constants.

El primer bloc que es troba en la programació és el de reposició, per tal de simular en qualsevol error en les estacions i poder inicialitzar una altra vegada el procés de la cèl·lula amb el re posicionament corresponent.

En aquesta modelització es trobarà tot el procés de la cel·lula simulat, utilitzant 1 únic cicle de procés i idealitzant que no hi ha cap problema a l'hora de seleccionar el correcte estat de les peces.

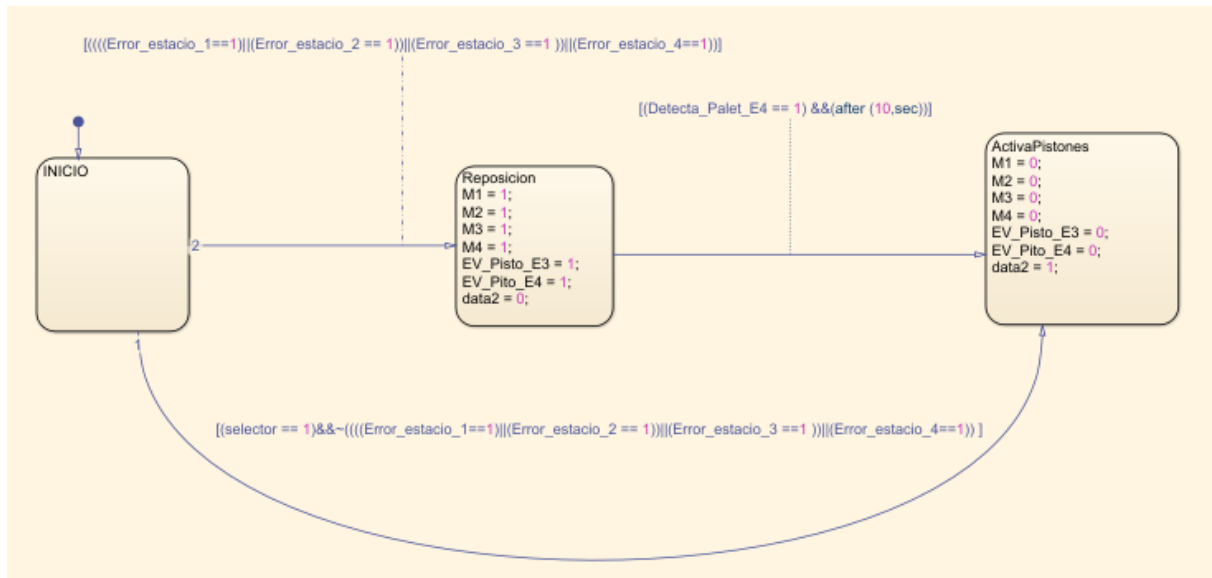
MODELITZACIÓ DE BLOC D'ESTACIÓ REPOSICIÓ



Il·lustració 36..Bloc de reposició màquina d'estats (font: Pròpia)

En la il·lustració 36, es troben les entrades habilitades per simular els errors de la cèl·lula i la entrada del selector d'engegada de corrent, per tal de simular quan el PLC te encesa la CPU. Així que aquest bloc serveix per la inicialització de la cèl·lula o per la reposició dels errors en cada estació.

MODELITZACIÓ DEL GRAFCET D'ESTACIÓ REPOSICIÓ

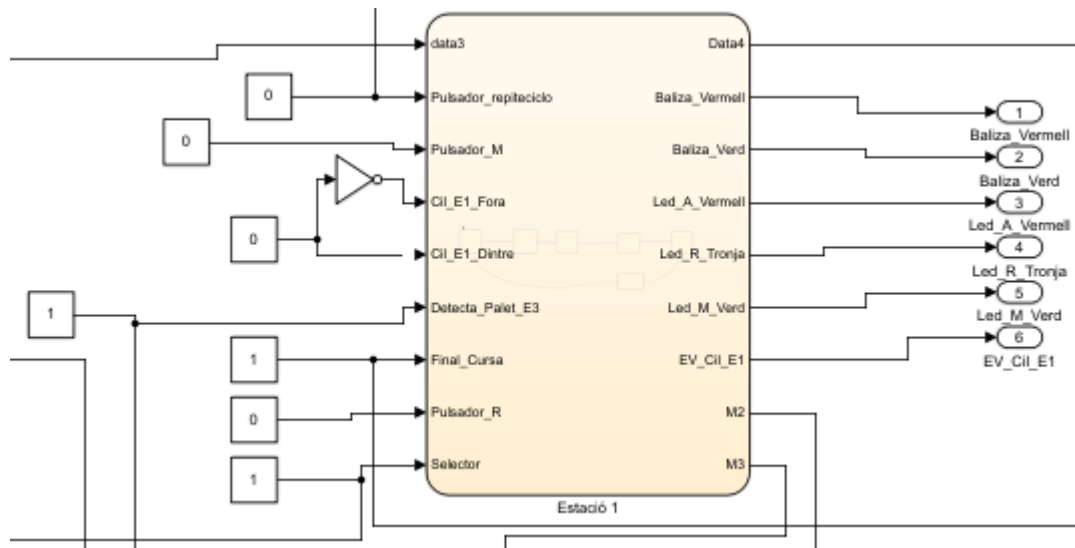


Il·lustració 37. Programació d'estats del bloc reposició (font: Pròpia)

Tal i com podem veure en la il·lustració 37, hi ha dos camins per inicialitzar la cèl·lula, el primer que es amb la programació d'un paral·lel dels errors possibles, fa que s'inicialitzin els motors de la cèl·lula i les electrovàlvules dels pistons, de tal forma que el palet de la cèl·lula pugui anar fins l'estació 1 sense cap bloqueig. El altre camí es si es dona tensió en el selector i no hi ha cap error, s'inicialitza la sortida data2 que aquesta habilita el següent bloc

Una vegada activada la sortida data2 fa activar la data3 de tal forma que habilita el bloc d'estació 1.

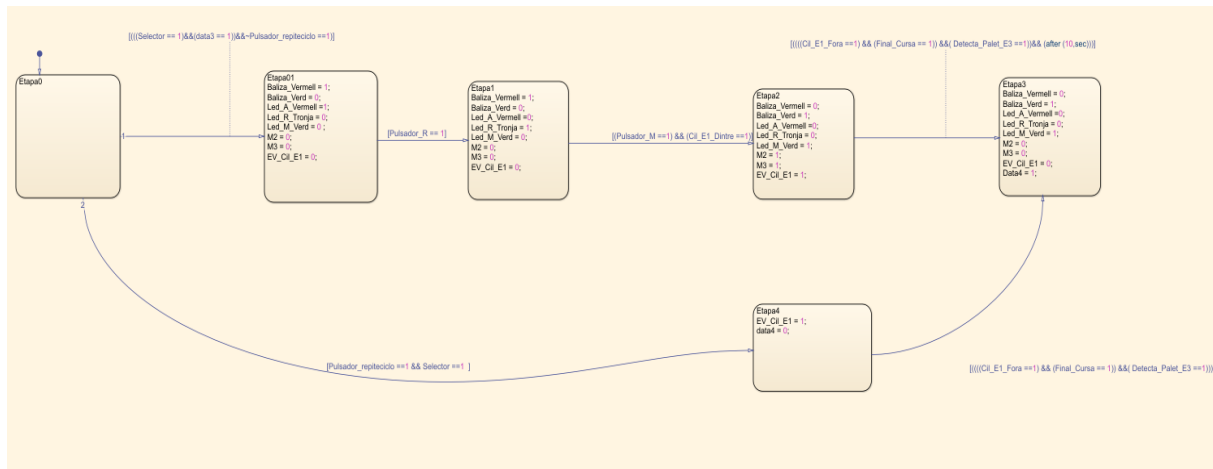
MODELITZACIÓ DE BLOC D'ESTACIÓ SUBMINISTRAMENT



Il·lustració 38. Bloc estació1 subministrament maquina d'estats (font: Pròpia)

En la il·lustració 38 es troba el bloc de l'estació 1 que es l'encarregada de subministrar les peces a la cèl·lula. Tal i com podem veure es troben les entrades data3 que s'encarrega d'habilitar el bloc, el pulsador de marxa que es l'encarregat d'inicialitzar el procés, l'estat del cil_E1 que es el cilindre que s'encarrega d'empènyer la peça del subministrador, el sensor inductiu detecta_palet_E3 que com pròpiament diu el nom es l'encarregat de donar feedback del correcte posicionament del palet , el final de cursa que s'encarrega de comprovar que la peça esta en el lloc específic i el pulsador de rearmament . En les sortides es troba l'estat visual de la balisa de la cèl·lula i l'estat dels led's que es troben en el quadre de control per tal de tenir el control visual de la cèl·lula i la electrovàlvula que habilita el cilindre de subministrament E1.

MODELITZACIÓ DEL GRAFCET D'ESTACIÓ SUBMINISTRAMENT

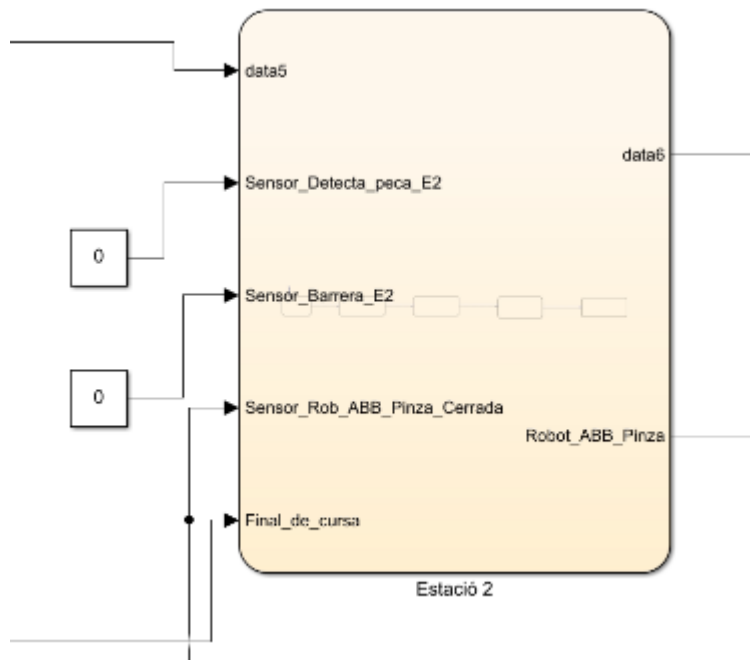


Il·lustració 39. Programació d'estats del bloc d'estació 1 subministrament (font: Pròpia)

En la il·lustració 39 es pot veure la programació d'estats de l'estació de subministrament, de tal forma que quan, es prem el pulsador de rearmament, pulsador de marxa, el cilindre de subministrament està sortit, el palet detecta que està al lloc apropiat i el final de cursa detecta que té peça, s'activa la Data4 per tal d'habilitar l'estació 2.

Una vegada activada la sortida data4 fa activar la data5 de tal forma que habilita el bloc d'estació 2.

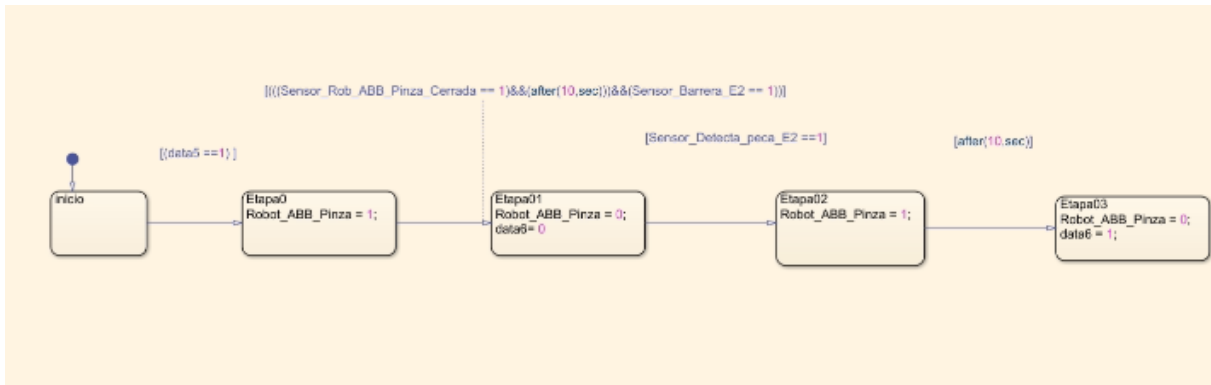
MODELITZACIÓ DE BLOC D'ESTACIÓ DETECCIÓ TIPUS DE PEÇA



Il·lustració 40..Bloc estació2 detecció tipus de peça maquina d'estats (font: Pròpia)

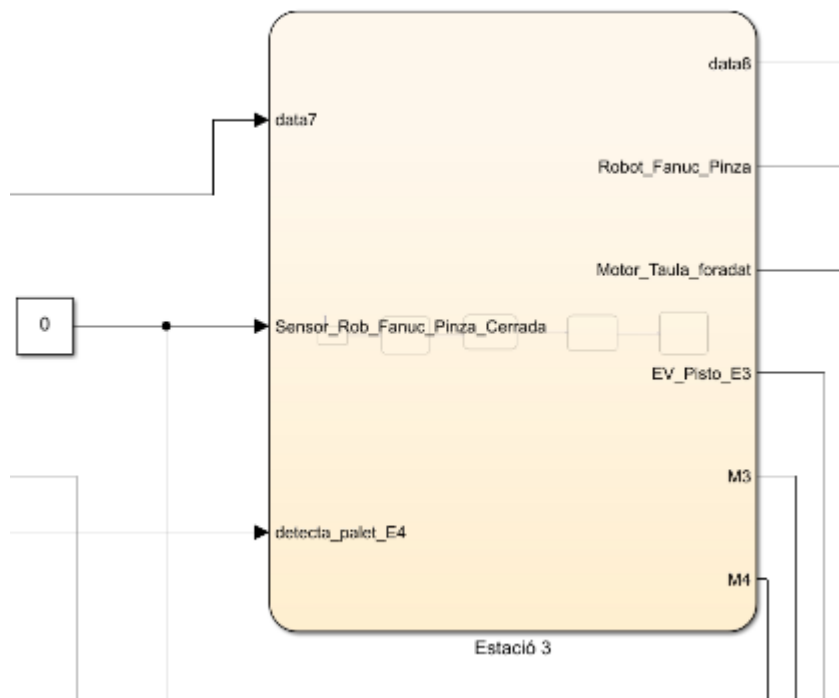
En la il·lustració 40 podem veure el bloc general de la estació encarregada en detecció de tipus de peça, consta de el sensor òptic de barrera per tal de detectar que hi ha peça i el sensor de detectar peça que es un inductiu encarregat de detectar les peces metàl·liques , es consta de el feedback del robot ABB per detectar que te la pinça tancada. Seguidament es troba el feedback del PLC al robot per donar l'ordre de tancar pinça

MODELITZACIÓ DEL GRAFCET D'ESTACIÓ DETECCIÓ TIPUS DE PEÇA



Il·lustració 41. Programació d'estats del bloc d'estació de detecció tipus de peça (font: Pròpia)

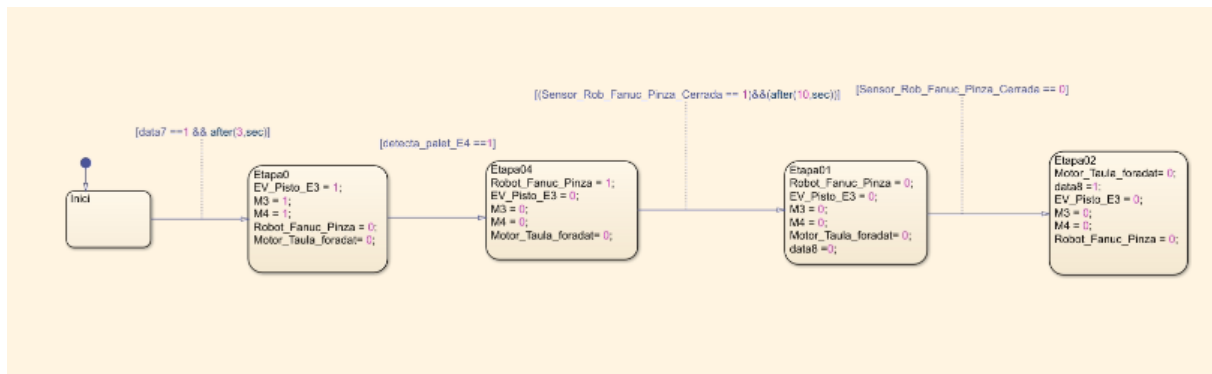
Tal i com es pot observar en la il·lustració 41 es pot determinar que principalment el robot té la pinça tancada amb la peça agafada, quan arriba al lloc destinat el robot ABB envia el feedback com que ha arribat al lloc desitjat i detecta el sensor òptic de barrera no detecta que hi ha peça, el robot obre la pinça, quan detecta que es troba una peça metàl·lica el robot torna a agafar la peça que es troba en la estació de detecció tipus de peça i espera 10 segons simulant el moviment del robot, una vegada passa el temps torna a obrir la pinça per deixar la peça en el palet.

MODELITZACIÓ DE BLOC DE PALET A ESTACIÓ FORADAT.

Il·lustració 42. Bloc estació3 de palet a foradat maquina d'estats (font: Pròpia)

En el bloc de simulació de la il·lustració 42, es troben les entrades de el feedback de que l'anterior procés ha finalitzat, el feedback de Fanuc pinça tancada i el detector inductiu de que esta en la estació de foradat. Com a sortides trobem el feedback de que el procés ha finalitzat, activació de la pinça del Fanuc, motor de la taula per fer girar, electrovàlvula del pistó E3 per deixar que el palet pugui arribar fins la estació de foradat i els motors de la cinta que en comptes de controlar la cinta amb un sol motor utilitzo 4 independents.

MODELITZACIÓ DEL GRAFCET D'ESTACIÓ DE PALET A FORADAT



Il·lustració 43. Programació d'estats del bloc d'estació de palet fins a foradat (font: Pròpia)

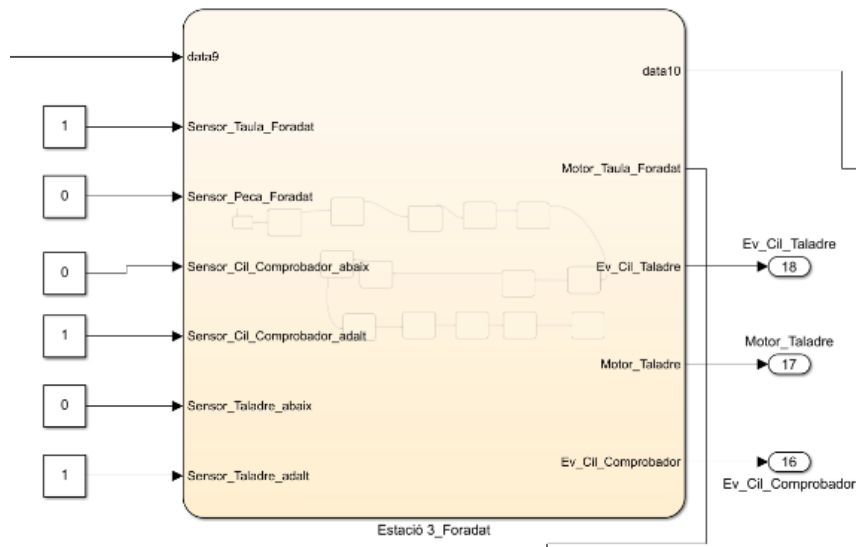
En la primera part del grafcet de simulació es troba un temps de seguretat de 3 segons per tal de que el robot ABB hagi deixat la peça dintre del palet, directament s'activa la EV del pistó E3 per deixar passar al palet i se inicialitzen els motors per tal de fer arribar el palet fins a estació de foradat.

Una vegada arriba al lloc desitjat es detecta per el sensor de palet en estació de foradat s'aturen els motors ,directament li envia un feedback al robot amb la consigna de que hi ha robot en posició perquè agafi la peça .

Quan pesen 10 segons per simular el moviment del robot de palet a plat el robot torna a obrir la pinça per tal de deixar la peça en el seu lloc .

Finalment el robot detecta que la pinça es oberta.

MODELITZACIÓ DE BLOC D'ESTACIÓ FORADAT

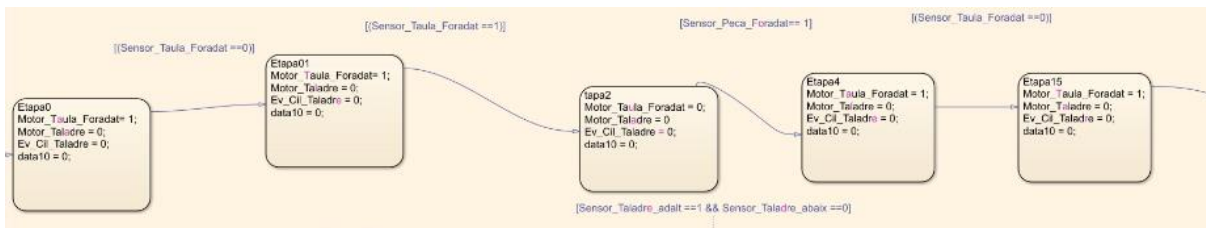


Il·lustració 44. Bloc estació3 foradat màquina d'estats (font: Pròpia)

En el bloc de la il·lustració 44 de programa, es troba la simulació de l'estació de foradat, consta de el sensor de taula foradat per tal de controlar la posició de la taula, un sensor inductiu per saber que la peça està circulant, dos sensors reed per detectar que el cilindre de comprovació està a dalt o baix i uns altres 2 per detectar quan el taladre hi és a dalt o a baix.

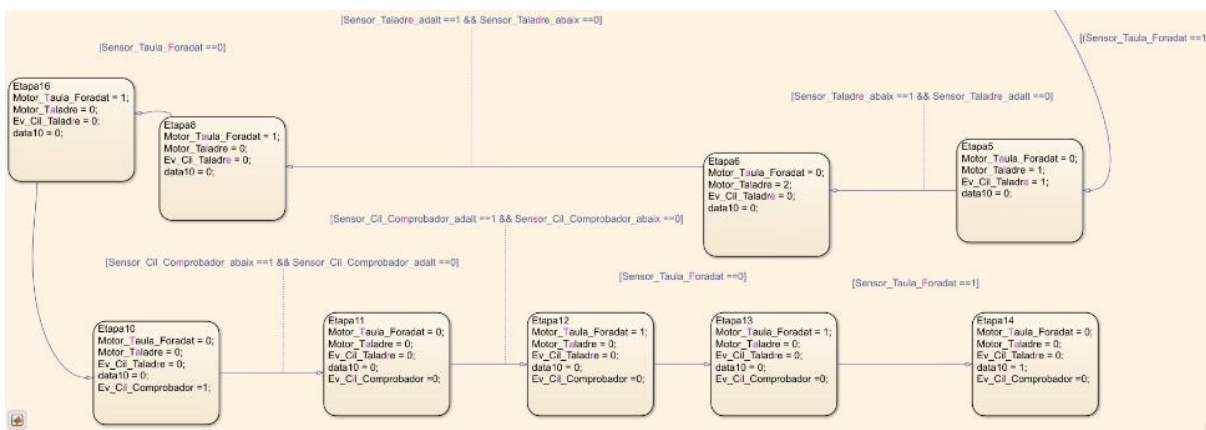
Com a actuadors s'utilitza el motor del taladre que pot girar en sentit horari per taladrar i en sentit anti-horari per tornar a posició de repòs, una EV de doble efecte del taladre i una altra EV de simple efecte retorn per molla del cilindre comprovador, a més del feedback de que aquest procés s'ha finalitzat.

MODELITZACIÓ DEL GRAFCET D'ESTACIÓ FORADAT



Il·lustració 44. Programació d'estats del bloc d'estació3 foradat part1 (font: Pròpia)

Principalment quan arriba el feedback de que l'anterior procés s'ha finalitzat, s'activa el motor de la taula per tal de portar la peça fins el taladre, per arribar fins a la posició desitjada s'ha de fer 2 flancs de pujada del sensor de taula .



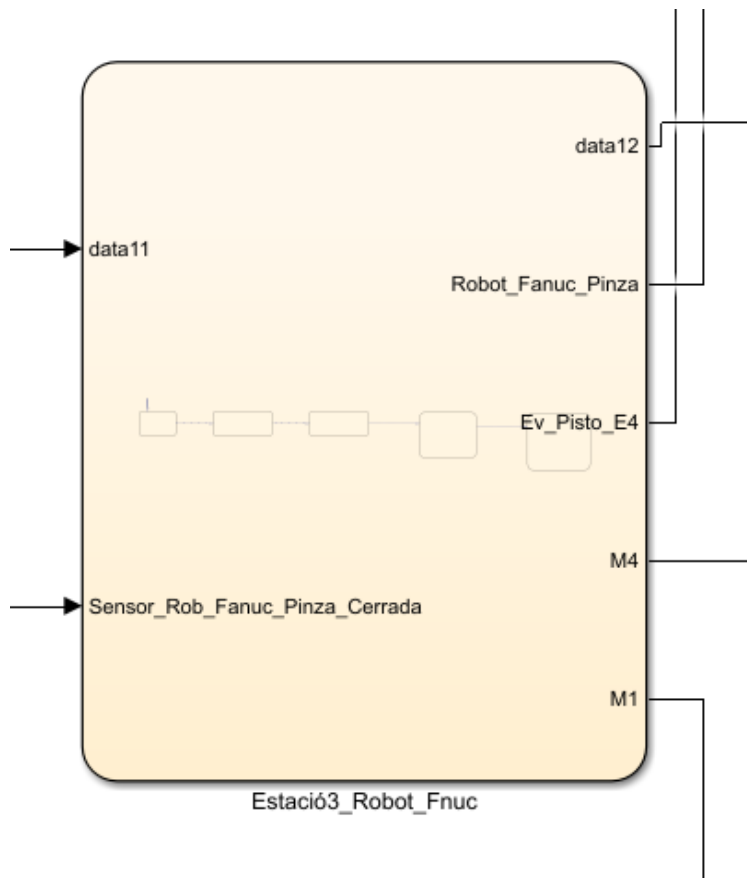
Il·lustració 45. Programació d'estats del bloc d'estació3 foradat part2 (font: Pròpia)

Una vegada ha finalitzat els 2 flancs de pujada del sensor de taula, s'atura el motor de la cinta i s'encén el motor en sentit horari, actua la EV de doble efecte del taladre per tal de fer baixar el taladre i començar a foradar la peça.

Quan el sensor reed de taladre abaix detecta que ha fet el forat i no detecta que esta a dalt, el motor canvia de girar en sentit horari i comença a funcionar en sentit anti horari per retirar el taladre conjuntament amb la desactivació de la vàlvula del cilindre taladre.

Seguidament quan el sensor reed de taladre a dalt detecta i el de sensor abaix deixa de detectar , s'encén el motor de la taula i la fa girar fins al lloc on es troba el cilindre de comprovació amb la ajuda de 1 flanc de pujada del sensor detecta taula.

Finalment baixa el pistó de comprovació actuat per la EV de comprovació i quan detecta que es correcta el forat amb el sensor reed que te a la part de abaix.

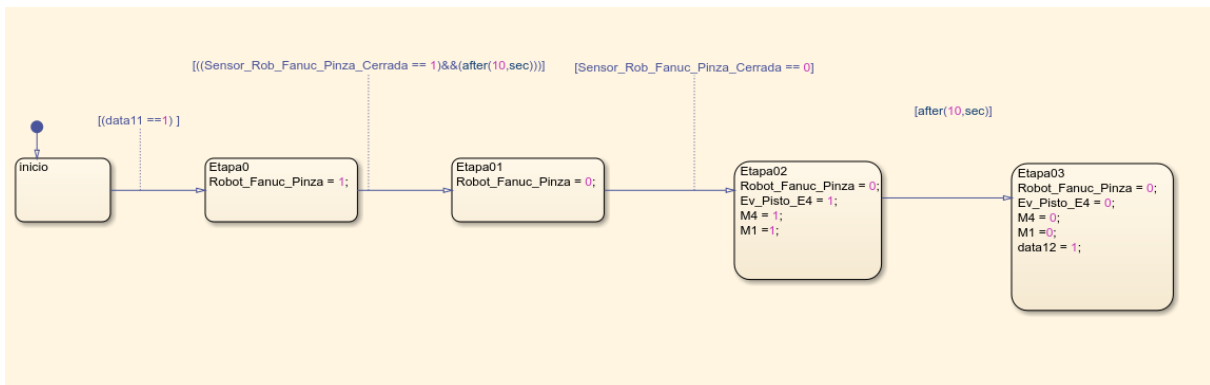
MODELITZACIÓ DE BLOC D'ESTACIO DE PLAT A PALET

Il·lustració 46.1. Bloc estació3 de plat a palet maquina d'estats (font: Pròpia)

En el bloc de la il·lustració 46.1 com podem observar de encarrega de agafar la peça del plat amb el robot Fanuc i transportar-la fins al palet.

Principalment es rep el feedback de que el procés anterior s'ha finalitzat, i controla amb el feedback del PLC al robot confirmant que la peça es llesta per ser agafada.

Com a sortides trobem el feedback de procés finalitzat, el feedback que li envia el robot al PLC confirmant que te la peça agafada, la EV de pistó E4 que s'encarrega de deixar passar el palet fins la estació 4 de emmagatzema, finalment per transportar el palet s'utilitza el feedback de 2 motors.

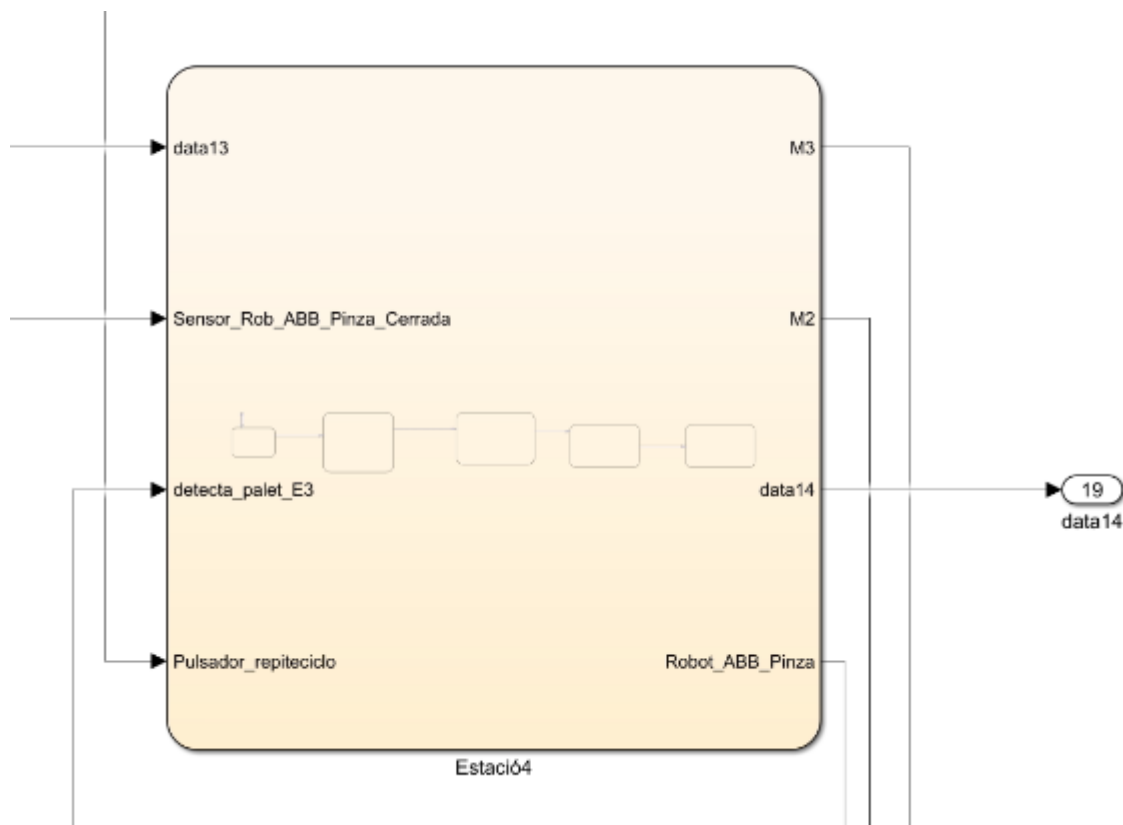
MODELITZACIÓ DEL GRAFCET DEL PLAT AL PALET

Il·lustració 46.2. Programació d'estats de blocs estació magatzem maquina d'estats (font: Pròpia)

Per tal de controlar el moviment de plat a palet, principalment confirma el feedback de que s'ha acabat l'estat anterior, el PLC envia feedback de que la peça esta en el lloc desitjat i conta 10 segons per simular el temps de moviment del robot, una vegada pesen els 10 segons el robot obra la pinça i deixa la peça en el palet, seguidament el robot li envia un feedback al PLC per confirmar de que ha deixat la peça en el lloc.

Finalment actua la EV deixant passar el palet amb la ajuda de l'actuació dels motors, conta 10 segons simulant que el palet arriba al lloc desitjat.

MODELITZACIÓ DE BLOC D'ESTACIÓ MAGATZEM

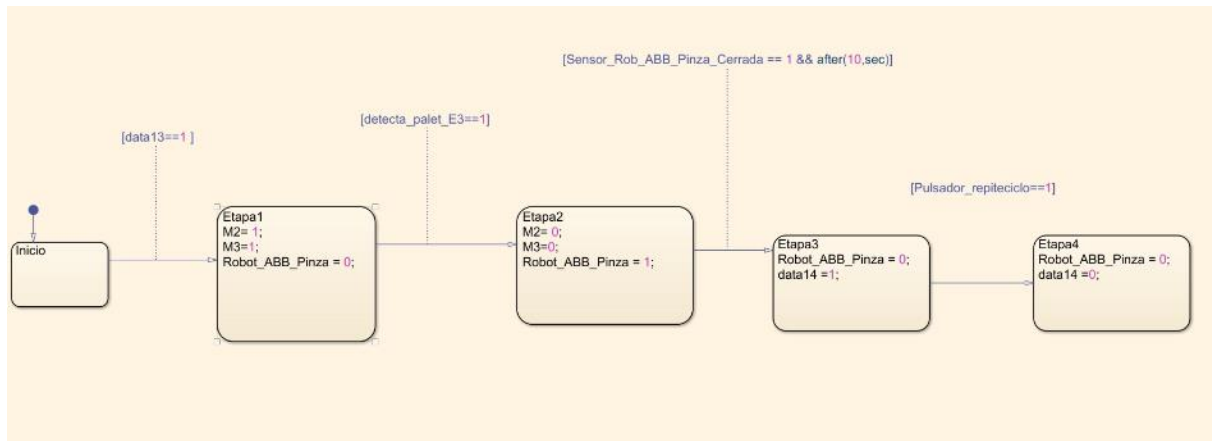


Il·lustració 47. Bloc estació4 emmagatzema peça maquina d'estats (font: Pròpia)

En aquest bloc de estació 4 magatzem es troba el feedback de que l'anterior procés s'ha realitzat correctament, el feedback que envia el PLC al robot ABB de pinça tancada per tal d'agafar la peça del palet i emmagatzemar-la, detector de palet en E3 que es el lloc on el robot anirà a agafar la peça es controlat amb un sensor per saber la seva posició i un polsador inutilitzat actualment que es el de repetir cicle de treball.

Com a actuadors es troba l'actuació dels motors per tal de traslladar el palet des de la estació de foradat fins la estaco de magatzem per les cintes mecàniques, es te el feedback del robot ABB que va cap a el PLC com a consigna de que te pinça tancada i el feedback de process finalitzat.

MODELITZACIÓ DEL GRAFCET D'ESTACIÓ MAGATZEM



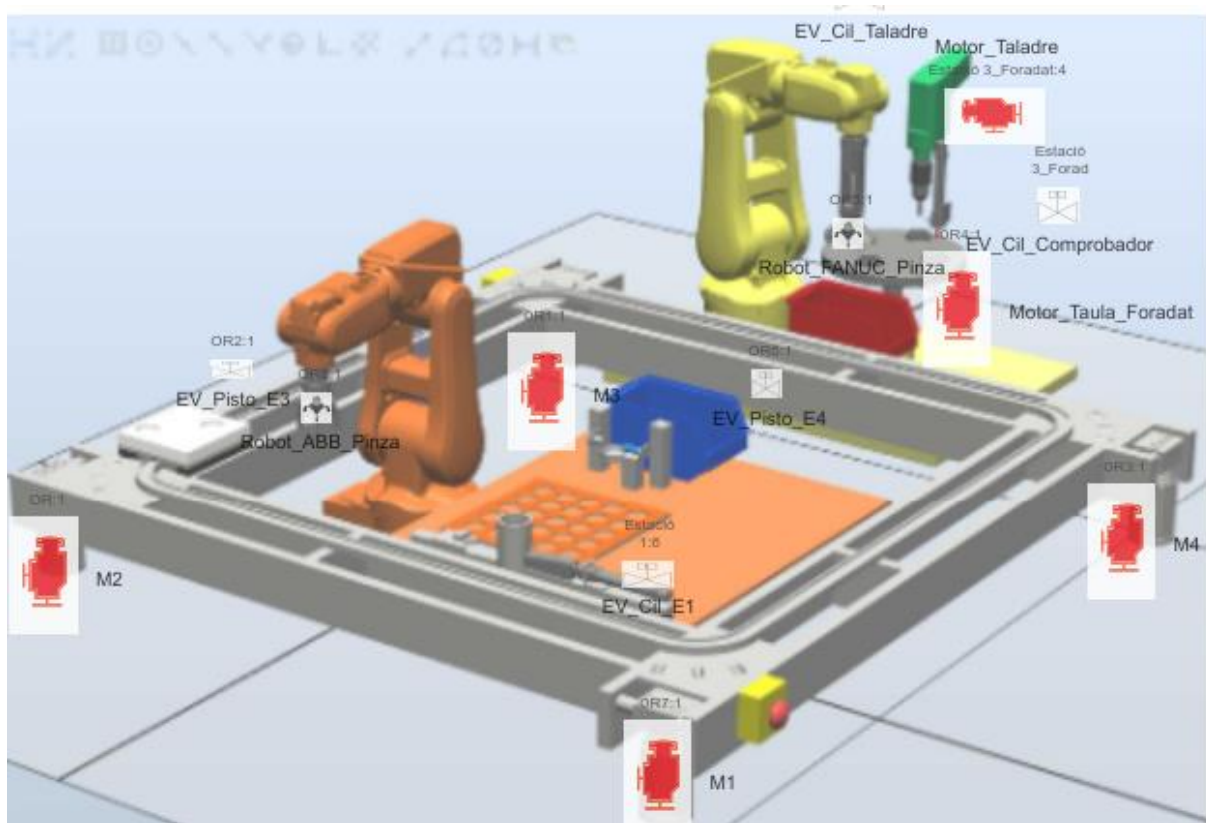
Il·lustració 48. Programació d'estats del bloc d'estació 4 emmagatzema peça (font: Pròpia)

Per la modelització del magatzem principalment hi ha aplicat un temps de seguretat de 3 segons per validar de que el robot FANUK transporta la peça del plat al palet, una vegada passa aquest temps s'inicialitzen els motors per tar de fer moure el palet fins a el lloc desitjat, quan arriba al detector de palet E3 s'aturen els motors i envia el feedback al robot de que la peça esta en lloc, el robot tanca la pinça i es simula el temps del moviment del robot amb 10 segons, una vegada ha passat aquest temps de simulació el robot obra la pinça i deixa la peça emmagatzemada.

Per finalitzar hi ha un pulsador per repetir cicle que esta inhabilitat.

3.2.2 Simulació de Maquina d'estats

La simulació de la cèl·lula de fabricació flexible consta de 2 grans apartats , la part gràfica que es un scada recreant la reposició i l'estat de tots els elements de la cèl·lula on es troben situats els actuadors

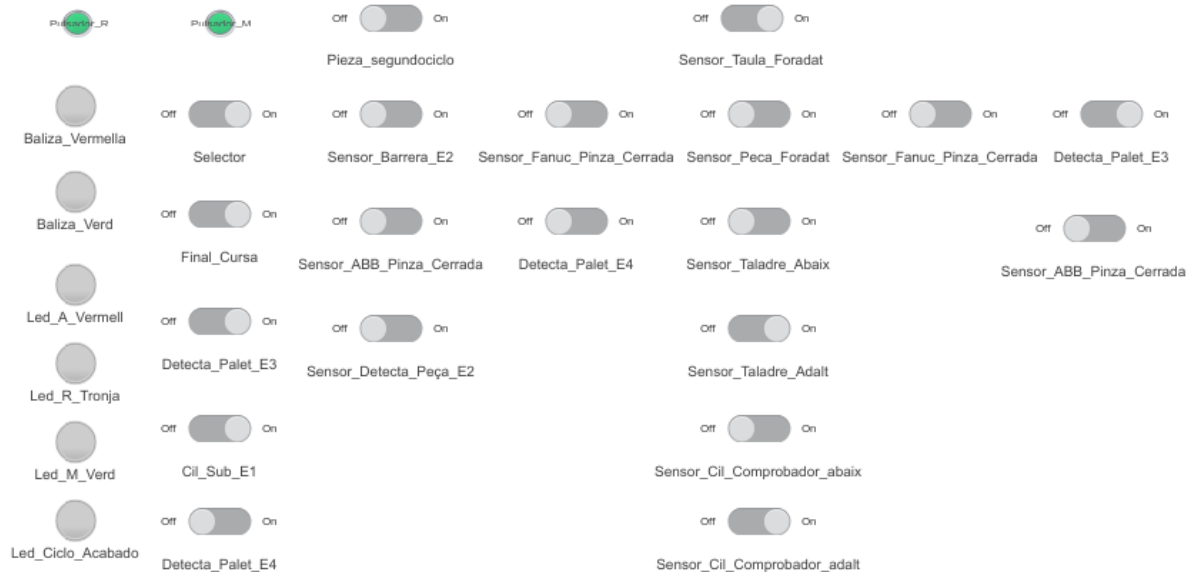


Il·lustració 49.Scada de simulació gràfic (font: Pròpia)

Tal i com es pot observar en la il·lustració 49 és pot veure gràficament l'actuació dels sensors i els actuadors de la cèl·lula , com podria ser els motors , les electrovàlvules , quan el robot te la pinça tancada i esta transportant la pinça entre d'altres per tal de poder seguir gràficament la modelització del sistema.

Quan els indicadors dels actuadors implementats estan en funcionament es posen de color verd i quan estan aturats es posen de color vermell. Per tal de poder simular el colors dels actuadors que es troben en l'scada de simulació es controla amb els elements outputs que estan enllaçats amb els indicadors(il·lustració 51 i 52).

El segon apartat conte el panel de control de simulació dels diferents sensors i actuadors que s'han de simular ,per tal de recrear el comportament de la cèl·lula.



Il·lustració 50.Panel de control de simulació (font: Pròpia)

Dintres del panel de control tal i com es pot observar , conte 2 pulsadors un de marxa per inicialitzar el procés de la maquina i un altre de rearmament per tal de rearmar les seguretats de la cèl·lula .

Tal i com es pot veure en la part de l'esquerra de la il·lustració 50 es troba els actuadors visuals , balisa vermella, balisa verd , LED de CC verd ,LED de CC vermell i LED de CC groc .Principalment hi ha un selector que simula quan es dona tensió al quadre , seguidament s'encén la llum de la balisa vermella i la llum del LED de CC vermell , quan es prem al pulsador de rearmament s'atura la llum de LED de CC vermell i s'encén la taronja que marca que esta esperen la marxa del process .

Quan es prem el pulsador de marxa s'atura la balisa vermella i el LED CC taronja i s'encén la balisa verd i el LED CC verd per identificar que la cèl·lula esta en funcionament.

Seguidament consta dels sensors que trobem a la cèl·lula i del feedback de robots per simular una comunicació amb el PLC. Com podem veure final de cursa, sensor detecta palet E3, sensor cilindre subministrament per detectar quan ha subministrat una peça, sensor detecta palet E4, sensor òptic de barrera per detectar que hi ha alguna peça en aquella estació, sensor detecta peça

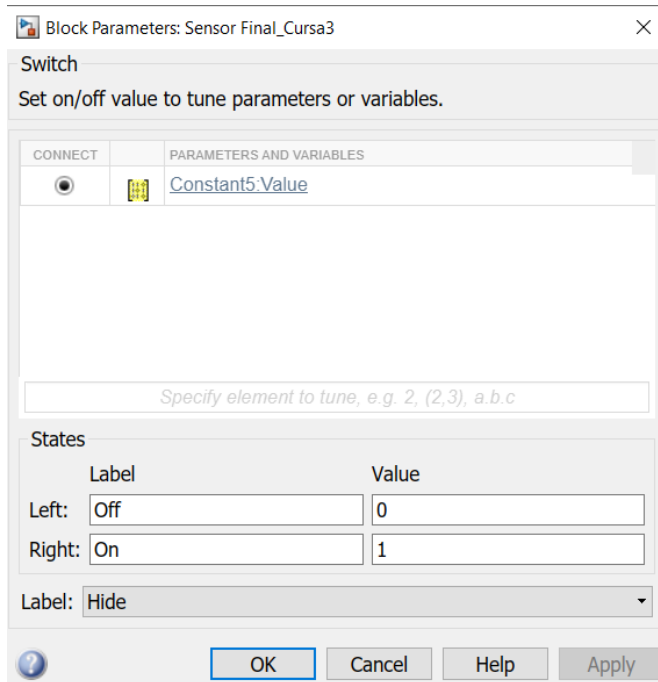
E2 que es un sensor inductiu per detectar les peces metàl·liques , els feedbacks dels robots ABB i FANUC de pinça tancada , el sensor de taula foradat per determinar la posició de la peça dintre de la taula amb la utilització de flancs de pujada finalment sensors de taladre i cilindre comprovador a dalt i abaix



Il·lustració 51.Simulació dels actuadors gràfics (font: Pròpia)

Com es pot veure en la il·lustració 51, es veu que el indicador gràfic esta connectat al senyal OR4.1 que pertany a la sortida apropiada que es vol simular. També es parametriza l'estat de la sortida perquè canvi el seu color, quan la sortia OR4.1 esta activada el estat es 1 llavors el es posa de color verd i quan es 0 esta aturat i es posa de color vermell.

Com que es una simulació i no es te connectat cap entrada s'han de simular, de tal forma que es simula amb polsadors i interruptors que es troben interconnectats amb les constants que es troben en els blocs de programació.



Il·lustració 52. Simulació dels sensors (font: Pròpia)

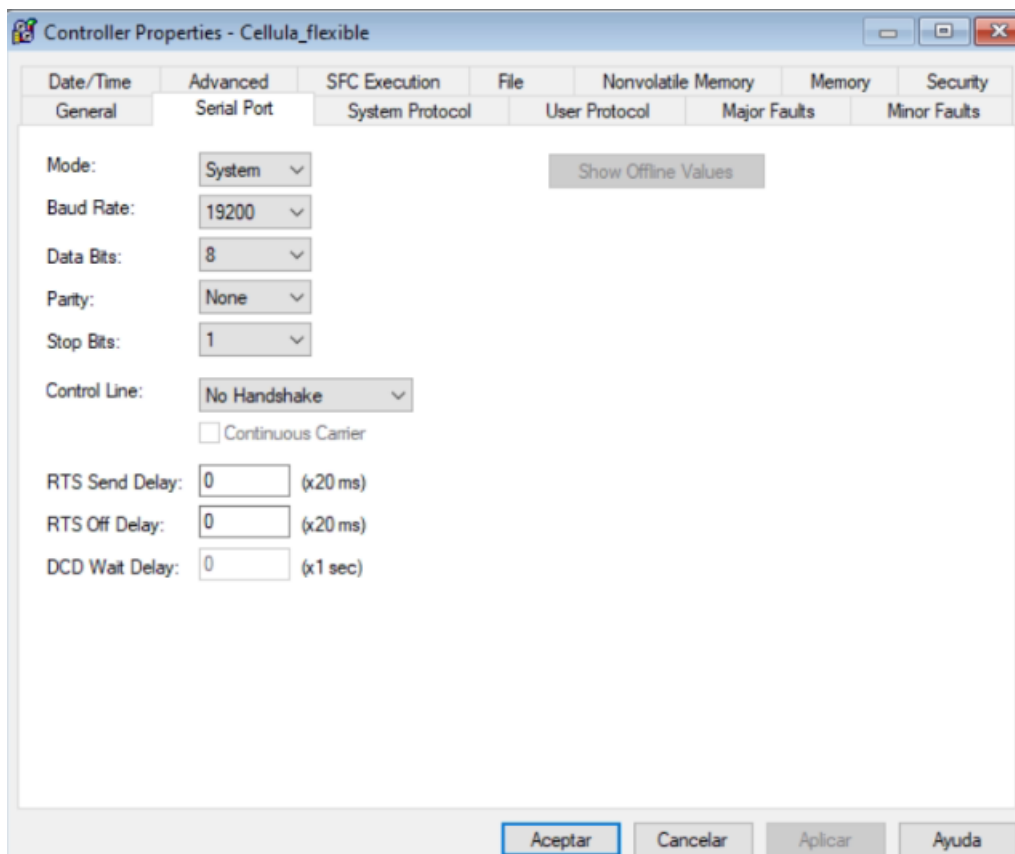
En la il·lustració 52, es pot veure que la simulació dels sensors es connecta directament amb la constant que es simula en el bloc de programació, en aquest cas es la constant5. El control de l'estat del interruptor quan esta a la esquerra el sensor no esta detectant i quan esta a la dreta es quan el sensor detecta.

3.2.3. Programació d'autòmat Rockwell

3.2.3.1. Configuració de hardware i comunicació

Principalment configurarem el protocol de comunicació via ethernet dintre del programa Rslogix5000.

Configuració del port serial



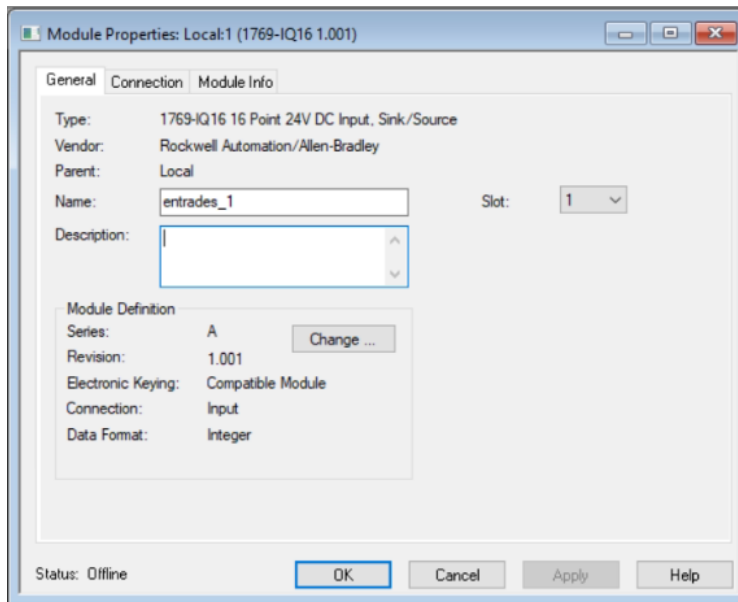
Il·lustració 53. Configuració tarja de comunicació ethernet (font: Pròpia)

Per la configuració de comunicació s'ha deixat tots els valors generats de fabrica, utilitzant una comunicació via ethernet.

Una vegada realitzat el protocol de comunicació, es configura les targes de entrades i sortides que tenim físicament dintre del programa Rslogix5000

Configuració de les targetes d'entrada del PLC y de sortida

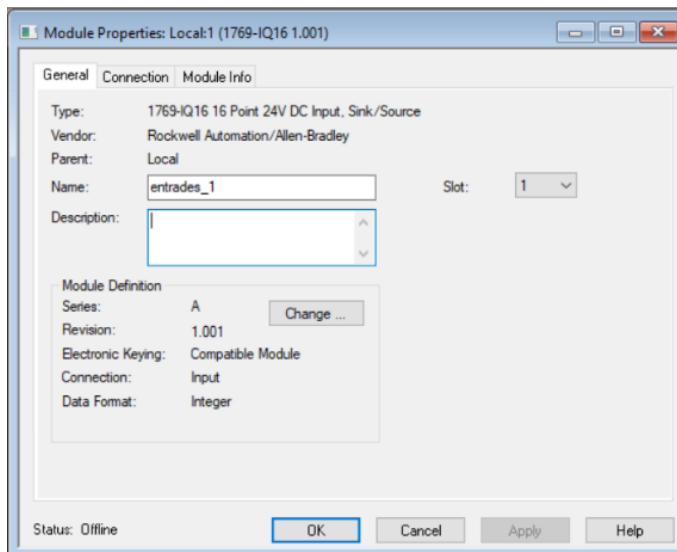
ENTRADA-1



Il·lustració 54. Configuració del mòdul d'entrades1 (font: Pròpia)

S'assigna la tarja del PLC 1769-IQ 16, en la qual es troba totes les botoneres i els sensors de procés

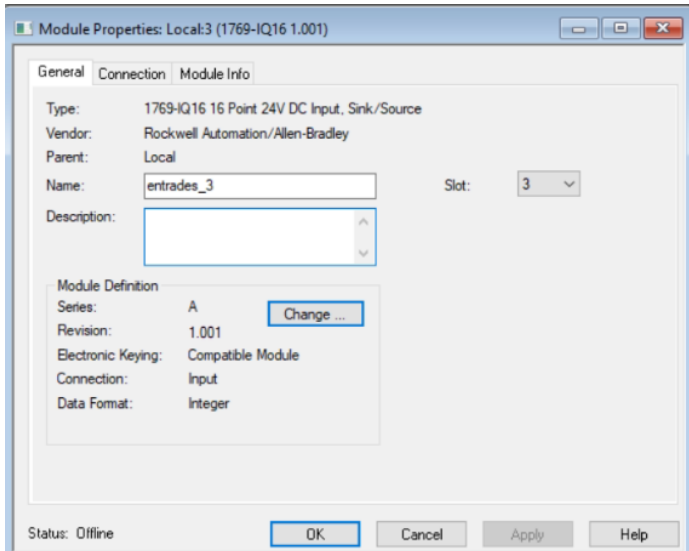
ENTRADA-2



Il·lustració 55. Configuració del mòdul d'entrades2 (font: Pròpia)

S'assigna la tarja del PLC 1769-IQ 16, en la qual es troba sensors de procés i altres canals lliures per futures ampliacions

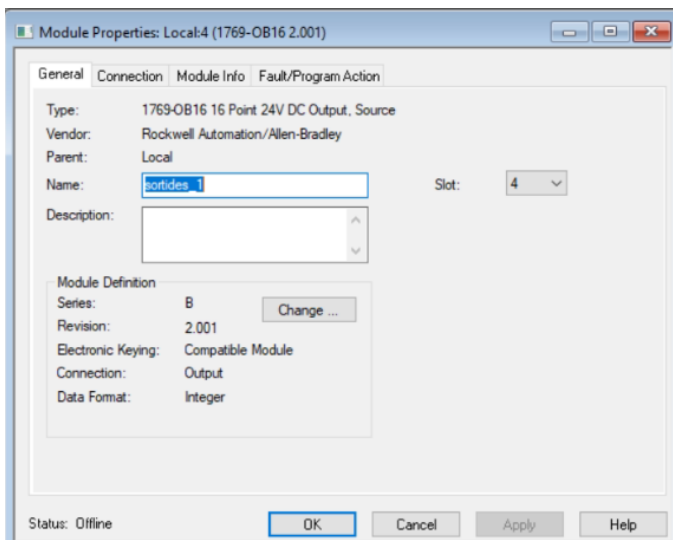
ENTRADA-3



Il·lustració 56. Configuració del mòdul d'entrades3 (font: Pròpia)

S'assigna la tarja del PLC 1769-IQ 16, en la qual es troba les sortides d'actuació del robot fanuc.

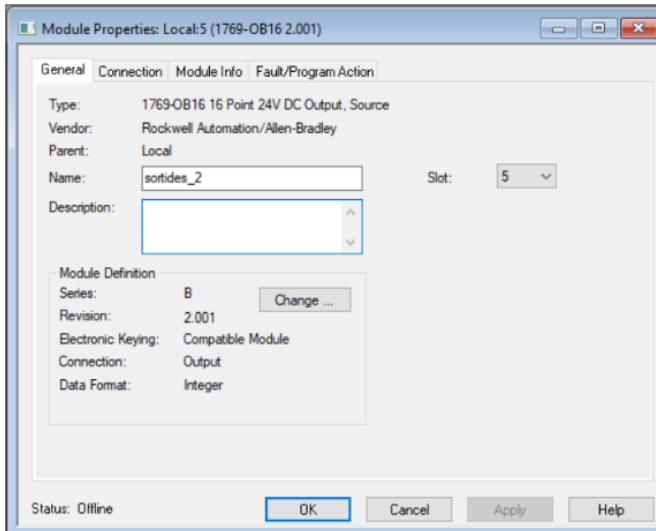
SORTIDES-1



Il·lustració 57. Configuració del mòdul de sortides1 (font: Pròpia)

S'assigna la tarja del PLC 1769-OB 16, en la qual es troba els indicadors i els actuadors de la cèl·lula

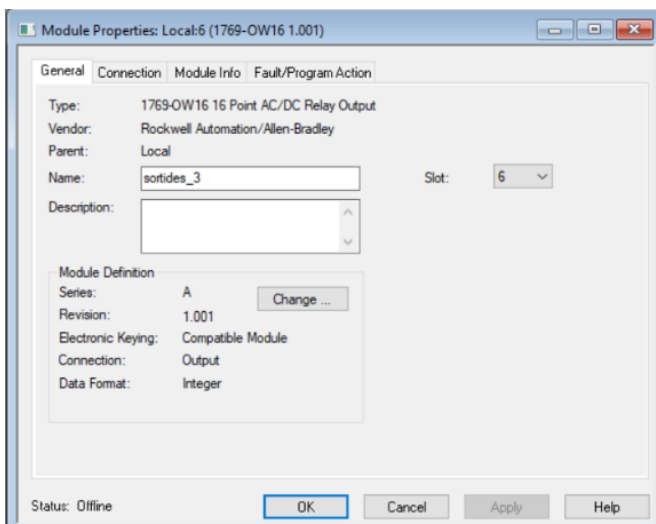
SORTIDES-2



Il·lustració 58. Configuració del mòdul de sortides2 (font: Pròpia)

S'assigna la tarja del PLC 1769-OB 16, en la qual es troba els actuadors de la estació de foradat amb canals lliures per futures ampliacions.

SORTIDES-3



Il·lustració 59. Configuració del mòdul de sortides3 (font: Pròpia)

S'assigna la tarja del PLC 1769-OB 16, en la qual es troba els indicadors del robot fanuc

3.2.3.2. Declaració de E/S

Per la declaració de les entrades i sortides de programa s'han separat en diferents targes del PLC segons el procés desitjat.

Entrades-1	Funció
I1.0	Boto Marxa
I1.1	Boto Aturada
I1.2	Boto Rearmamant
I1.3	Sensor Cil subm sortit
I1.4	Sensor Cil subm entrat
I1.5	Sensor de barrera Optic
I1.6	Sensor_Capacitiu_Deteccio_Barrera
I1.7	Sensor detecta peça E2
I1.8	Sensor detecta palet E3
I1.9	Fi de cursa E_subm
I1.10	Seta_Emergencia
I1.11	Output 4 del ABB/ ABB_PREPARAT/Detecta_contenedor/Detecta_Palet
I1.12	Output 2 del ABB/ ABB_detecta_subministrador
I1.13	Output 3 del ABB/ABB_palet_Magatzem
I1.14	Sensor Peça foradat
I1.15	Sensor Gir Taula

Taula 5. Asignació d'entrades digitals de la tarja 1

Dintres de la taula de E/S en la tarja d'entrades-1, tenim declarat tot els sensors de procés de la cèl·lula i el control de les accions del robot ABB.

Entrades-2	Funció
I2.0	Sensor taladre entrat
I2.1	Sensor taladre sortit
I2.2	Sensor Subjectador Entrat
I2.3	Sensor Subjectador Sortit
I2.4	Sensor verifica forat sortit
I2.5	Sensor verifica forat entrat
I2.6	Free
I2.7	Free
I2.8	Sensor detecta palet E4_foradat
I2.9	Free
I2.10	Free
I2.11	Free
I2.12	Free
I2.13	Free
I2.14	Free
I2.15	Free

Taula 6. Asignació d'entrades digitals de la tarja 2

Dintres de la taula de E/S en la tarja d'entrades-2, tenim declarat els sensors de l'estació de foradat.

Entrades-3	Funció
I3.0	Output 1 Fanuc/ Preparat
I3.1	Output 2 Fanuc/ Palet_E4
I3.2	Output 3 Fanuc/ Foradat contenidor
I3.3	Output 4 Fanuc/Foradadora_a_palet
I3.4	Free
I3.5	Free
I3.6	Free
I3.7	Free
I3.8	Free
I3.9	Free
I3.10	Free
I3.11	Free
I3.12	Free
I3.13	Free
I3.14	Free
I3.15	Free

Taula 7.Asignació d'entrades digitals de la tarja 3

Dintres de la taula de E/S en la tarja d'entrades-3, tenim declarat les accions de l'estació de foradat.

Sortides-1	Funció
Q4.0	EV_ Cilindre subministrador
Q4.1	INPUT 1 DEL ABB/PREPARAT
Q4.2	Input 2 del ABB/Abb_Dispensador_Detecció
Q4.3	Input 3 del ABB/Detecció Contenedor
Q4.4	LED VERD MARCHA
Q4.5	LED VERMELL ATUR
Q4.6	LED GROC REARME
Q4.7	Motors Cinta
Q4.8	Input 4 ABB/Detecta palet_E3
Q4.9	EV_Pisto_E3
Q4.10	Free
Q4.11	Input 6 ABB /Palet Magatzem
Q4.12	Input 7 del ABB
Q4.13	Input 8 del ABB
Q4.14	Input 9 del ABB
Q4.15	Motor Plat Giratori

Taula 8..Asignació de sortides digitals de la tarja 1

Dintres de la taula de E/S en la tarja de sortides-1, tenim declarat els actuadors de l'estació i els indicadors del moviment del robot ABB.

Sortides-2	Funció
Q5.0	Gir_horari_Trepant
Q5.1	Gir_Antihorari_Trepant
Q5.2	Ev_Trepant_Puja
Q5.3	EV_Trepant_baixa
Q5.4	Ev_subjectador_Taula
Q5.5	Ev_Verificador_Foradat
Q5.6	Free
Q5.7	Free
Q5.8	Free
Q5.9	Free
Q5.10	Free
Q5.11	Free
Q5.12	Free
Q5.13	Free
Q5.14	Free
Q5.15	EV_Pisto_E4

Taula 9. Asignació de sortides digitals de la tarja 2

Dintres de la taula de E/S en la tarja de sortides-2, tenim declarat els actuadors de l'estació de foradat.

Sortides-3	Funció
Q6.0	Input 1 Fanuc/Preparat
Q6.1	Input 2 Fanuc/Palet_E4
Q6.2	Input 4 Fanuc/Plat contenedor
Q6.3	Input 5 Fanuc/Plat Palet
Q6.4	Free
Q6.5	Free
Q6.6	Free
Q6.7	Free
Q6.8	Free
Q6.9	Free
Q6.10	Free
Q6.11	Free
Q6.12	Free
Q6.13	Free
Q6.14	Free
Q6.15	Free

Taula 10. Asignació de sortides digitals de la tarja 3

Dintres de la taula de E/S en la tarja de sortides-3, tenim declarat els indicadors del robot FANUC.

3.2.3.3. Programació de PLC en ST

ESTACIÓ 1 ESTAT INICIAL I SUBMINISTRAMENT

```
//ESTACIO_1 INICIAL SUBMINISTRAMENT

IF S:FS THEN
  etapa0 := 1;
END_IF;
  IF ((etapa0) AND (contador_piezas.PRE >0)) THEN

    etapal := 1;
    etapa0 := 0;

    Led_A_Vermell := 1;
    Led_R_tronja := 0;
    Led_M_Verd := 0;
    motors_cinta := 0;
    EV_Cil_El := 0;

  END_IF;

  IF etapa1 AND Polsador_R THEN

    etapal := 0;
    etapa2 := 1;

    Led_A_Vermell := 0;
    Led_R_tronja := 1;
    Led_M_Verd := 0;
    motors_cinta := 0;
    EV_Cil_El := 0;

  END_IF;
```

Il·lustració 60. Estació de subministrament part 1 (font: Pròpia)

Per inicialitzar el programa primerament es manté a la espera fins que marques la quantitat de cicles que vols realitzar, una vegada establert s'encén la llum vermella del quadre de control que marca que la maquina esta aturada.

Seguidament s'ha de prémer el polsador de rearmament per tal de validar que ni hi ha cap perill per poder iniciar la marxa.

```

//CUANDO HAY UNA PIEZA DEFECTUOSA LA ENVIA A LA BASURA Y VUELVE A PROPORCIONAR UNA NUEVA
IF (((Pulsador_M ) AND (Cil_El_Entrat)) AND (etapa2)) OR ((etapa9_Reposicio) AND (Cil_El_Entrat)) OR(etapa1)) THEN

    etapa3 := 1;
    etapa2 := 0;
    etapa9_Reposicio:=0;
    etapa1:=0;

    Led_A_Vermell :=0;
    Led_R_tronja := 0;
    Led_M_Verd := 1;
    motora_cinta := 1;
    EV_Cil_El := 1;

END_IF;

```

Il·lustració 61.Estació de subministrament part2 (font: Pròpia)

Una vegada rearmat el sistema , espera que es tingui les condicions inicials de per tal de poder inicialitzar la marxa , que es premi el pulsador de marxa , el cilindre de subministrament estigui en entrat i prèviament s'hagi rearmat el sistema.

ROBOT ABB ESTACIÓ SUBMINISTRAMENT A ESTACIÓ DE DETECCIÓ TIPUS DE PEÇA

```

//PONGO UN NEGADO DEL FINAL DE CARRERA , PORQUE ES UN CONTACTO CERRADO ASI QUE CUANDO ACTUA ES SE ABRE EL CONTACTO LOGICA INVERSA
IF (((Cil_El_Sortit) AND NOT(Final_Cursa )) AND (Detecta_Palet_ES)) AND (etapa3)) THEN

    etapa4 := 1;
    etapa3 := 0;

    Led_A_Vermell :=0;
    Led_R_tronja := 0;
    Led_M_Verd := 1;
    motora_cinta := 0;
    EV_Cil_El := 0;
    Output_ABB_Freparat := 1;

END_IF;

```

Il·lustració 62.Moviment del robot ABB de estació de subministrament a estació de detecció tipus de peça (font: Pròpia)

Seguidament s'envia un feedback cap a el robot ABB confirmant que la peça ja s'ha subministrat i esta llesta per ser traslladada des de la estació de subministrament a l'estació de detecció tipus de peça.

ESTACIÓ 2 DETECCIÓ TIPUS DE PEÇA

```
//ESTACIO_2 DETECCIÓ TIPUS DE PEÇA
// REVISAR LOS SENSORES HAY UNO DE ELLOS QUE NO TENDRIA QUE SER INDUCTIVO.

IF Sensor_Barrera and etapa6 THEN

    etapa6:= 0;
    etapa7 := 1;

END_IF;
```

Il·lustració 63. Estació de subministrament (font :Pròpia)

Quan el robot ABB deixa la peça en la estació de detecció de peça s'utilitza un sensor òptic de barrera per tal de verificar que ha arriat una peça

ROBOT ABB ESTACIÓ 2 A PLAT PRIMERA PEÇA

```
IF Sensor_Detecta_Peca_E2 AND etapa7 THEN

    etapa7 := 0;
    etapa8 := 1;
    Output_ABB_Detecta_Palet_E3 := 1;

END_IF;
```

Il·lustració 64.Moviment del robot ABB de estació de detecció tipus de peça a palet

Una vegada detectat que ha arribat una peça , si detecta que la peça es metàl·lica envia un feedback cap al robot ABB confirmant que la peça es correcte ,per tal de ser traslladada des de la estació de detecció tipus de peça fins el palet

```
IF (((etapa7) AND NOT ((Sensor_Detecta_Peca_E2)) AND (Sensor_Barrera))) THEN

    etapa7 := 0;
    etapa8_Pieza_mala := 1;

    mov_2_ABB_deteccio_contenedor := 1;

    Output_ABB_Detecta_Palet_E3 := 0;
END_IF;
```

Il·lustració 65..Moviment del robot ABB de estació de detecció tipus de peça a contenidor

Una vegada detectat que ha arribat una peça , si detecta que la peça es de plàstic envia un feedback cap al robot ABB confirmant que la peça es incorrecte ,per tal de ser traslladada des de la estació de detecció tipus de peça fins a la cubeta de descartades.

ROBOT DE ABB ESTACIÓ1 SUBMINISTRAMENT A ESTACIÓ2 DETECTA PEÇA

```
//ABB_ESTACIO2_A_PALET PIEZA 2

IF ((etapa9_1) OR (etapa9_7_Reposicio)) THEN

    etapa9_1 :=0;
    etapa9_2:=1;
    etapa9_7_Reposicio:=0;

    Led_A_Vermell :=0;
    Led_R_tronja := 0;
    Led_M_Verd := 1;
    motors_cinta := 1;
    EV_Cil_El := 1;

END_IF;

IF (((Cil_El_Sortit) AND NOT(Final_Cursa )) AND (Detecta_Palet_E3)) AND (etapa9_2)) THEN

    etapa9_2 :=0;
    etapa9_3 :=1;

    Led_A_Vermell :=0;
    Led_R_tronja := 0;
    Led_M_Verd := 1;
    motors_cinta := 0;
    EV_Cil_El := 0;
    Output_ABB_Preparat := 1;

END_IF;
```

Il·lustració 66.Moviment del robot ABB de estació de detecció tipus de peça a palet segona peça (font: Pròpia)

En aquesta part del programa s'utilitza per 2 coses per un bucle infinit si les peces son de plàstic mitjançant la variable (etapa9_7_Repetició) i l'altre per tal de fer el control de la segona peça metàl·lica utilitzant la variable (etapa9_1).Indistintament de qualsevol de les dues variables quan es compli les condicions de peça subministrada , li envia un feedback al robot ABB per portar el robot a posició de repos

```
IF ABB_Preparat AND etapa9_3 THEN

    etapa9_3 :=0;
    etapa9_4 :=1;
    Output_ABB_Preparat := 0;
    Output_ABB_Subministrament := 1;

END_IF;

IF ABB_Subministrament and etapa9_4 THEN

    etapa9_4:=0;
    etapa9_5:=1;
    Output_ABB_Subministrament := 0;

END_IF;

IF Sensor_Barrera and etapa9_5 THEN

    etapa9_5 :=0;
    etapa9_6 :=1;

END_IF;

    IF Sensor_Detecta_Peca_E2 AND etapa9_6 THEN

        etapa9_6 := 0;
        etapa9_7 := 1;
        Output_ABB_Detecta_Palet_E3 := 1;

    END_IF;
```

Il·lustració 67.Moviment del robot ABB de estació de detecció tipus de peça a palet segona peça part2 (font: Pròpia)

Quan el robot envia el feedback cap a el PLC de que esta en posició de repòs , el PLC li envia un altre feedback al robot ABB confirmant que esta llest per transportar la peça des de l'estació de subministrament fins l'estació de detecta peça.

Seguidament el robot envia un feedback al PLC de que la peça esta en la estació de detecció tipus de peça, quan detecta amb el sensor òptic de barrera que hi ha una peça, es verifica que la peça es metàl·lica , quan detecta que es correcte li envia un feedback al robot ABB confirmant-ho per tal de traslladar la peça des de la estació de detecció tipus de peça fins al palet.

```

IF (((etapa9_6) AND NOT ((Sensor_Detecta_Peca_E2)) AND (Sensor_Barrera))) THEN

    etapa9_6 := 0;
    etapa9_6_Pieza_mala := 1;

    mov_2_ABB_deteccio_contenedor := 1;
    Output_ABB_Detecta_Palet_E3 := 0;

END_IF;

IF ((etapa9_7) AND (ABB_Detecta_Palet_E3)) THEN

    etapa9_7 :=0;
    etapa9 := 1;
    Output_ABB_Detecta_Palet_E3 := 0;

    //PONER CONTADOR PARA CONTROLAR LAS 2 PIEZAS

END_IF;

IF etapa9_6_Pieza_mala AND DN_mov_2_ABB_deteccio_contenedor THEN

    etapa9_6_Pieza_mala :=0;
    etapa9_7_Reposicio :=1;
    mov_2_ABB_deteccio_contenedor := 0;
    Output_ABB_Detecta_Palet_E3 := 0;
END_IF;

```

Il·lustració 68.Moviment del robot ABB de estació de detecció tipus de peça a cubeta peces descartades(font: Pròpia)

Quan el sensor òptic de barrera detecta que hi ha peça ,però amb el sensor inductiu detecta que no es metàl·lica envia un feedback al robot ABB confirmant de que la peça es de plàstic , per tal de transportar-la des de la estació de detecció tipus de peça fins la cubeta de peces descartades.

PALET FINS A ESTACIÓ 3 FORADAT

```

//ESTACIO_3

IF etapa9 THEN

    etapa9:= 0;
    etapal0:= 1;
    motors_cinta := 1;
    EV_Pisto_E3 := 1;

END_IF;

IF etapal0 AND Sensor_Detecta_Palet_E4 THEN

    etapal0 := 0;
    etapal1 := 1;
    motors_cinta := 0;
    EV_Pisto_E3 := 0;

END_IF;

```

Il·lustració 69.Estació moviment de palet cinta fins a estació foradat (font: Pròpia)

Quan en el palet ja hi ha 2 peces , s'inicialitza el motor de les cintes per transportar el palet fins a l'estació de foradat.

FANUC ESTACIO3 A CINTA FORADAT

```
//FANUC_ESTACIO3_CINTA_FORADAT

IF etapal1 THEN

    etapal1 :=0;
    etapal2 := 1;
    mov_0_FANUC_preparat:= 1;

END_IF;

IF DN_mov_0_FANUC_preparat AND etapal2 THEN

    etapal2:= 0;
    etapal3:= 1;
    mov_0_FANUC_preparat:= 0;
    mov_1_FANUC_palet_plat:= 1;

END_IF;

IF etapal3 AND DN_mov_1_FANUC_palet_plat THEN

    etapal3 := 0;
    etapal4 := 1;
    mov_1_FANUC_palet_plat:= 0;

END_IF;
```

Il·lustració 70.Moviment del robot FANUC de palet a estació de foradat (font: Pròpia

Quan el palet es situat en el lloc desitjat de l'estació 3 , li envia un feedback de confirmació al robot FANUC per tal de traslladar la peça des de el palet fins la cinta de foradat.

ESTACIÓ FORADAT

```
//ESTACIO_FORADAT

IF etapa14 THEN
    etapa14:=0;
    etapa15:=1;
    motor_plat_giratori:=1;

    END_IF;

IF sensor_inductiu_plat AND etapa15 THEN
    etapa15:=0;
    etapa17:=1;
    motor_plat_giratori:=1;
END_IF;

IF NOT sensor_inductiu_plat AND etapa17 THEN
    etapa17:=0;
    etapa17_2:=1;
    motor_plat_giratori:=1;
END_IF;

IF sensor_inductiu_plat AND etapa17_2 THEN
    etapa17_2:=0;
    etapa17_3:=1;
    motor_plat_giratori:=1;
END_IF;

IF NOT sensor_inductiu_plat AND etapa17_3 THEN
    etapa17_3:=0;
    etapa17_4:=1;
    motor_plat_giratori:=1;
END_IF;
```

Il·lustració 71. Estació de foradat gira taula (font: Pròpia)

Una vegada es rep el feedback de que el robot FANUC ha deixat la peça en la cinta de foradat, comença a girar fent 2 flanc de pujada , per tal de fer moure la taula fins la posició del taladre.

```
IF sensor_inductiu_plat AND etapa17_4 THEN
    etapa17_4:=0;
    etapa17_5:=1;
    motor_plat_giratori:=0;
    EV_subjectador :=1;
END_IF;

IF FC_cilindre_subjectador_anterior AND etapa17_5 THEN
    etapa17_5:=0;
    etapa18:=1;
    motor_plat_giratori:=0;
    EV_subjectador :=1;
END_IF;

IF etapa18 THEN
    etapa18:=0;
    etapa19:=1;
    EV_trepant_baixar:=1;
    gir_horari_trepant:=1;
    EV_subjectador :=1;
END_IF;

IF etapa19 AND FC_cilindre_trepant_inferior THEN
    etapa19:=0;
    etapa20:=1;
    EV_trepant_baixar:=0;
    gir_horari_trepant:=0;
    EV_trepant_pujar:=1;
    gir_antihorari_trepant:=1;
    EV_subjectador :=1;
END IF;
```

Il·lustració 72. Estació de foradat subjecta peça i forada (font: Pròpia)

Seguidament quan la taula ha mogut la peça fins el lloc on es troba el taladre, s'activa el pistó pneumàtic per subjectar la peça per tal de evitar que es mogui a l'hora de fer el forat, amb la peça subjectada s'inicia el motor del trepant i comença a girar en sentit horari per tal de poder foradar-la juntament amb l'actuació de l'EV per fer baixar el trepant.

Quan el sensor detecta que el trepant esta a la distancia del forat desitjat , fa girar el motor en sentit anti-horari per tal de no seguir foradant i s'activa la EV que fa pujar el trepant.

```
IF etapa20 AND FC_cilindre_trepant_superior THEN
  etapa20:=0;
  etapa21:=1;
  EV_trepant_baixar:=0;
  gir_horari_trepant:=0;
  EV_trepant_pujar:=0;
  gir_antihorari_trepant:=0;
  motor_plat_giratori:=1;
  EV_subjectador :=0;
END_IF;

IF sensor_inductiu_plat AND etapa21 THEN
  etapa21:=0;
  etapa21_2:=1;
  motor_plat_giratori:=1;
END_IF;

IF NOT sensor_inductiu_plat AND etapa21_2 THEN
  etapa21_2:=0;
  etapa22:=1;
  motor_plat_giratori:=1;
END_IF;

IF sensor_inductiu_plat AND etapa22 THEN
  etapa22:=0;
  etapa23:=1;
  motor_plat_giratori:=0;
END_IF;
```

Il·lustració 73. Estació de foradat gira taula fins zona de verificació (font: Pròpia)

Quan detecta que el trepant a pujat a la posició de repòs , deixa de subjectar la peça i fa un flanc de pujada del sensor del plat per tal d'anar a la posició de verificació.


```

IF etapa23 THEN
  etapa23:=0;
  etapa24:=1;
  motor_plat_giratori:=0;
  EV_verificador:=1;
END_IF;

IF FC_cilindre_verificador_inferior AND etapa24 THEN
  etapa24:=0;
  etapa25:=1;
  motor_plat_giratori:=0;
  EV_verificador:=0;
END_IF;

IF FC_cilindre_verificador_superior AND etapa25 THEN
  etapa25:=0;
  etapa25_2:=1;
  motor_plat_giratori:=1;
  EV_verificador:=0;
END_IF;

IF NOT sensor_inductiu_plat AND etapa25_2 THEN
  etapa25_2:=0;
  etapa25_3:=1;
  motor_plat_giratori:=1;
END_IF;

IF sensor_inductiu_plat AND etapa25_3 THEN
  etapa25_3:=0;
  etapa26:=1;
  motor_plat_giratori:=0;
END_IF;

```

Il·lustració 74. Estació de foradat, verifica el forat (font: Pròpia)

Seguidament quan esta a la posició de verificació , s'activa una EV per fer baixar el pistó de verificació si el forat es correcte perquè detecta que el pistó es en la posició inferior , deixa d'actuar sobre la EV de simple efecte de retorn per molla per tornar a posició inicial, una vegada en posició inicial es fa un flanc de pujada per moure la peça fins la posició de peça correcta.

FANUC DE FORADAT A PALET

```

//FANUC_FORADAT_A_PALET

  IF etapa26 THEN
    etapa26:=0;
    etapa27:=1;
    mov_3_FANUC_plat_palet:=1;
  END_IF;

  IF DN_mov_3_FANUC_plat_palet AND etapa27 THEN
    etapa27:=0;
    etapa29_1:=1;
    motors_cinta:=0;
    EV_enclavament_palet_mecanitzat:=0
;
    mov_3_FANUC_plat_palet:=0;
  END_IF;

```

Il·lustració 75. Moviment de robot Fanuc de estació de foradat a palet correcte foradat

Una vegada s'ha verificat que el forat s'ha fet correctament i la peça es troba en la posició correcta , li envia un feedback al robot FANUC confirmant que esta preparat per tal de traslladar la peça de la taula de foradat al palet.

FANUC ESTACIO3 A CINTA FORADAT SEGONA PEÇA

```
//ESTACIÓ3_A_ESTACIO_FORADAT_SEGONA_PEÇA

IF etapa29_1 THEN

    etapa29_1 :=0;
    etapal2_2 := 1;
    mov_0_FANUC_preparat:= 1;

END_IF;

IF DN_mov_0_FANUC_preparat AND etapal2_2 THEN

    etapal2_2:= 0;
    etapal3_2:= 1;
    mov_0_FANUC_preparat:= 0;
    mov_1_FANUC_palet_plat:= 1;

END_IF;

IF etapal3_2 AND DN_mov_1_FANUC_palet_plat THEN

    etapal3_2 := 0;
    etapal4_2 := 1;
    mov_1_FANUC_palet_plat:= 0;

END_IF;
```

Il·lustració 76.Moviment del robot FANUC de palet a estació de foradat segona peça (font: Pròpia)

Una vegada finalitzat el foradat i el retorn de la peça al palet , s'envia el feedback cap a el robot ABB per tal de tornar el robot FANUC a reposició, quan el robot FANUC arriba a reposició envia un feedback cap a el PLC confirmant la seva posició, seguidament s'envia el feedback cap a el robot FANUC per tal de traslladar la segona peça del palet cap a la taula de foradat.

ESTACIÓ DE FORADAT SEGONA PEÇA

```

//ESTACIO_FORADAT SEGONA PEÇA

IF etapa14_2 THEN
    etapa14_2:=0;
    etapa15_2:=1;
    motor_plat_giratori:=1;

    END_IF;

IF sensor_inductiu_plat AND etapa15_2 THEN
    etapa15_2:=0;
    etapa17_22:=1;
    motor_plat_giratori:=1;
    END_IF;

IF NOT sensor_inductiu_plat AND etapa17_22 THEN
    etapa17_22:=0;
    etapa17_2_22:=1;
    motor_plat_giratori:=1;
    END_IF;

IF sensor_inductiu_plat AND etapa17_2_22 THEN
    etapa17_2_22:=0;
    etapa17_3_22:=1;
    motor_plat_giratori:=1;
    END_IF;

IF NOT sensor_inductiu_plat AND etapa17_3_22 THEN
    etapa17_3_22:=0;
    etapa17_4_22:=1;
    motor_plat_giratori:=1;
    END_IF;

```

Il·lustració 77. Estació de foradat gira taula segona peça (font :Pròpia)

Una vegada es rep el feedback de que el robot FANUC ha deixat la peça en la cinta de foradat, comença a girar fent 2 flanc de pujada , per tal de fer moure la taula fins la posició del taladre.

```

IF sensor_inductiu_plat AND etapa17_4_22 THEN
    etapa17_4_22:=0;
    etapa17_5_22:=1;
    motor_plat_giratori:=0;
    EV_subjectador :=1;
    END_IF;

IF FC_cilindre_subjectador_anterior AND etapa17_5_22 THEN
    etapa17_5_22:=0;
    etapa18_22:=1;
    motor_plat_giratori:=0;
    EV_subjectador :=1;

    END_IF;

IF etapa18_22 THEN
    etapa18_22:=0;
    etapa19_22:=1;
    EV_trepant_baixar:=1;
    gir_horari_trepant:=1;
    EV_subjectador :=1;

    END_IF;

IF etapa19_22 AND FC_cilindre_trepant_inferior THEN
    etapa19_22:=0;
    etapa20_22:=1;
    EV_trepant_baixar:=0;
    gir_horari_trepant:=0;
    EV_trepant_pujar:=1;
    gir_antihorari_trepant:=1;
    EV_subjectador :=1;
    END IF;

```

Il·lustració 78. Estació de foradat subjecta peça i forada segona peça (font: Pròpia)

Seguidament quan la taula ha mogut la peça fins el lloc on es troba el taladre, s'activa el pistó pneumàtic per subjectar la peça per tal de evitar que es mogui a l'hora de fer el forat, amb la peça subjectada s'inicia el motor del trepant i comença a girar en sentit horari per tal de poder foradar-la juntament amb l'actuació de l'EV per fer baixar el trepant.

Quan el sensor detecta que el trepant esta a la distancia del forat desitjat , fa girar el motor en sentit anti-horari per tal de no seguir foradant i s'activa la EV que fa pujar el trepant.

```
IF etapa20_22 AND FC_cilindre_trepant_superior THEN
  etapa20_22:=0;
  etapa21_22:=1;
  EV_trepant_baixar:=0;
  gir_horari_trepant:=0;
  EV_trepant_pujar:=0;
  gir_antihorari_trepant:=0;
  motor_plat_giratori:=1;
  EV_subjectador :=0;
END_IF;

IF sensor_inductiu_plat AND etapa21_22 THEN
  etapa21_22:=0;
  etapa21_2_22:=1;
  motor_plat_giratori:=1;
END_IF;

IF NOT sensor_inductiu_plat AND etapa21_2_22 THEN
  etapa21_2_22:=0;
  etapa22_22:=1;
  motor_plat_giratori:=1;
END_IF;

IF sensor_inductiu_plat AND etapa22_22 THEN
  etapa22_22:=0;
  etapa23_22:=1;
  motor_plat_giratori:=0;
END_IF;
```

Il·lustració 79. Estació de foradat gira taula fins zona de verificació segona peça(font: Pròpia)

Quan detecta que el trepant a pujat a la posició de repòs , deixa de subjectar la peça i fa un flanc de pujada del sensor del plat per tal d'anar a la posició de verificació.

```
IF etapa23_22 THEN
  etapa23_22:=0;
  etapa24_22:=1;
  motor_plat_giratori:=0;
  EV_verificador:=1;
END_IF;

IF FC_cilindre_verificador_inferior AND etapa24_22 THEN
  etapa24_22:=0;
  etapa25_22:=1;
  motor_plat_giratori:=0;
  EV_verificador:=0;
END_IF;

IF FC_cilindre_verificador_superior AND etapa25_22 THEN
  etapa25_22:=0;
  etapa25_2_22:=1;
  motor_plat_giratori:=1;
  EV_verificador:=0;
END_IF;

IF NOT sensor_inductiu_plat AND etapa25_2_22 THEN
  etapa25_2_22:=0;
  etapa25_3_22:=1;
  motor_plat_giratori:=1;
END_IF;

IF sensor_inductiu_plat AND etapa25_3_22 THEN
  etapa25_3_22:=0;
  etapa26_22:=1;
  motor_plat_giratori:=0;
END_IF;
```

Il·lustració 80. Estació de foradat, verifica el forat segona peça (font: Pròpia)

Seguidament quan esta a la posició de verificació , s'activa una EV per fer baixar el pistó de verificació si el forat es correcte perquè detecta que el pistó es en la posició inferior , deixa d'actuar sobre la EV de simple efecte de retorn per molla per tornar a posició inicial, una vegada en posició inicial es fa un flanc de pujada per moure la peça fins la posició de peça correcta.

FANUC DE FORADAT A PALET SEGONA PEÇA

```
//FANUC_FORADAT_A_ESTACIO4 SEGONA PEÇA

    IF etapa26_22 THEN
        etapa26_22:=0;
        etapa27_22:=1;
        mov_3_FANUC_plat_palet:=1;
    END_IF;

    IF DN_mov_3_FANUC_plat_palet AND etapa27_22 THEN
        etapa27_22:=0;
        etapa28_22:=1;
        motors_cinta:=1;
        EV_enclavament_palet_mecanitzat:=1;
        mov_3_FANUC_plat_palet:=0;
    END_IF;

    IF Detecta_Palet_E3 AND etapa28_22 THEN
        etapa28_22:=0;
        etapa29:=1;
        motors_cinta:=0;
        EV_enclavament_palet_mecanitzat:=0;
    END_IF;
```

Il·lustració 81. Moviment del robot FANUC de estació de foradat a palet segona peça y es mou palet fins a el destí desitjat (font: Pròpia)

Una vegada s'ha verificat que el forat s'ha fet correctament i la peça es troba en la posició correcte , li envia un feedback al robot FANUC confirmant que esta preparat per tal de traslladar la peça de la taula de foradat al palet, una vegada fa el moviment envia un feedback al PLC confirmant que el robot fanuc ha deixat la peça al palet, es desactiva el enclavament i comença a girar la cinta fins ha arribar a la posició desitjada per emmagatzemar.

ABB A MAGATZEM

```
//ABB_MAGATZEM

IF etapa29 THEN
    etapa29:=0;
    etapa30:=1;
    mov_4_ABB_palet_magatzem:=1;
END_IF;

IF DN_mov_4_ABB_palet_magatzem AND etapa30 THEN
    etapa30:=0;
    etapa29_2:=1;
    mov_4_ABB_palet_magatzem:=0;

END_IF;
```

Il·lustració 82.Moviment de robot ABB de palet a estació de magatzem (font: Pròpia)

Quan el palet ha arribat al lloc desitjat envia el feedback al robot ABB confirmant que esta llest per poder traslladar la peça del palet cap a el magatzem, una vegada ha emmagatzemat la peça envia un feedback al PLC confirmant que la peça s'ha emmagatzemat

ABB A MAGATZEM SEGONA PEÇA

```
//ABB_MAGATZEM_SEGONA_PEÇA

IF etapa29_2 AND NOT DN_mov_4_ABB_palet_magatzem THEN
    etapa29_2:=0;
    etapa30_2:=1;
    mov_4_ABB_palet_magatzem:=1;
END_IF;

IF DN_mov_4_ABB_palet_magatzem AND etapa30_2 THEN
    etapa30_2:=0;
    etapa31:=1;
    mov_4_ABB_palet_magatzem:=0;

END_IF;
```

Il·lustració 83.Moviment de robot ABB de palet a estació de magatzem segona peça(font: Pròpia)

Una vegada emmagatzemada la primera peça torna a fer el mateix procés però amb la segona , envia feedback al robot ABB confirmant que esta llest per poder traslladar la peça del palet cap a el magatzem, una vegada ha emmagatzemat la peça envia un feedback al PLC confirmant que la segona peça s'ha emmagatzemat,

Finalment per realitzar un altre cicle s'activa la etapa31 que fa que tornar el punter de programa a la (Il·lustració 61).

3.2.4 Programació robot ABB IRB 120

FUNCIÓ PRINCIPAL

```
31 | PROC main()
32 |
33 |     RESET doCierraPinze;
34 |     RESET DO10_4;
35 |     RESET DO10_3;
36 |     RESET DO10_2;
37 |
38 | IF DI10_1 =1 THEN
39 |
40 |     Path_INICIO;
41 |
42 | ENDIF
43 |
44 |
45 |
46 | IF DI10_2=1 THEN
47 |
48 |     SUBMINISTRADOR;
49 |
50 |
51 |
52 | ENDIF
53 |
54 | IF DI10_4=1 THEN IPEÇA CORRECTA DI10.4
55 |
56 |     DETECTA_PALET_E3;
57 |
58 |
59 | ENDIF
60 |
61 | IF DI10_6=1 THEN
62 |
63 |     Magatzem;
64 |
65 | ENDIF
66 |
67 | IF DI10_3 =1 THEN
68 |
69 |     CONTENIDOR;
70 |
71 | ENDIF
72 |
73 |
74 |
75 | ENDPROC
```

Il·lustració 84.Funció principal de programa Robot Studio (font: Pròpia)

El programa consta del programa principal, que s'encarrega de aturar totes les sortides del robot per tal de no generar cap problema amb la seqüència, seguidament es manté a la espera fins que rep algun feedback del PLC, dependent del valor executa una funció o una altra.

FUNCIÓ INICI


```

77 PROC Path_INICIO()
78
79 IREPOSO PIEZA SUBMINISTRADOR
80
81
82 MoveJ INICIO, v200, z50, tool0\WObj:=wobj0;
83 WaitTime 2;
84 Reset DO10_2;
85 Reset DO10_3;
86 SET DO10_4;
87 WaitTime 4;
88 ENDPROC

```

Il·lustració 85.Funció posició inici de programa Robot Studio (font: Pròpia)

Dintre de la funció de inici s'encarrega de reposicionar el robot, per tal d'esperar la següent funció desitjada.

FUNCIÓ SUBMINISTRAMENT

```

89 PROC SUBMINISTRADOR()
90
91
92
93 MoveJ [[269.83,285.5,373.63],[0.0159336,-0.835132,0.549754,-0.00842212],[0,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09]],v300,z50,tool0; IPIEZA SUBMINISTRADOR
94 MoveJ [[309.19,353.25,373.63],[0.0159243,-0.835135,0.54975,-0.00841567],[0,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09]],v300,z50,tool0; IPIEZA SUBMINISTRADOR
95 MoveJ [[340.71,374.41,230.94],[0.015924,-0.835132,0.549754,-0.00841319],[0,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09]],v300,z50,tool0; IPIEZA SUBMINISTRADOR
96
97 IICERRAR PINZA
98 WaitTime 2;
99 SET doCierrePinza;
100
101 MoveJ [[226.41,357.06,458.51],[0.00199281,0.835225,-0.549804,0.010488],[0,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09]],v300,z50,tool0; IPIEZA DE ESTACION 1 A ESTACION 2
102 MoveJ [[-125.29,336.05,470.13],[0.0159218,-0.835136,0.549749,-0.00841036],[1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09]],v300,z50,tool0; IPIEZA DE ESTACION 1 A ESTACION 2
103 MoveJ [[-123.72,334.51,300.34],[0.0159442,-0.83514,0.549742,-0.00838857],[1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09]],v300,z50,tool0; IPIEZA DE ESTACION 1 A ESTACION 2
104 WaitTime 2;
105 IIABRIR PINZA
106 RESET doCierrePinza;
107 WaitTime 2;
108 RESETE DO10_4;
109 RESETE DO10_3;
110 SET DO10_2;
111 WAITTIME 4;
112 ENDPROC

```

Il·lustració 86.Funció de subministrament Robot Studio (font: Pròpia)

Quan es rep el feedback de funció de subministrament s'encarrega d'anar cap a l'estació de subministrament, espera 2 segons per seguretat, tanca la pinça i torna a esperar 2 segons per seguretat. Seguidament transporta la peça cap a l'estació de detecció de tipus de peça, quan esta en la posició adequada espera 2 segons per seguretat, obra la pinça per deixar la peça i torna a contar 2 segons per seguretat, finalment envia un feedback al PLC per confirmar que la peça esta a l'estació de detecció tipus de peça durant 4 segons.

FUCIÓ DETECCIÓ PEÇES EN PALET

```

114 PROC DETECTA_PALET_E3()
115
116
117 IF Piezas =0 THEN
118
119 MoveJ [-100.31,340.39,376.25],[0.019764,-0.834117,0.551193,-0.00664833],[1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0;
120 MoveJ [-121.10,334.74,260.35],[0.0127347,-0.913716,0.405957,-0.0126118],[1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0;
121 !PIEZA DE ESTACION 2 A ESTACION 3
122 !CERRAR PINZA
123 WaitTime 2;
124 SET doCierraPinza;
125 WaitTime 2;
126 MoveJ [-109.27,334.49,358.36],[0.0128404,0.804209,-0.593835,-0.0210491],[1,0,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0;
127 MoveJ [-32.50,-227.31,477.76],[0.00258339,0.833282,-0.552841,-0.00114564],[-2,-1,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0;
128 MoveJ [-4.44,-223.79,325.42],[0.0160094,-0.835175,0.549687,-0.00841546],[-2,0,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0; !PIEZA DE ESTACION 2 A ESTACION 3
129 WaitTime 2;
130 RESET doCierraPinza;
131 !ABRIR PINZA
132 WaitTime 2;
133 MoveJ [29.79,-209.08,519.24],[0.0159474,-0.835136,0.549748,-0.00839068],[-1,0,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0; !PIEZA DE ESTACION 3 A REPOSO
134
135 RESET DO10_2; !DO10.2
136 SET DO10_4; !REPOS DO10_4
137 RESET DO10_3;
138 Piezas:= Piezas+1;
139 WAITTIME 4;
140

```

Il·lustració 87. Funció de detecció peces en palet Robot Studio (font: Pròpia)

En aquesta funció s'encarrega de detectar que es la primera peça a transportar cap a el palet.

Principalment fa els moviments necessaris per anar a la estació de detecció tipus de peça, una vegada allà espera 2 segons per seguretat ,tanca la pinça i torna a esperar 2 segons per seguretat.

Seguidament porta la peça cap a el palet, quan arriba al lloc desitjat , espera 2 segons per seguretat, obra la pinça i torna a esperar 2 segons per seguretat.

Finalment puja una mica cap a amunt per evitar qualsevol col·lisió , envia el feedback de que ha deixat la peça en el palet i incrementa el comptador de peces en 1 , per seguretat espera 4 segons.

```

145 ELSEIF Piezas = 1 THEN
146
147 MoveJ [-100.31,340.39,376.25],[0.019764,-0.834117,0.551193,-0.00664833],[1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0;
148 MoveJ [-121.10,334.74,260.35],[0.0127347,-0.913716,0.405957,-0.0126118],[1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0;
149 !PIEZA DE ESTACION 2 A ESTACION 3
150 !CERRAR PINZA
151 WaitTime 2;
152 SET doCierraPinza;
153 WaitTime 2;
154 MoveJ [-109.27,334.49,358.36],[0.0128404,0.804209,-0.593835,-0.0210491],[1,0,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0;
155 MoveJ [-32.50,-227.31,477.76],[0.00258339,0.833282,-0.552841,-0.00114564],[-2,-1,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0;
156 MoveJ [68.57,-305.23,518.79],[0.0160102,-0.835157,0.549715,-0.00840888],[-1,0,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0;
157 MoveJ [73.80,-302.34,353.61],[0.0160078,-0.835159,0.549711,-0.00840838],[-1,0,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0;
158 MoveJ [79.84,-312.26,353.61],[0.0160273,-0.835143,0.549736,-0.0083965],[-1,0,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0;
159 MoveJ [80.71,-311.78,326.14],[0.016012,-0.835153,0.54972,-0.00840594],[-1,0,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0;
160
161
162
163 WaitTime 2;
164 RESET doCierraPinza;
165
166
167 !ABRIR PINZA
168 WaitTime 2;
169 MoveJ [74.55,-315.19,520.76],[0.0160119,-0.835151,0.549724,-0.00840809],[-1,0,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09],v300,z50,tool0;
170 !PIEZA DE ESTACION 3 A REPOSO
171 RESET DO10_2; !DO10.2
172 SET DO10_4; !REPOS DO10_4
173 RESET DO10_3;
174 Piezas:=0;
175 WAITTIME 4;
176
177 ENDIF
178
179
180 ENDPROC

```

Il·lustració 88. Funció de subministrament segona peça Robot Studio (font: Pròpia)

En aquesta funció s'encarrega de detectar que es la segona peça a transportar cap a el palet.

Principalment fa els moviments necessaris per anar a la estació de detecció tipus de peça, una vegada allà espera 2 segons per seguretat ,tanca la pinça i torna a esperar 2 segons per seguretat.

Seguidament porta la peça cap a el palet, quan arriba al lloc desitjat , espera 2 segons per seguretat, obra la pinça i torna a esperar 2 segons per seguretat.

Finalment puja una mica cap a amunt per evitar qualsevol col·lisió , envia el feedback de que ha deixat la peça en el palet i decrementa el comptador de peces en 0 , per seguretat espera 4 segons.

FUNCIÓ MAGATZEM

```

181 PROC Magatzem()
182
183 IF Magatzem_2_Piezas =0 THEN
184
185 MoveJ [-2.63,-227.07,450.71],[0.000369522,0.718806,-0.69521,-0.000811032],[-2,-1,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
186 MoveJ [[2.52,-227.17,319.97],[0.000371904,0.718808,-0.695208,-0.000809852],[-1,-1,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
187 WaitTime 2;
188 Set doCierraPinza;
189 MoveJ [[2.87,-227.29,530.54],[0.000362231,0.718803,-0.695213,-0.000809346],[-1,-1,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
190 MoveJ [[60.14,202.09,529.10],[0.000371524,0.718803,-0.695213,-0.00080943],[0,0,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
191 MoveJ [[14.98,282.49,193.68],[0.000264724,0.803856,-0.594823,-0.000847615],[0,0,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
192 WaitTime 2;
193 Reset doCierraPinza;
194 WaitTime 2;
195 MoveJ [[16.24,282.53,356.25],[0.000729204,-0.772711,0.634755,0.00149954],[0,0,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
196 WaitTime 2;
197 Set DO10_3;
198 WaitTime 4;
199
200 Magatzem_2_Piezas :=1;
201
202
203 ELSEIF Magatzem_2_Piezas = 1 THEN
204
205
206 MoveJ [-2.63,-227.07,450.71],[0.000369522,0.718806,-0.69521,-0.000811032],[-2,-1,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
207 MoveJ [[2.52,-227.17,319.97],[0.000371904,0.718808,-0.695208,-0.000809852],[-1,-1,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
208 WaitTime 2;
209 Set doCierraPinza;
210 MoveJ [[2.87,-227.29,530.54],[0.000362231,0.718803,-0.695213,-0.000809346],[-1,-1,-3,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
211 MoveJ [[60.14,202.09,529.10],[0.000371524,0.718803,-0.695213,-0.00080943],[0,0,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
212 MoveJ [[14.98,282.49,193.68],[0.000264724,0.803856,-0.594823,-0.000847615],[0,0,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
213 WaitTime 2;
214 Reset doCierraPinza;
215 WaitTime 2;
216 MoveJ [[16.24,282.53,356.25],[0.000729204,-0.772711,0.634755,0.00149954],[0,0,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
217 WaitTime 2;
218 Set DO10_3;
219 WaitTime 4;
220
221 Magatzem_2_Piezas :=0;
222
223
224 ENDIF
225 ENDPROC

```

Il·lustració 89.Funció de magatzem Robot Studio (font: Pròpia)

La funció de magatzem consta de dos parts , una per emmagatzemar la primera peça i l'altre per la segona.

Per la primera peça , principalment va cap a la posició del palet una vegada allà espera 2 segons per seguretat, tanca la pinça i seguidament va cap a la posició on es desitja emmagatzemar la peça , una vegada en el lloc pertinent espera 2 segons per seguretat ,obra la pinça i torna a contar 2 segons per seguretat .

Finalment puja cap amunt per evitar qualsevol col·lisió ,per seguretat espera 2 segons i envia el feedback al PLC de que la peça ja ha sigut emmagatzemada , conta 4 segons per seguretat i incrementa el comptador de peces emmagatzemades en 1.

Per la segona peça , principalment va cap a la posició del palet una vegada allà espera 2 segons per seguretat, tanca la pinça i seguidament va cap a la posició on es desitja emmagatzemar la peça , una vegada en el lloc pertinent espera 2 segons per seguretat ,obra la pinça i torna a contar 2 segons per seguretat .

Finalment puja cap amunt per evitar qualsevol col·lisió ,per seguretat espera 2 segons i envia el feedback al PLC de que la peça ja ha sigut emmagatzemada , conta 4 segons per seguretat i decrementa el comptador de peces emmagatzemades en 0.

FUNCIÓ CONTENIDOR

```

231 PROC CONTENIDOR()
232
233
234 MoveJ [[-123.72,334.51,300.32],[0.0159296,-0.835176,0.549688,-0.0084091],[1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
235 MoveJ [[-115.47,324.76,258.18],[0.0414741,-0.850917,0.523322,0.0188101],[1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
236 WaitTime 2;
237 Set doCierraPinza;
238
239
240 WaitTime 2;
241 MoveJ [[-122.10,323.68,391.17],[0.029589,-0.843205,0.536748,0.00561295],[1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
242 MoveJ [[-250.23,224.14,382.73],[0.0295958,-0.843203,0.53675,0.00562585],[1,-1,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
243 MoveJ [[-247.58,230.79,264.12],[0.0295886,-0.843199,0.536756,0.00561906],[1,-1,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
244 WaitTime 2;
245
246
247 Reset doCierraPinza;
248 WAITTIME 2;
249 MoveJ [[-267.42,257.92,469.54],[0.0156565,-0.85092,0.524986,-0.00888607],[1,-1,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tool0;
250 WaitTime 2;
251 Reset DO10_3;
252 Reset DO10_2;
253 Set DO10_4;
254 WaitTime 2;
255
256
257
258
259 ENDPROC

```

Il·lustració 90.Funció de contenidor Robot Studio (font: Pròpia)

En la funció de contenidor quan rep el feedback del PLC , es mou fins la estació de detecció de tipus de peça, una vegada fa el moviment conta 2 segons per seguretat i tanca la pinça per tal d'agafar la peça per seguretat conta uns altres 2 segons.

Quan passa el temps mou el robot fins el contenidor de peces no metàl·liques, una vegada realitza el moviment espera 2 segons per seguretat.

Finalment obra la pinça, espera 2 segons per seguretat , puja cap amunt per evitar qualsevol col·lisió torna a contar 2 segons per seguretat i envia el feedback al PLC de que el robot ha deixat la peça en el contenidor i conta 2 segons.

3.3.5. Programació robot Fanuc LR MATE 200id 4S

VARIABLES PLC-FANU

Entrades PLC	Nom de les funcions	Sortides FANUC
I3.0	Output 1 Fanuc/ Preparat	DO117
I3.1	Output 2 Fanuc/ Palet_E4	DO115
I3.2	Output 3 Fanuc/ Foradat_contenedor	DO113
I3.3	Output 4 Fanuc/Foradadora_a_palet	DO111

Taula 11. Correspondència entre Entrades del PLC i Sortides FANUC

Sortides PLC	Nom de les funcions	Entrades FANUC
Q6.0	Input 1 Fanuc/Preparat	DI107
Q6.1	Input 2 Fanuc/Palet_E4	DI103
Q6.2	Input 4 Fanuc/Plat contenedor	DI105
Q6.3	Input 5 Fanuc/Plat_Palet	DI101

Taula 12. Correspondència entre Sortides del PLC i Entrades FANUC

PROGRAMA PRINCIPAL

```

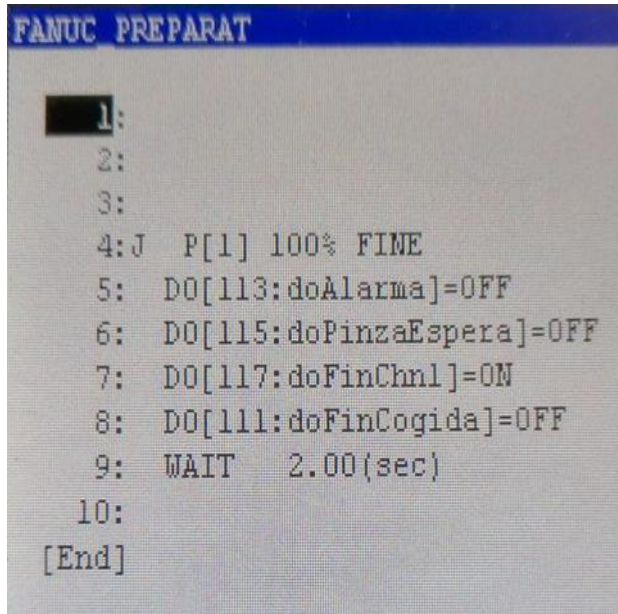
FANUC 1
1:
2: LBL[1]
3: DO[111:doFinCogida]=OFF
4: DO[113:doAlarma]=OFF
5: DO[115:doPinzaEspera]=OFF
6: DO[117:doFinChnl]=OFF
7: IF (DI[107:sys]),
   : CALL FANUC_PREPARAT
8: IF (DI[101:sys]),
   : CALL FANUC_PALET_PLAT
9: IF (DI[105:sys]),
   : CALL FANUC_PLAT_PALET
10: IF (DI[103:sys]),
   : CALL FANUC_CONTENIDOR
11: JMP LBL[1]
[End]

```

Il·lustració 91. Funció principal de robot FANUC roboguide (font: Pròpia)

Dintres del programa principal del robot FANUC principalment s'aturen totes les accions del robot i espera el feedback del PLC per tal de fer trucada a diferents programes, tota la estona va verificant els estats del feedback del PLC.

PROGRAMA FANUC PREPARAT



```
FANUC PREPARAT
1:
2:
3:
4: J P[1] 100% FINE
5: DO[113:doAlarma]=OFF
6: DO[115:doPinzaEspera]=OFF
7: DO[117:doFinChnl]=ON
8: DO[111:doFinCogida]=OFF
9: WAIT 2.00(sec)
10:
[End]
```

Il·lustració 92. Funció preparat de robot FANUC roboguide (font: Pròpia)

En aquest programa porta el robot fins a la posició de seguretat, atura totes les altres sortides i activa el feedback per comunicar al PLC que el robot esta a la posició correcta, finalment s'espera 2 segons de seguretat

PROGRAMA FANUC DE PALET A PLAT

```
FANUC PALET PLAT
1:
2: IF R[10]=0,JMP LBL[10]
3: IF R[10]=1,JMP LBL[20]
4:
5: LBL[10]
6: RO[1:EvPinzalAct]=OFF
7: RO[2:EvPinzalRep]=ON
8: RO[3:EvPinza2Act]=OFF
9: RO[4:EvPinza2Rep]=ON
10:
11:
12:J P[10] 100% FINE
13: WAIT 2.00(sec)
14: RO[2:EvPinzalRep]=OFF
15: RO[1:EvPinzalAct]=ON
16: RO[3:EvPinza2Act]=ON
17: RO[4:EvPinza2Rep]=OFF
18:
19:
20: WAIT 2.00(sec)
21:J P[11] 100% FINE
```

Il·lustració 93. Funció de palet a plat de robot FANUC roboguide (font: Pròpia)

Principalment en aquest programa es verifica dos condicions ,si no s'ha fet cap peça 0 o si ja s'ha fet una peça 1.

Quan es la primera peça habilitem els dos actuadors de la pinça per obrir-la completament , seguidament el robot va cap a la posició del palet , espera 2 segons per seguretat ,tanca la pinça i per seguretat torna a esperar 2 segons.

```
FANUC PALET PLAT
22:J P[12] 100% FINE
23:J P[13] 100% FINE
24: WAIT 2.00(sec)
25: RO[2:EvPinzalRep]=ON
26: RO[1:EvPinzalAct]=OFF
27: RO[3:EvPinza2Act]=OFF
28: RO[4:EvPinza2Rep]=ON
29:
30:
31: WAIT 2.00(sec)
32:J P[14] 100% FINE
33: DO[111:doFinCogida]=ON
34: DO[113:doAlarma]=OFF
35: DO[117:doFinChml]=OFF
36: DO[115:doPinzaEspera]=OFF
37: WAIT 3.00(sec)
38:
39: R[10]=1
40: JMP LBL[30]
41:
42: LBL[20]
```

Il·lustració 94. Funció palet a plat part2 de robot FANUC roboguide (font: Pròpia)

Una vegada la pinça ha agafat la peça comença a fer el moviment fins el plat , una vegada esta situat en la zona del plat espera 2 segons per seguretat i obre la pinça per tal de deixar la peça en el plat , per seguretat torna a esperar 2 segons.

Finalment quan la peça esta en el plat ,puja el robot per evitar qualsevol col·lisió envia el feedback cap a el PLC per comunicar que ha finalitzat el moviment, espera 3 segons per seguretat i incrementa el comptador en 1 i salta fins al final del programa .


```
FANUC PALET PLAT
43: R0[1:EvPinzalAct]=OFF
44: R0[3:EvPinza2Act]=OFF
45: R0[4:EvPinza2Rep]=ON
46: R0[2:EvPinzalRep]=ON
47:J P[4] 100% FINE
48:J P[9] 100% FINE
49:
50:J P[8] 100% FINE
51: WAIT 2.00(sec)
52: R0[2:EvPinzalRep]=OFF
53: R0[1:EvPinzalAct]=ON
54: R0[3:EvPinza2Act]=ON
55: R0[4:EvPinza2Rep]=OFF
56: WAIT 2.00(sec)
57:J P[2] 100% FINE
58:J P[7] 100% FINE
59:J P[3] 100% FINE
60:
61: WAIT 2.00(sec)
62: R0[2:EvPinzalRep]=ON
63: R0[1:EvPinzalAct]=OFF
```

Il·lustració 95.Funció palet a plat part3 de robot FANUC roboguide (font: Pròpia)

Quan es la segona peça habilitem els dos actuadors de la pinça per obrir-la completament , seguidament el robot va cap a la posició del palet , espera 2 segons per seguretat ,tanca la pinça i per seguretat torna a esperar 2 segons

Una vegada la pinça ha agafat la peça comença a fer el moviment fins el plat , una vegada esta situat en la zona del plat espera 2 segons per seguretat i obre la pinça per tal de deixar la peça en el plat , per seguretat torna a esperar 2 segons.

```
FANUC PALET PLAT
8
60:
61: WAIT 2.00(sec)
62: RO[2:EvPinzalRep]=ON
63: RO[1:EvPinzalAct]=OFF
64: RO[3:EvPinza2Act]=OFF
65: RO[4:EvPinza2Rep]=ON
66: J P[5] 100% FINE
67: J P[6] 100% FINE
68: WAIT 2.00(sec)
69: DO[113:doAlarma]=OFF
70: DO[111:doFinCogida]=ON
71: DO[115:doPinzaEspera]=OFF
72: DO[117:doFinChnl]=OFF
73: WAIT 3.00(sec)
74: R[10]=0
75:
76:
77:
78:
79: LBL[30]
[End]
```

Il·lustració 96. Funció palet a plat part4 de robot FANUC roboguide (font: Pròpia)

Finalment quan la peça esta en el plat ,puja el robot per evitar qualsevol col·lisió envia el feedback cap a el PLC per comunicar que ha finalitzat el moviment, espera 3 segons per seguretat i decrementa el comptador en 0 i finalitza el programa.

PROGRAMA FANUC DE PALAT A PALET

```
FANUC PLAT PALET
1:
2: IF R[40]=0,JMP LBL[40]
3: IF R[40]=1,JMP LBL[50]
4:
5: LBL[40]
6:J P[1] 100% FINE
7: WAIT 2.00(sec)
8: RO[2:EvPinzalRep]=OFF
9: RO[1:EvPinzalAct]=ON
10: RO[3:EvPinza2Act]=ON
11: RO[4:EvPinza2Rep]=OFF
12: WAIT 2.00(sec)
13:J P[5] 100% FINE
14:J P[6] 100% FINE
15:J P[2] 100% FINE
16: WAIT 2.00(sec)
17: RO[2:EvPinzalRep]=ON
18: RO[1:EvPinzalAct]=OFF
19: RO[3:EvPinza2Act]=OFF
20: RO[4:EvPinza2Rep]=ON
21: WAIT 2.00(sec)
```

Il·lustració 97.Funció plat a palet part1 de robot FANUC roboguide (font: Pròpia)

Principalment en aquest programa es verifica dos condicions ,si no s'ha fet cap peça 0 o si ja s'ha fet una peça 1,.

Quan es la primera peça el robot baixa fins la posició on es troba la peça, espera 2 segons per seguretat, seguidament tanca la pinça per tal d'agafar la peça, torna espera 2 segons per seguretat i va directament cap a el palet , una vegada fa el moviment espera 2 segons per seguretat , obra les pinces i torna a esperar 2 segons per seguretat

```
FANUC PLAT PALET
22:J @P[3] 100% FINE
23: D0[111:doFinCogida]=OFF
24: D0[113:doAlarma]=OFF
25: D0[117:doFinChnl]=OFF
26: D0[115:doPinzaEspera]=ON
27: WAIT 3.00(sec)
28: R[40]=1
29: JMP LBL[60]
30:
31:
32: LBL[50]
33:J P[1] 100% FINE
34: WAIT 2.00(sec)
35: R0[2:EvPinzalRep]=OFF
36: R0[1:EvPinzalAct]=ON
37: R0[3:EvPinza2Act]=ON
38: R0[4:EvPinza2Rep]=OFF
```

Il·lustració 98.Funció plat a palet part2 de robot FANUC roboguide (font: Pròpia)

Una vegada que ha deixat la peça en el palet, puja cap a amunt per evitar qualsevol col·lisió, seguidament envia un feedback cap a el PLC confirmant que s'ha deixat la peça en el palet , incrementa el comptador de programa en 1 i fa un salt fins al final del programa.

Quan es la segona peça el robot baixa fins la posició on es troba la peça, espera 2 segons per seguretat, seguidament tanca la pinça per tal d'agafar la peça.

```
FANUC PLAT PALET
39: WAIT 2.00(sec)
40: J P[5] 100% FINE
41: J P[6] 100% FINE
42: J P[4] 100% FINE
43:
44: J P[7] 100% FINE
45: WAIT 2.00(sec)
46: RO[2:EvPinzaRep]=ON
47: RO[1:EvPinzaAct]=OFF
48: RO[3:EvPinza2Act]=OFF
49: RO[4:EvPinza2Rep]=ON
50: WAIT 2.00(sec)
51: J @P[3] 100% FINE
52: DO[111:doFinCogida]=OFF
53: DO[113:doAlarma]=OFF
54: DO[117:doFinChnl]=OFF
55: DO[115:doPinzaEspera]=ON
56: WAIT 3.00(sec)
57: R[40]=0
58: LBL[60]
[End]
```

Il·lustració 99. Funció plat a palet part 3 de robot FANUC roboguide (font: Pròpia)

Quan ha agafat la peça espera 2 segons per seguretat ,va directament cap a la posició del palet , una vegada fa el moviment espera 2 segons per seguretat , obra les pinces i torna a esperar 2 segons per seguretat.

Finalment puja cap a dalt per evitar qualsevol col·lisió, envia el feedback al PLC confirmant que ha finalitzat el moviment , espera 3 segons per seguretat , decrementa el comptador de programa en 0 i finalitza el programa

PROGRAMA PLAT A CONTENIDOR

```
1:J P[1] 100% FINE
2:J P[2] 100% FINE
3: WAIT 2.00(sec)
4: RO[1:EvPinzaAct]=ON
5: RO[2:EvPinzaRep]=OFF
6: RO[3:EvPinza2Act]=ON
7: RO[4:EvPinza2Rep]=OFF
8: WAIT 1.00(sec)
9:J P[1] 100% FINE
10:J P[3] 100% FINE
11:J P[4] 100% FINE
12: WAIT 2.00(sec)
13: RO[1:EvPinzaAct]=OFF
14: RO[2:EvPinzaRep]=ON
15: RO[3:EvPinza2Act]=OFF
16: RO[4:EvPinza2Rep]=ON
17: WAIT 1.00(sec)
18: DO[111:doFinCogida]=OFF
19: DO[113:doAlarma]=ON
20: DO[115:doPinzaEspera]=OFF
21: DO[117:doFinChnl]=OFF
```

Il·lustració 100.Funció contenidor de robot FANUC roboguide (font: Pròpia)

Principalment en aquest programa el robot es mou cap a el plat per agafar la peça , passa 2 segons per seguretat ,tanca la pinça i conta 1 segon per seguretat.

Seguidament porta la peça cap a la posició del contenidor, quan arriba espera 2 segons per seguretat i obre la pinça , per seguretat espera 1 segon i finalment envia un feedback cap a el PLC per tal de confirmar que ha descartat la peça.

Capítol IV. Planificació del projecte

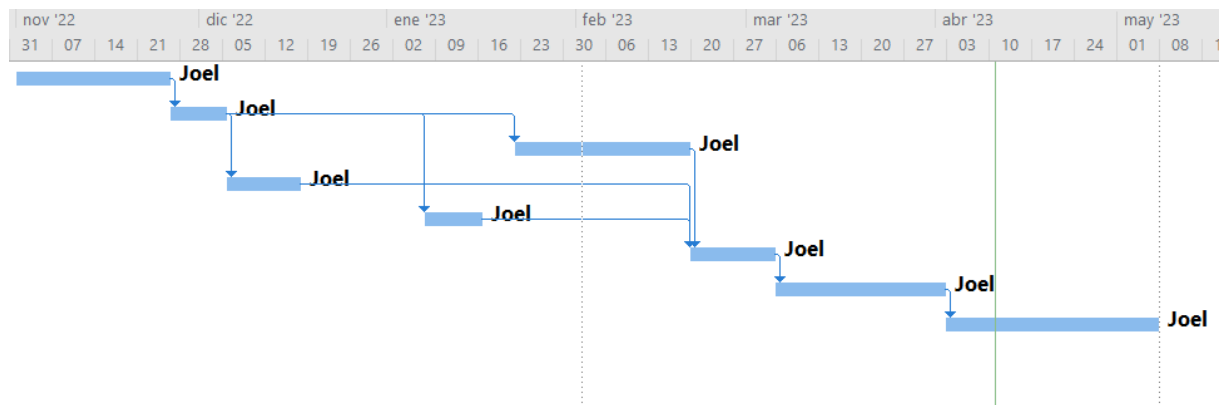
4.1. Planificació.

Principalment es realitzarà una planificació del projecte , on s’avaluarà quines tasques es realitzen en el projecte i s’establitzarà un temps de duració amb l’ajuda del software MS Projects.

Buscarem planificar la finalització del termini un mes abans de l’entrega del projecte per tal de tenir un ventall de temps evitant així futurs problemes.

Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
Avantprojecte	80 hores	mar 01/11/22	sáb 26/11/22		Joel
Modelització amb Matlab	32 hores	sáb 26/11/22	lun 05/12/22	1	Joel
Programació d’Autòmats Rockwell	85 hores	dom 22/01/23	dom 19/02/23	2	Joel
Programació Robot ABB IRB120	32 hores	lun 05/12/22	sáb 17/12/22	2	Joel
Programació Robot Fanuc LR Mate 200d 4S	32 hores	sáb 07/01/23	lun 16/01/23	2	Joel
Programació de scada en Hmi	40 hores	dom 19/02/23	dom 05/03/23	3;4;5	Joel
Probes de qualitat d'operació de programa	80 hores	dom 05/03/23	dom 02/04/23	6	Joel
Memòria del Projecte	100 hores	dom 02/04/23	dom 07/05/23	7	Joel

Taula 13. Planificació del projecte



Il·lustració 101. Diagrama de gannt del projecte

Avantprojecte:

Les 80h de l'avantprojecte les tenim dividides entre d'altres processos en la recerca d'informació per saber de que tracta el projecte, la recerca de possibles solucions per poder resoldre el problema i la verificació de que si es viable mediambiental, econòmica i tècnicament la nostra possible solució.

Modelització de matlab:

Dintres d'aquesta activitat trobarem la transferència del funcionament de la cèl·lula a un software de modelització per tal de comprovar si el sistema realitzat es viable o no.

Programació d'autòmat Rockwell:

En les 85h aplicades a la programació de l'autòmat , es té en compte principalment la conversió del coneixement de la cèl·lula en diagrames Grafcet per tal d'estructurar les idees i les subrutines i el aprenentatge i l'aplicació del llenguatge estructurat ST.

Programació Robot ABB IRB 120:

Per la programació del robot ABB es realitza prèviament un diagrama Grafcet que engloba el coneixement de l'autòmat i del robot , seguidament aconseguen els punts de coordenades desitjats i es programa en llenguatge RAPID de Robotstudio.

Programació Robot FANUC LR Mate 200d 4S:

Per la programació del robot Fanuc es realitza prèviament un diagrama Grafcet que engloba el coneixement de l'autòmat i del robot, seguidament aconseguen els punts de coordenades desitjats i es programa amb condicionals de C en el software Roboguide.

Programació de scada en HMI:

Per la programació de scada es realitza amb una comunicació directa amb el PLC per l'assignació de les variables i E/S del programa, el scada es realitza mitjançant el software factory talk view.

Proves de qualitat d'operació de programa:

Les 80h se proves serveix per garantir el correcte funcionament de la cèl·lula, comprovar que funcionen tots els elements i l'assemblatge dels diferents controladors programats ,els robots ,el PLC i el scada.

Memòria del projecte:

Finalment en aquesta activitat es realitza tota la documentació detallada i especifica del projecte.

Una vegada marcada la planificació del projecte amb les durades i la planificació , es tindrà en compte un seguiment del projecte per tal d'evitar futurs problemes i saber l'estat del projecte. Si fos necessari fer alguna modificació en la planificació del projecte es tindria els 30 dies de marge lliure establerts en la planificació.

Hi ha diferents mètodes a utilitzar:

Revisió de l'estat del projecte: realitzaran reunions regulars per el tractament de temes relacionat amb costos , qualitat i el programa

Sessions individuals: Es la recol·lecció d'informació abans de les reunions per tal de transferir l'estat del projecte

Formularis: Documents que entreguen individualment els mandres de cada equip a la persona encarregada per tal de tenir un feedback de l'estat del projecte. (Tecnocampus, 2023)

Punts de control

Durada del projecte	Freqüència mínima d'informació al client	Freqüència màxima d'informació al client
1 mes	Cada tres dies	Diari
1 any	Mensualment	Dos cops al mes
5 anys	Cada sis mesos	Cada dos mesos

Taula 14.Freqüència de punts de control

El nostre projecte consta d'una durada d'un any aproximadament, per lo consegüent farem una revisió amb el client cada mes aproximadament, degut a que la durada del nostre projecte es de mig termini hi s'ha de tenir un seguiment adequat per tal de que no succeeixin imprevistos.

Establint així els últims divendres de cada mes es realitzaran les reunions de seguiment, amb excepció d'alguna reunió esporàdica per algun tema transcendental.

Capítol V. Justificació de producte i Riscos

5.1. Pla de contingència

En aquest projecte es tindrà en compte un pla de contingència, per tal de prendre les mesures estratègiques pertinents prèviament preparades, de tal forma que es pugui fer front a situacions inesperades o de crisi evitant que afecti al projecte. L'objectiu a complir es minimitzar els impactes negatius que puguin sorgir.

El pla de contingència inclou una avaluació de riscos detallada, en la qual s'indiquen les possibles situacions d'emergència i les conseqüències, ha d'anar acompanyat d'un seguit d'accions per tal de remetre amb els inconvenients. Aquestes accions poden repercutir en resignació de recursos, canvi de processos de treball o estratègies de recuperació de tasques entre d'altres.

En definitiva, el pla de contingència servirà per poder anticipar-se a futurs problemes i poder garantir que el projecte es mantingui en el camí correcte.

5.1.1 Anàlisi de riscos

Principalment es realitzarà la identificació dels riscos que es troben en el projecte, per tal de centralitzar i controlar futurs problemes. Conjuntament amb una avaluació de l'impacte potencial i la probabilitat de que pugui succeir, per tal de prioritzar els riscos i decidir quins requereix major precaució.

Rubrica d'avaluació dels riscos

Per tal d'identificar i qualificar la gravetat dels riscos i el seu impacte dins del projecte, s'ha dissenyat la següent rúbrica que contempla la gravetat de la incidència, la freqüència amb la que succeeix i la facilitat de detectar-la.

Aquesta rúbrica té en compte la teoria de l'autocontrol i la qualitat, qualificada mitjançant un valor numèric. La suma del total ens indicarà quins són els riscos més perillosos.

Gravetat de la incidència		
Rúbrica	Valor	Criteri
Molt baixa	1	No existeixen precedents, però podria passar
Baixa	2-3	Esdeveniments aïllats en el passat
Moderada	4-6	Existeixen registres en projectes similars
Alta	7-8	Ja ha ocorregut algun cop en el passat
Molt alta	9-10	Casi inevitable, succeirà amb freqüència

Taula 15. Avaluació gravetat de la incidència

Freqüència de la incidència		
Rúbrica	Valor	Criteri
Molt baixa	1	No existeixen precedents, però podria passar
Baixa	2-3	Esdeveniments aïllats en el passat
Moderada	4-6	Existeixen registres en projectes similars
Alta	7-8	Ja ha ocorregut algun cop en el passat
Molt alta	9-10	Casi inevitable, succeirà amb freqüència

Taula 16. Avaluació freqüència de la incidència

Facilitat de detecció de la incidència		
Rúbrica	Valor	Criteri
Molt baixa	1	No existeixen precedents, però podria passar
Baixa	2-3	Esdeveniments aïllats en el passat
Moderada	4-6	Existeixen registres en projectes similars
Alta	7-8	Ja ha ocorregut algun cop en el passat
Molt alta	9-10	Casi inevitable, succeirà amb freqüència

Taula 17. Facilitat de detecció de la incidència

Seguidament una vegada establerta la rubrica d'avaluació de riscos, s'identificaran els riscos principals puntuant-los per veure quins son els riscos que necessiten mes precaució

Identificació dels riscos	Avaluació dels riscos	Total
Malversació del hardware de la cèl·lula de fabricació flexible, com podria ser Motors, Sensors o Actuadors	Gravetat de la incidència: 9 Frequència de la incidència: 3 Facilitat per detectar la incidència: 8	20
Obsolescència de llicències dels softwares utilitzats per la programació, Robotstudio, Roboguide, Rs-Logix 5000	Gravetat de la incidència: 7 Frequència de la incidència: 4 Facilitat per detectar la incidència: 1	12
Problemes amb l'electricitat o amb el compressor d'aire utilitzat.	Gravetat de la incidència: 10 Frequència de la incidència: 1 Facilitat per detectar la incidència: 9	20
Error de comunicació amb els robots, fent-se mal bé els connectors USB.	Gravetat de la incidència: 4 Frequència de la incidència: 1 Facilitat per detectar la incidència: 9	14
Malversació del flex pendal dels robots tant del ABB com del FANUC	Gravetat de la incidència: 9 Frequència de la incidència: 1 Facilitat per detectar la incidència: 9	19
Problemes de comunicació amb el PLC via Ethernet, podent fer-se mal bé el cable d'UTP	Gravetat de la incidència: 4 Frequència de la incidència: 2 Facilitat per detectar la incidència: 7	13
Mal estar o malaltia de treballadors específics.	Gravetat de la incidència: 7 Frequència de la incidència: 3 Facilitat per detectar la incidència: 6	16

Taula 18. Identificació de riscos i avaluació

Tal i com podem observar en la Taula 10 ,es veu que els riscos mes perillosos son la malversació del hardware i els problemes amb l'electricitat o la pressió subministrada en el quadre general de control i potencia amb una puntuació de 20 punts.

5.1.2.Pla d'actuació

Utilitzarem el pla d'actuació per tal d'establir les accions especificques que s'han de prendre per tal de prevenir ,mitigar o solucionar els riscos identificats prèviament en l'anàlisi de riscos del projecte.

Una vegada identificats els riscos en funció de l'anàlisi de risc i identificat les activitats mes critiques , es realitza una descripció de les accions que s'han de prendre per cada risc identificat, on ha inclou una descripció detallada de les accions que s'han de prendre.

Seguidament es defineix quins son els responsables de realitzar cada acció, per tal de que les accions es realitzin de forma efectiva , conjuntament amb el cronograma que indica el temps que triga en aplicar-se cada acció i els recursos necessaris ja sigui personal, equips o pressuposts entre d'altres

Principalment per prioritat s'han d'abordar els següents riscos:

1. Malversació del hardware, Motors , sensors o actuadors:

Per evitar aquests problemes o poder detectar el començament del mal funcionament. La part tècnica s'ha d'encarregar el departament de manteniment , començant a treballar amb una filosofia de manteniment preventiu, anant revisant periòdicament els estats dels instruments, com podria ser l'últim divendres de cada setmana. Comprovar que els sensors no donen falses senyals i detecten amb la precisió adequada , els paràmetres del motor de la cinta treballen amb bon funcionament sense tenir un sobre-consum o sorolls inusuals entre d'altres.

També té una responsabilitat important el departament de compres, el qual podria tenir un petit stock d'instrumentació no molt costosa i estandarditzada, com podria ser algun sensor reed o sensor inductiu , això generaria un cost adicional però ajudaria a no tenir problemes en la planificació del projecte.

2. Problemes de subministrament d'electricitat o pressió:

Aquest es un apartat complex, en el qual es difícil donar una solució òptima per minimitzar els problemes, degut a que el subministrament d'energia i de la pressió mitjançant un compressor bé proporcionada per una empresa externa . De tal forma que si es vol solucionar el problema s'hauria de trucar al servei tècnic subcontractat per tal de que solucionin l'averia.

Tal que la solució proposada es anar fent un seguiment gràfic del procés de la cèl·lula, amb la consegüent que si es genera un problema que inhabilita la utilització de la cèl·lula poder mostrar el procés d'automatització realitzat.

Igualment per la realització de la programació principal no es necessita estar connectat a la cèl·lula de fabricació flexible, així que es podria seguir programant sense cap problema.

3. Malversació de flex pendal dels robots ABB i Fanuc.

Es podria generar una malversació del flex pendal , amb una caiguda del comandament o algun problema de tal forma que els robots es quedarien inhabilitats.

S'hauria d'enviar el robot a l'empresa subministradora per reparar-ho o trucar al servei tècnic per tal de solucionar el problema.

Mentrestant es pot utilitzar una simulació gràfica amb els softwares que proporcionen cada proveïdor Robotstudio per ABB i Roboguide per Fanuc . Per tant es podria seguir amb la programació dels robots.

4. Mal estar o malaltia de treballadors específics

En algun dels casos, en mig d'alguna activitat el personal destinat a realitzar-la es podria posar malalt, consegüentment que afectaria proporcionalment a la planificació del projecte .

De tal forma que la planificació del projecte es planifica el temps de finalització un mes abans del termini d'entrega , així hi ha aquest ventall de temps per aquests inconvenients.

5. Errors de comunicació amb els robots o amb el PLC:

Per evitar que aquets errors de comunicació amb els robots o amb el PLC puguin afectar a la finalització del projecte.

Setmanalment es fa un backup dels programes dintre de la CPU dels robots i del PLC per tal d'emmagatzemar el programa en el interior dels controladors. Així evitar que si en algun futur es trenques els USB dels robots o el cable d'Ethernet de comunicació amb el PLC , ja hi ha un programa carregat dintre del controlador.

6. Obsolescència de les llicències del software:

Les llicències dels programes normalment són anuals. Per evitar el problema amb les llicències s'ha de parlar amb els proveïdors per tal de que quan quedi un mes per el venciment de la llicència, envia el pressupost de les noves llicències per tal de poder realitzar el pagament i així evitar quedar-nos sense la utilització del software.

5.2.Perspectiva de gènere

En l'estudi de perspectiva de gènere dins del projecte d'automatització, es tindrà en compte possibles patrons de discriminació de gènere.

Aquest projecte al ser un projecte destinat a la docència no es realitza cap tipus de formació específica ni diferencial cap a ningun gènere.

Degut a que es un projecte destinat a la docència, s'ha de tenir en compte les beques i ajudes per dones que s'estan generant a l'actualitat per tal de fomentar els graus d'enginyeria, com podria ser les beca de la Fundació Universitària.

La Fundació universitària ofereix beca per estudiants dones que vulguin estudiar graus o màsters en la Universitat Politècnica de Catalunya. Aquest tipus de beques arriben a el pagament de la matrícula i una retribució econòmica mensual per gestos de manutenció. (UPC, 2020)

Com que aquest projecte esta destinat a la universitat Tecnocampus, amb l'ajuda de la fomentació d'aquests estudis mitjançant les beques no es genera cap discriminació cap al sexe femení ni cap diferencia per la utilització i l'aprenentatge sobre la cèl·lula de fabricació flexible

Capítol VI. Conclusions

Les conclusions extretes una vegada realitzat tot el projecte, són principalment que es una gran ajuda la verificació de tot el procés amb la modelització de matlab ,per tal de comprovar que el procés es viable i no te cap errada. Simulant el procés amb la ajuda de modelització de matlab, ja que es una eina de treball que permet localitzar la falta d'algun sensor o d'algun actuador per realitzar tot el procés desitjat.

Un anàlisis que he extret, es el benefici que pot generar l'aplicació de sub-rutines i diferents funcions dintres del programa ,aquestes serveixen per separar els diferents processos de la cèl·lula ,de tal forma que això ajuda a centralitzar possibles errors o tenir una millor estructura de programa a l'hora de fer alguna ampliació.

Considero que la utilització de el llenguatge ST dona molts beneficis a l'hora de tenir un programa mes net i estructurat , ja que el Ladder fa que es generin moltes línies de codi i de contactes , depenent el procés pot arribar a ser molt confús.

Finalment m'ha semblat un projecte molt interesant, a l'hora de haver de configurar diferents tipus diferents tipus de robot , el ABB i el FANUC per tal de familiaritzar-te amb els llenguatges i la estructura amb la qual treballa cadascun,

Capítol VII. Referencias

1. Alvarado, D. d. (Primavera 2020). Disseny i programació d'una cèl·lula de fabricació flexible. *Tecnocampus Mataró-Maresme*, 98.
2. Graells, J. A. (Primavera 2011). Disseny d'una cèl·lula de treball robotitzada que utilitzarà un robot industrial asea irb 123. *Tecnocampus Mataró-Maresme*, 236.
3. Gutiérrez, C. C. (2021). *Disseny d'una cèl·lula robotitzada que utilitzarà un robot ASEA IRB 120*, 236.
4. Normalización, A. E. (s.f.). *Buscador de comités técnicos de normalización*. Obtenido de <https://www.une.org/encuentra-tu-norma/comites-tecnicos-de-normalizacion>
5. Peñalver, S. B. (Primavera 2021). Control de la cèl·lula de fabricació flexible i dels robots abb irb120 i fanuc lr mate 200id/4s mitjançant interfície hmi. *TecnoCampus Mataró-Maresme*, 91.
6. Tecnocampus. (2023). Seguiment de projectes, control i execució. 13.
7. UPC. (27 de Febrero de 2020). *Universitat Politecnica de catalunya*. Obtenido de <https://www.upc.edu/es/sala-de-prensa/noticias/el-barca-y-la-upc-impulsan-una-beca-para-promover-el-acceso-profesional-de-chicas-estudiantes-en-el-mundo-del-deporte-y-la-tecnologia>
8. Vasco, U. d. (s.f.). *Sistemas de fabricación flexible*. Obtenido de https://www.ehu.eus/manufacturing/docencia/1151_ca.pdf