

## Treball final de grau

**Estudi:** Grau en Enginyeria Electrònica Industrial i Automàtica

**Títol:** Paletització d'una línia productiva.

**Document:** 1. Memòria

**Alumne:** Gerard Falgueras Regincós

**Tutor:** Marc Palom Agustí

**Departament:** Enginyeria Elèctrica, Electrònica i Automàtica

**Àrea:** Enginyeria de Sistemes i Automàtica

**Convocatòria (mes / any):** setembre / 2023

## ÍNDEX

1.	INTRODUCCIÓ .....	4
1.1.	Antecedents .....	4
1.2.	Objecte .....	4
1.3.	Especificacions i abast .....	4
2.	FUNCIONALITATS DE LA PALETITZACIÓ.....	5
3.	INSTAL·LACIÓ ELÈCTRICA.....	7
3.1.	Quadre elèctric .....	7
3.2.	PLC .....	7
3.3.	Seccionador .....	8
3.4.	Interruptor diferencial .....	8
3.5.	Pantalla tàctil .....	8
4.	PROGRAMACIÓ .....	9
4.1.	Taula d'entrades i sortides: .....	9
4.2.	Procés de paletització .....	11
4.2.1	GRAFCET de conducció .....	12
4.2.2	GRAFCET de seguretat .....	12

4.2.3	Buffer caixes .....	13
4.2.4	Empenyedor.....	14
4.2.5	Transport de corròns.....	14
4.2.6	Separador .....	15
4.2.7	Transport corròns palet entrada.....	15
4.2.8	Transport corròns palet sortida .....	16
4.2.9	Buffer pila de palets .....	17
4.3.	Programa PLC.....	18
4.3.1	Configuració del projecte .....	19
4.3.2	Programació GRAFCET i lògica de control.....	20
4.3.3	Simulació.....	23
4.4.	Pantalles HMI .....	24
4.5.1	Pantalla General .....	24
4.5.2	Pantalla d'usuaris.....	25
4.5.3	Pantalla manual .....	27
4.5.4	Pantalla alarmes .....	28
4.5.5	Pantalla receptes .....	30

4.5.	Simulació amb el programari Factory I/O .....	31
4.6.	Programació Robot Staubli .....	34
4.2.1	Tipologia de mà.....	35
4.2.2	Comunicació entre robot i PLC .....	37
4.2.3	Tipus de paletitzats .....	38
4.2.4	Codi START .....	39
4.2.5	Codi STOP .....	52
5.	RESUM DEL PRESSUPOST .....	53
6.	CONCLUSIONS .....	54
7.	RELACIÓ DE DOCUMENTS .....	55
8.	BIBLIOGRAFIA .....	56
9.	GLOSSARI .....	57

## **1. INTRODUCCIÓ**

En els següents apartats es descriu el projecte que es pretén realitzar, la paletització automàtica d'una línia productiva.

### **1.1. Antecedents**

En el present projecte, s'ha plantejat l'automatització d'una línia productiva ja existent, on actualment es realitza la paletització de les caixes de manera manual. La infraestructura elèctrica i pneumàtica es reutilitzarà de la mateixa línia, ja que es va preveure aquesta possibilitat durant la instal·lació de la part productiva.

Els elements mecànics seran adquirits a proveïdors externs, mentre que la instal·lació elèctrica i pneumàtica serà portada a cap per personal especialitzat, i per tant els càlculs i dimensionaments de les diferents màquines estaran a càrrec dels mateixos proveïdors.

### **1.2. Objecte**

L'objectiu d'aquest projecte és l'automatització i el control del conjunt de màquines per a la implementació de la paletització automàtica, amb la finalitat de reduir els costos i els riscos laborals associats a aquesta tasca. Mitjançant l'automatització, una vegada que les caixes estiguin tancades per una precintadora automàtica, es procedirà a apilar-les formant un mosaic predeterminat pel programa seleccionat. Això permetrà optimitzar l'espai del palet i augmentar l'estabilitat del producte un cop paletitzat.

### **1.3. Especificacions i abast**

Per a la realització de l'automatització, s'optarà per utilitzar un PLC S7-1200 de la marca Siemens, juntament amb mòduls d'ampliació d'entrades i sortides. A més, s'incorporarà una pantalla tàctil HMI també de la marca Siemens, així com un robot de 6 eixos de la marca Staubli, juntament amb el seu controlador corresponent. Totes els senyals generades per aquest sistema seran de tipus digital. La comunicació entre els diversos elements de la instal·lació, com el PLC, la pantalla HMI, es durà a terme a través de PROFINET o mitjançant les diferents entrades i sortides previstes.

## 2. FUNCIONALITATS DE LA PALETITZACIÓ

La nova instal·lació comptarà amb un buffer de caixes, que estarà constituït per una cinta transportadora motoritzada amb prou longitud per emmagatzemar temporalment un nombre de caixes. Aquesta funcionalitat permetrà gestionar de manera eficient situacions imprevistes o anomalies en la zona del robot.

A continuació, es trobarà un transportador de corrns motoritzat, dissenyat per transportar i separar les caixes de la cinta transportadora anterior. Això és fonamental per garantir que el manipulador, en agafar el nombre de caixes programades, no n'agafi cap d'addicional o en deixi algun excés. Per a aquest propòsit, es comptarà amb un empenyedador al final del buffer de caixes, que facilitarà el canvi de direcció de les caixes, i un separador, que actuarà com una barrera física per prevenir l'entrada de caixes addicionals a la zona d'agafada del manipulador. Totes aquestes funcions seran controlades a través d'electrovàlvules pneumàtiques per assegurar una operació precisa i fiable.

D'altra banda, es disposarà d'un altre transportador de corrns motoritzat, encarregat de transportar una pila de palets per proveir el manipulador amb palets necessaris.

Una altra zona de la instal·lació comptarà amb dos transportadors de corrns idèntics. En aquesta àrea, a l'inici del cicle, es disposarà d'un palet buit. A mesura que les caixes vagin entrant, el robot les apilarà per construir una pila de caixes predeterminada pel programa. Un cop s'hagi completat la pila conforme al patró establert, el palet complet serà expulsat pel camí del transportador de corrns corresponent, i a continuació, el robot continuarà apilant nous palets en l'altre transportador de corrns.

Finalment, el nexa d'unió de tot el sistema serà el robot o manipulador, el qual serà de la marca Staubli, model TX200 range. Aquest robot tindrà una càrrega màxima de 130 kg, segons les especificacions del proveïdor, la qual cosa permetrà executar les tasques amb eficiència i sense problemes. Els operaris tindran la possibilitat d'interactuar amb la instal·lació a través de la pantalla HMI ubicada al quadre general. En cas de qualsevol problema o incidència, disposaran de dues portes habilitades per a aquest propòsit, les quals els permetran accedir a l'interior de la instal·lació.

A continuació, es presentaran les figures 1 i 2 de la nova instal·lació en formats 2D i 3D, amb l'objectiu de proporcionar una millor visualització del que s'obtindrà en la realitat.

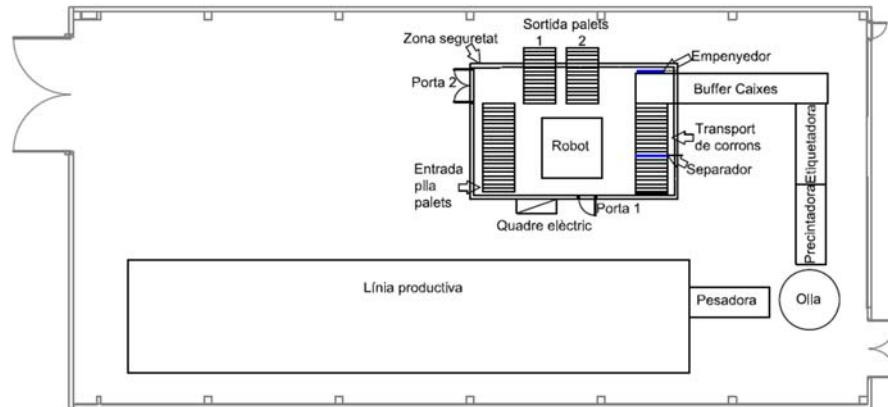


Figura 1. Situació nova instal·lació en 2D.



Figura 2. Situació nova instal·lació en 3D.

### **3. INSTAL·LACIÓ ELÈCTRICA**

En aquest apartat es descriurà el procés de la instal·lació, muntatge i configuració de tots els elements elèctrics de control i potència de la nova instal·lació. La xarxa serà alimentada amb 3 fases a 400 V més neutre a 50 Hz i es connectarà a terra per garantir la seguretat adequada.

#### **3.1. Quadre elèctric**

L'armari elèctric que s'ha seleccionat per a la instal·lació és de la marca ELDON, amb dimensions de 1950x1000x400 mm. Aquest és destinat a albergar tots els elements elèctrics de seguretat, potència i control requerits per a l'execució de la tasca. La gestió de la part de la sensòrica, electrovàlvules i motors es realitzarà des del mateix quadre, tot i que cal destacar que aquest estarà ubicat fora de la zona de treball.

La pantalla tàctil HMI juntament amb els pulsadors de marxa, aturada i parada d'emergència de la instal·lació s'instal·larà a la porta del quadre. Per millorar l'accessibilitat, l'armari es disposarà sobre una bancada específica. Aquesta disposició facilitarà la connexió i sortida de les mànegues de potència i control de la instal·lació, permetent un manteniment més senzill i còmode de tots els components elèctrics de la nova instal·lació.

#### **3.2. PLC**

S'utilitza un autòmat de la marca SIEMENS, concretament el model S7-1200, amb una CPU 1214 AC/DC/RLY per a aquesta aplicació específica. Aquest autòmat ofereix un total de 14 entrades digitals i 10 sortides digitals. Amb l'objectiu de garantir un control precís de la instal·lació, s'han incorporat tres mòduls addicionals: un de 16 entrades digitals, un altre de 8 entrades digitals i un mòdul més amb 16 sortides digitals.

La selecció d'aquest autòmat es basa principalment en la robustesa associada a la marca SIEMENS, juntament amb la capacitat de les seves sortides per controlar directament contactors i electrovàlvules sense requerir elements intermediaris. Això minimitza el risc de danys a les sortides. Concretament, les sortides es commuten mitjançant relés (Relay), segons la descripció de la CPU, aportant així major fiabilitat i seguretat en el control dels diversos elements de la instal·lació.



### **3.3. Seccionador**

El seccionador escollit és de la marca SCHNEIDER de 63 A i de 3P + N. Aquest element deixa circular el corrent mitjançant una maneta rotativa vermella, que quan es troba en posició vertical indica el pas de corrent i alimentació dins del quadre elèctric. Amb aquest corrent màxim de 63 A ja és suficient per protegir la instal·lació, ja que no se superarà en cap cas aquest valor.

### **3.4. Interruptor diferencial**

S'ha seleccionat un interruptor diferencial de la marca SCHNEIDER de 63A i 3P + N amb una sensibilitat de 30 mA per protegir la instal·lació de possibles fuites de corrent. Aquest interruptor diferencial és capaç de detectar fins a 30 mA de corrent de fuga i actuarà ràpidament per desconectar l'alimentació en cas de detectar una situació de perill.

La sensibilitat de 30 mA és suficient per proporcionar una protecció efectiva davant de possibles contactes indirectes o fuites de corrent a terra. Això ajuda a garantir la seguretat dels operaris i a prevenir accidents i danys als equips. Amb aquest interruptor diferencial, es millora la fiabilitat i la seguretat de tota la instal·lació elèctrica.

### **3.5. Pantalla tàctil**

S'ha optat per la pantalla tàctil SIMATIC HMI TP 1500 Basic color PN de la marca SIEMENS, amb una amplada de 15", com a element d'interacció amb la instal·lació. La resolució de pantalla d'aquesta pantalla és òptima perquè l'operari pugui interactuar de manera clara i precisa amb el sistema. A més, s'ha triat aquesta pantalla per la seva fàcil implementació i integració amb el programari de programació del PLC escollit.

## 4. PROGRAMACIÓ

La programació d'aquest projecte s'efectuarà mitjançant el programari TIA PORTAL V13, desenvolupat per l'empresa SIEMENS. S'optarà per utilitzar el diagrama de contactes (KOP) i el llenguatge estructurat (ST) com a llenguatges de programació per a la implementació del control de la instal·lació.

Mitjançant el TIA PORTAL V13, els enginyers podran crear, depurar i optimitzar el codi de control amb facilitat i eficàcia. El diagrama de contactes oferirà una representació visual i intuïtiva del funcionament dels elements del sistema, mentre que el llenguatge estructurat permetrà una programació més avançada i estructurada dels diferents processos.

La combinació d'aquests dos llenguatges de programació permetrà un control precís i eficient de tota la instal·lació, facilitant la integració i la interacció entre els diferents components. Amb el programari TIA PORTAL V13 i l'ús dels llenguatges KOP i ST, es garantirà una programació robusta i fiable per a l'automatització completa del sistema.

### 4.1. Taula d'entrades i sortides:

A continuació, es presentarà la figura 3 amb les entrades utilitzades en el projecte.

Nom	Tipus de dades	Direcció	Comentari
K1	Bool	%i0.0	Relé de seguretat 1
-	Bool	%i0.1	Reserva
SE1	Bool	%i0.2	Paro emergencia 1
SE2	Bool	%i0.3	Paro emergencia 2
SE3	Bool	%i0.4	Paro emergencia 3
B1	Bool	%i0.5	Entrada Buffer
B2	Bool	%i0.6	Sortida Buffer
B3	Bool	%i0.7	Entrada Caixes
B4	Bool	%i1.0	Entrada caixes Seperador
B5	Bool	%i1.1	Sortida caixes
B6	Bool	%i1.2	Palet 1
B7	Bool	%i1.3	Palet 2
B8	Bool	%i1.4	Palet sortida
B9	Bool	%i1.5	Palet entrada
B10	Bool	%i2.0	Empenyador baix
B11	Bool	%i2.1	Empenyador alt
B12	Bool	%i2.2	Separador baix
B13	Bool	%i2.3	Separador alt
S1	Bool	%i2.4	Paro quadre principal
S2	Bool	%i2.5	Paro entrada 1
S3	Bool	%i2.6	Paro entrada 2

Nom	Tipus de dades	Direcció	Comentari
S4	Bool	%I2.7	Marxa quadre principal
S5	Bool	%I3.0	Marxa entrada 1
S6	Bool	%I3.1	Marxa entrada 2
S7	Bool	%I3.2	Porta 1
S8	Bool	%I3.3	Porta 2
QM1	Bool	%I3.4	Contacte auxiliar guardamotor M1
QM2	Bool	%I3.5	Contacte auxiliar guardamotor M2
QM3	Bool	%I3.6	Contacte auxiliar guardamotor M3
QM4	Bool	%I3.7	Contacte auxiliar guardamotor M4
QM5	Bool	%I4.0	Contacte auxiliar guardamotor M5
Palet1_Dipositat_Robot	Bool	%I4.1	Senyal robot que indica que ha dipositat un palet
Palet2_Dipositat_Robot	Bool	%I4.2	Senyal robot que indica que ha dipositat un palet
Palet1_Complet_Robot	Bool	%I4.3	Senyal robot que indica que ha completat un palet
Palet2_Complet_Robot	Bool	%I4.4	Senyal robot que indica que ha completat un palet
Barrera_Palet1	Bool	%I4.5	Barrera fotoelectrica Palet 1
Barrera_Palet2	Bool	%I4.6	Barrera fotoelectrica Palet 2
-	Bool	%I4.7	Reserva

Figura 3 . Entrades

En la figura 4, es pot observar les sortides del PLC utilitzades en el projecte.

Nom	Tipus de dades	Direcció	Comentari
KM1	Bool	%Q0.0	Buffer caixes
KM2	Bool	%Q0.1	Entrada caixes corròns
KM3A	Bool	%Q0.2	Palet format 1 entrada
KM3B	Bool	%Q0.3	Palet format 1 Sortida
KM4A	Bool	%Q0.4	Palet format 2 entrada
KM4B	Bool	%Q0.5	Palet format 2 Sortida
KM5	Bool	%Q0.6	Buffer pila palets
H1	Bool	%Q0.7	Llum verda Marxa
H2	Bool	%Q1.0	Llum vermella Paro
H3	Bool	%Q1.1	Llum Groc Alarma
Y1	Bool	%Q2.0	Empanyedor
Y2	Bool	%Q2.1	Separador
-	Bool	%Q2.2	Reserva
-	Bool	%Q2.3	Reserva
-	Bool	%Q2.4	Reserva
-	Bool	%Q2.5	Reserva
-	Bool	%Q2.6	Reserva
-	Bool	%Q2.7	Reserva
-	Bool	%Q3.0	Reserva
-	Bool	%Q3.1	Reserva
-	Bool	%Q3.2	Reserva
-	Bool	%Q3.3	Reserva
-	Bool	%Q3.4	Reserva
-	Bool	%Q3.5	Reserva
-	Bool	%Q3.6	Reserva
-	Bool	%Q3.7	Reserva

Figura 4. Sortides

## 4.2. Procés de paletització

En aquest subapartat s'explicarà amb detall el procés de treball de cada màquina involucrada en el procés. S'utilitzaran els mateixos noms que es mostren a la Figura 1. Per iniciar el procés de producció, és imprescindible que els polsadors d'emergència i les portes de la instal·lació estiguin rearmades i que el contactor K1 estigui activat.

Totes aquestes condicions s'estableixen segons la guia GEMMA que es mostra a la Figura 5. Aquesta guia gràfica descriu els estats de producció, parada i de defecte i proporciona una aproximació funcional d'un procés automatitzat. La guia GEMMA és d'una gran utilitat per coordinar el funcionament dels diferents GRAFCET's en la instal·lació i per abordar diverses circumstàncies que es poden presentar durant el funcionament.

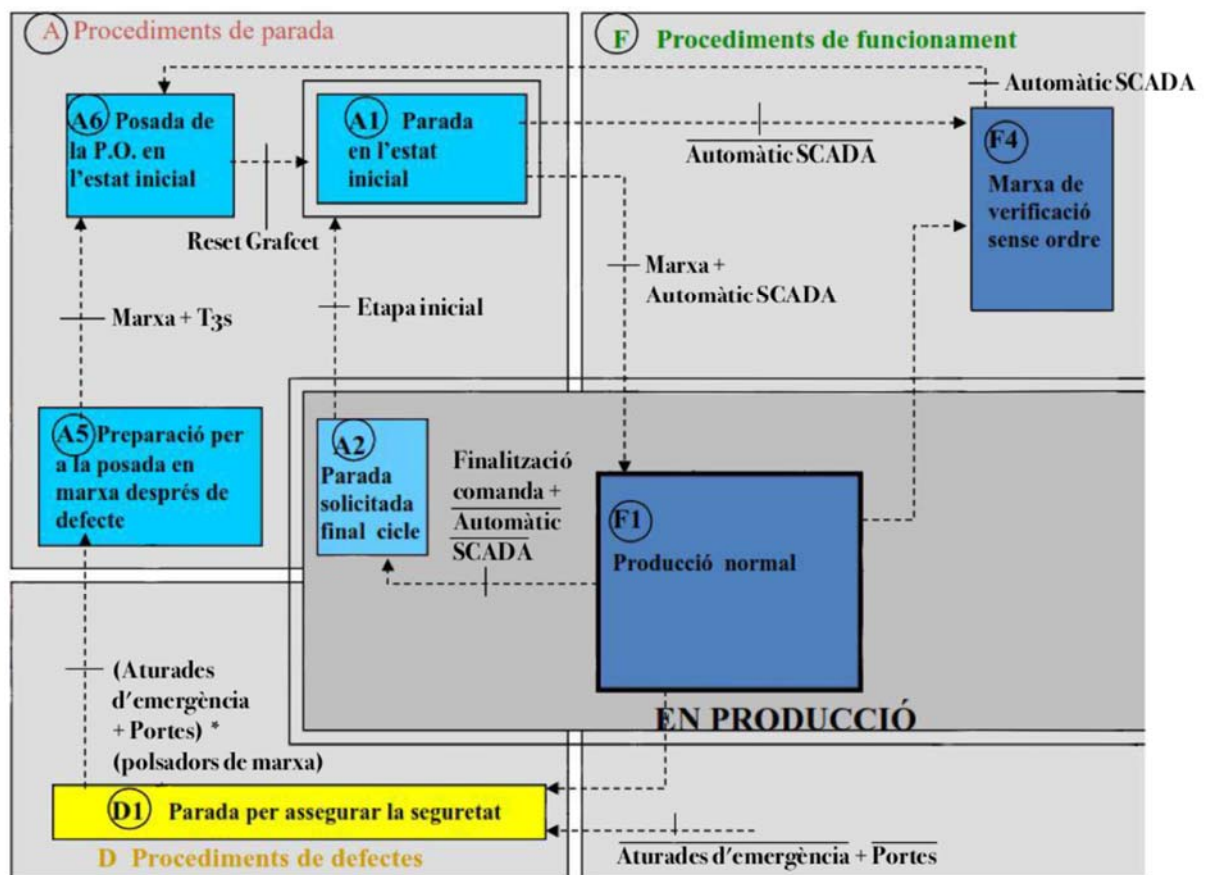


Figura 5. Guia GEMMA

#### 4.2.1 GRAFCET de conducció

En el següent apartat, s'explorarà el GRAFCET de conducció, visible a la figura 6, que defineix el comportament del sistema. El procés comença a l'estat A1, que representa la parada inicial. Quan es produeixen situacions com prémer algun polsador d'inici i tenir seleccionada l'opció d'automàtic a l'SCADA, el sistema passa a l'estat 101, que és F1, la producció normal. Després, si es completa l'ordre en curs, s'activa l'estat 102, A2, la parada sol·licitada al final del cicle. També es pot desseleccionar l'opció d'automàtic a l'SCADA, i en aquest cas, el sistema passa a l'estat 104.

A l'estat 100, si no es té seleccionada l'opció d'automàtic a l'SCADA, el sistema es mou a l'estat 104, que és F4, els moviments en mode manual. Per sortir d'aquest estat, cal seleccionar l'opció d'automàtic a l'SCADA. Si en qualsevol moment l'estat 110 deixa d'estar actiu, el sistema passa a l'estat 100 i no es pot iniciar la instal·lació fins que l'estat 110 torni a ser actiu.

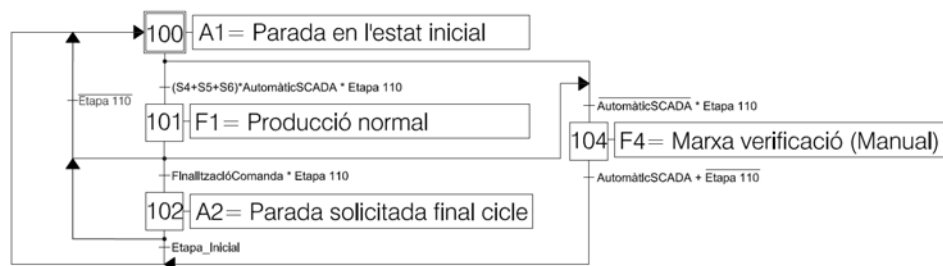


Figura 6. GRAFCET conducció

#### 4.2.2 GRAFCET de seguretat

En un GRAFCET de seguretat, s'empra per descriure el comportament seqüencial d'un sistema automàtic amb relació als requisits de seguretat, com es mostra a la figura 7. El procés s'inicia amb l'etapa 110, que representa l'estat "seguretats OK". Si s'activa alguna mesura de seguretat com polsadors d'emergència, portes o barreres fotoelèctriques, D1 s'actua per aturar la instal·lació i garantir la seguretat. Quan tots els elements de seguretat es troben en l'estat adequat i es prems algun botó d'inici, es passa a l'etapa 113, A5, la preparació per a la posada en marxa.

Per avançar a la següent etapa, cal prémer qualsevol polsador d'inici durant 3 segons. Un cop transcorreguts aquests segons, s'actua l'etapa 114, A6, que implica la restauració de la resta del GRAFCET a l'estat inicial.

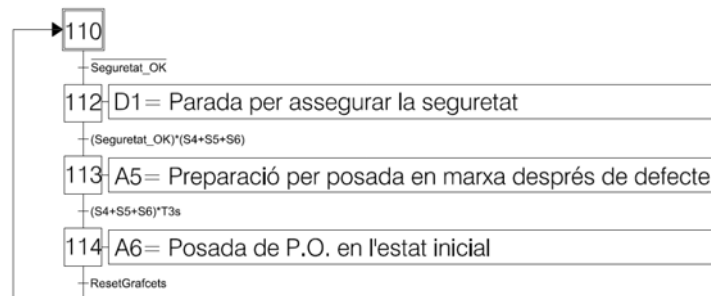


Figura 7. GRAFCET de seguretat

#### 4.2.3 Buffer caixes

La posada en marxa de la cinta transportadora del buffer de caixes es produeix quan concorren les següents condicions: l'estat F1 del GRAFCET de conducció es troba actiu, el sensor B1 detecta la presència de caixes i la memòria de marxa es troba activada. Un cop ha transcorregut un temps raonable i en absència de detecció de noves caixes en el procés, la cinta es deté. Aquesta aturada té lloc sempre que el pistó empenyedador es trobi en la posició adequada, com es detalla a la figura 8.

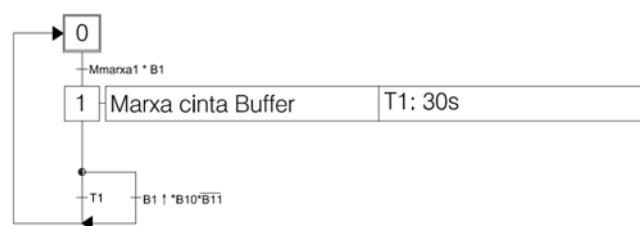


Figura 8. GRAFCET cinta Buffer

#### 4.2.4 Empenyedor

L'empenyedor és un pistó pneumàtic que canvia la direcció de les caixes. Requereix que l'estat F1 del GRAFCET de conducció sigui actiu, la fotocèl·lula B2 detecti una caixa i el pistó estigui a la posició inicial B10, com mostra la figura 9. Quan es detecta el sensor B11, l'empenyedor torna a la posició inicial.



Figura 9. GRAFCET Empenyedor

#### 4.2.5 Transport de corrns

El transportador de corrns funcionarà com a transportador de caixes des de l'empenyedor fins a la zona d'agafada del robot paletitzador. Per a la seva activació, és necessari que l'estat F1 del GRAFCET de conducció estigui actiu i que es rebí un flanc de pujada del sensor B11. El transport s'aturarà si no es detecta l'entrada de caixes durant un temps determinat pel temporitzador T2. Si es detecta l'entrada d'una caixa (B2) i un flanc de pujada de B11, el temporitzador es reiniciarà, segons es mostra a la figura 10.

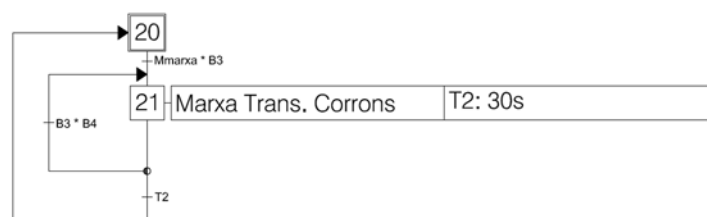


Figura 10. GRAFCET Transport corrns

#### 4.2.6 Separador

La funció d'aquest element és separar les caixes de la zona d'agafada del manipulador de la zona de transport. Amb els programes existents, el robot pot agafar només una caixa de cada vegada, per la qual cosa la seva funció principal és separar les caixes de manera individual.

Perquè aquesta zona de separació funcioni, les condicions següents han de ser satisfetes: l'estat F1 del GRAFCET de conducció ha de ser actiu, el sensor B4 ha de detectar una caixa, el separador ha d'estar en posició elevada (B13) i no pot haver cap caixa a B5. Quan aquestes condicions es compleixen, el separador pot ser abaixat per permetre l'entrada d'una caixa. Després que hagi passat el temps establert a T3 i quan la senyal indiqui que el separador està a la posició inferior (B12), el separador tornarà a la seva posició inicial, és a dir, elevada, com es mostra a la figura 11.

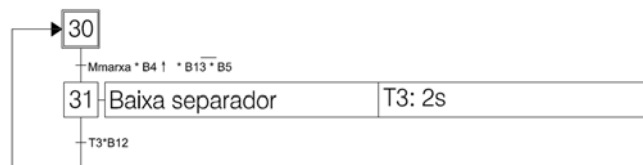


Figura 11. GRAFCET Separador

#### 4.2.7 Transport corròns palet entrada

En aquest procés, el funcionament dels transportadors de palets 1 i 2 serà idèntic, ja que s'han duplicat per millorar l'eficiència de la producció.

Quan l'estat F1 del GRAFCET de conducció estigui actiu i rebem la senyal del robot PaletDipositat1 o PaletDipositat2, activarem el transportador corresponent per a posicionar adequadament el palet i preparar-lo per a la càrrega de caixes. Aquest transportador es detindrà quan el sensor de presència de palets B6 o B7 sigui detectat, segons es mostra a les figures 12 i 13.





Figura 12. GRAFCET Transport corrone palet entrada 1



Figura 13. GRAFCET Transport corrone palet entrada 2

#### 4.2.8 Transport corrone palet sortida

Aquest procés també es troba duplicat com en el cas anterior, ja que involucra la mateixa maquinària per dur a terme un procés de sortida de palets.

Per activar aquest accionament, és necessari que l'estat F1 del GRAFCET de conducció estigui actiu i que es rebi una senyal del robot Palet1\_Complet\_Robot o Palet2\_Complet\_Robot. Un cop rebuda aquesta senyal, posarem en marxa el transportador de corrone corresponent durant un temps determinat pel funcionament dels temporitzadors T4 i T5, tal com es mostra a les figures 14 i 15.

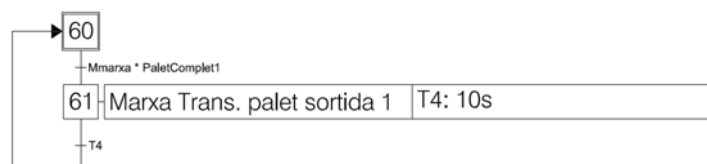


Figura 14. GRAFCET Transport corrone palet sortida 1

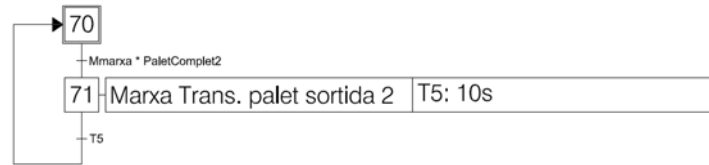


Figura 15. GRAFCET Transport corròns palet sortida 2

#### 4.2.9 Buffer pila de palets

Aquest transportador de palets té la funció d'augmentar el temps de funcionament autònom abans de la necessitat d'aturar la instal·lació per recarregar palets.

Aquest transportador té la capacitat de mantenir dues piles de palets carregades. Quan s'esgota la primera pila, el transportador s'activa per posicionar la segona pila a la zona de descàrrega de palets designada pel robot.

Aquest procés es posa en funcionament únicament quan l'estat F1 del GRAFCET de conducció està actiu, l'instal·lació està en funcionament i la fotocèl·lula B8 no detecta cap palet mentre la fotocèl·lula B9 en detecta un. En aquest cas, es posarà en marxa el transportador segons l'etapa 81 fins que el sensor B8 detecti la presència d'un palet a la zona de descàrrega.

Si no es detecta cap senyal de palet a cap de les dues fotocèl·lules, es passarà a l'etapa 82, que mostrarà un missatge d'error a la pantalla. En prémer el polsador habilitat per la pantalla HMI, s'activarà l'etapa 83, indicant que la zona d'entrada de palets ha estat recarregada. Quan es detecti el palet amb la fotocèl·lula B9, s'activarà el transportador buffer de palets, que funcionarà fins que es detecti el sensor B8, segons s'observa a la figura 16.

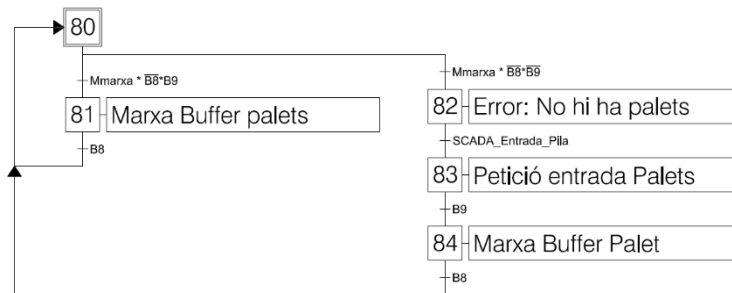


Figura 16. GRAFCET Corrons buffer pila palets

### 4.3. Programa PLC

Per portar a terme tots els GRAFCET's anteriorment explicats, s'utilitza el programa TIA PORTAL, un programari desenvolupat per SIEMENS. A continuació, s'explica el procés de programació del PLC amb TIA Portal:

En la configuració del projecte, es crea un nou projecte al TIA Portal i es configuren els paràmetres del PLC que s'utilitzarà a la instal·lació o procés automatitzat. Es tria el model específic del PLC adequat que compleixi amb els requisits de l'aplicació i tingui suficients entrades i sortides per connectar tots els dispositius necessaris. A més, es defineixen les entrades i sortides que es connectaran al PLC, relacionades amb els sensors i actuadors utilitzats. També s'estableixen les comunicacions per permetre la interacció amb altres dispositius com pantalles HMI o sensors remots.

La programació del GRAFCET i lògica de control es duu a terme amb el TIA Portal, utilitzant el diagrama de contactes (KOP) i altres llenguatges de programació disponibles. Es programen tots els GRAFCET's desenvolupats anteriorment i es realitza la manipulació i tractament de les diferents dades generades durant el funcionament del programa. Durant aquesta etapa, s'implementen les accions de control que el PLC realitzarà a cada estat o etapa del GRAFCET. Aquestes accions poden incloure l'activació o desactivació de sortides, el control de la velocitat dels motors i la lectura de sensors, entre altres tasques. La lògica programada ha de garantir un funcionament segur i eficient del procés automatitzat.

Finalment, es verifica i simula el funcionament del PLC en un entorn virtual abans de la seva implementació física a la instal·lació. Això permet detectar i corregir possibles errors abans de la posada en marxa del sistema. La simulació és una fase crucial per assegurar el correcte funcionament de tot el procés automatitzat.

A continuació, es detallaran exhaustivament tots els passos explicats anteriorment.

#### 4.3.1 Configuració del projecte

Començarem escollint els paràmetres del PLC descrits en apartats anteriors com el tipus de CPU escollida i mòduls d'entrades i sortides tal com es pot observar en la figura 17.

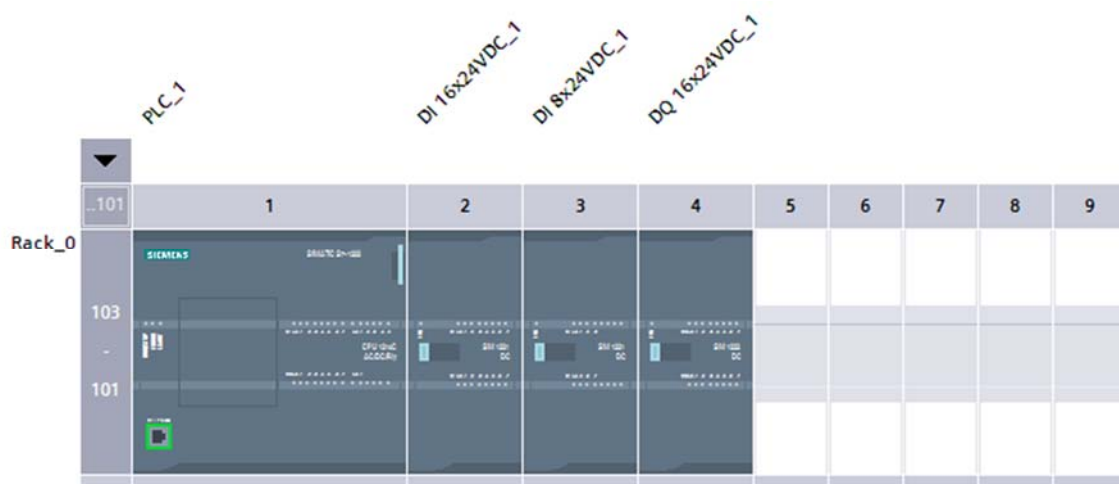


Figura 17. Vista general del PLC

També escollirem la pantalla HMI que utilitzarem per realitzar SCADA del projecte i escollirem la comunicació que tindran els elements del projecte que serà PROFINET tal i com es pot observar en la figura 18.

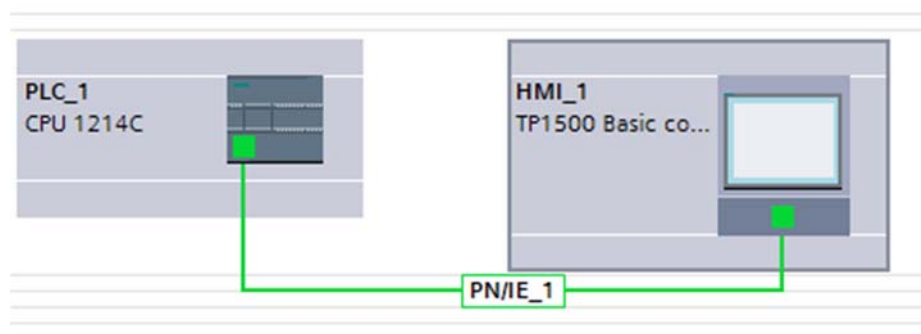


Figura 18. Vista de xarxes

### 4.3.2 Programació GRAFCET i lògica de control

Per plasmar els diversos GRAFCET en el programa, s'han creat blocs de funcions separats per a cada GRAFCET amb l'objectiu de mantenir un millor ordre, tal com es mostra a la figura 19. A més, s'ha generat un bloc de dades per a la transmissió bidireccional de la informació entre la pantalla HMI i el PLC.



Figura 19. Blocs de programa

El bloc de funció "Main [OB1]" és un element essencial en la programació del PLC. És un bloc que s'executa cíclicament i conté la lògica de control principal del sistema automatitzat. En altres paraules, és el nucli central de la lògica de funcionament de tot el programa.

En el bloc de funcionament "Main [OB1]" s'han integrat tots els blocs de funcions creats anteriorment, com també s'han establert les condicions principals perquè aquests blocs funcionin adequadament, com ara la comprovació de l'estat actiu de l'estat F1 del GRAFCET de conducció. Aquesta configuració és fonamental per garantir una operació coordinada i segura de tot el procés, tal com es pot observar a la figura 20.

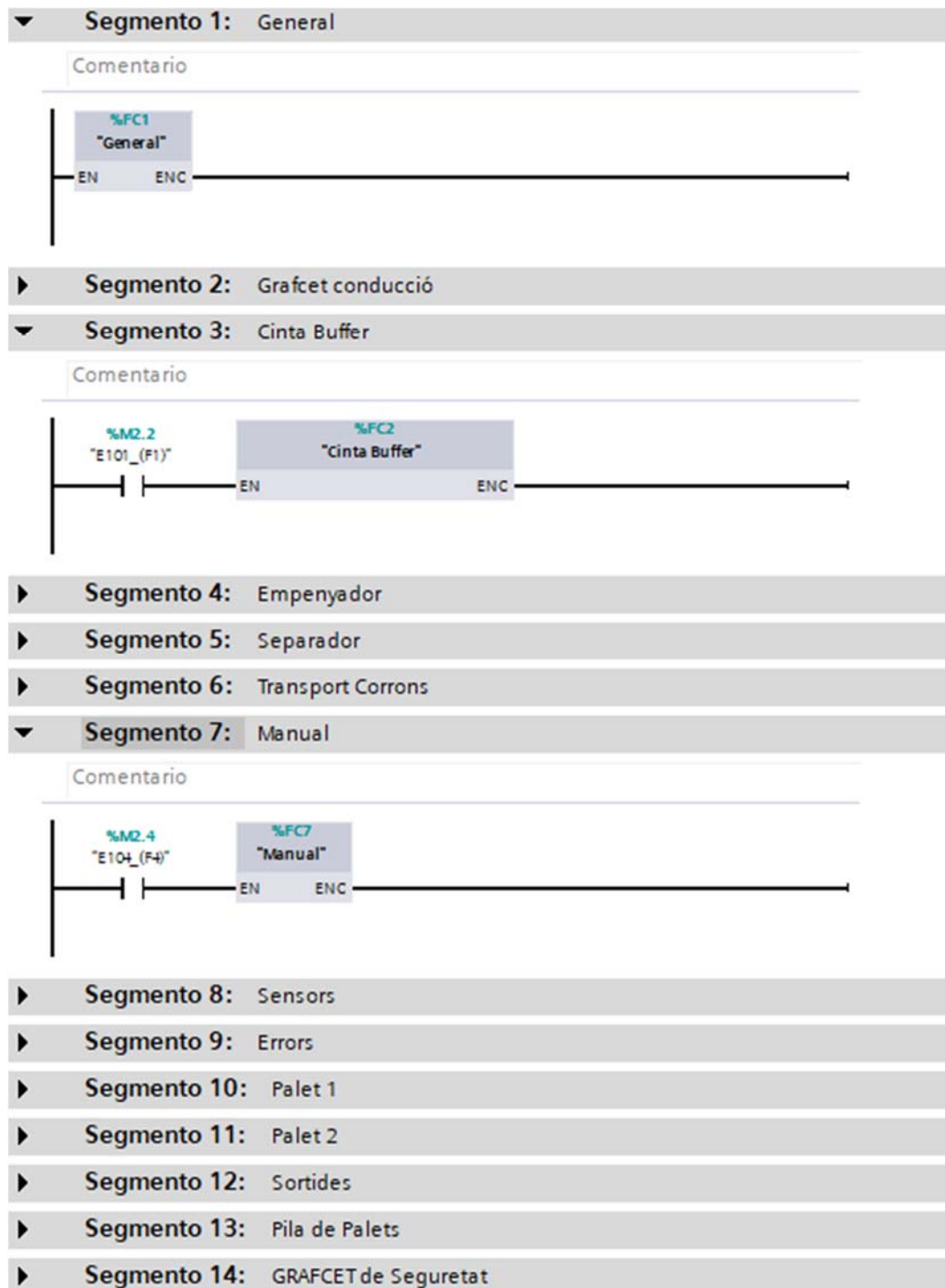


Figura 20. Main [OB1]

La tècnica de programació emprada és la dels blocs de funcions SET i RESET (S/R), que permeten implementar un control seqüencial i ordenat tal com es pot observar a la figura 21. Amb aquesta tècnica, s'utilitzen relés virtuals "SET" per activar condicions específiques que indiquen l'entrada en un determinat estat o acció. A més, es fan servir relés virtuals "RESET" per sortir d'aquests estats o aturar les accions en curs. D'aquesta manera, es pot gestionar eficientment el control del sistema i establir un flux de treball estructurat i fàcil de mantenir.

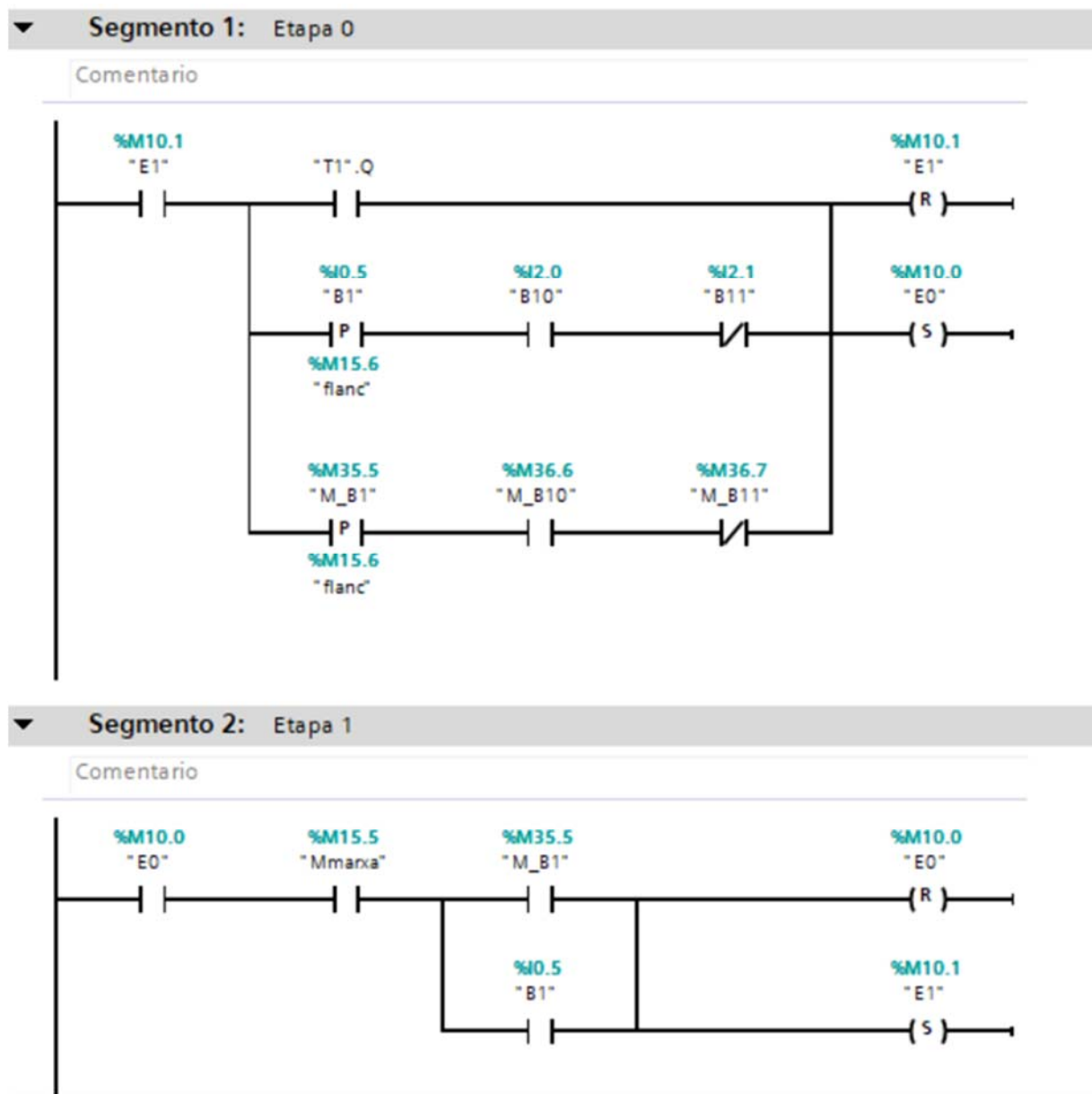


Figura 21. Mètode de programació

### 4.3.3 Simulació

En el desenvolupament del sistema d'automatització, es realitzarà una simulació completa del programa del PLC i la pantalla HMI utilitzant la plataforma PLCSIM. A través d'aquest programa, es podrà emular el comportament del controlador programat i interactuar amb la interfície gràfica de l'HMI com si s'estigués treballant amb el sistema en temps real. Això permetrà provar el funcionament de tots els GRAFCET's, les comunicacions, les funcions de control i altres aspectes crítics del sistema sense la necessitat de connectar físicament el PLC i la pantalla HMI a l'equipament real, oferint una manera eficient i segura de verificar i validar el comportament del sistema abans de la seva implementació efectiva.

Per garantir la fiabilitat i l'eficàcia del sistema, es realitzarà una sèrie de vídeos demostratius. Aquests vídeos mostraran el correcte funcionament del programa del PLC i la pantalla HMI en la simulació amb PLCSIM. La demostració estarà basada en l'actuació sobre unes memòries que simulen els sensors i polsadors que s'utilitzen a la instal·lació real. A través d'aquests vídeos, es podrà observar com els GRAFCET's es comporten segons les diferents condicions de funcionament, i com el programa del PLC reacciona i controla adequadament els diferents dispositius i etapes del procés automatitzat. Aquesta validació mitjançant vídeos proporcionarà una prova tangible de la precisió i efectivitat del sistema abans de la posada en marxa definitiva. Això assegurarà que el sistema compleixi amb els requisits establerts i estigui llest per a la seva implementació en un entorn real de producció.



#### 4.4. Pantalles HMI

En el següent apartat, es descriuran detalladament les diferents pantalles generades per facilitar la interacció de l'operari amb el procés. S'han creat un total de 5 pantalles, anomenades "General", "Manual", "Alarmes", "Receptes" i "Control d'usuaris". Cadascuna d'aquestes pantalles ofereix diverses funcionalitats i informació útil per a l'operari:

##### 4.5.1 Pantalla General

En la pantalla General es pot observar l'estat complet de la instal·lació, així com accedir a totes les altres pantalles de l'aplicació, tal com es mostra a la figura 22. Els sensors distribuïts per tota la instal·lació apareixeran representats amb cercles verds per indicar que estan detectant objectes. Aquesta característica és d'ajuda als operaris per identificar ràpidament possibles problemes i localitzar la causa d'errors. En cas que hi hagi algun polsador d'emergència accionat, també es mostrarà a la interfície gràfica SCADA.

A més, s'han implementat fletxes que indiquen si els transportadors estan en funcionament i en quina direcció es desplacen. A la finestra inferior es mostra informació sobre les alarmes actives, acompanyades de les seves respectives descripcions.

El botó "RESET" té la funció de reiniciar tots els processos que intervenen en la instal·lació en cas de problemes o situacions imprevistes. Amb el botó "USUARIS", es permet iniciar sessió amb els diferents usuaris de l'aplicació. Hi ha 3 usuaris creats a l'aplicació amb nivells d'accés específics: Administrador amb accés total, Manteniment amb accés a tot excepte a canviar contrasenyes, i Encarregat de Producció no tindrà accés per manipular la pantalla de manual. Les contrasenyes per als diferents usuaris es detallaran posteriorment.

El botó "MANUAL" permet accionar manualment tots els actuadors de l'aplicació i visualitzar l'estat actual de totes les entrades del sistema. Tot seguit, es troba el botó "ALARMES" que permet accedir a la pantalla d'alarmes, on es poden visualitzar totes les alarmes actives en aquell moment.

Just sota d'aquest botó, es troba "RECEPTES", que permet editar les receptes del sistema. A la mateixa finestra també es troba el botó "AUTOMÀTIC" per poder posar la instal·lació en

mode automàtic, i finalment, hi ha un botó per indicar que s'està entrant una pila de palets a la instal·lació.

Per a una millor supervisió del procés, es disposa d'un parell d'indicadors que mostren la quantitat actual de caixes i palets en el sistema en aquell instant. Aquesta informació facilita un seguiment eficient del progrés de l'automatització i dels recursos utilitzats.

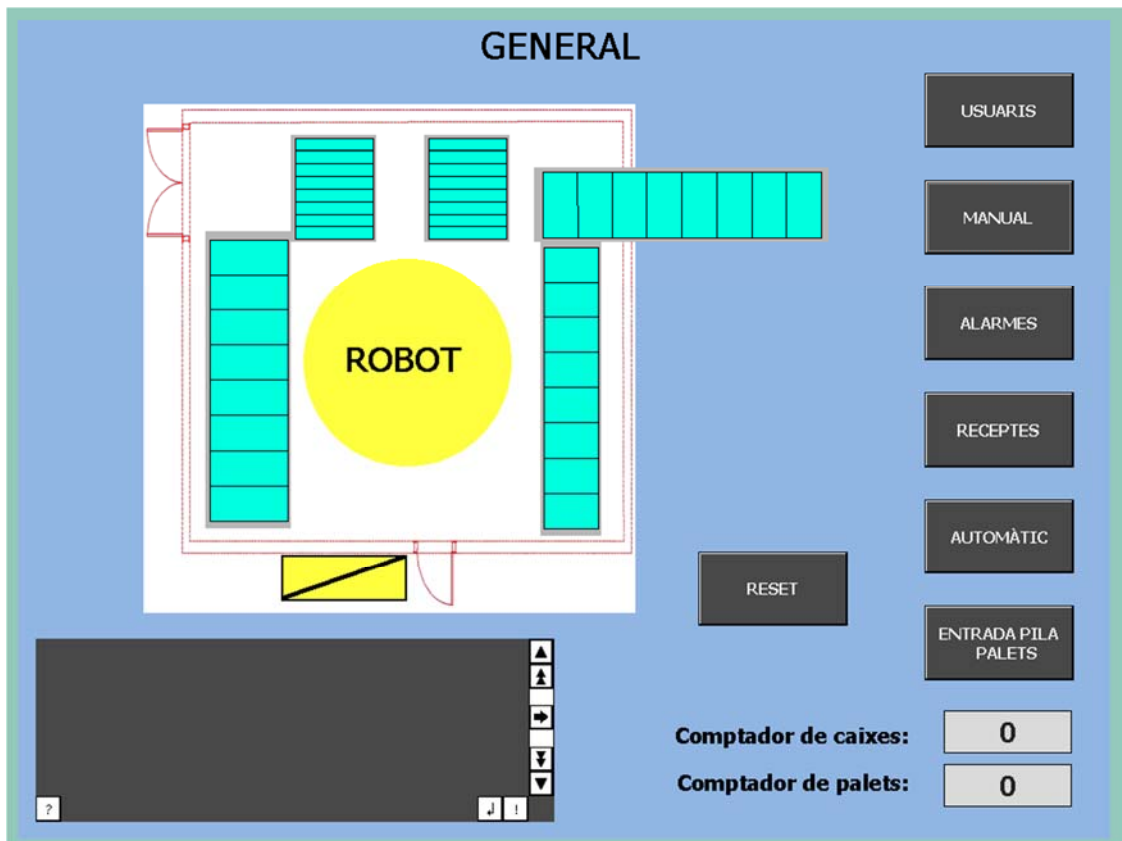


Figura 22. Pantalla General

#### 4.5.2 Pantalla d'usuari

En aquesta pantalla, tal com es pot observar a la figura 23, es pot iniciar sessió amb els diferents usuaris creats. S'han implementat restriccions d'accés a diverses pantalles per als usuaris no autenticats, això es fa per evitar possibles errors futurs i assegurar que només els operaris amb els coneixements adequats i els permisos necessaris puguin accedir i manipular determinades funcionalitats de la instal·lació.

Aquestes mesures de seguretat són essencials per garantir un funcionament adequat i evitar possibles errors que podrien afectar el rendiment del sistema. Per raons de seguretat, si la sessió de l'usuari roman inactiva durant 5 minuts, el sistema tancarà automàticament la sessió i serà necessari iniciar-la de nou. Aquesta funcionalitat addicional augmenta la protecció de les dades i la privadesa dels usuaris, evitant l'accés no autoritzat o incontrolat a la informació i el control del sistema.



Figura 23. Pantalla Usuaris

A la figura 24 es mostren els detalls dels 3 usuaris del sistema:

Usuaris	Contrasenya
Administrador	admin
Mant	mant
Enc_Prod	prod

Figura 24. Taula usuaris-contrasenya

Administrador: Té accés a totes les pantalles i té la capacitat de modificar i crear nous usuaris i contrasenyes.

Manteniment (Mant): Aquest usuari té accés a totes les pestanyes de l'aplicació, però no té permís per canviar les contrasenyes dels altres usuaris.

Encarregat de Producció: No té accés a la pantalla de "Manual".

L'accés no autenticat només permet veure la pantalla general i les alarmes. Per accedir a les altres pantalles, és necessari autenticar-se amb les credencials adequades.

Els botons situats a la dreta serveixen per a navegar entre les diferents pantalles de l'aplicació. A més, el botó inferior dret amb forma de casa té la funcionalitat de retornar a la pantalla general. Aquesta organització de les pantalles i els permisos d'usuari ofereix un control eficient i segur sobre les funcionalitats de l'aplicació, garantint que només els usuaris autoritzats tinguin accés a les funcions específiques.

#### 4.5.3 Pantalla manual

En la pantalla "Manual" es permet activar i desactivar els actuadors de la instal·lació, com es pot observar a la figura 25. Quan es prem el botó del corresponent actuator, aquest s'activarà si es tracta d'un transportador o canviarà de posició si es tracta de pistons pneumàtics.

A més, en aquesta pantalla s'ha habilitat una zona de visualització de les senyals dels sensors. Si un sensor està detectant algun objecte, es mostrarà en color verd; en cas contrari, el color serà vermell.

Per canviar del mode "Manual" al mode "Automàtic", es disposa del botó corresponent. Per a poder actuar sobre els actuadors, és necessari tenir el mode "Manual" activat, el qual es mostra en color negre. Aquesta disposició permet controlar amb precisió els actuadors i garantir un funcionament òptim del procés automatitzat.

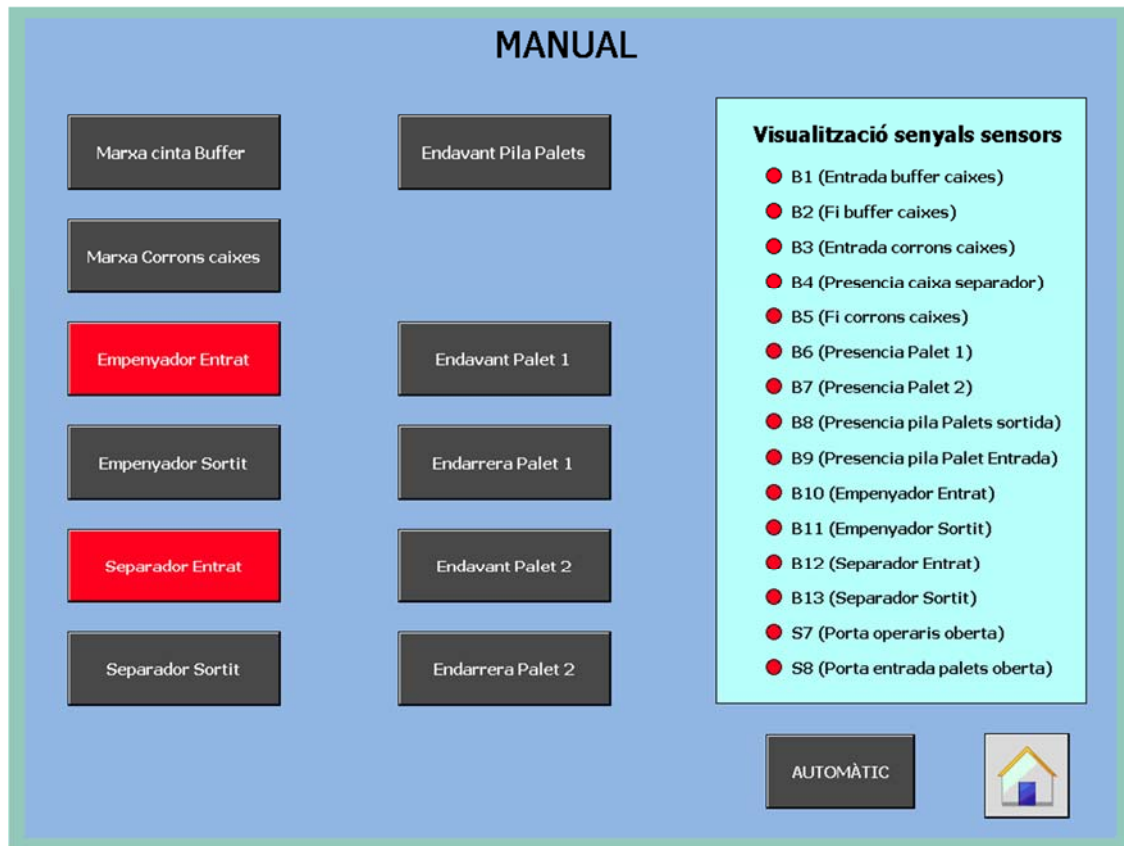


Figura 25. Pantalla Manual

#### 4.5.4 Pantalla alarmes

La pantalla d'alarmes té la funció de visualitzar amb més detall les alarmes actives en el moment actual. A més, s'han incorporat accessos directes a les diferents pantalles per agilitzar el procés de canvi entre les interfícies i poder resoldre els problemes de manera eficient, tal com es pot observar a la figura 26. Aquesta disposició permet una gestió més eficaç de les alarmes i una resposta ràpida i precisa davant de qualsevol incidència que pugui sorgir en el sistema automatitzat.

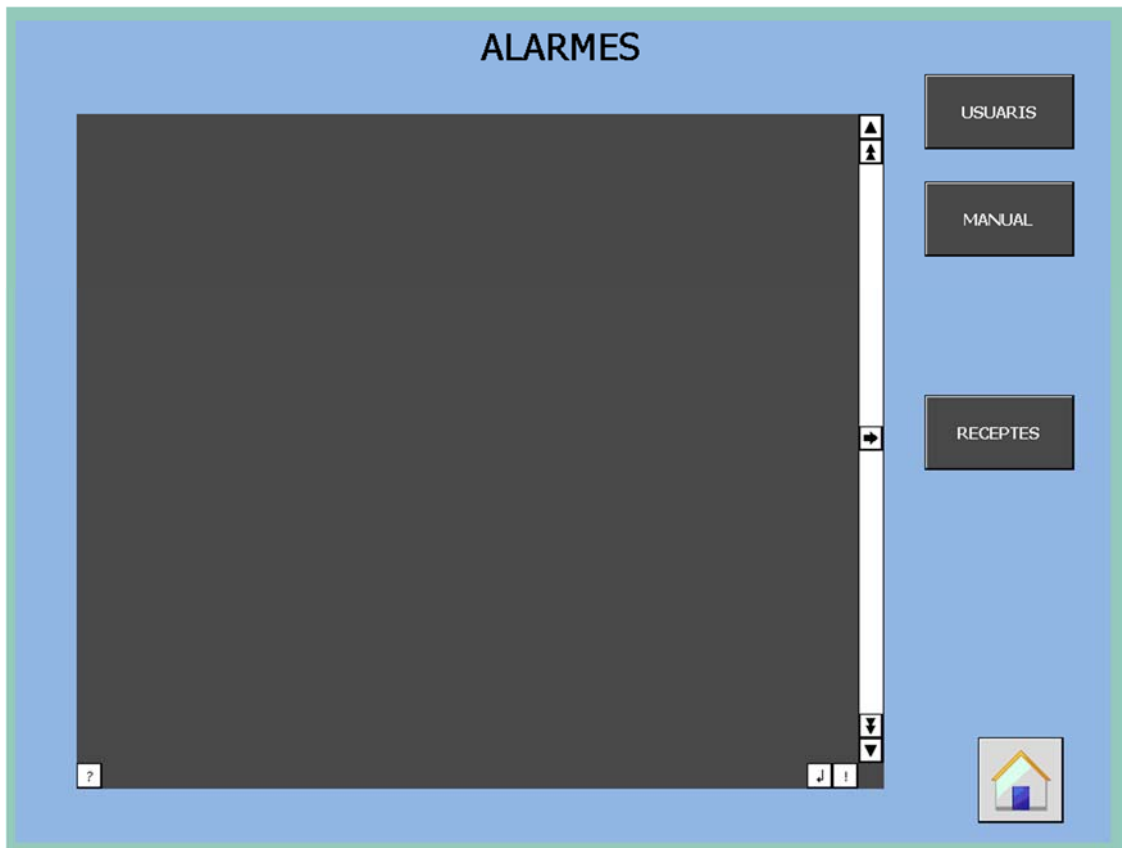


Figura 26. Pantalla Alarmes

Les possibles alarmes que poden aparèixer a la pantalla HMI es troben especificades a la figura 27, la qual il·lustra el contingut que s'afegirà a la finestra emergent en cas d'alarma.

Text d'avís
Guardamotor desconnectat
Polsador emergència quadre principal accionat
Polsador emergència porta 1 accionat
Polsador emergència porta 2 accionat
Porta 1 oberta
Porta 2 oberta

Figura 27. Alarmes

#### 4.5.5 Pantalla receptes

En la pantalla receptes que podem observar en la figura 28 podem escollir el tipus de caixa que tenim entre gran, mitjana i petita. Un cop escollida la caixa ens trobarem que cada tipus de caixa te estandarditzats el número de caixes per pis, número de pisos i les dimensions de la caixa.

Per últim s'han creat unes alarmes de comptador per tal que quan arribem el número de palets i o caixes que tenim en la comanda ens ho indiqui per pantalla i ens pari la instal·lació. Per poder posar a 0 els comptadors s'ha generat un boto de RESET de comptadors.

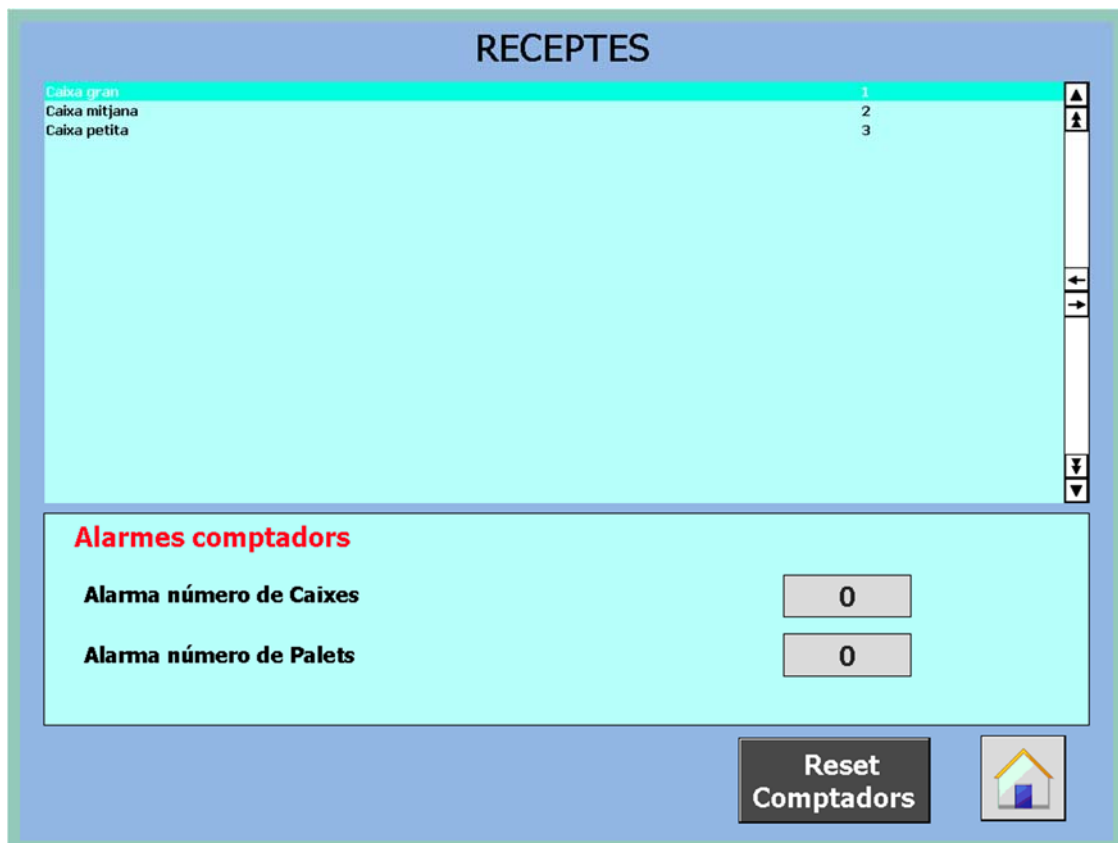


Figura 28. Pantalla Alarmes

#### 4.5. Simulació amb el programari Factory I/O

Per portar a terme la simulació del projecte s'ha fet amb el programa Factory I/O, que és un programa de simulació 3D per a sistemes de control industrial. S'utilitza per crear i provar sistemes de control industrial sense necessitat de dispositius físics. Factory I/O és un programa potent que pot ser utilitzat per a una àmplia gamma de tasques.

Aquest programa permet enllaçar el PLCSIM que és el programa amb el qual simulem el PLC amb els elements de la instal·lació amb 3D. També ens permet visualitzar si el sistema que em pensat per la instal·lació funcionarà i també hem és una gran eina per vendre el producte als clients, ja que permet ensenyar el projecte funcionant. Degut a tenir una versió gratuïta del programa no s'han pogut enllaçar els dos programes.

En les figures 29 a la 34 mostrarem el projecte realitzat i s'adjuntaran uns enllaç de funcionament de la instal·lació. Cal destacar que amb la llicència gratuïta no tens accés per poder posar el robot antropomòrfic de 6 eixos com el que hem utilitzat en la nostra projecte. Pel que s'ensenyarà el projecte funcionant per parts. Una serà la zona d'entrada de caixes, l'altre l'entrada de la pila de palets i per acabar es mostrarà l'entrada de palet carregat pel robot i la sortida del palet un cop completat.



Figura 29. Vista general (Sortida Palets)



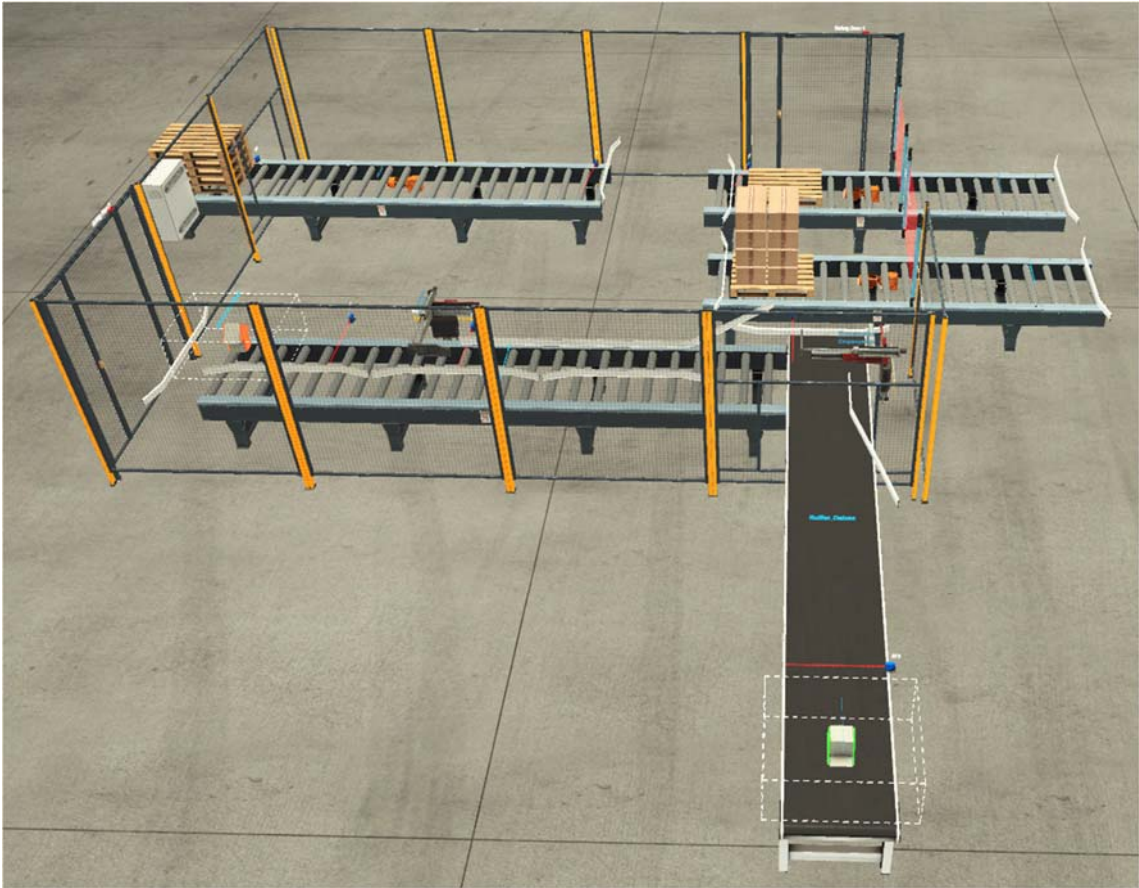


Figura 30. Vista general (Entrada caixes)



Figura 31. Vista general (Entrada pila palets)

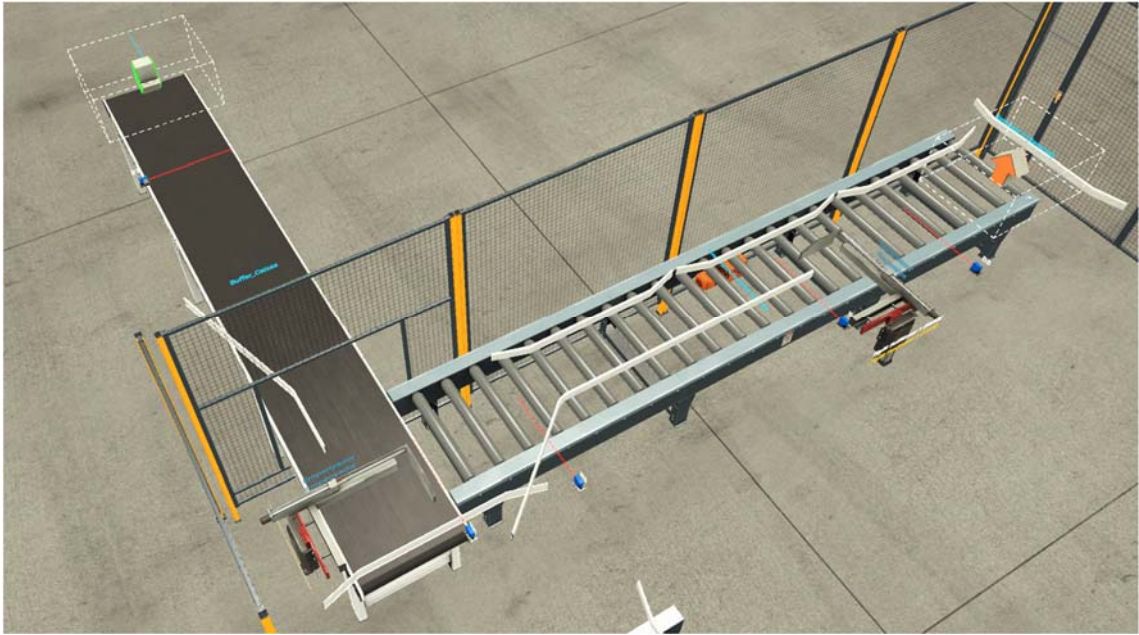


Figura 32. Vista entrada caixes

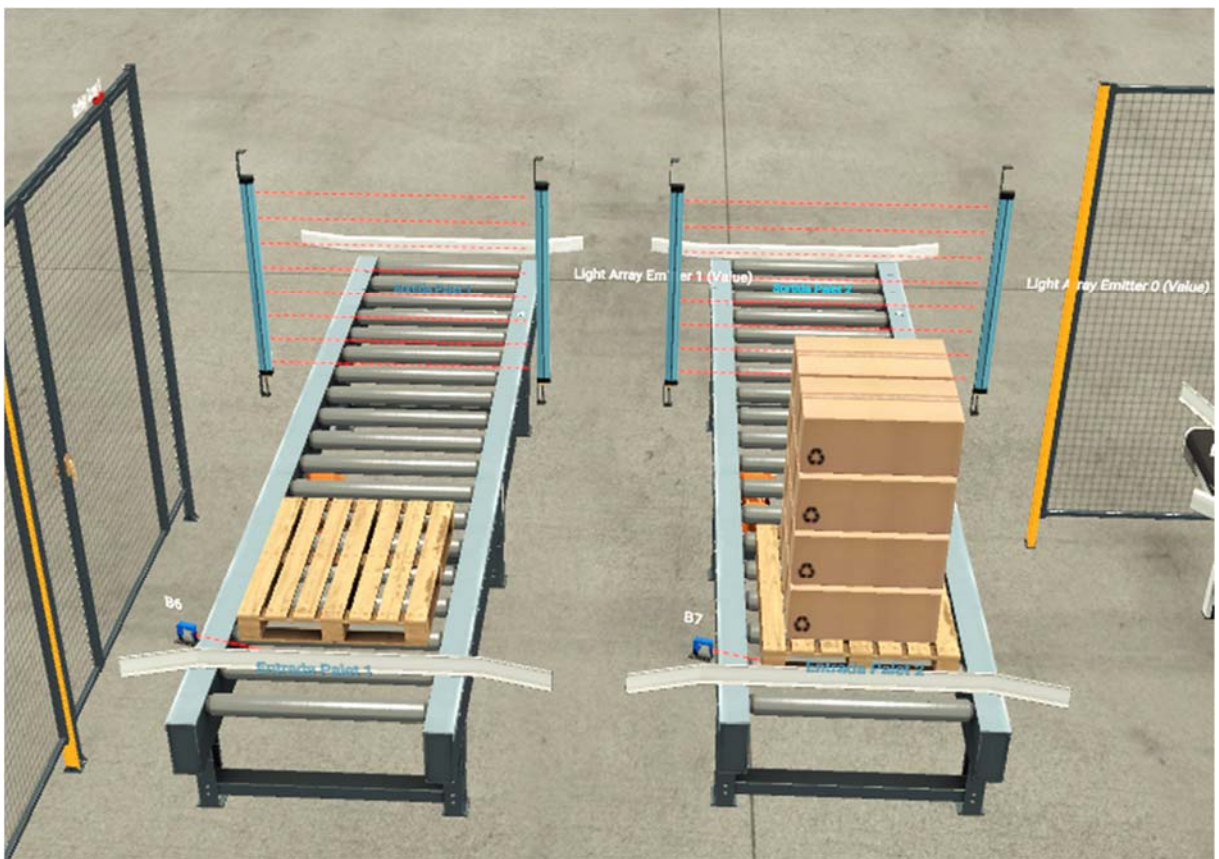


Figura 33. Vista sortida palets





Figura 34. Vista Entrada pila palets

#### 4.6. Programació Robot Staubli

En aquest apartat, abordarem comandes i instruccions clau per al moviment, interacció i realització de diverses funcions amb el robot, també s'explicarà com ens relacionem amb el PLC de la instal·lació i quines solucions s'han portat a terme pels diferents problemes que han anat sorgint.

Cal destacar que, en aquesta ocasió, el programa no ha estat desenvolupat utilitzant l'entorn STAUBLI ROBOTICS SUITE a causa de la manca de llicència. En comptes d'això, s'ha utilitzat l'eina de desenvolupament Visual Studio Code (VS CODE) per crear el codi necessari. A causa d'aquesta problemàtica no s'ha pogut simular.

Durant el procés de programació, hem generat dos conjunts de codi: un destinat a iniciar (START) el programa i un altre per aturar-lo (STOP). En les següents línies, detallarem i explicarem les diferents línies de codi utilitzades, així com les seves funcions i objectius específics. Ens centrarem en les comandes rellevants per al control del robot, incloent-hi instruccions per al moviment, la recollida i deixada d'objectes, a més d'altres funcions útils per a la interacció amb l'entorn. Aquesta guia tindrà com a objectiu proporcionar una comprensió clara i pràctica de la programació del robot Staubli.

#### 4.2.1 Tipologia de mà

El robot ha estat equipat amb dues eines específiques per a les seves funcions. La primera eina és dissenyada per a la recollida de caixes, proporcionant una capacitat òptima per a la manipulació segura i eficient d'aquestes peces. La segona eina, d'altra banda, està especialment concebuda per a la captura de palets europeus, permetent al robot interactuar amb aquest tipus d'estructures per a la seva manipulació i desplaçament. Aquestes dues eines proporcionen al robot la versatilitat necessària per afrontar diferents tasques i situacions en l'entorn industrial amb precisió i eficàcia.

Cal destacar que tant la primera eina com la segona estaran accionades per pistons pneumàtics, els quals permetran obrir i tancar cada eina de manera controlada i eficient. D'aquesta manera, el robot podrà adaptar-se a les necessitats específiques de cada situació, garantint un rendiment òptim en les seves tasques d'agafament i manipulació. La utilització de pistons pneumàtics ofereix un funcionament ràpid i fiable, permetent una interacció precisa i segura amb les caixes i els palets europeus. Tot seguit adjuntarem algunes imatges de com s'han dissenyat les eines per manipular les caixes i els palets.

Amb els cercles marcats en vermell a la Figura 35, podem observar l'eina destinada a la recollida de palets. Aquesta eina consta de 2 pistons accionats per una sola electrovàlvula de simple efecte. Quan la vàlvula no està alimentada, els braços es troben recollits, i aquesta posició no afecta el procés de recollida de caixes. Però, quan accionem la electrovàlvula, els braços es tanquen, i en cas de trobar un palet a la seva trajectòria, les terminacions dels braços permeten subjectar el palet de manera segura, evitant qualsevol risc de caiguda. A la imatge, podem observar els braços en estat de recollida del palet. A més a més, per detectar el palet, utilitzarem un sensor regulable que ens permetrà ajustar amb precisió la detecció dels palets i assegurar un funcionament òptim del sistema.

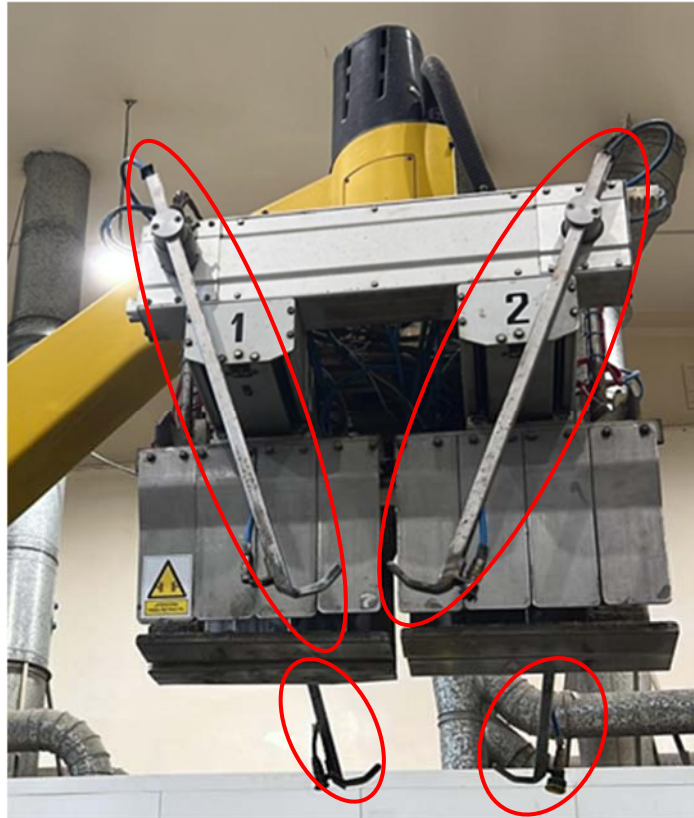


Figura 35. Eina palet

A les Figures 36 i 37 es pot observar un sistema de dos pistons lineals solidaris, els quals són accionats per una sola electrovàlvula. Aquest mecanisme permet l'obertura i tancament de les làmines que són utilitzades per recollir i dipositar les caixes en les zones desitjades. Les làmines en qüestió estan identificades amb cercles vermells i es troben actualment en la posició de recollida de caixes. Aquesta configuració ens proporciona un control precís per manipular amb eficàcia les caixes i realitzar les operacions desitjades de manera fiable i precisa.



Figura 36. Eina caixa

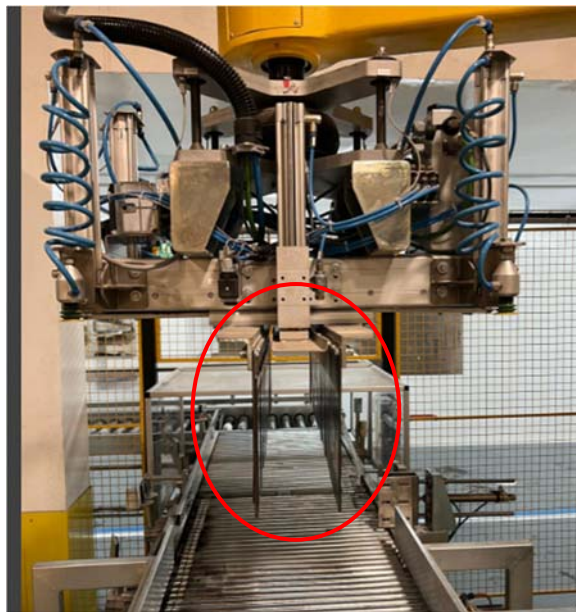


Figura 37. Eina caixa

#### 4.2.2 Comunicació entre robot i PLC

La comunicació entre el robot i el PLC es realitzarà mitjançant senyals d'entrada i sortida booleanes. Les senyals d'entrada seran generades pel PLC i enviades al robot per permetre l'activació de diverses parts del programa. D'altra banda, les senyals de sortida seran utilitzades per informar el PLC quan el robot ha completat tasques específiques o per

accionar les eines prèviament esmentades. Aquesta interacció entre el robot i el PLC mitjançant senyals booleanes ens permetrà una coordinació eficaç i un control precís de les operacions, optimitzant el funcionament global del sistema automatitzat.

Tot seguit es mostraran les figures 38 i 39 de les entrades i sortides emprades en el robot.

Direcció	Nom
In 0	B6
In 1	B7
In 2	B8
In 3	Seguretat_OK
In 4	Permis_Carrega_Caixa
In 5	Sensor_Proximitat_Palet
In 6	Programa_1
In 7	Programa_2

Figura 38. Entrades

Direcció	Nom
Out 0	Cil_Palet
Out 1	Cil_Caixa
Out 2	Palet1_Complet
Out 3	Palet2_Complet
Out 4	Palet1_Dipositat
Out 5	Palet2_Dipositat
Out 6	Reserva
Out 7	Reserva

Figura 39. Sortides

#### 4.2.3 Tipus de paletitzats

En la nostra instal·lació, tindrem tres tipus de caixes: petites, mitjanes i grans. Per tant, comptarem amb tres tipus de mosaics d'apilats diferents. En la figura 40, podem observar el mosaic per a les caixes petites, mitjanes i grans. En el qual podrem apilar 8 caixes per pis, amb un màxim de 8 pisos. Aquest mosaic s'invertirà d'una capa a l'altra per millorar l'estabilitat de la caixa.

Per les caixes mitjanes, el mosaic ens permetrà apilar 3 caixes per pis, amb un màxim de 4 pisos. Al igual que amb les caixes petites, també invertirem el mosaic d'un pis a l'altre per garantir la seguretat dels apilaments.

Finalment, per a les caixes grans, el mosaic ens permetrà apilar 2 caixes per palet, amb un màxim de 4 pisos. Aquest disseny proporciona la capacitat adequada per a la manipulació i emmagatzematge eficient de les caixes de diferents dimensions en la nostra instal·lació.



Figura 40. Tipus de paletitzats

#### 4.2.4 Codi START

En aquest apartat, s'aborden els trossos de codi principals de l'inici del programa (START). Explicarem les funcions principals que realitza cada part del codi per a una comprensió més clara del seu funcionament.

En el codi que es pot veure a continuació podem veure una part del programa de control del robot. Aquí s'està inicialitzant algunes variables i paràmetres importants:

```
mNormalSpeed.vel=50
mLentSpeed.vel=10
dx = 0
dy = 0
dz = 0
rx = 0

Palet1_Complet = false
Palet2_Complet = false
Palet1_Dipositat = false
Palet2_Dipositat = false
```



mNormalSpeed.vel=50: Estableix la velocitat de treball normal del robot al 50%. Aquesta variable representa la velocitat a la qual el robot es mou per realitzar tasques habituals amb eficiència.

mLentSpeed.vel=10: Estableix la velocitat de treball lenta del robot al 10%. Aquesta velocitat és més baixa que la normal i s'utilitza per a tasques que requereixen precisió o moviments més suaus.

dx = 0: Inicialitza la variable dx amb el valor zero. Aquesta variable està destinada a emmagatzemar un desplaçament en l'eix X del robot.

dy = 0: Inicialitza la variable dy amb el valor zero. Aquesta variable està destinada a emmagatzemar un desplaçament en l'eix Y del robot.

dz = 0: Inicialitza la variable dz amb el valor zero. Aquesta variable està destinada a emmagatzemar un desplaçament en l'eix Z del robot.

rx = 0: Inicialitza la variable rx amb el valor zero. Aquesta variable està destinada a emmagatzemar un angle de rotació del robot al voltant d'un eix específic.

També posem les variables Palet1\_Complet, Palet2\_Complet, Palet1\_Dipositat i Palet2\_Dipositat a fals.

El següent codi és una part del programa de control del robot que es dedica a la inicialització de les entrades i sortides del sistema. Les entrades i sortides estan relacionades amb els senyals d'entrada i sortida que el robot utilitza per interactuar amb els components externs del sistema, com ara sensors, actuadors i altres dispositius.

A continuació, s'explica cada línia de codi que es pot veure complert tot seguit.

```
//Inicialització Entrades
dioLink(B6,io:bIn0)
dioLink(B7,io:bIn1)
dioLink(B8,io:bIn2)
dioLink(Seguretat_OK,io:bIn3)
dioLink(Permis_Carrega_Caixa,io:bIn4)
dioLink(Sensor_Proximitat_Palet,io:bIn5)
dioLink(Programa_1, io:bIn6)
```

```

dioLink(Programa_2, io:bIn7)
//Inicialització Sortides
dioLink(Cil_Palet,io:bOut0)
dioLink(Cil_Caixa,io:bOut1)
dioLink(Palet1_Complet,io:bOut2)
dioLink(Palet2_Complet,io:bOut3)
dioLink(Palet1_Dipositat,io:bOut4)
dioLink(Palet2_Dipositat,io:bOut5)

```

dioLink(B6,io:bln0): Aquesta línia enllaça la variable B6 amb l'entrada d'E/S bln0. Aquesta associació permetrà que la variable B6 sigui utilitzada per llegir dades a través d'aquesta entrada. Les següents línies de codi realitza la mateixa tasca però en direccions diferents.

La següent part del codi està relacionada amb la inicialització de les sortides.

dioLink(Cil\_Palet,io:bOut0): Aquesta línia enllaça la variable Cil\_Palet amb la sortida d'E/S bOut0. Això permetrà que la variable Cil\_Palet controli l'actuador connectat a aquesta sortida. La resta de codi fa el mateix.

Aquesta configuració d'entrades i sortides permetrà al robot interactuar amb els components externs del sistema, detectar senyals d'entrada dels sensors, i controlar els actuadors per a les diferents funcions i tasques programades.

```

while ( Seguretat_OK () = true)
//Inicialitzador robot
movej(pos0[0],Eina_palet[0],mNormalSpeed)
open (Eina_palet[0])
open (Eina_Caixa)
putln("Posició inicial assolida. ")

```

En el bucle "while" que es mostra anteriorment en les línies de codi, es controla el comportament del robot en funció de l'estat de la variable "Seguretat\_OK". El bucle "while" es repetirà contínuament mentre "Seguretat\_OK" sigui veritable, és a dir, mentre es compleixi la condició.

Dins del bucle "while", es realitzen les següents accions:

`movej(pos0[0],Eina\_palet[0],mNormalSpeed)`: Aquesta línia de codi mou el robot a la posició "pos0" utilitzant l'eina "Eina\_palet[0]" amb una velocitat normal ("mNormalSpeed"). Aquest moviment portarà el robot a una posició inicial específica des d'on començarà les seves tasques.

`open(Eina\_palet[0]): Aquesta línia obre l'eina "Eina\_palet[0]".

`open(Eina\_Caixa): Aquesta línia obre l'eina "Eina\_Caixa".

`putln("Posició inicial assolida. ")`: Aquesta línia simplement imprimeix un missatge indicant que el robot ha assolit la posició inicial i està llest per començar les seves tasques.

El bucle "while" continuarà executant aquestes accions repetidament mentre "Seguretat\_OK" segueixi sent veritable.

```
//Programa agafem palet a la pila.  
if (B8 and (!B6 or !B7)) //Si detectem palets a B8 i (no detectem a B6 o B7)  
endIf  
  
//Recollim informació de quin programa estem utilitzan (1= caixa gran)  
if(Programa_1 == 1 and Programa_2 == 0 and (B6 || B7))  
  
//Recollim informació de quin programa estem utilitzan (2 = caixa Mitjana)  
elseif(Programa_1 == 0 and Programa_2 == 1 and (B6 || B7))  
  
//Recollim informació de quin programa estem utilitzan (1+2 = 3 caixa Petita)  
elseif(Programa_1 == 1 and Programa_2 == 1 and (B6 || B7))  
  
// Tanquem programa 3  
endIf  
  
//Acabem sequencia paletitzat  
endWhile
```

En el fragment anterior es pot observar que el bucle "while" prèviament mencionat romandrà actiu mentre totes les condicions de seguretat estiguin activades. Això significa que el bloc de codi dins del bucle "while" s'executarà repetidament mentre les condicions de seguretat siguin satisfetes. No obstant això, en el moment en què alguna de les condicions de seguretat no es compleixi, el programa s'aturarà i sortirà del bucle "while".

Aquesta estructura ens permet assegurar-nos que el robot no realitzarà cap acció si no es compleixen totes les mesures de seguretat requerides. En altres paraules, el bucle "while" ens proporciona una manera eficient de controlar i monitorar constantment les condicions de seguretat abans d'executar tasques crítiques amb el robot.

```

//Programa agafem palet a la pila.
if (B8 and (!B6 or !B7)) //Si detectem palets a B8 i (no detectem a B6 o B7)
    open Eina_palet //Obrim cilindre palet
    delay(3)
    movej(appro(pos1[0],{0,0,-100,0,0,0}), Eina_palet[0] ,mNormalSpeed)
    movel(pos1[0], Eina_palet[0] ,mLentSpeed) //Ens posem a posició 1 Agafar
    palet a la pila

    while(!Sensor_Proximitat_Palet) //Si (no detectem palet)
        movel(pos1[0],{0,0,-144,0,0,0} Eina_palet[0] ,mLentSpeed) //Baixem
        14,4 cm fins a trobar palet

    endwhile

    close(Eina_palet[0]) //Tanquem eina
    putln("Posició 1 assolida. Element terminal tancat.")
    movel(appro(pos1[0],{0,0,-100,0,0,0}),Eina_palet[0],mLentSpeed)

    //Posicionem palet en la ubicació 1 o 2.
    if(!B6)
        movej(appro(pos2[0],{0,0,-100,0,0,0}),Eina_palet[0],mNormalSpeed)
        movel(pos2[0],Eina_palet[0],mLentSpeed)
        open (Eina_palet[0])
        Palet1_Dipositat = true //Enviem senyal a PLC conforme ja tenim palet
        posicionat.
        Palet1_Complet = false
        putln("Posicio 2 assolida. Palet 1 dipositat.")
        movel(appro(pos2[0],{0,0,-100,0,0,0}),Eina_palet[0],mLentSpeed)
        movej(pos0[0],Eina_palet[0],mNormalSpeed) //Tornem a posició origen
        per evitar colisions

    elseif(!B7)
        movej(appro(pos3[0],{0,0,-100,0,0,0}),Eina_palet[0],mNormalSpeed)
        movel(pos3[0],Eina_palet[0],mLentSpeed)
        open (Eina_palet[0])
        Palet2_Dipositat = true //Enviem senyal a PLC conforme ja tenim palet
        posicionat.
        Palet2_Dipositat = false
        putln("Posicio 3 assolida. Palet 2 dipositat.")
        movel(appro(pos3[0],{0,0,-100,0,0,0}),Eina_palet[0],mLentSpeed)
        movej(pos0[0],Eina_palet[0],mNormalSpeed) //Tornem a posició origen
        per evitar colisions

    endif

endif
endif

```

El codi previ descriu un programa que permet al robot agafar un palet d'una pila.

La condició es comprova si hi ha palets detectats al sensor B8 (que indica la presència de palets) i, al mateix temps, si no es detecten palets als sensors B6 o B7. Això significa que el robot només procedirà si hi ha palets a B8 i no n'hi ha als sensors B6 o B7 actius pel que voldrà dir que no tenim palet.

Si es compleix la condició de "if", el cilindre de l'eina palet s'obre, preparant-se per agafar el palet de la pila.

El robot es mou a la posició apropiada (posició 1) per a l'agafament del palet a la pila. Primer es realitza un moviment ràpid (mNormalSpeed) per a una aproximació inicial, i després es realitza un moviment lent (mLentSpeed) per acostar-se amb precisió a la posició de l'agafament.

El bucle s'executa mentre no es detecti el palet al sensor de proximitat (Sensor\_Proximitat\_Palet). Aquest sensor està dissenyat per detectar la presència del palet a la distància perfecta perquè l'eina pugui agafar el palet.

Dins del bucle, el robot es mou cap avall (en l'eix Z) per a buscar el palet a la pila. Utilitza la funció "movel" per realitzar un moviment de tipus lineal amb velocitat lenta (mLentSpeed). El desplaçament és de 14,4 cm cap avall fins que el sensor de proximitat detecti la presència del palet. Quan el sensor de proximitat detecta el palet, el bucle "while" finalitza.

Amb la funció "close(Eina\_palet[0])", el programa tanca l'eina que te la funció agafar el palet de la pila.

Tot seguit es mostra un missatge de text utilitzant la funció "putln". El missatge diu "Posició 1 assolida. Element terminal tancat." Aquesta línia és simplement una indicació de que s'ha arribat a la Posició 1 i que l'eina d'agafament ha estat tancada amb èxit.

Amb la funció "movel(appro(pos1[0],{0,0,-100,0,0,0}),Eina\_palet[0],mLentSpeed)", el robot es mou a una posició d'aproximació per la Posició 1. Aquest moviment és realitzat utilitzant la funció "movel" amb una velocitat lenta ("mLentSpeed"). La posició d'aproximació és important per a una manipulació precisa, permetent al robot alinear-se adequadament abans de realitzar la següent acció.

Aquest codi s'executa després d'haver agafat un palet i es responsabilitza de posicionar el palet agafat en una de les dues ubicacions disponibles (ubicació 1 o 2).

Es comprova si la condició "IB6" és certa, el que indica que no s'ha detectat res a l'entrada B6 (Sensor d'ubicació 1). Si és així, això significa que la ubicació 1 està disponible per dipositar el palet agafat.

Si la ubicació 1 està disponible, el robot es mou a una posició d'aproximació per la ubicació 1 utilitzant la funció "movej" amb una velocitat normal ("mNormalSpeed"). A continuació, es mou a la posició d'arribada utilitzant la funció "movei" amb una velocitat lenta ("mLentSpeed").

Es realitza l'acció d'obrir l'eina de subjecció ("Eina\_palet") per deixar anar el palet en la ubicació 1.

Es configura la variable "Palet1\_Dipositat" a cert (true) per enviar una senyal al PLC que indica que el palet ha estat dipositat a la ubicació 1. Al mateix temps, la variable "Palet1\_Complet" es configura a fals (false) per indicar que la ubicació 1 ja pot començar a ser paletitzada.

Un cop canviades les variables es mostra un missatge de text utilitzant la funció "putln" per indicar que s'ha assolit la posició 2 i s'ha dipositat el palet 1.

Tot seguit es mou a la posició d'aproximació a 2 utilitzant la funció "movei" amb una velocitat lenta.

Per finalitzar aquesta línia de codi es mou a la posició d'origen ("pos0[0]") per evitar col·lisions amb altres objectes o estructures.

Si la condició "!B7" és certa, això indica que no s'ha detectat res a l'entrada B7 (Sensor d'ubicació 2). Si és així, això significa que la ubicació 2 està disponible per dipositar el palet agafat. Pel que es realitzarà el mateix codi anteriorment descrit però substituint la posició de deixada 2 per la 3.

Aquest codi permet que el robot pugui dipositar els palets agafats en les ubicacions corresponents, ja sigui en la ubicació 1 o en la ubicació 2, depenent de la disponibilitat d'aquestes ubicacions segons l'estat dels sensors B6 i B7.

```

//Recollim informació de quin programa estem utilitzan (1= caixa gran)
if(Programa_1 == 1 and Programa_2 == 0 and (B6 || B7))

//Recollim informació de quin programa estem utilitzan (2 = caixa Mitjana)
elseif(Programa_1 == 0 and Programa_2 == 1 and (B6 || B7))

//Recollim informació de quin programa estem utilitzan (1+2 = 3 caixa Petita)
elseif(Programa_1 == 1 and Programa_2 == 1 and (B6 || B7))

// Tanquem programa 3
endif

```

Aquestes línies que precedeixen es mostren com estan dissenyades per avaluar diferents condicions relacionades amb els programes que controlen el robot i l'estat dels sensors B6 i B7.

"if(Programa\_1 == 1 and Programa\_2 == 0 and (B6 || B7))" Aquesta condició comprova si el "Programa\_1" està actiu (és igual a 1), mentre que el "Programa\_2" no està actiu (és igual a 0), i a més, almenys un dels sensors B6 o B7 està actiu (és igual a 1). Aquesta condició es verificarà si només el "Programa\_1" està actiu i un dels sensors B6 o B7 està actiu. Si aquesta condició és certa, s'executarà el codi del programa 1.

"elseif(Programa\_1 == 0 and Programa\_2 == 1 and (B6 || B7))" Aquesta condició comprova si el "Programa\_1" no està actiu (és igual a 0), mentre que el "Programa\_2" està actiu (és igual a 1), i a més, almenys un dels sensors B6 o B7 està actiu (és igual a 1). Aquesta condició es verificarà si només el "Programa\_2" està actiu i un dels sensors B6 o B7 està actiu. Si aquesta condició és certa, s'executarà el codi del programa 2.

"elseif(Programa\_1 == 1 and Programa\_2 == 1 and (B6 || B7))" - Aquesta condició comprova si tant el "Programa\_1" com el "Programa\_2" estan actius (ambdós iguals a 1), i a més, almenys un dels sensors B6 o B7 està actiu (és igual a 1). Aquesta condició es verificarà si tots dos programes estan actius i un dels sensors B6 o B7 està actiu. Si aquesta condició és certa, s'executarà el codi del programa 3.

```

//Recollim informació de quin programa estem utilitzan (1= caixa gran)
if(Programa_1 == 1 and Programa_2 == 0 and (B6 || B7))
  Num_Pisos = 4
  Num_Caixes_Pis = 2

  //Escollim ubicació de treball Palet 1
  if(Palet1_Complet == false and Palet1_Dipositat == true)

  elseif(Palet2_Complet == false and Palet2_Dipositat == true)
  endif
//Tanquem programa 1

```

En l'extracte anterior de codi es descriu, en el cas d'estar treballant amb el programa 1, assignarem el valor de 4 a la variable "Num\_Pisos" i el valor de 2 a la variable "Num\_Caixes\_Pis".

A continuació, entrarem en un bloc condicional "if" que avaluarà les següents condicions, "Palet1\_Complet == false" Comprova si el Palet 1 està complet, o no a la ubicació 1, "Palet1\_Dipositat == true" Comprova si el Palet 1 està disponible, és a dir, si s'ha dipositat un palet a la ubicació 1.

Si aquestes dues condicions són certes, significa que el Palet 1 està buit i està disponible per a la paletització de les caixes. En aquest cas, el programa continuarà paletitzant les caixes a la ubicació 1. Si alguna de les condicions no és certa, es passarà a paletitzar les caixes a la ubicació 2 en cas de complir-se les condicions anteriorment descrites per la ubicació 2.

```
//Inicialitzem la variable "Pis", li indiquem quants pisos tindrem.
for Pis=1 to Num_Pisos
  put("Pis: ")
  putln(Pis+1)

//Inicialitzem la variable "Caixa", li indiquem quantes caixes
tindrem per pis.

for Caixa=1 to Num_Caixes_Pis
  put("Caixa: ")
  putln(Caixa+1)
```

En el codi utilitzat anteriorment es pot veure un bucle "for" per inicialitzar dues variables: "Pis" i "Caixa". El bucle "for" permet repetir un conjunt de comandes un cert nombre de vegades, definint un valor inicial, un valor final i un pas d'increment.

"for Pis=1 to Num\_Pisos": Aquesta línia inicialitza la variable "Pis" amb el valor 1 i estableix com a valor final el contingut de la variable "Num\_Pisos". Aquest bucle executarà el bloc de codi contingut a dins per a cada valor de "Pis" des del 1 fins al valor de "Num\_Pisos" que en el programa 1 és igual a 4.

Dins del primer bucle "for", trobem un altre bucle "for" anomenat "Caixa=1 to Num\_Caixes\_Pis": Aquest segon bucle inicialitza la variable "Caixa" amb el valor 1 i estableix com a valor final el contingut de la variable "Num\_Caixes\_Pis". Aquest segon bucle



executarà el bloc de codi contingut a dins per a cada valor de "Caixa" des del 1 fins al valor de "Num\_Caixes\_Pis" que serà igual a 2.

Dins dels bucles "for", trobem dues comandes "put" que mostren per pantalla els valors de les variables "Pis+1" i "Caixa+1". Aquestes comandes mostren els valors dels comptadors "Pis" i "Caixa" incrementats en 1 perquè els bucles "for" comencen a comptar des del valor inicial especificat.

```
//Comença iteració posicionament caixes.
if(Caixa==1)
  dx = 0
  dy = 0
  dz = 0

//Segona caixa desplaçem 380mm en y
elseif(Caixa==2)
  dy=dy+380

endif
```

Aquest codi és una part de la iteració que es fa per al posicionament de caixes en el programa de robot. Es tracta d'un conjunt de condicionals que ens permeten controlar les coordenades de posició (dx, dy, dz) del robot per a cada caixa en un pis determinat.

"if(Caixa==1)": Aquesta línia estableix la condició de si el valor de la variable "Caixa" és igual a 1, s'executarà el bloc de codi que ve a continuació. On ens trobem tres línies que assignen valors a les variables "dx", "dy" i "dz". Si la condició anterior es compleix, es posarà "dx" a 0, "dy" a 0 i "dz" a 0.

"elseif(Caixa==2)": Aquesta línia és una altra condició. Si la condició anterior no es compleix i el valor de "Caixa" és igual a 2, s'executarà el bloc de codi que ve a continuació. On tenim una sola línia que modifica el valor de la variable "dy". Si la condició "elseif" es compleix, el valor de "dy" s'incrementarà en 380 mm.

```
//Agafem caixes
if(Permis_Carrega_Caixa)
  //Agafem caixa
  movej (appro(pos4[0],{0,0,-100,0,0,0}),Eina_Caixa[0]
  ,mNormalSpeed)
  movel (pos4[0],Eina_Caixa[0],mLentSpeed)
  close (Eina_Caixa[0])
  movel(appro(pos4[0],{0,0,-100,0,0,0}),Eina_Caixa[0]
  ,mLentSpeed)
```

```
waitEndMove()

//Deixem caixa a la ubicació Palet 1
movej(appro(pos5[0],{-dx,-dy,(dz-100),0,0,0}),
Eina_Caixa[0],mNormalSpeed)
movel(appro(pos5[0],{-dx,-dy,dz,0,0,0}),
Eina_Caixa[0],mLentSpeed)
open(Eina_Caixa[0])
movel(appro(pos5[0],{-dx,-dy,(dz-100),0,0,0}),
Eina_Caixa[0],mLentSpeed)

//incrementem Caixa
Caixa = Caixa + 1

endIf
```

Aquest codi pertany a la part del programa de robot que es dedica a agafar caixes de la zona de càrrega i col·locar-les a la ubicació 1, que es correspon amb la posició 5 del robot.

"if(Permis\_Carrega\_Caixa)": Aquesta línia estableix una condició per verificar si el robot té permís per carregar caixes. Si la variable "Permis\_Carrega\_Caixa" és certa (true), el programa procedeix amb la seqüència d'agafar i col·locar les caixes. Aquesta variable la dona el PLC de la instal·lació.

Dins del bloc de codi de la condició "if", hi ha una sèrie de moviments que el robot realitza per agafar una caixa de la zona de càrrega i després col·locar-la a la ubicació 1 (posició 5 del robot). El robot primer es mou cap a una posició d'aproximació ("appro") per preparar-se per agafar la caixa. Aquest moviment està definit a partir de la posició 4 del robot, amb una petita elevació en l'eix Z perquè la pinça estigui a una altura adequada per a la recollida.

Seguidament, el robot realitza un moviment lenticular ("movel") cap a la posició 4, és a dir, la posició exacta on es troba la caixa a la zona de càrrega. A continuació, el robot tanca la pinça ("close") per agafar la caixa i la aixeca una mica ("movel") per separar-la de la zona de càrrega.

Un cop agafada la caixa, el robot torna a moure's cap a la posició d'aproximació ("appro") per a la ubicació 1 (posició 5 del robot), on col·locarà la caixa. El robot executa un moviment lenticular ("movel") cap a la posició 5, tenint en compte les coordenades "dx" i "dy" per a la correcta disposició de la caixa en la ubicació.

Un cop el robot és a la posició de dipòsit, obre la pinça ("open") per deixar anar la caixa. Finalment, el robot torna a moure's lleugerament cap a l'amunt ("move") per allunyar-se de la caixa col·locada.

Per concloure, es fa un increment de la variable "Caixa" en 1 per preparar-se per agafar la següent caixa de la zona de càrrega.

```
endFor
//Al augmentar de pis augmentem l'altura de deixada amb l'altura
de la caixa
dx=0
dy=0
dz = dz + 250

endFor

Palet1_Complet = true
```

El codi previ s'executa una vegada s'han recollit totes les caixes d'un pis i s'han deixat en la ubicació del palet (Palet 1). L'objectiu d'aquesta part del codi és ajustar la posició vertical (eix Z) per a la següent capa de caixes que s'apilarà sobre el palet.

Es reinicien les variables "dx" i "dy" a 0. Aquestes variables s'utilitzen per controlar el desplaçament horitzontal (eixos X i Y) de les caixes, i és important restablir-les a 0 abans d'iniciar el següent pis per assegurar que les caixes es col·loquin correctament.

A continuació, s'incrementa la variable "dz" en 250. Aquesta variable controla l'alçada vertical de les caixes, específicament, la posició en l'eix Z. En augmentar "dz" en 250, aconseguim que les caixes es col·loquin a una alçada superior respecte al pis anterior, preparant-se per a la següent capa.

Finalment, es tanca el bucle "for" amb l'expressió "endFor", indicant que s'han completat totes les iteracions per a cada capa de caixes, i el programa passa a la següent part de la lògica.

En acabar el bucle, es canvia el valor de la variable "Palet1\_Complet" a "true". Aquesta variable s'utilitza per comunicar al PLC que el palet 1 està completament ple amb les caixes i està llest per ser transportat fora de la instal·lació. Aquesta modificació és important per assegurar que el programa segueixi la lògica adequada i pugui passar a altres tasques.

En el paletitzat de la ubicació 2, tot el procediment serà pràcticament idèntic al de la primera ubicació, però amb la variant de modificar algunes variables específiques per adaptar-se a la nova ubicació. Això ens permetrà una gran reutilització de codi i una gestió més eficient dels diferents programes.

Pel que fa als altres programes, és cert que compartiran l'estructura principal amb les adaptacions pertinents, com ara l'ajustament del nombre de caixes per pis i el nombre de pisos segons les especificacions del procés de paletitzat. No obstant això, una de les variacions més rellevants serà el patró de paletitzat d'un pis a l'altre. Aquesta diferència es degut a la inversió de l'ordre en què les caixes són col·locades al palet, amb l'objectiu de millorar la seva estabilitat i garantir una paletització segura i ben organitzada.

És essencial assenyalar que, malgrat aquestes petites diferències, els tres programes mantenen una estructura global similar. Això és gràcies a l'ús d'estructures de control com bucles i condicionals, que ens permeten adaptar la lògica de paletitzat de manera flexible i eficient. A més, l'ús adequat de variables ens facilita la reutilització de codi i la gestió coherent de les diferents ubicacions.

Per completar la nostra descripció, cal destacar la figura 41, que representa una taula exhaustiva amb totes les posicions emmagatzemades en el robot. Aquesta taula és d'una importància crucial, ja que proporciona un recurs valuós per a la recollida i dipòsit dels diversos elements que intervenen en el nostre sistema.

Posició	Descripció
Pos0	Posició inici del robot
Pos1	Agafada palet a la pila de palets
Pos2	Deixada palet a la ubicació 1
Pos3	Deixada palet a la ubicació 2
Pos4	Agafada de caixes
Pos5	Deixada de caixes en la ubicació 1. Programa 1
Pos6	Deixada de caixes en la ubicació 2. Programa 1
Pos7	Deixada de caixes en la ubicació 1. Programa 2
Pos8	Deixada de caixes en la ubicació 2. Programa 2
Pos9	Deixada de caixes en la ubicació 1. Programa 3
Pos10	Deixada de caixes en la ubicació 2. Programa 3

Figura 41. Taula posicions robot

#### 4.2.5 Codi STOP

```

begin
  movej(Pos0[0],Eina_palet,mNormalSpeed)
  waitEndMove()
  open(Eina_Caixa)
  open(Eina_palet)
  putln("*****")
  putln("  Posició final assolida. Element terminal obert.")
  putln("                Fi de l'aplicació")
  putln("*****")
  resetMotion()
end

```

Aquest codi "STOP" té com a objectiu finalitzar l'aplicació i retornar el robot a la seva posició inicial de manera segura. A continuació, una explicació a grans trets del que fa cada part del codi tal i com es pot veure en la figura 53.

``movej(Pos0[0], Eina_palet, mNormalSpeed)``: Aquesta línia mou el robot a la posició inicial (Pos0). La velocitat del moviment és normal.

``waitEndMove()``: Aquesta funció fa que el programa esperi fins que el moviment anterior s'hagi completat abans de continuar amb les següents instruccions. Això assegura que el robot arribi a la posició desitjada abans de seguir amb les tasques següents.

``open(Eina_Caixa)``: Aquí s'obre l'eina de caixa (Eina\_Caixa). Això pot ser útil perquè l'operador del robot pugui treure les caixes que s'han manipulat durant l'aplicació.

``open(Eina_palet)``: En aquesta línia, s'obre l'eina de palet (Eina\_palet). Això permet que l'operador del robot pugui retirar els palets que s'han utilitzat durant l'aplicació.

Missatges: Les línies de codi que contenen ``putln`` serveixen per mostrar missatges en el panell d'interfície del robot. En aquest cas, s'imprimeixen diversos missatges que indiquen que s'ha arribat a la posició final, s'han obert les eines terminals i que l'aplicació ha finalitzat.

``resetMotion()``: Aquesta funció restableix el robot a la seva posició inicial, eliminant qualsevol moviment actual i configurant el robot per a futurs moviments segurs.

## **5. RESUM DEL PRESSUPOST**

El projecte de paletització automàtica d'una línia productiva té un cost total de seixanta-sis mil quatre-cents noranta-tres euros amb cinquanta-quatre cèntims sense IVA inclòs.

## 6. CONCLUSIONS

Aquest projecte s'ha realitzat amb la intenció de reduir l'esforç físic dels treballadors de la línia productiva, augmentar la productivitat i reduir costos de producció, ja que per poder paletitzar manualment es necessiten dues persones per torn.

Gràcies a aquesta automatització es redueix el risc d'accident o malaltia professional degut a estar carregant pes durant la jornada laboral, i aconseguint que aquestes persones puguin dedicar el seu temps en augmentar el control de qualitat del producte que s'està fabricant.

Aquest projecte no es portarà a terme en cap empresa. És un projecte per demostrar tots els coneixements adquirits durant el grau, en conjunt amb l'experiència pròpia viscuda en diverses empreses en les quals he treballat. Per poder demostrar que el programa funciona correctament es simularan els diferents programes dels projecte.

Una de les grans dificultats del projecte han estat els diferents programaris utilitzats, ja que al utilitzar una versió d'estudiants hi ha algunes parts dels diferents programes limitades en el seu ús pel que suposat una dificultat afegida. També s'ha vist dificultat l'execució del projecte no comptar amb la llicència del programa del robot STAUBLI, ja que no s'ha pogut verificar el funcionament del codi creat.

Gerard Falgueras Regincós

Graduat en Enginyeria Electrònica i Automàtica

Girona, 04 de setembre de 2023

## **7. RELACIÓ DE DOCUMENTS**

Aquest projecte està constituït pels següents documents: memòria, plànols, plec de condicions, estat d'amidaments i pressupost.



## 8. BIBLIOGRAFIA

MasterPLC. (15 de maig de 2020). Instalación de FACTORY I/O (Primeros pasos en Factory IO Actualizado 2020) #3 [Arxiu de vídeo]. Youtube. [https://www.youtube.com/watch?v=ZohQ-jnlwg0&ab\\_channel=MasterPLC](https://www.youtube.com/watch?v=ZohQ-jnlwg0&ab_channel=MasterPLC)

PLC ESPAÑOL. (27 de juliol de 2021). AGREGAR CAMPO DE ENTRADA de DATOS + CONFIGURACIÓN + VARIABLE WINCC TIA PORTAL - PLC ESPAÑOL !!! 137 [Arxiu de vídeo]. Youtube. [https://www.youtube.com/watch?v=54kYjuqtbWc&list=PLk-E5o5NaGkiUbAK3xksnqEdnGflzs9gL&index=15&ab\\_channel=PLCESPA%C3%91OL](https://www.youtube.com/watch?v=54kYjuqtbWc&list=PLk-E5o5NaGkiUbAK3xksnqEdnGflzs9gL&index=15&ab_channel=PLCESPA%C3%91OL)

PLC ESPAÑOL. (10 d'agost de 2021). AGREGAR BOTONES + CONFIGURACIÓN + VARIABLES con WINCC TIA PORTAL - PLC ESPAÑOL !!! 139 [Arxiu de vídeo]. Youtube. [https://www.youtube.com/watch?v=xqwsgap3RBE&t=261s&ab\\_channel=PLCESPA%C3%91OL](https://www.youtube.com/watch?v=xqwsgap3RBE&t=261s&ab_channel=PLCESPA%C3%91OL)

PLC ESPAÑOL. (26 d'octubre de 2021). Cómo CONFIGURAR AVISOS y ALARMAS en el Panel HMI - PLC ESPAÑOL !!! 148 [Arxiu de vídeo]. Youtube. [https://www.youtube.com/watch?v=-hC-mt4ao0o&ab\\_channel=PLCESPA%C3%91OL](https://www.youtube.com/watch?v=-hC-mt4ao0o&ab_channel=PLCESPA%C3%91OL)

PLC ESPAÑOL. (14 de novembre de 2021). Cómo se Crea y Administra el Acceso de Usuarios a un Panel HMI - PLC ESPAÑOL !!! 151 [Arxiu de vídeo]. Youtube. [https://www.youtube.com/watch?v=xEYAFJdvmA8&t=37s&ab\\_channel=PLCESPA%C3%91OL](https://www.youtube.com/watch?v=xEYAFJdvmA8&t=37s&ab_channel=PLCESPA%C3%91OL)

PLC ESPAÑOL. (28 de novembre de 2021). Cómo se configuran las RECETAS en un Panel HMI - PLC ESPAÑOL !!! 153 [Arxiu de vídeo]. Youtube. [https://www.youtube.com/watch?v=8azFdp-pCho&ab\\_channel=PLCESPA%C3%91OL](https://www.youtube.com/watch?v=8azFdp-pCho&ab_channel=PLCESPA%C3%91OL)

STÄUBLI ROBOTICS (2006). Manual programació VAL3 REFERENCE MANUAL version 5.3 (<https://docplayer.net/199893014-Val3-reference-manual.html>)

## **9. GLOSSARI**

GEMMA: Guia d'Estudi dels Modes de Marxes i de Parades

GRAFCET: Graphe Fonctionnel de Commande Etape Transition

PLC: Programmable Logic Controller