

Índex de figures

Figura 2.1: Imatges explicatives de les parts d'un palet.....	012
Figura 2.2: Imatge banc de clavats manual.....	013
Figura 2.3: Imatge línia semi automàtica.....	013
Figura 2.4: Imatge clavadora hidràulica.....	014
Figura 2.5: Imatge mòduls finalització del palet.....	014
Figura 2.6: Imatge alimentació automàtica de fustes travesseres.....	015
Figura 2.7: Imatge clavadora de parrilles de canvi ràpid.....	015
Figura 2.8: Imatge cargador automàtic de fustes superiors amb robot.....	016
Figura 2.9: Imatge sistema de visió artificial pel control de qualitat.....	016
Figura 2.10: Imatge de programa de disseny 3D de palets.....	017
Figura 2.11: Diagrama explicatiu procés de fabricació d'un palet en una línia de clavats automàtica.....	018
Figura 3.1: Diagrama de les comunicacions dels components de control en el projecte.....	021
Figura 5.1: Esquema de l'encapsulat del Modbus TCP/IP.....	032
Figura 6.1: Pantalla producció abans de la realització del projecte.....	033
Figura 7.1: Estructura DDT, que conté les variables de la producció.....	037
Figura 7.2: Array creada dels 25 dies de producció.....	037
Figura 7.3: Diagrama de flux de la "Producció_dia".....	040
Figura 7.4: Diagrama de flux de la "Programació_horari_Torns".....	041
Figura 7.5: Diagrama de flux del "Control_Solapament_Horari_Torns".....	042
Figura 7.6: Pantalla principal de monitorització de la producció del terminal tàctil.....	044
Figura 7.7: Pantalla de monitorització de la producció per torns.....	044
Figura 7.8: Principals controls de la interfície creada en VB(Imatge1).....	045
Figura 7.9: Principals controls de la interfície creada en VB(Imatge2).....	046
Figura 7.10: Principals controls de la interfície creada en VB(Imatge3).....	047

Figura 7.11: Principals controls de la interfície creada en VB(Imatge4).....	048
Figura 7.12: Menú principal de l'aplicació.....	050
Figura 7.13: Diagrama de flux de la funció "Buto_Conectar_Click"	050
Figura 7.14: Selecció lectura informació de la producció per dia.....	051
Figura 7.15: Diagrama de flux de la funció "Buto_llegir_Click" (opció Dia).....	052
Figura 7.16: Diagrama de flux de la funció "Envio_comando_escritura".....	053
Figura 7.17: Diagrama de flux de la funció "Extracció_dades_dia".....	054
Figura 7.18: Visualització de la taula de les dades de la producció diària.....	055
Figura 7.19: Diagrama de flux de la funció "Refrescar_taula_dia".....	055
Figura 7.20: Mostra els diferents objectes "Textbox" que visualitzen les dades de la producció diària per recepta.....	056
Figura 7.21: Diagrama de flux de la funció "Buto_llegir_Click"(opció Recepta).	057
Figura 7.22: Visualització de les gràfiques de la quantitat de palets produïda.	058
Figura 7.23: Diagrama de flux de la funció Buto_Refrescar_Gràfica_Click.....	058
Figura 7.24: Monitorització de les alarmes actives.....	059
Figura 7.25: Diagrama de flux de la funció "Timer_Alarmes_Tick".....	060
Figura 7.26: Diagrama de flux de la funció "Crear_Llibre_Excel".....	061
Figura 7.27: Diagrama de flux de la funció "Omplir_Arxiu_Excel".....	062
Figura 8.1: Panell del simular del PLC.....	071
Figura 8.2: Taula d'animació amb el valor animat de les variables internes del PLC.....	071
Figura 8.3: Execució online d'un bloc de funció DFB del PLC.....	072
Figura 8.4: Taula d'animació amb el valor de les variables internes del terminal tàctil.....	073
Figura 8.5: Emulació del terminal tàctil mitjançant el simulador.....	073
Figura 8.6: Controlador terminal HMI amb la direcció del simulador del PLC....	074
Figura 8.7: Interrupció del codi del programa Visual Basic per a la seva depuració.....	075
Figura 8.8: Fragment de codi on es canvia l'adreça IP, per poder treballar amb mode simulació.....	075
Figura 8.9: Exemple trama resposta a una petició d'escriure múltiples registres.....	076
Figura 10.1: Esquema de la integració d'un dispositiu mòbil a la xarxa industrial.....	079

Figura 10.2: Exemple de la connectivitat d'una aplicació utilitzant TeslaModbusSCADA.....	079
--	-----

Índex de taules

Taula 3.1: Principals característiques de l'autòmat M340 2010.....	022
Taula 3.2: Principals característiques gràfiques del terminal XBTGTO.....	023
Taula 5.1: Trama genèrica en funció de la codificació utilitzada ASCII o RTU..	029
Taula 5.2: Estructura del prefixe del Modbus TCP.....	030
Taula 5.3: Estructura del missatge Modbus TCP.....	031
Taula 5.4: Funcions Bàsiques i codis d'operació.....	031
Taula 7.1: Variables producció diària.....	035
Taula 7.2: Entrades i sortides del Bloc "Producció Diària".....	039
Taula 7.3: Entrades i sortides del Bloc "Programació_horari_Torns".....	041
Taula 7.4: Entrades i sortides Bloc "Control_Solapament_Horari_Torns".....	042
Taula 7.5: Principals controls de la interfície creada en VB(Taula1).....	045
Taula 7.6: Principals controls de la interfície creada en VB(Taula2).....	047
Taula 7.7: Principals controls de la interfície creada en VB (Taula3).....	048
Taula 7.8: Principals controls de la interfície creada en VB(Taula4).....	049

Índex

1.-OBJECTIUS I ESTRUCTURACIÓ DEL TREBALL.....	008
1.1.- Objecte.....	008
1.2.- Finalitat.....	008
1.3.- Objectius.....	008
1.4.- Justificació.....	009
1.5.- Especificacions bàsiques.....	009
1.6.- Estructuració del treball.....	009
2.- INTRODUCCIÓ.....	011
2.1.- Evolució de la construcció industrial de palets.....	012
2.2.- Línia automàtica de canvi ràpid.....	017
3.-MAQUINARI UTILITZAT	020
3.1.- M340 cpu 2010.....	021
3.2.- XBTGTO5310.....	022
3.3.- Router Ewon Cosy.....	023
3.4.- Ordinador personal (PC).....	024
4.- PROGRAMARI UTILITZAT.....	025
4.1.- Unity V6.0.....	025
4.2.- Vijeo Designer V6.1.....	026
4.3.- Visual Basic 2008.NET.....	026
4.4.- Ecatcher V4.0.....	006
4.5.- Wireshark.....	027
5.- COMUNICACIÓ MODBUS TCP/IP.....	028

5.1.- Estructura de xarxa.....	028
5.2.- Protocol.....	028
5.3.- Comunicació Modbus TCP/IP.....	029
5.4.- Característiques del Modbus TCP/IP.....	029
5.5.- Estructura del Modbus TCP/IP.....	030
5.6.- Encapsulat del Modbus TCP/IP.....	032
6.- CARACTERÍSTIQUES PROGRAMARI DESENVOLUPAT.....	033
7.- PROGRAMACIÓ REALITZADA.....	035
7.1.- Programació PLC.....	035
7.1.1.- DFB Producció diària.....	038
7.1.2.- DFB Programació horari torns.....	041
7.1.3.- DFB Control solapament horari torns.....	042
7.2.- Programació terminal tàctil XBTGTO5310.....	043
7.3.- Programació en Visual Basic 2008.....	045
7.3.1.- Establir connexió amb el procés.....	049
7.3.2.- Lectura de les dades productives del procés (opció dia).....	051
7.3.3.- Extracció de les dades productives (opció dia).....	053
7.3.4.- Extracció de les dades productives (opció receptes).....	056
7.3.5.- Monitorització gràfica de les dades de producció.....	057
7.3.6.- Enregistrament de les alarmes i la producció.....	059
7.3.7.- Exportació de les dades a un arxiu Excel.....	061
8.- POSTA EN MARXA I DEPURACIÓ.....	070
8.1.- Depuració del prgorama PLC.....	070
8.2.- Depuració del programa HMI.....	072
8.3.- Depuració de la comunicació PLC→Terminal tàctil.....	074
8.4.- Depuració del programa en Visual Basic 2008.....	074
8.5.- Depuració de la comunicació (PLC→HMI)→Visual Basic 2008.....	075

8.6.- Posada en funcionament del programari final.....	076
9.- CONCLUSIONS.....	077
10.- FUTURS PROJECTES.....	079
11.- BIBLIOGRAFIA.....	081

1.- OBJECTIUS I ESTRUCTURACIÓ DEL TREBALL

1.1.- OBJECTE

L'objectiu final d'aquest projecte és el disseny i implementació d'un programari d'adquisició de les dades productives d'un procés, per l'empresa Eglinton Timber Products LTD. Mitjançant el treball realitzat a l'empresa Mecànica Cape S.L

1.2.- FINALITAT

Aquest projecte té com a finalitat crear un programari per tal de dotar de millores funcionals a un producte ja existent al mercat, com és una línia automàtica de fabricació de palets, de l'empresa Mecànica Cape S.L per a la seva posterior comercialització.

Per realitzar-ho es procedirà a l'elecció dels diferents components elèctrics i electrònics necessaris, a la programació dels diferents dispositius de control i, finalment, a la posta en marxa i depuració del correcte funcionament d'aquest "software".

1.3.- OBJECTIUS

Els temes tractats específicament en el projecte són els descrits a continuació:

- Breu introducció al passat, futur i present de les línies de clavats automàtiques de palets.
- Descripció dels diferents components escollits.
- Estudi del funcionament de les comunicacions industrials "Modbus TCP/IP" el qual permet el funcionament del nostre projecte.
- Estudi del funcionament del programari desenvolupat.
- Realització de la programació del diferents dispositius.
- Posta en marxa i depuració.

Tenint en compte els temes que es tractaran en el projecte, l'objectiu que es vol aconseguir és el de millorar un producte existent i adaptar-l'ho a les noves tecnologies, fer-l'ho més funcional i alhora més competitiu.

1.4.- JUSTIFICACIÓ

Des de la perspectiva de l'empresa Mecànica Cape S.L s'utilitza aquest projecte amb l'objectiu principal de millorar qualitativament un producte existent, però també com a punt de partida per iniciar la comercialització de nous productes, i ampliar així el catàleg de solucions industrials a les necessitats de les empreses de fabricació de palets.

Per part de l'empresa Eglinton Timber Products LTD, aquest projecte els facilita una eina per l'adquisició de les dades productives del seu procés. L'enregistrament d'aquestes dades permetrà poder fer-ne un seguiment i analitzar-les amb posterioritat, ja sigui amb les eines analítiques de l'aplicació o mitjançant la transferència d'aquestes dades a una fulla de càlcul o base de dades.

1.5.- ESPECIFICACIONS BÀSIQUES

Las especificacions bàsiques que es desitgen aconseguir al realitzar el projecte són les següents:

- Diàleg home-màquina interactiu i intuïtiu.
- Donar en tot moment al departament de producció la màxima informació útil sobre la línia.
- El programari ha de treballar de forma autònoma i correcte.
- El programari ha de disposar d'eines d'anàlisi de les dades obtingudes.
- Exportació de les variables obtingudes del procés en un format per a la seva posterior manipulació, fulla de càlcul o base de dades.
- Facilitat per incorporar el sistema d'adquisició de dades d'aquest projecte en d'altres línies existents.

1.6.- ESTRUCTURACIÓ DEL TREBALL

Aquest treball s'ha estructurat en els següents capítols:

Al capítol 2 es fa una breu descripció del passat, del present i el futur de les línies automàtiques de clavat, i es situa aquest projecte dins d'aquest marc.

Al capítol 3 s'expliquen tots els components seleccionats a nivell de maquinari per poder implementar el projecte.

El capítol 4 fa referència a tot el programari que hem fet servir per poder desenvolupar el projecte

El capítol 5 es centra en el Modbus TCP/IP. Quines són les seves característiques, de quin conjunt de funcions està compostat, quina és la funció de cadascuna i com l'utilitzarem dins del projecte.

Al capítol 6 ens centrarem en les característiques del programari desenvolupat,

Al capítol 7 es fa una descripció de tota la programació realitzada en el projecte, dividida en les diferents parts que el constitueixen, quina informació aporta, com està enllaçada tota aquesta informació amb el procés i com s'extreu cap a l'exterior.

Al capítol 8 s'expliquen els passos que s'han seguit per fer la posta en marxa i depuració del projecte, així com els resultats finals obtinguts.

Finalment, al capítol 9 es treuen les conclusions d'aquest projecte, i en el capítol 10 s'exposen els futurs projectes i/o ampliacions.

2.- INTRODUCCIÓ

Mecànica Cape ,S.L es va fundar al 1967, començant la seva activitat en el sector de la maquinària per treballar la fusta. A causa de la demanda d'equips per al nou mercat del palet, es va anar especialitzant en aquest tipus de maquinària fins a convertir-se en el primer fabricant d'Espanya i un dels primers del món, tant a nivell de màquines individuals, línies de canvi ràpid per sèries curtes, línies de clavat automàtic en cicle continu per grans sèries, així com línies per palets especials.

El departament d'enginyeria de mecànica Cape S.L té com a prioritat la constant innovació tecnològica per a una millor competitivitat, així com per donar una solució personalitzada a cada client. És dins d'aquesta innovació constant en la qual es situa aquest projecte. En concret, aquest treball està emmarcat dins la necessitat de disposar d'eines de gestió i anàlisis de la producció que tenen la gran majoria de les empreses. Aquesta tendència també existeix al sector dels fabricants de palets, els quals veuen la utilitat que els suposa tenir un sistema d'adquisició de dades per al seu posterior diagnòstic i avaluació a nivell de gestió.

Actualment, existeixen molts programaris que satisfan la necessitat anteriorment descrita, els anomenats "SCADA". També és cert que aquests programes tenen un cost elevat i que en determinades aplicacions no se n'extreu tot el seu potencial, ja que disposen de moltes eines específiques que s'infrautilitzen o directament no s'utilitzen. És amb aquesta idea que neix aquest projecte. Desenvolupar una aplicació que tingui un cost raonable, que faci la funció d'integració del procés amb l'àrea de gestió de l'empresa amb la singularitat que estigui dimensionat d'acord amb l'escala i necessitats del client.

Abans de continuar, voldria fer un apunt a la terminologia que s'utilitza al llarg d'aquesta memòria per fer referència a aspectes, eines, termes, etc, exclusius d'aquest món de la fusta, els quals la traducció exacte al català, o no existeix, o no s'utilitza. Per exemple, és el cas del nom "parrilla", utilitzat per tothom per referir-se a la part superior que conforma un palet, quan en realitat el mot correcte seria graella o tapa. Així en moltes ocasions s'utilitzarà la terminologia pròpia del món del palet.

Les següents imatges (Figura 2.1) ens seran útils per identificar les diferents parts que conformen un palet, i així ajudar a fer el seguiment dels diferents apartats de la memòria, ja que sovint hi farem referència.

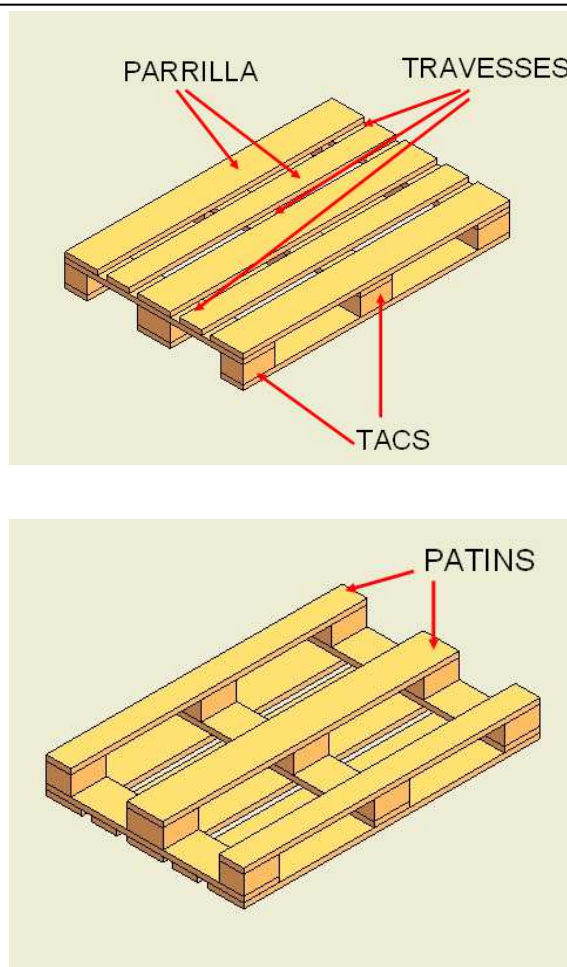


Figura 2.1: Imatges explicatives de les parts d'un palet.

2.1.- EVOLUCIÓ DE LA FABRICACIÓ INDUSTRIAL DE PALETS

Si fem una mirada retrospectiva, veurem que en els inicis de la fabricació dels palets no hi havia gaire automatització, sinó que era més aviat un treball manual, ja que s'utilitzaven els anomenats bancs de clavat (Figura 2.2), en el quals l'operari era el que ho feia tot. El treballador ficava les fustes dins unes guies de metall que li feien de referència, clavava la part superior del palet i posteriorment, girava el palet manualment per clavar la part inferior.

Tot i que encara s'utilitzen, no és el més habitual per l'elevat cost de producció, però sí que per segons quins tipus de palets especials de sèries curtes pot ser útil. Els bancs d'avui en dia s'han millorat i permeten fer la majoria de les accions necessàries per confeccionar el palet de manera semiautomàtica.



Figura 2.2: Imatge banc de clavats manual.

Posteriorment es van començar a implantar les línies semiautomàtiques (Figura 2.3), que consistien o consisteixen, en un carro de clavats que disposa de clavadors pneumàtics, en el qual l'operari ha de configurar la part superior del palet (la parrilla) i després el carro es desplaça clavant-la a unes posicions prefixades. Posteriorment, aquesta parrilla passa a una cinta transportadora en la qual un operari ensambla la parrilla prèviament clavada amb la part inferior (patí) de manera manual. Un cop el palet està acabat completament, s'envia a un apilador per tal de fer-ne una pila.



Figura 2.3: Imatge línia semiautomàtica

La següent evolució en les línies de clavats va ser la introducció de les línies automàtiques de cicle continu, pensades per fer sèries de quantitats elevades d'un mateix palet, ja que el temps d'ajustatge mecànic és considerable. Aquestes consisteixen en la substitució dels carros de clavats pneumàtic per clavadores hidràuliques (Figura 2.4), afegir els mòduls d'acabat dels palets (Figura 2.5) i les alimentacions automàtiques de tacs i fustes travesseres (Figura 2.6). Aquest va ser un canvi qualitatiu important, a nivell de producció, de flexibilitat, i de qualitat del palet, a part que et permetia aconseguir, al final de la línia, un palet completament acabat.



Figura 2.4: Imatge clavadora hidràulica



Figura 2.5: Imatge mòduls finalització del palet



Figura 2.6: Imatge alimentació automàtica de fustes travesseres

Més tard, es van començar a implantar línies de clavats automàtiques de canvi ràpid (Figura 2.7), les quals potencien la facilitat per part de l'usuari per fer el canvi de tipus de palet a fabricar de manera simple, amb els mínims ajustos mecànics en la clavadora i en tota la línia. Aquest és el principal canvi en referència a les línies de cicle continu que hem explicat amb anterioritat, és a dir, l'usuari únicament necessita fer la programació de la comanda i les dimensions del palet, a continuació la línia es regula automàticament a les noves dimensions. Aquest tipus de màquines estan altament recomanables per sèries curtes de palets.



Figura 2.7: Imatge clavadora de parrilles de canvi ràpid

Actualment, totes les tendències passen per inserir en les línies, ja siguin de cicle continu o de canvi ràpid, les noves tecnologies industrials i de la comunicació i la informació:

- Sistemes d'alimentació automàtics el més autònoms possibles, la majoria de les vegades mitjançant la robòtica (Figura 2.8).
- Sistemes de control de la qualitat mitjançant visió artificial (Figura 2.9).
- Sistemes de Disseny 3D del producte (Figura 2.10)
- Sistemes integrats de gestió de la producció.



Figura 2.8: Imatge cargador automàtic de fustes superiors amb robot



Figura 2.9: Imatge sistema de visió artificial pel control de qualitat.

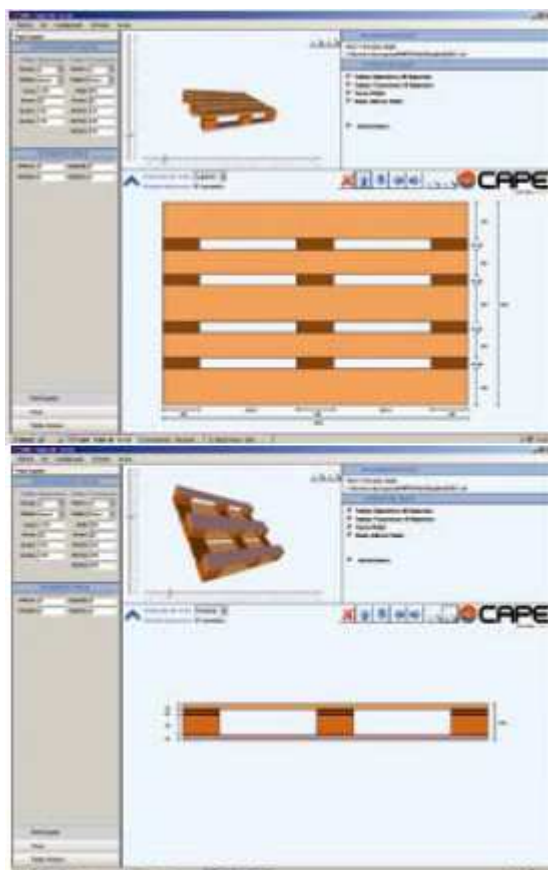


Figura 2.10: Imatge programa de disseny 3D de palets.

2.2.- LINIA AUTOMÀTICA DE CANVI RÀPID

Aquest projecte quedaria emmarcat dintre la família de línies automàtiques de canvi ràpid, però s'ha realitzat de manera que sigui flexible i es pugui integrar fàcilment en les línies de cicle continu.

Les diferents màquines que formen una línia automàtica de canvi ràpid són les següents:

- T3C**: Màquina automàtica de tall, amb sistema triple de retastat de tacs i mordassa individual per cada serra. Inclou alimentació automàtica de barres de dues amplades diferents.
- ATG2700**: Cinta automàtica d'alimentació i càrrega triple de tacs, amb sistema individual de càrrega per cada pinça.

- CPA**: Clavadora automàtica de parrilles, amb sistema de clavats per biela, rematxat, per palet de tac o de cabiró.
- CPE**: Clavadora automàtica de canvi ràpid, amb sistema de clavats per biela, quadraments del palet abans del clavats i sistema de càrrega de parrilles robotitzat. Dimensions màximes de palet 1400x1400 i mínimes de 600x600.
- VR**: Voltejador de palets rotatori per palets 1400x1400 a 600x600.
- CPI**: Clavadora automàtica de canvi ràpid amb sistema de clavats per biela, quadraments de fustes inferiors i bases perimetrals. Dimensions màximes de palet 1400x1400 i mínimes de 600x600.
- AMD**: Alimentador automàtic de fustes inferiors per la segona clavadora, de 1 fins a 9 fustes inferiors, d'una única dimensió i una doble cadena per fer palet perimetral.
- **Mòdul test**: Mòdul que permet expulsar palets defectuosos i reintroduir palets reparats a la línia.
- MMF**: Mòduls encarregats de fer els acabats del palet, i finalment fer una pila.

La següent figura il·lustra el procés de fabricació d'un palet, pas per pas.

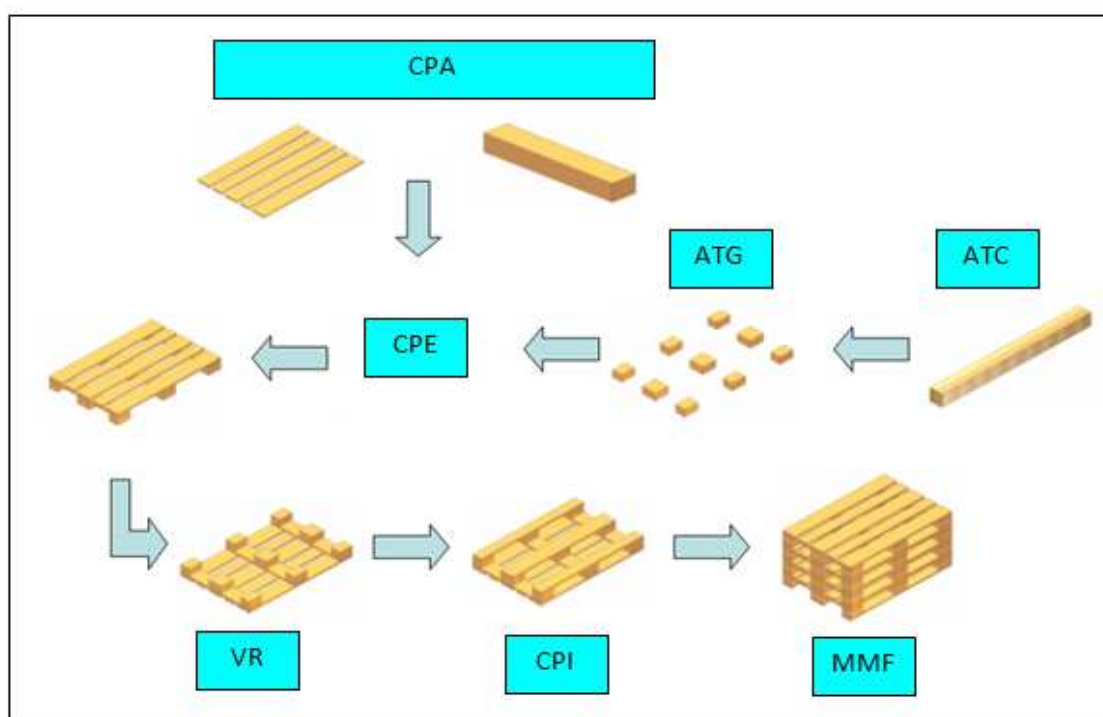


Figura 2.11: Diagrama explicatiu del procés de fabricació d'un palet en una línia de clavats automàtica.

En aquest projecte únicament ens centrarem en la part que comprèn la realització del programari d'adquisició de dades, tot i que també s'ha hagut de fer tota la programació, control, posta en marxa, etc, del resta de la línia, per tal d'aconseguir obtenir el producte acabat. Sobre aquest procés d'obtenció d'un palet al final de la línia, hem controlat i extret les variables de la producció com ja explicarem en els punts posteriors.

3.- MAQUINARI UTILITZAT

En aquest punt justificarem l'elecció dels dispositius de control així com els dispositius de comunicació industrial que hem utilitzat en el nostre projecte.

En la majoria dels dispositius no ha calgut fer una recerca exhaustiva sobre quin era el millor per desenvolupar la tasca d'aquest projecte, ja que s'han aprofitat els existents en la línia automàtica de palets ja comercialitzada per Mecànica Cape. Únicament, s'ha verificat que poguessin portar a terme la nova funció per aquest projecte.

Per a la realització d'aquest projecte i la seva posterior memòria, s'han deixat de banda tots els diferents dispositius a nivell d'actuadors, preactuadors, sensors etc. Sense els quals seria impossible el funcionament de la línia automàtica de clavats, ja que l'objectiu d'aquest treball és centrar-nos únicament en l'aplicació o programa creat i amb els diferents dispositius que s'han utilitzat per dur-ho a terme. També només farem referència al maquinari de l'última part del procés la que s'encarrega de gestionar el producte acabat.

Per a l'automatització i control del procés, s'ha utilitzat un autòmat (PLC) i un terminal tàctil (HMI). Aquests elements ara, a part de la funció de controlar el correcte funcionament del procés que ja tenien, hauran de recopilar i emmagatzemar la informació rellevant pel que fa a la producció. D'aquesta manera, es parteix dels mateixos recursos en quant a dispositius de control del procés, i únicament s'afegirà un o varis ordinadors (PC) amb sistema operatiu windows per a la gestió de les dades recopilades del procés. La connectivitat d'aquests podria ser mitjançant una connexió directa a la xarxa de camp o a través d'una connexió d'Internet

Les comunicacions entre els diferents elements són possibles gràcies a una targeta ethernet Modbus TCP/IP en el PLC i a un router, tal com mostra la següent imatge.

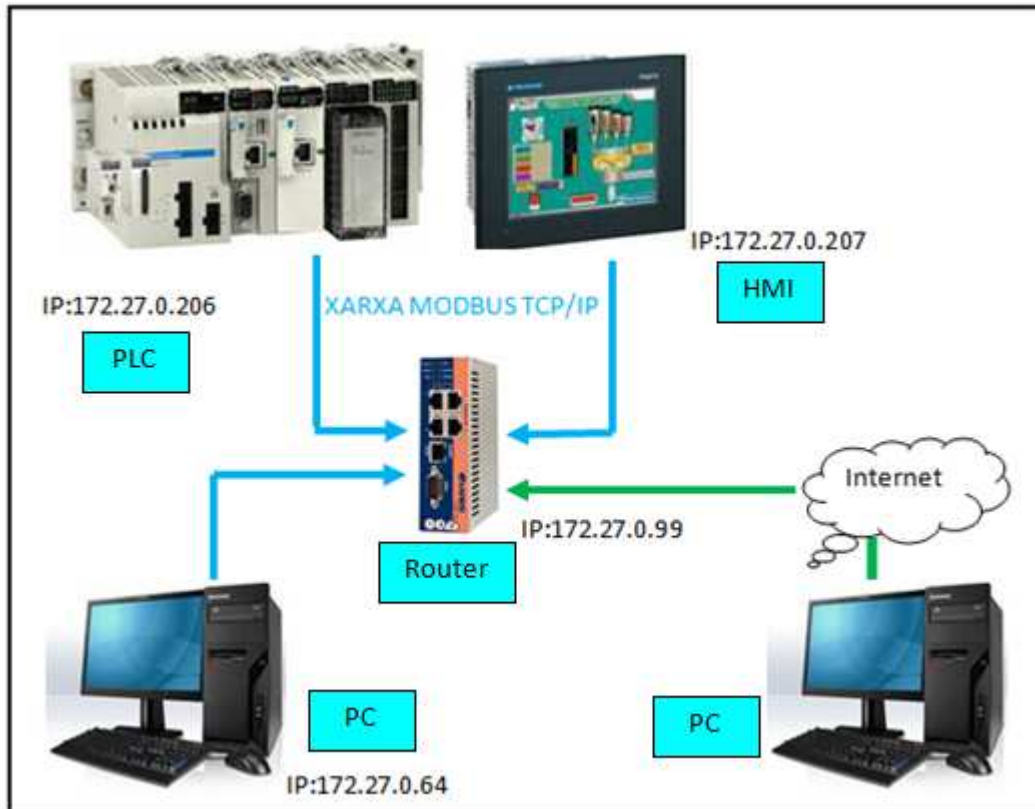


Fig 3.1 Diagrama de les comunicacions dels components de control en el projecte

3.1.- M340 CPU 2010

El PLC Modicon M340 és un autòmat modular, d'alt rendiment. Està especialment dissenyat per a maquinària complexa i infraestructures de tamany mig. El processador del Modicon M-340 és un ARM de 32 bits, amb una memòria RAM interna de 4 MB, 256 kB de dades y SD per backups de 8 MB. A més, disposa de múltiples variants i ampliacions per adaptar-lo a les necessitats de l'usuari.

Les principals característiques es resumeixen a la taula 3.1. D'aquestes, la més crítica era l'espai en memòria interna que ocupaven les diferents *arrays* on el PLC emmagatzema les variables de producció com ja explicarem en l'apartat dedicat al programa implementat en el PLC. La capacitat interna és de 256Kbytes, després de finalitzar el projecte encara ens queda un 28,6% d'espai lliure per futures ampliacions.

Característica			Disponible
Funciones	Número máximo de	Entradas/salidas binarias del bastidor	1.024
		Entradas/salidas analógicas del bastidor	256
		Canales expertos	36
		Canales Ethernet	2
		Bus de campo AS-i	BMX P34 2010: 0 BMX P34 20102: 4
		Comunicación simultánea EF	16
	Cantidad máxima de módulos	USB	1
		Puerto de enlace Modbus serie incorporado	1
		Puerto maestro CANopen incorporado	1
		Puerto Ethernet incorporado	-
Reloj de tiempo real que puede guardarse		Sí	
Capacidad de memoria de los datos de aplicación que puede guardarse		256 Kb	
Estructura de la aplicación	Tarea MAST		1
	Tarea FAST		1
	Procesamiento de eventos		64
Velocidad de ejecución del código de aplicación	RAM interna	100% booleano	8,1 Kins/ms (1)
		65% booleano + 35% digital	6,4 Kins/ms (1)
Tiempo de ejecución	Una instrucción booleana básica		0,12 µs
	Una instrucción digital básica		0,17 µs
	Una instrucción de coma flotante		1,16 µs

Taula 3.1 Principals característiques de l'autòmat M340 2010

3.2.- XBTGTO 5310

La pantalla XBTGTO 5310 és un terminal compacte i d'altres prestacions tant a nivell gràfic, com a nivell tècnic. A nivell de comunicacions destaca gràcies als seus tres ports, també cal subratllar una bona capacitat de memòria interna, ja que la seva Flash Eprom és de 96Mbytes. A més disposa de un *slot* per a una targeta SD de fins a 4GB.

En aquest projecte hem posat especial atenció en que les pàgines de control de la producció no sobrepassessin la capacitat de memòria de l'aplicació d'usuari del terminal.

Tipo de pantalla	Pantalla LCD de color TFT	
Tamaño de la pantalla	10.4"	
Resolución	640 x 480 píxeles (VGA)	
Área de visualización efectiva	Ancho 211,2 x Alto 158, 4 mm (8,31 x 6,24 in.)	
Colores de la pantalla	65.536 colores (Sin parpadeo) / 16.384 colores (Parpadeo)	
Retroiluminación	LED blanco (no reemplazable por el usuario. Cuando sea necesario reemplazarlo, póngase en contacto con su distribuidor local.)	
Duración de la retroiluminación	50.000 horas o más (funcionamiento continuo a 25 °C [77 °F] antes de que el brillo de la retroiluminación disminuya a 50%)	
Control de brillo	16 niveles (Ajustado con el panel táctil o el software)	
Fuentes de idiomas	ASCII: (Página de códigos 850) alfanuméricos (incluidos los caracteres europeos) Chino: (Códigos GB2312-80) fuentes de chino simplificado Japonés: ANK 158, Kanji: 6.962 (JIS estándares 1 y 2) (incluidos 607 caracteres no kanji) Coreano: (Códigos KSC5601 - 1992) fuentes Hangul Taiwanés: (Códigos Big 5) fuentes de chino tradicional	
Tamaños de los caracteres	Fuentes de 8 x 8, 8 x 16, 16 x 16 y 32 x 32 píxeles	
Tamaños de las fuentes	Puede aumentar el ancho y el alto hasta ocho veces. ¹	
Texto	8 x 8 píxeles	80 caracteres por fila x 60 filas
	8 x 16 píxeles	80 caracteres por fila x 30 filas
	16 x 16 píxeles	40 caracteres por fila x 30 filas
	32 x 32 píxeles	20 caracteres por fila x 15 filas

Taula 3.2 Principals característiques gràfiques del terminal XBTGTO

3.3.- Ewon Cosy

L'Ewon Cosy és un complet router industrial amb funcions d'enrutament entre la xarxa de la fàbrica i la de la màquina. Això permet crear un túnel VPN i habilitar una connexió remota des d'un ordinador amb internet de manera totalment transparent per la xarxa de la fàbrica. L'ordinador podrà accedir a tots els dispositius connectats a la xarxa de la màquina a través d'ethernet.

Aquesta funcionalitat l'hem utilitzat per poder accedir a les dades de la producció desde qualsevol lloc, i d'aquesta manera ser independents de tenir o no un ordinador dedicat per aquesta tasca en el departament de gestió i planificació de la producció del client.

3.4.- ORDINADOR PERSONAL (PC)

Finalment per poder dur a terme el nostre projecte, necessitem un PC on instal·lar-hi el nostre software. Aquest ordinador ha de tenir les següent característiques:

- Sistema operatiu Windows XP, o Windows 7 amb 32 o 64 bits, per poder garantir el correcte funcionament de l'aplicació, ja que ha estat compilada per aquestes plataformes.

- Office 2007 o superior instal·lat, per poder exportar les dades a un fitxer .xlsx.

- Disposar d' una targeta extra d'ethernet per poder-nos connectar a la xarxa de la línia automàtica de palets o d'una connexió a internet.

4.- PROGRAMARI UTILITZAT

En aquest apartat farem una breu pinzellada dels diferents programes que han permès dur a terme aquest projecte, deixant de banda els integrats dins del windows.

4.1.- UNITY V6.0

Entorn de Schneider pensat per a la programació dels seus dispositius lògics programables, de la sèrie M340. Permet realitzar la programació de diferents maneres, en aquest projecte les hem utilitzat totes i a continuació les resumirem breument.

La programació de contactes o "Ladder", és la més usual i implantada en el món industrial . Aquesta és la manera predominant del projecte, ja que és la més còmoda de treballar.

Una altra opció és la programació estructurada, en la qual l'usuari escriu mitjançant codi, les instruccions que es volen executar. És una forma de programar de més baix nivell, però alhora és més potent; i a part que ocupa menys espai de memòria, permet fer operacions amb registres de manera més fàcil, implantar-hi estructures condicionals, així com punters, etc

També es pot programar en Grafcet. Això té la seva part negativa en forma de despesa de recursos de la CPU però facilita la llavor del programador, tant en el disseny inicial com en la seva posterior depuració i posta en marxa.

Finalment, l'opció més interessant d'aquest software, és que permet implementar blocs de funció dintre el programa del PLC. D'aquesta manera es pot crear una part de programa de manera autònoma i que s'executi en funció d'unes inputs. Aquesta part de programa pot ser exportada i importada en d'altres PLCs i pel seu funcionament, únicament necessitem connectar-hi les inputs corresponents.

4.2.- VIJEO DESIGNER V 6.1

Aquest software, és l'utilitzat per fer el programa del terminal tàctil XBTGTO, és un entorn que disposa de totes les eines necessàries per crear una aplicació que ens ajudi a poder interactuar amb la nostra instal·lació, és a dir, que ens mostri dades i que a la vegada ens permeti controlar el procés a nivell d'usuari.

Una opció interessant i útil d'aquest software és que, a part de les típiques pàgines de control, on hi ha tots els objectes animats, registres de dades, visualitzadors, etc; permet crear "scripts", petits programes, que permeten realitzar determinades funcions, que d'altra manera la seva realització seria més complexa. En aquest projecte també els hem utilitzat.

4.3.- VISUAL BASIC 2008.NET

Visual basic.net (VB.NET) és un llenguatge de programació orientat a objectes, que es pot considerar una evolució del Visual Basic implementada sobre el framework .NET.

En concret, pel nostre projecte utilitzarem la versió Visual Basic 2008, aquesta versió està inclosa dins el paquet informàtic Visual Studio 2008.NET . Les seves principals característiques són:

- Possibilitat de treballar amb diferents .Net frameworks (2.0,3.0,3.5).
- Common Language Runtime (CLR) millorat.
- Ampliació de la biblioteca de classes del .NET framework.
- Utilització de ADO.NET, per dades i XML.
- Utilització de ASP.NET per formularis web i serveis web.
- Utilització del system.windows.form i system.drawing per crear la interfície d'usuari.
- Jerarquia d'espais de noms, classes i mètode.

4.4.- ECATCHER V4.0

Aquest programa propietat de eWON, permet crear, a través d'Internet, una connexió segura VPN (virtual private network) entre un usuari i un router EWON, mitjançant un servidor allotjat a Internet.

La línia de clavat, tal com ja hem explicat anteriorment, disposa d'una xarxa de comunicacions ethernet, en la qual està connectat un router EWON que li dóna accés a Internet. D'aquesta manera nosaltres podrem accedir a les dades de la producció des de qualsevol ordinador i desde qualsevol lloc, únicament necessitarem una connexió Internet i tenir instal·lat en el nostre ordinador el programari d'adquisició de dades i el eCatcher.

4.5.- WIRESHARK

El wireshark és un programari “espia”, que permet capturar paquets / trames de comunicació que circulin per una xarxa ethernet. D'aquesta manera es poden mostrar per pantalla de la manera més detallada per a la seva posterior anàlisi.

Nosaltres l'hem utilitzat en el moment de depurar la nostra aplicació ja acabada .Hem pogut fer un seguiment de les trames que s'estaven enviant a través de la xarxa Modbus TCP/IP i que comunicaven el PC amb el PLC. Aquest apartat s'explicarà més endavant amb més profunditat, en el capítol de posta en marxa i depuració.

5.- COMUNICACIÓ MODBUS TCP/IP

En aquest capítol aprofundirem en la teoria del bus de comunicacions sobre el qual es recolza aquest projecte. El Modbus és un protocol d'enllaç. Degut a això, pot implementar-se en diversos tipus de xarxes físiques on generalment cada fabricant sol proporcionar un software d'aplicació propi. Aquest protocol va ser desenvolupat per Modicom i en l'actualitat és d'accés lliure, una característica que l'ha convertit en un protocol àmpliament utilitzat.

Les seves principals característiques les descriurem en els punts següents.

5.1.- ESTRUCTURA DE XARXA

El medi físic de connexió pot ser un bus semidúplex (half dúplex) (RS485 o fibra òptica) o dúplex (full dúplex) (Rs422, BC 0-20mA o fibra òptica). La comunicació és asíncrona i les velocitats de transmissió van de 75 bauds a 19200 bauds. La màxima distància entre estacions depèn del medi físic, però pot arribar fins a un màxim de 1200m sense repetidors.

Per fer l'accés al medi, l'estructura lògica és del tipus mestre-esclau, on el número màxim d'esclaus és 63 i 1 mestre per un total de 64 elements. Els intercanvis de missatges poden ser de dos tipus:

- Intercanvis punt a punt que comparteixen sempre dos missatges: una petició del mestre i una resposta de l'esclau (pot ser simplement un reconeixement).

- Missatges difosos. Aquests consisteixen en una direcció unidireccional del mestre a tots els esclaus. Aquest tipus de missatges no solen tenir resposta per part dels esclaus i s'utilitzen per enviar dades comunes de configuració, inicialització, etc.

5.2.- PROTOCOL

La codificació de les dades dins d'una trama pot fer-se en mode ASCII o purament binari, segons l'estàndard RTU (remote transmission unit). En qualsevol dels dos casos, cada missatge obeeix a una trama que conté 4 camps principals, segons es mostra en la figura 1. Les úniques diferències entre les dues, és que la trama ASCII inclou una capçalera 3ª i els caràcter CR i LF al final del missatge.

També és diferent la forma per calcular la paraula de control d'errors (CRC), el format Rtu utilitza una fórmula polinòmica i no una suma en mòdul 16. A continuació es mostren cada una de les dues trames:

.	Nº esclau	Codi de	Subfuncions,	LRC(16)		CR	LF
(3Axh)	(00-3Fhx)	funció	dades	H	L	(0Dhx)	(0axh)

Nº esclau	Codi de	Subfuncions, dades	CRC(P16)
(00-3Fhx)	funció		H L

Taula 5.1. Trama genèrica en funció de la codificació utilitzada ASCII o RTU

5.3.- COMUNICACIÓ MODBUS TCP/IP

El Modbus TCP/IP és un protocol dissenyat per comunicar equips industrials dins una xarxa, utilitza la capa de transport TCP/IP. Va ser dissenyat per a la supervisió i control d'equips d'automatització. Específicament, el protocol utilitza missatges Modbus dins d'un entorn d'Internet o intranet utilitzant el protocol TCP/IP.

Així doncs, el Modbus TCP/IP és un estàndard industrial molt utilitzat a causa del seu baix cost, simplicitat, necessitats mínimes de maquinari i sobretot, que es tracta d'un protocol obert. Qualsevol sistema, processador, ordinador que utilitzi TCP/IP pot utilitzar el Modbus TCP/IP.

5.4.- CARACTERÍSTIQUES DEL MODBUS TCP/IP

El protocol Modbus TCP/IP està orientat a la connexió, és a dir, abans de transmetre les trames, la màquina d'origen i de destí han hagut d'establir un canal de comunicacions primer. Aquesta connexió pot enviar múltiples transaccions independents i, el fet d'utilitzar el TCP, permet establir gran nombre de connexions diferents.

La codificació de dades Modbus utilitza el format "big-endian", en el qual el byte més significatiu es troba primer. Per exemple 16 bits el 0x1234 seria 0x12 0x34. Els codis de funció que portin una quantitat variable de dades en la sol·licitud o en la resposta, han d'anar precedides d'un byte que informi de la longitud de dades que es transmeten.

5.5.- ESTRUCTURA DEL MODBUS TCP/IP

L'estructura general del protocol Modbus TCP/IP, o la manera d'encapsular les dades per fer una petició o una resposta, és la mateixa que en les altres variants de Modbus, ja sigui el Modbus serial amb codificació ASCII o RTU així com el Modbus plus. Les úniques diferències són les especificacions dels delimitadors inicial i final del missatge (framing), el patró de control d' errors i la interpretació de la direcció.

Les sol·licituds normalment són enviades en forma half-duplex (les dades poden viatjar en qualsevol direcció però no de manera simultània) sobre una connexió establerta. Això implica que no hi ha cap benefici d' enviar més d'una petició si encara estem pendents de la resposta, d'altra banda, però, es poden aconseguir altes ràtios de transferència de dades, ja que permet establir múltiples connexions en un mateix destí.

Aquesta tècnica de consulta/resposta encaixa perfectament amb la naturalesa Mestre/esclau del Modbus, afegint l'avantatge del determinisme que les xarxes ethernet ofereixen als usuaris en la indústria. D'aquesta manera amb el protocol obert Modbus sobre TCP proporciona una solució per la gestió desde pocs nodes fins a milles.

El camp de la direcció de l'esclau de Modbus és substituït per un byte identificador de la unitat, el qual pot ser utilitzat per comunicar a través de dispositius com gateways. Els missatges de sol·licitud i resposta en Modbus TCP contenen un prefixe compost per sis bytes com s'aprecia a la següent taula.

Ref	Ref	00	00	len
-----	-----	----	----	-----

Taula 5.2. Estructura del prefixe del Modbus TCP

Els elements "ref ref" inicials son els dos bytes del camp referència de transacció, és un número que no té valor en el servidor, però que són copiats literalment desde la sol·licitud de resposta. Aquest camp és útil per tal de poder establir més d'una connexió simultània amb diferents servidors i poder identificar cada una de les transaccions.

El tercer i quart byte representa l'identificador de protocol, un número que ha de tenir el valor 0 i el terme "len" especifica la longitud en número de bytes que

segueixen. La longitud és un valor de 2 bytes, però el byte alt forçosament té un valor 0 ja que el número màxim de bytes que es poden enviar per trama són 256.

Així doncs l'estructura del missatge Modbus TCP és la mostrada a continuació:

Posició del Byte	Significat
Byte 0	Identificador de transacció. Copiat per el servidor, Normalment té el valor 0
Byte 1	Identificador de transacció. Copiat per el servidor, Normalment té el valor 0
Byte 2	Identificador de protocol = 0
Byte 3	Identificador de protocol = 0
Byte 4	Longitud (byte alt) = 0
Byte 5	Longitud (byte baix) = al número de bytes que venen a conitnuació
Byte 6	Identificador de unitat (antigament la direcció)
Byte 7	Codi de funció Modbus
Byte 8 i posteriors	Les dades necessàries

Taula 5.3. Estructura del missatge Modbus TCP

En la següent taula es mostra el conjunt de funcions bàsiques i més utilitzades per les transaccions Modbus, el valor d'aquesta funció es troba en el byte 7.

Funció	Codi (hex)	Tasca
0	00xh	Control estació esclava
1	01xh	Lectura de n bits de sortides o interns
2	02xh	Lectura de n bits de entrades
3	03xh	Lectura de n paraules de sortides o internes
4	04xh	Lectura de n paraules d'entrades
5	05xh	Espectura de un bit
6	06xh	Espectura de una paraula
7	07xh	Lectura ràpida de 8 bits
8	08xh	Control dels comptadors de diagnòstic 1 a 8
9	09xh	No utilitzat
10	0Axh	No utilitzat
11	0Bxh	Control del comptador de diagnòstic 9
12	0Cxh	No utilitzat
13	0Dxh	No utilitzat
14	0Exh	No utilitzat
15	0Fhx	Espectura de n bits
16	10xh	Espectura de n paraules

Taula 5.4. Funcions Bàsiques i codis d'operació

5.6.- ENCAPSULAT DEL MODBUS TCP/IP

El Modbus TCP simplement encapsula una trama Modbus dintre d'un segment TCP. D'aquesta manera s'aconsegueix un servei de connexió fiable, cosa que a la pràctica significa que tota consulta espera una resposta, facilitant així el control d'errors del protocol desenvolupat.

La següent figura il·lustra com funciona l'encapsulat Modbus TCP/IP.

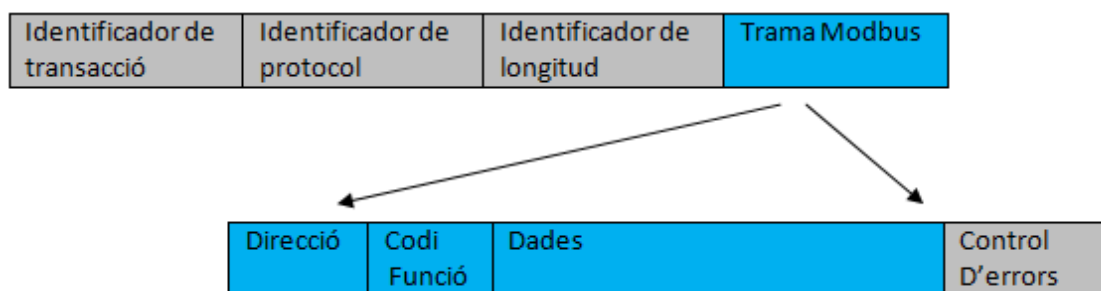


Figura 5.1. Esquema de l'encapsulat del Modbus TCP/IP

6.- CARACTERÍSTIQUES PROGRAMARI DESENVOLUPAT

En aquest apartat aprofundirem en les característiques de l'aplicació creada, i com ha millorat substancialment l'aplicació original existent. Posteriorment en parlarem més a fons, quan es descrigui el funcionament del programa.

En el moment de la realització d'aquest projecte, la línia de clavat de palets ja disposava d'una pantalla on es recopilava tota la informació referent a la producció. Però amb una funcionalitat bastant limitada, ja que només feia el càlcul dels valors acumulats absoluts de les variables de producció (temps de marxa, temps de paro, temps d'alarma, quantitat de producte fabricat i producció mitja). També amb la limitació que aquestes dades només es visualitzaven en el terminal operador i era impossible exportar-les cap a un altre dispositiu.



Figura 6.1 Pantalla producció abans de la realització del projecte

La nova aplicació a nivell d'usuari es pot dividir en dues parts:

-Aplicació creada en Visual Basic 2008 per ser instal·lada en un ordinador del departament de producció.

-L'aplicació de control de la producció de la línia creada en el terminal tàctil de l'operador.

El conjunt de les dues aplicacions, juntament amb el programa de control realitzat en el PLC, ens permet extreure tota la informació de la línia de clavat pel seu posterior anàlisi i control, amb un augment de prestacions considerable.

Les seves característiques principals són:

- Entorn gràfic millorat del panell operador.
- Visualització de la producció diària dels últims 25 dies.
- Càlcul dels valors acumulats absoluts de les variables de producció de cada dia (temps de marxa, temps de paro, temps d'alarma, quantitat de producte fabricat i producció mitja).
- Divisió de la producció diària segons els torns establerts.
- Divisió de la producció diària segons el tipus de palet fabricat.
- Exportació a un fitxer Excel de les variables de producció.
- Visualització i exportació de les alarmes actives de la línia.
- Exportació a un fitxer Excel de totes les alarmes durant un dia.
- Visualització i exportació de la producció diària dividida per hores.
- Exportació a un fitxer Excel de la producció diària dividida per hores.
- Crear gràfiques dels valors absoluts de les variables de producció dels últims 25 dies.
- Accessibilitat remota a través d'una connexió a Internet.

7.- PROGRAMACIÓ REALITZADA

En aquest projecte s'han fet diferents tipus de programes, per tal de dotar la maquinària del control necessari per realitzar les diferents funcions. Els podem agrupar bàsicament en tres grups: el programa del PLC, de la pantalla tàctil i l'aplicació en Visual Basic per un PC. Els dos primers actuen directament sobre el procés i l'últim s'encarrega de l'adquisició de dades de la producció.

7.1.- PROGRAMA PLC

La metodologia de programació del PLC ha sigut la de treballar amb blocs de funció DFB i amb estructures de dades DDT per tal de facilitar l'exportació del codi implementat en aquest projecte i importar-l'ho en d'altres projectes existents i així dotar-los de les noves millores i serveis creats en el nostre projecte.

Els blocs DFB, són parts de codi que ens permeten una major estructuració del nostre programa, ja que treballen de la mateixa manera que un bloc de les llibreries del Unity. Aquests blocs tenen l'avantatge que en el seu interior, enlloc d'existir un codi estàndard per a una funció donada, hi ha el codi creat per l'usuari per realitzar una funció específica. D'aquesta manera un usuari es pot crear una llibreria a mida per a una determinada funció i amb la particularitat que aquesta llibreria la pot reutilitzar en qualsevol altre projecte.

Una estructura de dades DDT és un tota una sèrie de variables elementals englobades en una sola entitat jeràrquicament superior. D'aquesta manera, en el nostre cas, hem pogut crear estructures de dades complexes partint de variables elementals simples.

Si posem com a exemple les dades de producció diària que recopilem, estaríem parlant de les variables que es mostren en la següent taula:

VARIABLE	FORMAT
Quantitat palets total dia	INT
Temps marxa total dia	INT
Temps paro total dia	INT
Temps alarma total dia	INT
Producció mitja total dia	REAL

Dia	INT
Any	INT
Quantitat palets total torn 1	INT
Temps marxa total torn 1	INT
Temps paro total torn 1	INT
Temps alarma total torn 1	INT
Producció mitja total torn 1	REAL
Quantitat palets total torn 2	INT
Temps marxa total torn 2	INT
Temps paro total torn 2	INT
Temps alarma total torn 2	INT
Producció mitja total torn 2	REAL
Quantitat palets total torn 3	INT
Temps marxa total torn 3	INT
Temps paro total torn 3	INT
Temps alarma total torn 3	INT
Producció mitja total torn 3	REAL

Taula7.1 Variables producció diària

En total estariem parlant de 22 variables, amb l'inconvenient que són de diferents formats, amb la qual cosa no podrien declarar-se conjuntament en una array, que seria el procediment habitual. També tenim el problema que, tal com ja s'ha explicat anteriorment, el nostre programa emmagatzema les dades de producció de 25 dies consecutius, amb la qual cosa el número de variables se'ns dispara a $22 \times 25 = 550$. Si aquestes 550 variables s'haguessin de definir individualment, seria un procés llarg i difícilíssim alhora de manipular-les. Això quedaria simplificat s'hi definim una estructura DDT (figura 7.1) amb totes les variables i posteriorment creem una array de 25 tipus de variable DDT creada (figura 7.2)

Nombre	Tipo
PROD_DIA	<Estruct.>
PROD_CPA_DIA_QNT_PALETS_TOTAL	INT
PROD_CPA_DIA_TEMPS_MRX_TOTAL	INT
PROD_CPA_DIA_TEMPS_PARO_TOTAL	INT
PROD_CPA_DIA_TEMPS_ALRM_TOTAL	INT
PROD_CPA_DIA_MITJA_TOTAL	REAL
PROD_CPA_DIA_DATA	INT
PROD_CPA_DIA_ANY	INT
PROD_CPA_DIA_QNT_PALETS_TORN_1	INT
PROD_CPA_DIA_TEMPS_MRX_TORN_1	INT
PROD_CPA_DIA_TEMPS_PARO_TORN_1	INT
PROD_CPA_DIA_TEMPS_ALRM_TORN_1	INT
PROD_CPA_DIA_MITJA_TORN_1	REAL
PROD_CPA_DIA_QNT_PALETS_TORN_2	INT
PROD_CPA_DIA_TEMPS_MRX_TORN_2	INT
PROD_CPA_DIA_TEMPS_PARO_TORN_2	INT
PROD_CPA_DIA_TEMPS_ALRM_TORN_2	INT
PROD_CPA_DIA_MITJA_TORN_2	REAL
PROD_CPA_DIA_QNT_PALETS_TORN_3	INT
PROD_CPA_DIA_TEMPS_MRX_TORN_3	INT
PROD_CPA_DIA_TEMPS_PARO_TORN_3	INT
PROD_CPA_DIA_TEMPS_ALRM_TORN_3	INT
PROD_CPA_DIA_MITJA_TORN_3	REAL

Figura 7.1 Estructura DDT que conté les variables de la producció de un dia

Nombre	Nº	Tipo
ARRAY_PROD_DIA		ARRAY[0..24] OF PROD_DIA
ARRAY_PROD_DIA[0]		PROD_DIA
PROD_CPA_DIA_QNT_PALETS_TOTAL		INT
PROD_CPA_DIA_TEMPS_MRX_TOTAL		INT
PROD_CPA_DIA_TEMPS_PARO_TOTAL		INT
PROD_CPA_DIA_TEMPS_ALRM_TOTAL		INT
PROD_CPA_DIA_MITJA_TOTAL		REAL
PROD_CPA_DIA_DATA		INT
PROD_CPA_DIA_ANY		INT
PROD_CPA_DIA_QNT_PALETS_TORN_1		INT
PROD_CPA_DIA_TEMPS_MRX_TORN_1		INT
PROD_CPA_DIA_TEMPS_PARO_TORN_1		INT
PROD_CPA_DIA_TEMPS_ALRM_TORN_1		INT
PROD_CPA_DIA_MITJA_TORN_1		REAL
PROD_CPA_DIA_QNT_PALETS_TORN_2		INT
PROD_CPA_DIA_TEMPS_MRX_TORN_2		INT
PROD_CPA_DIA_TEMPS_PARO_TORN_2		INT
PROD_CPA_DIA_TEMPS_ALRM_TORN_2		INT
PROD_CPA_DIA_MITJA_TORN_2		REAL
PROD_CPA_DIA_QNT_PALETS_TORN_3		INT
PROD_CPA_DIA_TEMPS_MRX_TORN_3		INT
PROD_CPA_DIA_TEMPS_PARO_TORN_3		INT
PROD_CPA_DIA_TEMPS_ALRM_TORN_3		INT
PROD_CPA_DIA_MITJA_TORN_3		REAL

Figura 7.2 Array creada dels 25 dies de producció

Un cop definides les variables que utilitzarem, s'ha prosseguit a crear els diferents DFB que hem utilitzat. En concret, hem creat 3 blocs nous:

- Producció Diària.
- Programació Horari Torns.
- Control Solapament Horari Torns.

Per facilitar la comprensió i la lectura d'aquest capítol, hem cregut convenient descriure la programació realitzada dins de cada bloc de funció, que correspondria a un codi "ladder" molt extens, mitjançant un diagrama de flux i una taula descriptiva de les entrades i sortides del bloc. De totes maneres per qualsevol consulta el codi complet està adjuntat en l'annex que acompanya aquesta memòria.

7.1.1.- DFB PRODUCCIÓ DIÀRIA.

El codi de programa que s'executa dins aquest bloc, té com a funció principal el càlcul estadístic de totes les variables de producció. Per realitzar aquesta tasca necessita com a entrades la informació de l'estat de la màquina.

També és necessària tota la informació referent als horaris dels torns de treball dels operaris, així com la referència i tipus de producte que s'està fabricant en cada moment determinat. Aquesta informació ens la subministra el terminal tàctil de l'operador.

Una altra dada molt important, és l'índex de producció que ens passa el programa en Visual Bàsic. Aquest índex és la sol·licitud de l'enviament del dia en concret que és vol rebre la informació de les dades productives.

La totalitat de les entrades i sortides amb les quals opera aquest bloc, són les mostrades en la taula T7.2.

ENTRADA	FORMAT	SORTIDA	FORMAT
Horari torns	DDT	Horari torns	DDT
Nivell usuari	DINT	Producció dia	DDT
Index produccio dia	INT	Produccio Cape	DDT
Index produccio Cape	INT	Producció recepta	DDT
Recepta torn	INT	Referència reecepta	DDT
Recepta acutal	DINT	Quantitat palets	UDINT
Reset produccio	BOOL	Quantitat palets Cape	UDINT
Reset proudccio Cape	BOOL		
Palet acabat	BOOL		
Màquina en marxa	BOOL		
Màquina en alarma	BOOL		
Quantitat palets recepta	UDINT		
Referencia recepta	String[32]		

Taula 7.2 Entrades i sortides del Bloc "Producció Diària"

A continuació, la figura 7.3 correspon al diagrama de fluxe de l'apartat del càlcul estadístic de la producció en un dia independentment del tipus de producte escollit. En el cas del càlcul per un producte i una referència específica el diagrama resultant, és similar en quant a contingut però amb moltes més iteracions, fet que dificultaria molt el seu seguiment. Per aquesta raó hem cregut que no era necessari adjuntar-l'ho.

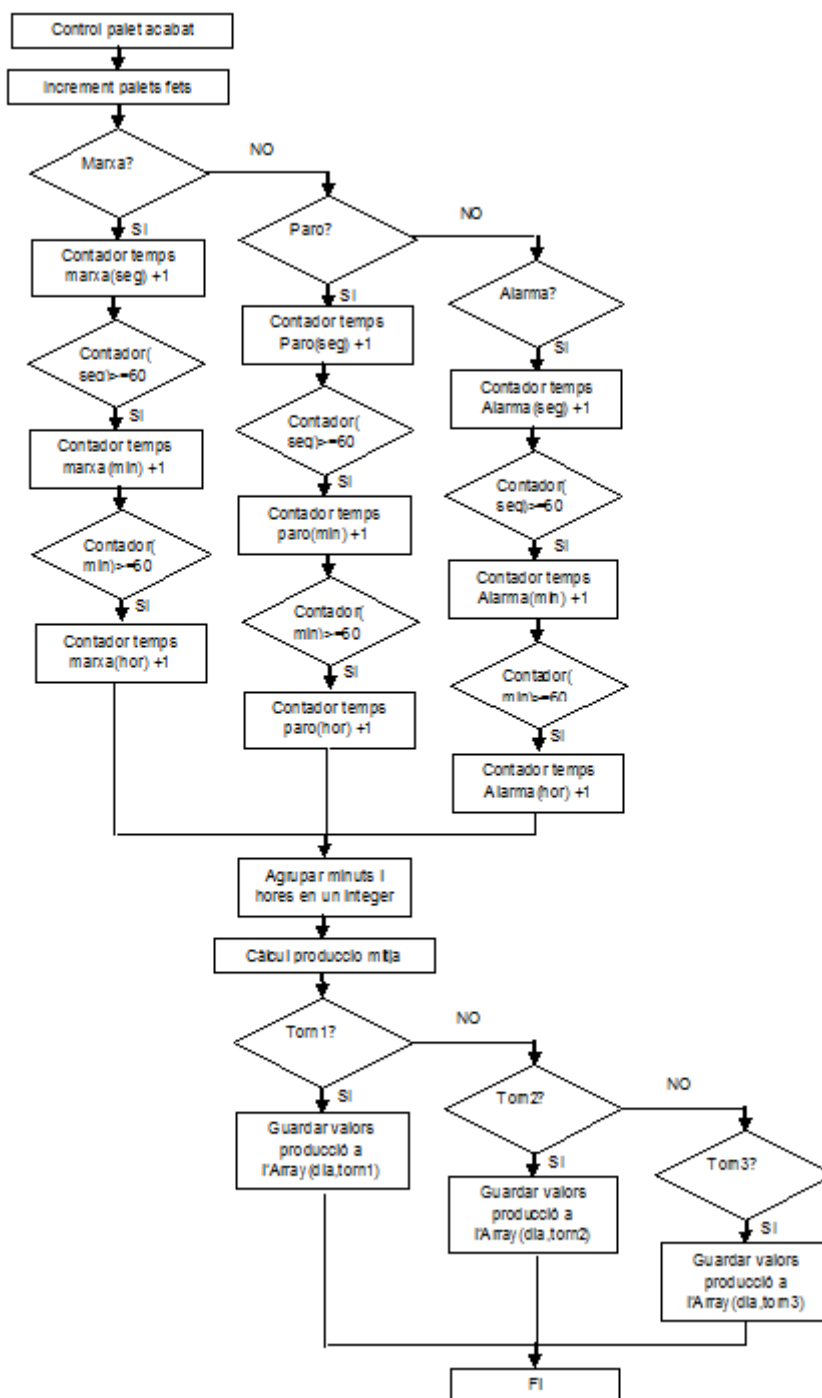


Figura 7.3 Diagrama de flux de la "Producció_dia"

7.1.2.- DFB PROGRAMACIÓ HORARI TORNS

Aquest bloc s'encarrega de, mitjançant l'hora interna del rellotge del PLC, saber en tot moment en quin torn de treball ens trobem. D'aquesta manera, les dades de producció s'emmagatzemen en l'àrea de memòria destinada al Torn 1, o al Torn 2, o el Torn 3.

Les entrades i sortides amb les quals opera aquest bloc, són les mostrades en la taula T7.3. i la figura 7.4 correspon al diagrama de flux.

ENTRADA	FORMAT	SORTIDA	FORMAT
Calcular	BOOL		
Horari torns	DDT	Horari torns	DDT

Taula 7.3 Entrades i sortides del Bloc "Programació_horari_Torns"

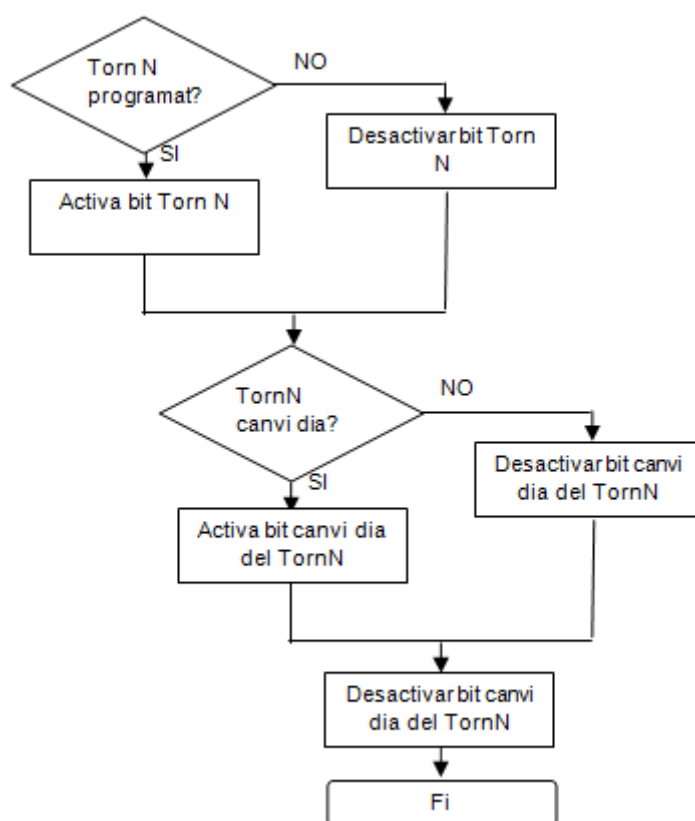


Figura 7.4 Diagrama de flux de la "Programació_horari_Torns"

7.1.3.- DFB CONTROL SOLAPAMENT HORARI TORNS

La missió d'aquest bloc és controlar la correcta programació dels horaris dels torns en el terminal tàctil, de manera que, en funció del horari seleccionat per un torn, s'evita que l'usuari cometí errors en la selecció del torn següent. També controla les hores de final d'un torn amb l'hora d'inici del torn següent, per tal d'evitar solapaments.

Les entrades i sortides amb les quals opera aquest bloc, són les mostrades en la taula T7.4. i la figura 7.5 correspon al diagrama de flux.

ENTRADA	FORMAT	SORTIDA	FORMAT
Calcular	BOOL		
Nivell usuari	DINT	Nivell usuari	DINT
Inici torn (minuts)	INT	Inici torn (minuts)	INT
Inici torn (hora)	INT	Inici torn (hora)	INT
Fi torn (minuts)	INT	Fi torn (minuts)	INT

Taula 7.4 Entrades i sortides Bloc "Control_Solapament_Horari_Torns"

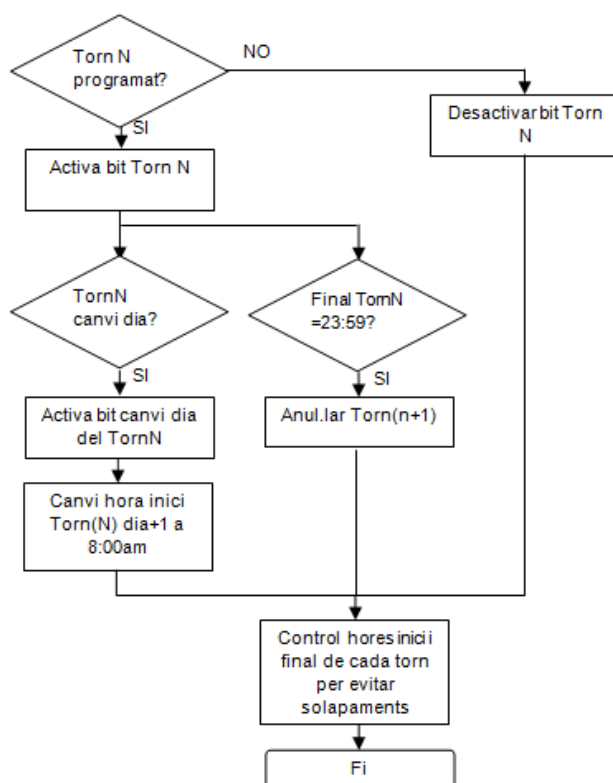


Figura 7.5 Diagrama de flux del "Control_Solapament_Horari_Torns"

7.2.-PROGRAMACIÓ TERMINAL TÀCTIL XBTGT5310

L'objectiu principal de l'aplicació amb el terminal tàctil XBTGT5310 és el de poder supervisar i controlar la línia de clavat. Es tracta d'una aplicació fàcil i agradable d'utilitzar, que permet controlar la línia sense grans complicacions, però d'una manera efectiva i que s'adapti sense problemes als diferents tipus de palets que s'han de fabricar.

Per poder realitzar la supervisió del procés, s'ha de produir un intercanvi d'informació continu entre el PLC i el terminal. El terminal necessita del PLC informació referent a la posició i estat de la màquina en tot moment, el contingut de les variables que s'han de visualitzar, i finalment les confirmacions de les ordres donades des de la pantalla, per poder representar-ho tot en el monitor. Mentre que el PLC necessita de la pantalla els valors dels paràmetres que intervénen en el procés i les ordres de control i comandament necessàries per poder funcionar correctament.

En el moment del començament d'aquest projecte l'aplicació existent en el terminal de la línia de clavat constava d'una sèrie de pàgines o pantalles, que permetien accedir a diferents funcions per interactuar amb el procés: pantalles de parametrització, de monitorització, d'alarmes, de programació, etc. Nosaltres, el que hem fet, és reestructurar les pantalles de producció per tal de dotar-les de més funcionalitat i afegir-hi les variables necessàries pel correcte funcionament del programari d'adquisició de dades.

L'aplicació realitzada té una extensió i complexitat considerable, per tant, no serà possible entrar a fons en l'explicació de tots els detalls de com s'han elaborat les pantalles i de quins controls s'han utilitzat, ja que sinó correm el risc de complicar molt la lectura d'aquesta memòria. Bàsicament, es tracta d'una pantalla on es monitoritza el rendiment de la línia, i on es poden visualitzar els valors de les variables de producció dels 25 dies emmagatzemats en el PLC dividides per torns (figura 7.6).



Figura 7.6 Pantalla principal de monitorització de la producció del terminal tàctil

També s'ha afegit una pantalla emergent, on l'encarregat de la fàbrica pot introduir els horaris dels torns assignats, i a més pot rebre informació detallada de com han funcionat tots els torns.



Figura 7.7 Pantalla de monitorització de la producció per torns

7.3.-PROGRAMACIÓ EN VISUAL BASIC

El nostre programa en Visual Basic per sistema operatiu Windows, està format per un conjunt d'objectes insertats en un formulari. Cada objecte té la seva funció i executa el seu codi associat.

Per tal de facilitar la lectura d'aquesta memòria i alhora ajudar a comprendre el funcionament de la interfície, abans d'entrar en el codi implementat descriurem els principals objectes utilitzats, on estan situats, quina és la seva funció associada, tipus de control VB, paràmetres utilitzats, etc. També per poder ajudar al seguiment de l'annex on hi ha el codi de tot el programa realitzat.

Principalment, l'aplicació disposa d'un menú inferior amb diferents opcions per l'usuari i d'un marc central on és visualitzen les diferents opcions escollides.

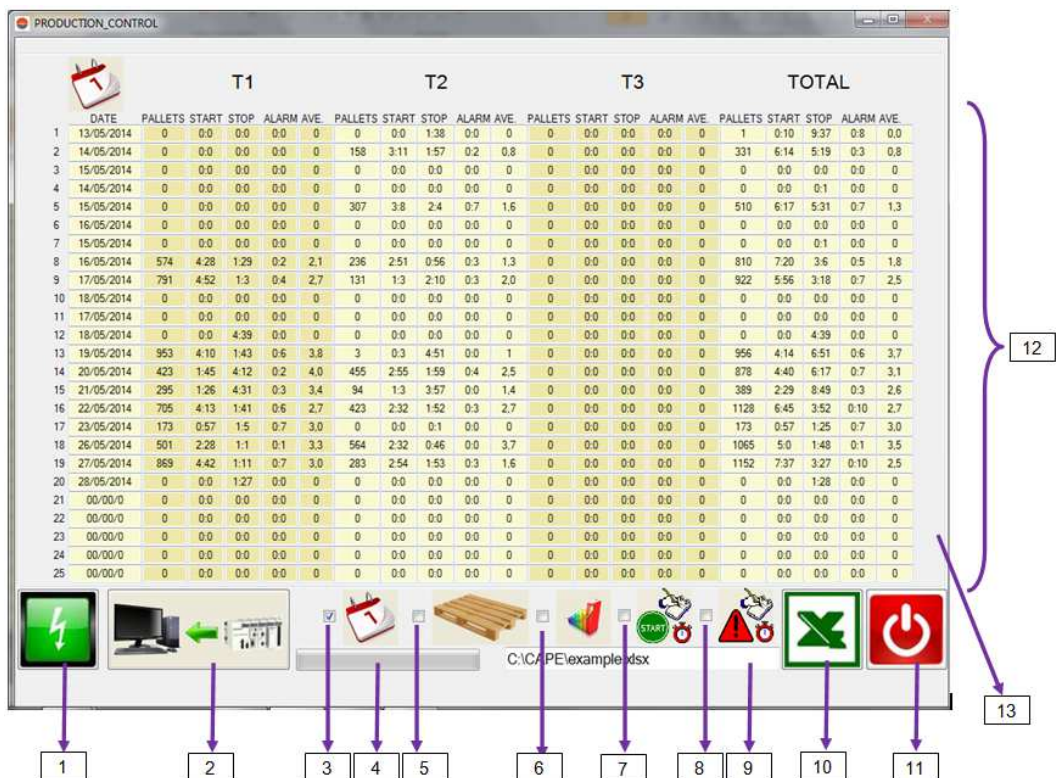


Figura 7.8 Principals controls de la interfície creada en VB(Imatge1)

	NOM OBJECTE	TIPUS	FUNCIÓ ASSOCIADA	CAMPS UTILITZATS
1	Buto_Conectar	Button	Buto_Conectar_Click	.click,.image,.cursor,
2	Buto_Llegir	Button	Buto_Llegir_Click	.visible,.click
3	Chk_Dia	CheckBox	Chk_Dia_CheckedChanged	.checkedchanged, .checked,.visible
4	ProgressBar1	ProgressBar	-	.value,.visible, .refresh()
5	Chk_Recepta	CheckBox	Chk_Recepta_CheckedChanged	checkedchanged, .checked,.visible
6	Chk_Grafiques	CheckBox	Chk_Grafiques_CheckedChanged	checkedchanged, .checked,.visible
7	Chk_Prod_Hores	CheckBox	Chk_Prod_Hores_CheckedChanged	checkedchanged, .checked,.visible
8	Chk_Alarmes	CheckBox	Chk_Alarmes_CheckedChanged	checkedchanged, .checked,.visible
9	Txt_Path	TextBox	-	.visible, .text
10	Buto_Excel	Button	Buto_Excel_Click	.click,.visible
11	Buto_Sortida	Button	Buto_Sortida_Click	.click,.cursor
12	Varis Txt_prod...	TextBox	-	.text
13	Gbox_Prod_Diari	GroupBox	-	.visible

Taula 7.5 Principals controls de la interfície creada en VB(Taula1)

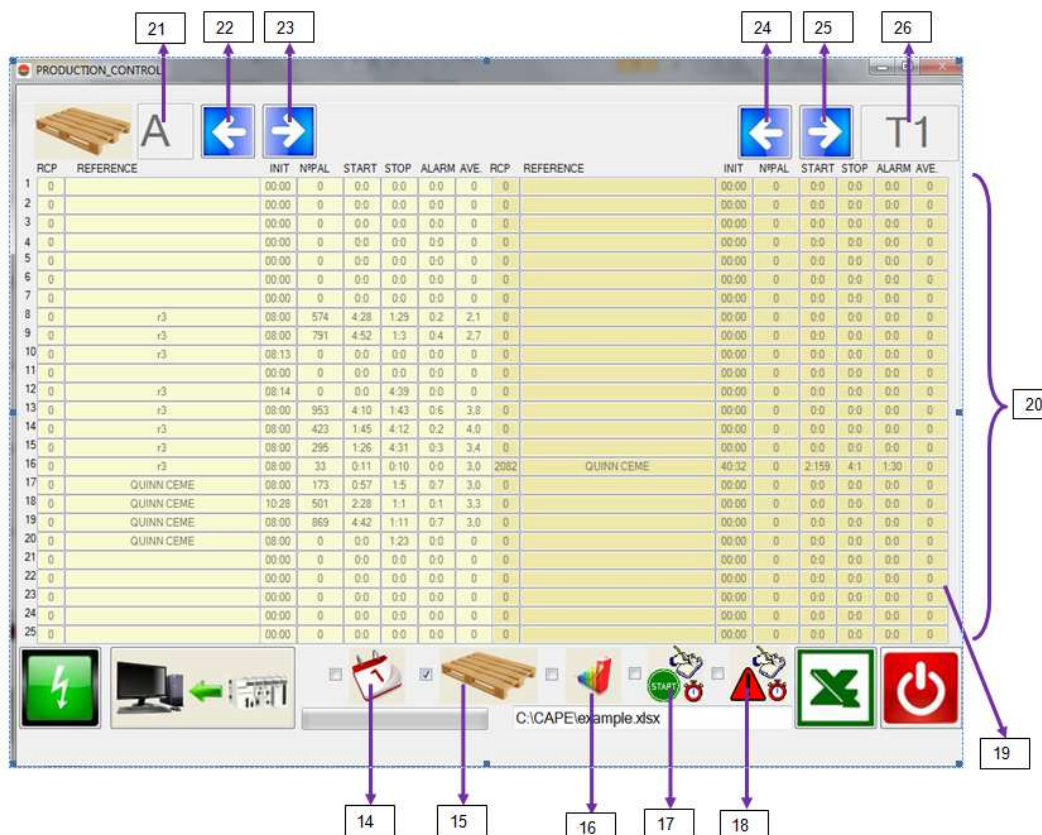


Figura 7.9 Principals controls de la interfície creada en VB(Imatge2)

	NOM OBJECTE	TIPUS	FUNCIÓ ASSOCIADA	CAMPS UTILITZATS
14	PictureBox1	PictureBox	-	-
15	PictureBox2	PictureBox	-	-
16	PictureBox3	PictureBox	-	-
17	PictureBox4	PictureBox	-	-
18	PictureBox5	PictureBox	-	-
19	Gbox_Receptes	GroupBox	-	.visible
20	Txt_prod_rec...	TextBox	-	.text
21	Txt_Indic_Bloc	TextBox	-	.text, .visible
22	Buto_Esq_Bloc	Button	Buto_Esq_Bloc_Torn_Click	.click,.visible
23	Buto_Drt_Bloc	Button	Buto_Drt_Bloc_Torn_Click	.click,.visible
24	Buto_Esq_Torn	Button	Buto_Esq_Num_Torn_Click	.click,.visible
25	Buto_Drt_Torn	Button	Buto_Drt_Num_Torn_Click	.click,.visible
26	Txt_Indic_Torn	TextBox	-	.text, .visible

Taula 7.6 Principals controls de la interfície creada en VB(Taula2)

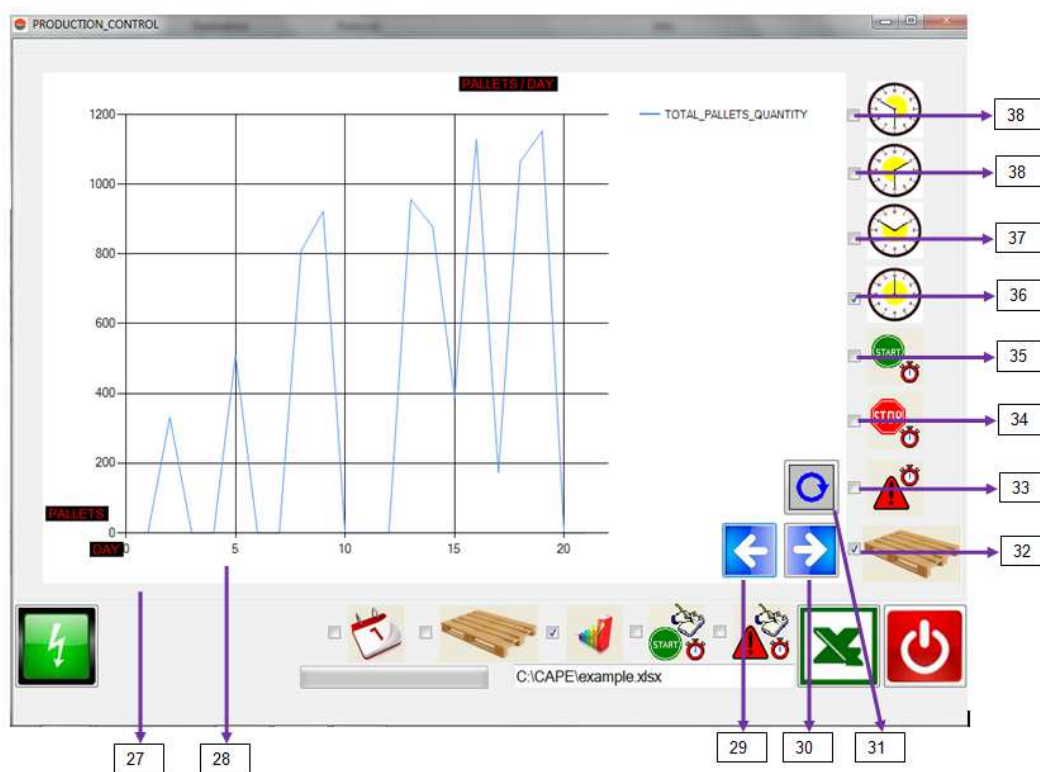


Figura 7.10 Principals controls de la interfície creada en VB(Imatge3)

	NOM OBJECTE	TIPUS	FUNCIÓ ASSOCIADA	CAMPS UTILITZATS
27	Gbox_Grafica	GroupBox	-	.visible
28	Grafica_Tot	Chart	-	.series.clear(), .add(),.chartType,.points, .resumeUpdates()
29	Buto_Zoom_Esq	Button	Buto_Zoom_Graf_Esq_Click	.click
30	Buto_Zoom_Drt	Button	Buto_Zoom_Graf_Drt_Click	.click
31	But_Refresc	Button	But_refresc_Grafica_Click	.click
32	Chk_Qnt_Palets	CheckBox	-	.checked
33	Chk_Alarma_Graf	CheckBox	-	.checked
34	Chk_Graf_Paro	CheckBox	-	.checked
35	Chk_Graf_Marxa	CheckBox	-	.checked
36	Chk_Graf_Tot	CheckBox	-	.checked
37	Chk_Graf_T3	CheckBox	-	.checked
38	Chk_Graf_T2	CheckBox	-	.checked
39	Chk_Graf_T1	CheckBox	-	.checked

Taula 7.7 Principals controls de la interfície creada en VB (Taula3)

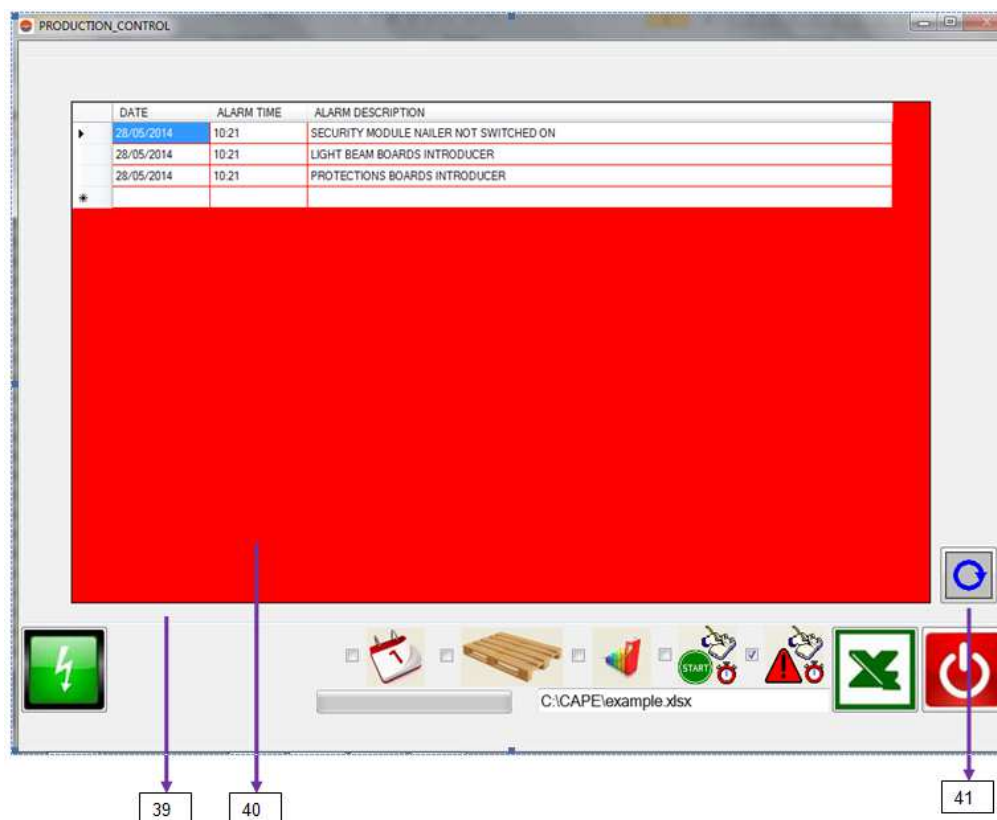


Figura 7.11 Principals controls de la interfície creada en VB(Imatge4)

	NOM OBJECTE	TIPUS	FUNCIÓ ASSOCIADA	CAMPS UTILITZATS
39	Gbox_Alarmes	GroupBox	-	.visible
40	Taula_Alarmes	DataGridView	-	.columns(),.width, .format, .DeafultCellStyle, .DataSource, .DataMember
41	Buto_Refresc_Alarmes	Button	Buto_Refresc_Alarmes_Click	.click

Taula 7.8 Principals controls de la interfície creada en VB(Taula4)

Fins aquest punt, només hem fet la introducció del programari desenvolupat. En els subapartats posteriors donarem una visió específica del procediment a seguir per part de l'usuari per utilitzar l'aplicació creada. Quins passos s'han de seguir, per quin ordre, quina informació s'extreu del procés, com es visualitza i finalment, com podem exportar-la en un altre format.

7.3.1.- ESTABLIR CONNEXIÓ AMB EL PROCÉS

Un cop s'executa el programa per primera vegada, apareix el menú principal com mostra la següent imatge (Figura 7.12). Llavors, l'usuari, el primer que haurà de fer és prémer l'objecte "Buto_conectar" i automàticament s'executarà el codi associat a l'event "click". La figura 7.13 correspon al diagrama de flux de la funció "Buto_Conectar_Click".

El resultat d'aquesta funció és saber si l'aplicació del PC ha pogut establir la comunicació amb el PLC. En cas afirmatiu, es podrà prosseguir operant amb el programa, d'altra banda, si per algun motiu és impossible establir un canal de comunicació entre el PLC-PC, es dona un missatge informatiu per pantalla i es finalitza l'execució del programa.

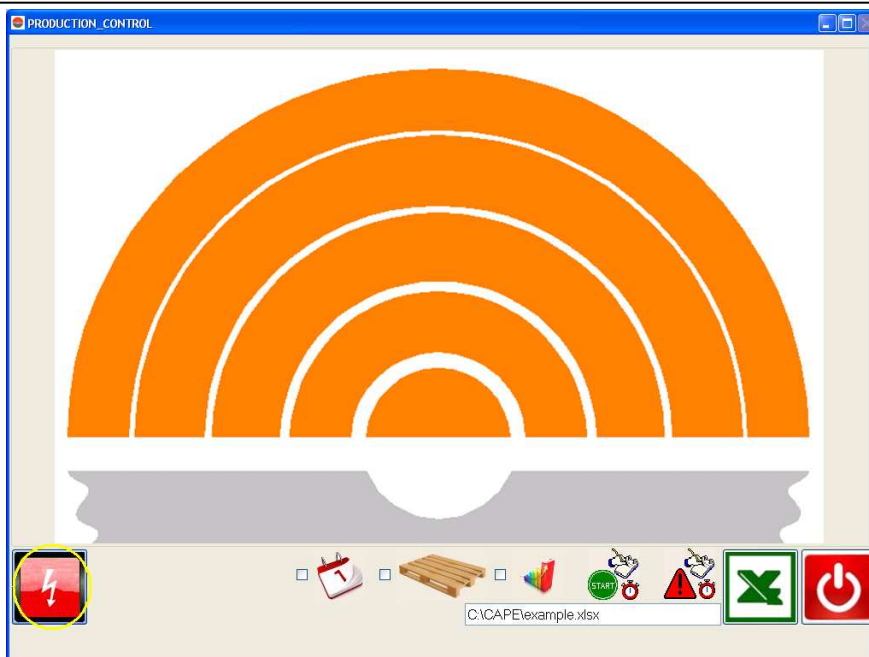


Figura. 7.12 Menú principal de l'aplicació

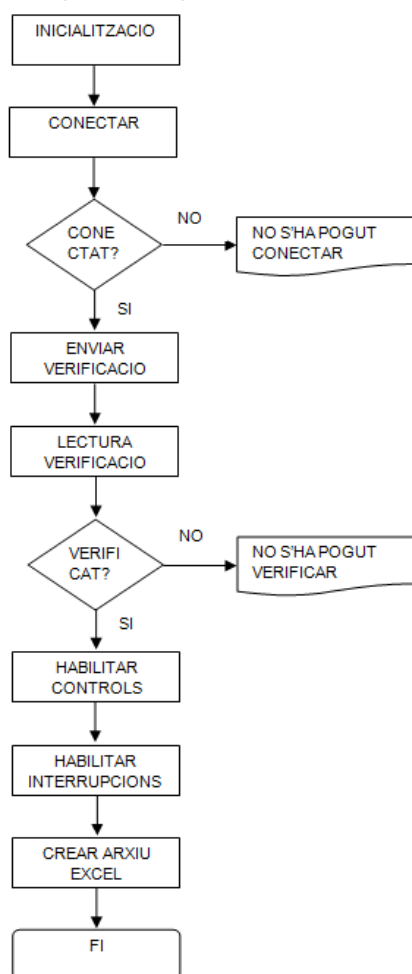


Figura. 7.13 Diagrama de flux de la funció Buto_Conectar_Click

7.3.2.- LECTURA DE LES DADES PRODUCTIVES DEL PROCÉS (OPCIÓ DIA)

Un cop verificat que el canal de comunicacions entre el PC i el PLC, funciona correctament, el segon pas seria fer una petició de lectura de les dades del procés al PLC. Per fer-ho, activarem el component "Chk_Dia" o el "Chk_Recepta" segons si volem fer una lectura de les dades de la producció únicament per dia o si volem obtenir informació sobre quin producte concret s'ha fabricat. Finalment, a l'igual que en el cas anterior, l'usuari haurà de prémer un botó, en aquest cas el "Buto_llegir" que farà executar l'event "botó_llegir_Click". Tal com mostra la figura 7.14. També s'adjunta en la figura 7.15 el diagrama de flux associat a la funció "botó llegir_Click".

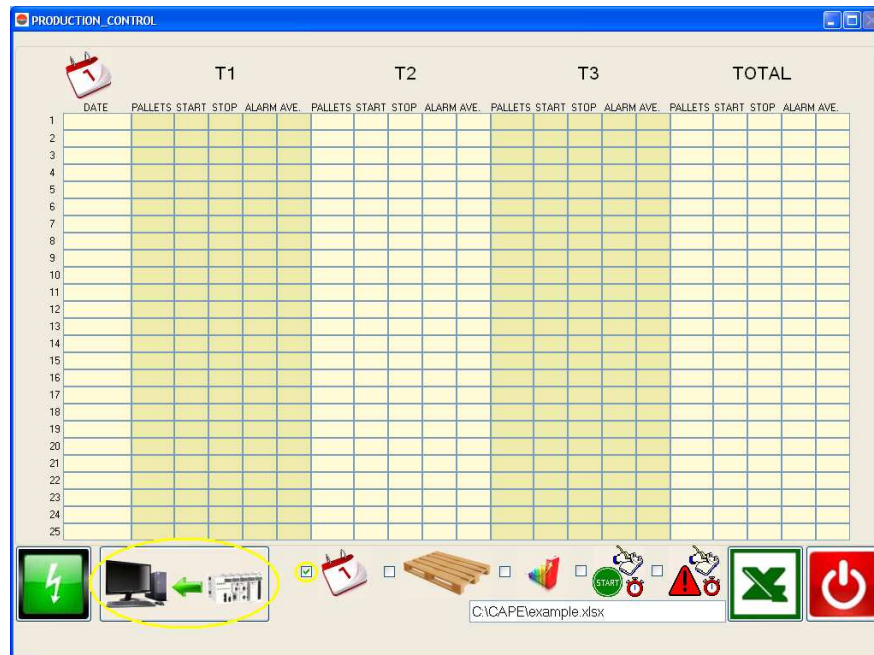


Figura 7.14 Selecció lectura informació de la producció per dia.

El resultat d'aquesta funció és l'obtenció de les dades emmagatzemades en el PLC. Per fer-ho s'utilitzen diferents bucles iteratius que realitzen les peticions de lectura dels diferents registres, mitjançant un índex de lectura que gestiona el PC i passa cap al PLC.

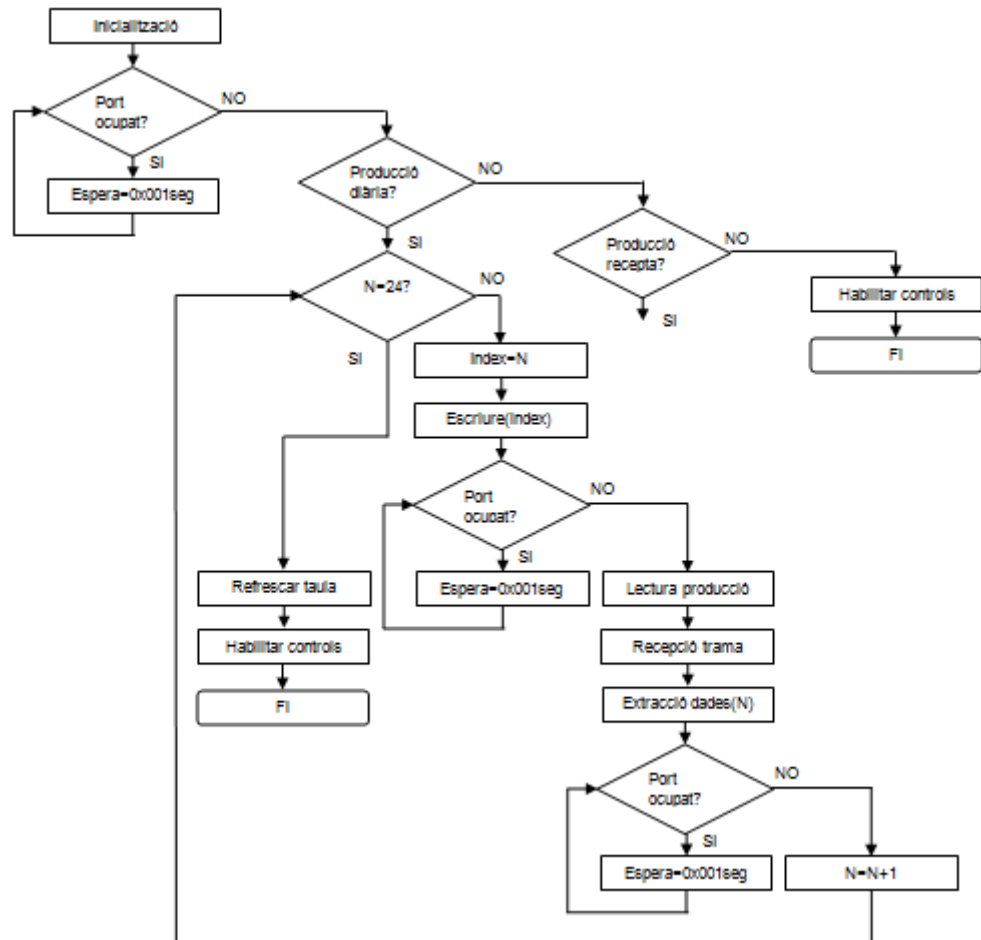


Figura. 7.15 Diagrama de flux de la funció Buto_Llegir_Click (opció Dia)

Dins de la funció “botó_Llegir_Click” hi ha d’altres subfuncions, les quals no tenen cap component associat, ja que són crides que realitza la funció anterior. Una de les més important és la de “envio_comando_escritura” i “envio_comando_lectura” que s’encarrega de la gestió de les trames que s’envien i es reben del PLC. La figura 7.16 mostra el diagrama de flux d’una d’elles, en concret del “Envio_Comando_Escritura”

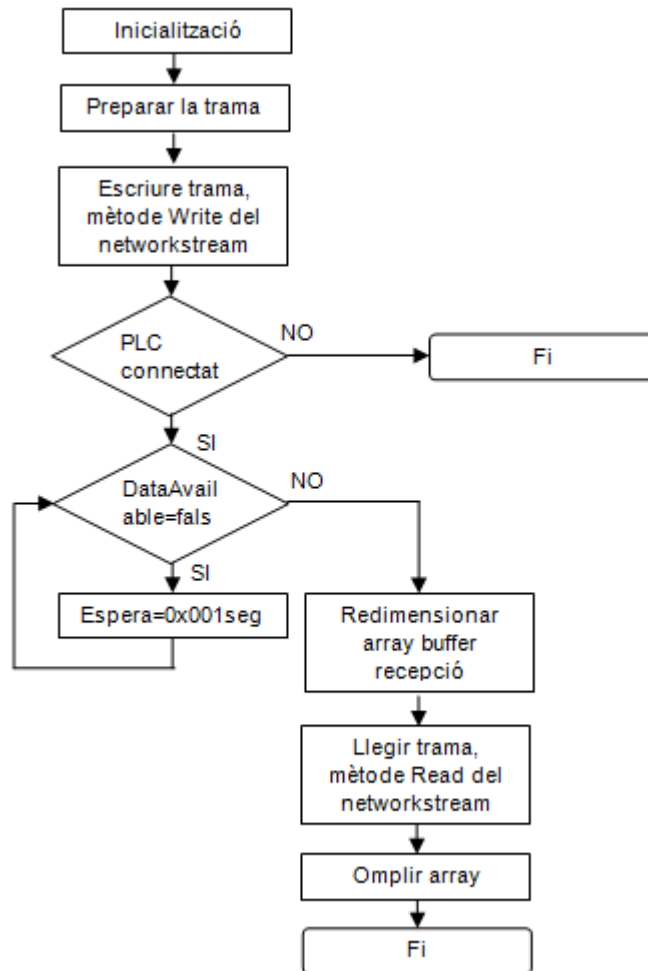


Figura. 7.16 Diagrama de flux de la funció Envio_comando_escriptura

7.3.3.- EXTRACCIÓ DE LES DADES PRODUCTIVES LLEGIDES DEL PROCÉS (OPCIÓ DIA)

Aquest subapartat no seria un pas independent dins el funcionament del programari sinó que estaria lligat de manera indivisible amb el subapartat anterior, però a efectes d'una millor estructuració de la memòria s'ha separat. Això és així, ja que tan l'extracció de les dades com la seva posterior monitorització serien subfuncions de la funció jeràrquicament superior "Buto_Llegir_Click".

La figura 7.17 correspon al diagrama de la funció que extreu les dades emmagatzemades en el buffer de recepció.

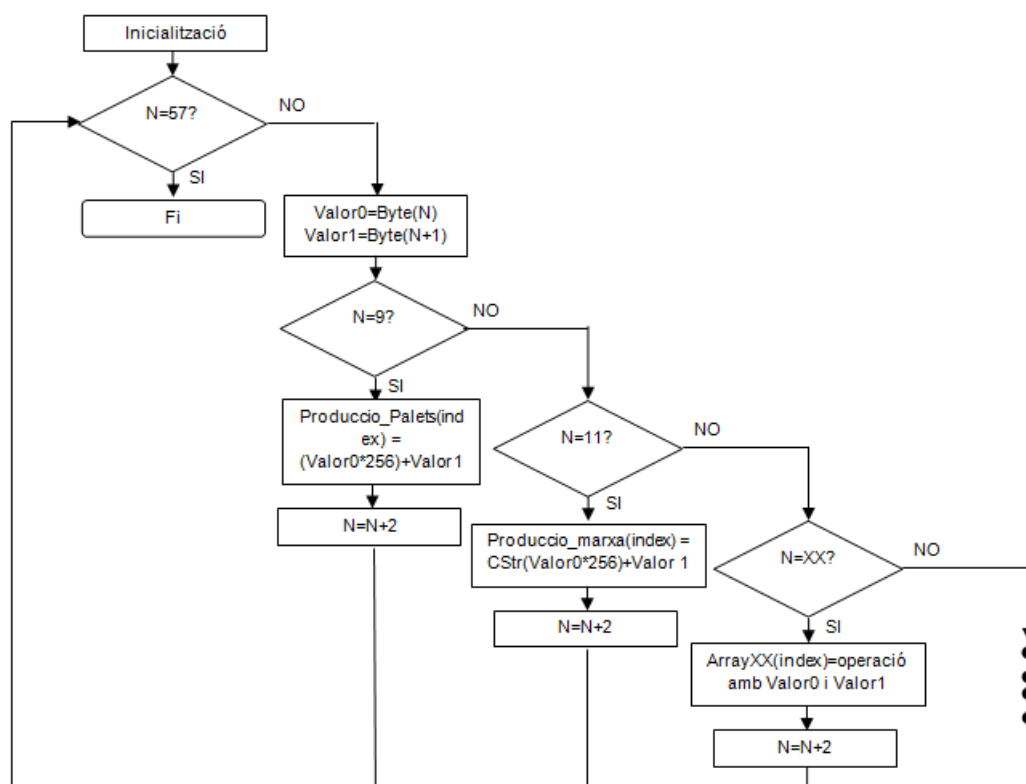


Figura 7.17 Diagrama de flux de la funció “Extracció_dades_dia”

Un cop tenim les dades de la producció guardades en les variables internes del Visual Basic, hem de mostrar-les en el formulari de la nostra aplicació. Per fer-ho utilitzem un objecte “Gbox_prod_diaria”, on en el seu interior hem creat una taula de “N” objectes “TextBox” cadascún dels quals correspòn a una variable de producció d’un dia i un torn específic. Això es pot observar en la figura 7.18. Per omplir amb els valors corresponents els diferents “TextBox” hem fet servir la rutina “refrescar_taula_dia”. El diagrama de flux d’aquesta funció és el mostrat en la figura 7.19.

7.3.4.- LECTURA I EXTRACCIÓ DE LES DADES PRODUCTIVES LLEGIDES DEL PROCÉS (OPCIÓ RECEPTES)

Si l'usuari hagués seleccionat l'opció de rebre les dades de producció amb la informació afegida de quin producte específic s'ha produït en la línia, s'hauria hagut d'activar el component "Chk_Recepta", llavors totes les rutines de funcionament del programari que hem descrit en els punts anteriors, "Buto_Llegir_Click", "Refrescar_Taula", "Envio_comando_escriptura/lectura", "Extracció_Dades_receptes", serien lleugerament diferents. També s'utilitzarien diferents components "Gbox_Prod_Receptes" i "textBox" per visualitzar els resultats.

RCP	REFERENCE	INIT	NPAL	START	STOP	ALARM	AVE	RCP	REFERENCE	INIT	NPAL	START	STOP	ALARM	AVE
1	GLANBIA	08:00	289	2:50	3:10	0:0	1.7	0		00:00	0	0:0	0:0	0:0	0
2	COPELAND L	08:00	0	0:1	0:24	5:35	0	0		00:00	0	0:0	0:0	0:0	0
3	COPELAND L	08:00	0	0:13	1:2	0:29	0	2372	SAICA PACK	40:44	0	2:87	1:9	0:56	0
4	EC NI 117	08:00	824	3:31	0:54	1:35	3.9	0		00:00	0	0:0	0:0	0:0	0
5	INVISTA HI	08:00	268	1:18	2:21	0:49	3.4	4656	LYCRA	39:FD	0	0:95	0:48	0:28	0
6	LYCRA	08:00	677	3:27	0:58	1:35	3.2	0		00:00	0	0:0	0:0	0:0	0
7	STOCK PALL	08:00	193	2:23	0:37	0:17	1.3	4375	r14	40:00	0	0:4	0:2	0:0	0
8	STOCK PALL	08:00	0	0:1	1:11	0:49	0	4098	GLANBIA-OF	40:63	0	2:47	2:37	0:52	0
9	GLANBIA-OF	08:00	1266	5:12	0:46	0:2	4.0	0		00:00	0	0:0	0:0	0:0	0
10	GLANBIA-OF	08:00	1051	3:36	0:28	0:10	4.8	0		00:00	0	0:0	0:0	0:0	0
11	GLANBIA-OF	08:00	21	0:10	0:37	1:13	2.0	4097	GLANBIA ID	40:2F	0	1:210	2:50	0:50	0
12	GLANBIA ID	08:00	1023	4:38	1:0	0:22	3.6	0		00:00	0	0:0	0:0	0:0	0
13	GLANBIA ID	08:00	608	4:4	0:34	1:21	2.4	0		00:00	0	0:0	0:0	0:0	0
14	CAUSEWAY P	08:00	1190	5:2	0:47	0:11	3.9	0		00:00	0	0:0	0:0	0:0	0
15	DERRYADD P	08:00	718	3:32	2:17	0:11	3.3	0		00:00	0	0:0	0:0	0:0	0
16	DERRYADD P	08:00	769	3:47	1:7	0:1	3.3	4695	STOCK PALL	40:88	0	1:7	0:46	0:16	0
17	STOCK PALL	08:00	1191	4:30	1:1	0:29	4.4	0		00:00	0	0:0	0:0	0:0	0
18	STOCK PALL	08:00	0	0:2	0:32	0:50	0	2342	QUINN GLAS	40:81	0	3:44	3:20	1:3	0
19	QUINN GLAS	08:00	164	0:57	1:5	0:24	2.8	4134	GLANBIA C	40:67	0	2:106	2:51	0:19	0
20	GLANBIA C	08:00	1154	4:55	0:51	0:14	3.9	0		00:00	0	0:0	0:0	0:0	0
21	GLANBIA C	08:00	953	4:1	1:50	0:8	3.9	0		00:00	0	0:0	0:0	0:0	0
22	GLANBIA C	08:02	0	0:0	0:0	5:58	0	0		00:00	0	0:0	0:0	0:0	0
23	GLANBIA C	08:00	1073	4:33	1:13	0:13	3.9	0		00:00	0	0:0	0:0	0:0	0
24	GLANBIA C	08:00	0	0:3	1:58	0:54	0	4182	FREDERICK	40:35	0	0:129	0:46	0:58	0
25	COPELAND L	08:00	961	4:44	1:10	0:6	3.4	0		00:00	0	0:0	0:0	0:0	0

Figura 7.20 Mostra els diferents objectes "Texbox" que visualitzen les dades de la producció diària per recepta.

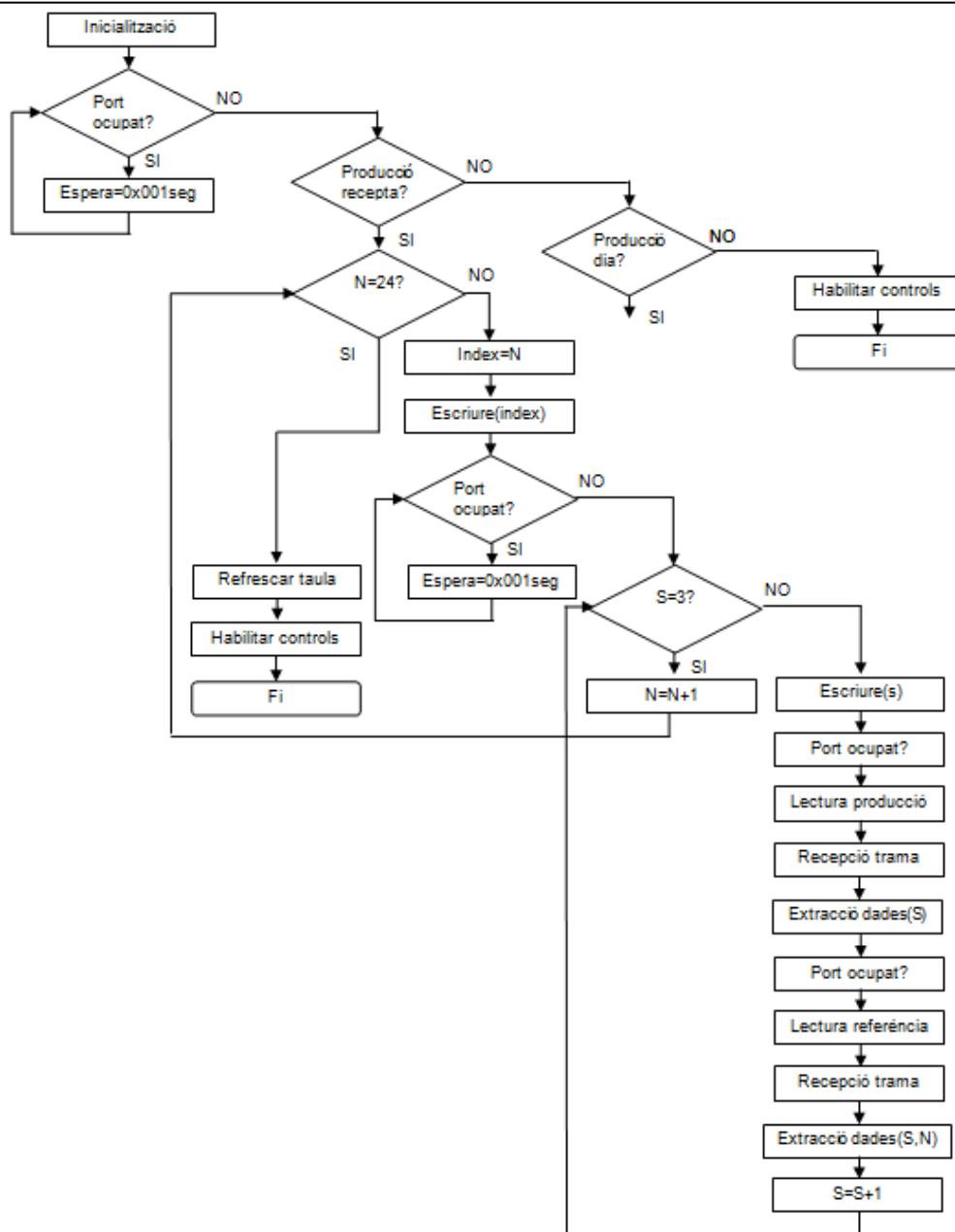


Figura. 7.21 Diagrama de flux de la funció “Buto_llegir_Click” (opció Recepta)

7.3.5.- MONITORITZACIÓ GRÀFICA DE LES DADES DE PRODUCCIÓ

Finalitzada la recepció de les dades del procés que ha enviat el PLC cap al nostre PC i que hem mostrat per pantalla, el programa dona l'opció de treballar amb aquestes dades i graficar-les per tal d'analitzar-les per la recerca de tendències, anomalies, etc. Per fer-ho, l'usuari ha de seleccionar el component “Chk_grafica” per tal d'habilitar el component “GBox_Grafica” i, a partir d'aquí,

operar amb les diferents seleccions del menú contextual de la dreta, on podrà escollir quina sèrie o sèries de variables vol a la gràfica.

En l'exemple de la figura 7.22, es veu la gràfica de la quantitat total de palets produïda (blau) per una banda i alhora, dins del mateix quadre, el control de la quantitat total de palets produïda per cada torn. Aquesta eina permet analitzar desviacions de la producció per torn.

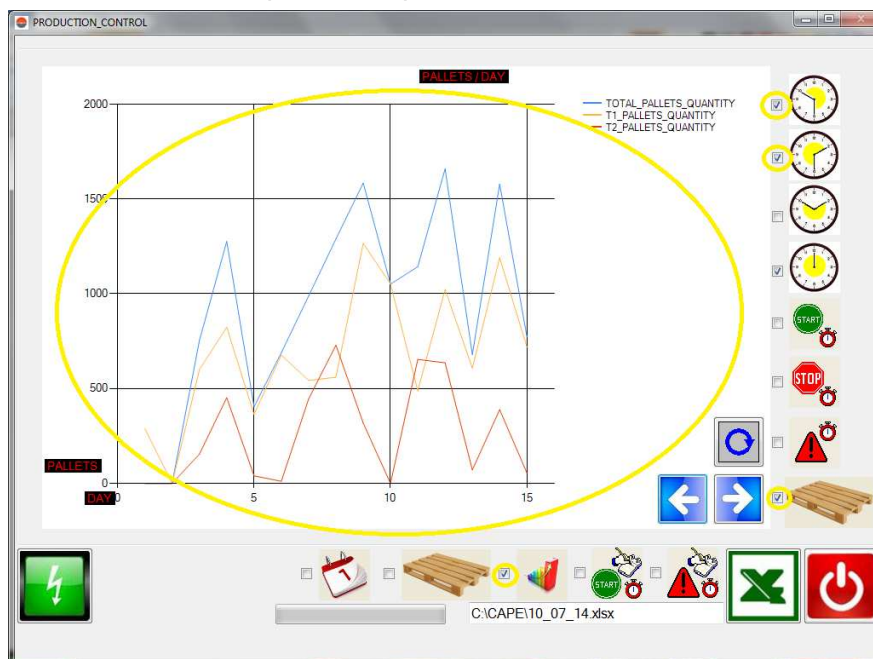


Figura 7.22. Visualització de les gràfiques de la quantitat de palets produïda

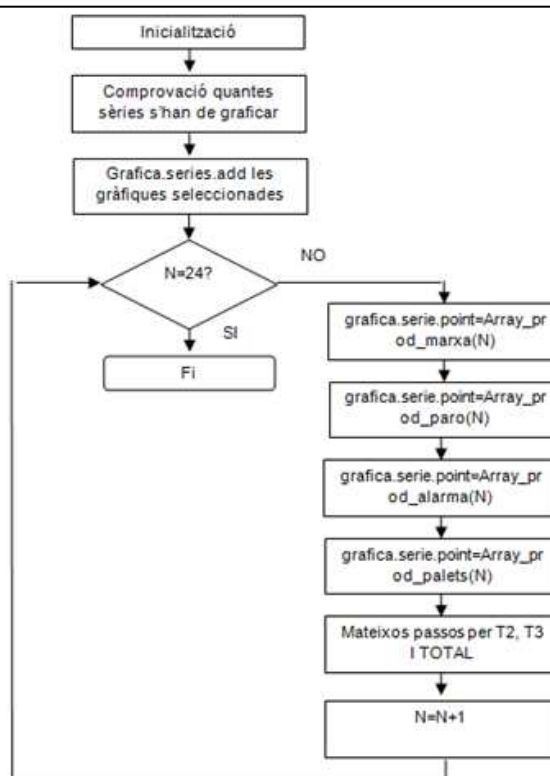


Figura 7.23 Diagrama de flux de la funció “Buto_Refrescar_Gràfica_Click”

7.3.6.- ENREGISTRAMENT DE LES ALARMES, I LA PRODUCCIÓ

Una opció que ens va semblar interessant de desenvolupar, era enregistrar la informació sobre quines alarmes s’havien produït durant el funcionament de la línia. De tal manera que aquesta informació quedés associada al dia que s’havia produït l’alarma o alarmes en qüestió i es pogués analitzar juntament amb la resta de dades del procés.

També es va voler dotar d’una altra funcionalitat al programa, l’adquisició de les dades productives del procés per un interval de temps determinat. En el nostre projecte es tractava d’enregistrar les dades cada hora. Això permet conèixer si durant un determinat període de la jornada la producció fluctua, per així poder aplicar les mesures adients per corregir-ho.

Per dur a terme aquestes opcions, en el codi del programari desenvolupat s’ha utilitzat un control de Visual Basic anomenat “timer” que genera una interrupció en un determinat interval de temps. De manera que l’aplicació del PC cada vegada que es produeix l’ interrupció fa una lectura de les alarmes actives o de les dades productives, i aquestes les emmagatzema en un arxiu Excel i

ahora les mostra per pantalla en una “Taula_Alarmes” tal com mostra la figura 7.24 o en una “Taula_Prod_Hores”

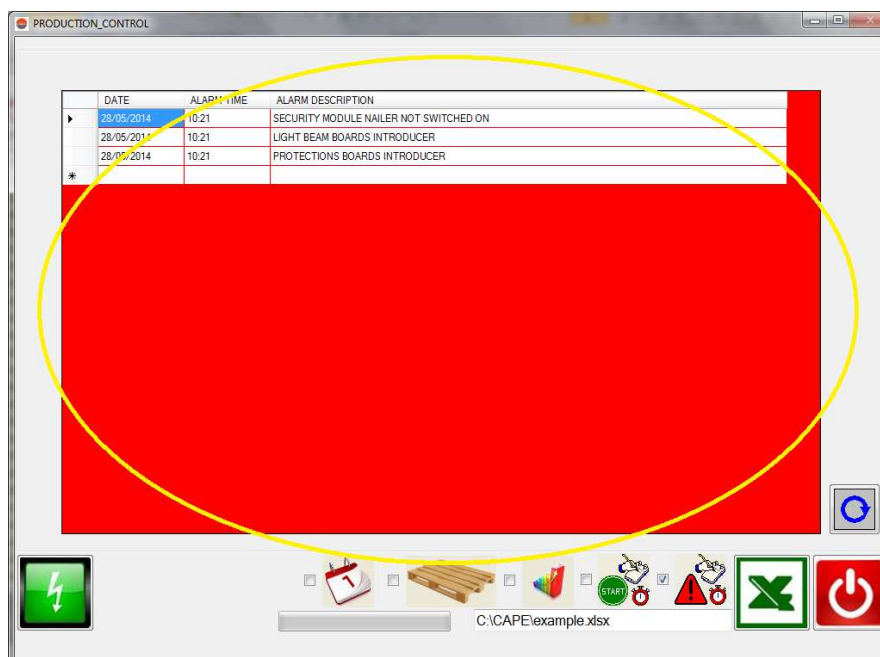


Figura 7.24. Monitorització de les alarmes actives

Per llegir les alarmes actives i emmagatzemar-les, s'executa la part de codi corresponent al diagrama de flux de la figura 7.25. En el cas de la lectura de les dades de producció, s'utilitza una altra funció molt semblant anomenada “Timer_Prod_Hores_Tick”.

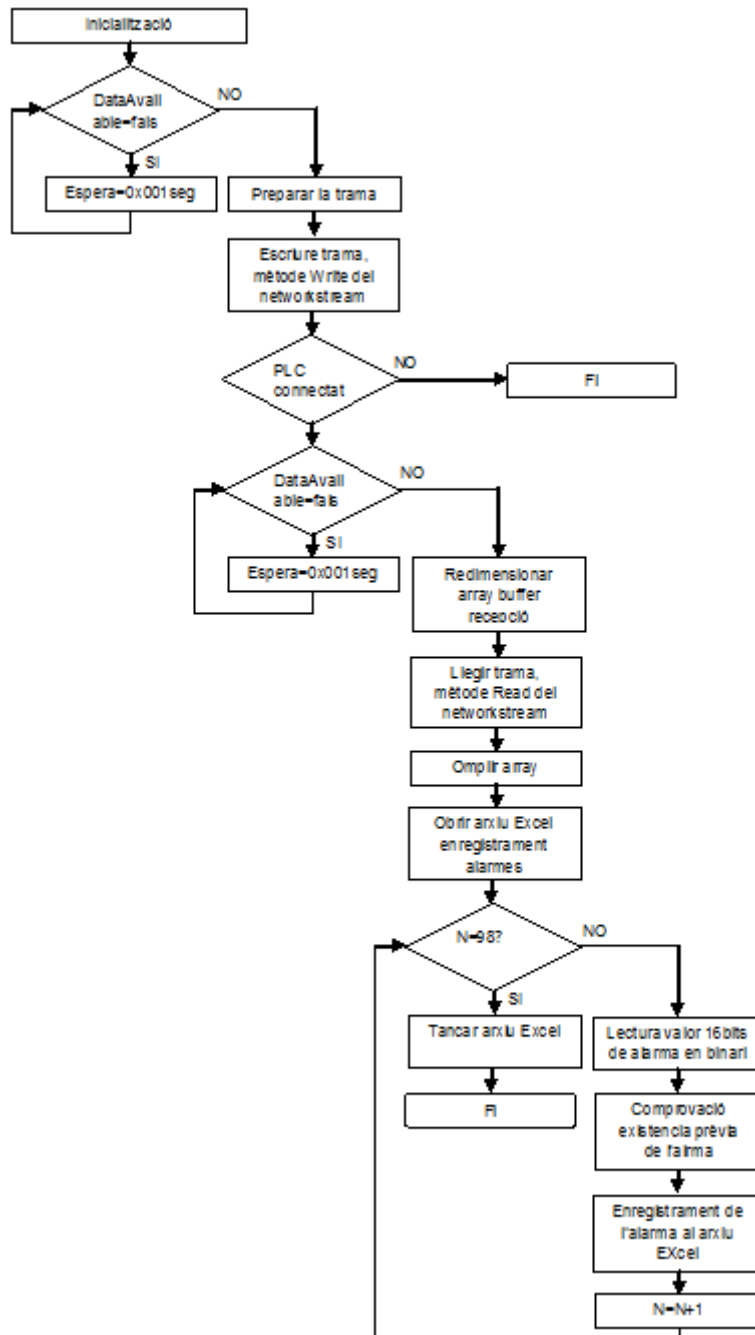


Figura 7.25 Diagrama de flux de la funció “Timer_Alarmes_Tick”

7.3.7.- EXPORTACIÓ DE LES DADES A UN ARXIU EXCEL

Finalment, l'aplicació permet a l'usuari exportar les dades llegides de la línia de clavat a un arxiu Excel. Aquest era un objectiu bàsic alhora de començar aquest projecte, ja que en l'actualitat és necessari poder extreure dades d'un procés productiu i adaptar-les a d'altres formats per ser utilitzades a posteriori.

El procediment per crear el nou arxiu Excel amb les dades i guardar-les en el PC de l'usuari, es fa en dos passos. Un primer, en el qual l'usuari ha d'introduir la ruta del directori on vol guardar l'arxiu, i prémer el botó amb el símbol del Microsoft Excel "Buto_Excel". Automàticament s'executa la funció "Buto_Excel_Click", que està dividida en dues subfuncions, la que crea l'arxiu (figura 7.26) i la que s'encarrega d'omplir les diferents cel·les de l'arxiu (figura 7.27).

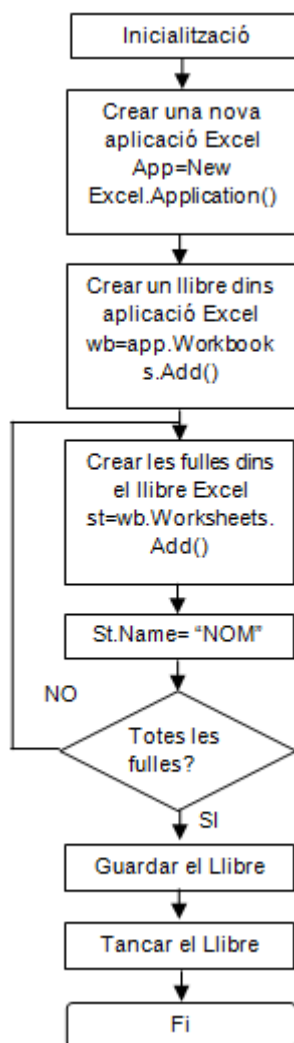


Figura 7.26 Diagrama de flux de la funció "Crear_Llibre_Excel"

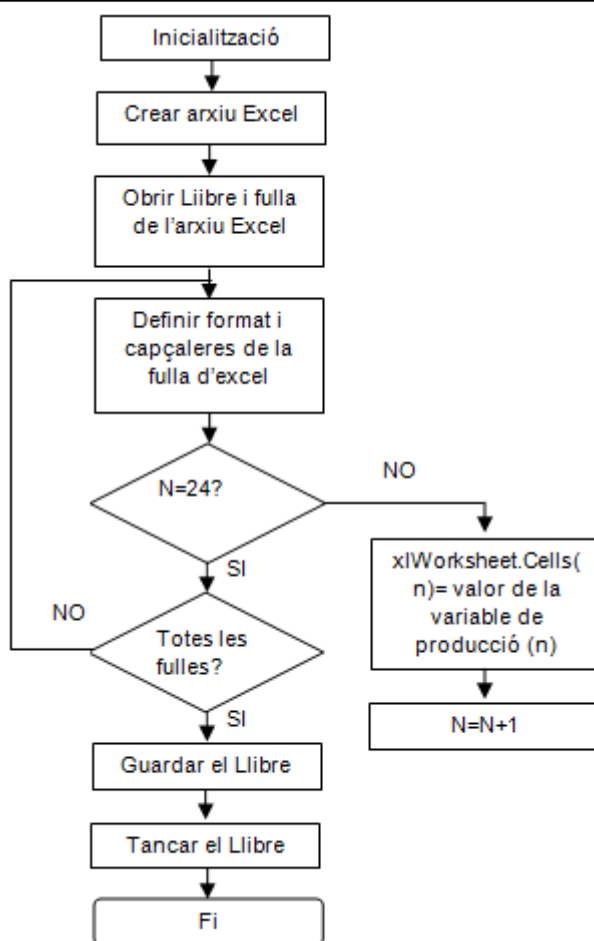


Figura 7.27 Diagrama de flux de la funció "Omplir_Arxiu_Excel"

8.- POSTA EN MARXA I DEPURACIÓ

Un cop finalitzada l'etapa de disseny i desenvolupament del programari, i finalitzat també el muntatge físic de les màquines, és l'hora de començar les primeres proves de funcionament de la línia i amb el sistema d'adquisició de dades integrat.

La depuració s'ha fet en diverses etapes, un primer nivell durant el disseny del programari, en el qual s'han fet successives compilacions i simulacions a mesura que es desenvolupaven els programes per tal de detectar possibles errors.

Aquestes verificacions del correcte funcionament del programari, s'han fet pas per pas per tal d'assegurar-ne el funcionament abans d'integrar una etapa superior. És a dir:

- 1.- Verificació funcionament programa PLC.
- 2.- Verificació funcionament programa terminal HMI.
- 3.- Verificació funcionament comunicació PLC → terminal HMI.
- 4.- Verificació funcionament programa Visual Basic 2008.
- 5.- Verificació funcionament comunicació (PLC → Terminal HMI) → Visual Basic.

8.1.- DEPURACIÓ DEL PROGRAMA PLC

Després de finalitzar un programa de PLC o una part del mateix, és convenient analitzar el projecte per tal d'assegurar-se que no hi ha error de configuració de hardware, de direccionament de variables o inclús errors tipogràfics. Tot això és possible fer-ho mitjançant les eines d'anàlisi i generació de que disposa el software Unity pro.

Un cop tenim el nostre programa correctament compilat, abans de descarregar-l'ho sobre un PLC real, el podem simular. El simulador és un programa annex al Unity pro que ens permet executar la nostra aplicació sobre aquest simulador com si es tractés d'un PLC real. En la següent figura és veu el panell del simulador executant-se.

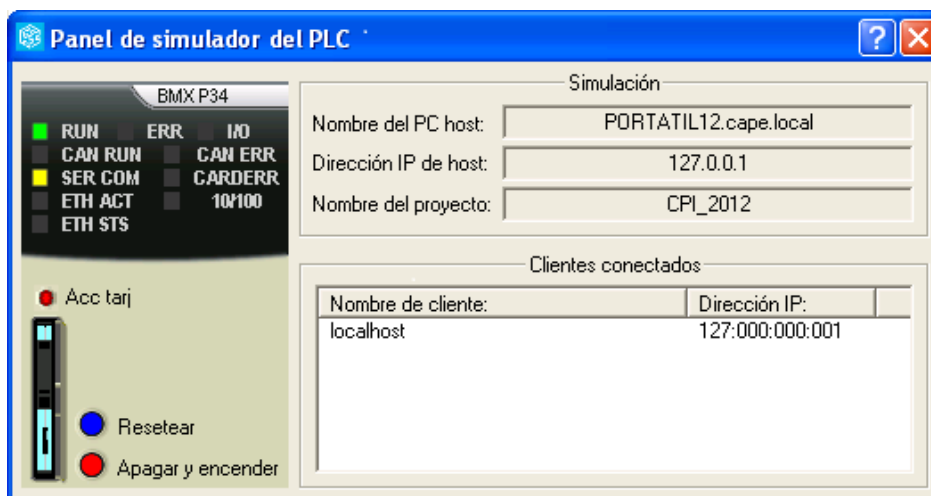


Figura 8.1. Panell del simulador del PLC

Amb el simulador en funcionament llavors es passa al control de la correcta execució del codi, això es pot fer mitjançant les taules d'animació per escriure i llegir els valors de les variables (figura 8.2) i de la simulació online de l'execució dels blocs de funció DFB (figura 8.3), graficet,ladder, etc del PLC.

Nombre	Valor	Tipo	Comentario
PROD_DIA		PROD_DIA	
PROD_CPA_DIA_QNT_PALE...	0	INT	
PROD_CPA_DIA_TEMPS_M...	0	INT	
PROD_CPA_DIA_TEMPS_PA...	0	INT	
PROD_CPA_DIA_TEMPS_AL...	14	INT	
PROD_CPA_DIA_MITJA_TO...	0.0	REAL	
PROD_CPA_DIA_DATA	1570	INT	
PROD_CPA_DIA_ANY	8212	INT	
PROD_CPA_DIA_QNT_PALE...	0	INT	
PROD_CPA_DIA_TEMPS_M...	0	INT	
PROD_CPA_DIA_TEMPS_PA...	0	INT	
PROD_CPA_DIA_TEMPS_AL...	0	INT	
PROD_CPA_DIA_MITJA_TO...	0.0	REAL	
PROD_CPA_DIA_QNT_PALE...	0	INT	
PROD_CPA_DIA_TEMPS_M...	0	INT	
PROD_CPA_DIA_TEMPS_PA...	0	INT	
PROD_CPA_DIA_TEMPS_AL...	0	INT	
PROD_CPA_DIA_MITJA_TO...	0.0	REAL	
PROD_CPA_DIA_QNT_PALE...	0	INT	
PROD_CPA_DIA_TEMPS_M...	0	INT	
PROD_CPA_DIA_TEMPS_PA...	0	INT	
PROD_CPA_DIA_TEMPS_AL...	0	INT	
PROD_CPA_DIA_MITJA_TO...	0.0	REAL	

Figura 8.2. Taula d'animació amb el valor animat de les variables internes del PLC

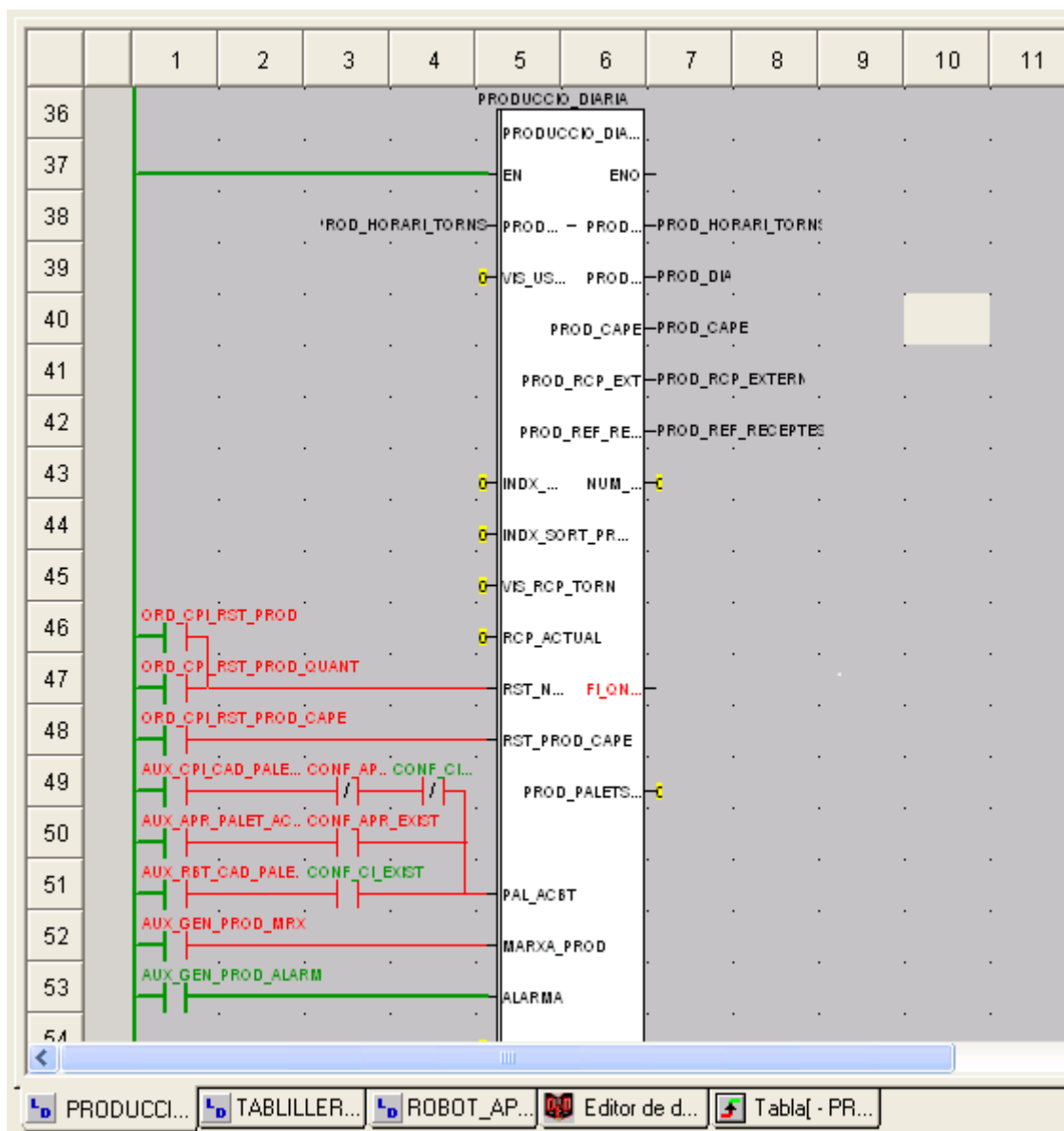


Figura 8.3. Execució online d'un bloc de funció DFB del PLC

8.2.- DEPURACIÓ DEL PROGRAMA HMI

El procediment per depurar el programa realitzat al terminal tàctil, és exactament el mateix que en el cas del PLC. Únicament canvien les eines per fer-ho, ja que són les específiques pel terminal tàctil.

Així doncs, un cop finalitzada l'aplicació, la compilarem i generarem per tal d'identificar possibles errors durant el disseny, edició i programació. Posteriorment,

simularem l'aplicació, de tal manera que el nostre ordinador emularà el funcionament al terminal.

A l'igual que en el PLC, també disposarem d'una taula animada per forçar valors i d'un simulador per recrear el comportament del terminal de manera online.

Variable	Dirección	Tipo	Valor	Simulaci
<input type="checkbox"/> PROD_CPA_DIA_MITJA...	%Mw738	REAL	0,00	
<input type="checkbox"/> PROD_CPA_DIA_DATA	%Mw740	INT	0	
<input type="checkbox"/> PROD_CPA_DIA_ANY	%Mw741	INT	0	
<input type="checkbox"/> PROD_CPA_DIA_QNT_P...	%Mw742	INT	100	
<input type="checkbox"/> PROD_CPA_DIA_TEMPS...	%Mw743	INT	0	
<input type="checkbox"/> PROD_CPA_DIA_TEMPS...	%Mw744	INT	0	
<input type="checkbox"/> PROD_CPA_DIA_TEMPS...	%Mw745	INT	0	
<input checked="" type="checkbox"/> PROD_CPA_DIA_MITJA...	%Mw746	REAL	4,50	
<input checked="" type="checkbox"/> PROD_CPA_DIA_QNT_P...	%Mw748	INT	10	
<input type="checkbox"/> PROD_CPA_DIA_TEMPS...	%Mw749	INT	0	
<input type="checkbox"/> PROD_CPA_DIA_TEMPS...	%Mw750	INT	0	
<input type="checkbox"/> PROD_CPA_DIA_TEMPS...	%Mw751	INT	0	
<input type="checkbox"/> PROD_CPA_DIA_MITJA...	%Mw752	REAL	0,00	
<input type="checkbox"/> PROD_CPA_DIA_QNT_P...	%Mw754	INT	0	
<input type="checkbox"/> PROD_CPA_DIA_TEMPS...	%Mw755	INT	0	
<input type="checkbox"/> PROD_CPA_DIA_TEMPS...	%Mw756	INT	0	
<input type="checkbox"/> PROD_CPA_DIA_TEMPS...	%Mw757	INT	0	
<input type="checkbox"/> PROD_CPA_DIA_MITJA...	%Mw758	REAL	0,00	

Figura 8.4. Taula d'animació amb el valor de les variables internes del terminal tàctil



Figura 8.5. Emulació del terminal tàctil mitjançant el simulador

8.3.- DEPURACIÓ DE LA COMUNICACIÓ PLC→TERMINAL TÀCTIL

Per lligar els programes que prèviament havíem verificat individualment, la única cosa que hem hagut de fer és configurar la mateixa adreça IP del simulador del PLC en el controlador Modbus TCP/IP del terminal tàctil (Figura 8.6). D'aquesta manera els dos softwares queden enllaçats internament, permetent interactuar desde el teu ordinador com si estiguéssis operant amb els dispositius reals.

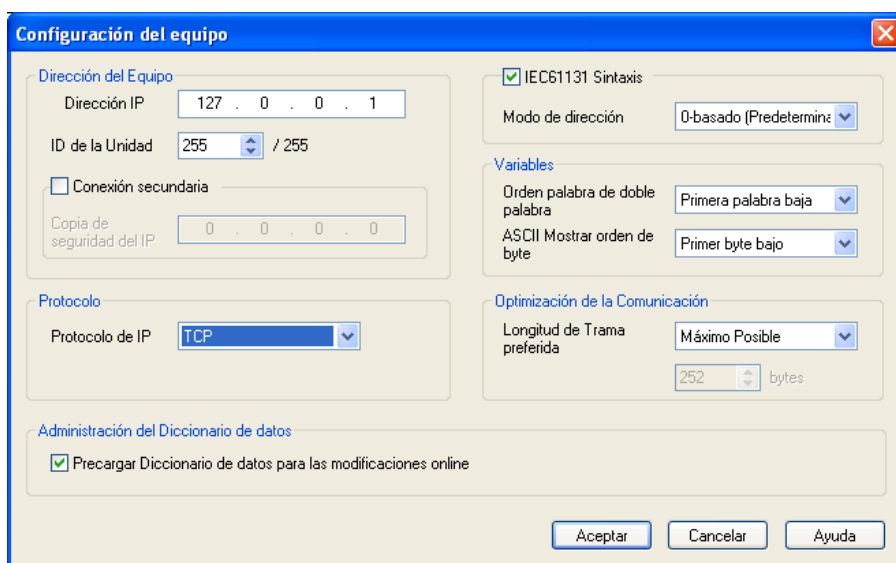


Figura 8.6. Controlador terminal HMI amb la direcció del simulador del PLC

8.4.- DEPURACIÓ DEL PROGRAMA EN VISUAL BASIC 2008

Els passos per depurar el programa en visual Basic han sigut similars als anteriors, és a dir, a mesura que es progressava en el disseny de la aplicació és prosseguia a compilar i depurar el formulari per tal d'identificar possibles errors i verificar-ne el correcte funcionament.

Una eina molt útil per aquesta tasca, és l'opció d'adherir punts d'interrupció dins el codi del Visual Basic 2008, de manera que quant s'executa la instrucció marcada amb un punt d'interrupció s'atura el programa facilitant el control de les variables en aquell punt, tal com mostra la següent figura.

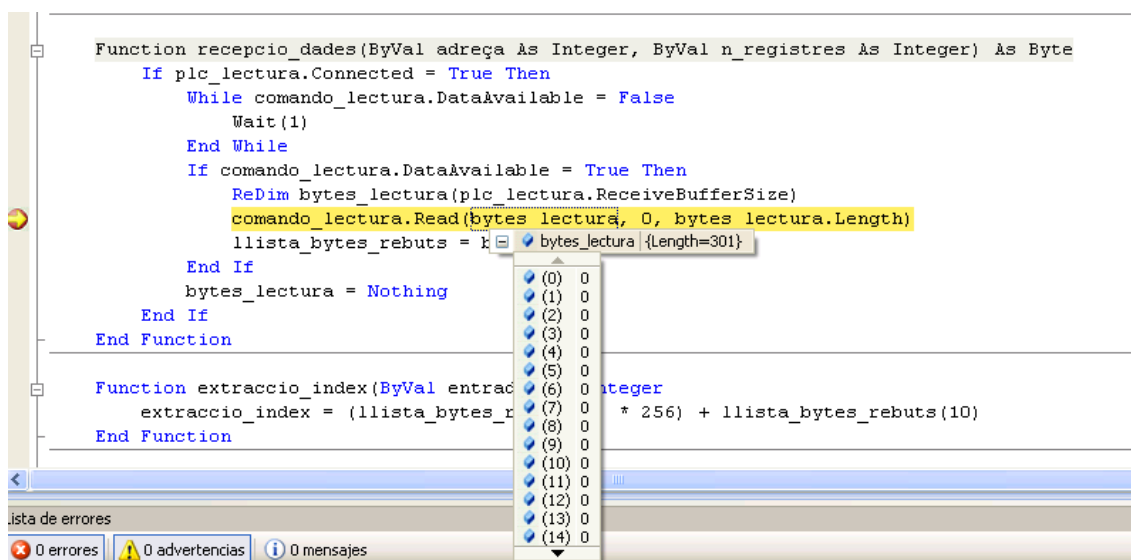


Figura 8.7. Interrupció del codi del programa Visual Basic per a la seva depuració

8.5.- DEPURACIÓ DE LA COMUNICACIÓ (PLC→TERMINAL TÀCTIL)→ VISUAL BASIC 2008

Finalment, per enllaçar el programa del Visual Basic amb els programes que ja hem desenvolupat (PLC,HMI) i així poder assegurar el correcte funcionament sense necessitat de cap hardware, s'ha de canviar l'adreça IP de la connexió al host remot. En el nostre cas, posarem la mateixa adreça del simulador del PLC, la qual també és la mateixa del controlador de la pantalla com hem explicat anteriorment.

```
Try
    'estableix una connexió a un host remot. El host s'especifica ,
    'cadena que conté un direcció IP en notació de quatre números
    plc_lectura.Connect(IPAddress.Parse("127.0.0.1"), 502)

Catch ex As Exception
    BUTO_CONECTAR.Image = My.Resources.desconectat
    BUTO_LLEGIR.Visible = False

End Try
```

Figura 8.8. Fragment de codi on es canvia l'adreça IP, per poder treballar amb mode simulació

8.6.- POSADA EN FUNCIONAMENT DEL PROGRAMARI FINAL

Aquest apartat consta de la posta en marxa a casa del client, en la qual es torna a verificar que el programari funcioni de manera correcte. S'ajusten els últims detalls, es fan modificacions puntuals a petició del client sempre que això sigui possible. Finalment es comprova que el programari compleixi amb les especificacions bàsiques que s'havien marcat.

Un cop es va instal·lar el programari en l'ordinador del client i es va posar en marxa la línia, es va verificar el funcionament correcte de les comunicacions Modbus TCP/IP mitjançant el software wireshark. Mitjançant el seu mode de captura de trames es va analitzar que les trames de lectura i escriptura de registres es transmetessin i rebessin en la seva totalitat i no hi hagués cap error.

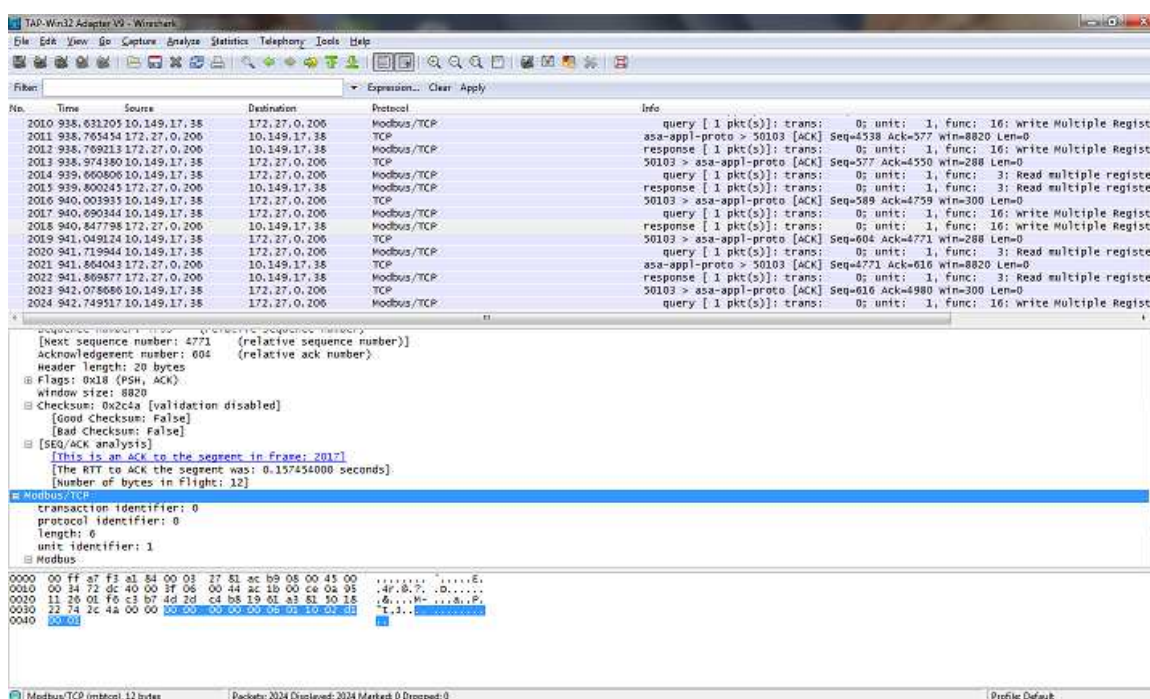


Figura 8.9. Exemple trama resposta a una petició d'escriure múltiples registres.

La visualització, in situ, de la línia automàtica de palets i el programari d'adquisició de dades, treballant correctament i mostrant les dades productives alhora que les emmagatzema, és el resultat real d'aquest projecte. Malauradament, això no es possible visualitzar-ho en la memòria escrita, només mencionar que es van assolir tots els objectius i que la línia i el programari funciona sense cap problema.

9.- CONCLUSIONS

En aquest capítol, un cop exposada tota la informació sobre el treball realitzat en aquest projecte, és el moment d'ordenar les idees i extreure'n conclusions. Presentant els aspectes positius, i resumint els conceptes més destacats del projecte.

En primer lloc, i com el fet més destacat, és que l'objectiu del projecte s'ha aconseguit d'una forma molt satisfactòria: s'ha dissenyat el programari per a l'adquisició de dades de la producció d'una línia automàtica de palets, el qual funciona de manera autònoma, correcte i complint les especificacions desitjades.

Un aspecte important i positiu d'aquest projecte ha sigut la utilització del protocol Motius/TCP en el programari creat. S'ha evidenciat que es tracta d'un protocol flexible i de fàcil manipulació, fet que denota el perquè de la seva gran difusió en entorns industrials.

La utilització del protocol Motius/TCP ens brinda la possibilitat de crear interfícies de supervisió i control, sobre plataformes de desenvolupament comunes. Únicament necessitem que la plataforma triada tingui llibreries per el protocol TCP/IP. A més, si aquesta plataforma té accés a Internet, les dades poden ser accessibles desde qualsevol lloc o dispositiu, obrint les portes a un gran ventall de possibles aplicacions.

En el camp de la programació, s'ha d'esmentar el redisseny, estructuració, ampliació i optimització de recursos, que ha sofert el programa del PLC, així com el terminal Tàctil.

Finalment, destacar que l'entorn de programació del Visual Basic 2008 és molt potent, el qual ens dóna la possibilitat de crear desde aplicacions simples fins a complexes i totes elles es poden comunicar fàcilment amb el dispositius de control industrial, mitjançant les seves llibreries TCP/IP. Aquesta capacitat, fa que sigui una bona alternativa per fer de nexa d'unió entre el procés i l'àrea de gestió de l'empresa.

Per l'empresa Eglinton Timber Products LTD, el resultat d'aquest projecte els facilita una eina per l'adquisició de les dades productives del seu procés. L'enregistrament d'aquestes dades permetrà poder fer-ne un seguiment i analitzar-les amb posterioritat

Per l'empresa Mecànica Cape S.L, la finalització satisfactòria d'aquest projecte els hi aporta una millora qualitativa i competitiva d'un producte, i permet ampliar les solucions comercials per a les necessitats dels seus client.

10.- FUTURS PROJECTES

El treball realitzat ha complert totes les expectatives creades, però tot i l'obtenció dels objectius d'aquest projecte es deixa oberta la porta per a futures millores, ampliacions i/o modificacions. En el moment de finalitzar aquest projecte, ja es treballava en l'estudi de possibles ampliacions.

Una millora seria la incorporació de més funcions i/o serveis en l'aplicació original, això acostaria l'aplicació en quant a prestacions als softwares comercials existents en la actualitat, però conservant la idea de simplicitat i dimensió de la idea original. Aquestes posteriors versions inclourien:

-Mòdul de Monitorització: Representació de les dades en temps real dels sensors, actuadors, etc.

-Mòdul de Control: Possibilitat de canviar consignes, dades, ordres de marxa, etc. Directament desde el ordenador.

-Mòdul de Supervisió: Incloure més eines de gestió per a la presa de decisions, com la generació d'informes d'incidències per tal de poder confeccionar plans de manteniment preventiu, etc.

-Mòdul d'emmagatzematge de dades: Utilitzar l'estàndard SQL per facilitar la comunicació amb bases de dades.

-Mòdul dinàmic de canvi d'idiomes: Ja que en l'actualitat el programari únicament disposa de l'anglès com a text escrit.

Una altra ampliació que s'està estudiant és la creació d'una aplicació per a dispositius mòbils, smartphones, tablets. Es mantindria la mateixa base, és a dir el programa de control i emmagatzematge de les dades productives estaria en la memòria del PLC, però enlloc de utilitzar un client Modbus TCP/IP que seria un PC aquest seria una tablet que obtindria les dades de la xarxa industrial mitjançant una connexió wifi.



Figura 10.1. Esquema de la integració d'un dispositiu mòbil a la xarxa industrial

Actualment existeixen diferents softwares que permet enllaçar i intercanviar dades entre un dispositiu mòbil i un dispositiu que utilitzi el protocol Modbus TCP/IP. El programari TeslaModbusSCADA per a android, és una opció molt interessant. Permet comunicar amb qualsevol dispositiu sobre el protocol Modbus TCP i inclou un complet editor per crear la pròpia aplicació del usuari, d'aquesta manera es podria fer un clon a petita escala del programa original d'aquest projecte per a dispositiu mòbil.



Figura 10.2. Exemple de la connectivitat d'una aplicació utilitzant TeslaModbusSCADA

11.- BIBLIOGRAFIA

Modicom M340 con Unity Pro

Manual configuracion

Schneider

Edició 04/2009

Unity Pro

Manual referència

Schneider

Edició 04/2009

XBT GT

Manual d'usuari

Schneider

Edició 2.0

Visual Basic 2008 in simple steps

Dreatech press

Edició 2009

Manejo de datos con Visual Basic y ASP.NET

Carlos Alberto Vanejas

Universidad Distrital Francisco José de Caldas

Edició 08/2010

Introduction to Modubs TCP/IP

Technical reference-Modbus TCP/IP

Acromag incorporated

Edició 2005

INFOGRAFIA

<http://social.msdn.microsoft.com/Forums>

Primavera 2014

<http://msdn.microsoft.com>

Primavera 2014

<http://www.elquille.info>

Primavera 2014

<http://www.visual-basic-tutorials.com/index.htm>

Primavera 2014

<http://www.control.com/>

Primavera 2014

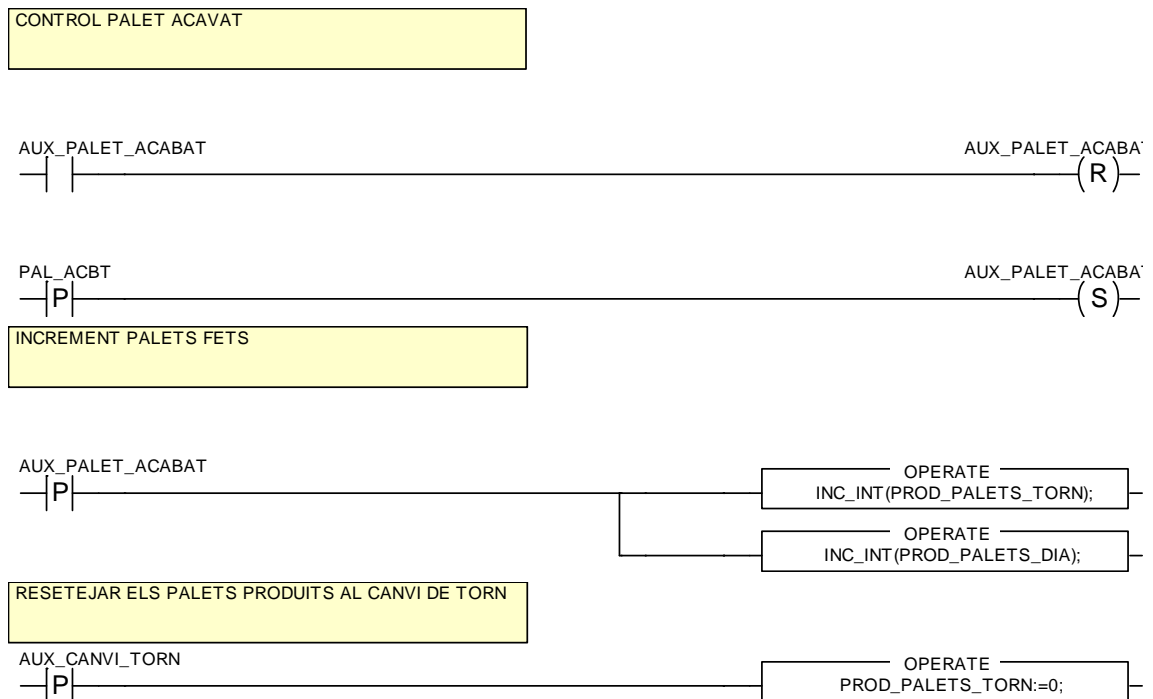
1.- INTRODUCCIÓ

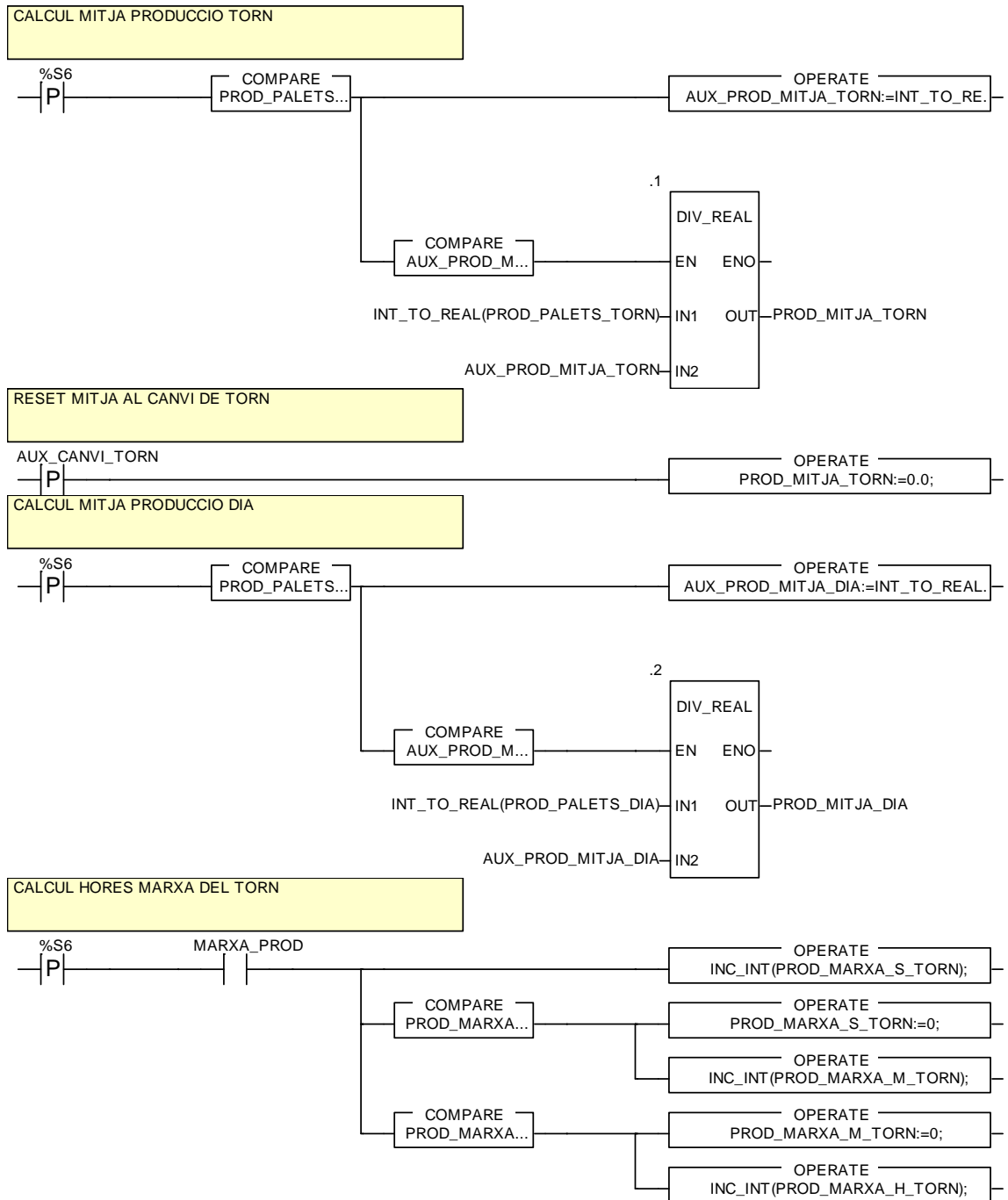
En aquest annex de la memòria es troben els diferents codis implementats en aquest projecte, ens hem abstingut d'adjuntar tot el programa desenvolupat en ladder, ja que ocuparia moltes pàgines i el seu seguiment seria complex, això sí hem inclòs una part representativa. La que correspon al DFB de producció diària. De totes maneres, el programa complet es troba en el compact disc adjunt a la memòria.

També hem adjuntat els codi desenvolupats en llenguatge Visual Basic de les rutines amb diagrama que s'han descrit en el punt 7 de la memòria.

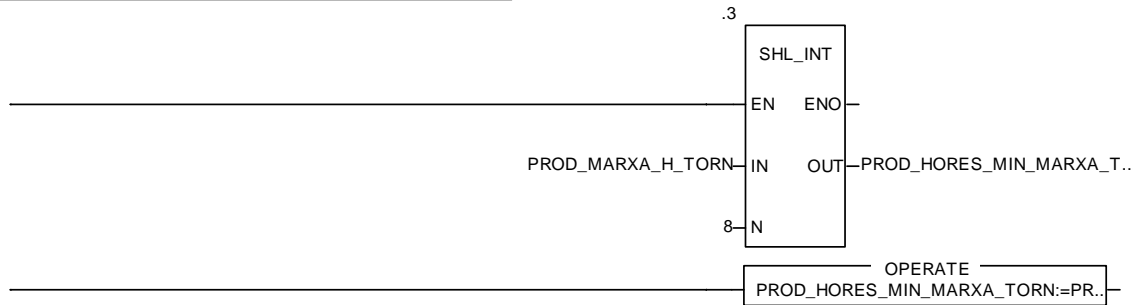
2.- PROGRAMA EN LADDER DEL PLC

2.1.- DFB PRODUCCIÓ DIÀRIA.

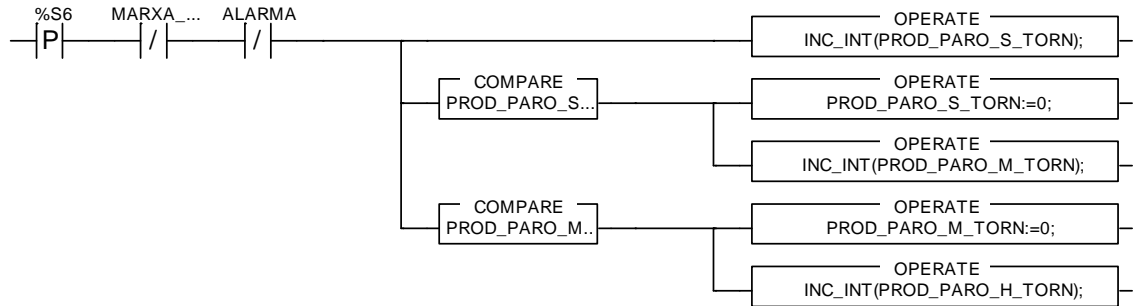




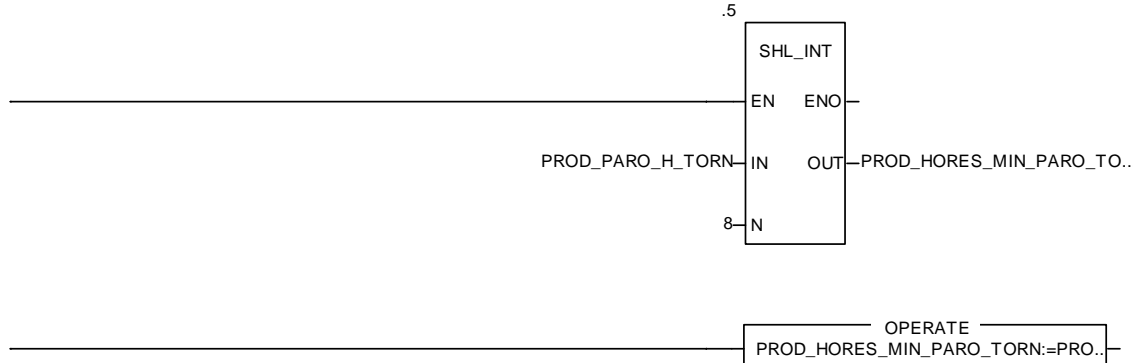
AGRUPAR EN UN INTEGER HORES I MINUTS DE MARXA



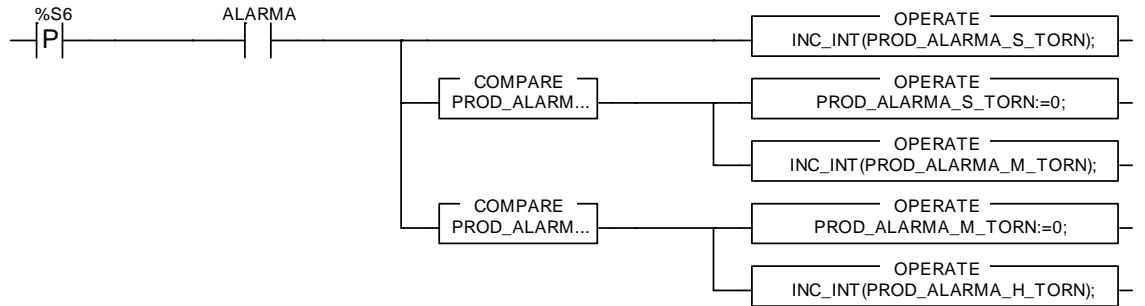
CALCUL HORES PARO DEL TORN



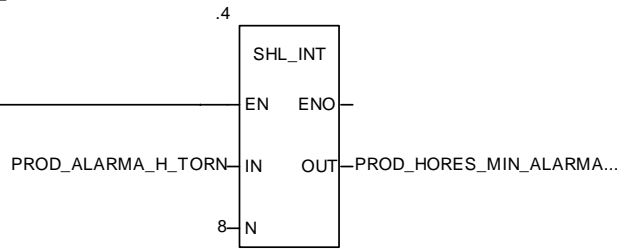
AGRUPAR EN UN INTEGER HORES I MINUTS DE PARO



CALCUL HORES INCIDENCIA DEL TORN



AGRUPAR EN UN INTEGER HORES I MINUTS DE INCIDÈNCIA



RESET VALORS TEMPS AL CANVI DE TORN



OPERATE
PROD_HORES_MIN_ALARMA_TORN:=P...

OPERATE
PROD_MARXA_S_TORN:=0;

OPERATE
PROD_MARXA_M_TORN:=0;

OPERATE
PROD_MARXA_H_TORN:=0;

OPERATE
PROD_PARO_S_TORN:=0;

OPERATE
PROD_PARO_M_TORN:=0;

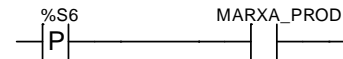
OPERATE
PROD_PARO_H_TORN:=0;

OPERATE
PROD_ALARMA_S_TORN:=0;

OPERATE
PROD_ALARMA_M_TORN:=0;

OPERATE
PROD_ALARMA_H_TORN:=0;

CALCUL HORES MARXA DEL DIA



OPERATE
INC_INT(PROD_MARXA_S_DIA);

COMPARE
PROD_MARXA...

OPERATE
PROD_MARXA_S_DIA:=0;

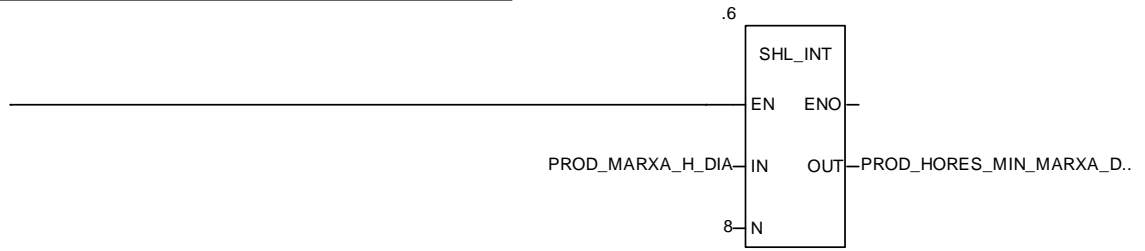
OPERATE
INC_INT(PROD_MARXA_M_DIA);

COMPARE
PROD_MARXA...

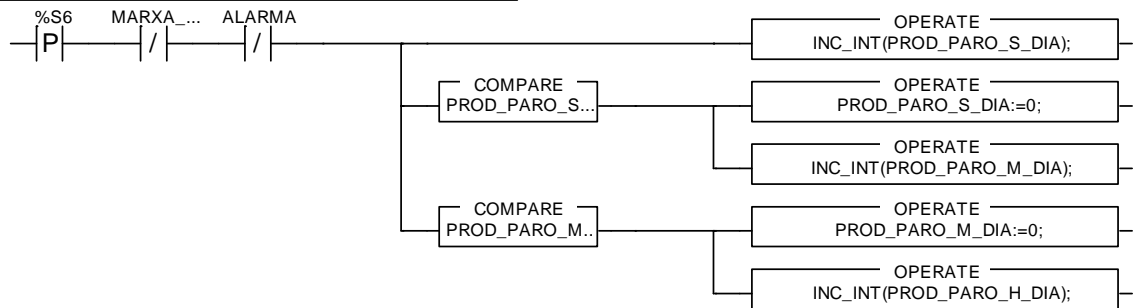
OPERATE
PROD_MARXA_M_DIA:=0;

OPERATE
INC_INT(PROD_MARXA_H_DIA);

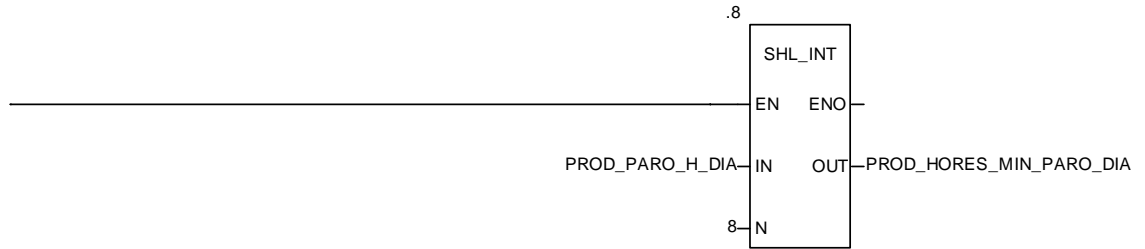
AGRUPAR EN UN INTEGER HORES I MINUTS DE MARXA



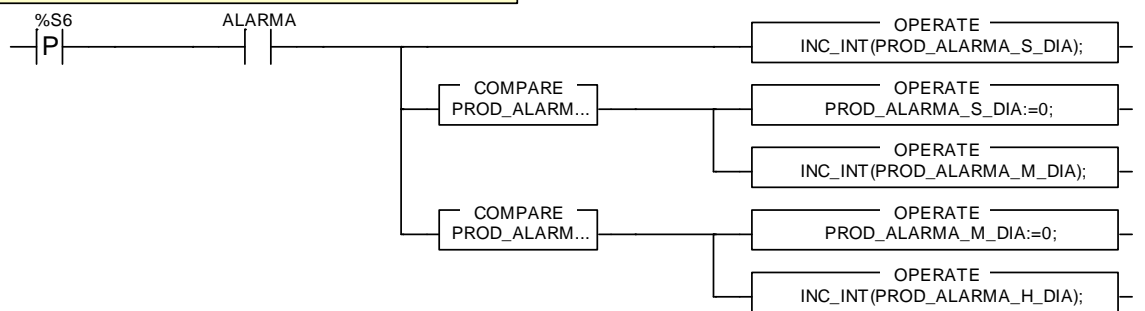
CALCUL HORES PARO DEL DIA



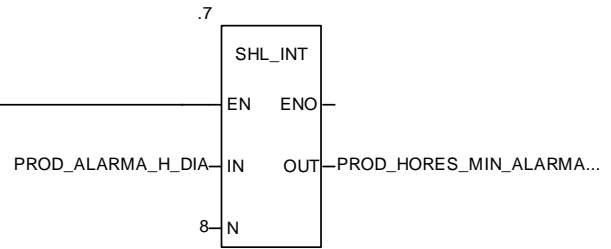
AGRUPAR EN UN INTEGER HORES I MINUTS DE PARO



CALCUL HORES INCIDENCIA DEL DIA



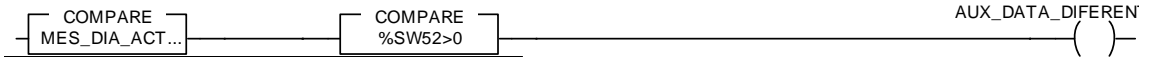
AGRUPAR EN UN INTEGER HORES I MINUTS DE INCIDÈNCIA



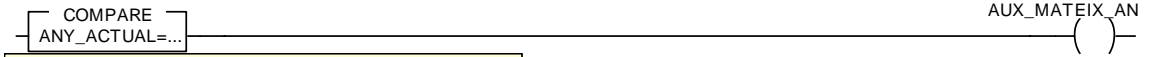
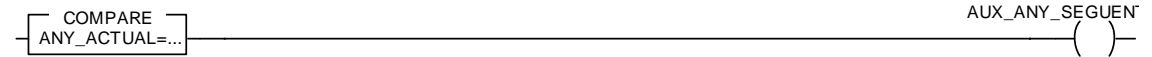
OPERATE
PROD_HORES_MIN_ALARMA_DIA:=PR...

GESTIO ARRAYS PRODUCCIO DIA

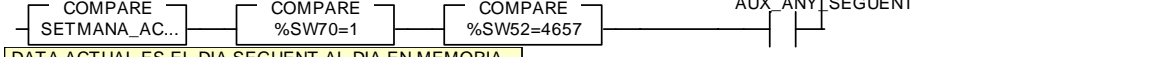
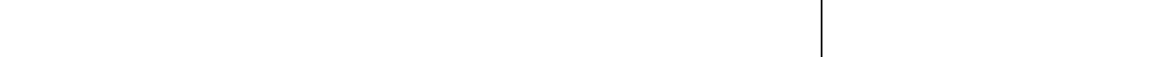
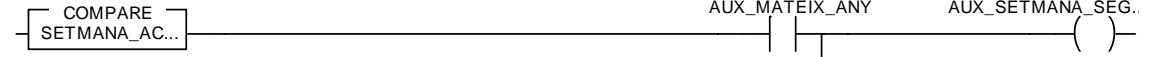
DETECCIO CANVI DE DIA



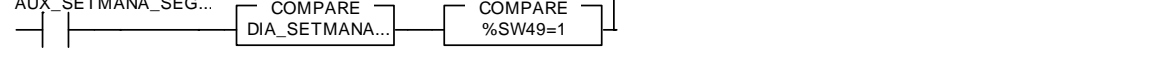
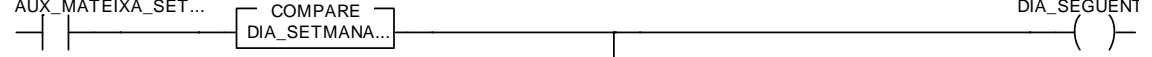
ANY ACTUAL IGUAL O SEGUENT A L'ANY EN MEMORIA

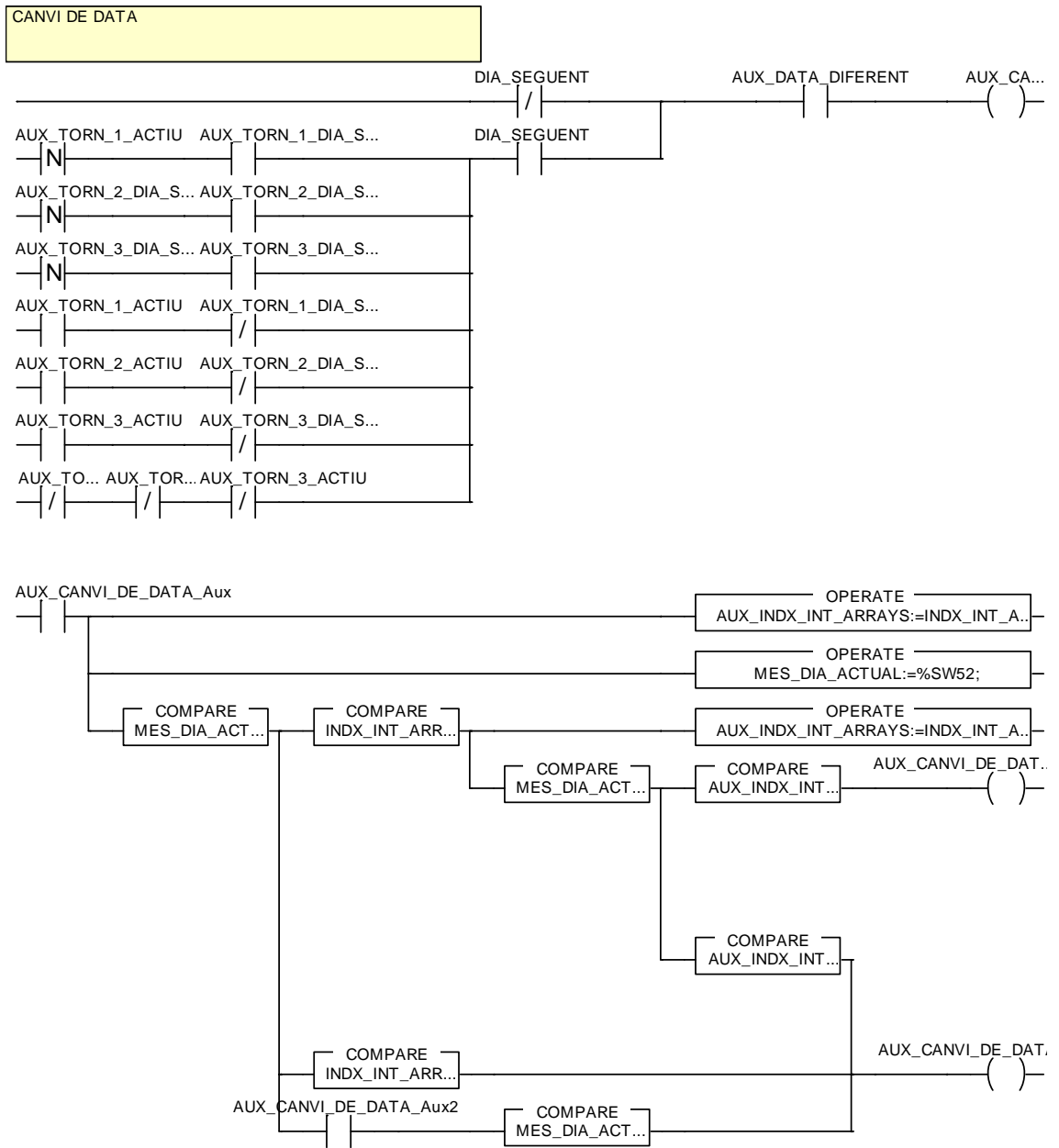


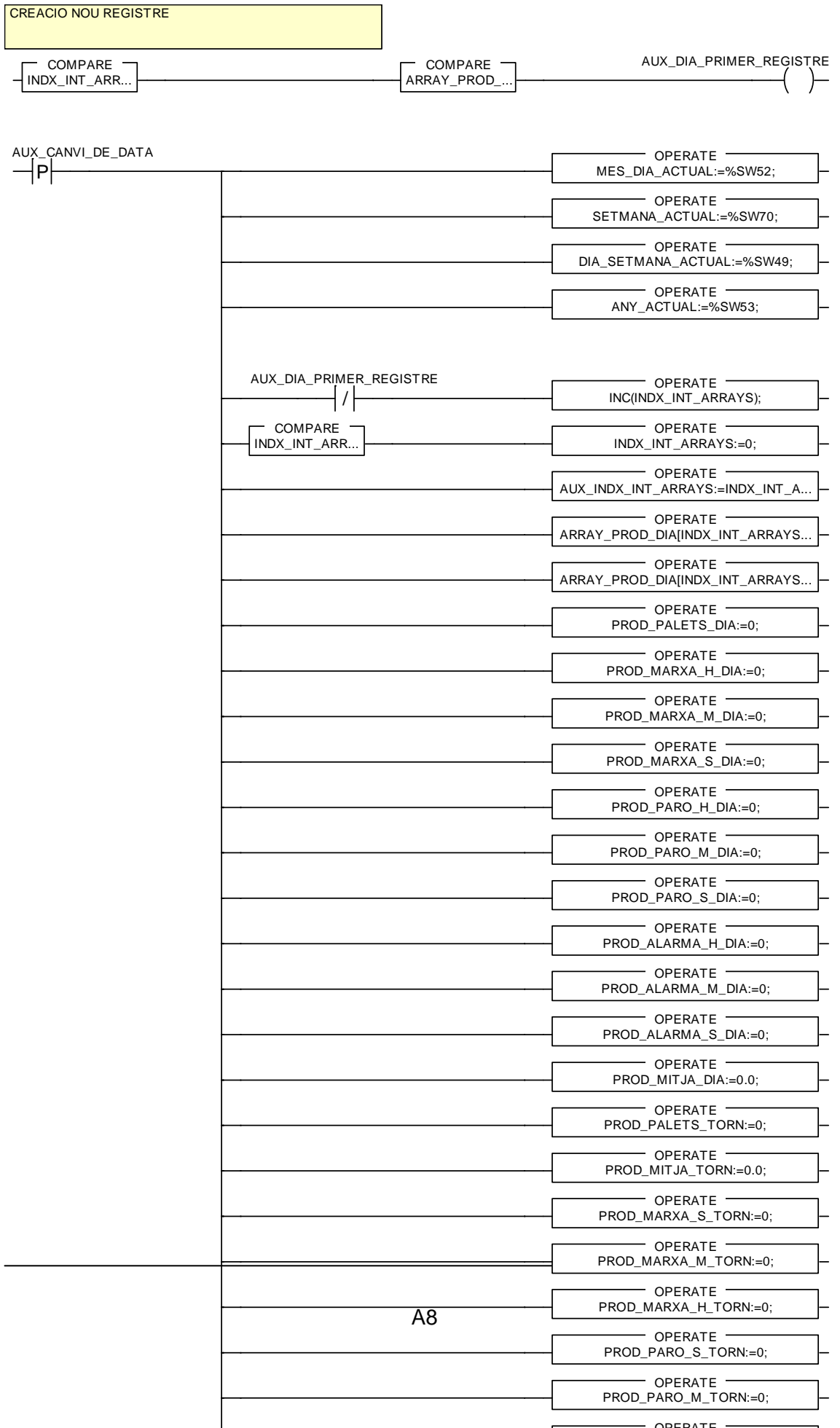
SETMANA ACTUAL IGUAL O SEGUENT A LA SETMANA EN



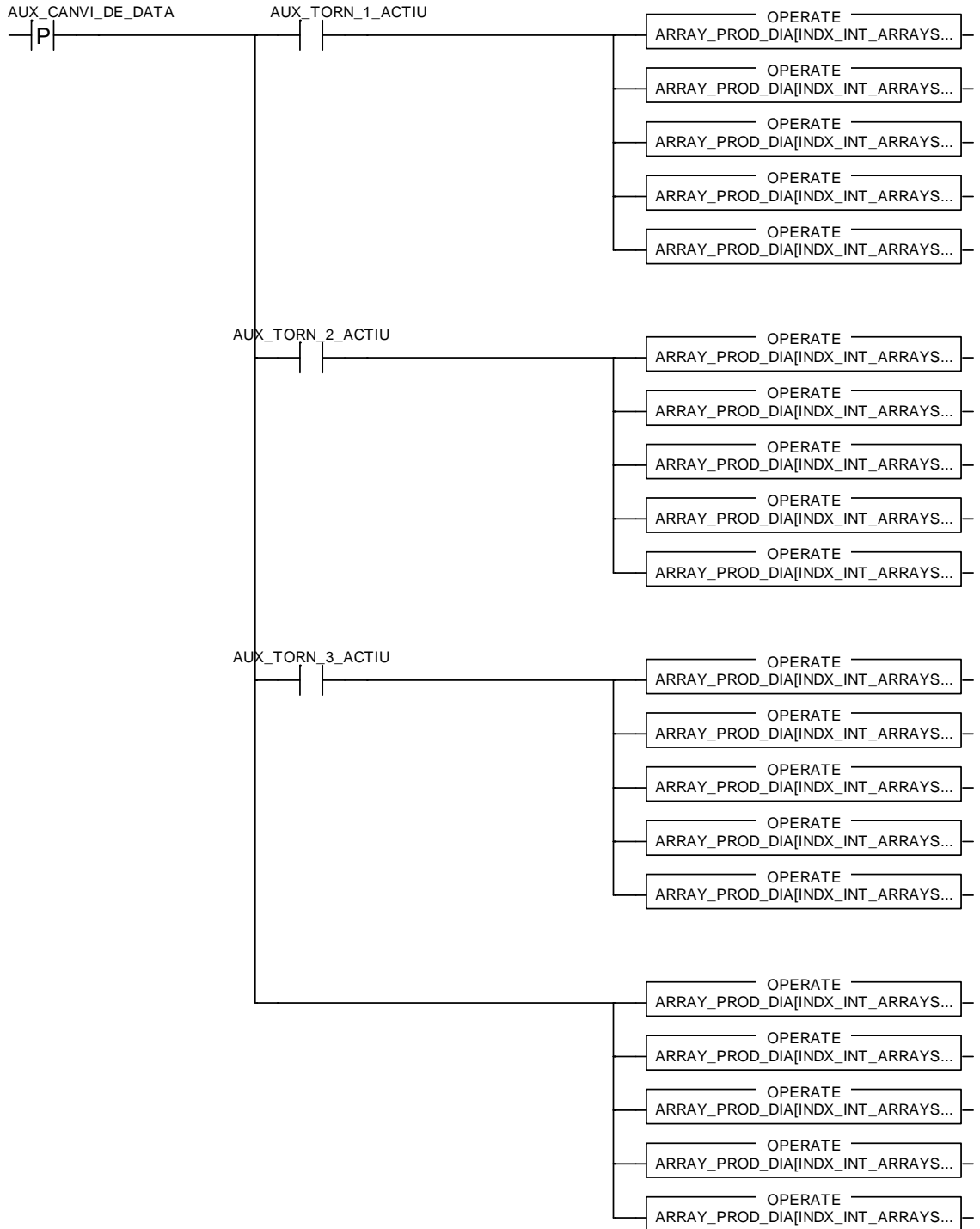
DATA ACTUAL ES EL DIA SEGUENT AL DIA EN MEMORIA





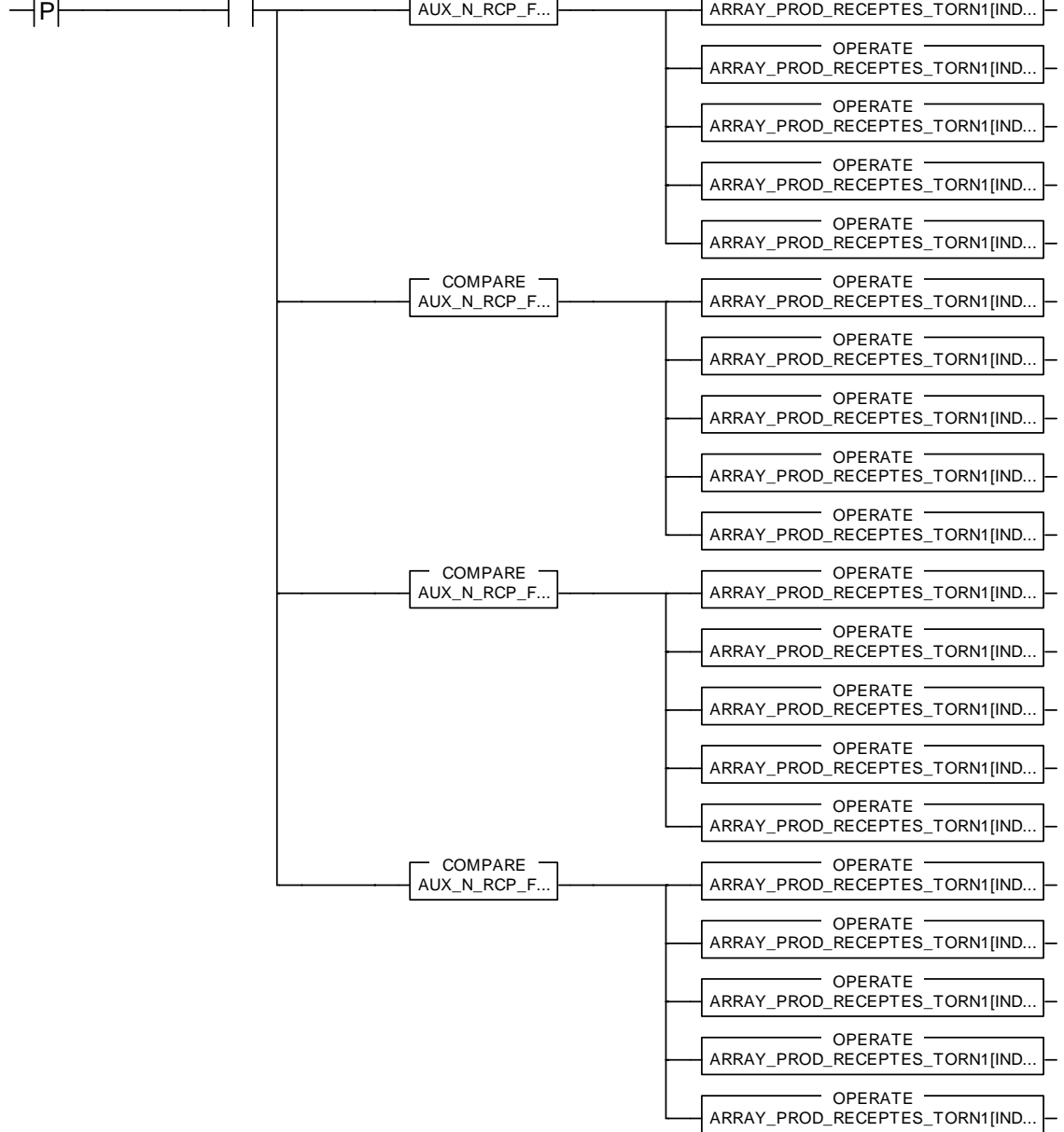


NETEJAR DATOS REGISTRE NOU QUE COMENCEM A
UTILITZAR PRODUCCIO DIA



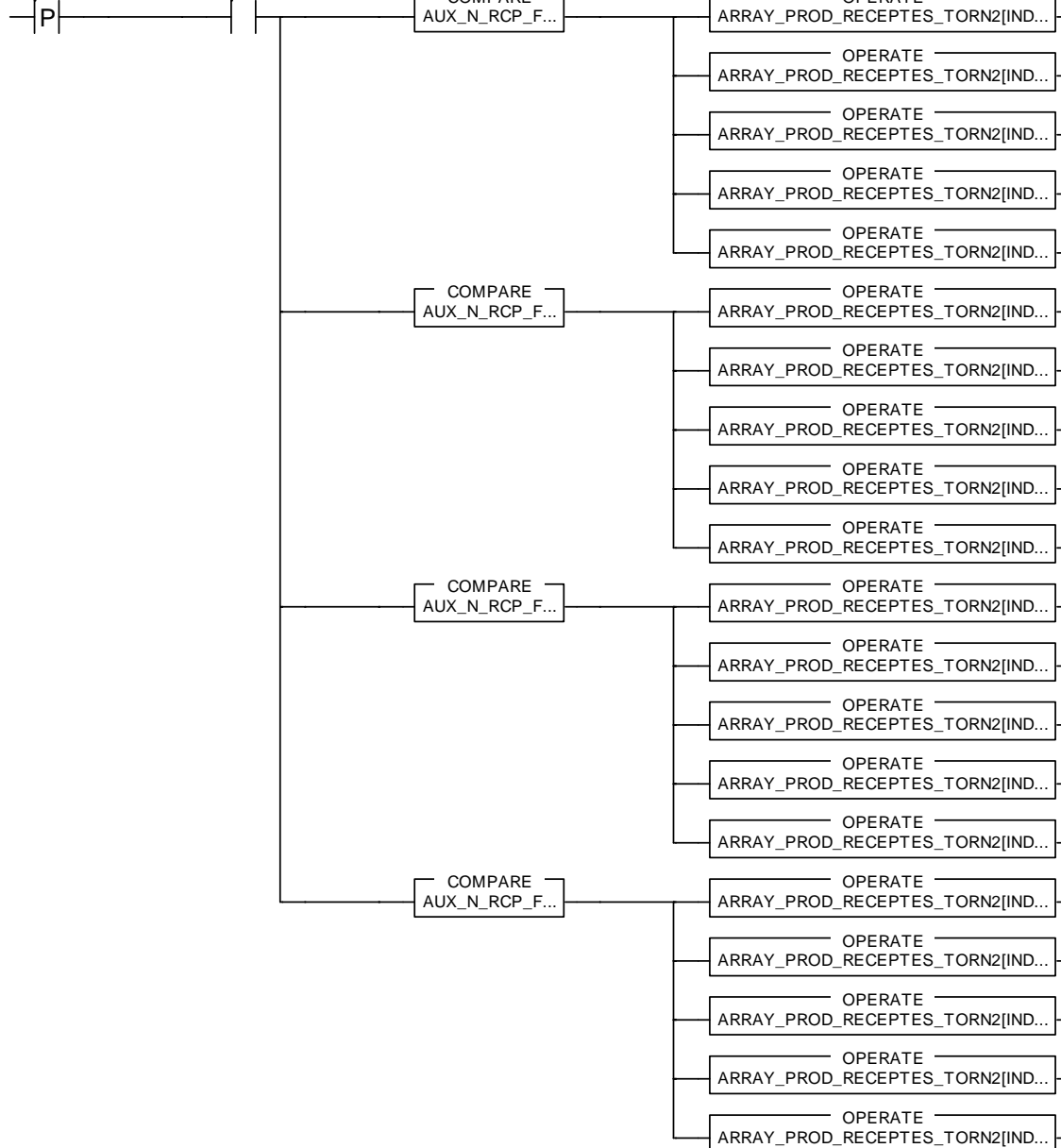
NETEJA REGISTRE QUE COMENCEM A UTILITZAR TORN 1
PER RECEPTES

AUX_CANVI_DE_DATA AUX_TORN_1_ACTIU



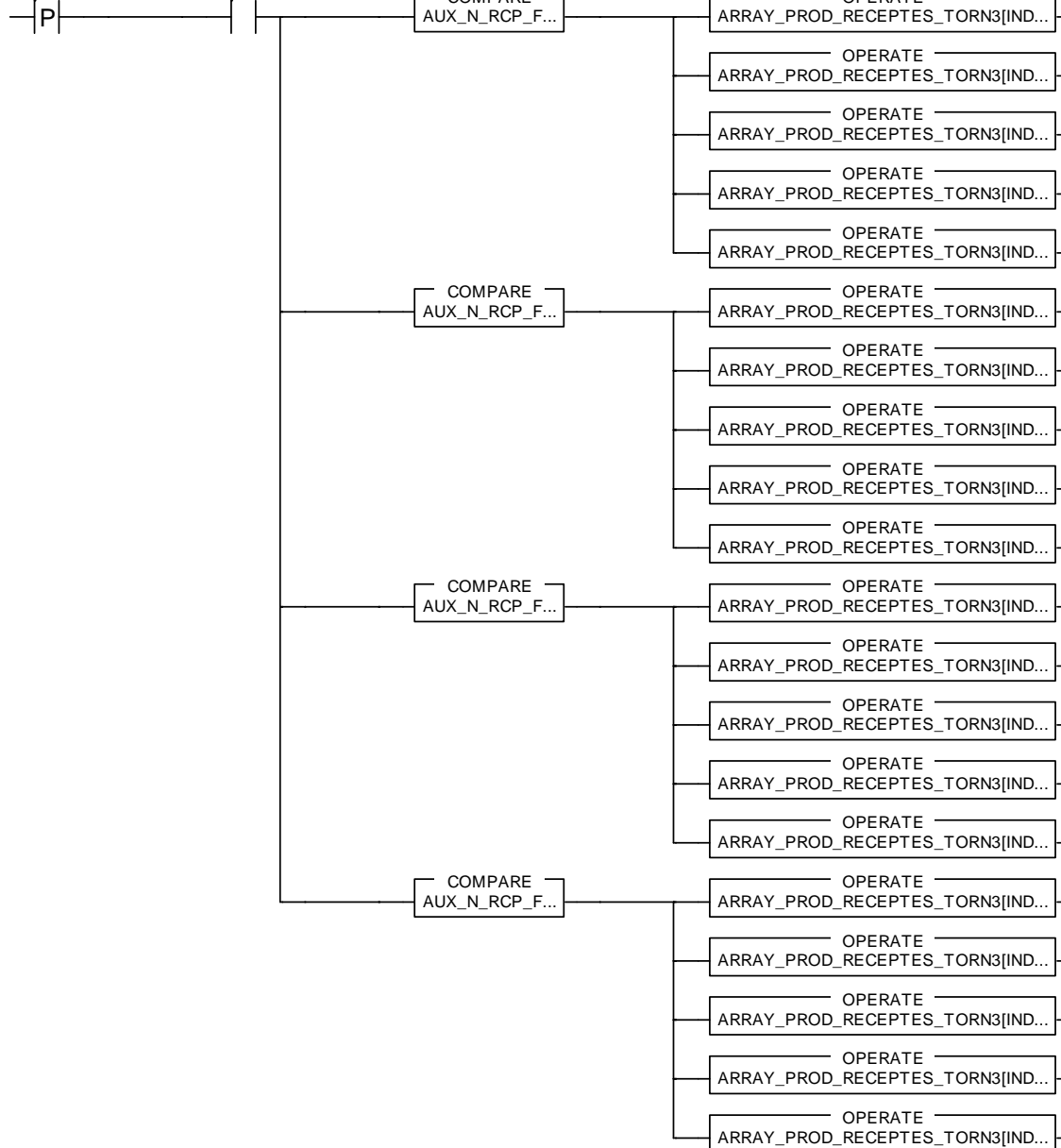
NETEJA REGISTRE QUE COMENCEM A UTILITZAR TORN 2
PER RECEPTES

AUX_CANVI_DE_DATA AUX_TORN_2_ACTIU



NETEJA REGISTRE QUE COMENCEM A UTILITZAR TORN 3
PER RECEPTES

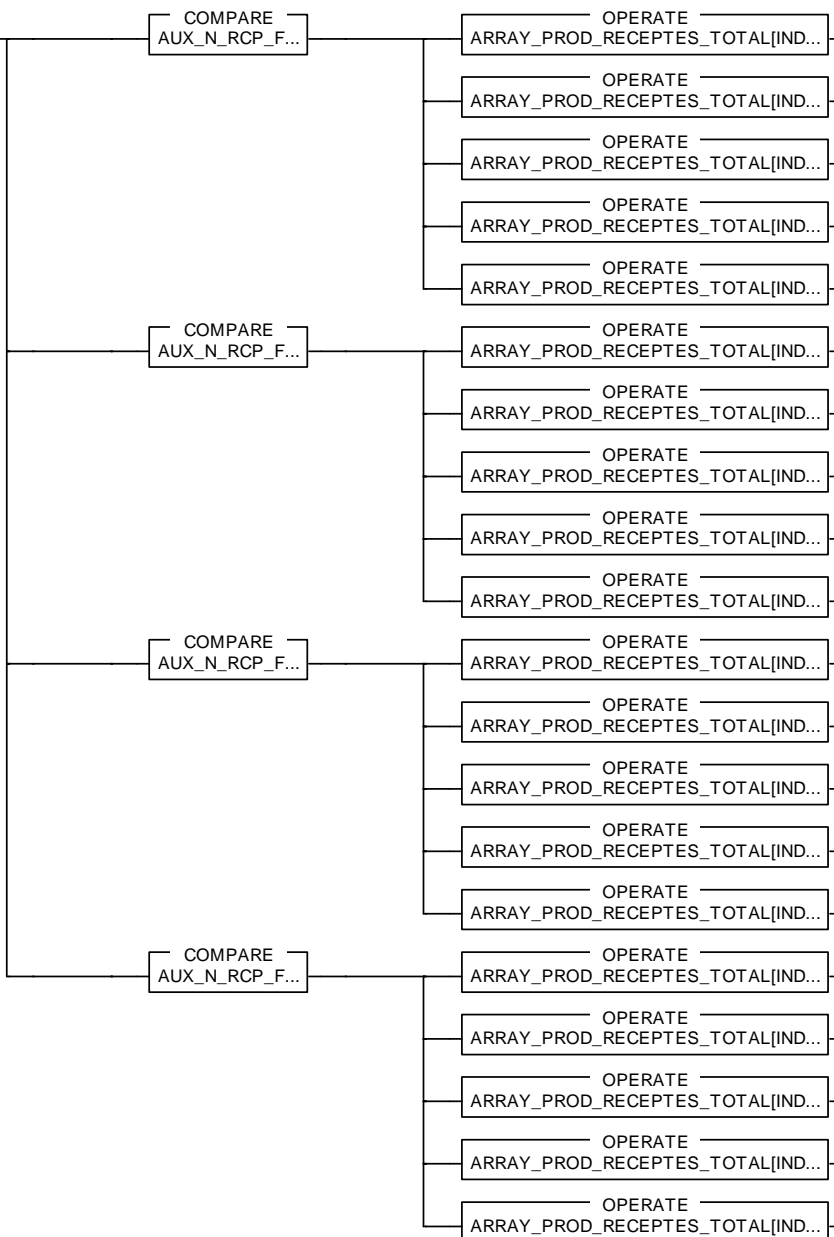
AUX_CANVI_DE_DATA AUX_TORN_3_ACTIU



ACTUALITZACIO DE PALETS PRODUITS TOTAL PER RECEI

AUX_CANVI_DE_DATA

P



3.- PROGRAMA EN VISUAL BASIC

3.1.- BUTO_CONECTAR_CLICK

```
Private Sub CONECTAR_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles BUTO_CONECTAR.Click
```

```
    Dim value As Integer  
    Dim verificacio As Integer  
    verificacio = 0  
    value = 300
```

```
    If plc_lectura.Connected = False Then  
        plc_lectura = New TcpClient  
        plc_lectura.NoDelay = True  
        plc_lectura.SendBufferSize = value  
        plc_lectura.ReceiveBufferSize = value
```

```
    Try  
        plc_lectura.Connect(IPAddress.Parse("172.27.0.206"), 502)  
    Catch ex As Exception  
        BUTO_CONECTAR.Image = My.Resources.desconectat  
        BUTO_LLEGIR.Visible = False  
    End Try
```

```
    If plc_lectura.Connected = True Then  
        BUTO_CONECTAR.Image = My.Resources.conectat  
        Timer_hores.Enabled = True  
        Timer_almes.Enabled = True  
        'Verificar configuracio plc  
        comando_lectura = plc_lectura.GetStream  
        envio_comando_lectura(990, 1)  
        recepcio_dades(990, 1)  
        verificacio = llista_bytes_rebut(10)  
        verificacio = verificacio  
        If verificacio = 0 Then  
            BUTO_LLEGIR.Visible = False  
            CHK_RECEPTA.Visible = False  
            CHK_GRAFIQUES.Visible = False  
            CHK_DIA.Visible = False  
            CHK_RECEPTA.Checked = False  
            CHK_DIA.Checked = False
```

```
CHK_GRAFIQUES.Checked = False
MsgBox("AQUEST SOFTWARE NO ESTAR REGISTRAT")
plc_lectura.Close()
If plc_lectura.Connected = False Then
    BUTO_CONECTAR.Image = My.Resources.desconectat
    BUTO_LLEGIR.Visible = False
End
End If
Else
    CHK_RECEPTA.Visible = True
    CHK_DIA.Visible = True
    CHK_GRAFIQUES.Visible = True
    CrearLLibreEXcel_HORES("C:\CAPE\CAPE" + TextBox8.Text + ".xlsx")
    'crear objecte excel
    xlApp = New Excel.ApplicationClass
    xlWorkBook = xlApp.Workbooks.Open("C:\CAPE\CAPE" + TextBox8.Text +
".xlsx")
    xlWorkSheet = xlWorkBook.Worksheets("ALARMS")
    omplir_taula_excel_ALARMES("ALARMS")
    xlWorkSheet = xlWorkBook.Worksheets("TIME")
    omplir_taula_excel_HORES("TIME")
    xlAppBD = New Excel.ApplicationClass
    xlWorkBookBD =
xlAppBD.Workbooks.Open("C:\CAPE\NO_DELETE\alarmes_cpiA.xlsx")
    xlWorkSheetBD = xlWorkBookBD.Worksheets("alarms")
    taula_excel = xlWorkSheetBD.Range("A1:CW31").Value
    inici_casella_prod_hores = 3
    inici_casella_prod_alarms = 3
    CHK_ALARMES.Visible = True
    CHK_PROD_HORES.Visible = True
End If

If CHK_DIA.Checked = True Or CHK_RECEPTA.Checked = True Then
    BUTO_LLEGIR.Visible = True
End If
comando_lectura = plc_lectura.GetStream
End If
Elseif plc_lectura.Connected = True Then
    plc_lectura.Close()
    If plc_lectura.Connected = False Then
        BUTO_CONECTAR.Image = My.Resources.desconectat
        BUTO_LLEGIR.Visible = False
```

```
    Timer_hores.Enabled = False
    Timer_almes.Enabled = False
    xlWorkBookBD.Close()
    xlAppBD.Quit()
    releaseObject(xlAppBD)
    releaseObject(xlWorkBookBD)
    releaseObject(xlWorkSheetBD)
End If
End If
End Sub
```

3.2.- BUTO_LLEGIR_CLICK

```
Private Sub BUTO_LLEGIR_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BUTO_LLEGIR.Click
    ProgressBar1.Value = 0
    ProgressBar1.Visible = True
    lectura_index = 30
    lectura_torn = 5
    plc_lectura.ReceiveBufferSize = 300

    While comando_escritura.DataAvailable = True Or comando_lectura.DataAvailable
= True
        Wait(1)
    End While
    Cursor = Cursors.WaitCursor
    Timer_hores.Enabled = False
    Timer_almes.Enabled = False

    If CHK_DIA.Checked = True Then
        For n = 0 To 24
            index = n
            If plc_lectura.Connected = True Then
                comando_escritura = plc_lectura.GetStream
                envio_comando_escritura(721, n
                Wait(300)
                While comando_escritura.DataAvailable = True Or
comando_lectura.DataAvailable = True
                    Wait(1)
                End While
            End If
            If plc_lectura.Connected = True Then
```

```
comando_lectura = plc_lectura.GetStream
envio_comando_lectura(734, 100)
If plc_lectura.Connected = True Then
    recepcio_dades(734, 100)
    extraccio_dades_dia(0)
End If
ProgressBar1.Value = ProgressBar1.Value + 1
ProgressBar1.Refresh()
End If
Wait(300)
While comando_escritura.DataAvailable = True Or
comando_lectura.DataAvailable = True
    Wait(1)
End While
Next n
refrescar_taula_dia(0)
ProgressBar1.Value = 0
End If

If CHK_RECEPTA.Checked = True Then
    lectura_index = 30
    While comando_escritura.DataAvailable = True Or
comando_lectura.DataAvailable = True
        Wait(1)
    End While
    For n = 0 To 24
        index = n
        comando_escritura = plc_lectura.GetStream
        envio_comando_escritura(721, n)
        Wait(300)
        While comando_escritura.DataAvailable = True Or
comando_lectura.DataAvailable = True
            Wait(1)
        End While
        For s = 0 To 3
            envio_comando_escritura(712, s)
            'While lectura_torn <> s
            Wait(300)
            While comando_escritura.DataAvailable = True Or
comando_lectura.DataAvailable = True
                Wait(1)
            End While
```

```
comando_lectura = plc_lectura.GetStream
envio_comando_lectura(760, 100)
If plc_lectura.Connected = True Then
  recepcio_dades(760, 100)
  extraccio_dades_recepta(s)
  Wait(300)
  While comando_escritura.DataAvailable = True Or
comando_lectura.DataAvailable = True
    Wait(1)
  End While
  comando_lectura = plc_lectura.GetStream
  envio_comando_lectura(800, 80)
  If plc_lectura.Connected = True Then
    recepcio_dades(800, 80)
    extraccio_dades_ref_recepta(s, n)
  End If
End If
Next s
ProgressBar1.Value = ProgressBar1.Value + 1
Wait(300)
While comando_escritura.DataAvailable = True Or
comando_lectura.DataAvailable = True
  Wait(1)
End While
Next n
ProgressBar1.Value = 0
refrescar_taula_rcp(1, 0)
txt_indic_bloc_torn.Text = "A"
txt_indic_torn.Text = "T1"
ProgressBar1.Visible = False
End If
BUTO_EXCEL.Visible = True
txt_path.Visible = True
Timer_hores.Enabled = True
Timer_almes.Enabled = True
Cursor = Cursors.Arrow
End Sub
```

3.3.- ENVIO_COMANDO_ESCRIPTURA

Function envio_comando_escritura(ByVal adreça As Integer, ByVal valor As Integer) As
Byte

```
Dim adreça_H As Byte
Dim adreça_L As Byte
Dim valor_escritura_H As Byte
Dim valor_escritura_L As Byte
adreça_H = adreça Mod 256
adreça_L = adreça \ 256
valor_escritura_H = valor Mod 256
valor_escritura_L = valor \ 256

trama_escriure = "0000000000000000"
bytes_escriure = Encoding.Default.GetBytes(trama_escriure)
bytes_escriure(0) = "0"
bytes_escriure(1) = "0"
bytes_escriure(2) = "0"
bytes_escriure(3) = "0"
bytes_escriure(4) = "0"
bytes_escriure(5) = "9"
bytes_escriure(6) = "1"
bytes_escriure(7) = "16"
bytes_escriure(8) = adreça_L
bytes_escriure(9) = adreça_H
bytes_escriure(10) = "0"
bytes_escriure(11) = "1"
bytes_escriure(12) = "2"
bytes_escriure(13) = valor_escritura_L
bytes_escriure(14) = valor_escritura_H

comando_escritura.Write(bytes_escriure, 0, bytes_escriure.Length)
trama_escriure = ""
bytes_escriure = Nothing
If plc_lectura.Connected = True Then
    While comando_escritura.DataAvailable = False
        Wait(1)
    End While
    If comando_escritura.DataAvailable = True Then
        ReDim bytes_escriure(plc_lectura.ReceiveBufferSize)
        comando_escritura.Read(bytes_escriure, 0, bytes_escriure.Length)
        llista_bytes_rebut = bytes_escriure
    End If
    bytes_escriure = Nothing
End If
End Function
```

3.4.- EXTRACCIÓ_DADES_DIA

Function extraccio_dades_dia(ByVal entrada As Integer) As Integer

```
Dim n As Integer
Dim data1 As String
Dim dia As String
Dim mes As String
paraula0 = llista_bytes_rebuts(9)
paraula1 = llista_bytes_rebuts(10)
paraula2 = llista_bytes_rebuts(11)
paraula3 = llista_bytes_rebuts(12)
```

```
For n = 9 To 57 Step 2
    paraula0 = llista_bytes_rebuts(n)
    paraula1 = llista_bytes_rebuts(n + 1)
    paraula2 = llista_bytes_rebuts(n + 2)
    paraula3 = llista_bytes_rebuts(n + 3)
```

```
Select Case n
```

```
Case 9
```

```
    prod_dia_palets_tot(index) = (paraula0 * 256) + paraula1
```

```
Case 11
```

```
    prod_dia_tmarxa_tot(index) = CStr((paraula0 * 256) + paraula1)
```

```
    prod_dia_tmarxa_tot(index) = hora_minuts(prod_dia_tmarxa_tot(index))
```

```
    prod_dia_tmarxa_tot_aux(index) =
```

```
convert_temps_to_num(prod_dia_tmarxa_tot(index))
```

```
    prod_dia_tmarxa_tot_aux(index) = prod_dia_tmarxa_tot_aux(index) * 60
```

```
Case 13
```

```
    prod_dia_tparo_tot(index) = CStr((paraula0 * 256) + paraula1)
```

```
    prod_dia_tparo_tot(index) = hora_minuts(prod_dia_tparo_tot(index))
```

```
Case 15
```

```
    prod_dia_talarm_tot(index) = CStr((paraula0 * 256) + paraula1)
```

```
    prod_dia_talarm_tot(index) = hora_minuts(prod_dia_talarm_tot(index))
```

```
Case 17
```

```
    ' prod_dia_mitja_tot(index) = enter_a_real(paraula0, paraula1, paraula2,
paraula3)
```

```
    If prod_dia_tmarxa_tot_aux(index) <> 0 Then
```

```
        prod_dia_mitja_tot(index) = prod_dia_palets_tot(index) /
```

```
prod_dia_tmarxa_tot_aux(index)
```

```
        prod_dia_mitja_tot(index) = Strings.Left(prod_dia_mitja_tot(index), 3)
```

```
    Else
```

```
        prod_dia_mitja_tot(index) = 0.0
```



```
End If
Case 21
    data1 = Hex((paraula0 * 256) + paraula1)
    data1 = "0000" + data1
    data1 = Microsoft.VisualBasic.Right(data1, 4)
    mes = Microsoft.VisualBasic.Right(data1, 2)
    dia = Microsoft.VisualBasic.Left(data1, 2)
    prod_dia_data(index) = mes + "/" + dia
Case 23
    prod_dia_any(index) = Hex((paraula0 * 256) + paraula1)
    prod_dia_data(index) = prod_dia_data(index) + "/" + prod_dia_any(index)
Case 25
    prod_dia_palets_t1(index) = CStr((paraula0 * 256) + paraula1)
Case 27
    prod_dia_tmarxa_t1(index) = CStr((paraula0 * 256) + paraula1)
    prod_dia_tmarxa_t1(index) = hora_minuts(prod_dia_tmarxa_t1(index))
    prod_dia_tmarxa_t1_aux(index) =
convert_temps_to_num(prod_dia_tmarxa_t1(index))
    prod_dia_tmarxa_t1_aux(index) = prod_dia_tmarxa_t1_aux(index) * 60
Case 29
    prod_dia_tparo_t1(index) = CStr((paraula0 * 256) + paraula1)
    prod_dia_tparo_t1(index) = hora_minuts(prod_dia_tparo_t1(index))
Case 31
    prod_dia_talarm_t1(index) = CStr((paraula0 * 256) + paraula1)
    prod_dia_talarm_t1(index) = hora_minuts(prod_dia_talarm_t1(index))
Case 33
    If prod_dia_tmarxa_t1_aux(index) <> 0 Then
        prod_dia_mitja_t1(index) = prod_dia_palets_t1(index) /
prod_dia_tmarxa_t1_aux(index)
        prod_dia_mitja_t1(index) = Strings.Left(prod_dia_mitja_t1(index), 3)
    Else
        prod_dia_mitja_t1(index) = 0.0
    End If
Case 37
    prod_dia_palets_t2(index) = CStr((paraula0 * 256) + paraula1)
Case 39
    prod_dia_tmarxa_t2(index) = CStr((paraula0 * 256) + paraula1)
    prod_dia_tmarxa_t2(index) = hora_minuts(prod_dia_tmarxa_t2(index))
    prod_dia_tmarxa_t2_aux(index) =
convert_temps_to_num(prod_dia_tmarxa_t2(index))
    prod_dia_tmarxa_t2_aux(index) = prod_dia_tmarxa_t2_aux(index) * 60
Case 41
```

```
    prod_dia_tparo_t2(index) = CStr((paraula0 * 256) + paraula1)
    prod_dia_tparo_t2(index) = hora_minuts(prod_dia_tparo_t2(index))
Case 43
    prod_dia_talarm_t2(index) = CStr((paraula0 * 256) + paraula1)
    prod_dia_talarm_t2(index) = hora_minuts(prod_dia_talarm_t2(index))
Case 45
    If prod_dia_tmarxa_t2_aux(index) <> 0 Then
        prod_dia_mitja_t2(index)      =      prod_dia_palets_t2(index)      /
prod_dia_tmarxa_t2_aux(index)
        prod_dia_mitja_t2(index) = Strings.Left(prod_dia_mitja_t2(index), 3)
    Else
        prod_dia_mitja_t2(index) = 0.0
    End If
Case 49
    prod_dia_palets_t3(index) = CStr((paraula0 * 256) + paraula1)
Case 51
    prod_dia_tmarxa_t3(index) = CStr((paraula0 * 256) + paraula1)
    prod_dia_tmarxa_t3(index) = hora_minuts(prod_dia_tmarxa_t3(index))
    prod_dia_tmarxa_t3_aux(index)                                     =
convert_temps_to_num(prod_dia_tmarxa_t3(index))
    prod_dia_tmarxa_t3_aux(index) = prod_dia_tmarxa_t3_aux(index) * 60
Case 53
    prod_dia_tparo_t3(index) = CStr((paraula0 * 256) + paraula1)
    prod_dia_tparo_t3(index) = hora_minuts(prod_dia_tparo_t3(index))
Case 55
    prod_dia_talarm_t3(index) = CStr((paraula0 * 256) + paraula1)
    prod_dia_talarm_t3(index) = hora_minuts(prod_dia_talarm_t3(index))
Case 57
    If prod_dia_tmarxa_t3_aux(index) <> 0 Then
        prod_dia_mitja_t3(index)      =      prod_dia_palets_t3(index)      /
prod_dia_tmarxa_t3_aux(index)
        prod_dia_mitja_t3(index) = Strings.Left(prod_dia_mitja_t3(index), 3)
    Else
        prod_dia_mitja_t3(index) = 0.0
    End If
End Select
Next n
End Function
```

3.5.- REFRESCAR_TAULA_DIA

Function refrescar_taula_dia(ByVal entrada As Integer) As Integer

```
Dim txt_prod_dia_data As TextBox() = New TextBox() {txt_prod_dia_data1,  
txt_prod_dia_data2, txt_prod_dia_data3, txt_prod_dia_data4, txt_prod_dia_data5,  
txt_prod_dia_data6, txt_prod_dia_data7, txt_prod_dia_data8, txt_prod_dia_data9,  
txt_prod_dia_data10, txt_prod_dia_data11, txt_prod_dia_data12, txt_prod_dia_data13,  
txt_prod_dia_data14, txt_prod_dia_data15, txt_prod_dia_data16, txt_prod_dia_data17,  
txt_prod_dia_data18, txt_prod_dia_data19, txt_prod_dia_data20, txt_prod_dia_data21,  
txt_prod_dia_data22, txt_prod_dia_data23, txt_prod_dia_data24, txt_prod_dia_data25}
```

```
Dim txt_prod_dia_t1_pal As TextBox() = New TextBox() {txt_prod_dia_t1_pal1,  
txt_prod_dia_t1_pal2, txt_prod_dia_t1_pal3, txt_prod_dia_t1_pal4, txt_prod_dia_t1_pal5,  
txt_prod_dia_t1_pal6, txt_prod_dia_t1_pal7, txt_prod_dia_t1_pal8, txt_prod_dia_t1_pal9,  
txt_prod_dia_t1_pal10, txt_prod_dia_t1_pal11, txt_prod_dia_t1_pal12,  
txt_prod_dia_t1_pal13, txt_prod_dia_t1_pal14, txt_prod_dia_t1_pal15,  
txt_prod_dia_t1_pal16, txt_prod_dia_t1_pal17, txt_prod_dia_t1_pal18,  
txt_prod_dia_t1_pal19, txt_prod_dia_t1_pal20, txt_prod_dia_t1_pal21,  
txt_prod_dia_t1_pal22, txt_prod_dia_t1_pal23, txt_prod_dia_t1_pal24,  
txt_prod_dia_t1_pal25}
```

```
Dim txt_prod_dia_t1_m As TextBox() = New TextBox() {txt_prod_dia_t1_m1,  
txt_prod_dia_t1_m2, txt_prod_dia_t1_m3, txt_prod_dia_t1_m4, txt_prod_dia_t1_m5,  
txt_prod_dia_t1_m6, txt_prod_dia_t1_m7, txt_prod_dia_t1_m8, txt_prod_dia_t1_m9,  
txt_prod_dia_t1_m10, txt_prod_dia_t1_m11, txt_prod_dia_t1_m12, txt_prod_dia_t1_m13,  
txt_prod_dia_t1_m14, txt_prod_dia_t1_m15, txt_prod_dia_t1_m16, txt_prod_dia_t1_m17,  
txt_prod_dia_t1_m18, txt_prod_dia_t1_m19, txt_prod_dia_t1_m20, txt_prod_dia_t1_m21,  
txt_prod_dia_t1_m22, txt_prod_dia_t1_m23, txt_prod_dia_t1_m24, txt_prod_dia_t1_m25}
```

```
Dim txt_prod_dia_t1_p As TextBox() = New TextBox() {txt_prod_dia_t1_p1,  
txt_prod_dia_t1_p2, txt_prod_dia_t1_p3, txt_prod_dia_t1_p4, txt_prod_dia_t1_p5,  
txt_prod_dia_t1_p6, txt_prod_dia_t1_p7, txt_prod_dia_t1_p8, txt_prod_dia_t1_p9,  
txt_prod_dia_t1_p10, txt_prod_dia_t1_p11, txt_prod_dia_t1_p12, txt_prod_dia_t1_p13,  
txt_prod_dia_t1_p14, txt_prod_dia_t1_p15, txt_prod_dia_t1_p16, txt_prod_dia_t1_p17,  
txt_prod_dia_t1_p18, txt_prod_dia_t1_p19, txt_prod_dia_t1_p20, txt_prod_dia_t1_p21,  
txt_prod_dia_t1_p22, txt_prod_dia_t1_p23, txt_prod_dia_t1_p24, txt_prod_dia_t1_p25}
```

```
Dim txt_prod_dia_t1_al As TextBox() = New TextBox() {txt_prod_dia_t1_al1,  
txt_prod_dia_t1_al2, txt_prod_dia_t1_al3, txt_prod_dia_t1_al4, txt_prod_dia_t1_al5,  
txt_prod_dia_t1_al6, txt_prod_dia_t1_al7, txt_prod_dia_t1_al8, txt_prod_dia_t1_al9,  
txt_prod_dia_t1_al10, txt_prod_dia_t1_al11, txt_prod_dia_t1_al12, txt_prod_dia_t1_al13,  
txt_prod_dia_t1_al14, txt_prod_dia_t1_al15, txt_prod_dia_t1_al16, txt_prod_dia_t1_al17,  
txt_prod_dia_t1_al18, txt_prod_dia_t1_al19, txt_prod_dia_t1_al20, txt_prod_dia_t1_al21,  
txt_prod_dia_t1_al22, txt_prod_dia_t1_al23, txt_prod_dia_t1_al24, txt_prod_dia_t1_al25}
```

```
Dim txt_prod_dia_t1_mj As TextBox() = New TextBox() {txt_prod_dia_t1_mj1,  
txt_prod_dia_t1_mj2, txt_prod_dia_t1_mj3, txt_prod_dia_t1_mj4, txt_prod_dia_t1_mj5,  
txt_prod_dia_t1_mj6, txt_prod_dia_t1_mj7, txt_prod_dia_t1_mj8, txt_prod_dia_t1_mj9,  
txt_prod_dia_t1_mj10, txt_prod_dia_t1_mj11, txt_prod_dia_t1_mj12,  
txt_prod_dia_t1_mj13, txt_prod_dia_t1_mj14, txt_prod_dia_t1_mj15}
```

```
txt_prod_dia_t1_mj16,          txt_prod_dia_t1_mj17,          txt_prod_dia_t1_mj18,  
txt_prod_dia_t1_mj19,          txt_prod_dia_t1_mj20,          txt_prod_dia_t1_mj21,  
txt_prod_dia_t1_mj22,          txt_prod_dia_t1_mj23,          txt_prod_dia_t1_mj24,  
txt_prod_dia_t1_mj25}
```

```
Dim txt_prod_dia_t2_pal As TextBox() = New TextBox() {txt_prod_dia_t2_pal1,  
txt_prod_dia_t2_pal2, txt_prod_dia_t2_pal3, txt_prod_dia_t2_pal4, txt_prod_dia_t2_pal5,  
txt_prod_dia_t2_pal6, txt_prod_dia_t2_pal7, txt_prod_dia_t2_pal8, txt_prod_dia_t2_pal9,  
txt_prod_dia_t2_pal10,          txt_prod_dia_t2_pal11,          txt_prod_dia_t2_pal12,  
txt_prod_dia_t2_pal13,          txt_prod_dia_t2_pal14,          txt_prod_dia_t2_pal15,  
txt_prod_dia_t2_pal16,          txt_prod_dia_t2_pal17,          txt_prod_dia_t2_pal18,  
txt_prod_dia_t2_pal19,          txt_prod_dia_t2_pal20,          txt_prod_dia_t2_pal21,  
txt_prod_dia_t2_pal22,          txt_prod_dia_t2_pal23,          txt_prod_dia_t2_pal24,  
txt_prod_dia_t2_pal25}
```

```
Dim txt_prod_dia_t2_m As TextBox() = New TextBox() {txt_prod_dia_t2_m1,  
txt_prod_dia_t2_m2, txt_prod_dia_t2_m3, txt_prod_dia_t2_m4, txt_prod_dia_t2_m5,  
txt_prod_dia_t2_m6, txt_prod_dia_t2_m7, txt_prod_dia_t2_m8, txt_prod_dia_t2_m9,  
txt_prod_dia_t2_m10, txt_prod_dia_t2_m11, txt_prod_dia_t2_m12, txt_prod_dia_t2_m13,  
txt_prod_dia_t2_m14, txt_prod_dia_t2_m15, txt_prod_dia_t2_m16, txt_prod_dia_t2_m17,  
txt_prod_dia_t2_m18, txt_prod_dia_t2_m19, txt_prod_dia_t2_m20, txt_prod_dia_t2_m21,  
txt_prod_dia_t2_m22, txt_prod_dia_t2_m23, txt_prod_dia_t2_m24, txt_prod_dia_t2_m25}
```

```
Dim txt_prod_dia_t2_p As TextBox() = New TextBox() {txt_prod_dia_t2_p1,  
txt_prod_dia_t2_p2, txt_prod_dia_t2_p3, txt_prod_dia_t2_p4, txt_prod_dia_t2_p5,  
txt_prod_dia_t2_p6, txt_prod_dia_t2_p7, txt_prod_dia_t2_p8, txt_prod_dia_t2_p9,  
txt_prod_dia_t2_p10, txt_prod_dia_t2_p11, txt_prod_dia_t2_p12, txt_prod_dia_t2_p13,  
txt_prod_dia_t2_p14, txt_prod_dia_t2_p15, txt_prod_dia_t2_p16, txt_prod_dia_t2_p17,  
txt_prod_dia_t2_p18, txt_prod_dia_t2_p19, txt_prod_dia_t2_p20, txt_prod_dia_t2_p21,  
txt_prod_dia_t2_p22, txt_prod_dia_t2_p23, txt_prod_dia_t2_p24, txt_prod_dia_t2_p25}
```

```
Dim txt_prod_dia_t2_al As TextBox() = New TextBox() {txt_prod_dia_t2_al1,  
txt_prod_dia_t2_al2, txt_prod_dia_t2_al3, txt_prod_dia_t2_al4, txt_prod_dia_t2_al5,  
txt_prod_dia_t2_al6, txt_prod_dia_t2_al7, txt_prod_dia_t2_al8, txt_prod_dia_t2_al9,  
txt_prod_dia_t2_al10, txt_prod_dia_t2_al11, txt_prod_dia_t2_al12, txt_prod_dia_t2_al13,  
txt_prod_dia_t2_al14, txt_prod_dia_t2_al15, txt_prod_dia_t2_al16, txt_prod_dia_t2_al17,  
txt_prod_dia_t2_al18, txt_prod_dia_t2_al19, txt_prod_dia_t2_al20, txt_prod_dia_t2_al21,  
txt_prod_dia_t2_al22, txt_prod_dia_t2_al23, txt_prod_dia_t2_al24, txt_prod_dia_t2_al25}
```

```
Dim txt_prod_dia_t2_mj As TextBox() = New TextBox() {txt_prod_dia_t2_mj1,  
txt_prod_dia_t2_mj2, txt_prod_dia_t2_mj3, txt_prod_dia_t2_mj4, txt_prod_dia_t2_mj5,  
txt_prod_dia_t2_mj6, txt_prod_dia_t2_mj7, txt_prod_dia_t2_mj8, txt_prod_dia_t2_mj9,  
txt_prod_dia_t2_mj10,          txt_prod_dia_t2_mj11,          txt_prod_dia_t2_mj12,  
txt_prod_dia_t2_mj13,          txt_prod_dia_t2_mj14,          txt_prod_dia_t2_mj15,  
txt_prod_dia_t2_mj16,          txt_prod_dia_t2_mj17,          txt_prod_dia_t2_mj18,  
txt_prod_dia_t2_mj19,          txt_prod_dia_t2_mj20,          txt_prod_dia_t2_mj21,
```

```
txt_prod_dia_t2_mj22,          txt_prod_dia_t2_mj23,          txt_prod_dia_t2_mj24,  
txt_prod_dia_t2_mj25}
```

```
Dim txt_prod_dia_t3_pal As TextBox() = New TextBox() {txt_prod_dia_t3_pal1,  
txt_prod_dia_t3_pal2, txt_prod_dia_t3_pal3, txt_prod_dia_t3_pal4, txt_prod_dia_t3_pal5,  
txt_prod_dia_t3_pal6, txt_prod_dia_t3_pal7, txt_prod_dia_t3_pal8, txt_prod_dia_t3_pal9,  
txt_prod_dia_t3_pal10,          txt_prod_dia_t3_pal11,          txt_prod_dia_t3_pal12,  
txt_prod_dia_t3_pal13,          txt_prod_dia_t3_pal14,          txt_prod_dia_t3_pal15,  
txt_prod_dia_t3_pal16,          txt_prod_dia_t3_pal17,          txt_prod_dia_t3_pal18,  
txt_prod_dia_t3_pal19,          txt_prod_dia_t3_pal20,          txt_prod_dia_t3_pal21,  
txt_prod_dia_t3_pal22,          txt_prod_dia_t3_pal23,          txt_prod_dia_t3_pal24,  
txt_prod_dia_t3_pal25}
```

```
Dim txt_prod_dia_t3_m As TextBox() = New TextBox() {txt_prod_dia_t3_m1,  
txt_prod_dia_t3_m2, txt_prod_dia_t3_m3, txt_prod_dia_t3_m4, txt_prod_dia_t3_m5,  
txt_prod_dia_t3_m6, txt_prod_dia_t3_m7, txt_prod_dia_t3_m8, txt_prod_dia_t3_m9,  
txt_prod_dia_t3_m10, txt_prod_dia_t3_m11, txt_prod_dia_t3_m12, txt_prod_dia_t3_m13,  
txt_prod_dia_t3_m14, txt_prod_dia_t3_m15, txt_prod_dia_t3_m16, txt_prod_dia_t3_m17,  
txt_prod_dia_t3_m18, txt_prod_dia_t3_m19, txt_prod_dia_t3_m20, txt_prod_dia_t3_m21,  
txt_prod_dia_t3_m22, txt_prod_dia_t3_m23, txt_prod_dia_t3_m24, txt_prod_dia_t3_m25}
```

```
Dim txt_prod_dia_t3_p As TextBox() = New TextBox() {txt_prod_dia_t3_p1,  
txt_prod_dia_t3_p2, txt_prod_dia_t3_p3, txt_prod_dia_t3_p4, txt_prod_dia_t3_p5,  
txt_prod_dia_t3_p6, txt_prod_dia_t3_p7, txt_prod_dia_t3_p8, txt_prod_dia_t3_p9,  
txt_prod_dia_t3_p10, txt_prod_dia_t3_p11, txt_prod_dia_t3_p12, txt_prod_dia_t3_p13,  
txt_prod_dia_t3_p14, txt_prod_dia_t3_p15, txt_prod_dia_t3_p16, txt_prod_dia_t3_p17,  
txt_prod_dia_t3_p18, txt_prod_dia_t3_p19, txt_prod_dia_t3_p20, txt_prod_dia_t3_p21,  
txt_prod_dia_t3_p22, txt_prod_dia_t3_p23, txt_prod_dia_t3_p24, txt_prod_dia_t3_p25}
```

```
Dim txt_prod_dia_t3_al As TextBox() = New TextBox() {txt_prod_dia_t3_al1,  
txt_prod_dia_t3_al2, txt_prod_dia_t3_al3, txt_prod_dia_t3_al4, txt_prod_dia_t3_al5,  
txt_prod_dia_t3_al6, txt_prod_dia_t3_al7, txt_prod_dia_t3_al8, txt_prod_dia_t3_al9,  
txt_prod_dia_t3_al10, txt_prod_dia_t3_al11, txt_prod_dia_t3_al12, txt_prod_dia_t3_al13,  
txt_prod_dia_t3_al14, txt_prod_dia_t3_al15, txt_prod_dia_t3_al16, txt_prod_dia_t3_al17,  
txt_prod_dia_t3_al18, txt_prod_dia_t3_al19, txt_prod_dia_t3_al20, txt_prod_dia_t3_al21,  
txt_prod_dia_t3_al22, txt_prod_dia_t3_al23, txt_prod_dia_t3_al24, txt_prod_dia_t3_al25}
```

```
Dim txt_prod_dia_t3_mj As TextBox() = New TextBox() {txt_prod_dia_t3_mj1,  
txt_prod_dia_t3_mj2, txt_prod_dia_t3_mj3, txt_prod_dia_t3_mj4, txt_prod_dia_t3_mj5,  
txt_prod_dia_t3_mj6, txt_prod_dia_t3_mj7, txt_prod_dia_t3_mj8, txt_prod_dia_t3_mj9,  
txt_prod_dia_t3_mj10,          txt_prod_dia_t3_mj11,          txt_prod_dia_t3_mj12,  
txt_prod_dia_t3_mj13,          txt_prod_dia_t3_mj14,          txt_prod_dia_t3_mj15,  
txt_prod_dia_t3_mj16,          txt_prod_dia_t3_mj17,          txt_prod_dia_t3_mj18,  
txt_prod_dia_t3_mj19,          txt_prod_dia_t3_mj20,          txt_prod_dia_t3_mj21,  
txt_prod_dia_t3_mj22,          txt_prod_dia_t3_mj23,          txt_prod_dia_t3_mj24,  
txt_prod_dia_t3_mj25}
```

```
Dim txt_prod_dia_tot_pal As TextBox() = New TextBox() {txt_prod_dia_tot_pal1,  
txt_prod_dia_tot_pal2,          txt_prod_dia_tot_pal3,          txt_prod_dia_tot_pal4,  
txt_prod_dia_tot_pal5,          txt_prod_dia_tot_pal6,          txt_prod_dia_tot_pal7,  
txt_prod_dia_tot_pal8,          txt_prod_dia_tot_pal9,          txt_prod_dia_tot_pal10,  
txt_prod_dia_tot_pal11,         txt_prod_dia_tot_pal12,         txt_prod_dia_tot_pal13,  
txt_prod_dia_tot_pal14,         txt_prod_dia_tot_pal15,         txt_prod_dia_tot_pal16,  
txt_prod_dia_tot_pal17,         txt_prod_dia_tot_pal18,         txt_prod_dia_tot_pal19,  
txt_prod_dia_tot_pal20,         txt_prod_dia_tot_pal21,         txt_prod_dia_tot_pal22,  
txt_prod_dia_tot_pal23, txt_prod_dia_tot_pal24, txt_prod_dia_tot_pal25}
```

```
Dim txt_prod_dia_tot_m As TextBox() = New TextBox() {txt_prod_dia_tot_m1,  
txt_prod_dia_tot_m2, txt_prod_dia_tot_m3, txt_prod_dia_tot_m4, txt_prod_dia_tot_m5,  
txt_prod_dia_tot_m6, txt_prod_dia_tot_m7, txt_prod_dia_tot_m8, txt_prod_dia_tot_m9,  
txt_prod_dia_tot_m10,          txt_prod_dia_tot_m11,          txt_prod_dia_tot_m12,  
txt_prod_dia_tot_m13,          txt_prod_dia_tot_m14,          txt_prod_dia_tot_m15,  
txt_prod_dia_tot_m16,          txt_prod_dia_tot_m17,          txt_prod_dia_tot_m18,  
txt_prod_dia_tot_m19,          txt_prod_dia_tot_m20,          txt_prod_dia_tot_m21,  
txt_prod_dia_tot_m22,          txt_prod_dia_tot_m23,          txt_prod_dia_tot_m24,  
txt_prod_dia_tot_m25}
```

```
Dim txt_prod_dia_tot_p As TextBox() = New TextBox() {txt_prod_dia_tot_p1,  
txt_prod_dia_tot_p2, txt_prod_dia_tot_p3, txt_prod_dia_tot_p4, txt_prod_dia_tot_p5,  
txt_prod_dia_tot_p6, txt_prod_dia_tot_p7, txt_prod_dia_tot_p8, txt_prod_dia_tot_p9,  
txt_prod_dia_tot_p10, txt_prod_dia_tot_p11, txt_prod_dia_tot_p12, txt_prod_dia_tot_p13,  
txt_prod_dia_tot_p14, txt_prod_dia_tot_p15, txt_prod_dia_tot_p16, txt_prod_dia_tot_p17,  
txt_prod_dia_tot_p18, txt_prod_dia_tot_p19, txt_prod_dia_tot_p20, txt_prod_dia_tot_p21,  
txt_prod_dia_tot_p22, txt_prod_dia_tot_p23, txt_prod_dia_tot_p24, txt_prod_dia_tot_p25}
```

```
Dim txt_prod_dia_tot_al As TextBox() = New TextBox() {txt_prod_dia_tot_al1,  
txt_prod_dia_tot_al2, txt_prod_dia_tot_al3, txt_prod_dia_tot_al4, txt_prod_dia_tot_al5,  
txt_prod_dia_tot_al6, txt_prod_dia_tot_al7, txt_prod_dia_tot_al8, txt_prod_dia_tot_al9,  
txt_prod_dia_tot_al10,          txt_prod_dia_tot_al11,          txt_prod_dia_tot_al12,  
txt_prod_dia_tot_al13,          txt_prod_dia_tot_al14,          txt_prod_dia_tot_al15,  
txt_prod_dia_tot_al16,          txt_prod_dia_tot_al17,          txt_prod_dia_tot_al18,  
txt_prod_dia_tot_al19,          txt_prod_dia_tot_al20,          txt_prod_dia_tot_al21,  
txt_prod_dia_tot_al22,          txt_prod_dia_tot_al23,          txt_prod_dia_tot_al24,  
txt_prod_dia_tot_al25}
```

```
Dim txt_prod_dia_tot_mj As TextBox() = New TextBox() {txt_prod_dia_tot_mj1,  
txt_prod_dia_tot_mj2, txt_prod_dia_tot_mj3, txt_prod_dia_tot_mj4, txt_prod_dia_tot_mj5,  
txt_prod_dia_tot_mj6, txt_prod_dia_tot_mj7, txt_prod_dia_tot_mj8, txt_prod_dia_tot_mj9,  
txt_prod_dia_tot_mj10,          txt_prod_dia_tot_mj11,          txt_prod_dia_tot_mj12,  
txt_prod_dia_tot_mj13,          txt_prod_dia_tot_mj14,          txt_prod_dia_tot_mj15,  
txt_prod_dia_tot_mj16,          txt_prod_dia_tot_mj17,          txt_prod_dia_tot_mj18,  
txt_prod_dia_tot_mj19,          txt_prod_dia_tot_mj20,          txt_prod_dia_tot_mj21,
```

txt_prod_dia_tot_mj22, txt_prod_dia_tot_mj23, txt_prod_dia_tot_mj24,
txt_prod_dia_tot_mj25}

For s = 0 To 24

```
'txt_prod_dia_any(s).Text = prod_dia_any(s)
txt_prod_dia_data(s).Text = prod_dia_data(s)
txt_prod_dia_t1_pal(s).Text = prod_dia_palets_t1(s)
txt_prod_dia_t1_m(s).Text = prod_dia_tmarxa_t1(s)
txt_prod_dia_t1_p(s).Text = prod_dia_tparo_t1(s)
txt_prod_dia_t1_al(s).Text = prod_dia_talarm_t1(s)
txt_prod_dia_t1_mj(s).Text = prod_dia_mitja_t1(s)
txt_prod_dia_t2_pal(s).Text = prod_dia_palets_t2(s)
txt_prod_dia_t2_m(s).Text = prod_dia_tmarxa_t2(s)
txt_prod_dia_t2_p(s).Text = prod_dia_tparo_t2(s)
txt_prod_dia_t2_al(s).Text = prod_dia_talarm_t2(s)
txt_prod_dia_t2_mj(s).Text = prod_dia_mitja_t2(s)
txt_prod_dia_t3_pal(s).Text = prod_dia_palets_t3(s)
txt_prod_dia_t3_m(s).Text = prod_dia_tmarxa_t3(s)
txt_prod_dia_t3_p(s).Text = prod_dia_tparo_t3(s)
txt_prod_dia_t3_al(s).Text = prod_dia_talarm_t3(s)
txt_prod_dia_t3_mj(s).Text = prod_dia_mitja_t3(s)
txt_prod_dia_tot_pal(s).Text = prod_dia_palets_tot(s)
txt_prod_dia_tot_m(s).Text = prod_dia_tmarxa_tot(s)
txt_prod_dia_tot_p(s).Text = prod_dia_tparo_tot(s)
txt_prod_dia_tot_al(s).Text = prod_dia_talarm_tot(s)
txt_prod_dia_tot_mj(s).Text = prod_dia_mitja_tot(s)
```

Next

End Function

3.6.- BUTO_REFRESC_GRAFICA

Private Sub But_refresc_grafica_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles But_refresc_grafica.Click

```
Dim TEMPS_VISUALITZAR(25) As Single
GRAFICA_TOT.Series.Clear()
GRAFICA_TOT.Series.ResumeUpdates()
If Chk_graf_TOT.Checked = True Then
    If chk_graf_marxa.Checked = True Then
        GRAFICA_TOT.Series.Add("TOTAL_START_TIME")
    End If
End If
```

```
If chk_graf_paro.Checked = True Then
    GRAFICA_TOT.Series.Add("TOTAL_STOP_TIME")
End If
If chk_alarma_graf.Checked = True Then
    GRAFICA_TOT.Series.Add("TOTAL_ALARM_TIME")
End If
If Chk_qnt_palets.Checked = True Then
    GRAFICA_TOT.Series.Add("TOTAL_PALLETS_QUANTITY")
    GRAFICA_TOT.Series("TOTAL_PALLETS_QUANTITY").ChartType =
DataVisualization.Charting.SeriesChartType.Line
End If
End If

If Chk_graf_T1.Checked = True Then
    If chk_graf_marxa.Checked = True Then
        GRAFICA_TOT.Series.Add("T1_START_TIME")
    End If
    If chk_graf_paro.Checked = True Then
        GRAFICA_TOT.Series.Add("T1_STOP_TIME")
    End If
    If chk_alarma_graf.Checked = True Then
        GRAFICA_TOT.Series.Add("T1_ALARM_TIME")
    End If
    If Chk_qnt_palets.Checked = True Then
        GRAFICA_TOT.Series.Add("T1_PALLETS_QUANTITY")
        GRAFICA_TOT.Series("T1_PALLETS_QUANTITY").ChartType =
DataVisualization.Charting.SeriesChartType.Line
    End If
End If

If Chk_graf_T2.Checked = True Then
    If chk_graf_marxa.Checked = True Then
        GRAFICA_TOT.Series.Add("T2_START_TIME")
    End If
    If chk_graf_paro.Checked = True Then
        GRAFICA_TOT.Series.Add("T2_STOP_TIME")
    End If
    If chk_alarma_graf.Checked = True Then
        GRAFICA_TOT.Series.Add("T2_ALARM_TIME")
    End If
    If Chk_qnt_palets.Checked = True Then
        GRAFICA_TOT.Series.Add("T2_PALLETS_QUANTITY")
    End If
End If
```



```
        GRAFICA_TOT.Series("T2_PALLETS_QUANTITY").ChartType           =  
DataVisualization.Charting.SeriesChartType.Line  
    End If  
End If  
  
If Chk_graf_T3.Checked = True Then  
    If chk_graf_marxa.Checked = True Then  
        GRAFICA_TOT.Series.Add("T3_START_TIME")  
    End If  
    If chk_graf_paro.Checked = True Then  
        GRAFICA_TOT.Series.Add("T3_STOP_TIME")  
    End If  
    If chk_alarma_graf.Checked = True Then  
        GRAFICA_TOT.Series.Add("T3_ALARM_TIME")  
    End If  
    If Chk_qnt_palets.Checked = True Then  
        GRAFICA_TOT.Series.Add("T3_PALLETS_QUANTITY")  
        GRAFICA_TOT.Series("T3_PALLETS_QUANTITY").ChartType       =  
DataVisualization.Charting.SeriesChartType.Line  
    End If  
End If  
  
    If chk_graf_marxa.Checked = True Or chk_graf_paro.Checked = True Or  
chk_alarma_graf.Checked = True Then  
        Label119.Visible = True  
        Label120.Visible = True  
    Else  
        Label119.Visible = False  
        Label120.Visible = False  
    End If  
  
    If Chk_qnt_palets.Checked = True Then  
        Label118.Visible = True  
        Label121.Visible = True  
    Else  
        Label118.Visible = False  
        Label121.Visible = False  
    End If  
  
    For N = 0 To 24  
        If Chk_graf_TOT.Checked = True Then  
            If chk_graf_marxa.Checked = True Then
```

```
        TEMPS_VISUALITZAR(N) =
convert_temps_to_num(prod_dia_tmarxa_tot(N))

GRAFICA_TOT.Series("TOTAL_START_TIME").Points.Add(Math.Round(TEMPS_VISU
ALITZAR(N), 2))
    End If
    If chk_graf_paro.Checked = True Then
        TEMPS_VISUALITZAR(N) = convert_temps_to_num(prod_dia_tparo_tot(N))

GRAFICA_TOT.Series("TOTAL_STOP_TIME").Points.Add(Math.Round(TEMPS_VISUA
LITZAR(N), 2))
    End If
    If chk_alarma_graf.Checked = True Then
        TEMPS_VISUALITZAR(N) =
convert_temps_to_num(prod_dia_talarm_tot(N))

GRAFICA_TOT.Series("TOTAL_ALARM_TIME").Points.Add(Math.Round(TEMPS_VISU
ALITZAR(N), 2))
    End If
    If Chk_qnt_palets.Checked = True Then

GRAFICA_TOT.Series("TOTAL_PALLETS_QUANTITY").Points.Add(prod_dia_palets_tot
(N))
    End If
    End If

    If Chk_graf_T1.Checked = True Then
        If chk_graf_marxa.Checked = True Then
            TEMPS_VISUALITZAR(N) =
convert_temps_to_num(prod_dia_tmarxa_t1(N))

GRAFICA_TOT.Series("T1_START_TIME").Points.Add(Math.Round(TEMPS_VISUALIT
ZAR(N), 2))
        End If
        If chk_graf_paro.Checked = True Then
            TEMPS_VISUALITZAR(N) = convert_temps_to_num(prod_dia_tparo_t1(N))

GRAFICA_TOT.Series("T1_STOP_TIME").Points.Add(Math.Round(TEMPS_VISUALITZ
AR(N), 2))
        End If
        If chk_alarma_graf.Checked = True Then
            TEMPS_VISUALITZAR(N) = convert_temps_to_num(prod_dia_talarm_t1(N))
```

```
GRAFICA_TOT.Series("T1_ALARM_TIME").Points.Add(Math.Round(TEMPS_VISUALITZAR(N), 2))
```

```
End If
```

```
If Chk_qnt_palets.Checked = True Then
```

```
GRAFICA_TOT.Series("T1_PALLETS_QUANTITY").Points.Add(prod_dia_palets_t1(N))
```

```
End If
```

```
End If
```

```
If Chk_graf_T2.Checked = True Then
```

```
If chk_graf_marxa.Checked = True Then
```

```
TEMPS_VISUALITZAR(N) =
```

```
convert_temps_to_num(prod_dia_tmarxa_t2(N))
```

```
GRAFICA_TOT.Series("T2_START_TIME").Points.Add(Math.Round(TEMPS_VISUALITZAR(N), 2))
```

```
End If
```

```
If chk_graf_paro.Checked = True Then
```

```
TEMPS_VISUALITZAR(N) = convert_temps_to_num(prod_dia_tparo_t2(N))
```

```
GRAFICA_TOT.Series("T2_STOP_TIME").Points.Add(Math.Round(TEMPS_VISUALITZAR(N), 2))
```

```
End If
```

```
If chk_alarma_graf.Checked = True Then
```

```
TEMPS_VISUALITZAR(N) = convert_temps_to_num(prod_dia_talarm_t2(N))
```

```
GRAFICA_TOT.Series("T2_ALARM_TIME").Points.Add(Math.Round(TEMPS_VISUALITZAR(N), 2))
```

```
End If
```

```
If Chk_qnt_palets.Checked = True Then
```

```
GRAFICA_TOT.Series("T2_PALLETS_QUANTITY").Points.Add(prod_dia_palets_t2(N))
```

```
End If
```

```
End If
```

```
If Chk_graf_T3.Checked = True Then
```

```
If chk_graf_marxa.Checked = True Then
```

```
TEMPS_VISUALITZAR(N) =
```

```
convert_temps_to_num(prod_dia_tmarxa_t3(N))
```

```
GRAFICA_TOT.Series("T3_START_TIME").Points.Add(Math.Round(TEMPS_VISUALITZAR(N), 2))
    End If
    If chk_graf_paro.Checked = True Then
        TEMPS_VISUALITZAR(N) = convert_temps_to_num(prod_dia_tparo_t3(N))

GRAFICA_TOT.Series("T3_STOP_TIME").Points.Add(Math.Round(TEMPS_VISUALITZAR(N), 2))
    End If
    If chk_alarma_graf.Checked = True Then
        TEMPS_VISUALITZAR(N) = convert_temps_to_num(prod_dia_talarm_t3(N))

GRAFICA_TOT.Series("T3_ALARM_TIME").Points.Add(Math.Round(TEMPS_VISUALITZAR(N), 2))
    End If
    If Chk_qnt_palets.Checked = True Then

GRAFICA_TOT.Series("T3_PALLETS_QUANTITY").Points.Add(prod_dia_palets_t3(N))
    End If
    End If

    Next
    ZOOM_GRAFICA = 24

End Sub
```

3.7.- TIMER_ALARMES_TICK

```
Private Sub Timer_alarmes_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Timer_alarmes.Tick
    Cursor = Cursors.WaitCursor
    Dim adreça As Integer
    Dim bytes2(11) As Byte
    Dim adreça_H As Byte
    Dim adreça_L As Byte
    Dim PUNTER_REGISTRE_ALARMES As Integer
    Dim valor_alarma As Integer
    Dim valor_alarma2 As Integer
    Dim valor_alarma_binari As String
    Dim valor_temporal_alarma As Integer
    Dim valor_index_alarma As String
```

```
Dim valor_index_alarma2 As String
valor_index_alarma2 = ""
valor_alarma_binari = ""
comando_escritura = plc_lectura.GetStream
```

```
While comando_escritura.DataAvailable = True Or comando_lectura.DataAvailable
= True
```

```
    Wait(1)
End While
adreça = 3000
adreça_L = adreça Mod 256
adreça_H = adreça \ 256
bytes2(0) = 0
bytes2(1) = 0
bytes2(2) = 0
bytes2(3) = 0
bytes2(4) = 0
bytes2(5) = 6
bytes2(6) = 1
bytes2(7) = 4
bytes2(8) = adreça_H
bytes2(9) = adreça_L
bytes2(10) = 0
bytes2(11) = 50
```

```
comando_escritura.Write(bytes2, 0, bytes2.Length)
```

```
bytes2 = Nothing
If plc_lectura.Connected = True Then
    While comando_escritura.DataAvailable = False
        Wait(1)
    End While
    If comando_escritura.DataAvailable = True Then
        ReDim bytes2(plc_lectura.ReceiveBufferSize)
        comando_escritura.Read(bytes2, 0, bytes2.Length)
        llista_bytes_rebut = bytes2
    End If
```

```
    bytes2 = Nothing
End If
xlApp = New Excel.ApplicationClass
xlWorkBook = xlApp.Workbooks.Open("C:\CAPE\CAPE" + TextBox8.Text + ".xlsx")
```

```
xlWorkSheet = xlWorkBook.Worksheets("ALARMS")
For PUNTER_REGISTRE_ALARMES = 0 To 98 Step 2

    valor_alarma = (llista_bytes_rebut(PUNTER_REGISTRE_ALARMES + 9) * 256)
+ llista_bytes_rebut(PUNTER_REGISTRE_ALARMES + 10)

    valor_alarma2 = valor_alarma
    valor_alarma_binari = ""
    Do
        valor_temporal_alarma = valor_alarma2 Mod 2
        valor_alarma_binari = Str(valor_temporal_alarma) + valor_alarma_binari
        valor_alarma2 = valor_alarma2 \ 2
    Loop Until valor_alarma2 = 0

    For n = 1 To 31 Step 2
        valor_index_alarma = Strings.Right(valor_alarma_binari, n)
        If n >= 2 And valor_index_alarma <> "" Then
            valor_index_alarma = Strings.Left(valor_index_alarma, n - (n - 1))
        End If
        If valor_index_alarma <> "1" Then
            AUXILIAR(PUNTER_REGISTRE_ALARMES, n) = "0"
        End If
        If n >= 2 And valor_index_alarma <> "" Then
            valor_index_alarma = Strings.Left(valor_index_alarma, n - (n - 1))
        End If

        If valor_index_alarma = "1" Then
            If AUXILIAR(PUNTER_REGISTRE_ALARMES, n) <> 1 Then
                If n = 1 Then
                    descripcio_alarma = taula_excel(n, PUNTER_REGISTRE_ALARMES +
3)
                    AUXILIAR(PUNTER_REGISTRE_ALARMES, n) = 1
                End If
                If n > 1 Then
                    descripcio_alarma = taula_excel(n, PUNTER_REGISTRE_ALARMES +
3)
                    AUXILIAR(PUNTER_REGISTRE_ALARMES, n) = 1
                End If

                omplir_valors_taula_excel_ALARMES("ALARMS",
inici_casella_prod_alarms)
                inici_casella_prod_alarms = inici_casella_prod_alarms + 1
```

```
        End If
    End If

    Next
Next

xlWorkBook.Save()
xlWorkBook.Close()
xlApp.Quit()
releaseObject(xlApp)
releaseObject(xlWorkBook)
releaseObject(xlWorkSheet)
Cursor = Cursors.Arrow
End Sub
```

3.8.- CREAR_LLIBRE_EXCEL

```
Private Sub CrearLLibreEXcel(ByVal fileName As Object)

    Dim app As Excel.Application = Nothing
    Dim wb As Excel.Workbook = Nothing
    Dim st As Excel.Worksheet = Nothing

    Try
        app = New Excel.Application()
        wb = app.Workbooks.Add()
        st = wb.Worksheets.Add()
        st.Name = "T3"
        st = wb.Worksheets.Add()
        st.Name = "T2"
        st = wb.Worksheets.Add()
        st.Name = "T1"
        st = wb.Worksheets.Add()
        st.Name = "DAY"

        Dim dlgs As Excel.Dialogs = app.Dialogs
        Dim dlg As Excel.Dialog = dlgs(Excel.XIBuiltInDialog.xIDialogSaveAs)

        wb.SaveAs(fileName, missing, missing, missing, missing, missing, _
            Excel.XISaveAsAccessMode.xlNoChange, missing, missing, _
            missing, missing, missing)
        m_canceled = False ' NO ELIMINAR.
```

```
Catch ex As Exception
    m_canceled = True
Finally
End Try
```

```
wb.Close()
app.Quit()
app = Nothing
app = Nothing
releaseObject(wb)
releaseObject(app)
```

```
End Sub
```

3.9.-BUTO_EXCEL_CLICK

```
Private Sub BUTO_EXCEL_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles BUTO_EXCEL.Click
```

```
    Dim a As Integer
    Dim primera_cela_lliuere_excel As Integer
    Dim inici_dades As Integer
    inici_dades = 0
    primera_cela_lliuere_excel = 3
    Cursor = Cursors.WaitCursor
    CrearLlibreEXcel(txt_path.Text)
```

```
Try
    'crear objecte excel
    xlApp = New Excel.ApplicationClass
    xlWorkBook = xlApp.Workbooks.Open(txt_path.Text)
    xlWorkSheet = xlWorkBook.Worksheets("DAY")
    omplir_taula_excel("DAY")
```

```
    For n = inici_dades To 24
        If prod_dia_data(n) <> "00/00/0" Then
            xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 2) =  
prod_dia_data(n)
            xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 3) =  
prod_dia_palets_tot(n)
```


xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 4) =
prod_dia_tmarxa_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 5) =
prod_dia_tparo_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 6) =
prod_dia_talarm_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 7) =
prod_dia_mitja_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 8) =
prod_rcp_referencia_rcp1_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 9) =
prod_rcp_h_inici_rcp1_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 10) =
prod_rcp_num_pal_rcp1_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 11) =
prod_rcp_t_marxa_rcp1_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 12) =
prod_rcp_t_paro_rcp1_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 13) =
prod_rcp_t_alarm_rcp1_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 14) =
prod_rcp_mitja_rcp1_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 15) =
prod_rcp_referencia_rcp2_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 16) =
prod_rcp_h_inici_rcp2_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 17) =
prod_rcp_num_pal_rcp2_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 18) =
prod_rcp_t_marxa_rcp2_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 19) =
prod_rcp_t_paro_rcp2_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 20) =
prod_rcp_t_alarm_rcp2_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 21) =
prod_rcp_mitja_rcp2_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 22) =
prod_rcp_referencia_rcp3_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 23) =
prod_rcp_h_inici_rcp3_tot(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 24) =
prod_rcp_num_pal_rcp3_tot(n)

```
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 25) =  
prod_rcp_t_marxa_rcp3_tot(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 26) =  
prod_rcp_t_paro_rcp3_tot(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 27) =  
prod_rcp_t_alarm_rcp3_tot(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 28) =  
prod_rcp_mitja_rcp3_tot(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 29) =  
prod_rcp_referencia_rcp4_tot(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 30) =  
prod_rcp_h_inici_rcp4_tot(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 31) =  
prod_rcp_num_pal_rcp4_tot(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 32) =  
prod_rcp_t_marxa_rcp4_tot(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 33) =  
prod_rcp_t_paro_rcp4_tot(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 34) =  
prod_rcp_t_alarm_rcp4_tot(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 35) =  
prod_rcp_mitja_rcp4_tot(n)  
End If  
Next  
  
'omplir taula T1  
xlWorksheet = xlWorkbook.Worksheets("T1")  
omplir_taula_excel("T1")  
  
For n = inici_dades To 24  
If prod_dia_data(n) <> "00/00/0" Then  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 2) =  
prod_dia_data(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 3) =  
prod_dia_palets_t1(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 4) =  
prod_dia_tmarxa_t1(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 5) =  
prod_dia_tparo_t1(n)  
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 6) =  
prod_dia_talarm_t1(n)
```

xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 7) =
prod_dia_mitja_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 8) =
prod_rcp_referencia_rcp1_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 9) =
prod_rcp_h_inici_rcp1_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 10) =
prod_rcp_num_pal_rcp1_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 11) =
prod_rcp_t_marxa_rcp1_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 12) =
prod_rcp_t_paro_rcp1_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 13) =
prod_rcp_t_alarm_rcp1_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 14) =
prod_rcp_mitja_rcp1_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 15) =
prod_rcp_referencia_rcp2_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 16) =
prod_rcp_h_inici_rcp2_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 17) =
prod_rcp_num_pal_rcp2_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 18) =
prod_rcp_t_marxa_rcp2_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 19) =
prod_rcp_t_paro_rcp2_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 20) =
prod_rcp_t_alarm_rcp2_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 21) =
prod_rcp_mitja_rcp2_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 22) =
prod_rcp_referencia_rcp3_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 23) =
prod_rcp_h_inici_rcp3_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 24) =
prod_rcp_num_pal_rcp3_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 25) =
prod_rcp_t_marxa_rcp3_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 26) =
prod_rcp_t_paro_rcp3_t1(n)
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 27) =
prod_rcp_t_alarm_rcp3_t1(n)

```
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 28) =  
prod_rcp_mitja_rcp3_t1(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 29) =  
prod_rcp_referencia_rcp4_t1(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 30) =  
prod_rcp_h_inici_rcp4_t1(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 31) =  
prod_rcp_num_pal_rcp4_t1(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 32) =  
prod_rcp_t_marxa_rcp4_t1(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 33) =  
prod_rcp_t_paro_rcp4_t1(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 34) =  
prod_rcp_t_alarm_rcp4_t1(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 35) =  
prod_rcp_mitja_rcp4_t1(n)  
End If  
Next  
  
'omplir taula T2  
xlWorkSheet = xlWorkBook.Worksheets("T2")  
omplir_taula_excel("T2")  
  
For n = inici_dades To 24  
If prod_dia_data(n) <> "00/00/0" Then  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 2) =  
prod_dia_data(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 3) =  
prod_dia_palets_t2(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 4) =  
prod_dia_tmarxa_t2(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 5) =  
prod_dia_tparo_t2(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 6) =  
prod_dia_talarm_t2(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 7) =  
prod_dia_mitja_t2(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 8) =  
prod_rcp_referencia_rcp1_t2(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 9) =  
prod_rcp_h_inici_rcp1_t2(n)
```

xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 10) =
prod_rcp_num_pal_rcp1_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 11) =
prod_rcp_t_marxa_rcp1_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 12) =
prod_rcp_t_paro_rcp1_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 13) =
prod_rcp_t_alarm_rcp1_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 14) =
prod_rcp_mitja_rcp1_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 15) =
prod_rcp_referencia_rcp2_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 16) =
prod_rcp_h_inici_rcp2_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 17) =
prod_rcp_num_pal_rcp2_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 18) =
prod_rcp_t_marxa_rcp2_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 19) =
prod_rcp_t_paro_rcp2_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 20) =
prod_rcp_t_alarm_rcp2_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 21) =
prod_rcp_mitja_rcp2_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 22) =
prod_rcp_referencia_rcp3_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 23) =
prod_rcp_h_inici_rcp3_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 24) =
prod_rcp_num_pal_rcp3_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 25) =
prod_rcp_t_marxa_rcp3_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 26) =
prod_rcp_t_paro_rcp3_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 27) =
prod_rcp_t_alarm_rcp3_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 28) =
prod_rcp_mitja_rcp3_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 29) =
prod_rcp_referencia_rcp4_t2(n)
xlWorksheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 30) =
prod_rcp_h_inici_rcp4_t2(n)

```
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 31) =  
prod_rcp_num_pal_rcp4_t2(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 32) =  
prod_rcp_t_marxa_rcp4_t2(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 33) =  
prod_rcp_t_paro_rcp4_t2(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 34) =  
prod_rcp_t_alarm_rcp4_t2(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 35) =  
prod_rcp_mitja_rcp4_t2(n)  
End If  
Next  
'omplir taula T3  
xlWorkSheet = xlWorkBook.Worksheets("T3")  
omplir_taula_excel("T3")  
  
For n = inici_dades To 24  
If prod_dia_data(n) <> "00/00/0" Then  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 2) =  
prod_dia_data(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 3) =  
prod_dia_palets_t3(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 4) =  
prod_dia_tmarxa_t3(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 5) =  
prod_dia_tparo_t3(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 6) =  
prod_dia_talarm_t3(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 7) =  
prod_dia_mitja_t3(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 8) =  
prod_rcp_referencia_rcp1_t3(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 9) =  
prod_rcp_h_inici_rcp1_t3(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 10) =  
prod_rcp_num_pal_rcp1_t3(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 11) =  
prod_rcp_t_marxa_rcp1_t3(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 12) =  
prod_rcp_t_paro_rcp1_t3(n)  
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 13) =  
prod_rcp_t_alarm_rcp1_t3(n)
```

xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 14) =
prod_rcp_mitja_rcp1_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 15) =
prod_rcp_referencia_rcp2_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 16) =
prod_rcp_h_inici_rcp2_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 17) =
prod_rcp_num_pal_rcp2_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 18) =
prod_rcp_t_marxa_rcp2_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 19) =
prod_rcp_t_paro_rcp2_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 20) =
prod_rcp_t_alarm_rcp2_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 21) =
prod_rcp_mitja_rcp2_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 22) =
prod_rcp_referencia_rcp3_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 23) =
prod_rcp_h_inici_rcp3_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 24) =
prod_rcp_num_pal_rcp3_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 25) =
prod_rcp_t_marxa_rcp3_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 26) =
prod_rcp_t_paro_rcp3_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 27) =
prod_rcp_t_alarm_rcp3_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 28) =
prod_rcp_mitja_rcp3_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 29) =
prod_rcp_referencia_rcp4_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 30) =
prod_rcp_h_inici_rcp4_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 31) =
prod_rcp_num_pal_rcp4_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 32) =
prod_rcp_t_marxa_rcp4_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 33) =
prod_rcp_t_paro_rcp4_t3(n)
xlWorkSheet.Cells(primera_cela_lliuere_excel + (n - inici_dades), 34) =
prod_rcp_t_alarm_rcp4_t3(n)

```
xlWorkSheet.Cells(primera_cela_lliure_excel + (n - inici_dades), 35) =  
prod_rcp_mitja_rcp4_t3(n)  
End If  
Next
```

```
xlWorkBook.Close(True, missing, missing)  
xlApp.Quit()  
releaseObject(xlApp)  
releaseObject(xlWorkBook)  
releaseObject(xlWorkSheet)  
Catch ex As Exception  
a = 0  
End Try
```

```
Cursor = Cursors.Arrow
```

```
End Sub
```

```
Private Sub omplir_taula_excel(ByVal entrada As String)
```

```
xlWorkSheet.Range("A1").EntireColumn.ColumnWidth = 5  
xlWorkSheet.Range("B1").EntireColumn.ColumnWidth = 10  
xlWorkSheet.Range("C1").EntireColumn.ColumnWidth = 10  
xlWorkSheet.Range("D1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("E1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("F1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("G1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("H1").EntireColumn.ColumnWidth = 60  
xlWorkSheet.Range("I1").EntireColumn.ColumnWidth = 10  
xlWorkSheet.Range("J1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("K1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("L1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("M1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("N1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("O1").EntireColumn.ColumnWidth = 60  
xlWorkSheet.Range("P1").EntireColumn.ColumnWidth = 10  
xlWorkSheet.Range("Q1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("R1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("S1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("T1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("U1").EntireColumn.ColumnWidth = 7  
xlWorkSheet.Range("V1").EntireColumn.ColumnWidth = 60
```



```
xlWorkSheet.Range("W1").EntireColumn.ColumnWidth = 10
xlWorkSheet.Range("X1").EntireColumn.ColumnWidth = 7
xlWorkSheet.Range("Y1").EntireColumn.ColumnWidth = 7
xlWorkSheet.Range("Z1").EntireColumn.ColumnWidth = 7
xlWorkSheet.Range("AA1").EntireColumn.ColumnWidth = 7
xlWorkSheet.Range("AB1").EntireColumn.ColumnWidth = 7
xlWorkSheet.Range("AC1").EntireColumn.ColumnWidth = 60
xlWorkSheet.Range("AD1").EntireColumn.ColumnWidth = 10
xlWorkSheet.Range("AE1").EntireColumn.ColumnWidth = 7
xlWorkSheet.Range("AF1").EntireColumn.ColumnWidth = 7
xlWorkSheet.Range("AG1").EntireColumn.ColumnWidth = 7
xlWorkSheet.Range("AH1").EntireColumn.ColumnWidth = 7
xlWorkSheet.Range("AI1").EntireColumn.ColumnWidth = 7
xlWorkSheet.Range("B2:G2").Interior.Color = RGB(251, 252, 205)
xlWorkSheet.Range("O2:U2").Interior.Color = RGB(251, 252, 205)
xlWorkSheet.Range("AC2:AI2").Interior.Color = RGB(251, 252, 205)
xlWorkSheet.Range("H2:N2").Interior.Color = RGB(164, 236, 66)
xlWorkSheet.Range("V2:AB2").Interior.Color = RGB(164, 236, 66)
```

```
xlWorkSheet.Cells(2, 2) = "DATE"
xlWorkSheet.Cells(2, 3) = "PALLETS.Q"
xlWorkSheet.Cells(2, 4) = "START"
xlWorkSheet.Cells(2, 5) = "STOP"
xlWorkSheet.Cells(2, 6) = "ALARM"
xlWorkSheet.Cells(2, 7) = "AVERAGE"
xlWorkSheet.Cells(2, 8) = "REFERENCE1"
xlWorkSheet.Cells(2, 9) = "INIT.TIME"
xlWorkSheet.Cells(2, 10) = "PALLETS.Q"
xlWorkSheet.Cells(2, 11) = "START"
xlWorkSheet.Cells(2, 12) = "STOP"
xlWorkSheet.Cells(2, 13) = "ALARM"
xlWorkSheet.Cells(2, 14) = "AVERAGE"
xlWorkSheet.Cells(2, 15) = "REFERENCE2"
xlWorkSheet.Cells(2, 16) = "INIT.TIME"
xlWorkSheet.Cells(2, 17) = "PALLETS.Q"
xlWorkSheet.Cells(2, 18) = "START"
xlWorkSheet.Cells(2, 19) = "STOP"
xlWorkSheet.Cells(2, 20) = "ALARM"
xlWorkSheet.Cells(2, 21) = "AVERAGE"
xlWorkSheet.Cells(2, 22) = "REFERENCE3"
xlWorkSheet.Cells(2, 23) = "INIT.TIME"
xlWorkSheet.Cells(2, 24) = "PALLETS.Q"
```

```
xlWorkSheet.Cells(2, 25) = "START"  
xlWorkSheet.Cells(2, 26) = "STOP"  
xlWorkSheet.Cells(2, 27) = "ALARM"  
xlWorkSheet.Cells(2, 28) = "AVERAGE"  
xlWorkSheet.Cells(2, 29) = "REFERENCE4"  
xlWorkSheet.Cells(2, 30) = "INIT.TIME"  
xlWorkSheet.Cells(2, 31) = "PALLETS.Q"  
xlWorkSheet.Cells(2, 32) = "START"  
xlWorkSheet.Cells(2, 33) = "STOP"  
xlWorkSheet.Cells(2, 34) = "ALARM"  
xlWorkSheet.Cells(2, 35) = "AVERAGE"
```

End Sub