



**Facultad de Ingeniería Eléctrica
Departamento de Control Automático**

**Sistema de automatización basado en control de eventos discretos
para la etapa de empaquetado de la fábrica de producción de
bloques de hormigón POYATOS**

(Tesis en opción al título de Máster en Ciencias)

Autora

Ing. Liday Valenciano Hernández

Tutor: Dr. Israel Benítez Pina

Tutor: Dr. Israel Mazaira Morales

Consultante: MSc. Juan Carlos Sánchez Perera

**Santiago de Cuba
2012**



**Facultad de Ingeniería Eléctrica
Departamento de Control Automático**

**Sistema de automatización basado en control de eventos discretos
para la etapa de empaquetado de la fábrica de producción de
bloques de hormigón POYATOS**

(Tesis en opción al título de Máster en Ciencias)

Autora

Ing. Liday Valenciano Hernández

Tutor: Dr. Israel Benítez Pina

Tutor: Dr. Israel Mazaira Morales

Consultante: MSc. Juan Carlos Sánchez Perera

**Santiago de Cuba
2012**

Dedicatoria

A mi familia, en especial a mis padres.

Agradecimientos

A mis tutores Israel Benitez e Israel Mazaira, por toda su atención y ayuda.

A Juan Carlos por hacer posible la realización de este proyecto.

A todos los amigos que de alguna manera contribuyeron a la realización de este trabajo.

A todos mis compañeros del Departamento de Control Automático, por la ayuda brindada a lo largo de la maestría.

Al personal de la fábrica de bloques perteneciente a la Empresa de Materiales de la Construcción de Santiago de Cuba.

A todos, Muchas Gracias.

Resumen

El presente proyecto, se enmarca en el ámbito de la automatización de procesos industriales basada en el control de eventos discretos. Específicamente, utilizando como metodología formal para el diseño de la automatización de procesos el modelado matemático basado en Máquinas de Estados Finitos Deterministas, que sirve de base a los diagramas GRAFCET utilizados para la representación de los algoritmos y su traducción a la programación de PLC en lenguaje ST. Para completar la automatización integral, se realiza un estudio de los sistemas SCADA como mecanismo de supervisión, gestión y control remoto de procesos, así como de las comunicaciones industriales sobre bus de campo abierto Modbus y servidores OPC para establecer el intercambio de información entre dispositivos.

El caso concreto de estudio se centra en el análisis y desarrollo de un sistema de automatización para mejorar la eficiencia del proceso de empaquetamiento de la producción de bloques de hormigón de la planta tipo POYATOS, perteneciente a la Empresa de Materiales de la Construcción de Santiago de Cuba. Para ello, se realiza un estudio riguroso del problema y se plantea la solución del mismo, ajustándose a las características y condiciones específicas de la planta. Teniendo en cuenta este punto, se evalúan las características principales de todo el equipamiento necesario para la solución. Se justifican además las características y ventajas de la metodología utilizada estableciendo con esto las bases para hacerla extensiva al resto de procesos de la planta.

Palabras Claves: *automatización industrial, empaquetado de bloques de hormigón, máquinas de estados, sistemas de eventos discretos*

Abstract

The present project, it is framed within the scope of industrial process automation based on discrete event control. Specifically using as a formal methodology in description of process, the mathematical model based on Deterministic Finite State Machines, in conjunction with the GRAFCET diagrams, used to represent algorithms and programming of PLC. As a consequence, it is performed a study of SCADA systems as a mechanism for monitoring, management and remote control of processes as well as industrial communications on open Modbus fieldbus and OPC servers to establish the exchange of information between devices.

The case study focuses on the analysis and development of an automation system for process control of concrete block production to POYATOS type plant, belonging to the Company of Building Materials in Santiago de Cuba. To do this, there has been a rigorous study of the problem and the solution was raised from the same adjustment to the characteristics and specific conditions of the plant. Considering this point, it has assessed the main characteristics of all the equipment necessary for the solution. It also justifies the characteristics and advantages of the methodology studied with this by setting the basis for extending it to other plant processes.

Keywords: *industrial automation, packaging concrete block, state machines, discrete event systems*

Índice

Introducción	1
Problema.....	2
Objeto	2
Objetivo.....	2
Objetivos específicos.....	3
Campo de Acción.....	3
Hipótesis	3
Tareas de la Investigación.....	4
Técnicas y métodos empleados en la investigación	4
Aportes de la tesis.....	4
Estructura de la tesis.....	5
Capítulo I	6
1.1 Introducción.....	6
1.2 PLC y Variadores de frecuencia en la industria	6
1.2.1 PLC.....	6
1.2.2 Variadores de frecuencia y Arrancadores	7
1.3 Automatización de sistemas industriales	9
1.3.1 Etapa de diseño.....	10
1.3.1.1 Modelado de Sistemas de eventos Discretos usando Máquinas de Estados Finitos Deterministas.....	12
1.3.1.2 Aspectos de construcción de las Máquinas de Estados Finitos Deterministas	16
1.3.1.2.1 Deadlock	17
1.3.1.2.2 Acciones.....	17
1.3.1.2.3 Clasificación de las Máquinas de Estados Finitos Deterministas	17
1.3.1.3 Ventajas del modelado en Máquinas de Estados Finitos Deterministas	22
1.3.1.4 GRAFica de Control de Etapas de Transición (GRAFCET)	23
1.3.2. Programación y simulación	26
1.4 Sistemas de supervisión.....	27
1.4.1 Sistemas SCADA.....	28
1.4.2 Servidor para la gestión de la comunicación (OPC)	29
1.4.2.1 Tipos de servidores OPC.....	30
1.4.3 Comunicación industrial	30
1.5 Descripción del flujo productivo de bloques de hormigón de la planta POYATOS	31
1.5.1. Dosificación	32
1.5.2. Prensado	33
1.5.3. Transportación y Curado.....	33
1.5.4. Empaquetado	34
1.6 Descripción del proceso de empaquetado de bloques.....	34
1.7 Conclusiones.....	37
Capítulo II	38

2.1	Introducción.....	38
2.2	Etapa de diseño para la automatización del proceso de empaquetado de bloques	38
2.2.1	Traslación de los bloques que serán empaquetados	38
2.2.2	Empaquetado de bloques	39
2.2.3	Traslación del paquete de bloques para su recogida	43
2.2.4	Sistema de automatización para el control del empaquetado de bloques	43
2.2.5	Funcionamiento del sistema automatizado	47
2.2.6	Modelado y diseño de la automatización del proceso de empaquetado de bloques	50
2.2.6.1	Modelado del modo Manual	51
2.2.6.2	Modelado del modo Automático	52
2.2.6.2.1	Modelado de la cinta transportadora de bloques para empaquetar	53
2.2.6.2.2	Modelado del manipulador secuencial	55
2.2.6.2.3	Modelado de la cinta transportadora de paquetes de bloques	60
2.3	Algoritmo y programación del Autómata Industrial.....	62
2.3.1	Algoritmo para el control de la cinta transportadora de bloques para empaquetar	62
2.3.2	Algoritmo para el control del manipulador secuencial	63
2.3.3	Algoritmo para el control de la cinta transportadora encargada del traslado del paquete de bloques	66
2.3.4	Programación del PLC Industrial.....	66
2.3.4.1	Programación del movimiento de las cintas transportadoras	68
2.3.4.2	Programación del movimiento del manipulador	69
2.4	Sistema de supervisión para el proceso de empaquetado de bloques	73
2.4.1	Niveles de acceso.....	75
2.4.2	Gestión de Alarmas	76
2.4.3	Mímico del proceso	77
2.4.4	Comunicación para la supervisión del proceso	79
2.4.4.1	Principio de las comunicaciones industriales	80
2.4.5	Servidor Matrikon-OPC para la gestión de la comunicación.....	82
2.5	Validación del sistema automatizado.....	84
2.6	Conclusiones.....	87
	Conclusiones	89
	Recomendaciones	90
	Referencias	91
	Anexos.....	96
	Anexo 1: Ventajas de los Variadores de Velocidad	96
	Anexo 2: Simbología y reglas del GRAFCET	98
	Anexo 3: Funciones básicas de un sistema de supervisión y adquisición de datos	100
	Anexo 4: Esquema de conexión del variador de frecuencia Altivar 28.....	102
	Anexo 5: Esquema de conexión del arrancador Altistart 01.....	103
	Anexo 6: Esquema eléctrico para el control de tres motores por un variador	105
	Anexo 7: Programa en ST para el control del proceso de empaquetado de bloques en plantas POYATOS	106

Introducción

El creciente auge y la gran complejidad de los sistemas de automatización actuales en los más diversos campos, y en particular en la industria, vienen aparejados a la necesidad de obtener resultados cada vez más rápidos y eficientes, lo cual estimula el hecho de investigar y desarrollar metodologías, técnicas y herramientas orientados a la solución de este tema [1].

El control de sistemas de eventos discretos [2], [3] incluye técnicas de modelado dinámico de sistemas, que se utilizan con el objetivo de diseñar sistemas de elevada complejidad en diferentes sectores como: robótica, sistemas de tráfico, sistemas de manufactura flexible, inteligencia artificial, programación de componentes electrónicos, sistemas de tiempo real, lenguajes formales y técnicas de compilación, entre otros. Se caracteriza por un control en función de la planificación de ocurrencia de eventos en instantes concretos de tiempo [3], [4]. La ejecución de un evento puede desencadenar la generación de nuevos eventos futuros. Un proceso de eventos discretos es definido en términos algebraicos y su comportamiento es dado por un lenguaje formal apropiado [4], [5]. Un sistema complejo puede consistir en un conjunto de procesos asíncronos que interactúan y operan concurrentemente. Un problema de control para tales sistemas es la supervisión de varios componentes para asegurar su interacción armoniosa y un flujo ordenado de eventos resultantes [4], [5].

En este sentido, el presente trabajo de tesis de máster se centra en la investigación de técnicas de control de eventos discretos y en particular en el modelo matemático algebraico de Máquinas de Estados Finitos Deterministas en conjunto con los diagramas GRAFCET para el modelado formal de sistemas de automatización de procesos industriales. Específicamente en la industria de la construcción, se demuestra la efectividad de este enfoque a través del análisis y desarrollo de un sistema de automatización, para optimizar el proceso de empaquetado de bloques de hormigón de la planta de fabricación de la firma española POYATOS del complejo Los Guao, perteneciente a la Empresa de Materiales de la Construcción de Santiago de Cuba, disminuyendo al máximo los tiempos empleados y respetando la calidad de la producción. A la par, se abordan temas relacionados con los sistemas de supervisión para la gestión y control a distancia del sistema automatizado y como consecuencia el estudio de las comunicaciones industriales.

El sector en el que se enmarca el caso de estudio constituye, sin dudas, uno de los más prioritarios en nuestro país, la necesidad y la demanda de construcción marca todos y cada uno de los restantes sectores de la sociedad. Actualmente, en la fabricación de bloques en la planta existen fases que se realizan completa o parcialmente de forma manual. Además, en algunas de estas fases parcialmente automatizadas, el paso del tiempo ha provocado el deterioro y la obsolescencia de los sistemas, provocando prolongados tiempos de fabricación y gran esfuerzo para la obtención de un volumen de este producto que no satisface la creciente demanda. Debido a esto, la automatización en cualquier etapa de producción redundará en un aumento considerable de la producción y calidad, y con ello una mayor eficiencia del proceso.

Por tales motivos en el diseño de esta investigación se define como:

Problema

Baja eficiencia y calidad de la producción por la carencia de un sistema de automatización fundamentado en técnicas de control que garantice la eficiencia en cada uno de los procesos de la planta, debido fundamentalmente al funcionamiento manual del proceso de empaquetado de bloques de hormigón, lo que trae consigo elevados índices de peligrosidad desde el punto de vista del personal operario, y como es lógico el inevitable hecho de que se cometan errores humanos.

Objeto

Automatización de procesos industriales basada en técnicas de control de eventos discretos.

Objetivo

Diseñar un sistema de automatización completo para elevar la eficiencia del control del proceso de empaquetado de bloques de hormigón, aplicando técnicas de control de eventos discretos, concretamente Máquinas de Estados Finitos Deterministas (MEFD).

Objetivos específicos

- Analizar el estado del arte de la automatización industrial y el uso en su diseño de las técnicas de control de eventos discretos basadas en las MEFD y el GRAFCET.
- Analizar las características del proceso de empaquetado en conjunto con las necesidades de la planta, y seleccionar el equipamiento que será utilizado en el sistema de automatización, teniendo en cuenta para ello, las ventajas en cuanto a características técnicas y su coste económico acorde a las necesidades de la empresa.
- Aplicar el modelado formal basado en MEFD de la secuencia de automatización del proceso, como base para desarrollar un algoritmo de control utilizando los diagramas GRAFCET y su traducción a programas ST (Lenguaje Texto Estructurado) que será implementado en el PLC seleccionado.
- Analizar las potencialidades del modelado MEFD para su posterior utilización en los demás procesos de la planta.
- Desarrollar un sistema SCADA para supervisión y control remoto de las variables del proceso utilizando para ello herramientas de actualidad.

Campo de Acción

Técnicas de control de eventos discretos y modelado de procesos basado en Máquinas de Estados Finitos Deterministas.

Hipótesis

Si se desarrolla un sistema de automatización eficiente aplicando técnicas de control de eventos discretos para el control y gestión del proceso de empaquetado de bloques, se aumentará la producción de bloques empaquetados con la disminución del tiempo empleado para ello, disminuyendo también los niveles de peligrosidad y la generación de errores humanos, garantizando con ello una mejora considerable en la eficiencia y calidad de la producción.

Tareas de la Investigación

1. Investigación de los aspectos teórico-prácticos y del estado del arte, referentes al diseño de sistemas de automatización mediante el modelado de procesos de eventos discretos a través de máquinas de estados, así como el estado del arte de los sistemas de supervisión y el uso de PLC y variadores de frecuencia en la industrias.
2. Análisis del funcionamiento del proceso de producción de bloques de hormigón en plantas POYATOS y definición de las principales deficiencias que repercuten negativamente en la producción.
3. Análisis del hardware y software a emplear de acuerdo a las necesidades del proceso a automatizar.
4. Elaboración del diseño de un sistema de automatización eficiente para el proceso de empaquetado de bloques de hormigón, utilizando las MEFD para su modelado, su conversión en un algoritmo en GRAFCET para el control del proceso y su traducción a la programación del PLC verificado mediante simulación.
5. Elaboración de un SCADA para la supervisión del proceso.
6. Validación del cumplimiento de los requerimientos funcionales del sistema.
7. Documentación del trabajo.

Técnicas y métodos empleados en la investigación

1. Técnicas empíricas: Observación, Entrevistas.
2. Análisis de fuentes documentales.
3. Método histórico-lógico.
4. Método de análisis y síntesis.
5. Métodos experimentales: Diseño y Simulación.

Aportes de la tesis

Sistema de automatización integrado (local y SCADA) para el control del proceso de empaquetado de bloques de la planta, basado en el modelado de Máquinas de Estados Finitos Deterministas, garantizando la productividad, calidad y eficiencia en cuanto a tiempo, energía, seguridad y organización del sistema, lo cual se revierte en beneficio económico.

Estructura de la tesis

El informe del proyecto de tesis está estructurado de la siguiente forma:

1. *Capítulo 1:* Está dedicado a presentar el marco teórico y estado del arte que relaciona los diferentes aspectos que se deben tener en cuenta para la automatización del proceso. Se introducen los conceptos fundamentales de las técnicas de control de eventos discretos y del modelo matemático de MEFD y diagramas GRAFCET, como metodología de diseño para el control eficiente de los procesos industriales. Se introducen los conceptos fundamentales de los sistemas SCADA y las comunicaciones industriales. Se particulariza esto para el caso bajo estudio que es el flujo tecnológico del proceso de construcción de bloques de hormigón.
2. *Capítulo 2:* Está dedicado al diseño y desarrollo del sistema de automatización del proceso. Se describe formalmente el modelo de la automatización del proceso de empaquetado de bloques basado en MEFD enlazados con el diseño de los lazos de control, y a partir de dicho modelo se crean los diagramas GRAFCET que representan el algoritmo para el control de dicho proceso. Se presenta, además, el diseño y desarrollo del sistema SCADA para la supervisión, gestión y control remoto del proceso, utilizando un servidor OPC y el protocolo de comunicación industrial Modbus. Se analiza la eficiencia que proporciona el sistema automatizado con respecto al sistema actual.
3. *Conclusiones:* Se describe el cumplimiento de cada uno de los objetivos planteados, y se comentan las bases sentadas para el posterior desarrollo de las tareas de automatización de la planta en general utilizando la metodología presentada en este trabajo.

Capítulo I

1.1 Introducción

En este capítulo se presenta de forma breve el estado del arte del uso de los PLC en la industria y de los variadores de frecuencia para el control de motores. Para el diseño del sistema de automatización se analizan las técnicas de control de eventos discretos utilizando el modelado de sistemas basado en MEFD. Se analizan además los diagramas GRAFCET, como caso particular de las MEFD, utilizados para la descripción y representación de algoritmos y para la programación de PLC. Para completar la automatización hasta el nivel jerárquico superior, se hace una breve referencia a los sistemas de adquisición de datos, supervisión y control, explotando sus potencialidades mediante las comunicaciones industriales y los servidores OPC. Finalmente, se particulariza en el caso bajo estudio a partir del flujo tecnológico de la planta de construcción de bloques de hormigón POYATOS, enfatizando en el proceso de empaquetado de bloques.

1.2 PLC y Variadores de frecuencia en la industria

Los PLC son ampliamente usados en la industria por las características y ventajas que proporcionan en el control de procesos. En la más reciente encuesta realizada por la revista Control Engineering en conjunto con el centro Reed Research, mostró que el 57% de los encuestados usan PLC para aplicaciones en planta y un 22% usa PC. El 21% restante usa ambos [6]. Por otra parte el uso de los variadores de frecuencia se incrementa cada vez más, debido principalmente, al control eficiente que realizan y por la cantidad de máquinas que existen en las industrias [7].

1.2.1 PLC

El continuo avance de las ciencias y el desarrollo tecnológico en el campo industrial se acelera cada día más. El control, si se ha realizado de forma correcta en cuanto a análisis, modelado, programación, selección y uso racional de los elementos y la seguridad, es de vital importancia en todo proceso, ya que garantiza el correcto funcionamiento del mismo de

manera eficiente. Es por ello que los PLC se han introducido en todos los niveles de automatización industrial, siendo cada día más sofisticados [1], [8]. Una investigación reciente realizada sobre los PLC [6], concluye que los PLC son usados en un 82% en el control de máquinas en la industria, siendo esta la aplicación más popular en la que se utilizan. En el control de procesos son usados en un 74% y en el control de movimiento un 55%. Existen otras aplicaciones en las que se utilizan pero en menor cuantía, tales como, el control de lotes en un 31% y en el diagnóstico de aplicaciones en un 25%. El menor uso de PLC se reporta en las aplicaciones de seguridad en solo un 1%.

Los PLC tienen la gran ventaja de que el mismo controlador básico puede ser usado para el control de un amplio rango de sistemas, dando como resultado sistemas flexibles que pueden ser utilizados en sistemas de control que varían completamente su naturaleza y complejidad [9], [10], [11].

Independientemente de las ventajas que presenten los PLC en el control de procesos en la industria, la programación de forma correcta para el control no es posible sin antes realizar un modelado de la planta¹ a controlar [12], [13]. Esto se debe a que se pueden cometer errores en la programación o simplemente puede que no se tengan en cuenta especificaciones del funcionamiento y/o de la estructura. Es por ello que el modelado de la planta es necesario para crear el diseño formal del sistema automatizado [14], y a partir de este se realiza la programación del PLC para un funcionamiento correcto de la misma.

1.2.2 Variadores de frecuencia y Arrancadores

El control de motores es la esencia de la automatización industrial. Mientras que los avances tecnológicos modifican la arquitectura de los sistemas, el control de motores ha llegado a ser un elemento de creciente importancia como parte de los sistemas industriales. La tendencia a un costo menor, mayor funcionalidad y más inteligencia entre los dispositivos significa un mosaico de opciones aún más amplio para que los clientes satisfagan las necesidades de una aplicación específica [15].

¹ K. Ogata, "Modern control engineering". Editorial Felix Varela, Tomo I, Pág. 2

Al comprender los requisitos de la aplicación e implementar el nivel adecuado en el control de motores, se pueden predecir, evitar y remediar los potenciales efectos adversos de los actuales motores en todas las aplicaciones. Las nuevas y avanzadas tecnologías en la protección y control de motores ofrecen un número de beneficios significativos: productividad mejorada, protección de aplicaciones y del equipamiento, y una mejora en la seguridad del personal.

Actualmente los variadores de frecuencia desempeñan un lugar preponderante dentro de la industria en general [7], por su flexibilidad y multiplicidad en opciones de utilización a un costo muy razonable. Muchas son las aplicaciones en donde la variación de velocidad se ha convertido en la vida misma de la fabricación de cualquier tipo de productos. Bien puede ser para arranques y paros frecuentes, cargas de alta inercia, troquelado, extrusión, bombeo, ventilación, coordinación de movimientos en líneas de producción, uso en regulación de velocidad, posicionamiento, sincronización, e infinidad de otras aplicaciones que antes solo era posible hacerlas parcialmente con corriente directa usando motores y controles costosos de adquirir y de mantener [7].

Un variador de frecuencia es un sistema para el control de la velocidad rotacional de un motor de corriente alterna, por medio del control de la frecuencia de alimentación suministrada al motor [9], [16], [17], [18]. Generalmente son usados para el control de motores de inducción trifásicos. Esto se debe a la potencia necesaria a emplear en las instalaciones industriales para mover grandes volúmenes de carga, siendo los variadores de frecuencia los más apropiados para la mayoría de las aplicaciones, y constituyen por lo general la elección más económica. Las diferentes ventajas de los variadores se muestran en el anexo 1.

Los arrancadores son dispositivos más sencillos que los variadores, pero de gran aplicación en la industria. Son utilizados principalmente para el control de los motores cuando es necesario reducir los pares de arranque, acelerar, desacelerar o frenar suavemente, para la seguridad de las personas u objetos transportados, y arrancar máquinas progresivamente, en especial aquellas de fuerte inercia [16], [19]. Los arrancadores tienen gran aplicación en sistemas de bombeo, compresores, transportes horizontales, ventiladores y centrífugas.

Una ventaja muy importante en el uso de ambos dispositivos, es el mejor aprovechamiento de la energía. Por tanto un equipo accionado mediante un variador de velocidad o un arrancador, emplea generalmente menor energía que si dicho equipo fuera activado a una velocidad fija sin el uso de ninguno de estos dispositivos [16].

1.3 Automatización de sistemas industriales

La automatización de sistemas industriales, tiene una fuerte componente definida por la coordinación de las actividades de los diferentes subsistemas que se integran en las redes de control distribuido industrial que se utilizan actualmente. La coordinación de esas actividades se torna más compleja en la medida que se incrementan las posibilidades de desarrollar diferentes producciones en el mismo sistema productivo. La industria moderna tiene un desarrollo orientado a los sistemas flexibles de producción conocidos como Sistemas de Manufactura Flexible. Eso garantiza el nivel de atención al cliente que requiere de un comercio electrónico competitivo moderno. Por tanto, la proyección de sistemas de control industrial incrementa su complejidad y necesita de métodos formales para el modelado de estos sistemas [20].

Las etapas de desarrollo de cualquier automatización moderna incluyen [21]:

1. *Diseño*: Donde se requiere hacer el análisis y síntesis del sistema automatizado. Esto es solo posible con el uso del Modelado Formal de la planta industrial, sus especificaciones de seguridad y comportamiento como sistema controlado.
2. *Programación*: Los modelos deben ser programados en lenguajes normalizados de automatización con facilidades de simulación de su funcionamiento. Como los PLC son los dispositivos de mayor uso, la IEC61131 es la norma más usada, así como sus ambientes de edición y simulación.
3. *Implementación*: En la automatización integrada actual se requiere crear estructuras jerárquicas en redes buscando eficiencia y facilidades de ampliación, modificación y mantenimiento para reducir tiempo y costos en el proyecto y luego en el funcionamiento de la empresa.

Esta última etapa no forma parte de los objetivos inmediatos de este trabajo de tesis y se llevará a cabo en una etapa siguiente. Esto se debe a que forma parte de un trabajo posterior

mucho más ambicioso, en el que se pretende realizar la automatización general de toda la planta, siguiendo la misma filosofía en la programación y selección del equipamiento de automatización. Es por ello, que los puntos tratados en este trabajo se centran fundamentalmente en la automatización de la etapa de empaquetado como punto de partida para el resto de las etapas.

1.3.1 Etapa de diseño

El diseño eficiente de la coordinación de actividades de los sistemas de automatización integrada modernos no puede realizarse dentro de la teoría clásica de control de variables continuas. Los modelos utilizados en Sistemas de Control de Variables Continuas no son aplicables en los Sistemas de Control de Eventos Discretos. Por tanto, existen otros métodos formales para Sistemas de Eventos Discretos (SED).

Los Sistemas de Eventos Discretos son sistemas dinámicos cuyo cambio de estado ocurre en puntos discretos del tiempo, con la ocurrencia de eventos aislados, como por ejemplo, comandos para operación de máquinas o señales de activación de sensores. Tienen su aplicación práctica en las áreas de redes de comunicación, robótica, sistemas de tráfico, sistemas de manufactura flexible, inteligencia artificial, y muchos otros [3], [4].

El uso de los métodos formales para el diseño de sistemas de automatización presenta numerosas facilidades, las cuales se mencionan a continuación:

- Facilidad para el análisis de las propiedades principales de los Sistemas de Control de procesos continuos y SED integrantes de cualquier Sistema de Automatización.
- Menos costo en la validación del sistema en frente de situaciones normales y de situaciones peligrosas.
- Reutilización del diseño.
- Repetitividad del diseño sobre cualquier plataforma de implementación.
- Facilidad de análisis de sistemas complejos y de gran tamaño.
- Buena precisión del proceso de modelado.
- Facilidad de documentación.

Entre los Modelos Lógicos orientados al análisis del comportamiento secuencial del sistema de eventos discretos se encuentran:

- Redes de Petri.
- Autómatas Finitos.
- Lógica Temporal.
- Sistemas Condición-Evento.

Los SED pueden ser usados como modelo para muchos procesos discretos o como estimados convenientes de la función de un sistema real. El trabajo de Ramadge y Wonham (R&W) [5], introduce un paradigma para el estudio del control de tales sistemas [2]. Este trabajo se desarrolla basándose en los Autómatas Finitos Deterministas (AFD). El modelo de planta de R&W se obtiene mediante una descripción de los procesos de la planta mediante autómatas finitos deterministas cuyo alfabeto está formado por un conjunto (finito) de eventos.

Las especificaciones del sistema se describen en términos del lenguaje formal generado por el modelo. El controlador se construye a partir de un supervisor, que es a su vez un AFD, con un objetivo específico. Esto permite solucionar el problema de estados y cadenas prohibidas [22], [23]. Un supervisor de un SED es diseñado cuando se está seguro del comportamiento deseado del sistema. Este método para el control probó ser muy conveniente para el propósito de investigación de SED y ha generado muchas extensiones. Una de esas extensiones ha sido el control modular jerárquico para SED, que se basa en la agregación de estados para un control modular clásico, lo que provee la descomposición de los problemas, quedando niveles de jerarquía, donde dichos niveles están relacionados, y en cada uno de estos niveles se resuelve un problema en específico. Esta técnica, propuesta por varios autores tales como Torrico y Cury [24], es ampliamente aplicada en sistemas muy complejos o de gran magnitud.

Otra de las extensiones del trabajo presentado por R&W inicialmente, es el propuesto por Silveira y Combacau [25], donde se propone un método de control para distribuir modelos de control centralizados de SED, para obtener una representación modular del proceso de producción. Una de las ventajas de este método es que se preservan las propiedades del modelo principal y que el conocimiento adquirido durante el proceso de modelado es aún

valioso para la nueva estructura. Otro aspecto importante es que el modelo distribuido (submodelos) puede ser modificado para incrementar la flexibilidad y alcanzar una mayor representación del comportamiento real.

En resumen, según el trabajo de R&W, un controlador puede ser sintetizado a partir de las especificaciones que requiere un sistema determinado con el objetivo de evitar algunos estados prohibidos específicos, o algunas secuencias específicas de eventos (cadenas). Las extensiones coinciden en el modelado modular. Por tanto se utilizará en esta tesis la síntesis de controladores mediante modelado modular de SED en autómatas finitos.

1.3.1.1 Modelado de Sistemas de eventos Discretos usando Máquinas de Estados Finitos Deterministas

Para la mayoría de los casos en modelos de ingeniería es necesario solamente considerar los aspectos del sistema que afectan su estudio. Por otro lado, el modelo debe ser suficientemente detallado para permitir conclusiones válidas sobre el sistema real.

Las principales ventajas de utilizar AFD como herramientas de modelado de un SED las brinda su fundamentación matemática y su representación compacta del comportamiento del sistema, por lo que es preciso modelar los estados que son importantes para la correcta secuencia lógica del sistema. Su naturaleza gráfica ayuda en la comprensión de sistemas complejos. Por todo ello, los AFD resultan una opción útil para describir y estudiar sistemas que se caracterizan por ser concurrentes, asincrónicos, distribuidos o paralelos.

Desde el punto de vista del control del sistema podemos tener diferentes tipos de eventos discretos en los SED:

Sea E el conjunto de todos los eventos que afectan nuestro SED, entonces se puede tener [3]:

- *Eventos controlables* ($E_c \subseteq E$): Son aquellos que pueden ser activados o desactivados por agentes externos (pueden ser autorizados o no por el control). Por ejemplo: apretar una tecla, conectar un equipamiento, enviar un paquete de mensajes, etc.

- *Eventos no controlables* ($E_u \subseteq E$): Son aquellos que no pueden ser evitados de ocurrir y por eso son considerados permanentemente habilitados (el control no tiene acciones sobre él). Por ejemplo: fin de tarea operatoria sobre una pieza, fallos espontáneos de equipamientos, pérdida de paquetes de mensajes, etc.
- *Eventos observables* ($E_o \subseteq E$): Son aquellos en que puede ser conocido su instante de ocurrencia (es obtenido por sensores o cálculos). Por ejemplo: señales de llaves, termostatos, etc.
- *Eventos no observables* ($E_{no} \subseteq E$): Son aquellos en que no puede ser conocido su instante de ocurrencia (no tiene forma de obtenerse). Por ejemplo: variables que no tienen sensores y no son calculables, como velocidad de movimiento de un elemento sin tener ningún sensor.

Es importante destacar que los eventos en un SED solo pueden ser controlables o no controlables, es decir

$$E_u = E - E_c \quad (1)$$

O pueden ser observables o no observables

$$E_{no} = E - E_o \quad (2)$$

La clasificación de eventos nos permite definir las posibilidades de acciones de control en la síntesis de controladores durante el diseño de SED.

Los Autómatas Finitos Deterministas (AFD) también son conocidos en otras literaturas como Máquinas de Estados Finitos Deterministas (MEFD) [26]. En todo el trabajo se mencionan indistintamente, aunque, se utiliza preferentemente MEFD.

Un AFD es un modelo matemático formado por una 5-tupla, tal como se describe a continuación [4], [5], [26], [27], [28]:

$$G = (S, E, T, s_0, F) \quad (3)$$

Donde:

1. $S = \{s_0, s_1, \dots, s_{n-1}\}$ es un conjunto finito de estados. Donde, cada s_i es un estado (nodo) del AFD y n representa la cantidad de estados.
2. $E = \{e_0, e_1, \dots, e_{m-1}\}$ es un conjunto finito de símbolos de entrada. En el caso concreto de este trabajo, se considerará que, cada e_j corresponde a la señal generada por el sensor j -ésimo del sistema (se debe destacar que existen además, otras señales muy útiles generadas a conveniencia y que no necesariamente provienen de sensores físicos, por ejemplo, la señal generada cuando un temporizador lógico termina).
3. $T: S \times E \rightarrow S$ es una función de transición, que transforma pares estado-símbolo en otro estado. Matemáticamente se representa con la tupla (s_i, e_j, s_k) , donde, s_i es el estado de partida, e_j la señal que activa la transición y s_k el estado destino. Esta función de transición se puede representar de manera más simple como una tabla, donde, las filas corresponden a los nodos del conjunto S y las columnas corresponden a las señales del conjunto E , el resultado de combinar una fila i (estado), con una columna j (señal) es un único estado (estado destino) que se coloca en la celda t_{ij} de la tabla, en otras palabras, esto significa que si el sistema se encuentra en el estado s_i y llega la señal e_j , se pasará a un nuevo estado s_k .
4. $s_0 \in S$ es un estado que se considera estado inicial. Este estado representa el punto de entrada al sistema.
5. $F \subseteq S$ es un conjunto finito de estados que se consideran estados de aceptación o finales. En otras palabras, son estados que reconocen o aceptan una cadena (en este caso secuencia de eventos).

Un AFD puede tener estados inaccesibles (estados que jamás pueden ser alcanzados a partir del estado inicial), en el caso contrario son estados accesibles. Para garantizar el funcionamiento deseado del sistema todos los estados deben ser alcanzables o accesibles, pues a partir de un estado inicial muchas veces se requiere que el proceso de eventos discretos avance automáticamente hasta un estado dado, si no tiene esta capacidad, no podrá ejecutar el comportamiento deseado. Si se generaliza esta capacidad a todo el sistema, es aquí donde pueden detectarse partes del proceso que detienen su funcionamiento (como lazos cerrados y bloqueos), lo cual permite eliminar estas situaciones anormales en el programa del PLC desde esta etapa inicial de diseño.

Otro parámetro importante a tener en cuenta es que el proceso de eventos discretos sea controlable para evitar un comportamiento aleatorio. Esto se logra a partir de los eventos controlables que pueden habilitar o deshabilitar transiciones, permitiendo el paso de un estado a otro cuando se decida.

Un AFD se puede representar en forma de diagrama mediante un grafo dirigido etiquetado, conocido también como grafo de transiciones. En este diagrama los nodos son los estados y las aristas etiquetadas por los símbolos representan la función de transición, las aristas representan, además, condiciones desde el punto de vista lógico. Dos aspectos importantes que caracterizan a los AFD son los siguientes:

1. Ningún estado tiene una transición que no esté etiquetada. En este caso quiere decir, que no existe un paso de un estado a otro que no sea provocado por alguna de las señales del sistema, como consecuencia de una condición lógica.
2. Para cada estado $s \in S$ y cada símbolo de entrada $e \in E$, existe a lo sumo una arista etiquetada e que sale de s . Es decir, no puede haber dos transiciones (aristas) que salgan del mismo estado y correspondan a la misma señal, o de otra forma, si existe una transición e correspondiente a una señal desde el estado s , esta sería la única que sale de s y puede estar etiquetada como e .

Como consecuencia de esto es muy fácil determinar el funcionamiento de un AFD, puesto que hay a lo sumo un camino etiquetado con una secuencia de símbolos (señales). Por este hecho, es que se denomina determinista a un autómata finito, ya que en cada instante, solo puede estar en un único estado y a partir de él solo se puede seguir un único camino con la llegada de una señal.

En la figura 1.1 se muestra un ejemplo de representación gráfica de una MEFD, para tener una visualización más sencilla y rápida de su contenido. Aunque, también pueden representarse mediante una tabla o matriz de transición como se mencionó anteriormente, epígrafe 2.2.6.

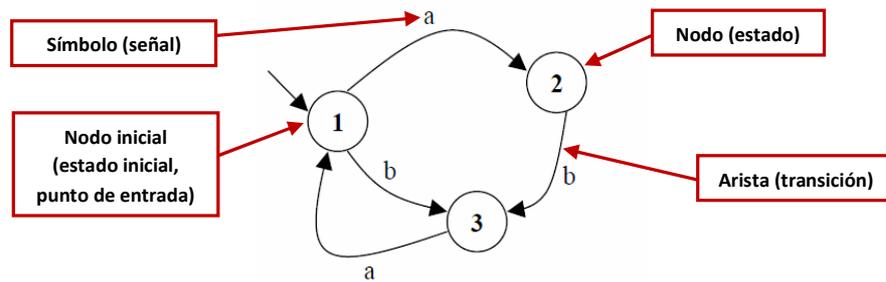


Fig.1.1 Representación gráfica de MEFD

En los estados se realizarán las acciones que determinarán el funcionamiento del sistema. La ocurrencia de estas acciones estará determinada por el estado en el que se encuentre la MEFD, lo cual dependerá de la activación de las señales de entrada. Las transiciones son funciones que dependen de los estados y de las señales de entrada, y definen el paso de un estado a otro. Debido a esto, pueden ser representadas mediante una tabla que relaciona el paso de un estado a través de la ocurrencia de los eventos.

1.3.1.2 Aspectos de construcción de las Máquinas de Estados Finitos Deterministas

Una MEFD se puede implementar esencialmente mediante un programa de computadora que representa una secuencia de instrucciones o pasos lógicos, donde, cada instrucción depende del estado actual de la MEFD y del actual estímulo que recibe [26], [29]. Concretamente, su comportamiento se describe como una secuencia de eventos que ocurren en instantes discretos de tiempo ($t = 1, 2, 3, \dots, etc$).

La selección de un conjunto de estados para representar una MEFD dada no es un proceso único, esto no es una limitación muy seria, ya que el principal objetivo es describir el comportamiento general de entrada-salida de la MEFD en vez de su construcción. El modelo de una MEFD posee sintaxis y semántica formales, y sirve para representar aspectos dinámicos que no se expresan en otros diagramas. En muchas ocasiones es posible descomponer un problema complejo en problemas más pequeños y más fáciles de tratar. Esto también es aplicable a las MEFD, es decir, una MEFD puede descomponerse en varias MEFD más simples.

1.3.1.2.1 Deadlock

Un deadlock (también conocido como: punto muerto, estado de bloqueo, interbloqueo) se presenta en una MEFD si existe algún nodo $S \in S$, desde el que no se efectúa ninguna transición hacia otro estado diferente. Note que un deadlock puede ocurrir incluso fuera de un estado final. La existencia de deadlock no indica un mal modelado de la MEFD y en algunas aplicaciones es deseable, pero en el control de SED por lo general son considerados estados indeseables, que deben evitarse porque provocan la salida del ciclo de trabajo del sistema. En la figura 1.2 se muestra un ejemplo, donde se puede apreciar que desde el nodo 2 no sale ninguna arista, en este caso se puede considerar como deadlock.

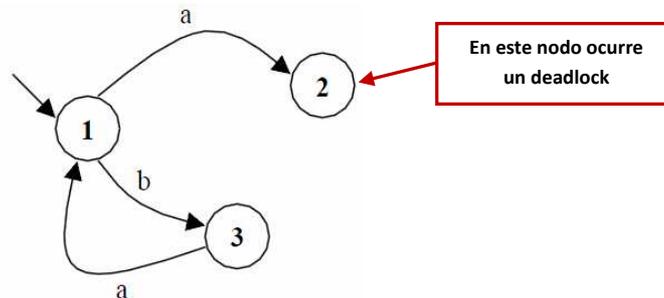


Fig.1.2 Deadlock en MEFD

1.3.1.2.2 Acciones

Las acciones se pueden realizar en los estados e inmediatamente después de ejecutar una transición y antes de llegar al estado de destino. Las acciones, desde el punto de vista de la programación, son instrucciones simples o complejas como asignaciones o instrucciones de flujo. En algunos contextos, dentro de las acciones, también se lanzan eventos que permiten luego sincronizar otras MEFD.

1.3.1.2.3 Clasificación de las Máquinas de Estados Finitos Deterministas

Las MEFD pueden clasificarse de diferentes maneras: en dependencia de las salidas, de la geometría o topología, y de la relación entre sus estados [29], [30], [31]. A continuación se mencionan algunas de las clasificaciones.

1. En dependencia de las salidas:

- *Moore*: Donde las salidas dependen únicamente del estado actual, como se muestra en la figura 1.3.

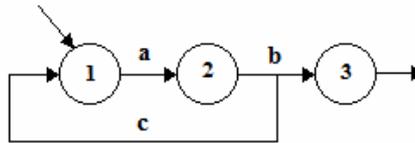


Fig.1.3 Máquina de Moore.

- *Mealy*: Donde las salidas, además de depender del estado actual, también dependen del valor de las entradas, como se observa en la figura 1.4.

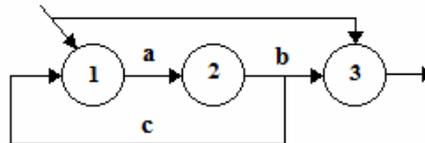


Fig. 1.4 Máquina de Mealy.

2. En dependencia de la geometría:

- *Lineal*: En este tipo de máquinas de estados están geométricamente alineados uno después del otro, como se muestra en la figura 1.5. Estas máquinas son ideales para procesos por lotes o para comunicaciones, en que el reinicio depende de una señal externa.

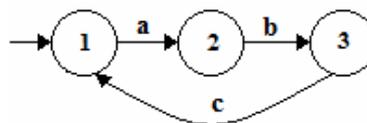


Fig.1.5 Máquina de estados Lineal.

- *Multicolumna*: Son máquinas de estados donde existe un punto que se bifurca dependiendo de las señales de entrada, y cada brazo retorna a un estado inicial. Es usado típicamente para las unidades de control de dispositivos complejos. En la figura 1.6 se muestra un ejemplo de este tipo de máquinas.

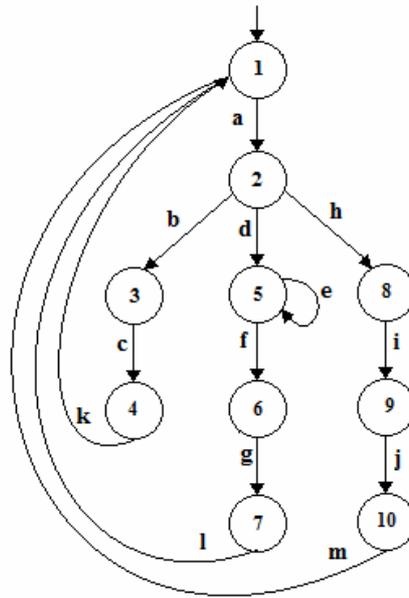


Fig.1.6 Máquina de estados Multicolumnna

- *Multicolumnna bidireccional*: Este tipo de máquinas de estados se diferencia de las anteriores porque éstas retornan al estado inicial por el mismo brazo, como se muestra en la figura 1.7. Son utilizados en secuencias con un punto común.

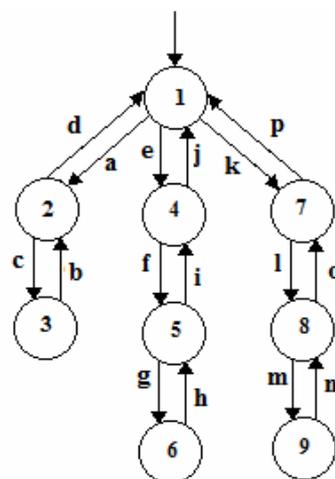


Fig.1.7 Máquina de estados Multicolumnna Bidireccional

- *Caótica*: Las máquinas de estados caóticas no tienen una forma geométrica definida, hay que evitarlas al máximo y nunca trabajar estas máquinas con muchos estados. En la figura 1.8 se muestra un ejemplo.

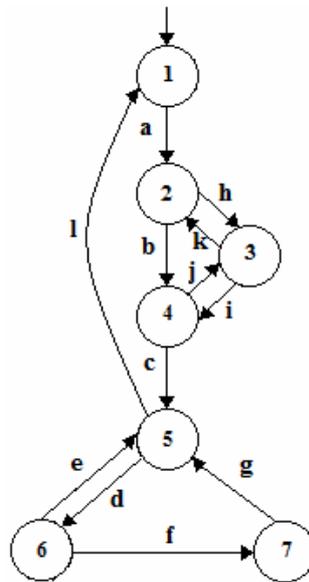


Fig.1.8 Máquina de estados Caótica

- Concéntrica*: El ejemplo típico de estas máquinas de estados es el que se muestra en la figura 1.9. Este tipo de máquinas de estados puede modelar el sistema de control de luces de un semáforo, secuenciadores o controles de motores de paso. Se diferencia de las lineales en que estas últimas son mucho más largas y no necesariamente son cíclicas. Son las máquinas de más frecuente implementación.

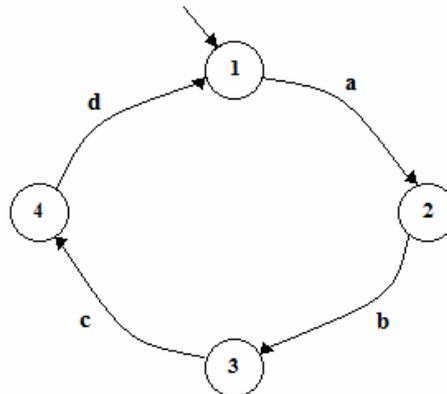


Fig.1.9 Máquina de estados Concéntrica

- En dependencia de la relación entre sus estados:

- Máquinas Completamente Especificadas:* Son aquellas donde cada estado de la máquina de estados reconoce los eventos del conjunto de entrada, y además genera un símbolo de salida cuando se realiza una transición, ya sea a otro estado de la máquina de estados o al mismo. En la figura 1.10 se muestra una máquina de estados que pertenece a esta categoría. Es decir, en una máquina de estados completamente especificada no puede existir ningún estado en el que no entre una arista proveniente de otro estado diferente, ningún estado del cual no salga una arista dirigida a otro estado diferente y además, todas las aristas tienen que estar etiquetadas.

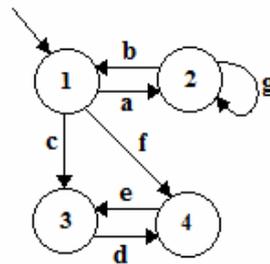


Fig.1.10 Máquina Completamente Especificada

- Máquinas Parcialmente Especificadas:* Son aquellas en las que una o más transiciones son indefinidas o no especificadas, o la máquina de estados reconoce el símbolo de entrada pero no genera una salida. Es decir, en una máquina parcialmente especificada pueden existir transiciones indefinidas (no etiquetadas) y estados desde los que no salga o entre ninguna arista dirigida o proveniente, respectivamente, de un estado diferente. La figura 1.11 muestra un ejemplo de máquinas parcialmente especificadas, ya que existe un estado desde el cual no sale aristas hacia otro estado que no sea el mismo (estado 3), y otro estado al que no llega ninguna arista proveniente de otro estado diferente (estado 1).

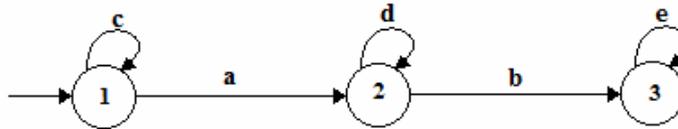


Fig.1.11 Máquina Parcialmente Especificada

- Máquinas Fuertemente Conectadas:* Una máquina de estados con un conjunto S de estados es fuertemente conectada, sí para cualquier par de estados $s_i, s_j \in S$, existe al menos una secuencia de entrada (es decir un camino) tal que el estado s_j pueda ser alcanzado por s_i . Las máquinas fuertemente conectadas son interesantes (teórica y prácticamente) porque tienen la propiedad de que cualquier estado de la máquina puede ser alcanzado desde cualquier otro estado. En la figura 1.12 se muestra un ejemplo típico de este tipo de máquina de estados, ya que desde cualquiera de los dos estados existe un camino (secuencia de aristas) que permite llegar al otro.

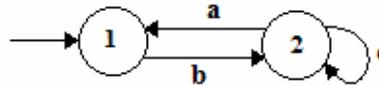


Fig.1.12 Máquina Fuertemente Conectada

Para un proceso industrial lo más conveniente es trabajar con máquinas de estados fuertemente conectadas ya que representan procesos de eventos discretos controlables, debido a que se pueden alcanzar todos los estados al menos a través de una secuencia de transiciones. Según [30] en la teoría de Control Automático el concepto de Sistema Controlable es análogo a la noción de máquina fuertemente conectada.

1.3.1.3 Ventajas del modelado en Máquinas de Estados Finitos Deterministas

La utilización del modelado de SED mediante MEFD, tiene numerosas ventajas, las cuales se mencionan a continuación [32]:

- Las MEFD hacen al código más eficiente, más fácil de depurar y ayudan a organizar el flujo del programa.

- Su simplicidad hace fácil realizar la implementación.
- Predictibilidad (determinismo), dado un grupo de entradas y un estado actual conocido, puede predecirse la transición de estados.
- Dada su simplicidad, las MEFD son rápidas de diseñar, rápidas de implementar y rápidas de ejecutar.
- Las MEFD son relativamente flexibles. Existen varias maneras de implementar un sistema basado en MEFD en términos de su topología, y es fácil incorporar muchas otras técnicas.
- La transferencia desde una representación abstracta del conocimiento a una implementación es fácil.
- Bajo uso del procesador; apropiado para dominios donde el tiempo de ejecución está compartido entre varios módulos o subsistemas. Solo el código del estado actual ha de ser ejecutado, además de un poco de lógica para determinar el estado actual.
- Es fácil determinar si se puede llegar o no a un estado, en las representaciones abstractas, resulta obvio si se puede o no llegar a un estado desde otro, y que requerimientos existen para hacerlo.

De las ventajas presentadas anteriormente se tienen en cuenta para realizar el modelado de procesos industriales su predictibilidad muy útil para modelar procesos secuenciales, su simplicidad y flexibilidad, y su fácil implementación y comprensión.

1.3.1.4 GRAFica de Control de Etapas de Transición (GRAF CET)

En las MEFD no se detallan explícitamente todas las acciones a realizar en los estados, como puede ser el encendido y el apagado de motores, y además dichas acciones pueden estar sujetas a condiciones. Debido a esto la aplicación de MEFD unido al GRAFCET constituye un método válido para el diseño de automatización. En este caso las MEFD aportarían el modelo formal del funcionamiento deseado del sistema, y el GRAFCET como complemento para realizar el algoritmo detallado para la programación a partir de dicho modelo [1], [33].

Una de las herramientas basadas en MEFD usadas para describir o representar sistemas es el GRAFCET, que no es más que un método gráfico de modelado de sistemas de control secuenciales. El GRAFCET consiste en un diagrama gráfico de etapas y transiciones, por

medio del cual se puede llevar a cabo con facilidad la programación del PLC elegido de acuerdo con el software del mismo [1], [34].

El GRAFCET surgió en Francia a mediados de los años 70, y fue creado por una agrupación de algunos fabricantes de PLC, en concreto Telemecanique y Aper, junto con dos organismos oficiales, AFCET (Asociación Francesa para la Cibernética, Economía y Técnica) y ADEPA (Agencia Nacional para el Desarrollo de la Producción Automatizada). Fue homologado en Francia (NFC), Alemania (DIN), y con posterioridad por la Comisión Electrotecnia Internacional (IEC 848, en 1998) [9], [34].

La evolución de un proceso representado mediante un gráfico GRAFCET, está formada por una sucesión de etapas que representan cada uno de sus estados, llevando cada una de ellas asociada una o varias acciones a realizar sobre el proceso, que se ejecutarán si las etapas correspondientes están activas [9], [33], [35]. En la figura 1.13 se muestra un diagrama GRAFCET.

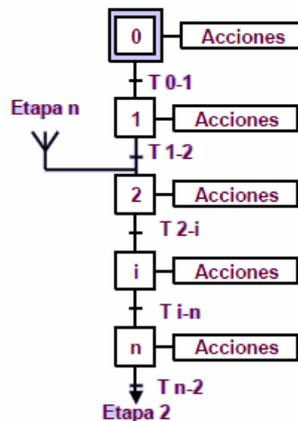


Fig.1.13 Ejemplo de diagrama GRAFCET

Las acciones, en una primera clasificación, pueden ser incondicionales que son aquellas que se ejecutan con solo quedar activadas las etapas correspondientes, o condicionales que son aquellas acciones que necesitan el cumplimiento de una condición además de la propia activación de la etapa correspondiente. Las acciones también pueden ser internas que son aquellas que se producen en el equipo de control, por ejemplo temporizaciones, contadores, cálculos matemáticos; y también pueden ser externas que son aquellas acciones que se

producen sobre el proceso, por ejemplo abrir o cerrar una válvula o activar o desactivar una bomba.

El gráfico GRAFCET correspondiente a un proceso, se compone de una serie de etapas secuenciales que se activan una tras otra, unidas mediante una transición. Las transiciones indican la posibilidad de evolución entre etapas, es decir, la evolución del estado interno del sistema. El paso de una etapa a la siguiente se realiza dependiendo de si se cumple o no la condición de transición entre ellas. Toda transición lleva asociada una condición de transición o función lógica, que puede ser verdadera o falsa.

Los temporizadores son muy empleados para conseguir que el proceso permanezca en un estado (realizando una labor) un determinado número de segundos. Dado que el diagrama en GRAFCET del proceso recoge los estados por los que ha de pasar el proceso, estos temporizadores también han de ser incluidos en el diagrama. El cambio de estado del temporizador puede ser utilizado como condición en la transición. La simbología y las reglas para hacer el diagrama en GRAFCET se muestran en el anexo 2.

En este trabajo es importante el uso del GRAFCET debido a que es la herramienta que se utiliza para describir el algoritmo del programa. Desde el punto de vista de aplicación, tiene más poder expresivo que las MEFD debido a que, además de permitir representar el flujo de transiciones, también permite detallar el comportamiento de cada una de las acciones que ocurren dentro de cada estado. Este poder de descripción no lo tienen las MEFD, y por tanto el GRAFCET es un complemento de las mismas para realizar el algoritmo del programa a partir del modelo realizado previamente.

El método MEFD-GRAFCET permite la realización del modelado y el algoritmo del programa de forma sencilla y fácil traducción al lenguaje de programación. El modelado con MEFD presenta un lenguaje más natural y fácil de entender en comparación con las Redes de Petri, las cuales pueden tornarse muy complejas y de difícil comprensión. Aunque ambos métodos son utilizados para el control de procesos de eventos discretos, las Redes de Petri se pueden hacer extremadamente difíciles para tratar problemas de gran complejidad, a diferencia de las MEFD, ya que la red puede tornarse ilegible cuando es medianamente grande. Esto trae aparejado que cuando se desee modificar el programa puede resultar

complejo en este tipo de red, a diferencia de las MEFD las cuales debido a la organización en el flujo del programa es muy fácil la modificación en la programación.

1.3.2. Programación y simulación

La programación se realiza a partir del modelado obtenido para el control del proceso. Si el modelo del proceso se ha hecho correctamente, la traducción del algoritmo al lenguaje de programación del PLC garantiza el control del proceso. Sin embargo, esta traducción es muy sencilla cuando se parte del algoritmo en GRAFCET, el cual es realizado a partir de un modelo de MEFD, manteniéndose una programación secuencial fácil de modificar en caso de ser necesario.

El SFC (Sequential Function Charts) es un lenguaje de programación basado en diagramas GRAFCET que ya se ha implementado en diferentes dispositivos de algunos fabricantes [11], [35], [36]. La analogía entre este método de algoritmo y los lenguajes de programación es muy sencilla [1], [33], [34], [36].

Una simulación es la ejecución abstracta de un sistema real existente o no por medio de un software de computación. Como una alternativa para el análisis de técnicas desarrolladas, la simulación aparece como un buen compromiso entre modelos analíticos/matemáticos que a menudo son demasiado abstractos con respecto a comportamientos del sistema, y la necesidad de construir sistemas reales experimentales que a menudo son demasiado caros [37]. La simulación del programa para el control del proceso se realiza en el propio ambiente del programa, para así lograr la depuración del mismo en cuanto a errores, teniendo en cuenta el funcionamiento deseado del proceso.

Los lenguajes de programación de PLC han permanecido flexibles, pero las encuestas realizadas por la Control Engineering reflejan claramente favoritos entre las opciones. El lenguaje de programación de contactos (Ladder) es utilizado en un 96% y el de Bloque de Funciones en un 50%, siendo los primeros en la lista de lenguajes de programación de PLC. Sin embargo, el Lenguaje de Texto Estructurado (ST) ha ganado fuerza moviéndose del lugar seis en el 2005 con 13% de uso al tercer lugar en el 2007 con un 24% de uso en la programación de PLC, quedando incluso por encima del SFC, el cual es usado en el 20% de las aplicaciones [6]. Como bien muestran las estadísticas el uso del lenguaje estructurado ha

aumentado, debido principalmente a su analogía con lenguajes de alto nivel, por su forma compacta y fácil modificación en caso de ser necesario, es por ello que se utilizará este lenguaje para la programación del PLC.

1.4 Sistemas de supervisión

Para el control de procesos industriales, además del control realizado para el funcionamiento deseado de la planta, es fundamental la supervisión del proceso, debido a que se pueden implementar sistemas de alarmas, monitoreo, estadísticas de producción, manipulación de variables entre otras funcionalidades. Un sistema de supervisión se encuentra en el nivel superior de la automatización industrial permitiendo la integración de todo el sistema. Se conoce que los sistemas de automatización industrial pueden dividirse en dos niveles de automatización y por tanto dos métodos de modelado específicos [38]:

1. *Sistemas supervisorios*: Generalmente son desarrollados en niveles superiores de redes de automatización industrial, por tanto, son utilizados sistemas de desarrollo de SCADA profesionales. Entre las funciones de supervisión del funcionamiento del sistema, partiendo de las variables históricas más importantes se tiene: atención de alarmas, estadísticas de producción, la planeación de mantenimiento. Además este nivel está evolucionando hacia una integración con la automatización de empresas incrementando sus niveles de inteligencia para flexibilizar la producción en tiempo real acorde con las necesidades del mercado [20].
2. *Automatización Local*: Son utilizadas tecnologías programables en redes de PLC con capacidad de controlar redes de campo de sensores y actuadores inteligentes, una interface hombre-máquina local y comunicaciones con buses de campo para así enlazar con un sistema de supervisión del sistema de automatización. Sus funciones principales están orientadas al control de producción, desarrollo de secuencias de operaciones, estabilización de parámetros principales del proceso, atención a fallas y operación local del sistema. No obstante, el desarrollo de los sistemas inteligentes distribuidos está integrando estas funciones con el sistema de supervisión, para crear un sistema integrado mucho más flexible y eficiente.

En la actualidad, los sistemas de automatización industrial están siendo integrados con los sistemas de automatización empresarial y elevando el nivel de eficiencia, flexibilidad y

seguridad de todas sus funciones. Este desarrollo está siendo acelerado por la necesidad de mantener una alta competitividad de las empresas frente al comercio electrónico globalizado, que obliga a las empresas a garantizar los requerimientos de los usuarios de forma rápida y segura [20].

Otros aspectos importantes a tener en cuenta en el desarrollo de un sistema de supervisión, es la comunicación con el PLC. Los protocolos de redes de PLC también ofrecen una flexibilidad casi ilimitada para el ingeniero en control. Aunque hay algunos favoritos tales como Ethernet, RS232/RS485 y 4-20mA en ese orden. La lista de protocolos de redes usados es larga y variada [6]. Otro aspecto interesante a tener en cuenta es el uso de servidores OPC en caso de que se tengan dispositivos de diferentes fabricantes que necesiten drivers.

1.4.1 Sistemas SCADA

El término SCADA usualmente se refiere a un sistema central que monitoriza y controla un proceso completo o una parte de un proceso. Es un sistema basado en computadores que permite supervisar y controlar variables de proceso a distancia, proporcionando comunicación con los dispositivos de campo (controladores autónomos), y controlando el proceso de forma automática por medio de un software especializado [24], [40], [41]. También provee de toda la información que se genera en el proceso productivo a diversos usuarios, tanto del mismo nivel como de otros usuarios supervisores dentro de la empresa (supervisión, control calidad, control de producción, almacenamiento de datos, entre otros, que contribuyen a la toma de decisiones) [42].

Un sistema SCADA incluye un hardware para señales de entrada y salida, controladores, interfaz hombre-máquina (HMI), redes, comunicaciones, base de datos y software. En este tipo de sistemas usualmente existe un ordenador, que efectúa tareas de supervisión y gestión de alarmas, así como tratamiento de datos y control de procesos. La comunicación se realiza mediante buses especiales o con redes LAN. Todo esto se ejecuta normalmente en tiempo real, y están diseñados para dar al operador de planta la posibilidad de supervisar y controlar los procesos que desee [39].

Entre las principales funciones de los SCADA se encuentra la supervisión remota, que permite conocer el estado de las instalaciones y coordinar eficientemente las labores de producción y mantenimiento en el campo. Mediante el control remoto de instalaciones se activan o desactivan equipos remotos de manera automática o a solicitud del operador. Igualmente es posible realizar ajustes de parámetros en lazos de control analógicos (punto de consigna). Otra de las funciones más utilizadas es la presentación de alarmas, mediante la cual se alerta al operador sobre la ocurrencia de condiciones anormales o eventos que pudieran requerir su intervención. Normalmente, la criticidad del evento o alarma se indica mediante el uso de colores y/o señales auditivas. Las alarmas se registran para análisis posteriores [40], [42], [43]. En el anexo 3 se muestran las funciones básicas de un SCADA.

Dentro de los SCADA más conocidos patentados por diferentes empresas se encuentran WinCC de Siemens, Vijeo Citect de Schneider, CX-One Supervisor de Omron, entre otros.

1.4.2 Servidor para la gestión de la comunicación (OPC)

En un sistema de control, los dispositivos que intervienen en el mismo no tienen que ser necesariamente de la misma firma, por lo que para lograr una comunicación entre ellos a veces es necesario usar drivers, los cuales generalmente son costosos. Esto se puede solucionar con un servidor para el control de proceso basado en tecnología de Microsoft (OPC, de las siglas OLE (Object Linking and Embedding) for Process Control).

Un servidor OPC es un estándar de comunicación en el campo del control y supervisión de procesos industriales. Este ofrece una interface común para comunicación permitiendo que componentes software individuales interaccionen y compartan datos. La comunicación se realiza a través de una arquitectura cliente-servidor. Es una solución abierta y flexible al clásico problema de los drivers propietarios [44]. Prácticamente los mayores fabricantes de sistemas de control, instrumentación y de procesos han incluido OPC en sus productos.

El servidor OPC hace de interfaz, comunicando por un lado con una o más fuentes de datos que utilizan sus protocolos nativos (típicamente PLC, básculas, módulos I/O, controladores, etc.) y por otro lado con clientes OPC (típicamente SCADA, HMI, generadores de informes, generadores de gráficos, aplicaciones de cálculos) [27]. Las comunicaciones entre el cliente

y el servidor son bidireccionales, lo que significa que los clientes pueden leer y escribir en los dispositivos a través del servidor OPC.

1.4.2.1 Tipos de servidores OPC

Existen cuatro tipos de servidores OPC definidos por la OPC Foundation, los cuales se mencionan a continuación [44]:

- *Servidor OPC DA*: Basado en la especificación de Acceso de Datos, especialmente diseñado para la transmisión de datos en tiempo real.
- *Servidor OPC HDA*: Basado en la especificación de Acceso a Datos Historizados, que provee al Cliente OPC HDA de datos históricos.
- *Servidor OPC A&E Server*: Basado en la especificación de alarmas y eventos, transfiere alarmas y eventos desde el dispositivo hacia el Cliente OPC A&E.
- *Servidor OPC UA*: Basado en la especificación de Arquitectura Unificada, basado en el set más nuevo y avanzado de la OPC Foundation, permite a los servidores OPC trabajar con cualquier tipo de datos.

En conjunto, los tres primeros tipos de servidores OPC se conocen como servidores OPC "clásicos" para distinguirlos de OPC UA que se convertirá en la base de las futuras arquitecturas OPC.

Dentro de las principales ventajas de un servidor OPC se debe notar que los fabricantes de hardware sólo tienen que hacer un conjunto de componentes de programa para que los clientes los utilicen en sus aplicaciones, y que los fabricantes de software no tienen que adaptar los drivers ante cambios de hardware. En aplicaciones industriales donde el conjunto de dispositivos involucrados es generalmente heterogéneo, los servidores OPC desempeñan un papel importante en el intercambio y control de los datos de todos estos elementos.

1.4.3 Comunicación industrial

Las redes de comunicaciones industriales deben su origen a la fundación Fieldbus. Esta fundación desarrolló un nuevo protocolo de comunicación para la medición y el control de procesos donde todos los instrumentos pueden comunicarse en una misma plataforma. La

tecnología Fieldbus (bus de campo) es un protocolo de comunicaciones digital de alta velocidad creada para remplazar la clásica señal de 4-20mA que aún se utiliza en muchos de los sistemas de control distribuido, PLC, instrumentos de medida y transmisión y válvulas de control. La arquitectura Fieldbus conecta estos instrumentos con computadores que se usan en diferentes niveles de coordinación y dirección de la planta.

Físicamente podemos considerar a un bus como un conjunto de conductores que conectan conjuntamente varios circuitos para permitir el intercambio de datos. Contrario a una conexión punto a punto, donde solo dos dispositivos intercambian información, un bus consta normalmente de un número de usuarios superior, además, generalmente transmite datos en modo serial [45].

La comunicación industrial por medio de los buses de campo ha incrementado su implementación debido a las diversas ventajas que brinda:

- El intercambio puede llevarse a cabo por medio de un mecanismo estándar.
- Flexibilidad de extensión.
- Conexión de módulos diferentes en una misma línea.
- Posibilidad de conexión de dispositivos de diferentes procedencias.
- Distancias operativas superiores al cableado tradicional.
- Reducción masiva de cables y costo asociado.
- Simplificación de la puesta en servicio.

Existen buses de campo tales como Profibus, AS-I, WorldFIP, CANopen, DiveceNet, Modbus entre otros.

1.5 Descripción del flujo productivo de bloques de hormigón de la planta POYATOS

Para el control de procesos industriales es necesario usar todos los elementos antes descritos para su automatización. Como caso de estudio se propone una sección del flujo de producción de bloques de hormigón de la planta POYATOS del complejo Los Guaos perteneciente a la Empresa de Materiales de la Construcción de Santiago de Cuba.

La producción de bloques de hormigón en el país es una necesidad de primer orden, provocado por la creciente demanda del elemento en las diversas obras constructivas que se desarrollan a lo largo y ancho de la isla. El método de fabricación más utilizado en el país es el método mecanizado, de ahí que se cuente con grandes complejos productivos en los que se encuentran fábricas de diversos tipos de sistema productivo. Las plantas productivas tipo POYATOS son las más utilizadas, de ellas dependen un 73,8 % de la producción general de bloques del país [38]. Por tales razones, el estudio, mantenimiento y mejoramiento del sistema productivo de dichas fábricas son tareas necesarias para lograr mejores volúmenes de producción. La producción de bloques en las plantas POYATOS se puede dividir en cuatro etapas que funcionan de forma secuencial y se repiten cíclicamente como se muestra en la figura 1.14.

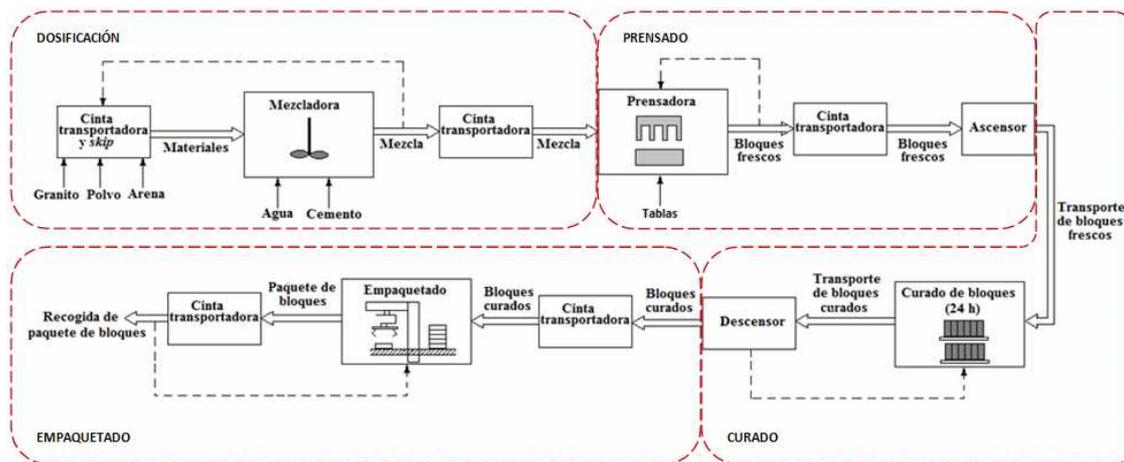


Fig.1.14 Flujo tecnológico del proceso de producción

1.5.1. Dosificación

El proceso comienza con una dosificación de los áridos, en específico arena, polvo de piedra y granito 3/8 pulgadas de granulometría. Estos áridos se encuentran en tolvas individuales con salidas controladas por una pequeña banda transportadora que al ser puesta en marcha permiten la caída sobre una banda transportadora más grande que lleva el material hasta el carro colector o *skip*. La caída de cada material se realiza de forma simultánea.

Paralelamente se inicia la dosificación del cemento almacenado en tolvas por medio de un tornillo sinfín que descarga sobre una báscula. Una vez terminada la dosificación, el *skip* es

elevado hacia una mezcladora donde descarga y se mezcla el árido con agua y cemento durante un tiempo determinado. Al estar lista la mezcla, ésta es descargada en el cajón de prensado por medio de otra banda transportadora.

1.5.2. Prensado

Ya en la máquina de prensado, apoyado en las tablas soporte se llenan los moldes de los bloques y son prensados a una presión determinada durante un tiempo previamente definido en la planificación y que es introducido por el operador. Estas tablas soporte llegan inyectadas al sistema de prensado por medio de un sistema neumático de pistones llamado “inyector de tablas” que las recoge al ser liberadas de los bloques secos. Luego de ser prensada la mezcla se obtienen los bloques frescos, los cuales serán transportados en sus tablas soportes hacia un ascensor, donde al llegarle dos tablas soportes las eleva hasta acumular 10 pisos.

Los bloques que se producen pueden tener diferentes tamaños según el molde que se esté usando en la producción. En la tabla 1.1 se muestran los diferentes tipos de bloques de hormigón.

Tipo de bloque	Cantidad de la prensa	Función
Bloque 20	5	Bloque de carga (zapata de edificios)
Bloque 15	7	Construcciones normales
Bloque 10	10	Construcciones normales

Tabla 1.1: Tipos de bloques.

1.5.3. Transportación y Curado

Una vez lleno el ascensor, un carro transportador, comandado por un operador, recoge los bloques y desplazándose sobre dos raíles, los lleva hacia las celdas de curado, las cuales, poseen la misma cantidad de pisos que el ascensor. Estos bloques son sometidos a curado al aire libre en sus respectivas celdas. Al ser depositados los bloques, se procede con la recogida en una de las celdas adyacentes de bloques secos de producciones anteriores y son llevados al *descensor*. Este *descensor* será el encargado de ir descendiendo piso a piso las tablas soporte y colocándolas en la estructura de transportación de los bloques hacia su empaquetamiento.

1.5.4. Empaquetado

En el empaquetado un sistema automatizado agrupa los bloques y los traslada hacia la línea de salida. Una vez liberadas las tablas quedan nuevamente en el sistema inyector de tablas y los bloques son llevados hacia un depósito en espera de la terminación del curado y el otorgamiento del estado de listo para la venta.

Este proceso descrito se repite de forma cíclica durante toda la jornada de producción.

1.6 Descripción del proceso de empaquetado de bloques

En la actualidad la etapa de empaquetado de bloques de hormigón opera de forma manual e ineficiente, por lo que se describirá de forma detallada este subproceso de la planta.

Para el empaquetado de los bloques se utiliza un manipulador secuencial. Tiene tres grados de libertad, un movimiento a lo largo del eje vertical, en el horizontal y además gira la pinza al colocar los bloques. Los bloques llegan por lotes de 10, 14 ó 20 unidades (en dependencia del tipo de bloque que se esté produciendo, ubicados uno al lado del otro, en forma de fila) a través de una cinta transportadora. Inicialmente la cinta transportadora comienza a moverse por la acción del operador, la cinta se detendrá cuando se active el sensor que indica que hay presencia de bloques debajo de la pinza.

Inicialmente el brazo está arriba, cuando detecta los bloques, baja, los recoge cerrando la pinza y comienza a desplazarse hacia arriba en el eje vertical. Después de llegar al límite superior, denominado posición inicial, se desplaza horizontalmente hasta donde va a dejar los bloques, en este punto, denominado posición final, gira los bloques 0° o 90° en dependencia del índice de la fila transportada, luego los baja de manera vertical y los deposita sobre una cinta transportadora. Posteriormente sube desplazándose en el eje vertical con la pinza abierta, hace un giro en sentido contrario al realizado anteriormente y vuelve a desplazarse horizontalmente hacia la posición inicial.

El ciclo realizado por el manipulador secuencial se muestra en la figura 1.15. Se debe notar que todo el proceso del manipulador es realizado por un operador, el cual manipula el funcionamiento de los motores a través de relés, lo que hace el proceso más lento, además

de que se pueden cometer errores poniendo en peligro la producción de bloques y la seguridad del operario. A la lentitud del proceso también contribuye la existencia de variadores mecánicos, que evidentemente no regulan el arranque y la parada de los motores eficientemente, además, el sentido de movimiento de los motores también lo determina el operador. Debido a este funcionamiento los motores sufren averías con cierta frecuencia repercutiendo negativamente en el proceso de producción.

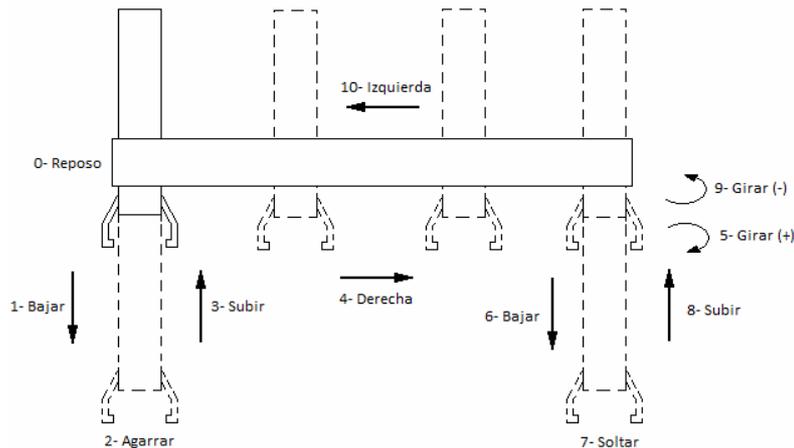


Fig.1.15 Ciclo de trabajo del manipulador secuencial

En el sitio donde el manipulador secuencial deposita los bloques, se forma un paquete de cinco filas de bloques, una encima de la otra, por lo que para completar un paquete el manipulador debe repetir la misma secuencia explicada anteriormente cinco veces. Luego el conjunto de cinco filas de bloques se mueve por la cinta transportadora, dejando el espacio libre para comenzar de nuevo con otro paquete. Las filas de bloques se depositan a cierta altura que está determinada por la cantidad de filas que ya han sido depositadas. El manipulador girará para colocar los bloques de manera alterna, las filas impares se giran 0° , en otras palabras, se colocan de la misma forma que se cogen y las filas pares se giran 90° . El objetivo de esta forma de empaquetamiento es garantizar que la pila compuesta por las cinco filas de bloques no se desmorone. Los motores de ambas cintas tienen variadores mecánicos que mantienen la velocidad constante, pero que no controlan eficientemente el arranque y la parada suave de los motores. En el caso de la cinta encargada de la traslación del empaquetado de bloques, el arranque es más difícil debido al gran volumen de carga que debe mover.

Las características de los motores que intervienen en el movimiento del manipulador secuencial se muestran en la tabla 1.2.

Denominación	Marca	Tipo	kW	Vn	In	RPM	Hz
Mov. Vertical	ASEA	MBTF 112 Mc-6	2,2	440	5,1	1125	60
Mov. Horizontal	ASEA	MBTF 90 Lc-4	1,5	440	3,2	1700	60
Giro pinza	ASEA	MBTF 80 B-6	0,75	440	1,8	1690	60

Tabla 1.2: Parámetros de los motores para el movimiento del manipulador

Los requerimientos que debe cumplir el proceso se detallan a continuación

- Productividad y eficiencia en la secuencia automatizada.
- Funcionamiento de una articulación a la vez para el movimiento del manipulador.
- Se debe cumplir el ciclo completo del manipulador para completar el empaquetado de bloques.
- Variar la altura a la que se depositan los bloques.
- Arranque suave y parada suave de los motores de las cintas transportadoras.
- Control de velocidad de los motores del manipulador secuencial
- Velocidades del manipulador no superior a los 0,8 m/s debido a los grandes volúmenes de carga.
- Verificación de las alarmas.
- Estadísticas de bloques empaquetados.

Debido a que es necesario el incremento de la productividad, lo que se traduce en ganancias para el país, se requiere la automatización eficiente de la planta, lo que también contribuye a aumentar la fiabilidad del funcionamiento del sistema. El proceso descrito anteriormente puede definirse como un proceso de eventos discretos por su funcionamiento, por esta razón se modelará el proceso de empaquetado de bloques utilizando MEFD, y a partir de dicho modelo se realizará el algoritmo en GRAFCET. Para el control del proceso se utilizará un PLC, programado a partir del algoritmo en GRAFCET para lograr el funcionamiento deseado de la planta, teniendo en cuenta el uso de dispositivos de control para controlar los motores implicados en el proceso. Para lograr una automatización integrada que aumente las potencialidades del funcionamiento del sistema, también es necesario el desarrollo de un

sistema de supervisión. Con todo ello lo que se pretende es aumentar la producción de bloques empaquetados con la disminución de los tiempos del proceso, disminuyendo además, los niveles de peligrosidad y la generación de errores humanos, con lo que se garantiza una mejora considerable en la eficiencia y calidad del proceso. Las herramientas presentadas a lo largo del capítulo constituyen la base para la automatización del proceso de empaquetado de bloques.

1.7 Conclusiones

Se realizó un breve estado del arte del uso de los PLC en control de procesos industriales, así como, el uso de los variadores de frecuencia ampliamente utilizados para el control de máquinas. Se analizaron las técnicas de control de eventos discretos y en concreto el modelado de sistemas a partir de MEFD en conjunto con el desarrollo de algoritmos basados en diagramas GRAFCET, mostrando sus principales características y ventajas. Se realizó además, un análisis sobre los sistemas SCADA para la supervisión, gestión y control a distancia de procesos, así como, de las comunicaciones industriales utilizando buses de campo y servidores OPC. Se desarrolló la caracterización del flujo tecnológico correspondiente a la planta productiva de bloques de hormigón POYATOS. Enfatizando en el bajo nivel de automatización de la etapa de empaquetado de bloques de hormigón, como caso particular de estudio. Todo esto permite acometer con rigor científico el diseño y programación del sistema de automatización del proceso de empaquetado de bloques.

Capítulo II

2.1 Introducción

En este capítulo se analiza el funcionamiento automático del proceso de empaquetado de bloques de hormigón de la planta POYATOS, y los lazos cerrados implementados para el control. Se desarrolla completamente el modelo del sistema automatizado, basado en MEFD (epígrafe 2.2.6), en conjunto con el desarrollo del algoritmo del programa para el control, utilizando diagramas GRAFCET que se describen en los epígrafes 2.3.1, 2.3.2 y 2.3.3. Se analizan algunos aspectos de la programación y de la simulación del programa desarrollado en lenguaje estructurado. Se explica además, la configuración del sistema SCADA desarrollado para la supervisión, gestión y control del proceso a distancia, haciendo uso de un servidor OPC para manipular las variables del proceso, usando Modbus como bus de campo (epígrafe 2.4). Finalmente, se analiza la eficiencia que proporciona el sistema automatizado con respecto al sistema actual en diferentes aspectos.

2.2 Etapa de diseño para la automatización del proceso de empaquetado de bloques

En el capítulo anterior, se expuso de forma detallada la descripción del proceso para la construcción de bloques de hormigón en la planta POYATOS. Se analizaron los principales problemas del funcionamiento de la planta y sus deficiencias. Es por ello que a continuación se exponen los detalles de la automatización del proceso.

Para lograr la automatización del proceso es necesario definir las etapas del proceso, los estados y las señales necesarias para el control, tal como se describe a continuación.

2.2.1 Traslación de los bloques que serán empaquetados

La primera etapa del proceso consiste en la traslación de los bloques que serán empaquetados, a través de una cinta transportadora, a la posición donde posteriormente serán recogidos por el manipulador secuencial. Se definen dos estados principales para

automatizar el movimiento de la cinta transportadora 1 del proceso de empaquetado, siendo esta la primera etapa de este proceso.

0. *Reposo o inicial:* Es la entrada o punto de partida del ciclo, describe el comportamiento en que debe encontrarse la cinta transportadora antes de comenzar la ejecución de un ciclo. En el mismo, se realizan acciones de inicialización y verificación.
1. *Desplazamiento de los bloques por la cinta transportadora:* Se ejecuta cuando el manipulador secuencial haya recogido los bloques. La cinta transportadora se detendrá, cuando se active un sensor final de carrera que indique la llegada de las tablas con bloques que serán empaquetados.
 - En este estado interviene el motor para el movimiento de la cinta transportadora y el arrancador que accionará sobre el motor.
 - *Detección de bloques:* Final de carrera ($S1^2$) que se activa cuando se posicionan las dos bandejas con bloques debajo de la pinza. El sensor escogido es de la familia XCK-J de Schneider [46], como el mostrado en la figura 2.1. La cabeza tiene movimiento rectilíneo con varilla. Todos los finales de carrera utilizados son de este tipo.

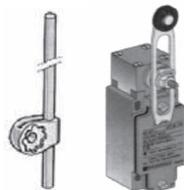


Fig.2.1 Interruptor de posición de Schneider

2.2.2 Empaquetado de bloques

En esta segunda etapa interviene el manipulador secuencial. Se definen once estados principales para automatizar el movimiento del manipulador del proceso de empaquetado, teniendo en cuenta la secuencia de pasos que realiza el manipulador para completar un ciclo, tal como se describe en la figura 1.15. El significado de los mismos es el siguiente:

² Notar que para no entrar en contradicción ni hacer abuso de notación, los estados se denotan con la letra s pero en minúscula, mientras que los sensores se denotan con la letra S en mayúscula más un número.

0. *Reposo o inicial:* Es la entrada o punto de partida del ciclo, comprueba las precondiciones que deben cumplirse en el manipulador antes de comenzar la ejecución de un ciclo. En el mismo, se pueden realizar otras acciones de inicialización y verificación.
 - En este estado el manipulador está en posición inicial, determinado por un sensor que detecta la posición del soporte de la pinza en esa posición, en este estado la pinza estará abierta.
1. *Desplazamiento de la pinza hacia abajo de forma vertical:* Se ejecuta cuando se activa el sensor de final de carrera (S1, es la misma señal proveniente del sensor utilizado en la cinta transportadora 1) que detecta la presencia de bloques.
 - En este estado interviene el motor designado para el movimiento vertical, así como el variador que controlará el mismo.
2. *Cierre de la pinza:* Se ejecuta de forma mecánica cuando ésta entra en contacto con los bloques, es decir, cuando llega a la posición de agarrar.
 - *Posición de agarrar:* Comparación de la altura actual con una referencia correspondiente a la posición de la pinza para agarrar los bloques, la cual es siempre la misma. Esto se logra con un encoder óptico rotativo. Se escogió un encoder incremental de la familia de Schneider, con referencia XCC1506 [47].
3. *Desplazamiento vertical hacia arriba con la pinza cerrada con bloques:* Se ejecuta cuando la pinza ha cerrado completamente.
 - En este estado interviene el motor designado para el movimiento vertical, así como el variador que controlará el mismo.
 - *Pinza cerrada:* Sensor inductivo (S2) que se activa cuando la pinza cierra totalmente. El sensor inductivo escogido es de la firma de Schneider, modelo XS618B1PAM12 [48], el mismo se muestra en la figura 2.2. Todos los sensores inductivos utilizados serán de este tipo.



Fig.2.2 Sensor inductivo de proximidad de Schneider

4. *Desplazamiento horizontal del soporte de la pinza con ésta incluida:* Se ejecuta cuando se detecte la pinza cerrada en la posición inicial. Se detendrá el movimiento de traslación cuando se llegue a la posición final.
 - En este estado interviene el motor designado para el movimiento horizontal y el variador que controlará el mismo.
 - *Posición inicial:* Es una operación AND entre la señal emitida por un sensor inductivo (S3) que ubica el soporte de la pinza en esa posición inicial y la posición de pinza arriba, se determina de manera similar a la posición de agarrar, comparación de la altura actual con una referencia. Esto se logra con el mismo encoder óptico rotativo ya que es el mismo motor.

5. *Giro de la pinza cerrada con bloques en sentido positivo:* Se ejecuta cuando el manipulador ha llegado a su posición final, los giros realizados son de 0° y 90° para filas con índices impares y pares, respectivamente. En este estado se interactúa con el proceso de la cinta transportadora, es decir, se verifica si la cinta transportadora está en movimiento, y si es así el manipulador no pasa al siguiente estado debido a que no se pueden depositar los bloques mientras la cinta transportadora se mueve.
 - En este estado interviene el motor designado para el giro de la pinza y el variador designado para ello.
 - *Posición final:* Es una operación AND entre la señal emitida por un sensor inductivo (S4) que ubica el soporte de la pinza en esa posición final y la posición de la pinza arriba, se determina de manera similar a la posición de agarrar, comparación de la altura actual con una referencia. Esto se logra con el mismo encoder óptico rotativo ya que es el mismo motor.

6. *Desplazamiento de la pinza hacia abajo de forma vertical:* Se ejecuta cuando el manipulador ha llegado a la posición final y se ha girado la pinza.
 - En este estado interviene el motor designado para el movimiento vertical, así como el variador que controlará el mismo.
 - *Posición de pinza girada en sentido positivo* (definido a priori): Sensor inductivo (S6) que se activa cuando la pinza giró 90° en sentido positivo.

7. *Apertura de la pinza:* Se ejecuta cuando ésta llega a la posición en la que debe depositar los bloques, está abierta completamente cuando se activa un sensor que indica que la pinza está completamente abierta.
 - *Posición de depositar bloques:* Comparación de la altura actual con una referencia correspondiente a la posición de la pinza para depositar los bloques, teniendo en cuenta que esta referencia varía según la cantidad de filas que ya se hayan colocado en el paquete de bloques. Esto se logra con un encoder óptico rotativo siguiendo la ecuación (4).

8. *Desplazamiento vertical hacia arriba con la pinza abierta sin bloques:* Se ejecuta cuando la pinza se ha abierto completamente.
 - En este estado interviene el motor designado para el movimiento vertical, así como el variador que controlará el mismo.
 - *Pinza abierta:* Sensor inductivo (S5) que se activa cuando la pinza se abre completamente.

9. *Giro de la pinza abierta sin bloques en sentido negativo:* Se ejecuta cuando se detecte la pinza abierta en su posición final, los giros realizados son de 0° y -90° para filas con índices impares y pares, respectivamente.
 - En este estado interviene el motor designado para el giro de la pinza y el variador designado para ello.
 - *Posición de pinza girada en sentido negativo* (definido a priori): Sensor inductivo (S7) que se activa cuando la pinza gira 90° en sentido negativo.

10. *Desplazamiento horizontal del soporte de la pinza con ésta incluida*: Se ejecuta cuando se detecte la pinza abierta en la posición final. Se detendrá el movimiento de traslación cuando se llegue a la posición inicial.

- En este estado interviene el motor designado para el movimiento horizontal y el variador que controlará el mismo.

2.2.3 Traslación del paquete de bloques para su recogida

La tercera etapa del proceso consiste en la traslación del paquete de bloques a la posición donde serán recogidos. Se definieron 2 estados principales para automatizar el movimiento de la cinta transportadora 2 del proceso de empaquetado.

0. *Reposo o inicial*: Es la entrada o punto de partida del ciclo, describe el comportamiento en que debe encontrarse la cinta transportadora antes de comenzar la ejecución de un ciclo, en el mismo se realizan acciones de inicialización y verificación.
1. *Desplazamiento del paquete de bloques por la cinta transportadora*: Se ejecuta cuando se hayan completado las cinco filas de bloques. Este movimiento se detendrá cuando se active un sensor que indique la llegada del paquete de bloques al lugar donde será recogido.
 - En este estado interviene el motor para el movimiento de la cinta transportadora y el arrancador que accionará sobre el motor.
 - *Detección de paquete de bloques en la cinta transportadora*: Final de carrera (S8) que indica que el paquete está listo para recogerse.

2.2.4 Sistema de automatización para el control del empaquetado de bloques

El diseño de la estructura general del sistema que muestra la interacción de las partes que lo conforma se muestra en la figura 2.3.

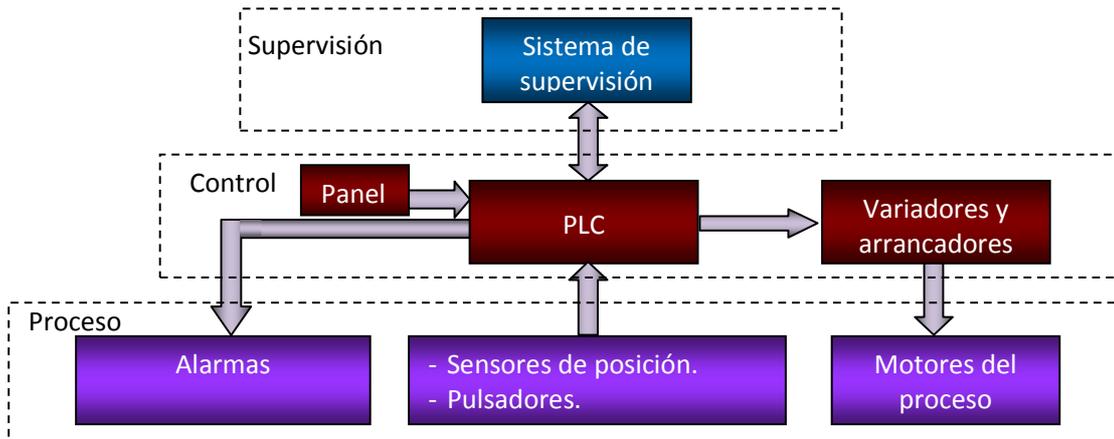


Fig.2.3 Esquema de automatización

La figura anterior muestra los tres niveles a tener en cuenta para la automatización del proceso de empaquetado de bloques. Un nivel inferior compuesto por los sensores de posición ubicados en el proceso, los pulsadores de emergencias, las alarmas y los motores que garantizan los movimientos de las cintas transportadoras y del manipulador. En un nivel medio se encuentra el PLC donde estará programado el control que permitirá el funcionamiento del sistema. El PLC interactúa con el proceso a través de los módulos de entradas-salidas que posee. En este nivel también se encuentran los variadores de frecuencia que permiten controlar la velocidad de los motores del proceso. En un nivel superior se encuentra el sistema de supervisión, a través del cual se podrán manipular algunas variables, y donde se visualiza, además, el estado de las diferentes señales provenientes de los sensores instalados en el sistema. La comunicación del sistema de supervisión con el PLC se logra con un servidor OPC con conexión Modbus al PLC.

Para la automatización del proceso de empaquetado, se seleccionó el PLC BMX P34 2010, de la familia MODICON M340 de Schneider. La CPU tiene puertos de comunicación integrados de buses CANopen y Modbus [49]. Dentro de las ventajas de este tipo de PLC se destacan las siguientes [49]:

- Todos los módulos se pueden quitar y poner en “caliente” con la CPU en RUN (excepto la fuente de alimentación (PS) y la CPU). Alta robustez, esto se debe a la arquitectura basada en rack.
- Debido a que no incorporan módulos de E/S de serie, permiten una configuración libre al usuario, multiplicando sus posibilidades mediante el rack extensible opcional.

- Maneja una amplia gama de módulos de entradas y salidas.
- Trabaja normalmente hasta altitudes de 4000 m.
- Para implementar las variables en el UnityPro, la nomenclatura que se utiliza queda avalada por la normativa IEC-1131. En esta normativa las variables de E/S se identifican en función de la posición del módulo que ocupen físicamente dentro del PLC.

Debido a las exigencias de la aplicación del empaquetado de bloques, además por características técnicas y por homogeneidad con el PLC y los equipos ya montados, se seleccionó el ALTIVAR 28 (ATV 28) para el control de los motores del manipulador secuencial. Este variador de frecuencia, mostrado en la figura 2.4, es de una de las familias de variadores de Schneider. El mismo posee entradas y salidas digitales y analógicas incorporadas, y un panel de visualización y configuración incluido, que permite adaptarlo a cualquier aplicación [50].



Fig.2.4 Variador Altivar 28

Las características técnicas del variador escogido, teniendo en cuenta las características de los motores del manipulador secuencial, se muestran en la tabla 2.1, y el valor de la resistencia mínima de frenado será de 25 Ω [51].

Número de catálogo	Corriente de línea entrante ^[1]		Potencia del motor		Corriente de salida nominal ^[3]	Corriente de salida transitoria ^[2]	Potencia disipada total a la carga nominal	Corriente nominal de cortocircuito
	400 V	460 V	kW	hp				
ATV28HU41N4U	8,0	7,0	2,2	3	5,5	8,3	81	5

Tabla 2.1: Tensión de corriente eléctrica trifásica: 400/460 V~ -15%, +15%, 50/60 Hz

Para el control del manipulador secuencial solo se han utilizado dos velocidades configuradas en el variador de frecuencia debido a la corta distancia de traslación que recorre el brazo mecánico. Una velocidad será cero y la otra estará determinada por la referencia. También se han configurado los sentidos de giro del motor y la parada, esta última se realiza utilizando la función de parada por inyección directa que brinda este variador, permitiendo con esto anular completamente en un intervalo de tiempo muy pequeño la inercia del motor, esto además es posible, por la baja velocidad que lleva el motor debido a al peso de la carga que se transporta.

Los arrancadores son equipos que también permiten el control de motores, pero sin presentar todas las potencialidades que tiene un variador de frecuencia. Para el control del motor de la cinta transportadora se escogió el arrancador Altistart 01 (ATS 01), que se muestra en la figura 2.5, el cual pertenece a una de las familias de arrancadores de Schneider.



Fig.2.5 Arrancador Altistart 01

El arrancador Altistart 01 es un limitador de torque, que mejora los comportamientos de arranque de los motores asíncronos, permitiendo un arranque suave, controlado y sin golpes. Su utilización permite la supresión de los choques mecánicos que causan el desgaste, tiempos de mantenimiento y tiempos de detención de la producción. Este tipo de arrancador está destinado para aplicaciones tales como, cintas transportadoras, bombas, ventiladores, compresores, puertas automáticas, entre otras [52]. Algunas características técnicas del arrancador a usar en la aplicación, así como el modelo, se muestran en la tabla 2.2.

Arrancador suave para motores de 0,37 a 5,5 kW				
Motor	Arrancador			
Potencia motor	Potencia	Corriente	Referencia	Peso
400 V	disipada	nominal		
kW	W(1)	W(2)	A	kg
Tensión de alimentación trifásica 200...480 V 50/60 Hz				
5,5	1	61	12	ATS 01N112FT 0,280

Tabla 2.2: Arrancador Altistart 01

Las conexiones eléctricas del variador ATV 28 y del arrancador ATS 01 se muestran en los anexos 4 y 5, respectivamente. En este punto se debe señalar, que para el control realizado se puede usar un variador para cada motor, que sería el caso ideal recomendado, pero también es posible usar un solo variador para controlar los tres motores, ya que el proceso es secuencial, y nunca hay dos motores funcionando al mismo tiempo. Independientemente de esto la programación funciona para ambos casos, la única diferencia sería la conexión eléctrica, la cual se muestra en el anexo 6 en caso de usar solamente un variador. Se debe aclarar también que con este tipo de conexión se logra una disminución en la inversión, pero se pierde en seguridad y fiabilidad del sistema. A lo largo del documento se hablará de un variador por cada motor.

2.2.5 Funcionamiento del sistema automatizado

El sistema de control para la etapa de empaquetado de bloques de hormigón en la planta productiva POYATOS, puede operar en modo manual o automático. En el modo manual, el operador podrá accionar directamente el manipulador secuencial mediante el uso de botones y llaves de doble posición. Esto provocará los diferentes eventos que definen el movimiento del manipulador de manera libre y bajo su criterio de apreciación, tal como sucede actualmente. Aunque, si es necesario transitar los estados del sistema de manera organizada. En la figura 2.6 se muestra de forma general la solución para la automatización del proceso de empaquetado de bloques de hormigón, considerándose la entrada al sistema las filas de bloques, y la salida, el paquete de bloques confeccionado.

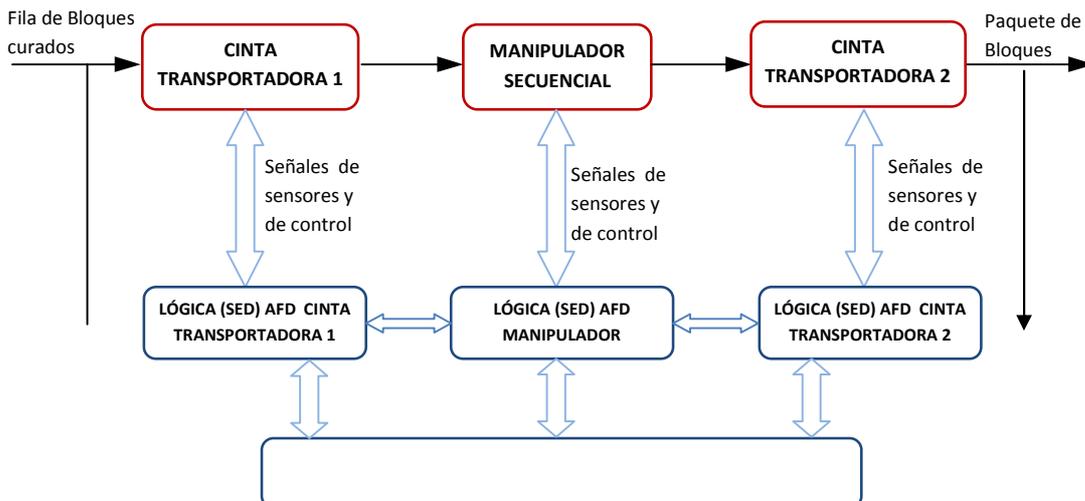




Fig.2.6. Esquema general de la solución de automatización del proceso de empaquetado

Inicialmente la cinta transportadora encargada del traslado de los bloques que se empaquetaran comienza a moverse, y se detendrá cuando se active el sensor que indica presencia de bloques debajo de la pinza.

Cuando se detecta la presencia de bloques, se activa S1, la pinza comienza a moverse verticalmente hacia abajo, mediante el motor designado para el movimiento vertical (motor1). Cuando la pinza esté a la altura deseada (altura para agarrar los bloques), lo cual se sabrá por la comparación de la lectura del codificador incremental y un SetPoint (a partir de ahora SP) equivalente a la altura deseada, entonces se detendrá el motor1, y se comenzará a cerrar la pinza de forma mecánica para el agarre de los bloques. Al activarse el sensor S2 se indica que la pinza está cerrada totalmente, y comienza a moverse la pinza hacia arriba, es decir, el motor1 se mueve en sentido contrario, deteniéndose cuando la distancia recorrida verticalmente es igual al SP correspondiente a la altura. Esta posición del brazo manipulador con la pinza arriba y la activación del sensor S3 es denominada posición inicial.

Cuando se detecta esta posición con la pinza cerrada, el manipulador se desplaza horizontalmente en el sentido de la posición final mediante el motor designado para el movimiento horizontal (motor2). La posición final está definida por la activación del sensor S4 y la pinza arriba. Cuando el manipulador llega a esta posición se detiene el motor2. Notar que los sensores S3 y S4 indican la posición del soporte de la pinza, y la posición de la pinza, arriba o abajo, se indica con el codificador.

Cuando el manipulador está en la posición final se girará la pinza 0° o 90° (en el caso de 90° se activa S6) en dependencia del índice par o impar de la fila de bloques transportada. Esto se debe a que es necesario depositar los bloques de manera alterna (las filas impares se

colocan de la misma forma que se cogen, es decir, se les realiza un giro de 0° y las filas pares se giran 90° , el giro de la pinza es realizado por el motor3 mediante el variador designado para ello). Después de haber girado la pinza, se procede a depositar los bloques poniendo en marcha el motor1 correspondiente al movimiento de la pinza verticalmente. La pinza se detendrá según la altura para depositar, la cual viene determinada por el número de filas que se hayan depositado, por lo que el SP de la altura a la que se depositará los bloques variará a medida que se vayan colocando filas (la variación de la distancia está determinada por la altura de un bloque). El motor1 se detendrá cuando la lectura del decodificador (altura medida) sea igual al SP.

Al detenerse el motor1 en la altura deseada se abre la pinza. Al activarse S5, el cual indica que la pinza se ha abierto completamente, comienza a desplazarse la pinza hacia arriba, hasta recorrer la altura deseada. Al llegar a la posición final con la pinza abierta, se activa S4, lo que provoca la detención del motor1. En esta posición la pinza se gira 0° o -90° , es decir en sentido opuesto al giro realizado anteriormente según corresponda, activándose S7 cuando concluye el giro. Luego el manipulador comienza a moverse horizontalmente hasta llegar a la posición inicial para comenzar un nuevo ciclo.

Para completar un paquete de bloques el manipulador debe realizar el ciclo anterior cinco veces. El proceso relacionado con la pinza y la descarga de los bloques está condicionado a que la cinta transportadora no esté en movimiento, como es lógico, no se depositarán bloques mientras la cinta transportadora esté moviéndose. Debido a esto, en el estado 5 del proceso justo antes de depositar los bloques se verificará esta condición, quedando el proceso en paro temporal hasta que la cinta transportadora haya finalizado su ciclo y se encuentre nuevamente en estado de reposo. Por otro lado, el proceso relacionado con la cinta comienza su ejecución cuando se cumple la condición de haber depositado cinco filas de bloques, una encima de la otra. En este momento la cinta transportadora comienza a moverse hasta que se activa el final de carrera S8, que indica que el paquete está en la posición deseada para la recogida del mismo, cuando se ha retirado la pila de bloques el proceso de la cinta está listo para iniciar un nuevo ciclo.

2.2.6 Modelado y diseño de la automatización del proceso de empaquetado de bloques

El sistema de control propuesto puede operar en modo manual o automático, además, tiene implementado un sistema de alarmas. En el modo manual, el operador podrá accionar directamente el manipulador secuencial. En modo automático el sistema transita por todos los estados por medio de las señales provenientes de los sensores, es decir funciona a lazo cerrado como se explicó anteriormente.

Se realizó el modelado del sistema sobre la base del funcionamiento deseado del proceso. En la figura 2.7 se muestra la MEFD correspondiente al funcionamiento manual/automático del proceso. La MEFD tiene un estado inicial (estado 0), en el cual se realizan las inicializaciones del proceso, se verifican las variables de alarmas y se realizan determinadas acciones en caso de que estén activadas. Si el funcionamiento está en modo manual, se ejecutarán las acciones correspondientes en el estado 1, y si está en modo automático se pasará al estado 2. Cabe destacar en este punto que cada uno de los estados 1 y 2 son a su vez modelados como MEFD.

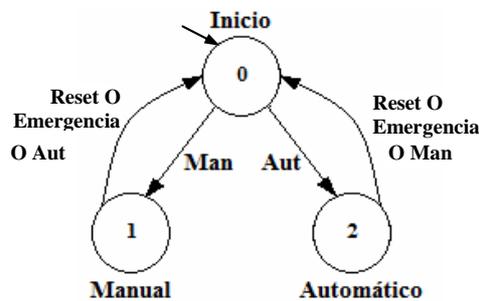


Fig.2.7 Diagrama en MEFD del funcionamiento manual-automático del proceso

Para el funcionamiento manual/automático del proceso de empaquetado la 5-tupla correspondiente a la MEFD queda representada de la siguiente manera:

$$S = \{s_0, s_1, s_2\}, \quad \text{conjunto finito de estados (3 estados)}$$

$$E = \{e_0, e_1, e_2\}, \quad \text{conjunto finito de señales (3 señales)}$$

$s_0 = s_0$, *estado inicial o de partida*

$F = \emptyset$, *conjunto de estados finales, vacío en este caso*

La función de transiciones $T: S \times E \rightarrow S$ se representa mediante la tabla de transiciones (tabla 2.3) correspondiente al proceso, donde, las filas representan los estados, las columnas representan las señales y el resto de celdas representan las transiciones.

Estado/Señal	Man (I)	Aut (II)	Reset o Emergencia (III)
0-Inicio	1-Manual	2-Automático	
1-Manual		0-Inicio	0-Inicio
2-Automático	0-Inicio		0-Inicio

Tabla 2.3: Tabla de transiciones para el funcionamiento manual-automático del proceso

El funcionamiento manual o automático estará determinado por la selección del operador, como se muestra en la tabla de transiciones, si está en modo manual y el operador selecciona modo automático, pasará a funcionar como automático y viceversa. En caso de que se active el Reset o alguna Emergencia se activarán las alarmas configuradas en el estado inicial.

2.2.6.1 Modelado del modo Manual

En el modo manual se definen cinco MEFD, una para cada motor que interviene en el proceso de empaquetado de bloques. La MEFD correspondiente a cualquiera de las dos cintas en el proceso manual se muestra en la figura 2.8. Las MEFD correspondientes al funcionamiento de los motores del manipulador son similares a la presentada en la figura 2.8.

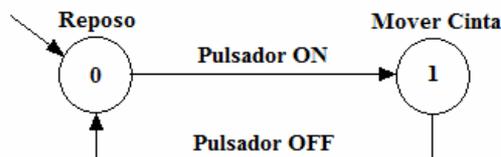


Fig.2.8 Diagrama en MEFD de la Cinta transportadora en modo manual

La 5-tupla correspondiente a la MEFD queda representada de la siguiente manera:

$S = \{s_0, s_1\}$, *conjunto finito de estados (2 estados)*

$E = \{e_0, e_1\}$, conjunto finito de señales (2 señales)

$S_0 = s_0$, estado inicial o de partida

$F = \emptyset$, conjunto de estados finales, vacío en este caso

La función de transiciones $T: S \times E \rightarrow S$ se representa mediante la tabla de transiciones (tabla 2.4) correspondiente al proceso, donde, las filas representan los estados, las columnas representan las señales y el resto de celdas representan las transiciones.

Estado/Señal	Pulsador ON (I)	Pulsador OFF (II)
0-Reposo	1-Mover cinta	
1-Mover cinta		2-Reposo

Tabla 2.4: Tabla de transiciones para la cinta transportadora en modo manual

Como se muestra en la tabla de transición la cinta transportadora de bloques se moverá cuando el operador presione el pulsador llevándolo a ON, y se detendrá cuando el operador lleve el pulsador a OFF.

2.2.6.2 Modelado del modo Automático

En el estado automático hay tres máquinas de estados diseñadas para el control del proceso de empaquetado de bloques. Una primera MEFD para el control de la cinta transportadora de bloques que serán empaquetados. Una segunda MEFD para el control del manipulador secuencial, y una tercera MEFD para el control de la cinta transportadora de los paquetes de bloques. Estas tres MEFD funcionan de manera independiente aunque están relacionadas. La cinta transportadora de bloques (cinta transportadora 1) garantiza que los bloques estén en la posición donde serán recogidos por la pinza, es decir, que mientras no se active la señal que detecta los bloques el manipulador no podrá comenzar el empaquetado. Por otro lado, en el caso de que se hayan depositado cinco filas de bloques en la cinta transportadora y el manipulador comience un nuevo ciclo, tendrá que esperar a que la cinta transportadora se detenga para poder continuar. El sincronismo del proceso se logra a través de señales internas del programa o externas provenientes de sensores.

Para que el proceso funcione de forma continua, es decir, sin tiempos de esperas en las cintas transportadoras ni en el manipulador, es necesario garantizar ciertas condiciones de tiempo. Por ejemplo, el tiempo que tarda la cinta transportadora encargada de trasladar los bloques curados en posicionar los bloques, debe de ser menor que el tiempo de ciclo del manipulador, con esto se garantiza que siempre haya bloques en posición para ser recogidos por el manipulador. También se debe garantizar que el tiempo de transporte del paquete de bloques, es decir, el tiempo que emplea la cinta transportadora de paquete de bloques en trasladar los bloques, sea menor que el tiempo de ciclo del manipulador. Con esto se garantiza que cuando el manipulador vaya a depositar la primera fila de un paquete en la cinta transportadora, la misma se encuentre detenida. Estas condiciones se logran con las configuraciones de las velocidades de los motores a través de los variadores, siempre y cuando, el resto de las etapas cumplan con los tiempos estrictamente establecidos de producción, para garantizar que el flujo de bloques sea continuo. Otro aspecto que se debe tener en cuenta son las paradas por eventos inesperados.

El sistema de control está compuesto por cinco lazos cerrados. Tres lazos para el movimiento del manipulador secuencial, y dos para controlar el funcionamiento de las cintas transportadoras.

2.2.6.2.1 Modelado de la cinta transportadora de bloques para empaquetar

En el lazo correspondiente a la cinta transportadora de bloques que serán empaquetados, mostrado en la figura 2.9, intervienen varios dispositivos, tales como, el PLC encargado del control del proceso, el arrancador ATS 01, encargado de accionar sobre el motor que mueve la cinta transportadora y un sensor de posición electromecánico. La cinta transportadora se pondrá en movimiento cuando no se detecten bloques debajo de la pinza, y se detendrá al activarse un sensor de posición electromecánico con el contacto de los bloques, lo cual indica que los bloques pueden ser recogidos por la pinza para ser empaquetados.

Posición de bloques

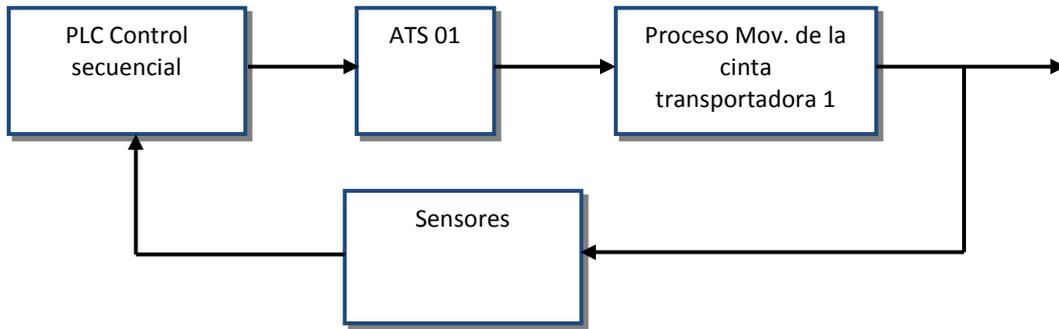


Fig.2.9 Diagrama de bloques del lazo de control de la cinta transportadora 1

Para el control de esta etapa, la cinta transportadora presenta dos estados explicados en el epígrafe 2.2.1, un estado donde la cinta transportadora estará en reposo y otro estado donde la cinta estará en movimiento. El diagrama de la MEFD de la cinta transportadora 1 encargada del transporte del paquete de bloques, se muestra en la figura 2.10.

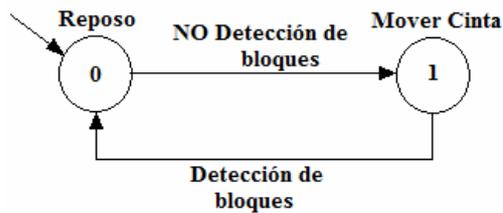


Fig.2.10 Diagrama en MEFD de la Cinta transportadora 1

Para el movimiento secuencial de la cinta transportadora del proceso de empaquetado la 5-tupla correspondiente a la MEFD queda representada de la siguiente manera:

$S = \{s_0, s_1\}$, conjunto finito de estados (2 estados)

$E = \{e_0, e_1\}$, conjunto finito de señales (2 señales)

$s_0 = s_0$, estado inicial o de partida

$F = \emptyset$, conjunto de estados finales, vacío en este caso

La función de transiciones $T: S \times E \rightarrow S$ se representa mediante la tabla de transiciones (tabla 2.5) correspondiente al proceso, donde, las filas representan los estados, las columnas representan las señales y el resto de celdas representan las transiciones.

Estado/Señal	NO Detección de bloques (I)	Detección de bloques (II)
0-Reposo	1-Mover cinta	
1-Mover cinta		2-Reposo

Tabla 2.5: Tabla de transiciones para la cinta transportadora 1

En la tabla de transiciones se muestra que la cinta comenzará a moverse cuando no haya detección de bloques y se detendrá cuando se active la señal de detección de bloques. En este caso la relación solo se muestra en la diagonal principal debido a que la MEFD es lineal.

2.2.6.2.2 Modelado del manipulador secuencial

Para el control del manipulador secuencial se propone el lazo cerrado mostrado en la figura 2.11. La realimentación del proceso viene determinada por una red de sensores (sensores inductivos de presencia, de posición electromecánicos y un encoder óptico), los cuales determinan las señales para las transiciones. En el caso del manipulador secuencial, son tres lazos de control, representados de forma general en uno solo, para controlar cada grado de libertad del manipulador, en este caso tres. Un primer lazo para controlar el movimiento vertical del manipulador, otro para el control del movimiento horizontal y un último lazo para el control del giro de la pinza.

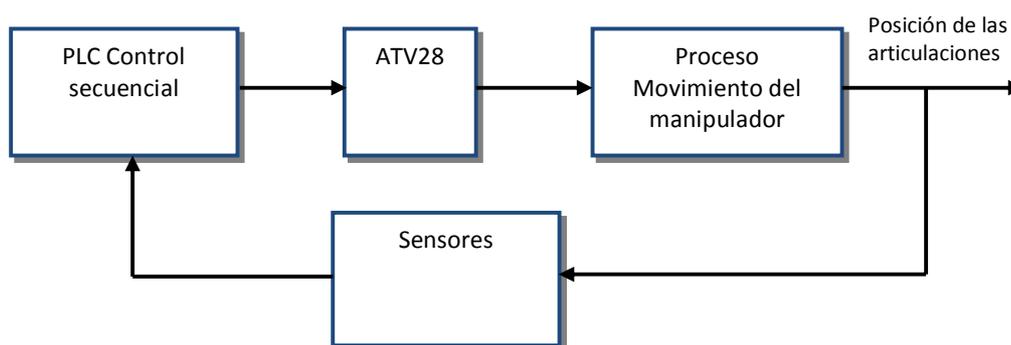
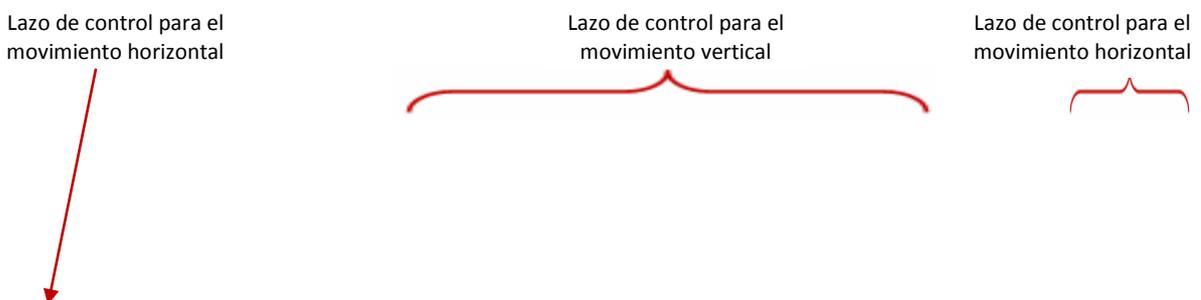


Fig.2.11 Diagrama de bloques del lazo de control del manipulador secuencial

En el PLC estará programado el control para lograr el comportamiento deseado del manipulador. Este control estará basado en el modelo del proceso en MEFD estudiado previamente, la MEFD tendrá 12 estados, teniendo en cuenta los 11 estados explicados en el epígrafe 2.2.2 más un estado de verificación. El paso de un estado a otro en el control secuencial, estará determinado por las diferentes señales provenientes de los sensores ubicados en el proceso, en su mayoría sensores presenciales que simplemente indican cuando el manipulador ha llegado a una posición determinada. Para el control secuencial también se realimenta la señal proveniente de un encoder óptico, ubicado en el motor del movimiento vertical, esto permite posicionar la pinza en el lugar indicado tanto para la carga como para la descarga de bloques. En cada uno los estados se realizarán las acciones indicadas para el correcto funcionamiento del manipulador. Estas acciones en su mayoría son realizadas por el variador de frecuencia ATV 28 encargado de manipular los motores.

El diagrama de la MEFD correspondiente al modelado del movimiento del manipulador se puede observar en la figura 2.12. Debe notarse que para comenzar el empaquetado de bloques, la señal *Detección de bloques* debe estar activa, lo que garantiza el sincronismo con la cinta transportadora encargada de llevar los bloques debajo de la pinza para su empaquetado. Debe notarse además, que se incluye un estado de verificación llamado Alarma (Estado B).



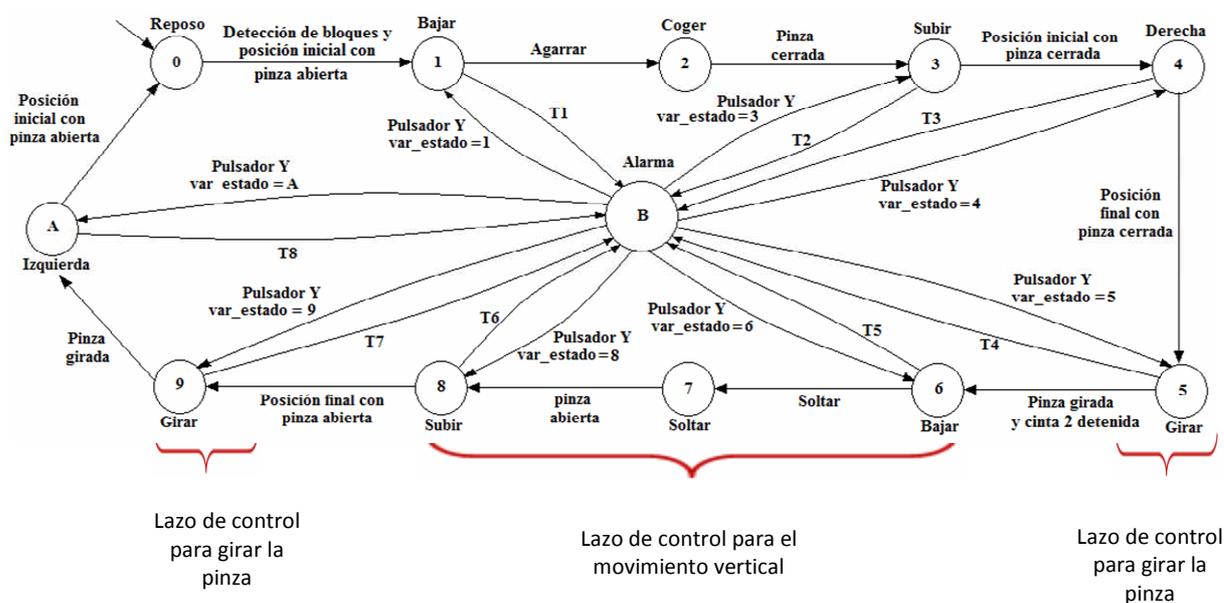


Fig.2.12 Diagrama en MEFD del manipulador secuencial

El diagrama anterior representa físicamente el modelo del movimiento secuencial del manipulador del proceso de empaquetado de bloques explicado anteriormente. En él se representan los 11 estados explicados en el epígrafe 2.2.2 que indican el funcionamiento correcto y las transiciones, las cuales dependen de las señales de la red de sensores del sistema. Al ser un proceso secuencial, la representación en MEFD es muy cómoda y natural, demostrándose las ventajas mencionadas en el epígrafe 1.3.1.3.

Para el movimiento secuencial del manipulador del proceso de empaquetado la 5-tupla correspondiente a la MEFD queda representada de la siguiente manera:

$$S = \{s_0, s_1, \dots, s_{11}\}, \quad \text{conjunto finito de estados (12 estados)}$$

$$E = \{e_0, e_1, \dots, e_{27}\}, \quad \text{conjunto finito de señales (28 señales)}$$

$$s_0 = s_0, \quad \text{estado inicial o de partida}$$

$$F = \emptyset, \quad \text{conjunto de estados finales, vacío en este caso}$$

La función de transiciones $T: S \times E \rightarrow S$ se representa mediante la tabla de transiciones (tabla 2.6) correspondiente al proceso, donde, las filas representan los estados, las columnas representan las señales y el resto de celdas representan las transiciones.

Estado/ Señal	Detección de bloques (I)	Agarrar (II)	Pinza cerrada (III)	Posición inicial con pinza cerrada (IV)	Posición final con pinza cerrada (V)	Pinza Girada (VI)	Soltar (VI)	Pinza abierta (VII)	Posición final con pinza abierta (IX)	Pinza Girada (X)	Posición inicial con pinza abierta (0)	T1 (XI)	T2 (XII)	T3 (XII)	T4 (XI V)	T5 (XV)	T6 (XV I)	T7 (XV II)	T8 (XV III)	Pulsador y var_estado=1 (XIX)	Pulsador y var_estado=3 (XX)	Pulsador y var_estado=4 (XXI)	Pulsador y var_estado=5 (XXII)	Pulsador y var_estado=6 (XXIII)	Pulsador y var_estado=8 (XXIV)	Pulsador y var_estado=9 (XXV)	Pulsador y var_estado=A (XXVI)	
0- Reposo	1- Bajar																											
1- Bajar		2- Agarrar										B- Alarma																
2- Cogeer			3- Subir																									
3- Subir				4- Derecha									B- Alarma															
4- Derecha					5- Girar									B- Alarma														
5- Girar						6- Bajar									B- Alarma													
6- Bajar							7- Soltar									B- Alarma												
7- Soltar								8- Subir																				
8- Subir									9- Girar								B- Alarma											
9- Girar										A- Izquierda								B- Alarma										
A- Izquierda											0- Reposo																	
B- Alarma																				1-Bajar	3-Subir	4- Derecha	5-Girar	6-Bajar	8-Subir	9-Girar	A- Izquierda	

Tabla 2.6: Tabla de transiciones para el manipulador secuencial

Como resultado de las relaciones observadas en la tabla de transiciones para el manipulador, se comprueba que la relación entre las señales y los estados está determinada por la relación lineal y secuencial existente. Esto garantiza que no existan deadlocks ni estados prohibidos que puedan afectar el funcionamiento del sistema. En el caso del funcionamiento correcto del manipulador intervienen las señales provenientes de los sensores. Por cada señal de entrada de los sensores que se activa en el proceso (columna 2 -13 de la tabla de transiciones) se detiene un motor y arranca otro, excepto con las señales Agarrar y Soltar que simplemente indican el cierre y apertura de la pinza de forma mecánica, con esto se garantiza que solo haya una articulación en movimiento a la vez. Esto se comprueba al observar que cada columna está asociada con un valor diferente de cero a una sola fila (estado), garantizando la secuencia lineal deseada. Este modelado permite una secuencia automatizada, además la tabla de transiciones también muestra como para completar el ciclo hay que transitar por todos los estados hasta llegar al estado inicial de Reposo, cumpliéndose con todo esto los requerimientos establecidos en el epígrafe 1.6.

Para la verificación del funcionamiento deseado del proceso, y para determinar averías en el mismo, se incluye un estado de verificación que tiene asociado una alarma. En caso de que los tiempos en los estados que intervienen en el movimiento del manipulador (estados 1, 3, 4, 5, 6, 8, 9 y A) sean superiores a los establecidos, se tomará como indicador de que los motores no están funcionando, por lo que se activa una alarma que indicará al operador la situación. De la columna 13 a la 20 se relacionan las variables temporales, que se toman como indicadores de afectaciones en los estados donde intervienen los motores, y están relacionadas con el estado de Alarma. Se volverá al estado en el que se encontraba el proceso cuando sea resuelto el problema por el cual el sistema se detuvo, con la acción de un operador sobre un pulsador, también se retornará al mismo estado en el que estaba el proceso, esto garantiza que si el sistema se detuvo cuando se transportaban los bloques, continúe el funcionamiento después de solucionado el problema, lo que se puede comprobar con la relación entre las columnas 21-28 con la última fila de la tabla.

2.2.6.2.3 Modelado de la cinta transportadora de paquetes de bloques

El lazo correspondiente a la cinta transportadora de paquetes de bloques, mostrado en la figura 2.13, funciona de manera similar a la cinta transportadora encargada de trasladar los bloques que serán empaquetados. La cinta transportadora de paquetes de bloques se

pondrá en movimiento cuando se haya completado el número de filas establecido, valor que se llevará en el programa del PLC y se detendrá al activarse un sensor de final de carrera con el contacto del paquete de bloques, lo cual indica que el paquete ha llegado a la posición deseada para ser recogido. Para el movimiento de la cinta se utiliza un arrancador que actuará sobre el motor.

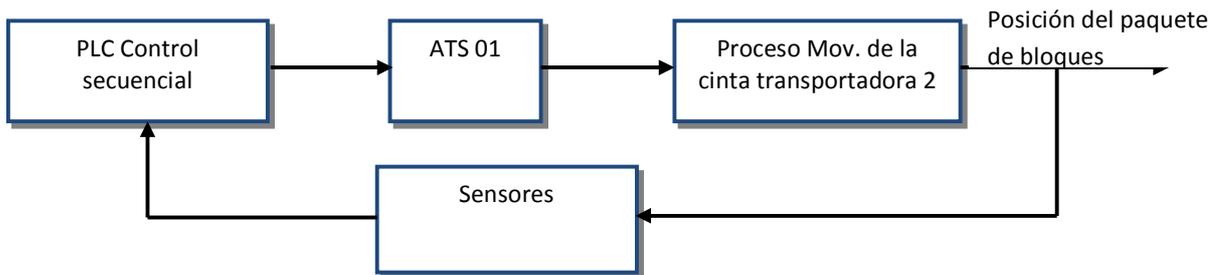


Fig.2.13 Diagrama de bloques del lazo de control de la cinta transportadora 2

Para el control de esta etapa, la cinta transportadora presenta dos estados explicados en el epígrafe 2.2.3, un estado donde la cinta transportadora estará en reposo, y otro estado donde la cinta estará en movimiento. El diagrama de la MEFD de la cinta transportadora 2, encargada del transporte del paquete de bloques, se muestra en la figura 2.14. En este caso la señal que activa el movimiento de la cinta indica que se ha depositado la quinta fila de bloques, lo que muestra que es necesario contar la cantidad de filas depositadas por el manipulador, en esto radica el sincronismo de la MEFD de la cinta transportadora del paquete de bloques con el manipulador secuencial.

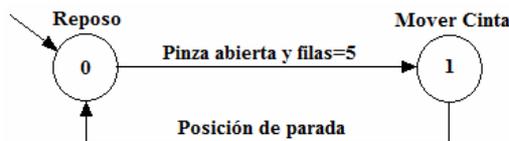


Fig.2.14 Diagrama en MEFD de la Cinta transportadora 2

Para el movimiento secuencial de la cinta transportadora del proceso de empaquetado la 5-tupla correspondiente a la MEFD queda representada de la siguiente manera:

$$S = \{s_0, s_1\}, \quad \text{conjunto finito de estados (2 estados)}$$

$$E = \{e_0, e_1\}, \quad \text{conjunto finito de señales (2 señales)}$$

$s_0 = s_0$, *estado inicial o de partida*

$F = \emptyset$, *conjunto de estados finales, vacío en este caso*

La función de transiciones $T: S \times E \rightarrow S$ se representa mediante la tabla de transiciones (tabla 2.7) correspondiente al proceso, donde, las filas representan los estados, las columnas representan las señales y el resto de celdas representan las transiciones.

Estado/Señal	Pinza abierta y Filas=5 (I)	Posición de parada (II)
0-Reposo	1-Mover cinta	
1-Mover cinta		2-Reposo

Tabla 2.7: Tabla de transiciones para la cinta transportadora 2

En la tabla de transiciones se muestra que la cinta comenzará a moverse cuando la pinza esté abierta y haya cinco filas sobre la cinta transportadora, ésta última señal se activa por programa con el conteo de las filas que se van depositando. La cinta transportadora se detendrá cuando se active la señal de posición de parada. En este caso la relación solo se muestra en la diagonal principal debido a que la MEFD es lineal.

2.3 Algoritmo y programación del Autómata Industrial

La programación del proceso, en general, se llevó a cabo confeccionando, en primer lugar, el algoritmo, utilizando para ello el análisis previo modelado con MEFD presentado en el epígrafe 2.2.6.2. El algoritmo para la programación del PLC se realizó utilizando el GRAFCET a partir del modelado del proceso. Solo son presentados los algoritmos correspondientes a las cintas transportadoras y del manipulador secuencial, por ser los más importantes para la automatización del proceso. Los restantes se diseñaron de manera similar a partir de las MEFD correspondientes.

2.3.1 Algoritmo para el control de la cinta transportadora de bloques para empaquetar

Utilizando la simbología mostrada en el anexo 2, se realizó el diagrama en GRAFCET para el control de la cinta transportadora 1, el cual se muestra en la figura 2.15. El diagrama GRAFCET se diseña partiendo del modelo en MEFD mostrado en el epígrafe 2.2.6.2.1, se

definen las etapas de acuerdo a los estados, y las transiciones de acuerdo a las señales utilizadas en el modelado. El diagrama GRAFCET tiene la particular diferencia, que en él se especifican y detallan cuidadosamente las acciones producidas en cada estado, así como las condiciones que provocan las transiciones y los eventos a que conllevan estas transiciones. Como se muestra en el diagrama GRAFCET de la cinta transportadora, la actuación sobre la cinta transportadora se realiza mediante un arrancador conectado de forma directa al PLC.

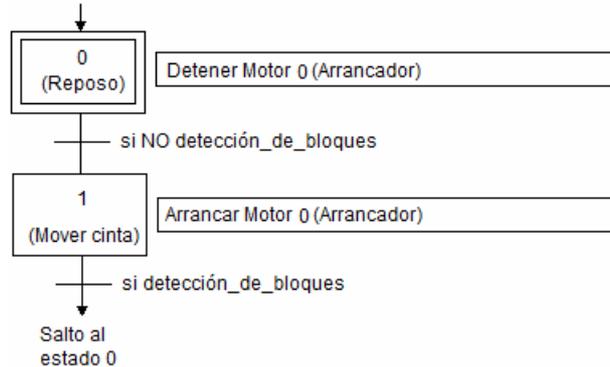


Fig.2.15 Diagrama GRAFCET del control para la cinta transportadora 1

El diagrama GRAFCET anterior tiene dos estados al igual que la MEFD de la cinta transportadora 1. Las acciones para el control del proceso, están definidas en cada una de las etapas. Estas etapas están unidas por las transiciones, que no son más que las señales provenientes de los sensores ubicados en el proceso lo que garantiza el paso de un estado a otro, y con ello el funcionamiento deseado.

2.3.2 Algoritmo para el control del manipulador secuencial

El diagrama GRAFCET correspondiente al movimiento del manipulador se muestra en la figura 2.16. Se diseña partiendo del modelo en MEFD mostrado en el epígrafe 2.2.6.2.2, por lo que tiene la misma cantidad de etapas que estados de la MEFD, y las transiciones coinciden con las señales utilizadas en el modelado. La actuación sobre el manipulador secuencial se realiza mediante los tres variadores de velocidad conectados de forma directa al PLC, los cuales a su vez actúan sobre los motores encargados del movimiento del manipulador secuencial. Tal como en la MEFD del manipulador, en caso de que el tiempo en que las etapas estén activas sea mayor que el tiempo establecido se da un salto a la etapa número 11, en la cual se activa una alarma indicando al operador de un mal funcionamiento. Para salir de esta etapa y retornar a la etapa en la que se encontraba la MEFD, es necesario

que el operador presione un pulsador que indica que el problema fue resuelto, también se pregunta por la variable de control de los estados, esto se muestra detalladamente en la programación del PLC.

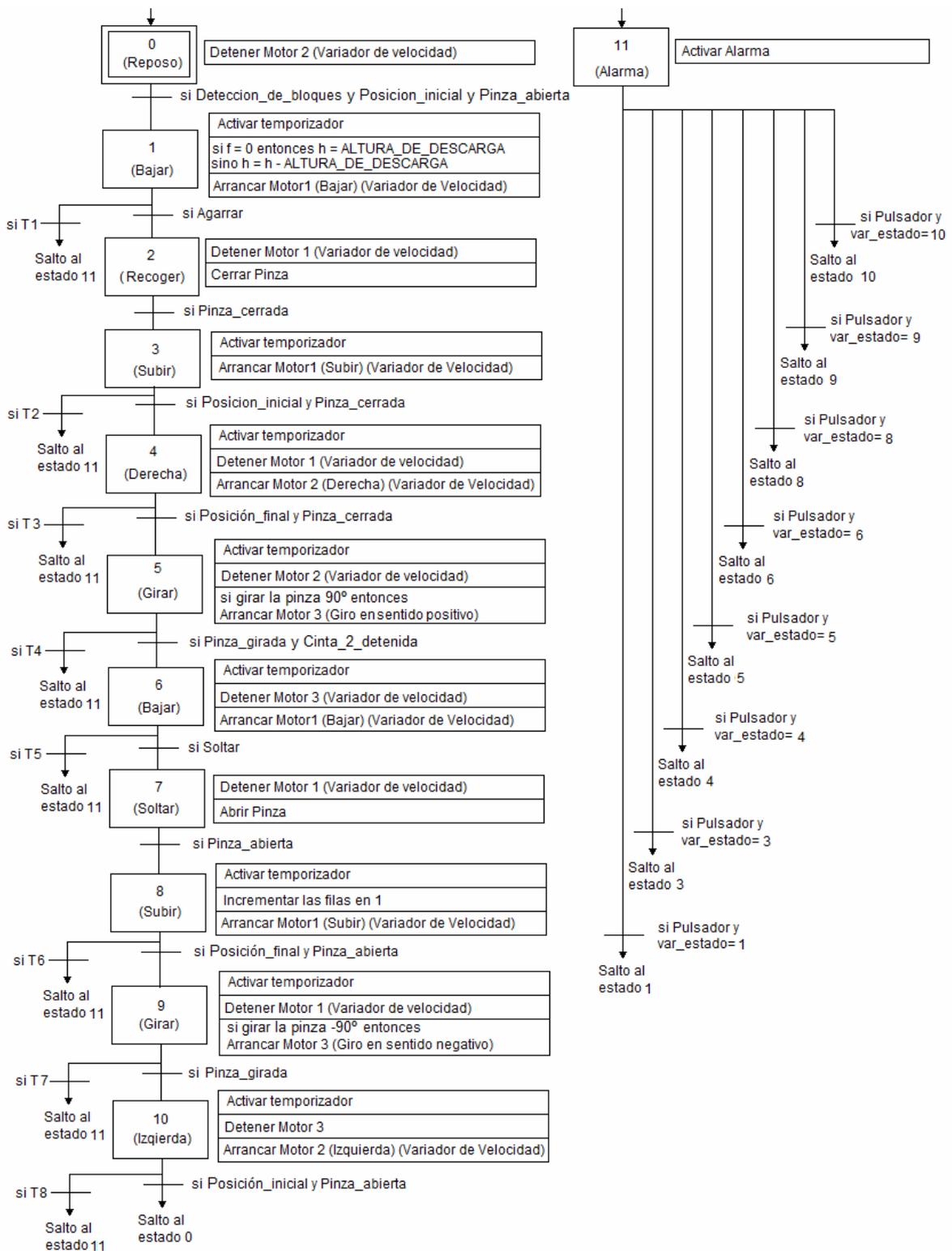


Fig.2.16 Diagrama GRAFCET del control para el manipulador secuencial

2.3.3 Algoritmo para el control de la cinta transportadora encargada del traslado del paquete de bloques

El diagrama en GRAFCET para el control de la cinta transportadora se muestra en la figura 2.17. Al igual que en los casos anteriores, el GRAFCET se diseña a partir del modelo en MEFD mostrado en el epígrafe 2.2.6.2.3.

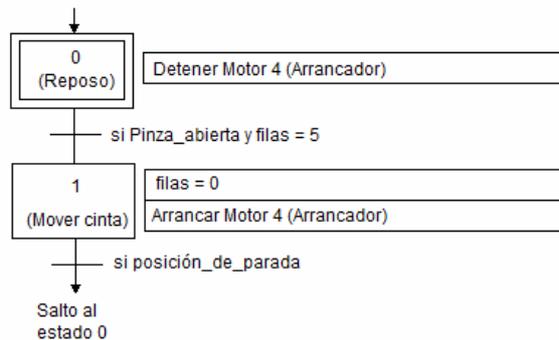


Fig.2.17 Diagrama GRAFCET del control para la cinta transportadora 2

El diagrama GRAFCET anterior tiene dos estados al igual que la MEFD de la cinta transportadora 2. La actuación sobre la cinta transportadora se realiza mediante un arrancador conectado de forma directa al autómatas.

2.3.4 Programación del PLC Industrial

El PLC seleccionado para el control pertenece a la familia MODICON M340, específicamente la CPU BMX P34 2010. El UnityPro es el software único de programación, puesta a punto y explotación de los autómatas MODICON M340, Premium, Quantum y coprocesadores Atrium. Este software recoge todas las ventajas de los softwares PL7 Pro y Concept y ofrece un conjunto completo de nuevas funciones para obtener más productividad y apertura hacia otros softwares. Dentro de sus principales ventajas se encuentra el simulador de PLC en PC, que permite depurar los programas para el control de procesos antes de su implementación. Mediante este software se realizó toda la programación del PLC teniendo en cuenta las entradas y salidas del proceso, así como el control secuencial modelado en el epígrafe 2.2.6. El lenguaje de programación utilizado fue Literal Estructurado (ST) debido a que genera un

programa más compacto de fácil comprensión, además de que es muy sencilla su modificación en caso de ampliación, cambio o mantenimiento.

Se tuvo en cuenta las protecciones necesarias en caso de una emergencia, así como los modos de funcionamiento del proceso. El funcionamiento manual/automático del proceso, se decide a través de una variable booleana (mode) declarada en el PLC correspondiente a un switch de entrada de dos posiciones. En modo manual el operador manipulará el proceso tomando sus propias decisiones, y en el modo automático el PLC controla el proceso siguiendo el control secuencial programado. Sin embargo, hay parámetros que se verifican tanto en modo automático como manual, por ejemplo, los sensores térmicos de los motores que se activan con el calentamiento del motor, esto provoca la activación de las alarmas correspondientes. Otro de los parámetros que se verifica constantemente es la acción realizada por un operador sobre un pulsador de enclavamiento en caso de alguna emergencia, lo cual detiene inmediatamente todo el proceso y la activación de una alarma tanto lumínica como sonora que indica fallo en el sistema.

Otro de los parámetros a tener en cuenta es el reset, el cual se activa cuando se acciona un pulsador sin enclavamiento en el sistema, con lo cual se logra llevar a cero todas las variables que intervienen en el control del proceso, dejando al sistema listo para iniciar un nuevo ciclo de empaquetado de bloques. En el programa también se declararon tres variables de salida correspondientes a tres balizas de iluminación:

- *luz_encendido*: Para indicar que el sistema está encendido.
- *luz_operacion*: Para indicar que ya está trabajando (notar que el sistema puede estar encendido y no trabajando).
- *luz_emergencia*: En caso de alarma.

Estas variables tienen especial importancia para el personal operativo, ya que indican en cada momento el estado en el que se encuentra el proceso.

La conexión de los sensores inductivos y de los finales de carrera se realiza de forma directa o a través de un bornero a las entradas digitales del autómatas. Esto es posible debido a que la señal de salida que generan es estándar, cumpliendo con la misma normativa y especificaciones.

Como el sistema está basado en MEFD se garantiza, que ante una parada de emergencia del sistema, todos los procesos comiencen su ejecución hasta completar el ciclo correspondiente, siempre que sea posible, en el mismo estado en que estaban antes de pulsar el botón de emergencia o de ocurrir un fallo. De este hecho, se infiere que las variables que controlan los estados tanto en el proceso del manipulador como en el de la cinta transportadora, deben ser variables remanentes que garanticen que ante cualquier emergencia o fallo, como puede ser un corte de alimentación inesperado, el sistema tenga la capacidad de recuperarse (tolerancia a fallos) y comenzar en el mismo punto en el que estaba antes de ocurrir el fallo.

2.3.4.1 Programación del movimiento de las cintas transportadoras

Para el control del movimiento de las cintas transportadoras se desarrolló un programa en ST siguiendo los algoritmos propuestos en los epígrafes 2.3.1 y 2.3.3. En la cinta transportadora 1 se utiliza solamente la señal del final de carrera que detecta la presencia de bloques debajo de la pinza. En la cinta transportadora 2 se utilizan dos señales provenientes de sensores ubicados en el proceso. Para la actuación sobre los motores que garantizan el movimiento de cada cinta transportadora, se utilizan dos arrancadores suaves para motores asíncronos del mismo modelo (uno para cada una), cumpliéndose uno de los requerimientos establecidos en el epígrafe 1.6, el Altistart 01. Por la configuración adoptada en cada arrancador la entrada lógica a controlar es LI2 en ambos, garantizando la parada y el arranque de los motores con el cambio de estado de esta entrada.

Partiendo del diagrama GRAFCET se desarrolla el programa mostrado en el anexo 7, donde la forma específica de realizar las transiciones es la siguiente:

Si estando en el estado $s_k \in S$ llega una señal $e_p \in E$ (o el conjunto de señales e_p, \dots, e_{p+i}), entonces:

1. Se pasa al estado $s_t \in S$ (s_k y s_t no tienen necesariamente que ser estados consecutivo).
2. Estando en el estado s_t se realiza la acción a_q (o la secuencia de acciones a_q, \dots, a_{q+r}) correspondiente(s).

A continuación se muestran los segmentos más importantes del programa desarrollado para el control de la cinta transportadora 1. En el programa están bien definidos los dos estados que fueron diseñados en el modelo de MEFD y luego en el GRAFCET. En el primer estado la cinta está en reposo y en el segundo la cinta está en movimiento, dependiendo del estado de la señal que detecta la presencia de bloques debajo de la pinza.

(* estado de reposo de la cinta *)

if estado_cinta1 = 0 then

motor0_LI1:=false; (* detener motor *)

Señal de control al arrancador

if final_carrera_deteccion_bloques then

estado_cinta1 := 1;

end_if;

end_if;

(* mover la cinta *)

if estado_cinta1 = 1 then

motor0_LI1:=true; (* arrancar motor *)

Señal de control al arrancador

if not final_carrera_deteccion_bloques then

estado_cinta := 0;

end_if;

end_if;

2.3.4.2 Programación del movimiento del manipulador

Para el control de cada motor del manipulador secuencial se utiliza un variador como parte de los requerimientos establecidos en el epígrafe 1.6, el ALTIVAR 28, cuyas características generales se mostraron en el epígrafe 2.2.4, siendo las entradas a controlar en el variador las siguientes según la configuración escogida.

- *LI1*: (Parada) En su estado en OFF detendrá el motor quitándole la alimentación.
- *LI2*: (Adelante) Garantiza que el motor vaya hacia adelante, es decir, en un sentido cuando esta entrada esté en ON (Entrada de valor opuesto a *LI3*).
- *LI3*: (Atrás) Garantiza que el motor se mueva en sentido contrario cuando esta entrada esté activa (Entrada de valor opuesto a *LI2*).
- *LI4*: (PS2) Determina las dos velocidades del motor, es decir, en ON tendrá una velocidad mínima, y en OFF tendrá una velocidad máxima prefijada, este análisis se muestra en la tabla 2.8.

Como se mencionó anteriormente para el control del manipulador secuencial se han utilizado dos velocidades. También se han configurado los sentidos de giro del motor y la parada. La

velocidad mínima seleccionada es cero y la velocidad máxima será la adecuada para el movimiento del transporte de los bloques establecida por parámetros de la planta (se debe notar que esta máxima no es la velocidad máxima a la que puede ir el motor sino la máxima prefijada de acuerdo a las limitación y condiciones del proceso).

Bit de Parada	Bit para el sentido hacia "Adelante"	Bit para el sentido hacia "Atrás"	Bit de Velocidad (0=mín, 1=máx)	Acción del motor
0	-	-	-	Parada del motor
1	1	0	0	Movimiento hacia adelante (vel=mín)
1	1	0	1	Movimiento hacia adelante (vel=máx)
1	0	1	0	Movimiento hacia atrás (vel=mín)
1	0	1	1	Movimiento hacia atrás (vel=max)

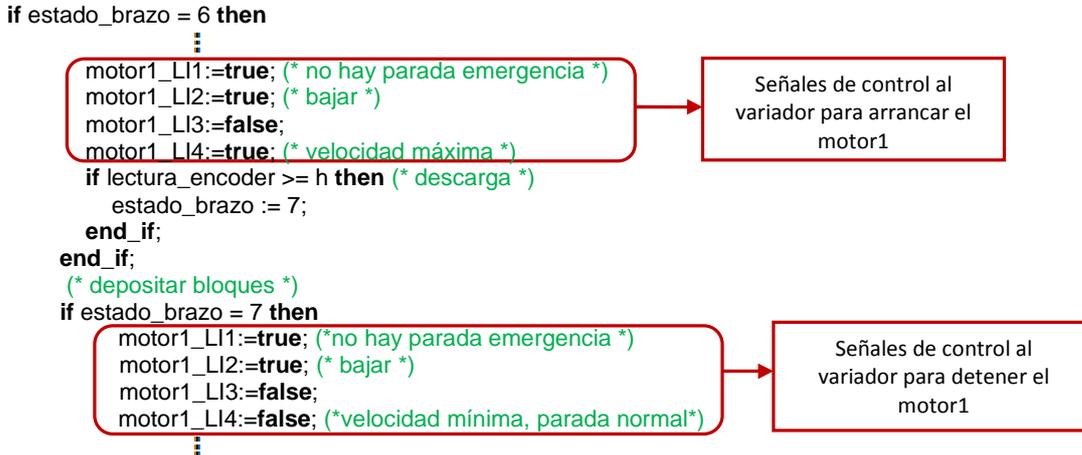
Tabla 2.8: Configuraciones para el control del motor

La parada normal de cada motor se realiza dándole velocidad cero y la parada correspondiente a la entrada LI1 se realiza en caso de emergencia, ya que es una parada mucho más rápida. Para el control de cada variador se seleccionaron cuatro salidas digitales del autómatas que tomarán los valores de la tabla anterior según la acción que se desee realizar sobre el motor.

Para el control del movimiento del manipulador se desarrolló un programa en ST siguiendo el algoritmo propuesto en el epígrafe 2.3.2. Tanto en el modelo en MEFD como en el GRAFCET el sistema transita por 11 estados para completar un ciclo completo del manipulador secuencial, dependiendo del paso de una etapa a otra de las señales provenientes de los sensores ubicados en el proceso y de otras generadas en el programa. Siguiendo el algoritmo también se programa la etapa 11 de Alarma para la verificación en caso de que no se cumpla con los tiempos establecidos. Partiendo del diagrama GRAFCET se desarrolla el programa mostrado en el anexo 7, donde la forma de realizar las transiciones es igual a la mostrada en el epígrafe 2.3.2.

Uno de los aspectos importantes en el proceso de empaquetado de bloques es que es necesario controlar la altura, tanto para la carga como para la descarga de los bloques. Para realizar este control es necesario medir la posición de la pinza en el eje vertical y realimentar esta señal. Esto se realiza mediante un encoder incremental, cuya salida estará conectada a un módulo de conteo del autómatas. Esta señal es correspondiente a una altura real de la

pinza y se compara con el SP en cada caso, cuando la señal medida es igual a la deseada entonces se da la orden de detener el motor correspondiente al movimiento vertical de la pinza. Un ejemplo de esto se muestra en el fragmento de programa a continuación.



Este fragmento corresponde al estado 6 del GRAFCET (epígrafe 2.3.2), donde se procede a bajar la pinza cerrada que sujeta los bloques hasta una altura determinada (h). En el estado 0 del programa, se modifica el valor del SP de esta altura utilizando la ecuación 4, en dependencia de la cantidad de filas agrupadas, cumpliéndose uno de los requerimientos establecidos en el epígrafe 1.6:

$$h = h - ALTURA_{DEL_BLOQUE} \tag{4}$$

Donde $ALTURA_{DEL_BLOQUE}$ es una variable de valor constante, como se muestra en la figura 2.18. Las variables y constantes más significativas utilizadas en el programa se muestran en la tabla 2.9.

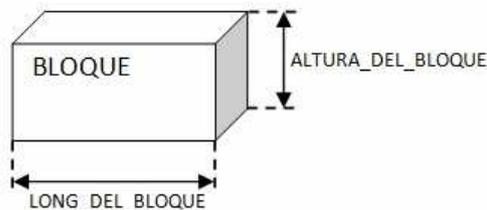


Fig.2.18 Dimensiones del bloque

En el estado 6, después de realizar otras acciones, se arranca el motor del movimiento vertical (motor1), el cual estará funcionando hasta que se pase al estado 7, donde se detendrá el motor1 y se abrirá la pinza. La transición del estado 6 al 7 solo se realizará cuando se cumpla la condición de la transición, es decir, cuando la altura medida por el encoder sea mayor o igual que la altura deseada (lectura_encoder >= h).

Variables (minúsculas)	Significado de las Variables
c	Contador de la cantidad de bloques agrupados
f	Contador de filas de bloques
h	Controla la distancia vertical (altura) que debe recorrer el brazo en dependencia de la cantidad de filas que hay hasta el momento y de la propia altura del bloque. La mayor distancia vertical (h) se conoce inicialmente y se debe fijar a priori en el programa, a medida que se van descargando filas se recalcula h.
Constantes (mayúsculas)	Significado de las constantes
CANTIDAD_DE_FILAS	Cantidad de filas máxima en la zona de descarga (5 en este caso).
CANTIDAD_DE_BLOQUES_EN_UNA_FILA	Cantidad de bloques que tiene una fila.
ALTURA_DEL_BLOQUE	Altura del bloque, como todos los bloques están ordenados en una fila es la misma que la altura de la fila y por tanto, es la cantidad negativa en la que debe variar h cada vez que se descarga una fila.
ALTURA_DE_CARGA	Distancia vertical inicial o máxima de carga que debe recorrer el brazo cuando va a cargar una fila de bloques.
ALTURA_DE_DESCARGA	Distancia vertical inicial o máxima de descarga que debe recorrer el brazo cuando va a descargar una fila de bloques.

Tabla 2.9: Principales variables y constantes utilizadas en el programa

En las etapas donde se realiza el control de motores a través de los variadores, se controla el tiempo de activación de la etapa. Se utilizó un temporizador TON, el cual activa su salida cuando se alcanza el tiempo programado. Con esto se garantiza, que si el tiempo de activación de la etapa, es mayor que el tiempo establecido para el movimiento del manipulador en esa etapa, es porque ocurre algún desperfecto técnico, principalmente se realiza para el cuidado de los motores. Para esto se utilizó un temporizador que se activará con la activación del estado, como se muestra en la siguiente sección de programa, y se desactivará si se cumplen las transiciones correctas a través de la activación de los sensores.

```

Enable := true; (* activar temporizador *)
      ⋮
cambio_estado_brazo := estado_brazo; (* variable en la que se guarda el estado actual *)
if FBI_0.Q then (* pasó el tiempo establecido *)
    estado_brazo := 11; (* alarma *)
end_if;
      ⋮
    
```

En la programación de esos estados, la transición al estado 11 sucede cuando se activa la salida del temporizador, la cual indica en este caso que ya se ha superado el tiempo establecido para el movimiento del manipulador.

2.4 Sistema de supervisión para el proceso de empaquetado de bloques

En el epígrafe 1.4 se hizo una breve referencia a los sistemas de supervisión y control, y su importancia para el control de procesos. Un sistema de supervisión y control permite que un operador ubicado en una estación central, incluso a grandes distancias de la ubicación de los procesos industriales, pueda hacer ajustes o cambios en los controladores locales de los procesos. En este caso es importante conocer el estado de los interruptores de seguridad de un sistema, monitorear el estado de las alarmas del proceso y obtener información de las variables del proceso involucradas, tanto para realizar análisis y comparativas estadísticas como para la toma de decisiones.

En el proceso de empaquetado de bloques de hormigón es necesario tener un sistema de alarmas como parte de los requerimientos establecidos en el epígrafe 1.6, debido a la peligrosidad del proceso por los grandes volúmenes de carga que se manipulan. Además, es necesario conocer la cantidad de bloques y lotes de bloques que se producen, que depende del tipo de bloque. Esto garantiza que al hacer un balance de la producción, deben coincidir la cantidad de materiales empleados con una determinada cantidad de bloques producidos, es decir, almacenamiento de datos y análisis estadísticos como se mencionó anteriormente, que evidentemente inciden en el control, eficiencia y calidad de la producción.

En caso de que se detenga un motor por calentamiento se activan las alarmas, pero además, se muestran en el sistema de supervisión. En caso de que esto suceda se puede poner el proceso en funcionamiento manual, según crea el operador. También se muestra el funcionamiento general del proceso, mediante el cual, el operador puede observar el movimiento del manipulador y de las cintas transportadoras.

El sistema de supervisión y control se desarrolló utilizando el software WinCC de Siemens. La ventana principal para la supervisión del proceso se muestra en la figura 2.19.

En la pantalla se podrán observar las diferentes posiciones del manipulador correspondiente al modelo MEFD, simulando de esta manera el comportamiento real del proceso. Se visualizan además, las señales provenientes de los sensores y otras variables del programa.

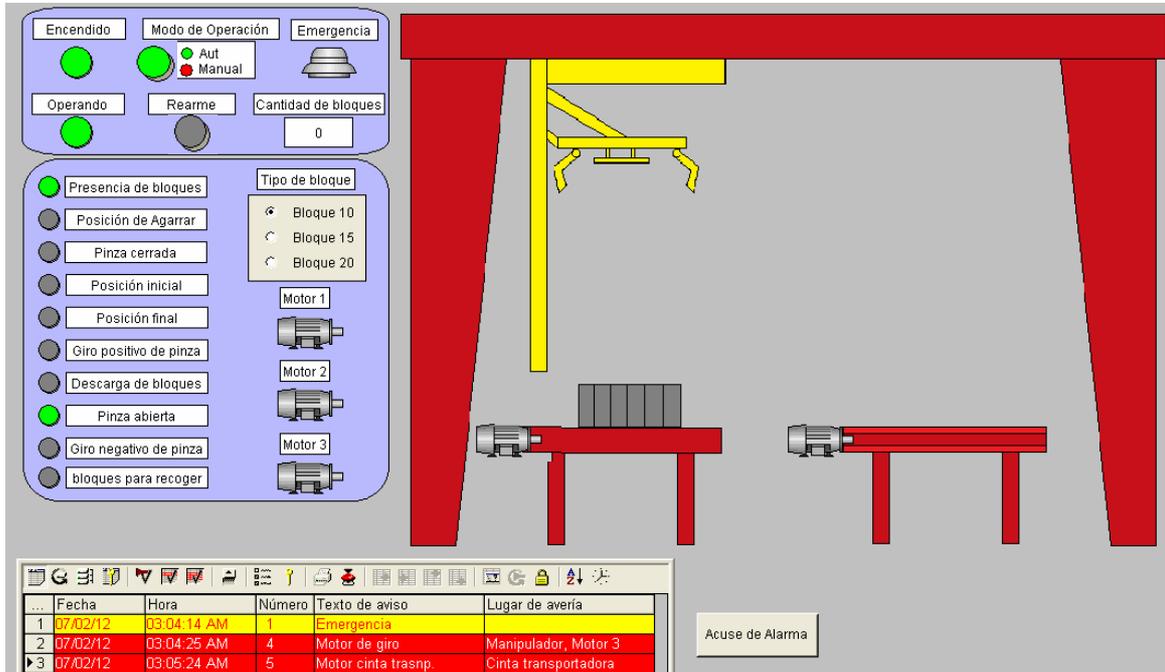


Fig.2.19 Supervisión del proceso de empaquetado de bloques.

La pantalla para la supervisión del proceso tiene un panel de control, mostrado en la figura 2.20, donde se muestran las variables que intervienen en el proceso. Estas variables son en su mayoría variables provenientes de los sensores. Se podrán modificar desde la pantalla la forma en que opera el sistema, si es en manual o automático, se podrá escoger el tipo de bloque que se producirá entre los tres tipos que se construyen, además, se incluye un botón de rearme para inicializar las condiciones del proceso cuando sea necesario. Se indica si el sistema está energizado y si está funcionando³, y también se muestra el estado de los motores en caso de que estén funcionando o no.

³ Notar que el sistema puede estar encendido y no estar en modo de operación

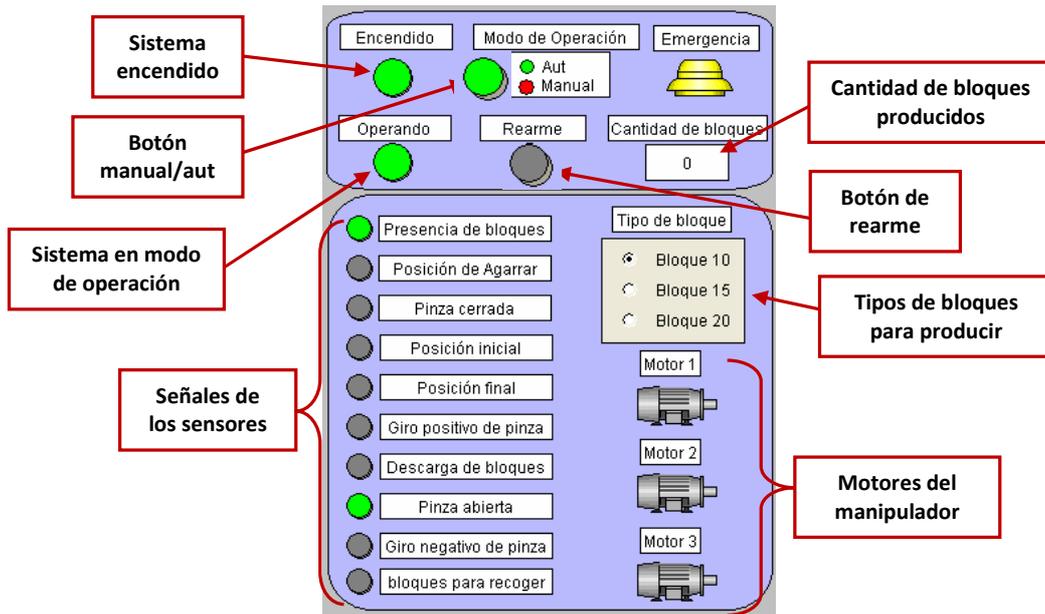


Fig.2.20 Panel de control en el SCADA

2.4.1 Niveles de acceso

En el SCADA se hace una verificación del control de acceso, esto se debe a que este SCADA corresponde a una sección de la planta. La seguridad del SCADA cuenta con dos niveles de acceso para determinados usuarios, en caso de que se desee manipular alguna variable. Estos niveles son:

- *Operador*: Solo puede modificar los parámetros a los cuales esté autorizado, por defecto el SCADA arranca en esta sesión.
- *Administrador*: Como indica su nombre tiene todos los permisos. Para activar esta sesión se deben pulsar simultáneamente las teclas *Alt + I* apareciendo la ventana que se muestra en la figura 2.21.

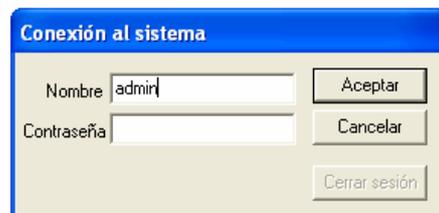


Fig.2.21 Ventana de acceso de usuario

2.4.2 Gestión de Alarmas

Las alarmas o avisos son de gran importancia ya que garantizan la comunicación necesaria entre el operario y el sistema, y además ayudan a la toma de decisiones. A través de ellas se informa acerca de los estados de funcionamiento y de fallo que se producen durante el proceso. Sirven para poder detectar con antelación situaciones críticas y para evitar períodos de inactividad. Al configurar las alarmas se definen qué eventos activarán las mismas, y si estarán visibles hasta que alguien las acuse.

En el panel de control se incluye una alarma intermitente que se activa con la señal de emergencia proveniente del PLC. Esta alarma también se gestiona por el SCADA, mediante la ventana que se muestra en la figura 2.22, lo que se observa es el histórico de la alarma diseñada, es decir, la fecha y la hora en que se activó, si fueron acusadas o no, y si se eliminaron estas alarmas. También se incluyen las alarmas correspondientes al disparo de los motores por calentamiento, lo que provoca la detención de los motores tanto en el proceso como la visualización en el sistema de supervisión, mostrándose además, el lugar de la avería y un texto de aviso como se muestra en la figura 2.23.



Fig.2.22 Alarma intermitente y ventana de gestión de alarmas

Para la gestión de las alarmas se implementó un sistema de acuse de recibo obligatorio mediante la activación de una variable, con lo cual se garantiza que el encargado de la instalación o el que tenga los permisos para hacer el acuse de recibo, lea el aviso aunque la alarma ya se haya desactivado. Se debe notar que cuando se realiza el acuse de recibo, no desaparece la alarma de la tabla de gestión de alarmas en caso de que ésta permanezca activa. En caso de que se active la alarma se debe pulsar el botón para realizar el acuse de recibo, figura 2.23.



Fig.2.23 Alarmas

Si el operario que está supervisando el proceso no tiene los permisos para hacer el acuse de la alarma o intenta modificar algunas de las variables del proceso por el SCADA, saldrá una ventana mostrando el siguiente aviso.



Fig.2.24 Mensaje de aviso

2.4.3 Mímico del proceso

Para realizar el mímico del proceso, se diseña de manera tal que el manipulador pase por los estados correspondientes a la MEFD del manipulador secuencial, mostrado en el epígrafe 2.2.6.2.2, coincidiendo en este caso el modelado general. La diferencia consiste en que cada estado de la MEFD para la programación tiene asociada otras máquinas de estados, las cuales determinan el posicionamiento del manipulador en los ejes de coordenadas x e y de la ventana del SCADA, así como la visibilidad o invisibilidad de los dibujos. Para ilustrar lo explicado se muestra en la figura 2.25 la MEFD correspondiente al mímico en la posición inicial. Se diseñó una pinza abierta y una cerrada. En la posición inicial después de recoger los bloques la pinza está cerrada, y está abierta cuando regresa de haber depositado los bloques en la cinta transportadora

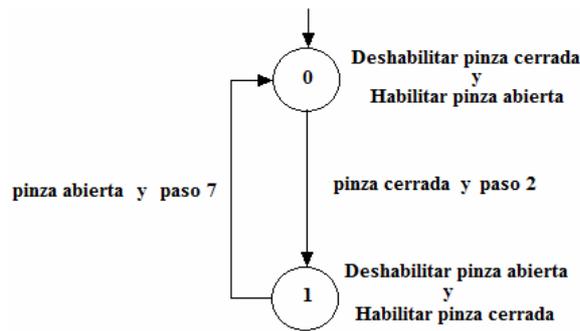


Fig.2.25 MEFD correspondiente a la posición inicial

La 5-tupla correspondiente a la MEFD queda representada de la siguiente manera:

$S = \{s_0, s_1\}$, conjunto finito de estados (2 estados)

$E = \{e_0, e_1\}$, conjunto finito de señales (2 señales)

$s_0 = s_0$, estado inicial o de partida

$F = \emptyset$, conjunto de estados finales, vacío en este caso

La función de transiciones $T: S \times E \rightarrow S$ se representa mediante la siguiente tabla de transiciones.

Estado/Señal	Pinza cerrada y paso 2 (I)	Pinza abierta y paso 7 (II)
0-Desahabilitar pinza cerrada y Habilitar pinza abierta	1- Desahabilitar pinza abierta y Habilitar pinza cerrada	
1- Desahabilitar pinza abierta y Habilitar pinza cerrada		2- Desahabilitar pinza cerrada y Habilitar pinza abierta

Tabla 2.10: Tabla de transiciones para la cinta transportadora

El resto de MEFD se realizó de la misma manera, siendo esta la más compleja por converger en una misma posición cuatro estados, representados de manera simplificada en dos. Una breve descripción del funcionamiento del mímico se realiza a continuación.

Al activarse el sensor de detección de bloques, se visualiza en el SCADA, en color verde el led correspondiente a esta señal, inmediatamente aparecen los bloques debajo de la pinza, luego, al activarse la señal de agarrar manejada por el autómata, la pinza se pone en posición de agarrar los bloques, pasando luego a la posición inicial, depositándolos después

en la cinta transportadora. En la cinta transportadora irán apareciendo las filas de bloque una encima de la otra hasta que haya 5 filas, cuando esto suceda volverá a comenzar el ciclo. En la figura 2.26 se muestra parte de la secuencia realizada por el SCADA para describir la secuencia del manipulador.

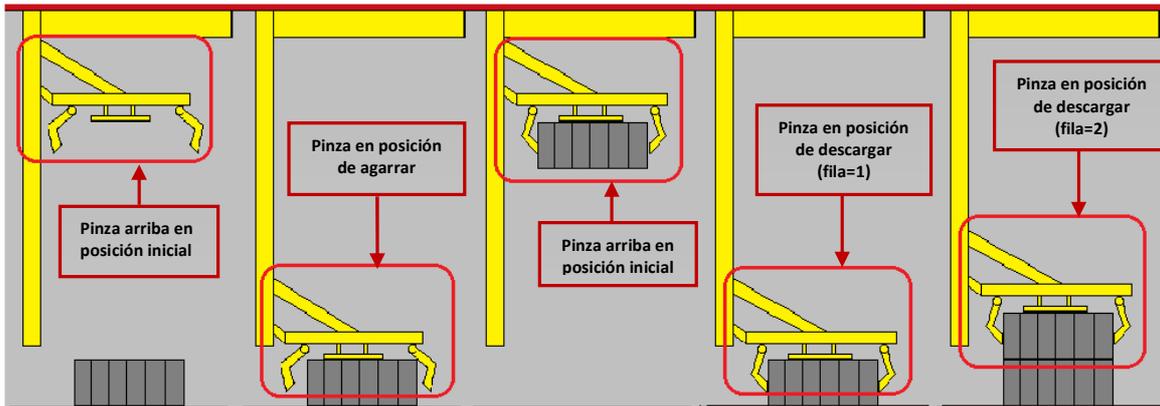


Fig.2.26 Secuencia descrita por el manipulador vista desde el SCADA

2.4.4 Comunicación para la supervisión del proceso

Para la supervisión del proceso mediante el SCADA es necesario comunicar éste con el PLC. Como el PLC pertenece a la firma Schneider y el SCADA desarrollado se implementó en WinCC perteneciente a la firma Siemens, es necesario utilizar buses de campo y un servidor OPC, que garanticen de manera fiable el intercambio de información.

Los buses de campo son ampliamente utilizados en la industria para el control y la supervisión de procesos, debido principalmente al gran volumen de datos que manejan y a su alta velocidad de transmisión. Para la comunicación entre el proceso y el SCADA el bus de campo utilizado es Modbus, que no es más que un protocolo de comunicación abierto de muy amplio uso en la industria. Entre otras ventajas, destaca el hecho de ser un protocolo de comunicación gratuito que los fabricantes utilizan para construir sus equipos sin tener que pagar regalías. Modbus se ha convertido en un protocolo de comunicaciones estándar en la industria, y ahora es el medio más generalizado de la conexión de dispositivos electrónicos industriales. Es un protocolo que proporciona la comunicación cliente/servidor entre dispositivos conectados en diferentes tipos de buses o redes. No especifica estrictamente el tipo de red de comunicaciones a utilizar, por lo que se puede implementar sobre redes

basadas en Ethernet, RS-485, RS-232 [53], [54]. Las versiones del protocolo Modbus existen para las líneas serie (Modbus RTU y ASCII, Modbus) y Ethernet (Modbus TCP).

El PLC BMX P34 2010 de Schneider tiene un puerto serie integrado que soporta protocolo Modbus (ASCII y RTU), como se muestra en la figura 2.27.

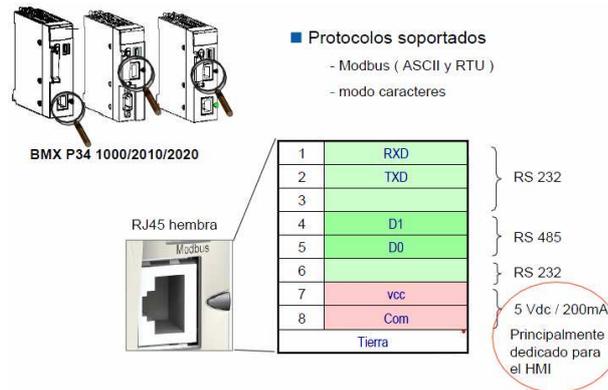


Fig.2.27 Protocolos de comunicación soportados por el PLC

2.4.4.1 Principio de las comunicaciones industriales

Modbus es un protocolo de diálogo que crea una estructura jerárquica, un dispositivo maestro y varios dispositivos esclavos, un ejemplo de esto se muestra en la figura 2.28, que permite que el dispositivo maestro interroge a uno o varios dispositivos esclavos inteligentes. Un enlace de múltiples derivaciones efectúa la conexión entre el maestro y los esclavos [53], [54].

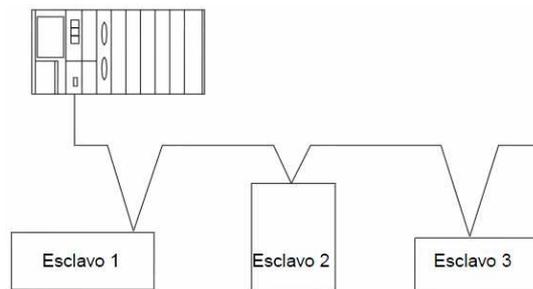


Fig.2.28 Comunicación entre el dispositivo maestro y los dispositivos esclavos

La secuencia básica en las comunicaciones Modbus consiste siempre en una trama de pregunta, seguida de su correspondiente trama de respuesta [45], [53], [54]:

- *Pregunta:* Con el código de función que indica al esclavo qué operación ha de realizar, y los bytes necesarios (datos, comprobación) para su ejecución.
- *Respuesta:* Con la confirmación o datos resultantes de la ejecución de la función.

Hay dos tipos de diálogos posibles entre los dispositivos maestros y esclavos [53]:

- El dispositivo maestro se comunica con el dispositivo esclavo y espera una respuesta.
- El dispositivo maestro se comunica con todos los dispositivos esclavos y no espera una respuesta (difusión de mensaje).

Las comunicaciones Modbus se pueden realizar, en modo ASCII o en modo RTU. En modo ASCII los bytes se envían codificados en ASCII, es decir, que por cada byte a transmitir se envían dos caracteres ASCII (2 bytes) con su representación hexadecimal (esto permite leer las tramas con un simple editor de texto). En modo RTU se envían en binario [45], [53], [54].

El dispositivo maestro inicia y administra los intercambios de comunicaciones, repite la pregunta si hay un intercambio incorrecto, y declara ausente al dispositivo esclavo interrogado si no recibe ninguna respuesta dentro de un plazo determinado. Solamente un dispositivo puede transmitir en la línea en un determinado momento. Ningún dispositivo esclavo puede enviar un mensaje a menos que reciba una invitación para ello. No está permitida la comunicación lateral (entre esclavos). Por lo tanto, el software de comunicaciones del dispositivo maestro debe estar diseñado para interrogar a un esclavo y devolver la información recibida a otro esclavo. Para la simulación del SCADA se utilizó un simulador de Modbus cuya pantalla principal se muestra en la figura 2.29.

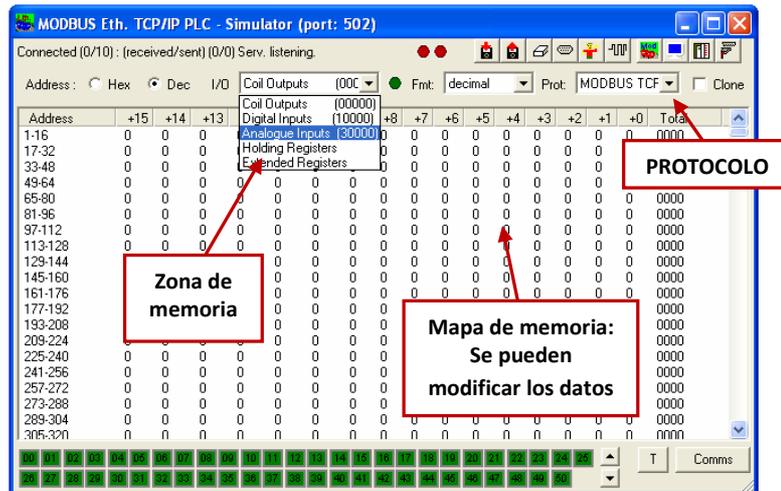


Fig.2.29 Simulador de Modbus

Los datos del simulador se pueden modificar en el mapa de memoria como si fueran los datos que vienen del PLC, mostrándose los cambios en el SCADA y viceversa.

2.4.5 Servidor Matrikon-OPC para la gestión de la comunicación

Como se analizó en el epígrafe 1.4.2, un servidor OPC es un estándar de comunicación en el campo del control y supervisión de procesos industriales que ofrece un interface común para la comunicación. La comunicación OPC se realiza a través de una arquitectura cliente/servidor. El servidor OPC es la fuente de datos y cualquier aplicación basada en OPC puede acceder a dicho servidor para leer/escribir cualquier variable que ofrezca el servidor. Es una solución abierta y flexible al clásico problema de los drivers propietarios. Es por ello que prácticamente los mayores fabricantes de sistemas de control, instrumentación y de procesos han incluido OPC en sus productos.

El servidor OPC utilizado como enlace entre el sistema de supervisión y el autómatas con Modbus es un Matrikon OPC Server, el cual es un Servidor OPC DA, es decir, basado en la especificación de Acceso de Datos para la transmisión en tiempo real. La configuración de la comunicación del servidor OPC con el PLC con Modbus se muestra en la figura 2.30. El SCADA y el PLC funcionarán como clientes del servidor OPC.

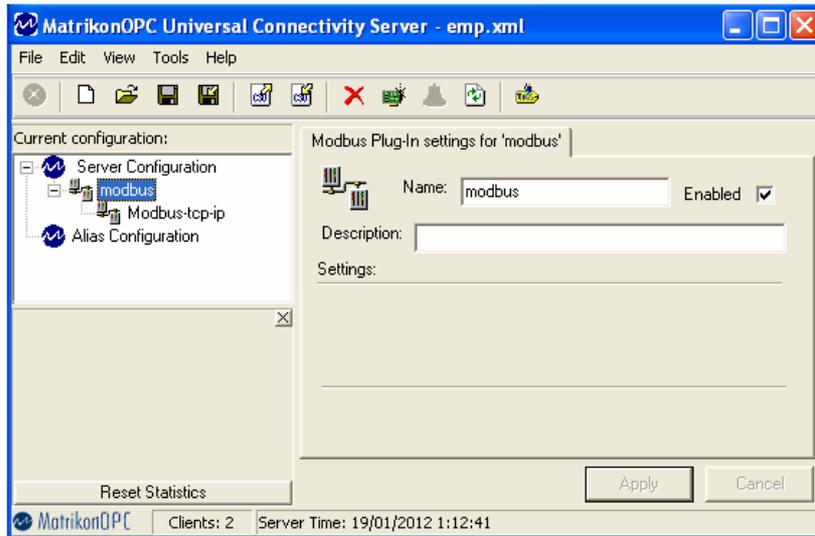


Fig.2.30 Configuración del servidor OPC para la comunicación

La configuración del servidor se puede verificar mediante el explorador del Matrikon OPC Server, el cual se muestra en la figura 2.31, donde se puede ver que las variables configuradas en el servidor están conectadas al simulador del Modbus, coincidiendo los valores de las variables.

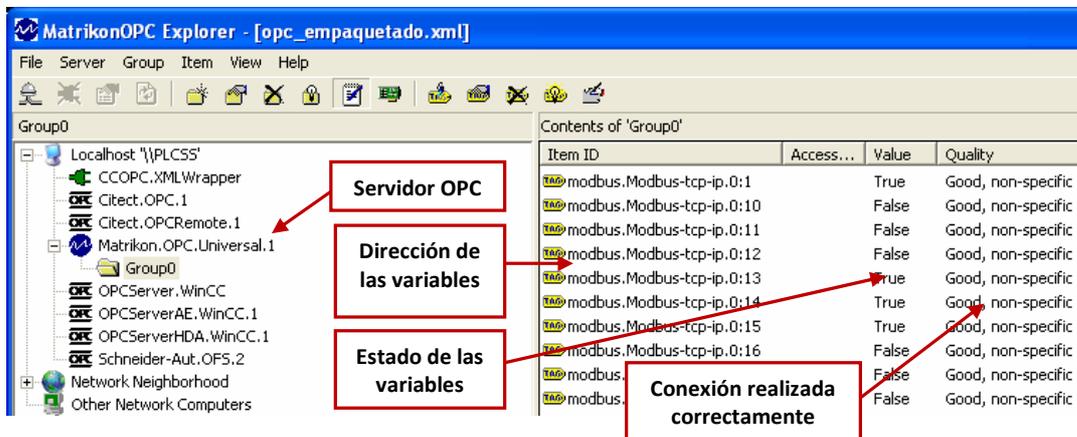


Fig.2.31 Explorador del Matrikon OPC Server

Después de haber configurado el servidor y comprobado la comunicación, es necesario declarar en el SCADA las variables que estarán conectadas al servidor OPC, como se muestra en la figura 2.32.

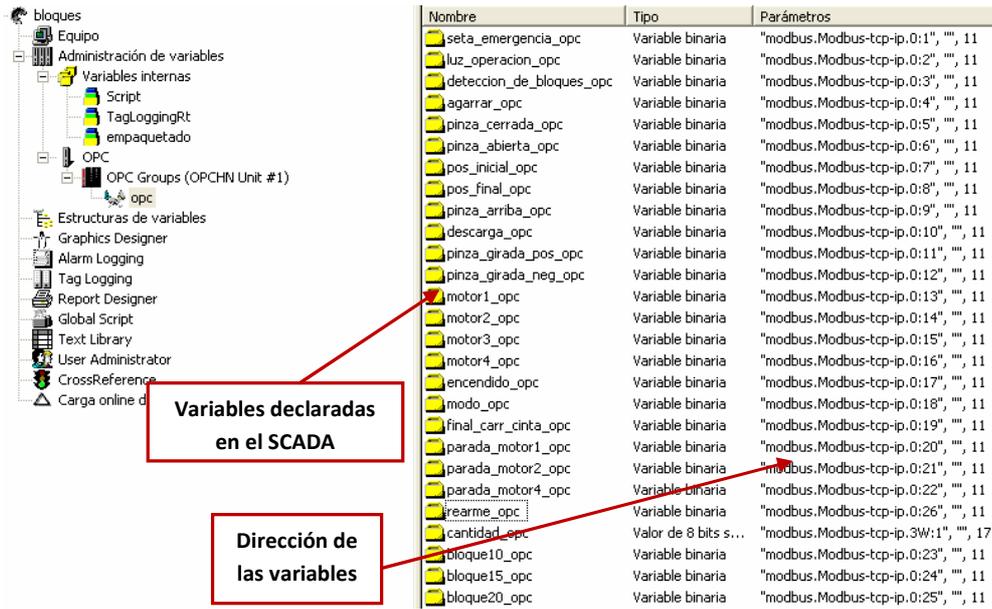


Fig.2.32 Variables declaradas en el SCADA

La configuración anterior para la comunicación del SCADA con el simulador de Modbus, funciona en tiempo real. Se comprobó la efectividad del uso de este sistema de comunicación por medio de la simulación, permitiendo al SCADA obtener los valores de las diferentes variables que intervienen en el control de proceso de empaquetado de bloques, pudiendo efectuarse la supervisión del mismo.

2.5 Validación del sistema automatizado

Se debe tener en cuenta que en la actualidad (proceso sin automatizar) las órdenes para el movimiento del manipulador las suministra un operario, lo cual conlleva a posibles errores humanos y posibles distracciones del mismo, lo que también afecta la continuidad del proceso. Además, existe una alta peligrosidad del ambiente debido a que hay operadores trabajando directamente en el proceso, expuestos sin ninguna barrera de protección tan cerca de la maquinaria. Por otro lado, la velocidad a la que se recorre cada uno de los segmentos en el proceso sin automatizar es menor a la prevista en el proceso automatizado. Hay que añadirle a este hecho que el tiempo de respuesta de los sensores en el proceso automatizado es mejor que el tiempo de los sensores existentes en la actualidad y que la acción sobre el proceso también es menor que si es manipulado por un operador.

En los resultados de la simulación en el caso automatizado, los elementos incluidos como los variadores y sensores permiten ir a mayor velocidad con más seguridad. Los variadores de velocidad permiten arrancar suavemente una determinada carga sin golpear la transmisión, variando el tiempo de aceleración, algo que no es posible con otros sistemas de variación mecánica, y como es lógico el aumento de la velocidad disminuye los tiempos de traslación, lo cual se traduce en una disminución de los tiempos de producción. Se debe notar que con la disminución de los tiempos aumenta considerablemente la producción de paquetes de bloques. En la tabla 2.11 se muestran los tiempos aproximados en cada uno de los segmentos existentes en el proceso de empaquetado. Como se mencionó en el epígrafe 2.2.4, para el movimiento del manipulador solo se configuraron dos velocidades, siendo una de ellas cero, debido a que las distancias recorridas por la pinza son cortas, teniendo en cuenta de manera general que la velocidad es constante en todo momento, se tomaron valores de velocidad por debajo del límite permisible, en este caso 0,8m/s en el caso del manipulador y 2m/s en el caso de las cintas transportadoras, sin embargo, en la cinta transportadora 2, encargada del traslado del paquete de bloques, la velocidad seleccionada está muy por debajo del valor permisible por un problema de seguridad, debido a que el volumen de la carga es grande.

	Sin automatizar		Automatizado		
Tiempo de movimiento de la Cinta transportadora 1	$t=3s$ ($t*5=15s$) ⁴	$s=3m$	$t=2s$ ($t*5=10s$)	$s=3m$	1.5m/s
Tiempo empleado en el Desplazamiento vertical	$t=5s$ ($t*4=20s$) ⁵	$s=1,5m$	$t=3s$ ($t*4=12s$)	$s=1,5m$	$v=0,5m/s$
Tiempo empleado en la apertura o cierre de la Pinza	$t=1s$ ($t*2=2s$) ⁶		$t=1s$ ($t*2=2s$)		
Tiempo empleado en el Desplazamiento horizontal	$t=6s$ ($t*2=12s$) ⁷	$s=2,5m$	$t=4s$ ($t*2=8s$)	$s=2,5m$	0,625m/s
Tiempo empleado en el realizar el Giro de la pinza	$t=0.5s$ ($t*2=1s$) ⁸		$t=0.5s$ ($t*2=1s$)		
Tiempo de movimiento de la Cinta transportadora 2	$t=10s$	$s=5m$	$t=7s$	$s=5m$	0,714m/s

Tabla 2.11: Tiempos de producción

Las siguientes expresiones matemáticas permiten estimar razonablemente el tiempo de cada sistema y confirman la eficiencia temporal del sistema automatizado. Se suma a esto, que

⁴ Se multiplica por 5 porque tiene que realizar cinco viajes de filas de bloques.

⁵ Se multiplica por 4 porque el brazo tiene que bajar dos veces y subir dos veces el segmento vertical.

⁶ Se multiplica por 2 porque la pinza realiza dos acciones abrir y cerrar.

⁷ Se multiplica por 2 porque el brazo tiene que recorrer dos veces el segmento horizontal.

⁸ Se multiplica por 2 porque el brazo en realiza el giro dos veces un sentido positivo otro en sentido negativo.

aún cuando no se tuvieran en cuenta los tiempos estimados, el sistema automatizado seguiría siendo superior al sistema actual, ya que las interrupciones que pudieran provocarse por causas directas o indirectas del operario en el sistema actual no ocurren con el sistema automatizado. En el cálculo no se ha tenido en cuenta ni el tiempo de respuesta de los sensores y actuadores, ni el tiempo de comunicación, ni el tiempo de ciclo del autómatas, por ser tan pequeños.

El tiempo del manipulador se puede estimar sumando cada uno de los tiempos que consumen estas etapas, ver tabla 2.11, desde que se detectan los bloques hasta que son depositados en el sitio correspondiente, en resumen. El tiempo total se estima sumando el tiempo que consume el manipulador en transportar una fila de bloques completa, desde que éstos son detectados hasta que se depositan en el sitio correspondiente y multiplicando este tiempo por 5, ya que son 5 las filas que debe trasladar el manipulador para terminar de completar un lote de bloques, más el tiempo que consume la primera cinta en realizar un recorrido completo multiplicado por 5, más el tiempo que consume la segunda cinta en realizar su recorrido. Las ecuaciones 6-8 muestran el cálculo de tiempo realizado y los resultados.

- Tiempo en el sistema actual, sistema sin automatizar:

$$t_m = 5 * (20s + 2s + 10s + 1s) \approx 165s \quad (5)$$

$$t_t = 15s + 165s + 10s \approx 190s \quad (6)$$

- Tiempo en el sistema automatizado:

$$t_m = 5 * (12s + 2s + 8s + 1s) = 5 * 21s \approx 115s \quad (7)$$

$$t_t = 10s + 115s + 7s \approx 132s \quad (8)$$

Donde t_m es el tiempo del manipulador y t_t el tiempo total del proceso, en ambos el peor de los casos.

En el caso del tiempo total de la secuencia del manipulador para completar un lote de 5 filas de bloques se tomó el peor de los casos en los dos sistemas, es decir, el caso en el que el brazo manipulador realiza el traslado de la primera fila, ya que es cuando se toma mayor

tiempo, debido a que realiza el mayor recorrido a la hora de depositar los bloques. Teniendo esto en cuenta no se afecta el resultado del análisis realizado. Se debe notar por las ecuaciones 6 y 8 que el tiempo total fue calculado con la suma del tiempo de la cinta transportadora 1, del manipulador y de la cinta transportadora 2, sin embargo, en realidad los tres subprocesos funcionan en paralelo, por lo que el tiempo que determina el empaquetado de bloques es básicamente el tiempo del manipulador.

Además, el uso de variadores y arrancadores en el sistema mejora el factor de potencia y el consumo de corriente de la instalación, disminuyendo los costos operativos. Es posible adecuar con toda precisión el par o torque del motor a la carga mecánica para obtener mínimo desgaste y esfuerzo del sistema, ahorrando energía eléctrica. Esta eficiencia energética se traduce en ahorro y ganancias para el país. Además, el uso de los variadores influye en la calidad del proceso debido a la transportación suave y segura de la mercancía, y por otro lado se disminuye la interacción hombre-máquina haciendo más segura la instalación en cuanto a recursos humanos.

2.6 Conclusiones

Se modeló el proceso de empaquetado de bloques de hormigón utilizando técnicas de control de eventos discretos a partir de Máquinas de Estados Finitos Deterministas, se diseñó su algoritmo utilizando diagramas GRAFCET a partir de las MEFD. Se desarrolló completamente la programación del PLC en base a los diagramas GRAFCET utilizando lenguaje estructurado sobre el software UnityPro, y se comprobó su correcto funcionamiento en modo de simulación. Se analizaron las principales características de los diferentes sensores y actuadores utilizados en el proyecto de automatización de la etapa de empaquetado de bloques de hormigón de la planta.

Se implementó un sistema SCADA para la supervisión y control remota del proceso con diferentes niveles de acceso y dos modos de funcionamiento, además de la gestión de alarmas, utilizando para ello, un servidor OPC capaz de establecer la comunicación entre el PLC, los dispositivos y el sistema de supervisión haciendo uso del bus de campo Modbus.

Se desarrolló el análisis de eficiencia operativa a modo de comparación entre el sistema actual, no automatizado, y el sistema de automatización desarrollado, teniendo en cuenta el

aspecto temporal, de seguridad y de consumo energético, y las ventajas que garantizan los dispositivos utilizados.

Conclusiones

Con la realización del presente trabajo de tesis se cumplieron los objetivos marcados en cada una de las etapas previstas. En concreto esto permitió arribar a las siguientes conclusiones generales:

- Se desarrolló un sistema de automatización integrado, para el control del proceso de empaquetado de bloques de hormigón en la planta de producción POYATOS de la Empresa de Materiales de la Construcción de Santiago de Cuba, que permitió:
 - Disminuir el tiempo de empaquetado significativamente.
 - Una mejora considerable en la eficiencia y calidad de la producción.
 - La disminución automáticamente de los niveles de peligrosidad y la generación de errores humanos, como consecuencia de la disminución en la interacción hombre-maquinaria.
- Se realizó un amplio análisis de la instrumentación utilizada (sensores, actuadores, CPU) para garantizar el control, teniendo en cuenta las particularidades de la planta, ajustándose principalmente a las necesidades económicas de la misma.
- Se analizó y aplicó un método apropiado para el diseño de la automatización de procesos industriales, basada en técnicas de control de eventos discretos y en el modelo matemático de Máquinas de Estados Finitos Deterministas, para describir formalmente el proceso, en conjunto con los diagramas GRAFCET para desarrollar el algoritmo del mismo, y que posteriormente se traducen a la programación concreta del PLC.
- Como parte del sistema de automatización integrada se desarrolló un sistema SCADA para la supervisión gestión y control remoto del proceso de empaquetado de bloques. Este sistema garantiza además, el almacenamiento de datos para posteriores análisis estadísticos.
- El análisis y la metodología aplicada constituyen la base para la automatización completa de cada uno de los procesos de producción de la planta, proyectando en ellos todos los beneficios alcanzados en este estudio, y dotando la totalidad de la misma de un modelo homogéneo y único de automatización.

Recomendaciones

- Como una consecuencia inmediata de los resultados obtenidos en este trabajo, se propone la implementación directa en planta del sistema de control automatizado, para verificar en un periodo prudencial de prueba el comportamiento del mismo y hacerlo extensible al resto de procesos.
- Se recomienda, desarrollar e implementar los sistemas homólogos en el resto de las plantas Poyatos existentes en el país, para de esta forma dar un impulso mayor a los programas de construcción de viviendas.
- De la misma manera se propone el desarrollo de un sistema SCADA centralizado, que sea capaz de intercomunicar a todos los subsistemas, y desde el cual sea posible supervisar y controlar a distancia al proceso global de la planta.
- Se recomienda, potenciar el estudio de la metodología usada en este trabajo en otras esferas de la automatización industrial.

Referencias

- [1] J. Castro; J. Padilla; E. Romero, *Metodología para realizar una automatización utilizando PLC*. Impulso, Revista de Electrónica, Eléctrica y Sistemas Computacionales, 2005.
- [2] L. Grigoro; K. Rudie, *Near-Optimal Online Control of Dynamic Discrete-Event Systems*. © Springer Science + Business Media, LLC 2006.
- [3] P. E. Miyagi, *Controle Programável Fundamentos do Controle de Sistemas a Eventos Discretos*, Editorial: EDGARD BLÜCHER LTDA, 1997.
- [4] B. Hruz; M.C. Zhou, *Modeling and Control of Discrete-event Dynamic Systems with Petri Nets and Other Tool*, Editorial: Springer, ISBN-13: 978-1-84628-872-2, 2007.
- [5] S. E. Bourdon; M. Lawford; W. M. Wonham, *Robust Nonblocking Supervisory Control of Discrete-Event Systems*. IEEE TRANSACTIONS ON AUTOMATIC CONTROL, Vol. 50, No. 12, ISSN 0018-9286, DECEMBER 2005.
- [6] D. Johnson, *Research on Programmable Logic Controllers*. Control Engineering Journal, 12/1/2007.
- [7] Potencia Electromecánica S.A. de C.V., *Variadores de Velocidad o Inversores de Frecuencia* [en línea]. 2009 [ref: Enero de 2012]. Disponible en World Wide Web:

<http://www.potenciaelectromecanica.com/variadores_frecuencia.php>
- [8] I. Bravo, *Introducción a los PLCs*. Departamento de Electrónica. Universidad de Alcalá, Escuela Politécnica Superior, Madrid, España, 2010.
- [9] J. R. Hackworth; F. D. Hackworth Jr., *Programmable Logic Controllers: Programming Methods and Applications*. Editorial: Prentice Hall, ISBN-13: 9780130607188, 2004.
- [10] W. Bolton, *Programmable Logic Controllers*, Editorial: Oxford Newnes (ELSEVIER), 4ta Ed., ISBN-13: 978-0750681124, 2006.
- [11] H. Jack, *Automating Manufacturing Systems with PLCs*. ISBN: 0750681329, 2005.
- [12] P. Leitão; A. W. Colombo, *Petri net based Methodology for the Development of Collaborative Production Systems*. IEEE TRANSACTIONS ON AUTOMATIC CONTROL, ISSN 1-4244-0681-1, 2006.

- [13] W. Han; M. A. Jafari, *Component and Agent-Based FMS Modeling and Controller Synthesis*. IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS—PART C, Vol. 33, No. 2, ISSN 1094-6977, Mayo 2003.
- [14] S. Peng; M. Zhou, *Conversion between Ladder Diagrams and PNs in Discrete-Event Control Design - A Survey*. IEEE TRANSACTIONS ON AUTOMATIC CONTROL, ISSN 0-7803-77, 2001.
- [15] Rokwell Automation Allen-Bradley, *Tendencias en Control de Motores*, Publicación 193-2.11 ES, Mayo 1998.
- [16] Schneider Electric, *Variadores de velocidad y Arrancadores electrónicos*. Manual, Capítulo 4, Francia, 2008.
- [17] N. Mohan, *Electric Drives. An Integrative Approach*. Editorial: MNPERE, ISBN-13: 978-0971529250, 2003.
- [18] F. Mazda, *Power Electronics Handbook*. Editorial: Oxford Newnes, 3ra Ed., ISBN: 0750629266, 1997.
- [19] Schneider Electric, *Arrancadores suaves para motores asíncronos*. Manual, Francia, 2009.
- [20] R.W. Brennan, *Toward Real-Time Distributed Intelligent Control: A Survey of Research Themes and Applications*. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, Vol. 37, 2007.
- [21] I. Benitez, *Integración internacional de plataformas de enseñanza a distancia de automatización con PLCs*, IEEE-RITA. Revista Iberoamericana de Tecnologías de Aprendizaje. Vol 1. No 1. Nov. ISSN 1932-8540, 2006.
- [22] M. H. De Queiroz; J. E. R. Cury; W. M. Wonham, *Multitasking Supervisory Control of Discrete-Event Systems*. © Springer Science + Business Media, 2005.
- [23] C.G. Cassandras, *FROM PROGRAMMABLE LOGIC CONTROL (PLC) TO DISCRETE EVENT SYSTEMS (DES)*. Dept. of Manufacturing Engineering, Universidad de Boston, Boston, EUA, 2000.
- [24] C. R. C. Torrico; J. E. R. Cury, *CONTROLE SUPERVISÓRIO HIERÁRQUICO MODULAR POR AGREGAÇÃO DE ESTADOS*. Revista Controle & Automação, Vol.15 No.3, Julio, Agosto y Septiembre, 2004.
- [25] M. R. Da Silveira; M. Combacau, *SUPERVISION AND CONTROL OF HETERARCHICAL DISCRETE EVENT SYSTEMS*. Revista Controle & Automação, Vol.17 no.1, Janeiro, Febrero y Marzo, 2006.

- [26] J. Hopcroft; R. Motwani; J. Ullman, *Introduction to Automata Theory, Languages and Computation*. Editorial: Addison-Wesley, 3ra Ed., ISBN-13: 9780321455369, 2007.
- [27] A. Aho; R. Sethi; J. Ullman, *Compilers: Principles, Techniques and Tools*. Editorial: Addison-Wesley, 2da: Ed., ISBN-13: 978-0321547989, 2007.
- [28] S. Moral, *Teoría de Autómatas y Lenguajes Formales*. Departamento de Ciencias de la Computación e I.A., ETSI Informática, Universidad de Granada, 2011.
- [29] J. Aguilar, *Análisis de Máquinas de Estados Finitos Usando la Programación Evolutiva*. IV CIBELEC, V JIELECTRIC, ISBN: 978-980-7185-1, 2010.
- [30] M. A. Pérez; M. Readman, *Sistemas digitales y el entrenador Lógico CE300*. División de electrónica y computación, CUCEI, Universidad de Guadalajara, México, 2005.
- [31] E. J. Gómez; D. Vargas, *Diseño de Máquinas de estados*. Universidad Distrital Francisco José de Caldas, Facultad Tecnológica. Ingeniería en Control Electrónico e Instrumentación, Bogotá, Colombia 2003.
- [32] R. Álvarez, *Introducción a las máquinas de estado finito* [en línea]. 2005 [ref: Enero de 2012]. Disponible en World Wide Web:

< <http://www.tecbolivia.com/index.php?view=article> >
- [33] J. Rodríguez, *Metodología de diseño de automatismos secuenciales: GRAFCET*. UPCO ICAI, Departamento de Electrónica y Automática, Madrid España, 2001.
- [34] J. M. Gea, *Introducción al GRAFCET* [en línea]. Marzo de 2006 [ref: Enero de 2012]. Disponible en World Wide Web:

<<http://www.automatas.org/redes/grafcet.htm>>
- [35] F. Ferreira, *Grafcet gráfico de comando etapa de transición*. Departamento de electrónica, Universidad de Buenos Aires, 2004.
- [36] K. John; M. Tiegelkamp, *IEC 61131-3: Programming Industrial Automation Systems*. Editorial: Springer, ISBN3-540-67752-6, 2001.
- [37] S. Haddad; B. Haverkort; h. Hermanns, otros, *Performance Models for Discrete Event Systems with Synchronisations: Formalisms an Analysis Techniques*, Folleto para escuela de verano Jaca, España, 1998.

- [38] J.C. Sánchez, *Diseño de un Sistema de Control Automatizado Eficiente para el Transporte de Bloques Hormigón en Plantas Productivas tipo "POYATOS" usando Redes Petri*. Departamento de Control Automático, Facultad de Ingeniería Eléctrica, Universidad de Oriente, Santiago de Cuba, Cuba, 2011.
- [39] Autómatas Industriales, *Sistemas SCADA* [en línea]. Marzo de 2006 [ref: Enero de 2012]. Disponible en World Wide Web:

<<http://www.automatas.org/redes/scadas.htm>>
- [40] Carmen D'Sousa, *SISTEMAS SCADA* [en línea]. 2010 [ref: Enero de 2012]. Disponible en World Wide Web:

<<http://www.monografias.com/trabajos11/sisco/sisco.shtml>>
- [41] I. Benitez; O. Gomis; A. Sudriá; otros, *Flexible manufacturing cell SCADA system for educational purposes*. Journal Computer Applications in Engineering Education, Vol. 16, No 1, ISSN 1061-3773, Mayo 2008.
- [42] D. Bailey; E. Wright, *Practical SCADA for Industry*. Editorial: Newnes (ELSEVIER), ISBN 07506 58053, 2003.
- [43] J. Villajulca, *¿Qué es SCADA?* [en línea]. 2011 [ref: Enero de 2012]. Disponible en World Wide Web:

<<http://www.instrumentacionycontrol.net/es/cursos-sistemas-scada/146-que-es-scada-.html>>
- [44] Compañía Matrikon OPC, *¿Qué es un servidor OPC?* [en línea]. Barcelona, España 2011 [ref: Enero de 2012]. Disponible en World Wide Web:

<<http://www.matrikonopc.es/opc-servidor/index.aspx>>
- [45] M. A. Pérez; J. Álvarez; J. Campo; otros, *Instrumentación electrónica*. Editorial THOMSON, ISBN: 8497321669, 2004.
- [46] Schneider Electric, *Interruptores de posición*. Manual, Francia, 2010.
- [47] Schneider Electric, *Codificadores giratorios optoelectrónicos Osicoder*. Catálogo, Francia, 2006.
- [48] Schneider Electric, *Sensores de Proximidad Inductivos Oxiprox*. Manual, Francia, 2009.

- [49] Schneider Electric, *Hardware MODICON M340*. Instituto de Formación, Francia, 2007.
- [50] Schneider Electric, *Características ATV 28* [en línea]. [Francia] 2011 [ref: Enero de 2012]. Disponible en World Wide Web:

<<http://www.schneider-electric.com/site/home/index.cfm/es/>>
- [51] Schneider Electric, *Variadores de velocidad ajustable ALTIVAR 28. Guía del usuario*. Manual de Usuario, Francia, 2003.
- [52] Schneider Electric, *Arranadores suaves para motores asíncronos*. Manual, Francia, 2009.
- [53] ECURED, *Protocolo de comunicación Modbus* [en línea]. 2011 [ref: Enero de 2012]. Disponible en World Wide Web:

<http://www.ecured.cu/index.php/Protocolo_de_Comunicaci%C3%B3n_Modbus>
- [54] Automatas Industriales, *Modbus Protocol* [en línea]. Marzo de 2006 [ref: Enero de 2012]. Disponible en World Wide Web:

<<http://www.automatas.org/modbus/intr7.html>>

Anexos

Anexo 1: Ventajas de los Variadores de Velocidad

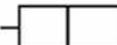
Entre las diversas ventajas en el control del proceso proporcionadas por el empleo de variadores de velocidad destacan:

- Es posible mover la velocidad del motor según las necesidades de la producción o el proceso, esto puede ser hecho de forma automática o manual sin la necesidad de costosos motores especiales de corriente directa y controladores de corriente directa que en conjunto resultan más costosos sin ninguna ventaja sino por el contrario, conllevan mayores costos de mantenimiento porque el motor tiene múltiples puntos de falla.
- El variador cuenta con una carátula en donde se indican corriente circulante, voltaje, velocidad del motor, fallas, etc. no disponibles en otro tipo de controles de velocidad como son los mecánicos o los de corriente directa.
- No se requieren protecciones adicionales de sobre corriente para la protección del motor, esto lo hace el propio variador de forma fina y ajustable sin ningún costo adicional.
- Es posible automatizar el sistema utilizando otras características propias del inversor de frecuencia utilizando para ellos circuitos de lazo cerrado con encoder o similar. También es posible un excelente control de velocidad o par usando lazo abierto mediante la característica de “control vectorial” de los variadores.
- Se puede arrancar suavemente una determinada carga sin golpear la transmisión variando el tiempo de aceleración lo que no es posible con otros sistemas de variación mecánica.
- Se puede regular la corriente de arranque del motor aumentando la capacidad de arranques y paros frecuentes (hasta 20-25 por minuto) sin sobrecalentar el motor y disminuyendo el consumo de energía.
- Mejora el factor de potencia y el consumo de corriente de la instalación disminuyendo los costos operativos.
- Es posible manejar automáticamente varios variadores de frecuencia funcionando en cascada en donde la velocidad de unos dependerán de la indicación de otro llamado

variador maestro. Cualquiera de los variadores puede ser el variador maestro que mandará a los demás. Si se mueve la velocidad de un tramo de banda transportadora, automáticamente variará la velocidad de otros tramos de banda interconectados en el mismo o diferente proceso que tengan motores diferentes.

- Es posible adecuar con toda precisión el par o torque del motor a la carga mecánica para obtener mínimo desgaste y esfuerzo del sistema ahorrando energía eléctrica y en algunos casos será posible mover una misma carga con menos potencia que cuando se utiliza un arrancador convencional (alta inercia).
- Es factible usar el variador de frecuencia para controlar el tiempo de desaceleración del sistema mecánico.
- El costo inicial de adquisición es menor que un variador mecánico o un conjunto de corriente directa.

Anexo 2: Simbología y reglas del GRAFCET

Simbolo	Nombre	Descripción
	Etapa inicial	Indica el comienzo del esquema GRAFCET y se activa al poner en RUN el autómata. Por lo general suele haber una sola etapa de este tipo.
	Etapa	Su activación lleva consigo una acción o una espera.
	Unión	Las uniones se utilizan para unir entre si varias etapas.
	Transición	Condición para desactivarse la etapa en curso y activarse la siguiente etapa. Se indica con un trazo perpendicular a una unión.
	Direccionamiento	Indica la activación de una y/u otra etapa en función de la condición que se cumpla/n. Es importante ver que la diferencia entre la "o" y la "y" en el grafcet es lo que pasa cuando se cierran
	Proceso simultáneo	Muestra la activación o desactivación de varias etapas a la vez.
	Acciones asociadas	Acciones que se realizan al activarse la etapa a la que pertenecen.

Reglas del GRAFCET

En el GRAFCET se cumplen una serie de reglas que se mencionan a continuación:

- La etapa inicial de un gráfico GRAFCET se activa de forma incondicional. Esta situación inicial se corresponde en general con una situación de reposo.
- Una transición está en disposición de ser validada cuando todas las etapas inmediatamente precedentes, unidas a dicha transición, están activadas. La activación de una transición se produce cuando está validada y la condición de transición o receptividad es verdadera. Se podría definir una etapa como activable cuando la transición precedente está validada.

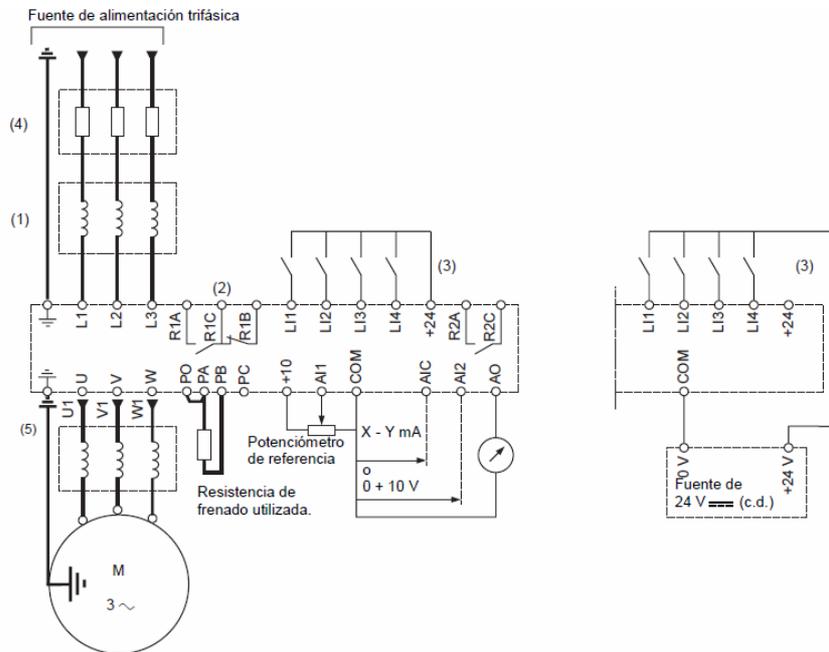
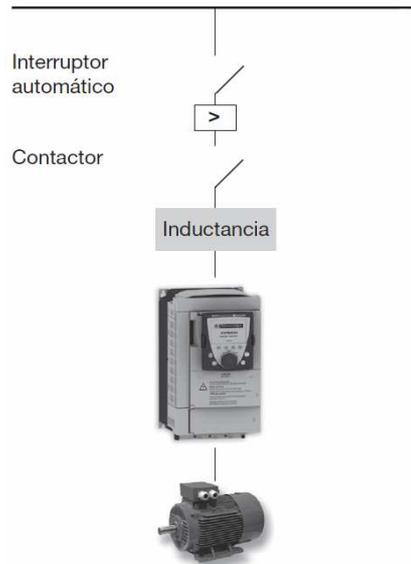
- Franquear una transición implica la activación de todas las etapas siguientes inmediatas, y la desactivación de las inmediatas precedentes.
- Transiciones conectadas en paralelo, se activan de forma simultánea si se cumplen las condiciones para ello.
- Una o varias acciones se asocian a cada etapa. Estas acciones sólo están activas cuando la etapa esta activa.

Anexo 3: Funciones básicas de un sistema de supervisión y adquisición de datos

- *Supervisión Remota de Instalaciones:* Mediante esta función, una de las principales del sistema SCADA, el usuario es capaz de conocer el estado de las instalaciones bajo su responsabilidad y coordinar eficientemente las labores de producción y mantenimiento en el campo. El intervalo de recolección periódica de la información del campo depende de las dimensiones, pero generalmente está en el orden de unos cuantos milisegundos.
- *Control Remoto de Instalaciones:* Los sistemas SCADA permiten activar o desactivar equipos remotos (Por ejemplo: interruptores, transformadores, bombas, válvulas, compresores, etc.) de manera automática o a solicitud del operador. Igualmente es posible realizar ajustes en parámetros en lazos de control analógicos (punto de consigna).
- *Procesamiento de Información:* En algunos casos, los datos capturados requieren procesamiento adicional, a efectos de consolidar información proveniente de diferentes lugares remotos.
- *Presentación de Gráficos Dinámicos:* Esto se refiere al despliegue de pantallas con el diagrama del proceso conteniendo información instantánea del comportamiento del mismo.
- *Generación de Reportes:* Los sistemas SCADA permiten la generación automática, o a petición, de reportes impresos de producción y balances.
- *Presentación de Alarmas:* Mediante esta función se alerta al operador sobre la ocurrencia de condiciones anormales o eventos que pudieran requerir su intervención. Normalmente, la criticidad del evento o alarma se indica mediante el uso de colores y/o señales auditivas. Las alarmas se registran para análisis posteriores.
- *Almacenamiento de información histórica:* Los sistemas SCADA permiten registrar y almacenar información operacional y alarmas. Por ejemplo, se pueden llevar datos de los últimos 5 minutos, 1 hora, 1 día, 1 mes, hasta un año y más.

- *Presentación de Gráficos de Tendencias:* Con información en tiempo real o histórico, se pueden construir gráficos e inferir el comportamiento de variables operacionales en el tiempo.
- *Programación de Eventos:* Se refiere a la posibilidad de programar en el tiempo la generación de reportes, despliegue de diagramas del proceso o activación de tareas o comandos del sistema.

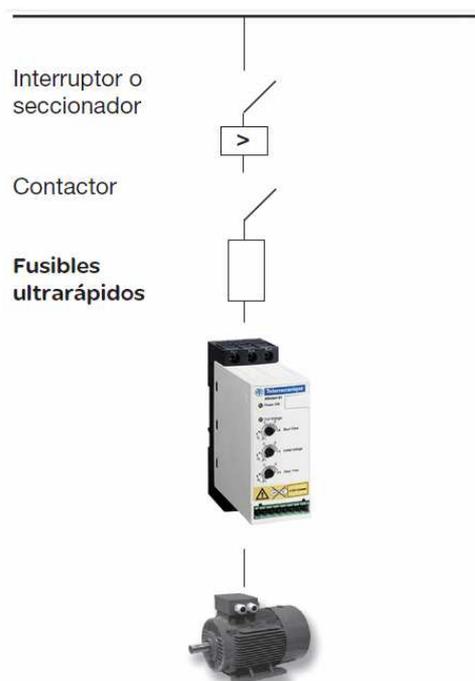
Anexo 4: Esquema de conexión del variador de frecuencia Altivar 28

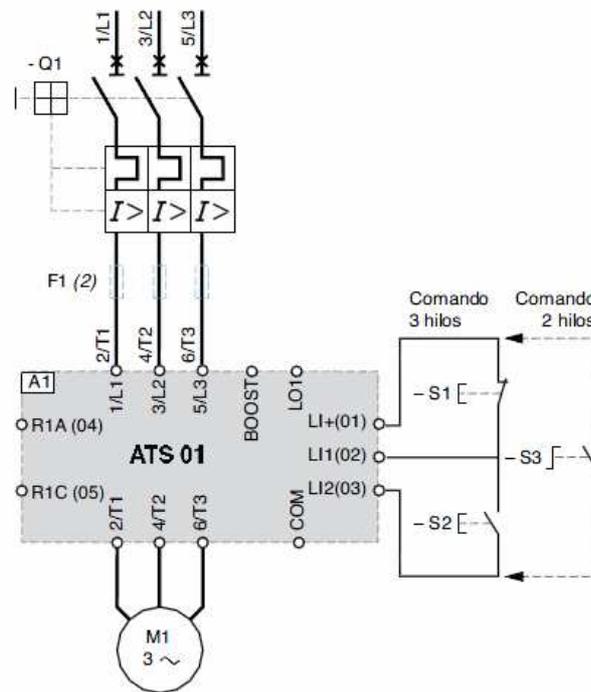


- (1) Reactancia de línea utilizada. Todas las instalaciones de 575 V deberán incluir una reactancia de línea.
 (2) Contactos del relé de falla para indicar a distancia el estado del variador.

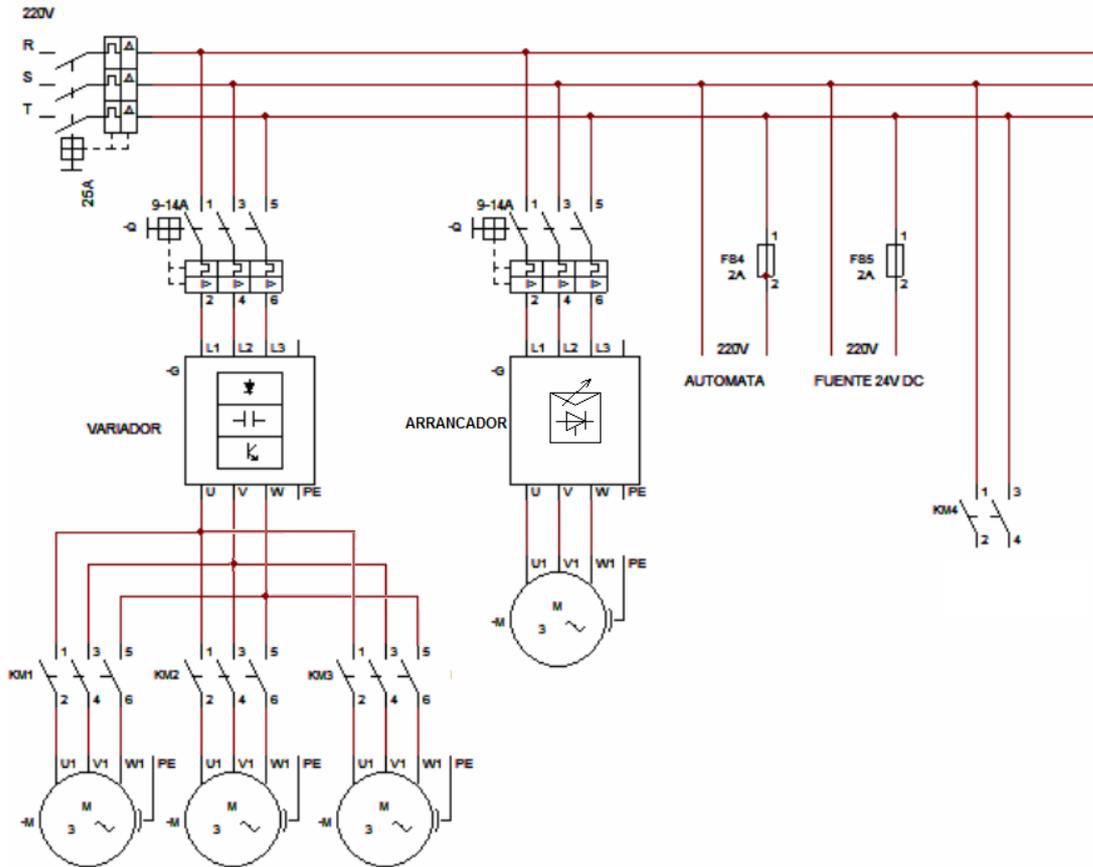
- (3) Interno de +24 V (DC). Si se utiliza una fuente externa de +24 V, conecte 0 V/terminal común de la fuente a la terminal COM. No utilice la terminal de +24 V del variador.
- (4) Coloque los fusibles aquí.

Anexo 5: Esquema de conexión del arrancador Altistart 01





Anexo 6: Esquema eléctrico para el control de tres motores por un variador



Anexo 7: Programa en ST para el control del proceso de empaquetado de bloques en plantas POYATOS

```

(* encendido del sistema (todo tiene alimentacion) *)
if encendido then (* entrada del plc activa si hay alimentacion *)
  luz_encendido := true; (* luz blanca *)
else
  luz_encendido := false; (* luz blanca *)
end_if;

(* seta de emergencia siempre se verifica tanto en modo automatico como manual *)
if seta_emergencia and termico0 and termico1 and termico2 and termico3 and termico4
and Alarma then (* pulsador de enclavamiento *)
  luz_operacion := false; (* luz verde *)
  luz_emergencia := true; (* luz roja *)
  motor1_LI1:=false; (* parada motor vertical *)
  motor2_LI1:=false; (* parada motor horizontal *)
  motor3_LI1:=false; (* parada motor giro *)
  motor4_LI1:=false; (* parada motor cinta *)
end_if;

(* reset (o reinicio) siempre se verifica tanto en modo automatico como manual *)
if primer_ciclo or reset then (* pulsador sin enclavamiento *)
  luz_emergencia := false; (* luz roja *)
  c := 0;
  f := 0;
  estado_brazo := 0;
  cambio_estado_brazo := 0;
  estado_cinta := 0;
  cambio_estado_cinta := 0;
  (* resto de las inicializaciones o acciones correspondientes *)
end_if;

FBI_0 (IN := Enable, PT := T#7s);

if modo then (* modo = true significa automatico *)

  if not seta_emergencia and not reset then
    luz_operacion := true; (* luz verde *)

    (-----*)
    (* PROCESO CORRESPONDIENTE A LA CINTA 1 *)
    (-----*)

    (* estado de reposo de la cinta *)
    if estado_cintal = 0 then
      motor0_LI1:=false;
      if final_carrera_deteccion_bloques then
        estado_cintal := 1;
      end_if;
    end_if;

    (* mover la cinta *)
    if estado_cintal = 1 then
      motor0_LI1:=true;
      if not final_carrera_deteccion_bloques then
        estado_cintal := 0;
      end_if;
    end_if;
  end_if;
end_if;

```

```

end_if;

(*-----*)
(* PROCESO CORRESPONDIENTE AL BRAZO *)
(*-----*)

(* deteccion de bloques, estado inicial *)
if estado_brazo = 0 then
  motor2_LI1:=true;
  motor2_LI2:=false;
  motor2_LI3:=true;
  motor2_LI4:=false; (* parada normal *)
  cambio_estado_brazo := estado_brazo;
  if final_carrera_deteccion_bloques and f <> CANTIDAD_DE_FILAS and
inductivo_posición_inicial and inductivo_pinza_abierta then
    estado_brazo := 1;
  end_if;
end_if;

(* bajar a cargar *)
if estado_brazo = 1 then
  if cambio_estado_brazo <> estado_brazo then
    if f = 0 then
      h := ALTURA_DE_DESCARGA;
    else
      h := h - ALTURA_DEL_BLOQUE;
    end_if;
  end_if;
  Enable:=true; (* activar temporizador *)
  set_point_velocidad:=VELOCIDAD_VERTICAL;(* en caso de usar un solo variador
*)

  motor1_LI1:=true;
  motor1_LI2:=true; (* bajar *)
  motor1_LI3:=false;
  motor1_LI4:=true; (* velocidad máxima *)
  cambio_estado_brazo := estado_brazo;
  if FBI_0.Q then (* paso el tiempo establecido *)
    estado_brazo := 11; (* alarma *)
  end_if;
  if (lectura_encoder >= ALTURA_DE_CARGA) and not FBI_0.Q then (* agarrar *)
    agarrar:=true; (* variable para el SCADA *)
    estado_brazo := 2; (* incluir una variable agarrar=1 *)
    Enable:=false; (* desactivar temporizador *)
  end_if;
end_if;

(* coger el bloques *)
if estado_brazo = 2 then
  motor1_LI1:=true;
  motor1_LI2:=true; (* bajar *)
  motor1_LI3:=false;
  motor1_LI4:=false; (* parada normal *)
  cambio_estado_brazo := estado_brazo;
  if inductivo_pinza_cerrada then
    estado_brazo := 3;
  end_if;
end_if;

(* subir la carga *)
if estado_brazo = 3 then
  Enable:=true; (* activar temporizador *)
  set_point_velocidad:=VELOCIDAD_VERTICAL;(* en caso de usar un solo variador
*)

  motor1_LI1:=true;
  motor1_LI2:=false;
  motor1_LI3:=true; (* subir *)
  motor1_LI4:=true; (* velocidad máxima *)

```

```

agarrar:=false; (* variable para el SCADA *)
cambio_estado_brazo := estado_brazo;
if FBI_0.Q then (* paso el tiempo establecido *)
    estado_brazo := 11; (* alarma *)
end_if;
if inductivo_posicion_inicial and inductivo_pinza_cerrada then
    estado_brazo := 4;
    Enable:=false; (* desactivar temporizador *)
end_if;
end_if;

(* mover a la derecha con carga *)
if estado_brazo = 4 then
    Enable:=true; (* activar temporizador *)
    motor1_LI1:=true;
    motor1_LI2:=false;
    motor1_LI3:=true; (* subir *)
    motor1_LI4:=false; (* parada normal *)
    set_point_velocidad:=VELOCIDAD_HORIZONTAL;(* en caso de usar un solo
variador *)
    motor2_LI1:=true;
    motor2_LI2:=true; (* derecha *)
    motor2_LI3:=false;
    motor2_LI4:=true; (* velocidad máxima *)
    cambio_estado_brazo := estado_brazo;
    if FBI_0.Q then (* paso el tiempo establecido *)
        estado_brazo := 11; (* alarma *)
    end_if;
    if inductivo_posicion_final and inductivo_pinza_cerrada then
        estado_brazo := 5;
        Enable:=false; (* desactivar temporizador *)
    end_if;
end_if;

(* girar pinza *)
if estado_brazo = 5 then
    Enable:=true; (* activar temporizador *)
    motor2_LI1:=true;
    motor2_LI2:=true;
    motor2_LI3:=false; (* derecha *)
    motor2_LI4:=false; (* parada normal *)
    if f mod 2 = 1 then (* si f impar, porque f empieza en 0 *)
        set_point_velocidad:=VELOCIDAD_GIRO;(* en caso de usar un solo variador
*)
        motor3_LI1:=true;
        motor3_LI2:=true; (* giro sentido + *)
        motor3_LI3:=false;
        motor3_LI4:=true; (* velocidad máxima *)
    end_if;
    cambio_estado_brazo := estado_brazo;
    if FBI_0.Q then (*paso el tiempo establecido*)
        estado_brazo := 11; (* Alarma *)
    end_if;
    if (f mod 2 = 0 or inductivo_giro_positivo) and estado_cinta <> 1 then (*
estado_cinta <> 1 indica que la cinta esta parada. aqui pinza_girada es lo mismo que f
mod 2 = 0 or inductivo_giro_positivo *)
        estado_brazo := 6;
        Enable:=false; (* desactivar temporizador *)
    end_if;
end_if;

(* bajar a descargar *)
if estado_brazo = 6 then
    Enable:=true; (* activar temporizador *)
    motor3_LI1:=true;
    motor3_LI2:=true; (* giro sentido + *)
    motor3_LI3:=false;
    motor3_LI4:=false; (* parada normal *)
    set_point_velocidad:=VELOCIDAD_VERTICAL;(* en caso de usar un solo variador

```

```

*)
    motor1_LI1:=true;
    motor1_LI2:=true; (* bajar *)
    motor1_LI3:=false;
    motor1_LI4:=true; (* velocidad máxima *)
    cambio_estado_brazo := estado_brazo;
    if FBI_0.Q then (* paso el tiempo establecido *)
        estado_brazo := 11; (* alarma *)
    end_if;
    if lectura_encoder >= h then
        descarga:=true; (* variable para el scada *)
        estado_brazo := 7;
        Enable:=false; (* desactivar temporizador *)
    end_if;
end_if;

(* soltar el bloques *)
if estado_brazo = 7 then
    motor1_LI1:=true;
    motor1_LI2:=true; (* bajar *)
    motor1_LI3:=false;
    motor1_LI4:=false; (* parada normal *)
    cambio_estado_brazo := estado_brazo;
    if inductivo_pinza_abierta then
        estado_brazo := 8;
    end_if;
end_if;

(* subir despues de descarga *)
if estado_brazo = 8 then
    Enable:=true; (* activar temporizador *)
    if cambio_estado_brazo <> estado_brazo then
        f := f + 1; (* se cuenta una fila depositada *)
        c := c + CANTIDAD_DE_BLOQUES_EN_UNA_FILA; (* se cuentan los bloques *)
    end_if;
    set_point_velocidad:=VELOCIDAD_VERTICAL;(* en caso de usar un solo variador
*)
    motor1_LI1:=true;
    motor1_LI2:=false;
    motor1_LI3:=true; (* subir*)
    motor1_LI4:=true; (* velocidad máxima *)
    descarga:=false; (* variable para el scada *)
    cambio_estado_brazo := estado_brazo;
    if FBI_0.Q then (* paso el tiempo establecido *)
        estado_brazo := 11; (* alarma *)
    end_if;
    if inductivo_posicion_final and inductivo_pinza_abierta then
        estado_brazo := 9;
        Enable:=false; (* desactivar temporizador *)
    end_if;
end_if;

(* girar pinza *)
if estado_brazo = 9 then
    Enable:=true; (* activar temporizador *)
    motor1_LI1:=true;
    motor1_LI2:=false;
    motor1_LI3:=true; (* subir *)
    motor1_LI4:=false; (* parada normal *)
    if f mod 2 = 0 then (* si f par, porque f empieza en 0 y ya se incremento
*)
        set_point_velocidad:=VELOCIDAD_GIRO;(* en caso de usar un solo variador *)
        motor3_LI1:=true;
        motor3_LI2:=false;
        motor3_LI3:=true; (* giro sentido - *)
        motor3_LI4:=true; (* velocidad máxima *)
    end_if;
    cambio_estado_brazo := estado_brazo;
    if FBI_0.Q then (* paso el tiempo establecido *)

```

```

        estado_brazo := 11; (* alarma *)
    end_if;
    if f mod 2 = 1 or inductivo_giro_negativo then (* estado aqui pinza_girada
es lo mismo que f mod 2 = 1 or inductivo_giro_negativo *)
        estado_brazo := 10;
        Enable:=false; (* desactivar temporizador *)
    end_if;
end_if;

(* mover a la izquierda *)
if estado_brazo = 10 then
    Enable:=true; (* activar temporizador *)
    motor3_LI1:=true;
    motor3_LI2:=false;
    motor3_LI3:=true; (* giro sentido - *)
    motor3_LI4:=false; (* parada normal *)
    set_point_velocidad:=VELOCIDAD_HORIZONTAL;(* en caso de usar un solo
variador *)
    motor2_LI1:=true;
    motor2_LI2:=false;
    motor2_LI3:=true; (* izquierda *)
    motor2_LI4:=true; (* velocidad máxima *)
    cambio_estado_brazo := estado_brazo;
    if FBI_0.Q then (* paso el tiempo establecido *)
        estado_brazo := 11; (* alarma *)
    end_if;
    if inductivo_posicion_inicial and inductivo_pinza_abierta then
        estado_brazo := 0;
        Enable:=false; (* desactivar temporizador *)
    end_if;
end_if;
(* alarma *)
if estado_brazo = 11 then
    Alarma:=true;
    Enable:=false;
    if Pulsador then
        Alarma:=false;
        estado_brazo:= cambio_estado_brazo;
    end_if;
end_if;

(*-----*)
(* PROCESO CORRESPONDIENTE A LA CINTA 2 *)
(*-----*)

(* estado de reposo de la cinta *)
if estado_cinta = 0 then
    motor4_LI1:=false;
    cambio_estado_cinta := estado_cinta;
    if inductivo_pinza_abierta and f = CANTIDAD_DE_FILAS then
        estado_cinta := 1;
    end_if;
end_if;

(* mover la cinta *)
if estado_cinta = 1 then
    if cambio_estado_cinta <> estado_cinta then
        f := 0;
    end_if;
    motor4_LI1:=true;
    cambio_estado_cinta := estado_cinta;
    if final_carrera_cinta then
        estado_cinta := 0;
    end_if;
end_if;

end_if;

```

```
else (* modo = false significa modo manual *)  
    (* resto de operaciones que se deben realizar en modo manual *)  
    (* aqui por ejemplo se pueden realizar las operaciones necesarias si el brazo se  
    quedara en un estado en el que no es posible continuar desde el modo automatico,  
    realizando cuidadosamente las operaciones correspondientes *)  
end_if;
```