



Universidad de Oriente
Facultad de Ingeniería Eléctrica
Departamento de Control Automático

TRABAJO DIPLOMA

Tema: Sistema de Automatización Integrada para la etapa de Recepción en la fábrica de piensos de Baire.

Autor: Suzel Lora Alonso
Tutores: Dr. Israel Francisco Benítez Pina
MSc. Guillermo J. Acosta Montoya

Santiago de Cuba
2016-2017

Dedicatoria:

Dedico este trabajo a toda mi familia, en especial a mis padres, a mis hermanos, a mi tutor y compañero, y por sobre todas las cosas a mi querido hijo.

A todos, gracias por formar parte de mi vida y hacer que me sienta cada día más orgullosa de formar parte de la suya.

Agradecimientos:

A mis tutores MSc Guillermo J. Acosta Montoya y Dr.C. Israel Benítez Pina por su comprensión ayuda y dedicación durante todos estos meses de trabajo.

A todos los profesores de la facultad que durante estos 5 años han sido mi guía y ejemplo como futura ingeniera.

A mis amigos que siempre me apoyaron en especial a Roxana.

A mis suegros por apoyarme en la difícil tarea de ser madre e ingeniera a la vez.

A las personas que de alguna forma colaboraron con esta realización.

A todos, Muchas gracias.

Resumen

Debido a la situación del poco equipamiento de instrumentación y control en la Fábrica de Piensos de Baire se propone un Sistema de Automatización Integrada para la etapa de Recepción empleando las Redes de Petri como herramienta de modelado y simulación. Primeramente, se caracteriza de forma teórica los niveles de la arquitectura del sistema para seleccionar los elementos de medición y control más apropiados. Antes de definir la solución se decide modelar en Redes de Petri el flujo de datos del sistema, el control de la automática local y la dinámica de funcionamiento del Supervisorio con el objetivo de proponer un algoritmo de programación que incorpore los conceptos del mantenimiento preventivo, la detección de errores de comunicación y la atención de alarmas y tratamiento de fallos.

La correcta interpretación de cada modelo permite definir una solución comparable con los estándares mundiales de automatización, pues se emplea el bus AS-i para el control de sensores y actuadores, que a través de la comunicación Profibus garantizan almacenar la información del proceso en el PLC (S7-400) con un nombre y dirección específica, con el objetivo de calcular las condiciones de control para cumplir la secuencia de operación del proceso en el Supervisorio (WinCC). Como aporte de la investigación se lleva a cabo el modelado del funcionamiento de la comunicación AS-i y del principio de programación con reutilización de bloques y encapsulamiento de datos empleados en la función de control de los actuadores de la planta, permitiendo identificar las condiciones para detectar posibles alarmas y sugerir una solución.

Palabras Claves: *Redes de Petri (RdP); Sistema de Automatización Integrada (SAI).*

Abstract

Due to the situation of the poor instrumentation and control equipment in the Baire Feeds Factory, an Integrated Automation System is proposed for the Reception stage, using Petri Nets techniques as a modeling and simulation tool. First, the levels of the system architecture are theoretically characterized to select the most appropriate measurement and control elements. Before defining the solution, it is decided to model in Petri Nets the data flow of the system, control of the local automatic and the dynamics of operation of the supervisor with the objective of proposing a programming algorithm that incorporates the concepts of preventive maintenance, detection of communication errors and alarms attention and failures treatment.

The correct interpretation of each model allows to define a solution comparable with the world-wide standards of automation, because the AS-i bus is used for the control of sensors and actuators, that through the communication Profibus guarantee to store the process information in the PLC (S7-400) with a specific name and address, in order to calculate the control conditions to comply with the sequence of operation of the process in the Supervisor (WinCC). As a contribution of the research, the AS-i communication modeling and the programming principle reutilizing blocks with capsulizing of data are used to control the actuators of the plant, allowing the identification of conditions for detecting possible alarms and suggesting a solution.

Keyword: *Petri Nets (PN); Integrated Automation System (IAS).*

Índice

Resumen	ii
Abstract	iii
Introducción.....	1
Capítulo 1: Caracterización de los SAI y definición teórica de las Redes de Petri como herramienta de modelado y simulación.....	4
1.1 Sistemas de Automatización Integrada (SAI).	4
1.1.1 Antecedentes.....	4
1.1.2 Sistemas de Automatización en la Actualidad.....	4
1.1.3 Características y Generalidades de SAI.....	5
1.1.4 Diseño de un Sistema de Automatización Integrada (SAI).....	6
1.2 El empleo de las Redes de Petri para motivar el modelado formal de SAI. 8	
1.2.1 Redes de Petri.	8
1.2.2 Propiedades de las redes de Petri.....	9
1.2.3 Modelado de SAI a través de las redes GHENeSys.	10
1.3 Descripción del Flujo Tecnológico y caracterización de los elementos que conforman el proceso.....	13
1.3.1 Medición de Nivel máximo y mínimo en los silos	14
1.3.2 Sensor Inductivo para detectar fallo de la cadena del elevador.....	14
1.3.3 Actuadores de la Planta.	14
1.3.4 Bus de campo para sensores y actuadores.	15
1.4 Bus ASI.	15
1.4.1 Beneficios de una red AS-i.....	16
1.4.2 Características principales de una red AS-i.	16
1.4.3 Elementos que conforman una red AS-i.....	17
1.5 Protocolo de comunicación Profibus DP.....	21
1.6 Conclusiones parciales del Capítulo 1.....	22
Capítulo 2: Modelos que definen el comportamiento del SAI.	23
2.1 Necesidad de modelar en Redes de Petri la dinámica de funcionamiento del SAI.....	23
2.2 Modelo que describe el Flujo de datos en el SAI.	24
2.2.1 Explicación del Modelo:	25
2.2.2 Importancia del Modelo:.....	27

2.3 Modelado en Redes de Petri de cada nivel del SAI propuesto.....	31
2.3.1 Principio de Funcionamiento de la Red ASI.	31
2.3.2 Control de los actuadores de la planta a través del bus ASI. Control de Motores y Electroválvulas.....	34
2.4 Automática Local del Proceso.	37
2.4.1 Requerimientos Funcionales del proceso (Etapa de Recepción).....	37
2.4.2 Modelo para la Selección del Modo de Operación Manual o Automático.....	39
2.4.3 Modelo que define el funcionamiento de la Automática Local del Proceso.....	40
2.4.4 Traducción de los modelos de la automática local a lenguaje de programación de PLC.....	42
2.5 Funcionamiento del Sistema Supervisorio.....	45
2.6 Atención a Alarmas y Tratamiento de Fallos.....	47
2.6.1 Modelo que define la función de Atención a Alarmas y Tratamiento de Fallos.....	48
2.7 Propuesta de automatización.....	52
2.8 Valoración económica	53
2.9 Valoración medioambiental.....	53
2.10 Conclusiones del capítulo 2.	54
Conclusiones generales:	55
Recomendaciones.....	56
Bibliografía.....	57
Anexos:	59

Introducción

La Fábrica de Piensos de la provincia Santiago de Cuba está ubicada en el poblado de Baire perteneciente al municipio de Contramaestre. Produce una variedad de piensos para el consumo animal a partir de materias primas, dosificadas de modo manual a partir de recetas elaboradas por nutricionistas.

El estado actual del poco equipamiento de instrumentación y control, además del Centro de Control de Motores, red de aire para instrumentos y accionamientos mecánicos, así como el cableado existente, están en muy mal estado, no presentan ni electroválvulas ni pistones neumáticos para el cierre y apertura de las compuertas de los silos, por lo que en el presente dicha planta solamente se opera de forma manual. El sistema que presentan actualmente no permite conocer el estado de los motores. No hay datos estadísticos ni puntuales de consumo, tiempo de trabajo, parámetros eléctricos, etc., detalles estos de gran incidencia en el mantenimiento y disponibilidad de la planta. Además, no se tiene conocimiento del nivel en los silos, por lo que esto se realiza a través del operador de forma visual. No presenta Arquitectura de Control, poseen una pizarra sinóptica y un pupitre de accionamiento, todo esto en muy mal estado técnico, lo que tiene una gran desventaja por el uso de una gran cantidad de cables de fuerza y control.

El proceso de producción de piensos consta de tres etapas fundamentales, Recepción de Materia Prima, Molienda y Mezclado. Debido al mal funcionamiento de estas áreas y a las dificultades antes mencionadas, se decide llevar a cabo una propuesta de automatización que contribuya a evitar pérdidas en la producción por fallos técnicos en motores y electroválvulas, principalmente en la primera área mencionada. Por ello se plantea el siguiente diseño metodológico:

Problema: Necesidad de diseñar un sistema automático para controlar el proceso de producción de piensos en la fábrica de Baire.

Objeto de la investigación: Sistemas de Automatización Integrada (SAI).

Objetivo general: Diseñar un Sistema de Automatización Integrada empleando las técnicas de Redes de Petri como herramienta de modelado y simulación.

Objetivo específico: Modelar en Redes de Petri la dinámica de funcionamiento de una red ASI y su interacción con los demás niveles del SAI.

Campo de Acción: El empleo de las Redes de Petri en modelado de redes industriales de automatización.

Hipótesis: Si se lleva a cabo el diseño de un SAI basado en los modelos en Redes de Petri que describen la Automática Local del proceso, la dinámica de funcionamiento del Sistema Supervisorio y el chequeo del flujo de datos, se garantizaría una solución de un sistema más eficiente para el proceso de producción de piensos de esta fábrica basado en los estándares mundiales de Automatización que demuestran la efectividad de las redes industriales dentro de este SAI.

Técnicas y métodos empleados en la investigación:

- Valoración de expertos, con el objetivo de seleccionar los sensores más apropiados para las mediciones que se realizan en la etapa de recepción.
- Método de análisis y síntesis, con el objetivo de resumir las bibliografías consultadas.
- Software: Visual Object Net, con el objetivo de modelar en Redes de Petri el funcionamiento del SAI.
- Software: Step 7, versión 5.5, para llevar a cabo la programación del algoritmo de control definido en los modelos en Redes de Petri.

Tareas:

1. Definir desde el punto de vista gnoseológico, histórico y actual los SAI.
2. Identificar las Redes de Petri como herramienta de modelado para Sistemas de Eventos Discretos (SED) y caracterizar sus propiedades funcionales.
3. Estudiar la tecnología ASI como bus de campo para sensores y actuadores que constituyen la base de las comunicaciones industriales del SAI.
4. Modelar en Redes de Petri el flujo de datos del sistema, la automática local del proceso y la dinámica de funcionamiento del Supervisorio para verificar la efectividad de las redes industriales en la automatización de la fábrica de piensos.
5. Proponer el algoritmo de control para definir la programación de los modos de operación de la planta soportados en redes industriales.
6. Presentar la solución propuesta respondiendo a la arquitectura del SAI diseñado.

Aporte:

Modelar en Redes de Petri el principio de funcionamiento de la comunicación AS-i con el objetivo de diseñar una red bien estructurada para el control de sensores y actuadores que constituyen la base del flujo de datos principal del SAI. Estos modelos posibilitan caracterizar los elementos necesarios para construir la red, teniendo en cuenta los posibles errores en la transacción entre el Maestro y cada uno de sus esclavos y permiten agrupar de forma organizada las señales que determinan el estado real del proceso. Esto aporta diseños eficientes reutilizables en sistemas de automatización similares.

Desde el punto de vista de la programación, el modelado del funcionamiento de la red AS-i permite clasificar los estados que intervienen en la función de control de cada uno de los equipos de la planta, permitiendo identificar acertadamente las señales que constituyen interbloqueo, programadas para dar el arranque o el paro de cada componente del proceso en modo automático, permitiendo identificar las condiciones para la detección de posibles alarmas y de esta forma sugerir una potencial solución, teniendo como principal objetivo la seguridad del flujo de datos en las redes industriales que soportan este SAI.

Organización de la Tesis:

La presente investigación consta de resumen, introducción, dos capítulos, conclusiones, recomendaciones, Bibliografía y Anexos.

En el primer capítulo se caracteriza el flujo productivo de la fábrica, así como los aspectos relacionados con la fundamentación teórica sobre los Sistemas de Automatización y el bus para sensores y actuadores AS-interface, además de concebir a las Redes de Petri como herramienta de modelado y simulación de Sistemas de Eventos Discretos (SED), empleados en la etapa de diseño del SAI propuesto.

En el segundo capítulo se muestra la Propuesta de Automatización, basada en los modelos en Redes de Petri que describen la Automática local de la planta y el funcionamiento de la red AS-i y del sistema Supervisorio. Se muestra además el empleo de la tecnología Siemens para diseñar la propuesta de Automatización llevando a cabo una caracterización de cada uno de los niveles de la Pirámide de Automatización. Además, se realiza una valoración económica y medioambiental de este proyecto.

Capítulo 1: Caracterización de los SAI y definición teórica de las Redes de Petri como herramienta de modelado y simulación.

Introducción.

En este capítulo se realiza una caracterización teórica de los SAI, teniendo en cuenta el empleo de las Redes de Petri como herramienta de modelado y simulación empleada en la etapa de diseño del SAI propuesto. Se describe además el Proceso Tecnológico de la fábrica de Piensos con el objetivo de proponer un Sistema de Automatización para el Área de Recepción. Para ello se obtienen los modelos en Redes de Petri que describen el funcionamiento de la planta y del Supervisorio.

1.1 Sistemas de Automatización Integrada (SAI).

1.1.1 Antecedentes.

La automatización industrial no es un fenómeno reciente, desde que la actividad artesanal comenzó a ser sustituida por la industrial, las empresas han procurado la obtención del máximo rendimiento del trabajo mediante la acción combinada de herramientas, máquinas y organización. Los beneficios que este tipo de tecnología proporciona son innumerables, desde aumentos en los niveles de producción, hasta llegar a una mayor eficacia y precisión en la ejecución de las operaciones [2]. Inicialmente los sistemas de automatización se representaban mediante sinópticos del proceso realizado en “marquetería”. Las visualizaciones se realizaban con lámparas y el accionamiento mediante interruptores, pulsadores, conmutadores, etc.

En la década de los 80 eran utilizadas tarjetas inteligentes ubicadas en el bus de los PLC's con los cuales se podían controlar de forma autónoma un monitor y un teclado, se intercambiaban datos con la CPU del PLC por el bus del PLC y ofrecía herramientas “sencillas” de configurar y programar, con prestaciones muy variadas como pantallas alfanuméricas, graficas, etc.

1.1.2 Sistemas de Automatización en la Actualidad.

Hoy en día, la tecnología de automatización está casi siempre ligada a las tecnologías de accionamiento, control y a la informática, debido al acelerado desarrollo de la tecnología de microcontroladores y ordenadores. Se utilizan Interfaces HMI para representar los sistemas de automatización, de fácil programación, Robustos y con funciones para establecer comunicación con el dispositivo de control. Estas interfaces

se realizan mediante ordenadores y software SCADAS que posee características diseñada para controlar el proceso a través de ordenadores [2,4].

En la actualidad existen tecnologías líderes en el desarrollo de soluciones integrales para procesos industriales basados precisamente en los niveles de la pirámide de automatización industrial, explicados más adelante.

Las aplicaciones de Redes de automatización integrada que existen en la actualidad se pueden organizar en los cuatro grupos siguientes [4]:

- Automatización para Edificios Inteligentes
- Automatización en Sistemas de Manufactura
- Automatización de sistemas de producción continua.
- Automatización en procesos por lotes

1.1.3 Características y Generalidades de SAI.

Los sistemas de automatización industrial comprenden el conjunto de métodos y medios de los cuales se auxilia el hombre para efectuar de manera segura el control de procesos, para lo cual no solo se nutre de los avances tecnológicos, sino también de herramientas de diseño y programación para implementar las diversas propuestas de automatización [2,4]. Desde sus inicios han sido caracterizados por poseer 3 niveles básicos explicados a continuación:

- 1. Nivel de instrumentación de campo:** es donde se encuentra la instrumentación ya sea descentralizada o no destinada a la medición de las principales magnitudes físicas del proceso, además también es el nivel donde se ubican los actuadores para el control de los accionamientos de la planta.
- 2. Nivel de control:** cuyas funciones principales están orientadas al control de la producción en los diferentes procesos, al desarrollo de secuencias de operación, a la estabilización de los parámetros principales, a la atención a fallas y a la operación local del sistema, ya sea a través de autómatas, PC o tarjetas de adquisición de datos.
- 3. Nivel de supervisión:** es donde se supervisa y controla a través de un SCADA la secuencia de trabajo de la planta. Sus funciones principales están orientadas a la supervisión del funcionamiento del sistema en su totalidad, permitiendo la integración de las variables históricas más importantes teniendo en cuenta la planificación del mantenimiento preventivo de los equipos que así lo requieran, estadísticas de la producción que sirvan en la toma de decisiones, graficar tendencias y la atención a alarmas, etc.

1.1.4 Diseño de un Sistema de Automatización Integrada (SAI).

La automatización industrial tiene una fuerte componente definida al control de sistemas industriales. En la Fig. 1.1 es presentado el ciclo de diseño de un SAI, donde el recorrido exterior es usado frecuentemente por los proyectistas, pero con el incremento de la complejidad de la automatización integrada es necesario utilizar el recorrido interno para garantizar el cumplimiento de los objetivos antes indicados.



Fig. 1.1 Ciclo de diseño de la Automatización Industrial ^[13].

El desarrollo de un SAI actual incluye las siguientes etapas: ^[11]

- 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 con facilidades de simulación. Como los PLC's son los dispositivos de mayor uso ^[5], la IEC61131 es la norma más usada, así como sus ambientes de edición y simulación.
- 3. Implementación:** En un SAI se requiere crear estructuras jerárquicas en redes buscando eficiencia y facilidades de mantenimiento para reducir los costos del proyecto. Si la parte de hardware correspondiente al diseño está funcionalmente bien definida, el SAI estará determinado por el software asociado a este. El 83% del costo de los proyectos de automatización se centra en el desarrollo, verificación y validación del software ^[5]. Para esto son fundamentales los métodos formales.

1.1.4.1 Métodos formales.

Los métodos formales de diseño de SAI ^[13] utilizan la teoría de diseño de Sistemas de Eventos Discretos (SED) para su modelado, verificación y validación formal ^[3]. Estos sistemas se caracterizan por representar el comportamiento dinámico a través de los cambios de estado generados por la ocurrencia de eventos, y están compuestos por dos elementos principales:

- Estados: representado por conjuntos discretos (equipamiento ocupado, ocioso, mantenimiento; mensajes en el canal de comunicación; etc.).
- Evolución del sistema: por la ocurrencia de eventos que provocan transiciones en los estados del mismo. Entonces el sistema es afectado por un conjunto de Eventos (llegada de pieza, inicio de operación, temperatura en el límite especificado, etc).

Existen dos clases principales de modelos formales para SEDs:

1. Modelos Lógicos: Orientados al análisis del comportamiento lógico – secuencial.

- Redes de Petri (RdP)
- Autómatas Finitos Deterministas (AFD)

2. Modelos de Comportamiento: Orientados al análisis de desempeño del sistema.

- Modelos Analíticos: Cadenas de Markov, Redes de Filas, etc.
- Modelos p/ Simulación: Arena, Automod, Promodel, etc.

Redes de Petri: constituyen la herramienta de mayor popularidad para el análisis del comportamiento lógico y secuencial de sistemas, tanto para el campo de la computación como para los sistemas de automatización industrial, por garantizar una representación gráfica y sencilla, fundamentada matemáticamente [9,10,14].

Las Redes de Petri permiten una descripción compacta de la planta y pueden representar de forma gráfica las especificaciones y el objeto de análisis. Los modelos sobre Redes de Petri pueden ser utilizados tanto para el análisis de las propiedades comportamentales y la evaluación de su realización, como para la construcción sistemática del controlador [7,8].

Autómatas finitos: son un conjunto de diagramas representados mediante un grafo dirigido y 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, además representan condiciones desde el punto de vista lógico [20]. Su fundamentación matemática y su representación compacta del comportamiento del sistema. Es preciso modelar los estados que son importantes para la correcta secuencia lógica del sistema y su naturaleza gráfica ayuda en la comprensión de sistemas complejos y resultan una opción útil para describir sistemas concurrentes, asíncronos, o paralelos [12].

Aunque las Redes de Petri y los Autómatas Finitos tienen la misma fortaleza teórica-matemática para representar los modelos, las primeras tienen mayor expresividad

gráfica para las aplicaciones de automatización industrial, porque las condiciones de disparo de las transiciones controlables aparecen de forma explícita en lugares auxiliares con arcos habilitadores e inhibidores y no simplemente etiquetados sobre los arcos como en los Autómatas Finitos [16,17].

Teniendo en cuenta ambas definiciones, se decide escoger las Redes de Petri como el método de análisis del comportamiento lógico y secuencial de este sistema. Pues sustentado en que GHENeSys incluye otros elementos que la acercan más a la automatización industrial con PLC's [8,15] debido a que se establece una jerarquía entre macro-elementos muy similar a la estructuración de programas, se valida esta elección. Ya que existe una función (Q) que relaciona acciones sobre el proceso con los lugares y transiciones facilitando la interpretación física y la traducción a lenguajes de programación.

De esta manera se evidencia que las Redes de Petri resulta una solución más eficiente para enfrentar la realización de un SAI pudiendo modelar los estados del proceso y el funcionamiento del SCADA.

1.2 El empleo de las Redes de Petri para motivar el modelado formal de SAI.

1.2.1 Redes de Petri.

Las Redes de Petri (RdP) fueron introducidas por Carl Adam Petri en 1962. Su éxito se debe básicamente a la simplicidad de su mecanismo básico. Constituye un término agrupador que designa un amplio número de modelos lógicos y secuenciales de sistemas, procedimientos, técnicas y patrones descriptivos relacionados entre sí, en el sentido de que están todos basados en el mismo principio:

Transición–Acción / Condición-Evento [9,23].

Esta herramienta permite la identificación de los aspectos básicos de los sistemas distribuidos tanto conceptual como matemáticamente. Es una herramienta de modelado de sistemas secuenciales discretos y concurrentes. Permiten visualizar el comportamiento dinámico de un sistema con simulación gráfica, siendo un grafo orientado, formado por **Lugares** que determinan las acciones, **Transiciones**, para llevar el sistema de un lugar a otro y **Arcos orientados** que unen lugares con transiciones o viceversa, donde las fichas situadas en los lugares se denominan **Marcas** y determinan el estado del sistema [24].

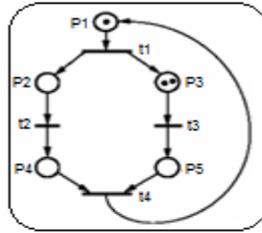


Fig. 1.2 Ejemplo de una red de Petri.

La estructura de una Red de Petri estándar es un grafo bipartito que incluye un conjunto de lugares P , de transiciones T y de arcos dirigidos A . Un lugar p es una entrada para una transición t si existe un arco incidente a la transición (p,t) . Un lugar p es una salida de una transición t si existe un arco incidente de la transición al lugar (t,p) . El conjunto de arcos puede ser particionado en el conjunto de arcos de entrada a transiciones A_i y el de arcos de salida de transiciones A_o . Estos conceptos permiten representar la evolución del comportamiento dinámico de la red conocido como reglas de disparo de las transiciones. [23]

1.2.2 Propiedades de las redes de Petri.

En el análisis y diseño de modelos basados en RdP se tienen en cuenta un conjunto de propiedades [23], pudiendo clasificarse éstas como:

Propiedades funcionales: Son aquellas propiedades dependientes del marcaje inicial y reflejan el comportamiento dinámico del sistema.

- **Limitación**
- **Vivacidad**
- **Reversibilidad**
- **Persistencia**

En el análisis de modelos PN la propiedad de **Vivacidad** de la red es la que permite la navegación entre pantallas, si generalizamos esta capacidad a todo el sistema, y es aquí donde pueden detectarse partes de la red que detienen su funcionamiento (como lazos cerrados o bloqueos), permitiendo eliminar estas situaciones anormales en el programa del PLC desde esta etapa inicial de diseño.

También los sistemas de control no pueden tener un comportamiento ilimitado en la mayoría de sus elementos por las propias limitaciones físicas del sistema, fundamentalmente capacidad de almacenes o recursos compartidos. Por tanto, también debe vigilarse la propiedad de **Limitación** funcional de la red [25].

Reversibilidad y estado particular: La red tiene que garantizar su repetitividad, ya que el trabajo de todo sistema de control es cíclico y más en el caso de los PLC, porque también tiene que garantizar que este comportamiento no sea aleatorio, sino que sea controlable.

1.2.3 Modelado de SAI a través de las redes GHENeSys.

En las aplicaciones con PLC's es más aconsejable el uso de GHENeSys [24, 25, 26], redes jerárquicas extendidas para el modelado de SAI que garantizan mayor uso de las herramientas de análisis y síntesis. También las redes GHENeSys están más cerca de los modelos más simples porque ellas permiten que los **Lugares auxiliares** no sólo representen mediciones de sensores del proceso, también estados de información que vienen de otras subredes o de partes de la misma red, lo que simplifica la estructura de redes complejas y lo que permite mayor efectividad de las herramientas clásicas de análisis y síntesis de RdP para estos controladores, como se muestra en la Fig. 1.3.

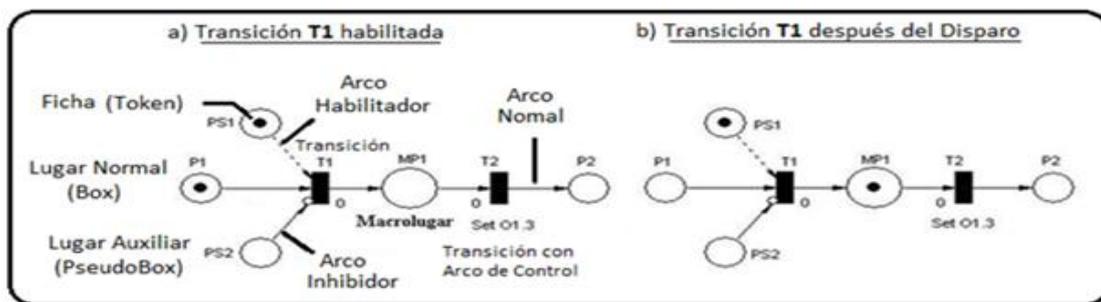


Fig. 1.3 Elementos y reglas de disparo de PN extendidas GHENeSys.

La definición de **subredes (Macro-elementos)** en GHENeSys ayuda a la modularidad del modelo y garantizan reusabilidad en la conformación de redes jerárquicas. Estos macro-elementos también permiten abarcar el campo de las aplicaciones no binarias en PLC's sin tener que pasar a redes de alto nivel y con una mayor semejanza con los lenguajes de PLC's. El modelado es desarrollado de forma modular mediante el uso de subsistemas y funciones con estructuras típicas. Por ejemplo, los motores son modelados como una estructura típica de tres estados: Parado, Funcionando y Fallo.

La dinámica de las redes es muy importante para el modelado de sistemas de automatización. En GHENeSys son aplicadas las mismas tres reglas de disparo de las PN clásicas, adicionando el caso particular de marcación permanente en los Lugares Auxiliares. En el disparo, las transiciones de Macro-elementos se comportan como elementos simples pero la interpretación es realizada con una llamada a una subred que desarrolla una operación determinada. La metodología para el proceso de diseño

de sistemas de automatización con PLC's ya es aplicada en nuestro país en algunos trabajos del ámbito industrial con efectividad [8]. Los pasos fundamentales de la metodología para garantizar la efectividad del modelado están concentrados en la verificación de propiedades y en la validación de requisitos funcionales. La verificación de sistemas informáticos requiere de una plataforma para el modelado del sistema y de un método de verificación. En este trabajo se utiliza como plataforma de modelado el Visual Object Net 2.7 (VON2.7) [11] con los modelos de RdP extendidas GHENeSys, cumpliendo con las propiedades funcionales antes mencionadas.

Para llevar a cabo la realización de los modelos que describen el funcionamiento del SAI, es necesario demostrar el cumplimiento de las propiedades funcionales (Vivacidad, Limitabilidad y Reversibilidad) en cada uno de los modelos, teniendo en cuenta el empleo de Macro-elementos para representar estados importantes en el sistema y lugares auxiliares que constituyen habilitadores e inhibidores de las condiciones funcionales del proceso. Con este principio se asegura modelar subredes vivas, que interactuando entre ellas aseguran la conformación de un sistema con ausencia de bloqueos, cíclico, limitado y controlable [1].

Limitabilidad: Una RdP se dice ser limitada (definida) si el número de tokens en cada lugar no excede su capacidad para cualquier estado alcanzable desde uno inicial. Los lugares en esta red son usados a menudo para representar buffers y registros para almacenar datos intermedios [1, 23].

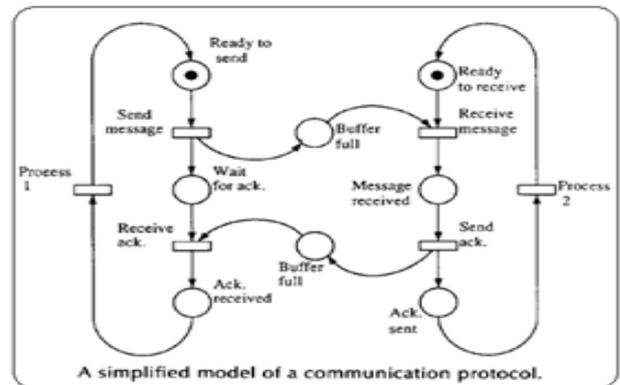


Fig.1.4 Sub modelo para modelar la comunicación entre dos sistemas.

Vivacidad: ausencia de lazos que limitan la dinámica de la red a una sola parte, sin caer nunca en los restantes lugares. El modelo que permite la navegación entre pantallas evita lazos cerrados sin detener el funcionamiento a una parte de la red.

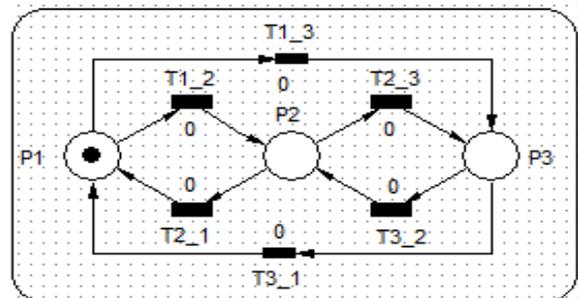


Fig.1.5 Sub modelo que describe la navegación entre las pantallas del SCADA.

Reversibilidad: el trabajo cíclico del PLC debe garantizar un comportamiento No Aleatorio, sino Controlable y para cualquier marcado alcanzable es posible volver al Estado Inicial. Las redes secuenciales son siempre vivas y reversibles.

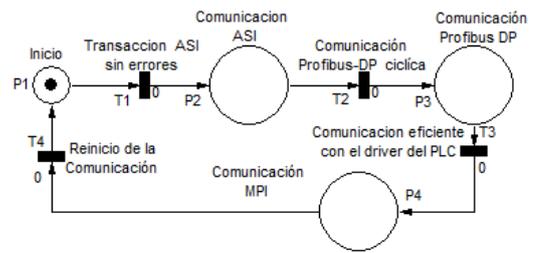


Fig. 1.6 Sub modelo empleado para el Chequeo de las Comunicaciones.

Persistencia: Una RdP es persistente si para dos transiciones habilitadas el disparo de una transición no habilitará la otra. Una vez que una transición es habilitada en una red persistente, esta permanecerá así hasta que se dispare [1, 23].

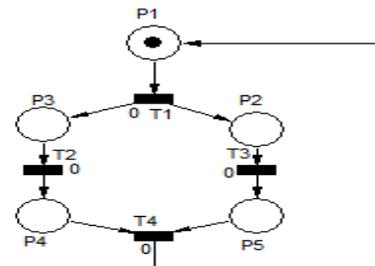


Fig. 1.7 Sub modelo empleado en el programa para efectuar acciones paralelas.

Finalmente se puede definir que la validación consiste en comprobar que se cumplen las propiedades de Vivacidad, Limitación y Reversibilidad en los modelos realizados, demostrando de esta manera que se cumplen las exigencias funcionales del sistema.

1.2.4 Traducción de las redes modeladas en GHENeSys a lenguajes de programación.

La traducción no es más que la transformación de las subredes modeladas a códigos de programación equivalentes basado en el algoritmo de control diseñado, teniendo en cuenta la estructura típica de cada red en función de la problemática modelada y su equivalencia en el lenguaje de programación de PLC seleccionado (LD, ST, FBD).

El uso de las redes GHENeSys para el modelado formal de SAI tiene muchas ventajas por su analogía al modelo de programación IEC61131 compatible. A continuación, se explican las reglas generales de analogía de los elementos de la red jerárquica extendida de GHENeSys que permiten construir el cuadro de equivalencias entre los modelos y las secciones de programas que quedan como herramientas de la traducción [25].

Para los **Boxes** (lugares) se mantiene la equivalencia a un bit interno del PLC. Es decir, el estado 0 o 1 corresponderá con el estado de una variable binaria interna. Los Boxes y Pseudoboxes dan las condiciones de disparo de las transiciones del modelo en

GHENeSys, entonces ellos corresponden con la parte condicional de las secciones de programas resultantes.

Los Pseudoboxes corresponderán a diferentes tipos de señales de acuerdo con su uso:

- Si representan mediciones de sensores del proceso, ellos corresponderán con entradas del PLC (que pueden usar un identificador simple o su direccionamiento directo).
- Si representan informaciones que vienen de otras subredes u otras partes de la propia red, ellos corresponderán con variables binarias internas (bits internos o banderas).
- Si representan un bit de salida de un temporizador o contador, ellos corresponderán con variables representadas por el identificador del temporizador seguido por "Q".
- Si representan la salida binaria de algún bloque funcional, ellos tendrán el nombre del bloque funcional seguido por "nombre" de la salida del bloque.

Una equivalencia típica muy utilizada la constituye el empleo de operaciones lógicas entre estas señales, que estarán en dependencia de la estructura que las entrelazan con las transiciones que luego se disparan. Por ejemplo, si las señales entran a una única transición, entonces ellas conforman una operación lógica **AND**, y si llegan a varias transiciones que se unen en un único Lugar, entonces serán un **OR** ^[25].

La traducción de esta parte condicional para lenguajes LD e IL está prácticamente implícita en la definición de los lenguajes IEC 1131 compatibles. En LD ellos son contactos en serie o paralelo o **AND** u **OR** respectivamente. En lenguaje IL son, en realidad, instrucciones que representan estas funciones binarias. En el caso de traducción para lenguaje **ST** estas condicionales se convierten en las expresiones de condicionamiento de alternativa **IF** ^[25].

El Macro-elemento, empleado para no perder formalidad en el modelo y la existencia de Macro-Lugares y Macro-Transiciones, equivalen a la creación de subredes propias representadas en el modelo por un único elemento de mayor tamaño, indicando la ejecución de varias acciones que pueden ser complejas, sin afectar la interrelación con otros elementos del modelo, por este motivo un bloque funcional podrá ser representado por Macro-elementos de entrada, de salida o de ejecución ^[25].

Esto permite simplificar la representación de temporizadores (TON, TOF, TP) y contadores (CU, CD y CUD) y operaciones con bytes, porque estas son encapsuladas en el Macro-elemento y no exigen ser modeladas. Los Pseudoboxes condicionan el

momento de disparo de la transición de salida del bloque para la ejecución del resultado de la acción.

De este modo se mantiene la potencialidad de verificación gráfico-analítica del modelo para la representación de cualquier aplicación de PLC's, que posibilita realizar su traducción a lenguajes de programación con el objetivo de tener una programación estructurada modularmente y que responde fielmente a las exigencias del proceso.

1.3 Descripción del Flujo Tecnológico y caracterización de los elementos que conforman el proceso.

La fabricación de piensos consiste en la mezcla de materias prima previamente molidas, como cereales u otros cultivos, se ejecuta de manera secuencial y se tiene en cuenta la receta del tipo de pienso que se quiere producir, decir además que consta de tres áreas principales:

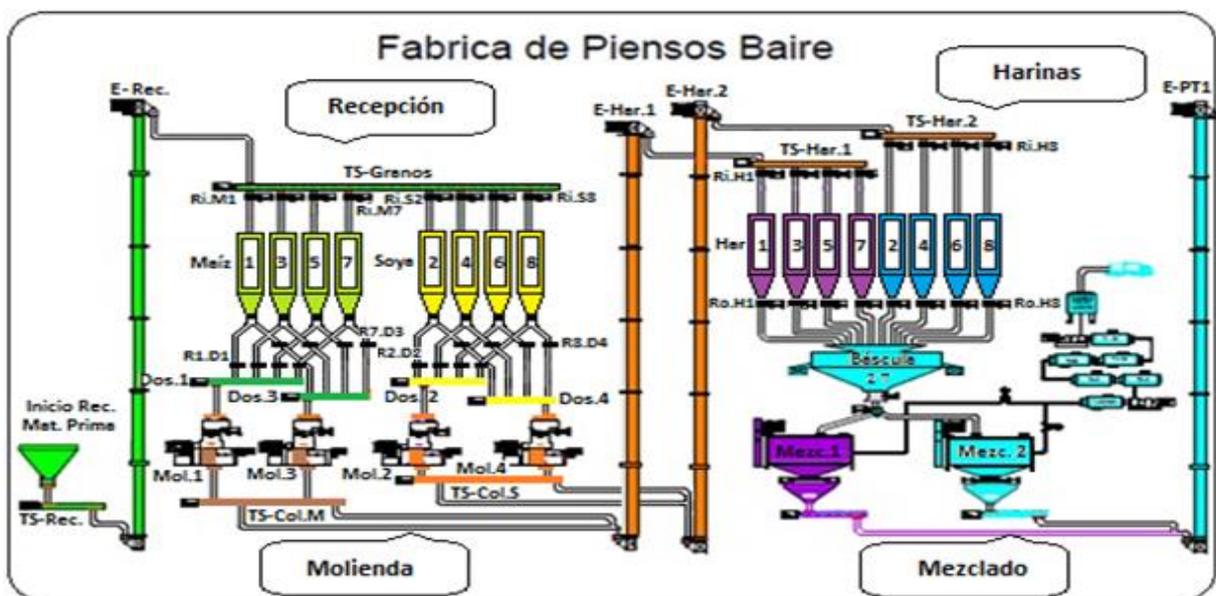


Fig.1.8 Flujo Tecnológico de la Fábrica de Piensos de Baire.

Recepción: es el área donde se almacena la materia prima seleccionada, es decir Soya o Maíz, las cuales se trasladan hacia Silos de almacenamiento de granos a través de una estera transportadora y un elevador de cangilones.

Molienda: es donde se muelen dicha Materia Prima, ya que es preciso molturarlos antes de la mezcla con el resto de las sustancias para facilitar la homogenización de la mezcla y luego son transportados hacia la siguiente área.

Mezclado: consiste en la homogenización de la mezcla, añadiéndole ciertos productos aditivos de manera manual como sales, calcio o melaza, minerales, vitaminas, urea, etc., obteniendo como producto final el pienso listo para empaquetar.

El proceso productivo se realiza actualmente de manera manual, por a la poca existencia de elementos de instrumentación y control. Debido a esto se diseña una

propuesta de automatización que mejorará el estado técnico del proceso. A continuación, se describe de forma teórica los elementos que intervienen en dicha propuesta, analizando su principio de funcionamiento y sus principales características.

1.3.1 Medición de Nivel máximo y mínimo en los silos (Sensor de Nivel capacitivo).

El sensor que se decidió emplear en nuestra propuesta de automatización (MiniCap) funciona bajo este principio por consiguiente a continuación se explica su funcionamiento. ^[19]

Es una medición de nivel bastante utilizada y el principio de medición consiste en que una sonda metálica (aislada) y la propia pared del depósito actúan como dos placas de un condensador. La capacidad del condensador depende del medio que haya entre la sonda y la pared. Si sólo hay aire, es decir, si el depósito está vacío, la capacidad del conductor es baja. Cuando parte de la sonda esté cubierta por el producto, la capacidad se incrementará.

1.3.2 Sensor Inductivo para detectar fallo de la cadena del elevador.

Un elevador de cangilones es un mecanismo que se emplea para el acarreo o manejo de materiales a granel verticalmente utilizadas generalmente para el transporte de granos. Sería el equivalente vertical de la cinta transportadora y no basta con obtener la confirmación eléctrica del funcionamiento del motor, es necesario también reconocer la confirmación mecánica del funcionamiento de las cadenas del elevador indicando la transportación del material hacia la etapa siguiente. En esta aplicación se emplea un sensor inductivo de 24 VDC, conectado a tres hilos y acoplado mecánicamente a pocos milímetros de la cadena del elevador, con el objetivo de dar una señal hacia el PLC mientras que los cangilones están en movimiento.

1.3.3 Actuadores de la Planta.

En esta planta existen dos tipos principales de actuadores:

- **Electroválvulas:** son empleadas para abrir y cerrar las compuertas (Raceras) de los Silos durante el proceso de descarga de Materia Prima (MP).
- **Motores:** son utilizados para el movimiento de la MP a través de motores acoplados a un tornillo sin fin (TS). Para transportar la MP verticalmente se emplean elevadores (E), y para dosificar el suministro de material se utilizan los tornillos Sin Fin, pero se les incorpora un control de velocidad permitiendo graduar el abastecimiento de MP y son

conocidos como dosificadores (TS-Dosif). Para efectuar la trituración de MP se trabaja con Molinos (Mol) permitiendo disminuir la granulometría del producto de manera eficaz. Cada uno de estos motores son controlados por Contactores (KM), Arrancadores Suaves (AS) o Variadores de Velocidad (VF) según el tipo de arranque y el principio de control de cada motor.

1.3.4 Bus de campo para sensores y actuadores.

El cableado de los sensores y actuadores supone uno de los procesos más laboriosos en el montaje de los sistemas de automatización, además, suele ser una de las mayores fuentes de errores en la puesta en marcha de la instalación. Conforme el sistema se complica, la gran cantidad de sensores y actuadores que se requieren, y la necesidad de emplear 2, 3, 4 o incluso más hilos por sensor, no sólo hacen que la dificultad se incremente exponencialmente, sino que supone un aumento considerable del coste final de la instalación.

En esta etapa se incluye un bus de campo para actuadores y sensores diseñado para transmitir las señales de campo a través de una interfaz AS-i. Esta red recorre toda la planta y mediante estaciones esclavas agrupa las entradas y las de salidas digitales, para almacenarlas en la memoria de una pasarela DP/ASI.

1.4 Bus ASI.

Un sistema industrial formado por redes AS-i es considerado como el más económico e ideal para la comunicación entre actuadores y sensores, siendo sistema estandarizado (EN 50295) y abierto que conecta los dispositivos más simples de las soluciones de automatización. La conexión de los elementos puede ser realizada en estructura de árbol, estrella, lineal o en combinación con las anteriores. Debido a que no existen conexiones convencionales reducen el número de interconexiones en bornes y conectores, reduciendo costos, tiempo de montaje y errores ^[3].

Anteriormente, los sensores y actuadores tenían que ser conectados al controlador a través de terminales, conectores y bloques de terminales. AS-i proporciona una reducción en los costos de instalación y mantenimiento. Ahora, un cable estandarizado con 2 hilos permite el intercambio de información y al mismo tiempo la alimentación de los equipos. Los esclavos son conectados directamente en el bus sin la necesidad de interconexión adicional.

1.4.1 Beneficios de una red AS-i

- **Simplicidad:** Una **rede** AS-i es muy simple, requiere apenas solo un único cable para conectar los módulos de entradas y salidas de cualquier fabricante. Los usuarios de una red AS-i no necesitan conocimientos a fondo en sistemas industriales o protocolos de comunicación [3].
- **Desempeño:** Los sistemas AS-i son eficientes y muy rápidos, haciéndolos capaces de sustituir los sistemas grandes y con altos costos. Existen maestros AS-i, especialmente, desarrollados para comunicarse con los sistemas de control y proporcionan una integración perfecta entre las tecnologías existentes. Lo mejor de todo es que se logra de una forma simple y sencilla [3].
- **Flexibilidad:** La expansibilidad es muy fácil de conseguir, solo tiene que conectar un módulo, direccionar y, luego conecte el cable de la red, verificando si el indicador de la fuente de alimentación está conectado y, luego si está conectado al siguiente módulo. La red AS-i soporta cualquier topología de cableado: estrella, bus, árbol, anillo o cualquier otra configuración con hasta 100 metros de cable. Mediante la adición de repetidores es posible expandir el sistema hasta 300 metros y no necesita de terminadores en los puntos extremos [3].

1.4.2 Características principales de una red AS-i.

El nombre Actuator Sensor Interface representa a su propio concepto. Es una solución simple y elegante para la integración de sensores y actuadores discretos en un sistema de control de procesos. Esta red posee una serie de características que son representadas de la siguiente manera. [3]

- **Compatibilidad:** Sensores y Actuadores de diferentes fabricantes pueden ser conectados a una interfaz digital serial estandarizada.
- **Control de acceso al medio:** Un solo maestro y realiza un sondeo cíclico.
- **Direccionamiento:** Esclavos reciben un direccionamiento permanente del maestro.
- **Topología:** Sin restricciones (lineal, anillo, estrella o árbol).
- **Medio de transferencia:** Dos cables no-trenzados y sin blindaje para datos y energía (24 VDC), típicamente hasta 200 mA por esclavo, y hasta 8A por bus.
- **Instalación rápida:** Por medio de conectores auto-perforantes.
- **Longitud del cable:** Máximo de 100 m o hasta 300 m con el uso de repetidores.
- **Señales y alimentación:** Están presentes en un mismo bus (24VDC).
- **Número de esclavos:** Hasta 31 esclavos por Máster.

- **Telegramas:** Contiene el direccionamiento y la respuesta directa del esclavo.
- **Datos:** 4 entradas y 4 salidas para cada esclavo.
- **Carga útil:** Transmite 4 bits/esclavo/mensaje. Todos los esclavos son llamados secuencialmente por el maestro y reciben 4 bits de datos. Cada esclavo responde inmediatamente con 4 bits de datos. Tiempo de ciclo: 10 ms para la versión 2.1.
- **Detección de errores:** Detección eficiente y retransmisión de telegramas incorrectos.
- **Chip AS-Interface:** 4 E/S configurables para datos, 4 parámetros de salidas y 2 salidas de control.
- **Funciones del maestro:** Barrido cíclico en todos los esclavos, transmisiones de datos para esto y para la unidad de control (PLC). Inicialización de la red, identificación de los esclavos, diagnóstico de los esclavos y de datos transferidos. Además, los informes de errores en el controlador y la dirección de los esclavos sustituidos.
- **Válvulas:** Son instaladas directamente en la aplicación, reduciendo tuberías y aumentando la velocidad de respuesta de los actuadores.
- **Bajo costo:** Bajo costo de conexión por esclavo y elimina módulos de entradas y salidas en el PLC.
- **Confiabilidad:** Alto nivel de confiabilidad operacional en ambientes industriales agresivos.
- **Estándar abierto:** Elaborado por diversos fabricantes, afiliados a la Asociación Internacional AS-i, cuyo protocolo de transmisión es normalizado.
- **Opcional:** Cable de alimentación para salidas y control de paro.

1.4.3 Elementos que conforman una red AS-i

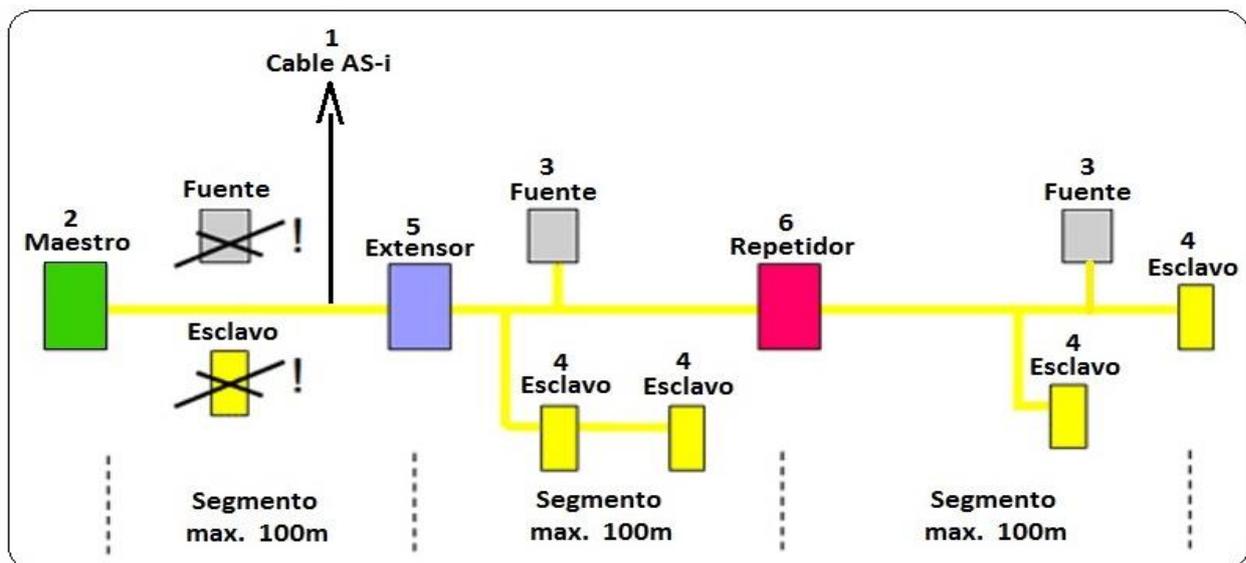


Fig.1.9 Esquema de una red AS-i con todos sus componentes.

Componentes que conforman una Red AS-i.

1. Cable Flexible Estándar:

El cable AS-i, no blindado, no trenzado, tiene dos conductores paralelos y transmite tanto datos como la alimentación de los esclavos. Su cubierta externa es de color amarillo y se caracteriza por su forma geométrica, y fue diseñada para evitar la conexión con polaridad inversa [3], como se muestra en la Fig.1.10.

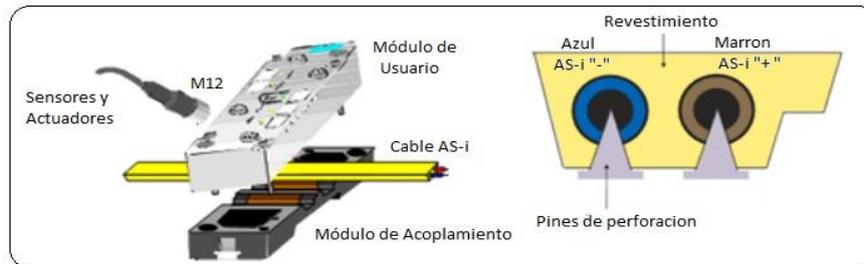


Fig.1.10 Cable AS-i.

2. Fuente de Alimentación.

La fuente de alimentación AS-i tiene cuatro funciones esenciales en el sistema:

- **Fuente de Alimentación:** La fuente de alimentación AS-i trabaja con una tensión de 29,5V a 31,6V DC y en condiciones normales de operación, suministra una corriente de 0A a 8A. La fuente suministra alimentación a los esclavos y parcialmente al maestro a través de dos hilos, y el mismo cable para transmisión de datos AS-i, y se puede conectar en la red en cualquier punto. En las líneas largas, las caídas de tensión deben ser consideradas y generalmente no debe ser mayor de 3V. La fuente posee internamente un circuito de protección de sobrecarga para limitar la corriente [3].
- **Balanceamiento:** La fuente también es responsable de balancear a la red AS-i. AS-i es operado como un sistema simétrico, **no aterrizado**. Para inmunidad a los ruidos, el cable AS-i, debe ser instalado de forma simétrica lo más posible. Esto es posible a través del circuito de Balanceamiento mostrado en la Fig.1.11. La conexión de blindaje debe ser conectada en un punto apropiado de la máquina o sistema. También ese punto debe ser conectado a tierra (GND) del sistema.

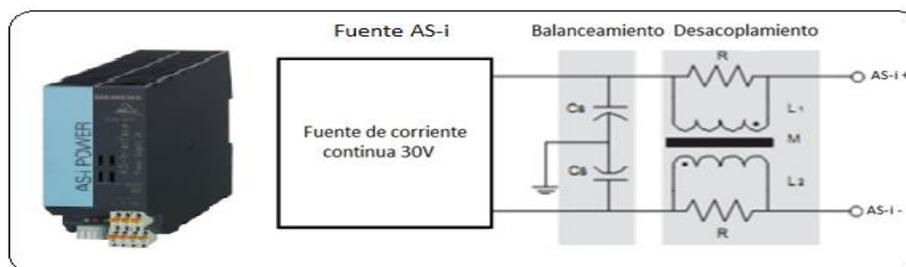


Fig. 1.11 Fuente de alimentación AS-i.

- **Desacoplamiento de los Datos:** La tercera función de la fuente de alimentación es proveer el desacoplamiento de los datos. La red de desacoplamiento, que en general se encuentra en el mismo módulo de la fuente de alimentación, consiste de dos inductores de $50\mu\text{H}$ cada uno (L1 y L2) y dos resistores en paralelo de 39Ω cada uno. Los inductores realizan una operación de diferenciación sobre los pulsos de tensión para convertir los pulsos de corriente generados por los transmisores conectados a la red. Al mismo tiempo, previenen un corto circuito en el cable. El acoplamiento entre los inductores debe ser lo más cercano posible de 1 lo que equivale a decir que la inductancia mutua debe ser de aproximadamente de $200\mu\text{H}$.
- **Seguridad:** La cuarta función es una consideración de seguridad. El sistema AS-i fue diseñado como un sistema para tensiones pequeñas con aislamiento seguro. Esto significa que de acuerdo con los estándares IEC relevantes, "aislante seguro" es requerida de la fuente entre la red de alimentación y la red AS-i.

3. Esclavos AS-i.

El chip esclavo AS-i permite que los sensores, actuadores o módulos puedan ser conectados al bus AS-i como un dispositivo esclavo, reconociendo los comandos de salida del maestro y enviando información de respuesta. Una gran variedad de sensores y actuadores utilizados últimamente en la automatización, permiten un bajo costo por conexión en dispositivos esclavos AS-i. En el caso de los dispositivos analógicos, los datos de más de 4 bits de información útil por ciclo. En este caso, los datos son divididos y enviados en varios ciclos.

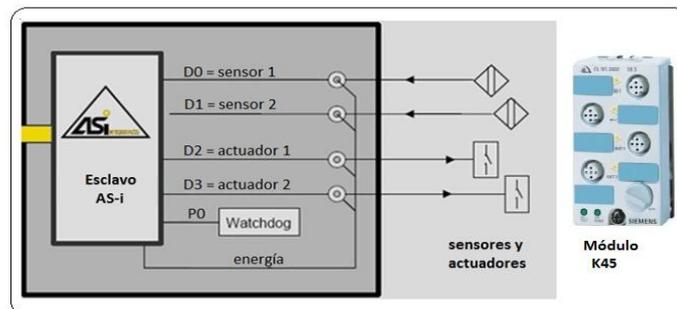


Fig. 1.12 Esclavo AS-i: K45 2 entradas-2 salidas.

En la Fig.1.12 se representa un módulo con dos entradas para sensores y dos salidas para actuadores binarios. Independientemente de la forma en que fue implementado el esclavo. En la ruta de alimentación las señales de datos son separadas del componente de alimentación por medio de una inductancia. Para implementarla en un CI, esta inductancia es realizada electrónicamente, y garantiza que el esclavo mantenga una resistencia suficientemente alta en la banda de frecuencia de los datos de comunicación [3].

4. Trama de un dato AS-i.

El acceso al medio del tipo maestro-esclavo es empleando el principio del sondeo cíclico, en el cual el maestro envía un telegrama que es recibido por el esclavo en una dirección particular y el esclavo con esta dirección responde dentro del tiempo previsto. Esta operación es designada como una transacción. En el caso de las pequeñas perturbaciones en la red, el maestro puede, por ejemplo, repetir telegramas a la dirección del cual no recibe respuesta, o respuesta invalida. Los mensajes AS-i, por lo tanto, son de dos tipos: las que son enviadas por el maestro y las respuestas del esclavo. La Fig. 1.13 ilustra una transacción, constituida por: una solicitud del maestro, una pausa del maestro, una respuesta del esclavo y una pausa del esclavo [3].

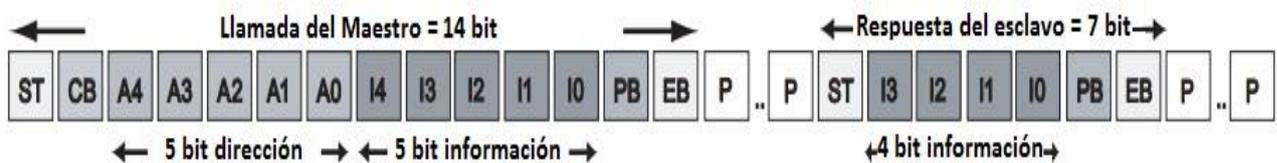


Fig. 1.13 Formato de trama AS-i.

La solicitud del maestro se extiende siempre a 14 tiempos de bit, y la respuesta del esclavo a 7. Un tiempo de bit corresponde a un intervalo de tiempo uniforme de 6µs. La pausa del maestro debe estar en un intervalo de un mínimo de 2 tiempos de bit y un máximo de 10 tiempos de bit. Si el esclavo está sincronizado, puede comenzar a transmitir su respuesta lo más pronto después de más de 2 tiempos de bit, en caso contrario necesitara más de dos tiempos, hasta que se pueda estar monitoreando la pausa del maestro en este período para cualquier información adicional antes de que pueda aceptarla como válida. Sin embargo, si el maestro no ha recibido respuesta dentro de un máximo de 10 tiempos de bit, puede asumir que no habrá respuesta e iniciar la transacción siguiente. La pausa entre el final de una respuesta del esclavo es la siguiente solicitud del maestro y debe estar entre 1,5 y 2 tiempos de bit [3].

5. Maestro AS-i.

El bus AS-i es gobernado por un solo Maestro, esto significa que por cada red AS-i solo puede haber un Maestro controlando el intercambio de datos de un máximo de 31 esclavos. La técnica de acceso es por polling, es decir, el maestro AS-i encuesta a cada esclavo del bus y espera por una respuesta. El bus AS-i acepta distintos tipos de maestros como Controladores Lógicos Programables (Siemens Simatic o Telemecanique TSX) pero en este caso se designa como módulo Maestro AS-i un dispositivo con capacidad de conexión directa con el bus AS-i o Gateway, cuya función es establecer la comunicación AS-i con una red Profibus DP presente en el nivel de control [3].



Fig. 1.14 Pasarela AS-i/DP

La Pasarela DP/AS-i de Siemens, mostrada en la Fig. 1.14, funciona como maestro para la red AS-i y al mismo tiempo como esclavo del sistema para el bus Profibus DP. El funcionamiento de este dispositivo se basa en suministrar señales binarias al sistema de bus superior para su posterior tratamiento con el programa del PLC. Contiene además un método de encuesta a cada esclavo de la red que le permite conocer el buen funcionamiento del mismo o la ausencia de alguna estación esclava en la propia red AS-i. Esta pasarela designa un espacio de memoria de entrada y salida para almacenar las señales digitales del campo, realizando de esta forma una imagen del estado del proceso en la memoria del programa del PLC a través de la comunicación Profibus DP [3].

1.5 Protocolo de comunicación Profibus DP.

Profibus DP especifica el uso de pares trenzados de cobre apantallados empleando la norma RS-485. Las distancias potenciales de bus van desde 100 m a 24 Km (con repetidores y fibra óptica). La velocidad de transmisión puede estar entre 9600 bit/s y 12 Mbit/s en dependencia de las distancias y la calidad del cable. Se admiten hasta 32 estaciones por segmento, con el uso de repetidores se pueden ampliar hasta 127 estaciones. Utiliza mensajes de hasta 244 bytes de datos. La topología es tipo bus con terminadores en los extremos, pero se admiten derivaciones [18].

Características básicas: La red eléctrica utiliza un cable bifilar trenzado apantallado como medio de transmisión. La Interfaz RS 485 funciona con diferencia de tensión. Por este motivo, es más inmune a las interferencias que una interfaz de tensión o de corriente. Las plataformas hardware utilizadas para soportar Profibus se basan en microprocesadores de 16 bits más procesadores de comunicaciones especializados [18]. En esta aplicación se emplea el protocolo Profibus para la transmisión de la información de campo hacia el nivel de control, siendo el medio de comunicación más adecuado para conectar las Pasarelas AS-i que no son más que estaciones esclavas en la red Profibus permitiendo almacenar el estado de las señales de campo en la memoria del PLC con un nombre y dirección única.

1.6 Conclusiones parciales del Capítulo 1.

1. La selección de las Redes de Petri como herramienta para modelar sistemas de eventos discretos permite definir apropiadamente los estados para controlar la dinámica de funcionamiento del SAI.
2. Caracterizar el flujo productivo de la planta permite seleccionar los sensores teniendo en cuenta el principio de medición más adecuado para cada aplicación.
3. Elegir el bus de campo AS-i para el control de los sensores y actuadores de esta planta constituye una solución económica que garantiza el intercambio de información de manera rápida y efectiva.
4. Describir la transacción AS-i permite conocer los momentos de la trama entre un maestro y un esclavo, posibilitando identificar los posibles errores presentes en la comunicación.

Capítulo 2: Modelos que definen el comportamiento del SAI. Explicación, Importancia e Interpretación.

Introducción.

En este capítulo se explica la necesidad de modelar en Redes de Petri la dinámica de funcionamiento del Sistema de Automatización Integrada propuesto para la etapa de Recepción de piensos de la fábrica de Baire teniendo como resultado la presentación de la arquitectura de un sistema que responde a los estándares mundiales de automatización y control. Luego se lleva a cabo el diseño de los modelos que definen el principio de funcionamiento de cada nivel en el SAI. Luego se desarrollan los modelos que definen la automática local del proceso, el flujo de datos del sistema y el funcionamiento del Supervisorio y al mismo tiempo se explica cómo se lleva a cabo la interacción entre ellos. El desarrollo de estos modelos incorpora el modelado de los conceptos del mantenimiento preventivo, el chequeo de las comunicaciones para la detección de posibles fallos de comunicación y la atención de alarmas y tratamiento de fallos. Finalmente se presenta una propuesta de automatización (solución) empleando tecnología Siemens basada en cada uno de los modelos desarrollados con el objetivo de satisfacer los requerimientos funcionales del sistema.

2.1 Necesidad de modelar en Redes de Petri la dinámica de funcionamiento del SAI.

El objetivo de llevar a cabo el modelado de los niveles que conforman un sistema de automatización es en primer lugar elegir la correcta estrategia de control que garantice un buen funcionamiento de la automática local de proceso. Un modelo en Redes de Petri permite conocer las acciones que se deben ejecutar para cumplimentar el encendido secuencial de los equipos del proceso a través de redes industriales, de manera que estos puedan ser controlables por lugares auxiliares que representan los sensores del proceso y los eventos del operador que son transmitidos por medio de comunicaciones industriales.

También resulta de suma importancia el desarrollo de los modelos que describen el intercambio de datos entre los niveles del SAI, pues estos patrones posibilitan caracterizar en detalle el flujo de la información en el sistema, permitiendo definir los lugares que provocan fallos de comunicación con el objetivo de clasificarlos para sugerirle al operador una posible solución según la previa tipificación del fallo.

Construir modelos para describir la manera correcta de controlar los actuadores del proceso constituye una gran ventaja, pues a través de estos esquemas se tiene en cuenta los estados que activan o desactivan a motores y electroválvulas, posibilitando reconocer el estado de funcionamiento de los actuadores de la planta a través del lugar de confirmación empleado para la visualización del estado de los equipos en el Supervisorio soportado en modernas redes industriales.

Modelar el control de los actuadores de la planta en Redes de Petri, permite tener en cuenta los estados que provocan la falla de algunos de estos elementos, como son el sobreconsumo del motor, la ausencia de un esclavo en la red, o la detección del fallo de recorrido después de efectuar un comando de apertura o cierre de una racera. Estos modelos permiten clasificar los posibles fallos del sistema constituyendo esto una gran ventaja a la hora de concebir la atención de alarmas y el tratamiento de fallos en el Sistema Supervisorio.

Con esta herramienta de modelado se puede diseñar fácilmente un correcto algoritmo de control que satisfaga los requerimientos funcionales del proceso, teniendo en cuenta el control del proceso en modo manual y automático, con el objetivo principal de diseñar subrutinas (submodelos) que tengan en cuenta los siguientes conceptos:

- a. Control de la Automática local incorporando el Mantenimiento preventivo.
- b. Chequeo del Flujo de datos del sistema para detectar Fallos de Comunicación.
- c. Dinámica de funcionamiento del Supervisorio con Atención de Alarmas y Tratamiento a Fallos.

2.2 Modelo que describe el Flujo de datos en el SAI.

El modelo presentado a continuación muestra de forma dinámica el flujo de datos en el SAI propuesto, y tiene como objetivo analizar los distintos protocolos de comunicación por donde transitan los datos, partiendo desde el nivel más bajo de la pirámide (nivel de campo) donde se obtienen los datos, pasando por el nivel intermedio (nivel de control) donde se procesan dichos datos a través de un algoritmo de control hasta llegar hasta el nivel más alto (nivel de supervisión) en el cual se muestran en forma de valores y mediciones del proceso. Este modelo está concebido para identificar los posibles fallos de comunicación en el sistema teniendo en cuenta su previa clasificación.

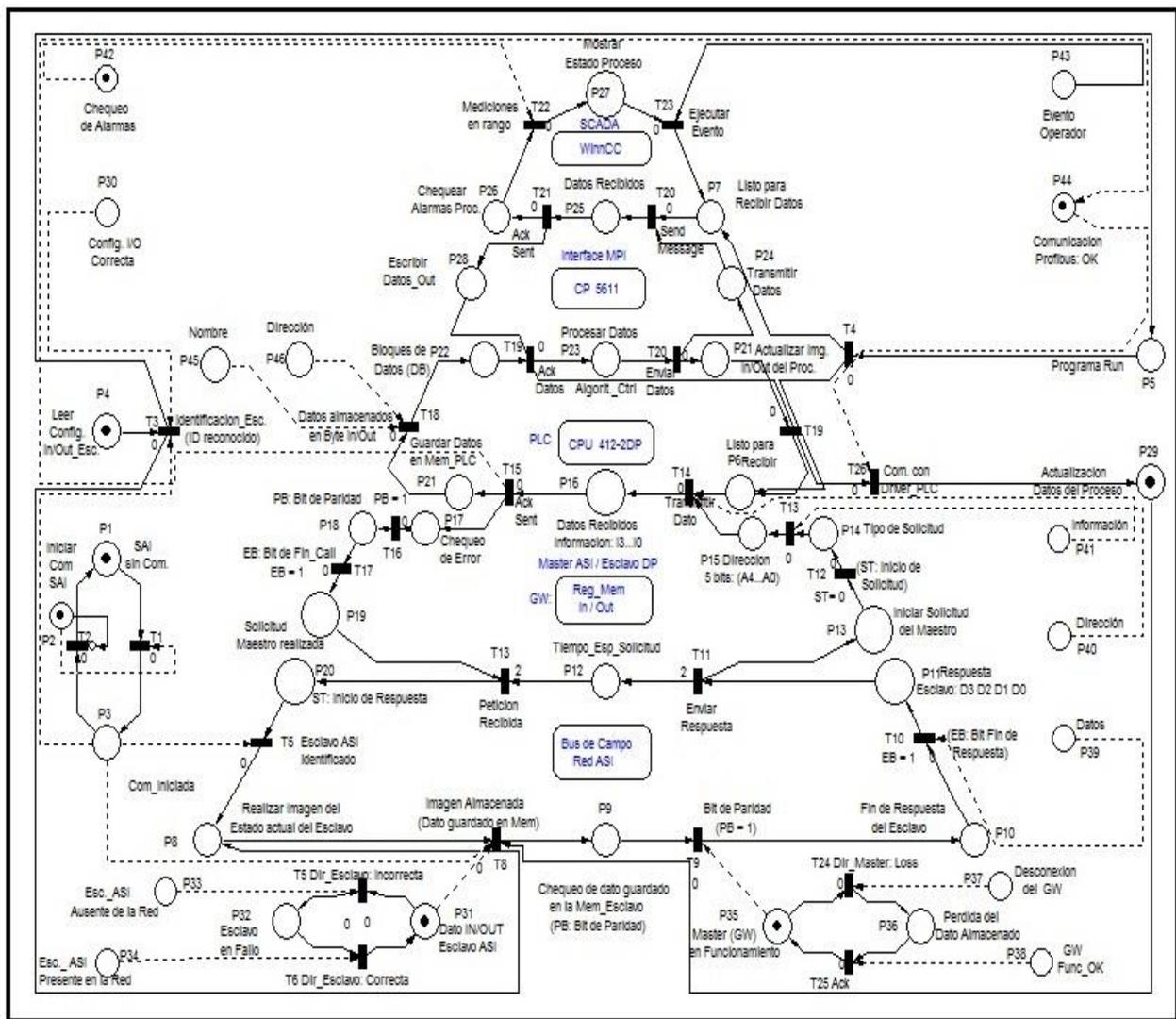


Fig. 2.1 Modelo que describe el flujo de datos del SAI.

2.2.1 Explicación del Modelo:

Este modelo permite describir el proceso de transmisión de datos efectuado en el sistema de automatización propuesto, definiendo un submodelo de **Inicio** para iniciar la comunicación entre los niveles de campo, de control y de supervisión. En el mismo se emplean subredes diseñadas para modelar el principio de funcionamiento de cada una de estas etapas por las cuales transita el dato, teniendo como principal objetivo la detección de fallos de comunicación en el sistema.

El inicio del flujo de datos se lleva a cabo a través de la activación del lugar auxiliar P1, el cual permite dar comienzo a la comunicación entre el nivel de campo y el nivel de control, mediante una subred que describe el principio de funcionamiento del bus de campo AS-i donde se lleva a cabo la realización de una imagen de los datos de campo. El objetivo de esta acción inicial es reconocer la configuración correcta del esclavo AS-i para poder efectuar en el siguiente ciclo la transacción con el maestro de la red y se

ejecuta siempre y cuando la transición T3 sea habilitada por el lugar auxiliar P4, representando la configuración entrada/salida correcta del esclavo en la red. El submodelo inferior describe el modo en que se efectúa la lectura de datos en un esclavo AS-i, cuando este le responde al Maestro, a través de la activación secuencial de los lugares que definen los momentos de una **Respuesta** en una Transacción AS-i. Esta subred define cada acción que debe llevar a cabo un esclavo para guardar la información en una memoria no volátil, empleando para ello cuatro bits de datos (D3 D2 D1 D0) para almacenar el estado actual de sus entradas y salidas.

La comunicación entre el Maestro y los Esclavos es modelada a través de la red que contiene los lugares que representan las etapas de una Transacción AS-i, mediante la subred que describe la Solicitud del Maestro y la Respuesta del Esclavo. Este proceso se lleva a cabo mediante la activación secuencial de los lugares que definen la **Solicitud** del Maestro (desde P13 hasta P19), teniendo en cuenta los lugares auxiliares P40 y P41 (**Información** y **Dirección**), que simbolizan la dirección del esclavo que se desea encuestar con la información del comando solicitado por el Maestro.

Una vez que el esclavo reconoce el comando solicitado, representado por el lugar P20, debe responder con la información que contiene el estado actual del proceso. Esta etapa está modelada por la activación secuencial de los lugares P8 hasta P12, y representa el almacenamiento de una imagen del proceso en 4 bits de datos.

Después que la información del esclavo está guardada en la memoria de la pasarela se lleva a cabo el almacenamiento de esta información en la memoria del PLC. Este proceso es modelado teniendo en cuenta el lugar auxiliar P21, habilitando a T18, representando el funcionamiento correcto de la Red Profibus, permitiendo guardar en la memoria del PLC en forma de byte el estado actual de los datos del proceso.

Una vez que los datos estén almacenados, se lleva a cabo su procesamiento por parte de un algoritmo de control implementado en el PLC. El procesamiento de esta información se modela a través de una subred que simboliza el nivel de control, simbolizada por los lugares P22 (Mem._CPU) y P23 (Algoritmo de Control), que permiten simular el direccionamiento en forma de byte en la memoria del programa con un nombre (P45) y dirección (P46) específica y luego se procesan a través del código de programación con el objetivo de satisfacer los requerimientos funcionales de la planta y visualizar estado real del proceso en el nivel superior.

Finalmente, el dato procesado por el PLC se transmite al nivel de supervisión donde se muestra en forma de medición o variable del proceso. Esta etapa es modelada mediante una pequeña subred ubicada en el nivel superior de la pirámide, diseñada para representar la visualización de la información real de la planta en la aplicación SCADA a través del lugar P27, (Mostrar Estado del Proceso). Esta subred muestra la posibilidad que tiene el operador de efectuar cambios en el funcionamiento de la planta, representado por la activación del lugar P43 (Evento_Operador) que significa la escritura de un nuevo valor en la dirección del dato transmitida en el nivel de control.

2.2.2 Importancia del Modelo:

El objetivo principal de este modelo es definir una correcta arquitectura del SAI, eligiendo una adecuada interface de comunicación para cada nivel que garantice la transmisión de la información desde el nivel de campo hasta el nivel de supervisión. Aquí se definen los lugares que posibilitan la transmisión de la información, teniendo en cuenta los procesos de obtención, almacenamiento, procesamiento y visualización de datos que se lleva a cabo en el sistema. Este modelo relaciona los diferentes protocolos para la transmisión de datos reconociendo el medio físico y el nivel de enlace de cada red, especificando el empleo de un bus de campo para obtener la información del proceso a través de una red AS-i, el empleo del protocolo Profibus DP para la comunicación con el nivel de control y una interfaz de comunicación serie mediante una red MPI para transmitir el dato hacia el nivel de supervisión.

Este modelo está diseñado para identificar los lugares que pueden provocar fallos de comunicación, permitiendo su clasificación para poder detectarlas en el momento que sucedan y de este modo sugerir una posible solución. A continuación se presenta la arquitectura que define la propuesta de automatización del sistema, que constituye el resultado de la interpretación correcta del modelo mostrado anteriormente.

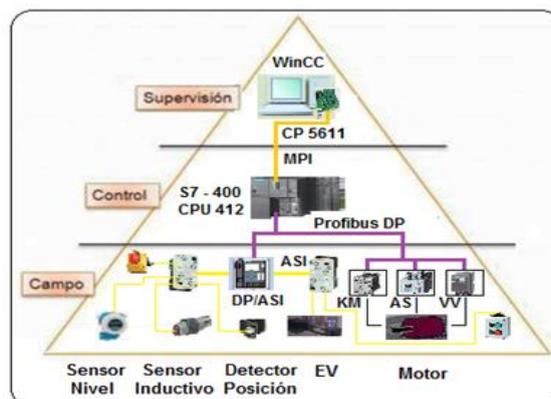


Fig. 2.2 Arquitectura de la Propuesta de Automatización.

Esta arquitectura constituye una adecuada interpretación del modelo en Redes de Petri que describe el flujo de datos en el SAI, definiendo una red AS-i para sensores y actuadores en el nivel de campo, permitiendo actuar de forma local o remota sobre motores y electroválvulas. Detalla además un PLC de la familia Siemens (S7- 400), en el nivel de control con una red de comunicación Profibus DP que tiene en cuenta el uso de una pasarela DP/AS-i, como uno de sus esclavos Profibus para guardar los datos del proceso. Para el nivel de supervisión se propone el empleo de las herramientas de WinCC para el desarrollo de la aplicación SCADA montada en una PC, que mediante una tarjeta de comunicación (CP 5611) se comunica con la red MPI configurada en el puerto MPI de la CPU, permitiendo el intercambio de datos con el nivel de supervisión.

La interpretación de esta arquitectura se consolida en la solución mostrada en la Fig. 2.16 al final de este capítulo. Esta propuesta está basada en la arquitectura presentada anteriormente y responde a los estándares de automatización industrial empleados por países del primer mundo en tareas de modernización y control. Por ello se decide emplear tecnología Siemens basada en redes industriales para llevar a cabo las tareas de sensar, controlar y supervisar el funcionamiento de la planta.

2.2.2.1 Explicación de la Arquitectura de la Propuesta de Automatización.

Nivel de Campo: En esta etapa se definen los sensores de Nivel para la medición del estado máximo y mínimo de granos en los silos, eligiendo el transmisor MiniCap, caracterizado en el Anexo 1, el sensor inductivo para la confirmación del movimiento de la cadena del Elevador (SI) mostrado en el Anexo 2, y los detectores electromagnéticos de posición para confirmar el estado abierto o cerrado de las raceras, los cuales van acoplados en el extremo del cuerpo del pistón como se muestra en el Anexo 3.

Las electroválvulas de simple efecto son los actuadores escogidos para controlar el desplazamiento de las raceras, como se aprecia en el Anexo 4, empleadas para abrir y cerrar las compuertas de los silos. El control de los motores de menor potencia se realiza a través de contactores magnéticos (KM) SimoCode mostrado en el Anexo 5, los cuales incorporan una bobina de control para el arranque/paro de las máquinas contempladas con arranque directo, como son los Tornillos Sin-Fin y los Elevadores de cangilones. Para los motores de mayor potencia se emplean Arrancadores Suaves (AS), (ver Anexo 6), permitiendo el arranque progresivo de los Molinos y de esta forma suavizar el proceso de arranque definido por una rampa de aceleración. Este arrancador suave es necesario ya que permite brindar al PLC la señal de fin de la

rampa, cuyo objetivo es informarle al sistema que el molino está listo para trabajar en su régimen nominal de operación. Para los motores dosificadores (Dosif.) que alimentan a los molinos se utilizan variadores de velocidad (VV), permitiendo poder variar su velocidad de trabajo, (ver Anexo 7). Este variador es de suma importancia ya que permite regular la entrada de materia prima al molino a través de un lazo de control de velocidad teniendo como referencia la corriente del molino.

En este nivel se define un bus de campo AS-i que recorre toda la planta y mediante estaciones esclavas se satisfacen las entradas y salidas digitales del proceso. El Maestro AS-i es quien almacena una imagen del estado real del proceso y la guarda en la memoria del PLC situado en el nivel de control.

Nivel de Control: comprende los elementos que permiten controlar el funcionamiento del proceso. Para este nivel se propone una CPU 412-2DP, de la familia Siemens, la cual trabaja con una red Profibus DP para el intercambio de datos con cada uno de sus esclavos DP (Periferia Descentralizada), que no son más que los Máster AS-i de cada área y los SimoCode. Esta red se origina en el PLC ubicado en la sala de control y recorre todas las áreas de la planta y se emplea para recibir el estado de los parámetros de funcionamiento de los motores y para transmitir los mandos de arranque/paro de los motores y apertura/cierre de las electroválvulas, mediante los comandos enviados al Máster AS-i.

Para llevar a cabo la comunicación entre la aplicación SCADA y el PLC es necesario configurar una red MPI para la transmisión de datos desde el autómatas hacia el supervisor, para de esta manera establecer un enlace con la interfaz de comunicación utilizada en la aplicación HMI. A continuación, se muestra la configuración del PLC seleccionado para llevar a cabo esta propuesta de automatización.

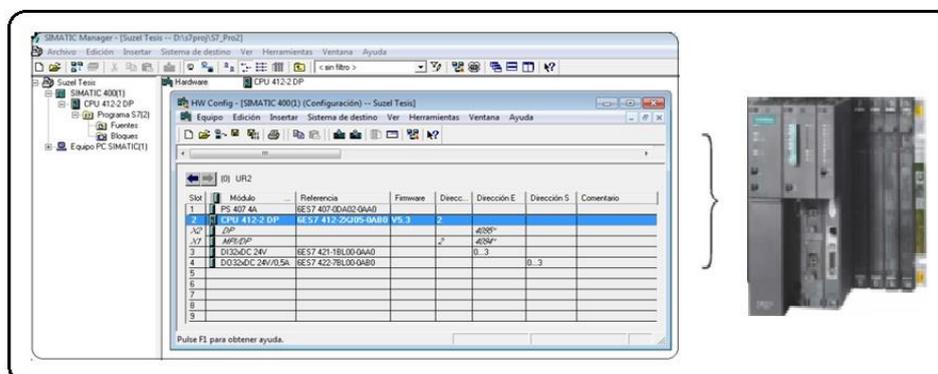


Fig.2.3: Configuración de Hardware del PLC SIMATIC S7-400.

Características principales del PLC seleccionado ^[21]:

- PLC de alto rendimiento para tareas de gamas media y alta con posibilidad de ampliación en caso de que aumenten las tareas.
- Aplicación flexible gracias a la fácil implementación de estructuras descentralizadas (distribuidas) con un cómodo sistema de conexión.
- Posibilidades óptimas de comunicación, supervisión y conexión en red.

Nivel de Supervisión: En este nivel se propone el uso de WinCC como software para la implementación de la aplicación SCADA. El objetivo principal de este Supervisor es proporcionar una pantalla principal para controlar el funcionamiento del proceso con un nivel de control limitado, brindando la posibilidad de navegar hacia una ventana para la atención de Alarmas y otra para graficar el comportamiento continuo en forma de Tendencias de las variables analógicas del sistema.

En este nivel se gestionan los datos mediante un Procesador de Comunicación (CP 5611), cuya función es establecer la comunicación entre la PC aplicación y la CPU, constituyendo el interfaz de comunicación entre el autómatas y la aplicación HMI, conectada a una red de comunicación multipunto (MPI) configurada en el NetPro, como se muestra en el Anexo 8. Esta red es la interface de comunicación más sencilla que existe entre los equipos S7, ya que se puede transferir o acceder a cualquier zona del mapa de memoria del PLC con el que se desea comunicar y sólo es necesario indicar dónde se quiere que se dejen los datos leídos o cuales es la fuente de datos a enviar. La tarjeta CP 5611 A2 está equipada con una interface Profibus hasta 12 Mbps y está orientada para la operación en PGs y PCs a través de una interface PCI estándar, permitiendo conectar hasta 32 dispositivos en el bus. El acoplamiento físico de la interface MPI/DP es sobre una interface flotante RS-485 que forma parte de la red.

Sistema de Supervisión WinCC.

Para esta aplicación se decide emplear WinCC Flexible de Siemens como software para desarrollar la aplicación SCADA por ofrecer librerías de funciones para lenguajes de uso general que permiten personalizar de manera muy amplia la aplicación que desee realizarse y por integrar los siguientes módulos para las actividades de adquisición, supervisión y control que se llevan a cabo mediante un paquete de funciones que incluye zonas de programación en un lenguaje de uso general ^[27].

Módulos que ofrece WinCC:

Configuración: permite al usuario definir el entorno de trabajo de su SCADA, adaptándolo a la aplicación particular que se desea desarrollar.

Interfaz gráfico del operador: proporciona al operador las funciones de control y supervisión. El proceso se representa mediante sinópticos gráficos generados desde el editor incorporado en el SCADA o importados desde otra aplicación.

Módulo de proceso: ejecuta las acciones de mando pre-programadas a partir de los valores actuales de variables leídas.

Gestión y archivo de datos: se encarga del almacenamiento y procesado ordenado de los datos, de forma que otra aplicación o dispositivo pueda tener acceso a ellos.

Comunicaciones: se encarga de la transferencia de información entre la planta y la arquitectura, hardware que soporta el SCADA, el resto de elementos informáticos.

2.3 Modelado en Redes de Petri de cada nivel del SAI propuesto.

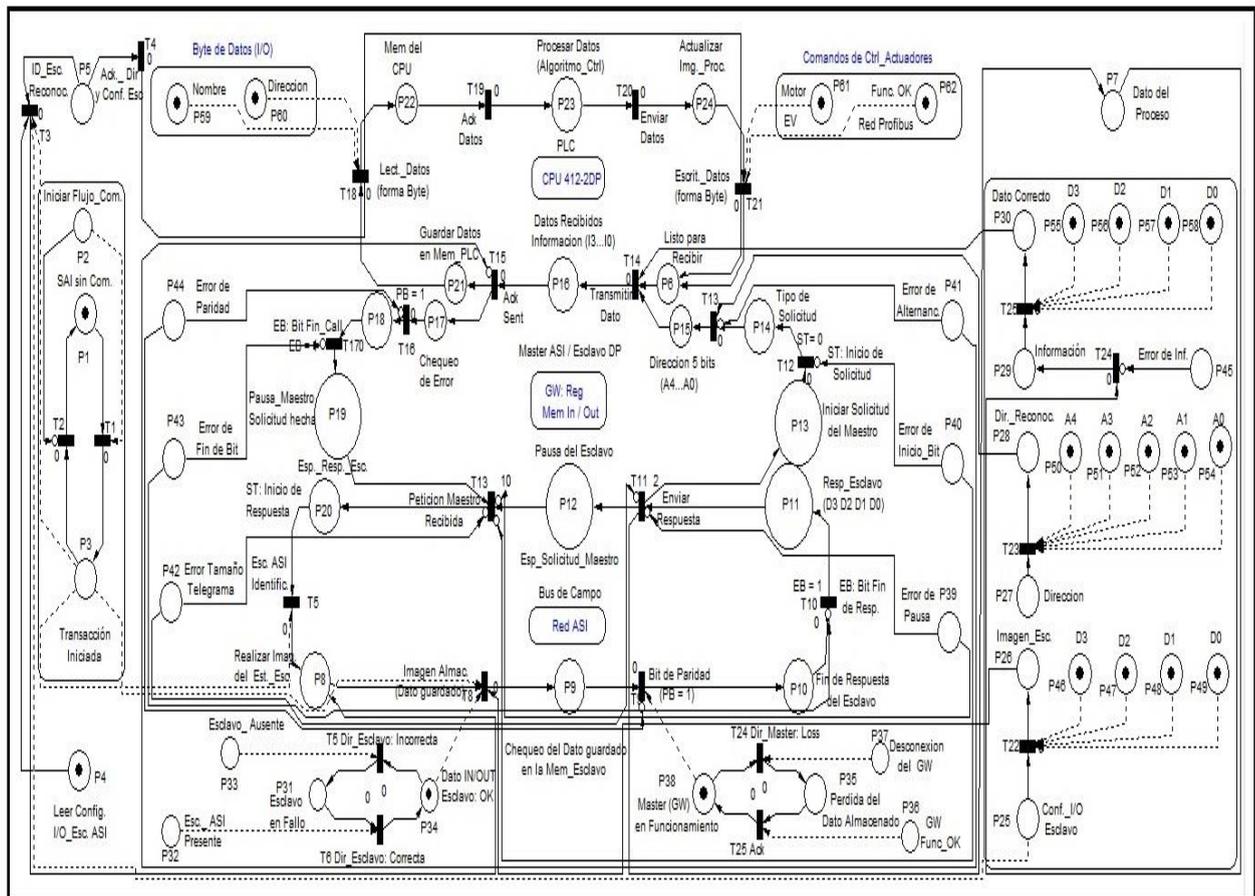
En los siguientes epígrafes se realizará el modelado en Redes de Petri de cada nivel del SAI propuesto, teniendo como principal objetivo definir los estados y condiciones necesarias para la operación del sistema, describiendo el principio de trabajo de cada subred que conforma la solución. El modelado de cada etapa constituye la herramienta teórica empleada como método de aprobación de la arquitectura propuesta, permitiendo definir la dinámica de funcionamiento de cada nivel del SAI presentado.

2.3.1 Principio de Funcionamiento de la Red ASI.

Este modelo explica el principio de funcionamiento de la comunicación en la red AS-i, teniendo en cuenta los momentos de una transacción llevada a cabo entre el Maestro y cada uno de sus Esclavos. Esta red es desarrollada teniendo como base el modelo simplificado de un protocolo de comunicación definido en la Fig.2.4 [3].

2.3.1.1 Explicación del Modelo:

El propósito fundamental de esta red es modelar el funcionamiento de la comunicación AS-i, garantizando la activación de manera secuencial de los 4 lugares que intervienen en la conversación entre el Maestro y sus esclavos. Este modelo se construye para conocer las condiciones que permiten la activación de los 4 momentos de la Transacción AS-i representados en la Fig. 2.4 y explicado a través de los siguientes Macro lugares.



III. Macro lugar 3: (Respuesta del Esclavo) lugar que representa la imagen actual del esclavo almacenada en 4 bit de información. La transición T22 es habilitada por el estado activo de los 4 lugares auxiliares (P46 P47 P48 P49) indicando el almacenamiento del estado de las señales de campo en la memoria del esclavo (D3 D2 D1 D0), datos que luego van a ser leídos por el Maestro una vez que se guardan en los lugares P55, P56, P57, P58 cumpliendo con el proceso de respuesta. Esta acción significa que el esclavo ha respondido con el dato solicitado por el maestro siempre y cuando no esté activo el lugar P45 que simboliza Error de Información.

IV. Macro lugar: (Pausa del esclavo) tiempo de espera por el esclavo representado por el lugar P12 que simula una pausa entre 1.5 y 2 tiempo de bit a través de la transición temporizada T11 definida para esperar la Solicitud del Maestro. Este lugar estará habilitado siempre y cuando no esté activo el lugar auxiliar P39 (Error de Pausa).

2.3.1.2 Importancia del Modelo.

Uno de los objetivos importante de este modelo es definir cada uno de los lugares que conforman la trama de un dato ASI, con el propósito de conocer en qué lugar del telegrama se guarda la información recogida del esclavo que luego se almacena en la memoria de la CPU. Esta red permite conocer la dirección específica en el PLC donde se almacena el estado de las señales de campo guardadas con un nombre y dirección única como se muestra en la siguiente tabla, que reserva los bits 4, 5, 6 y 7 del primer byte para indicar el estado de la pasarela y de sus esclavos.

I/O byte	Bit 7-4				Bit 3-0			
n+0	reserved				Slave 1			
					Bit 3	Bit 2	Bit 1	Bit 0
n+1	Slave 2				Slave 3			
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-								
-								
n+15	Slave 30				Slave 31			
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

Tabla 2.1: Datos AS-i almacenados en la memoria del PLC en forma de byte de entrada/salida.

Resulta de gran importancia también la detección de los errores relacionados con la información y con la longitud del telegrama, teniendo como objetivo conocer si la transacción entre el Maestro y un Esclavo se ha efectuado correctamente. Pues al conocer cada uno de los lugares que conforman un dato AS-i se puede detectar los estados que pueden ocasionar fallos en cada transacción. En este modelo se define claramente los 7 lugares (P39, P40, P41, P42, P43, P44, P45), que provocan error en la comunicación, permitiendo de esta manera poder detectarlos una vez que ocurran a

través de la apropiada programación de la detección de errores en el menú de fallos de la pasarela. El objetivo de esta etapa es repetir la transacción cuando se detecta alguno de estos eventos que considera fallida la comunicación e invalida la trama recibida.

2.3.2 Control de los actuadores de la planta a través del bus ASI. Control de Motores y Electroválvulas.

Este modelo está compuesto por varias subredes que representan las etapas por las cuales transita el dato cuando se manda a activar los actuadores del proceso a través de los comandos de control. Su objetivo es representar la adecuada estructuración de los elementos en el bus AS-i para enviar correctamente la señal al PLC a través de la red Profibus, direccionando apropiadamente el dato en la memoria de la pasarela, quien luego escribe la información en la memoria del esclavo permitiendo actuar rápidamente sobre motores y electroválvulas.

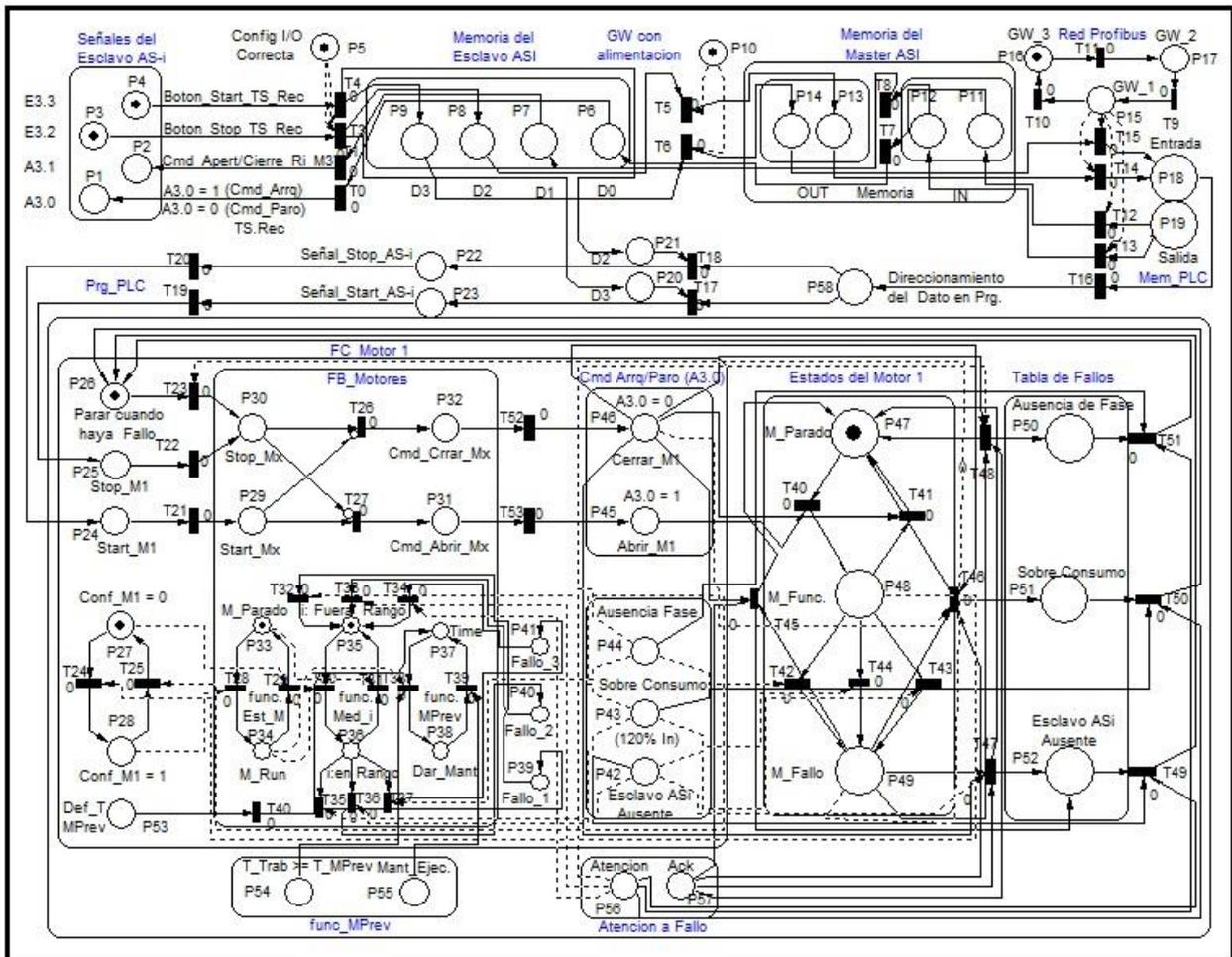


Fig. 2.5 Modelo para el control de motores.

A continuación, se explican las subredes que integran este Modelo.

2.3.2.1 Explicación del Modelo:

Esclavo AS-i: Subred que representa una estación esclava de la red AS-i, definida por su configuración entrada/salida a través de los lugares P1, P2, P3 y P4. El módulo esclavo que modela esta red está configurado para conectar dos entradas digitales en los lugares P3 y P4 y dos salidas en P1 y P2. Los lugares P3 y P4 se emplean para conectar las señales provenientes de los sensores de nivel máximo y mínimo o los botones de Arranque y Paro de los motores. P1 y P2 constituyen los lugares donde se escriben los comandos de arranque/paro de motores y apertura/cierre de las compuertas permitiendo actuar sobre la bobina de control de los actuadores. En esta red se utilizan los lugares P6, P7, P8 y P9 para representar la memoria (no volátil) del esclavo que almacena su configuración de trabajo y la imagen actual del proceso.

Master AS-i: Subred que representa la memoria de la pasarela que gestiona la transacción AS-i y permite el almacenamiento del estado real de todos los esclavos de la red en registros de memoria de entrada y salida a través de los lugares P11, P12, P13 y P14. Este submodelo almacena los datos de la respuesta de los esclavos ante la solicitud del maestro y los comandos de control para activar las salidas de los esclavos.

Red Profibus DP: Subred definida por los lugares P15, P16 y P17, que representan el principio de funcionamiento cíclico que lleva a cabo la red Profibus en su proceso de encuesta a los esclavos DP. Este submodelo demuestra que esta comunicación es posible gracias al comportamiento limitado y repetitivo del modelo que define la comunicación en la red Profibus, la cual permite la lectura y escritura de datos tanto en la memoria de la pasarela AS-i como en la memoria del programa. Esta red representa el almacenamiento de los datos presente en la memoria de entrada-salida de la pasarela DP/AS-i en la memoria del PLC en forma de byte I/O.

Programa del PLC: Este submodelo direcciona los datos guardados en la memoria del PLC con un nombre y una dirección específica, teniendo en cuenta si son direcciones de entrada o de salida, las cuales van a ser procesadas por el algoritmo de control para llevar a cabo la secuencia del proceso a través del programa. Esta red permite convertir la acción de Arrancar (Start: P3) y Parar (Stop: P4) ejecutadas en el nivel de campo, en señales de entradas digitales representado por los lugares P24 y P25, los cuales le indican al programa la necesidad de efectuar los comandos de arranque o paro de un motor según corresponda. Esta red está compuesta por la subred Bloque FC del Motor explicada a continuación.

Bloque FC_Motor: El objetivo esencial de este modelo es representar el principio de programación estructurada con reusabilidad de bloques y encapsulamiento de datos, a través de la definición en el programa de una Función de Control (FC) donde se realiza la llamada a un bloque funcional (FB) para darle atención a cada motor. Esta FC contiene los lugares que definen las señales de entrada digital Start (P24) y Stop (P25), así como la confirmación de trabajo del motor (P28) y como salidas los comandos de Arranque y Paro del motor representado por P31 y P32. También permite modelar la respuesta del programa cuando el sistema reconoce la ocurrencia de un fallo.

FB_Motores: Esta subred tiene en cuenta la llamada de un FB dentro de un FC, corroborando el concepto de reutilización de bloques en el programa, con el objetivo de controlar el funcionamiento de cada motor de la planta, a través de la programación de los comandos de arranque y paro. Este bloque contiene la función de medición de corriente definida por el lugar P36 y se activa cuando el equipo comienza a trabajar, activando además a la función que calcula el tiempo trabajado por el motor para dar la señal de Mantenimiento Preventivo (P38) una vez que el tiempo trabajado por el motor sea mayor o igual que el tiempo definido por usuario para dar mantenimiento (P53) y además procesa la medición de corriente para obtener las señales que signifiquen fallos, identificados por los lugares P42, P43 y P44.

2.3.2.2 Importancia del Modelo:

Este modelo demuestra que mediante la programación estructurada y la correcta selección de los elementos que conforman el bus, mostrados en la Fig. 2.6, se controla adecuadamente cada equipo de la planta mediante sus comandos de control.

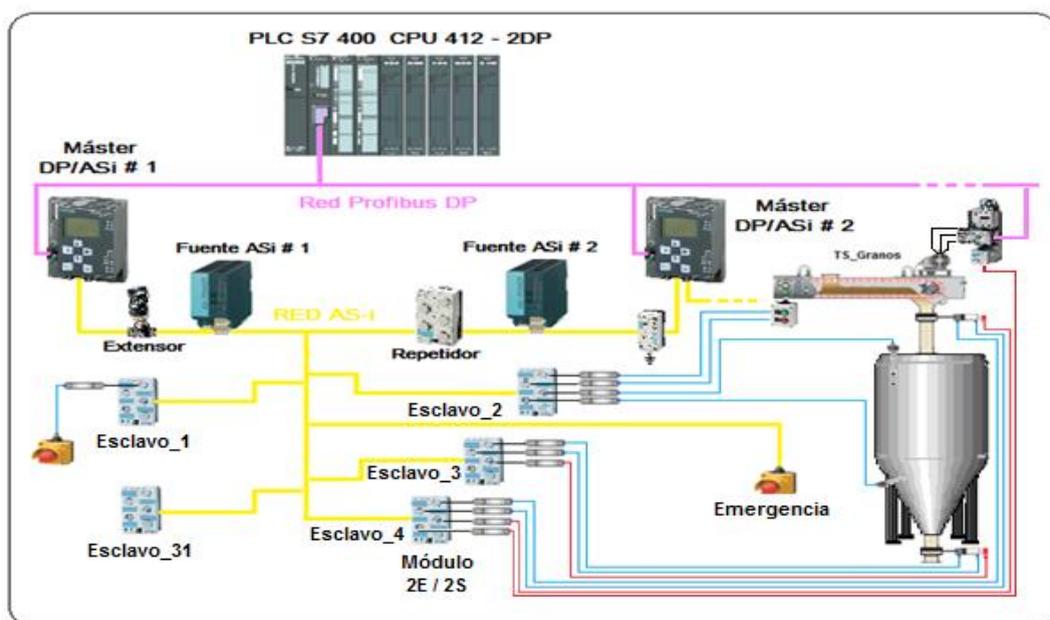


Fig. 2.6. Arquitectura de comunicación para el control de los sensores y actuadores.

La concepción de implementar esta programación permite la reusabilidad de bloques en el programa, la organización del código necesario para concebir las tareas de mantenimiento y la definición de jerarquía entre bloques y funciones. Además posibilita implementar funciones para controlar el proceso, con el objetivo de mostrar en cada momento los estados de Parado, Funcionando o Falla de los motores y Abierto, Cerrado o Error de Posición para las Raceras.

2.4 Automática Local del Proceso.

El modo de operación de la planta se debe concebir de forma Manual o Automática. Por ello es necesario caracterizar cada uno de estos modos a través de modelos en Redes de Petri que posibilitan definir claramente las condiciones de control de los comandos de arranque/paro para los motores y apertura/cierre para las raceras, ya sea de forma automática, garantizando el encendido inverso secuencial de la ruta de trabajo seleccionada o de forma manual a través de WinCC (Modo Remoto) o desde las botoneras de campo (Modo Local).

Este principio de modelación permite programar la secuencia de trabajo del proceso a través de dos unidades funcionales (UF), Automático y Manual, incorporando la programación de los conceptos de los Interbloques Tecnológicos (Interlock de Arranque, de Proceso y de Paro) para cada componente cuando el sistema esté en Automático y la selección de los modo de operación Local y Remoto de los equipos cuando esté habilitado el modo de trabajo Manual.

2.4.1 Requerimientos Funcionales del proceso (Etapas de Recepción).

1. Concebir la selección de los modos de trabajo Manual o Automático a través de un botón tipo interruptor en el Supervisorio.
2. Función de selección de Ruta que permita elegir Origen y Destino.
3. Contemplar el Arranque y Paro del proceso a través de un botón en el supervisorio que permita iniciar o detener la secuencia de trabajo.
4. Definir como condición de seguridad para comenzar la Recepción de materia prima, el estado cerrado de la Racera de salida del silo de destino seleccionado ($Ro_Sx = 0$) siempre y cuando este no esté lleno ($Nmax_Sx = 0$).
5. En modo Automático concebir el Arranque de los equipos en sentido Inverso Secuencial (AIS), encendiendo los aparatos correspondientes a la ruta seleccionada de atrás hacia adelante, teniendo en cuenta el tiempo de espera de fallo del equipo previamente arrancado (Tf) y la señal que indica confirmación de marcha ($Conf_Run$).

6. En modo Automático concebir el Paro del proceso en sentido Directo y de forma Secuencial (PDS) al activarse la condición de paro, (nivel máximo del silo de destino seleccionado: $N_{max_Sx} = 1$), permitiendo evacuar el material no recepcionado aún, teniendo en cuenta el tiempo de evacuación (T_{Evac}).
7. En modo Automático efectuar un Paro Rápido del proceso ante la ocurrencia de un Fallo relacionado con los equipos pertenecientes a la ruta de trabajo actual.
8. Contemplar la parada de Emergencia a través de un botón de Paro Rápido, cuya función sea detener inmediatamente el proceso.
9. En modo Manual cada equipo puede trabajar de forma Local mediante una botonera de campo o de forma Remota desde WinCC.
10. Incorporar en la función de control de cada componente la programación de los conceptos de Interbloques Tecnológicos: **Interlock de Arranque (IL_Arranque)** e **Interlock de Proceso (IL_Proceso)** ^[22] como condición de arranque propia de cada dispositivo en modo Automático, considerando el momento en el proceso en que le corresponde trabajar.

Definiciones de los Interbloques Tecnológicos: condiciones del proceso que se tienen en cuenta para el control de cada componente y son denominados Interlocks (IL) ^[22]. Son señales que se calculan con el objetivo de determinar el momento exacto para el arranque y el paro de los equipos.

En esta aplicación se tienen en cuenta estas condiciones para la programación de la función de control de cada componente en modo de trabajo Automático.

IL_Arranque: condiciones necesarias para el arranque que garantiza el encendido del equipo si este está **Listo** para trabajar. Se calculan con el objetivo de evitar el encendido transitorio de cada mecanismo.

IL_Proceso: condición obligatoria que debe garantizar el proceso para permitir que el equipo trabaje continuamente. Este lugar reconoce que deben cumplirse los estados definidos por el proceso para que el equipo trabaje de forma permanente. Tiene en cuenta el estado de buen funcionamiento del equipo arrancado previamente.

IL_Paro: Condición del proceso que provoca un paro no inmediato del equipo, debido a la ausencia de algún estado que impide el funcionamiento correcto de la secuencia de trabajo. Constituye una condición definida por el proceso que imposibilita que se continúe con el flujo productivo, reconociendo el Paro en sentido directo secuencial (PDS) de los equipos teniendo en cuenta el tiempo de evacuación del material (T-Evac).

2.4.2 Modelo para la Selección del Modo de Operación Manual o Automático.

El siguiente modelo explica cómo se realiza la selección del modo de trabajo del proceso incorporando una subred programada para la efectuar un paro rápido ante un comando de emergencia. El objetivo fundamental de este modelo es programar modularmente la selección de los modos de trabajo de la planta mediante dos UF (Manual y Automático), donde en cada una de ellas sea necesario satisfacer los requerimientos funcionales de la planta.

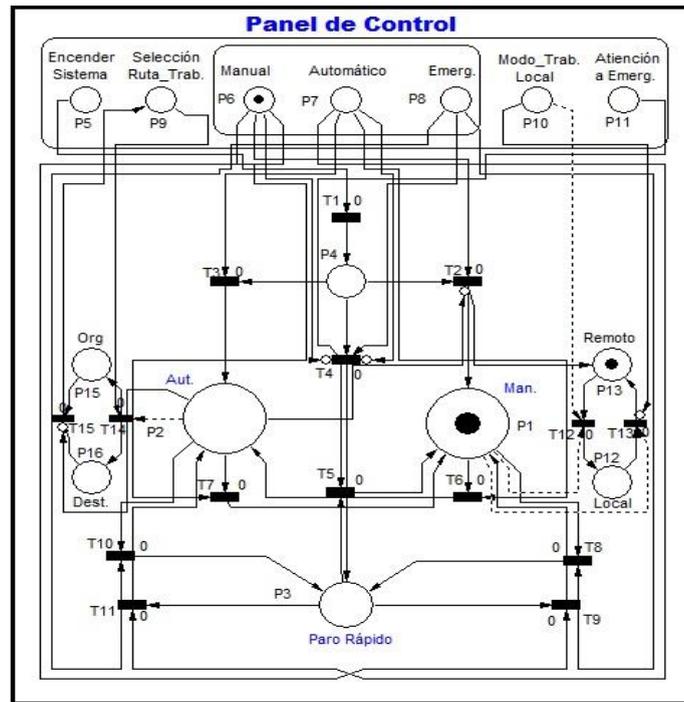


Fig.2.7 Modelo para selección del modo de operación del proceso empleando UF.

2.4.2.1 Explicación del Modelo:

La selección del modo de trabajo debe llevarse a cabo por el operador a través de un selector desde el Supervisorio representado por los lugares P6 y P7 después de estar encendido el sistema (P5). La activación de uno de estos lugares auxiliares habilita los lugares Manual (P1) o Automático (P2). El sistema ha sido concebido con la inicialización en modo manual por defecto, con el objetivo de permitir operar la planta de forma manual como condición de seguridad. Nunca se podrá activar más de un modo de trabajo a la vez y cuando se efectúa un comando de Emergencia (P8), se lleva a cabo la parada inmediata del proceso a través de la activación del lugar P3.

En modo automático el sistema exige la selección de una ruta de trabajo a través de la activación del lugar P9 que simula la selección del origen y destino de Recepción (P15 y P16). En modo manual el sistema puede ser operado en modo Local cuando esté activo el lugar P10 o en modo Remoto cuando no lo esté.

2.4.2.2 Importancia del Modelo:

El objetivo fundamental de este modelo es permitir programar estructuralmente la selección del modo de operación mediante dos UF (Manual y Automático) que corresponden a los lugares P1 y P2 del modelo, donde en cada una de ellas sea necesario satisfacer los requerimientos funcionales de la planta. En modo Automático el sistema garantiza el cumplimiento de la secuencia de trabajo del proceso mediante el algoritmo de control, mientras que en modo Manual solo se realizan pruebas y acciones de mantenimiento a los equipos de la planta.

2.4.3 Modelo que define el funcionamiento de la Automática Local del Proceso.

En el siguiente modelo se define el control de la secuencia de trabajo del proceso especificando las condiciones necesarias para el encendido de cada equipo mediante una lógica de programación secuencial, con el objetivo de concebir un funcionamiento controlado de los elementos pertenecientes a la ruta de trabajo seleccionada.

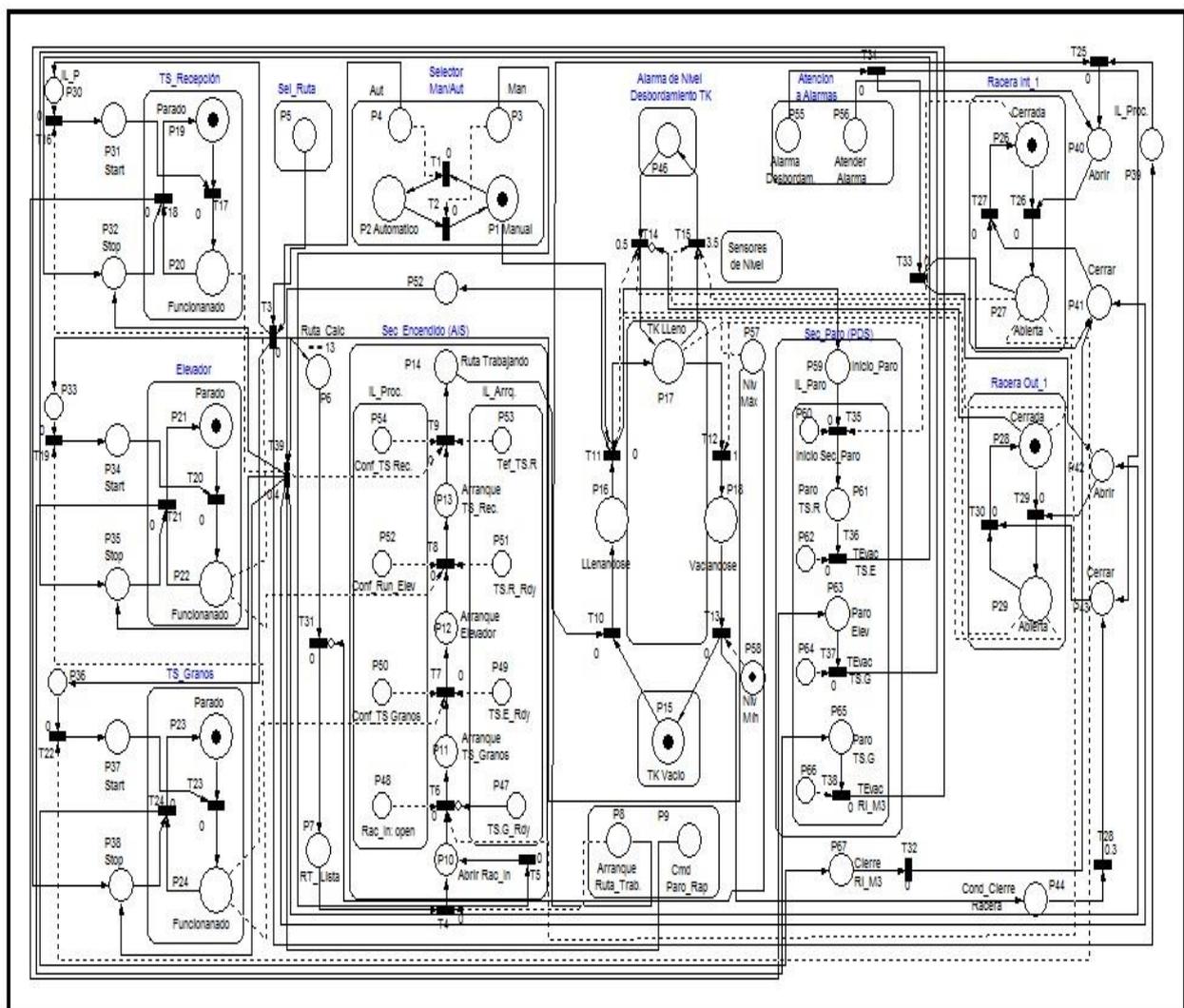


Fig. 2.8 Modelo para la Automática Local del proceso

2.4.3.1 Explicación del Modelo:

En modo Manual el control de los motores y raceras se realiza mediante la activación de los lugares auxiliares Start (P31, P34, P37) y Stop (P32, P35, P38) para los transportadores de materia prima y, Abrir (P40, P42) y Cerrar (P41, P43) para el control de la apertura y cierre de las compuertas de los silos.

En modo Automático el sistema está diseñado para cumplimentar el arranque inverso secuencial (AIS) de los equipos que conforman la etapa de Recepción, cumpliendo con la activación secuencial de los lugares P10, P11, P12 y P13 que activan el lugar P14 simbolizando que la ruta seleccionada está trabajando. Para llevar a cabo la programación del encendido de esta ruta es necesario tener en cuenta los lugares que representan los Interlocks de Arranque y de Proceso para cada equipo, identificados claramente en el modelo a través de los lugares P47, P49, P51y P53 (IL_Arrq) y P48, P50, P52y P54 (IL_Proc). Como Interlock de Arranque se tiene en cuenta la señal que indica que el equipo está preparado para trabajar y como Interlock de Proceso la confirmación de marcha del equipo arrancado previamente. En la Tabla 2.2 se definen las condiciones del proceso para cada equipo que constituyen los interbloqueos.

Una vez que se encuentre funcionando la ruta seleccionada, el Paro del proceso está determinado por la condición que define el Interlock de Paro (P59) y ante el inicio de la secuencia Paro, representada por el lugar P60, se ejecuta el apagado de forma directa secuencial (PDS) de los equipos, comenzando por detener el TS_Rec (P61), luego el Elevador (P63), después el TS_Granos (P65) y por último se cierra la Racera de entrada al silo (P67), teniendo en cuenta el tiempo de evacuación (T_Evac) de cada equipo, representado por los lugares P62, P64 y P66.

Por otro lado ante la ocurrencia de alguna avería que exija el paro inmediato de los equipos se define el lugar P9 para efectuar un comando de Paro Rápido que detiene el proceso inmediatamente.

2.4.3.2 Importancia del Modelo:

Este modelo ayuda a implementar en el programa la secuencia de encendido y apagado de la etapa de Recepción a través de un algoritmo de control secuencial y estructurado. La interpretación correcta de esta red permite calcular las condiciones que constituyen Interbloqueo Tecnológico de Arranque, de Proceso y de Paro para cada dispositivo mostrado en la siguiente tabla.

Equipos	IL_Proceso	IL_Arranque	IL_Paro
Ri_Sx	Arrq_Ruta_Trab	Ruta_Sel_lista	Nmax_Sx & TE_G.
TS_Granos	Ri_Sx_open = 1	Ri_Sx_Rdy	Nmax_Sx & TE_E.
Elevador	Conf. Run_TS_G	Elev_Rdy	Nmax_Sx TE_R.
TS_Rec.	Conf. Run_Elev. & SI	TS.G_Rdy	Nmax_Sx

Tabla 2.2: Cálculo de los Interbloques Tecnológicos (Interlocks) para cada equipo.

2.4.4 Traducción de los modelos de la automática local a lenguaje de programación de PLC.

En este epígrafe se introduce el algoritmo de control para el proceso, mediante la traducción de los modelos realizados a lenguaje de programación (LD, ST, FBD).

La interpretación del modelo anterior permite concebir un programa jerárquico en varios niveles de la aplicación, donde cada nivel puede ser formado por Unidades Funcionales (UF) que se ejecutan a través de funciones desarrolladas mediante subredes propias (una entrada y una salida) que son llamadas en el programa principal. Estas subredes propias son equivalentes a funciones implementadas en el programa para ser llamadas en el programa principal, garantizando una correspondencia con la programación estructurada y modular que la IEC 61131 propone por medio del empleo de funciones y rutinas permitidas [22].

Esta metodología de trabajo con Unidades Funcionales que contienen Subredes Propias y Redes Modulares, permite establecer:

1. La implementación de Subredes Propias para ser llamadas desde un Programa Principal, lo cual es equivalente al empleo de **FC** llamadas en el OB1 para controlar cada objeto de la aplicación, favoreciendo de esta forma a la modularidad del código del programa, ver Anexo 13.2.
2. La construcción de una Red Modular para solucionar una tarea específica que puede ser común para varios equipos del sistema, siendo equivalente a un **FB** reutilizable en varias secciones del programa. (**FB_Motor y FB_Racera**).
3. Estructura Modular Jerárquica del Programa, Anexo 13.1, para la definición de un Algoritmo de Control, definido en la Fig. 2.10, que especifica la jerarquía de los **FC** respecto a los **FB**, con el objetivo de concebir un código de programación reutilizable con encapsulamiento de datos.

Un ejemplo de esta estructuración lo demuestra la Fig. 2.9, donde cada equipo es controlado mediante su Función de Control, en la cual se realiza la llamada a un único Bloque Funcional (FB), que tiene implementado la programación del equipo que se desea controlar en esta función, que en este caso es el TS_Rec mostrado en el Anexo 13.8.

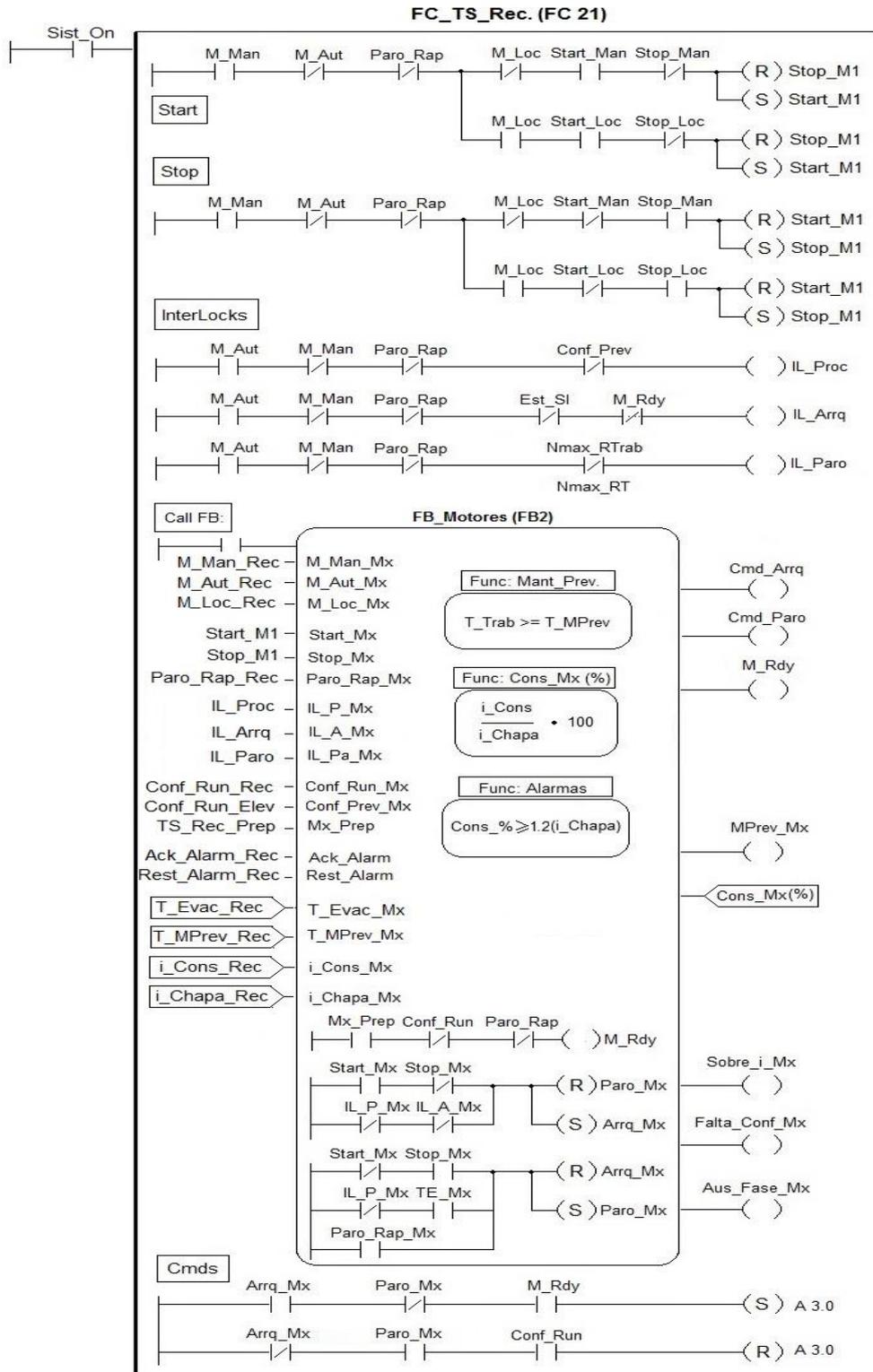


Fig.2.9 Función de Control para los motores de la planta.

Este principio de programación permite reutilizar funciones para el mantenimiento preventivo, cuyo objetivo es tener en cuenta el tiempo de trabajo de los equipos para conocer su período de mantenimiento. Esta función se aprecia en la Fig. 2.9 en la llamada del **FB_Motor** y se implementa para controlar la salida del bloque **MPrev_Mx**, que se activa una vez que el tiempo de trabajado por el equipo sea mayor o igual que el tiempo para mantenimiento definido por el usuario, siendo este un parámetro de entrada de la función de control y está almacenado en una palabra de entrada del mapa de memoria del sistema.

La llamada secuencial de funciones para cumplir con la secuencia de operación del proceso en modo Automático está definida en el FC1, mostrado en el Anexo 13.5, función que está declarada dentro de la estructura de bloques del programa llamada en el OB1, ver Anexo 13.2, cuya estructura se explica a través del algoritmo de programación presentado en la Fig. 2.10.

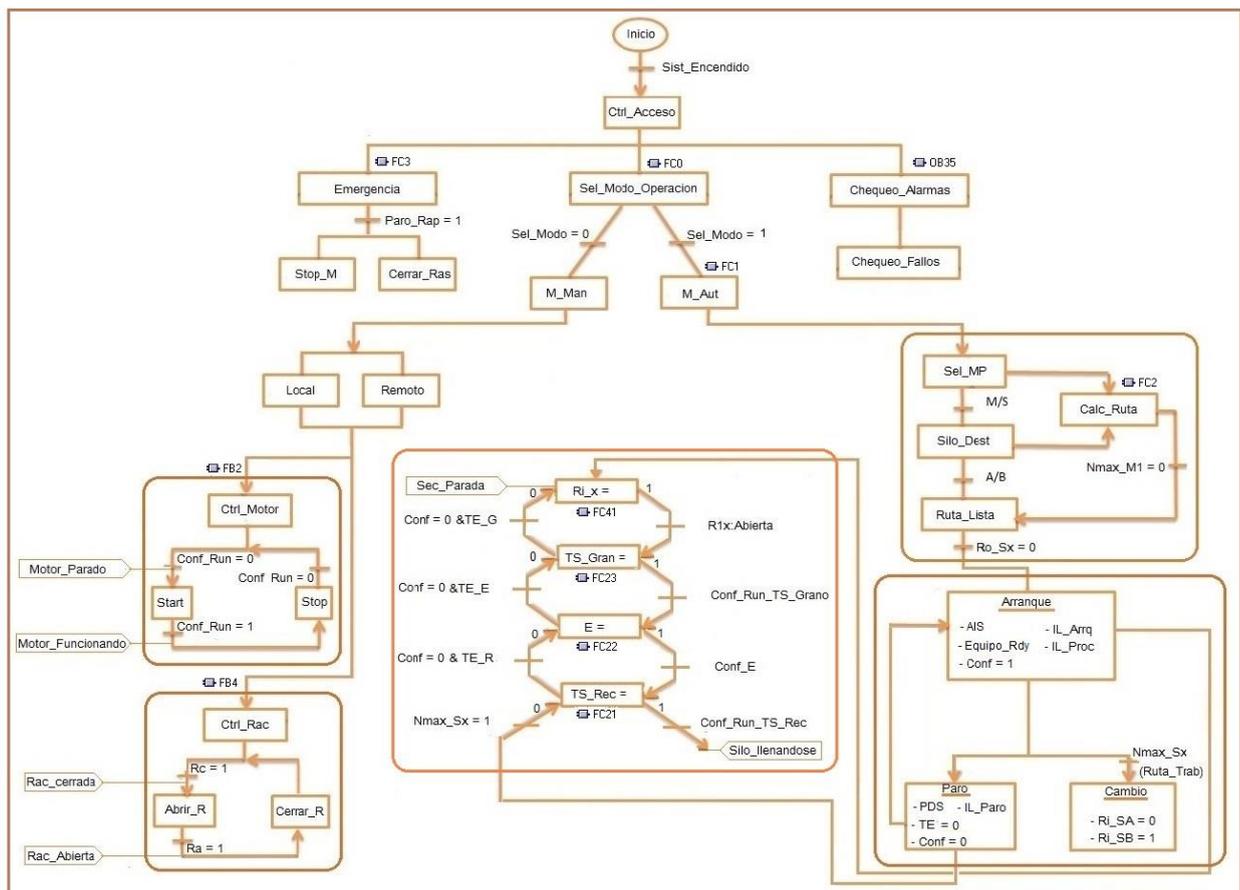


Fig. 2.10 Algoritmo de Programación que responde a la estructura de bloques en el programa.

Este algoritmo de control responde a la estructuración del programa, con el objetivo de garantizar la modularidad de la aplicación, facilitando labores de mantenimiento,

permitiendo la reusabilidad de sus bloques funcionales (FB) y la rápida identificación de la función de control (FC) de cada equipo del proceso.

2.5 Funcionamiento del Sistema Supervisorio.

El objetivo principal de este modelo es mostrar la posibilidad de Navegación entre las pantallas que ofrece el Supervisorio después de vencer la etapa de control de acceso como condición de seguridad para entrar al sistema.

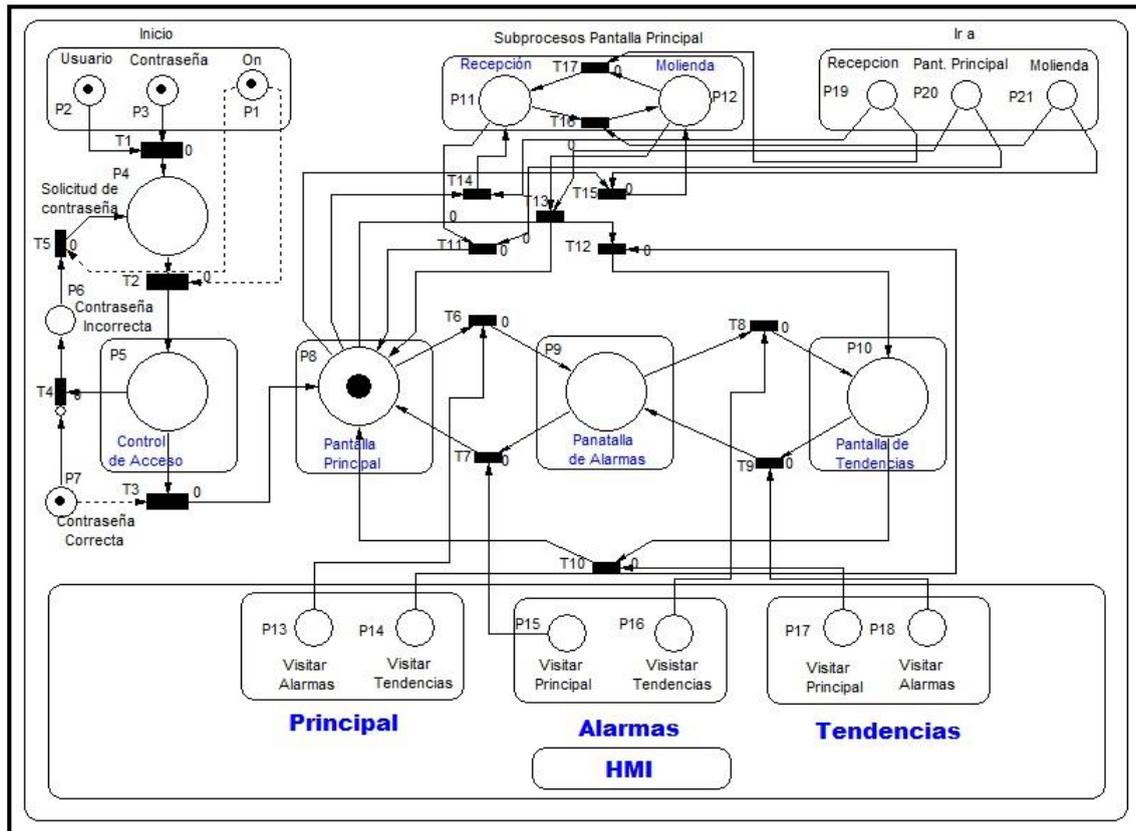


Fig. 2.11 Modelo que define el funcionamiento del Sistema Supervisorio

2.5.1 Explicación del Modelo en Redes de Petri:

Este modelo está concebido con una primera etapa donde se controla el acceso al Supervisorio a través de los estados Usuario (P2) y Contraseña (P3), que permiten entrar al Macro lugar que representa la Pantalla Principal. En esta pantalla se muestran las etapas de Recepción (P11) y Molienda (P12), teniendo la posibilidad de visitar específicamente cada una de estas áreas del proceso mediante la selección de los lugares auxiliares P19 y P21. La posibilidad de navegación hacia estas áreas específicas facilita el control y una mejor monitorización de los equipos presentes en las etapas de Recepción y Molienda, permitiendo un mayor nivel de supervisión y control. También ofrece la posibilidad de navegación hacia otras dos pantallas de suma importancia, las ventanas de Tendencias y Alarmas mediante los lugares auxiliares P13

y P14, que permiten el acceso hacia los Macro lugares P9 y P10. El acceso al Macro lugar Tendencias se diseña para graficar en forma de tendencias las variables que tienen un comportamiento continuo en el proceso, como por ejemplo la corriente consumida por los motores. El acceso al Macro lugar Alarmas se concibe para darle atención a las alarmas del proceso y tratamiento a los posibles fallos que permiten continuar con el funcionamiento del proceso.

2.5.2 Importancia del modelo:

Este modelo permite definir el control de acceso al Sistema Supervisorio posibilitando hasta tres niveles de acceso (operador, ingeniero y directivo), determinado por la jerarquía del personal de la fábrica programado a través de usuarios y contraseñas. Resulta de relevante importancia delimitar la dinámica de navegación entre las pantallas del SCADA, destacando la posibilidad de acceder desde la pantalla principal, mostrada en el Anexo 12, hacia otras dos ventanas donde se personalizan las opciones para el control de las etapas de Recepción expuesta en la Fig. 2.12 con el objetivo de mostrar de forma detallada las mediciones del proceso, el estado de funcionamiento de motores y electroválvulas y la selección dinámica de la ruta de trabajo.

2.5.3 Diseño de la pantalla de Recepción.

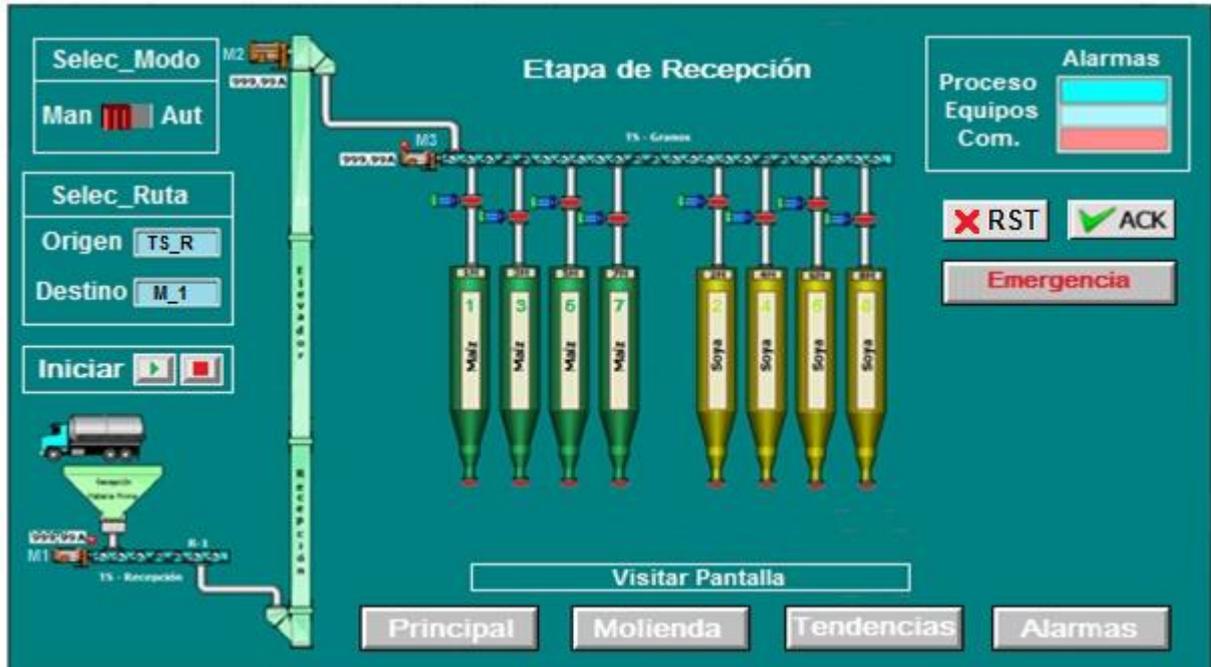


Fig. 2.12 Pantalla para el control de la etapa de Recepción desde el Supervisorio.

En la pantalla de recepción se lleva a cabo el control de la secuencia de operación de esta etapa de modo manual y automático. En modo automático es necesario seleccionar la ruta de trabajo a través de la selección del origen y destino de recepción

y una vez que se activa el botón de arranque el programa ejecuta el encendido inverso secuencial de los equipos de esta área, mientras que en modo manual el cumplimiento de la secuencia de operación la lleva a cabo el operador de forma manual a través de los Faceplates de los Motores y las Raceras, mostrados en la siguiente figura.

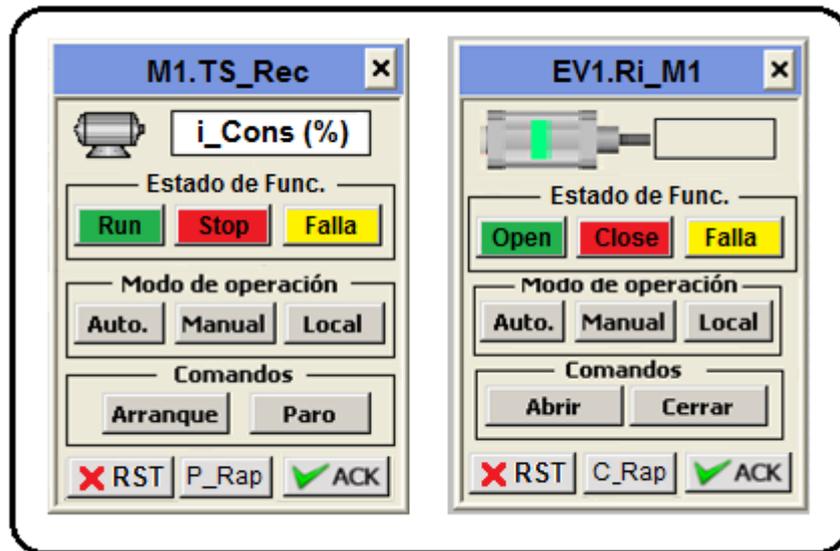


Fig. 2.13 Faceplates para el control de Motores y Raceras desde el Supervisorio.

2.6 Atención a Alarmas y Tratamiento de Fallos.

El objetivo de desarrollar un modelo para la Atención de Alarmas y el Tratamiento de Fallos es en primer lugar poder clasificar las posibles averías del sistema para poder brindarle al operador una potencial sugerencia que pueda solucionar el problema ante la ocurrencia de un fallo. Además, mediante esta red se puede modelar como responde el SAI ante el comportamiento crítico de alguna medición, con el objetivo de reconocer la alarma surgida y luego reconocer el momento en que esta ha sido solucionada.

Este modelo se crea con el objetivo de implementar en el programa una subrutina que permita reconocer las distintas situaciones críticas en el proceso, a través del OB35 (Bloque de Organización relacionado con las alarmas atendido como una interrupción cíclica cada 100ms). Esto es posible gracias a que el sistema clasifica el tipo de falla ocurrido, teniendo en cuenta el tipo de medición y el lugar donde ha sucedido, sugiriendo una posible solución para su atención, teniendo como base para la solución, el principio de funcionamiento del elemento que está relacionado con el fallo.

2.6.1 Modelo que define la función de Atención a Alarmas y Tratamiento de Fallos.

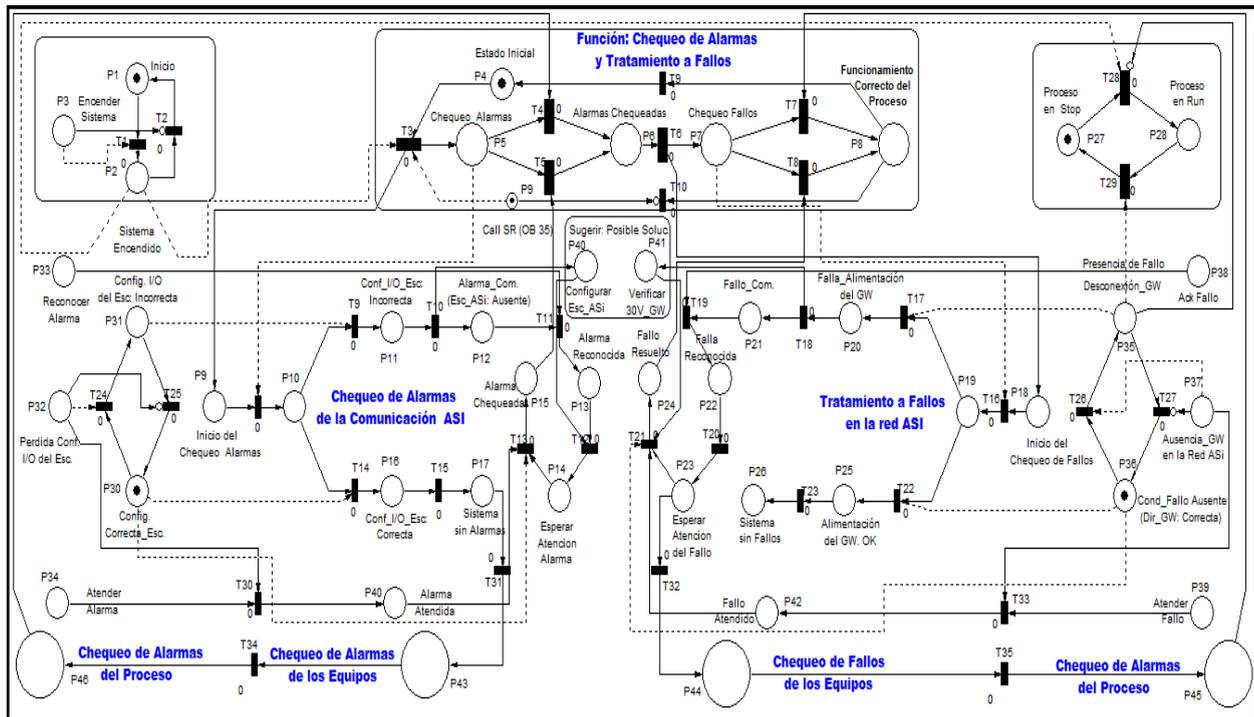


Fig. 2.14 Modelo para el Chequeo de Alarmas y Tratamiento de Fallos.

2.6.2 Explicación del Modelo en Redes de Petri:

Este modelo está conformado por la subred de atención a Alarmas iniciada con la activación del lugar P5 que permite se chequeen las condiciones que constituyen advertencias para el proceso mediante la habilitación de los lugares P9 y P10, sin llegar a detener el funcionamiento del mismo y por la subred que chequea la presencia de fallos, representada por la activación del lugar P7, cuya función es iniciar con el tratamiento de la falla a través de los lugares P18 y P19.

La función de la subred de Alarmas es reconocer la presencia de condiciones alarmantes en el proceso definidas en el lugar P31, permitiendo clasificarlas, con el objetivo de sugerirle al operador una posible solución a través del lugar P40.

La función de la subred de chequeo de Fallos es clasificar el fallo identificado en el lugar P35, permitiendo brindarle una posible solución al operador con la habilitación del lugar P41, llevando a cabo un paro inmediato del proceso (P27).

Tanto en el modelo de Alarma como en el de Fallos se debe reconocer este una vez que se ha producido y luego de localizar la avería y mostrar un mensaje con alguna sugerencia se le exige al operador la solución del mismo representado por el lugar P34 y P39.

2.6.3 Importancia del modelo:

EL modelo de Atención a Alarmas y Tratamiento de Fallos es capaz de reconocer la existencia de una condición anómala, que de empeorarse podría convertirse en una avería que ocasionaría el paro inmediato del proceso. Esta red está conformada por condiciones previamente conocidas, unas constituyen Alarmas y otras significan Fallos. Ambas deben ser reconocidas por parte del operador, por esta razón esta estructura tiene el objetivo de clasificar la advertencia sucedida en tres tipos fundamentales de Alarmas o Fallos para efectuar un mejor reconocimiento de la condición no deseada.

Clasificación de las Alarmas del proceso.

1. Alarma de Comunicación. (Ausencia temporal de un esclavo AS-i en la Red).
2. Alarma de Equipo (Alteración de un parámetro del motor: Sobreconsumo).
3. Alarma de Proceso (señal $N_{max} = 1$: indica posible condición de desbordamiento).

Clasificación de los Fallos del proceso.

1. Fallo de Proceso (Condición de Desbordamiento del tanque presente en el proceso).
2. Fallo de Equipo (Mal funcionamiento de Motores y Electroválvulas: Ausencia de confirmación de marcha del motor o Error de Posición de la Racera).
3. Fallo de Comunicación. (Transacción fallida o Ausencia de un esclavo AS-i en la Red, Falta de alimentación en el bus, o Fallo en el funcionamiento de la pasarela)

Con presencia de Alarma el sistema puede trabajar, pero con la ocurrencia de Fallo el sistema se detiene. La Alarma es una condición fuera de rango de alguna medición que significa una advertencia y no provoca el paro de la planta. El Fallo es la condición fuera del límite permisible o un comportamiento crítico de alguna señal que provoca el paro inmediato del proceso. En este modelo no están representadas los últimos dos tipo de clasificaciones, pues la secuencia es la misma y se repite para los demás tipos de alarmas y fallos, por ello esta secuencia está representada por los macro lugares P43, P44, P45 y P46.

El objetivo de clasificar cada tipo de alarma producida es primeramente reconocer y localizar la avería, comprender la causa de la misma y proporcionarle al operador las herramientas para su posible solución. Este modelo se ejecuta como una subrutina de atención a las Alarmas implementada en el OB 35, Anexo 13.3, el cual se atiende cada 100 ms como una interrupción cíclica que chequea las condiciones del proceso que pueden provocar alarmas o fallos y funciona de la siguiente manera.

- 1. Sistema sin alarmas:** cuando las mediciones del proceso están en rango, los equipos estén funcionando bien y no haya fallos de comunicación, el sistema no presenta alarmas.
- 2. Ocurrencia de una Alarma:** si alguna medición está fuera de rango el sistema define la ocurrencia de una alarma, y procede a su clasificación para poder mostrarla en el supervisorio.
- 3. Clasificación de la Alarma:** después de ocurrir la alarma el sistema la clasifica teniendo en cuenta la causa que la ha provocado y el lugar donde ha sucedido, con el objetivo de mostrarle al operador el tipo de alarma presentada sin necesidad de detener el funcionamiento del proceso.
- 4. Reconocer la Alarma:** la acción de reconocer la Alarma se realiza a través de un botón presente en el supervisorio identificado con el nombre **Ack_Alarma**. Con esta acción el programa reconoce que el operador tiene conocimiento del estado no deseado y después del reconocimiento el sistema toma la decisión de silenciar la sirena (Buzzer) que indica el comportamiento anormal en el proceso.
- 5. Atención de la Alarma:** después de reconocer la alarma el sistema espera la atención de la misma por parte del operador. Esta acción debe llevarla a cabo personal calificado que tenga dominio técnico del tipo de alarma producida teniendo en cuenta la sugerencia brindada. En este paso el sistema resetea las alarmas después que se ha solucionado la avería, reconociendo el cambio del proceso a condiciones normales de operación.
- 6. Presencia de un Fallo:** si después de que el sistema reconoce una alarma, ésta no se soluciona y la medición sobrepasa su límite permisible no deseado, la acción inmediata por parte del programa es parar el proceso y mostrar la presencia de un Fallo. Cuando acontece una condición crítica que afecta el funcionamiento del proceso ($Conf_Run_M1=0$) el sistema ejecuta la parada inmediata de la planta y pasa a clasificar el fallo sucedido.
- 7. Clasificación del Fallo:** después de la presencia de un fallo, el sistema debe clasificarlo según los tipos de fallos definidos anteriormente. Este paso permite definir la avería para poder mostrarla en el Supervisorio, permitiéndole al operador identificar la posible causa que la ha provocado con el objetivo de buscarle una solución rápida.
- 8. Reconocer el Fallo:** Después que el sistema está en Fallo es necesario reconocerlo a través del botón **Ack_Fallo**, con el objetivo de silenciar el Buzzer que indica el estado de Falla de la planta. En este paso el sistema espera por la atención del Fallo para restablecer el funcionamiento del proceso.

9. Atención del fallo: Una vez que se ha reconocido el fallo el sistema está preparado para su solución, la cual debe ser dada por personal calificado teniendo en cuenta el tipo de fallo, la causa del mismo y la sugerencia brindada. Cuando el programa reconoce el restablecimiento a condiciones normales de operación del proceso se elimina la condición de fallo existente y el sistema vuelve a la normalidad.

10. Almacenar Alarmas y Fallos: Las alarmas y los fallos sucedidos se deben almacenar en una Tabla para su posterior análisis, pudiéndose eliminar a través del botón **Rest_Alarmas** ubicado en la pantalla de Alarmas del Sistema Supervisorio.

2.6.4 Diseño de la Pantalla de Alarmas para el Sistema Supervisorio.

A continuación, se muestra la pantalla de Alarmas diseñada para mostrar en el Supervisorio las potenciales alarmas de la planta con el objetivo de clasificarlas y de brindar una sugerencia como posible solución. Esta constituye la correcta interpretación del modelo realizado en Redes de Petri y se implementa en el OB35. En la programación de este bloque se tiene en cuenta la condición que provoca el fallo según el principio de funcionamiento de los equipos del proceso, activando la variable que se utiliza para la visualización de forma animada del fallo existente cuando se cumpla esta condición en el programa. Esta ventana está concebida fundamentalmente para proporcionar al operador las herramientas necesarias para la atención de las averías que puedan sucederse y consta de los botones de **ACK** que permite el reconocimiento de la alarma y el botón **RST** para dar un restablecimiento de la falla.



Fig. 2.15: Pantalla de Alarmas del Sistema Supervisorio.

2.7 Propuesta de automatización.

Después de haber modelado cada uno de los niveles que conforma el SAI para la etapa de Recepción de la fábrica de piensos se decide mostrar la siguiente figura como la solución que satisface los requerimientos funcionales de la planta. En este esquema se detalla la presencia de un bus AS-i para los sensores y actuadores, la representación de una red Profibus DP que transmite la información hacia el nivel de control y se muestra una red MPI que permite el intercambio de datos entre el PLC (S7400) y la aplicación SCADA. Esta solución está basada en la arquitectura descrita en la Fig. 2.2 y tiene como objetivo interconectar los tres niveles de comunicación presente en el sistema, el diseño de un Centro de Control de Motores (CCM), proponiendo un Sistema Supervisor empleando las herramientas de WinCC para el control y supervisión del proceso. Para llevar a cabo esta propuesta se decide emplear tecnología Siemens con el objetivo de ofrecer una solución comparable con los estándares mundiales de automatización y control.

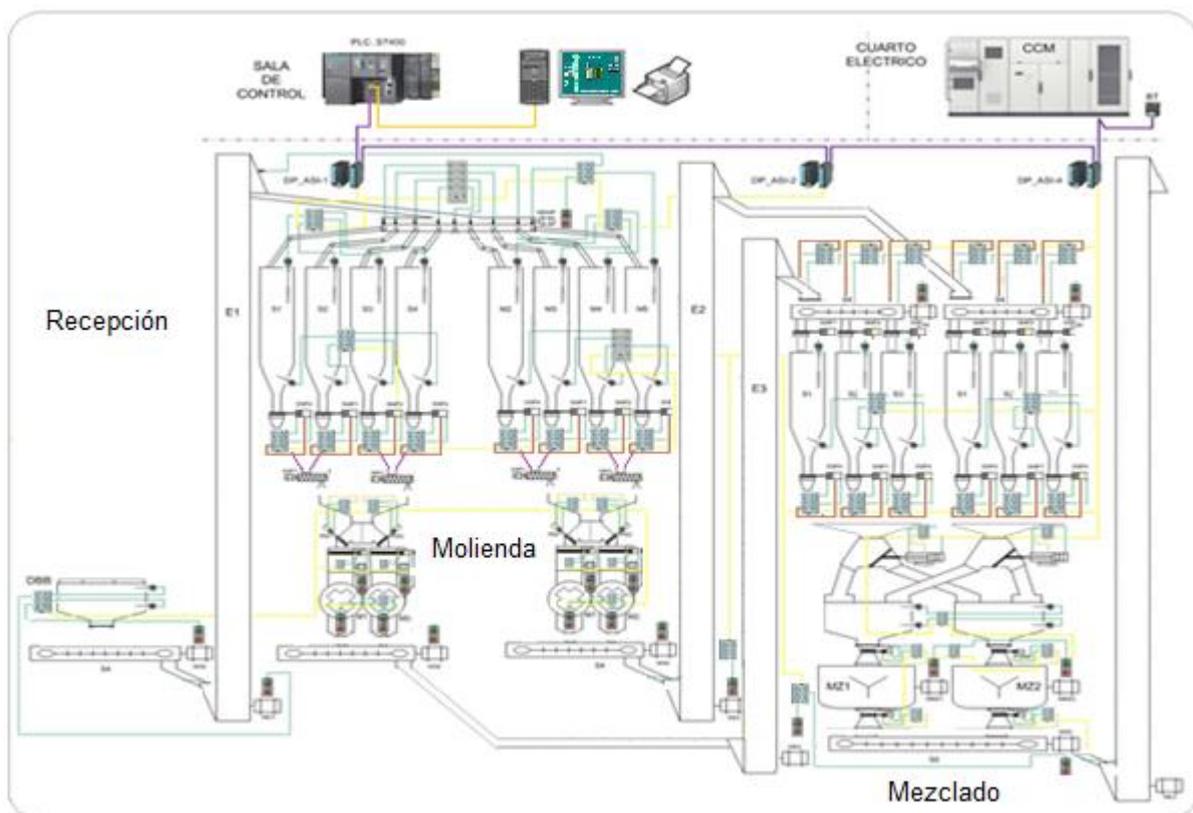


Fig. 2.16 Solución de la automatización basada en la arquitectura propuesta.

2.8 Valoración económica

En este tipo de proyectos siempre es importante realizar una valoración del costo de la propuesta teniendo en cuenta el gasto material y de mano de obra en caso de su implementación.

Las redes AS-i típicamente reducen el costo de cableado e instalación alrededor de un 50% en comparación con otros sistemas de cableado convencional. El uso de un solo cable para la conexión de los equipos discretos reduce la necesidad de gabinete y bandejas. Los ahorros generados pueden ser significativos ya que se disminuyen los costos de instalación, mantenimiento y, por ser una red simple, las horas de ingeniería.

En el Anexo 9 se presenta una tabla que relaciona la cantidad y precio del conjunto de materiales necesarios para llevar a cabo la propuesta de automatización, arrojando un resultado de 16 291.75 (CUC) en gastos materiales y 2 270.00 (CUP) en gastos por mano de obra, Anexo 10. El montaje del sistema se pronostica en un tiempo de 3 meses, por lo que el gasto de mano de obra ascendería a 6 810.00 (CUP) lo que equivale a 272.40 (CUC).

El costo total de la propuesta ascendería a un total de 53 087.91 (CUC), Anexo 11, por lo que se evidencia que los valores de la misma serían amortizados en aproximadamente 4 años teniendo en cuenta la reducción del consumo de energía, el aumento de la productividad y la eficiencia del proceso.

2.9 Valoración medioambiental.

En el marco del creciente empeño de nuestro país en la preservación del medio ambiente, se considera importante desarrollar sistemas que no solo resuelvan la problemática detectada, sino que lo hicieran de la manera más eficiente y con el menor impacto medioambiental negativo, por lo cual esta propuesta incluye:

- Los operarios no estén expuestos a altos niveles de polvo, mientras son revisados los niveles de los silos, y accionando válvulas e interruptores, lo cual representa un daño eminente para su salud.
- El ahorro de portadores energéticos.
- Con el uso de lámparas LED en las señalizaciones luminosas, se reducen los gastos energéticos y, por consiguiente, disminuye la emisión de CO₂ por concepto de generación eléctrica.

- Ergonomía de las pantallas del supervisorio y de la sala de control, proporciona al operador y técnicos de la planta mayor comodidad en el trabajo, al tener en cuenta aspectos como:
 - Situar el monitor en línea recta a la línea de visión del usuario, para que la pantalla se encuentre a la misma altura de los ojos. De esta manera no tendrá que doblar el cuello para mirarla.
 - Mantener una distancia de 50 o 60 cm. entre la persona y el monitor o a una distancia equivalente a la longitud de su brazo.
 - Evitar que la luz del ambiente produzca reflejos sobre la pantalla; en todo caso, cambiar la posición del monitor, o disminuir la iluminación del ambiente.
 - El asiento debe tener una altura que mantenga un ángulo de 90°, evitando así el dolor en las cervicales, lumbagos o problemas de disco; además debe tener un respaldo que permita apoyarse correctamente.
 - La utilización adecuada de los contrastes y colores en las pantallas de la SCADA protege la salud visual del operador.

2.10 Conclusiones del capítulo 2.

1. La modelación en Redes de Petri del funcionamiento de la comunicación AS-i permite definir los momentos de una trama con el objetivo de identificar los posibles errores de comunicación y poder repetir la transacción entre un Maestro y un esclavo.
2. El desarrollo de los modelos que definen el comportamiento del SAI permitió diseñar un algoritmo de control necesario para efectuar la programación de los requerimientos funcionales del sistema.
3. La definición de estados controlables para cada equipo de la planta en el modelo que describe la automática local permitió identificar las circunstancias de Arranque y Paro para los actuadores del proceso, lo cual permitió calcular las condiciones necesarias para el control de motores y electroválvulas que fueron programadas en forma de Interbloqueos.
4. La interpretación de los modelos realizados permitió adoptar el principio de reutilización de bloques funcionales parametrizables para el control de los actuadores de la planta, teniendo como resultado la implementación de un programa modular y bien estructurada.

Conclusiones generales

1. Incorporar las redes de comunicación industriales en el diseño de la propuesta de automatización para la fabricación de piensos permite desarrollar una solución comparable con los estándares mundiales de automatización y garantiza la relación entre los diferentes protocolos empleados para la transmisión de datos hasta el Sistema Supervisorio.

2. El desarrollo de los modelos en Redes de Petri posibilita el modelado y la programación de conceptos tecnológicos importantes empleados en automatización, como son:
 - la programación de los Interbloques tecnológicos para el control de los actuadores;
 - el mantenimiento preventivo de los equipos;
 - el chequeo de las comunicaciones para la detección de posibles errores;
 - la atención de alarmas y tratamiento de fallos.

3. Concebir el control de la secuencia de operación de la planta a través de la programación lógica de los interbloques tecnológicos con reutilización de bloques funcionales (FB) en las funciones de control (FC) de cada componente con encapsulamiento de datos (DB), garantiza la modularidad y estructuración del programa y permite la rápida identificación de las funciones de control para tareas de mantenimiento o modificación de la aplicación.

4. Desarrollar un modelo para la Atención de Alarmas y el Tratamiento de Fallos permite clasificar las posibles averías del sistema para poder brindarle al operador una potencial sugerencia que pueda solucionar el fallo.

5. El empleo de una red AS-i reduce el costo de la instalación alrededor de un 50% en comparación con otros sistemas de cableado convencional. Los ahorros generados pueden ser significativos ya que se disminuyen, además, los costos de mantenimiento y, por ser una red simple, las horas de ingeniería, aspectos a tener en cuenta para calcular el periodo de amortización del proyecto.

Recomendaciones

1. Modelar el comportamiento de las variables analógicas de las demás etapas del proceso (flujo, peso, velocidad, etc.) mediante redes de Petri híbridas para definir una correcta programación de la función de control.
2. Extender la propuesta de automatización a las áreas de Molienda y Mezclado teniendo en cuenta la solución propuesta.
3. Implementar un sistema SCADA para el control y supervisión del subproceso, que tenga en cuenta las funciones de mantenimiento preventivo y sea tolerante a fallo, además que incorpore la programación de funciones para el control gerencial del sistema como son cálculo de los niveles de productividad y de los índices de eficiencia.
4. Implementar la propuesta de automatización en la fábrica de piensos de Baire y calcular el período de amortización real del proyecto.

Bibliografía

- [1] Acosta, G.: "Diseño formal del Sistema de Automatización Integrada para la Estación de Bombeo San Juan, empleando las técnicas de Redes de Petri". MSc. Tesis 2012-15, Área: Automática. Santiago de Cuba, Cuba. Enero 2015.
- [2] Aguilar, J.; Rios Bolivar, A.; Hidrobo, F.; Cerrada, M.: "Sistemas Multi-Agentes y sus Aplicaciones en Automatización Industrial". Universidad de Los Andes, Mérida, Venezuela. 2012, 2013.
- [3] Ampliación de Automática Industrial. Bus de sensores y actuadores AS-Interface. [ref.: octubre de 2016]. Disponible en World Wide Web: <http://www.uhu.es/antonio.barragan>.
- [4] Autómatas Industriales, Sistemas SCADA [ref: Enero 2014]. Disponible en World Wide Web: <http://www.automas.org/redes/scadas.htm>
- [5] Baier, C.; Kaoten, J.: "Principles of Model Checking". The MIT Press Cambridge, Massachusetts, USA, 2008.
- [6] Bailey, D.; Wright, E.: "Practical SCADA for Industry". Editorial Elsevier. Great Britain. 2003.
- [7] Benitez, I.: "Low Cost Automation Using Ghenesys Extended PN To Modeling Mechatronic System". 8th IFAC Symposium on Cost Oriented Automation: Affordable Automation Systems. International Convention Informatica07. Havana. 12-15 febrero 2007.
- [8] Benitez, I.; Silva, J.R.; Villafurela, L.; Gomis, O.; Sudriá, A.: "Modeling extended Petri nets compatible with GHENeSys IEC61131 for industrial automation". The international Journal of Advanced Manufacturing Technology. Springer London. Vol.36. Number 11-12 Pg. 1180-1190. UK. 2008.
- [9] David, R.; Alla, H.: "Discrete, Continues, and Hybrid Petri Nets". Springer-Verlag Berlin Heidelberg 2010.
- [10] Desel, J.; Esparza, J.: "Free Choice Petri Nets". Cambridge University Press. Great Britain. 1995.
- [11] Drath, R.: "A Mathematical Approach to Describing a Class of Hybrid Systems". IEEE Workshop on Parallel and distributed Real Time Systems. Genf. 1997.
- [12] 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.

- [13] Frey, G.: "Analysis of Petri Net based Control Algorithms - Basic Properties". Proceedings of the American Control Conference ACC 2000. Chicago. June 28-30, 2000.
- [14] Frey, G.; Litz, L.: "Formal Methods in PLC Programming". Proceedings of the IEEE. SMC 2000. Nashville. TN. Octubre 08-11. 2000.
- [15] Glez, P.: "GHENESYS: Una Red Extendida Orientada a Objetos para Proyecto de Sistemas Discretos". MSc. Tesis. Departamento de Ingeniería Mecánica-Mecatrónica. Universidad de Sao Paulo. 2001.
- [16] J. Aguilar, Análisis de Máquinas de Estados Finitos Usando la Programación Evolutiva. IV CIBELEC, V JIELECTRIC, ISBN: 978-980-7185-1, 2010.
- [17] M. H. De Queiroz; J. E. R. Cury; W. M. Wonham, Multitasking Supervisory Control of Discrete-Event Systems. © Springer Science + Business Media, 2005.
- [18] Manual de funcionamiento del bus Profibus DP. [ref: diciembre 2016]. Disponible en World Wide Web: [http// www.profibus.com](http://www.profibus.com).
- [19] Manual de medición de Nivel. Transmisores Siemens, Serie Sitrans. 1994.
- [20] Manual de Sensores industriales, característica y principio de funcionamiento. [ref.: abril 2015]. Disponible en World Wide Web: [http// www.instrumentacionycontrol.net](http://www.instrumentacionycontrol.net)
- [21] Manual de Siemens. SIMATIC S7-400 [ref: agosto 2016]. Disponible en World Wide Web: [htt://www.siemens.com](http://www.siemens.com).
- [22] Memoria Descriptiva para el proyecto de automatización de pienso. CEDAI 2016.
- [23] Murata, T.: "Petri Nets: Properties, analysis and applications". Proceedings of IEEE, vol. 77, No. 4. April, 1989.
- [24] Sánchez, J.C.: "Diseño de un Sistema de Control Automatizado Eficiente para el Transporte de Bloques Hormigón en Plantas Productivas tipo POYATOS usando Redes Petri". MSc. Tesis. 2009-11. Área: Automática. Santiago de Cuba, Cuba. Junio 2011.
- [25] Sanz, M.: " Las Redes de Petri como herramienta formal en el diseño de sistemas de automatización integrada." MSc. Tesis. 2011-13. Área: Automática. Santiago de Cuba, Cuba. Junio 2013.
- [26] Valenciano, L.: "Sistema de Automatización basado en control de eventos discretos para la etapa de empaquetado de la fábrica de producción de bloques de hormigón POYATOS". MSc. Tesis. 2011-12. Área: Automática. Univ. de Oriente. Santiago de Cuba, Cuba. Mayo 2012.
- [27] WinCC. Trabajar con WinCC. [ref.: noviembre 2014]. Disponible en World Wide Web: <http://www.automation.siemens.com>.

Anexos:

Anexo 1 Sensor MiniCap



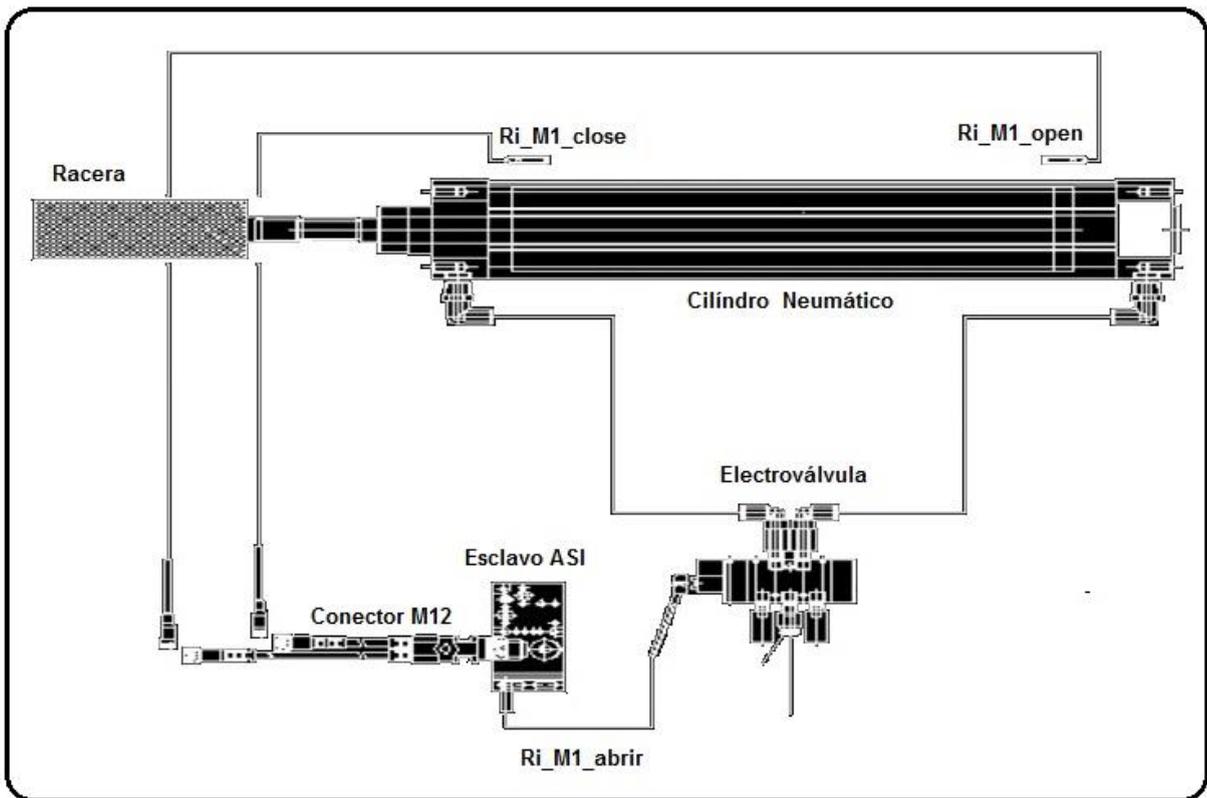
- Tipo de medición: Interruptor
- Principio de funcionamiento: Capacitivo
- Rango de medición: 50-2000mm
- Salida: Variable eléctrica para el control on/off
- Temperatura de trabajo: -50°C a 70°C
- Formas de conexión: Normal y Sanitaria
- Materiales de la caja del sensor: Plástico, Aluminio y acero inoxidable.

Anexo 2 Sensor Inductivo: Principio de funcionamiento capacitivo??????

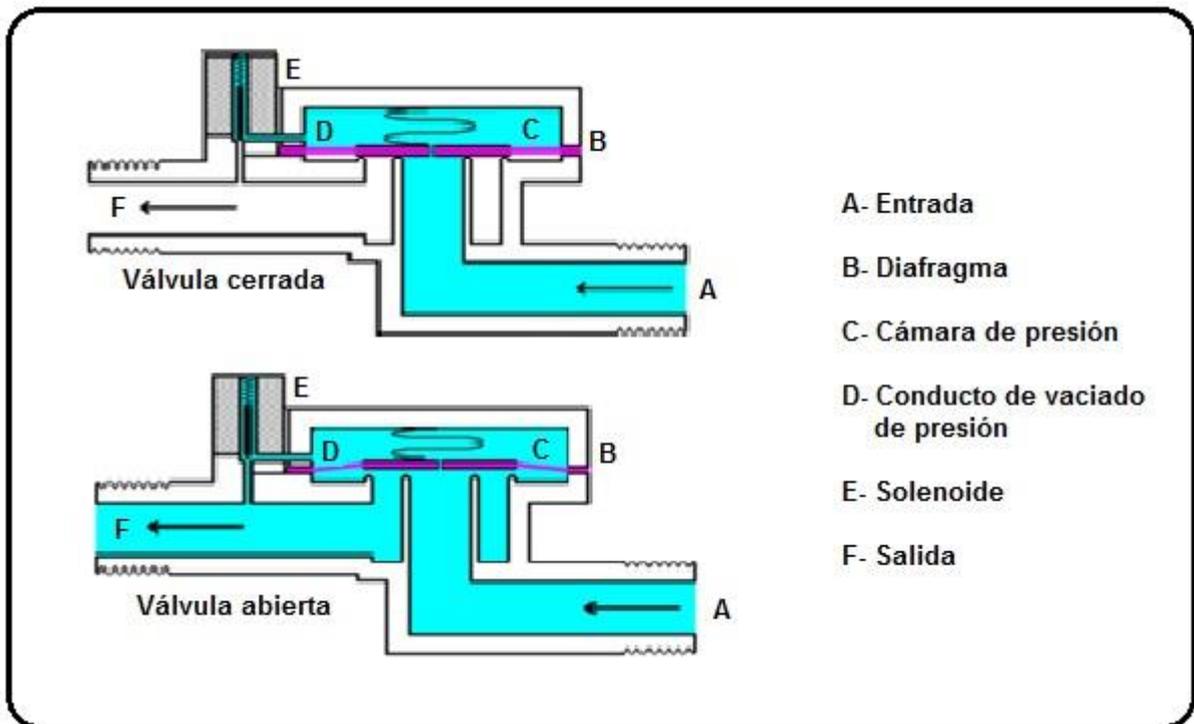


- Principio de funcionamiento: Capacitivo
- Conexión: Acoplamiento en forma de Tornillo
- Salida: Variable eléctrica para el control On/Off
- Tipo de Medición: Interruptor con lógica positiva
- Cableado: tres hilos con Alimentación de 24Vdc.
- Rango de medición: 0 – 60 cm, (sensibilidad variable)

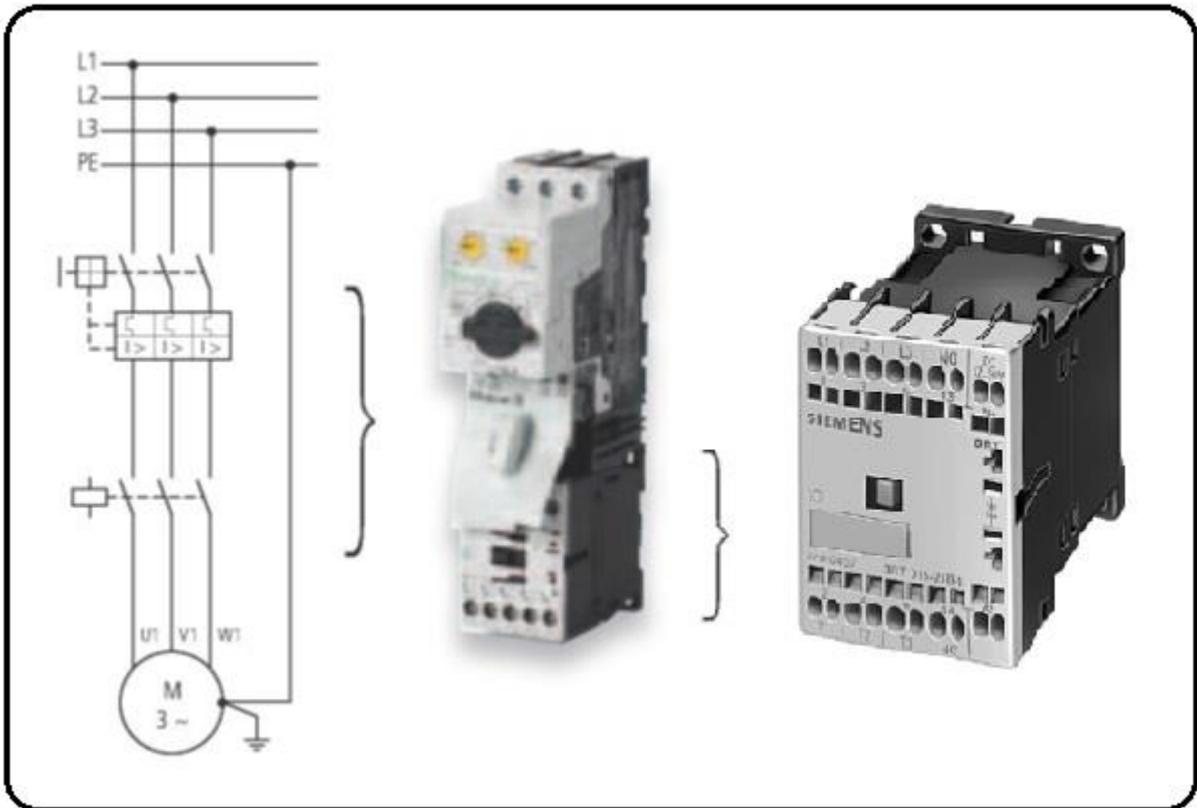
Anexo 3 Detectores de Posición Electromagnéticos:



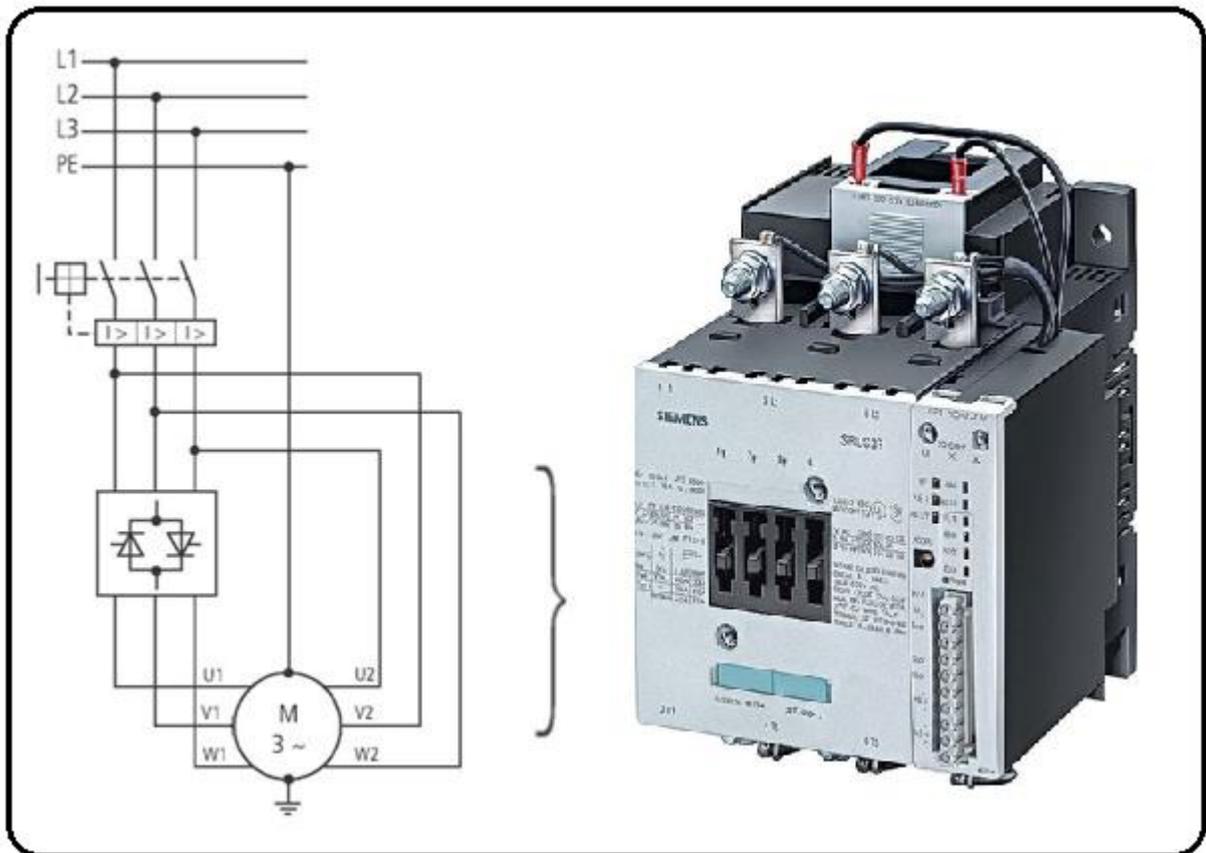
Anexo 4: Electroválvula de Simple Efecto.



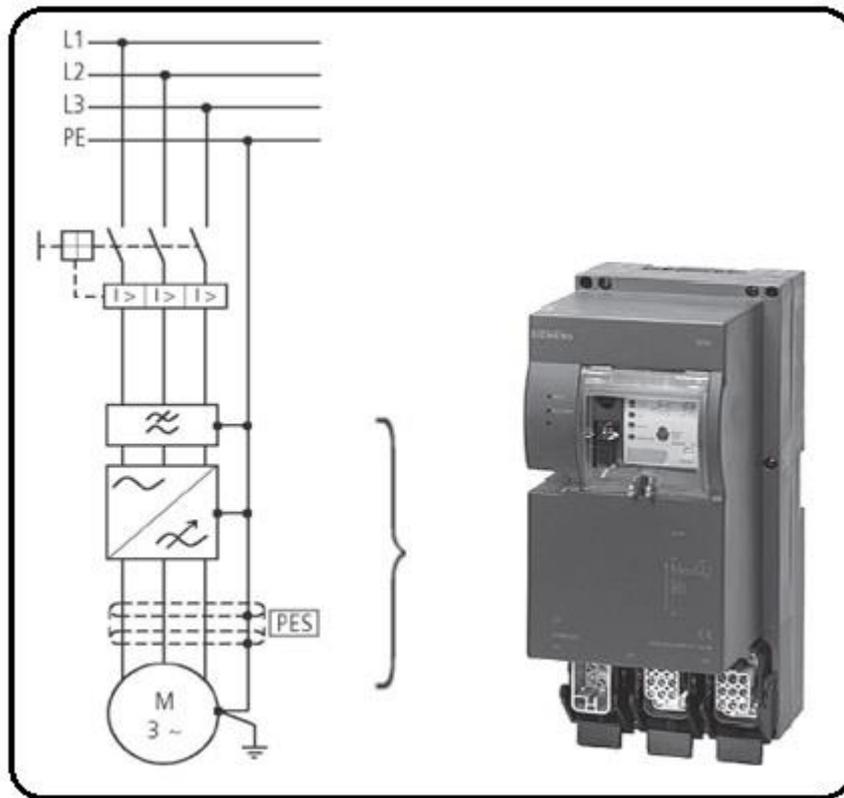
Anexo 5: Arranque directo de motores:



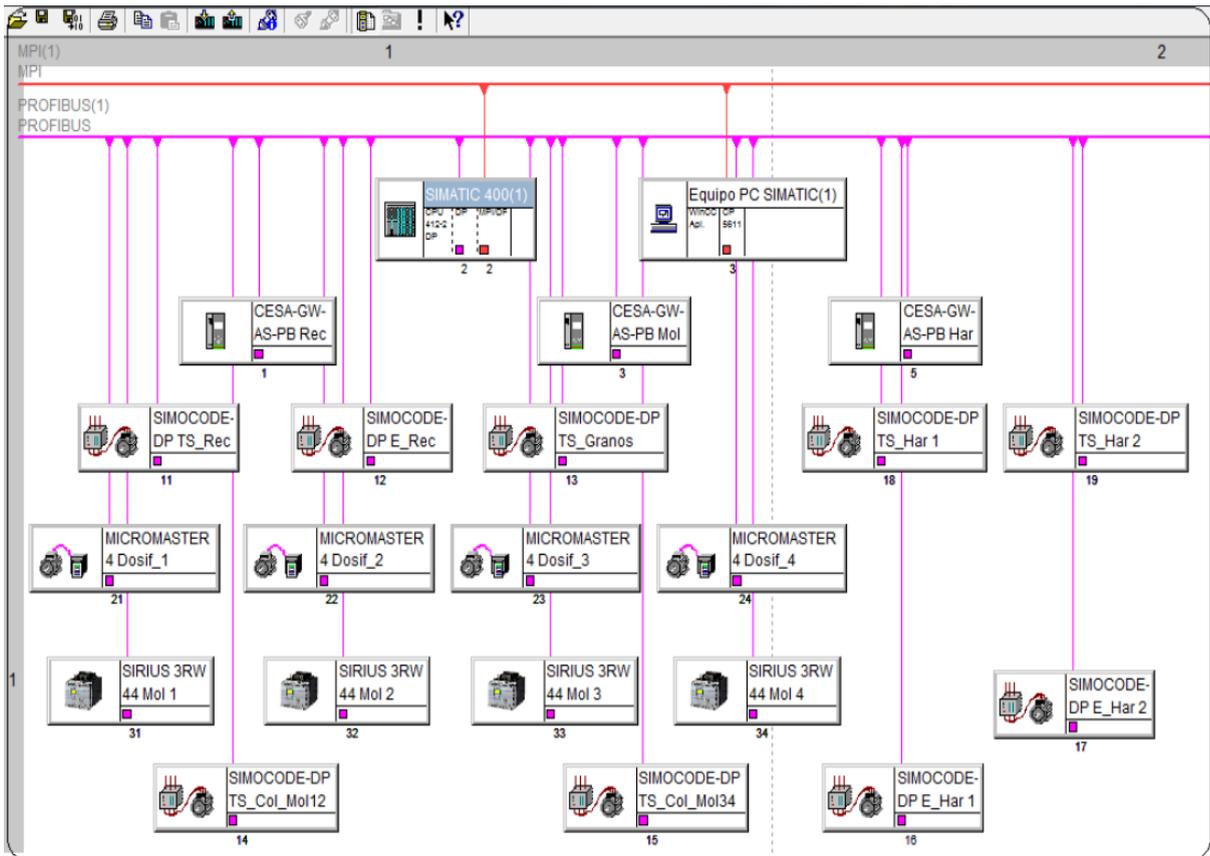
Anexo 6: Arranque Suave de Motores:



Anexo 7: Variador de Velocidad:



Anexo 8: Configuración de la red Profibus en el software de programación.



Anexo 9: Tabla 1.1 Relación de materiales.

Referencia	Denominación	Cant.	Precio (CUC)
	Equipamiento AS-i		
RK1400-1BQ20-0AA3	Modulo compacto AS-i K45, IP67, Digital, 2 Entradas/Salidas 2X1 Entrada, MAX. 200MA, PNP, 2X1 salida, electr.; 1,5A, DC 24V MAX. 3A	10	575.00
3RK1400-1MQ01-0AA4	Arrancador AS-INTERFACE 24V DC IP67, Digital, 4E/2S, arrancador doble modelo K60, 2X2 entradas, 200mA, asig. y, 2X salida, 3A para 2X motores 24V DC;	4	276.60
3RK1901-1KA00	AS-INTERFACE, tapa de cierre M12, módulos IP 67	11	70.33
3RK1901-1MA00	Módulo de acoplamiento. AS-Interface C. Hembrilla de dirección integral para 2 cables AS-i (AMARILLO)	1	33.00
3RK1901-1MX00	Accesorios AS-Interface extensión Plug duplicar longitud de cable a 200m	1	42.75
3RK1901-1MX01	Accesorios AS-Interface extensión plug plus para duplicar la longitud a 200m de un segmento AS-I con esclavo integrado. para diagnostico grado de IP67	4	116.60
3RK1901-1NR10	Derivación AS-Interface M12 AS-I SIN U-AUX, con conector M12 IIP 67/68/69K, MAX. 4A	5	91.25
3RK1901-1NR22	Derivación AS-Interface M12 AS-I Y U-AUX, cable 2M con conector angular M12 IP 67/68/69K, MAX. 4A	1	31.40
3RK1901-1TR00	Accesorios AS-Interface DIST. EN T IP68 AS-I + 24V DC para cable redondo M12, 1 X conector M12 , 2H	4	31.20
3RK1901-2DA00	Placa de montaje K45 AS-I para fijación a perfil o sistemas de perfiles, 1X conductor AS-I (AMARILLO), 1X conductor AS-I (NEGRO)	11	159.50
3RK1901-3QA00	Clip de conductor para adaptador, conductor de cable plano	1	21.00
3RK2703-3BB02-1AA1	Manual del sistema AS-Interface (Edición en Ingles), informaciones técnicas	1	15.50
3RX9012-0AA00	Conductor AS-I, perfilado, amarillo, cubierta de caucho, 2X 1.5QMM, 1000M, 1 PIEZA = 1000M	3	2095.80
3RX9017-0AA00	Cable perfilado AS-Interface amarillo; TPE; 100 M	5	362.25
3RX9502-0BA00	AS-I Power 5A 120V/230VAC IP20, fuente de alimentación AS-Interface, IN: 120V/230V AC, OUT: AS-I, 5A (30V DC), IP20, con detector de sobre carga y desacoplamiento de datos AS-I	8	758.40
3RX9801-0AA00	Red AS-I transferencia de conductor AS-I al M12.	6	86.10
3SF5402-1AA03	Adaptador F para AS-Interface para paro de emergencia 3SB3, bornes de tornillo, 2E	1	21.70
3SF5402-1AA04	Adaptador F para AS-Interface para paro de emergencia 3SB3, bornes de resorte, 2E	1	21.70
6ES5497-4UC11	Extractor para terminales tipo pinza	1	32.20

Anexos

6ES7158-0AD01-0XA0	Acoplador DP/DP para conectar dos redes PROFIBUS DP	1	46.00
6ES7194-1KA01-0XA0	Y-Enchufe M12 para conexión de sensor	6	255.90
	Automática		
6ES7407-0DA02-0AA0	Fuente de alimentación PS407 4A; AC 120/230V -> DC 5V/24V	1	241.85
6ES7400-1AJ01-0AA0	UR2, aparato central/ampliación; 9 conectores, bus K	1	81,20
6ES7414-2XK05-0AB0	S7-CPU 414-2; 2x256KB de mem. de trabajo; 1 MPI/DP, 1 DP	1	1028.0
6ES7422-1BH11-0AA0	Salida digital 16DA; DC 24V; 2A	1	308.90
6ES7431-1KF00-0AB0	Entrada analógica 8EA, 13 bits	1	561.00
6ES7492-1AL00-0AA0	Conector frontal para módulos de señales; bornes de tornillo	2	89.40
6ES7492-1BL00-0AA0	Conector frontal para módulos de señales; bornes de resorte	1	44.70
6ES7492-1CL00-0AA0	Conector frontal para módulos de señales; term. pinza	1	44.70
6ES7810-4CC10-0YE5	Step7 V5.5, Floating License, Upgrade a V5.5	1	342.80
6ES7921-4AD00-0AA0	Módulo de enchufe frontal para módulos de salidas digitales 2A	1	102.15
6ES7921-4AG00-0AA0	Módulo de enchufe frontal para módulos analógicos	1	188.65
6ES7923-0BC50-0CB0	Cable de conexión sin pantalla con conectores IDC, 2,5 m	2	71.60
6ES7923-0BC50-0DB0	Cable de conexión apantallado con conectores IDC, 2,5 m largo	4	229.00
6ES7924-0BB10-0AA0	Módulo de conexión TP2 para conexión a 2 hilos (2A, 8 SD)	2	126.00
6ES7924-0CC10-0AA0	Módulo de conexión TPA para señales analógicas	4	284.60
6ES7952-0KF00-0AA0	Flash Memory Card, larga; 64 kbytes	1	32.50
6ES7971-0BA00	Pila tampón 3,6 V/1,9 Ah para PS 405/PS 407	1	64.50
6ES7972-0AA02-0XA0	Repeater RS 485 para PROFIBUS	2	139.60
6ES7972-0BA52-0XA0	PROFIBUS FastConnet Enchufe RS485, Salida de cables 90°	5	99.50
6ES7972-0BB52-0XA0	Conector RS485 PROFIBUS FastConnet, salida de cable 90°	25	657.50
6ES7972-0DA00-0AA0	Elemento de cierre activo RS 485 para PROFIBUS	2	63.00
6GK1210-0SA01	Repetidor para AS-Interface	6	628.80
6GK1415-2AA10	DP/AS-i Link 20E; transición de red, IP20	4	1630.60
6GK1500-0FC10	PROFIBUS Conector FastConnet RS485, Sal. cable axial (180°)	1	19.90
6GK1905-0AA00	SplitConnect Tap; 10 Piezas	10	117.00
6XV1831-2A	PROFIBUS FC cable estándar IS GP, de 20m a 1000m	10	1615.00

Anexos

	Equipamiento Eléctricos (Control de Motores)		
3RA6120-0AB30	SIRIUS, derivación compacta, Arrancador directo 690 V, AC/DC 24 V, 50-60 HZ, 0,1-0,4 A, IP20.	4	565.00
3RA6250-0AB30	SIRIUS, derivación compacta, arrancadores inversores 690 V, AC/DC 24 V, 50-60 HZ, 0,1-0,4 A, IP20	4	1150.60
3RA6812-8AB	Bloque circuito principal: conexión por tornillos, 3 SLOT P/ derivación compacta conexión bornes MAX.	1	77.60
6SL3511-0PE17-5AM0	SINAMICS G110d AC-Drive, without Repair Switch. with filter	4	1799.00
6RL531-ELH34-JR45	Bloque de control, motor de arranque directo, SIMOCODE-PRO 690 V, AC/DC 24 V, 50-60 HZ, 0,1-0,4 A,	4	896.60
6XV1830-5EH10	SIMEAIC NEE, PS, FIBU PFC CAB ES DE BUS P. IEC 1158-2 OLORPDE CUBIERTIBAZUL PLO BAPLICACCOAES 6X,PBIFILAR,BAPANTALLADS, VEF A PS7 METRSS:1000M	1	86.40
	Cable de fuerza AWG10, long.100mXBob	5	755.75
	Cable de control AWG18, long.100mXBob.	5	263.50
Total	16 291.75 CUC		

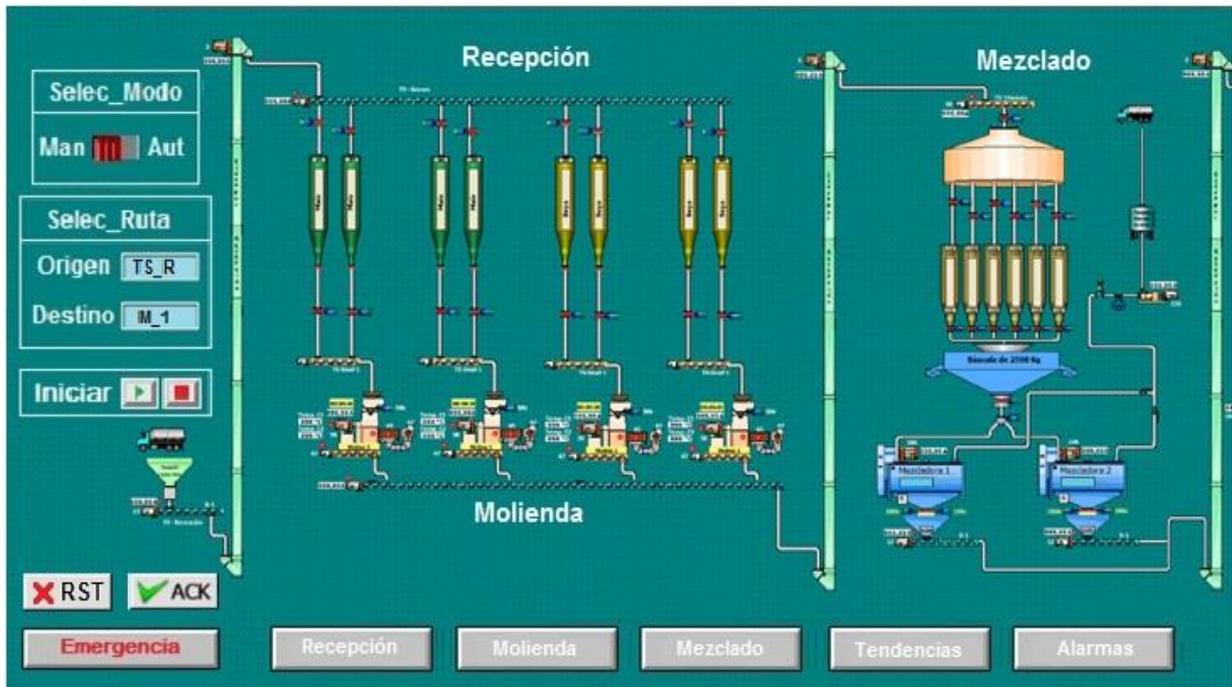
Anexo 10: Tabla 1.2 Gastos en mano de obra

Denominación	Cantidad	Salario (CUP)	Importe(CUP)
Ingeniero	1	450	450
Técnico	1	380	380
Montadores	2	325	650
Ayudantes	2	270	540
Chofer	1	250	250
Total			2 270.00

Anexo 11: Tabla 1.3: Costo total de la propuesta

No	Denominación	Importe(CUC)
1	Gasto material	16 291.75
2	Mano de obra	272.4
3	Costo directo (1+2)	16 564.15
4	Costo indirecto (5 % de 3)	828.21
5	Costo total (3+4)	1 7392.36
6	Ganancia (10 % de 5)	1 739.24
	Costo Total	53 087.91

Anexo 12: Pantalla Principal



Anexo 13: Programación

Anexo 13.1: Estructura del programa

The screenshot shows the SIMATIC Manager interface. On the left, a tree view displays the project structure: 'Piensos_Baire_Recepcion' containing 'SIMATIC 400(1)', 'CPU 412-2 DP', and 'Programa S7(1)'. Under 'Programa S7(1)', there are 'Fuentes' and 'Bloques'. The main window displays a table of objects with the following columns: 'Nombre del objeto', 'Nombre simbólico', 'Lenguaje', 'Tamaño en la memor...', 'Tipo', 'Versión (encabezado)', and 'Nombre (e...'. The table lists various objects including system data, organizational blocks (OB), function blocks (FB), function calls (FC), and data blocks (DB).

Nombre del objeto	Nombre simbólico	Lenguaje	Tamaño en la memor...	Tipo	Versión (encabezado)	Nombre (e...
Datos de sistema	---	---	---	SDB	---	---
OB1	Main	AWL	106	Bloque de organizaci...	0.1	
OB35	Chequeo_Alarmas	KOP	80	Bloque de organizaci...	0.1	
FB1	Sel_MP_Ruta	KOP	38	Bloque de función	0.1	
FB2	FB_Ctrl_Motores	KOP	38	Bloque de función	0.1	
FB4	FB_Ctrl_Rac	KOP	38	Bloque de función	0.1	
FC0	Sel_Modo_Trabajo	KOP	54	Función	0.1	
FC1	Sec_Oper_Rec	KOP	242	Función	0.1	
FC2	Sel_Ruta	KOP	140	Función	0.1	
FC3	Parada de Emergencia	KOP	68	Función	0.1	
FC21	Ctrl_TS_Rec	KOP	888	Función	0.1	
FC22	Ctrl_Elev	KOP	888	Función	0.1	
FC23	Ctrl_TS_Granos	KOP	888	Función	0.1	
FC41	Ctrl_Ri_M1	KOP	38	Función	0.1	
FC42	Ctrl_Ri_S2	KOP	38	Función	0.1	
FC43	Ctrl_Ri_M3	KOP	38	Función	0.1	
FC44	Ctrl_Ri_S4	KOP	38	Función	0.1	
FC45	Ctrl_Ri_M5	KOP	38	Función	0.1	
FC46	Ctrl_Ri_S6	KOP	38	Función	0.1	
FC47	Ctrl_Ri_M7	KOP	38	Función	0.1	
FC48	Ctrl_Ri_S8	KOP	38	Función	0.1	
DB21	Datos_TS_Rec	DB	70	DB de instancia del...	0.1	
DB22	Datos_Elev	DB	70	DB de instancia del...	0.1	
DB23	Datos_TS_Granos	DB	70	DB de instancia del...	0.1	
DB41	Datos_Ri_M1	DB	44	DB de instancia del...	0.1	
DB42	Datos_Ri_S2	DB	44	DB de instancia del...	0.1	

At the bottom of the window, there is a status bar with the text 'Pulse F1 para obtener ayuda.' on the left, 'PC Adapter(MPI)' in the center, and '106 bytes' on the right.

Anexo 13.2: OB1: Programa Principal

OB1 : "Main Program Sweep (Cycle)"

Comentario:

Segm. 1: Llamada de la funcion de Seleccion del Modo de Trabajo

En esta funcion se define el modo de trabajo del sistema (Man/Aut)

```
CALL "Sel_Modo_Trabajo"      FC0
CALL "Parada de Emergencia"  FC3
```

Segm. 2 : Llamada de la funcion Sel_Ruta para calcular la Ruta de Trabajo

Con los par. de ent. Org y Dest se calc. la Ruta_Trab_sel.por el operador, (par. necesarios para conocer cuales son los equipos q se arranacarán)

```
CALL "Sel_Ruta"              FC2
Org      :="Org_WinCC"       MW300
Dest     :="Dest_WinCC"     MW302
Ruta_Calc:="Ruta_Trab"      MW304
```

Segm. 3 : Ejecutar la Sec_de Operacion de Recepcion

M_Man:

M_Aut:Una vez se haya elegido la ruta se lleva a cabo el encendido en sentido inverso sec. de los equipos que conforman la ruta seleccionada, calculando los cmd de Arrq y Apert empleados en las entradas de FC para el ctrl de Mot y Rac.

```
CALL "Sec_Oper_Rec"         FC1
```

Segm. 4: Paro Rapido del proceso ante un comando de Emergencia

Efectuar el paro inmediato del proceso ante un comando de paro, ya sea desde WinCC o desde el hongo en el nivel de campo

```
CALL "Parada de Emergencia" FC3
```

Segm. 5: Llamada de la Funcion de control de las Raceras

En estas funciones se implementa el funcionamiento de los Motores a través de la llamada del FB_Ctrl_Motores (FB2) con los datos de cada motor segun corresponda, declarados en los DB (21 22 y 23)

```
CALL "Ctrl_TS_Rec"          FC21
CALL "Ctrl_Elev"            FC22
CALL "Ctrl_TS_Granos"       FC23
```

Segm. 6 : Llamada de la Funcion de control de las Raceras

En estas funciones están programados los comandos para el control del funcionamiento de las Raceras a traves de la llamada del FB_Ctrl_Raceras con los Datos de cada Racera.

```
CALL "Ctrl_Ri_M1"           FC41
CALL "Ctrl_Ri_S2"           FC42
CALL "Ctrl_Ri_M3"           FC43
CALL "Ctrl_Ri_S4"           FC44
CALL "Ctrl_Ri_M5"           FC45
CALL "Ctrl_Ri_S6"           FC46
CALL "Ctrl_Ri_M7"           FC47
CALL "Ctrl_Ri_S8"           FC48
```

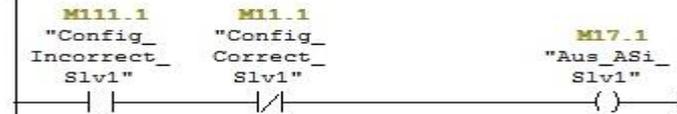
Anexo 13.3: OB35: Alarmas

OB35 : "Cyclic Interrupt"

Comentario:

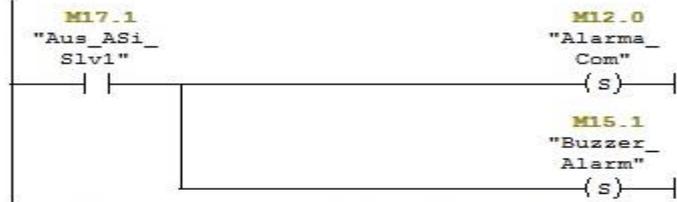
Segm. 1: Detectar la Ocurrencia de la Alarma

Alarma: Ausencia en la red ASi del Esclavo_1



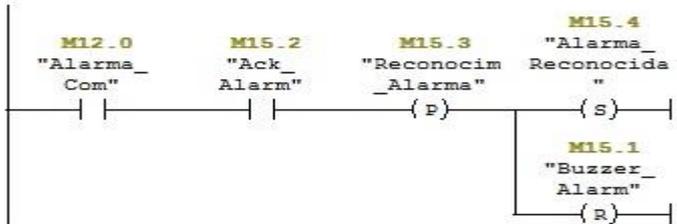
Segm. 2 : Clasificacion de la Alarma y encender la Sirena

Alarma de Tipo: Comunicacion



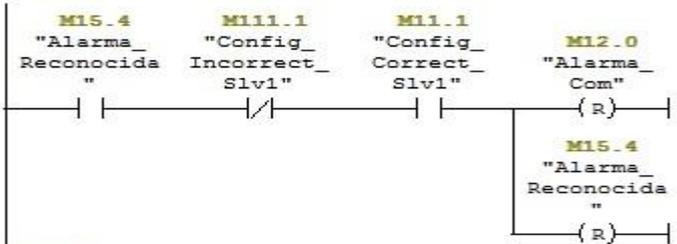
Segm. 3 : Reconocimiento de la Alarma y Apagar la Sirena

Despues q existe la Alarma es necesario reconocerla para luego atenderla. Cuando sucede este evento se pasa a silenciar el Buzzer.



Segm. 4 : Detectar cuando se ha solucionado la Alarma

Cuando se soluciona la Alarma se limpian los bit relacionados con estas.



Segm. 5: Reconocer la presencia de alguno de los tres tipo de Alarmas

Ante cualquier tipo de alarma se activa el bit que indica Presencia de Alarmas



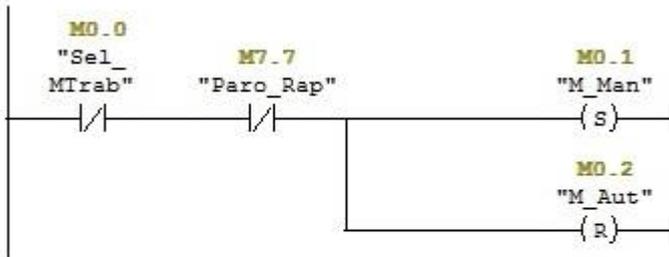
Anexo 13.4: FC0: Selección del modo de trabajo

FC0 : Funcion para determinar el Modo de Trabajo del Sist (Man/Aut)

El sist. inicia con la señal Sel_Modo_Trab = 0,
por defecto ent. el sist. inicia en modo Man.

Segm. 1: Seleccion del Modo de Trabajo Man.

Si no está activo la señal del Modo de Trabajo (Sel_Modo_Trab),
entonces el sistema se trabajara en modo Manual



Segm. 2: Seleccion del Modo de trabajo Aut.

Si está activo la señal del Modo de Trabajo (Sel_MTrab),
entonces el sistema se declara en modo Automatico



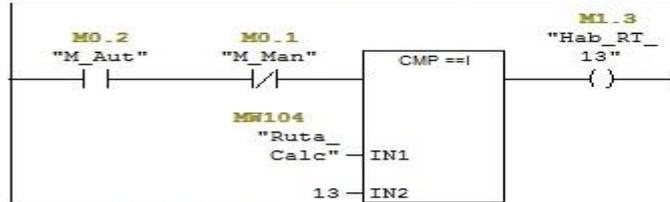
Anexo: 13.5 FC1

FC1 : Cumplimiento de Sec. de Arranque de Etapa Recepcion

Comentario:

Segm. 1: Habilitacion de la la Ruta de Trabajo #13

Cuando la variable Ruta_Calc (salida de la funcion de Seleccion_Ruta) sea igual 13 (el valor de las decenas equivalen al Org y el valor de las unidades equivalen al dest.), ent. habilite la Ruta de Trabajo 13. Nota: Hacer lo mismo para las demas rutas



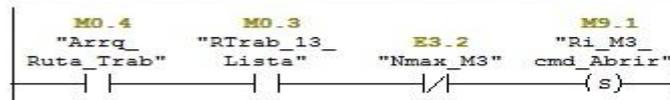
Segm. 2 : Ruta Lista

Cuando este habilitada la ruta 13 y no este activo el sensor de NMax_M3 la Ruta esta lista para empezar a trabajar



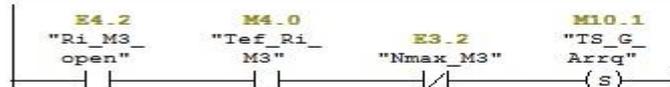
Segm. 3 : Inicio del encendido de los equipos pertenecientes a Ruta 13

Inicio del Arranque Inverso Secuencial (AIS)
Abrir la Ri_M3 teniendo en cuenta IL_Arrq, IL_Proc y IL_Paro



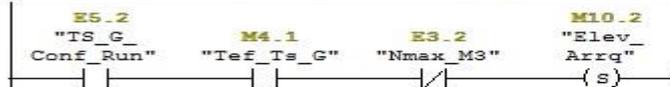
Segm. 4 : Dar orden de Arrq al TS_G, despues de abrir la Ri_M3

Enciende el TS_G teniendo en cuenta las condiciones en el proceso que determinan su momento de arranque (Interbloqueo tecnologicos)



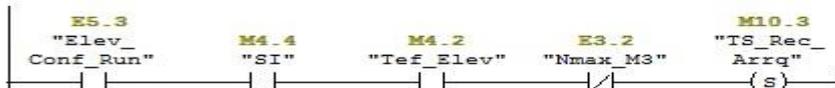
Segm. 5 : Dar orden d arrq al Elev despues que TS_G esta trab. sin fallo

Enciendo el Elev teniendo en cuenta las condiciones en el proceso que determinan su momento de arranque (Interbloqueo tecnologicos)



Segm. 6 : Dar orden d arrq al TS_R despues que Elev esta trab. sin fallo

Enciendo el TS_R teniendo en cuenta las condiciones en el proceso que determinan su momento de arranque (Interbloqueo tecnologicos)



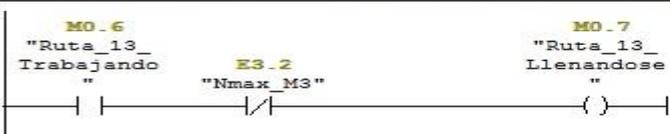
Segm. 7 : Señalar Ruta Trabajando

Cunado todos los equipos pertenecientes a la Ruta13 estan activos, estoy en condiciones de dar la señal: Ruta 13 Trabajando



Segm. 8 : Tanque M3: LLenandose

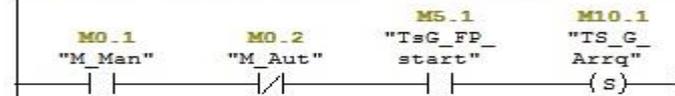
Cuando la Ruta 13 esta activa y no este hecho el sensor de niv_Max del silo de Maiz 3, entonces este silo se estará LLenando



Anexos

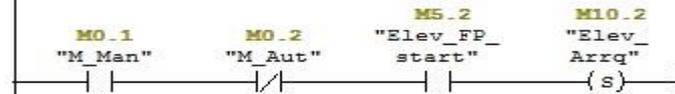
Segm. 9 : Mando Manual de TS_G

Cuando el sistema este modo manual y se efectue el comando de arrq de TS_G desde el FacePlate, se ejecuta el comando Abrir para el Motor



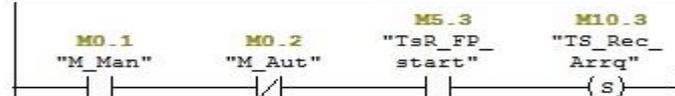
Segm. 10 : Mando Manual de Elev

Cuando el sistema este modo manual y se efectue el comando de arrq de Elev desde el FacePlate, se ejecuta el comando Abrir para el Motor



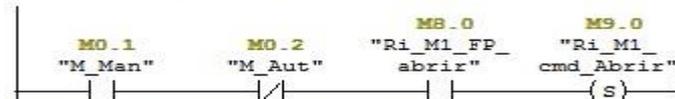
Segm. 11 : Mando Manual de TS_Rec

Cuando el sistema este modo manual y se efectue el comando de arrq de TS_Rec desde el FacePlate, se ejecuta el comando Abrir para el Motor



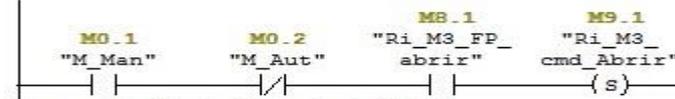
Segm. 12 : Mando Manual de Ri_M1

Cuando el sistema este modo manual y se efectue el comando de abrir Ri_M1 desde el FacePlate, se ejecuta el comando de apertura de la EV.



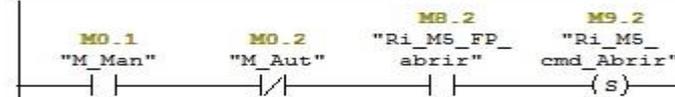
Segm. 13 : Mando Manual de Ri_M3

Cuando el sistema este modo manual y se efectue el comando de abrir Ri_M3 desde el FacePlate, se ejecuta el comando de apertura de la EV.



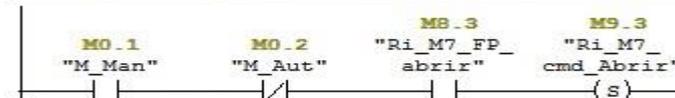
Segm. 14 : Mando Manual de Ri_M3

Cuando el sistema este modo manual y se efectue el comando de abrir Ri_M3 desde el FacePlate, se ejecuta el comando de apertura de la EV.



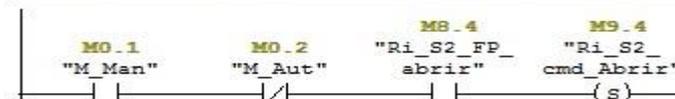
Segm. 15 : Mando Manual de Ri_M3

Cuando el sistema este modo manual y se efectue el comando de abrir Ri_M3 desde el FacePlate, se ejecuta el comando de apertura de la EV.



Segm. 16 : Mando Manual de Ri_M3

Cuando el sistema este modo manual y se efectue el comando de abrir Ri_M3 desde el FacePlate, se ejecuta el comando de apertura de la EV.



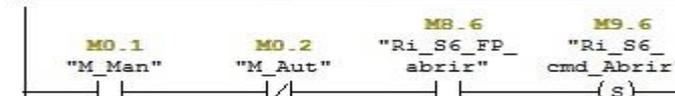
Segm. 17 : Mando Manual de Ri_M3

Cuando el sistema este modo manual y se efectue el comando de abrir Ri_M3 desde el FacePlate, se ejecuta el comando de apertura de la EV.



Segm. 18 : Mando Manual de Ri_M3

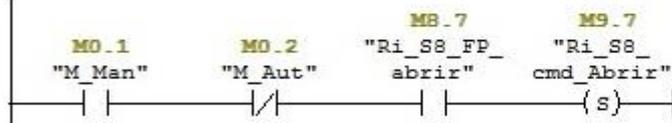
Cuando el sistema este modo manual y se efectue el comando de abrir Ri_M3 desde el FacePlate, se ejecuta el comando de apertura de la EV.



Anexos

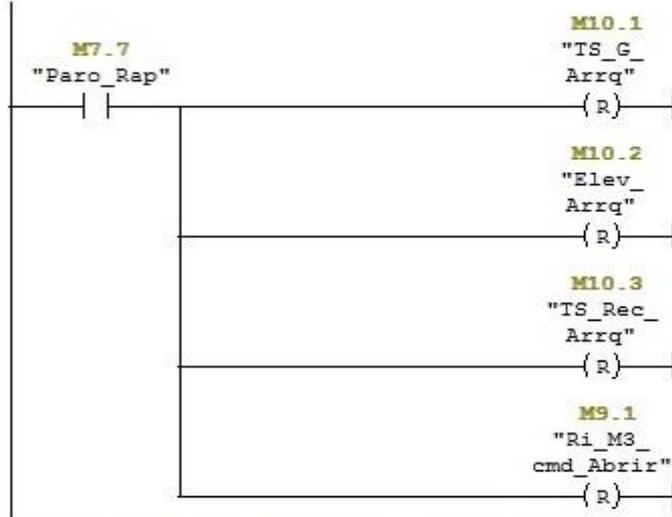
Segm. 19 : Mando Manual de Ri_M3

Quando el sistema este modo manual y se efectue el comando de abrir Ri_M3 desde el FacePlate, se ejecuta el comando de apertura de la EV.



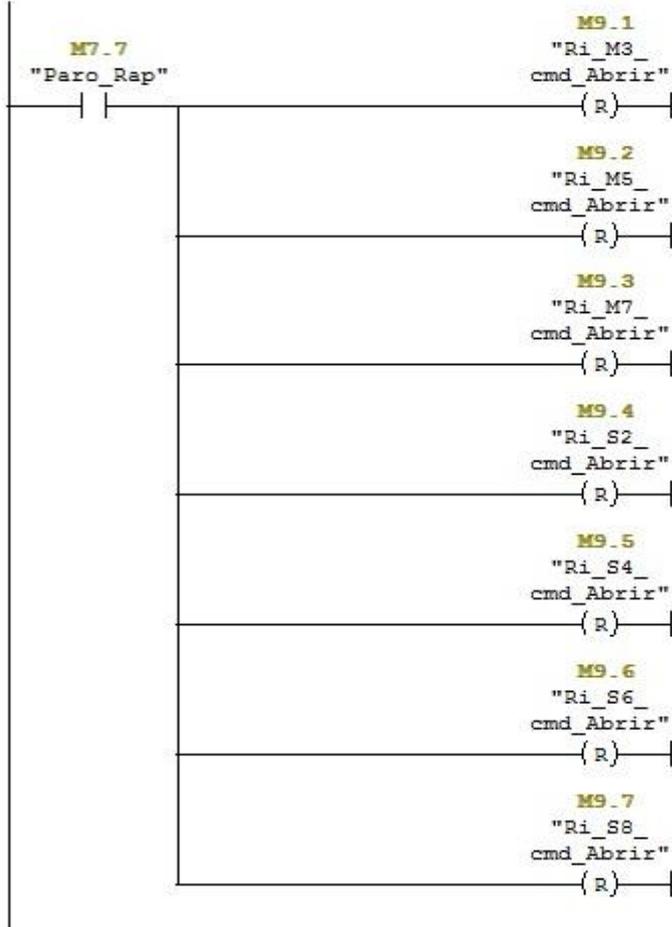
Segm. 20 : Paro Rapido desde Botonera de Campo o desde winCC

Mandar a Resetear los equipos perteniente a la ruta seleccionada



Segm. 21 : Resetear los comandos de la Raceras ante un Paro Rapido

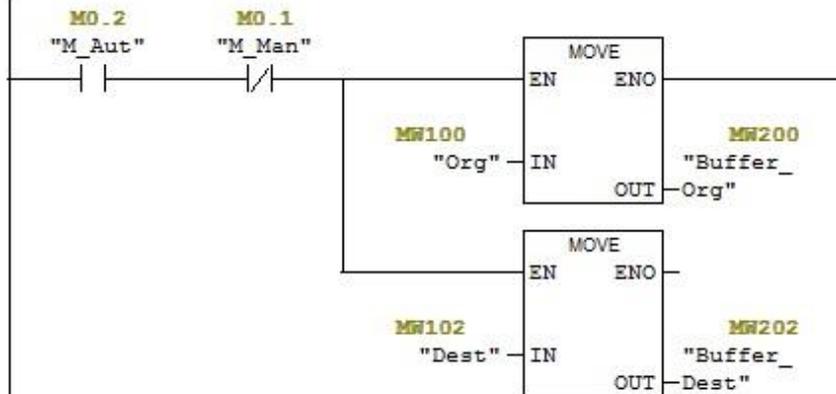
Comentario:



Anexo 13.6: FC2: Selección de ruta

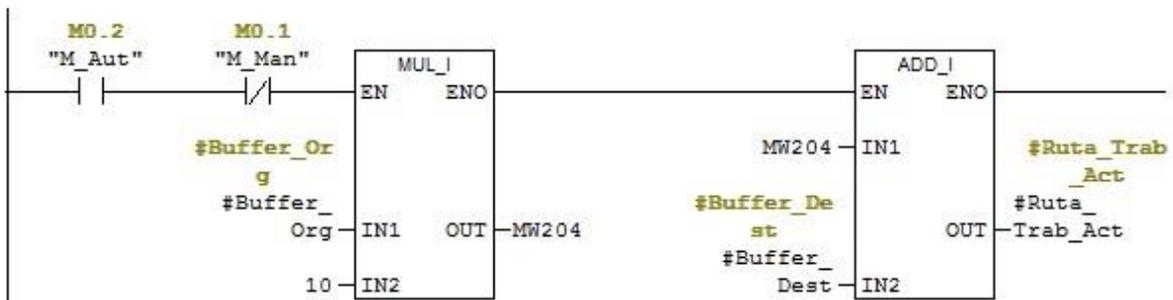
Segm. 1 : Seleccionar la Ruta de trabajo en modo Aut

Comentario:



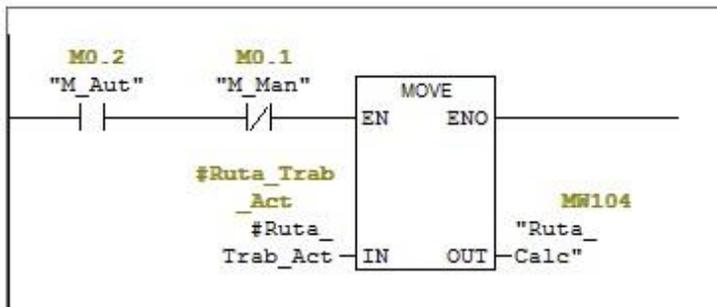
Segm. 2 : Calacular la Ruta Trabajo Seleccionada desde winCC

Org (10)+ Dest. = Ruta Calc. (Org y Dest)



Segm. 3 : Ruta de Trabajo actual = Ruta Calc.

Comentario:



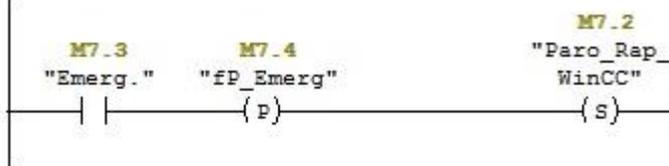
Anexo 13.7: FC3: Activación del estado de Emergencia.

FC3 : Funcion para activar el Paro de Emergencia

Parar el Proceso cuando se efectue el comando de Paro Rapido, ya sea desde WinCC o desde la botonera de Campo

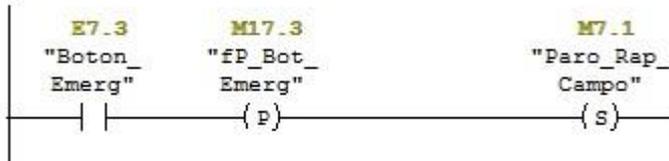
Segm. 1: Parar el Sist cuando exista Emergencia desde el Supervisorio

Parar el Sist cuando exista emergencia desde WinCC



Segm. 2 : Paro inmediato del Proceso cuando hay Emergencia

Parar el Sist cuando exista emergencia desde Botonera de Campo
Un comando de emergencia desde el Campo es un Paro Rap para el Sist.



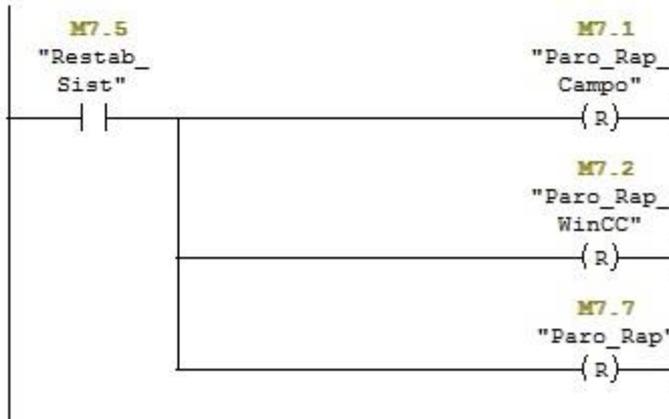
Segm. 3 : Calculo del Paro_Rapido

Se activa la marca de Paro_Rapido una vez que esten activos las señales de Paro_Rap desde el Campo o desde WinCC



Segm. 4 : Restablecer el Sist. a sus condiciones de trabajo

Despues que se atiende el fallo y se confirma que se ha solucionado el problema, es necesario dar la señal Restab_Sist. para resetear la marca de Paro_Rapido



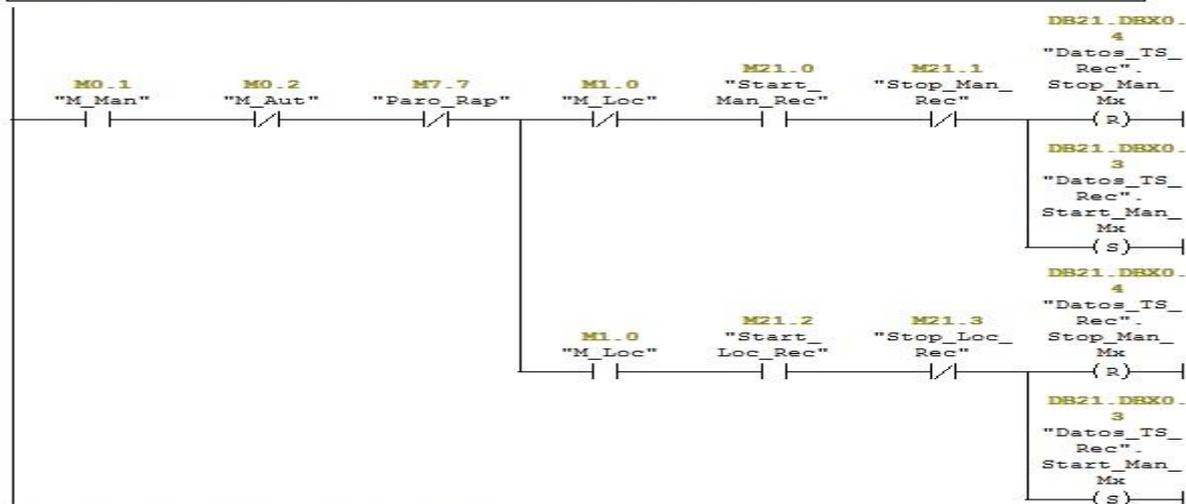
Anexo 13.8: FC21: Control de los motores.

FC21 : Funcion de Ctrl del TS_Rec con llamada al FB_Motores

Comentario:

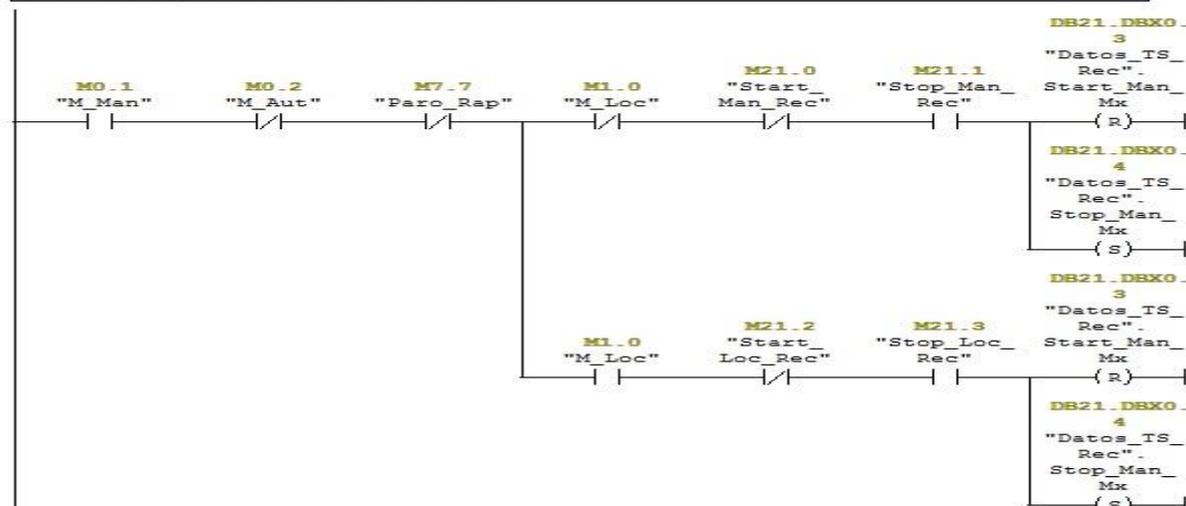
Segm. 1 : Start del Motor en Modo Manual

Habilitar el Mando Man, desde WinCC (cuando no este activo M_Loc) y desde la botonera de Campo (cuando este activo M_Loc)



Segm. 2 : Stop del Motor en Modo Manual

Stop del Motor desde WinCC (M_Loc = 0) o desde botonera de campo (M_Loc = 1)



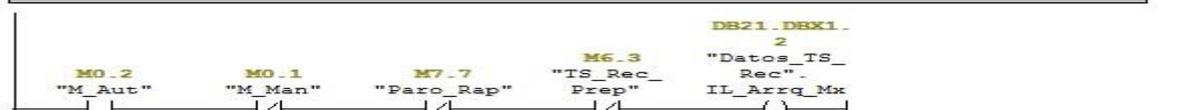
Segm. 3 : En modo Aut calcular el IL_Proc del Motor

Cuando Conf_Run_Prev = 1, es la condicion que activa el IL_Proc



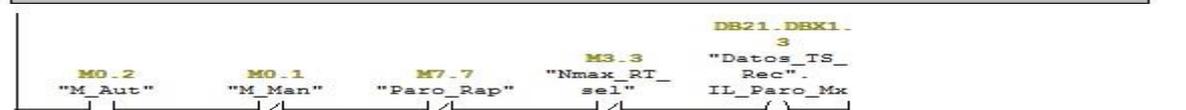
Segm. 4 : En modo Aut calcular el IL_Arrq del Motor

Cuando TS_Rec este preparado (TS_R_Prep), es la cond. que activa el Paro c IL



Segm. 5 : En modo Aut calcular el IL_Paro del Motor

Cuando Nmax_RT = 1, es la condicion que activa el Paro con IL



Anexos

Segm. 6 : Llamada del FB Motor

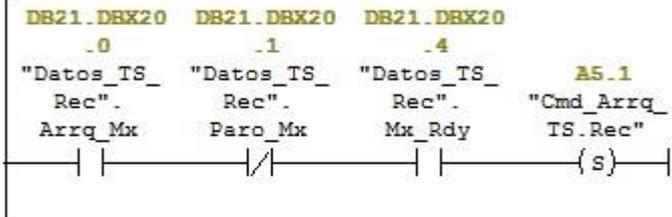
El FB Motor es el bloque funcional reutilizado para controlar el motor

E0.0		DB21		DB21	
"Sist_On"		"Datos_TS_Rec"		"Datos_TS_Rec"	
		"FB_Ctrl_Motores"		"FB_Ctrl_Motores"	
		EN		ENO	
DB21.DBX0.0	"Datos_TS_Rec".M_Man_Mx	M_Man_Mx	Arrq_Mx	DB21.DBX20.0	"Datos_TS_Rec".Arrq_Mx
DB21.DBX0.1	"Datos_TS_Rec".M_Aut_Mx	M_Aut_Mx	Paro_Mx	DB21.DBX20.1	"Datos_TS_Rec".Paro_Mx
DB21.DBX0.2	"Datos_TS_Rec".M_Loc_Mx	M_Loc_Mx	Est_Func_Mx	DB21.DBX20.2	"Datos_TS_Rec".Est_Func_Mx
DB21.DBX0.3	"Datos_TS_Rec".Start_Man_Mx	Start_Man_Mx	Fallo_Mx	DB21.DBX20.3	"Datos_TS_Rec".Fallo_Mx
DB21.DBX0.4	"Datos_TS_Rec".Stop_Man_Mx	Stop_Man_Mx	Mx_Rdy	DB21.DBX20.4	"Datos_TS_Rec".Mx_Rdy
DB21.DBX0.5	"Datos_TS_Rec".Start_Loc_Mx	Start_Loc_Mx	Cons_Mx_pc	DB21.DBD22	"Datos_TS_Rec".Cons_Mx_pc
DB21.DBX0.6	"Datos_TS_Rec".Stop_Loc_Mx	Stop_Loc_Mx	Sobre_i_Mx	DB21.DBX26.0	"Datos_TS_Rec".Sobre_i_Mx
DB21.DBX0.7	"Datos_TS_Rec".Start_Aut_Mx	Start_Aut_Mx	Falla_Conf_Mx	DB21.DBX26.1	"Datos_TS_Rec".Falla_Conf_Mx
DB21.DBX1.0	"Datos_TS_Rec".Stop_Aut_Mx	Stop_Aut_Mx	Aus_Fase_Mx	DB21.DBX26.2	"Datos_TS_Rec".Aus_Fase_Mx
DB21.DBX1.1	"Datos_TS_Rec".IL_Proc_Mx	IL_Proc_Mx	T_Trabajado_Mx	DB21.DBD28	"Datos_TS_Rec".T_Trabajado_Mx
DB21.DBX1.2	"Datos_TS_Rec".IL_Arrq_Mx	IL_Arrq_Mx	MPrev_Mx	DB21.DBX32.0	"Datos_TS_Rec".MPrev_Mx
DB21.DBX1.3	"Datos_TS_Rec".IL_Paro_Mx	IL_Paro_Mx			
DB21.DBX1.4	"Datos_TS_Rec".Mx_Prep	Mx_Prep			
DB21.DBX1.5	"Datos_TS_Rec".Nmax_RT_sel	Nmax_RT_sel			
DB21.DBX1.6	"Datos_TS_Rec".Paro_Rap_Mx	Paro_Rap_Mx			
DB21.DBX1.7	"Datos_TS_Rec".Conf_Run_Mx	Conf_Run_Mx			
DB21.DBX2.0	"Datos_TS_Rec".Conf_Prev_Mx	Conf_Prev_Mx			
DB21.DBX2.1	"Datos_TS_Rec".Alarma_Mx	Ack_Alarma_Mx			
DB21.DBX2.2	"Datos_TS_Rec".Alarma_Mx	Rest_Alarma_Mx			
DB21.DBD4	"Datos_TS_Rec".T_Evac_Mx	T_Evac_Mx			
DB21.DBD6	"Datos_TS_Rec".T_MPrev_Mx	T_MPrev_Mx			
DB21.DBD12	"Datos_TS_Rec".i_Chapa_Mx	i_Chapa_Mx			
DB21.DBD16	"Datos_TS_Rec".i_Cons_Mx	i_Cons_Mx			

Anexos

Segm. 7 : Dar el Comando de Arranque del Motor si este está listo

Activar el cmd_Arrq siempre y cuando el motor este listo para encender



Segm. 8 : Dar el comando de Paro

Cuando se da la orden de Paro se resetear el cmd_Arrq, siempre y cuando el motor este funcionando

