

**Santiago de Cuba
Universidad de Oriente
Facultad de Ingeniería Eléctrica**

LAS REDES DE PETRI COMO HERRAMIENTA FORMAL EN EL DISEÑO DE SISTEMAS DE AUTOMATIZACIÓN INTEGRADA

Tesis en opción al Título de Master en Ciencias.

Autor: Ing. Michel Sanz Pérez.

Tutor: Dr. Israel Benítez Pina.

Consultante: Msc. Juan Carlos Sánchez Perera

**2013
“Año 55 de la Revolución”**

Pensamiento

El camino del conocimiento conduce hacia la victoria; el de la ignorancia, hacia la derrota.

Sun-Tzu

Dedicatoria

A mi familia en general y en especial a mis padres.

Agradecimientos

A mi tutor Dr.C. Israel Benítez Pina por su atención, paciencia y cooperación en el desarrollo de este trabajo.

A mi oponente Dr.C. Reinel Beltrán Aguedo por ser tan crítico y por ayudarme a entender.

A todos los profesores de la Maestría en Automática.

A Saddid, mi compañera de tesis por no rendirse nunca.

A mis viejos amigos que entendieron los desplantes por un bien mayor.

A mis compañeros del Departamento de Control Automático, por ser excelentes profesionales, por alentarme a terminar aun cuando quería estar con ellos en el campismo.

A todas las personas que de alguna forma hicieron posible el desarrollo y culminación de este trabajo.

A todos, Muchas Gracias.

Resumen

En la actualidad el desarrollo vertiginoso de la ciencia y la técnica, en ramas como la electrónica, la computación y más recientemente las comunicaciones, así como el incremento exponencial en tamaño y complejidad de los procesos industriales incrementan considerablemente la probabilidad de ocurrencia de errores en el diseño de los sistemas de automatización integrada.

La presente investigación tiene como objetivo demostrar que el empleo de las redes de Petri (PNs) como método formal de modelado en la etapa de diseño de un sistema de automatización integrada permite a través de la verificación y validación de sus propiedades la obtención de un sistema de automatización con menos probabilidad de ocurrencia de errores, más eficiente, y más seguro.

Para ello se realizó un estudio de las diferentes variantes y posibilidades de las PNs, sus características, propiedades y métodos empleados para su análisis, aplicándolos a través de una metodología en el diseño del sistema de automatización integrada del subproceso Dosificación en la planta de fabricación de bloques tipo POYATOS del complejo productivo “Los Guaos” perteneciente a la Empresa de Materiales de la Construcción de Santiago de Cuba.

Abstract

Nowadays, the vertiginous development of science and technics, at branches like electronics, computation, and more recently communications, as well as the exponential increment in size and complexity of industrial processes, increase considerably the probability of funny remark of errors in the design of the integrated systems of automatization.

Present investigation aims to demonstrate that use of Petri's nets (PNs) like a formal method for modeling in the designing stage of integrated automatization systems permits, through the verification and validation of its properties, the obtaining of a automatization system with less probability of remark of errors, more efficient, and plus insurance.

In order to achieve this goal, a study of different PNs variants and possibilities, its characteristics, properties and methods used for its analysis were accomplished, them applying to crosswise of a methodology in the systems design of integrated automatization for the Dosification sub-process, in the blocks manufacture plant type POYATOS, situated in productive complex "Los Guaos" and belonging to Construction Materials Enterprise of Cuba.

Índice

INTRODUCCIÓN.....	1
CAPÍTULO 1: DISEÑO DE SISTEMAS DE AUTOMATIZACIÓN INTEGRADA.	6
1.1 CARACTERIZACIÓN HISTÓRICA GNOSEOLÓGICA DE LOS SISTEMAS DE AUTOMATIZACIÓN INTEGRADA.....	6
1.1.1 DISEÑO DE AUTOMATIZACIÓN INTEGRADA.....	11
1.2 MÉTODOS FORMALES.	13
1.2 REDES DE PETRI.....	14
1.2.1 TIPOS DE REDES ELEMENTALES.....	15
1.2.2 PROPIEDADES DE LAS REDES DE PETRI.....	16
1.2.3 MÉTODOS DE ANÁLISIS DE PROPIEDADES.	19
1.2.2 MODELADO DE SISTEMAS DE AUTOMATIZACIÓN MEDIANTE REDES DE PETRI.....	20
1.3 MODELADO DE SISTEMAS DE AUTOMATIZACIÓN MEDIANTE LAS REDES GHENESYS.....	23
1.3.1 VERIFICACIÓN SOBRE GHENESYS.	26
1.3.2 VALIDACIÓN SOBRE GHENESYS.....	29
1.3.3 TRADUCCIÓN DE MODELOS GHENESYS A PROGRAMAS IEC61131COMPATIBLES.....	30
CONCLUSIONES PARCIALES.....	38
CAPÍTULO 2: DISEÑO Y PROGRAMACIÓN DEL SISTEMA DE AUTOMATIZACIÓN INTEGRADA DEL SUBPROCESO “DOSIFICACIÓN”.	39
2.1 DESCRIPCIÓN DEL FLUJO TECNOLÓGICO DEL PROCESO DE PRODUCCIÓN DE BLOQUES DE HORMIGÓN EN PLANTAS TIPO “POYATOS”.....	39
2.1.1 SUBPROCESO “DOSIFICACIÓN”.....	41

2.2 DISEÑO DEL SISTEMA DE AUTOMATIZACIÓN INTEGRADA PROPUESTO.....	43
2.2.1 PROPUESTA DE AUTOMATIZACIÓN.....	43
2.2.2 ESTRUCTURA DEL SISTEMA DE AUTOMATIZACIÓN INTEGRADA PROPUESTO.....	46
2.3 FORMALIZACIÓN DEL SISTEMA DE AUTOMATIZACIÓN INTEGRADA PROPUESTO.	48
2.3.1 MODELADO EN GHENESYS DE LA UNIDAD FUNCIONAL “PROGRAMA PRINCIPAL”	50
2.3.2 MODELADO EN GHENESYS DE LA UNIDAD FUNCIONAL “AUTOMÁTICO” PRESENTE EN EL PROGRAMA PRINCIPAL.	53
2.3.3 MODELO EN GHENESYS DE LA UNIDAD FUNCIONAL “DOSIFICACIÓN ÁRIDOS” PRESENTE EN EL MACRO LUGAR “AUTOMÁTICO ON”.	56
2.4 VERIFICACIÓN.....	59
2.4.1 VERIFICACIÓN POR EL MÉTODO DE TÉCNICAS DE REDUCCIÓN O DESCOMPOSICIÓN.	59
2.4.1.1 Reducción de la PN principal correspondiente al modelado no controlado del Programa Principal.....	60
2.4.1.2 Reducción de la PN principal correspondiente al modelado del macrolugar Automático.	60
2.4.1.3 Reducción de la red correspondiente al macro lugar Dosificación Áridos.....	61
2.4.2 VERIFICACIÓN APLICANDO EL ENFOQUE DE LA MATRIZ DE INCIDENCIA.....	62
2.4.2.1 Cálculo de invariantes para la red correspondiente al programa Principal.	62
2.4.2.2 Cálculo de invariantes para la red correspondiente al macro lugar Automático...	64
2.4.2.3 Cálculo de invariantes para la red correspondiente al macro lugar Dosificación de Áridos.....	66
2.5 VALIDACIÓN.	68
2.6 TRADUCCIÓN Y PROGRAMACIÓN.....	68
2.6.1 PROGRAMACIÓN DEL MÓDULO PROGRAMA PRINCIPAL.	69
2.6.2 PROGRAMACIÓN DEL MÓDULO AUTOMÁTICO.....	70
2.6.3 PROGRAMACIÓN DEL MÓDULO DOSIFICACIÓN DE ÁRIDOS.....	71
2.7 DISEÑO DEL SUPERVISORIO.	71

2.8 VERIFICACIÓN Y VALIDACIÓN A NIVEL DE LABORATORIO.....	72
CONCLUSIONES PARCIALES.....	73
CONCLUSIONES GENERALES	74
RECOMENDACIONES	75
BIBLIOGRAFÍA.....	76
ANEXOS	

Introducción

El control automático es parte vital del progreso industrial desarrollado durante lo que ahora se conoce como la segunda revolución industrial. El uso intensivo de la ciencia del control automático es producto de una evolución como consecuencia del uso difundido de las técnicas de medición y control. Su estudio intensivo ha contribuido al reconocimiento universal de sus ventajas.

Del mismo modo la automatización como una disciplina de la ingeniería es más amplia que un mero sistema de control, abarca desde la instrumentación industrial, que incluye los sensores y transmisores de campo, hasta los sistemas de control y supervisión, los sistemas de transmisión y recolección de datos, y las aplicaciones de software en tiempo real para supervisar y controlar las operaciones de plantas o procesos industriales.

En sus inicios los sistemas de automatización eran sistemas de control cerrados, buenos en el control individual de los procesos y basados mayormente en circuitos eléctricos con relés que utilizaban lógica cableada y en menor medida por componentes neumáticos. Esta tecnología presentaba limitaciones importantes como son: uso ineficiente del espacio, falta de flexibilidad, aplicabilidad limitada a problemas sencillos, tiempos de localización y reparación de averías grandes, entre otras [43].

En 1960, la industria comenzó a buscar en las nuevas tecnologías electrónicas una solución más eficiente, pero no fue hasta finales de la década de los sesenta e inicio de los setenta que aparecieron los Controladores Lógicos Programables (PLC) [4, 14]. Dispositivos que ganaron mucha popularidad en la industria debido principalmente a las ventajas que ofrecían, como son: adecuado costo para controlar sistemas complejos, flexibilidad y reusabilidad para controlar otros sistemas de manera rápida, capacidades computacionales que permiten un control más sofisticado, ayudas en la localización de problemas que hacen la programación más fácil y reducen el período de inactividad por fallas, componentes fiables que los hacen propensos a funcionar por años antes fallar [33, 44].

Desde su surgimiento hasta nuestros días los PLCs han sido los dispositivos mayormente utilizados en la solución de problemas de automatización y control [25, 38]. Su uso se extiende a soluciones muy diversas, que van desde máquinas herramientas hasta plantas industriales, desde juguetes hasta el confort de hoteles y edificios, desde elevadores hasta sistemas telefónicos, entre otros, utilizándose en muchas ocasiones en sistemas de seguridad crítica, como es el caso de sistemas de control de tráfico, plantas químicas, monitoreo de pacientes, y sistemas de generación de electricidad [22, 39].

Dentro de los objetivos que persigue la automatización están:

- Mejora de la productividad de la empresa, reduciendo costes de la producción y mejorando la calidad de la misma.
- Mejorar las condiciones de trabajo del personal, suprimiendo los trabajos penosos e incrementando la seguridad.
- Realizar las operaciones imposibles de controlar intelectual o manualmente.
- Mejorar la disponibilidad de los productos, pudiendo proveer las cantidades necesarias en el momento preciso.
- Simplificar el mantenimiento de forma que el operario no requiera grandes conocimientos para la manipulación del proceso productivo.
- Integrar los niveles de gestión y producción.

Sin embargo las malas prácticas y la persistencia en la realización empírica del diseño de sistemas de automatización integrada de acuerdo a la experiencia del ingeniero conllevan a la posibilidad de cometer errores que luego se ven reflejados en las fases subsiguientes de programación e implementación.

El desarrollo vertiginoso de la ciencia y la técnica, en ramas como la electrónica, la computación y más recientemente las comunicaciones, así como el incremento exponencial en tamaño y complejidad de los procesos industriales es un fenómeno inevitable que, unido a las exigencias de la industria moderna en cuanto a flexibilidad, tiempos productivos, competitividad e integración, incrementan considerablemente la probabilidad de ocurrencia de estos errores. Más aún, algunos de estos errores pueden conllevar a pérdidas catastróficas de dinero, tiempo, o incluso la vida de un ser humano [16, 24].

Es por eso que la comunidad científica ha venido incrementando desde los últimos años del siglo pasado su labor investigativa en el área de los métodos formales, y dentro de estos, en mayor medida las redes de Petri (PNs) en muchas de sus extensiones, como paradigmas que permiten la especificación, modelado, verificación y validación de propiedades y requisitos, así como en el desarrollo de herramientas que las sustenten de manera que se reconozca a través de resultados concretos la importancia, y al mismo tiempo, la necesidad por parte de los ingenieros de incorporarlos en el proceso de diseño de sistemas complejos con el principal objetivo de desarrollar automatismos fiables independientemente de su complejidad.

Aunque el uso de métodos formales a priori no garantiza un resultado cien por ciento correcto, puede sin embargo, incrementar grandemente nuestra comprensión del sistema revelando incongruencias, ambigüedades e imprecisiones que de otra manera podrían pasar desapercibidas [16].

La automatización moderna abarca todas las esferas de la vida actual de forma inevitable, siempre que se desee marchar lo más cercano posible al promedio del desarrollo técnico-económico mundial. Aparte de esto, en la actualidad, la automatización es casi imprescindible en áreas tan disímiles como el turismo, la agricultura, la industria, los servicios, etc. Por tanto, es indiscutible que la eficiencia y productividad industrial sólo se puede lograr con una buena automatización.

Por lo anteriormente expuesto queda definido como **problema de la investigación** la necesidad del empleo de métodos formales en el diseño de sistemas de automatización integrada con el objetivo de disminuir la probabilidad de ocurrencia de errores durante la etapa de diseño, posibilitando la obtención de una automatización más eficiente y segura, con el consecuente ahorro de tiempo que esto implica.

Constituyéndose en **objeto de la investigación** los sistemas de automatización integrada.

El **objetivo** radica en demostrar que el empleo de las PNs como método formal en la etapa de diseño de un sistema de automatización integrada (subproceso Dosificación en la planta de fabricación de bloques tipo POYATOS del complejo productivo “Los Guaos” perteneciente a la Empresa de Materiales de la Construcción de Santiago de Cuba) permite a través de una

mejor comprensión del mismo la obtención de un sistema de automatización con menos probabilidad de ocurrencia de errores, más eficiente y más seguro.

Precisando el **campo de acción** la aplicación de las PNs como método formal en el diseño de sistemas de automatización integrada.

Se propone entonces la siguiente **hipótesis**: si se demuestra que el empleo de las PNs como método formal en la etapa de diseño de un sistema de automatización integrada (subproceso Dosificación de Áridos en la planta de fabricación de bloques tipo POYATOS del complejo productivo “Los Guaos” perteneciente a la Empresa de Materiales de la Construcción de Santiago de Cuba) permite disminuir la posibilidad de ocurrencia de errores y así obtener un sistema de automatización más eficiente y seguro se contribuirá a través del conocimiento a resolver el problema del aumento en la probabilidad de ocurrencia errores al no hacer uso de métodos formales en la etapa de diseño de sistemas de automatización integrada.

El aporte de la investigación es la demostración de la aplicabilidad de las PNs como método formal en el diseño de sistemas automatización integrada con las ventajas que esto conlleva.

Tareas de la investigación:

1. Caracterizar histórica y gnoseológicamente los sistemas de automatización industrial integrada.
2. Revisar en la bibliografía publicada por la comunidad científica cuáles son los métodos formales más empleados en el diseño de sistemas de automatización industrial integrada.
3. Analizar el paradigma de las PNs como método formal en el diseño de sistemas de automatización.
4. Revisar y profundizar en el estudio del flujo tecnológico de la fábrica de bloques tipo POYATO, en especial del proceso de dosificación de áridos.
5. Elaborar el diseño a partir de los requerimientos funcionales de sistema automatizado.
6. Programar la automática local y el supervisorio.
7. Evaluar los resultados.
8. Redacción final del Informe.

En el transcurso de la investigación se utilizaron las siguientes **técnicas y métodos**:

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

Significación práctica de la investigación:

En el presente trabajo, la utilización de las PNs como método formal en la etapa de diseño demuestra las capacidades de estas como método formal de modelado y permite la obtención de un diseño de automatización más eficiente y con una posibilidad mínima de errores en el subproceso de Dosificación de la planta de fabricación de bloques.

Estructura y organización del Informe:

El informe está estructurado en: Introducción general, dos capítulos con introducción y conclusiones parciales cada uno, conclusiones generales, recomendaciones, bibliografía y anexos.

En el Capítulo 1 es donde se presenta el marco teórico y estado del arte fundamental que relaciona los diferentes aspectos a tener en cuenta en la aplicación de las redes PNs como herramienta formal de modelado, verificación y validación en las etapas de diseño y programación de sistemas de automatización integrada.

En el Capítulo 2 se diseña el sistema de automatización integrada del proceso caso de estudio (subproceso de Dosificación de la planta de producción de bloques) realizando primeramente un estudio del flujo productivo general de la planta y un estudio de las funciones específicas del proceso de Dosificación dentro del mismo. Determinando los requerimientos funcionales y la estructura del sistema de automatización a emplear y realizando con la ayuda de las redes de PNs el modelado, verificación y validación de las principales propiedades y requisitos funcionales más importantes para la posterior programación del sistema en su totalidad.

Capítulo 1: Diseño de sistemas de automatización integrada.

Hoy en día, el abordaje de la mayor parte de los problemas de ingeniería modernos es casi inconcebible sin la utilización de técnicas de modelado y simulación. Debido a cuestiones de riesgo y costo, la experimentación directa sobre sistemas reales está dejando su lugar a la experimentación sobre modelos de simulación.

La temática de la eficiencia de los sistemas de automatización integrada a partir de su modelado está siendo abordada por varios especialistas e instituciones de investigación principalmente del primer mundo a nivel internacional [45, 37, 48, 58, 22, 35, 36, 57, 59]. Esto se debe a la importancia que reviste la posibilidad de validar los modelos formalmente a partir de los requerimientos que motivaron su diseño antes de pasar a la implementación de los mismos con el consiguiente ahorro en tiempo y recursos que esto representa.

1.1 Caracterización histórica gnoseológica de los sistemas de Automatización integrada.

El término automatización industrial se refiere a la aplicación de la automatización como ciencia al ámbito industrial, a su vez, la automatización como disciplina de la ingeniería abarca desde la instrumentación industrial, que incluye los sensores y transmisores de campo, hasta los sistemas de control y supervisión, los sistemas de transmisión y recolección de datos, y las aplicaciones de software en tiempo real para supervisar y controlar las operaciones de plantas o procesos industriales. Es decir, la automatización constituye de manera general el sistema mediante el cual se transfieren tareas de producción, realizadas habitualmente por operadores humanos a un conjunto de elementos tecnológicos [43].

Los sistemas de automatización han sido caracterizados desde sus inicios por poseer 3 niveles básicos cómo se puede observar en la figura 1.1.

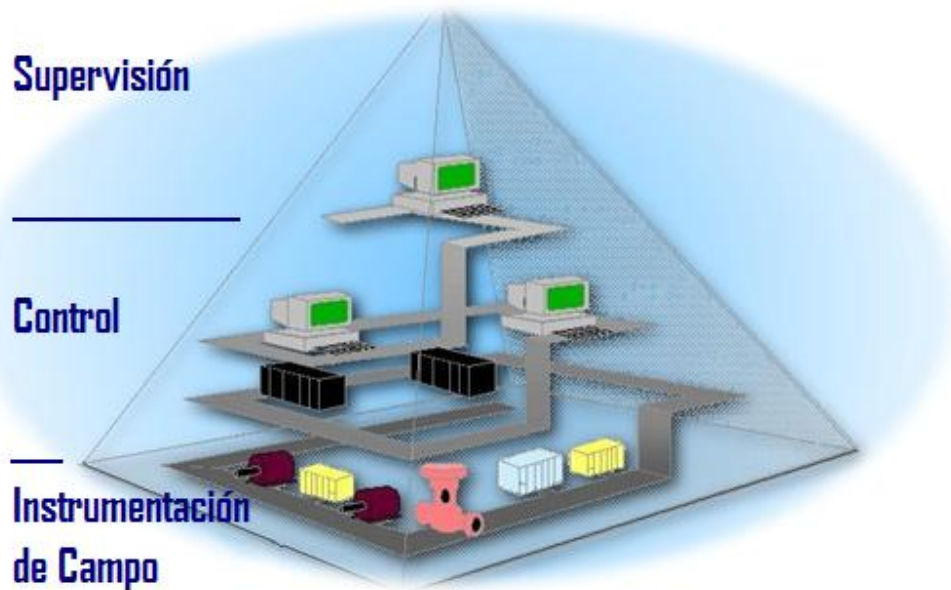


Figura 1.1 Pirámide de automatización.

- **Un nivel de supervisión:** donde las funciones principales están orientadas a la supervisión del funcionamiento del sistema en su totalidad permitiendo la integración de varios sistemas de automatización de nivel local y a partir de las variables históricas más importantes tener: planificación del mantenimiento de los equipos que así lo requieran, estadísticas de la producción que sirvan en la toma de decisiones, atención a alarmas, etc.
- **Un nivel de control local:** cuyas funciones principales están orientadas al control de la producción en las diferentes máquinas o procesos, al desarrollo de secuencias de operación, a la estabilización de los parámetros principales del proceso, a la atención a fallas y a la operación local del sistema.
- **Un nivel de instrumentación de campo:** en el que se encuentra la instrumentación ya sea descentralizada o no, inteligente o no, de medición o control.

Es con este nivel de supervisión que surge la palabra automatización integrada pero que no se hace popular hasta nuestros días.

En esta última década el incremento acelerado de las exigencias en flexibilidad y reconfiguración [13], así como la explosiva proliferación de las nuevas tecnologías y la

comunicación ha provocado que surjan conceptos nuevos en la automatización, como por ejemplo [54]:

- **Automatización Industrial Integrada:** una estrategia en cuyo centro está la tecnología que permite una información transparente entre el cliente, las operaciones de fabricación y los suministradores.
- **Manufactura Colaborativa o e-manufactura:** una estrategia que mezcla los procesos de negocios y las estrategias de fabricación creando una vía óptima para el sistema de desarrollo, eficiencia y productividad en la empresa.
- **Sistemas integrados de Producción:** un sistema empresarial totalmente integrado que brinda una sincronización perfecta entre todas las actividades funcionales en todos los campos de fabricación y los negocios usando tecnologías de la información y las comunicaciones.

Todo parece indicar una tendencia creciente en la integración vertical a niveles de dirección en los que mientras algunos buscan solamente potenciar los flujos de información y control de materiales de una empresa enlazando todas sus funciones y entidades funcionales (sistemas de información, dispositivos, aplicaciones, operadores, etc.) para incrementar la comunicación, cooperación y coordinación consiguiendo un comportamiento integrado, con más productividad, flexibilidad y capacidad en las gestiones empresariales otros buscan también con la integración aprovechar las coyunturas del mercado internacional garantizando mínimo almacenamiento y subcontrataciones internacionales haciendo una producción industrial cooperativa entre varias plantas que pueden estar situadas en diversos lugares del mundo.

De esta tendencia existen ya varios gigantes de la automatización en diferentes partes del mundo que se han visto influenciados y así lo reflejan en sus proyecciones, por ejemplo:

- Siemens A&D en Europa presentó en la feria internacional de Hanover 2000 un conjunto tecnología industrial para un Sistema de Integración vertical y horizontal de toda su empresa (figura 1.2).

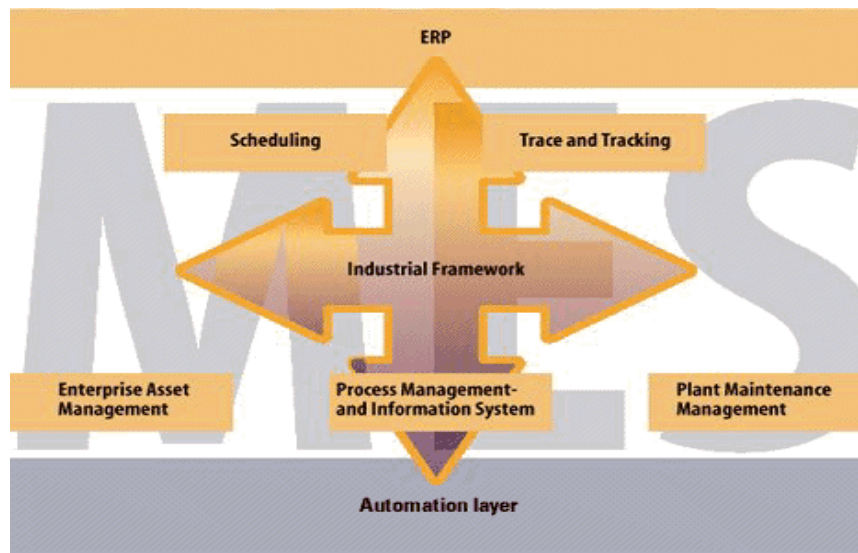


Figura 1.2 Sistema de Automatización Integrada Siemens.

- Yokogawa en Asia desarrolló un nuevo sistema de automatización integrada llamado Integrated Production Control Centum CS3000 R3, que es un sistema integrado de manufactura en tiempo real que utiliza el estándar internacional de comunicaciones industriales OPC entre los tres niveles de automatización (Bajo, Medio y Alto) en el MES (Manufacturing Execution System) (figura 1.3).

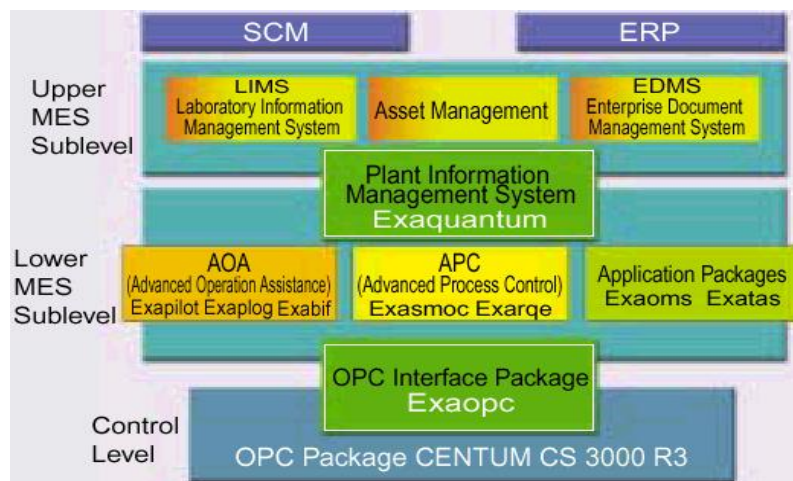


Figura 1.3 Sistema de Automatización Integrada Yokogawa.

- Factory Wonderware en América incluye sistemas de desarrollo para los programas InTouch, SCOUT Family y SQL Industrial y los módulos SPC (Statistical Process Control) y Recipe Manager (Administrador de Recetas) (figura 1.4).

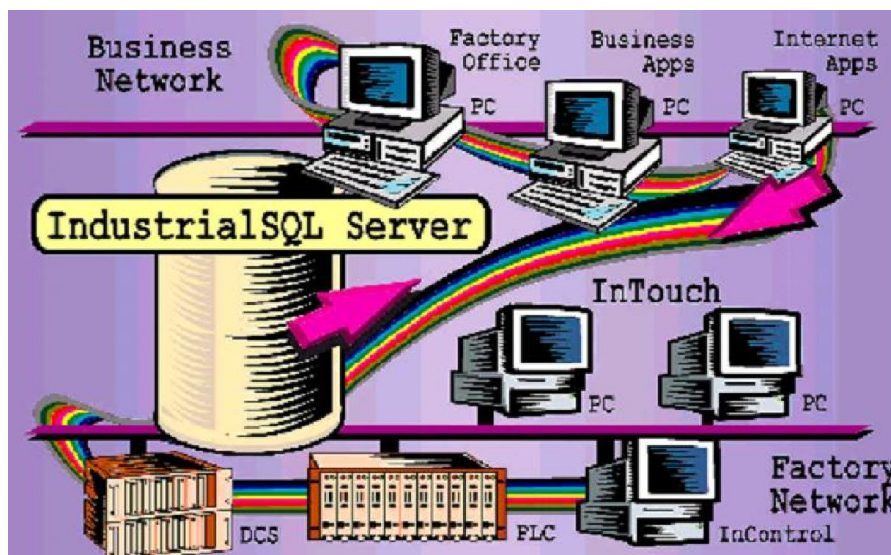


Figura 1.4 Sistema de Automatización Integrada WonderWare.

La integración de la automatización se logra fundamentalmente en el nivel superior de la pirámide de automatización, dominada por los sistemas de supervisión industrial.

Los sistemas de supervisión tradicionalmente han sido denominados sistemas SCADA (Supervisory Control and Data Acquisition) o SCD (Distributed Control Systems), pero este último término ha sido prácticamente absorbido por el primero a nivel industrial.

El término SCADA es un sistema basado en computadores que permite supervisar y controlar variables de proceso a distancia, proporcionando comunicación con los dispositivos de campo (controladores autónomos), y controlando el proceso de forma automática por medio de un software especializado [53, 50, 9]. También provee de toda la información que se genera en el proceso productivo a diversos usuarios, tanto del mismo nivel como de otros usuarios supervisores dentro de la empresa (supervisión, control calidad, control de producción, almacenamiento de datos, entre otros, que contribuyen a la toma de decisiones) [3].

Un sistema SCADA incluye un hardware de señales de entrada y salida, controladores, interfaz hombre-máquina (HMI), redes, comunicaciones, base de datos y software. En este tipo de sistemas usualmente existen ordenadores, que efectúan tareas de supervisión y gestión de alarmas, así como tratamiento de datos y control de procesos. La comunicación

se realiza mediante buses especiales o redes LAN. Todo esto se ejecuta normalmente en tiempo real, y están diseñados para dar al operador de planta la posibilidad de supervisar y controlar dichos procesos [53].

Entre las principales funciones de los SCADA se encuentra la supervisión remota, que permite conocer el estado de las instalaciones y coordinar eficientemente las labores de producción y mantenimiento en el campo. Mediante el control remoto de instalaciones se activan o desactivan equipos remotos de manera automática o a solicitud del operador. Igualmente es posible realizar ajustes en parámetros en lazos de control analógicos (punto de consigna). Otra de las funciones más utilizadas es la presentación de alarmas, mediante la cual se alerta al operador sobre la ocurrencia de condiciones anormales o eventos que pudieran requerir su intervención. Normalmente, la criticidad del evento o alarma se indica mediante el uso de colores y/o señales auditivas. Las alarmas se registran para análisis posteriores [50, 32, 3].

Dentro de los SCADAs más conocidos patentados por diferentes empresas se encuentran WinCC de Siemens, Vijeo Citect de Schneider, CX-One Supervisor de Omron entre otros. En nuestro país se desarrollan algunos SCADA nacionales, de los cuáles el más completo y divulgado en el ámbito industrial es el **Sistema de Supervisión y Control de Procesos Eros**, destacándose por la facilidad con que puede ser operado y configurado (en caliente) ofreciendo funcionalidades predeterminadas (estadísticas, alarmas, recetas, datos históricos) lo que disminuye los costos de puesta en marcha y con una experiencia acumulada durante más de 10 años por el uso en más de 180 instalaciones en diferentes procesos en todo el país y en el extranjero [30].

1.1.1 Diseño de automatización integrada.

La automatización de sistemas industriales tiene una fuerte componente definida por la coordinación de las actividades de los diferentes subsistemas que se integran en las redes de control distribuido industrial que se utilizan actualmente.

En la figura siguiente figura 1.5 es presentado el **ciclo de diseño (proyección) de la Automatización Industrial**. El recorrido exterior fue usado durante mucho tiempo por los

proyectistas de sistemas automáticos industriales, pero con el incremento de la complejidad de la automatización integrada es necesario utilizar el recorrido interno para garantizar los objetivos indicados en las etapas anteriormente presentadas.

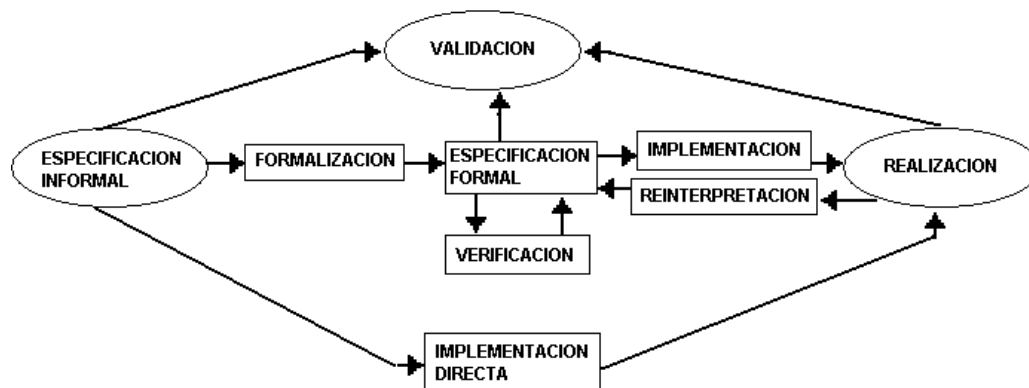


Figura 1.5 Ciclo de diseño de la Automatización Industrial [23].

Las etapas de desarrollo de cualquier automatización moderna incluyen [4]:

1. **Diseño:** Donde se requiere hacer el análisis y síntesis del sistema automatizado. Esto es solo posible con el uso del Modelado Formal de la planta industrial, sus especificaciones de seguridad y comportamiento como sistema controlado.
2. **Programación:** Los modelos deben ser programados en lenguajes normalizados de automatización con facilidades de simulación de su funcionamiento. Como los PLCs son los dispositivos de mayor uso [34], la IEC61131 es la norma más usada, así como sus ambientes de edición y simulación.
3. **Implementación:** En la automatización integrada actual se requiere crear estructuras jerárquicas en redes buscando eficiencia y facilidades de ampliación, modificación y mantenimiento para reducir tiempo y costos en el proyecto y luego en el funcionamiento de la empresa.

Este trabajo se centra en la primera etapa como base para el desarrollo de la segunda y para la versión de laboratorio de la tercera. Por esto se debe profundizar en el estudio de los métodos formales de diseño.

Si la parte de hardware correspondiente al diseño está funcionalmente bien definida, y garantiza la operación correcta del controlador, entonces el funcionamiento del sistema de automatización estará determinado por el software asociado a este, es decir, por la programación de la aplicación. En [52, 22] se especifica que un 83% del costo de los proyectos de automatización se centra en el desarrollo, verificación y validación del software. Para esto son fundamentales los métodos formales.

1.2 Métodos formales.

Los métodos formales de diseño de sistemas de automatización [25] utilizan la teoría de diseño de SISTEMAS A EVENTOS DISCRETOS (SED) para su modelado, verificación y validación formal [2]. Estos sistemas se caracterizan por representar el comportamiento dinámico de los sistemas a través de los cambios de estado generados por la ocurrencia de eventos. De aquí que se pueda definir que están compuestos por dos elementos principales:

- ESTADO representado por conjuntos discretos. Por ejemplo: equipamiento ocupado, ocioso, en mantenimiento; nivel de estoque igual a N; mensajes en 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 E de EVENTOS: llegada de pieza, inicio de operación, temperatura llega un límite dado, etc.

Existen dos clases principales de modelos formales para SEDs:

1. **Modelos Lógicos:** Orientados para el análisis del comportamiento lógico (secuencial) del sistema.
 - Redes de Petri (PNs)
 - Autómatas finitos
2. **Modelos de Comportamiento:** Orientados para el análisis de desempeño (cuantitativo) del sistema.
 - Modelos Analíticos: Cadenas de Markov, Redes de Filas, etc.
 - Modelos p/ Simulación: Arena, Automod, Promodel, etc.

Las PNs constituyen la técnica de mayor popularidad en el diseño de controladores, tanto para el campo de la computación como para los sistemas de automatización industrial, por garantizar una representación gráfica sencilla, así como tener un formalismo matemático bien fundamentado [17, 18, 25].

Las PNs tienen una serie de ventajas sobre otros métodos. Primeramente, por permitir una descripción compacta y segundo, por poder representar de forma gráfica las especificaciones y la planta objeto de análisis, sin requerir de notaciones matemáticas o descripciones textuales ambiguas, las que pudieran dificultar su comprensión. Y finalmente, porque los modelos sobre PNs pueden ser utilizados tanto para el análisis de las propiedades comportamentales y la evaluación de su realización, como para la construcción sistemática del controlador [11, 6].

1.2 Redes de Petri

Red de Petri (PN) es un **término agrupador**, que en el transcurso del tiempo ha venido a designar un amplio número de modelos de sistemas, procedimientos, técnicas y patrones descriptivos relacionados unos con otros en el sentido de que están todos basados en el mismo principio específico (Teoría de las Redes de Petri) [17, 41].

El **principal atractivo** radica, como se menciona anteriormente, en que permiten la identificación de los aspectos básicos de los sistemas distribuidos tanto conceptual como matemáticamente. Tienen la dualidad de ser una herramienta de modelado matemática y gráfica aplicable a muchos sistemas, lo cual incluye además, el atractivo de los sistemas visuales de simulación gráfica.

La **notación gráfica de cualquier PN** responde a un **gráfico bipartito, direccionado, ordenado y no vacío sin nodos aislados**, donde existen dos conjuntos básicos (los conjuntos de estados (Lugares) y transiciones) asociados a **círculos** y a **cajas o cuadrados** respectivamente, siendo sus interconexiones las relaciones de flujo llamados **arcos con direcciones apropiadas**. A esta estructura estática se adiciona el **marcaje representado por los puntos (tokens o marcas)** en los estados. Esto permite representar la evolución del comportamiento dinámico de la red conocido como reglas de disparo de las transiciones [41].

1.2.1 Tipos de redes elementales.

Existen tres grandes grupos de PNs: las **redes clásicas**, que incluyen los Sistemas Condición/Evento (C/E-systems), las Redes Lugar/Transición (P/T-nets) y **las extensiones derivadas de ellas**, y las **redes de alto nivel**, que abarcan fundamentalmente las Redes Predicado/Evento (P/E-nets), Redes coloridas y Redes relacionadas. Estos grupos se diferencian fundamentalmente por la forma en que son marcadas las redes (cantidad de marcas o *tokens*) y por la adición de elementos auxiliares.

En los **sistema C/E** sus lugares son simplemente marcados o no marcados (sólo es posible un *token*).

En el modelo de **sistema P/T-nets**, los lugares pueden contener un cierto número de *tokens* sin diferenciación. Esto agrega el análisis de la capacidad de cada lugar (**K**) y el peso asociado a los arcos (**W**).

Las **extensiones a las redes P/T** está dadas por la adición de arcos habilitadores e inhibidores, temporización y otros elementos.

En las **redes de alto nivel**, los lugares son marcados por diferentes tipos de tokens.

Las aplicaciones de automatización con PLC's abarcan principalmente el sector de las Redes P/T extendidas. Un ejemplo clásico es el modelado de la compartición de recursos, como el uso compartido de los sistemas de comunicación, de áreas de almacenaje para diferentes líneas de producción, etc.

Existen tres reglas de disparo de las transiciones en las P/T-nets:

1. Sensibilización (Habilitación) de transición: donde todos los lugares de entrada de la transición deben contener un número de fichas (tokens) mayor o igual al peso del arco correspondiente.

2. Ocurrencia del evento asociado con la transición: usualmente el tiempo en las redes extendidas.

3. Disparo de transición: provoca el cambio en el marcaje de la red de M para M', retirando de cada lugar de entrada de t, un número de fichas (tokens) igual al peso del arco de

entrada, y coloca en cada lugar de salida de t , un número de fichas (tokens) igual al peso del arco de salida.

Las extensiones y modificaciones adecuan el modelo a condiciones específicas de determinadas aplicaciones. Estos son los casos de las Redes Temporalizadas [1, 41], Redes Interpretadas [17, 23, 24], Redes Extendidas [41, 49], y otras más.

La **representación matemática de una PN** viene dada por la definición de la **Matriz de incidencia** que representa la estructura de la red y puede ser descrita de la forma siguiente:

$$A = [a_{ij}] \text{ mxn} \quad \text{con} \quad a_{ij} = a_{ij}^+ - a_{ij}^-$$

donde:

$a_{ij}^+ = W(j,i)$ es el peso del arco que va de la transición j al lugar i .

$a_{ij}^- = W(i,j)$ es el peso del arco que va del lugar i a la transición j .

m - es el número de lugares y n - el número de transiciones.

Ecuación de estado en una red está dada por:

$$M_{k+1} = M_k + A \cdot v_k$$

donde M_k y M_{k+1} son los estados actual y siguiente respectivamente y v_k es el vector de habilitación actual.

1.2.2 Propiedades de las redes de Petri.

En el análisis y diseño de modelos basados en PN's se tienen en cuenta un conjunto de propiedades [41], pudiendo clasificarse éstas como:

- **Propiedades funcionales:** Son aquellas propiedades dependientes del marcaje inicial y reflejan el comportamiento dinámico del sistema.
- **Propiedades estructurales:** Son aquellas propiedades independientes del marcaje inicial de la red y por tanto inherentes a la estructura de la misma.

Propiedades funcionales:

- Alcanzabilidad (*Reachability*)
- Limitación (*Boundedness*)
- Vivacidad (*Liveness*)
- Reversibilidad y Estado particular (*Reversibility and Home State*)
- Cobertura (*Coverability*)
- Persistencia (*Persistence*)
- Distancia sincrónica (*Synchronous distance*)
- Disparabilidad (*Fairness*)

Propiedades estructurales:

- Vivacidad estructural (*Structural Liveness*)
- Controlabilidad (*Controllability*)
- Limitación estructural (*Structural Boundedness*)
- Conservabilidad (*Conservativeness*)
- Repetitividad (*Repetitiveness*)
- Consistencia (*Consistency*)
- Invariantes S y T (*S- and T-Invariants*)
- Disparabilidad limitada estructural (*Structural B-Fairness*)

La definición de cada una de estas propiedades se sale de los marcos de este trabajo y pueden ser encontradas en varios textos básicos sobre PN's como el caso de [41]. Por tanto, se trabajará sólo con algunas de ellas por su importancia en el modelado de la automática.

En el análisis de modelos PN de sistemas de automatización es muy importante la propiedad de **Alcanzabilidad** de estados, pues a partir de un estado inicial muchas veces se requiere que la red avance automáticamente hasta un estado dado, si no tiene esta capacidad, no podrá ejecutar el comportamiento deseado. A esto está relacionada la **Vivacidad** de la red, 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, bloqueos, *deadlock*), lo

cual nos permite 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 estructural o funcional de la red**.

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.

Dentro de las propiedades estructurales, las **Invariantes S y T** son importantes medios para el análisis de las redes de Petri, debido a que permiten investigar la estructura de la red, independientemente de los procesos dinámicos.

Las invariantes son **características algebraicas de las redes de Petri** y son usadas en varias situaciones, como para **verificar limitación (boundedness), periodicidad (periodicity) y otras propiedades estructurales**. Su definición formal es la siguiente:

Invariante S para las redes FC: Una solución entera X del sistema homogéneo $AX=0$, donde A es la matriz de incidencia, donde el conjunto de lugares de dicha P/T-net no tienen cambios del total de tokens durante los disparos de las transiciones. Estas invariantes se representan por un **vector n-columna x** , donde n es el número de lugares de la red de Petri. Las **entradas del vector x que no son cero, son los lugares invariantes**.

Invariante T para las redes FC: Una solución entera X del sistema homogéneo $A^t X=0$, donde A^t es la transpuesta de la matriz de incidencia. Indican cómo, comenzando desde alguna marcación M , cada transición tiene que disparar (exactamente $v(t)$ veces), para reproducir esta marcación. Estas invariantes son representadas por un **vector m-columna y** , donde m es el número de transiciones de la red. Este vector **y almacena un número entero en las posiciones que pertenecen a las transiciones invariantes**, y almacena el valor cero en los otros casos. Los números enteros representan la cantidad de veces que la transición correspondiente puede ser disparada para que la red vuelva a su estado inicial.

1.2.3 Métodos de análisis de propiedades.

Los métodos de análisis de las PN's [41] se pueden clasificar en tres grupos:

- Método del árbol de cobertura o de Alcanzabilidad (*Coverability or Reachability Tree Method*).
- Método analítico o enfoque por la ecuación matricial (*Matrix-Equation Approach*).
- Método de técnicas de reducción o descomposición.

El primer método incluye la enumeración de todos los marcajes alcanzables. Se puede aplicar a todas las redes, pero está limitado a redes “pequeñas” por su complejidad debido a la explosión de estados para sistemas grandes (inmenso número de lugares).

Los otros dos métodos son prácticos y potentes, pero en muchos casos, solo son aplicables a subclases especiales de PN's o situaciones especiales (tiene limitaciones). Para el análisis se deben asumir, en primer lugar, sólo PN's puras (sin lazos infinitos) o transformadas a puras (agregando un par lugar-transición a cada lazo infinito) [41] para admitir la definición de las ecuaciones matriciales y sus invariantes S y T.

El de técnicas de reducción o descomposición [12, 18, 41] facilita y simplifica el análisis de sistemas “grandes” al reducir estructuralmente el modelo obtenido a una descripción más general, denominada **subred o “macro”**, la cual mantiene las propiedades originales de la red que le dio origen [41]. Estas técnicas de transformación para las PN's, se utilizan para analizar vivacidad, seguridad, y limitación mediante la comprobación de buena conformación de la red [18]. Las transformaciones clásicas definidas por [41] son las representadas en la figura 1.2.

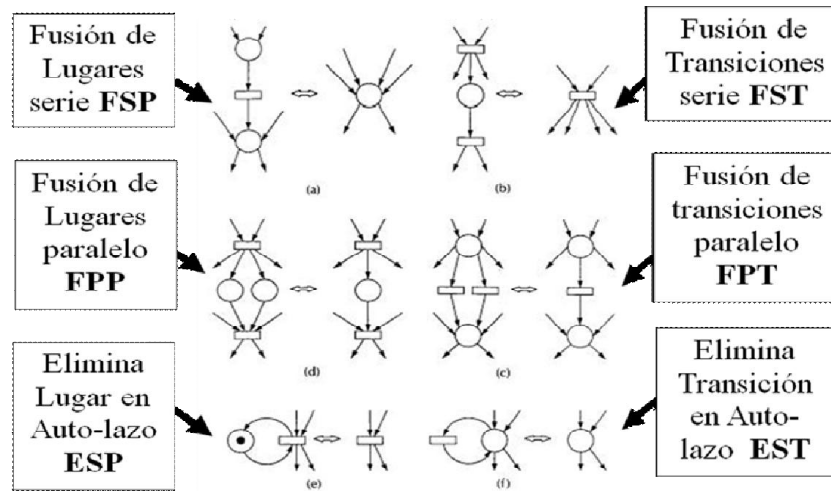


Figura 1.2 Ciclo de diseño de la Automatización Industrial [41].

1.2.2 Modelado de sistemas de automatización mediante redes de Petri.

Dentro de las PNs aplicadas al ámbito de automatización basada en PLCs se han desarrollado diferentes variantes y extensiones que difieren en su concepción, forma de representación, interpretación y los métodos de análisis que utilizan. En [31] se definen dos líneas principales hacia las que se orientan estos trabajos:

- **Control de realimentación de estados** (State feedback control): Basado en la adición de lugares de control a la PN, creando las **CtlPN** (Controlled PN).
- **Control de realimentación de eventos** (Event feedback control): Basado en la asignación de eventos a las transiciones, creando las **LabPN** (Labeled PN).

Las **CtlPN** son una tripleta $N_c = (N, C, B)$, donde $N = (P, T, F)$ es una red de Petri, C es un conjunto finito de lugares de control disjuntos de P y T , mientras que $B \subseteq (C \times T)$ es el conjunto de arcos desde los lugares de control a las transiciones.

Las **LabPN** (Labeled PN o PN generator) son una quintupla $G = (N, \Sigma, l, M_0, F)$, donde $N = (P, T, F)$ sigue siendo una PN, Σ es un conjunto (alfabeto) finito de eventos, $l: \Sigma \rightarrow T$ es una función que asigna eventos a las transiciones, M_0 es el marcaje inicial y F un conjunto finito de marcajes finales.

Sobre la primera tendencia se han desarrollado recientemente las **Automation Petri Nets** (APN) [55] y **GHENeSys** [29], mientras que a la segunda se asocian las **Signals Interpreted Petri Nets** (SIPN) [23].

En la primera tendencia fue desarrollado el modelo **APN (Autómata PN)** [55], lo cual está definido como $APN = (P, T, Pre, Post, In, En, X, Q, Mp)$, donde P son los lugares, T son las transiciones, $Pre (PxT)$ y $Post (TxP)$ son arcos normales entre P y T , In (arcos inhibidores) y En (arcos habilitadores) lo que representa la conexión de lugares auxiliares asociados a la presencia o ausencia de lectura de sensores o informaciones del sistema, X es el conjunto de condiciones asociado a las transiciones, Q es el conjunto de acciones atribuidas a los lugares (ellos pueden ser señales de Impulso y de nivel), Mp es la marcación inicial.

También se asocia a esta primera línea la Red de Petri **GHENeSys** [29] que puede ser definida como un séxtuplo $N=(L,A,F,K,M,Q,\Pi)$ donde: los elementos del conjunto L son llamados lugares y está compuesto por la unión de los conjuntos B y P (boxes y pseudoboxes respectivamente). Los elementos del conjunto A son llamados actividades. F es la relación de flujo ($F \subseteq (L \times A) \cup (A \times L)$) y sus elementos son llamados arcos. $K: B \rightarrow N^+ \cup \{\infty\}$ es la función de capacidad que indica la capacidad máxima permitida para cada Lugar B . $M: L \rightarrow N^+$ es la marcación inicial de la red con respecto a las capacidades de cada lugar, Q es una función que asocia acciones de nivel (atribución de resultado binario) para el subconjunto de los Boxes (B) o de impulso (Activación o Desactivación, “Set o Reset”) para el disparo de algunas actividades (A). Π es una función que marca la diferencia entre los elementos L y A , atribuyendo el valor “0” a los lugares simples y “1” a los Macro-elementos (elementos que representan sub-redes)

En esta definición inicial [29], fue incluida también la función Q [11], que permite a GHENeSys estar mucho más cerca de la programación en el PLCs, porque vincula su interpretación real con la estructura de la red de PN. Además, esta inclusión permite hacer traducción directa del modelo en PN para el programa en IEC1131.

A la segunda línea pertenecen las **Signal Interpreted PN** [23] que está definida como $SIPN = (P,T,F, M0,I,O,\phi,\omega,\Omega)$ donde $(P,T,F,M0)$ es una PN con su marcación inicial $M0$, I es un conjunto de entradas lógicas, O es el conjunto de las salidas lógicas, ϕ asocia toda transición

a una condición booleana (binaria) de I , ω asocia todos los lugares a los valores (0,1, -) de una salida O , Ω es una función que combina los valores obtenidos para cada salida con los lugares para evitar conflictos e indeterminaciones..

En [24] es indicado dentro de la definición de SIPN que la función Ω asocia a cada salida condiciones de: indefinidas (-), cero (0), uno (1), contradictorio (c), redundante a cero (r0), redundante a 1 (r1), y combinaciones de c con las dos últimas (c0, c1, c01). Aquí aparece una cuarta regla de disparo que habla de la repetición (prácticamente instantánea) de las sucesiones de disparo hasta que se aparece un estado estable donde ninguna transición es disparada (el proceso de disparo es iterado hasta que una marcación estable es alcanzada y mientras que esto sucede no cambian las condiciones de entrada (todo es incluido dentro de un ciclo del PLC)). Después que la marcación estable sea alcanzada las señales de salidas son recalculadas aplicando Ω a esta marcación. Esto complica el comportamiento de la red que queda lejos de las OPN.

En resumen, la red GHENeSys de Glez&Silva y la red APN de Uzam&Jones pertenecen al grupo de CtlPN donde hay gran semejanza entre los **Lugares de Control (CtlPN)**, las señales de sensores (APN) y los pseudoboxes (GHENeSys), en tanto la SIPN de Frey pertenece al grupo de **LabPN donde son atribuidos eventos a las transiciones**.

Los tres modelos (APN, GHENeSys y SIPN) usan la particularidad de **atribuir acciones para el proceso para los lugares** que permiten la extensión al trabajo de desempeño en el proceso controlado que no están presentes en las definiciones iniciales de CtlPN y LabPN de Holloway [31], pero en el caso de las SIPN asocia el valor de la salida y no la acción, lo cual obliga a tener una **función adicional (Ω)** de ajuste del valor final de la salida al terminar cada ciclo del PLC, lo que aumenta la explosión de estados.

Estas líneas también se difieren en la **forma de tratar las señales de los sensores** de estados del proceso. La variante de SIPN (cerca de las LabPN) limita la expresividad gráfica de la PN, al no representar en la estructura del modelo la acción de los sensores, tanto en el gráfico, como en la ecuación de estado. El modelo APN las representa gráficamente, pero también puede utilizarse la atribución de eventos a las transiciones. Mientras que el modelo

GHENeSys sólo permite los lugares auxiliares sin eventos de sensores lo que da una representación gráfica más completa.

En resumen, la variante de las **CtiPN** es considerada más esclarecedora y cercana a la teoría clásica de PNs, pero adicionando las facilidades de los **arcos inhibidores y habilitadores** para la conexión de los **Lugares auxiliares a las transiciones** (presente tanto en el caso de las APN como en GHENeSys). La acción de los sensores está de este modo garantizando su inclusión en el modelo con la persistencia de la marcación en esos Lugares especiales (ellos no son afectados por el disparo de transiciones). Las APN incluyen un campo más amplio (ellas pueden usar peso de arcos y hasta diferenciación de tokens) que puede llegar hasta las redes coloreadas, perdiendo así el poder de cálculo matricial simple. Esto obliga a que los métodos de análisis y síntesis con APN estén basados sólo en el análisis de gráficos de alcanzabilidad (RG), influenciado directamente por la explosión de estados. Las redes GHENeSys permiten usar las herramientas de análisis tanto de RG como las matriciales simples.

1.3 Modelado de sistemas de automatización mediante las redes GHENeSys.

Para un **gran campo de aplicaciones de PLCs** es más aconsejable el uso de GHENeSys [56, 47, 8, 10, 5] para el **modelado de los sistemas de automatización industrial**, ya que garantiza un 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** (pseudoboxes) no sólo representen mediciones de sensores del proceso, sino también estados de información que vienen de otras sub-redes o de partes de la misma red, lo que simplifica la estructura de redes complejas [29], y lo que permite mayor efectividad de las herramientas clásicas de análisis y síntesis de PNs para estos controladores.

La definición de **sub-redes (Macro-elementos o simplemente Macro)** en GHENeSys ayuda a la modularidad del modelo, favoreciendo la reusabilidad y la conformación de redes jerárquicas. Estos macro-elementos también permiten a GHENeSys abarcar el campo de las

aplicaciones no binarias en PLCs sin tener que pasar a redes de alto nivel (como las APN) y con una mayor semejanza con los lenguajes de PLCs.

La red GHENeSys permite la creación de módulos en bibliotecas que pueden construir un modelo mayor basado en estructuras típicas de control y entonces sumar una jerarquía. Esto favorece la construcción del modelo y su traducción para lenguajes de PLCs.

El modelado es desarrollado de forma modular por medio del uso de los subsistemas y estructuras típicas [28, 7, 55]. Por ejemplo, el caso de los motores es modelado en PN para sistemas de automatización como una estructura típica de tres estados: Parado, Funcionando y en 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 pseudoboxes. En el disparo, las transiciones de macro-elementos se comportan como elementos simples (Normales) pero la interpretación es realizada con una llamada a una sub-red que desarrolla una operación o función preprogramada por un especialista de PLC o por un usuario en un bloque funcional almacenado en la biblioteca del software de edición de programas.

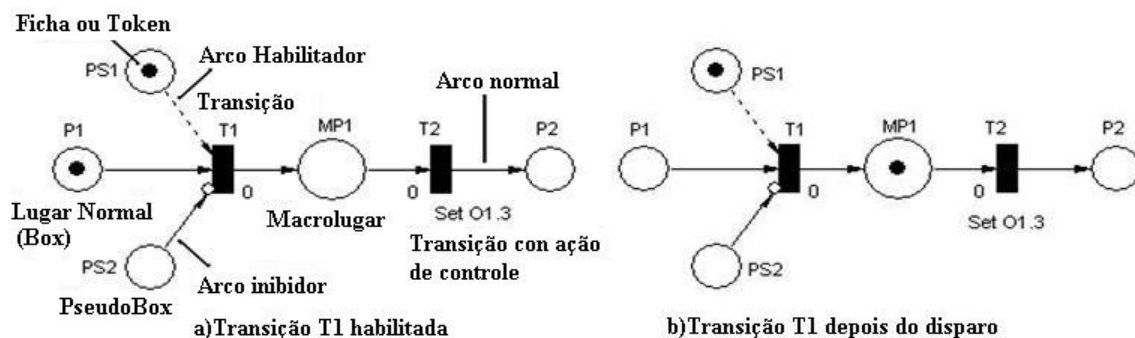


Figura 1.3 Tipos de elementos y reglas de disparo de PN extendidas GHENeSys

A continuación se presenta la metodología propuesta en [11] en el proceso de diseño de sistemas de automatización con PLC's, aplicada ya en nuestro país en algunos trabajos del ámbito industrial con efectividad, y que se compone por los siguientes pasos:

1. Estudio del sistema a controlar y los requerimientos funcionales del usuario.

2. Determinación de la mayor cantidad de unidades funcionales del sistema de control, sin coincidir necesariamente con su futura implementación.
3. Definición de acciones internas e interdependencias de cada unidad funcional y el nivel jerárquico en que deben estar de acuerdo a estas interdependencias y su función en el sistema de control.
4. Utilización de un diseño *Top-down* para establecer un modelo jerárquico del sistema, estableciendo una red GHENeSys para todo el modelo del sistema de control.
5. Refinamiento *Bottom-up* del modelo para detallar la estructura de cada subred del modelo, buscando su representación en un tipo de PN lo más sencilla posible. En casos que su simplificación sea imposible, y se requieran redes más complejas, se deben crear nuevas subredes para atender solo a esa situación. Esto permite aplicar a la mayor parte del sistema controlado métodos de diseño de modelos simples de PN's y las herramientas complejas solo en pequeñas subredes.
6. Clasificación de cada subred y determinación de sus propiedades funcionales (fundamentalmente *live* y *safe*) y estructurales (fundamentalmente S- y T-Invariantes).
7. Aplicación del método de reglas de reducción simples para revisar la estructura de la red, y lograr la mayor independencia entre selección y concurrencia de las ramas del modelo.
8. Reclasificación de cada subred modificada y redeterminación de sus propiedades, realizando adecuaciones que permitan su cumplimiento.
9. Simulación del trabajo de cada subred y comprobación del cumplimiento de los requisitos funcionales del usuario.
10. Reclasificación de cada subred modificada y redeterminación final de sus propiedades.
11. Estudio de las características particulares del equipamiento para su implementación y modelado de la subred que realiza la sincronización de las comunicaciones.
12. Traducción del modelo a un programa SFC en todos los niveles, considerando el criterio del usuario para seleccionar otro tipo de lenguajes (ST, IL, LD) para redes básicas.

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 del sistema. Por tanto es importante definir estos dos conceptos tan importantes en Ingeniería de Software.

El **Glosario de IEEE Standard** [40] define **Validación y Verificación (V&V)** como:

“el proceso de **determinar si los requerimientos de un sistema o componente están completos y correctos**, si los productos de cada fase de desarrollo **cumplen los requerimientos o condiciones impuestas por la fase previa**, y si el sistema o componente final **tiene conformidad con los requisitos especificados**.”

Diferentes autores distinguen **validación y verificación como dos pasos complementarios de garantía de calidad** [40, 15, 51, 21]. En resumen argumentan las dos etapas de la siguiente forma:

VERIFICACIÓN: Está orientada a garantizar las **propiedades generales** del modelo y la satisfacción de una **fórmula dada por el modelo**. Su principal característica es que se refiere a la **corrección interna de un modelo** del proceso. Como **opera sobre la estructura formal del modelo** del proceso, **puede ser realizada sin considerar el mundo real**.

VALIDACIÓN: Aborda la **consistencia del modelo con el universo de discurso**, o sea, el proceso del mundo real. Como es un **criterio de corrección externa**, es más difícil y ambiguo para decidir. **Exige la consulta de las especificaciones** y un proceso de discusión con las partes interesadas. A pesar de que validación exige el juzgamiento humano como una característica fundamental, importa resaltar que **los métodos formales son útiles para apoyarlo**. Por ejemplo, simulación, animación o derivación de declaraciones en lenguaje natural facilitan la validación de un modelo de proceso por parte de los utilizadores.

1.3.1 Verificación sobre GHENeSys.

Durante la verificación formal, la garantía inicial de una reducción de errores en el diseño parte desde la buena formación de los modelos creados. En Mendling [40] se resumen 7 **recomendaciones** que deben seguirse en cualquier proceso de modelado:

1. Usar la menor cantidad posible de elementos en el modelo.
2. Minimizar las rutas o caminos por elementos.
3. Usar redes con sólo un inicio y un fin.
4. Realizar modelos lo más estructurados posible.
5. Evitar selecciones indeterminadas mediante elementos OR.
6. Usar la mayor cantidad de comentarios e informaciones adicionales a cada paso del modelo.
7. Descomponer los modelos con más de 50 elementos.

El uso de estas reglas es fundamental en el desarrollo de aplicaciones de automatización con PLCs ayudando también a la simplicidad y posibilidades de expansión y reparación.

En GHENeSys se aprovecha la propuesta de **transformar el modelo en una PN pura** (PN sin fuentes y sumideros) [40] al eliminar los pseudoboxes y los arcos que van a las transiciones controlables para de este modo ejecutar la verificación de las propiedades de la red como Vivacidad y Seguridad en una PN pura. De este modo se cumple la condición necesaria de Vivacidad propuesta por Murata [41] (no tener fuentes y sumideros). También, muchos de los modelos de los subsistemas controlados generalmente pueden tener una baja clasificación dentro de las PN clásicas, lo cual permite métodos más simples de verificación.

La mayoría de las aplicaciones de automatización con PLCs pueden ser consideradas dentro de la gama de las **Redes de Petri de Libre Selección (Free-Choice, FC-nets)** [41]. Entonces el análisis de Vivacidad y Seguridad de los modelos controlados propuesta en GHENeSys puede usar los métodos propuestos por Desel y Esparza en su libro [18]. De acuerdo con el Cap.7 (Reducción y síntesis) de este libro, las **reglas de reducción** dan un método alternativo de análisis de buena-formación (well-formedness) en FC-nets. El algoritmo de prueba de buena-formación puede ser transformado fácilmente en un algoritmo de prueba de Vivacidad (Liveness) y Seguridad o Limitación (Boundedness) de sistemas FC.

La verificación de sistemas informáticos requiere de una plataforma para el modelado del sistema y de un método de verificación. La herramienta de diseño para GHENeSys [42] se encuentra todavía en proceso de desarrollo. Por tanto, en este trabajo se utiliza como plataforma de modelado el Visual Object Net 2.7 (VON2.7) [19, 20] con los modelos de redes de Petri extendidas GHENeSys y se emplean dos métodos de verificación formal: reglas de reducción simples y el enfoque analítico mediante la ecuación de estado. Como el VON2.7 no tiene opciones automáticas de cálculo analítico de los modelos, se utiliza la herramienta PIPE 2.5.

Se considera al VON2.7 como la herramienta que mejor se adapta a nuestros intereses por sus facilidades visuales de edición y simulación, aunque tiene la desventaja de no dar grandes facilidades de análisis, cálculo de invariantes, técnicas de reducción automática, implementación automática a lenguajes de PLC's IEC1131 compatibles, etc. No obstante, éstas pueden ser suplidas parcialmente añadiendo una herramienta como PIPE 2.5, que aunque no cuenta con las facilidades de simulación que VON, si implementa la posibilidad de análisis de propiedades funcionales, de desempeño, de invariantes, extracción de la matriz de incidencia entre otras menos comunes como la clasificación y comparación de PN's y que sólo utilizaremos para obtener las invariantes.

En la industria a menudo es considerado que lo importante es tener una representación detallada del sistema y simularlo. Sin embargo cuando los resultados no se corresponden con lo esperado (cuando contradicen el modelo cognitivo implícito del diseñador), tres conclusiones pueden derivarse:

- El sistema realmente se comporta así (el modelo cognitivo es incorrecto y/o incompleto)
- Hay un error en la especificación del modelo (debe corregirse para ajustarse al modelo cognitivo)
- Hay un error en el simulador (contactar al asesor del programa)

Por tanto, es importante destacar que sin un profundo análisis y verificación, es muy difícil seleccionar una de estas opciones. De aquí que se debe siempre verificar propiedades antes de validar mediante simulación

1.3.2 Validación sobre GHENeSys.

En [27] se explica que los métodos de análisis de modelos en PN incluyen la confirmación de si existe **correspondencia funcional** entre el modelo con las especificaciones de exigencias originales (típicamente expresadas informalmente), lo que es probado en la Validación. Alcanzar esta correspondencia requiere de experiencia de modelado y conocimiento de las técnicas que ayudan a la construcción de modelos. Para este objetivo es incluido también el **completamiento (completeness)** de las especificaciones de exigencias. Estas generalmente son expresadas como relaciones de E/S del sistema. Hay entradas que no están definidas en las exigencias iniciales y que deberían ser completadas. Otro aspecto importante es la **Consistencia (consistency)** de las especificaciones de exigencias. No existe Consistencia (inconsistencia) cuando una combinación de entradas da varias combinaciones de salidas. Todo eso debería ser probado minuciosamente durante la validación del modelo.

En este libro es considerada también la Simulación de eventos discretos como otra variante para el test de las propiedades del sistema. Para esto es usado un algoritmo de ejecución para simular el funcionamiento del modelo de la red. Esta es una técnica extensa y consumidora de tiempo. Muestra la presencia de propiedades indeseables, pero no prueba la **Corrección (Correctness)** del modelo en casos generales. No obstante, permanece como una propuesta para muchos autores. Específicamente se considera esto una herramienta de análisis de comportamiento cuando las limitaciones de memoria de los computadores no permiten generar los gráficos de alcanzabilidad. Se presenta que la simulación permite observar cuantas veces un lugar es marcado o no y calcular la probabilidad de esto. Entonces la **simulación de la dinámica de tokens** puede generar análisis estadístico importante del comportamiento de la red. Mucho más cuando la herramienta de simulación permite trabajar con tiempos reales de ejecución del sistema en el modelo.

En la validación en GHENeSys de esta metodología debe tenerse en cuenta el estudio simulado del comportamiento del modelo controlado para **verificar el cumplimiento de las restricciones de estados prohibidos y la secuencia del comportamiento correcto del controlador**.

Considerando el **modelado modular jerárquico en GHENeSys**, las sub-redes para analizar son **propias** (una entrada, una salida y caminos de E/S que cubren todos los nodos de la red) y por tanto durante la verificación simulada se debe alcanzar la vivacidad de la sub-red. Entonces como estas sub-redes no tienen dimensiones grandes pueden ser restablecidos los pseudoboxes y realizada la simulación del comportamiento como un medio de evaluación de la ejecución de las especificaciones del usuario para el sistema controlado.

Por ello es muy importante definir una herramienta eficaz en la simulación del comportamiento de una red (el Visual Object Net 2.7 [19, 20]).

1.3.3 Traducción de modelos GHENeSys a programas IEC61131compatibles.

La metodología que usa las **redes jerárquicas extendidas GHENeSys** para el modelado de sistemas de automatización industrial con PLCs tiene muchas ventajas por los pasos que se ejecutan en ella y por las analogías del modelo con la programación IEC61131 compatible. Por ello es usada la metodología anteriormente descrita hasta la obtención del **modelo formal validado**.

Esto permite obtener un **modelo jerárquico** en varios niveles de la aplicación entera, donde cada nivel puede ser formado a través de tantas sub-redes como unidades funcionales (UF) que tengan la suficiente independencia como para ser consideradas UF. Esto garantiza una equivalencia con la **programación estructurada modular** que la IEC61131 propone por medio de los bloques funcionales y rutinas permitidas.

La restricción de que estas sub-redes sean **propias** (una entrada y una salida y todos los nodos envueltos en caminos directos E/S), **permiten rutinas y bloques funcionales con estas características**.

La garantía de la metodología para crear **redes bien-formadas y vivas**, permite alcanzar las estructuras exigidas para generar automáticamente los **bloques funcionales o programas, libres de bloqueos, con auto-recuperación y seguridad de operación**. Para lo que contribuyen los pasos de verificación ya realizados.

También la **validación**, ya llevada a cabo, garantiza que el programa que es obtenido cumple las **exigencias funcionales** que fueron definidas por el usuario, porque el modelo

demostró esto. Pero también está en correspondencia con la solución generada a los problemas de estados prohibidos y cadenas deseadas considerados durante todas las fases del análisis y síntesis formal hasta llegar al modelo validado a través de simulación.

La concepción de la creación de cada sub-red integrando las **estructuras típicas** de Sistemas de Eventos Discretos permite subdividir las estrategias de implementación o reinterpretación de acuerdo con estas mismas estructuras que fueron usadas en la construcción del modelo. Entonces permiten prever desde las primeras fases de la construcción del modelo, cuales son las posibles estructuras del programa de implementación.

Pero cada sub-rede creada por el usuario es una nueva estructura que puede ser **reutilizable**, entonces puede ser almacenado en una **biblioteca** para la conformación de sub-redes más complejas. De este modo además de las estructuras típicas ya almacenadas, cada proyectista podrá ir adecuando su entorno de trabajo con nuevas bibliotecas de sub-redes que ya tienen pre-elaboradas y validadas, para luego alcanzar la integración de sub-redes más complejas para la conformación de su sistema.

De acuerdo con todo lo mencionado arriba tenemos muchas posibilidades por medio de la metodología de GHENeSys.

Entonces, a continuación, serán explicadas las **reglas generales de analogías** entre los elementos de la red jerárquica extendida de GHENeSys y los lenguajes de programación IEC61131 compatibles. Esto permite construir el cuadro de equivalencias entre las estructuras básicas de los modelos y las secciones de programas IEC1131 compatibles que quedan como herramientas de la traducción.

Como se trabaja en una red GHENeSys, no pueden ser usadas las reglas establecidas por Frey para las SIPN, pero tampoco son aplicables todas las analogías presentadas en el trabajo de Uzam [55] para las APN, por las diferencias previamente descritas.

Entonces se presentan **nuevas analogías** relacionadas a continuación:

Para los **boxes** se mantiene la equivalencia a un bit interno (marca de bit o bandera) del PLC. Es decir, el estado 0 o 1 corresponderá con el estado de una variable binaria interna del PLC.

Los **pseudoboxes** corresponderán con tipos de señales diferentes de acuerdo con su uso:

- Si representan mediciones de **sensores** del proceso, ellos corresponderán con **entradas** del PLC (que de acuerdo con la IEC1131 pueden usar un identificador simple o su direccionamiento directo en el PLC).
- Si representan **informaciones** que vienen de otras sub-redes u otras partes de la propia red, ellos corresponderán con variables binarias internas (**bits internos o banderas**) del PLC (que también pueden usar un identificador simple de acuerdo con la IEC1131 o su direccionamiento directo).
- Si representan un **bit de salida de un temporizador o contador**, ellos corresponderán con variables representadas por el **identificador del temporizador o contador seguido por " .Q "** como lo indica la norma IEC1131.
- Si representan la **salida binaria de algún bloque funcional**, ellos tendrán el **nombre del bloque funcional seguido por ".nombre" de la salida del bloque** como establece la IEC1131-3.

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 (p.e. la parte condicional (instrucciones LDN, LD, AND) de cualquier sección de un programa de PLCs). Las operaciones lógicas entre estas señales estarán en dependencia de la estructura que las entrelazan con las transiciones que luego se dispararán. Es decir, si todas las señales entran a una única transición, entonces ellas conforman una operación lógica **AND**. Si ellas llegan a varias transiciones que después se unen en un único Lugar, entonces son traducidos en una función lógica **OR**. Combinaciones de estas estructuras también darán combinaciones de esas operaciones lógicas en el programa resultante.

La traducción de esta parte condicional para lenguajes LD y IL está prácticamente implícita en la definición de los lenguajes IEC 1131 compatibles. En **LD ellos son contactos en serie**

o paralelo (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.**

Se considera estas PNs deben ser libre-contacto a la hora de traducirlas, en caso contrario ellas deberían ser transformadas para libre-contacto usando la transformación del complemento [41, 45]. Esto elimina la necesidad de incorporar a las traducciones la parte del examen de los estados de los lugares de salida de las transiciones propuesta por Frey [24], favoreciendo el proceso inverso de reinterpretación.

El **resultado del disparo** de una transición conformaría la **parte de acción** (p.e. parte de acción (instrucciones S, R, ST) en las secciones de un programa de PLCs) eso continua a toda la parte condicional de una red de programación elemental en cualquier lenguaje de PLCs (ejemplo el IL o el LD). Mas aquella acción obligatoriamente indica el **reseteo (desactivación) de todos los bits internos asociados a los lugares de entrada y el seteo de los bits internos asociados para los Lugares de salida.** Esto corresponde con una técnica también empleada en programación directa de PLCs que consiste en representar por medio de banderas (bits internos o marcas) el estado de activación de la secuencia del programa para entonces poder usarlos en cualquier otra parte del programa que se requiera para conocer este estado, o para saltos u otras acciones auxiliares. Esto puede extender algo más el programa resultante (principalmente en IL), pero darían más seguridad para la aplicación. No obstante, el usuario nombra en el modelo las variables para los lugares y la traducción sólo usaría nuevos bits internos si aquel estado no tiene nombrada ninguna otra variable y de este modo es evitada la redundancia de estados en la traducción automática del modelo para el programa.

Entonces esta parte de acción resultante de la traducción del disparo de aquella transición será aumentada con las **acciones de impulso asociadas** a la misma, quiere decir, el seteo (activación) o reseteo (desactivación) (como es 1 o -1 el valor de la asociación de la función Q en la definición de GHENeSys) de las salidas para el proceso que requerirán ejecutar en el cambio de estado que representa esta transición en el proceso real.

Indudablemente **en LD esta parte de acción es representada en bobinas de tipos diferentes** (Normal, Negada, S, R). En **IL conforma el conjunto de las instrucciones de tareas de seteo y reseteo**. Para el lenguaje **ST es el cuerpo de la parte ENTONCES de una alternativa SI (en inglés IF)**.

Atrás de cada transición existe por lo menos un Lugar (sólo Boxes). Como los **Boxes pueden tener acciones de nivel** en salidas para el proceso asociado al estado por medio de la función Q, fue **consultado por el estado del bit interno** asociado para aquel Lugar y entonces son **ejecutadas las tareas directas o negadas** (como es 1 o -1 el valor de la asociación de la función Q) en las salidas para el proceso.

Todas estas salidas pueden ser nombradas por el direccionamiento directo en el PLC o por un identificador que entonces se asocia a aquella dirección como en la norma IEC1131. El uso de identificadores da bastante generalidad en el programa para desligar esto de su implementación y facilitar su reutilización. Entonces la configuración del programa en el ambiente de edición de la aplicación permite esta tarea (p.e. uso del configurador de la E/S en la norma ISaGRAF).

Para la traducción a lenguajes IL, LD y ST de la consulta para el bit interno del Lugar y entonces las acciones de nivel que tienen asociadas siguen las mismas reglas previamente expresadas para cualquier condicional y acción.

Si estas acciones de tarea, seteo (activación) o reseteo (desactivación) son ejecutados en **entradas de un bloque temporizador, contador u otro bloque funcional** cualquiera, el nombre de la variable corresponde entonces con el **identificador del bloque funcional seguido por “.nombre_de la_entrada”**.

Además, cualquier tipo de bloque funcional puede no recibir sólo entradas binarias, ni tiene sólo salidas binarias. Generalizando para cualquier bloque funcional, es necesaria una fase de carga de señales de diversos tipos (a veces ninguna binaria), luego el procesamiento de estas señales, y al fin la transferencia de los resultados para las variables asociadas a las salidas del bloque (algunos casos ninguna binaria). Para esto la IEC 1131 da una solución normalizada a la inclusión de ellas en la programación LD que es el uso de las entradas EN y ENO para la asociación de bloques funcionales para estos diagramas de escalera.

Entonces, este modelo tiene la ventaja de esta regla que es tan común entre los programadores de PLCs, lo cual queda muy bien para el tratamiento de señales analógicas dentro de una OPN. Esto mantiene la potencialidad gráfico-analítica de la OPN para el proceso de verificación y validación sin afectar la representatividad del modelo en frente del proceso real, pero sin complicar esto.

También lleva ventaja otro elemento del sistema GHENeSys, fundamental para no perder formalidad en este modelo, los **Macro-elementos**.

Las redes en GHENeSys permiten la existencia de Macro-Lugares y Macro-Transiciones, (*Macroboxes y Macroactivities*) que son sub-redes propias representadas en el modelo por un único elemento de mayor tamaño. Los Macro-elementos indican la ejecución de varias acciones que pueden ser complejas, pero que no afectan la inter-relación con los otros elementos del modelo, aparte de las entradas y salidas de aquel Macro-elemento representadas en el diagrama.

Entonces cualquier bloque funcional podrá ser representado por tres tipos de Macro-elementos:

1. **Macro-Lugar de entrada:** Representa una sub-red que ejecuta la carga de todas las variables del proceso para señales de entrada al bloque funcional.
2. **Macro-Lugar de salida:** Representa una sub-red que ejecuta la transferencia de todas las señales de salida del bloque funcional para las variables del proceso.
3. **Macro-Transición o Macro-Lugar de ejecución:** Representa la sub-red que realiza la ejecución de las operaciones internas del bloque funcional. Aquí hay dos tipos fundamentales de operaciones normalizadas en todos los lenguajes de PLCs IEC1131 compatibles: las operaciones normalizadas en un lado y los bloques funcionales y funciones normalizadas en el otro. En el primero caso, ellas serán representadas por Macro-Transiciones, y en el segundo, por Macro-Lugares, porque en el segundo caso la complejidad es mayor al requerir la llamada al bloque funcional o función y muchos de ellos generalmente requieren de una transición resultante activada por un bit de salida de aquella operación (p.e. bit de salida de temporizadores).

Esto permite gran simplificación en la representación de temporizadores y contadores en el modelo en OPN (no es necesario usar transiciones con tiempo, ni peso de arcos como en las APN de Uzam [55]). También permite ejecutar operaciones con bytes o palabras en los modelos con OPN, porque estas acciones son encapsuladas en la sub-red del Macro-elemento y ellos no exigen ser modeladas al nivel de esta red.

Quiere decir que en esta metodología no es exigido dejar las OPNs para representar funciones de temporización, conteo, ni otra función analógica como propone el método de Uzam [55]. Ni es necesario introducir dentro del modelo estructuras complejas para modelar funciones de temporización, conteo o comparación como propone el método de Vieira [Vieira99]. Ni es necesario asociar operaciones lógicas entre señales para las transiciones como propone el método de Frey [24] oscureciendo la representación gráfica y matemática del modelo.

Esta variante de modelado aprovecha la normalización aprobada en la IEC1131 para integrar los bloques funcionales y operaciones normalizadas en la programación LD al modelado en GHENeSys, es decir, el uso de las entradas EN y salida ENO si este bloque no tiene entradas y salidas booleanas (binarias) adaptado al modelado, lo que le da una fuerte compatibilidad con esta norma aprobada por todos los fabricantes y usuarios de PLCs desde 1993.

La metodología también permite modelar el comportamiento de los tres temporizadores (TON, TOF, TP) y los tres tipos de contadores (CU, CD y CUD) de esta norma internacional que no es posible en las propuestas previas.

El modelo de la aplicación no exige que le sea anexado el modelo del comportamiento de estos temporizadores y contadores (porque esta encapsulada en los Macro-elementos), como también de cualquier otro tipo de bloque funcional, igualmente sucede en los programas en lenguajes de PLCs que sólo se programan las entradas y salidas del bloque funcional y no su estructura interna. Esto coincide con la metodología de PLCs de la IEC1131 en la cual el usuario no necesita interferir en la programación interna del temporizador, contador o cualquier otro bloque funcional, sólo requiere anexar al programa una instanciación de esto. De igual forma el modelo sólo anexa una instanciación de este bloque

(Macro-elemento) usando las entradas y salidas, el resto es modelado interno propio del ambiente de modelado o de la programación. En el caso que se utiliza un ambiente de modelado estandarizado sólo será requerida esta anexión (modelo interno del Macro-elemento) en el caso que se precise de una simulación automática completa de la operación del sistema controlado entero. En este caso además de anexar este bloque deberán ser anexados los modelos que simulan el comportamiento del proceso a controlar (igual a como es llevado a cabo en los ambientes de programación IEC 1131 compatible). De este modo en el modelo de la red para verificación está sólo la parte que representa el programa de control (es decir, el controlador) y entonces aquí es donde se tiene certeza del cumplimiento de las propiedades para tener una síntesis eficiente y por tanto un programa resultante también más eficiente, ese es el objetivo central de la introducción de los métodos formales en la programación de PLCs.

En los bloques funcionales que son representados en el modelo en GHENeSys son usados los pseudoboxes para condicionar el momento de disparo de la transición de salida del modelo del bloque para la verdadera ejecución del resultado de la acción del bloque funcional (p.e. bit de salida de temporizadores y contadores). Cuando no hay bit de salida de la ejecución del bloque, esta transición es ejecutada automáticamente después de tener activado el Macro-elemento de ejecución interna del bloque funcional.

De este modo está resuelto el gran problema de representar por medio de OPN las operaciones no binarias combinadas con las binarias en los modelos, lo cual es tan común en muchas aplicaciones de PLCs, sin necesidad de usar redes coloreadas u otro tipo de redes de alto nivel. De este modo mantenemos la potencialidad de verificación gráfico-analítica del modelo sin dejar de representar cualquier aplicación de PLCs.

También queda una correspondencia más directa entre la estructura de un programa IEC1131 compatible con el modelo en PNs, porque los bloques funcionales son previstos desde la fase de diseño sin tener que llegar al método extenso propuestos por Vieira.

Este trabajo utilizará la automatización del subproceso Dosificación en la planta de fabricación de bloques tipo POYATOS del complejo productivo “Los Guaos” perteneciente a la Empresa de Materiales de la Construcción de Santiago de Cuba como caso de uso para

demostrar las cualidades de esta metodología de modelado formal de sistemas de automatización industrial utilizando las redes de Petri jerárquicas extendidas GHENeSys y las herramientas del VON y PIPE, lo cual será detallado en el siguiente capítulo.

Conclusiones Parciales.

1. De la reseña realizada sobre los métodos formales de diseño se evidencia que las redes de Petri son sin duda uno de los métodos formales más aceptados por investigadores y académicos en la representación tanto abstracta como matemática de los sistemas de automatización.
2. La combinación de las características gráficas y el basamento matemático de las redes de Petri, así como su compatibilidad con los lenguajes de programación IEC 61131 hacen de ella una herramienta útil para el modelado, verificación y simulación de los mismos.
3. Dentro de las PNs aplicadas al ámbito de automatización basada en PLCs, GHENeSys es sin duda la más representativa, por incorporar en su definición las características de interés para el modelado de sistemas de automatización.
4. Después de tener una idea general de la evolución histórica de los métodos formales de diseño de sistemas automatización industrial, se puede afirmar que para garantizar la disminución de la posibilidad de ocurrencia de errores y así obtener un sistema de automatización más eficiente y seguro se deben emplear las PNs como método formal en la etapa de diseño de un sistema de automatización integrada.

Capítulo 2: Diseño y programación del sistema de automatización integrada del subproceso “Dosificación”.

La producción de bloques de hormigón en el país es una necesidad de primer orden, provocado por la creciente demanda de este producto en las diversas obras constructivas que se desarrollan a todo lo largo y ancho de la isla. El método de fabricación más utilizado en el país es el mecanizado, de ahí que contamos con grandes complejos productivos dentro de los que se encuentran plantas encargadas de esta función. Las plantas productivas de fabricación española “POYATOS” son las más utilizadas, de ellas dependen un 73,8% de la producción general de bloques del país.

Por tales razones el estudio, mantenimiento y mejoramiento del sistema productivo de dichas fábricas son tareas necesarias para el logro de mejores volúmenes de producción.

En este capítulo se aplicarán las PNs como formalismo en el diseño del sistema de automatización integrada del subproceso de dosificación en la planta de producción de bloques de hormigón tipo “POYATO”, perteneciente al combinado “Los Guaos” de Santiago de Cuba, con el objetivo de mejorar el diseño del sistema de automatización.

Se pretende obtener un sistema modular que cumpla las principales propiedades estructurales y funcionales de un modelo en PN y que garantice el cumplimiento de los requerimientos funcionales exigidos por el usuario.

2.1 Descripción del flujo tecnológico del proceso de producción de bloques de hormigón en plantas tipo “POYATOS”.

El flujo tecnológico en plantas tipo “POYATOS” puede dividirse en cuatro etapas principales como se muestra en la figura 2.1, estas etapas estarán definidas por las funciones que se realiza en cada una y reciben los nombres de: dosificación, prensado, transportación y empaquetado.

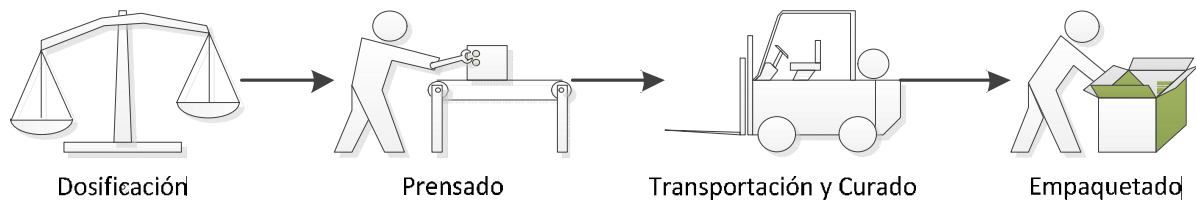


Figura 2.1 Flujo tecnológico del proceso de producción.

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

Paralelamente se inicia la dosificación del cemento almacenado tolvas por medio de un tornillo sinfín que descarga sobre una báscula. Una vez terminada la dosificación, el *Skip* es elevado hacia una mezcladora donde descarga y se mezclan el árido, el cemento, y posteriormente agua durante un tiempo determinado. Al estar lista la mezcla, que ahora podemos llamar hormigón, es descargada en el cajón de la prensa por medio de otra banda transportadora.

- **Prensado:** ya en la máquina de prensado, apoyado en las tablas soporte se llenan los moldes de los bloques y son prensados a una presión 120 Kg/cm^2 durante un tiempo definido por el operador. Estas tablas soporte llegan inyectadas al sistema de prensado por medio de un sistema neumático de pistones llamado “inyector de tablas” que las recoge al ser liberadas de los bloques secos. Luego de ser prensada la mezcla obtenemos los bloques frescos los cuales serán transportados en sus tablas soportes hacia un ascensor, donde al llegarle dos tablas soportes las eleva 40cm acumulando 10 pisos.
- **Transportación y Curado:** Una vez lleno el ascensor, un carro transportador comandado por un operador recoge los bloques frescos y, desplazándose sobre dos raíles, los lleva hacia las celdas de curado donde serán sometidos a un tiempo de

curado al aire libre mientras el hormigón fragua. Al ser depositados los bloques se procede con la recogida en una de las celdas adyacentes de los bloques secos de producciones anteriores, que son llevados en esta ocasión al descensor, que será el encargado de ir descendiendo piso a piso las tablas soporte y colocándolas en la estructura de transportación de los bloques hacia su empaquetamiento.

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

Estas cuatro etapas descritas anteriormente se repiten de forma cíclica durante toda la jornada de producción.

2.1.1 Subproceso “Dosificación”.

El subproceso “Dosificación”, también conocido como “Planta de Hormigón”, está constituido de manera general por tres tolvas de áridos con tres alimentadores respectivamente, una cinta de áridos, un carro colector o *Skip*, dos silos de cemento cada uno con un sinfín, una tolva de cemento, una mezcladora horizontal, una cinta de hormigón y finalmente el cajón de la prensa como se puede observar en el Anexo 1. Su objetivo dentro de la planta de fabricación de bloques “POYATOS” es garantizar la existencia de hormigón en el cajón de la prensa durante el ciclo productivo.

Con el fin de cumplir con este objetivo se deben realizar un sin número de tareas que pueden clasificar sea nivel general según la función que realizan cómo:

- **Dosificación de los áridos:** Proceso mediante el cual se descargan simultáneamente el árido polvo de arena, arena y granito a través de pequeñas cintas transportadoras (alimentadores) en las tolvas correspondientes sobre la cinta transportadora encargada de servir el *Skip* (cinta de áridos). En este caso la dosificación es volumétrica, lo que quiere decir que la cantidad de materiales estará dada en relación

al tiempo de funcionamiento de los alimentadores, tiempos de funcionamiento que no son iguales y dependen de la receta utilizada en la fabricación del hormigón. Un motor vibrador en la tolva correspondiente al polvo de arena evita que este se adhiera a las paredes. Este proceso se repite un vez que se haya descargado el *Skip*.

- **Dosificación de cemento:** Proceso mediante el cual se extrae cemento de los silos habilitados a través de los tornillos sinfines correspondientes y se descarga en una pequeña tolva (tolva de cemento) provista de una célula de carga hasta llegar al peso requerido. La dosificación en este caso es gravimétrica, lo que quiere decir que la cantidad de material estará dada por un peso específico según la receta utilizada en la fabricación del hormigón. Este proceso se repite inmediatamente que se haya descargado el cemento de la tolva.
- **Descarga de áridos:** Proceso mecánico que ocurre al llegar el carro colector (*Skip*) a la posición de descarga sobre la mezcladora una vez que ha sido serviciado. La descarga estará determinada por un tiempo, luego del cual el *Skip* vuelve a su posición inicial en la zona de carga.
- **Descarga de cemento:** Proceso mediante el cual se vacía en la mezcladora el cemento dosificado en la pequeña tolva a través de una trampilla en el fondo de la misma. Un motor vibrador evita que este se adhiera a las paredes.
- **Dosificación de agua:** Proceso mediante el cual se bombea una cantidad específica de agua en litros dentro de la mezcladora a través de una válvula y un caudalímetro de según la receta utilizada en la fabricación del hormigón.
- **Mezclado de los materiales:** Proceso mediante el cual se mixtura durante un tiempo determinado los áridos, el cemento y el agua vertidos en la mezcladora según la receta utilizada en la fabricación del hormigón.
- **Descarga del hormigón:** Proceso mediante el cual se descarga el hormigón de la mezcladora hacia el cajón de la prensa a través de una compuerta y una cinta transportadora (cinta de hormigón), el proceso de descarga estará definido por un tiempo, luego del cual la mezcladora estará lista para recibir nuevamente los materiales.

En la actualidad el funcionamiento de este proceso se realiza de forma manual, debido al deterioro de algunos de los medios técnicos necesarios para su funcionamiento automático ocasionado mayormente por el tiempo de explotación y la falta de mantenimiento. Esto conlleva a una baja eficiencia del sistema productivo debido a la introducción de demoras en el proceso y al incremento de la probabilidad de ocurrencia de errores, que pueden contribuir al aumento de la peligrosidad y al deterioro acelerado de los medios de producción por mala manipulación, tornándose en pérdidas económicas para el país. El diseño del programa era lineal, sin módulos, ni diferenciación de funciones del modo manual, la dosificación del agua era solo temporizada, existía una posición intermedia del carro colector que no cumplía su objetivo, no se realimentaba al PLC el accionamiento de las protecciones térmicas de los motores, no se recibía información de la posición de los pistones neumáticos de las compuertas de cemento ni de la mezcladora. Tampoco existía ningún tipo de supervisorio que permitiera visualizar el funcionamiento en diferentes modos de operación, las tendencias de las principales variables, el registro de las alarmas y fallos y las estadísticas de producción. Además, el sistema no contaba con manual de mantenimiento y reparación. Todo esto, unido a la obsolescencia del PLC existente (TSX 2720), hace del sistema un proceso poco fiable a largo plazo e impone la necesidad de realizar una automatización integrada que permita corregir los problemas actuales, mejorar la calidad del proceso de “Dosificación” y por consiguiente la calidad en la producción de bloques.

2.2 Diseño del sistema de automatización integrada propuesto.

En la etapa de diseño de cualquier automatización se requiere realizar el análisis y síntesis del sistema automatizado. Esto es solo posible de manera eficiente con el uso del Modelado Formal de la planta industrial teniendo en cuenta sus especificaciones de seguridad y comportamiento como sistema controlado, al cual se le realiza la verificación de sus propiedades principales y validación del cumplimiento de las exigencias del cliente por medio de la simulación del funcionamiento de la planta con su control integrado.

2.2.1 Propuesta de automatización.

Atendiendo a las necesidades de la empresa y tomando en consideración los criterios del personal de la planta se proponen los siguientes modos de operación para el sistema automatizado:

- **Manual:** se utilizará solamente para fines de pruebas, mantenimiento, y para la inicialización del proceso en caso que lo amerite. El operador podrá accionar directamente los diferentes actuadores mediante los pulsadores y llaves de múltiple posición presentes en un panel.
- **Manual Extendido:** se empleará en la inicialización del proceso antes de pasar a automático o para la producción en caso de que sea necesario. A diferencia del manual una vez accionados los botones correspondientes a dosificación del árido1, arido2, árido3, cemento y agua, los procesos relacionados a dichos botones solo se detendrán una vez alcanzados los valores definidos en la receta de fabricación de hormigón. No obstante, es responsabilidad del operador garantizar la secuencia adecuada de los procesos descritos anteriormente.
- **Automático:** permitirá, al contrario de los dos anteriores, la realización de todo el proceso de forma cíclica y automatizada sin la intervención del operador. Ejecutando ordenadamente las funciones definidas en el proceso de “Dosificación”.

Para garantizar la capacidad de respuesta y la funcionalidad del sistema automatizado, es necesario que este tenga la mayor información posible del entorno, concretamente, de los estados observables del proceso, lo que se logra con la correcta selección y ubicación de sensores, y al mismo tiempo es necesario que el sistema sea capaz de actuar en consecuencia con esta información de manera que se cumpla el objetivo final para el que fue diseñado, capacidad que le proporcionan los distintos actuadores utilizados.

En el Anexo 2 se puede observar la distribución de sensores y actuadores en el subproceso “Dosificación”, la mayoría en existencia desde la adquisición de la planta “POYATOS”, y una tabla con la descripción de cada uno de ellos teniendo en cuenta el tipo y función que realiza dentro del proceso.

También garantizar la operatividad del sistema impone contar con un panel como el que se muestra en el Anexo 3 desde el cual se pueda a través de pulsadores y selector energizar los elementos de fuerza, seleccionar los modos de operación, seleccionar que silo de cemento emplear, arrancar o parar el sistema, accionar los diferentes actuadores (muy importante teniendo en cuenta los modos Manual y Manual Extendido propuestos) y realizar parada de emergencia.

En base al análisis profundo, a la observación del proceso productivo en varias visitas a la planta, y al criterio del cliente se formuló una serie de requisitos necesarios para el correcto funcionamiento automático de la planta, entre los que podemos encontrar:

- El proceso de dosificación de áridos se realizará inicialmente siempre que el *Skip* se encuentre abajo y se repetirá una vez que este haya sido vaciado.
- La cinta transportadora de áridos funcionará únicamente si el *Skip* se encuentra abajo inicialmente o después de haber sido vaciado.
- Los alimentadores de los áridos 1, 2 y 3 solo trabajarán si la cinta transportadora de áridos está funcionando.
- El proceso de dosificación del cemento (carga de cemento en la tolva) se realizará siempre que la tolva esté vacía y la compuerta de descarga cerrada.
- El *Skip* subirá los áridos a la mezcladora si la dosificación de cemento ha terminado, el motor de la mezcladora se encuentra encendido, y ésta ha vaciado el hormigón de alguna operación anterior y el selector de sonda/continuo se encuentra en continuo, o en sonda, y se recibe la señal de nivel bajo en el cajón de hormigón de la prensa.
- La descarga del cemento en la mezcladora se realizará siempre después que el *Skip* descargue los áridos.
- La dosificación de agua se realiza después de la descarga de áridos cemento.
- La descarga del hormigón al cajón de la prensa se hará una vez terminado el tiempo de mezclado.
- En el panel de operación local se debe tener la opción de parar y arrancar el sistema una vez energizado sin desactivar el pupitre o activar emergencia.

- Se necesita diferenciar la operación manual para mantenimiento y reparación, del manual extendido que garantiza la receta de fabricación de hormigón.
- El sistema sólo puede estar operando en uno de sus tres modos de operación de forma segura y eficiente.
- Se requiere eliminar la posición intermedia del Skip porque aumenta el número de arranques del motor del Skip con carga que genera un consumo excesivo innecesario y un deterioro del mecanismo de subida.
- Se necesita que el programa responda automáticamente ante el fallo de sobreconsumo de algún motor.
- Se requiere un método más eficiente de dosificación de agua para mejorar la calidad de la mezcla.
- Se debe contar con la información de la posición de los pistones neumáticos de las compuertas de cemento y de la mezcladora.
- Se requiere un programa modular que permita el mantenimiento, reparación o ampliación del sistema de forma ágil y segura.
- Se requiere una supervisión al nivel del estándar de automatización mundial para una operación eficiente en los diferentes modos de operación.
- Se necesita realizar la configuración del sistema (definir los tiempos y cantidades de material según la receta de fabricación de hormigón)
- Se debe graficar y visualizar numéricamente las tendencias de las variables de producción fundamentales (consumo de energía, horas de encendido, etc.).
- Se debe registrar los fallos y alarmas, tiempos de operación de motores, consumo eléctrico de las diferentes partes de la instalación, consumo de materiales, etc.

2.2.2 Estructura del sistema de automatización integrada propuesto.

Partiendo del número de funciones y requisitos, así como de la cantidad de sensores y actuadores necesarios para acometer la automatización del subproceso de “Dosificación” se propone entonces un sistema automatizado con la estructura mostrada en la figura 2.1 donde coexisten los tres niveles básicos a tener en cuenta según la pirámide de automatización.

- **En el nivel más bajo o de instrumentación de campo:** se encuentran los sensores y actuadores ubicados en el proceso.
- **En el nivel intermedio o de control:** se encuentran los elementos de automatización local que incluyen: el panel de control donde se selecciona el funcionamiento del sistema y su actuación según el modo de trabajo seleccionado, el PLC que según las señales recibidas de los sensores y del propio panel de control a través de los módulos de entrada, y de acuerdo a la programación implementada accionará sobre los pre-actuadores y actuadores según sea el caso con el fin de ejecutar las acciones requeridas ya sea en modo Manual, Manual Extendido o Automático sobre el proceso.
- **En un nivel superior o de supervisión:** se encuentra, como su nombre indica, el sistema supervisorio a través del cual se podrá configurar el sistema (definir los tiempos y cantidades de material según la receta de fabricación de hormigón), observar el comportamiento dinámico del mismo (estado en que se encuentra en el tiempo), graficar y visualizar numéricamente tendencias (consumo de energía, horas de encendido, etc,) y sobre todo observar la ocurrencia de eventos no deseados (alarmas o fallos).

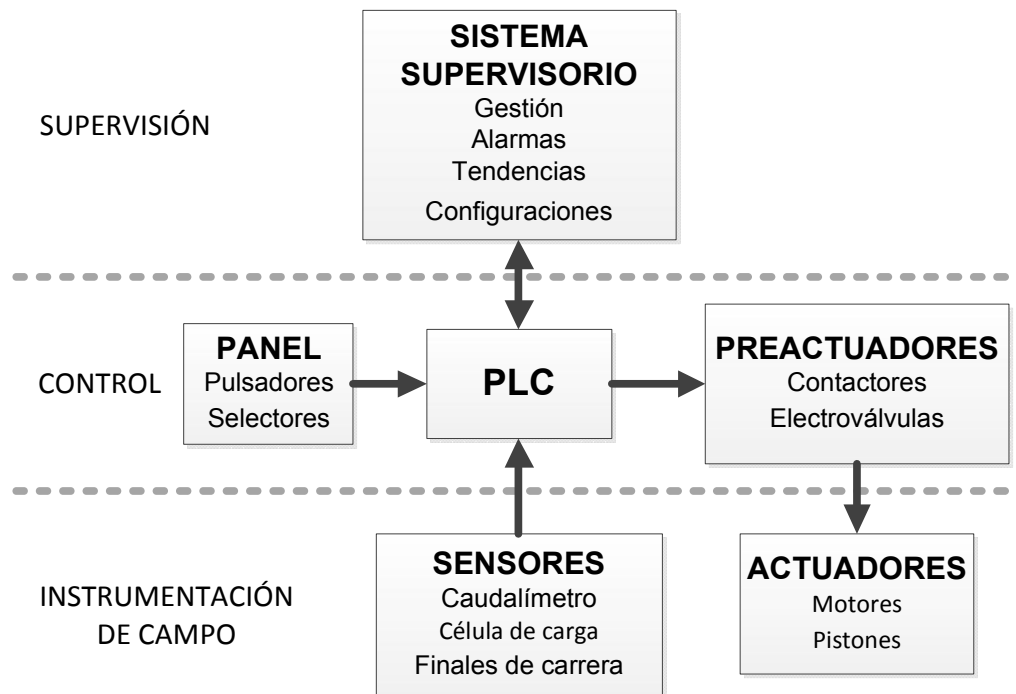


Figura 2.1 Estructura de sistema de automatización integrada propuesto.

El PLC seleccionado fue el versátil MODICON TSX 2732 de Telemecanique, una marca de Schneider Electric. Siendo una autómatas modular en su gama Micro adecuado para aplicaciones sencillas, con facilidades de mantenimiento y configuración, y tres conectores integrados: dos para contaje rápido y otro para ocho entradas y una salida analógicas. Posee dos puertos de comunicaciones para diálogo de operadores (modbus, unitelway o ASCII) y dos ranuras para la extensión de memoria y/o una tarjeta de comunicaciones (modbus plus, fipway, MODEM, etc).

Como se vio en el capítulo 1, el funcionamiento del sistema de automatización estará determinado por el software asociado a este, es decir, por el programa de su algoritmo de control y la reducción de errores se logra mediante el modelado formal del sistema.

2.3 Formalización del sistema de automatización integrada propuesto.

El término informal comprende todo lo que no se ha definido estrictamente, es decir, no esté bien definido tanto sintácticamente como semánticamente. Como se pudo constatar en el Capítulo 1, actualmente existe una búsqueda en la comunidad científica y académica de métodos formales para el modelado y diseño de sistemas de automatización, donde juega un papel importante la formalización, verificación y validación.

Seguendo la metodología de diseño sobre GHENeSys propuesta en [11], el primer aspecto a considerar en el **paso 1** es el estudio del proceso a controlar y sus requerimientos, que se corresponde con la fase de **especificaciones informales** típicas de las aplicaciones con PLCs.

Igualmente los **pasos del 2 al 5** de la metodología están estrechamente relacionados, y se asocian a la fase de **formalización** de las aplicaciones con PLCs, garantizando con ello la obtención del modelo idóneo. Estos pasos tienen como objetivo fragmentar el sistema en las menores unidades funcionales posibles, definir los niveles de jerarquía en la estructura del modelo e identificar sus interrelaciones entre los módulos (subredes) que la forman, correspondiéndose las unidades funcionales definidas con las subredes, tratando de representarlas en un tipo de OPN lo más sencilla posible, según lo permita la aplicación dada.

Siguiendo estos principios, el sistema puede ser descompuesto en la siguiente estructura modular jerárquica:

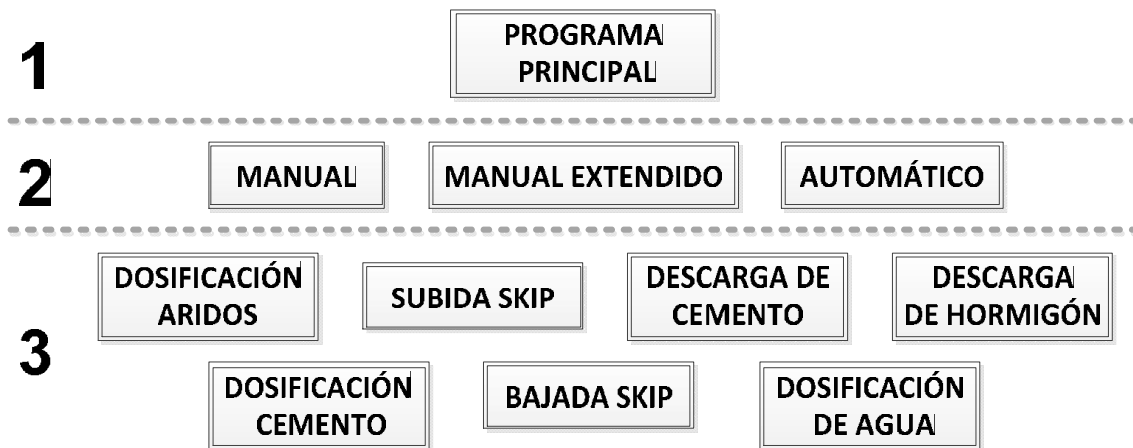


Figura2.2 Estructura modular jerárquica del sistema.

En esta estructura podemos ver que existen tres niveles de jerarquía, en el nivel más abstracto o primero se encontrará el módulo (red) correspondiente a la unidad funcional “Programa Principal” (cuya función básica está dirigida a la selección del estado y modo de funcionamiento del sistema). En el segundo nivel existirán tres módulos (subredes) correspondientes a las unidades funcionales “Manual” (cuya función es accionar los actuadores según los controles en el panel), “Manual Extendido” (cuya función difiere del manual exclusivamente en la función realizada al accionar los interruptores Arido1, Arido2, Arido3, Cemento y Agua) y “Automático” (cuya función es llevar la secuencia del proceso únicamente a partir de la información recibida por los sensores y estados internos de los módulos (subredes del primer nivel) empleados. Y por último, en el tercer nivel encontraremos los módulos (subredes del segundo) correspondientes a las unidades funcionales “Dosificación de Áridos”, “Dosificación de Cemento”, “Subida Skip”, “Bajada Skip”, “Descarga de Cemento”, “Dosificación de Agua” y “Descarga de Hormigón”, utilizadas en su conjunto dentro de la unidad funcional “Automático”.

Estructura que puede cambiar, pues vale destacar que resulta difícil desde una etapa primaria del diseño, prever la cantidad total de módulos, su nivel de jerarquía y sus interrelaciones, esto lo genera todo el estudio ulterior, depurándose y refinándose

gradualmente a medida que el proceso toma mayor alcance y profundidad (diseño Top-down y Bottom-up).

Como se puede apreciar aún desde esta etapa primaria, el proceso de Dosificación de Áridos consta con 11 módulos funcionales a modelar en PN. La razón de tantos modelos fue dada por nuestra intención de lograr la mayor estructura modular posible junto a las limitaciones propias del programa utilizado, en este caso el *Visual Object Net 2.7*(VON).

En lo adelante describiremos solo tres de estos modelos que por su importancia, complejidad y similitud con los demás creemos los más indicados para demostrar las ventajas que brindan las PN's como herramienta formal en el modelado, verificación y validación de los sistemas de automatización.

2.3.1 Modelado en GHENeSys de la unidad funcional “Programa Principal”.

Para realizar el modelado de la unidad funcional “Programa Principal” se identificaron los siguientes recursos y eventos controlables que se generan a partir de estos y que intervienen en su funcionamiento.

Recursos	Eventos
• Línea (pulsador) • Emergencia (pulsador) • Arranque/Parada (interruptor) • Manual-Manual Extendido-Automático (selector)	• Energizar elementos de fuerza • Desenergizar elementos de fuerza • Arrancar el sistema • Parar el sistema • Selección de Manual • Selección de Manual Extendido • Selección de Automático

A partir de del análisis de estos recursos y eventos y el cumplimiento de los requerimientos funcionales, según la metodología de modelado se crean los siguientes subsistemas:

Subsistema 1 (Rojo en Anexo 4.1)

Se propone inicialmente un subsistema equivalente a las estructuras de lenguajes de programación definidas en [28] que permite modelar si los elementos de fuerza están

energizados o no a través de los lugares **Línea ON y Línea OFF**, relacionados con la acción de los recursos Pulsador Línea y Pulsador Emergencia de la siguiente manera:

- El subsistema cambiará del estado Línea OFF a Línea ON si el pulsador Línea es presionado, y en sentido contrario si el pulsador Emergencia es presionado.

Subsistema 2 (Amarillo en Anexo 4.1)

Una red principal del tipo Libre Elección [18, 41] que permita modelar la selección de las funciones habilitar Manual, Manual Extendido, Automático y Parar a través de los lugares **ActivarM, ActivarME, ActivarA y Parar**, relacionadas con los recursos interruptor Arranque/Parada, selector Manual-ManualExtendido-Automático y el estado del lugar Línea ON de la siguiente manera:

- Inicialmente si el lugar Línea ON (elementos de fuerza energizados) está desactivado el sistema estará detenido completamente. Si se activa el lugar Línea ON y el interruptor Arranque/Parada está en posición “Parada” los elementos de fuerza se encontrarán energizados y la acción Parar (deshabilitar cualquier modo de operación activo) será ejecutada. Por el contrario, si el lugar Línea ON se mantiene activo y el interruptor Arranque/Parada se encuentra en posición “Arranque” los elementos de fuerza continuarán activos y se ejecutarán las acciones ActivarM, ActivarME o ActivarA (habilitar los modos Manual, MExtendido y Automático) según el estado del selector Manual-ManualExtendido-Automático.

Subsistema 3 (Verde en Anexo 4.1)

Permite modelar si el modo de operación Manual está activo o no a través de los lugares **Manual ON, Manual OFF**, relacionados con los estados ActivarM, ActivarME, ActivarA y Parar de la siguiente manera:

- El subsistema cambiará del estado Manual OFF a Manual ON y se mantendrá en él si el lugar ActivarM se habilita. Sólo pasará nuevamente al estado Manual OFF si se activan los lugares ActivarME, ActivarA o Parar.

Subsistema 4 (Verde en Anexo 4.1)

Permite modelar si el modo de operación Manual Extendido está activo o no a través de los lugares **MExtendido ON**, **MExtendido OFF**, relacionados con el los estados ActivarM, ActivarME, ActivarA y Parar de la siguiente manera:

- El subsistema cambiará del estado MExtendido OFF a MExtendido ON y se mantendrá en él si el lugar ActivarME se activa, solo pasará nuevamente al estado MExtendido OFF si se activan los lugares ActivarM, ActivarA o Parar.

Subsistema 5 (Verde en Anexo 4.1)

Permite modelar si el modo de operación Automático o no a través de los lugares **Automático OFF y Automático ON**, relacionados con el los estados ActivarM, ActivarME, ActivarA y Parar de la siguiente manera:

- El subsistema cambiará del estado Automático OFF a Automático ON y se mantendrá en él si el lugar ActivarA se activa, solo pasará nuevamente al estado Automático OFF si se activan los lugares ActivarM, ActivarME o Parar.

Subsistema 6 (Azul en Anexo 4.1)

Permite modelar si el sistema está parado, en funcionamiento o en emergencia a través de los lugares **Sistema Parado**, **Sistema Funcionando y Sistema en Emergencia**, relacionados con los estados de los macro lugares Manual ON, MExtendido ON, Automático ON y el estado del lugar Línea OFF de la siguiente manera:

- El sistema se encontrará en el estado Sistema Parado si ninguno de los estados Manual ON, MExtendido ON y Automático ON está activo y pasará a funcionando solo cuando uno de ellos cualesquiera se activado. Una vez funcionando pasará a emergencia si el estado Línea OFF se activa.

Del análisis de las diferentes relaciones entre la red principal y los subsistemas propuestos se obtuvo el modelo de la Anexo 4.1 correspondiente a la unidad funcional “Programa Principal”.

Donde los macrolugares Manual ON, MExtendido ON y Automático ON significan la habilitación a este nivel de las unidades funcionales “Manual”, “Manual Extendido” y “Automático”.

El macro lugar inicio en este caso responde a una nueva unidad funcional donde se realizaran en cada ciclo tareas como: las asignaciones de las entradas físicas del PLC a bits y palabras de memoria, las asignaciones de bits de memoria a las salidas físicas del PLC, la adquisición y comparación de los valores analógicos correspondientes al peso, la contabilización y comparación de los pulsos correspondientes a la cantidad de agua, y la contabilización de los tiempos de descarga arido1, arido2 y arido3.

2.3.2 Modelado en GHENeSys de la unidad funcional “Automático” presente en el programa Principal.

Igualmente para realizar el modelado de la unidad funcional “Automático” se identificaron los siguientes recursos y eventos controlables que se generan a partir de estos, y que intervienen en su funcionamiento.

Se debe tener en cuenta que en este nivel jerárquico la mayoría de los recursos que garantizan la secuencia correcta de los eventos no son en realidad recursos físicos (entradas al PLC de sensores, botones, pulsadores), sino estados almacenados (estados o marcas internas en la memoria del PLC) modificados usualmente dentro de los módulos en dependencia del completamiento de sus funciones específicas y por los temporizadores que definen las actividades marcadas por el tiempo. Así pues el tránsito hacia un macrolugar se asocia a la terminación de la función realizada en un macrolugar previo o por el término de una operación anterior temporizada.

Recursos

Eventos

- Sistema en Emergencia (estado)
- Dosificación Áridos terminada (estado)
- Dosificación Cemento terminada (estado)
- Subida *Skip* terminada (estado)
- Descarga Áridos terminada (estado)
- Bajada *Skip* terminada (estado)
- Descarga Cemento terminada (estado)
- Dosificación Agua terminada (estado)
- Mezclado terminado (estado)
- Descarga Hormigón terminada (estado)
- Sonda-Continuo (selector)
- Sonda (sensor)
- MM (motor mezcladora)
- Fallo MM (relé térmico)
- Dosificar Cemento
- Dosificar Áridos
- Subir *Skip*.
- Descargar Áridos
- Bajar *Skip*.
- Descargar Cemento
- Dosificar Agua
- Mezclar materiales
- Descargar Hormigón
- Parar Mezcladora

A partir de aquí se proponen los siguientes subsistemas:

Subsistema 1

Una red principal que permita modelar la secuencia del funcionamiento del sistema de control automatizado a través de los lugares y macrolugares **Dosificación Áridos, Dosificación Cemento, Subida *Skip*, Descarga Aridos, Bajada *Skip*, Descarga Cemento, Dosificación Agua, Mezclado, Descarga Hormigón y Parar Mezcladora**, relacionados con los estados Sistema en Emergencia, Dosificación Áridos terminada, Dosificación Cemento terminada, Subida *Skip* terminada, Descarga Áridos terminada, Bajada *Skip* terminada, Dosificación Agua terminada, Mezclado terminado, Descarga Hormigón terminada y el estado de los recursos selector Sonda-Continuo y sensor Sonda de la siguiente manera:

- El sistema funcionará en primer lugar si está activo el macro lugar “Automático” dentro de la red que modela la unidad funcional “Programa Principal”, estableciendo las condiciones iniciales en el macro lugar “Inicio”, y comenzando automáticamente con las funciones de los macro lugares Dosificación Áridos y Dosificación Cemento, una vez terminadas estas funciones, se activará el macrolugar Subida *Skip* (debe encender la Mezcladora), terminada la subida, se activará el lugar Descarga Áridos por un

tiempo determinado (tiempo de vaciado del *Skip*) a partir del cual se activan entonces los macro lugares Bajada *Skip* y Descarga Cemento. Habiendo terminado Bajada *Skip* se activará nuevamente el macrolugar Dosificación Áridos, y por otro lado, la terminación de Descarga Cemento activará el macrolugar Dosificación Agua y nuevamente el macrolugar Dosificación Cemento. El ciclo del hormigón continuará entonces cuando termine Dosificación Agua con la activación del lugar Mezclado por un tiempo determinado (tiempo de amasado) a partir del cual se activará el macrolugar Descarga Hormigón. Una vez terminada Descarga Hormigón se volverá a comenzar el ciclo si el selector Sonda-Continuo se encuentra en la posición Continuo, de lo contrario, si está en la posición Sonda, se activará el lugar Parar Mezcladora y el ciclo se repetirá cuando del sensor Sonda se encuentre en estado desactivado (cajón de la prensa vacío).

Subsistema 2

Un subsistema que similar a la propuesta en los trabajos de Uzam [55], permita modelar dos o tres estados en una máquina, en este caso los estados encendido, apagado y en fallo del motor de la mezcladora (MM), debido a que su funcionamiento se define en varios niveles de jerarquía en instantes diferentes del ciclo de producción, amerita ser modelado aquí.

Este subsistema modelará entonces si el motor de la mezcladora estará Apagado, Encendido o en Fallo a través de los estados **MM OFF**, **MM ON**, **MM fallo**, relacionado con los estados del recurso Fallo MM, del macrolugar Subida *Skip* y del lugar Parada Mezcladora de la siguiente manera:

- El subsistema pasa del estado MM OFF a MM ON si se activa el macrolugar Subida *Skip*; del estado MM ON a MM fallo si se activa el recurso relé térmico Fallo MM (sobreconsumo de corriente del motor de la mezcladora), volviendo al estado MM ON una vez resuelto el problema; y del estado MM ON a MM OFF si se activa el lugar Parada Mezcladora.

Del análisis de las diferentes relaciones e interrelaciones de los subsistemas propuestos y la red principal se obtuvo el modelo del Anexo 4.2 correspondiente a la unidad funcional

“Programa Principal”, donde se pueden apreciar características comunes de los sistemas dinámicos de eventos discretos como la concurrencia y sincronización de actividades.

Luego del análisis de las relaciones entre la red principal y el subsistema 2 se tomó en consideración también que la ocurrencia de un fallo en el motor de la mezcladora afectaría el ciclo productivo a este nivel de las siguientes maneras:

- Si el fallo ocurre durante el tiempo de Descarga Áridos se debe pausar el tiempo, permitiendo si el problema se resuelve que transcurra el tiempo faltante.
- Si el fallo ocurre durante el tiempo de Mezclado se debe pausar el tiempo, permitiendo si el problema se resuelve que transcurra el tiempo faltante.
- Si el fallo ocurre después de que el macrolugar Descarga Hormigón termina y el selector Sonda-Continuo se encuentra en la posición Continuo, se debe detener la secuencia antes de que se active macro lugar Subir *Skip*.

De la misma manera, si se activara el estado sistema en Emergencia ocurrirán las mismas afectaciones anteriores y las necesarias dentro de cada macrolugar correspondiente a una unidad funcional.

2.3.3 Modelo en GHENeSys de la unidad funcional “Dosificación Áridos” presente en el macro lugar “Automático ON”.

Siguiendo los pasos utilizados para obtener el modelo de las unidades funcionales “Programa Principal” y “Automático”, se identificaron los siguientes recursos y eventos controlables generados a partir de estos, y que intervienen en el funcionamiento de la unidad funcional “Dosificación de Áridos”.

Recursos

Eventos

- Skip Abajo (sensor)
- FMA1(relé térmico)
- MA1 (motor árido1)
- Caudal A1 (sensor)
- FMA2(relé térmico)
- MA2 (motor árido2)
- Caudal A2 (sensor)
- FMA3 (relé térmico)
- MA3 (motor árido3)
- Caudal A3 (sensor)
- MC (motor cinta)
- FMC (relé térmico)
- Dosificación Áridos Terminada (estado)
- Sistema en Emergencia (estado)
- Encender cinta áridos
- Pausar tiempo cinta áridos
- Encender alimentador1
- Pausar tiempo alimentador1
- Encender alimentador2
- Pausar tiempo alimentador2
- Encender alimentador3
- Pausar tiempo alimentador3
- Apagar alimentador1
- Apagar Parar alimentador2
- Apagar Parar alimentador3
- Apagar cinta áridos

A partir de aquí se proponen los siguientes subsistemas:

Subsistema 1

Una red principal que permita modelar la secuencia de encendido y apagado de los motores correspondientes a la cinta transportadora de áridos y los alimentadores de los áridos1, 2 y 3a través de los lugares **CintaON, CintaOFFAA1ON,AA1OFF, AA2ON,AA2OFF, AA3ON yAA3OFF**, relacionados con el estado de los recursos sensor *Skip Abajo*, *Caudal A1*, *Caudal A2* y *Caudal A3*, los estados almacenados *DosificaciónÁridos Terminada* y *Sistema en Emergencia*, y los estados *MCony MCfallo(encendido y fallo del motor de la cinta de áridos)* de la siguiente manera:

- El sistema funcionará en primer lugar si está activo el macro lugar “*Dosificación Áridos*” dentro de la red que modela la unidad funcional “*Automático*” y pasará desde un estado Inicio (desactivar estado *Bajada Skip Terminada*) a activar el lugar **CintaON** (encender la cinta de áridos) cuando esté activo el recurso sensor *Skip Abajo* y desactivado el estado *Dosificación Áridos Terminada*; una vez que se detecte el estado activo del lugar **MCon** (motor cinta de áridos encendido) los lugares **AA1ON, AA2ON** y

AA3ON (encender alimentador1, alimentador 2 y alimentador3 respectivamente) se activarán cada uno por un tiempo determinado (tiempo de áridos1, tiempo de áridos 2 y tiempo de áridos 3) que podrá ser pausado por el estado inactivo de los recursos sensor Caudal A1, Caudal A2 y Caudal A3(detectores de salida de arido1, 2 y 3 de las tolvas correspondientes) o por el estado activo de los recursos relé térmico FMA1, FMA2 y FMA3 (sobreconsumo de los motores de los alimentadores 1, 2 y 3) respectivamente, y de manera global por los estados Sistema en Emergencia activo o MCfallo activo (fallo en motor de la cinta de áridos). Transcurrido los tiempos para cada árido se activarán los lugares AA1OFF, AA2OFF y AA3OFF(apagar alimentador1, alimentador 2 y alimentador3 respectivamente) según corresponda, y una vez todos los alimentadores apagados, se activará el lugar CintaOFF (apagar la cinta de áridos y activar estado Dosificación Áridos Terminada)transcurrido el tiempo de vaciado de la cinta.

Subsistema 2

Un subsistema que modelará si el motor de la cinta de áridos está apagado, encendido o en fallo a través de los estados **MCoff**, **MCon** y **MCfallo**, relacionado con los estados del recurso relé térmico FMC, de los lugares CintaON y CintaOFF de la siguiente manera:

- El subsistema pasa del estado MCoffa MCon si se activa el lugar CintaON; del estado MCon a Mcfallosi se activa el recurso relé térmico FMC (sobreconsumo de corriente del motor de la cinta de áridos),volviendo al estado MCon una vez resuelto el problema; y del estado MCon a MCoff si se activa el lugar CintaOFF.

Subsistema 3

Un subsistema que modelará si el motor del alimentador1 está apagado, encendido o en fallo a través de los lugares **MA1off**, **MA1on** y **MA1fallo**, relacionado con los estados del recurso relé térmico FMA1, de los lugares MCon, AA1ON, MCfallo y AA1OFF de la siguiente manera:

- El subsistema pasa del estado MA1off a MA1on si se activan los lugaresAA1ON y MCon; del estado MA1on a MA1fallosi se activa el recurso relé térmico FMA1

(sobreconsumo de corriente del motor del alimentador1), volviendo al estado MA1on una vez resuelto el problema; y del estado MCon a MCoff si se activa el lugar AA1OFF o el lugar MCfallo.

Se proponen además otros dos subsistemas idénticos al anterior en funcionamiento y en relaciones con la red principal y el subsistema que modela los estados del motor de la cinta de áridos pero referidos a los alimentadores 2 y 3.

En el Anexo 4.3 se puede observar el modelo obtenido a partir de las relaciones y dependencias de la red principal propuesta y los subsistemas modelados.

2.4 Verificación.

Como fue explicado en el Capítulo 1, sin un profundo análisis y verificación, es muy difícil definir el origen de los errores de diseño. Debido a esto, en este epígrafe, siguiendo los pasos 6, 7 y 8 de la metodología [11], haremos la verificación de algunas de las principales propiedades estructurales y funcionales los modelos descritos anteriormente por dos de los métodos mencionados en el Capítulo1: el método gráfico aplicando técnicas de reducción o descomposición, y el método analítico a través de la matriz de incidencia.

Debe valorarse que ya desde los pasos iniciales de análisis y diseño se tuvieron en cuenta las 7 guías para modelado de procesos definidas en [40], lo cual ya reduce la posibilidad de errores de diseño en el modelo.

2.4.1 Verificación por el método de técnicas de reducción o descomposición.

Los modelos obtenidos anteriormente están conformados por redes principales que definen la secuencia de las operaciones necesarias para realizar determinada función en los niveles jerárquicos dos y tres, y la libre selección de los posibles modos de operación y estado del sistema en el primer nivel jerárquico. Además de subsistemas que modelan tres estados en el caso de los motores y el comportamiento funcional de la planta y dos lugares en el caso de los sistemas alternativos de los modos de operación.

El análisis de los subsistemas de dos o tres lugares resulta trivial, es por eso que a continuación se analizan solamente las redes principales que conforman esos modelos por la

importancia que revisten aplicando las seis transformaciones más simples que pueden ser usadas en el análisis de vivacidad, seguridad y limitación definidas en [41].

2.4.1.1 Reducción de la PN principal correspondiente al modelado no controlado del Programa Principal.

En la Anexo 5.1 se puede observar la PN principal correspondiente al modelo no controlado del Programa Principal.

En este caso podemos observar que los pares de transiciones T4-T13, T5-T12, T6-T10 y T25-T26 pueden fusionarse en una sola transición aplicando la regla **FST**. Para no poner nombres demasiado largos en las transiciones al fusionar los pares antes mencionadas simplemente se dejará como nombre de las transiciones resultantes el de las primeras transiciones de cada par, quedando la red de la manera en que se muestra en el Anexo 5.2 (a).

Luego la red resultante puede ser reducida nuevamente a la mostrada en el Anexo 5.2 (b) una vez que se fusionan las transiciones T4, T5, T6 y T25 aplicando la regla **FPT** en la transición T4.

De una manera simple podemos ver que la red resultante es viva, limitada y segura [41], por lo que la red principal correspondiente al modelo no controlado del Programa Principal también lo es. Esto garantiza la ausencia de bloqueos en el modelo y el número constante de una marca sea cual fuere el marcado.

2.4.1.2 Reducción de la PN principal correspondiente al **modelado del macrolugar Automático.**

A partir de la PN principal correspondiente al modelo no controlado del macro lugar Automático que se muestra en el Anexo 6.1 se puede llegar a la red mostrada en el Anexo 6.2 (a) aplicando las reglas:

1. **FST** entre los pares de transiciones T8-T11 y T6-T5 obteniéndose como transiciones resultantes T8 y T6.
2. **FPT** entre la transición T10 y T8 obteniéndose como transición resultante T10.

3. **FST** entre la transición T10 y T6. obteniéndose cómo transición resultante T10.
4. **FSP** entre los lugares P2 y P3. Quedando cómo lugar resultante P2.

Luego de la red mostrada en el Anexo 6.2 (a) se puede llegar a la red mostrada en el Anexo 6.2 (b) aplicando las reglas:

1. **FST** entre la transición T4 y T10 obteniéndose cómo transición resultante T4.
2. **FST** entre la transición T4 y T3 obteniéndose cómo transición resultante T4.

Así mismo de la red mostrada en el Anexo 6.2(b) se puede llegar a la red mostrada en el Anexo 6.3(a) aplicando las reglas:

1. **FST** entre la transición T4 y T7 obteniéndose cómo transición resultante T4.
2. **FST** entre la transición T1 y T4 obteniéndose cómo transición resultante T1.

Finalmente se llega a la red mostrada en el Anexo 6.3 (b) a partir del Anexo 6.3 (a) después de haber eliminado los lugares P9 y P10 aplicando la regla de **ESP**.

La red resultante demuestra al igual que en el caso anterior que la red principal correspondiente al modelo no controlado del macro lugar Automático es viva, limitada y segura. Garantizando la ausencia de bloqueos y el número constante de una marca sea cual fuere el marcado.

2.4.1.3 Reducción de la red correspondiente al macro lugar Dosificación Áridos.

Cómo en los dos casos anteriores se toma la red principal correspondiente al modelo no controlado del sistema objeto de estudio y se reduce aplicando las mismas reglas.

En este caso la red mostrada en el Anexo 7.1 (b) se obtiene a partir de la red mostrada en el Anexo 7.1 (a) aplicando las reglas:

1. **FSP** a los pares de lugares P16-P19, P15-P12 y P3-P11 obteniéndose como lugares resultantes P16, P15 y P3.
2. **FST** a las transiciones T7 y T15 obteniéndose como transición resultante T15.

Luego la red mostrada en el Anexo 7.2 (a) es obtenida a partir de la red en el Anexo 7.1 (b) al aplicar la regla de **FPP** a los lugares P16, P15 y P3 obteniéndose como lugar resultante P15.

A partir de esta red la reducción hasta la red mostrada en el Anexo 7.2 (b) y posteriormente (c) se debe únicamente a aplicar la regla de **FST** en primer lugar a las transiciones T14 y T7, quedando como resultante T14, y luego a las transiciones T1 y T14 quedando como resultante T1.

La red resultante entonces demuestra que la red principal correspondiente al modelo no controlado del macro lugar Dosificación Áridos es viva, limitada y segura. Garantizando la ausencia de bloqueos y el número constante de una marca sea cual fuere el marcado.

2.4.2 Verificación aplicando el enfoque de la matriz de incidencia.

En este caso, aunque el sistema es pequeño, lo que permite aplicar sólo las técnicas de reducción, se aplicará la verificación mediante la matriz de incidencia para demostrar la efectividad del método y el cumplimiento de algunos requerimientos funcionales del sistema.

2.4.2.1 Cálculo de invariantes para la red correspondiente al programa Principal.

Con la ayuda de la herramienta de modelado PIPE, se realizó la verificación por el enfoque de matriz de incidencia de la red del programa principal (Anexo 4.1), obteniéndose la siguiente matriz e invariantes:

Matriz de incidencia

	T4	T5	T6	T25	T13	T12	T10	T26
P19	0	0	0	1	0	0	0	-1
P1	-1	-1	-1	-1	1	1	1	1
P11	0	0	1	0	0	0	-1	0
P12	0	1	0	0	0	-1	0	0
P13	1	0	0	0	-1	0	0	0

Invariantes de transición (T)

T4	T5	T6	T25	T13	T12	T10	T26
1	0	0	0	1	0	0	0
0	1	0	0	0	1	0	0
0	0	1	0	0	0	1	0
0	0	0	1	0	0	0	1

La red está cubierta por invariantes positivas de transición, por lo que se puede considerar limitada y viva.

Esto demuestra que todas las transiciones de la red son habilitables en algún momento de la dinámica del sistema (ausencia de bloqueos) y que todas las transiciones se habilitan sólo una vez en cada ciclo del funcionamiento del sistema (limitación estructural 1). Cada una de las invariantes T (filas de la tabla anterior) tienen sólo dos transiciones de la red principal habilitadas una vez que corresponden a los cuatro caminos alternativos de dicha red.

La interpretación física es que todos los elementos de la red principal están, al menos, en un estado activo (vivos) sin bloquearse ni mantenerse en lazos indefinidos porque siempre se habilita la transición siguiente. Por tanto se garantiza la operación de forma eficiente y segura como se solicita en los requerimientos funcionales del sistema. Además se verifica que se encuentre en sólo uno de los cuatro estados posibles (3 modos de operación o parado), porque las invariantes T independizan los caminos alternativos.

Invariantes de lugar (P).

P1	P11	P12	P13	P19
1	1	1	1	1

La red está cubierta por invariantes positivas de lugar, por lo que se considera limitada.

Esto demuestra que el proceso de la red principal se ejecutará secuencialmente y que al menos uno de los estados siempre estará habilitado con un marcado máximo de un token, lo cual se acostumbra a representar mediante la siguiente ecuación:

$$M(P19) + M(P1) + M(P11) + M(P12) + M(P13) = 1$$

La interpretación física de esta invariante es que el sistema siempre estará en uno de sus tres modos de operación o parado, lo cual corresponde al cumplimiento del requerimiento funcional de los tres modos de operación.

2.4.2.2 Cálculo de invariantes para la red correspondiente al macro lugar Automático.

Con la ayuda de la herramienta de modelado PIPE, se realizó la verificación por matriz de incidencia de la red del programa de Automático (Anexo4.2), obteniéndose la siguiente matriz e invariantes:

Matriz de incidencia

	T1	T2	T3	T4	T5	T6	T7	T8	T10	T11
P1	-1	0	0	0	0	0	0	1	1	0
P2	1	-1	0	0	0	0	0	0	0	0
P3	0	1	-1	0	0	0	0	0	0	0
P4	0	0	1	-1	0	0	0	0	0	0
P5	0	0	0	1	-1	0	0	0	0	0
P6	0	0	0	0	1	-1	0	0	0	0
P7	0	0	0	0	0	1	0	0	-1	-1
P8	0	0	1	0	0	0	-1	0	0	0
P9	-1	0	0	0	0	0	1	0	0	0
P10	-1	0	0	1	0	0	0	0	0	0
P11	0	0	0	0	0	0	0	-1	0	1

Invariantes de transición (T)

T1	T2	T3	T4	T5	T6	T7	T8	T10	T11
1	1	1	1	1	1	1	0	1	0
1	1	1	1	1	1	1	1	0	1

La red está cubierta por invariantes positivas de transición, por lo que se puede considerar limitada y viva.

Esto demuestra que todas las transiciones de la red principal son habilitables en algún momento de la dinámica del sistema (ausencia de bloqueos) y que todas las transiciones se habilitan sólo una vez en cada ciclo del funcionamiento del sistema (limitación estructural 1). Las dos invariantes definen los dos posibles caminos alternativos de la secuencia de automático

La interpretación física es que todos los elementos de la red principal de automático están, al menos, en un estado activo (vivos) sin bloquearse ni mantenerse en lazos indefinidos. Por tanto se garantiza la operación automática de forma eficiente y segura como se solicita en los requerimientos funcionales del sistema. Los dos caminos alternativos son el ciclo que se realiza de forma continua o el que se repite en dependencia del sensor Sonda, según el selector Sonda/Continuo.

Invariantes de lugar (P)

P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
0	1	1	0	0	0	0	1	1	0	0
0	1	1	1	0	0	0	0	0	1	0
1	1	1	1	1	1	1	0	0	0	1

La red está cubierta por invariantes positivas de lugar, por lo que se considera limitada.

Esto demuestra que el proceso de la red principal de automático se ejecutará secuencialmente en sus tres lazos de persistencia de tokens (ver los componentes de cada fila de la tabla anterior) y que al menos uno de los estados siempre estará habilitado con un marcado máximo de un token, lo cual se acostumbra a representar mediante las siguientes ecuaciones:

$$M(P2) + M(P3) + M(P8) + M(P9) = 1$$

$$M(P2) + M(P3) + M(P4) + M(P10) = 1$$

$$M(P1) + M(P2) + M(P3) + M(P4) + M(P5) + M(P6) + M(P7) + M(P11) = 1$$

La interpretación física de esta invariante es que el sistema automático realiza de forma paralela la dosificación de áridos y de cemento con la subida y descarga del Skip (adicionando la descarga de cemento solo en la dosificación de cemento), de forma a estar nuevamente preparados para el siguiente ciclo automático.

Esto demuestra el cumplimiento de los requerimientos funcionales que referidos a:

1. que la dosificación de cemento sólo se realiza si se ha descargado la tolva de cemento (ecuación de invariante 2). Esto significa que el subsistema de dosificación de cemento

estará en uno de sus tres estados: dosificando cemento, esperando por la subida y descarga del Skip o descargando cemento

2. que se realiza la dosificación de áridos si ya se ha bajado el Skip (ecuación de invariante 1). Esto significa que el Skip sólo puede estar en uno de sus estados subiendo, bajando, arriba (descarga de áridos) o abajo (Dosificación de áridos).
3. Además se cumple el requerimiento de que para reiniciar el ciclo debe estar realizada la dosificación de áridos, dosificación de cemento y la descarga de hormigón del ciclo anterior (ecuación de invariante 3)

2.4.2.3 Cálculo de invariantes para la red correspondiente al macro lugar Dosificación de Áridos.

Con la ayuda de la herramienta de modelado PIPE, se realizó la verificación por matriz de incidencia de la red del programa de dosificación de áridos (Anexo 4.3), obteniéndose la siguiente matriz e invariantes:

Matriz de incidencia

	T1	T14	T8	T7	T13	T6	T15
P1	-1	0	0	0	0	0	1
P2	1	-1	0	0	0	0	0
P3	0	1	0	0	0	-1	0
P11	0	0	0	-1	0	1	0
P15	0	1	-1	0	0	0	0
P12	0	0	1	-1	0	0	0
P16	0	1	0	0	-1	0	0
P19	0	0	0	-1	1	0	0
P23	0	0	0	1	0	0	-1

Invariantes de transición (T)

T1	T14	T8	T7	T13	T6	T15
1	1	1	1	1	1	1

La red está cubierta por invariantes positivas de transición, por lo que se puede considerar limitada y viva.

Esto demuestra que todas las transiciones de la red principal de dosificación de áridos son habilitables en algún momento de la dinámica del sistema (ausencia de bloqueos) y que dichas transiciones se habilitan sólo una vez en cada ciclo del funcionamiento del sistema de dosificación de áridos (Limitabilidad estructural 1). Existe sólo una invariante T (sólo una fila en la tabla anterior) porque no existen caminos alternativos en este ciclo de dosificación de áridos, es decir que todas las transiciones de esta red principal tienen que dispararse una vez en cada ciclo.

La interpretación física es que todos los elementos de la red principal están, al menos, en un estado activo (vivos) sin bloquearse ni mantenerse en lazos indefinidos porque siempre se habilita la transición siguiente. Por tanto se garantiza la operación de forma eficiente y segura como se solicita en los requerimientos funcionales del sistema. Además se verifica que el inicio de los alimentadores 1, 2 y 3 se ejecutan en cada ciclo de forma paralela.

Invariantes de lugar (P)

P1	P2	P3	P10	P11	P14	P15	P18	P22
1	1	1	1	0	0	0	0	1
1	1	0	0	0	0	1	1	1
1	1	0	0	1	1	0	0	1

La red está cubierta por invariantes positivas de transición, por lo que se puede considerar limitada y viva.

Esto demuestra que el proceso de la red principal se ejecutará secuencialmente por tres caminos paralelos y que los estados de cada camino se ejecutan al menos una vez en cada ciclo con un marcado máximo de un token, lo cual se acostumbra a representar mediante las siguientes ecuaciones:

$$M(P1) + M(P2) + M(P3) + M(P10) + M(P22) = 1$$

$$M(P1) + M(P2) + M(P15) + M(P18) + M(P22) = 1$$

$$M(P1) + M(P2) + M(P11) + M(P14) + M(P22) = 1$$

La interpretación física de esta invariante es que el sistema siempre ejecutará los tres caminos de forma paralela porque luego de habilitarse P1 y P2 en cada ecuación, se ejecutan los tres caminos y se cierran todos con P22.

2.5 Validación.

Se comprobó el comportamiento funcional de cada módulo en todas sus variantes (funcionamiento normal, ante fallos, etc) demostrándose el cumplimiento de los requerimientos funcionales mediante su interpretación física.

Se comprueba la evolución funcional de los estados de la red de acuerdo a la habilitación o deshabilitación de las transiciones controlables según los valores de los lugares auxiliares (pseudoboxes) que representan los estados de los sensores, selectores, botones o marcas internas del proceso y del conteo de los tiempos en las transiciones temporizadas.

Para todos los casos se perfeccionó el funcionamiento de los modelos permitiendo la última versión de los mismos que se presenta en este trabajo. Esto se realiza según los pasos 9 y 10 de la metodología [11] que indica la simulación para comprobar todo el funcionamiento de los requerimientos funcionales y luego la reclasificación y verificación de propiedades de cada modelo.

2.6 Traducción y programación.

A partir de los modelos en PN se crearon los programas en LD con ayuda del software PL7Pro de Schneider Electric. Para esto se tuvo en cuenta la compatibilidad de las redes de Petri jerárquicas extendidas GHENeSys y el lenguaje LD IEC61131 compatible explicada en el capítulo 1.

En la figura 2.13 se representa la estructura de la programación elaborada en LD del PL7Pro, donde aparecen todos los programas y subrutinas que la componen y su nivel de relación.

La compatibilidad con los módulos propuestos en la Figura 2.2 de la estructura modular jerárquica del sistema es casi total, sólo fueron adicionados los dos módulos de inicialización del programa principal y del programa automático que aparecen modelados en la etapa de formalización al inicio de cada uno de los modelos de estos módulos, pero por su importancia

fueron separados para una mejor ejecución de la programación y sus posibilidades de mejoras y expansión con alta modularidad.

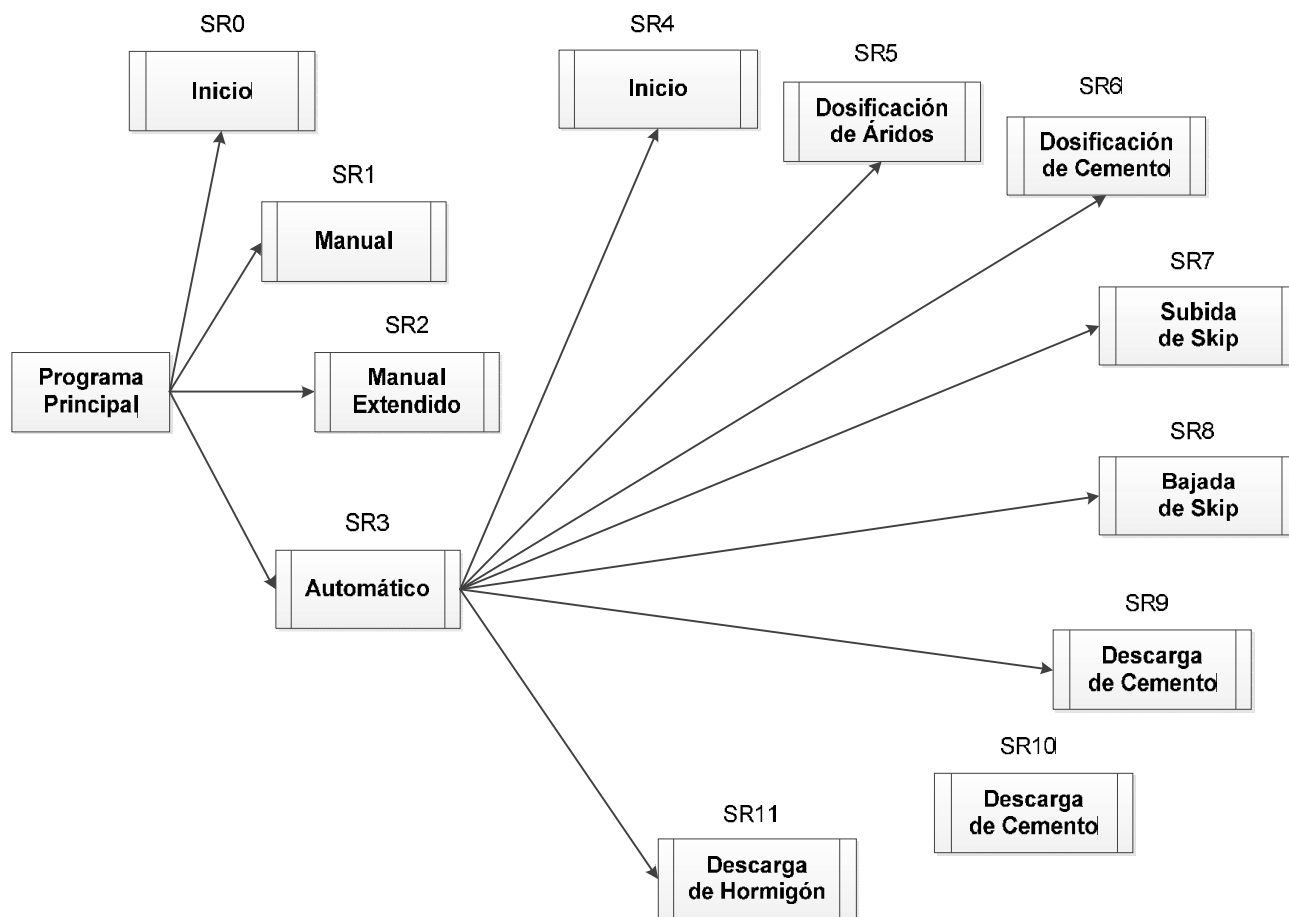


Figura 2.13 Estructura de la programación elaborada en LD del PL7Pro.

A continuación se explican algunos de los programas de los módulos donde se demuestran las analogías del modelado y la programación

2.6.1 Programación del módulo programa principal.

En el Anexo 8.1 se representa todas las líneas del programa principal en LD del PL7Pro.

La primera línea representa la ejecución en cada ciclo de una llamada a la subrutina SR0 correspondiente al lugar inicio, donde se relacionan los valores de las entradas y salida físicas del PLC con los bits y palabras de memoria respectivos, así como la contabilización de los tiempos de descarga áridos explicados en el macro lugar inicio del punto 2.3.1.

Las líneas 2 y 3 programan los cambios de estado del subsistema 1 y 6 del programa principal explicado en el punto 2.3.1.

Las línea con derivaciones final del programa el subsistema 2 de la red de libre selección explicada en el punto 2.3.1.

2.6.2 Programación del módulo Automático.

En el Anexo 8.2 aparece el programa de funcionamiento automático.

Las primeras cuatro líneas son los llamados a subrutinas SR3, SR5, SR6 y Sr7 que corresponden a los macrolugares P1 (Inicio), P9 (Dosificación de áridos), P10 (Dosificación de cemento) y P2 (Subir el Skip). Estos son los macrolugares donde se inicia la ejecución del automático en el modelo (Anexo 4.2).

El siguiente lugar en la secuencia de automático del Anexo 4.2 es el P3 de descarga de áridos, que sólo es una espera al tiempo de descarga en la transición temporizada de salida T3. Debido a esto, la siguiente línea del programa automático es la parametrización del temporizador %T3 (Tdeard: Tiempo de descarga de áridos)

Al terminar la descarga de áridos se activan las subrutinas SR8 y SR9 que corresponden a los macrolugares P4 (descarga de cemento) y P8 (bajada del Skip) en el modelo del Anexo 4.2.

Al terminar la SR9 se activa la SR10 que corresponde al macrolugar P5 (dosificación de agua).

Al terminar la descarga del agua dosificada se inicia el conteo del tiempo de mezclado en la transición de salida del lugar P6 (transición T6). De esta forma se realiza en el programa la parametrización del temporizador %T4 (Tmescl).

Al terminar el mezclado se activa la subrutina SR11 que corresponde al macrolugar P7 de descarga de hormigón.

Al terminar la descarga activa una línea de programa que si está el selector en continuo (B_cont, %M23) se activa el estado marca E_restar que corresponde al estado del lugar P1

(Inicio) a partir del segundo ciclo del programa. En el caso de que este en sonda, se realiza o no el reinicio del ciclo automático según el estado del sensor sonda en el cajón de lamprensa.

2.6.3 Programación del módulo Dosificación de áridos.

En el Anexo 8.3 aparece el programa de dosificación de áridos.

La primera línea corresponde al encendido de la cinta de áridos si se cumple que el Skip está abajo, el sistema no está en emergencia, no hay fallo en el motor de la cinta de áridos y no ha culminado el tiempo de funcionamiento de la cinta de árido luego de culminar dosificación de los áridos 1, 2 y 3. Esto corresponde con la activación del lugar P2 en el inicio modelo de dosificación de áridos mediante el disparo de la transición T1

En el modelo, al activarse el lugar P2 se habilita la transición T14 que permite la activación de los tres caminos paralelos de dosificación de los tres tipos de áridos. Esto corresponde en el programa a la activación de las marcas internas %M55, %M56 y %M57 (O_ard1, 2 y 3) que son las salidas a las motores de los alimentadores de los tres áridos.

Al final de las tres dosificaciones se activa el tiempo de movimiento de la cinta de áridos que a la transición temporizada T7.

2.7 Diseño del Supervisorio.

Con vista a mejorar la operación del sistema de Dosificación de Áridos completando la pirámide típica de la automatización se desarrolla el programa de supervisión de todo el sistema mediante el software Eros. Este consta de las siguientes ventanas (Anexo 9):

1. Principal (Mímico de operación en sus tres estados de funcionamiento).
2. Configuración (Permite ajustar la receta de dosificación).
3. Tendencias (gráficas de comportamiento en el tiempo de la operación del sistema)
4. Tabla de registro de alarmas (estadísticas de fallos y alarmas del sistema)

Esto completa el cumplimiento de los requisitos funcionales exigidos en el sistema.

2.8 Verificación y validación a nivel de laboratorio.

El software PL7 PRO v4.4 propio del fabricante y del autómatas seleccionado no permite la simulación, así pues todos los programas confeccionados se probaron descargándolos directamente al PLC y comprobando el correcto funcionamiento de los mismos ante diferentes escenarios posibles en el ámbito del proceso Dosificación de Áridos.

Para ello se simularon:

- Las diferentes entradas discretas correspondientes a interruptores, selectores, pulsadores y sensores con interruptores conectados a una fuente de 24 v.
- La entrada analógica (canal1) correspondiente a la señal de Peso de la tolva de cemento a través de un potenciómetro de 49 K Ω como especifica el manual y la fuente de referencia propia del módulo analógico integrado.

Asímismo se comprobó la comunicación entre PLC y el SCADA hecho en EROS mediante el cable XC1301 y el servidor OPC KEPLware v4.5, lo que evidenció una programación correcta del supervisor al poder visualizar los diferentes estados sobre el mímico del proceso y poder configurar parámetros a partir de éste, así como la visualización en gráficas de las tendencias asociadas al consumo energético de los motores.

El sistema desarrollado no descuida las prestaciones del sistema anterior, conservando las bondades de este y recuperando otras ya perdidas, incorporando un gran número de nuevas posibilidades, haciéndolo un sistema seguro, fiable y actualizado tecnológicamente.

En caso de implementación en un futuro se mantendrán partes y componentes tecnológicos que se encuentran en buen estado, como los sistemas mecánicos (cintas, mezcladora, *Skip*, tornillos sinfín), el sistema eléctrico de potencia (breakers, contactores magnéticos, entre otros), cambiando principalmente el controlador, y algunos medios técnicos de medición, constituyendo así una solución rápida y económica.

Conclusiones Parciales.

1. La aplicación de las redes PN como método formal permitió obtener un modelo a partir del cual se genera el programa para el control del sistema.
2. Mediante el análisis de propiedades se logró comprobar en el proceso de diseño la validez de los modelos formales obtenidos en cuanto a vivacidad y limitación.
3. Con la simulación de los modelos obtenidos se corroboraron muchas de las propiedades verificadas, permitiendo además validar los requerimientos funcionales del sistema en todos sus modos de operación.
4. Se comprobó a escala de laboratorio el funcionamiento real del programa en el autómata.
5. Se diseñó además el sistema supervisorio y se comprobó su funcionamiento a escala de laboratorio.

Conclusiones Generales

La automatización industrial integrada tiene un alto nivel de complejidad al integrar sus tres niveles básicos y por tanto la metodología y herramientas de modelado formal mediante Redes de Petri jerárquicas extendidas GHENeSys constituyen un arma efectiva para su diseño y programación.

En este trabajo se logran demostrar las ventajas de esta metodología en el desarrollo del sistema de automatización del subproceso de Dosificación en la planta de producción de bloques del combinado Los Guaos al lograr:

- La definición de requerimientos funcionales de la automatización del proceso.
- La modularidad requerida para el cumplimiento de las 7 guías de modelado definidas en Mendling [40].
- La aplicación del método de reducción para el cumplimiento de las propiedades básicas.
- La aplicación del método matemático basado en las invariantes de S y T de la matriz de incidencia para reafirmar la verificación de propiedades estructurales y la validación de algunos requerimientos funcionales del sistema.
- El uso de la simulación dinámica del comportamiento del modelo para la validación de todos los requerimientos funcionales de la automática local.
- La programación del sistema a partir de la traducción por analogías entre los modelos en PN y programas en lenguaje LD de PLCs.
- La creación del sistema de supervisión mediante el EROS para completar la integración del sistema de automatización al nivel estándar de actualidad internacional.
- La implementación a escala de laboratorio de todo el sistema propuesto.

Todo lo anterior permite una futura implementación industrial con amplias ventajas para dicho proceso y la empresa.

Recomendaciones

1. Comprobar el uso de las PNs cómo método formal en otros tipos de sistemas incluyendo el modelado del sistema supervisorio.
2. Implementar y comprobar el desempeño del sistema en el complejo de producción de bloques los Guaos de Santiago de Cuba.

Bibliografía

- 1 Ajmone, M.: "An introduction to generalized Stochastic Petri Nets". Dipartimento di Elettronica. Politecnico di Torino, Italy. Second International Course on Petri Nets for Latin America. Campina Grande, PB, Brazil. November 1995.
- 2 Baier, C.; Kaoten, J.: "Principles of Model Checking". The MIT Press Cambridge, Massachusetts, USA, 2008.
- 3 Bailey, D.; Wright, E.: "Practical SCADA for Industry". Editorial Elsevier. Great Britain. 2003
- 4 Benitez, I.: "Integración internacional de plataformas de enseñanza a distancia de automatización con PLCs". IEE-RITA. Revista Iberoamericana de Tecnologías de Aprendizaje. Vol 1. No 1. Nov. ISSN 1932-8540, 2006.
- 5 Benitez, I.: "Diseño formal de la automatización del molino de Clinker". Revista: Ciencia en su PC. MEGACEN, Santiago de Cuba. ISSN 1027-2887. Indexada en Latindex, UNAM, folio 4322 y EBSCO. 2010.
- 6 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 feb 2007.
- 7 Benitez, I.: "Modelado de sistemas de control con redes GHENeSys, una contribución al modelado de sistemas de automatización híbrida". Automatica08. Proceedings of FIE08. ISBN 978-84-00-08680-0 .Santiago de Cuba. Cuba, Julio, 2008.
- 8 Benitez, I.; Avila, E.; González, R.: "Modelado sobre Redes de Petri Estocásticas Generalizadas para el análisis de desempeño de redes de comunicaciones". VI Simposio de Telecomunicaciones, Convención Internacional Informática 2013. Ciudad Habana. 19 al 21 de Marzo 2013.
- 9 Benitez, I.; Gomis, O.; Sudriá, A.; otros. "Flexible manufacturing cell SCADA system for educational purposes". Journal Computer Applications in Engineering Education, Vol.16, No1, ISSN 1061-3773, Mayo 2008.
- 10 Benitez, I.; Mendes, B.; Sicchar, J.R.; Valente, D.: "A design method for flexible

- manufacturing system based on Petri Nets and UML". Conferencia Portuguesa de Control y Automática. Coimbra, Portugal. CONTROLO2010, 8 al 10 de sep. 2010.
- 11 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.
 - 12 Bilinski, K.; Adamski, M.; Saul, J.M.; Dagless, E.L.: "Petri Net Based Algorithms for Parallel Controller Synthesis". IEE Proceeding Comp. Technical. Vol. 141. Nov. 6. November. 1994.
 - 13 Brenan, R.: "Toward Real-Time Distributed Intelligent Control: A Survey of Research Themes and Applications". IEEE Transactions on Systems, Man, And Cybernetics Part C: Applications And Reviews, VOL. 37, NO. 5, September 2007.
 - 14 Bryan, L.A.; Bryan, E.A.: "Programmable Controllers. Theory and Implementation". Industrial Text Company. Second Edition. United States of America. 1997.
 - 15 Carneiro, E.; MACiel, P.; Callou, G.; Tavares, E.; Nogueira, B.: "Mapping SysML State Machine Diagram to Time Petri Net for Analysis and Verification of Embedded Real-Time Systems with Energy Constraints." Advances in Electronics and Micro-electronics, 2008. ENICS '08. International Conference on Digital Object Identifier: 10.1109/ENICS, Publication 19, Page(s): 1-6, Year: 2008.
 - 16 Clarke, E.M.; Wing, J.M.: "Formal Methods: State of the Art and Future Directions". Carnegie Mellon University. 1996.
 - 17 David, R.; Alla, H.: "Petri Nets and Grafcet – Tools for modelling Discrete Event Systems. Prentice hall, New York, London. 1992.
 - 18 Desel, J.;Esparza, J.: "Free Choice Petri Nets". Cambridge University Press. Great Britain. 1995.
 - 19 Drath, R.: "A Mathematical Approach to Describing a Class of Hybrid Systems". IEEE Workshop on Parallel and distributed Real Time Systems. Genf. 1997.
 - 20 Drath, R.: "Hybrid Object Nets: An Object Oriented Concept for Modeling Complex Hybrid Systems". In: Hybrid Dynamical Systems. 3rd International Conference on Automation of Mixed Processes. ADPM'98. Reims. 1998.

- 21 Du, Y.; Jiang, Ch.; Zhou, M.: "A Petri Net-Based Model for Verification of Obligations and Accountability in Cooperative Systems". IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, VOL. 39, NO. 2, pp 299-308, MARCH 2009.
- 22 Ferrarini, L.; Dedè, A.; Salaün, P.; Duang, T.; Fogliazza, G.: "Domain Specific Views in Model-driven Embedded Systems Design in Industrial Automation." 7th IEEE Internacional Conference on Industrial Informatics. INDIN 2009
- 23 Frey, G.: "Analysis of Petri Net based Control Algorithms - Basic Properties". Proceedings of the American Control Conference ACC 2000. Chicago. June 28-30, 2000.
- 24 Frey, G.: "Automatic Implementation of Petri Net Based Control Algorithms on PLC". Proceedings of the American Control Conference. pp. 2819-2823. ACC 2000. Chicago. June 28-30. 2000.
- 25 Frey, G.; Litz, L.: "Formal Methods in PLC Programming". Proceedings of the IEEE. SMC 2000. Nashville. TN. October 08-11. 2000.
- 26 Frey, G.; Litz, L.: "Verification and Validation of Control Algorithms by Coupling of Interpreted Petri Nets". Proceedings of the IEEE. SMC 1998. San Diego. October 11-14. 1998.
- 27 Girault, C.; Valk, R.: "Petri Nets for Systems Engineering. A guide to Modelling, Verification, Validation, and applications". Springer-Verlag. Berlin. 2001
- 28 Glez, P.: "GHENESYS: Uma Rede Estendida Orientada a Objetos para o Projeto de Sistemas Discretos". M.Sc. Thesis. Departamento de Engenharia Mecânica-Mecatrónica. Universidad de Sao Paulo. 2001.
- 29 Glez, P.; Silva, R.: "GHENESYS: Uma rede estendida para a modelagem, análise e projeto de sistemas complexos". Proc. of SBAI2001. Sao Paulo, Brasil. Nov. 2001.
- 30 Grupo de Desarrollo. "Sistema de Supervisión y Control de Procesos EROS". Grupo Eros Serconi, Nicaro ISMM, Moa.
- 31 Holloway, L.E.; Krogh, B.H. Giua, A. "A Survey of Petri Net Methods for Controlled Discrete Event Systems". Journal of Discrete Event Dynamic Systems, Vol 7 No. 2. 1997.
- 32 Hopcroft, J.; Motwani, R.; Ullman, J.: "Introduction to Automata Theory, Languages and Computation". Editorial: Addison-Wesley, 3ra Ed., ISBN- 13:9780321455369, 2007.

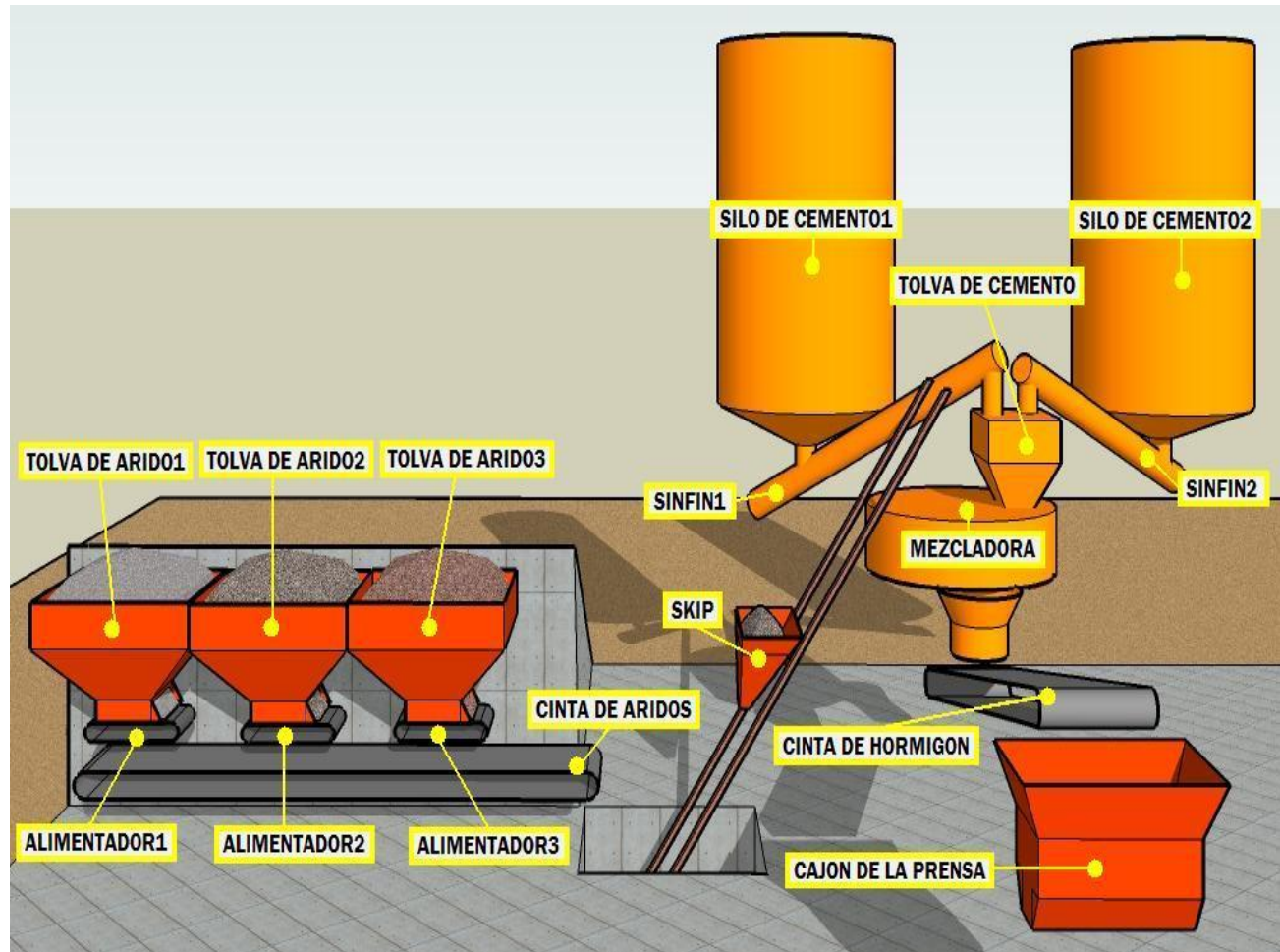
- 33 Jack, H.: "Automating Manufacturing Systems with PLCs". GNU Free Public Documentation License. Versión 4.2. April 3. 2003.
- 34 Jhonson, D.: "Research on Programmable Logic Controllers". Control Engineering, 12/1/2007.
- 35 Júlvez J.; Boel, R.: "A Continuous Petri Net Approach for Model Predictive Control of Traffic Systems". IEEE.
- 36 Kloetzer, M.; Mahulea, C.; Belta, C.; Silva, M.: "An Automated Framework for Formal Verification of Timed Continuous Petri Nets" IEEE Transactions on Industrial Informatics. VOL. 6. NO. 3. AUGUST 2010.
- 37 Leitao, P.: "Agent-based distributed manufacturing control: A state of the art survey". Polytechnic Institute of Braganca, Quinta Sta Apolonia, Apartado 1134, 5301-857 Braganca, Portugal. Engineering Applications of Artificial Intelligence 22, 979–991. 2009.
- 38 Lipson, P.; Zalm, G.: "Inside Machines: PC versus PLC: Comparing control options". Control Engineering Journal, 11/2011.
- 39 Mader, A.: "A Classification of PLC Models and Applications". WODES 2000. 5th Workshop on Discrete Event System. August 21-23. Gent. Belgium. 2000.
- 40 Medling, J.: "Empirical Studies in Process Model Verification". K. Jensen and W. van der Aalst (Eds.): ToPNoC II, LNCS 5460, pp. 208–224, 2009. Springer-Verlag Berlin Heidelberg 2009
- 41 Murata, T.: "Petri Nets: Properties, analysis and applications". Proceedings of IEEE, vol. 77, No. 4. April, 1989.
- 42 Olivera, A.: "Herramienta para la enseñanza de modelado de sistemas de automatización con redes de Petri GHENeSys". MSc. Thesis. Univ. de Oriente. Santiago de Cuba, Cuba. 2009.
- 43 Piedrafita, M.: "Ingeniería de la Automatización Industrial". Editorial RA-MA. Madrid. 1999.
- 44 Pietrusewicz, K; Urbański, Ł.: "Cover Story: Balancing PLCs, PACs, IPCs". W. Pomeranian Univ of Tech. Control Engineering Journal, 01/2011.
- 45 Reisig, W.: "Petri Nets, An introduction". Springer-Verlag. 1982.
- 46 Ribeiro, O.R.; Fernandes, J.M.: "Validation of Scenario-based Business Requirements

- with Coloured Petri Nets". 2009 Fourth International Conference on Software Engineering Advances. ICSEA '09. Page(s): 250 – 255. USA. 2009.
- 47 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. Thesis. 2009-11. Área: Automática. Santiago de Cuba, Cuba. Junio 2011.
- 48 Seidner, C.; Roux, O.: "Formal Methods for Systems Engineering Behavior Models" IEEE Transactions on Industrial Informatics, VOL. 4, NO. 4, Page(s): 280-291, NOVEMBER 2008.
- 49 Silva, J.R.; Miyagi, P.F.; Matos, L.M.; Afsarmanesh, H.: "A High Level Net to the Modeling of Discrete Manufacturing Systems". (Eds). Balanced Automation Systems – Architectures and Design Methods. IFIP/Hapman & Hall. London. 1995.
- 50 Silveira, M.: "Supervision and Control of Heterarchical Discret Event Systems". Revista Controle & Automazación, Vol.17 No.1, Enero, Febrero y Marzo, 2006.
- 51 Simonak, S.; Hudak, S.; Korecko S.: "Protocol Specification and Verification Using Process Algebra and Petri Nets". International Conference on Computational Intelligence, Modelling and Simulation, pp110 – 114. 2009.
- 52 Strasser, T.; Rooker, M.; Hegny I.; Wenger, M.; Zotil, A.; Ferrarini, L.; Deter, A; Colla, M.: "A research roadmap for Model-driven Embedded Systems Design for Automation Components". 7th IEEE internacional Conference on Industrial Informatics. INDIN 2009.
- 53 Torrico, C.: "Control Supervisorio Hierárquico Modular por Agregación de Estados". Rvista Control & Automatización, Vol.15 No.3, Julio, Agosto y Septiembre, 2004.
- 54 Tremosa, L.: "Llega el e-manufacturing". Revista Automática e Instrumentación. Cetisa Editores SA. Barcelona. España. Febrero 2002.
- 55 Uzam, M.: "Petri Net Based Supervisory Control of Discrete Event Systems and their Ladder Logic Diagram Implementations". Ph. D. Thesis. University of Salford. UK. 1998.
- 56 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. Thesis. 2011-12. Área: Automática. Univ. de Oriente. Santiago de Cuba, Cuba. Mayo 2012.
- 57 Vázquez, R.; Silva, M.: "Stochastic Continuous Petri Nets: An Approximation of Markovian Net Models" IEEE Transactions on Systems, Man, and Cybernetics-Prt A:

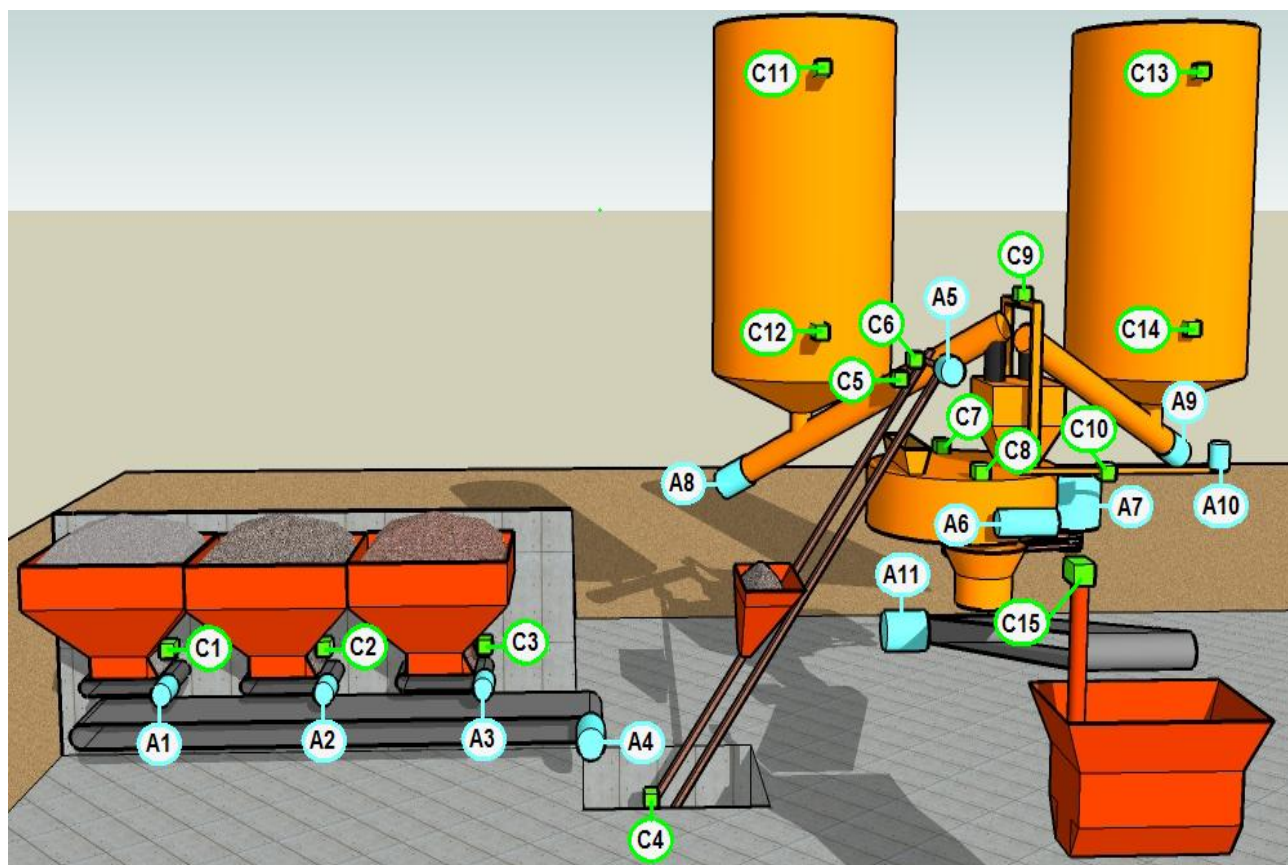
Systems and Humans, USA. 2011.

- 58 Wei, Sh.; Bai-gen, C.; Jian, W.; Yan, W.; Chen-xi, Gou.: “Research of System Modeling and Verification Method Combine with UML Formalization Analysis and Colored Petri Net”. Third International Symposium on Intelligent Information Technology Application, 2009. IITA 2009. Volume: 3 Digital Object Identifier: 10.1109/IITA.2009.489 Publication, Page(s): 488 – 491, Year: 2009.
- 59 Xu, J.; Recalde, L.; Silva, M.: “Tracking Control of Join-Free Timed Continuous Petri Net” Systems under Infinite Servers Semantics Discrete Event. Dyn Syst 18: 263–283. January 2008.

Anexo 1. Componentes principales del subproceso “Dosificación”.



Anexo 2. Distribución de sensores y actuadores en el subproceso “Dosificación”, también conocido como Planta de Hormigón.



Sensor	Tipo	Función
C1	Fin de carrera.	Detecta salida de material de la tolva de arido1.
C2	Fin de carrera.	Detecta salida de material de la tolva de arido2.
C3	Fin de carrera.	Detecta salida de material de la tolva de arido3.
C4	Fin de carrera.	Detecta que el <i>Skip</i> está abajo.
C5	Fin de carrera.	Detecta que el <i>Skip</i> está arriba.
C6	Fin de carrera.	Detecta que el <i>Skip</i> está arriba (por seguridad).
C7	Fin de carrera.	Detecta si se abre la tapa de la mezcladora.
C8	Fin de carrera.	Detecta si se abre la tapa de la mezcladora.
C9	Galgas de esfuerzo.	Censa el peso del cemento en la tolva.
C10	Sensor de caudal de	Totaliza la cantidad de litros de agua vertida en la

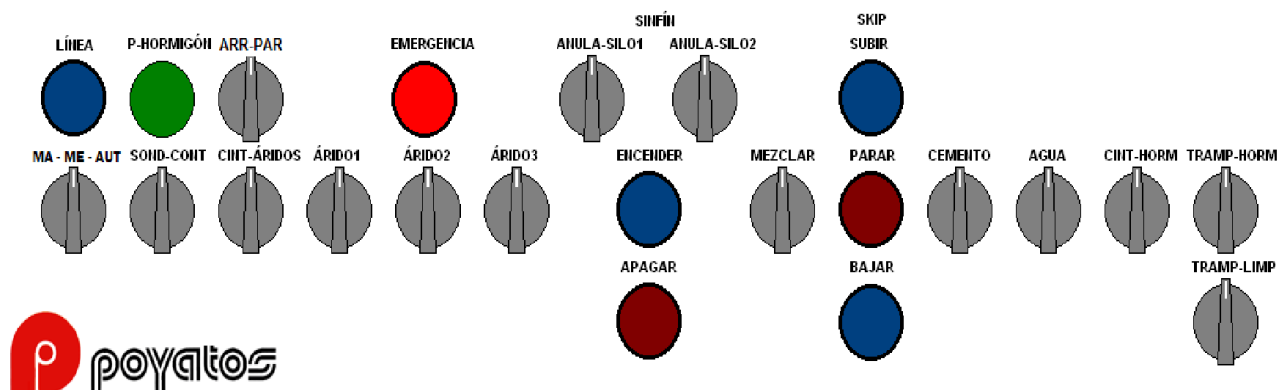
	desplazamiento positivo.	mezcladora en cada dosificación.
C11	De paleta rotativa.	Detecta nivel máximo en el silo de cemento 1.
C12	De paleta rotativa.	Detecta nivel mínimo en el silo de cemento 1.
C13	De paleta rotativa.	Detecta nivel máximo en el silo de cemento 2.
C14	De paleta rotativa.	Detecta nivel mínimo en el silo de cemento 2.
C15	Sonda conductiva.	Detecta nivel mínimo en el cajón de hormigón de la prensa.
A1	Motor 3	Mueve la cinta encargada de extraer el polvo de arena de la tolva árido1 hacia la cinta de áridos.
A2	Motor	Mueve la cinta encargada de extraer la arena de la tolva árido2 hacia la cinta de áridos.
A3	Motor	Mueve la cinta encargada de extraer la grava de la tolva árido3 hacia la cinta de áridos.
A4	Motor	Mueve la cinta de áridos encargada de trasladar los materiales extraídos al <i>Skip</i> .
A5	Motor	Es el encargado de trasladar el <i>Skip</i> desde la zona de carga hacia la mezcladora y viceversa.
A6	Pistón neumático de doble efecto.	Abre o cierra la compuerta para descargar el hormigón de la mezcladora hacia la cinta de hormigón.
A7	Motor	Mueve los encargados de mezclar los áridos, el cemento, y el agua para formar el hormigón.
A8	Motor	Mueve el tornillo sinfín encargado de extraer el cemento del silo 1 hacia la tolva de cemento.
A9	Motor	Mueve el tornillo sinfín encargado de extraer el cemento del silo 2 hacia la tolva de cemento.
A10	Electroválvula	Permite abrir o cerrar el paso de agua hacia la mezcladora.
A11	Motor	Mueve la cinta de hormigón encargada de trasladar el material al cajón de la prensa.

Por razones de espacio en la imagen no se visualiza pero existen también:

- Un pistón neumático de doble efecto en un lateral de la tolva de cemento cuya función es abrir o cerrar la compuerta que permite el vaciado de la misma.
- Un motor vibrador acoplado al cuerpo de la tolva de cemento para que el mismo no se adhiera a las paredes.
- Un motor vibrador acoplado al cuerpo de la tolva de árido 1 (polvo de arena) para que igualmente este no se adhiera a las paredes y descienda de una forma uniforme.
- Una bomba de agua que garantiza el flujo de del líquido hacia la mezcladora.
- Un electro freno acoplado al motor del *Skip* cuya función es garantizar seguridad ante un corte de energía, deteniendo automáticamente el *Skip* en posición en que se encuentre.

Vale destacar que tanto el pistón neumático (A6) cómo el que se mencionó anteriormente cuentan con sensores magnéticos para detectar efectivamente en que extremos se encuentra el vástago, lo que se relaciona directamente con la apertura o cierre de la compuerta correspondiente.

Anexo 3. Panel de control del subproceso “Dosificación”.



En este Anexo se pueden observar los controles existentes en el pupitre y su disposición para el mando de los diferentes recursos existentes en el subproceso de dosificación, controles que cumplen las diferentes funciones descritas a continuación:

LÍNEA: Es un botón pulsador que energiza los elementos de fuerza (pre-actuadores utilizados entre el autómatas y los motores: contactores) existentes y los controles del panel.

P-HORMIGÓN: Es un indicador de que los elementos de fuerza y los controles ya están energizados.

EMERGENCIA: Es un botón pulsador para la parada de emergencia de este subproceso, al ser presionado desconecta los elementos de fuerza y los controles del panel.

MAN-AUT: Es un botón selector de tres posiciones que define el modo de funcionamiento manual, manual extendido o automático del subproceso.

SOND-CONT: Es un botón selector de dos posiciones (interruptor) que define si el ciclo productivo se repite continuamente o en dependencia de si la sonda en el cajón de la prensa no detecta hormigón.

CINT-ÁRIDOS: Es un interruptor que permite encender o apagar la cinta transportadora de áridos.

ÁRIDO1: Es un interruptor que permite encender o apagar simultáneamente la cinta transportadora de la tolva de polvo de arena (alimentador 1) y su correspondiente motor vibrador.

ÁRIDO2: Es un interruptor que permite encender o apagar la cinta transportadora de la tolva de arena (alimentador 2).

ÁRIDO3: Es un interruptor que permite encender o apagar la cinta transportadora de la tolva de granito (alimentador 3).

ANULA-SILO1: Es un interruptor que permite habilitar o deshabilitar el uso del silo de cemento 1.

ANULA-SILO2: Es un interruptor que permite habilitar o deshabilitar el uso del silo de cemento 2.

ENCENDER: Es un botón pulsador que permite encender el tornillo sinfín correspondiente al silo de cemento habilitado.

APAGAR: Es un botón pulsador que permite apagar el tornillo sinfín correspondiente al silo de cemento habilitado.

MEZCLAR: Es un interruptor que permite encender o apagar la mezcladora.

SUBIR: Es un botón pulsador que permite desplazar el carro colector desde la zona de carga de áridos hacia la mezcladora o zona de descarga.

PARAR: Es un botón pulsador que permite parar el desplazamiento del carro colector sea cual fuere el sentido de su desplazamiento.

BAJAR: Es un botón pulsador que permite desplazar el carro colector desde la mezcladora o zona de descarga hacia la zona de carga de áridos.

CEMENTO: Es un interruptor que permite abrir o cerrar una compuerta en la tolva de cemento (báscula) para la descarga del cemento en la mezcladora así como encender o apagar simultáneamente el vibrador instalado en la misma.

AGUA: Es un botón de dos posiciones que permite encender o apagar la bomba de agua que alimenta la mezcladora y al mismo tiempo abrir o cerrar la electroválvula que la deja pasar.

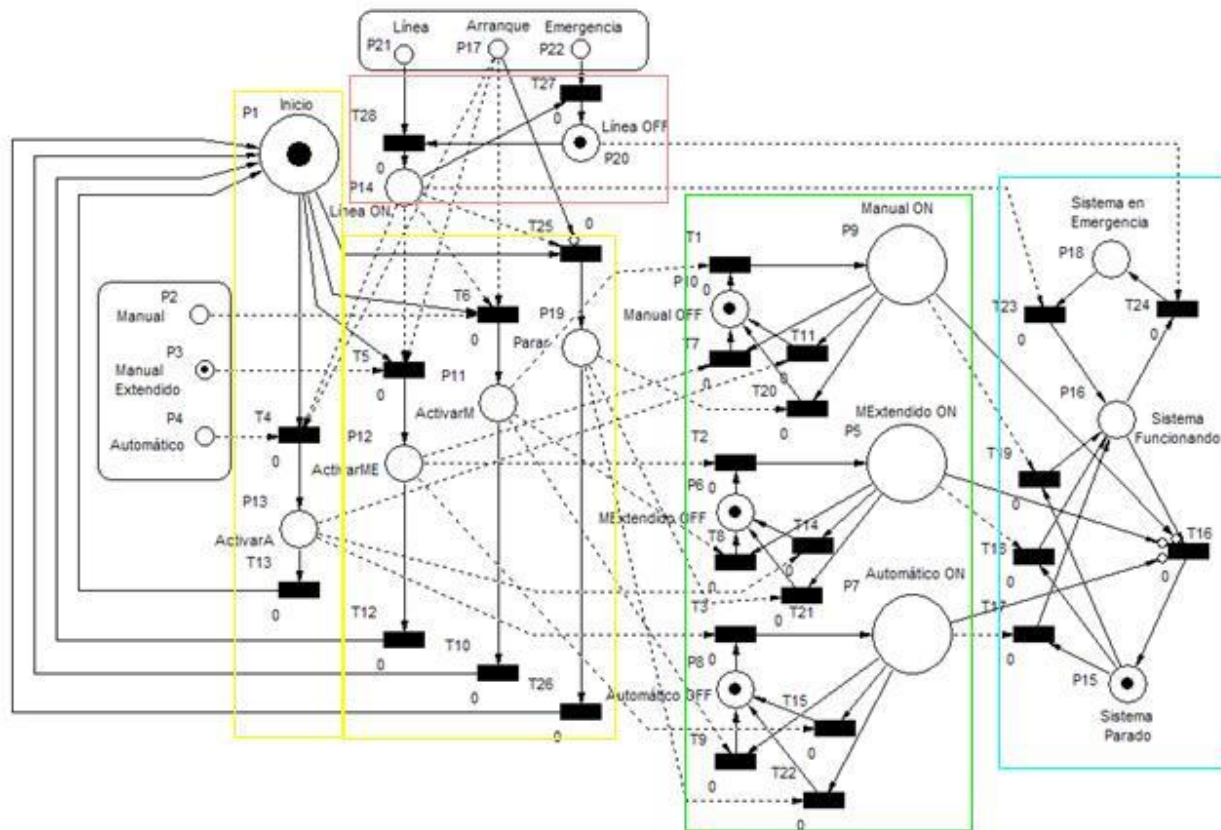
CINT-HORMIGÓN: Es un interruptor que permite encender o apagar la cinta transportadora de hormigón.

TRAMP-HORM: Es un interruptor que permite abrir o cerrar una compuerta en la mezcladora para la evacuación del hormigón hacia la cinta transportadora de hormigón.

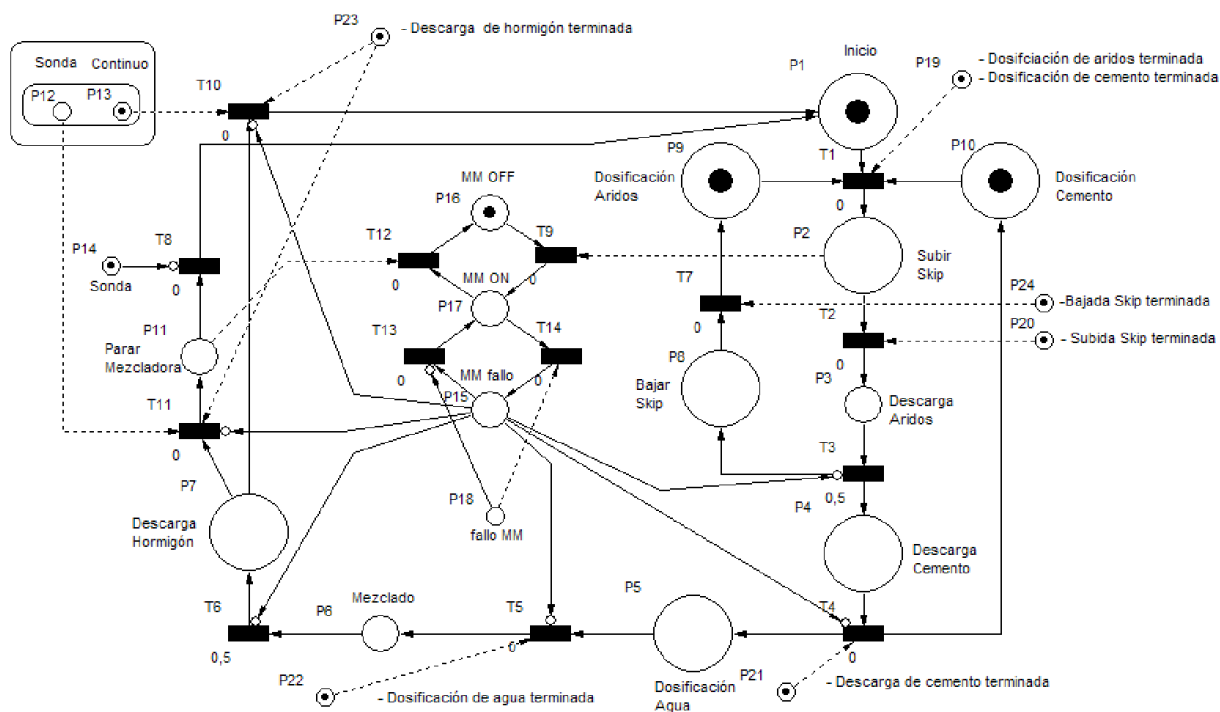
TRAMP-LIMP: Es un interruptor que permite abrir o cerrar una compuerta para la evacuación del material residual durante la limpieza de la mezcladora.

Anexo 4. Modelos principales del subproceso de dosificación

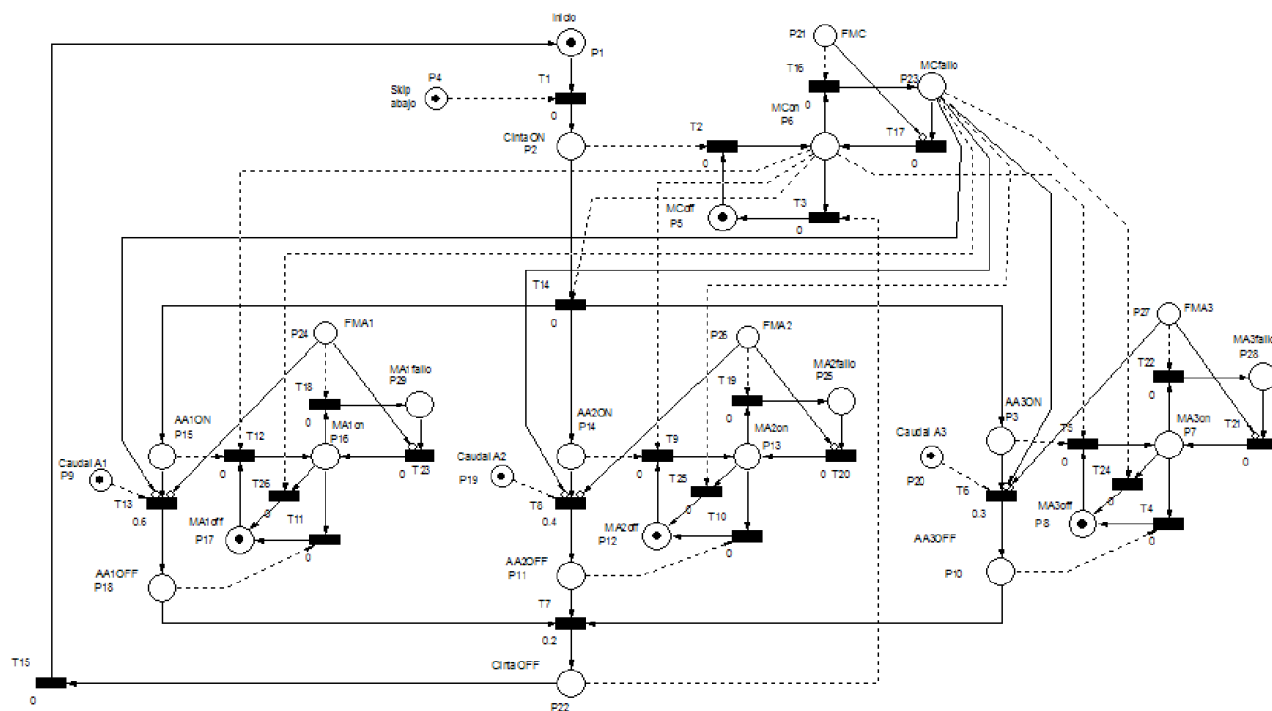
Anexo 4.1 Modelo en PN de la unidad funcional “Programa Principal”.



Anexo 4.2 Modelo en PN de la unidad funcional “Automático”.

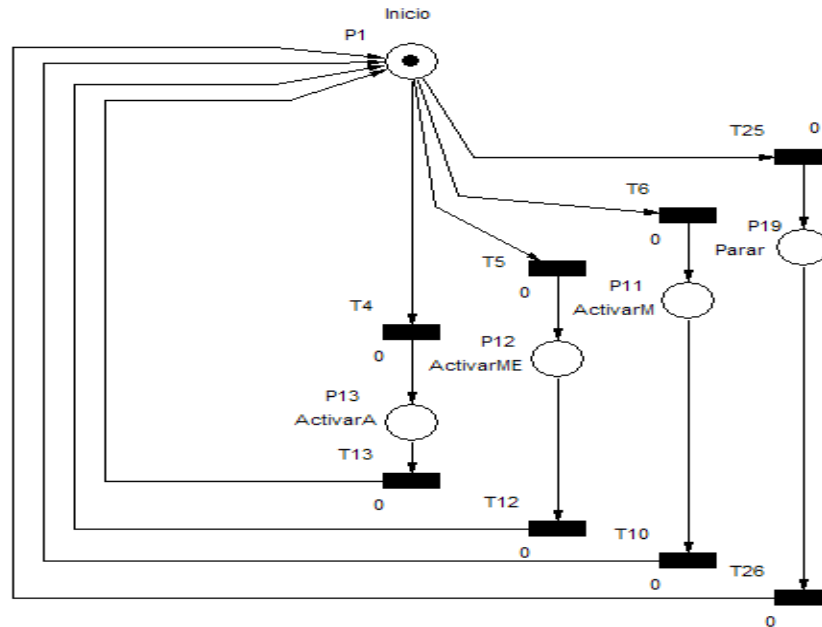


Anexo 4.3 Modelo en PN de la unidad funcional “Dosificación Áridos”.

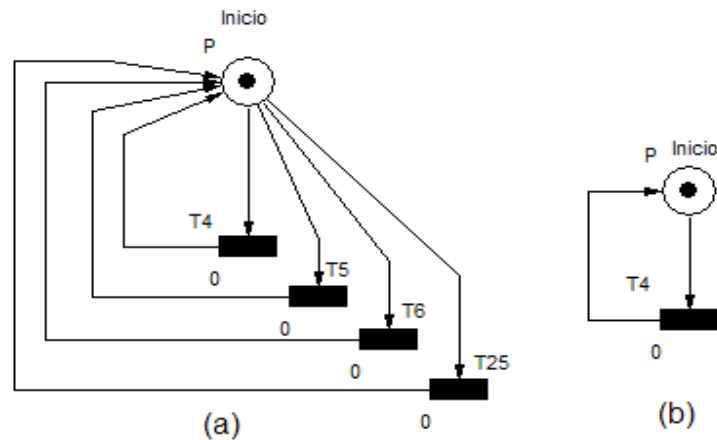


Anexo 5. Reducción de modelo del Programa principal

Anexo 5.1 PN principal correspondiente al modelo no controlado del Programa Principal del Anexo 4.1.

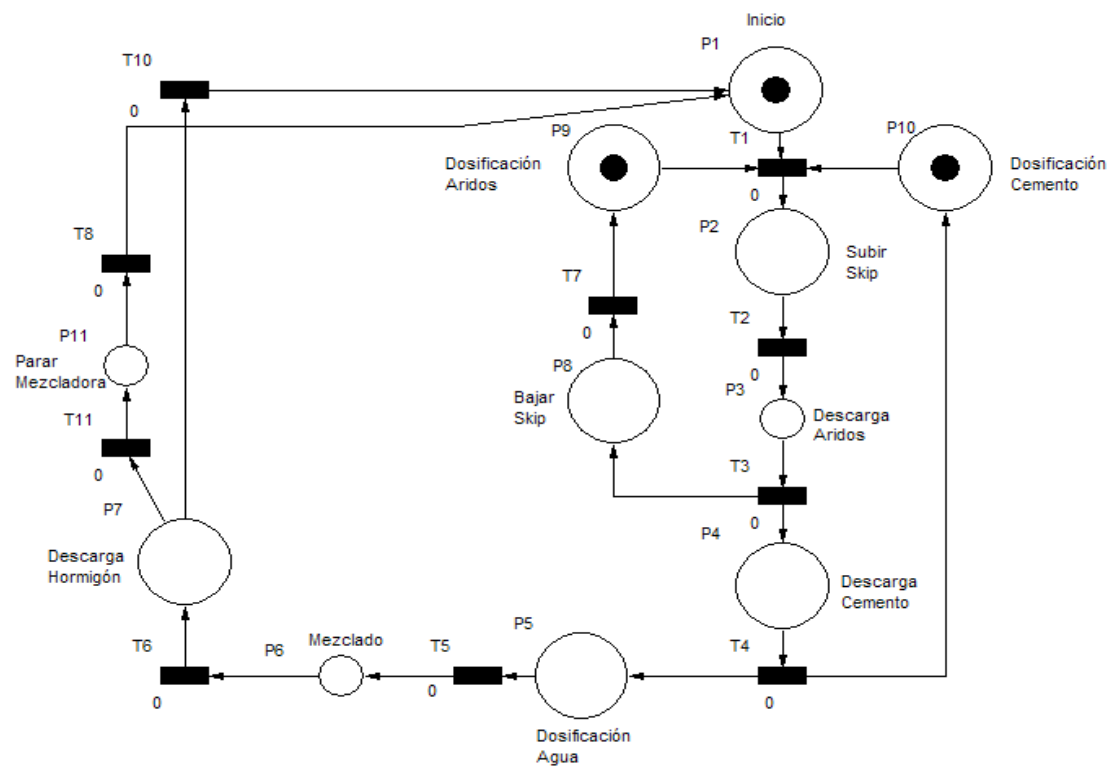


Anexo 5.2 Reducción de la PN principal correspondiente al modelo no controlado del Programa Principal. **(a)** Resultado de la primera reducción de la PN mostrada en el Anexo 5.1. **(b)** Resultado final de la reducción de la PN mostrada en (a).

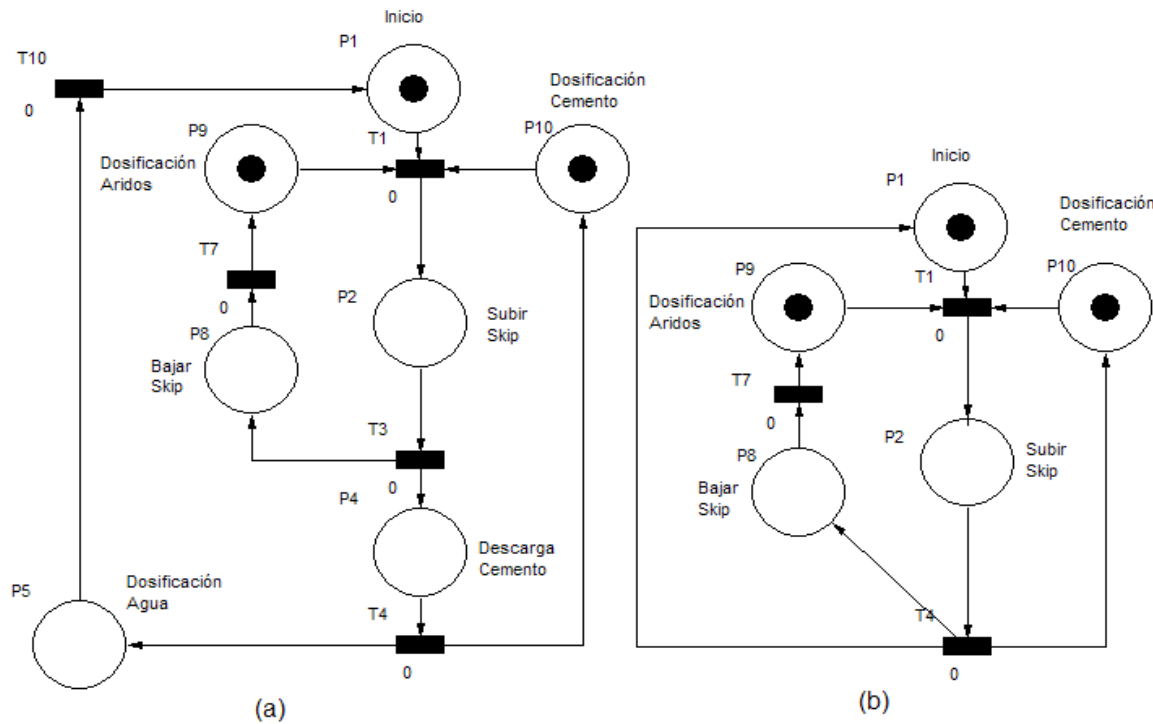


Anexo 6. Reducción de modelo del Programa Automático

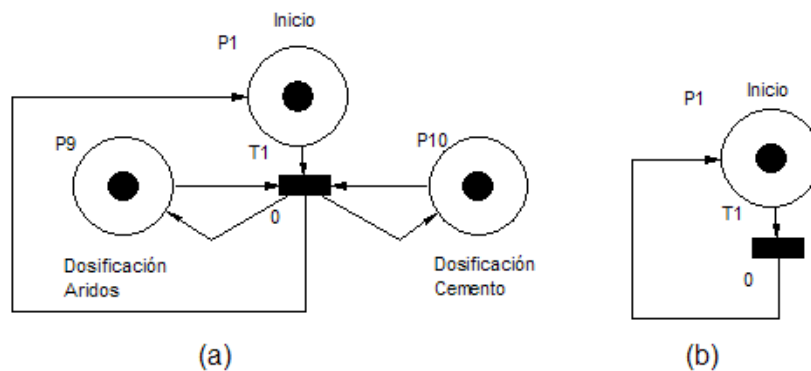
Anexo 6.1 PN principal correspondiente al modelo no controlado del macro lugar Automático del Anexo4.2.



Anexo 6.2 Reducción de la PN principal correspondiente al modelo no controlado del macro lugar Automático. **(a)** Resultado de la primera reducción de la PN mostrada en el Anexo 6.1. **(b)** Resultado de la segunda reducción de la PN mostrada en (a).

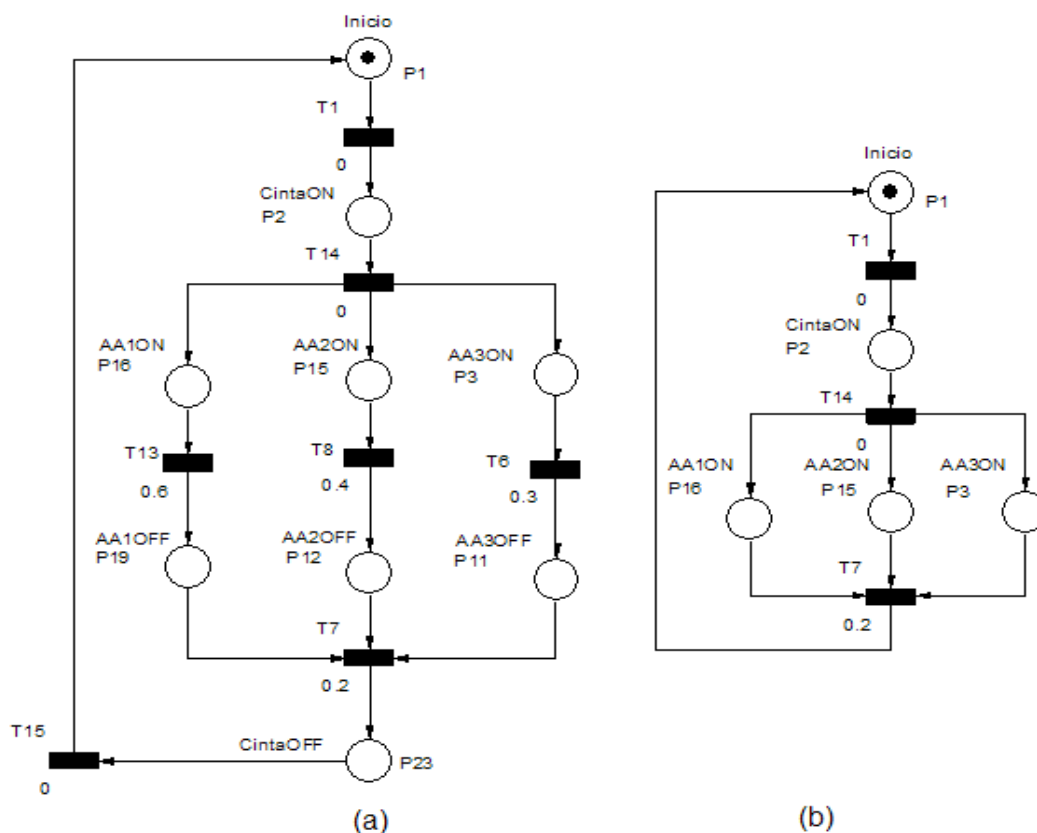


Anexo 6.3 Reducción de la PN principal correspondiente al modelo no controlado del macro lugar Automático. **(a)** Resultado de la tercera reducción de la PN mostrada en el Anexo 6.2 **(b).** **(b)** Resultado final de la reducción.

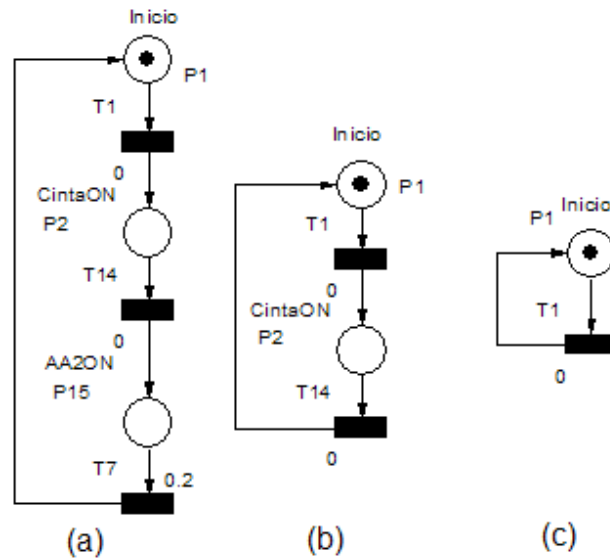


Anexo 7. Reducción de modelo del Programa Dosificación de Áridos

Anexo 7.1 Reducción de la PN principal correspondiente al modelo no controlado del macro lugar Dosificación Áridos. **(a)** PN correspondiente al modelo no controlado del macro lugar Dosificación Áridos del Anexo 4.3. **(b)** Resultado de la primera reducción de la PN mostrada en (a).

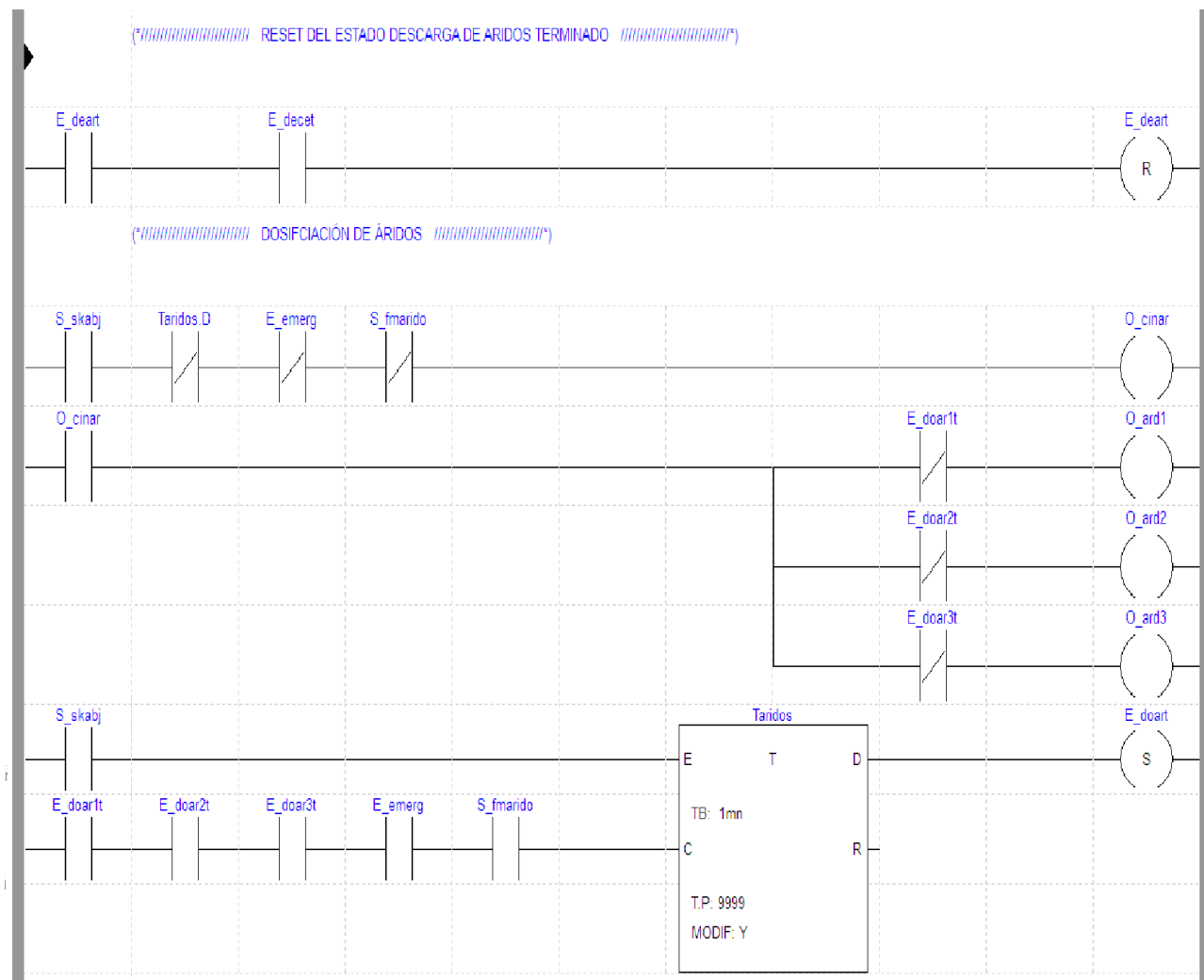


Anexo 7.2 Reducción de la PN principal correspondiente al modelo no controlado del macro lugar Dosificación Áridos. **(a)** Resultado de la segunda reducción de la PN mostrada en el Anexo 7.1 (b). **(b)** Resultado de la tercera reducción de la PN mostrada en (a). **(c)** Resultado final de la reducción.

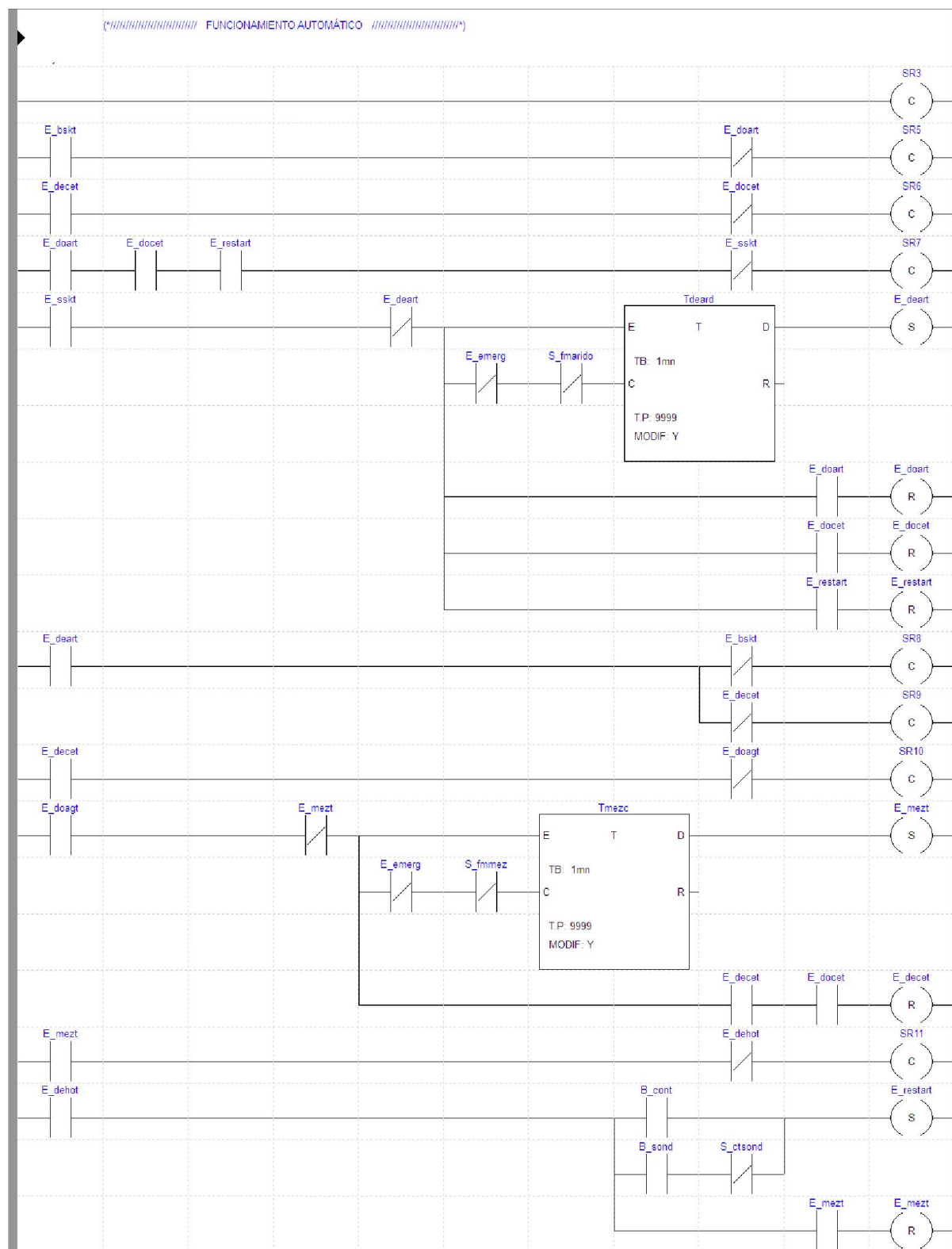


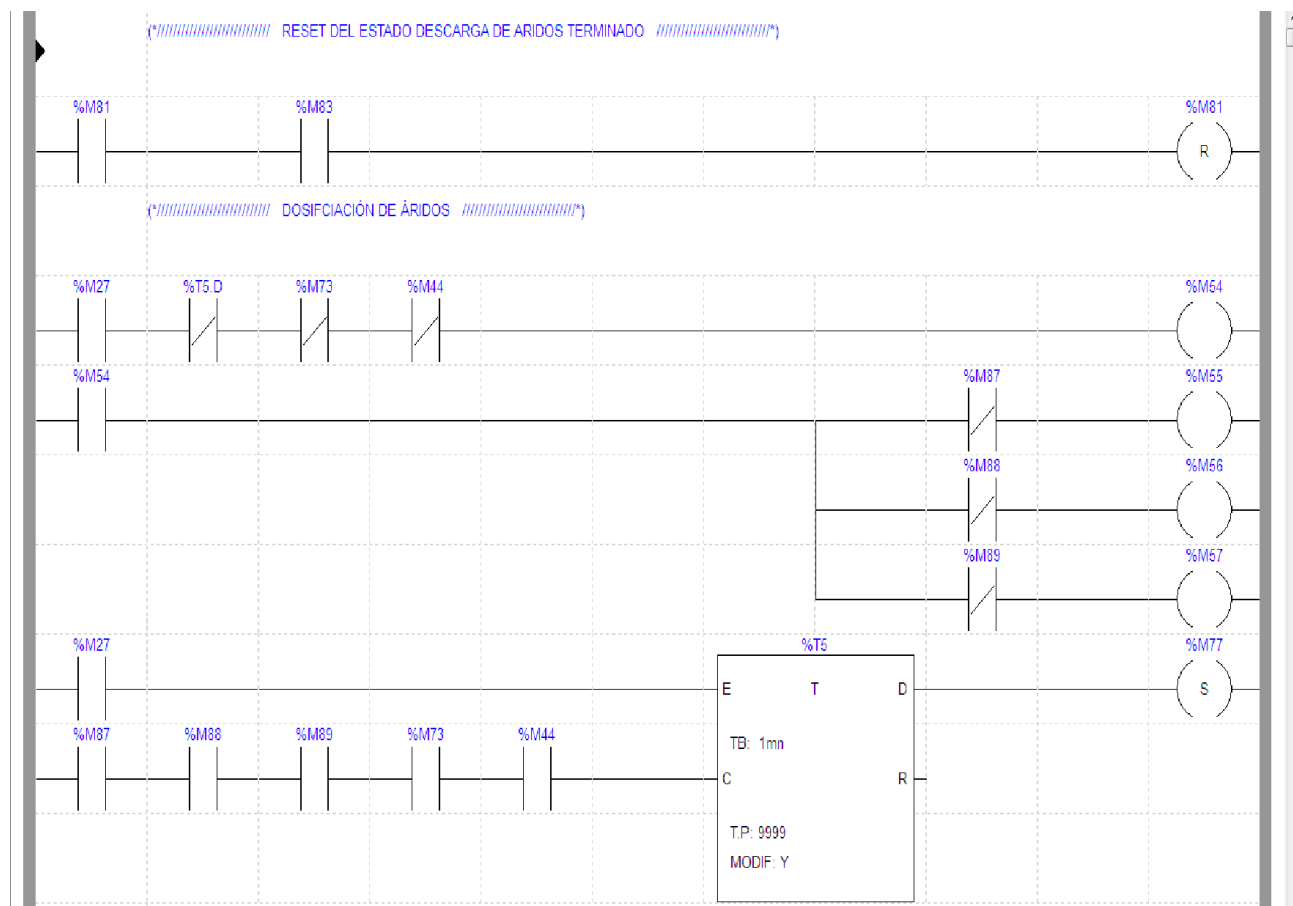
Anexo 8. Programación en LD del PL7Pro de Dosificación

Anexo 8.1 Programa Principal en LD del PL7Pro.



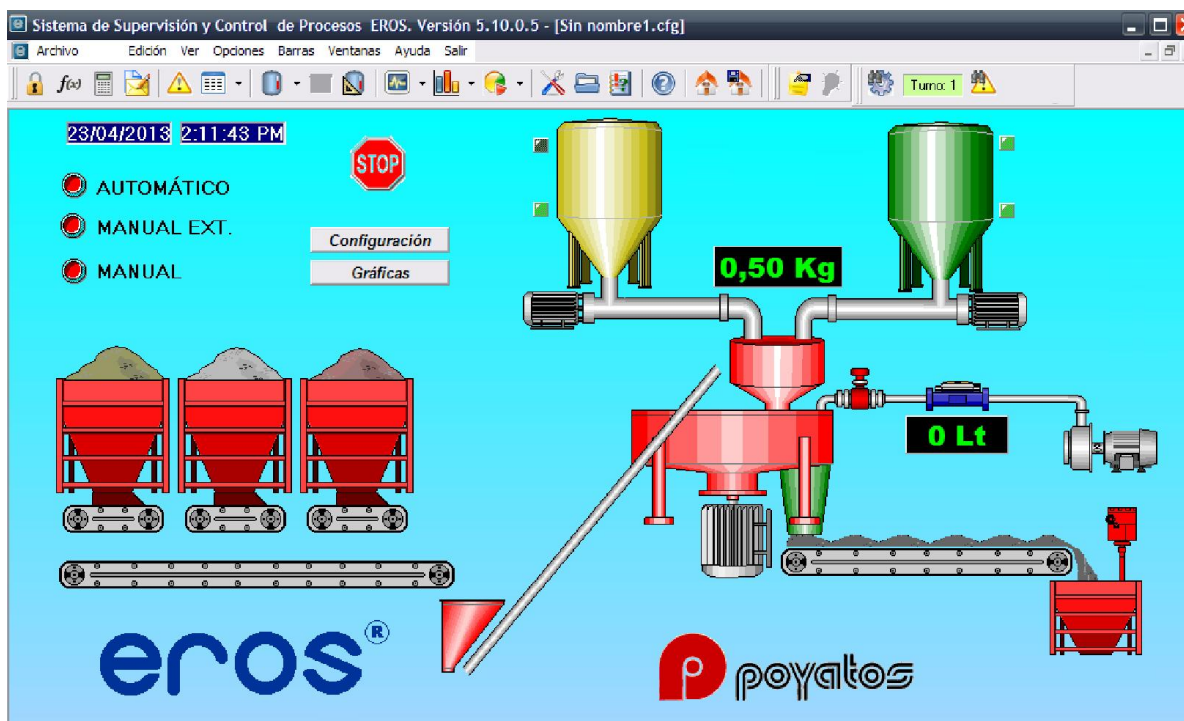
Anexo 8.2 Programa Automático en LD del PL7Pro.



Anexo 8.3 Programa Dosificación de áridos en LD del PL7Pro.

Anexo 9. Programación en Eros del Supervisorio para Dosificación.

Anexo 9.1 Ventana principal (mímico).



Anexo 9.2 Ventana de configuración.

