

Implementación de Supervisores en una Arquitectura de Referencia Basadas en Sistemas de Manufactura Holónicos

E. Chacón, M. Indriago

*LaSDAI, Departamento de Computación, Universidad de Los Andes. Av.
Tulio Febres Cordero, Mérida 5101, Venezuela*

Resumen: La Arquitectura de Integración para la Automatización de Complejos de Producción es una propuesta de arquitectura de referencia basada en sistemas de manufacturas holónicas que divide los procesos en unidades de producción. Cada unidad de producción representa una parte del proceso (y su sistema de control) con autonomía y capacidad para negociar con las otras unidades de producción. Este trabajo presenta un método para modelar e implementar supervisores de eventos discretos en la unidad de producción. El supervisor se modela con una red de Petri a la que se le agrega una interfaz para poder leer los eventos y ejecutar las acciones del proceso que se quiere controlar. El método permite simular, probar y modificar el supervisor fácilmente esto hace que la unidad de producción sea flexible a los cambios en el producto o en el proceso. *Copyright © 2010 CEA.*

Palabras Clave: Sistemas de manufactura holónicos, Supervisor, Sistemas de eventos discretos, Redes de Petri.

1. INTRODUCCIÓN

Desde siempre se ha buscado mejorar los procesos industriales de tal forma que los productos se elaboren en menor tiempo, a un menor costo y con mejor calidad. Una vía para lograr esto es tener procesos flexibles, y se logra con nuevas tecnologías, y estructuras organizacionales que permitan adaptarse a los cambios que exige el mercado. En este sentido existen distintos enfoques como los holónicos (Wyns, 1999), los heterárquicos (Silveira and Combacau, 2003; Saad *et al.*, 1995) y los jerárquicos (Li, 1994; ISA, 1995). Dentro de los enfoques holónicos se encuentra la Arquitectura de Integración para la Automatización de Complejos de Producción (Chacón *et al.*, 2002), en ella se basará nuestra propuesta.

Los sistemas de manufactura holónicos se originan de una teoría acerca de sistemas complejos adaptativos. La palabra holón fue introducido en los años sesenta por Arthur Koestler, y sugiere un comportamiento doble, el holón se puede comportar como un todo pero a su vez también se puede comportar como una parte (efecto Janus). Los sistemas de manufacturas holónicas están formados por comunidades de holones, cada holón puede formar parte de otro holón que lo contiene y a su vez estar formado por otros holones. Los sistemas de manufactura holónicos aprovechan las ventajas de los sistemas jerárquicos y de los sistemas heterárquicos, en este sentido un holón es un ente autónomo capaz de tomar sus propias decisiones y a su vez tiene la capacidad de negociar con otros holones con la finalidad de alcanzar las metas en común. La comunidad de holones relacionados entre sí es lo que se conoce como holarquía.

La Arquitectura de Integración para la Automatización de Complejos de Producción divide cada proceso en subprocesos denominados unidades de producción (UP). Cada unidad de producción está formada por una parte física que le da valor

agregado al producto y un sistema de control que tiene entre sus funciones: mantener las variables continuas y el estado discreto en los valores que permitan alcanzar las metas fijadas, manejar las contingencias, registrar información para el mantenimiento, medir variables que tengan impacto económico y negociar con otras unidades de producción. Las unidades de producción son autónomas y tienen capacidad para negociar.

Para que un proceso sea flexible los cambios en el sistema de control deben ser pocos complicados y rápidos, además la implementación debe ser sencilla y confiable. El diseño y puesta en funcionamiento del sistema de control involucra cuatro etapas: modelo de la planta, diseño del control, simulación e implementación. El modelo de la planta depende de su dinámica, la cual puede ser continua, de eventos discretos o híbrida. Para el desarrollo de los modelos se usan herramientas tales como ecuaciones diferenciales (Lennartson *et al.*, 1994), redes de Petri (Polic and Jezernik, 2005; Moody and Antsaklis, 1998; Murata, 1989; Peterson, 1981) o una combinación de ambos (Moody and Antsaklis, 1998; Frey, 1998). Resulta útil incluir en el sistema de control un supervisor de eventos discretos que ejecute acciones según el estado en que se encuentre la planta, algunos autores (Zapata and Carrasco, 2002; Frey, 1998) utilizan redes de Petri interpretadas para su elaboración. Existen diversas técnicas que ayudan al diseño de supervisores muchas de ellas están basadas en redes de Petri. La implementación del supervisor se debe hacer en un equipo de control, es común utilizar controladores lógicos programables (PLC) o computadores personales (PC) para este fin.

Para cumplir con las funciones del sistema de control de la unidad de producción, en este trabajo se presenta una propuesta que divide el sistema de control en tres capas: control regulatorio que se encarga de mantener las variables continuas en el valor deseado, supervisor que determina y controla el estado discreto y un sistema multiagente que maneja las contingencias

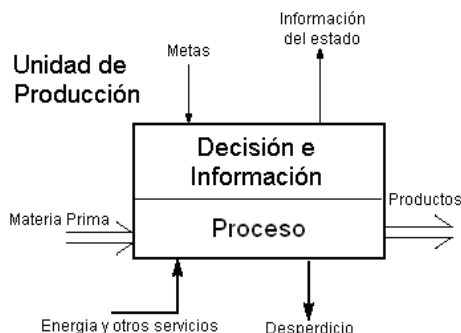


Figura 1. Entradas y salidas de una unidad de producción.

y se en carga de las negociaciones. El alcance de este trabajo se limita a detallar la capa del supervisor. Actualmente no se tiene un método para implementar un sistema supervisor en la Arquitectura de Integración para la Automatización de Complejos de Producción. El método que se presenta propone un supervisor de eventos discretos compuesto por un modelo de la planta construido usando redes de Petri, una función de control que ejecuta las acciones dependiendo del estado del modelo y una interfaz con el proceso que convierte las entradas en eventos y las acciones en salidas, además tiene la capacidad de intercambiar información con las otras dos capas. También se muestra una forma para implementar el supervisor en el equipo de control.

El artículo está formado por las siguientes secciones: Una introducción en la sección 1. La sección 2 define la Arquitectura de Integración para la Automatización de Complejos de Producción. La sección 3 habla sobre el modelo del supervisor. La sección 4 presenta dos alternativas para la implementación de los supervisores de eventos discretos. En la sección 5 se muestra un caso de estudio. Finalmente, en la sección 6 se dan las conclusiones.

2. ARQUITECTURA DE INTEGRACIÓN PARA LA AUTOMATIZACIÓN DE COMPLEJOS DE PRODUCCIÓN

La Arquitectura de Integración para la Automatización de Complejos de Producción es una propuesta desarrollada en la Universidad de Los Andes, Venezuela, está basada en sistemas de manufactura holónicos y tiene como objetivo fundamental facilitar la implementación del sistema de control en complejos de producción (CP), esto lo logra dividiendo los complejos de producción en partes llamadas unidades de producción.

En la unidad de producción se ejecutan los procesos de producción y las funciones de regulación y control. La unidad de producción “conoce” su estado, sus capacidades, sus restricciones y como alcanzar sus metas de producción; tiene un alto grado de autonomía y es capaz de negociar con otras unidades de producción.

Dos elementos principales forman a la unidad de producción, estos son el proceso y el sistema de control también llamado sistema de información y toma de decisiones (SI&TD). El proceso está formado por los elementos físicos y/o procesos químicos que transforman los materiales de entrada en otros productos. El sistema de información y toma de decisiones se encarga del control y regulación del proceso e intercambia información con otros sistemas para tener el control y regulación global del complejo. La autonomía de la unidad de producción implica que está en capacidad de tomar decisiones con respecto a su

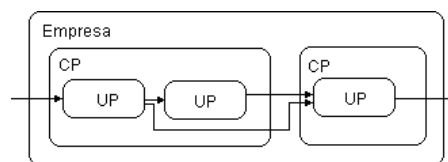


Figura 2. Estructura holónica

misión tales como: cambiar los procedimientos para elaborar el producto, detectar si una meta ha sido modificada e informar si esta meta puede ser aceptada, además debe informar si una meta ha sido completada o si no puede ser alcanzada debido a alguna situación anormal. La materia prima, las metas, la energía y otros servicios son entradas de la unidad de producción, mientras que los productos, información del estado y desperdicio son salidas. La figura 1 muestra un esquema de los elementos, las entradas y las salidas de una unidad de producción.

Las unidades de producción se integran para formar un complejo de producción, y varios complejos de producción conforman una empresa. En la figura 2 se muestra el enfoque dado por la Arquitectura de Integración para la Automatización de Complejos de Producción, el cual difiere del enfoque clásico como el que se muestra en la figura 3.

El complejo de producción está orientado a coordinar las unidades de producción con el propósito de optimizar los recursos y el plan de operaciones, además establece la cantidad de producción, envía las órdenes, ejecuta funciones de contabilidad e inventario, realiza actividades de mantenimiento y establece requerimientos de materiales.

En la empresa son ejecutadas las funciones de planificación estratégica y el manejo centralizado en el que se decide acerca de las finanzas, recursos humanos, mercado, investigación y actividades de desarrollo. Además se coordinan las actividades de producción ejecutadas en los complejos de producción.

La unidad de producción tiene como características la autonomía, capacidad de negociación, agregación (puede formar parte de otras unidades de producción) y especialización (puede estar formado por otras unidades de producción), las cuales son análogas a las características del holón. En este mismo sentido observamos que tanto la empresa como los complejos de producción también pueden ser considerados unidades de producción, la figura 4 muestra un diagrama donde se ilustra esta relación.

2.1 Sistema de información y toma de decisión

El sistema de información y toma de decisiones de la unidad de producción interactúa con el proceso determinado el estado del mismo a través de sensores, esta medición del estado es

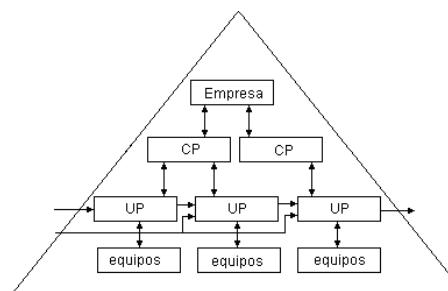


Figura 3. Estructura jerárquica.

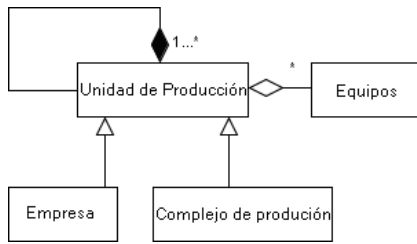


Figura 4. Modelo de una arquitectura de empresas.

comparada con el estado que debe tener el proceso para permitir alcanzar las metas. El resultado de la comparación se evalúa para decidir las acciones que son llevadas al proceso a través de actuadores, esto es lo que se conoce como ciclo de toma de decisiones. La figura 5 muestra el ciclo de toma de decisiones de la unidad de producción en forma gráfica.

El sistema de información y toma de decisiones se divide en tres capas, como se muestra la figura 6. A cada capa se le asignan determinadas funciones, esto hace más sencillo el diseño. La primera capa es el control regulatorio encargado del manejo de las variables analógicas mediante algoritmos de control tales como lazos PID. La segunda capa es el supervisor de eventos discretos donde se determina el estado del proceso y se establecen las acciones que permiten al proceso mantenerse en un estado deseado. La capa superior es un sistema conformado con agentes capaces de manejar las restricciones, conocer los cambios de metas, cambiar los procedimientos para elaborar un producto, informar si una meta ha sido alcanzada y negociar con otras unidades de producción. El proceso de negociación le permite a la unidad de producción cooperar para alcanzar la meta en común del sistema al que pertenece.

Cada capa cumple sus funciones e intercambia información con las otras dos. De las variables analógicas se extraen eventos que sirven de entrada al supervisor y al sistema multiagentes, a su vez las acciones que se generan en el supervisor cambian los parámetros en los algoritmos del control regulatorio. De manera similar el sistema multiagentes maneja un modelo de eventos discretos que indica el modo de operación en el que se encuentra la planta, los eventos de entrada a este modelo provienen del modelo de la planta que se encuentra en el supervisor, además el sistema multiagentes genera acciones que modifican la conducta del supervisor y del control regulatorio.

El intercambio de información entre las capas puede ser de diferentes formas dependiendo de la arquitectura donde se implemente el sistema información y toma de decisiones, cada capa puede residir en equipos diferentes o en el mismo equipo de control. Durante la etapa de diseño se debe definir el intercambio de información entre las diferentes capas para incluir la estructura de datos necesaria para este fin. En el sistema de información y toma de decisiones no es obligatorio la existencia

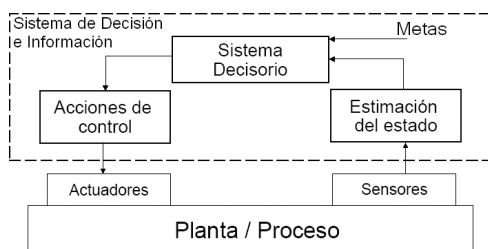


Figura 5. Ciclo de toma de decisiones.

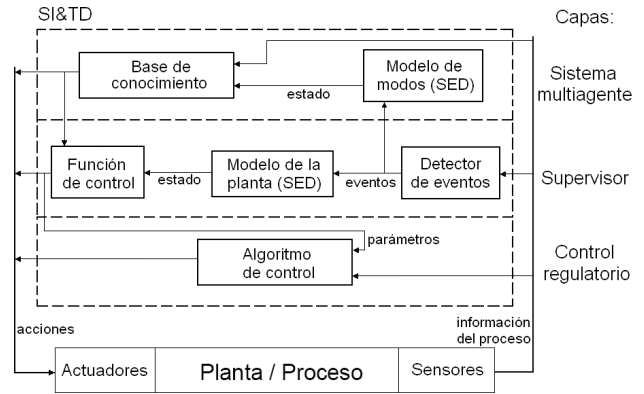


Figura 6. Capas del sistema de decisión e información en conexión con el proceso.

de las tres capas, se puede prescindir de una o dos capas para aplicaciones simples.

El ciclo de toma de decisiones de la figura 5 se repite en todas las capas del sistema de información y toma de decisiones (Chacón *et al.*, 2008). El sistema decisorio para el control regulatorio es un algoritmo de control con una variable de proceso como entrada la cual se quiere mantener igual a una consigna o referencia, en el supervisor es un modelo de eventos discretos y una función de control que tienen como entrada la secuencia de eventos del proceso y para el sistema multiagente es un conjunto de reglas lógicas con los valores de las premisas como entrada. En cada capa se pueden presentar situaciones no deseadas, en el control regulatorio una diferencia muy grande entre la variable de proceso y la consigna o referencia genera una alarma, en el supervisor determinados estados indican condiciones de operación fuera de la región permitida y el sistema multiagente combina la información de las capas inferiores con la información que llega (o no llega) de otras unidades de producción para determinar condiciones de fallas locales o problemas en las negociaciones.

El diseño del control de regulatorio y el sistema multiagentes son tratados en otros trabajos (Kowalski, 2010; Russell and Norvig, 2009). En la siguiente sección se explicará como modelar e implementar el supervisor de eventos discretos.

3. SUPERVISOR DE EVENTOS DISCRETOS

El diseño del supervisor varía según la información que se tenga de la planta, su dinámica, el modelo que se derive de ella, los eventos que se puedan observar y los eventos que se puedan controlar. Son bien conocidos los trabajos de síntesis de supervisores para sistemas de eventos discretos basados en autómatas (Ramadge and Wonham, 1987) donde el supervisor mantiene la conducta del proceso en una región deseada habilitando y deshabilitando eventos según el estado en que se encuentre.

3.1 Modelo del supervisor

El supervisor propuesto está formado por un detector de eventos, un modelo de la planta y una función de control. El ciclo de decisión de la unidad de producción mostrado en la figura 5 se realiza dentro del supervisor de la siguiente forma: las señales provenientes de los sensores son transformadas en eventos que sirven de entrada al modelo de la planta, a su vez la función de control ejecuta las acciones dependiendo del estado del modelo. De este ciclo se deriva la siguiente definición:

Definición 1: Un supervisor (S) está formada por la siguiente tupla:

$$S = (P, T, D^+, D^-, \mu, \Sigma, e, I, O, h, y)$$

$P = \{p_1, p_2, \dots, p_n\}$ es un conjunto finito de lugares,

$T = \{t_1, t_2, \dots, t_m\}$ es un conjunto finito de transiciones,

$D^+ \in \mathbb{Z}^{n \times m}$ es la matriz de los lugares de salida,

$D^- \in \mathbb{Z}^{n \times m}$ es la matriz de los lugares de entrada,

μ es un vector de dimensión $n \times 1$ que contiene la marcación de la red de Petri,

Σ es el conjunto de los eventos,

$e : \Sigma \rightarrow T$ es la función que asocia un evento a una transición,

I es el conjunto de las entradas al supervisor,

O es el conjunto de salida del supervisor,

$h : I \rightarrow \Sigma$ es una función que relaciona a las entradas con los eventos,

$y : \mu \times I \rightarrow O$ es la función de control que relaciona la marcación de la red de Petri y las entradas con la salida.

Para que el modelo esté completo se debe contar con el estado inicial del mismo que se representa con la marcación inicial (μ_0).

Los equipos que forman parte de la unidad de producción se mencionan a continuación:

- *Equipos de control.* Donde residen los algoritmos y estrategias para controlar el proceso. Ejemplo de equipos de control son los controladores lógicos programables, computadores personales, sistemas de control distribuidos entre otros.
- *Equipos de medición o sensores.* Pasan información a los equipos de control del estado del proceso. Ejemplo de estos equipos son: sensores de posición (detectores de proximidad), transmisores (presión, temperatura, flujo, nivel) entre otros.
- *Equipos de proceso.* Tienen como función darle valor agregado al producto. Ejemplo de equipos de proceso son los tanques, reactores, intercambiadores, entre otros.
- *Actuadores.* Las acciones del control se ejecutan en los actuadores y tienen como consecuencia un cambio de estado en el proceso. Ejemplo de actuadores son los motores y las válvulas.

Una lista que enumere los equipos y una lista con todas las señales de entradas y salidas deben formar parte de las especificaciones para el diseño del supervisor.

El modelo en el supervisor está compuesto por los modelos de los equipos proceso y actuadores que forman parte de la planta. Todas las transiciones controlables del modelo están en los actuadores. Existen una gran variedad de actuadores que pueden estar presentes en los procesos, en este trabajo se consideran cuatro tipos de uso muy común (Indriago and Lucena, 2004) que se diferencian entre ellos por el comportamiento de las entradas y salidas (se denominan entradas a las señales que van desde el actuador hacia el controlador y salidas las que van desde el controlador hacia el actuador). Los tipos de actuadores se describen a continuación:

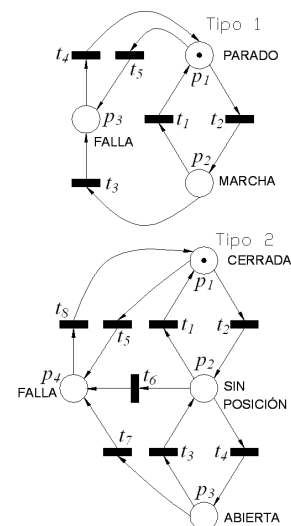


Figura 7. Modelo de los actuadores tipo 1 y 2.

- *Tipo 1. Una entrada y una salida.* El controlador envía un comando de salida y el actuador responde confirmando que lo ha recibido a través de una entrada. Un ejemplo típico de este tipo de equipos son los motores.
- *Tipo 2. Dos entradas y una salida.* Este tipo de actuadores tiene dos posiciones, cada entrada se activa para indicar que el actuador se encuentra en alguna de las dos posiciones. El controlador da un comando de salida para ir de una posición a otra y lo quita para regresar al estado inicial. Un ejemplo de estos actuadores son las válvulas con un solenoide de activación y reposición por resorte.
- *Tipo 3. Dos entradas y dos salidas.* Son muy similares al anterior, con la diferencia que para ir de una posición a otra se activa una de las salidas y para regresar se activa la otra salida. Un ejemplo de estos tipos de actuadores son las válvulas de dos solenoides.
- *Tipo 4. Una entrada y dos salidas.* Estos actuadores tienen dos modos de operación, cada modo es activado por una de las salidas. La confirmación de operación de ambos modos se recibe por la misma entrada. Un ejemplo de estos actuadores son los motores reversibles.

En la figura 7 se muestra el modelo de red de Petri de los actuadores tipo 1 y 2. En el actuador tipo 1 la transición t_2 se dispara cuando la entrada está presente y la transición t_1 se dispara cuando la entrada está ausente. Las transiciones t_3 y t_5 se disparan cuando ocurre una falla mientras que la transición t_4 se dispara cuando la falla desaparece.

Además de la lista de equipos y la lista de entradas y salidas, también debe haber una filosofía de operación como parte de las especificaciones, en ella se describe detalladamente como opera cada actuador en función del estado de la planta.

La filosofía de control, escrita en forma de restricciones y reglas, es usada junto con el modelo de la planta para obtener el modelo de lazo cerrado o supervisor, esto se hace por medio de algún método para síntesis de supervisores (Moody and Antsaklis, 1998; Martínez *et al.*, 2008). En la implementación del supervisor no se usa el modelo de lazo cerrado, éste se sustituye por el modelo de lazo abierto (modelo de la planta) y la representación en red de Petri de la filosofía de control la cual está formada por el conjunto de arcos, transiciones y lugares que quedan luego de quitar del modelo de lazo cerrado

los elementos del modelo de lazo abierto. De la representación en red de Petri de la filosofía de control se deduce la función de control la cual está conformada por una serie de premisas lógicas que reproducen las restricciones y reglas usadas para la obtención del supervisor y que tienen como entrada la marcación del proceso (μ) y las condiciones de entradas (I) y como salidas las acciones que se ejecutan sobre los actuadores (O). Cada actuador tiene asociado un conjunto de variables lógicas mediante las cuales se conoce su estado o se ejecuta una acción sobre él. Estas variables se mencionan a continuación:

- *Modo (MO)*. Cambia el modo del actuador entre automático (verdadero) y manual (falso).
- *Arranque en manual (AM)*. Se coloca en verdadero para un arranque en manual.
- *Arranque en automático (AA)*. Se coloca en verdadero para un arranque en automático.
- *Dirección (DI)*. Indica la dirección de acción en los actuadores de dos salidas.
- *Comando de salida (CS)*. Manipulado esta variable se logra la acción sobre el actuador.
- *Comando de salida inverso (CI)*. Manipulando esta variable se logra la acción sobre los actuadores de dos salidas.
- *Entrada 1 (NA)*. Indica el estado del actuador.
- *Entrada 2 (NC)*. En conjunto con la entrada 1 indica el estado del actuador en los actuadores de dos entradas.
- *Permisivos (PE)*. Agrupa el estado de todos los permisivos del actuador, en condiciones de operación tiene un valor de verdadero.
- *Alarmas (AI)*. Agrupa el estado de todas las alarmas del actuador excepto la de falla de respuesta al comando de salida, en condiciones de operación tiene un valor de verdadero.
- *Falla de respuesta al comando de salida (FR)*. Se coloca en verdadero cuando hay una inconsistencia entre los comandos de salidas y las entradas.
- *Alarma general (AL)*. Se coloca en verdadero cuando hay una alarma o falla de la respuesta al comando de salida.
- *Borrar alarmas (BO)*. Restablece los valores de *AI*, *FR* y *AL*.

Las variables de modo, arranque manual, dirección y borrar normalmente son manipulados en la estación de operación o de trabajo (estación que usa el operador para manejar el proceso). Las variables de arranque en automático y dirección cambian de estado en la función de control. El valor de las entradas 1 y 2 depende directamente del estado del actuador. Las variables comando de salida, comando de salida inverso, falla de respuesta de marcha, permisivos, alarmas y alarma general son determinadas por una función denominada función estándar. La función estándar es intrínseca del actuador y siempre va a estar presente en el controlador independientemente del proceso. La función estándar se encarga de ejecutar la acción, limpiar alarmas, verificar la respuesta de marcha y desactivar el actuador cuando ocurre una alarma, una parada de emergencia o un cambio de modo. A continuación se muestran las reglas que forman la función estándar expresadas en premisas lógicas para un actuador tipo 1 (una entrada y una salida) con j premisas y k alarmas:

- $((AM \wedge \neg MO) \vee (AA \wedge MO)) \wedge AL \wedge PE \Leftrightarrow CS$
- $(CS \wedge \neg NA) \vee (\neg CS \wedge NA) \Rightarrow FR$
- $FR \vee \neg AI \Rightarrow AL$
- $BO \Rightarrow \neg AL$
- $BO \Rightarrow \neg FR$

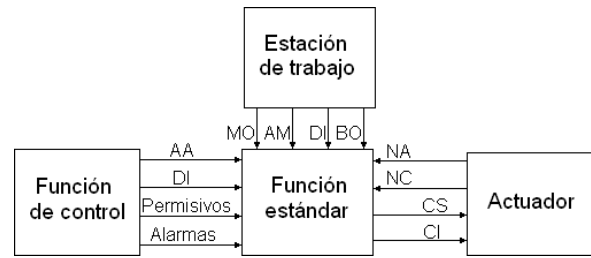


Figura 8. Flujos de las variables del actuador.

- $BO \Rightarrow AI$
- $MO \vee AL \vee PARADA_EMERGENCIA \Rightarrow \neg AM$
- $\neg MO \vee AL \vee PARADA_EMERGENCIA \Rightarrow \neg AA$
- $\neg PERMISIVO_1 \vee \dots \vee \neg PERMISIVO_j \Leftrightarrow PE$
- $\neg ALARMA_1 \vee \dots \vee \neg ALARMA_K \Rightarrow AI$

Algunas de las reglas varían dependiendo del tipo de actuador, la figura 8 muestra el flujo de las variables entre la función de control, estación de trabajo, la función estándar y el actuador.

La función de control se diseña de tal forma que la conducta del proceso sea la que se describe en filosofía de operación, cada acción que se ejecuta depende del tipo de actuador, para los actuadores tipo 1 se ejecutan las acciones de marcha y parada, para los tipo 2 y 3 se ejecutan las acciones de abrir y cerrar o la acción de cambio de posición y para los tipo 4 se ejecutan las acciones de marcha directa, marcha inversa y parada. En la función de control también se especifican los permisivos y alarmas de cada actuador. A continuación se muestra un ejemplo de una función de control para un sistema con n actuadores:

*actuador*₁ (Tipo 1):

- condición de marcha $\Rightarrow AA$
- condición de parada $\Rightarrow AA$
- lista de permisivos
- lista de alarmas

*actuador*₂

⋮

actuador _{n}

La condición de marcha y la condición de parada son variables lógicas obtenidas como resultado de operaciones lógicas y de comparación que tienen como argumentos la marcación de la red de Petri (μ) y elementos del conjunto de entradas (I). Cada permiso de la lista es una variable lógica, si algún permiso de la lista toma el valor de verdadero entonces la variable *PE* adquiere un valor de falso indicando que el actuador no tiene permiso. Las alarmas funcionan de forma similar, si alguna de las alarmas adquiere el valor de verdadero la variable *AI* toma el valor de falso, esto causa que el equipo se detenga.

En el proceso de diseño del supervisor se debe analizar el modelo de la planta para determinar estados indeseados como bloqueos (Murata, 1989; David and Alia, 2005), las propiedades de alcance y vivacidad de las redes de Petri son útiles para este fin. El análisis del modelo permite mejorar la filosofía de operación lo que se traduce en una función de control más eficiente.

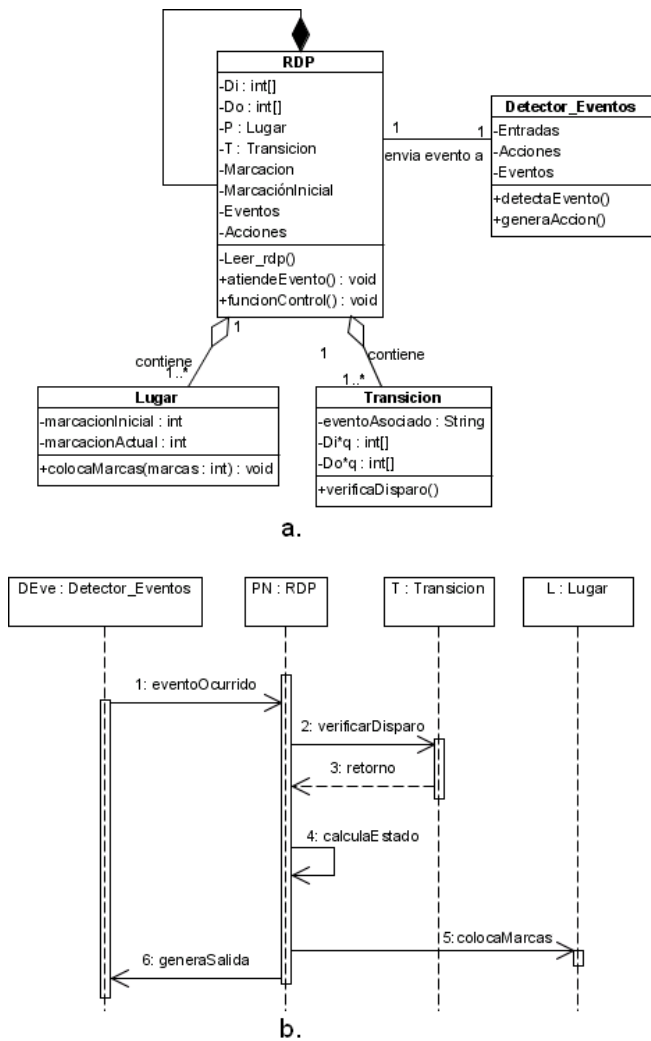


Figura 9. a. Diagrama de clases, b. Diagrama de secuencias.

4. IMPLEMENTACIÓN DEL SUPERVISOR

El supervisor se debe programar en un equipo de control como por ejemplo un computador personal o controlador lógico programable. En la figura 9 se muestra como se relacionan los elementos del supervisor y la conducta del mismo ante la llegada de los eventos, estos diagramas se pueden implementar en un programa orientado a objetos y ser ejecutados en un computador. En el diagrama de clases se pueden ver como atributos o métodos todos los elementos presente en la definición del supervisor. Un objeto de la clase Rdp contiene el modelo de la planta y unos de sus métodos es la función de control. A través del detector de eventos entran y salen los datos del supervisor, los cuales pueden provenir de diversas fuentes tales como base de datos, manejador de comunicaciones o directamente de una red.

Es común utilizar gráficos de función secuencial cuando el supervisor se implementa en un controlador lógico programable (Brinksma *et al.*, 2002; David, 1995; Hellgren, 2002), sin embargo no todos ellos cuentan con esta herramienta, por esta razón se presenta una vía para implementar el supervisor usando diagrama escalera, la mayoría de los controladores lógicos programables que existen en el mercado pueden ser programados usando este lenguaje.

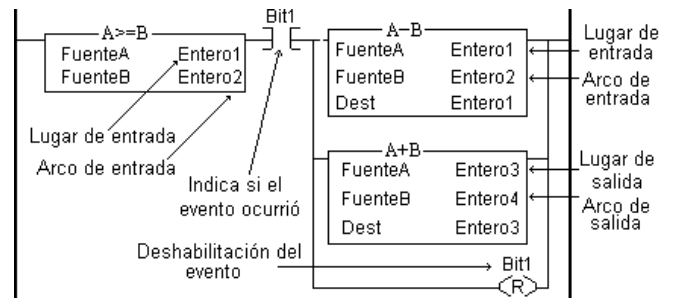


Figura 10. Diagrama escalera de una transición con un lugar de entrada y un lugar de salida.

Tabla1. Relación entre la red de Petri y la memoria del PLC.

Elemento de la RDP	Memoria del PLC
Lugar	Variable entera cuyo valor indica la cantidad de marcas actual en el lugar.
Arco	Variable entera cuyo valor representa el peso del arco.
Transición	Peldaño del programa que contiene instrucciones de comparación, sumas, restas, e instrucciones de bit.
Evento	El evento no tiene representación en la memoria del controlador, sin embargo, hay una variable binaria asociada a cada transición que indica que el evento ha ocurrido

La tabla 1 muestra la relación que se usa para implementar la red de Petri en el controlador lógico programable. Con esta asignación una transición t_j corresponde a un peldaño del programa escalera, el cual tendrá como instrucciones de entrada tantas instrucciones de comparación del tipo “mayor o igual que” (\geq) como lugares de entrada tenga la transición, además una instrucción “contacto normalmente abierto” (-) [-] con la dirección del bit que indica si el evento asociado con la transición ha ocurrido. La relación entre las transiciones y los eventos está indicada por la función e de la definición 1. Las instrucciones de salida del peldaño del programa escalera que representa la transición t_j incluirán tantas instrucciones de resta como lugares de entradas tenga la transición, tantas instrucciones de suma como lugares de salida tenga la transición y una instrucción de restablecimiento (-R-) para borrar el bit que indica que el evento que disparó la transición ha ocurrido. Las instrucciones de resta eliminan tantas marcas del lugar de entrada como peso tenga el arco que lo une a la transición y las instrucciones de suma agregan tantas marcas al lugar de salida como peso tiene el arco que lo une a la transición, estas operaciones se corresponde al cálculo de la marcación en el diagrama de secuencia de la figura 9. La figura 10 ilustra una transición en lógica escalera.

Se debe incluir en el programa escalera una rutina para la detección de eventos, en esta rutina se asocian los elementos del conjunto I con los elementos del conjunto Σ tal como indican la función h . El programa también debe contar con una rutina en donde se ejecuta la función l de control y . En cada controlador puede haber más de una red de Petri, cada una representa diferentes unidades de producción. Todos los elementos usados para implementar el modelo de eventos discreto en el controlador lógico programable (diagrama escalera, instrucciones de suma, resta, comparación, contacto normalmente abierto y bobina de

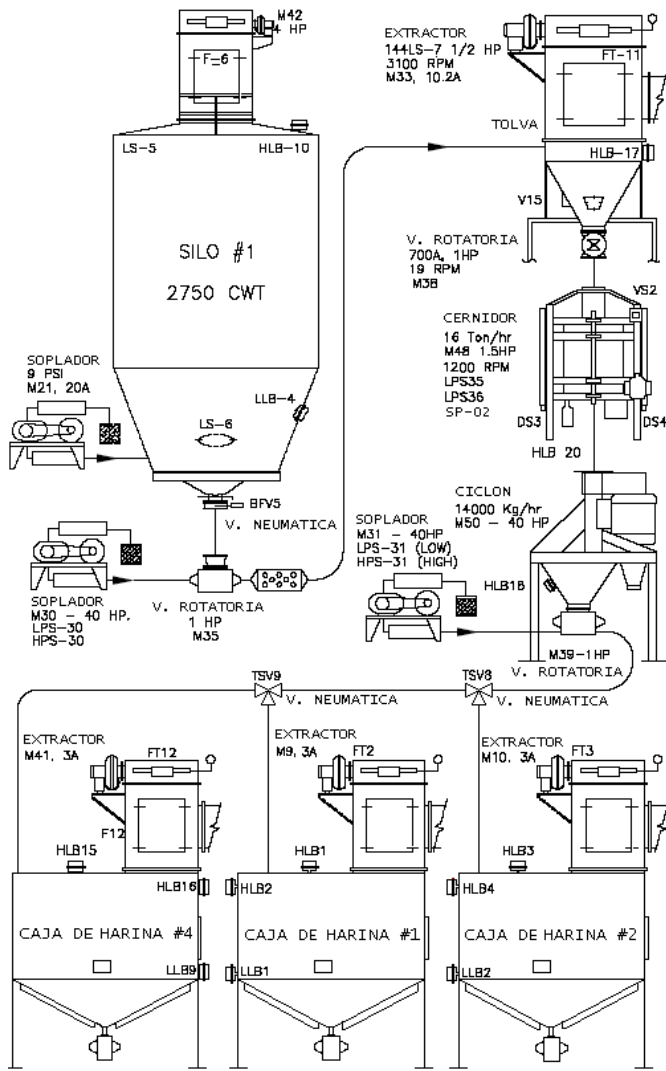


Figura 11. Proceso de transporte de harina.

restablecer) están contemplados en la norma IEC 61131-3 (IEC, 2003).

5. CASO DE ESTUDIO

El caso de estudio se tomó de en una fábrica de galletas donde se implementó un supervisor como el propuesto, la figura 11 muestra el diagrama de proceso del sistema que tiene como objetivo transportar harina desde un silo hasta los recipientes que alimentan las líneas de fabricación.

Para su traslado se deja caer la harina en una tubería a la que se inyecta aire con un soplador, la harina es trasladada a una tolva que alimenta a un cernidor y de allí a un ciclón para luego ser transportada por otro soplador hasta las cajas de harina. En la figura 12 se muestra el modelo de la planta.

5.1 Arquitectura de Sistema

Para manejar el sistema se cuenta con un controlador lógico programable como equipo de control, éste se comunica por medio de una red serial (ethernet) a una interfaz gráfica de operación que cuenta con las siguientes funciones:

1. Seleccionar las cajas de harinas que van a ser llenadas.

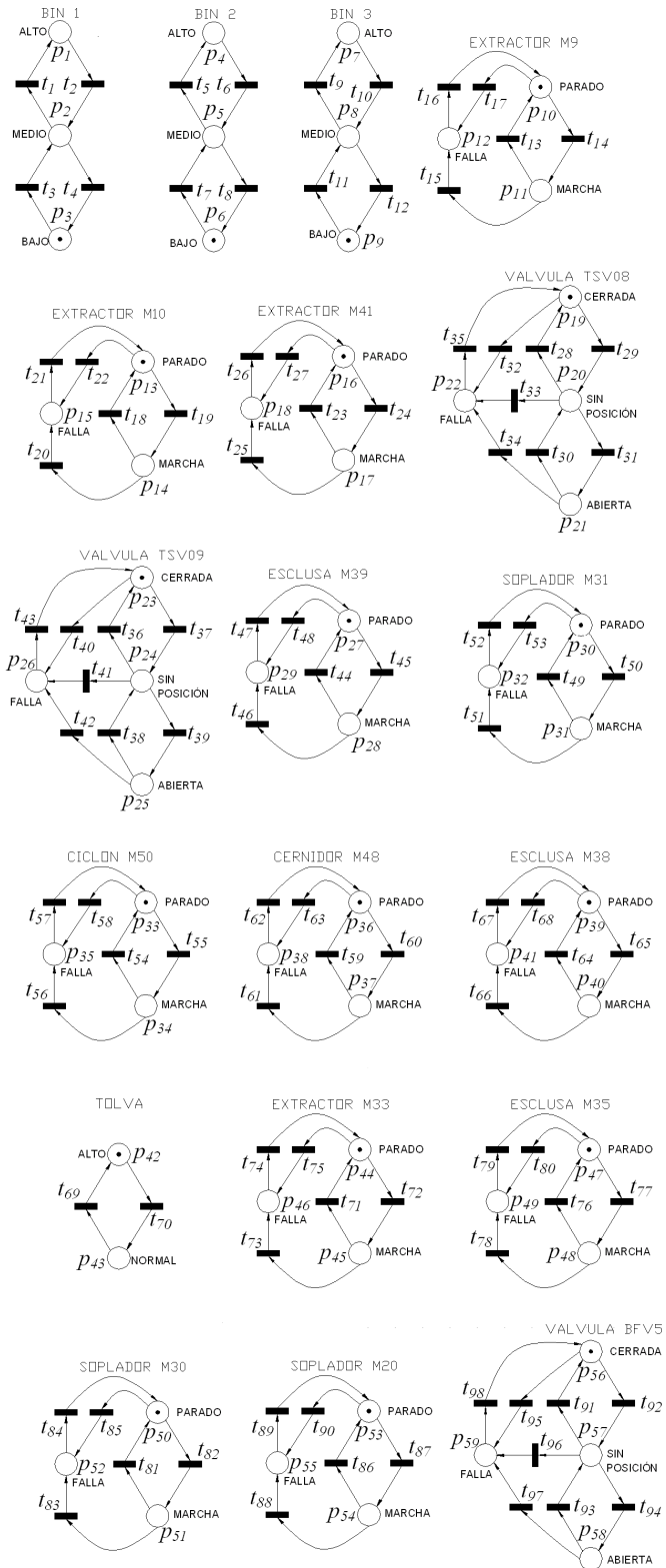


Figura 12. Modelo en redes de Petri del proceso de transporte de harina.

2. Mostrar el estado de los equipos que forman parte del sistema.
3. Mostrar las alarmas.

Los motores de los sopladores M30 y M31 y el ciclón M50 son manejados con arrancadores suaves el resto de los motores son manejados con arrancadores electromecánicos a los que se

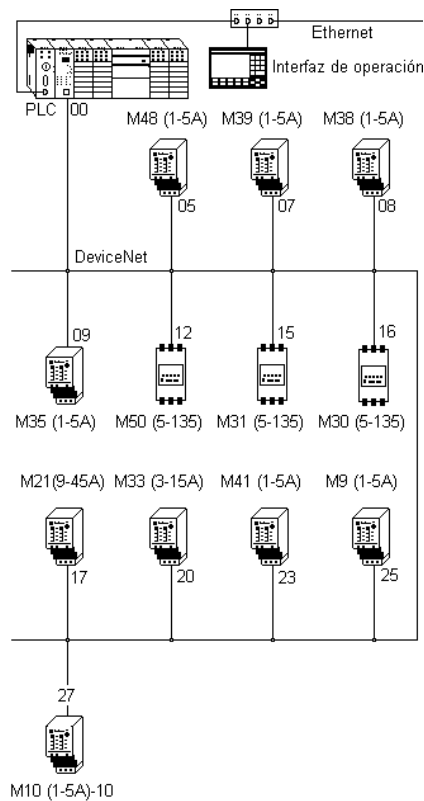


Figura 13. Arquitectura del sistema de transporte de harina.

Se le adiciona un circuito de protección térmica con capacidad de entrada/salida. Tanto los arrancadores suaves como los circuitos de protección térmica tienen capacidad para conectarse a una red de campo (devicenet). El resto de las señales de campo tales como sensores de presión, nivel, solenoides de las válvulas, botones para la operación local de los motores entre otras llegan por medio de tarjetas de entradas y salidas montadas en el bastidor local del controlador lógico programable. La figura 13 muestra la arquitectura de control del sistema de transporte de harina.

5.2 Filosofía de operación

Para la operación del sistema existen tres selectores, uno por caja, cada selector indica si la respectiva caja se va a llenar o no. La petición de llenado para cada caja aparece cuando el selector de la caja indica que se va a llenar y la misma tiene nivel bajo. La petición de llenado desaparece si el selector respectivo indica que la caja no debe llenarse o si la caja tiene un nivel alto de harina. No se pueden llenar dos cajas de harina al mismo tiempo. Cada caja de harina puede tener nivel alto, normal y bajo. Los niveles se determinan por medio de dos sensores, uno colocados para indicar el 90 % de la capacidad de llenado y el otro para indicar el 50 % de la capacidad de llenado. Los equipos del sistema arrancan de forma secuencial, primero los equipos que están cerca de las cajas de harina y de último los equipos que están cerca del silo. El objetivo principal del sistema de control es mantener a las tres cajas en nivel normal. Si ocurre una parada imprevista en algún equipo todos los equipos que están hacia el silo de harina deben detenerse de forma inmediata para evitar derrame de la materia prima. La operación de los equipos se describe a continuación:

Extractor del filtro de la caja de harina 1 (M9)

- Condiciones de marcha:
 - Petición de llenado de la caja 1.
- Condición de parada:
 - No hay petición de llenado de la caja 1 y han pasado 3 segundos que la válvula TSV9 se colocó en la posición para llenar la caja 4.

Extractor del filtro de la caja de harina 2 (M10)

- Condición de marcha:
 - Petición de llenado de la caja 2.
- Condición de parada:
 - No hay petición de llenado de la caja 2 y han transcurrido 3 segundos que la válvula TSV8 se colocó en la posición para llenar la caja 4.

El extractor del filtro de la caja de harina 4 (M41)

- Condición de marcha:
 - Petición de llenado de la caja 4.
- Condición de parada:
 - No hay petición de llenado de la caja 4 y han transcurrido 3 segundos después de se detuvo M33.

Válvula neumática de la caja de harina 1 (TSV9) en condiciones de reposo se encuentra hacia la dirección de la caja 4.

- Condición para posición hacia la caja 1:
 - Petición de llenado en la caja 1.
- Condición para posición hacia la caja 4:
 - No hay petición de llenado de la caja 1 y han transcurrido tres segundos desde que paro el extractor del filtro de la tolva superior o
 - No hay petición de llenado de la caja 1 y hay una petición de llenado de la caja de harina 2 o
 - No hay petición de llenado de la caja 1 y hay una petición de llenado de la caja de harina 4.

Válvula neumática de la caja de harina 2 (TSV8) en condiciones de reposo se encuentra hacia la dirección de la caja 4.

- Condición para posición hacia la caja 2:
 - Petición de llenado de la caja 2.
- Condición para posición hacia la caja 4:
 - No hay petición de llenado de la caja 2 y han transcurrido tres segundos desde que paro el extractor del filtro de la tolva superior o
 - No hay petición de llenado de la caja 2 y hay una petición de llenado de la caja de harina 1 o
 - No hay petición de llenado de la caja 2 y hay una petición de llenado de la caja de harina 4.

Extractor del filtro de la tolva superior (M33)

- Condiciones de marcha:
 - La caja 1 tiene petición de llenado, TSV9 está dirigida hacia la caja 1, TSV8 está dirigida hacia la caja 4 y M09 está en marcha o
 - La caja 2 tiene petición de llenado, TSV8 está dirigida hacia la caja 2 y M10 está en marcha o
 - La caja 2 tiene petición de llenado, TSV9 y TSV8 están dirigidas hacia la caja 4 y M41 está en marcha.
- Condición de parada:
 - No hay petición de llenado de ninguna caja y han transcurrido tres segundos de haberse detenido el soplador M31.

Soplador (M31)

- Condición de marcha:
 - Hay petición de llenado y han pasado tres segundos que el extractor del filtro de la tolva superior (M33) se puso en marcha.
- Condiciones de parada:
 - No hay petición de llenado y han transcurrido tres segundos de haberse detenido la válvula rotatoria del ciclón.

Válvula rotatoria del ciclón (M39)

- Condiciones de marcha:
 - Hay petición de llenado y hay presión en el soplador M31.
- Condición de parada:
 - Si no hay petición de llenado y hace tres segundos que el ciclón (M50) se ha detenido o
 - El soplador M31 está detenido.

Ciclón (M50)

- Condición de marcha:
 - Hay petición de llenado y han pasado 5 segundos luego que la válvula rotativa (M39) se puso en marcha.
- Condición de parada:
 - No hay petición de llenado y han transcurrido tres segundos de haberse detenido el cernidor (M48) o
 - La válvula rotatoria del ciclón está parada.

Cernidor (M48)

- Condición de marcha:
 - Hay una petición de llenado y han transcurrido 5 segundos desde que el ciclón (M50) se ha puesto en marcha.
- Condición de parada:
 - No hay petición de llenado y han transcurrido 60 segundos de haberse detenido la válvula rotatoria de la tolva superior (M38) o
 - El ciclón (M50) está detenido o
 - Se ha activado el interruptor de vibración del cernidor.

Válvula rotatoria de la tolva superior (M38)

- Condición de marcha:
 - Hay una petición de llenado y han transcurrido tres segundos de la puesta en marcha del cernidor (M48).
- Condición de parada:
 - No hay petición de llenado y han transcurrido 3 segundos de haberse parado el soplador M30 o si el cernidor (M31) está detenido.

Soplador del silo (M30)

- Condición de marcha:
 - Hay una petición de llenado y han pasado 3 segundos de haberse puesto en marcha la válvula rotatoria de la tolva superior (M38).
- Condición de parada:
 - No hay petición de llenado y pasaron 20 segundos de haberse detenido la válvula rotatoria del silo M35 o
 - Tolva superior está llena y han pasado 20 segundos de haberse parado M35.

Soplador (M21)

- Condición de marcha:

- Hay una petición de marcha y han transcurrido tres segundos de haberse puesto en marcha el soplador M30.
- Condición de parada:
 - No hay petición de llenado y la válvula rotativa del silo (M35) está parada o cuando la tolva superior está llena.

Válvula rotativa del silo (M35)

- Condiciones de marcha:
 - Hay una petición de llenado y los sopladores M30 y M21 están en marcha.
- Condiciones de parada:
 - Han transcurrido cinco segundos después de que cerró la válvula BFV5 y no hay petición de llenado o
 - Han transcurrido cinco segundos después de que cerró la válvula BFV5 y se llenó la tolva superior o
 - El soplador M31 está detenido.
- Permisivos
 - 7 segundos dosificando y 7 segundos sin dosificar.

Válvula neumática (BFV5)

- Condiciones para abrir:
 - Hay petición de llenado y han pasado tres segundos de la puesta en marcha de la válvula rotatoria M35.
- Condiciones para cerrar:
 - No hay petición de llenado o
 - El soplador M30 está parado.

Además las siguientes señales hacen que desaparezca la petición de llenado de las cajas lo que provoca una parada secuencial del sistema.

- Nivel alto en la tolva superior (HLB-17).
- Interruptor de vibración del cernidor (VS2).
- Interruptor de presión de aire para instrumento del cernidor (SP-02).
- Interruptor de alta presión del soplador M30 (HPS-30).
- Interruptor de baja presión del soplador M30 (LPS-30).
- Interruptor de alta presión del soplador M31 (HPS-30).
- Interruptor de baja presión del soplador M31 (LPS-30).
- La señal de alarma general (AL) de cualquier actuador del sistema.

5.3 Función de control

La función de control para los actuadores M35 y TSV9 se muestra en la figura 14, en ella se puede observar la estrecha relación que existe con la filosofía de operación, se agregaron temporizadores para lograr los tiempos de la secuencia de arranque.

Las etiquetas relacionadas con las variables de los actuadores tienen el formato *An_nombre_descripción*, donde *n* es el número del actuador, *nombre* se refiere al nombre del actuador y la *descripción* está asociada a las variables del actuador, por ejemplo la etiqueta A1_M9_AA se refiere a la variable arranque en automático del motor M9, el cual es el primer actuador en la función de control.

La gráfica de la figura 15 muestra la operación de los equipos del proceso, en ella se puede observar como el sistema de control responde para mantener las tres cajas en el nivel normal, cuando una caja alcanza el nivel bajo las válvulas TSV8 y

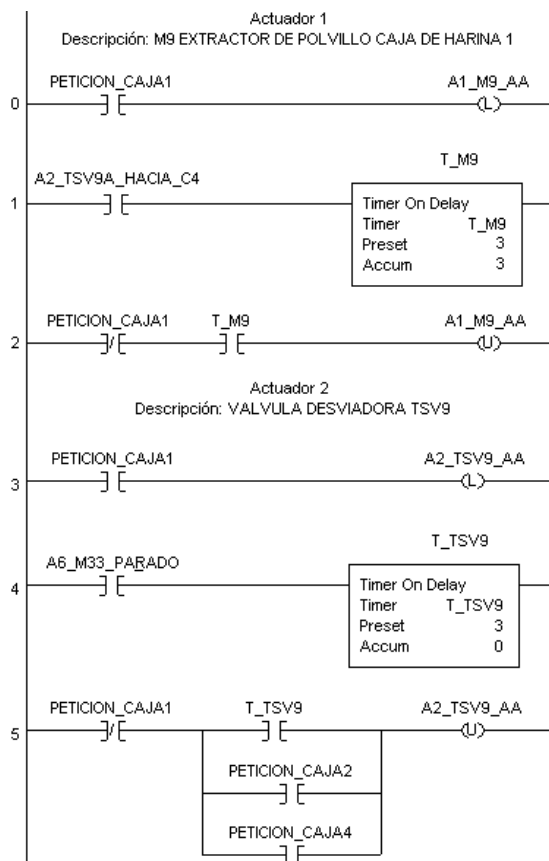


Figura 14. Función de control para los actuadores 1 y 2 del sistema de transporte de harina

TSV9 se acomodan para llenar esta caja. Una altura de 50 % para el sensor inferior garantiza para el caso en que haya más de una caja de harina con nivel bajo que el sistema pueda llenarlas antes de que alguna de ellas se quede sin harina. En la figura también se puede ver el arranque secuencial de los equipos en el siguiente orden: extractor del filtro de la caja a llenar (M9, M10 o M31), válvulas de posición (TSV8 y TSV9), extractor del filtro de la tolva superior (M33), válvula rotatoria del ciclón (M39), soplador del ciclón (M31), ciclón (M50), cernidor (M48), válvula rotatoria de la tolva superior (M38), soplador de transporte del silo (M30), soplador de fluidización del silo (M21), válvula rotatoria del silo (M35) y válvula de salida del silo (BFV5), la secuencia de parada ocurre en orden inverso cuando los tres silos tienen nivel normal. Además, en la figura 15 se observa como responde el supervisor ante una falla (indicada con el evento F) en la válvula TSV9 mientras se llena la caja de harina 1. La falla tiene como resultado una parada secuencial y cuando es corregida (evento B) se produce un arranque secuencial para llenar la caja de harina 2 debido a que ésta indicaba nivel bajo.

Este caso de estudio no tiene elementos en la capa de control regulatorio, si se quisiera agregar algún elemento a esta capa, un lazo de control PID por ejemplo, éste podría programarse en el PLC y la comunicación con el supervisor se haría directamente a través de la memoria de datos del controlador, una marcación determinada puede tener como acción el cambio del algún parámetro en el algoritmo de control, por ejemplo cambiar la consigna del lazo PID. La comunicación con el computador donde se ejecuta el sistema multiagente se hace a través de la red ethernet, los datos enviados tienen información referente a

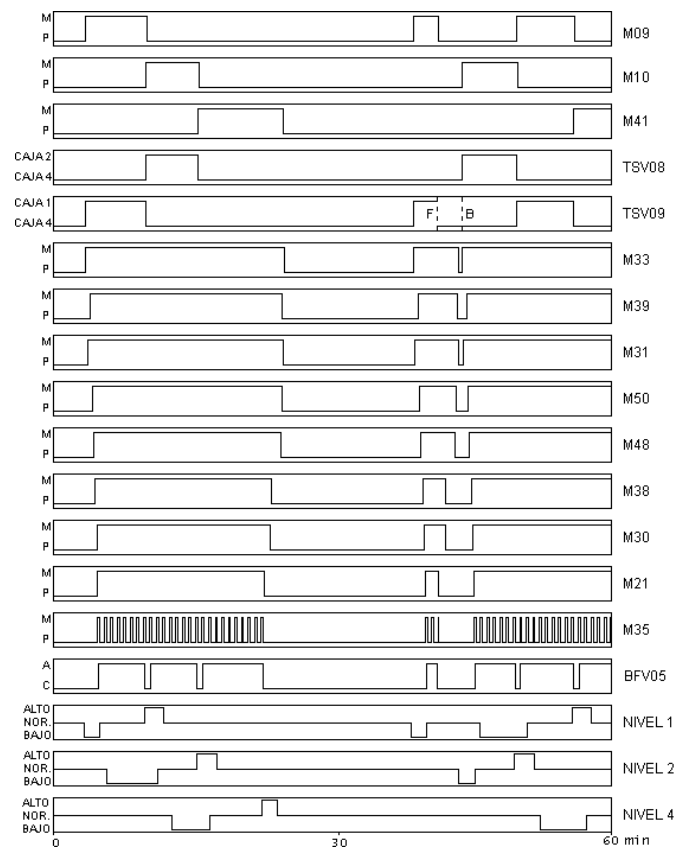


Figura 15. Resultados del sistema de control para el transporte de harina. M: marcha, P: parado, A: abierta, C: cerrada.

la cantidad de cajas de harina que están en operación, la corriente de los motores (usada para para el mantenimiento predictivo) y el estado del sistema (falla, parado o en operación). El sistema multiagente puede colocar o quitar la petición de llenado de cualquiera de las cajas.

6. CONCLUSIONES

Este trabajo presenta como modelar e implementar el supervisor de eventos discretos que forma parte del sistema de control de una unidad de producción. Las unidades de producción se integran para formar complejos de producción y éstos a su vez se integran para formar la empresa. La unidad de producción se divide en tres capas formadas por el control regulatorio, supervisor y sistema multiagente. La definición del supervisor que se presenta contiene un modelo de la planta, una función de control y una interfaz para intercambiar información con el proceso, esta definición permite la integración con las capas control regulatorio y sistema multiagente. La implementación se hace introduciendo el modelo del supervisor en un equipo de control tal como un computador personal o en un controlador lógico programable. En la metodología se detalla el modelo de algunos tipos de actuadores.

Además, este trabajo sirve como base para el desarrollo de una herramienta que permita modelar, hacer síntesis de supervisores, simular y generar el código para controladores lógicos programables en un sólo ambiente lo que permite hacer los cambios de forma rápida y eficiente, esto hace más flexible a la Arquitectura de Integración para la Automatización de Complejos de Producción lo que se traduce en ahorro de tiempo

y dinero en el desarrollo de la automatización y reconfiguración del sistema de control en procesos industriales.

REFERENCIAS

- Brinksma, E., A. Mader and A. Fehnker (2002). Verification and optimization of a plc control schedule. *International Journal on Software Tools for Technology Transfer* **4**(1), 21–53.
- Chacón, E., I. Besembel, F. Narciso, J. Montilva and E. Colina (2002). An integration architecture for the automation of a continuous production complex. *ISA transaction* **41**, 95–113.
- Chacón, E., I. Besembel, M. Rivero and J. Cardillo (2008). *Robotics, Automation and Control*. Chap. The Holonic Production Unit: an Approach for an Architecture of Embedded Production Process. 1ra ed.. I-Tech. Education and Publishing.
- David, R. (1995). Grafcet: A powerful tool for specification of logic controllers. *IEEE Trans. on Control Systems Technology* **3**(3), 253–268.
- David, R. and H. Alia (2005). *Discrete, Continuous, and Hybrid Petri Nets*. Vol. 3-540-22480-7. Springer. Berlin.
- Frey, G (1998). Simulation of hybrid systems based on interpreted petri net. In: *Proceedings of the IEE International Conference on Simulation - Innovation Throught Simulation*. Vol. 30.09. York. pp. 168–175.
- Hellgren, A. (2002). On the Implementation of Discrete Event Supervisory Control with focus on flexible manufacturing systems. Phd thesis. Katholieke Universiteit Leuven, Departamento de Ingeniería Mecánica. Control and Automation Laboratory. Department of Signals and Systems. Chalmers University of Technology. Göteborg, Sweden.
- IEC (2003). *Standard 61131. Programmable controllers Part 3: Programming languages*. 2nd ed.
- Indriago, M. and Z. Lucena (2004). Optimización en el desarrollo de un programa para plc. *Revista Ingeniería UC*. **11**, 70–78.
- ISA (1995). *ANSI/ISA-S88.01-1995. Batch Control Part 1: Models and Terminology*.
- Kowalski, R. (2010). *How to be Artificially Intelligent - Computational Logic and Human Life*. Department of Computing Imperial College London. Draft: <http://www.doc.ic.ac.uk/~rak/papers/newbook.pdf>.
- Lennartson, Bengt, Bo Egardt and Michael Tittus (1994). Hybrid systems in process control. In: *In Proc. of the 33rd CDC*. pp. 3587–3592.
- Li, H. (1994). A Formalization and Extension of the Purdue Enterprise Reference Architecture and the Purdue Methodology. Ph.d. thesis. Purdue University, West Lafayette. USA.
- Martinez, C., A. Atto and S. Amari (2008). Supervision of an industrial plant subject to a maximal duration constraint. In: *Proceedings of the 9th International Workshop on Discrete Event Systems*. Göteborg, Sweden. pp. 254–259.
- Moody, J. and P. Antsaklis (1998). *Supervisory control of discrete event systems using Petri net*. 1ra ed.. Kluwer Academic Publisher. Norwell, USA.
- Murata, T. (1989). Petri nets: Properties, analysis and applications. *IEEE, Proc* **77**(4), 541–580.
- Peterson, J. (1981). *Petri Net Theory and the Modeling of Systems*. 1ra ed.. Perntice Hall. USA.
- Polic, A. and K. Jezernik (2005). Closed-loop matrix based model of discrete event systems for machine logic control design. *IEEE Transactions on Industrial Informatics* **1**(1), 39–46.
- Ramadge, P. J. and W. M. Wonham (1987). Supervisory control of a class of discrete-event processes. *SIAM Journal of Control and Optimization* **25**(5), 206–230.
- Russell, S. and P. Norvig (2009). *Artificial Intelligence: A Modern Approach*. 3ra ed.. Prentice Hall. USA.
- Saad, A., K. Kawamura and G. Biswas (1995). Evaluation of contract net-based heterarchical scheduling for flexible manufacturing systems. In: *In Proceedings of the 1995 International Joint Conference on Artificial Intelligence (IJCAI'95), Workshop on Intelligent Manufacturing*. pp. 310–321.
- Silveira, M. and M. Combacau (2003). Supervision and control of heterarchical discrete event systems: The laas approach. In: *Journal of SBA - Sociedade Brasileira de Automática*.
- Wyns, J. (1999). Reference Architecture for Holonic Manufacturing Systems. Phd thesis. Katholieke Universiteit Leuven, Departamento de Ingeniería Mecánica. Leuven, Bélgica.
- Zapata, G. and E. Carrasco (2002). Estructuras generalizadas para controladores lógicos modeladas mediante redes de petri. *DYNA* (135), 65–74.