



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE HIDALGO

INSTITUTO DE CIENCIAS BÁSICAS E INGENIERÍA
ÁREA ACADÉMICA DE INGENIERÍA
CENTRO DE INVESTIGACIÓN AVANZADA EN
INGENIERÍA INDUSTRIAL

Validación de Algoritmos de Control en Diagrama
Escalera para Detección de Condiciones de Riesgo con
Redes de Petri.

T E S I S

QUE PARA OBTENER EL GRADO DE
DOCTOR EN CIENCIAS EN
INGENIERÍA INDUSTRIAL

P R E S E N T A :

JOSÉ CARLOS QUEZADA QUEZADA

DIRECTOR DE TESIS: DR. JOSELITO MEDINA MARÍN

CODIRECTOR: DR. JUAN CARLOS SECK TUOH MORA

MINERAL DE LA REFORMA, HGO., SEPTIEMBRE DE 2015.



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE HIDALGO

INSTITUTO DE CIENCIAS BÁSICAS E INGENIERÍA

DIRECCIÓN

ICBI/AAI/064/2015

M. EN A. JULIO CÉSAR LEINES MEDÉCIGO
DIRECTOR DE ADMINISTRACIÓN ESCOLAR
PRESENTE

Por este conducto le comunico que el jurado asignado al pasante, del Doctorado en Ciencias en Ingeniería Industrial, C. José Carlos Quezada Quezada, con número de cuenta 038840, quien presenta el trabajo de tesis titulado "Validación de Algoritmos de Control en Diagrama Escalera para Detección de Condiciones de Riesgo con Redes de Petri", después de revisar el trabajo en reunión de sinodales, ha decidido autorizar la impresión del mismo una vez realizadas las correcciones que fueron acordadas.

A continuación se anotan las firmas de conformidad de los integrantes del jurado.

- PRESIDENTE: Dr. Ernesto Flores García
SECRETARIO: Dr. Juan Carlos Seck Tuoh Mora
VOCAL: Dr. Joselito Medina Marín
SUPLENTE: Dr. Norberto Hernández Romero

Handwritten signatures of the jury members on lines.

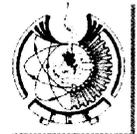
Sin otro particular, reitero a Usted la seguridad de mi atenta consideración.

ATENTAMENTE
"AMOR, ORDEN Y PROGRESO"
MINERAL DE LA REFORMA, HGO., A 08 DE SEPTIEMBRE DE 2015.

DR. ORLANDO ÁVILA POZOS
DIRECTOR DEL ICBI



Ciudad Universitaria Carretera Pachuca-Tulancingo
Km. 4.5. s/n Col. Carboneras C.P. 42184
Mineral de la Reforma, Hidalgo. México.
Tel: (771)7172000 ext. 2230 , 2231



Agradecimientos

Agradezco a Dios por la oportunidad de la vida y por ende por todo lo que me ha dado.

Nunca dejare de agradecer a mi madre en el cielo por su consejo “no dejes de estudiar”, el último día que la vi con vida, gracias.

A mi razón de ser, mi familia quien ha soportado de forma humilde y respetuosa la decisión que conlleva el continuar preparándose. A Lore mi esposa, Sofy, Carlos, e Irekani mis hermosos hijos, y Carlitos (Cacho) mi nueva oportunidad de continuar presente en esta vida, gracias a todos.

A mis Padres, Rosalino (Pelayo) y Ana María quienes tuvieron y han tenido la voluntad de estar conmigo en todo momento, gracias.

Agradecimientos

De manera especial quiero agradecer a quien se da la oportunidad de enseñar, orientar y guiar a otras personas para el logro de sus metas y objetivos. En particular al Dr. Joselito Medina Marín y al Dr. Juan Carlos Seck-Tuoh Mora, quienes me acompañaron durante el doctorado, asesorando y siendo parte de los logros de la presente investigación, gracias Doctores.

En mis años como académico, agradezco a quien directa o indirectamente, pero con honestidad me llevaron a tomar la decisión de aventurarme en los estudios de doctorado. Médico Internista Adrian Valdez Espinosa, Doctor Miguel Ángel Miguez Escorcía, Doctor Asdrúbal López Chau y Doctor Ernesto Flores García, gracias a todos.

A mis compañeros de la Escuela Superior de Tizayuca por sus consejos, opiniones y confianza, gracias a todos.

A mis hermanos, y a los que la vida me regalo como hermanos, en especial a mis compadres Gil y Héctor, entre muchos otros, gracias a todos.

Índice general

Agradecimientos	I
Lista de Figuras	VI
Lista de Tablas	X
Glosario	XI
Resumen	XIII
Abstract	XV
Introducción	1
1. Estándar IEC-61131 Controladores Programables	17
1.1. Sistemas basados en PLC	18
1.2. Controlador Lógico Programable	19
1.2.1. Ciclo de trabajo del PLC	20
1.3. Lenguaje de programación diagrama escalera	21
1.4. Análisis de líneas de control en los algoritmos en LD	25
1.5. Conclusiones del capítulo	30
2. Redes de Petri	32
2.1. Definición formal de red de Petri (PN)	32
2.2. Reglas de disparo de las PN	34
2.3. Representación de comportamientos de DES con PN	36
2.4. Propiedades de la PN	40
2.4.1. Árbol de alcanzabilidad	40
2.4.2. Matriz de incidencia de PN	41

2.4.3. Ecuación de estado de PN	43
2.5. Conclusiones del capítulo	44
3. Modelado de algoritmos de control desarrollados en diagrama escalera con redes de Petri	46
3.1. Distribución de señales de LD en términos de PN	47
3.2. Problema de acumulación de marcas	51
3.3. Problema de restablecimiento de lugares	53
3.4. Estructuras en PN de líneas de control en LD	54
3.5. Definición de la red PNLN ordinaria	76
3.5.1. Marcado de la red LDPN	78
3.5.2. Metodología para transformar algoritmos de control en LD a LDPN	78
3.5.3. Reglas de disparo de transición de la LDPN	79
3.6. Conclusiones del capítulo	80
4. Validación de la red PNLN	82
4.1. Fallas físicas en sistemas basados en PLC	84
4.1.1. Falla de corto-circuito	84
4.1.2. Falla de circuito-abierto	86
4.2. Matriz de incidencia de la PNLN	87
4.3. Validación de la red LDPN	89
4.4. Simulación de la red LDPN	94
4.5. Conclusiones del capítulo	99
5. Casos de estudio	100
5.1. Sistema de control de motor reversible	100
5.1.1. Algoritmo del sistema de control de motor reversible	101
5.1.2. Red LDPN del sistema de control de motor reversible	101
5.1.3. Simulación de la red LDPN del sistema de control de motor reversible	103
5.1.4. Validación de la red LDPN del sistema de control de motor reversible	106
5.2. Sistema de control de auto-lavado de vehículos	107
5.2.1. Algoritmo de control del sistema de auto-lavado de vehículos	109
5.2.2. Red LDPN del sistema de control de auto-lavado de vehículos	111
5.2.3. Simulación de la red LDPN del sistema de control de auto-lavado de vehículos	114

5.2.4. Validación de la red LDPN del sistema de control de auto-lavado de vehículos	132
5.3. Sistema de envasado	133
5.3.1. Algoritmo de control de sistema de envasado	137
5.3.2. Red LDPN del sistema de envasado	141
5.3.3. Simulación de la red LDPN del sistema de envasado	144
5.3.4. Validación de la red LDPN del sistema de envasado	154
5.4. Conclusiones del capítulo	155
Trabajos a futuro	159
Bibliografía	161

Índice de figuras

1.1.	Diagrama a bloques de elementos involucrados con el algoritmo de control.	19
1.2.	Ciclo de trabajo con respecto al algoritmo de control en LD.	20
1.3.	Equivalencia entre un relé electromecánico y el correspondiente código en LD.	22
1.4.	Comportamiento de contactos NO y NC.	24
1.5.	Comportamiento de una bobina estándar y una negada.	24
1.6.	Contactos en serie.	25
1.7.	Contactos en paralelo.	26
1.8.	Contactos serie - paralelo.	26
1.9.	Lógica de bobinas de retención <i>Set-Reset</i>	27
1.10.	Contactos para bloqueo de línea de control.	28
1.11.	Auto-lazo.	28
1.12.	Comportamiento de ejecución secuencial en LD.	29
1.13.	Comportamiento de decisión o conflicto en LD.	29
1.14.	Comportamiento de concurrencia en LD.	30
2.1.	Representación gráfica de los conjuntos de lugares y transiciones.	34
2.2.	Ejemplo de disparo de una transición en PN.	35
2.3.	Transición fuente y sumidero.	36
2.4.	Comportamiento de ejecución secuencial en PN.	36
2.5.	Comportamiento de decisión o conflicto en PN.	37
2.6.	Comportamiento de concurrencia o paralelismo en PN.	38
2.7.	Comportamiento de sincronización en PN.	38
2.8.	Comportamiento de agrupación en PN.	39
2.9.	Comportamiento de inhibición en PN.	39
2.10.	PN de auto-lazo.	40
2.11.	Transformación de la PN de auto-lazo a lazo.	40
2.12.	PN y su árbol de alcanzabilidad.	42

2.13. PN y su matriz de incidencia.	43
2.14. PN y su ecuación de estado.	44
3.1. Señales en LD	48
3.2. Distribución de señales de LD en elementos de PN.	49
3.3. Estructura PN de la lógica AND.	55
3.4. Marcado en estructura PN de lógica AND	55
3.5. Estructura PN de lógica AND con restablecimiento.	56
3.6. Comportamiento dinámico de la estructura PN de la lógica AND.	57
3.7. Estructura PN de la lógica OR.	57
3.8. Marcado de la estructura PN de la lógica OR.	58
3.9. Estructura PN de la lógica OR con restablecimiento.	58
3.10. Comportamiento dinámico de la estructura PN de la lógica OR.	59
3.11. Estructura PN de ambas lógicas AND and OR.	60
3.12. Estructura PN de la lógica AND y OR con lugar de restablecimiento.	61
3.13. Comportamiento dinámico de la estructura PN de la lógica AND y OR.	62
3.14. Estructura PN de la lógica Set - Reset.	63
3.15. Comportamiento dinámico de la estructura PN de la lógica Set - Reset.	64
3.16. Estructura PN de la lógica de bloqueo.	65
3.17. Estructura PN de la lógica de bloqueo con lugares de restablecimiento.	66
3.18. Comportamiento dinámico de la estructura PN de la lógica de bloqueo.	67
3.19. Comportamiento dinámico de la estructura PN de la lógica de bloqueo.	67
3.20. Estructura PN del comportamiento de ejecución secuencial.	68
3.21. Estructura PN del comportamiento de ejecución secuencial con restablecimiento.	68
3.22. Estructura PN del comportamiento de ejecución secuencial generalizada.	69
3.23. Comportamiento dinámico de la estructura PN del comportamiento de ejecución secuencial.	69
3.24. Estructura PN para el comportamiento de decisión.	70
3.25. Estructura PN para el comportamiento de decisión con lugar de restablecimiento.	71
3.26. Estructura PN generalizada para el comportamiento de decisión.	71
3.27. Comportamiento dinámico de la estructura PN del comportamiento de decisión.	72
3.28. Estructura PN del comportamiento de concurrencia.	73
3.29. Estructura PN del comportamiento de auto-lazo.	74
3.30. Comportamiento dinámico de la estructura PN del comportamiento de auto-lazo.	75

4.1. Contexto para el diseño de algoritmos de control en LD.	83
4.2. Falla de corto-circuito, a) en el sub-sistema de entradas del PLC, y b) en la red LDPN.	85
4.3. Falla de corto-circuito en el sub-sistema del PLC, a) en el cableado, b) en el módulo de salidas y c) en la red LDPN.	85
4.4. Falla de circuito-abierto, a) en el sub-sistema del PLC, y b) en la red LDPN.	86
4.5. Red LDPN ejemplo.	88
4.6. Diagrama de flujo del enfoque de validación de la red LDPN.	93
4.7. Lugares I_2^c con marca, LDPN arranque y paro de un motor.	95
4.8. Marca en el lugar I_1 , equivalente al accionamiento del botón de arranque.	96
4.9. Disparo de la transición \mathbf{I}_1^o	96
4.10. Disparo de la transición \mathbf{L}_1	97
4.11. Disparo de la transición \mathbf{O}_1^o	97
4.12. Disparo de la transición \mathbf{L}_2	97
4.13. Actualización del marcado y disparo de la transición \mathbf{I}_2^c	98
4.14. Inhabilitado de la transición \mathbf{I}_2^c , marca en lugar de restablecimiento $G(\mathbf{L}_2)_{AND}$	98
5.1. Algoritmo de control en LD de motor reversible.	102
5.2. LDPN de sistema de control de motor reversible.	102
5.3. LDPN de sistema de control de motor reversible, a) y b).	103
5.4. LDPN de sistema de control de motor reversible, c) y d).	104
5.5. LDPN de sistema de control de motor reversible, e) y f).	105
5.6. Sistema de control de auto-lavado de vehículos.	107
5.7. Algoritmo de control de sistema de auto-lavado de vehículos, parte 1.	109
5.8. Algoritmo de control de sistema de auto-lavado de vehículos, parte 2.	110
5.9. LDPN del sistema de control de auto-lavado de vehículos.	112
5.10. Matriz de incidencia de la LDPN del auto-lavado de vehículos.	113
5.11. LDPN del sistema de control de auto-lavado de vehículos, marca en los lugares I_i^c y B_b^c	117
5.12. LDPN del sistema de control de auto-lavado de vehículos, M_0	118
5.13. LDPN del sistema de control de auto-lavado de vehículos, disparo de transición \mathbf{B}_0^o	119
5.14. LDPN del sistema de control de auto-lavado de vehículos, primer ciclo de trabajo del PLC.	120
5.15. LDPN del sistema de control de auto-lavado de vehículos, actualización del marcado, S1=1.	121

5.16. LDPN del sistema de control de auto-lavado de vehículos, segundo ciclo de trabajo del PLC.	122
5.17. LDPN del sistema de control de auto-lavado de vehículos, actualización del marcado, S2=1.	123
5.18. LDPN del sistema de control de auto-lavado de vehículos, tercer ciclo de trabajo del PLC.	124
5.19. LDPN del sistema de control de auto-lavado de vehículos, actualización del marcado, segunda vez de S1=1.	125
5.20. LDPN del sistema de control de auto-lavado de vehículos, cuarto ciclo de trabajo del PLC.	126
5.21. LDPN del sistema de control de auto-lavado de vehículos, actualización del marcado, segunda vez de S2=1.	127
5.22. LDPN del sistema de control de auto-lavado de vehículos, marca en lugares de restablecimiento.	128
5.23. LDPN del sistema de control de auto-lavado de vehículos, accionamiento de $P = 1$ en LD, $B_5 = 1$ en LDPN.	130
5.24. LDPN del sistema de control de auto-lavado de vehículos, $O_1 = 1$ en condiciones de paro.	131
5.25. Sistema de envasado.	134
5.26. Algoritmo de control en LD del sistema de envasado, parte uno.	138
5.27. Algoritmo de control en LD del sistema de envasado, parte dos.	139
5.28. Algoritmo de control en LD del sistema de envasado, parte tres.	140
5.29. Algoritmo de control en LD del sistema de envasado, parte cuatro.	141
5.30. LDPN de báscula uno	142
5.31. LDPN de báscula dos	143
5.32. LDPN de báscula uno, marcado en lugares I_i^c , B_b^c y O_o^c	145
5.33. LDPN de báscula uno, marcado en condiciones de funcionamiento.	148
5.34. LDPN de báscula uno, segundo ciclo de trabajo del PLC.	149
5.35. LDPN de báscula uno, tercer ciclo de trabajo del PLC.	150
5.36. LDPN de báscula uno, cuarto ciclo de trabajo del PLC.	151
5.37. LDPN de báscula uno, descargando.	152
5.38. LDPN de báscula uno, S_CHAP_B1=1.	153

Índice de tablas

2.1. Definición formal de una PN	33
2.2. Símbolos gráficos utilizados en una PN	34
5.1. Variables físicas de sistema de control de motor reversible.	101
5.2. Variables físicas de sistema de control de auto-lavado de vehículos.	108
5.3. Variables de señales físicas de entrada y salida del sistema de envasado.	136
5.4. Variables de señales de memoria del sistema de envasado.	137

Glosario

Acrónimo	Definición (Español)	Definition (Inglés)
PLC	Controlador Lógico Programable	Programmable Logic Controller
IEC	Comisión Electrotécnica Internacional	International Electrotechnical Commission
ST	Texto Estructurado	Structured Text
IL	Lista de Instrucciones	Instruction List
FBD	Diagrama de Bloques de Funciones	Function Block Diagram
SFC	Diagrama de Funciones Secuenciales	Sequential Function Chart
LD	Diagrama Escalera	Ladder Diagram
DECS	Sistemas de Control de Eventos Discretos	Discrete Events Control Systems
DES	Sistemas de Eventos Discretos	Discrete Events Systems
PN	Red de Petri	Petri Net
LDPN	Red de Petri de Diagrama Escalera	Ladder Diagram Petri Net
RTPN	Red de Petri de Tiempo Real	Real-Time Petri Net
APN	Red de Petri de Automatización	Automation Petri Net
IPN	Red de Petri Interpretada	Interpreted Petri Net
SIPN	Red de Petri de Señal Interpretada	Signal Interpreted Petri Net
RTSPN	Red de Petri de Etapa de Tiempo Real	Real-Time Stage Petri Net
V & V	Validación y Verificación	Validation & Verification
CPN	Red de Petri de Control	Control Petri Net
SPNC	Controlador de Red de Petri Simplificado	Simplified Petri Net Controller
TPL	Lógica Pasa Marcas	Tokens Pass Logic
RAC	Componentes de Automatización Re-utilizables	Reusable Automation Components
UML	Lenguaje de Modelado Unificado	Unified Modeling Language
SMV	Verificador de Modelo Simbólico	Symbolic Model Checker
OPC	Plataforma de Comunicación Abierta	Open Platform Communications

Glosario

Acrónimo	Definición (Español)	Definition (Inglés)
FDS	Sistema de Diagnóstico de Falla	Fault Diagnosis System
DESC	Controlador de Sistema de Eventos Discretos	Discrete Events Systems Controller
LDPN	Red de Petri Diagrama Escalera	Ladder Diagram Petri Net

Validación de Algoritmos de Control en Diagrama Escalera para Detección de Condiciones de Riesgo con Redes de Petri.

Resumen

Los sistemas basados en Controladores Lógicos Programables (PLC, por sus siglas en inglés) se caracterizan por su trabajo cíclico, leer el estado de las señales de entrada física que vienen de los sensores de los procesos, realización de la copia de los estados de dichas señales, con los que se evalúa el algoritmo de control, el resultado de la evaluación es copiado para enviar los estados de las señales en un mismo momento a los módulos de las salidas físicas, los cuales están conectados a los actuadores del proceso.

El desarrollo de los algoritmos de control de procesos industriales se basan en las especificaciones formales y de las interpretaciones y experiencia de los programadores responsables del control, los cuales no utilizan teorías formales para su análisis y comprobación antes de ser implementado en los procesos industriales.

Las redes de Petri son una herramienta gráfica y matemática para analizar, modelar y simular sistemas de eventos discretos como los algoritmos de control desarrollados en el lenguaje gráfico Diagrama Escalera, el cual está contemplado en el estándar IEC 61131 donde se especifica su sintaxis y semántica respectiva.

El presente trabajo propone una metodología de diseño formal para transformar algoritmos de control desarrollados en Diagrama Escalera a redes de Petri, denominada LDPN (por sus siglas en inglés) que considera estructuras lógicas y comportamientos característicos de los sistemas de eventos discretos como es la concurrencia, el bloqueo, la ejecución secuencial, entre otro. Además, considera el problema de acumulación de marcas en la red a través de ecuaciones lógicas y algebraicas para el comportamiento dinámico equivalente al ciclo de trabajo de los sistemas basados en PLC, así como el comportamiento de energizar o des-energizar bobinas, como lo es en los procesos industriales.

La matriz de incidencia y la ecuación de estado de las redes de Petri es la base para validar la red LDPN propuesta en condiciones de falla de corto-circuito o circuitoabierto en el sub-sistema de entradas físicas y apoyar a los responsables del control a tomar la mejor decisión, para evitar la combinación de fallas en el algoritmo de control.

Tres algoritmos de control en Diagrama Escalera son analizados, modelados, simulados y validados con el enfoque propuesto demostrando la utilidad para el desarrollo de control seguro en los procesos industriales o de manufactura.

Validation of Control Algorithms in Ladder Diagram to Detect Risk Conditions with Petri Nets.

Abstract

The PLC-based systems are characterized by their cyclic work, reading the state of the physical inputs signals from sensors in the process, copy of the states of those signals with which the control algorithm is evaluated. The evaluation results are copied to send at same time to the modules of the physical outputs signals, which are connected to actuators in the process.

The Development of control algorithms of process industrial are based on the formal specifications, in the interpretations and experience of the programmer responsible of control, who they do not use formal theories for analysis, and testing before being implemented in the industrial process.

The Petri nets are graphical and mathematical tools to analyze, model and simulate discrete events systems such as control algorithms developed in the graphic language Ladder Diagram, which is contemplated by the IEC 61131 standard where is specified the syntax and semantics respective.

This work proposes a formal design methodology for transforming control algorithms in Ladder Diagram to Petri nets, called LDPN, which's considers logical structures and behaviors characteristic of systems discrete events such as the concurrent, blocking, sequential execution, among others. Also, the LDPN included the problems of accumulation of marks on the network through of logical and algebraic equations for the dynamic behavior equivalent to the cycle work of the PLC based systems, as well as, behavior of energizing or deenergizing coils in industrial processes.

The incidence matrix and equation of state of the Petri nets is the base to validate the LDPN proposed under fault conditions short-circuit or open-circuit in the subsystem of physical inputs, and support those responsables of the control to make the better decision, to avoid the combination of failure in the control algorithm.

Three control algorithms in Ladder Diagram are analyzed, modeled, simulated and validated with the proposed approach demonstrating the utility for developing secure control in industrial or manufacturing processes.

Introducción

Actualmente, el control seguro es uno de los más importantes aspectos estudiados por los investigadores internacionales, en el campo de diseño y desarrollo de sistemas de producción automatizada, que evite accidentes y se incremente la producción, por ende, la competitividad de las empresas. De ahí, la importancia de simular, modelar y validar los algoritmos de control con metodologías formales para reducir o evitar errores en los sistemas basados en Controladores Lógicos Programables (PLC, por sus siglas en inglés) [1].

Para estandarizar las definiciones y características de la aplicación de controladores en aplicaciones y ambientes industriales, la Comisión Electrotécnica Internacional (IEC, por sus siglas en inglés) definió el estándar IEC 61131, el cual se compone de ocho apartados. En específico, el estándar IEC 61131-3 muestra la sintaxis y semántica de cinco lenguajes para la programación de PLC [2]; Texto Estructurado (ST, por sus siglas en inglés) y Lista de Instrucciones (IL, por sus siglas en inglés) son considerados lenguajes textuales; Diagrama de Bloques de Funciones (FBD, por sus siglas en inglés), Diagrama de Funciones Secuenciales (SFC, por sus siglas en inglés) y Diagrama Escalera (LD, por sus siglas en inglés) son considerados lenguajes de programación gráficos.

Los sistemas basados en PLC continúan siendo utilizados en la industria para el control de procesos discretos o Sistemas de Control de Eventos Discretos (DECS, por sus siglas en inglés) y dominan la industria moderna automatizada debido a su robustez y relativo bajo costo [3]. Sin embargo, el problema de los algoritmos de control para DES, específicamente para sistemas basados en PLC, continúan desarrollados de forma heurística y con base en la experiencia de los responsables del control en la industria, y no se basan en teorías formales para la simulación y/o validación del algoritmo antes de ser implementados [4]. Además, se considera que los enfoques tradicionales para mejorar la fiabilidad de las funciones para diagnosticar fallas en los sistemas basados en PLC no son desarrollados con base en las posibles fallas de las señales, y los desarrolladores

y/o programadores de los algoritmos de control no son capaces de identificar todas las posibles fallas que pueden ocurrir durante la vida del sistema [5].

Los algoritmos de control desarrollados en el lenguaje LD, tiene como base de funcionamiento el comportamiento de relés electromecánicos funcionando en un mismo tiempo, además, por su semejanza con los diagramas eléctricos continua siendo uno de los más utilizados para el control de DECS. Sin embargo, el problema persiste, estos algoritmos de control son desarrollados por los usuarios en la industria y no emplean teorías formales para la simulación y/o validación de éstos, y por ende, no garantizan su calidad antes de ser implementados en los sistemas basados en PLC para el control de las máquinas o procesos de producción, es decir, los algoritmos de control se continúan desarrollando con base en la experiencia de los programadores y/o ingenieros responsables de la operación, mantenimiento y actualización de los sistemas de control para procesos industriales [6], [7].

Por otro lado, las Redes de Petri (PN, por sus siglas en inglés) son una herramienta para modelar de forma gráfica y/o matemáticamente los DECS, los cuales pueden tener comportamientos de concurrencia, asíncronos, distribuidos, paralelos, estocásticos, entre otros. La PN de forma general tiene lugares de entrada, transiciones y lugares de salida, los cuales pueden ser interpretados como señales de entrada física, algoritmo de control y señales de salida física, respectivamente, en los componentes de los sistemas basados en PLC.

El problema de modelar y/o validar algoritmos de control en LD ha sido estudiado con la teoría de PN, lo que ha generado la propuesta de diversos enfoques que tratan la transformación, comparación, modelado y/o validación de LD a PN o viceversa. Sin embargo, hasta nuestro conocimiento, los enfoques propuestos no consideran el comportamiento elemental de los sistemas basados en PLC, que es su operación cíclica, además, de la naturaleza del proceso, la necesidad de energizar y/o des-energizar bobinas, así como una validación sustentada en fallas reales de corto-circuito y/o circuito-abierto en el sub-sistema de entradas físicas. Estas situaciones son analizadas en el presente trabajo, el cual propone una extensión de la PN para transformar, modelar y validar algoritmos de control desarrollados en LD, denominada *LDPN* (Ladder Diagram Petri Nets, por sus siglas en inglés).

Estado del arte

De forma cronológica, se muestran aspectos relevantes de los enfoques o metodologías propuestas para el problema de modelar, simular y validar algoritmos de control en LD utilizando PN.

Il Moon [8], propone un método de verificación para determinar la seguridad y operatividad de los sistemas basados en PLC. El modelo está basado en expresiones booleanas que representan el comportamiento del algoritmo de control, entonces, realizan proposiciones en lógica temporal para su aseveración, y en el caso de encontrar errores, proporcionan contra-ejemplos, los cuales son estructuras básicas utilizadas en los LD. Modela el comportamiento de los contactos Normalmente Abiertos (NO, por sus siglas en inglés) y Normalmente Cerrados (NC, por sus siglas en inglés) y bobina con estructuras *if - then*.

La comparación entre el diseño de lógicas AND, OR, concurrencia, temporizadores y sincronización en constructores *if - then* y su estructura en PN con líneas en LD es propuesta por Kurapati, M. Zhou y Reggie [9] y S. Peng y M. Zhou [10]. Los autores introducen una modificación de PN llamada *Red de Petri de Tiempo Real* (RTPN, por sus siglas en inglés) de tipo ordinaria. La RTPN adiciona un vector de tiempos para modelar acciones temporizadas, un vector de señales de entrada y uno de salida para asociar atributos entre las señales y su correspondiente lugar p_i , además, dos tipos de transiciones “disparo-inicio” y “disparo-final” para remover marcas de los lugares de entrada y depositar marcas en los lugares de salida respectivamente.

A. Taholakian y W. M. M. Hales [11] proponen una metodología para generar algoritmos de control “correctos” directamente de las especificaciones formales utilizando PN. Los autores definen el grafo PN considerando las entradas de lado izquierdo y las salidas de lado derecho de dos líneas verticales, entonces para colocar una marca en un lugar de salida se debe trabajar hacia atrás y determinar las condiciones a través de saber que transiciones-habilitadas deberían existir. Los autores definen “arco-retorno” donde el lugar de entrada puede recibir su marca después que la transición fue disparada, “arco-conmutación apagado” para restablecer un lugar, duplicación de lugares para utilizar la señal en más de una transición y un “lugar dren” para consumir la marca de un lugar cuando una “transición-conmutación apagado” es disparada.

Una extensión de PN implementada para generar algoritmos de control en LD considerando la norma IEC-1131-3 es presentada por M. Uzam y A. H. Jones [12], deno-

minada PN de automatización (APN, por sus siglas en inglés). Su propuesta de red considera 9-tuplas, donde se definen funciones de entrada y salida que a la vez definen arcos ordinarios directos de lugares a transiciones y de transiciones a lugares respectivamente, además, funciones de entrada para inhabilitar o habilitar arcos que van de lugares a transiciones. Aunado a lo anterior, los autores proponen dos conjuntos para las condiciones de disparo de las transiciones y para las acciones que podrían ser asignadas a los lugares, es de resaltar, que estos dos conjuntos están en función de las especificaciones formales.

El enfoque de PN Interpretadas (IPN, por sus sigla en inglés) para verificación y validación de algoritmos de control fue presentado por G. Frey y L. Litz [13]. Resalta la consideración de formalizar las especificaciones informales al implementar un algoritmo de control en un proceso industrial y/o de manufactura. La IPN es una red ordinaria y libre de lazos, se compone de 8-tuplas, donde se definen conjuntos finitos de señales de entrada y salida binarias, además, un mapa que asocia cada transición con una condición de disparo externa y un mapa que asocia a cada lugar con una acción, la cual es activada mientras el lugar de salida tenga marca. Posteriormente, los autores presentaron una extensión de la IPN, llamada PN de señal interpretada (SIPN, por sus siglas en inglés) [14], la cual agrega una función de salida que combina la salida de todos los lugares con marca en la red, pudiendo ser esta indefinida, uno, cero, contradictoria, cero o uno redundante, o una combinación de contradicciones y redundancia. Esta función es re-calculada en cada marcado estable de la red.

Una propuesta para diseñar controladores lógicos usando PN y su implementación en algoritmos de control en LD es presentada por S. G. Tzafestas, M. G. Pantelidis y D. L. Kostis [15]. La propuesta incluye auto-lazos, y considera la verificación de la red en conceptos teóricos como el nivel de seguridad, vivacidad, alcanzabilidad y consistencia. Al cumplir con estos requisitos la red, se puede generar el algoritmo de control en LD, mapeando el modelo en PN, donde a cada lugar le corresponde un bit de la memoria del PLC y para el ciclo de trabajo del PLC se definen las sub-rutinas *MAPIN* y *MAPOUT* para cada sensor y actuador de los sistemas.

Otro método propuesto para diseñar controladores para DES, e implementar en sistemas basados en PLC a través de algoritmos de control en LD fue presentado por G. Music, D. Matko [16]. Los autores usan la teoría de control supervisorio para generar un lenguaje formal llamado *Autómata* compuesto de 5-tuplas; conjunto de estados, conjunto de símbolos asociados con eventos, función de transición de estados, marcado inicial y sub-marcados. La propuesta sólo considera autómatas simples, es decir, cada

especificación formal de los procesos de manufactura es analizada y modelada de forma independiente.

G. Zapata y E. Carrasco [17], modelan estructuras de control utilizadas en los algoritmos en LD mediante PN. Los autores proponen las estructuras en PN de comportamientos lógicos AND, OR, Set-Reset, Flanco, Receptividad, Concurrencia y Retardo, las cuales sólo cumplen el comportamiento de energizado de la bobina correspondiente, además, el ejemplo y su PN propuesta es totalmente secuencial.

Una nueva extensión de la RTPN (S. Peng y M. Zhou [18]) llamada PN de etapas de tiempo real (Real-time stage Petri Net, RTSPN por sus siglas en inglés). La red propuesta se compone de una RTPN y de una red de control de etapas. La RTSPN agrega dos vectores; uno para escanear la combinación lógica de las señales de entrada y salida de la RTPN para cada etapa y otro para disparar una marca a todos los lugares en la red de control y habilitar transiciones de la RTPN. Los autores en su propuesta consideran el principio de los PLC, que es el ciclo de trabajo como una actividad secuencial.

G. Frey propone una métrica para la calidad de software utilizados en los sistemas basados en PLC [19], en los que resalta el concepto de *Funcionalidad* como la exactitud del algoritmo de control proveído por los enfoques de validación y verificación y el de *Fiabilidad* como la exactitud y robustez del algoritmo de control sobre mínimos supuestos acerca de entornos del controlador que incluya posibles errores. Es de resaltar, la propuesta de “incluir” las especificaciones no formales en la evaluación, aclarando que un mejor entendimiento del proceso por los responsables del control, permite resolver estos inconvenientes. Posteriormente M. B. Younis y G. Frey consideran que los dos enfoques para formalizar se basan en ingeniería inversa y métodos de validación y verificación (V & V) [20], de estos últimos sobresale la aplicación de la teoría de PN.

Una extensión de PN denominada PN de Control (CPN, por sus siglas en inglés) fue presentada por G. B. Lee, H. Zandong y J. S. Lee [21] para generar automáticamente algoritmos de control en LD. Su enfoque se basa sobre la regulación de los disparos de las transiciones, la relación entre los lugares, condiciones y eventos son formulados en funciones Booleanas.

La propuesta de estructuras de unión y distribución AND y OR en PN para convertir a líneas de control en LD fue presentado por T. Suesut, P. Inban, P. Nilas, P. Rerngreun y S. Gulphanich [22].

Peng y Zhou [23], hacen referencia a constructores de tipo gráfico para las lógicas AND y OR mostrando su equivalencia entre los métodos LD y PN.

Equivalencias entre lógica AND y OR, así como la distribución concurrente es propuesta por S. Lee and Hsu [24], los cuales utilizan reglas *if-then* para ir construyendo las consignas de control.

El enfoque denominado Controlador de PN simplificado (SPNC, por sus siglas en inglés) es presentado por J. S. Lee y P. L. Hsu [25] para diseñar e implementar controladores secuenciales en sistemas de manufactura. La red propuesta se constituye de 5-tuplas, una de las cuales introduce el concepto de “estado del sensor” que se convierte en una señal más para el disparo de una transición. Las reglas de disparo se basan en la metodología de Lógica de Pasar Marcas (TPL, por sus siglas en inglés), la cual es una red de 6-tuplas donde se consideran los conjuntos para bits de memoria, entradas de sensores y salidas a actuadores.

La propuesta de transformar modelos de PN a algoritmos de control en LD de sistemas basados en PLC a través de la técnica de mapeo uno-a-uno es presentada por D. Thapa, S. Dangol y G. Wang [26]. Los autores para la transformación consideran las especificaciones informales, las cuales suponen se pueden modelar con PN, análisis de las propiedades, evaluar si existen errores, corregir y convertir a LD. La evaluación la proponen en función de características de las PN; Alcanzabilidad, vivacidad, seguridad, reversibilidad y marcado inicial.

G. Frey y F. Wagner presentan una caja de herramientas basadas en la red SIPN [27]. Los autores consideran que los pasos de un desarrollo formal de algoritmos de control para sistemas basados en PLC son: diseño, evaluación, verificación, validación, pruebas e implementación. Para la validación y verificación proponen un modelo cheador simbólico basado en diagramas de decisión booleanas.

El desarrollo de un simulador virtual para validación de algoritmos de control utilizados en los sistemas basados en PLC visualmente fue presentado por C. M. Park, S. M. Bajimaya, S. C. Park, G. N. Wang, J. G. Kwak, K. H. Han y M. Chang [28]. Los autores hacen una replica del proceso y aplican un modelo para simular el comportamiento, el cual esta constituido por 7-tuplas; conjunto de entradas, salidas, estados binarios, función de transiciones externas, función de transiciones internas y función avance de tiempo.

El enfoque de verificación y validación fuera de línea (offline) de algoritmos de control es presentado por D. Thapa, C. M. Park, S. Dangol y G. N. Wang [29]. La propuesta se fundamenta en tres fases para verificación y validación, el cual considera pruebas en manual, modelo chegador y puesta en servicio virtual para las especificaciones del sistema. Los autores consideran que después de cumplir con estas pruebas, el código puede ser implementado en un sistema basado en PLC. Posteriormente los autores presentan la propuesta de realizar un ambiente que habilite la verificación visual de los algoritmos de control en un ambiente gráfico a 3D del sistema a controlar que tiene como base un mapeo del estado de las entradas y salidas físicas del sistema basado en PLC [30].

Una metodología basada en PN para el diseño de control lógico de sistemas de manufactura flexibles y modelar la secuencia concebida como un proceso incremental es presentado por A. Castelnovo, L. Ferrarini y L. Piroddi [31]. El enfoque propuesto por los autores, es una red PN que va adicionando sub-redes a un modelo parcial hasta cumplir con todas las especificaciones formales e informales del proceso.

Una estructura de trabajo para componentes de automatización re-utilizables (RAC, por sus siglas en inglés) el cual puede ser usado con el estándar IEC 61131 es presentado por O. Ljungkrantz, K. Åkesson y M. Fabian [32]. Los RAC consideran la implementación y especificaciones del comportamiento de cada componente, el cual puede ser representado en LD con un argumento basado en lógica temporal utilizado para describir el orden en tiempo en el cual los estados de los componentes cambian. Las especificaciones para la estructura de verificación considera cinco componentes; pre-condiciones de operación, comportamiento de operación, condiciones de excepción, comportamientos de excepción e invariantes.

Lee and S. Lee [6], proponen la técnica de síntesis de módulos para la conversión de LD a PN, considerando los núcleos básicos del lenguaje LD definidos en la norma IEC-61131-3. Para modelar el estado de contactos NO y NC de una señal, proponen un lugar unido a dos transiciones a través de un arco habilitador y un arco in-habilitador. Los autores presentan un ejemplo de un algoritmo de control con funciones lógicas AND y OR. Sin embargo, todas las variables en el algoritmo de control son independientes, es decir, no existen contacto(s) de alguna de las bobinas como retroalimentación en las líneas de control del LD.

La propuesta de un método para construir un modelo abstracto para analizar y diagnosticar algoritmos de control en LD usando PN es presentado por J. I. Tsai and C. C. Teng [33]. Los autores definen una PN Booleana (BPN, por sus siglas en inglés) donde

los lugares representan el estado de la acción y se asigna a las bobinas y a las salidas de los bloques de control, mientras que los contactos NO y/o NC de las señales de entrada física son representadas por las transiciones, las cuales tiene una expresión booleana.

De igual forma, Korotkin, Zaidner, Coben, Ellenbogen, Arad y Cohen [2], proponen una metodología para implementar modelos basados en PN para DECS, los autores definen la red binaria *PNDEC* y sus ecuaciones para el marcado dinámico. La evaluación propuesta es solo en un barrido del PLC, y el algoritmo de control está basado en condiciones booleanas asignadas a los lugares de salida. El marcado dinámico es generalizado, sin embargo, este no considera los estados iniciales de bobinas energizadas.

Un método de transformación y verificación es presentado por I. Grobelna, M. Grobelny y M. Adamski [34], tiene como base las especificaciones del usuario, primero lo implementan en lenguaje UML (Unified Modeling Language) y posteriormente lo transforman en PN para su validación. El proceso de verificación se realiza a través de una herramienta NuSMV (Symbolic Model Checker), donde el modelo “checador” es con base en las especificaciones y propiedades del sistema, las cuales se definen en lógica temporal.

Un procedimiento general para detectar fallas en sistemas basados en PLC es presentado por A. H. Malik, Tahir Mehmood, Muhammad Ahmad Choudhry y Aamir Hanif, Member [35]. Los autores consideran problemas de hardware y software para determinar una falla genérica, apoyándose de los indicadores luminosos en los módulos que integran el sistema basado en PLC. Importante rescatar, la mención de que un mejor entendimiento del sistema permite una solución efectiva y eficaz de las fallas.

V. Năvrăpescu, M. Popescu, I. D. Deaconu y A. I. Chirilă [36], proponen una metodología basada en tablas de verdad (mapas de Karnaugh) para representar los estados de las variables de los algoritmos de control y funciones booleanas para las transiciones. El ejemplo que muestran es el arranque y paro de un motor con una sola línea de control en LD.

Una extensión de las PN llamada Coloreadas es utilizada para obtener automáticamente modelos formales de los algoritmos de control en LD [37]. Da Silva, Gorgonio, Perkusich y Martins, consideran que cada línea de control en LD puede ser escrita en lógica proposicional. Su enfoque considera los pasos secuenciales del PLC; lectura de las señales de entradas físicas, inicio del ciclo y acceso a la memoria del PLC. Estos elementos son analizados considerando el flujo de energía de forma secuencial a través

de los renglones del LD, sin embargo, no se considera el principio de funcionamiento del barrido de los sistemas basados en PLC.

T. Alenljung, B. Lennartson y M. N. Hosseini [38] proponen una *grafo sensor* para modelar DES y verificar algoritmos de control de sistemas basados en PLC. El núcleo del sensor grafo contiene vectores para las señales de entradas y salidas, además de, un vector para el estado de las variables, un conjunto de eventos, conjunto de predicados para las transiciones, una función de salida para asignarle un valor y un conjunto para las condiciones iniciales, las invariantes y limitaciones del sistema. Los autores agregan para su propuesta un conjunto de localizaciones, uno de localizaciones abstractas, un conjunto de objetos del sistema, una función de relación sensor-localización, una función de sub-localización de las localizaciones abstractas, una función de capacidad, conjunto de uno o dos caminos con predicado y secuencia de localización, conjunto de localizaciones invariantes, y uno más de localización del modelo de limitaciones del sistema.

Un método para transformar algoritmos de control en LD a PN de tipo ordinaria es presentado por X. Chen, J. Luo y P. Qi [39]. Los autores definen un *LD grafo* de dos tuplas; un conjunto de vértices, uno para representar la unión del riel de energía izquierdo con el último renglón en el LD, otro similar para el riel de energía derecho y otro para representar tres características de las variables (tipo de señal, tipo de elemento y consecutivo en LD); y un conjunto de flancos que representan las líneas de unión entre dos elementos en LD. Para la transformación, los autores proponen una variable booleana para cada contacto o bobina, un par de lugares donde su marcado representa el valor de 0 o 1 respectivamente, para modelar el estado de un contacto o bobina. Para cada variable de entrada le corresponden dos transiciones para conectar con sus dos lugares, y modelar su estado. Para el comportamiento de la red, se definen dos conjuntos de rutas; las que energizan y las que des-energizan una bobina.

Conversión de algoritmos de control en máquinas B para su análisis formal de limitaciones de seguridad es presentado por H. Barbosa and D. Déharbe [40]. La generación de la máquina B queda en función de las especificaciones del proyecto. Las especificaciones informales o limitaciones no explícitas son integradas al refinamiento del algoritmo de control de forma “*manual*”.

En [41], R. Wang, M. Zhou, L. Yin, L. Zhang, J. Sun y G. Mingse presentan el modelado y validación de un sistema control basado en PLC usando la estructura de trabajo BIP (Behavior-Interaction-Priority). Proponen un monitor para cada una de las propiedades a validar, integran todos los monitores para generar el sistema global para

su simulación. Si un requerimiento es violado, entonces el monitor transita a estado de error. El enfoque se aplica a un sistema real donde se encuentran “errores” en el diseño global del sistema.

Un estudio de detección de fallas y diagnóstico para sistemas basados en PLC es propuesto por S. Qin y G. Wang [42]. La propuesta de la operación y estados de los sensores y actuadores son monitoreados continuamente a través de servidor que tiene una plataforma de comunicación abierta (OPC, por sus siglas en inglés) conectado al PLC. Comparan un vector de valores normales de operación de las señales con los valores observados en tiempo real, si existe discrepancia lo indican con una alarma. Una condición de falla puede coincidir con el estado correspondiente de ese momento del proceso, lo que permitiría una secuencia más en el proceso.

La propuesta de un diagnosticador con base en PN difusas de tiempo-real es presentado por Z. Wu y S. J. Hsieh [43]. La red permite la lectura de los estados de entradas y salidas del proceso, los cuales se comparan con valores predeterminados, si existe una diferencia entonces la información se trata con la PN difusa para diagnosticar y encontrar la raíz de la causa que originó la falla. Para la ecuación de estado, agregan una igualdad entre el mapeo de los posibles valores del conjunto de entradas y salidas con los marcados alcanzables desde un marcado inicial.

L. Huai y C. Cheng [44], consideran que las causas principales de falla en las señales de entrada son el corto-circuito y circuito-abierto por daño en las líneas de conexión de los sensores a los PLC o por fallas en los contactos mecánicos de los interruptores o por daño en los sensores electrónicos. Para la fiabilidad de las señales de entrada proponen que varios sensores sean con alta fiabilidad y remover las “causas” para evitar el corto-circuito, circuito-abierto o línea de conexión al PLC. La credibilidad de la señal de entrada desde el sitio de producción de PLC puede ser estimado de acuerdo a las características del sistema de control, así como la relación entre las señales.

Un algoritmo de control secuencial en LD es considerado por E. V. Kuzmin y V. A. Sokolov [45] para mostrar la “vulnerabilidad” de este. La apertura de una puerta con la secuencia de presionar cuatro botones instantáneos y un sensor del estado de la puerta (cerrada – abierta), así como botones para restablecer las condiciones del sistema. En el artículo se considera que al presionar todos los botones en un mismo momento y en un mismo barrido del PLC se abriría la puerta debido a que el algoritmo de control se ejecuta cada ciclo de izquierda a derecha y de arriba a abajo. La propuesta de presionar todos los botones en un mismo instante es el equivalente al caso extremo de falla de

corto-circuito para todos los sensores de entrada al PLC. Sin embargo, para este ejemplo, la puerta no abriría, ya que el algoritmo de control en los PLC se ejecuta cada ciclo con la copia de los estados de las señales de entrada de un mismo “momento” (etapa de lectura de señales de entrada en el ciclo del PLC). El lenguaje Candence SMV es utilizado para validar algoritmos de control desarrollados en LD. La base del modelado es la conversión del algoritmo de control en LD en lógica AND, OR y NOT.

J. C. Quezada, J. Medina, E. Flores, J. C. Seck Tuoh, y N. Hernández proponen una metodología para transformar algoritmos de control en LD a PN [46]. La extensión de PN de tipo ordinario, denominada Red de Petri de Diagrama Escalera (*LDPN*). La propuesta contempla la distribución de las señales físicas y de memoria de los sistemas basados en PLC y sus contactos NO y/o NC, modela el comportamiento de energizar o des-energizar bobinas, y presenta las estructuras en PN de las lógicas AND, OR, AND-OR, Set-Reset y Auto-lazo.

J. Bao, H. Wu y Y. Yan [47] proponen un sistema de diagnóstico para mejorar la fiabilidad de los sistemas basados en PLC. Consideran que los desarrolladores del sistema y programadores no son capaces de identificar cada falla que pueda ocurrir en el propio sistema. Su enfoque FDS-PLC (Sistema diagnóstico de falla - PLC), se ejecuta en “paralelo” con el sistema de control en el PLC y el sistema de diagnóstico que tiene como base una máquina de estado finita, se ejecuta en una computadora personal conectada al PLC. El enfoque de diagnóstico propone un estado inicial del sistema con base en las especificaciones, se copia el estado de las señales de entrada, se realiza la comparación de la entrada copiada con la del estado inicial, si no corresponde a las especificaciones se reporta la “falla o situación desconocida”, de lo contrario es actualizado el estado del sistema, continúa con la lectura de las señales de entrada y con la comparación de su estado de forma periódica.

En general y hasta nuestro conocimiento los enfoques propuestos no consideran los siguientes aspectos para el modelado dinámico de una algoritmo de control en LD de los sistemas basados en PLC, y que son considerados en la presente investigación como aportaciones y originalidad del trabajo:

- Comportamiento de energizar y des-energizar bobinas.
- Condición inicial de una bobina sea energizada.

- Contactos de bobinas en otras líneas de control como retroalimentación o funcionamiento secuencial del algoritmo.
- Contacto de la misma bobina en una línea de control como retroalimentación de auto-lazo.
- No se considera la operación cíclica de los sistemas basados en PLC.
- La condición de disparo de una transición de algunos enfoques propuestos no consideran la acumulación de marcas en los lugares de salida.
- La verificación de los enfoques propuestos de transformación de LD a PN consideran aspectos teóricos, y no fallas reales que se presentan en los sistemas basados en PLC.

Con base a lo anterior, la presente investigación propone en un inicio modelar algoritmos de control desarrollados en LD con una extensión de PN, denominada *LDPN*, que constituye la base para validar los algoritmos de control en función de las fallas reales de corto-circuito y/o circuito-abierto que puedan presentarse en el sub-sistema de entradas físicas de los sistemas basados en PLC, y por ende, apoyar a los responsables o programadores del control a reducir los riesgos en los procesos industriales.

Justificación

La necesidad de contar con medios de validación de los algoritmos de control desarrollados por los usuarios e implementados en los PLC para el control de máquinas, protecciones de procesos continuos, sistemas de manufactura, entre otros, permitiría reducir los accidentes de trabajo en las industrias y los costos adherentes a éstos, permitiendo un ambiente seguro de trabajo. La validación de los algoritmos de control dará confiabilidad en los procesos reduciendo los tiempos muertos máquina-proceso por fallas no previstas o no detectadas en los algoritmos de control, lo que mejora la productividad de las empresas y reduce los riesgos de daño en los componentes y máquinas utilizadas en los procesos industriales [6].

Hoy en día, los sistemas de control deben ser flexibles para que los procesos de producción y de manufactura industrial también sean flexibles. Los PLC son un tipo especial de computador para automatizar los DES en la industria, cumplen con la flexibilidad necesaria para procesos que tienen una gran cantidad de señales de entradas y

salidas físicas, al igual que, muchas líneas de programación para su respectivo control. Mientras más señales de entradas y salidas se necesitan para el control de los procesos, el desarrollo del algoritmo de control es más complejo y mayor es la incertidumbre del buen funcionamiento de éste y del tiempo que se requiere para su ejecución [48].

IEC-61131-3 permite la utilización de cinco lenguajes de programación para ser implementados en los PLC, sin embargo, solo incluye la sintaxis y semántica de éstos lenguajes. Independientemente, del tipo de PLC y del lenguaje en el que se decida programar los algoritmos de control, estos son desarrollados por los usuarios finales con base en su experiencia y conocimiento del proceso de producción, así como del manejo de un lenguaje específico utilizado por este. Algoritmos de control en LD siguen siendo la herramienta gráfica popular de programación de los PLC, pero el principal problema es que la programación se lleva de forma heurística y el algoritmo de control en LD se dificulta para el análisis y evaluación del desempeño [49].

Las PN son una herramienta gráfica y matemática para modelar y simular comportamientos dinámicos de DES. Una interpretación típica es utilizar los lugares de entradas como señales discretas de entradas, las transiciones como la lógica a cumplir y los lugares de salida como señales discretas de salidas. Además, son capaces de describir comportamientos asíncronos, concurrentes, distribuidos, paralelos, estocásticos y lógicos de los sistemas, características que tienen los algoritmos de control desarrollados en LD, por lo tanto, las PN son la base para la presente investigación.

En la actualidad, continúa la demanda de algoritmos de control para sistemas de producción con mayor complejidad y calidad, que incluya requerimiento de seguridad críticos para los DES hacen esencial la aplicación de métodos formales en la programación de PLCs [2], [39].

Hipótesis

Es factible llevar a cabo el modelado y validación de algoritmos de control para sistemas basados en Controladores Lógicos Programables y desarrollados en el lenguaje de Diagrama Escalera, mediante el uso de la teoría de redes de Petri, aprovechando su representación gráfica y matemática.

Objetivos

La presente investigación tiene como objetivo general, desarrollar una metodología para modelar algoritmos de control desarrollados en el lenguaje de programación diagrama escalera utilizando la teoría de redes de Petri, que permita analizar y evaluar el comportamiento dinámico de estos, así como identificar posibles situaciones de riesgo en el control de procesos y/o máquinas que no se hayan contemplado en el algoritmo de control.

Los objetivos específicos son:

- Analizar las propiedades y características del lenguaje de programación diagrama escalera,
- Revisar la teoría base de las redes de Petri,
- Analizar los enfoques propuestos que tienen correlación entre los métodos diagrama escalera y redes de Petri,
- Proponer en términos de redes de Petri las líneas de control más utilizadas en los diagramas escalera,
- Proponer en términos de redes de Petri los bloques de control de tiempo y conteo,
- Proponer un nuevo enfoque de transformación de algoritmos de control desarrollados en diagrama escalera a redes de Petri,
- Probar el enfoque desarrollado en algoritmos de control de aplicaciones reales.

Contribuciones

En la presente investigación, se propone un nuevo enfoque para modelar algoritmos de control desarrollados en LD, considerando un análisis exhaustivo en el comportamiento de los sistemas basados en PLC, particularmente en las etapas del barrido de este, así como el comportamiento de los estados de bobinas y contactos; energizados o des-energizados. Una descripción breve de las contribuciones es:

1. Modelos en elementos de PN de las líneas de control con mayor frecuencia de utilización en algoritmos de control en LD para DECS.

2. Metodología para transformar algoritmos de control en LD a PN es la base de la propuesta para el modelado correspondiente, considerando el comportamiento de energizado y des-energizado de bobinas.

Productos Generados

Los productos generados de la investigación, a la fecha son:

- Formal design methodology for transforming ladder diagram to Petri nets, J. C. Quezada, J. Medina, E. Flores, J. C. Seck Tuoh y N. Hernández, The International Journal of Advanced Manufacturing Technology, Springer, Volume 73, Combined 5-8, disponible en línea desde mayo de 2014, ISSN 0268-3768, DOI 10.1007/s00170-014-5715-9 [46]
- Simulación mediante PLC-HMI de las Características en Redes de Petri de Sistemas de Eventos Discretos, J. C. Quezada, J. Medina y E. Flores, 2º Congreso Nacional en Tecnologías de la Información 2013, ESTL-UAEH, marzo de 2013, ISBN: 978-607-482-324-0 [54]
- Analysis of Ladder Diagram for a Packaging System with Petri Nets, J. Carlos Quezada, Joselito Medina and Ernesto Flores, XIII Congreso Nacional de Ingeniería Electromecánica y de Sistemas CNIES, IPN, Noviembre de 2012. [55]

Organización de la tesis

La tesis está conformada de una sección que muestra el estado del arte entre los algoritmos de control en LD y las PN para su transformación, modelado y validación entre ambos enfoques. También se describe la justificación de la investigación, sus objetivos y se mencionan los productos científicos presentados en revistas y/o eventos académicos. La tesis tiene cinco capítulos, que a continuación se describen:

En el capítulo 1 se presenta el ámbito de aplicación del estándar internacional IEC 61131 *Controladores programables*, haciendo énfasis en el lenguaje de programación *Diagrama Escalera* para el desarrollo de algoritmos de control de DES.

La teoría de PN es revisada en el capítulo 2. Se describen los elementos gráficos de las PN, las reglas de habilitado y disparo de transiciones para simular y modelar el comportamiento del sistema, los métodos de análisis matemático que relacionan los lugares con las transiciones, la matriz de incidencia y la ecuación de estado para describir el comportamiento dinámico de las redes.

La transformación de algoritmos de control en LD a PN se muestra en el capítulo 3. Se propone una distribución de señales de los elementos del LD a términos en PN, son analizadas líneas de control utilizadas en los LD para DES, se propone y se define la extensión LDPN con sus elementos y reglas para modelar y simular el comportamiento dinámico de dicha red.

El análisis de las fallas reales de corto-circuito y/o circuito-abierto que se pueden presentar en los sub-sistemas de señales de entradas físicas es presentado en el capítulo 4, además, es propuesto un enfoque para validación de la LDPN en función de las fallas y poder determinar los posibles errores en los algoritmos de control en LD.

Por último, en el capítulo 5 se presenta la aplicación de los enfoques propuestos de transformación, modelado, simulación y validación de tres algoritmos de control en LD de sistemas reales para mostrar los beneficios de la investigación. De igual forma, al final se presentan las conclusiones y los trabajos a futuro que permitan continuar fortaleciendo la línea de investigación.

Capítulo 1

Estándar IEC-61131 Controladores Programables

El estándar IEC-61131 Controladores Programables hoy en día es utilizado en la industria de la automatización por fabricantes para el desarrollo y diseño de sus tecnologías (Hardware), así como de las plataformas de programación (Software). El estándar se constituye de ocho apartados, los cuales se describen a continuación de forma general [50], [51]. Posteriormente, se analiza de forma particular el apartado IEC 61131-3 correspondiente a los lenguajes de programación para sistemas basados en controladores programables.

- IEC-61131-1 Información general: Contiene definiciones y características funcionales típicas las cuales distinguen a los sistemas basados en PLC de otros sistemas. Describe el procesamiento cíclico de la ejecución de los algoritmos de control, resaltando la imagen almacenada de los valores de señales de entrada y salida.
- IEC-61131-2 Requerimientos y pruebas de equipos: Este apartado define los requerimientos eléctricos, mecánicos y funcionales de los dispositivos y las pruebas de calidad que deben cumplir, además de las condiciones ambientales que deben soportar.
- IEC-61131-3 Lenguajes de programación: aquí se describe la sintaxis y la semántica de los cinco lenguajes de programación para PLC.

- IEC-61131-4 Guías de usuario: la sección es una ayuda para los usuarios en todas las fases del proyecto de automatización.
- IEC-61131-5 Comunicaciones: refiere a la comunicación entre diferentes marcas de PLC y con otros dispositivos.
- IEC-61131-6 Seguridad funcional: el objetivo es adaptar los requerimientos de los estándares IEC 61508 e IEC 62061 de seguridad funcional de los sistemas de seguridad programable y seguridad en maquinarias a los PLC.
- IEC-61313-7 Programación de control difuso: el objetivo de esta sección es estandarizar entre los fabricantes y usuarios la integración de aplicaciones de control difuso basadas en la apartado 3 del estándar.
- IEC-61313-8 Guía para la aplicación e implantación de los lenguajes de programación: ofrece interpretaciones para cuestiones no contempladas en el estándar, como instrucciones de uso para el usuario final y asistencia en la programación.

Conceptos del estándar sobre sistemas basados en controladores lógicos, así como la sintaxis y semántica del lenguaje de programación LD son considerados en el presente trabajo.

1.1. Sistemas basados en PLC

Los sistemas de automatización basados en PLC continúan siendo la base para el control y/o monitoreo de señales de los DES en los procesos industriales. Los sistemas de control basados en PLC se caracterizan por tener señales de entradas y salidas físicas que vienen y van de sensores y actuadores respectivamente, los cuales están en contacto directo con el proceso de producción o de manufactura, como muestra la Fig. 1.1 donde puede observarse que el sistema tiene inmerso el programa de usuario o algoritmo de control en LD, el cual es el objeto del presente trabajo; primero transformarlo a la teoría de PN, obtener un modelo gráfico y matemático y validar su comportamiento en los casos de falla de corto-circuito o circuito-abierto en el sub-sistema de entradas.



Figura 1.1: Diagrama a bloques de elementos involucrados con el algoritmo de control.

Las señales de entrada física, un controlador programable (algoritmo de control) y las señales de salida física constituyen la base de un sistema basado en PLC.

1.2. Controlador Lógico Programable

Para el control de procesos industriales se requiere de la medición de señales digitales y/o analógicas. Una señal digital es aquella que puede tener solo dos estados, por ejemplo, abierto o cerrado. Mientras que una señal analógica es continua en el tiempo, por ende, puede tener valores en un cierto intervalo, por ejemplo; 0 - 100 % correspondiente al nivel en un tanque. Ambas señales son acondicionadas a niveles de voltaje o corriente para su interpretación en los módulos de entrada y/o salida de los sistemas basados en PLC.

Del apartado IEC 61131-1 [53] se toma la definición de PLC.

Controlador Lógico Programable:

Definición 1.1 Sistema electrónico operando digitalmente, diseñado para uso en ambiente industrial, el cual usa una memoria programable para el almacenamiento interno de instrucciones de usuario orientadas para implementar funciones específicas tales como lógica, secuencias, retardos, conteo y aritmética para control de varios tipos de máquinas o procesos a través de entradas y salidas digitales o analógicas.

Del estándar IEC 61131-2 [52], desde una perspectiva electrónica, una señal de entrada digital es la conversión de una señal con dos estados a una señal binaria de un solo bit. Mientras que una señal de salida digital es la conversión de una señal binaria de un solo bit a una señal de dos estados.

El algoritmo de control se evalúa de forma periódica, siendo el ciclo de trabajo de los sistemas basados en PLC la característica principal que los diferencia de un compu-

tador tradicional o de otros sistemas programables.

1.2.1. Ciclo de trabajo del PLC

La ejecución periódica o cíclica es la base de funcionamiento de los sistemas basados en PLC. La Figura 1.2 [46] solo muestra las acciones cíclicas con respecto al algoritmo de control. Un sistema basado en PLC considera etapas de auto-chequeo y de comunicaciones.

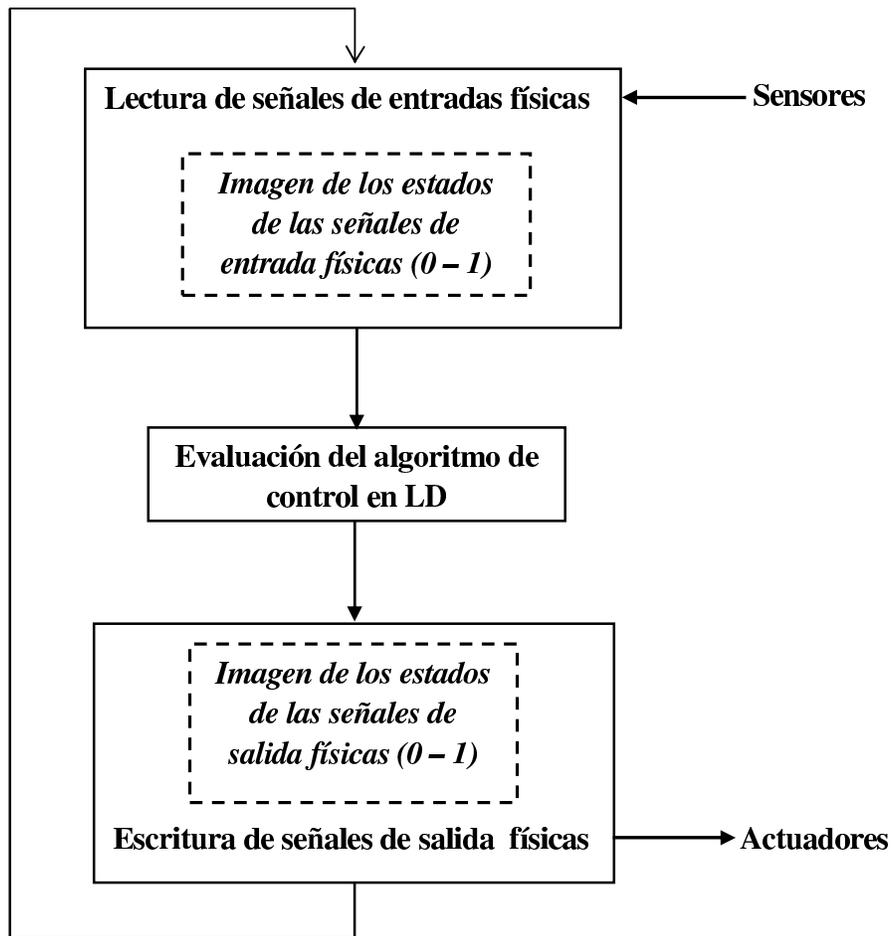


Figura 1.2: Ciclo de trabajo con respecto al algoritmo de control en LD.

El ciclo de trabajo es básicamente; lectura de las señales de entrada física, el sistema PLC realiza una copia de los estados de estas señales (0 o 1), con esta copia, se realiza la evaluación del algoritmo de control en LD, los resultados de los estados de las señales de salida son copiados para posteriormente enviar a los módulos de salida la acción de energizar o des-energizar bobinas.

Idealmente, durante el tiempo de evaluación del algoritmo de control en un ciclo, un cambio en el estado de las señales de entrada física no afecta la ejecución del control, si no hasta que es actualizada la imagen de los estados de las señales de entrada. Lo anterior permite evaluar de forma independiente el algoritmo de control en función de los posibles estados de las señales de entrada física, lo cual es la base de la presente investigación para modelar el comportamiento de los algoritmos de control en LD con PN, por lo que a continuación se analizan las líneas de control con mayor frecuencia de utilización en los algoritmos de control en LD.

En la siguiente sección, se analiza el lenguaje de programación LD.

1.3. Lenguaje de programación diagrama escalera

LD es uno de los cinco lenguajes de programación considerado en el estándar IEC-61131-3 para desarrollar algoritmos de control para PLC. LD es considerado un lenguaje de tipo gráfico y tiene como base de funcionamiento el comportamiento de un relé electromecánico, de [2] y [7] se considera la definición de LD respectivamente:

Diagrama Escalera:

Definición 1.2 Uno o más redes de contactos, bobinas, funciones representadas gráficamente, bloques de funciones, elementos de datos, etiquetas y elementos de conexión, delimitados por la parte izquierda y derecha por rieles de energía.

Definición 1.3 modelado de redes de elementos electromecánicos funcionando simultáneamente tales como bobinas y contactos de relés, temporizadores, contadores, etc.”.

La Figura 1.3 muestra la equivalencia entre el principio de funcionamiento de un relé electromecánico y el lenguaje LD. En el circuito eléctrico, la señal *sensor* abre o cierra el circuito, lo que hace que se des-energice o energice la *bobina*, por consiguiente, sus contactos NC y NO cambian de estado cuando la *bobina* está energizada y están

en su estado inicial si la *bobina* está des-energizada. En el LD, se tiene el mismo comportamiento, si la señal *sensor* está activa, energiza la señal *bobina*, la cual activa o des-activa sus contactos en la línea dos.

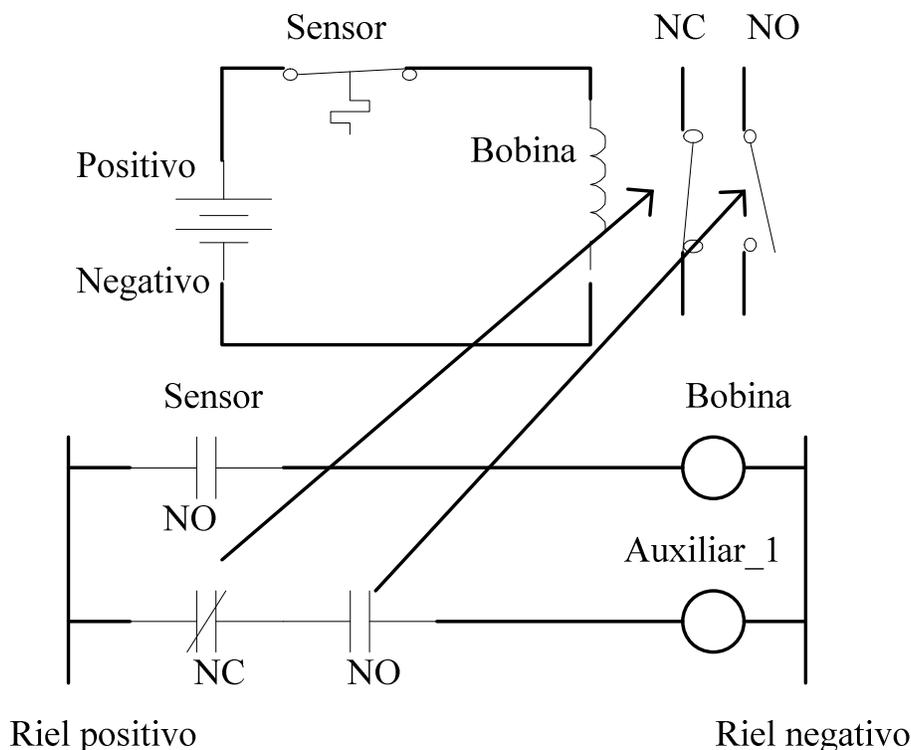


Figura 1.3: Equivalencia entre un relé electromecánico y el correspondiente código en LD.

Los rieles de energía simulan la alimentación a los sensores y actuadores, es decir, el riel izquierdo es el positivo de una señal de Corriente Directa (CD) o la línea de un sistema de Corriente Alterna (CA). Mientras que el riel derecho es el negativo de un sistema de CD o el neutro de uno de CA. Una condicionante en las líneas de control básicas es que deben terminar con una bobina, que representa el cierre del circuito eléctrico.

Un contacto puede estar normalmente cerrado (NC) o normalmente abierto (NO). Para un sistema basado en PLC, un contacto NO y/o NC puede provenir de un sensor mecánico o electrónico que cierra o abre el circuito eléctrico al módulo de entradas físicas (señal sensor en la Figura 1.3), el cual detecta presencia o ausencia de voltaje para el

estado (0 o 1) de la variable correspondiente. El nivel de voltaje y tipo de señal (directa o alterna) está en función del tipo de módulo de entradas. También, un contacto NO y/o NC puede ser una señal interna de memoria que esté ligada a una bobina también interna. Una señal de entrada física puede considerarse cuantas veces sea necesaria en el algoritmo de control mediante contactos NO y/o NC. De igual forma, una bobina de salida o interna puede tener los contactos NO y/o NC que requiera el algoritmo de control en una o diversas líneas.

De igual forma, de [53], se considera la definición de contactos y bobinas que son utilizados en los algoritmos de control en LD.

Contacto:

Definición 1.4 Es un elemento el cual transmite un estado a la unión horizontal sobre su lado derecho el cual es igual a la lógica booleana AND del estado de la unión horizontal y su lado izquierdo con una función booleana apropiada de una variable de entrada, salida o memoria. Un contacto no modifica el valor de la variable booleana asociada.

Para la presente investigación, se omite la sintaxis referente a las uniones horizontales y verticales en los algoritmos de control en LD. Se considera que un contacto NO no-activo no permite el flujo de energía en la línea de control y viceversa, mientras que un contacto NC no-activo permite el flujo de energía en la línea de control y viceversa.

La Figura 1.4 muestra ambos comportamientos de los contactos NO y NC de cuando están activos y no-activos. En la primer línea de control, solo el contacto NO del Sensor_1 está activo y el contacto NO del sensor_2 no, lo que genera que la bobina de salida Actuador_1 esté des-energizada.

En la segunda línea del LD, los contactos NC de las señales Sensor_3 y Sensor_4 están no-activos, permitiendo el flujo de energía hacia la bobina de salida Actuador_2.

Bobina:

Definición 1.5 Una bobina copia el estado de la unión sobre su izquierda a la unión sobre su derecha sin modificación, y almacena una función apropiada del estado o de la transición de la unión izquierda dentro de una variable booleana asociada.

Una bobina estándar es energizada cuando el flujo de energía a través de la línea de control llega a esta, lo cual eléctricamente cierra el circuito y está energizada. De

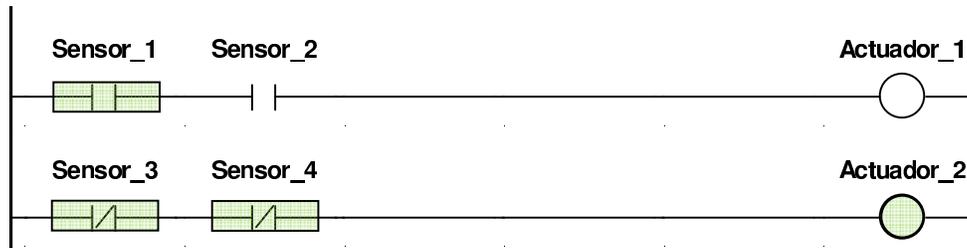


Figura 1.4: Comportamiento de contactos NO y NC.

lo contrario, su estado es des-energizada. La bobina negada tiene el comportamiento inverso, es decir, cuando el flujo de energía de la línea de control no llega a la bobina, esta está energizada, y viceversa.

La Figura 1.5 muestra el comportamiento de ambas bobinas. En la primer línea de control los contactos NO de las señales Sensor_1 y Sensor_2 están activos y permiten el flujo de energía a la bobina de salida Actuador_1. Por otro lado, La bobina negada Actuador_2 está des-energizada ya que los contactos NC de las señales Sensor_3 y Sensor_4 están desactivados y permiten el flujo de energía hacia esta.

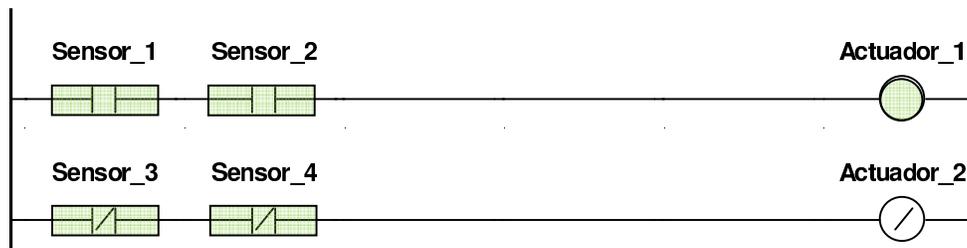


Figura 1.5: Comportamiento de una bobina estándar y una negada.

Ahora bien, arreglos de contactos y bobinas en líneas de control en LD generarán comportamientos que en la mayoría de los casos puede ser representado por lógicos utilizados en el diseño electrónico digital o en el modelado de procesos con DES [46], [54].

1.4. Análisis de líneas de control en los algoritmos en LD

Comportamientos lógicos, secuenciales, asíncronos, concurrentes, entre otros, pueden ser implementados a través de líneas de control en LD. Un análisis de los mismos es presentado.

A) Línea de control de contactos en serie - lógica *AND*:

La Figura 1.6, muestra la conexión de contactos en serie. Solo cuando todos los contactos permiten el flujo de energía entonces la bobina de salida es energizada, es decir, $IN1 = IN2 = \dots = INn = 1$, entonces la bobina $OUT1 = 1$. Si por uno o más contactos no fluye la energía (contacto desactivado) entonces, la bobina de salida $OUT1$ es des-energizada. Esta línea de control con solo contactos en serie tiene el comportamiento de la lógica *AND*, donde el número de contactos es el número de entradas de dicha lógica.

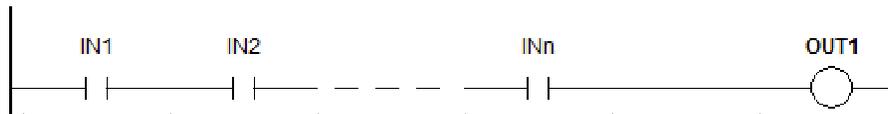


Figura 1.6: Contactos en serie.

B) Contactos en paralelo - lógica *OR*:

La línea de control con solo contactos en paralelo permite que la bobina de salida tenga el comportamiento de la lógica *OR*. La Figura 1.7 muestra este tipo de conexión, si alguno de los contactos permite el flujo de energía $IN1 = 1$ o $IN2 = 1$ o $INn = 1$, entonces la bobina $OUT1 = 1$, que es la condición de energizada. Solo cuando todos los contactos de entrada están inactivos $IN1 = IN2 = \dots = INn = 0$, entonces la bobina $OUT1 = 0$ y por lo tanto está des-energizada.



Figura 1.7: Contactos en paralelo.

C) Contactos en serie/paralelo - lógica *AND-OR*:

La conexión de contactos en serie - paralelo permite energizar una bobina por diversas rutas de flujo de energía, la cual está en función de las combinaciones de los contactos hacia la bobina. En la Figura 1.8 puede observarse que la salida *OUT1* puede ser energizada a través del flujo de energía por la ruta de contactos $IN1 = IN2 = INn = 1$ o por la ruta $IN1 = IN3 = INn = 1$ o ambos. Puede observarse que las señales *IN2* e *IN3* pueden estar ambas activas, es decir, $IN1 = IN2 = IN3 = INn = 1$ y permitir energizar la bobina $OUT1 = 1$, sin afectar el comportamiento del algoritmo de control.

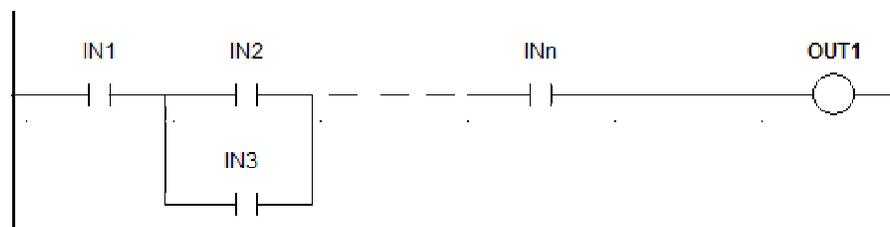


Figura 1.8: Contactos serie - paralelo.

D) Lógica de bobinas de retención *Set - Reset*:

Las señales de retención asíncronas de *Set* (colocar a 1) y *Reset* (colocar a 0) solo necesitan ser activadas por un instante para mantener el estado cero o uno, correspondiente de la bobina, es decir, no es necesario que la secuencia de contactos que

activaron a la bobina continúen permitiendo el flujo de energía hacia esta.

En los algoritmos de control en LD es necesario que ambas bobinas tengan la misma dirección. La Figura 1.9 muestra dos líneas de control con bobinas *Set* y *Reset*. Cuando $IN1 = 1$, entonces la bobina $Set - OUT1 = 1$ es energizada, y así se mantendrá independientemente que el contacto cambie de estado $IN1 = 0$. Si el contacto $IN2 = 1$ es activo, entonces la bobina $Set - OUT1 = 0$ es des-energizando.

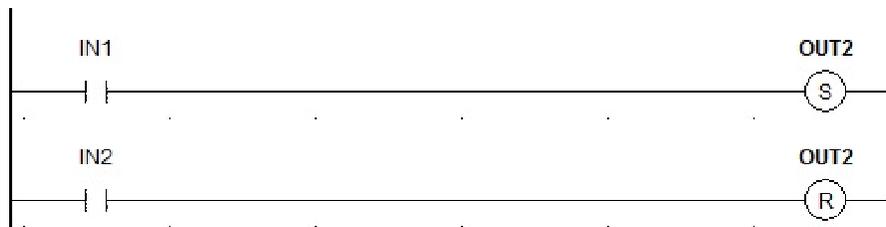


Figura 1.9: Lógica de bobinas de retención *Set-Reset*

E) Contactos de bloqueo de línea de control - *interlocking*:

Colocar un contacto de una bobina en otra línea de control en el algoritmo de control en LD, permite el control secuencial a través de activar o des-activar el flujo de energía de dicho contacto, y por consecuencia de energizar o des-energizar una bobina. Esta propiedad es bastante utilizada en los Controladores de Sistemas de Eventos Discretos (DESC, por sus siglas en inglés) para generar control secuencial cíclico en los procesos industriales. La Figura 1.10 muestra dos líneas de control, para energizar la bobina $OUT1$ se requiere que $IN1 = 1 \dots = INn = 1$ y $OUT2 = 0$. De igual forma, para energizar la bobina $OUT2$ es necesario que $IM1 = 1 \dots = IMm = 1$ y $OUT1 = 0$. Nunca puede darse el estado de que ambas bobinas $OUT1$ y $OUT2$ estén energizadas.

F) Auto-lazo - *retroalimentación*:

El auto-lazo también es muy utilizado en los algoritmos de control en LD para mantener activa una bobina utilizando un contacto de esta en la misma línea de control. La Figura 1.11, muestra la función de auto-lazo. Es necesario que la condición de

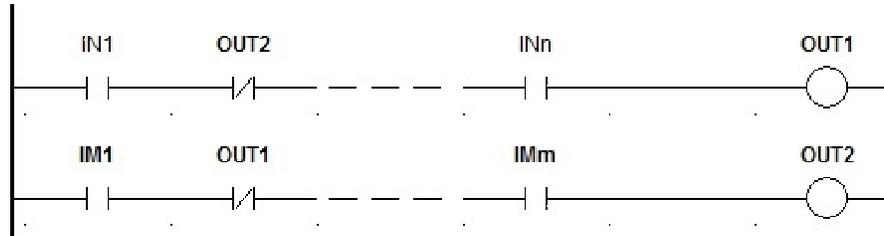


Figura 1.10: Contactos para bloqueo de línea de control.

activación de la bobina $OUT1$ se da cuando $IN1 = IN2 = \dots = INn = 1$. Al energizar la bobina $OUT1$ su contacto $OUT1$ permite el flujo de energía, manteniendo la bobina energizada a través de la ruta $OUT1 = IN2 = \dots = INn = 1$, aunque la señal $IN1 = 0$ cambie de estado.

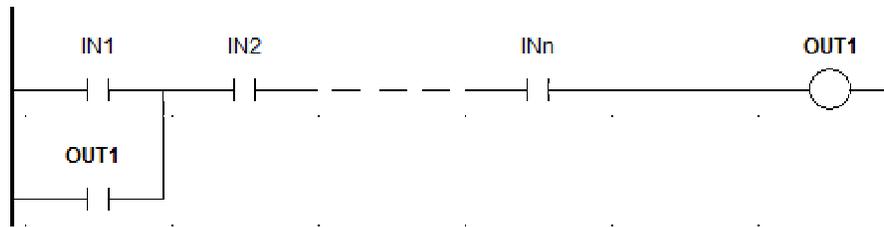


Figura 1.11: Auto-lazo.

G) Ejecución secuencial:

Una característica importante de los DES para procesos industriales es la ejecución secuencial de actividades. La Figura 1.12 muestra este comportamiento, si la señal de entrada $IN1 = 1$ en la primera línea de control, entonces permite flujo de energía a la señal de salida $OUT1 = 1$, la cual activa su contacto en la segunda línea de control, permitiendo que cuando la señal $IN2 = 1$ se energice la bobina de salida $OUT2 = 1$. Si la señal $IN1$ no se activa, la señal de salida $OUT2$ no se puede activar, aunque se activara la señal $IN2$.

H) Decisión o conflicto:

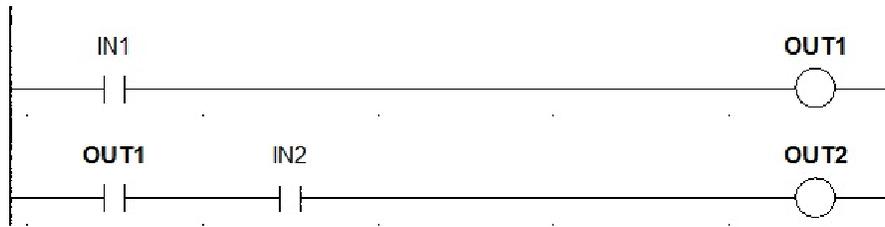


Figura 1.12: Comportamiento de ejecución secuencial en LD.

El comportamiento de decisión o conflicto se muestra en la Figura 1.13. Si la señal de entrada $IN1 = 1$ está activa, entonces permite que las tres líneas de control puedan energizar su señal de salida correspondiente, la cual queda en función de las señales de entrada $IN2$, $IN3$ o $IN4$. Sin embargo, este algoritmo de control puede tener activas una, o dos o las tres señales de salida $OUT1$, $OUT2$ y $OUT3$ si todas las señales de entrada están activas, es decir, $IN1 = IN2 = IN3 = IN4 = 1$.

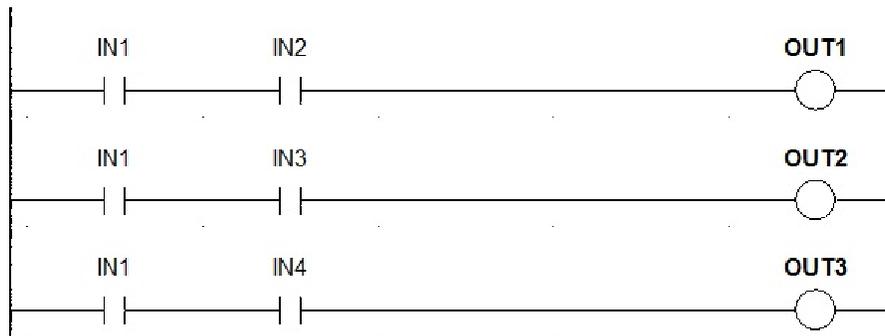


Figura 1.13: Comportamiento de decisión o conflicto en LD.

I) Concurrencia:

La concurrencia o ejecución paralela de actividades se muestra en la Figura 1.14. Si la señal de entrada $IN1 = 1$ está activa, entonces las señales de salida $OUT1 = OUT2 = OUT3 = 1$ están energizadas. Sin embargo, si la señal de entrada no está activa $IN1 = 0$, todas las señales de salida están des-energizadas.

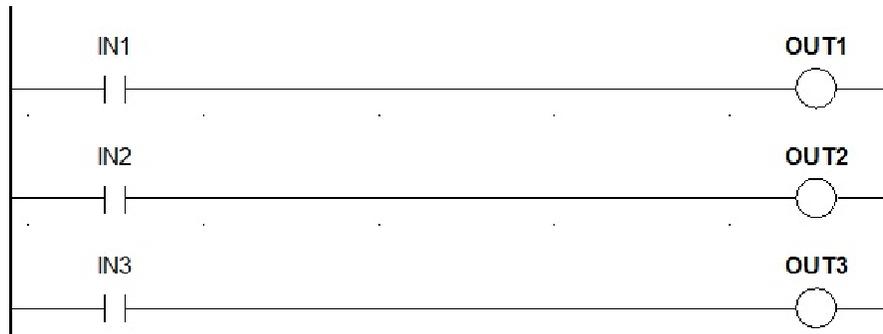


Figura 1.14: Comportamiento de concurrencia en LD.

1.5. Conclusiones del capítulo

Los sistemas basados en PLC se caracterizan por tener señales de entradas físicas - algoritmo de control - señales de salidas físicas el cual es un comportamiento que puede ser modelado por la teoría de PN. En particular, las bobinas y contactos que son los elementos base del lenguaje de programación LD para el desarrollo de algoritmos de control también pueden ser representados por lugares y transiciones, así como el comportamiento de energizado y des-energizado con las reglas de habilitado y disparo correspondientes.

Por otro lado, es de suma importancia considerar el comportamiento base de los PLC, que es su ciclo de trabajo. La evaluación del algoritmo de control solo se lleva una vez en cada ciclo con la copia de los estados de las señales de entrada físicas, lo que permite poder modelar el comportamiento dinámico del proceso en función de la secuencia de activación y/o des-activación de los sensores.

Desde las características de los sensores, su circuito de conexión eléctrica puede ser NO y/o NC, y en el algoritmo de control puede considerarse mantener lógica directa o invertir en las líneas de control al colocar un contacto de esta señal con lógica inversa. Además, una sola señal de un sensor pueden tener n -contactos NO y/o NC en diferentes líneas de control.

Otra situación a considerar en los procesos, es la condición de qué actuadores son NO o NC en su condición inicial y que es con base en la seguridad del proceso en condiciones de falla de energía, es decir, que actuadores deben quedarse abiertos o activos y cuales cerrados cuando existe falta de energía eléctrica, o neumática o hidráulicas,

entre otras. Este comportamiento debe ser considerado en el diseño de los algoritmos de control, en particular, en el energizado y des-energizado de las bobinas en LD.

Los comportamientos lógicos de los DES son comúnmente utilizados en líneas de control en los algoritmos en LD para la ejecución de actividades; ejecución secuencial, en paralelo o concurrentes, decisión o conflicto, agrupación, entre otros.

Capítulo 2

Redes de Petri

Las PN son una teoría propuesta por Carl Adam Petri como resultado de su tesis doctoral, presentada en el año de 1962 en Alemania. Las PN son una herramienta de modelado gráfico y matemático para representar sistemas con eventos discretos, así como su comportamiento dinámico. Las PN permiten describir y estudiar procesos con características de concurrencia, asíncronos, distribuidos, no determinísticos y/o estocásticos [56].

El capítulo muestra las propiedades fundamentales de análisis matemático de las PN.

2.1. Definición formal de red de Petri (PN)

Los DES se basan fundamentalmente en los conceptos de eventos y condiciones. Un evento genera un cambio en las variables de estado y una condición contiene la relación entre dos eventos. En una línea de control en LD, el conjunto de contactos representa la condición y el energizado o des-energizado de la bobina representa el evento.

Las PN son un grafo dirigido, ponderado y bipartito, con dos tipos de componentes; *lugares* y *transiciones*. La relación entre estos elementos es a través de arcos dirigidos que une lugares con transiciones y transiciones con lugares. Los arcos tienen un *peso* que es requerido en los lugares de entrada o transmitido a uno o más lugares de salida. El marcado inicial son las condiciones iniciales en cada lugar del sistema. La Tabla 2.1 muestra la definición formal de una PN en su forma básica.

Tabla 2.1: Definición formal de una PN

Una red de Petri es una 5-tupla, $PN=(P, T, F, W, M)$ donde:

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

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

$F \subseteq (P \times T) \cup (T \times P)$ es un conjunto de arcos,

$W : F \rightarrow \{1, 2, 3, \dots\}$ es una función de peso,

$M_0 : P \rightarrow \{0, 1, 2, \dots\}$ es un marcado inicial, y

$P \cap T = \emptyset$ y $P \cup T \neq \emptyset$

La Tabla 2.2 presenta los símbolos gráficos usados en una PN.

Un arco desde un lugar p_j a una transición t_i define un lugar de entrada p_j a la transición t_i , lo cual se denota por (p_j, t_i) . Y un arco desde una transición t_i hacia un lugar p_j define a un lugar de salida p_j de la transición t_i , el cual se denota por (t_i, p_j) . El peso para un arco de un lugar de entrada a una transición es denotado por $w(p_j, t_i)$ y para un arco que une una transición con un lugar de salida es denotado por $w(t_i, p_j)$.

El conjunto de lugares de entrada para una transición $t \in T$ se representa por el conjunto $\bullet t = \{p | p \text{ es un lugar de entrada de } t\}$. El conjunto de lugares de salida de la transición $t \in T$ se representa por el conjunto $t \bullet = \{p | p \text{ es un lugar de salida de } t\}$. Para el conjunto de transiciones de entrada a $p \in P$ se representa por $\bullet p = \{p | (t, p) \in F\}$, y para el conjunto de transiciones de salida de $p \in P$ se representa por $p \bullet = \{p | (p, t) \in F\}$. La Figura 2.1 es la representación gráfica de los conjuntos.

Las *marcas* en los lugares son utilizadas para simular el comportamiento dinámico del sistema. Y el arco inhibitor tiene la función de habilitar una transición cuando el lugar de entrada a dicha transición no tienen marcas.

Tabla 2.2: Símbolos gráficos utilizados en una PN

Símbolo	Descripción
○	Lugar.
●	Marca.
█	Transición.
→	Arco dirigido.
—○	Arco inhibidor.

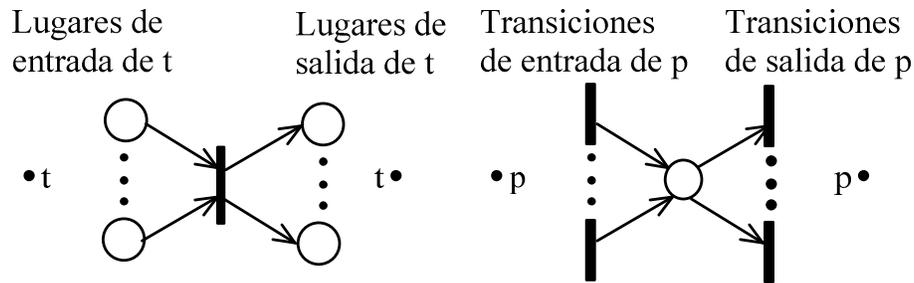


Figura 2.1: Representación gráfica de los conjuntos de lugares y transiciones.

2.2. Reglas de disparo de las PN

Para la simulación del comportamiento dinámico, con base a un marcado inicial M_0 de una PN, la siguiente regla de transición es considerada [57]:

1. Una transición t_i está activa si cada uno de los lugares p_j contienen al menos $w(p_j, t_i)$ marcas, es decir, $M(p_j) \geq w(p_j, t_i) \forall p_j \in \bullet t_i$.
2. Una transición habilitada puede ser o no ser disparada, dependiendo si el evento realmente ocurre o no.

- Como resultado de disparar una transición habilitada t , se eliminan $w(p_j, t_i)$ marcas de cada lugar p_j a la entrada de t_i y se adicionan $w(t_i, p_j)$ a cada lugar de salida p_j de la transición t_i , es decir, un nuevo marcado es obtenido, representado por $M'(p_j) = M(p_j) - w(p_j, t_i) + w(t_i, p_j), \forall p \in P$.

La Figura 2.2 muestra el comportamiento de una PN antes y después de disparar la transición t_1 . Los arcos que no indican su peso, tiene un valor de uno. La transición t_1 está habilitada ya que se cumple que para todos sus lugares de entrada el número de marcas es mayor o igual al peso de los arcos correspondientes, es decir, $M(p_1) = w(p_1, t_1) = 1$, $M(p_2) > w(p_2, t_1) = 1$ y $M(p_3) = w(p_3, t_1) = 3$. Cuando es disparada la transición t_1 , se consume una marca del lugar p_1 , dos del lugar p_2 y tres del lugar p_3 , además, al lugar de salida p_4 se le añaden dos marcas y a p_5 se le adiciona una marca, quedando ambos lugares con dos marcas. El lugar de entrada p_2 queda con dos marcas, sin embargo, la transición t_1 queda inhabilitada.

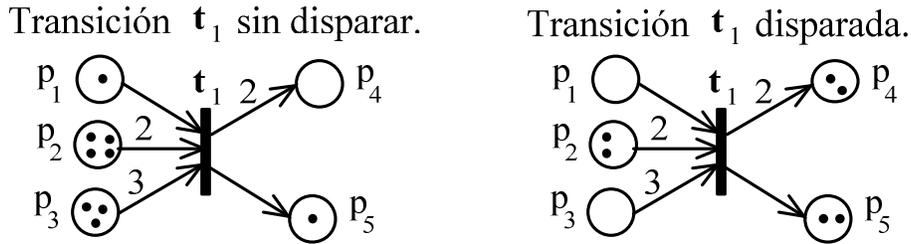


Figura 2.2: Ejemplo de disparo de una transición en PN.

Cuando en una PN todos sus arcos tienen peso unitario, es decir, $w(p_j, t_i) = w(t_i, p_j) = 1$ es llamada PN ordinaria.

Existen en las PN dos tipos de transiciones especiales; la *transición fuente* la cual no tiene lugares de entrada, y la *transición sumidero* la cual no tiene lugares de salida, como muestra la Fig. 2.3.

La transición fuente siempre está habilitada, adicionando marcas a sus lugares de salida, mientras que el disparo de una transición sumidero consume todas las marcas de sus lugares de entrada.

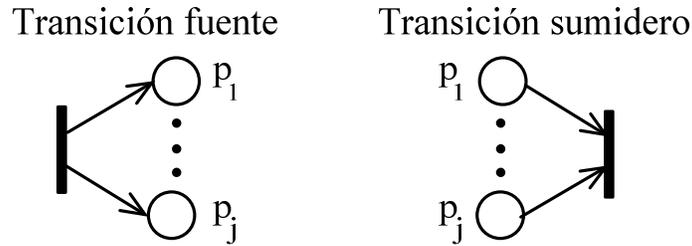


Figura 2.3: Transición fuente y sumidero.

2.3. Representación de comportamientos de DES con PN

El comportamiento de ejecución secuencial, decisión o conflicto, concurrencia, sincronización, agrupación e inhibición son características de los DES y pueden ser modelados con las PN [57].

A) Ejecución secuencial:

La ejecución secuencial refleja la dependencia causal entre dos o más eventos. La Figura 2.4 muestra una PN con este comportamiento, el lugar P_1 tiene una marca, la cual es removida al disparar la transición t_1 y se coloca una marca en lugar P_2 , la cual habilita la transición t_2 , si se dispara, entonces se coloca una marca en el lugar P_3 y se consume la del lugar P_2 .

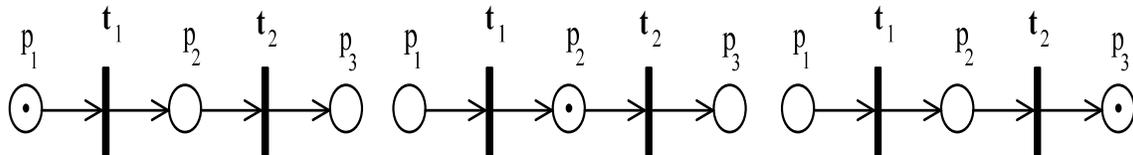


Figura 2.4: Comportamiento de ejecución secuencial en PN.

B) Decisión o conflicto:

La Figura 2.5 muestra el comportamiento de decisión o conflicto de eventos. Las transiciones t_1 , t_2 y t_3 están habilitadas, ya que el lugar de entrada P_1 tiene una marca. El disparo de la transición t_2 deshabilita a las otras dos al consumir la marca del lugar de entrada y colocando una marca en el lugar de salida P_3 .

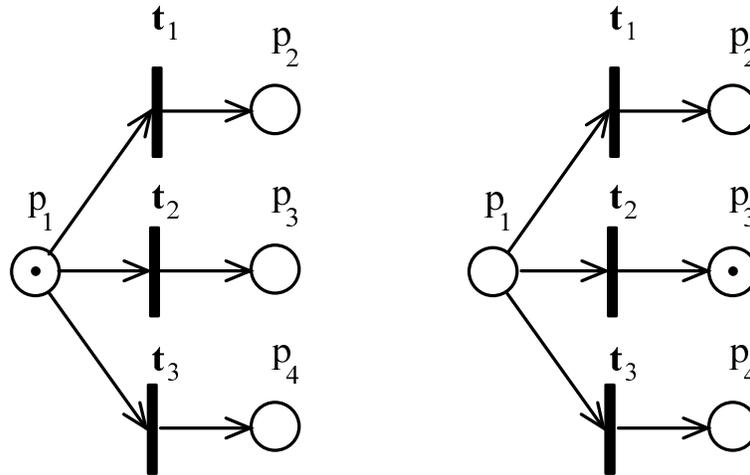


Figura 2.5: Comportamiento de decisión o conflicto en PN.

C) Concurrencia:

La concurrencia es uno de los comportamientos más importantes en los DES, al poder ejecutar eventos en paralelo. La Figura 2.6 muestra que las transiciones t_2 , t_3 y t_4 están habilitadas y son concurrentes o paralelas. El disparo de las transiciones t_2 y t_4 consumen las marcas de los lugares P_1 y P_3 y colocan marca en los lugares de salida P_4 y P_6 y la transición t_3 continua habilitada por la marca que tiene el lugar P_2 .

D) Sincronización:

El comportamiento de sincronización de eventos se muestra en la Figura 2.7. Se requiere que los lugares de entrada P_1 , P_2 y P_3 tengan marca para habilitar la transición t_1 , la cual al ser disparada, consume las marcas de dichos lugares y coloca una marca en el lugar de salida P_4 .

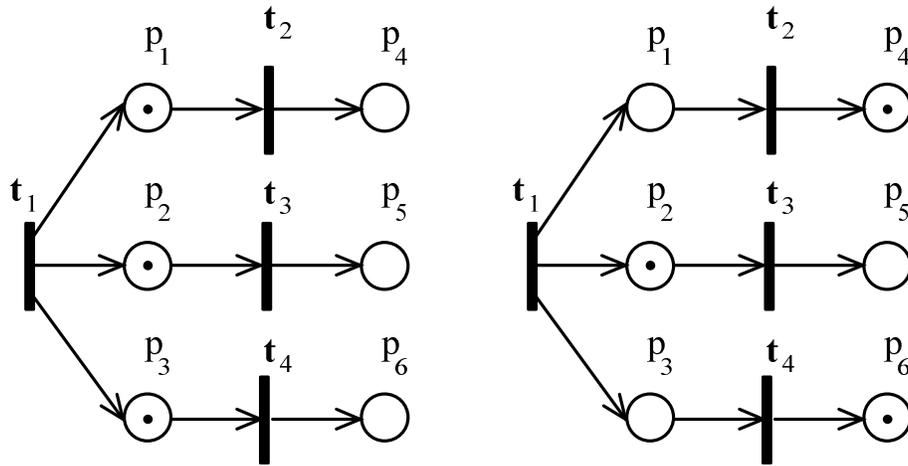


Figura 2.6: Comportamiento de concurrencia o paralelismo en PN.

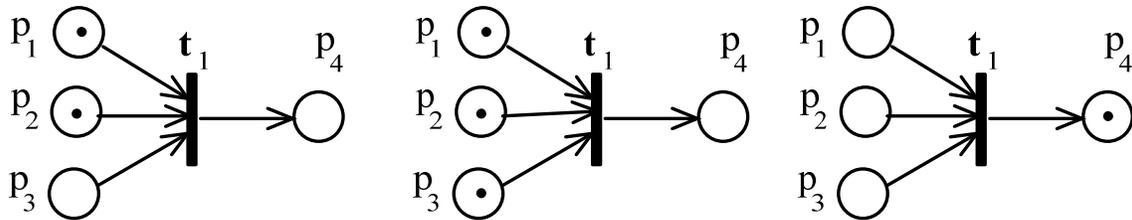


Figura 2.7: Comportamiento de sincronización en PN.

E) Agrupación:

La Figura 2.8 muestra el comportamiento de agrupación de eventos. Las transiciones t_1 , t_2 y t_3 pueden ir agrupando marcas en el lugar P_1 , la cual habilita la transición t_4 .

F) Inhibición:

El comportamiento de inhibición de un evento gráficamente en PN se realiza con un arco inhibidor que termina con un círculo y realiza el comportamiento inverso de un arco habilitador. La Figura 2.9 muestra una PN donde la transición t_1 tiene dos lugares de entrada P_1 y P_2 con arco inhibidor y habilitador respectivamente. La transición t_1 es habilitada cuando el lugar P_1 no tiene marca y el lugar P_2 si tiene

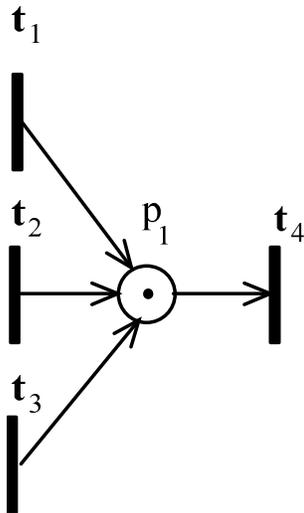


Figura 2.8: Comportamiento de agrupación en PN.

marca. Si es disparada t_1 se consume la marca del lugar P_2 y coloca marca en el lugar de salida P_3 .

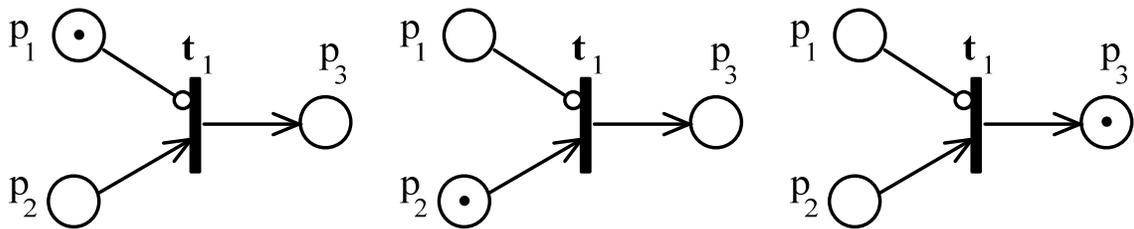


Figura 2.9: Comportamiento de inhibición en PN.

G) Auto-lazo:

Un auto lazo en PN tiene la característica que el lugar de salida también es lugar de entrada de una misma transición. La Figura 2.10 muestra la PN correspondiente. Sin embargo, una marca en el lugar P_1 , y el disparo continuo de la transición t_1 no ocasiona cambio en la PN. En [56], proponen transformar la PN de auto-lazo a lazo, como muestra la Figura 2.11. Esta transformación, solo implica que el disparo secuencial de las transiciones t_2 y t_1 la marca del lugar P_1 se “retrase” para volver

a ser colocada en el lugar.

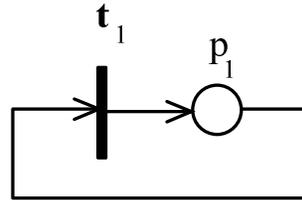


Figura 2.10: PN de auto-lazo.

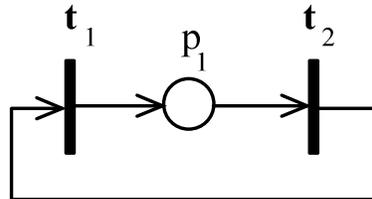


Figura 2.11: Transformación de la PN de auto-lazo a lazo.

Las PN gráficamente son una herramienta útil para visualizar la simulación y el comportamiento de un proceso, aunado a esto, existen herramientas de análisis matemático que permiten modelar y simular el comportamiento dinámico de los sistemas [56].

2.4. Propiedades de la PN

Una de las propiedades de las PN como herramienta de modelado es la capacidad de poder analizar un sistema que al dividirse en partes y obtener su modelo de cada una de estas, y al integrar los modelos se puede construir el modelo total del sistema. La teoría de PN matemáticamente tiene herramientas de análisis cuantitativo que ayudan a describir los modelos, simular, verificar y validar el comportamiento de los DES.

2.4.1. Árbol de alcanzabilidad

Alcanzabilidad es un concepto fundamental para el análisis de las propiedades dinámicas de los sistemas. El disparo de transiciones en las PN originan nuevos marca-

dos de la red.

Alcanzabilidad - Reachability :

Definición 2.1 Se refiere a secuencia de disparos σ de transiciones que permiten alcanzar un marcado M_n a partir del marcado inicial M_0 , donde $\sigma = M_0 t_1 M_1 t_2 M_2 \cdots t_n M_n$ y es denotado por $R(M_0)$.

Una forma de determinar el conjunto de estados que puede alcanzar la red, es construyendo un árbol de alcance, el cual guarda en cada nodo un estado del sistema, y los arcos que conectan un nodo con nodos inferiores indican la transición habilitada a disparar. Una metodología para la construcción del árbol de alcanzabilidad es [57]:

1. La raíz del árbol es un nodo que almacena el estado inicial del sistema, M_0 .
2. Para cada nodo del árbol se generan los tantos nodos hijo como transiciones existan activadas en la PN que representa dicho nodo.
3. Si se genera un nodo hijo que ya existe en el árbol, éste se marca como nodo “old” y no se pueden generar nuevos nodos de él.
4. Un nodo que no tenga transición habilitada se marca como “dead end”.

La Figura 2.12 muestra una PN con su árbol de alcanzabilidad. El marcado inicial de la red es $M_0 = (2, 1, 0)$ que es el primer nivel del árbol. La transición t_1 está habilitada, la cual al ser disparada genera el marcado $M_1 = (1, 1, 1)$, originando que ambas transiciones t_1 y t_2 estén habilitadas. Si se dispara a t_1 se obtiene el marcado $M_2 = (0, 1, 2)$ y continúa habilitada la transición t_2 , la cual al ser disparada genera el marcado $M_3 = (1, 1, 1)$ el cual se convierte en un nodo *old*, lo anterior es la rama izquierda del árbol. El ramal derecho, que se origina de disparar la transición t_2 en el segundo nivel del árbol, genera un marcado $M_4 = (2, 1, 0)$ el cual también es un nodo *old*.

Existen dos secuencias de disparo de transiciones de la PN, $\sigma = (M_0 t_1 M_1 t_1 M_2 t_2)$ y $\sigma = (M_0 t_1 M_1 t_2)$.

2.4.2. Matriz de incidencia de PN

Para representar el comportamiento dinámico de las PN, se usan las ecuaciones matriciales que describen el modelo del comportamiento de los sistemas concurrentes, de

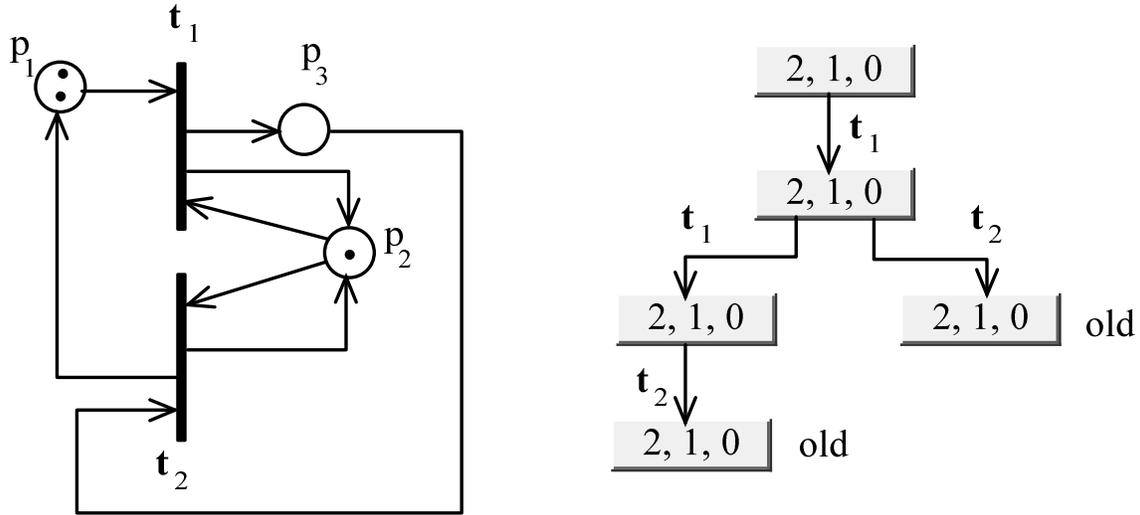


Figura 2.12: PN y su árbol de alcanzabilidad.

[56] se consideran las herramientas matemáticas de matriz de incidencia y ecuación de estado de las PN.

Matriz de incidencia de las PN:

Definición 2.2 Para una PN con n número de transiciones y m número de lugares, su matriz de incidencia $A = [a_{ij}]$ es una matriz de enteros de $n \times m$ de los pesos de los arcos de entrada y salida de la red, representada por la Ecuación matricial 2.1:

$$a_{ij} = a_{ij}^+ - a_{ij}^- \quad (2.1)$$

Donde a_{ij}^+ representa los pesos de los arcos que salen de las transiciones a lugares, y a_{ij}^- representa los pesos correspondientes de los arcos que salen de los lugares a transiciones.

La Figura 2.13 muestra una PN con cinco lugares y cuatro transiciones, y la relación de los pesos de sus arcos se muestran en la matriz correspondiente.

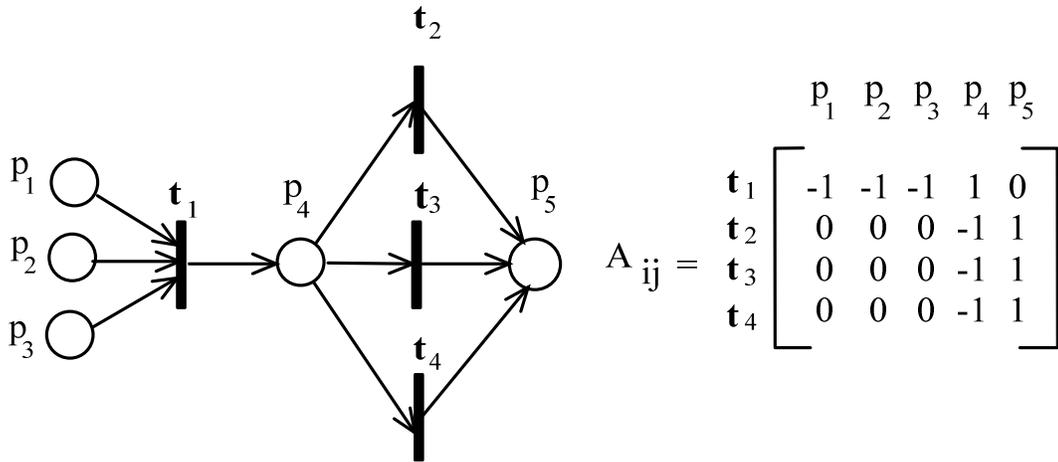


Figura 2.13: PN y su matriz de incidencia.

2.4.3. Ecuación de estado de PN

La ecuación de estado de una PN muestra el comportamiento del sistema a través de los marcados de la red que se van generando al disparar transiciones habilitadas. Un marcado M_k como un vector columna de $m \times 1$. La j -ésima entrada de este vector M_k , expresa el número de marcas en el lugar j inmediatamente después de k -ésimo disparo de alguna secuencia de disparos. El k -ésimo disparo o vector de control u_k es un vector columna de $n - 1$ 0's y una entrada distinta de cero, ahora bien, un 1 en la i -ésima posición indica que la transición i fue disparada por el k -ésimo disparo. Ya que el i -ésimo renglón de la matriz de incidencia A genera el cambio del marcado como resultado de disparar la transición i , lo cual puede ser representada por la Ecuación 2.2.

$$M_k = M_{k-1} + A^T u_k \tag{2.2}$$

A la red de la Figura 2.13 se le considera un marcado inicial $M_0 = [11100]^T$ y un vector de disparo $u_k = [1001]^T$. El comportamiento dinámico de la red se muestra en la Figura 2.14 de forma gráfica y a través de la ecuación de estado al disparar el vector u_k correspondiente.

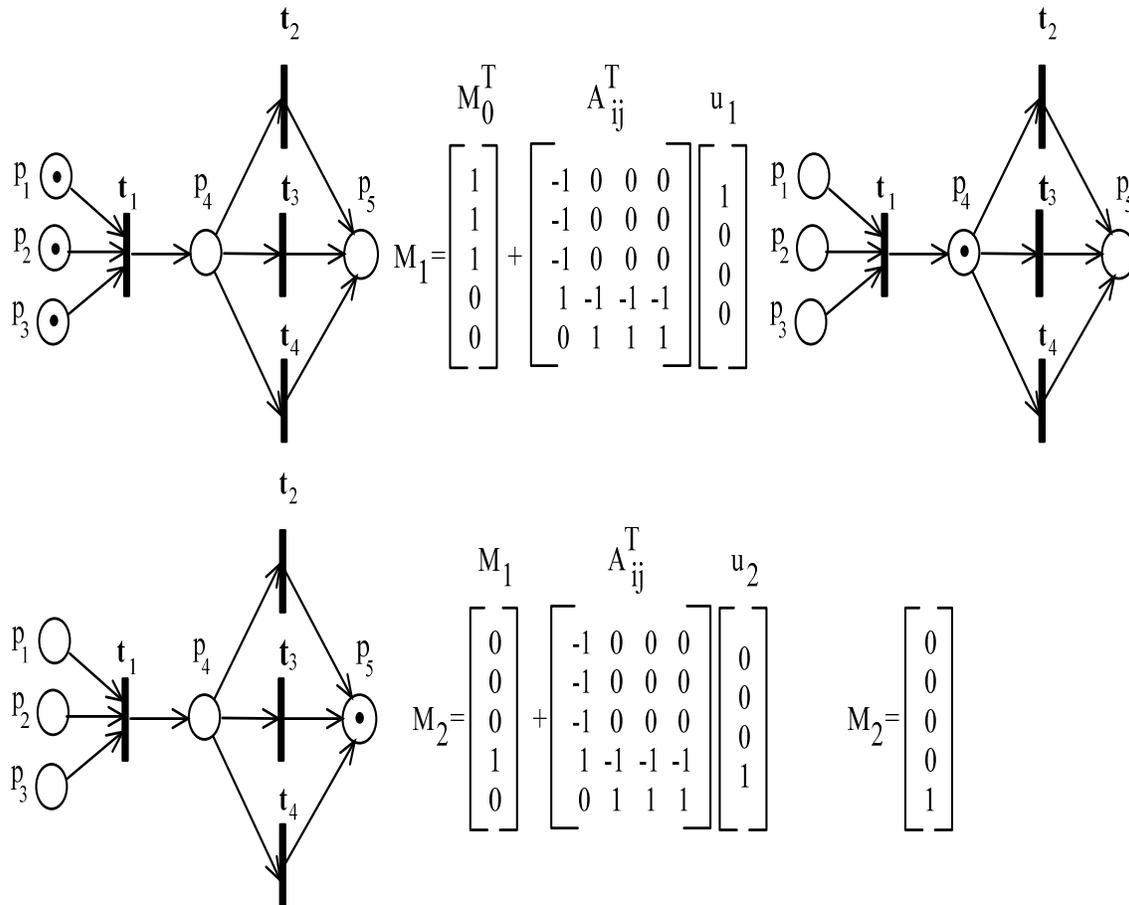


Figura 2.14: PN y su ecuación de estado.

2.5. Conclusiones del capítulo

Las PN tienen distintas formas de interpretación para los lugares y transiciones, para el presente trabajo, los lugares son señales de entrada y/o salida que representan a las señales de entradas y salidas físicas de los sistemas basados en PLC, y las transiciones como las condiciones lógicas que deben cumplir y que se forman en las líneas de control de los algoritmos en LD.

Las PN son una herramienta gráfica que permite modelar el comportamiento de los sistemas, al igual que los algoritmos de control en LD son el modelo gráfico estándar del control de un sistema, con la diferencia que los algoritmos de control en LD se

desarrollan de forma heurística y con base en la experiencia de los programadores y las PN tienen herramientas de análisis matemático que permiten formalizar los comportamientos dinámicos del sistema.

La herramienta de árbol de alcanzabilidad permite obtener los marcados posibles de una PN, los cuales pueden ser analizados para la toma de decisión de condiciones no seguras o inestables de los sistemas.

La matriz de incidencia de las PN relacionan los pesos de los arcos que salen de lugares hacia una o más transiciones y de los arcos que salen de las transiciones y van hacia uno o más lugares de la red. La matriz de incidencia es la base del comportamiento de la PN, el cual en conjunto con un marcado inicial y un vector de disparo permiten modelar y simular con ecuaciones matriciales el comportamiento dinámico de la red para su verificación o validación correspondiente, esta herramienta matemática es la ecuación de estado de las PN.

Capítulo 3

Modelado de algoritmos de control desarrollados en diagrama escalera con redes de Petri

El modelo de un sistema puede ser representado gráficamente o matemáticamente. El lenguaje de programación LD es una representación gráfica de las condiciones y acciones que deben ejecutarse en un proceso industrial. El algoritmo en LD es el modelo en control de la planta de producción o sistema de manufactura. Sin embargo, el diseño de estos algoritmos de control no son desarrollados con formalismos que permitan modelar, simular y validar su comportamiento dinámico.

Como se ha descrito anteriormente, el formalismo de las PN son una herramienta gráfica y matemática que permite modelar, simular y verificar DES al poder representar de forma explícita los estados de las variables y eventos del sistema, así como representar comportamientos de concurrencia, bloqueo, agrupación, secuencia, entre otros, de forma sencilla. Una particular interpretación de las PN es considerar a los lugares como señales de entrada y salida que viene de sensores y actuadores respectivamente, y a las transiciones como los condicionantes lógicos del algoritmo de control. Esta interpretación es la generalización de un sistema basado en PLC; señales de entrada física (sensores) - algoritmo de control (LD) - señales de salida física (actuadores).

Los sensores discretos independientemente de su tipo de salida (contacto eléctrico, transistor, SCR, opto-acoplador, entre otros) abren o cierran el circuito eléctrico hacia los módulos de entradas digitales de los PLC. Esta apertura o cierre del circuito puede relacionarse con m-número de contactos NO y/o n-número de contactos NC en el

algoritmo de control en LD. De igual forma, la señal de los módulos de salidas pueden energizar o des-energizar los actuadores independientemente de la condición inicial de estos en el proceso, es decir, un actuador puede ser NO o NC, lo cual está en función de la seguridad de los procesos en caso de falla de energía eléctrica o neumática por ejemplo.

En previos trabajos, solo es considerado de forma general el comportamiento de energizar una o más bobinas en el algoritmo de control, lo que implica, que el sistema a controlar no tiene actuadores con condición inicial de energizado, lo cual, ocurre en sistemas con pocas variables. Sin embargo, el desarrollo del algoritmo de control esta en función del sistema a controlar, y puede ser necesario energizar - des-energizar y/o des-energizar - energizar bobinas en algún momento del proceso de control. Además, no todos los trabajos hasta nuestro conocimiento consideran lógicas de control como: auto-lazo y bloqueos entre líneas de control (conocido como “interlocking” en la industria). El presente trabajo propone como base una metodología de transformación de algoritmos de control desarrollados en LD a PN de tipo ordinaria que considere el comportamiento dinámico de energizar y/o des-energizar bobinas, y en consecuencia activar o desactivar contactos, además de modelar y simular la red obtenida para validar los algoritmos de control en condiciones de fallas físicas reales de corto-circuito y/o circuito-abierto.

3.1. Distribución de señales de LD en términos de PN

Como se ha mencionado en párrafos anteriormente, una señal de entrada y/o salida física, o de memoria del PLC puede tener dos estados. Estos estados en el algoritmo de control en LD se ven reflejados a través de los contactos NO y NC o de la bobina de la señal correspondiente. La Figura 3.1 muestra tres líneas de control en LD, con la intención de mostrar los tres tipos de señales. La variable *Sensor* es una señal de entrada física, y esta puede tener contactos NO y/o NC en diversas líneas de control. (Nota: las señales de entrada física no pueden representarse como bobina en los LD). La variable *Actuador* es una señal de salida física representada por una bobina, la cual puede tener contactos NO y/o NC en diferentes líneas de control. Por último, la variable *Memoria* es una señal interna del PLC, y esta puede ser representada por una bobina y contactos NO y/o NC. En general, solo puede haber una variable para una bobina, la cual puede tener m-número de contactos NO y/o n-números de contactos NC en al algoritmo de control en LD.



Figura 3.1: Señales en LD

Una señal de entrada física I en LD puede tener contactos NO o NC o de ambos tipos. Sin embargo, las señales de salida física O y de memoria B pueden o no tener contactos en el algoritmo, si no estar solo como bobina, aunque la excepción implica algoritmos de control dependientes de las señales de entrada física y de procesos de baja complejidad.

En general, una señal discreta en LD puede ser representada en elementos de PN, utilizando los dos tipos de arcos: habilitador e inhibidor. Considerando lógica positiva en el diseño de los algoritmos, una marca en el lugar p habilita la transición correspondiente a los contactos NO, y la ausencia de marca habilita la transición de los contactos NC.

Con base en el análisis de los tipos de señales en los LD y sus respectivos contactos y su comportamiento respectivo, se propone una distribución de estas señales en elementos de PN. La Figura 3.2 muestra la propuesta para a) señal de entrada física I , b) señal de salida física O y c) para señal interna de memoria B . Donde las Ecuaciones 3.1a - 3.1f muestran la correspondencia con el número de contactos y de su tipo de las señales utilizadas en los algoritmos de control en LD y los lugares en elementos de PN respectivamente.

La distribución de señales es ordinaria, es decir, los pesos de todos los arcos es unitario y cada lugar solo puede tener una marca. En cuanto al marcado de la red, posteriormente se analiza el problema de acumulación de marcas. Para describir el comportamiento dinámico de la distribución de señales, el cual está en función de los

arcos habilitador e inhibidor, si los lugares de entrada I_i , O_o y B_b tienen marca, entonces se habilita la transición \mathbf{I}_i^o , \mathbf{O}_o^o y \mathbf{B}_b^o equivalente a activar los contactos NO de la señal en LD correspondiente. De igual forma, si los lugares de entrada I_i , O_o y B_b no tienen marca, entonces se habilita la transición \mathbf{I}_i^c , \mathbf{O}_o^c y \mathbf{B}_b^c equivalente a permitir el flujo de energía por los contactos NC de la señal en LD correspondiente. Las Ecuaciones 3.2a - 3.2c modelan este comportamiento respectivamente para cada señal.

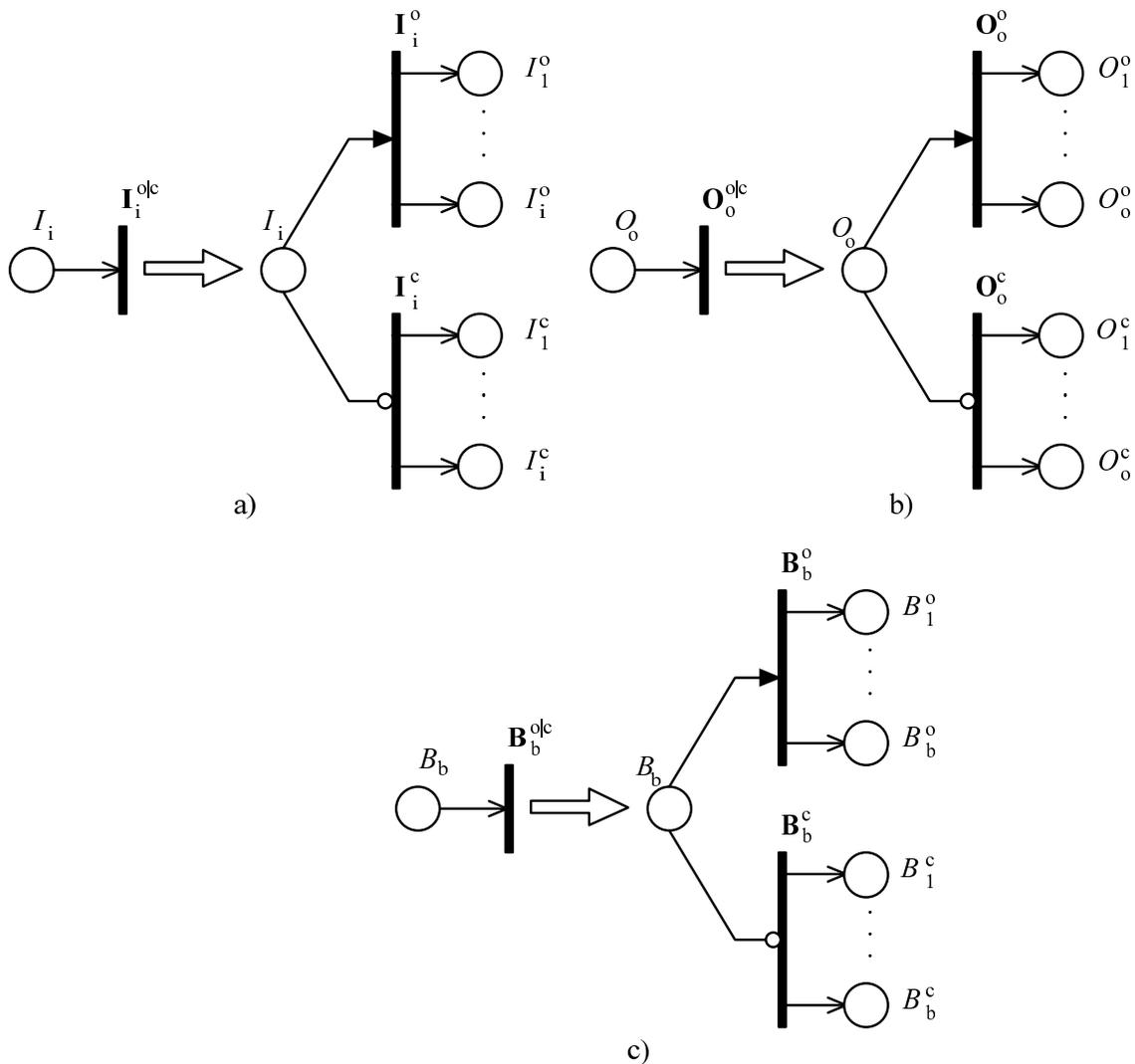


Figura 3.2: Distribución de señales de LD en elementos de PN.

$$I_i^o = \# \text{ contactos NO de señales de entrada física} \quad (3.1a)$$

$$I_i^c = \# \text{ contactos NC de señales de entrada física} \quad (3.1b)$$

$$O_o^o = \# \text{ contactos NO de señales de salida física} \quad (3.1c)$$

$$O_o^c = \# \text{ contactos NC de señales de salida física} \quad (3.1d)$$

$$B_b^o = \# \text{ contactos NO de señales de memoria} \quad (3.1e)$$

$$B_b^c = \# \text{ contactos NC de señales de memoria} \quad (3.1f)$$

$$M(I_i) = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{ entonces } \left\{ \begin{array}{l} M(I_i^o) = 0 \text{ y } M(I_i^c) = 1 \\ M(I_i^o) = 1 \text{ y } M(I_i^c) = 0 \end{array} \right\} \quad (3.2a)$$

$$M(O_o) = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{ entonces } \left\{ \begin{array}{l} M(O_o^o) = 0 \text{ y } M(O_o^c) = 1 \\ M(O_o^o) = 1 \text{ y } M(O_o^c) = 0 \end{array} \right\} \quad (3.2b)$$

$$M(B_b) = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{ entonces } \left\{ \begin{array}{l} M(B_b^o) = 0 \text{ y } M(B_b^c) = 1 \\ M(B_b^o) = 1 \text{ y } M(B_b^c) = 0 \end{array} \right\} \quad (3.2c)$$

Para la parte gráfica de la PN propuesta, se omite colocar transiciones sumidero para consumir las marcas que pudieran estar en los lugares de salida de la distribución de señales.

El ciclo de trabajo de los sistemas basados en PLC actualiza el estado de las señales discretas en tiempo real, lo que implica el problema de acumulación de marcas en lugares de una PN. En la siguiente sección se proponen restricciones a los marcados de los lugares para evitar su acumulación.

Entonces, modelar algoritmos de control desarrollados en LD debe considerar diversos aspectos; estructurales en cuanto a los elementos núcleos que utiliza el lenguaje de

programación LD, comportamiento dinámico en cuanto a elementos activos (contactos por donde fluye energía) o desactivados (contactos por donde no fluye energía) en un ciclo de trabajo, el problema de acumulación de marcas en los lugares de la red, energizado y des-energizado de bobinas, así como funciones de control lógico: AND, OR, AND-OR, Set-Reset, bloqueo y autolazo, siendo las utilizadas con mayor frecuencia en los algoritmos de control en LD.

En la siguiente sección se analiza el problema de acumulación de marcas debido a que el funcionamiento de los sistemas basados en PLC son cíclicos.

3.2. Problema de acumulación de marcas

Los sistemas basados en PLC basan su funcionamiento en el ciclo de trabajo, el cual tiene la particularidad de evaluar el algoritmo de control con una copia de los estados de las señales de entrada física, entonces si una señal se activa o desactiva, su estado se reflejara hasta la siguiente acción de lectura de las entradas para ser considerado en la evaluación del algoritmo de control.

Si una señal activa en LD no cambia su estado en varios ciclos de trabajo, entonces, en la teoría de PN ocasiona que en cada ciclo el lugar correspondiente recibiría una marca, acumulándose en cada barrido, lo cual es un problema para modelar el comportamiento dinámico de la red.

Considerando la nomenclatura de [56], para el marcado los lugares de entrada y salida, se considera lo siguiente:

$\bullet t = \{p : (p, t) \in F\}$, Conjunto de lugares de entrada de t .

$t \bullet = \{p : (t, p) \in F\}$, Conjunto de lugares de salida de t .

El problema de acumulación de marcas en lugares de PN transformadas desde algoritmos de control basados en PLC se ha propuesto resolverlo principalmente en dos formas. Los autores de [3] y [37] consideran un vector en la PN de los estados de las señales de entrada, con el cual evalúan ecuaciones booleanas para actualizar un vector de salidas, entonces cada vez que se desea evaluar la PN correspondiente se actualiza el vector de los estados de las señales de entrada, es importante mencionar que la red es

de tipo ordinaria y por consiguiente la actualización es en colocar o mantener marca en un lugar. La segunda forma propuesta, es usar arco inhibidor del lugar de salida hacia su transición que la habilita para que una vez que el lugar de salida tenga una marca, entonces inhabilita a la transición y por ende la acumulación de marcas [4].

La propuesta de la presente investigación para el problema de acumulación de marcas, es colocar funciones lógicas y aritméticas para habilitar el marcado de los lugares O_o y B_b de la PN. Las Ecuaciones 3.3a y 3.3b son para habilitar el marcado del lugar de salida O_o y B_b respectivamente que tienen como entrada un arreglo AND. Las Ecuaciones 3.4a y 3.4b son para cuando la estructura de los lugares de entrada deben aplicar una lógica OR. Para cuando la estructura de los lugares de entrada son una combinación de lógica AND y OR, las Ecuaciones 3.5a y 3.5b habilitan el marcado correspondiente de los lugares de salida O_o y B_b .

$$O(t\bullet)_{AND} = \prod M(\bullet t) = 1 \text{ AND } O(t\bullet) = 0 \quad (3.3a)$$

$$B(t\bullet)_{AND} = \prod M(\bullet t) = 1 \text{ AND } B(t\bullet) = 0 \quad (3.3b)$$

$$O(t\bullet)_{OR} = \sum M(\bullet t) = 1 \text{ AND } O(t\bullet) = 0 \quad (3.4a)$$

$$B(t\bullet)_{OR} = \sum M(\bullet t) = 1 \text{ AND } B(t\bullet) = 0 \quad (3.4b)$$

$$O(t)_{ANDOR} = \sum \left(\prod (M(\bullet t)_{L1} = 1), \dots, \left(\prod (M(\bullet t)_{Li} = 1) \right) = 1 \text{ AND } O(t\bullet) = 0 \quad (3.5a)$$

$$B(t)_{ANDOR} = \sum \left(\prod (M(\bullet t)_{L1} = 1), \dots, \left(\prod (M(\bullet t)_{Li} = 1) \right) = 1 \text{ AND } B(t\bullet) = 0 \quad (3.5b)$$

3.3. Problema de restablecimiento de lugares

Las Ecuaciones 3.2a - 3.2b modelan el comportamiento para establecer el marcado en los lugares que corresponden a los contactos NO y/o NC de cada una de las señales utilizadas en los algoritmos de control en LD.

Para los lugares de salida O_o y B_b se considera el marcado de los lugares de entrada y del propio lugar de salida, así como el tipo de lógica de la red. Las Ecuaciones 3.6a y 3.6b son para restablecer lugares de salida O_o y B_b respectivamente para una estructura AND. Las Ecuaciones 3.7a y 3.7b para una lógico OR y las Ecuaciones 3.8a y 3.8b para una estructura con lógica AND y OR.

$$G(L\bullet)_{AND} = \prod M(\bullet t) = 0 \text{ AND } O(t\bullet) = 1 \quad (3.6a)$$

$$G(t\bullet)_{AND} = \prod M(\bullet t) = 1 \text{ AND } B(t\bullet) = 0 \quad (3.6b)$$

$$G(t\bullet)_{OR} = \sum M(\bullet t) = 0 \text{ AND } O(t\bullet) = 1 \quad (3.7a)$$

$$G(t\bullet)_{OR} = \sum M(\bullet t) = 0 \text{ AND } B(t\bullet) = 1 \quad (3.7b)$$

$$G(t)_{ANDOR} = \sum \left(\prod (M(\bullet t)_{L1} = 1), \dots, \left(\prod (M(\bullet t)_{Li} = 1) \right) = 0 \text{ AND } O(t\bullet) = 1 \right) \quad (3.8a)$$

$$G(t)_{ANDOR} = \sum \left(\prod (M(\bullet t)_{L1} = 1), \dots, \left(\prod (M(\bullet t)_{Li} = 1) \right) = 0 \text{ AND } B(t\bullet) = 1 \right) \quad (3.8b)$$

En cada transformación propuesta de las líneas de control en LD a elementos de PN y de algunos comportamientos característicos presentes en los DES se indican los

pormenores del marcado de la red respectiva, así como la ecuación de restablecimiento de los lugares de salida [46], [54].

3.4. Estructuras en PN de líneas de control en LD

Se analizan y se describen las prerrogativas de comportamiento de las líneas de control utilizadas en los algoritmos de control, así como comportamientos característicos que tienen los DES en estructuras de PN y que al igual que en el LD serán base para la propuesta de transformación de algoritmos de control en LD a PN.

De forma general se utilizan lugares que representan señales de entrada con contactos abiertos I_i^o , sin embargo, como se ha comentado, a una transición puede tener señales de entrada de las otras dos señales O_o^{olc} y B_b^{olc} .

A) Estructura PN de la lógica AND:

La Figura 3.3 muestra la estructura en PN de la lógica AND. Solo cuando todos sus lugares de entrada tienen marca, entonces se habilita la transición \mathbf{L}_1 . Es decir, si $I_1^o, I_2^o, \dots, I_i^o = 1$, y la transición \mathbf{L}_1 es disparada se coloca una marca en lugar de salida O_1 . Este comportamiento es equivalente a la evaluación de la línea de control AND en LD en un ciclo donde es energizada la bobina de salida correspondiente $OUT1$ de la Figura 1.6.

Si se considera que en la línea de control en LD no hubo cambio en los estados de los contactos en el siguiente ciclo de trabajo del PLC, por consiguiente la bobina sigue energizada, es necesario limitar el comportamiento de la red PN propuesta, ya que los lugares de entrada $I_1^o, I_2^o, \dots, I_i^o$ tendrán marca nuevamente y habilitan de nuevo a la transición \mathbf{L}_1 , la cual si se dispara colocaría otra marca en el lugar de salida O_1 , lo cual no es el comportamiento de la línea de control de contactos en serie en LD.

La Ecuación 3.6a es para habilitar el marcado en el lugar de salida de la estructura PN de la lógica AND, implica que el marcado del lugar de salida de una transición puede ser habilitada si cumple con que todos los lugares de entrada tienen marca y

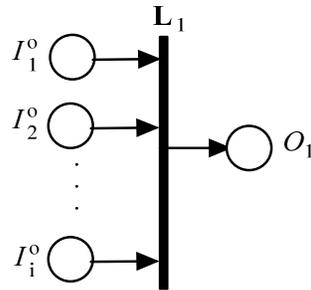


Figura 3.3: Estructura PN de la lógica AND.

el lugar de salida no tienen marca, equivalente al comportamiento lógico AND. La transición L_1 es habilitada si y solo si los lugares de entrada $I_1^o = I_2^o = \dots = I_i^o = 1$ tienen marca y el lugar de salida $O_1 = 0$ no tiene marca. La Figura 3.4 muestra el marcado de la red con la; a) transición L_1 habilitada, b) el disparo de la misma y c) resolviendo el problema de acumulación de marcas.

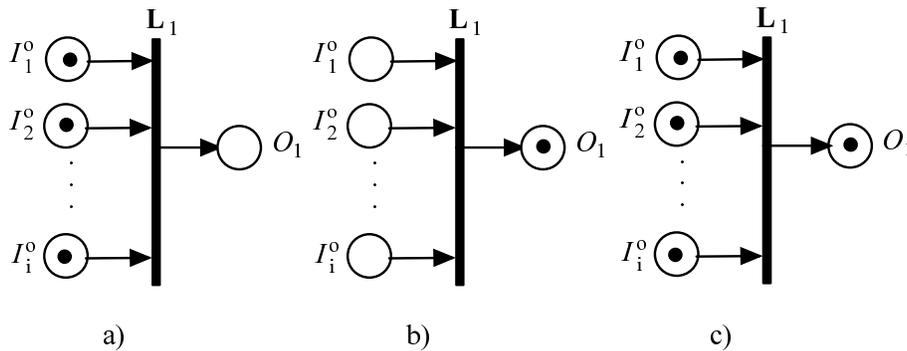


Figura 3.4: Marcado en estructura PN de lógica AND

Consumir la marca del lugar de salida O_1 de la estructura AND, es equivalente a des-energizar la bobina de la línea de control en LD, la cual ocurre cuando uno o más de los contactos en serie que la energizan cambia del estado activado a desactivado o viceversa. Para resolver lo anterior, y obtener el comportamiento de des-energizado de una bobina se propone un lugar de restablecimiento $G(L_1)_{AND}$ con un arco habilitador hacia una transición R_1 , la cual también recibe un arco habilitador desde el lugar de salida O_1 , como muestra la Figura 3.5, donde el lugar de restablecimiento G está en función del marcado de los lugares de entrada $M(\bullet t) = 1$ principalmente, sin embargo, para modelar el comportamiento de des-energizar la bobina, primero

debe estar energizada, lo que implica que el lugar de salida O_1 debe tener marca, $M(t \bullet) = 1$, correspondiente a la Ecuación 3.6a.

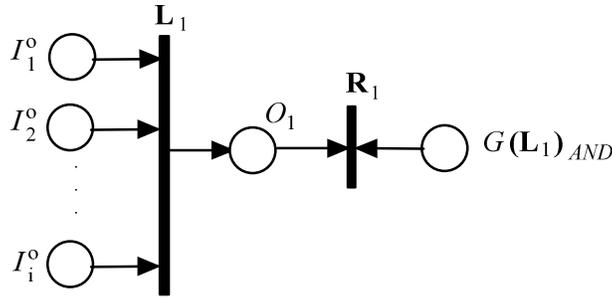


Figura 3.5: Estructura PN de lógica AND con restablecimiento.

La Figura 3.6 muestra el comportamiento dinámico de la estructura PN para la lógica AND. a) marca en los lugares de entrada $I_1^o = I_2^o = \dots = I_i^o = 1$ y transición \mathbf{L}_1 habilitada, b) disparo de la transición \mathbf{L}_1 y marca en el lugar de salida O_1 , c) actualización del marcado de los lugares de entrada y \mathbf{L}_1 inhabilitada por la Ecuación 3.3a, c) cambio de estado de la señal de entrada I_2^o en función de la Ecuación 3.2a y marcado del lugar $G(\mathbf{L}_1)_{AND}$ en función de la Ecuación 3.6a lo que habilita la transición \mathbf{R}_1) y d) disparo de la transición \mathbf{R}_1 , lo que implica el restablecimiento del lugar de salida O_1 .

B) Estructura PN de la lógica OR:

La estructura de PN para la lógica OR se muestra en la Figura 3.7. Si uno o más de los lugares de entrada tiene marca, entonces se habilitan sus respectivas transiciones, es decir, $I_1^o + I_2^o + \dots + I_i^o = 1$. Si $I_1^o + I_2^o = 1$, entonces las transiciones \mathbf{L}_1 y \mathbf{L}_2 están habilitadas, el disparo de cualquiera de estas coloca una marca en el lugar de salida O_1 , comportamiento equivalente a la línea de control de contactos en paralelo de la Figura 1.7, sin embargo, si se disparan ambas transiciones colocarían dos marcas en el lugar de salida O_1 , lo cual no es el comportamiento deseado.

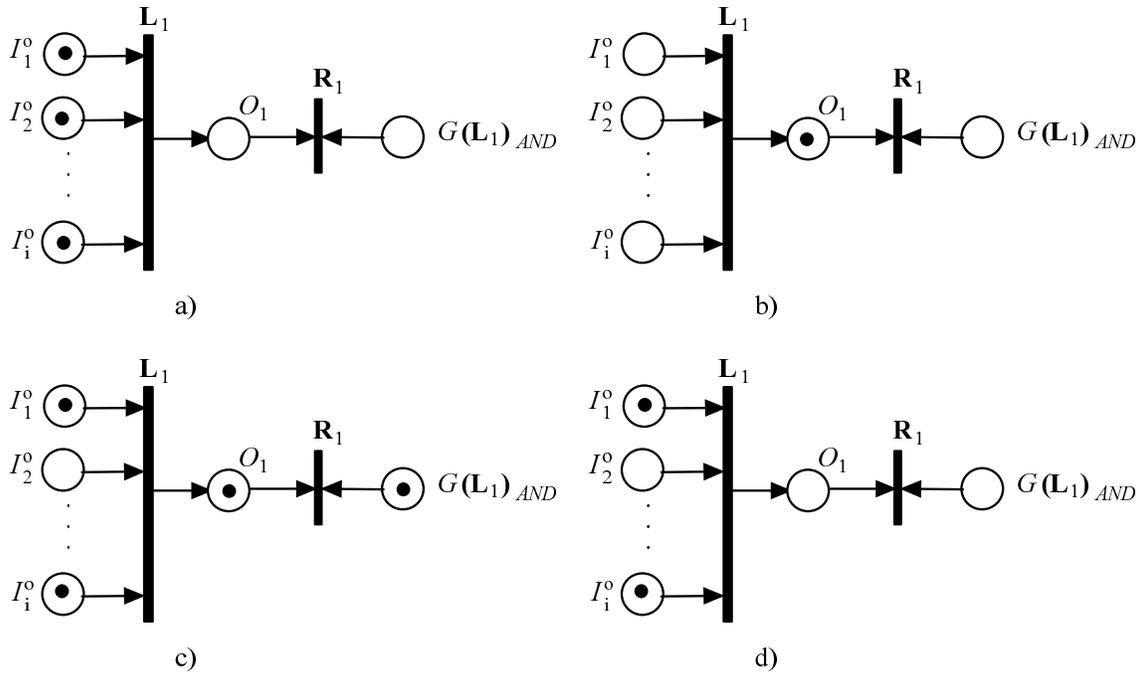


Figura 3.6: Comportamiento dinámico de la estructura PN de la lógica AND.

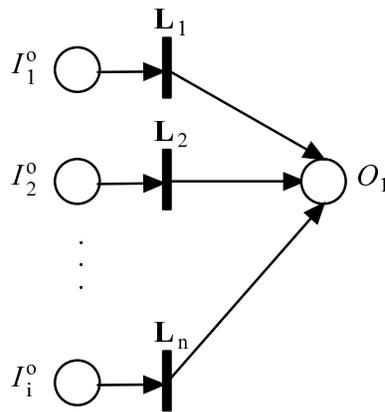


Figura 3.7: Estructura PN de la lógica OR.

La Ecuación 3.4a resuelve el problema de acumulación de marcas para la estructura PN de la lógica OR. implica que para poder ser habilitada una de las transiciones, la suma del marcado de los lugares de entrada $M(\bullet t) = 1$ y que el marcado del lugar de salida $M(t\bullet) = 0$, es decir, la primer transición que sea disparada coloca marca

en el lugar de salida O_1 y deshabilita el disparo de las otras transiciones. La Figura 3.8 muestra este comportamiento.

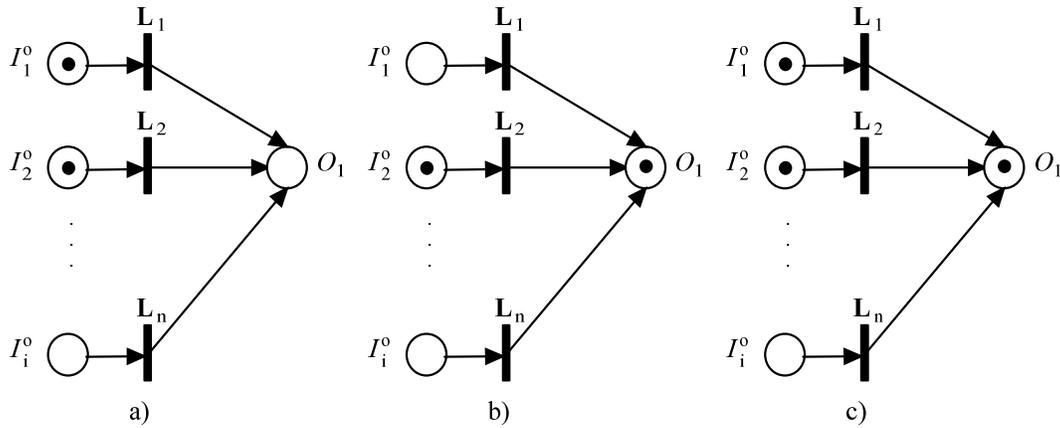


Figura 3.8: Marcado de la estructura PN de la lógica OR.

De igual forma se tiene el problema de restablecer el lugar de salida O_1 equivalente a des-energizar la bobina de una línea de control de contactos en paralelo. La Figura 3.9 muestra la estructura PN propuesta, donde el lugar de restablecimiento $G(\mathbf{L}_1, \mathbf{L}_2, \dots, \mathbf{L}_n)_{OR}$ esta en función del marcado de los lugares de la red, correspondiente a la Ecuación 3.7a.

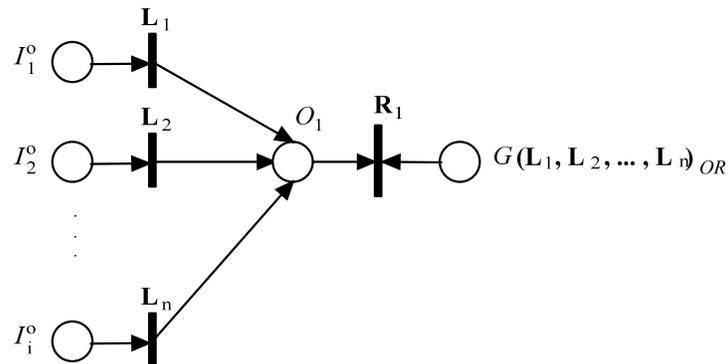


Figura 3.9: Estructura PN de la lógica OR con restablecimiento.

La Figura 3.10 muestra el comportamiento dinámico de la estructura PN de la lógica OR. a) marca en los lugares de entrada $I_1^o = I_n^o = 1$, por consiguiente transición \mathbf{L}_1 y \mathbf{L}_n habilitadas, b) disparo de la transición habilitada \mathbf{L}_1 y colocación de marca en el lugar de salida O_1 , c) actualización del marcado de los lugares de entrada y transiciones \mathbf{L}_1 y \mathbf{L}_n inhabilitadas por la Ecuación 3.3a, d) lugares de entrada $I_1^o = I_2^o = I_n^o = 0$ sin marca, lo cual coloca marca en el lugar de restablecimiento $G(\mathbf{L}_1, \mathbf{L}_2, \dots, \mathbf{L}_n)_{OR}$ habilitando la transición \mathbf{R}_1 y consumiendo la marca del lugar de salida O_1 .

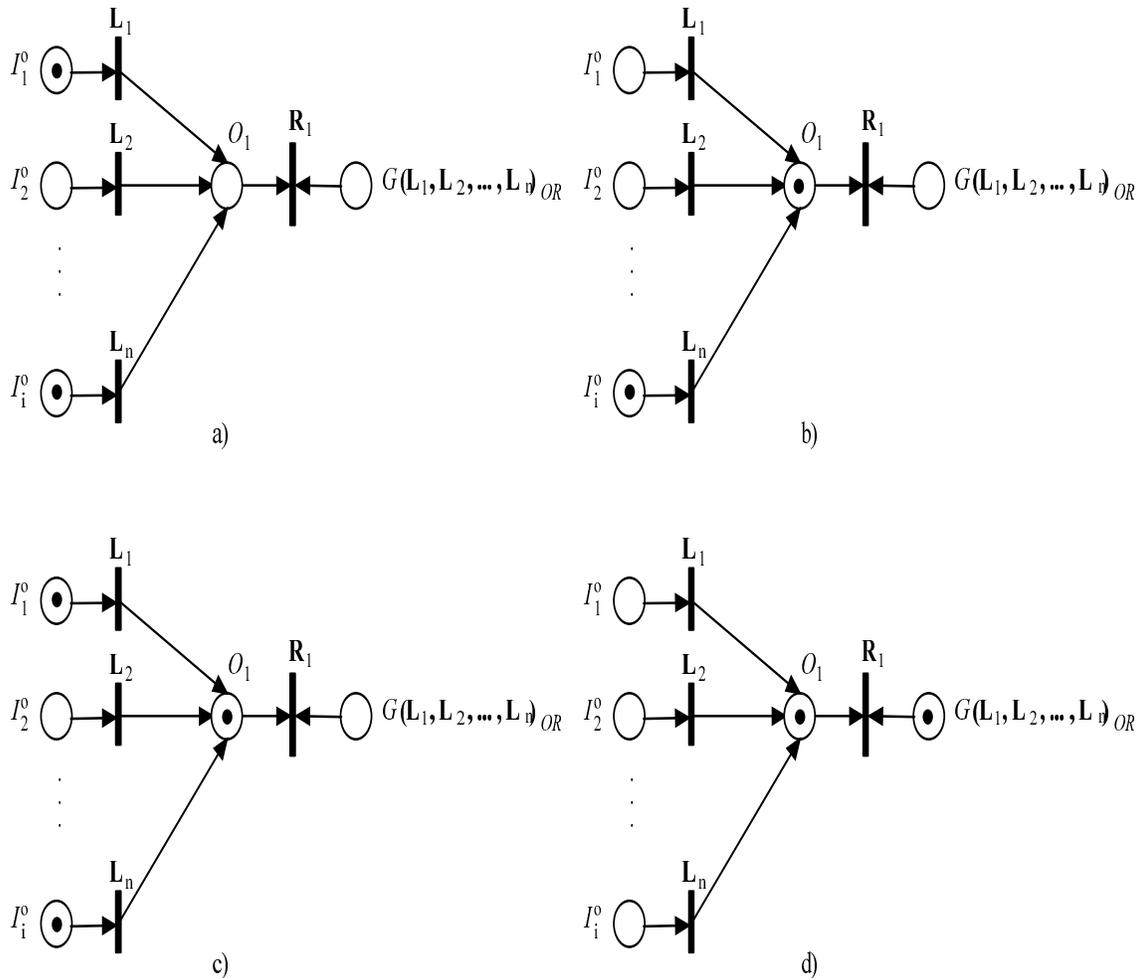


Figura 3.10: Comportamiento dinámico de la estructura PN de la lógica OR.

C) Estructura PN de la lógica AND-OR:

La estructura PN con ambas lógicas *AND* and *OR* se muestra en la Figura 3.11. El lugar de salida O_1 puede recibir marca de las transiciones \mathbf{L}_1^o o \mathbf{L}_2^o , las cuales están en función de las marcas de sus lugares de entrada $I_1^o = I_2^o, \dots, = I_i^o = 1$ y/o $B_1^o = B_2^o, \dots, = B_b^o = 1$ respectivamente. Considerando la Ecuación 3.5a para resolver el problema de acumulación de marcas, se tiene que la transición que sea habilitada y disparada primero \mathbf{L}_1^o o \mathbf{L}_2^o , coloca una marca en el lugar de salida O_1 , y cuando se hace la actualización de marcas en los lugares de entrada, las transiciones están inhabilitadas.

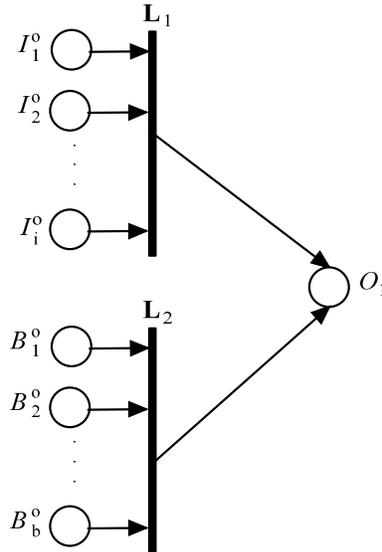


Figura 3.11: Estructura PN de ambas lógicas AND and OR.

De igual forma, para para consumir la marca del lugar de salida O_1 , el lugar de restablecimiento $G(\mathbf{L}_1, \mathbf{L}_2)_{ANDOR}$ esta en función de la Ecuación 3.8a y la red se muestra en la Figura 3.12.

Es decir, se requiere que cada transición hacia el lugar de salida O_1 este inhabilitada, y que el lugar de salida tenga marca para colocar marca en el lugar de restablecimiento $G(\mathbf{L}_1, \mathbf{L}_2), \dots, \mathbf{L}_l)_{ANDOR}$, habilitar la transición \mathbf{R}_1 , la cual al dispararla se consume la marca del lugar de salida O_1 .

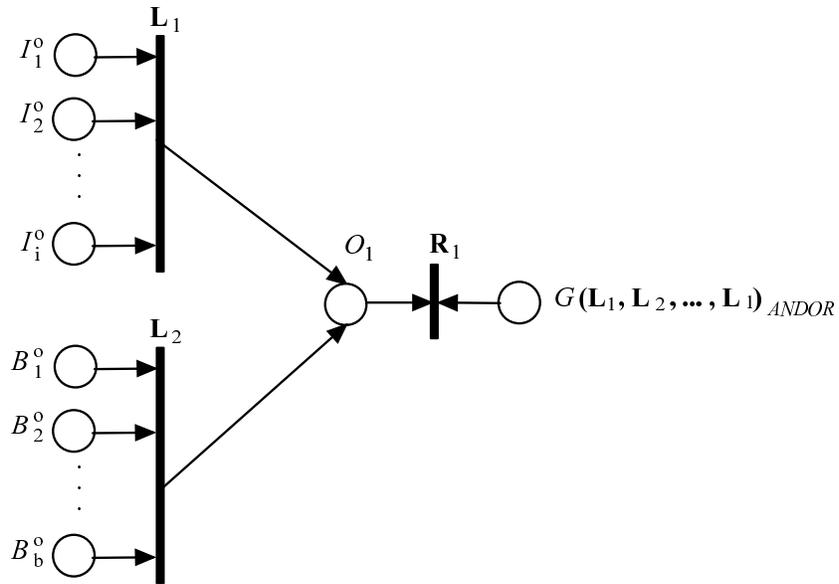


Figura 3.12: Estructura PN de la lógica AND y OR con lugar de restablecimiento.

La Figura 3.13 muestra el comportamiento dinámico de la estructura PN que contiene ambas lógicas AND y OR. a) marcado de los lugares de entrada con transición L_2 habilitada, b) disparo de la transición L_2 y colocación de marca en el lugar de salida O_1 , c) actualización del marcado e inhabilita la transición L_2 por la Ecuación 3.5a, d) marcado en el lugar de restablecimiento por la Ecuación 3.8b.

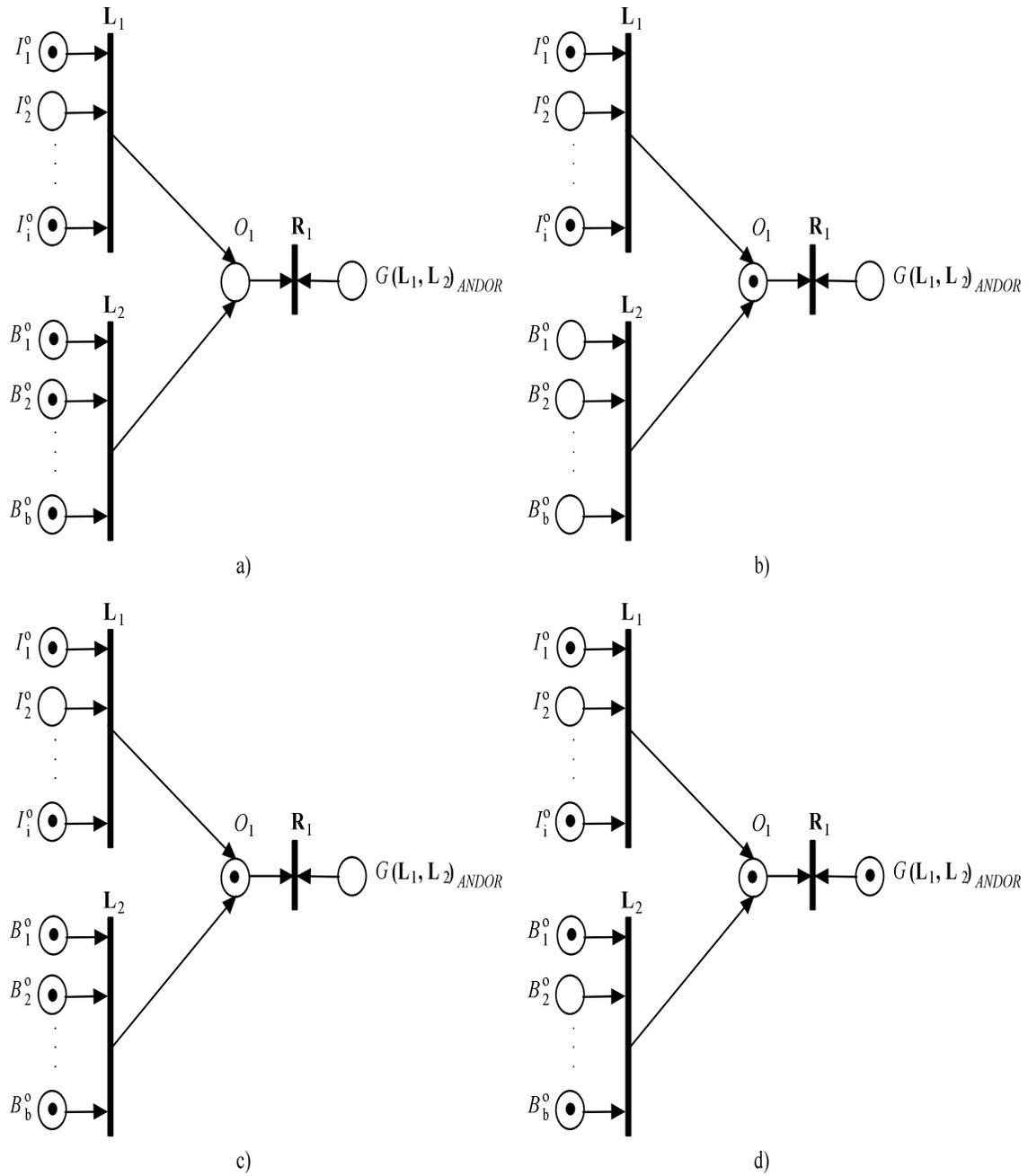


Figura 3.13: Comportamiento dinámico de la estructura PN de la lógica AND y OR.

D) Estructura PN de la lógica Set - Reset:

La lógica Set - Reset son señales asíncronas de control con prioridad en la señal Reset, es decir, la posibilidad de que ambas señales Set y Reset estén presentes en un mismo momento, implica que la señal de salida es cero. La estructura PN propuesta para la lógica Set - Reset se muestra en la Figura 3.14. Los lugares de entrada a las transiciones \mathbf{L}_1 y \mathbf{R}_1 pueden ser solo un lugar o una de las estructuras PN de las configuraciones AND y/o OR anteriormente descritas.

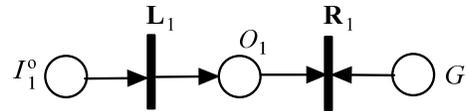


Figura 3.14: Estructura PN de la lógica Set - Reset.

Una marca en el lugar I_1^o habilita la transición \mathbf{L}_1 , la cual al ser disparada coloca una marca en el lugar de salida O_1 , y por la Ecuación 3.5a se inhabilita a la transición \mathbf{L}_1 (independientemente si es una estructura PN de lógica AND, OR o ANDOR), además que su actualización del marcado no afecta, es decir, la marca en el lugar de salida O_1 se mantiene ya que la función del lugar de restablecimiento G es independiente del marcado de los lugares de entrada a la transición \mathbf{L}_1 .

El marcado de lugar de restablecimiento G está en función de la estructura PN correspondiente (AND, OR o ANDOR), por lo tanto, las Ecuaciones 3.6a o 3.7a o 3.8a describen su comportamiento.

La Figura 3.15 muestra el comportamiento dinámico de la estructura PN de la lógica Set - Reset. a) marca en el lugar de entrada I_1^o y transición \mathbf{L}_1 , b) disparo de la transición \mathbf{L}_1 y marcado en el lugar de salida O_1 , c) actualización del marcado del lugar de entrada I_1^o (si el lugar tiene marca, entonces por la Ecuación 3.5a se evita la acumulación de marcas en el lugar de salida O_1 . Si el lugar no tiene marca, la transición \mathbf{L}_1 está inhabilitada), d) marca en el lugar de salida O_1 lo que posibilita el marcado en el lugar de restablecimiento del lugar G por la Ecuación 3.6a para este ejemplo.

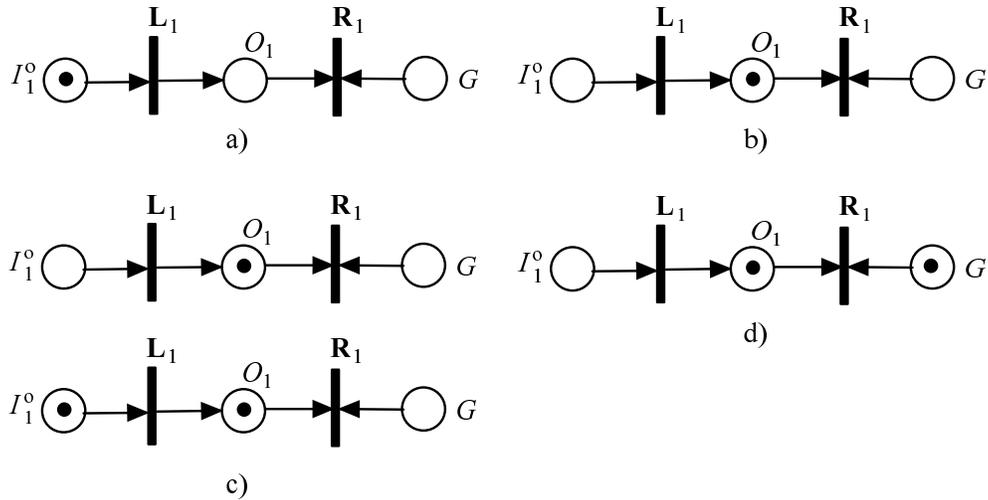


Figura 3.15: Comportamiento dinámico de la estructura PN de la lógica Set - Reset.

E) Estructura PN de la lógica de bloqueo - *interlocking*:

La estructura PN de lógica de bloqueo, permite separar acciones de actividades en paralelo, solo permitiendo la terminación de una (la que primero cumpla con las condiciones), para posteriormente, permitir la terminación de la segunda (si cumple con las condiciones), mientras la primera continua con su proceso, así sucesivamente en operación.

La estructura PN de la Figura 3.16 muestra la estructura PN de la lógica de bloqueo o *interlocking* que puede darse entre dos o más transiciones. Si los lugares de salida O_1 y O_2 no tienen marca, entonces las transiciones O_1^c y O_2^c están habilitadas por sus arcos inhibidores respectivamente, los cuales al ser disparadas colocan marca en los lugares O_1^o y O_2^o .

Si todos los lugares de entrada de la transición L_1 tienen marca, es decir, $I_1^o = O_2^c = \dots = I_i^o = 1$, entonces está habilitada dicha transición, al dispararse coloca una marca en el lugar de salida O_1 , lo que implica que la transición O_1^c sea inhabilitada, y por la Ecuación 3.2b se drena la marca del lugar de entrada O_1^c de la transición L_2 , por lo consiguiente no puede ser habilitada. De igual forma, si la transición L_2 fuera la habilitada y disparada (sin marca en el lugar de salida), entonces se drena

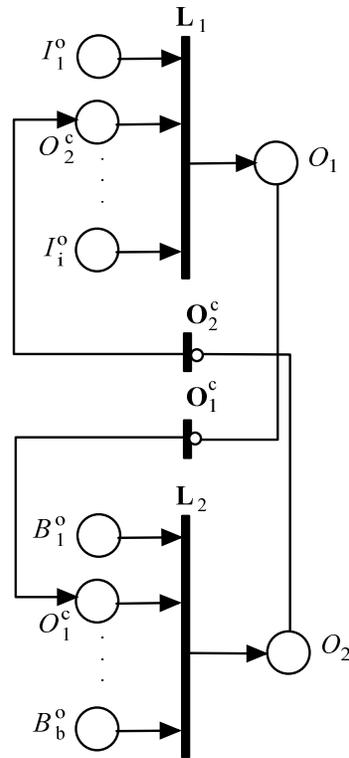


Figura 3.16: Estructura PN de la lógica de bloqueo.

el lugar de entrada O_2^c , lo cual inhabilita a la transición L_1 .

De igual forma, el problema de acumulación de marcas en alguno de los lugares de salida O_1 y O_2 se resuelve con la Ecuación 3.3a, ya que el marcado de los lugares de restablecimiento $G(L_1)$ y $G(L_2)$ están en función del marcado de los lugares de entrada a las transiciones L_1 y L_2 respectivamente.

Dejar solo habilitada una transición L_1 o L_2 , es decir, que uno de los lugares O_1^c o O_2^c no puedan tener marca, es equivalente a dejar parte del funcionamiento de un sistema o máquina fuera de servicio en los algoritmos de control en LD.

Para el comportamiento de consumir la marca en los lugares de salida, se proponen la estructura de la Figura 3.17, donde cada lugar de salida tiene su lugar de restablecimiento en función del marcado de los lugares de entrada a dicha transición,

Ecuación 3.7a.

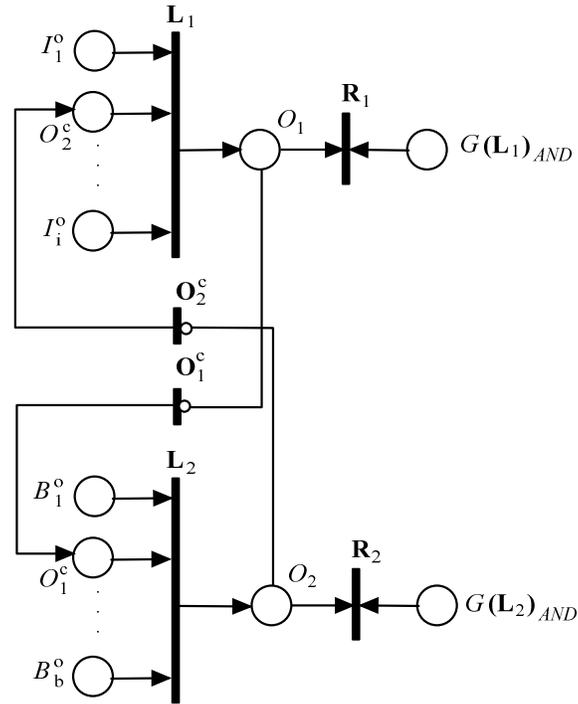


Figura 3.17: Estructura PN de la lógica de bloqueo con lugares de restablecimiento.

La Figura 3.18 y 3.19, muestran el comportamiento dinámico de la estructura PN de la lógica de bloqueo. a) Lugares de entrada $I_1^o = O_2^c = \dots = I_i^o = 1$ habilitando la transición L_1 y con marca en el lugar O_1^c por arco inhibidor, b) disparo de la transición L_1 , colocación de marca en el lugar de salida O_1 , deshabilitado de la transición O_2^o y drenado de la marca del lugar O_1^c por Ecuación 3.2b, c) actualización de marcado de la estructura PN, transición L_1 inhabilitada por Ecuación 3.3a, d) marca en lugar de restablecimiento $G(L_1)_{AND}$ por Ecuación 3.7a.

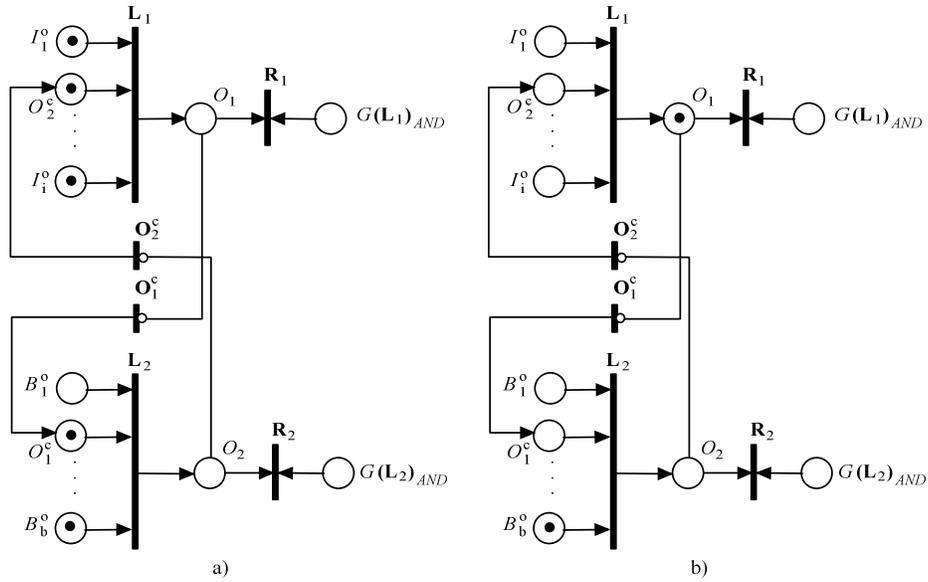


Figura 3.18: Comportamiento dinámico de la estructura PN de la lógica de bloqueo.

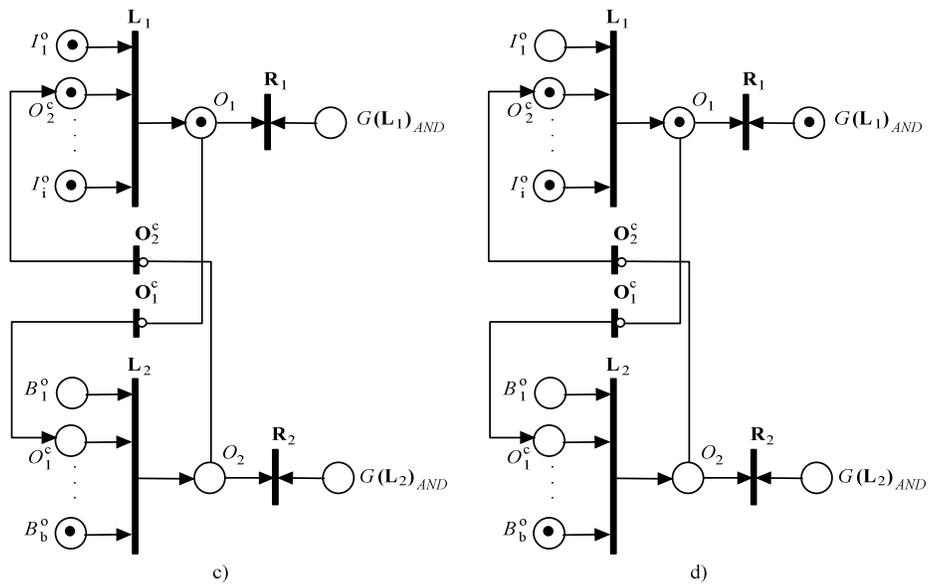


Figura 3.19: Comportamiento dinámico de la estructura PN de la lógica de bloqueo.

F) Estructura PN del comportamiento de ejecución secuencial:

En PN la ejecución secuencial se generalizan en una secuencia determinada constante de habilitado y disparo de transiciones, que permite el flujo de una o más marcas a través de los lugares. La Figura 3.20 muestra la estructura PN para el comportamiento de ejecución secuencial. Si el lugar de entrada I_1^o habilita la transición \mathbf{L}_1 , la cual al dispararse consume la marca del lugar I_1^o y coloca en el lugar de salida B_1 , habilitando la transición \mathbf{L}_2 , que si es disparada, consume la marca del lugar B_1 y coloca marca en el lugar O_1 .

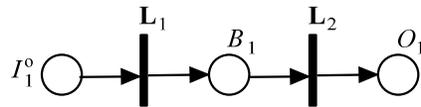


Figura 3.20: Estructura PN del comportamiento de ejecución secuencial.

El problema de acumulación de marcas y de restablecimiento para la estructura PN del comportamiento de ejecución secuencial se cubre con la Ecuación 3.3a y 3.6a, donde el lugar de restablecimiento está en función del marcado del lugar de entrada inicial I_1^o , es decir, si $I_1^o=0$ y $O_1=1$, entonces el lugar $G(\mathbf{L}_1)=1$. La Figura 3.21 muestra la estructura PN correspondiente.

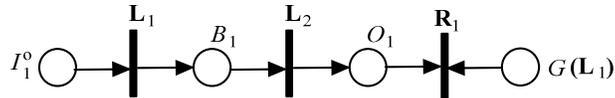


Figura 3.21: Estructura PN del comportamiento de ejecución secuencial con restablecimiento.

Sin embargo, existe un problema en el comportamiento de la estructura PN de ejecución secuencial si el lugar de salida $B_1 = 1$ tiene marca y el lugar de entrada $I_1^o = 0$ no tiene marca, ya que la transición \mathbf{L}_2 está habilitada y puede colocar marca en el lugar de salida O_1 . Se requiere que cada lugar intermedio entre el lugar de entrada I_1^o y el de salida O_1 también sean restablecidos si tienen algún lugar queda con marca y el lugar de entrada $I_1^o = 0$. La Figura 3.22 muestra la propuesta de estructura de PN para el comportamiento de ejecución secuencial de n-transiciones.

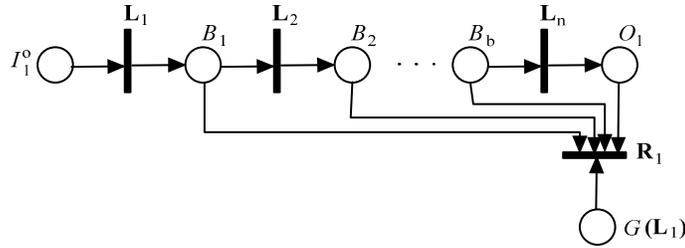


Figura 3.22: Estructura PN del comportamiento de ejecución secuencial generalizada.

Cada transición puede tener como entrada estructuras lógicas arriba descritas y por consiguiente cada una de éstas queda en función de la ecuación de marcado de sus lugares de entrada correspondiente. La Figura 3.23 muestra el comportamiento dinámico de estructura PN del comportamiento de ejecución secuencial. a) marca en el lugar de entrada I_1^o , b) disparo de la transición L_1 , consumo de la marca del lugar de entrada I_1^o y colocación de marca en el lugar B_1 , c) actualización del marcado en I_1^o , disparo de las transiciones L_1 y L_2 , colocación de marca en el lugar de salida O_1 (la actualización del marcado se realiza dos ocasiones). El problema de acumulación de marcas queda resuelto por la Ecuación 3.3a, d) marcado en el lugar de restablecimiento $G(L_1)$ por Ecuación 3.6a.

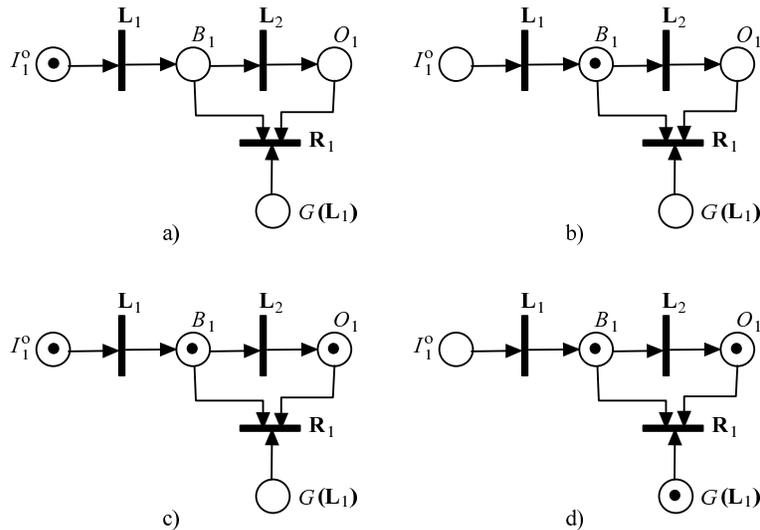


Figura 3.23: Comportamiento dinámico de la estructura PN del comportamiento de ejecución secuencial.

G) Estructura PN del comportamiento de decisión:

La estructura PN del comportamiento de decisión se muestra en la Figura 3.24. Para habilitar la transición \mathbf{L}_1 se requiere de marca en los lugares de entrada $I_1^o = 1$ y $I_2^o = 1$, sin embargo, si el lugar de entrada $I_3^o = 1$ también tiene marca, entonces también está habilitada la transición \mathbf{L}_2 . La primera transición que se dispare, consume la marca del lugar de entrada I_1^o , y deshabilita a la otra transición.

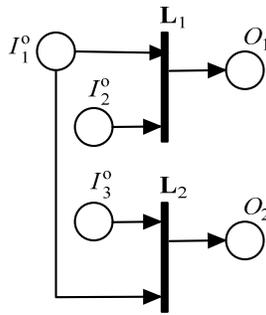


Figura 3.24: Estructura PN para el comportamiento de decisión.

Para que la estructura PN del comportamiento de decisión sea equivalente al comportamiento de la línea de control de la Figura 1.13 es necesario consumir la marca de los lugares de salida, por lo cual se agrega el lugar de restablecimiento $G(\mathbf{L}_1)$ y $G(\mathbf{L}_2)$ para cada una de estas, y donde el lugar de restablecimiento está en función del marcado de los lugares de entrada a la transición \mathbf{L}_1 y \mathbf{L}_2 respectivamente con base en la Ecuación 3.6a, y el problema de acumulación de marcas en función de la Ecuación 3.3a.

La generalización para l-transiciones de la estructura PN del comportamiento de decisión se muestra en la Figura 3.26, donde puede observarse que cada lugar de salida queda en función del marcado de los lugares de entrada de la transición que la une, y que las transiciones en general para su habilitación requieren de que el lugar de entrada $I_1^o = 1$ tenga marca.

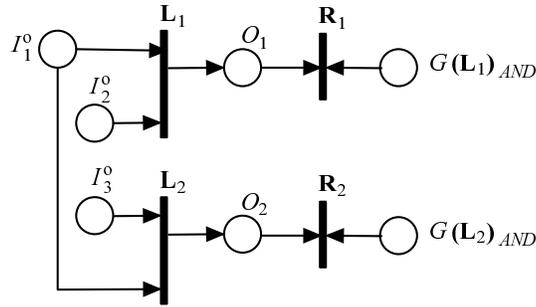


Figura 3.25: Estructura PN para el comportamiento de decisión con lugar de restablecimiento.

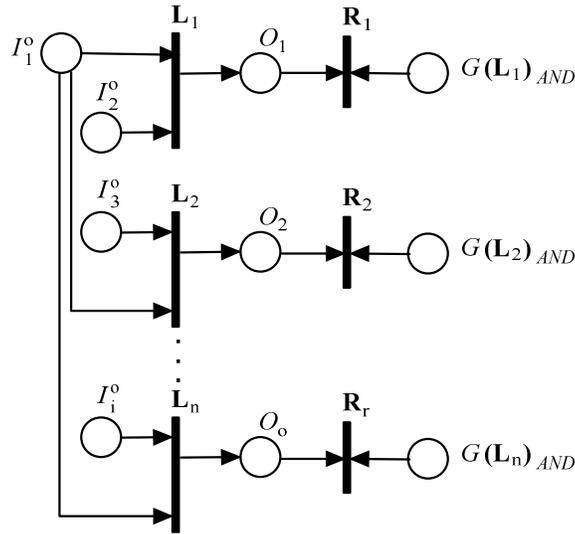


Figura 3.26: Estructura PN generalizada para el comportamiento de decisión.

El comportamiento dinámico de la estructura PN de decisión para dos opciones de decisión se muestra en la Figura 3.27. a) marca en los lugares de entrada $I_1^o = 1$ y $I_3^o = 1$, transición L_2 habilitada, b) disparo de la transición L_2 y colocación de marca en el lugar de salida $O_2 = 1$, c) actualización del marcado en lugares de entrada, se habilita la transición L_1 ya que se activó la entrada correspondiente al lugar $I_2^o = 1$, la transición L_2 está inhabilitada por la Ecuación 3.3a, d) disparo de transición L_1 y colocación de marca en el lugar de salida $O_1 = 1$, d) actualización del marcado de los lugares de entrada con $I_1^o = 0$, lo que habilita el marcado de los lugares de restablecimiento $G(L_1)_{AND} = G(L_2)_{AND} = 1$ por la Ecuación 3.6a.

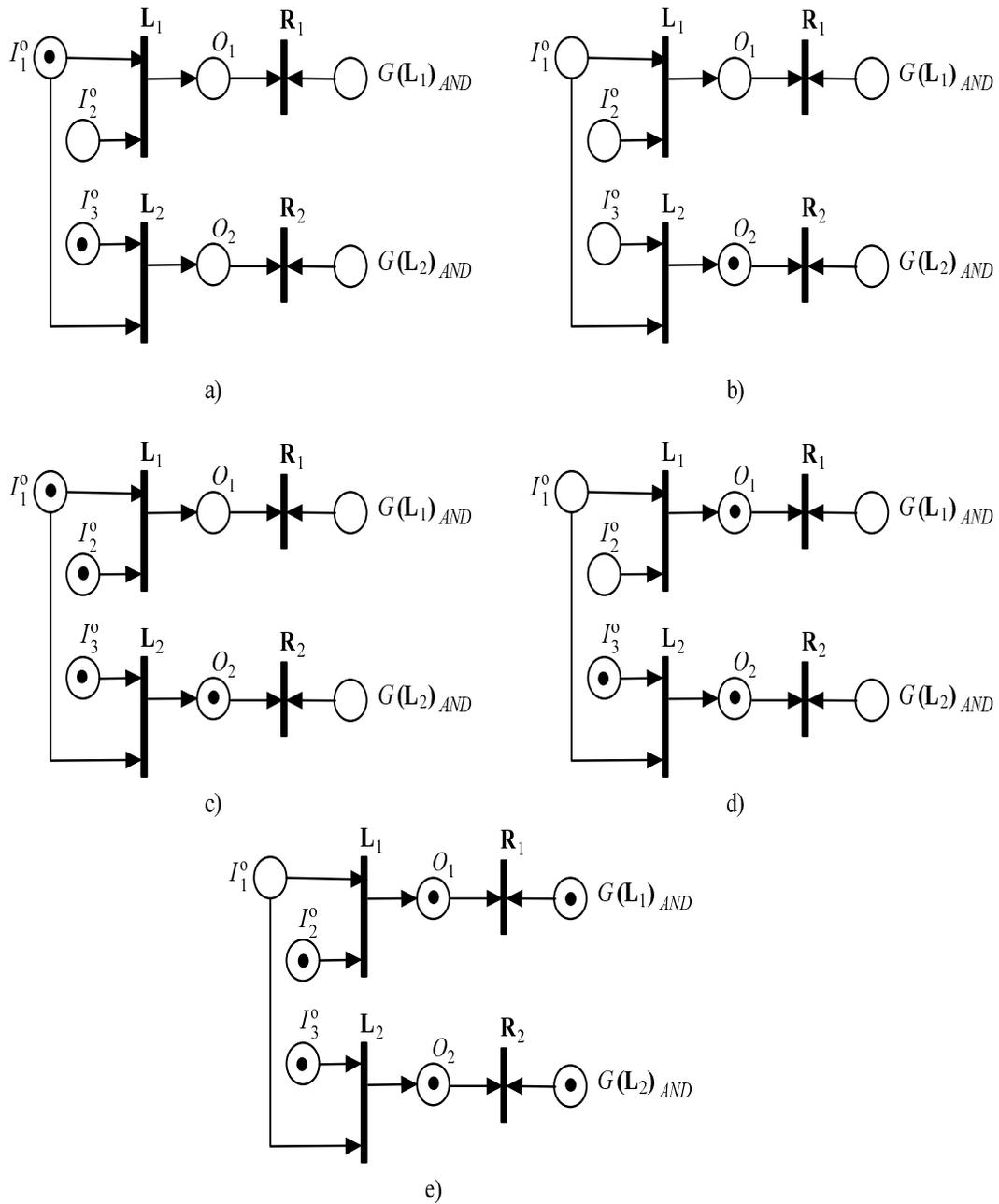


Figura 3.27: Comportamiento dinámico de la estructura PN del comportamiento de decisión.

H) Estructura PN del comportamiento de concurrencia:

La Figura 3.28 muestra la estructura PN del comportamiento de concurrencia, la cual es similar a la estructura PN del comportamiento de ejecución secuencial independiente de dos procesos, como también se observa en la línea de control en LD de la Figura 1.14 donde todas las señales son diferentes.

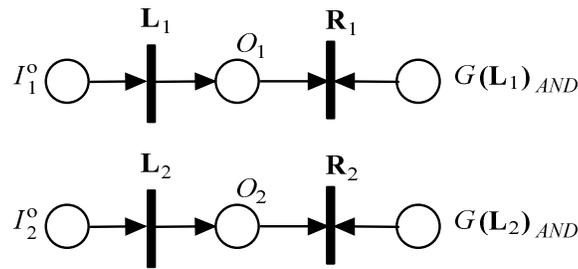


Figura 3.28: Estructura PN del comportamiento de concurrencia.

El marcado del lugar de salida O_1 depende del marcado de lugar de entrada I_1^o de la transición L_1 , así como, el lugar de restablecimiento $G(L_1)$ depende del marcado del lugar de entrada I_1^o de la transición L_1 , la cual su habilitado esta en función de la Ecuación 3.3a para evitar el acumulamiento de marcas en el lugar de salida O_1 . De igual forma es para la estructura PN de los elementos I_2^o , L_2 , O_2 , R_2 y $G(L_2)$.

I) Estructura PN del comportamiento de sincronización y agrupación:

El comportamiento de sincronización y agrupación en los algoritmos de control en LD es equivalente a las estructuras anteriormente analizadas para las lógicas AND y OR respectivamente, por lo que se omite volver a presentarlas.

J) Estructura PN del comportamiento de auto-lazo:

Para modelar el comportamiento de un auto-lazo de la línea de control en LD de la Figura 1.11, puede realizarse como si fueran dos estructuras PN de la lógica AND donde solo el lugar de entrada I_1^o es de la transición L_1 y el lugar de salida O_1 es también lugar de entrada de la transición L_2 , como muestra la Figura 3.29. Para

poder colocar una marca en el lugar de salida O_1 es necesario que se habilite la transición \mathbf{L}_1 , ya que la transición \mathbf{L}_1 depende del marcado del propio lugar de salida O_1 y el cual está en función de la Ecuación 3.3a. Por lo tanto, si los lugares de entrada $I_1^o = 1$ y $I_2^o = 1$ tienen marca, se habilita la transición \mathbf{L}_1 , al dispararse, coloca una marca en el lugar de salida O_1 , lo cual habilita a la transición \mathbf{L}_2 , al dispararla consume la marca y coloca una marca en el lugar de salida O_1 . El problema de acumulación de marcas en ambas transiciones es resuelto con la Ecuación 3.5a y el marcado del lugar de restablecimiento $G(\mathbf{L}_2)$ está en función de la Ecuación 3.6a que corresponde al marcado de los lugares de entrada de la transición \mathbf{L}_2 que es la que mantiene la marca en el lugar de salida O_1 .

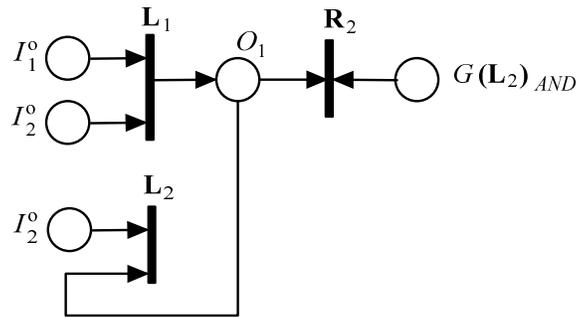


Figura 3.29: Estructura PN del comportamiento de auto-lazo.

La Figura 3.30 muestra el comportamiento dinámico de la estructura PN del comportamiento de un auto-lazo equivalente a la línea de control de la Figura 1.11. a) marca en los lugares de entrada $I_1^o = I_2^o = 1$, transición \mathbf{L}_1 habilitada, b) disparo de transición \mathbf{L}_1 coloca marca en el lugar de salida O_1 , c) actualización del marcado de los lugares de entrada (observe que el lugar $I_1^o = 0$, equivalente a desactivar una señal de entrada física, botón retentivo), se habilita la transición \mathbf{L}_2 , d) disparo de la transición \mathbf{L}_2 consume y coloca marca del lugar de salida O_1 , e) actualización del marcado de la red, f) marca en lugar de restablecimiento $G(\mathbf{L}_2)$ en función de la Ecuación 3.6a.

Si existieran más actualizaciones de las marcas de los lugares de entrada, por la Ecuación 3.3a para ambas transiciones $G(\mathbf{L}_1)$ y $G(\mathbf{L}_2)$ se evita la acumulación de marca en el lugar de salida O_1 .

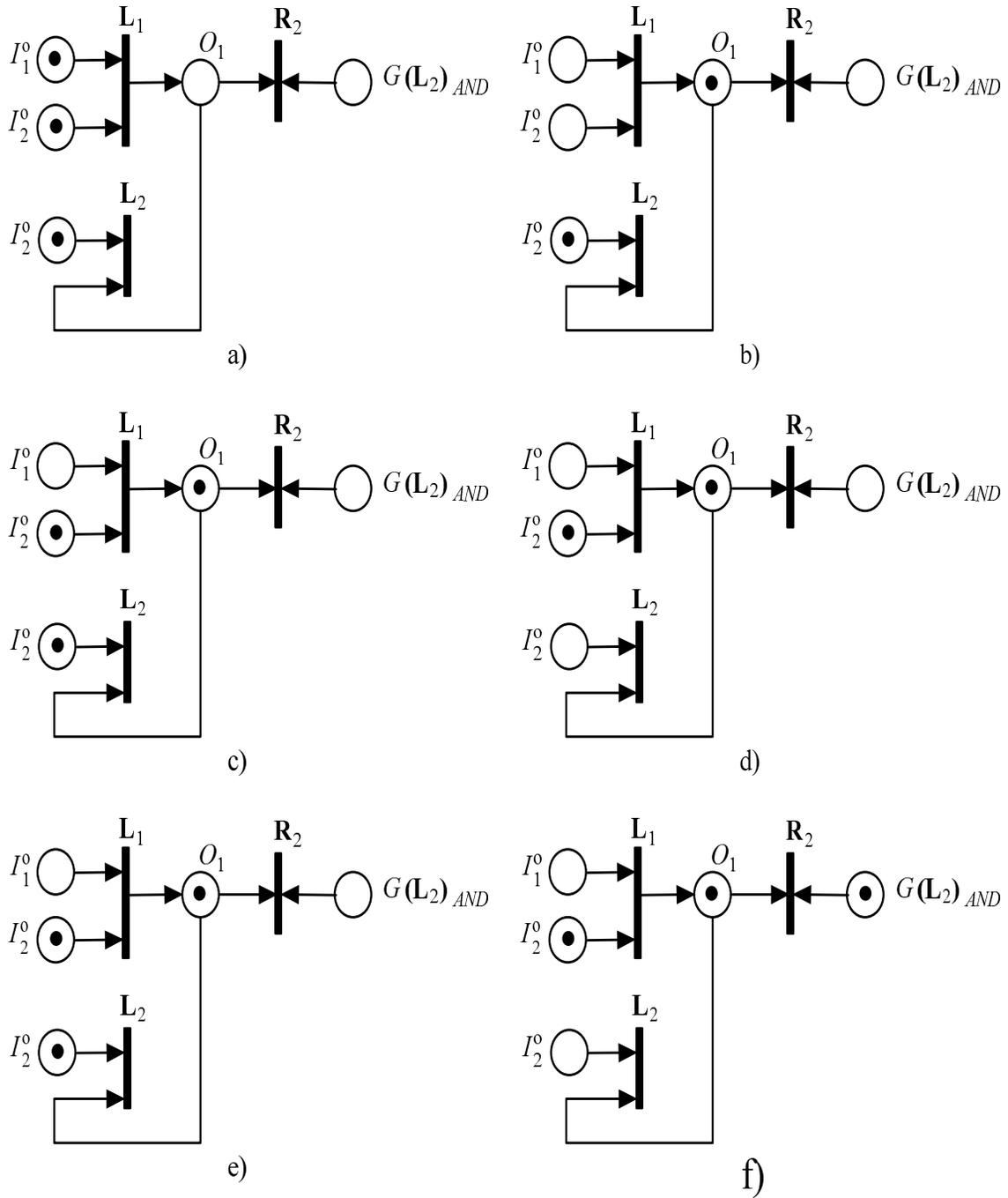


Figura 3.30: Comportamiento dinámico de la estructura PN del comportamiento de auto-lazo.

Tal vez, no se pueda asegurar que sean los únicos tipos de líneas de control y comportamiento de los DES, o que todo algoritmo de control pueda ser reducido a alguna de las estructuras en PN propuestas y descritas, sin embargo son la base para definir una metodología de transformación de algoritmos de control desarrollados en LD a PN a través de la definición formal de una extensión de PN denominada LDPN.

3.5. Definición de la red PNL D ordinaria

Con base en el análisis de líneas de control y comportamiento en el lenguaje de programación LD, de la consideración del ciclo de trabajo de los sistemas basados en PLC, de la distribución de señales físicas a elementos gráficos de PN y sus Ecuaciones 3.1a - 3.1f, 3.2a - 3.2c, de las estructuras PN de los comportamientos lógicos y de los DES; AND, OR, AND-OR, Set - Reset, Bloqueo, Ejecución secuencial, Decisión, Concurrencia, Sincronización, Agrupación y Auto-lazo sustentados en las Ecuaciones 3.3a y 3.4a para resolver el problema de acumulación de marcas en lugares de salida y en las Ecuaciones 3.6a, 3.6b, 3.7a, 3.7b, 3.8a y 3.8b para el marcado de los lugares de restablecimiento, se define la LDPN de tipo ordinario:

Definición Formal de la red LDPN de tipo ordinario :

Definición 3.1 *Una LDPN de tipo ordinario es una 5-tupla $(\mathbf{P}, \mathbf{T}, \mathbf{W}, \mathbf{F}, \mathbf{M}_0)$, donde:*

$\mathbf{P} = \{I \cup O \cup B \cup G\}$ *es un conjunto finito de lugares, donde:*

$I = \{I_1, I_2, I_3, \dots, I_i\}$ *es un conjunto finito de lugares que representan señales de entradas físicas, y por las Ecuaciones 3.1a y 3.1a de la distribución de señales: $I_1 = \{I_1^o \cup I_1^c\}$, $I_2 = \{I_2^o \cup I_2^c\}$, $I_3 = \{I_3^o \cup I_3^c\}$, ... , $I_i = \{I_i^o \cup I_i^c\}$ son los lugares que representan los contactos NO y NC de cada señal de entrada física, y su marcado queda en función de la Ecuación 3.2a.*

$O = \{O_1, O_2, O_3, \dots, O_o\}$ *es un conjunto finito de lugares que representan señales de salidas físicas, y por las Ecuaciones 3.1c y 3.1d de la distribución de señales: $O_1 = \{O_1^o \cup O_1^c\}$, $O_2 = \{O_2^o \cup O_2^c\}$, $O_3 = \{O_3^o \cup O_3^c\}$, ... , $O_o = \{O_o^o \cup O_o^c\}$ son los lugares que representan los contactos NO y NC de cada señal de salida física, y su marcado queda en función de la Ecuación 3.2b.*

$B = \{B_1, B_2, B_3, \dots, B_b\}$ es un conjunto finito de lugares que representan señales de memoria, y por las Ecuaciones 3.1e y 3.1f de la distribución de señales: $B_1 = \{B_1^o \cup B_1^c\}$, $B_2 = \{B_2^o \cup B_2^c\}$, $B_3 = \{B_3^o \cup B_3^c\}$, ... , $B_b = \{B_b^o \cup B_b^c\}$ son los lugares que representan los contactos NO y NC de cada señal de memoria, y su marcado queda en función de la Ecuación 3.2c.

$G = \{G(\mathbf{T}_1), G(\mathbf{T}_2), G(\mathbf{T}_3), \dots, G(\mathbf{T}_g)\}$ es un conjunto finito de lugares para restablecer lugares de salida, donde su marcado queda definido por las Ecuaciones 3.6a, 3.6b, 3.7a, 3.7b, 3.8a y 3.8b, y donde \mathbf{T} es un tipo de transición de la red.

$\mathbf{T} = \{\mathbf{I}^{cl_o} \cup \mathbf{O}^{cl_o} \cup \mathbf{B}^{cl_o} \cup \mathbf{L} \cup \mathbf{R}\}$ es un conjunto finito de transiciones, donde:

$\mathbf{I}^{cl_o} = \{\mathbf{I}_1^{cl_o}, \mathbf{I}_2^{cl_o}, \mathbf{I}_2^{cl_o}, \dots, \mathbf{I}_i^{cl_o}\}$ es un conjunto finito de transiciones que tiene como lugar(es) de entrada(s) I , y donde $\mathbf{I}_1^{cl_o} = \{\mathbf{I}_1^c \cup \mathbf{I}_1^o\}$, $\mathbf{I}_2^{cl_o} = \{\mathbf{I}_2^c \cup \mathbf{I}_2^o\}$, $\mathbf{I}_3^{cl_o} = \{\mathbf{I}_3^c \cup \mathbf{I}_3^o\}$, ..., $\mathbf{I}_i^{cl_o} = \{\mathbf{I}_i^c \cup \mathbf{I}_i^o\}$ son transiciones que tiene como entrada lugares I_i^c y I_i^o que representan contactos NC y NO respectivamente.

$\mathbf{O}^{cl_o} = \{\mathbf{O}_1^{cl_o}, \mathbf{O}_2^{cl_o}, \dots, \mathbf{O}_o^{cl_o}\}$ es un conjunto finito de transiciones que tienen como lugar(es) de entrada(s) O , y donde $\mathbf{O}_1^{cl_o} = \{\mathbf{O}_1^c \cup \mathbf{O}_1^o\}$, $\mathbf{O}_2^{cl_o} = \{\mathbf{O}_2^c \cup \mathbf{O}_2^o\}$, $\mathbf{O}_3^{cl_o} = \{\mathbf{O}_3^c \cup \mathbf{O}_3^o\}$, ..., $\mathbf{O}_o^{cl_o} = \{\mathbf{O}_o^c \cup \mathbf{O}_o^o\}$ son transiciones que tiene como entrada lugares O_o^c y O_o^o que representan contactos NC y NO respectivamente.

$\mathbf{B}^{cl_o} = \{\mathbf{B}_1^{cl_o}, \mathbf{B}_2^{cl_o}, \dots, \mathbf{B}_b^{cl_o}\}$ es un conjunto finito de transiciones que tienen como lugar(es) de entrada y/o de salida B , y donde $\mathbf{B}_1^{cl_o} = \{\mathbf{B}_1^c \cup \mathbf{B}_1^o\}$, $\mathbf{B}_2^{cl_o} = \{\mathbf{B}_2^c \cup \mathbf{B}_2^o\}$, $\mathbf{B}_3^{cl_o} = \{\mathbf{B}_3^c \cup \mathbf{B}_3^o\}$, ..., $\mathbf{B}_b^{cl_o} = \{\mathbf{B}_b^c \cup \mathbf{B}_b^o\}$ son transiciones que tiene como entrada lugares B_b^c y B_b^o que representan contactos NC y NO respectivamente.

$\mathbf{L} = \{\mathbf{L}_1, \mathbf{L}_2, \dots, \mathbf{L}_l\}$ es un conjunto finito de transiciones auxiliares que pueden tener como lugar(es) de entrada y/o de salida combinaciones de lugares I, O , y B .

$\mathbf{R} = \{\mathbf{R}_1, \mathbf{R}_2, \dots, \mathbf{R}_r\}$ es un conjunto finito de transiciones que tiene de entrada lugar(es) de restablecimiento G .

$F \subseteq (P \times T) \cup (T \times P)$ es un conjunto de arcos,

$W : F \rightarrow \{1\}$ todos los pesos de los arcos de la red son igual a 1. y,

$M_0 = P \rightarrow \{0, 1\}$ marcado inicial.

Convertir o transformar algoritmos de control en LD a LDPN permite utilizar las herramientas gráficas y matemáticas de la teoría de redes de Petri para modelar, simular y validar comportamientos de los primeros.

3.5.1. Marcado de la red LDPN

Con base en la Ecuaciones de la distribución de señales 3.2a, 3.2b y 3.2c, en las Ecuaciones para evitar el acumulamiento de marcas en los lugares de salida 3.3a, 3.3b, 3.4a, 3.4b, 3.5a y 3.5b y en las Ecuaciones para restablecer o consumir la marca en los lugares de salida 3.6a, 3.6b, 3.7a, 3.7b, 3.8a y 3.8b, se tienen las condiciones para actualizar el marcado en la red LDPN posterior a cada evaluación de la ecuación de estados, es decir, en cada marcado M_{k+1} obtenido de la ecuación de estados se debe considerar la actualización de los lugares en función de las Ecuaciones propuestas para obtener el modelado dinámico de la LDPN equivalente al comportamiento cíclico de un algoritmo de control en LD.

$$M = \begin{bmatrix} I_i & I_i^o & I_i^c & \dots & B_b & B_b^o & B_b^c & \dots & O_o & O_o^o & O_o^c & \dots & G_{AND} & \dots & G_{OR} & \dots & G_{ANDOR} \\ \text{Proceso} & 3,2a & 3,2a & & & 3,2b & 3,2b & & & 3,3c & 3,3c & & & & & & \\ & & & & 3,3b & & & & 3,3a & & & & & & & & \\ & & & & 3,4b & & & & 3,4a & & & & & & & & \\ & & & & 3,5b & & & & 3,5a & & & & & & & & \\ & & & & & & & & & & & & 3,6a & & 3,7a & & 3,8a \\ & & & & & & & & & & & & 3,6b & & 3,7b & & 3,8b \end{bmatrix}$$

El marcado de los lugares I están en función de las señales de entradas físicas (sensores). En la ecuación de estado de la LDPN, cada vez que se dispare una o más transiciones se debe evaluar la Ecuación o Ecuaciones correspondientes para el marcado de los lugares B , O y G para modelar el comportamiento dinámico del sistema.

3.5.2. Metodología para transformar algoritmos de control en LD a LDPN

Con base en las estructuras PN de comportamiento de DES y sus Ecuaciones para evitar la acumulación de marcas y restablecer los lugares de salida y de la definición formal de la red LDPN, se propone una metodología formal para transformar algoritmos

de control en LD a PNL D para su modelo, simulación y validación.

- I) Identificar en el algoritmo de control los tipos de señales utilizadas: señales de entrada y salida físicas y de memoria.
- II) Con base en la distribución de señales, realizar la estructura PN correspondiente considerando el número de contactos NO y NC de cada una de las señales. La distribución para las señales de entradas físicas colocarlas de lado izquierdo, las de memoria al centro y la de salidas físicas a la derecha de la red que se va formando.
- III) Identificar el tipo de línea de control en LD y transformar a la estructura PN correspondiente. Si la línea de control contiene arreglos AND-OR es necesario considerar el mismo número de lugares al mismo número de rutas posibles a energizar la bobina, es decir, si un contacto NO está en tres posibles rutas de energizar la bobina, entonces el lugar que representa ese contacto debe presentarse en tres círculos conectado hacia cada transición que representa la ruta posible de energizar la bobina.
- IV) Analizar en cada estructura PN que Ecuación debe aplicarse para el problema de acumulación de marcas.
- V) Analizar en cada lugar de salida que Ecuación debe aplicarse para consumir la marca correspondiente.

Con la transformación del algoritmo de control en LD a LDPN se tiene la capacidad de aplicar las herramientas de PN.

3.5.3. Reglas de disparo de transición de la LDPN

Al igual que un algoritmo de control en LD, donde los contactos NC permiten el flujo de energía, entonces los lugares I_i^c, O_o^c, B_b^c tienen marca, siendo este el marcado inicial M_0 del LDPN. Los siguientes marcados M_k de la LDPN están en función de que

lugares de entrada (correspondiente a la señal de los sensores) deben irse marcado de acuerdo con el comportamiento de los procesos.

Para describir y simular el comportamiento dinámico de un algoritmo de control en LD a través de la red LDPN se consideran las siguientes reglas de disparo de transiciones:

- a) Una transición $\mathbf{T} = \{\mathbf{I}^{cl_o}, \mathbf{O}^{cl_o}, \mathbf{B}^{cl_o}, \mathbf{L}, \mathbf{R}\}$ es habilitada si cada lugar de entrada $\mathbf{P} = (I, O, B, G)$ de \mathbf{T} tienen marca, es decir, $M(\mathbf{P}) = \mathbf{W}(P, T) = 1$.
- b) La transición o transiciones habilitadas deben ser disparadas en un mismo tiempo.
- c) La LDPN es binaria, por lo que el disparo de una transición habilitada \mathbf{T} remueve la única marca $\mathbf{W}(P, T) = 1$ de cada lugar de entrada \mathbf{P} de \mathbf{T} , y coloca una marca $\mathbf{W}(T, P) = 1$ a cada lugar de salida \mathbf{P} de \mathbf{T} .
- d) Al igual que el ciclo de trabajo de los sistemas basados en PLC, debe de actualizarse el marcado y considerarse las Ecuaciones 3.2a - 3.2c para drenar o marcar lugares de la distribución de señales ($I_i^{cl_o}, O_o^{cl_o}, B_b^{cl_o}$), además, las Ecuaciones 3.3a, 3.3b, 3.4a, 3.4b, 3.5a y 3.5b para el problema de acumulación de marcas y las Ecuaciones 3.6a, 3.6b, 3.7a, 3.7b, 3.8a y 3.8b para el restablecimiento de los lugares de salida.

3.6. Conclusiones del capítulo

La propuesta de distribución de señales permite representar las señales físicas y de memoria utilizadas en los algoritmos de control desarrollados en LD en elementos de PN, incluso, el comportamiento de los contactos NO y NC de una misma señal, es decir, que solamente un tipo de contacto puede estar activo, por ende solo un tipo de lugares puede tener marca. Además, si la señal solo tiene un tipo de contacto NO o NC, la distribución de señales solo contendría un tipo de transición con lugares de entrada correspondientes al tipo de contacto.

El trabajo cíclico de los sistemas basados en PLC, que incluye a los algoritmos de control en LD, realizan la lectura de los estados de las señales de entrada en cada ciclo, entonces a los lugares que representan a cada una de estas señales se les tendría que colocar una marca en cada actualización, lo que habilitaría nuevamente a la transición

correspondiente y si es disparada esta, entonces el lugar de salida tendría una acumulación de marcas, lo cual no es equivalente al comportamiento correspondiente de una bobina. Para lo anterior, la propuesta de Ecuaciones 3.3a y 3.4a de habilitar la transición en función del marcado del lugar de salida evita el problema de acumulación de marcas, logrando que un lugar de salida solo tenga una marca aunque los lugares de entrada continúen actualizando su marcado.

Comportamientos lógicos y característicos de DES en los algoritmos de control en LD pueden ser representados gráficamente por elementos de PN, en particular se propusieron y analizaron en cuanto a su comportamiento equivalente entre LD y PN fueron las lógicas AND, OR, AND - OR, Set - Reset, bloqueo, ejecución secuencial, decisión, concurrencia, sincronización, agrupación y auto-lazo. En todas las estructuras PN propuestas se incluye el problema de acumulación de marcas.

En los algoritmos de control en LD, una ruta de contactos NO y/o NC en función de su condición de permitir o no el flujo de energía para energizar o des-energizar una bobina es un comportamiento elemental para la ejecución continua y periódica en el proceso industrial. Para modelar este comportamiento en las PN se propusieron las Ecuaciones 3.6a, 3.6b, 3.7a, 3.7b, 3.8a y 3.8b, las cuales están en función del marcado de los lugares de entrada de la transición que mantiene activa la marca en el lugar de salida correspondiente, equivalente a la ruta de contactos que energizan una bobina.

Las estructuras propuestas en elementos de PN de los comportamientos lógicos y característicos de líneas de control utilizadas en los LD, las Ecuaciones para resolver el problema de acumulación de marcas y para el restablecimiento de los lugares de salida son la base de la definición formal de una extensión de PN, denominada LDPN de tipo ordinario para transformar, modelar, simular y validar algoritmos de control en LD.

Para modelar y simular la red LDPN se define la regla de disparo de transición correspondiente, lo que implica, modelar y simular el comportamiento del algoritmo de control equivalente.

Capítulo 4

Validación de la red PNL

Para el diseño de algoritmos de control en LD se considera principalmente dos tipos de especificaciones; formales e informales. Las especificaciones formales incluyen las señales de seguridad y de operación del proceso. Las especificaciones informales son propuestas por el diseñador al estar analizando el proceso y desarrollando el algoritmo de control correspondiente, para posteriormente hacer pruebas de este en la puesta en operación del sistema de producción.

Como se ha mencionado, el diseño de algoritmos de control en LD son desarrollados heurísticamente y con base en la experiencia del programador o responsable del control del proceso [24]. El problema de garantizar algoritmos de control seguros ha sido tratado en la teoría a través de diferentes enfoques que tienen como base principal las especificaciones formales del sistema a controlar y su validación o verificación, los cuales se consideran limitados al centrarse principalmente en conceptos teóricos (seguridad, vivacidad y alcanzabilidad) [30]. La Figura 4.1 muestra el contexto para el diseño de algoritmos de control en LD para DES.

El enfoque de validación que se presenta es con base en la posibilidad de que se presenten fallas reales (corto-circuito y/o circuito-abierto) en el sub-sistema sensor-cableado-módulo de entradas de los sistemas basados en PLC, lo que permite predecir condiciones de riesgo o peligro en máquinas y procesos industriales.

Una condición de riesgo es el accionamiento no deseado de un actuador en un proceso industrial o como parte de una máquina sofisticada. En el algoritmo de control una condición de falla se refleja en el energizado de una bobina de salida física, es decir, la señal de salida física que está conectada a un actuador en el proceso. Este compor-

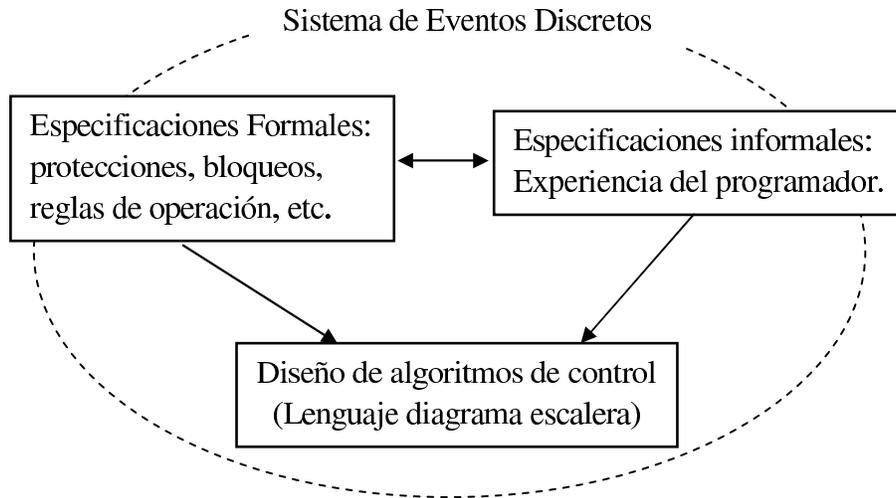


Figura 4.1: Contexto para el diseño de algoritmos de control en LD.

tamiento de riesgo, en la red LDPN es cuando un lugar de salida O_o tiene marca, por consiguiente, se considera un comportamiento seguro en el algoritmo de control en LD cuando en condiciones de falla de las señales de entradas físicas I_i no existen bobinas de señales de salidas físicas O_o energizadas. Es importante descartar el comportamiento del algoritmo de control en funcionamiento normal de operación.

El enfoque de validación de la presente investigación consiste en verificar si en la red LDPN existe uno o más lugares de salida O_o con marca al evaluar un marcado del conjunto de fallas posibles de corto-circuito y/o circuito-abierto en una o más señales del sub-sistema de entradas físicas. Si existe marca en uno o más lugares de salida O_o , se debe analizar la condición de falla que lo origina para su consideración en el algoritmo de control. Es decir, la validación consiste en verificar el comportamiento del algoritmo de control en LD en condiciones de falla de corto-circuito y/o circuito-abierto en una o más señales del sub-sistema de entradas física.

En los sistemas basados en PLC, los sensores y su conexión a los módulos de entrada, y los módulos de salida y su conexión a los actuadores pueden presentar principalmente dos tipos de fallas, las cuales son analizadas en la siguiente sección.

4.1. Fallas físicas en sistemas basados en PLC

Independientemente del principio de funcionamiento de los sensores y actuadores, los sub-sistemas sensor-cableado-módulo de entradas físicas y módulos de salidas físicas-cableado-actuador pueden presentar dos tipos de fallas: corto-circuito o circuito-abierto, lo cual origina que una señal de entrada o salida física del sistemas basado en PLC quede activa o desactiva siempre, ocasionando que el algoritmo de control tenga un comportamiento no deseado que afecta la producción, pero sobre todo la seguridad de las personas y de la maquinaria sofisticada.

4.1.1. Falla de corto-circuito

La falla de corto-circuito en el sub-sistema de entradas físicas puede presentarse en el sensor, o en el cableado o en una de las secciones del módulo de entradas. Este tipo de falla, ocasiona que la señal de entrada física correspondiente permanezca activa para el algoritmo de control, es decir, en cada barrido del PLC la señal en corto-circuito siempre será 1 para sus contactos NO y 0 para los NC, lo que indicaría que los lugares de esta señal en la red LDPN no cambiaría su marcado, es decir, el lugar $I_i^o = 1$ y $I_i^c = 0$, respectivamente. La Figura 4.2 muestra la condición de falla de corto-circuito, a) sub-sistema de entradas físicas en los sistemas basados en PLC, y b) comportamiento equivalente en la distribución de señales.

Para el caso de que la falla ocurra en el sub-sistema de salidas físicas de los sistemas basados en PLC. Si la falla es el cableado, entonces los actuadores no se podrán energizar, sin embargo, en el algoritmo de control si se reflejarían los cambios de la señal de salida, por lo que el algoritmo se continuaría evaluando. Si el corto-circuito es en una sección del módulo de salidas, entonces la salida en falla siempre estaría activa y en consecuencia el actuador correspondiente también siempre estaría activado, equivalente a que el lugar de salida $O_o^o = 1$ y $O_o^c = 0$. de la señal correspondiente. La Figura 4.3 muestra la falla de corto-circuito para ambos casos en el sub-sistema de salidas físicas de un sistema basado en PLC, y el correspondiente comportamiento en la distribución de señales de la red LDPN para el caso de falla en el módulo de salidas.

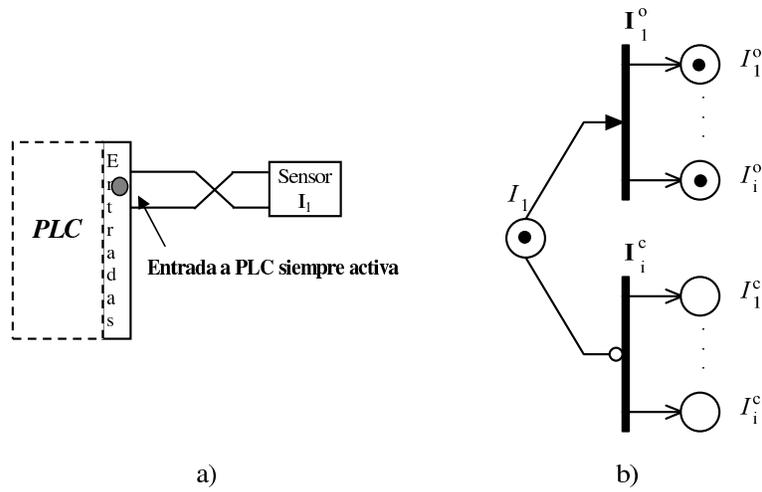


Figura 4.2: Falla de corto-circuito, a) en el sub-sistema de entradas del PLC, y b) en la red LDPN.

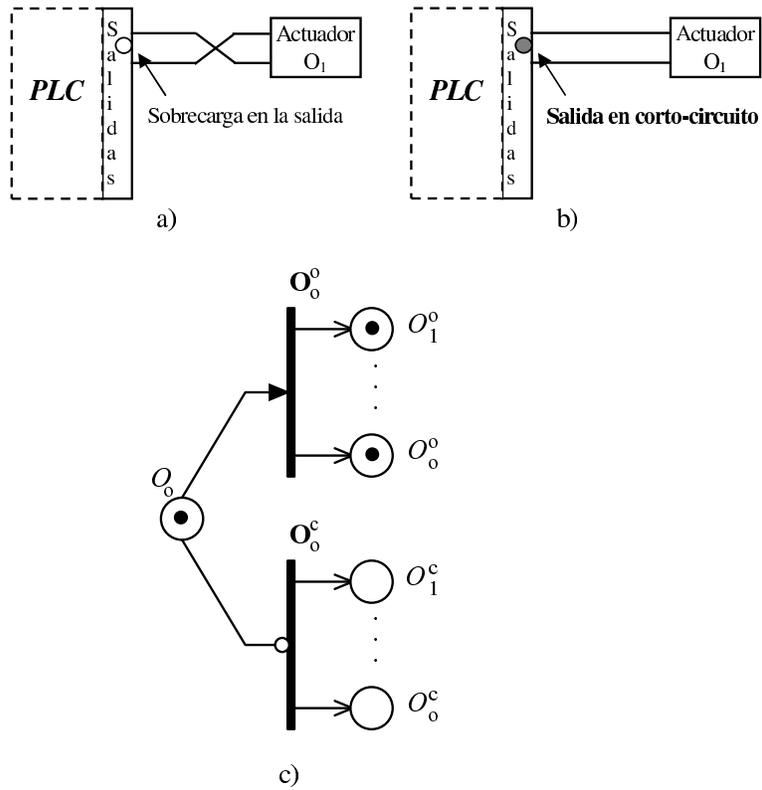


Figura 4.3: Falla de corto-circuito en el sub-sistema del PLC, a) en el cableado, b) en el módulo de salidas y c) en la red LDPN.

4.1.2. Falla de circuito-abierto

La falla de circuito-abierto en el sub-sistema de entradas también puede estar en el sensor, en el cableado o en una de las secciones del módulo de entradas físicas, ocasionando que la señal de entrada física correspondiente permanezca desactivada para el algoritmo de control, lo cual produce, que sus contactos *NO* siempre estarán abiertos y los *NC* siempre estarán cerrados, equivalente a que los lugares en la LDPN $I_i^o = 0$ no tengan marca y $I_i^c = 1$ tengan marca siempre, respectivamente. La Figura 4.4 muestra la falla de circuito-abierto en el sub-sistema de entradas físicas de un sistema basado en PLC y el marcado correspondiente en la distribución de señales de la LDPN.

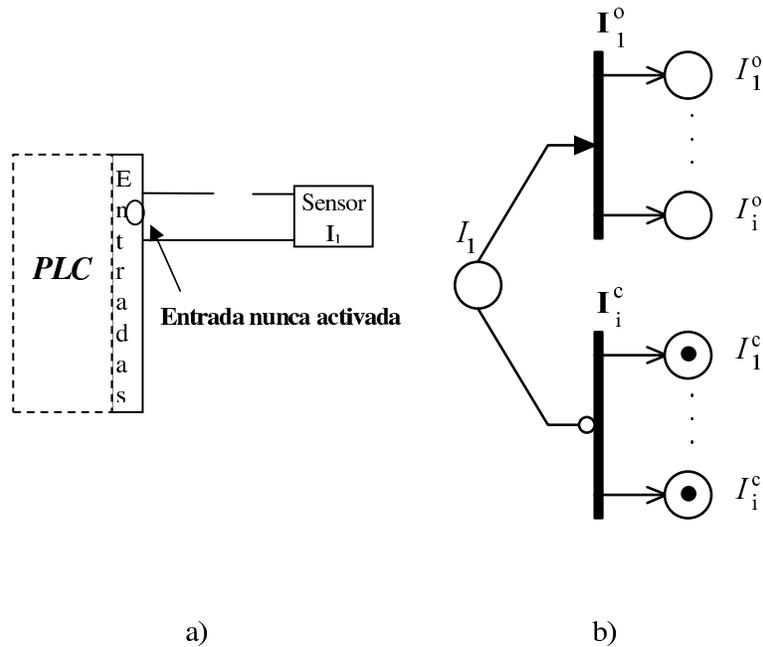


Figura 4.4: Falla de circuito-abierto, a) en el sub-sistema del PLC, y b) en la red LDPN.

Para el caso de falla de circuito-abierto en el sub-sistema de salidas, independientemente donde se presente, sección del módulo de salidas, cableado o actuador, nunca se realizaría la acción correspondiente en el proceso, es decir, el actuador nunca se energizaría, sin embargo, el algoritmo de control funcionaría conforme a la lógica establecida, al mandar energizar la bobina aunque no haya resultados, por lo que se omite figura al respecto.

Con base en el análisis descrito arriba de los efectos que originan los dos tipos de fallas de corto-circuito y circuito-abierto en los sistemas basados en PLC, se puede determinar que el comportamiento del algoritmo de control (no del proceso de producción) es afectado principalmente por las fallas en el sub-sistema de entradas. Por lo tanto, la presente investigación propone un método de validación de los algoritmos de control mediante la red LDPN para estados de fallas de corto-circuito y/o circuito-abierto en el sub-sistema de entrada de los sistemas basados en PLC.

4.2. Matriz de incidencia de la PNLD

De las Ecuaciones 3.1a - 3.1f de la distribución de señales se puede deducir que el comportamiento del arco inhibidor en la red LDPN ordinaria, tiene el mismo comportamiento que un arco habilitador cuando una transición \mathbf{I}_i^c , \mathbf{O}_o^c o \mathbf{B}_b^c es disparada. Por lo consiguiente el arco inhibidor en la red LDPN sigue teniendo la misma relación matemática en la matriz de incidencia correspondiente. La Ecuación 4.1 muestra la matriz de incidencia de la red LDPN para la distribución de la señal de entrada I de la Figura 3.2.

$$a_{ij} = \begin{bmatrix} I_i & I_1^o & I_2^o & \dots & I_i^o & I_1^c & I_2^c & \dots & I_i^c \\ \mathbf{I}_i^o & -1 & 1 & 1 & \dots & 1 & 0 & 0 & \dots & 0 \\ \mathbf{I}_i^c & -1 & 0 & 0 & \dots & 0 & 1 & 1 & \dots & 1 \end{bmatrix} \quad (4.1)$$

Donde a y b representan el número de lugares que representan al número de contactos NO y NC de la señal I_i . La red LDPN es ordinaria y el peso de todos sus arcos $W(\mathbf{I}_i^o, I_i^o) = 1$ y $W(\mathbf{I}_i^c, I_i^c) = 1$ se puede reducir la matriz de incidencia considerando las Ecuaciones 4.2a y 4.2b, la cual se muestra en la Ecuación 4.4.

$$A = \sum_{a=1}^a W(\mathbf{I}_i^o, I_a^o) \quad (4.2a)$$

$$B = \sum_{b=1}^b W(\mathbf{I}_i^c, I_b^c) \quad (4.2b)$$

$$aw_{ij} = \begin{bmatrix} I_i & I_i^o & I_i^c \\ \mathbf{I}_i^o & -1 & A & 0 \\ \mathbf{I}_i^c & -1 & 0 & B \end{bmatrix} \quad (4.3)$$

El comportamiento es igual para la distribución de señales de los lugares O_o y B_b .

Gráficamente la reducción no es posible debido a que cada lugar de la distribución de señales I_i^o , I_i^c , O_o^o , O_o^c , B_b^o , B_b^c son independientes y cada lugar tiene una única relación con una transición en la red LDPN. Dos o más lugares de la misma señal, por ejemplo dos lugares I_i^o como entrada a una misma transición es equivalente a tener dos contactos NO de una señal en una misma línea de control, lo cual es una redundancia inoperante en el control.

La Figura 4.5 muestra la red LDPN del algoritmo de control de arranque y paro de un motor, y la Ecuación 4.4 su matriz de incidencia considerando la sumatoria de los pesos de los arcos de lugares de una misma señal.

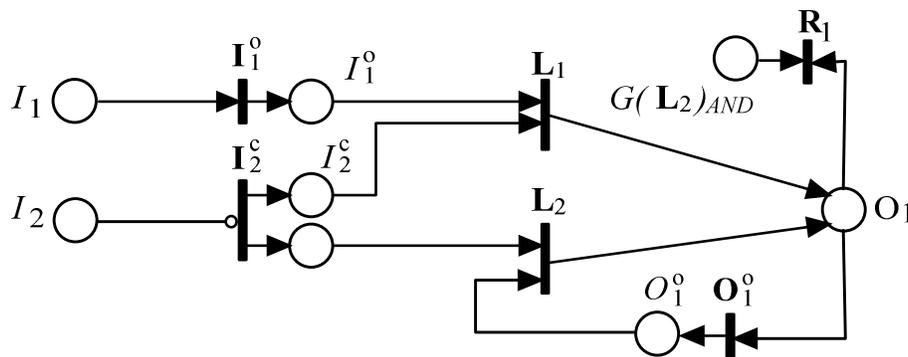


Figura 4.5: Red LDPN ejemplo.

$$aw_{ij} = \begin{bmatrix} & I_1 & I_1^o & I_2 & I_2^c & O_1 & O_1^o & G(\mathbf{L}_2)_{AND} \\ \mathbf{I}_1^o & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{I}_2^c & 0 & 0 & -1 & 2 & 0 & 0 & 0 \\ \mathbf{L}_1 & 0 & -1 & 0 & -1 & 1 & 0 & 0 \\ \mathbf{L}_2 & 0 & 0 & 0 & -1 & 1 & -1 & 0 \\ \mathbf{O}_1^o & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ \mathbf{R}_1 & 0 & 0 & 0 & 0 & -1 & 0 & -1 \end{bmatrix} \quad (4.4)$$

Puede observarse que gráficamente existen dos lugares I_2^c en la red LDPN y que la suma de los pesos de los arcos de la transición \mathbf{I}_2^c a estos lugares es 2, valor que está en la matriz de incidencia de la Ecuación 4.4.

Es importante destacar, que para la simulación del comportamiento dinámico de la red LDPN el marcado inicial es con marca en los lugares correspondientes a las señales de los contactos NC, es decir, los lugares I_i^c , B_b^c y O_o^c .

4.3. Validación de la red LDPN

Un sistema basado en PLC puede tener N-número de entradas físicas en función de la complejidad del proceso industrial a controlar. Cada sub-sistema de entrada puede presentar falla de circuito-abierto o corto-circuito. Ahora bien, dos o más señales pueden presentar la misma falla o algunas señales estar en corto-circuito y otras señales en circuito-abierto, así como otras señales estar funcionando correctamente.

Considerando las fallas, ahora una señal de entrada física puede tener cuatro estados; activo, desactivo, en corto-circuito o en circuito-abierto. Considerando la red LDPN de la Figura 4.5, el marcado inicial M_0 en condiciones de funcionamiento normal es: $M_0 = [0002000]$ donde el lugar I_2 habilita la transición \mathbf{I}_2^c por el arco habilitador. Sin embargo, si uno de los lugares I_1 y/o I_2 presentan falla, el marcado inicial se ve afectado. Los marcados en condiciones de falla de una sola señal a la vez de la red LDPN a continuación se muestra, donde sc=1 (corto-circuito) y oc=0 (circuito-abierto):

$$M_0 = \begin{bmatrix} I_1 & I_1^o & I_2 & I_2^c & O_1 & O_1^o & G(\mathbf{L}_2)_{AND} \\ sc & 0 & 0 & 2 & 0 & 0 & 0 \\ oc & 0 & 0 & 2 & 0 & 0 & 0 \\ sc & 0 & 1 & 0 & 0 & 0 & 0 \\ oc & 0 & 1 & 2 & 0 & 0 & 0 \\ 0 & 0 & sc & 0 & 0 & 0 & 0 \\ 0 & 0 & oc & 2 & 0 & 0 & 0 \\ 1 & 0 & sc & 0 & 0 & 0 & 0 \\ 1 & 0 & oc & 2 & 0 & 0 & 0 \end{bmatrix}$$

La Ecuación 4.5 determina el número de posibles marcados iniciales en condiciones de falla Ft que pueden presentarse en el sub-sistema de entradas de un control basado en PLC, considerando el marcado inicial de funcionamiento normal.

$$Ft = 2^{N_I} * N_I \quad (4.5)$$

N_I = número de señales de entrada físicas.

Sin embargo, si se considera que la falla de corto-circuito o circuito-abierto puede presentarse en las señales de entradas físicas independientemente del estado que tenga esta señal o las demás señales, es decir, las fallas son señales asíncronas a la secuencia de las señales del proceso, y por consiguiente pueden presentarse en cualquier momento, lo que origina que las posibles combinaciones de posibles fallas se incrementen. Continuando con la red LDPN del ejemplo, se muestran las posibles combinaciones de los marcados iniciales en condiciones de falla.

$$M_0 = \begin{bmatrix} I_1 & I_1^o & I_2 & I_2^c & O_1 & O_1^o & G(\mathbf{L}_2)_{AND} \\ sc & 0 & 1 & 0 & 0 & 0 & 0 \\ oc & 0 & 1 & 0 & 0 & 0 & 0 \\ sc & 0 & 0 & 2 & 0 & 0 & 0 \\ oc & 0 & 0 & 2 & 0 & 0 & 0 \\ sc & 0 & sc & 0 & 0 & 0 & 0 \\ sc & 0 & oc & 2 & 0 & 0 & 0 \\ oc & 0 & sc & 0 & 0 & 0 & 0 \\ oc & 0 & oc & 2 & 0 & 0 & 0 \\ 0 & 0 & sc & 0 & 0 & 0 & 0 \\ 0 & 0 & oc & 2 & 0 & 0 & 0 \\ 1 & 0 & sc & 0 & 0 & 0 & 0 \\ 1 & 0 & oc & 2 & 0 & 0 & 0 \end{bmatrix}$$

La Ecuación 4.6 representa el número de posibles marcados iniciales de una red LDPN en condiciones de falla que pudieran darse en un momento dado.

$$Ft = 4^{N_I} - 2^{N_I} \quad (4.6)$$

Sea \mathbf{M}_{FT} el conjunto de marcados en condiciones de falla de la red LDPN. Cada marcado en condición de falla debe ser evaluado para determinar que lugares de salida se activan y tomar una decisión de solución para el algoritmo de control en LD. Es importante mencionar que el enfoque permite matemáticamente tener noción del comportamiento de un algoritmo de control en condiciones de falla de circuito-abierto o corto-circuito en el sub-sistema de entradas de un sistemas basado en PLC, sin embargo, el análisis y la determinación de que marcados en condición de falla se deben considerar en el algoritmo de control para evitar que se presente la combinación correspondiente, es el responsable del diseño del control del proceso industrial.

La ecuación de estado de la LDPN se muestra en la Ecuación 4.7.

$$M_{Ft_{k+1}} = M_{Ft_k} + aw_{ij}^T * u_k \quad (4.7)$$

Donde aw_{ij} es la matriz de incidencia de la LDPN, y u_k el vector de disparo, con $k = 1, 2, \dots, Ft$

Con base en las especificaciones formales de operación de los algoritmos de control aplicados en la red LDPN transformada, se puede determinar los marcados validos equivalentes a los estados activo o desactivo de los elementos (contactos NO, NC y bobinas) del LD. Utilizando la herramienta de árbol de alcanzabilidad, matriz de incidencia y la ecuación de estado de la LDPN.

Al evaluar los marcados \mathbf{M}_{Ft} originados por las posibles fallas de corto-circuito y circuito-abierto, se obtienen marcados en la red LDPN, los cuales tienen que ser analizados y evaluados para determinar que no sea un marcado de funcionamiento normal del algoritmo de control y si es considerado una situación de riesgo, entonces realizar la lógica que evite energizar o des-energizar considerando la falla o la combinación de fallas que dieron el marcado en cuestión.

La Figura 4.6 muestra el diagrama de flujo para el enfoque de validación de la red LDPN. Determinar el número de entradas físicas o lugares que los representan, elegir la Ecuación 4.5 o 4.6 y complementar los marcados de falla, el marcado inicial de los lugares B , O y G no se ven afectados, aplicar la ecuación de estado de la red LDPN, si existen lugares de salida $O = 1$ se debe analizar si es un marcado de funcionamiento normal o es un marcado a considerar en el algoritmo de control, si se concluyó de evaluar todos los marcados de falla, entonces se termina el proceso.

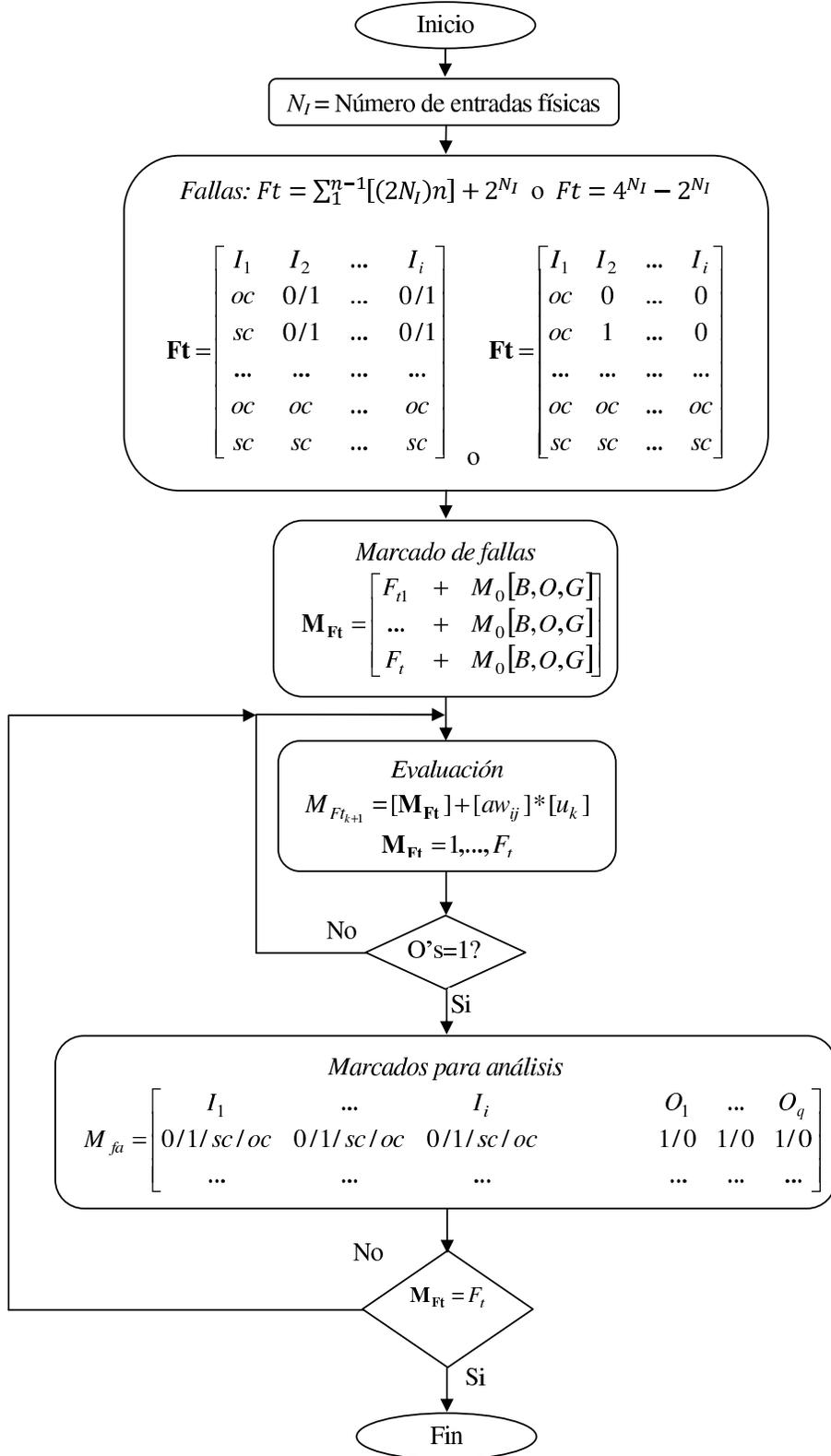


Figura 4.6: Diagrama de flujo del enfoque de validación de la red LDPN.

Para la presente investigación, se usa una Interfaz Humano Máquina (HMI, por sus siglas en inglés) para simular el comportamiento de la red LDPN de los ejemplos de algoritmos de control en el Capítulo 5. La HMI tiene la ventaja de estar en línea con el sistema basado en PLC y leer el estado de las variables desde la memoria interna del sistema.

4.4. Simulación de la red LDPN

Es fundamental, considerar el ciclo de trabajo de los algoritmos de control en los sistemas basados en PLC. En particular, la acción de copiar los estados de las señales de las estradas físicas para la evaluación del algoritmo. En un nuevo ciclo de trabajo se vuelven a leer los estados y se vuelve a evaluar el algoritmo de control, y así sucesivamente. Para simular el comportamiento dinámico de la red LDPN es necesario considerar el ciclo de trabajo, así como el problema de acumulación de marcas y el comportamiento de restablecer un lugar de salida. De forma general se enuncian los pormenores para la simulación de la red LDPN.

- I) Los lugares I_i^c , B_b^c y O_o^c correspondientes a contactos NC tiene marca de inicio, es decir, las transiciones habilitadas de inicio son \mathbf{I}_i^c , \mathbf{B}_b^c y \mathbf{O}_o^c respectivamente.
- II) Con base en la secuencia de del proceso, activar o desactivar sensores, colocar marca a los lugares de entrada respectivos I_i y disparar la transición de la distribución de señales \mathbf{I}_i^o o \mathbf{I}_i^c , considerar las Ecuaciones 3.1a - 3.1f. La primera activación de los sensores implica el marcado inicial de la red LDPN.
- III) Todas las transiciones habilitadas se deben disparar, para simular el comportamiento de un ciclo de trabajo del sistema basado en PLC.
- IV) La actualización del marcado debe realizarse cuando uno o más lugares de salida O_o de la LDPN, y gráficamente se debe considerar las Ecuaciones 3.3a, 3.3b, 3.4a, 3.4b, 3.5a y 3.5b para el marcado de los lugares O_o y B_b , lo anterior es la simulación de un nuevo ciclo de trabajo del sistema basado en PLC, en particular, la lectura y copia de los estados de las señales de entradas físicas. En la ecuación

de estado de la LDPN, se omite la actualización del marcado ya que no afecta el comportamiento de la red.

- V) Si las condiciones del marcado de los lugares de entrada a una transición \mathbf{L}_l han cambiado se debe considerar las Ecuaciones 3.6a, 3.6b, 3.7a, 3.7b, 3.8a y 3.8b para consumir la marca del lugar de salida correspondiente.

Para ejemplificar la simulación de la red LDPN, considere la red LDPN del arranque y paro de un motor, el lugar de entrada I_1 e I_2 representan señales de botones de accionamiento instantáneo, es decir, que la marca en ambos lugares solo está presente una vez cuando son activados. Se puede simular el comportamiento dinámico correspondiente para obtener los marcados de funcionamiento normal de forma gráfica y usando la ecuación de estados de PN, a continuación se muestra en las Figuras 4.7 - 5.4.

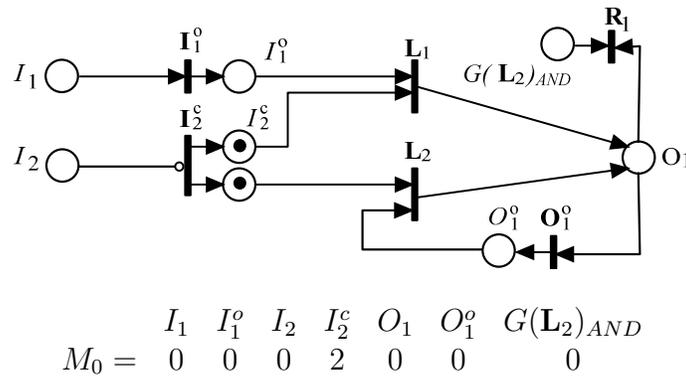


Figura 4.7: Lugares I_2^c con marca, LDPN arranque y paro de un motor.

Observar que con el marcado inicial M_0 no existe transición habilitada. En el algoritmo de control, sería necesario presionar el botón de arranque correspondiente a colocar una marca en el lugar de entrada I_1 , habilitando la transición \mathbf{I}_1^o . Este marcado en realidad es el marcado inicial para simular la red LDPN, Figura 4.8.

Si se dispara la transición \mathbf{I}_1^o , se coloca marca en el lugar I_1^o y se habilita la transición \mathbf{L}_1 , como muestra la Figura 4.9, observe que la actualización de marcas no afecta a la red LDPN.

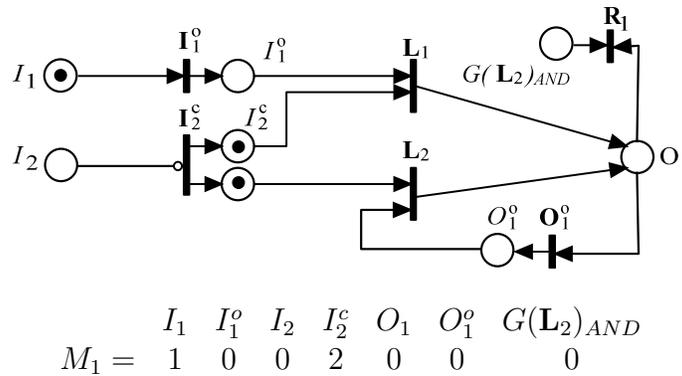


Figura 4.8: Marca en el lugar I_1 , equivalente al accionamiento del botón de arranque.

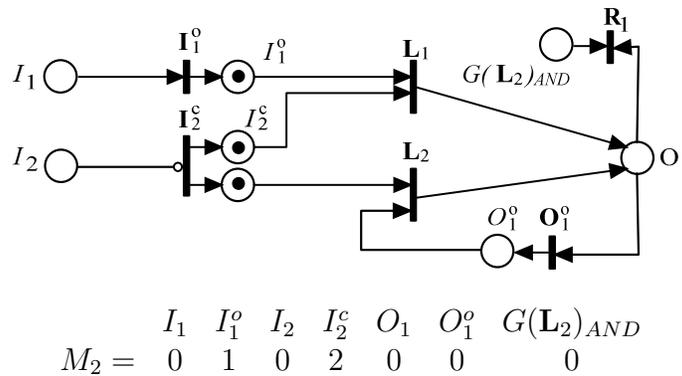
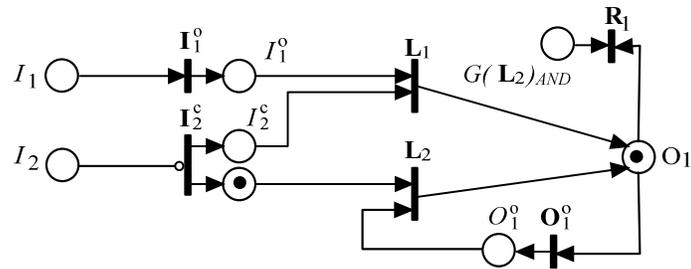


Figura 4.9: Disparo de la transición I_1^o .

Disparo de la transición habilitada L_1 , consume marca de los lugares I_1^o y I_2^c y colocación de marca en el lugar de salida O_1 y habilitación de la transición O_1^o , Figura 4.10.

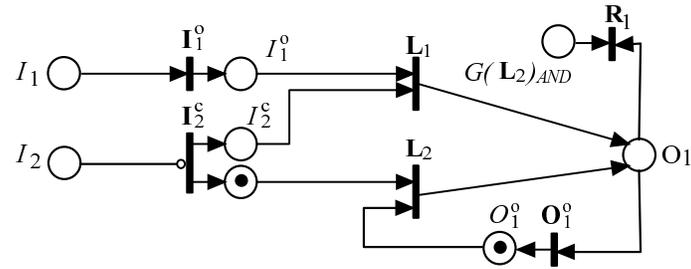
La Figura 5.1 muestra el disparo de la transición O_1^o y habilitado de la transición L_2 .

El disparo de la transición L_2 , consume la marca de los lugares I_2^c y O_1^o y coloca nuevamente marca en el lugar de salida O_1 , como muestra la Figura 5.2. La red LDPN en estas condiciones es equivalente a la retención o enclave de la bobina de salida en el algoritmo de control.



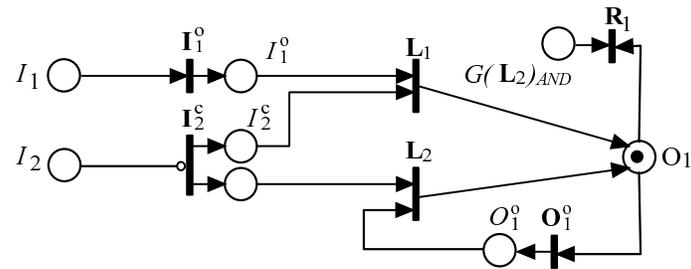
$$M_3 = \begin{matrix} & I_1 & I_1^o & I_2 & I_2^c & O_1 & O_1^o & G(\mathbf{L}_2)_{AND} \\ & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{matrix}$$

Figura 4.10: Disparo de la transición L_1 .



$$M_4 = \begin{matrix} & I_1 & I_1^o & I_2 & I_2^c & O_1 & O_1^o & G(\mathbf{L}_2)_{AND} \\ & 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{matrix}$$

Figura 4.11: Disparo de la transición O_1^o .



$$M_5 = \begin{matrix} & I_1 & I_1^o & I_2 & I_2^c & O_1 & O_1^o & G(\mathbf{L}_2)_{AND} \\ & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{matrix}$$

Figura 4.12: Disparo de la transición L_2 .

La actualización del estado de señales habilita la transición I_2^c , al dispararse coloca marca en los lugares I_2^c , como muestra la Figura 5.3.

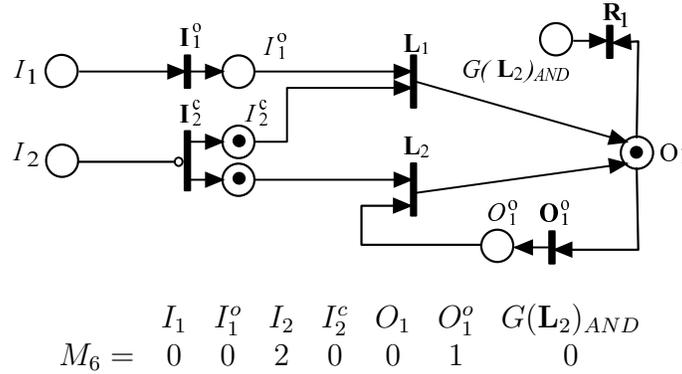


Figura 4.13: Actualización del marcado y disparo de la transición I_2^c .

Si es presionado el botón de paro correspondiente al lugar I_2 en la red LDPN, se deshabilita la transición I_2^c y por la Ecuación 3.1b se drenan las marcas de los lugares I_2^c y se coloca marca en el lugar de restablecimiento $G(\mathbf{L}_2)_{AND}$ para consumir la marca del lugar de salida O_1 , equivalente a des-energizar la bobina en el algoritmo de control, Figura 5.4.

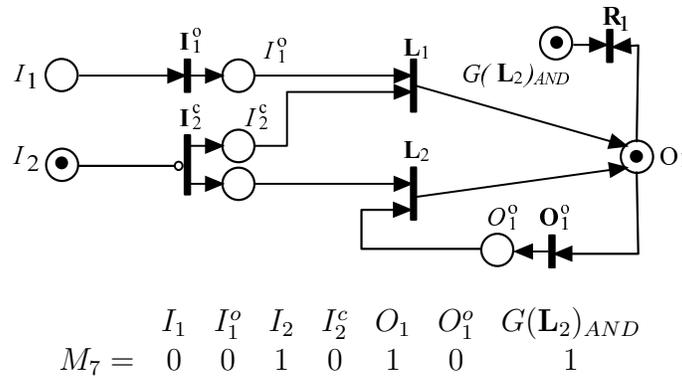


Figura 4.14: Inhabilitado de la transición I_2^c , marca en lugar de restablecimiento $G(\mathbf{L}_2)_{AND}$.

4.5. Conclusiones del capítulo

El diseño de algoritmos de control en LD se conforma de las especificaciones formales del proceso industrial y de la experiencia de los responsables del control industrial, lo que conlleva a la posibilidad de no contemplar situaciones de riesgo en el proceso. Todo sistema basado en PLC tiene la posibilidad de presentar fallas de corto-circuito o circuito-abierto en los sub-sistemas de las señales de entradas y salidas físicas, lo cual genera que las señales correspondientes no cambien de estado, ocasionando comportamientos no deseados en el algoritmo de control y por ende en el procesos industrial.

La red LDPN permite validar el comportamiento de los algoritmos de control en LD en condiciones de falla de corto-circuito y circuito-abierto en el sub-sistema de entradas de los sistemas basados en PLC. El resultado de la evaluación de cada marcado de falla debe ser analizado para tomar la decisión de considerar o no la corrección en el algoritmo de control. Es importante resaltar la necesidad de personal calificado que conozca el proceso lo suficiente para tomar las mejores decisiones en cuanto a que combinaciones de fallas en las entradas se deben evitar.

La distribución de señales propuesta permite reducir la matriz de incidencia de la LDPN al considerar la suma de los pesos de los arcos que salen de una transición a lugares del mismo tipo.

La simulación de la red LDPN se puede realizar gráficamente y utilizando la ecuación de estado, para lo cual, se deben considerar las ecuaciones de habilitado del marcado en lugares de salida, de los lugares de restablecimiento y de la distribución de señales.

Capítulo 5

Casos de estudio

Para demostrar la utilidad del enfoque propuesto, se analizan tres algoritmos de control en LD de tres procesos industriales: control del arranque y paro de un motor reversible, control de un sistema de auto lavado y control de bascula de envasado.

La simulación se realiza en tiempo real a través de interfaces gráficas desarrolladas en un HMI que lee los estados de las variables de entradas y salidas físicas, así como las de memoria. En las interfaces se implementa la lógica de cada una de las marcas en los lugares de las respectivas redes LDPN de los sistemas analizados.

5.1. Sistema de control de motor reversible

De [58] se considera el algoritmo de control de un sistema de motor reversible, con las siguientes especificaciones formales y con las variables mostradas en la Tabla 5.1:

Las especificaciones formales son:

- 1.- El motor debe girar para ambos lados, giro hacia la derecha y hacia la izquierda,
- 2.- El sistema tiene un botón de paro, independiente del giro del motor,
- 3.- Cada arranque de giro del motor es independiente, es decir, el sistema debe tener dos botones; uno para arranque del motor con giro hacia la derecha y otro para arranque con giro hacia la izquierda,

- 4.- Si el motor está girando hacia la derecha, el sistema debe evitar el arranque hacia el giro izquierdo, y viceversa.

Tabla 5.1: Variables físicas de sistema de control de motor reversible.

Variable	Tipo	Descripción
I_0	Entrada física	Botón instantáneo para paro de motor reversible.
I_1	Entrada física	Botón instantáneo para arranque de motor, giro hacia la derecha.
I_2	Entrada física	Botón instantáneo para arranque de motor, giro hacia la izquierda.
O_0	Salida física	Bobina de salida para arranque de motor, giro hacia la derecha.
O_1	Salida física	Bobina de salida para arranque de motor, giro hacia la izquierda.

5.1.1. Algoritmo del sistema de control de motor reversible

El algoritmo de control en LD del sistema de control de motor reversible se muestra en la Figura 5.1, el cual tiene dos líneas de control, ambas tiene lógica AND-OR y auto-lazo con la bobina de salida O_0 y O_1 , además, contactos de bloqueo de O_1 y I_2 para la primer línea y los contactos O_0 y I_1 para la segunda línea. Ambas líneas tiene un contacto de la señal de paro de motor I_0 .

Puede observarse que el algoritmo de control está solo en función de las señales de entrada y salida físicas, es decir, no contempla señales internas o de memoria del sistema basado en PLC. De igual forma, existe doble redundancia de bloqueo en cada línea de control con la señal de arranque y la bobina para cada sentido de giro del motor.

5.1.2. Red LDPN del sistema de control de motor reversible

La Figura 5.2 muestra la transformación del algoritmo de control en LD del sistema de control de motor reversible de la Figura 5.1 a la red LDPN. La Matriz de incidencia aw_{ij} de la LDPN del sistema de control de motor reversible se muestra en la Ecuación 5.1.

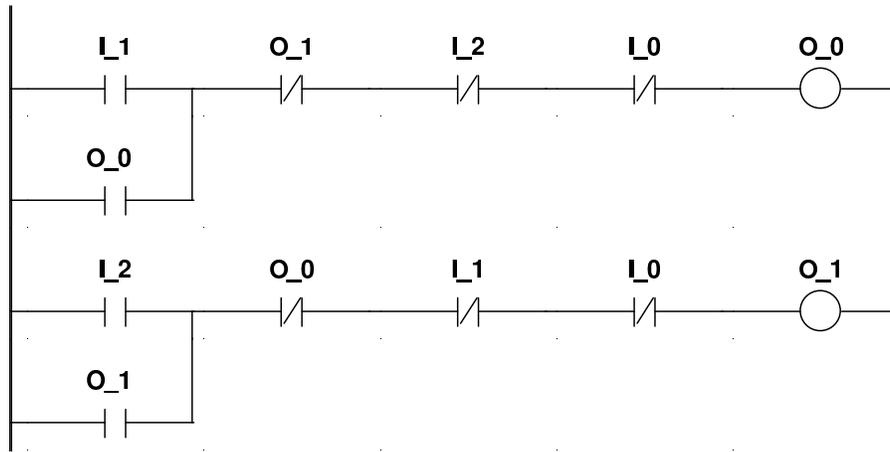


Figura 5.1: Algoritmo de control en LD de motor reversible.

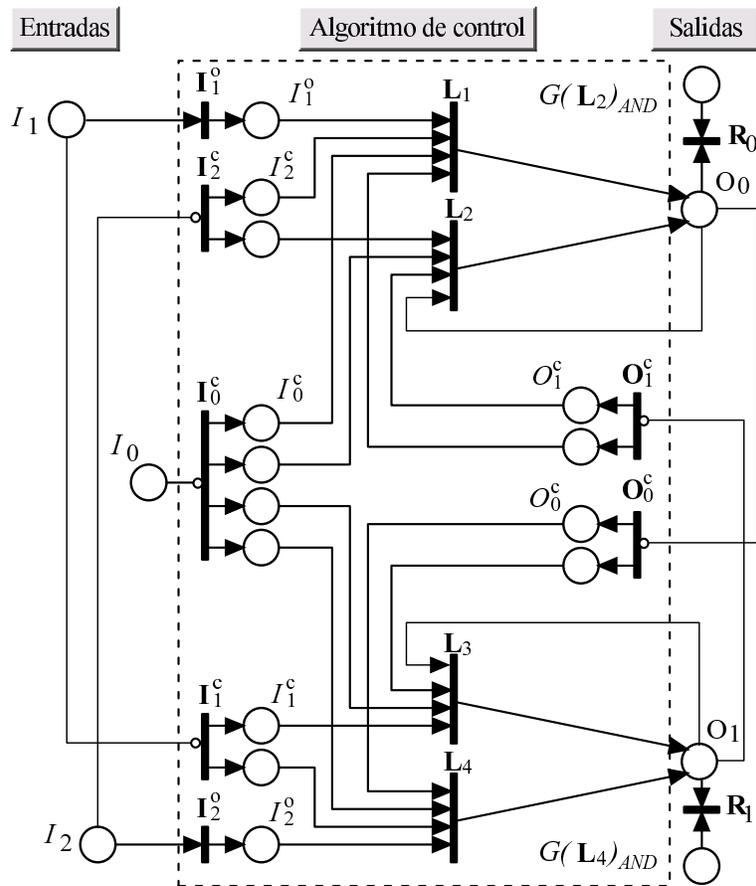


Figura 5.2: LDPN de sistema de control de motor reversible.

$$aw_{ij} = \begin{bmatrix} & I_0 & I_0^c & I_1 & I_1^o & I_1^c & I_2 & I_2^o & I_2^c & O_0 & O_0^c & O_1 & O_1^c & G(\mathbf{B}_2^o)_{AND} & G(\mathbf{B}_4^o)_{AND} \\ I_1^o & -1 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ I_1^c & 0 & 0 & -1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ I_2^o & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ O_0^c & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & 0 & 0 & 0 & 0 \\ O_1^c & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & 0 & 0 \\ I_1^o & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ I_2^o & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{B}_2^o & 0 & -1 & 0 & -1 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & -1 & 0 & 0 \\ \mathbf{B}_3^o & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & -1 & 0 & 0 \\ \mathbf{B}_4^o & 0 & -1 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ \mathbf{R}_0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 \\ \mathbf{R}_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \quad (5.1)$$

5.1.3. Simulación de la red LDPN del sistema de control de motor reversible

La Figura 5.3 muestra en a) el marcado de los lugares I_0^c , I_1^c , I_2^c , O_0^c y O_1^c correspondientes a los contactos NC de las señales I_0 , I_1 , I_2 , O_0 y O_1 del algoritmo de control en LD.

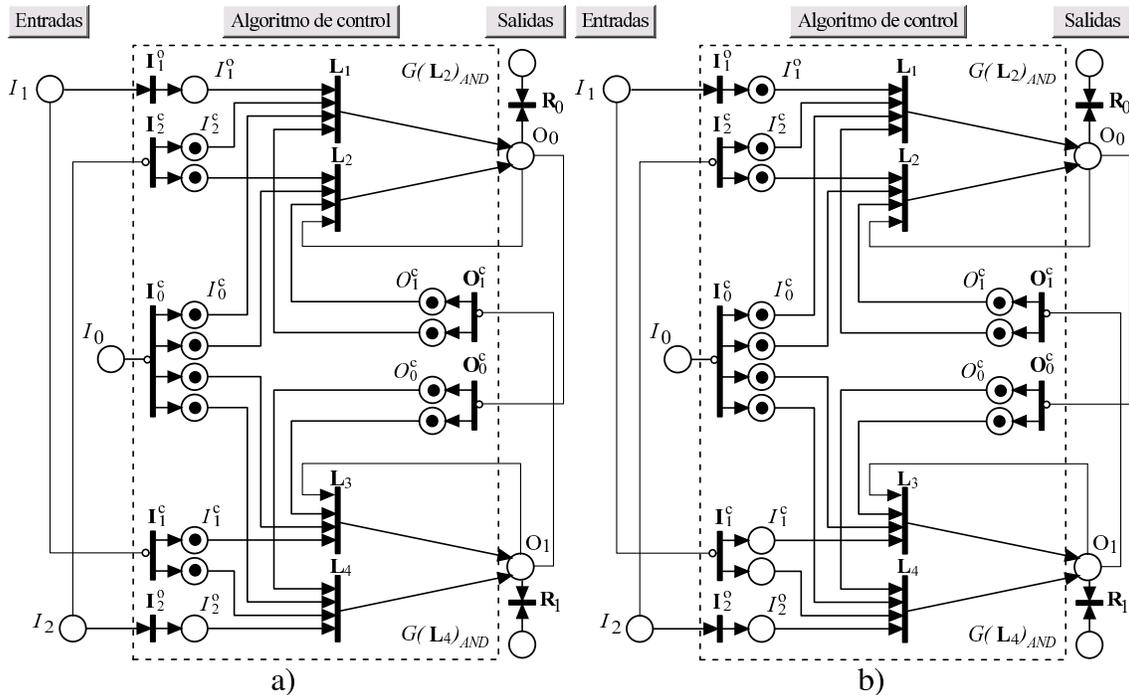


Figura 5.3: LDPN de sistema de control de motor reversible, a) y b).

En b) se modela la activación del botón de arranque del motor con giro hacia la derecha, se coloca marca en el lugar I_1 lo que habilitó a la transición I_1^o , al dispararse colocó marca en el lugar I_1^c , habilitando la transición L_1 . También, la marca en el lugar I_1 deshabilitó la transición I_1^c y por la Ecuación 3.1b se drenaron las marcas en los lugares I_1^c .

La Figura 5.4 muestra en c) el marcado de la red LDPN después de disparar la transición L_1 . Además, la marca en el lugar de salida O_0 inhabilita la transición O_0^c y las marcas de los lugares O_0^c se drenan por la Ecuación 3.1d. En d) muestra el marcado después de disparar la transición L_2 . Hasta aquí, es el modelado de un ciclo de trabajo del algoritmo de control del sistema de control de motor reversible. Puede observarse en el marcado de la LDPN, que aunque se activará la señal de arranque del motor con giro a la izquierda, la marca en el lugar I_2^c no habilita la transición L_4 por falta de marca en el lugar O_0^c , la cual representa en el algoritmo de control el contacto NC en la configuración de bloqueo de la señal O_0 en la línea de control dos.

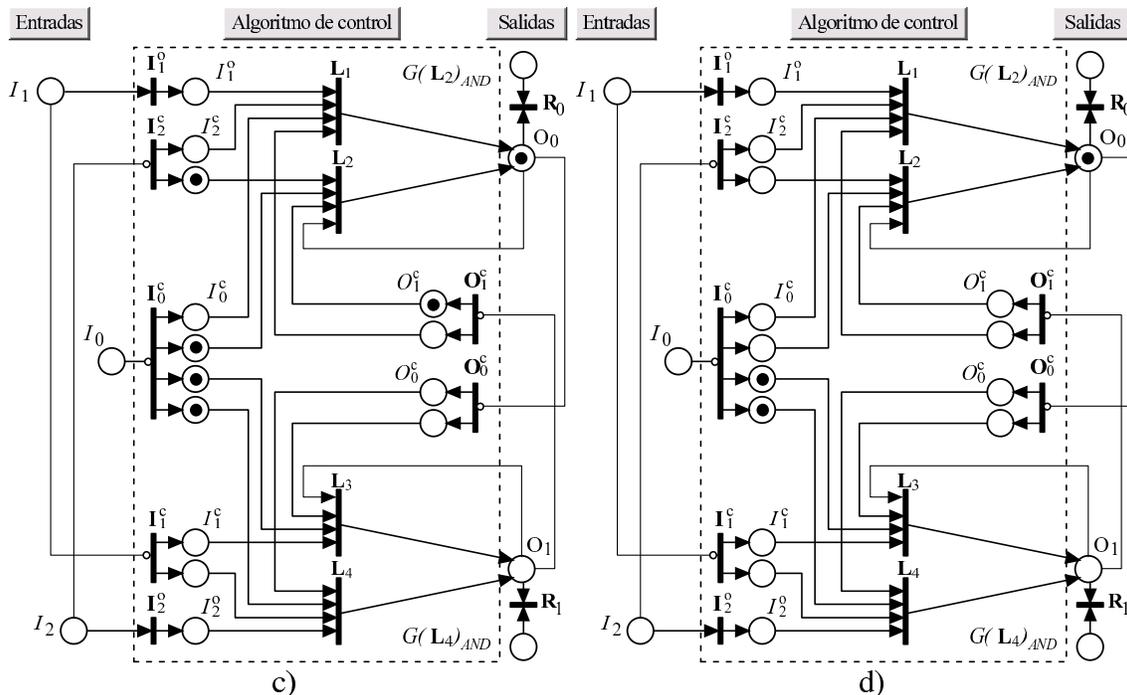


Figura 5.4: LDPN de sistema de control de motor reversible, c) y d).

La actualización del marcado de la LDPN se muestra en la Figura 5.5 e), equivalente a un nuevo ciclo de trabajo del sistema basado en PLC. La Ecuación 3.3a evita la acumulación de marca del lugar de salida O_0 . Ahora bien en f) se muestra la condición de cuando se activa la señal de paro de motor I_0 del algoritmo de control, equivalente a que el lugar I_0 tenga marca, lo cual inhabilita la transición I_0^c y el drenado de las marcas de los lugares I_0^o por la Ecuación 3.1b, además de habilitar el marcado en el lugar $G(L_2)_{AND}$ para consumir la marca del lugar de salida O_0 y restablecer las condiciones de la LDPN.

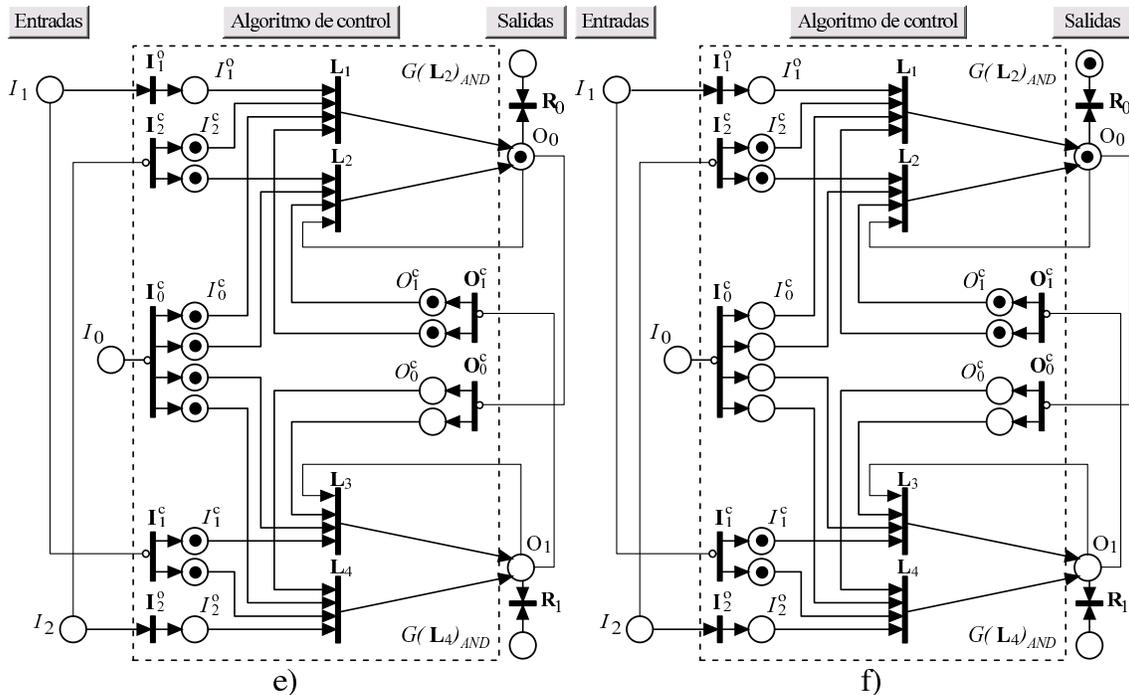


Figura 5.5: LDPN de sistema de control de motor reversible, e) y f).

Con el consumo de la marca en el lugar de salida O_0 en la red LDPN del sistema control de motor reversible, en una nueva actualización del marcado se podría simular el comportamiento correspondiente a arrancar el motor con giro a la izquierda, el cual es similar al descrito para el arranque del motor con giro a la derecha.

5.1.4. Validación de la red LDPN del sistema de control de motor reversible

Para evaluar las posibles condiciones de falla de corto-circuito y circuito-abierto para las tres señales de entrada I_0 , I_1 y I_2 del algoritmo de control del sistema de control de motor reversible, se considera la Ecuación 4.6.

$$Ft = 4^3 - 2^3 = 56 \quad (5.2)$$

Por la poca cantidad de señales de entrada, se muestra el conjunto de posibles combinaciones de fallas que deben ser evaluados para analizar el comportamiento de la red LDPN equivalente al algoritmo de control en LD del sistema de control de motor reversible.

$$\mathbf{M}_{Ft} = \left[\begin{array}{ccc|ccc|ccc|ccc|ccc|ccc} I_0 & I_1 & I_2 & I_0 & I_1 & I_2 \\ 0 & 0 & oc & 0 & 0 & sc & 0 & 1 & oc & 0 & 1 & sc & 0 & oc & 0 \\ 0 & oc & 1 & 0 & oc & oc & 0 & oc & sc & 0 & sc & 0 & 0 & sc & 1 \\ 0 & sc & oc & 0 & sc & sc & 1 & 0 & oc & 1 & 0 & sc & 1 & 1 & oc \\ 1 & 1 & sc & 1 & oc & 0 & 1 & oc & 1 & 1 & oc & oc & 1 & oc & sc \\ 1 & sc & 0 & 1 & sc & 1 & 1 & sc & oc & 1 & sc & sc & oc & 0 & 0 \\ oc & 0 & 1 & oc & 0 & oc & oc & 0 & sc & oc & 1 & 0 & oc & 1 & 1 \\ oc & 1 & oc & oc & 1 & sc & oc & oc & 0 & oc & oc & 1 & oc & oc & oc \\ oc & oc & sc & oc & sc & 0 & oc & sc & 1 & oc & sc & oc & oc & sc & sc \\ sc & 0 & 0 & sc & 0 & 1 & sc & 0 & oc & sc & 0 & sc & sc & 1 & 0 \\ sc & 1 & 1 & sc & 1 & oc & sc & 1 & sc & sc & oc & 0 & sc & oc & 1 \\ sc & oc & oc & sc & oc & sc & sc & sc & 0 & sc & sc & 1 & sc & sc & oc \\ sc & sc & sc & & & & & & & & & & & & \end{array} \right]$$

Con base en la matriz de incidencia de la Ecuación 5.1 y aplicando la ecuación de estados de la LDPN 4.7 se pueden obtener los marcados de la LDPN en condición de falla, considerando las posibles combinaciones de Ft .

Las condiciones críticas del sistema de control de motor reversibles son cuando la señal de paro I_0 queda en circuito-abierto, ya que si el motor es puesto en marcha, no se podría detener por control del algoritmo. Normalmente un paro de emergencia cubriría esta debilidad del algoritmo, al reducir la posibilidad que dos señales queden

en corto-circuito o circuito-abierto a la vez.

5.2. Sistema de control de auto-lavado de vehículos

De [59] se considera el ejemplo del control automático de un tren de lavado de coches como muestra la Figura 5.6.

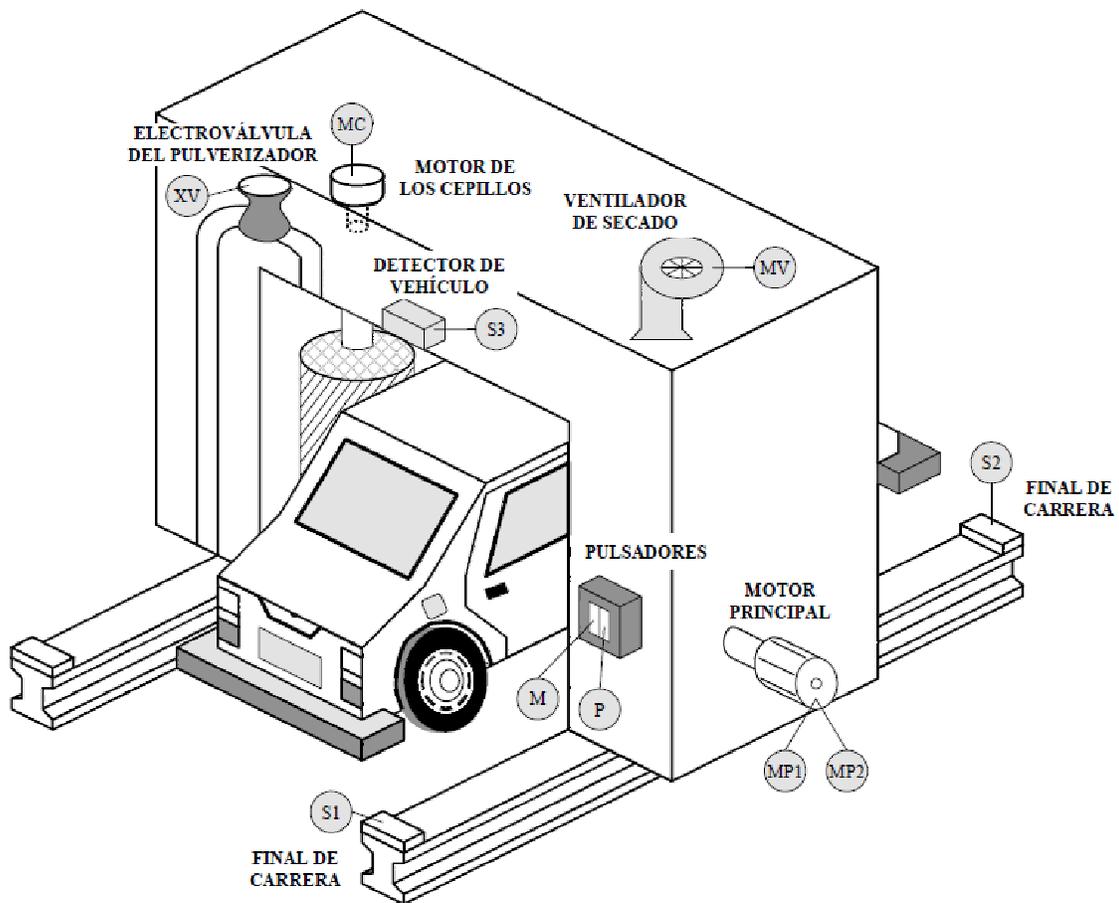


Figura 5.6: Sistema de control de auto-lavado de vehículos.

Las especificaciones formales del sistema de auto-lavado de vehículos son:

- 1.- El estado inicial de la máquina es en el extremo derecho (S2 activado),
- 2.- Carro en posición de lavado (S3 activado), presionar el botón instantáneo M para iniciar funcionamiento del sistema,
- 3.- Máquina debe hacer un recorrido de ida y vuelta con las electro-válvula XV y el motor de los cepillos MC funcionando,
- 4.- Cuando la máquina regresa al extremo derecho (S2 se vuelve a activar) debe realizar otro recorrido de ida y vuelta en el que sólo el ventilador MV está en marcha. Finalizado el recorrido la máquina queda en su estado inicial.
- 5.- Si se acciona el botón instantáneo P de paro, la máquina debe regresar automáticamente a su posición inicial.

La Tabla 5.2 muestra la asignación de variables del sistema de control de auto-lavado de vehículos.

Tabla 5.2: Variables físicas de sistema de control de auto-lavado de vehículos.

Variable	Tipo	Símbolo	Descripción
<i>I</i> _0	Entrada física	S1	Interruptor de límite para detener la máquina en el extremo derecho del riel.
<i>I</i> _1	Entrada física	S2	Interruptor de límite para detener la máquina en el extremo izquierdo del riel.
<i>I</i> _2	Entrada física	S3	Sensor magnético para detección de vehículo.
<i>I</i> _3	Entrada física	M	Botón instantáneo para inicio de funcionamiento de la máquina.
<i>I</i> _4	Entrada física	P	Botón instantáneo para paro de la máquina.
<i>O</i> _0	Salida física	MP1	Bobina de salida para arranque de motor, giro hacia la derecha de la máquina.
<i>O</i> _1	Salida física	MP2	Bobina de salida para arranque de motor, giro hacia la izquierda de la máquina.
<i>O</i> _2	Salida física	MV	Bobina de salida para arranque de motor del ventilador de secado.
<i>O</i> _3	Salida física	MC	Bobina de salida para arranque de motor de los cepillos de tallado.
<i>O</i> _4	Salida física	XV	Bobina de salida para energizar la solenoide para atomizar líquido de lavado.

5.2.1. Algoritmo de control del sistema de auto-lavado de vehículos

Las Figuras 5.7 y 5.8 muestran el algoritmo de control del sistema de auto-lavado de vehículos. Puede observarse el comportamiento de ejecución secuencial entre la bobina de salida *M00* de la línea uno con las líneas dos, tres, cuatro, siete y nueve. Existen combinaciones de lógicas AND - OR y auto-lazos en las líneas de control uno, dos, tres cuatro y seis. Las líneas de control cinco y nueve representan la lógica AND.

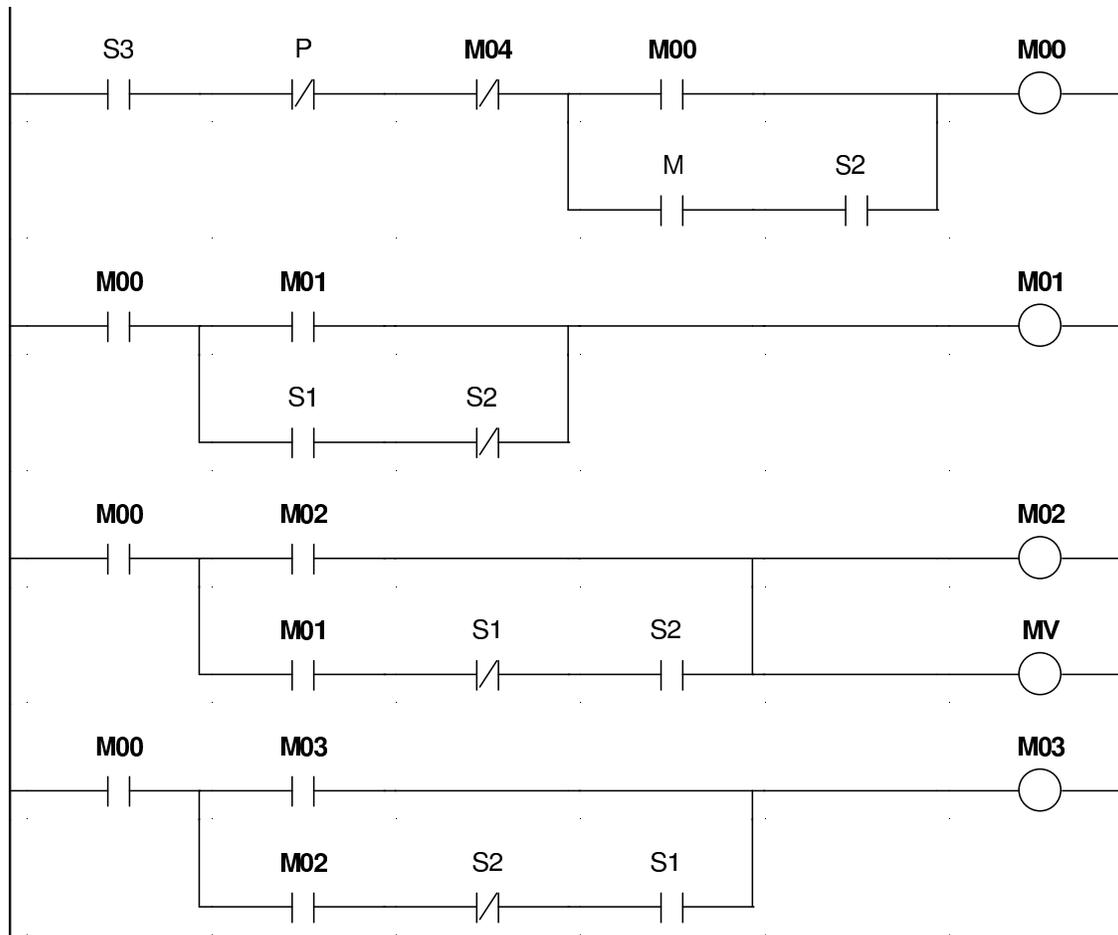


Figura 5.7: Algoritmo de control de sistema de auto-lavado de vehículos, parte 1.

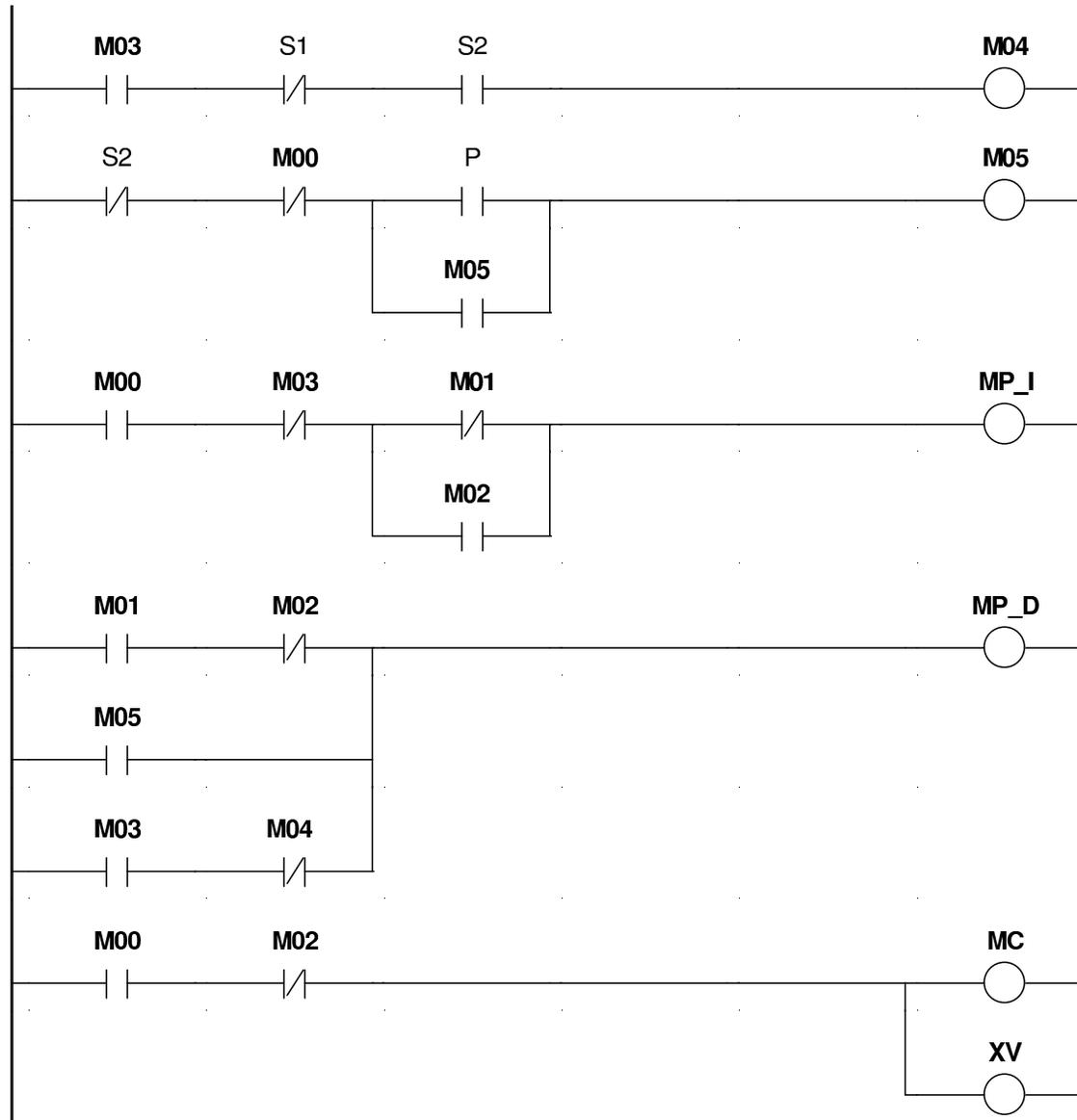


Figura 5.8: Algoritmo de control de sistema de auto-lavado de vehículos, parte 2.

5.2.2. Red LDPN del sistema de control de auto-lavado de vehículos

La Figura 5.9 muestra la transformación del algoritmo de control en LD del sistema de control de auto-lavado de vehículos a la red LDPN y la Figura 5.10 es la matriz de incidencia correspondiente, se omiten las transiciones \mathbf{R} y los lugares de restablecimiento G . La matriz de incidencia es de 38×39 columnas por renglones.

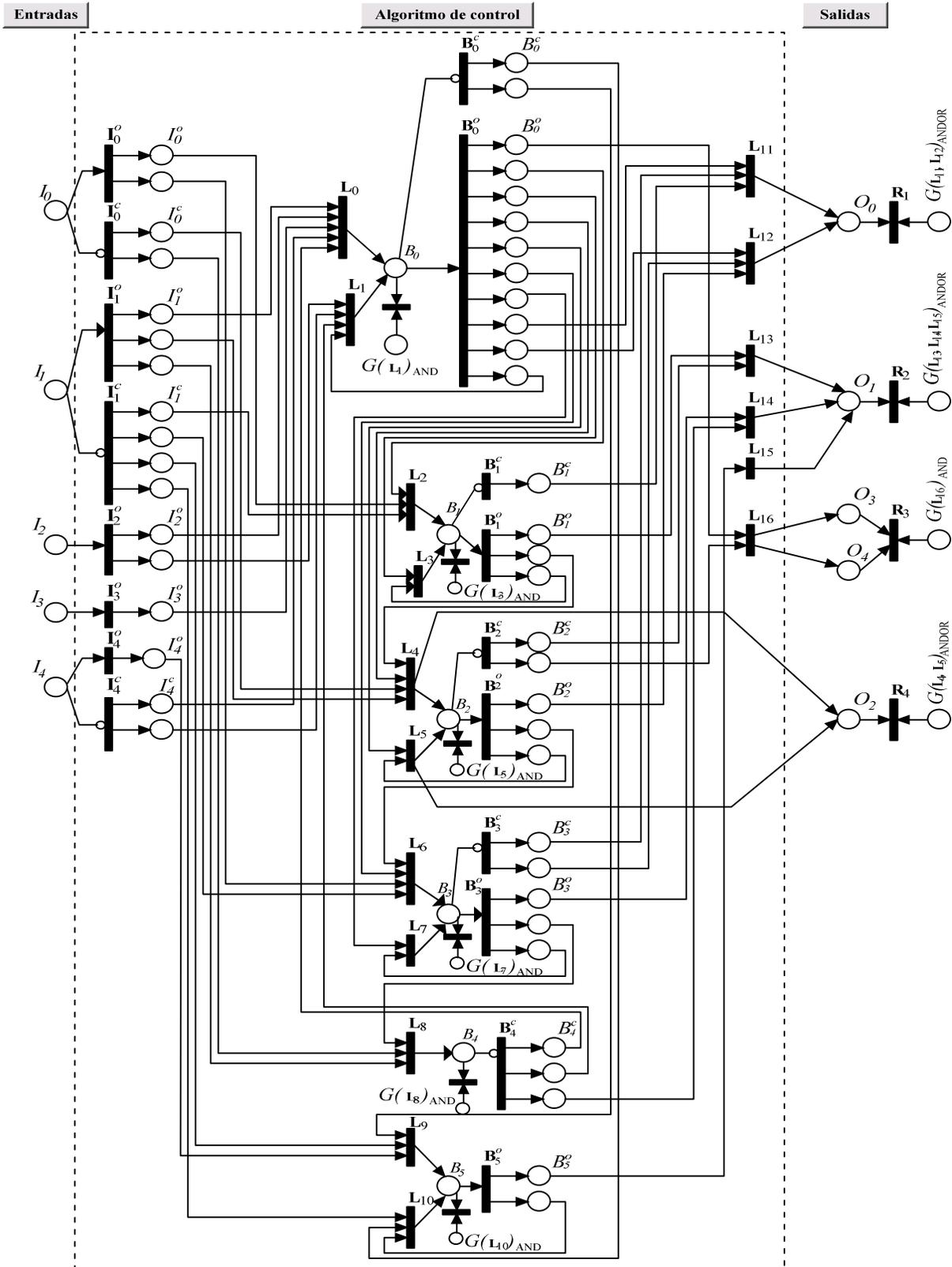


Figura 5.9: LDPN del sistema de control de auto-lavado de vehículos.

T/P	I0	I1	I2	I3	I4	I00	I0C	I10	I1C	I20	I30	I40	I4C	B0	B1	B2	B3	B4	B5	B0C	B00	B1C	B10	B2C	B20	B3C	B30	B4C	B50	Q0	Q1	Q2	Q3	Q4			
I00	-1	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
I0C	-1	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
I10	0	-1	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
I1C	0	-1	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
I20	0	0	-1	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
I30	0	0	0	-1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
I40	0	0	0	0	-1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
I4C	0	0	0	0	-1	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
L0	0	0	0	0	0	0	0	-1	0	-1	-1	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	
L1	0	0	0	0	0	0	0	0	0	-1	0	0	-1	1	0	0	0	0	0	0	0	-1	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	
L2	0	0	0	0	0	-1	0	0	-1	0	0	0	0	0	1	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
L3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	-1	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	
L4	0	0	0	0	0	0	-1	-1	0	0	0	0	0	0	0	1	0	0	0	0	0	-1	0	-1	0	0	0	0	0	0	0	0	0	1	0	0	0
L5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	-1	0	0	0	-1	0	0	0	0	0	0	0	1	0	0	0
L6	0	0	0	0	0	-1	0	0	-1	0	0	0	0	0	0	0	1	0	0	0	0	-1	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0
L7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	-1	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0
L8	0	0	0	0	0	-1	-1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0
L9	0	0	0	0	0	0	0	0	-1	0	0	-1	0	0	0	0	0	0	1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
L10	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	1	-1	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0
L11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	0	0	0	-1	0	0	0	1	0	0	0	0	0	
L12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	-1	-1	0	0	0	1	0	0	0	0	0	
L13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	0	0	0	0	0	0	1	0	0	0	0	0	
L14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	0	0	1	0	0	0	0	0	
L15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	1	0	0	0	0	0	
L16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	-1	0	0	0	0	0	0	0	0	0	0	1	1	
B0C	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
B00	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
B1C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
B10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	
B2C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	
B20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	
B3C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	
B30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	
B4C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	
B50	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	

Figura 5.10: Matriz de incidencia de la LDPN del auto-lavado de vehículos.

5.2.3. Simulación de la red LDPN del sistema de control de auto-lavado de vehículos

La Figura 5.11 muestra el marcado de la red LDPN del sistema de control de auto-lavado de vehículos, solo considerando los estados correspondientes a los contactos NO de las señales I , B y O del algoritmo de control. Como todo proceso industrial, se requiere de la secuencia de activación o des-activación de sensores para el funcionamiento del sistema.

El sistema requiere que la máquina esté posicionada en el lado izquierdo del riel y presencia de vehículo, es decir, $S2 = 1$ y $S3 = 1$, equivalente a colocar marca en los lugares $I_0^o = 1$ e $I_2^o = 1$ al disparar las transiciones habilitadas \mathbf{I}_0^o e \mathbf{I}_2^o respectivamente. El marcado de la red LDPN en estas condiciones se muestra en la Figura 5.12, y sería el marcado inicial M_0 de operación de la red.

Solo falta presionar el botón instantáneo $M = 1$ para dar inicio al lavado del automóvil, en la red LDPN es colocar una marca en el lugar I_3 , disparando la transición \mathbf{I}_3^o se coloca marca en el lugar $I_3^o = 1$ y se habilita la transición \mathbf{L}_0 , al disparar coloca marca en el lugar B_0 , lo que origina que la transición \mathbf{B}_0^c se deshabilite y por la Ecuación 3.1f se drenan las marca de los lugares $B_0^c = 0$, se habilita la transición \mathbf{B}_0^o , al dispararla coloca marca en los lugares $B_0^o = 1$, como muestra la Figura 5.13.

En la Figura 5.13 puede observarse que están habilitadas las transiciones \mathbf{L}_1 , que es para hacer la función de enclave o retención de la bobina $M00 = 1$ en la línea uno de control y lugar B_0 en la LDPN; la transición \mathbf{L}_11 para la función de energizar la bobina $MP.D = 1$ correspondiente a poner en marcha el motor con giro a la derecha, lugar de salida $O_0 = 1$; y la transición \mathbf{L}_16 para energizar las bobinas $MC = 1$ y $XV = 1$ del motor de los cepillos y la solenoide de la electro-válvula de liquido para lavado, en la LDPN los lugares $O_3 = 1$ y $O_4 = 1$ respectivamente. El disparo de estas transiciones generan el nuevo marcado en la red LDPN como se muestra en la Figura 5.14, y corresponde al primer ciclo de trabajo del sistema basado en PLC.

Al empezar a funcionar la máquina hacia el lado derecho, se desactiva el interruptor de límite $S2 = 0$, por lo que la marca en los lugares $I_1^o = 0$ se drenan y disparando la transición habilitada \mathbf{I}_1^c se coloca marca en los lugares $I_1^c = 1$. Realizando la actualización del marcado y considerando que la máquina llega al extremo derecho del riel, donde se activa el interruptor de límite de lado derecho $S1 = 1$, con lo que se habilitan las transición \mathbf{L}_2 , al dispararse coloca marca en el lugar $B_1 = 1$, lo cual inhabilita la

transición \mathbf{B}_1^c y se drena la marca de los lugares $B_1^c = 0$ lo que inhabilita la transición \mathbf{L}_{11} , y activando el marcado del lugar de restablecimiento $G(\mathbf{L}_{11}, \mathbf{L}_{12})_{ANDOR} = 1$ por la Ecuación 3.8a, equivalente a des-energizar la bobina $MP_I = 0$ y deteniendo el giro hacia la izquierda del motor. El marcado correspondiente se muestra en la Figura 5.15

Ahora bien, disparando la transición \mathbf{B}_1^o se coloca marca en los lugares $B_1^o = 1$ y se habilita la transición \mathbf{L}_{13} , si es disparada se coloca marca en el lugar de salida $O_1 = 1$, equivalente a energizar la bobina $MP_I = 1$ para poner en marcha el motor con giro hacia la izquierda. Los lugares de salida $O_3 = 1$ y $O_4 = 1$ continúan con marca y se evita el problema de acumulación de marcas por la Ecuación ec:001, equivalente a continuar energizado el motor de los cepillos y la solenoide de la electro-válvula de liquido para lavar en el algoritmo de control en LD. Este marcado se muestra en la Figura 5.16, donde además se considera el enclave de la marca del lugar B_1 , equivalente a un segundo ciclo de trabajo del sistema basado en PLC.

En la red LDPN de sistema de auto-lavado de vehículos se actualiza el marcado. La máquina se mueve hacia la izquierda del riel, se desactiva el interruptor de límite $S1 = 0$, por lo que la marca en los lugares $I_0^o = 0$ se drenan, y si se dispara la transición habilitada \mathbf{I}_0^c se coloca marca en los lugares $I_0^c = 1$. La maquina continua su recorrido hasta activar nuevamente el interruptor de límite de lado izquierdo $S2 = 1$, deteniendo el motor de giro hacia la izquierda, en la LDPN se drena los lugares de $I_1^c = 0$, se habilita la transición \mathbf{I}_1^o , al dispararse se coloca marca en los lugares $I_1^o = 1$ lo que habilita la transición \mathbf{L}_4 , si se dispara coloca marca en el lugar $B_2 = 0$, lo que inhabilita la transición \mathbf{B}_2^c y se drena la marca de los lugares de $B_2^c = 0$ lo que inhabilita a las transiciones \mathbf{L}_{13} y \mathbf{L}_{16} lo que activa el marcado de los lugares de restablecimiento $G(\mathbf{L}_{13}, \mathbf{L}_{14}, \mathbf{L}_{15})_{ANDOR} = 1$ y $G(\mathbf{L}_{16})_{AND} = 1$ respectivamente. La transición \mathbf{L}_4 también coloca marca en el lugar de salida $O_2 = 1$ equivalente a energizar la bobina $MC = 1$ del algoritmo de control. La Figura 5.17 muestra el marcado en estas condiciones. La acumulación de marcas en los lugares B_0 y B_1 se evitan con la Ecuación 3.3a.

Disparando la transición habilitada \mathbf{B}_2^o se coloca marca en los lugares $B_2^o = 1$ habilitando la transición \mathbf{L}_{12} , si se dispara coloca marca en el lugar de salida $O_0 = 1$ equivalente a energizar nuevamente la bobina $MP_D = 1$ en el algoritmo el control. De igual forma, con marca en los lugares $B_2^o = 1$ habilita la transición \mathbf{L}_5 , la cual al dispararse mantiene marca en lugar $B_2 = 1$ y $O_2 = 1$. En estas condiciones la máquina se mueve hacia la derecha con el ventilador encendido equivalente a un tercer ciclo de trabajo del sistema basado en PLC. La Figura 5.18 muestra el marcado correspondiente a este comportamiento.

La máquina al moverse hacia la derecha se desactiva el interruptor de límite $S2 = 0$, equivalente a drenar la marca de los lugares $I_1^o = 0$ y colocar marca en los lugares $I_1^c = 1$ al disparar la transición \mathbf{I}_1^c . Al terminar su recorrido la máquina se activa el interruptor de límite $S1 = 1$, equivalente a drenar la marca de los lugares $I_0^c = 0$ y colocar marca en los lugares $I_0^o = 1$ al disparar la transición \mathbf{I}_0^o , con base en la Ecuación ???. El nuevo marcado habilita la transición \mathbf{L}_6 , al dispararse coloca marca en el lugar $B_3 = 1$, lo que inhabilita la transición \mathbf{B}_3^c drenando la marca de los lugares B_3^c lo que habilita el marcado del lugar de restablecimiento $G(\mathbf{L}_{11}, \mathbf{L}_{12})_{ANDOR} = 1$. La Figura 5.19 muestra el marcado correspondiente.

Disparando la transición habilitada \mathbf{B}_3^o se coloca marca en los lugares $B_3^o = 1$ lo que habilita la transición \mathbf{L}_{14} , al dispararse coloca marca en el lugar de salida $O_1 = 1$, equivalente nuevamente a energizar la bobina $MP_I = 1$ en algoritmo de control en LD y poner en marcha el motor de giro hacia la izquierda. La marca en el lugar de salida $O_2 = 1$ se mantiene por las condiciones de marcado en los lugares de entrada de la transición \mathbf{L}_5 . El marcado correspondiente se muestra en la Figura 5.20 y corresponde a un cuarto ciclo de trabajo del sistema basado en PLC.

Nuevamente, al desplazarse la máquina hacia la izquierda se desactiva el interruptor de límite $S1 = 0$ por lo que en la red LDPN se drena los lugares $I_0^o = 0$ y se coloca marca en los lugares $I_0^c = 1$ al disparar la transición \mathbf{I}_1^c . La máquina al llegar al extremo izquierdo del riel activa de nueva cuenta el interruptor de límite $S2 = 1$ lo que en la red LDPN se ve reflejado en el drenado de la marca de los lugares $I_1^c = 0$ y la colocación de marca en los lugares $I_1^o = 1$ al disparar la transición \mathbf{I}_1^o . Estas condiciones deshabilitan a las transiciones \mathbf{L}_1 y \mathbf{L}_{14} lo que habilita el marcado en los lugares de restablecimiento $G(\mathbf{L})_{AND}$ y $G(\mathbf{L}_{13}, \mathbf{L}_{13}, \mathbf{L}_{13})_{ANDOR}$ respectivamente, como muestra la Figura 5.21.

La marca en el lugar $B_0 = 1$ inhabilita la transición \mathbf{B}_0^o drenando las marcas de los lugares $B_0^o = 0$ lo que habilita el marcado de los lugares de restablecimiento de $G(\mathbf{L}_3)_{AND} = 1$, $G(\mathbf{L}_5)_{AND} = 1$ y $G(\mathbf{L}_7)_{AND} = 1$, como muestra la Figura ???. Al consumir la marca de los lugares $B_1 = 0$, $B_2 = 0$ y $B_3 = 0$ se drenan las marcas de los lugares $B_1^o = 0$, $B_2^o = 0$ y $B_3^o = 0$ lo que genera que se habilite el marcado de lugar de restablecimiento $G(\mathbf{L}_8)$, al consumir la marca del lugar $B_4 = 0$ se restablece el marcado la red LDPN del sistemas de control de auto-lavado de vehículos en posición de máquina en el extremo izquierdo del riel, como al Figura 5.12.

Una condición especial del sistema de control de auto-lavado de vehículos es cuando

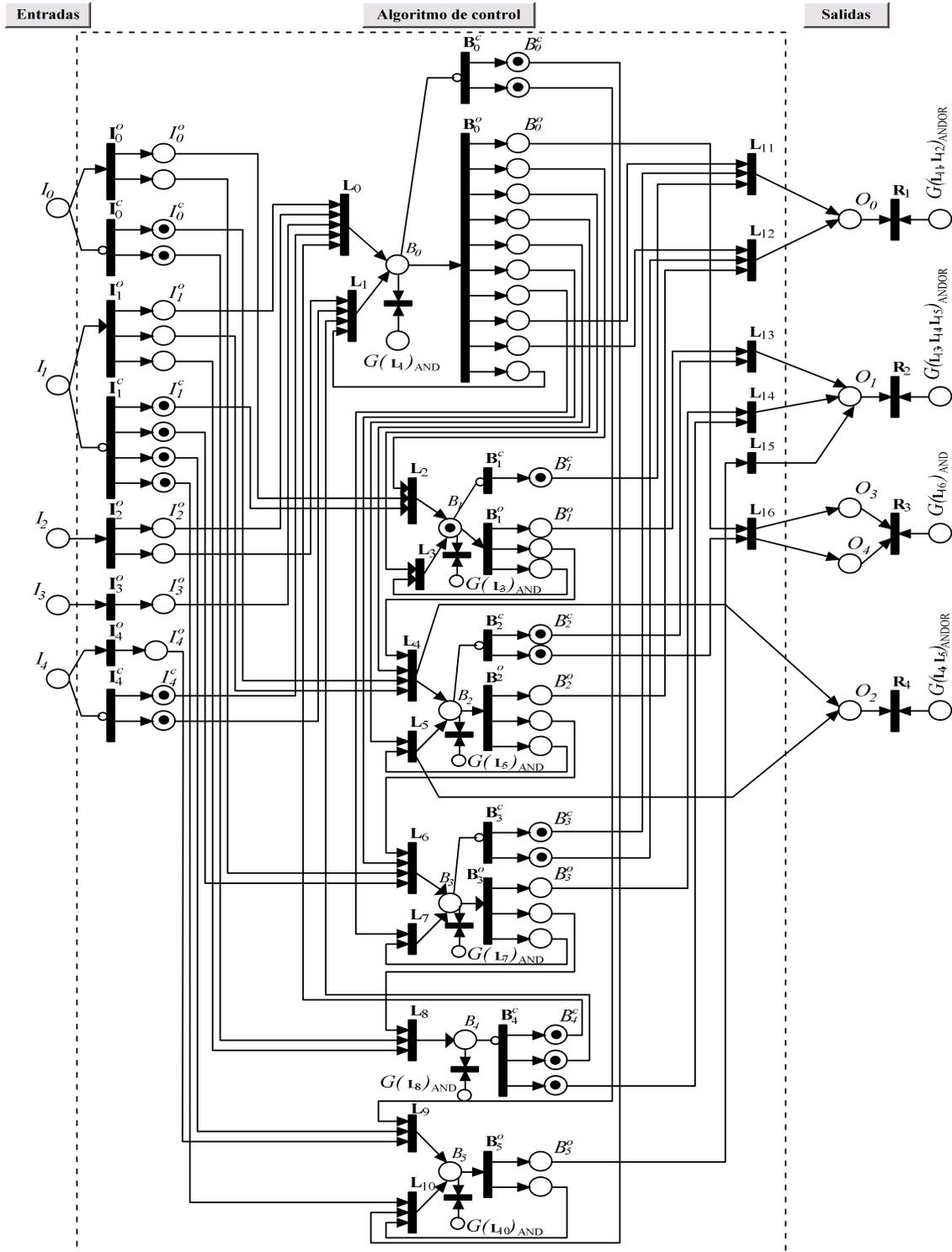


Figura 5.11: LDPN del sistema de control de auto-lavado de vehículos, marca en los lugares I_i^c y B_i^c .

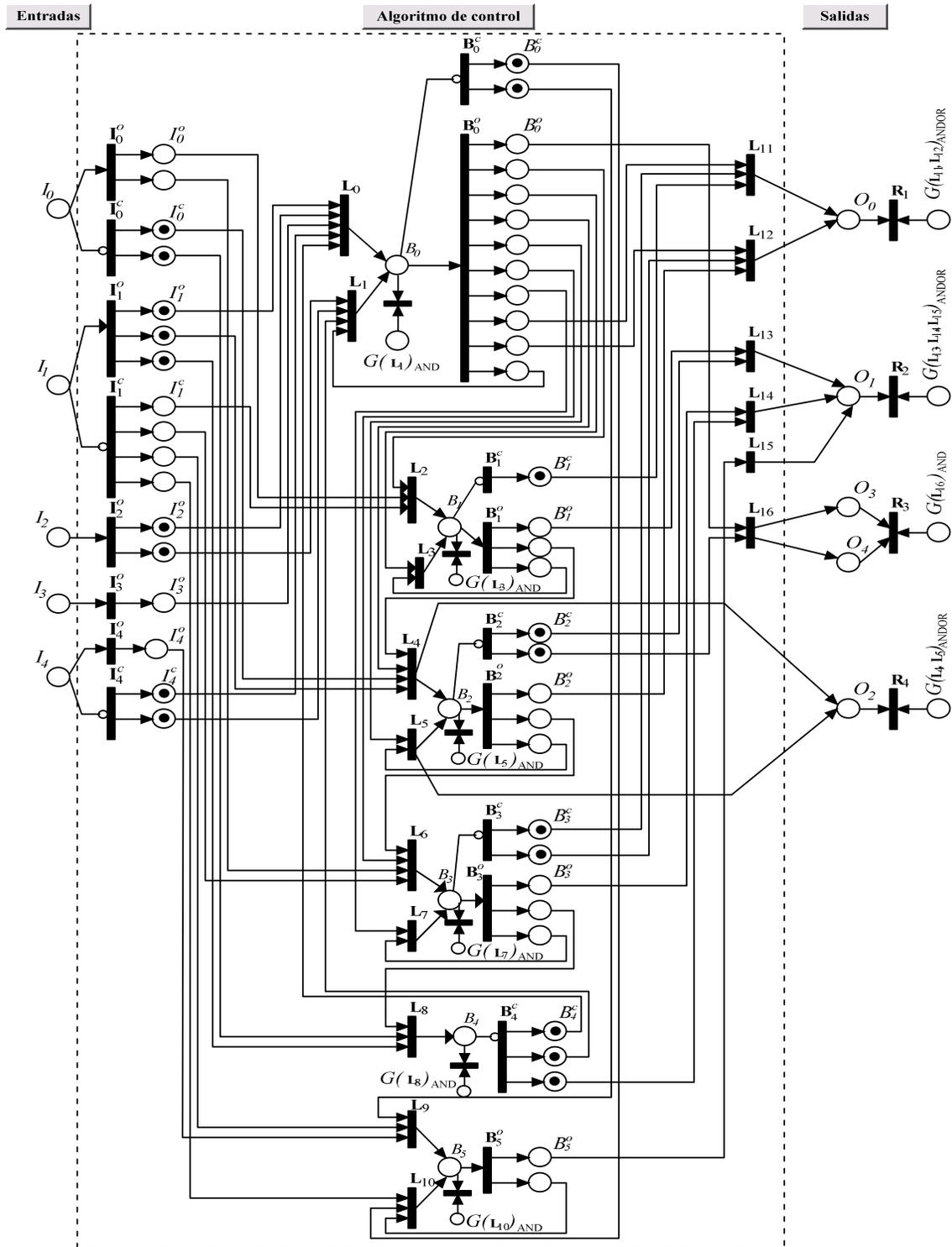


Figura 5.12: LDPN del sistema de control de auto-lavado de vehículos, M_0 .

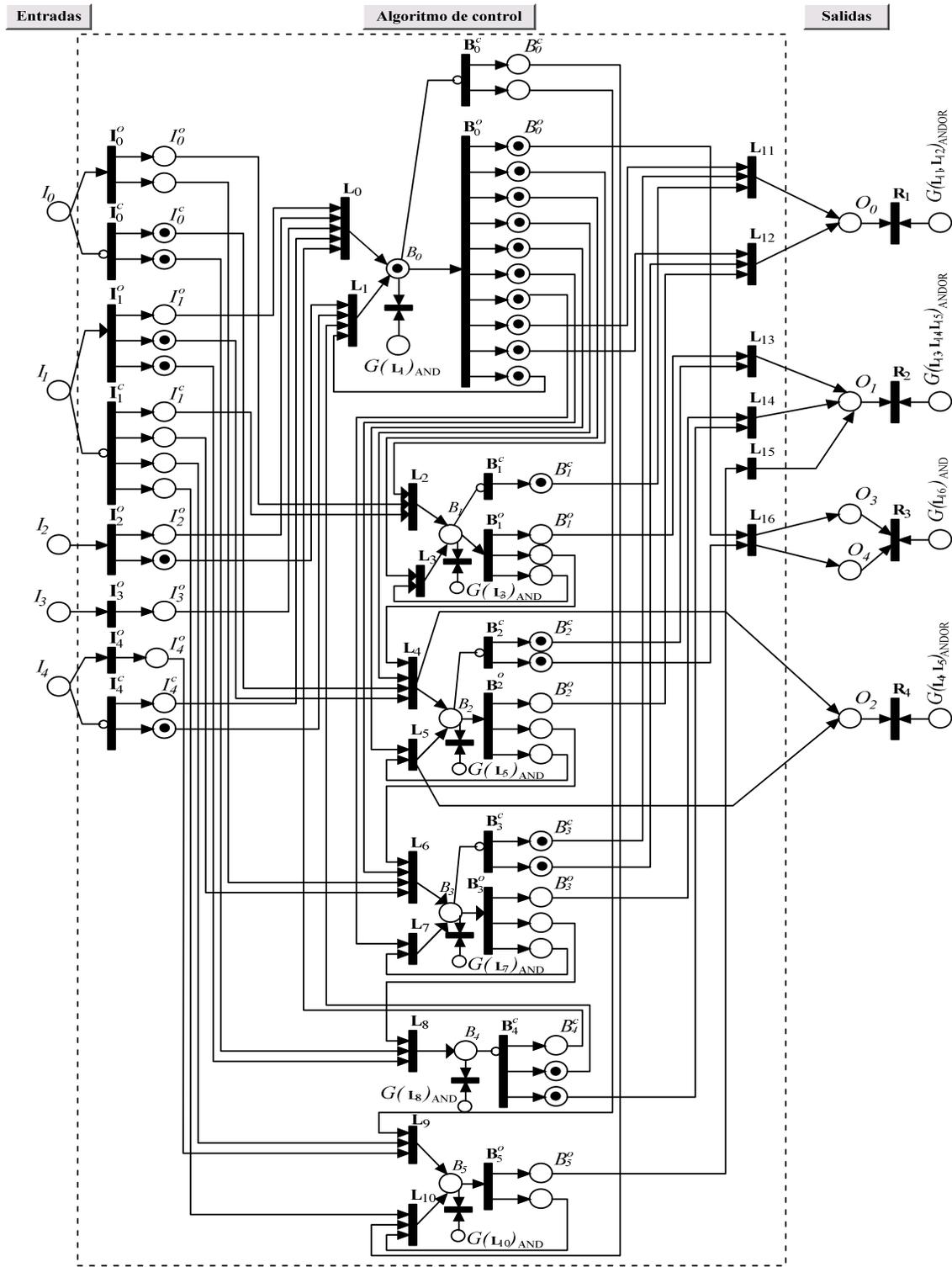


Figura 5.13: LDPN del sistema de control de auto-lavado de vehículos, disparo de transición B_0^o .

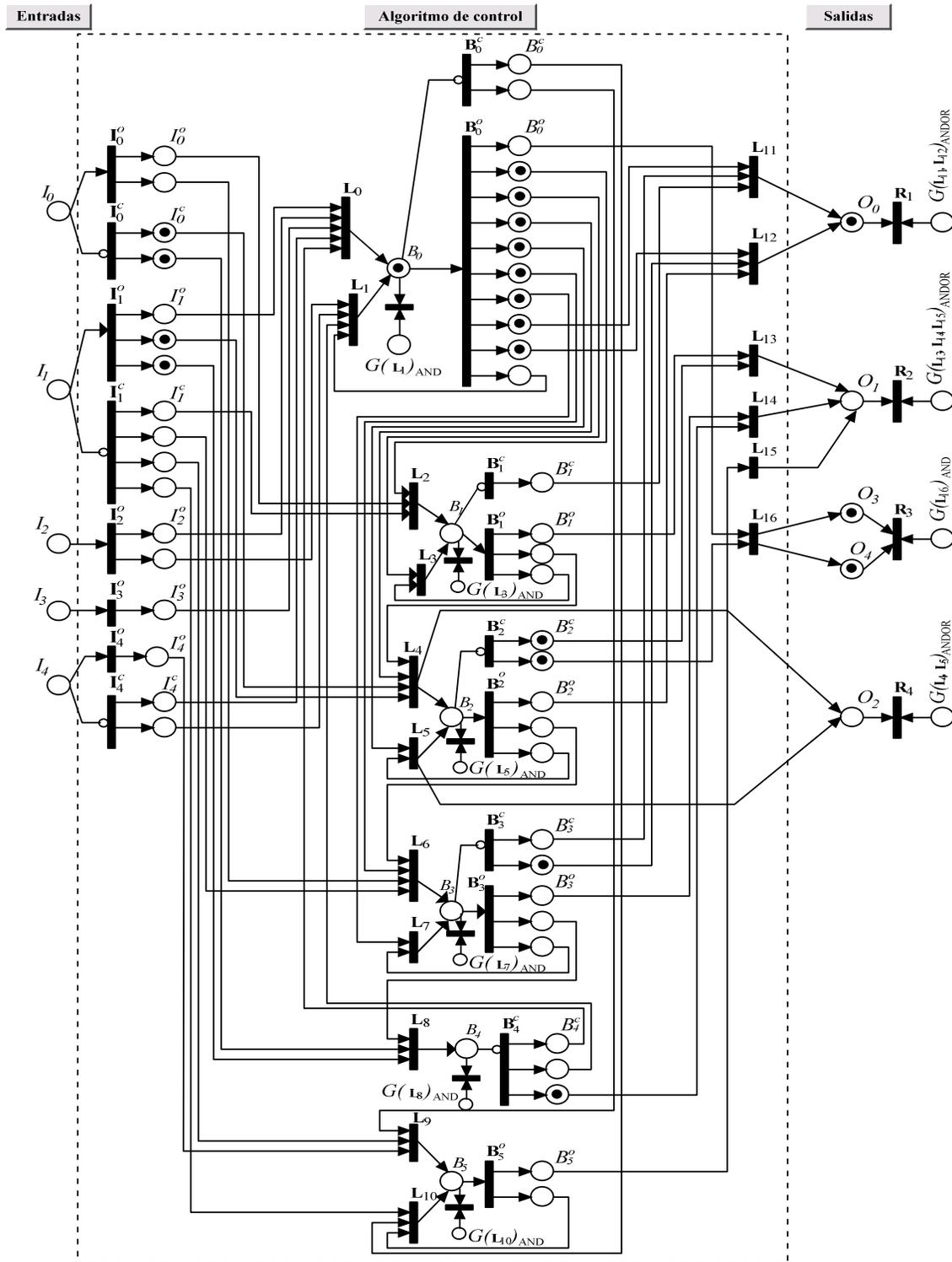


Figura 5.14: LDPN del sistema de control de auto-lavado de vehículos, primer ciclo de trabajo del PLC.

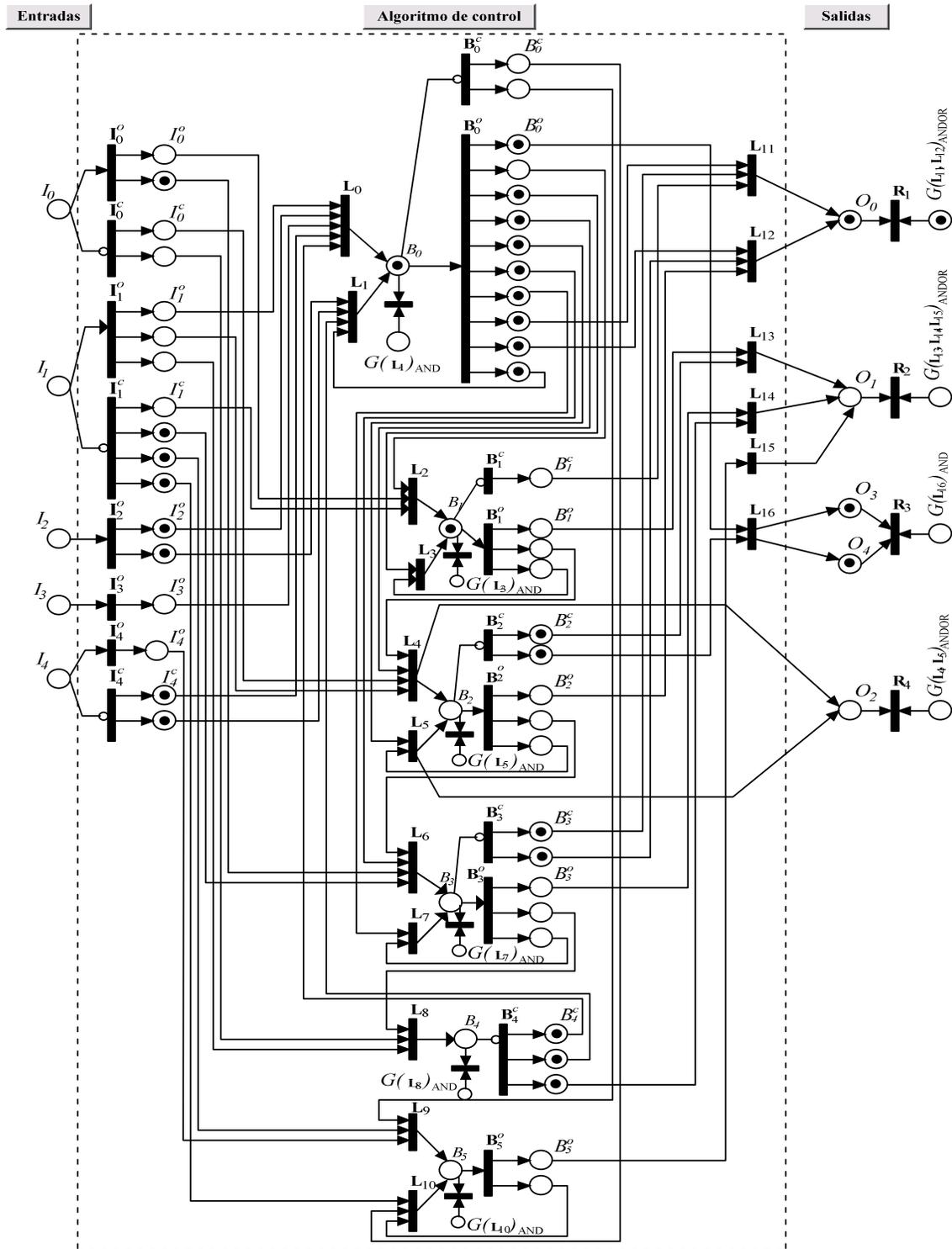


Figura 5.15: LDPN del sistema de control de auto-lavado de vehículos, actualización del marcado, S1=1.

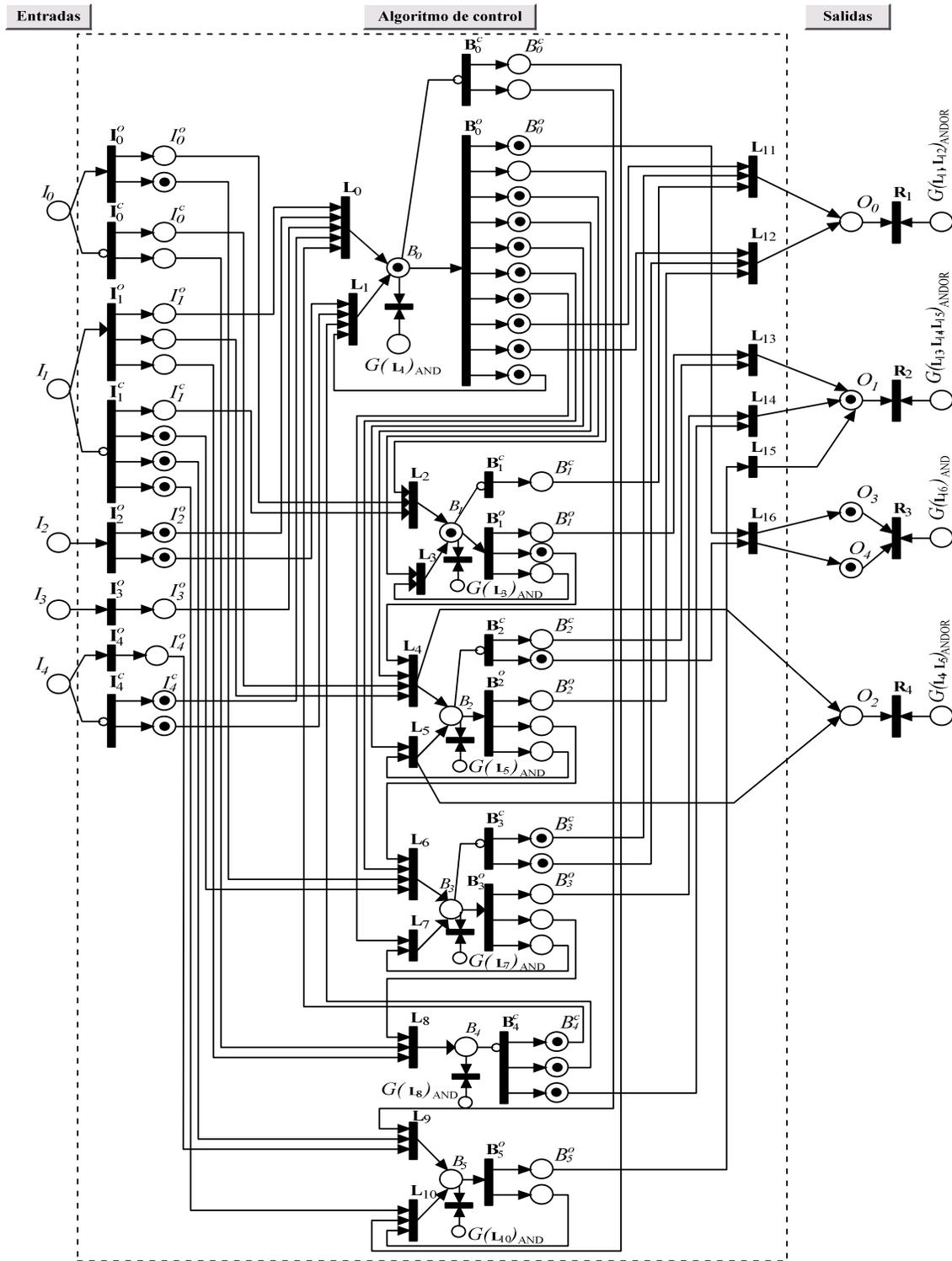


Figura 5.16: LDPN del sistema de control de auto-lavado de vehículos, segundo ciclo de trabajo del PLC.

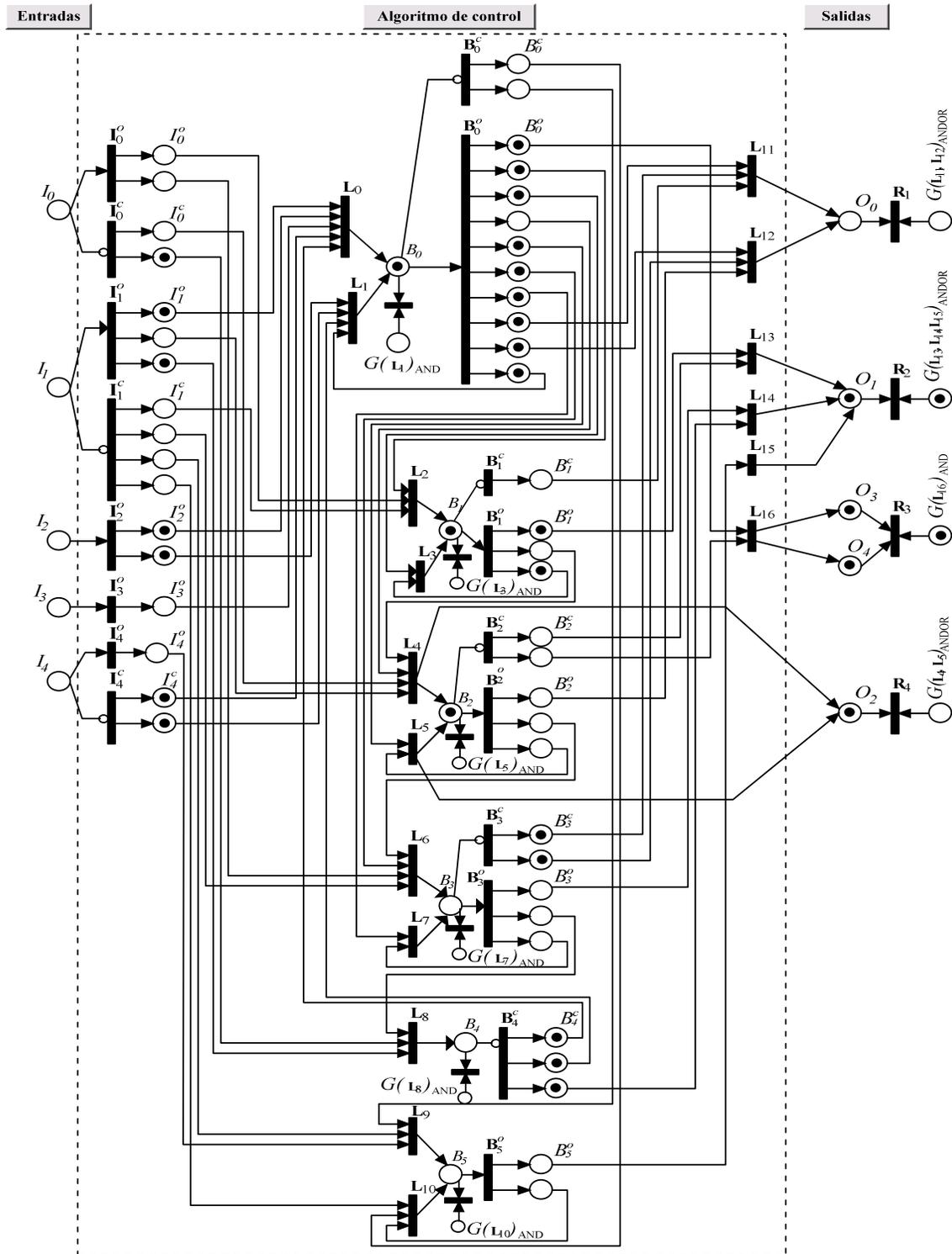


Figura 5.17: LDPN del sistema de control de auto-lavado de vehículos, actualización del marcado, S2=1.

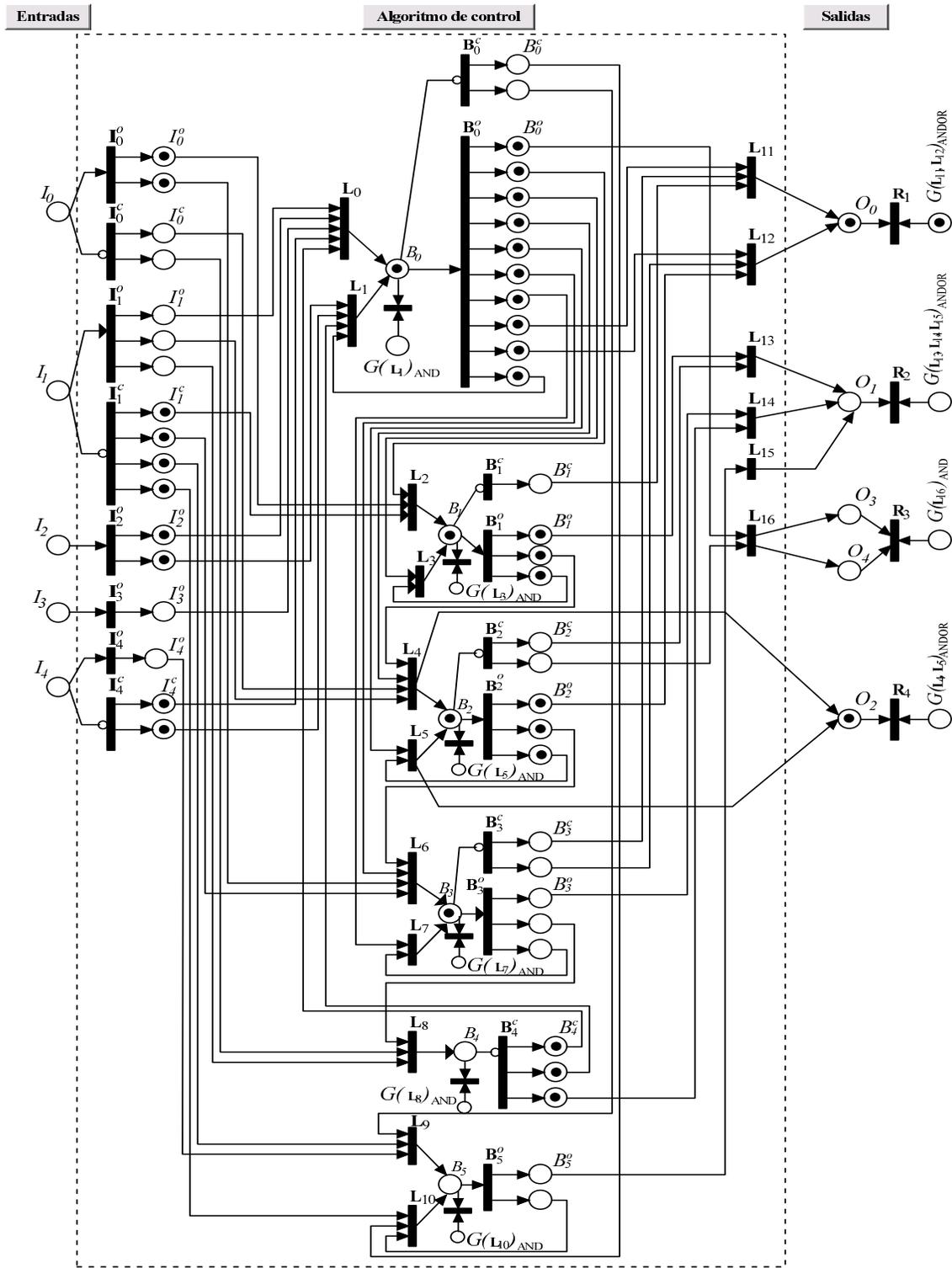


Figura 5.19: LDPN del sistema de control de auto-lavado de vehículos, actualización del marcado, segunda vez de S1=1.

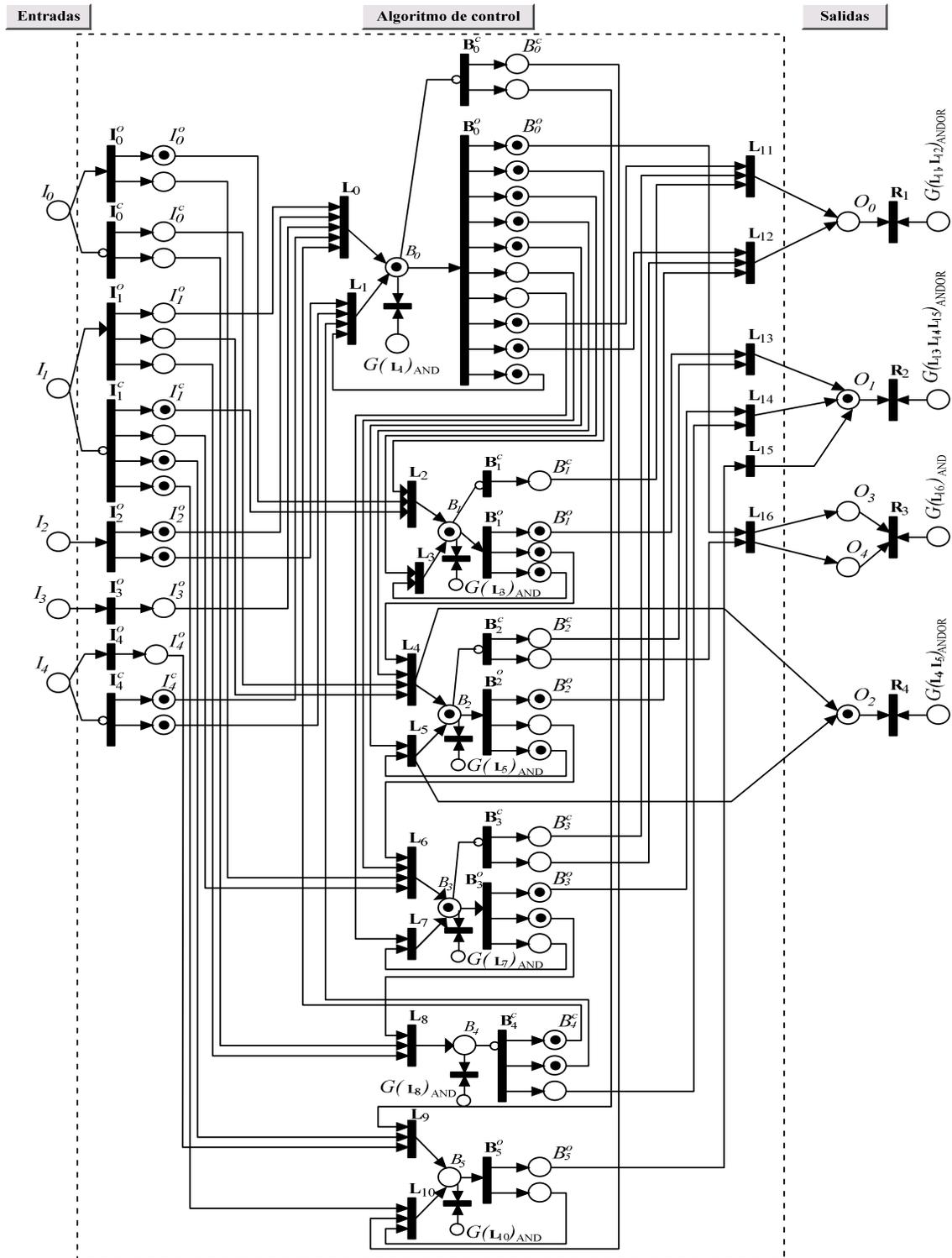


Figura 5.20: LDPN del sistema de control de auto-lavado de vehículos, cuarto ciclo de trabajo del PLC.

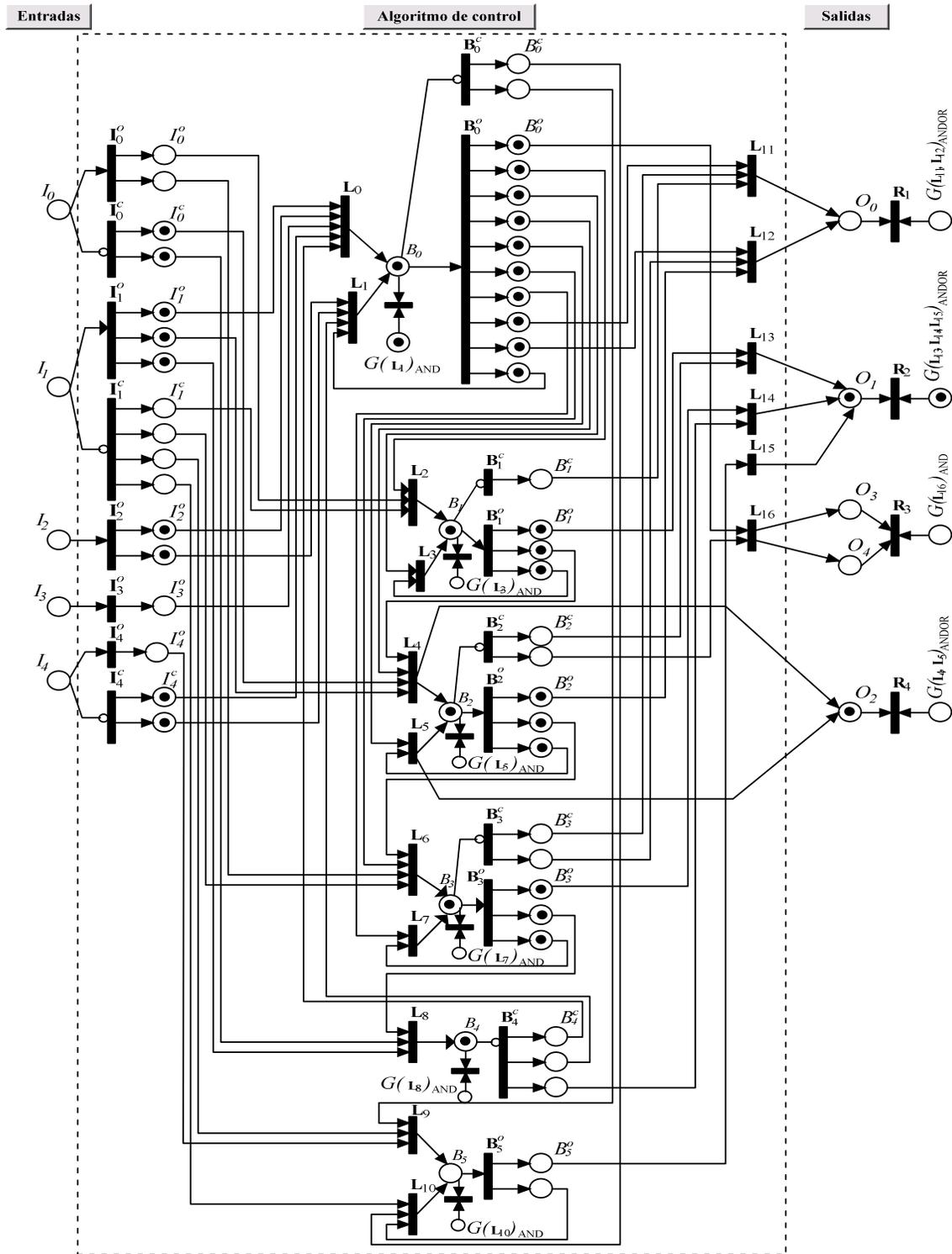


Figura 5.21: LDPN del sistema de control de auto-lavado de vehículos, actualización del marcado, segunda vez de $S2=1$.

la máquina no se encuentra en uno de los extremos, es decir, $S1 = 0$ y $S2 = 0$, en la LDPN marca en los lugares $I_0^c = 1$ y $I_1^c = 1$ respectivamente. Si es presionado el botón de paro $P = 1$, equivalente a colocar marca en $I_4^o = 1$ al disparar la transición \mathbf{I}_4^o y drenar la marca de los lugares $I_4^c = 0$, lo que genera se habilite la transición \mathbf{L}_9 que al dispararse coloca marca en el lugar $B_5 = 1$ como muestra la Figura 5.23. En el marcado se considera presencia de carro, marca en los lugares $I_2^o = 1$ ya que el paro de la máquina puede ocurrir en cualquier momento de la operación del sistema.

Si se dispara la transición \mathbf{B}_5^o se coloca marca en los lugares B_5^o lo que habilita la transición \mathbf{L}_{15} y coloca marca en el lugar de salida $O_1 = 1$ al ser disparada, equivalente a poner en marcha el motor con giro hacia la izquierda, bobina $MP_I = 1$. También se habilita la transición \mathbf{L}_{10} que sirve para mantener con marca en el lugar $B_5 = 1$ como muestra la Figura 5.24. Cuando la máquina llega al extremo izquierdo del riel, se activa el interruptor de límite $S2 = 1$, ocasionando el drenado de los lugares $I_1^c = 0$ y restableciendo las condiciones de la red en el marcado inicial de operación del sistema de auto-lavado de vehículos.

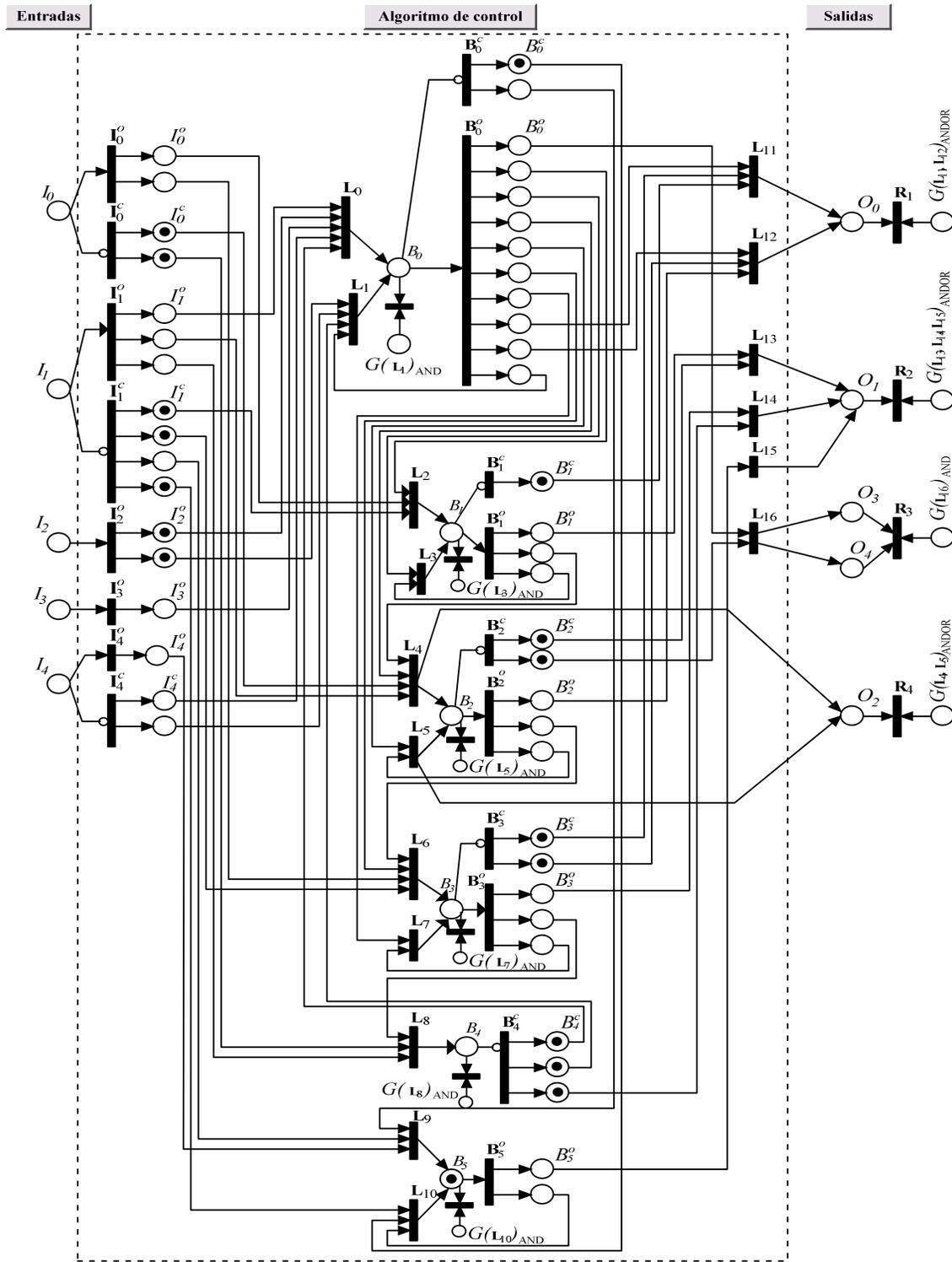


Figura 5.23: LDPN del sistema de control de auto-lavado de vehículos, accionamiento de $P = 1$ en LD, $B_5 = 1$ en LDPN.

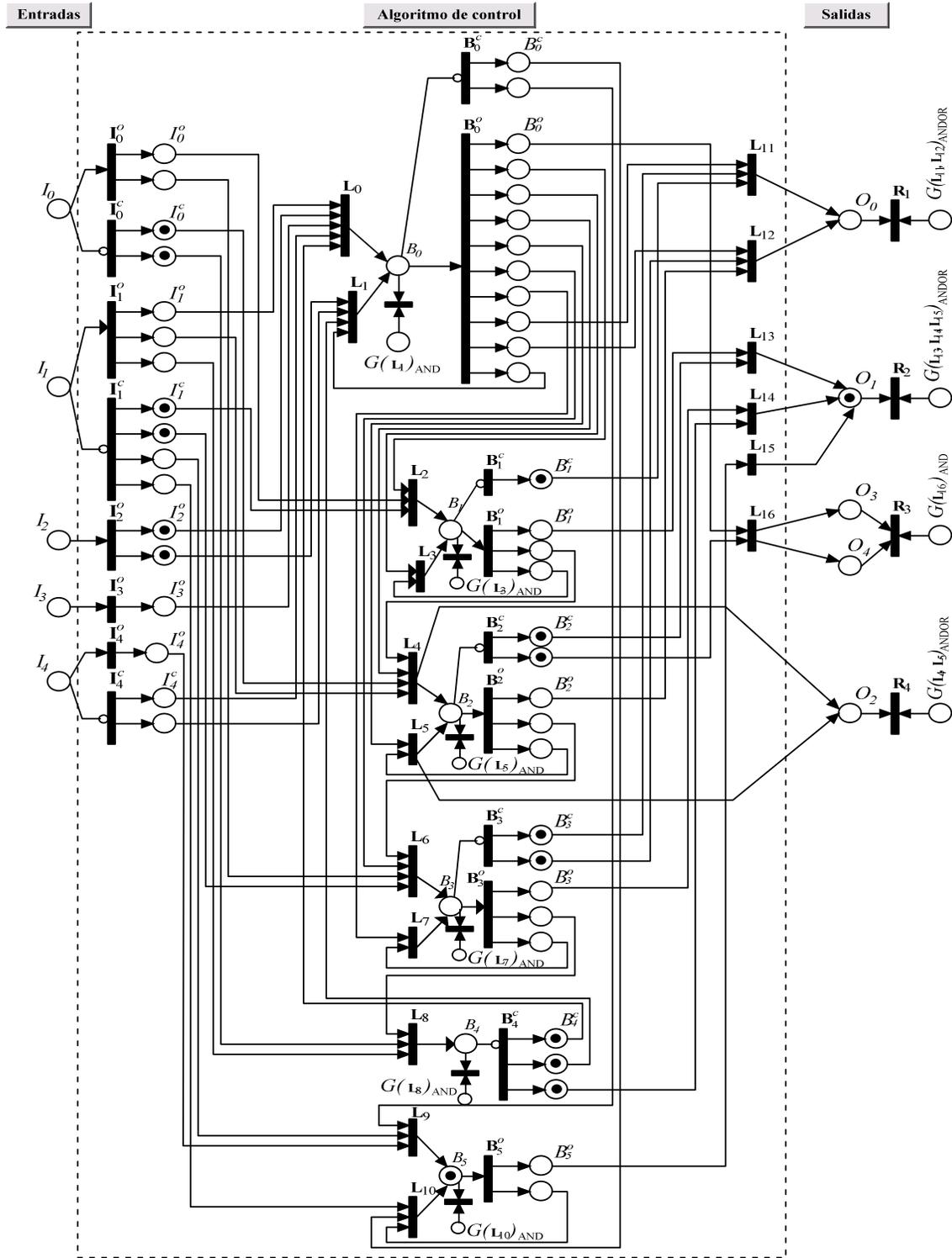


Figura 5.24: LDPN del sistema de control de auto-lavado de vehículos, $O_1 = 1$ en condiciones de paro.

5.2.4. Validación de la red LDPN del sistema de control de auto-lavado de vehículos

El algoritmo de control del sistema de auto-lavado de vehículos tiene cinco señales de entrada físicas, S1, S2, S3, M y P, por lo que con base en la Ecuación 4.5 se tiene 160 combinaciones posibles como muestra la Ecuación 5.3, y por la Ecuación 4.6 sería de 992 posibles combinaciones de falla de circuito-abierto o corto-circuito como muestra la Ecuación 5.4.

$$Ft = 2^5 * 5 = 160 \quad (5.3)$$

$$Ft = 4^5 - 2^5 = 992 \quad (5.4)$$

Por razones de espacio se omite mostrar el conjunto \mathbf{M}_{Ft} de las combinaciones posibles de falla.

Con base en los resultados de la evaluación de las condiciones de falla en tiempo real con la HMI, la matriz de la Ecuación 5.8 muestra las condiciones de riesgo del sistema de control de auto-lavado de vehículos.

$$\begin{bmatrix} I_0 & I_1 & I_2 & I_3 & I_4 & \dots & O_0 & O_1 & O_2 & O_3 & O_4 \\ 0/1 & 0/1 & 0/1 & 0/1 & sc & \dots & 0 & 1 & 0 & 0 & 0 \\ sc & 0/1 & 0/1 & 0/1 & sc & \dots & 0 & 1 & 0 & 0 & 0 \\ 0/1 & 0/1 & sc & 0/1 & sc & \dots & 0 & 1 & 0 & 0 & 0 \\ 0/1 & 0/1 & 0/1 & sc & sc & \dots & 0 & 1 & 0 & 0 & 0 \\ sc & sc & 0/1 & 0/1 & sc & \dots & 0 & 1 & 0 & 0 & 0 \\ sc & 0/1 & 0/1 & sc & sc & \dots & 0 & 1 & 0 & 0 & 0 \\ sc & 0/1 & sc & sc & sc & \dots & 0 & 1 & 0 & 0 & 0 \\ sc & sc & sc & sc & 0/1 & \dots & 1 & 0 & 0 & 1 & 1 \end{bmatrix} \quad (5.5)$$

Las observaciones al algoritmo de control de auto-lavado de vehículos son:

- a) La falla de circuito-abierto no genera marcado en los lugares O_o de salida del sistema.

- b) Independientemente de si en otros lugares tienen marca por falla u operación normal del sistema, si $I_4 = 1$ (accionamiento del botón de paro de la máquina P) e $I_1 = 0$ (interruptor de límite del extremo izquierdo del riel $S2 = 0$), entonces se coloca marca en el lugar de salida $O_1 = 1$, equivalente al accionamiento del motor con giro hacia la izquierda, es decir $MP_I = 1$ en el LD. Si $S2 = sc$, entonces la máquina no se detendría al llegar al extremo izquierdo, lo cual es una condición de riesgo para las personas y para el propio sistema.
- c) Si la señal de paro es activada $P = 1$, entonces la máquina se debe detener para revisión física de posible situación de riesgo, y no arrancar el motor con giro hacia la izquierda.
- d) Si los lugares $I_0 = I_1 = I_2 = I_3 = sc$ están en falla de corto-circuito, entonces se coloca una marca en los lugares de salida $O_0 = O_3 = O_4 = 1$, equivalente a energizar las bobinas $MP_I = MC = XV = 1$ que accionarían al motor que mueve la máquina hacia el lado izquierdo, al motor de los cepillos y al energizado de la electro-válvula, lo cual también representa una condición de alto riesgo para las personas y el sistema.

5.3. Sistema de envasado

El sistema de envasado consta de dos básculas que descargan en un mismo conducto para el llenado de costales de 50 kg. La Figura 5.25 muestra el diagrama esquemático. Es importante resaltar, que las electro-válvulas de peso grueso y fino de cada báscula son normalmente energizadas para que en caso de falla de energía eléctrica se cierren las compuertas y se evite el derrame del producto que se este envasando. Las especificaciones formales son:

- 1.- La alimentación de energía eléctrica y la banda de extracción en operación son señales necesarias para funcionamiento del sistema de envasado I16 e I15 respectivamente.
- 2.- El sistema de envasado puede funcionar con ambas básculas si los selectores I8 y I12 están en funcionamiento.

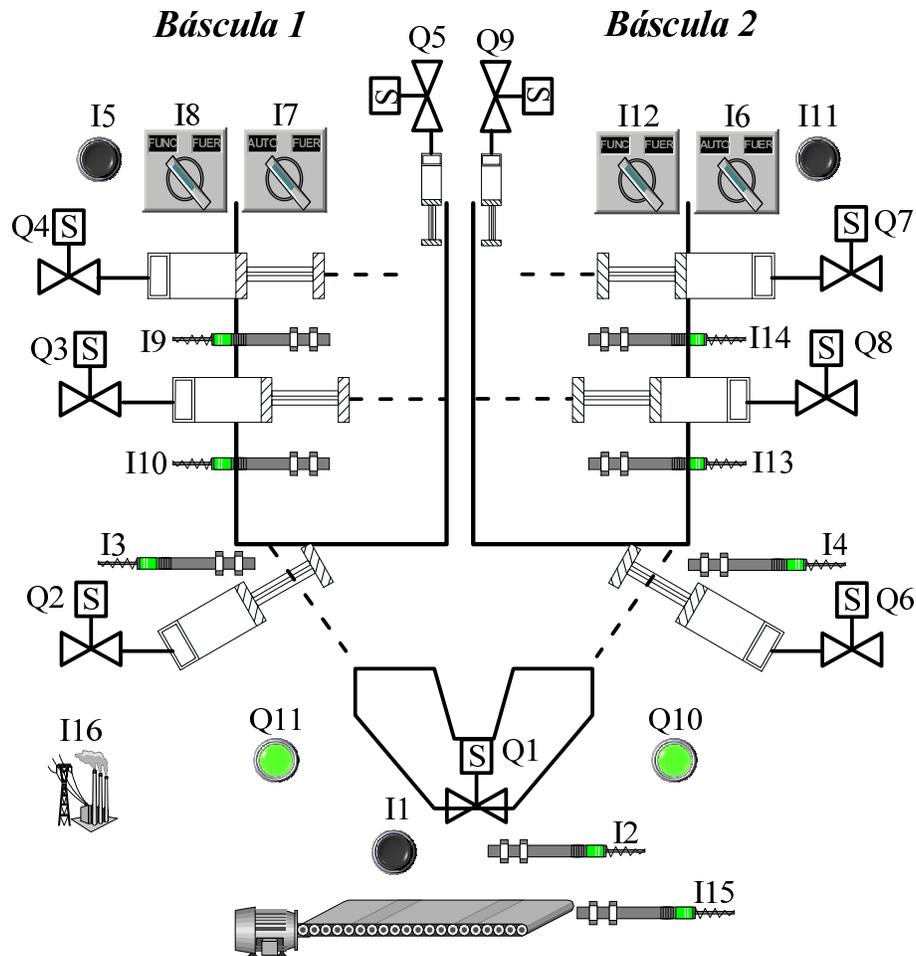


Figura 5.25: Sistema de envasado.

- 3.- El sistema de envasado puede funcionar con una sola báscula, I8 o I12 está en funcionamiento.
- 4.- Cada báscula tiene la opción de funcionar en automático o manual, señal I7 para báscula uno y señal I6 para báscula dos.
- 5.- Con peso de 45 kg el sensor de peso grueso (señal I9 para báscula uno y señal I14 para báscula dos) debe mandar a des-energizar la electro-válvula del pistón correspondiente (señal Q4 para báscula uno y Q7 para báscula dos) para cerrar una compuerta que limite el llenado de la báscula. Cada sistema es independiente.

- 6.- Con peso de 50 kg el sensor de peso fino (señal I10 para báscula uno y señal I13 para báscula dos) debe mandar a des-energizar la electro-válvula del pistón correspondiente (señal Q3 para báscula uno y Q8 para báscula dos) para cerrar una compuerta que evita la entrada de más producto.
- 7.- Para poder hacer una descarga de una báscula, debe haber costal en el zuncho, señal I2.
- 8.- Con ambas básculas en automático, su descarga se va alternando en función de cual alcance el peso primero, y se debe energizar la electro-válvula del pistón de la tapa inferior para abrirla (señal Q2 para la báscula uno y Q6 para la báscula dos), la condición de la tapa es a través de la señal I3 para la báscula uno e I4 para la báscula dos.
- 9.- Con funcionamiento de una sola báscula, su descarga debe ser cada vez que alcance el peso esta.
- 10.- Cuando ocurre una descarga, se debe energizar la electro-válvula del pistón de alto rendimiento para estabilizar la báscula, señal Q5 para la báscula uno y Q9 para la báscula dos.
- 11.- La activación de la electro-válvula del zuncho, señal Q1, también puede realizarse con un botón instantáneo.
- 12.- Cada báscula tiene un botón instantáneo para descarga manual, señal I5 para báscula uno e I11 para báscula dos.
- 13.- Cuando una báscula alcance su peso, se enciende una alarma luminosa, señal Q11 para la báscula uno y Q10 para báscula dos.

La Tabla 5.3 muestra las variables de entradas y salidas físicas y la Tabla 5.4 muestra las variables internas o de memoria utilizadas en el algoritmo de control en LD.

Tabla 5.3: Variables de señales físicas de entrada y salida del sistema de envasado.

Variable	Símbolo	Descripción
I.1	PB_ZUNCHO	Botón instantáneo operación manual del zuncho.
I.2	S_ZUNCHO	Sensor de presencia de costal en el zuncho.
I.3	S_CHAP_B1	Sensor de tapa inferior de báscula uno.
I.4	S_CHAP_B2	Sensor de tapa inferior de báscula dos.
I.5	PB_DESC_B1	Botón instantáneo de descarga de bascula uno.
I.6	SEL_AUT_B2	Selector de funcionamiento automático de báscula dos.
I.7	SEL_AUT_B1	Selector de funcionamiento automático de báscula uno.
I.8	SEL_FUNC_B1	Selector de funcionamiento báscula uno.
I.9	S_GRUE_B1	Sensor de pesado grueso de báscula uno.
I.10	S_FINA_B1	Sensor de pesado fino de báscula uno.
I.11	PB_DESC_B2	Botón instantáneo de descarga de bascula dos.
I.12	SEL_FUNC_B2	Selector de funcionamiento báscula dos.
I.13	S_GRUE_B2	Sensor de pesado grueso de báscula dos.
I.14	S_FINA_B2	Sensor de pesado fino de báscula dos.
I.15	BANEX_FUNC	Transportado de sacos llenos en funcionamiento.
I.16	ALIM.127VCA	Sistema de alimentación de energía eléctrica.
Q.1	V_ZUNCHO	Electro-válvula del zuncho.
Q.2	V_CHAP_B1	Electro-válvula de la tapa de inferior de la báscula uno.
Q.3	V_PGRU_B1	Electro-válvula del pistón de peso grueso de la báscula uno.
Q.4	V_PFIN_B1	Electro-válvula del pisto de peso fino de la báscula uno.
Q.5	V_PALR_B1	Electro-válvula del pistón de alto rendimiento de la báscula uno.
Q.6	V_CHAP_B2	Electro-válvula de la tapa de inferior de la báscula dos.
Q.7	V_PFIN_B2	Electro-válvula del pistón de peso fino de la báscula dos.
Q.8	V_PGRU_B2	Electro-válvula del pisto de peso grueso de la báscula uno.
Q.9	V_PALR_B2	Electro-válvula del pistón de alto rendimiento de la báscula dos.
Q.10	XL_B2	Alarma luminosa de báscula dos llena.
Q.11	XL_B1	Alarma luminosa de báscula uno llena.

Tabla 5.4: Variables de señales de memoria del sistema de envasado.

Variable	Símbolo	Descripción
B_1	P_SISTEMA	Permisivos generales del sistema de envasado.
B_2	P_ZUNCHO	Permisivos del zuncho.
B_3	F_AUT_ZUNCH	Sistema de envasado en automático.
B_4	P_B1	Permisivos de báscula uno.
B_5	P_B2	Permisivos de báscula dos.
B_6	P_APER_B1	Permisivos de descarga de báscula uno.
B_7	PESO_B1	Peso completo de báscula uno.
B_8	P_APER_B2	Permisivos de descarga de báscula dos.
B_9	ENC_PF_B2	Retención del pistón de peso fino de báscula dos.
B_10	PESO_B2	Peso completo de báscula uno.
B_11	ENC_PG_B2	Retención del pistón de peso grueso de báscula dos.
B_12	ENC_PF_B1	Retención del pistón de peso fino de báscula uno.
B_13	ENC_PG_B1	Retención del pistón de peso grueso de báscula uno.
B_14	TON_0	Retardo de tiempo para abrir el zuncho.
B_15	TON_1	Retardo de tiempo para cerrar la tapa inferior de la báscula uno.
B_16	TON_2	Retardo de tiempo para des-energizar el pistón de alto rendimiento de la báscula uno.
B_17	TON_3	Retardo de tiempo para cerrar la tapa inferior de la báscula dos.
B_18	TON_4	Retardo de tiempo para des-energizar el pistón de alto rendimiento de la báscula dos.

5.3.1. Algoritmo de control de sistema de envasado

El algoritmo de control del sistema de envasado tiene 29 líneas de control, además tiene bloques de tiempo que son considerados para la presente transformación, modelado y validación como acciones de retardo que no afectan el enfoque propuesto. Las Figuras 5.26, 5.27, 5.28 y 5.29.

El algoritmo de control en LD tiene varias estructuras lógicas: AND, AND-OR, auto-lazo, bloqueo, comportamiento de ejecución secuencia, además, contempla el funcionamiento de bobinas energizadas de inicio para operación del sistema de envasado.

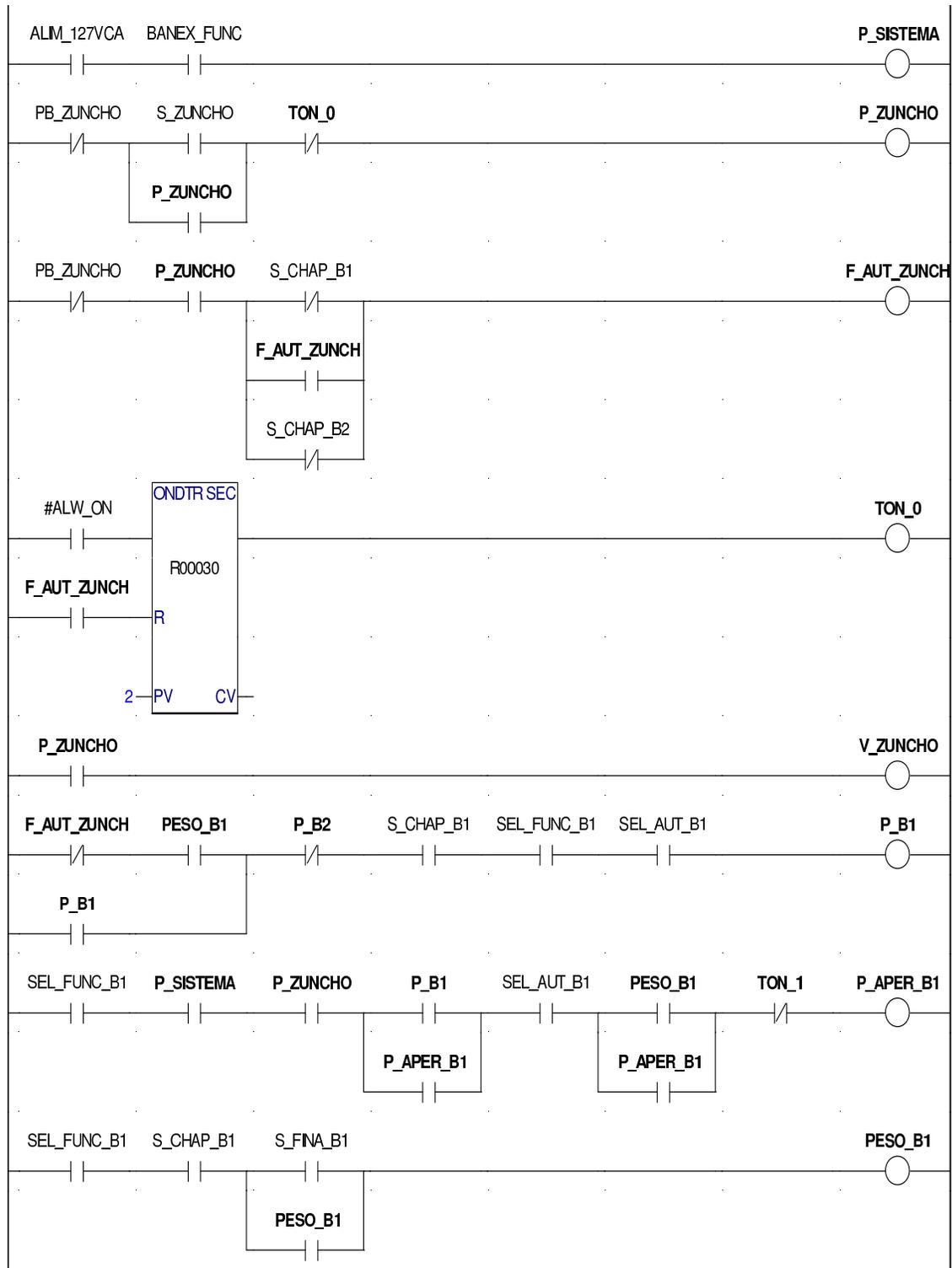


Figura 5.26: Algoritmo de control en LD del sistema de envasado, parte uno.

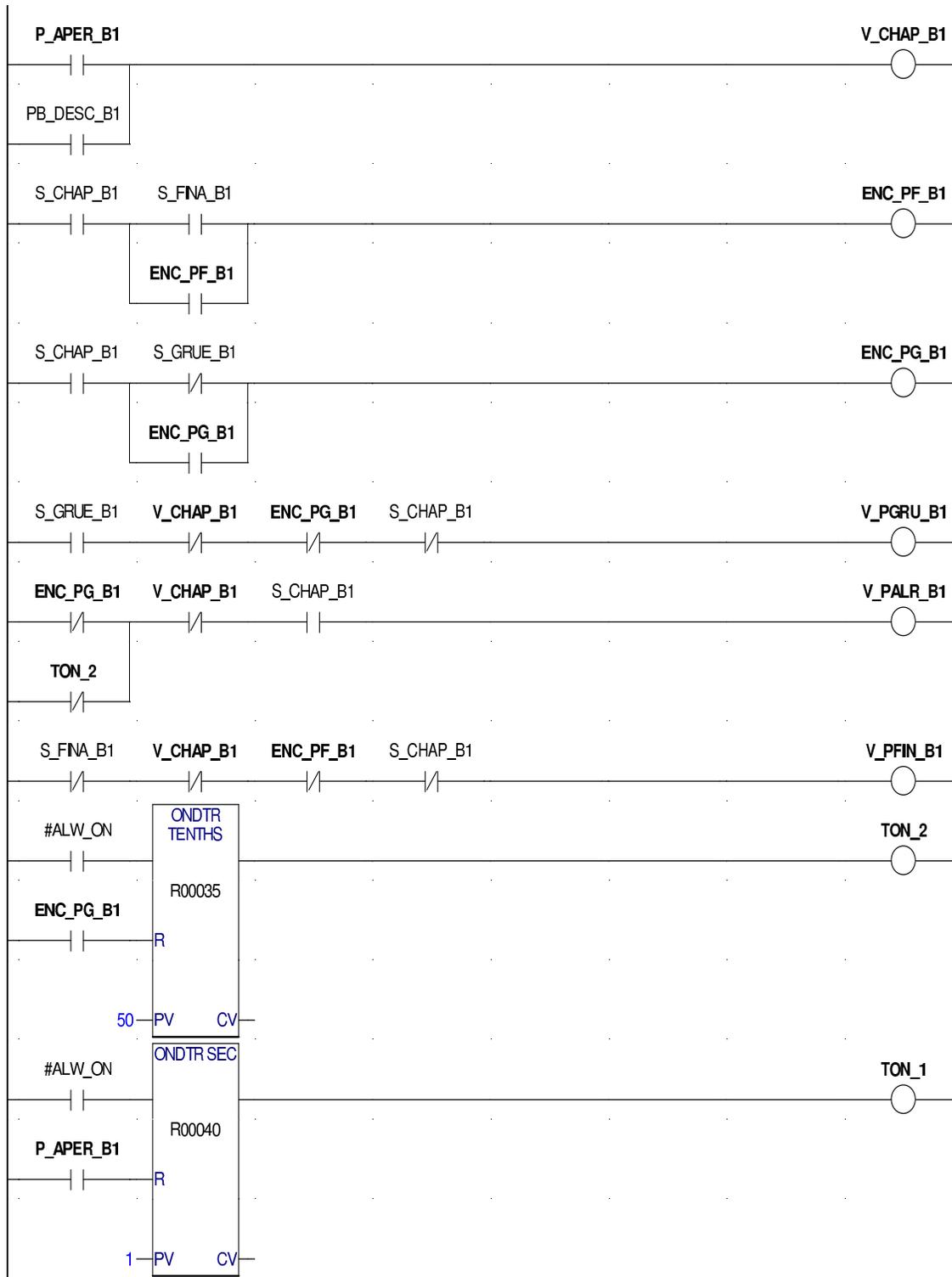


Figura 5.27: Algoritmo de control en LD del sistema de envasado, parte dos.

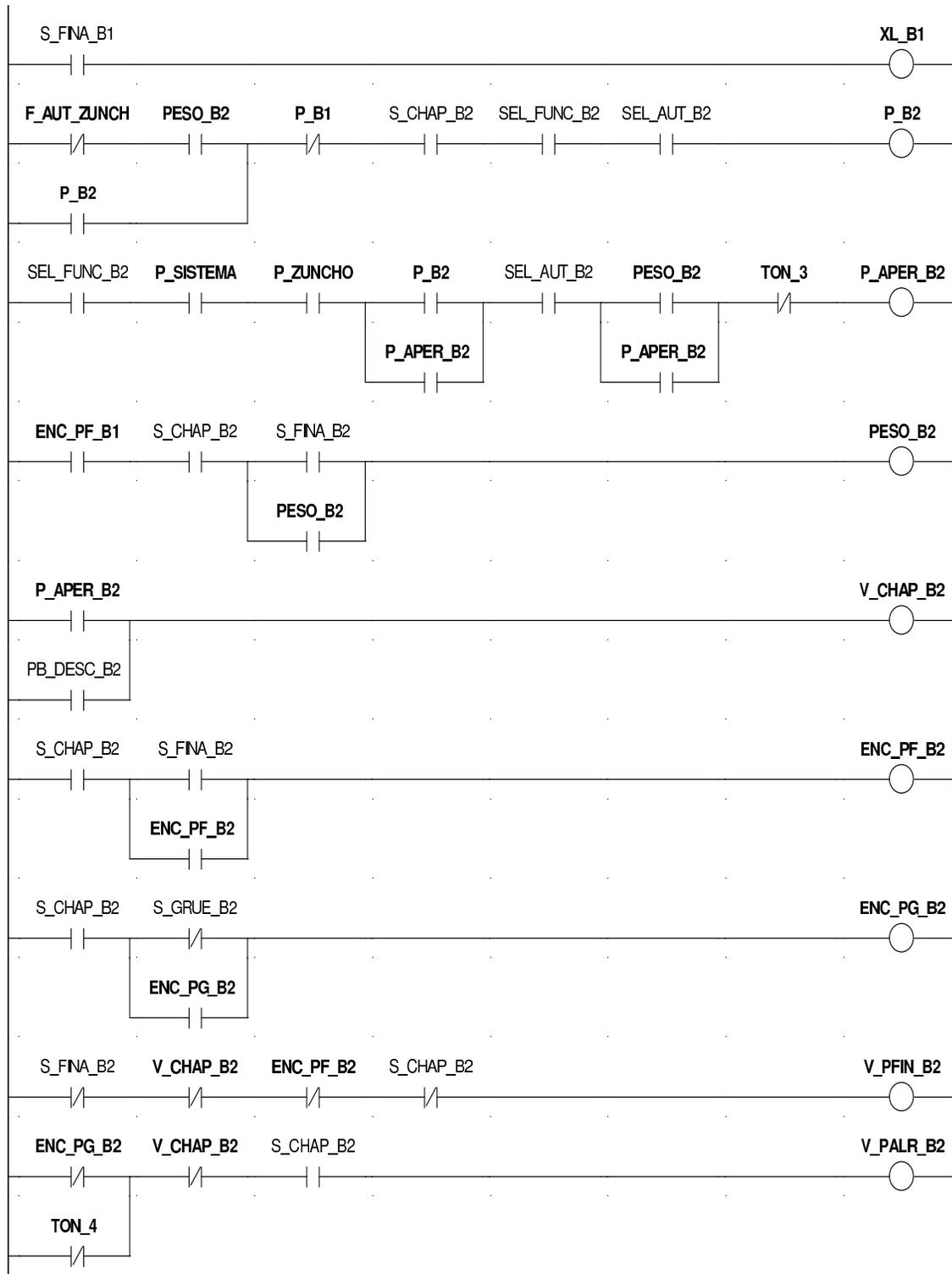


Figura 5.28: Algoritmo de control en LD del sistema de envasado, parte tres.

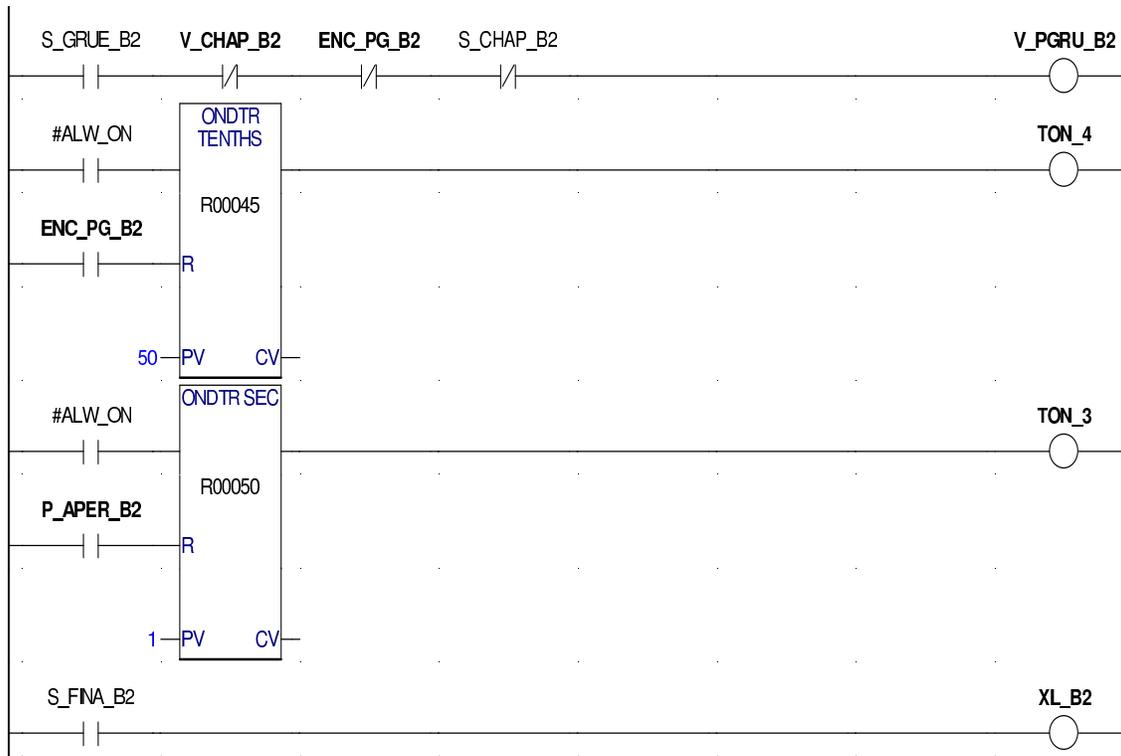


Figura 5.29: Algoritmo de control en LD del sistema de envasado, parte cuatro.

5.3.2. Red LDPN del sistema de envasado

La Figura 5.30 muestra la red LDPN correspondiente a la báscula uno del sistemas de envasado y la Figura 5.31 a la báscula dos. Por razones de espacio en las figuras se omite colocar las transiciones **R**'s en la parte interior de la red, que corresponde al algoritmo de control, pero si se colocan los lugares de restablecimiento respectivo con su función para habilitar el marcado correspondiente.

El lugar B_1 representa los permisos del sistemas, por lo que es necesario para el funcionamiento de ambas básculas, equivalente en algoritmo de control a la bobina P_SISTEMA. De igual forma, los permisos del zuncho representados en la LDPN por el lugar B_2 es necesario para la descarga alternadas de las básculas o del funcionamiento de solo una, equivalente a la bobina P_ZUNCHO.

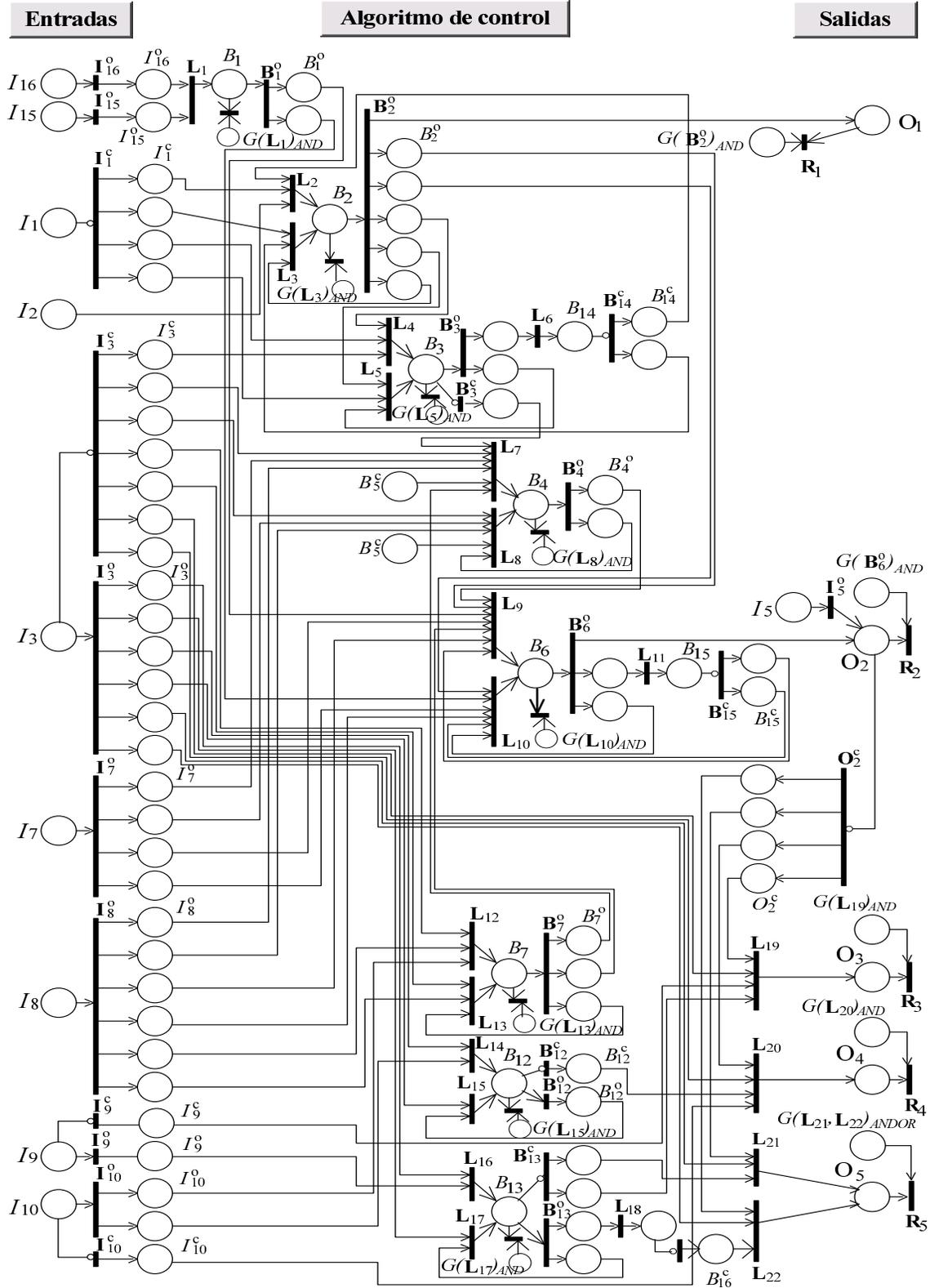


Figura 5.30: LDPN de báscula uno

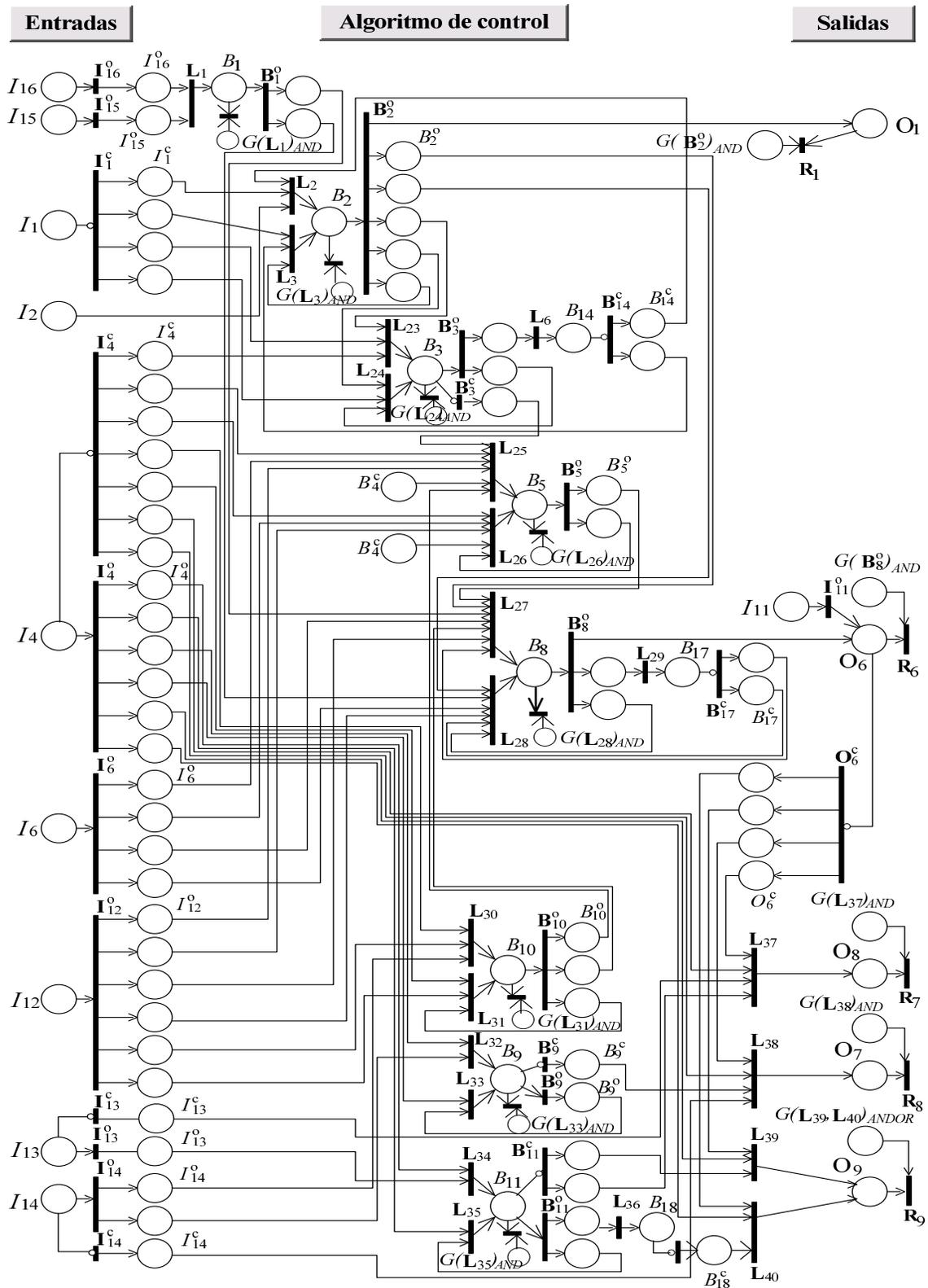


Figura 5.31: LDPN de báscula dos

El lugar B_3 representa el funcionamiento automático del zuncho, y puede recibir marca de la transición \mathbf{L}_2 y \mathbf{L}_{23} en función de que báscula se descargue, equivalente a la bobina F_AUT_ZUNCH que tiene una configuración AND-OR con señales S_CHAP_B1 y S_CHAP_B1 en paralelo, línea de control número tres.

La función de bloqueo se ve reflejada en la red LDPN de la báscula uno en las transiciones \mathbf{L}_7 y \mathbf{L}_8 que tienen como lugar de entrada B_5^c que representan un contacto NC de la bobina P_APER_B2 en el algoritmo de control. De igual forma, en la LDPN de la báscula dos en las transiciones \mathbf{L}_{25} y \mathbf{L}_{26} tienen como lugar de entrada B_4^c que representa un contacto NC de la bobina P_APER_B1.

Con las consideraciones arriba descritas, la matriz de incidencia del sistema de envasado puede ser determinada de la red LDPN respectiva. Se omite su presentación por razones de espacio.

5.3.3. Simulación de la red LDPN del sistema de envasado

Para la simulación de la red LDPN del sistema de envasado se considera solo el funcionamiento de una báscula, ya que el comportamiento dinámico de ambas básculas es similar de forma independiente o trabajando juntas, ya que con el bloqueo de una con la otras, solo puede ser descargada una a la vez. Se considera la red LDPN de la báscula uno para la simulación.

La Figura 5.32 muestra el marcado de los lugares que corresponden a los contactos NC del algoritmo de control en LD. Note que con este marcado las transiciones \mathbf{L}_{19} y \mathbf{L}_{20} están habilitadas con este marcado.

Para funcionamiento del sistema de envasado es necesario la activación permanente de las señales ALIM_127VCA=1 y BANEX_FUNC15=1 que energizan la bobina de permisivos del sistema P_SISTEMA= 1 en el LD, equivalente a tener marca en los lugares I_{15} e I_{16} respectivamente, habilitando las transiciones \mathbf{I}_{15}^o e \mathbf{I}_{16}^o , al ser disparadas colocan marca en los lugares I_{15}^o e I_{16}^o y habilitan la transición \mathbf{L}_1 que al dispararse coloca marca en el lugar B_1 . De igual forma, son necesarias la selección de las señales SEL_AUT_B1=1 y SEL_FUNC_B1=1 correspondiente a colocar marca en los lugares $I_7 = 1$ e $I_8 = 1$ respectivamente, habilitando las transiciones \mathbf{I}_7^o e \mathbf{I}_8^o que al ser disparadas colocan una marca en los lugares $I_7^o = 1$ e $I_8^o = 1$. Además al disparar las transiciones \mathbf{L}_{19} y \mathbf{L}_{20} se coloca marca en los lugares $O_3 = 1$ y $O_4 = 1$ en la red LDPN. Lo anterior es

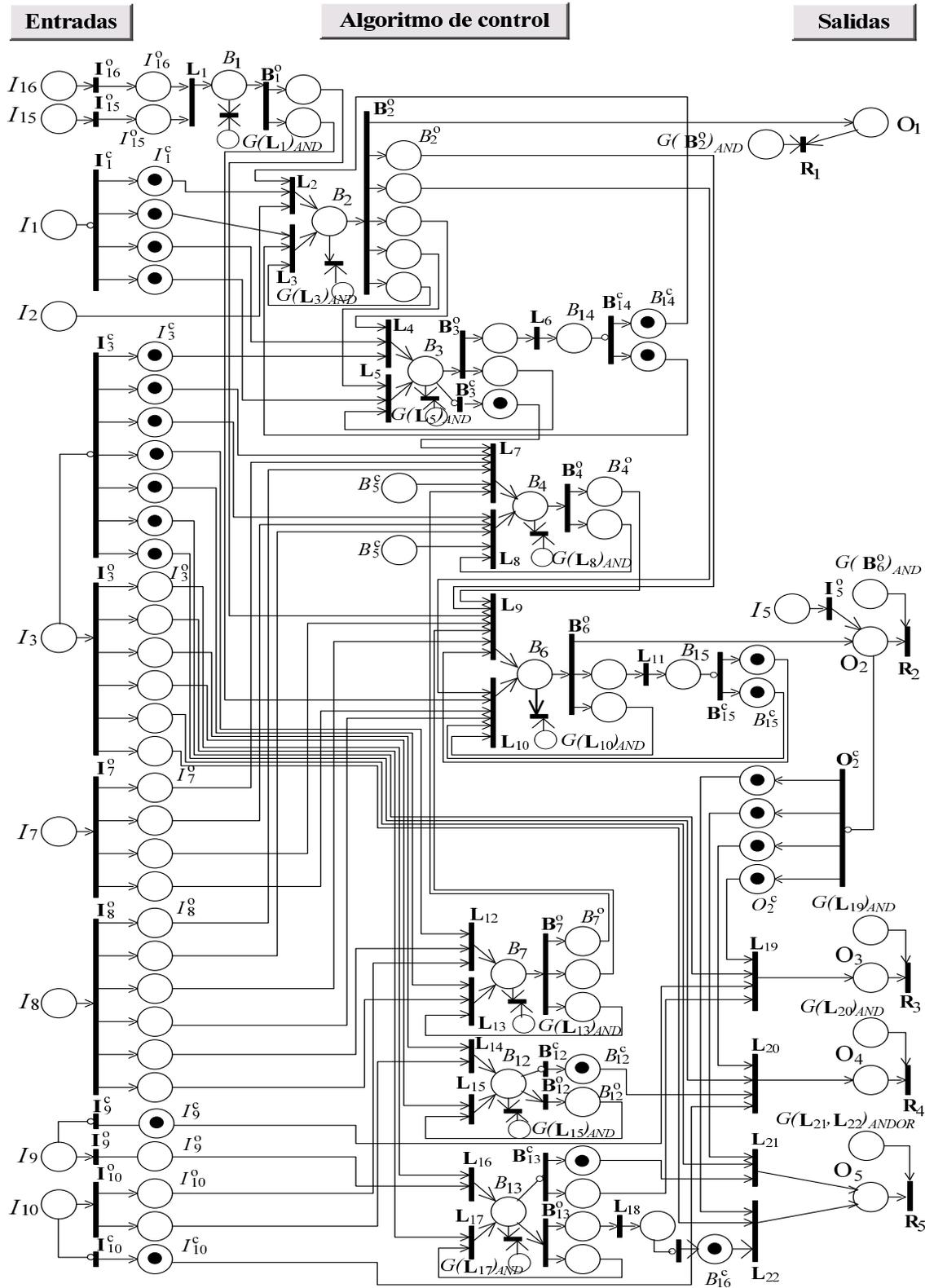


Figura 5.32: LDPN de báscula uno, marcado en lugares I_i^c , B_b^c y O_o^c .

equivalente a energizar las bobinas $V_PGRU_B1=1$ y $V_PFIN_B1=1$ respectivamente, lo que implica que en el caso de falla de energía eléctrica en el sistema de envasado, entonces las compuertas de llenado de la báscula quedan cerradas como condición inicial. El marcado se muestra en la Figura 5.33, y corresponde a tener en condiciones de funcionamiento el sistema de envasado solo con la báscula uno en operación y a un ciclo de operación del sistema basado en PLC.

El sistema de envasado se encuentra en proceso de llenado de la báscula uno, al llegar a los 45 kg el sensor $S_GRUE_B1=1$ en el algoritmo de control en LD, equivalente a colocar una marca en el lugar $I_9 = 1$, habilitando la transición \mathbf{I}_9^o y colocando marca en el lugar $I_9^c = 0$. Actualizando el marcado de la LDPN, el lugar $I_9^c = 0$ habilitando el marcado del lugar de restablecimiento $G(\mathbf{L}_{19})_{AND} = 1$ para drenar la marca en el lugar $Q_3 = 0$, equivalente a des-energizar la bobina $V_PGRU_B1=0$ en el LD. Hasta aquí serían dos ciclos de trabajo del sistema basado en PLC, por lo que se actualiza el marcado como muestra la Figura 5.34. La acumulación de marcas se evita por las Ecuaciones 3.3a y 3.4a.

La báscula uno del sistema de envasado continúa llenándose hasta llegar a los 50 kg, donde el sensor $S_FINA_B1=1$ en el algoritmo de control en LD, equivalente a colocar marca en el lugar $I_{10} = 1$, deshabilitando la transición \mathbf{I}_{10}^c y drenando la marca del lugar $I_{10}^c = 0$ por la Ecuación 3.1b, lo que habilita el marcado del lugar de restablecimiento $G(\mathbf{L}_{20})_{AND} = 1$ para drenar la marca del lugar $O_4 = 0$, equivalente a des-energizar la bobina $V_PFIN_B1=0$ en el LD. También se habilita la transición \mathbf{I}_{10}^o que al dispararse coloca marca en el lugar $I_{10}^o = 1$, con este marcado se habilita la transición \mathbf{L}_{12} que al ser disparada coloca marca en el lugar $B_7 = 1$, equivalente a energizar la bobina $PESO_B1=1$. La Figura 5.35 muestra el marcado correspondiente y es la conclusión de un tercer ciclo de trabajo del sistema basado en PLC.

Disparando la transición \mathbf{B}_7^o se coloca marca en los lugares $B_7^o = 1$ habilitando las transiciones \mathbf{L}_{13} de su auto-lazo para retener la marca y \mathbf{L}_7 (báscula dos fuera de funcionamiento, marca en los lugares B_5^o) que al ser disparada coloca marca en el lugar $B_4 = 1$, equivalente a energizar la bobina $P_B1=1$ en el algoritmo de control en LD. La Figura 5.36 muestra el marcado correspondiente y es la terminación del cuarto ciclo de trabajo del sistema basado en PLC.

Al disparar la transición \mathbf{B}_4^o se coloca marca en los lugares $B_4^o = 1$ habilitando la transición \mathbf{B}_4^o que al ser disparada coloca marca en los lugares $B_4^o = 1$. Se considera su auto-lazo a través de la transición \mathbf{L}_8 para mantener con una marca el lugar $B_4 = 1$.

La báscula uno está llena y en condiciones de ser descargada. Al colocar el costal en el zuncho la señal S_ZUNCHO=1 en el LD, equivalente a colocar una marca en el lugar $I_2 = 1$ en la LDPN, habilitando la transición \mathbf{L}_2 que al dispararse coloca marca en el lugar $B_2 = 1$ y se habilita la transición \mathbf{B}_2^o que al ser disparada coloca marca en el lugar de salida $O_1 = 1$ correspondiente a energizar la bobina V_ZUNCHO=1 en el LD. El disparo de la transición \mathbf{B}_2^o , también coloca marca en los lugares $B_2^o = 1$ con lo que se habilita la transición \mathbf{L}_9 que al ser disparada coloca marca en el lugar $B_6 = 1$ equivalente a energizar la bobina P_APER_B1=1 en algoritmo de control en LD. Al disparar la transición \mathbf{B}_6^o se coloca marca en el lugar de salida $O_2 = 1$ similar a energizar la bobina de V_CHAP_B1=1 en el LD. También se coloca marca en los lugares $B_6^o = 1$. AL disparar la transición habilitada \mathbf{L}_{11} se coloca marca en el lugar $B_{15} = 1$ con lo que se inhabilita la transición \mathbf{B}_{15}^c , correspondiente a dar el tiempo de apertura de la tapa inferior de la báscula uno. De forma similar, la transición \mathbf{L}_4 es habilitada, al dispararla se coloca marca en el lugar $B_3 = 1$, lo que habilita la transición \mathbf{B}_3^o colocando marca en los lugares $B_3^o = 1$ y se habilita la transición \mathbf{L}_6 colocando marca en el lugar $B_{14} = 1$ inhabilitando la transición \mathbf{B}_{14}^c , correspondiente a dar el tiempo para des-energizar la bobina P_ZUNCHO y en consecuencia la bobina V_ZUNCHO en el LD. La Figura 5.37 muestra el marcado correspondiente y puede observarse que solo los lugares $I_9^o = 1$, $I_{10}^o = 1$ y dos lugares $I_3^c = 1$ tienen marca debido a la actualización.

Un nuevo ciclo del sistema basado en PLC implica actualizar el marcado, además de que el sistema está abriendo la tapa inferior de la báscula uno y el sensor de la tapa inferior cambia de estado a S_CHAP_B1=1 lo que habilita el marcado de las transiciones de restablecimiento $G(\mathbf{L}_5)_{AND}$, $G(\mathbf{L}_8)_{AND}$ y $G(\mathbf{L}_{13})_{AND}$, además de habilitar las transiciones \mathbf{L}_{14} y \mathbf{L}_{16} que al ser disparadas colocan marca en los lugares $B_{12} = 1$ y $B_{13} = 1$ respectivamente, en ambas estructuras se mantiene marca debido al auto-lazo a través de las transiciones \mathbf{L}_{15} para el lugar $B_{12} = 1$ y \mathbf{L}_{17} para el lugar $B_{13} = 1$. El marcado a este comportamiento se muestra en la Figura 5.38.

Después de transcurrido el tiempo de para la descarga de la báscula uno, la señal del sensor de la tapa inferior S_CHAP_B1=0, restableciendo el marcado de funcionamiento de la LDPN del sistema de envasado al habilitar los lugares de restablecimiento $G(\mathbf{L}_{10})_{AND}$, $G(\mathbf{L}_{15})_{AND}$, $G(\mathbf{L}_{17})_{AND}$, $G(\mathbf{B}_2^o)_{AND}$ y $G(\mathbf{B}_6^o)_{AND}$, y consumir las marcas de los lugares $B_6 = 0$, $B_{12} = 1$, B_{13} , $O_1 = 0$ y $O_2 = 0$. Además, los sensores de pesado grueso y fino restablecen el marcado en los lugares $I_9^o = 0$ y $I_{10}^o = 0$ y la actualización del marcado en los lugares $I_{16} = 1$, $I_{15} = 1$, $I_7 = 1$ e $I_8 = 1$ de los permisivos del sistema y de los selectores de funcionamiento de la báscula uno, como la Figura 5.33.

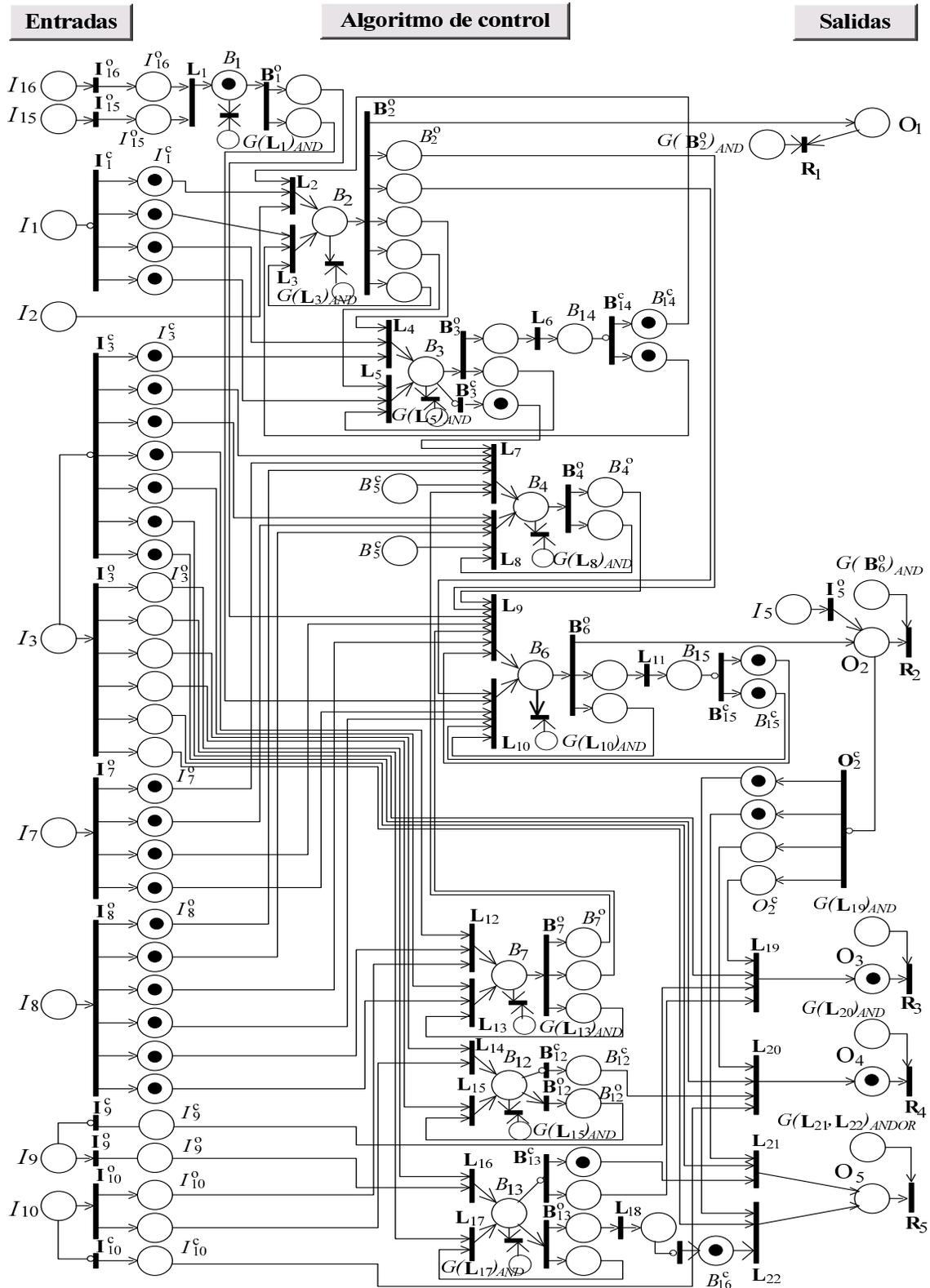


Figura 5.33: LDPN de báscula uno, marcado en condiciones de funcionamiento.

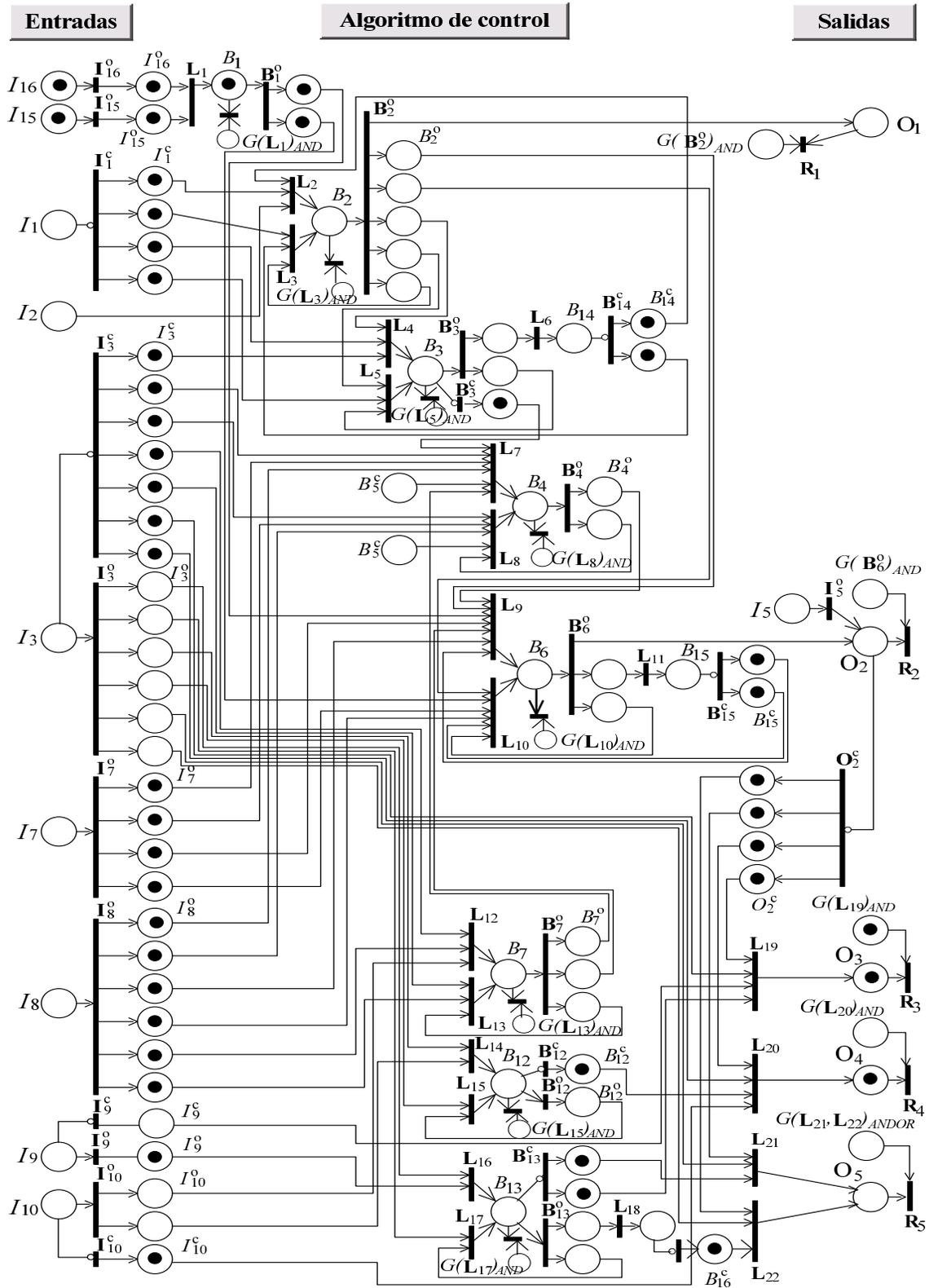


Figura 5.34: LDPN de báscula uno, segundo ciclo de trabajo del PLC.

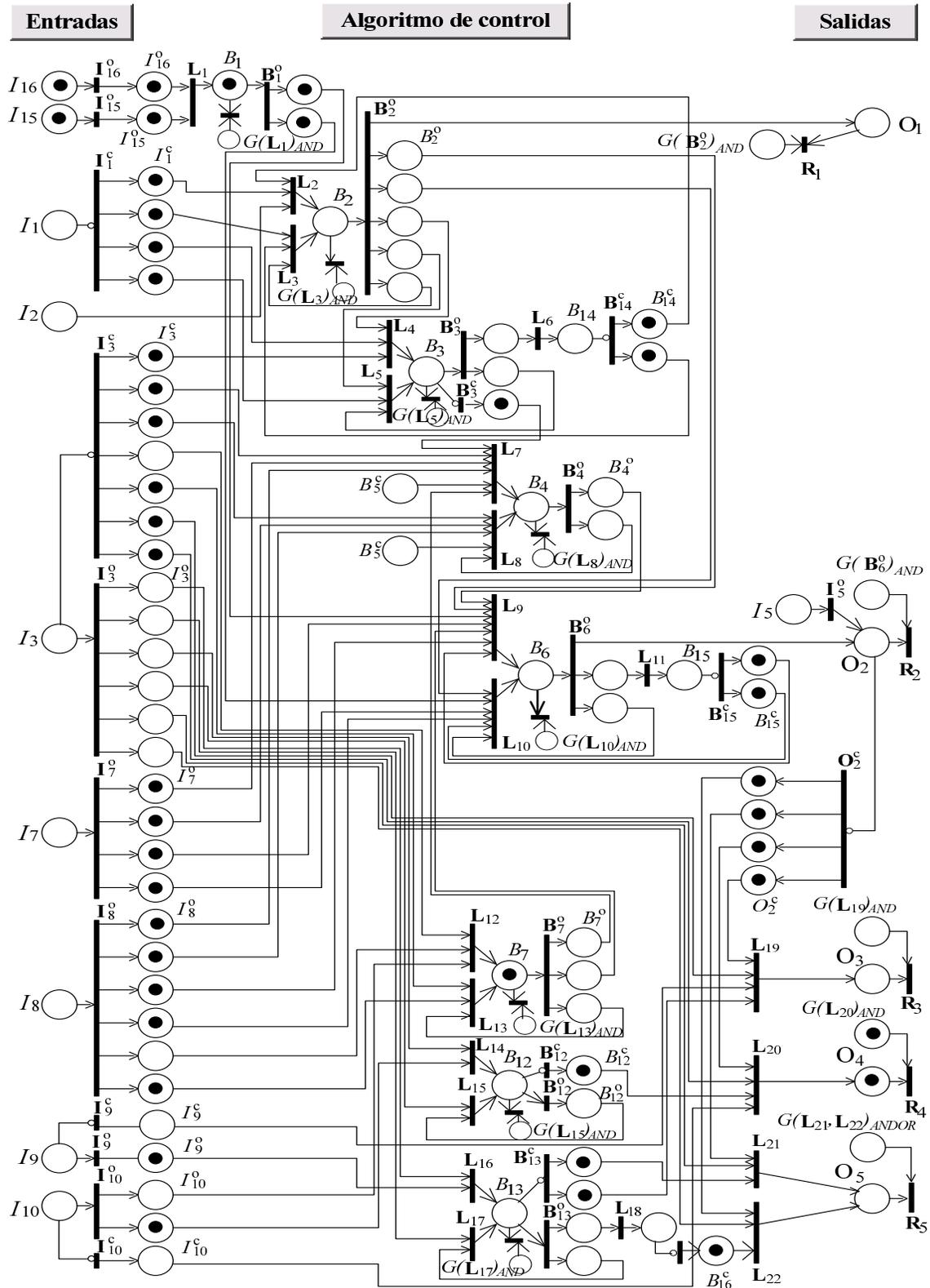


Figura 5.35: LDPN de báscula uno, tercer ciclo de trabajo del PLC.

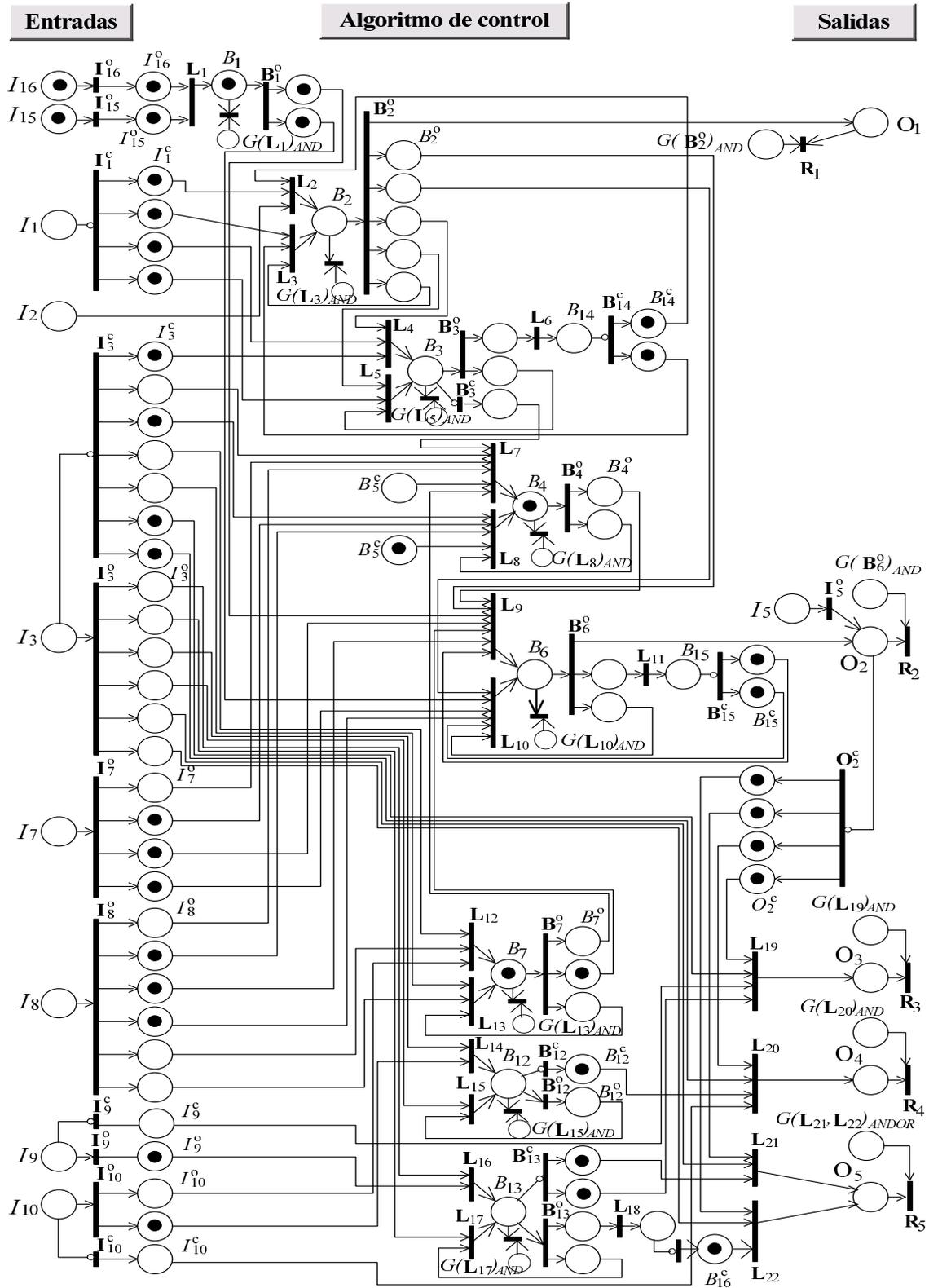


Figura 5.36: LDPN de báscula uno, cuarto ciclo de trabajo del PLC.

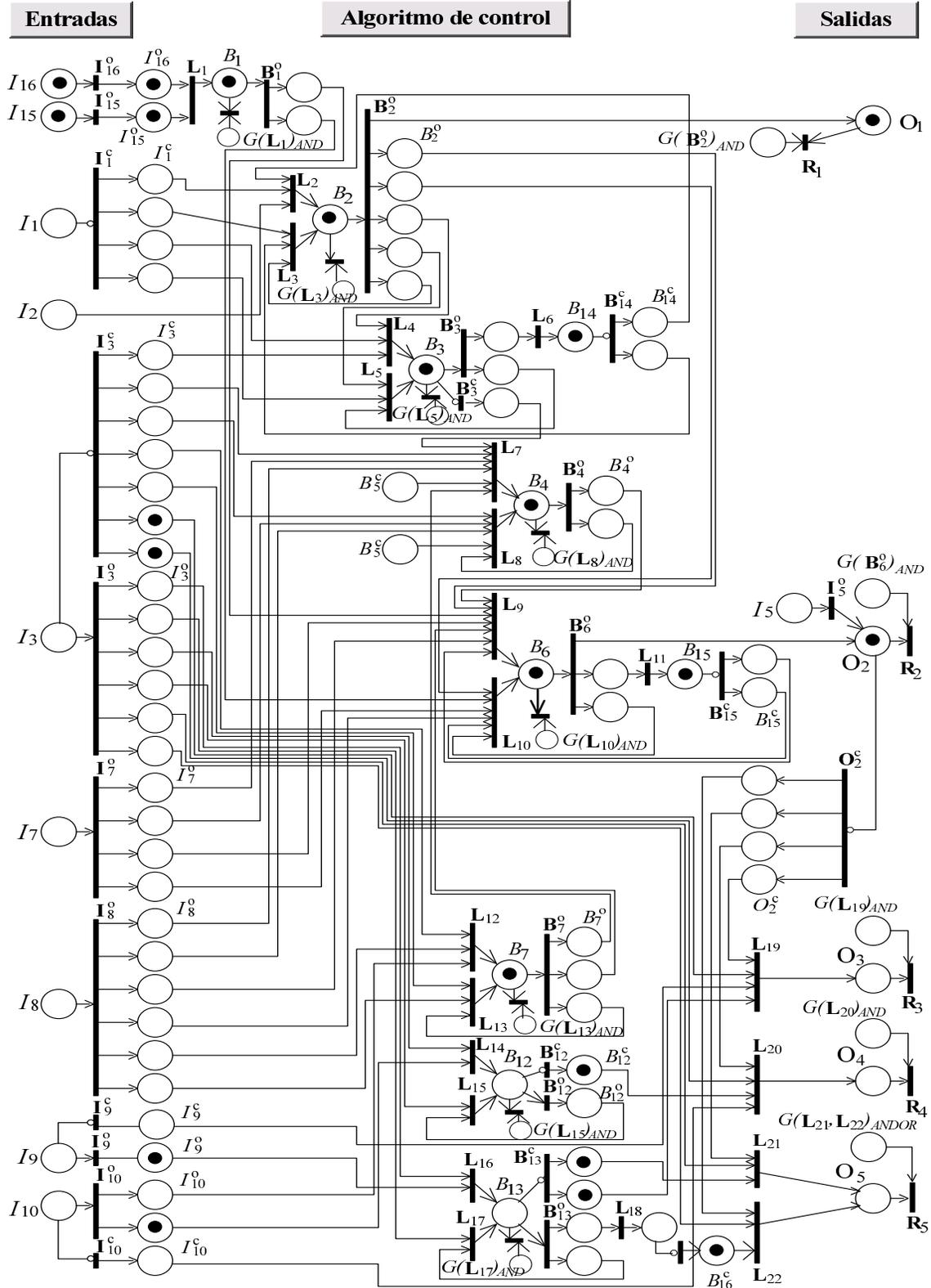


Figura 5.37: LDPN de báscula uno, descargando.

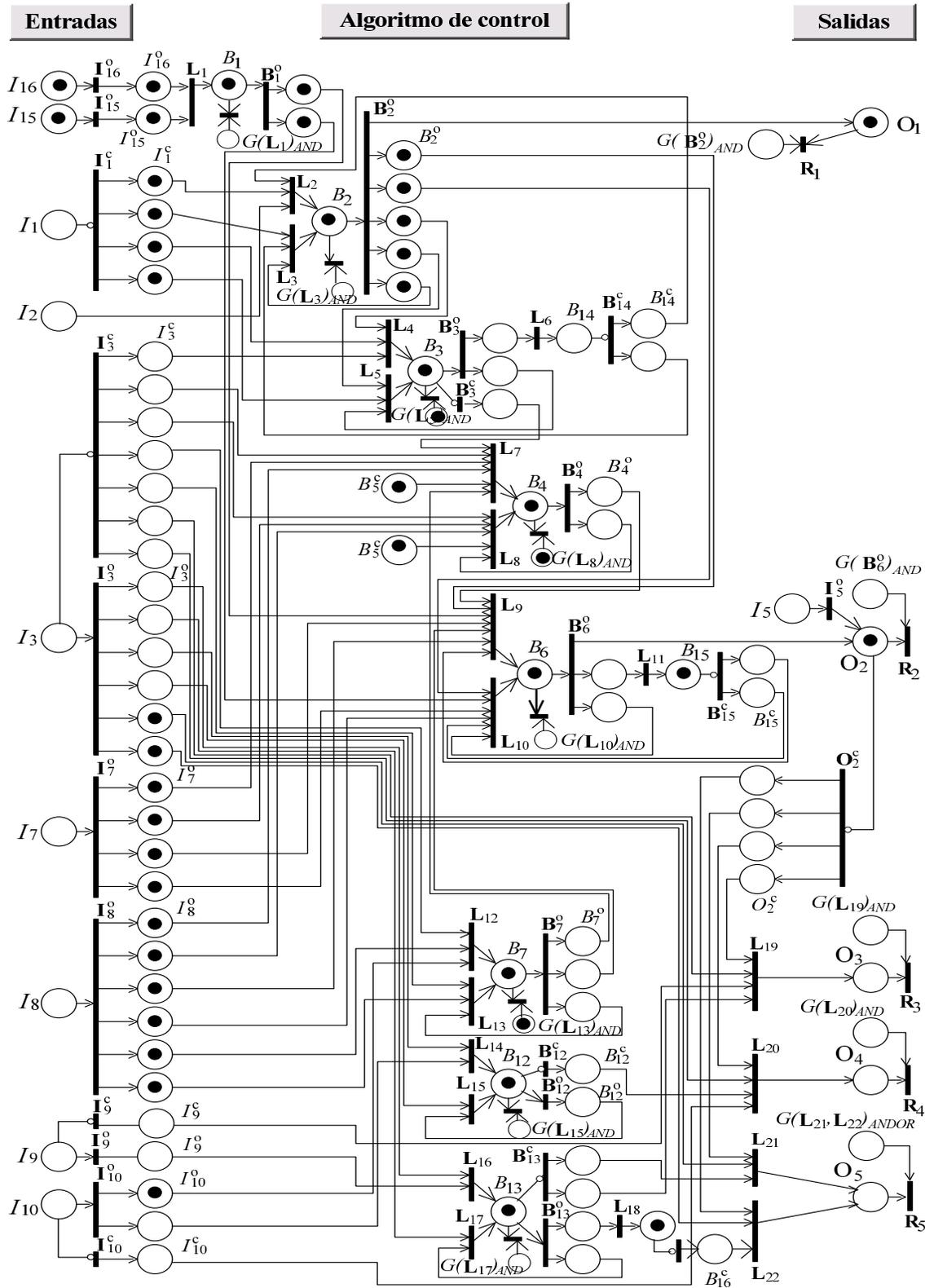


Figura 5.38: LDPN de báscula uno, S.CHAP_B1=1.

5.3.4. Validación de la red LDPN del sistema de envasado

El algoritmo de control del sistema de envasado tiene 16 señales de entrada físicas, por lo que con base en la Ecuación 4.5 se tiene 69360 combinaciones posibles como muestra la Ecuación 5.6, y por la Ecuación 4.5 serían 4,294,901,760 posible combinaciones de falla de circuito-abierto o corto-circuito como muestra la Ecuación 5.7.

$$Ft = 2^{16} * 16 = 1,048,576 \quad (5.6)$$

$$Ft = 4^{16} - 2^{16} = 4,294,901,760 \quad (5.7)$$

Por razones de espacio se omite mostrar el conjunto \mathbf{M}_{Ft} de las combinaciones posibles de falla.

Con base en los resultados de la evaluación de las condiciones de falla en tiempo real con la HMI, la matriz de la Ecuación 5.8 muestra las condiciones de riesgo del sistema de control de envasado.

$$\begin{bmatrix} I_3 & I_4 & I_5 & I_6 & I_7 & I_8 & I_9 & I_{10} & I_{11} & I_{12} & I_{13} & I_{14} & O_1 & O_2 & O_3 & O_4 & O_5 & O_6 & O_7 & O_8 & O_9 \\ 0/1 & 0/1 & sc & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ oc & 0/1 & 0/1 & 0/1 & oc & 0/1 & oc & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ oc & 0/1 & 0/1 & 0/1 & 0/1 & oc & oc & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ oc & 0/1 & 0/1 & 0/1 & oc & 0/1 & 0/1 & oc & 0/1 & 0/1 & 0/1 & 0/1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ oc & 0/1 & 0/1 & 0/1 & 0/1 & oc & 0/1 & oc & 0/1 & 0/1 & 0/1 & 0/1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ sc & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & sc & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & sc & 0/1 & 0/1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0/1 & oc & 0/1 & oc & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & oc & 0/1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0/1 & oc & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & oc & oc & 0/1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0/1 & oc & 0/1 & oc & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & oc & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0/1 & oc & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & oc & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0/1 & sc & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & 0/1 & sc & 0/1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.8)$$

Los observaciones al algoritmo de control de auto-lavado de vehículos son:

- La falla de corto-circuito en las señales I_5 e I_{11} accionan la descarga de las básculas respectivamente, sin embargo, son marcado de operación del sistema de envasado.
- Para la báscula uno, la falla de circuito-abierto de las señales $I_3 = 0$, $I_9 = 0$ e $I_7 = 0$ o $I_3 = 0$, $I_9 = 0$ e $I_8 = 0$ dejan energizada la señal de salida $O_3 = 1$, lo cual es equivalente a no cerrar la compuerta de peso grueso.

- c) De igual forma, la combinación de fallas de las señales $I_3 = 0$, $I_{10} = 0$ e $I_7 = 0$ o $I_3 = 0$, $I_{10} = 0$ e $I_8 = 0$ dejan energizada la bobina de salida O_4 , lo cual es equivalente a no cerrar la compuerta de peso fino de la báscula uno.
- d) La falla de corto-circuito en las señales $I_3 = 1$ e $I_9 = 1$ energizan la bobina de salida O_5 que activa al pistón de alto rendimiento de la báscula uno.
- e) Para la báscula dos, ocurre el mismo comportamiento cuando se presenta falla de circuito-abierto de las señales $I_4 = 0$, $I_{14} = 0$ e $I_6 = 0$ o $I_4 = 0$, $I_{14} = 0$ e $I_{12} = 0$ se queda energizada la señal de salida O_7 , lo cual es equivalente a no cerrar la compuerta de peso grueso.
- f) De igual forma, la combinación de fallas de las señales $I_4 = 0$, $I_{13} = 0$ e $I_6 = 0$ o $I_4 = 0$, $I_{13} = 0$ e $I_{12} = 0$ dejan energizada la bobina de salida O_8 , lo cual es equivalente a no cerrar la compuerta de peso fino de la báscula dos.
- g) La falla de corto-circuito en las señales $I_4 = 1$ e $I_{13} = 1$ energizan la bobina de salida O_9 que activa al pistón de alto rendimiento de la báscula dos.

5.4. Conclusiones del capítulo

Las redes LDPN de los algoritmos de control en LD analizados tienen estructuras PN propuestas en el presente trabajo, comportamientos lógicos como AND, OR, AND-OR, bloqueo, ejecución secuencial, auto-lazos, concurrencia. La transformación de los algoritmos de control en LD a LDPN permite modelar y simular gráficamente el comportamiento de los sistemas.

En particular la LDPN del control de motor reversible muestra la utilización de auto-lazos para mantener con marca el lugar de salida y del bloqueo para evitar la ejecución de dos acciones, que físicamente representa querer cambiar el giro de un motor que ya está en marcha, lo que puede ocasionar daños en el equipo.

La LDPN del algoritmo de control del sistema de lavado de vehículos muestra la necesidad de colocar y drenar marcas en función del comportamiento del sistema, equivalente a energizar y des-energizar bobinas en diferentes etapas del sistema en el LD. La validación de la LDPN de este sistema muestra los riesgos existentes en el control en caso de falla de corto-circuito en sensores que deben detener la marcha de equipo en movimiento.

El sistema de envasado tiene la particularidad de tener en condiciones de funcionamiento, bobinas energizadas como estado inicial. La LDPN de este sistema tiene estructuras PN de lógica AND, OR, AND-OR, ejecución secuencial, bloqueo, concurrencias y auto-lazos.

La simulación en tiempo real de la red LDPN de los algoritmos de control permite modelar el comportamiento de los sistemas a controlar, la actualización del marcado no afecta el comportamiento de la red LDPN ya que es equivalente en el algoritmo de control en LD los contactos NO o NC de una misma señal siempre están activos o no activos.

La validación de la red LDPN es la validación del comportamiento de los algoritmos de control en LD en caso de ocurrir fallas en el sub-sistema de entradas físicas del sistema basado en PLC, lo que ayuda a los responsables del control industrial a tomar acciones en el propio algoritmo o en el sistema (agregar sensores) para evitar que se presente una combinación de las posibles fallas de corto-circuito y/o circuito-abierto que ponga en riesgo a las personas y a la maquinaria.

Conclusiones

Los sistemas basados en PLC se caracterizan por su trabajo cíclico, realizando principalmente las tareas de leer el estado de las señales de entrada físicas que provienen de los sensores montados en las máquinas o procesos, se realiza la copia de los estados de los sensores, con la cual se evalúa el algoritmo de control, los resultados de la evaluación reflejados en un estado para las señales de salidas son enviados a los actuadores mecánicos u eléctricos para modificar las condiciones del proceso. La evaluación del algoritmo de control solo se lleva una vez en cada ciclo con la copia de los estados de las señales de entradas físicas, lo que permite modelar el comportamiento dinámico del proceso en función de la secuencia de activación y/o des-activación de los sensores y en consecuencia en energizar o des-energizar bobinas acopladas a las señales de salidas físicas.

En particular, las bobinas y contactos que son los elementos base del lenguaje de programación LD para el desarrollo de algoritmos de control se pueden representar por lugares y transiciones, elementos de las PN, así como el comportamiento de energizar o des-energizar bobinas con las reglas de habilitado para los lugares de restablecimiento.

Independientemente de el tipo de salida de los sensores y su conexión eléctrica en el sistema basado en PLC, una señal de entrada física solo tiene dos tipos de contactos, NO y NC, los cuales pueden estar en una o más líneas de control en el algoritmo.

Una condición importante y no considerada en ejemplos propuestos en artículos es la condición inicial de los actuadores, NO o NC mecánicamente lo que implica que la bobina esté energizada o des-energizada de inicio, lo que depende de la seguridad del proceso principalmente en condiciones de falla de energía, es decir, que actuadores deben quedarse abiertos o activos y cuales cerrados cuando existe falta de energía eléctrica, o neumática o hidráulicas, entre otras. Este comportamiento debe ser considerado en el diseño de los algoritmos de control, en particular, en el energizado y des-energizado de

las bobinas en LD. La distribución de señales I_i , B_b y O_o propuesta permite modelar el comportamiento de los contactos NO y NC de una señal.

Las PN tienen distintas formas de interpretación para los lugares y transiciones, para el presente trabajo, los lugares son señales de entrada y/o salida que representan a las señales de entrada y salida físicas de los sistemas basados en PLC, y las transiciones como las condiciones lógicas que deben cumplir y que se forman en las líneas de control de los algoritmos en LD.

Comportamientos lógicos de los DES y en particular de los algoritmos de control en LD son modelados con estructuras PN; lógicas AND, OR, AND-OR, Set - Reset, bloqueo, ejecución secuencial, agrupación, concurrencia y auto-lazo formando la base de redes LDPN con mayor complejidad.

Las PN son una herramienta gráfica que permite modelar el comportamiento de los sistemas, al igual que los algoritmos de control en LD son el modelo gráfico estándar del control de un sistema, con la diferencia que los algoritmos de control en LD se desarrollan de forma heurística y con base en la experiencia de los programadores y las PN tiene herramientas de análisis que permiten formalizar los comportamientos dinámicos del sistema, en particular, energizar o des-energizar bobinas.

El diseño de algoritmos de control en LD se conforma de las especificaciones formales del proceso industrial y de la experiencia de los responsables del control industrial, lo que conlleva a la posibilidad de no contemplar situaciones de riesgo en el proceso. Todo sistema basado en PLC tiene la posibilidad de presentar fallas de corto-circuito o circuito-abierto en los sub-sistemas de las señales de entradas y salidas físicas, lo cual genera que las señales correspondientes no cambien de estado, ocasionando comportamientos no deseados en el algoritmo de control y por ende en los procesos industriales.

La red LDPN permite validar el comportamiento de los algoritmos de control en LD en condiciones de falla de corto-circuito y circuito-abierto en el sub-sistema de entradas de los sistemas basados en PLC. El resultado de la evaluación de cada marcado de falla debe ser analizado para tomar la decisión de considerar o no la corrección en el algoritmo de control. Es importante resaltar la necesidad de personal calificado que conozca el proceso lo suficiente para tomar las mejores decisiones en cuanto a que combinaciones de fallas en las entradas se deben evitar.

La simulación de la LDPN se realiza en tiempo real, es decir, a través de una HMI de

la LDPN donde cada marca tiene la función lógica de entrada del sistema o la Ecuación de habilitado del mercado para los lugares de salida y los lugares de restablecimiento para evitar el acumulamiento de marcas. Una característica fundamental para la simulación es considerar la actualización de las marcas en la LDPN cuando es colocada una marca en uno o más lugares de salida de tipo O_o .

La LDPN del motor reversible permite demostrar la utilización de la lógica de boqueo en el control de dos actividades que no pueden ejecutarse al mismo tiempo, y por el tipo de sensores utilizados la necesidad de mantener una señal a través del comportamiento de auto-lazo.

El comportamiento de energizar y des-energizar bobinas se ve reflejado en la simulación de la LDPN de sistemas de auto-lavado de vehículos, debido a que el proceso requiere de movimientos de la máquina más de una vez. Por otro lado, la validación de la red LDPN muestra una gran debilidad en el algoritmo de control en LD en cuanto a seguridad del sistema.

El comportamiento de tener bobinas energizadas como condición de inicio de un proceso por seguridad en caso de falla de energía eléctrica se mostró en la LDPN del sistema de envasado, además de ser un ejemplo con varios comportamientos revisados en el presente trabajo. La validación de la LDPN respectiva también muestra situaciones de riesgo en caso de ocurrir combinaciones de falla de corto-circuito y/o circuito-abierto en el sub-sistema de las señales de entradas físicas del sistema basado en PLC.

La validación de la red LDPN permite visualizar situaciones de riesgo en los algoritmos de control en LD y en consecuencia en los procesos industriales en los casos de ocurrir fallas en el sub-sistema de entradas físicas del sistema basado en PLC, lo anterior, puede ayudar a los responsables del control industrial a tomar acciones lógicas en el propio algoritmo control o en el sistema (agregar sensores, redundancia) para evitar que se presente una combinación de las posibles fallas de corto-circuito y/o circuito-abierto que ponga en riesgo a las personas o a la maquinaria.

Trabajos a futuro

Se considera que el alcance de la presente investigación tiene aún ramas en estudiar o mejorar, por lo que interesa profundizar en:

- Modelar y analizar algoritmos de control en LD que considere bloques de funciones utilizados en el control de procesos industriales, como lo son, temporizadores, contadores, relaciones matemáticas, entre otros, que permita extender el enfoque de la LDPN ordinaria al manejo de datos analógicos.
- Realizar una estructura de trabajo desde una HMI que tome la información desde la memoria de trabajo de los sistemas basados en PLC para evaluar el algoritmo de control en condiciones de falla.
- Realizar interfaces de usuario gráfico en HMI que ayuden en el aprendizaje de los algoritmos de control desarrollados en diagrama escalera y el formalismo de redes de Petri para modelar comportamiento dinámico del control industrial.

Bibliografía

- [1] J. Mendes; J. Creissac; F. Soares; C. Pinto; J. Ferreira;, Simulation and formal verification of industrial systems controllers, ABCM Symposium Series in Mechatronics, vol. 3, pp. 461-470, 2008.
- [2] International Electrotechnical Commission; Programable Controller - Part 3: Programming Languages, IEC 61131-3, Edition 2.0, 2003.
- [3] S. Korotkin; G. Zaidner; B. Cohen; A. Ellenbogen; M. Arad; Y. Cohen;, A Petri net formal design methodology for discrete-event control of industrial automated systems, IEEE 26-th Convention of Electrical and Electronics Engineers in Israel, 2010.
- [4] J. Lee; J. S. Lee;, Conversion of ladder diagram to Petri net using module synthesis technique, International Journal of Modeling and Simulation, vol.29, no.1, pp.79-88, 2009.
- [5] J. Bao; H. Wu; Y. Yan;, A fault diagnosis system-PLC design for system reliability improvement, The International Journal of Advanced Manufacturing Technology, vol. 75, No. 1-4, pp. 523-534, 2015.
- [6] R. Wang; M. Zhou; L. Yin; L. Zhang; J. Sun; G. Ming;, Modeling and validation of PLC-controlled systems: A case study, IEEE Sixth International Symposium on Theoretical Aspects of Software Engineering, pp. 161-166, 2012.
- [7] International Electrotechnical Commission; Programable Controller - Part 8: Guidelines for the application and implementation of programming languages, IEC 61131-8, Edition 2.0, 2003.
- [8] I. Moon;, Modeling programmable logic controllers for logic verification, Automation Science and Engineering, Control Systems, IEEE vol. 14, Issue 2, pp. 53-59, 1994.

- [9] K. Venkatesh; M. C. Zhou; R. J. Caudill;, Comparing ladder logic diagrams and Petri nets for sequence controller design through a discrete manufacturing system, *Transaction on Industrial Electronics, IEEE*, vol. 41, No. 6, pp. 611-619, 1994.
- [10] S. S. Peng; M. C. Zhou;, Petri net based PLC stage programming for discrete-event control design, *Systems, Man, and Cybernetics, IEEE International Conference on*, vol. 4, pp. 2706 -2710, 2001.
- [11] A. Taholakian; W. M. M. Hales;, PN-PLC: A methodology for designing, simulating and coding PLC based control systems using Petri nets, *International Journal of Production Research*, vol. 35, No. 6, pp. 1743-1762, 1997.
- [12] M. Uzam; A. H. Jones;, Discrete event control system design using automation Petri nets and their ladder diagram implementation, *The International Journal of Advanced Manufacturing Technology*, vol. 14, pp. 716-728, 1998.
- [13] G. Frey; L. Litz;, Verification and validation of control algorithms by coupling of interpreted Petri nets, *Systems, Man, and Cybernetics, IEEE International Conference on*, vol. 1, pp. 7-12, 1998.
- [14] G. Frey;, Analysis of Petri nets based control algorithms basic properties, *American Control Conference, Proceedings of*, pp. 3172-3176, 2000.
- [15] S. G. Tzafestas; M. G. Pantelelis; D. L. Kostis;, Desing and implementation of a logic controller using Petri nets and ladder logic diagrams, *Systems Analysis Modelling Simulation, Taylor and Francis*, vol. 42, pp. 135-167, 2002.
- [16] G. Music; D. Matko;, Discrete event control theory applied to PLC programming, *Automatika*, vol. 43, pp. 21-28, 2002.
- [17] G. Zapata; E. Carrasco;, Estructuras generalizadas para PLC modelados mediante redes de Petri, *Revista Dyna*, vol. 135, pp. 66-74, 2002.
- [18] S. S. Peng; M. C. Zhou;, Sensor-based Petri net modeling for PLC stage programming discrete-event control design, *Robotics and Automation, Proceedings. ICRA '02. IEEE International Conference on*, vol. 2, pp. 1907-1912, 2002.
- [19] G. Frey;, Software quality in logic controller, *Systems, Man and Cybernetics, IEEE International Conference on*, vol. 1, pp. 515-520, 2002.

- [20] M. B. Younis; G. Frey;, Formalization of existing PLC programs: A survey, The Multiconference on Computational Engineering in Systems Applications, Proceedings of, pp. 0234-0239, 2003.
- [21] G. B. Lee; H. Zandogg; J. S. Lee;, Automatic generation of ladder diagram with control Petri nets, Journal of Intelligent Manufacturing, vol. 15, pp. 245-252, 2004.
- [22] T. Suesut; P. Inban; P. Nilas; P. Rerngreun; S. Gulphanich;, Interpretation Petri net model to IEC 1131-3: LD for programmable logic controller, Robotics, Automation and Mechatronics, IEEE Conference on, vol. 2, pp. 1107-1111, 2004.
- [23] S. S. Peng; M. C. Zhou;, Ladder diagram and Petri-net-based discrete-event control design methods, Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on , vol.34, no.4, pp. 523,531, Nov. 2004.
- [24] J. S. Lee; P. L. Hsu;, An improved evaluation of ladder logic diagrams and Petri nets for the sequence controller desing in manufacturing systems, International Journal on Advance Manufacturing Technology, vol. 24, pp. 279-287, 2004.
- [25] J. S. Lee; P. L. Hsu;, A systematic approach for the sequence controller design in manufacturing systems; International Journal of Manufacturing Technology, vol. 25, pp. 754-760, 2005.
- [26] D. Thapa; S. Dangol; G. N. Wang;, Transformation from Petri nets model to programmable logic controller using one-to-one mapping technique, Computational Intelligence for Modelling, Control and Automation, and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on, vol. 2, pp. 228-233, 2005.
- [27] G. Frey; F. Wagner;, A toolbox for the development of logic controllers using Petri nets, The 8th International Workshop on Discrete Event Systems, Proceedings of, vol. 1, pp.473-474, 2006.
- [28] C. M. Park; S.M. Bajimaya; S.C. Park; G. N. Wang; J. G. Kwak; K. H. Han; M. Chang;, Development of virtual simulator for visual validation of PLC program, Computational Intelligence for Modelling, Control and Automation, and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on, 2006.
- [29] D. Thapa; C. M. Park; S. Dangol; G. Wang;, III-Phase verification and validation of IEC standard programmable logic controller, International Conference on

- Computational Intelligence for Modelling Control and Automation, and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, 2006.
- [30] S. C. Park; C. M. Park; G. Wang; PLCStudio: Simulation based PLC code verification, Winter Simulation Conference, pp. 222-228, 2008.
- [31] A. Castelnovo; L. Ferrarini; L. Piroddi;, An incremental Petri net-based approach to the modeling of production sequences in manufacturing systems, Automation Science and Engineering, IEEE Transactions on, vol. 4, No. 3, pp. 424-434, 2007.
- [32] O. Ljungkrantz; K. Åkesson;, M. Fabian;, Formal specification and verification of components for industrial logic control programming, Automation Science and Engineering, IEEE International Conference on, pp. 935-940, 2008.
- [33] J. I. Tsai; C. C. Teng;, Constructing an abstract model for ladder diagram diagnosis using Petri nets, Asian Journal of Control, vol. 12, No. 3, pp. 309-322, 2010.
- [34] I. Grobelna; M. Grobelny; M. Adamski;, Petri nets and activity diagrams in logic controller specification - transformation and verification, 17th International Conference Mixed Design of Integrated Circuits and Systems, pp. 607-612, 2010.
- [35] A. H. Malik; T. Mehmood; M. A. Choudhry; A. Hanif;, A generic procedure for troubleshooting of PLC based control systems, IEEE 11th International Conference of Control, Automation, Robotics and Vision, pp. 732-737, 2010.
- [36] V. Năvrăpescu; M. Popescu; I. D. Deaconu; A. I. Chirilă;, Designing the control of an electrical drive using Petri nets, 7th International Symposium on Advanced Topics in Electrical Engineering (ATEE), 2011.
- [37] E. A. da Silva Oliveira; L. D. da Silva; K. Gorgonio; A. Perkusich; A. F. Martins;, Obtaining formal models from Ladder diagrams, Industrial Informatics (INDIN), 2011 9th IEEE International Conference on, pp. 796-801, 2011.
- [38] T. Alenljung; B. Lennartson; M. N. Hosseini;, Sensor graphs for discrete event modeling applied to formal verification of PLCs, Control Systems Technology, IEEE Transactions on, vol. 20, No. 6, pp. 1506-1521, 2012.
- [39] X. Chen; J. Luo; P. Qi;, Method for translating ladder diagrams to ordinary Petri nets, Decision and Control (CDC), IEEE 51st Annual Conference on, pp. 6716-6721, 2012.

- [40] H. Barbosa; D. Déharbe;, Formal verification of PLC programs using the B method, *Lecture Notes Computer Science*, vol. 7316, pp. 353-356, 2012.
- [41] R. Wang; M. Zhou; L. Yin; L. Zhang; J. Sun; G. Ming;, Modeling and validation of PLC-controlled systems: A case study, *IEEE 6th International Symposium on Theoretical Aspects of Software Engineering*, pp. 161-166, 2012.
- [42] S. Qin; G. Wang;, A study of fault detection and diagnosis for PLC controlled manufacturing system, *IEEE International Conference Semantic Computing, Part I*, pp. 373–382, 2012.
- [43] Z. Wu; S. Hsieh; A realtime fuzzy Petri net diagnoser for detecting progressive faults in PLC based discrete manufacturing system, *International Journal Advance Manufacturing Technology*, vol. 61, pp. 405-421, 2012.
- [44] L. Huai; C. Cheng;, Reliability design of PLC-based Control System, *IEEE 9th International Conference on Natural Computation*, pp. 1671-1675, 2013.
- [45] E. V. Kuzmin; V. A. Sokolov;, On construction and verification of PLC programs, *Automatic Control and Computer Sciences*, vol. 47, No. 7, pp. 443–451, 2013.
- [46] J. C. Quezada; J. Medina; E. Flores; J. C. Seck Tuoh; N. Hernández;, Formal design methodology for transforming ladder diagram to Petri nets, *The International Journal of Advanced Manufacturing Technology*, vol. 73, No. 5-8, pp. 821-836, 2014.
- [47] J. Bao;, H. Wu;, Y. Yan;, A fault diagnosis system-PLC design for system reliability improvement, *The International Journal of Advanced Manufacturing Technology*, published online, july 2014.
- [48] T. Suesut; P. Inban; P. Nilas; P. Rerngreun; S. Gulphanich;, Interpretation Petri net model to IEC 1131-3: LD for programmable logic controller, *Robotics, Automation and Mechatronics, IEEE Conference on*, 2004.
- [49] M. M. Gomaa;, Petri net to ladder logic diagram converter and a batch process simulation, *ARNP Journal of Engineering and Applied Sciences*, 2011.
- [50] www.PLCopen.org
- [51] Karl-Heinz John;, Micahel Tiegelkamp, *IEC 61131-3: Programming industrial automation systems: concepts* , Second Edition, Springer, 2010.

- [52] International Electrotechnical Commission; Programmable Controller - Part 2: Equipment requirements and tests, IEC 61131-2, Edition 2.0, 2003.
- [53] International Electrotechnical Commission, Programmable Controller - Part 1: General information, IEC-61131-1, Edition 2.0, 2003.
- [54] J. C. Quezada;, J. Medina;, E. Flores;, Simulación mediante HMI-PLC de las características en redes de Petri de sistemas de eventos dinámicos, 2^{do} Congreso Nacional de Tecnologías de la Información, pp. 149 - 155, ISBN: 978-607-482-324-0, 2013.
- [55] J. Carlos Quezada;, Joselito Medina; y Ernesto Flores;, Analysis of Ladder Diagram for a Packaging System with Petri Nets, XIII Congreso Nacional de Ingeniería Electromecánica y de Sistemas CNIES, IPN, Noviembre de 2012.
- [56] Murata T.;, Petri nets: Properties, analysis and applications, Proceedings of the IEEE , vol.77, no.4, pp.541,580, 1989.
- [57] A. Guasch;, M. A. Piera;, J. Casanovas;, J. Figueras;, Modelado y simulación - Aplicación a procesos logísticos de fabricación y servicios, Alfaomega, ISBN: 970-15-0995-1, 2005.
- [58] Zhang, H., Jiang, Y., Hung, W. N., Yang, G., Gu, M., and Sun, J. New strategies for reliability analysis of Programmable Logic Controllers, Mathematical and Computer Modelling, vol. 55(7/8), pp. 1916-1931, 2012
- [59] E. Mandado;, J. M. Acevedo;, C. Fernández;, J. I. Armesto;, Autómatas Programables y Sistemas de Automatización, Segunda edición, Alfaomega, ISBN: 978-607-7686-73-6