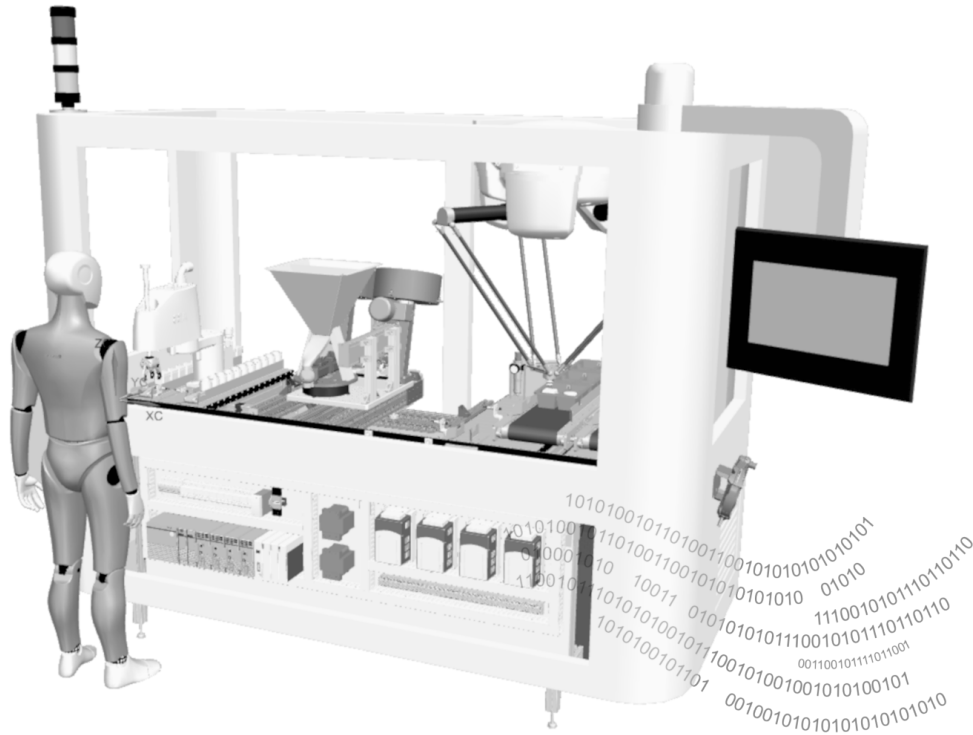




UNIVERSIDAD
NACIONAL
DE COLOMBIA



Virtual Factory Test of a Cyber-Physical Modular Production System Station

Validación y verificación de un sistema de manufactura virtual bajo control discreto con funcionalidades ciberfísicas

Trabajo final de Maestría en Ingeniería Mecánica

UBALDO GARCIA ZARAGOZA

TRABAJO FINAL DE MAESTRÍA 2020

Virtual Factory Test of a Cyber-Physical Modular Production System Station

Validación y verificación de un sistema de manufactura virtual bajo
control discreto con funcionalidades ciberfísicas

UBALDO GARCIA ZARAGOZA

Trabajo de grado presentado como requisito parcial para optar al título de:
Magister en Ingeniería Mecánica

Director:

Víctor Hugo Grisales Palacio, Ph.D

Co-Director:

Pedro Fabián Cárdenas Herrera, Ph.D

Línea de Investigación:

Automatización, Control y Mecatrónica

Grupo de Investigación:

UNRobot

Facultad de Ingeniería

Departamento de Ingeniería Mecánica y Mecatrónica

UNIVERSIDAD NACIONAL DE COLOMBIA

Bogotá, Colombia 2020

Virtual Factory Test of a Cyber-Physical Modular Production System Station.

Validación y verificación de un sistema de manufactura virtual bajo control discreto con funcionalidades ciberfísicas.

© UBALDO GARCIA ZARAGOZA, 2020.

Director: Victor Hugo Grisales Palacio, PhD
Co-Director: Pedro Fabian Cardenas Herrera, PhD

Trabajo Final de Maestría 2020
Facultad de Ingeniería
Departamento de Ingeniería Mecánica y Mecatrónica
Universidad Nacional de Colombia, Bogotá D.C.
Grupo de Investigación (UNRobot)
Correo: ugarciaz@unal.edu.co

Portada: Estación de producción modular virtual para validar los programas de control discreto en un gemelo digital de un controlador con funcionalidades ciberfísicas.

Escrito en L^AT_EX
Impreso en PDF
Bogotá, Colombia 2020

Agradecimientos

De las etapas formativas de esta trabajo al proyecto final, tengo una inmensa deuda de gratitud con mi director Victor Hugo Grisales, PhD y mi co-director Pedro Fabian Cardenas, PhD por sus sólidos consejos y cuidadosa orientación los cuales fueron invaluable al examinar la validación y verificación de un sistema de manufactura virtual bajo control discreto con funcionalidades ciberfísicas.

Para mi esposa Nathalia Bran Castro y mi hijo Matias García Bran por su apoyo, estímulo, paciencia y amor inquebrantable. Su tolerancia a mi ocasional falta de atención hacia ellos es un testimonio en sí mismo de su inquebrantable devoción y amor.

Deseo agradecer a todos mis amigos de XMARTPLM SAS por apoyar en todo momento, y en especial agradecer a Julián Mora Orozco por alentarme y apoyarme durante mi estadía en la universidad.

Por último, pero no menos importante, me gustaría agradecer a mi familia en México por su paciencia y comprensión durante toda esta nueva etapa. Especialmente a mis padres Ubaldo García Quezada y Rosita Zaragoza Orozco por todo el apoyo y los consejos brindados durante estos años.

Ubaldo Garcia Zaragoza, Bogotá, Colombia 2020

Validación y verificación de un sistema de manufactura virtual bajo control discreto con funcionalidades ciberfísicas

UBALDO GARCIA ZARAGOZA

Departamento de Ingeniería Mecánica y Mecatrónica
Universidad Nacional de Colombia

Resumen

El propósito de este trabajo final de maestría es definir pautas para validar y verificar un sistema de manufactura con elementos como: procesos, máquinas, sensores y actuadores por medio de un gemelo digital y de esta manera validar el control discreto con funcionalidades ciberfísicas. El enfoque propuesto puede ser usado con modelos mecatrónicos de sistemas existentes o en modelos de concepto para asegurar la operación correcta y segura del sistema.

Mediante el uso de gemelos digitales como el de un controlador virtual y un sistema de manufactura virtual se valida y verifica la lógica de programación de funcionamiento del control discreto y coordinación de movimiento aplicando el método GRAFCET, LADDER y la guía GEMMA de modos de marcha y parada. En contraste con la puesta en marcha tradicional, la puesta en marcha del sistema con funcionalidades ciberfísicas, abre una oportunidad al área de automatización debido a que al incorporar un sistema SCADA con funcionalidades web, permitiendo acceso a la información tanto de escritura como de lectura en múltiples plataformas y dispositivos logrando conectar sistemas físicos y virtuales en un mismo entorno.

En este documento se presenta las pautas para implementar sistemáticamente la validación y verificación de un sistema de manufactura virtual bajo control discreto con funcionalidades ciberfísicas. Para su aplicación, se propusieron 3 módulos que constituyan un sistema de manufactura virtual considerando criterios de integración y modularidad representando un sistema de producción real. Inicialmente se mostraron las metodologías utilizadas para el desarrollo del proyecto, buscando estandarizar y crear unas guías para la puesta en marcha virtual pensadas desde el criterio de modelado en un software de diseño asistido por computador, factores importantes en las comunicaciones industriales y conceptos de programación teniendo en cuenta modos seguros de operación.

Palabras clave: puesta en marcha virtual, sistema modular de producción, gemelo digital, comunicaciones industriales.

UBALDO GARCIA ZARAGOZA

Department of Mechanical and Mechatronics Engineering
Universidad Nacional de Colombia

Abstract

The purpose of this final master's work is to define guidelines to validate and verify a manufacturing system with elements such as: processes, machines, sensors and actuators through a digital twin and thus validate the discrete control with cyber-physical functionalities. The proposed approach can be used with mechatronic models of existing systems or in concept models to ensure the correct and safe operation of the system.

Through the use of digital twins such as a virtual controller and a virtual manufacturing system, the discrete control and motion coordination operation programming logic is validated and verified by applying the GRAFCET, LADDER method and the GEMMA guide for gait modes and stop. In contrast to the traditional start-up, the start-up of the system with cyber-physical functionalities opens an opportunity to the automation area because it incorporates a SCADA system with web functionalities, allowing access to both writing and reading information in multiple platforms and devices connecting physical and virtual systems in the same environment.

This document presents the guidelines to systematically implement the validation and verification of a virtual manufacturing system under discrete control with cyber-physical functionalities. For its application, 3 modules were proposed that constitute a virtual manufacturing system considering integration and modularity criteria, representing a real production system. Initially, the methodologies used for the development of the project were shown, seeking to standardize and create guides for virtual start-up thought from the modeling criteria in computer-aided design software, important factors in industrial communications and programming concepts having consider safe modes of operation.

Key words: virtual commissioning, modular production system, digital twin, industrial communications.

Contenido

Lista de Figuras	XIII
Lista de Tablas	XV
1. Introducción	1
1.1. Marco teórico	1
1.2. Justificación	9
1.3. Pregunta de investigación	11
1.4. Aplicación industrial	12
1.5. Alcances y delimitaciones	15
2. Metodologías	16
2.1. Enfoque de investigación	16
2.2. Infraestructura virtual	17
2.3. Comunicaciones industriales	21
2.4. Modos seguros y de operación	23
2.5. Validación y verificación virtual	27
3. Implementación	29
3.1. Infraestructura virtual de la aplicación	29
3.2. Arquitectura de conectividad de la aplicación	39
3.3. Modo de producción de la aplicación	46
3.4. Modo seguro de la aplicación	53
4. Resultados y Análisis	57
4.1. Validación y verificación de la aplicación	57
4.2. Consideraciones de desempeño al simular	63
4.3. Aplicaciones derivadas de la investigación	65
5. Conclusiones	66
5.1. Conclusiones	66
5.2. Trabajo futuro	67
Bibliografía	69
A. Anexo: Graficets a nivel automata del sistema de manufactura	I
B. Anexo: Guía validación y verificación virtual - Física Básica	VI

C. Anexo: Guía validación y verificación virtual - Automatización	XVIII
D. Anexo: Guía validación y verificación virtual - Lógica/Control	XXIX
E. Anexo: Guía validación y verificación virtual - Comunicaciones	XLI
F. Anexo: Ladder - Rutina Automática	LXV
G. Anexo: Ladder - Rutina SCARA Home	LXVII
H. Anexo: Ladder - Rutina SCARA Automática	LXX
I. Anexo: Ladder - Rutina Robot Cartesiano Home	LXXVII
J. Anexo: Ladder - Rutina Robot Cartesiano Automática	LXXXI
K. Anexo: Ladder - Rutina Robot DELTA Home	LXXXV
L. Anexo: Ladder - Rutina Robot DELTA Automática	LXXXVIII

Lista de Figuras

1.1.	Ejemplo de gráfico GRAFCET.[17]	2
1.2.	Gráfico GEMMA.[10]	2
1.3.	Evolución de la industria en revoluciones industriales.[16]	3
1.4.	Niveles de los gemelos digitales.[3]	4
1.5.	Celda robótica en ABB RobotStudio Universidad de Chalmers.[5]	4
1.6.	Celda con robot industrial en Simumatik3D Universidad de Skövde.[5]	5
1.7.	Arquitectura de validación del sistema.[5]	6
1.8.	Flujo de trabajo para validación virtual.[4]	7
1.9.	Ingeniería basada en modelos con funcionalidades ciberfísicas [11]	8
1.10.	Modelo virtual en NX MCD de Siemens PLM® [Fuente Autor]	9
1.11.	Sistema de Manufactura Modular.[Autor]	12
1.12.	Estación de almacén.[Autor]	13
1.13.	Banda transportadora principal.[Autor]	13
1.14.	Estación de llenado-tapado.[Autor]	14
1.15.	Estación de despacho.[Autor]	14
2.1.	Metodología de Investigación.[Autor]	17
2.2.	Flujo de validación virtual de sistemas de manufactura.[Autor]	18
2.3.	Flujo de construcción del modelo digital 3D.[Autor]	19
2.4.	Flujo de virtualización de automata programable.[Autor]	20
2.5.	Esquema de comunicaciones de un sistema ciberfísico.[Autor]	21
2.6.	Arquitectura de comunicaciones de un sistema ciberfísico.[Autor]	22
2.7.	Estructuras GRAFCET: a)Lineal b)Selectiva c)Simultaneidad.[Autor]	23
2.8.	Coordinación entre diagrama principal y secundario.[Autor]	24
2.9.	Representación gráfica Guía GEMMA Genérica.[Autor]	25
2.10.	Flujo de validación, verificación y aceptación.[Autor]	27
2.11.	Flujo de validación del sistema de manufactura virtual.[Autor]	28
3.1.	Ensamble y subensambles del sistema de manufactura virtual.[Autor]	29
3.2.	Cuerpos rígidos y de colisión del sistema de manufactura virtual.[Autor]	30
3.3.	Parámetros de configuración cuerpos de colisión.[Autor]	31
3.4.	Parámetros de configuración de cuerpos rígidos.[Autor]	31
3.5.	Esquema de cuerpos estáticos y cadenas cinemáticas.[Autor]	32
3.6.	Configuración de cuerpos estáticos y cadenas cinemáticas.[Autor]	33
3.7.	Configuración de uniones en cadenas cinemáticas.[Autor]	34
3.8.	Configuración de unión para aplicaciones pick and place.[Autor]	34

3.9.	Esquema de sensores y actuadores del sistema virtual.[Autor]	35
3.10.	Configuración de un actuador del sistema virtual.[Autor]	36
3.11.	Configuración de un sensor del sistema virtual.[Autor]	36
3.12.	Configuración de banda transportadora del sistema virtual.[Autor]	37
3.13.	Creación y configuración de un autómeta industrial virtual.[Autor]	37
3.14.	Configuración del driver de comunicaciones virtual.[Autor]	38
3.15.	Arquitectura de comunicaciones industriales.[Autor]	39
3.16.	Configuración de servidor OPC DA en RSLinx Classic.[Autor]	40
3.17.	Configuración de servidor OPC DA en Ignition.[Autor]	41
3.18.	Configuración de servidor OPC UA de Ignition.[Autor]	41
3.19.	Configuración de la seguridad del servidor OPC UA.[Autor]	42
3.20.	Exposición de variables en el diseñador de Ignition.[Autor]	42
3.21.	Tiempo de muestreo de variables de Ignition.[Autor]	43
3.22.	Configuración del adaptador de señales en NX MCD.[Autor]	44
3.23.	Configuración de conexiones externas en NX MCD.[Autor]	45
3.24.	Mapeo de señales entre el servidor OPC UA y NX MCD.[Autor]	46
3.25.	Grafcet Nivel 1 Estación de almacén.[Autor]	47
3.26.	Configuración de ejes virtuales y sistemas coordinados.[Autor]	48
3.27.	Estructura de programación de bloque MCLM.[Autor]	48
3.28.	Grafcet Nivel 1 Estación de llenado-tapado.[Autor]	49
3.29.	Zonas de sincronización de llenado al vuelo.[Autor]	49
3.30.	Esquema de sincronización de llenado al vuelo.[Autor]	50
3.31.	Parámetros de velocidad y perfil en zona para el encuentro.[Autor]	51
3.32.	Grafcet Nivel 1 Estación de despacho.[Autor]	52
3.33.	Zonas de sincronización de empacado al vuelo.[Autor]	53
3.34.	Esquema de parada pedida a final de ciclo.[Autor]	54
3.35.	Esquema de marcha de verificación con orden.[Autor]	54
3.36.	Esquema de parada de emergencia.[Autor]	55
3.37.	Grafcet Nivel 1 principal aplicando guía GEMMA.[Autor]	55
3.38.	HMI para accionar y evidenciar estados de la guía GEMMA.[Autor]	56
4.1.	Validación de actuadores virtuales del modelo digital 3D.[Autor]	57
4.2.	Validación de sensores virtuales del modelo digital 3D.[Autor]	58
4.3.	Validación del adaptador de señales del modelo digital 3D.[Autor]	58
4.4.	Validación de la lógica del programa en el controlador virtual.[Autor]	59
4.5.	Validación de HMI y servidor OPC DA del controlador virtual.[Autor]	60
4.6.	Validación de procedimientos de funcionamiento en HMI.[Autor]	60
4.7.	Validación de procedimientos de puesta en marcha en HMI.[Autor]	61
4.8.	Validación de procedimientos de funcionamiento GEMMA.[Autor]	61
4.9.	Validación de procedimientos de funcionamiento llenado.[Autor]	62
4.10.	Validación de procedimientos de parada en HMI.[Autor]	62
4.11.	Rendimiento computacional del equipo utilizado.[Autor]	63
4.12.	Escalamiento del tiempo de simulación.[Autor]	64
4.13.	Error de precisión y exactitud de la simulación.[Autor]	64
4.14.	Aplicaciones en la nube de sistemas ciberfísicos.[Autor]	65

Lista de Tablas

2.1.	Consideraciones de desempeño en cuerpos de colisión.	20
2.2.	Tipos de transiciones o receptividades.	24
2.3.	GEMMA Grupo A: Proceso en parada o puesta en marcha.	25
2.4.	GEMMA Grupo F: Proceso en funcionamiento.	26
2.5.	GEMMA Grupo D: Procedimientos de defecto.	26
3.1.	Árbol de ensamble del sistema de manufactura virtual.	30
3.2.	Listado de sensores y actuadores.	35
3.3.	Listado de señales de salida del sistema de manufactura.	44
3.4.	Listado señales de entradas del sistema de manufactura.	45
4.1.	Proyectos curso de automatización de procesos de manufactura	65

1

Introducción

En este capítulo se muestran los antecedentes del proyecto, su porqué, propósito, preguntas de investigación, se describe el caso de aplicación industrial, tanto en su funcionamiento como aspectos relevantes de operación, así como, los alcances y delimitaciones del proyecto.

1.1. Marco teórico

Cuando la industria se vio en la necesidad de integrar máquinas, herramientas, energía y trabajo para producir y transformar materia prima o material, formó los sistemas de manufactura. Estos sistemas se clasificaron en dos categorías: aquellos que son medidos por variables de carácter físico o químico cuyo rango es continuo, denominados procesos continuos y aquellos que son medidos por escalas de conteo o numeración donde su producción son piezas, partes y ensamblajes, denominados procesos discretos[14]. Independientemente del caso, aumentar la productividad de estos sistemas de manufactura ha generado la necesidad de la construcción y el diseño de máquinas que sean capaces de reproducir los movimientos de las personas y/o simplifiquen las tareas de estos sistemas, dando origen a lo que denominaremos como sistemas de manufactura automáticos bajo control continuo o discreto acorde a la naturaleza del proceso.

En los sistemas de manufactura automáticos interactúan sistemas mecánicos, eléctricos, electrónicos y en algunos casos la supervisión y/o manipulación de personas para hacerlos funcionar o verificar su desempeño. La mayoría de estos sistemas para que operen en su totalidad o parcialmente por sí solos, necesitan las instrucciones de un control lógico programable (PLC, por sus siglas en inglés), estos componentes se pueden programar bajo el estándar IEC 1131, de los métodos propuestos en este estándar es el diagrama GRAFCET de control con etapas y transiciones (Graphe Fonctionnel de Commande Etape Transition, por sus siglas en francés) a manera de ejemplo ver la figura 1.1, siendo un grafo funcional de control muy útil para representar funciones de control discreto en procesos de esta naturaleza o de control secuencial [15]. Este tipo de representación gráfica tiene gran aplicación en la automatización de sistemas de manufactura de procesos discretos ya que permite describir los procesos a automatizar, considerando las acciones a ejecutar, y los procesos intermedios que generan estas acciones basándose en una sintaxis simple, que ofrece de manera sistémica el desarrollo del automatismo.

1. Introducción

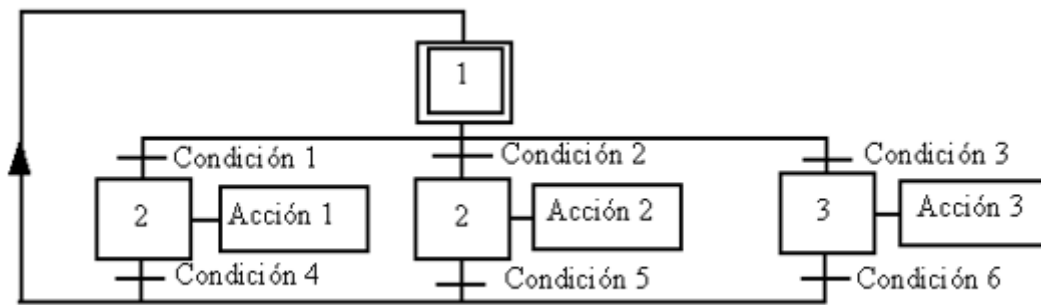


Figura 1.1: Ejemplo de gráfico GRAFCET.[17]

De igual manera a la programación, para validar los modos seguros de arranque, marcha y parada de sistemas de manufactura automáticos, es necesaria una herramienta que genere especificaciones para evitar inconsistencias e imprevistos [7][8], para ello el grupo de investigación ADEPA (Agence pour le Développement of Appliquée Productique, por sus siglas en francés) desarrolló GEMMA una guía que busca identificar todos los estados en los que un proceso de automatización puede evolucionar. GEMMA es una guía gráfica, que consiste en unas zonas que definen los estados que pueden ocurrir en los sistemas de manufactura automáticos como se observa en la figura 1.2.

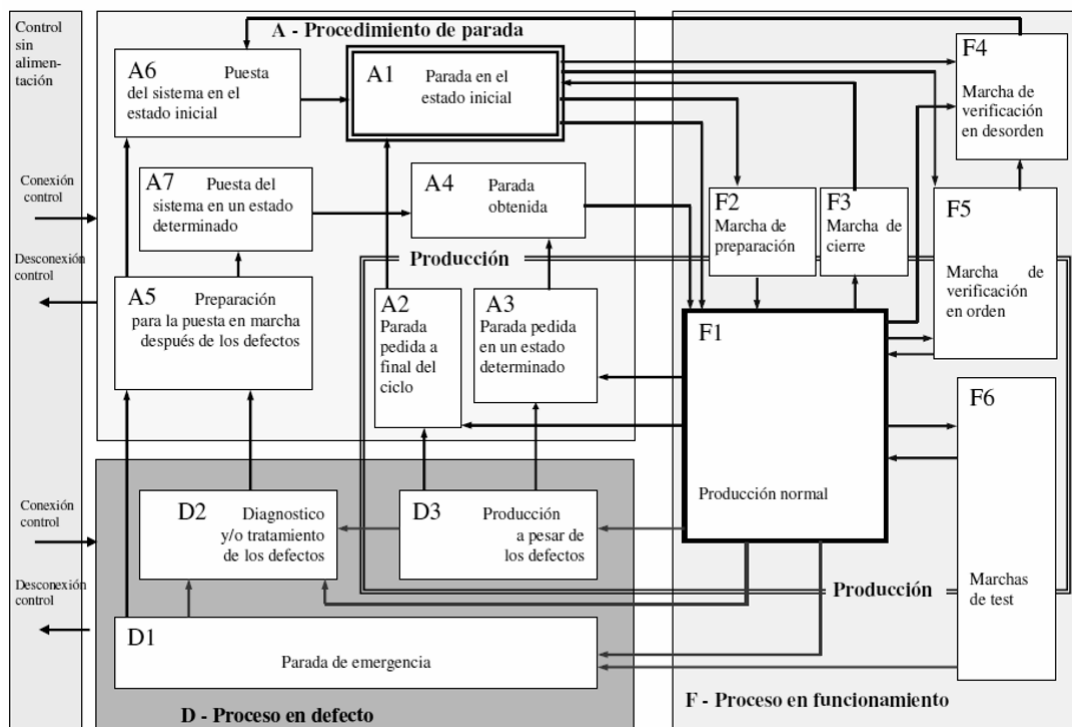


Figura 1.2: Gráfico GEMMA.[10]

Una vez se generan los programas para el sistema de manufactura automático, en la actualidad se llevan a cabo dos pruebas de validación, la prueba FAT (Factory

Acceptance Test, por sus siglas en inglés) la cual es desarrollada en el entorno de fabricación buscando probar el sistema además de desarrollar manuales, planos eléctricos, etc. y la prueba SAT (Site Acceptance Test, por sus siglas en inglés) la cual busca repetir en sitio las pruebas FAT para garantizar que todo opera correctamente después de la instalación. Con la llegada de la cuarta revolución industrial (Industry 4.0 denominado en el inglés) se da inicio a una nueva era donde es posible registrar y procesar datos de manera digital, abriendo la oportunidad de unir los procesos de manufactura de una manera en la que los sistemas físicos y virtuales puedan comunicarse mediante redes interconectadas, creando una tercera prueba denominada VFAT (Virtual factory acceptance test, por sus siglas en inglés) la cual busca validar la lógica del PLC, ubicación de los componentes y funcionalidad, además de identificar posibles errores de software y construcción antes de su fabricación y/o operación [13].

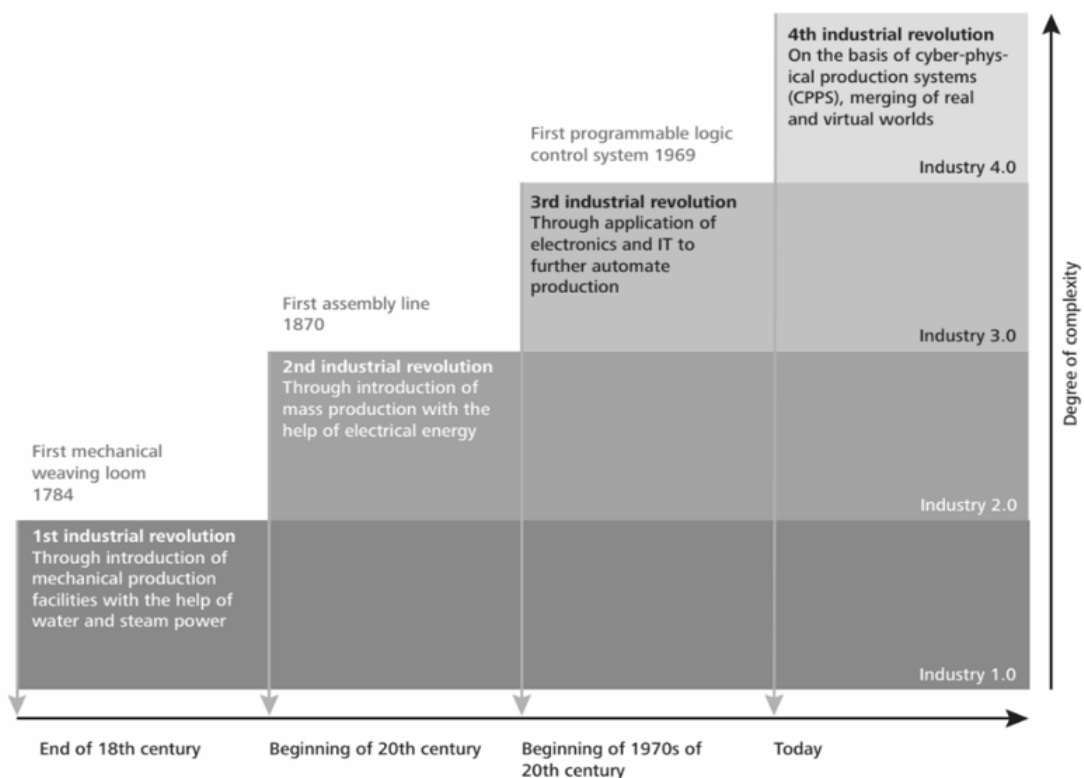


Figura 1.3: Evolución de la industria en revoluciones industriales.[16]

Como se observa en la figura 1.3, la base de la cuarta revolución industrial son los sistemas de manufactura con funcionalidades ciberfísicas los cuales no solamente ayudarán a mejorar el diseño del sistema sino también mejorar la planificación de procesos [16]. Acorde al autor Azad et al [3] un gemelo digital puede definirse como un modelo virtual dinámico, de un sistema, proceso o servicio. Por ello un gemelo digital puede representar tanto elementos de un modelo tridimensional de un sistema de manufactura, como un autómatas industrial virtual emulando sus funciones a nivel de software sin requerir hardware específico.

1. Introducción

Level	Model Sophistication	Physical Twin	Data Acquisition from Physical Twin	Machine Learning (Operator Preferences)	Machine Learning (System/Environment)
1 Pre-Digital Twin	virtual system model with emphasis on technology/technical-risk mitigation	does not exist	Not applicable	No	No
2 Digital Twin	virtual system model of the physical twin	exists	performance, health status, maintenance; batch updates	No	No
3 Adaptive Digital Twin	virtual system model of the physical twin with adaptive UI	exists	performance, health status, maintenance; real-time updates	Yes	No
4 Intelligent Digital Twin	virtual system model of the physical twin with adaptive UI and reinforcement learning	exists	performance, health status, maintenance, environment; both batch/real-time updates	Yes	Yes

Figura 1.4: Niveles de los gemelos digitales .[3]

En la figura 1.4 se observan los diferentes niveles en que se pueden catalogar los gemelos digitales y su nivel de sofisticación acorde a sus funcionalidades ciberfísicas, adquisición de datos y procesamiento de información [3]. Con ello se buscan controlar y basar los sistemas de manufactura automáticos en la información de su funcionamiento y del entorno físico, mediante la conexión y procesamiento de sus datos de manera digital.

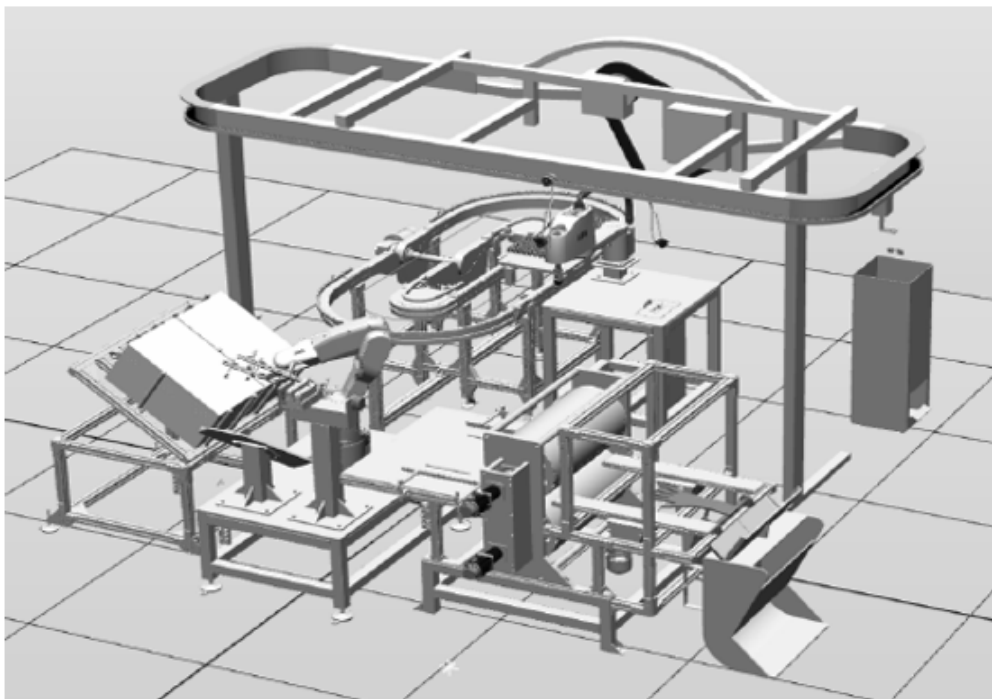


Figura 1.5: Celda robótica en ABB RobotStudio Universidad de Chalmers .[5]

Ejemplos en este campo se han realizado en universidades como Chalmers en Gotemburgo, Suecia, donde se ha trabajado en la validación de un sistema de manufactura en la cual se integra una celda robótica en el software ABB RobotStudio [13] (Ver figura 1.5). Esta validación se realizó de manera virtual sin integrar equipos físicos, buscando resaltar la importancia de validar virtualmente sistemas de manufactura automáticos para ganar experiencia y conocimiento de los procesos. También, en la universidad de Skövde en Suecia, se desarrolló un proyecto conjuntamente con Volvo GTO, donde se buscaba emular robots industriales y PLCs en el software Simumatik3D [9] como se muestra en la figura 1.6, validando la programación inicialmente de forma virtual y posteriormente de manera tradicional para finalmente comparar sus comportamientos ante un escenario de falla.

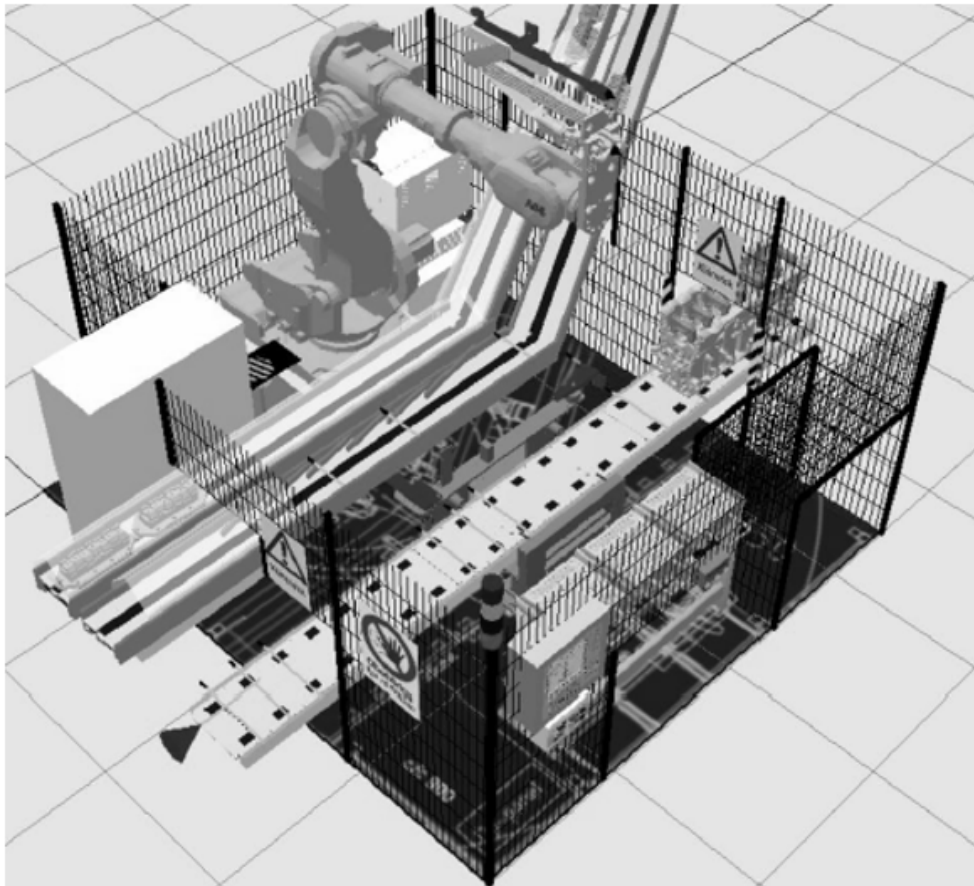


Figura 1.6: Celda con robot industrial en Simumatik3D Universidad de Skövde.[5]

En esta misma línea, en la Universidad Técnica Checa en Praga se desarrolló una celda robótica industrial en la plataforma Tecnomatix de Siemens PLM®, donde se simuló un robot KUKA KR5 para procesos de soldadura [14], inicialmente se desarrolló una validación virtual para posteriormente conectar esa simulación con el controlador físico vía OPC, sin embargo, no fue posible sincronizar los movimientos del robot con el modelo virtual por lo cual no corría en tiempo real. De esto se puede concluir que en cualquiera de los casos anteriores las plataformas de software utilizadas contemplaban únicamente la cinemática del sistema de manufactura automático, validándolo de forma virtual al emular tanto el controlador como el

1. Introducción

sistema, dejando abierta la integración de los sistemas físicos y virtuales de sistemas de manufactura automáticos al encontrar inconvenientes en este proceso.

Desde el año 1999, se ha tratado de definir estrategias de validación virtual que puedan brindar beneficios a la industria de la automatización, aunque en esa época se carecía de recursos informáticos y de hardware, la comunidad científica fue capaz de probar los beneficios de este enfoque, logrando clasificarlos en 4 categorías principalmente [2]:

1. Validación Tradicional: la cual consta de pruebas físicas del sistema de manufactura automático utilizando controladores físicos sin asistencia de entornos virtuales.
2. Hardware-in-the-loop (o Soft Commissioning): la cual emplea un modelo virtual del sistema de manufactura automático para probar controladores físicos.
3. Reality in the loop: la cual emplea un controlador virtual para ver la respuesta del sistema de manufactura automático físico.
4. Validación Virtual (o Software-in-the-loop): la cual propone verificar la respuesta tanto del controlador como el sistema de manufactura automático de forma virtual en una simulación.

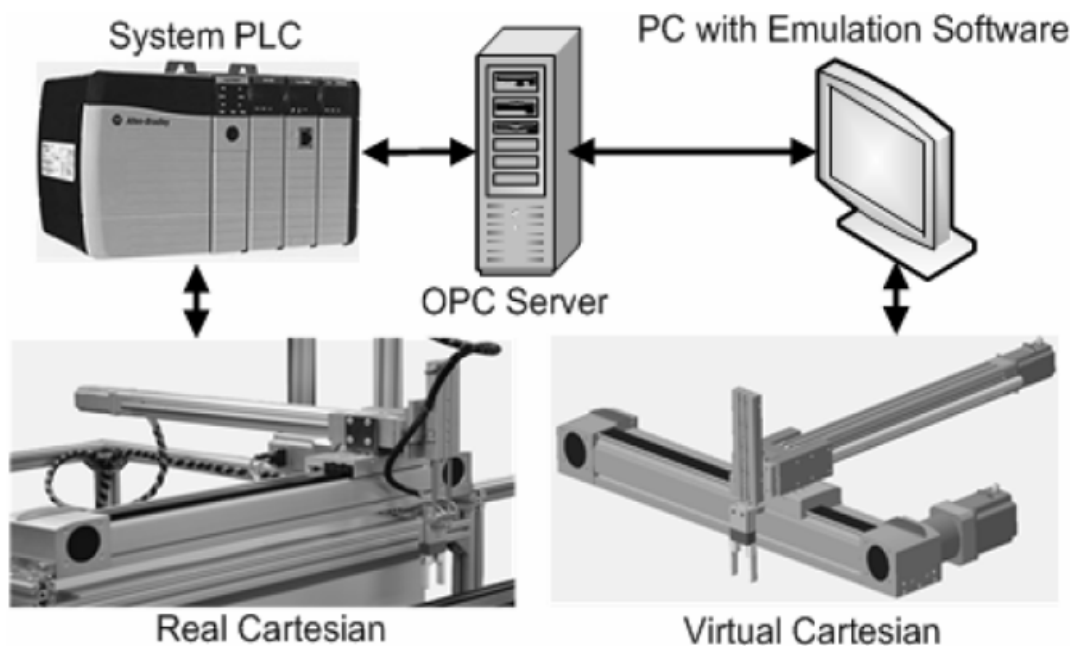


Figura 1.7: Arquitectura de validación del sistema.[5]

Los requerimientos industriales necesarios para intercambiar información en una validación Hardware-in-the-loop [6] se pueden enunciar de la siguiente manera: código de control real (deberá ser posible usar código para PLCs, robots y HMI) y conectividad (deberá permitir la comunicación con una gran variedad de proveedores de PLC, robots y HMI). Para lograr la conectividad entre el sistema de manufactura virtual y el programa en el controlador es necesario el uso de una topología y

arquitectura de comunicaciones como la de un servidor OPC [5], este permitirá la integración con múltiples plataformas como se muestra en la figura 1.7, dando vía a toda la información de señales de entrada y salida las cuales viajarán vía Ethernet estableciendo una comunicación bidireccional.

En la figura 1.8 es posible apreciar los pasos para realizar la validación virtual en un sistema de manufactura automático [4], en él se busca esquematizar los módulos y elementos involucrados comúnmente en un sistema de manufactura como lo son: diseño mecánico, ingeniería eléctrica, ingeniería de automatización, programación de: robots, controladores lógicos programables (PLC) e interfaces humano máquina (HMI).

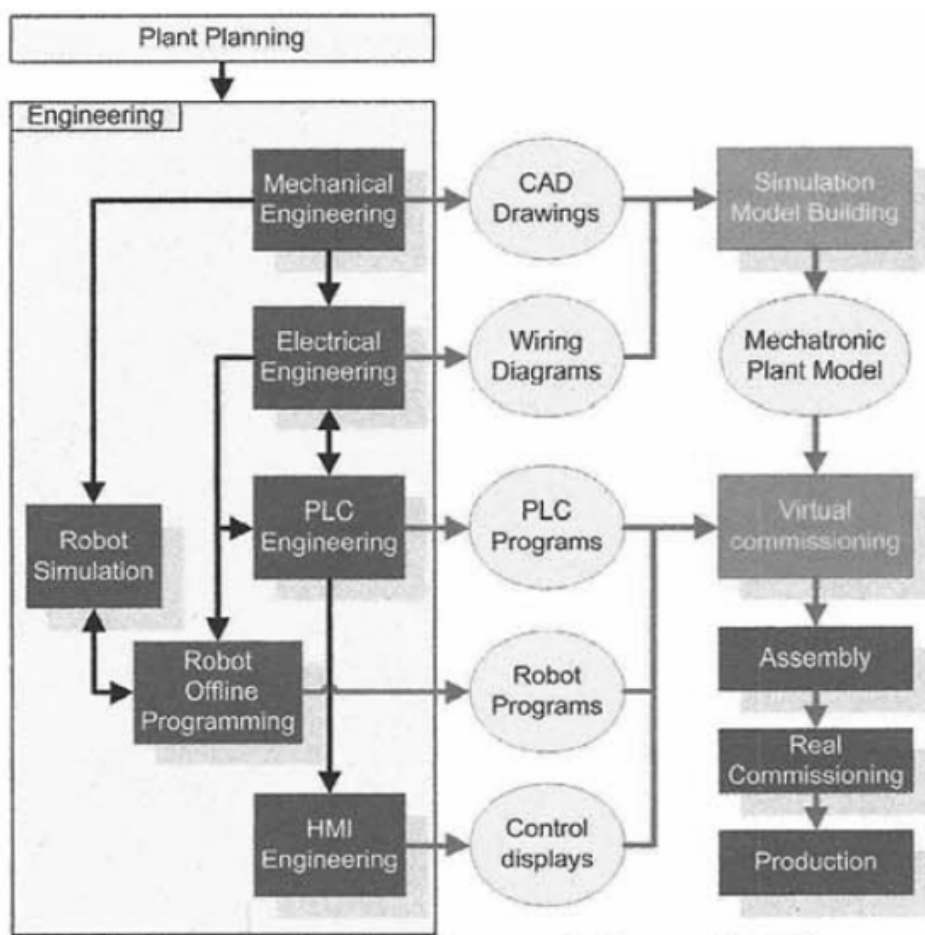


Figura 1.8: Flujo de trabajo para validación virtual.[4]

Para la validación virtual de un sistema de manufactura automático es necesario tener en cuenta aspectos como: el nivel de detalle y funcionalidades de intercambio de información del gemelo digital tanto del controlador como del sistema de manufactura. En cuanto al nivel de detalle este podrá ser dividido en dos grupos principales [4]: bajo nivel de detalle y alto nivel de detalle. Inicialmente se propone utilizar un nivel de detalle bajo, pero con la posibilidad de intercambiar posteriormente el modelo 3D por un modelo detallado. El modelo detallado se debe preparar para la

1. Introducción

validación virtual teniendo en cuenta los siguientes aspectos: 1) La complejidad debe ser reducida considerando el consumo de representación gráfica y 2) Se debe considerar las dependencias entre los objetos móviles y estáticos teniendo como principal enfoque actuadores y sensores.

Finalmente, para validar y verificar las funcionalidades ciberfísicas el sistema de manufactura virtual bajo control discreto (Ver figura 1.9) [12], se han propuesto los siguientes parámetros: 1) animación (que el modelo se comporte gráficamente acorde a lo esperado) donde deberá responder a las funciones de producción esperadas y en el caso de tener una representación real no deberán existir discrepancias entre ambos entornos, 2) validarlo en condiciones extremas (por ejemplo, si la entrada es nula, la salida deberá ser consistente con ello), el controlador deberá enviar y recibir información tanto de los sensores como actuadores, verificando de esta manera el estado de la comunicación de las señales y la lógica de control desarrollada [14].

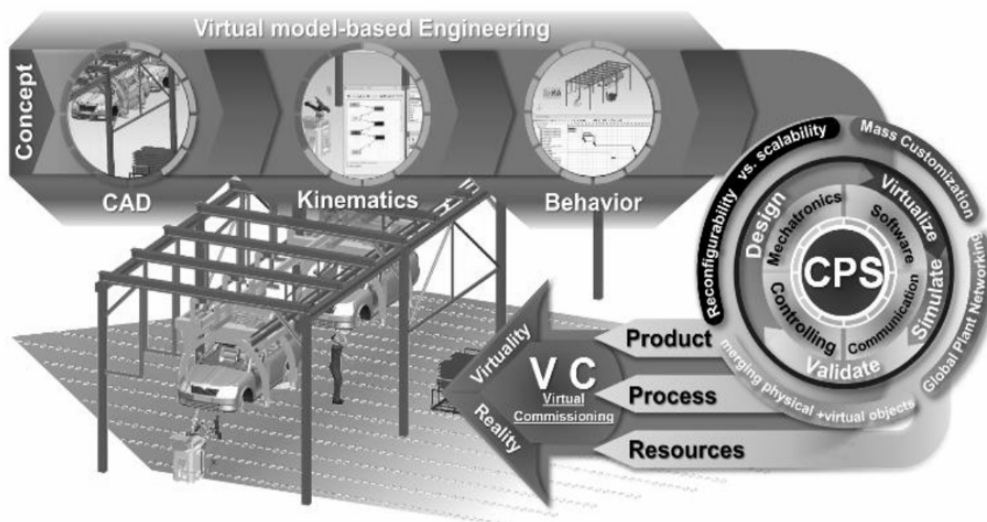


Figura 1.9: Ingeniería basada en modelos con funcionalidades ciberfísicas [11]

Dentro del marco de investigación en la Universidad Nacional de Colombia sede Bogotá, se han realizado avances en el desarrollo de modelos virtuales en la plataforma NX MCD de Siemens-PLM® y su conectividad con sistemas físicos de control discreto PAC Allen Bradley 1769-L30ERM. En la figura 1.10 se observa una máquina etiquetadora la cual tiene como objetivo colocar pegamento sobre la tapa de una caja que se mueven desde la parte frontal a la parte posterior mediante una banda transportadora. En este módulo de manufactura automático existen dos estados importantes, el primero es la posición de inicio para dar referencia a los ejes y el segundo la rutina de movimiento aplicado el pegamento. Se tiene un sensor que detecta la entrada de cajas para activar la secuencia y desplazar los ejes lineales repitiendo esta secuencia hasta tantas cajas entren al sistema. El objetivo final de la máquina es aplicar en línea recta un cordón de pegamento de extremo a extremo del ancho de la caja sin detener la banda transportadora denominado al vuelo.

Es posible evidenciar que la validación del control automático en entornos virtuales con funcionalidades ciberfísicas para mejorar los productos y/o sistemas en condiciones de diseño y operación puede ofrecer las siguientes ventajas:

1. Desarrollo conceptual: pruebas en etapas tempranas con diseños de concepto en un modelo que puede predecir el comportamiento dinámico da una mirada más clara a la toma de decisiones de diseño.
2. Diagnóstico en línea: validar el desempeño del modelo virtual en paralelo con el modelo real puede proveer valiosa información sobre los problemas que pueden surgir con el tiempo de operación y uso.
3. Sensores Virtuales: teniendo en cuenta que el modelo digital responde a las leyes físicas, es posible validar su respuesta ante escenarios incluyendo o suprimiendo sensores para determinar su comportamiento ante una reparación o simplemente una optimización del sistema de control.
4. Mantenimiento preventivo: El impacto sobre la carga dinámica en rodamientos, engranajes, motores, causados por los ciclos de trabajo es uno de los factores difíciles de predecir, con la ayuda de un gemelo digital es posible determinar el impacto en el ciclo de vida del producto, así como las cargas sobre el mismo.
5. Valor agregado al producto: Sistemas de supervisión y monitoreo como SCADA, son llevados cada vez más a nivel de equipos, lo cual dan más información al operario como elemento diferenciador y de innovación. Estos sistemas pueden ser depurados y validados usando el modelo virtual para crear comunicación en tiempo real.
6. Ventas: fuera del departamento de ingeniería los modelos digitales pueden ser usados como herramienta de ventas, al brindar la posibilidad de evaluar escenarios y requerimientos del cliente sin necesidad de consultar al equipo de ingeniería.
7. Seguridad: en procesos donde una parada del sistema es crítico y la incorporación de nuevo personal es necesaria, es posible formar personas no solamente en el uso y operación de los equipos sino también para los desarrolladores y programadores validar nuevas propuestas de control.

Del mismo modo se observa la necesidad de explorar los sistemas ciberfísicos como una metodología novedosa en el ámbito educativo en la Universidad ya que contar con una diversidad y amplia variedad de sistemas de manufactura virtuales nos ayudará a integrar conceptos que por seguridad no había sido posible implementar del todo como lo son la guía GEMMA (modos seguros de arranque, marcha y parada), donde podremos evaluar condiciones extremas del sistema.

Sin embargo, a través del análisis de los desarrollos previos hemos evidenciado la necesidad de explorar una metodología sistemática de validación para llegar a crear modelos virtuales, permitiendo de esta manera a los estudiantes adquirir habilidades avanzadas en el desarrollo de programas de control discreto en PLCs y así mismo a generar acercamientos de la Universidad con empresas, dado que se podrían validar escenarios industriales y poder tomar decisiones más certeras al evaluarlos de manera más extensiva.

1.3. Pregunta de investigación

Desde sus inicios, la idea de validar virtualmente sistemas de manufactura ha buscado resolver una serie de problemas especialmente cuando es necesario integrar equipos y procesos. Sin embargo, hoy en día para lograrlo, se requiere integrar estos sistemas bajo las instrucciones de un control lógico programable (PLC) el cual puede beneficiarse de la interacción con una representación virtual del sistema de manufactura logrando una emulación precisa del comportamiento del sistema.

Esto ha incluido un reto en la industria de la automatización, ya que esta información no está disponible en fases tempranas del diseño. Otro aspecto importante es la complejidad debido a la diversidad de combinaciones que pueden existir entre los elementos virtuales [1] y físicos del sistema, presentado un reto de comunicaciones. Por ello nace la necesidad de aplicar técnicas que ayuden a solventar estos problemas reduciendo significativamente retrasos, costos asociados a la tarea de integración de sistemas y puesta en marcha de proyectos de manufactura y automatización .

De acuerdo con las descripciones anteriores se observa la importancia de establecer una metodología de validación virtual para la automatización que integre controladores lógicos programables y una topología de comunicaciones entre los elementos físicos y virtuales, para validar programas de control lógico en fases tempranas del diseño. Como consecuencia surge la siguiente interrogante:

¿Cuáles son las pautas específicas que permiten la validación y verificación efectiva de un sistema de manufactura virtual orientado a la automatización con control discreto y funcionalidades ciberfísicas?

1.4. Aplicación industrial

Dado el alcance del presente trabajo, se propuso y validó un sistema de manufactura integrado y modular que permitiera evidenciar elementos como procesos, máquinas, sensores y actuadores. El proceso dosifica el llenado de envases partiendo de dos posibles configuraciones para después ser llevadas a una banda transportadora la cual se mantiene a una velocidad constante y no se detiene en ningún momento dentro del estado de producción normal, esto implica que el proceso de llenado se realice al vuelo, para posteriormente ser tapados y empacados en pallets y finalmente ser despachados. Como se observa en la figura 1.11, el sistema está dividido en 3 módulos: estación de almacén, estación de llenado-tapado y estación de despacho. Ver Figura 1.11

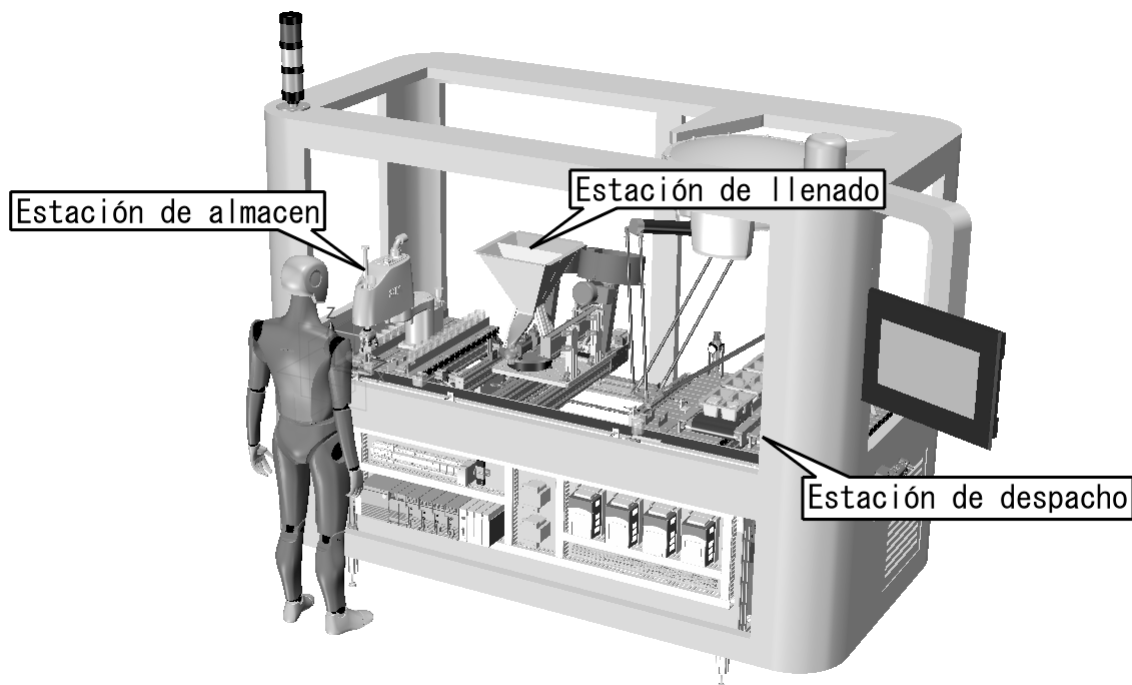


Figura 1.11: Sistema de Manufactura Modular.[Autor]

El proceso comienza con el requerimiento de producción de forma manual o automático en la estación de almacén. La cual está compuesta por dos bandas transportadoras, las cuales ingresan una configuración distinta de envase de producto cada una respectivamente. Para su detención se utilizará un sensor en cada una de ellas que permitan la detección del ingreso del envase en el proceso, una vez alistado el envase acorde al requerimiento de producción este será llevado a la banda transportadora principal por un robot SCARA, realizando una tarea de pick-and-place, el cual tendrá como posiciones determinadas el punto de toma de producto1, producto 2, posición segura de transporte del producto y descargue del producto. Ver Figura 1.12

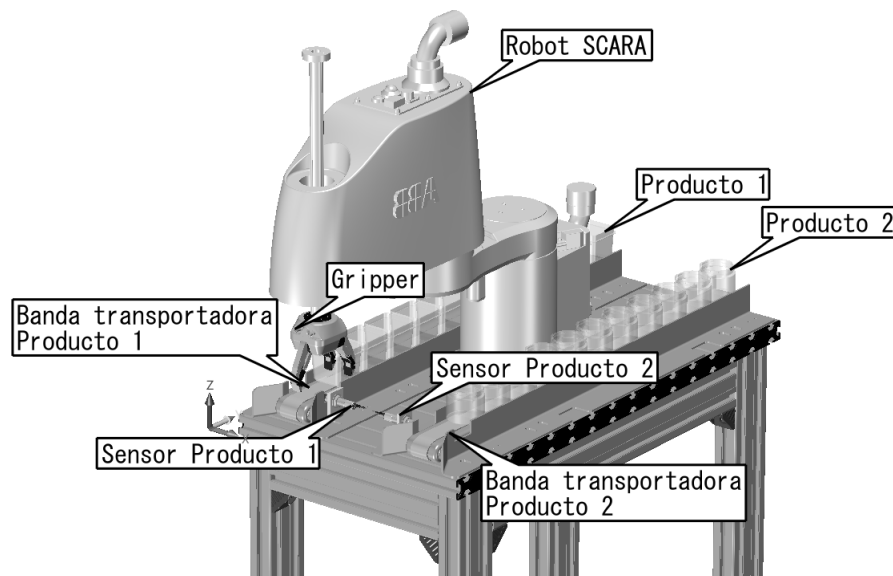


Figura 1.12: Estación de almacén.[Autor]

Una vez el producto es descargado en la banda transportadora principal, el sensor de ingreso de producto lo detectará dando inicio manteniéndola a una velocidad constante de 0.6 cm/s. Esto permitirá que se pueda mantener un ritmo de producción constante y pueda dar la flexibilidad de parametrización del sistema de manufactura. Al mismo tiempo, el sensor de ingreso enviará la señal a la siguiente estación de llenado del producto acorde al requerimiento de producción. En la banda principal también se encuentra el sensor de ingreso de despacho para activar la rutina de empaqueo, la cual envía una señal de alistamiento en posición para el proceso de pick del robot delta. Ver Figura 1.13

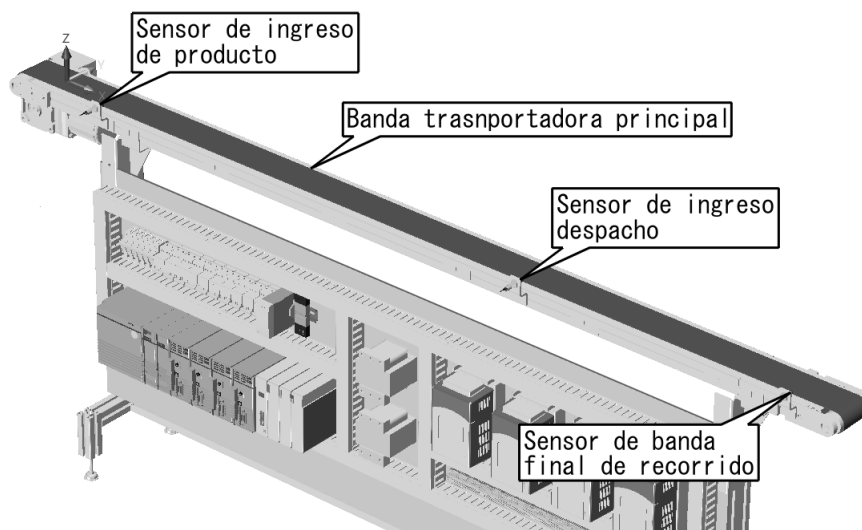


Figura 1.13: Banda transportadora principal.[Autor]

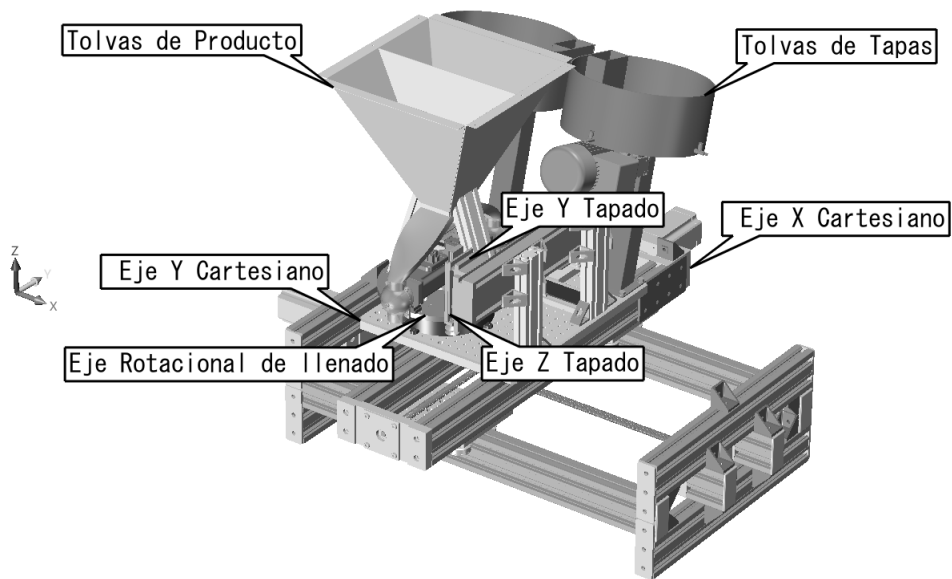


Figura 1.14: Estación de llenado-tapado.[Autor]

En el segundo módulo, la estación de llenado y tapado, está compuesta por un sistema cartesiano XY que desplaza la mesa desde una posición inicial hasta aproximar la banda transportadora principal donde se ejecutará el llenado al vuelo, esta mesa posee dos ejes debido a la necesidad de llenado de tapas y producto en la parte posterior. Adicionalmente este módulo cuenta con un eje rotacional el cual le permite a la tolva girar y dosificar el producto 1 o producto 2 acorde al envase y requerimiento de producción. Finalmente, este módulo cuenta un sistema de tapado con un eje redundante en dirección del eje Y y un eje Z para tapar el producto con la tapa respectiva a cada envase. Ver Figura 1.14

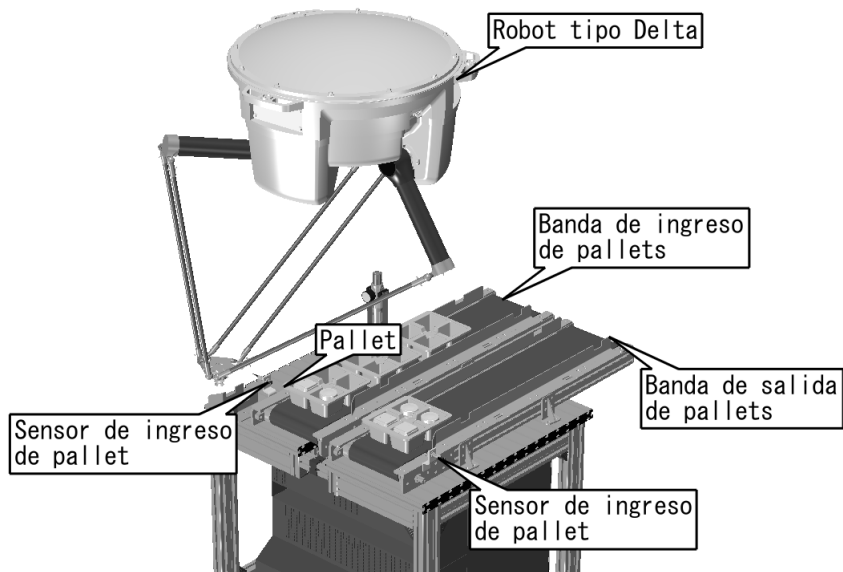


Figura 1.15: Estación de despacho.[Autor]

En el tercer módulo, la estación de despacho, está compuesta por dos bandas transportadoras una de llegada de empaques vacíos y una de salida con empaques lleno, dichas bandas toman la señal de accionamiento de dos sensores que determina el estado del empaque si está o no en posición y así detenerlas. Para llenar el empaque es usado un robot de arquitectura paralela tipo delta el cual toma el producto de la banda transportadora y lo coloca en el empaque hasta alcanzar un total de 4 unidades, posteriormente toma el empaque de la primera banda y lo traslada a la banda de salida para finalizar el despacho del producto. Ver Figura 1.15

Por otro lado, cada estación cuenta con un sistema que requiere control de movimiento coordinado comenzando por la estación de almacén donde se observa un robot SCARA con una cadena cinemática R-R-P, la estación de llenado cuenta con un robot cartesiano con ejes XYZ y cadena cinemática PPR, donde se requiere interpolar los movimientos de los tres ejes en el posicionado para el llenado al vuelo, y finalmente el robot delta que representa una cadena cinemática cerrada donde las tres articulaciones de los actuadores tipo R deben coordinar la llegada de la posición del efector final.

Finalmente, se desglosó la guía GEMMA y se estructuró una propuesta a nivel funcional con evolución a estados seguros del sistema de manufactura atendiendo los estados más comunes en condición de parada y arranque. Logrando desarrollar los programas a nivel de autómatas en el lenguaje SFC y Ladder y posteriormente cargarlos a un controlador virtual Studio 5000 Logix Emulate. Para poder validar la lógica de control fue necesario establecer una comunicación industrial entre el emulador y el modelo virtual del sistema de manufactura utilizando comunicaciones de tipo OPC UA, una arquitectura unificada, la cual precede a la comunicación por OPC DA y está diseñada para satisfacer los requerimientos de la automatización industrial actual.

1.5. Alcances y delimitaciones

En primer instancia, se propone ejemplificar un sistema de manufactura virtual, limitado a 3 módulos, con un máximo de 50 sensores y 20 actuadores. En segundo lugar, se revisarán las metodologías de validación virtual de sistemas de manufactura, para identificar pautas específicas propuestas por los autores en el material bibliográfico recopilado. No se clasificarán o compararán herramientas de software para este efecto, realizando únicamente el modelo virtual en el software adquirido NX MCD de Siemens PLM®. En tercer lugar, se generará el GRAFCET del sistema de automatización en manufactura propuesto teniendo en cuenta las condiciones de evolución entre estados consideradas en GEMMA que para el caso apliquen. Únicamente se programará el GRAFCET a nivel de autómatas para el PAC Allen Bradley usando el software Studio 5000 de Rockwell Automation. Finalmente, la selección de la arquitectura de comunicaciones estará sujeta a los recursos disponibles en las instalaciones del laboratorio de automatización de la Universidad Nacional de Colombia sede Bogotá.

2

Metodologías

En este capítulo se muestra la metodología utilizada para el desarrollo del proyecto, buscando estandarizar y crear unas pautas para el validación virtual como: criterios de modelado en un software de diseño asistido por computador, factores importantes en las comunicaciones industriales de los sistemas ciberfísicos y consideraciones de programación teniendo en cuenta modos seguros de operación.

2.1. Enfoque de investigación

El propósito de esta tesis es definir unas pautas para validar y verificar un sistema de manufactura con elementos como: procesos, máquinas, sensores y actuadores por medio de un gemelo digital y de esta manera validar el control discreto con funcionalidades ciberfísicas. El enfoque propuesto puede ser usado con modelos mecatrónicos de sistemas existentes o en modelos de concepto para asegurar la operación correcta y segura del sistema. Una serie de pasos ver figura 2.1 fueron adoptados para el desarrollo de esta investigación y se enuncian a continuación.

- 1) Definir objetivos: se definió un plan de trabajo, para alcanzar una solución eficiente y el objetivo principal.
- 2) Identificar limitaciones y alcances: la disponibilidad de recursos, limitó a utilizar los softwares: SIEMENS PLM NX, Studio 5000 e IGNITION.
- 3) Desarrollar marco teórico: se realizó la revisión literaria, permitiendo entender oportunidades de investigación y errores a evitar al momento de diseñar el sistema.
- 4) Implementar: se realizó el diseño de la aplicación industrial en el software, se configuró las comunicaciones para intercambio de datos y finalmente se desarrollaron los programas de control discreto aplicando la guía GEMMA.
- 5) Validar y verificar: se evaluó el desempeño y se realizó un análisis de las capacidades del sistema, en esta etapa los resultados se compararon con los objetivos, documentando los problemas no previstos y ajustes realizados.
- 6) Documentación: en esta etapa se documentaron las herramientas de ingeniería, procedimientos y todo lo requerido para la implementación, así como la descripción de los resultados obtenidos y trabajo futuro.

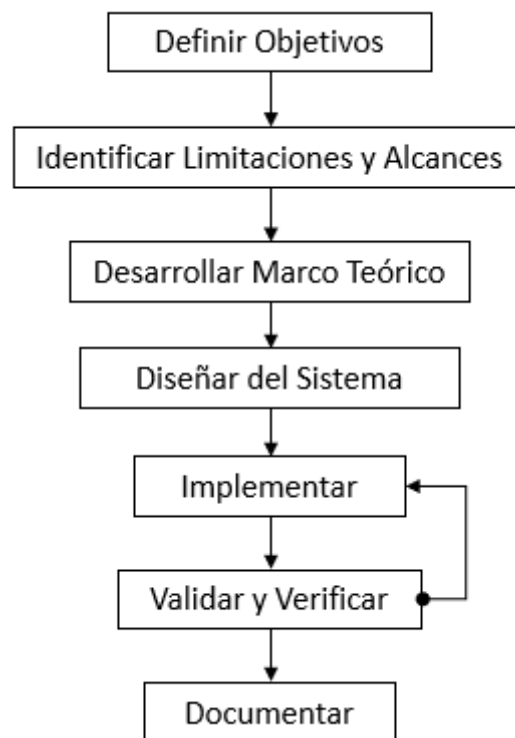


Figura 2.1: Metodología de Investigación.[Autor]

2.2. Infraestructura virtual

Para el diseño y puesta en operación de sistemas de manufactura es importante tener una visión completa desde la fase de ingeniería de concepto, la validación virtual y su implementación [21]. En la figura 2.2 es posible observar tres fases para desarrollar y validar un sistema de manufactura virtual: fase de planificación, validación y verificación del sistema de manufactura virtual y producción.

La fase de planificación parte desde el diseño conceptual donde se plantean soluciones desde diversos campos como: el diseño mecánico, el diseño eléctrico, el diseño de control, diseño de interfaces humano máquina e integración de sistemas robóticos. El sistema de manufactura virtual es conformado por los productos obtenidos como resultado de la primera fase, el modelo del diseño mecánico y el modelo del diseño eléctrico forman el modelo 3D virtual del sistema de manufactura y los productos del área de control e interfaz al ser validados mediante un controlador virtual forman el modelo 2D virtual del sistema de control [12]. Es importante entender que un gemelo digital no solamente significa un modelo tridimensional, si no también puede ser un modelo bidimensional como lo es un emulador de un controlador industrial. En esta sección se abordarán consideraciones importantes a tener en cuenta al momento de validar de forma virtual un sistema de manufactura.

Estos elementos interactuarán por medio de una plataforma de comunicaciones industriales física, logrando enviar y recibir señales debidas a sensores o actuadores

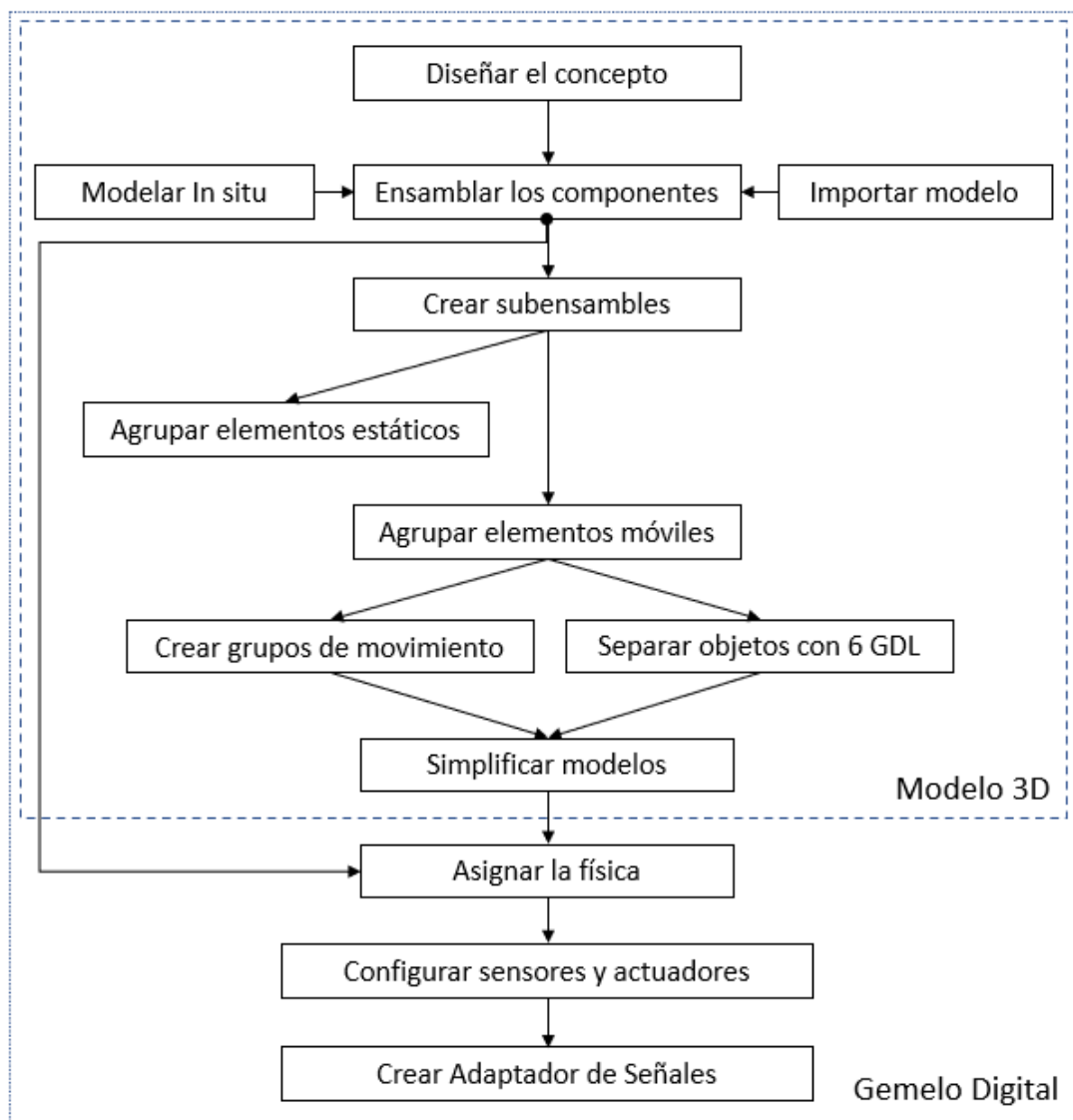


Figura 2.3: Flujo de construcción del modelo digital 3D.[Autor]

Una vez preparado el modelo tridimensional en el software de diseño asistido por computador es posible configurar el gemelo digital agregando la física definiendo si el cuerpo es un elemento rígido o de colisión, por el hecho de que es importante detectar su interacción con otros cuerpos. Para asignar los cuerpos de colisión se recomienda tomar en cuenta las recomendaciones expuestas en la tabla 2.1. Al incrementar la precisión de la geometría y acercarse más a su forma original del cuerpo sólido es más propenso a generar fallas de simulación y reducir el desempeño de la simulación. Es recomendado usar formas simplificadas de colisión en lo mayor posible para reducir el riesgo de inestabilidad en la simulación (atravesándose, pegándose, oscilando) y maximizar el desempeño [19]. También, es importante prestar atención a los modelos que son importados y que pueden tener errores de geometría al momento de ser convertidos, causando inestabilidad al momento de simular. Es importante resaltar que no todos los programas de validación virtual brindan la capacidad de simular la física del sistema, pasando así a la etapa de configuración de actuadores y sensores.

Tabla 2.1: Consideraciones de desempeño en cuerpos de colisión.

<i>Tipo</i>	Precisión Geométrica	Confiabilidad	Desempeño
<i>Box</i>	Baja	Alto	Alto
<i>Sphere</i>	Baja	Alto	Alto
<i>Cylinder</i>	Baja	Alto	Alto
<i>Capnsule</i>	Baja	Alto	Alto
<i>Convex</i>	Medio	Alto	Medio
<i>Multi – Convex</i>	Medio	Alto	Medio
<i>Mesh</i>	Alto	Bajo	Bajo

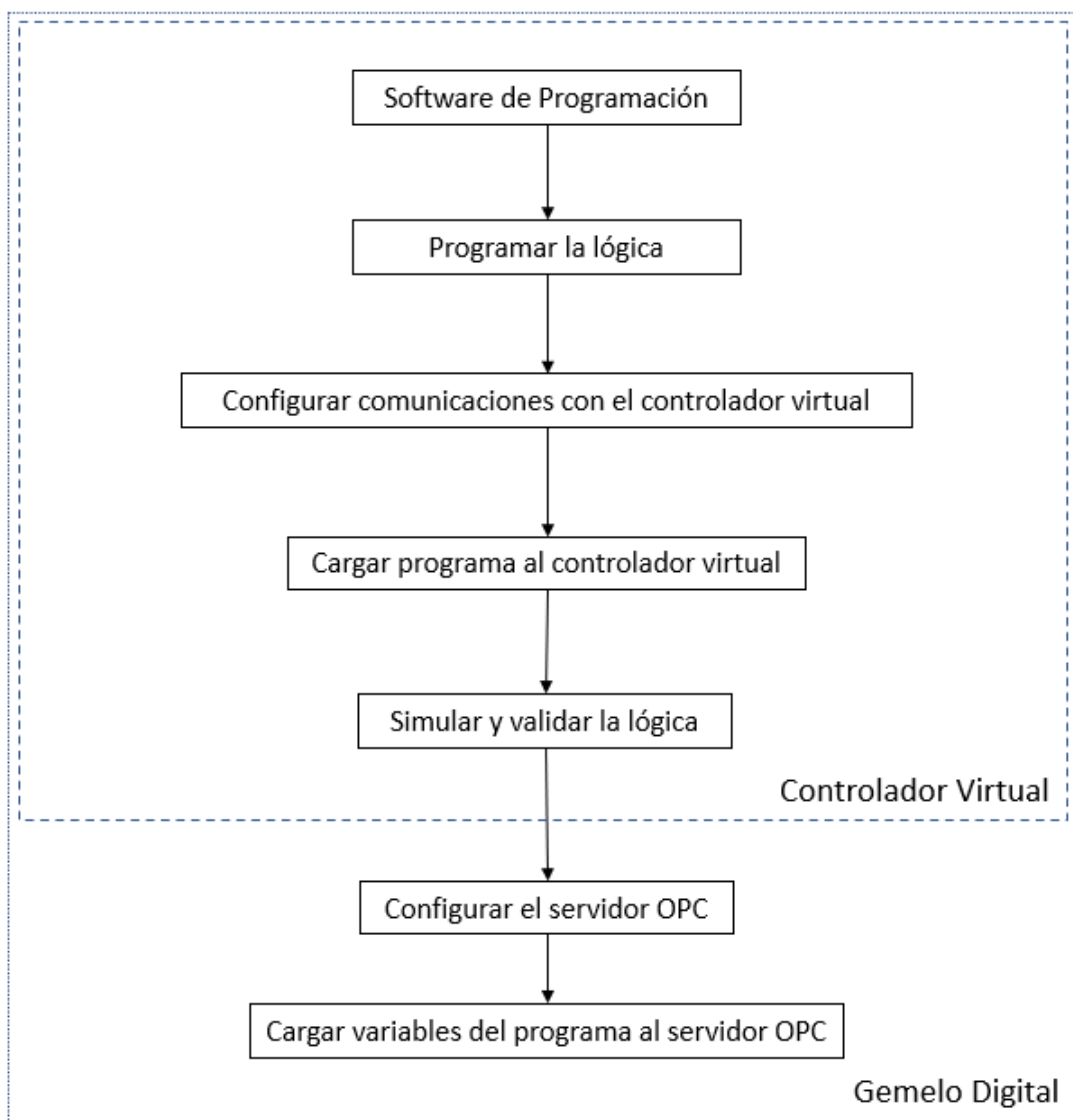


Figura 2.4: Flujo de virtualización de automata programable.[Autor]

Finalmente, se definen y configuran las restricciones de movimiento y señales de cada cuerpo con sus respectivas variables de entrada y salida para poder comunicarlas.

Para validar los programas desarrollados de control discreto y las HMI es necesario utilizar un controlador virtual que emule las funciones del autómatas a utilizar en la fabricación del sistema de manufactura virtual [20]. Al configurar el gemelo digital del controlador es necesario tener en cuenta los siguientes aspectos: el software que se desee utilizar debe brindar la capacidad de crear programas de control, simularlos y contar con un emulador del controlador PLC. En este punto es posible crear y validar un programa de manera individual sin conexión con el exterior. Si el objetivo es llevar las señales de entrada y salida a un equipo físico o virtual es necesario configurar un servidor OPC DA o OPC UA que permita escribir y leer dichas variables, siguiendo el flujo presentado en la figura 2.4 logrando así un gemelo digital de un controlador PLC.

2.3. Comunicaciones industriales

Las comunicaciones industriales son elemento esencial para la interacción de un gemelo digital con su entorno ya sea el modelo tridimensional o el emulador del autómatas programable. Se propone el siguiente esquema donde es posible observar las combinaciones posibles y redundancias que se pueden presentar en el sistema ciberfísico para realizar la validación virtual de un sistema de manufactura en los 4 niveles mencionados en la sección 1.1 [2]. Como se puede observar en la figura 2.5 puede existir comunicación entre el modelo digital 3D y el controlador virtual. También, se pueden comunicar con el exterior a elementos físicos reales, que a su vez se pueden comunicar directamente o por medio de comunicaciones industriales.

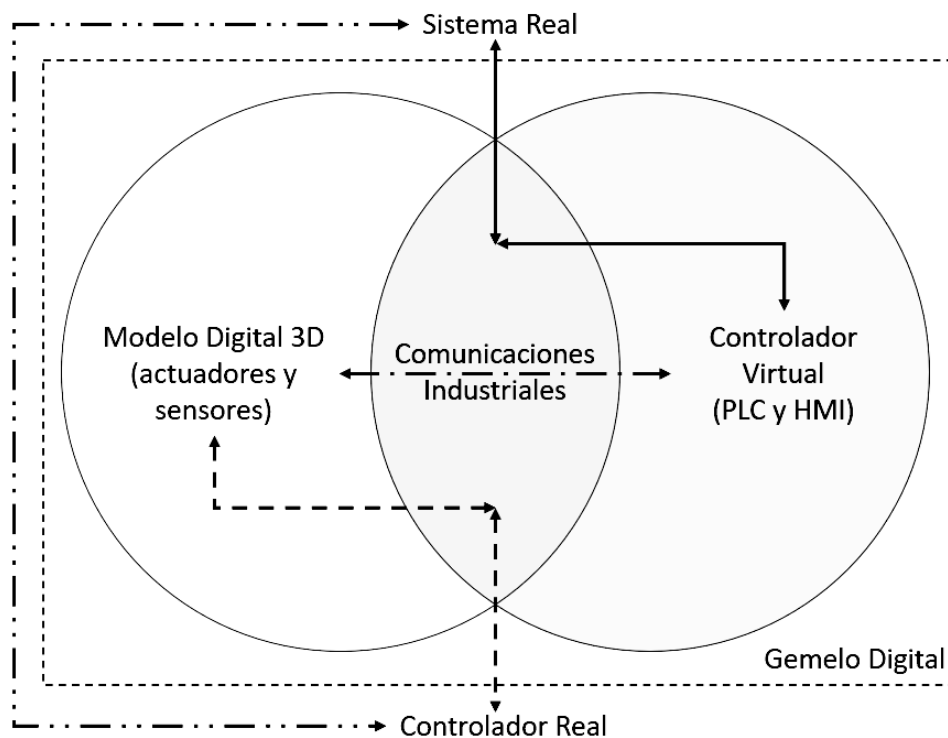


Figura 2.5: Esquema de comunicaciones de un sistema ciberfísico.[Autor]

Existen varios factores importantes a tener en cuenta al momento de cablear los sistemas y comunicarlos: inicialmente se debe revisar los intervalos de muestreo ya que de esto depende la respuesta y eficiencia del sistema. Seguido, no todos los dispositivos se comunican por el mismo protocolo así que debe existir forma o compatibilidad entre ellos para entablar comunicación. Los protocolos industriales más comúnmente utilizados son: el estándar OPC DA que permite el acceso a datos, TCP/IP y OPC UA siendo esta última un estándar pensado para la industria de la automatización. Las comunicaciones por OPC UA permiten un servicio web gracias a su protocolo `opc:tcp://server`, brindando la posibilidad de estructurar sistemas ciberfísicos al conectar múltiples dispositivos en múltiples plataformas en un mismo entorno.

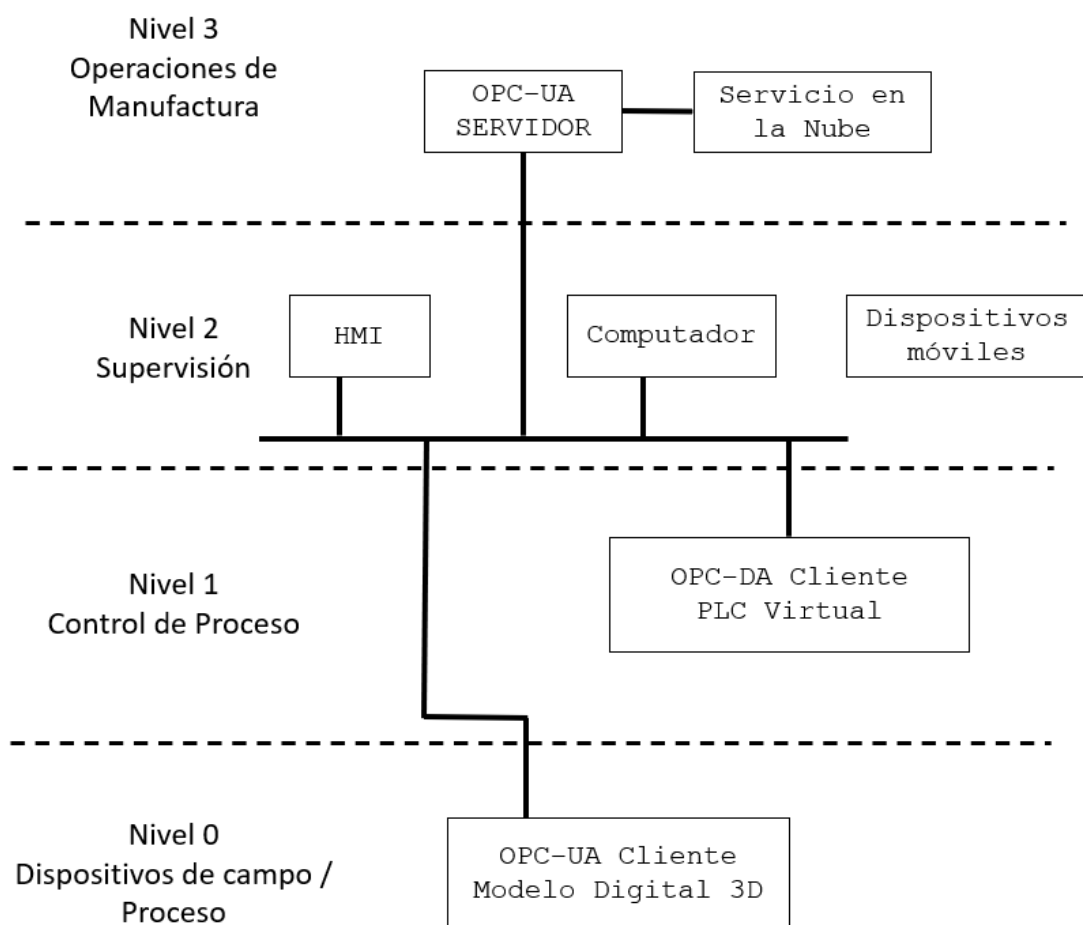


Figura 2.6: Arquitectura de comunicaciones de un sistema ciberfísico.[Autor]

En la figura 2.6 se ejemplifica una arquitectura generalizada para las conexiones y comunicaciones en los diversos niveles de la automatización. De esta manera es posible conectar el nivel de campo digital con los niveles de supervisión y control virtuales, mediante un servidor físico con posibilidades de conectividad en la nube. En el se observa la inclusión de dispositivos móviles para la supervisión y control complementando el sistema ciberfísico en el marco de industria 4.0 e Internet Industrial de las Cosas [16].

2.4. Modos seguros y de operación

En procesos discretos, es muy común tener sistemas secuenciales y el uso de gráficos provee una herramienta útil para la programación de autómatas. El lenguaje gráfico GRAFCET está conformado por etapas enmarcadas dentro de un cuadrado y transiciones entre ellas [17]. Estas últimas son condiciones necesarias para el paso a la siguiente etapa normalmente asociadas a las entradas del PLC y en cada etapa se ejecutan acciones normalmente asociadas a las salidas del PLC. En la figura 2.7 se observan tres tipos de estructuras que rigen la secuencia del programa: a) estructura lineal que busca cumplir el ciclo activando y desactivando etapas en una sucesión lineal, b) estructura selectiva o con direccionamiento donde debe cumplirse una condición para dar paso a las ramificaciones que cumplan la condición y c) estructura de simultaneidad donde varios ciclos funcionan en paralelo cuando se cumple la condición que comparten.

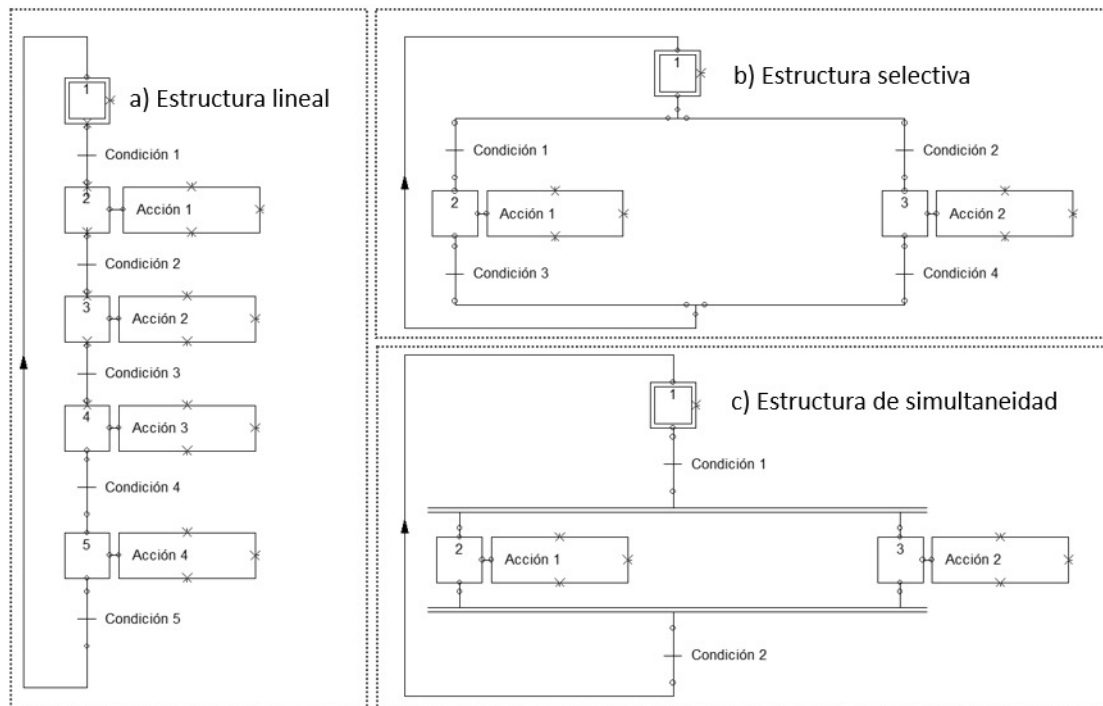


Figura 2.7: Estructuras GRAFCET: a)Lineal b)Selectiva c)Simultaneidad.[Autor]

Para procesos complejos de sistemas de manufactura bajo control discreto aplicando modos seguros y de operación es recomendable coordinar los diagramas GRAFCET utilizando un diagrama principal donde sus etapas permitan activar la secuencia de diagramas secundarios, congelando la secuencia principal [18]. Por otro lado, las transiciones principales de los diagramas secundarios se activan usando las etapas del diagrama principal y para retornar el diagrama principal la ultimas etapas de los diagramas secundarios reactivan la evolución del diagrama principal y así los diagramas secundarios entran en la etapa inicial como se muestra en la figura 2.8.

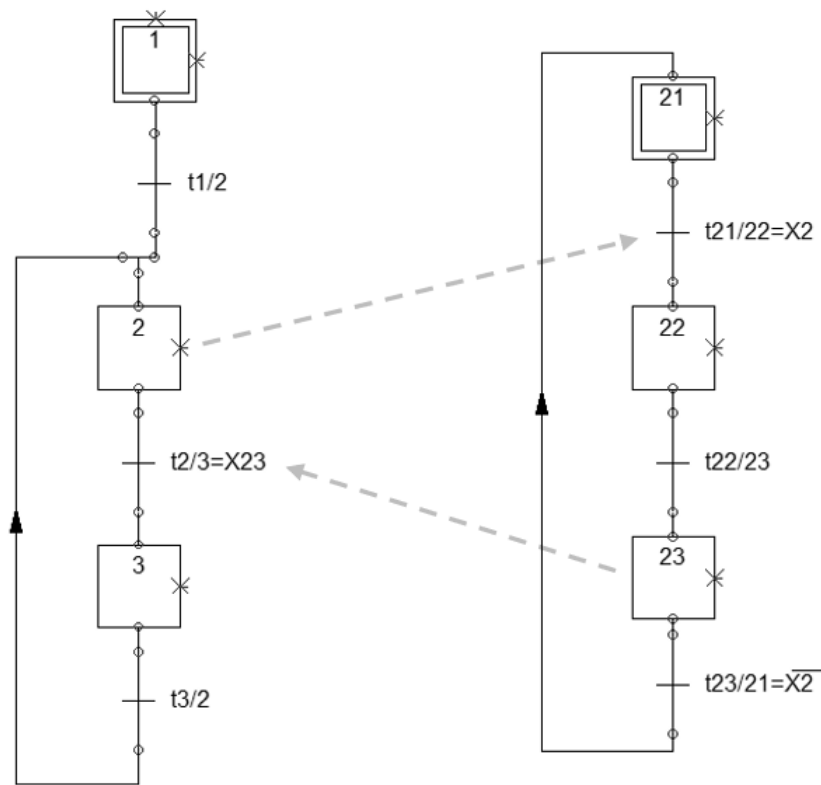


Figura 2.8: Coordinación entre diagrama principal y secundario.[Autor]

Estas interacciones son posibles mediante receptividades en las transiciones, las cuales son entendidas como ecuaciones lógicas a cumplir y pasar a la siguiente etapa. Los tipos de transición pueden clasificarse en: receptividad simple, receptividad compleja, receptividad siempre cierta, receptividad siempre falsa y otras como temporizadas, ejemplos de cada tipo se muestran en la tabla 2.2.

Tabla 2.2: Tipos de transiciones o receptividades.

<i>Receptividad simple</i>	[S10]
<i>Receptividad compleja</i>	[(S10*X2)+B4]
<i>Receptividad cierta</i>	[=1]
<i>Receptividad falsa</i>	[=0]
<i>Otras receptividades (temporizador, ...)</i>	[TIM04]

Cuando realizamos la automatización de un sistema de manufactura normalmente nos centramos en la etapa de producción, es decir siempre queremos que el diseño de nuestro proceso sea lo más productivo posible en el menor tiempo y nos olvidamos de prever algún tipo de situación que perjudique al operario o a las máquinas. Debido a esto, es necesario prever todos los estados posibles como: funcionamiento manual, funcionamiento semiautomático, situaciones de fallo, paradas de emergencia y puestas en marcha, teniendo como prioridad máxima en el proceso de automatización la seguridad, es decir el sistema debe poder evolucionar hacia un estado seguro [10].

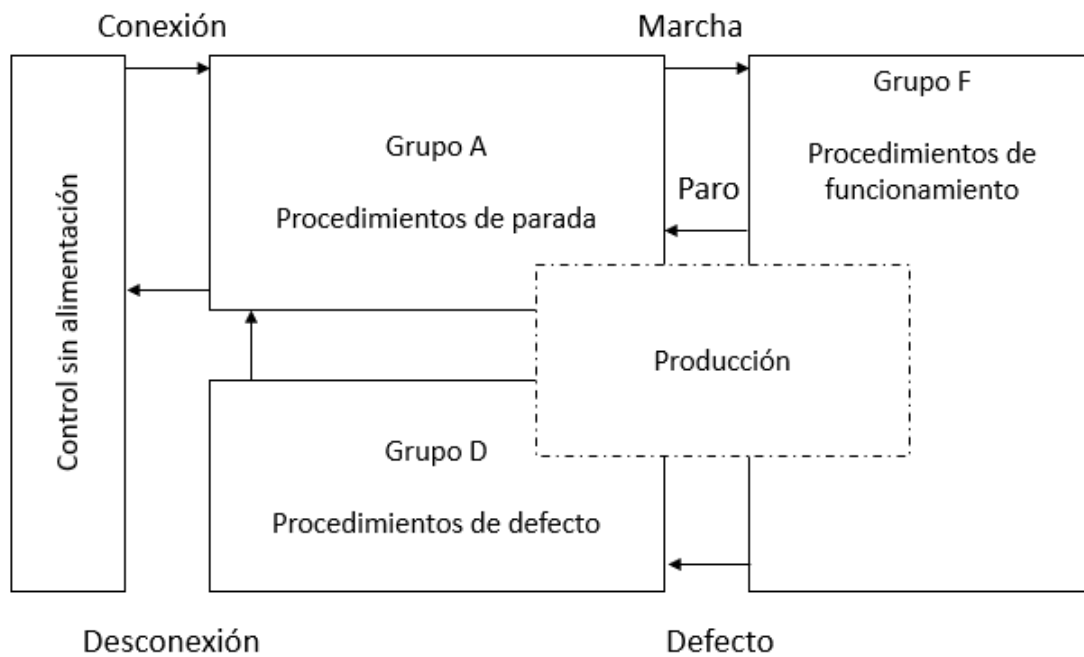


Figura 2.9: Representación gráfica Guía GEMMA Genérica.[Autor]

La guía GEMMA es una guía gráfica que representa el estudio de los modos en marchas y paradas es decir, con ella se realiza el estudio sistemático de todos los modos o estados en que se puede encontrar un proceso de producción automatizado pero también cubre el estudio de salto o transiciones entre modos.

Tabla 2.3: GEMMA Grupo A: Proceso en parada o puesta en marcha.

A1	Parada en el estado inicial
A2	Parada pedida a final de ciclo
A3	Parada pedida en un estado determinado
A4	Parada obtenida
A5	Preparación para la puesta en marcha después del defecto
A6	Puesta del sistema en el estado inicial
A7	Puesta del sistema en un estado determinado

El grupo A corresponde a todas la paradas por causa externa al proceso y se compone de: A1 Parada en estado inicial, siendo este el estado normal de reposo de la máquina, normalmente la etapa inicial del GRAFCET, A2 Parada pedida a final de ciclo, esta parada se produce solo cuando acaba el ciclo actual y pasa a estar parada en el ciclo inicial, A3 Parada pedida en un estado determinado, este puede ser una parada en cualquier momento sin esperar llegar al estado inicial, A4 Parada obtenida, es un estado de reposo de la máquina diferente al estado inicial, A5 Preparación para la puesta en marcha después del defecto, que corresponde a la fase de vaciado, limpieza o puesta en orden que se realiza después de un defecto. A6, Puesta del sistema en el estado Inicial, este estado nos lleva a A1, A7 Puesta del sistema en un estado

determinado, el sistema es llevado hasta la situación concreta diferente al estado inicial, una vez realizado la máquina debe parar.

Tabla 2.4: GEMMA Grupo F: Proceso en funcionamiento.

<i>F1</i>	Producción normal
<i>F2</i>	Marcha en preparación
<i>F3</i>	Marcha de cierre
<i>F4</i>	Marchas de verificación sin orden
<i>F5</i>	Marchas de verificación con orden
<i>F6</i>	Marchas de prueba

El grupo F son los procedimientos necesarios para la obtención de la producción, este se compone de: F1 Producción normal, es el estado en que la máquina produce normalmente, F2 Marcha de preparación, corresponde a la preparación de la máquina para el funcionamiento ya sea precalentamiento, preparación de componentes, etc., F3 Marcha de cierre, corresponde a la fase de vaciado o limpieza, F4 Marchas de verificación sin orden, puede realizar cualquier movimiento o unos movimientos preestablecidos se usa para tareas de mantenimiento y verificación, F5 Marchas de verificación con orden, en esta caso la máquina realiza el ciclo completo de funcionamiento en orden pero al ritmo fijado por el operador, F6 Marchas de prueba, permite realizar las marchas de ajuste y mantenimiento preventivo.

Tabla 2.5: GEMMA Grupo D: Procedimientos de defecto.

<i>D1</i>	Parada de emergencia
<i>D1</i>	Diagnóstico y/o tratamiento de los defectos
<i>D2</i>	Producción a pesar de los defectos

El grupo D denominado procedimiento de fallos o defectos y está compuesto por: D1 Parada de emergencia, contiene todas aquellas acciones necesarias para que la máquina evolucione a una etapa segura, D2 Diagnóstico y/o tratamiento de los defectos, estado que permite el examen de una máquina después de un defecto para determinar los motivos del fallo puede operar con o sin ayuda del operador, D3 Producción a pesar de los defectos, estado correspondiente donde se debe continuar produciendo a pesar de los defectos, por ejemplo, en cadenas de producción puede sustituirse parte del trabajo por un operario hasta reparar dicho defecto.

Finalmente, el diseñador tendrá libertad acorde a las especificaciones del sistema de manufactura de tomar los procedimientos necesarios para garantizar la operación y estados seguros de operación. Utilizando la técnica de coordinación de diagramas, se asigna como GRAFCET principal los modos requeridos de la representación gráfica GEMMA genérica como etapas y sus condiciones de evolución como transiciones acorde a la figura 2.9. Posteriormente, se programan los GRAFCET secundarios teniendo presente las etapas y transiciones requeridas para desarrollar cada modo a usar de la guía GEMMA, como por ejemplo: A1, A2, A6, F1, F5, D1, etc.

2.5. Validación y verificación virtual

Entenderemos en la automatización de un sistema a la validación como el hecho de que el sistema opera y la verificación como la acción de confirmar que opera acorde a lo pronosticado. Una vez se cumplen los objetivos planteados se procede al proceso de aceptación donde se aprueba y se libera para su manufactura e instalación. Este proceso se denomina pruebas FAT por sus siglas en inglés (Factory Acceptance Test) el cual es un proceso sistemático para garantizar que se cumplen las especificaciones de diseño [21], ver figura 2.10 .

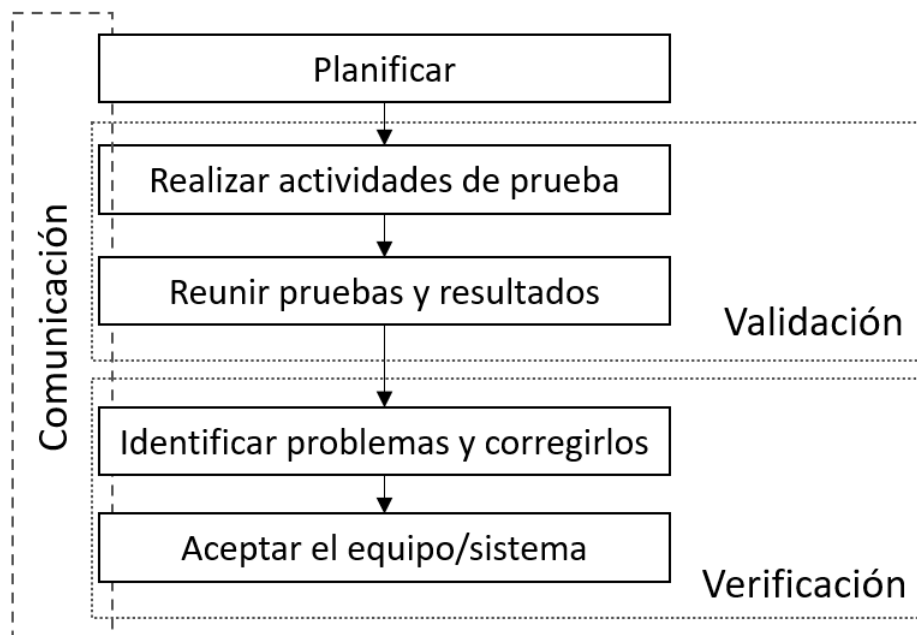


Figura 2.10: Flujo de validación, verificación y aceptación.[Autor]

En la fase de planificación se determinan y documentan las pruebas a llevar a cabo, se identifican los criterios de prueba, protocolos, herramientas necesarias, así como variaciones y niveles de aceptación. En la fase de pruebas se trabaja coordinadamente para llevar a cabo las pruebas acorde a las especificaciones y acuerdos planeados. Los resultados son recopilados y medidos acorde a la planificación y en esta fase se predetermina si los objetivos y criterios se cumplen o no. Una vez identificadas las fallas, errores y alarmas estas deberán ser corregidas y resueltas para ser aceptados en una prueba adicional. Finalmente, cuando se cumplan los objetivos se acepta y libera el diseño para su manufactura y puesta en marcha.

Poder interactuar con un modelo virtual de un sistema de manufactura permite ver el funcionamiento y operación desde la perspectiva de producción utilizando un modelo digital 3D y programas de control discreto a nivel de autómatas programables [20]. Además, nos brinda la posibilidad de identificar en una fase muy temprana inconvenientes y el estado de acoplamiento entre las diferentes áreas del proyecto. Esto da como resultado una validación y verificación virtual en una fase temprana

del trabajo colaborativo de las áreas de diseño, permitiendo retroalimentación en paralelo de todos los interesados en el desarrollo del proyecto de forma ágil.

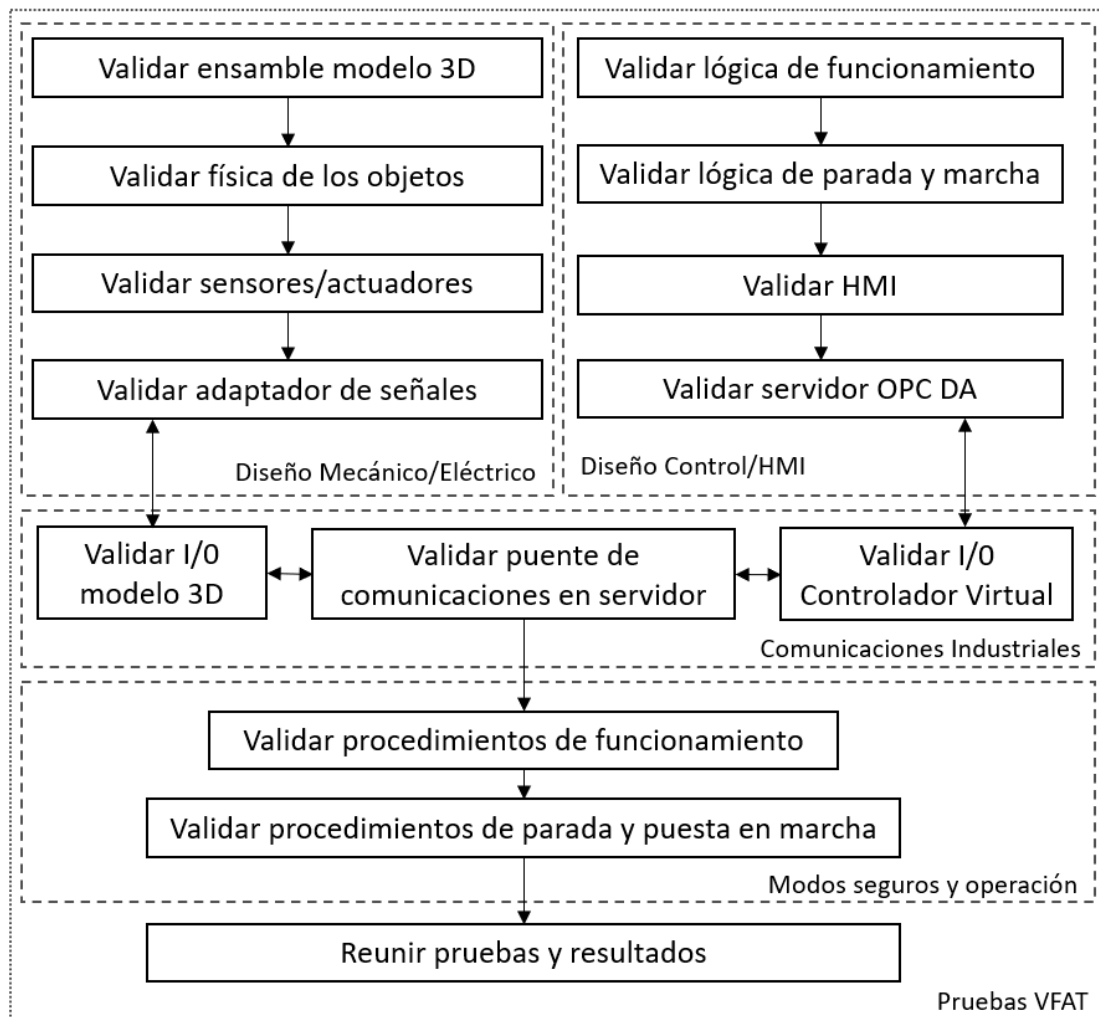


Figura 2.11: Flujo de validación del sistema de manufactura virtual.[Autor]

En la figura 2.11 se observa el flujo para la validación del sistema de manufactura virtual donde el diseño mecánico/eléctrico y diseño de control/HMI se realizan en simultaneo y convergen en las comunicaciones industriales y así, dar paso a las pruebas de funcionamiento, marcha y parada. Dentro de cada una de las etapas se debe buscar la operación acorde a los objetivos planteados como: que el modelo digital 3D tenga los elementos acordados tanto de la parte mecánica como eléctrica, que los actuadores respondan y se comporten acorde a lo esperado, que la lógica del programa envíe y reciba las señales para la producción con los equipos planteados, que exista comunicación bidireccional entre el modelo digital 3D y el controlador virtual, así como que el sistema de manufactura virtual realice la producción y atienda las instrucciones de parada y marcha mediante el uso de una HMI. Si al final de las pruebas se cumple con los criterios y variaciones acordadas se acepta el sistema de manufactura virtual.

3

Implementación

En este capítulo se muestra los procedimientos, herramientas y configuraciones requeridas para: desarrollar el sistema de manufactura virtual, configurar las comunicaciones industriales con el controlador virtual, desarrollar los programas del autó-mata industrial en modo de producción y finalmente la aplicación de la guía GEM-MA para modos seguros.

3.1. Infraestructura virtual de la aplicación

Acorde al enfoque propuesto de infraestructura virtual se realizó y adecuó el ensamble del sistema de manufactura virtual organizando el ensamble principal y los subsensamblés que componen cada módulo del sistema teniendo en cuenta aquellos elementos que tendrán 6 grados de libertad y las cadenas cinemáticas de los sistemas de movimiento. En la figura 3.1 y la tabla 3.1 se puede observar la relación entre el modelo tridimensional y el árbol de ensamble.

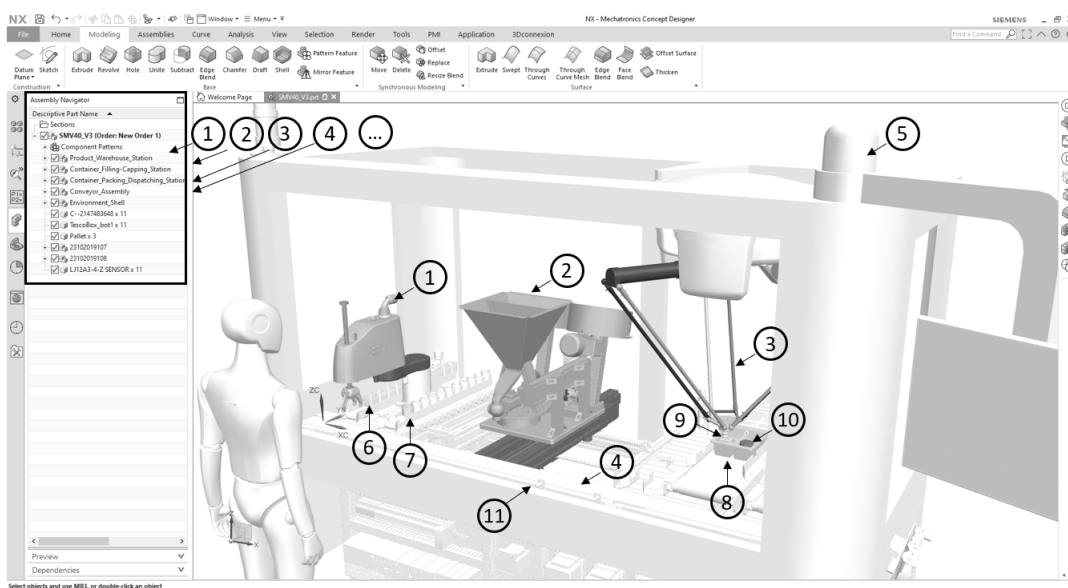


Figura 3.1: Ensamble y subsensamblés del sistema de manufactura virtual.[Autor]

3. Implementación

Tabla 3.1: Árbol de ensamble del sistema de manufactura virtual.

1	Subensamble estación de almacén
2	Subensamble estación de llenado-tapado
3	Subensamble de despacho
4	Subensamble banda transportadora principal
5	Subensamble ecosistema exterior
6	Producto 1
7	Producto 2
8	Pallet
9	Producto tapado 1
10	Producto tapado 2
11	Sensores

Posteriormente se configuró la física del sistema de manufactura virtual mediante el uso del grupo de mecánica del software de diseño. En este grupo podemos encontrar comandos para configurar propiedades de los cuerpos como su interacción con otros cuerpos durante la simulación. Todos los cuerpos que interactúen en la simulación deberán ser definidos como cuerpo rígido y solamente aquellos con propiedades de colisión se les asignará dicha propiedad. Es muy importante poner atención a estos cuerpos porque pueden afectar el desempeño de la simulación ya que requerirán mayor cómputo. En este caso particular los elementos de colisión serán las superficies de transporte, los productos y las estibas. Dichos cuerpos de colisión serán simplificados acorde a las recomendaciones entre precisión y desempeño mostradas en la tabla 2.1.

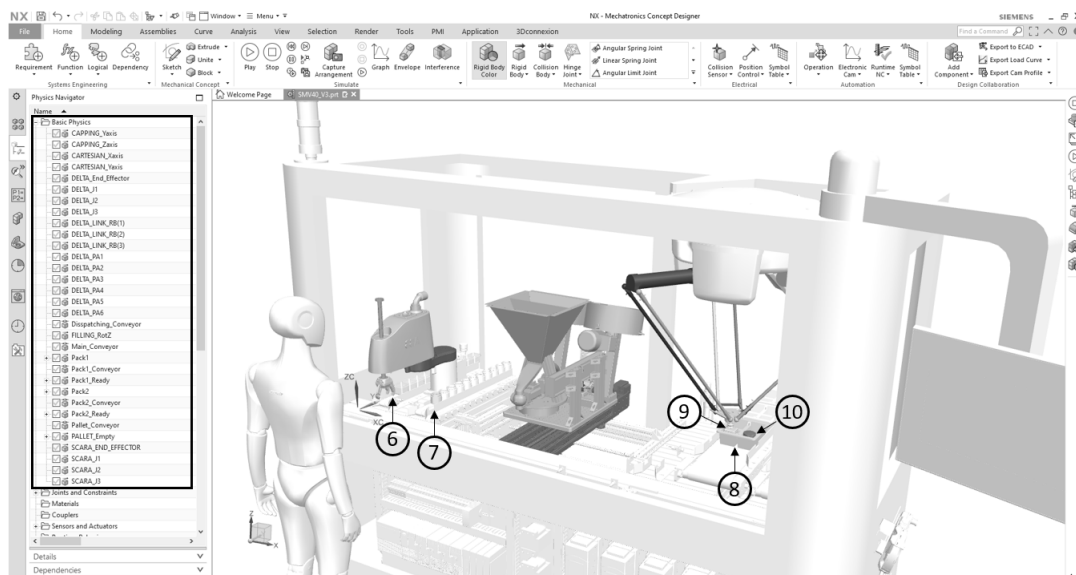


Figura 3.2: Cuerpos rígidos y de colisión del sistema de manufactura virtual. [Autor]

En la figura 3.3 se muestran los parámetros de configuración de la banda transportadora principal donde solamente se asigna como cuerpo de colisión, este procedimiento es recomendado especialmente para este tipo de objetos. Dentro del cuadro

de diálogo se debe asignar el objeto tridimensional como se indica en la imagen, posteriormente el tipo de simplificación geométrica, para este caso se utiliza el tipo BOX el cual no afecta la precisión, ganando desempeño y finalmente se asigna el nombre del atributo.

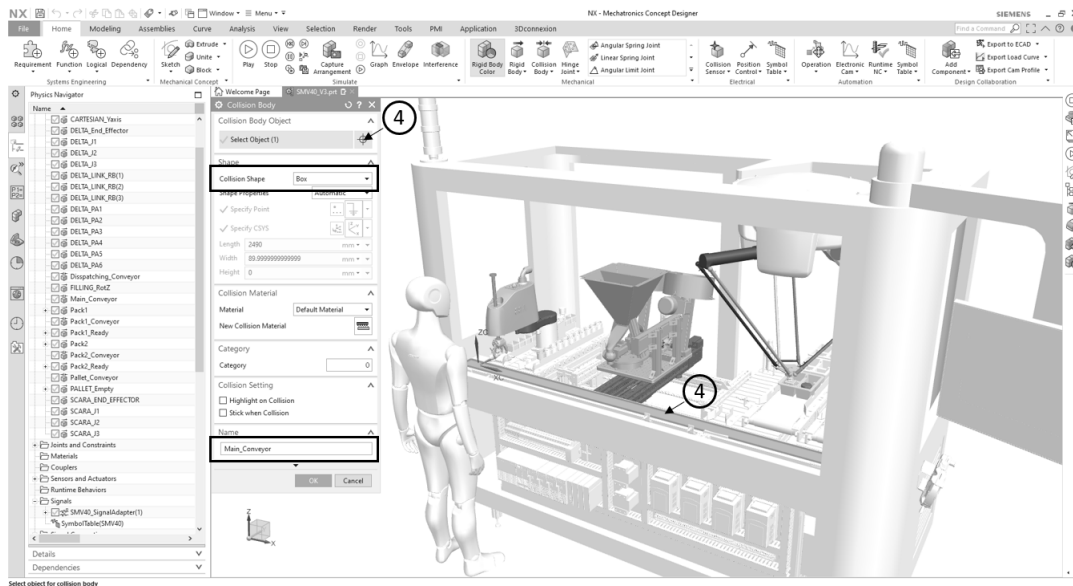


Figura 3.3: Parámetros de configuración cuerpos de colisión.[Autor]

En la figura 3.4 se muestran los parámetros de configuración de uno de los recipientes el cual colisionará y así mismo tendrá 6 grados de libertad por ello es necesario asignarlo como cuerpo rígido de igual manera. Cuando se define un cuerpo rígido es necesario seleccionar el o los objetos que se moverán solidarios, asignar un color es muy importante al simular ya que permite identificar el grupo de objetos del cuerpo rígido y finalmente asignar un nombre referente al cuerpo.

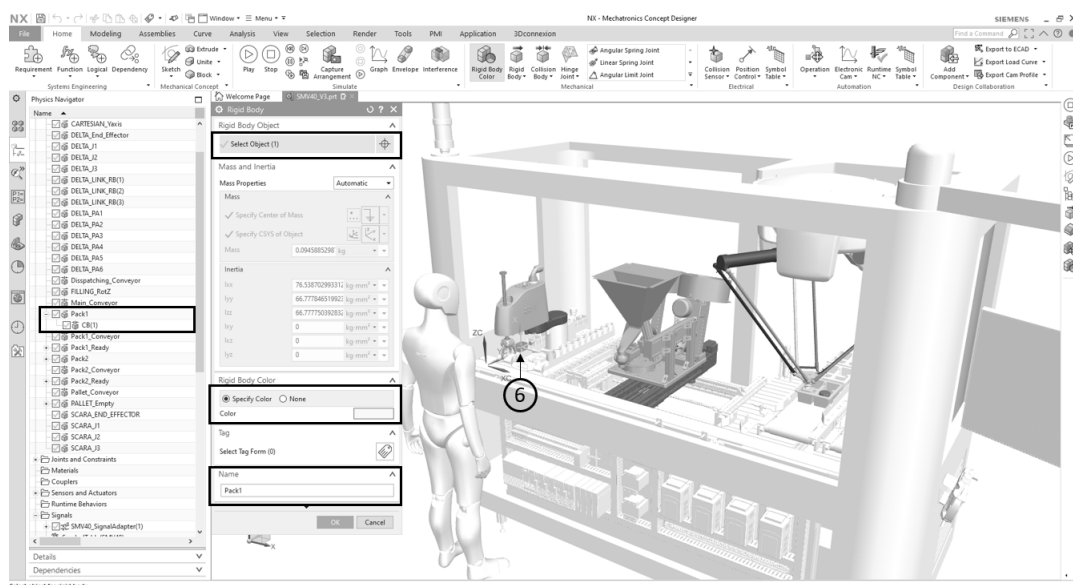


Figura 3.4: Parámetros de configuración de cuerpos rígidos.[Autor]

Para definir las restricciones y el comportamiento de movimiento del sistema de manufactura virtual es importante tener claridad de cada grupo estático y móvil, además de su relación para poder definir las cadenas cinemáticas. En la figura 3.5 se observa la estructura del sistema de manufactura donde se asignó a el grupo estático los elementos pertenecientes a: la estructura, la banda transportadora y los sensores, entendiendo este por no configurable para cuerpo rígido o de colisión. El grupo móvil se clasificó en dos grupos: el primer grupo con los cuerpos que carecen de restricciones como lo son el recipiente 1 y el recipiente 2 que viajarán en el sistema libremente y el segundo grupo con las cadenas cinemáticas de los equipos, una por cada robot.

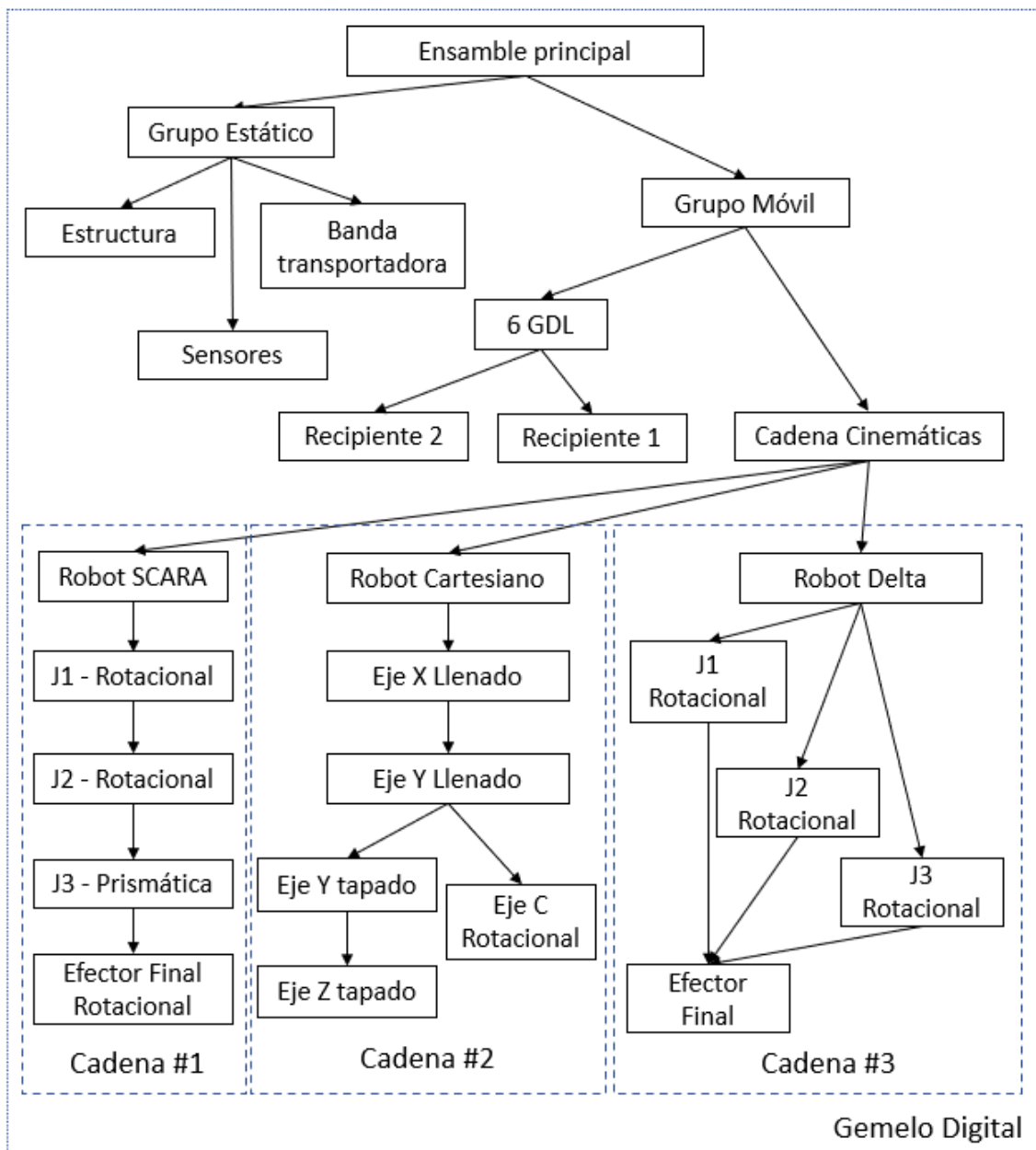


Figura 3.5: Esquema de cuerpos estáticos y cadenas cinemáticas.[Autor]

En la figura 3.6 se puede apreciar en el panel de física en la carpeta de uniones y restricciones las tres cadenas cinemáticas asociadas a cada robot con sus respectivas restricciones articulares de cada sistema robótico. En la estación de despacho es posible evidenciar 4 restricciones acorde al esquema planteado anteriormente con 3 juntas de tipo rotacional y una prismática, adicionalmente es posible percibir una junta de tipo fija la cual es utilizada para realizar tareas de pick and place la cual será descrita en detalle posteriormente. La estación de llenado y empaclado cuenta con 4 juntas prismáticas para los ejes lineales y una rotacional para la tolva que gira al momento de llenar con el producto 1 o con el producto 2, adicionalmente cuenta con una restricción fija para el pick and place de las tapas de los productos. Finalmente, la estación de despacho cuenta con 12 juntas esféricas y 6 rotacionales pasivas para garantizar el movimiento de los paralelogramos, 3 juntas rotacionales para los actuadores y una fija para el efector final para las tareas de pick and place.

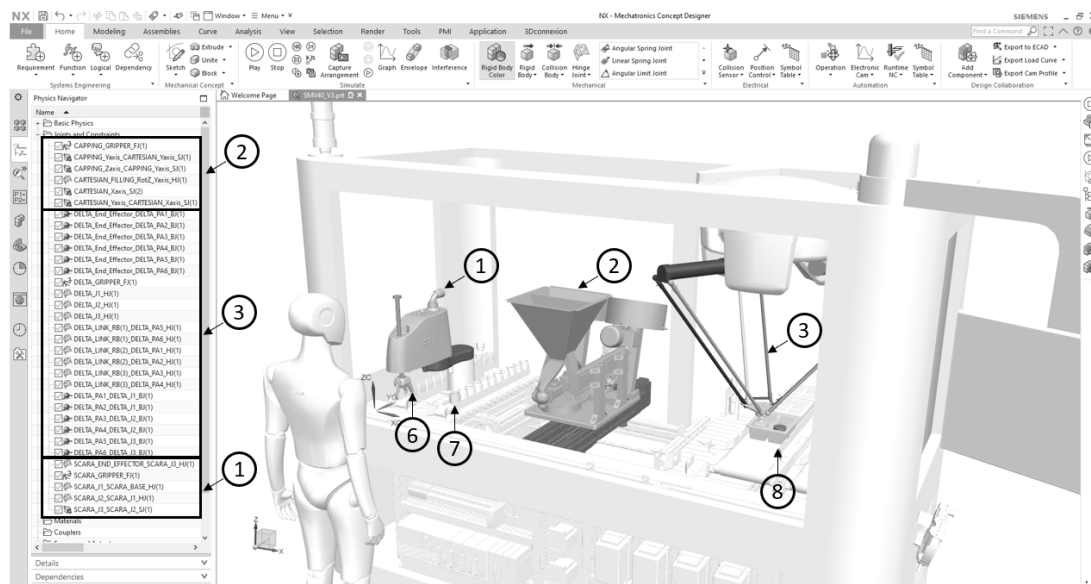


Figura 3.6: Configuración de cuerpos estáticos y cadenas cinemáticas.[Autor]

En la figura 3.7 es posible apreciar los parámetros necesarios de configuración para una junta articular. En este caso se ejemplifica las juntas del robot SCARA J1 y J2, comenzando por la junta J1 la cual es el comienzo del eslabón por lo cual no es necesario declarar la base a la cual esta depende, por el contrario, la junta J2 requiere definir la base J1 y el elemento J2 para que el movimiento de la primera junta afecte la segunda la posición del siguiente objeto. En ambos casos es necesario seleccionar el vector de rotación al ser una junta rotacional y el punto de origen donde se ubica dicha junta, además de asignar un nombre. Las juntas de tipo prismáticas, adicionalmente al objeto y la base, únicamente requieren ingresar el vector de desplazamiento lineal. Las juntas de tipo esférico adicionalmente al objeto y la base, únicamente requieren ingresar el punto de rotación del sistema.

3. Implementación



Figura 3.7: Configuración de uniones en cadenas cinemáticas.[Autor]

En la figura 3.8 se muestra la configuración necesaria para realizar el efecto de pick and place, para ello es necesario crear una junta fija, asignando únicamente la base con el objeto del efector final. Este procedimiento se realizó para los 3 módulos del sistema de manufactura ya que en cada uno se requería realizar una tarea de este tipo. Uno de los errores más comunes al asignar esta junta es seleccionar el objeto diferente a base, causando errores en la simulación, al crear una inestabilidad en la respuesta de la simulación. Siendo este el procedimiento más simple y efectivo de realizar dicha tarea, y que puede ser configurable con una señal externa para ser comandada mediante el uso de un sensor que posteriormente se describirá.

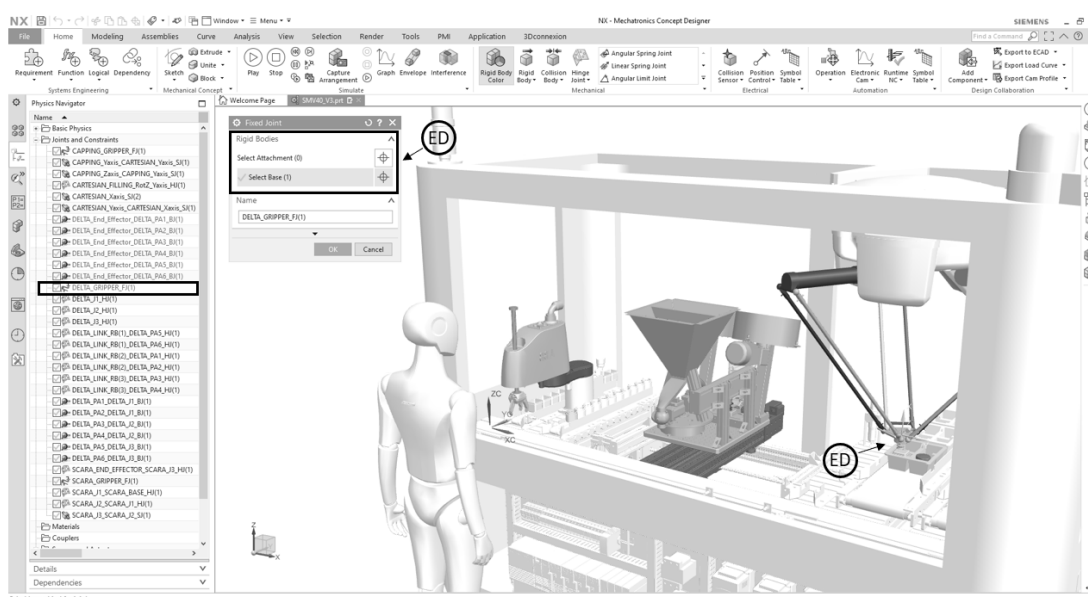


Figura 3.8: Configuración de unión para aplicaciones pick and place.[Autor]

En la figura 3.9 se observan los actuadores y sensores requeridos para la simulación agrupados por módulos incluyendo las bandas transportadoras y los sensores en ellas. El módulo de almacén cuenta con dos sensores uno para cada recipiente respectivamente, el tercero se utilizará para la función de pick and place, adicionalmente cuenta con los actuadores consistentes con las articulaciones descritas anteriormente. El módulo de llenado-tapado cuenta con dos sensores de tapas para cada producto, y cinco actuadores acordes a las articulaciones de movimiento que serán comandadas desde el control. El módulo de despacho cuenta con dos sensores para las bandas trasportadoras, uno para la función de pick and place y uno para reducir la carga de simulación en el pallet de despacho. Finalmente, existen cinco bandas transportadoras en el sistema para mover los productos y pallets.

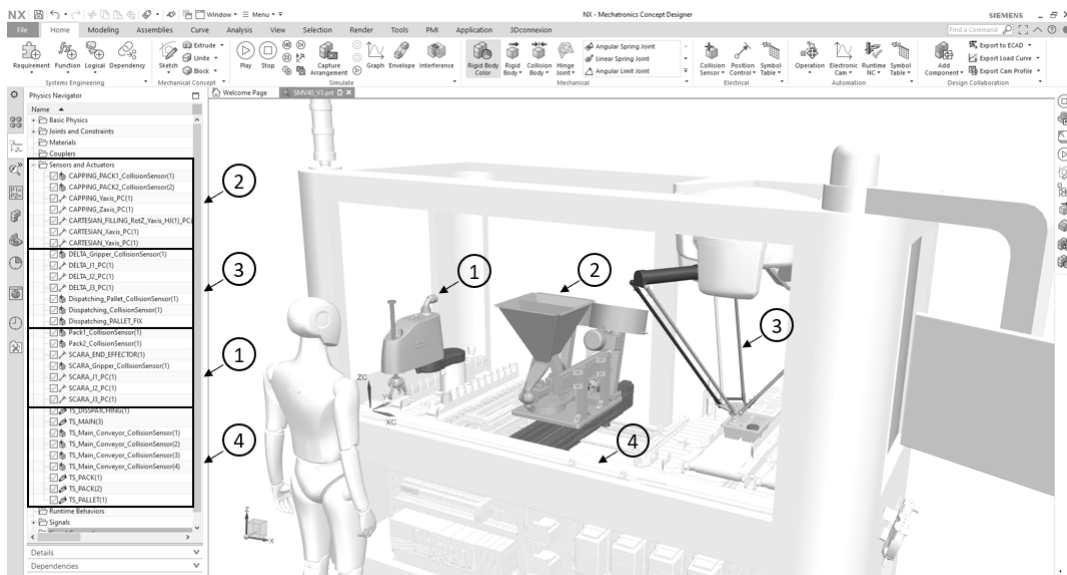


Figura 3.9: Esquema de sensores y actuadores del sistema virtual.[Autor]

Tabla 3.2: Listado de sensores y actuadores.

<i>Componentes</i>	<i>Cantidad</i>
<i>Actuadores</i>	12
<i>Sensores</i>	13
<i>SuperficesdeTrasnporte1</i>	5

En la figura 3.10 se evidencia la configuración de un actuador controlado por posición, el software permite crear diversos tipos de actuadores entre ellos neumáticos, hidráulicos, control de velocidad, etc. En este caso se muestra la configuración del actuador del eje Y redundante para el tapado de los recipientes donde inicialmente se selecciona el objeto como se ilustra en la imagen con la letra M, posteriormente se configuran los parámetros de posición final y velocidad dejando la posición en cero para tomar la posición actual como home e iniciar en esa posición la simulación, la velocidad se dejará en 1000 mm/s dado que es un número suficientemente grande para no afectar la simulación ya que el control es por posición.

3. Implementación

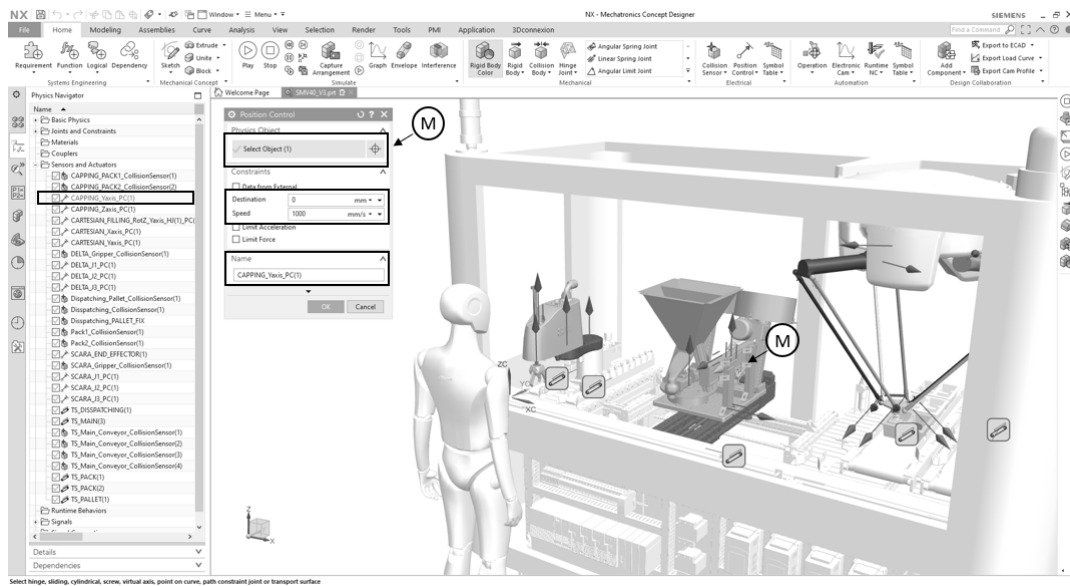


Figura 3.10: Configuración de un actuador del sistema virtual.[Autor]

En la figura 3.11 se muestra la configuración de un sensor de colisión, aunque el software posee diversas opciones de sensores como el de distancia, giroscopio, etc. Dentro de los parámetros de los sensores a configurar se encuentran: el tipo de comportamiento, el objeto al cual se le asigna el sensor, su forma usando figuras simplificadas como línea, caja, esfera, etc. Para este caso utilizaremos de tipo línea con una longitud de 100 mm, definiendo el punto de inicio del sensor y su dirección. Existe otro elemento importante de configuración: la categoría del sensor, esta aplicación es muy útil para darle propiedades de identificación a un grupo específico de objetos como color, material etc. La opción de tipo de detección permite probar el sensor de forma automática o manual, lo cual es muy útil para validar la lógica de programación.

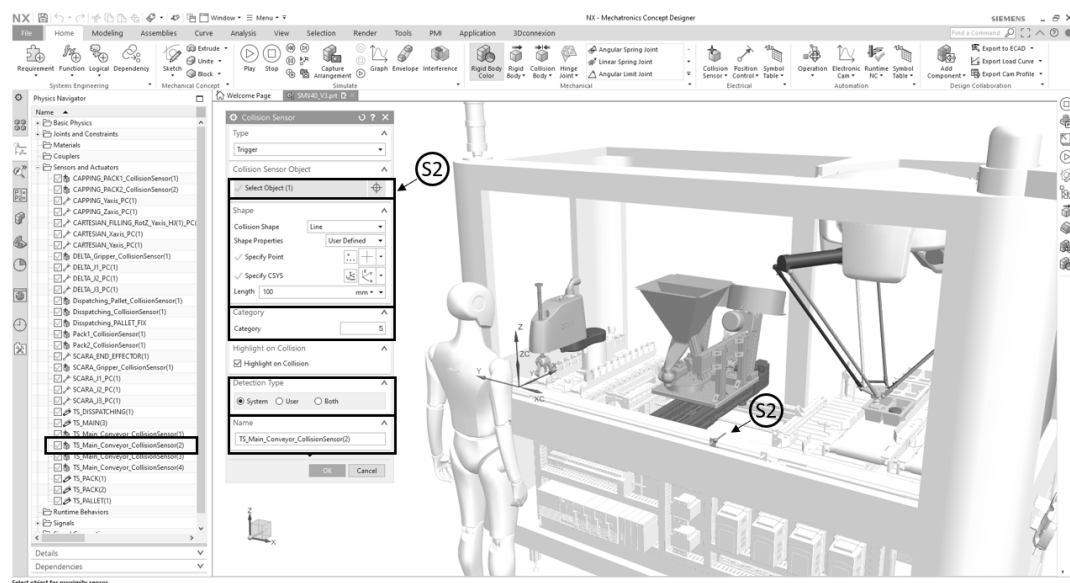


Figura 3.11: Configuración de un sensor del sistema virtual.[Autor]

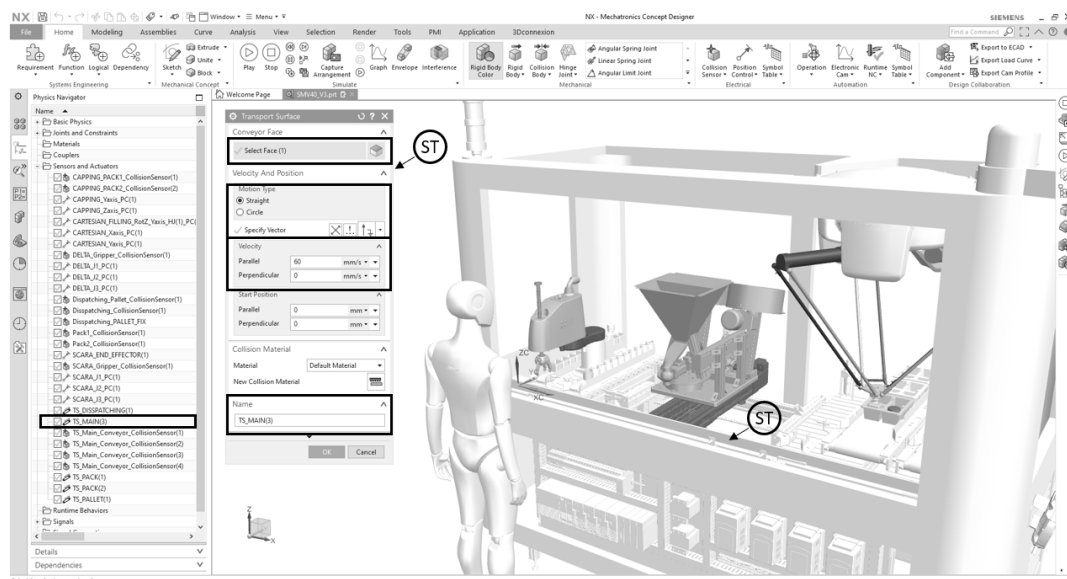


Figura 3.12: Configuración de banda transportadora del sistema virtual.[Autor]

En la figura 3.12 se ejemplifica la configuración de una banda transportadora en este caso la principal. Los parámetros de configuración son la cara con la que harán contacto los objetos sobre ella, su tipo de movimiento recto o circular, el vector de desplazamiento y su velocidad. En este caso se propone una velocidad de 60 mm/s en dirección paralela al vector de movimiento. De esta misma manera se configuraron las bandas para el ingreso de recipientes a una velocidad de 20 mm/s y las de ingreso y salida de las estibas a una velocidad de 45 mm/s.

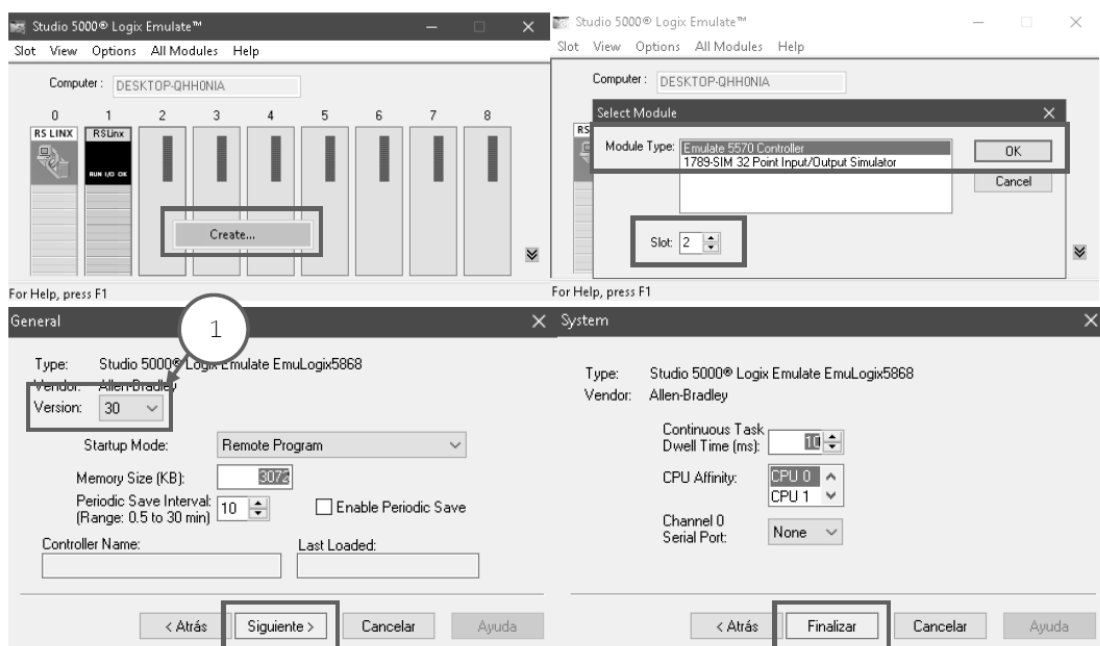


Figura 3.13: Creación y configuración de un autómata industrial virtual.[Autor]

3. Implementación

Otra herramienta útil al momento de validar un sistema de manufactura, es un gemelo digital o emulador del autómatas industrial. Este brinda la posibilidad a los desarrolladores de validar su lógica antes de implementarla en el sistema real. Es importante elegir adecuadamente esta herramienta, ya que algunos comandos podrían no ser emulados limitando la validación virtual.

Para esta aplicación se configuró la herramienta de Allen Bradley Studio 5000 Logix Emulate instalando el módulo a la plataforma de Studio 5000 en el computador en el que se realizó la programación del sistema de manufactura virtual. Una vez desplegado la ventana emergente como se muestra en la figura 3.13, se creó con clic derecho en la ventana emergente, el controlador virtual 5570 del emulador. En la siguiente ventana se asignó la versión del programa en la que se buscaba trabajar y la memoria del controlador, este valor debe ser superior al tamaño del archivo a cargar de lo contrario tendrá inconvenientes al momento de simular.

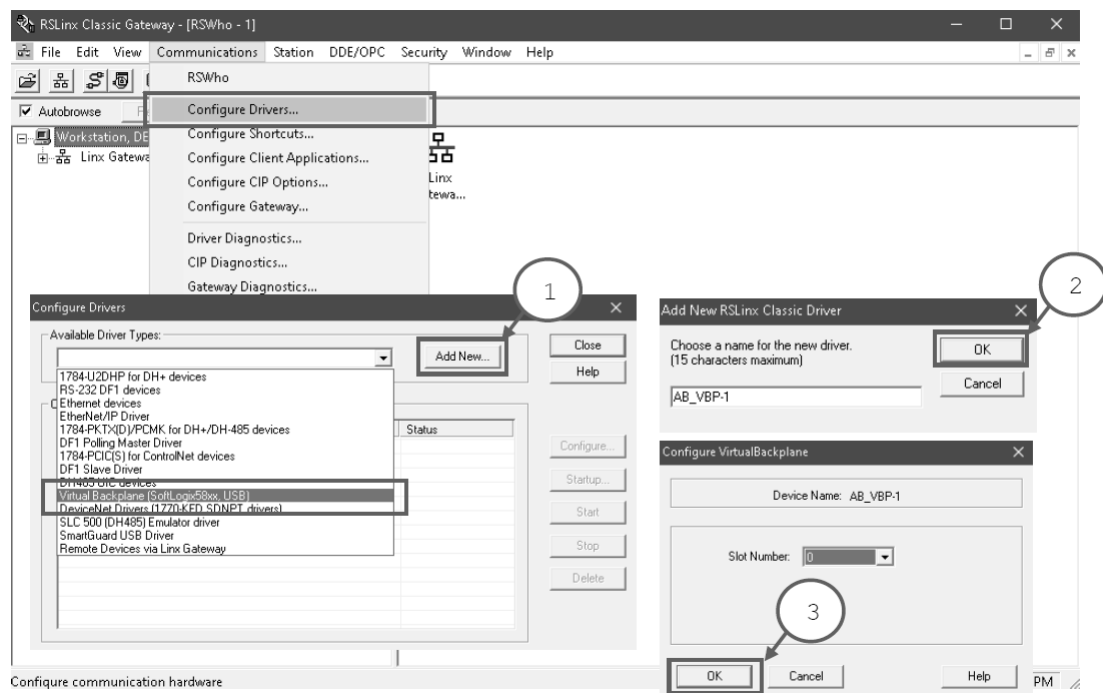


Figura 3.14: Configuración del driver de comunicaciones virtual.[Autor]

Una vez creado el controlador industrial virtual es necesario configurar el driver de comunicaciones por medio del software RSLinx que se instala al momento de agregar el Emulador a la instalación de Studio 5000. Sin embargo, cabe resaltar que inicialmente el software queda en versión RSLinx lite el cual permite cargar programas y validarlos. Si el objetivo es tener un controlador virtual con ese procedimiento es suficiente, pero si la idea es crear un gemelo digital que sea capaz de comunicarse con el exterior es necesario cambiar la licencia para usar la versión RSLinx Classic o superior. Una vez se ingrese al Gateway o puerta de enlace en el menú de comunicaciones, seleccionamos la opción configurar drivers. En la ventana emergente seleccionamos agregar nuevo y aceptamos los valores por defecto ver figura 3.14.

3.2. Arquitectura de conectividad de la aplicación

Un elemento esencial para conectar los gemelos digitales entre si o con el exterior es contar con una plataforma que genere el puente entre dichos sistemas. En esta aplicación se configuró una arquitectura como se muestra en la imagen 3.15 pasando por los diversos niveles que componen un proceso de automatización con la finalidad de darle funcionalidades ciberfísicas y contar con funcionalidades de monitoreo que ofrece un sistema SCADA. La plataforma IGNITION de Inductive Automation nos permitió contar con un servidor web con la posibilidad de escalar el sistema a dispositivos físicos como paneles, dispositivos móviles, estaciones de trabajo, además de una amplia capacidad de conectarnos con sensores y actuadores por los protocolos: OPC-DA y el OPC UA.

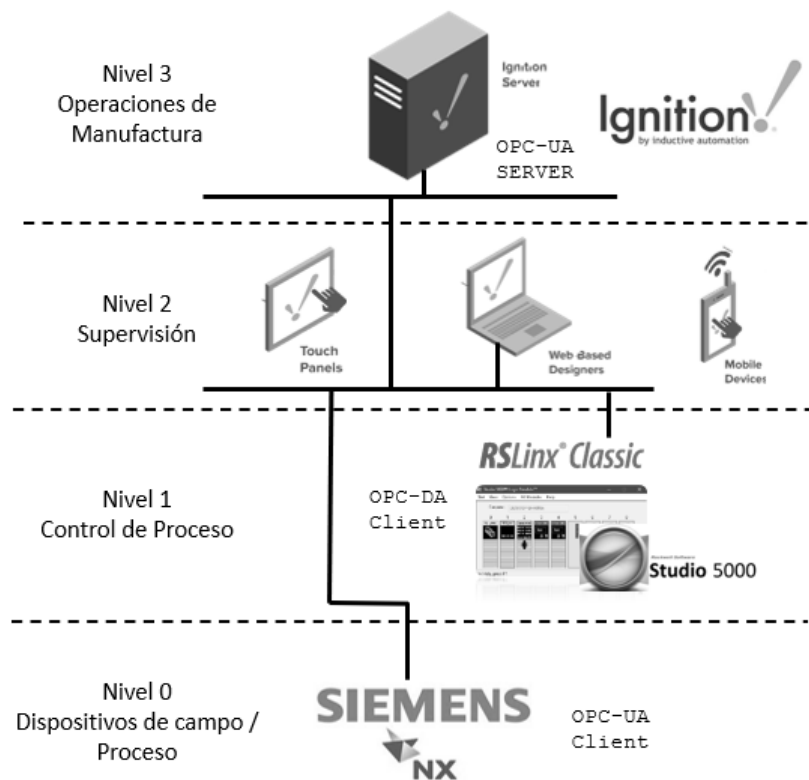


Figura 3.15: Arquitectura de comunicaciones industriales.[Autor]

Como se mencionó anteriormente para poder comunicar con el exterior las variables del programa de Studio 5000, es necesario comunicarse por medio del servidor OPC DA y contar con una licencia de RSLinx Classic o superior. Inicialmente, en la ventana de RSLinx debe crearse un nuevo tópico de comunicaciones, el cual puede configurarse como se muestra en la figura 3.16. Se debe asignar el controlador con el cual se desea realizar las comunicaciones y configurar el archivo del programa del cual se estarán obteniendo los datos, permitiendo así su lectura y escritura. Es importante resaltar que el programa RSLinx debe correr como servicio mediante el uso de la herramienta RSLinx Classic launch control panel.

3. Implementación

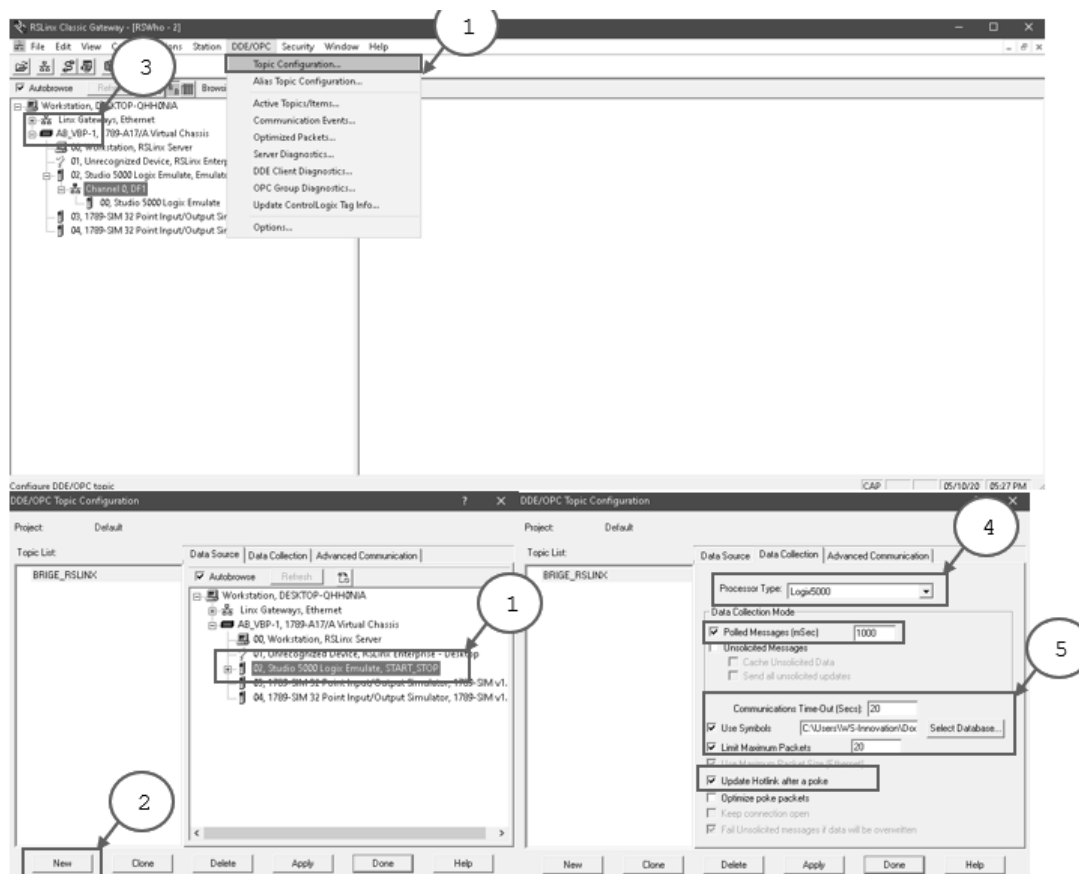


Figura 3.16: Configuración de servidor OPC DA en RSLinx Classic.[Autor]

En la plataforma de Ignition de Inductive Automation una vez instalada se procedió a configurar el cliente OPC de RSLinx para tener acceso a sus datos y administrarlos desde el servidor. Ignition cuenta con un asistente como se muestra en la figura 3.17 para la adición de clientes de tipo OPC DA y OPC UA. Para este caso particular es necesario únicamente asignar un nombre a la conexión, colocar una descripción aceptando los parámetros por defecto y crear la conexión. De esta manera es posible agregar tantos servicios como sea necesario, brindando la posibilidad de integrar múltiples proveedores y equipos en una misma interfaz. Esto facilita las comunicaciones entre el sistema ciberfísico, logrando integrar gemelos digitales y hardware. Una ventaja de concentrar la información en un servidor web, es su practicidad para integrar dispositivos móviles y así monitorear o en su defecto controlar la operación a nivel de campo.

Para dar acceso al servidor de OPC UA de Ignition en NX de Siemens PLM es necesario cambiar los parámetros de seguridad, ya que por el momento no es soportado por esta plataforma. Estos parámetros de seguridad incluyen el concepto de ciberseguridad al encriptar los archivos y proteger la información mediante el uso de usuarios en distintos roles y autenticaciones para el acceso y manipulación de la información.

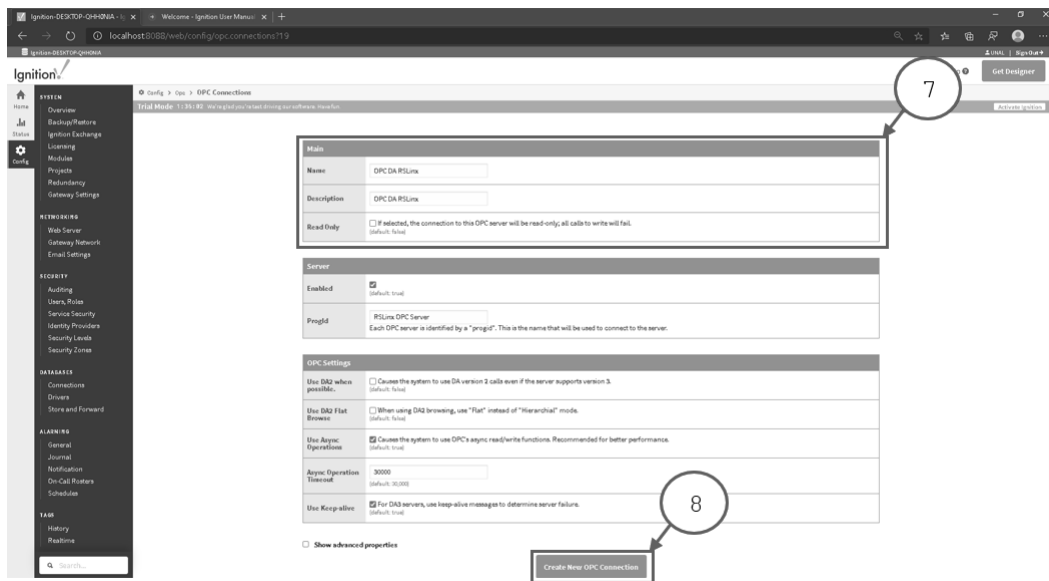


Figura 3.17: Configuración de servidor OPC DA en Ignition.[Autor]

En la figura 3.18 se observan los tres parámetros esenciales para desactivar la seguridad y permitir acceso al exponer las etiquetas de las variables de los servidores cliente que se agreguen al servidor OPC UA. Inicialmente, las políticas de seguridad se asignaron en None y se activaron las opciones de permitir acceso anónimo y exponer proveedores de tags o etiquetas en la sección propiedades avanzadas. Una vez realizados estos cambios se aceptaron y guardaron, dando así la posibilidad de ser vistos por otros servicios sin necesidad de autenticación.

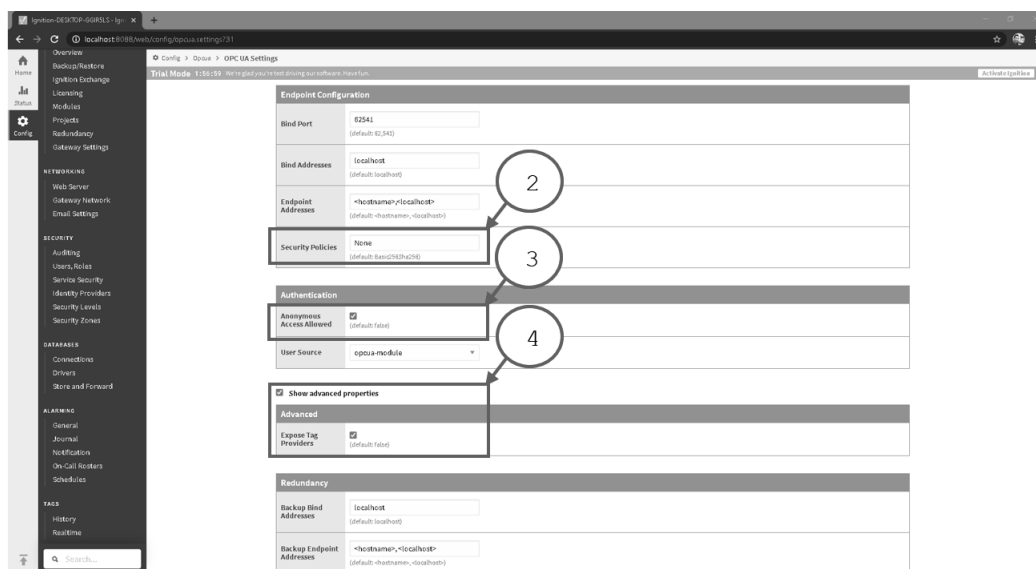


Figura 3.18: Configuración de servidor OPC UA de Ignition.[Autor]

3. Implementación

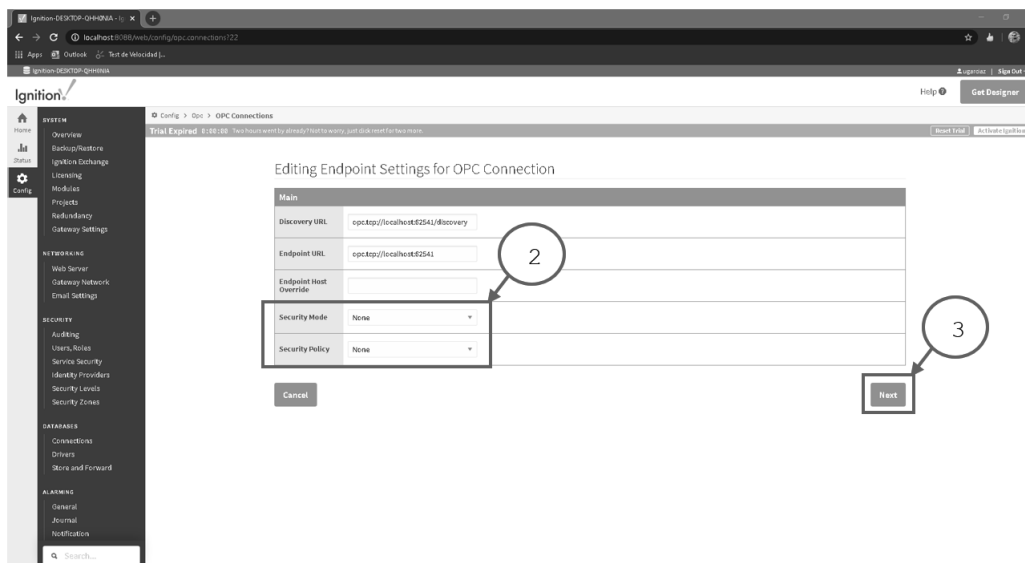


Figura 3.19: Configuración de la seguridad del servidor OPC UA.[Autor]

Posteriormente fue necesario cambiar las propiedades del endpoint del servidor en la opción de configuraciones avanzadas como se muestra en la figura 3.19, donde tanto la opción de modo de seguridad como de políticas de seguridad se asignaron en None. En este punto fue necesario tomar nota de la dirección de URL del servidor pues será la puerta de enlace para la conexión con otros sistemas. En este caso tomamos la opción asignada por default `opc.tcp://localhost:62541/discovery` ya que esta por la función `discovery` permite encontrarla de manera directa. Antes de comenzar es importante realizar el SIGN-OUT de Ignition y posteriormente parar e iniciar el servicio con los archivos en la ruta de instalación.

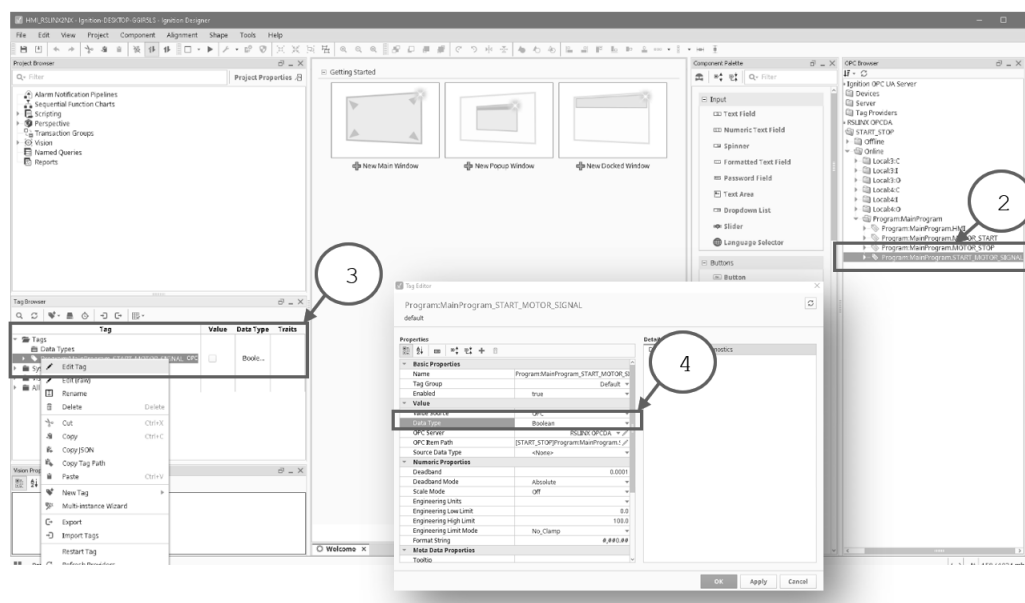


Figura 3.20: Exposición de variables en el diseñador de Ignition.[Autor]

Para exponer las variables en otro servicio es importante hacer uso del diseñador de Ignition y crear un proyecto, el cual nos sirvió para diseñar el panel de operaciones e interfaz con el usuario. El diseñador es una herramienta que se instaló adicionalmente y permite crear paneles mediante una interfaz gráfica. Una vez en el proyecto arrastramos las variables que queremos exponer a la carpeta Tags como se muestra en la figura 3.20, independientemente de que no sean usados en el HMI genera una copia en el servidor habilitándolo como variable expuesta. En este punto es importante verificar que el tipo de dato corresponda con los requerimientos del cliente a conectar ya que esto puede causar errores en la comunicación.

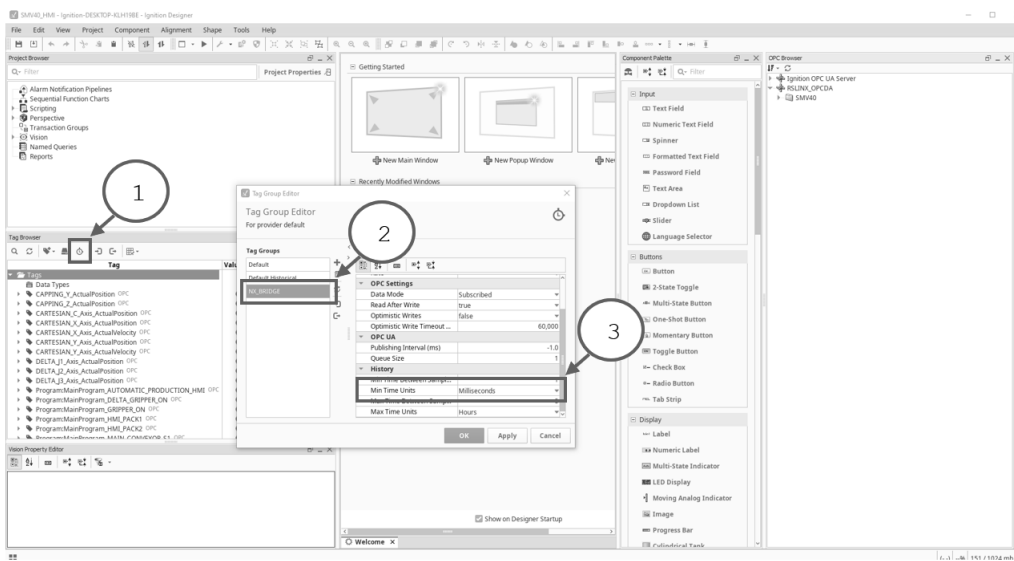


Figura 3.21: Tiempo de muestreo de variables de Ignition.[Autor]

Un factor relevante en comunicaciones es el tiempo de muestreo, especialmente para esas variables de mayor tasa de cambio. En este caso como se buscaba controlar los motores por posición fue necesario ajustar estos parámetros. En la figura 3.21 se observa los pasos para crear un grupo y asignarle una tasa de muestreo mínima en milisegundos siendo esta la tasa mínima posible en el software NX Siemens PLM. Posteriormente a esto a cada variable que requirió una tasa de muestreo como esta, se le asignó este grupo en sus propiedades.

Una vez listas las señales en el servidor de OPC UA, en el gemelo digital del sistema de manufactura se configuró el adaptador de señales de entradas y salidas del sistema. En la figura 3.22 se muestra el procedimiento de configuración de accionamiento de la banda transportadora de ingreso de estibas. Como primer paso se debe seleccionar en la ventana gráfica el elemento a controlar y posteriormente su parámetro, dependiendo del objeto podemos crear señales desde posición, estados, par, aceleración, y aquellas propiedades que posea el objeto. Finalmente, una vez seleccionado el parámetro es necesario crear una señal, a la cual se le asignará mediante una fórmula el parámetro que controla.

3. Implementación

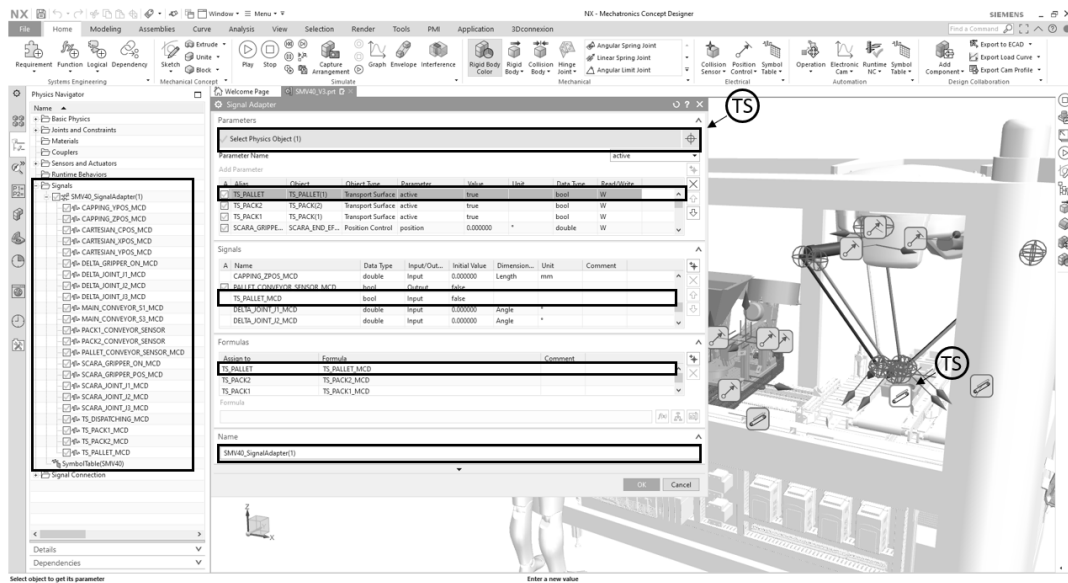


Figura 3.22: Configuración del adaptador de señales en NX MCD.[Autor]

En la tabla 3.4 y en la tabla 3.3 se enuncian las variables configuradas en el adaptador de señales que posteriormente fueron utilizadas para desarrollar el programa en el autómatas industrial. Estas variables pueden ser creadas en cualquier momento según la dinámica de desarrollo y validación, permitiendo flexibilidad cuando se está en fase de diseño.

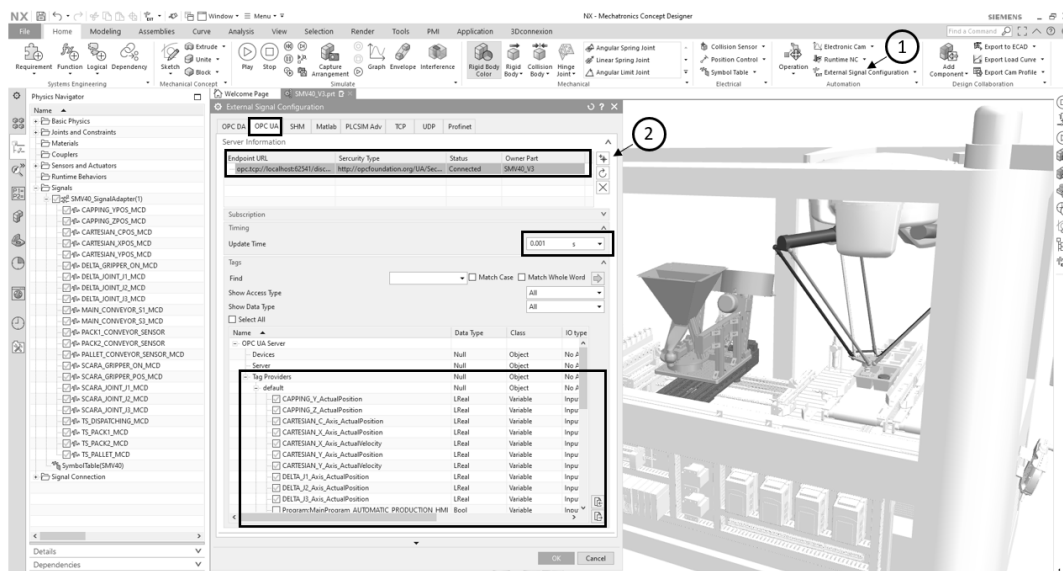
Tabla 3.3: Listado de señales de salida del sistema de manufactura.

Señal PLC	Señal NX MCD
<i>TSPalletSignal</i>	TS Pallet MCD
<i>TSPack1Singal</i>	TS Pack1 MCD
<i>TSPack2Singal</i>	TS Pack2 Singal MCD
<i>TSDispatchingSignal</i>	TS Dispatching Signal MCD
<i>SCARAJ1AxisActualPosition</i>	SCARA Joint J1 MCD
<i>SCARAJ2AxisActualPosition</i>	SCARA Joint J2 MCD
<i>SCARAJ3AxisActualPosition</i>	SCARA Joint J3 MCD
<i>SCARAGripperPOS</i>	SCARA Gripper POS MCD
<i>SCARAGripperOn</i>	SCARA Gripper ON MCD
<i>DELTAJ1AxisActualPosition</i>	DELTA Joint J1 MCD
<i>DELTAJ2AxisActualPosition</i>	DELTA Joint J2 MCD
<i>DELTAJ3AxisActualPosition</i>	DELTA Joint J3 MCD
<i>DELTA GripperOn</i>	DELTA Gripper ON MCD
<i>CARTESIANX AxisActualPosition</i>	CARTESIAN XPOS MCD
<i>CARTESIANY AxisActualPosition</i>	CARTESIAN YPOS MCD
<i>CARTESIANC AxisActualPosition</i>	CARTESIAN CPOS MCD
<i>CAPPINGY AxisActualPosition</i>	CAPPING YPOS MCD
<i>CAPPINGZ AxisActualPosition</i>	CAPPING ZPOS MCD

Tabla 3.4: Listado señales de entradas del sistema de manufactura.

<i>Señal PLC</i>	Señal NX MCD
<i>PalletConveyorSensor</i>	Pallet Conveyor Sensor MCD
<i>Pack1ConveyorSensor</i>	Pack1 Conveyor sensor MCD
<i>Pack2ConveyorSensor</i>	Pack2 Conveyor sensor MCD
<i>MainConveyorS1</i>	Main Conveyor S1 MCD
<i>MainConveyorS3</i>	Main Conveyor S3 MCD

Una vez configuradas todas las señales es necesario traer las variables del servidor OPC UA mediante el asistente de señales externas. En la figura 3.23 se observa la configuración del servidor OPC UA, con un tiempo de actualización a un milisegundo y la activación de las variables que fueron utilizadas en el sistema de manufactura virtual. Cabe resaltar que la herramienta de software permite conectar en simultaneo o individualmente con diversos protocolos de comunicaciones como los son: OPC DA, OPC UA, TCP, UDP, Profinet entre otros. Adicionalmente, se recomienda cerrar el programa y reiniciar cuando se realizan cambios o adiciones de nuevas variables para actualizar de manera completa la comunicación.

**Figura 3.23:** Configuración de conexiones externas en NX MCD.[Autor]

Como paso final de las comunicaciones se requirió realizar el mapeo de señales y así conectar el adaptador con las señales del servidor. Para ello en el software ingresamos al comando mapeo de señales como se observa en la figura 3.24 y del lado izquierdo se selecciona la señal de NX MCD y del lado derecho la señal del servidor, siempre y cuando el tipo de dato sean compatibles es posible conectarlos mediante el icono central y de esta manera aparecerán en la parte inferior todas y cada una de las señales activas. Para esta aplicación se realizaron 23 mapeos de señales entre entradas y salidas.

3. Implementación

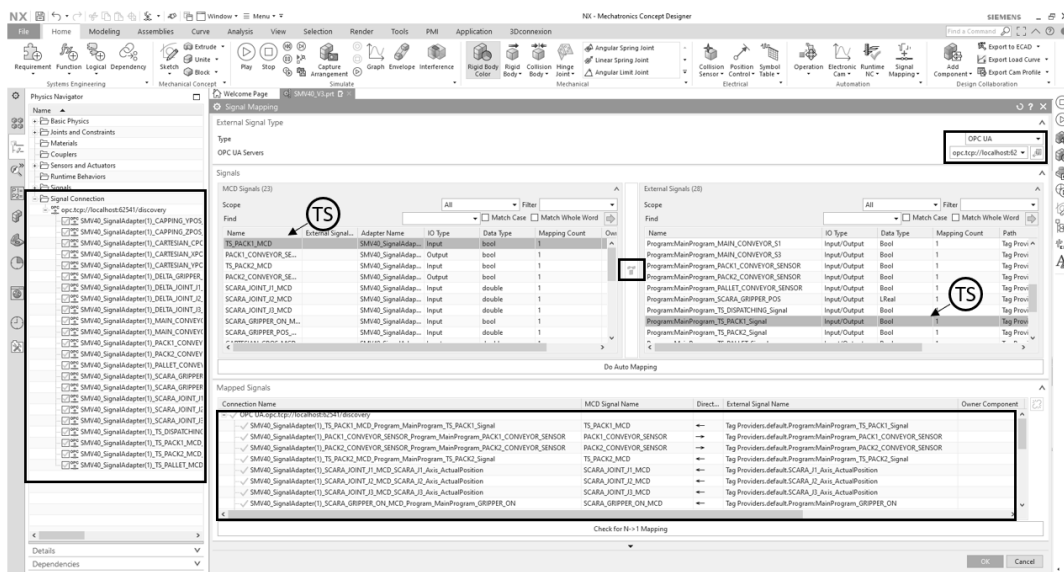


Figura 3.24: Mapeo de señales entre el servidor OPC UA y NX MCD.[Autor]

3.3. Modo de producción de la aplicación

Para poner en funcionamiento el sistema de manufactura con funcionalidades ciberfísicas es necesario atender inicialmente las funciones de operación para alcanzar la producción esperada. Como primer paso se realizó el GRAFCET de descripción funcional del módulo de almacén como se observa en la figura 3.25. Iniciando con el encendido del módulo, una vez se cumpla dicha condición dependiendo de la orden de fabricación existen dos posibilidades: producto 1 o producto 2 creando una bifurcación selectiva. Independientemente del caso se seguirán etapas muy similares, comenzando por el encendido de la banda de recipientes, para aproximar el recipiente al sensor al final de la banda el cual una vez sea activado enviará una señal de apagar banda pasando a la etapa 2. Una vez se apague se cumplirá la condición para pasar a la siguiente etapa de inicio de rutina de pick and place del robot SCARA. Finalmente, cuando el robot entregue el recipiente en la banda transportadora principal se retornará al inicio del GRAFCET, comenzando la rutina nuevamente.

Las rutinas de control de movimiento se desarrollaron en Ladder para ejemplificar el salto de GRAFCET a Ladder y poder hacer uso de los bloques de movimiento coordinado. Para este trabajo no se desarrolló cinemática inversa de ningún sistema, comandando los actuadores en cinemática directa por medio del comando MCLM Motion Coordinated Linear Move. Para hacer uso de este comando y continuar con el desarrollo del gemelo digital se configuraron servos virtuales para cada eje de movimiento. Adicionalmente, se crearon grupos coordinados de movimiento que permitieron interpolar los movimientos de los ejes acoplados para cada una de las instrucciones. Controlar los ejes y grupos en un mismo bloque permite controlar parámetros como posición, velocidad, perfiles de velocidad, aceleración, jerk en el grupo de manera coordinada, además de poder unir varios movimientos de la trayectoria con la función merge.

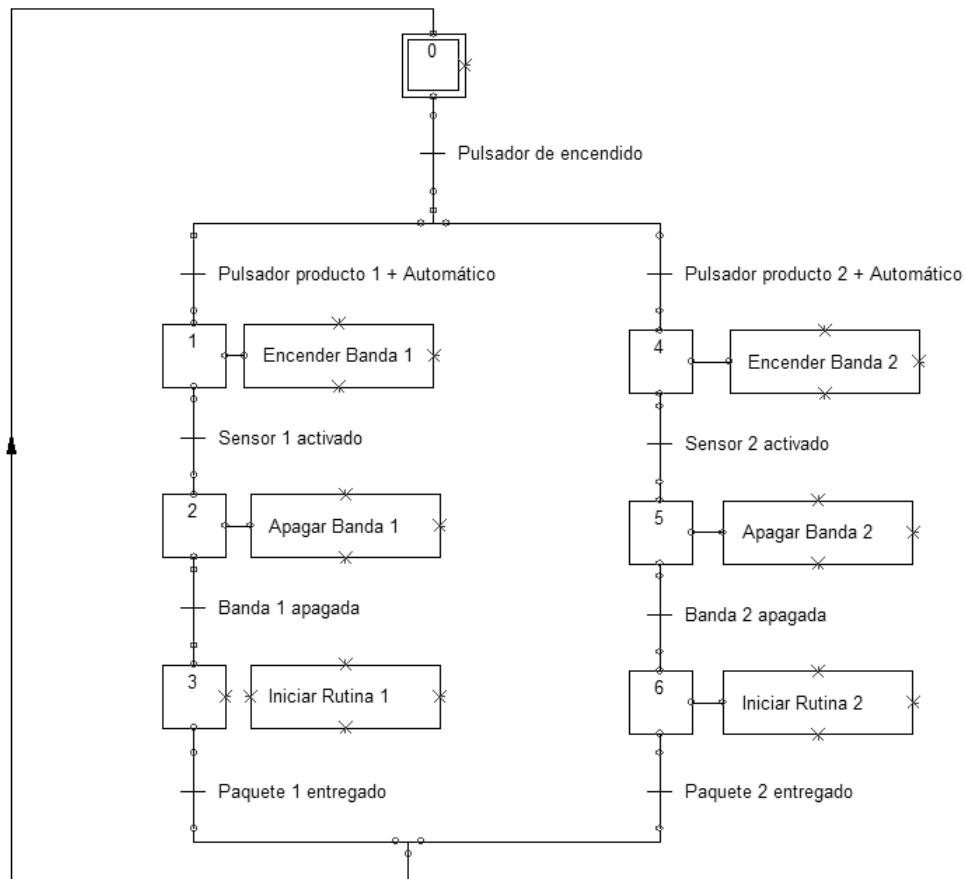


Figura 3.25: Grafset Nivel 1 Estación de almacén.[Autor]

En la figura 3.26 se puede observar el proceso de creación de un eje virtual y un sistema de coordenadas que agrupa varios ejes para la coordinación de movimiento. Antes de comenzar la programación es importante alistar los ejes y los sistemas coordinados. En la estación de almacén se crearon 3 ejes asignando Scara J1, Scara J2 y Scara J3 para cada articulación respectivamente y se creó un sistema denominado SCARA Coordinated. A cada eje se le asignó unas características dinámicas de máxima velocidad, aceleración, desaceleración, máxima aceleración de jerk y máxima desaceleración de jerk. De igual manera al sistema coordinado se le asignaron dichas propiedades para verlo como un conjunto siendo estas las capacidades resultantes del movimiento coordinado, además de una tolerancia de posición.

Para las siguientes estaciones se utilizó el mismo flujo de trabajo creando para la estación de llenado y tapado los ejes Capping Y, Capping Z, Cartesian C Axis, Cartesian, X Axis y Cartesian Y Axis, agrupando únicamente los ejes Cartesian XYZ para el movimiento coordinado buscando la interpolación de dichos ejes en el sistema Cartesian Coordinated y dejando indexados a los ejes Capping YZ. Para la estación de despacho se crearon los ejes Delta J1, Delta J2, Delta J3 y un sistema coordinado Delta Coordinated. Con esto se logró virtualizar no solamente señales, si no también ejes de movimiento que permitieron validar control de posición en actuadores.

3. Implementación

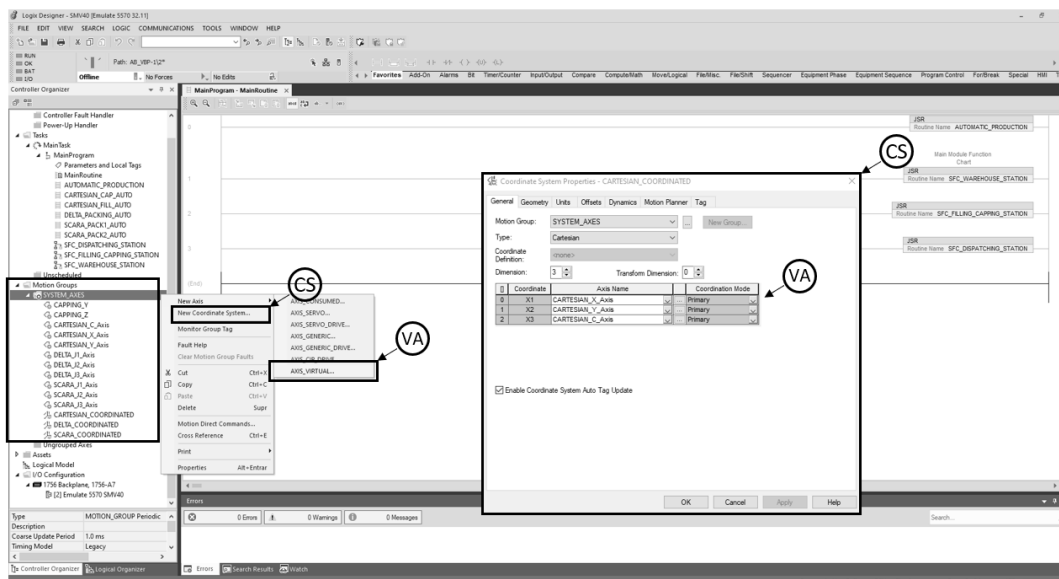


Figura 3.26: Configuración de ejes virtuales y sistemas coordinados.[Autor]

En la figura 3.27 se puede observar la estructura recomendada de programación de un bloque de movimiento coordinado, donde se recomienda esperar al fin de la ejecución del movimiento para enviar otra tarea y de esta manera darle paso al siguiente movimiento. En este bloque se pueden observar los siguientes estados: .EN indicando que el bloque esta activo, .DN que ha sido asignada la tarea y está en cola, .ER indicando el error por alguna causa en la etapa de verificación del movimiento, .IP indicando que se encuentra en proceso y los ejes están moviéndose a su destino y finalmente cuando termine el desplazamiento se activará la bobina PC. Dentro de los parámetros requeridos en este bloque está el sistema coordinado que ejecutará la acción, y la posición final después del movimiento de cada eje asociado al grupo, esta se declaró una variable de tipo real [1,3] creando un arreglo para almacenar las coordenadas de movimiento. Del mismo modo es necesario ingresar las variables de velocidad, aceleración, etc. que debe experimentar la resultante de movimiento y el bloque internamente calculará las condiciones para cada eje.

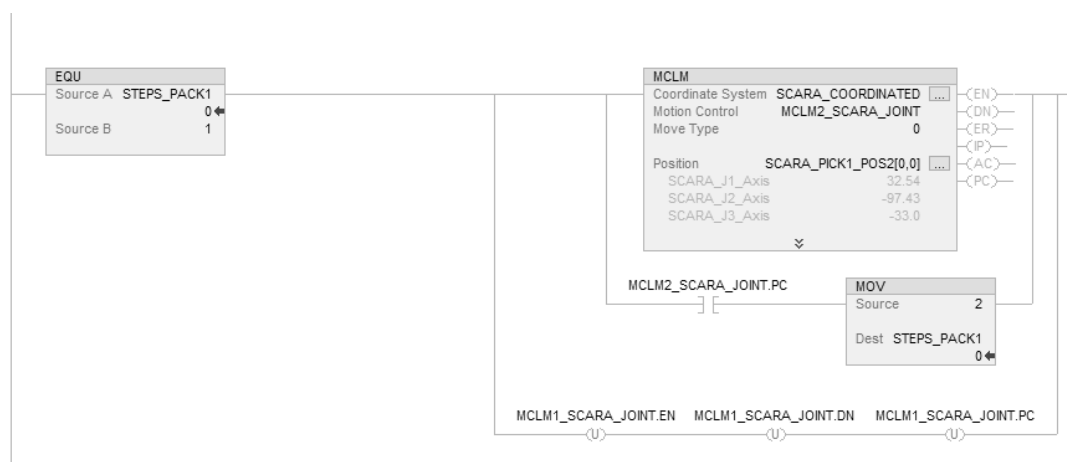


Figura 3.27: Estructura de programación de bloque MCLM.[Autor]

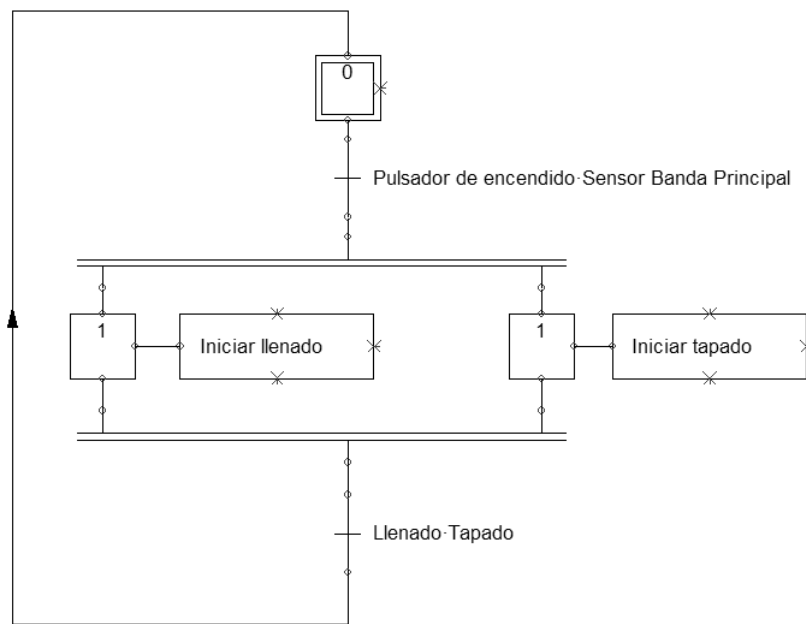


Figura 3.28: Grafcet Nivel 1 Estación de llenado-tapado.[Autor]

En la figura 3.28 se puede observar el GRAFCET de descripción funcional del módulo de llenado y tapado, el cual ejemplifica una bifurcación en simultaneidad que implica que ambas tareas sean desarrolladas al mismo tiempo, una vez se cumplan las condiciones del pulsador de encendido y la activación del sensor de la banda principal. Al final del proceso ambas condiciones se deben cumplir de igual manera, para dar paso a la siguiente etapa. En el proceso de llenado se realizó una tarea de llenado al vuelo lo que implica una coordinación de movimiento de los ejes lineales y rotacionales y para el proceso de tapado se realizan movimientos indexados.

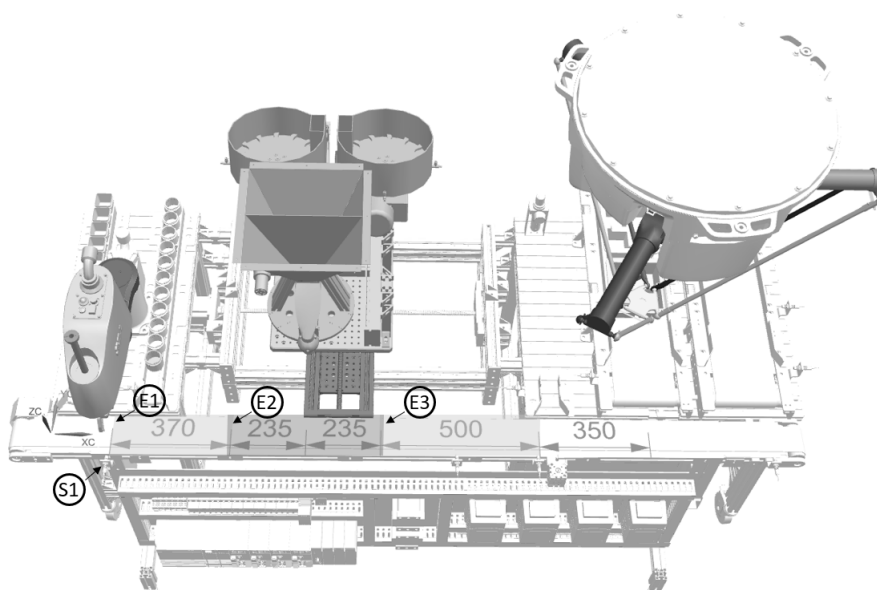


Figura 3.29: Zonas de sincronización de llenado al vuelo.[Autor]

3. Implementación

En la figura 3.29 se observan los puntos que representan las etapas de sincronización, E1 la zona para el encuentro, E2 la zona de sincronización y E3 la zona de desincronización. De esto se puede entender que: una vez el recipiente ingrese a la banda transportadora recorrerá una distancia de $E1=370$ mm a una velocidad constante de $V_R=60$ mm/s y una vez se encuentre con el sistema cartesiano tendrá $E2=470$ mm para el llenado del recipiente, una vez concluya dicho proceso el sistema de tapado estará listo para aplicar la tapa.

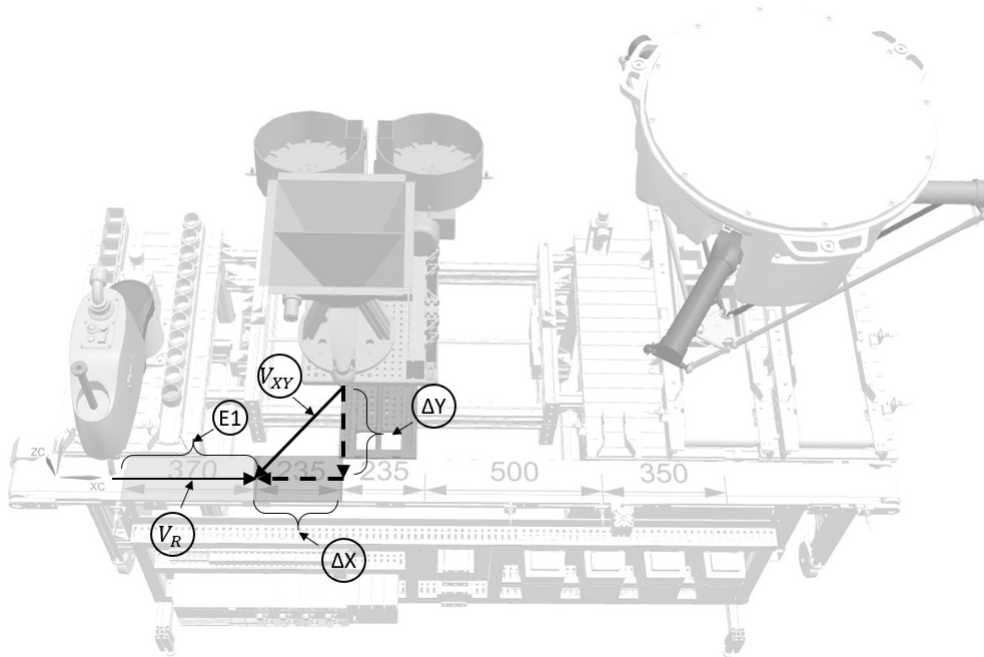


Figura 3.30: Esquema de sincronización de llenado al vuelo.[Autor]

Para determinar la velocidad que se emplearía en el bloque de coordinación de movimiento lineal para el sistema cartesiano se utilizó la ecuación 3.3 basada en el triángulo de velocidades como se muestra en la figura 3.30, tomando como base que el tiempo de llegada del recipiente una vez ingresado a la banda transportadora será igual al tiempo de llegada del sistema de llenado al punto de encuentro E2, ver ecuación 3.1 y ecuación 3.2. Esta ecuación nos permite parametrizar el sistema, para que una vez deje el estado de inicio pueda estar en cualquier posición y llegar coordinadamente al encuentro en el punto E2.

$$t_R = t_{XY} \therefore \quad (3.1)$$

$$\frac{E1}{V_R} = \frac{\Delta_X}{V_{XY}} \therefore \quad (3.2)$$

$$V_{XY} = \frac{V_R \Delta_X \cos(\arctan(\frac{\Delta_Y}{\Delta_X}))}{E1} \quad (3.3)$$

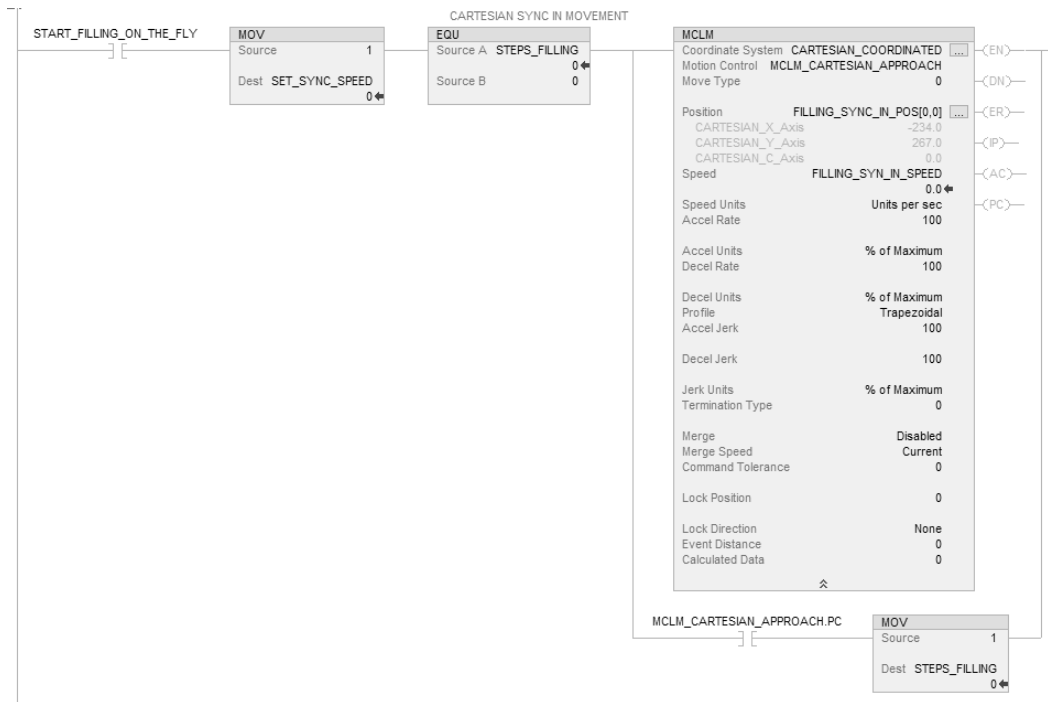


Figura 3.31: Parámetros de velocidad y perfil en zona para el encuentro.[Autor]

En la figura 3.31 se observa la configuración del bloque de movimiento lineal coordinado donde se ingresa el valor del vector de velocidad resultante indicando la posición de encuentro y la velocidad calculada anteriormente. El bloque nos permite seleccionar entre perfil trapezoidal o en curva-s para este caso se seleccionó un perfil trapezoidal asignando el 100 por ciento de la aceleración y desaceleración.

$$a = \frac{4,5d_t}{t_t^2} \quad (3.4)$$

Para la configuración del eje virtual se consideró la ecuación 3.4 para determinar la aceleración máxima necesaria para el recorrido, obteniendo esta de un perfil trapezoidal estándar, donde 1/3 del tiempo es usado para la aceleración, 1/3 para velocidad constante y 1/3 para la desaceleración. Una vez el sistema de llegada se encuentra en el punto de encuentro se programa un bloque MCLM a velocidad constante para coordinar la velocidad de ambas partes y lograr el llenado al vuelo. El bloque MCLM en la opción MERGE brinda la capacidad de unir movimientos y no desacelerar al momento de conectar dos movimientos esta opción se activa para la segunda parte y no generar una desaceleración en el punto de encuentro y comenzar la sincronización.

El bloque MCLM se encargará de calcular las velocidades y aceleraciones de cada eje en el grupo de movimiento del sistema cartesiano. De esta manera los ejes interpolarán generando un movimiento coordinado y sus variables podrán ser leídas de los atributos del eje virtual en forma de variables como: Axis.ActualPosition, Axis.ActualVelocity, etc.

3. Implementación

En la figura 3.32 se puede observar el GRAFCET de descripción funcional del módulo de despacho, el cual ejemplifica una ejecución lineal sin ramificaciones. Este proceso comienza con el encendido del módulo para pasar a ingresar una estiba al proceso, una vez toca el sensor al final de la banda transportadora esta se detiene y da inicio a la rutina de llenado que se realiza por medio del robot delta realizando una tarea de pick and place hasta alcanzar una capacidad de 4 recipientes y posteriormente traslada la estiba a una segunda banda que se encarga de dar salida al producto mediante un sensor que activa la misma y de esta manera se repite el ciclo.

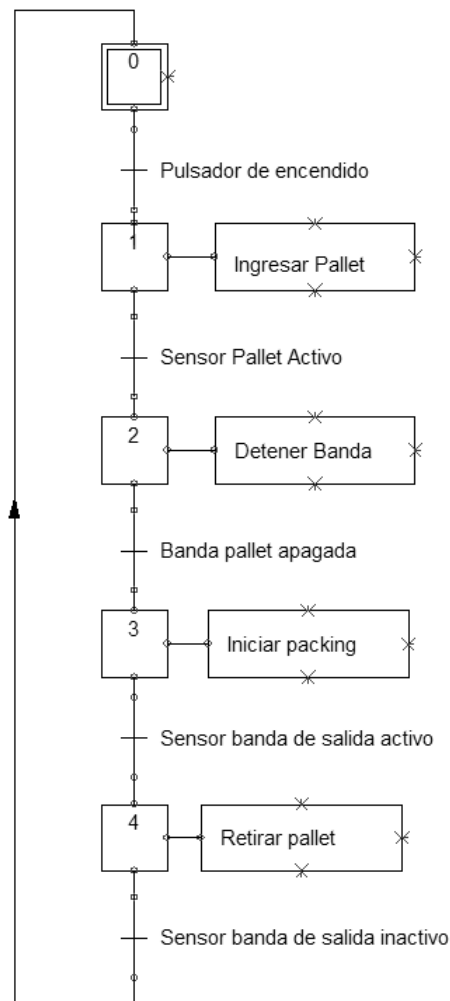


Figura 3.32: Grafcet Nivel 1 Estación de despacho.[Autor]

Para el empaclado al vuelo se realizó un calculo similar al del llenado al vuelo definiendo una zona para el encuentro E2 como se observa en la figura 3.33, esta señal es activada una vez el recipiente es detectado por el sensor S3 donde el recipiente viajará un recorrido de 350 mm a una velocidad constante de 60 mm/s, y el delta tendrá ese tiempo para llegar al punto E2 donde se realizará la tarea de pick.

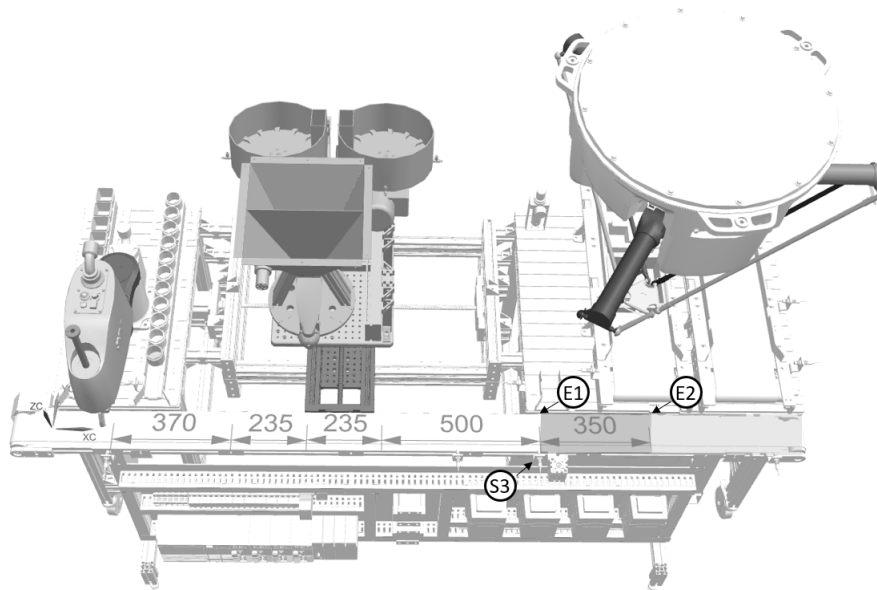


Figura 3.33: Zonas de sincronización de empaqueo al vuelo.[Autor]

3.4. Modo seguro de la aplicación

Es importante resaltar que no en todos los procesos industriales requieren desarrollar todas las etapas de la guía GEMMA, esto dependerá de su complejidad y nivel de seguridad. Para este caso particular se seleccionaron 4 estados de evolución que deben estar presentes en un sistema de manufactura como lo son: el de producción normal o automático, la parada a final de ciclo, el de inicio marcha etapa a etapa con verificación de orden para validar la correcta operación de todos los sistemas después de un abastecimiento, mantenimiento o falla y la parada de emergencia debida a un estado de defecto en operación normal.

Para evaluar la evolución por los modos de la guía GEMMA fue necesario el uso de una interfaz gráfica, que sirvió de apoyo para validar el estado de cada una de las acciones y para accionar algún estado en particular. Esta interfaz se desarrollo en la plataforma DESIGN LAUNCHER de INDUCTIVE AUTOMATION en IGNITION el cual toma las variables del servidor, permitiendo lectura y escritura de las mismas y así contar con funcionalidades de supervisión y control.

El primer modo expuesto es el de marcha por ciclos y parada a fin de ciclo, este procedimiento comienza en la etapa A1, que es el estado inicial de reposo de la máquina y cuando las condiciones necesarias se cumplen se pasa a funcionar en modo normal F1 y cuando el operador pulsa el pulsador de parada de fin de ciclo pasará a A2, parada solicitada a final de ciclo y cuando termine el ciclo pasará al estado inicial A1 como se observa en la figura 3.34. Este modo permite la operación mínima necesaria de un sistema de manufactura.

3. Implementación

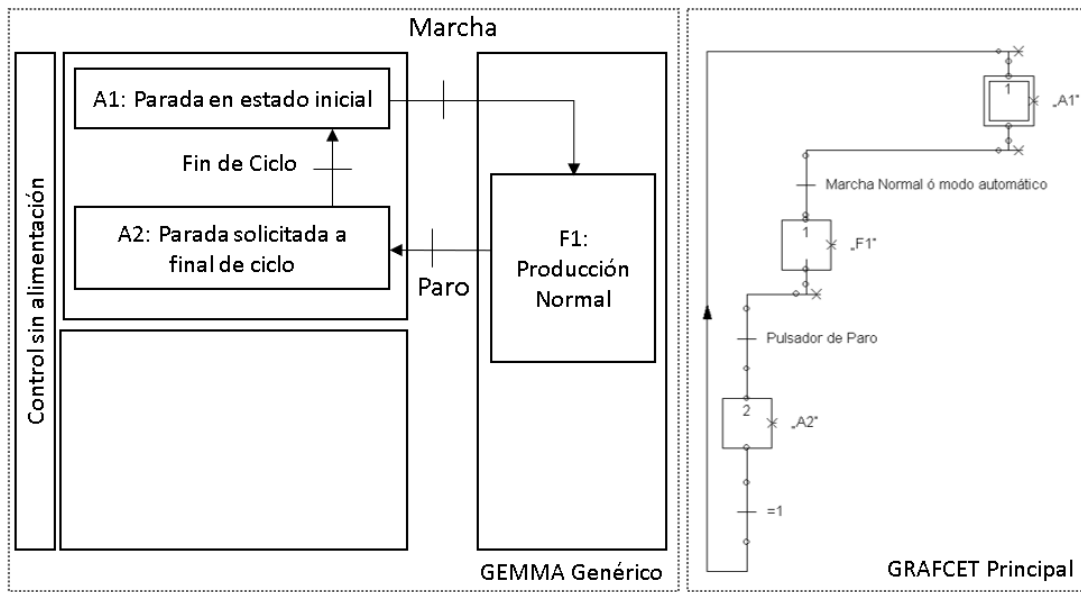


Figura 3.34: Esquema de parada pedida a final de ciclo.[Autor]

El siguiente modo expuesto es el de marcha de verificación con orden, ver figura 3.35, iniciando en A1, estado inicial de reposo de la máquina y puede iniciar en dos caminos, el primero es el de marcha normal el cual entrará al estado de producción normal y posteriormente si el operario pulsa el botón de paro pasaría a la etapa A2, parada pedida a final de ciclo y en seguida a la etapa A1. Por otro lado, si se va a marcha de etapa a etapa pasaría a F5, una vez en esta etapa llegaría a la etapa de producción a través de la marcha normal o la etapa de producción normal.

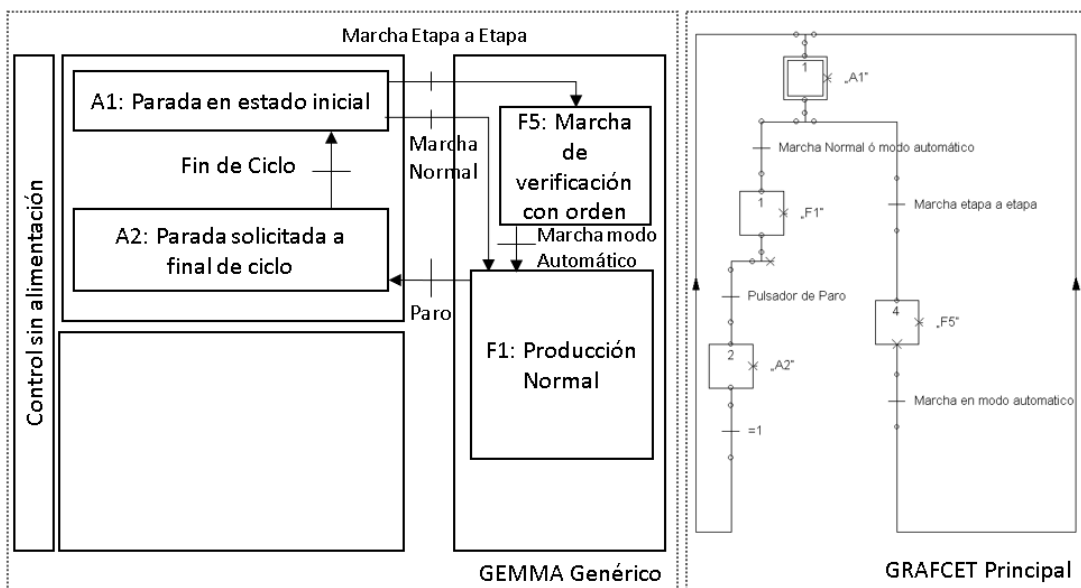


Figura 3.35: Esquema de marcha de verificación con orden.[Autor]

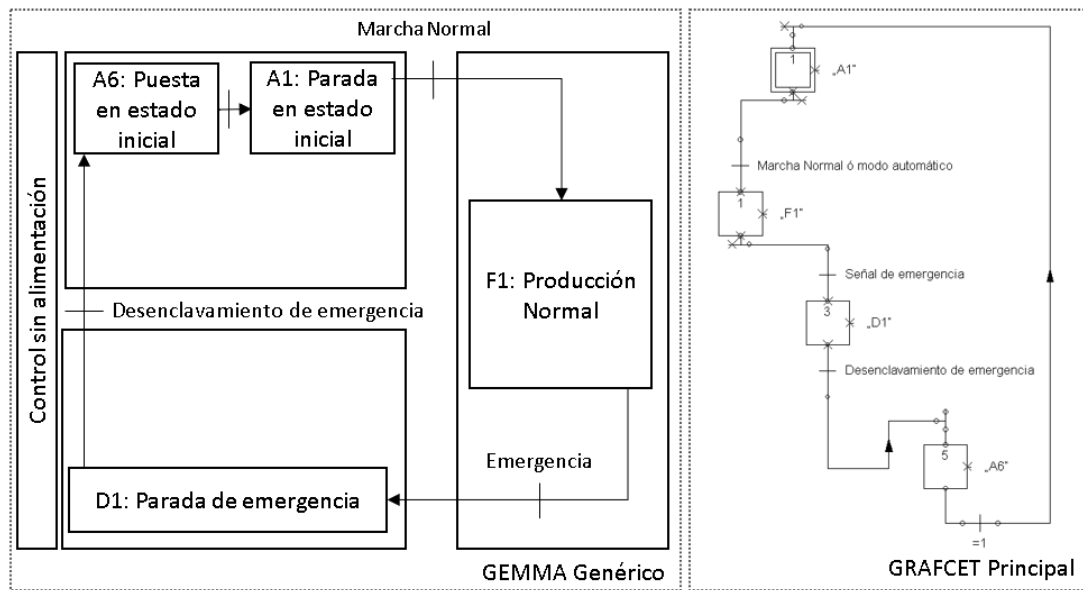


Figura 3.36: Esquema de parada de emergencia.[Autor]

Finalmente, uno de los procesos más importantes: el de parada de emergencia, ver figura 3.36, en este caso si se está en el estado de producción normal F1, y se pulsa el pulsador de parada de emergencia entonces toda la planta se quedará sin alimentación excepto el sistema de control y posteriormente el pulsador de parada de emergencia informa al control que pase a D1, y al desenclavarse el pulsador de emergencia pasará a A6 puesta en estado inicial y si todas estas condiciones se cumplen pasaría a A1.

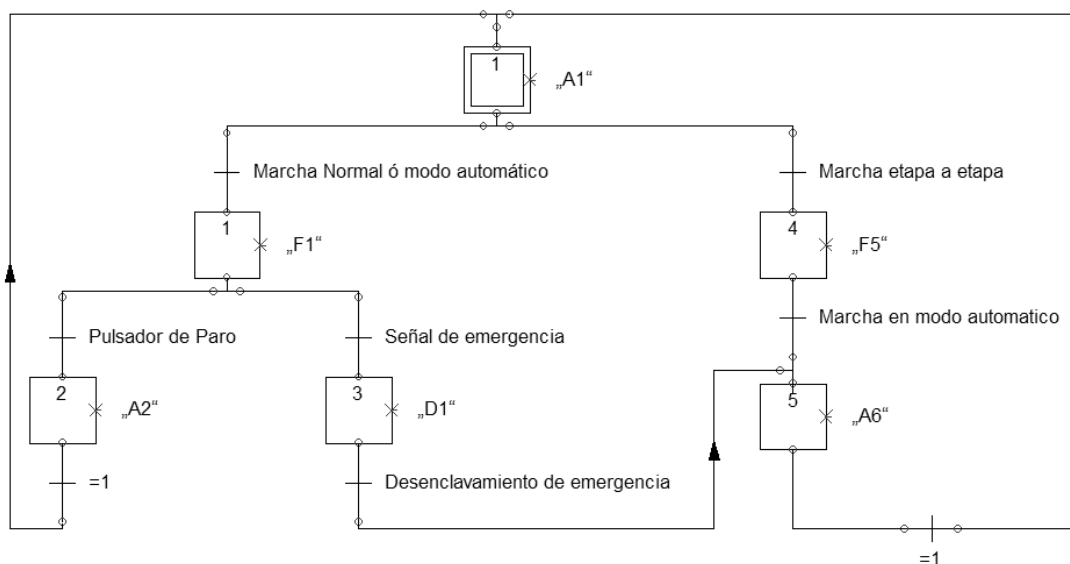


Figura 3.37: Grafcet Nivel 1 principal aplicando guía GEMMA.[Autor]

Para este sistema de manufactura virtual se integraron los tres modos mencionados anteriormente en los cuales puede evolucionar el código dependiendo la situación

3. Implementación

dada como se muestra en el graficet principal en la figura 3.37. en el se puede evidenciar el estado de producción normal o automático, el estado de marcha etapa a etapa para la verificación de funcionamiento de todas las partes y la condición de parada de emergencia para casos de defectos dentro de la operación.

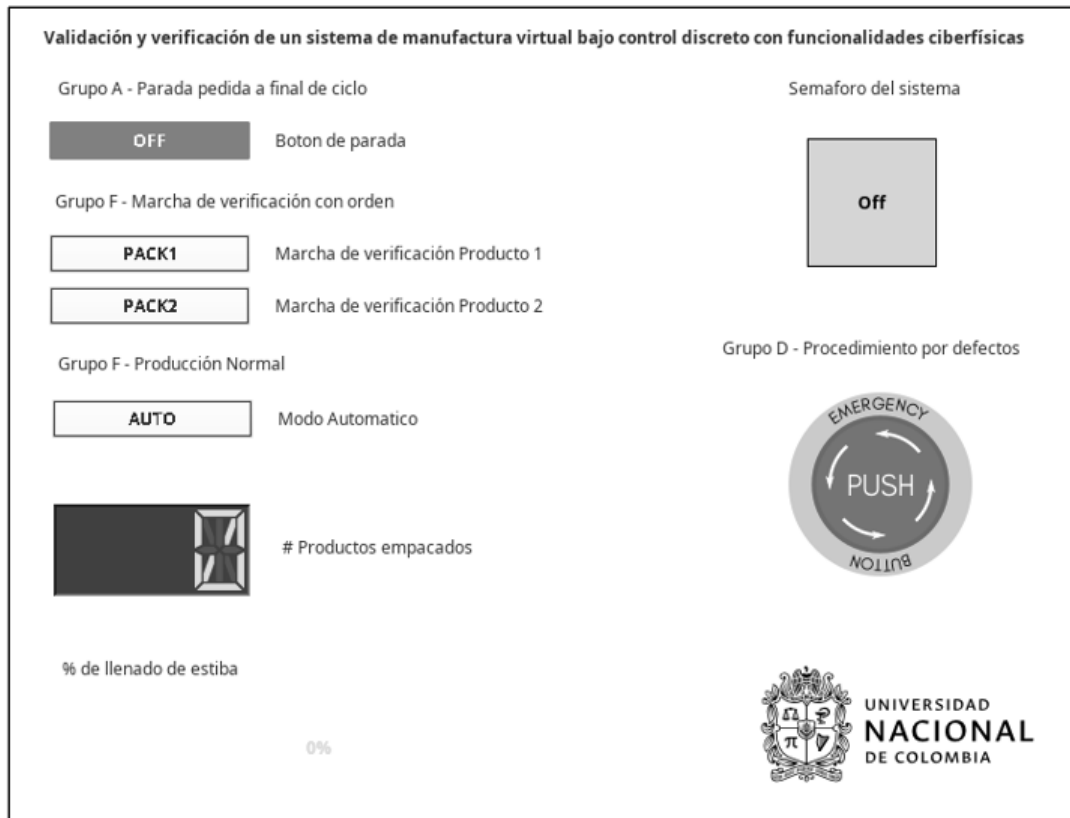


Figura 3.38: HMI para accionar y evidenciar estados de la guía GEMMA.[Autor]

A continuación se muestra la interfaz en la imagen 3.38, para verificar el estado de evolución de los GRAFCET, la cual del lado izquierdo posee botones de accionamiento con dos estados, tanto para la pedida de parada a final de ciclo, la marcha de verificación con orden y la producción normal, también cuenta con un indicador para saber el estado de productos empacados y llenado de estiba para verificar el modo de producción normal ó automático. Del lado derecho se encuentra un semáforo de 4 estados indicando: apagado, modo manual, automático y falla para validar de forma visual el estado del sistema de manufactura y para la parada de emergencia un botón en caso de una condición de defecto, el cual da la orden de evolución a dicho estado.

Con todos estos elementos se ejemplifica una forma sistemática de desarrollar lógica teniendo en cuenta factores más allá de solamente la función de producción, además de integrarlos a un sistema de manufactura virtual y su posible validación en el gemelo digital del controlador.

4

Resultados y Análisis

En este capítulo se contrastan los resultados obtenidos después de la implementación del sistema desarrollado en el capítulo anterior. Diversas herramientas fueron utilizadas para alcanzar el objetivo planteado al inicio de este documento. El sistema de manufactura virtual fue validado y verificado exitosamente con los resultados discutidos en esta sección.

4.1. Validación y verificación de la aplicación

Acorde a la metodología descrita en la sección 2.5 sobre las pruebas de validación virtual se probó cada grupo individualmente al momento de construir el sistema de manufactura virtual. Inicialmente se validó el diseño mecánico/eléctrico revisando que el ensamble con los actuadores, sensores y estructura se encontrara acorde a las especificaciones planteadas. Para validar la física de los objetos se inicia la simulación y de esta manera aquellos cuerpos que carezcan de restricciones caerán por efecto de la gravedad. Posteriormente, para validar el funcionamiento de actuadores se utilizó el comando JOG JOGGER como se muestra en la figura 4.1 donde se probó el movimiento individual de cada actuador mediante el uso de la barra deslizante al inducir el movimiento.

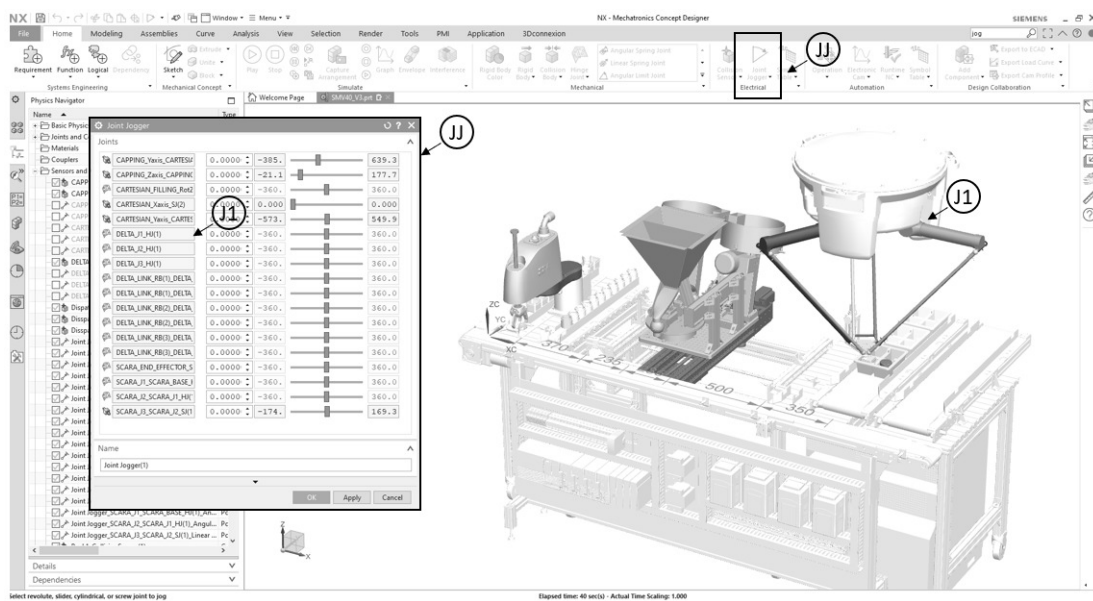


Figura 4.1: Validación de actuadores virtuales del modelo digital 3D.[Autor]

4. Resultados y Análisis

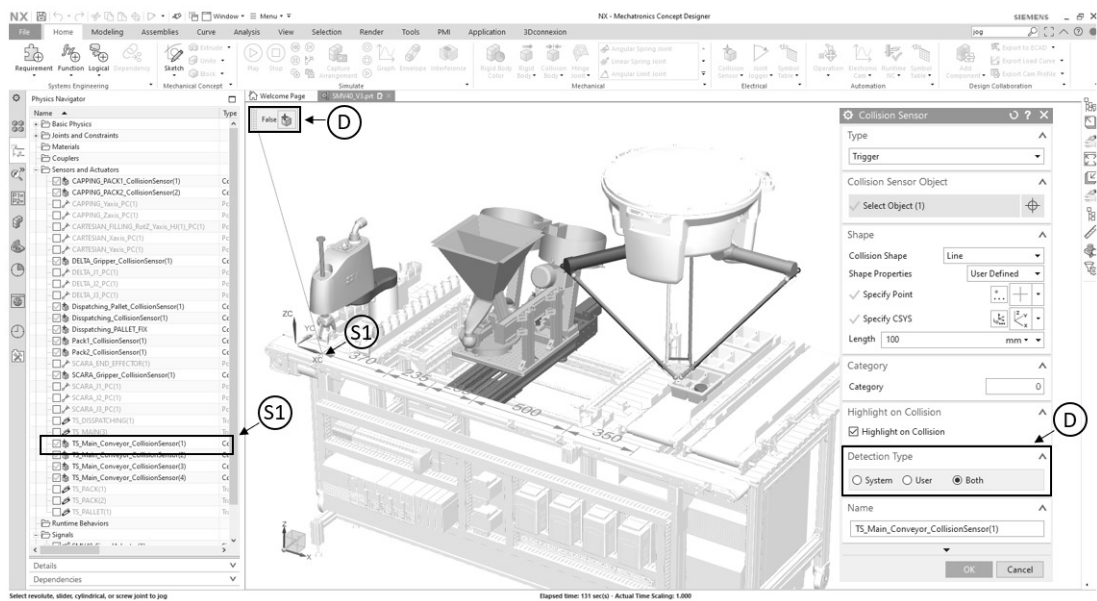


Figura 4.2: Validación de sensores virtuales del modelo digital 3D.[Autor]

Para validar los sensores se utilizaron las funcionalidades del software que permiten ver el estado y manipular su condición como se muestra en la figura 4.2. En la ventana gráfica cuando se crea la configuración del sensor es posible adicionar la funcionalidad de detección donde podemos seleccionar entre definición por el usuario, el sistema ó ambos para simular su accionamiento. Este procedimiento se realizó para verificar el estado de forma independiente antes de su integración y operación de la producción de los sensores utilizados en el sistema de manufactura. También, mediante este procedimiento es posible evaluar o accionar estados una vez el sistema está siendo operado por señales externas y realizar una verificación de acciones en el programa de control.

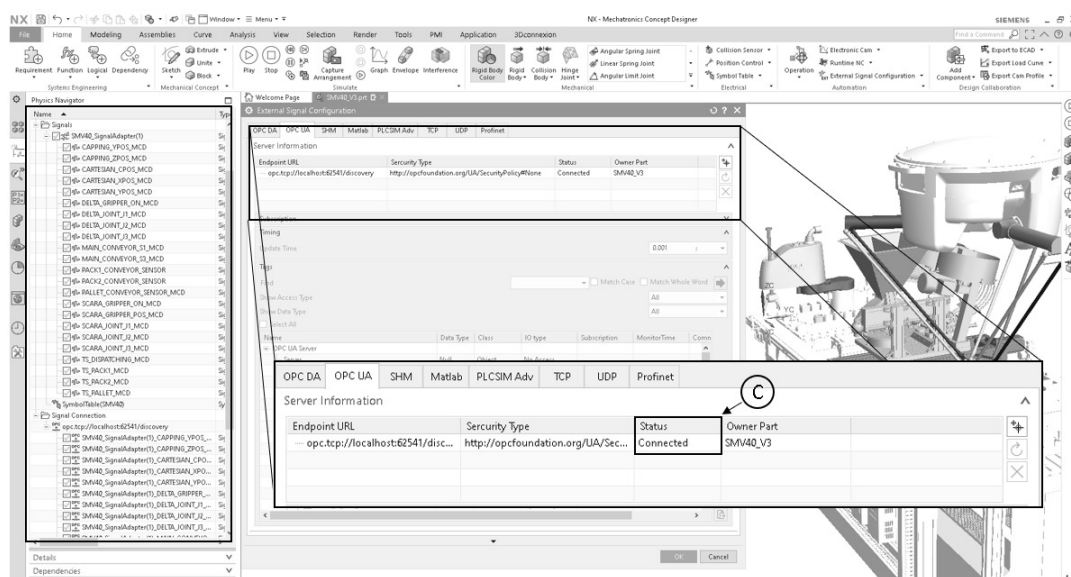


Figura 4.3: Validación del adaptador de señales del modelo digital 3D.[Autor]

La validación de la comunicación tanto de señales de entrada y salida al modelo digital 3D se observa en la figura 4.3. Mediante la ventana emergente del adaptador de señales es posible observar el estado en que puede encontrarse. Son posibles dos estados: conectado o sin conexión, quedando este último aislado al exterior. Este procedimiento se realiza cada vez que se inicie el programa o se realice un cambio ya sea por adición o supresión de alguna variable y de esta manera garantizar su existencia y concordancia con las variables configuradas.

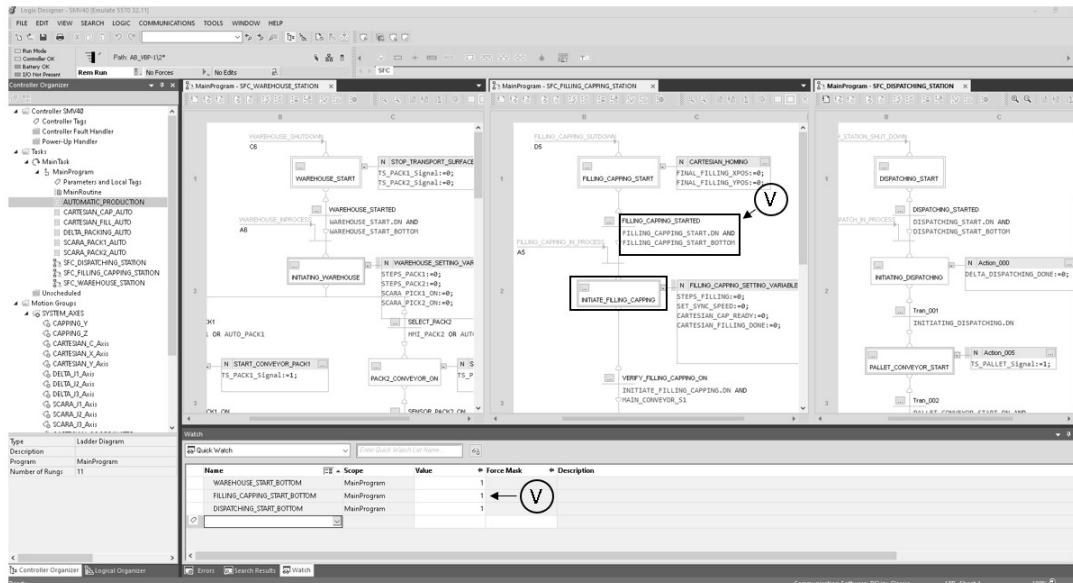


Figura 4.4: Validación de la lógica del programa en el controlador virtual.[Autor]

El proceso de validación de la lógica del programa de control mediante la herramienta WATCH como se ilustra en la figura 4.4, se pueden observar los valores de las variables, pero también es posible escribir sobre ellas y de esta manera validar la lógica del programa. Este procedimiento es muy útil para el programador para validar su programa antes de la integración con el sistema de manufactura. En esta etapa se validaron los programas realizados en GRAFCET y Ladder del sistema de manufactura virtual y su paso del principal con GEMMA a los secundarios de cada módulo. También, se pudo probar la lógica de los programas en Ladder de control de movimiento de los actuadores de cada equipo.

Cuando no se encuentra correctamente enlazada la HMI con los datos del controlador virtual debido al error de comunicación con el servidor OPC DA se evidencia error en el aplicativo de diseño como se muestra en la figura 4.5. Este procedimiento de validación permite revisar varios aspectos, inicialmente la comunicación entre el programa de la lógica de control y el controlador virtual, así mismo este debe comunicarse con el servidor OPC DA y permitir el acceso a los datos en el servidor OPC UA que este a su vez permite la lectura y escritura al diseñador de la interfaz HMI.

4. Resultados y Análisis

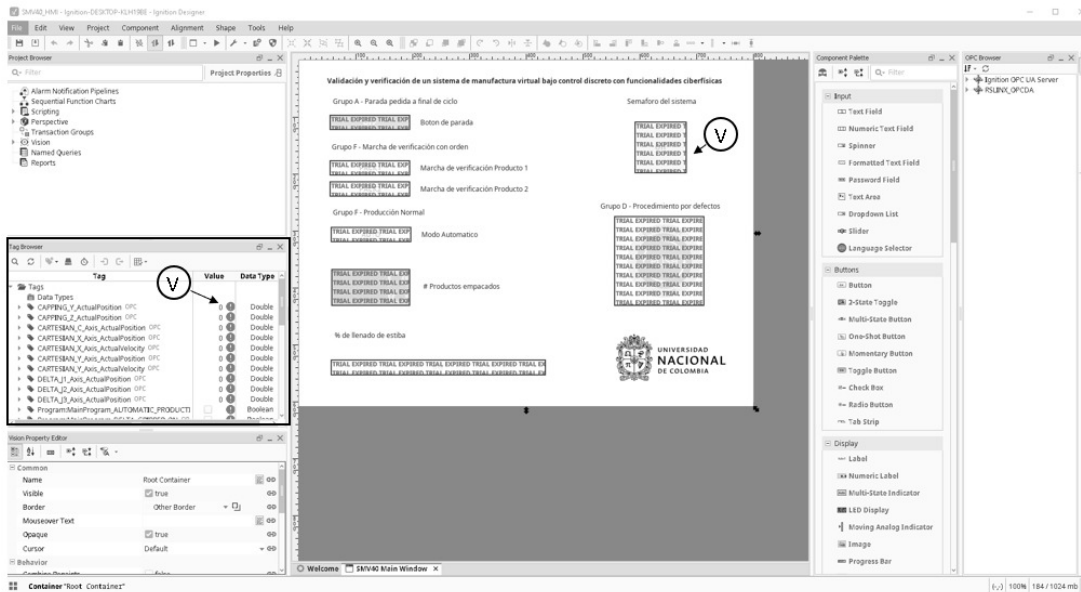


Figura 4.5: Validación de HMI y servidor OPC DA del controlador virtual.[Autor]

En la figura 4.6 se evidencia el estado de operación en modo automático donde se valida el procedimiento de funcionamiento. En ella se puede observar que el semáforo se encuentra en automático, el contador y la barra de progreso indican el estado del proceso. De esta manera es posible observar el estado de evolución del programa de control adicionalmente al movimiento del modelo digital 3D.

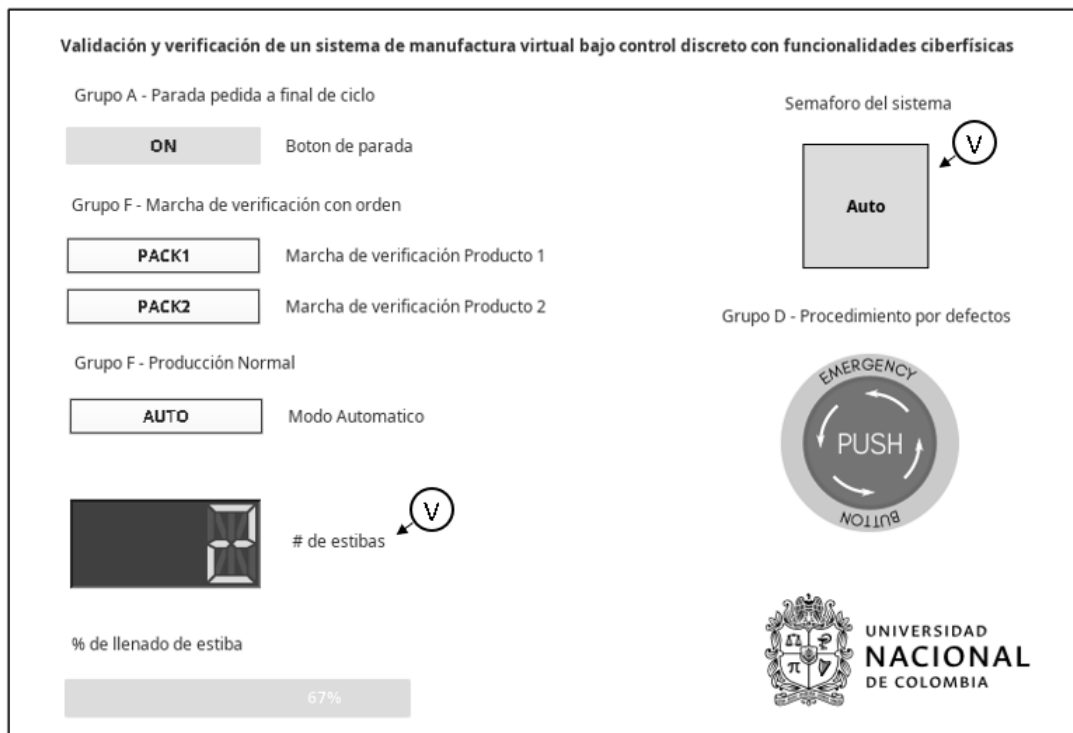


Figura 4.6: Validación de procedimientos de funcionamiento en HMI.[Autor]

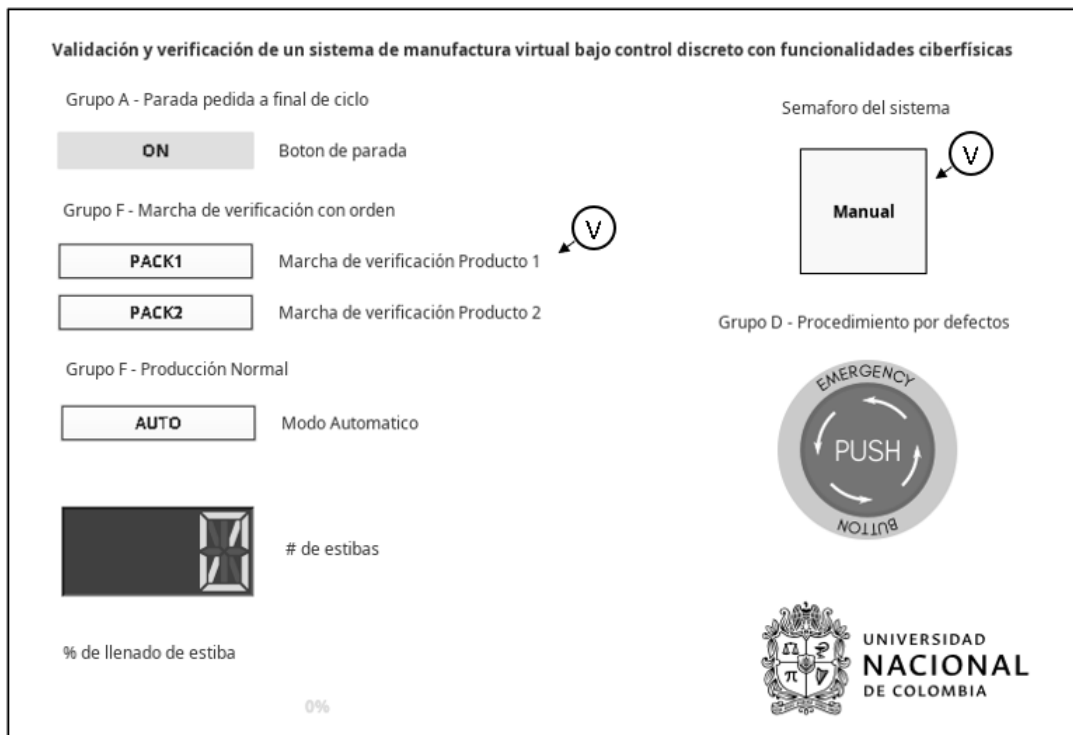


Figura 4.7: Validación de procedimientos de puesta en marcha en HMI.[Autor]

En la figura 4.7 se evidencia el estado de operación en modo manual donde se valida el procedimiento de marcha con verificación de orden donde se puede observar que el semáforo se encuentra en manual, para verificar la producción tanto del producto 1 y el producto 2. De esta manera es posible observar el estado de evolución del programa de control en una etapa de inicio y su movimiento etapa a etapa del modelo digital 3D.

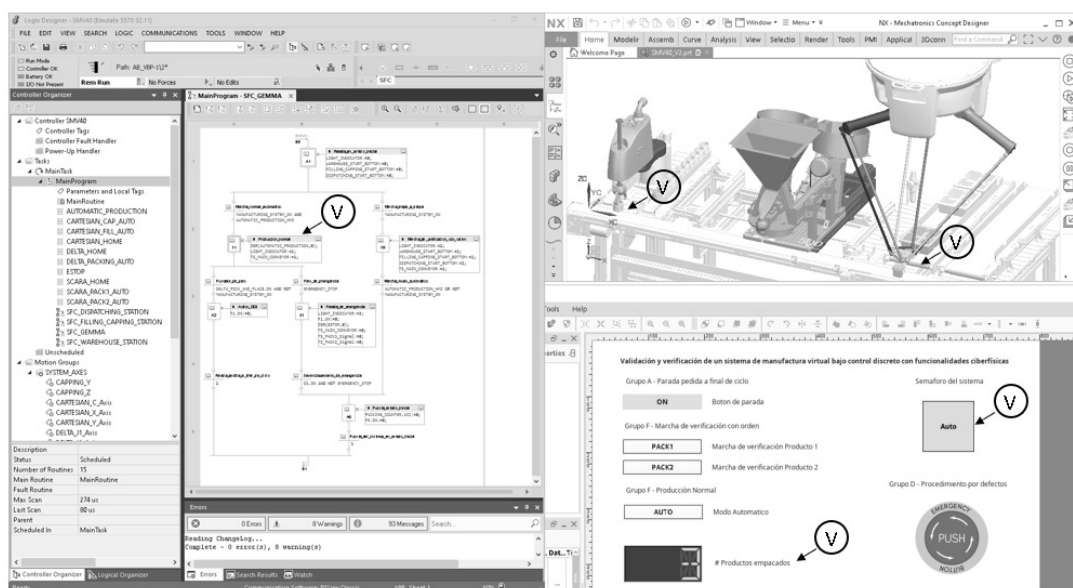


Figura 4.8: Validación de procedimientos de funcionamiento GEMMA.[Autor]

4. Resultados y Análisis

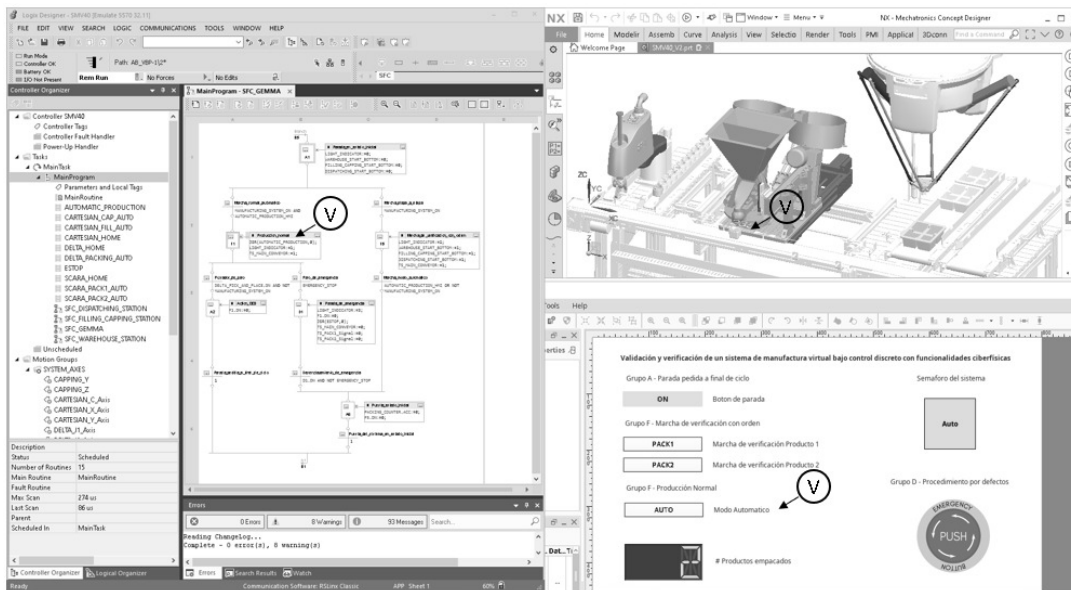


Figura 4.9: Validación de procedimientos de funcionamiento llenado.[Autor]

En la figura 4.8 y en la figura 4.9 se se observa la operación del conjunto de aplicaciones en concordancia realizando la ejecución de la lógica de control discreto en el sistema de manufactura virtual con funcionalidades ciberfísicas evolucionando en los modos acorde a la guía GEMMA. Es posible observar el funcionamiento de los tres equipos tanto del ingreso, llenado y empackado del recipiente.

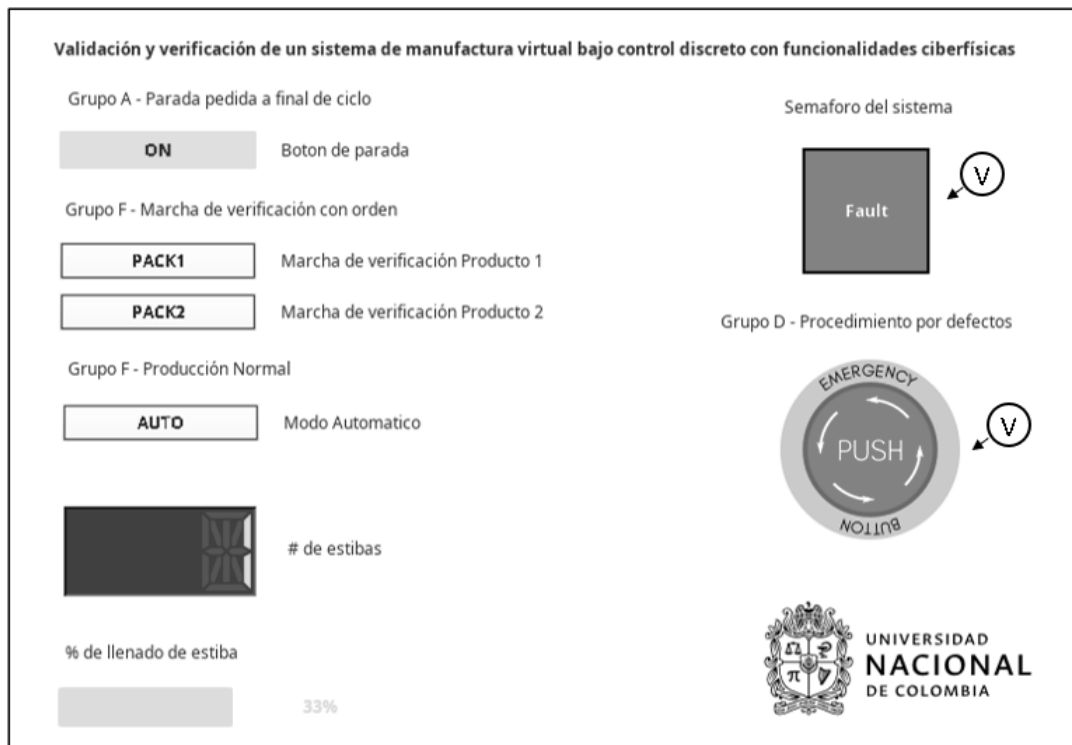


Figura 4.10: Validación de procedimientos de parada en HMI.[Autor]

En la figura 4.10 se evidencia el estado de operación debido a un defecto donde se valida el procedimiento de parada de emergencia donde se puede observar que el semáforo se encuentra en FAULT y se detiene la producción, así como los equipos. De esta manera es posible observar el estado de evolución del programa de control en una etapa de debida a un defecto y su resultado en el modelo digital 3D. Finalmente, en el siguiente video se puede observar la validación y verificación del sistema de manufactura virtual bajo control discreto con funcionalidades ciberfísicas: <https://drive.google.com/file/d/1xsjiAiDXCGx7YdWtRZWT1t2ZtA9hEWIv/view?usp=sharing>

4.2. Consideraciones de desempeño al simular

En la figura 4.11 es posible observar el gasto computacional que requiere poder simular todas las herramientas de software en un mismo equipo [20], procesando en este caso particular datos de control de movimiento para 9 actuadores los cuales se transmiten a una tasa de un milisegundo. En este caso se observa el uso del 100 por ciento del CPU del equipo y un consumo de memoria de 8.2 GB lo cual nos da una idea de las características del equipo a utilizar en el momento de realizar la validación y verificación de un sistema de manufactura virtual bajo control discreto con funcionalidades ciberfísicas. Para descargar el uso del equipo, se recomienda usar cada software de manera independiente conectándolos por medio de red y así dedicar un equipo a cada tarea siendo estas: Simulación virtual, SCADA-HMI y simulación de control.

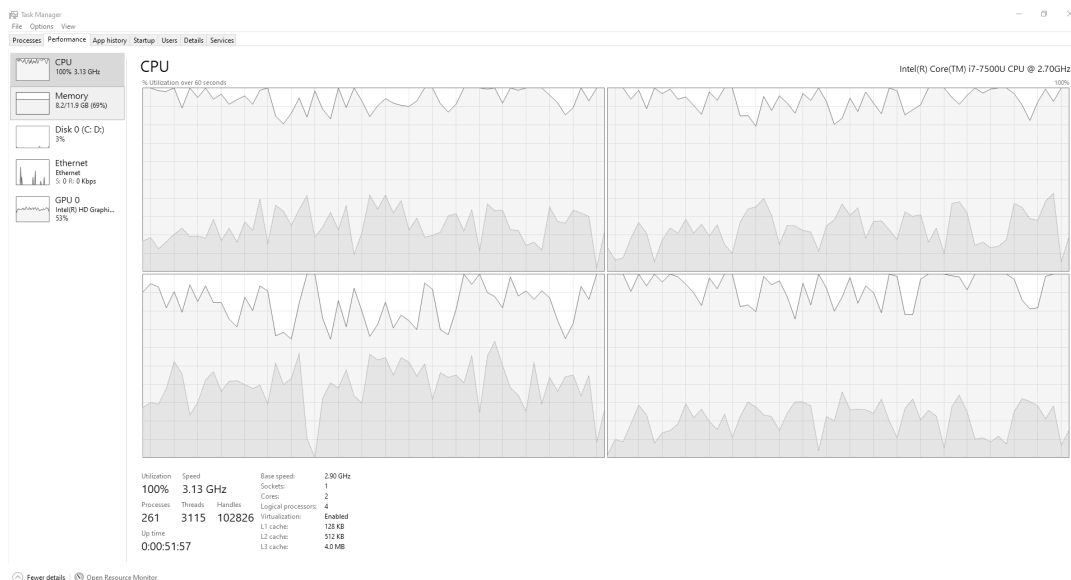


Figura 4.11: Rendimiento computacional del equipo utilizado.[Autor]

Durante el proceso de simulación fue necesario identificar varios aspectos para garantizar sincronización específicamente en la tarea de llenado al vuelo. Como se observa en la figura 4.12 la herramienta de software cuenta con una ventana para el escalamiento del tiempo, indicando en la parte inferior el tiempo actual, el tiempo de retraso en la simulación y el tiempo escalado, esto puede ser debido a la cantidad

4. Resultados y Análisis

de recursos requeridos en la simulación, otros programas corriendo en simultáneo y cantidad de cuerpos de colisión interactuando en el sistema.

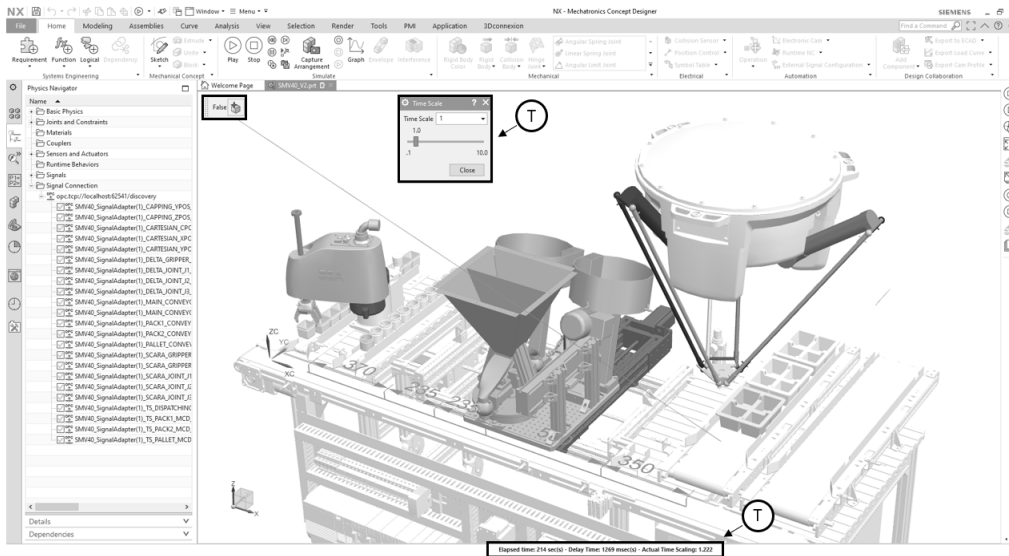


Figura 4.12: Escalamiento del tiempo de simulación.[Autor]

Para este caso particular, aunque se configuraron todos los parámetros no fue posible en todas las ocasiones garantizar el 100 por ciento de sincronización. En la figura 4.13 se observan los resultados obtenidos durante el llenado al vuelo donde se evidencia un desfase promedio de 3 mm, debido a dos factores principales: la velocidad de las comunicaciones y los recursos computacionales para la dinámica del sistema. Sin embargo, para fines de programación y revisión del llenado al vuelo cumple con el requerimiento, ya que es posible observar la aceleración para llegar al punto de encuentro y mantener la velocidad constante una vez se alcanza la zona de sincronización.

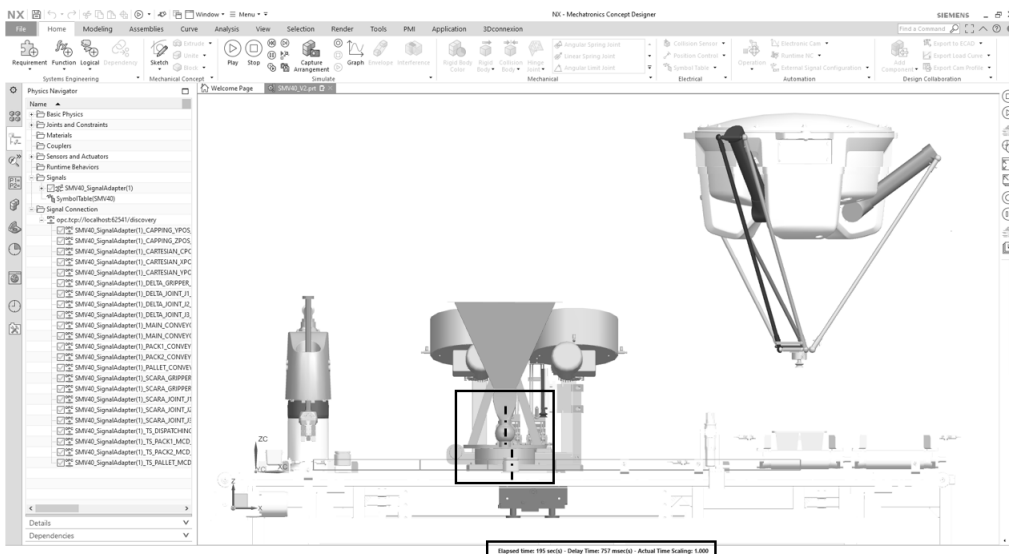


Figura 4.13: Error de precisión y exactitud de la simulación.[Autor]

4.3. Aplicaciones derivadas de la investigación

El presente proyecto también busca fortalecer los contenidos de espacios académicos en asignaturas como: Automatización de procesos de manufactura, Servomecanismos, Motores y controladores industriales, donde se pudo observar el aprendizaje de conceptos de sistemas ciberfísicos mediante el uso de herramientas de validación virtual y su aplicación en proyectos de curso. Para alcanzar dicho objetivo, se desarrollaron guías y material de apoyo que se encuentran en los anexos, para llevar paso a paso a estudiantes y docentes en la implementación y validación de sistemas de manufactura automatizados utilizando estas herramientas.

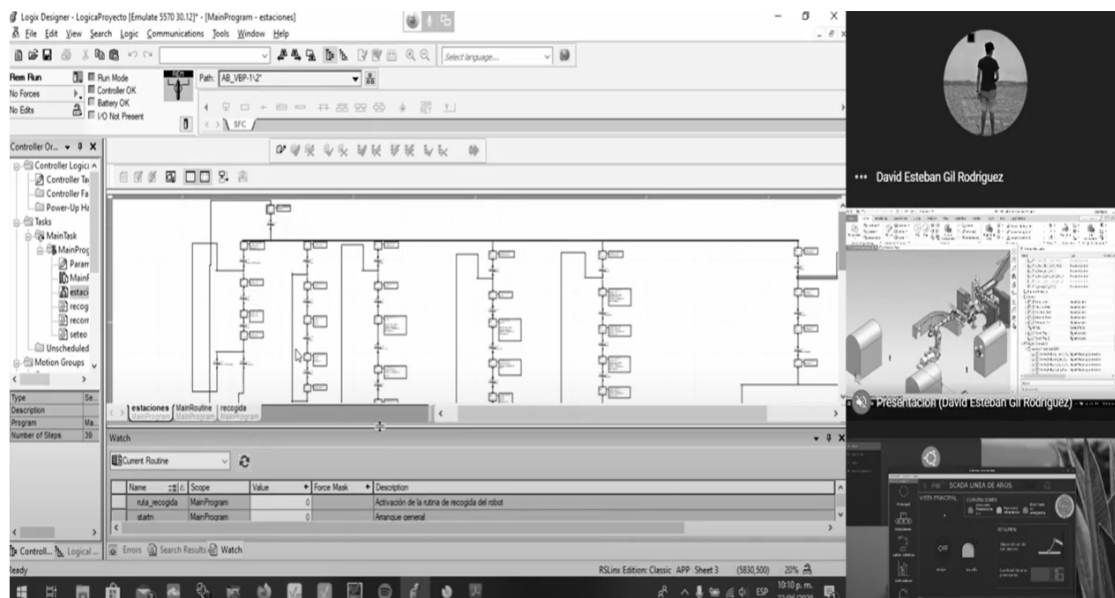


Figura 4.14: Aplicaciones en la nube de sistemas ciberfísicos.[Autor]

En algunos casos se logró extrapolar a otras marcas de software logrando integrarlas con el mismo enfoque, al incorporar un sistema WEB como lo es Ignition de Inductive Automation se abrió la posibilidad de conectar dispositivos en diversos lugares. Dentro de los trabajos realizados por los estudiantes de clase se evidencia la conexión en la nube al usar estas herramientas utilizando un servicio cloud y se logró operar en un computador el sistema de manufactura virtual, en otro computador la HMI y en otro computador el controlador virtual desde ubicaciones geográficas distintas, como se muestra en la figura 4.14.

Tabla 4.1: Proyectos curso de automatización de procesos de manufactura

<i>APM2020 – 1GrupoAros</i>	https://youtu.be/06PMY6gtrz4
<i>APM2020 – 1GrupoDiscos</i>	https://youtu.be/WZ6ulFKisFs
<i>APM2020 – 1GrupoEnsamble</i>	https://youtu.be/pn4KRDMCw1k
<i>APM2020 – 1GrupoAcabados</i>	https://youtu.be/RRyomkZHSWY

5

Conclusiones

En este capítulo se presentan las conclusiones del trabajo realizado, además de los retos que deberán ser resueltos para continuar avanzando en el desarrollo y validación de sistemas ciberfísicos.

5.1. Conclusiones

En este documento se presentan las pautas para implementar sistemáticamente la validación y verificación de un sistema de manufactura virtual bajo control discreto con funcionalidades ciberfísicas. Para su aplicación, se propusieron 3 módulos que constituyen un sistema de manufactura virtual considerando criterios de integración y modularidad representando un sistema de producción real. Inicialmente se mostraron las metodologías utilizadas para el desarrollo del proyecto, buscando estandarizar y crear unas guías para la puesta en marcha virtual pensadas desde el criterio de modelado en un software de diseño asistido por computador, factores importantes en las comunicaciones industriales y conceptos de programación teniendo en cuenta modos seguros de operación.

Se mostraron los procedimientos, herramientas y configuraciones requeridas para concebir un sistema ciberfísico donde se lleva a un entorno virtual el sistema propuesto incorporando sensores y actuadores. Se evidenció la relación entre un modelo o gemelo digital de un sistema de manufactura y su programa de automatización. Se mostraron los elementos a tener en cuenta en el proceso de validación virtual de sensores, actuadores y tareas como llenado y empacado al vuelo. El tipo de consideraciones de los cuerpos que deben ser asignadas de acuerdo con su funcionalidad y las limitaciones en aquellos que requieran un análisis dinámico durante la simulación.

Se describieron las configuraciones y arquitectura de comunicaciones industriales buscando obtener una flexibilidad y capacidad de comunicaciones que nos permitiera llegar a la mayoría de los niveles de la pirámide de automatización. En el documento se describen las configuraciones de tipo OPC DA y OPC UA, esta última una de las más utilizadas recientemente por el sector industrial de automatización permitiendo conectar dispositivos y servidores no solamente en una red local sino también en la nube. Con esto, se propone una arquitectura de comunicaciones para validar y verificar un sistema de manufactura virtual bajo control discreto con funcionalidades ciberfísicas

Se desarrolló igualmente la programación en un autómata industrial virtual el control discreto del sistema de manufactura virtual, aplicando la guía GEMMA (modos seguros de arranque, marcha y parada), donde se abordó inicialmente las funciones de operación descritas desde los GRAFCET nivel 1, de cada módulo hasta las consideraciones de control de movimiento coordinado para los equipos que requerían sincronización. Se propusieron 3 modos, los cuales son los más utilizados por su funcionalidad al momento de automatizar un proceso de manufactura, para ello se realizaron unos esquemas simplificados de cada uno de estos modos y se aplicaron en la ejecución del código de programación. Finalmente, se incorporó una herramienta de supervisión que permitió desarrollar una interfaz y compartir datos en un servidor dando la capacidad de conectar y dar el paso a la industria 4.0, en particular a los sistemas ciberfísicos.

Al incorporar estas herramientas se observa que se puede tener una visión muy completa de todos los procesos involucrados en la fase de diseño de sistemas automáticos. Partiendo desde el diseño mecatrónico con una herramienta de software de modelado 3D abarcando el ciclo de vida del producto PLM, que permite diseñar los componentes del sistema al incorporar soluciones de modelado y herramientas de ingeniería asistida por computador. En el campo de la programación se cuenta con una herramienta que permite crear ejes virtuales y validar coordinación de movimiento, además de validar el programa en un autómata industrial virtual.

5.2. Trabajo futuro

La metodología explorada sumada a los gemelos digitales, habilita a los desarrolladores nuevas herramientas para probar escenarios en un ambiente seguro y factible de validación tanto del diseño del concepto o el sistema, como los programas bajo ciertas condiciones. El enfoque presentado provee una forma sistemática de validar y verificar sistemas de manufactura en un ambiente virtual con funcionalidades ciberfísicas. Sin embargo, existen áreas que necesitan ser exploradas y desarrolladas con más funcionalidades.

Un elemento para profundizar es el desarrollo de interfaces humano máquina para abarcar no solamente operaciones que afecten la producción directamente, sino ampliadas pensando desde la perspectiva de mantenimiento y condiciones que se presentan en la guía GEMMA y de esta manera tener una guía sistemática y estructurada del desarrollo de las mismas.

La adición e interacción con un sistema MES (Manufacturing execution system), brindará una nueva capacidad de planificar y asignar tareas pensando desde la producción global y los indicadores que esta pueda obtener desde una fase temprana reduciendo las incertidumbres al momento de implementarla. De esta manera se seguirá escalando en la pirámide de la automatización llegando cada vez mas a etapas de gestión.

Bibliografía

- [1] Hoffmann, P., Stadtweg, R., and Grove, F. (2010). Virtual Commissioning Of Manufacturing Systems A Review And New Approaches For Simplification. In Proceedings of the 24th European Conference on Modelling and Simulation (ECMS 2010),At: Kuala Lumpur, Malaysia (pp. 175–181).
- [2] Auinger, F., Buchtela, G. (1999). Interface driven domain-independent modeling architecture for soft-commissioning and reality in the loop. In Proceedings of the 1999 Winter Simulation Conference P. (pp. 798–805).
- [3] Puntel-Schmidt, P., Fay, A. (2015). Levels of detail and appropriate model types for virtual commissioning in manufacturing engineering. IFAC-PapersOnLine, 28(1), 922–927. <https://doi.org/10.1016/j.ifacol.2015.05.027>
- [4] Hoffmann, P., Schuman, R., M.A., T. M., Premier, G. C. (2012). Research on simplified modelling strategy for virtual commissioning. In Proceedings of the European Modelling and Simulation Symposium (pp. 293–302). Proceedings of the European Modelling and Simulation Symposium.
- [5] Vermaak, H. (2017). Virtual Commissioning : A Tool to Ensure Effective System Integration. In 2017 IEEE International Workshop of Electronics, Control, Measurement, Signals and their Application to Mechatronics (ECMSM). <https://doi.org/10.1109/ECMSM.2017.7945899>
- [6] Schamp, M., Hoedt, S., Claeys, A., Aghezzaf, E. H., Cottyn, J. (2018). Impact of a virtual twin on commissioning time and quality. In IFAC-PapersOnLine (Vol. 51, pp. 1047–1052). Elsevier B.V. <https://doi.org/10.1016/j.ifacol.2018.08.469>.
- [7] Khan, A., Falkman, P., Fabian, M. (2017). Virtual Engineering Framework for Automatic Generation of Control Logic including Safety. In 13th IEEE Conference on Automation Science and Engineering (CASE) Xi'an, China.
- [8] Alvarez, M. L., Burgos, A., Sarachaga, I., Estévez, E., Marcos, M. (2012). GEMMA based approach for generating PLCopen Automation projects. IFAC Conference on Embedded Systems, Computational Intelligence and Telematics in Control (Vol. 45, pp. 230–235).

- [9] Binnberg, D., Johansson, V., Larsson, M., Berntsson, S., Ayani, M. (2016). University Diploma Project in automation technology 22.5 ECTS Spring term Year. University of Skovde.
- [10] Univerisdad de Oviedo. (2019, March 15). A - Procedimiento de parada. Retrieved from http://isa.uniovi.es/docencia/iea/teoria/gemma_resumen.pdf
- [11] Vielhaber, M., Siadat, A. (2018). Virtual validation of Design decentrally controlled manufacturing systems with cyber-physical functionalities. *Procedia CIRP*, 72, 509–514. <https://doi.org/10.1016/j.procir.2018.03.195>
- [12] Armendia, M., Ghassempouri, M., Ozturk, E., Peysson, F. (2019). *Twin-Control: A Digital Twin Approach to Improve Machine Tools Lifecycle*. (M. Armendia, M. E. O. Ghassempouri, F. Peysson, Eds.). Gothenburg, Sweden: Springer.
- [13] Älegård, S., Knutsson, S. (2017). *Virtual Commissioning of Smart Factory*. Master Thesis. Chalmers University of Technology.
- [14] Konopa, M. (2013). *Simulation of Production Processes*. Master Thesis. Czech Technical University.
- [15] Jovanny, D. (2014). Structured design of automatic systems: Applying the GEMMA/SFC approach to a mechatronics teaching system. In *3rd International Congress of Engineering Mechatronics and Automation, CIIMA 2014* (pp. 1–5). <https://doi.org/10.1109/CIIMA.2014.6983449>
- [16] Deloitte. (2019, April 8). *Industry 4.0 Challenges and solutions for the digital transformation and use of exponential technologies*. <https://doi.org/10.4324/9781315300412-3>
- [17] Guerrero, R. L. Y. V. (2017). *Autómatas Programables SIEMENS GRAFCET y Guía GEMMA con TIA Portal*. (SMC, Ed.) (1st ed.). Barcelona: Marcombo.
- [18] Josep Balcells, J. L. R. (1997). *Autómatas Programables* (1st ed.). Barcelona: Marcombo.
- [19] Kazmi, S. M. A. (2019). *Methodology For Validating Mechatronic Digital Twin*. Master Thesis. Tampere University.
- [20] Hoernicke, M., Greifeneder, J. (2011). *Next generation factory acceptance test*. ABB Corporate Research Center Germany - Annual Report 2011, (April 2012).
- [21] Runde, C. (2018). *Whitepaper Virtual Acceptance in the field of Mechanical Engineering Virtual Dimension Center (VDC) Fellbach*, (August). <https://doi.org/10.13140/RG.2.2.16553.65122>

A

Anexo: Grafets a nivel automata del sistema de manufactura

A. Anexo: Graficets a nivel automata del sistema de manufactura

SFC_GEMMA - Sequential Function Chart

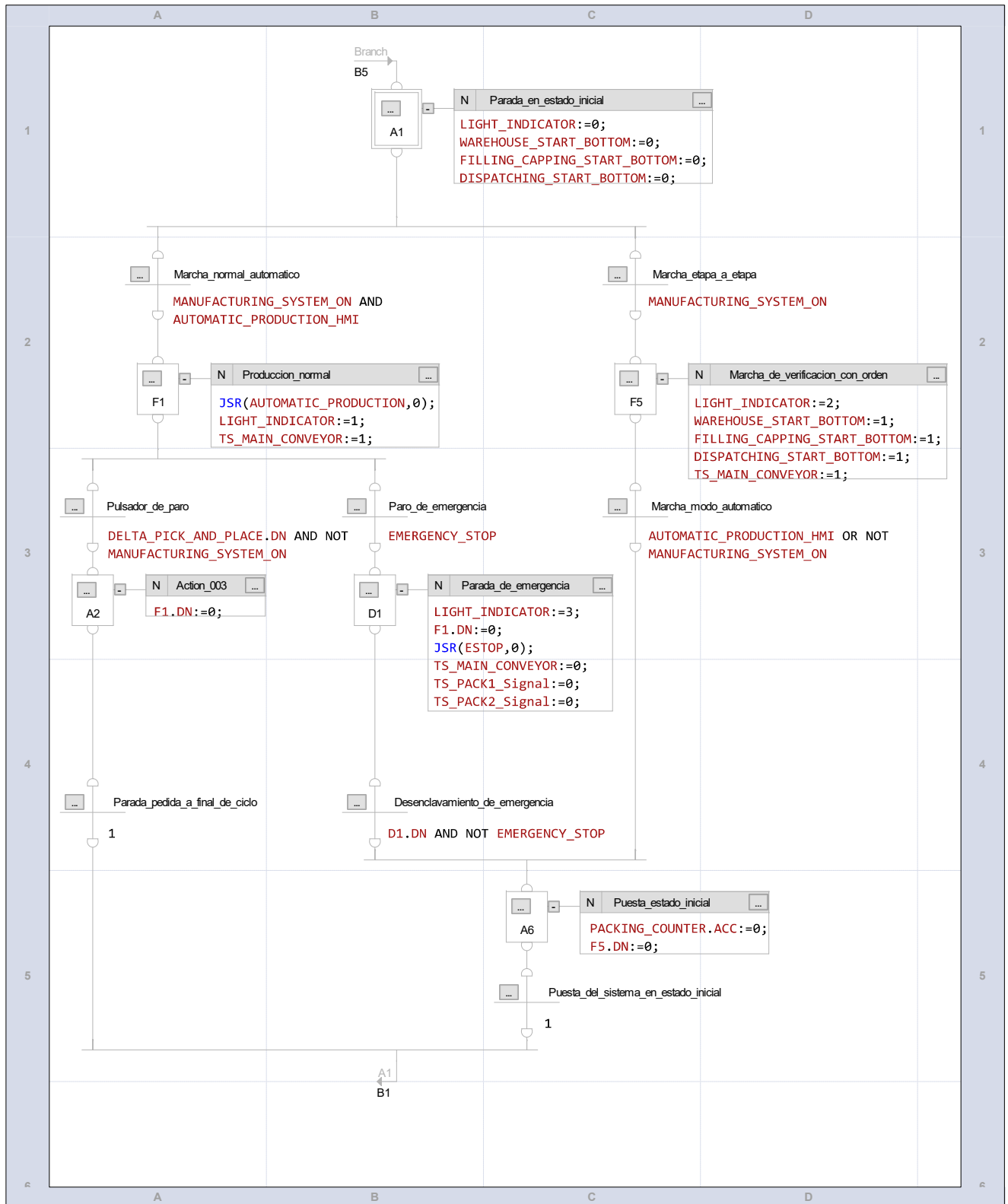
SMV40:MainTask:MainProgram

1 of 1 total sheets in routine

Page 1

15/07/2020 12:05:55 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



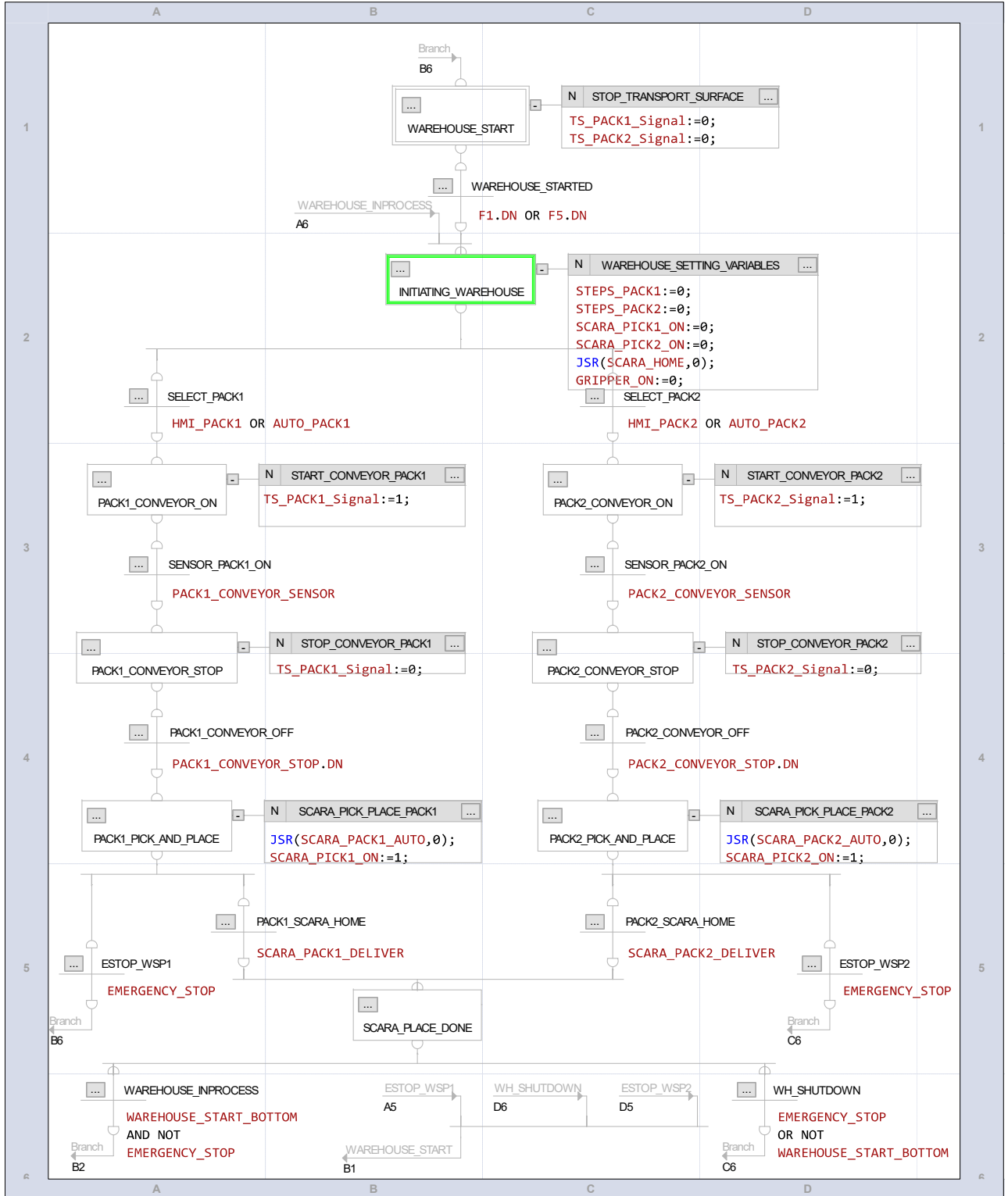
SFC_WAREHOUSE_STATION - Sequential Function Chart

SMV40:MainTask:MainProgram
1 of 4 total sheets in routine

15/07/2020 12:14:08 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD

Main Module Function Chart

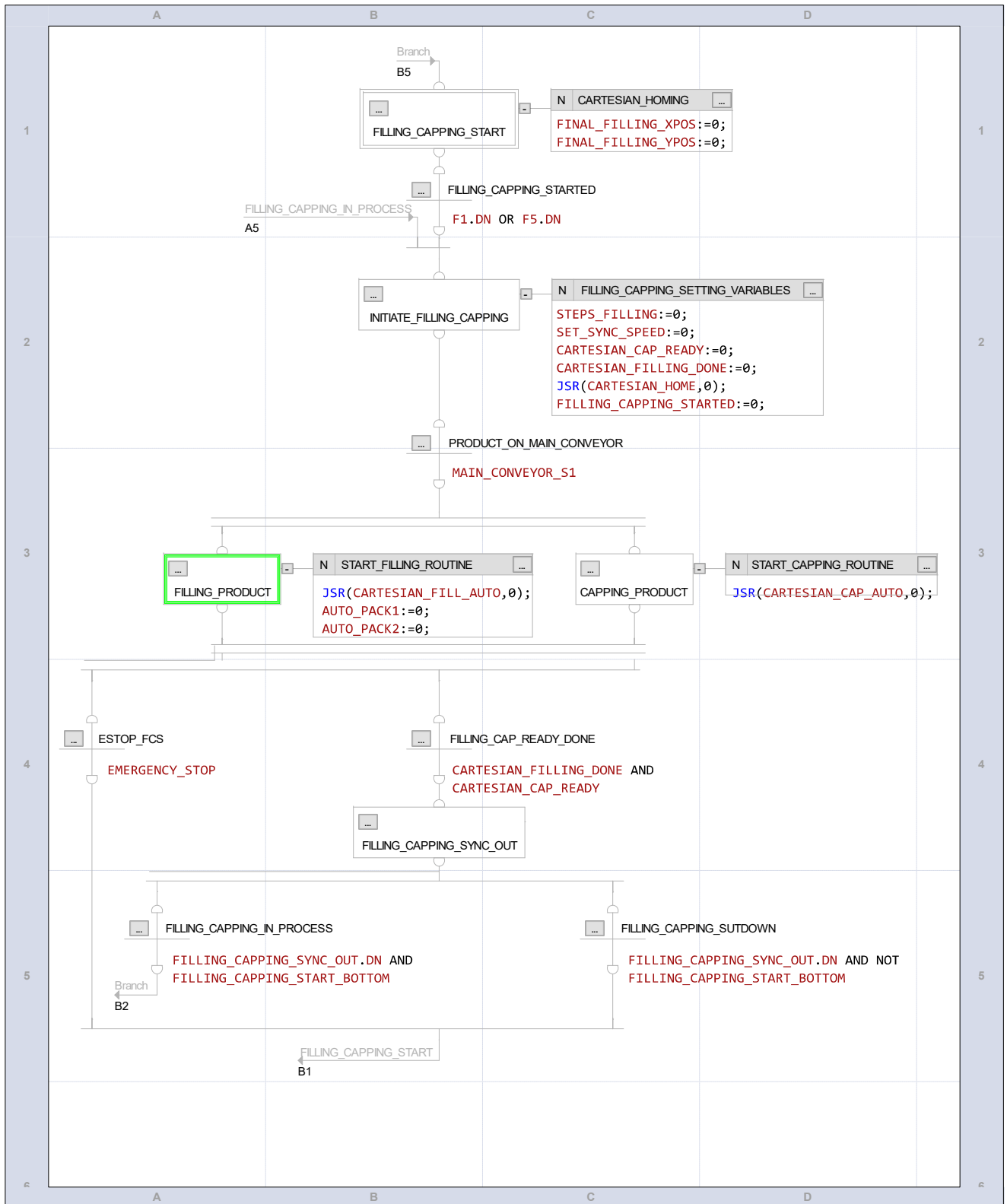


SFC_FILLING_CAPPING_STATION - Sequential Function Chart

SMV40:MainTask:MainProgram
1 of 1 total sheets in routine

15/07/2020 12:15:05 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



A. Anexo: Graficets a nivel automata del sistema de manufactura

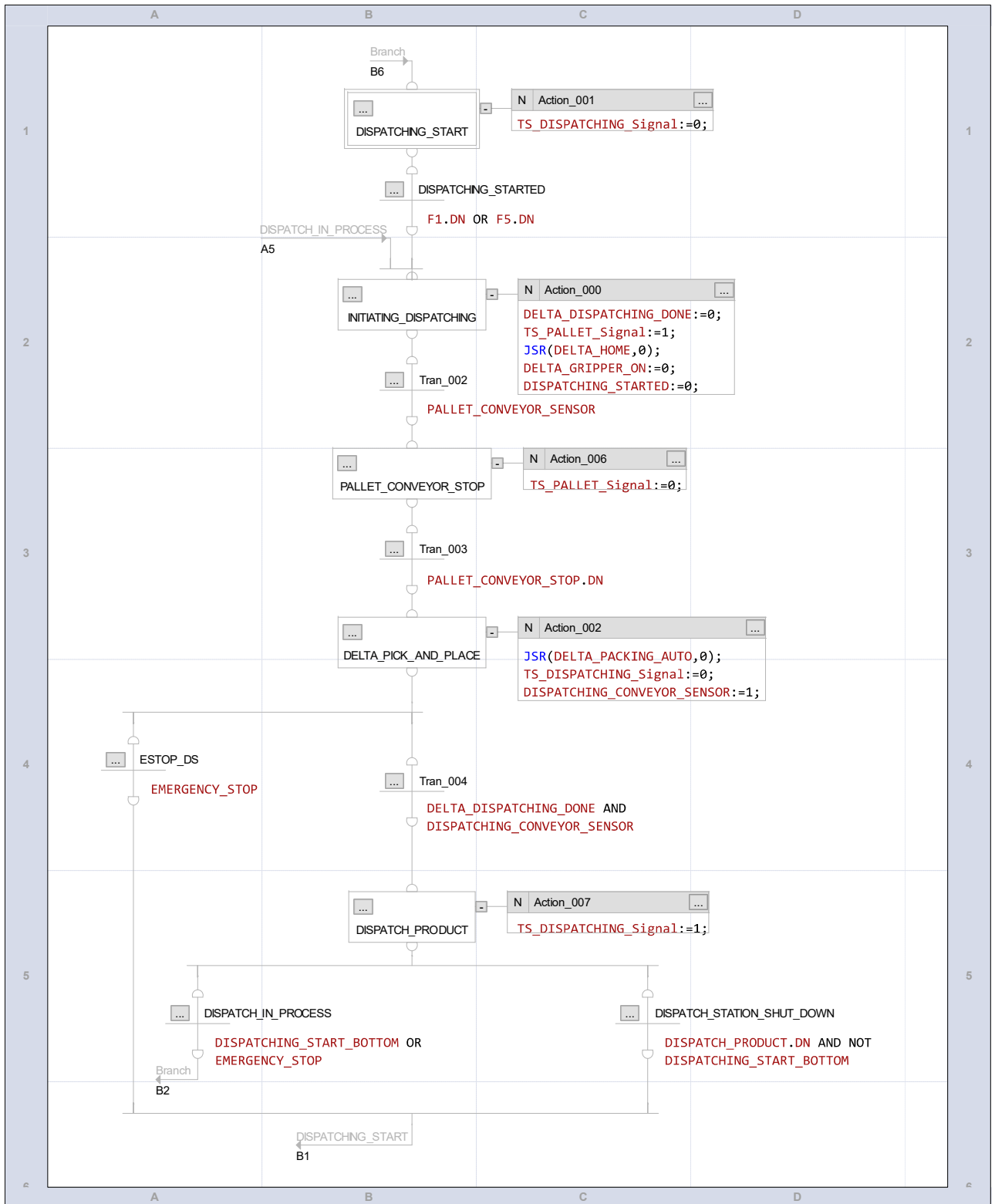
SFC_DISPATCHING_STATION - Sequential Function Chart

SMV40:MainTask:MainProgram
1 of 2 total sheets in routine

Page 1


15/07/2020 12:17:44 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



B

Anexo: Guía validación y verificación virtual - Física Básica

Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá	 UNIVERSIDAD NACIONAL DE COLOMBIA
---	---	---

DEPARTAMENTO DE INGENIERÍA MECÁNICA Y MECATRÓNICA

AUTOMATIZACION DE PROCESOS DE MANUFACTURA	2017280-2
SERVOMECANISMOS	2017288-1
MOTORES Y CONTROLADORES INDUSTRIALES	2027978-1

Desarrollado por:	Ubaldo Garcia Zaragoza E-mail: ugarciaz@unal.edu.co , Oficina: Edificio 411, Laboratorio 101
Docentes:	Víctor Hugo Grisales Palacio (Coordinador de Laboratorio) E-mail: vhgrisalesp@unal.edu.co , Oficina: Edificio 411, Laboratorio 101 Eduardo Barrera Gualdrón E-mail: ebarrerag@unal.edu.co , Oficina: Edificio 411, Oficina 204
Horario:	Acorde al espacio académico (Verificar horario asignado)
Salón:	Edificio 411-101 Laboratorio de Automatización

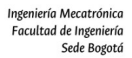

Guía para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas #1 – Física Básica.

1. CONTEXTO

Desde el año 1999, se ha tratado de definir estrategias de validación virtual que puedan brindar beneficios a la industria de la automatización, aunque en esas épocas se carecía de recursos informáticos y de hardware la comunidad científica fue capaz de probar los beneficios de este enfoque, logrando clasificarlos en 4 categorías principalmente [1]:

- **Validación Tradicional:** la cual consta de pruebas físicas del sistema de manufactura automático utilizando controladores físicos sin asistencia de entornos virtuales.
- **Validación Virtual** (o Software-in-the-loop): la cual propone verificar la respuesta tanto del controlador como el sistema de manufactura automático de forma virtual en una simulación.
- **Reality in the loop:** la cual emplea un controlador virtual para ver la respuesta del sistema de manufactura automático físico.
- **Hardware-in-the-loop** (o Soft Commissioning): la cual emplea un modelo virtual del sistema de manufactura automático para probar controladores físicos.

Para virtualizar un sistema de manufactura automático es necesario determinar el nivel de detalle necesario para construir el modelo virtual. Este podrá ser dividido en dos grupos principales [2]: bajo nivel de detalle y alto nivel de detalle. Inicialmente se propone utilizar un nivel de detalle bajo, pero con la posibilidad de intercambiar posteriormente el modelo 3D por un modelo detallado. El modelo detallado se debe preparar para la validación virtual teniendo en cuenta los siguientes aspectos: 1) La complejidad debe ser reducida considerando el consumo de renderizado y visualización y 2) Se debe considerar las dependencias entre los objetos móviles y estáticos teniendo como principal enfoque actuadores y sensores.

Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	 
---	---

2. DESCRIPCIÓN:

Esta guía permitirá que el participante identifique, seleccione y configure la física básica de los componentes relevantes para la simulación de un sistema de manufactura virtual. Logrando comprender la interacción entre componentes acorde a su funcionalidad, además de las implicaciones de confiabilidad y desempeño de la simulación debido a la configuración asignada.

En esta guía se mostrará la configuración de algunas instrucciones tales como: cuerpos rígidos, cuerpos de colisión, precisión de la colisión y categoría.

Se debe tener en cuenta que el desarrollo de la totalidad de la guía estará orientada a configurar un sistema de manufactura virtual donde se mostrará las siguientes configuraciones.

- Identificar los componentes relevantes para la simulación del sistema de manufactura virtual.
- Seleccionar y clasificar los componentes dependiendo la interacción y su funcionalidad.
- Configurar los componentes rígidos y de colisión del sistema de manufactura virtual.
- Asignar la configuración de colisión para garantizar confiabilidad y desempeño en la simulación.

3. OBJETIVOS DE LA PRACTICA:

- 3.1 Identificar los componentes relevantes de un Sistema de Manufactura Virtual.
- 3.2 Configurar la física de los elementos a interactuar de forma virtual de un Sistema de Manufactura.

4. RECURSOS PARA LA PRACTICA:

- 4.1 Estación de trabajo PC.
- 4.2 Software PLM NX Mechatronics Concept Designer.

5. TIEMPO SUGERIDO PARA EL DESARROLLO DE LA PRACTICA: 1 día.


6. PRACTICA A REALIZAR:

- 6.1 Leer la secuencia de configuración que se encuentra en el contenido de esta Práctica.
- 6.2 Garantizar licencia en su estación de trabajo del software Siemens NX Mechatronics Concept Designer.
Nota: Verificar la conexión con el servidor de licencias, mediante ping a la IP 168.176.36.140.
- 6.3 Abrir el archivo 3D 03CPS_PICK_AND_PLACE_V0 en el entorno de NX Mechatronics Concept Designer.
- 6.4 Usar el software NX MCD para configurar la física de los componentes a interactuar en la simulación.

7. ENTREGA DE INFORME:

Cada grupo realizara un informe en formato libre de la práctica realizada, que básicamente corresponde a documentar la parte 6 de este documento, respondiendo a la siguiente interrogante ¿Qué aprendizajes le deja esta práctica y en que contextos lo puede aplicar? Se sugiere ayudarse de registros multimedia que faciliten la lectura del informe.

El informe se debe subir al Moodle en formato PDF de acuerdo con la programación del curso.

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	 <p style="font-size: small;">Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p> <p style="font-size: small;">UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	---

6.2 LICENCIAMIENTO NX MCD:

Lo primero que se tiene que hacer es verificar la comunicación física entre la estación de trabajo PC y el gestor de licencias de NX. Verifique que el cable de red efectivamente está conectado en el PC y que se encuentra conectado a la red UNAL. Posteriormente podemos realizar un ping, para verificar la conexión física entre el PC y el servidor de licencia de NX:

```

Microsoft Windows [Versión 10.0.17134.765]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

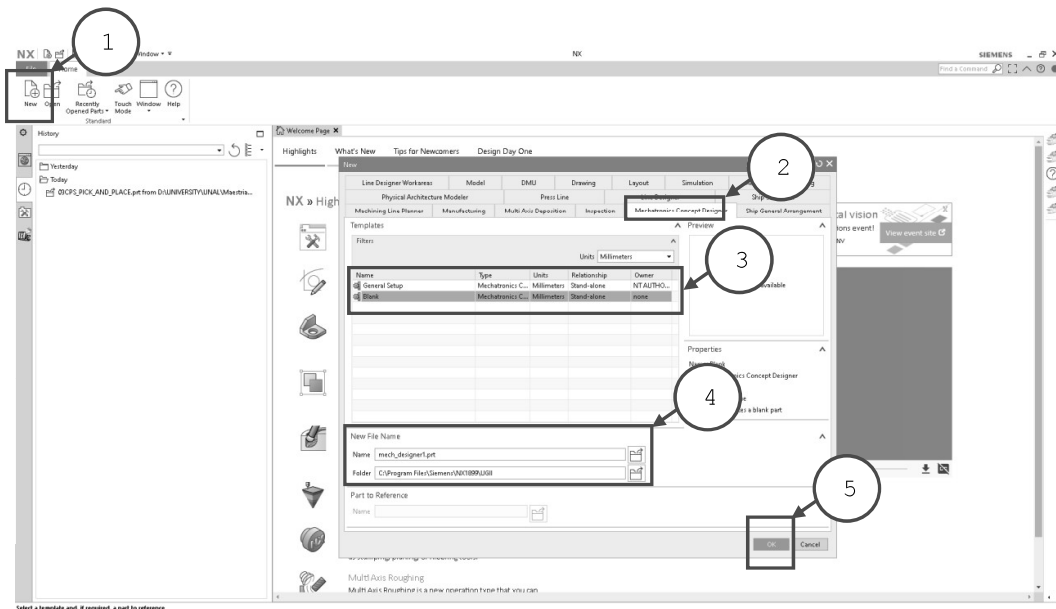
C:\Users\WS-Innovation>ping 168.176.36.140

Haciendo ping a 168.176.36.140 con 32 bytes de datos:
Respuesta desde 168.176.36.140: bytes=32 tiempo=9ms TTL=126
Respuesta desde 168.176.36.140: bytes=32 tiempo=11ms TTL=126
Respuesta desde 168.176.36.140: bytes=32 tiempo=313ms TTL=126
Respuesta desde 168.176.36.140: bytes=32 tiempo=5ms TTL=126


Estadísticas de ping para 168.176.36.140:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
              (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 5ms, Máximo = 313ms, Media = 84ms
    
```

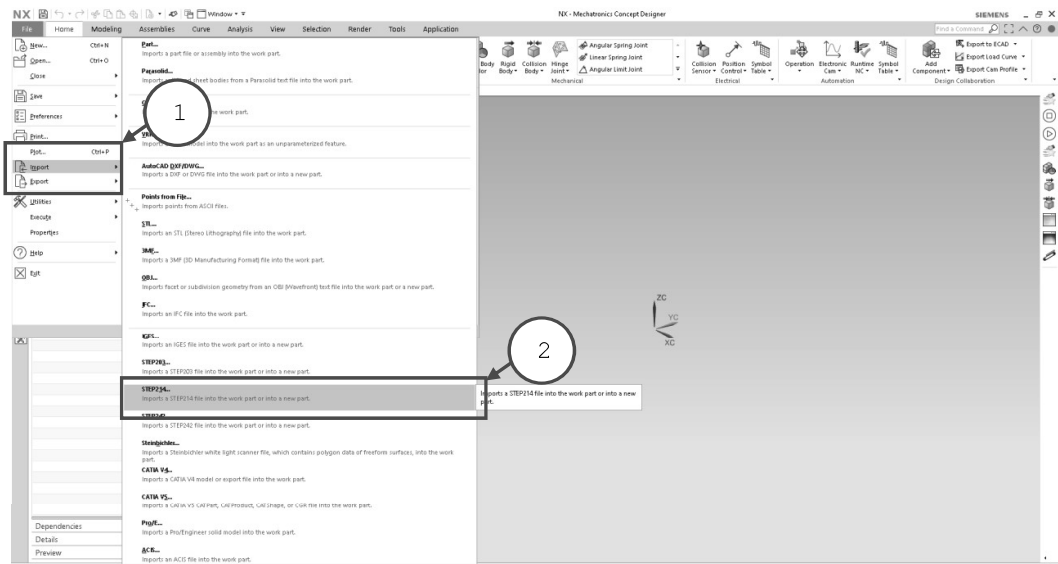
6.3 ABRIR EL ARCHIVO 3D 03CPS_PICK_AND_PLACE_V0 EN NX MCD:

Lo primero que se tiene que hacer es crear un archivo nuevo (1), seleccionar la pestaña Mechatronics Concept Designer (2), seleccionar entre las plantillas disponible una en blanco (3), asignar el nombre y ruta del archivo (4) y crear el nuevo archivo (5):

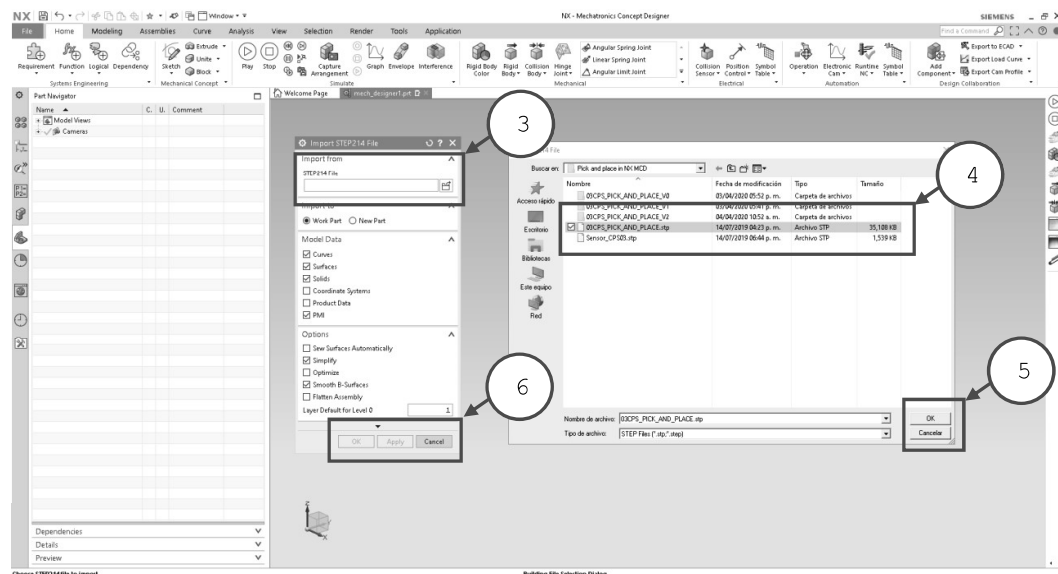


Posteriormente es necesario agregar o importar los modelos CAD 3D del sistema de manufactura virtual, en este caso importaremos el archivo en formato STEP el cual fue desarrollado en otro software de CAD.


Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá	 UNIVERSIDAD NACIONAL DE COLOMBIA
--	---	---



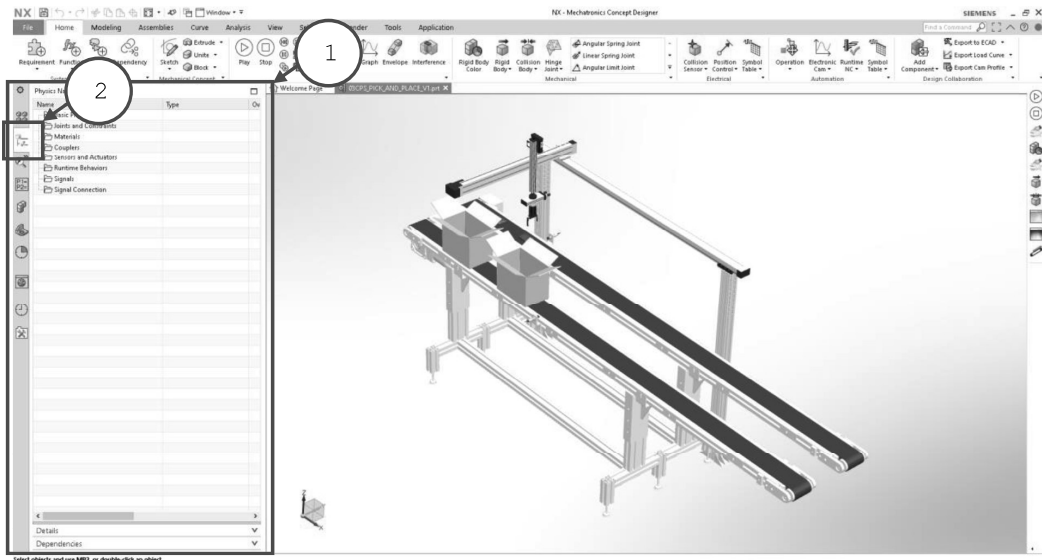
Es importante ubicar la ruta donde se encuentra el archivo (3-4) en el caso de realizar la importación del modelo, también el software NX habilita en la pestaña de modelado la posibilidad de crear componentes en formato nativo de esta plataforma. Una vez concluidos los pasos aceptar el proceso (5-6) y esperar a que se realice la importación del modelo.



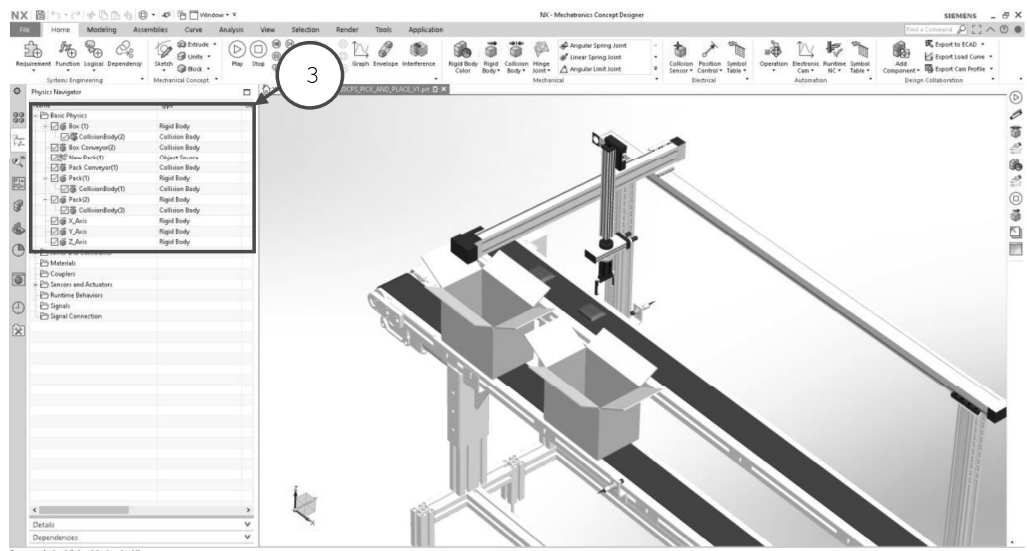
Una vez en el entorno de NX MCD es necesario familiarizarse con en el panel de navegación al costado izquierdo, el cual será de gran utilidad para configurar los parámetros de la simulación virtual del sistema de manufactura.

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>  <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--


6.4 USAR EL SOFTWARE NX MCD Y CONFIGURAR LA FÍSICA DE LOS COMPONENTES

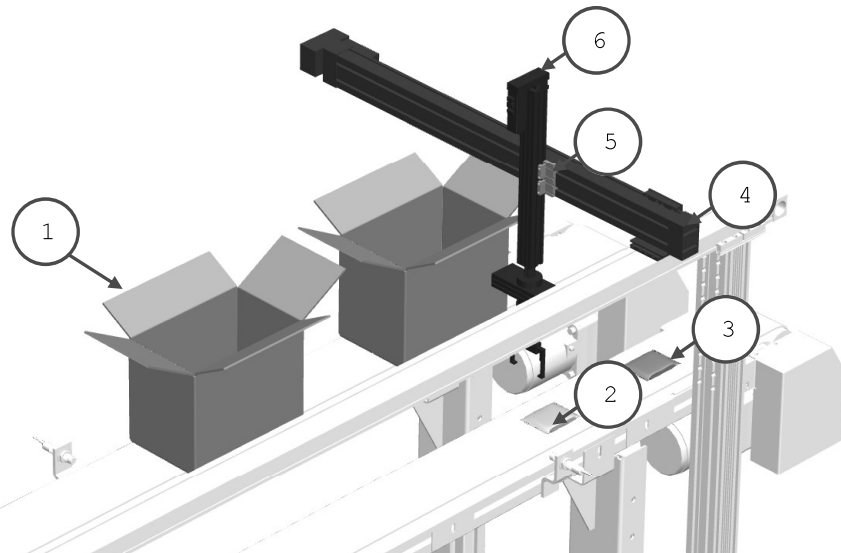


En esta guía nos enfocaremos en el panel de navegación (1) de física (2) en la cual encontrara varias carpetas, comenzando por la de física básica, uniones y restricciones, materiales, sensores y actuadores, señales, etc.

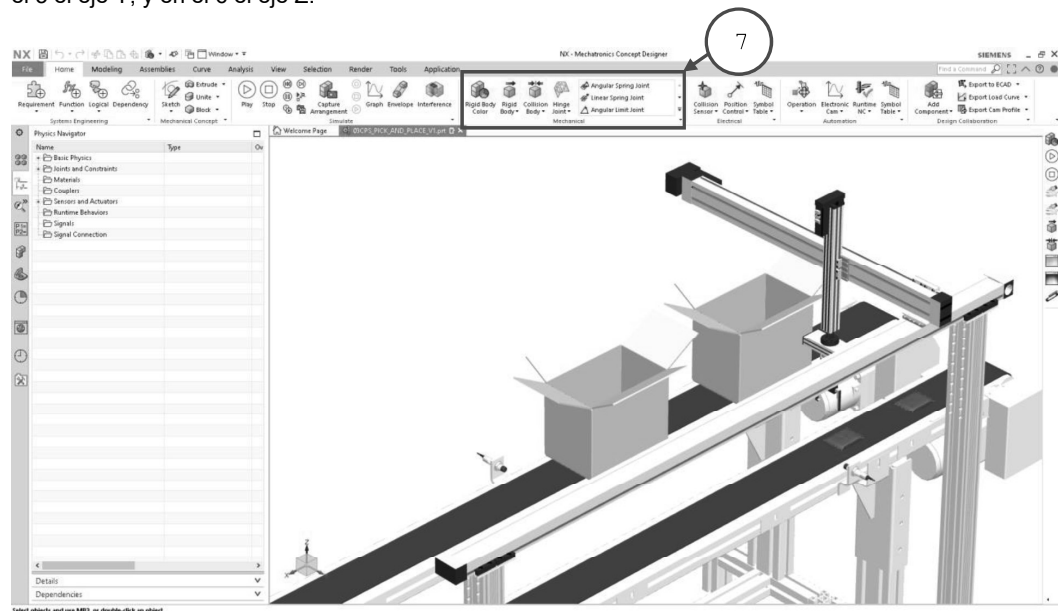


A continuación, nos enfocaremos en configurar los elementos que interactuaran en nuestra simulación, para ello identificaremos los elementos estáticos y móviles que se encuentran en nuestro sistema de manufactura virtual.


Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>  <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
---	--

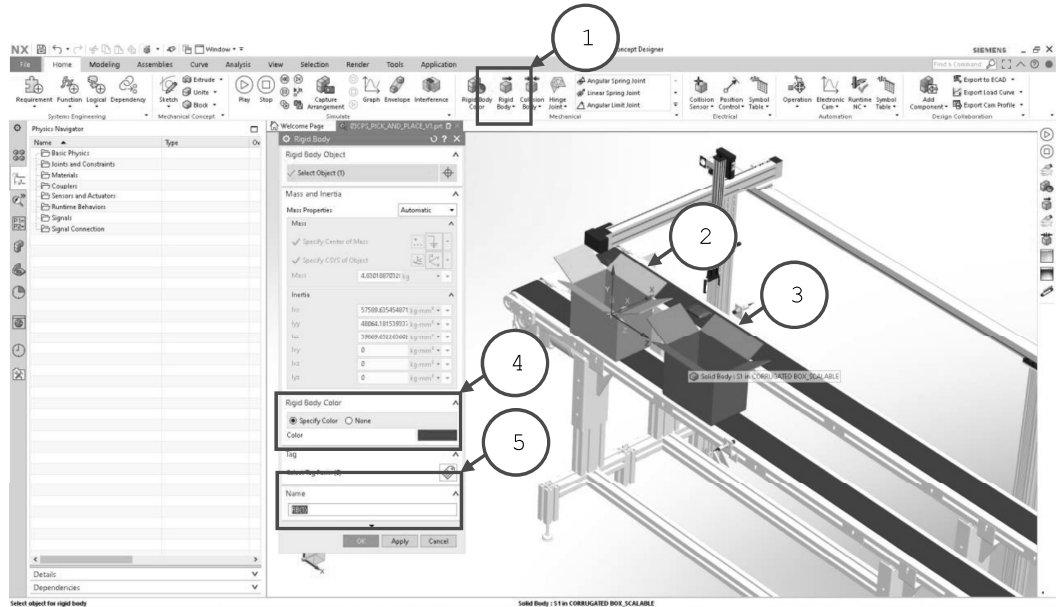


Inicialmente podemos identificar que los elementos en color blanco serán aquellos que se encontrarán de manera estática y aquellos de color serán los móviles dentro de la simulación del sistema de manufactura virtual. En el numero 1 encontramos las cajas, en el 2 y 3 los paquetes, en el 4 el eje X, en el 5 el eje Y, y en el 6 el eje Z.

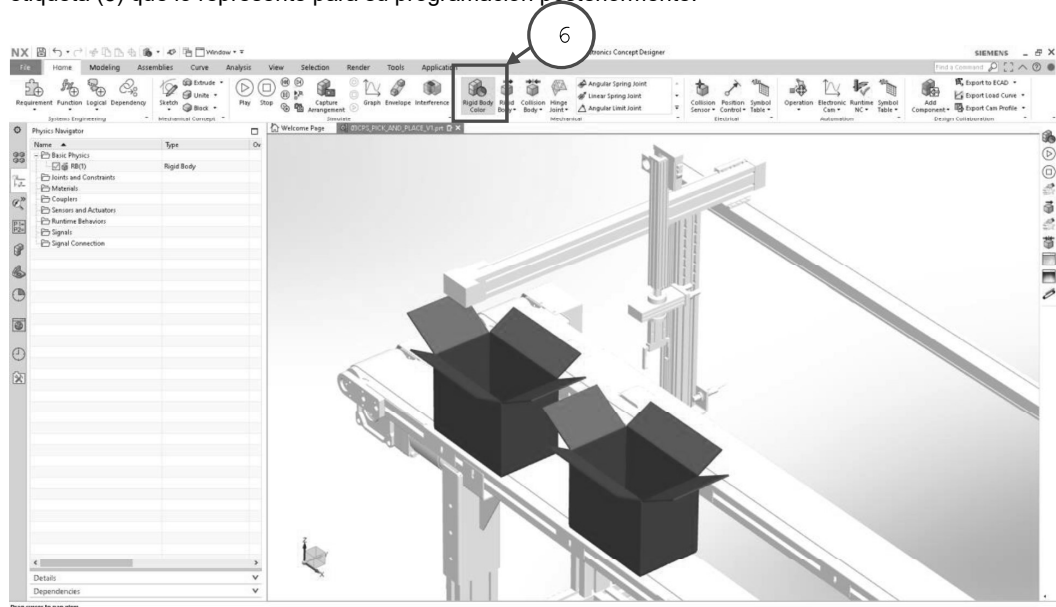


Para configurar cada uno de estos componentes como cuerpos rígidos haremos uso del grupo de mecánica (7) de la pestaña inicio en el módulo de NX MCD.


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>  <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--

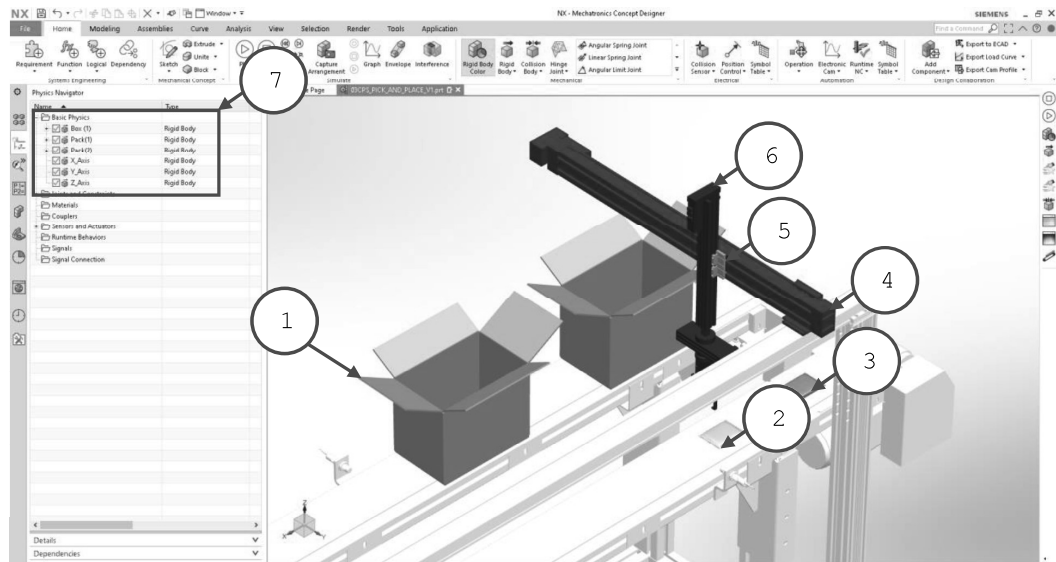


Iniciaremos configurando los cuerpos rígidos mediante el uso del comando (1) en el grupo de mecánica, seleccionando inicialmente cada elemento(s) del modelo virtual del sistema de manufactura (2-3), posteriormente configuraremos un color (4) que lo identifique y finalmente asignaremos un nombre o etiqueta (5) que lo represente para su programación posteriormente.

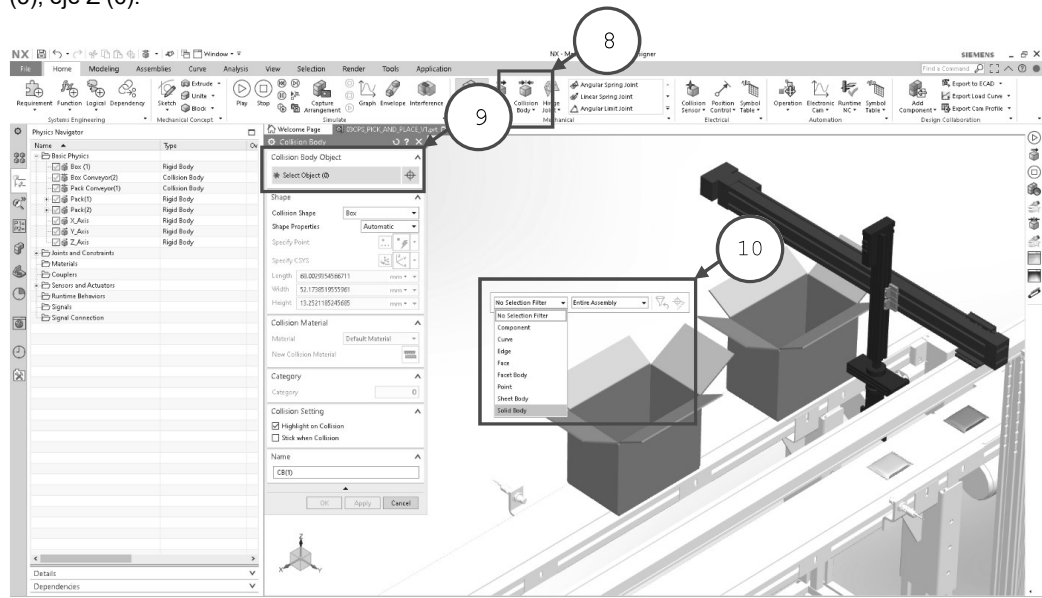


Mediante el comando color (6) en el grupo de mecánica es posible ir revisando las configuraciones asignadas a cada cuerpo para identificar los cuerpos móviles de los estáticos en color blanco.


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>  <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--

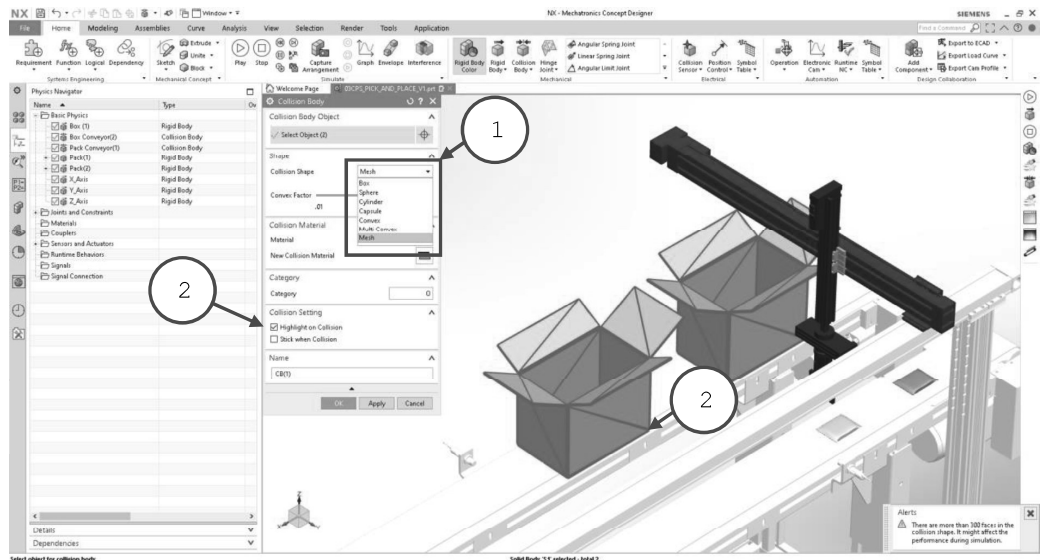


Realizar la configuración de cuerpo rígido para los elementos: cajas (1), paquetes (2-3), eje X (4), eje Y (5), eje Z (6).



Una vez configurados los cuerpos rígidos, procederemos a configurar los elementos que colisionan e interactúan en el movimiento, mediante el comando cuerpo de colisión (8) y seleccionar los cuerpos sólidos (9) activando el filtro de selección (10) con clic derecho en la ventana grafica.

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	 <p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p> <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	---



Para configurar adecuadamente los cuerpos de colisión es necesario definir la forma en que serán procesados ya que de esto dependerá el desempeño y la calidad de la simulación (1). Para visualizar la forma es necesario activar la casilla en opciones de colisión (2) la cual aparecerá en la ventana gráfica.


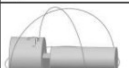





En NX Mechatronics Concept Designer es posible calcular las colisiones usando formas simplificadas de los cuerpos a interactuar.


Use un cuerpo de colisión para definir cómo una pieza de geometría colisionará con otros elementos que también tienen un cuerpo de colisión. Los objetos sin cuerpo de colisión pasan a través de otros objetos. Además, puede adjuntar un cuerpo de colisión a otros cuerpos de colisión durante las simulaciones.

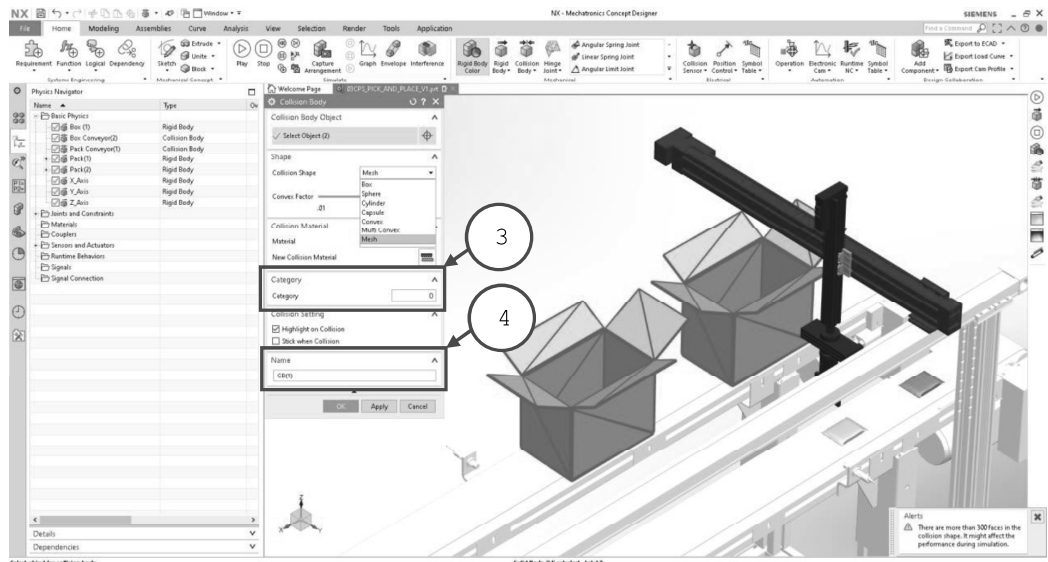
Al incrementar la precisión de la geometría y acercarse más a su forma original del cuerpo sólido es más propensa a generar fallas de simulación y reducir el desempeño de la simulación.

Es recomendado usar formas simplificadas de colisión en lo mayor posible para reducir el riesgo de inestabilidad en la simulación (atravesándose, pegándose, oscilando) y maximizar el desempeño.

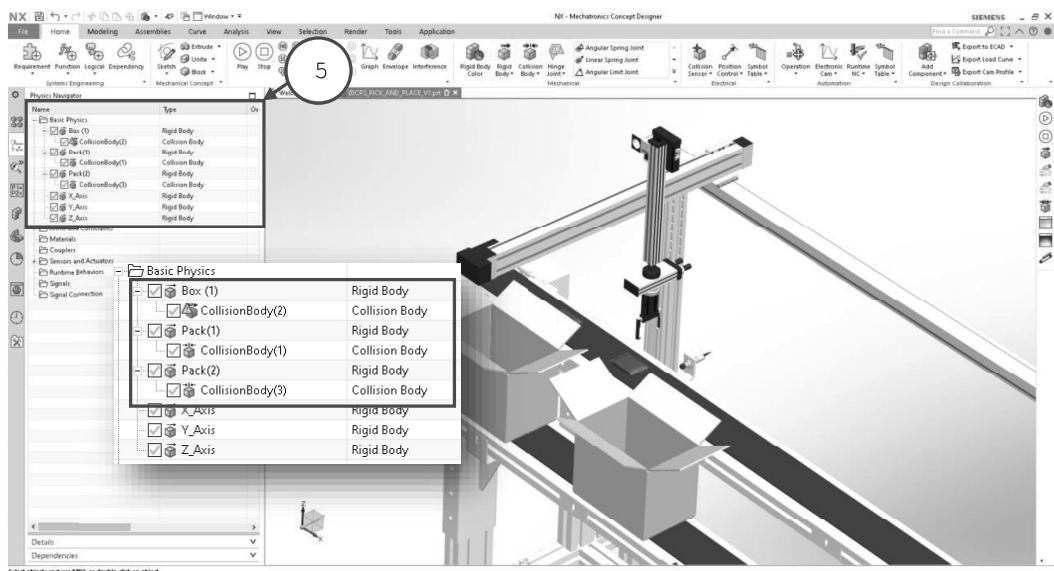
Es posible usar el factor de convección en la barra deslizante en el cuadro de diálogo para ajustar el nivel de detalle de la colisión del cuerpo.

Type	Geometric Accuracy	Reliability	Simulation Performance
 Box	Low	High	High
 Sphere	Low	High	High
 Cylinder	Low	High	High
 Capsule	Low	High	High
 Convex	Medium	High	Medium
 Multi Convex	Medium	High	Medium
 Mesh	High	Low	Low


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>	 <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--	---

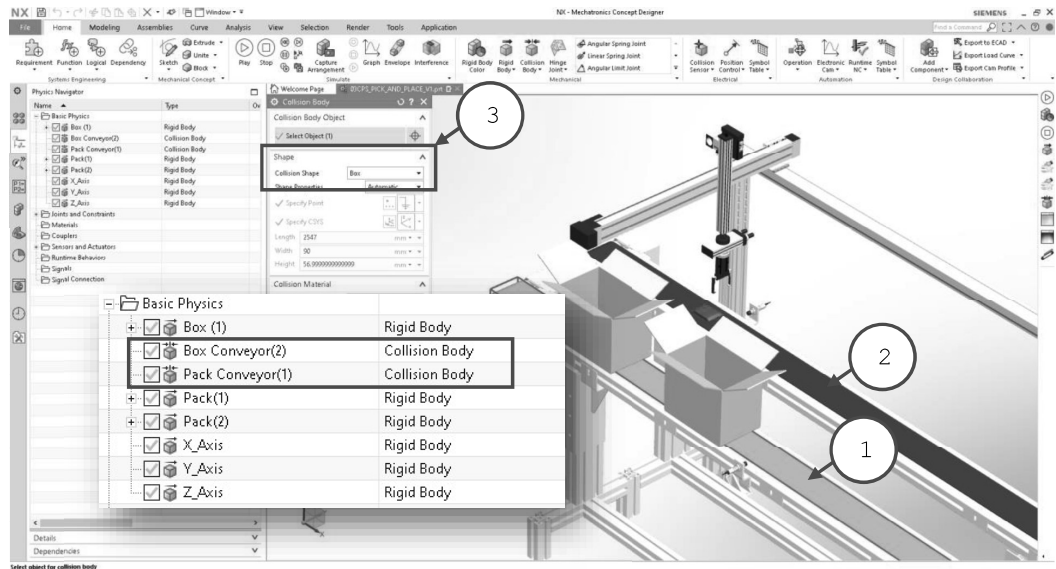


En el cuadro de dialogo de colisión es posible configurar parámetros como el de categoría (2) que es útil para identificar el cuerpo en simulaciones donde el color es importante o el material. No olvidar asignar un nombre (4) fácil de identificar para programar la rutina de movimiento.

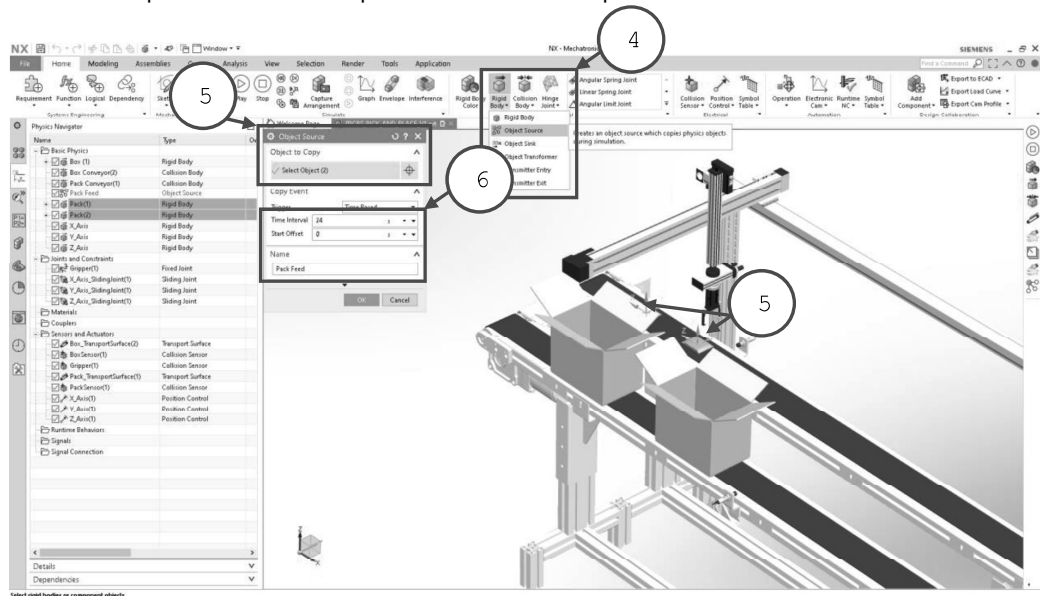


En este punto realizaremos la configuración de los elementos que interactúan en colisión en el sistema de manufactura virtual siendo: las cajas, paquete 1 y paquete 2 utilizando la forma MESH, ya que queremos que la forma de interacción sea la más real posible.

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>  <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--



Otro elemento importante son las bandas transportadoras (1-2), las cuales entraran en contacto con las cajas y los paquetes, por ello realizaremos la configuración como cuerpos de colisión. Seleccionar el cuerpo sólido y utilizar la forma BOX (3) para que su forma sea simplificada y no afecte el desempeño de la simulación para cada banda transportadora de forma independiente.



Finalmente, es posible configurar parámetros adicionales como aparecer y ocultar componentes a un tiempo determinado (6) o bajo una indicación de una señal, mediante los comandos fuente de objetos (4) y sumidero de objetos. Asignar un nombre para ser utilizado posteriormente en la programación.

C

Anexo: Guía validación y verificación virtual - Automatización

Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas

Ingeniería Mecatrónica
Facultad de Ingeniería
Sede Bogotá



DEPARTAMENTO DE INGENIERÍA MECÁNICA Y MECATRÓNICA

AUTOMATIZACION DE PROCESOS DE MANUFACTURA	2017280-2
SERVOMECANISMOS	2017288-1
MOTORES Y CONTROLADORES INDUSTRIALES	2027978-1

Desarrollado por:	Ubaldo Garcia Zaragoza E-mail: ugarciaz@unal.edu.co , Oficina: Edificio 411, Laboratorio 101
Docentes:	Víctor Hugo Grisales Palacio (Coordinador de Laboratorio) E-mail: vhgrisalesp@unal.edu.co , Oficina: Edificio 411, Laboratorio 101 Eduardo Barrera Gualdrón E-mail: ebarrerag@unal.edu.co , Oficina: Edificio 411, Oficina 204
Horario:	Acorde al espacio académico (Verificar horario asignado)
Salón:	Edificio 411-101 Laboratorio de Automatización


Guía para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas #2 – Actuadores y sensores.

1. CONTEXTO

Desde el año 1999, se ha tratado de definir estrategias de validación virtual que puedan brindar beneficios a la industria de la automatización, aunque en esas épocas se carecía de recursos informáticos y de hardware la comunidad científica fue capaz de probar los beneficios de este enfoque, logrando clasificarlos en 4 categorías principalmente [1]:

- **Validación Tradicional:** la cual consta de pruebas físicas del sistema de manufactura automático utilizando controladores físicos sin asistencia de entornos virtuales.
- **Validación Virtual** (o Software-in-the-loop): la cual propone verificar la respuesta tanto del controlador como el sistema de manufactura automático de forma virtual en una simulación.
- **Reality in the loop:** la cual emplea un controlador virtual para ver la respuesta del sistema de manufactura automático físico.
- **Hardware-in-the-loop** (o Soft Commissioning): la cual emplea un modelo virtual del sistema de manufactura automático para probar controladores físicos.

Para virtualizar un sistema de manufactura automático es necesario determinar el nivel de detalle necesario para construir el modelo virtual. Este podrá ser dividido en dos grupos principales [2]: bajo nivel de detalle y alto nivel de detalle. Inicialmente se propone utilizar un nivel de detalle bajo, pero con la posibilidad de intercambiar posteriormente el modelo 3D por un modelo detallado. El modelo detallado se debe preparar para la validación virtual teniendo en cuenta los siguientes aspectos: 1) La complejidad debe ser reducida considerando el consumo de renderizado y visualización y 2) Se debe considerar las dependencias entre los objetos móviles y estáticos teniendo como principal enfoque actuadores y sensores.

Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá	 UNIVERSIDAD NACIONAL DE COLOMBIA
---	---	--

2. DESCRIPCIÓN:

Esta guía permitirá que el participante identifique, seleccione y configure los actuadores y sensores relevantes para la simulación de un sistema de manufactura virtual. Logrando comprender los parámetros principales a tener en cuenta tanto para actuadores como sensores.

En esta guía se mostrará la configuración de algunas instrucciones tales como: juntas prismáticas, juntas fijas, superficie de transporte y sensor de colisión.

Se debe tener en cuenta que el desarrollo de la totalidad de la guía estará orientada a configurar un sistema de manufactura virtual donde se mostrará las siguientes consideraciones:

- Configurar las restricciones y juntas de la cinemática del sistema de manufactura virtual.
- Asignar la configuración de actuadores para garantizar los movimientos en la simulación
- Configurar las superficies de transporte para desplazar producto en la simulación.
- Asignar la configuración de sensores para garantizar las señales necesarias en la simulación.

3. OBJETIVOS DE LA PRACTICA:

- 3.1 Configurar los sistemas de actuación virtuales de un Sistema de Manufactura.
- 3.2 Configurar los sensores virtuales de un Sistema de Manufactura.

4. RECURSOS PARA LA PRACTICA:

- 4.1 Estación de trabajo PC.
- 4.2 Software PLM NX Mechatronics Concept Designer.
- 4.3 Archivo 03CPS_PICK_AND_PLACE previamente configurado con componentes rígidos y de colisión.
Nota: Ver y realizar guía 1, configuración de la física básica.

5. TIEMPO SUGERIDO PARA EL DESARROLLO DE LA PRACTICA: 1 día.


6. PRACTICA A REALIZAR:

- 6.1 Leer la secuencia de configuración que se encuentra en el contenido de esta Práctica.
- 6.2 Garantizar licencia en su estación de trabajo del software Siemens NX MCD.
Nota: Verificar la conexión con el servidor de licencias, mediante ping a la IP 168.176.36.140.
- 6.3 Configurar los actuadores del sistema de manufactura virtual en el entorno de NX MCD.
- 6.4 Configurar los sensores del sistema de manufactura virtual en el entorno de NX MCD.

7. ENTREGA DE INFORME:

Cada grupo realizara un informe en formato libre de la práctica realizada, que básicamente corresponde a documentar la parte 6 de este documento, respondiendo a la siguiente interrogante ¿Qué aprendizajes le deja esta práctica y en que contextos lo puede aplicar? Se sugiere ayudarse de registros multimedia que faciliten la lectura del informe.

El informe se debe subir al Moodle en formato PDF de acuerdo con la programación del curso.

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>  <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--

6.2 LICENCIAMIENTO NX MCD:

Lo primero que se tiene que hacer es verificar la comunicación física entre la estación de trabajo PC y el gestor de licencias de NX. Verifique que el cable de red efectivamente está conectado en el PC y que se encuentra conectado a la red UNAL. Posteriormente podemos realizar un ping, para verificar la conexión física entre el PC y el servidor de licencia de NX:

```

Microsoft Windows [Versión 10.0.17134.765]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

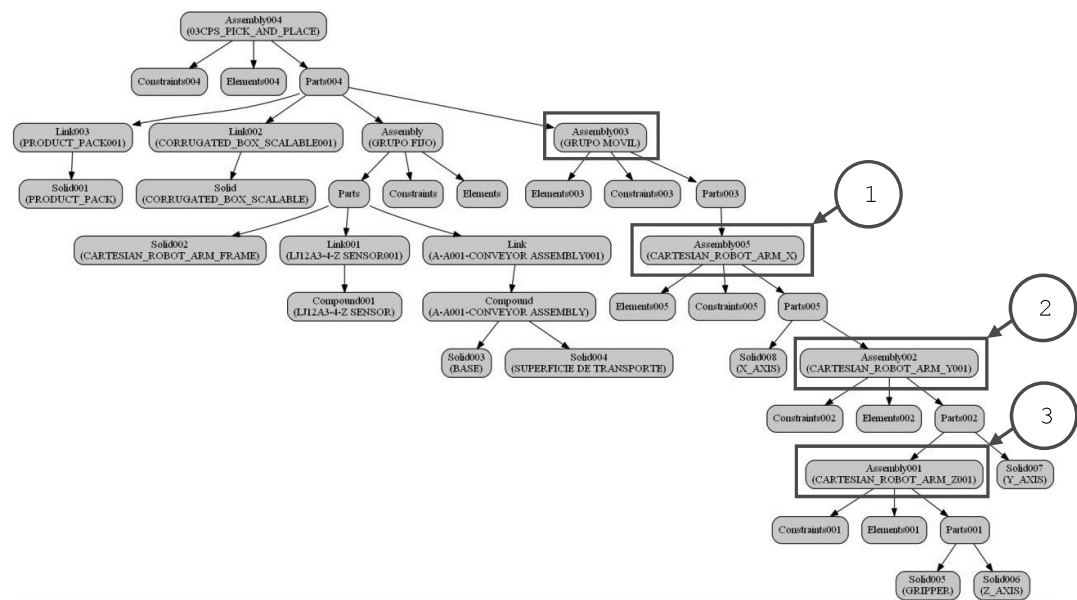
C:\Users\WS-Innovation>ping 168.176.36.140

Haciendo ping a 168.176.36.140 con 32 bytes de datos:
Respuesta desde 168.176.36.140: bytes=32 tiempo=9ms TTL=126
Respuesta desde 168.176.36.140: bytes=32 tiempo=11ms TTL=126
Respuesta desde 168.176.36.140: bytes=32 tiempo=313ms TTL=126
Respuesta desde 168.176.36.140: bytes=32 tiempo=5ms TTL=126


Estadísticas de ping para 168.176.36.140:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
              (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 5ms, Máximo = 313ms, Media = 84ms
    
```

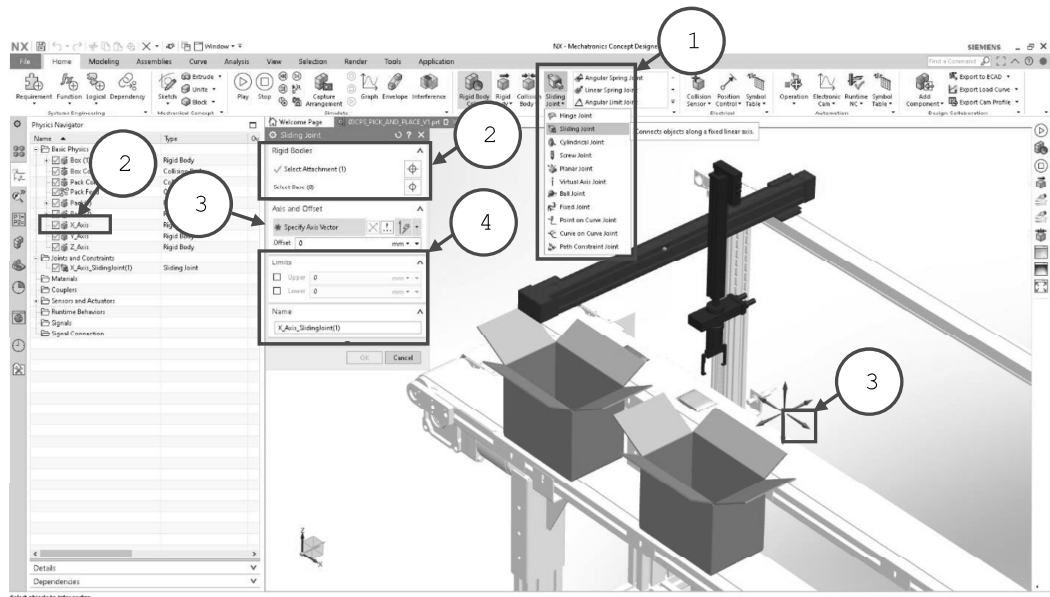
6.3 CONFIGURAR LOS ACTUADORES DEL SISTEMA DE MANUFACTURA VIRTUAL EN NX MCD.

Antes de definir la parte eléctrica de los actuadores es necesario realizar la configuración mecánica del sistema de manufactura virtual definiendo la cinemática de los mecanismos mediante la dependencia de cada una de las articulaciones (1)(2)(3).

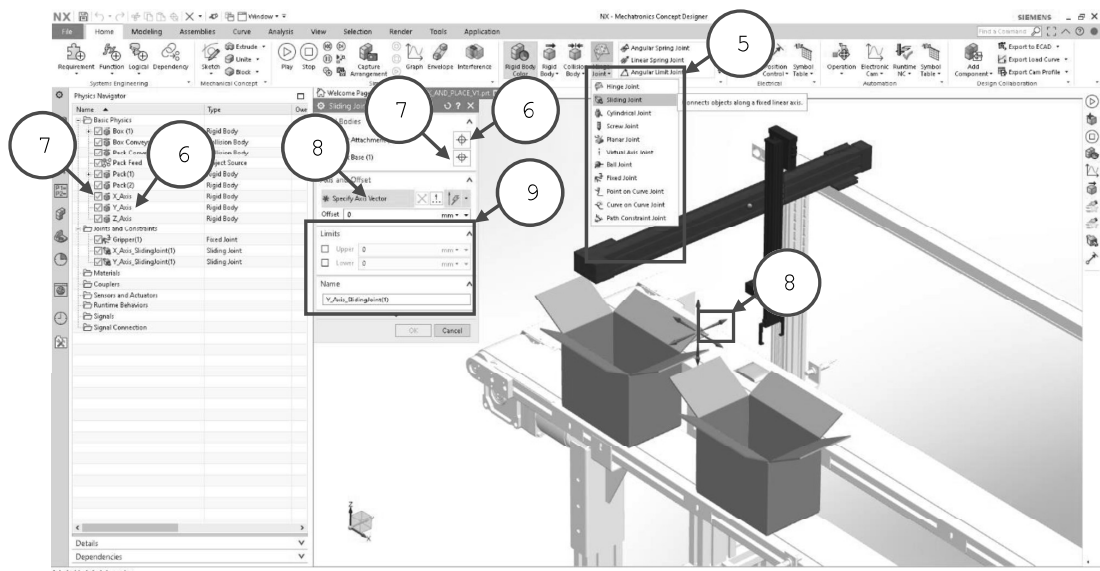


En este caso observamos una cinemática de cadena serial donde el eje X (1) no es dependiente, el eje Y (2) depende del movimiento del eje X, y el eje Z (3) depende del movimiento del eje Y.


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>	 <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--	---

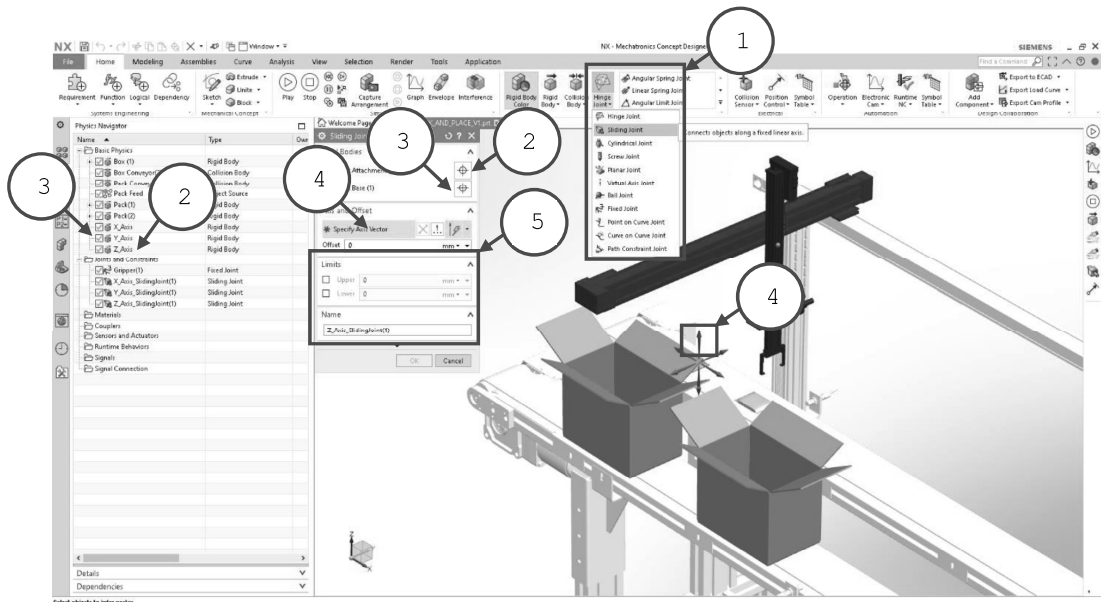


Este sistema de manufactura posee juntas de tipo prismática (1), para cada junta es necesario asignar el cuerpo rígido (2) el vector de movimiento (3), límites (en el caso que se requiera definir) y nombre de la junta (4), en este ejemplo se realiza la configuración del eje X sin dependencia.

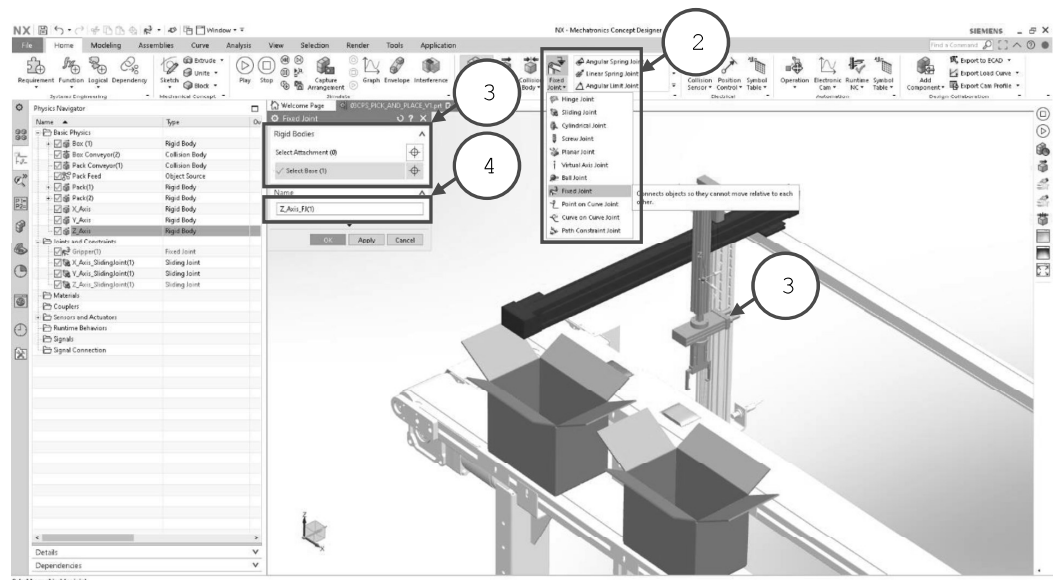


La junta del eje Y es de tipo prismática (5), es necesario asignar el cuerpo rígido (6) y su antecedente cinemático eje X (7). Posteriormente, el vector de movimiento (8), límites (en el caso que se requiera definir) y nombre de la junta (9), en este ejemplo se realiza la configuración del eje Y con dependencia.

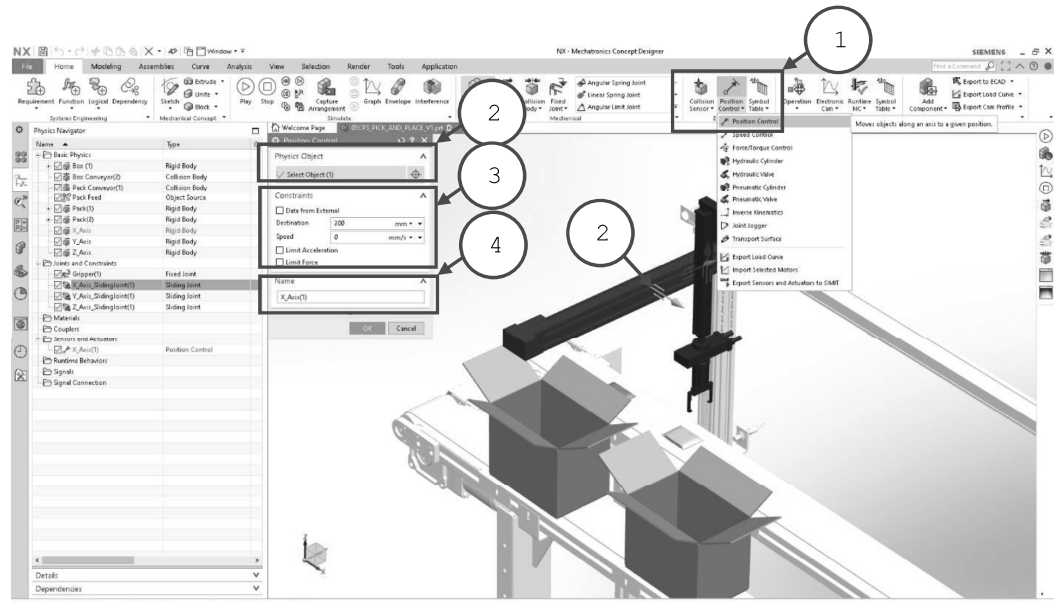
<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>	 <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--	--



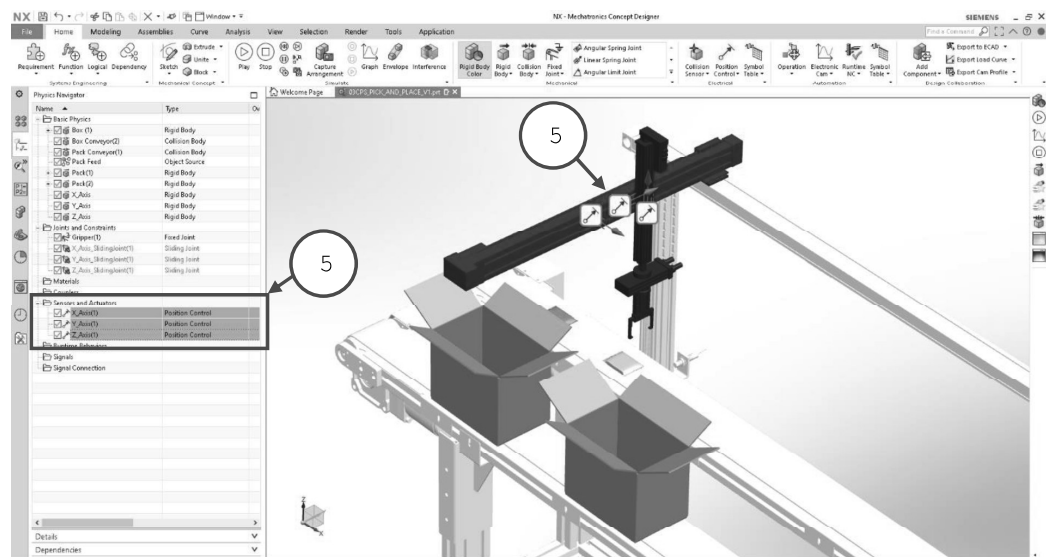
La junta del eje Z es de tipo prismática (1), es necesario asignar el cuerpo rígido (2) y su antecedente cinemático eje Y (3). Posteriormente, el vector de movimiento (4), límites (en el caso que se requiera definir) y nombre de la junta (5), en este ejemplo se realiza la configuración del eje Z con dependencia.




Para aplicaciones **pick and place** es necesario crear una junta de restricción fija (2) a la cual es necesario asignar como base la geometría del cuerpo rígido (3) en este caso el eje Z y asignar un nombre (4) que lo represente para reconocerlo fácilmente más adelante en la configuración de los sensores.

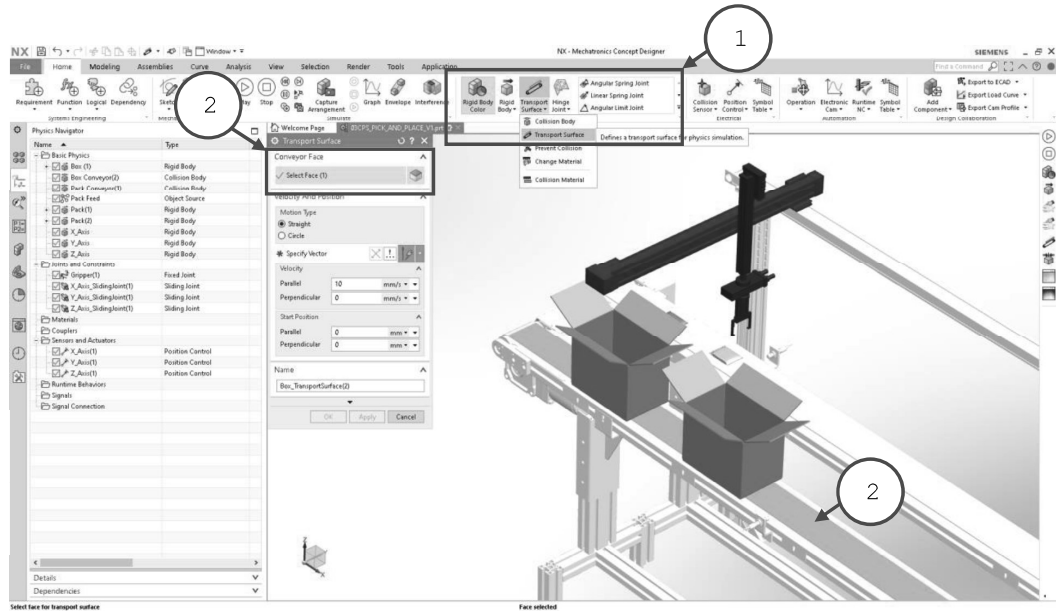


Una vez configurada las juntas y restricciones usaremos el comando control de posición (1) en el grupo eléctrico para comandar los ejes lineales seleccionando en la ventana gráfica la junta (2) que le corresponde a cada eje respectivamente. En este caso controlaremos la posición final del actuador del eje X en 200 mm (3) y asignaremos un nombre al actuador (4).

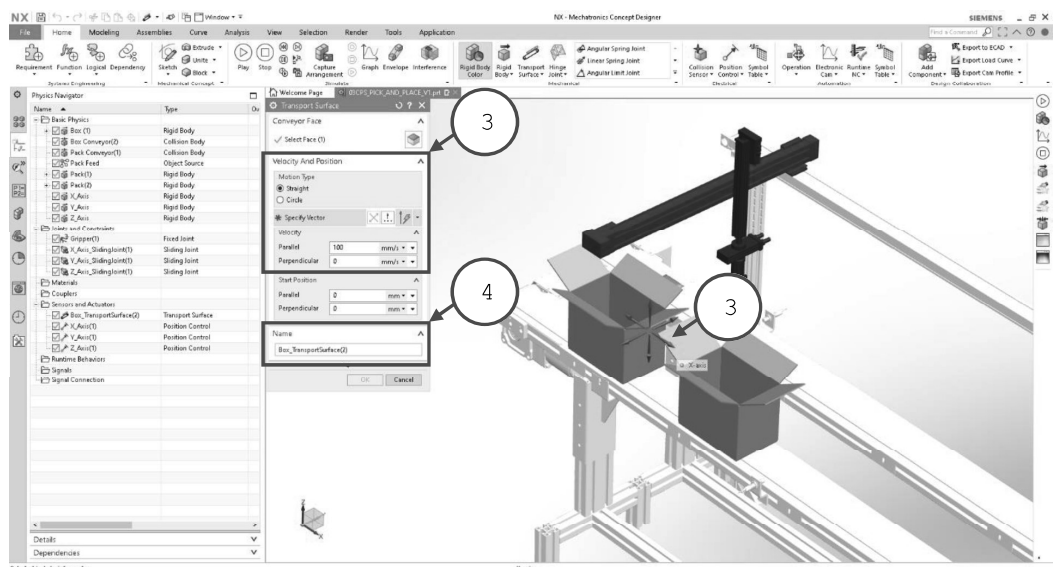


Realizar la configuración eléctrica para los tres actuadores teniendo en cuenta para el eje Y una posición final de 200 mm y para el eje Z una posición final de 200 mm igualmente (5).


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>  <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--

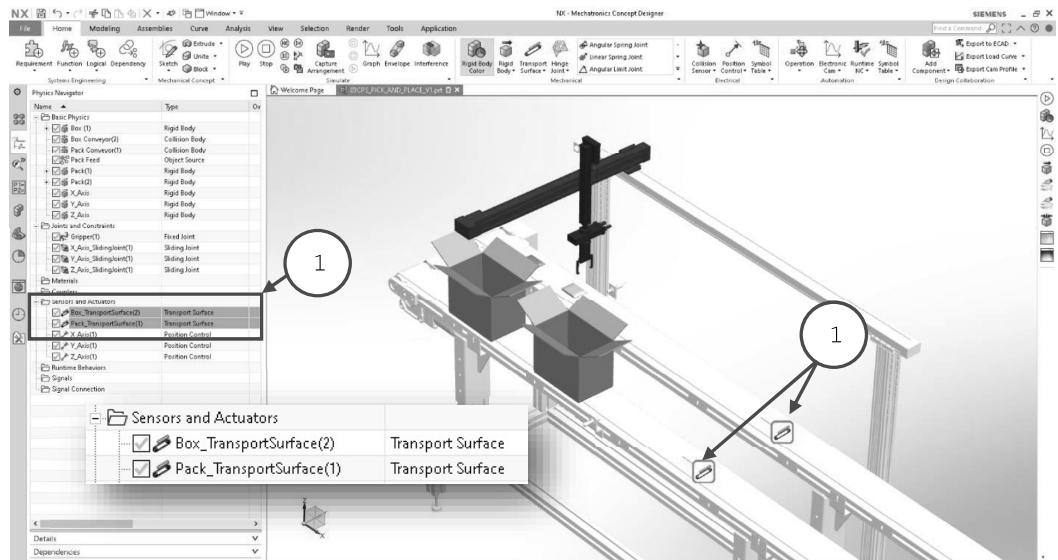


Otro actuador muy común de sistemas de manufactura son las bandas transportadoras, estas se pueden configurar en el grupo mecánico con el comando superficie de transporte (1). En este comando es necesario seleccionar la superficie donde se efectuará el recorrido (2).



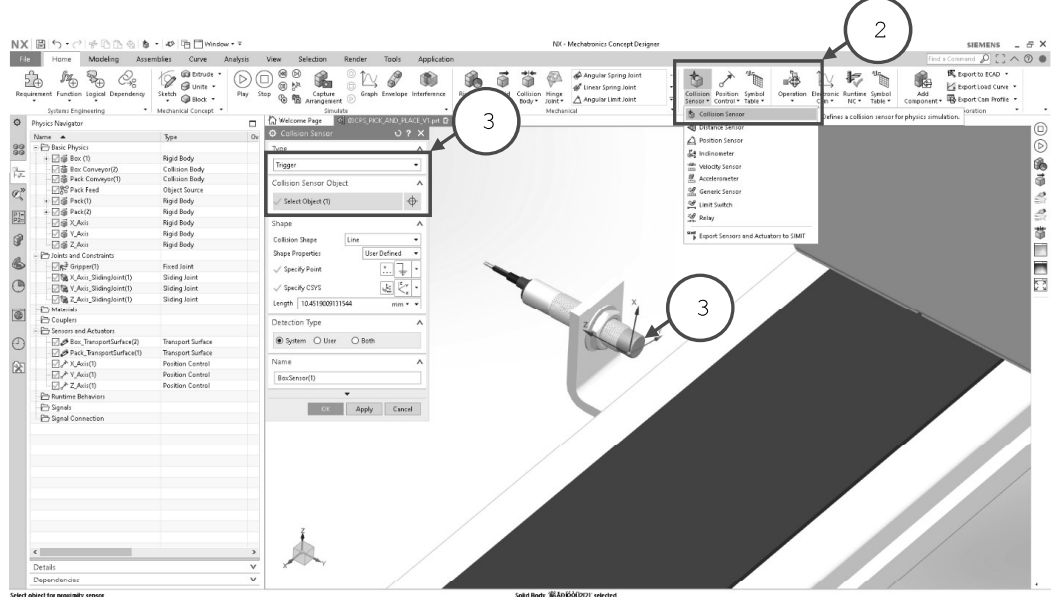
Dentro de este comando se puede asignar el tipo de movimiento y el vector que indica la dirección (3) además de su velocidad, para este caso colocaremos 100 mm/s paralela a la dirección asignada. No olvidar asignar una característica para facilitar la programación.

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>	 <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--	--

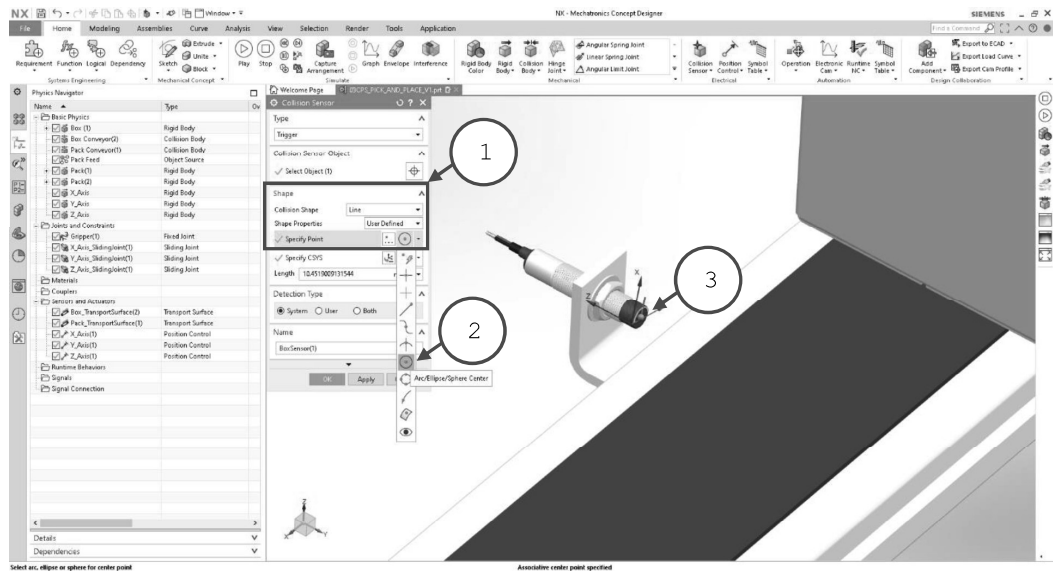


Configurar ambas bandas transportadoras en condiciones similares de dirección y velocidad.

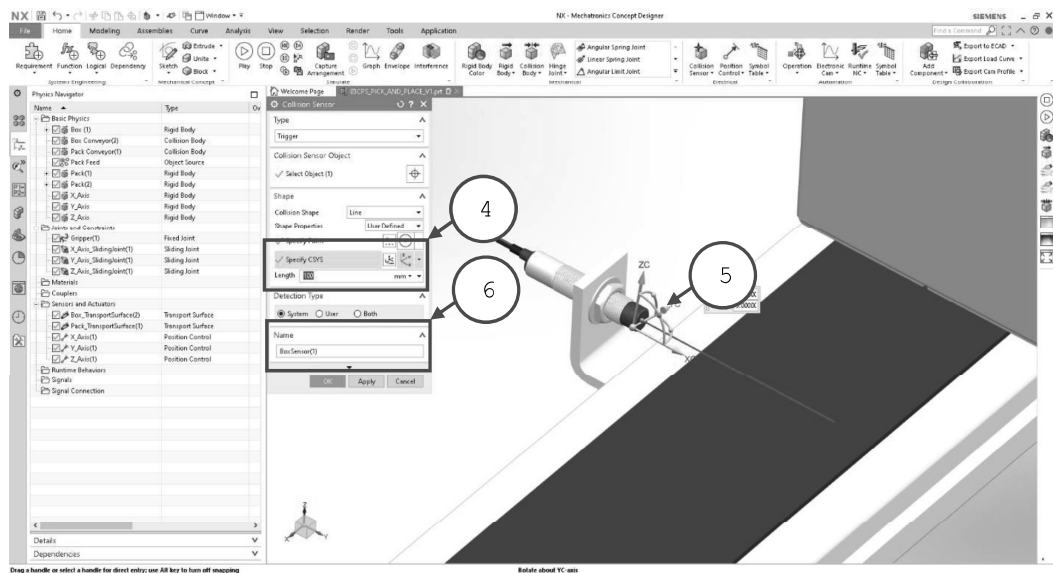
6.4 CONFIGURAR LOS SENSORES DEL SISTEMA DE MANUFACTURA VIRTUAL EN NX MCD




Un sensor de colisión es útil para detectar objetos en una posición específica, configuraremos uno para cada banda transportadora. Inicialmente configurar el sensor de colisión que se encuentra en el grupo eléctrico (2). En este caso tipo TRIGGER y podemos seleccionar un objeto asociado al sensor (3).

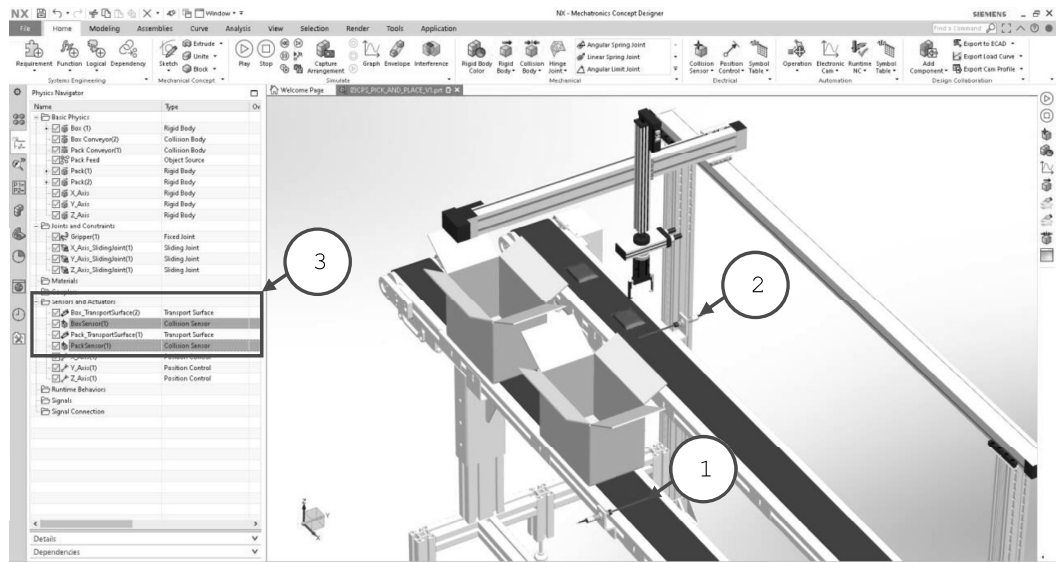


En este caso definiremos una línea de contacto para el sensor y las propiedades de forma serán definidas por el usuario (1). Para ubicar el punto de inicio de la señal del sensor especificaremos el punto inicial con ayuda de la referencia centro en arco (2) y seleccionando el arco en la ventana gráfica (3).

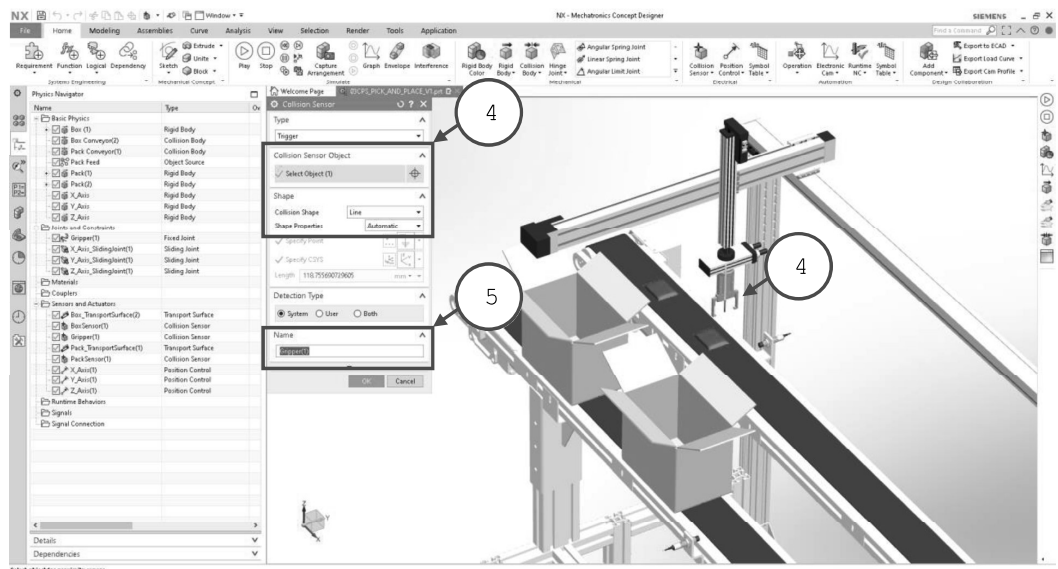


Para configurar la orientación del sensor usaremos especificaremos el CSYS (4) y con apoyo en la ventana grafica giraremos el sistema coordinado apuntando el eje X positivo en dirección que emite señal el sensor (5), y finalmente asignamos una longitud de la señal de 100 mm y el nombre al sensor.

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>	 <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--	--



En este caso se requiere definir ambos sensores (3) para detectar el ingreso de las cajas (1) y el ingreso de los paquetes (2) al mover las bandas transportadoras.



A continuación, definiremos el sensor para realizar el pick and place de producto, definiendo un sensor al seleccionar el efector final (4), definiendo su forma de colisión de línea y sus propiedades de forma automática. Finalmente asignar un nombre al sensor (5) para recordar su función.

D

Anexo: Guía validación y verificación virtual - Lógica/Control

Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas

Ingeniería Mecatrónica
Facultad de Ingeniería
Sede Bogotá



UNIVERSIDAD
NACIONAL
DE COLOMBIA

DEPARTAMENTO DE INGENIERÍA MECÁNICA Y MECATRÓNICA

AUTOMATIZACION DE PROCESOS DE MANUFACTURA	2017280-2
SERVOMECANISMOS	2017288-1
MOTORES Y CONTROLADORES INDUSTRIALES	2027978-1

Desarrollado por: Ubaldo Garcia Zaragoza
E-mail: ugarciaz@unal.edu.co, Oficina: Edificio 411, Laboratorio 101

Docentes: Víctor Hugo Grisales Palacio (Coordinador de Laboratorio)
E-mail: vhgrisalesp@unal.edu.co, Oficina: Edificio 411, Laboratorio 101
Eduardo Barrera Gualdrón
E-mail: ebarrerag@unal.edu.co, Oficina: Edificio 411, Oficina 204

Horario: Acorde al espacio académico (Verificar horario asignado)

Salón: Edificio 411-101 Laboratorio de Automatización

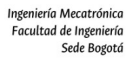

Guía para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas #3 – Secuencias de control.

1. CONTEXTO

Cuando la industria se vio en la necesidad de integrar máquinas, herramientas, energía y trabajo para producir y transformar materia prima o material, formó los sistemas de manufactura [1]. Estos sistemas se clasificaron en dos categorías: aquellos que son medidos por variables de carácter físico o químico cuyo rango es continuo, denominados procesos continuos y aquellos que son medidos por escalas de conteo o numeración donde su producción son piezas, partes y ensamblajes, denominados procesos discretos [2].

En los sistemas de manufactura automáticos interactúan sistemas mecánicos, eléctricos, electrónicos y en algunos casos la supervisión y/o manipulación de personas para hacerlos funcionar o verificar su desempeño. La mayoría de estos sistemas para que operen en su totalidad o parcialmente por sí solos, necesitan las instrucciones de un control lógico programable (PLC, por sus siglas en inglés), estos componentes se pueden programar bajo el estándar IEC 1131.

El diagrama GRAFCET de control con etapas y transiciones (Graphe Fonctionnel de Commande Etape Transition, por sus siglas en francés), es un grafo funcional de control muy útil para representar funciones de control secuencial o control discreto [3]. Este tipo de representación gráfica tiene gran aplicación en la automatización de sistemas de manufactura de procesos discretos ya que permite describir los procesos a automatizar, considerando las acciones a ejecutar, y los procesos intermedios que generan estas acciones basándose en una sintaxis simple, que ofrece de manera sistémica el desarrollo del automatismo.

Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	 
---	---

2. DESCRIPCIÓN:

Esta guía permitirá que el participante comprenda, configure y condicione los actuadores y sensores en una secuencia de control dentro de la simulación de un sistema de manufactura virtual. Logrando comprender los parámetros principales para tener en cuenta tanto para actuadores como sensores.

En esta guía se mostrará la configuración de algunas instrucciones tales como: parámetros y tiempo de ejecución, condiciones, etiquetas y simulación.

Se debe tener en cuenta que el desarrollo de la totalidad de la guía estará orientada a configurar un sistema de manufactura virtual donde se mostrará las siguientes consideraciones:

- Comprender el GRAFCET de descripción funcional de un Sistema de Manufactura
- Configurar superficies de transporte bajo una condición dada del sistema de manufactura virtual.
- Configurar actuadores bajo una condición dada del sistema de manufactura virtual.
- Realizar una operación de pick and place entre el efector y un producto.

3. OBJETIVOS DE LA PRACTICA:

- 3.1 Interpretar el GRAFCET de descripción funcional de un Sistema de Manufactura discreto.
- 3.2 Configurar la secuencia de control discreto de un Sistema de Manufactura Virtual.

4. RECURSOS PARA LA PRACTICA:

- 4.1 Estación de trabajo PC.
- 4.2 Software PLM NX Mechatronics Concept Designer.
- 4.3 Archivo 03CPS_PICK_AND_PLACE con componentes rígidos, de colisión, sensores y actuadores.
Nota: Ver y realizar guía 2, configuración de sensores y actuadores.

5. TIEMPO SUGERIDO PARA EL DESARROLLO DE LA PRACTICA: 1 día.


6. PRACTICA A REALIZAR:

- 6.1 Leer la secuencia de configuración que se encuentra en el contenido de esta Práctica.
- 6.2 Garantizar licencia en su estación de trabajo del software Siemens NX MCD.
Nota: Verificar la conexión con el servidor de licencias, mediante ping a la IP 168.176.36.140.
- 6.3 GRAFCET de descripción funcional del sistema de manufactura virtual.
- 6.4 Secuencia de control del sistema de manufactura virtual en el entorno de NX MCD.

7. ENTREGA DE INFORME:

Cada grupo realizara un informe en formato libre de la práctica realizada, que básicamente corresponde a documentar la parte 6 de este documento, respondiendo a la siguiente interrogante ¿Qué aprendizajes le deja esta práctica y en que contextos lo puede aplicar? Se sugiere ayudarse de registros multimedia que faciliten la lectura del informe.

El informe se debe subir al Moodle en formato PDF de acuerdo con la programación del curso.

Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá	 UNIVERSIDAD NACIONAL DE COLOMBIA
---	---	--

6.2 LICENCIAMIENTO NX MCD:

Lo primero que se tiene que hacer es verificar la comunicación física entre la estación de trabajo PC y el gestor de licencias de NX. Verifique que el cable de red efectivamente está conectado en el PC y que se encuentra conectado a la red UNAL. Posteriormente podemos realizar un ping, para verificar la conexión física entre el PC y el servidor de licencia de NX:

```

Microsoft Windows [Versión 10.0.17134.765]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

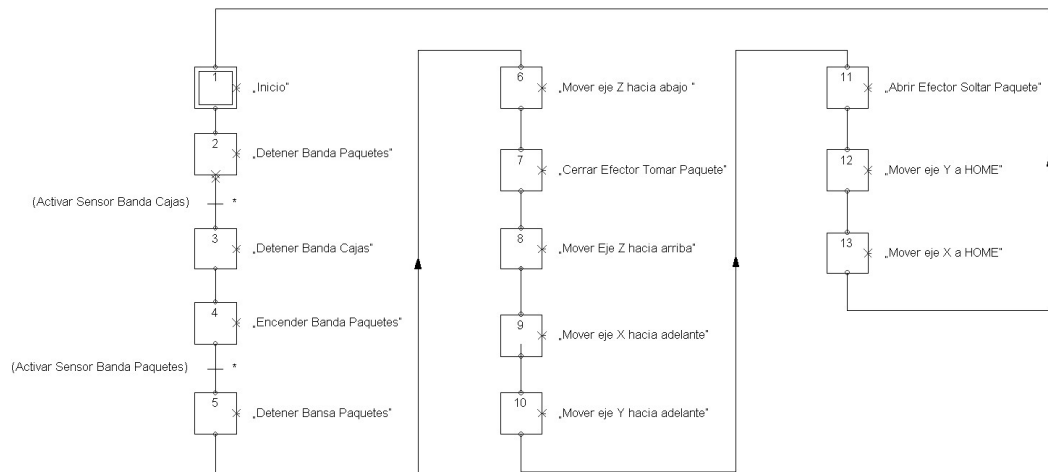
C:\Users\WS-Innovation>ping 168.176.36.140

Haciendo ping a 168.176.36.140 con 32 bytes de datos:
Respuesta desde 168.176.36.140: bytes=32 tiempo=9ms TTL=126
Respuesta desde 168.176.36.140: bytes=32 tiempo=11ms TTL=126
Respuesta desde 168.176.36.140: bytes=32 tiempo=313ms TTL=126
Respuesta desde 168.176.36.140: bytes=32 tiempo=5ms TTL=126

Estadísticas de ping para 168.176.36.140:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
              (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 5ms, Máximo = 313ms, Media = 84ms
    
```

6.3 GRAFCET DESCRIPCIÓN FUNCIONAL DEL SISTEMA DE MANUFACTURA VIRTUAL.

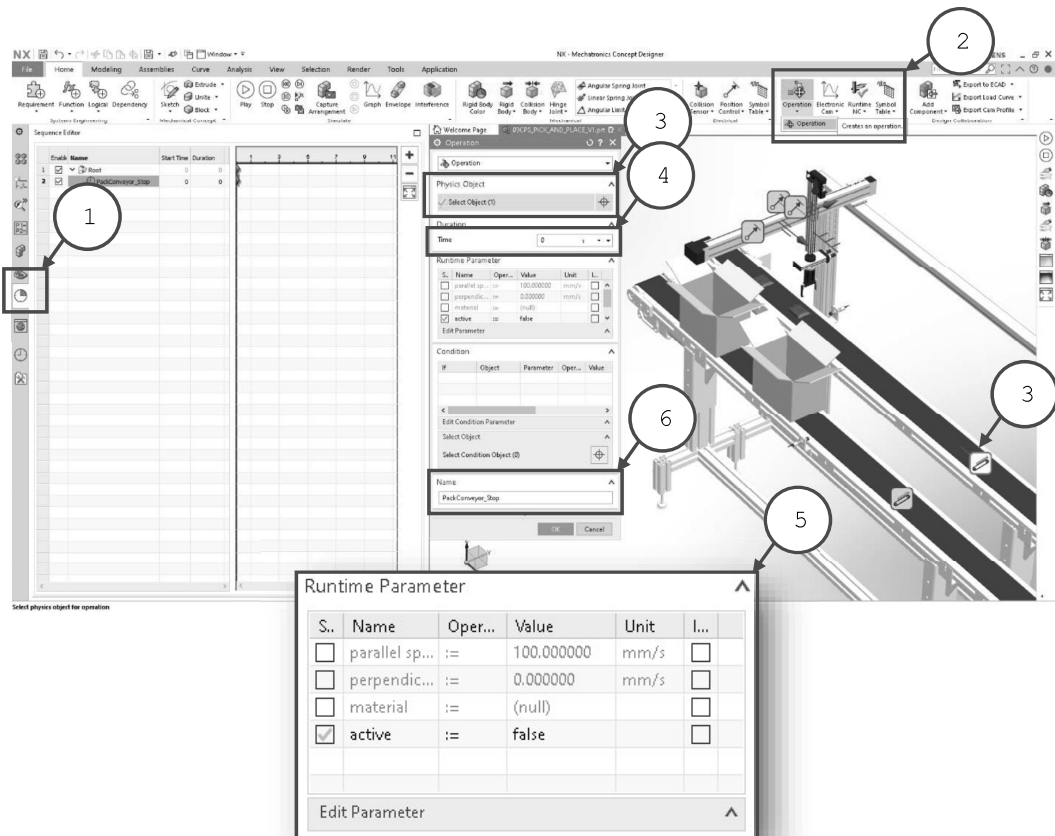
Antes de definir la secuencia de control en el editor es necesario disponer de un GRAFCET con descripción funcional donde se evidencien las etapas y transiciones del sistema de manufactura.



En este caso particular se evidencian 13 etapas y dos transiciones que están relacionadas con los sensores de cada banda respectivamente. Para simplificar el ejercicio los movimientos de los actuadores lineales no dispondrán de finales de carrera, solamente se comandarán en posición de la siguiente manera: en posición HOME X es igual a 0 mm, Y igual a 0 mm y Z igual a 0 mm, para el primer movimiento Z descenderá -104 mm, después subirá a una posición de 109 mm una vez sujeto el producto. El eje X se desplazará 400 mm y el eje Y -279 mm indexando cada eje. Finalmente, una vez esté en esa posición soltará el paquete y retornara a su posición inicial en los ejes X y Y retornando a HOME.


6.4 SECUENCIA DE CONTROL DEL SISTEMA DE MANUFACTURA VIRTUAL.

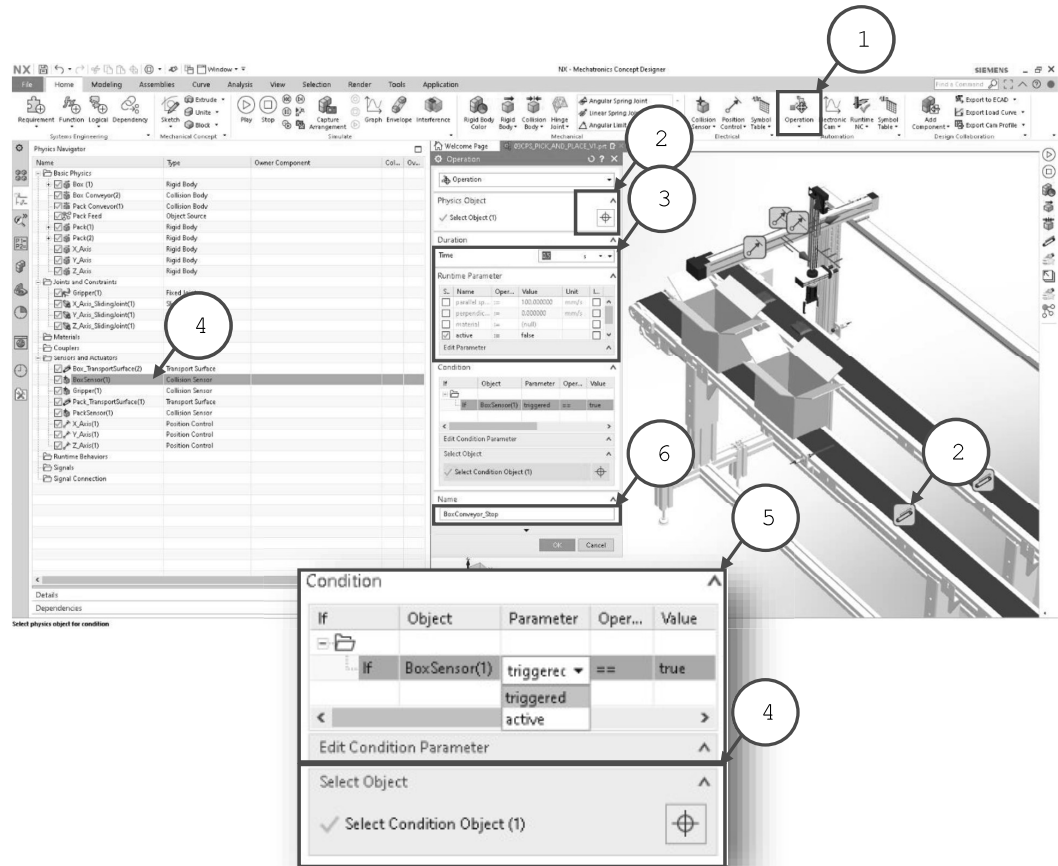
En el editor de secuencias (1) de NX MCD definiremos la secuencia de control antes de conectarla con el programa en el PLC, utilizando el comando operación (2) del grupo de automatización para crear una nueva instrucción por cada etapa que se requiera para el sistema de manufactura virtual.



En la ventana emergente seleccionaremos el actuador que ejecutara la instrucción, en este etapa la banda transportadora de los paquetes (3), dando clic en la ventana gráfica y seleccionando el icono. Posteriormente asignaremos el tiempo que dura dicha instrucción, en este caso asignaremos un valor de 0 segundos (4) ya que queremos que se detenga inmediatamente.

En la sección de parámetros podemos seleccionar distintas propiedades a controlar (5) que se ajuste al requerimiento del sistema de manufactura virtual, en este caso el parámetro de interés es indicar si esta encendida o apagada la banda transportadora por ello configuraremos el parámetro ACTIVE=FALSE, cabe resaltar que existen otros parámetros como el de la velocidad paralela o perpendicular, estos parámetros dependerán de los atributos de cada actuador. Finalmente colocamos una etiqueta a la operación PackConveyor_Stop (5).

Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá	 UNIVERSIDAD NACIONAL DE COLOMBIA
---	---	---




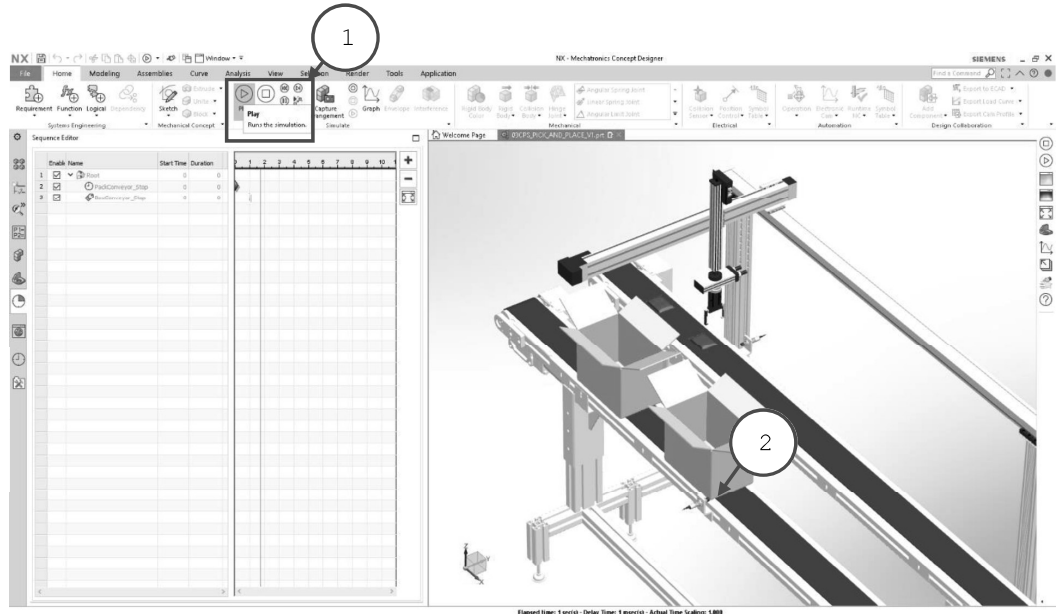
Procederemos a crear la segunda etapa con el comando operación en el grupo de automatización (1), en este caso configuraremos una operación condicionada, esto significa que su acción depende de otra señal externa, es importante tener activo el navegador de física para tener a la mano los sensores y actuadores a configurar (4).

Asignaremos el actuador de la banda transportadora de la caja (2) como objeto de control, adicionalmente configuraremos el tiempo de duración a 0 segundos y en la pestaña de parámetros configuraremos este actuador (3) ACTIVE=FALSE indicando que se apague una vez cumplida la condición de igual manera que el anterior.

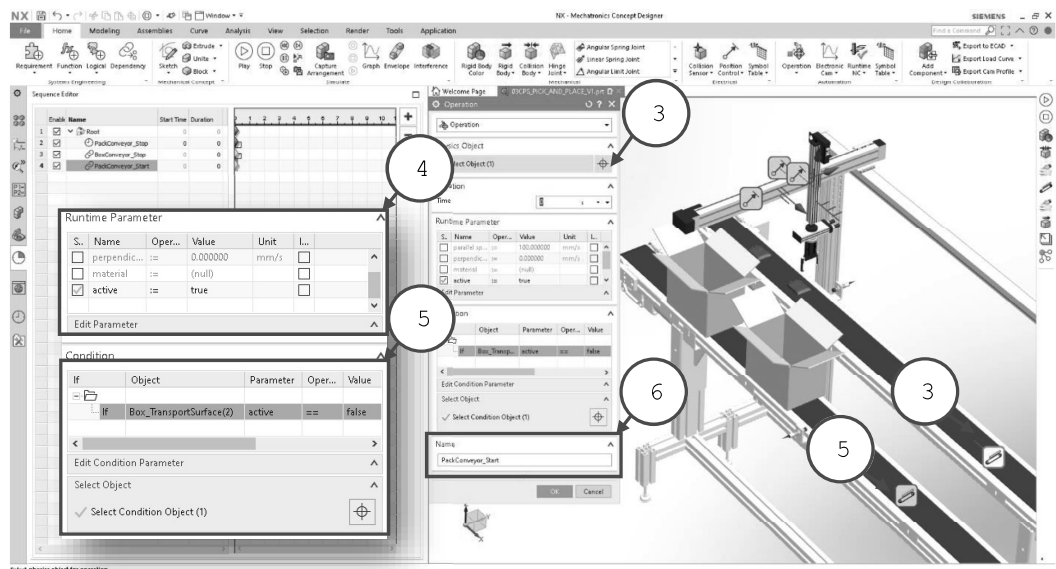
Para configurar la condición primero debemos seleccionar el elemento que condiciona, en este caso el sensor (4) de la banda transportadora de la caja, seleccionándolo en el navegador de física. Posteriormente necesitamos asignar el tipo de condición TRIGGERED==TRUE (5) indicando que cuando el sensor se active se detendrá la banda transportadora. En este caso dependiendo del tipo de sensor o elemento que condiciona se podrá seleccionar diversos parámetros para la condición.

Una vez configurado la(s) condición(es) requeridas en el sistema de manufactura virtual, colocaremos una etiqueta al nombre de la operación, en este caso asignaremos: BoxConveyor_Stop (6).


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	 <p style="font-size: small;">Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p> <p style="font-size: small;">UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	---

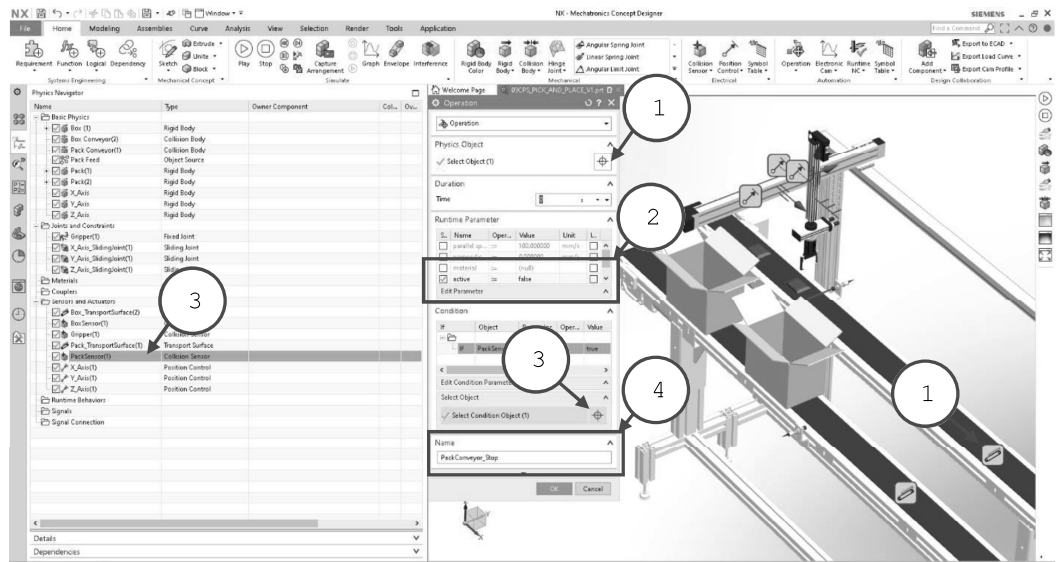


En esta etapa podemos verificar la simulación con el comando PLAY (1) del grupo de simulación y apreciar el movimiento de la caja y su detención cuando el sensor la detecta (2). **No olvidar detener la simulación con el comando STOP para continuar editando la secuencia de control.**

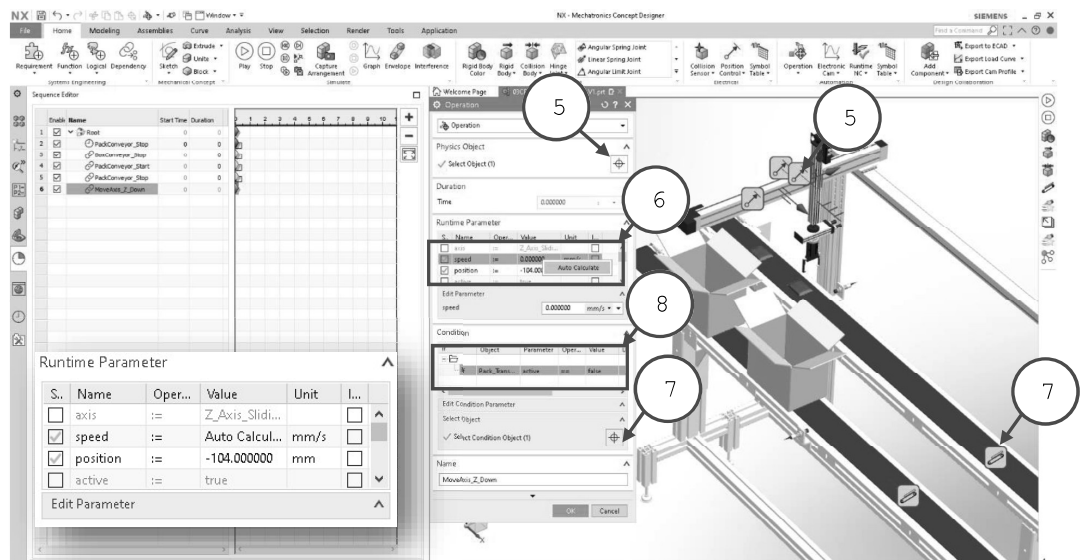


En esta etapa configurar el actuador de la banda transportadora de los paquetes (3), al iniciar su funcionamiento ACTIVE=TRUE (4) y condicionar su acción TRIGGERED==FALSE con la detención de la banda transportadora de las cajas (5) y colocar la etiqueta PackConveyor_Start (6).


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	 <p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p> <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	---

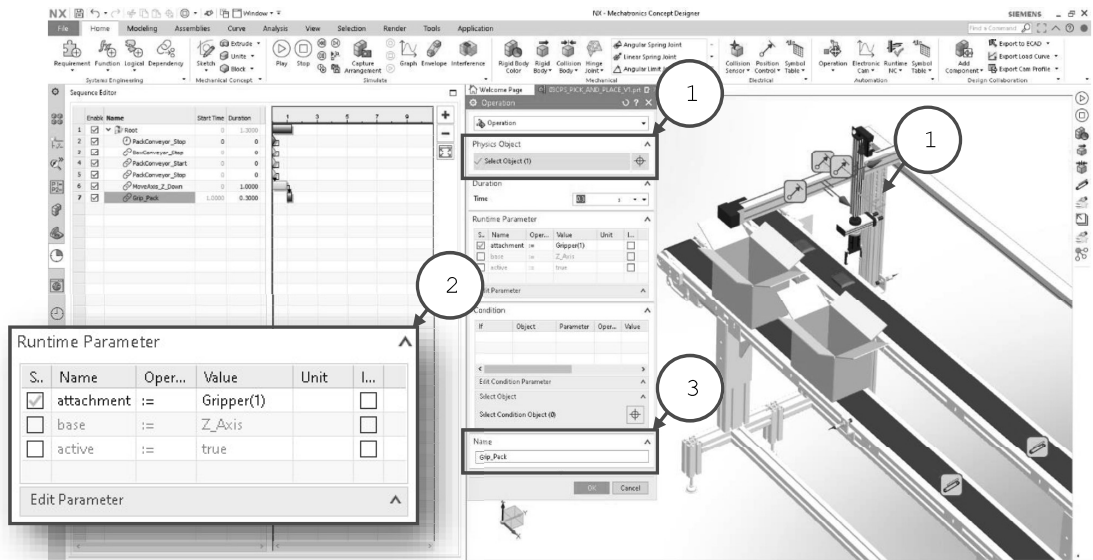


En esta etapa buscaremos para la banda transportadora de los paquetes mediante el parámetro ACTIVE=FALSE (2), condicionando su acción con la activación del sensor de paquetes (3) TRIGGERED==TRUE y colocando la etiqueta PackConveyor_Start (4).

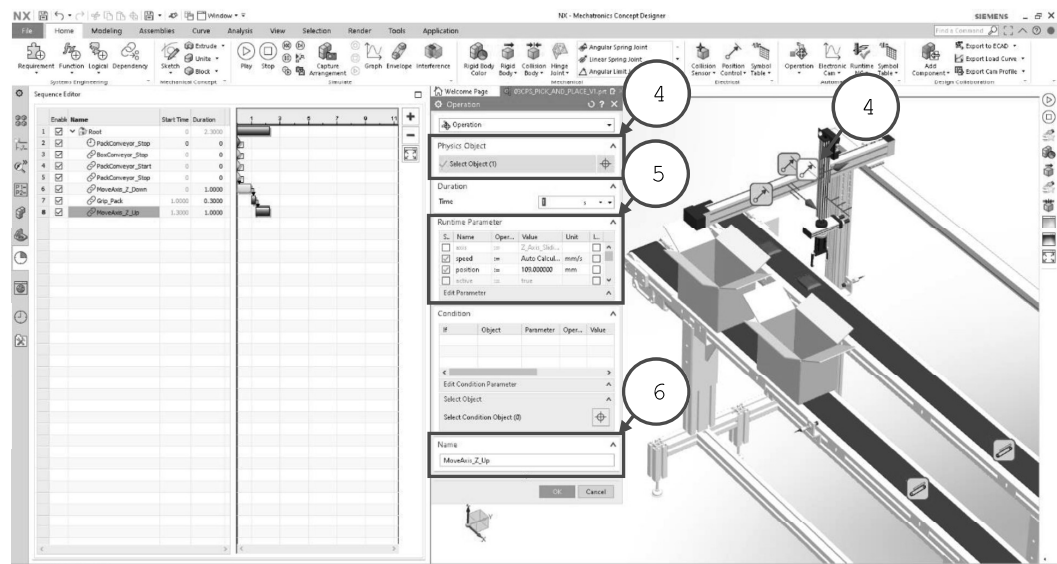


En esta etapa se mueve el eje Z (5) hacia abajo por un 1 segundo. Ajustar el parámetro de velocidad con clic derecho: Auto Calculada, (6) a una posición de -104 mm, condicionando su accionamiento con la detención de la banda transportadora de paquetes (7) ACTIVE=FALSE (8) y etiqueta MoveAxis_Z_Down.


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	 <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--

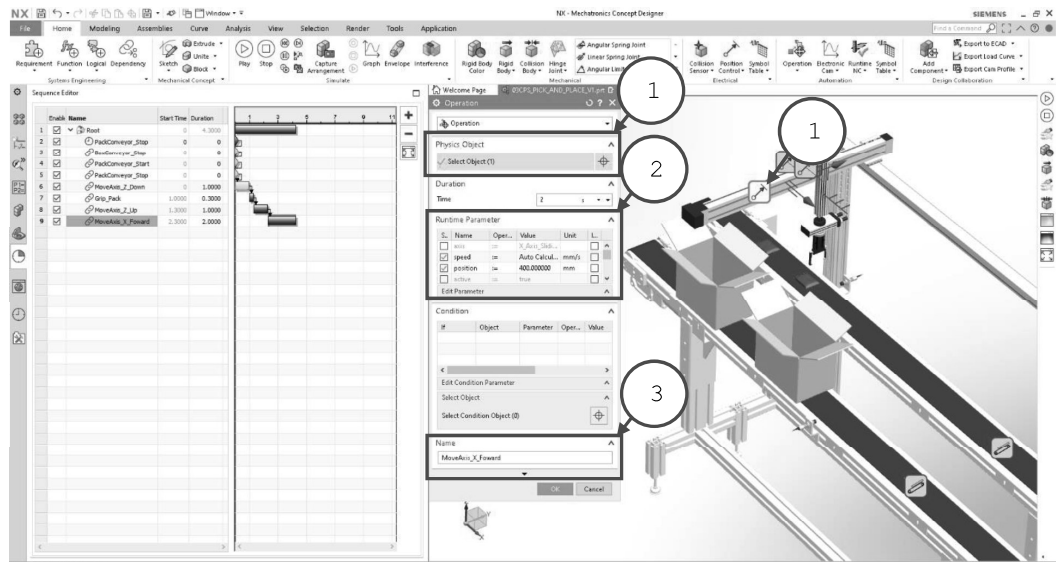


En esta etapa accionaremos el efector para sujetar el paquete seleccionando la restricción fija (1) dando un tiempo de 0.3 segundos de accionamiento. Asignaremos el parámetro ATTACHMENT=GRIPPER (2) para dar la funcionalidad de agarre y colocaremos la etiqueta Grip_Pack (3).

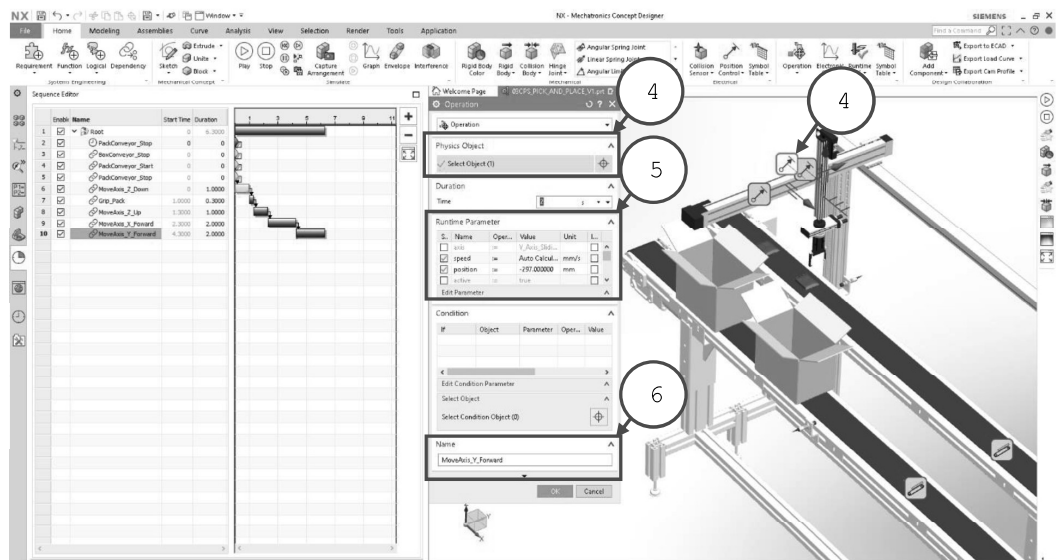


En esta etapa se mueve el eje Z (4) hacia arriba por un 1 segundo, en este caso con el paquete en el efector. Ajustar el parámetro de velocidad con clic derecho: Auto Calculada, (5) a una posición de 109 mm, y colocar la etiqueta MoveAxis_Z_Up.


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>	 <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--	--

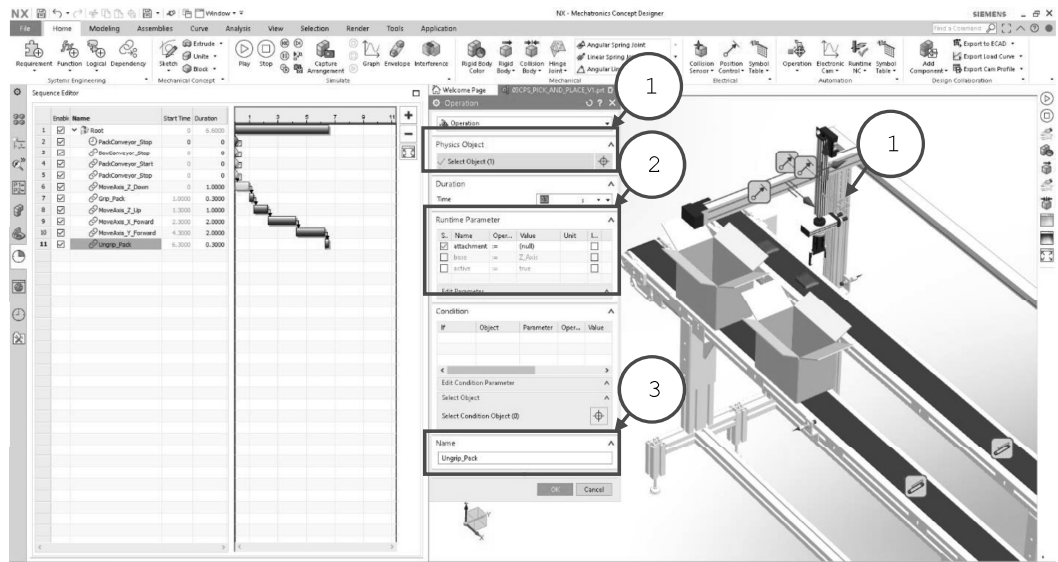


En esta etapa se mueve el eje X (1) hacia adelante por un 2 segundo, en este caso con el paquete en el efector. Ajustar el parámetro de velocidad con clic derecho: Auto Calculada, (2) a una posición de 400 mm, y colocar la etiqueta MoveAxis_X_Forward (3).

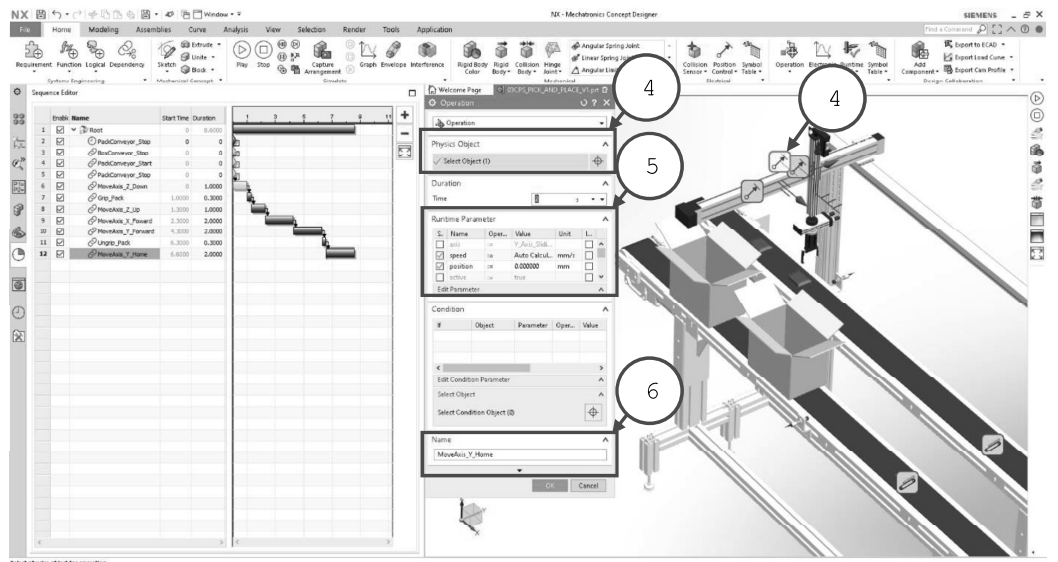


En esta etapa se mueve el eje Y (4) hacia adelante por un 2 segundo, en este caso con el paquete en el efector. Ajustar el parámetro de velocidad con clic derecho: Auto Calculada, (5) a una posición de -279 mm, y colocar la etiqueta MoveAxis_Y_Forward (6).


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	
--	---

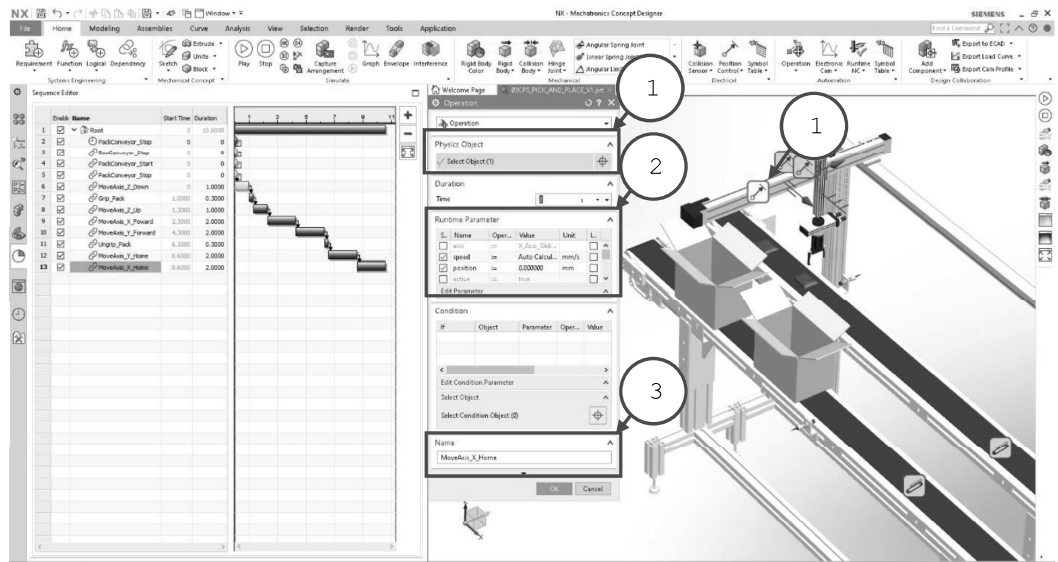


En esta etapa accionaremos el efector para soltar el paquete seleccionando la restricción fija (1) dando un tiempo de 0.3 segundos de accionamiento. Asignaremos el parámetro ATTACHMENT=(NULL) (2) para dar la funcionalidad de soltar y colocaremos la etiqueta Ungrip_Pack (3).

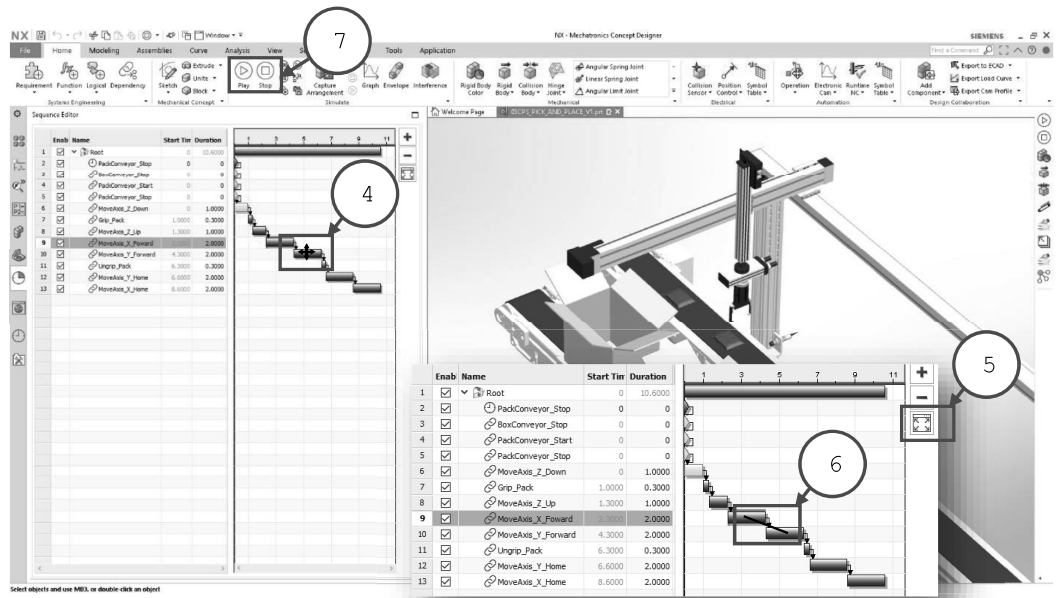


En esta etapa se mueve el eje Y (4) hacia home por un 2 segundo, en este caso sin el paquete en el efector. Ajustar el parámetro de velocidad con clic derecho: Auto Calculada, (5) a una posición de 0 mm, y colocar la etiqueta MoveAxis_Y_Home (6).

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>	 <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--	--



En esta etapa se mueve el eje X (1) hacia home por un 2 segundo, en este caso sin el paquete en el efector. Ajustar el parámetro de velocidad con clic derecho: Auto Calculada, (2) a una posición de 0 mm, y colocar la etiqueta MoveAxis_X_Home (3).



Desplazar las etapas de forma secuencial en la línea del tiempo con el cursor (4), generar dependencia entre etapas arrastrando sin soltar el cursor conectando las etapas dependientes (6). Finalmente, ajustar la línea del tiempo (5) y ejecutar la simulación (7) para verificar y validar el sistema de manufactura virtual.

E

Anexo: Guía validación y verificación virtual - Comunicaciones

Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas

Ingeniería Mecatrónica
Facultad de Ingeniería
Sede Bogotá



UNIVERSIDAD
NACIONAL
DE COLOMBIA

DEPARTAMENTO DE INGENIERÍA MECÁNICA Y MECATRÓNICA

AUTOMATIZACION DE PROCESOS DE MANUFACTURA	2017280-2
SERVOMECANISMOS	2017288-1
MOTORES Y CONTROLADORES INDUSTRIALES	2027978-1

Desarrollado por: Ubaldo Garcia Zaragoza
E-mail: ugarciaz@unal.edu.co, Oficina: Edificio 411, Laboratorio 101

Docentes: Víctor Hugo Grisales Palacio (Coordinador de Laboratorio)
E-mail: vhgrisalesp@unal.edu.co, Oficina: Edificio 411, Laboratorio 101
Eduardo Barrera Gualdrón
E-mail: ebarrerag@unal.edu.co, Oficina: Edificio 411, Oficina 204

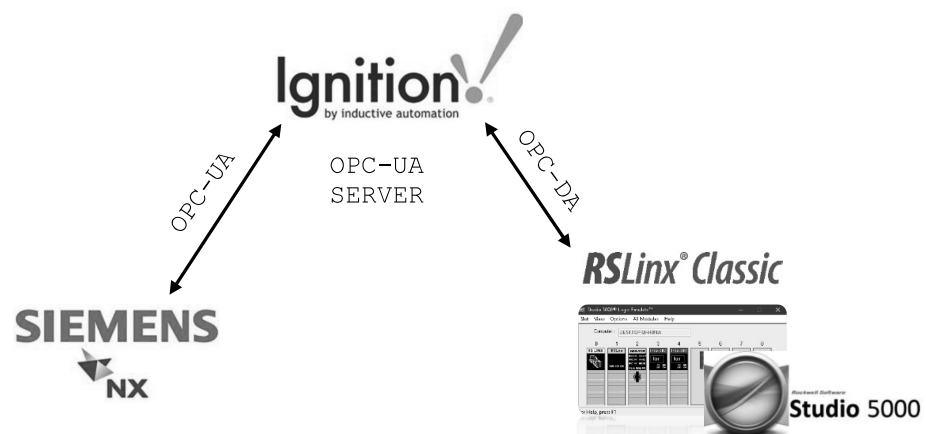
Horario: Acorde al espacio académico (Verificar horario asignado)

Salón: Edificio 411-101 Laboratorio de Automatización



Guía para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas #4 – Comisionamiento Virtual.

1. CONTEXTO

Para lograr la conectividad entre el sistema de manufactura virtual y el programa en el controlador es necesario el uso de una topología y arquitectura de comunicaciones como la de un servidor OPC [1], este permitirá la integración con múltiples plataformas como se muestra en la figura, dando vía a toda la información de señales de entrada y salida las cuales viajarán vía ethernet estableciendo una comunicación bidireccional.



Fuente: Autor

Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	 
---	---

2. DESCRIPCIÓN:

Esta guía permitirá que el participante comprenda, configure y condicione un emulador virtual de una unidad de control, además de descargar programas para probar la lógica. Logrando comprender los parámetros principales de configuración al momento validar y verificar un programa en un sistema de manufactura virtual con funcionalidades ciberfísicas.

En esta guía se mostrará la configuración de algunas instrucciones tales como: emulador virtual, servidor OPC DA, cliente OPC DA, mapeo de señales, y simulación y verificación.

Se debe tener en cuenta que el desarrollo de la totalidad de la guía estará orientada a configurar un sistema de manufactura virtual donde se mostrará las siguientes consideraciones:

- Comprender el funcionamiento y configuración de un emulador virtual de PLC
- Configurar red cliente-servidor OPC DA.
- Configurar servidor y señales de mapeo en un sistema de manufactura virtual.
- Validar y verificar el funcionamiento de un sistema de manufactura virtual bajo control discreto.

3. OBJETIVOS DE LA PRACTICA:

- 3.1 Configuración y descarga de programas en unidad de control virtual.
- 3.2 Configuración de las comunicaciones industriales virtuales OPC DA.
- 3.3 Comunicación entre el servidor OPC DA y el sistema de manufactura virtual.

4. RECURSOS PARA LA PRACTICA:

- 4.1 Estación de trabajo PC.
- 4.2 Software PLM NX Mechatronics Concept Designer.
- 4.3 Software Studio 5000
- 4.4 Software RSLINX Classic
- 4.5 Software Studio 5000 Logix Emulate
- 4.6 Archivo 03CPS_PICK_AND_PLACE con secuencias de control.
Nota: Ver y realizar guía 3, configuración de secuencias con señales de sensores y actuadores.

5. TIEMPO SUGERIDO PARA EL DESARROLLO DE LA PRACTICA: 1 día.

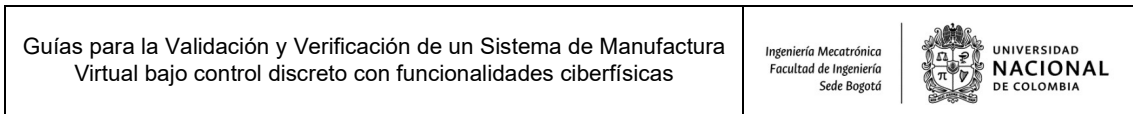
6. PRACTICA A REALIZAR:

- 6.1 Leer la secuencia de configuración que se encuentra en el contenido de esta Práctica.
- 6.2 Configuración y descarga de programas en unidad de control virtual
- 6.3 Configuración de las comunicaciones industriales virtuales OPC DA-OPC UA.
- 6.4 Garantizar licencia en su estación de trabajo del software Siemens NX MCD.
Nota: Verificar la conexión con el servidor de licencias, mediante ping a la IP 168.176.36.140.
- 6.5 Comunicación entre el servidor OPC UA y el sistema de manufactura virtual

7. ENTREGA DE INFORME:

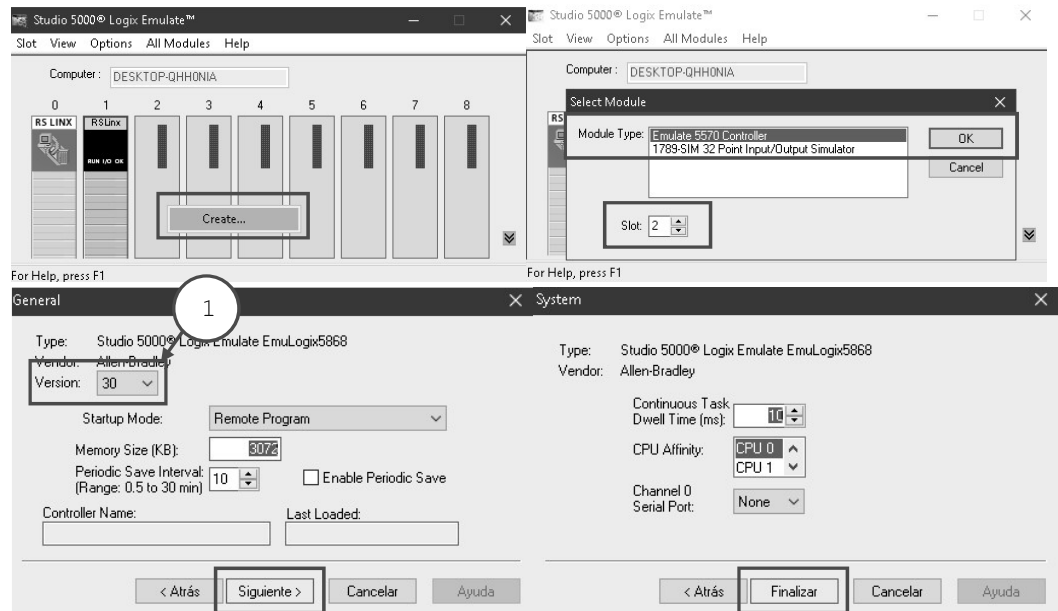
Cada grupo realizara un informe en formato libre de la práctica realizada, que básicamente corresponde a documentar la parte 6 de este documento, respondiendo a la siguiente interrogante ¿Qué aprendizajes le deja esta práctica y en que contextos lo puede aplicar? Se sugiere ayudarse de registros multimedia que faciliten la lectura del informe.

El informe se debe subir al Moodle en formato PDF de acuerdo con la programación del curso.

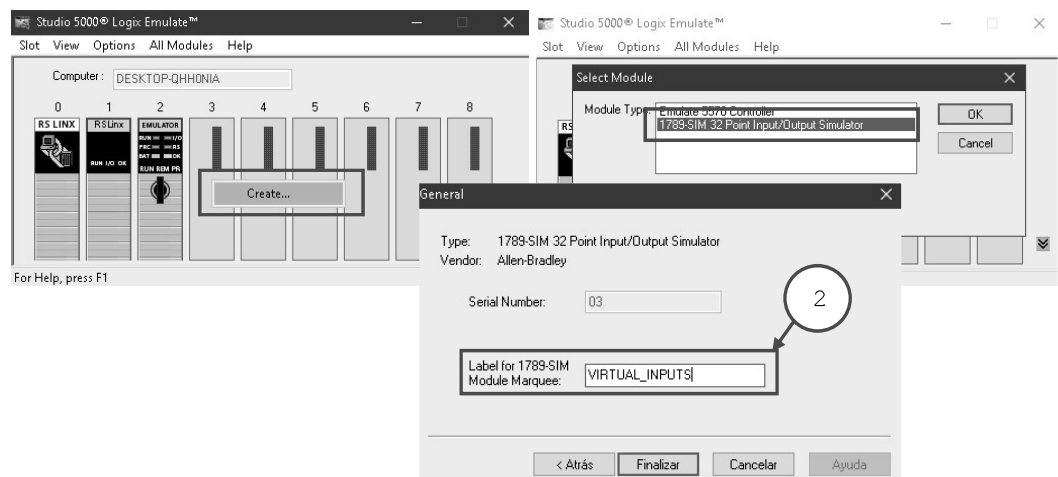


6.2 CONFIGURACIÓN Y DESCARGA DE PROGRAMAS EN UNIDAD DE CONTROL VIRTUAL:


Como primera instancia configuraremos el emulador del controlador STUDIO 5000 LOGIX EMULATE:

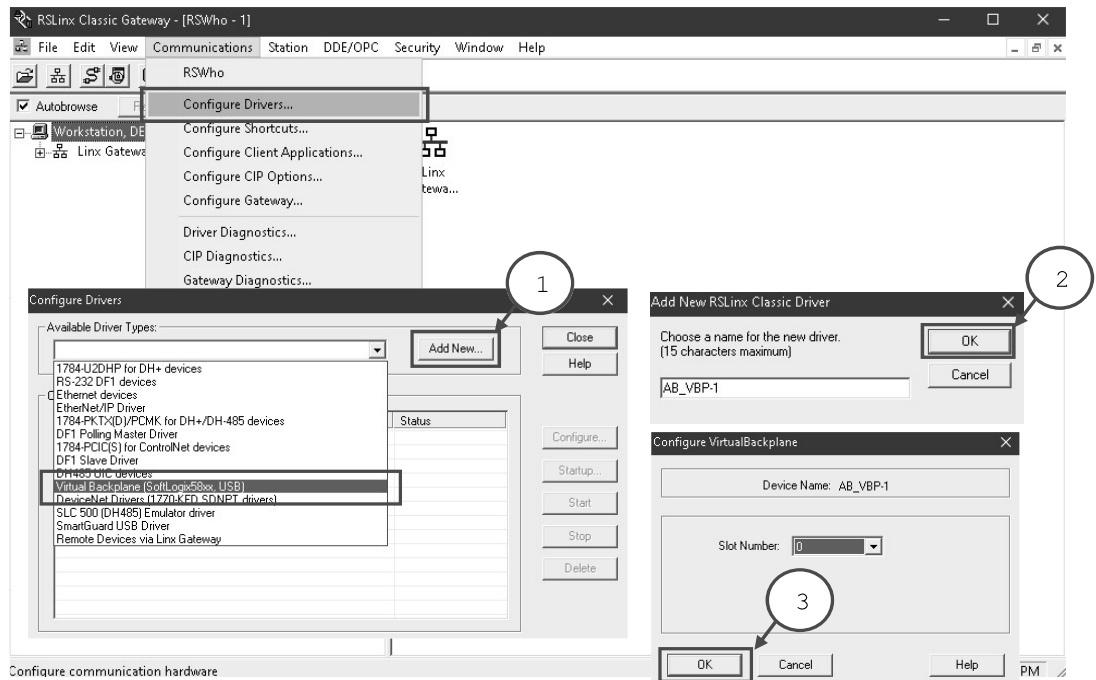


Asignaremos una CPU como se muestra en la siguiente imagen, es importante resaltar que la versión (1) debe ser consistente con la versión del programa desarrollado en STUDIO 5000.

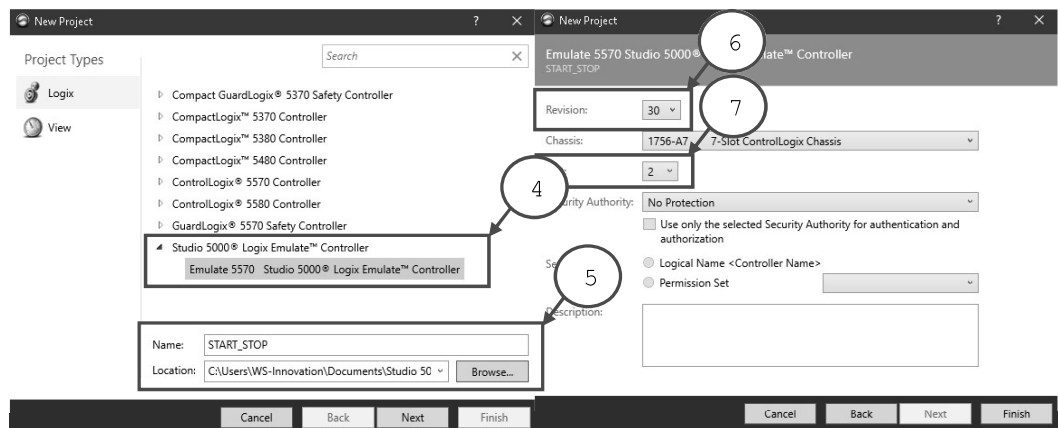


Posteriormente, crearemos dos módulos de señales uno para entradas en la posición 3 y uno para salidas en la posición 4, colocando su etiqueta respectivamente (2): VIRTUAL_INPUTS y VIRTUAL_OUTPUTS.


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	 <p style="font-size: small;">Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p> <p style="font-weight: bold; font-size: small;">UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	---

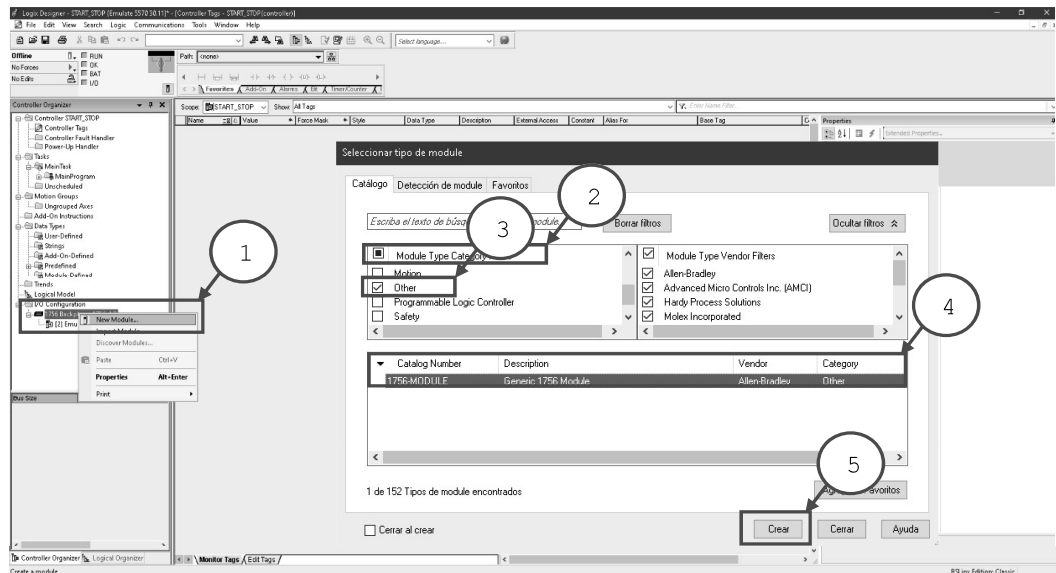


Para garantizar la comunicación y poder cargar programas a la unidad de control es necesario configurar RSLinx, agregando una nueva configuración (1), seleccionando VIRTUAL BACKPLANE, dando clic en OK (2-3) y finalmente cerramos la ventana en CLOSE, sin cerrar la ventana de RSLinx.



Una vez configurado la unidad de control virtual procederemos a crear un programa nuevo (5) en Studio 5000 y configurarlo para cargarlo en el emulador de Studio 5000 Logix Emulate Controller (4). La versión del programa (6) debe coincidir con la versión del emulador y la posición en la que se encuentra en el emulador (7), y dar clic en FINISH, para iniciar el programa Studio 5000 y comenzar a crear el programa.


Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	 UNIVERSIDAD NACIONAL DE COLOMBIA
---	---

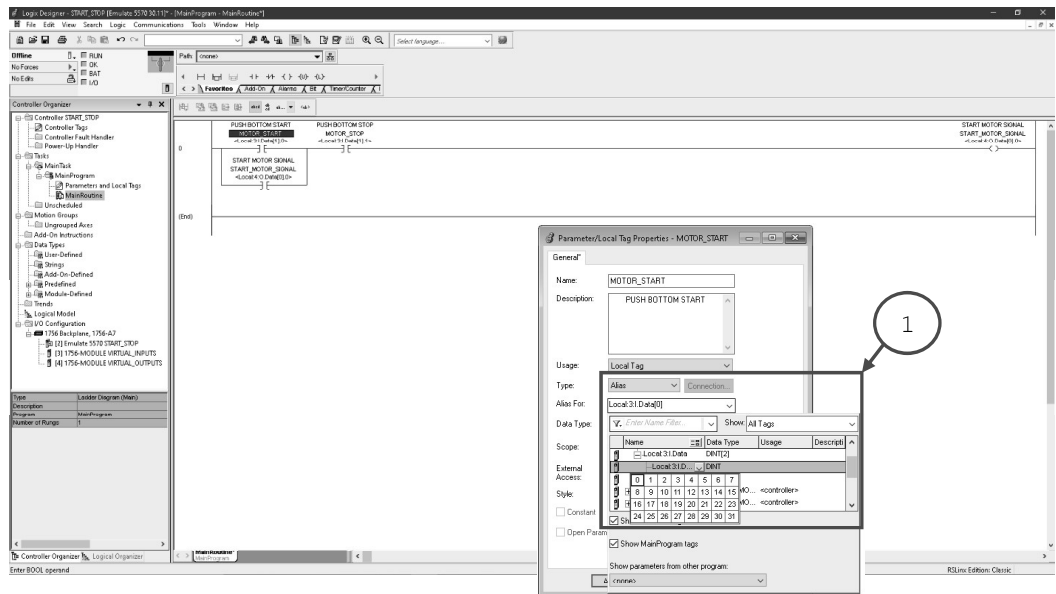


Agregar los módulos de entradas y salidas con clic izquierdo sobre 1756 Backplane, 1756-A7 y agregar nuevo módulo (1), deseleccionaremos todos los módulos (2) para filtrar la búsqueda y seleccionar únicamente Other (3), con el número de catálogo 1756MODULE (4) y dar clic en crear (5).

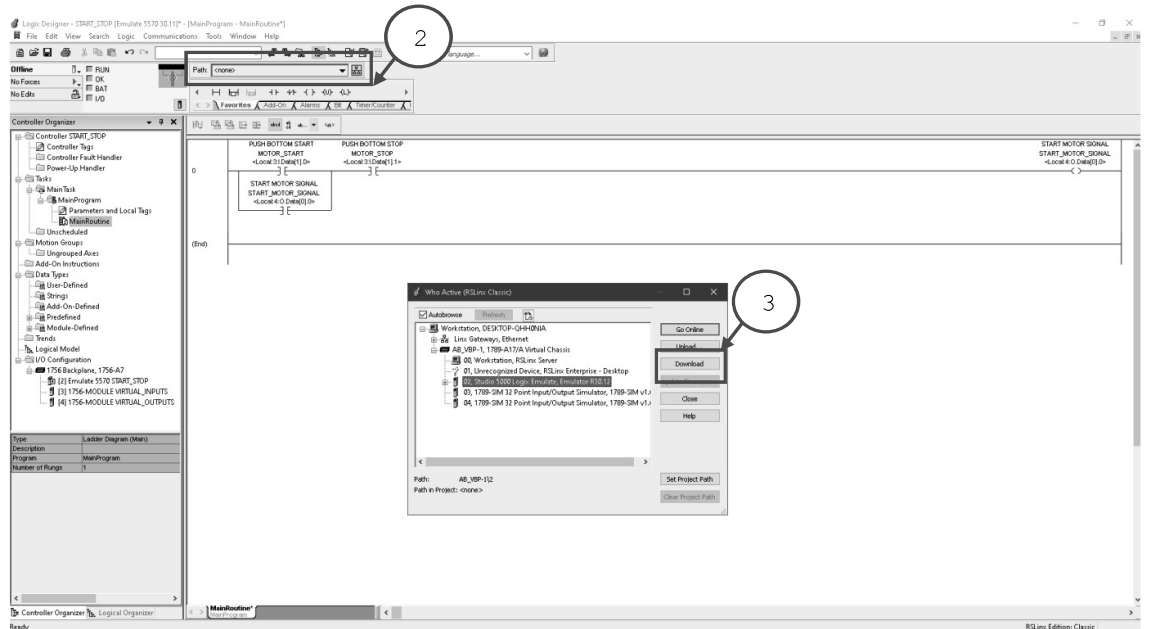


Repetir los pasos 1-9 para crear el módulo de salidas, cambiando el nombre (10) y la posición (11).


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	 <p style="margin: 0;">Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p> <p style="margin: 0; font-weight: bold;">UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	---

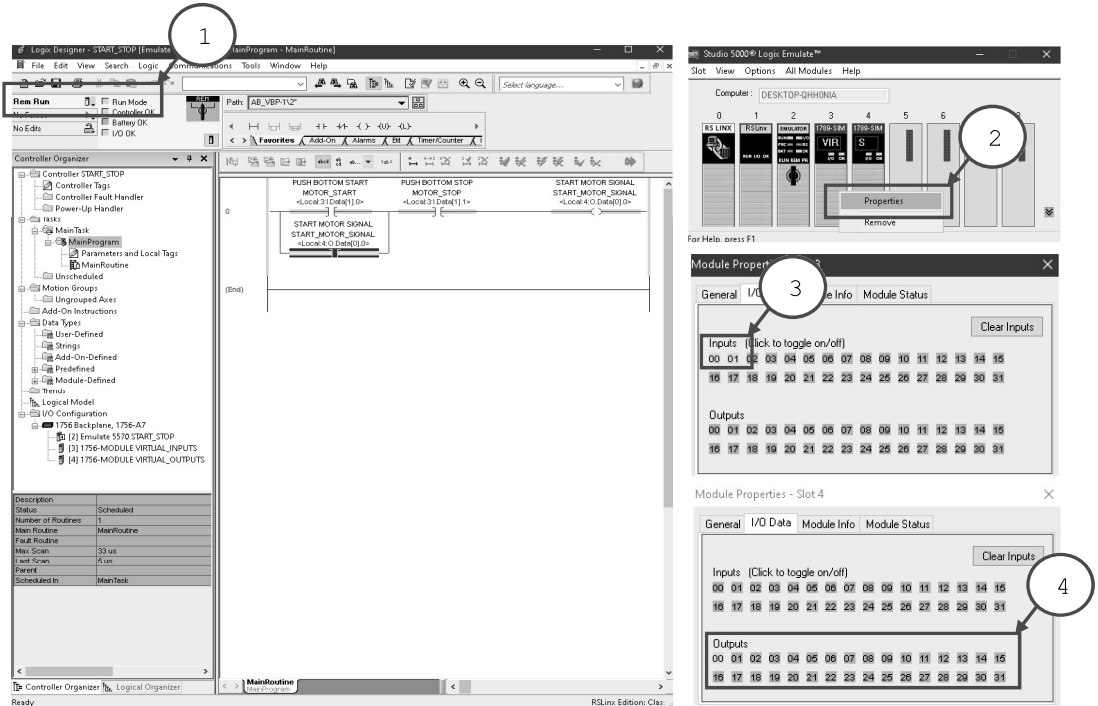


Desarrollar un programa START_STOP asignando los TAG como alias (1) y apuntando Local:3:I.Data[1] para las entradas y Local:4:O.Data[0] para las salidas.



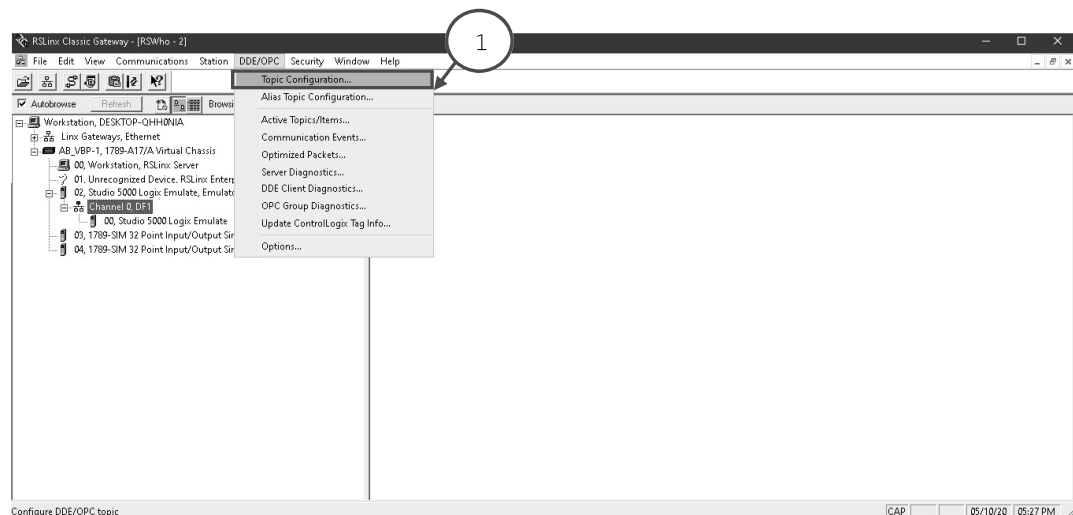
Una vez listo y configurado el programa de START_STOP, cargar el programa al CPU asignando la ruta (2), mediante el botón de descargar (3) en la ventana emergente.

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	 <p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p> <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	---




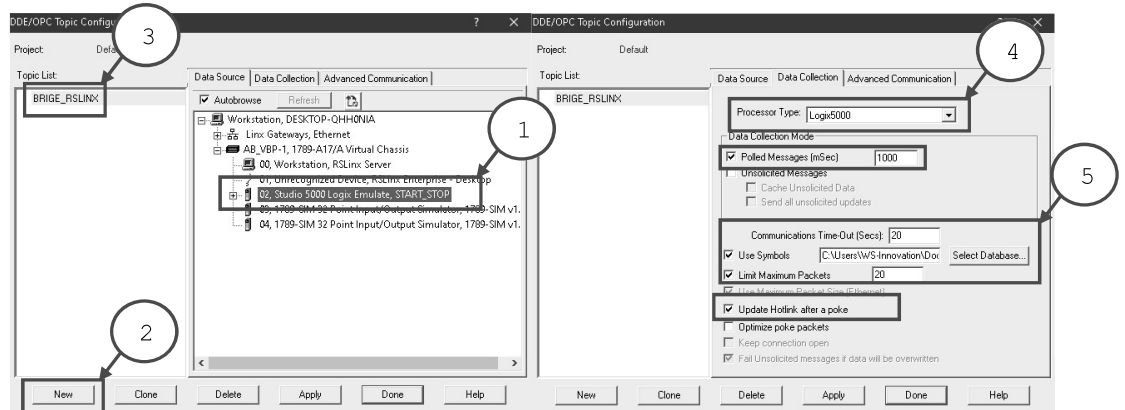
Es importante validar las comunicaciones y ejecución del programa, colocando en modo RUN y desplegando las propiedades (2) con clic derecho sobre cada módulo respectivamente: entradas (3) y salidas (4). Accionar la entrada dando clic sobre las entradas en la ventana (3) o mediante TOGGLE BIT.

6.3 CONFIGURACIÓN DE LAS COMUNICACIONES INDUSTRIALES VIRTUALES OPC DA-OPC UA:

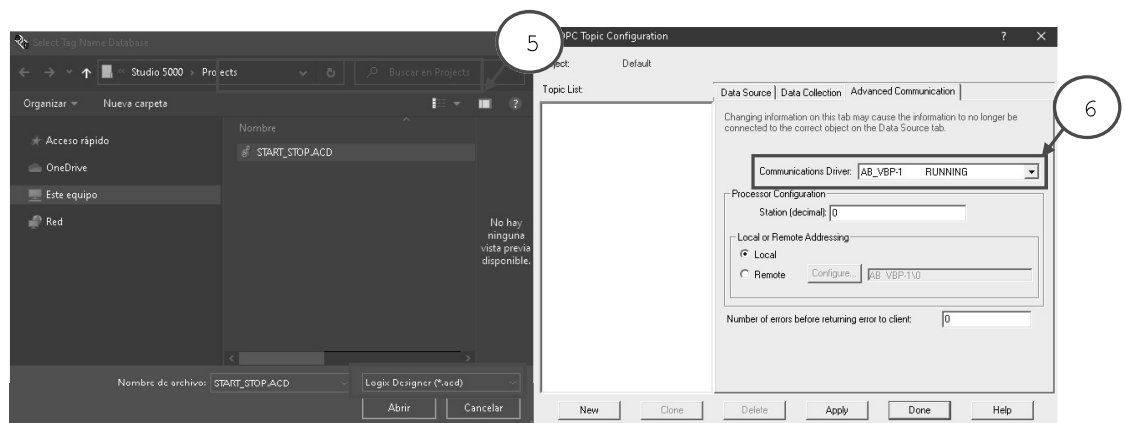


Para comunicar la unidad de control virtual con el exterior debemos configurar en RSLinx el servidor (1).

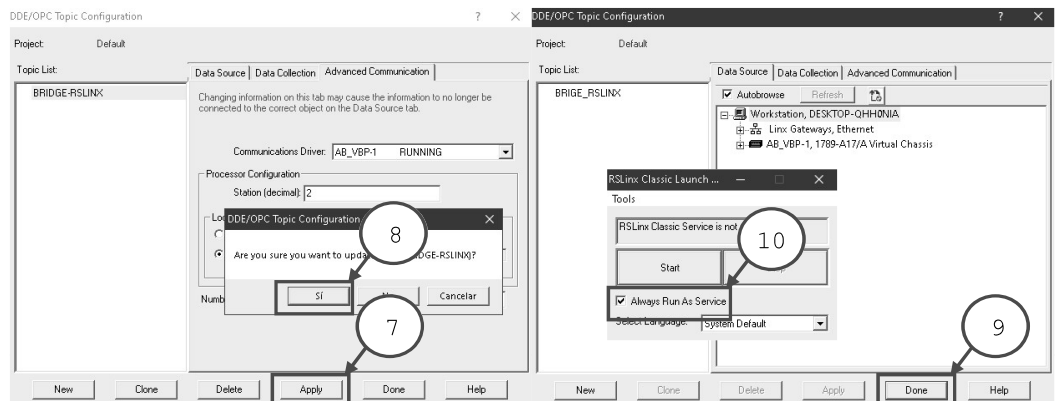
<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	 <p style="font-size: small;">Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p> <p style="font-weight: bold; font-size: small;">UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	---




Crear un nuevo tópico (2) seleccionando la unidad de control (1) y asignar un nombre (3) y configurar.



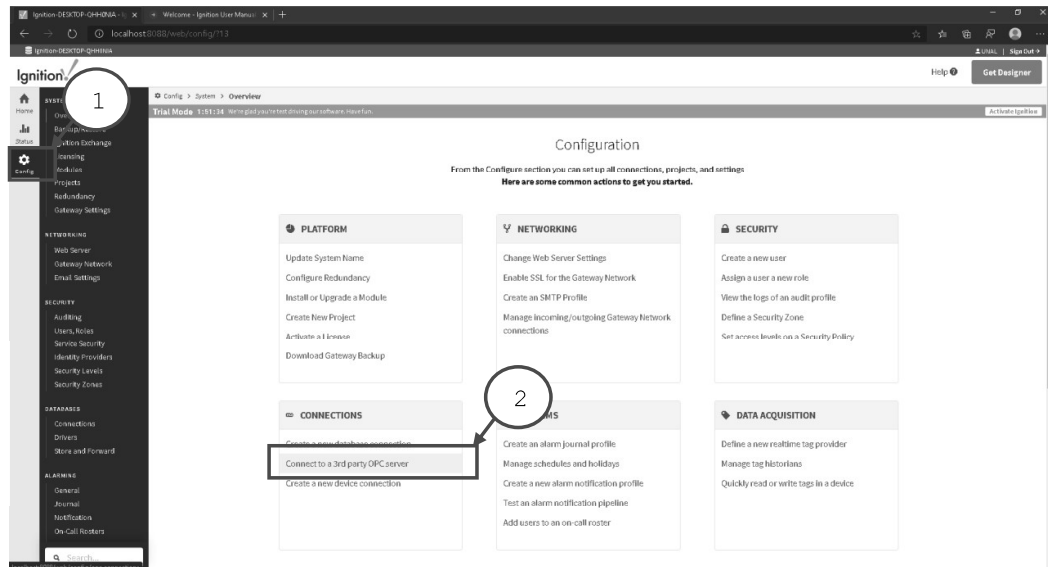
Seleccionar base de datos (5) archivo. ACD de Studio 5000 y en comunicaciones el driver AB_VBP-1 (6).



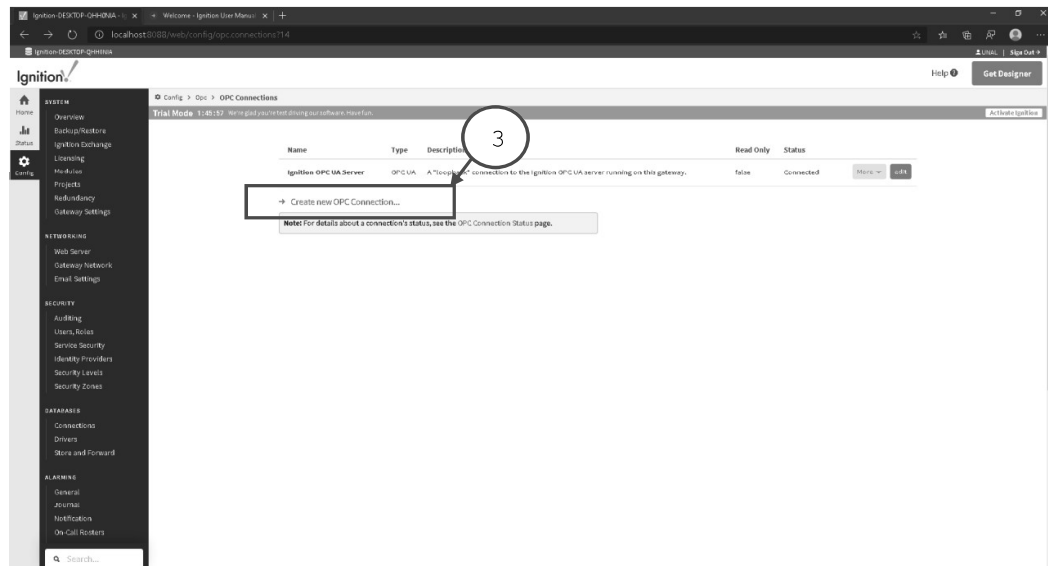
Para finalizar, debemos aplicar los cambios (7), aceptar la actualización (8) y finalizar la configuración (9). Ejecutar el **RSLinx Classic Launcher** detener el servicio y seleccionar always run as service e iniciar.

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>	 <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--	---


En este punto crearemos una nueva conexión, para conectar IGNITION de Inductive Automation con RSLinx de Rockwell y poder tener acceso a los datos con sus respectivas etiquetas. Posteriormente, estas podrán ser leídas, editadas y/o comandadas desde una interfaz SCADA.



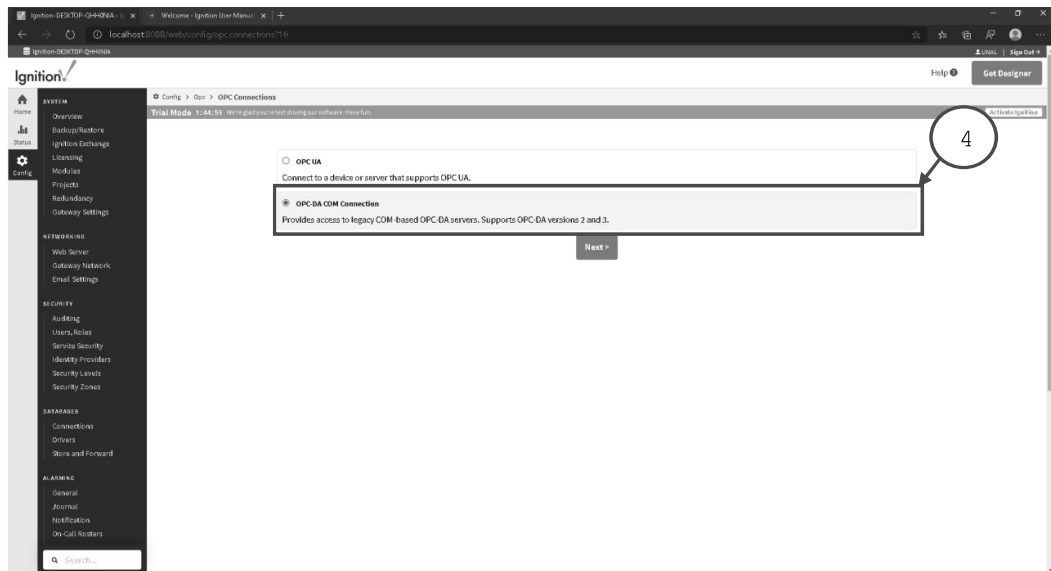
Como primer paso una vez dentro de IGNITION nos dirigimos a la pestaña de configuraciones (1), en la sección de conexiones ingresaremos a la opción conectar a un tercero por servidor OPC (2).



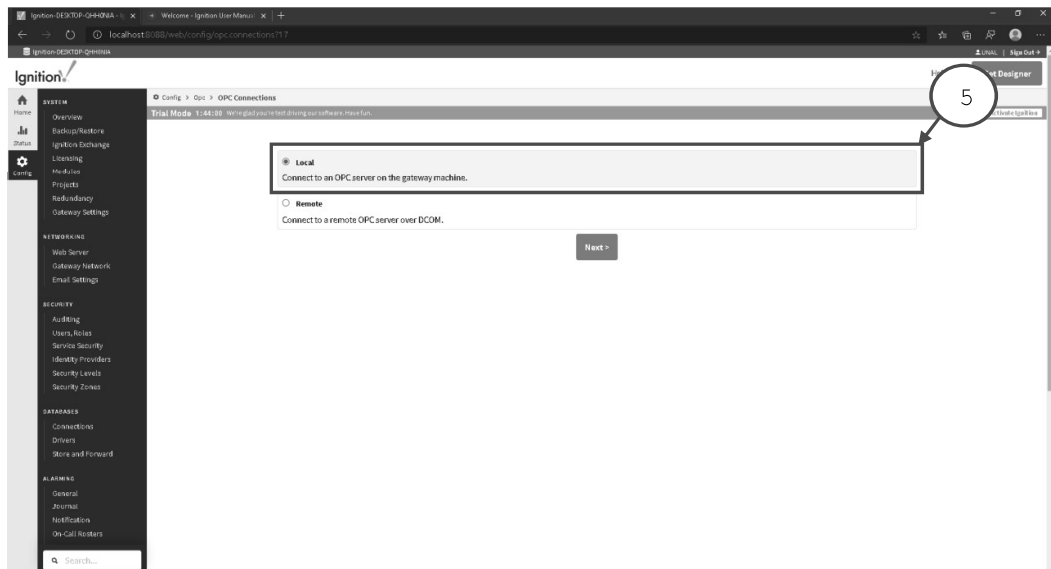
Para configurar el cliente OPC DA en IGNITION seguir los pasos 1-8 sin modificar lo que no se indica.

Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	 <p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p> <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
---	--

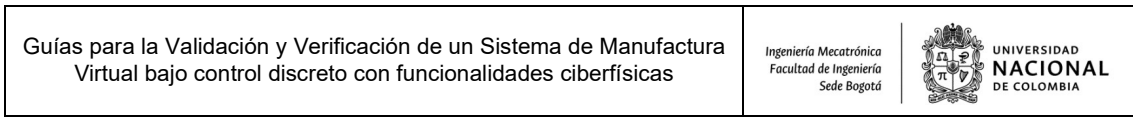
En este paso es posible crear dos tipos de conexiones OPC: una donde los elementos a conectar son de Arquitectura Unificada llamada UA, o si solo permite un Acceso a los Datos denominada DA (4).



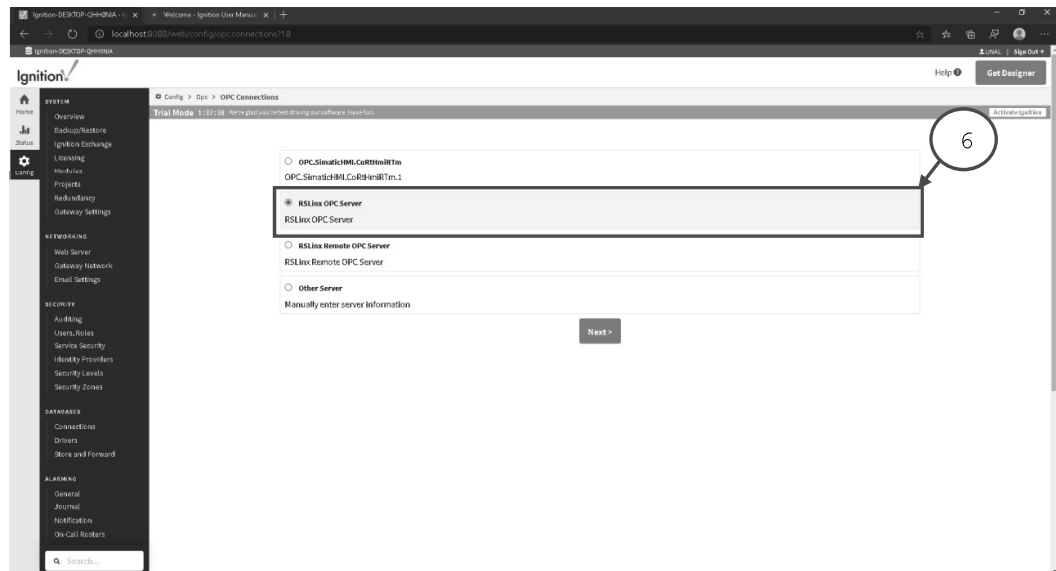
Estas conexiones pueden ser de dos tipos: Locales (5) cuando el servidor y los clientes se encuentran en el mismo equipo o de tipo remoto cuando se encuentran en una red fuera del equipo.



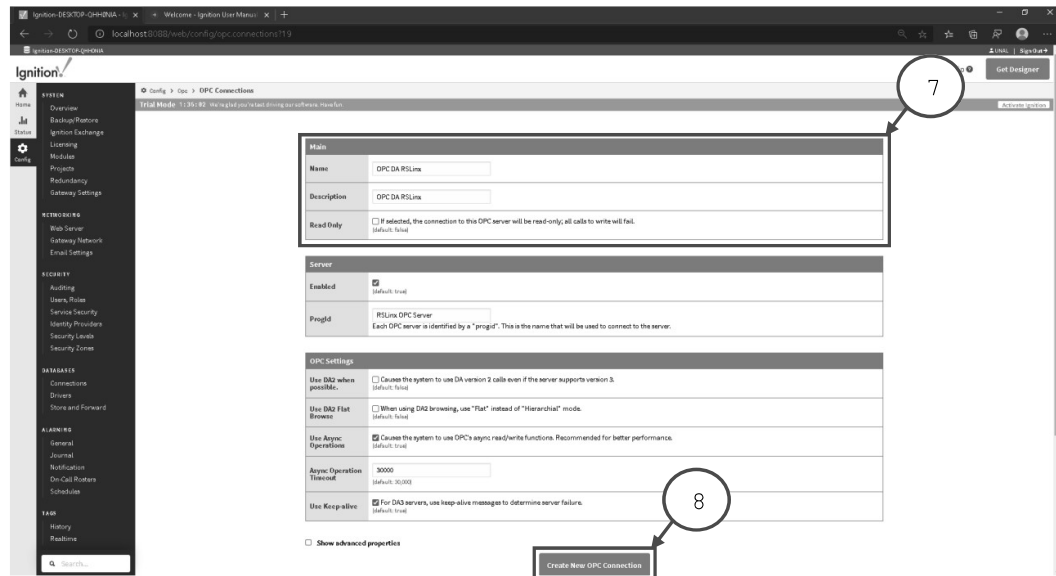
Este tipo de conexiones nos ayudara a entender la arquitectura de comunicaciones entres sistemas que se encuentran en un mismo equipo o en una red para comunicarlos entres si y compartir información la cual servirá para tomar decisiones y realizar acciones.




En este paso podremos observar todos los servicios locales que podemos acceder desde IGNITION, seleccionando el servidor RSLinx OPC Server (6) para tener acceso a sus datos.



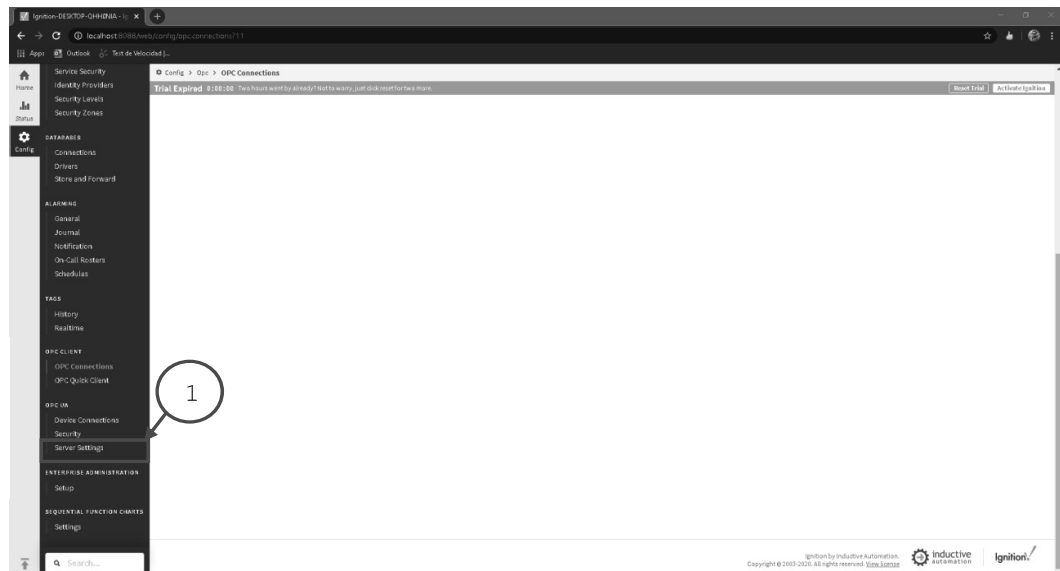
A cada conexión que se realice con IGNITION es necesario asignarle un nombre para identificarlo posteriormente. Importante usar un nombre (7) de fácil identificación que nos permita reconocer que tipo de datos intercambiamos para su uso en una HMI o sistema SCADA.



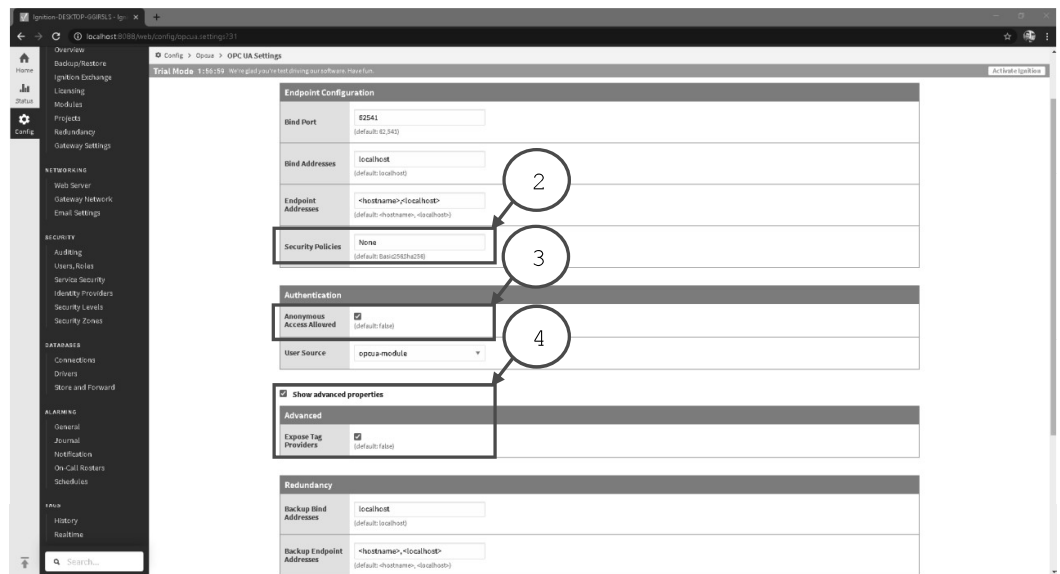
A este punto aceptamos crear la nueva conexión (8). Posteriormente, nos llevara al panel principal donde podremos observar el estado de la conexión y sus configuraciones.

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>  <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--

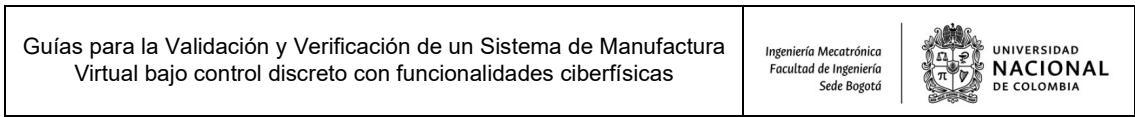
Para lograr la conexión entre NX y el servidor OPC UA de IGNITION debemos realizar ajustes de seguridad de las comunicaciones.



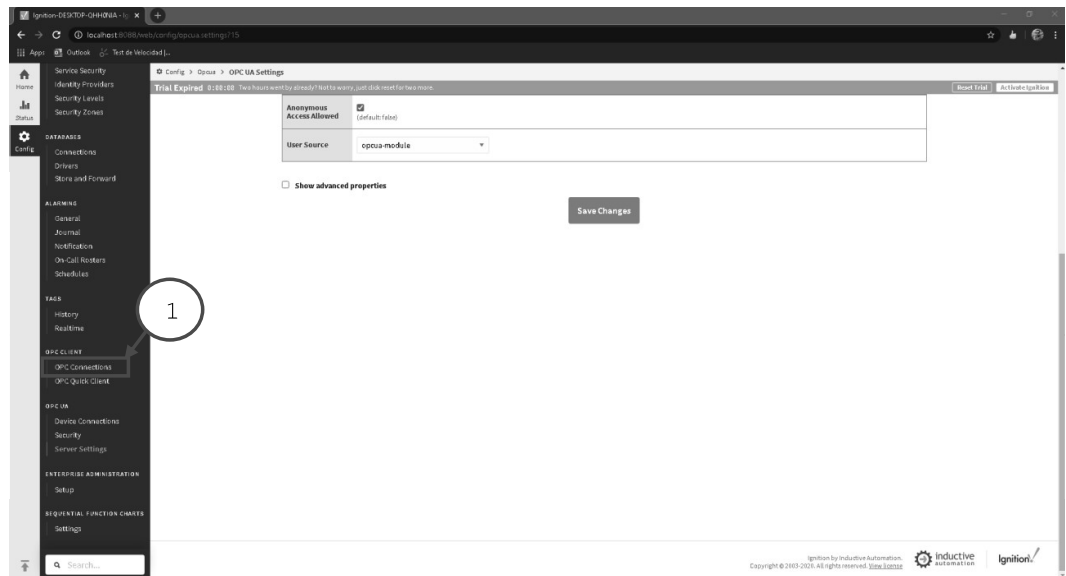
Como primer paso debemos entrar en la pestaña de configuraciones y seleccionar las opciones del servidor (1), donde se puede configurar las propiedades como puerto, IP y seguridad de los datos.



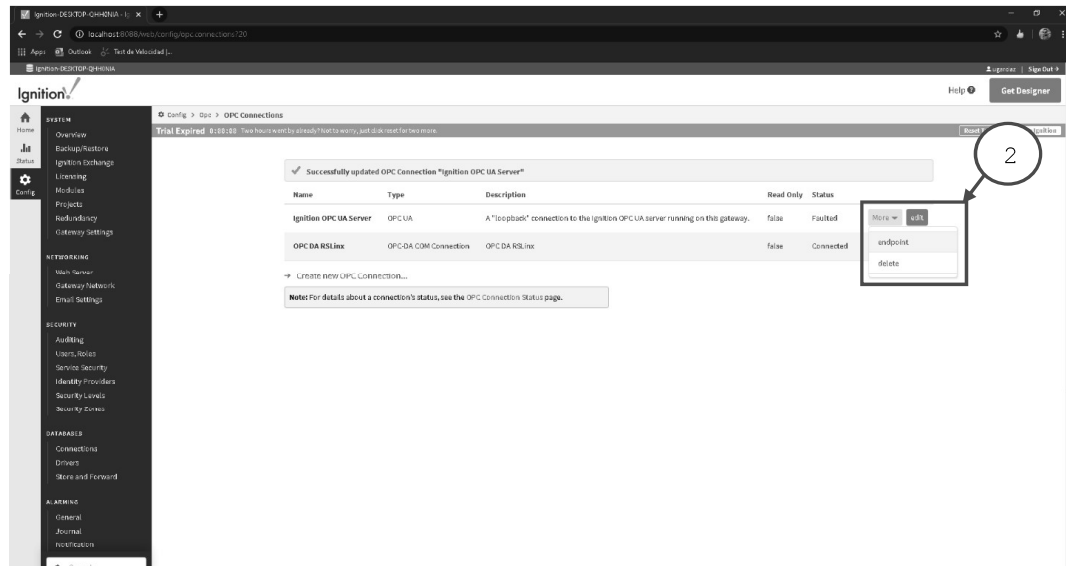
En la ventana de configuraciones tres elementos son importantes de activar, las políticas en **None**, activar el acceso anónimo (2) y activar exponer los datos (3) para que los vea NX, finalmente damos clic en guardar cambios.




Para verificar el estado de la conexión damos clic en conexiones OPC (1) donde podremos ver si las conexiones se encuentran en falla y poder editar sus parámetros.



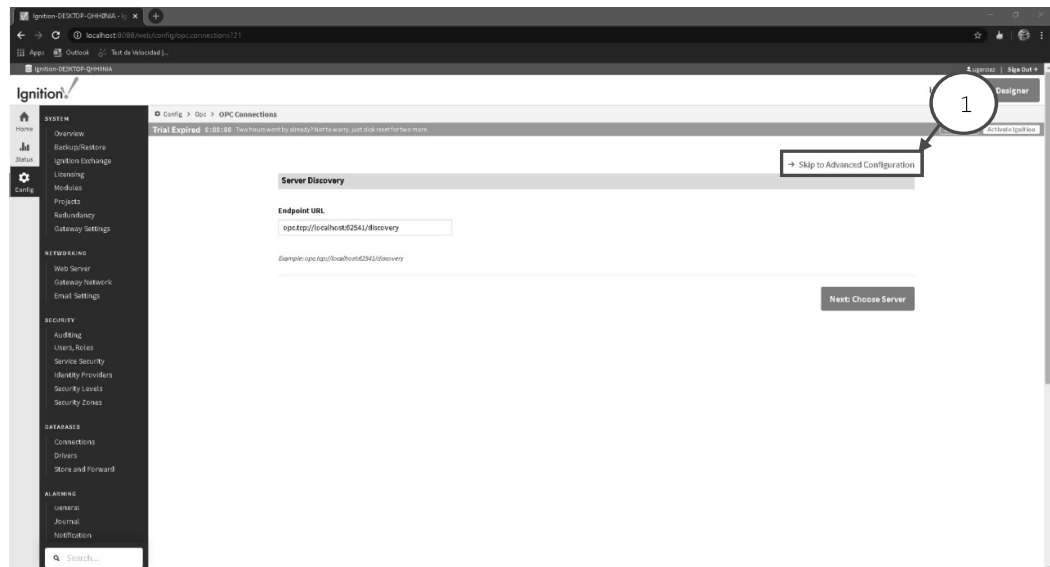
En este caso se puede observar que la conexión redundante del servidor de Ignición OPC UA se encuentra en falla por ello debemos realizar ajustes en términos de seguridad.



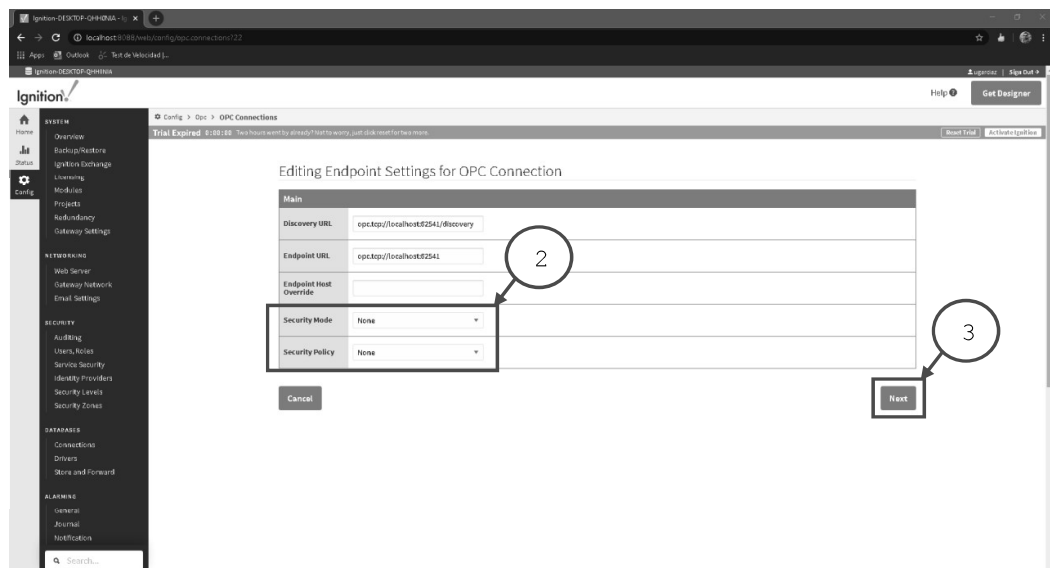
Editaremos la configuración del ENDPOINT del servidor OPC UA dando clic en más (2) antes del botón editar.

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>  <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--


Para modificar de forma completa los parámetros del servidor daremos clic en saltar a configuración avanzada (1) y poder editar todos los parámetros del ENDPOINT.



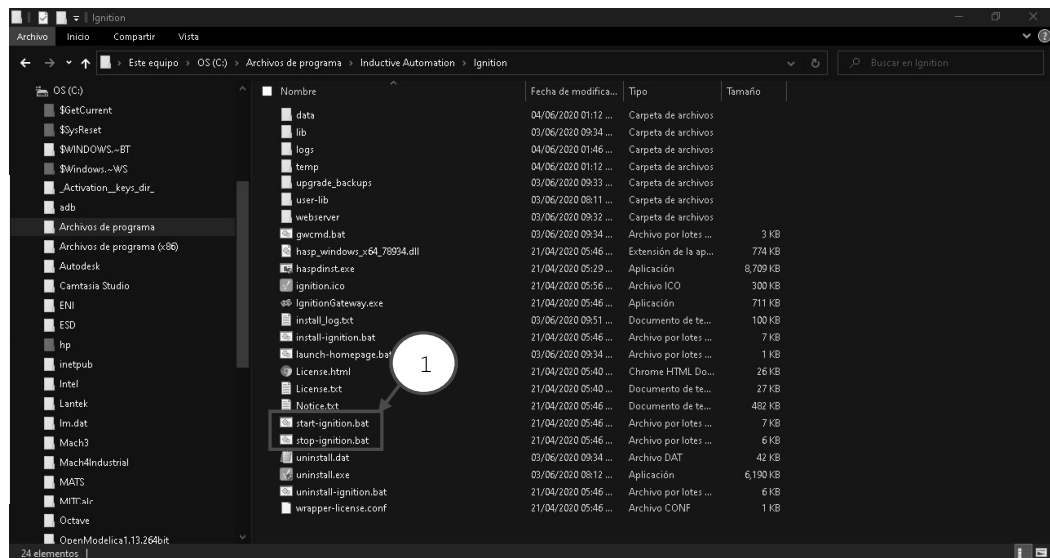
Ya que hemos modificado la seguridad en el servidor OPC UA es necesario que dichos parámetros sean consistentes con el ENDPOINT por ello es necesario cambiar los atributos de modo y políticas.



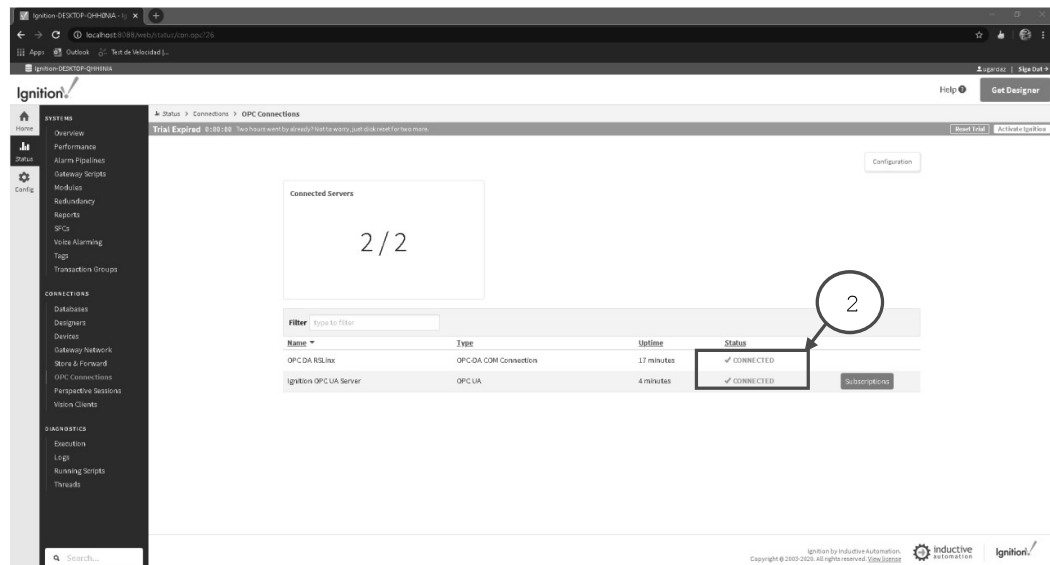
El modo de seguridad (2) debe ser seleccionado la opción **NONE** al igualmente la opción de políticas de seguridad (2). Dichas opciones habilitan la posibilidad de ver los datos desde cualquier cliente sin encriptación de archivos.

Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá	 UNIVERSIDAD NACIONAL DE COLOMBIA
---	---	---


Cada vez que se realicen cambios en el servidor es recomendable hacer un reinicio mediante las funciones de START/STOP que pueden ser encontradas en: C:\Program Files\Inductive Automation\Ignition (1).



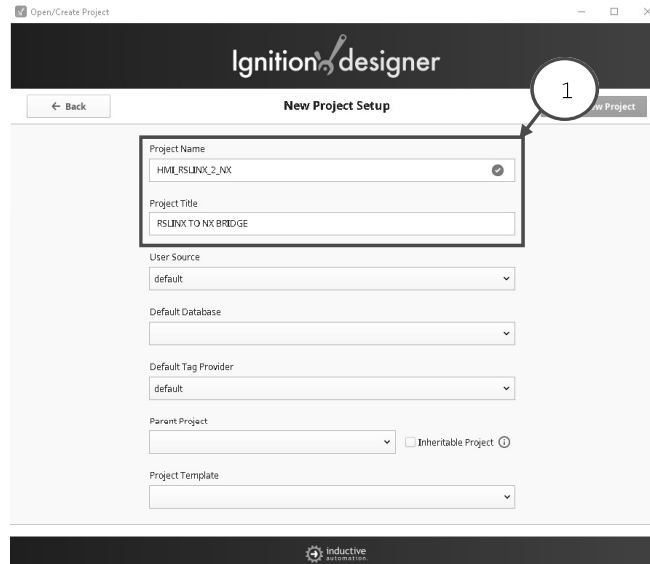
Antes de comenzar el proceso es importante realizar el SIGN-OUT de IGNITION y posteriormente dar clic en el archivo STOP **indicando que inicie con privilegios de administrador** clic derecho, una vez concluya el proceso y la ventana se cierre, iniciamos el servicio con el archivo START de la misma manera.



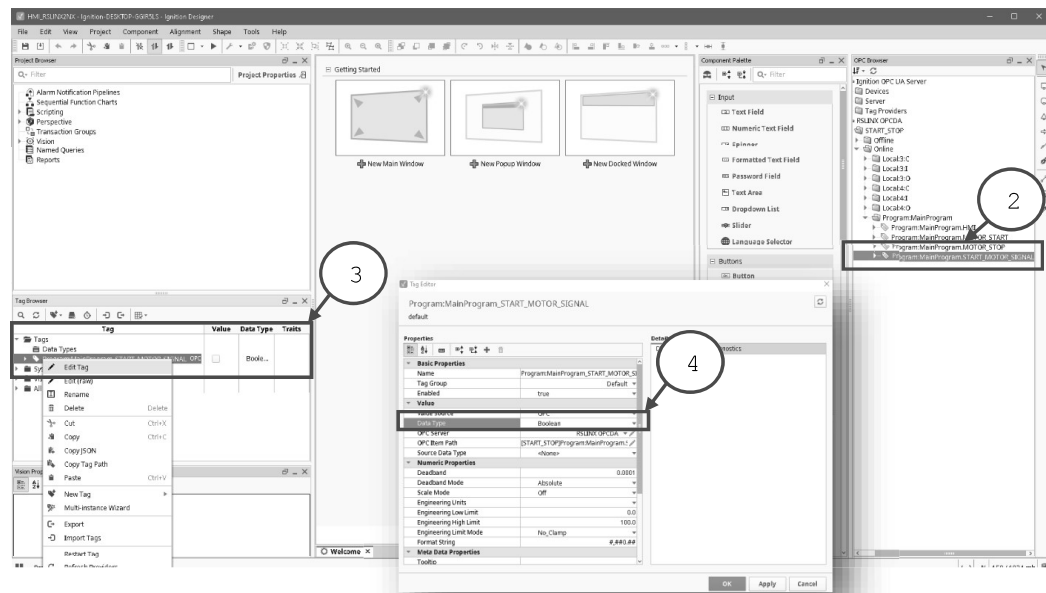
Una vez concluya el reinicio ingresamos nuevamente al GATEWAY de IGNITION donde podremos observar que los servicios se encuentran en orden y conectados (2).

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	
--	---


Para que los datos queden visibles entre el cliente OPC DA de RSLINX y el servidor OPC UA es necesario crear un archivo de INGNITION DESIGNER el cual nos servirá para crear HMI.



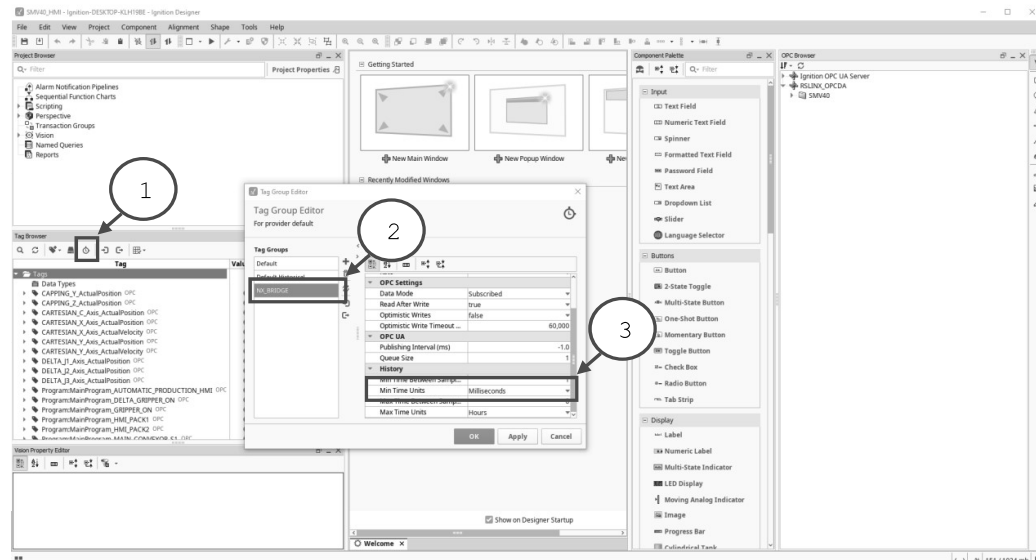
A este Proyecto lo llamaremos HMI_RSLINX_2_NX (1) y no modificaremos ningún parámetro adicional dejando los valores por defecto que el asistente predetermina del proyecto.



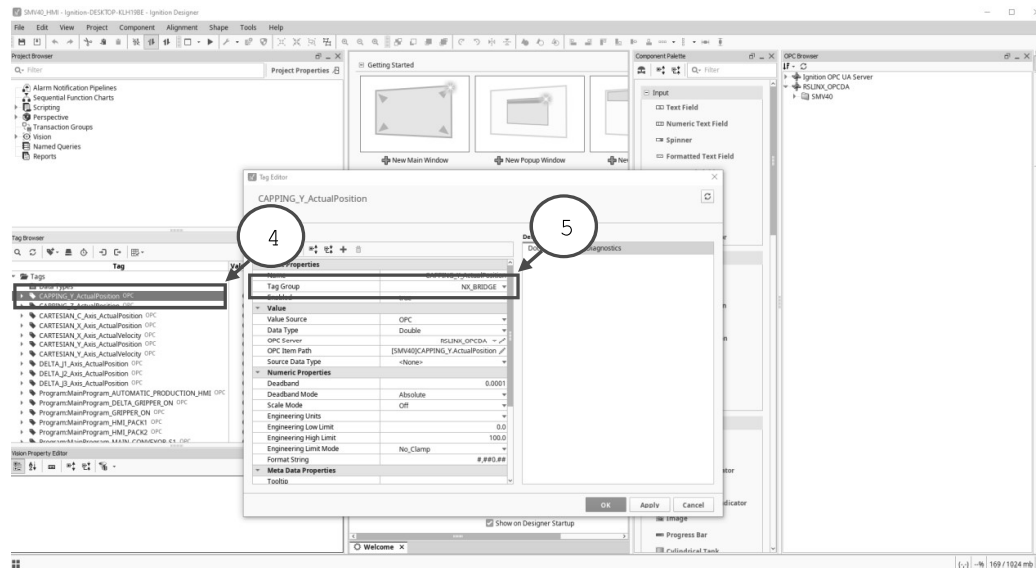
Una vez en el proyecto arrastramos los tags (2) que queremos exponer a la carpeta Tags (3) independientemente de que no sean usados en el HMI, no olvidar verificar que el tipo de dato corresponda al que requerimos. en este caso hay que asegurar que sea de tipo BOOL.

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>	 <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--	--


Para que los datos se puedan muestrear a una tasa mas frecuente es necesario habilitar un nuevo grupo que este configurado en una tasa de muestreo más pequeña.



En la opción grupo de tags (1) damos clic ingresando a la ventana emergente en ella en el símbolo de + agregamos un nuevo grupo al cual asignaremos un nombre y en la sección del lado derecho al final del menú encontraremos una opción de unidades para el tiempo mínimo de muestreo en este caso milisegundos.



Una vez creado el grupo en cada tag que se requiera debemos entrar al editor (4) y asignar el grupo (5).

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>  <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--

6.4 LICENCIAMIENTO NX MCD:

Lo primero que se tiene que hacer es verificar la comunicación física entre la estación de trabajo PC y el gestor de licencias de NX. Verifique que el cable de red efectivamente está conectado en el PC y que se encuentra conectado a la red UNAL. Posteriormente podemos realizar un ping, para verificar la conexión física entre el PC y el servidor de licencia de NX:

```

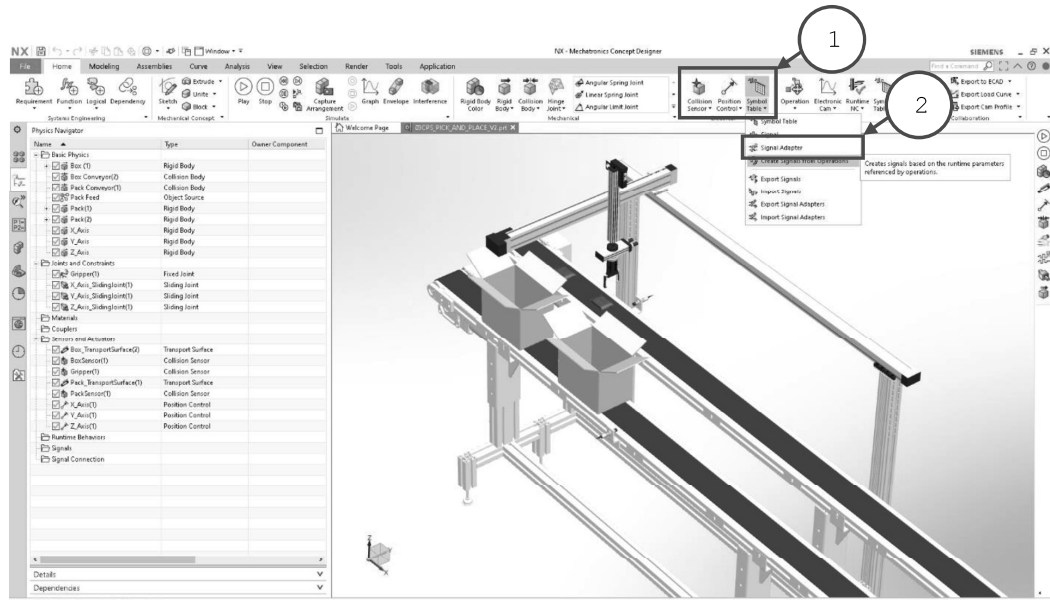
Microsoft Windows [Versión 10.0.17134.765]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Users\WS-Innovation>ping 168.176.36.140


Haciendo ping a 168.176.36.140 con 32 bytes de datos:
Respuesta desde 168.176.36.140: bytes=32 tiempo=9ms TTL=126
Respuesta desde 168.176.36.140: bytes=32 tiempo=11ms TTL=126
Respuesta desde 168.176.36.140: bytes=32 tiempo=313ms TTL=126
Respuesta desde 168.176.36.140: bytes=32 tiempo=5ms TTL=126

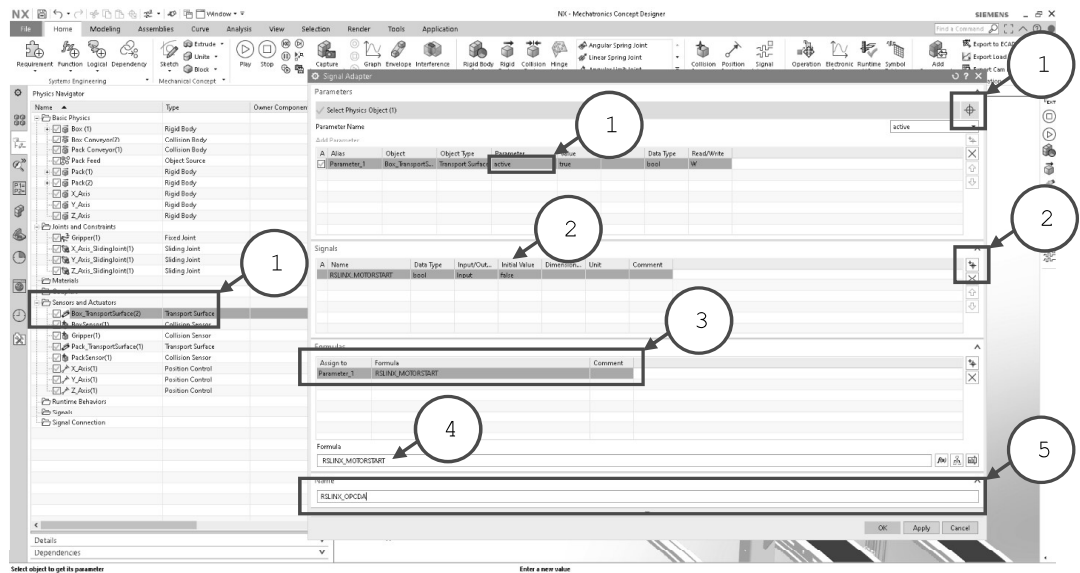
Estadísticas de ping para 168.176.36.140:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
            (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 5ms, Máximo = 313ms, Media = 84ms
    
```

6.5 COMUNICACIÓN ENTRE EL SERVIDOR OPC UA Y EL SISTEMA DE MANUFACTURA VIRTUAL:

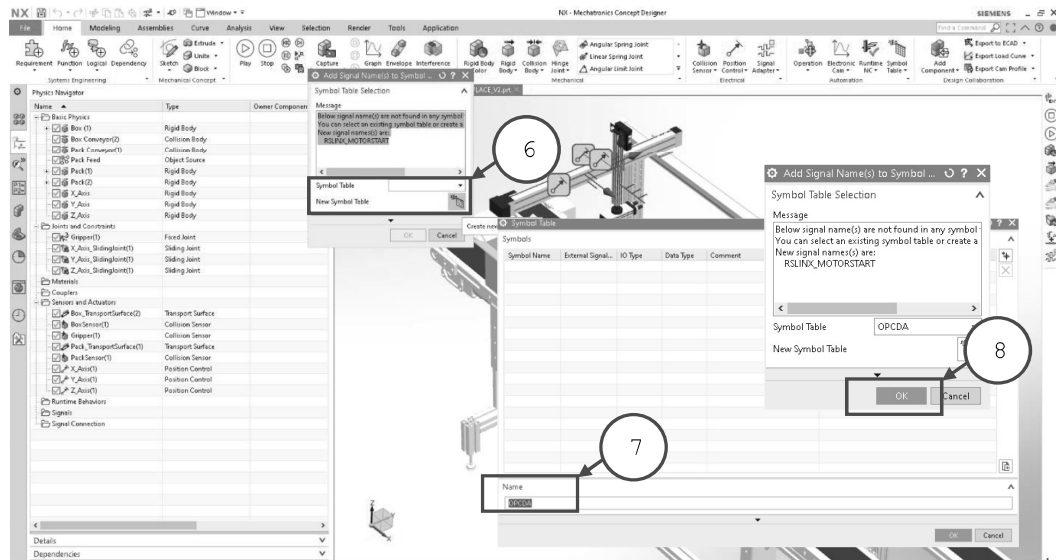


Para validar y verificar la lógica de control del programa realizado en Studio 5000 es necesario ir al grupo eléctrico (1) de la pestaña de HOME y seleccionar el comando adaptador de señal (2), para crear una tabla de señales a conectar con el servidor OPC DA. Deben declararse todas las entradas y salidas requeridas para ser mapeadas en NX MCD y RSLINX SERVER.


Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá	 UNIVERSIDAD NACIONAL DE COLOMBIA
--	---	---

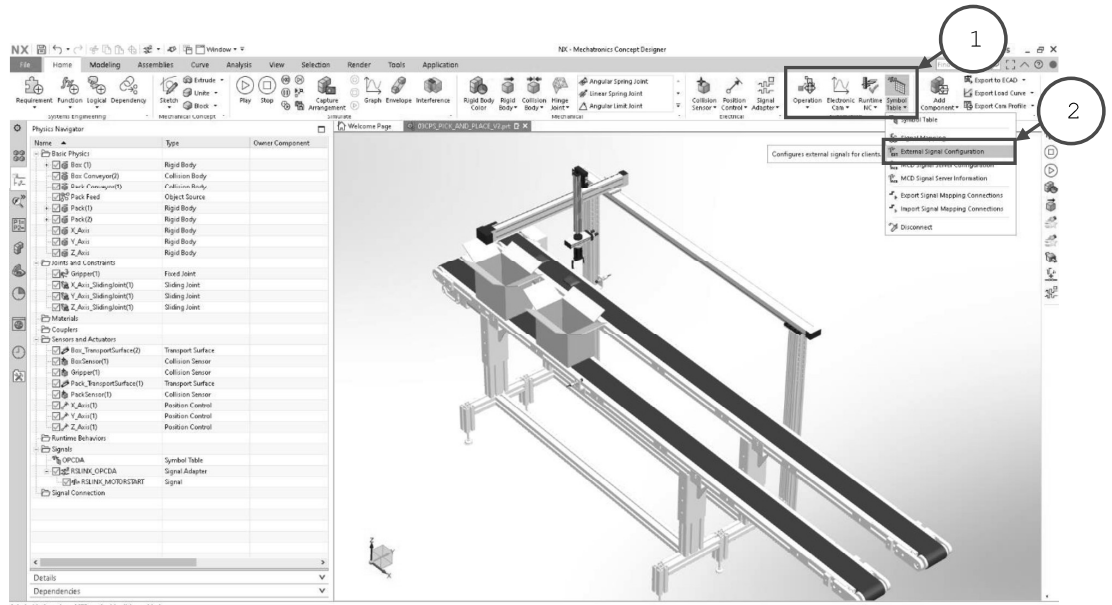


En el adaptador de señales seleccionamos el objeto a controlar, la superficie de transporte de las cajas (1), verificar que la opción de parámetros se encuentre en ACTIVE (1). Posteriormente, creamos una señal (2) y seleccionamos que sea tipo de entrada INPUT y se encuentre en FALSE para que inicia hasta que le enviemos señal. En la sección de formulas debemos enlazar el parámetro con la señal (3-4), escribir el nombre de la señal y dar ENTER. Finalmente, asignar un nombre para el adaptador (5).

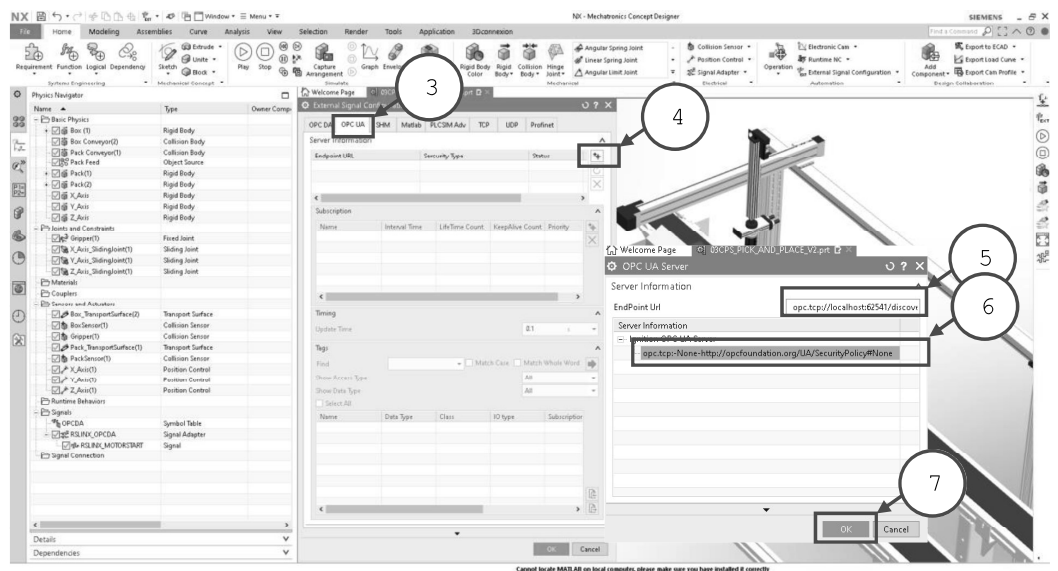


A continuación, creamos la tabla de símbolos siguiendo los pasos (6-8) asignando un nombre a la tabla.


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	 <p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p> <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	---

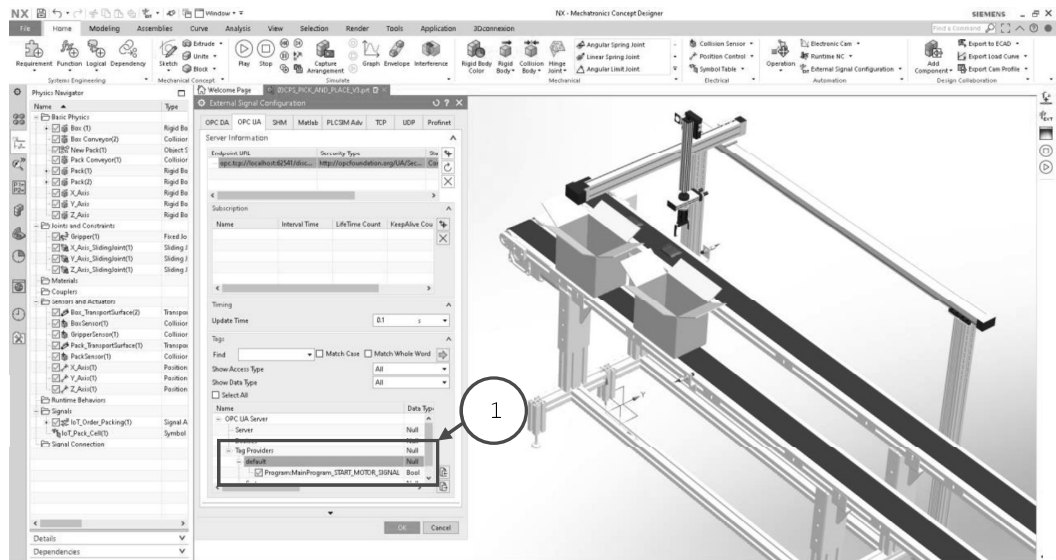


Es necesario ir al grupo de automatización (1) de la pestaña de HOME y seleccionar el comando configuración externa de señales (2), para conectar NX MCD con el servidor OPC UA.

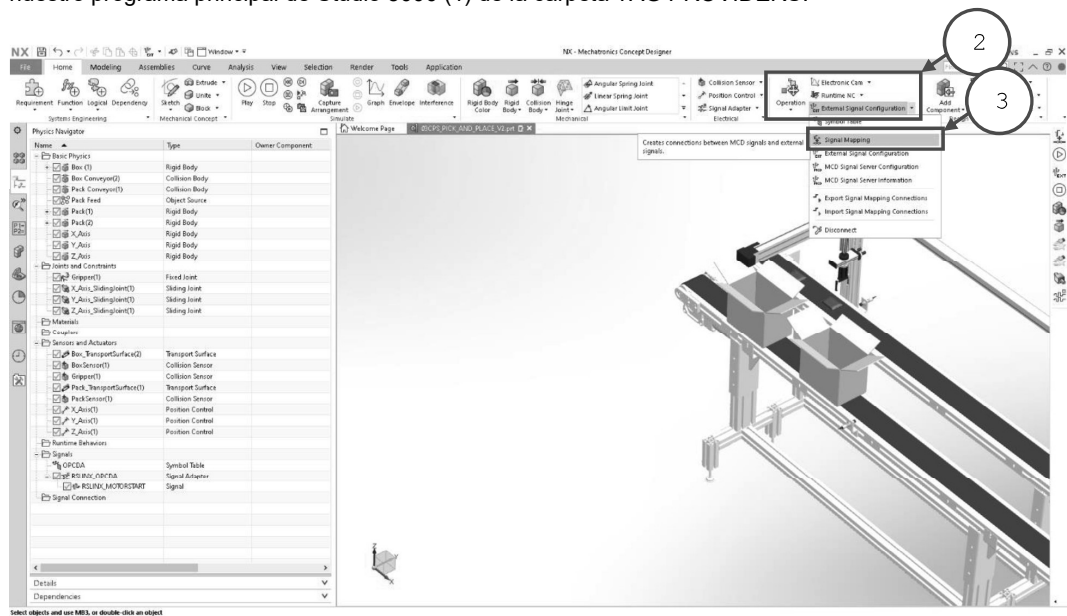


Existen varios tipos de adaptadores de señal para conectar NX MCD, para este caso seleccionamos la pestaña (3) OPC UA y creamos uno nuevo (4). En la ventana emergente ingresamos el ENDPOINT **opc.tcp://localhost:62541/Discovery**, seleccionamos el enlace (6) revisando que este en NONE la seguridad y aceptamos.


<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	<p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p>	 <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	--	--

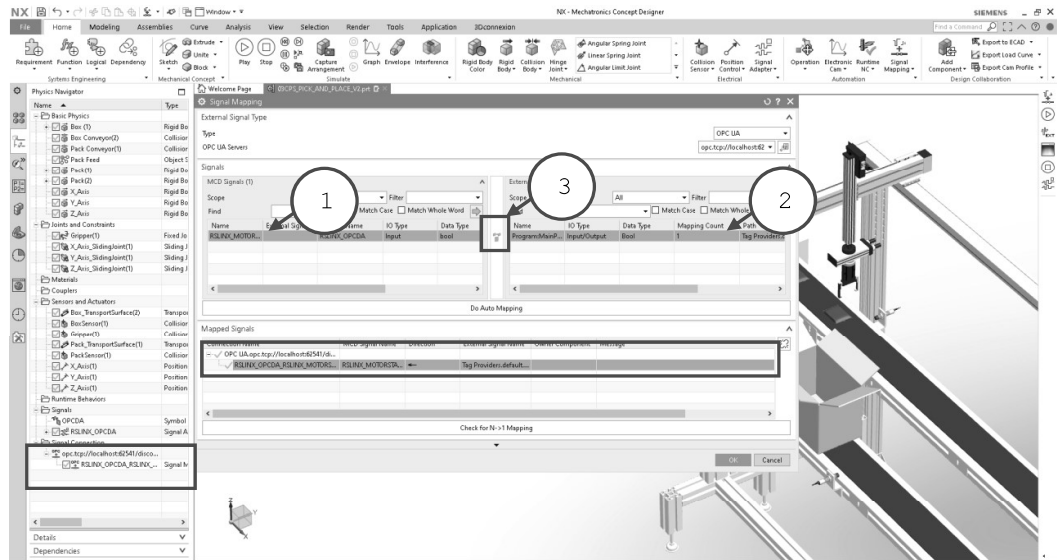


Una vez se cargue el servidor OPC UA de IGNITION buscamos las variables a mapear y las activamos para que sean visible en NX MCD. Para este caso particular activamos la señal de salida del motor de nuestro programa principal de Studio 5000 (1) de la carpeta TAG PROVIDERS.

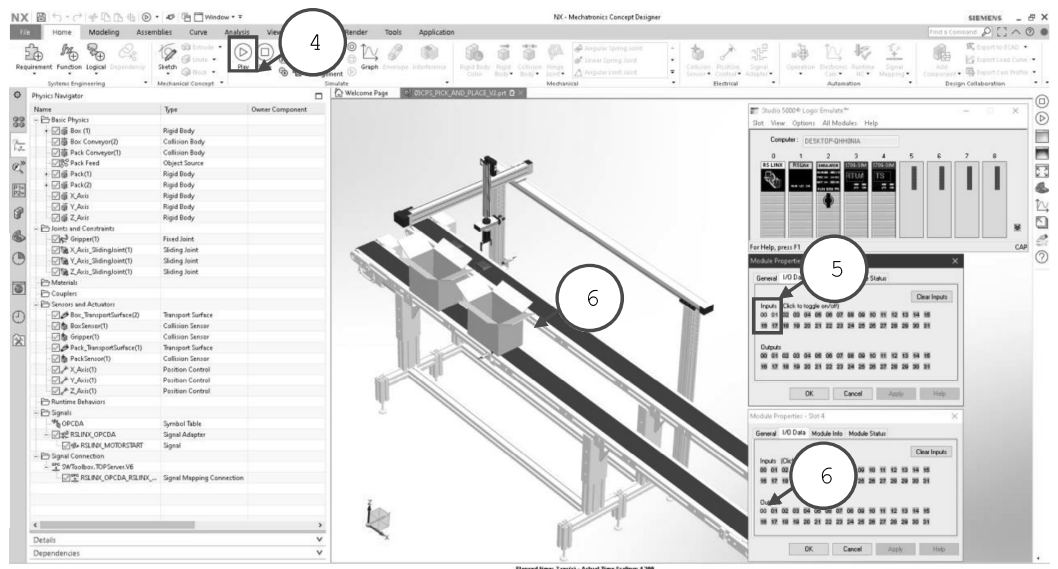


Para mapear señales en NX MCD, encontraremos en el grupo de Automatización (2) el comando Mapeo de Señales (3), para enlazar variables de la unidad de control con las señales del sistema de manufactura virtual.

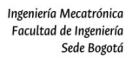

<p>Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas</p>	 <p>Ingeniería Mecatrónica Facultad de Ingeniería Sede Bogotá</p> <p>UNIVERSIDAD NACIONAL DE COLOMBIA</p>
--	---



En el adaptador de señales del lado izquierdo (1) seleccionar la señal a mapear de NX, en el lado derecho (2) seleccionar la variable a conectar del servidor de OPC DA y conectarlas mediante el icono (3). **Nota: es importante que las variables sean del mismo tipo de data de lo contrario no son compatibles.**



Finalmente, aceptamos el mapeo de las señales y procedemos a validar y verificar el programa de la unidad de control en el sistema de manufactura virtual. Iniciamos la simulación (4), activamos las señales (5) y la superficie de transporte se activará y moverá las cajas (6).

Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas	  UNIVERSIDAD NACIONAL DE COLOMBIA
---	---

7 BIBLIOGRAFIA:

[1] H. Vermaak, "Virtual Commissioning: A Tool to Ensure Effective System Integration," no. Mayo, 2017

8 AGRADECIMIENTOS:

En sus propósitos de mejoramiento continuo con orientación académico-industrial, el Laboratorio de Automatización de Máquinas 411-101 de la Facultad de Ingeniería – Sede Bogotá, Universidad Nacional de Colombia, se complace en poner a disposición de la comunidad académica la primera versión de las Guías para la Validación y Verificación de un Sistema de Manufactura Virtual bajo control discreto con funcionalidades ciberfísicas.

Estas guías son el resultado del trabajo final de maestría con el apoyo del Director del Laboratorio, Profesor Víctor Hugo Grisales Palacio y el docente Eduardo Barrera Gualdrón, quienes compartimos el interés común de aprovechar y diseminar el conocimiento relativo a las tecnologías de automatización y control disponibles en el Laboratorio gracias al apoyo de la Universidad Nacional de Colombia, en particular en el alcance de estas guías, en lo concerniente a aspectos de Control de Movimiento.

Un total de cuatro guías incluyendo la presente, ofrecen al participante un acercamiento al Control de Movimiento de Servomecanismos usando el software Siemens PLM NX Mechatronics Concept Designer y la Plataforma Industrial CompactLogix de Allen Bradley, facilitando un conocimiento aplicado de diversas características y configuraciones de los equipos de control industrial PAC (Controladores de Automatización Programables), Drivers Kinetix 350, así como también de los servomotores utilizados para las prácticas. A través de las guías se espera que los participantes estén en la capacidad de comprender los pasos para crear una aplicación industrial de control de movimiento desde los aspectos básicos de concepción del sistema automatizado hasta la configuración, programación y validación del sistema. Las guías han sido desarrolladas en forma progresiva por lo cual al final de estas se espera que el participante pueda programar y validar virtualmente un sistema de manufactura bajo control discreto.

Esperamos que este material sea de utilidad e interés para la comunidad académica. Agradecemos de antemano la retroalimentación de los participantes con los ajustes o mejoras que consideren pertinentes, la cuales pueden enviar al email del director.

Cordialmente,

Prof. Víctor Hugo Grisales Palacio
vhgrisalesp@unal.edu.co
Director Laboratorio de Automatización de Máquinas 411 – 101
Universidad Nacional de Colombia – Sede Bogotá
Bogotá, Mayo 01 de 2020

F

Anexo: Ladder - Rutina Automática

F. Anexo: Ladder - Rutina Automática

AUTOMATIC PRODUCTION - Ladder Diagram

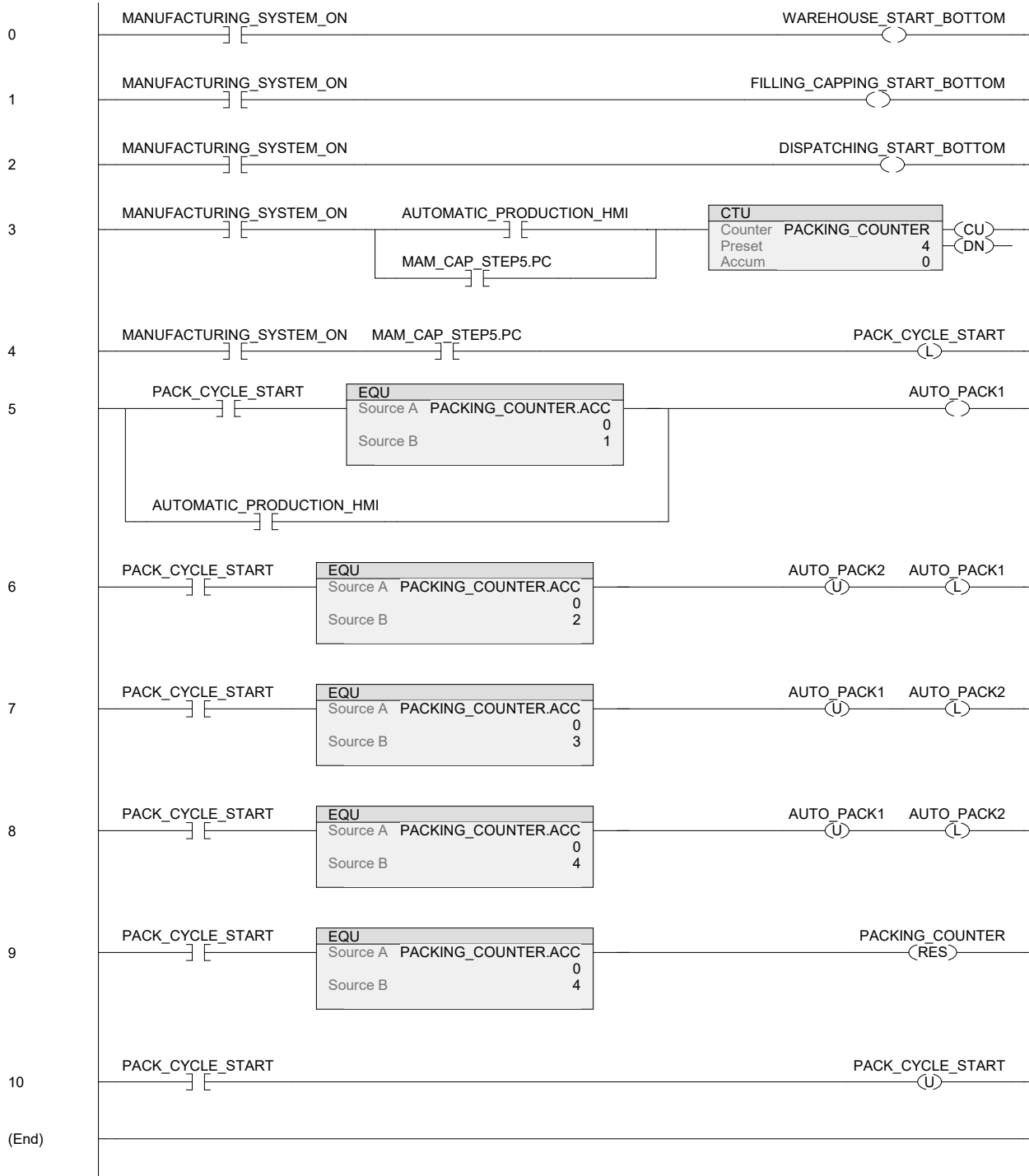
SMV40:MainTask:MainProgram

Total number of rungs in routine: 11

Page 1

15/07/2020 3:35:10 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



G

Anexo: Ladder - Rutina SCARA Home

G. Anexo: Ladder - Rutina SCARA Home

SCARA_HOME - Ladder Diagram

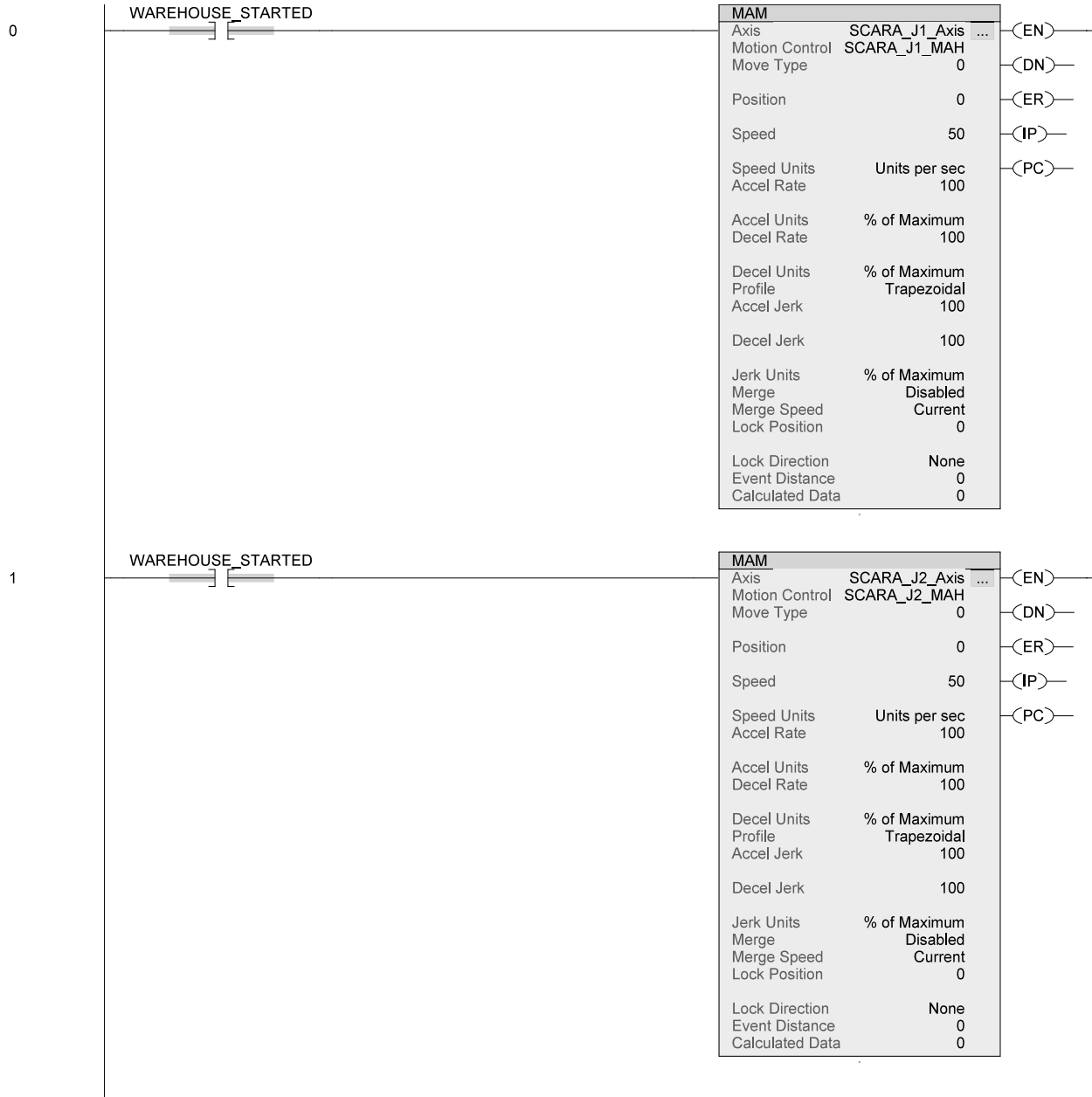
SMV40:MainTask:MainProgram

Total number of rungs in routine: 6

Page 1

15/07/2020 3:30:11 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



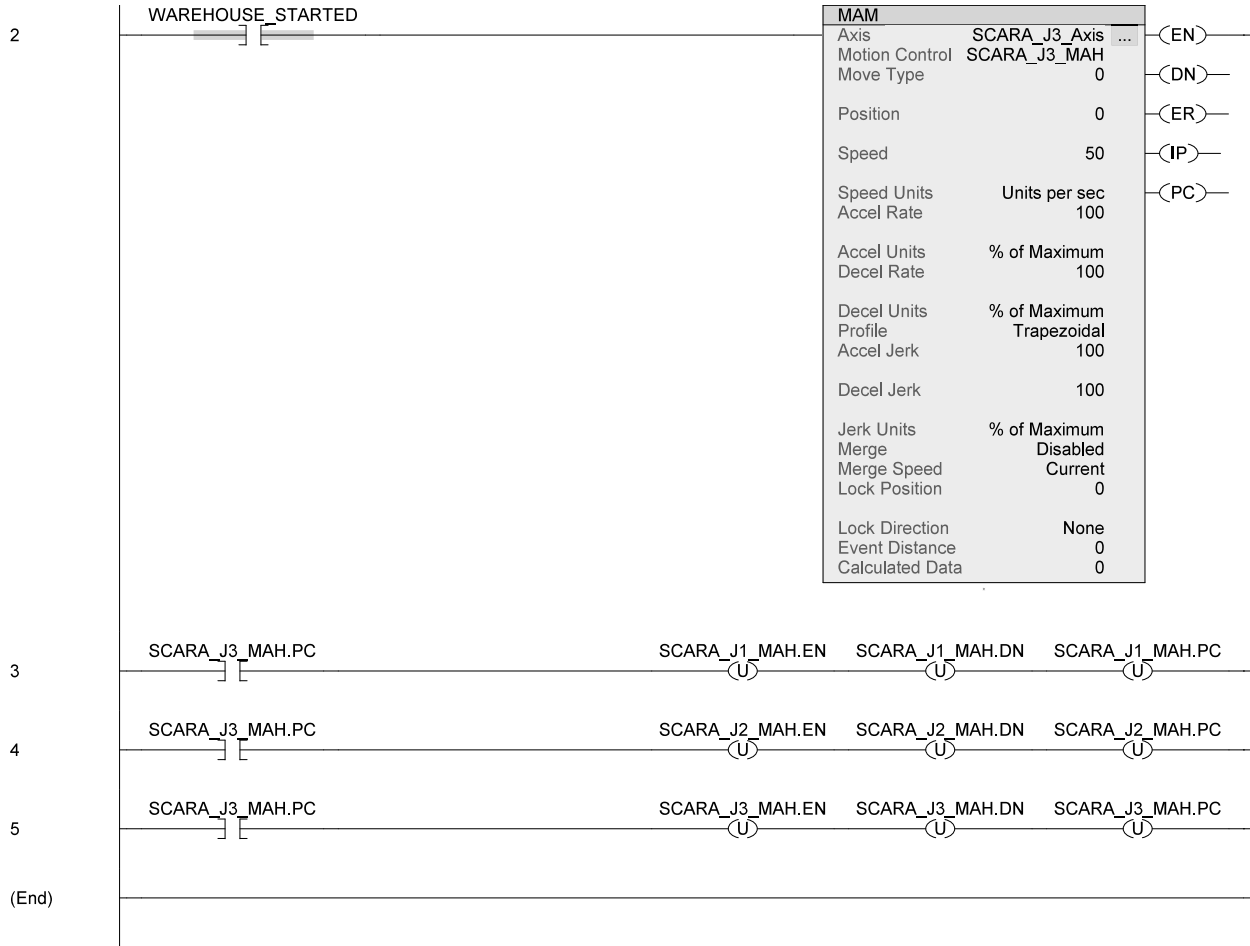
SCARA_HOME - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 6

15/07/2020 3:30:12 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



H

Anexo: Ladder - Rutina SCARA Automática

H. Anexo: Ladder - Rutina SCARA Automática

SCARA_PACK2_AUTO - Ladder Diagram

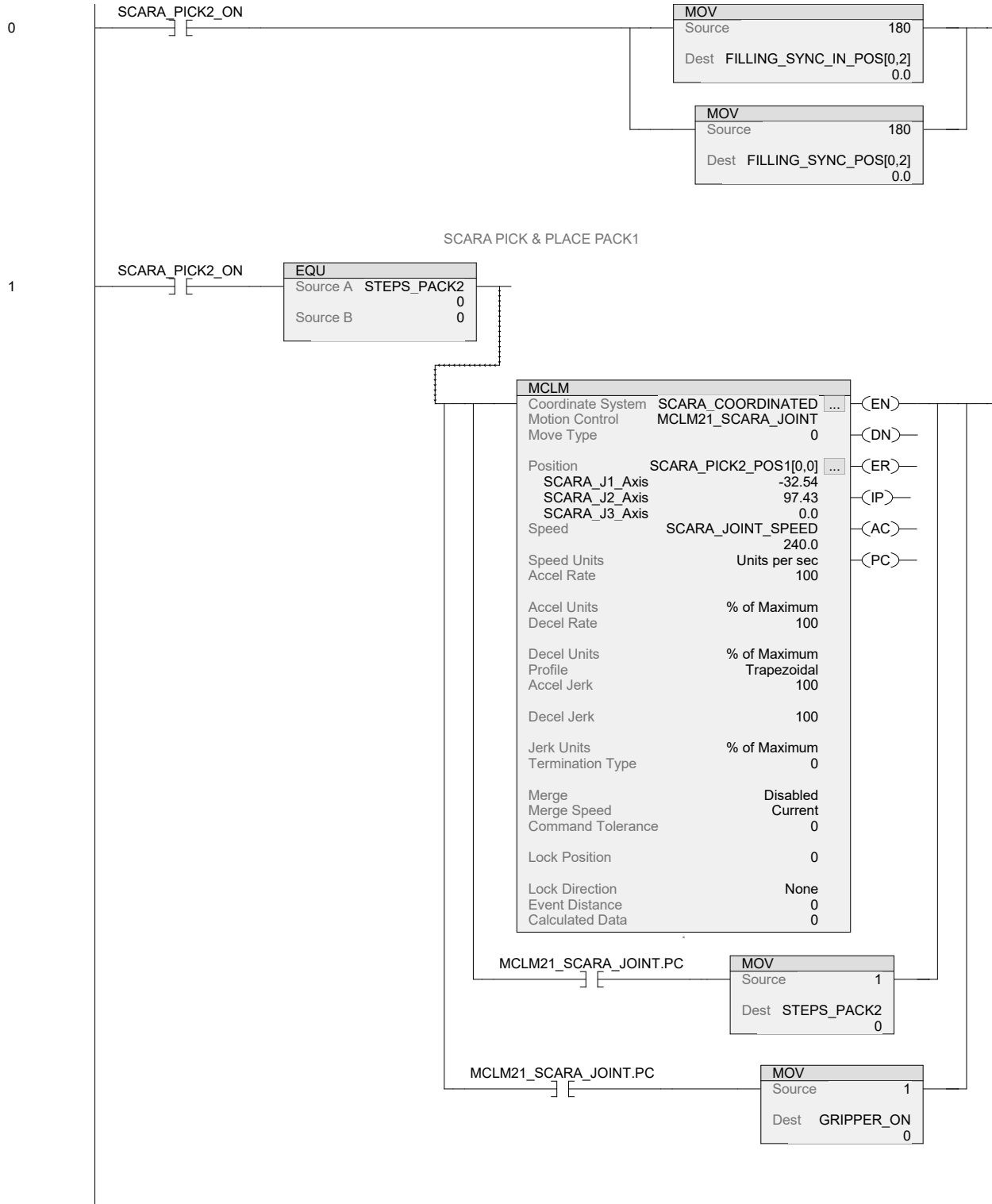
SMV40:MainTask:MainProgram

Total number of rungs in routine: 9

Page 1

15/07/2020 3:25:44 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



H. Anexo: Ladder - Rutina SCARA Automática

SCARA_PACK2_AUTO - Ladder Diagram

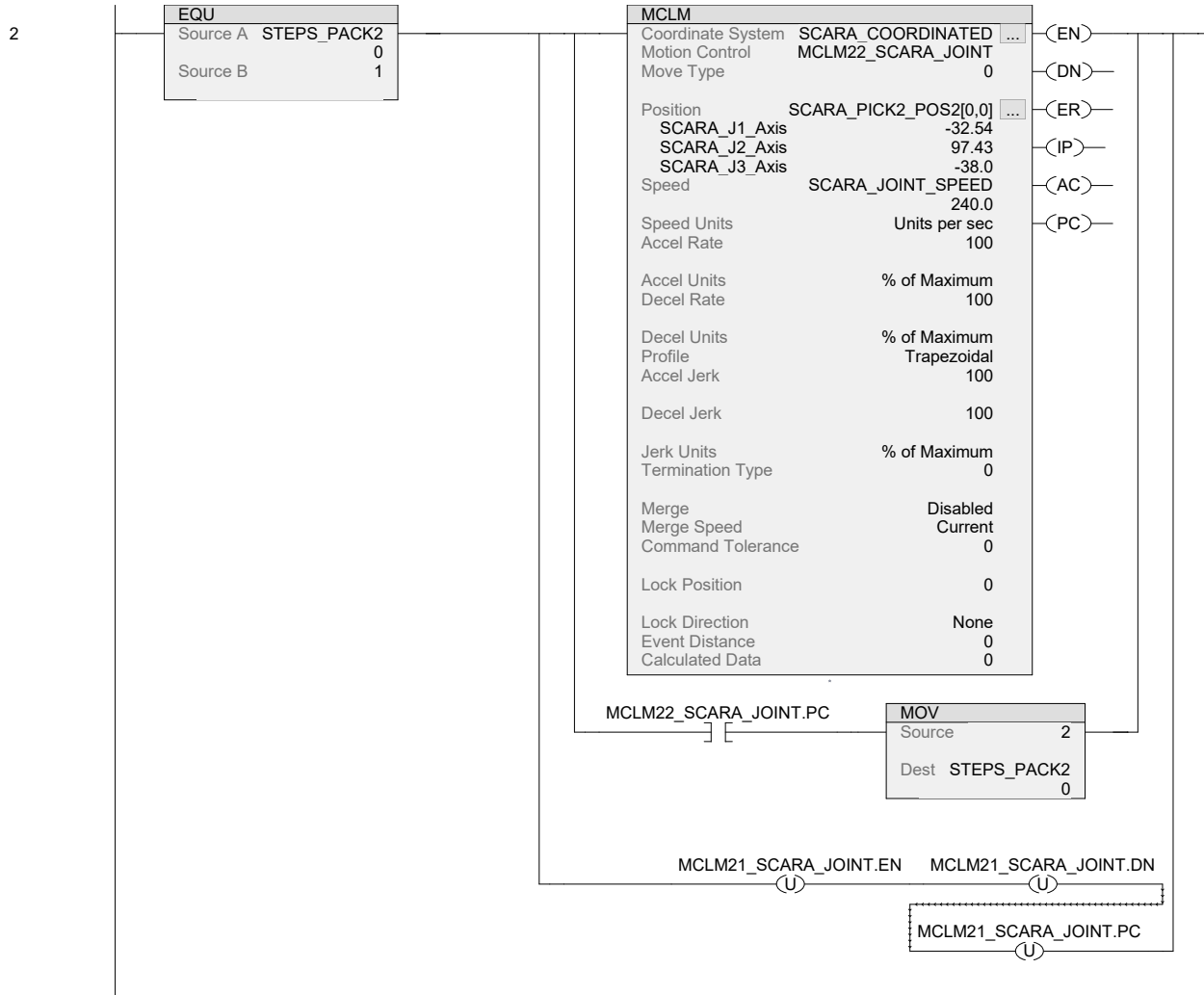
SMV40:MainTask:MainProgram

Total number of rungs in routine: 9

Page 2

15/07/2020 3:25:45 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



H. Anexo: Ladder - Rutina SCARA Automática

SCARA_PACK2_AUTO - Ladder Diagram

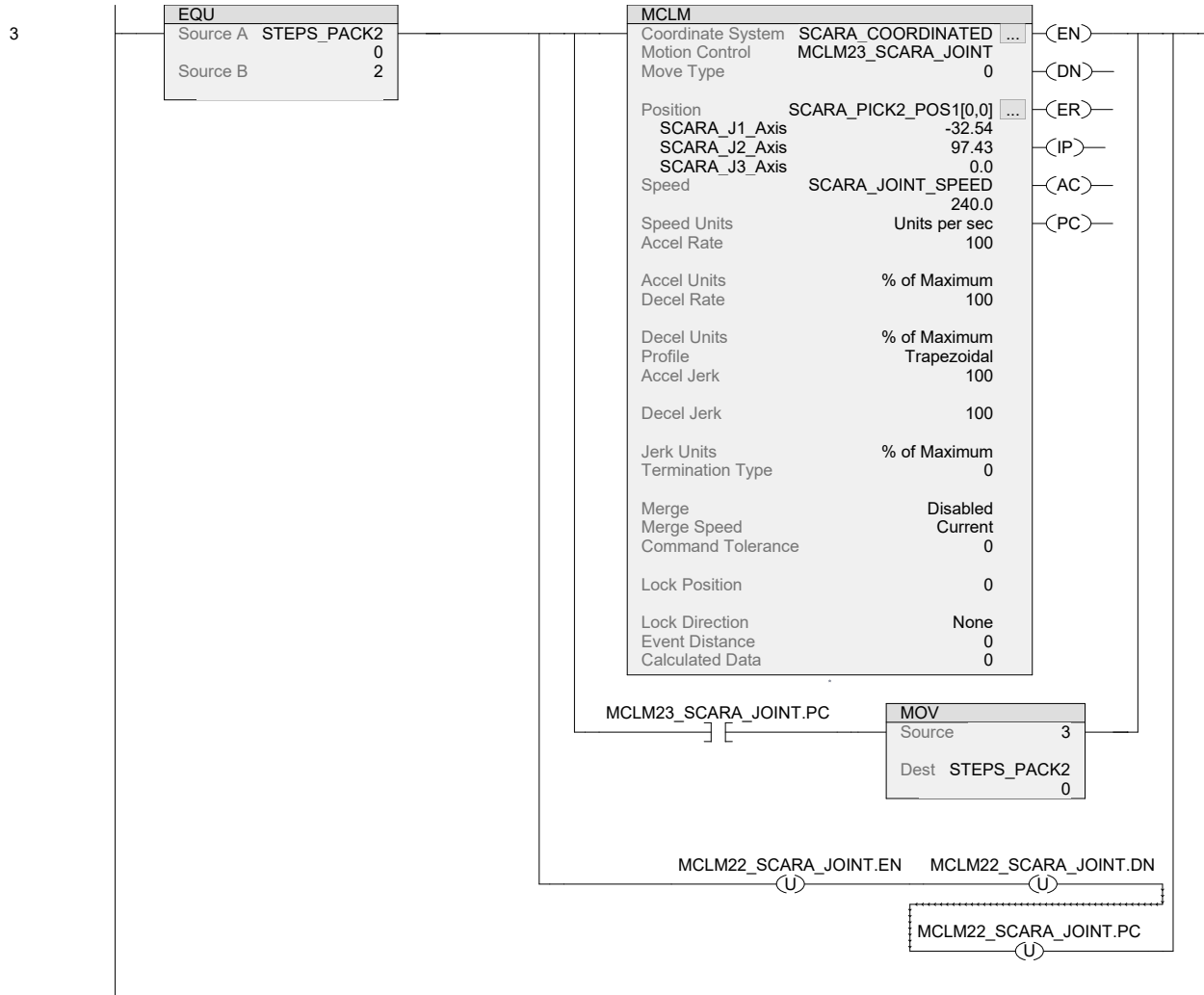
SMV40:MainTask:MainProgram

Total number of rungs in routine: 9

Page 3

15/07/2020 3:25:45 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



H. Anexo: Ladder - Rutina SCARA Automática

SCARA_PACK2_AUTO - Ladder Diagram

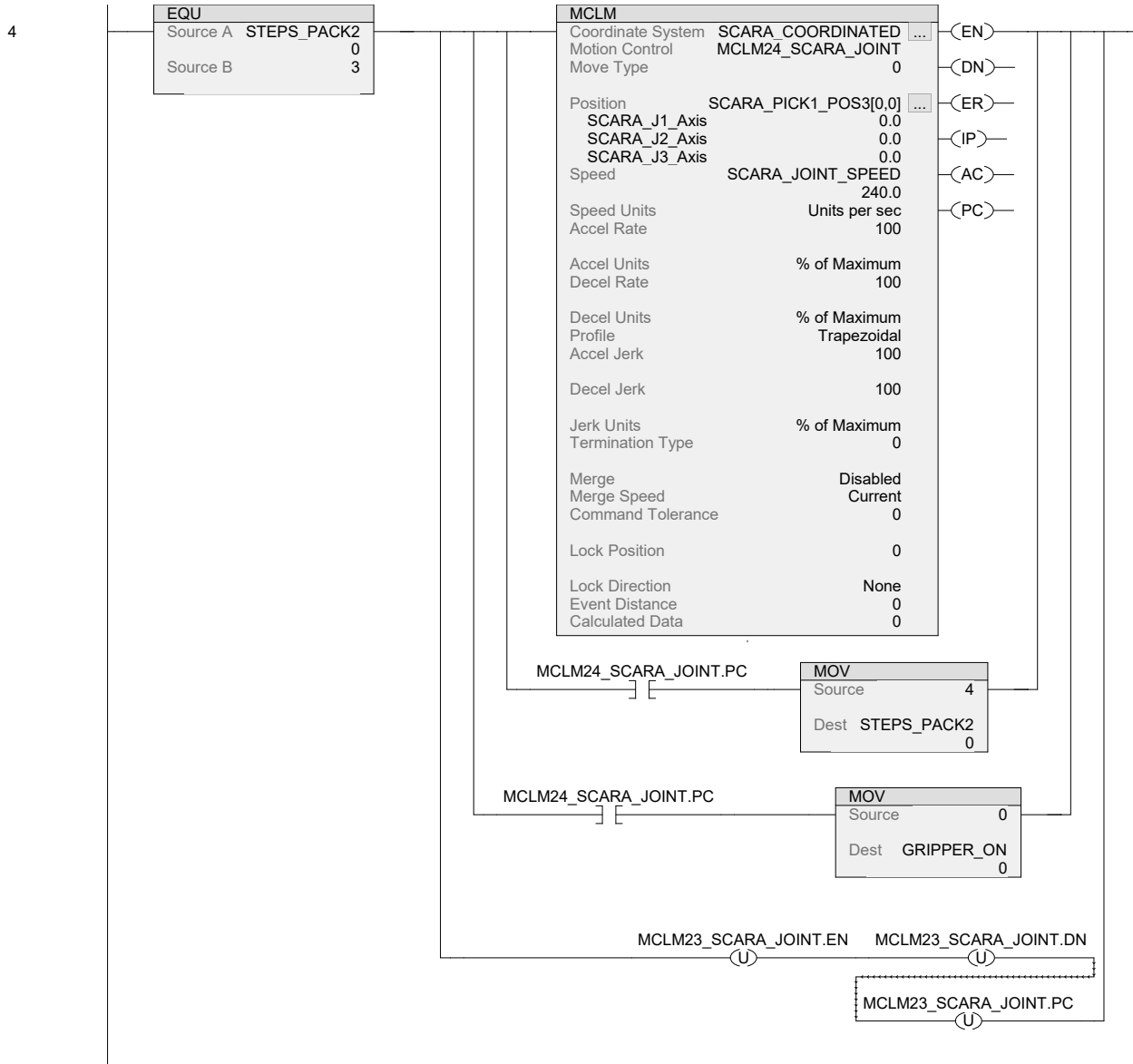
SMV40:MainTask:MainProgram

Total number of rungs in routine: 9

Page 4

15/07/2020 3:25:45 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



H. Anexo: Ladder - Rutina SCARA Automática

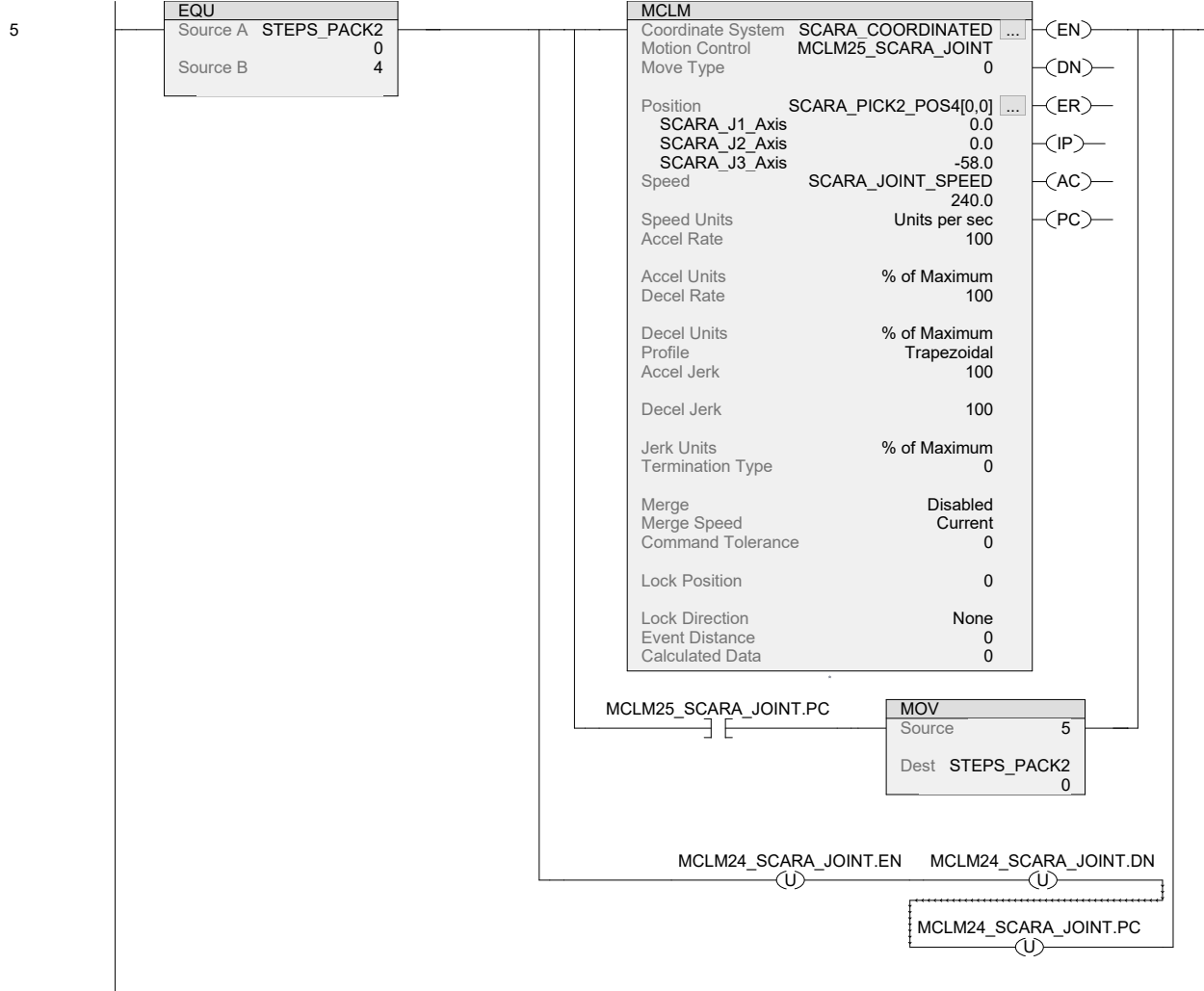
SCARA_PACK2_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 9

15/07/2020 3:25:45 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



H. Anexo: Ladder - Rutina SCARA Automática

SCARA_PACK2_AUTO - Ladder Diagram

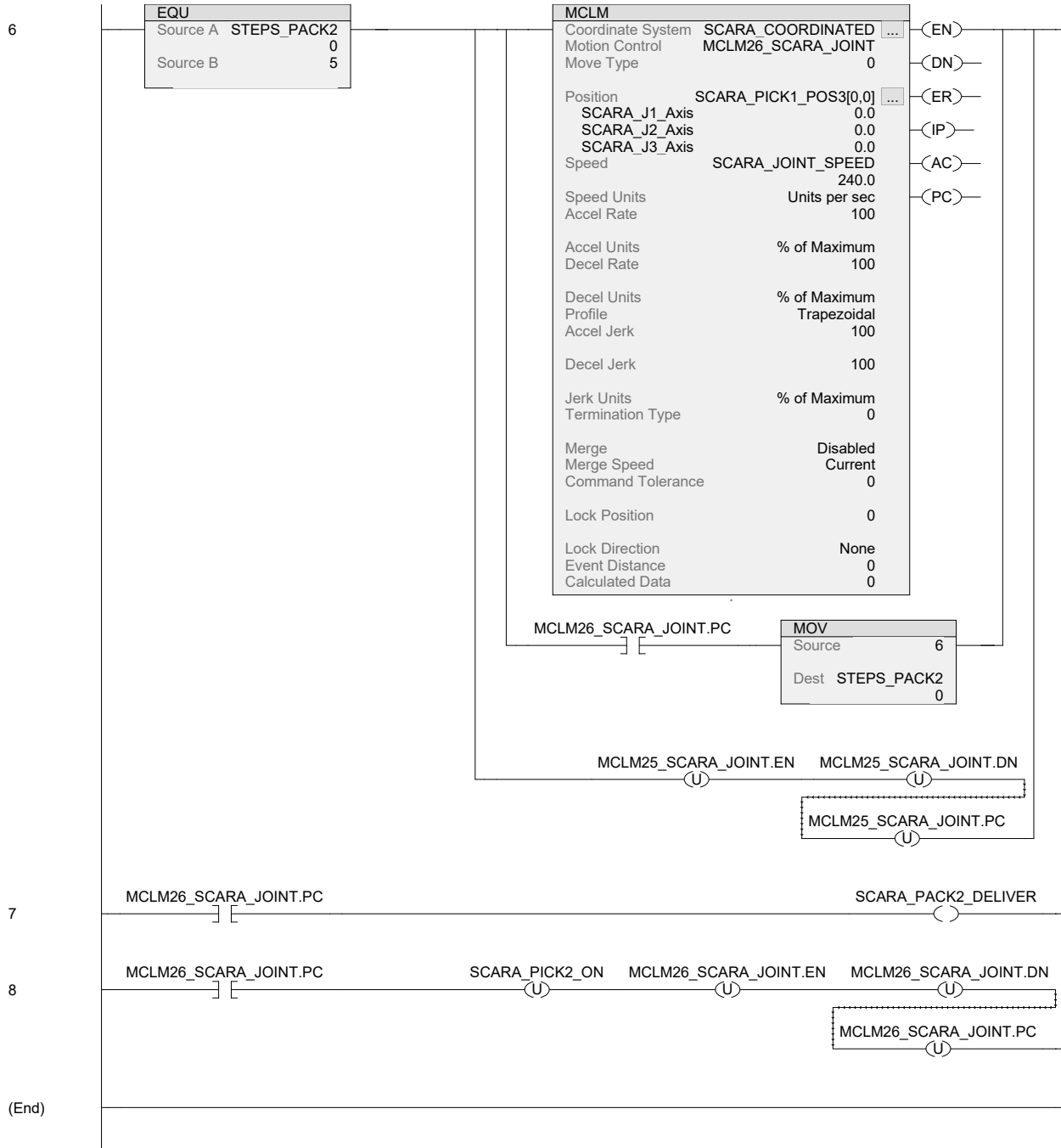
SMV40:MainTask:MainProgram

Total number of rungs in routine: 9

Page 6

15/07/2020 3:25:45 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



I

Anexo: Ladder - Rutina Robot Cartesiano Home

I. Anexo: Ladder - Rutina Robot Cartesiano Home

CARTESIAN_HOME - Ladder Diagram

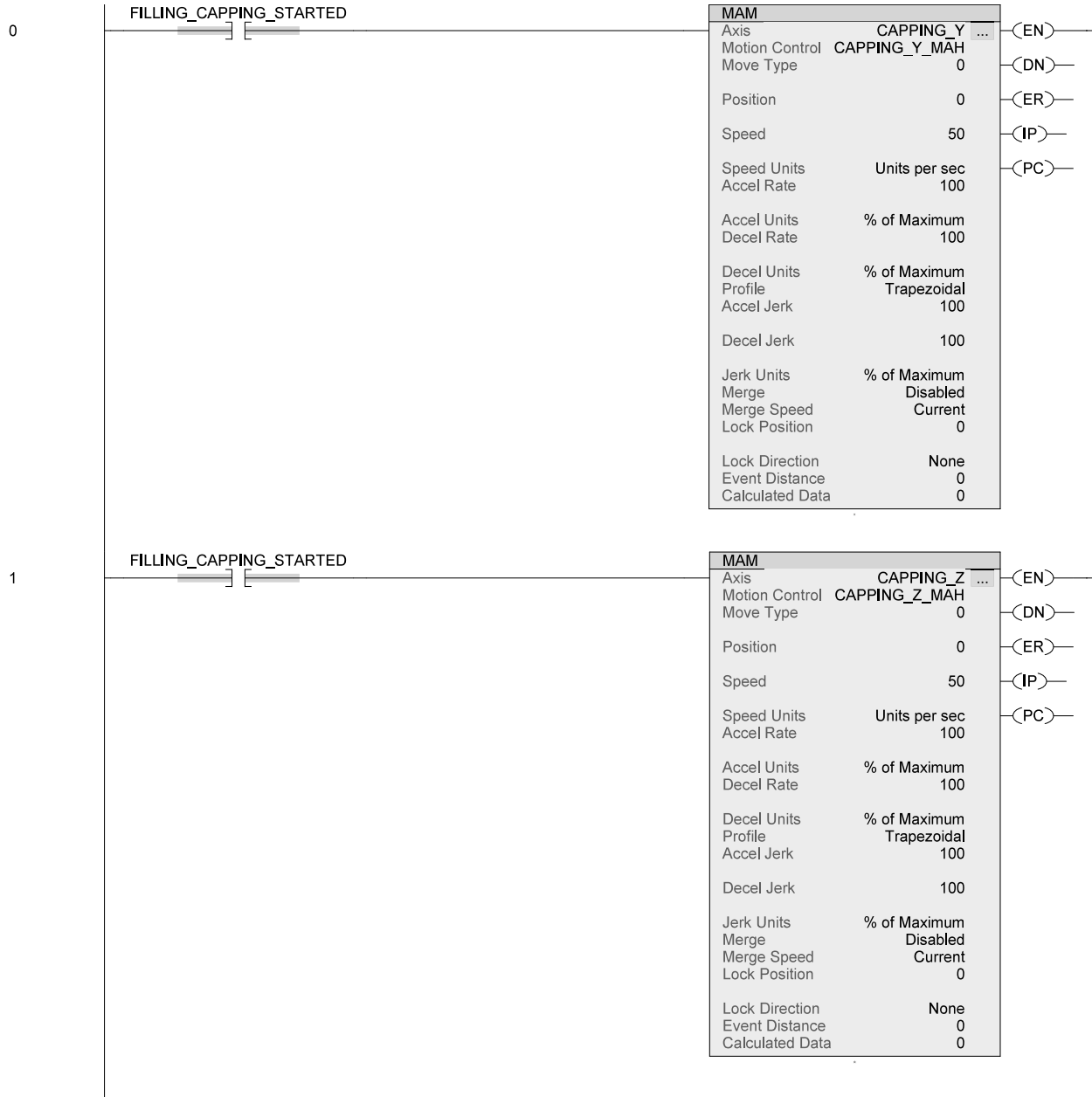
SMV40:MainTask:MainProgram

Total number of rungs in routine: 8

Page 1

15/07/2020 3:38:20 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



I. Anexo: Ladder - Rutina Robot Cartesiano Home

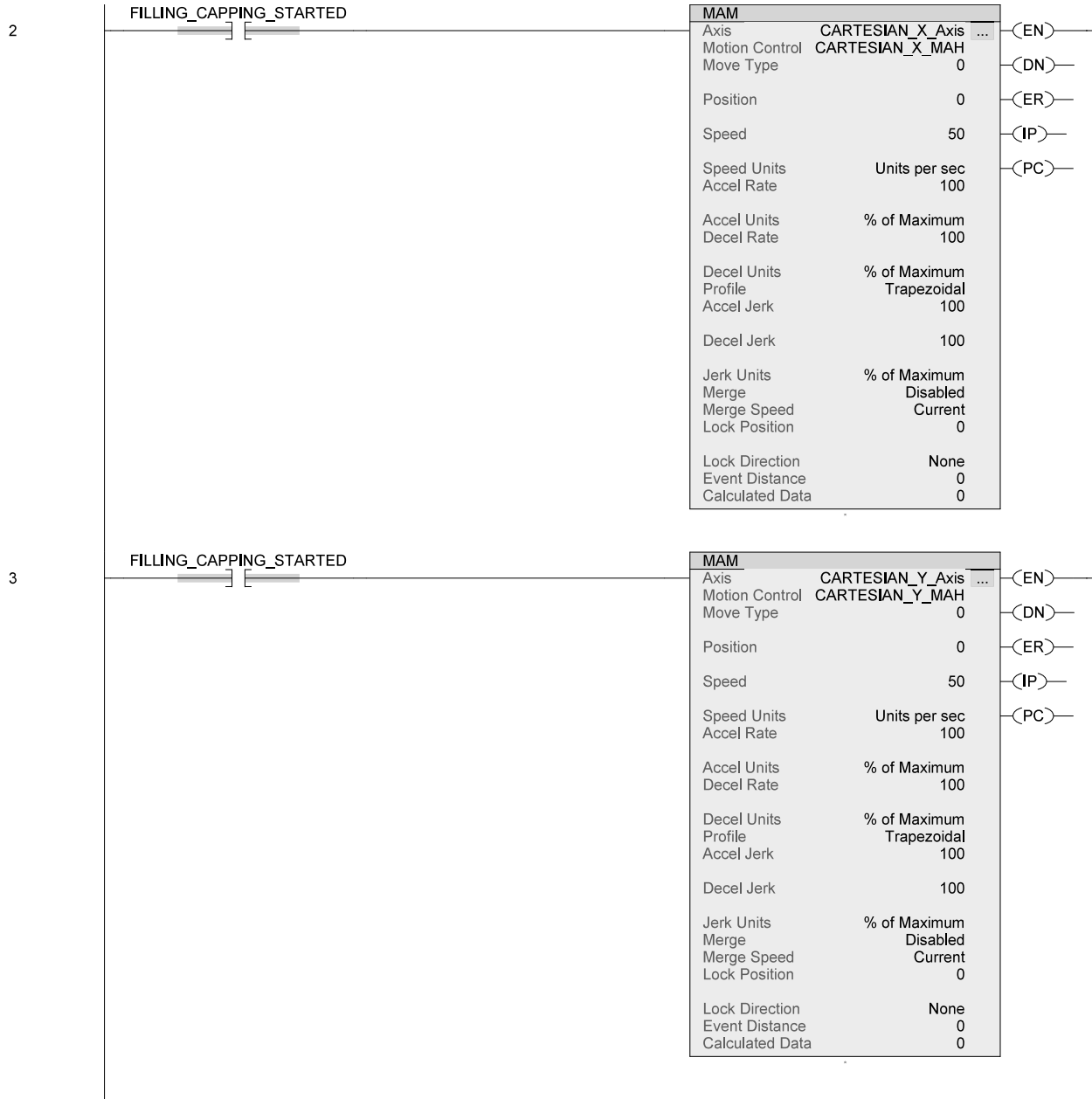
CARTESIAN_HOME - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 8

15/07/2020 3:38:20 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



I. Anexo: Ladder - Rutina Robot Cartesiano Home

CARTESIAN_HOME - Ladder Diagram

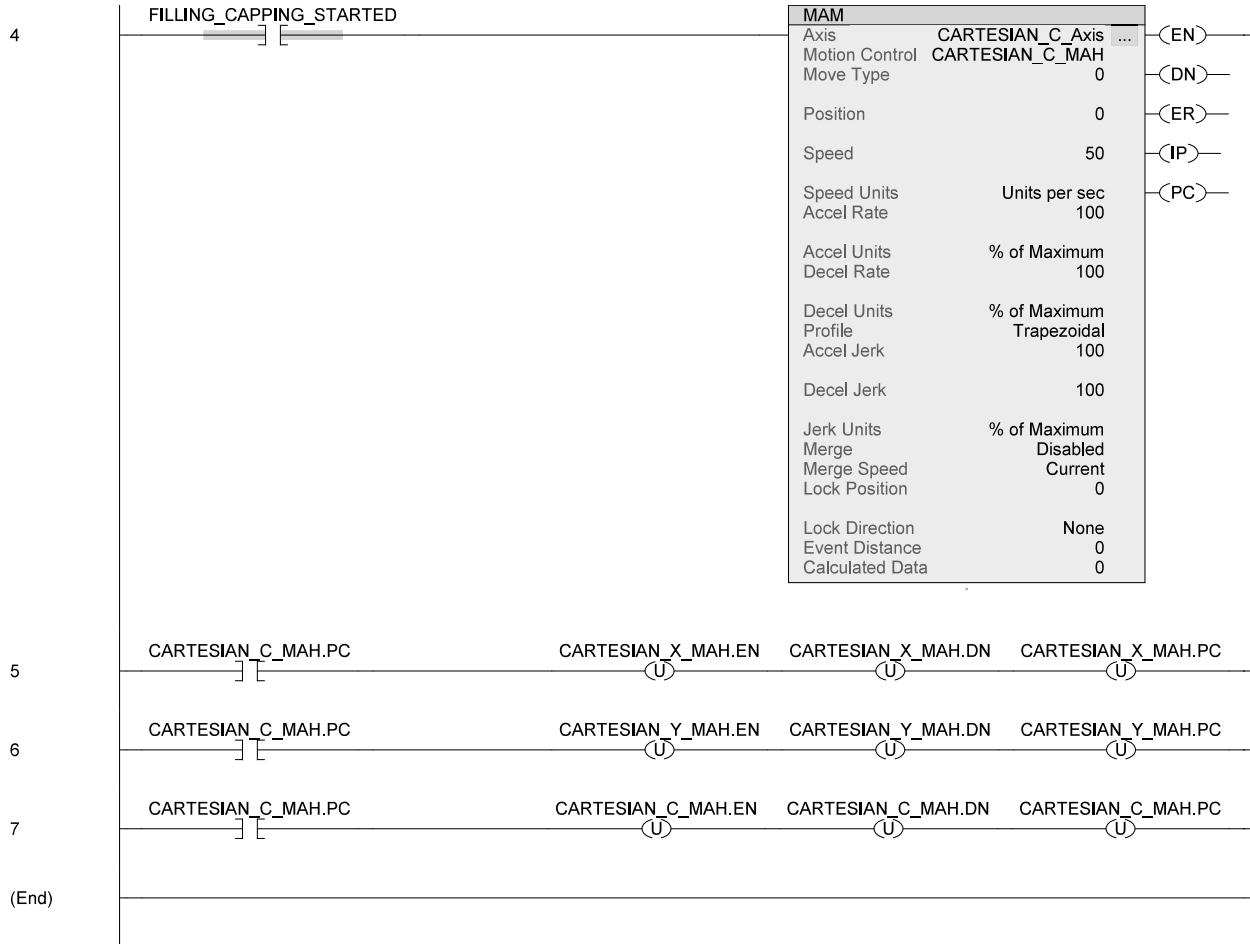
SMV40:MainTask:MainProgram

Total number of rungs in routine: 8

Page 3

15/07/2020 3:38:20 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



J

Anexo: Ladder - Rutina Robot Cartesiano Automática

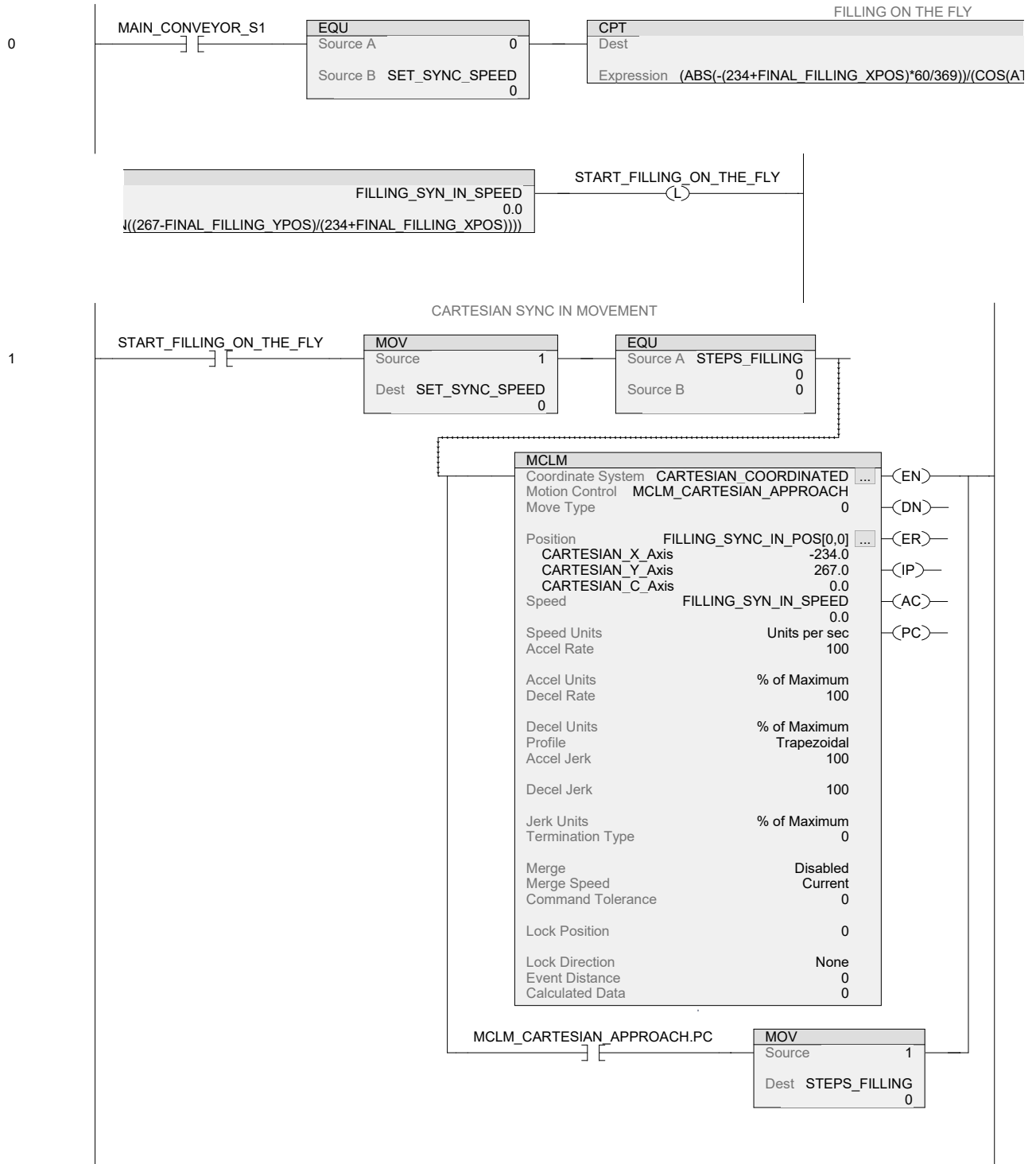
CARTESIAN_FILL_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

15/07/2020 3:41:36 PM

Total number of rungs in routine: 5

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



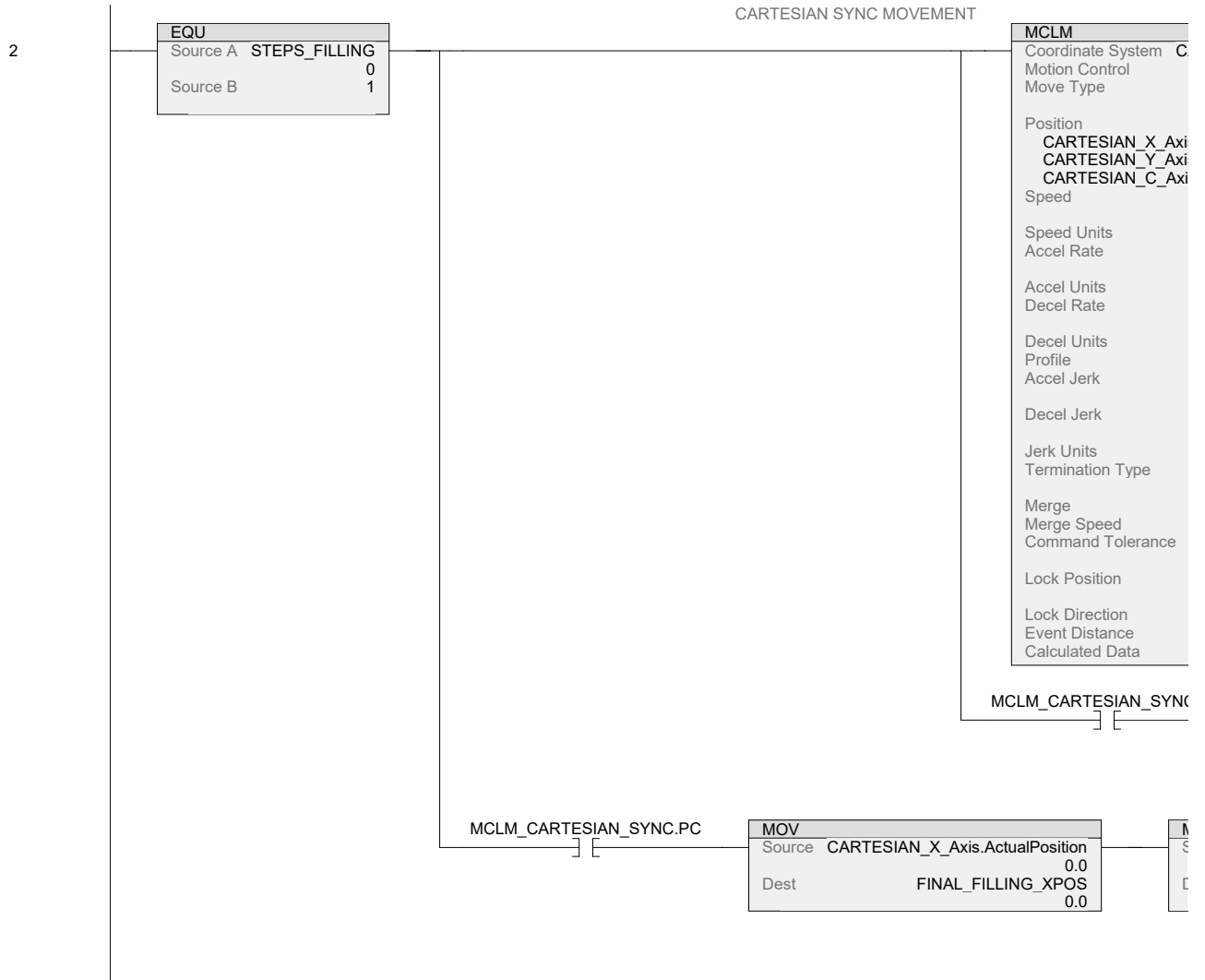
CARTESIAN_FILL_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 5

15/07/2020 3:41:36 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



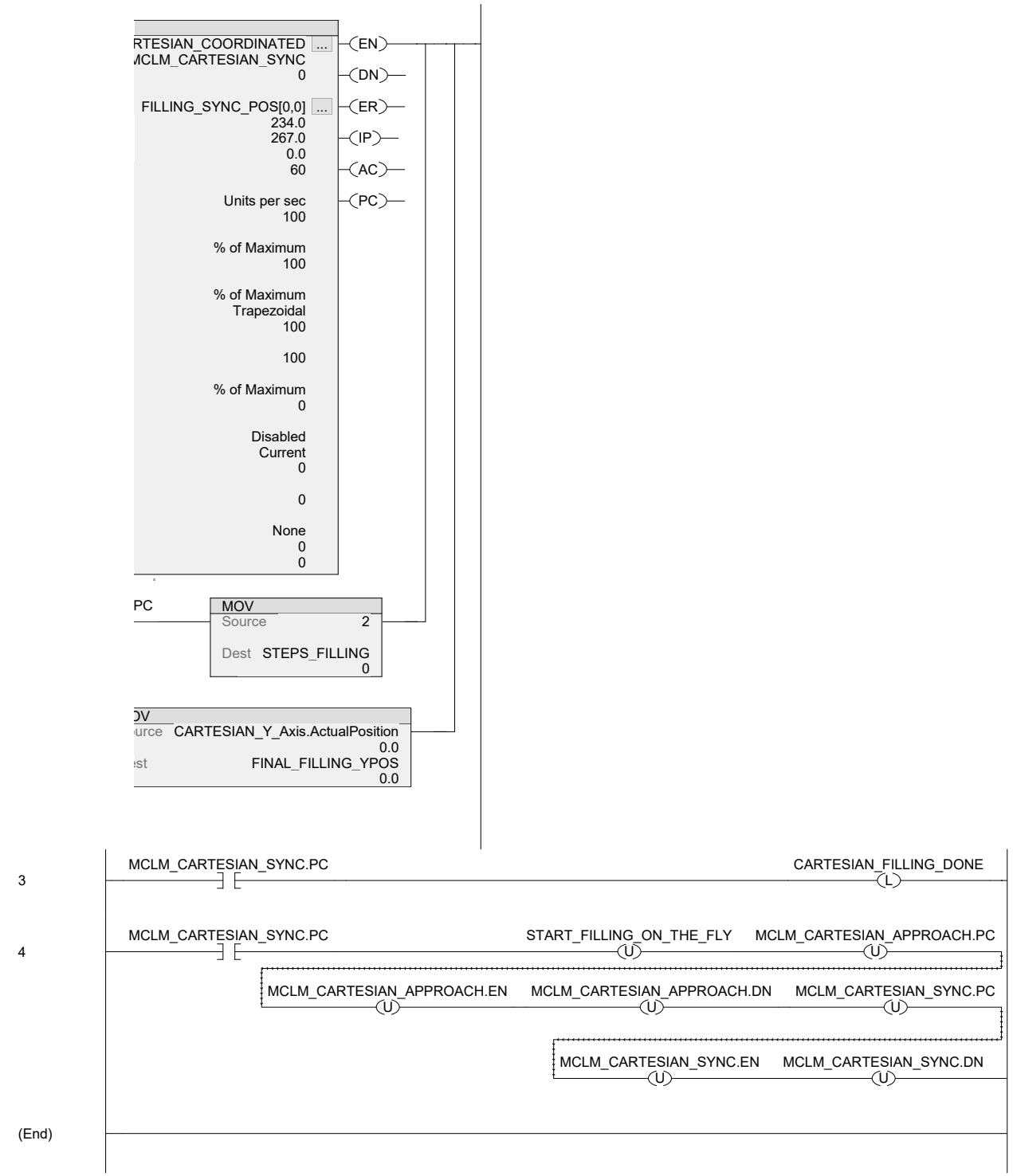
CARTESIAN_FILL_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 5

15/07/2020 3:41:36 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



K

Anexo: Ladder - Rutina Robot DELTA Home

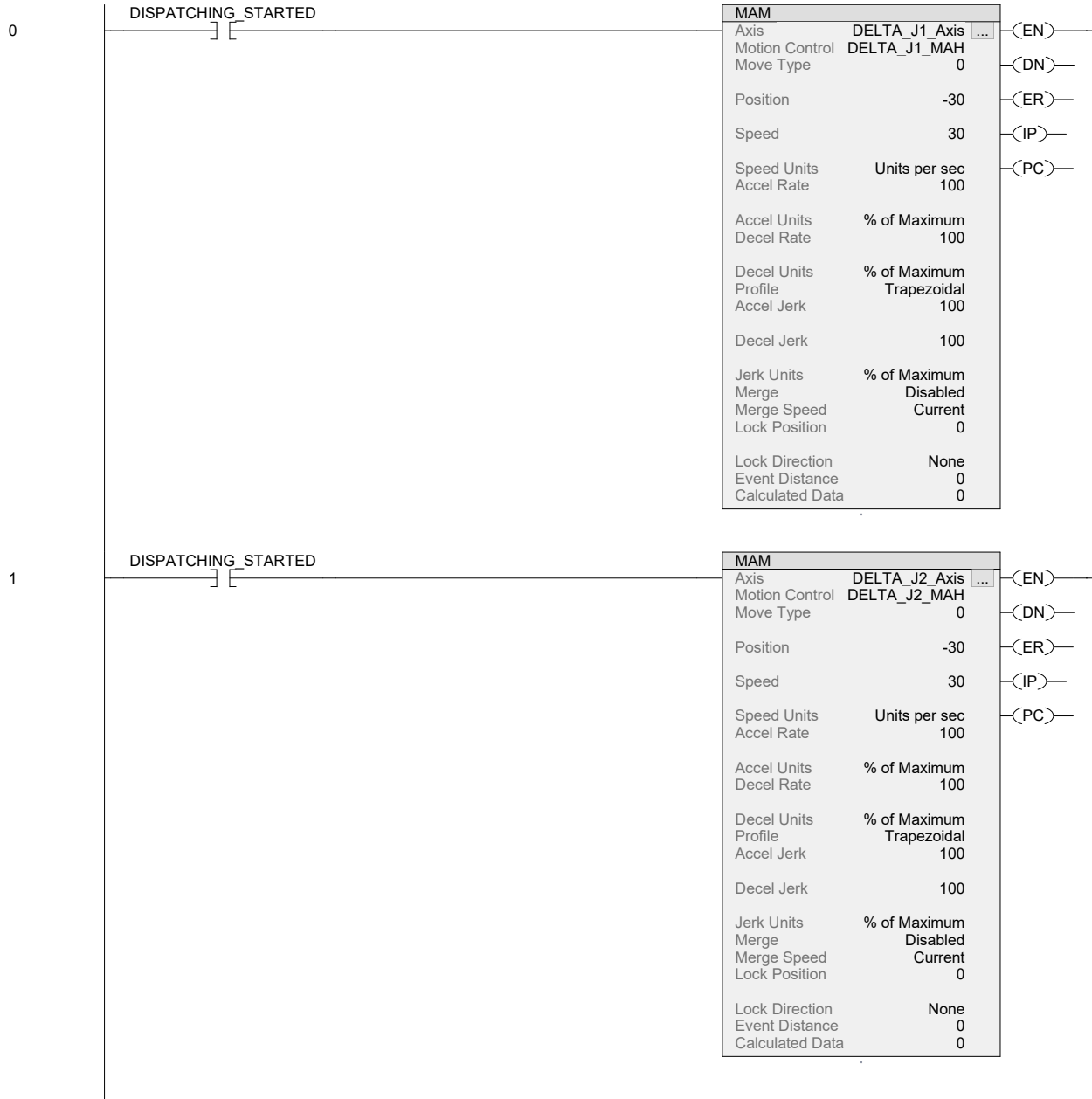
DELTA_HOME - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 6

15/07/2020 3:45:01 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



K. Anexo: Ladder - Rutina Robot DELTA Home

DELTA_HOME - Ladder Diagram

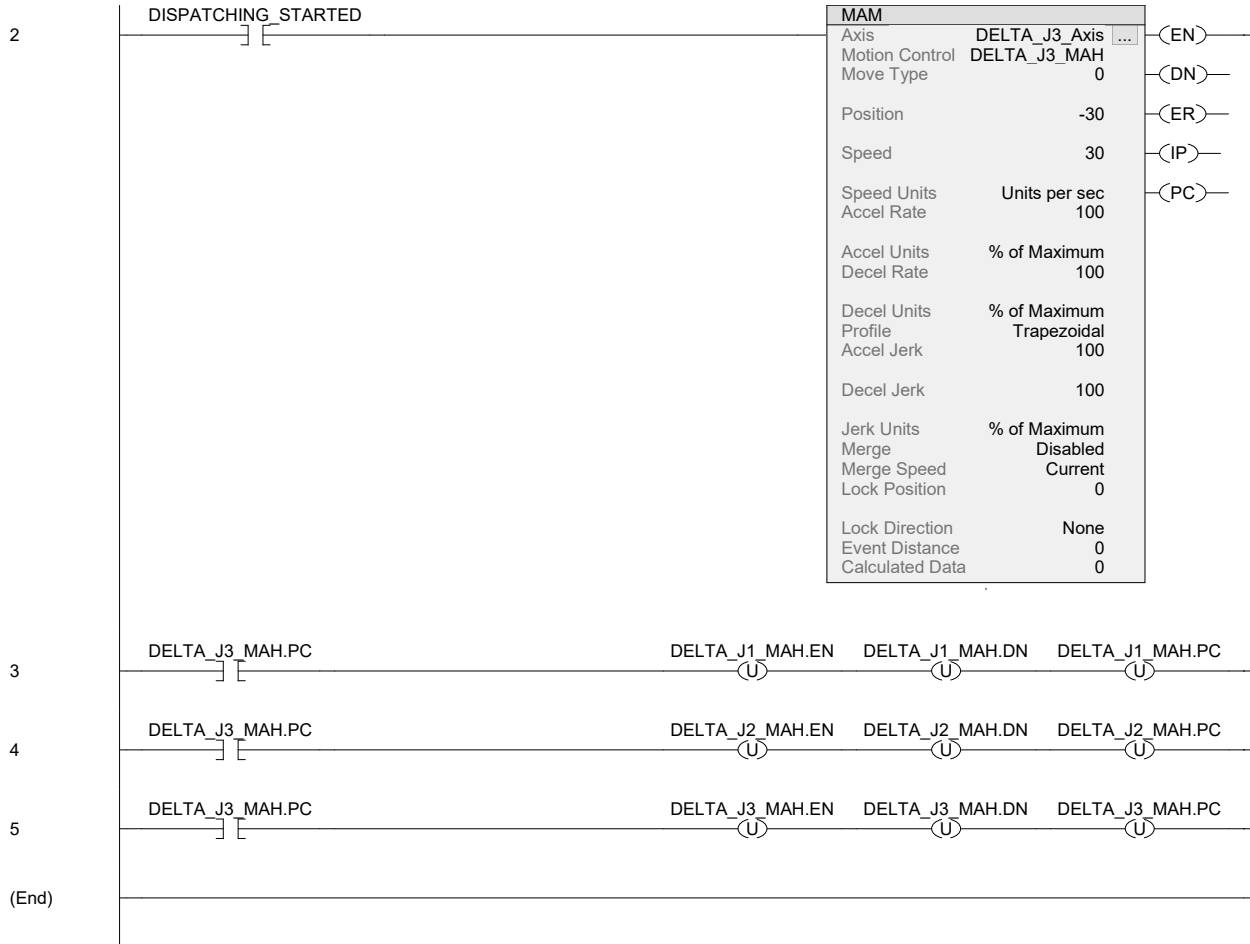
SMV40:MainTask:MainProgram

Total number of rungs in routine: 6

Page 2

15/07/2020 3:45:01 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



L

Anexo: Ladder - Rutina Robot DELTA Automática

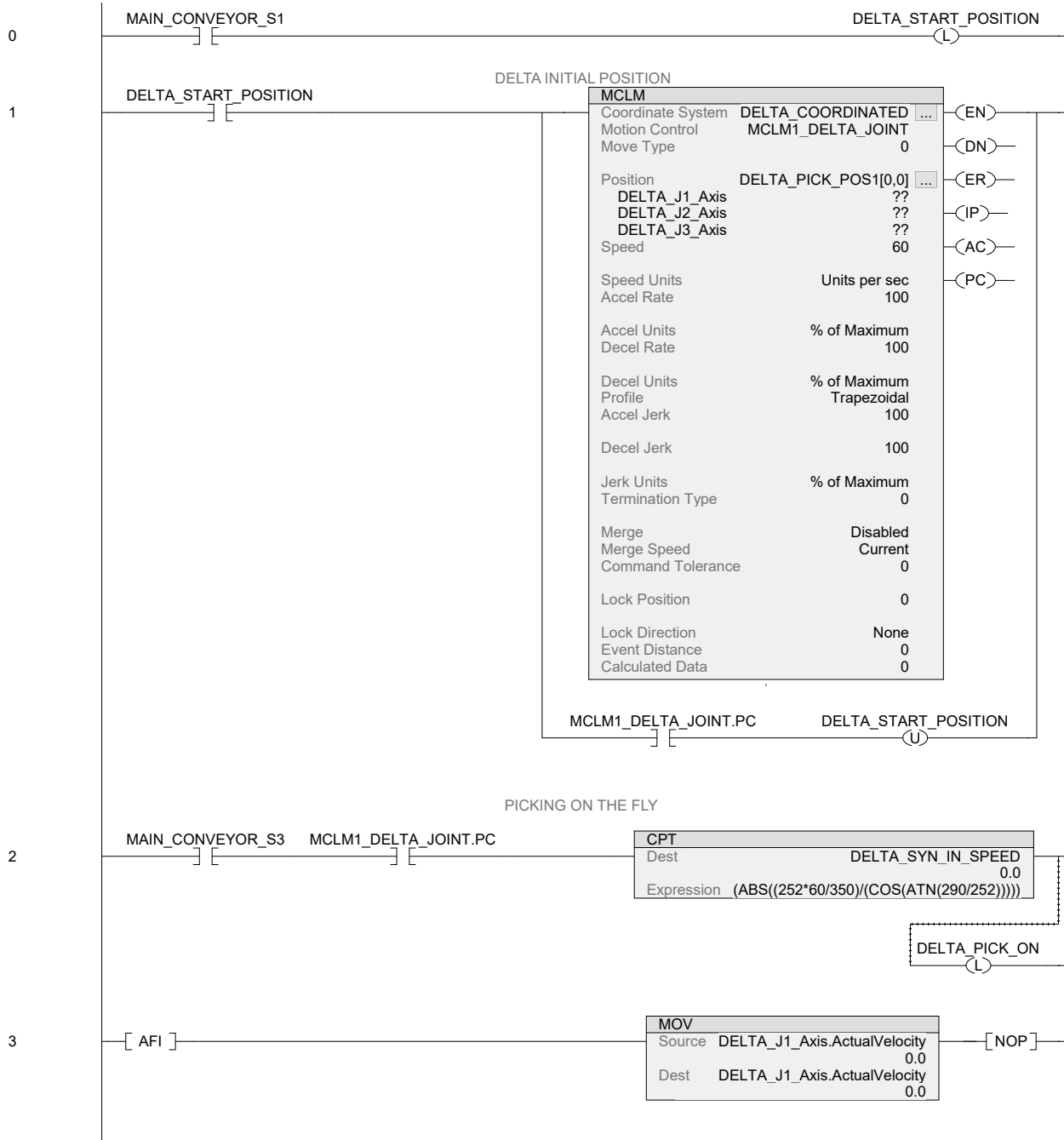
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



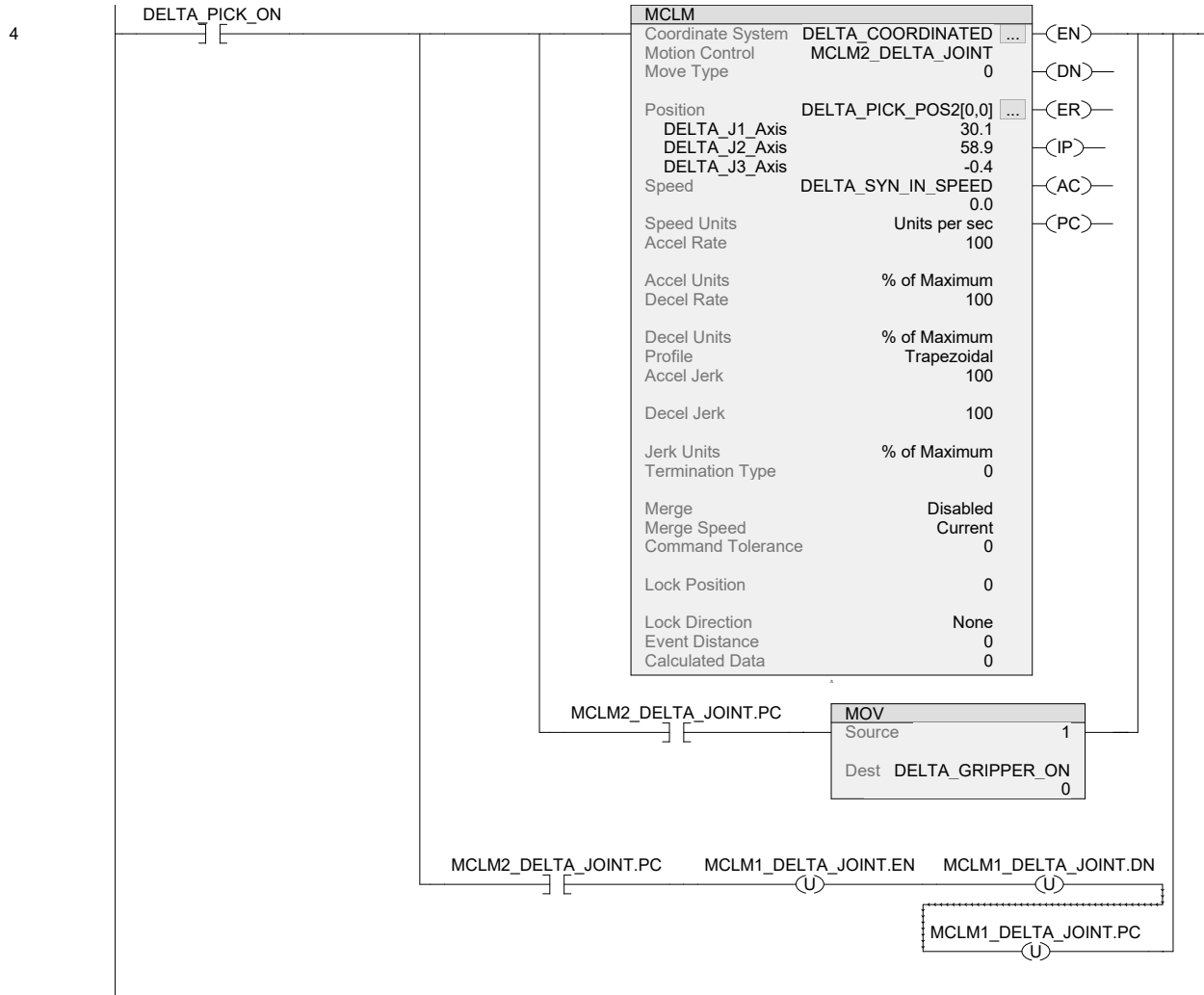
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



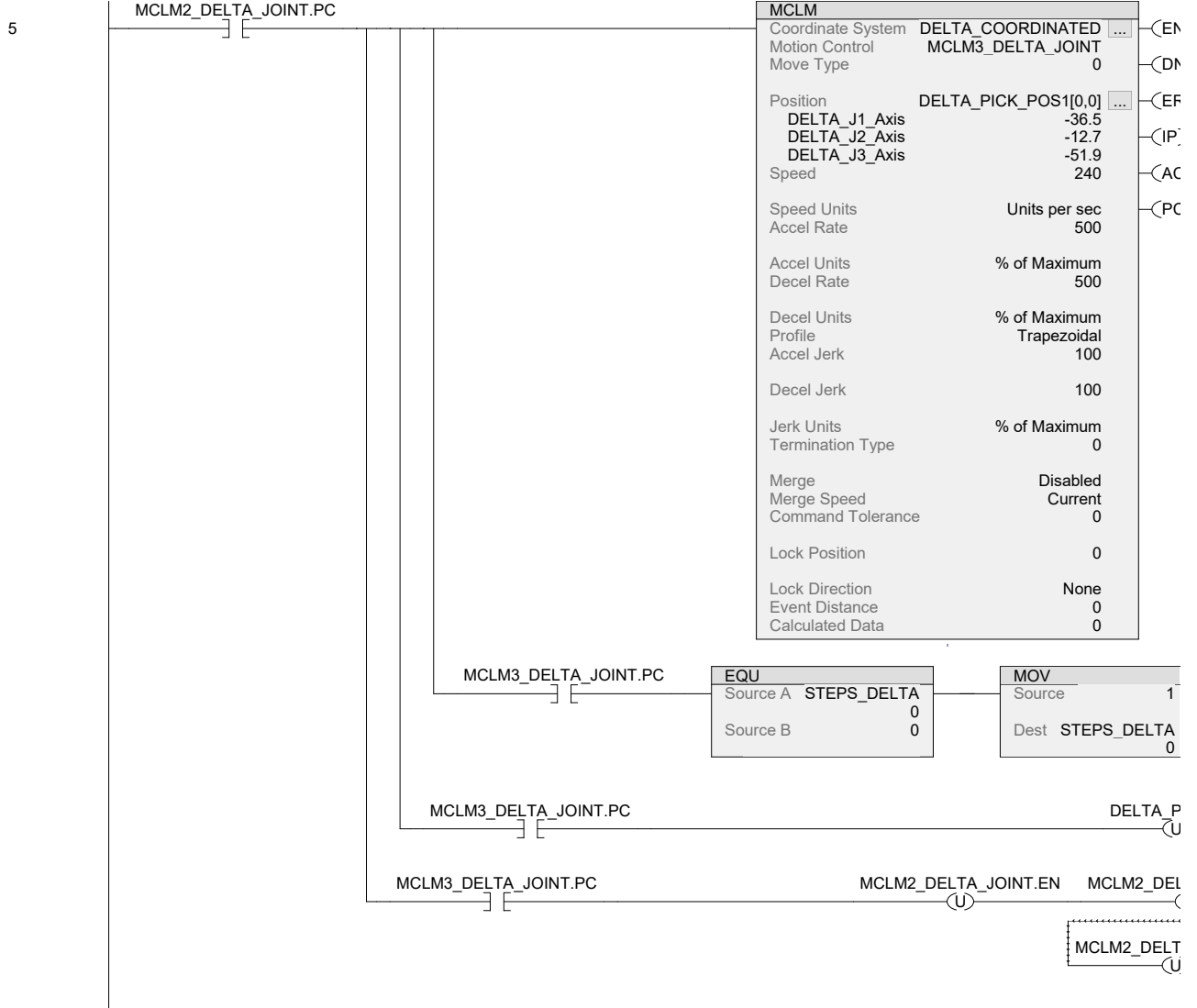
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



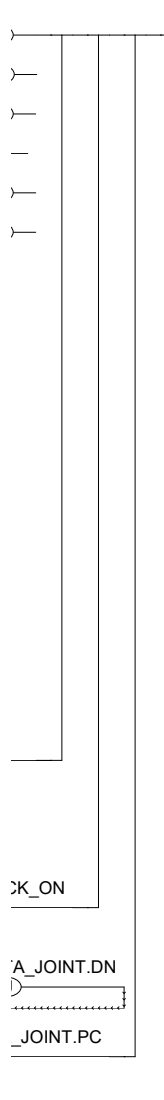
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



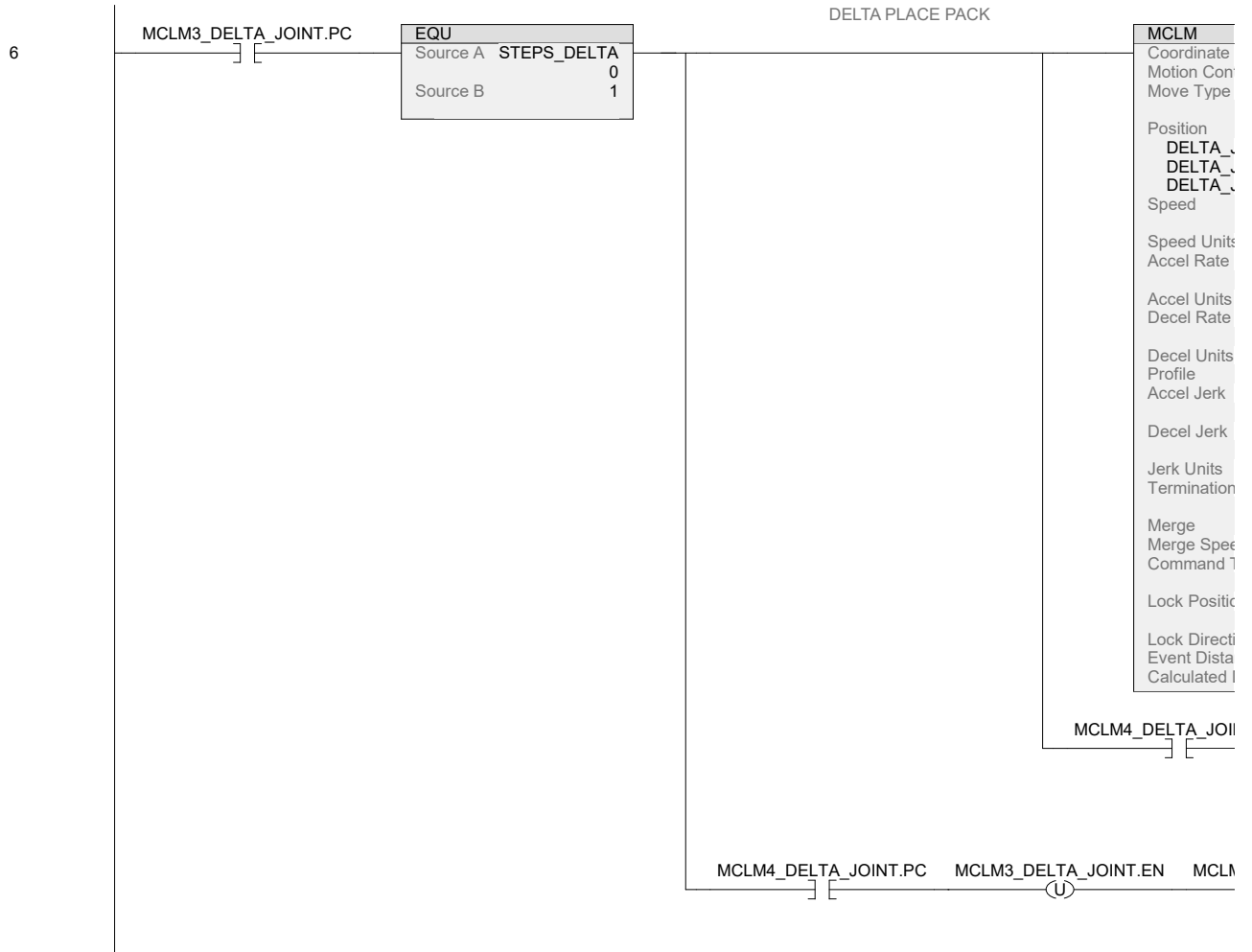
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

15/07/2020 3:47:23 PM

Total number of rungs in routine: 26

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



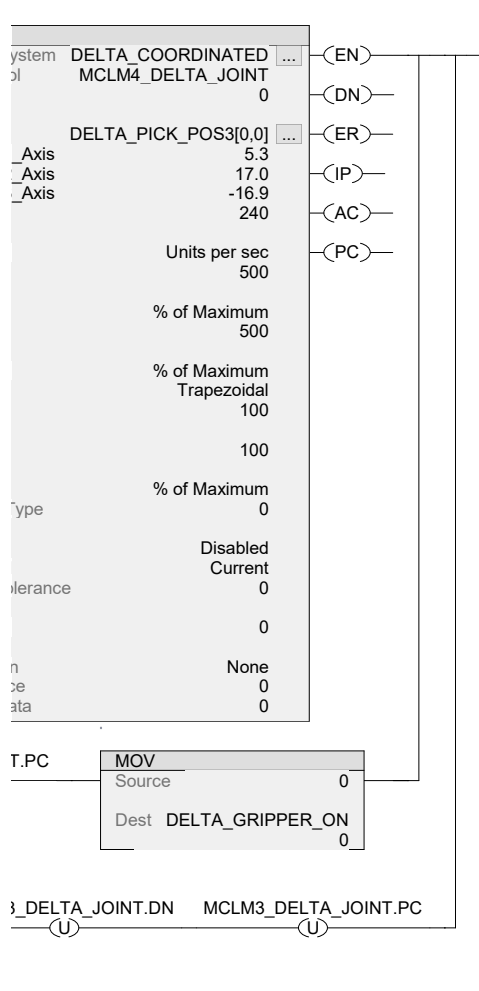
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



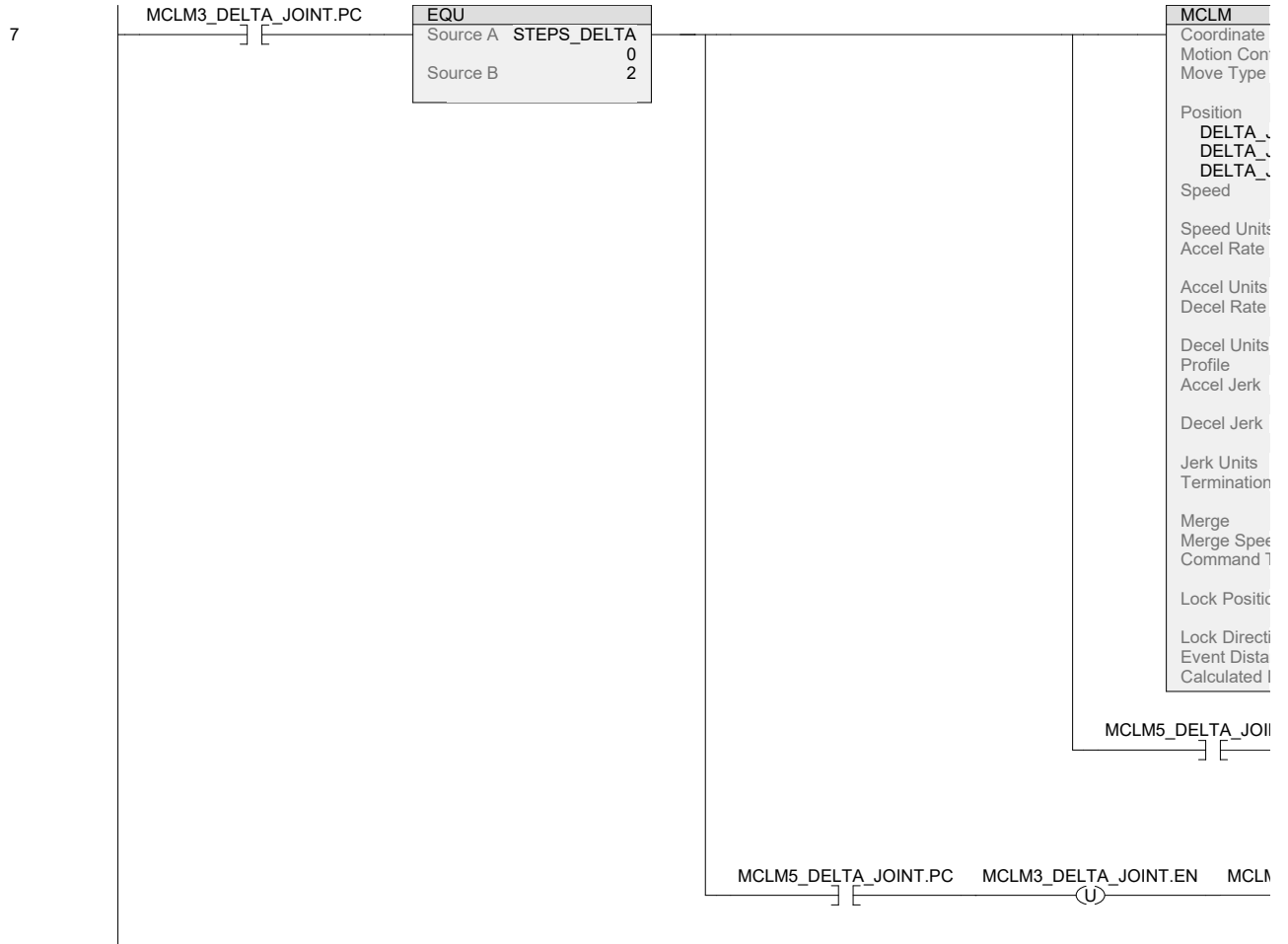
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



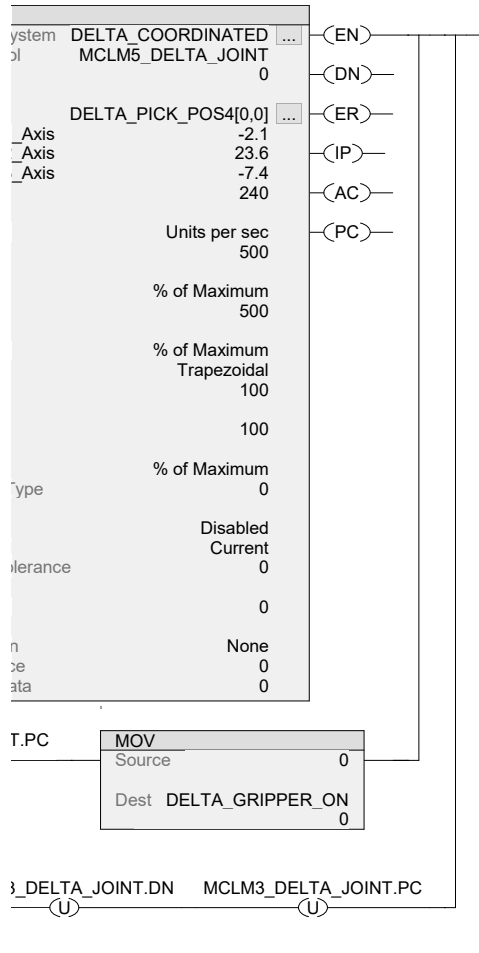
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



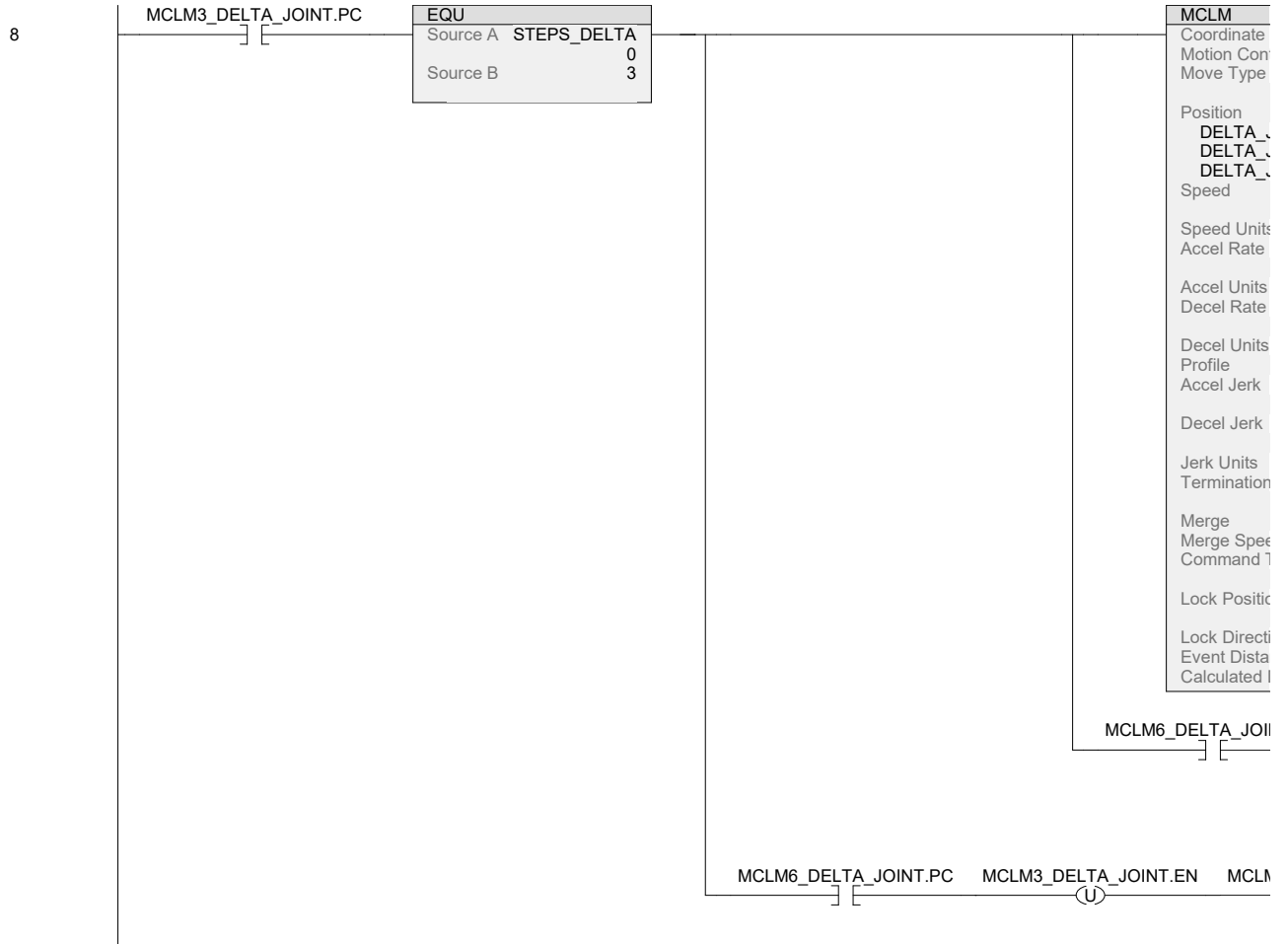
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



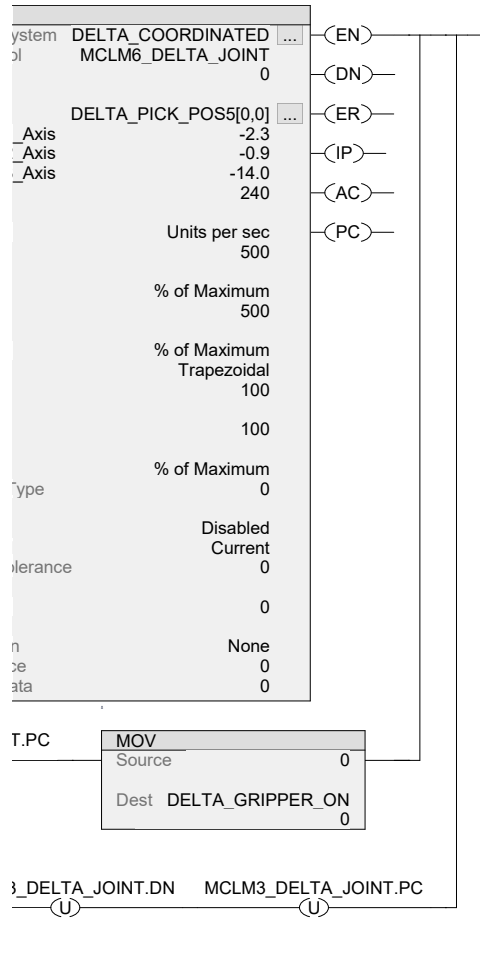
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



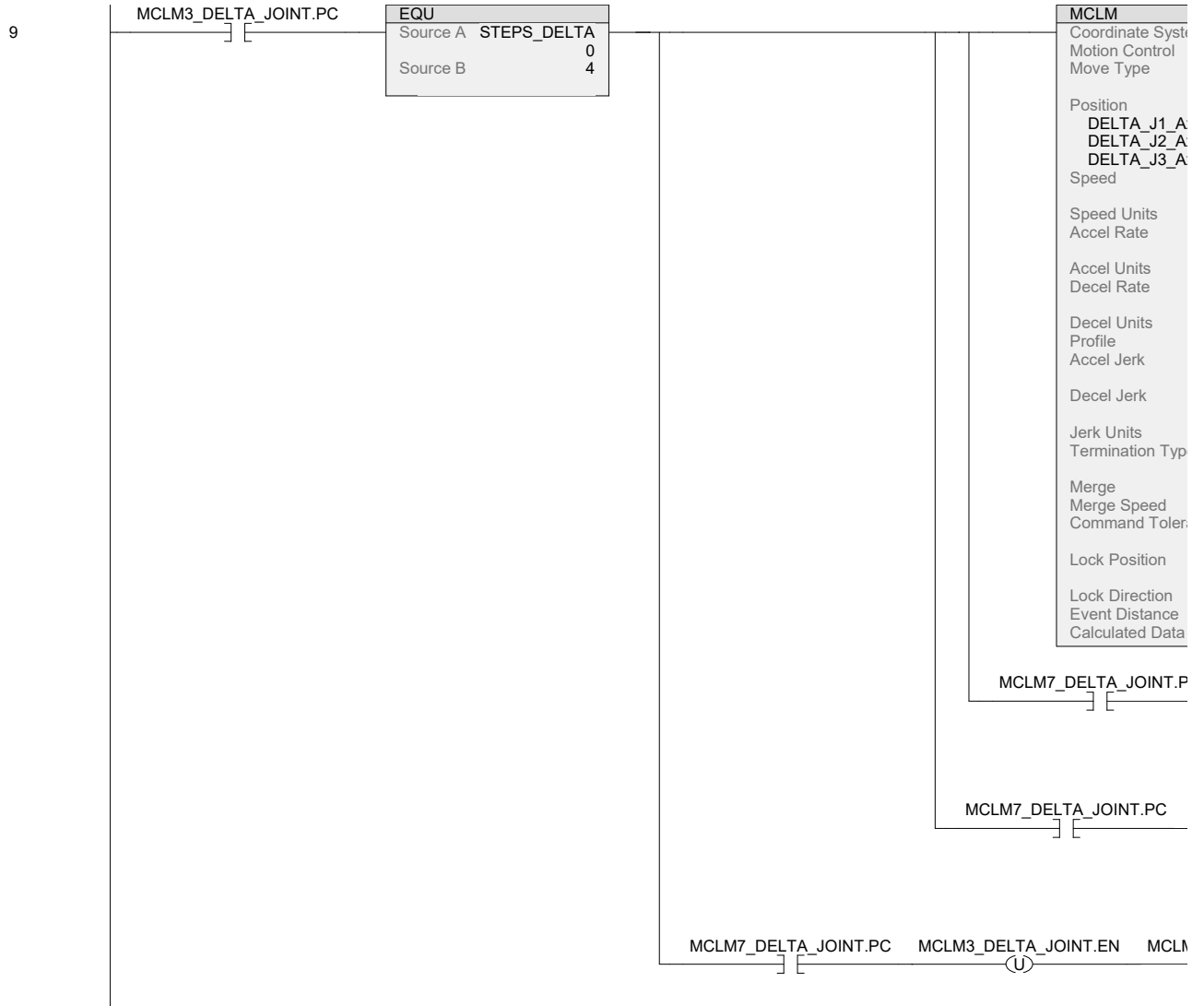
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



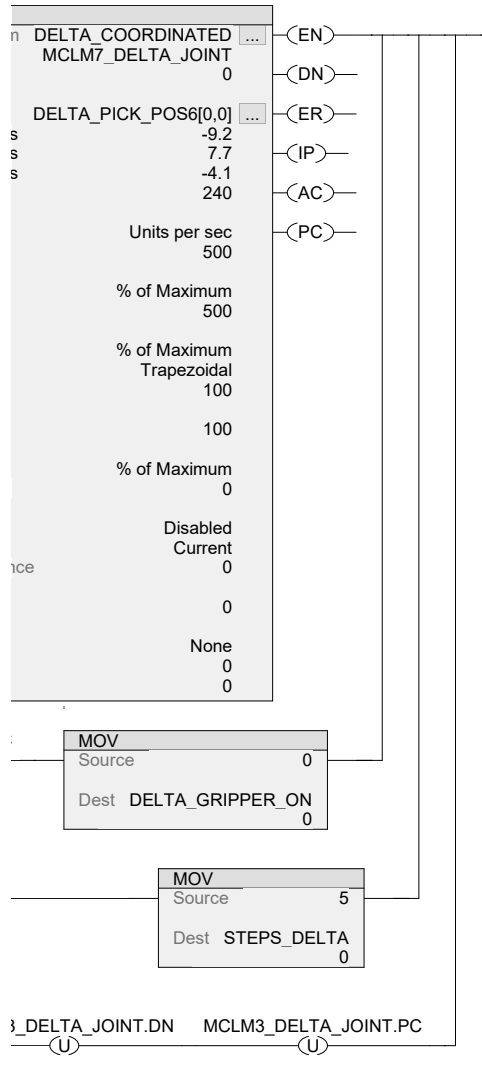
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



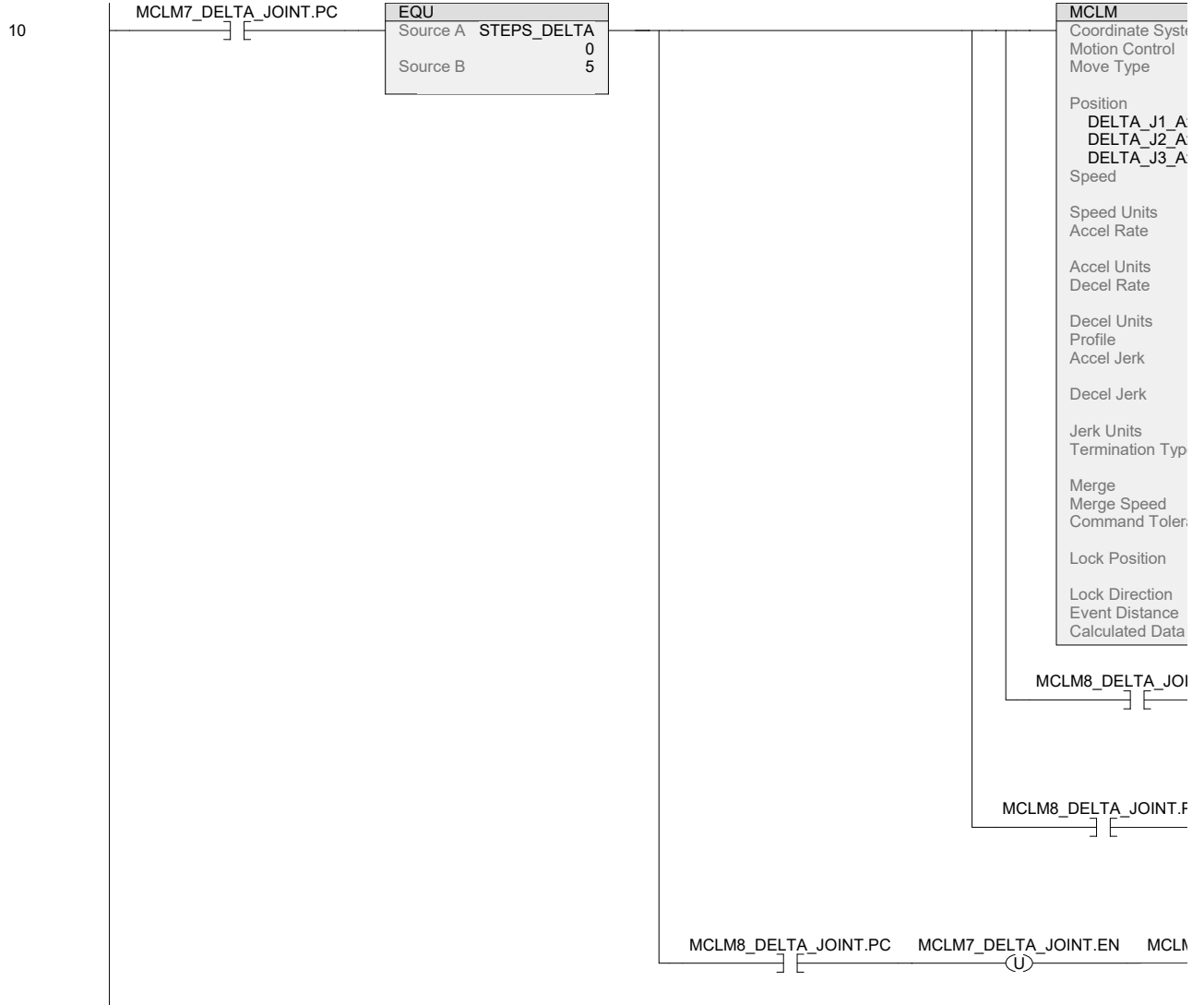
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



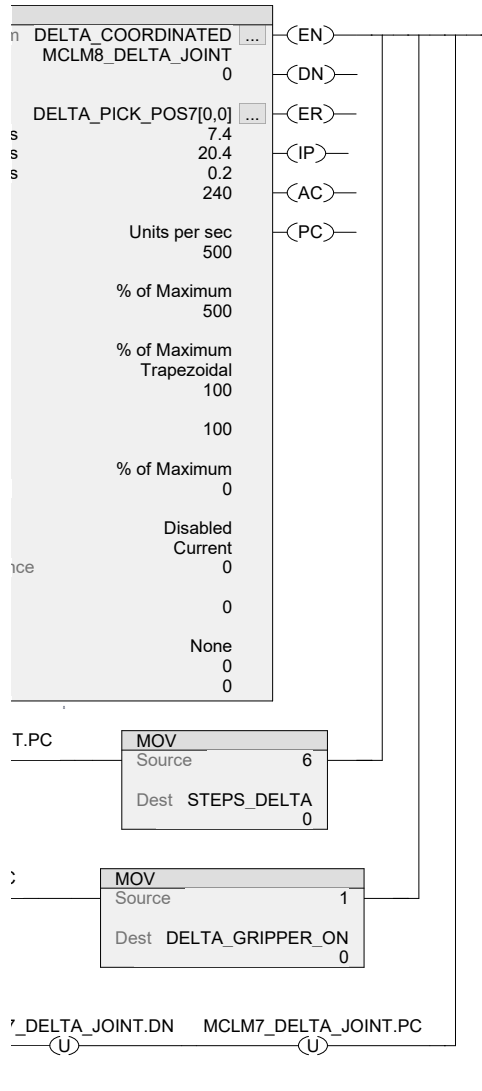
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



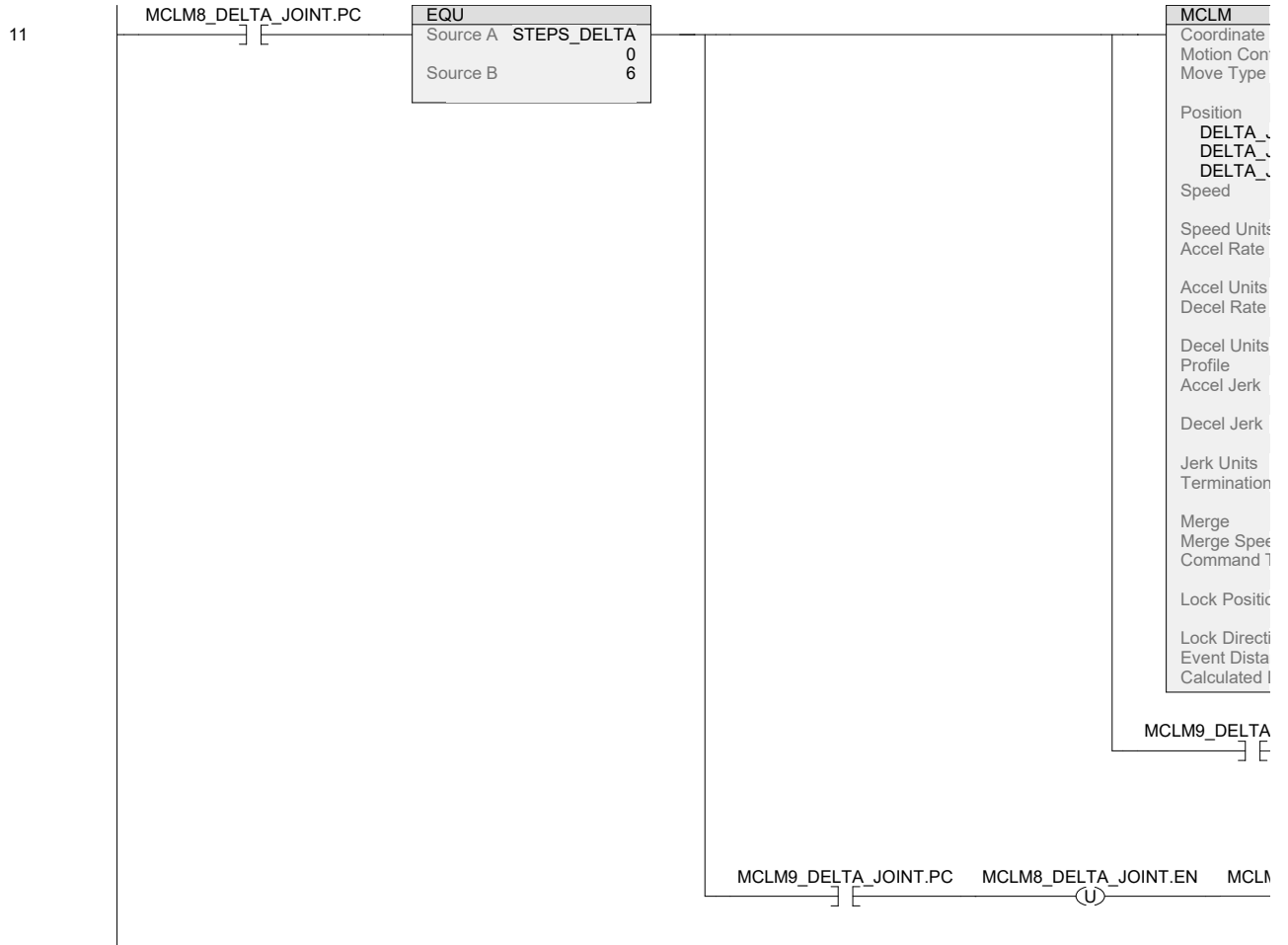
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



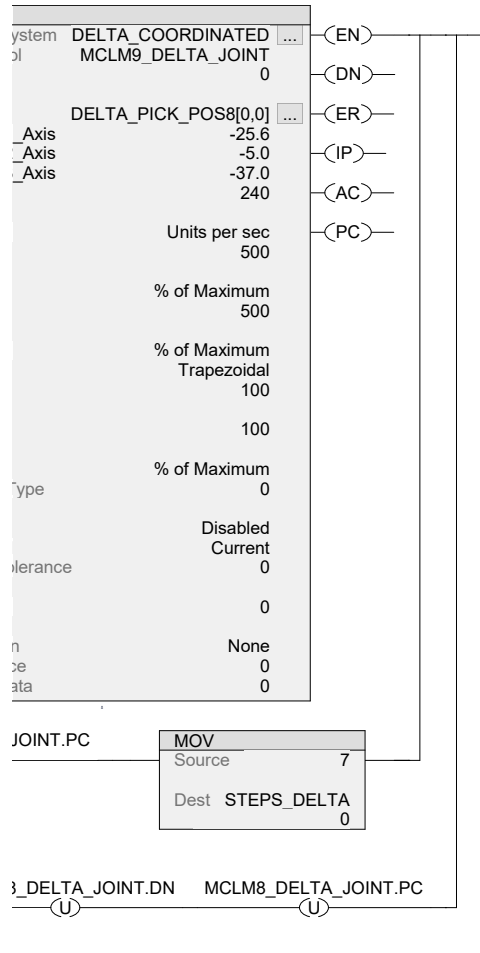
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



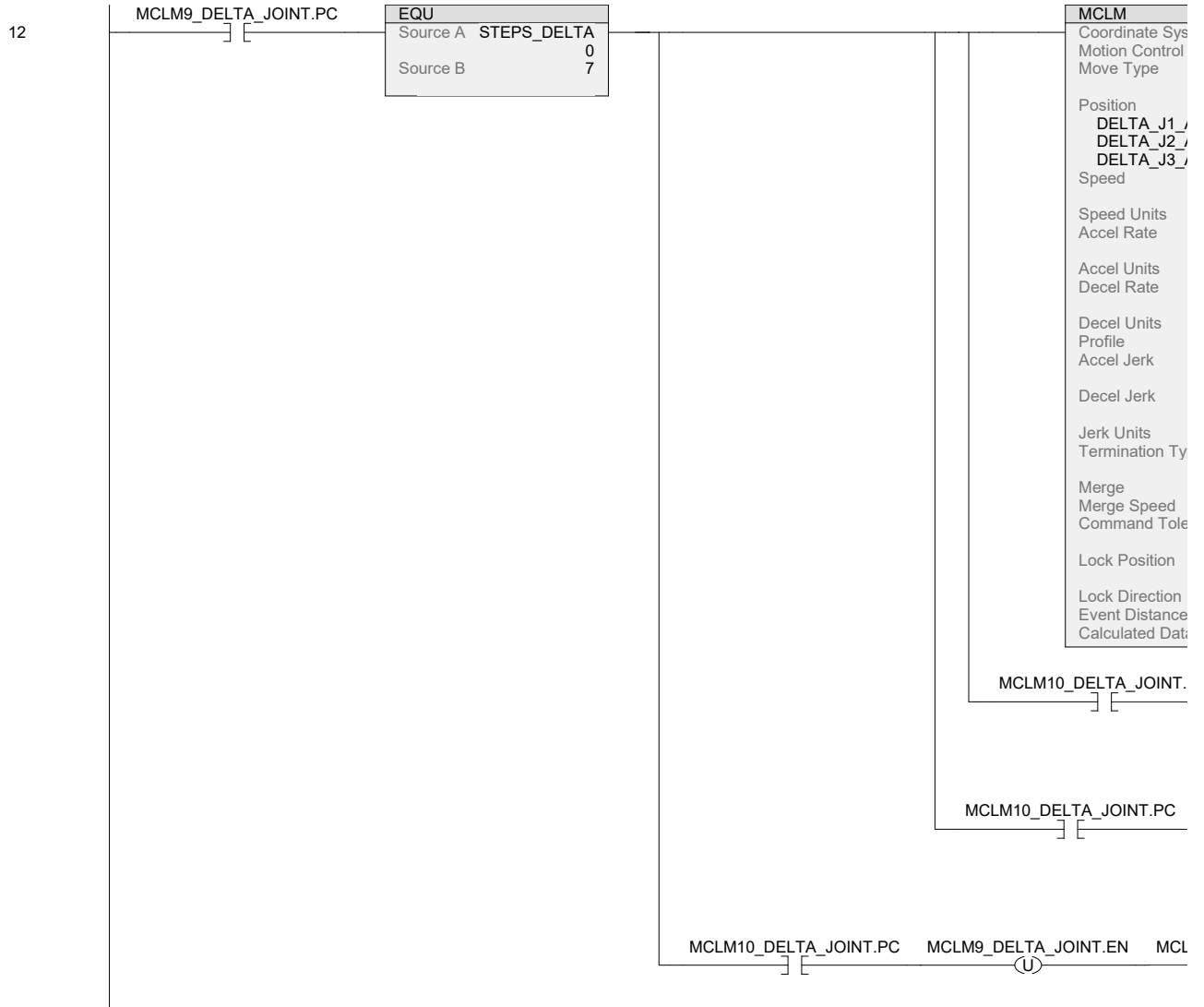
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



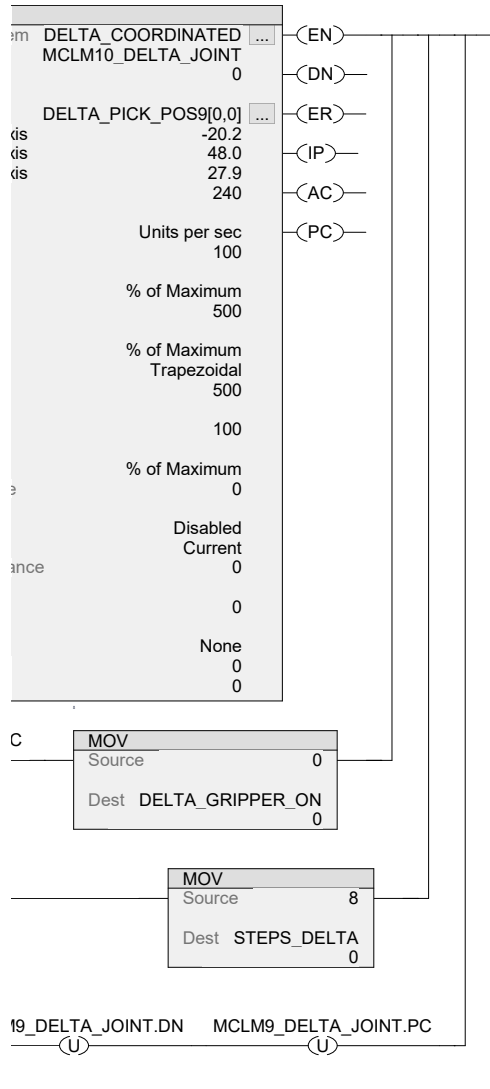
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



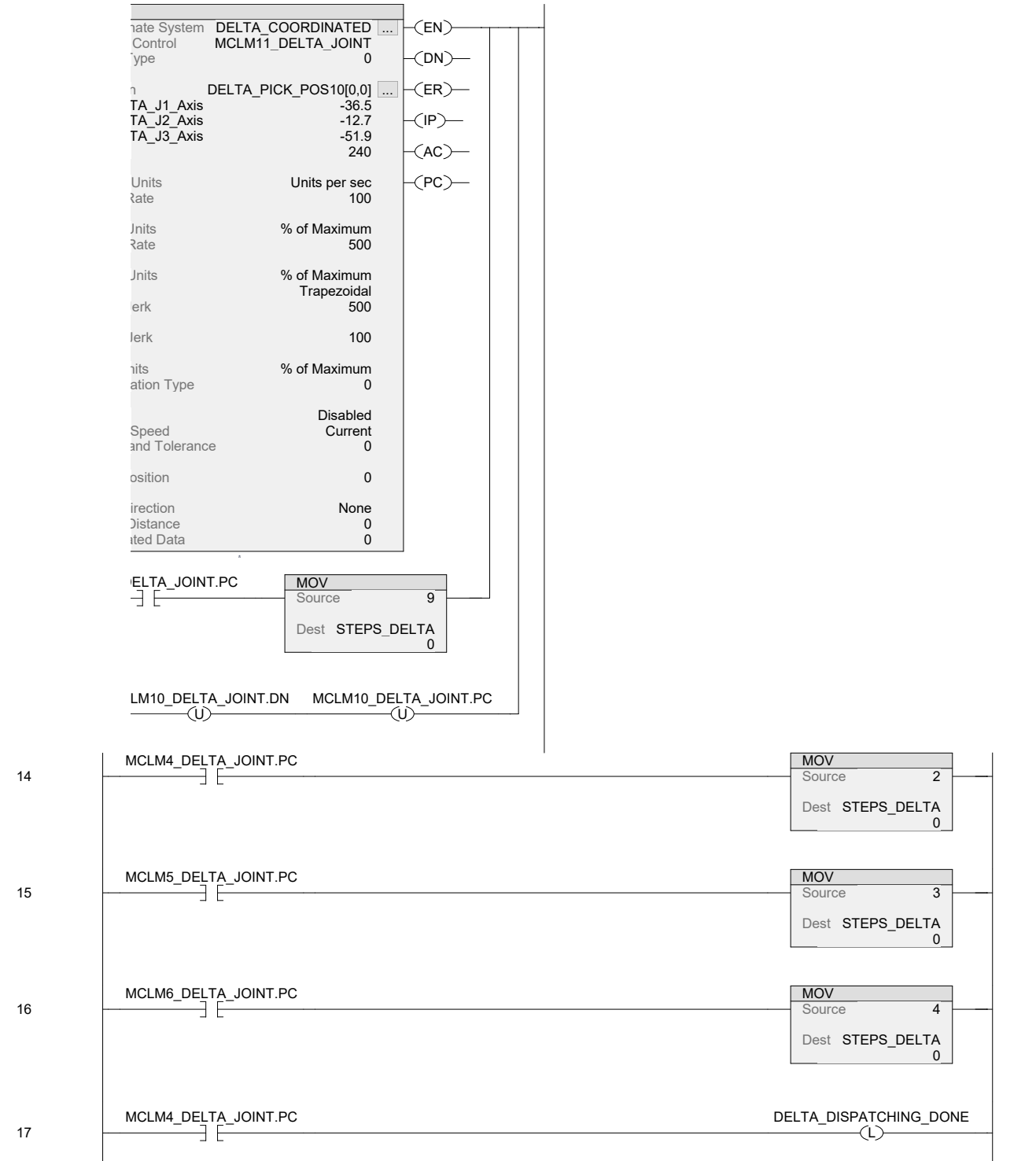
DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

15/07/2020 3:47:23 PM

Total number of rungs in routine: 26

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD



DELTA_PACKING_AUTO - Ladder Diagram

SMV40:MainTask:MainProgram

Total number of rungs in routine: 26

15/07/2020 3:47:23 PM

C:\Users\WorkStation\Documents\Studio 5000\Projects\SMV40.ACD

