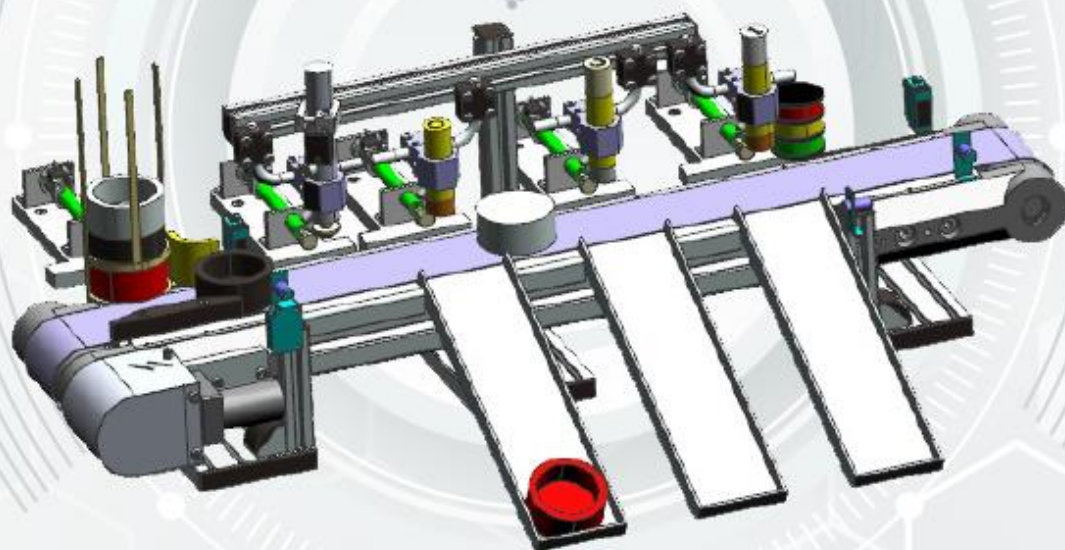




LABORATORIOS VIRTUALES PARA AUTOMATIZACIÓN



Tania J. Ortiz
Victor M. Pérez

Luis A. Carmona
Germán R. Esparza

Laboratorios Virtuales para Automatización

1ª Edición

**Tania Judith Ortiz Ortiz
Luis Alberto Carmona Martínez
Víctor Manuel Pérez Pérez
y Germán Roberto Esparza Sedas**

“Laboratorios Virtuales para Automatización”

Tania Judith Ortiz Ortiz

Luis Alberto Carmona Martínez

Víctor Manuel Pérez Pérez

Germán Roberto Esparza Sedas

© Asociación Mexicana de Mecatrónica A.C.
Prolongación Corregidora Norte No. 168, interior E
Col. El Cerrito, C.P. 76160, Querétaro, Qro.
México.

Internet: <http://www.mecamex.net>

1ª Edición, 27 de abril, 2023.

ISBN: 978-607-9394-26-4

Derechos reservados.

Esta obra es propiedad intelectual de sus autores y los derechos de publicación han sido legalmente transferidos a la editorial. Las opiniones y la información que se muestran en los capítulos del libro son responsabilidad única y exclusivamente de los autores y no representan la postura de la Asociación Mexicana Mecatrónica A.C. Prohibida su reproducción total o parcial por cualquier medio sin la autorización por escrito del propietario de los derechos del copyright.

Impreso en México – Printed in Mexico

Agradecimientos de los Autores

El finiquito de este libro ha sido posible gracias a la visión de proyectar a la Universidad Tecnológica de Querétaro como líder y pionera en implementar el modelo de trabajo conjunto de triple hélice Academias-Estado-Empresas en el Estado de Querétaro, a través de importantes convenios como lo es con la empresa Alemana Siemens PLM, logros muy loables por parte del Centro de Innovación y Creatividad I4.0 de la misma UTEQ.

Los softwares empleados nos permitieron impulsar con seguridad el uso de Gemelos Digitales para el estudio del producto digital mediante simulaciones para el desarrollo de competencias profesionales en el área de Automatización Industrial; herramientas muy pertinentes durante la pandemia de Covid-19, donde el trabajo en línea, uso de laboratorios virtuales en Ingeniería y trabajo colaborativo aportó valioso material, el cual facilitó las cátedras vía remota y nos permitió también vivir la grata experiencia de compartir los saberes y la práctica entre instituciones y empresas.

Gracias a nuestro H. Rector José Carlos Arredondo Velázquez por su visión, al Ing. Joaquín Amaro Cisneros, por su amable orientación como representante de las Alianzas Académicas y Estrategias por parte de Siemens PLM y al coordinador del CIC I4.0 el Lic. Bernardo Carranza Vázquez por su tan acertado convenio.

Índice

PRÓLOGO	1
Los Autores	3
<i>Capítulo I</i>	
<i>LA TRANSFORMACIÓN DIGITAL EN LA FORMACIÓN SUPERIOR</i>	5
I.1 Introducción	6
I.2 Antecedentes	7
I.3 Problemática General	9
I.4 Laboratorios Virtuales de Ingeniería	9
I.5 Marco Teórico	10
I.6 Industria 4.0	11
I.7 Gemelo Digital	13
I.8 Objetivos Académicos y el Modelo de Formación Educativa Dual	14
I.9 Definición del Modelo de Educación Dual del TecNM	14
I.10 Laboratorios Físicos vs Laboratorios Virtuales de Ingeniería	15
I.11 Metodología Laboratorios Virtuales de Ingeniería	17
I.12 Beneficios de la utilización de Laboratorios Virtuales de Ingeniería	20
I.13 Menor costo de implementación y mantenimiento	21
Hemerografías digitales para consulta que complementan al Capítulo I	23
Industria 4.0	23
<i>Capítulo II</i>	
<i>COMISIONAMIENTO VIRTUAL CON NX MCD</i>	24
II.1 Introducción	25
II. 2 Flujo de trabajo de NX MCD	27
II. 3 Menú e iconos de la herramienta NX MCD ver 12.0	28

Índice

II. 4 Navegador de Física	29
II. 4.1 Mecánico	29
II. 5 Barra de recursos	37
II. 6 Configuración de los parámetros para la simulación	38
Bibliografía Capítulo II	41

Capítulo III

CASO PRÁCTICO

ESTACIÓN DE PROCESAMIENTO	42
III.1 Función de la estación de trabajo de procesamiento didáctica, Marca Festo	43
III.2 Precomisionamiento virtual	47
Bibliografía Capítulo III	80

Capítulo IV

CO-SIMULACIÓN CON NX MCD-TIA

PORTAL-PLCSIM ADVANCED	81
IV.1 Co-Simulación con NX MCD, TIA Portal y PLCSIM Advanced	82
IV.2 Características generales de PLCSIM Advanced	83
IV.2.1 S7-PLCSIM V14	83
IV.2.2 S7-PLCSIM V5.x	84
IV.3 Requerimientos para la simulación	84
IV.4 PLC Virtual dado de alta para la co-simulación	86
IV.5 Configuración para el intercambio de datos entre el PLCSIM Advanced y NX MCD	89
IV.6 Programa en escalera para la co-simulación	104
IV.7 Co-simulación	110
Bibliografía Capítulo IV	113

Índice

Capítulo V

INTRODUCCIÓN AL USO

DEL SOFTWARE SIMIT 114

V.1 Introducción	115
V.2 Instalación y descarga del software SIMIT	116
V.3 SIMIT SP Demo	122
V.4 Herramientas de la interfaz de SIMIT	124
V.5 Cómo empezar	125
V.6 La interfaz gráfica de usuario de SIMIT	125
V.7 Navegación de proyectos	126
V.8 Configuración con TIA Openness para poder realizar comunicación usando couplings	127
V.9 Agregar usuarios al grupo de usuarios "Siemens TIA Openness"	128
V.10 Interacción con otros softwares como S7-PLCSIM Advanced y Siemens NX MCD	
(Mechatronics Concept Designer)	131
V.10.1 S7-PLCSIM Advanced	132
V.10.2 Co-simulación NX MCD y SIMIT	135
V.11. Descripción proyecto ejemplo: “Selector de cajas por peso”	136
V.11.1 Secuencia Lógica	137
V.11.2 Desarrollo en SIMI T	140
V.11.3 Programación en el software TIA Portal	140
V.11.4 Generación de un Proyecto de Simulación – SIMIT	142
V.11.5 Conexión con PLCSIM	142
V.11.6 Construcción del Panel de Control	144
V.11.7 Construcción de las animaciones de los elementos	152
V.11.8 Creación de funciones del sistema	156
V.11.9 Implantación de las Funciones en las Animaciones	159
V.11.10 Simulación planta virtual	163

Índice

V.11.11 Puesta en marcha virtual de SIMATIC/SIMOTION con hardware en el bucle (HiL)	166
V.11.12 Principio de operación	167
V.12 Unidad SIMIT	170
V.12.1 SIMIT	170
V.13 Mechatronics Concept Designer (MCD)	171
V.14 Código programado en lenguaje KOP a través del software TIA Portal	172
Bibliografía Capítulo V	183

Capítulo VI

CASO PRÁCTICO ROBOT SCARA 184

VI.1 Introducción	185
VI.2 Desarrollo	187
VI.2.1 Configuración del gemelo digital SCARA en Siemens NX MCD	187
VI.2.2 Conceptos teóricos de configuración	197
VI.2.3 SIMIT Bloque equivalente	197
VI.2.4 TIA Portal Objeto Tecnológico	201
VI.2.5 Configuración TIA Portal (accionamientos Sinamic V90)	208
VI.2.6 Configuración objetos tecnológicos	213
VI.2.7 Integración de bloques de control de Motion Control en TIA Portal	216
VI.2.8 Desarrollo de interfaz de HMI	221
VI.2.9 Configuración en SIMIT para los ejes 1 y 2 y sus encoders	231
VI.2.10 Integración de señales provenientes de NX MCD	240
VI.2.11 Simulación de Axis 1 y Axis 2 en NX MCD	244
VI.2.12 Configuración de Axis 3 y Pinza	251
VI.2.13 Código para el eje 3 y pinza	257
VI.2.14 Interfaz HMI modo automático y tabla de variables generales	258
VI.2.15 Secuencia final de modo automático	263
VI.2.16 Bloque lógico FC modo automático	264

PRÓLOGO

Si usted es una persona que intenta ampliar sus conocimientos sobre el diseño por computadora de automatismos, la evaluación de máquinas automáticas a través de escenarios virtuales y el potencial que presenta la integración de tecnologías orientadas a la industria del Internet de las Cosas (IIoT) puedo afirmar sin temor a equivocarme que en este libro encontrará información muy valiosa sobre aspectos prácticos que le permitirán desarrollar habilidades especializadas en el marco de lo que hoy denominamos “Metaverso Industrial”.

Como antecedente a una sincera y profesional opinión sobre el material que está a punto de conocer, he de aplaudir el gran trabajo y el compromiso que asumieron los autores cuando decidieron realizar este proyecto, ya que esta tarea los llevó a integrar habilidades y experiencias que lograron plasmar mediante elementos didácticos que facilitan la comprensión del diseño de automatismos mediante la creación de escenarios virtuales de máquinas de tipo industrial. Dicha tarea la efectuaron por cerca de un año sin descuidar sus compromisos en sus actuales trabajos, sacrificando sin duda horas de descanso y convivencia familiar, entre otras cosas. Valga este espacio para agradecerles su dedicación y entusiasmo en realizar este libro en un tema de vanguardia que hasta el momento ha sido poco desarrollado en Instituciones Educativas de habla hispana.

Vale la pena reconocer que después de una década de iniciar la cuarta revolución industrial, aún hace falta mucho trabajo por hacer para satisfacer la demanda actual que tienen las industrias en contar con capital humano especializado en tecnologías afines a la Industria 4.0, nos guste o no la evolución tecnológica hacia una nueva etapa ha comenzado por lo que es fundamental lograr que las instituciones educativas se actualicen ante esta necesidad que está demostrando una nueva forma de diseñar y elaborar procesos, productos y servicios mediante el análisis inteligente de grandes volúmenes de información digital. Es por ello que considero que la presente obra facilita la enseñanza de automatismos industriales en un mundo real a través la simulación por computadora de automatismos industriales virtuales llamados: gemelos digitales.

Me parece oportuno y adecuado indicar que el libro va llevando poco a poco al lector a una comprensión de los elementos indispensables para conocer y valorar una metodología que permite la creación de laboratorios virtuales para automatización de procesos de tipo industrial. En el primer capítulo se muestran conceptos y definiciones que son de gran valía a lo largo de los siguientes temas, así como una exposición sobre aspectos educativos a nivel superior y de postgrado que permite reconocer ventajas y sinergias entre la educación dual y la virtualización de laboratorios en la enseñanza de ingenierías afines a la mecatrónica. Así mismo, desde mi punto de vista también es un material de utilidad para quienes toman decisiones estratégicas en instituciones educativas de nivel superior y postgrado.

En el capítulo 2 se expone la forma en que se construyen ambientes virtuales para simular situaciones propias de un proceso automatizado, indicando aspectos fundamentales de configuración de la herramienta NX MCD ver. 12.0 así como la estructura de trabajo

relacionada con a) Ajustes y pruebas de funcionalidad de sensores y actuadores virtuales (*precomisionamiento*), b) Ajustes y pruebas del funcionamiento del sistema automático virtual (*comisionamiento*), c) Ajustes y pruebas sobre el arranque, paro de emergencia y funcionalidad de subsistemas para simular escenarios que garanticen un trabajo eficiente y controlado (*puesta en marcha*).

Con el propósito de constatar la eficacia de los ambientes virtuales en la automatización, en el capítulo siguiente se presenta un caso práctico de comisionamiento que permite conocer y valorar una estación de manufactura flexible representativa de la industria 3.0, la cual fue construida a detalle para lograr lo mejor posible escenarios virtuales que representen el comportamiento físico de la estación de trabajo en el mundo real.

En el capítulo 4 se expone una experiencia que a mi juicio es de gran valía para aquellas personas interesadas en integrar diferentes herramientas digitales orientadas a la automatización, ya que se expone la simulación conjunta (*co-simulación*) del sistema electromecánico creado en el capítulo anterior vinculado a una programación mediante herramientas propias de un Controlador Lógico Programable (PLC).

Tomando en cuenta que el uso de motores eléctricos es prácticamente indispensable en las industrias que desarrollan procesos, productos y servicios, es entendible que el tema de motores eléctricos es ampliamente estudiado en las escuelas y facultades de ingeniería. En este sentido, el capítulo 5 muestra una introducción al uso de un software que permite la co-simulación de componentes de control de potencia para algunos de los motores que son ampliamente utilizados en la industria. De esta forma, se exponen las bases para crear animaciones en escenarios virtuales donde pueden converger sistemas mecánicos accionados por motores eléctricos.

El último capítulo desarrolla otro caso práctico de co-simulación para aquellas personas interesadas en los manipuladores industriales. Un robot tipo SCARA es construido de forma virtual a detalle utilizando las herramientas de software expuestas en los capítulos anteriores, de esta forma se muestra el proceso de diseño, configuración y evaluación de la manipulación robótica mediante las herramientas de software utilizadas en capítulos anteriores.

Finalmente, he de mencionar que el libro desarrolla temas innovadores en la enseñanza actual de la automatización industrial, lo cual ha sido posible gracias al apoyo de la Universidad Tecnológica de Querétaro y su Centro de Innovación y Creatividad 4.0, así como de la empresa Siemens PLM. Logrando con ello una sinergia que plasma el talento y visión de los autores en esta obra que está a punto de conocer.

J. Emilio Vargas
Fundador
Asociación Mexicana de Mecatrónica A.C

Los Autores



**Víctor Manuel
Pérez Pérez**

Maestro con dos títulos en Ingeniería: Civil otorgado por la Universidad Autónoma de Puebla e Ingeniero Mecánico, título otorgado por el Instituto Tecnológico de Puebla; cuenta también con el grado de maestría en Ingeniería Mecatrónica conferido por el CENIDET (Centro Nacional de Investigación y Desarrollo Tecnológico). Responsable de varios proyectos PEI- CONACYT realizados en el estado de Sinaloa para el área agrícola y rehabilitación dentro del periodo 2015 al 2018. Ha participado en proyectos de automatización para la iniciativa privada realizando automatización en procesos agrícolas y de alimentos. Actualmente trabaja en el Instituto Tecnológico de Culiacán en el área de Ingeniería Mecatrónica impartiendo las materias de especialidad como lo es la de redes industriales y de automatización, cuenta con 13 años de experiencia como docente.

Pionero en la implementación de prácticas como herramientas didácticas con Comisionamientos Virtuales, metodologías de diseño propias de I4.0 en el Tecnológico Nacional de México (TecNM) Campus Culiacán.

E-mail: victor.pp@culiacan.tecnm.mx



**Tania Judith
Ortiz Ortiz**

Ingeniera en Electrónica egresada de la Universidad Autónoma Metropolitana con Maestría en Ingeniería en Sistemas Electrónicos y Computacionales (con título) por la Universidad de la Salle del Bajío. Obtuvo mención honorífica en sus estudios de licenciatura. Cuenta con un historial laboral de 25 años como docente a nivel de educación superior en instituciones como la Universidad Cuauhtémoc campus Querétaro y actualmente en la Universidad Tecnológica de Querétaro (UTEQ), participando en la formación en competencias profesionales de jóvenes queretanos en los programas educativos de Mantenimiento Industrial, de la licenciatura en Sistemas Informáticos, de Electrónica, de Automatización y Mecatrónica; en cuanto a la capacitación industrial colaboró con el Colegio Nacional de Educación Profesional Técnica campus San José Iturbide (Qro). En campo colaboró en dar soluciones para la integración en control y automatización industrial, además de ventas técnicas en el área de automatización en sus primeros 7 años al egresar.

E-mail: tania.ortiz@uteq.edu.mx



**Germán Roberto
Esparza Sedas**

Ingeniero físico egresado de la Universidad Autónoma Metropolitana cuenta con Especialización en Dirección de Operaciones y Maestría en Ingeniería (en proceso de titulación) por la Universidad Panamericana; Maestro de Francés por la UNAM y Diploma de Administración por Micro computación por el ITAM. Obtuvo el premio anual de ciencias básicas e ingeniería otorgado por la UAM en 1991. Cuenta con cerca de 30 años de experiencia combinada entre academia en instituciones como la UNAM, Universidad Anáhuac y la Universidad Tecnológica de Querétaro y laboral en las empresas IBM, Dassault Systemes, Oracle y Siemens.

Cuenta con amplia experiencia en el desarrollo de soluciones para Seguridad Informática, Gestión de Servicio y Activos, Gestión del Ciclo de Vida del Producto (PLM), Industria 4.0, Digitalización.

Ha participado en el desarrollo del portafolio de software para digitalización de Siemens para centros de maquinado CNC en Latinoamérica en Siemens Industry Software y participa activamente en la difusión e integración de soluciones de software para la academia.

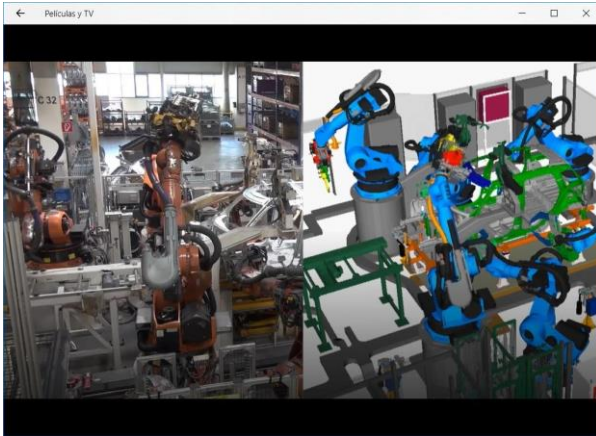
E-mail: german.esparza@uteq.edu.mx



**Luis Alberto
Carmona
Martínez**

Ingeniero en Mecatrónica egresado del Instituto Tecnológico Superior de Poza Rica, cuenta con cerca de 6 años de experiencia académica en la Universidad Tecnológica de Querétaro (UTEQ) y cerca de 10 años de experiencia en programación de equipo industrial, además de una amplia experiencia en programación de controladores industriales de distintas marcas, programación de sistemas embebidos, aplicaciones para internet industrial de las cosas y redes e interfaces industriales.

E-mail: alberto.carmona@uteq.edu.mx



Capítulo I

LA TRANSFORMACIÓN DIGITAL EN LA FORMACIÓN SUPERIOR

En el ámbito industrial tanto en la I3.0 y más aún en la I4.0, la alta gestión de proyectos debe considerarse un determinado método de administración que establezca y utilice un conjunto de procedimientos y herramientas con el fin de estandarizar los procesos, para un uso eficiente del tiempo y los recursos involucrados para el diseño CAD, validación, actividades clasificadas como: precomisionamiento, comisionamiento y puesta en marcha en ambientes virtuales, aportando ejemplos de análisis de problemáticas reales simuladas para los Institutos de Educación Superior (IES), además de contribuir con material con el fin de reforzar las competencias técnicas en las empresas que colaboran en procesos de formación Dual y/o capacitación de su propio personal

***Competencias blandas
y técnicas actualmente
requeridas:***

***-Trabajo concurrente,
colaborativo
interdisciplinario.***

***-Gemelos Digitales
completos y/o
funcionales.***

***-Validaciones de multi
físicas de los modelos
3D (CAE).***

***-Co-simulaciones para
validar el
funcionamiento del
Gemelo Completo o
funcional con su
control***

I.1 Introducción

A raíz de la globalización, la cercanía con los Estados Unidos de Norteamérica y principalmente a la firma de tratados de libre comercio con distintas regiones del mundo, México se ha posicionado como un país atractivo para la inversión de empresas multinacionales de alta tecnología para la manufactura de sus productos por su alta calidad de mano de obra y costo.

Estas industrias incluidas principalmente la automotriz, aeronáutica, electrónica, farmacéutica y sus cadenas de proveedores, se encuentran solicitando a nivel nacional más ingenieros y técnicos con una formación y conocimiento de las nuevas tecnologías, entre ellas Industria 4.0, IIoT, la nube, ciberseguridad, calidad, logística, finanzas y gestión del ciclo de vida del producto (PLM) entre otras.

El Instituto Tecnológico de Gustavo A Madero (ITGAM) y la Universidad Tecnológica de Querétaro (UTEQ), fueron dos instituciones académicas públicas seleccionadas para introducir un piloto en la práctica docente los laboratorios virtuales, por ser parte esencial de la estrategia de educación superior del gobierno mexicano para satisfacer la demanda de cuadros técnicos a nivel superior e ingenieros que demanda el país.

La estrategia didáctica accionada con el diseño de “Laboratorios Virtuales de Ingeniería”, en lo sucesivo LVI, representa una innovación educativa que permitirá integrar tecnologías computacionales junto con conceptos de Industria 4.0, con la intención de superar las barreras de actualización en el diseño de prácticas alineadas a los programas académicos bajo un modelo tecnológico actualizado y flexible permitiendo una mayor cobertura de alumnos a un menor costo que con los laboratorios físicos de manera presencial y/o remota.

Los LVI's proveen a las instituciones educativas de nivel superior (IES) comprometidas en la formación con calidad educativa pertinente de los ingenieros y técnicos superiores en México, de un nuevo modelo de prácticas virtuales con características de flexibilidad tanto en la mejora del diseño en sus contenidos, así como su uso en distintas materias y/o en las subsecuentes, las cuales sean pertinentes y respondan a las necesidades actuales y futuras que demanda la industria nacional de manera eficiente y a un costo accesible que les permita hacerlo de forma sistematizada y sostenible, dichas prácticas son diseñadas bajo el concepto de enseñanza basada en resolución de problemas (PBL), las cuales parten del planteamiento del problema (casos prácticos), resolviendo por lo tanto de lo global a lo particular, facilitando el proceso de enseñanza y aprendizaje por competencias profesionales.

Las prácticas presentadas en este libro integran soluciones de software PLM (Product Lifecycle Management, por sus siglas en inglés) para diseño en 3D (tres dimensiones), tecnología de Industria 4.0 de Gemelos Digitales dentro de los objetivos académicos para conformar los llamados “Laboratorios Virtuales de Ingeniería”, orientados a satisfacer las necesidades de las empresas en términos de formación práctica aprovechando el modelo de

educación dual para sus egresados. Las principales actividades realizadas en ambos institutos fueron:

- a) Determinar qué está demandando la Industria para conservar y lograr su competitividad que se incorporará al diseño de los LVI.
- b) Cuáles son los aspectos académicos que se desarrollan en los estudiantes para lograr las competencias que solicita la industria.
- c) Cuáles son los componentes tecnológicos que permiten satisfacer los puntos a) y b).
- d) Hacer un análisis y mapa de los programas académicos a ser incorporados en los Laboratorios Virtuales de Ingeniería
- e) Determinar las características de Hardware y Software para construir el Gemelo Digital a ser utilizado en el laboratorio

I.2 Antecedentes

En México la Subsecretaría de Educación Superior (SES) a través del Tecnológico Nacional de México (TecNM) y de la Coordinación General de Universidades Tecnológicas y Politécnicas (CGUTyP) cuentan de acuerdo con el “Quinto Informe de Labores 2016-2017 de la Secretaría de Educación Pública” Capítulo IV, con 254 Institutos Tecnológicos con 581,835 alumnos, 115 Universidades Tecnológicas con 242,100 y 62 Universidades Politécnicas con 92,931 alumnos dando un total de 471 escuelas con 916,866 alumnos a nivel nacional abarcando más del 50% del total de los egresados en carreras técnicas superiores a nivel nacional.

Estos se encuentran distribuidos en toda la república sin embargo enfrentan los siguientes retos tal como lo especifica en la gaceta del Senado de la Republica del “Martes 11 de Noviembre de 2014 Gaceta: LXII/3PPO-50/51158” en su página web: <http://www.senado.gob.mx/index.php?ver=sp&mn=2&sm=2&id=51158> , con relación a los Institutos Tecnológicos del País:

“La Secretaría de Educación Pública, dentro de sus facultades y atribuciones tiene a su cargo la Dirección General de Educación Superior Tecnológica (DGEST), la cual a su vez implementa el Sistema Nacional de Institutos Tecnológicos (SNIT); sistema constituido por 263 instituciones, de las cuales 126 son Institutos Tecnológicos federales, 131 Institutos Tecnológicos Descentralizados.

A pesar de que el SNIT atiende aproximadamente a 491 mil jóvenes, enfrenta un déficit tanto de infraestructura y equipamiento como en el número de programas de calidad que imparte, así como de profesores que cuentan con estudios de especialización y posgrado.”

Por otro lado, en los últimos años el Estado de Querétaro ha tenido un crecimiento considerable en el número de empresas de alta tecnología (automotriz, aeronáutica, sector salud, electrónica, entre otras) y su cadena de proveedores que se han establecido ahí, o ampliado sus capacidades productivas aumentando la demanda de ingenieros y técnicos preparados para cubrir los nuevos puestos de trabajo necesarios en la industria.

Por esta razón los LVI se encuentran en el proceso de su implementación en diferentes IES públicas que tienen más impacto a nivel nacional:

- El Instituto Tecnológico de Gustavo A. Madero en los posterior ITGAM.
- La Universidad Tecnológica de Querétaro en lo posterior UTEQ.

La situación en cada caso fue diferente, en el caso del ITGAM es una escuela que por antigüedad (menos de 10 años), disponibilidad de espacio físico y limitación de recursos, la cual aún no cuenta con laboratorios físicos de ingeniería por lo que el laboratorio virtual es la única alternativa para realizar las prácticas de los estudiantes de ingeniería en las instalaciones del instituto.

Mientras que en la UTEQ, siendo una escuela con una mayor antigüedad (más de 25 años) y con un alto prestigio en la ciudad de Querétaro; la necesidad de resolver fue más orientada a ampliar las capacidades de desarrollo de prácticas de laboratorio utilizando tecnología de punta sustentada en conceptos de Industria 4.0 que complemente el centro de innovación y competitividad industria 4.0, (CIC I4.0), inaugurado en el año 2017, de acuerdo con las necesidades de la industria instalada en la región que les permita conservar el liderazgo regional.

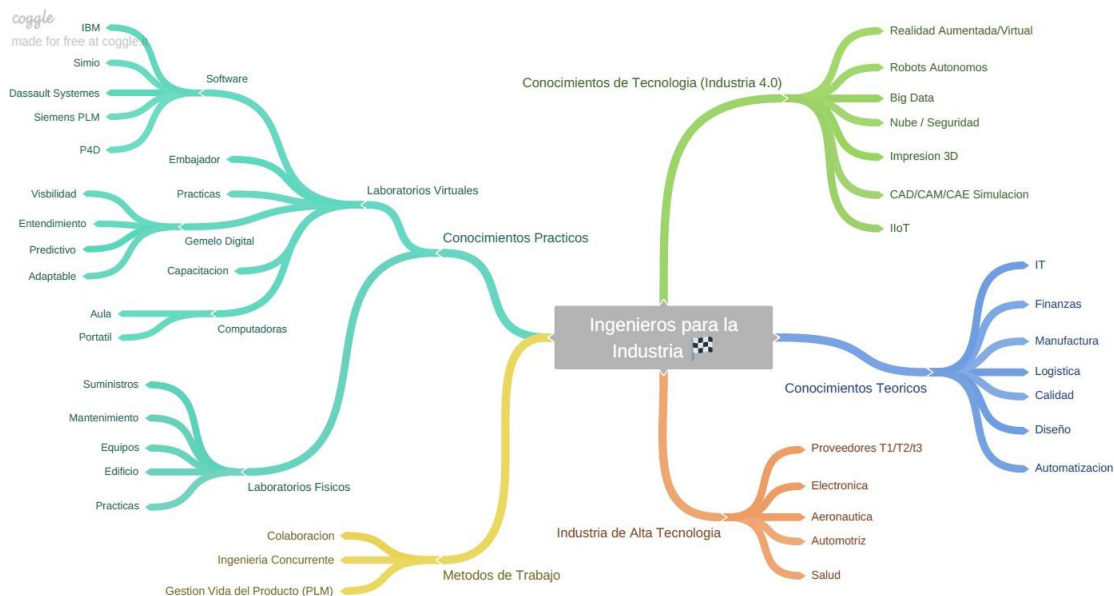


Figura I.1 Mapa de factores relevantes a considerar en la formación actual en Ingeniería.

I.3 Problemática General

La industria requiere de profesionistas técnicos que al incorporarse a la empresa sean capaces de entender y aplicar nuevos conceptos tales como Industria 4.0 para alcanzar o continuar con una posición de liderazgo y competitividad a nivel global.

En México existen por los menos 25 carreras o estudios técnicos superiores relacionados con ingeniería impartidos en las distintas escuelas públicas y privadas algunos ejemplos de ellas ofertadas en los institutos tecnológicos, universidades tecnológicas y politécnicas según sus páginas web: <https://www.tecnm.mx/> y <https://dgutyp.sep.gob.mx/> tales como: mecánica, industrial, logística, aeronáutica, de energías renovables, calidad, ambiental, automotriz, mecatrónica, petrolera, naval, minera, eléctrica, electromecánica, mantenimiento, automatización, procesos industriales y biomédica entre otras.

La mayoría de las escuelas no cuentan con la infraestructura de laboratorios de ingeniería o la capacidad de éstos es insuficiente o no están actualizados, por otra parte ¿cuántas de las prácticas cumplen con el desarrollo de las competencias requeridas por la industria actual? y ¿qué tanto aportan a una formación integral, actualizada y flexible? con el objetivo de satisfacer la formación de ingenieros y técnicos superiores que se integren de forma eficiente al entorno laboral actual y futuro que necesitan las empresas para mantenerse competitivas, ya que éstas deben invertir en capacitaciones especializadas en los egresados contratados o buscan profesionistas fuera de México que tengan los conocimientos sobre el uso de herramientas de PLM e Industria 4.0.

Es importante mencionar que las universidades tecnológicas cuentan con procedimientos periódicos de diálogos con empresarios, en los cuales validan si las competencias profesionales al egresar son pertinentes a las necesidades industriales y sobre qué es necesario actualizar en los programas educativos; las últimas actualizaciones en los programas educativos (PE) sobre tópicos de I4.0 fueron en el año 2018.

Por otra parte, los procedimientos de compras de equipos industriales, didácticos y de software especializado en las universidades, son muy lentos y desfasados en tiempo al surgimiento y aplicación de las tecnologías, incluso de aquellas conocidas como emergentes.

I.4 Laboratorios Virtuales de Ingeniería

Los LVI tienen las siguientes ventajas competitivas en comparación a los laboratorios físicos:

- Mayor participación de alumnos en trabajo en equipo o de manera individual.
- Se aplican métodos y tecnologías en software para comisionamiento virtual en el diseño del producto, diseño de maquinaria y en el diseño de procesos industriales; métodos que le permiten a la industria ser más productiva y competitiva.
- Son flexibles al poder establecer distintas prácticas de laboratorio que se adecuen a diferentes industrias aprovechando el modelo de educación dual.

-
- No presentan los altos costos de instalación, mantenimiento y obsolescencia de los equipos en los actuales laboratorios físicos.
 - Los Institutos de Educación Superior (IES), por lo tanto, podrán responder de manera más pertinente a la demanda de ingenieros y técnicos superiores, requeridos en las nuevas empresas de manufactura en México.

I.5 Marco Teórico

En el año 2010 el concepto de Industria 4.0 a través del uso de la digitalización de la información de la empresa y combinando las mejores prácticas de diseño y fabricación como PLM han permitido a las empresas lograr mejoras sustanciales en sus procesos para alcanzar una mayor satisfacción de sus clientes a un menor costo y riesgo. El diseño de los Laboratorios Virtuales de Ingeniería propone la integración de los componentes de:

- Industria 4.0 conocido como Gemelo Digital, réplicas exactas o funcionales de productos, maquinaria y plantas productivas principalmente.
- Métodos y herramientas PLM (Product Lifecycle Management) para la construcción del Gemelo Digital.
- Análisis y mapeo de los objetivos académicos para alcanzar las competencias a adquirir a través de las asignaturas seriadas y transversales de las ofertas educativas de cada institución.
- Aplicación pertinente para el modelo de formación dual.

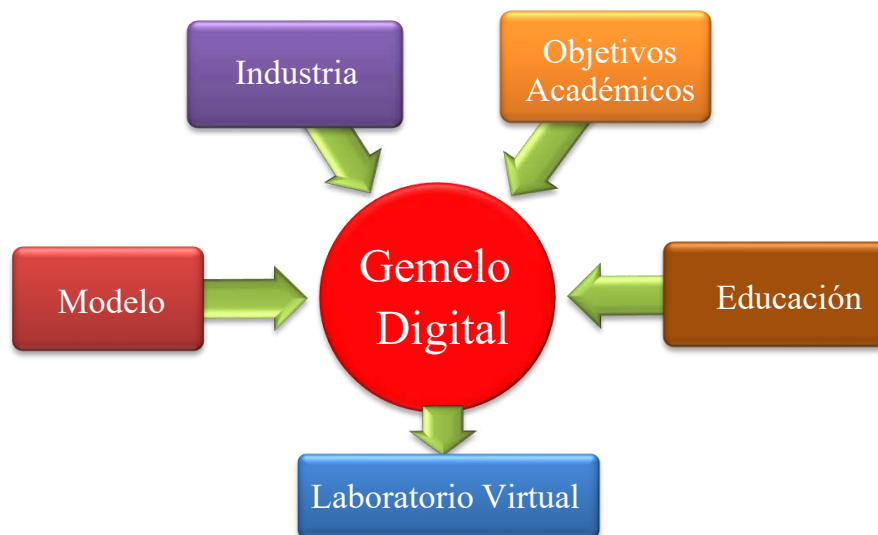


Figura I.2 Integración de los Objetivos Académicos, los conceptos de Industria 4.0 y PLM en el "Gemelo Digital, conforman los LV para Ingeniería.

I.6 Industria 4.0

La cuarta revolución industrial, también conocida como Industria 4.0 surge como una iniciativa del gobierno alemán para hacer más competitiva a su industria. Tal como se cita en la página web de la ZVEI (Asociación Alemana de Fabricantes de Eléctrica y Electrónica) <https://www.zvei.org/en/press-media/publications/implementation-strategy-industrie-40-report-on-the-results-of-industrie-40-platform>

“La Industria 4.0 se sustenta en la digitalización e integración de la cadena industrial de valor. A través de la combinación de tecnologías de información y comunicación con tecnología de automatización para formar el internet de las cosas y aunado con servicios de valor permite un alto nivel de integración y comunicación entre las plantas de producción, los proveedores y el cliente.

Esto permite ir mano a mano con la digitalización de los productos y servicios ofertados permitiendo nuevos modelos de negocio. En última instancia la Industria 4.0 es la realización de la fábrica inteligente en el contexto de la red digital de valor.”

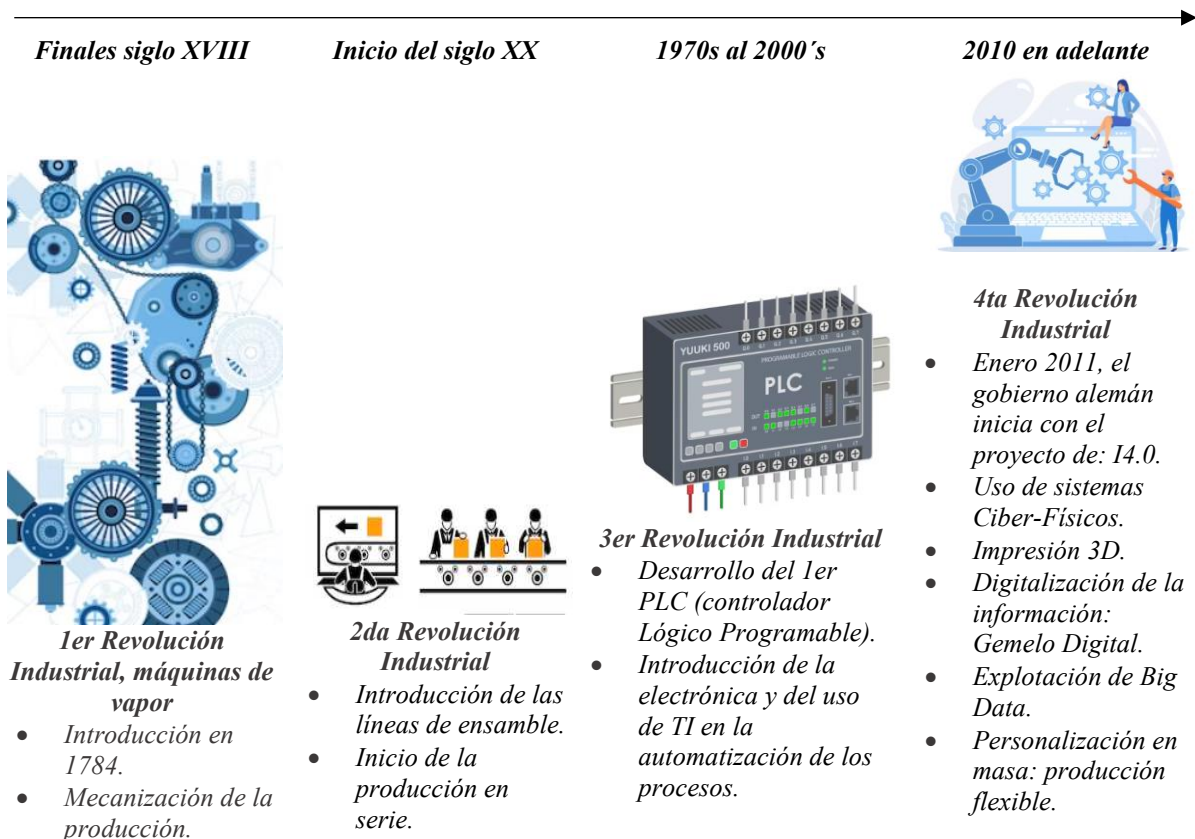


Figura I.3 Como surge la Industria 4.0.

De acuerdo con el “Grupo de Plataforma de Industria 4.0” auspiciado por el Gobierno Alemán en su página web www.plattform-i40.de para lograr la integración de las cadenas productivas de forma digital, la Industria 4.0 se basa en 9 pilares tecnológicos: *Robots Autónomos, la Simulación, la Integración de Software, el Internet Industrial de las Cosas, la Ciberseguridad, la Nube, la Manufactura Aditiva, la Realidad Aumentada y el Big Data & Analytics.*

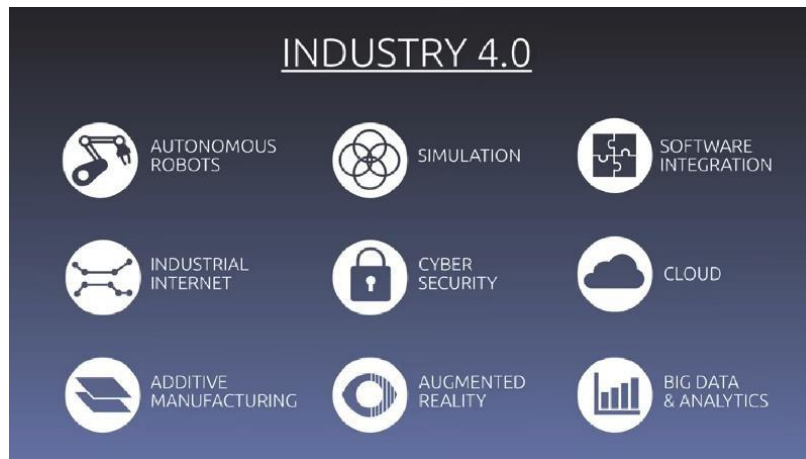


Figura I.4 Tecnologías para Industria 4.0.

La Industria 4.0:

- Conecta y une producción con las tecnologías de información y comunicación.
- Conecta la información de los clientes con la información de las máquinas.
- Las máquinas se comunican con las máquinas.
- Los componentes y las máquinas gestionan de forma autónoma la producción de forma flexible, eficiente y ahorrando recursos.

Obteniendo los siguientes beneficios en la industria:

- Mejorar la calidad.
- Aumentar la flexibilidad.
- Aumentar la productividad.
- Desarrollo estandarizado.
- Reducir tiempo de lanzamiento de productos.
- Evaluación y comparación continua de la industria (benchmarking).
- Competitividad global.
- Creación de puestos de trabajo más atractivos en las áreas de automatización.
- Nuevos servicios y modelos de negocio.

I.7 Gemelo Digital

De acuerdo con el *Dr. Michael Grieves* de la Universidad de Michigan en su artículo “*Digital Twin: Manufacturing Excellence through Virtual Factory Replication*”, el concepto de Gemelo Digital (Digital Twin) se lo atribuye a *John Vickers* cuando la NASA construía modelos análogos en computadora para monitorear y modificar las naves espaciales en 2002. Posteriormente fue introducido por el mismo *Dr. Grieves* en su libro “*Virtually Perfect: Driving Innovative and Lean Products through Product Lifecycle Management*” para ser finalmente adoptado de forma sistemática en la industria a partir de 2012. Un gemelo digital es un modelo virtual de un proceso, producto o servicio, el cual permite aprender en un entorno virtual y trasladar el aprendizaje de lo físico del análisis de datos y el monitoreo de sistemas para evitar problemas antes de que se produzcan. Según *International Data Corporation (IDC)*, para 2018 las empresas que inviertan en tecnología de gemelos digitales verán una mejora de 30% en los tiempos de ciclo de procesos críticos.



Figura I.5 Diferentes Gemelos Digitales en la I4.0.

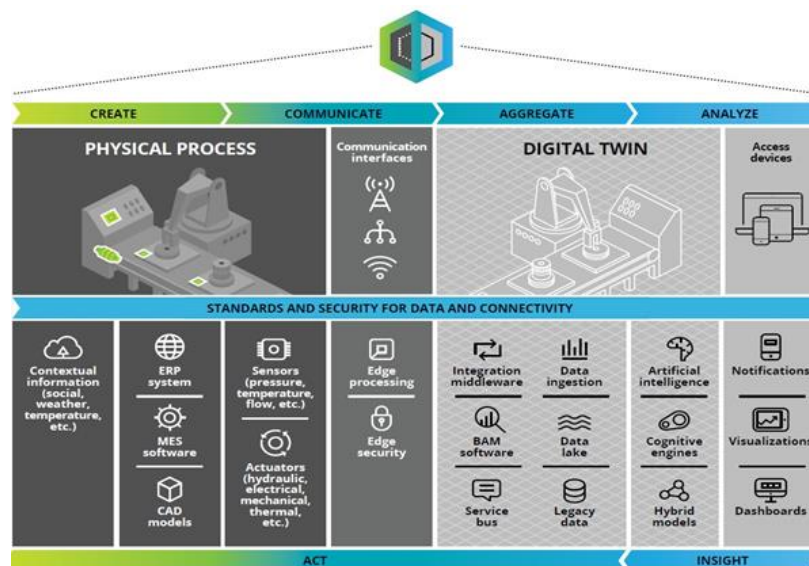


Figura I.6 Deloitte University Press (2018).
Digital Twin Conceptual Architecture, Londres (Reino Unido)
dupress.deloitte.com

El Gemelo Digital nos proporciona durante todo el ciclo de vida del producto:

-
- Visibilidad: ¿Qué está sucediendo?
 - Entendimiento: ¿Por qué está sucediendo?
 - Es Predictivo (Estar preparado): ¿Cuándo y que sucederá?
 - Es Adaptable (Auto-optimizable): ¿Cómo dar una respuesta autónoma?

I.8 Objetivos Académicos y el Modelo de Formación Educativa Dual

Los objetivos académicos por incorporar en el Laboratorio Virtual Digital de Ingeniería deben estar alineados inicialmente con los planes de estudio existentes para las distintas carreras. Sin embargo, en un entorno tecnológico y de negocios dinámico estos objetivos y su forma de lograrlos deben ser flexibles para poderse adaptar a los nuevos requerimientos y complementados con el modelo de formación educativa dual.

Para el Tecnológico Nacional de México el modelo de educación dual se establece desde 2013 de acuerdo con su página web como:

“En el año 2013 la Dirección General de Educación Superior Tecnológica (actualmente TecNM) a través de la Dirección de Docencia presentó la iniciativa para el diseño del Modelo de Educación Dual, iniciando la transformación de este sistema hacia la innovación educativa y fortaleciendo su modelo educativo. Esta implementación del Modelo de Educación Dual, permite al estudiante el desarrollo de competencias en un ambiente laboral que le faculta para actuar de manera pertinente en un contexto específico de su ejercicio profesional, en el que movilice saberes, quehaceres y actitudes tales como la iniciativa, creatividad, ética, liderazgo, trabajo en equipo, compromiso social, emprendedurismo y sustentabilidad...”

I.9 Definición del Modelo de Educación Dual del TecNM

“Con base en el apartado anterior para el TecNM el Modelo de Educación Dual, es una estrategia de carácter curricular flexible que consiste en la adquisición y perfeccionamiento de competencias profesionales del estudiante, definidas en un plan formativo que se desarrolla en ambientes de aprendizaje académico y laboral en coordinación con las organizaciones del entorno, considerando el enfoque y alcance de los perfiles de egreso.

En este sentido la Educación Dual es una manera en que el estudiante forma y desarrolla sus competencias profesionales establecidas en el perfil de egreso comúnmente a lo largo de su estancia en el Instituto Tecnológico, a un esquema donde se convierte en trabajador y algunas competencias las desarrollará en un ambiente laboral y/o científico en un tiempo determinado y preferentemente de tiempo completo.”

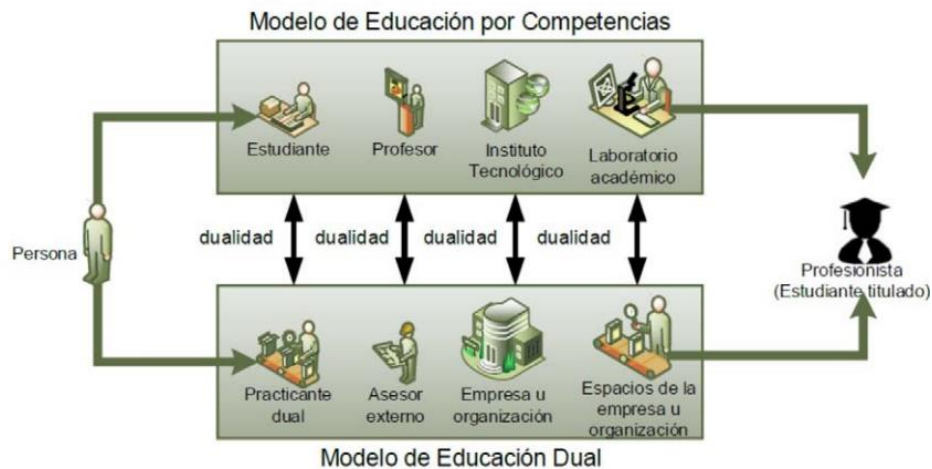


Figura I.7 Modelos Dual en TecNM.

a) Estudiantes:

- Titulación integral,
- experiencia laboral,
- manejo de tecnologías y equipo de vanguardia,
- certificaciones laborales y
- alta probabilidad de contratación.

b) Profesor:

- Estancias,
- actualización profesional,
- manejo de tecnologías y equipo de vanguardia,
- y certificaciones laborales.

I.10 Laboratorios Físicos vs Laboratorios Virtuales de Ingeniería

Los planes de estudio de las carreras técnicas como el de las ingenierías y técnicos superiores complementan los medios de enseñanza aprendizaje a través de experiencias prácticas que realizan generalmente en laboratorios con equipamiento especializado en distintas áreas.

Existen distintos tipos de laboratorios para realizar estas prácticas y para fines de este trabajo nos concentraremos en la comparación de los más usados en las carreras técnicas; los laboratorios físicos que compararemos con esta propuesta de Laboratorios Virtuales de Ingeniería.

Laboratorios Físicos, conocidos como laboratorios físicos, son lo que contienen equipo para prácticas de fabricación de partes, pruebas destructivas, control de calidad, creación y simulación de celdas de manufactura con robots y PLC, centros de manuales de maquinado y/o controlados por computadora, bombas hidráulicas, sistemas de enfriamiento, en general equipos que regularmente se utilizan en la industria de transformación. En estos equipos los estudiantes reproducen las prácticas de laboratorio para reforzar los conceptos aprendidos.



Figura I.8 Equipamiento típico de un Laboratorio Físico.

Laboratorio Virtual de Ingeniería se compone de equipos de cómputo en donde se instala el software para diseño de productos y procesos de fabricación que pretende reproducir un ambiente de producción industrial real y que mediante simulaciones interactivas con el Gemelo Digital desarrollándose las prácticas de laboratorio con el fin de lograr los objetivos de aprendizaje por asignaturas.



Figura I.9 Equipamiento típico de un Laboratorio Virtual de Ingeniería.

Las diferencias entre ambos laboratorios son radicales, mientras en los laboratorios físicos podemos atender a un número muy limitado de estudiantes generalmente entre 5 y 10 por práctica en los laboratorios virtuales de ingeniería podemos atender de 20 a 40 alumnos simultáneamente.

<i>Característica del Laboratorio</i>	<i>Pesado</i>	<i>Virtual de Ingeniería</i>
ATENCIÓN A ALUMNOS SIMULTÁNEAMENTE	5 a 10	20 a 40
ESPACIO REQUERIDO	15 a 30 m2 por equipo	MENOS DE 1 M2 POR EQUIPO
USO Y MANTENIMIENTO	Requiere herramientas, refacciones, lubricantes, calibración, instalación especial, supervisión	SEGURO, SIN SUPERVISIÓN, MANTENIMIENTO MÍNIMO, SIN CONSUMIBLES, SIN INSTALACIÓN ESPECIAL
ACTUALIZACIÓN	ALTO COSTO Y ALTA OBSOLESCENCIA	MUY BAJO COSTO Y FÁCIL ACTUALIZACIÓN

Tabla I.1 Tabla comparativa de laboratorios físicos vs virtuales.

I.11 Metodología Laboratorios Virtuales de Ingeniería

La alternativa más rentable para lograr dar una formación de clase mundial a los futuros ingenieros y técnicos superiores es la utilización de Laboratorios Virtuales de Ingeniería, los cuales permiten la integración de los objetivos de aprendizaje por medio de prácticas de laboratorios simulando mediante softwares especializados de ingeniería las condiciones reales de trabajo en la industria a través de Gemelos Digitales Funcionales y/o réplicas exactas del producto, de la maquinaria, de las líneas de producción. La metodología para el desarrollo de LVD consta de las siguientes etapas:



Figura I.10 Metodología Propuesta para Conformación de Laboratorios Virtuales Digitales.

Para la Universidad Tecnológica de Querétaro (UTEQ), el análisis de los requisitos se ha realizado, a continuación, se enlista los eventos importantes para este punto:

- 2017: Inauguración del Edificio ***Centro de Creatividad e Innovación para Industria 4.0*** (CIC 4.0), \$38 millones MXN, operando bajo el concepto de triple hélice.
- 2017: Inauguración del Centro de Formación Digital ***Ingenuity Lab UTEQ-Siemens***, Pared demostrativa de equipo de control industrial Siemens, \$23 millones USD; y firma Convenio de 100 asientos de licencias de software: ***NX, Tecnomatix y Solid Edge*** principalmente.
- 2019: Desarrollo de 2 laboratorios virtuales.
- 2020-2021: Servidor para la nube de la UTEQ y laboratorio ***PLM-UTEQ***, \$5 millones MXN.
- 2020 Julio: Inauguración del Edificio de Posgrado, ***Investigación, Desarrollo y Emprendimiento Tecnológico*** (PIDET), \$35 millones MXN.
- 2021 May-Ago: Implementación de metodología de ***Comisionamiento Virtual*** con gemelo digital de estación de trabajo: ***NX MCD, TIA Portal y PLCSIM Advanced***.
- 2022 Ene-Abr: Configuración servidor y clientes virtualizados.
- 2022 May-Ago: Aulas digitales acondicionadas tanto en equipo, software instalado y red.
- 2022 Ago: Capacitación a 2 profesores con el rol de administradores de la plataforma Xcelerator Academy.

Como ejemplo del punto 2, se ilustra el análisis para una sola asignatura del plan de estudios de la continuidad de estudios en la UTEQ de la Ingeniería en Tecnologías de la Automatización:

<i>Cuatri- mestre</i>	<i>Asignatura</i>	<i>Competencia asignatura</i>	<i>Unidad Temática</i>	<i>Objetivo Unidad</i>	<i>Software por utilizar</i>
7mo	ELECTRICIDAD INDUSTRIAL	<i>Desarrollar proyectos de automatización y control, a través del diseño, la administración y la aplicación de nuevas tecnologías para satisfacer las necesidades del sector productivo.</i>	III. Instalaciones Eléctricas Residenciales, Comerciales e Industriales.	<i>El alumno explicará los métodos para el diseño de instalaciones eléctricas residenciales, fuentes alternas de generación, cálculo, selección de materiales y componentes de protección comercial e industrial para elaborar proyectos requeridos de la instalación y puesta en servicio conforme a la Norma Oficial Mexicana NOM001-SEDE-2012, y otras normas aplicables (IEEE, ANSI, IEC, DIN).</i>	<i>NX Industrial Electrical Design (Routing, MCAD, ECAD Tools).</i>

Tabla I.2 Análisis asignatura para propuesta de uso de software PLM.

Claro está que dicho análisis se realizó para todos los programas educativos (PE) de la UTEQ divisiones: Industrial, Tecnologías en Automatización e Información y Ambiental, el análisis fue compartido para directores y coordinadores para la gestión de academias y uso de laboratorios de cómputo.

I.12 Beneficios de la utilización de Laboratorios Virtuales de Ingeniería

- Atender a menor costo y con soluciones integrales digitalizadas, la creciente demanda de ingenieros y/o técnicos superiores en el Estado de Querétaro con formación académica hacia Industria 4.0.
- Reforzar el logro de los objetivos académicos, acorde con las necesidades actuales y futuras de la industria en la región.
- El tiempo de implementación y adopción es mucho menor al de los laboratorios físicos (4 meses vs más de 2 años).
- Permiten configuraciones ilimitadas y actualización de los Laboratorios Virtuales de Ingeniería sin requerir espacio físico, equipos que se dañen, o que requieran mantenimiento.

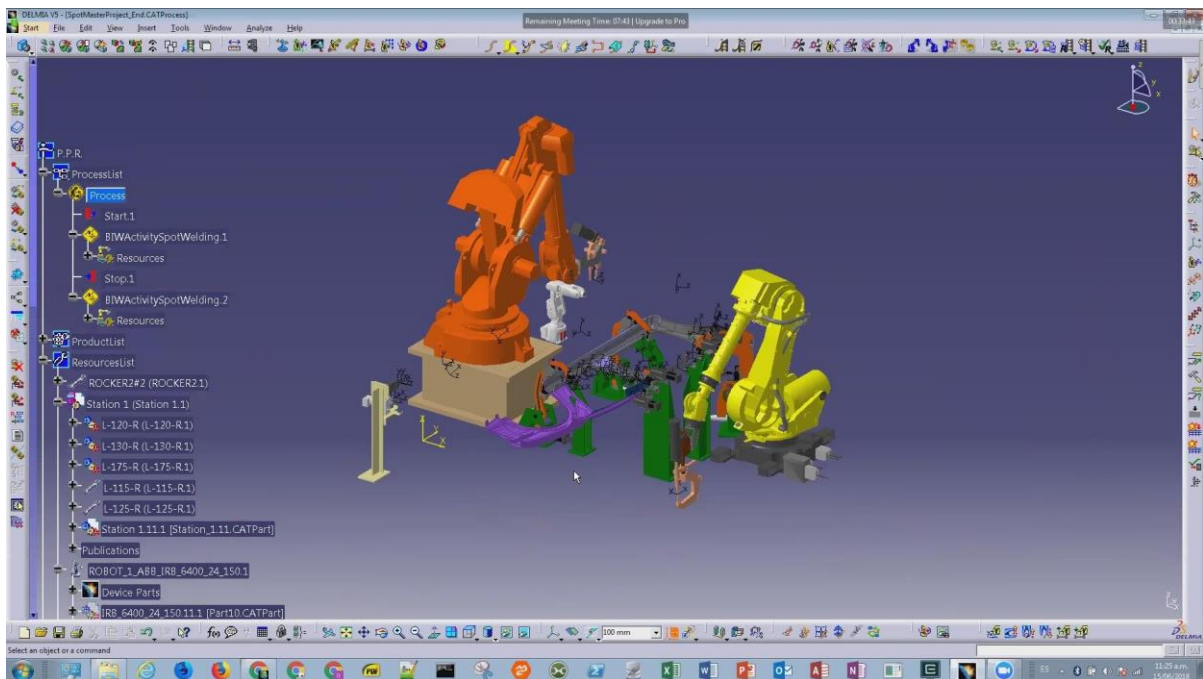


Figura I.11 Ejemplo de Gemelo Digital para práctica de laboratorio de robótica.

I.13 Menor costo de implementación y mantenimiento

A continuación, se presenta en la tabla 2 de manera cuantitativa un comparativo entre laboratorios físicos versus laboratorios virtuales de ingeniería (LVI) sobre los resultados obtenidos en El Instituto Tecnológico de Gustavo A. Madero (ITGAM).



Métricas	Laboratorios Virtuales	Laboratorios Físicos
Inversión	884 unidades	21,000 unidades
Tiempo de implementación	4 meses para los primeros 2 laboratorios	7 meses para desplantar el edificio
Capacidad Laboratorios	200 participantes simultáneos	50 participantes simultáneos
Rentabilidad/costo por usuario SIMULTANEO	4.42 unidades	420 unidades
Rentabilidad/población objetivo (2,000 alumnos)	0.442 unidades	10.5 unidades
Cantidad de gemelos digitales (laboratorios)	Ilimitada	2 o 3 celdas fijas
Cantidad de prácticas por cada laboratorio	Ilimitada	Depende de la celda configurada

Tablas I.3 Ejemplo de unidades de inversión de laboratorios físicos vs laboratorios virtuales.

Se hablará de unidades y no de costos por confidencialidad, el objetivo es visualizar los ahorros en general. Al aplicar estos Laboratorios Virtuales de Ingeniería en el ITGAM y la UTEQ los principales resultados obtenidos son:

- Diseño de nuevas y mayor variedad de prácticas de laboratorio debido a que los laboratorios virtuales prácticamente no tienen límite en la creación de Gemelos Digitales para aplicar en las prácticas específicas.
- Motivación para los alumnos a que realicen prácticas de laboratorio virtuales, en sustitución a las realizadas en los laboratorios físicos y en el caso del ITGAM inclusive no cuentan con laboratorios físicos.
- Mejor entendimiento y vinculación con la industria debido a que se pueden simular problemas reales para entrenar a los estudiantes a analizarlos y resolverlos como se hace en la industria de forma didáctica.
- Mayor alcance de número de alumnos para la realización de sus prácticas de laboratorio, ya que se pueden combinar los centros de cómputo con equipos personales para los estudiantes que cuenten con uno.

-
- Variedad casi ilimitada de equipos y dispositivos a modelar a través de librerías de robots, centros de maquinado o equipos y dispositivos virtuales provistos por el SW utilizado.
 - Reducción costos por mantenimiento o daño a equipos físicos y un tiempo menor de restablecimiento de los laboratorios virtuales.
 - Se elimina la obsolescencia de los equipos físicos adquiridos por falta de refacciones o servicio de mantenimiento a equipos viejos.
 - El costo de montar y operar un laboratorio virtual es menor que uno físico pues solo se necesita el convenio de utilización de Software, el equipo de cómputo en donde se instalará el Software y se realizaran las prácticas.

Posteriormente, el poder utilizar tecnología de Realidad Aumentada que permite interacción entre el estudiante y el modelo virtual.

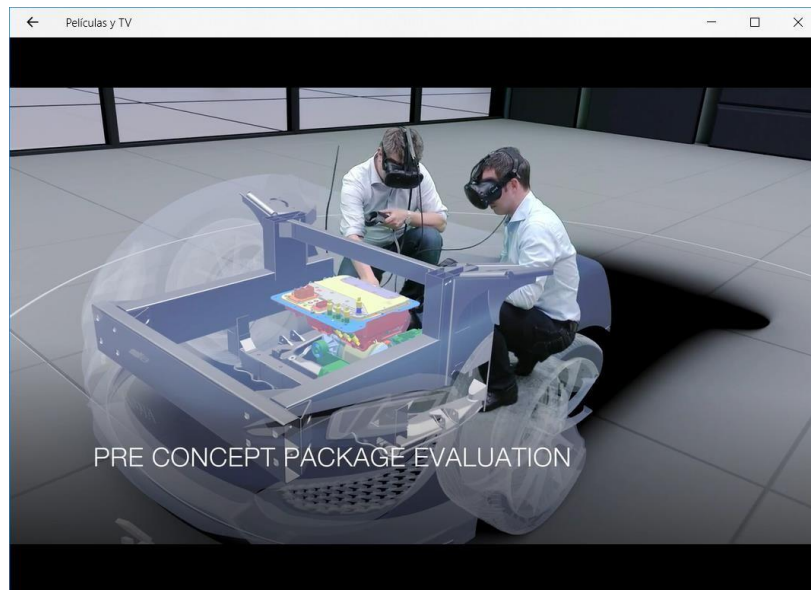


Figura I.12 Empleo de realidad virtual como extensión a los laboratorios virtuales de ingeniería

Finalmente, la implementación de Laboratorios Virtuales de Ingeniería (LVI) permite a las escuelas poner a disposición a más alumnos a menor costo la práctica y experiencia del uso de prácticas diseñadas con casos prácticos en ambientes virtuales con el objetivo de facilitar el aprendizaje y la enseñanza de los conceptos expuestos de manera teórica, donde los casos prácticos son escenarios virtuales más cercanos a la realidad, con mayor grado de complejidad, a un costo y tiempo menores de desarrollo.

Hemerografías digitales para consulta que complementan al Capítulo I

<http://www.mexicox.gob.mx/about>

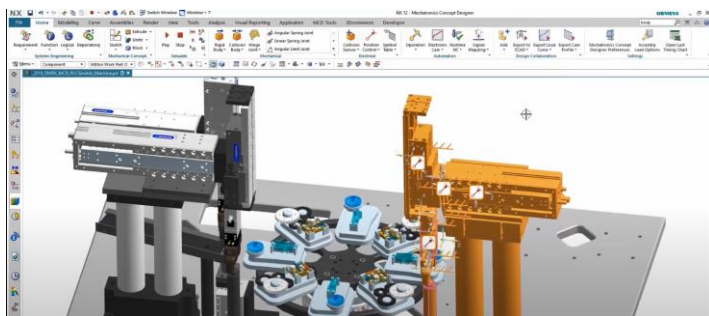
https://www.ecured.cu/Laboratorio_virtual

http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1405-66662014000300013

<http://www.revistagentegroo.com/la-triple-helice-modelo-de-desarrollo-para-mexico/>

Industria 4.0

<http://i40.semantic-interoperability.org/index.html#portfolio>



Capítulo II

COMISIONAMIENTO VIRTUAL CON NX MCD

Al denominado hilo digital de información conecta a personas, procesos, sistemas y máquinas, siendo una parte fundamental de la digitalización propia de la industria 4.0.

Las fábricas inteligentes con sistemas de fabricación automatizados y flexibles están controladas por programas que se ejecutan en los controladores lógicos programables (PLC), computadoras industriales, control distribuido. Tanto en los procesos de diseño, precomisionamiento, comisionamiento y puesta en marcha; la mejora continua puede realizarse en un entorno virtual antes de integrarlos en equipos reales. Al simular y validar un sistema automatizado, es posible confirmar que se encuentran operando óptimamente y por lo tanto reducir significativamente el tiempo de arranque y evitar en lo posible paros de línea, así como para ser utilizados en un modelo de formación dual.

Simulaciones en entornos virtuales permiten:

-Detecciones oportunas de colisiones entre partes móviles

-Simulaciones de errores en los comisionamientos

-Entrenamientos en ambientes seguros y verificación del funcionamiento completo del sistema

II.1 Introducción

El trabajo bien planificado de distintas disciplinas ingenieriles da por resultado el poder diseñar y poner en marcha un sistema automatizado, con documentación basada en normas que pueda ser útil para la mejora continua, ser consultado por cualquier ingeniero y capacitar también; además de contactar con los softwares especializados que permitan la gestión de vida del diseño de un producto, maquinaria, proceso o planta productiva, así como toda la información de ingenierías útiles como son: diseños mecánicos y sus validaciones distintas en entornos de trabajo simulados, diseños eléctricos, programas de control lógicos y/o cableados, conectividad según protocolos de comunicación, etc. Las etapas de trabajo de diseño que actualmente se conocen con las empresas denominadas Integradoras de Sistemas Automáticos son:

1. *Precomisionamiento*: ajustes de rangos de funcionamiento y sus pruebas de funcionamiento de toda la instrumentación, así como de los actuadores principalmente.
2. *Comisionamiento*: ajustes del conjunto de sensores y actuadores ya instalados en el sistema mediante un programa lógico de prueba de funcionamiento con la máquina.
3. *Puesta en marcha*: pruebas de la secuencia automática completa más las requeridas por norma: arranque, paros de emergencia, pruebas de funcionamiento de sub sistemas de la maquinaria, estaciones de trabajo o líneas de producción, para poder arrancar de manera segura, gradual y controlada.

El concepto central del *Comisionamiento*, permite planear y ejecutar de manera estructurada, eficaz y documentada el conjunto de acciones que se requieren para lograr un arranque efectivo, a fin de acelerar su entrada en producción con eficiencia y con seguridad, para la integridad de:

- Personas,
- equipos,
- y de la producción.

Por lo que plataformas informáticas denominadas PLM (*Product Life Management*), son de suma importancia en un mundo altamente competitivo, donde los productos y servicios tienden a la personalización, por lo que los procesos de producción deben ser modificables, configurables con mayor rapidez, así que los 3 pasos mencionados anteriormente ya es posible validarlos e interactuar con herramientas integradas de manera virtual, en ambientes seguros tanto para simular y co-simular; sin contar con el sistema físico electromecánico ni el control.

Mechatronics Concept Designer (MCD) es una aplicación de NX, es un paquete de software CAD/CAM/CAE desarrollado por la compañía Siemens PLM Software, dentro del mismo se tiene la herramienta de diseño mecatrónico, la cual simula de forma interactiva el control de movimiento complejo automatizado definido por el mecanismo, el software permite la integración del modelo 3D correctamente ensamblado con la definición de

componentes y/o características de funcionamiento de partes mecánicas, eléctricas, electrónicas, sensores.

Es una solución que transforma el proceso de creación de máquinas con un enfoque de diseño holístico y eficiente mecatrónico en un ambiente totalmente virtual y seguro, lográndose:

- Diseñar y simular interactivamente el movimiento complejo de los sistemas electromecánicos.
- En *NX MCD*, es posible definir comportamientos de todos los componentes físicos (sensores y actuadores) del diseño electromecánico en un solo archivo.
- Capacidad de simulación basada en fenómenos físicos.
- Validación temprana del sistema,
- Generación de históricos, gráficos.
- Colaboración multidisciplinaria (mecánicos, eléctricos, electrónicos, automatizadores y diseñadores industriales).
- Permite la creación de librerías de subcomponentes o subsistemas, para poder reutilizados en diferentes diseños.
- Herramientas de análisis y/o definición como son los perfiles de movimiento de sistemas totalmente mecánicos, electromecánicos o servomotores.
- Herramientas de definición de operaciones que apoyan en la simulación de movimiento del sistema electromecánico.

NX MCD proporciona un entorno de diseño colaborativo interactuando los siguientes roles, mencionados en el párrafo anterior:

- Los ingenieros de sistemas pueden administrar los requisitos y facilitar la comunicación interdisciplinaria.
- Los ingenieros mecánicos pueden crear el diseño basado en formas y cinemáticas en 3D.
- Los ingenieros eléctricos pueden seleccionar y posicionar sensores y actuadores.
- Los programadores de automatización pueden diseñar el comportamiento lógico básico de la máquina comenzando con un comportamiento basado en el tiempo y luego desarrollándolo en un control basado en eventos.

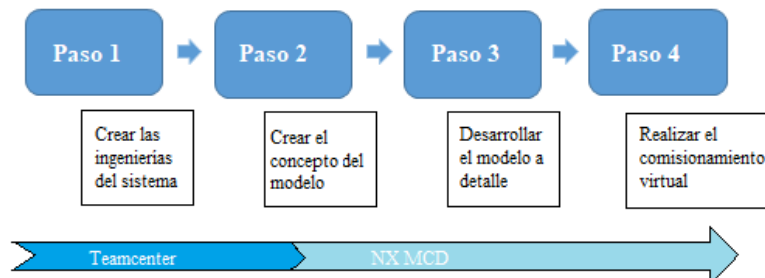


Figura II. 1 Pasos básicos del flujo de trabajo multidisciplinario en *NX MCD*.

II. 2 Flujo de trabajo de NX MCD

A continuación se muestra un flujo de trabajo de diseño de máquina típico en *NX MCD*:

- Definir y gestionar los requisitos de diseño.
- Crear un modelo funcional.
- Crear un modelo lógico.
- Definir el modelo básico del sistema.
- Crear enlaces de trazabilidad entre el modelo funcional y el modelo lógico.
- Definir el concepto mecánico.
- Agregar cinemática y dinámica.
- Añadir física básica y señales.
- Agregar restricciones de velocidad y actuadores con restricción de posición.
- Agregar adaptadores de señal.
- Asignar objeto adaptador de señal a función y árbol lógico.
- Definir operaciones basadas en el tiempo.
- Definir cómo los actuadores son controlados por las operaciones.
- Organizar la secuencia de operación con una noción basada en el tiempo.
- Asignar operaciones a las funciones y lógicas correspondientes.
- Añadir sensores.
- Añadir objetos sensores activados por colisiones de elementos del sistema o sensores definidos por un adaptador de señal.
- Definir operaciones basadas en eventos.
- Reemplazar modelos conceptuales con modelos detallados.
- Transferir objetos físicos con geometrías aproximadas a detalladas.

La figura II.2 es la ventana principal al ejecutar la aplicación.

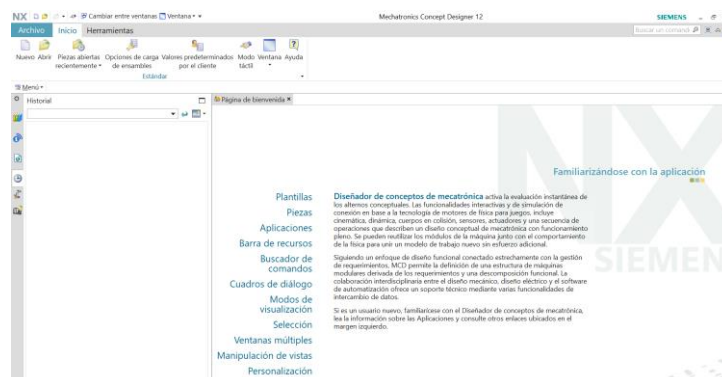


Figura II. 2 Ventana principal de NX MCD ver 12.0.

II. 3 Menú e iconos de la herramienta NX MCD ver 12.0

En la tabla II.1 se explica cómo se clasifican las herramientas y su función dentro de NX MCD, los números son los que se indican en la figura II.2.



Figura II. 2 Menú e iconos.

1	En la parte de Ingeniería de sistemas, es posible importar o exportar modelos funcionales como los diagramas eléctricos.
2	En la parte de Concepto de mecánica, se encuentran todas las herramientas para editar o corregir diseños 3D y del ensamble dentro del ambiente de NX MCD.
3	En la parte de Simular, se interactúa con el sistema cuando toda la física se encuentra definida aún sin comunicarlo con la parte de control proveniente del programa lógico del PLC, variadores de velocidad y/o servomotores. También con este menú se interactúa para validar el sistema electromecánico con el control, a lo que denominamos: co-simulación con los softwares TIA Portal, PLCSim Advanced, SIMIT, como por ejemplo.
4	En la parte de Mecánica, se tienen todas las herramientas para definir el comportamiento de las diferentes articulaciones, dar de alta los cuerpos rígidos, cuáles de éstos se colisionan, etc.
5	En la parte de Automatización, se tienen las herramientas para definir las señales de Entradas/Salidas de intercambio entre NX MCD y el programa de la lógica de control del PLC Virtual empleado: ET200 y/o S7 1500, acoplamiento de las señales si es necesario y los protocolos de comunicación entre NX MCD con el sistema de control automático.
6	En la parte de Diseño en colaboración, exportamos las curvas de movimiento logradas al optimizar, perfiles de comportamientos de las levas, etc.

Tabla II. 1 Clasificación de herramientas en NX MCD.

II. 4 Navegador de Física

Aquí se encuentran las carpetas donde se irán guardando todas las características físicas y de funcionamiento del sistema electromecánico.

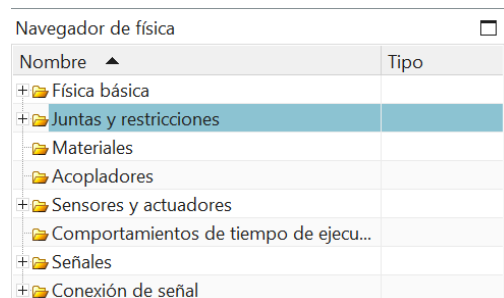


Figura II. 3 Árbol de trabajo, navegador de física.

En la carpeta de Física básica se encuentran todos los componentes del modelo 3D que se moverán entre sí, por lo que hay que declararlos como cuerpos rígidos, con nombre relacionado a su función en el mecanismo, cuerpos que serán afectados por lo tanto por la variable de gravedad definida para la simulación; en la figura II.4 se enumeran las herramientas para la definición de las características mecánicas que se clasificarán en el navegador de física, la del de cuerpo rígido es el número 1.

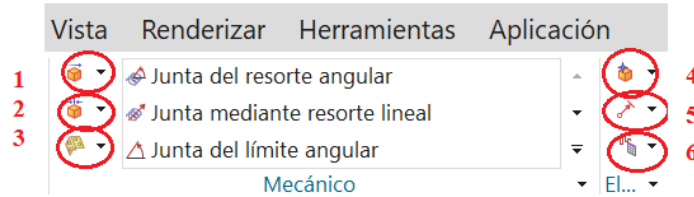


Figura II. 4 Menú para definiciones de comportamientos Mecánicos.

II. 4.1 Mecánico

En cada uno de los iconos enumerados y señalados en la figura II.4 muestran listas desplegables, que se describirán de manera breve en el apartado II.4.1 Mecánico y que se irán dando de alta en cada carpeta del navegador de física.

1. Cuerpo rígido.
2. Cuerpos de colisión.
3. Juntas.
4. Sensores
5. Accionamientos.
6. Señales.

Donde encontraremos al desplegar la lista para el cuerpo rígido (1), el siguiente menú:







1		Cuerpo rígido	<i>1.-Define un cuerpo rígido para la simulación física</i>
2		Fuente de objetos	<i>2.-Crea una fuente de objetos que copia los objetos, los cuales ingresan a la simulación de manera periódica, tiempo definido por el usuario</i>
3		Sumidero de objetos	<i>3.-Crea un sumidero de objetos que elimina las copias de los objetos físicos durante la simulación</i>
4		Transformador de objeto	<i>4.- Crea un transformador de objetos que transforma un cuerpo rígido en otro durante la simulación</i>
5		Entrada de transmisor	<i>5.-Define una entrada del transmisor para enviar los cuerpos detectores al puerto</i>
6		Salida de transmisor	<i>6.-Define una salida en el transmisor para recibir los cuerpos en el puerto</i>

Tabla II. 2 Descripción de funciones icono cuerpo rígido (1).

Una vez definidos los cuerpos rígidos del mecanismo se señalan si existe entre los cuerpos colisiones simples (2), las cuales dependen del tipo de material; aquí se dan de alta también los componentes de transporte, por ejemplo: bandas transportadoras. En la tabla II.3 se describe lo que contiene el menú del icono de cuerpos de colisión.






1		Cuerpo de colisión	<i>1.-Define un cuerpo de colisión</i>
2		Superficie de transporte	<i>2.-Define una superficie de transporte</i>
3		Evitar la colisión	<i>3.-Cambia las propiedades de la colisión entre un par de cuerpos específicos</i>
4		Cambiar el material	<i>4.-Cambia las propiedades de fricción del material</i>
5		Material de colisión	<i>5.-Define un material de colisión</i>

Tabla II. 3 Descripción de funciones icono cuerpos de colisión (2).

Las juntas y sus restricciones (3), aquí se guardan todos los movimientos específicos entre los cuerpos definidos como rígidos, en la tabla II.4 se detallan sus funcionalidades.












1		Junta de charnela	1.-Conecta los objetos a lo largo de un eje de rotación
2		Junta deslizante	2.-Conecta los objetos a lo largo de un eje lineal fijo
3		Junta cilíndrica	3.- Conecta los objetos a lo largo de un eje lineal rotatorio
4		Junta para tornillo	4.-Conecta los objetos a lo largo de un eje lineal rotatorio con un valor fijo del paso de rosca del tornillo
5		Junta planaria	5.-Conecta los objetos para que puedan desplazarse libremente y rotar de forma relativa entre sí, mientras permanecen en contacto entre superficies
6		Junta del eje virtual...	6.-Crea una junta de eje virtual que contiene la información de la cinemática sin la geometría
7		Junta esférica	7.-Conecta los objetos en un punto compartido
8		Junta fija	8.-Conecta los objetos para que no se puedan mover de forma relativa de uno al otro
9		Junta de punto sobre curva	9.-Conecta un objeto que se mueve a lo largo de una curva
10		Junta de curva sobre curva	10.-Mueve un objeto curvo a lo largo de otra curva
11		Junta de restricción en la trayectoria	11.- Conecta un objeto que se mueve a lo largo de una curva y restringe la orientación con una secuencia de sistema de coordenadas

Tabla II. 4 Descripción de funciones icono juntas (3).

Los sensores de colisión (4), en la tabla II.5 se detallan brevemente sus funcionalidades.











1	 Sensor de colisión	<i>1.-Define un sensor de colisión para la simulación</i>
2	 Sensor de distancia	<i>2.-Define un sensor de distancia</i>
3	 Sensor de posición	<i>3.- Define un sensor de posición</i>
4	 Inclinómetro	<i>4.-Define un inclinómetro</i>
5	 Sensor de velocidad	<i>5.-Define un sensor de velocidad</i>
6	 Acelerómetro	<i>6.-Define un acelerómetro</i>
7	 Sensor genérico	<i>7.-Sensor genérico para detectar colisiones</i>
8	 Interruptor limitador	<i>8.-Define sensores de contacto tipo interruptores de límite(limit switch)</i>
9	 Relé	<i>9.-Da de alta características para relevadores</i>
10	 Exportar los sensores y actuadores en SIMIT	<i>10.-Exporta las características de los sensores y actuadores definidos en el mecanismo para co-simular con variadores de velocidad y servodrives (elementos de potencia para servomotores)</i>

Tabla II. 5 Descripción de funciones icono sensores de colisión (4).

Las funciones denominadas accionadores (5), en la tabla II.6 se detallan las opciones con las que cuenta.












1		Control de posición	1.-Mueve los objetos a lo largo de un eje hasta una posición deseada
2		Control de velocidad	2.- Mueve los objetos continuamente a lo largo de un eje a una velocidad deseada
3		Control de fuerza/par de torsión	3.- Aplica una fuerza de torsión a una junta de un eje
4		Cilindro hidráulico	4.-Define un cilindro hidráulico con características específicas como variables de estado y parámetros fijos dimensionales
5		Válvula hidráulica	5.- Define un cilindro neumático con características específicas como variables de estado y parámetros fijos dimensionales
6		Cilindro neumático	6.- Define una válvula neumática con características específicas como cantidad de vías y presiones
7		Válvula neumática	7.-Define una superficie de transporte lineal o rotatoria, con parámetros del tipo de movimiento (distancias)
8		Superficie de transporte	8.-Exporta la información sobre el control del eje
9		Exportar la curva de carga	9.-Importa los datos de las características de los motores
10		Importar los motores seleccionados	10.-Exporta las funcionalidades de actuadores y sensores al software SIMIT
11		Exportar los sensores y actuadores en SIMIT	

Tabla II. 6 Descripción de funciones icono accionadores (5).

En el menú desplegable de señales (6), en la tabla II.7 se detallan de manera breve las funciones.







1		Tabla de símbolos	1.-Define una tabla de símbolos para el nombre de las señales a utilizar para Entradas/Salidas, las cuales serán los mismos símbolos para el programa lógico del PLC
2		Señal	2.- Define una señal de MCD de una conexión externa
3		Adaptador de señales	3.- Define un adaptador de señal que se comparten en la co-simulación MCD y control automático, las cuales tienen distintos formatos de los datos (señales); en el caso de OPC el adaptador se define como lo requiere dicho protocolo
4		Crear las señales de las operaciones	4.-Crea las señales basadas en los parámetros de tiempo de ejecución referenciadas por las operaciones
5		Exportar las señales	5.- Exporta las señales
6		Importar las señales	6.- Importa las señales de un archivo externo

Tabla II. 7 Descripción de funciones icono señales (6).

A continuación se puntualizan de manera general las funciones de las propiedades mecánicas de los componentes o piezas que conforman el ensamble que presentarán algún movimiento con topologías de mecanismos (configuración geométrica), en la figura II.5 se observan los iconos para definir restricciones para juntas (1), iconos que permiten dar de alta los distintos tipos de acoplamientos mecánicos (2) y los que configuran comportamientos personalizados (3).

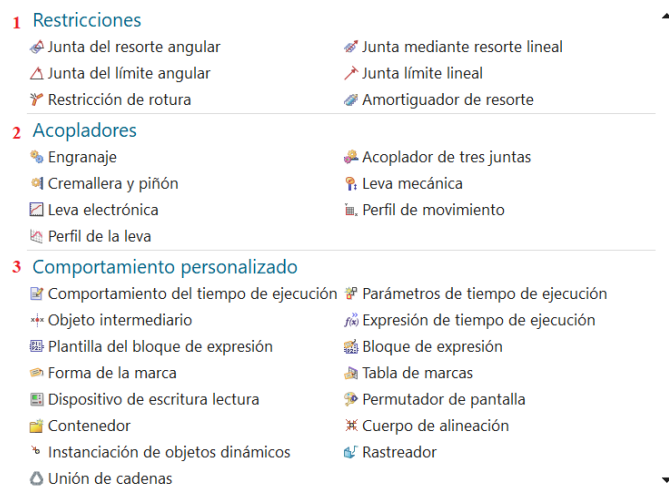


Figura II.5 Iconos para definir restricciones para juntas, acopladores Mecánicos.

En la sección de restricciones de juntas de la figura II.5, en la tabla II.8 se detallan las funciones.

- 1 Junta del resorte angular
- 2 Junta del límite angular
- 3 Restricción de rotura
- 4 Junta mediante resorte lineal
- 5 Junta límite lineal
- 6 Amortiguador de resorte

- 1.-Aplica una fuerza de tipo resorte entre los objetos, la fuerza se incrementa a medida que los objetos rotan en forma relativa el uno al otro
- 2.- Detiene el movimiento de los objetos cuando la posición relativa se mueve más allá de un grado determinado o se acercan demasiado
- 3.-Especifica que se puede romper una junta si la fuerza aplicada a la misma excede un valor determinado
- 4.- Aplica una fuerza lineal de tipo resorte entre los objetos, la fuerza se incrementa a medida que los objetos se alejan o se acercan más
- 5.- Detiene el movimiento de los objetos cuando la posición relativa sobrepasa una distancia o se aproxima demasiado
- 6.- Crea un elemento flexible para ejercer fuerza o par de torsión dentro de una junta del eje

Tabla II. 8 Descripción restricciones de juntas figura II.5 (1).

En la sección de restricciones de acopladores de la figura II.5, en la tabla II.9 se detallan de manera breve las opciones.








1	 Engranaje	1.-Conecta el movimiento de dos ejes para que se muevan a una razón fija
2	 Cremallera y piñón	2.-Conecta el movimiento de ejes lineal y rotatorio para que se muevan a una razón fija
3	 Leva electrónica	3.-Aplica los valores de la leva electrónica al control del eje con el tiempo
4	 Perfil de la leva	4.-Crea un perfil de leva utilizado para restringir la leva
5	 Acoplador de tres juntas	5.-Conecta el movimiento de tres ejes para que se muevan a una razón fija
6	 Leva mecánica	6.-Conecta el movimiento de dos ejes para que se muevan según una función
7	 Perfil de movimiento	7.-Crea un perfil de movimiento usado para restricción de leva

Tabla II. 9 Descripción menú acopladores, figura II.5 (2).

En la sección de comportamientos personalizados de la figura II.5, en la tabla II.10 se detallan las funciones.
















1	 Comportamiento del tiempo de ejecución	1.-Agrega un objeto de código escrito por el usuario que puede interactuar con el comportamiento de la simulación
2	 Objeto intermediario	2.-Define un objeto intermediario correspondiente a la simulación de un cuerpo físico
3	 Plantilla del bloque de expresión	3.-Define un archivo de plantilla que se utilizar para instanciar una copia del bloque de ejecución
4	 Forma de la marca	4.-Define una forma de la marca para la simulación
5	 Dispositivo de escritura lectura	5.-Define un dispositivo escritura/ lectura para la simulación
6	 Contenedor	6.-Crea una carpeta contenedora en el navegador de física
7	 Instanciación de objetos dinámicos	7.-Agrega una tabla de objetos dinámicos para mover el cuerpo rígido configurado a una ubicación específica mediante el mapeo del cuerpo de alineación de definición y el cumplimiento del cuerpo de alineación
8	 Unión de cadenas	8.-Conecta los objetos con las juntas de charnela
9	 Parámetros de tiempo de ejecución	9.- Define un objeto para los parámetros de ejecución correspondientes a la simulación de un cuerpo físico
10	 Expresión de tiempo de ejecución	10.-Agrega una expresión del tiempo de ejecución que puede interactuar con la simulación
11	 Bloque de expresión	11.-Define un bloque de expresión basado en una plantilla
12	 Tabla de marcas	12.-Define una tabla de marcas para la simulación
13	 Permutador de pantalla	13.-Modifica las propiedades de visualización del cuerpo rígido
14	 Cuerpo de alineación	14.-Agrega un cuerpo de alineación para posicionar un cuerpo o hacer que dos cuerpo yaczan exactamente uno sobre el otro
15	 Rastreador	15.-Rastrea la trayectoria de un punto y la dibuja

Tabla II.10 Descripción comportamientos personalizados, figura II.5 (3).

II. 5 Barra de recursos

La barra de recursos es el conjunto de pestañas ubicadas a lo largo del borde de la ventana de *NX*, estas pestañas varían según su configuración específica y la aplicación activa. La barra de recursos se divide en:


- 
1. Navegantes
 2. Herramientas HD3D (Reportes visuales de modelos 3D interactivos)
 3. Ventana del navegador integrada (solo Windows)
 4. Paletas

Figura II. 6 Barra de recursos.

Esta barra incluye los siguientes iconos relevantes para *NX MCD* que permiten:







	<i>Navegador del sistema: éste sirve para ver los requisitos, los modelos funcionales y lógicos de un producto</i>
	<i>Navegador de física: éste se emplea para mostrar las propiedades físicas y lógicas de los elementos mecánicos</i>
	<i>Inspector de tiempo de ejecución: éste se utiliza para inspeccionar las propiedades de los objetos de tiempo de ejecución de física durante la simulación</i>
	<i>Expresión de tiempo de ejecución: éste se usa para ver las expresiones que creó para aplicar ecuaciones, proporciones y relaciones entre propiedades físicas</i>
	<i>Navegador de ensamblajes: éste se emplea para ver los componentes utilizados en el ensamblaje y acceder a los comandos de ensamblajes</i>
	<i>Editor de secuencias: en él puede crear operaciones basadas en tiempo y en eventos. Después de crear operaciones, use el Editor de secuencias para ver y crear una secuencia de operaciones, similar a la lógica de función secuencial</i>

Tabla II.11 Descripción iconos relevantes de la barra de recursos.

II. 6 Configuración de los parámetros para la simulación

Es necesario configurar en qué sistema de unidades se trabajará para los cálculos y mediciones, así como también el vector de la fuerza de gravedad, tiempos de ejecución, entre otros, para lo cual es necesario buscar en el menú de la barra de iconos de visualización del modelo 3D en el área de trabajo y no el principal la opción de preferencias, después elegir la opción de Diseñador de conceptos de mecatrónica.

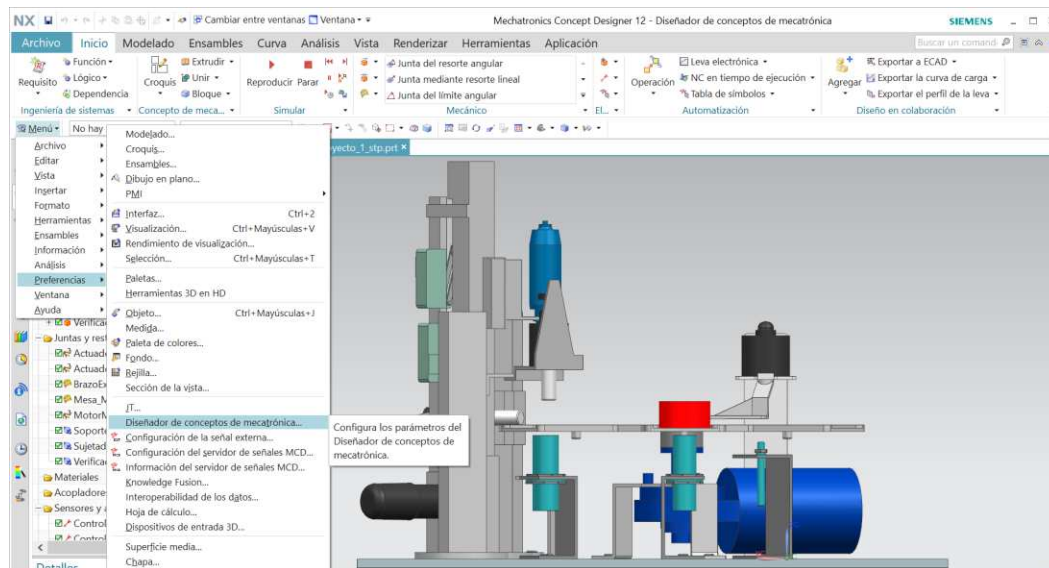


Figura II. 7 Opción para Configuración parámetros para simulación en NX MCD.

Para las dos opciones de co-simulación (simulación conjunta) se tiene: PLCSIM Advanced y SIMIT, en la figura II. 8 se aprecia que es elegible el maestro para NX MCD o SIMIT.

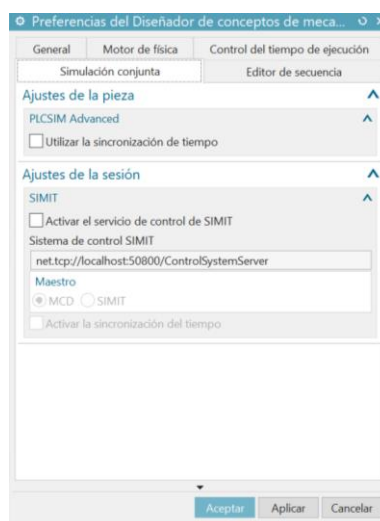


Figura II. 8 Pestaña simulación conjunta.

Para ajustar los parámetros y unidades del *solver* del software hay que dar clic en la pestaña de motor de física, en el caso de no comprender a precisión los conceptos de esta ventana, contamos con la opción de restaurar el valor predeterminado.

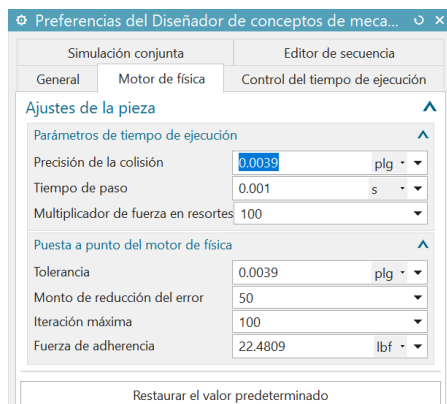


Figura II. 9 Pestaña motor de física.

En la pestaña general es donde se debe especificar las unidades de medición, así como otros parámetros para los fenómenos de fricción y el vector de gravedad.

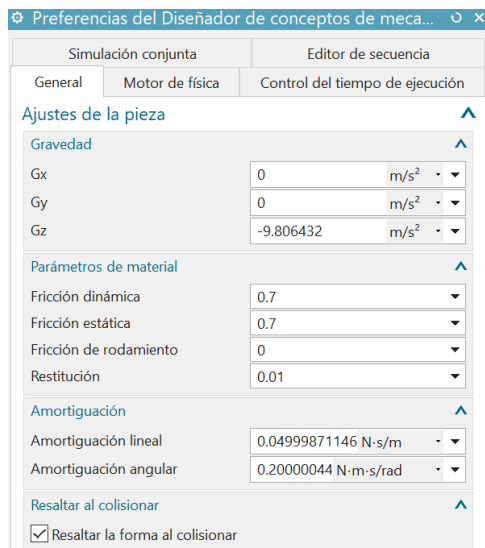


Figura II. 10 Pestaña General de la caja Preferencias del Diseñador mecatrónico.

Y finalmente definir los tiempos de ejecución para la simulación:

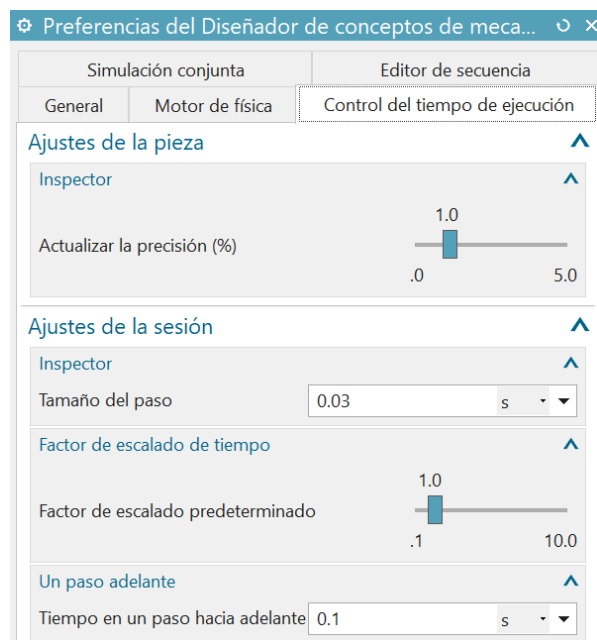
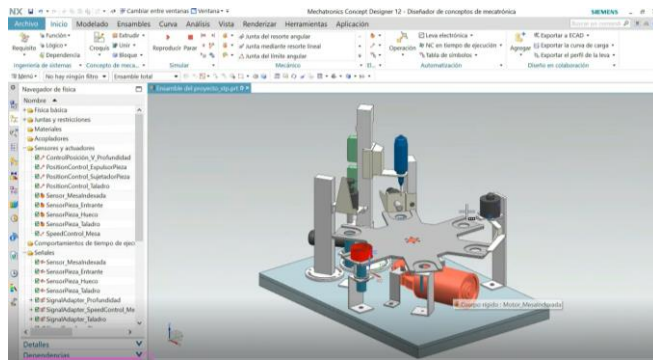


Figura II. 11 Pestaña Control de tiempos de ejecución para simulaciones.

Bibliografía Capítulo II

Este capítulo tiene como referencia el manual:

Leonardo Llumiquinga, CURSO MECHATRONICS CONCEPT DESIGNER Manual (2019). Ecuador: TechDesign



Capítulo III

CASO PRÁCTICO ESTACIÓN DE PROCESAMIENTO

La estación de trabajo denominada de procesamiento forma parte de una celda didáctica flexible representativa de Industria 3.0 de la marca Festo, donde las características de sus entradas y salidas son todas digitales, sensores inductivos y capacitivos de la misma marca que trabajan a 24 Vcd y sus actuadores son accionados con corriente directa a 24 Volts.

Se eligió como un ejemplo sencillo para introducir al comisionamiento virtual con NX MCD.

Comisionamiento en NX MCD define:

- Cuerpos rígidos
- Cuerpos en colisión
- Articulaciones
- Sensores
- Acopladores de
señales para las co-
simulaciones
- Bandas
transportadoras
- Suministro
automático de piezas
y por tiempos
definidos

III.1 Función de la estación de trabajo de procesamiento didáctica, Marca Festo

Primeramente se explica la función de la estación didáctica, *procesamiento* es un término genérico para los pasos de producción tales como formación, cambio de forma, mecanizado y unión. Los símbolos utilizados para las operaciones de manipulación están avalados según la norma VDI. 2860, la formación es la creación de cuerpos geoméricamente determinados, el cambio de forma es el cambio de formas geométricas y/o de las dimensiones de los cuerpos, el mecanizado es el cambio de características del material y/o acabado superficial, la unión es la asociación permanente de varios cuerpos, ejemplo: procesos con soldadura.

Las funciones en general de la estación de procesamiento son:

- Para comprobar las características de las piezas de trabajo (posicionamiento correcto, profundidad de la abertura en la pieza con forma circular),
- para mecanizar piezas y
- para suministrar piezas a una estación posterior.



Figura III.1 Estación didáctica de Procesamiento, marca Festo.

La estación de procesamiento está conformada de las siguientes partes:

1. Módulo de mesa rotativa indexada.
2. Módulo de prueba de profundidad de abertura de pieza.
3. Módulo de perforación.
4. Módulo de sujeción.
5. Módulo de puerta de clasificación, ver figura III.2.

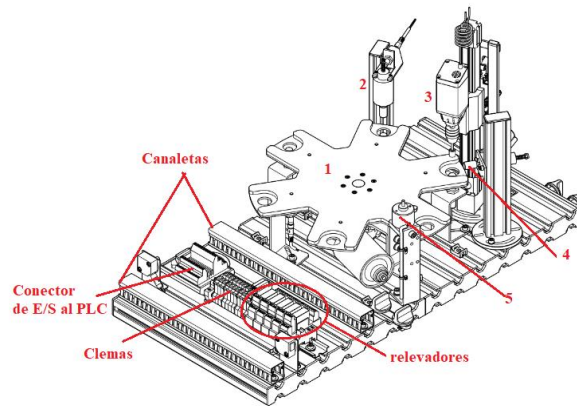


Figura III.2 Partes de la estación de Procesamiento.

Componentes para el montaje del mecanismo y del control:

1. Placa de perfil
2. Gabinete con desplazamiento
3. Control de consola, ver figura III.3

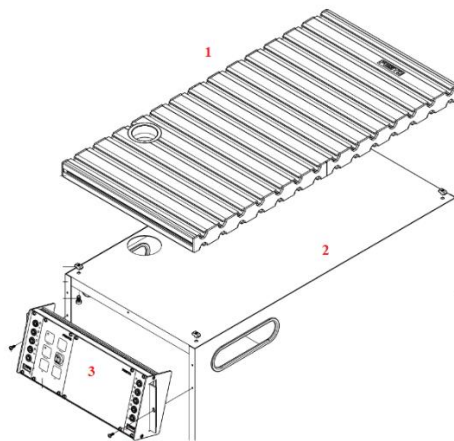


Figura III.3 Componentes para el montaje.

Su función se describe a continuación con mayor detalle:

Las piezas de trabajo se prueban y procesan en un indexador giratorio (mesa), la mesa giratoria de indexación es accionada por un motor de corriente continua, con un reductor y con acoplamiento mecánico. La mesa para mediante un circuito eléctrico con relevadores para desactivar la acción del motor, cuando la posición de la mesa es detectada por un sensor inductivo. En la mesa de indexación giratoria, las piezas de trabajo se prueban y taladran en dos procesos paralelos, un actuador magnético con sensor inductivo comprueba que las

piezas se inserten en la posición correcta. Durante el taladrado, la pieza de trabajo es sujeta por un actuador accionado por una señal digital (1 lógico: 24 Vcd y 0 lógico: 0 Vcd). Las piezas acabadas se transmiten a través del eyector eléctrico, accionado también por señal digital a 24 Vcd.

Es importante indicar que todos los sensores inductivos y capacitivos trabajan a 24 Vcd y los actuadores también.

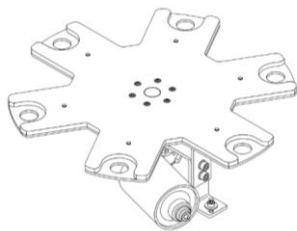
- Posición inicial (pieza dentro de la estación).
- Mesa rotativa indexada posicionada.
- Comprobación del émbolo del solenoide levantado (comprobación abertura hacia arriba, pieza colocada correctamente).
- Taladro en posición elevada, mediante el actuador lineal.
- El motor de la máquina del taladro debe estar apagado.
- Dispositivo de sujeción retraído.
- Mando de arranque no accionado.

La secuencia se describe en 5 pasos básicamente:

1. La mesa de indexación gira 60° y para, si se detecta una pieza de trabajo en el retenedor de la pieza de trabajo mediante un sensor capacitivo y se presiona el botón de arranque.
2. El émbolo del solenoide se mueve hacia abajo y verifica si la pieza de trabajo está insertada con la abertura hacia arriba. La mesa de indexación es rotada nuevamente 60° y para, si el resultado de la comprobación es OK.
3. El dispositivo de sujeción sujeta la pieza de trabajo. El motor de la perforadora es encendido. El eje lineal mueve la máquina perforadora hacia abajo.
4. Cuando la máquina perforadora ha alcanzado su posición inferior, se mueve a su posición superior hasta detenerse de nuevo por el eje lineal.
5. El motor de la máquina perforadora se apaga y el dispositivo de sujeción es retraído. La mesa giratoria de indexación gira nuevamente 60°.
6. La parte eléctrica de expulsión pieza, pasa la pieza a una estación posterior.

III.1.1 Descripción componentes principales

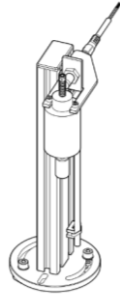
Módulo de mesa Rotatoria indexada



El accionamiento del módulo de la mesa divisora rotativa es operado por un motor reductor de corriente continua. Las 6 posiciones de la placa giratoria están definidas por los tornillos de posicionamiento en la mesa giratoria, los cuales son detectados por medio de un sensor inductivo.

Cada uno de los 6 retenedores de piezas de trabajo semicirculares de la placa tienen un orificio en el centro para facilitar la detección de pieza por medio de un sensor de proximidad capacitivo.

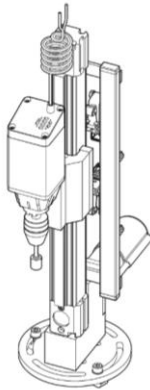
Módulo de prueba



Se comprueba la posición correcta de una pieza de trabajo insertada. Si el agujero apunta hacia arriba, entonces, la armadura del solenoide de prueba alcanza su posición final.

Un sensor de proximidad inductivo se activa a través de una tuerca en el extremo superior de la armadura, *el cual en la comisión virtual no se representó, para la puesta en marcha virtual es necesario agregarlo.*

Módulo de taladrado

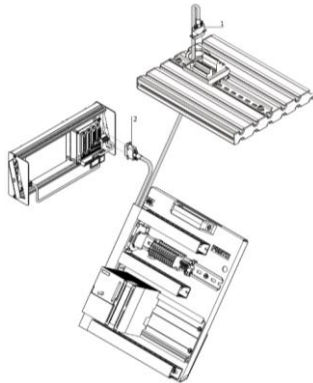


El módulo con el Taladrado se utiliza para simular el pulido del orificio de la pieza de trabajo.

Un dispositivo de sujeción eléctrico retiene la pieza de trabajo, accionado con 24Vcd. Las acciones de alimentación y retorno del taladro se realizan mediante un eje lineal con transmisión por correa dentada. Un motor reductor eléctrico acciona el eje lineal y un circuito de relé se utiliza para activar el motor.

El motor de la máquina perforadora funciona con 24Vcd y la velocidad no es ajustable. La detección de la posición final se realiza mediante finales de carrera eléctricos. La aproximación de los finales de carrera provoca una inversión del sentido de movimiento del eje lineal. *En el comisionamiento virtual se simulará de manera conjunta con el PLC, mandando el valor del desplazamiento adecuado, validación solo virtual.*

Conexiones de placa



En la imagen se tiene el conector número 1 que lleva las señales de entradas (sensores) de la estación al PLC y las señales de salidas (actuadores) del PLC a la estación. El cable de conexión con número 2, es el que manda las señales de mando al PLC.

Tabla III. 1 Descripción componentes principales.

III.2 Precomisionamiento virtual

Para realizar el *precomisionamiento virtual* se trabajó con el diseño a escala del modelo 3D del ensamble, el cual fue realizado con el software Solid Edge de la empresa Siemens PLM. Ya que la formación de recurso humano en los proyectos académicos es importante, el alumno participante se certificó en dos niveles en Solid Edge asociado nivel I y nivel II; posteriormente realizó el Gemelo Funcional de la estación de procesamiento con escala 1:1. Ya que el objetivo principal es la aplicación del *comisionamiento en simulación conjunta* del sistema electromecánico y su control, en un entorno virtual. Ver figura III.4.

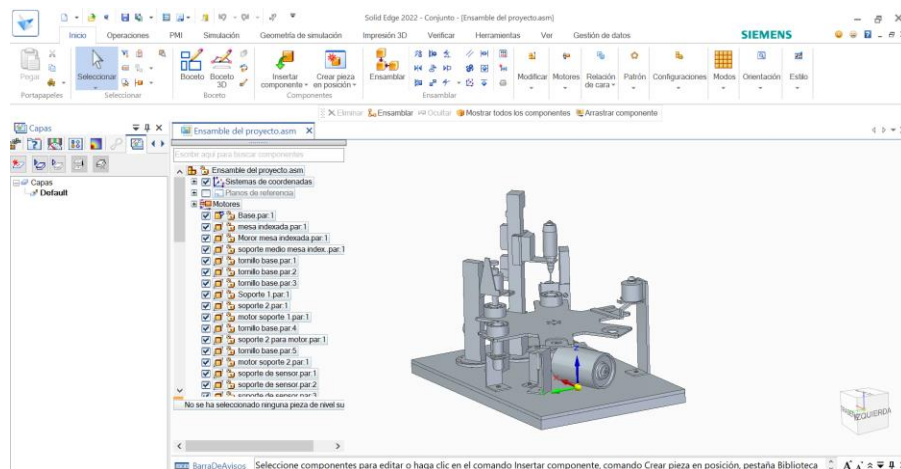


Figura III.4 Gemelo funcional: estación de procesamiento en Solid Edge.

El archivo del proyecto del ensamble debe ser guardado con la extensión: .stp, para poder después abrirse en *NX MCD*, es posible realizar el mismo procedimiento no importando el software CAD utilizado. Al abrirlo en *NX MCD* se ve la siguiente figura:

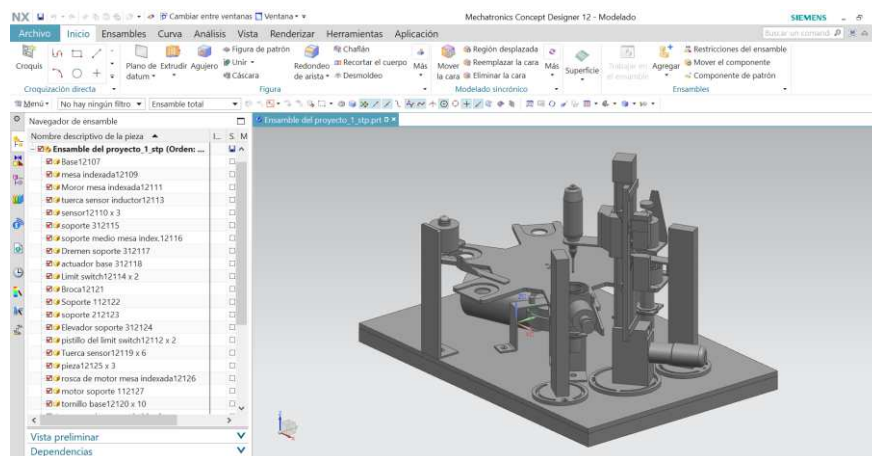


Figura III.5 Gemelo funcional estación de procesamiento en NX.

En la misma área de trabajo podemos cambiar los colores de las piezas, con solo dar clic en la pieza, luego botón derecho y elegir el icono de editar visualización, como se puede ver en la figura III.6

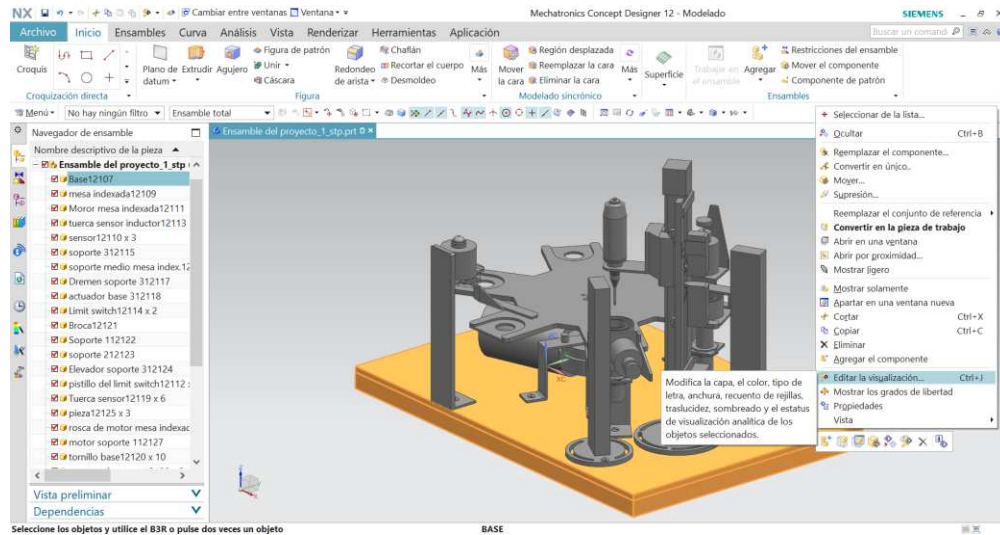


Figura III.6 Edición de visualización por piezas.

Se abre un cuadro de diálogo, donde se da clic en la opción de colores para tener una amplia posibilidad de éstos para elegir, como se puede observar en la siguiente figura donde se encuentra otra pieza seleccionada (sensor pieza posición taladro).

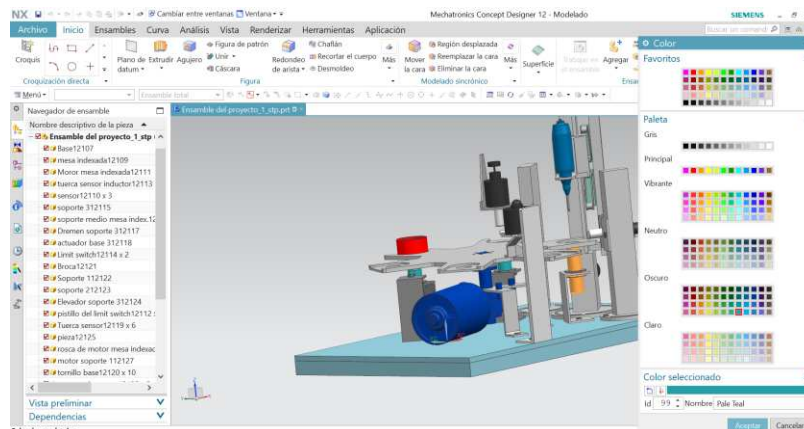


Figura III.7 Paleta de colores.

En la siguiente figura se muestra el sistema completo con colores.

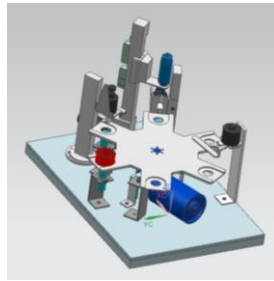


Figura III. 8 Visualización del Gemelo Digital Funcional escala 1:1.

Para comenzar a trabajar en *NX MCD*, es necesario ir al menú y elegir: *archivo* → *preferencias* → *todas las aplicaciones* → y finalmente dar clic en *diseñador de conceptos de mecatrónica*, ver figura III.9.

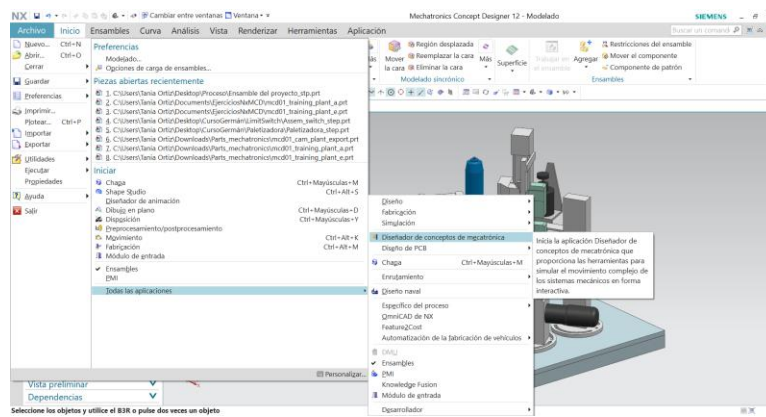


Figura III.9 Selección de la aplicación *NX MCD*.

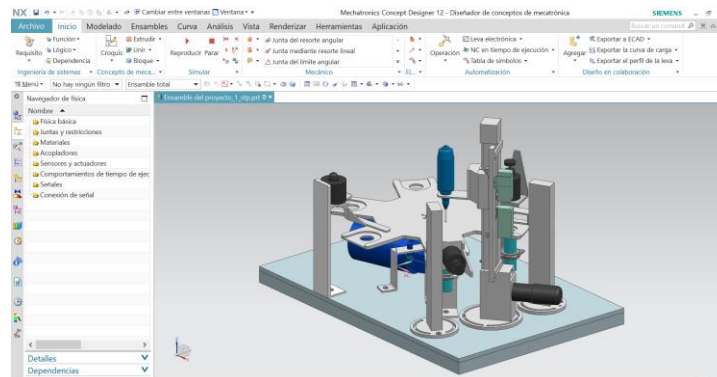


Figura III.10 Gemelo Digital en el entorno de *NX MCD*.

Se sugiere primeramente dar de alta a los cuatro sensores del gemelo, por lo que es necesario ocultar la mesa indexada, para ello hay que seleccionarla, después elegir el menú de desplegar y ocultar → dar clic en ocultar. Ver figura III.11.

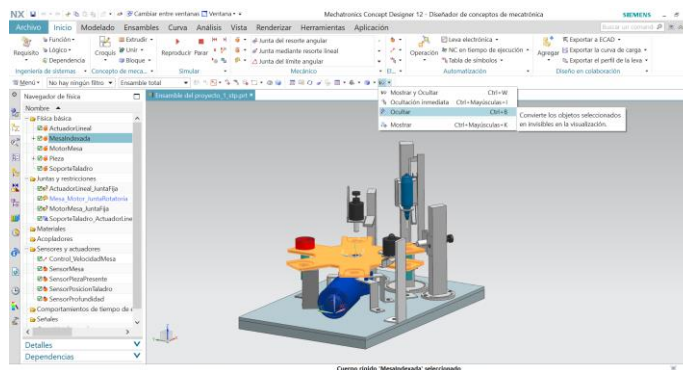


Figura III.11 Procedimiento para ocultar mesa indexada.

Una vez oculta la mesa se explica el cómo dar de alta los sensores de proximidad (3 capacitivos) y un sensor inductivo, éste último es el que detecta la posición de la mesa cada (60°). Dar clic en el menú desplegable de sensores y elegir el de colisión, ver la siguiente figura.

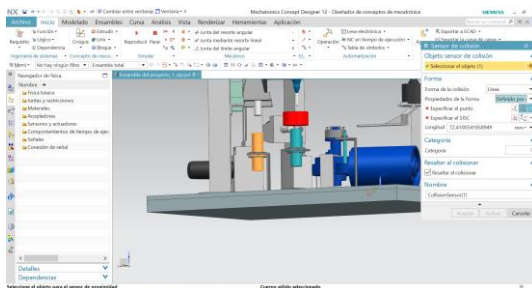


Figura III.12 Sensor de colisión.

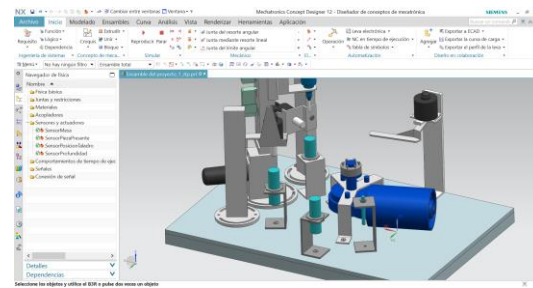


Figura III.13 Vista Gemelo sin mesa indexada.

En la caja de diálogo del sensor de tiene:

- Objeto sensor de colisión: posicionarse con el mouse en esta opción y después dar clic en el sensor en el gemelo digital.
- Forma: elegir la forma de colisión lineal.
- Propiedades de la forma: elegir definida por el usuario para poder indicar el punto de origen y el sistema de coordenadas.
 - Punto de origen: acercarse al sensor para verlo por arriba, elegir la circunferencia para poder visualizar el punto de origen y seleccionarlo. Ver figura III.14.

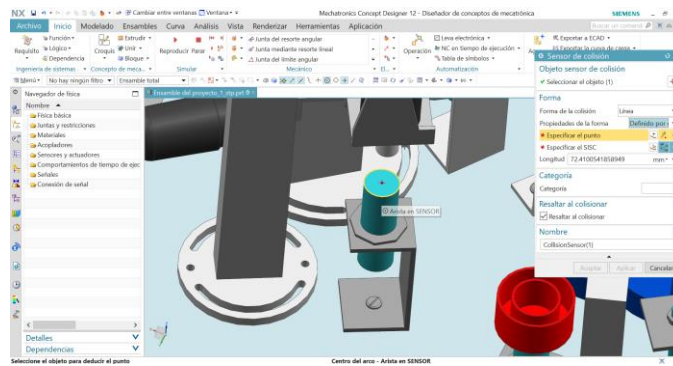


Figura III.14 Especificación de punto de origen sensor de colisión.

- Especifica el SISC: al dar clic en esta opción hay que colocar el modelo 3D en la mejor vista para poder encontrar el SISC (sistema de coordenadas) y poder orientar correctamente la línea de vista del sensor. Ver figura III.15, hay que girar el SISC con solo dándole clic en el punto central del semicírculo del plano del SISC, y colocar la longitud en mm deseada para la línea de vista.

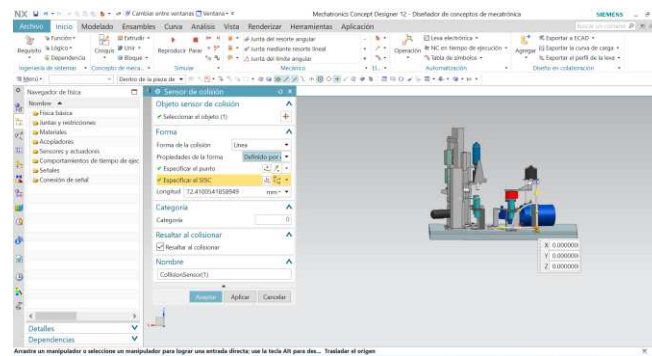


Figura III.15 Definición del SISC de la línea de vista del sensor.

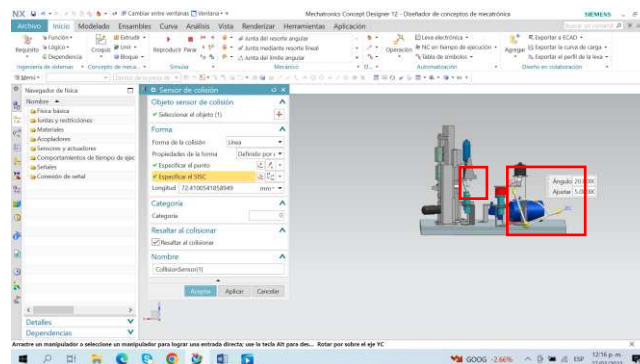


Figura III.16 Giro del SISC para colocar la línea de vista del sensor.

- Categoría: dejar 0.

- Resaltar al colisionar: elegir la opción de resaltar.
- Nombre: escribir el nombre del sensor según su función.

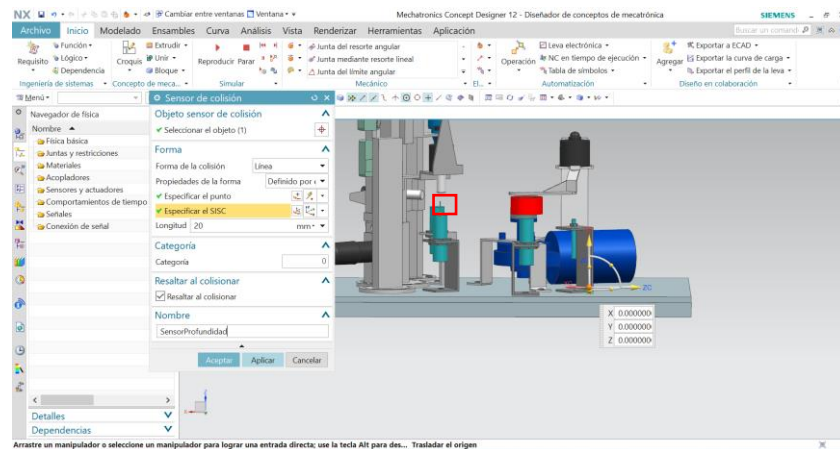


Figura III.17 Sensor de colisión configurado correctamente.

La siguiente configuración del navegador de física será dar de alta los cuerpos que se mueven entre sí, por lo que hay que configurarlos como cuerpos rígidos, los tres primeros que se darán de alta será la mesa, pieza y el motor de la mesa indexada. El cómo darlos de alta: se debe elegir primeramente el menú desplegable de cuerpos rígidos en la categoría de mecánica, se abre la caja de diálogo correspondiente y se selecciona el componente del gemelo para asignarlo como cuerpo rígido, en la misma caja se deja el cálculo de la masa e inercias de manera automática, en la marca no se escribe nada por el momento y finalmente se escribe el nombre del cuerpo rígido, en la figura siguiente es el de pieza de trabajo.

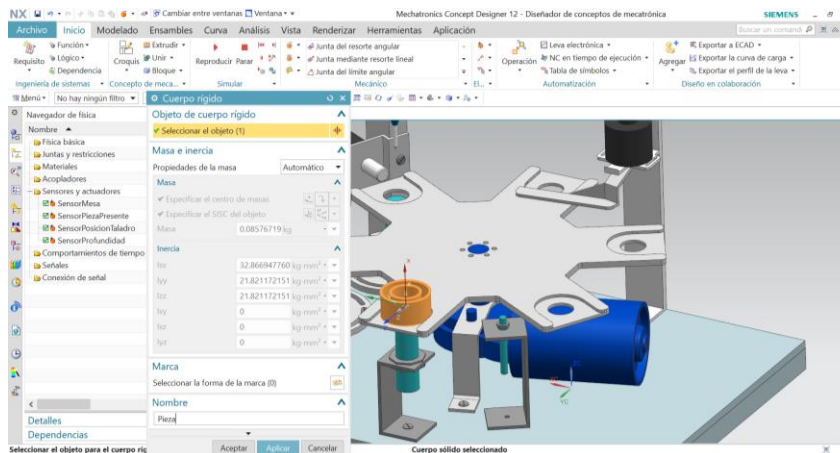


Figura III.18 Cuerpo rígido Pieza.

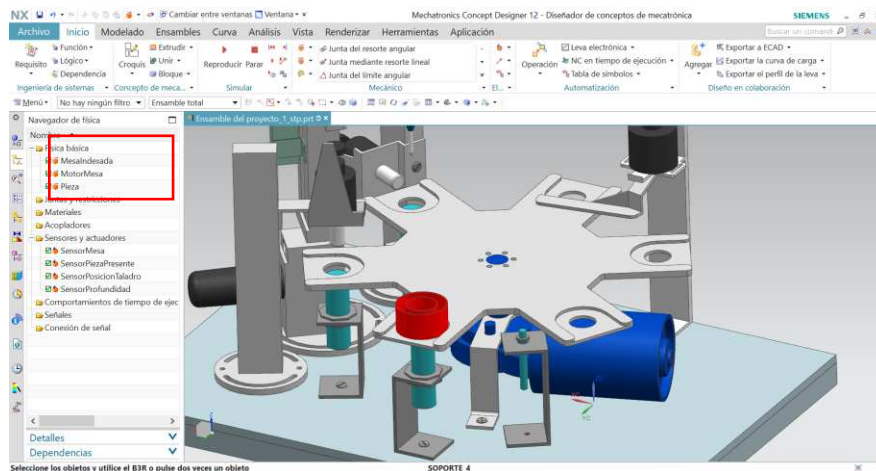


Figura III.19 Cuerpos rígidos dados de alta.

Con respecto al movimiento rotatorio de la mesa y del componente que carga al taladro, donde el segundo movimiento es lineal, es necesario tanto al motor como al actuador lineal darlos de alta como cuerpos rígidos, asignándoles juntas fijas a ambos, ya que estos dos actuadores serán definidos como referencias (base) en la junta de charnela para la mesa y junta deslizante para el actuador lineal.

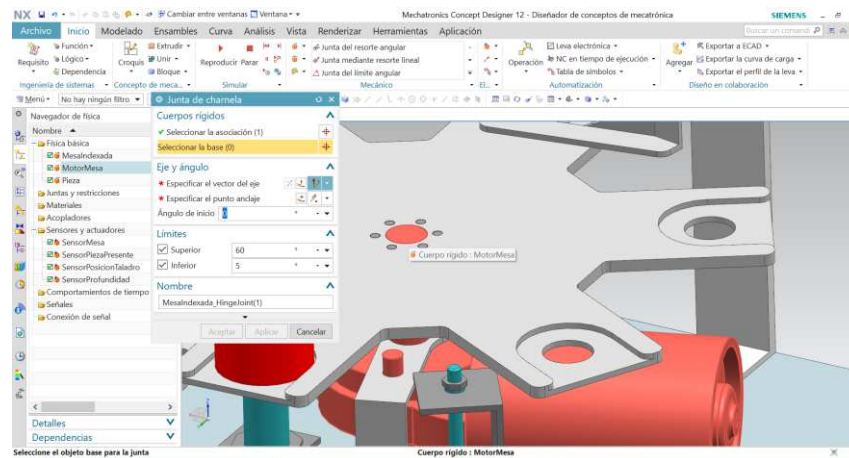


Figura III.20 Configuración junta de charnela motor-mesa.

Como se indica en la figura III.20, anterior, el componente mesa es el que debe ser elegido como asociación en la caja de diálogo de la junta de charnela y el motor como base. El siguiente paso es especificar el eje del movimiento rotatorio y el punto de anclaje de este eje, al colocarse en la opción de especificar el vector del eje, aparece un SISC (Sistema de coordenadas) en el área de trabajo de color violeta oscuro, es necesario dar clic en el vector requerido, que se observa después de color café, ver figura III.21, en el caso de cambiar el sentido solo hay que dar doble clic sobre el vector.

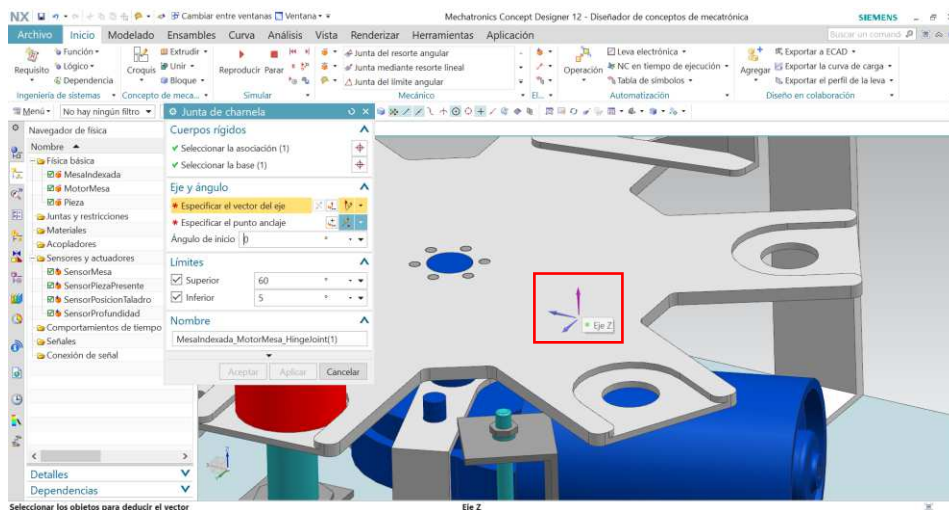


Figura III.21 Selección vector que indica el sentido del giro de la mesa.

Para el punto de anclaje es necesario hacer un acercamiento al círculo azul del centro de la mesa, el cual forma parte del motor reductor, y se observará un círculo remarcado con el centro del mismo, elegir el punto, en ese momento el eje anterior se traslada a ese punto de anclaje. No hay ángulo de inicio y tampoco límites del movimiento, sólo hay que poner el nombre: *Mesa_Motor_JuntaRotatoria*. Ver figura III.22.

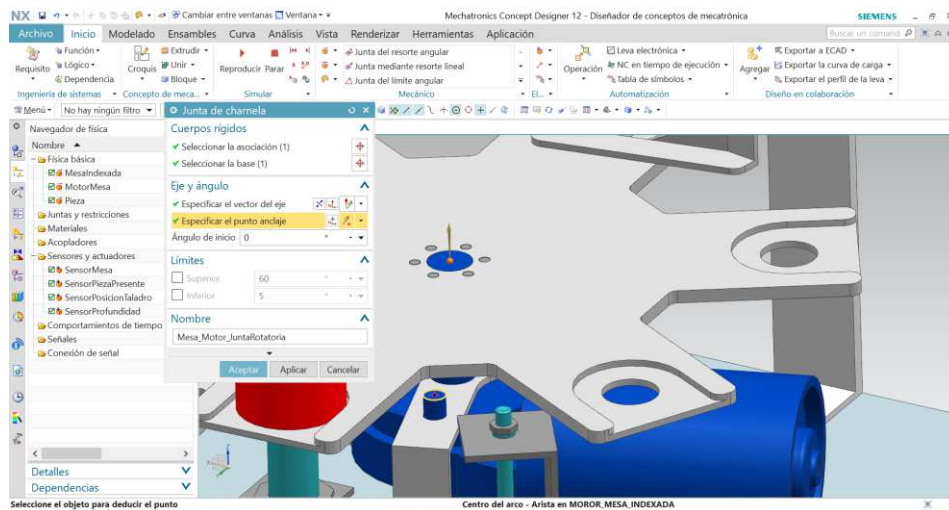


Figura III.22 Punto de anclaje del vector de referencia sobre el giro mesa.

No olvidar asignar junta fija al motor, ya que sí se simula se caen ambos componentes: mesa y motor, como se puede ver en la figura III.23.

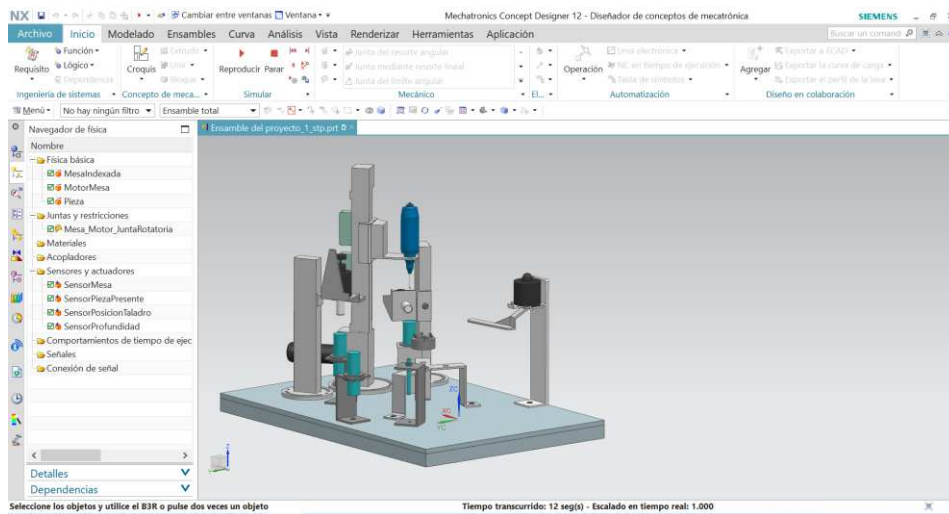


Figura III.23 Simulación, mesa y motor caen por efecto de la gravedad.

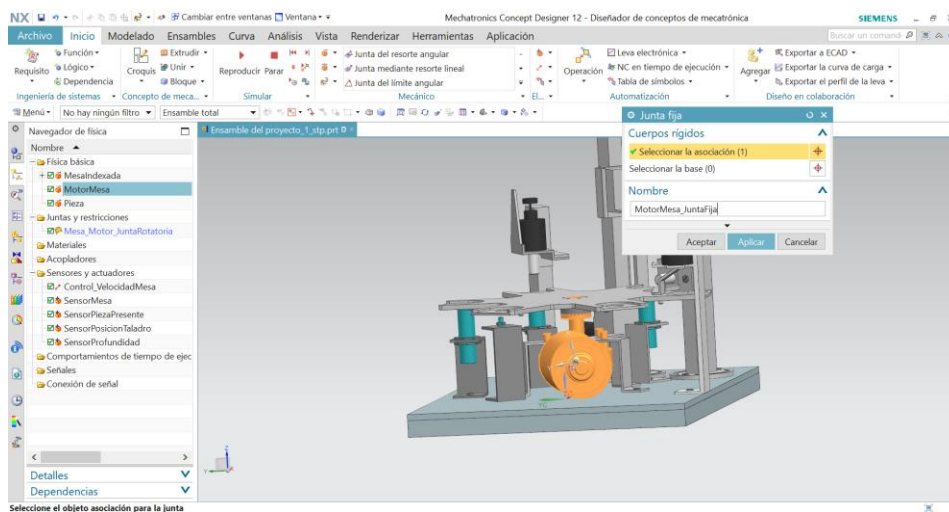


Figura III.24 Junta Fija al motor reductor.

Al colocar la junta fija al motor, se procede a generar un control de velocidad a la mesa y se simula para analizar el movimiento configurado. Ver figura III.25.

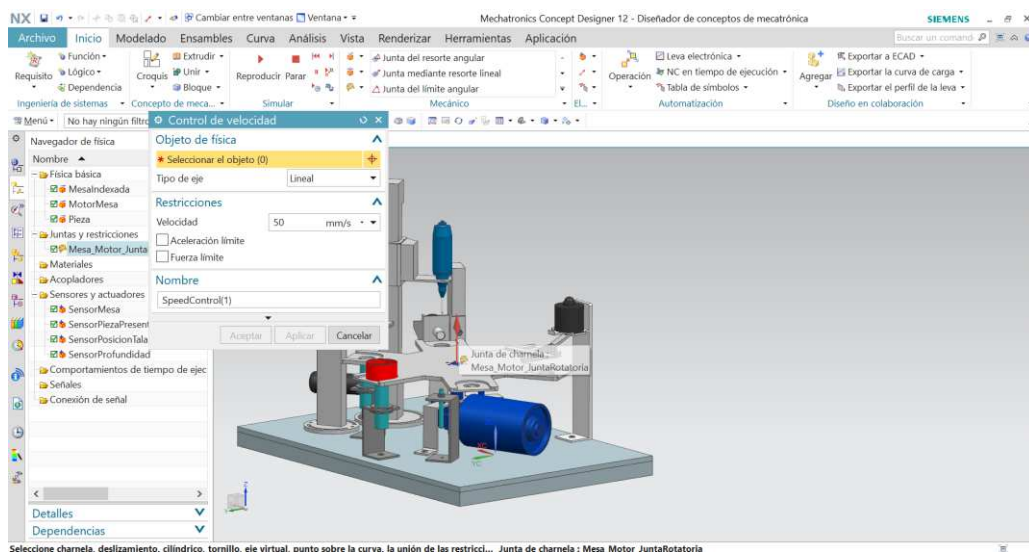


Figura III.25 Configuración de control de velocidad junta de charnela.

En la caja de diálogo para el control de velocidad, se tiene las siguientes opciones de configuración:

- Objeto de la física: Dar clic en el vector que aparece en la junta de charnela, ver figura anterior (III.25).
- Restricciones: si se elige aceleración límite y fuerza límite, esto lo calcula el *solver* propio del software con los parámetros especificados previamente en la sección de preferencias, al simular elegimos solo 20°/s.
- Nombre: dar un nombre según la función, *ControlVelocidad_Mesa*.

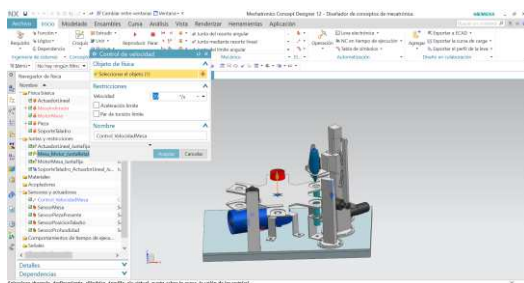


Figura III.26 Configuración control de velocidad mesa indexada.

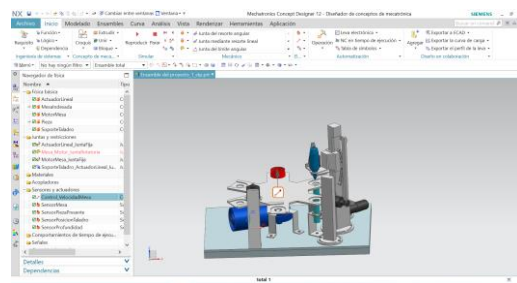


Figura III.27 Visualización control de velocidad, navegador-gemelo digital.

Se continúa con la configuración de la junta deslizante, recordar que se debe dar de alta como cuerpo rígido al actuador lineal y al soporte conjuntamente con el taladro. Ver figuras III.28 y III.29.

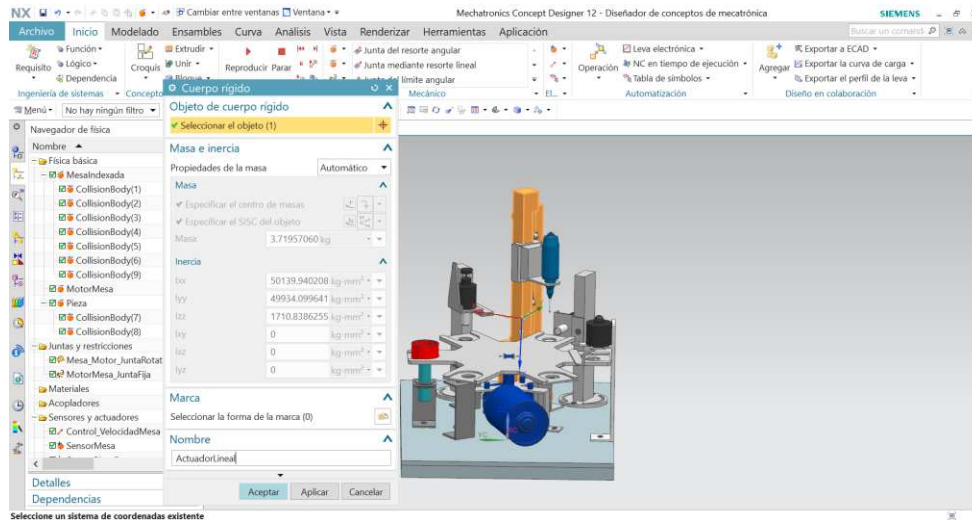


Figura III.28 Cuerpo rígido actuador lineal.

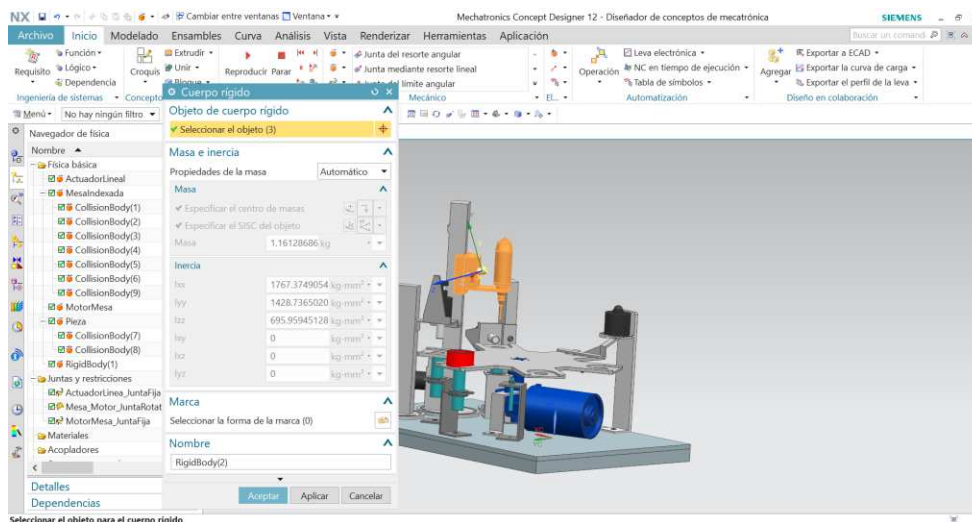


Figura III. 29 Cuerpo rígido de la pieza que soporta al taladro y taladro.

Hay que definir la junta fija al actuador lineal, ya que al agregar el cuerpo rígido del soporte del taladro, al simular se caen las piezas acopladas debido al efecto de gravedad, por lo que hay definir además el acoplamiento entre el actuador lineal y el soporte, éste será de tipo junta deslizante. Ver figura III.31.

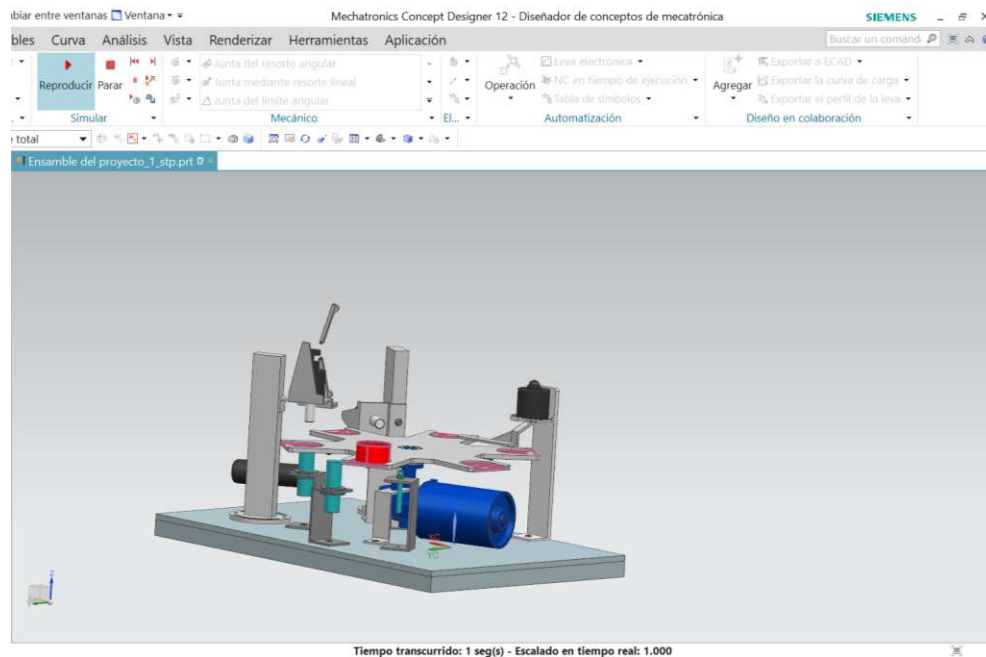


Figura III.30 Simulación sin junta fija del actuador lineal.

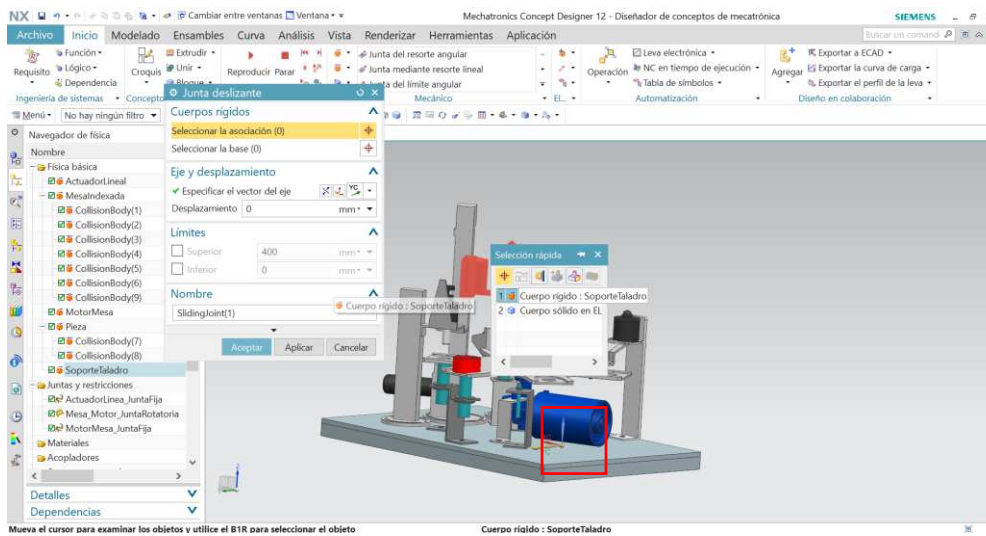


Figura III.31 Junta deslizante soporte taladro.

En la caja de diálogo de la junta deslizante se tienen las siguientes opciones:

- Cuerpos rígidos: componentes que se mueven entre sí.
 - Seleccionar la asociación: componente que se mueve de forma lineal con referencia a otro: *SoporteTaladro*.
 - Seleccionar la base: componente que es referente al movimiento lineal: *ActuadorLineal*.

- Eje y desplazamiento: se indica en el SISC que se observa en la figura III.31, indicado en el recuadro rojo, con solo dar clic en el vector en el sentido del movimiento, que en este caso es hacia arriba o hacia abajo, los interruptores límite como ya se mencionó anteriormente, están conectados eléctricamente de tal forma que al accionarse se desenergiza el motor y se cambia el giro, para luego subir sí bajo y viceversa.
- Límites: aquí se coloca la distancia mínima y máxima del movimiento lineal, es conveniente dejar la inferior como 0, definida desde el diseño en el ensamble.
- Nombre: colocar un nombre acorde a la función:
SoporteTaladro_ActuadorLienal_JuntaDeslizante.

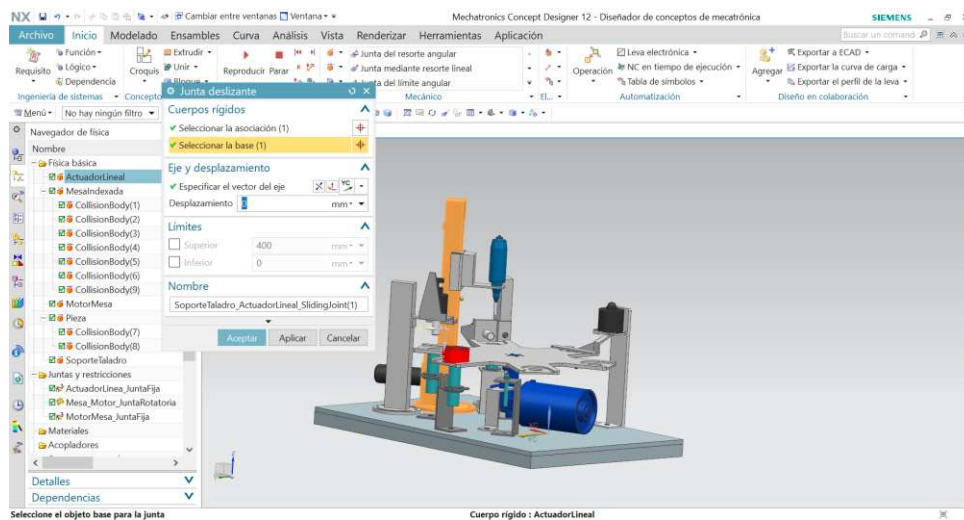


Figura III.32 Configuración junta deslizante.

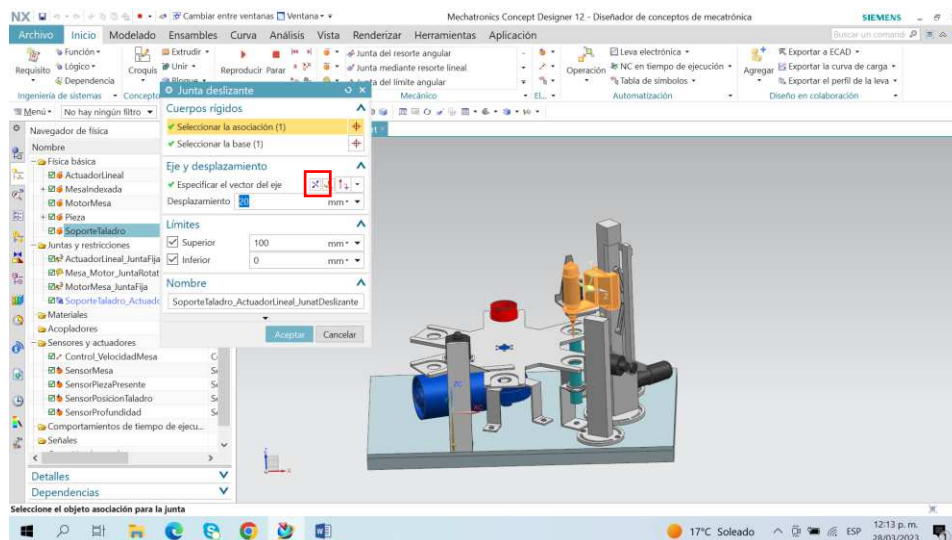


Figura III.33 Junta deslizante soporte taladro.

Se puede validar en la simulación como se puede ver los resultados en la figura III.34, solo fue para bajar, se tendría que cambiar el vector en el área de la caja de diálogo en: Especificar el vector del eje con solo dar clic en el recuadro en rojo de la figura III.33, donde se escribió un desplazamiento de 20 mm con un límite superior de 100 mm.

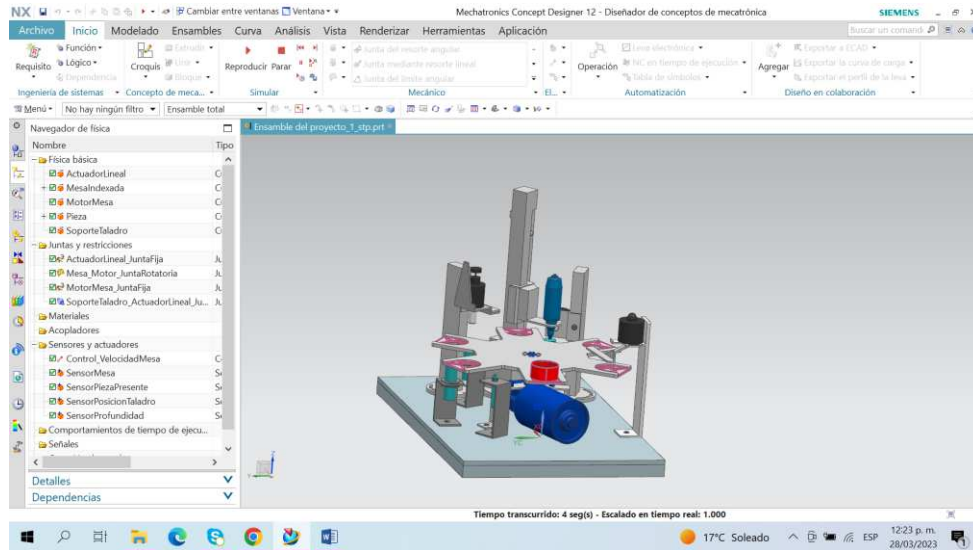


Figura III.34 Simulación descenso soporte taladro, según especificación junta deslizante.

A continuación se definen los cuerpos de colisión, tanto de la pieza de trabajo (color rojo en el modelo) y las seis posiciones de la mesa indexada donde se sujeta dicha pieza. Ver figuras siguientes:

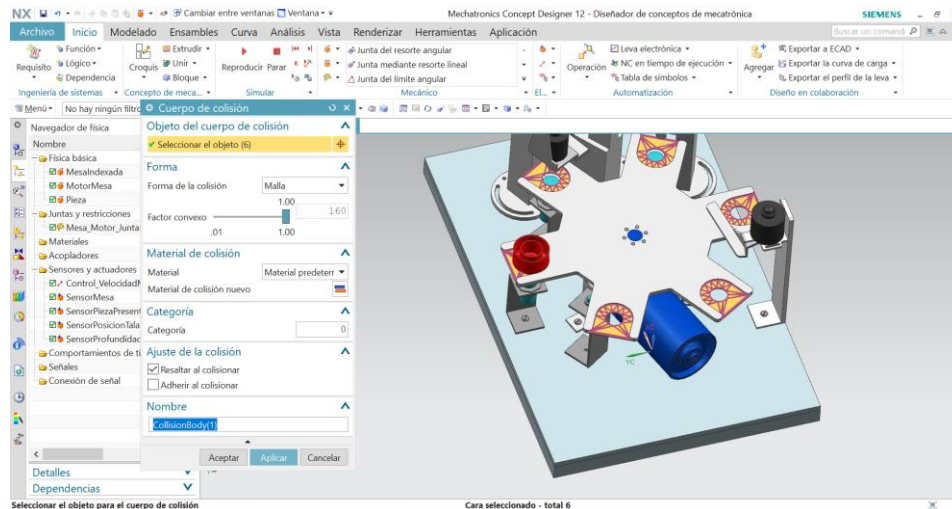


Figura III.35 Cuerpos de colisión mesa indexada.

En la caja de diálogo del cuerpo de colisión se tienen las siguientes opciones:

- Objeto del cuerpo de colisión: aquí se eligen todas las superficies del componente que tendrán contacto con otro elemento del modelo.
- Forma: se puede elegir entre distintas formas (cilindros, cuadrados, esferas..., etc), para las irregulares se selecciona de tipo Malla con incluso resolución de la misma, la cual se fija con el factor convexo de 1.0.
- Material de colisión: aquí se puede cambiar, si se deja en predeterminado, será la del modelo diseñado.
- Categoría: aquí se introducen números enteros que permiten definir categorías, lo cual nos permite sensar dependiendo de la categoría definida.
- Nombre: Colocar un nombre según la función, sino el software enumerará de forma secuencial.

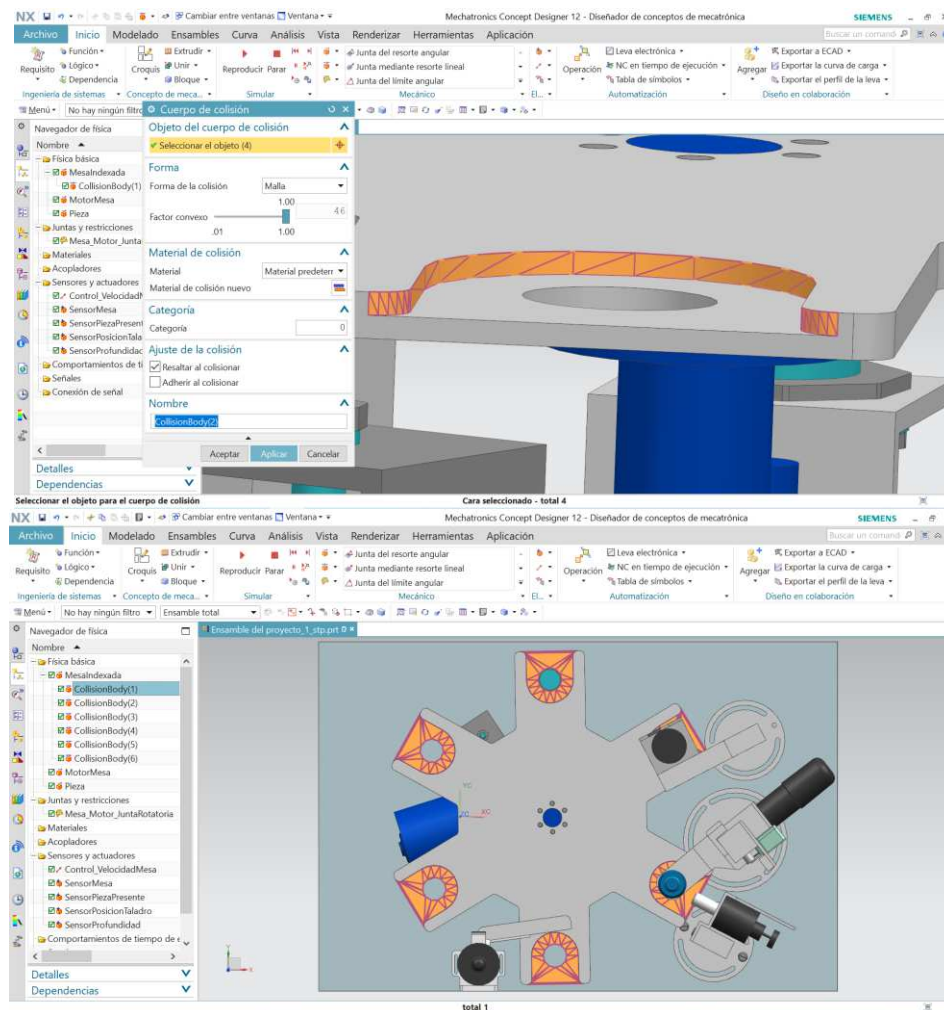


Figura III.36 Cuerpos de colisión mesa indexada.

En la figura III. 37 se puede observar las superficies definidas como cuerpos de colisión de la pieza de trabajo.

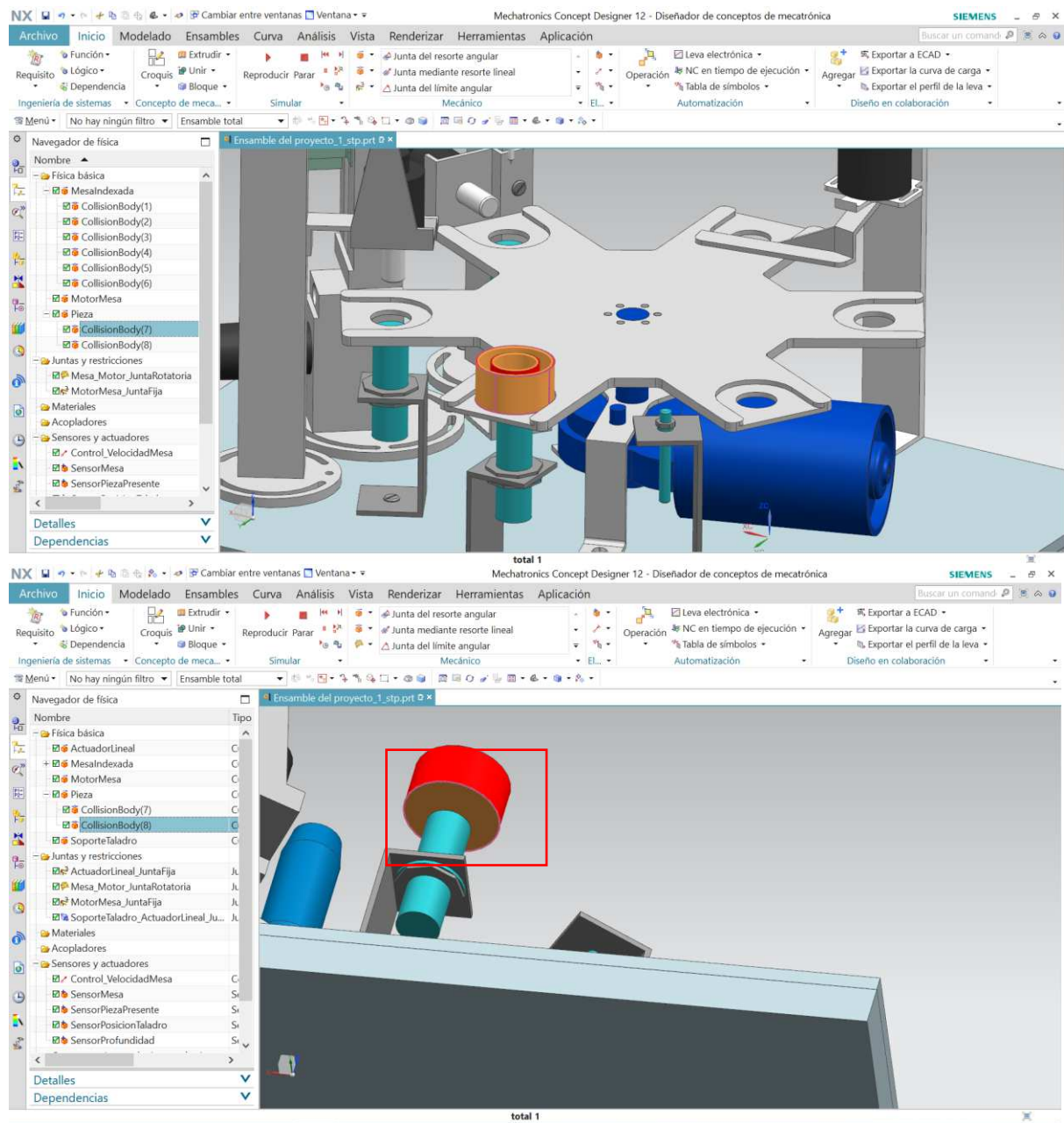


Figura III.37 Superficies de colisión para la pieza de trabajo.

Hasta el momento tenemos definido en el navegador de física lo que se observa en la figura III.38.

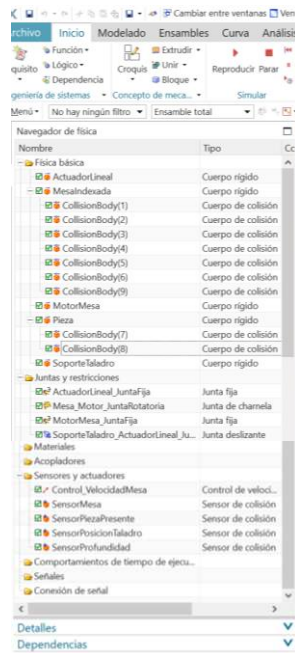


Figura III.38 Navegador de física.

Recordar que los sensores capacitivos detectan cualquier material, en el gemelo digital se hicieron pruebas colocando diferentes categorías en la superficie de colisión del cuerpo a detectar y en las 6 superficies de la mesa que sujetan la pieza, los 3 sensores capacitivos siguen detectando las 6 superficies y no solo la pieza. Para validar lo anterior hay que dar clic en el nombre de cada sensor en el navegador para activar la herramienta de inspección. Ver figura III. 39.

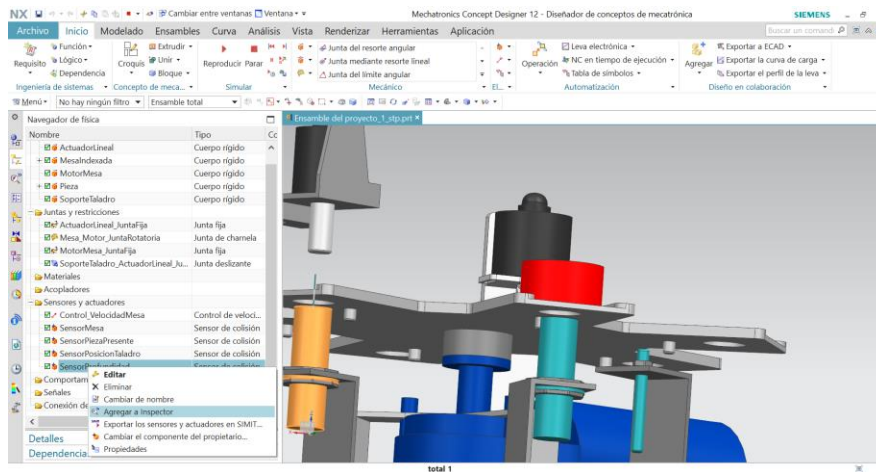


Figura III.39 Herramienta de inspección para estados de los sensores.

Se procede a la simulación y se validará lo que se describió en el párrafo anterior, por lo cual es necesario comentar que sucedería lo mismo en la estación de procesamiento

física, por tal motivo se agrega un circuito eléctrico con relevadores que desactiva al motor reductor de la mesa con apoyo de la señal del sensor inductivo. Todos los sensores son activos y en la posición de la mesa en la figura III.40, ninguno de los sensores se activa por tal motivo se observan falsos en sus estados en activado.

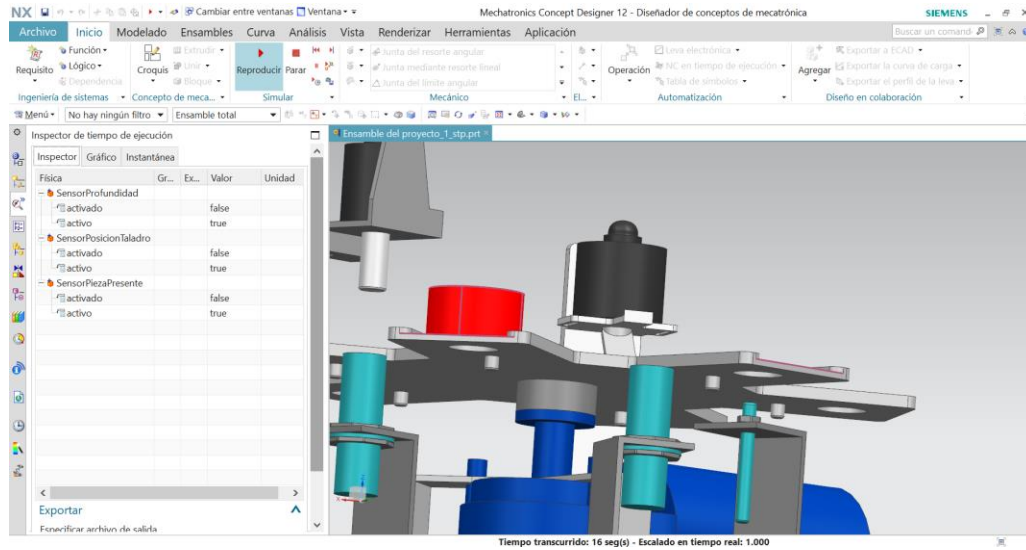


Figura III.40 Estados lógicos (0) de los sensores indicados en la sección de inspección.

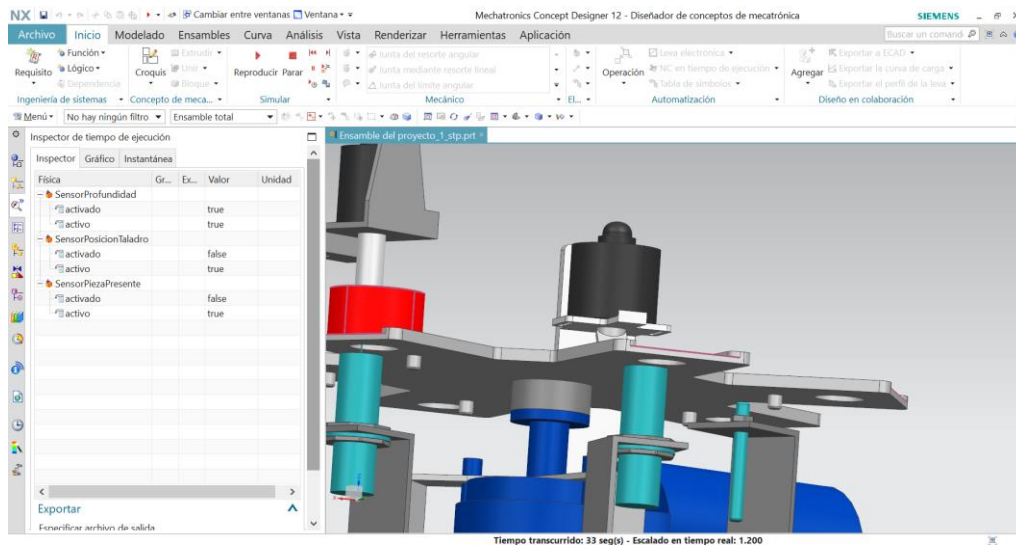
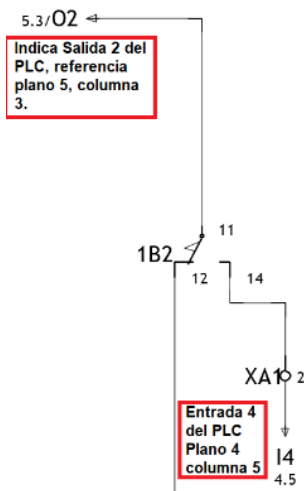


Figura III.41 Estado lógico (1) sensor de profundidad.

Diagrama de conexión de un sistema de control de iluminación. El diagrama muestra una conexión entre un 'Conector 25 pines' (con un chip X5023) y un 'Cableado' que incluye 'Clemas' (XAI:1 a XAI:0V 7/8), un relé 'R3' y 'Relevadores' (K1 a K5). El sistema está alimentado por 'Canaletas' (canales) en la parte superior e inferior, con conexiones a 'IP_N_FO' y 'IP_FI'.

Para la correcta interpretación de los esquemas eléctricos, se explica a continuación la nomenclatura empleada en la figura III.43:



65

En el primer plano se observan todos los sensores que tiene la estación de trabajo, en la parte inferior de cada una de las figuras se describe la función de los componentes:

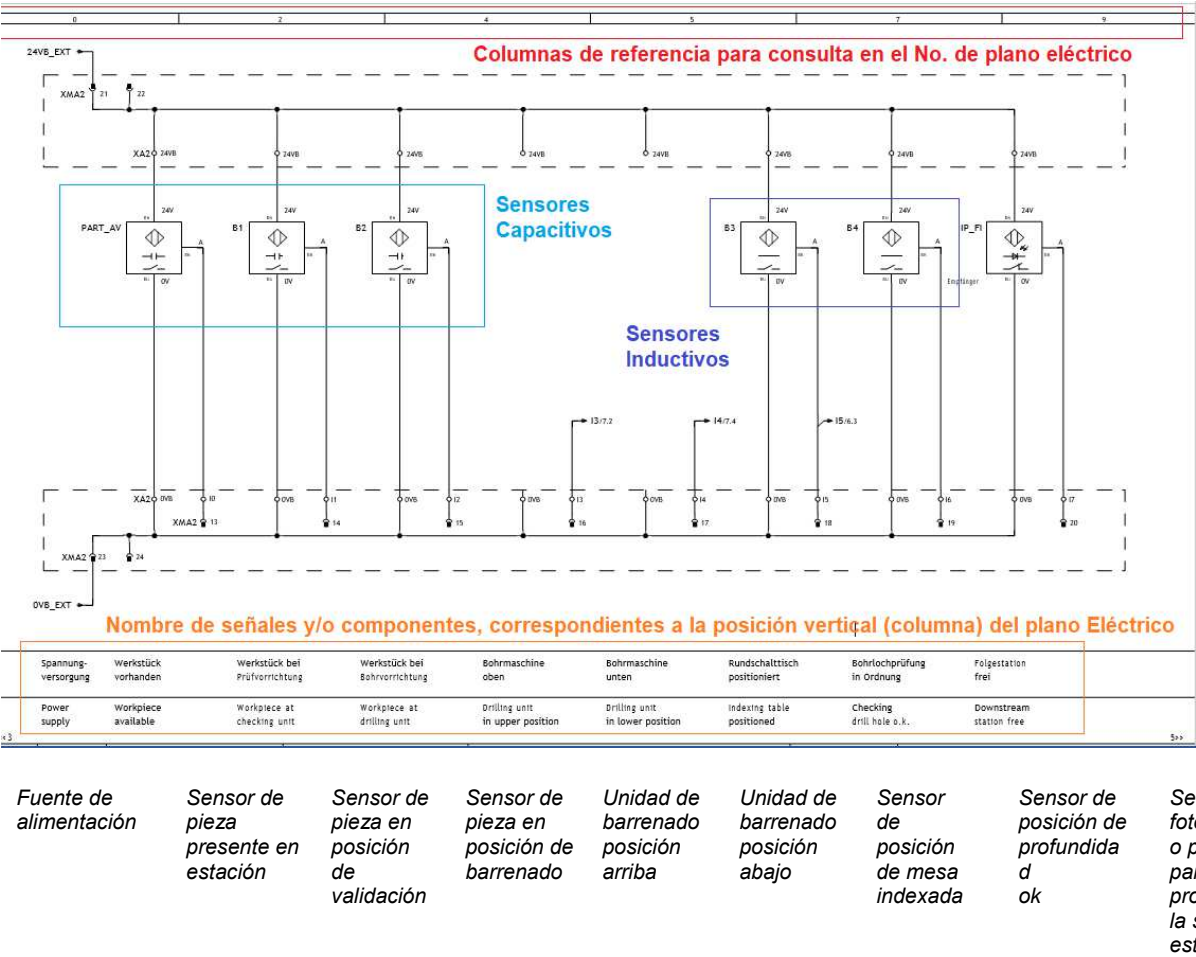


Figura III.44 Conexión sensores.

En el siguiente diagrama (figura III.45) se observa la conexión de los actuadores que controlan el componente que expulsa, el que sujeta la pieza de trabajo para la acción de barrenado y el que verifica si la pieza es colocada correctamente para después ser barrenada.

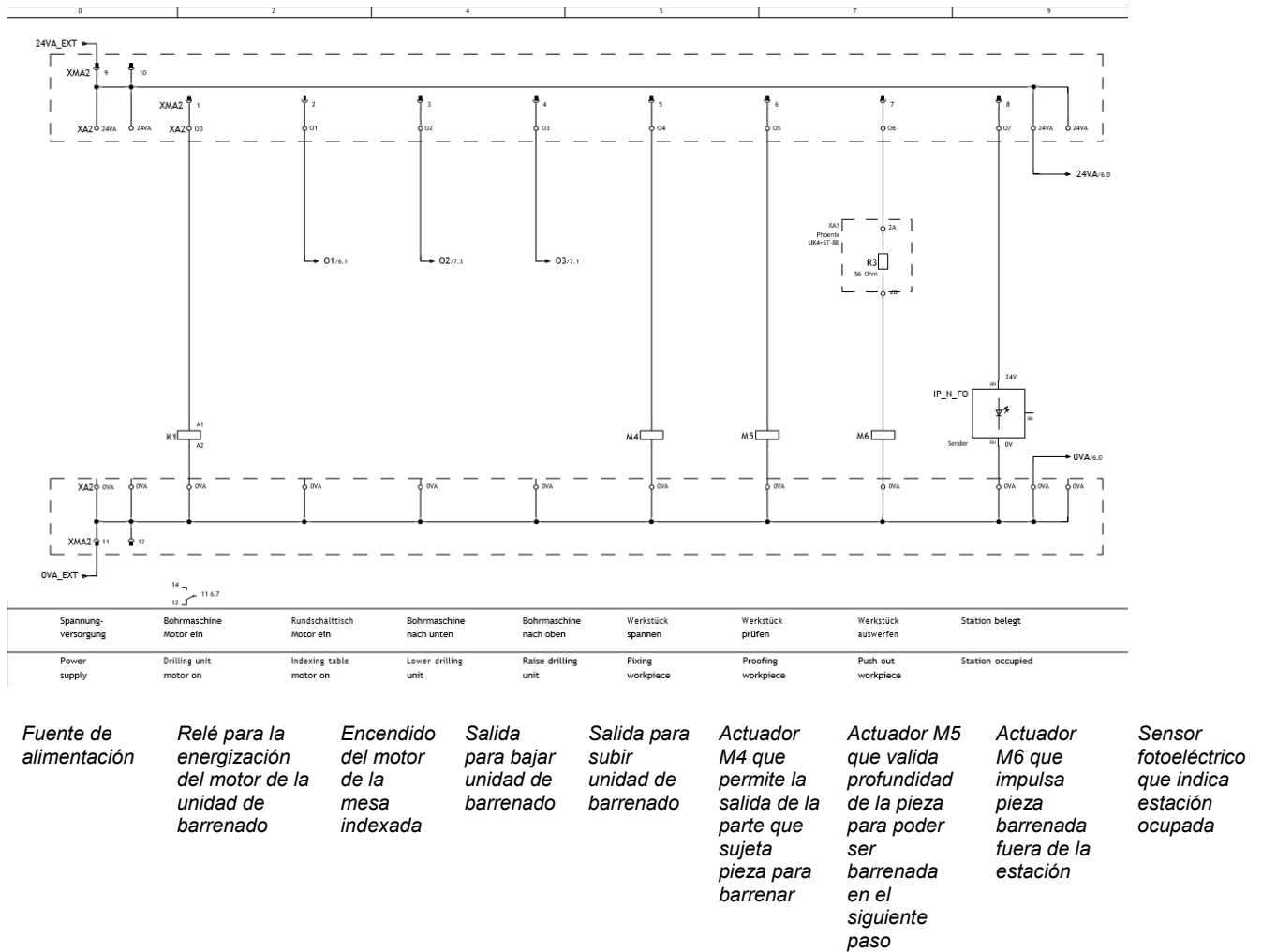
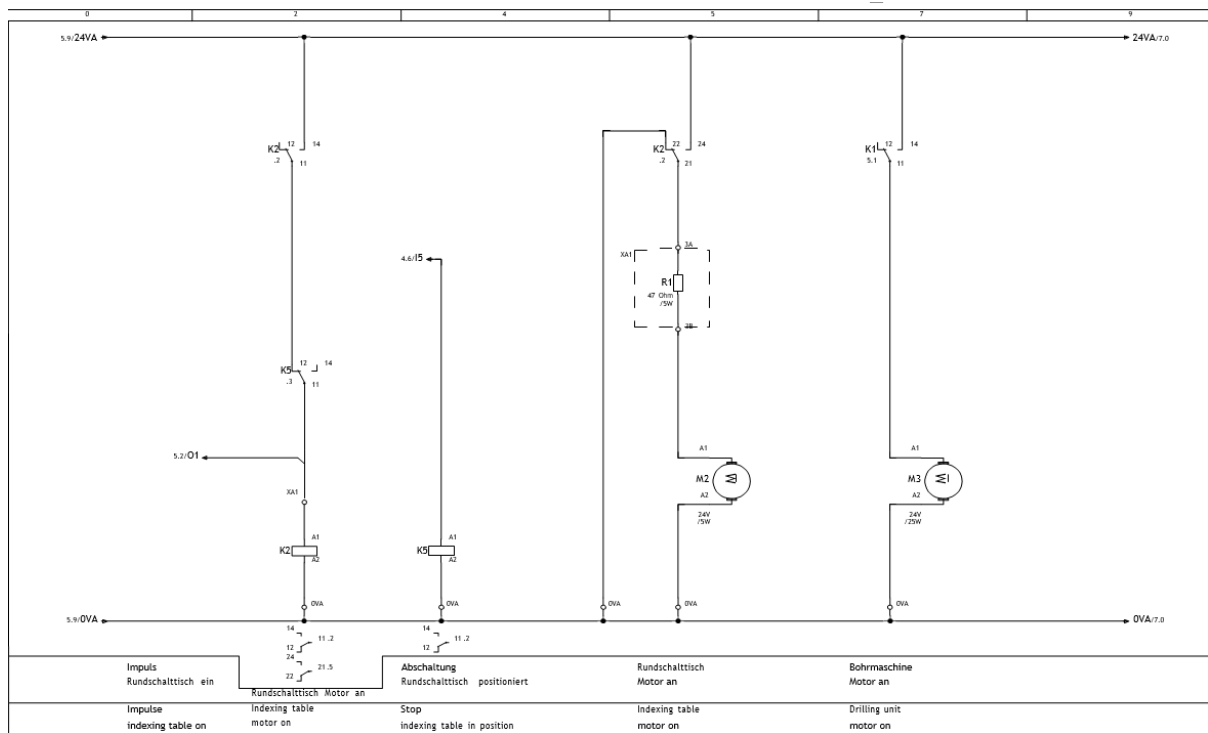


Figura III.45 Conexión actuadores M4, M5 y M6.

En la figura III.46, se observa las conexiones para el control de la mesa indexada.



Relé que permite la energización del motor M2 que mueve la mesa indexada

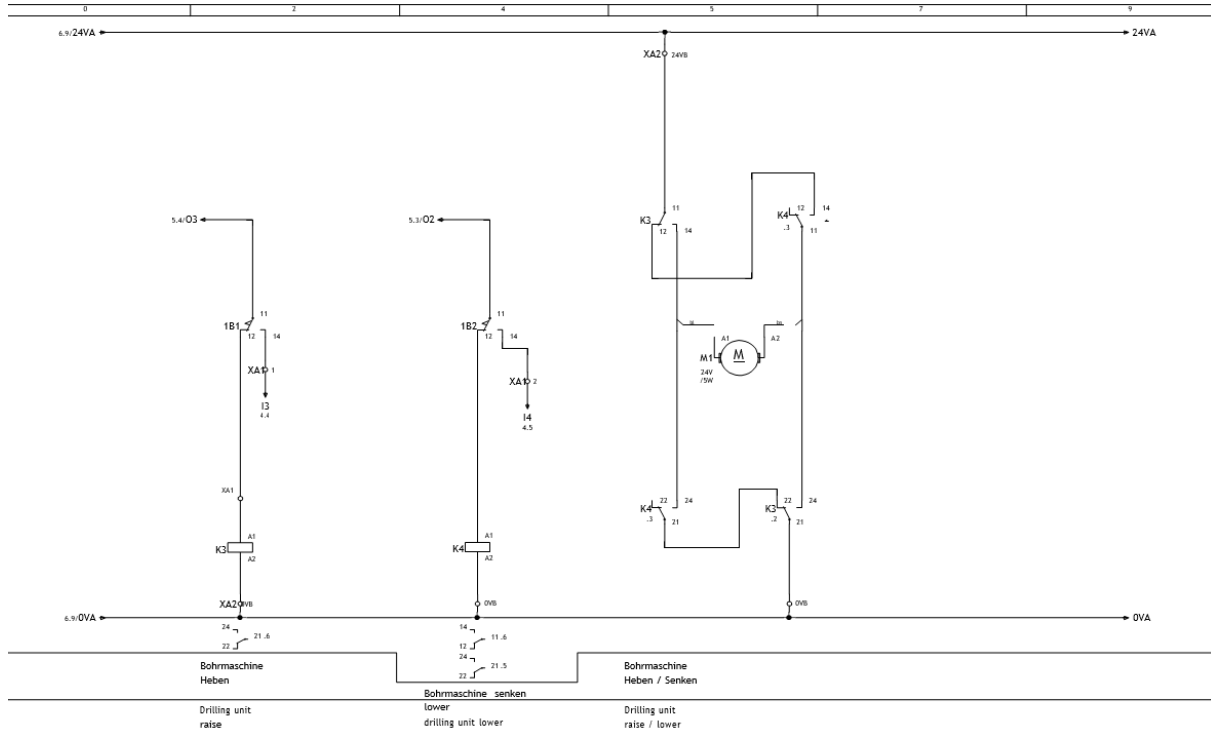
Relevador que para al motor M2 que mueve la mesa indexada

Motor M2 que mueve mesa indexada

Motor M3 que energiza la unidad de barrenado

Figura III.46 Esquema eléctrico control mesa indexada

Por último se presenta el esquema eléctrico sobre el control del actuador lineal, el que sujeta al taladro.



Limit switch 1B1 que indica unidad de barrenado ascendiendo.

Limit switch 1B2 que indica unidad de barrenado bajando

Motor M1 del actuador lineal que sujeta la unidad de barrenado, conectado a los relevadores: K3 Y K4 que permiten el cambio de giro del motor M1

Figura III.47 Esquema eléctrico control actuador lineal.

Para finalizar el *precomisionamiento virtual* a continuación se describirá el cómo se valida al actuador lineal y así como también el sistema de expulsión de la pieza de la estación.

Se configura la junta deslizante donde el actuador lineal, es la base y la asociación es el soporte del taladro, ver figura III.48. En la configuración de la junta se coloca desplazamiento 0 mm.

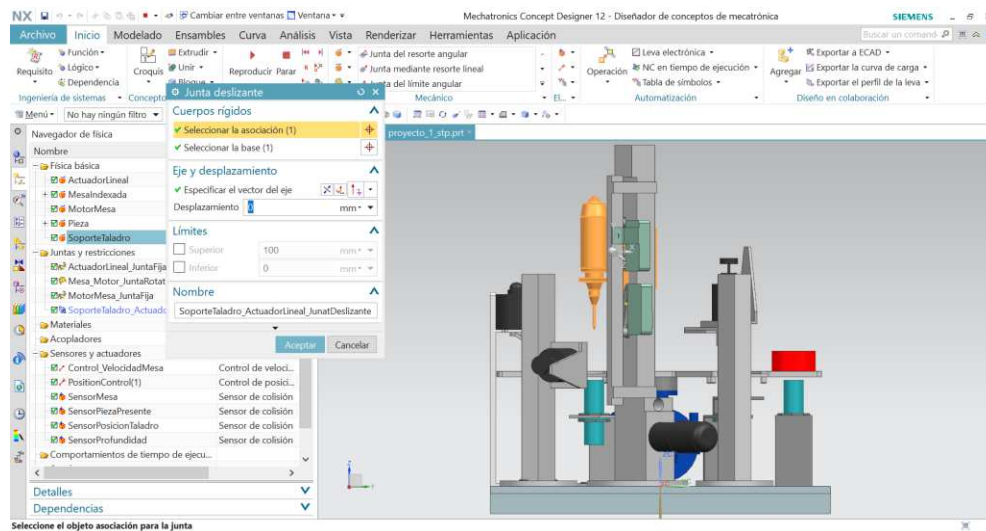


Figura III.48 Junta deslizante soporte taladro.

Se procede a realizar un *Control de Posición*, en la caja de diálogo que se observa en la figura III.49, en el recuadro rojo se observa una flecha grande en azul, hay que dar clic en ella para elegir que sobre dicha junta deslizante se aplicará un control de posición.

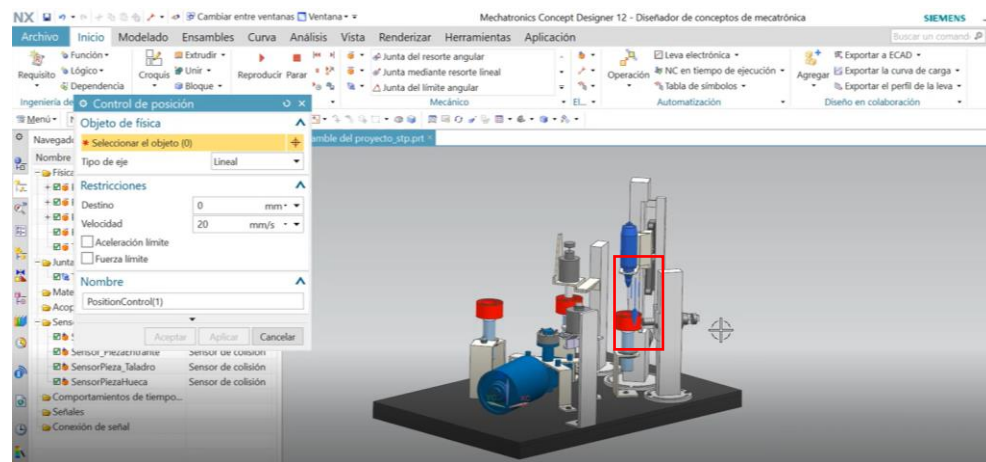


Figura III.49 Control de posición soporte taladro.

En la figura III.50 se colocan los valores de 40 mm de desplazamiento a una velocidad de 10 mm/s, en el navegador de física también se señala con cambio de colores los componentes relacionados con el control de posición, por el momento se validó solo el desplazamiento hacia abajo, el cual es importante para no taladrar más allá de lo necesario y con ello evitar dañar la pieza de trabajo y poder ajustar la posición de los interruptores límite. El color naranja del vector indica el sentido del movimiento de dicho control, el azul indica la posibilidad de configurar también ese sentido del movimiento de la junta deslizante.

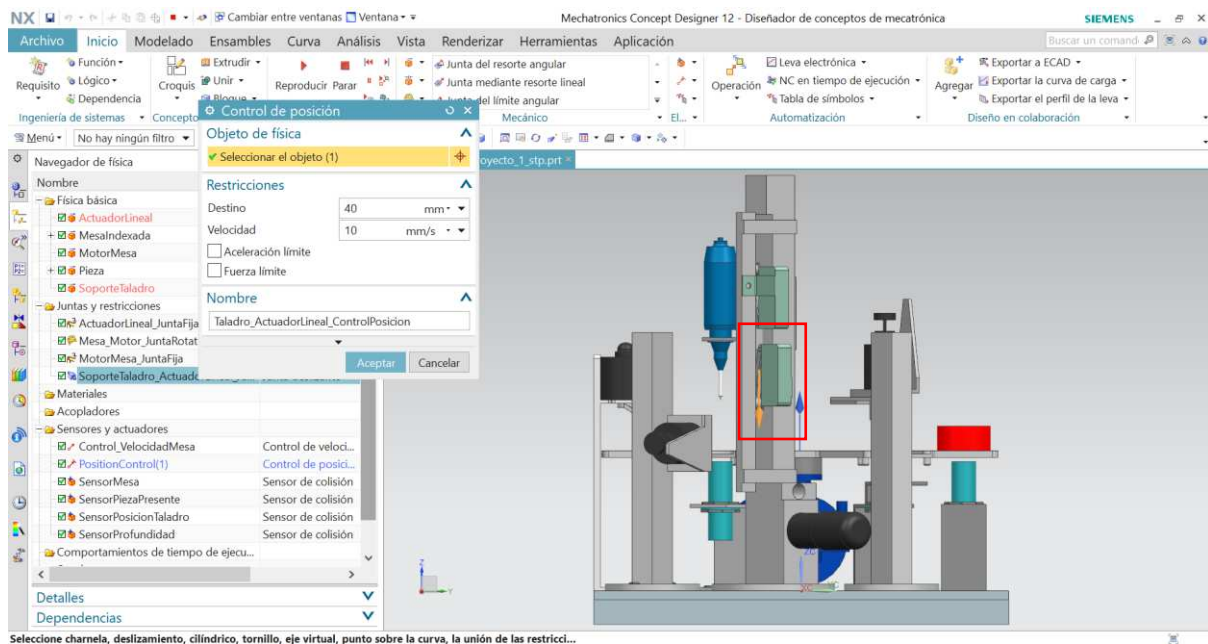


Figura III.50 Datos configuración control de posición soporte taladro.

Para el expulsor hay que declararlo primeramente como cuerpo rígido al igual el actuador que lo mueve, ver figura III. 52, dejando en ambos casos que la masa y el vector de inercias se calculen de manera automática.

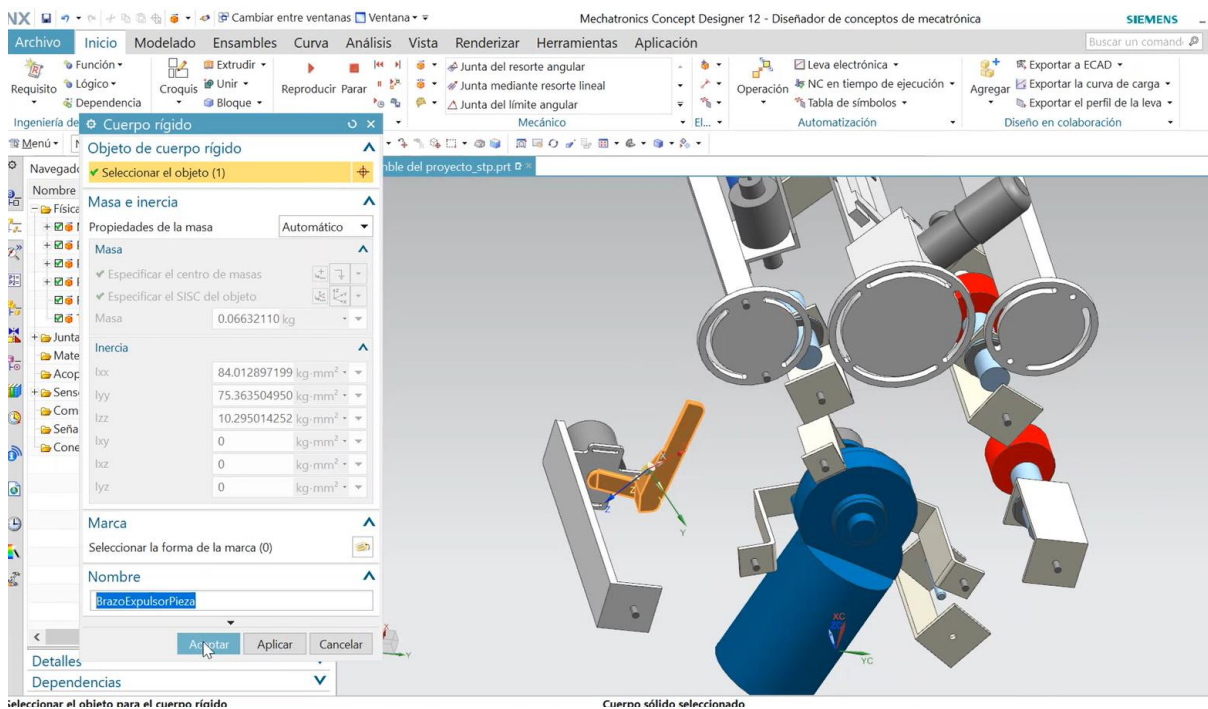


Figura III.51 Cuerpo rígido expulsor.

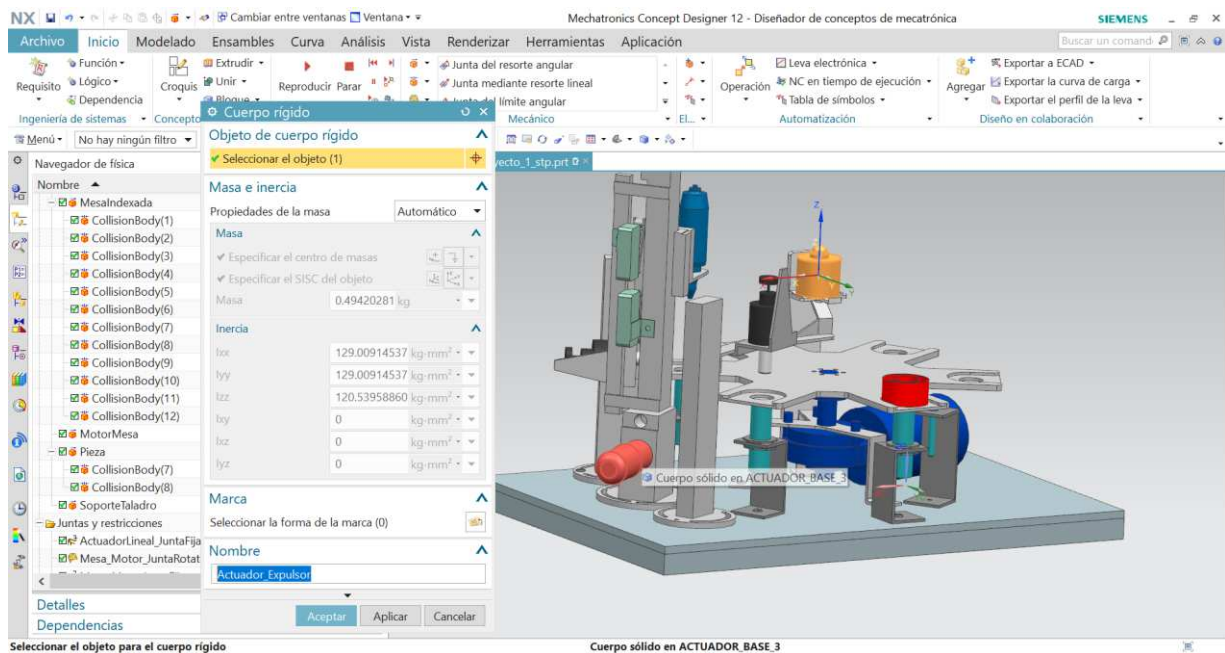


Figura III.52 Cuerpo rígido actuador que mueve al expulsor.

Se fija al actuador con una junta fija y se asigna un nombre según su función. Ver figura III.53.

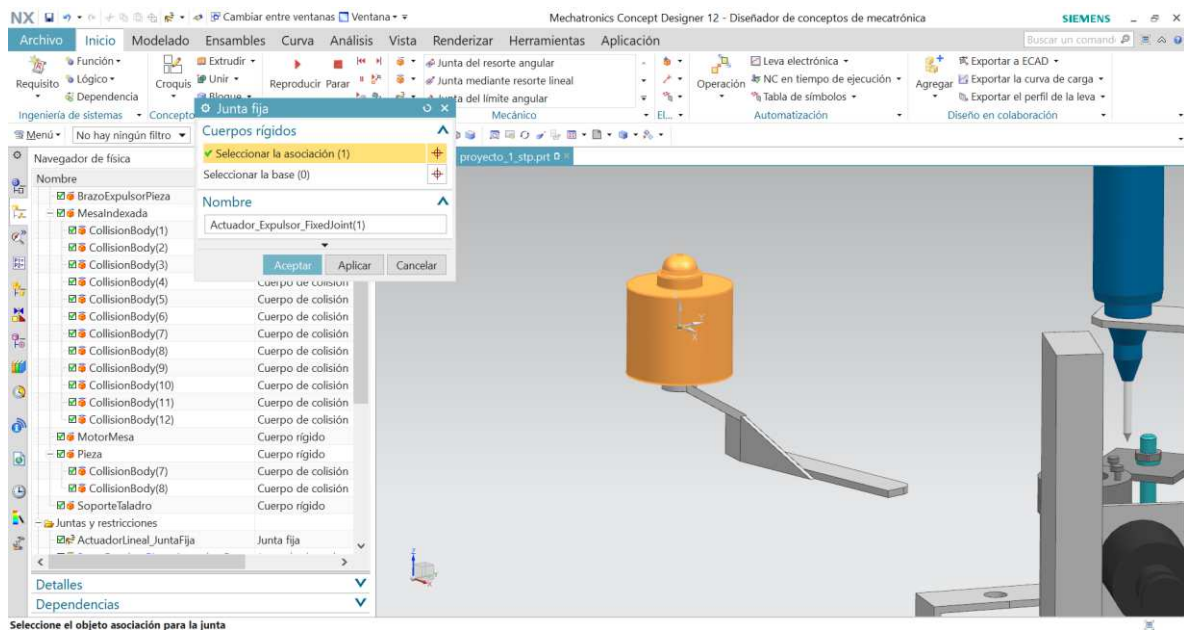


Figura III.53 Junta fija actuador que sostiene al expulsor.

Después se define la junta de charnela para el movimiento rotatorio del expulsor, en la opción de asociación se selecciona al expulsor y como base se elige al actuador, ver figura

III.54. El vector que se encuadra en la misma imagen, se ha elegido como el eje sobre el cual es el movimiento rotatorio.

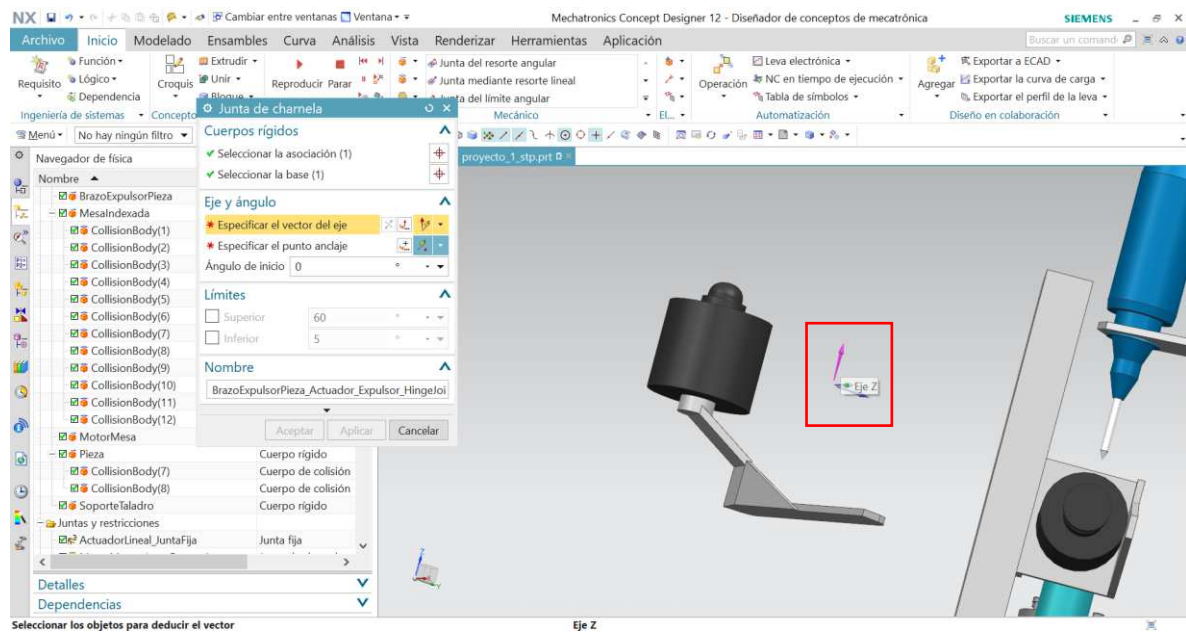


Figura III.54 Junta de charnela para el expulsor.

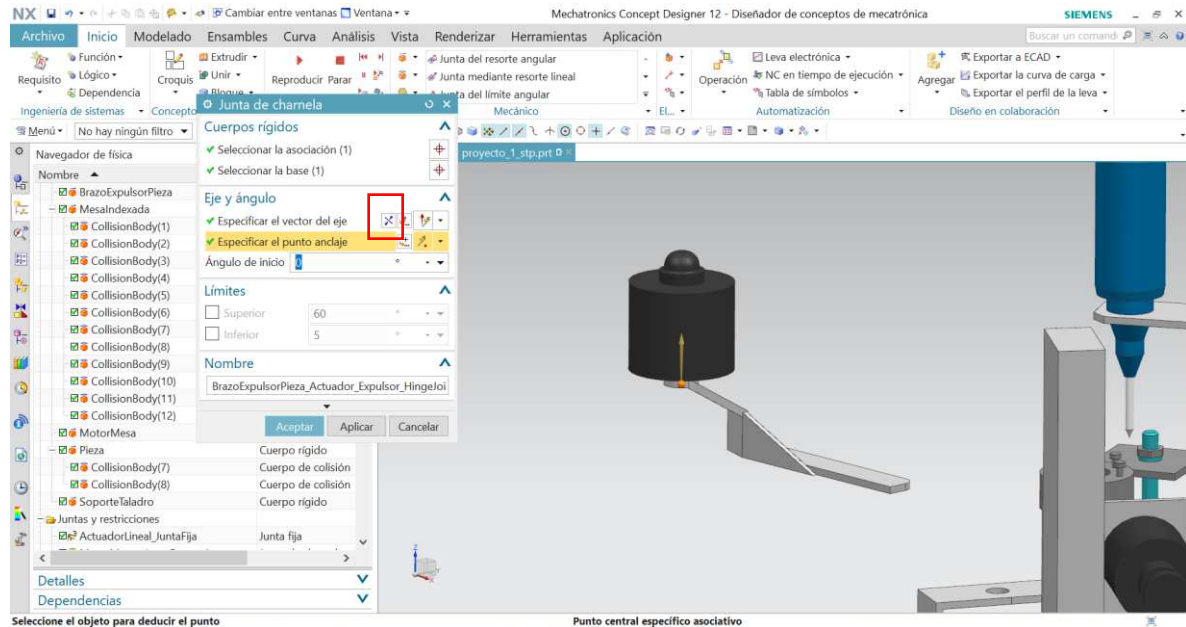


Figura III.55 Eje sobre el cual gira el expulsor.

Al simular se valida que el giro del expulsor debe ser en sentido contrario, por lo que se hace el cambio del sentido, con solo dar doble clic sobre el vector o en la caja de diálogo dar clic en el icono que se encuadra en la figura anterior (III.55).

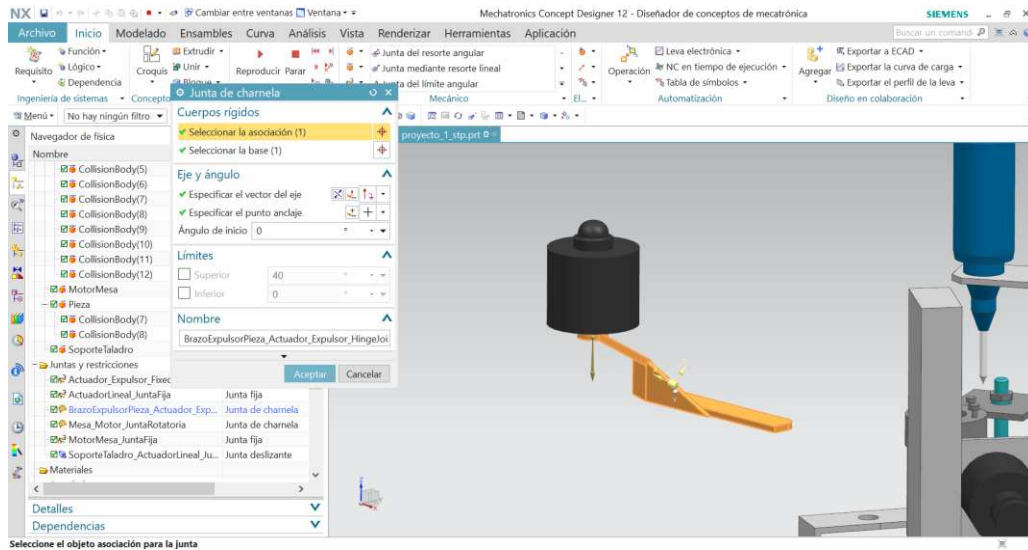


Figura III.56 Configuración final junta de charnела con límites en el movimiento rotatorio.

En la opción de límites se probó con los valores de 110° como límite superior, como límite inferior y con un ángulo de inicio de 0° , para observar en la simulación el movimiento rotatorio esperado, es necesario configurar un control de posición ya que no se requiere la rotación completa.

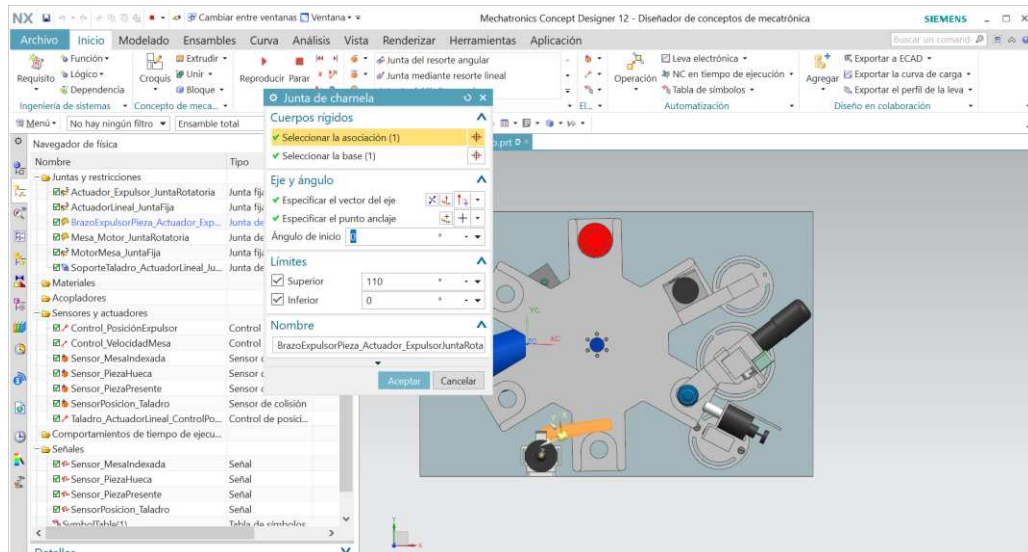


Figura III.57 Junta de charnела en el Expulsor.

En la siguiente figura se indica la posición final de 95° , la cual debe ser menor al límite superior definido en la junta de charnela, a una velocidad de $20^\circ/\text{s}$, en opciones de trayectoria angular se deja el sentido anti horario.

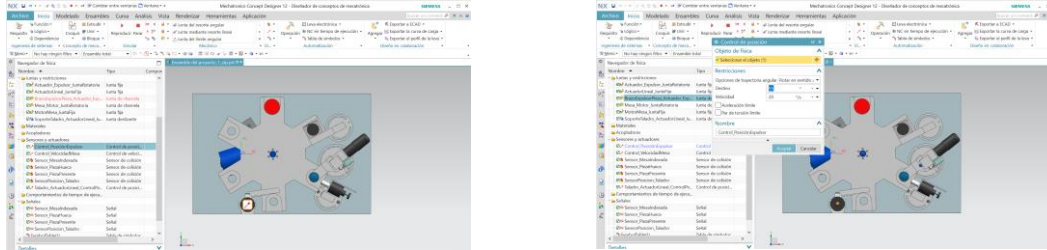


Figura III. 58 Control de posición junta de charnela Expulsor.

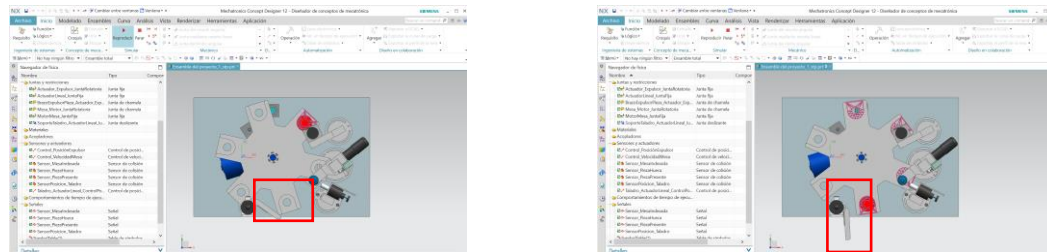


Figura III.59 Simulación Expulsor.

Finalmente para la pieza del expulsor se deben configurar sus superficies de colisión, las cuales se indican con un recuadro rojo en la figura III.60, es necesario esta configuración para que la pieza sea expulsada en la simulación. Dar un nombre a la superficie de colisión según su función, se emplea la forma de colisión de tipo malla con un factor de convexo de 1.0.

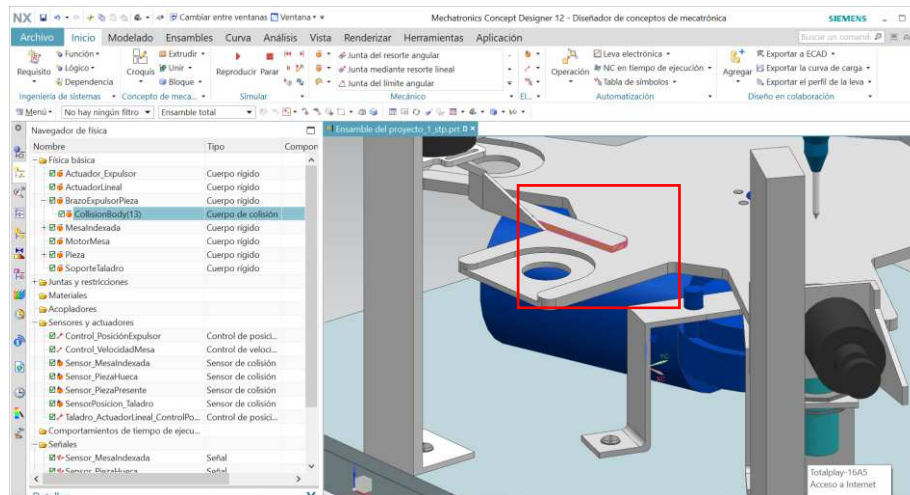


Figura III.60 Cuerpo de colisión expulsor.

Para finalizar el *precomisionamiento virtual* se deben configurar las juntas deslizantes en los cuerpos previamente definidos como rígidos:

1. La pieza que verifica la profundidad del hueco de la pieza colocada correctamente
2. y el que sujeta la pieza cuando va a ser barrenada.

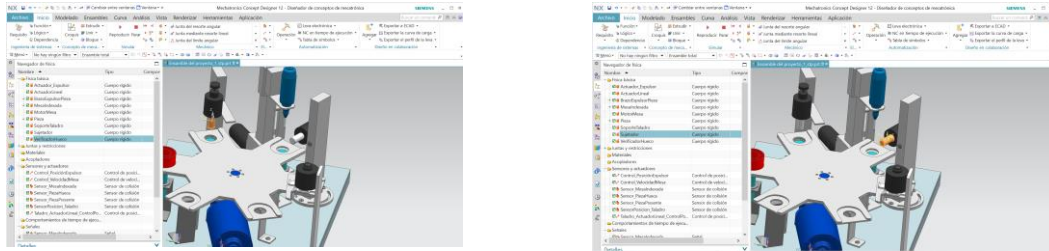


Figura III.61 Cuerpos rígidos *Verificador_Hueco* y *Sujetador*.

Se continúa con la definición de las juntas deslizantes para ambos: *Verificador_Hueco* y *Sujetador*.

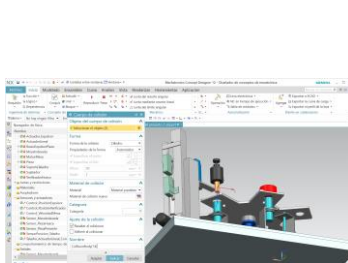


Figura III.62 Superficies de colisión

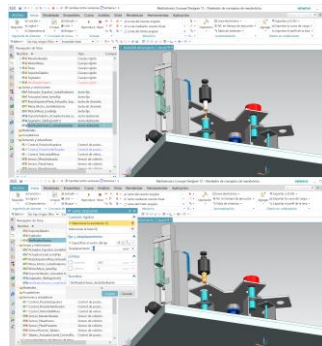


Figura III.63 Junta deslizante verificador, con 8 mm de desplazamiento.

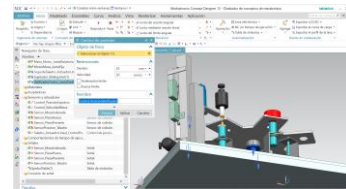


Figura III.64 Control de posición verificador, con límites de desplazamiento.

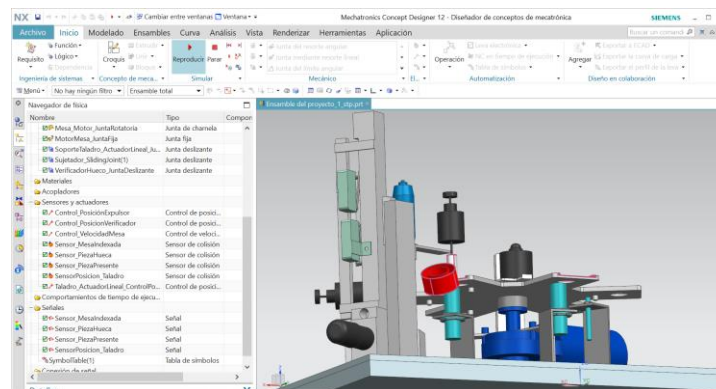


Figura III.65 Simulación colisión pieza con verificador totalmente desplazado.

Se ocultaron piezas para poder visualizar mejor el modelo y configurar la junta deslizante, con un desplazamiento de 20 mm y un límite de 25 mm a una velocidad de 5 mm/s.

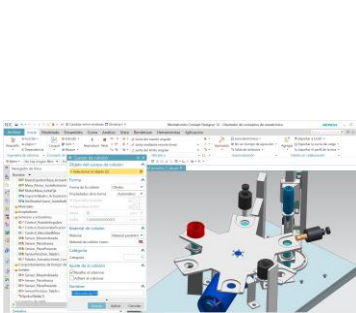


Figura III.66 Superficies de colisión

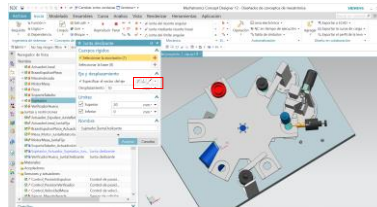


Figura III.67 Junta deslizante sujetador, con 10 mm de desplazamiento.

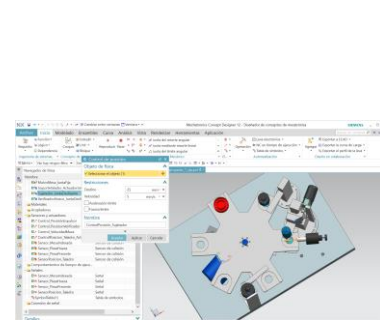


Figura III.68 Control de posición sujetador, con límites de desplazamiento.

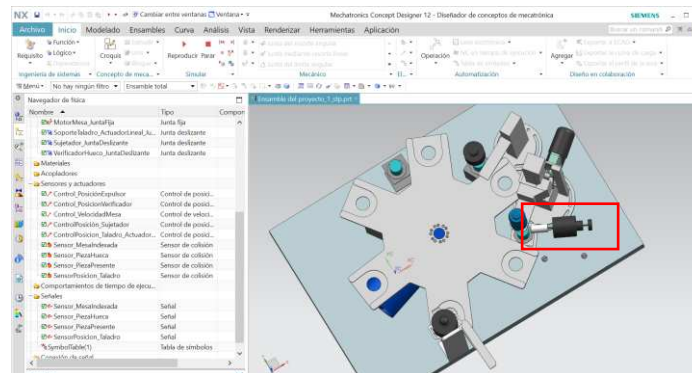


Figura III.69 Simulación.

Finalmente se han terminado las configuraciones de los componentes principales, para las dos últimas juntas deslizantes: la del sujetador y la del verificador, no fue necesario definir los cuerpo rígidos de los actuadores asociados y asignarles una junta fija a éstos; con solo tener los controles de posición correctamente configurados para dichas juntas es suficiente, y poder así realizar un programa con el PLC virtual y validarlo en una simulación conjunta, lo cual se describirá en el capítulo IV; sin embargo es importante mencionar que para el sujetador no se podía definir fácilmente el vector del desplazamiento, debido al SISC del diseño, por lo que se configuró entre dos puntos, en la figura III.67, en el recuadro rojo indica donde es posible realizar el cambio y en la última figura de este capítulo se observa el cuadro de diálogo para ello.

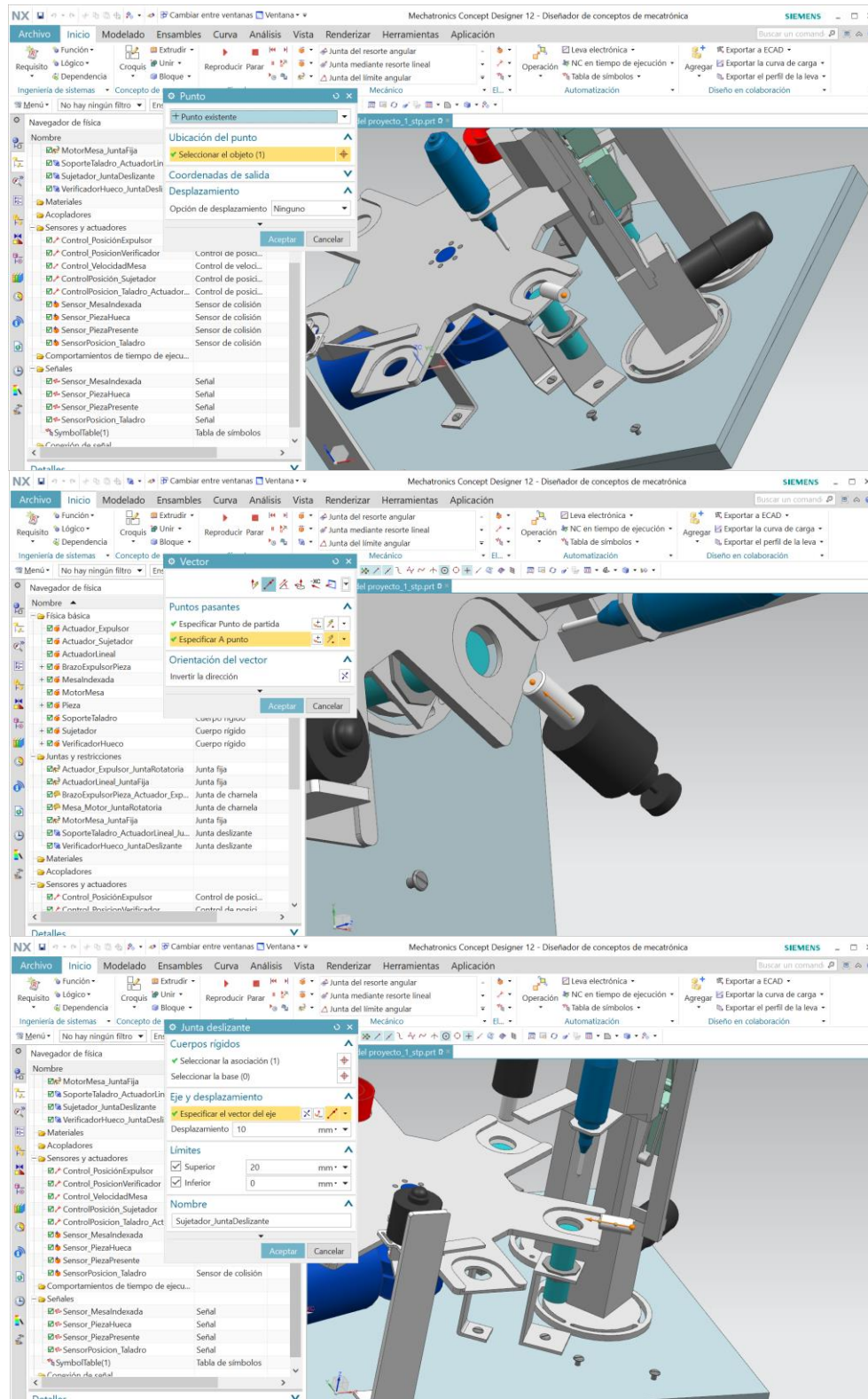


Figura III.70 Configuración vector desplazamiento junta deslizante sujetador.

Para poder comenzar con el *comisionamiento virtual* pruebas generales con el control (PLC) es necesario configurar el sensor inductivo para el control de la mesa indexada, que a continuación se ilustra:

1.- Definir los cuerpos de colisión en la parte posterior de la mesa, como se puede ver en la figura III.71.

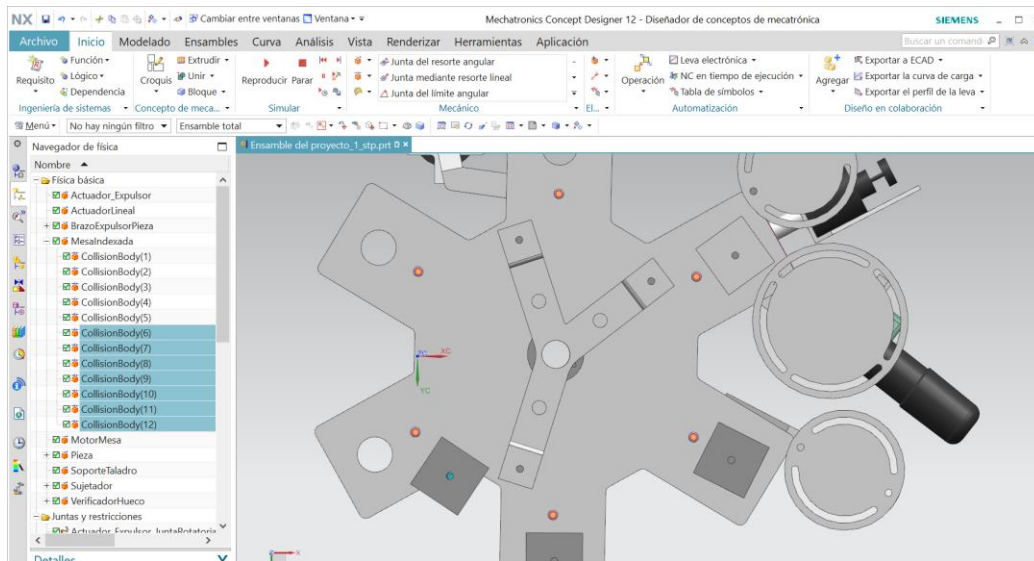


Figura III.71 Cuerpos de colisión tornillos.

2.- Configuración sensor inductivo como sensor de colisión con línea de vista de 6 mm.

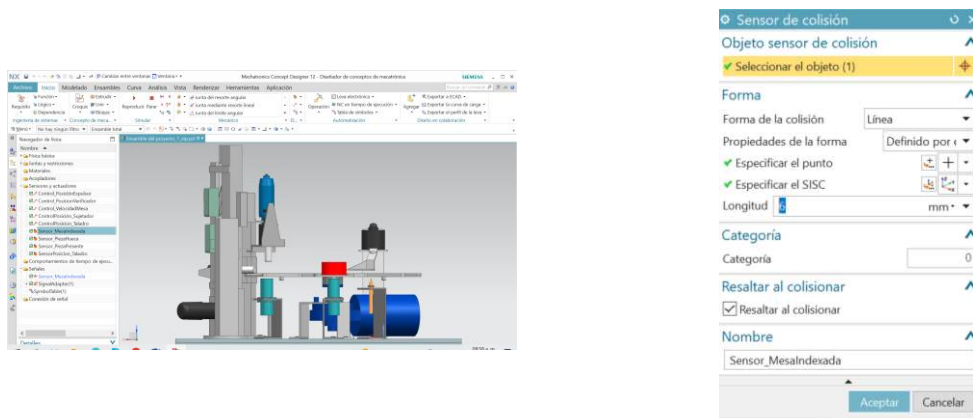


Figura III.72 Sensor de tipo colisión para el que detecta posición mesa cada 60°.

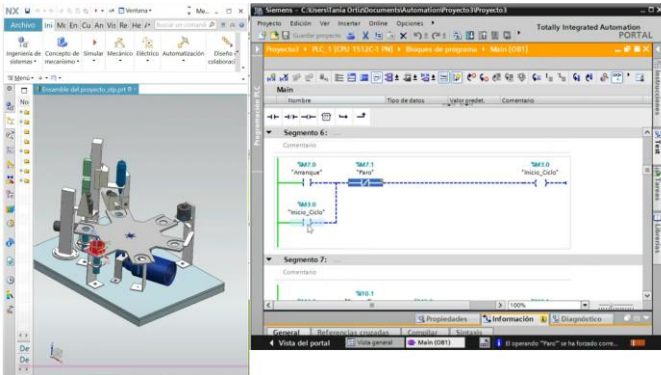
Es importante mencionar que es posible hacer cambios de los nombres en el navegador de física tanto para cuerpos rígidos, juntas, sensores y/o actuadores etcétera y la simulación sigue funcionando conservándose las relaciones de comportamiento definidas.

Bibliografía Capítulo III

Este capítulo tiene como referencias los manuales:

Frank Ebel, Markus Pany, Processing station Manual (2006). Alemania: Festo Didactic

Circuits diagrams System MPS-C, MPS Processing station (2006). Alemania: Festo Didactic



Capítulo IV

CO-SIMULACIÓN CON NX MCD-TIA PORTAL-PLCSIM ADVANCED

La simulación conjunta o co-simulación del sistema electromecánico definido con NX MCD y el control editado en lenguaje de escalera para su comisionamiento en el caso práctico descrito en el capítulo III se le denomina Software-in-the-loop (SIL), el cual es un método de prueba y validación en un entorno completamente virtual para detectar errores de forma rápida y rentable del sistema a automatizar.

Implementar pruebas SIL:

*-Acelera el proceso de
desarrollo*

*-Las pruebas pueden ser
más rápidas que en
tiempo real*

*- La simulación ofrece
flexibilidad y
repetibilidad*

*-Con subprocesos
múltiples, se pueden
realizar múltiples
pruebas al mismo tiempo*

IV.1 Co-Simulación con NX MCD, TIA Portal y PLCSIM Advanced

Se le llama *simulación conjunta* o *co-simulación* cuando diferentes softwares comparten información entre sí y la procesan, lográndose análisis de diversas áreas de la ingeniería en un ambiente totalmente digital, tanto en resultados de los cálculos en datos en conjunto con las visualizaciones del modelo electromecánico en 3D y gráficos de comportamientos principalmente.

Técnicas que permiten hacer mucho más eficientes los procesos de diseño multidisciplinario y concurrente, con la meta de colocar el producto en el mercado más rápido y por ende hacer del negocio cada vez más competitivo.

Los ejemplos descritos en el presente libro son sobre el diseño y validación en conjunto con el control automático, en los cuales se utilizan Controladores Lógicos Programables (PLC), en los ejemplos descritos en los capítulos IV y VI; en éste último además del PLC Virtual, se requieren otros tipos de motores para control de movimiento del Robot de tipo SCARA (servos motores) con sus controles: servo drives, para lo cual se requiere el software SIMIT.

Entre las ventajas de los *comisionamientos virtuales y puestas en marcha* en entornos virtuales, que se tienen principalmente son:

- Permiten detectar a tiempo colisiones de componentes diseñados al ejecutar el control automático.
- Probar el sistema con su control automático previo a su ensamble físico, ya que piezas manufacturadas, componentes eléctricos y/o de control tienen tiempos de entrega que generalmente no son inmediatos.
- Una selección inadecuada de componentes para realizar el trabajo de la máquina o estación de trabajo diseñada, como pueden ser elementos: eléctricos (motores), actuadores lineales y/o rotatorios que emplean aire comprimido o aceite, sensores; puede ser corregido a tiempo.
- Una selección inadecuada en tamaño de memoria de trabajo y del programa del PLC, velocidad de procesamiento, tipos de entradas y salidas, puede ser corregido a tiempo.

En resumen:

- Mejora la calidad de los proyectos de automatización.
- Acelera el tiempo de comercialización.
- Reduce los tiempos de producción.
- Reduce el riesgo de puesta en servicio.
- Evita costos de hardware en entornos de simulación.
- Aumenta la eficiencia en el mantenimiento.
- Entrenamiento para el mantenimiento, diseño y operatividad de la máquina.

La versión utilizada para el ejercicio descrito en el capítulo IV es el PLCSIM Advanced 2.0 SP1 y para el programa del PLC fue el TIA Portal fue la versión 15.1.



Figura IV.1 Softwares empleados para el control.

IV.2 Características generales de PLCSIM Advanced

La versión de *S7-PLCSIM Advanced* es compatible con las siguientes series de CPU SIMATIC:

- S7-1500, S7-1500F, S7-1500C, S7-1500T y S7-1500TF
- ET 200SP y ET 200SP F
- Comunicación: S7-PLCSIM Advanced admite la comunicación a través de Softbus o TCP/IP.
- S7 PLCSIM Advanced permite que los programas C++ y C# y el software de simulación accedan a la serie de CPU SIMATIC compatibles con S7-PLCSIM Advanced a través de la interfaz de usuario (API).
- S7-PLCSIM Advanced y S7-PLCSIM V12 no se pueden instalar en la misma PC o en la misma máquina virtual. La comunicación entre las dos aplicaciones no se puede simular.
- S7-PLCSIM V5.4 SP7 se instala automáticamente con S7-PLCSIM Advanced. La comunicación entre las dos aplicaciones se puede simular.

IV.2.1 S7-PLCSIM V14

La versión de S7-PLCSIM es compatible con las siguientes series de CPU SIMATIC:

- S7-1200 y S7-1200F
- S7-1500, S7-1500F, S7-1500C, S7-1500T y S7-1500TF
- ET 200SP y ET 200SP F
- Para la comunicación: S7-PLCSIM V14 admite la comunicación a través de Softbus.

IV.2.2 S7-PLCSIM V5.x

S7-PLCSIM V5.x simula la siguiente serie de CPU SIMATIC:

- S7-300 y S7-300F
- S7-400 y S7-400F
- S7-PLCSIM V5.x no se puede instalar en la misma PC o en la misma máquina virtual. S7-PLCSIM a partir de V12.
- S7-PLCSIM V5.4 SP7 se instala automáticamente con S7-PLCSIM Advanced.
- Para la comunicación:
 - S7 PLCSIM V5.x puede comunicarse a través de Softbus con instancias de S7-PLCSIM a partir de V12.
 - S7-PLCSIM V5.4 SP7 puede comunicarse a través de Softbus con instancias de S7-PLCSIM Advanced.

Áreas de aplicación:

- Software in the Loop-Simulation para la puesta en marcha virtual de estaciones de trabajo.
- En simulaciones conjuntas con software de terceros:
 - Simulación de máquinas y plantas de producción.
 - Simulación combinada de automatización y mecánica.

IV.3 Requerimientos para la simulación

Para utilizar un proyecto de STEP 7 con simulación, debe seleccionar la opción "Soporte de simulación durante la compilación de bloques" en la ficha "Protección" en las propiedades del proyecto.

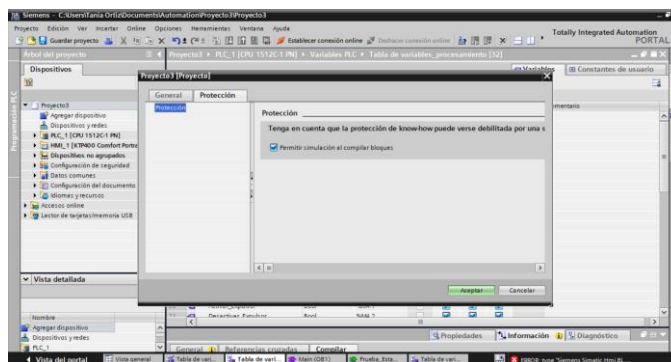


Figura IV. 2 Habilitación del proyecto para ser simulado.

Controladores soportados para ejecutarse virtualmente:

	<i>Familia</i>	<i>Número de parte</i>
Standard CPUs	CPU 1511-1 PN	6ES7511-1AK01-0AB0
	CPU 1513-1 PN	6ES7513-1AL01-0AB0
	CPU 1515-2 PN	6ES7515-2AM01-0AB0
	CPU 1516-3 PN/DP	6ES7516-3AN01-0AB0
	CPU 1517-3 PN/DP ³	6ES7517-3AP00-0AB0
	CPU 1518-4 PN/DP ³	6ES7518-4AP00-0AB0
	CPU 1518-4 PN/DP ODK ^{2,3}	6ES7518-4AP00-3AB0
Fail-safe CPUs	CPU 1511F-1 PN	6ES7511-1FK01-0AB0
	CPU 1513F-1 PN	6ES7513-1FL01-0AB0
	CPU 1515F-2 PN	6ES7515-2FM01-0AB0
	CPU 1516F-3 PN/DP	6ES7516-3FN01-0AB0
	CPU 1517F-3 PN/DP ³	6ES7517-3FP00-0AB0
	CPU 1518F-4 PN/DP ³	6ES7518-4FP00-0AB0
	CPU 1518F-4 PN/DP ODK ^{2,3}	6ES7518-4FP00-3AB0
Compact CPUs¹	CPU 1511C-1 PN	6ES7511-1CK00-0AB0
	CPU 1512C-1 PN	6ES7512-1CK00-0AB0
ET 200SP CPUs	CPU 1510SP-1 PN	6ES7510-1DJ01-0AB0
	CPU 1510SP F-1 PN	6ES7512-1SJ01-0AB0
	CPU 1512SP-1 PN	6ES7512-1DK01-0AB0
	CPU 1512SP F-1 PN	6ES7512-1SK01-0AB0
Technology CPUs	CPU 1511T-1 PN	6ES7511-1TK01-0AB0
	CPU 1515T-2 PN	6ES7515-2TM01-0AB0
	CPU 1517T-3 PN/DP ^{3,4}	6ES7517-3TP00-0AB0
	CPU 1517TF-3 PN/DP ^{3,4}	6ES7517-3UP00-0AB0

1.-No se simula la periferia integrada en las CPU compactas. La interfaz de simulación corresponde a la imagen del proceso.

2.-La funcionalidad ODK de esta CPU no está simulada.

3.-La simulación de esta CPU solo admite recursos de control de movimiento 5120.

4.-La simulación de esta CPU solo admite 64 levas.

Tabla IV.1 Familias de PLC's posibles de virtualizar.

Es importante para una simulación considerar las diferencias de correr un programa en un PLC virtual que uno real:

- S7-PLCSIM Advanced se ejecuta en unA PC con el sistema operativo Windows. Por lo tanto, el tiempo del ciclo de exploración y el tiempo exacto de las acciones en S7-PLCSIM Advanced no son los mismos que cuando estas acciones se ejecutan en el hardware físico.
- Si el programa depende en gran medida del tiempo necesario para ejecutar acciones, es recomendable no evaluar el programa basándose únicamente en los resultados del tiempo de simulación.
- Las instrucciones se simulan con algunas excepciones, por ejemplo, interrupciones. Los programas que se basan en interrupciones se comportan de manera diferente a las CPU reales en la simulación.

- S7-PLCSIM Advanced no simula sistemas de bus (PROFINET IO, PROFIBUS DP, backplane bus).
- S7-PLCSIM Advanced siempre actualiza las áreas de direcciones de la imagen del proceso en el punto de control del ciclo. Las particiones de imagen de proceso entre dos puntos de control de ciclo no se actualizan.
- S7-PLCSIM Advanced no puede simular el rango completo de todas las posibles entradas del búfer de diagnóstico.
- S7-PLCSIM Advanced no admite el uso de una tarjeta de memoria SD.
- S7-PLCSIM Advanced no simula el uso de recetas.

IV.4 PLC Virtual dado de alta para la co-simulación

El PLC indicado en color gris oscuro en la tabla IV.1 es el que se utilizó en el TIA Portal ya que también se cuenta con él físicamente y el cual está siendo instalado en la estación de procesamiento didáctica.

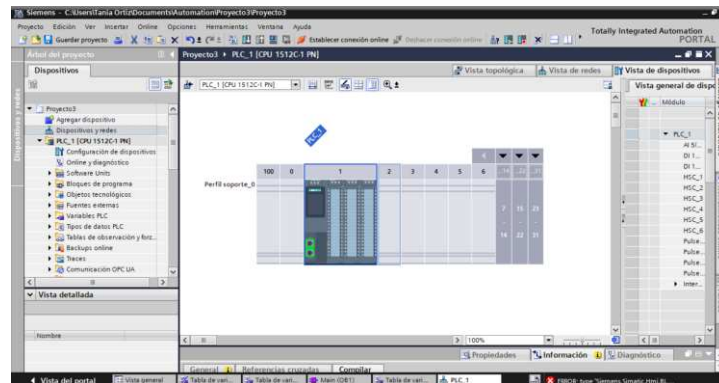


Figura IV.3 CPU 1512C-1 PN, utilizada en la co-simulación con la estación de procesamiento.

Se activaron las marcas de sistema y de ciclo, como se puede observar en la siguiente figura:

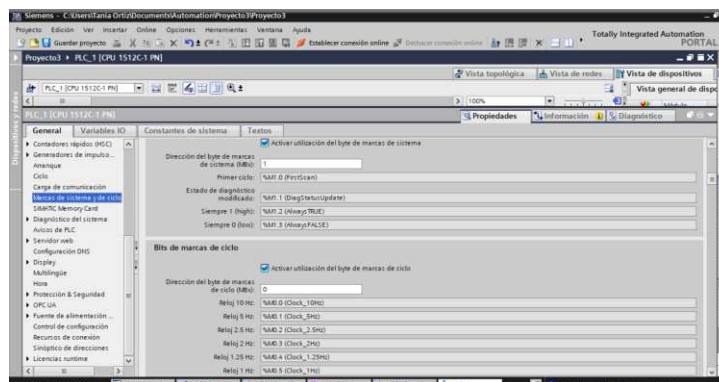


Figura IV.4 Marcas de ciclo y de sistema activadas.

En la siguiente imagen se pueden observar las direcciones de las marcas de sistema y de ciclo, así como las direcciones en formato de palabras (*word*: 8 bits), datos de tipo *Int* (entero) para entradas y salidas digitales.

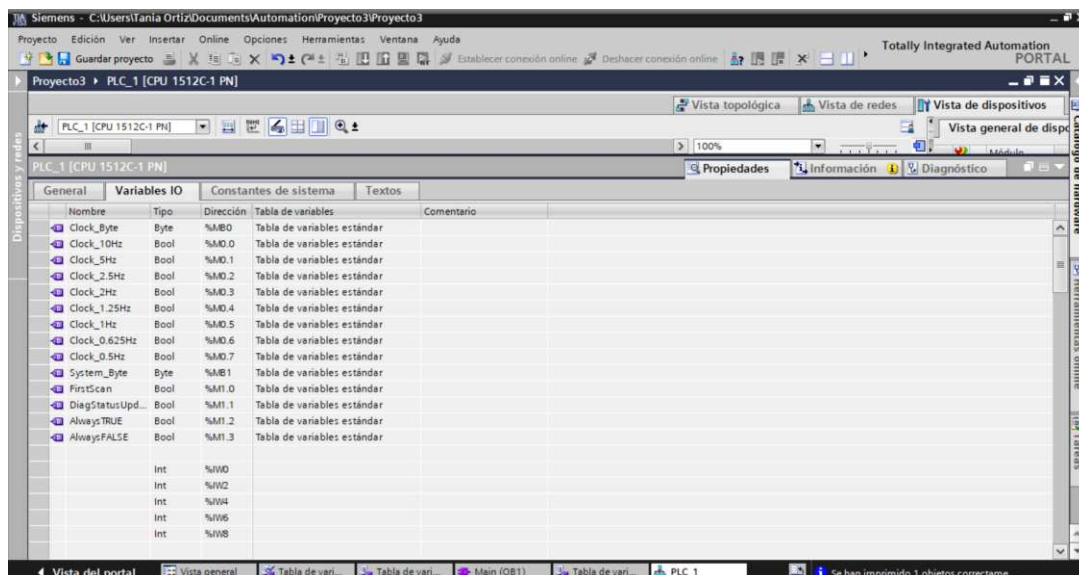


Figura IV.5 Marcas de ciclo y de sistema activadas.

A continuación se indican las direcciones de inicio y de fin para los módulos de E/S digitales y del módulo de E/S Analógicas

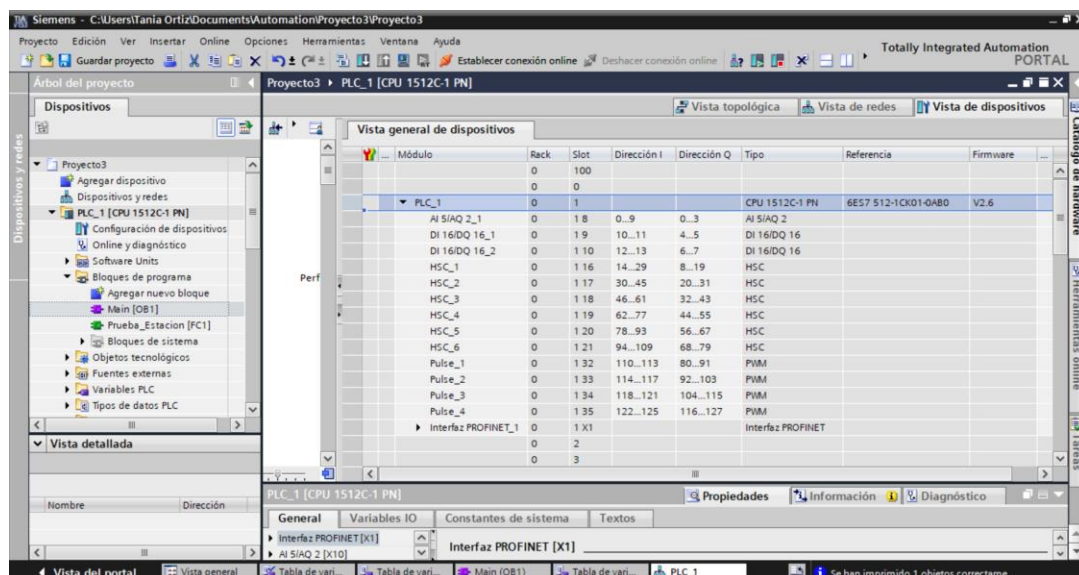


Figura IV.6 Direcciones de E/S.

En la siguiente figura se visualiza la dirección IP asignada al PLC que debe ser la misma que se asigna en la instancia en el PLCSIM Advanced.

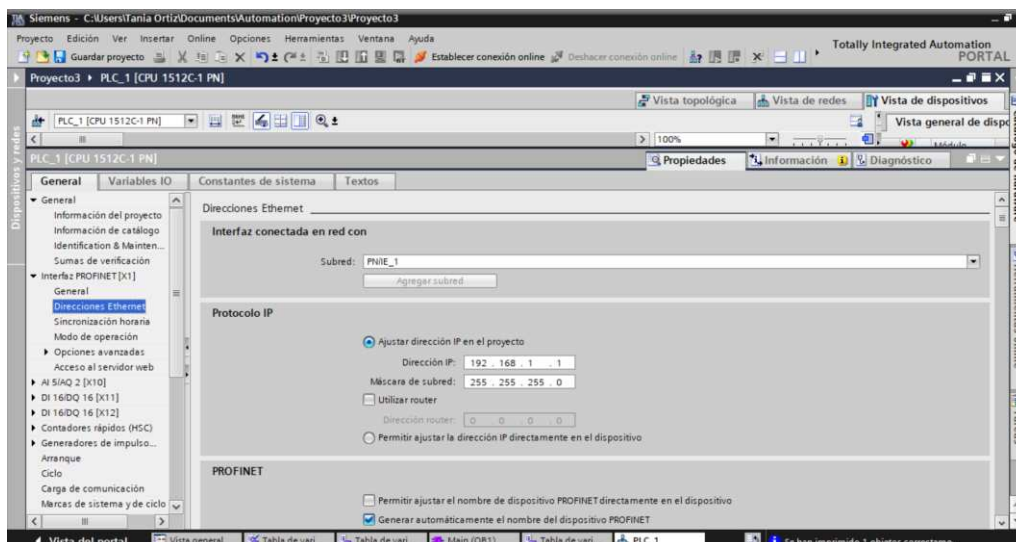


Figura IV.7 Dirección IP del PLC S7-1500.

A continuación se muestra la instancia con el PLC S7-1500 dado de alta en el PLCSIM Advanced con la misma dirección IP del PLC en el TIA Portal (figura IV.7), al observar la imagen se encuentra el PLC virtual en modo *Run* (ejecución); para simulaciones conjuntas se debe elegir el PLCSIM Virtual Ethernet Adapter.

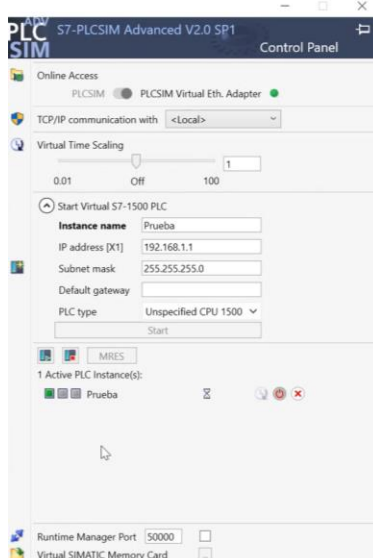


Figura IV.8 Instancia ejecutándose en el PLCSIM Advanced.

Al finalizar el capítulo III se mencionó y se validó que el cambiar el nombre en cuerpos rígidos, articulaciones, sensores, actuadores, controles y/o señales en el navegador de física del *NX MCD*, no se modifican las relaciones de comportamiento en las simulaciones, sin embargo para las señales digitales de entrada (sensores) y de salida (actuadores y/o controles) del PLC deberán conservar el mismo nombre tanto en el programa para el PLC Virtual como en el *NX MCD*.

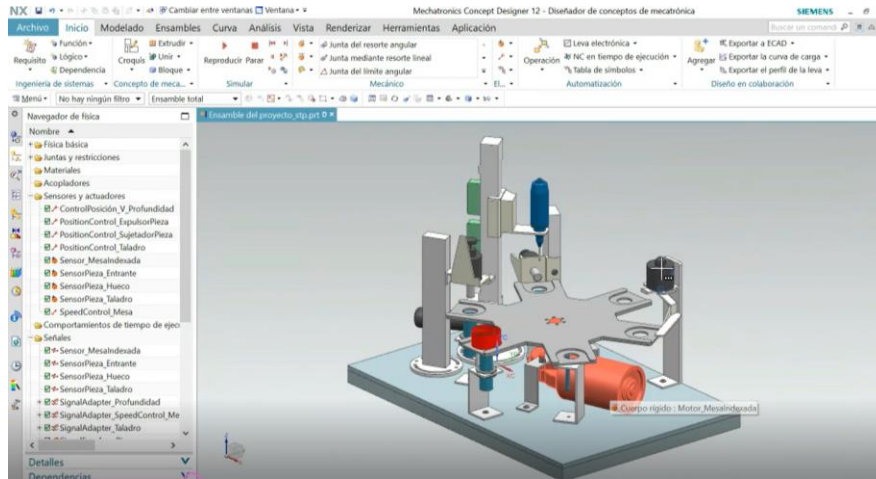


Figura IV.9 Nombres finales para sensores y controles.

IV.5 Configuración para el intercambio de datos entre el PLCSIM Advanced y NX MCD

En este apartado se describe primeramente para los sensores cómo dar de alta las señales con el mismo nombre para las variables de entrada al PLC virtual:

1.- Dar clic en la opción del menú de señales:

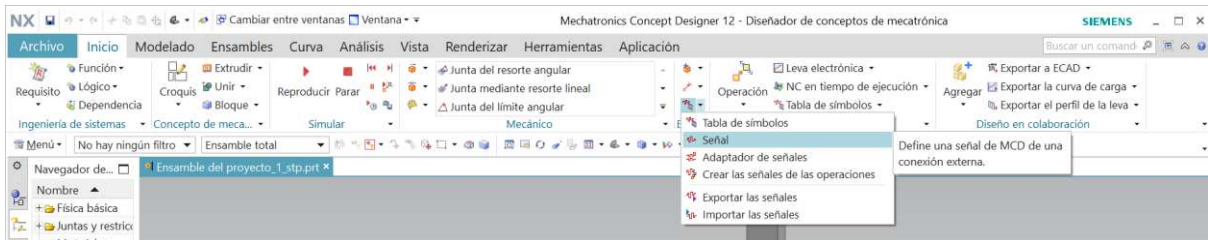


Figura IV.10 Menú para señales.

2.- En la caja de diálogo de señal, dar clic en conectar con el parámetro en tiempo de ejecución, seleccionar en el modelo o en el navegador de física al sensor y en la opción de nombre colocar el mismo del sensor, por ejemplo: Sensor_MesaIndexada, el cual será el nombre para la variable de entrada en el PLC Virtual. Dar clic en aceptar al finalizar.

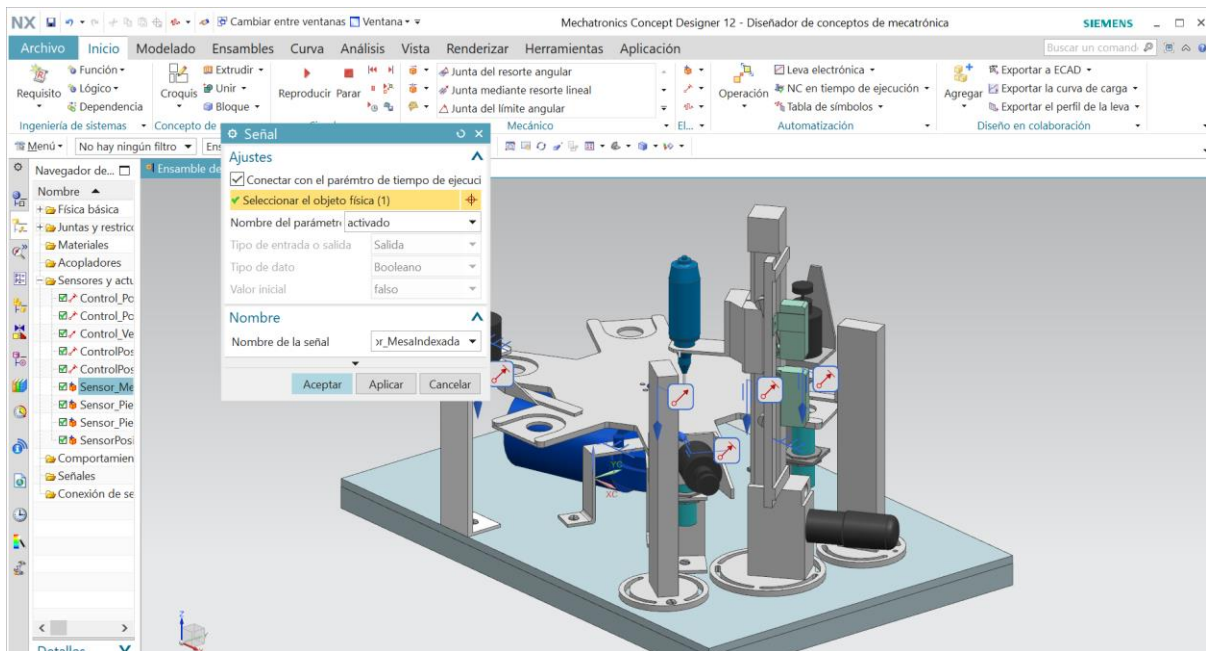


Figura IV.11 Caja de diálogo Señal.

3.- Dar clic en la caja de diálogo que se abre en la opción: Tabla de símbolos nueva.

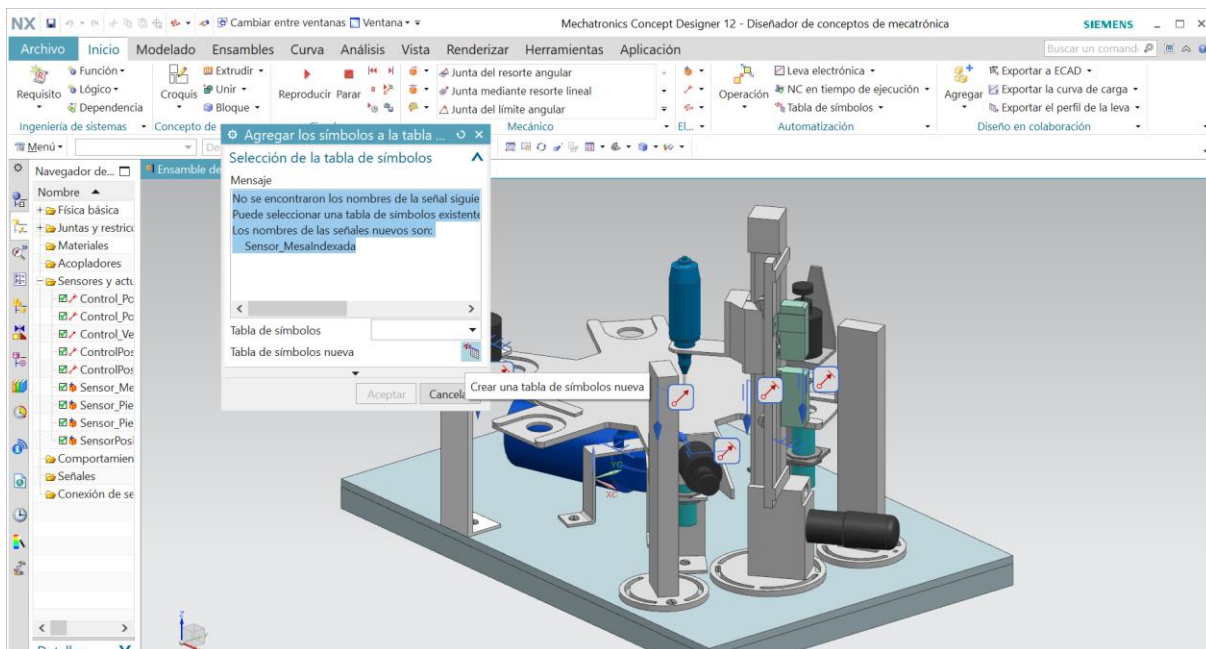


Figura IV.12 Opción para agregar una tabla de símbolos nueva.

4.- Dar clic en aceptar.

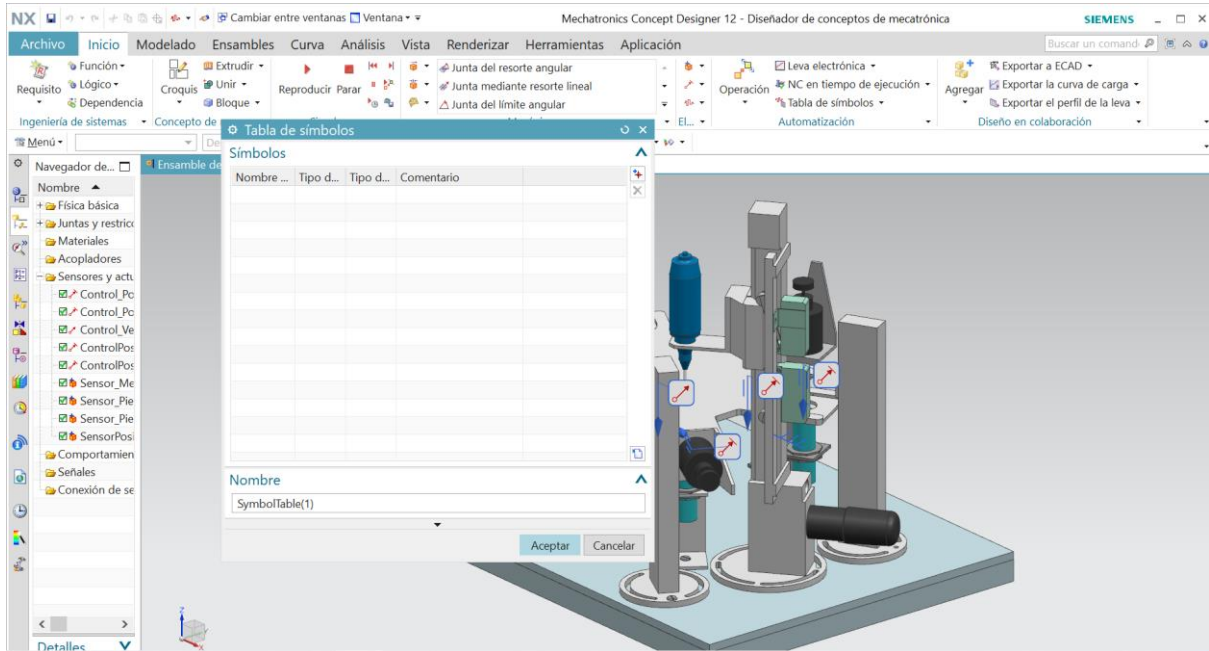


Figura IV.13 Tabla de símbolos nueva.

5.- Dar clic nuevamente en aceptar, aparece en el navegador de física la nueva señal y la nueva tabla de símbolos, a la cual se le puede cambiar también el nombre.

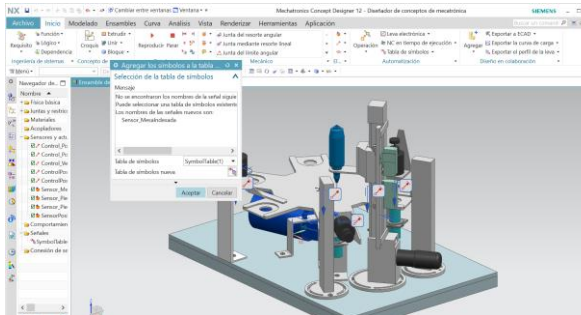


Figura IV.14 Tabla de símbolos creada.

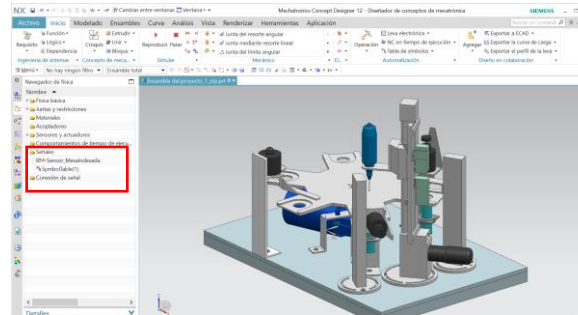


Figura IV.15 Señal y tabla visibles en el navegador de física.

Se realiza los 5 pasos anteriores para el resto de los sensores. Al dar doble clic en tabla (*SymbolTable(1)*) desde el navegador, se visualiza las señales que se van dando de alta. Después hay que dar de alta las señales externas, es decir, las variables de tipo bool para las entradas digitales del PLC virtual correspondientes a los sensores, estas variables deben tener el mismo nombre que tienen en el *NX MCD*, para ellos se debe:

1.- Dar clic en el menú de configuración de señal externa:

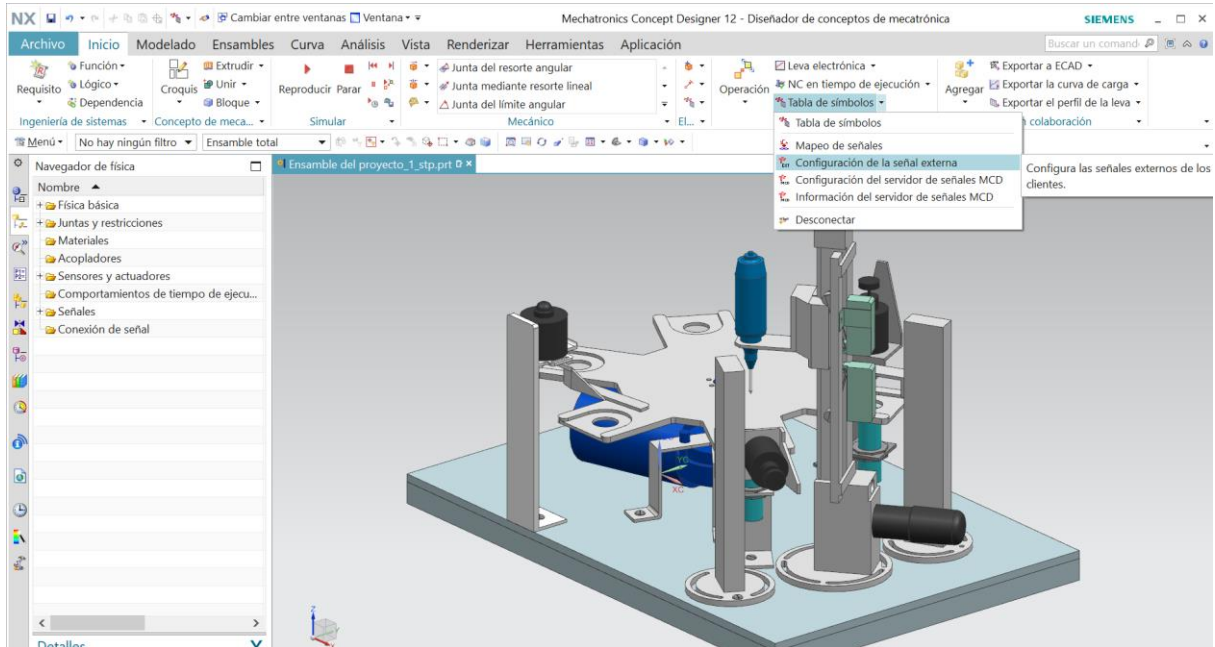


Figura IV.16 Menú configuración señal externa.

2.- En la siguiente caja de diálogo elegir la pestaña de PLCSIM Advanced y dar clic en *Actualizar las copias registradas*.

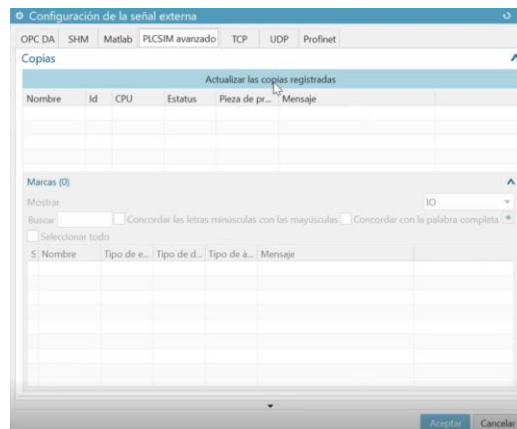


Figura IV.17 Lectura de señales externas.

3.- Aparecen todas las señales y el nombre de la instancia, así como el estado del PLC Virtual dado de alta en el PLCSIM Advanced. La visualización del tipo de señales se puede elegir, como por ejemplo: entre IO (entradas y salidas), M (marcas), IOM (entradas, salidas y marcas), etcétera. Ver figura IV.18.

2.- En la caja de diálogo en la esquina superior derecha elegir PLCSIM Advanced y dar clic en realizar el mapeo automático.

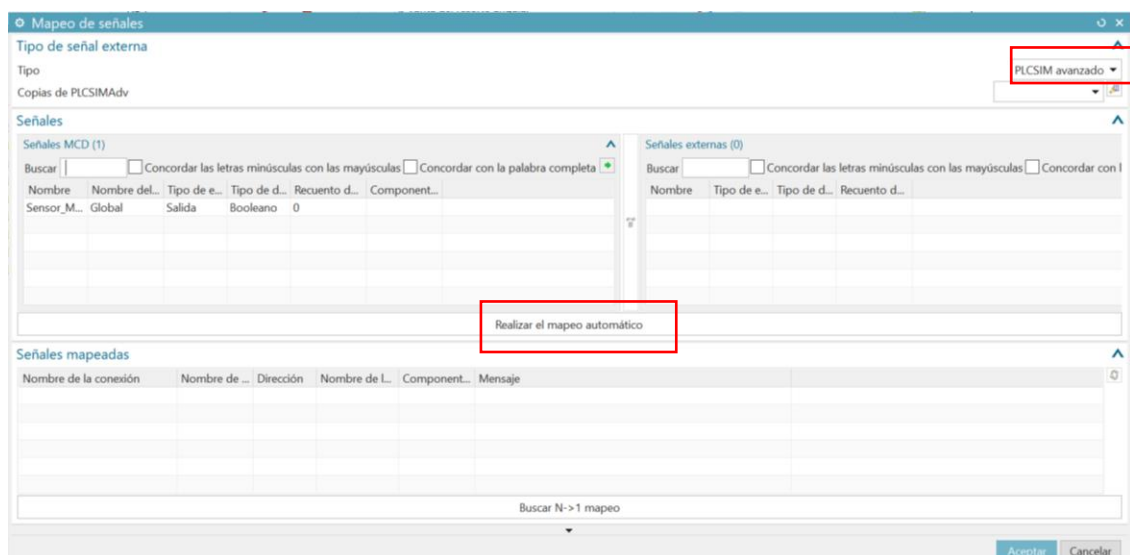


Figura IV.21 Mapeo automático de señales en NX MCD.

3.- Aparecerán todas las señales dadas de alta tanto en *NX MCD* como en el *TIA Portal*, el tipo de dato, si las variables son globales o locales; en el ejemplo las variables serán globales como fueron declaradas en el *TIA Portal* y en el sentido que son leídas, es decir, las señales de los sensores del *NX MCD* van hacia el PLC Virtual

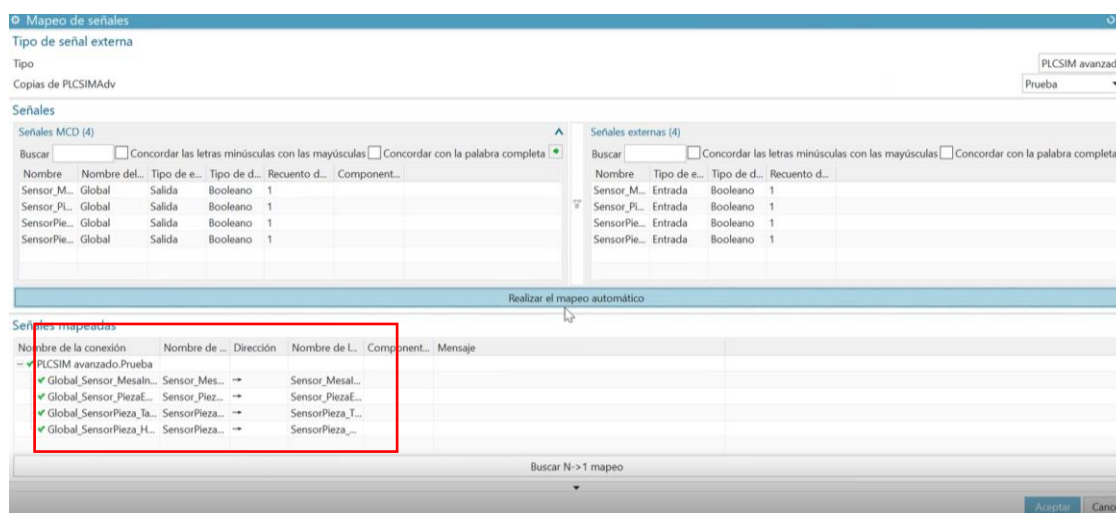


Figura IV.22 Señales mapeadas en NX MCD.

Hasta este momento podemos validar si se realiza la co-simulación correctamente con sólo tener mapeadas las señales de los sensores, por lo hay que:

1.- Compilar el proyecto desde la carpeta del PLC en el TIA Portal.

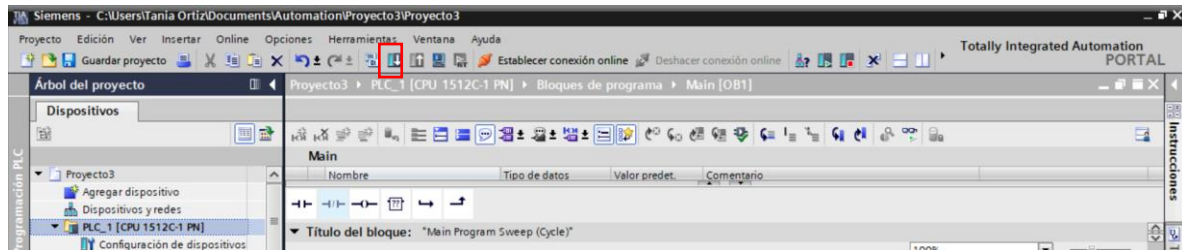


Figura IV.23 Compilación hardware y programa.

2.- Descargar el programa al PLC Virtual, con la instancia *Prueba* abierta en el PLCSIM Advanced.

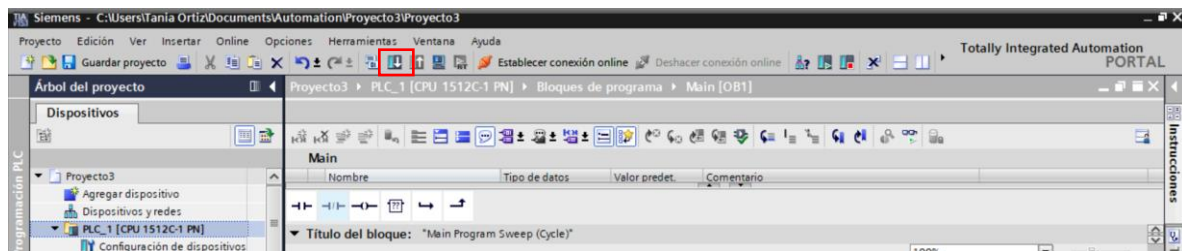


Figura IV.24 Descarga al PLC Virtual (instancia con nombre Prueba).

3.- Arrancar los módulos.

4.- Establecer conexión en línea.

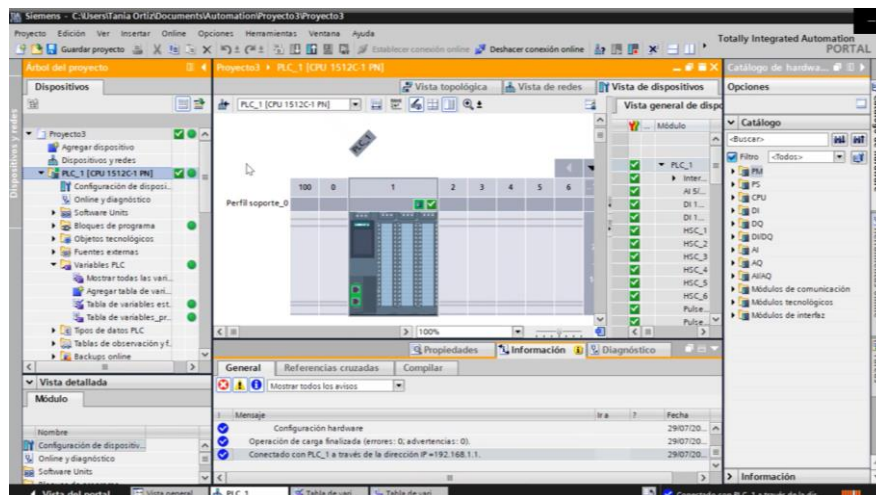


Figura IV.25 Conexión en línea exitosa.

5.- En el *NX MCD* se corre la simulación y el control de velocidad de la mesa hace que rote y tanto el sensor inductivo como los capacitivos conmutan de estado lógico, ya que están detectando, lo cual es observable en el TIA Portal en la sección de monitoreo de la tabla de variables, donde se puede ver el cambio de los estados de verdadero (1 lógico) y falso (0 lógico) de las señales mapeadas correspondientes a los sensores. Y así es como se valida la correcta simulación en conjunto entre los softwares.

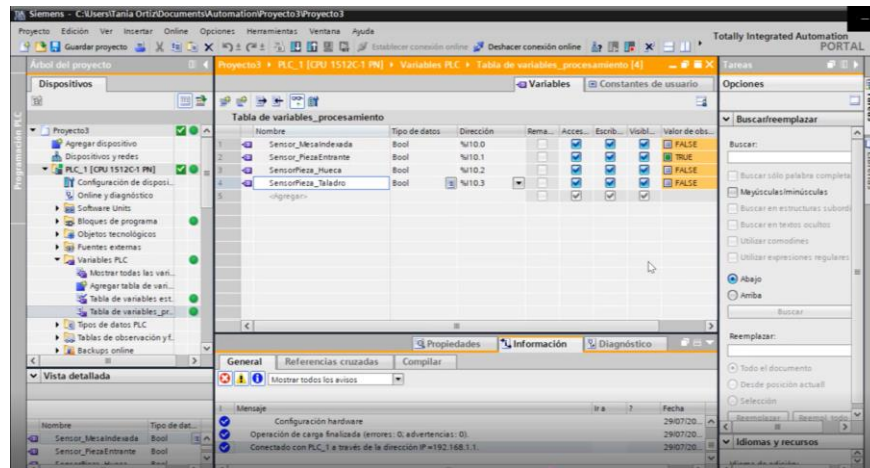


Figura IV.26 Monitoreo del estado lógico de los sensores, señales provenientes del *NX MCD*.

Ya que los tipos de datos para el manejo de los controles para los actuadores en el *NX MCD* son de tipo real, es necesario crear adaptadores de señales para que puedan recibir desde el TIA Portal un dato de tipo entero y no haya conflicto por el formato distinto del tipo de datos; recordar que el tipo de dato de tipo entero ocupa menos espacio en memoria, a continuación se describen de manera breve los pasos a seguir:

1.- Elegir en el menú la opción de adaptador de señales.

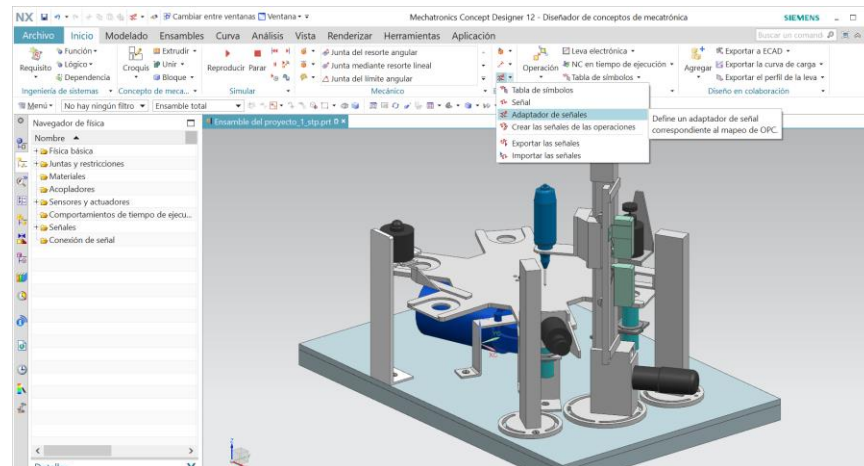


Figura IV.27 Menú adaptador de señales.

2.- En el navegador de física o en el modelo 3D se selecciona un control, como por ejemplo el de posición del sujetador del taladro, después cambiar en la opción del nombre del parámetro a *posición*.

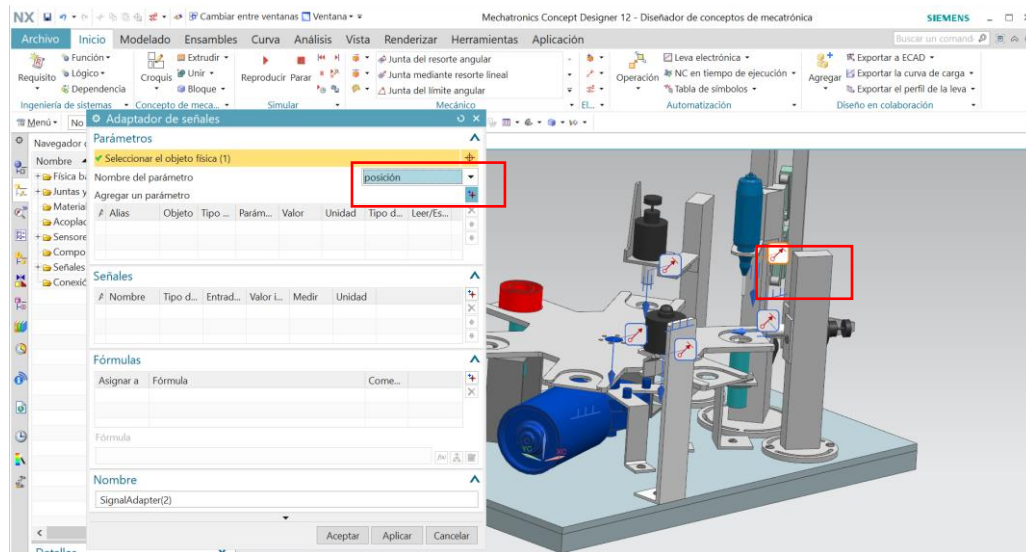


Figura IV.28 Caja de diálogo del adaptador de señales.

3.- Dar clic en agregar un parámetro como se observa en la siguiente imagen:

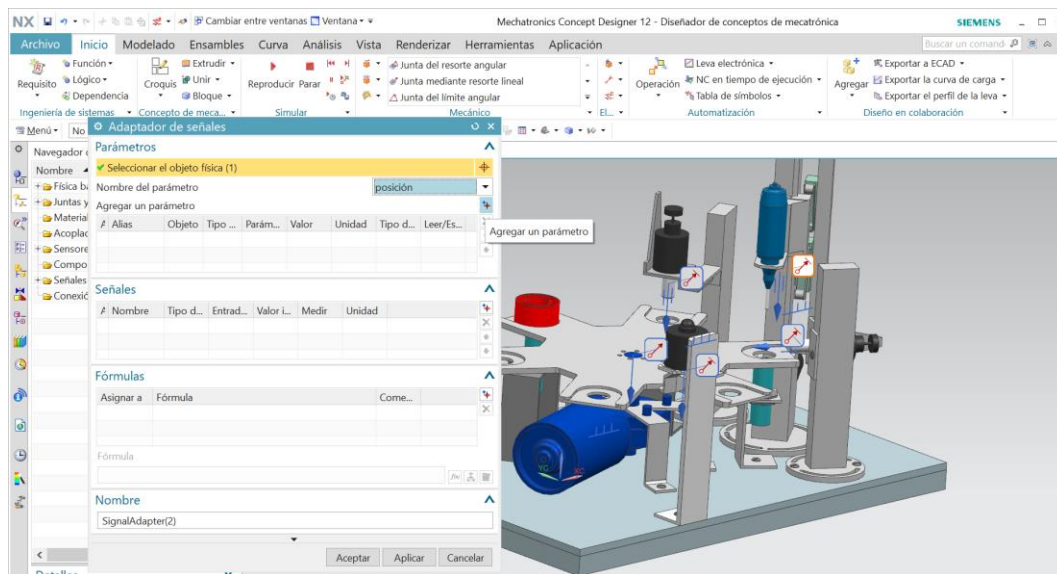


Figura IV.29 Icono de la caja de diálogo para agregar parámetro.

4.- Aparece el parámetro correspondiente al control de posición elegido y en la opción de valor será cambiado después en el mismo control por el valor de cero, se procederá de igual manera para el resto de los controles de posición (colocar 0 en la opción de destino), al igual que para el control de velocidad de la mesa indexada (colocar 0 en la opción de velocidad). Se observa que el dato es de tipo doble por ser real.

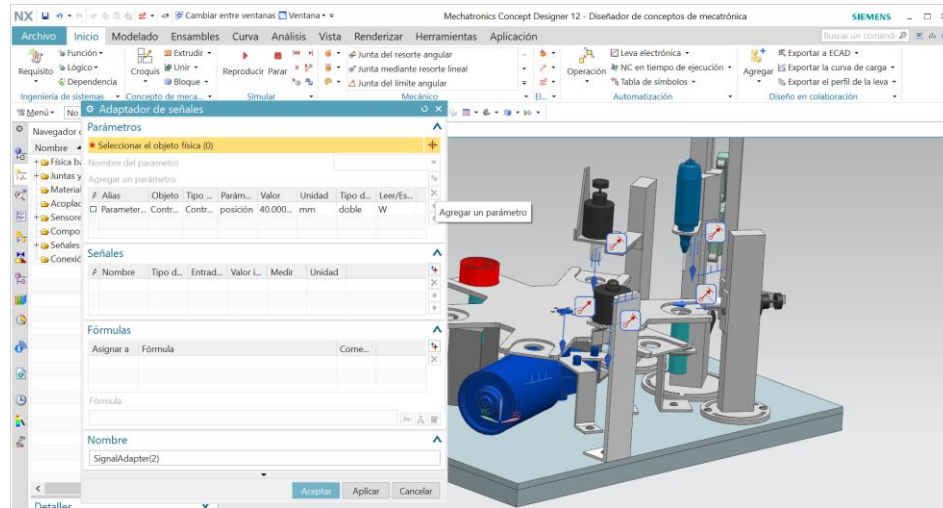


Figura IV.30 Parámetro del ControlPosición_Taladro.

5.- Elegir el parámetro (paloma pequeña de color verde) y dar clic en el icono de agregar una señal como se observa en la siguiente imagen, aparece en la sección de *Fórmulas* el Parámetro.

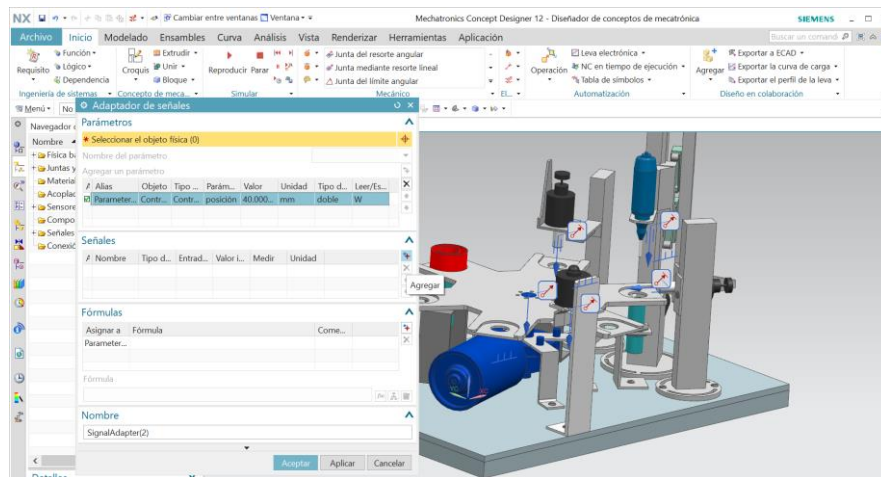


Figura IV.31 Icono para agregar señal.

6.- Cambiar el nombre de la señal con el mismo utilizado en el control del *NX MCD*, es decir *ControlPosicion_Taladro*, cambiar también a un tipo de dato entero, observando que es una señal de entrada con valor inicial de 0.

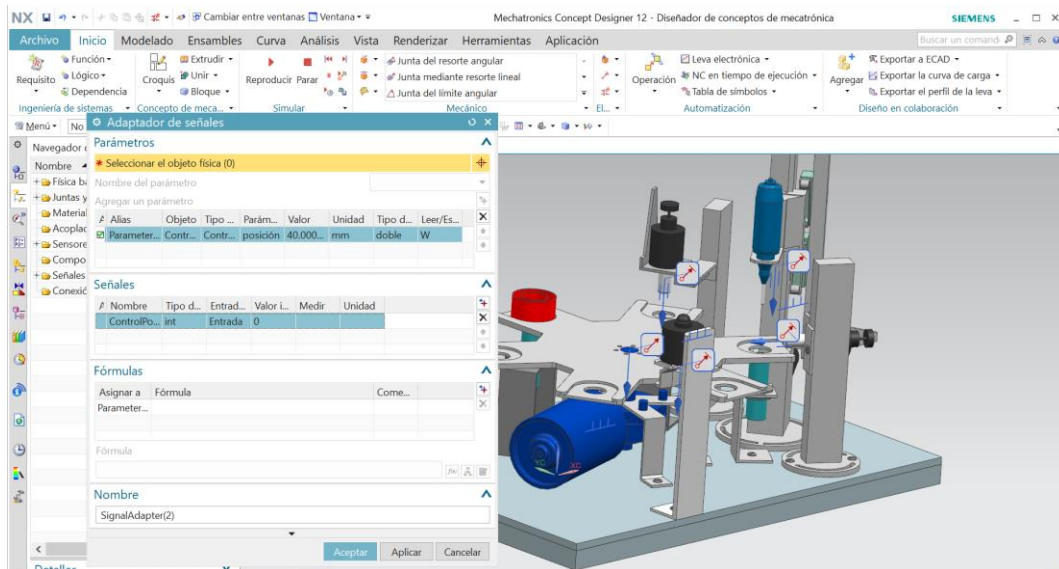


Figura IV.32 Señal agregada y modificada a lo requerido.

6.- Dejar de seleccionar el parámetro y el control, manteniendo la paloma pequeña de color verde en el parámetro.

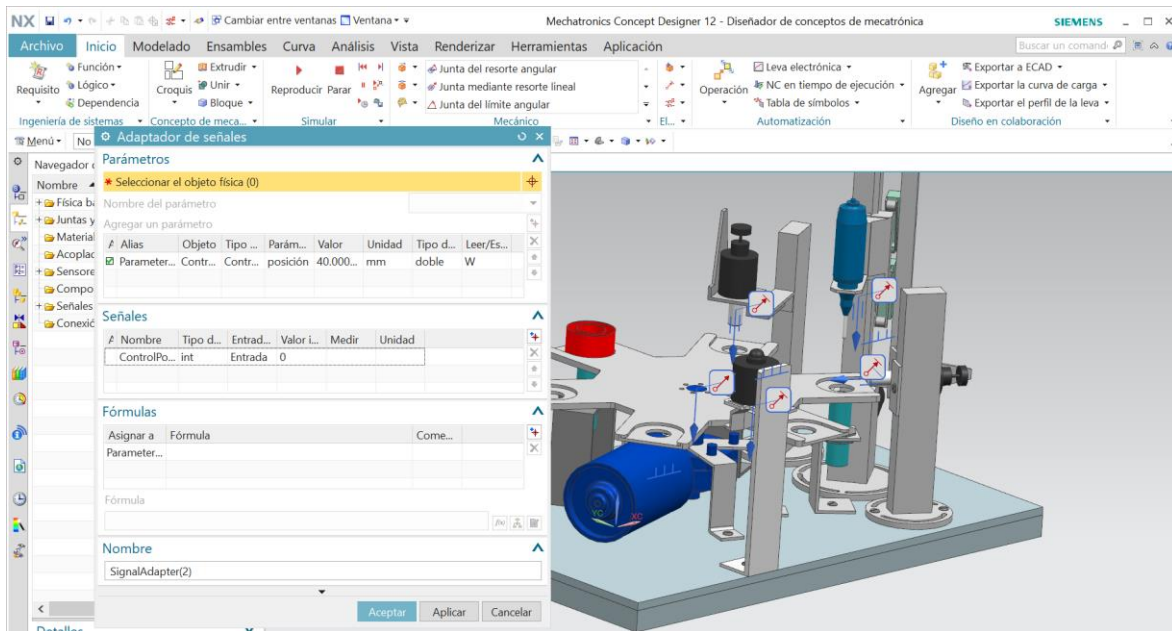


Figura IV.33 Caja de diálogo Adaptador de señales.

7.- Seleccionar en la sección de *Fórmulas* al *Parameter_1*.

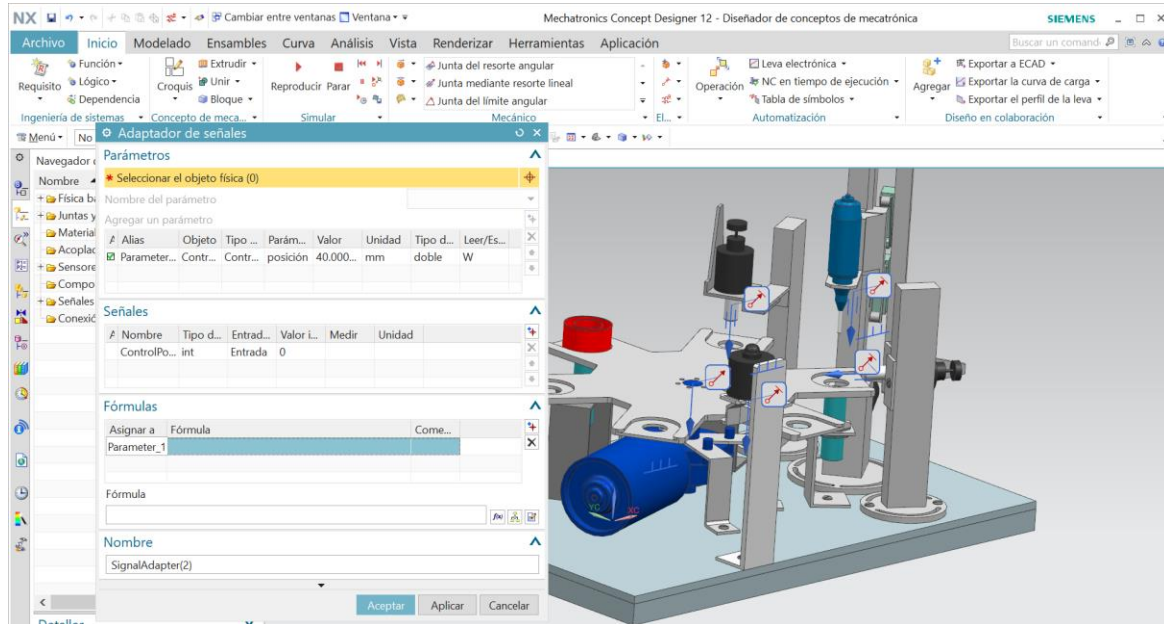


Figura IV.34 Activación ingreso de Fórmula.

8.- Se activa la opción de ingresar Fórmula, escribir una C para localizar el Control de posición elegido en la opción de *Seleccionar el objeto física*.

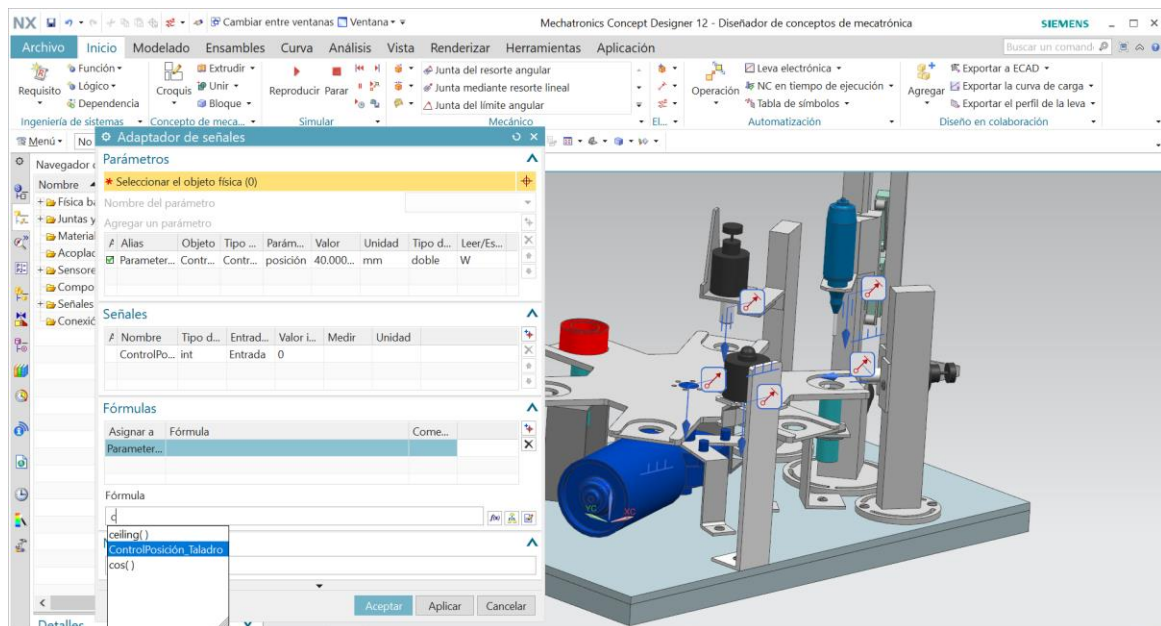


Figura IV.35 Selección del parámetro en Fórmula.

9.- Elegir el *ControlPosicion_Taladro*.

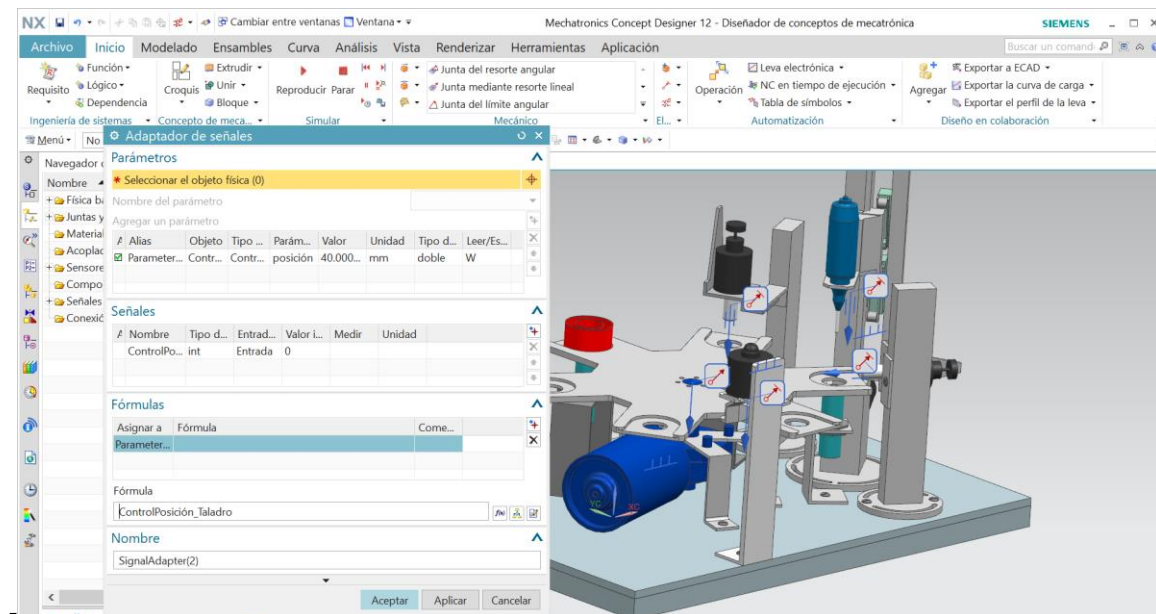


Figura IV.36 Ingreso del parámetro ControlPosición Taladro en Fórmula.

10.- Dar *enter*, no es necesario dar un nombre al adaptador, es conveniente pero no forzoso. Al dar clic en aceptar aparece la tabla de símbolos ya creada.

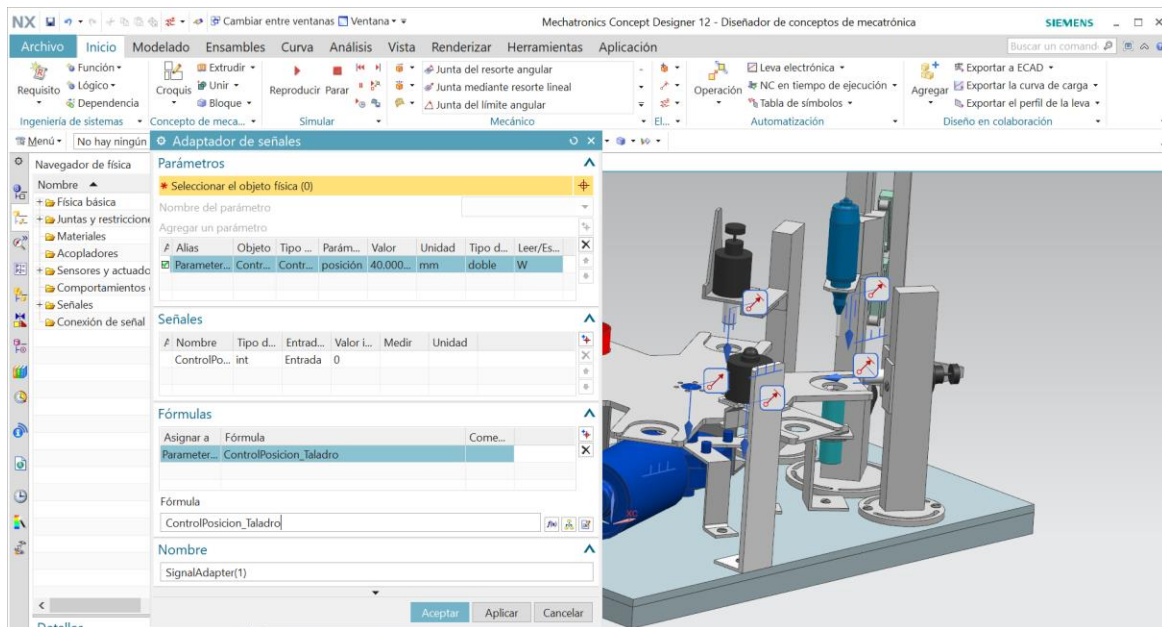


Figura IV.37 Parámetros y Fórmula completos.

11.- Dar clic en aceptar nuevamente, se trabaja en la misma tabla creada para las señales de los sensores.

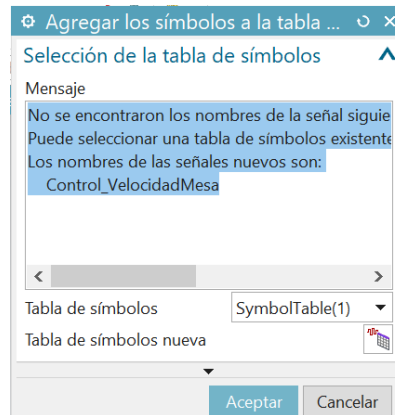


Figura IV.38 Tabla de símbolos (1) creada previamente.

Se ha creado en el navegador de física el adaptador de señal, ver figura siguiente, se trabaja con el nombre del control y no con el nombre del adaptador, cuando se intercambia la información en la simulación conjunta.

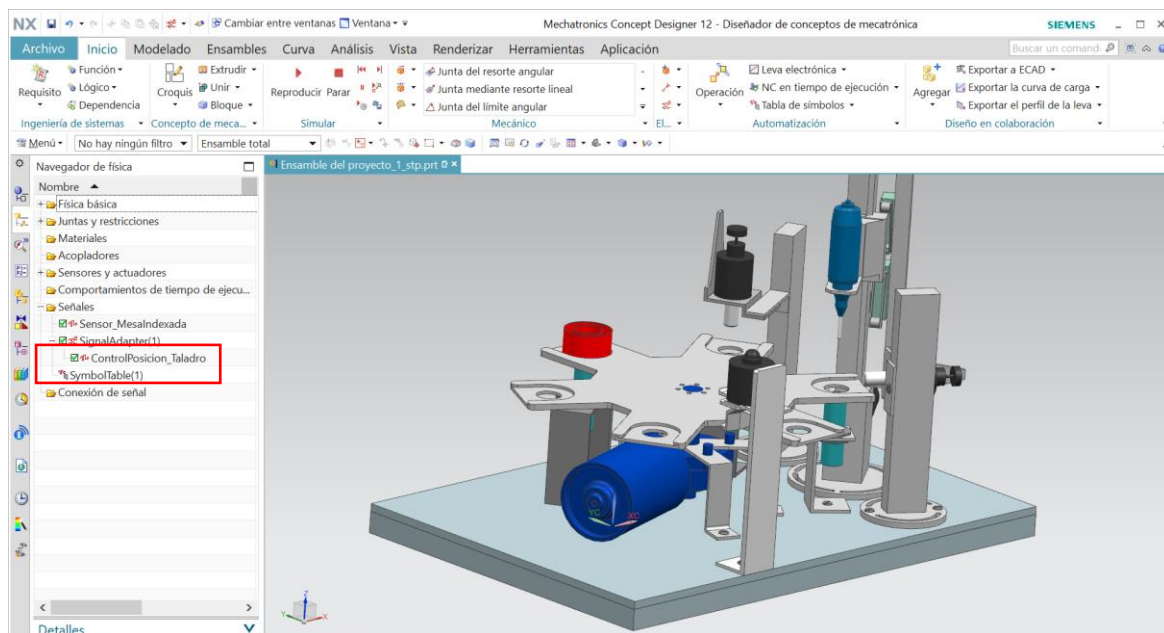


Figura IV.39 Adaptador creado para ControlPosicion_Taladro.

Se realiza lo mismo para cada control, del paso 1 al 11, finalizando se crean las variables con los mismos nombres en el PLC como marcas de tipo entero, se procede en seguida a compilar, cargar el programa en la instancia *Prueba* y se establece conexión en línea. Después se actualizan las señales mediante el menú de *configuración de la señal externa*, en su caja de diálogo, dar clic en el botón de *actualizar las copias registradas*, en la

ventana se cambia para poder visualizar a IOM, es decir, entradas/salidas y marcas, luego se busca y se selecciona la señal de *ControlPosicion_Taladro*, creada en el PLC y leída en *NX MCD* al *actualizar las copias registradas*, se cierra dicha caja de diálogo con tan solo dar clic en aceptar.

Posteriormente, es necesario mapear las señales creadas de todos los controles, en el menú correspondiente de *mapeo*, en su caja de diálogo se debe dar clic en realizar *mapeo automático*, se exhorta a observar las direcciones de las flechas de las señales mapeadas, con el fin de identificar que señales van al PLC y cuáles van al *NX MCD*; finalizar dando clic en aceptar y se cierra la ventana.

En estos últimos párrafos se explican los mismos pasos realizados para las señales de los sensores, pero no a detalle, ya que fue descrito anteriormente para los sensores.

Se menciona nuevamente que para la simulación conjunta, con respeto a las direcciones para los sensores se emplearon las entradas digitales del PLC, y para las direcciones de los actuadores se trabajaron los controles como marcas de tipo entero, en la tabla IV.2 se muestran las configuraciones definidas en el *NX MCD*, con restricciones de límites de movimiento a velocidades determinadas.

<i>Controles</i>	<i>Actuador</i>	<i>Juntas</i>	<i>Límites máximos juntas</i>
Posición	Verificador Profundidad	Deslizante	25 mm a 10 mm/s
Posición	Sujetador	Deslizante	20 mm a 5 mm/s
Posición	Expulsor	Deslizante	95° a 20°/s, girando sentido anti horario
Posición	Sujetador taladro	Deslizante	No tiene
Velocidad	Mesa Indexada	Charnela (rotatoria)	No tiene

Tabla IV.2 Configuración controles en NX MCD.

El tipo de dato en las restricciones y/o límites en la configuración de los controles en *NX MCD* son de tipo real, por lo que en el PLC deberían utilizarse las salidas analógicas; el tamaño de memoria ocupado en el PLC para datos de tipo real es mayor que el ocupado por datos de tipo entero; por lo tanto es mejor utilizar este último tipo de dato, para lo cual se utilizó la herramienta de *adaptador de señal* en el *NX MCD*, así podrá el PLC enviar al control del actuador un dato de tipo entero, configurándose para cada control mencionado en la tabla IV.2, un adaptador de señal para que pueda ser interpretado en el formato de tipo real que utiliza el *NX MCD* en los 4 controles de posición y un control de velocidad.

IV.6 Programa en escalera para la co-simulación

Es posible en este nivel, realizar un programa de prueba sólo para el ajuste y validación del buen funcionamiento de sensores y actuadores en el sistema electromecánico simulado en conjunto, con tan sólo utilizar las entradas digitales del PLC y marcas para los datos de tipo entero de los adaptadores de señal para los controles definidos para cada actuador en el *NX MCD*, es importante que los sensores tengan el mismo nombre en las variables del programa del PLC; así como también para los controles aunque sea mediante los adaptadores de señal. En la siguiente tabla IV.3, se presentan las variables y su tipo de dato, las cuales fueron definidas en el programa *Proyecto3* para simular en la instancia *Prueba* del *PLCSIM Advanced*, donde se emula el comportamiento del *PLC S7-1512C*.

Tabla de variables_procesamiento [32]

Variables PLC

Variable PLC	Nombre	Tipo de datos	Dirección	Remanencia	Accesible desde SIMATIC UA	Escribible desde SIMATIC UA	Visible en Supervisión	Comentario
10	Sensor_MonitoreoActa	Bool	N/I0.0	Falso	True	True	True	
11	SensorPasea_Embente	Bool	N/I0.1	Falso	True	True	True	
12	SensorPasea_Huico	Bool	N/I0.2	Falso	True	True	True	
13	SensorPasea_Taladro	Bool	N/I0.3	Falso	True	True	True	
14	SpeedControl_Mesa	Int	NAM10	Falso	True	True	True	
15	Paso1	Bool	NAM2.1	Falso	True	True	True	
16	Paso2	Bool	NAM2.2	Falso	True	True	True	
17	ControlPosicion_V_Profundidad	Int	NAM11	Falso	True	True	True	
18	ControlPosicion_Taladro	Int	NAM16	Falso	True	True	True	
19	Paso	Bool	NAM5.1	Falso	True	True	True	
20	Arreglo8	Bool	NAM5.0	Falso	True	True	True	
21	Inicio_Ciclo	Bool	NAM4.0	Falso	True	True	True	
22	Fin_Ciclo	Bool	NAM4.6	Falso	True	True	True	
23	Paso4	Bool	NAM2.4	Falso	True	True	True	
24	Paso3	Bool	NAM2.3	Falso	True	True	True	
25	Paso5	Bool	NAM2.5	Falso	True	True	True	
26	Paso6	Bool	NAM2.6	Falso	True	True	True	
27	ControlCajacion	Int	NAM19	Falso	True	True	True	
28	Expositor_Paso	Int	NAM23	Falso	True	True	True	
29	Activar_Expositor	Bool	NAM4.1	Falso	True	True	True	
30	Desactivar_Expositor	Bool	NAM4.2	Falso	True	True	True	
31	Reveres	Bool	NAM4.3	Falso	True	True	True	
32	Inicio_Prueba	Bool	NAM4.4	Falso	True	True	True	
33	Fin_Prueba	Bool	NAM4.5	Falso	True	True	True	
34	Paso7	Bool	NAM2.7	Falso	True	True	True	
35	Paso8	Bool	NAM3.0	Falso	True	True	True	
36	Paso9	Bool	NAM3.1	Falso	True	True	True	
37	Paso10	Bool	NAM3.2	Falso	True	True	True	
38	Paso11	Bool	NAM3.3	Falso	True	True	True	
39	Paso12	Bool	NAM3.4	Falso	True	True	True	
40	Paso13	Bool	NAM3.5	Falso	True	True	True	
41	Paso14	Bool	NAM3.6	Falso	True	True	True	

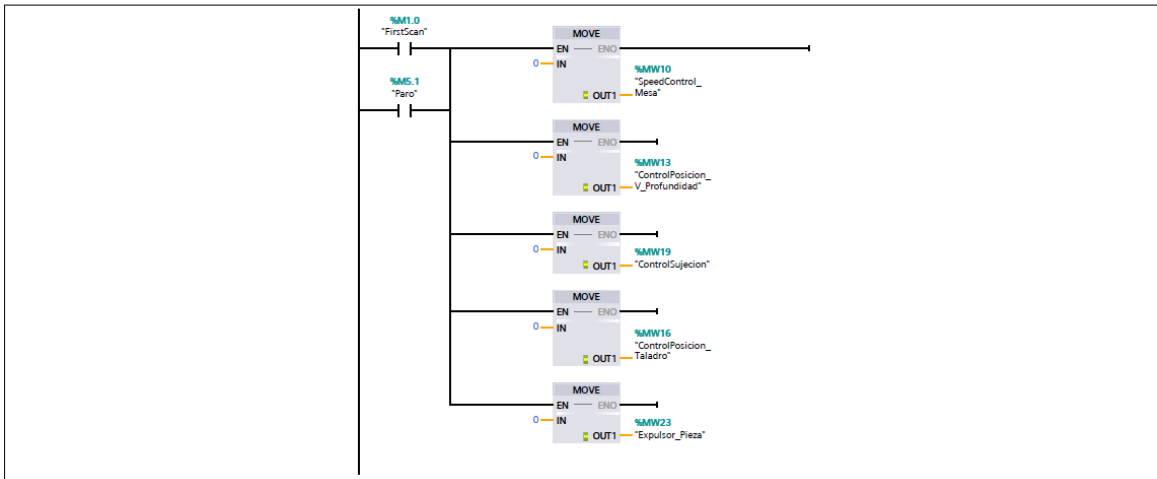
Tabla IV.3 Variables del programa *Proyecto3*, en lenguaje en escalera.

El método empleado es el de paso a paso, donde se activa el paso siguiente con las condiciones presentes generalmente sensores y marcas que memorizan algún evento, como sería la ejecución de un paso anterior, se memoriza también el paso activo en las líneas mediante enclavamientos tipo eléctrico con el paso a ser ejecutado y se desactiva en la misma línea con la activación de la marca y/o paso posterior o siguiente.

Se realizó el programa en lenguaje de escalera, en el objeto lógico principal (OB1), en los primeros segmentos se tienen las condiciones y en los últimos se han colocado las acciones a realizar en cada paso, las cuales hacen referencia al sistema descrito en el capítulo III, dichas acciones activan a los actuadores digitales que operan a 24 Vcd mediante los adaptadores de señal de los 4 controles de posición y un control de velocidad (mesa indexada) que accionan a los actuadores, enviando datos de tipo entero al *NX MCD* a través de marcas. A continuación se describe cada línea del programa.

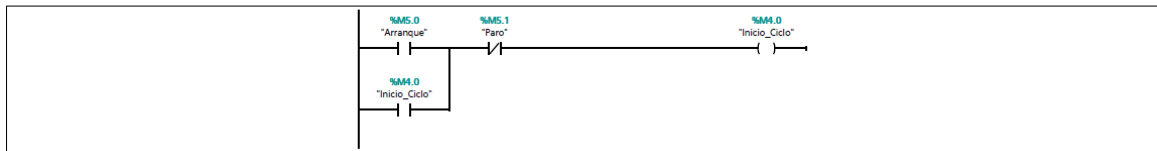
Segmento 1:

En el paro o en el primer paso del escaneo del programa (FirstScan), limpiar las variables de tipo entero, es decir, mover un 0 a la localidad de memoria para el control de velocidad de la masa indexada, al control de posición actuador que valida profundidad y pieza colocada correctamente, al control de sujeción y al actuador que expulsa la pieza fuera de la estación de procesamiento.



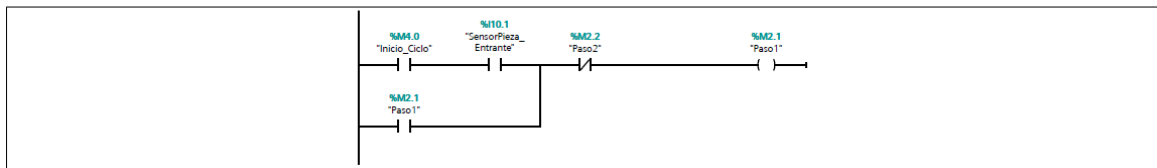
Segmento 2:

Se fuerza el bit de arranque o se activa a través del HMI y si no se activa el paro, entonces se inicia ciclo y se memoriza.



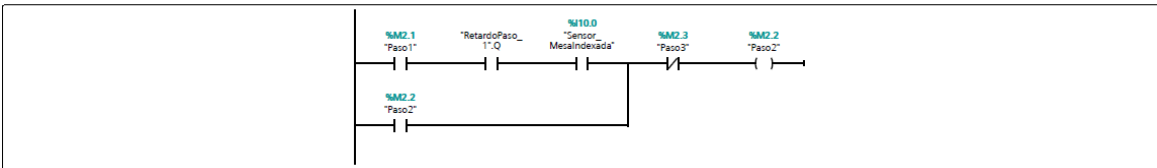
Segmento 3:

Iniciado el ciclo y si se detecta pieza en la estación entonces activamos Paso1 y se memoriza el paso.

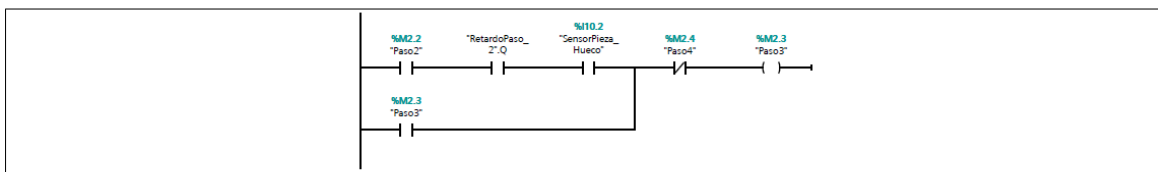


Segmento 4:

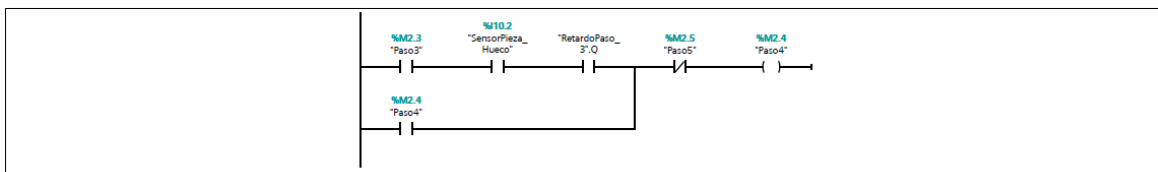
Una vez realizado el Paso1 en un tiempo determinado dado por el timer TON: RetardoPaso_1 y si el Sensor_MesaIndexada detecta posición al rotar mesa 60 Grados, se activa el bit Paso2 y se memoriza el paso.



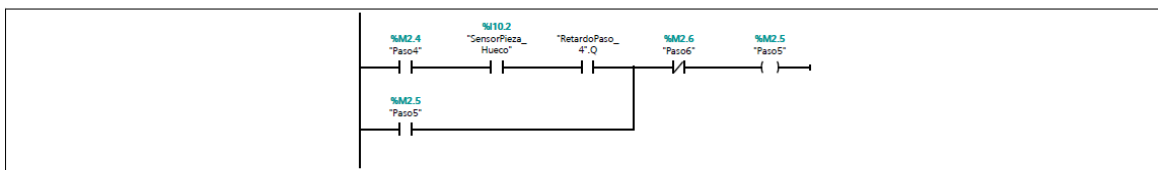
Una vez realizado el Paso2 en un tiempo determinado dado por el timer TON: RetardoPaso_2 y si el SensorPieza_Hueco detecta pieza, entonces se activa Paso3. y se memoriza el paso.



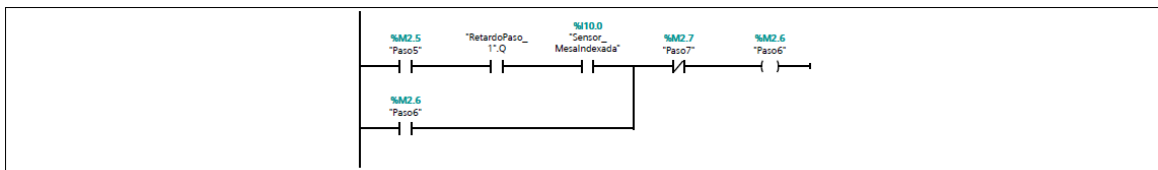
Una vez realizado el Paso3 en un tiempo determinado dado por el timer TON: RetardoPaso_3 (tiempo para validar pieza en posición correcta para ser barrenada) y si el SensorPieza_Hueco detecta pieza, entonces se activa Paso4 y se memoriza.



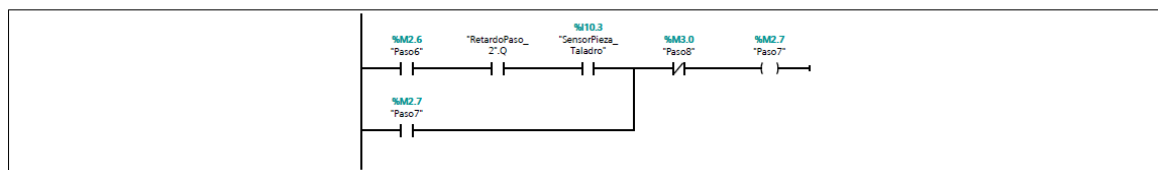
Una vez realizado el Paso4 en un tiempo determinado dado por el timer TON: RetardoPaso_4 (retardo para asegurar el regreso del actuador) y si el SensorPieza_Hueco detecta pieza, entonces se activa Paso5 y se memoriza el paso.



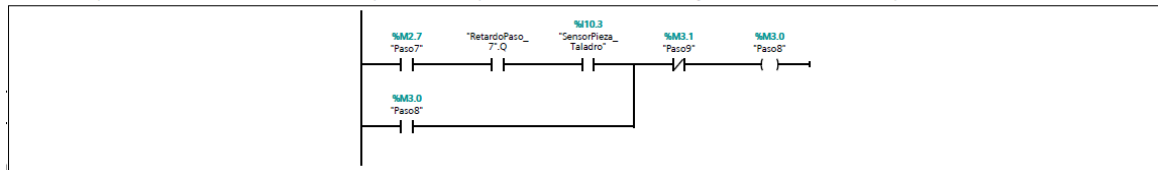
Una vez realizado el Paso5 en un tiempo determinado dado por el timer TON: RetardoPaso_1, tiempo asignado para el control de giro de la mesa, y si el Sensor_Mesalndexada detecta posición, entonces se activa Paso6 y se memoriza el paso.



Una vez realizado el Paso6 en un tiempo determinado dado por el timer TON: RetardoPaso_2, tiempo asignado para asegurar que una vez que la mesa giró 60 grados pare, entonces si el sensor correspondiente: SensorPieza_Taladro detecta presencia de la pieza, entonces se activa Paso7 y su retardo. Se memoriza el paso.

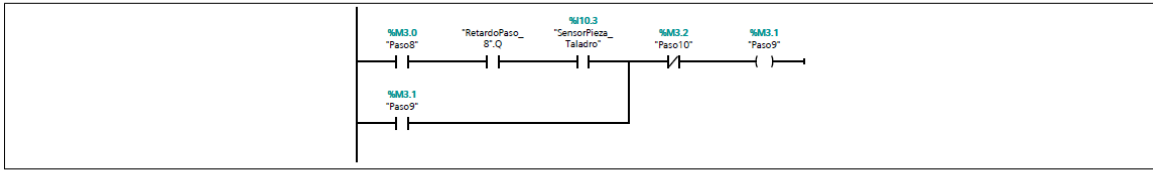


Una vez realizado el Paso 7 en un tiempo determinado dado por el timer TON: RetardoPaso_7, tiempo asignado para asegurar que la pieza es sujeta, entonces si el sensor correspondiente: SensorPieza_Taladro detecta presencia de la pieza, entonces se activa Paso 8 y su retardo. Se memoriza el paso.



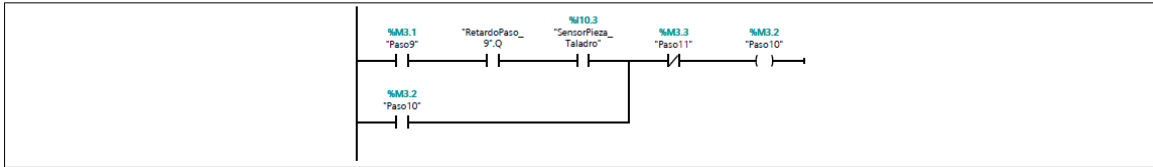
Segmento 11:

Una vez realizado el Paso 8 en un tiempo determinado dado por el timer TON: RetardoPaso_8, tiempo asignado para asegurar que el taladro barrenó, entonces si el sensor correspondiente: SensorPieza_Taladro continúa detectando presencia de la pieza, entonces se activa Paso 9 y su retardo. Se memoriza el paso.



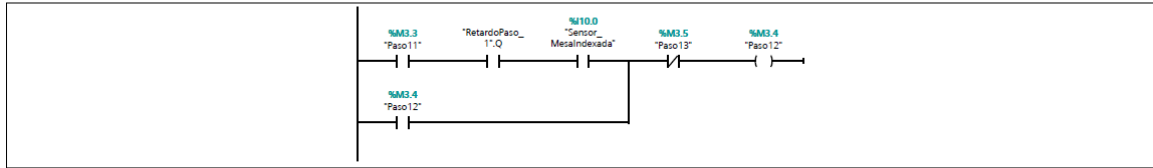
Segmento 12:

Una vez realizado el Paso 9 en un tiempo determinado dado por el timer TON: RetardoPaso_9, tiempo asignado para asegurar que el taladro subió, entonces si el sensor correspondiente: SensorPieza_Taladro continúa detectando presencia de la pieza, entonces se activa Paso 10 y su retardo. Se memoriza el paso.



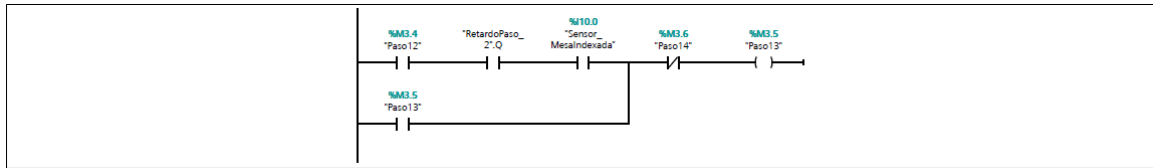
Segmento 14:

Una vez realizado el Paso 11 en un tiempo determinado dado por el timer TON: RetardoPaso_1, tiempo asignado para asegurar que la mesa gire, entonces si el sensor correspondiente: Sensor_MesaIndexada detecta posición, entonces se activa Paso 12 y su retardo. Se memoriza el paso.



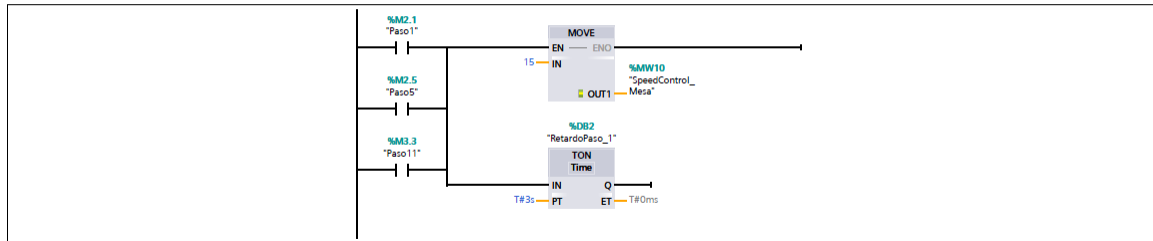
Segmento 15:

Una vez realizado el Paso 12 en un tiempo determinado dado por el timer TON: RetardoPaso_2, tiempo asignado para asegurar que la mesa para, entonces si el sensor correspondiente: Sensor_MesaIndexada detecta posición, entonces se activa Paso 13 y su retardo. Se memoriza el paso.



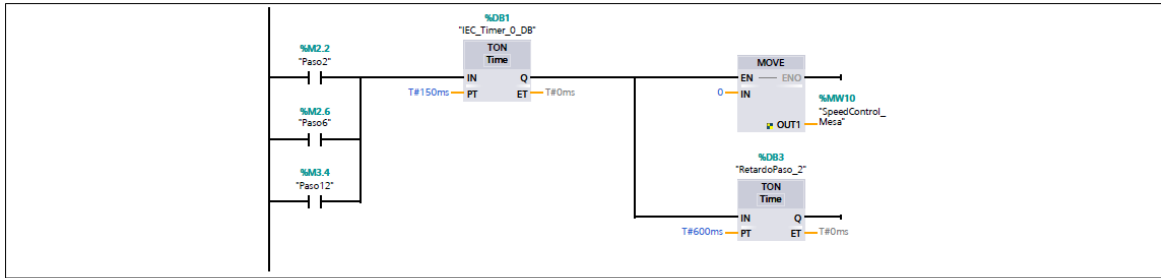
Segmento 16:

En los pasos 1, 5 y 11, la mesa debe rotar 60 grados a una velocidad de 15 grados por segundo, activamos un retardo de 3s.



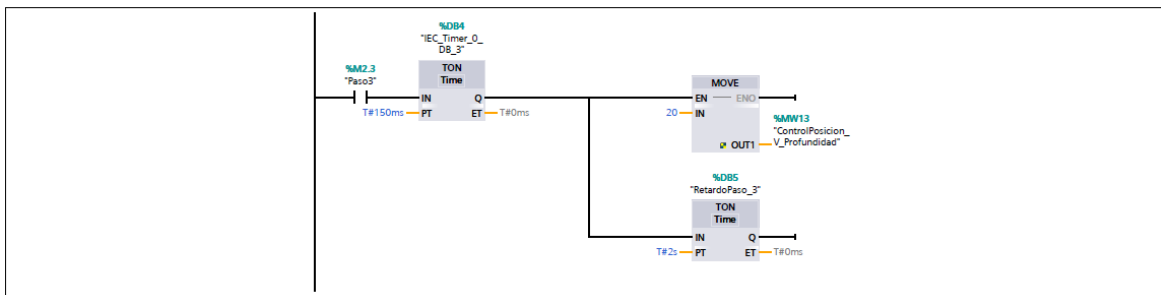
Segmento 17:

En los pasos 2, 6 y 12 a los 150 ms se para la mesa con un valor de 0 en la variable SpeedControl_Mesa, se debe activar nuevamente un RetardoPaso_2 de 600ms.



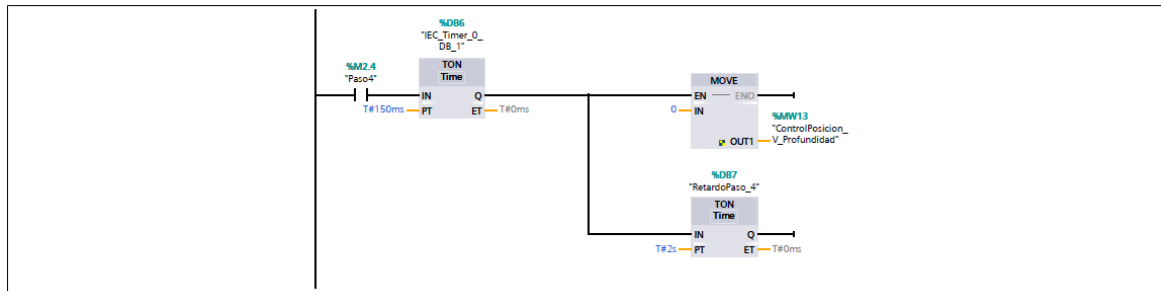
Segmento 18:

En el paso 3 a los 150 ms, se manda un valor de tipo entero de 20 (mm) a la variable de ControlPosición_V_Profundidad, activando al actuador que baja para validar que la pieza esté en la posición correcta con la profundidad adecuada. Se activa un RetardoPaso_3.



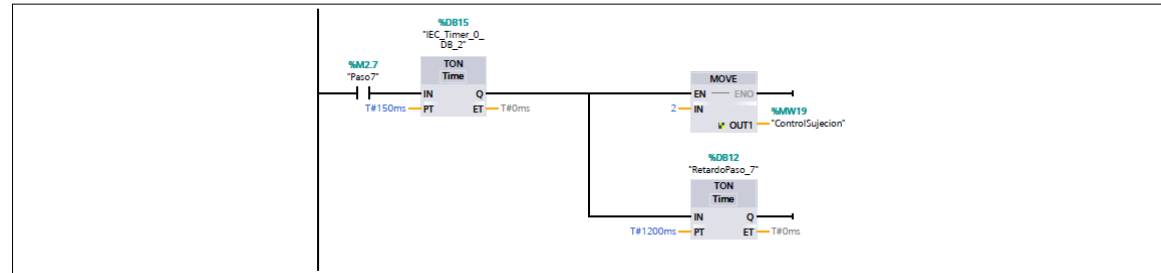
Segmento 19:

En el paso 4 a los 150ms, se manda un valor de tipo entero de 0 (mm) al controlPosición_V_Profundidad, para hacer regresar al actuador que valida posición y profundidad de la pieza para ser barrenada en el siguiente paso. Se activa un RetardoPaso_4 de 2s.



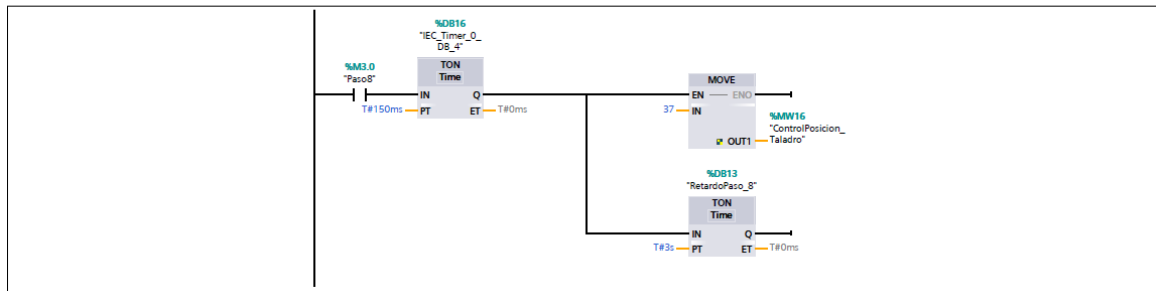
Segmento 20:

En el paso 7 a los 150 ms, se manda un valor de tipo entero: 2 (mm) al ControlSujeción y se activa un RetardoPaso_7 de 1200ms tiempo que asegura que se realice la acción de sujeción.



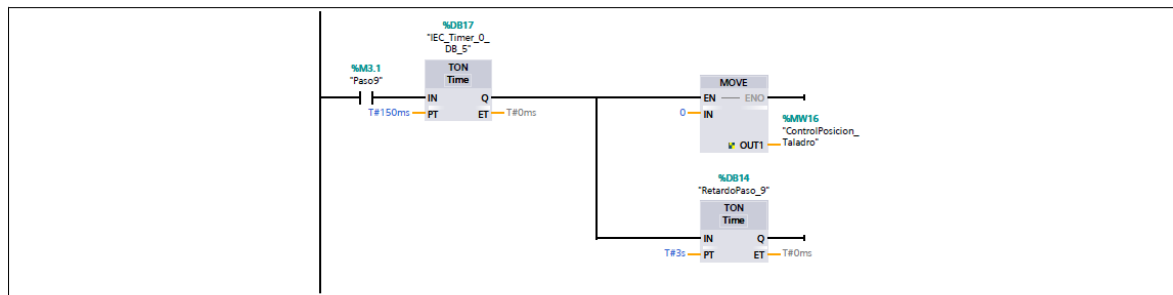
Segmento 21:

En el Paso 8, una vez sujeta la pieza a los 150 ms, se manda un valor de tipo entero: 37 (mm) al ControlPosición_Taladro, se activa un RetardoPaso_8 de 3 segundos, tiempo en el cual baja el sujetador del taladro.



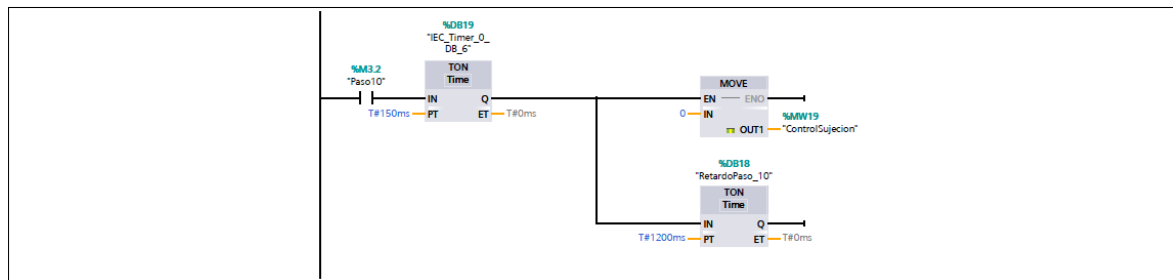
Segmento 22:

En el paso 9 a los 150 ms, se manda un valor de tipo entero a la variable ControlPosición_Taladro de 0, para que suba el sujetador del taladro en un tiempo de 3 segundos (RetardoPaso_9).



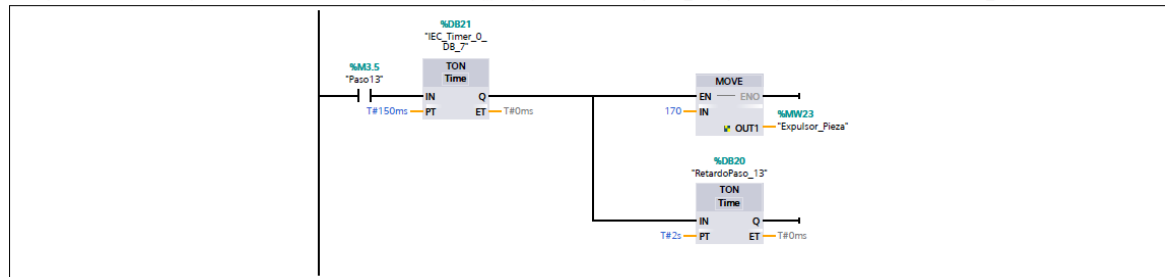
Segmento 23:

En el paso 10 a los 150 ms, se manda un valor de tipo entero al ControlSujeción para que regrese el actuador que sujeta la pieza para ser barrenada, en un de tiempo de 1200 ms (RetardoPaso_10).



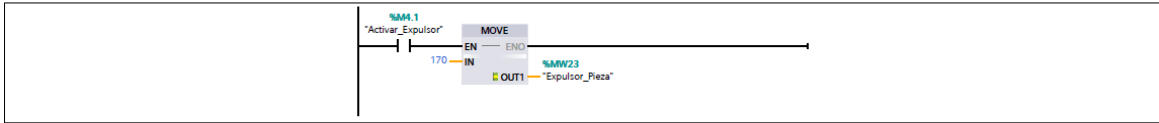
Segmento 24:

En el paso 13 a los 150 ms, se manda un valor de tipo entero de 170 (grados) a la variable Expulsor_Pieza, dándole un tiempo de 2s (RetardoPaso_13).



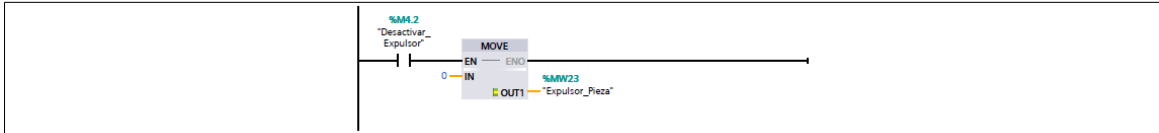
Segmento 25:

Mediante la marca Activar_Expulsor al forzarla estando en línea, se valida al mandar valores de tipo entero (grados) a la variable Expulsor_Pieza y al observar en la simulación conjunta con NX MCD, si el actuador expulsa correctamente la pieza fuera de la estación.



Segmento 26:

Mediante la marca Desactivar_Expulsor, se valida al mandar el valor de tipo entero: 0 (grados) a la variable Expulsor_Pieza y al observar en la simulación conjunta con NX MCD, que el actuador regrese a su posición inicial (home).



IV.7 Co-simulación

Para la simulación conjunta, se procede a:

- 1.- Dejar la instancia *Prueba* en modo Run en el PLCSIM Advanced.
- 2.- Compilar desde la carpeta del PLC en el árbol del proyecto (*Proyecto3*).
- 3.- Descargar el programa al PLC virtual.
- 4.- Arrancar módulos.

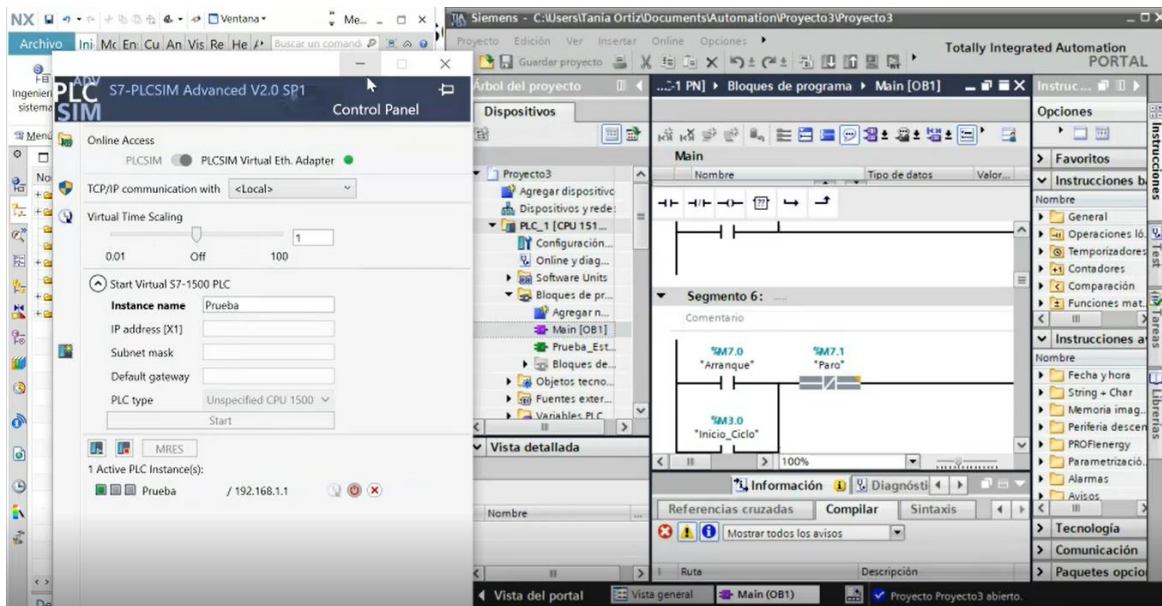


Figura IV.40 PLCSIM Advanced y TIA Portal.

5.- Conexión en línea para monitoreo del código.

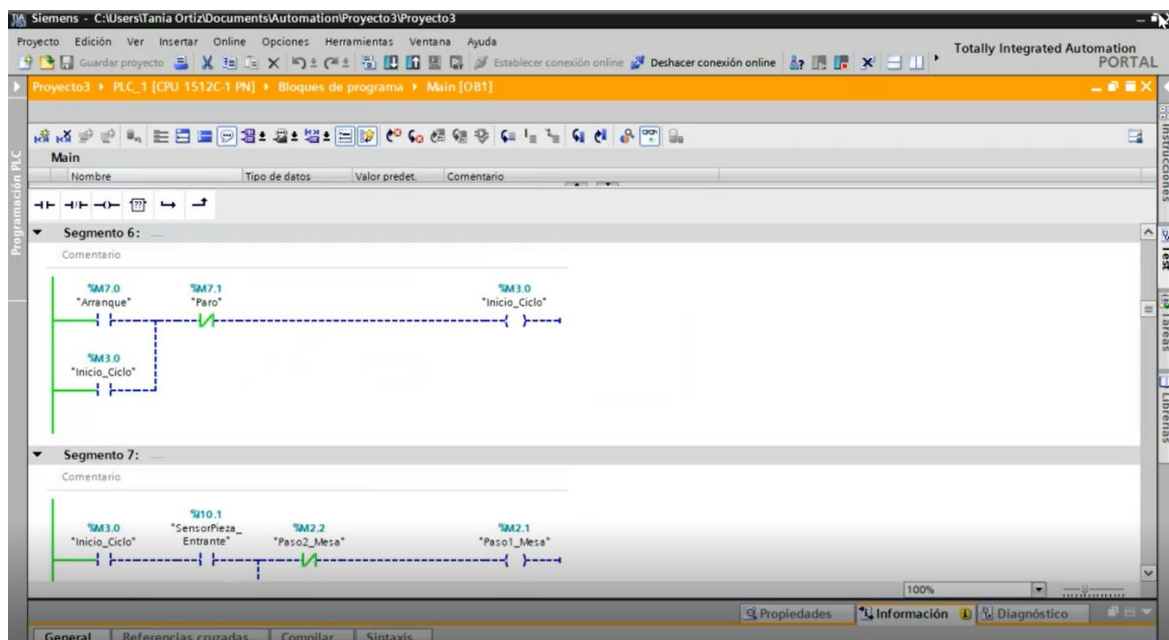


Figura IV.41 TIA Portal en línea.

6.- Activar simulación en NX MCD.

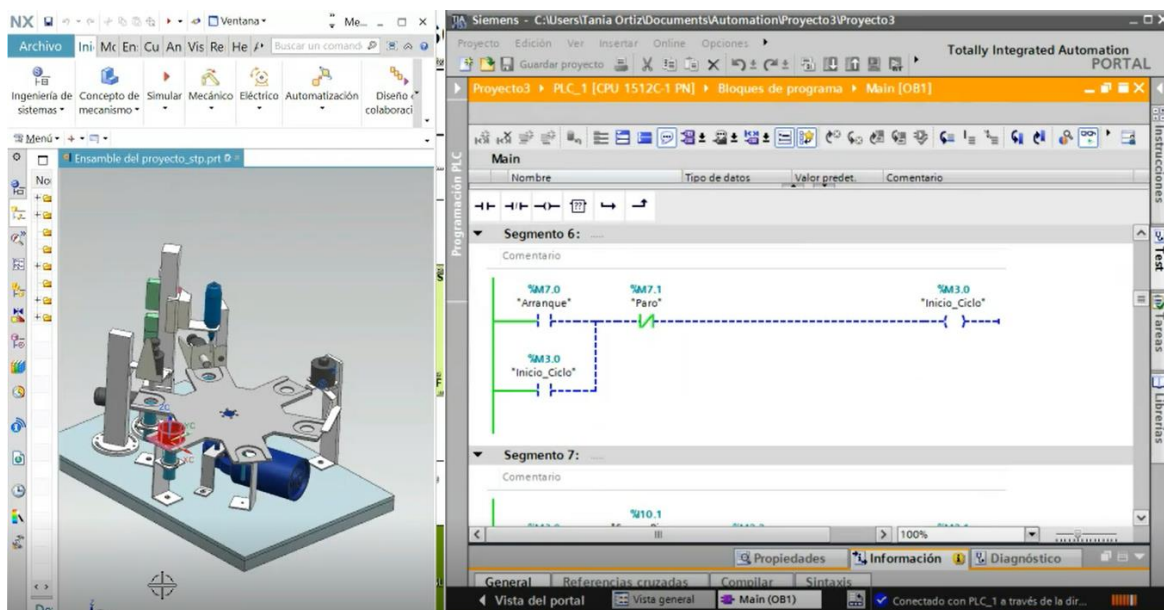


Figura IV.42 TIA Portal en línea y NX MCD corriendo simulación.

7.- Finalizar forzando el bit de arranque.

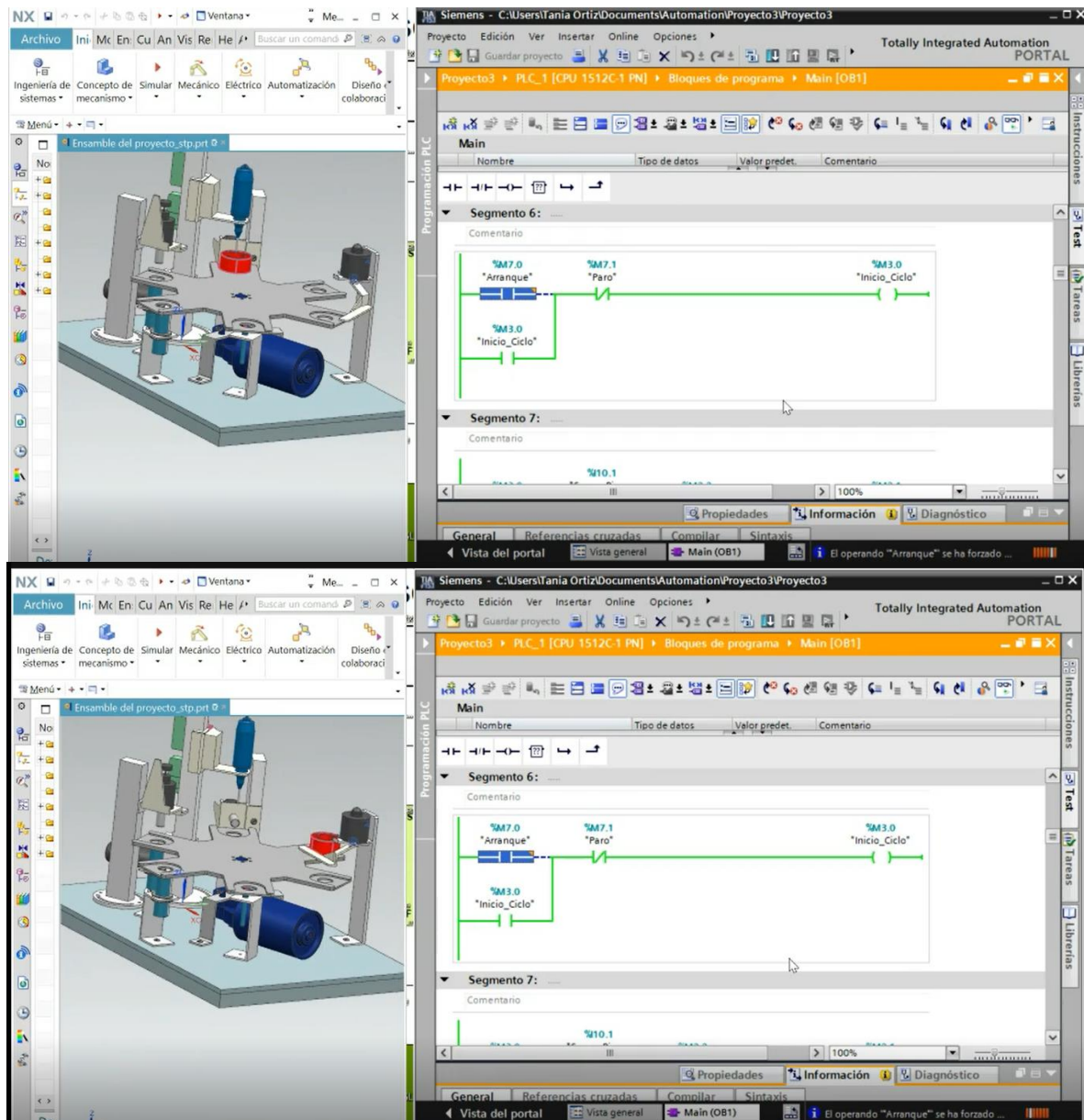


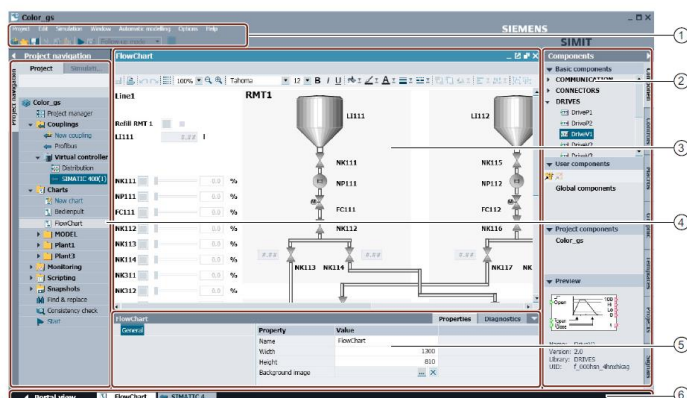
Figura IV.43 Simulación conjunta.

Bibliografía Capítulo IV

Este capítulo tiene como referencia los manuales:

SIMATIC S7-1500, S7-PLCSIM Advanced, Function Manual (2016). Alemania: Siemens AG, Division Digital Factory.

SIMATIC S7 y WinCC Engineering V15.1, Manual del sistema (2018). Alemania: Siemens AG, Division Digital Factory.



Co-simula componentes de control de potencia para:

-Motores de CA

-Servo motores

- Motores a pasos

-Motores de CD

Capítulo V

INTRODUCCIÓN AL USO DEL SOFTWARE SIMIT

Poner productos cada vez más personalizados en el mercado más rápido con demandas constantes de alta calidad requiere procesos cada vez más flexibles, por lo que el diseño de los mismos debe ser más eficiente, lo cual es posible lograrlo con el software SIMIT ya que permite una simulación y emulación virtual en tiempo real para la evaluación completa de las soluciones de automatización con menor riesgo; lográndose: Puesta en marcha virtual sencilla y flexible para mejorar la eficiencia de la ingeniería y una mayor fiabilidad en la planificación, formación segura y eficiente del personal de planta, o incluso en una formación a nivel superior y/o en una modalidad de formación DUAL.

V.1 Introducción

Gracias al software *SIMIT Simulation Framework V10 SCE Trainer*, la empresa Siemens ofrece una solución diseñada para entrenamientos y validaciones en el área de ingeniería en automatización de maquinaria y procesos industriales en ambientes totalmente virtuales, simulando dispositivos de estado sólido (variadores de velocidad, servo drives) o sistemas; facilitando a los estudiantes las pruebas de la programación del PLC y la integración completa de la solución tecnológica en condiciones reales.

De esta manera, la tarea propuesta se puede comprender más rápidamente, el estudiante se puede concentrar en la generación del programa PLC y probarlo cuantas veces sea necesario en el simulador; a través de la interesante representación de los dispositivos controlados y sistemas de simulación, se crea una emulación más pertinente con el objetivo de encontrar una solución técnica a la automatización correcta.

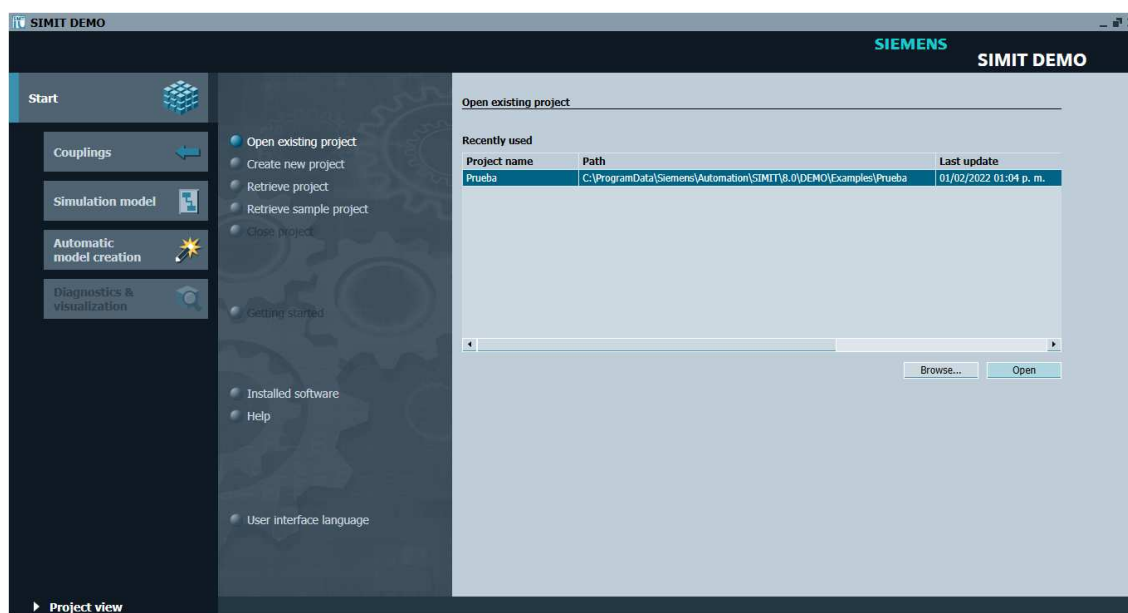


Figura V.1 Interfaz inicial de SIMIT.

V.2 Instalación y descarga del software SIMIT

Primeramente se ingresa a la página oficial de Siemens y se busca la versión de *SIMIT* que se desee descargar, en este libro se empleará la ver. 10.0, es necesario registrarse en la página. La descarga proporciona un archivo comprimido, el cual es necesario descomprimir, es importante que el archivo con extensión .exe se ejecute como *administrador*.

The screenshot shows the Siemens Industry Online Support website. The header includes the Siemens logo and navigation links like 'Home', 'Product Support', 'Contact', 'Help', 'Support Request', 'Site Explorer', and a search bar. The main content area is titled 'Simulation software SIMIT V10.2' and includes a 'Download' button. Below the title, there is a 'Readme file' section with a yellow box highlighting the 'Manuals' section, which lists the 'SIMIT V10.2 Add-in' and the 'SIMIT V10.2 Add-in for SIMATIC Manager'. The 'Installation file' section mentions the 'SIMIT V10.2 Add-in for SIMATIC Manager' (291.6 MB). The right sidebar contains sections for 'mySupport Cockpit', 'Product information', 'Related links', and 'Topic pages for this entry'.

Figura V.2 Página de Siemens de donde se descarga el archivo ejecutable.

Al ejecutar el archivo en la ventana emergente se indica la versión, dar clic en *next*.

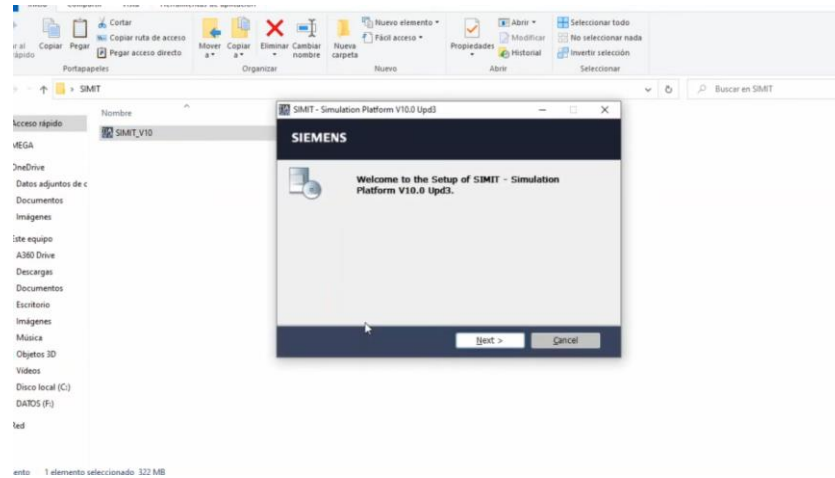


Figura V.3 Ventana de instalación.

A continuación, se selecciona el idioma en el que se desea instalar, dar clic en *next*.

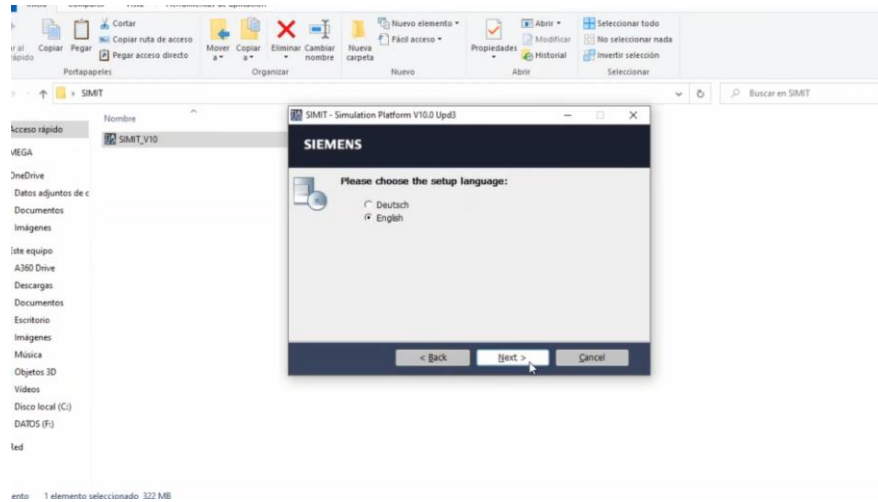


Figura V.1. Selección de idioma.

Con la opción *browse* se elige el disco y carpeta donde se instalará, dar clic en *next*.

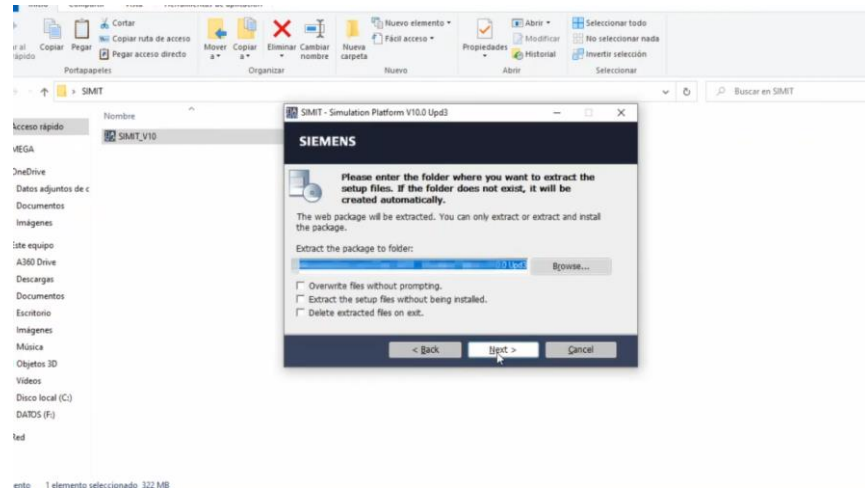


Figura V.5 Selección de dirección.

El proceso de extracción del contenido tarda unos segundos.

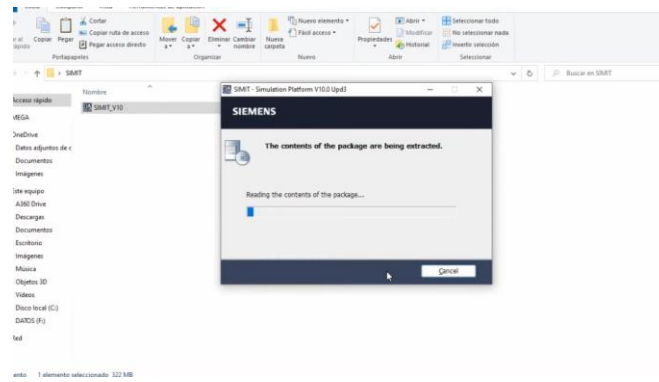


Figura V.2 Fin de la configuración inicial.

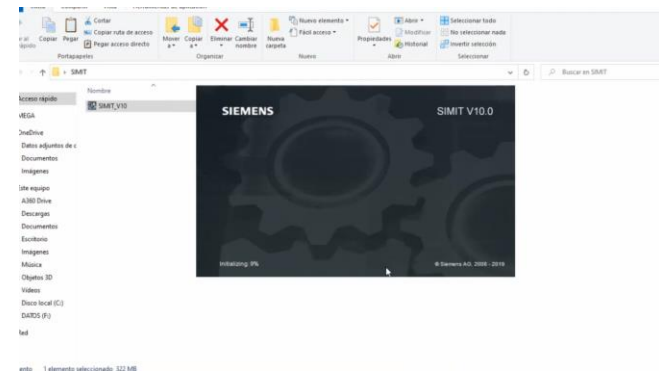


Figura V.3 Inicio de la configuración final.

Se válida la selección del idioma, después dar clic en *next*.



Figura V.4 Ajustes generales

En caso de haber cambiado la dirección donde se instala el software, se corrobora la dirección en la siguiente ventana, por otra parte, es importante seleccionar los dos software, dar clic en *next*.

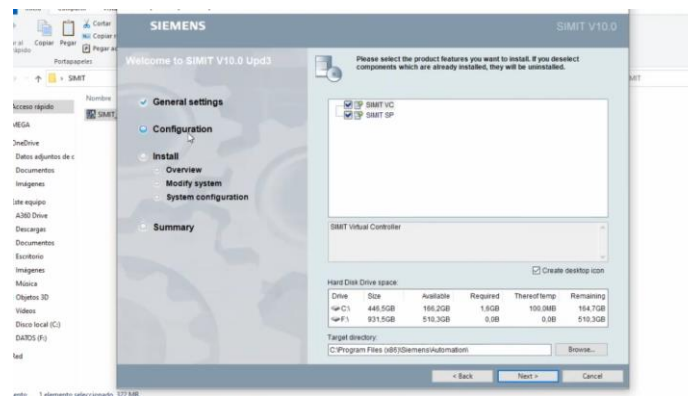


Figura V.5 Configuración.

Se aceptan los términos y condiciones, dar clic en *next*.

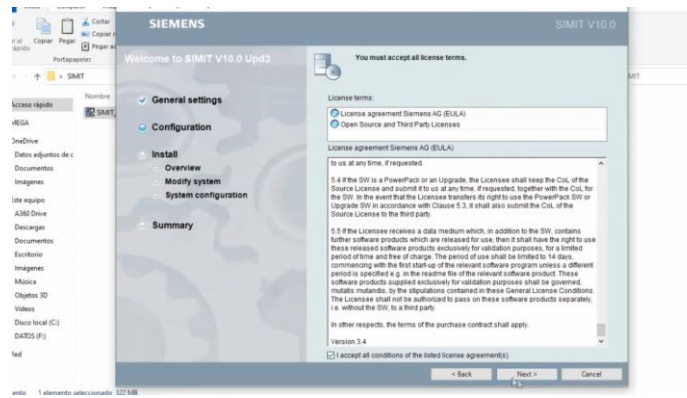


Figura V.6 Configuración: términos y condiciones.

Se selecciona la casilla que se encuentra al fondo y se da clic en *next*.

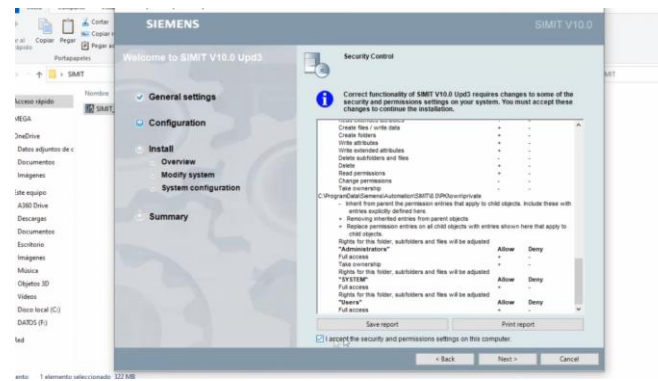


Figura V.7 Configuración: seguridad y permisos.

Se da clic en *install*.

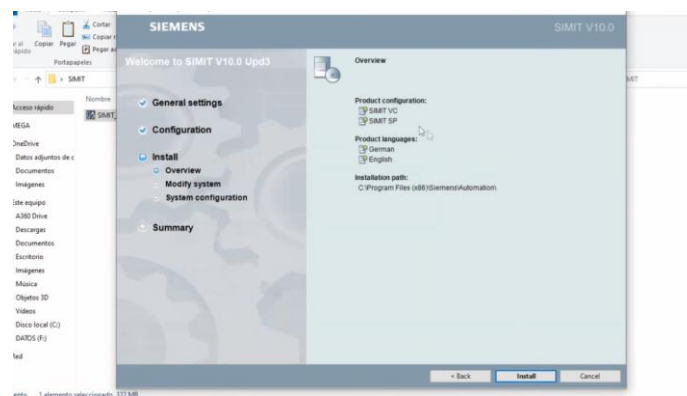


Figura V.8 Instalar.

Se iniciará la instalación, cada vez que se solicite reiniciar la computadora es necesario hacerlo.



Figura V.9 Final de la configuración final.

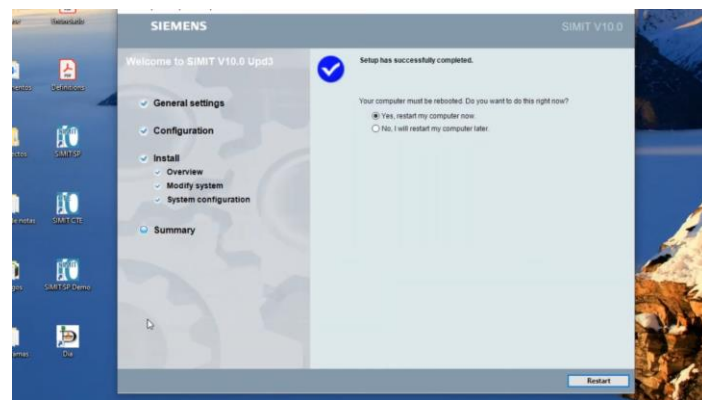


Figura V.14 Resumen de la instalación.

En el escritorio se tienen tres iconos, en caso de no contar con las licencias de *SIMIT*, se puede usar la versión DEMO, la cual funciona de la misma manera pero con algunas limitaciones, las cuales se explican en el siguiente apartado.

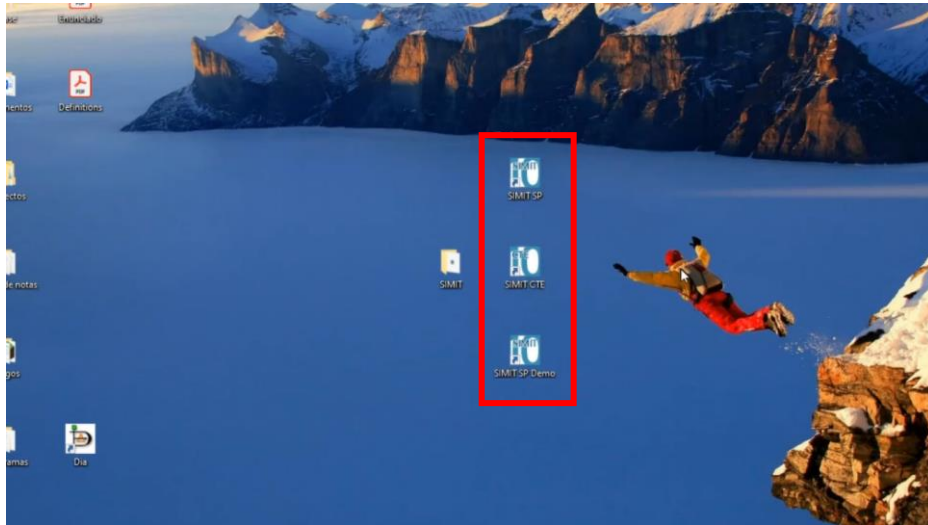


Figura V.15 Resultado final de la instalación.

V.3 SIMIT SP Demo

Para iniciar el modo DEMO es necesario dar un doble clic en el icono en el escritorio *SIMIT SP Demo*. En este modo sólo están disponibles los siguientes módulos de funciones de *SIMIT*:

- Editor de componentes de macros.
- Editor de gráficos.
- Editor de tendencias y mensajes.
- Creación automática de modelos.

SIMIT sólo tiene una funcionalidad limitada en las siguientes áreas en modo DEMO:

- Controlador virtual:

Puede ejecutar *SIMIT* en modo DEMO y el controlador virtual con licencia o viceversa. Sin embargo, las restricciones del modo DEMO se aplican tanto a *SIMIT* como al controlador virtual.

- Guardar y archivar:

Los proyectos, plantillas y macro componentes pueden guardarse en modo DEMO. Sin embargo, los proyectos, plantillas y macro componentes creados en modo DEMO sólo pueden utilizarse en el ordenador en el que se crearon.

Nota: Los proyectos, plantillas y macro componentes creados en modo DEMO no son compatibles con la versión completa de SIMIT. Los proyectos creados en modo DEMO no pueden actualizarse. No se pueden archivar proyectos en modo DEMO.

-
- Abrir y recuperar:

Puede abrir un proyecto en modo DEMO bajo las siguientes condiciones:

- ✓ El proyecto fue creado en este ordenador en modo DEMO.
- ✓ La versión del proyecto corresponde a la versión de software instalada.

Los proyectos creados con una versión completa no pueden abrirse. Se pueden recuperar proyectos que hayan sido archivados en una versión completa. Sin embargo, si el proyecto recuperado se modifica en modo DEMO, no podrá utilizarse entonces en la versión completa.

- Área de direcciones:

En el modo DEMO, se pueden conectar un máximo de 30 señales de acoplamiento en la simulación siempre que se cubra un máximo de 30 bytes, cada una en el área de entrada y en el área de salida.

- Tiempo de ejecución:

Puede utilizar *SIMIT* en modo DEMO todo el tiempo que desee, pero el tiempo de ejecución de una simulación está limitado a 45 minutos. La simulación termina automáticamente al final de esos 45 minutos. Puede reiniciar la simulación una vez que haya finalizado.

- Número de acoplamientos:

La interfaz de programación "Acoplamientos externos" no es compatible.

- Carpeta del proyecto:

En el modo DEMO, sólo puede guardar los proyectos en una ubicación de memoria especificada en el área de trabajo de *SIMIT*.

- Bibliotecas para macro componentes y plantillas:

En el modo DEMO, sólo puede guardar componentes de macro y plantillas dentro del área de trabajo de *SIMIT*. No puede abrir otras carpetas de biblioteca.

V.4 Herramientas de la interfaz de SIMIT

Se puede iniciar *SIMIT* mediante el menú de inicio de Windows:

"Todos los programas > Siemens Automation > SIMIT > Plataforma de simulación SIMIT", o mediante el icono que aparece en la pantalla.

Al iniciar *SIMIT*, seleccione la variante de tamaño (XS, S, M, L, XL). También puede cambiar la variante de talla más tarde a través de:

"Vista de portal > Inicio > Configuración > Variante de talla".

La vista del portal o la vista del proyecto se inicia tras ejecutarlo. La vista que se abre se ajusta a través de:

"Vista del portal > Inicio > Configuración".

La clara disposición de la vista de portal ayuda a familiarizarse rápidamente con *SIMIT*. Las funciones básicas más importantes pueden seleccionarse directamente:

- Gestionar proyectos.
- Crear acoplamientos.
- Crear modelos de simulación.
- Importar datos automáticamente desde diferentes formatos de archivo.

Las funciones adicionales son:

- Realizar comprobaciones de coherencia.
- Acceder a la función "Buscar y reemplazar".
- Crear una nueva tendencia.
- Procesar archivos.

Hacer clic en "Vista del proyecto" para cambiar a la vista del proyecto. La funcionalidad completa de *SIMIT* sólo está disponible en la vista de proyecto.

V.5 Cómo empezar

Después de abrir, crear o recuperar un proyecto, se abre la sección "Primeros pasos". Aquí se pueden añadir los objetos básicos necesarios para el funcionamiento de un proyecto *SIMIT*.

- Añadir acoplamiento.
- Añadir gráfico.
- Generar gráficos automáticamente.

Se pueden realizar estos pasos aquí o ir a la vista del proyecto para continuar editando el proyecto e iniciar la simulación.

V.6 La interfaz gráfica de usuario de SIMIT

La interfaz de usuario de SIMIT se divide en las siguientes áreas:

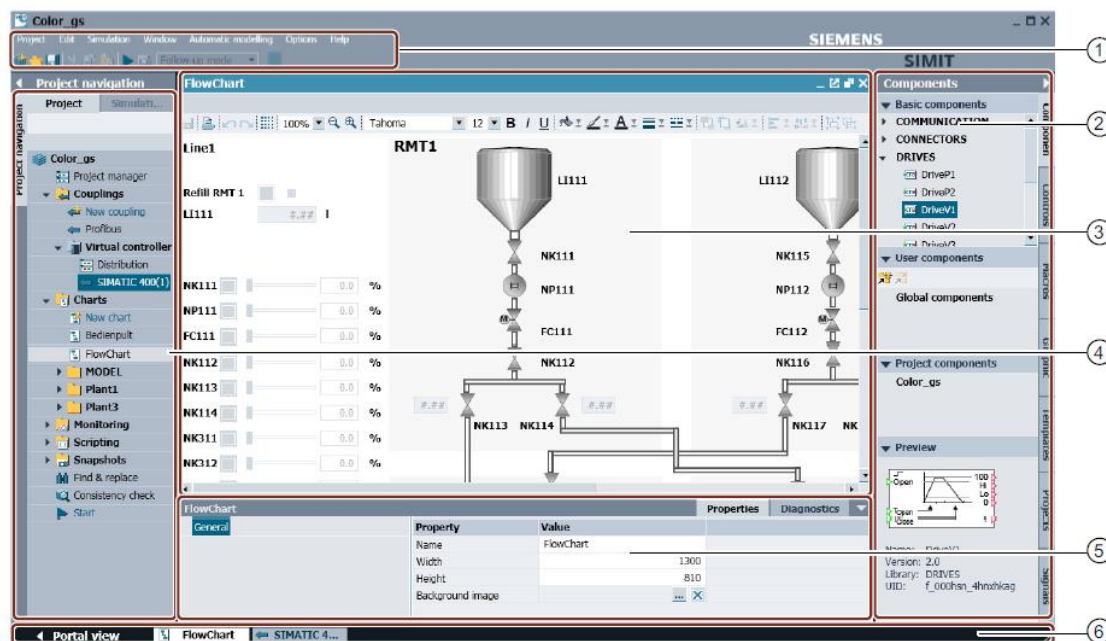


Figura V.16. Interfaz gráfica de usuario de SIMIT.

1. Barra de menú y barra de herramientas:

Acceso a las funciones de SIMIT. Otras funciones están disponibles en los menús de acceso directo en la navegación del proyecto.

2. Guías de tareas:

Aquí se enumeran los objetos, como los componentes de la biblioteca, los controles y los objetos gráficos que pueden utilizarse en el editor actualmente abierto. Estos objetos se clasifican en hojas de ruta.

3. Área de trabajo:

Aquí se abren los editores para la edición.

4. Navegación por el proyecto:

El proyecto actual se muestra aquí en una vista de árbol.

5. Vista de propiedades:





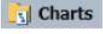


Aquí se muestran las propiedades del objeto seleccionado.

6. Barra del editor y barra de estado:

Aquí se puede cambiar entre los editores abiertos y la vista del portal. Si es necesario, aquí se muestra información sobre el estado actual de *SIMIT*. Todos los editores se abren en el área de trabajo. Las hojas de ruta son específicas para cada editor. El área de trabajo puede dividirse para abrir dos editores uno al lado del otro o uno debajo del otro en el área de trabajo.

V.7 Navegación de proyectos

Los proyectos se gestionan en la navegación de proyectos. Un proyecto *SIMIT* está dividido en los siguientes nodos de árbol, por lo que un proyecto no debe contener todos los objetos aquí enumerados:

	<p>Nombre del proyecto actual</p> <p>Nodo del árbol del gestor de proyectos. Los elementos de otros proyectos pueden copiarse en el proyecto actualmente abierto en el gestor de proyectos.</p> <p>Carpeta "Acoplamientos": Carpeta para almacenar los acoplamientos del proyecto actual.</p>
	
	<p>Símbolo de los acoplamientos. Los acoplamientos crean la conexión entre <i>SIMIT</i> y un controlador u otra aplicación.</p> <p>Nodo de árbol para crear un nuevo acoplamiento.</p>
	Carpeta "Charts": Carpeta para los gráficos.
	<p>Símbolo de los gráficos. Los gráficos contienen un modelo de simulación creado con componentes y controles de la biblioteca.</p> <p>Nodo de árbol para crear un nuevo gráfico.</p>
	<p>Carpeta "Monitoring": Esta carpeta contiene las siguientes funciones y es también su lugar de almacenamiento:</p> <ul style="list-style-type: none"> • Nueva tendencia. • Mensajes. • Archivo. <p>Carpeta "Scripting": Esta carpeta contiene la función "Nuevo script" y es la ubicación de almacenamiento de los scripts existentes.</p>
	Nodo de árbol para iniciar la simulación.

V.8 Configuración con TIA Openness para poder realizar comunicación usando couplings

En *TIA Portal* se puede integrar en el sistema de desarrollo de usuarios a través de la interfaz *API de WinCC* y *STEP 7*, y se pueden completar automáticamente varias tareas de configuración de ingeniería. Se puede utilizar un sistema de desarrollo externo para escribir aplicaciones de usuario.

1. Al instalar el software *TIA Portal* por primera vez, compruebe el componente *Apertura* durante el proceso de instalación.
2. Si ya ha instalado el software *TIA Portal*, puede ejecutar de nuevo el archivo *setup.exe* y seleccionar la opción "Modificar" para continuar con la siguiente operación de instalación. Durante el proceso de instalación, verifique el componente *TIA OPENESS*.
3. Una vez finalizada la instalación, abrir el software *TIA Portal* para comprobar si se ha instalado.

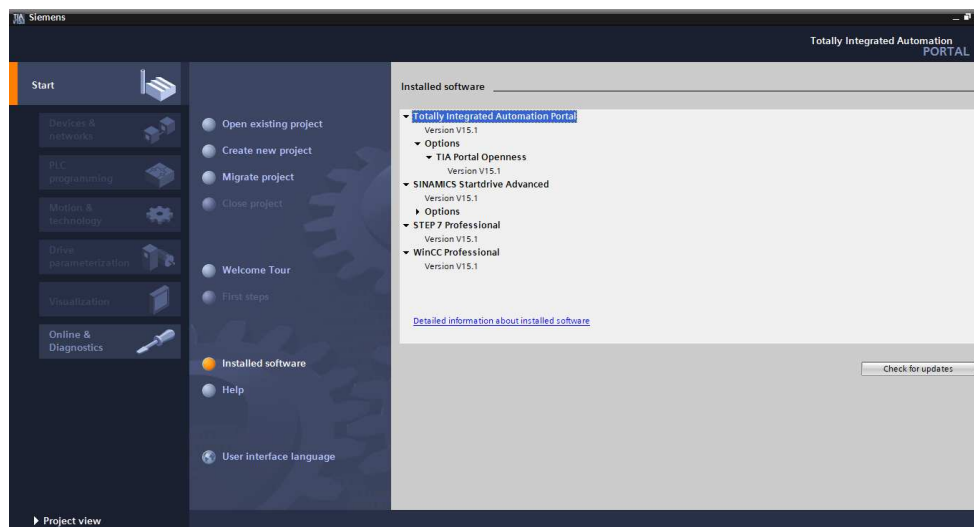


Figura V.10 TIA Portal.

V.9 Agregar usuarios al grupo de usuarios "Siemens TIA Openness"

Cuando se instala *TIA Portal Openness* en la computadora, se crea automáticamente el grupo de usuarios "Siemens TIA Openness". Siempre que accede al *TIA Portal* a través de la aplicación *TIA Portal Openness*, el software verifica que sí se es miembro directo o indirecto del grupo de usuarios "Siemens TIA Openness" a través de otro grupo de usuarios. Si es miembro del grupo de usuarios "Siemens TIA Openness", la aplicación *TIA Portal Openness* se iniciará y establecerá una conexión con *TIA Portal*.

En pocas palabras, el software de instalación aún no está disponible y se requiere una configuración adicional. De lo contrario, aparecerá el siguiente mensaje de error durante el uso:

"No se pudo iniciar la operación CAX. Compruebe si ha iniciado sesión como miembro del grupo "Siemens TIA Openness" y reinicie el proceso."

El siguiente paso sería agregar *TIA Openness* a los permisos del usuario, este procedimiento se divide en ocho pasos:

1. Ingrese a la administración de la computadora, seleccione *Grupo* en el directorio *Usuarios y grupos locales* a la derecha, y seleccione *Siemens TIA Openness* a la derecha.

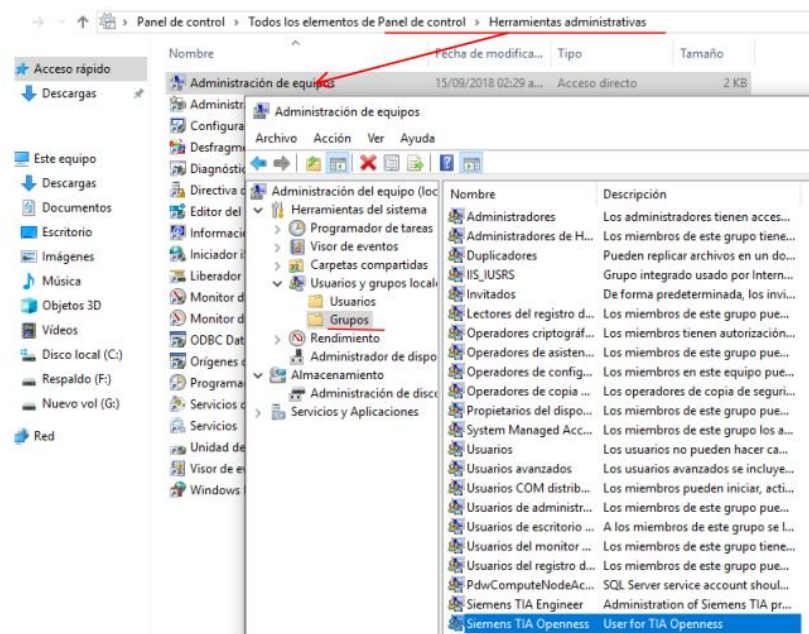


Figura V.11 Primera parte de la configuración.

2. Seleccione *Agregar al grupo en Operación* en la barra de menú.

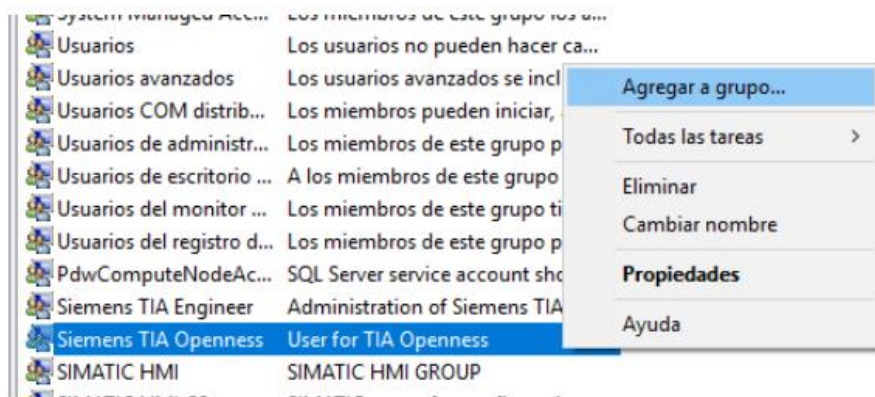


Figura V.12 Segunda parte de la configuración.

3. Haga clic en el botón *Agregar*.

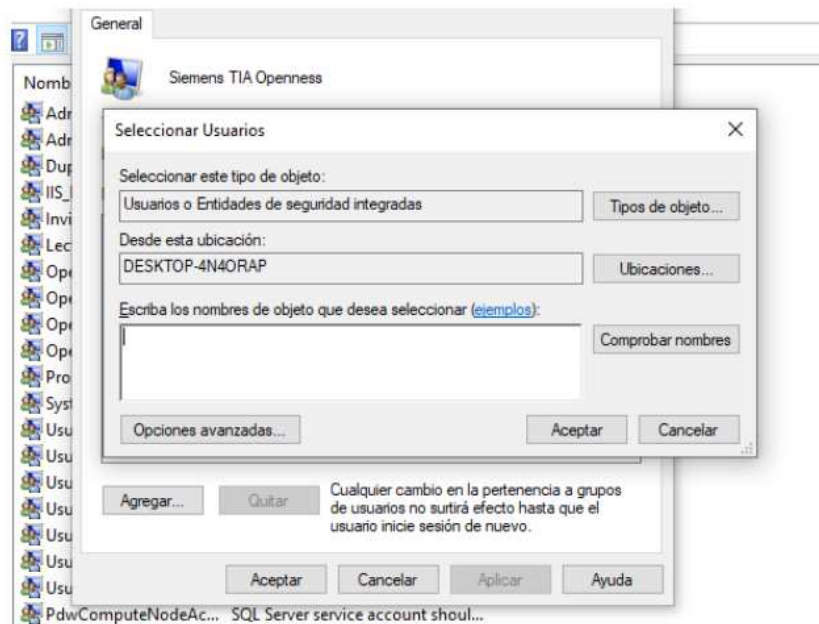


Figura V.13 Tercera parte de la configuración.

4. Este paso se debe seleccionar el usuario del sistema operativo y hacer clic en el botón *Avanzado*.

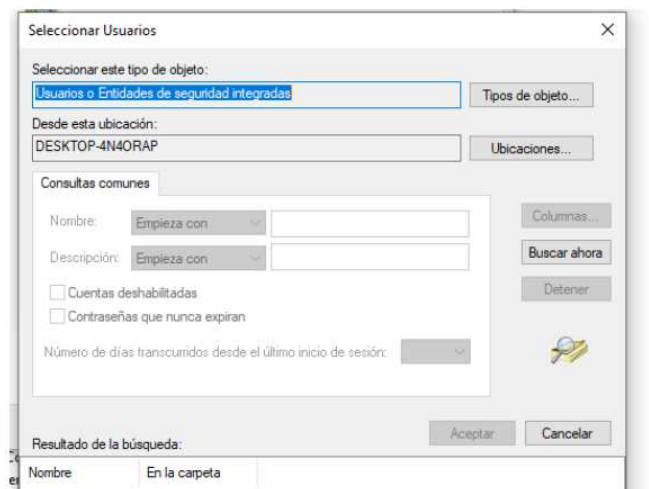


Figura V.14 Cuarta parte de la configuración.

5. Hacer clic en el botón *Buscar ahora* y el sistema escaneará automáticamente. Los resultados aparecerán en la lista *Resultados de la búsqueda*.

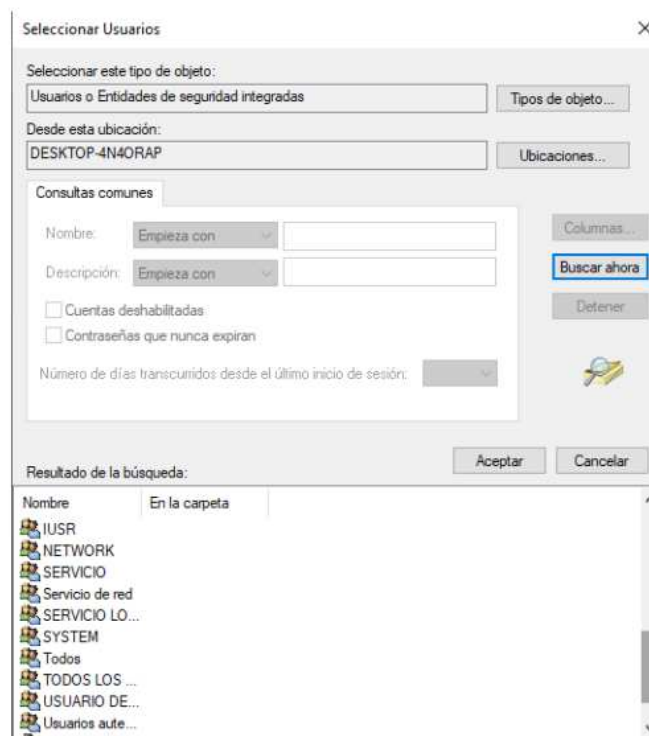


Figura V.15 Quinta parte de la configuración.

6. Elegir un nombre de usuario para el sistema, el cual debe usarse para iniciar sesión en el sistema operativo para realizar las siguientes operaciones de configuración de importación y exportación. Después de seleccionar, haga clic en el botón *Aceptar*.
7. Hacer clic en el botón *Comprobar nombre* para comprobar si es correcto, si no es correcto, volver a los pasos anteriores para corregirlo. Si es correcto, no se dará ningún aviso. Luego haga clic en el botón *Aceptar*.
8. En este punto, se puede ver que el nombre de usuario seleccionado anteriormente se ha incluido en la lista *Miembros*. La finalización de este paso significa que este usuario puede utilizar las funciones de *TIA Openness* y haga clic en el botón *Aceptar*.

V.10 Interacción con otros softwares como S7-PLCSIM Advanced y Siemens NX MCD (Mechatronics Concept Designer)

Una de las principales características de *SIMIT* es que se puede enlazar con otros softwares como lo son: *S7-PLCSIM Advanced* y *Siemens NX MCD*, procedimiento denominado: co-simulación, la cual permite validar la integración de distintos dispositivos de estado sólido y su parametrización y programación el control y automatización industrial.

V.10.1 S7-PLCSIM Advanced

SIEMENS es una empresa alemana que apuesta por convertir a distintas empresas en parte de la industria 4.0, razón por la que ha creado un simulador dotado con todas las capacidades de comunicación. Los controladores virtuales: S7-1500 y ET200 pueden validarse su programación en distintos lenguajes: escalera, estructurado, listado de instrucciones, bloques de funciones y *Grafcet* del francés *Graphe Fonctionnel de Commande Etape Transition*.

Una de las características más importantes del *S7 PLCSIM Advanced* es que permite variar los tiempos en los procesos simulados; al acelerar las emulaciones se obtienen datos y/o resultados rápidamente, además es capaz de realizar simulaciones completas de lógica programada. La ejecución del programa *S7 PLCSIM Advanced* se puede hacer con *TIA Portal*, tal como si fuera un PLC real; primero se agrega el dispositivo, se configura la dirección IP y se edita el programa que se quiere simular y se procede a la exportación. Para poder explicar cómo se hace la conexión entre estos dos softwares (*S7-PLCSIM Advanced* y *SIMIT*) se ejemplificará.

Primeramente, se debe contar con un proyecto creado en *SIMIT*, debemos de seleccionar la opción *new coupling*, la cual nos desplegará un menú donde debemos de seleccionar la opción *PLCSIM advanced*.

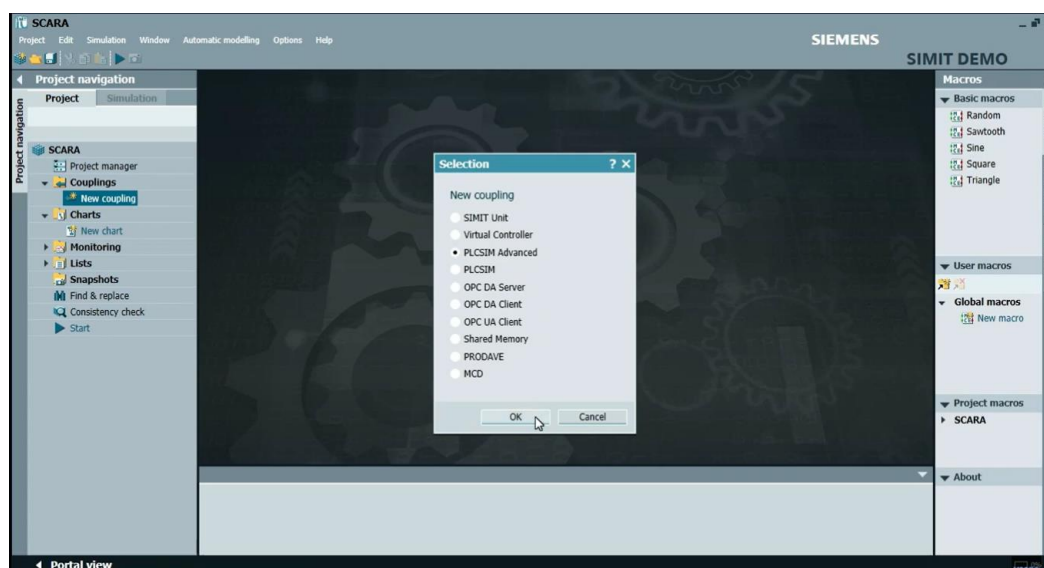


Figura V.16 Seleccionando el tipo de acople.

Después tenemos que seleccionar el archivo en *TIA Portal* donde se ha editado el programa e importarlo a *SIMIT*.

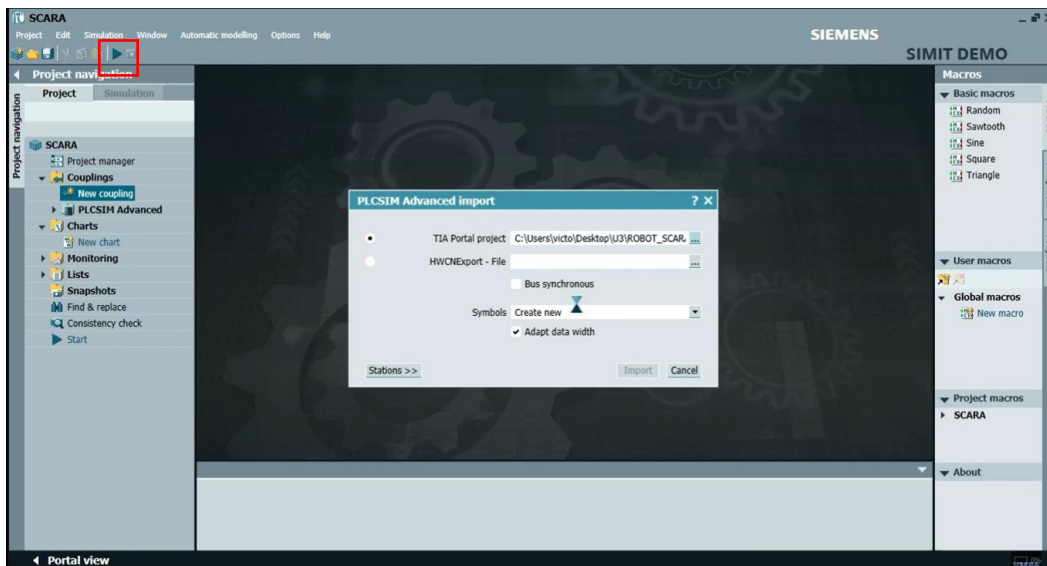


Figura V.17 Selección de proyecto en TIA Portal.

Seguido de esto, nos aparecerán todas las variables que hemos declarado en *TIA Portal*, posteriormente se inicia la simulación dando click en la flecha que se encuentra en la barra superior.

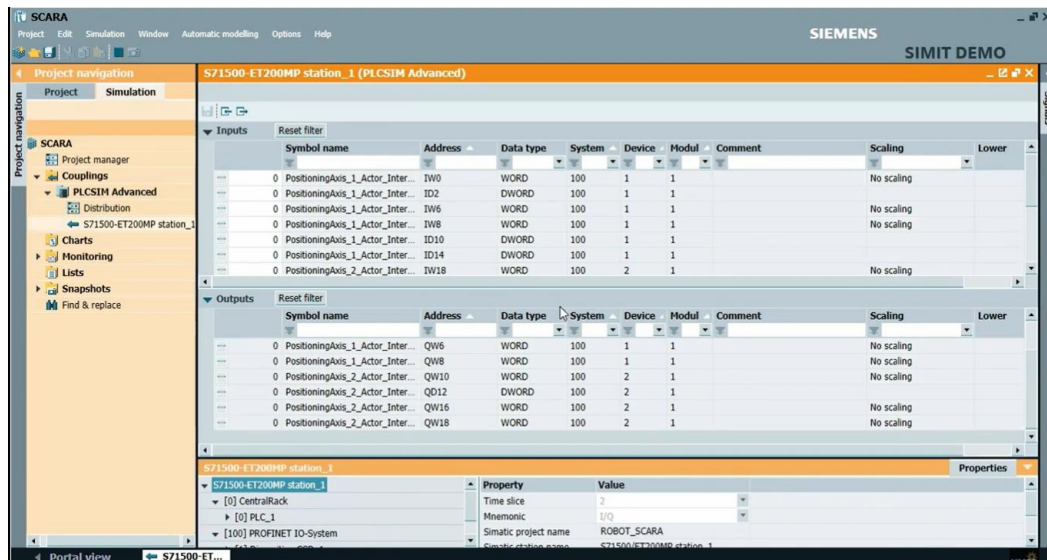


Figura V.18 Variables importadas y simulación iniciada.

Ahora se ira de nuevo a *TIA Portal*, se compila y carga el programa, el cómo cargar el programa se hace de la manera que se muestra en las figuras V.26 y V.27. Para hacer la

compilación solo es necesario dar clic en el icono que está a un lado de el de cargar, el cual se señala en la Figura V.26.

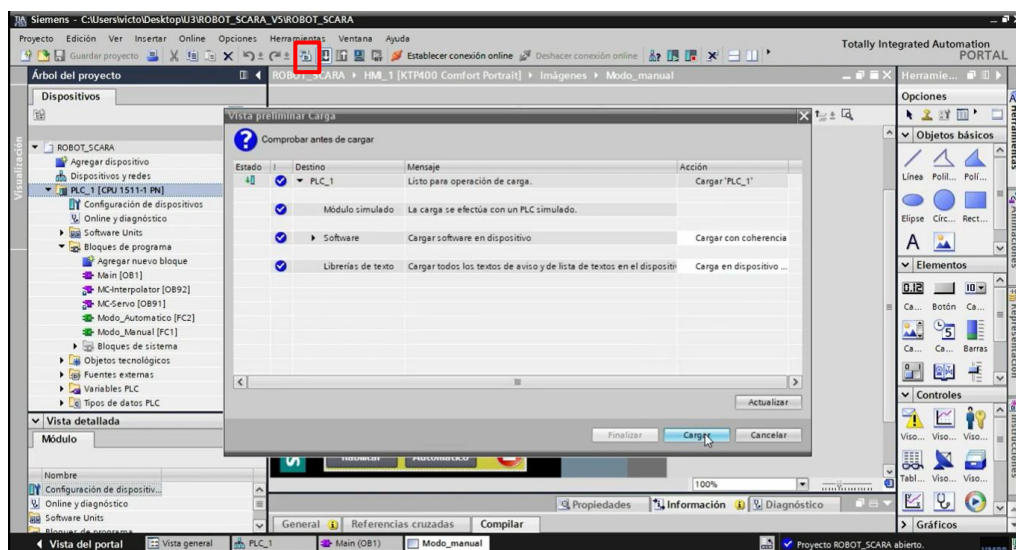


Figura V.19 Primer paso para cargar.

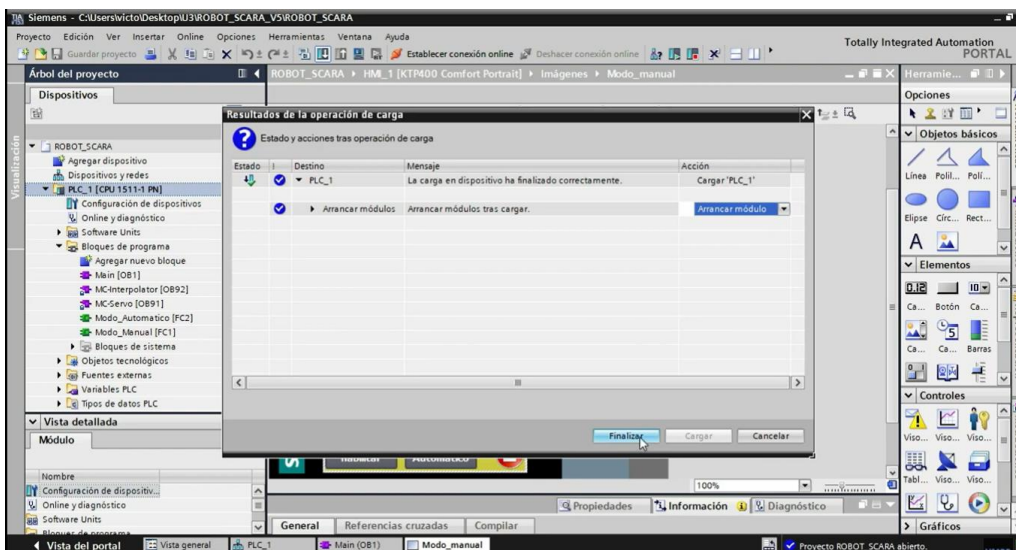


Figura V.20 Segundo paso para cargar.

Dar clic en *establecer conexión online* para poder comunicar en línea TIA Portal y SIMATIC, si no se cometió error en la conexión hasta este punto, en el árbol de proyecto de TIA Portal se observan círculos de color verde, en caso contrario se señala el o los errores y se especifica donde se encuentran éste o éstos en la ventana de inspección.

V.10.2 Co-simulación NX MCD y SIMIT

Para poder explicar cómo se hace la conexión entre estos dos softwares: *NX MCD* y *SIMIT*, se empleará un ejemplo, a continuación, se indican los pasos a seguir:

1.- Crear un nuevo acople, este no será de *PLCSIM advanced*, será de *NX MCD*.

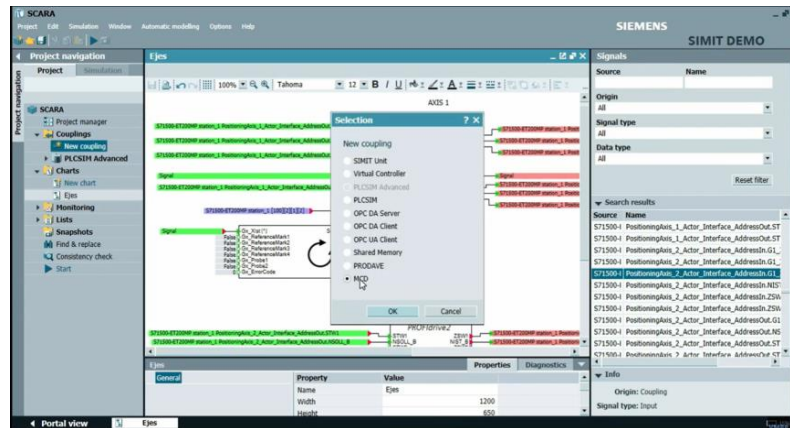


Figura V.21 Seleccionando el tipo de acople.

2.- Cargar el archivo de Siemens *NX MCD*, para lo cual seguir los pasos que se pueden ver en la Figura V.29.

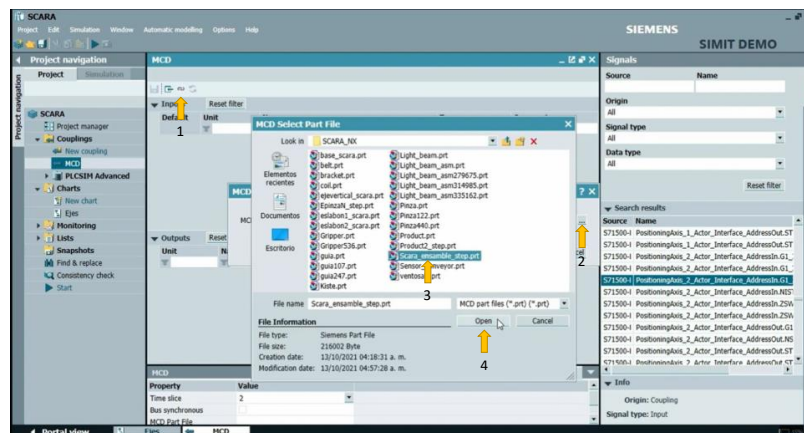


Figura V.22 Selección de proyecto en Siemens NX MCD.

3.- Si aparece el siguiente error, únicamente dar clic en aceptar.

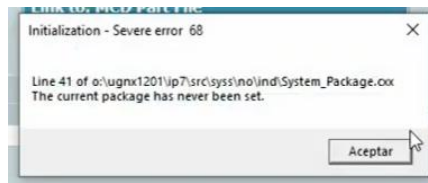


Figura V.23 Posible error.

4.- Todas las variables de Siemens *NX MCD* en *SIMIT*, en caso de ser necesario se pueden cambiar las unidades de las variables.

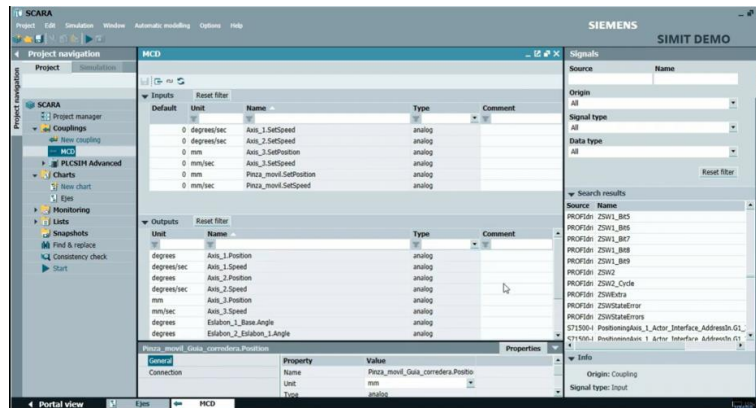


Figura V.24 Variables de Siemens *NX MCD* con su unidad correcta.

V.11. Descripción proyecto ejemplo: “Selector de cajas por peso”

El programa de control operará a través de *PLCSIM*, el cual controlará la emulación de nuestra planta virtual en *SIMIT*; es importante aclarar que son software de diferentes versiones el *PLCSIM* y el *S7-PLCSIM Advanced*, éste último es una evolución de los primeros simuladores de la marca para los controladores virtuales: S7-1500 y ET200.

El problema es el siguiente: una cinta transportadora C0, es la encargada de alimentar las cajas a la maqueta, las cajas son de diferentes pesos, estas llegan hacia un sensor SCP encargado de detectar que la caja está posicionada encima de la báscula, ésta la pesa, después del pesado pasan a la cinta C1, la cual es la encargada de clasificarla automáticamente. Las cajas con un peso de hasta 5 kg son desviadas hacia la cinta C2, las de peso de entre 5 a 10 kg hacia la cinta C4 y las cajas con un peso mayor de 10 kg se dirigen a la cinta C3. Las cajas tendrán un peso de entre 0 a 15 kg. El sistema tiene 5 motores: M0, M1, M2, M3, M4 uno por cada cinta transportadora: C0, C1, C2, C3, C4.

En el Panel de Control y Mando están dispuestos unos pilotos de señalización, con sus respectivas etiquetas, cuando un motor o cinta está en funcionamiento, este piloto pasará a estar en color verde, si estuviera apagado, tendría color gris. Los encargados de expulsar

las cajas de la cinta de clasificación C1 a sus respectivas cintas de salida, son los cilindros neumáticos, la planta está compuesta por 3 cilindros: P1, P2 y P3, éstos al igual que las cintas, están representados en el Panel de Control, con sus pilotos y etiquetas correspondientes.

Para detectar los movimientos de las cajas, se tienen 8 sensores dispuestos en nuestra maqueta, el sensor SCP, como se dijo anteriormente, es el encargado de detectar la caja sobre la báscula, mientras que los sensores SC1, SC2, SC3 y SC4, son los encargados de detectar que las cajas están en sus respectivas cintas, y los sensores SP1, SP2, y SP3, son lo que detectan que las cajas están posicionadas en los pistones. Los cilindros P1, P2 y P3, expulsan las cajas a sus respectivas cintas de transporte.

En el Panel de Control, se encuentran los pulsadores de Marcha y Paro, encargados de accionar y parar la automatización. Se dispone también de un pulsador de Impulso Inicial, encargado de establecer las condiciones iniciales al programa automático en el arranque, así como también de reiniciar nuestra simulación, en caso de algún fallo. Se posee también de un *display* (pantalla virtual) analógico, donde se debe introducir el peso de cada caja, para que el resto del proceso sea automático. Se tiene un pulsador “Intro/Peso”, para activar la acción de introducir el peso en el *display*.

En este panel también se encuentran indicadores de los sensores de las cintas (SC1,..., SC4) y de los pistones (SP1,..., SP3), además de los indicadores de energización de los motores que mueven las bandas o cintas (M0,..., M4.)

V.11.1 Secuencia Lógica

- a) Revisar condiciones iniciales (Actuadores retraídos, motores apagados).
- b) Se oprime botón de arranque y enciende la cinta de pesaje.
- c) Cuando sensor de pesaje detecta caja, la cinta se detiene y se realiza el pesaje.
- d) Después de pesar se enciende de nuevo al igual que la cinta distribuidora.
- e) Cuando la caja pasa frente al SC1. La cinta de pesaje se detiene y la cinta distribuidora continua activa.
- f) Dependiendo del peso se realiza la clasificación:
 - a) Si el peso es menor a 5 kg:
 - ✓ Si se activa SP2 se detiene cinta distribuidora C1.
 - ✓ El actuador neumático extiende su vástago CP2 y enciende cinta 2.
 - ✓ Cuando SC2 detecta caja, detiene cinta 2 y termina proceso.
 - b) Si el peso es mayor a 10 kg.
 - ✓ Si se activa SP3 se detiene cinta distribuidora C1.
 - ✓ El actuador neumático extiende su vástago CP3 y enciende cinta 3.
 - ✓ Cuando SC3 detecta caja, detiene cinta 3 y termina proceso.
 - c) Si el peso es mayor a 5 kg y menor a 10kg.
 - ✓ Si se activa SP4 se detiene cinta distribuidora C1.
 - ✓ El actuador neumático extiende su vástago CP4 y enciende cinta 4.

- ✓ Cuando SC4 detecta caja, detiene cinta 4 y termina proceso.

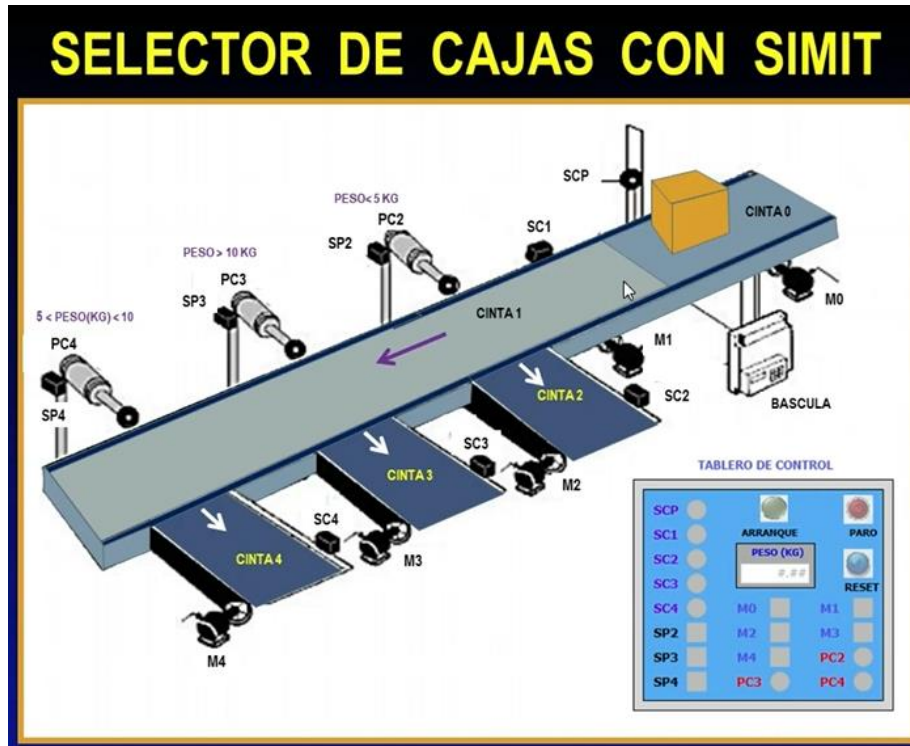


Figura V.252 Selector de cajas con SIMIT.

A continuación, se presenta el *Grafcet*, gráfico funcional de comandos etapa-transición, el cual nos describe la secuencia automática de la solución:

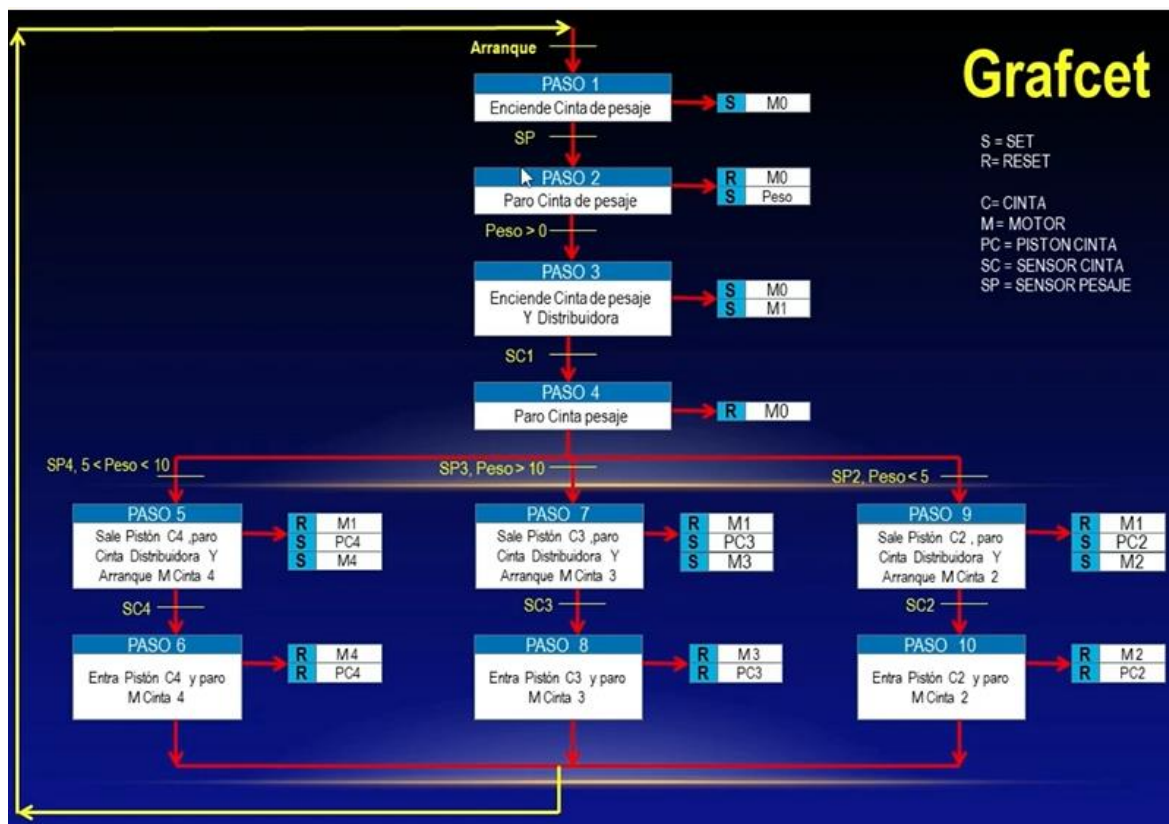


Figura V.33 Graficet descriptivo del proyecto

El *Graficet*, servirá de ayuda a la hora de implantar la simulación del sistema, ya que simular etapa por etapa facilita la comprensión de la automatización a realizar y siempre es más sencillo y didáctico visualizarlo de manera gráfica. A continuación, se observa en la tabla V.1 de variables del PLC:

<i>Nombre</i>	<i>Tipo</i>	<i>Dirección</i>	<i>Descripción</i>
<i>SCP</i>	Bool	<i>%I0.0</i>	Sensor Cinta Pesaje
<i>SC1</i>	Bool	<i>%I0.1</i>	Sensor Cinta Distribuidora
<i>SC2</i>	Bool	<i>%I0.2</i>	Sensor Cinta 2
<i>SC3</i>	Bool	<i>%I0.3</i>	Sensor Cinta 3
<i>SC4</i>	Bool	<i>%I0.4</i>	Sensor Cinta 4
<i>SP2</i>	Bool	<i>%I0.5</i>	Sensor Pistón 2
<i>SP3</i>	Bool	<i>%I0.6</i>	Sensor Pistón 3
<i>SP4</i>	Bool	<i>%I0.7</i>	Sensor Pistón 4
<i>Arranque</i>	Bool	<i>%I1.0</i>	Botón de Arranque
<i>Paro</i>	Bool	<i>%I1.1</i>	Botón de Arranque
<i>Reset</i>	Bool	<i>%I1.2</i>	Botón de Reset

<i>Peso</i>	Real	<i>%QD2</i>	Peso en kg.
<i>M0</i>	Bool	<i>%Q0.0</i>	Motor Cinta 0 o de pesaje
<i>M1</i>	Bool	<i>%Q0.1</i>	Motor Cinta 1 o de distribución
<i>M2</i>	Bool	<i>%Q0.2</i>	Motor Cinta 2
<i>M3</i>	Bool	<i>%Q0.3</i>	Motor Cinta 3
<i>M4</i>	Bool	<i>%Q0.4</i>	Motor Cinta 4
<i>PC2</i>	Bool	<i>%Q0.5</i>	Pistón Cinta 2
<i>PC3</i>	Bool	<i>%Q0.6</i>	Pistón Cinta 3
<i>PC4</i>	Bool	<i>%Q0.7</i>	Pistón Cinta 4

Tabla V.1 Variables del PLC, sin considerar las variables internas (marcas).

V.11.2 Desarrollo en SIMIT

- Realizar el acoplamiento entre *TIA Portal* y *SIMIT*, empleando *PLCSim Advanced* (explicado previamente en la sección V.10.1).
- Diseño del *CHART* y configuración básica.
- Diseño del tablero de control.
- Diseño de cajas o productos.
- Animaciones del producto.
- Diseño de sensores.
- Demostración del proceso (co-simulación).

V.11.3 Programación en el software TIA Portal

El programa se puede realizar ya sea en el *TIA Portal* o en el *SIMATIC STEP7*, en este ejemplo se usará el primero y el lenguaje de programación elegido es KOP (escalera). Es muy recomendable antes de empezar a programar directamente, identificar el número y tipo de variables que se van a emplear, es decir, las entradas, marcas, salidas, e instrucciones como temporizadores, contadores, etc. que se necesitan, para lo cual primeramente se dan de alta en la Tabla de Variables Estándar del PLC en el software *TIA Portal*, ya que de esta tabla se exportarán los datos a *SIMIT*, proceso que se explicará más adelante.

Nombre	Tipo de datos	Dirección	Rema...	Acces...	Escrib...	Visibl...	Supervis...	Comentario
SCP	Bool	%I 0.0						Sensor Cinta Pesaje
SC1	Bool	%I 0.1						Sensor Cinta Distribuidora
SC2	Bool	%I 0.2						Sensor Cinta 2
SC3	Bool	%I 0.3						Sensor Cinta 3
SC4	Bool	%I 0.4						Sensor Cinta 4
SP2	Bool	%I 0.5						Sensor Pistón 2
SP3	Bool	%I 0.6						Sensor Pistón 3
SP4	Bool	%I 0.7						Sensor Pistón 4
Arranque	Bool	%I 0.0						Botón de Arranque
Paro	Bool	%I 1.1						Botón de Paro
Reset	Bool	%I 1.2						Botón de Reset
Peso	Real	%Q 0.0						Peso en kg
M0	Bool	%Q 0.0						Motor Cinta 0 o de pesaje
M1	Bool	%Q 0.1						Motor cinta 1 o de distribución
M2	Bool	%Q 0.2						Motor cinta 2
M3	Bool	%Q 0.3						Motor cinta 3
M4	Bool	%Q 0.4						Motor cinta 4
PC2	Bool	%Q 0.5						Pistón cinta 2
PC3	Bool	%Q 0.6						Pistón cinta 3
PC4	Bool	%Q 0.7						Pistón cinta 4

Figura V.34 Tabla de variables en TIA Portal.

Una vez configurado el hardware y completada la tabla de variables, se puede empezar a programar y posteriormente a compilar, a continuación, se presenta el hardware dado de alta en el proyecto en el *TIA Portal*.

1. Dar de alta el PLC:1511-1 PN con dirección Ethernet: 192.168.0.1; siendo una CPU con *display* (pantalla); memoria de trabajo 150 KB para código y 1 MB para datos; tiempo de operación con bits 60ns; concepto de protección de 4 niveles, funciones tecnológicas integradas: *Motion* Control, regulación, contaje y medición; *trace* integrado; controlador PROFINET IO; soporta RT/IRT, 2 puertos, MRP, protocolo de transporte TCP/IP, comunicación S7, servidor web, equidistancia, *routing* (ruteo); firmware V1.8.
2. Dar de alta los módulos de expansión: DI 16x24Vcd /DQ 16x24Vcd (16 entradas y 16 salidas digitales a 24 Vcd, bytes 0 y 1 para ambas).
3. Módulo de entradas analógicas: AI 4xU/I/RTD/TC ST (4 canales con direcciones: IW2, IW4, IW6 y IW8).

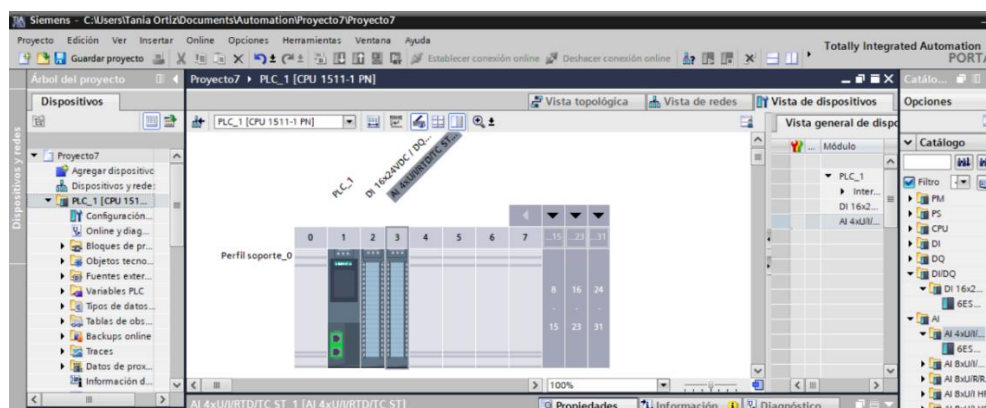


Figura V.35 Hardware del proyecto.

Después de compilar se retorna a la tabla de variables, en la pestaña de *variables* que se encuentra en la barra de *símbolos*, se da clic en el tercer icono para exportar, se elige *exportar variables* y se pulsa en los puntos suspensivos para elegir donde guardar nuestro archivo a exportar, el cual será un archivo con extensión: *xlsx* (archivo de Excel), después se da clic en *aceptar* y se espera al texto que diga: *Exportación finalizada correctamente*, eso es todo lo que se hará en *TIA Portal*.

V.11.4 Generación de un Proyecto de Simulación – SIMIT

Se ejecuta el Software *SIMIT Demo*, una vez que este se inicia, se muestra una ventana de diálogo, en ella los proyectos ya existentes se pueden abrir, al igual que se pueden iniciar nuevos proyectos, seleccione la opción *Create a new project* para crear un nuevo proyecto, y en la opción *Target folder* se especifica una carpeta donde se requiera almacenar el proyecto. Si ya se ha iniciado *SIMIT* sin hacer estos pasos, en la mencionada caja de diálogo en ella también se pueden abrir en el menú: *Project | New Project* y dar clic en *Create*.

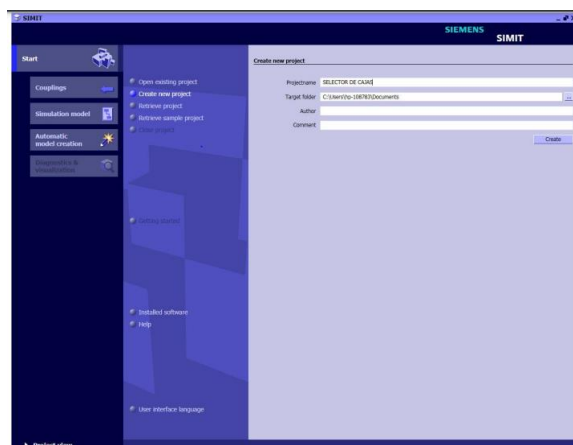


Figura V.36 Generación de nuevo proyecto.

V.11.5 Conexión con PLCSIM

En este ejemplo se cargará el programa para validarlo en co-simulación con el *SIMIT* en un simulador llamado *PLCSIM Advanced*. La carga en el simulador *PLCSIM Advanced* se detallará más adelante. Para establecer una nueva conexión, se da clic en *Add coupling* y después se selecciona *PLCSIM Advanced*.

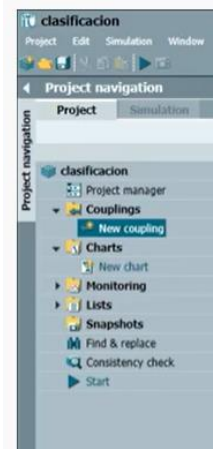


Figura V.37 Nueva conexión.

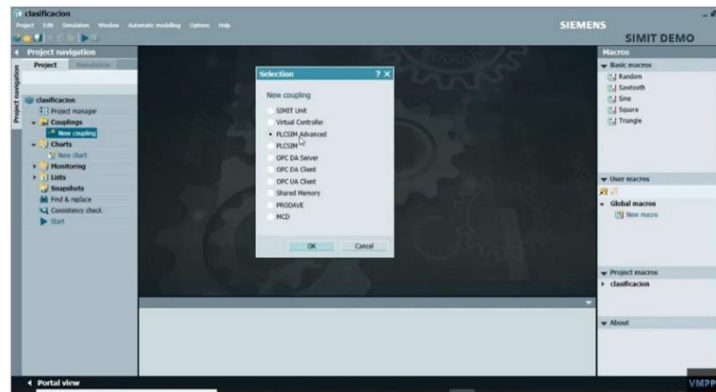


Figura V.38 Tipo de conexión nueva.

Para agregar las variables es necesario importar el archivo Excel que se creó anteriormente, para ello hay que dar clic en *Import*, en el menú de símbolos de la ventana *PLCSIM Advanced*; en la misma ventana no seleccionar la opción *All*.

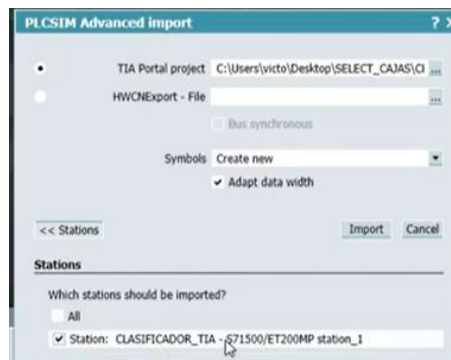


Figura V.39 Caja de diálogo de *PLCSIM Advanced Import*, Importación de variables.

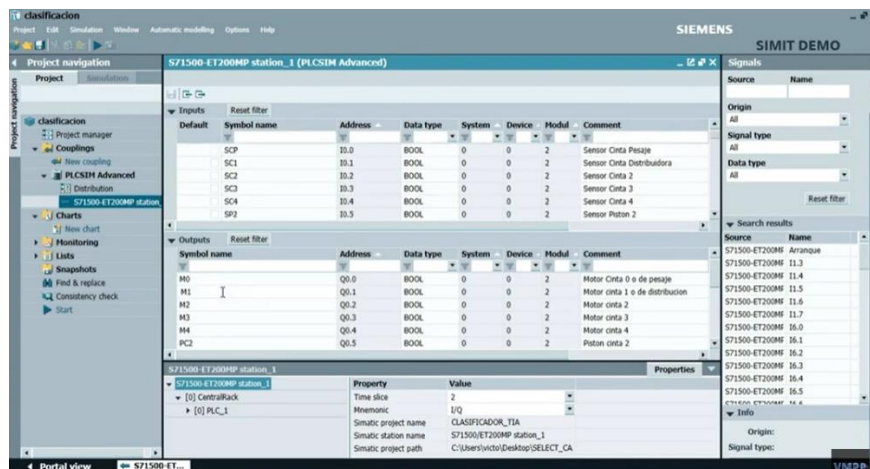


Figura V.40 Variables importadas.

Se recomienda que se guarden los avances realizados hasta el momento, haciendo clic en *Save all*. Para este punto ya se pueden observar las señales binarias y análogas, las cuales se pueden activar o desactivar desde *SIMIT* y/o desde el *PLCSIM Advanced*. Las señales y direcciones no deben modificarse en *SIMIT*, ya que la simulación provocará error. En el acoplamiento podemos cambiar el nombre del proyecto de donde se importan las variables del *TIA portal*, le dejaremos el nombre de **PLCSIM**.

Al comenzar la simulación, el intercambio de datos inicia estableciéndose inmediatamente. Para hacer una comprobación se puede iniciar la simulación, haciendo clic en *Start* en la barra de símbolos, una vez iniciada la simulación se pondrá de color naranja el título de la ventana del proyecto; si, por ejemplo, si se activa la salida Q0.7 en el *PLCSIM Advanced*, verá la reacción también en *SIMIT* y viceversa, o activar una entrada en *SIMIT SCE*, por ejemplo: I0. 7 se verá la reacción en el *PLCSIM Advanced*.

V.11.6 Construcción del Panel de Control

Aunque sea posible controlar directamente y supervisar todas las señales del E/S directamente desde la ventana de conexiones *Coupling*, como se ha visto en el apartado anterior, por regla general se quiere tener acceso a algunas señales que facilite el entendimiento del proceso, por lo que en una imagen en 2D y paneles de control (HMI: Interfaz Hombre Máquina) facilita la co-simulación y por lo tanto la comprensión de la solución en automatización del proceso. Con este fin, abrir la carpeta *Chart* y hacer clic en *New Chart*, se nombra como se deseé, en este caso el nombre que se ha elegido es el de *MAQUETA*.



Figura V.41 Creación de un chart.

Para que la maqueta y a su vez la simulación sea más real, se colocará una imagen de fondo, con el dibujo de la estructura de la máquina la cual se quiere automatizar. Para ello se debe mirar en la Ventana de Propiedades, y dar clic en los tres puntos (...) en *Background image* y buscar el archivo donde está ubicada la imagen. Es importante mencionar que en la versión demo de *SIMIT*, los gráficos para realizar las simulaciones son limitados; es decir sólo contamos con una cantidad limitada de variables y con 45 minutos para co-simular.

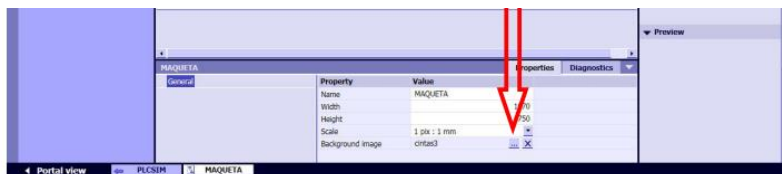


Figura V.42 Cambiar fondo de chart.

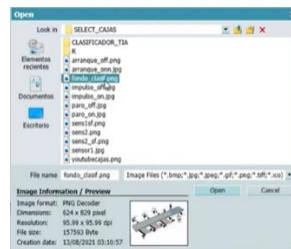


Figura V.43 Imagen a seleccionar.

Ahora se verá en la ventana que estaba en blanco, la imagen de fondo que se haya elegido. A la derecha, se tiene la Ventana de Herramientas con las etiquetas de *“Components”*, *“Controls”*, *“Graphic”*, *“Projects”* y *“Signals”*, con la opción de *Signals* se abre nuevamente una ventana para ver las variables importadas del *TIA Portal*.



Figura V.44 Ventana de herramientas.

Para crear el chart de control, primero se necesita dar clic en la etiqueta de *Controls*, luego se arrastra un *Binary display* desde su etiqueta hasta el chart, el piloto de señalización tiene una forma cuadrada inicialmente.



Figura V.45 Arrastre de binary display.

En este proyecto, conviene que los pilotos de señalización sean redondos, por lo que se selecciona el elemento y en la *ventana de propiedades*, se hace clic en *View* (vista), para cambiar la forma, se abre el menú desplegable de *Shape* y se elige *Round* (redonda). También se puede definir el color cuando el elemento está activado o desactivado, en este proyecto cuando el display este en modo off el color será gris, y cuando este en modo “On” el color será verde. También se puede cambiar el tamaño al elemento, haciendo clic y arrastrando uno de los 8 puntos azules que tiene alrededor.

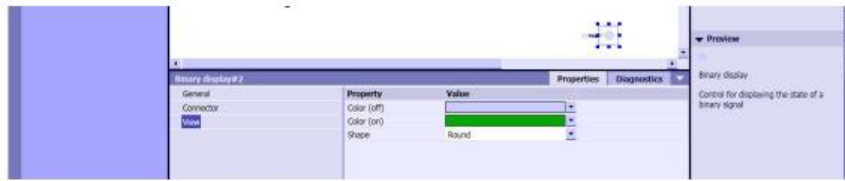


Figura V.46 Cambio de propiedades del display.

Para configurar la señal que lo activará, dentro de la misma Ventana de Propiedades, se da clic en *Connector*, donde *Name*, se deshabilita la imagen del ojo, la cual se tachará y se habilitará el cuadro de *Signal*. En el borde derecho de la pantalla, Ventana *Herramientas*, se da clic en *Signals*, y se busca la señal que corresponda, en este caso será *PLCSIM* de *Source* (Siempre será así) y *C0* como *Name*, *PLCSIM – C0* se arrastra hacia el cuadro en blanco de *Signal* y con esto se tendrá la conexión entre este elemento y la señal; y el elemento en la Ventana Maqueta, se puede ver como aparece sólo como un círculo gris.

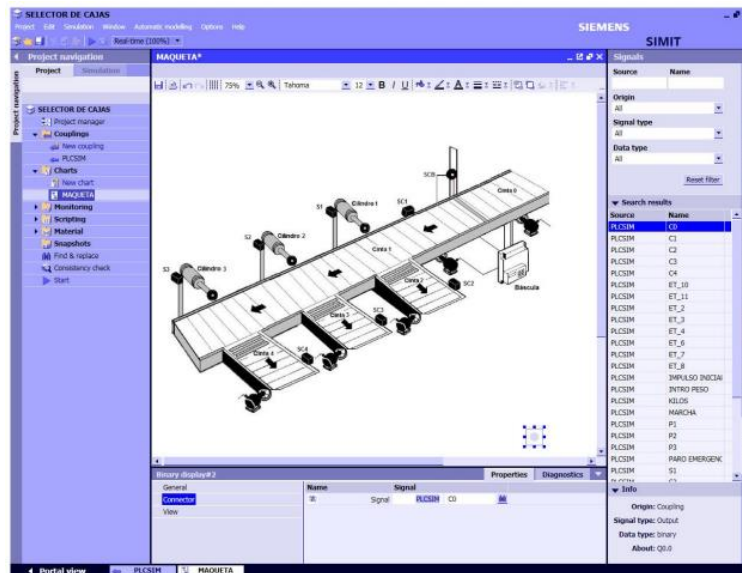


Figura V.26 Realización del conexionado entre elemento y señal.

Como este proyecto tiene 8 salidas, y por lo tanto se desea tener 8 display binarios y saber cuándo están activados o no, se procede a etiquetar cada piloto de señalización; en la pestaña *Graphic* en el borde derecho del programa, se da clic una vez en *Text*, con el ratón en la ventana Maqueta, ahí el cursor será una cruz, se determina el tamaño del cuadro de texto, se escribe lo que se desea, en este caso “C0”, se le da el tamaño de fuente correcto, y se pone cerca del display binario. Con esto se tiene una etiqueta y un piloto de señalización.

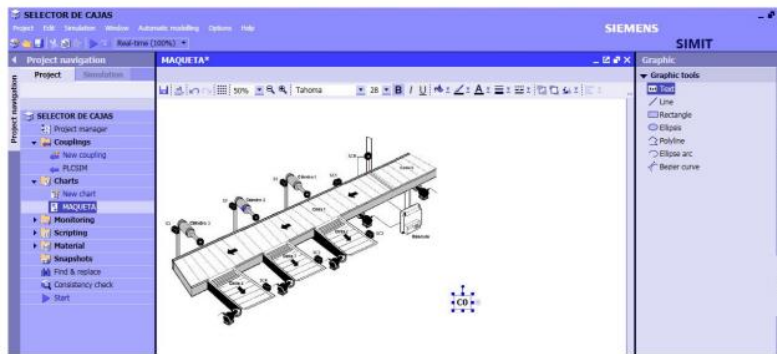


Figura V.27 Creación de un cuadro de texto.

En este proyecto, es necesario hacer 8 displays binarios y etiquetas, por lo que se repetirán los anteriores pasos, pero con las diferentes direcciones, y cambiando el nombre de las etiquetas. Ahora se tiene que agregar a nuestro chart todas las entradas, como pulsadores, finales de carrera, etc. se va a realizar el pulsador de marcha. en la ventana de herramientas, en *Controls*, en la zona de *Input*, se selecciona *Pushbutton with image* (pulsador con imagen), ya que este pulsador se le quiere poner una imagen descargada con antelación, para que nuestro panel de control quede más presentable. A continuación, con el elemento seleccionado, en la ventana de propiedades, se hace clic en *View*, y en *Image* (off) y (on) se selecciona la misma imagen de pulsador de marcha en la ubicación donde la tengamos. Para hacer la conexión con una señal, como ya se sabe, en la misma ventana de propiedades, se abre *Connector* y se deshabilita el ojo, y se arrastra desde *Signals* la conexión *PLCSIM-MARCHA*, para el caso del pulsador de Marcha.

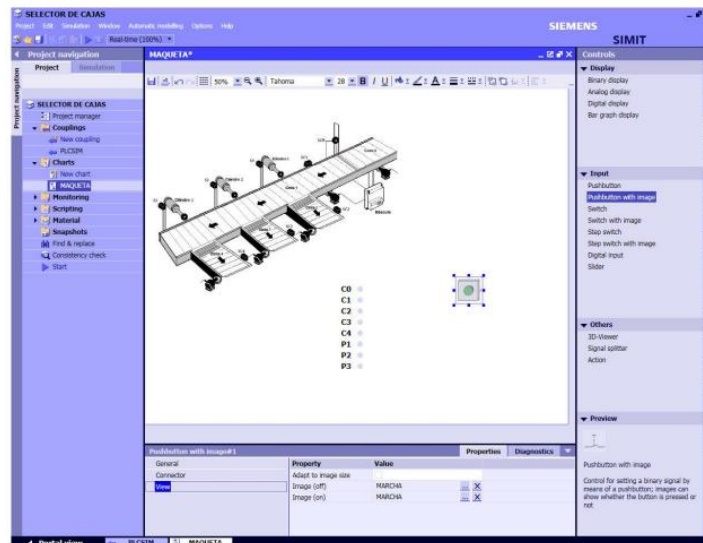


Figura V.28 Creación de un pulsador con imagen.

Se siguen los mismos pasos para crear el pulsador de Paro, pero con diferente dirección e imagen. Para realizar los demás pulsadores o sensores, se escoge dentro de la Ventana Herramientas y en *Controls* el elemento *Pushbutton*. Se conectará con las conexiones respectivas, como anteriormente, arrastrando desde *Signals* hasta el cuadro en blanco de *Connector*, al que previamente se ha deshabilitado el ojo. En estos pulsadores se puede o bien ponerle un texto interior, en la casilla *View* en *Text*, o hacer una etiqueta como previamente fue explicado.

Se tienen que realizar todos los pulsadores sin imagen, y quedarían según la imagen, la gran mayoría con sus etiquetas y otros con su texto interior, aunque eso es lo de menos, lo más importante es que se le den las direcciones adecuadas.



Figura V.50 Realización de todos los pulsadores.

El proyecto necesita un campo de entrada de datos analógicos, para introducir el peso que se desee de cada caja, ya que es una maqueta simulada. En la Ventana de Herramientas, pestaña *Controls* se podrá encontrar el elemento *Digital input*, que será el idóneo para el panel, tal y como dice en el *Preview* de la misma ventana: *Control para entrada numérica de valores analógicos o señal integrada*. Este elemento se configura de la misma forma que el anterior, por lo que se tendrá que hacer la conexión con su respectiva señal, en este caso la señal analógica *KILOS*.

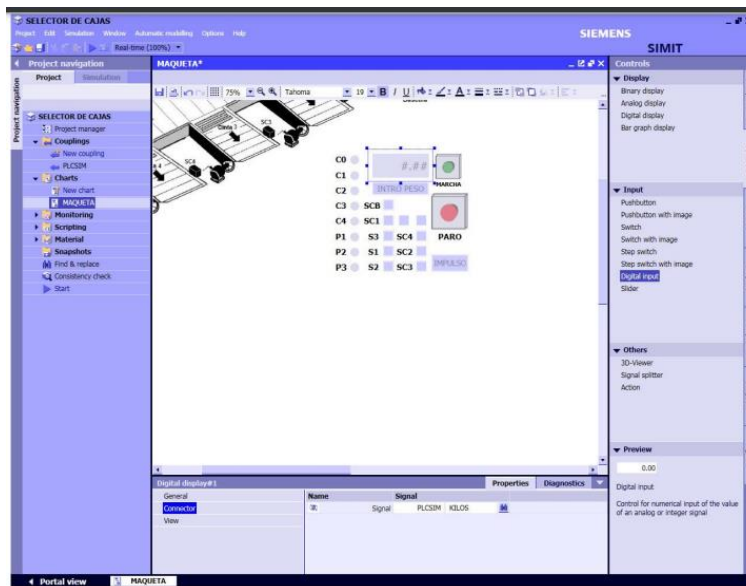


Figura V.51 Entrada numérica analógica.

Al tratar de hacerlo lo más real posible, se va a crear un panel de control, para que se asemeje lo máximo a los mandos de una maquina industrial. Para ello en la ventana herramientas, en la pestaña *Graphic*, se puede ayudar de los rectángulos y círculos, y se va jugando con la tonalidad de colores, para crear como la estructura metálica del panel de mandos. Una vez se haya creado estos dibujos, se podrá seleccionar todo y agruparlo, esta opción está en la ventana chart, en la barra de símbolos al final, hay tres puntos, se clic ahí, y sale la opción *Group* entre otras.

Una vez que se tenga esta figura agrupada, al llevarla a donde están todos los elementos de control, se encontrará con que la pieza metálica oculta los elementos, con la estructura seleccionada, se clic en *Send to background* (enviar al fondo), que se encuentra en la barra de símbolos.

Una vez puesta la imagen de la estructura en el fondo, se puede observar que las etiquetas se ven con el fondo blanco. Aquí hay dos opciones, o mientras se va creando, seleccionamos *No color* al color de fondo. O se cambia el color de todas en este paso.

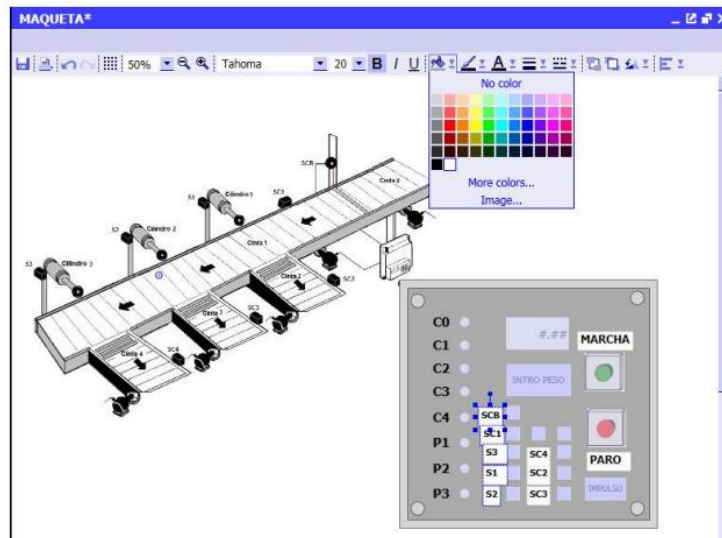


Figura V.52 Cambio de color de etiquetas.

Como se puede observar en la Figura V.53, el Panel de Control quedaría de la siguiente forma, donde se pueden encontrar todos los controles y los pilotos de visualización en el mismo conjunto. También se ha colocado alrededor de la máquina y en su ubicación idónea, cada piloto de señalización y pulsador correspondiente, tal y como se puede ver en la siguiente figura.

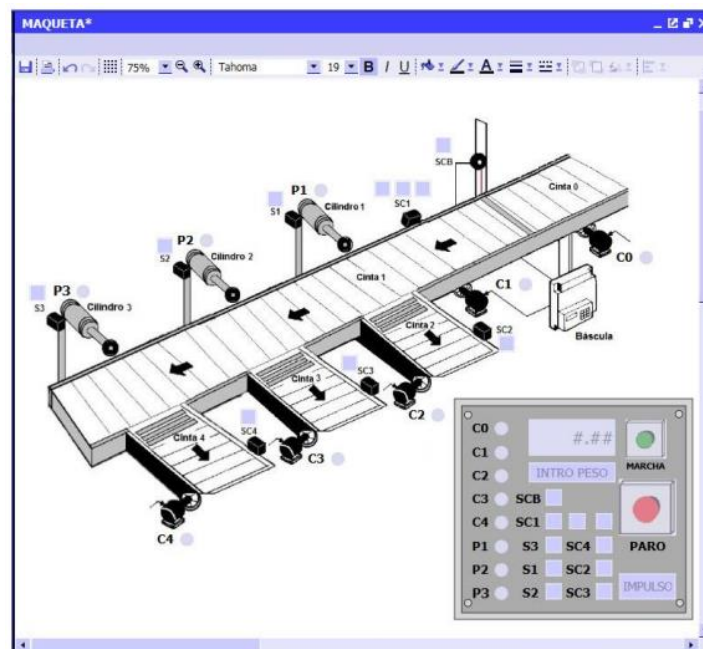


Figura V.53 Diseño final de la maqueta.

V.11.7 Construcción de las animaciones de los elementos

Lo primero que se hace es crear el prototipo de caja que se desee, con la ayuda de “Graphic” y todas sus herramientas, tales como: *Rectangle*, *Polyline*, *Line*, etc. Y en la ventana propiedades, donde aparece el apartado *Appearance*, se le dará el color, grosor de línea, etcétera. Una vez se finalice la caja, es el turno de darle movimiento. Para ello en la ventana propiedades del elemento, hay una opción, *Animations*, se hace clic en ella, y hay que elegir el tipo de animación que se desea dar. Todas las cajas tendrán dos tipos de Animaciones, *Movement* y *Visibility*.

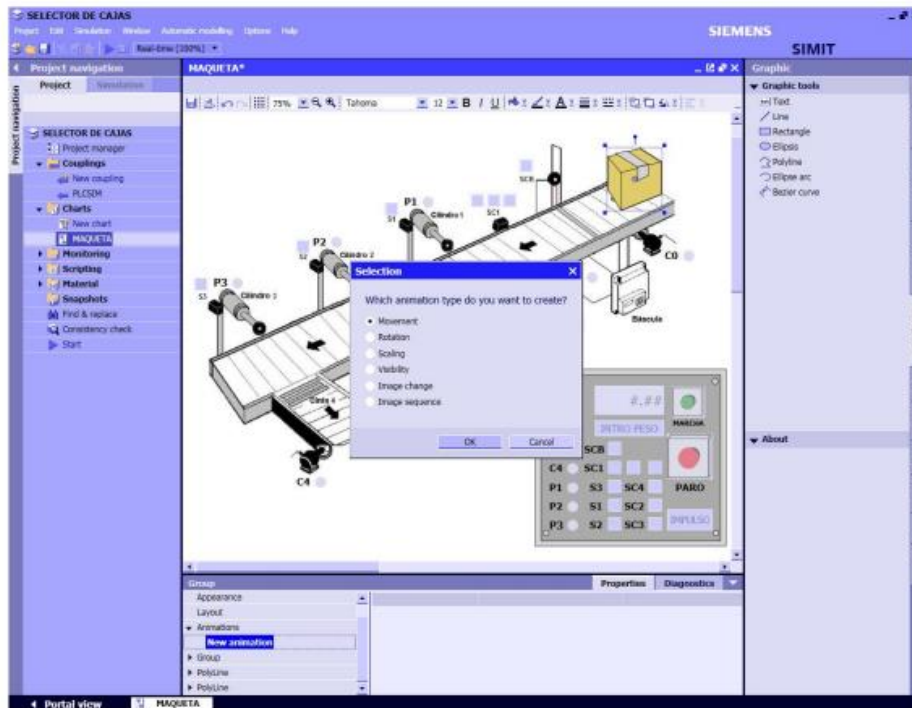


Figura V.54 Selección de animación de un elemento.

Se debe saber, que dotar la caja con la animación de visibilidad, es porque una sola caja es imposible que haga todo el recorrido de las cintas, y también es debido, a que la función que se usa para mover las cajas será una función *rampa* (la más idónea para este ejemplo), lo que quiere decir que, cuando llegue al final del recorrido que se le ha otorgado a la caja, y active el sensor que se ha configurado en dicha función, esta retornará por el mismo camino que avanzó.

Ahí es donde toma sentido la animación de visibilidad. Esto es un poco enrevesado al principio, pero conforme se vaya continuando en la explicación, se entenderá con facilidad.

Se explicará cómo definir los parámetros y como programar el ciclo de la caja (cajas), que pesa entre 0 a 5 kg, el inicio será cuando el operario acciona el pulsador de Marcha, y el final será cuando la caja sale de la cinta C2 y es detectada por el sensor SC2. Estará más claro con el tramo de grafcet de la mencionada caja.

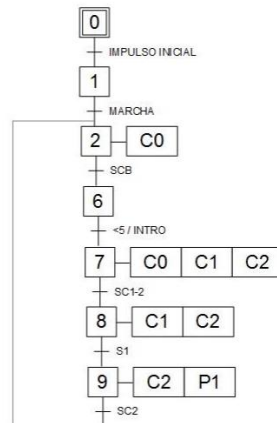


Figura V.55 Grafcet del tramo de la caja de 0 a 5 kg.

Una vez seleccionada la animación de movimiento para la caja, se ve el objeto en su posición inicial con unos colores casi transparentes, al igual que todo el entorno en la ventana chart, excepto una línea roja, que marcara el desplazamiento y la caja en su posición final.

Como se puede ver en el grafcet, la caja tiene 5 etapas, pero una etapa la caja está parada, por lo que tendrá 4 ciclos, o, mejor dicho, se tienen 4 cajas, con 4 movimientos. Se empezará desde el final hasta el principio, para superponer las cajas, y que el efecto sea de una caja en movimiento.

- ETAPA 9 – Inicia al detectar S1, en el cilindro P1, y finaliza cuando lo detecta el sensor SC2, en la cinta C2.

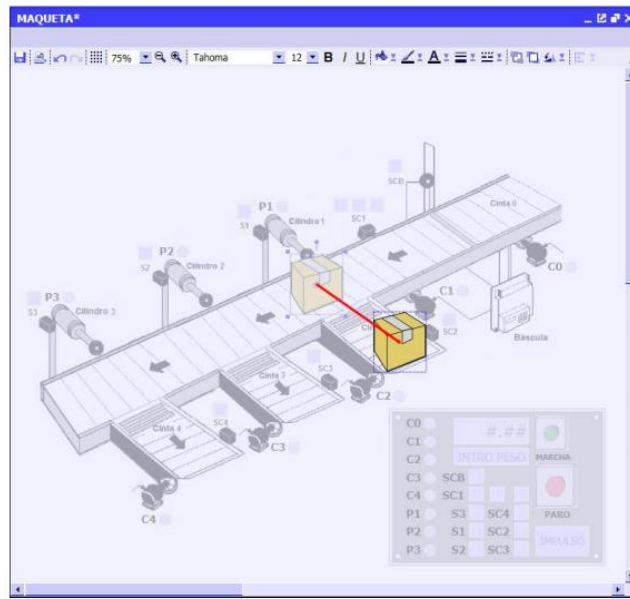


Figura V.56 Configuración de la animación Movimiento, Etapa 9.

- ETAPA 8 – Inicia cuando lo detecta SC1-2, y finaliza cuando lo detecta el sensor S1, en el Cilindro P1.

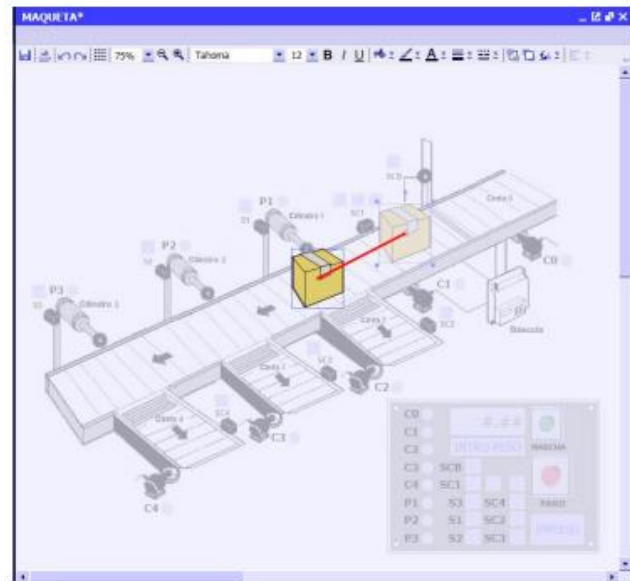


Figura V.57 Configuración de la animación Movimiento, Etapa 8.

- ETAPA 7 – Inicia cuando la Báscula finaliza el pesaje (<5kg), finaliza cuando entra en la cinta C1, y lo detecta SC1-2.

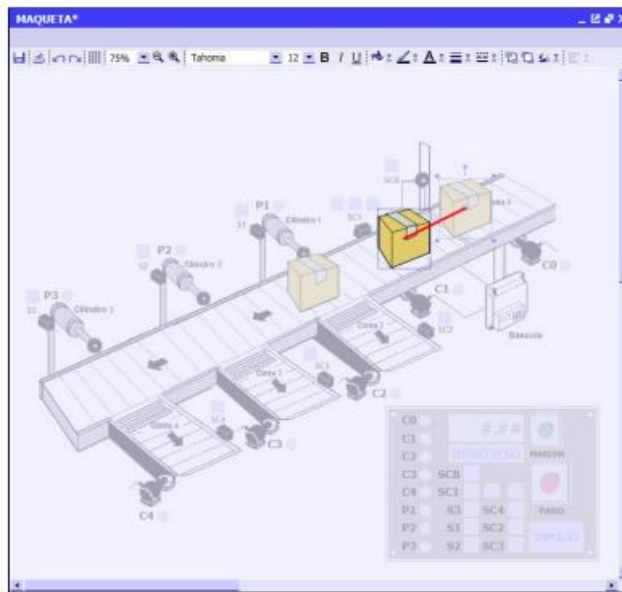


Figura V.58 Configuración de la animación Movimiento, Etapa 7.

- ETAPA 2 – Se inicia cuando el operario pulsa MARCHA, y finaliza cuando la caja activa el sensor SCB. Este ciclo será el único igual para todos, por lo que no habrá que repetirlo.

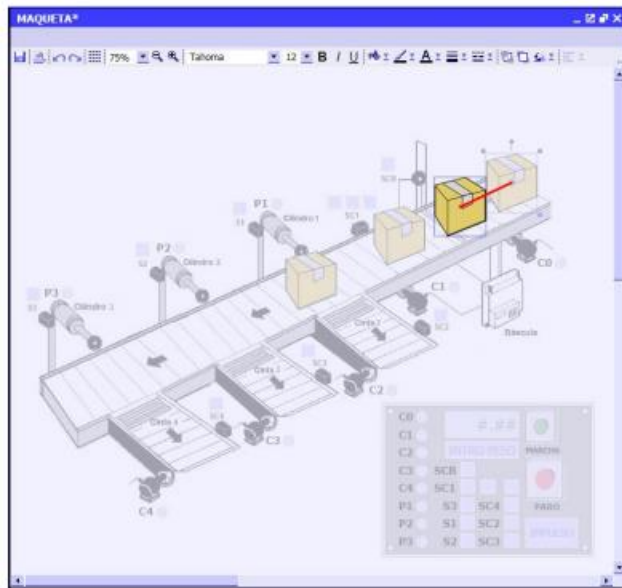


Figura V.59 Configuración de la animación Movimiento, Etapa 2.

Exceptuando el ciclo de la Etapa 2, los demás ciclos habrá que hacerlos para los otros dos intervalos de peso. Es decir, la caja que pese entre 5 y 10 kg, por ejemplo, los ciclos deberán iniciarse desde que la báscula detecta su peso hasta que llega a la cinta C4, y es detectado por el sensor SC4. Al igual que la caja que pese más de 10 kg, los ciclos deberán iniciarse desde que la báscula detecta su peso hasta que llega a la cinta C3, y es detectado por el sensor SC3.

Ahora es el momento en que se le da la animación de visibilidad a todas las cajas, para ello en la ventana propiedades del elemento, hay una opción, *Animations*, se hace clic en ella, y se elige *Visibility*. Este se hace con todas las cajas.

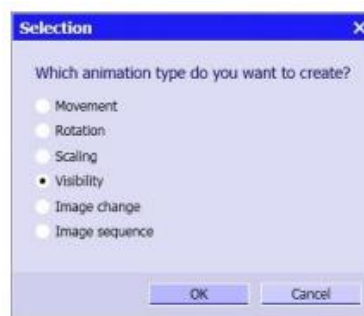


Figura V.60 Selección de animación de visibilidad.

V.11.8 Creación de funciones del sistema

En la Ventana de *Project*, en la carpeta *Chart*, se clic *New Chart*, y se nombra, en este caso se ha puesto *FUNCIONES MOVIMIENTO*, para que se entienda, aunque estas funciones no solo valdrán para la animación movimiento, ya que también actuarán sobre la animación visibilidad.



Figura V.61 Nuevo chart FUNCIONES MOVIMIENTO.

La función *Ramp* se puede encontrar en la Ventana de Herramientas, en la librería *Components / STANDARD* y dentro de estas, en elementos *AnalogExtended*. Arrastre este componente a su Chart.

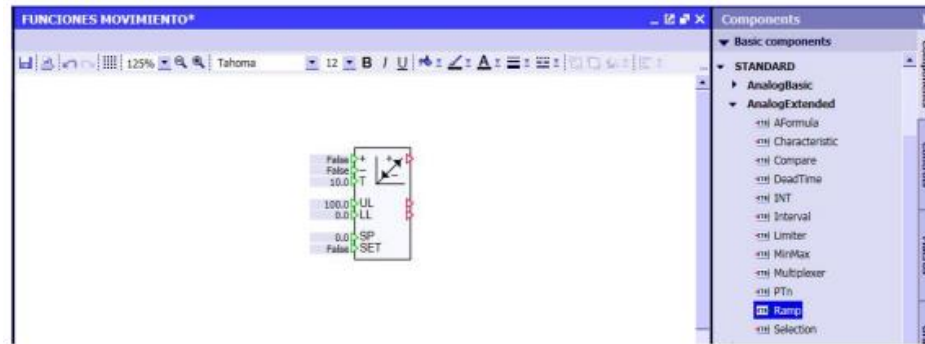


Figura V.62 Función ramp.

Este componente calcula el valor en funcionamiento de la rampa mediante el tiempo configurable “T” del límite más bajo “LL” al límite más alto “UL”, si a la entrada (+ / arriba) se le asigna el valor “1” y viceversa, si a la entrada (- / abajo) se le asigna el mismo valor “1”. Si uno de estos dos límites está conectado en las salidas del componente.

Esta función se activará con una sola señal o pulsador, y se conectará directamente a la entrada, y la segunda entrada tendrá conectada un negador.

La señal de entrada, la cual activa la animación de las cajas, será *PLCSIM C0*, esta ira a la entrada (+), por lo que se irá a la librería *Signals* y se arrastrará dicha señal con el botón del teclado *Shift* pulsado en todo momento, se pegará una señal con una etiqueta de color verde.

La señal de negación que se debe colocar en la entrada (-) y conectarla en paralelo con las otras dos entradas, será *NOTc* del *STANDARD | BinaryBasic*.

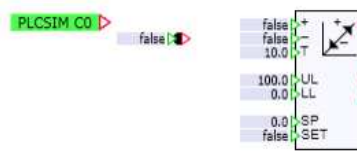


Figura V.63 Conexiones función ramp.

Se mueve la negación a la entrada (-) de la rampa, de modo que el triángulo rojo de la negación se superponga con el triángulo verde de la rampa. De esta manera, las dos conexiones se relacionan una con otra. Este atento, y agarre el componente en sí, y no su conexión.

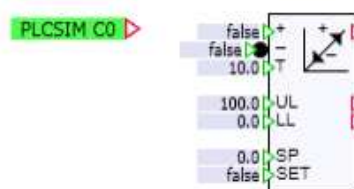


Figura V.64 Función rampa con entrada negativa

Las dos conexiones que todavía faltan se pueden hacer con las líneas de la señal. Se hace clic en el triángulo rojo de la señal *PLCSIM C0* y luego se hace clic en la entrada (+) de la rampa, y ahora para conectar en paralelo la entrada (-), se clic en el triángulo verde al lado de *false* en la *NOTc* y después clic en el triángulo totalmente rojo de la salida *PLCSIM C0*.

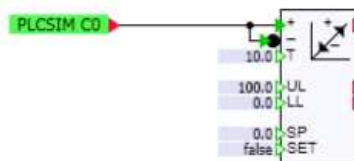


Figura V.65 Conexión de entrada en la función rampa.

Ahora, se necesita que cuando esta función llegue al máximo de la rampa, active un sensor o una señal. Para que se entienda, se pondrá un símil, por ejemplo, la rampa va de 0 a 10, tiene una señal la cual, cuando la rampa llega a 10 activa un final de carrera, y tiene otra señal, que cuando la rampa está en 0, está activando otro final de carrera.

Las señales anteriormente citadas, son los dos triángulos rojos abajo a la derecha de la función rampa, nosotros usaremos el que está más alto, que es el que activa o da una señal, cuando la rampa llega a su máximo valor.

En esta señal se conectará el sensor SCB de caja situada en la báscula, por lo que se irá a la librería *Signals* y arrastrar la señal *PLCSIM SCB* con el botón del teclado *Shift* pulsado en todo momento, se arrastrará una señal con una etiqueta de color verde. Entonces se une el conector de la periferia conectándolo de esta manera con la rampa.

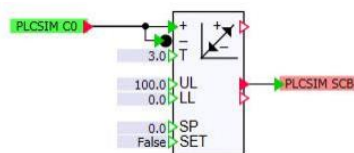


Figura V.66 Conexionado completado función rampa.

Finalmente, se baja el valor por defecto del tiempo de ejecución de la rampa, en este caso se puso 3 segundo, pero va a ir dependiendo de cada trayecto de las cajas y lo largo que sea cada tramo. Esto se puede hacer directamente en la función, haciendo doble clic en el campo grisáceo de la entrada T y sobrescribiendo el valor 3 (en este caso).

Y con ayuda del graficet, ya se tienen los conocimientos suficientes para realizar todas las funciones rampa del ciclo de la caja menor de 5 kg. Como son 3 rampas más las que hay que añadir, con sus diferentes tiempos, dependiendo el tramo de cada una.

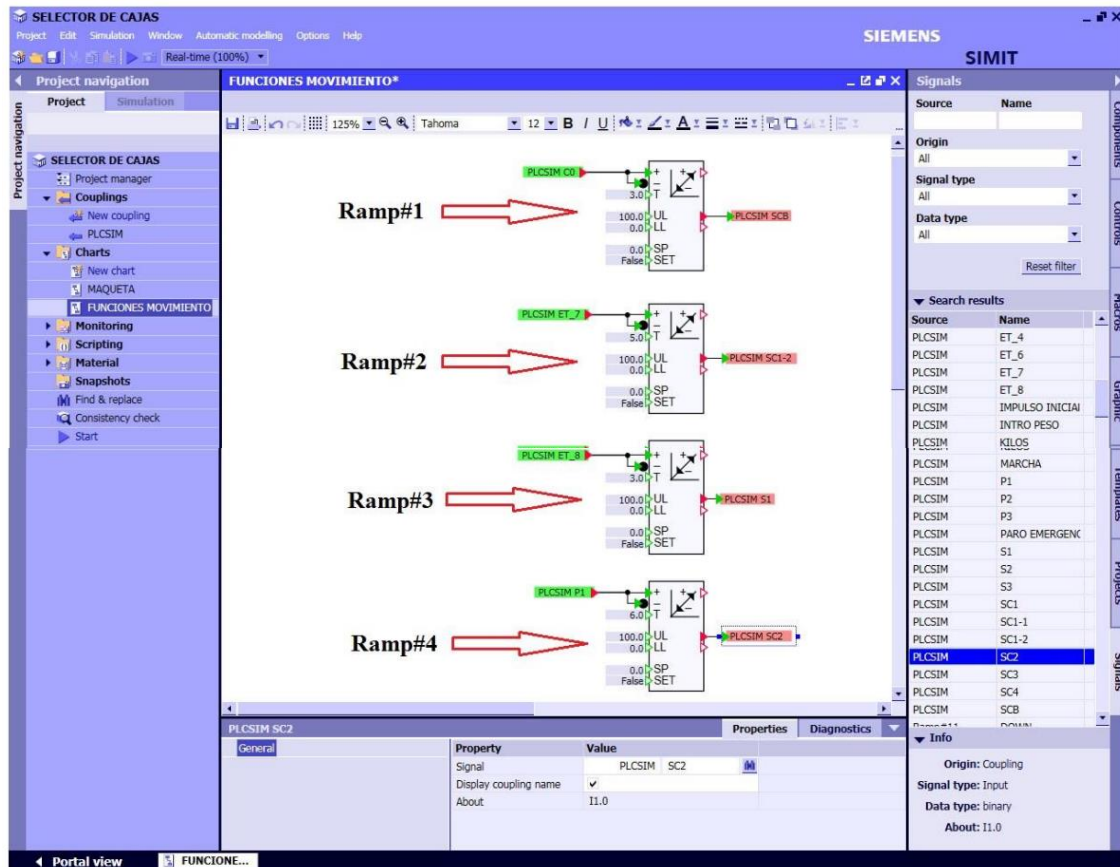


Figura V.67 Funciones rampa para el ciclo “Caja <5kg”.

V.11.9 Implantación de las Funciones en las Animaciones

Ahora se procede a aprender como instaurar cada una de las funciones en cada una de las cajas, y ponerle las diferentes funciones *Ramp* a cada animación. A cada caja le corresponde una rampa diferente, eso tiene que estar muy claro, y no confundirse a la hora de programarlo, ya que, por lo contrario, la simulación no hará lo deseado. Por lo que la programación será la siguiente:

- Ramp#1 → Primera caja
- Ramp#2 → Segunda caja
- Ramp#3 → Tercera caja
- Ramp#4 → Cuarta caja

También se ha de tener claro que, para todas las animaciones de movimientos, se escogen las variables $Ramp\#N - Y$, donde N será el número de función, dependiendo de que caja se quiera mover. Para darle el valor de $Ramp\#1$ al movimiento de la primera caja, se clicca sobre la caja, *Animations / Movement*, y en la ventana herramientas, se va a *Signals*, y se arrastra hasta la Ventana Propiedades, en *Value*, la señal $Ramp\#1 - Y$.



Figura V.68 Inserción función rampa en la animación movimiento.

Tener claro que, para todas las animaciones de visibilidad, se escogerán las variables $Ramp\#N - DOWN$, donde N será el número de función, dependiendo de que caja queramos mover. Para darle el valor de $Ramp\#1$ a la visibilidad de la primera caja, se hace clic sobre la caja, *Animations / Visibility*, y en la ventana herramientas, se va a *Signals*, y se arrastra hasta la Ventana Propiedades, en *Value*, la señal $Ramp\#1 - DOWN$.

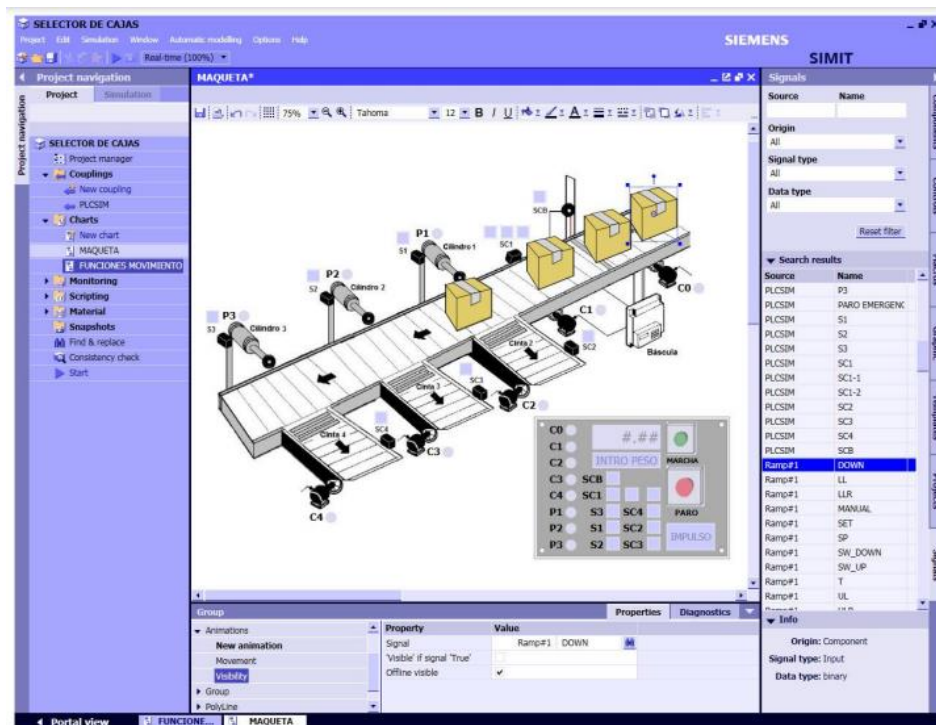


Figura V.69 Inserción función rampa en la animación visibilidad.

Se debe hacer estos pasos para todo el ciclo de “caja <5 kg”. Una vez se haya finalizado, sería recomendable simular este tramo, para ver si la simulación actúa tal y como se desea, ya que la programación de los demás tramos es similar. Con ayuda del graficet ya se tienen los conocimientos suficientes para realizar todas las funciones rampa de todo nuestro proyecto son 6 rampas más las que hay que añadir, con sus diferentes tiempos, dependiendo de la longitud del tramo de cada una. Realizando estas 10 rampas, como nos muestra la figura que tenemos abajo, ya se habría finalizado de construir todas las funciones de nuestro selector de cajas.

V.11.10 Simulación planta virtual

Se abre TIA Portal, se carga el programa, se compila para asegurarnos que no existe ningún error, y se hace clic en *Iniciar Simulación*, se encuentra en la barra de menú rápido, como vamos a cargar el programa en un simulador, en este caso PLCSIM, se tendrán que definir los siguientes parámetros:

- Tipo de interfaz: MPI
- Interfaz: PLCSIM

Una vez insertados estos parámetros, se clic en *Iniciar Búsqueda*. Encontrará el Tipo de dispositivo: CPU 300 sin especificar, se selecciona y se clic en *Cargar*.

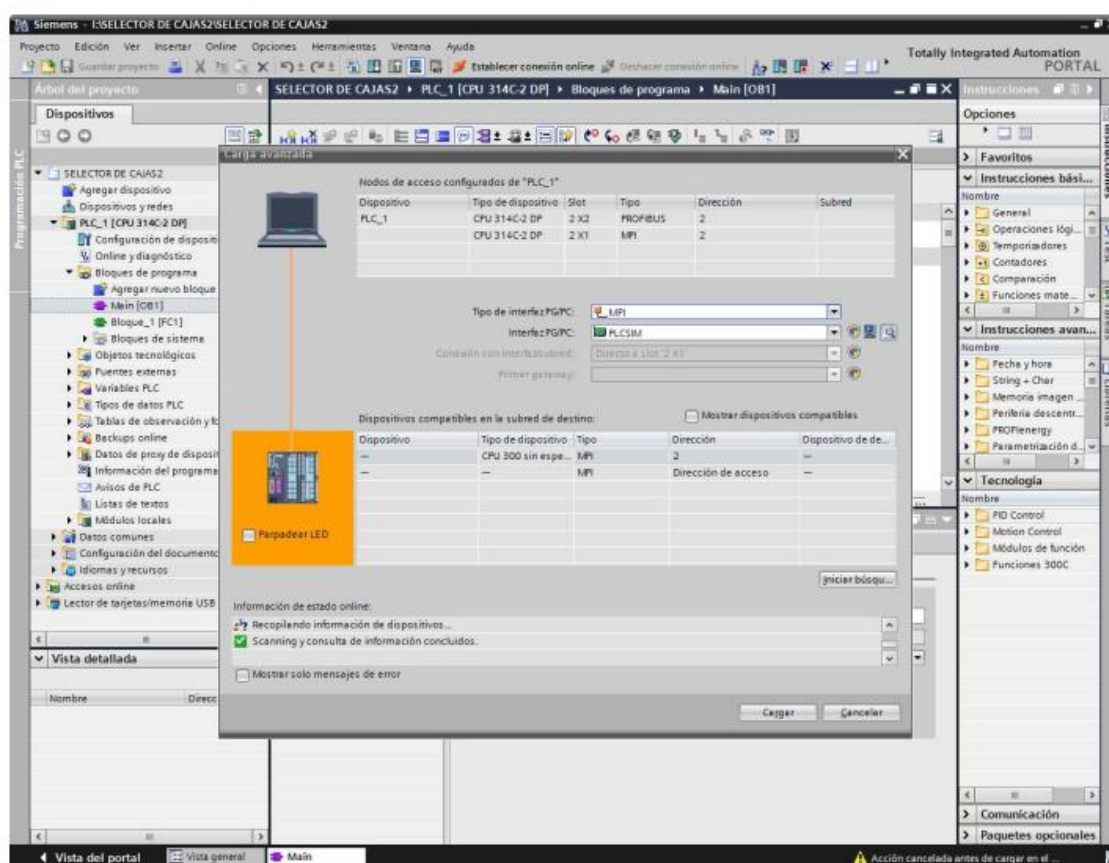


Figura V.72 Selección de dispositivo para cargar el programa.

Aparecerá una ventana de vista preliminar carga. Se define el parámetro: *Carga con coherencia*, si no estuviera por defecto, y se clic en *Cargar*.

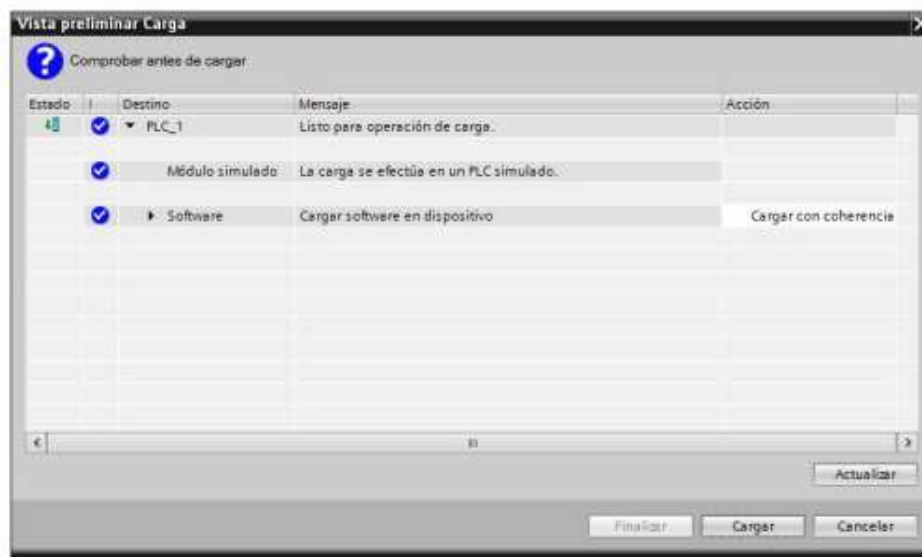


Figura V.73 Vista preliminar Carga.

Se abrirá el programa S7-PLCSIM1, se insertan nuestras entradas y salidas. En este caso tenemos 14 entradas digitales, es decir, desde I0.0 hasta I1.5, por lo que se insertan dos ventanas de entradas de 8 bits cada una. Una la se nombra IB0, donde irán los bits del 0.0 al 0.7. Y la otra IB1, que alberga los bits del 1.0 al 1.7.

También se tiene una entrada analógica, en este caso es del tipo Real, y la dirección es ID64 (es muy importante que la dirección sea la misma), y el formato numérico ha de ser “Regul:real”. Este programa tiene 8 bits exactos de salidas reales, por lo que con un QB es suficiente.



Figura V.74 Interfaz PLCSIM.

A continuación, se abre el SIMIT, se carga el programa, se abre nuestro Chart de simulación, en este caso *MAQUETA*, por medio de la barra de Símbolos se inicia la simulación, se clicca en *Start*. El software cambiará a color naranja.

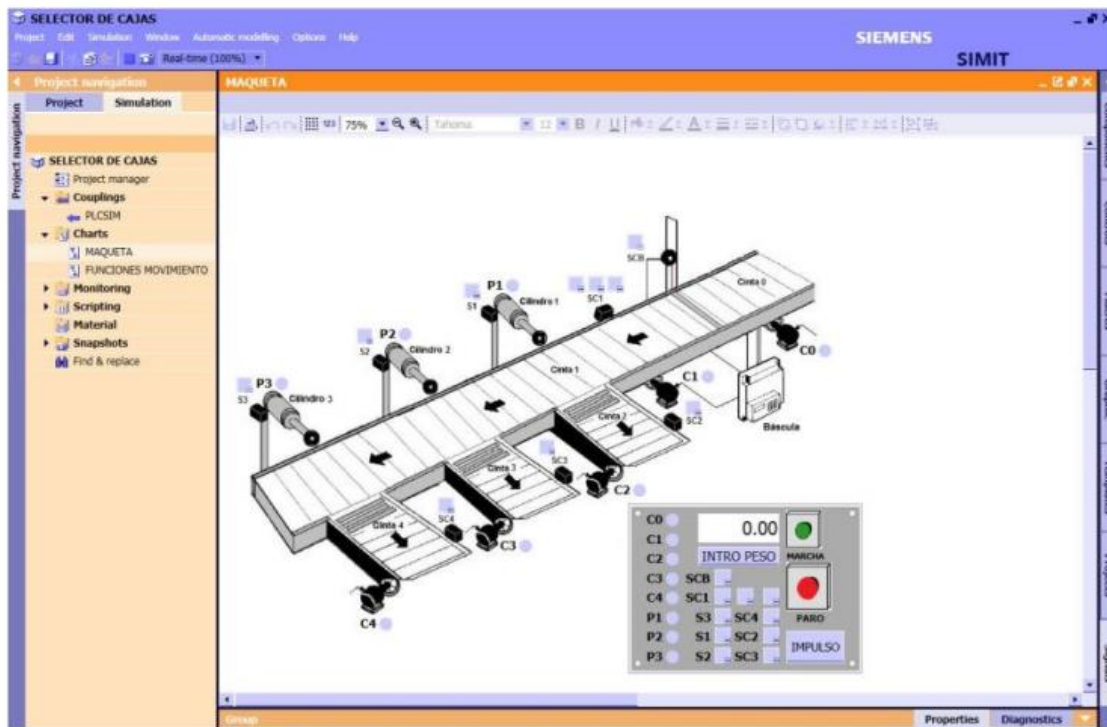


Figura V.75 Simulación iniciada.

Una vez se pulse el *IMPULSO INICIAL* y la *MARCHA*, una caja será alimentada por la cinta C0, llega hasta el sensor SCB, donde la báscula simula el pesaje, y espera que se introduzca un valor en kilogramos, una vez se escriba la cifra, habrá que pulsar *INTRO PESO*, dependiendo la cifra que se haya puesto, la caja ira para una cinta expulsora o hacia otra.

Cuando se introduce un peso de entre 0.01 a 5 kg, la caja es expulsada automáticamente por la cinta C2. Cuando se introduce un peso de entre 5.01 a 10 kg, la caja es expulsada automáticamente por la cinta C4. Cuando se inserta un peso de entre 10.01 a 15 kg, la caja es expulsada automáticamente por la cinta C3. En los anexos se puede ver el programa completo que se diseñó en TIA Portal.

V.11.11 Puesta en marcha virtual de SIMATIC/SIMOTION con hardware en el bucle (HiL)

Para realizar una puesta en marcha virtual, se necesita una imagen de la máquina real. Esta imagen se denomina gemelo digital de una máquina. Con la ayuda de un gemelo digital se puede simular y optimizar la interacción de los componentes individuales en el mundo virtual, sin tener un prototipo real. Para reducir los riesgos y el esfuerzo de la puesta en marcha real, la puesta en marcha virtual de una máquina ofrece una alternativa eficaz. Esto permite reducir el tiempo de comercialización y aumentar la flexibilidad, la eficiencia y la calidad.

STEP 7 y TIA Portal permiten crear un escenario "Hardware in the Loop" (HiL) para simular y validar el programa de usuario. Para el HiL, los componentes de hardware se despliegan en un entorno de simulación. Con NX Mechatronics Concept Designer (MCD), los fabricantes de máquinas pueden simular y probar los componentes mecánicos de su máquina en un entorno virtual. El comportamiento de los componentes activos, como los accionamientos o las válvulas, se emula con el software de simulación SIMIT. Por su parte, los dispositivos de campo de la planta real se emulan en la unidad SIMIT. Esta combinación ayuda a la preparación y a una puesta en marcha sin problemas. Además, estas herramientas permiten validar el concepto mecánico de la máquina y la interacción del sistema mecánico, el sistema eléctrico, el software y el programa de usuario, en una fase temprana de desarrollo de una planta.

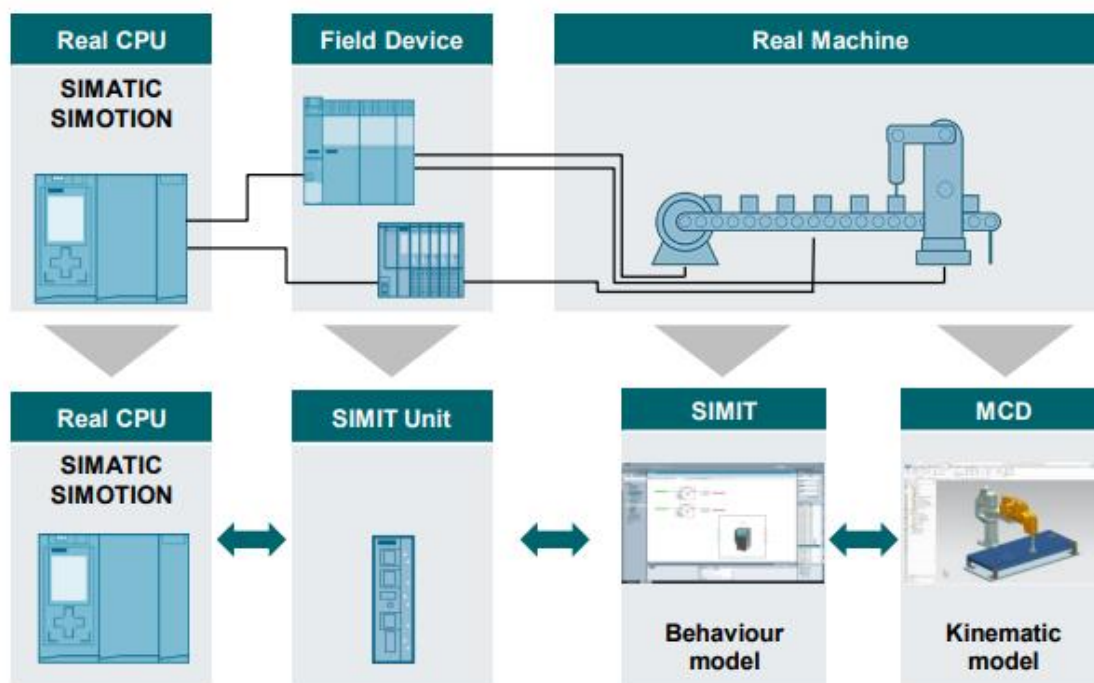


Figura V.76 Diagrama del funcionamiento.

V.11.12 Principio de operación

La siguiente figura muestra el principio de funcionamiento del ejemplo de aplicación.

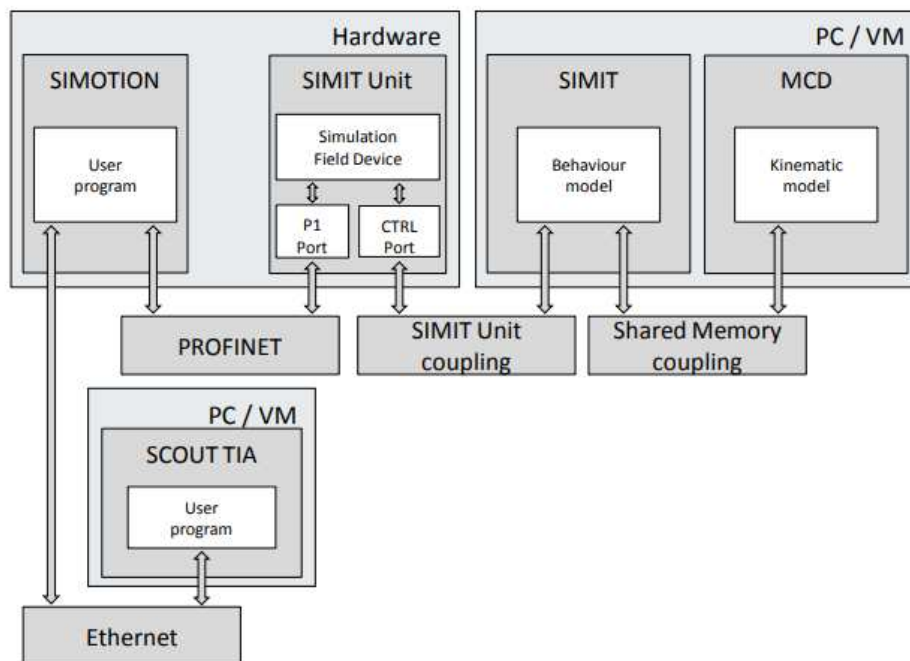


Figura V.77 Principio de operación.

Para explicar de mejor manera, se utilizará como ejemplo un proyecto, el cual controla una planta de transporte. Para realizar una prueba de funcionamiento completa, el programa se carga en un SIMOTION. La interacción con SIMIT y MCD permite validar el programa en el contexto de la planta. La siguiente figura muestra la planta de transporte con las asignaciones a los objetos tecnológicos asociados en el proyecto SCOUT.

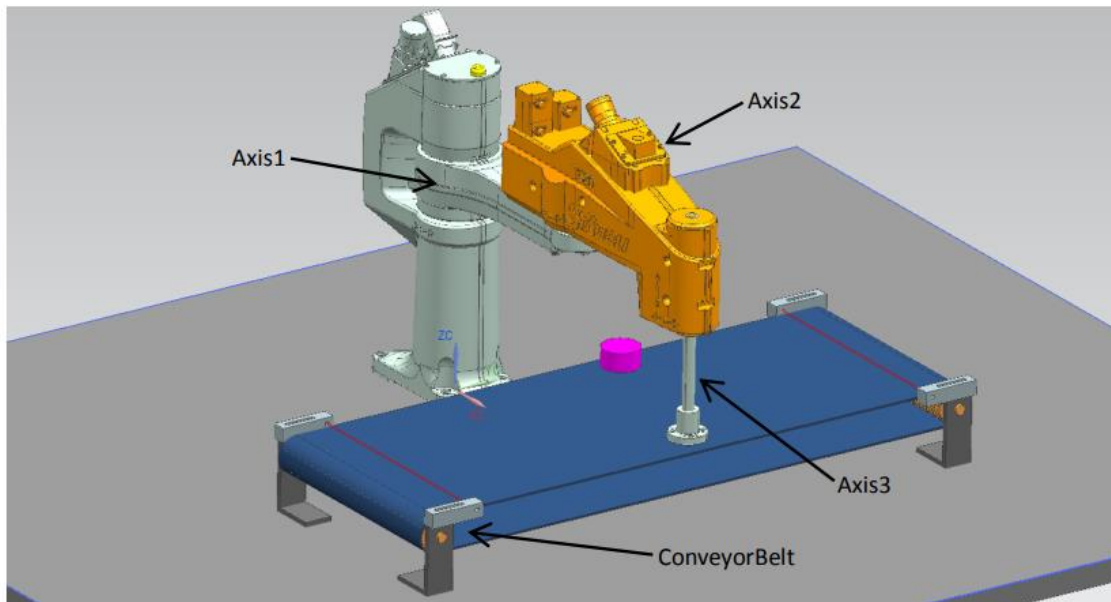


Figura V.78 Planta de transporte.

"Eje1", "Eje2" y "Cinta transportadora" se recorren mediante telegramas PROFIdrive. Estos ejes no son virtuales ni están en modo de simulación. Esto significa que estos telegramas PROFIdrive se interpretan y procesan durante la puesta en marcha. Para ello es necesario un modelo de comportamiento de los accionamientos en SIMIT. "Eje3" muestra una forma más sencilla de la puesta en marcha virtual. Este eje, en el proyecto TIA SCOUT, está en modo de simulación (para más detalles, véase Simulación de ejes en SCOUT).

Esto significa que no se requiere un modelo de comportamiento de los telegramas y que la posición real simulada del eje se copia en un programa a un área de direcciones de salida y se transfiere así al modelo virtual. Para ello, se crea un acoplador PN-PN en la vista del dispositivo. Con él se crean áreas de direcciones que pueden leerse y escribirse en SIMIT a través de la unidad SIMIT. El área de direcciones del acoplador PN-PN también se utiliza para las entradas y salidas digitales necesarias.

Como alternativa, se puede configurar un módulo de periferia distribuida para este fin. Si los ejes no se conmutan a la simulación, deben interconectarse con accionamientos externos a través de PROFINET. En este ejemplo de aplicación, todos los ejes están interconectados con accionamientos en el SINAMICS S120 externo.

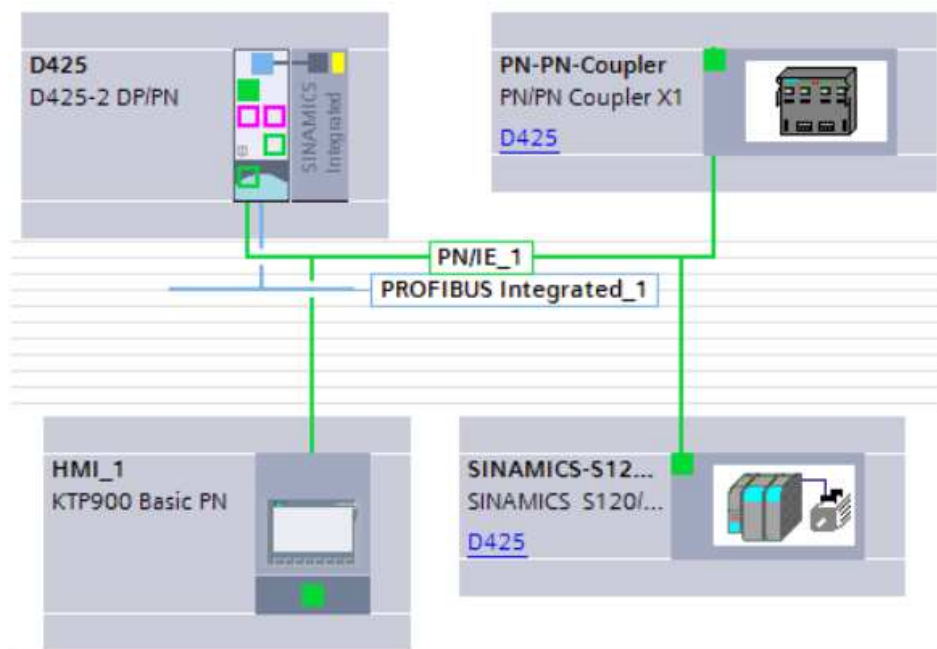


Figura V.79 Vista de red de la configuración del dispositivo.

La instalación de transporte a controlar está compuesta por una cinta transportadora y un robot SCARA. Son posibles tanto el modo de desplazamiento como el modo automático. Esto permite recorrer la cinta transportadora, así como los ejes 1-3 del robot SCARA. Para que el controlador pueda comunicarse con la planta de transporte, se definen las siguientes variables de entrada y salida:

Variable	Data type	Input/output
iboLightBarrierPickingPos	BOOL	Input
iboLightBarrierPlacingPos	BOOL	Input
iboBoxPicked	BOOL	Input
qboEnableVacuum	BOOL	Output
qr32Axis3ActualPosition	DWORD	Output

Figura V.80 Variables de entradas y salidas.

Los ejes no conmutados a simulación se comunican mediante telegramas PROFIdrive en las siguientes áreas de direcciones:

Variable	Data type	I/O address area
ConveyorBelt	PD_TEL1	260 ... 263
Axis1	PD_TEL105	264 ... 283
Axis2	PD_TEL105	284 ... 303

Figura V.81 Telegramas PROFIdrive.

La visualización de la planta se simula con WinCC Runtime. En WinCC Runtime, la planta idealmente se debe de controlar a través de una pantalla HMI y las posibles funciones deberían de ser: Accionar la cinta transportadora, desplazamiento de los ejes SCARA 1 – 3, activar el modo automático.

V.12 Unidad SIMIT

En el escenario "Hardware in the Loop", el programa de automatización se carga en el sistema de automatización real. Para ello, la unidad SIMIT simula las señales de E/S de los controladores y las señales de los dispositivos de campo, por ejemplo, actuadores y sensores. La unidad SIMIT constituye así la interfaz entre el hardware de automatización y el modelo de comportamiento en SIMIT. La conexión con PROFINET permite la simulación en tiempo real.

V.12.1 SIMIT

En SIMIT se modela el modelo de comportamiento de la instalación y los acoplamientos necesarios a otros programas. En este ejemplo de aplicación, se simula un accionamiento de tres ejes y una simple transferencia de posición. Para la simulación del accionamiento existen bloques de telegramas que emulan la comunicación y la máquina de estados de un accionamiento real. Esto significa que el programador no necesita hacer ningún cambio en el programa de usuario.

El área de entrada y salida del telegrama debe estar interconectada adecuadamente para permitir esta comunicación. Si la comunicación no debe realizarse necesariamente con telegramas PROFIdrive, también es posible una simple transferencia de posición. Para ello, el eje asociado en el TIA Portal se cambia a simulación.

Dado que las herramientas de software calculan con diferentes unidades base, las distintas variables se convierten en SIMIT. Para analizar el comportamiento de la planta en un entorno seguro, se pueden simular escenarios de fallo en SIMIT.

V.13 Mechatronics Concept Designer (MCD)

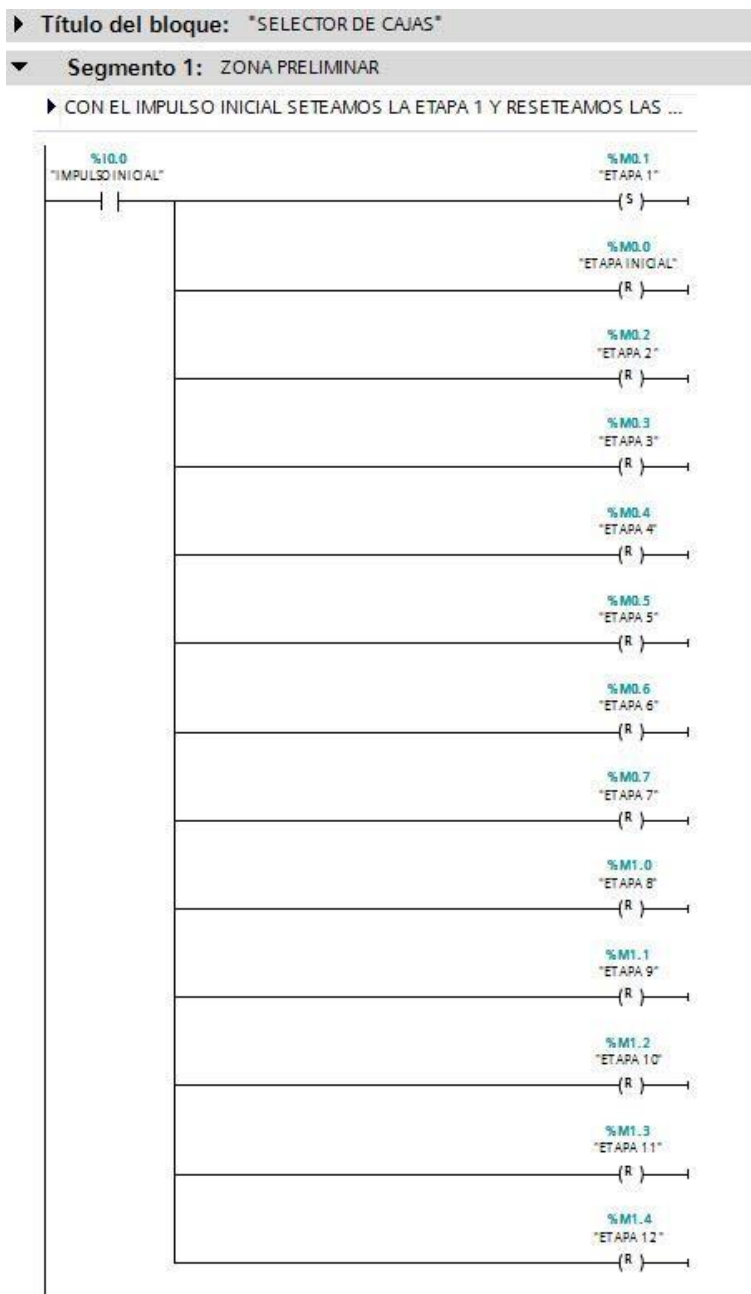
A partir de los datos CAD existentes, creados en NX12 o puestos a disposición con una importación, se modela el modelo cinemático de la planta en el MCD. Para ello, se definen los cuerpos rígidos y de colisión, se crean las articulaciones y se programan las operaciones.

Los parámetros establecidos en la interfaz con SIMIT se especifican a través de un adaptador de señales. El intercambio de datos tiene lugar a través del acoplamiento de memoria compartida.

Se puede observar que este es un proyecto un poco más complejo, es decir, requiere un poco más de habilidad ponerlo a funcionar, pero es una muy buena forma de poder ejemplificar todo lo que se puede hacer con los diversos softwares que nos ofrece Siemens, siendo uno de ellos claramente SIMIT.

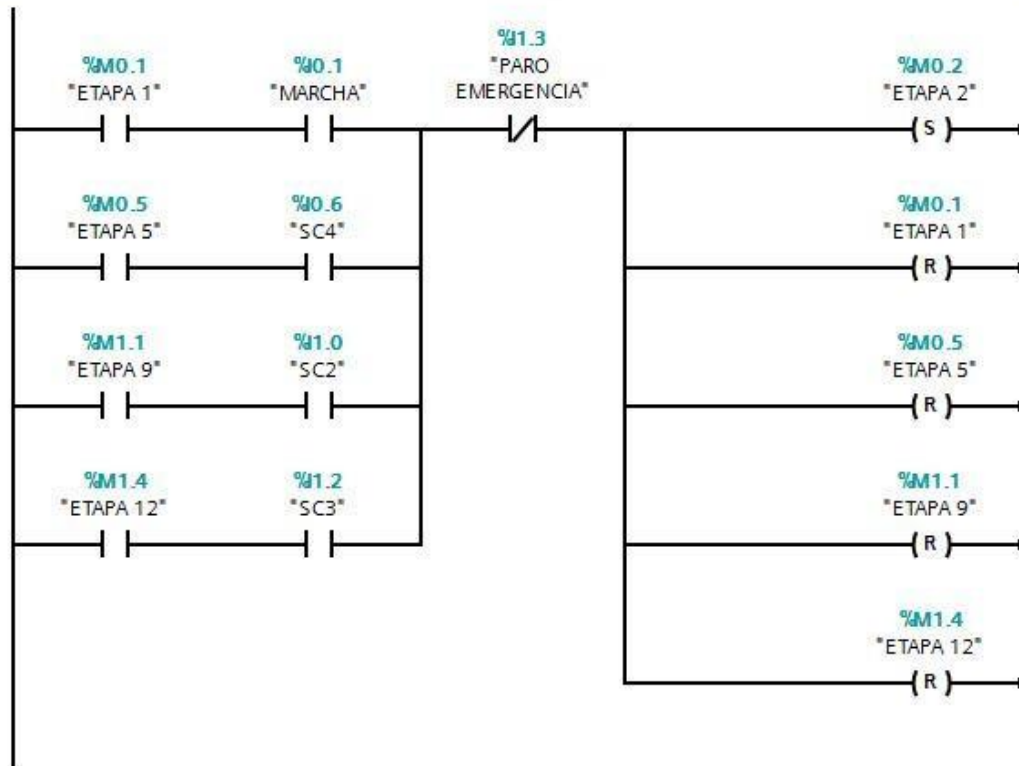
V.14 Código programado en lenguaje KOP a través del software TIA Portal

En este apartado se adjunta todas las líneas de código. *Bloque principal de segmentos:*



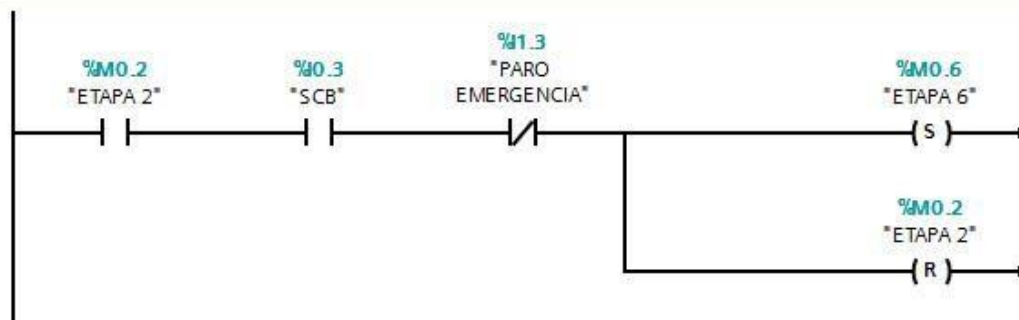
▼ **Segmento 2: ZONA SECUENCIAL**

► DESARROLLAMOS LA PARTE VERTICAL DE NUESTRO GRAFCET, DONDE LAS ENTRADAS SETEAN Y RESETEAN...



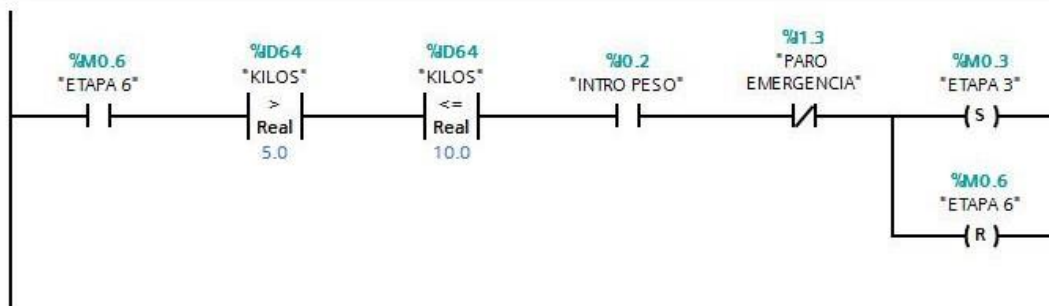
▼ **Segmento 3: ZONA SECUENCIAL - CAJA LLEGA A LA BÁSCULA**

Comentario



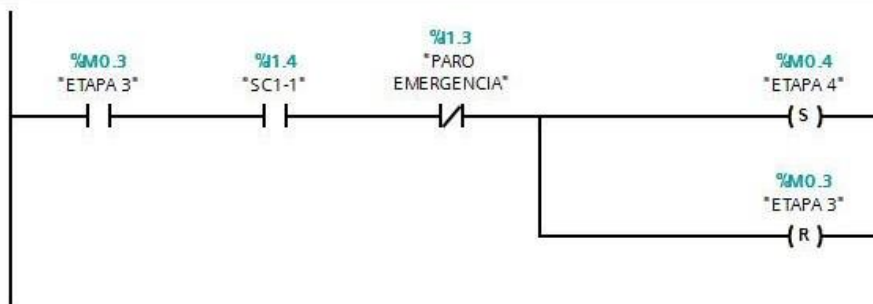
▼ **Segmento 4:** ZONA SECUENCIAL - SELECCION CAJA ENTRE 5 Y 10 KG

Comentario



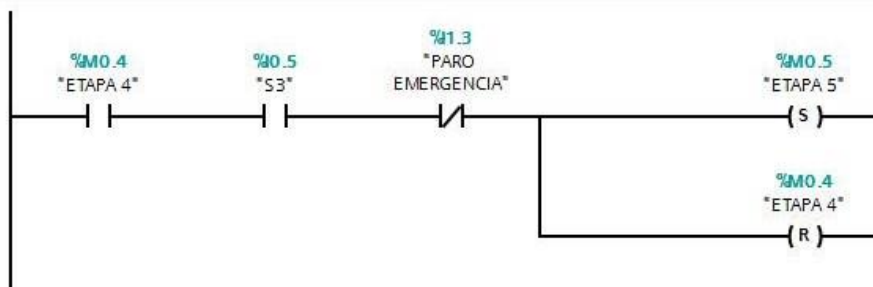
▼ **Segmento 5:** ZONA SECUENCIAL - SELECCION CAJA 5 Y 10 KG

Comentario



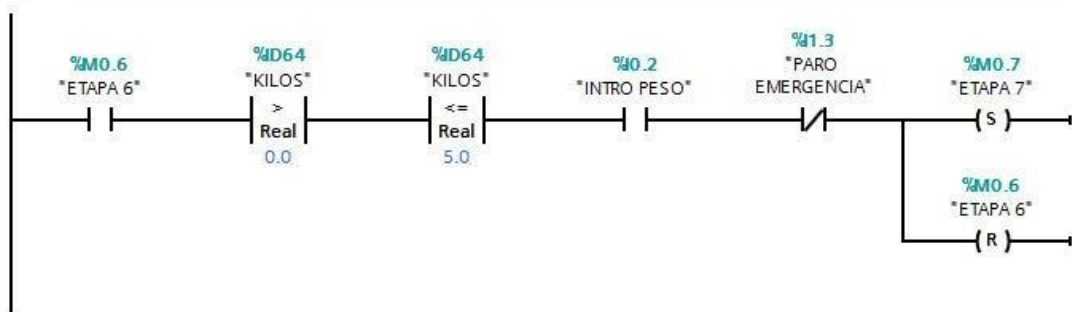
▼ **Segmento 6:** ZONA SECUENCIAL - SELECCION CAJA 5 Y 10 KG

Comentario



▼ Segmento 7: ZONA SECUENCIAL - SELECCION CAJA < 5KG

Comentario



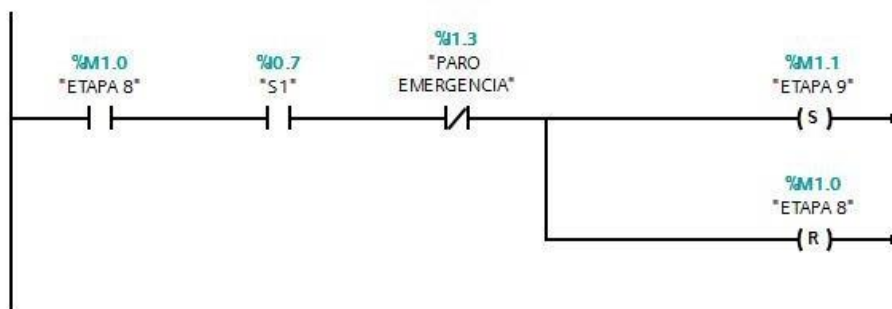
▼ Segmento 8: ZONA SECUENCIAL - SELECCION CAJA < 5KG

Comentario



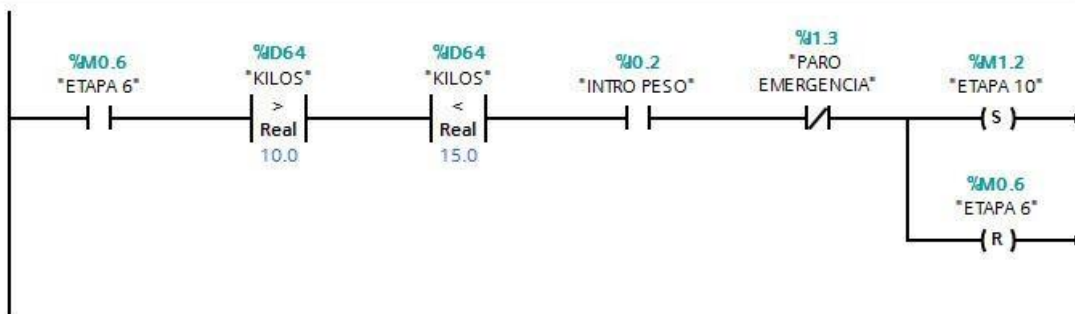
▼ Segmento 9: ZONA SECUENCIAL - SELECCION CAJA < 5KG

Comentario



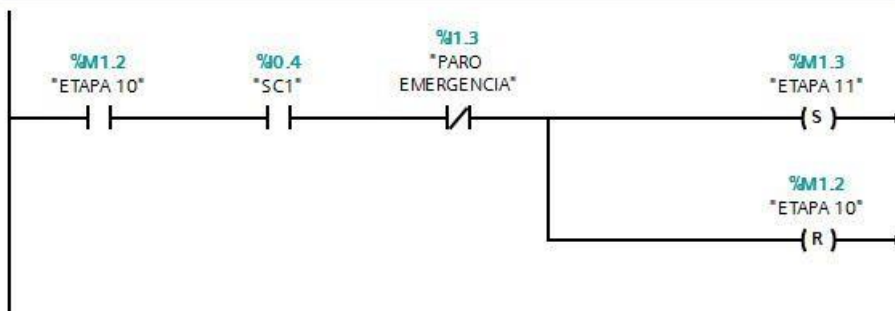
▼ **Segmento 10: ZONA SECUENCIAL - SELECCION CAJA > 10KG**

Comentario



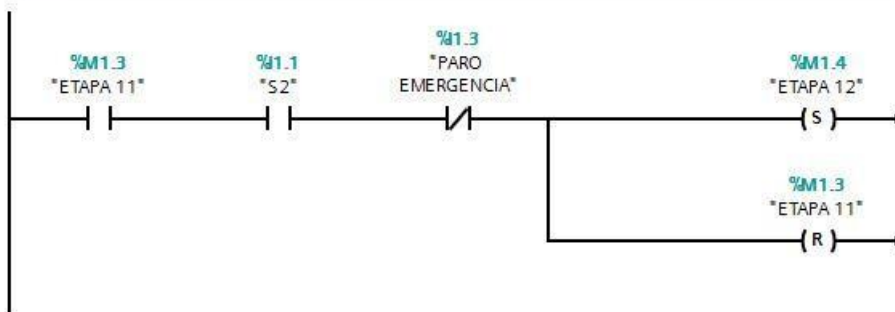
▼ **Segmento 11: ZONA SECUENCIAL - SELECCION CAJA > 10KG**

Comentario



▼ **Segmento 12: ZONA SECUENCIAL - SELECCION CAJA > 10KG**

Comentario



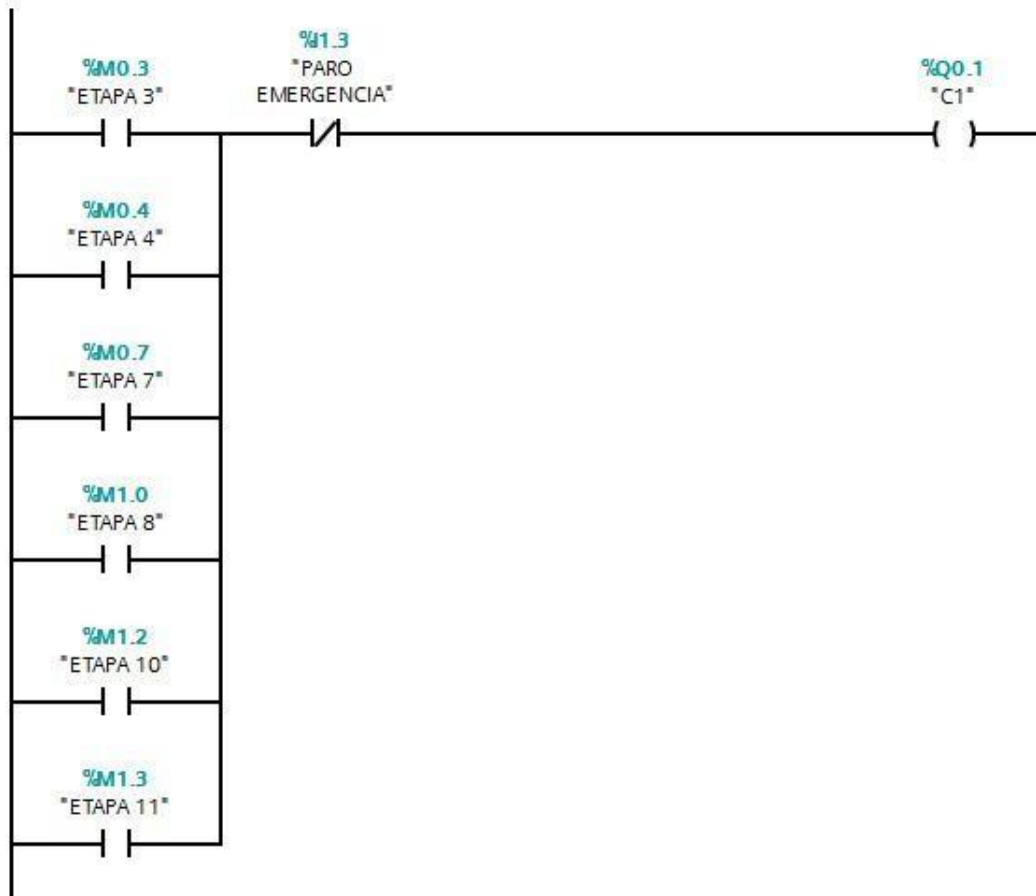
▼ **Segmento 13: ZONA DE ACCIONAMIENTOS**

DESALLORAMOS LA ZONA HORIZONTAL DEL GRAFCET, DONDE SE ACTIVAN LAS SALIDAS DEL SISTEMA.



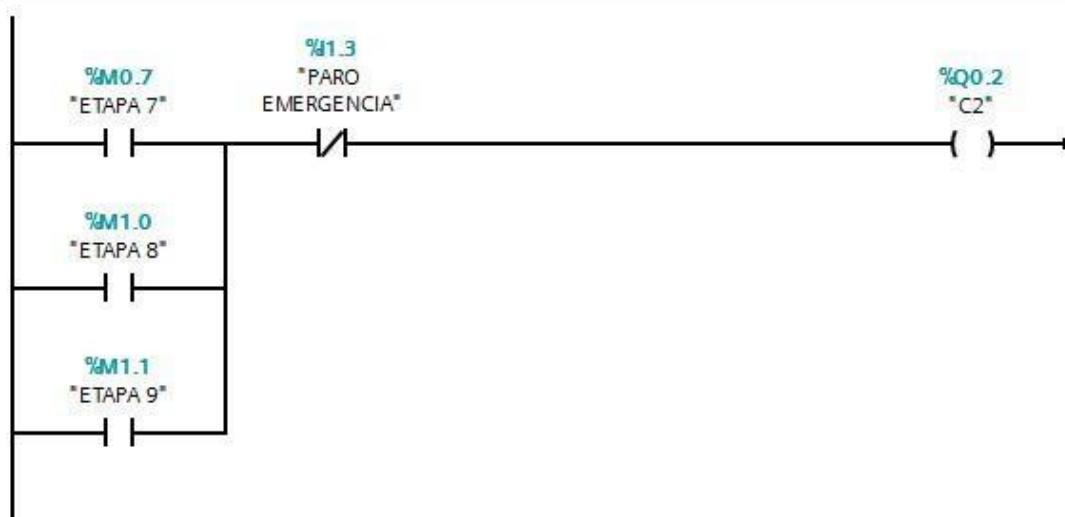
▼ **Segmento 14: ZONA DE ACCIONAMIENTOS**

DESALLORAMOS LA ZONA HORIZONTAL DEL GRAFCET, DONDE SE ACTIVAN LAS SALIDAS DEL SISTEMA.



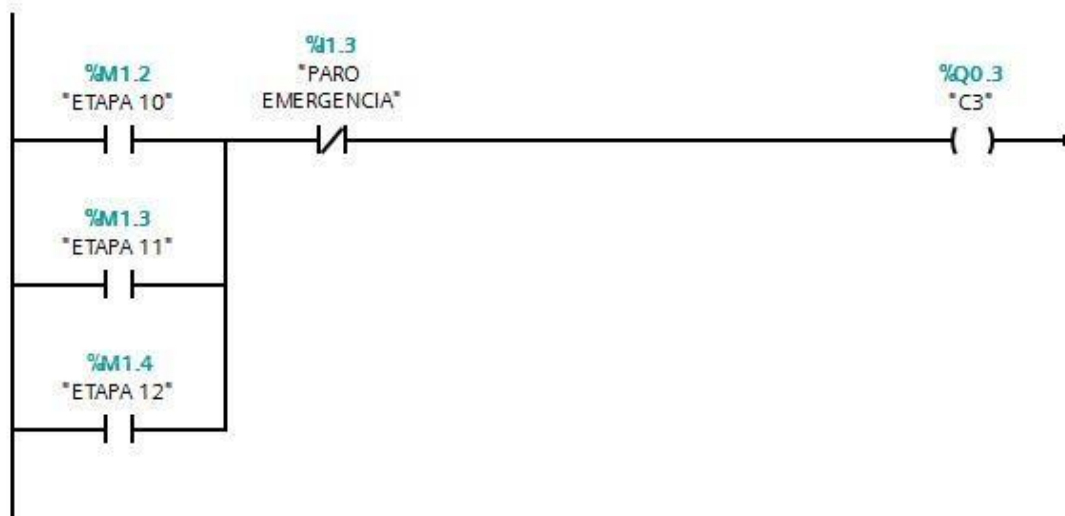
▼ **Segmento 15: ZONA DE ACCIONAMIENTOS**

DESALLORAMOS LA ZONA HORIZONTAL DEL GRAFCET, DONDE SE ACTIVAN LAS SALIDAS DEL SISTEMA.



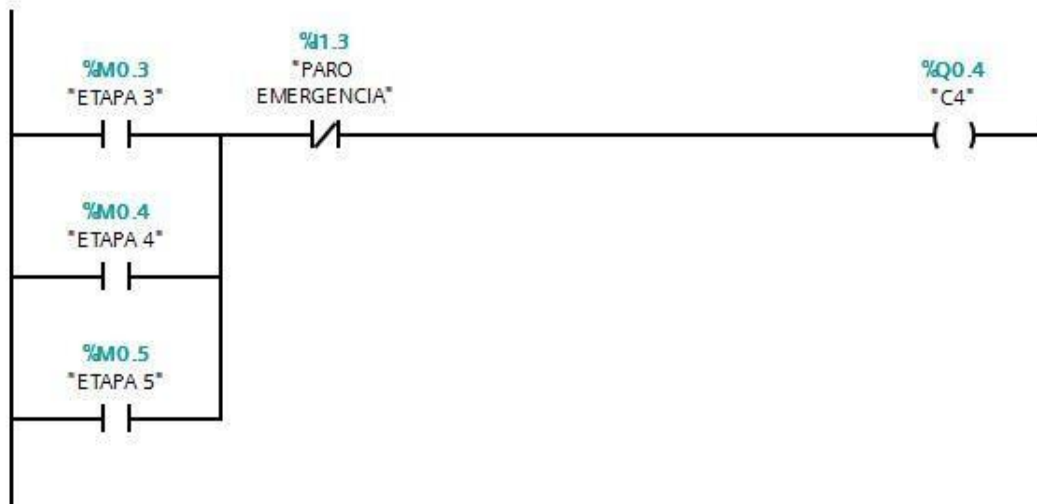
▼ **Segmento 16: ZONA DE ACCIONAMIENTOS**

DESALLORAMOS LA ZONA HORIZONTAL DEL GRAFCET, DONDE SE ACTIVAN LAS SALIDAS DEL SISTEMA.



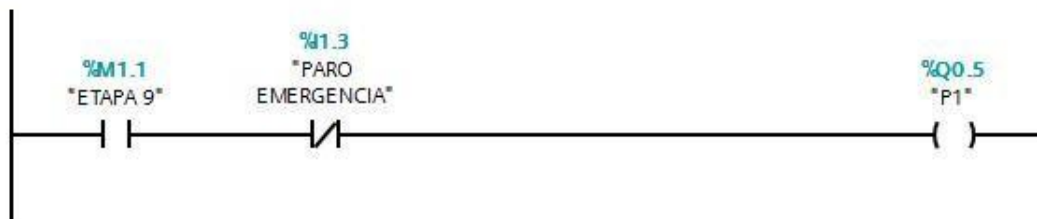
▼ **Segmento 17: ZONA DE ACCIONAMIENTOS**

DESALLORAMOS LA ZONA HORIZONTAL DEL GRAFCET, DONDE SE ACTIVAN LAS SALIDAS DEL SISTEMA.



▼ **Segmento 18: ZONA DE ACCIONAMIENTOS**

DESALLORAMOS LA ZONA HORIZONTAL DEL GRAFCET, DONDE SE ACTIVAN LAS SALIDAS DEL SISTEMA.



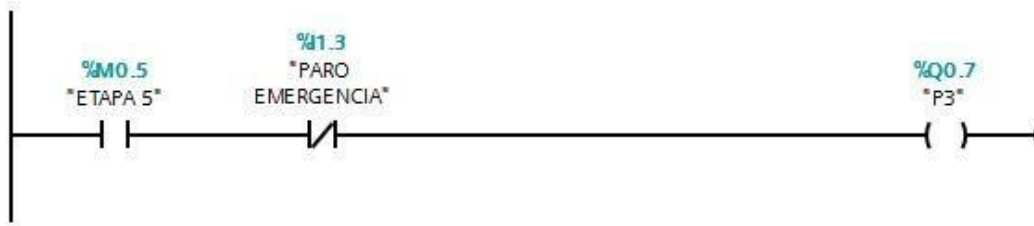
▼ **Segmento 19: ZONA DE ACCIONAMIENTOS**

DESALLORAMOS LA ZONA HORIZONTAL DEL GRAFCET, DONDE SE ACTIVAN LAS SALIDAS DEL SISTEMA.



▼ **Segmento 20: ZONA DE ACCIONAMIENTOS**

DESALLORAMOS LA ZONA HORIZONTAL DEL GRAFCET, DONDE SE ACTIVAN LAS SALIDAS DEL SISTEMA.



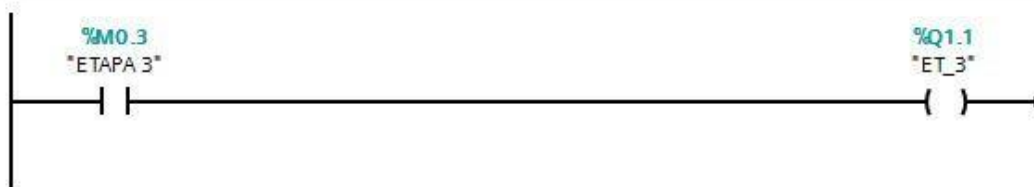
▼ **Segmento 21:**

Comentario



▼ **Segmento 22:**

Comentario



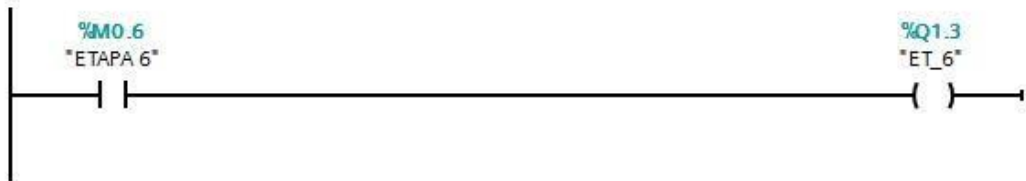
▼ **Segmento 23:**

Comentario



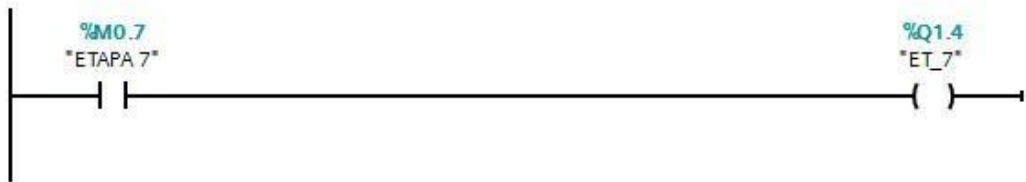
▼ Segmento 24:

Comentario



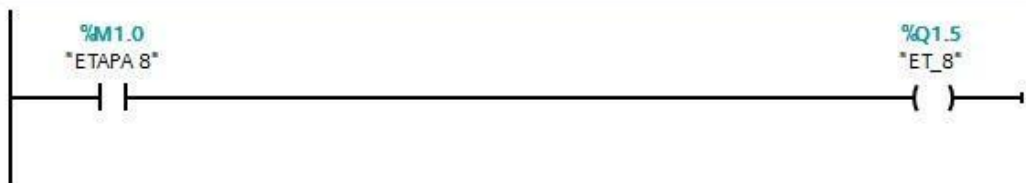
▼ Segmento 25:

Comentario



▼ Segmento 26:

Comentario



▼ Segmento 27:

Comentario

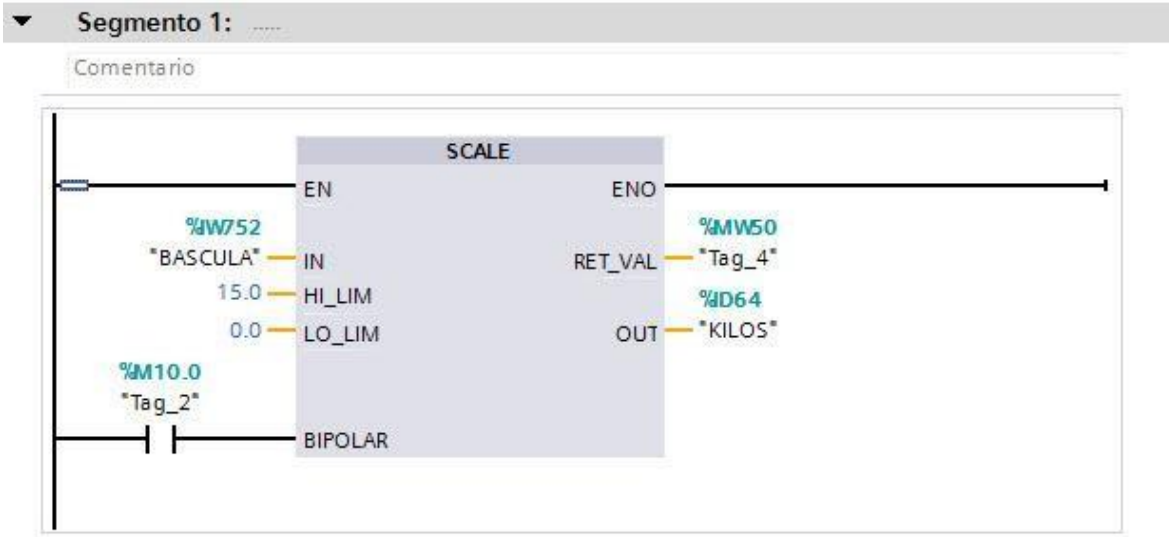


▼ Segmento 28:

Comentario



Bloque de función scale:



Bibliografía Capítulo V

Cortés López, O. (2017). Estudio teórico experimental con SIMIT. Universidad de Jaén, Escuela politécnica superior de linares.

https://tauja.ujaen.es/bitstream/10953.1/5451/1/Memoria_SIMIT.pdf

Video guía para llevar a cabo la instalación de SIMIT.

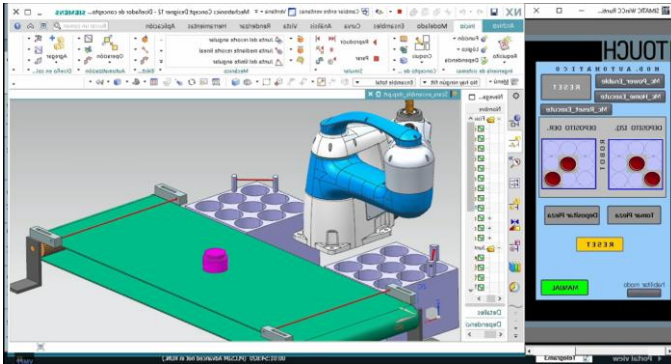
https://www.youtube.com/watch?v=EFKHZwGvYzc&ab_channel=Kurioseas

SIOS. (2020, 18 mayo). Siemens. Recuperado 26 de abril de 2022, de

<https://support.industry.siemens.com/cs/document/109780152/simulation-software-simit-v10-2?dti=0&lc=en-GR>

SIMATIC/SIMOTION virtual commissioning with "Hardware in the Loop"

https://cache.industry.siemens.com/dl/files/739/109758739/att_959577/v2/109758739_Documentation_HiL_en.pdf



Capítulo VI **CASO PRÁCTICO ROBOT SCARA**

En el ámbito industrial un área de alta especialización es la de control de movimiento de distintos grados de libertad con motores a pasos, de corriente directa, motores en alterna. La velocidad, la posición y la aceleración en los diversos ejes servo controlados son algunas de las variables que se deben tomar en cuenta para lograr un control de movimiento, actualmente dicho control emplea electrónica de estado sólido, dejando poco a poco atrás las tecnologías que emplean sistemas mecánicos como: engranajes, levas mecánicas, etcétera; gracias a estos avances se puede librar costos asociados con el mantenimiento y el desgaste de las piezas mecánicas. Un ejemplo concreto del uso del control de movimiento en las empresas son las máquinas paletizadoras, empaquetadoras o los transportadores. Implementar un sistema de control de movimiento es un proyecto complejo, que el poderlo hacer en un ambiente virtual es seguro para validaciones, reforzamiento de la venta y capacitaciones sobre la operatividad del sistema.

Co-simulación:

*-NX MCD con
SIMATIC WinCC*

*-NX MCD con TIA
Portal Ver 15.1*

*-NX MCD con PLC
Virtual (PLCSim
Advanced Ver 2.0
SP1)*

*-NX MCD con PLC
Virtual y SIMIT*

VI.1 Introducción

El siguiente documento presenta una explicación a detalle del desarrollo completo del caso práctico del *Control Robot SCARA usando SIMIT Y NX MCD*. Primeramente, se explicará el funcionamiento, como se puede ver la en la figura VI. 1 se tiene lo que sería el prototipo de un brazo tipo SCARA, el objetivo consiste en colocar la pieza de color rosa en una de las posiciones que se encuentran determinadas en la Human Machine Interface (HMI), (los círculos de color rojo), dependiendo de cuál se elija, el brazo colocará la pieza de manera automática. En la misma figura se indican los tres ejes, en azul es el eje 1, el blanco el 2 y el de color café es el 3.

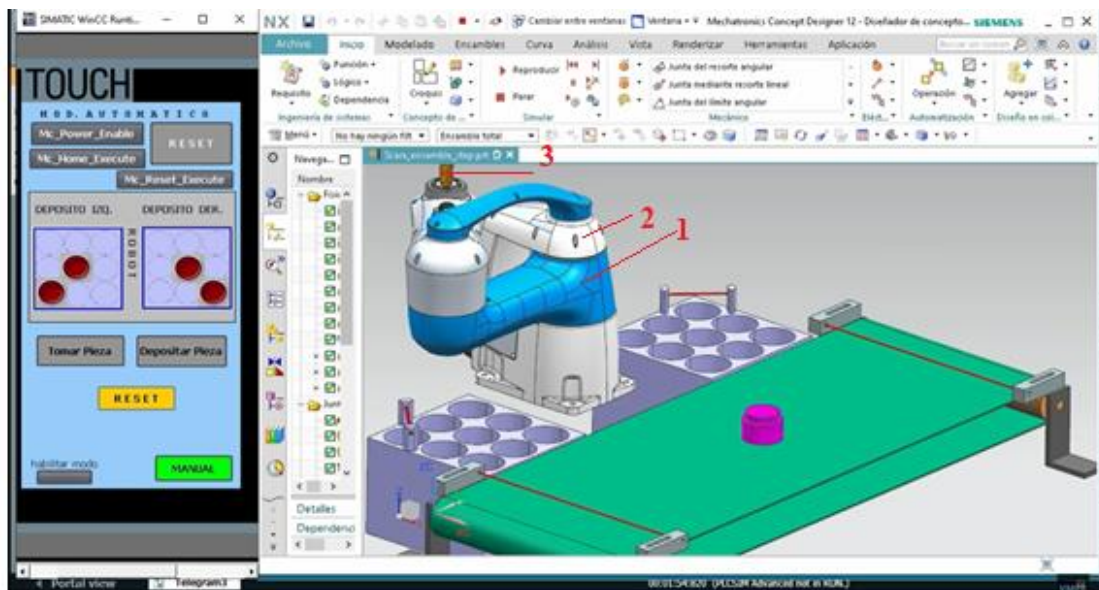


Figura VI.1 Aplicación control de posición Robot Scara.

En la interfaz HMI se pueden tener dos modos de control, uno manual y otro automático, el que se ve en la figura VI. 1 es el automático y el que se ve en la figura VI. 2 es el manual. Si observamos la figura VI. 1, en la parte superior de la interfaz HMI se tienen cuatro botones los cuales son los encargados de activar, los ejes del brazo, más abajo se tienen las posibles posiciones en las que se puede colocar la pieza, justo debajo de estos botones se tienen los botones con los que controlaremos el proceso:

- Tomar pieza: mueve el brazo hasta colocarlo sobre la pieza, baja el Axis (eje) 3 y acciona la pinza.
- Depositar pieza: dependiendo de la posición que se seleccione el brazo moverá la pieza hasta ahí y la dejará caer.
- Reset: sirve para reiniciar el proceso general.

En la parte inferior se tiene el botón verde de activación del modo, que en este caso es el manual y el botón que nos permite cambiar la interfaz HMI a lo que se ve en la figura VI. 2. El *modo manual* es más sencillo de explicar a pesar de que la interfaz HMI se ve un poco más

compleja, se divide en tres partes: la del primer eje, la del segundo eje y la del tercer eje más la pinza. Se coloca con un tablero un número entero que será la posición en grados en la que se desea posicionar al eje y se carga la posición al activar el botón de *RAbsolut*, cada vez que se cambie la posición será el mismo procedimiento. Para activar los ejes 1 y 2 es necesario primeramente accionar los botones *Mc_Power_Enable* y *Mc_Home_Enable* de cada eje. El tercer eje solo tiene dos movimientos subir o bajar, y la pinza solo cierra o abre.



Figura VI.2 Interfaz HMI del modo manual.

VI.2 Desarrollo

VI.2.1 Configuración del gemelo digital SCARA en Siemens NX MCD

Lo primero es declarar algunas propiedades físicas que se irán registrando en el navegador de físicas, para esto es necesario que tengamos el menú correcto, el cual es el que se puede ver en la figura VI.3, para poder trabajar se declaran propiedades físicas básicas, juntas, y actuadores.

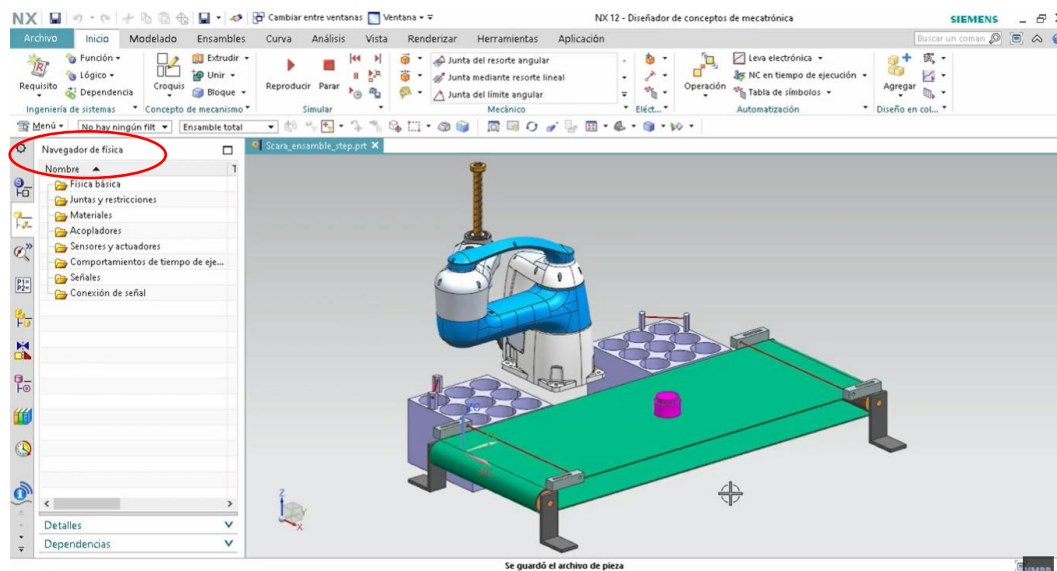


Figura VI.3 Menú navegador de físicas.

Lo que hay que configurar en la ventana navegador de física es lo que se observa en las figuras VI. 4, VI.5 y VI.6, en la primera se definen los cuerpos rígidos, los cuerpos de colisión, juntas fijas, juntas de bisagras, juntas de charnela y juntas deslizantes.

Name	Type
Basic Physics	
Bandas	Collision Body
Base	Rigid Body
Base_izq	Rigid Body
Eslabon_1	Rigid Body
Eslabon_2	Rigid Body
Eslabon_3	Rigid Body
Guia_corredera	Rigid Body
Pinza_fija	Rigid Body
Pinza_movil	Rigid Body
Producto	Rigid Body
Joints and Constraints	
Base	Fixed Joint
Eslabon_1_Base	Hinge Joint
Eslabon_2_Eslabon_1	Hinge Joint
Eslabon_3_Eslabon_2	Sliding Joint
Guia_corredera_Eslabon_3	Fixed Joint
Pinza_fija_Guia_corredera	Fixed Joint
Pinza_movil_Guia_corredera	Sliding Joint

Figura VI.4 Cuerpos rígidos, colisiones y juntas definidas del gemelo digital.

En la figura VI.5 se pueden observar los controles de velocidad y de posición que se declararon, es decir, así del eje 1 y 2 se requiere configurar para la simulación velocidad de control, para el eje 3 en el cual se encuentra la pinza móvil (efector final), se deben configurar controles de solo posición.

Materials	
Couplers	
Sensors and Actuators	
<input checked="" type="checkbox"/> Axis_1	Speed Control
<input checked="" type="checkbox"/> Axis_2	Speed Control
<input checked="" type="checkbox"/> Axis_3	Position Control
<input checked="" type="checkbox"/> Pinza_movil	Position Control
Runtime Behaviors	
Signals	
Signal Connection	

Figura VI.5 Configuración para actuadores con controles de posición y velocidad del brazo.

Algo importante a destacar es que las piezas a las que llamamos base izquierda, producto, pinza fija y pinza móvil requieren de cuerpos de colisión, los cuales son para que se pueda sujetar la pieza con las pinzas.

<input checked="" type="checkbox"/> Base_izq	
<input checked="" type="checkbox"/> CollisionBody(14)	
<input checked="" type="checkbox"/> CollisionBody(15)	
<input checked="" type="checkbox"/> Pinza_fija	
<input checked="" type="checkbox"/> CollisionBody(11)	
<input checked="" type="checkbox"/> CollisionBody(12)	
<input checked="" type="checkbox"/> CollisionBody(13)	
<input checked="" type="checkbox"/> Pinza_movil	
<input checked="" type="checkbox"/> CollisionBody(8)	
<input checked="" type="checkbox"/> CollisionBody(9)	
<input checked="" type="checkbox"/> CollisionBody(10)	
<input checked="" type="checkbox"/> Producto	
<input checked="" type="checkbox"/> CollisionBody(1)	
<input checked="" type="checkbox"/> CollisionBody(2)	
<input checked="" type="checkbox"/> CollisionBody(3)	
<input checked="" type="checkbox"/> CollisionBody(4)	
<input checked="" type="checkbox"/> CollisionBody(5)	
<input checked="" type="checkbox"/> CollisionBody(6)	
<input checked="" type="checkbox"/> CollisionBody(7)	

Figura VI.6 Configuración de superficies de colisión.

Estas superficies se enumeran conforme se seleccionan en el modelo 3D, sin embargo no es importante el orden, sólo que las superficies de contacto de cada componente que interactúe con otro se elijan.

A continuación se describe el cómo se dan de alta las características de los componentes como comportamientos físicos.

1. Activación del menú correcto.

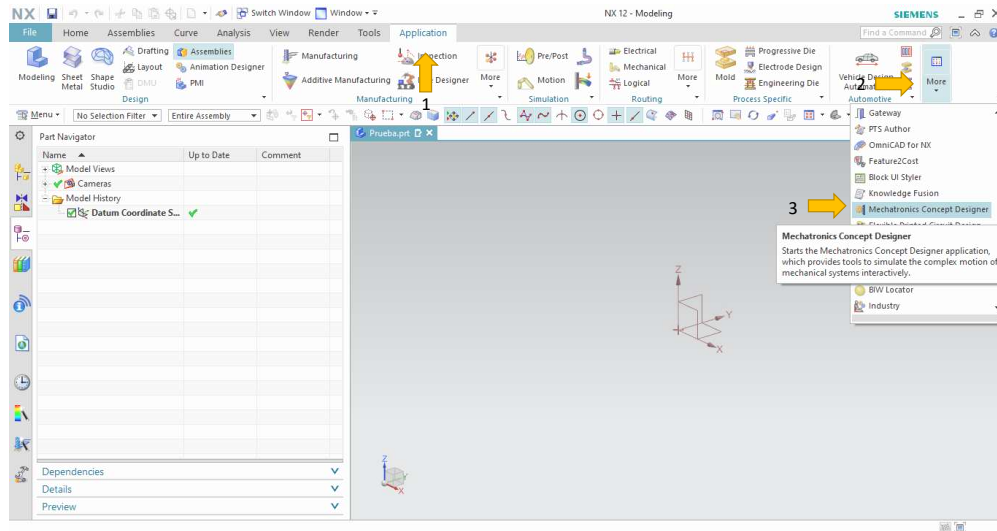


Figura VI.7 Herramienta de modelado de NX.

2. Declaración de cuerpos de colisión.

La banda es el único componente que solo se declara como cuerpo de colisión, los demás componentes se declararan de manera diferente.

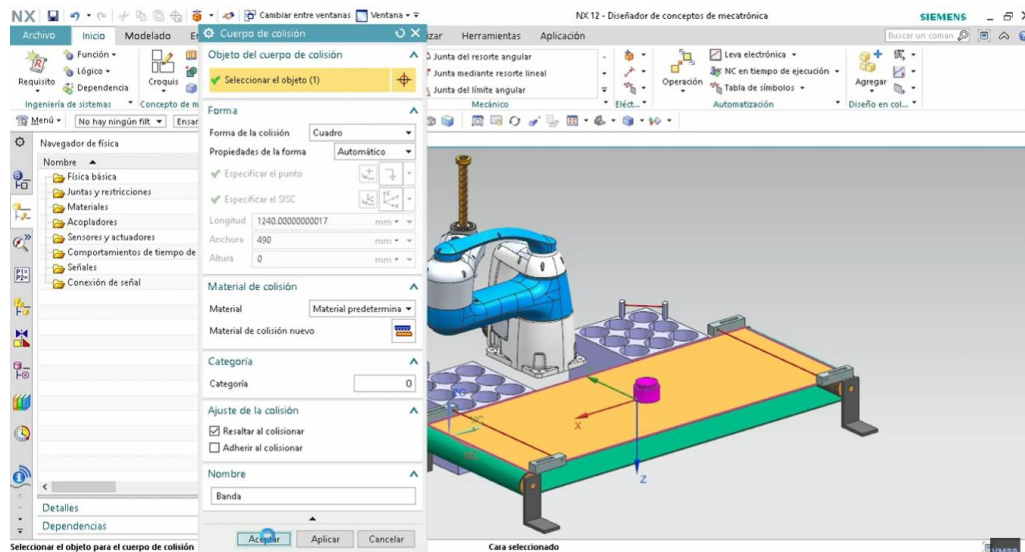


Figura VI.8 Cuadro de diálogo cuerpo de colisión.

En esta caja de diálogo se tienen las siguientes secciones:

- Objeto de colisión: seleccionamos el componente dando clic en él.
- Forma: tenemos distintas para abarcar la forma del componente, para formas irregulares se elige de tipo malla, para nuestra banda es el cuadrado como se aprecia en la figura anterior.
- Material: lo dejamos en automático, es posible cambiarlo, pero en el trabajo colaborativo, el modelo mecánico ya viene con los materiales apropiados y validados con las herramientas pertinentes de CAE (Ingeniería Asistida por Computadora, en español).
- Categoría: de igual manera no hacemos ningún cambio.
- Ajuste de la colisión: aquí elegimos que se muestre la superficie de colisión al momento de la simulación, para facilitar la validación correspondiente.
- Nombre: damos un nombre de acuerdo a la función del componente en el sistema a automatizar.

3. Declaración de cuerpos rígidos.

Se tienen bastantes cuerpos rígidos, todos se declaran de la misma manera, es importante poner un nombre según su función.

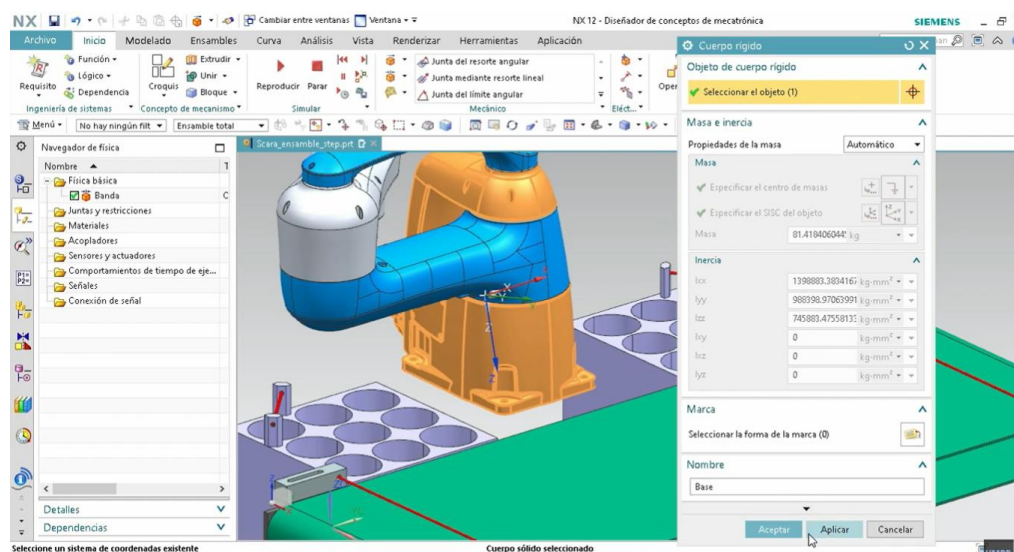


Figura VI.9 Cuadro de diálogo cuerpo rígido.

En esta caja de diálogo se tienen las siguientes secciones:

- Objeto de cuerpo rígido: seleccionamos el componente dando clic en él.
- Masa e inercia: se calcula automáticamente según el sistema de coordenadas del modelo, dimensiones geométricas y material, por lo que ahí elegimos la opción de automático.
- Marca: de igual manera no hacemos ningún cambio.

- Nombre: damos un nombre de acuerdo a la función del componente en el sistema a automatizar.

El total de los cuerpos rígidos que se declaran son los siguientes e indicados en color naranja porque están seleccionados en el software, se remarcan en el navegador en la carpeta de física básica (Basic Physics):

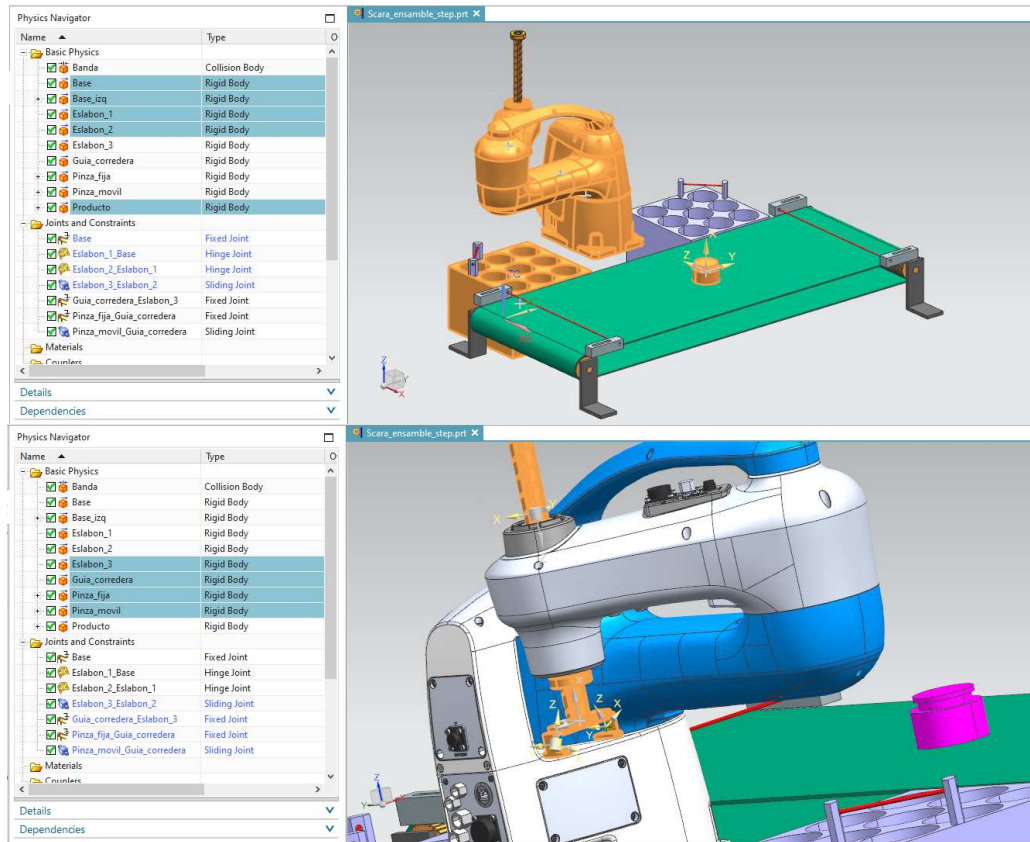


Figura VI.10 Cuerpos rígidos definidos en el sistema.

4. Declaración de junta fija.

Ésta es una junta para que la pieza no caiga al vacío al momento de simular cuando ya fue declarada como cuerpo rígido y se ve afectada por el vector de gravedad, la base del brazo se declara con este tipo de junta. En la caja de diálogo correspondiente solo hay que dar clic en el componente a declarar y darle un nombre.

Otras juntas fijas son las del Axis 3 con la pinza y la de la pinza con su pieza fija, es importante tener en cuenta que estas juntas son como si estuvieran soldadas, las dos piezas que se relacionen no se podrán mover individualmente, si no que quedarán fijas la una con la otra. Pero si una de estas dos piezas tiene una junta diferente con otra pieza, esto no afectará.

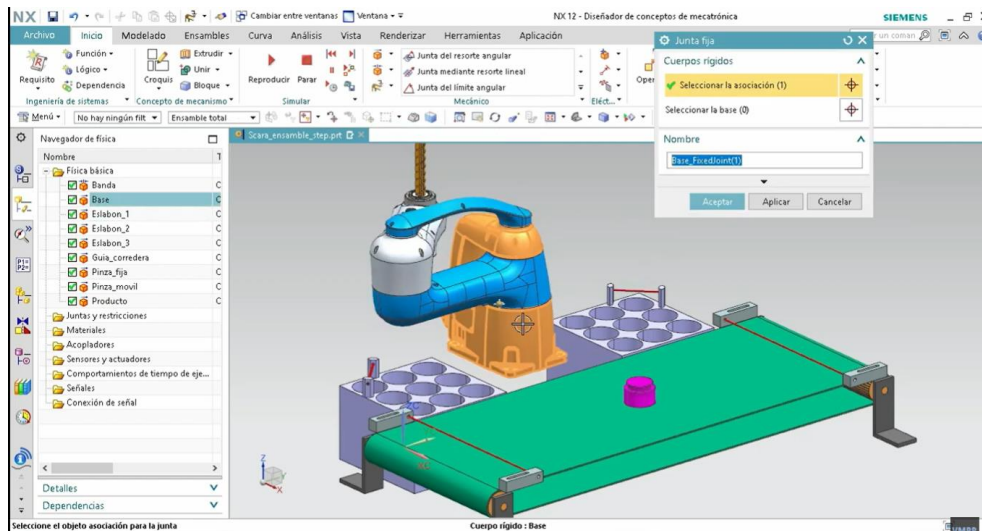


Figura VI.11 Junta fija para la base del Robot Scara.

5. Como ocultar o mostrar cosas.

Solo es necesario seleccionar esa función y se nos desplegará una pequeña ventana, seleccionamos la pieza que queremos ocultar, damos aceptar y la pieza desaparecerá. Para volverla a poner se sigue el mismo proceso solo que el lugar de seleccionar ocultar, se selecciona mostrar.

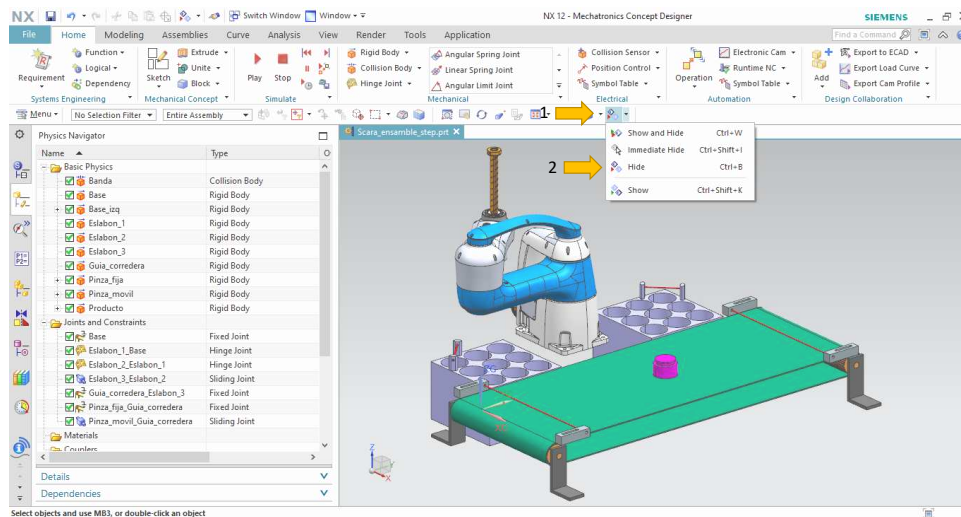


Figura VI.12 Menú para ocultar y mostrar componentes.

6. Declaración de juntas de charnela.

Para saber cómo se declararán estas juntas es importante poner atención al nombre, por ejemplo, la primera que se tiene se llama *Eslabon_1_Base*, esto quiere decir que la *asociación* es el eslabón 1 y la *base* (referencia del movimiento rotatorio) es la base del brazo; con base a

lo explicado será para declarar todas, tomando como referencia el nombre, la primera pieza siempre será la *asociación* y la segunda la *base*.

Para seleccionar el eje y el ángulo, solo se tiene que asegurar que el punto de referencia quede centrado en la pieza como será un movimiento giratorio, el eje podría estar hacia arriba o hacia abajo pero siempre es preferente manejarlos hacia arriba, por la regla de la mano derecha para movimientos rotatorios, ampliamente utilizado en robótica, si el eje que se observa en color café de la figura VI.13 se visualiza hacia abajo, para cambiar el sentido de éste, solo hay que darle doble clic.

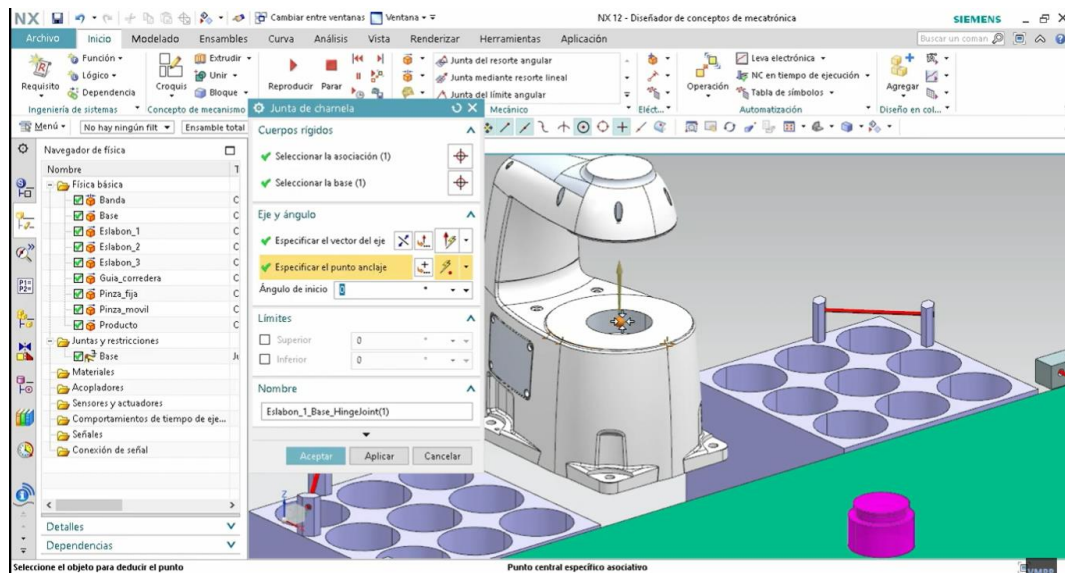


Figura VI.13 Menú junta de charnela aplicada en el eje 1 y 2.

7. Declaración de junta deslizante.

Para saber cuál pieza es la asociación y la base se puede aplicar lo mismo con todas las juntas, pero aquí es importante tener en cuenta el tipo de movimiento que se requiere hacer al momento de declarar al eje, ya que el primer movimiento que tendrá esta junta será hacia abajo para coger la pieza.

Esto mismo aplica para la otra junta deslizante que tenemos que es la de la pinza, tenemos que poner el vector en la dirección del primer movimiento de la pinza.

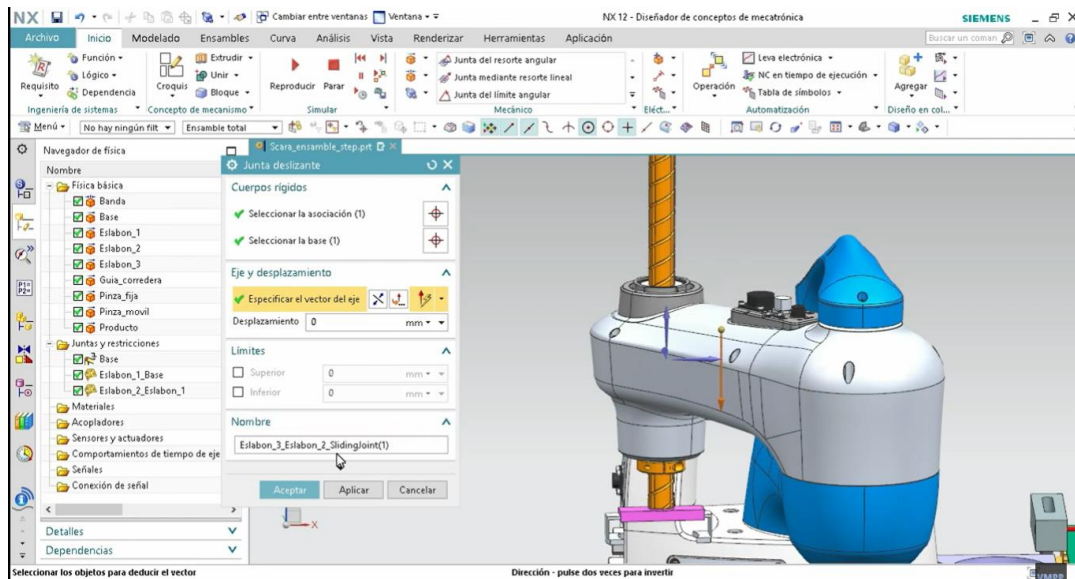


Figura VI.14 Menú junta de deslizante aplicada en el eje 3.

En esta caja de diálogo se tienen las siguientes secciones:

- **Cuerpos rígidos:** se elige la referencia fija (eje 2) y se da clic en el eje 3 que será el componente seleccionado como asociación.
- **Eje y desplazamiento:** en la figura VI.14 se observa de color violeta fuerte un sistema de coordenadas por lo que se elige la que requerida para el movimiento: eje z, y aparece otro vector al cual se le da doble clic para que cambie de sentido y se vea como en la figura VI.14, el cual está de color café.
- **Límites:** en este momento no establecemos límites, pero si es posible hacerlo
- **Nombre:** damos un nombre a la junta: Eslabón3_Eslabón2

8. Declaración de control de posición.

En este caso tenemos dos controles de posición, uno para el Axis 3 y otro para la pinza, para configurar estos simplemente se selecciona el control de posición y luego el eje de la junta que tendrá dicho control.

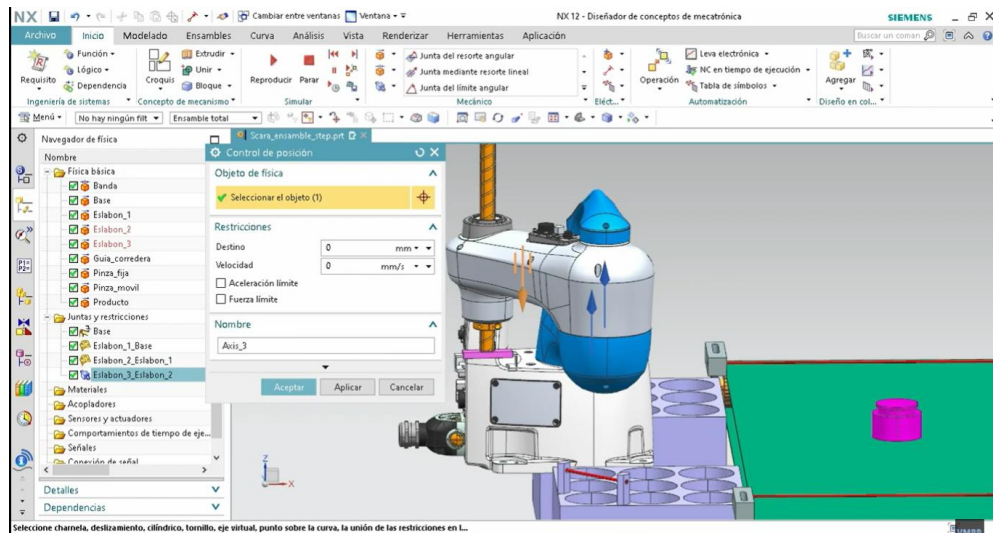


Figura VI.15 Menú Control de posición aplicado al eje 3 con junta deslizante.

En esta caja de diálogo se tienen las siguientes secciones:

- Objeto de física: se elige al eje tres ya declarado como cuerpo rígido, Eslabón_3.
- Restricciones: aquí es posible ajustar destino, velocidad, aceleración límite y fuerza límite, en este momento no se coloca ningún dato ni opción.
- Nombre: se nombra al control de posición Axis_3.

9. Declaración de control de velocidad.

Para el control de velocidad se siguen los mismos pasos que para el control de posición; el control de velocidad lo tendrán el Axis 1 y el Axis 2.

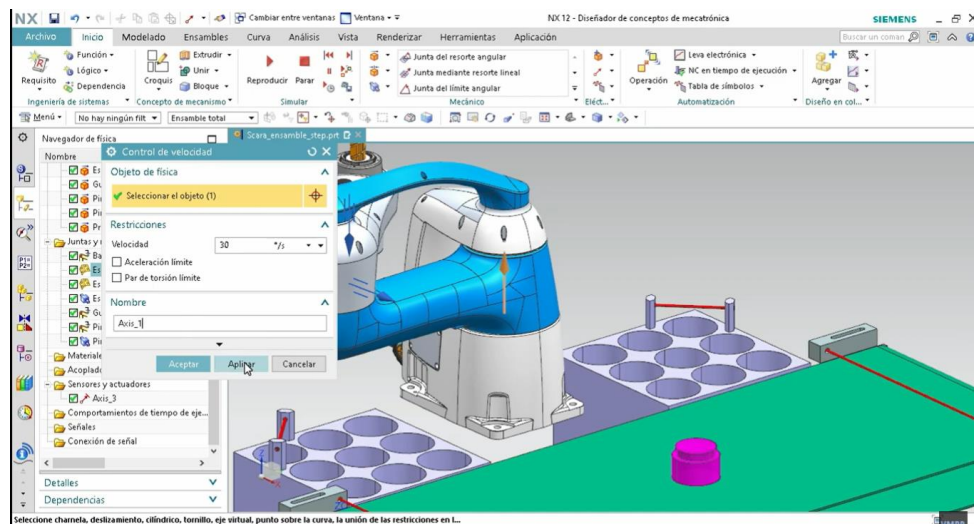


Figura VI.16 Menú Control de velocidad aplicado al eje 1 y 2 con juntas de charnela.

En esta caja de diálogo se tienen las siguientes secciones:

- Objeto de física: se elige al eje tres ya declarado como cuerpo rígido, Eslabón_3.
- Restricciones: estás cambiarán según la junta seleccionada y el movimiento que representa la junta, como en este caso es rotatoria, la restricción de velocidad se puede definir como grados/segundo, (si se tiene el sistema trabajando con el sistema métrico decimal), sin embargo en este momento no se da ningún dato.
- Nombre: damos un nombre al control de velocidad según sea la declaración Axis_1 y/o Axis_2.

10. Cuerpos de colisión.

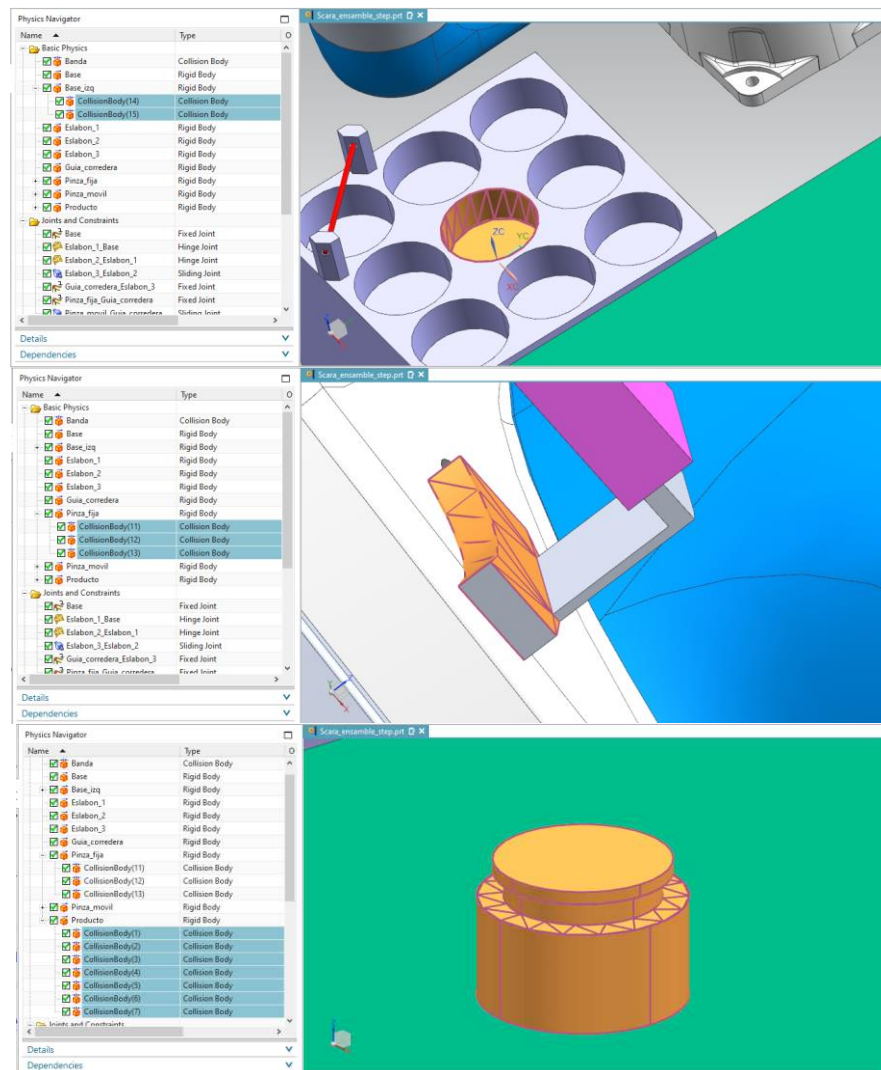


Figura VI.17 Cuerpos de colisión producto, pinza fija, base izquierda.

Recordar que en el cuadro de diálogo de cuerpos de colisión es posible elegir formas geométricas típicas de 1 dimensión a dos y cuando son formas irregulares se tiene la opción de malla como se observa en el producto, pinza_fija y base_izq de la Figura VI.17, indicando los cuerpos en color naranja y las líneas moradas como formando triángulos.

VI.2.2 Conceptos teóricos de configuración

A continuación, se introduce el cómo se realizará la configuración necesaria para la parte de potencia de los servomotores. Se cuenta con una unidad de control que es el PLC virtual de la familia S71500, como accionamientos se tendrán dos dispositivos Sinamic V90, estos dispositivos dinámicos tienen como entradas: datos del encoder (sensor digital integrado al cuerpo del servomotor) y salidas señales de control y de potencia al servomotor, es decir, la etapa de potencia. Las flechas rojas nos indican los ejes (articulaciones) que los motores controlan.

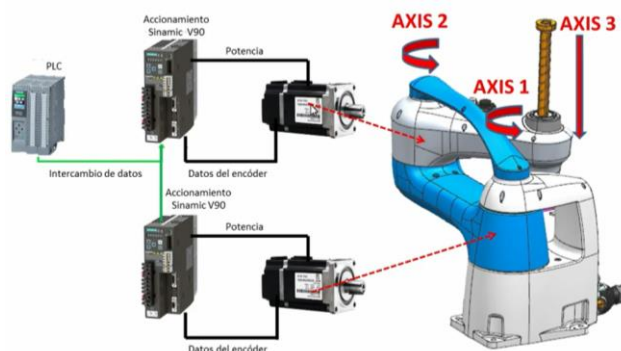


Figura VI.18 Esquema de los controles del Brazo Scara.

VI.2.3 SIMIT Bloque equivalente

En la figura VI.19 se puede ver lo que sería el bloque equivalente que proporciona el software SIMIT en versión demo para el control de la parte de potencia para los servomotores

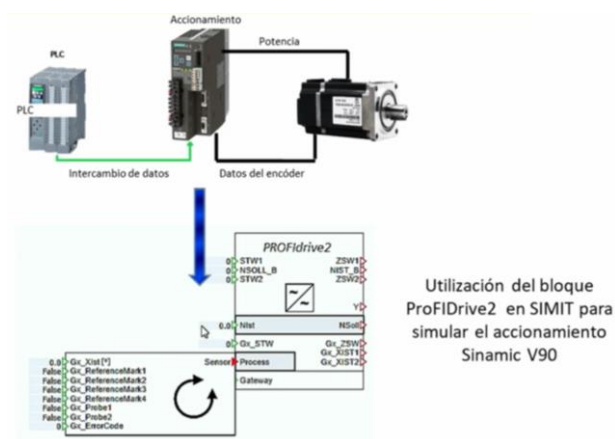


Figura VI.19 Bloque para simular el equipo Sinamic V90, ProFIdrive 2 en SIMIT.

Tomando como referencia lo anterior, algo importante, es identificar en dónde se encuentran estos bloques en el software SIMIT, lo que se indica en las figuras VI.20 y VI.21.

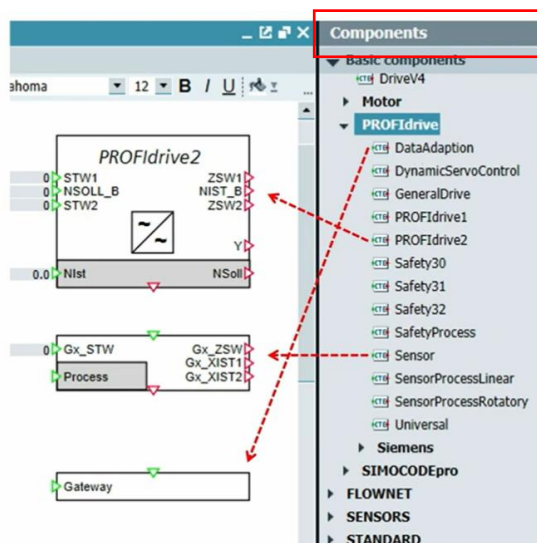


Figura VI.20 Primera parte de los bloques.

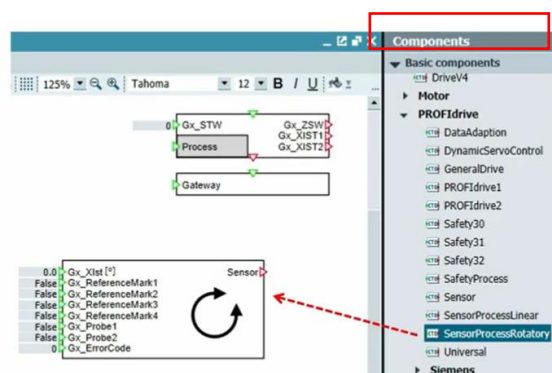


Figura VI.21 Segunda parte de los bloques.

Ahora bien, *PROFdrive* es el perfil de *PROFIBUS* y *PROFINET* para accionamientos con un amplio campo de aplicación en la automatización de procesos y manufacturera. Este es independiente del sistema de bus que se utilice *PROFIBUS* o *PROFINET*. El cómo se seleccione el telegrama a usar depende de la clase de aplicación *PROFdrive*. En la siguiente figura VI.22 se puede ver una tabla que muestra esquemáticamente las clases de aplicación *PROFdrive* que se alcanzan con los diferentes telegramas.

Telegrama	Descripción
1	Consigna de velocidad de 16 bits
2	Consigna de velocidad de 32 bits
3	Consigna de velocidad de 32 bits con 1 encóder de posición
4	Consigna de velocidad de 32 bits con 2 encóders de posición
5	Consigna de velocidad de 32 bits con 1 encóder de posición y Dynamic Servo Control
6	Consigna de velocidad de 32 bits con 2 encóders de posición y Dynamic Servo Control
7	Posicionador simple con selección de secuencia de desplazamiento
9	Posicionador simple con entrada directa de consigna (MDI)
20	Consigna de velocidad de 16 bits para VIK-NAMUR
81	Encóder estándar
82	Encóder estándar con velocidad real de 16 bits

Figura VI.22 Diferentes telegramas.

Para la aplicación descrita en el presente capítulo, se utilizará el telegrama 3, que permite realizar un control de posición.

	PZD 1	PZD 2-3	PZD 4	PZD 5	PZD 6-7	PZD 8-9
TEL_3 Nominal (Envío)	STW 1	NSOLL_B	STW 2	G1_STW		
TEL_3 Real (Respuesta)	ZSW 1	NIST_B	ZSW 2	G1_ZSW	G1_XIST 1	G1_XIST 2

Telegrama 3 de PROFIdrive

	NOMINAL		REAL
STW	Palabra de control	ZSW	Palabra de estado
NSOLL	Consigna de velocidad	NIST	Velocidad real
G1 STW	Palabra de control del encóder	G1 XIST1	Valor cíclico del encóder
		G1 XIST2	Valor absoluto del encóder
		G1 ZSW	Palabra de estado del encóder

Palabras del telegrama 3

Figura VI.23 Palabras de estado.

Es de suma importancia comprender cuales son las palabras que se encargan de enviar datos y de recibir datos, en la figura VI.23 se encuadra con color rojo los datos enviados por el PLC y los recibidos por el drive los cuales se encuadran con un recuadro de color café, además se lo anterior se ilustra más claramente en la figura VI.24.

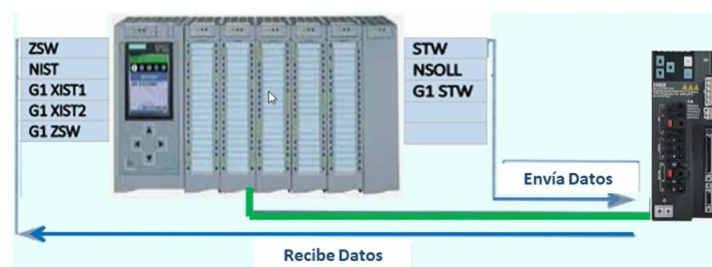


Figura VI.24 Dirección del envío de información empleando Telegramas.

A continuación se explicarán algunas de las características más importantes de algunos de los bloques mencionados anteriormente:

1. Bloque PROFIdrive 2.

El bloque *PROFIdrive 2* recibe las palabras de envío del telegrama 3 *STW1*, *NSOLL_B* y *STW2* y envía las palabras de recepción del telegrama 3 *ZSW1*, *NIST_B* y *ZSW2*.

También se comunica con *NX-MCD* mediante las señales *NIST* (velocidad actual) y *NSOLL* (consigna de velocidad).

2. Bloque Sensor y SensorProcessRotatory.

Los bloques *Sensor* y *SensorProcessRotatory* se encargan de simular el encoder, este tiene como entrada el valor real de posición. Este valor procede del actuador *Speed Control de NX-MCD*.

Por otro lado, el bloque *Sensor* tiene como entrada las palabras *GX_STW* (palabra de control del encoder) y como salidas las palabras *GX_ZSW* (palabra de estado del encoder), *GX_XIST1* (valor cíclico del encoder) y *GX_XIST2* (valor absoluto del encoder).

3. DataAdaption.

Los objetos tecnológicos del S7-1500 tienen una funcionalidad con la que sincronizan sus parámetros tecnológicos con los parámetros tecnológicos del convertidor. Este proceso se conoce como adaptación de datos. El tipo de componente *DataAdaptation* proporciona la información necesaria al objeto tecnológico mediante la comunicación a cíclica del registro de datos, este es siempre el último componente que se agrega a una unidad simulada.

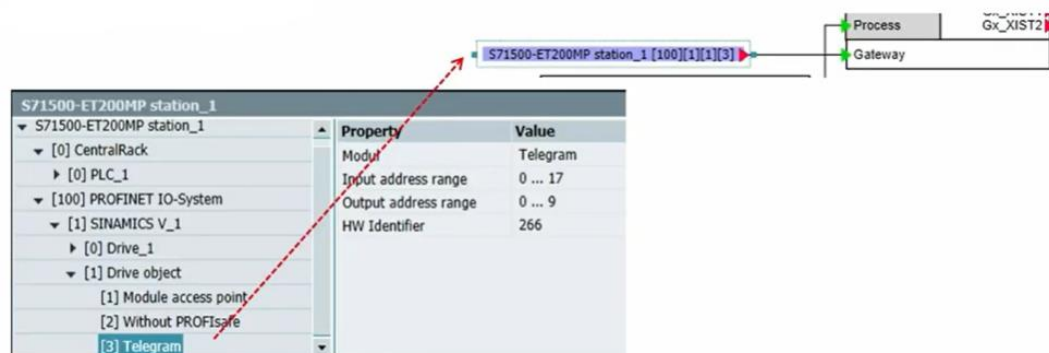


Figura VI.25 Configuración del bloque Gateway.

VI.2.4 TIA Portal Objeto Tecnológico

Una vez dado de alta el proyecto en el software de TIA Portal se busca en el navegador del proyecto la carpeta de bloques tecnológicos como se ve en la figura VI.26, dentro de este apartado se tiene el acceso al asistente para configurar el eje. El objeto tecnológico permite controlar ejes con distintas tecnologías de control de estado sólido, se selecciona el de posicionamiento.

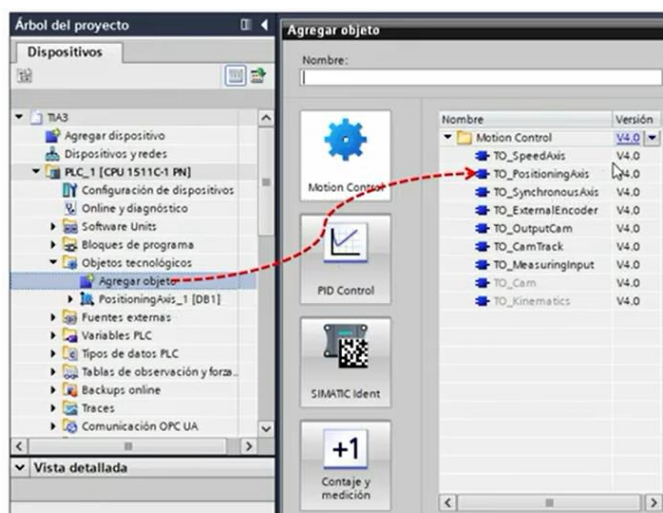


Figura VI.26 Menú objetos tecnológicos.

Un objeto tecnológico de tipo de posición tiene una funcionalidad genérica, pero se puede parametrizar y configurar para una aplicación concreta. Esta operación se realiza desde la carpeta/pestaña “configuración”.

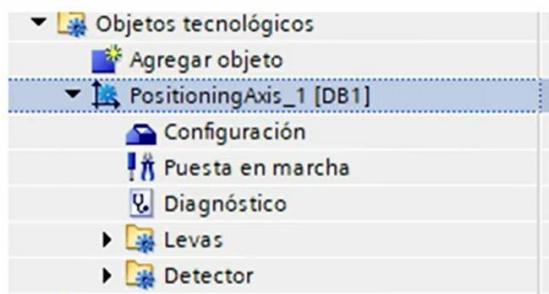


Figura VI.27 Parámetros de los objetos tecnológicos.

A continuación, se explicará de manera detallada como se realiza la configuración de los Parámetros básicos de los objetos tecnológicos:

En este primer apartado se especifica el tipo de eje (lineal o rotativo), se seleccionan las unidades de medida a utilizar. Para el ejemplo descrito se seleccionará rotativo, ya que se tienen dos articulaciones rotacionales en el brazo robótico.

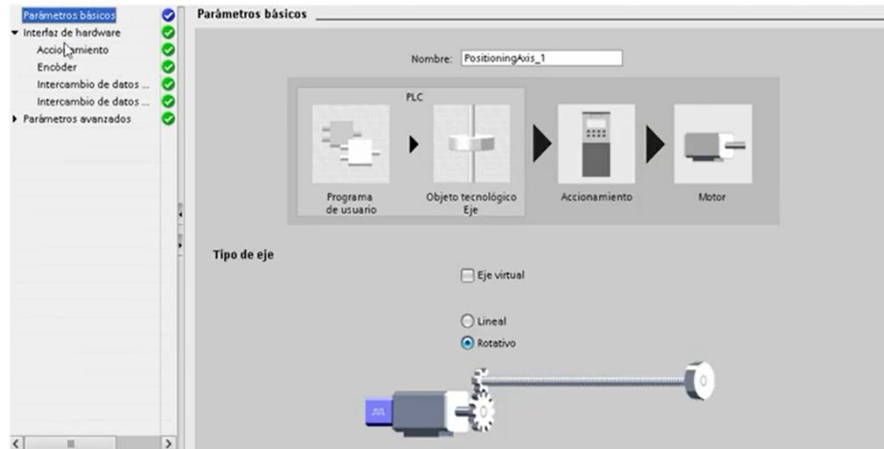


Figura VI.28 Elección de tipo de eje Rotativo.

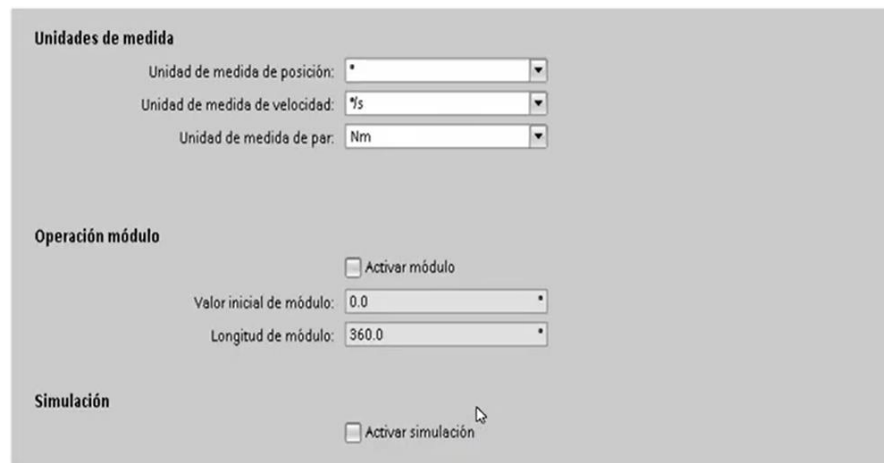


Figura VI.29 Unidades de medida.

1. Interfaz de hardware:

La configuración en esta pestaña es el accionamiento, después el tipo de encoder que se usará y el tipo de intercambio de datos que serán necesarios, la configuración requerida para el proyecto descrito es la que se observa en las siguientes figuras, donde se remarcan las elecciones en el recuadro rojo, por ejemplo el tipo de accionamiento: PROFIDrive, etc.

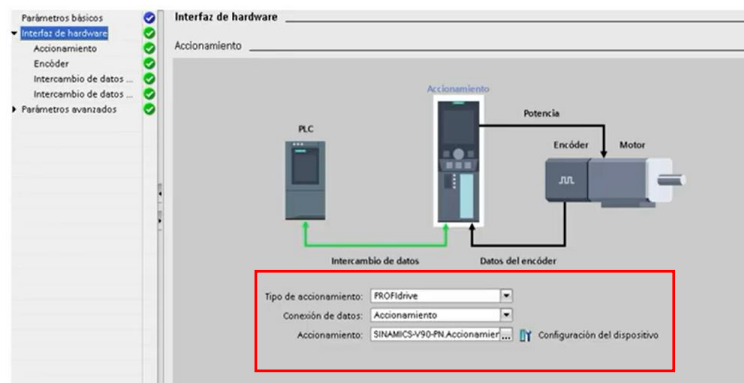


Figura VI.30 Configuración de accionamiento.

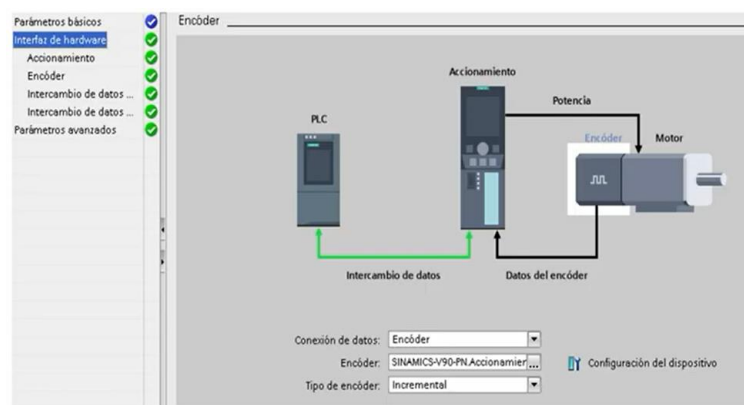


Figura VI.31 Configuración del encoder.

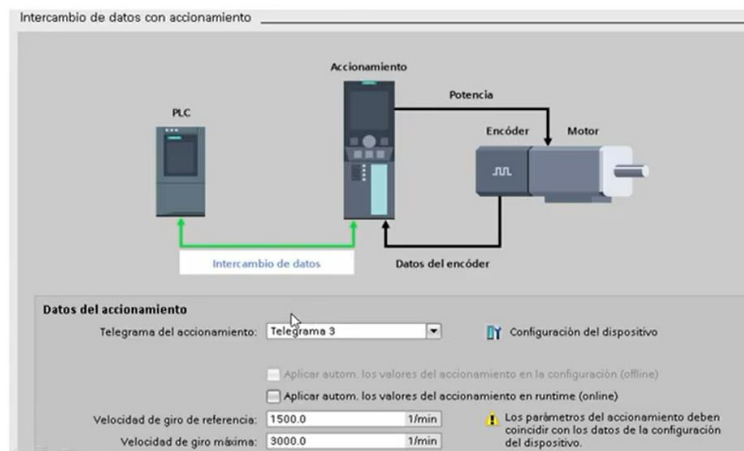


Figura VI.32 Intercambio de datos con accionamiento.

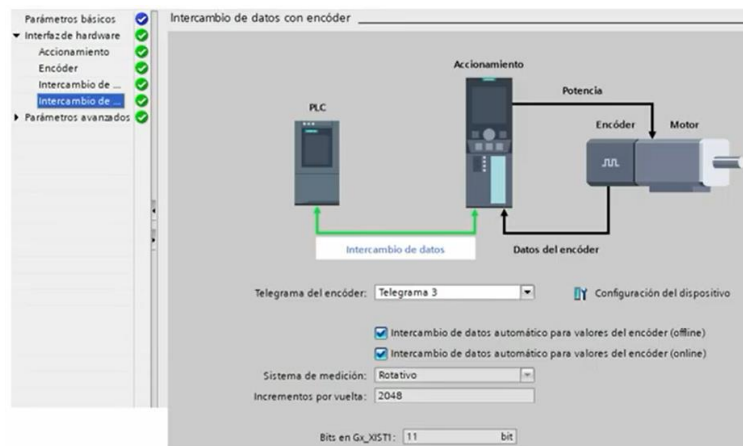


Figura VI.33 Intercambio de datos con el encoder.

2. Parámetros avanzados:

La configuración en esta pestaña es la mecánica, el pre ajuste de dinámica y el parado de emergencia, la configuración necesaria para este proyecto es la que se puede ver en las siguientes figuras VI.34, VI.35 y V.36.



Figura VI.34 Mecánica.

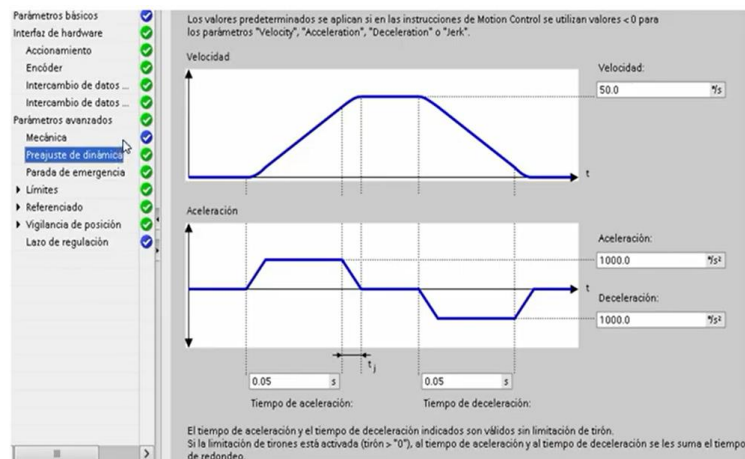


Figura VI.35 Pre ajuste de dinámica.

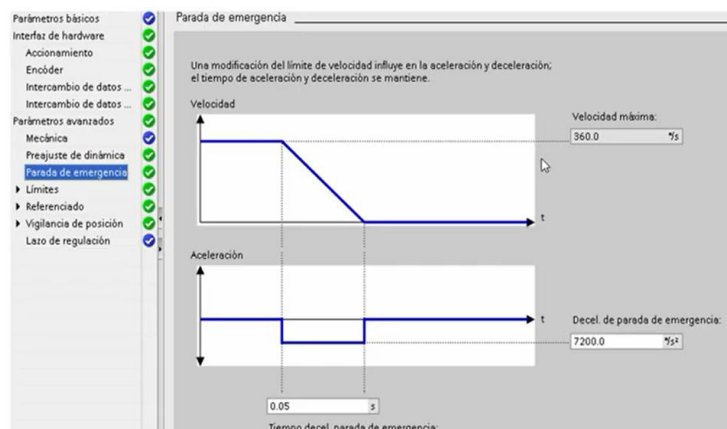


Figura VI.36 Parada de emergencia.

3. Límites:

En este apartado por el momento no es necesario cambiar todos los límites que se tienen, solo uno límites dinámicos.

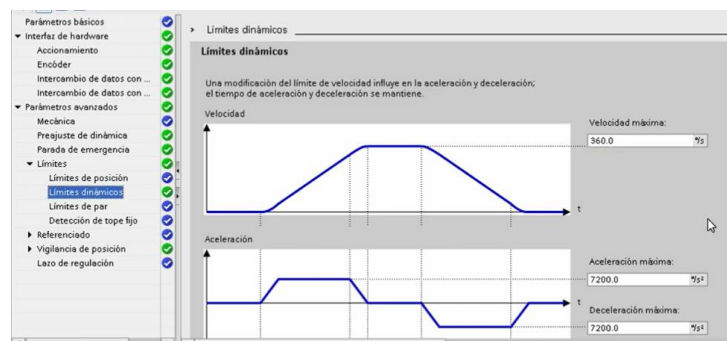


Figura VI.37 Límites dinámicos.

4. Referenciado:

El referenciado vincula la posición del objeto tecnológico a una posición mecánica. Existen dos formas de referenciado:

Referenciado pasivo: En el referenciado pasivo, la instrucción de Motion Control *Mc_Home* no realiza ningún movimiento de referencia. El desplazamiento necesario para ello deber ser realizado por el usuario con otras instrucciones *Motion Control*. El eje se referencia al detectarse el sensor del punto de referencia.

Referenciado activo: La aproximación al punto de referencia se ejecuta automáticamente. Para un eje lineal comúnmente se hace un referenciado activo y para una cinta es referenciado pasivo.

Para este proyecto, el referenciado que se empleará es el que se puede ver en la siguiente figura:

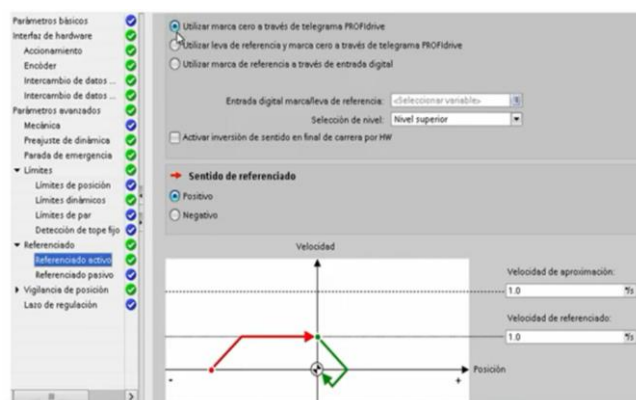


Figura VI.38 Referenciado activo.

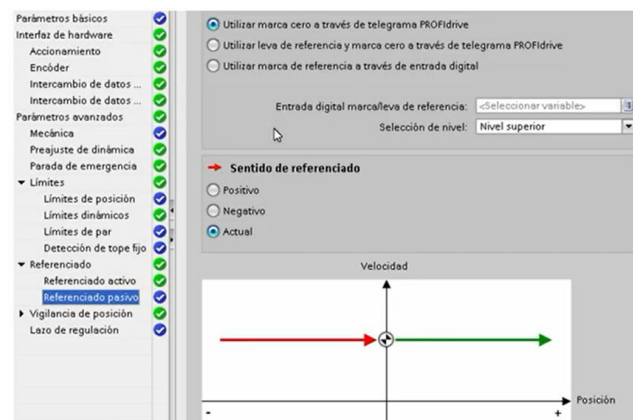


Figura VI.39 Referenciado pasivo.

5. Vigilancia de posición:

Lo configurable en esta pestaña es la vigilancia de posicionamiento, el error de seguimiento y la señal de velocidad cero, la configuración necesaria para este proyecto es la que se puede ver a continuación:

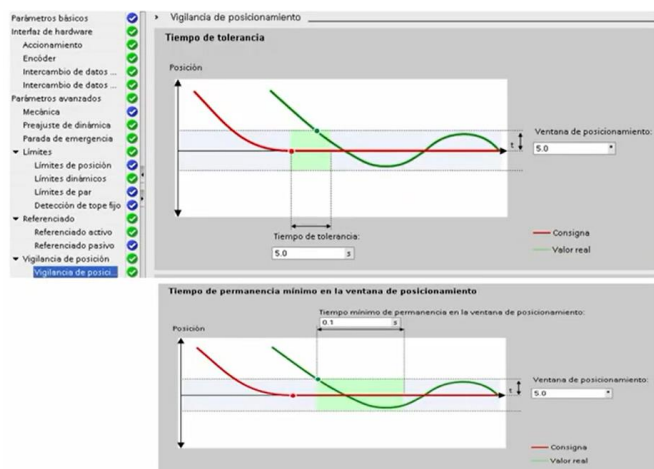


Figura VI.40 Vigilancia de posicionamiento.

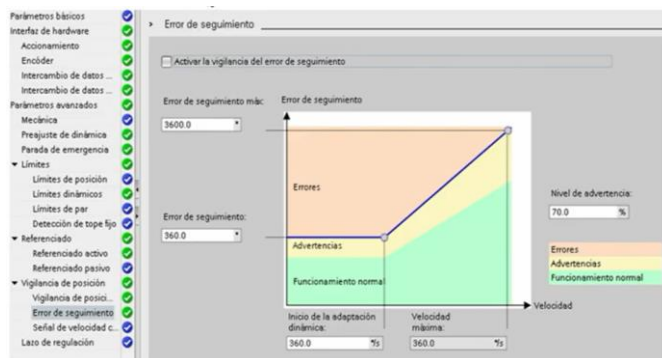


Figura VI.41 Error de seguimiento desactivado.

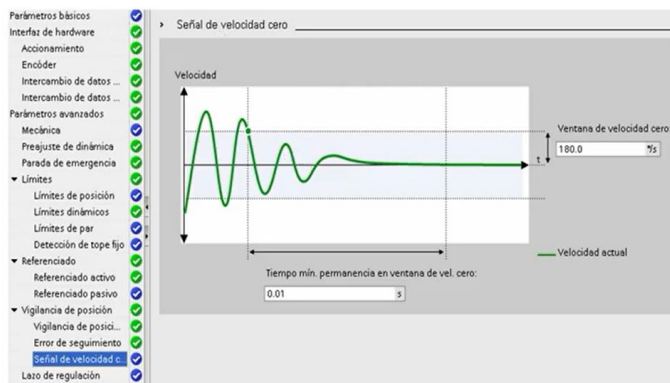


Figura VI.42 Señal de velocidad cero.

6. Factor de ganancia

Se podría ajustar la ganancia en el lazo de regulación si se quisiera hacer un ajuste más fino del movimiento. En este caso se deja el valor de 10.

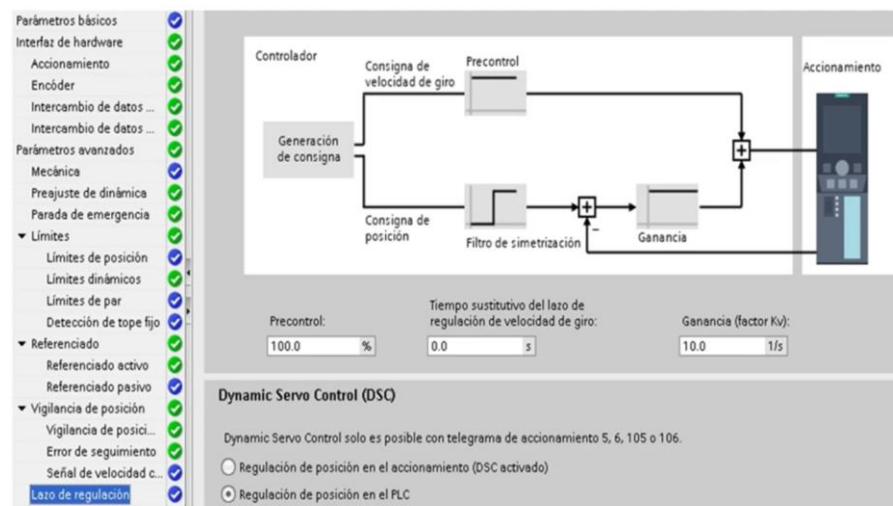


Figura VI.43 Lazo de regulación.

VI.2.5 Configuración TIA Portal (accionamientos Sinamic V90)

Se crea un nuevo proyecto en TIA Portal, el nombre de este proyecto será ROBOT_SCARA.

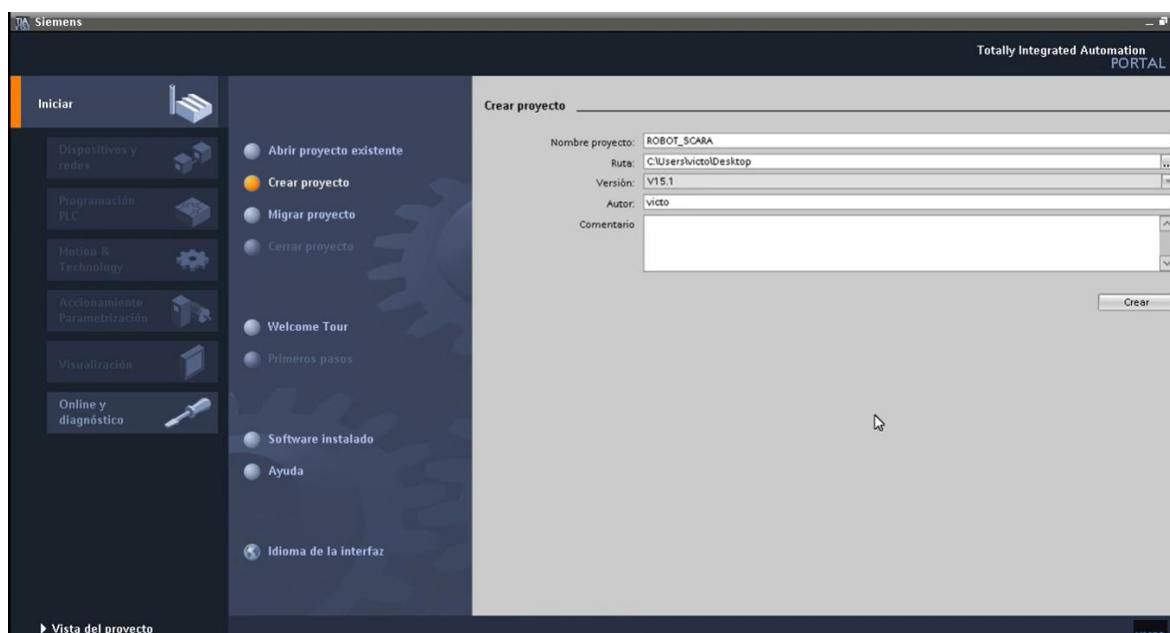


Figura VI.44 Interfaz inicial de TIA Portal.

Al crearse el proyecto se elige configurar un dispositivo, seguido de esto se debe de seleccionar el PLC que se utilizará, en este caso es un CPU 1511-1 PN, se desplegarán tres opciones, se debe de seleccionar la tercera.

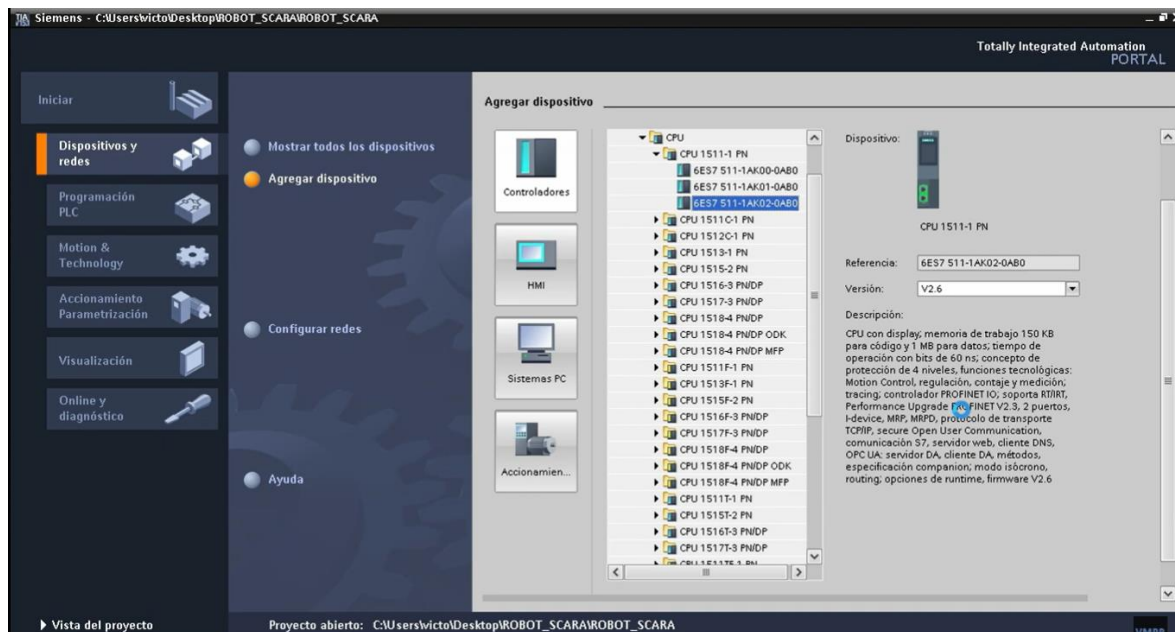


Figura VI.45 Selección de PLC.

Después de comprobar la dirección Ethernet del PLC asignada automáticamente, la cual es 192.168.0.1.

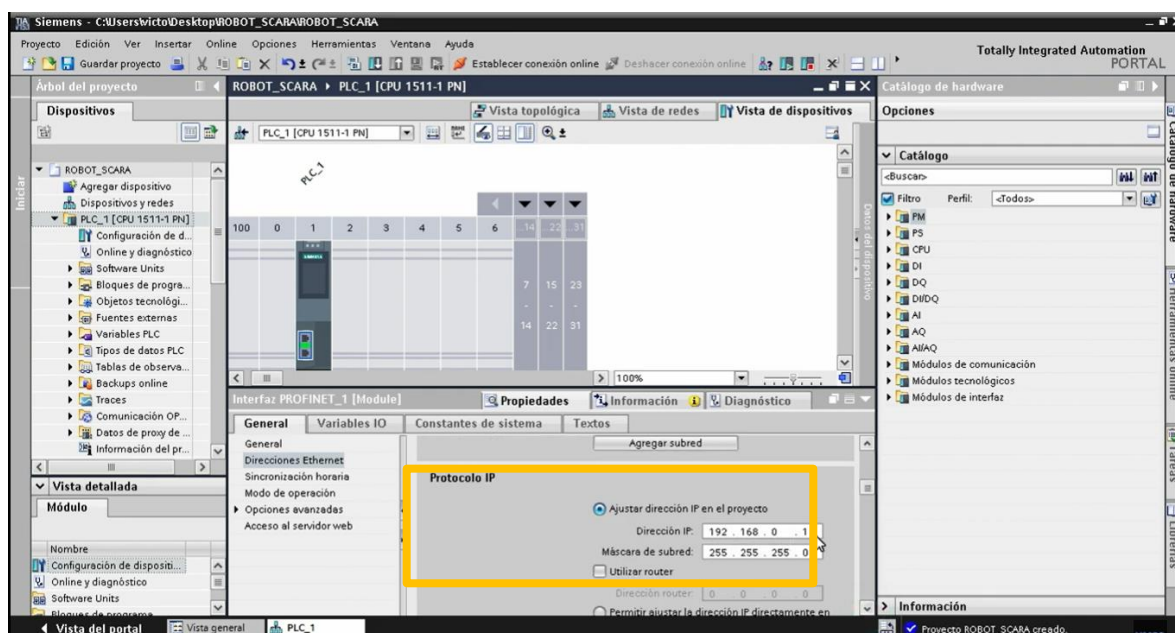


Figura VI.46 Dirección del PLC.

Es necesario agregar una pantalla HMI, la elegida para el proyecto es una *KTP400 Comfort Portrait*, solo aparece una opción.

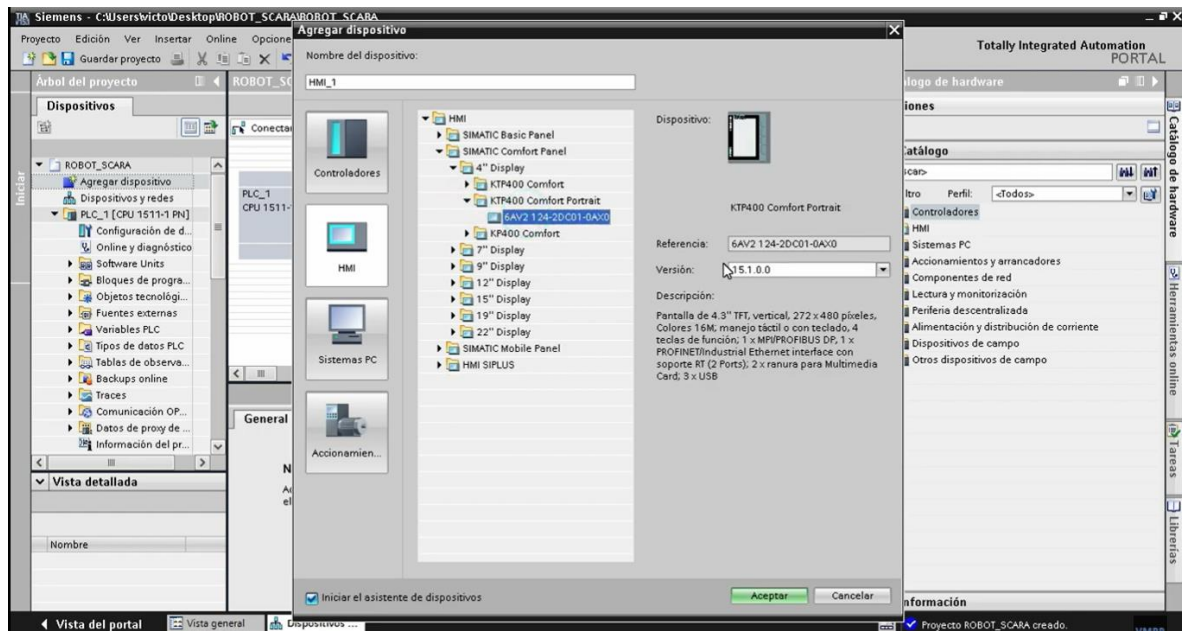


Figura VI.47 Selección de HMI.

Una vez que se agrega al proyecto se debe conectar al PLC y comprobar la dirección IP asignada automáticamente, la cual es 192.168.0.2.

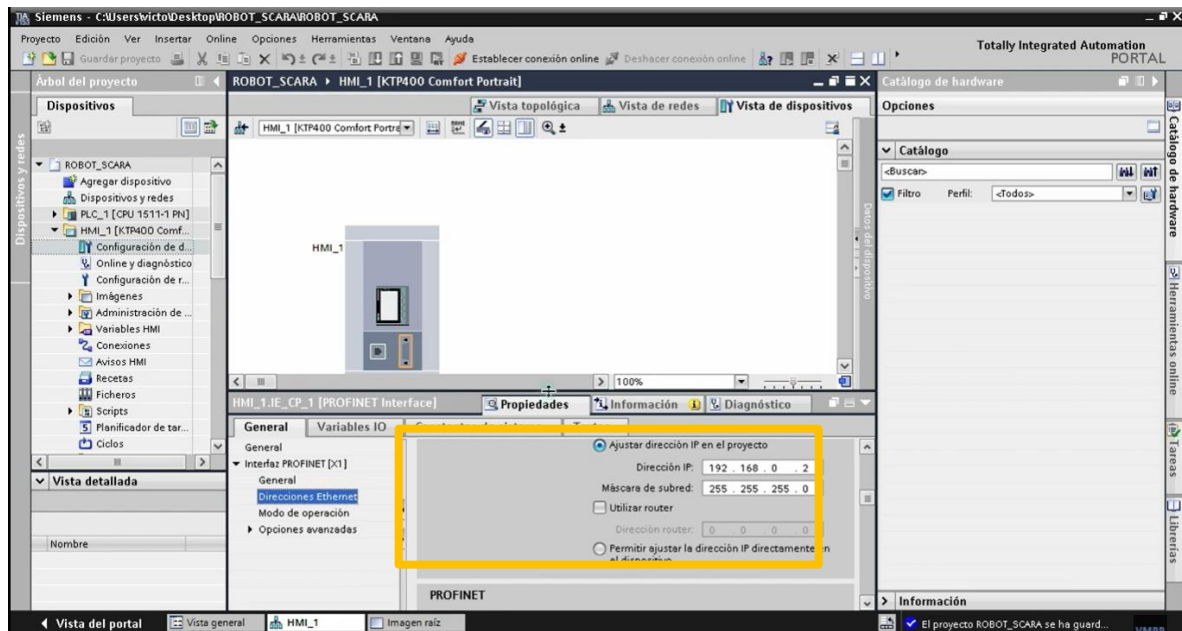


Figura VI.48 Dirección IP del HMI.

Una vez agregados los dispositivos al proyecto, es necesario en el apartado de protección del mismo, habilitar seleccionando la casilla: *permitir simulación al compilar bloques*, ya que si ésta no está activa, al momento de querer hacer la simulación conjunta SIMIT-PLC Virtual y NX MCD se presentará un error; para desplegar esta ventana se debe de dar clic derecho en el nombre del proyecto y en la caja de propiedades que se abre en la pestaña de *Protección*, se encuentra la opción.

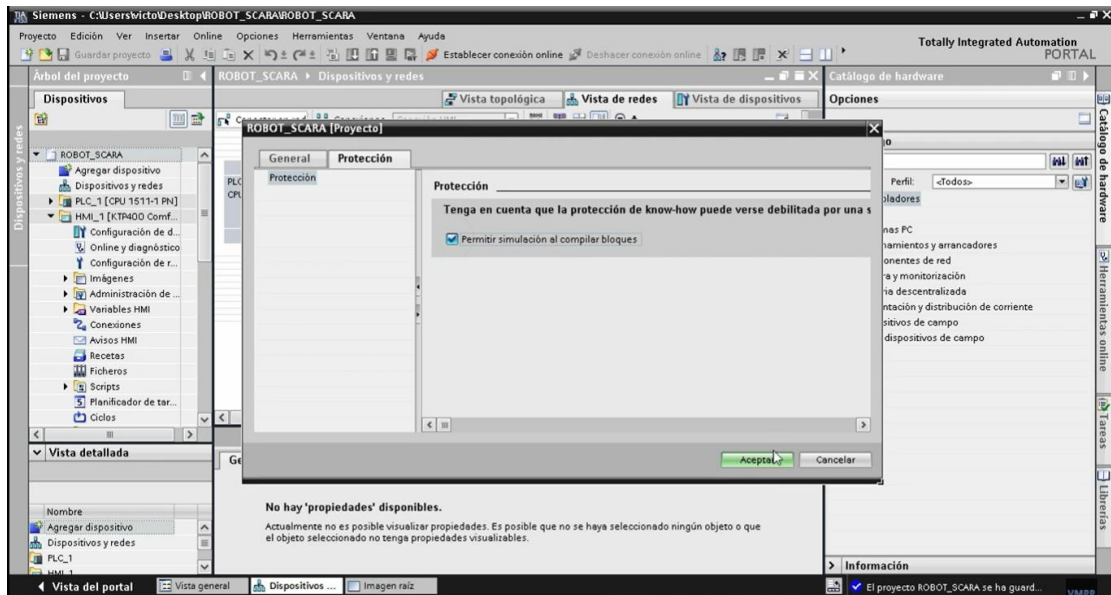


Figura VI.49 Protección del proyecto.

Una vez que se tenga el PLC y la HMI con la conexión con las direcciones deseadas, lo que se prosigue es agregar los accionamientos Sinamic V90; estando en la vista de redes, ir al apartado que dice catálogo de hardware y buscar SINAMICS V90 PN V1.0.

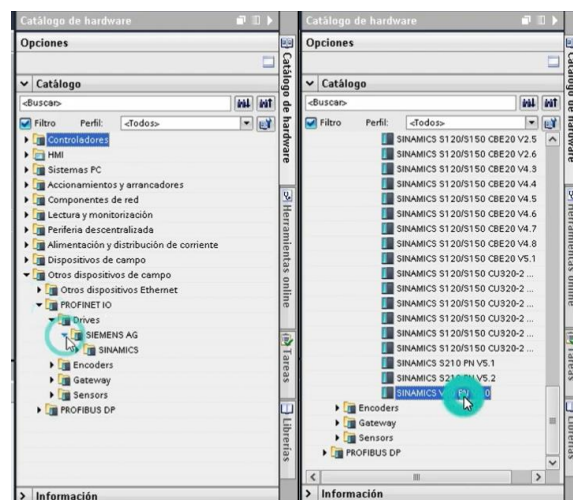


Figura VI.50 Selección de accionamiento.

En un principio tendrá un texto subrayado que dirá *no asignado*, es necesario seleccionar que el controlador sea el PLC que se agregó al principio.

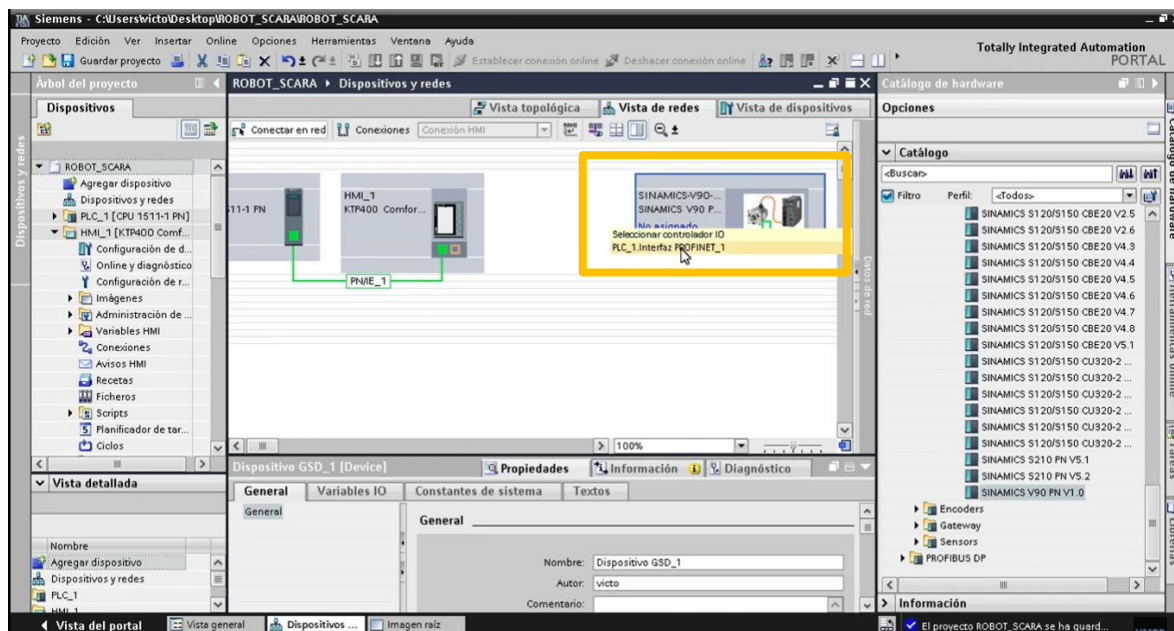


Figura VI.51 Conexión de accionamiento con PLC.

Después se revisa la dirección IP que tiene el controlador Sinamic V90, la cual debe estar en el mismo segmento que los otros dispositivos, el software le asignó automáticamente la dirección: 192.168.0.3.

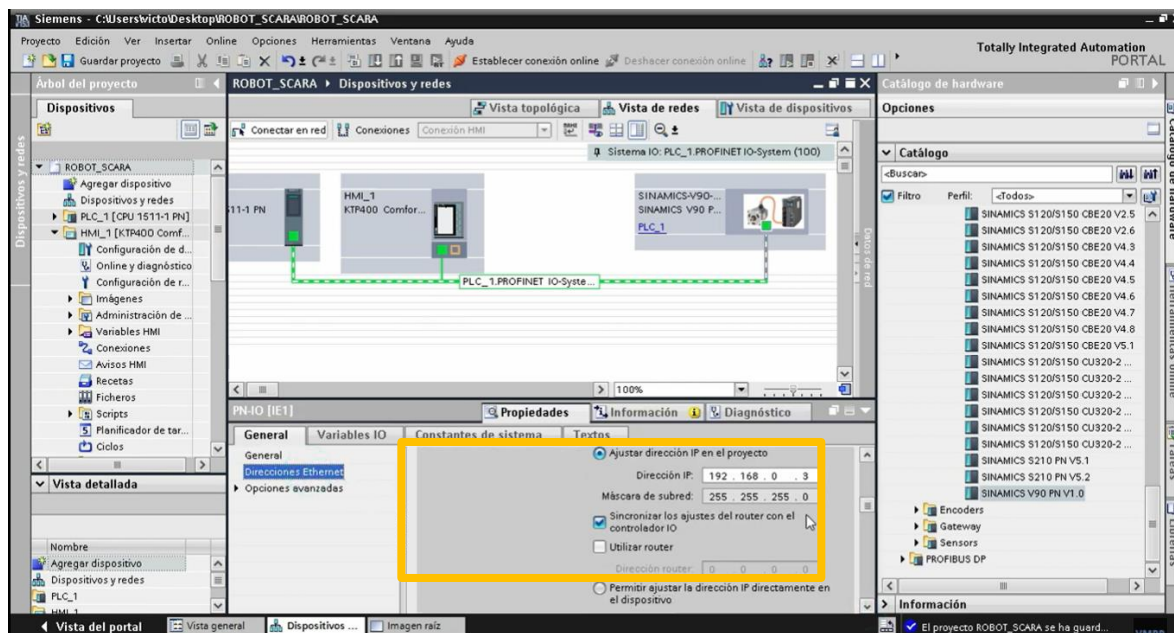


Figura VI.52 Dirección de accionamiento.

Se selecciona el rectángulo gris del V90 y aparecerá una pestaña que se llama *vista de dispositivos*, de igual manera las opciones del catálogo cambiarán, aquí es necesario seleccionar el Telegrama estándar 3, P2D-5/9.

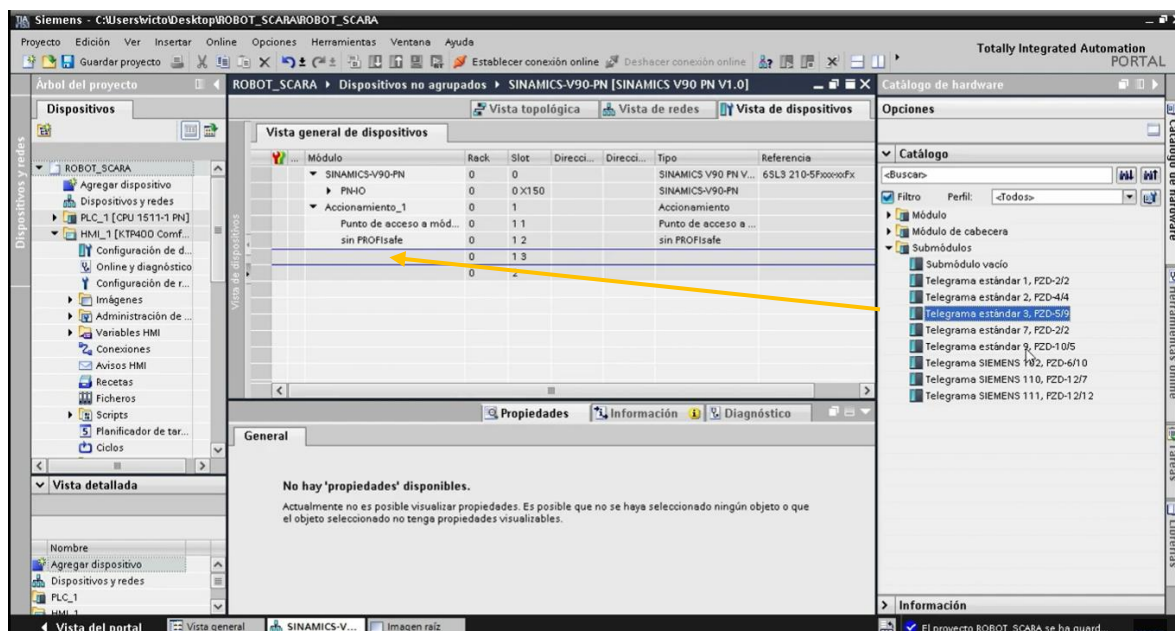


Figura VI.53 Telegrama de accionamiento.

Ya que contamos con dos accionamientos, se realizan los mismos pasos para agregar uno nuevo, el cual debe de tener la dirección IP 192.168.0.4.

El telegrama estándar 3 controla velocidad y posición. Por otra parte, hay que recordar que se usa el software SIMIT en su versión demo con una capacidad de 30 variables de entradas y salidas (E/S), por lo que para los acoplamientos hay que realizar lo más simple su configuración.

VI.2.6 Configuración objetos tecnológicos

Partiendo de la configuración que se explicó en el punto anterior, hay que agregar los objetos tecnológicos necesarios, en este caso son dos uno por cada accionamiento, lo que se observa en la figura VI. 54, es lo que se debe de tener hasta al momento, las direcciones deberían de ser las mismas.

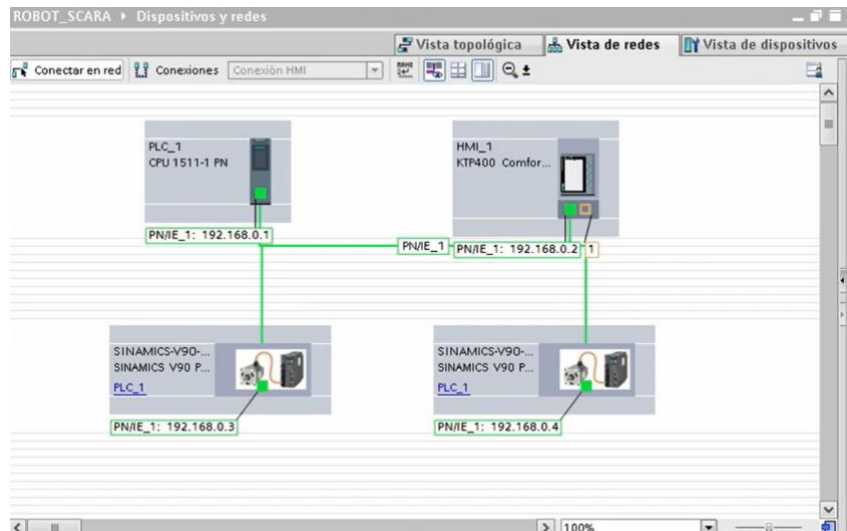


Figura VI.54 Vista de redes.

Es importante ver cómo es que se agrega el objeto tecnológico al igual que el tipo de objeto tecnológico, se deja el direccionamiento automático, esto se puede ver en la figura VI.55.

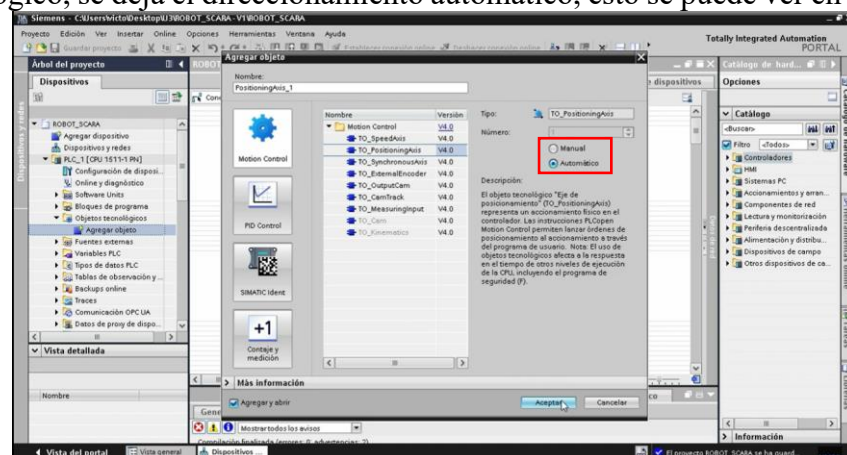


Figura VI.55 Selección del objeto tecnológico.

Una vez que se agregan al proyecto, es necesario hacer la configuración, la cual se explicó en el apartado VI.2.4 TIA Portal Objeto Tecnológico. Es importante señalar que se deben de agregar dos objetos tecnológicos correspondientes a los ejes: Axis 1 y el Axis 2, los dos se configuran de la misma manera.

Una realizada la configuración se debe revisar el rango de direcciones asignadas de manera automática, esto se ve en la misma pestaña donde se agregó el tipo de telegrama que se requería en cada accionamiento, es decir, en la vista de redes, para ello dar dos clics sobre el dispositivo Sinamic y aparecerá una nueva pestaña: la de Vista general de dispositivos, como se observan en las figuras VI.56 y VI.57.

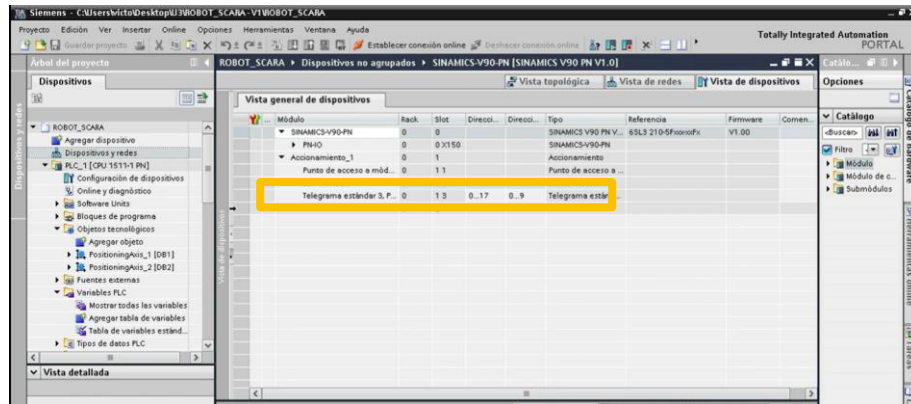


Figura VI.56 Direcciones del primer accionamiento.

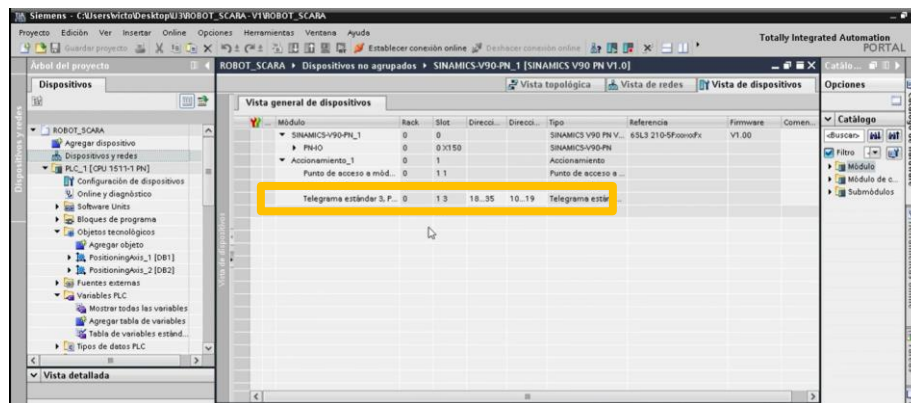


Figura VI.57 Direcciones del segundo accionamiento.

Por último, se comprueba que todas las variables estén en la tabla de variables estándar del proyecto y deben de aparecer las direcciones correspondientes, ver figura VI.58.

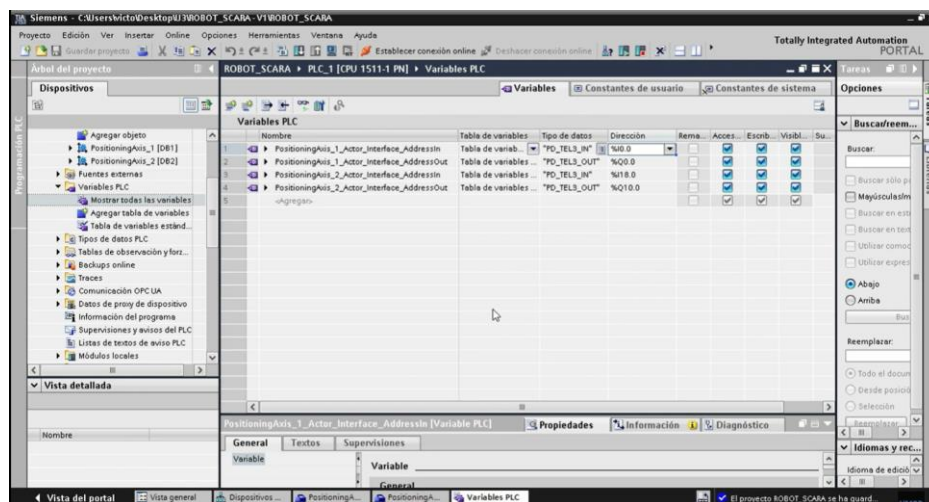


Figura VI.58 Tabla de variables.

VI.2.7 Integración de bloques de control de Motion Control en TIA Portal

- *Bloques de organización para control de movimiento*

Al crear el objeto tecnológico, se crean bloques de organización para procesar los objetos tecnológicos. Se crean los siguientes bloques: **MC-Servo [OB91]** cálculo de controlador de posición y **MC [OB92]** evaluación de las instrucciones de Motion Control, generación de consignas y funcionalidad de monitorización.

Los bloques de organización están protegidos (protección de know-how). El código del programa no se puede cambiar.

- *Ciclo de aplicación*

Puede configurar el ciclo de aplicación en el que se llama **MC-Servo [OB91]** en las propiedades del bloque de organización en “General>Tiempo de ciclo”. El **MC-Servo [OB91]** se llama cíclicamente con el ciclo de aplicación especificado. El ciclo de aplicación seleccionado debe ser lo suficientemente largo para poder procesar los objetos tecnológicos en un ciclo. Si el tiempo de procesamiento de los objetos tecnológicos es mayor que el ciclo de aplicación, se producirán desbordamientos.

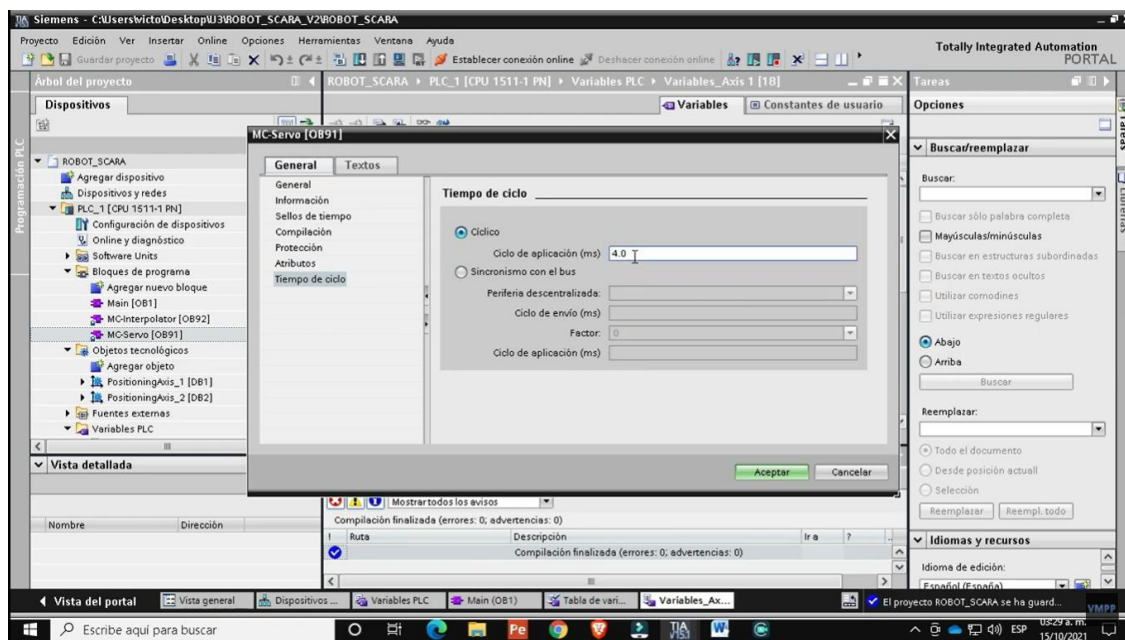


Figura VI.59 Tiempo de ciclo.

- *Integración de los bloques de control*

Una vez que se aclaró el punto anterior, es momento de comenzar con la integración de los bloques de control de Motion Control, se declara una nueva tabla de variables, la cual

llevará por nombre *Variables_Axis1*, esta tendrá 18 variables en las siguientes figuras se pueden ver cuáles son y el tipo de dato correspondiente:

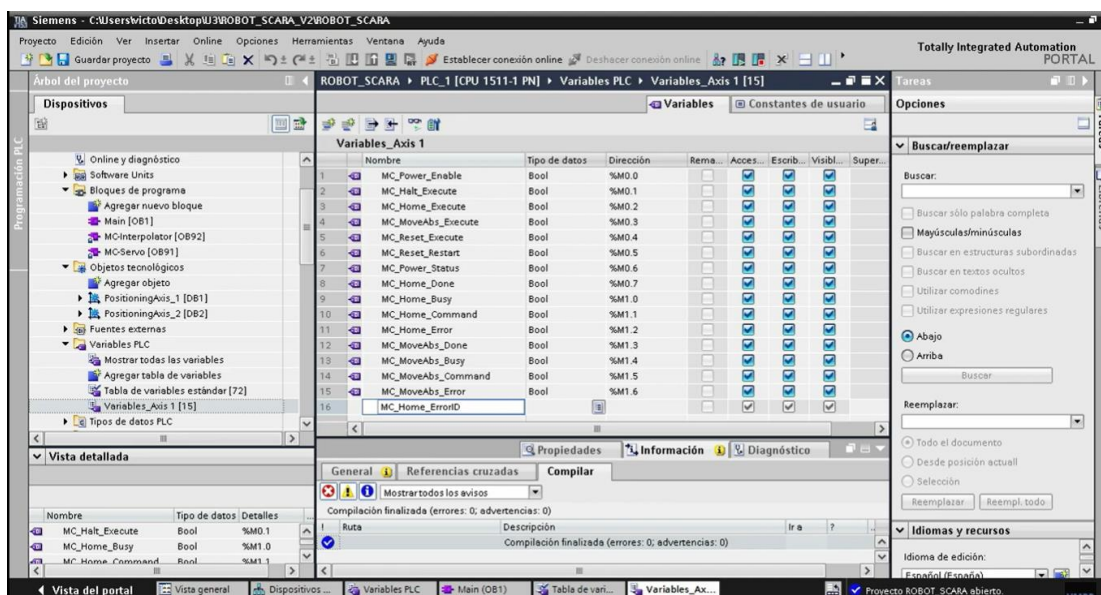


Figura VI.60 Primera parte de la tabla de variables Axis 1.

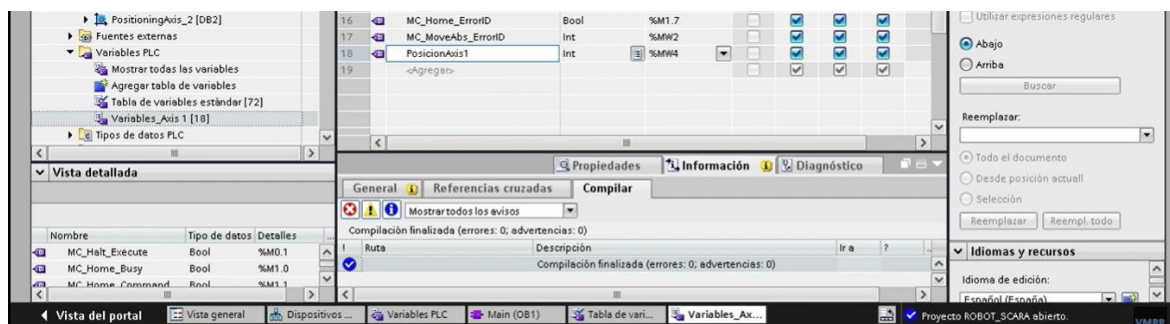


Figura VI.61 Segunda parte de la tabla de variables Axis 1.

Nota: La variable 16 también debe ser de tipo Int y tendrá la dirección MW6, ignorar el hecho de que en la tabla se encuentra como tipo Bool.

Una vez terminada la tabla de variables, se agrega un nuevo bloque de función al cual se le llamará *Modo_Manual*, es importante que el bloque creado se mande llamar en el bloque lógico main [OB1], el cual se ejecuta todo el tiempo y es en él donde deben llamarse los demás bloques de subrutinas que no sean de objetos tecnológicos.

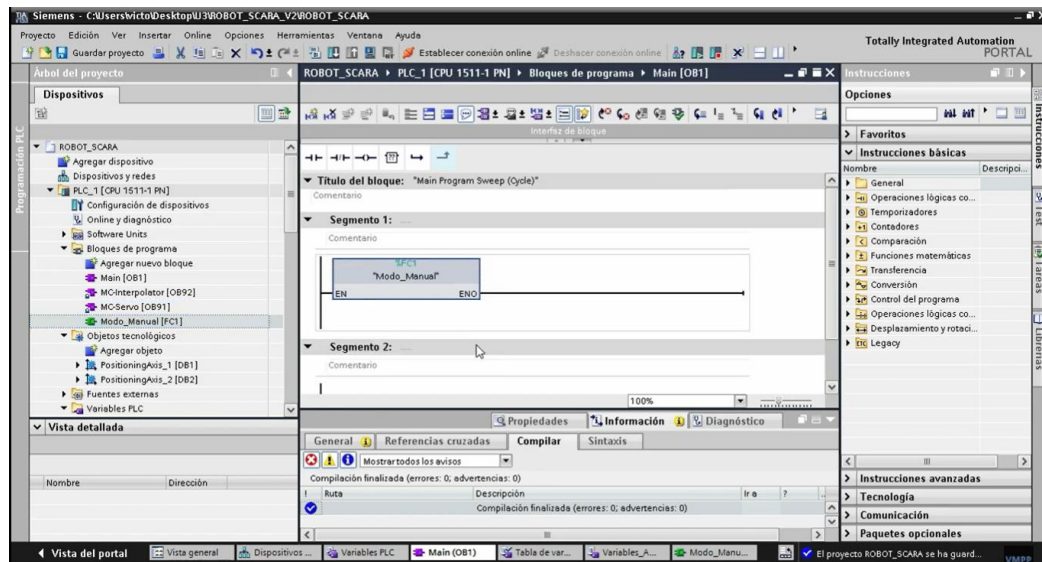


Figura VI.62 Bloque main con el bloque FC (código del Modo_Manual).

Dentro del bloque llamado *Modo_Manual* se mandarían a llamar cinco bloques, es decir, que se tendrían cinco segmentos, estos segmentos se pueden ver en las siguientes figuras al igual que las variables que se tienen que colocar en las entradas de cada bloque, la variable que se pone en la entrada *Axis* del bloque se arrastra directamente desde el árbol de proyecto, el cual es el objeto tecnológico que se declaró. Las funciones de motion, se encuentran en la ventana de instrucciones tecnológicas en la carpeta de *Motion Control*, para cada eje deben arrastrarse de este menú al editor del programa, no se recomienda copiar el código, ya que el software enumera automáticamente los bloques.

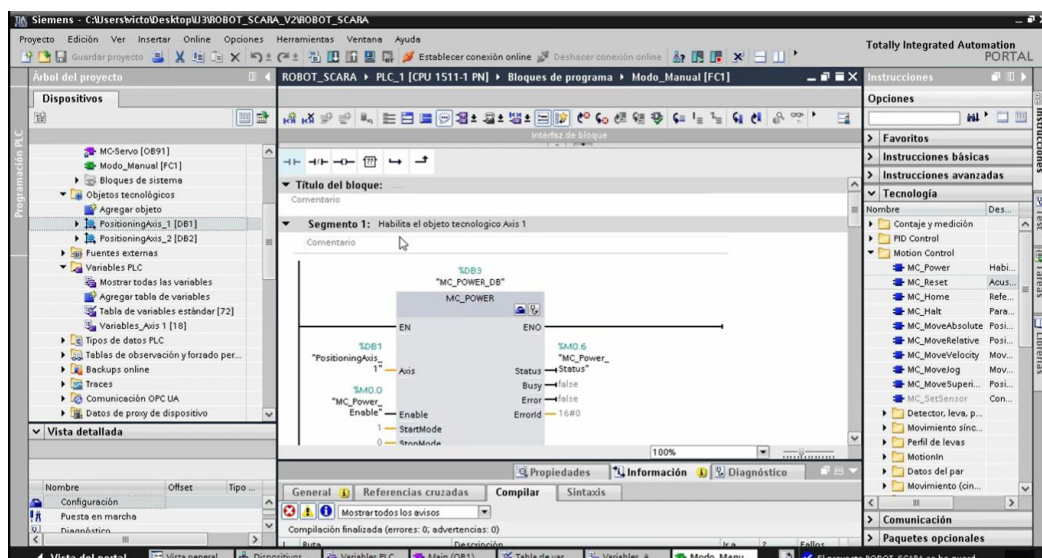


Figura VI.63 Bloque MC_Power

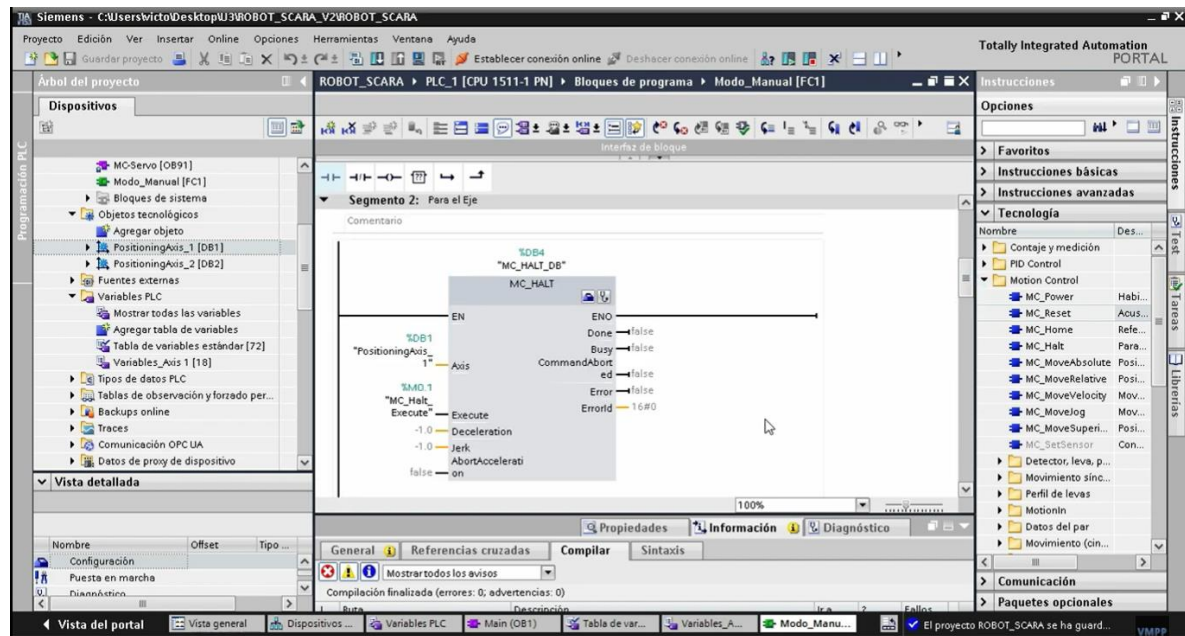


Figura VI.64 Bloque MC_Halt

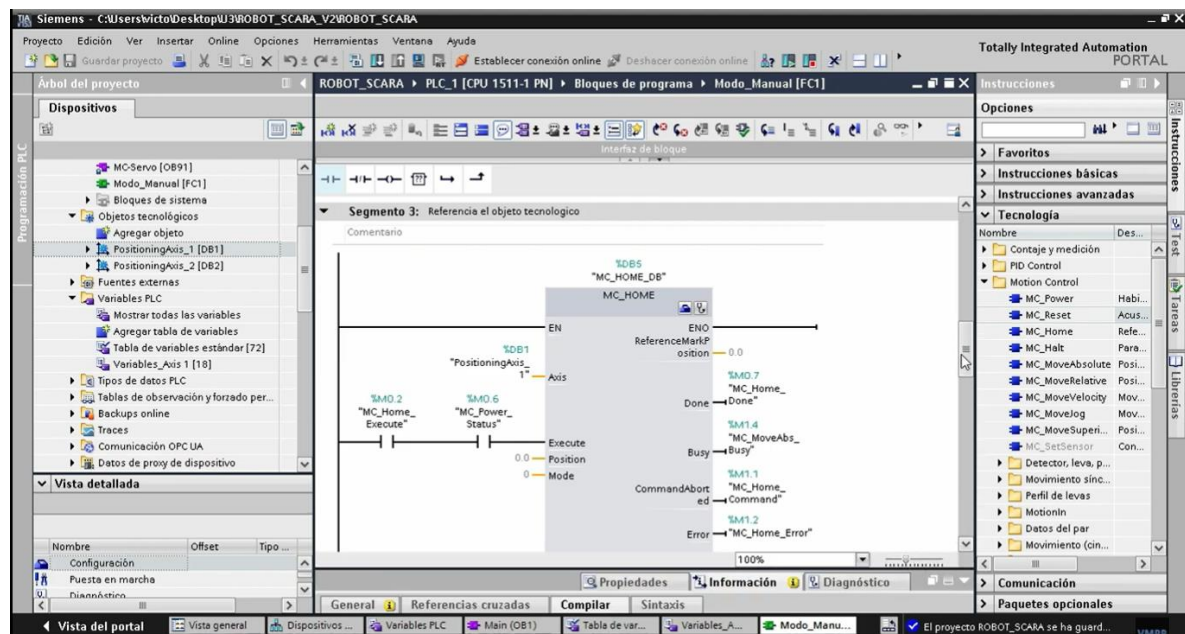


Figura VI.65 Bloque MC_Home

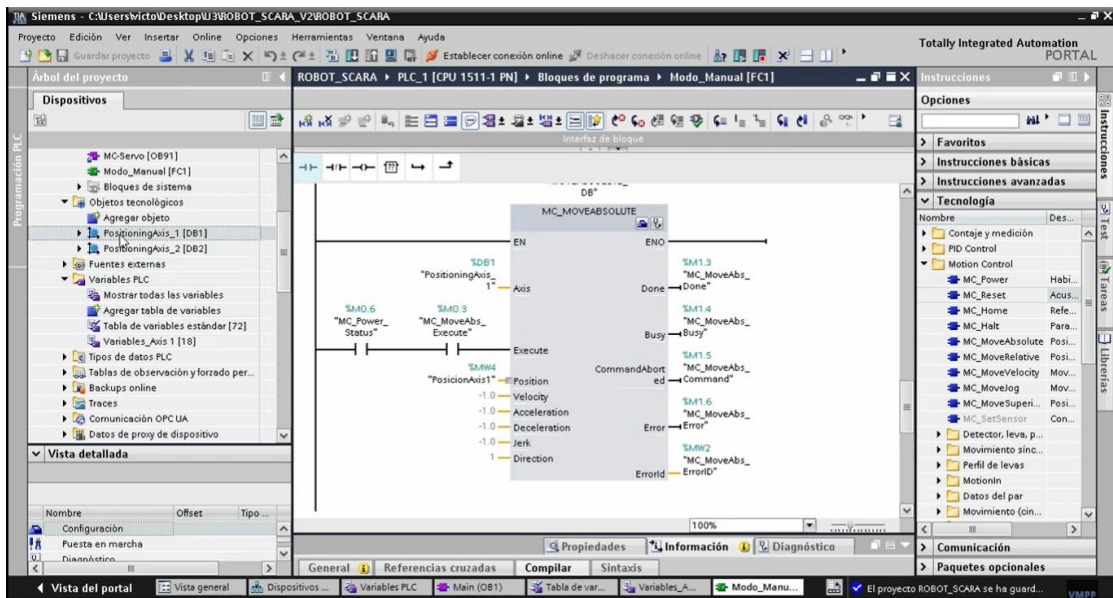


Figura VI.66 Bloque MC_MoveAbsolute

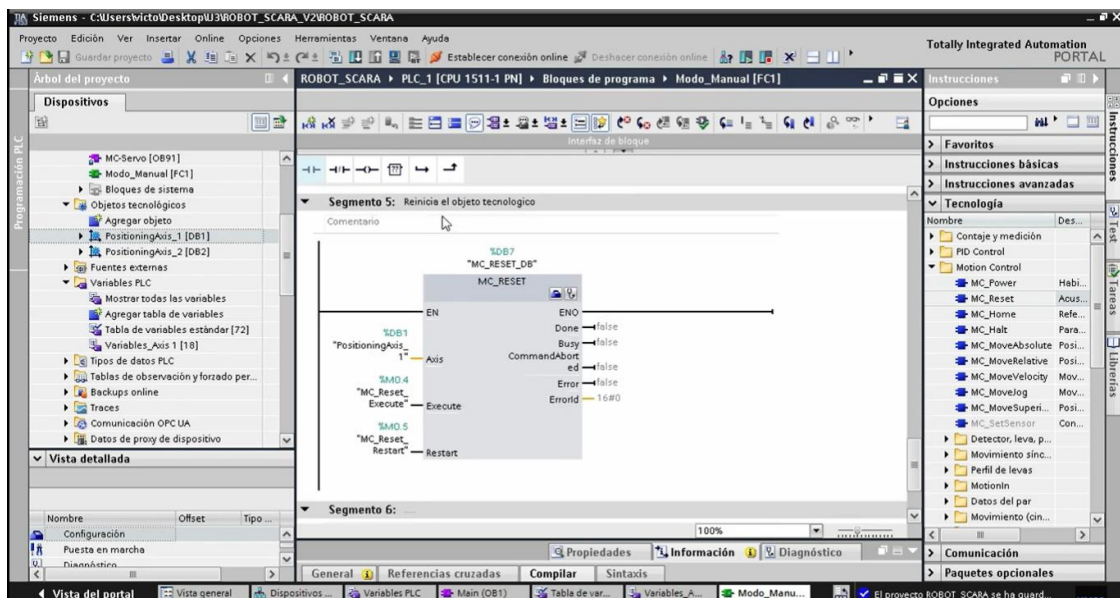


Figura VI.67 Bloque MC_Reset.

Es necesario hacer lo mismo para el Axis 2, declarar la tabla de variables con sus 18 variables y en el bloque de modo manual agregar de nuevo los cinco bloques necesarios para el control del eje 2, obviamente estos bloques tendrán las variables del Axis 2, con el mismo nombre solo agregando un 2 al final.

Para el diseño de las pantallas o imágenes como se les conoce en TIA Portal que tendrá la HMI, en este caso se declaran dos imágenes, una para el *Modo_Manual* y otra para el *Modo_automático*, pero primeramente se trabajará con la imagen del modo manual. Para ello

se necesitarán seis botones y una entrada de datos enteros para cada uno de los ejes, en la siguiente figura se presenta un ejemplo de lo que se necesita configurar para uno de los ejes.

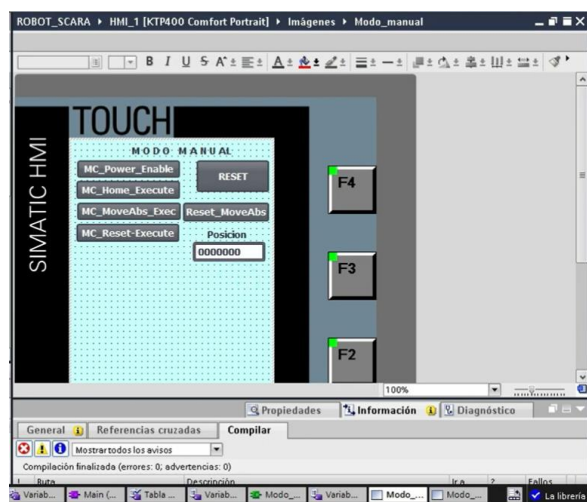


Figura VI.68 Diseño de la HMI.

VI.2.8 Desarrollo de interfaz de HMI

Primeramente se crea una nueva tabla de variables generales, en la cual se pondrán las dos variables como marcas que se pueden ver en la figura VI.69, ya se tiene un bloque de función que tiene por nombre *Modo_Manual* y el de *Modo_automático*, por el momento solo se describe el modo manual. En el bloque *main* se le pondrá un contacto normalmente abierto a cada uno de los bloques (*manual* y *automático*) para que estos solo se activen cuando se presionen los botones en la HMI con el mismo nombre, para lo cual se declaran las marcas: *Mod_Manual* y *Mod_Automático*.

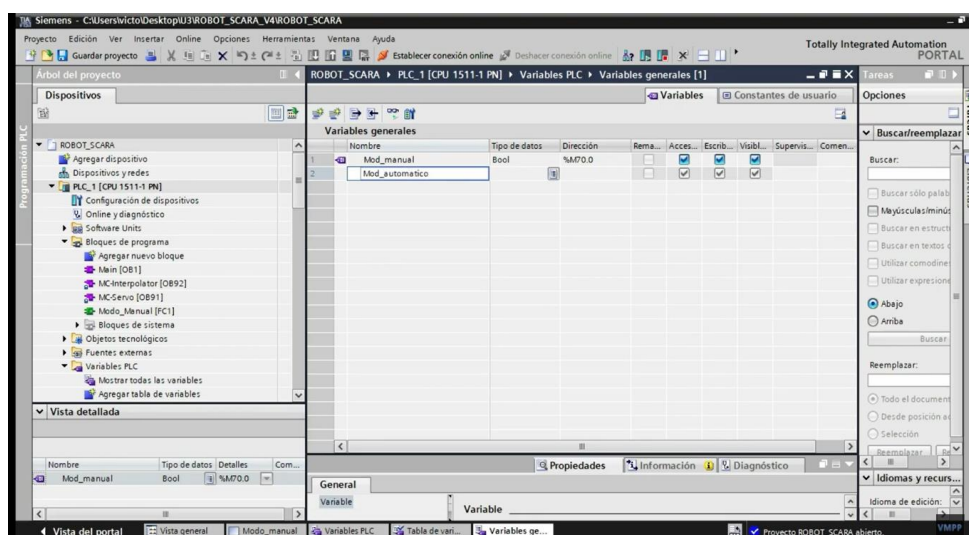


Figura VI.69 Tabla de variables generales.

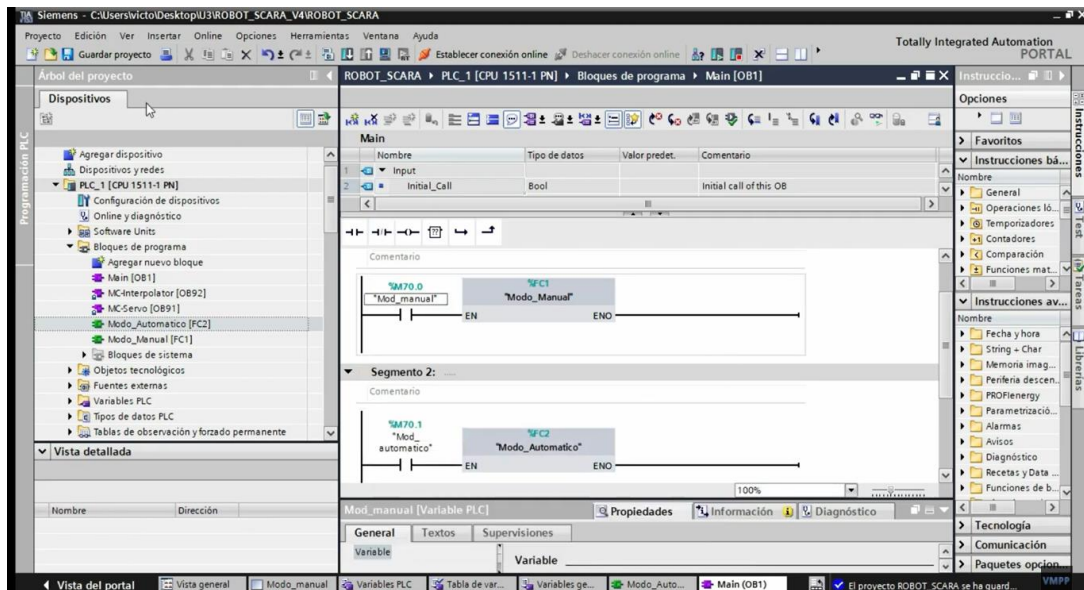


Figura VI.70 Bloque main con las modificaciones necesarias.

En el HMI se agregan tres botones, los cuales serán para habilitar alguno de los dos modos posibles, otra para poder cambiar de pantalla y un último para poder cerrar la ejecución del HMI y facilitar las validaciones en simulación del control de manera conjunta con *NX MCD* y *SIMIT*.

El botón para habilitar tiene dos eventos que se activan al pulsar, ambos son eventos que conmutan los estados de los bits de las marcas correspondientes, las funciones son: uno es para activar y otro para desactivar, la configuración depende de la pantalla en la que se active el botón, es decir, en la pantalla de *modo manual* al pulsar el botón de *modo manual* se activa el bit *Mod_manual* y se desactiva el *Mod_automático*, en la pantalla de *modo automático* será lo contrario, la configuración del botón se observa en la siguiente figura:

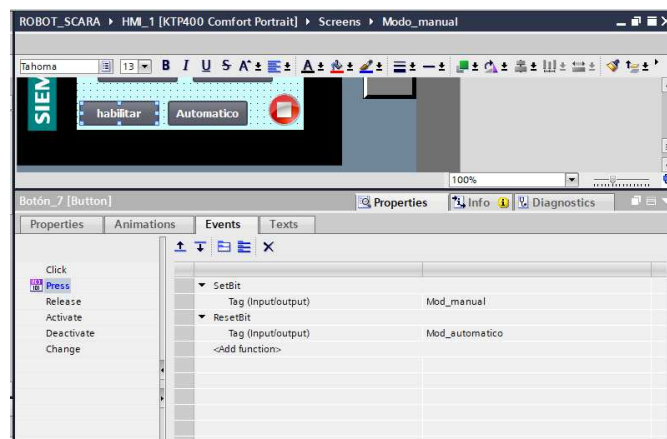


Figura VI.71 Configuración del botón para habilitar.

Los primeros dos botones explicados no tienen características especiales visualmente, pero el botón para cerrar el HMI sí, al cual se le agregó una imagen para poder cambiarle la apariencia, es posible observar los detalles de su configuración en las figuras VI.72, VI.73 y VI.74, la función asignada es la de cerrar simulación con el evento al pulsar (press).

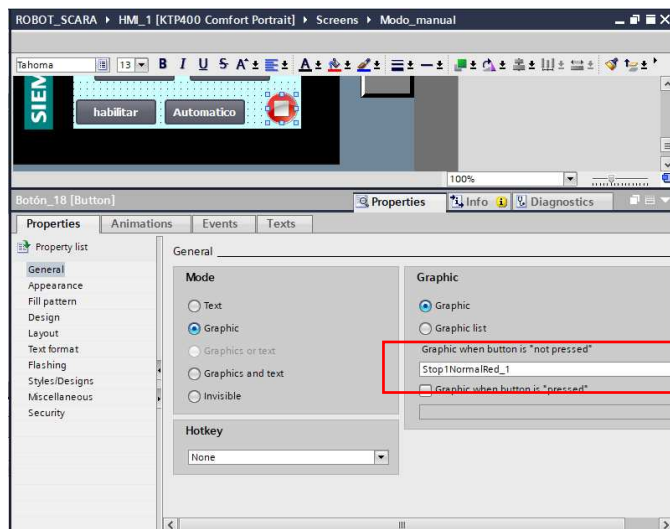


Figura VI.72 Primera parte de la configuración visual del botón para cerrar la simulación.

En el recuadro de la figura anterior, se da clic en la flecha y se abre un cuadro de diálogo para buscar donde se encuentra la imagen, en apariencia elegimos en *fill pattern* la opción de *Transparent* (transparente) y se deja el borde en 0, como se ve en la figura VI.73.

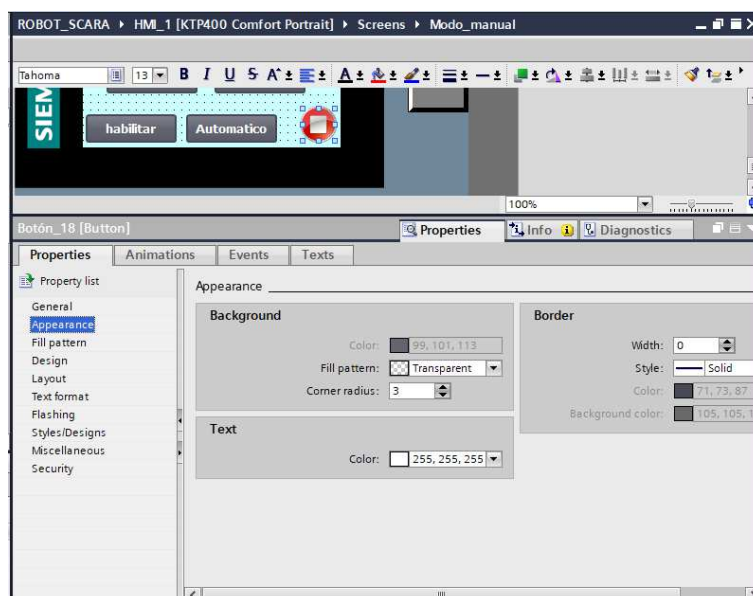


Figura VI.73 Segunda parte de la configuración visual del botón para cerrar la simulación.

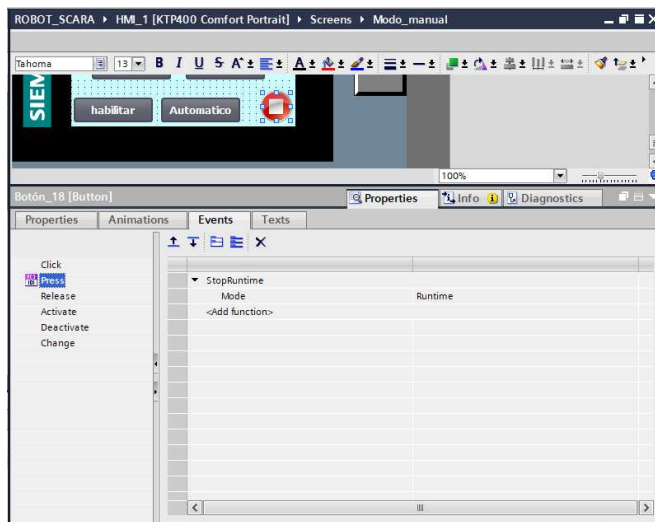


Figura VI.74 Configuración del evento del botón para cerrar la simulación.

Se requieren otros botones como se puede observar en la figura VI.75 en las dos pantallas. La pantalla de *modo manual* tiene bastantes botones, pero solo tiene dos pequeñas imágenes, la pantalla de *modo automático* tiene menos botones, pero esta cuenta con tres imágenes: la del brazo y dos cuadrados que sirven como guía de las posibles posiciones donde se coloca la pieza (producto), claro que para fines de validaciones solo se habilitarán dos de cada lado, son los que se observan como círculos en color rojo, para lo cual se emplearon botones para realizar dicha función, éstos se obtuvieron del menú de librerías de las pestañas verticales del lado derecho del TIA Portal, dichas pestañas se visualizan cuando se editan las pantallas del HMI, ver figura VI.76.

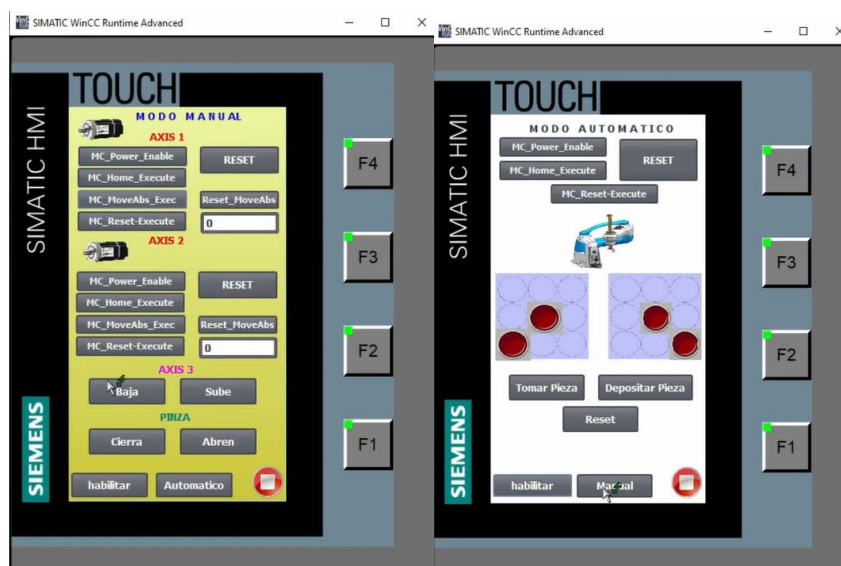


Figura VI.75 Pantalla del modo manual y modo automático.

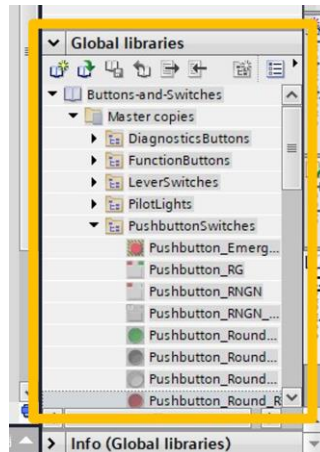


Figura VI.76 Botón usado en los círculos rojos de la imagen: modo automático.

Como se comentó anteriormente se tienen algunas imágenes en las pantallas, el cómo se agregan imágenes se indica en la figura VI.77, al momento de presionar donde apunta la flecha con el número 3, se abrirá el explorador de archivos y se podrá agregar la imagen requerida.

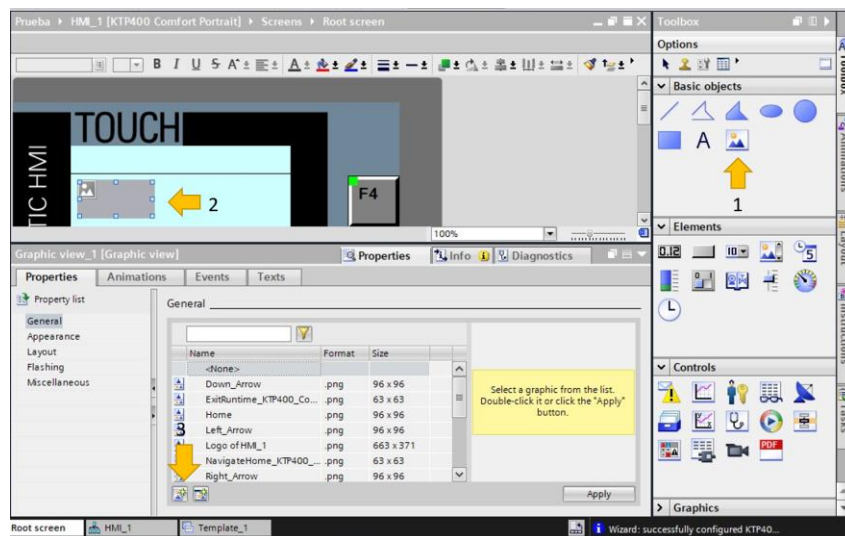


Figura VI.77 Pasos para agregar una imagen.

Al agregar una imagen se debe asegurar que ésta se coloque en la capa 0, ya que será en la que siempre se trabajará y si no se encuentra en ésta, no se podrá manejar con libertad cuáles imágenes se requieren de fondo y cuáles se desean al frente, cuando se superponen imágenes, Ver figura VI.78.

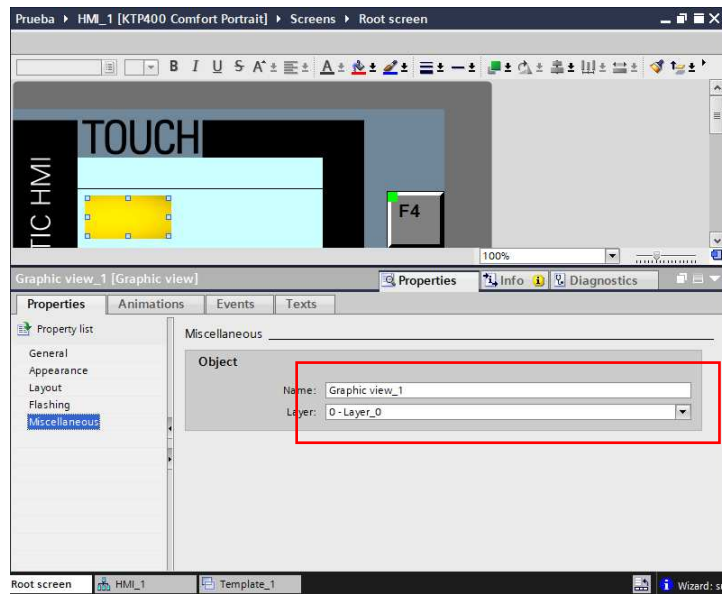


Figura VI.78 Donde revisar la capa en la que se encuentra el elemento.

Si se agrega una imagen y se desea quitarle el fondo, ver la configuración que se indica en la siguiente figura, en *fill pattern* elegir la opción de *Transparent*.

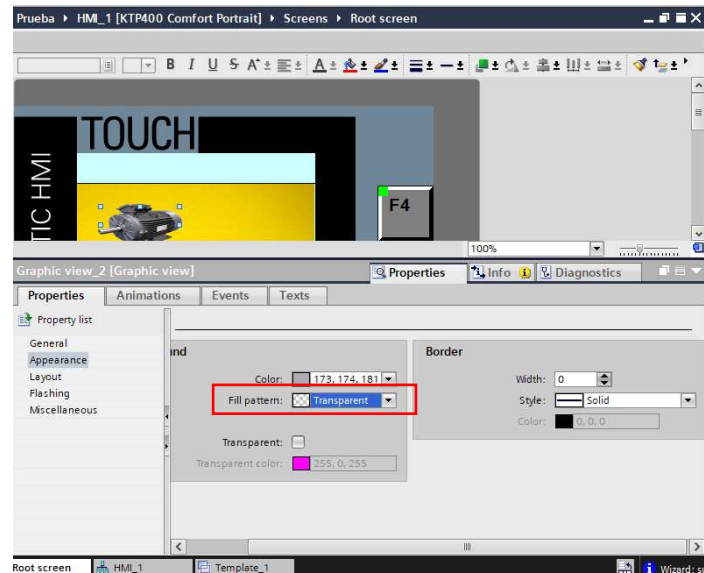


Figura VI.79 Quitar fondo a una imagen.

Para poder controlar qué imágenes se quieren de fondo y cuáles se quieren enviar al frente es necesario realizar lo que se indica en la figura VI.80, elegir imagen, dar clic botón derecho del ratón (mouse) y elegir del menú la opción que se requiera: bring to form,..., send to back, lo indicado también funciona con botones, visualizadores y cualquier componente que se pueda agregar a la HMI.

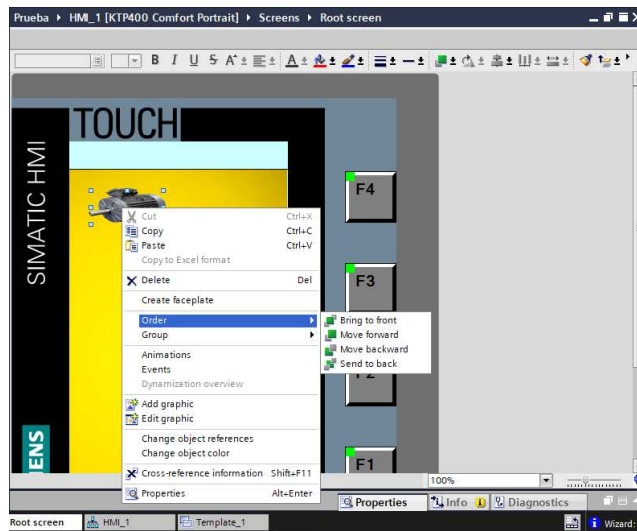


Figura VI.80 Mover un elemento al fondo o al frente.

Una vez que se agregaron todos los botones e imágenes necesarias; cada uno de los botones tiene su nombre correspondiente a su función, por lo que es necesario asignarles un evento al momento de pulsarlos (press).

En ambas pantallas en la parte inferior encontramos el botón habilitar, al cual se le asigna una función en la pestaña de eventos del objeto, la de pulsar se activa con la función de *ActivarBit*, y se elige el bit correspondiente a las marcas (figura VI.81) de *Mod_manual* y/o de *Mod_automático* según sea la pantalla donde se encuentre el botón con el mismo nombre y se desactiva el bit con la función de *DesactivarBit* de la pantalla contraria, es decir, en la pantalla de modo automático si se presiona el botón Habilitar, se pone en 1 el bit de la marca de *Mod_automático* y se desactiva el bit asociado a la marca de *Mod_manual*.

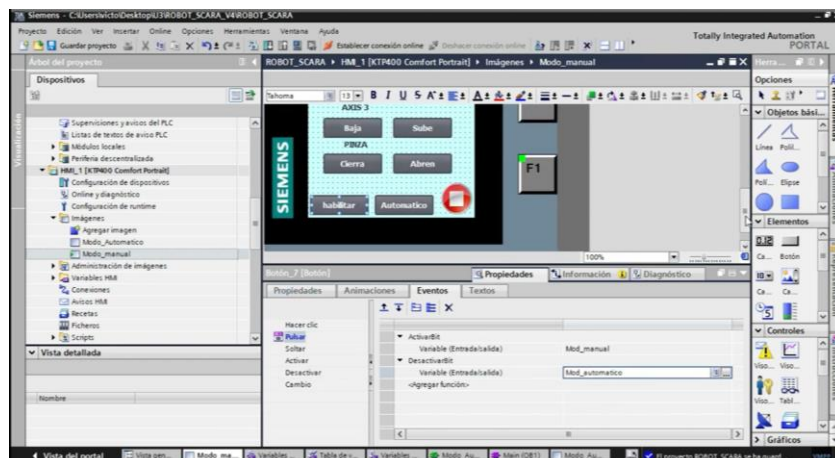


Figura VI.81 Botón habilitar ventana modo manual.

También se tienen los botones de Manual (Pantalla Modo Automático) y Automático (Pantalla Modo Manual), para poder cambiarnos entre éstas. En la figura VI.82 se muestra la configuración de dichos botones, con la acción pulsar se activa por ejemplo la imagen de Modo_Manual.

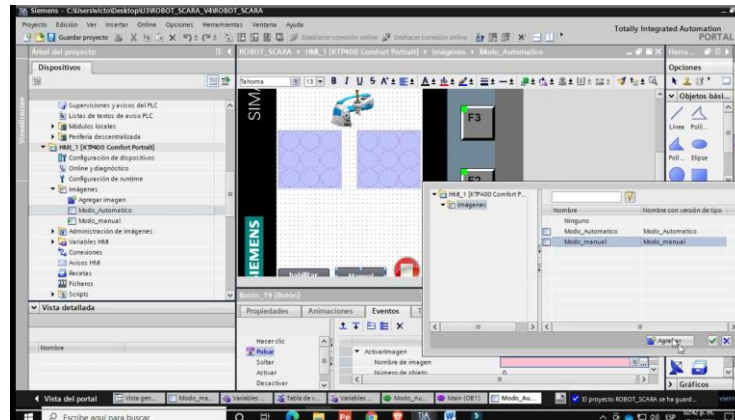


Figura VI.82 Botón ir ventana modo manual.

La configuración para los botones es muy similar, el nombre de cada uno de ellos es el correspondiente a la variable que debe activar, en la figura VI.83 se observan cómo se deben de configurar cada uno de ellos. Es importante tener cuidado con las variables de los ejes, para identificar las del Axis 2, se tiene un 2 al final del nombre.

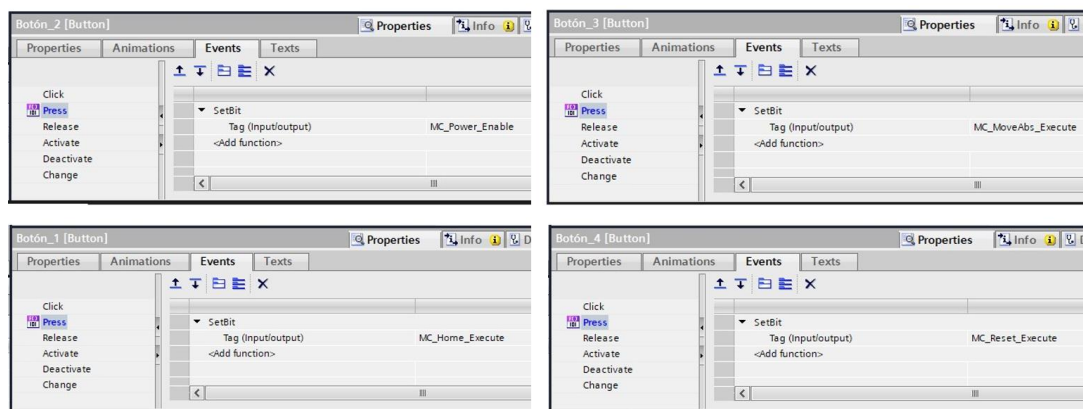


Figura VI.83 Configuración de botones de activación.

El botón de *RESET* de la imagen en la pantalla de modo manual, tiene una configuración especial ya que al momento de presionarlo se deben de desactivar las cuatro variables que se pueden activar con los botones de la izquierda, aparte de la desactivación que se puede ver en la figura de abajo se deben de agregar cuatro funciones más para desactivar los botones mencionados para el control del eje. Ver figura VI.84.

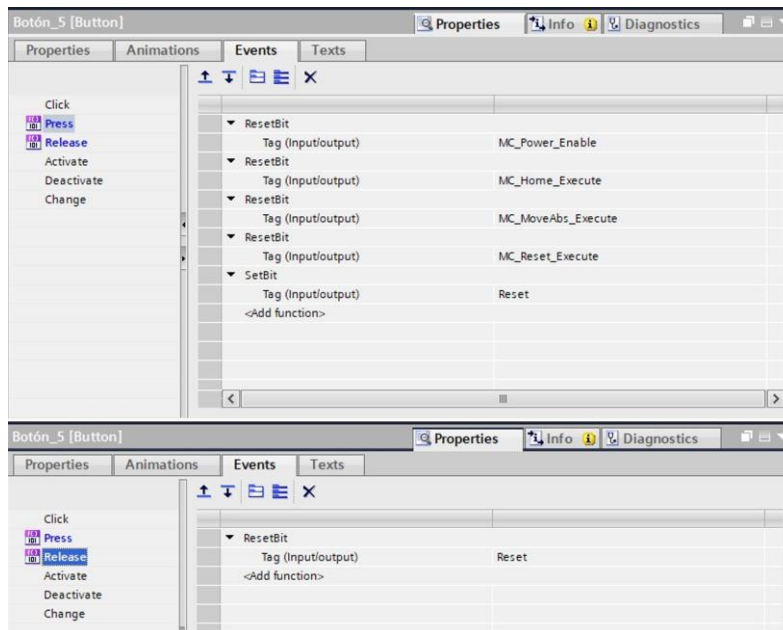


Figura VI.84 Configuración de RESET.

El botón que está debajo de *RESET* también se encarga de desactivar un bit, en la figura VI.85 se muestra la configuración del botón *Reset_Mov_Abs*, limpia la marca *MC_MoveAbs_Execute* y pone en 1 lógico a la marca *Reset_Mov_Abs1* y al soltar el botón se limpia su marca asociada.

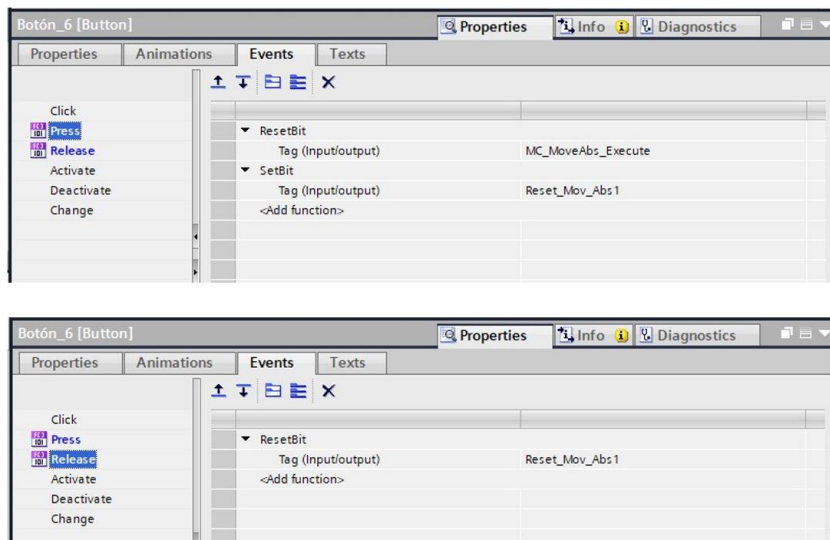


Figura VI.85 Configuración del botón de ResetMoveAbs.

La configuración de la entrada de grados para posicionar el eslabón o eje correspondiente, se puede observar en la figura VI.86 su configuración, esto para los dos ejes (1 y 2).

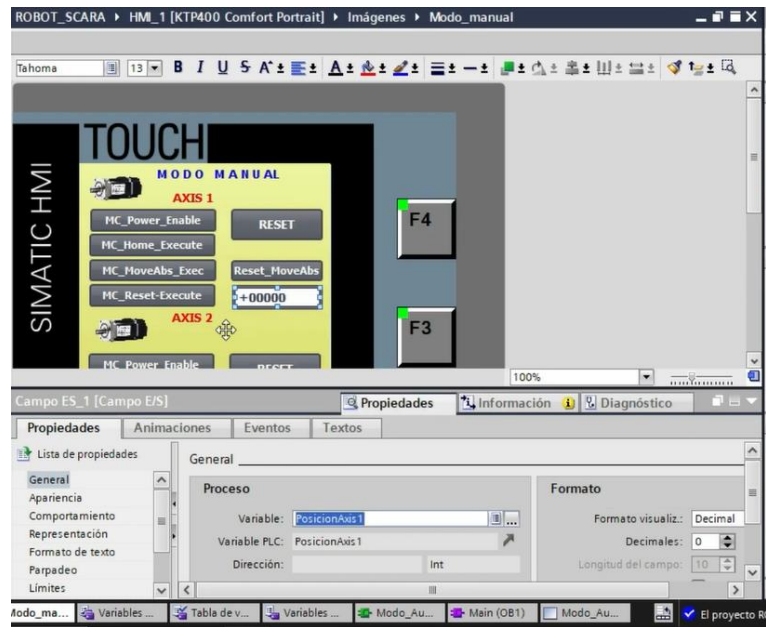


Figura VI.86 Configuración de la entrada de posición (°) por eje.

VI.2.9 Configuración en SIMIT para los ejes 1 y 2 y sus encoders

Para completar la co-simulación o simulación conjunta del *PLC Virtual*, *Sinamic's* y *NX MCD* es necesario hacer que TIA Portal tenga comunicación con SIMIT y contenga los bloques funcionales que emulan las señales de entrada y salida por cada servo drive de los servomotores ejes 1 y 2, para lo cual se crea un nuevo proyecto en SIMIT con nombre SCARA, una vez creado se debe seleccionar la opción *new coupling*, al desplegar su menú se elige la opción *PLCSIM Advanced*, como se observa en la figura VI.82.

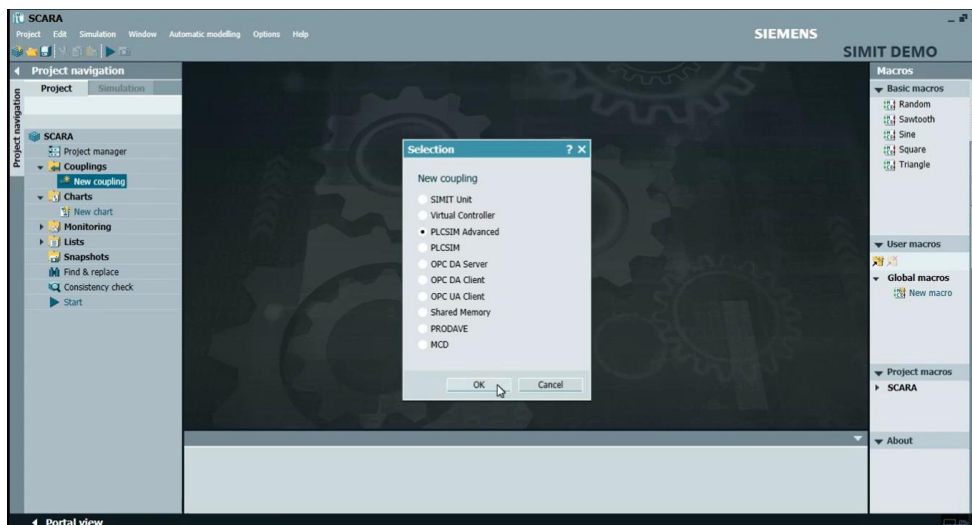


Figura VI.87 Seleccionando el tipo de acople.

Después se selecciona el archivo en TIA Portal del código para el control (Robot_Scara) e importarlo a SIMIT.

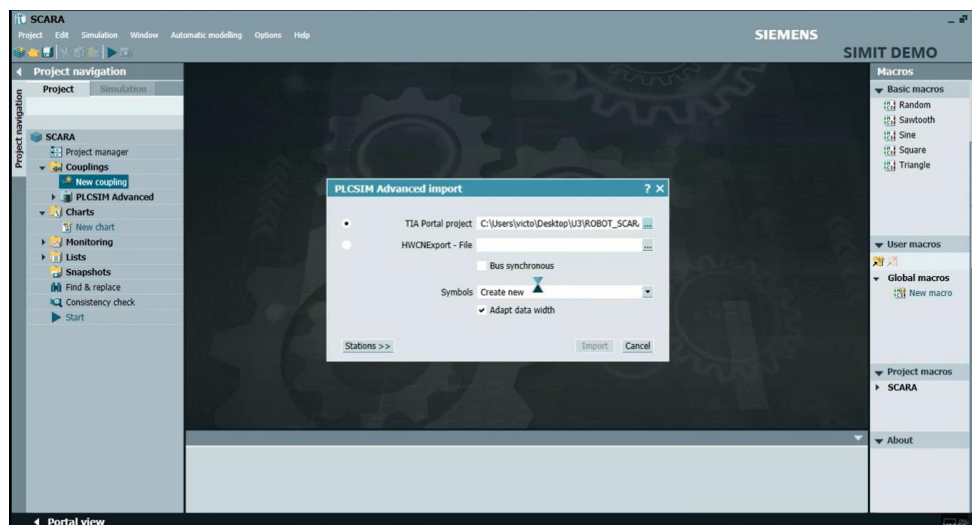


Figura VI.88 Selección de proyecto en TIA Portal.

Seguido de esto, aparecen todas las variables que se declararon en TIA Portal (variables de posicionamiento), para validar el correcto acoplamiento entre los dos softwares, se da click en la flecha que se encuentra en la barra superior, para iniciar simulación, la cual al iniciarse aparecen los títulos de las ventanas en color naranja.

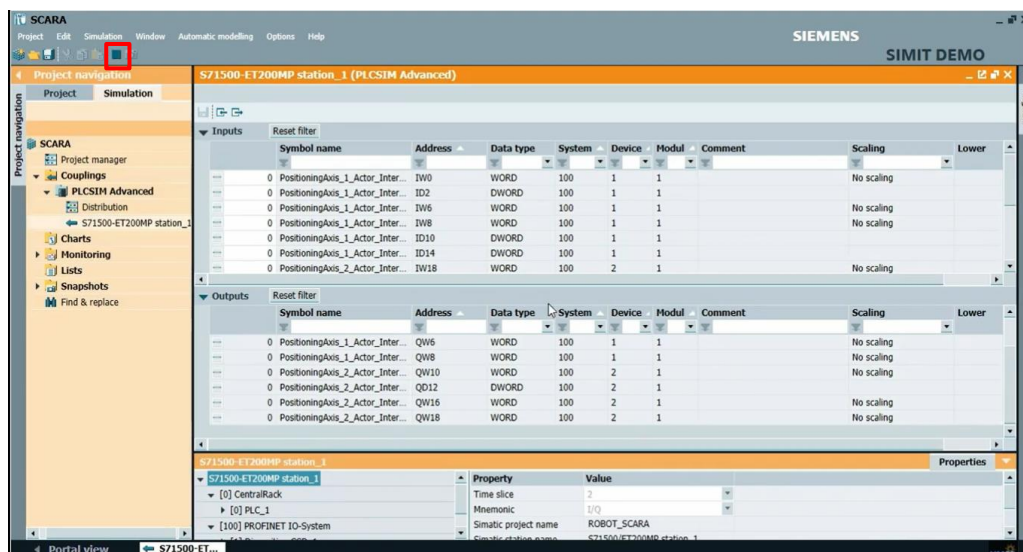


Figura VI.89 Variables importadas de posicionamiento de los ejes y simulación iniciada.

Al ir de nuevo a TIA Portal, se compila y se carga el programa, el cómo cargar el programa se hace de la manera que se muestra en las figuras VI.90 y VI.91. Para hacer la compilación solo es necesario dar clic en el icono que está al lado izquierdo del icono de cargar, éste último se encuentra señalado con un recuadro rojo en la figura siguiente.

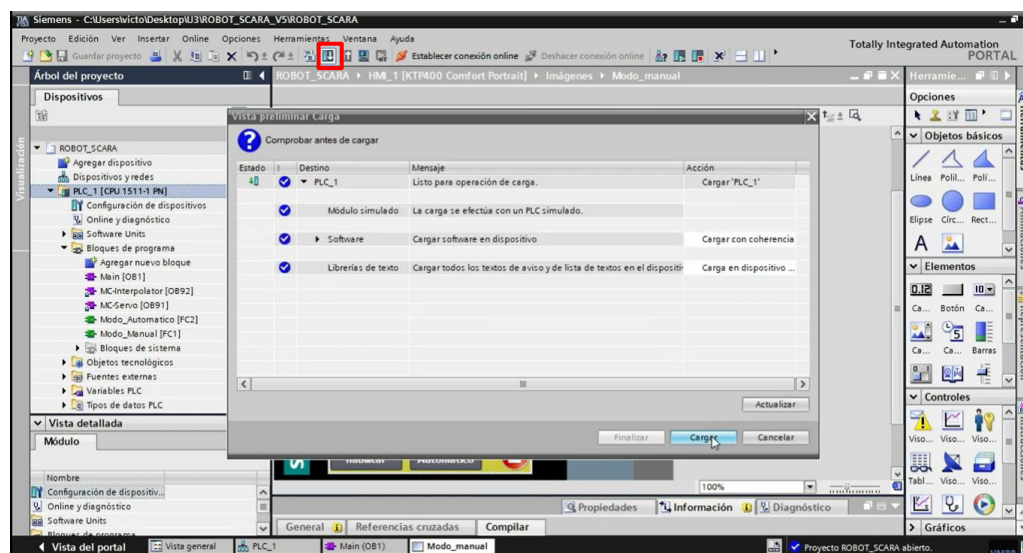


Figura VI.90 Primer paso para cargar.

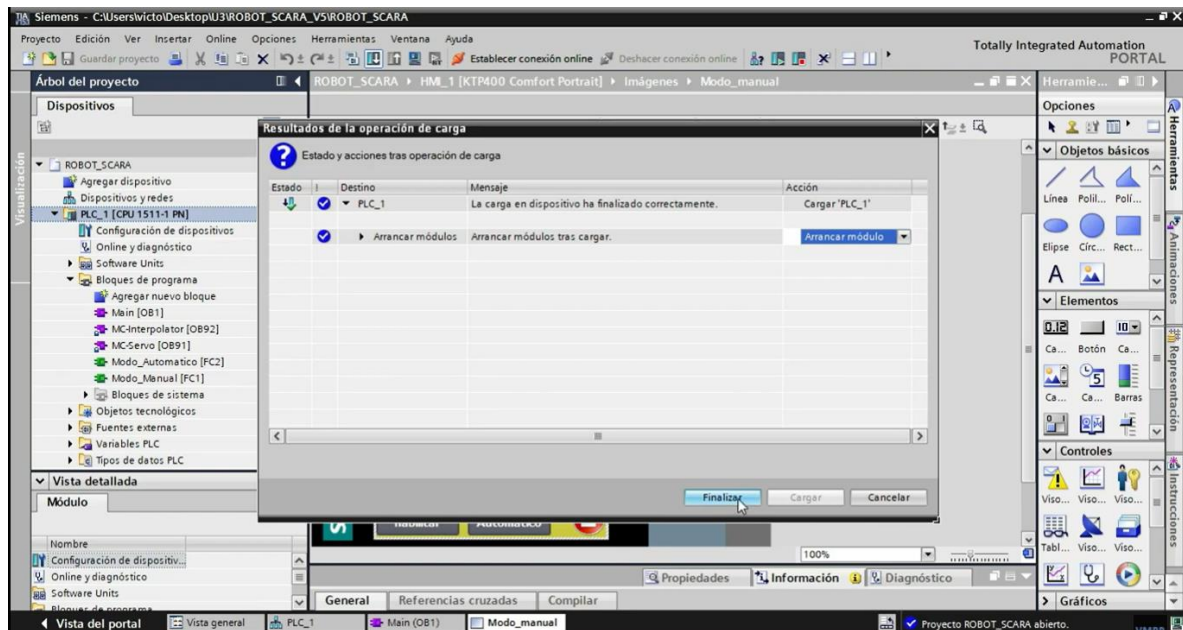


Figura VI.91 Segundo paso para cargar.

Si todo se hizo de manera correcta solo resta dar clic en *establecer conexión online* para poner en línea al TIA Portal y a SIMIT, si no se cometió ningún error hasta este punto, todo el árbol de proyecto de TIA Portal debe de tener círculos de color verde, en caso de que tengamos un error, se nos dirá en que parte lo tenemos en color rojo.

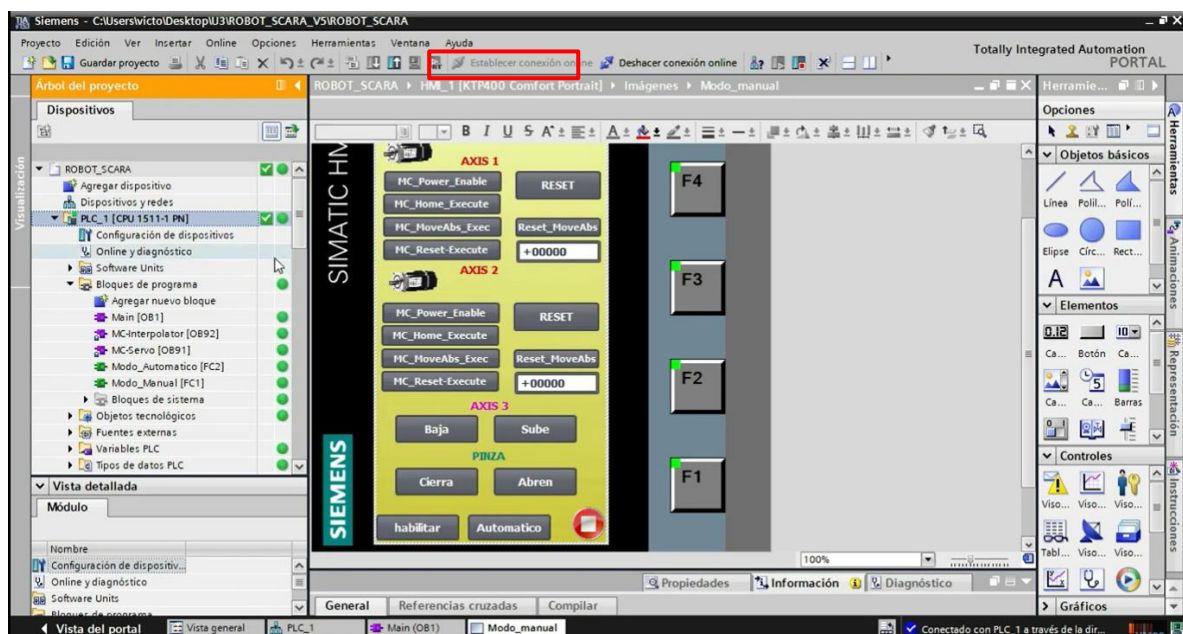


Figura VI.92 TIA Portal con una conexión exitosa.

Posteriormente se valida que los botones que se encargan de habilitar cada modo (automático/manual), lo realicen correctamente, en la siguiente figura se puede observar cómo es que se activa el *modo manual*, para lo cual se debe elegir la pantalla de este modo, luego dar clic en el botón de habilitar. Es importante mencionar que hay que compilar desde la carpeta del HMI del árbol del proyecto en TIA Portal, para luego simular ésta con el botón indicado en la figura VI.93 en recuadro rojo; en la misma figura al habilitar y al minitrorear el bloque lógico *main* [OB1], se activa la marca *Mod_manual* y manda llamar el *FC1* (Modo_Manual), se observa en color verde las condiciones e instrucciones lógicamente verdaderas.

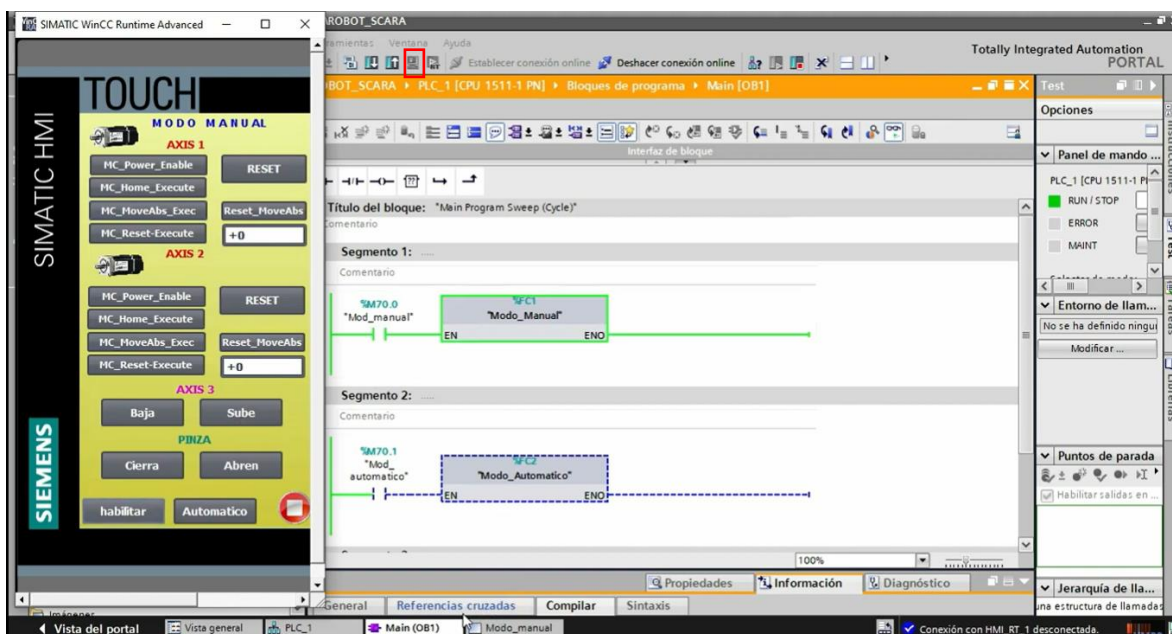


Figura VI.93 TIA Portal simulando.

Al monitorear el bloque lógico *FC1* y al dar clic en el botón *MC_Power_Enable* del eje 1, se habilita su bit correspondiente pero en las salidas del bloque no, al observar la función *MC_Home_Execute*, el bit correspondiente del botón *MC_Power_Enable* al emplearlo en la función de Home en la entrada *execute* se habilita (color verde de la instrucción de contacto cerrado) en la misma subrutina *FC1*, sin embargo las salidas del bloque no se activan tampoco, como se observa en la figura VI.94 en el recuadro de color rojo; por lo tanto es necesario simular el funcionamiento de los servo drives en el SIMIT, que hasta el momento aún no se han configurado. Para que podamos ver el color verde en el bloque debemos de activar el monitoreo, esto se hace con el icono de los anteojos.

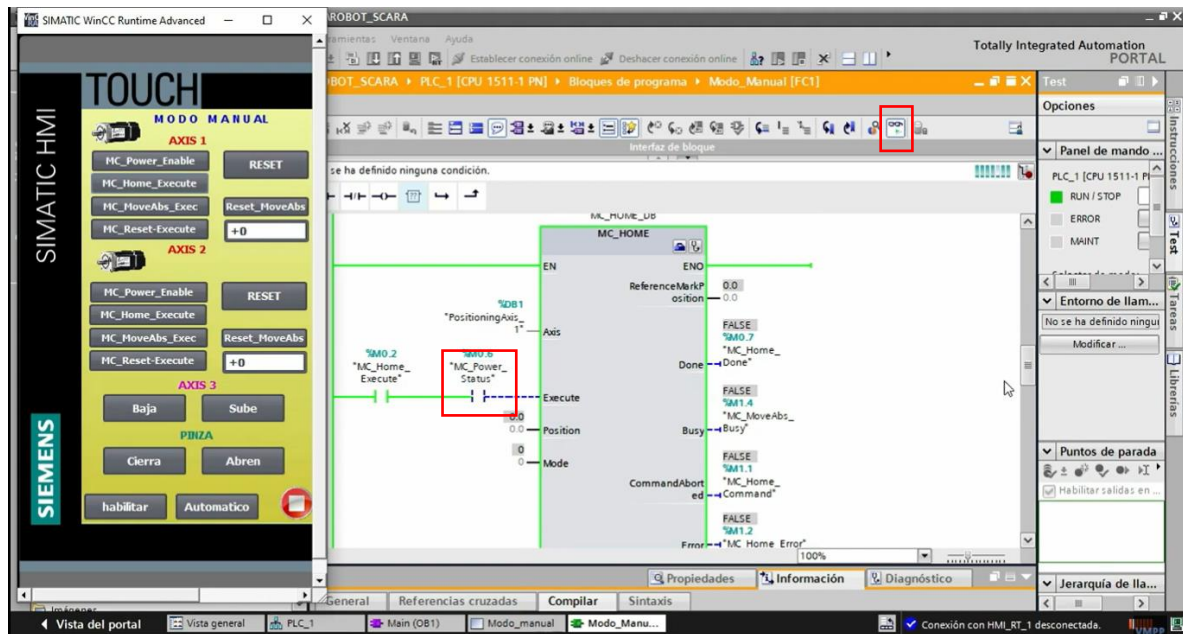


Figura VI.94 Prueba incompleta de TIA Portal.

Así que se procede a crear un nuevo gráfico (chart) donde se agregan todos los elementos que se requieren para hacer el bloque equivalente a cada servodrive; de dónde se toman estos componentes se puede revisar en el apartado donde se explicaron los *conceptos teóricos de configuración*, apartado VI.2.2; específicamente en el punto: *SIMIT Bloque equivalente*. Para las entradas y las salidas se especifica en la figura VI.95 el menú de donde se obtienen: *Connectors*.

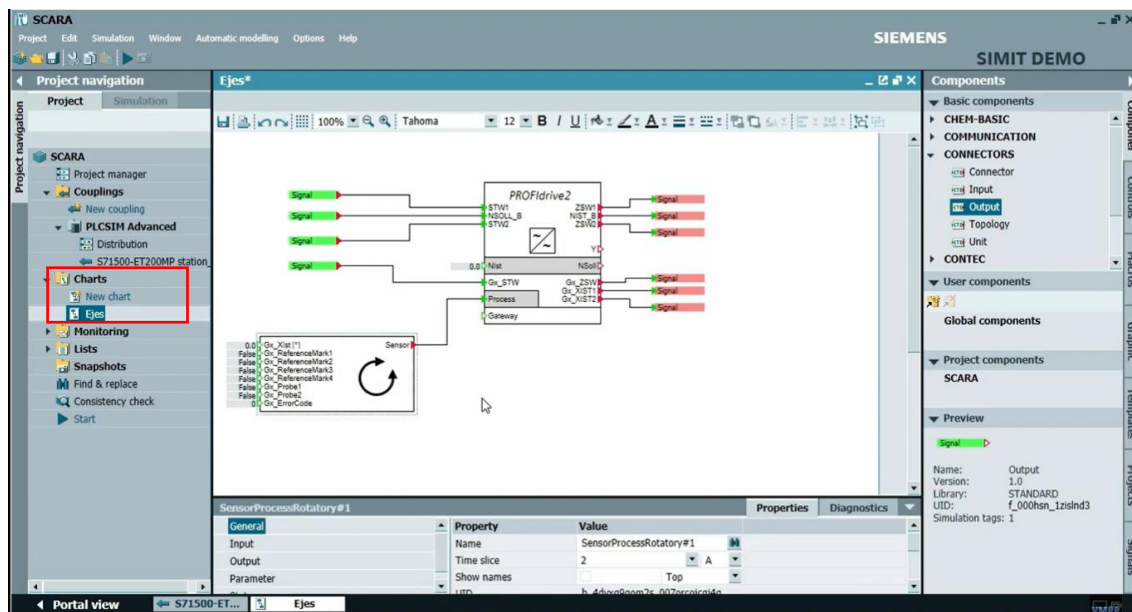


Figura VI.95 Generación del grafico a usar en SIMIT.

Después para colocar las señales es necesario, colocarse en la variable de E y/o S en el chart y aparecerá el menú de *señales*, en el cual se busca la variable de interés y se arrastra a la ventana inferior con título *S71500-ET200MP*, para colocarla en la parte de *Signal*, como se puede observar en el recuadro en color rojo en la VI.96, el primer bloque es para el Axis 1, por lo que todas las variables deben de tener el número 1 y además al final del nombre de la variable viene lo que nos ayuda a saber dónde va conectada cada una, por ejemplo, la primera señal que se tiene para el bloque *PROFdrive 2* es *STW1* entonces la variable que va ahí lleva por nombre *PositioningAxis_1_Actor_Interface_AddressOut.STW1*

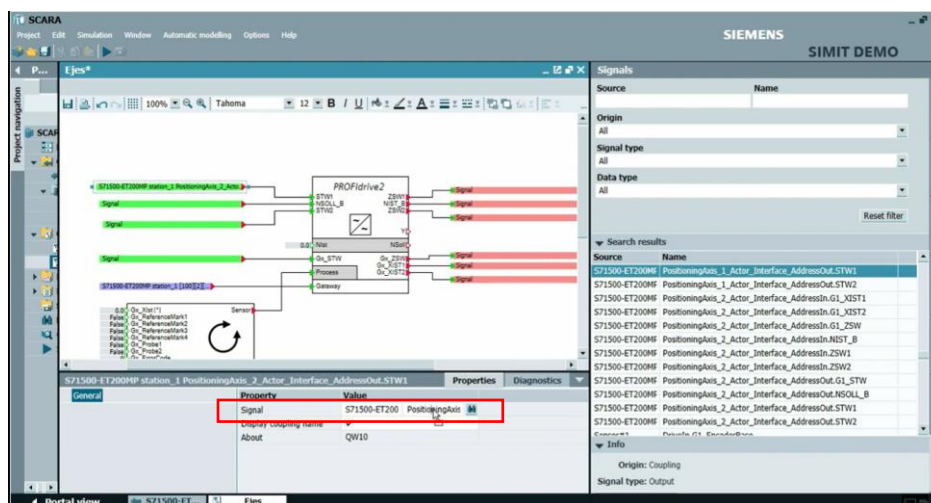


Figura VI.96 Ejemplo de llenado de señales.

Las siguientes se irán llenando de la misma manera, la que sigue en el bloque es *NSOLL_B* entonces la variable que se debe conectar lleva por nombre: *PositioningAxis_1_Actor_Interface_AddressOut.NSOLL_B*

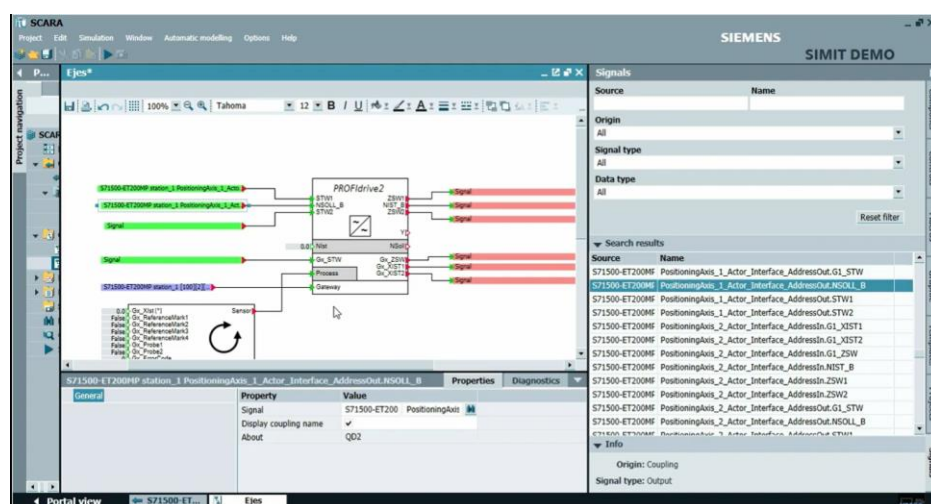


Figura VI.97 Ejemplo de llenado de señales.

Del otro lado ocurriría exactamente lo mismo, la primera variable del lado derecho del bloque es ZSW1 entonces la variable que debe conectarse lleva por nombre: *PositioningAxis_1_Actor_Interface_AddressOut.ZSW1*

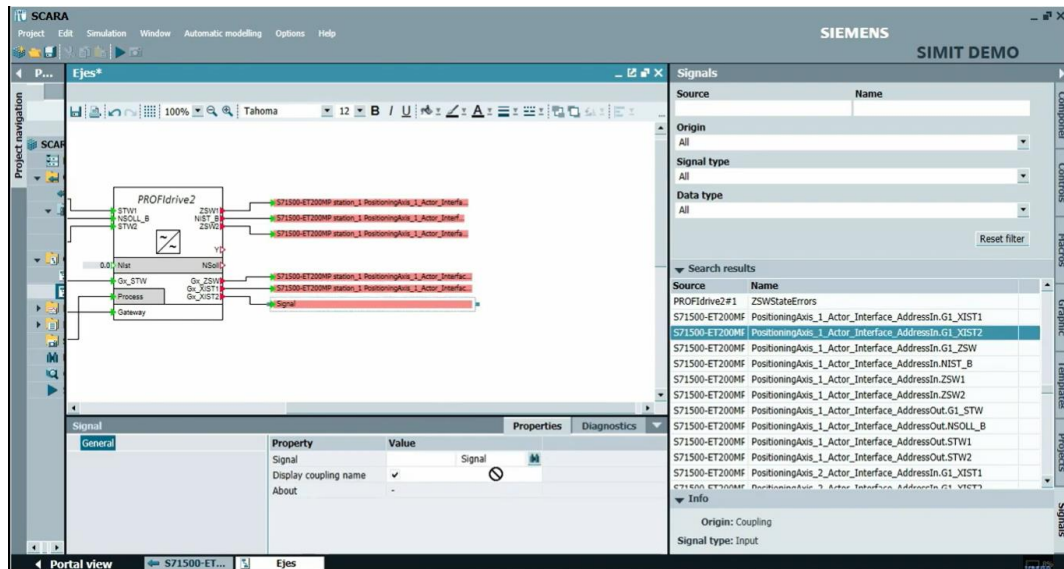


Figura VI.98 Ejemplo de llenado de señales.

Una vez que se termina de llenar todas las señales, se pondrá a correr la simulación de nuevo en *SIMIT* y en *TIA Portal*, se verá como en ese momento todas las señales se activan correctamente y por ende todos los bloques al activarse con los botones correspondientes del HMI, se activan las señales de salida del bloque que no representen algún error.

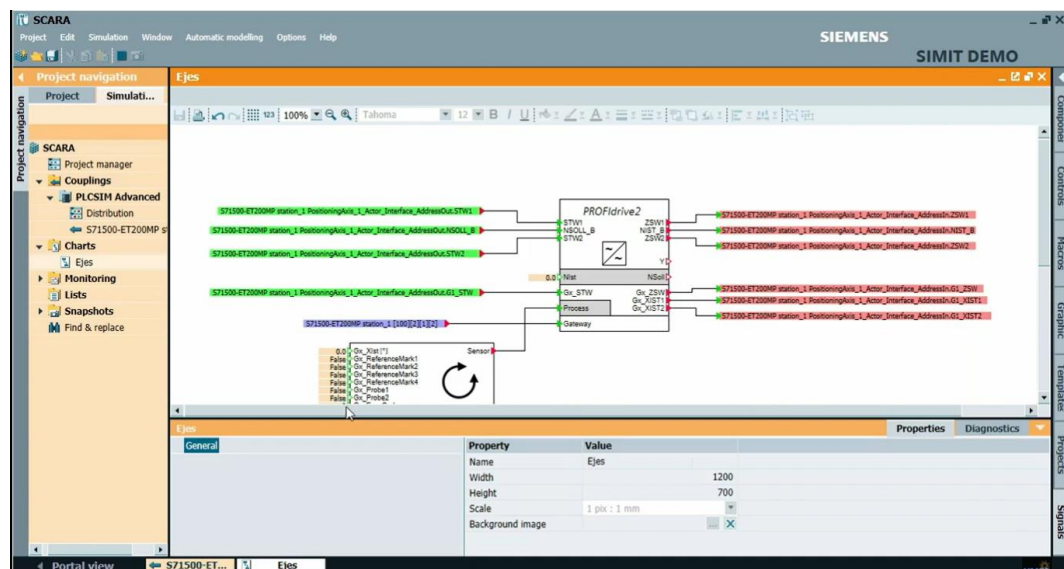


Figura VI.99 Simulación iniciada.

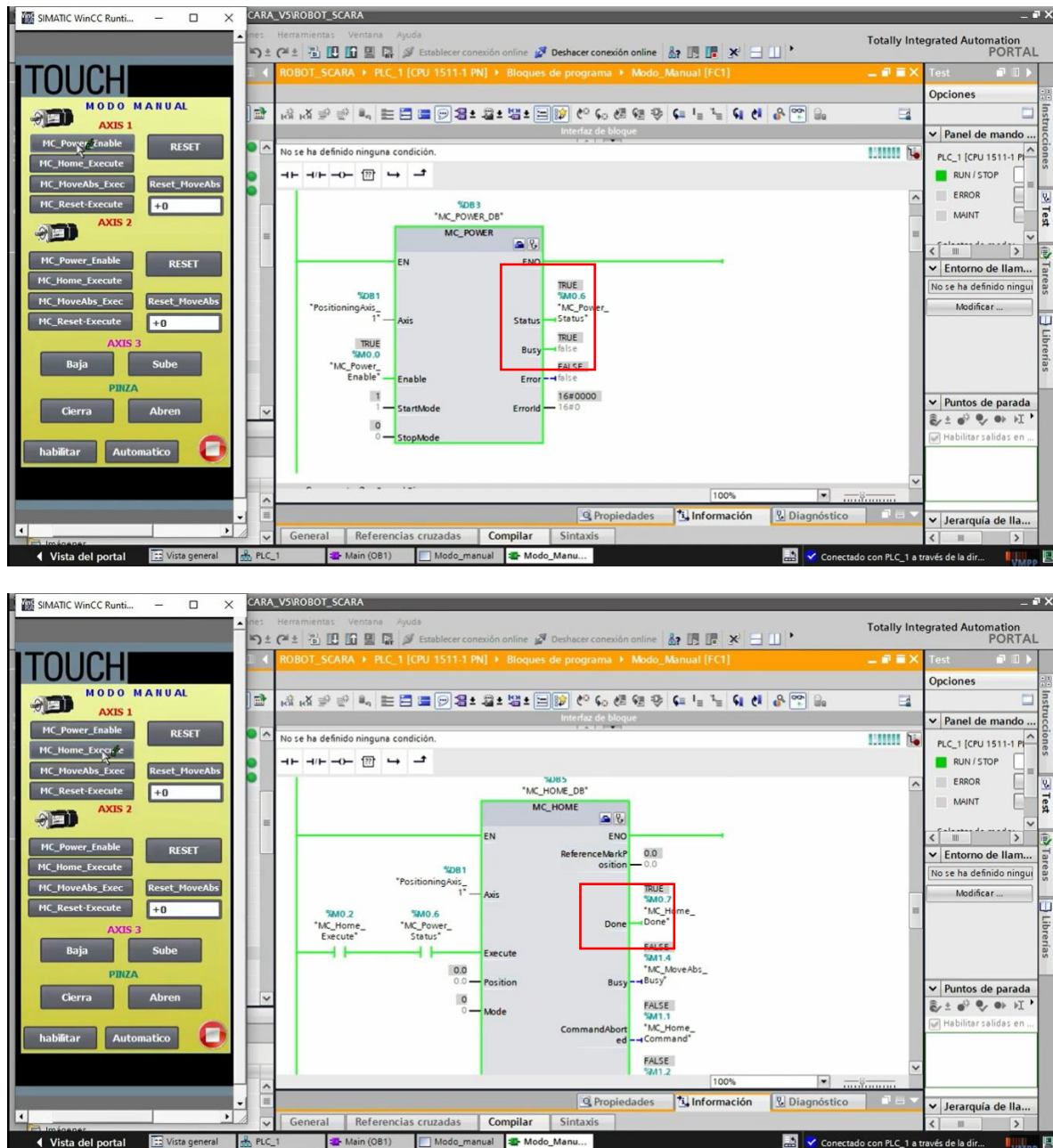


Figura VI.100 Prueba completa en TIA Portal.

Al tener un buen acoplamiento con el PLC Virtual y su programa lógico en escalera con SIMIT, en los bloques al activarlos mediante los botones del HMI, en las salidas de éstos bloques se activan las señales indicadas en los recuadros rojos de la figura VI.100. Una vez que se tuvo una validación completa con los botones y los bloques del *Sinamic*, daremos clic en el apartado de *puesta en marcha* que se encuentra dentro de los objetos tecnológicos del proyecto, se desplegará un nuevo panel de mando del eje.

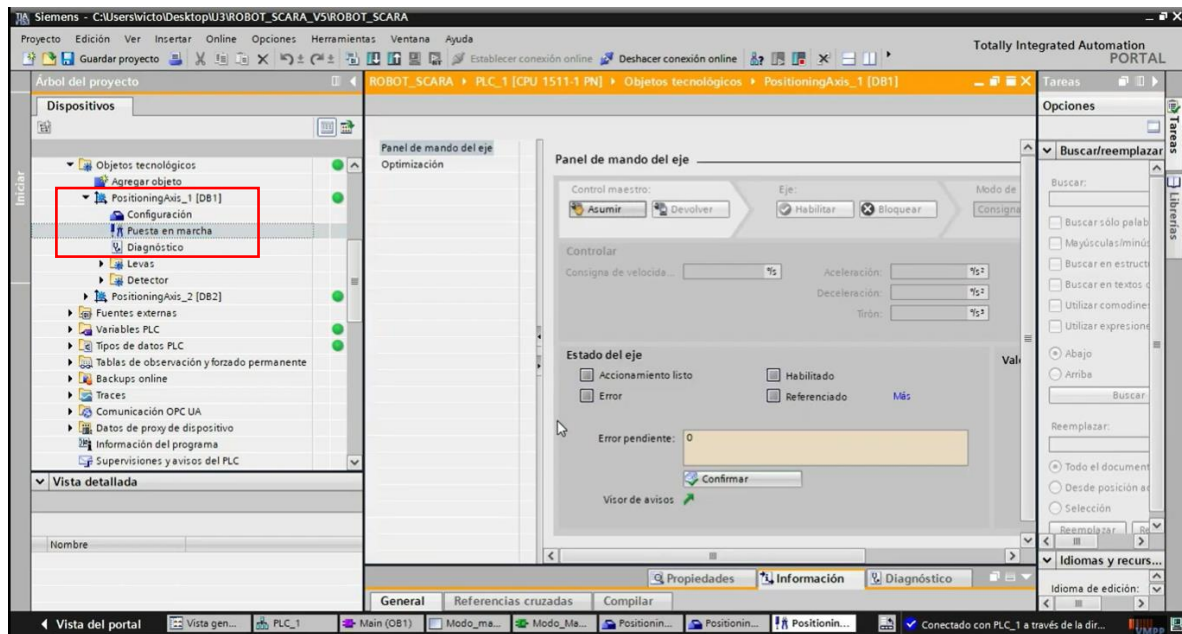


Figura VI.101 Panel de mando del eje.

Al momento de dar clic en *Asumir* para la prueba del eje, se lanza una pequeña advertencia, en la cual solo hay que dar del clic en la opción *sí* y se podrá continuar con la prueba.

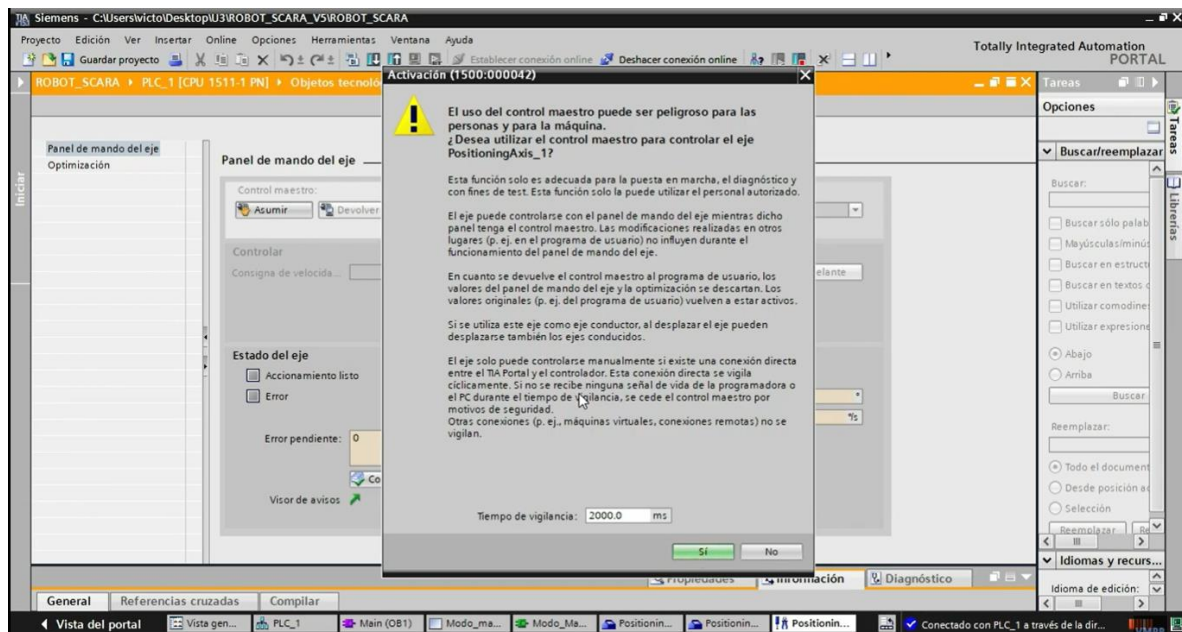


Figura VI.102 Alerta al Asumir el control del eje.

Debemos de dar clic en habilitar y en el modo de operación seleccionar el modo **Jog**, si se elige adelante o atrás, por el momento no ocurrirá ningún movimiento, porque aún no se tiene conexión con Siemens *NX MCD*, que es donde tenemos el gemelo digital del robot SCARA.

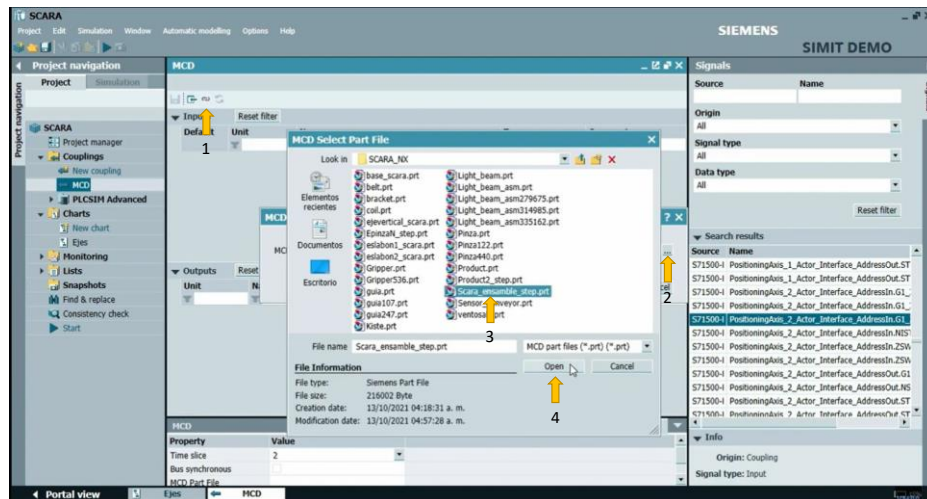


Figura VI.105 Selección de proyecto en Siemens NX.

Si aparece el siguiente error únicamente se da clic en aceptar.

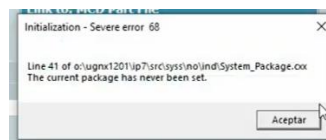


Figura VI.106 Posible error.

Al acoplar aparecen las variables de Siemens NX MCD en SIMIT, es necesario cambiar las unidades de todas ellas, las cuales deben ser en grados, segundos y milímetros.

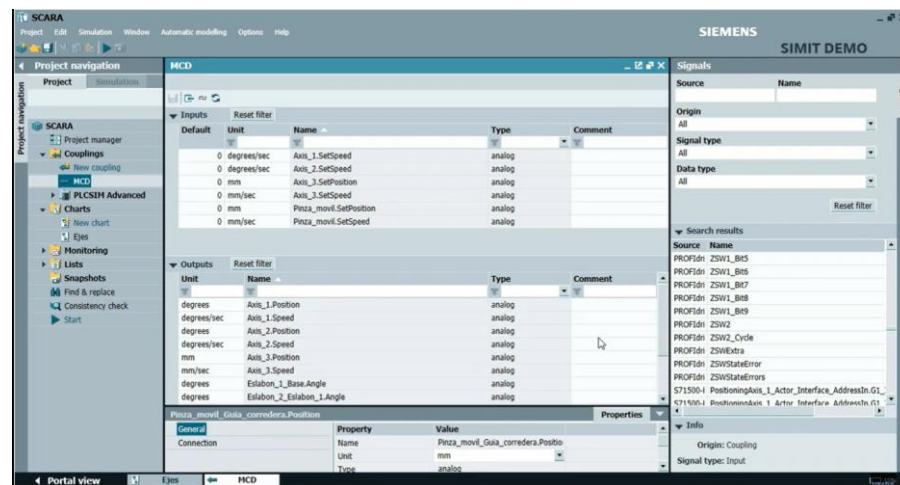


Figura VI.107 Variables de Siemens NX MCD con su unidad correcta.

Una vez que se tiene todas las variables con su unidad correspondiente se conectan en los bloques de *PROFIdrive2*. En el NSOLL se pondrá *Axis_1.SetSpeed*, en el NIST se pondrá *Axis_1.Speed* y en el sensor se pondrá *Axis_1.position*, al igual que en casos anteriores esto se

realiza también para el Axis 2, donde el nombre de las variables es el mismo, solo que comenzarán con Axis_2.

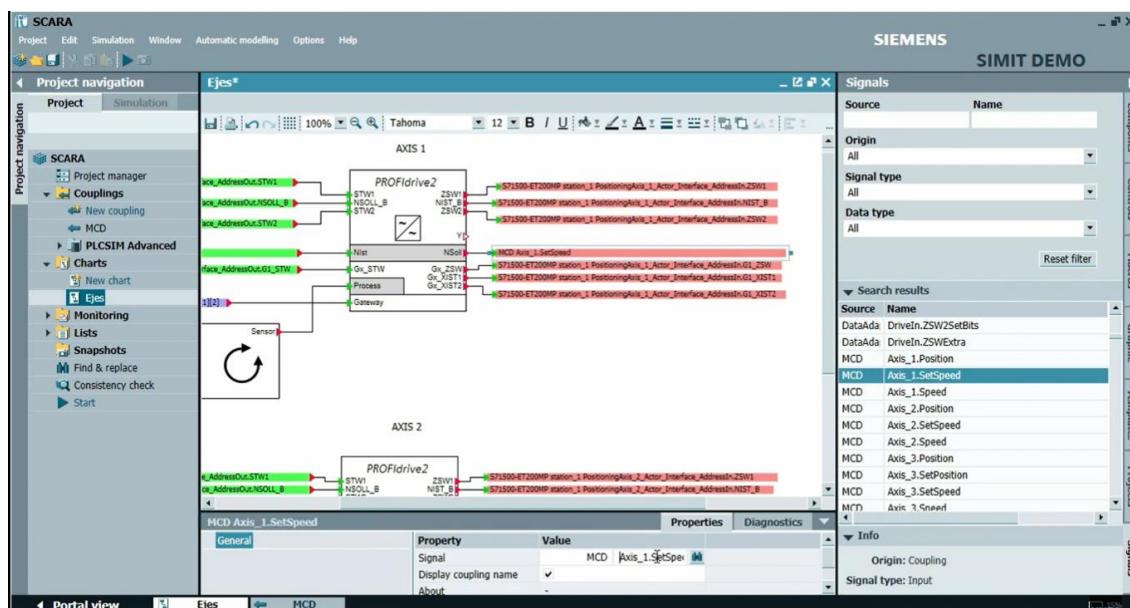


Figura VI.108 Configuración NSOLL.

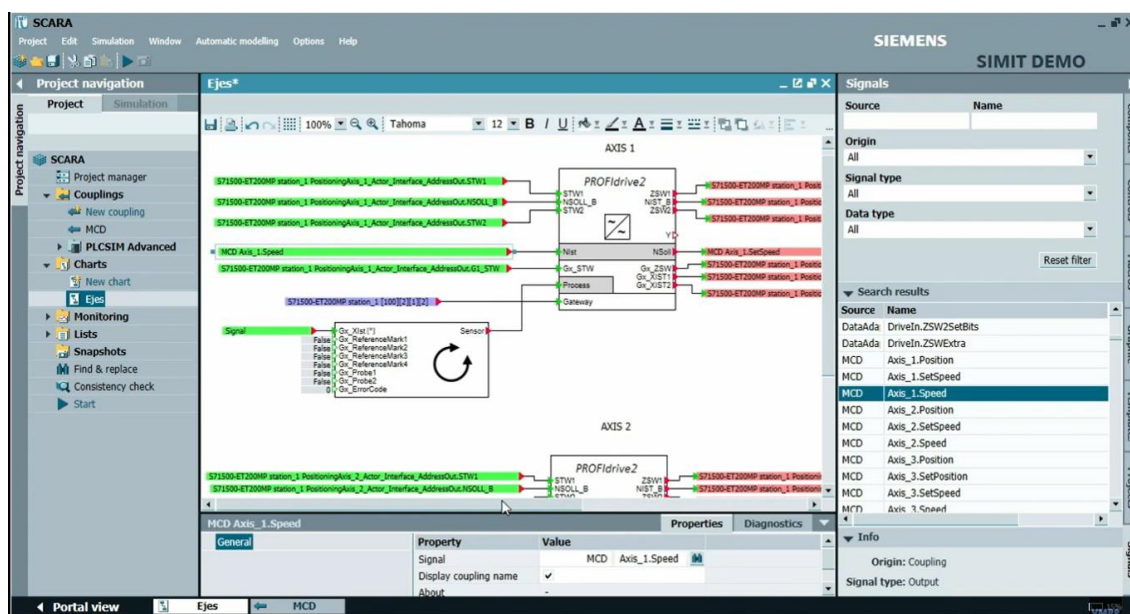


Figura VI.109 Configuración NIST.

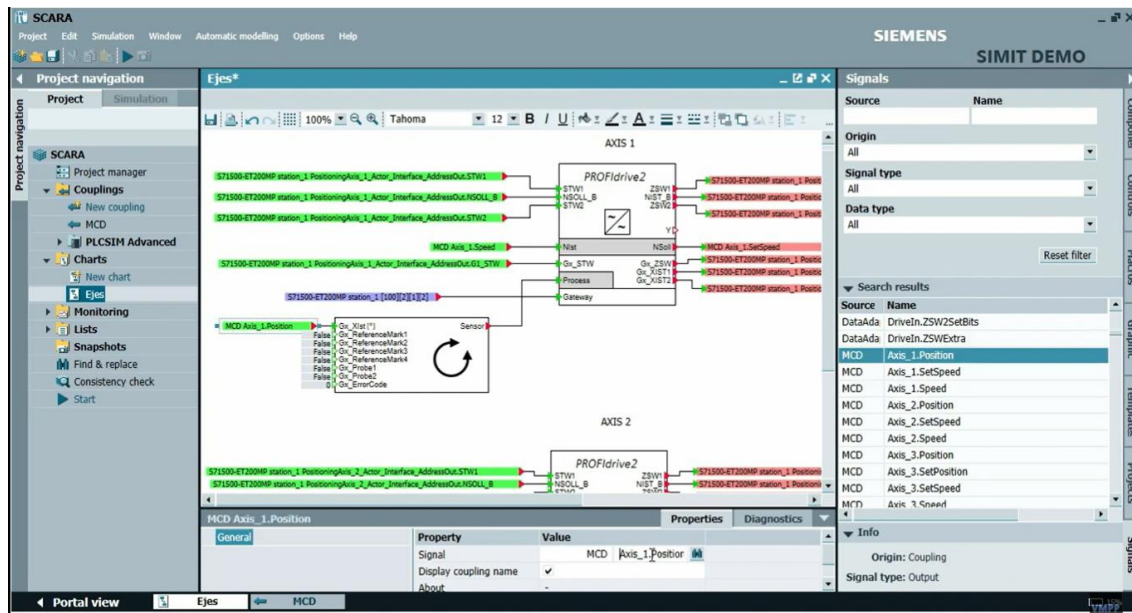


Figura VI.110 Configuración del sensor.

Al simular da un error, se tienen más variables de las que soporta la versión demo, por lo que se eliminan algunas, en la figura VI. 112 se puede ver como quedó el bloque del Axis 1, una vez que se retiraron las señales que no eran necesarias, se procede de igual forma con el bloque del Axis 2. Al simular automáticamente se ejecutará Siemens NX MCD.

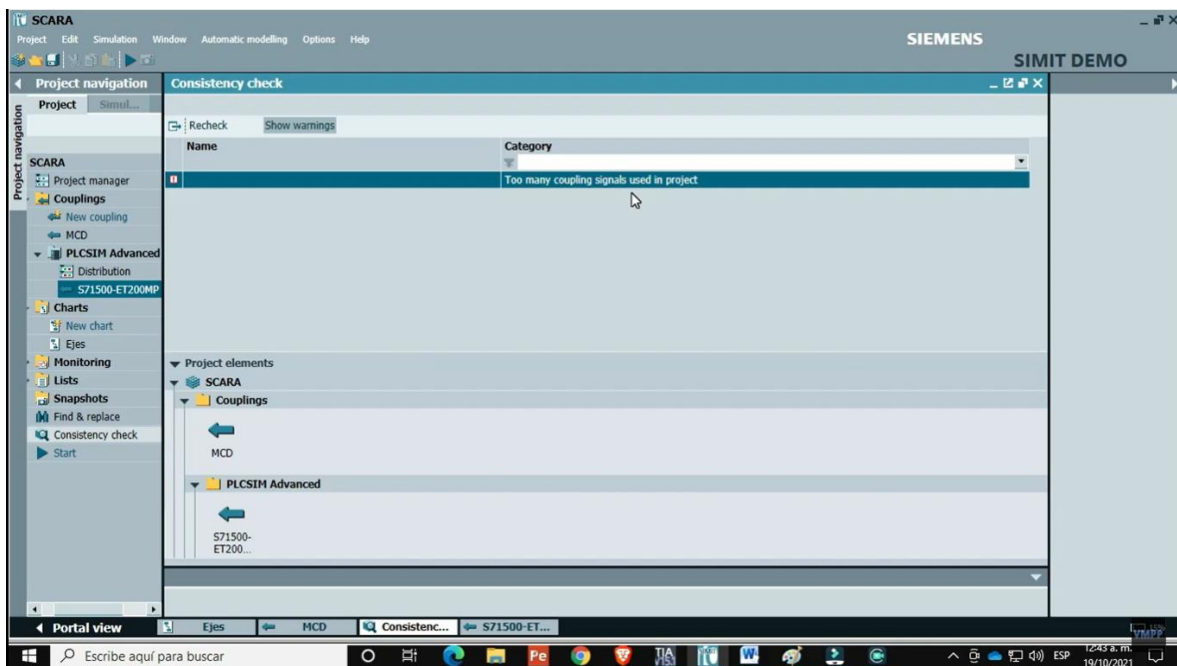


Figura VI.111 Error por cantidad de señales.

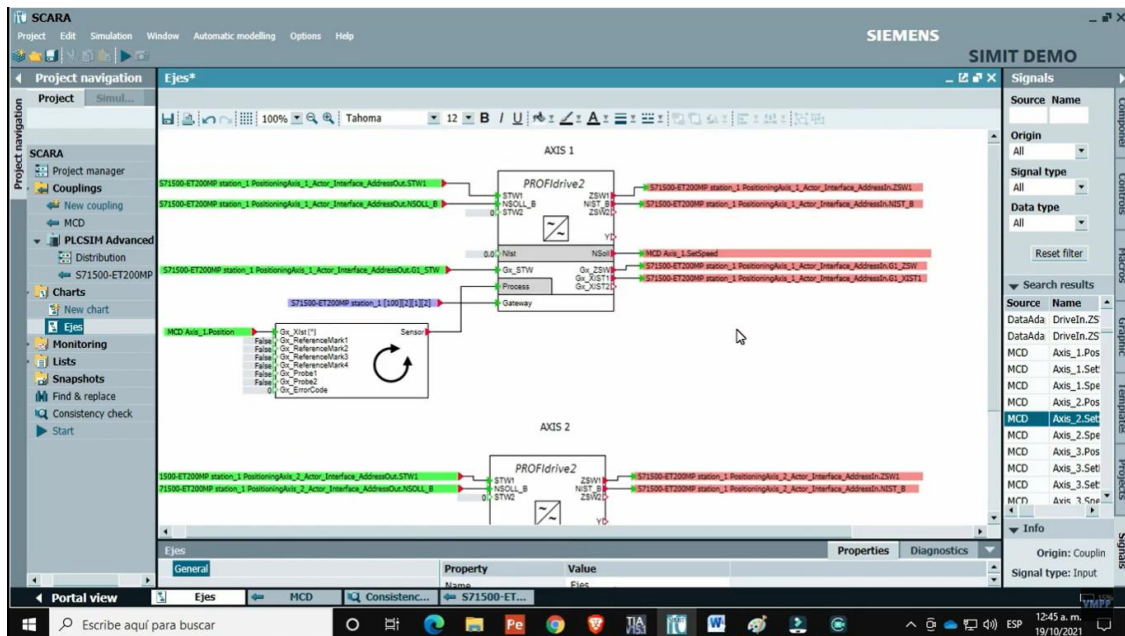


Figura VI.112 Gráfico del Axis 1 sin las señales innecesarias.

VI.2.11 Simulación de Axis 1 y Axis 2 en NX MCD

En el apartado anterior se terminó la configuración de los ejes 1 y 2 tanto en SIMIT como en TIA Portal; en la figura VI.113 se colocaron etiquetas para indicar de dónde provienen las señales y a donde son enviadas.

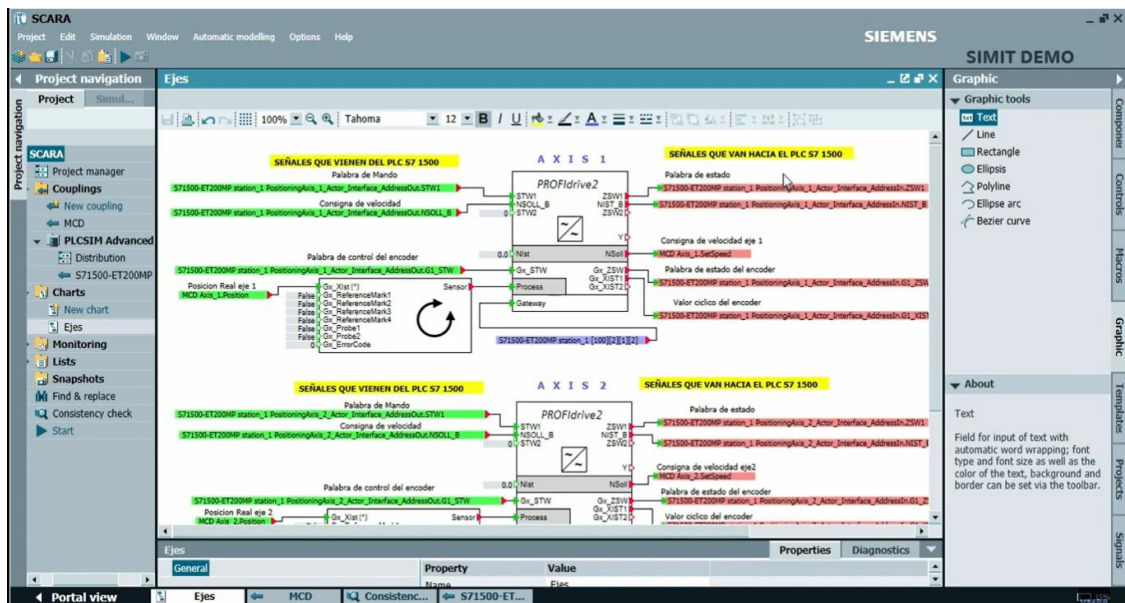


Figura VI.113 Resultado final de SIMIT para Axis 1 y Axis 2.

Pero aún quedan algunos detalles en las pantallas del HMI, como son los de visualización de los botones con animaciones en la apariencia, lo que apoya en saber cuáles han sido accionados por el usuario. Por lo tanto, lo primero que se agrega son las cuatro nuevas variables a la tabla de variables generales, éstas son únicamente para poder poner animaciones en los botones de reset del HMI.

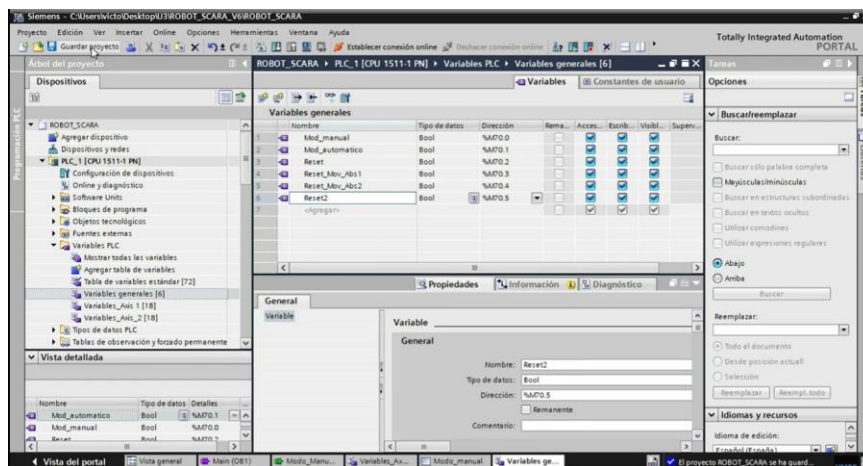


Figura VI.114 Tabla de variables generales.

Se agregan animaciones a los botones que se encargan de encender o accionar los ejes, estos cuando no se presionen serán grises, pero cuando se presionen serán azules y se mantendrán así. Por ejemplo en la figura VI.115 el botón *MC_Power_Execute* es el Botón 2, en la ventana de inspección del TIA Portal en la pestaña de propiedades, elegimos la sub pestaña de animaciones, se elige la opción de dinamizar colores y parpadeo, en la opción de apariencia se eligen rangos para los valores lógicos del botón (0 no pulsado y 1 pulsado), ver figura VI.116.

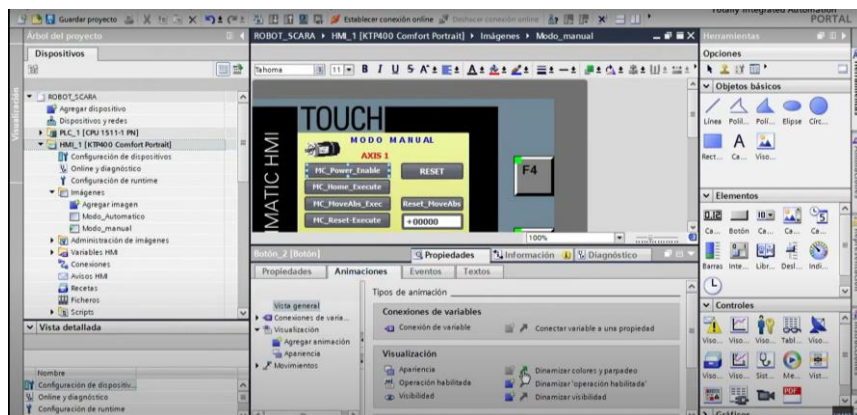


Figura VI.115 Animación botón *MC_Power_Enable*.

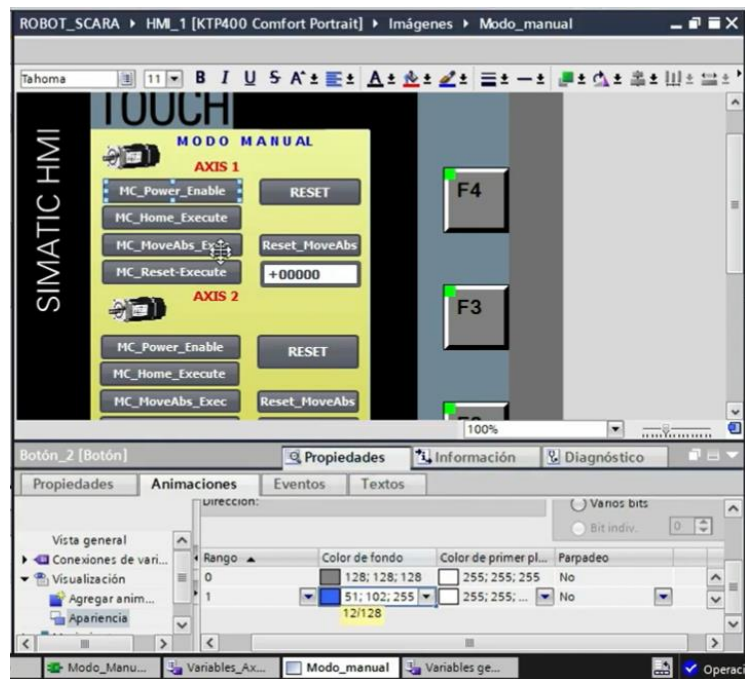


Figura VI.116 Configuración botón 2, animación de apariencia.

Para los tres restantes del eje 1 se realiza la misma animación. En el botón de Reset_MoveAbs, se desea que en un principio sea de color gris (0 lógico, no presionado) pero cuando se presione (1 lógico) cambie a color naranja.

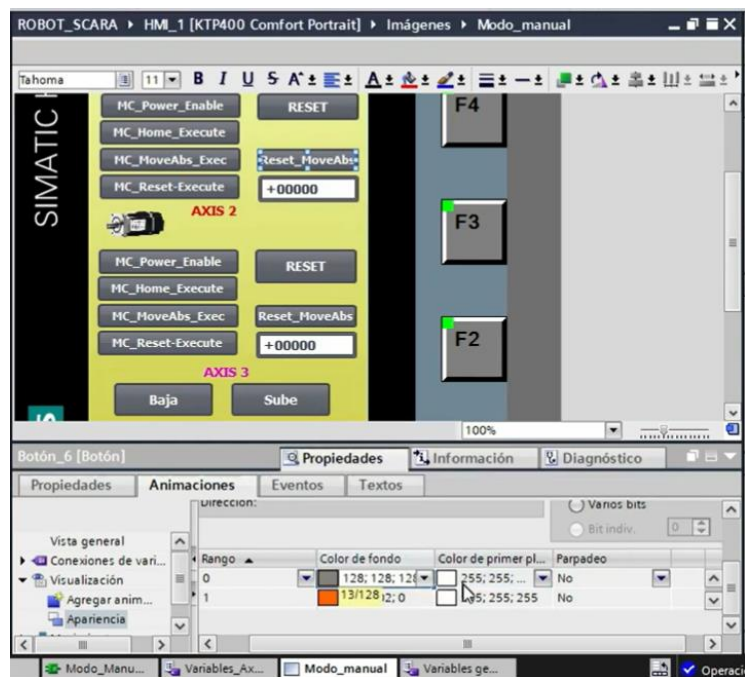


Figura VI.117 Configuración de botón de Reset_MoveAbs.

En el botón de *RESET* se requiere una configuración extra, agregar dos eventos, que cuando se presione active el bit de la variable Reset (%M70.2), la cual se observa en la figura VI.114, y cuando se deje de presionar desactive el bit de esta misma variable.

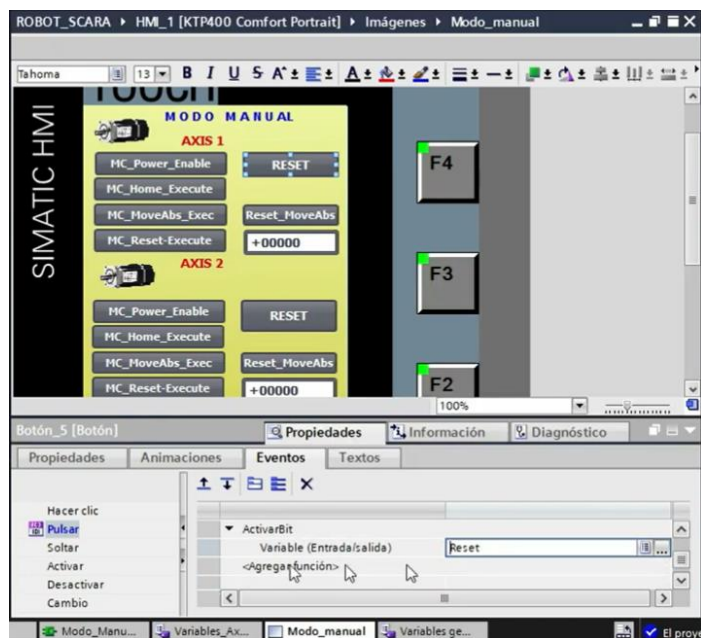


Figura VI.118 Activación de variable Reset.

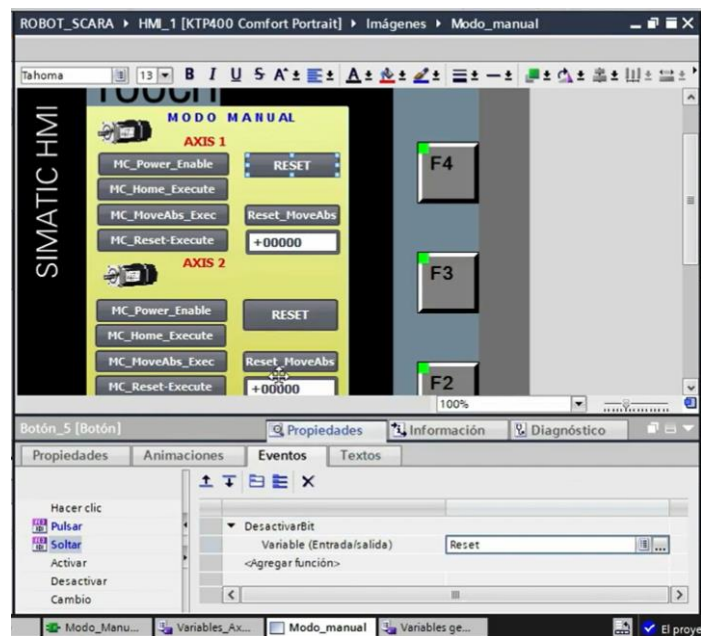


Figura VI.119 Desactivación de variable Reset.

Se realiza el mismo procedimiento con el botón que tiene *Reset_MoveAbs*, al igual que con el botón de *RESET*, se agregan dos eventos, para éstos se emplean otra de las variables

creadas (ver figura VI.114), en este caso, para el Axis 1 sería la variable *Reset_Mov_Abs_1*. Por ende para el Axis 2 se usarán las variables *Reset2* y *Reset_Mov_Abs_2*.

Se agrega también un led para indicar el estado del Axis, éste será controlado por la variable *MC_Power_Status* para el Axis 1 y para el Axis 2 sera *MC_Power_Status2*. Mismo procedimientos para las animaciones.

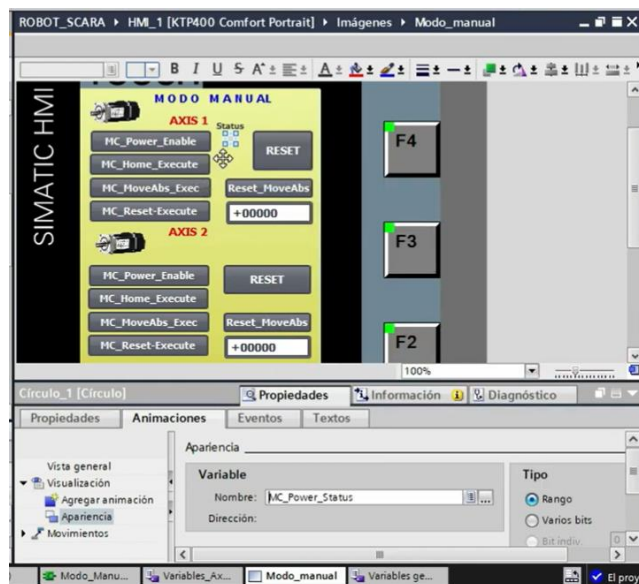


Figura VI.120 Selección de variable para el led.

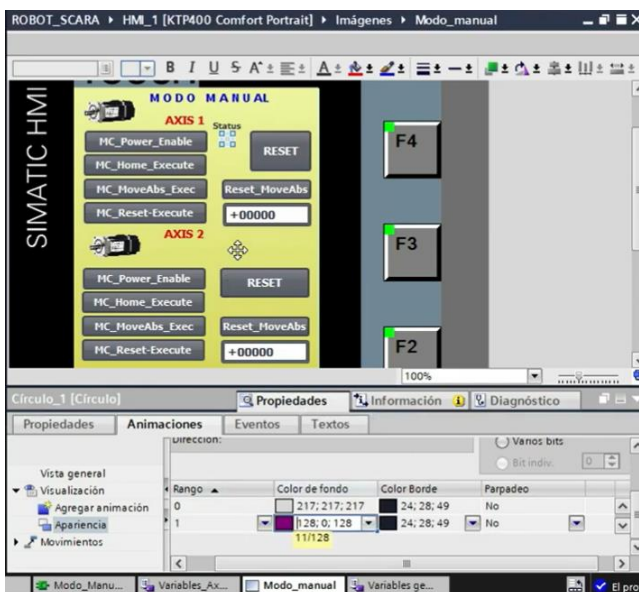


Figura VI.121 Configuración de colores para el led.

Al botón de habilitar también se le agregará una animación, éste estará parpadeando el tiempo que esté encendido y cuando esté apagado estará de color lila. Mismo procedimiento para animaciones.

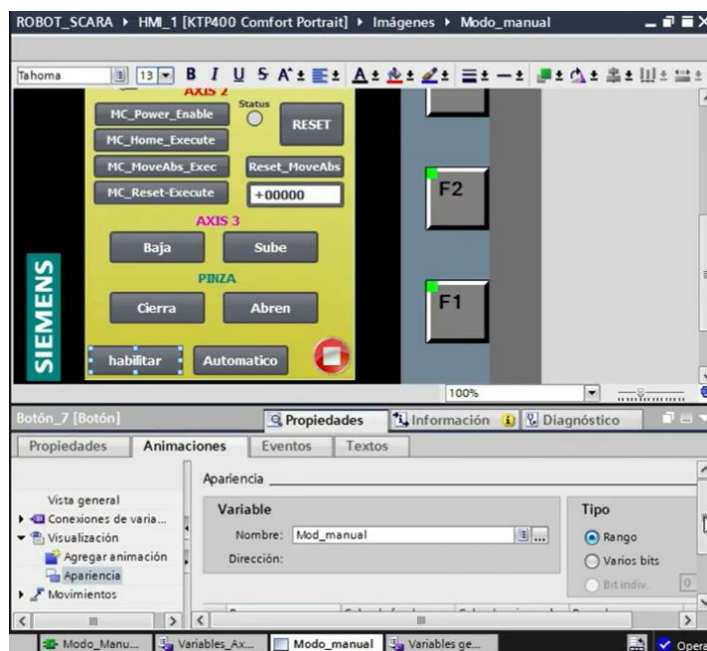


Figura VI.122 Selección de variable para el botón de habilitar.

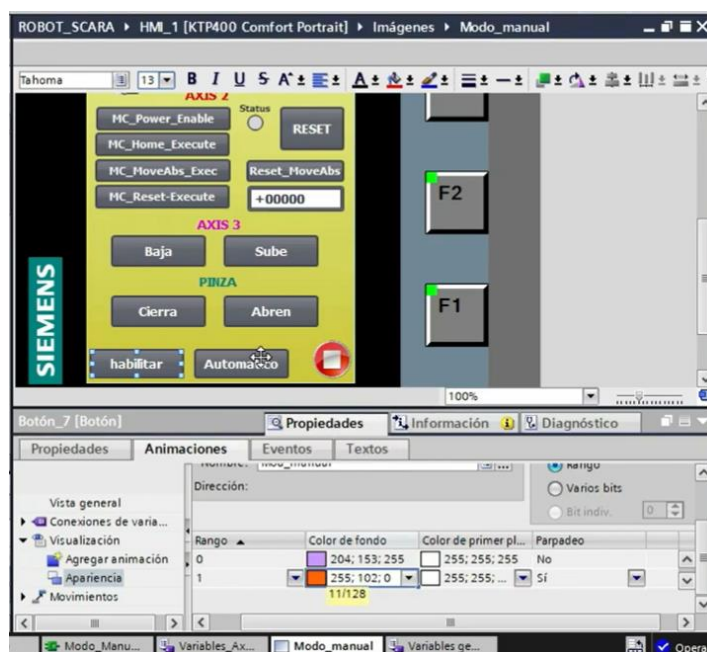


Figura VI.123 Configuración de colores para el botón de habilitar.

Se inicia la simulación para validar como es que funcionan los tres softwares al mismo tiempo, lo que se le conoce como simulación conjunta o co-simulación. Del *TIA Portal* se emplea la HMI diseñada y el código en lenguaje de escalera, *SIMIT* sirve para que pueda existir un intercambio de información de los servodrivs y en Siemens *NX MCD* se observa el resultado del control del ensamble en 3D con el movimiento del robot Scara, dependiendo de los comandos que reciba del HMI.

Primeramente se valida al momento de presionar el botón en la HMI que accione todo correctamente en TIA Portal, es decir las activaciones de los bloques de los objetos tecnológicos, recordar que para cada modo primero hay que dar clic en el botón de habilitar y después presionar los botones de encendido, de referencia (home), etcétera.

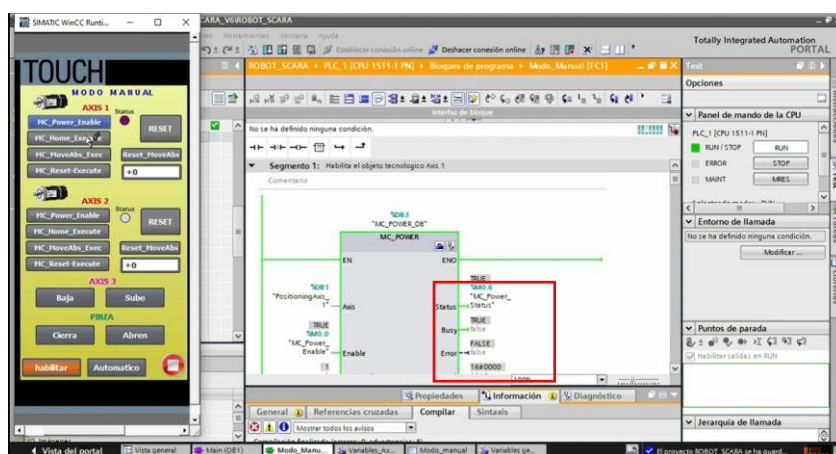


Figura VI.124 TIA Portal y SIMIT funcionando, bits de salida del bloque activos.

Se prueba dando un valor a través de la entrada para grados, se presiona el botón primeramente *MC_Home_Execute* una vez encendido el eje con el botón *MC_Power_Enable*, para después presionar *MC_MoveAbs_Exec* para mover el brazo lo indicado en grados del valor introducido mediante el objeto de entrada correspondiente del HMI.

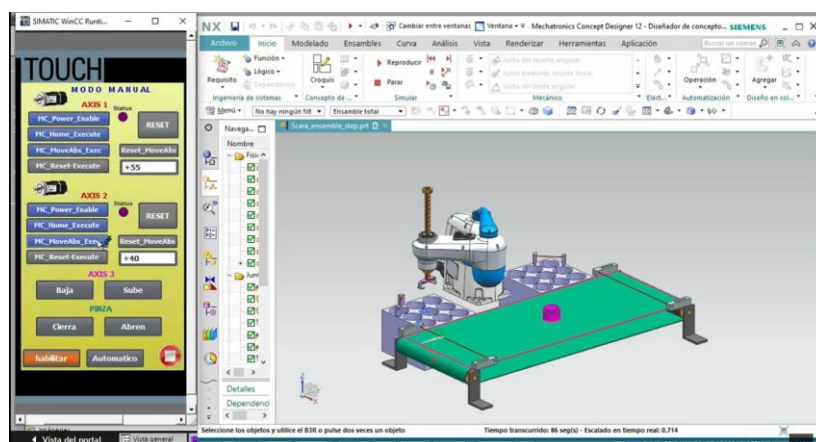


Figura VI.125 Siemens NX MCD recibiendo comandos del HMI, ambos ejes.

Al comprobar que los ejes se mueven en el *NX MCD* de manera correcta, se prueba los botones de RESET de ambos ejes y validan que se desactiven todos los botones. De igual forma para dejar las entradas en ceros en el programa almacenado en el PLC virtual se resetean con los botones de cada eje: *Reset_MoveAbs*.

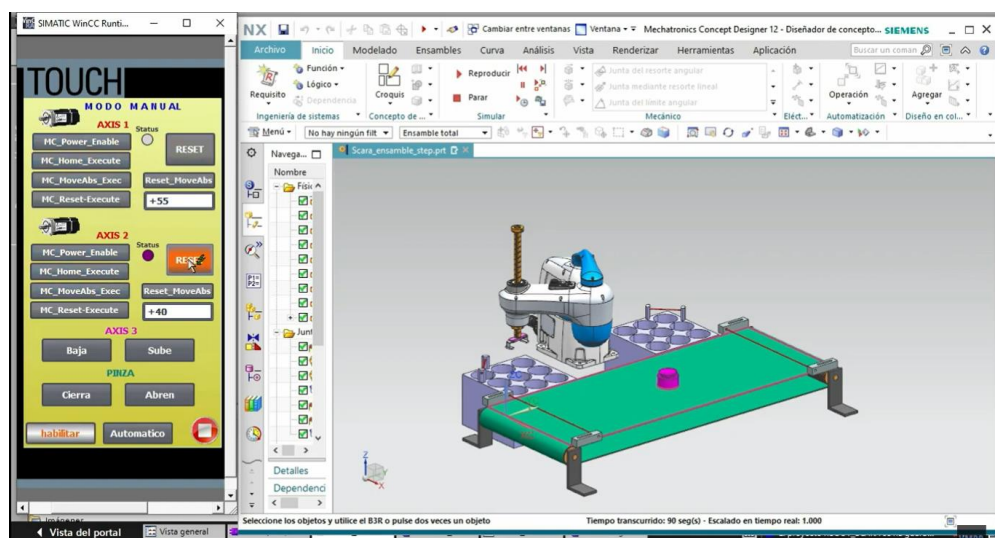


Figura VI.126 Se presiona RESET.

VI.2.12 Configuración de Axis 3 y Pinza

Para la configuración del eje 3 y la pinza es necesario agregar un módulo de salidas analógicas, esto se hace desde la vista de dispositivo del PLC en el TIA Portal, se selecciona del catálogo de hardware y se procede a revisar las direcciones de salida; dirección inicial es la 20 y la final es la 23. Ver figura VI.127.

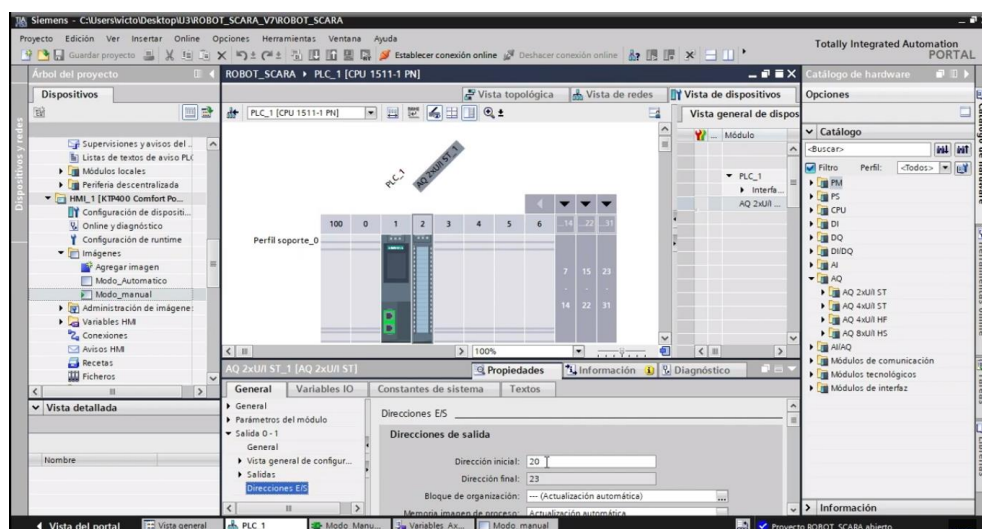


Figura VI.127 Integración de módulo de salidas analógicas.

Una vez que se agregó el módulo, se crea una nueva tabla de variables para el Axis 3, ésta tendrá seis variables, las direcciones de éstas deben de coincidir con las que tiene el módulo de salidas analógicas

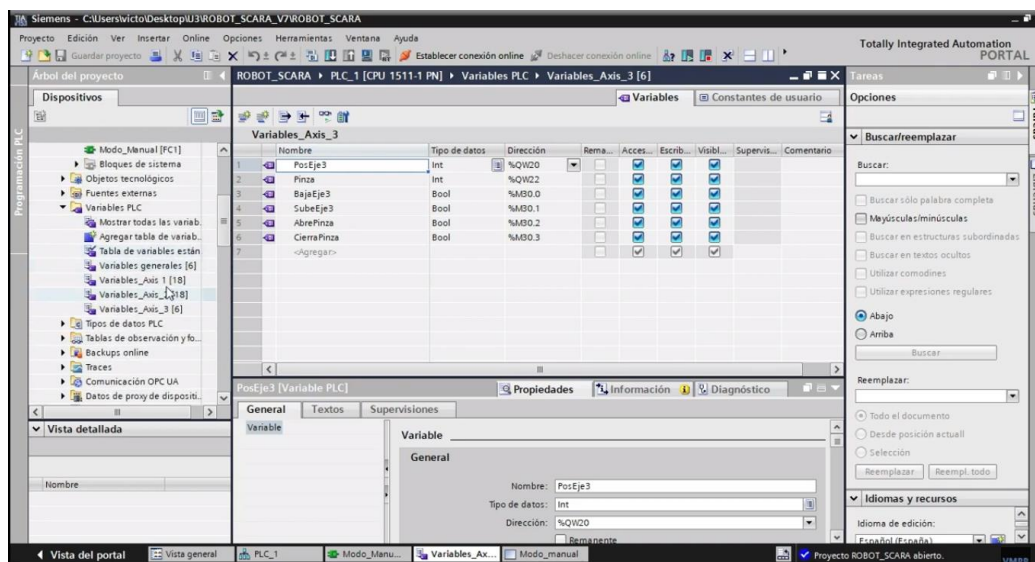


Figura VI.128 Tabla de variables Axis 3.

En el bloque de *modo manual* se agregan nuevos segmentos, el que hace que el Axis 3 vaya hacia abajo y otro para que vaya hacia arriba, para ello se empleó la instrucción **move**, el número entero se probó en el *NX MCD* con un control de posición, por lo que el valor de 109 es correcto.

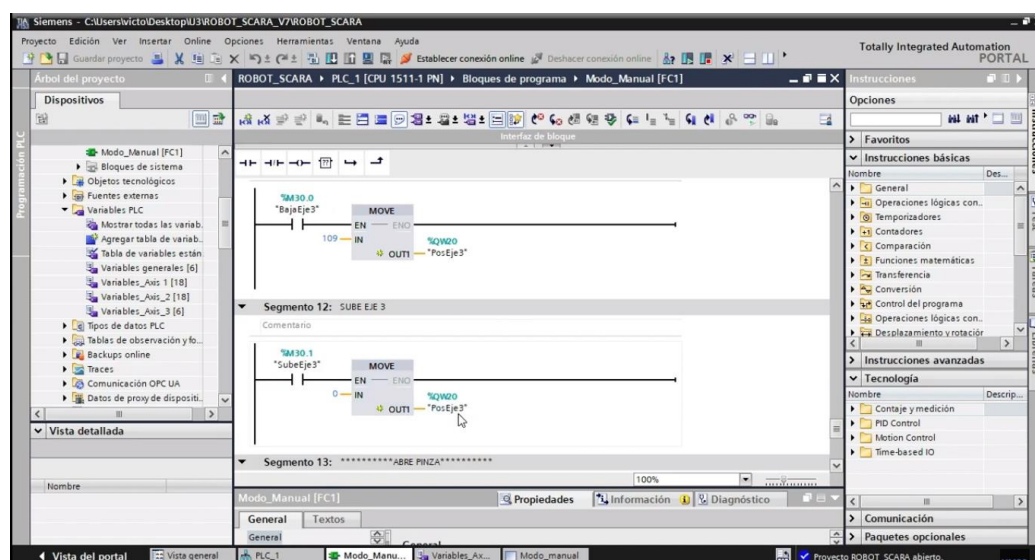


Figura VI.129 Primeros segmentos del Axis 3.

Y por último con la misma instrucción, se emplean otros dos segmentos para controlar la pinza, es decir, para la apertura y cierre de la misma, el número 15 se obtuvo de la misma manera como se describió en el párrafo anterior.

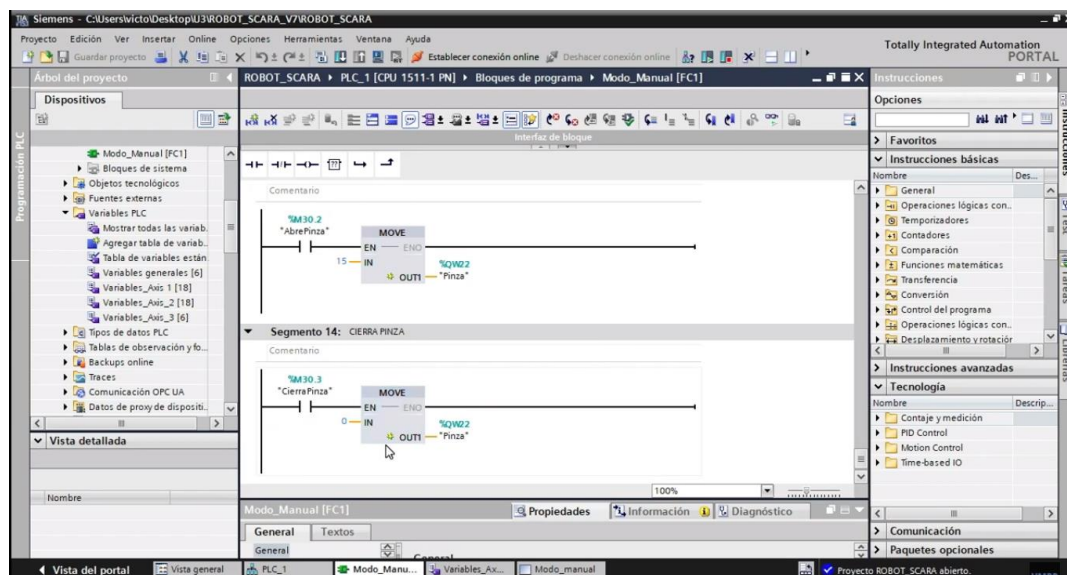


Figura VI.130 Segundos segmentos del Axis 3.

Si bien ya están los botones en el HMI para la manipulación de la pinza, aún no se han configurado. En la figura VI.131 se puede ver cómo es la configuración para el botón que se encarga de bajar el Axis 3, y para el botón que se encarga de subirlo se pondrá de manera inversa los comandos.

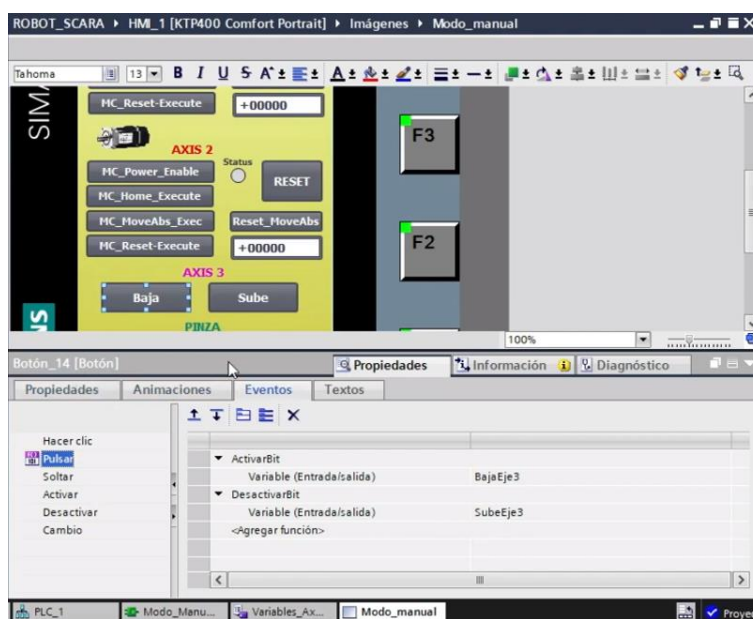


Figura VI.131 Primera parte de la configuración del botón de baja.

Estos botones también tendrán una animación, en la figura VI.132 se puede ver la configuración para los botones que se encargan de subir y bajar el Axis.

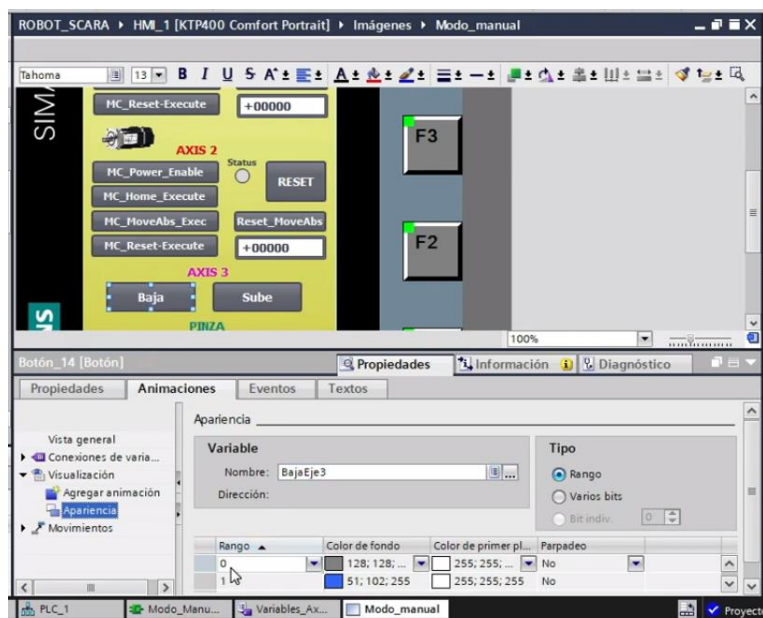


Figura VI.132 Segunda parte de la configuración del botón de baja.

También se debe configurar los botones que se encargan de abrir y cerrar la pinza, la configuración de éstos es parecida a la de los botones de baja y sube, ver en las figuras VI.133 y VI.134, pero es necesario señalar que se encuentra configurado con lógica negativa, por el control de posición configurado en el NX MCD.

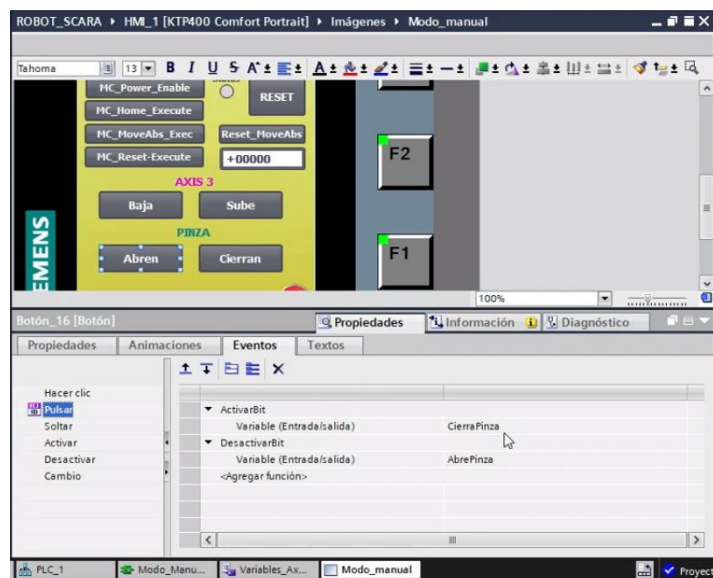


Figura VI.133 Primera parte de la configuración del botón de Abren.

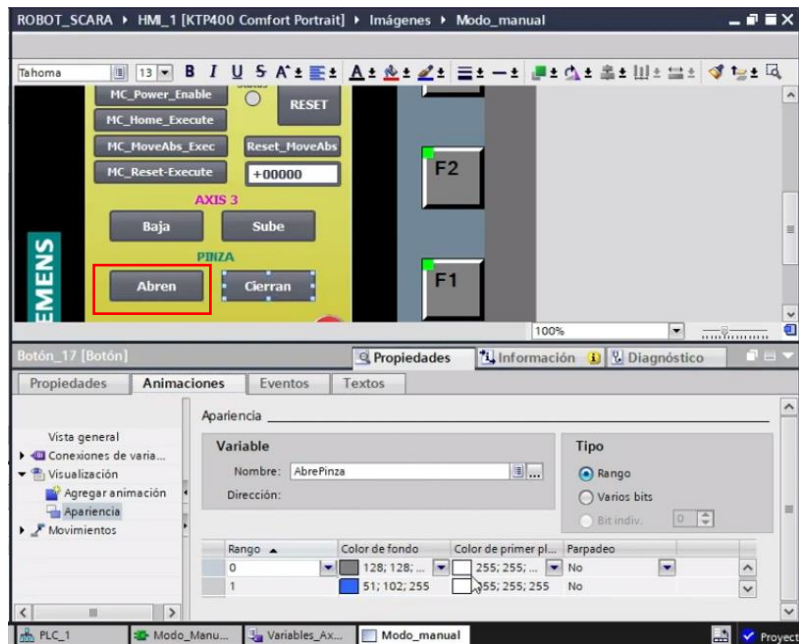


Figura VI.134 Segunda parte de la configuración del botón de Abren.

No olvidar agregar nuevos componentes en SIMIT, estos son para poder procesar las señales que se tendrán en TIA Portal y Siemens *NX MCD*, se agrega con incrementos de 50 mm/s.

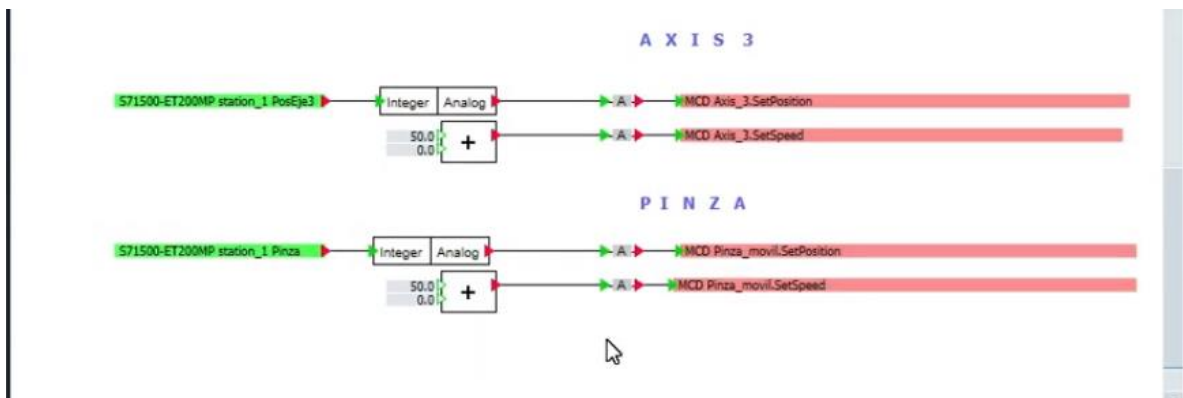


Figura VI.135 Control de Axis 3 y pinza en SIMIT.

Se ha terminado de configurar el *modo manual*, se prosigue a validar mediante simulación la apertura y cierre de la pinza y el movimiento del eje 3, en las siguientes figuras se puede observar tanto la pinza como el Axis 3 funcionan de manera correcta.

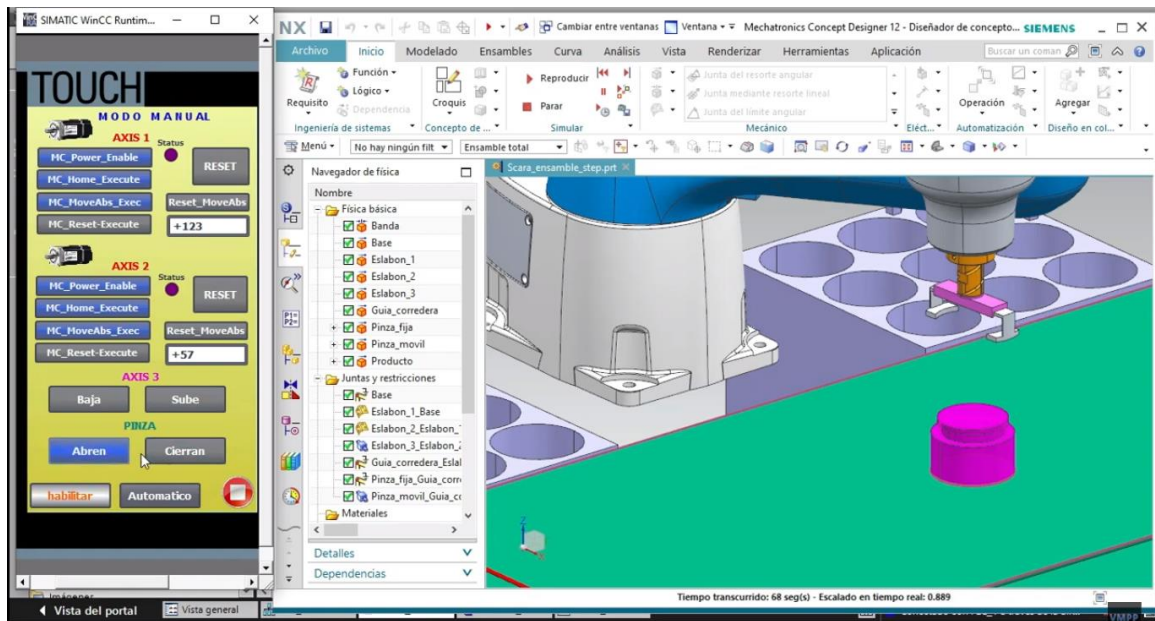


Figura VI.136 Simulación de pinza abierta.

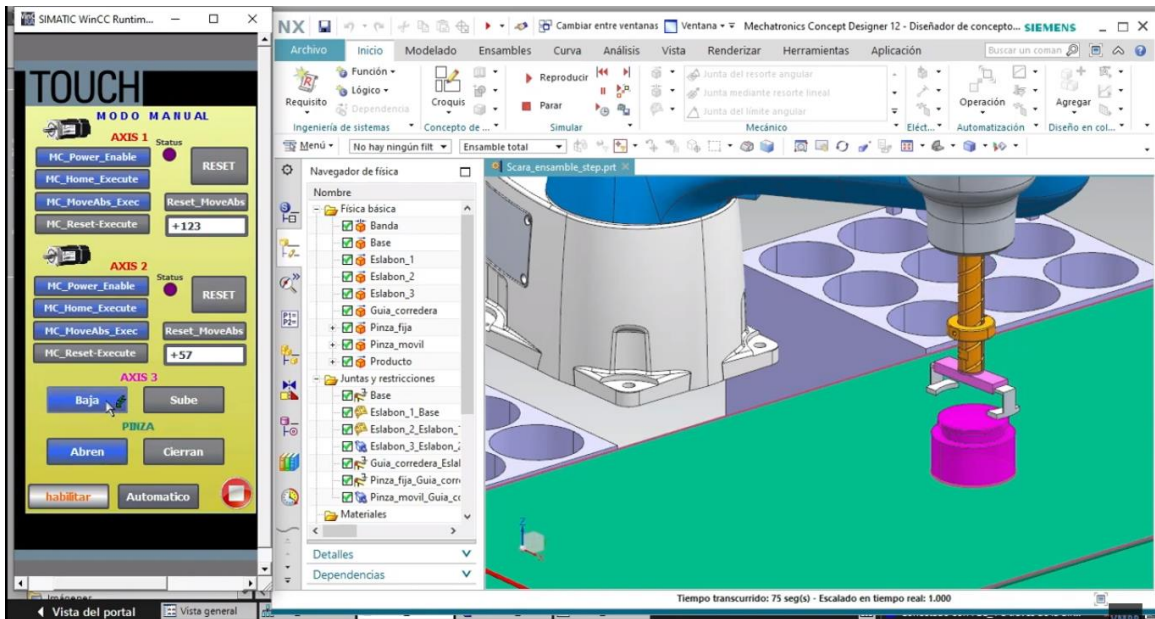


Figura VI.137 Simulación de Axis 3 bajando.

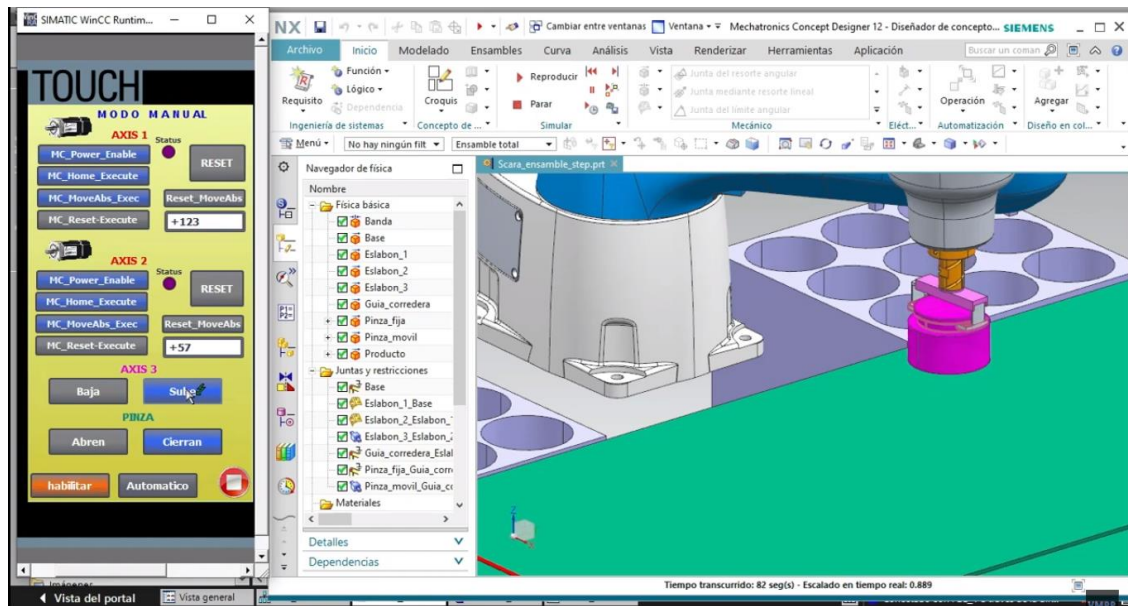


Figura VI.138 Simulación de pinza cerrada y Axis 3 subiendo.

VI.2.13 Código para el eje 3 y pinza

A continuación, se explicará el código en lenguaje de escalera para el control del eje 3 que comienza en el segmento 11 en el FC1 (*Modo_Manual*) como se ve en la figura VI.139.

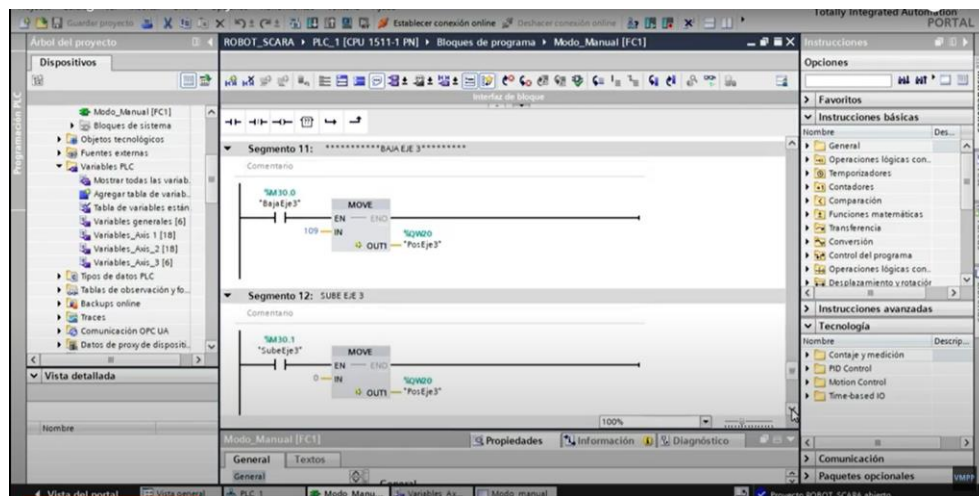


Figura VI.139 Código para el Axis 3.

Las distancias que se observan en las figuras VI.139 y VI.140 109 para el eje 3 y 15 para la pinza.

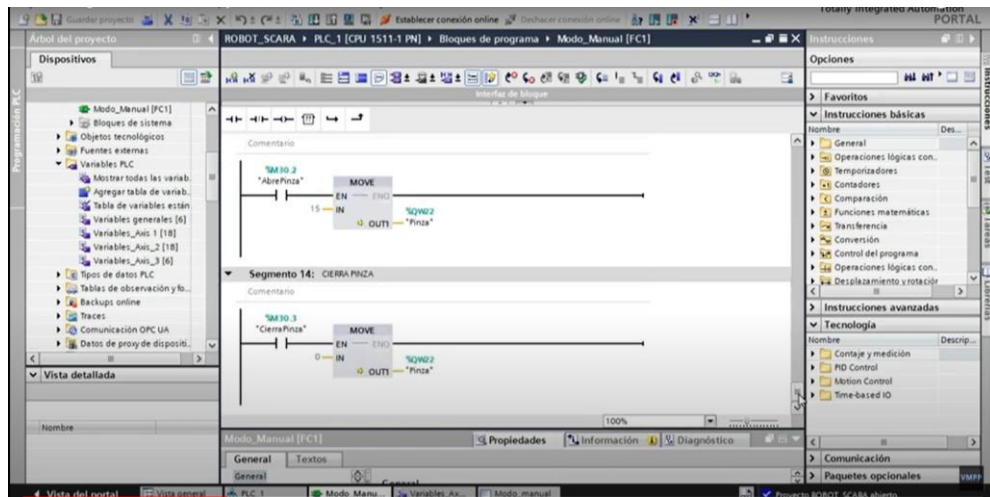


Figura VI.140 Código para la pinza.

VI.2.14 Interfaz HMI modo automático y tabla de variables generales

Es necesario hacer algunas modificaciones en la interfaz HMI del modo automático, algunas son de visualización, pero otras si son de gran importancia para el correcto funcionamiento del modo automático, el diseño de ésta ya se ha explicado anteriormente, así que en las figuras que se pueden ver más adelante solo se explicará la configuración de cada uno de los botones, no se profundizará mucho en esto debido a que cuando se configuró el modo manual se explicó de manera detallada.

Además de esto se agrega una figura donde se puede ver la tabla de variables generales, es importante observarla primero para poder llenar de manera adecuada la interfaz HMI del modo automático.

En la figura VI.142 y VI.143 se puede ver la configuración para el botón *MC_Power_Enable*, procedimientos que se emplearan para el resto de los botones de la parte superior, excepto el de RESET, la configuración de éste se puede ver en las figuras VI.144, VI.145 y VI.146.

Después se configura los botones con los que se decide en donde se requiere colocar la pieza, se tienen cuatro opciones, en las figuras VI.147, VI.148 y VI.149 se puede analizar esta configuración, será exactamente igual para los otros botones, solo que cambiará la variable llamada PIEZA1, será con PIEZA2, PIEZA3 y PIEZA4.

Por último, se tienen los botones de la parte inferior, comenzamos con el botón llamado tomar pieza, la configuración se puede ver en las figura VI.150, VI.151. El botón de colocar pieza tiene la misma configuración, pero con la variable llamada depositar pieza.

Por último, está el botón que se encarga de reiniciar el proceso, su configuración se puede analizar en las figuras VI.152 y VI.153.

ROBOT_SCARA > PLC_1 [CPU 1511-1 PN] > PLC tags > Variables generales [24]

Variables generales

	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Supervis...	Comment
1	Mod_manual	Bool	%M70.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
2	Mod_automatico	Bool	%M70.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
3	Reset	Bool	%M70.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
4	Reset_Mov_Abs1	Bool	%M70.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
5	Reset_Mov_Abs2	Bool	%M70.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
6	Reset2	Bool	%M70.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
7	Paso1	Bool	%M70.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
8	Paso2	Bool	%M70.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
9	Paso3	Bool	%M71.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
10	Paso4	Bool	%M71.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
11	Paso5	Bool	%M71.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
12	Paso6	Bool	%M71.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
13	Paso7	Bool	%M71.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
14	Paso8	Bool	%M71.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
15	TomarPza	Bool	%M71.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
16	ResetPza	Bool	%M71.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
17	DepositaraPza	Bool	%M72.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
18	PIEZA1	Bool	%M72.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
19	PIEZA2	Bool	%M72.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
20	PIEZA3	Bool	%M72.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
21	PIEZA4	Bool	%M72.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
22	Reset Auto	Bool	%M72.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
23	paso 9	Bool	%M72.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
24	paso 10	Bool	%M72.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		

Figura VI.141 Tabla de variables generales completa.

[Las variables 18, 19, 20 y 21 son las que nos servirán para poder configurar los botones que se encargarán de colocar las piezas en sus posiciones, marcas para configurar la HMI.

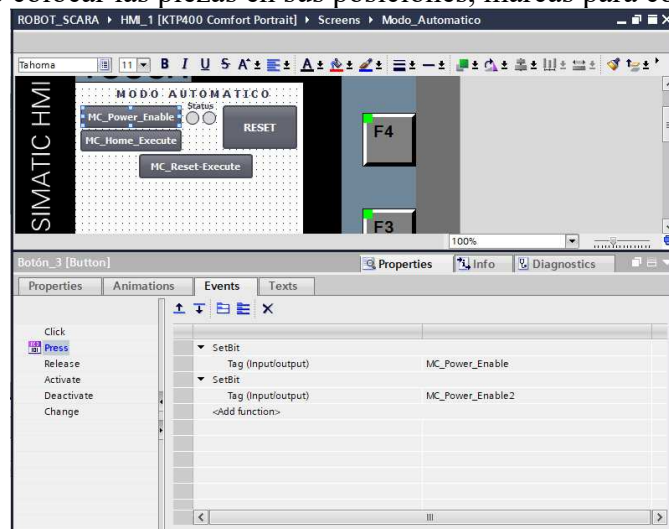


Figura VI.142 Primera parte del botón MC_Power_Enable.

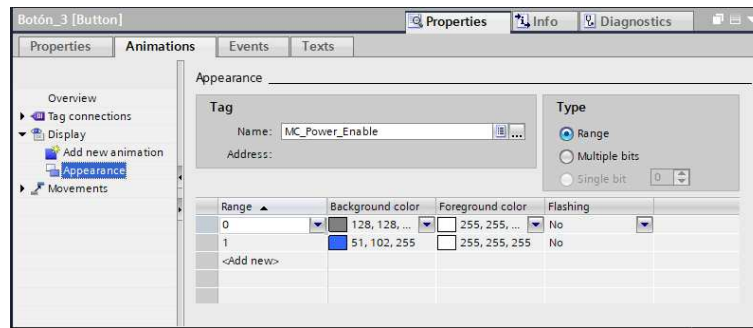


Figura VI.143 Segunda parte del botón MC_Power_Enable.

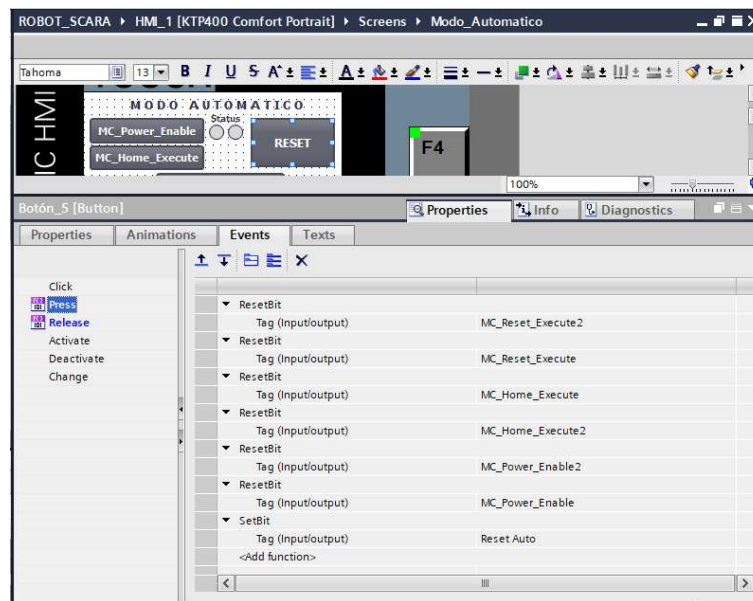


Figura VI.144 Primera parte del botón RESET.

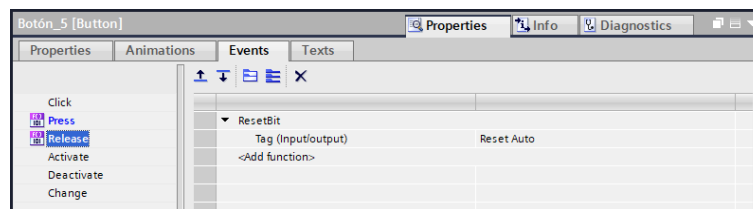


Figura VI.145 Segunda parte del botón RESET.

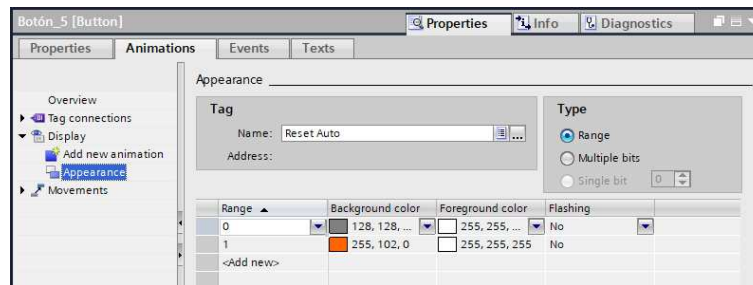


Figura VI.146 Tercera parte del botón RESET.

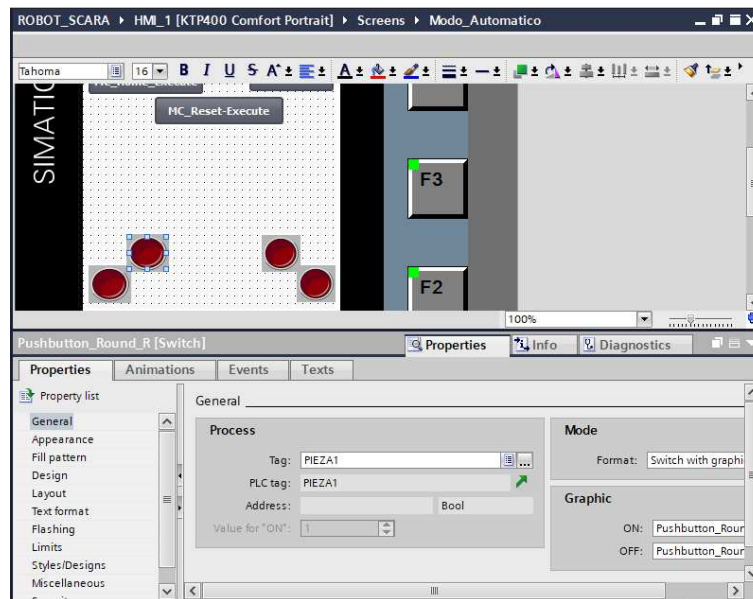


Figura VI.147 Primera parte del botón para colocar la pieza.

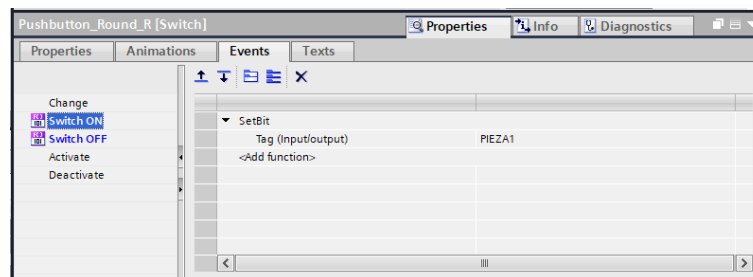


Figura VI.148 Segundo parte del botón para colocar la pieza.

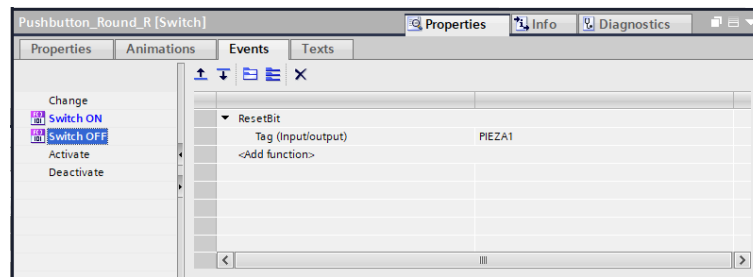


Figura VI.149 Tercera parte del botón para colocar la pieza.

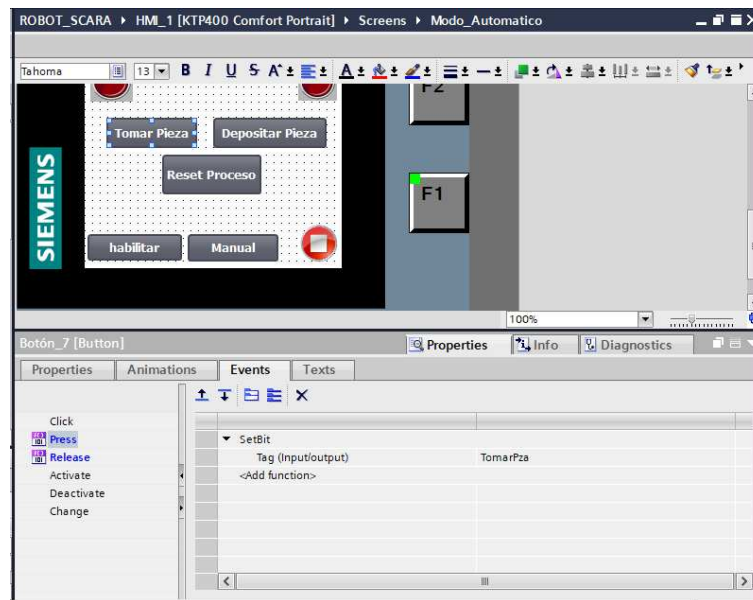


Figura VI.150 Primera parte del botón tomar pieza

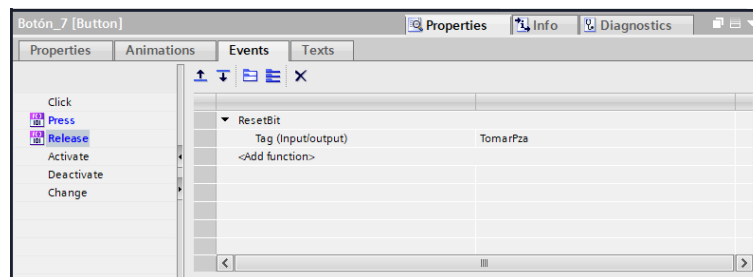


Figura VI.151 Segunda parte del botón tomar pieza.

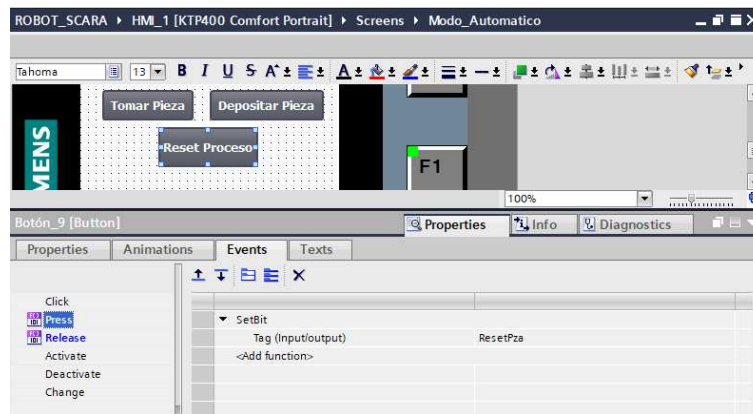


Figura VI.152 Primera parte del botón Reset Proceso.

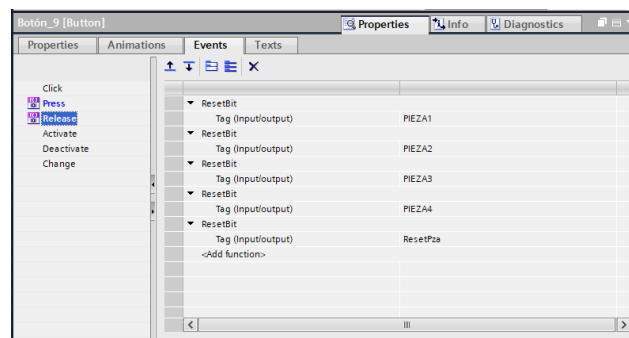


Figura VI.153 Segunda parte del botón Reset Proceso.

VI.2.15 Secuencia final Modo Automático

Ya se explicó de manera detallada lo que es el *modo manual* pero también se tiene el *modo automático*, el cual es bastante sencillo de explicar, tomando como referencia la figura VI.154, hay que “encender” el brazo para esto se presionan los botones 1 y 2, después de esto se debe de seleccionar una posición (paso 3) en donde se desea colocar la pieza de trabajo para después presionar el botón de tomar pieza (paso 4), seguido de esto se presiona el botón de depositar pieza (paso 5). El resultado final es el que se ve en la figura, cuando el brazo deposita la pieza vuelve a su posición original.

Pero si se desea repetir la simulación se debe de parar en Siemens *NX MCD*, esto se hace presionando donde indica la flecha 6 (Botón de *Reset Proceso*), para posteriormente se dar en reproducir, luego se vuelven a hacer todos los pasos descritos hasta el momento en este último apartado. El botón de *Reset Proceso* se presiona cuando se desea parar el proceso antes de que termine, este mandará el brazo a su posición inicial (Home).

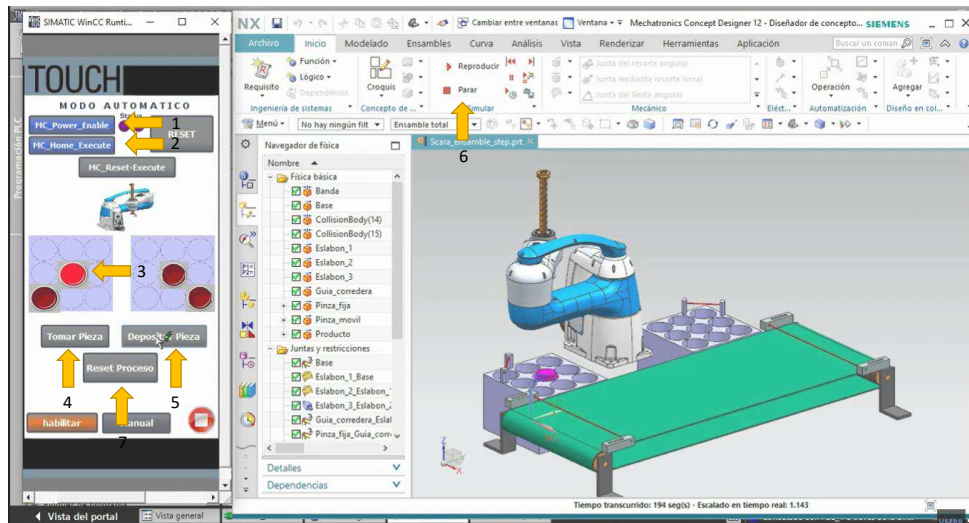


Figura VI.154 Simulación Modo Automático.

VI.2.16 Bloque lógico FC Modo Automático

En el segmento 1 se tiene el bloque que se encarga de encender un eje del brazo, en otras palabras, es el que se encarga de habilitar el objeto tecnológico, seguido de esto la referencia del objeto tecnológico. El bloque de posicionamiento absoluto fue modificado, ahora se agregaron pasos, estos se explicarán con detalle más adelante. También se integró lo que es bloque de *reset*, se explica también el otro bloque que es necesario para habilitar el otro eje del brazo y después se ven los mismos bloques que se mencionaron anteriormente ya que ambos ejes necesitan de todos estos bloques. Por último, se tiene el bloque de condiciones y el bloque de acciones, los cuales se explicarán más adelante.

- Primera parte del bloque de *modo automático*, Axis 1:

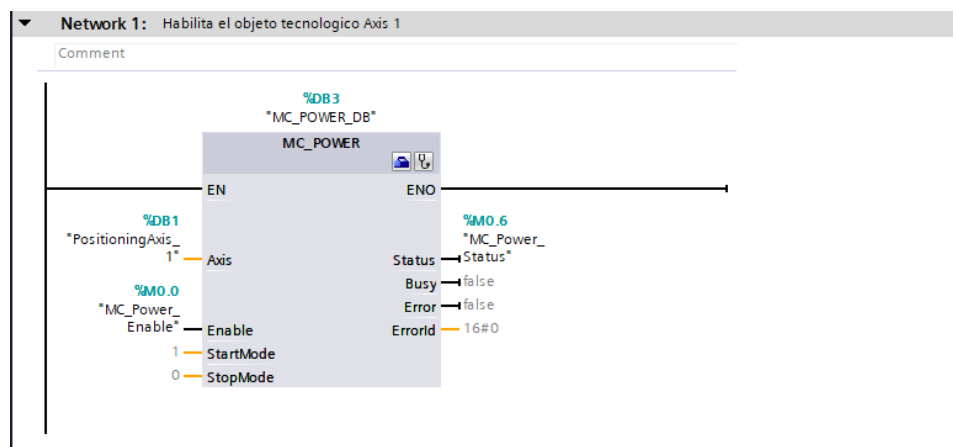


Figura VI.155 Bloque MC_Power.

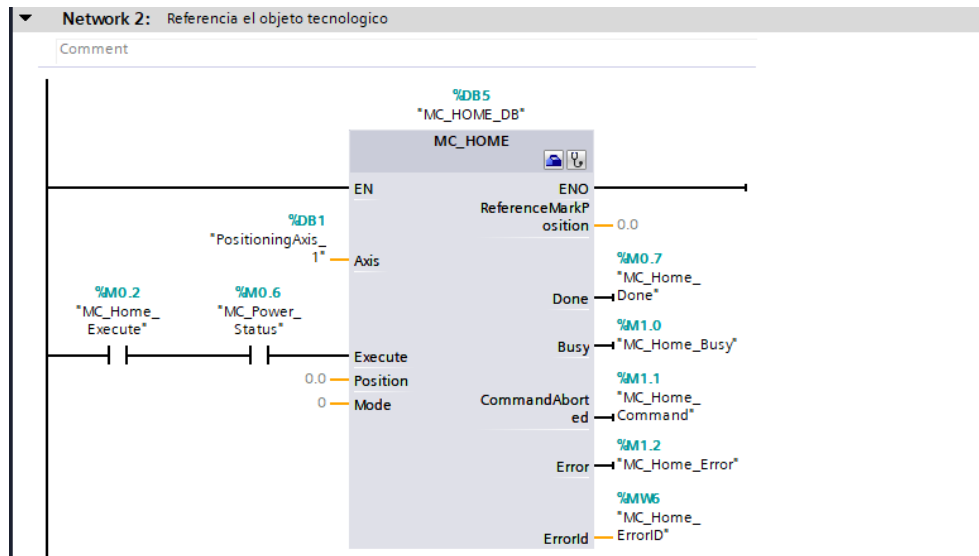


Figura VI.156 Bloque MC_Home.

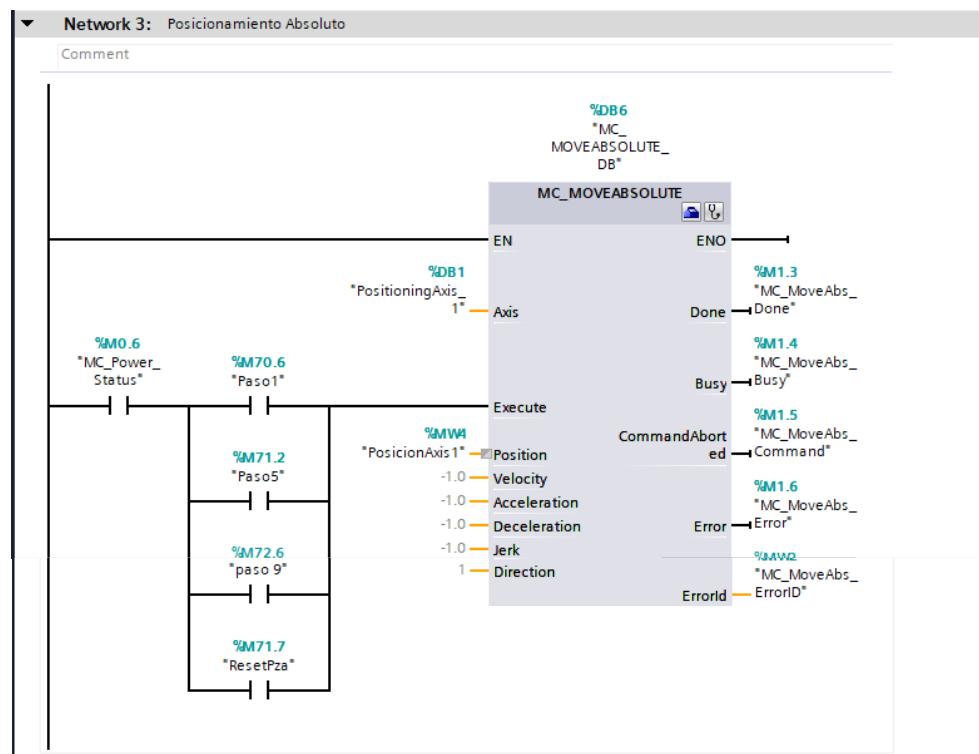


Figura VI.157 Bloque MC_MoveAbsolute.

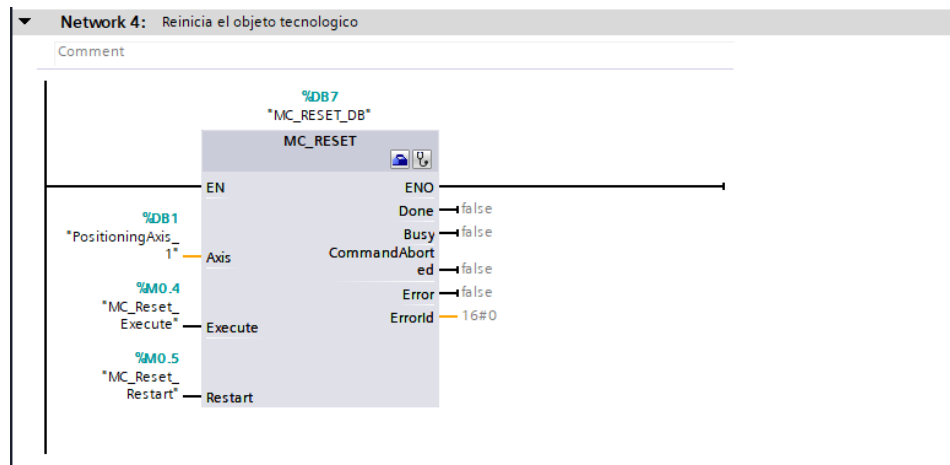


Figura VI.158 Bloque MC_Reset.

- Segunda parte del bloque de *modo automático*, Axis 2:

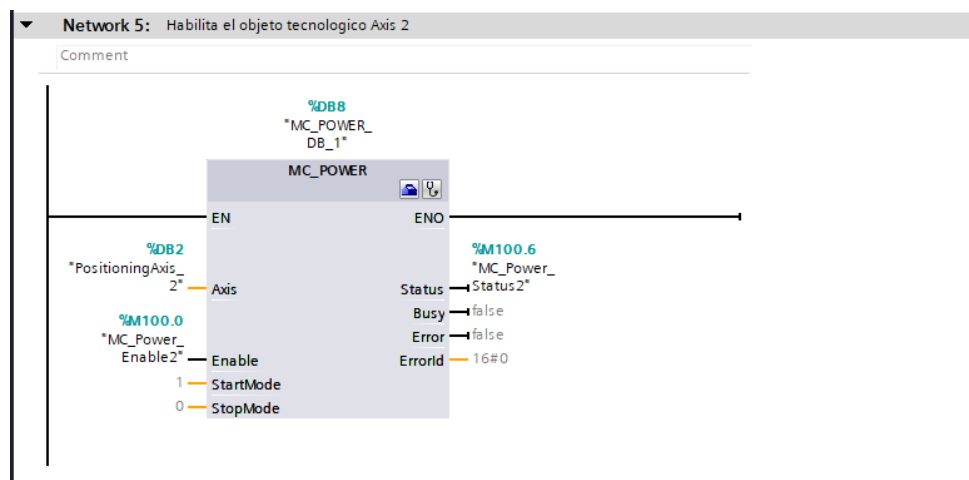


Figura VI.159 Bloque MC_Power.

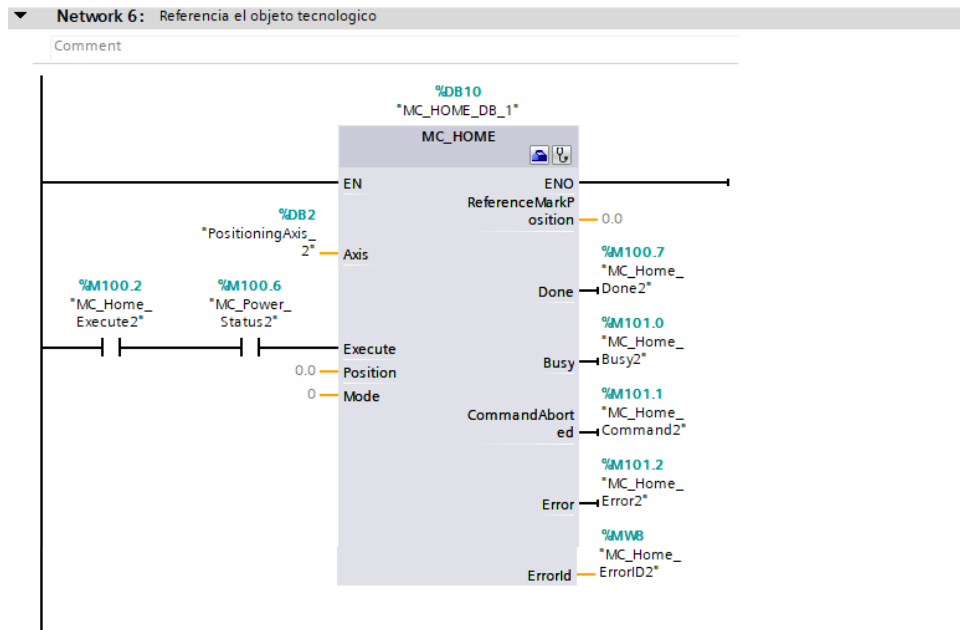


Figura VI.160 Bloque MC_Home.

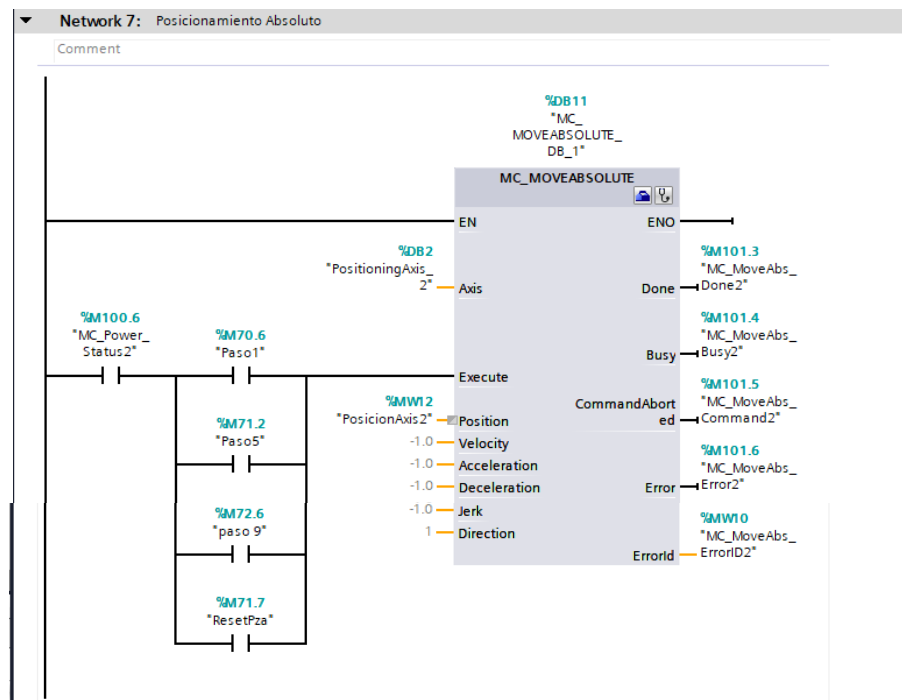


Figura VI.161 Bloque MC_MoveAbsolute.

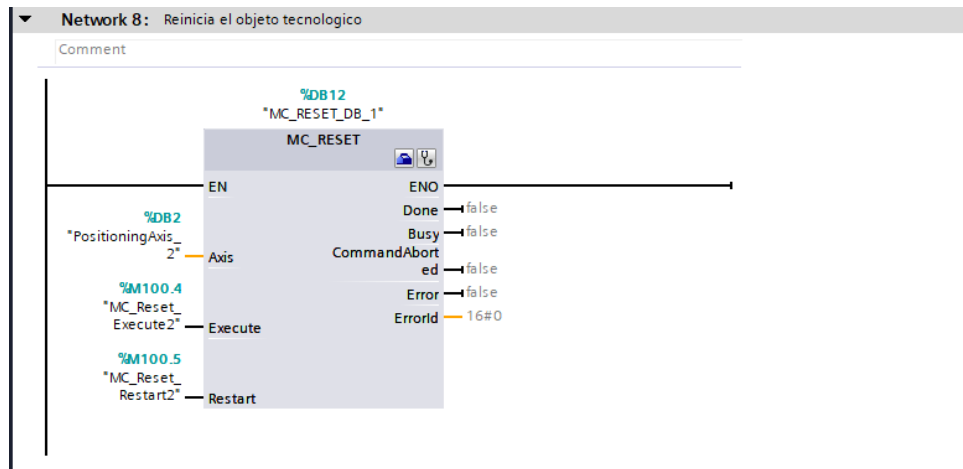


Figura VI.162 Bloque MC_Reset.

- Tercera parte del bloque de *modo automático*, bloque de condiciones y bloque de acciones.

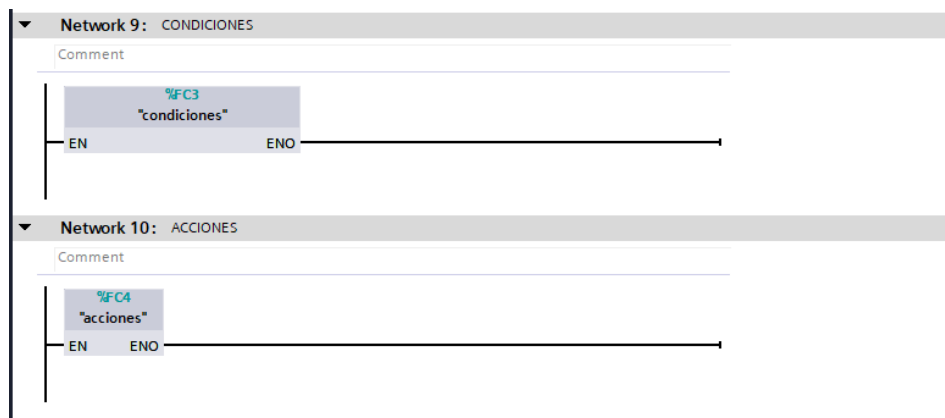


Figura VI.163 Bloque de condiciones y bloque de acciones para las pinzas y eje 3.

El código para el control de la pinza y del eje 3 en el modo automático se realiza en un bloque funcional de tipo FC3 uno para condiciones y otro FC4 para acciones, el cual se explicará a continuación por segmento:

El retardo de 3500 ms (T1) es el tiempo necesario para que el brazo llegue a su posición: 123, 57, previamente validada en el ensamble en *NX MCD*. Con valor de 0 se asegura que la pinza esté abierta antes de bajar a tomar la pieza de trabajo. La técnica de la programación es por pasos.

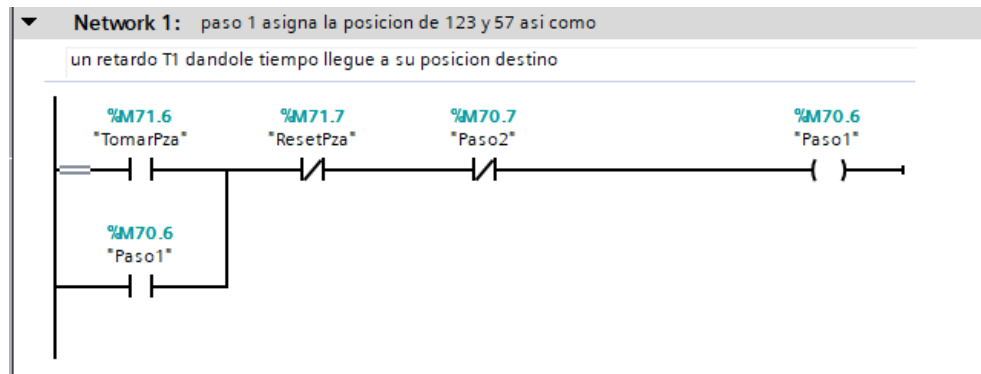


Figura VI.164 Segmento 1 condiciones.

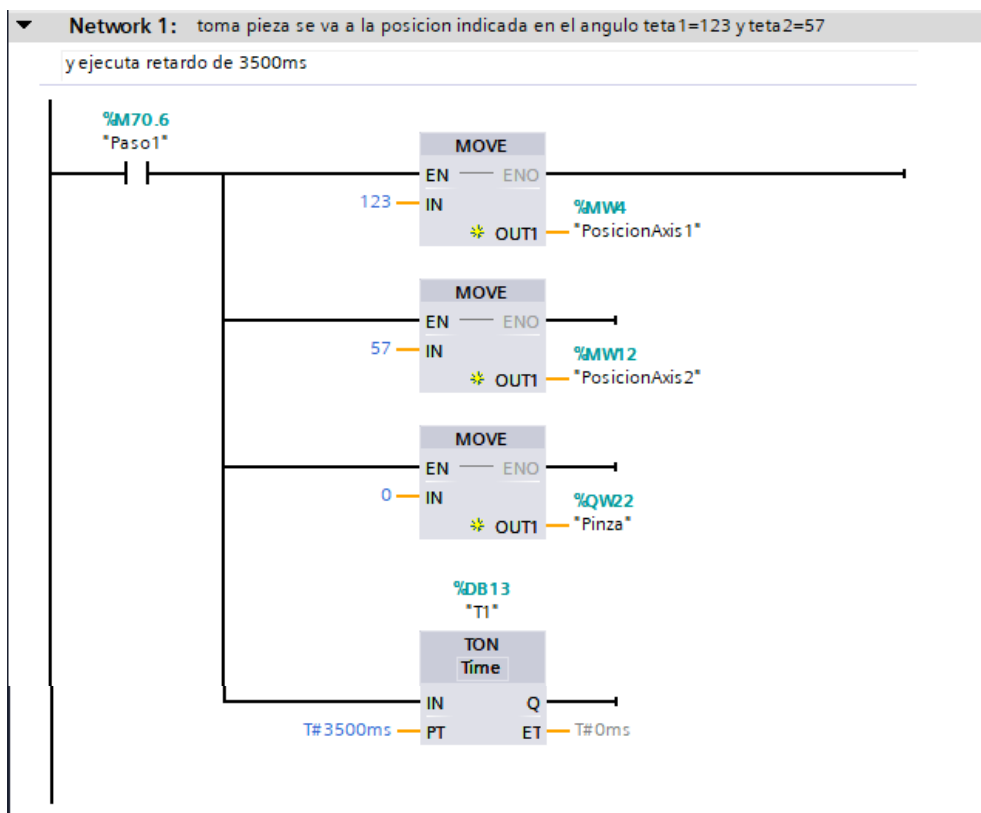


Figura VI.165 Segmento 1 acciones.

En el segmento 2 con el valor de 109 el Axis 3 baja para tomar la pieza de trabajo, el retardo de 3000 ms (T2) es el tiempo que el Axis 3 espera con la pinza abierta antes de cerrar la pinza.

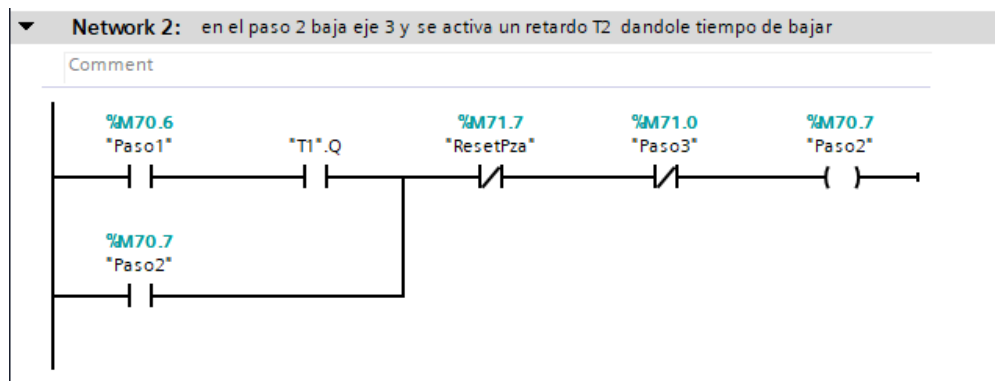


Figura VI.166 Segmento 2 condiciones.

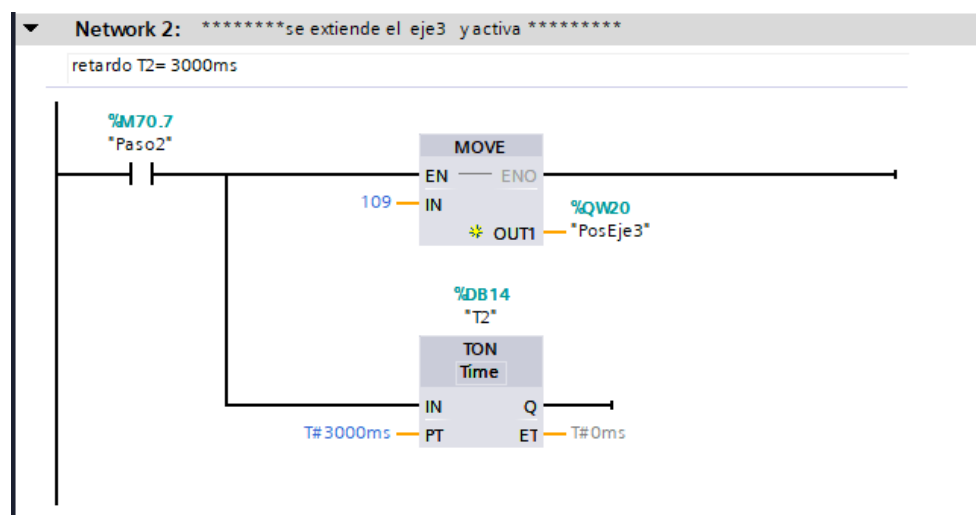


Figura VI.167 Segmento 2 acciones.

En el segmento 3 con el valor de 16 se cierran las pinzas y espera un tiempo de 1500 ms (T3) antes de subir el Axis 3 en el siguiente paso.

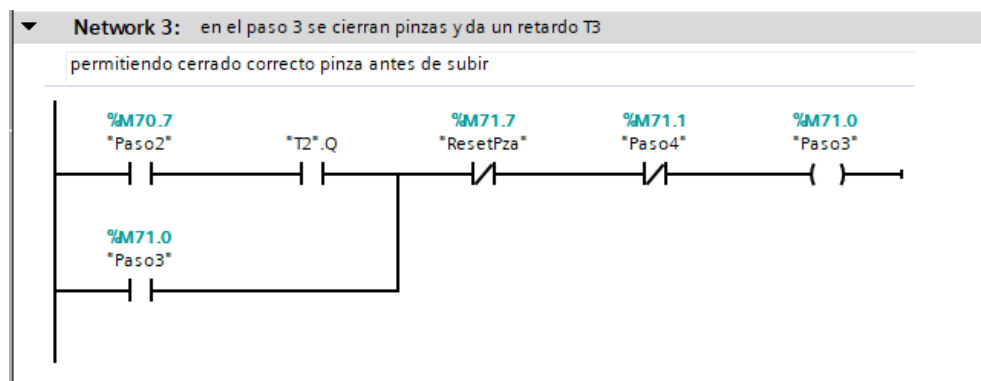


Figura VI.168 Segmento 3 condiciones.

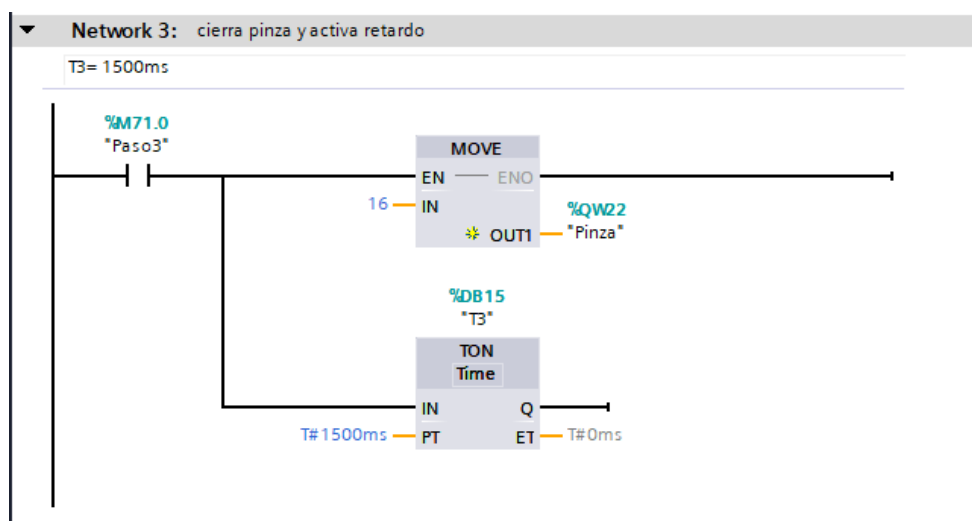


Figura VI.169 Segmento 3 acciones.

En este segmento con el valor de 0 sube el Axis 3 y se le da un tiempo de 2000 ms (T4) para permitir llegar el Axis 3 hasta la posición 0 antes de ejecutar el siguiente paso.

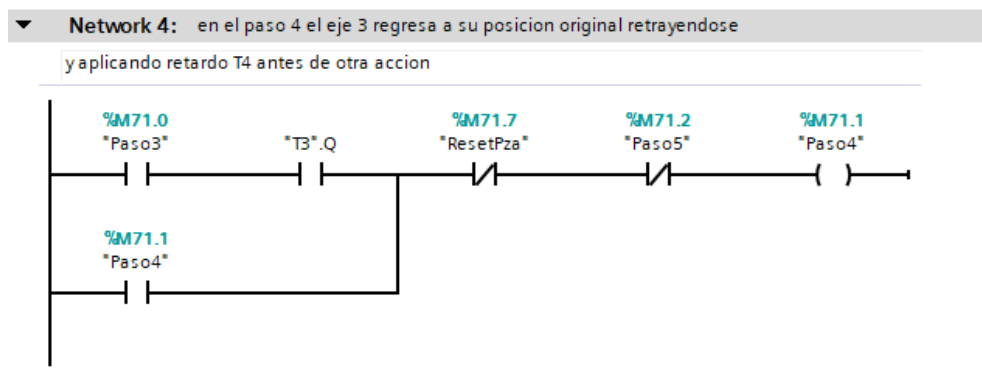


Figura VI.170 Segmento 4 condiciones.

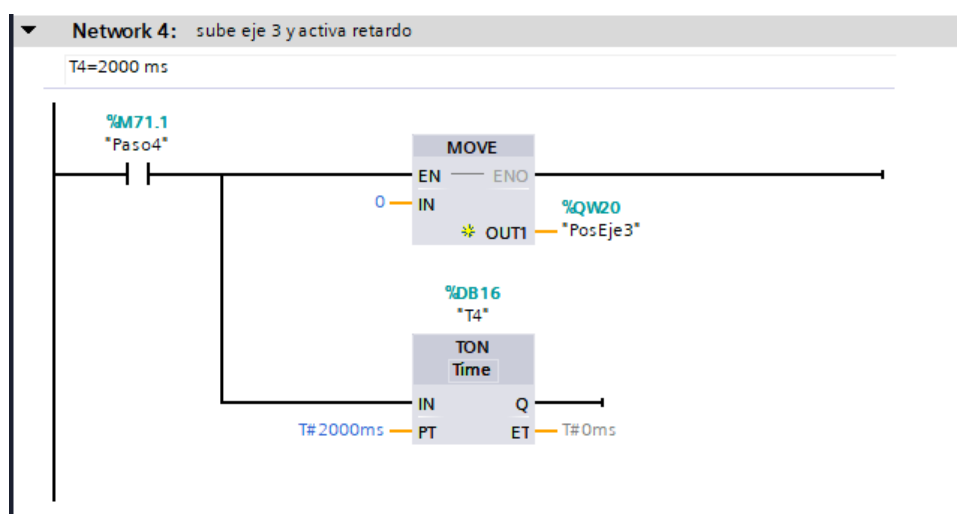


Figura VI.171 Segmento 4 acciones.

Ahora con el valor de 53, 28 y con la orden de depositar la pieza se posiciona el brazo en el lugar de depósito dando un retardo de 3500 ms (T5) antes de ejecutar el siguiente paso que es bajar a depositar la pieza el Axis 3.

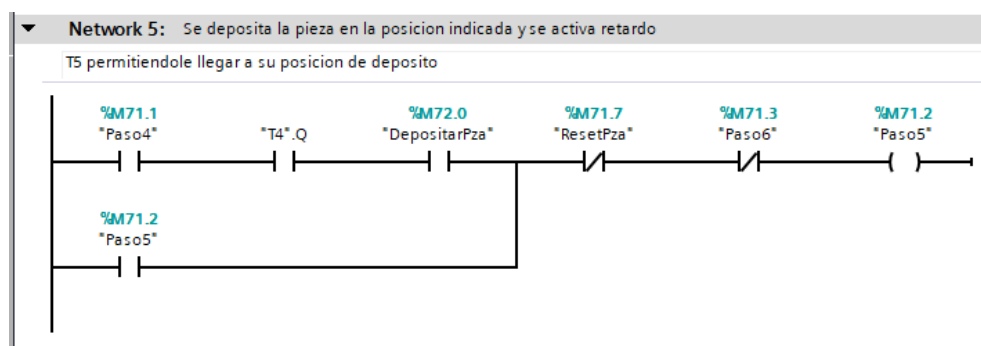


Figura VI.172 Segmento 5 condiciones.

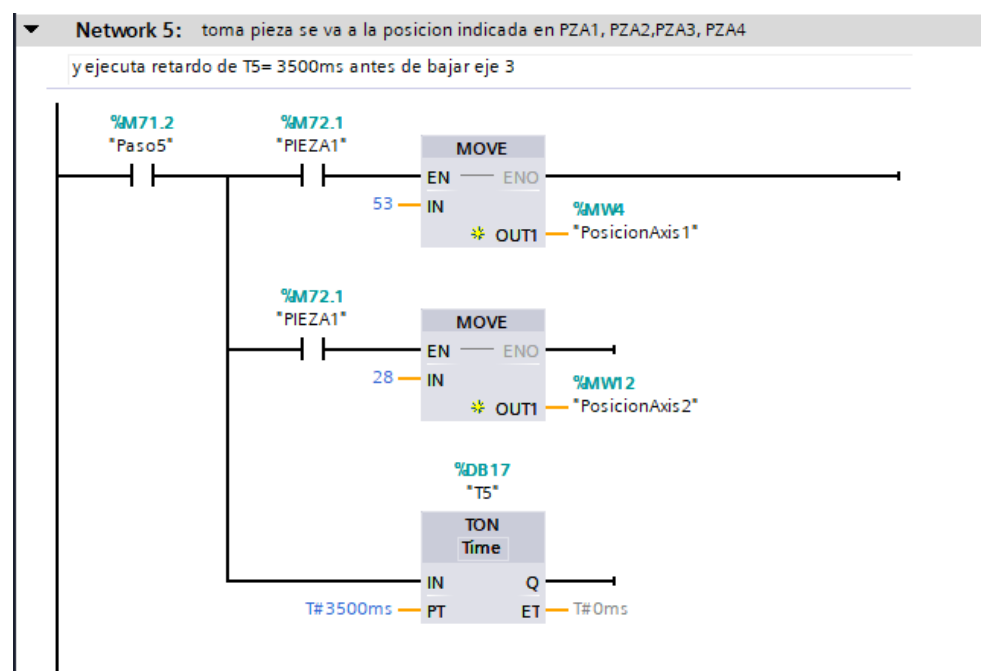


Figura VI.173 Segmento 5 acciones.

Con el valor de 115 se extiende el Axis 3 hasta la posición de depósito dando un retardo de 3500 ms (T6) para permitir que llegue a su posición de destina antes de ejecutar el siguiente paso.

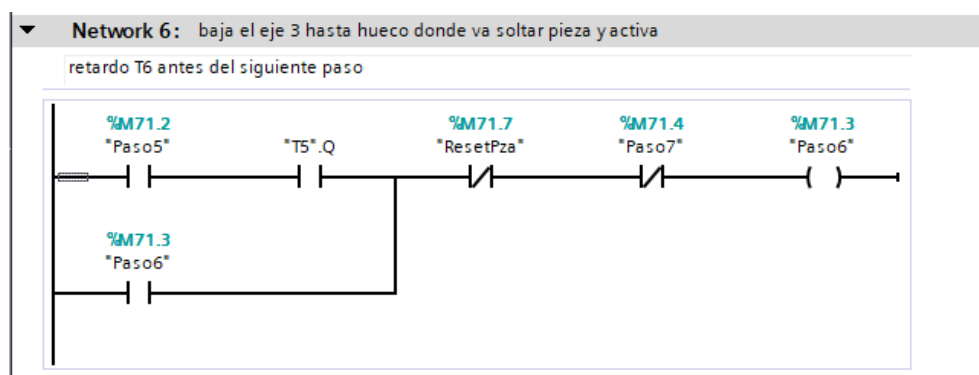


Figura VI.174 Segmento 6 condiciones.

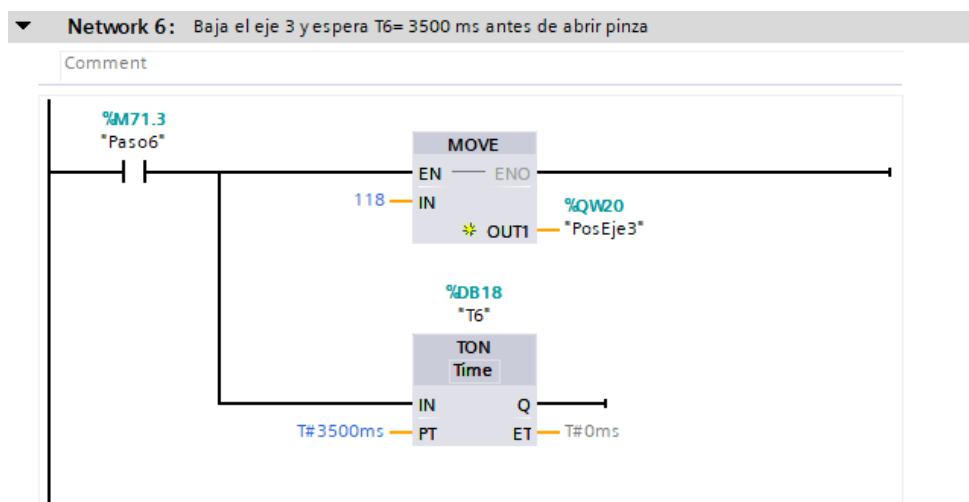


Figura VI.175 Segmento 6 acciones.

En el segmento 7 con el valor de 0 se abre la pinza permitiéndole soltar la pieza y se aplica un retardo de 1500 ms (T7) antes de subir el Axis 3.

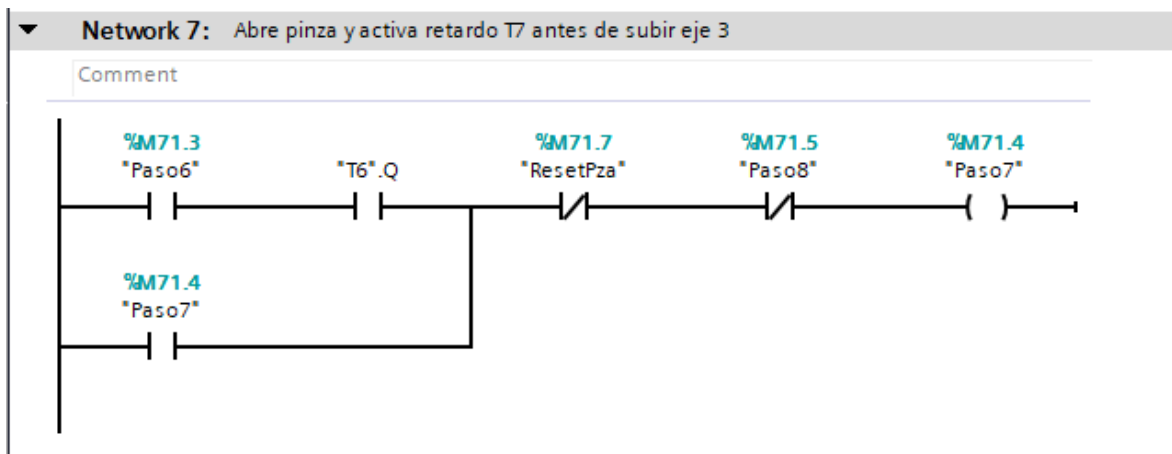


Figura VI.176 Segmento 7 condiciones.

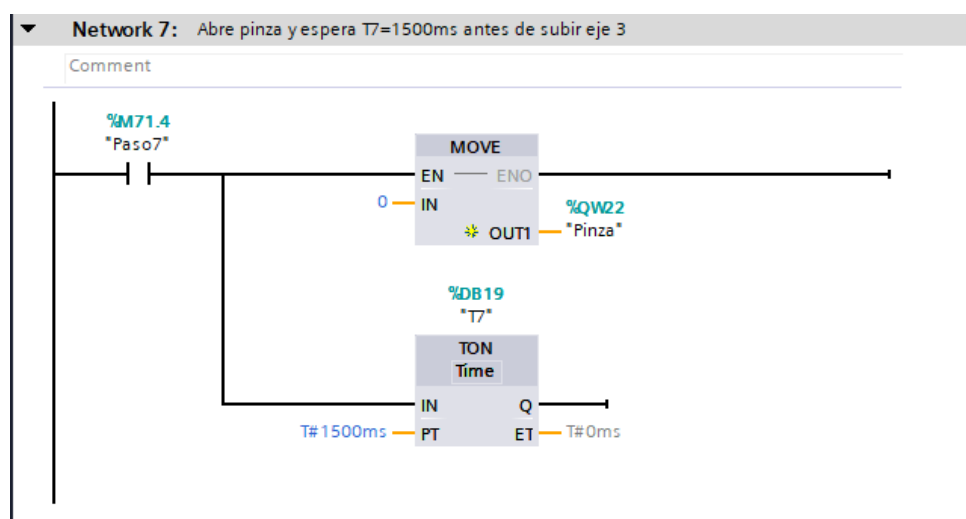


Figura VI.177 Segmento 7 acciones.

Ahora con el valor de 0 sube el Axis 3 y activa un retardo de 2000 ms (T8) antes de ir a la posición de origen.

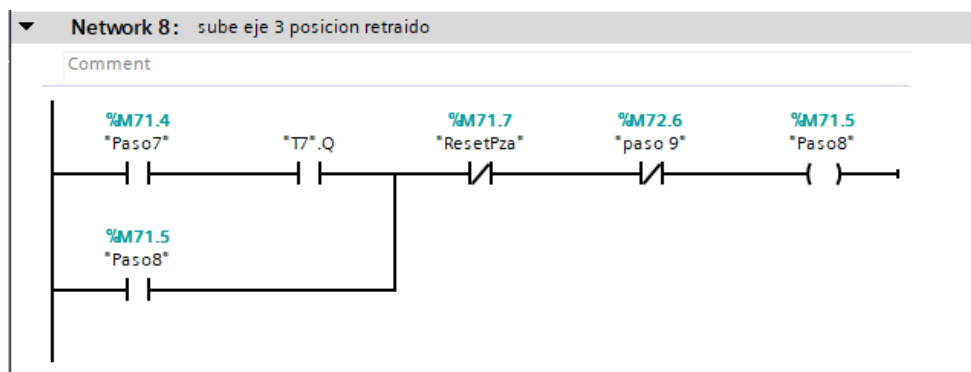


Figura VI.178 Segmento 8 condiciones.

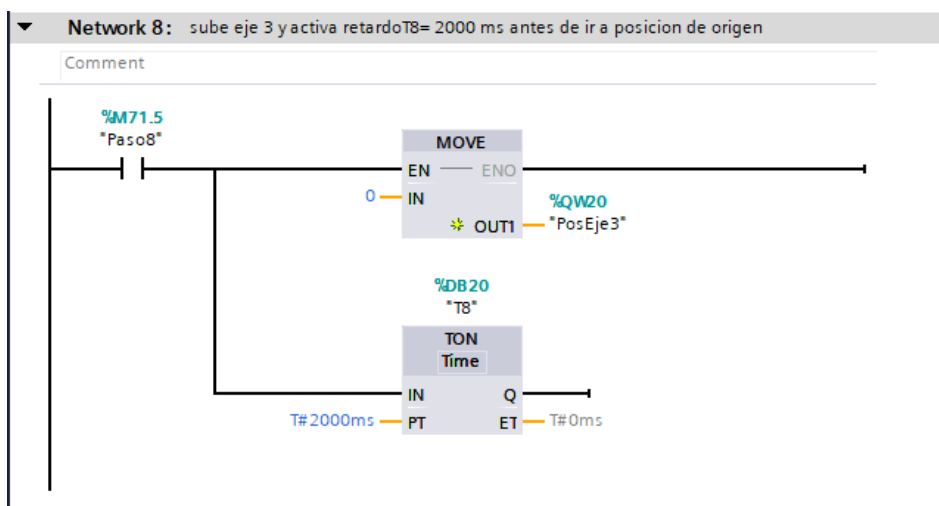


Figura VI.179 Segmento 8 acciones.

En el segmento 9 con el valor de 0 todos los elementos se van a la posición original y activa un retardo de 2500 ms (T9) dándole tiempo a llegar antes de que se apague el paso 9.

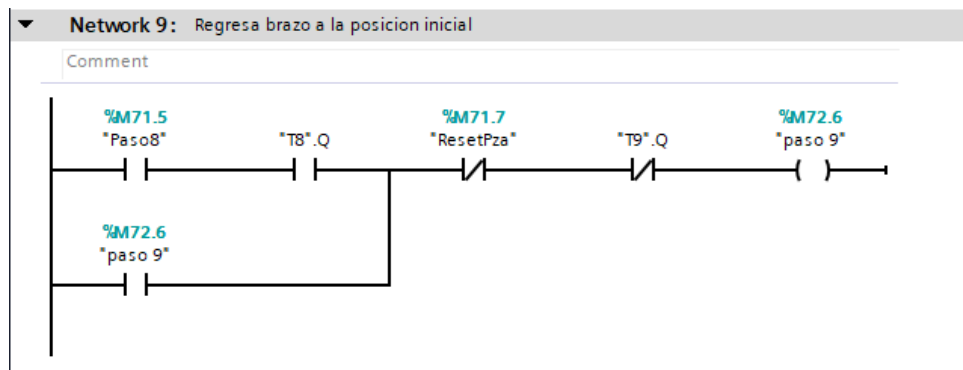


Figura VI.180 Segmento 9 condiciones.

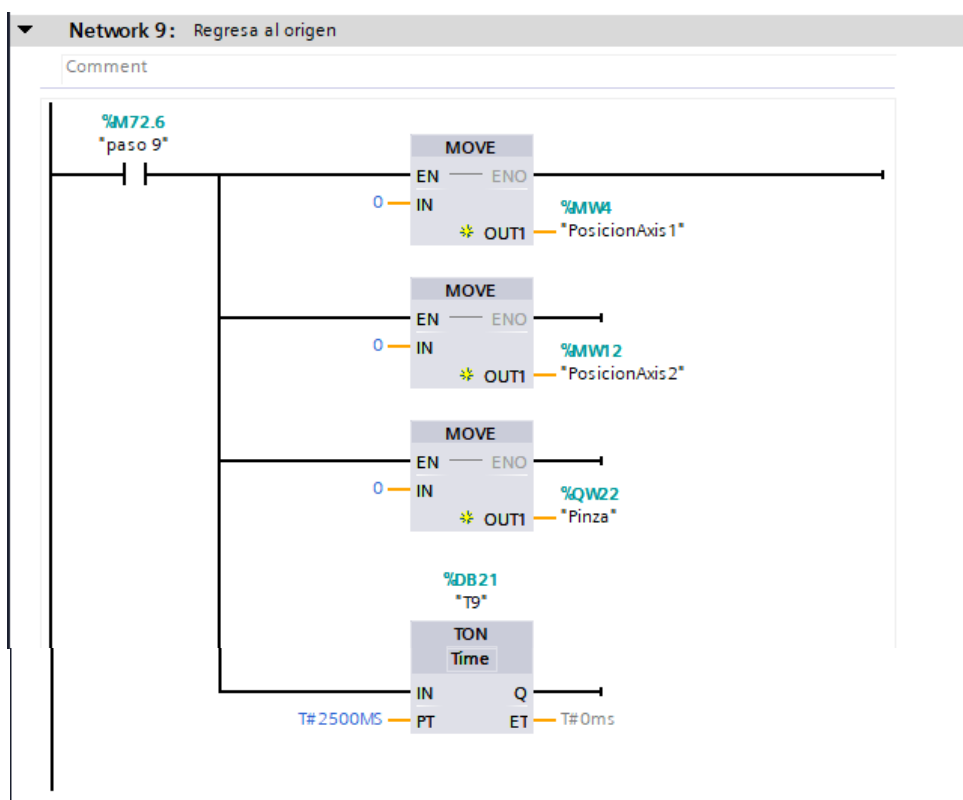


Figura VI.181 Segmento 9 acciones.

Si por alguna razón fuera necesario reiniciar el proceso antes de que termine completamente la secuencia, se tiene el segmento 10 con las condiciones que se encargan de poner todos los elementos en 0.

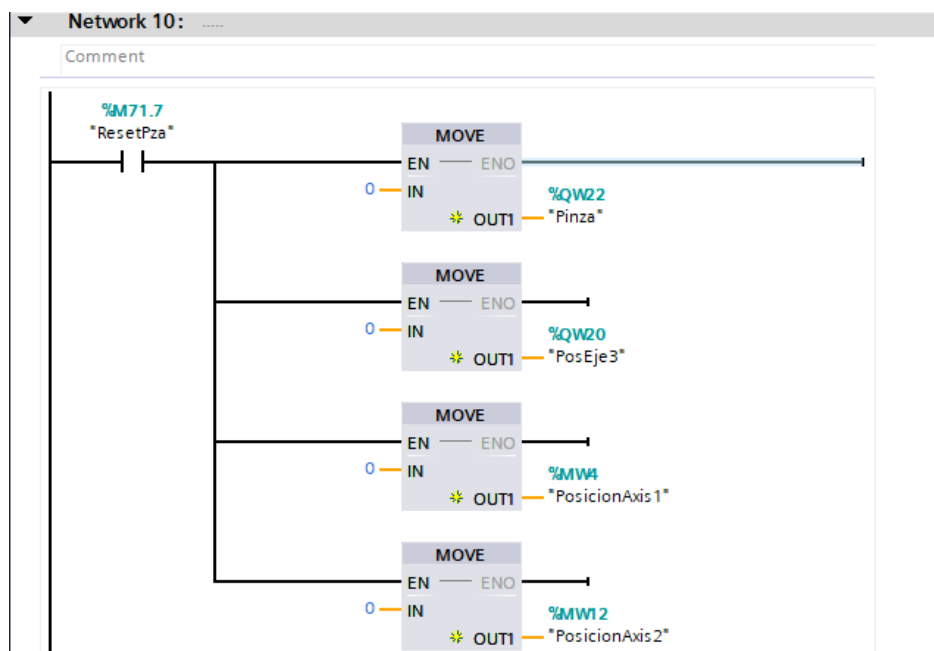


Figura VI.183 Segmento 10 condiciones.

Laboratorios Virtuales para Automatización

Asociación Mexicana de Mecatrónica A.C.
Prolongación Corregidora Norte No. 168, interior E
Col. El Cerrito, C.P. 76160, Querétaro, Qro.
Distribución en formato digital por internet,
formato: PDF, tiraje: 1000, pp. 280, Tamaño: 26.2 Mb.

ISBN: 978-607-9394-26-4

ISBN: 978-607-9394-26-4



9 786079 394264

