



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería del Diseño

MODELADO, SIMULACIÓN Y CONTROL DE UN GEMELO DIGITAL DE UNA MÁQUINA DE MONTAJE DE RODAMIENTOS

TRABAJO FINAL DEL

Máster U. en Diseño y Fabricación Integrada Asistidos por Computador

REALIZADO POR

M^a Paz García Simarro

TUTORIZADO POR

Antonio Correcher Salvador

CURSO ACADÉMICO: 2019/2020



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño

TRABAJO FIN DE MÁSTER

Modelado, simulación y control de un gemelo digital de una máquina de montaje de rodamientos

**Máster Universitario en Diseño y Fabricación
Integrada Asistidos por Computador**

2019 / 2020

Autora: M^a Paz García Simarro

Tutor: Antonio Correcher Salvador

Departamento de Ingeniería de
Sistemas y Automática

Albacete, noviembre 2020

A mi tutor Antonio, por motivar a los alumnos de la forma en que lo hace y contagiarnos su pasión, haciéndonos creer en nuestro potencial y motivándonos a seguir avanzando. Gracias por su apoyo y dedicación en todo momento, sin él este trabajo no hubiera sido posible.

A mis compañeras de máster, por haber sido un apoyo y ayuda en los momentos de presión y por el placer que supone hacer cualquier trabajo con ellas, y a mis compañeras de piso, Alba, Ainhoa y Antonella, por soportar mis nervios cada día y hacer más amena una etapa complicada.

A mi familia, en especial a mi madre, por apoyarme para seguir formándome y animarme a conseguir lo que me propongo, sin su apoyo mi formación no hubiera sido posible.

Resumen

El presente proyecto consiste en el desarrollo completo de un gemelo digital de una célula de montaje de rodamientos para su automatización.

En las primeras páginas se presenta brevemente en qué consiste este tipo de desarrollos y simulaciones, así como sus posibles aplicaciones y ventajas, tanto en la industria como en actividades formativas.

A continuación, se describirán las piezas y mecanismos que componen las distintas etapas y subprocesos que forman el proceso completo, incluyendo la parte física como los posibles movimientos.

Más adelante se detalla el funcionamiento y restricciones de cada etapa, así como la secuencia que debe seguir el proceso para que no se produzcan colisiones y el montaje se realice en el orden correcto.

Conociendo detalladamente el proceso real y habiendo conseguido un modelo 3D del mismo en Siemens® NX12, finalmente se estudiará y desarrollará el control del éste mediante CODESYS®, cuyo resultado permitirá además controlar de forma remota el modelo real.

Índice general

ÍNDICE DE TABLAS	7
ÍNDICE DE FIGURAS.....	8
1. Objeto	11
2. Alcance y exclusiones.....	12
3. Introducción	13
4. Normativa aplicable.....	15
5. Modelado del sistema de ensamblaje.....	16
5.1. Introducción	16
5.2. Descripción del sistema.....	17
5.3. Pieza a ensamblar.....	20
5.4. Alimentador de bases	21
5.5. Comprobador de orientación.....	22
5.6. Empujador a zona de montaje	23
5.7. Superficie de desplazamiento y zona de montaje	24
5.8. Montaje del rodamiento	25
5.9. Montaje del eje	26
5.10. Colocación de la tapa	27
5.11. Expulsión de la pieza	28
5.12. Modelo completo.....	29
6. Automatización del proceso	30
6.1. Descripción del proceso	30
6.1.1. Inicio del proceso.....	31
6.1.2. Alimentación de bases y comprobador de posición	32
6.1.3. Montaje del rodamiento	33
6.1.4. Montaje del eje del eje.....	34
6.1.5. Colocación de la tapa y expulsión	35
6.2. Definición de parámetros y componentes.....	36
6.2.1. Componentes mecánicos	36

6.2.2. Componentes eléctricos	40
6.3. Simulación del proceso en Siemens® NX12	44
6.4. Integración de Siemens® NX12 en CODESYS® a través de TCP/IP	47
6.4.1. Controles de posición	47
6.4.2. Sensores de posición y distancia	49
6.4.3. Agarre de piezas	50
6.4.4. Comunicación de señales	51
6.5. Automatización del gemelo digital.....	54
7. Valoración económica.....	55
7.1. Costes directos	55
7.1.1. Coste de mano de obra directa (C_{MOD})	55
7.1.2. Coste de material (C_{MAT})	56
7.1.3. Coste de equipos (C_{MAQ})	56
7.2. Costes indirectos y costes generales.....	57
7.3. Coste total del proyecto.....	57
8. Conclusiones.....	58
Anexo I. Conexión de señales.....	59
Mapeo de señales	61
Anexo II. Accionamientos monoestables	63
Referencias	64

ÍNDICE DE TABLAS

Tabla 1. Piezas que componen el alimentador	21
Tabla 2. Piezas que componen el comprobador de orientación.....	22
Tabla 3. Piezas que componen el empujador	23
Tabla 4. Piezas que componen la zona de montaje y superficie de desplazamiento	24
Tabla 5. Piezas que componen el manipulador del rodamiento	25
Tabla 6. Piezas que componen el manipulador de ejes	26
Tabla 7. Piezas que componen el manipulador de la tapa.....	27
Tabla 8. Piezas que componen el pistón de expulsión de piezas.....	28
Tabla 9. Resumen del total de piezas que componen el proceso.....	29
Tabla 10. Descripción de las entradas del sistema.....	30
Tabla 11. Descripción de las salidas del sistema.....	31
Tabla 12. Coste de mano de obra directa	55
Tabla 13. Costes de material correspondientes a las licencias de software.....	56
Tabla 14. Costes de directos de equipos y maquinaria.....	56
Tabla 15. Costes totales del proyecto	57

ÍNDICE DE FIGURAS

Figura 1. Ejemplo de SFC.....	15
Figura 2. Descripción de la misma acción en distintos lenguajes	15
Figura 3. Esquema de comunicaciones	16
Figura 4. Célula de ensamblaje SMC MAP-205	17
Figura 5. Estación de alimentación de bases	18
Figura 6. Estación de inserción del rodamiento.....	18
Figura 7. Estación de inserción del eje.....	19
Figura 8. Estación de colocación de la tapa	19
Figura 9. Pieza final obtenida en el proceso de montaje.....	20
Figura 10. Componentes ensamblados durante el proceso	20
Figura 11. Ensamblaje del alimentador de bases	21
Figura 12. Ensamblaje del comprobador	22
Figura 13. Ensamblaje del empujador	23
Figura 14. Ensamblaje de la zona de montaje	24
Figura 15. Ensamblaje del actuador.....	25
Figura 16. Ensamblaje del actuador.....	26
Figura 17. Ensamblaje del actuador.....	27
Figura 18. Ensamblaje del pistón de expulsión.....	28
Figura 19. Ensamblaje del proceso completo	29
Figura 20. GRAFCET de arranque del sistema.....	31
Figura 21. GRAFCET de la alimentación de bases	32
Figura 22. Ubicación de los sensores del primer subproceso.....	32
Figura 23. GRAFCET del montaje del rodamiento.....	33
Figura 24. Ubicación de los sensores del segundo subproceso	33
Figura 25. Ubicación de los sensores del tercer subproceso	34
Figura 26. GRAFCET del montaje del eje.....	34
Figura 27. Ubicación de los sensores del último subproceso	35
Figura 28. GRAFCET del montaje de tapa y expulsión	35
Figura 29. Componentes mecánicos que permite definir el programa	36
Figura 30. Física básica del proceso completo de montaje de rodamientos.....	37
Figura 31. Ventana del comando junta fija	38

Figura 32. Ventana del comando junta de charnela.....	38
Figura 33. Ventana del comando junta deslizante	39
Figura 34. Juntas del proceso completo de montaje de rodamientos	39
Figura 35. Componentes eléctricos que permite definir el programa.....	40
Figura 36. Ventana del comando sensor de colisión	40
Figura 37. Ventana del comando sensor de distancia	41
Figura 38. Ventana del comando sensor de posición	41
Figura 39. Ventana del comando Control de posición.....	42
Figura 40. Ventana del comando Superficie de transporte	42
Figura 41. Componentes eléctricos del proceso completo de montaje de rodamientos.....	43
Figura 42. Ventana de operación en Siemens® NX12 para iniciar o detener un movimiento.....	44
Figura 43. Ventana de visualización del Inspector de tiempo de ejecución	45
Figura 44. Editor de operaciones de la simulación del proceso de montaje de rodamientos.....	46
Figura 45. Controles de posición del movimiento de avance y contracción del alimentador	47
Figura 46. Señal de entrada del movimiento de avance del alimentador	48
Figura 47. Controles de posición del control externo.....	48
Figura 48. Definición de la señal que sujeta el comprobador de bases.....	48
Figura 49. Sensor de posición del pistón alimentador.....	49
Figura 50. Adaptador de señales del sensor a1.	49
Figura 51. Adaptador de señales de los sensores de piezas	50
Figura 52. Operaciones creadas para.....	50
Figura 53. Operaciones de activación y desactivación de la succión de la tapa	51
Figura 54. Mapeo de señales del sistema	52
Figura 55. Visualización del control en CODESYS.....	52
Figura 56. Ejemplo de etapa y transición.....	53
Figura 57. Estructura del controlador	53
Figura 58. Colocación de la tapa durante el control con CODESYS.....	54
Figura 59. Final del proceso de montaje.....	54
Figura 60. Reparto de los tiempos del proyecto.....	56
Figura 61. Configuración de las señales externas en Siemens® NX12	59
Figura 62. Variables definidas en CODESYS	60
Figura 63. Ventana de mapeo de señales en Siemens® NX12	61
Figura 64. Mapeo de todas las señales del sistema.....	62
Figura 65. Activación de las señales de retroceso en CODESYS.....	63

1. Objeto

El presente Trabajo de Fin de Máster tiene como objetivo principal la aplicación e integración de los distintos conocimientos adquiridos durante la realización del *Máster en Diseño y Fabricación Integrada Asistidos por Computador* ^[1] en la Universitat Politècnica de València, siendo la parte de Automatización Industrial el nexo entre las distintas materias.

El proyecto propuesto para ello consiste en la realización del gemelo digital de una célula de ensamblaje de rodamientos, que previamente se ha controlado de manera real en el laboratorio Tabarca de la Escuela Superior de Ingeniería del Diseño.

En la realización del gemelo digital se ha empleado el software Siemens® NX12 ^[2] para la parte de modelado 3D y CODESYS® ^[3] como entorno de programación del controlador, la conectividad entre ellos se realiza a través de un enlace que comunica el protocolo TCP/IP de NX con el protocolo MODBUS TCP de CODESYS®.

2. Alcance y exclusiones

La realización del gemelo digital se plantea inicialmente como un proceso experimental en el laboratorio Tabarca de la ETSID de la UPV, sin embargo, la restricción de acceso al mismo por la COVID-19 supone variar ligeramente el alcance y características del proyecto.

El proyecto en sí mismo puede suponer ventajas para la formación de nuevos estudiantes en materia de automatización, haciendo posibles las prácticas de esta materia de forma telemática utilizando este tipo de sistemas.

Las etapas del proyecto que se han desarrollado son las siguientes:

- Definición de gemelo digital y posibles aplicaciones presentes y futuras. Planteamiento de este método como posible solución a la creciente necesidad de trabajo y formación a distancia.
- Modelado tridimensional del proceso real del cual se pretende realizar el gemelo digital. Lo ideal es llevar a cabo esta etapa a partir de la toma de medidas de la célula real, pero por las restricciones de acceso, este paso se ha realizado de forma visual en base a la experiencia obtenida en la asignatura de Automatización y los videos obtenidos durante el curso.
- Definición del proceso de control. Definición de las entradas y salidas del sistema, desarrollo de los GRAFCET del proceso e identificación de los parámetros de control. El control del gemelo debe ser el mismo que se utiliza en el proceso real.
- Simulación del gemelo digital. Integración del programa de control con el de modelado.
- Valoración económica. Estimación de los costes que supone realizar un gemelo digital de un proceso de montaje de rodamientos.

Una de las principales dificultades del trabajo y que, por tanto, más tiempo ha requerido, ha sido el modelado tridimensional de la célula, pues las proporciones, posiciones y distancias han tenido que estimarse sin conocer ninguna medida ni existir planos del sistema.

Por último, al no tener acceso al laboratorio, queda excluida del proyecto la simulación conjunta del gemelo digital y de la célula. No obstante, sería de gran interés poder observar el funcionamiento de ambos sistemas para comprobar que es aplicable el mismo control.

3. Introducción

La creciente necesidad de aplicar nuevas tecnologías digitales ha acelerado la entrada de la llamada cuarta revolución industrial o Industria 4.0, siendo estas tecnologías la clave para la transformación digital de la sociedad. Un ejemplo de ello son los *gemelos digitales*, que según un estudio de Juniper Research, aumentarán su demanda un 17% en 2021. ^[4]

El *gemelo digital*, en su definición más amplia, se entiende como una réplica virtual de un producto, servicio o sistema real que permite experimentar con ella a través de sus sensores y mecanismos sin correr riesgos al buscar soluciones a un problema, lo que supone una importante herramienta en la toma de decisiones^[5]; es la combinación de un modelo matemático y un programa informático que representan el aspecto y comportamiento de un sistema real. ^[1]

El término fue acuñado en 2012 por la NASA, cuando pocos imaginaban el poder que iba a tener el concepto en la industria, y todavía hoy es un concepto en desarrollo con algunas barreras tecnológicas a la hora de implementarlo en la industria, debido especialmente a la dificultad técnica para monitorizar y digitalizar masivamente sus procesos. ^[6]

El gemelo digital es el modelo que resulta de integrar la información de las distintas áreas de la ingeniería (mecánica, mecatrónica, termodinámica) en el mundo virtual, dando lugar a un modelo extremadamente preciso que funciona como herramienta de comunicación entre las distintas áreas, y si se aplica correctamente, puede ayudar a: ^[7]:

- Predecir y prevenir resultados, lo que permite evitar problemas antes de que se produzcan.
- Crear mejores productos, más rápido y a escala.
- Mejorar procesos y prevenir tiempos de inactividad
- Reducir la brecha entre el diseño y la ejecución.
- Medir el rendimiento mejorado y personalizar la producción.

La prevención de fallos es una de las principales ventajas, pero sus aplicaciones pueden ser aprovechadas para reducir los costes energéticos tanto en el productor como en el consumidor final o reducir considerablemente las emisiones contaminantes. ^[8]

Además, los beneficios de un gemelo digital no se limitan al desarrollo de productos, sino que se aplican a todo el ciclo de vida del producto. Se puede utilizar para diagnosticar problemas con un sistema real o para el mantenimiento predictivo. ^[9]

En el caso de la docencia, la posibilidad de formación utilizando gemelos digitales supone una importante ventaja en cuanto a la independencia de los alumnos, ya que se elimina el riesgo de daños en los equipos por errores de inexperiencia y pueden realizarse simulaciones de forma individualizada, lo que incrementa la eficiencia en sesiones prácticas y aumenta el rendimiento. Además, debido a la creciente necesidad de formación telemática, este tipo de sistema permitiría la realización de prácticas y sesiones formativas sin necesidad de acceder al centro formativo.

El gemelo digital desarrollado en este proyecto pretende servir como herramienta para que futuros estudiantes de la asignatura de Automatización Industrial del Máster en Diseño y Fabricación Integrada Asistidos por Computador (o formaciones relacionadas) que no puedan asistir presencialmente al laboratorio, tengan posibilidad de automatizar y simular el proceso incluso desde casa. Por otra parte, se pretende reflejar cómo los distintos conocimientos que proporciona el Máster y las distintas áreas del proceso de diseño se pueden vincular.

4. Normativa aplicable

El estándar internacional IEC 61131-3 es la principal herramienta de estandarización de los lenguajes de programación en la automatización industrial ^[10], comprendiendo una parte de elementos comunes y otra de lenguajes de programación.

Un programa de automatización es una interacción de funciones y bloques funcionales que intercambian datos, que son las partes más básicas de construcción de un programa, y contienen la declaración de datos y variables y un conjunto de instrucciones.

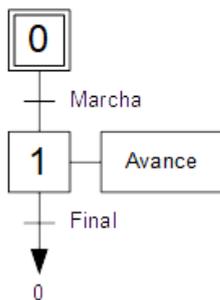


Figura 1. Ejemplo de SFC

Para describir gráficamente el comportamiento secuencial de un programa de control, el estándar emplea Gráficos Funcionales Secuenciales (SFC). Estos gráficos (Figura 1) estructuran la organización interna de un programa y lo subdividen en un conjunto de etapas y transiciones interconectadas por medio de enlaces directos. Cada etapa lleva asociados un conjunto bloques de acción y a cada transición va asociada una condición de transición que cuando se cumple, causa la desactivación de la etapa anterior a la transición y la activación de la siguiente.

Cada elemento puede ser programado en alguno de los cuatro lenguajes que recoge la norma (Figura 2), incluyendo además el propio SFC. Estos lenguajes pueden ser de tipo literal o gráfico:

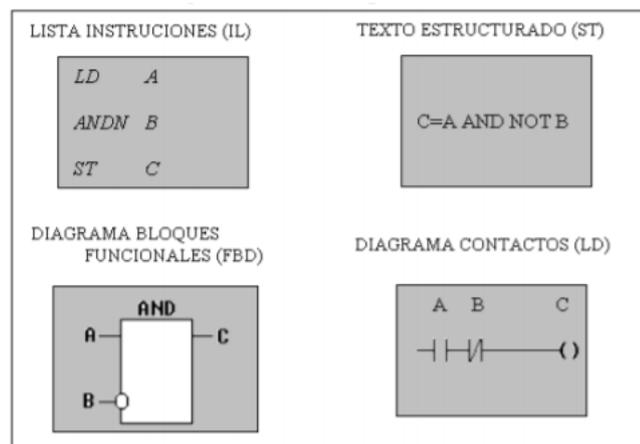
Literales:

- Lista de instrucciones (IL).
- Texto estructurado (ST).

Gráficos:

- Diagrama de contactos (LD).
- Diagrama de bloques funcionales (FBD)

Figura 2. Descripción de la misma acción en distintos lenguajes



En lo sucesivo, se han utilizado los GRAFCET o Gráficos Funcionales Secuenciales (SFC) para programar el funcionamiento del sistema de estudio, complementando las instrucciones mediante *Diagramas de contactos* (LD).

5. Modelado del sistema de ensamblaje

5.1. Introducción

El gemelo digital se basa en una maqueta automatizable de un sistema de ensamblaje de rodamientos modelo MAP-205 ^[11] del distribuidor SMC (Figura 4). Se ha realizado el modelado completo del proceso con el mayor detalle posible, pero simplificando las geometrías y componentes que no afectan a los movimientos ni la automatización. Es decir, se han modelado los mecanismos y las piezas que se manipulan, pero se han omitido la mayoría de cables, válvulas, etc. Además, el modelado se ha realizado a tamaño real aunque no exacto, pues debido a la imposibilidad de acceder al laboratorio, las dimensiones y geometrías de los sólidos se han estimado mediante fotografías y videos del proceso real realizados durante el curso.

Aunque el control en ambos sistemas es el mismo, la comunicación entre el gemelo digital y el controlador difiere ligeramente de la de la célula real (Figura 3), pues en el gemelo es necesario un enlace intermedio para conectar el software de modelado y mecatrónica con el controlador.

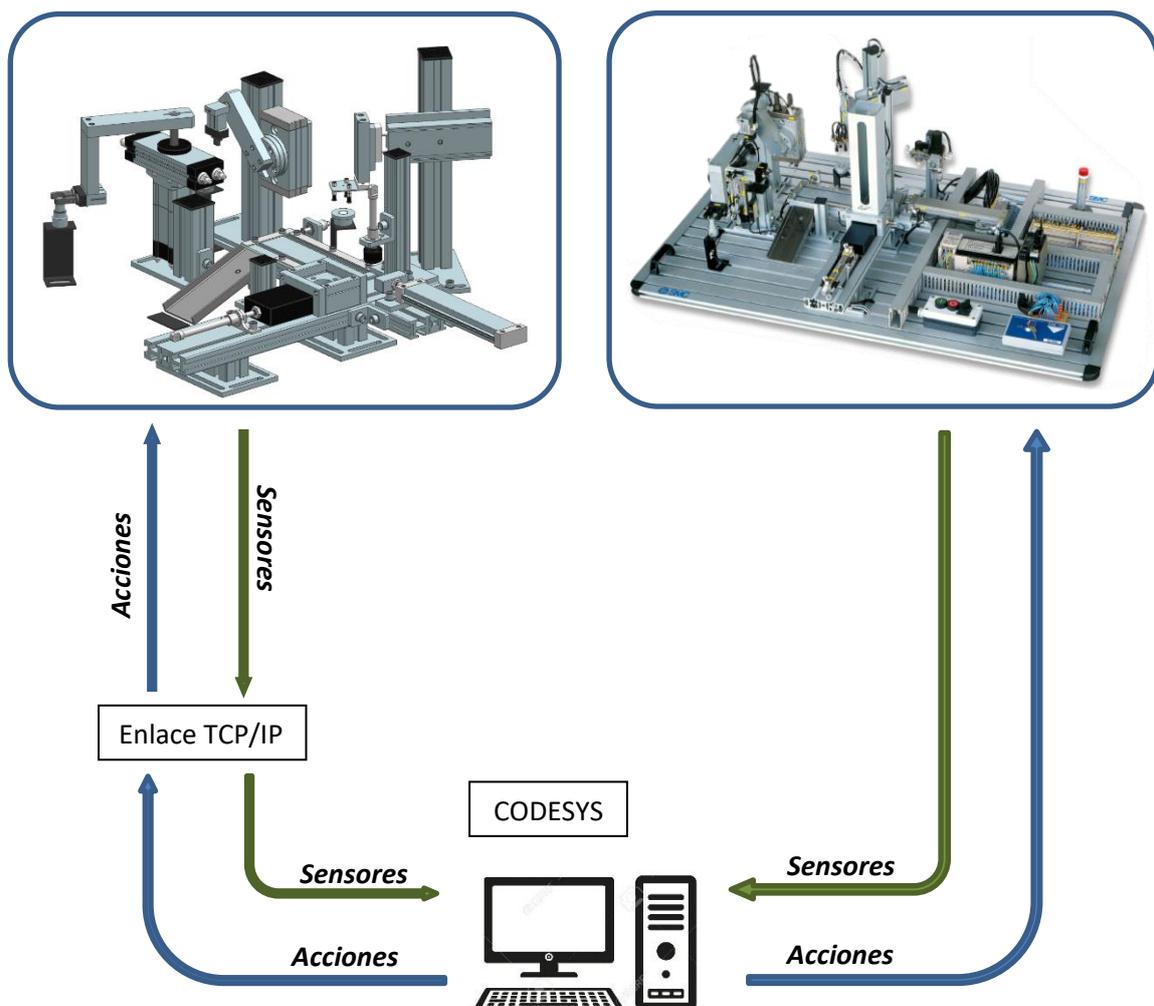


Figura 3. Esquema de comunicaciones

5.2. Descripción del sistema

El proceso consta de una etapa inicial en las que las bases de la pieza a ensamblar se introducen al sistema, a continuación un comprobador asegura que la base está en la orientación adecuada, tras esto, un pistón empuja dichas bases a la zona de montaje, donde un brazo manipulador coloca el rodamiento, luego un manipulador colca el eje del mismo y a continuación, un mecanismo con succión que coloca la tapa y finalmente, se realiza la expulsión de la pieza ya montada que cae por una superficie inclinada.

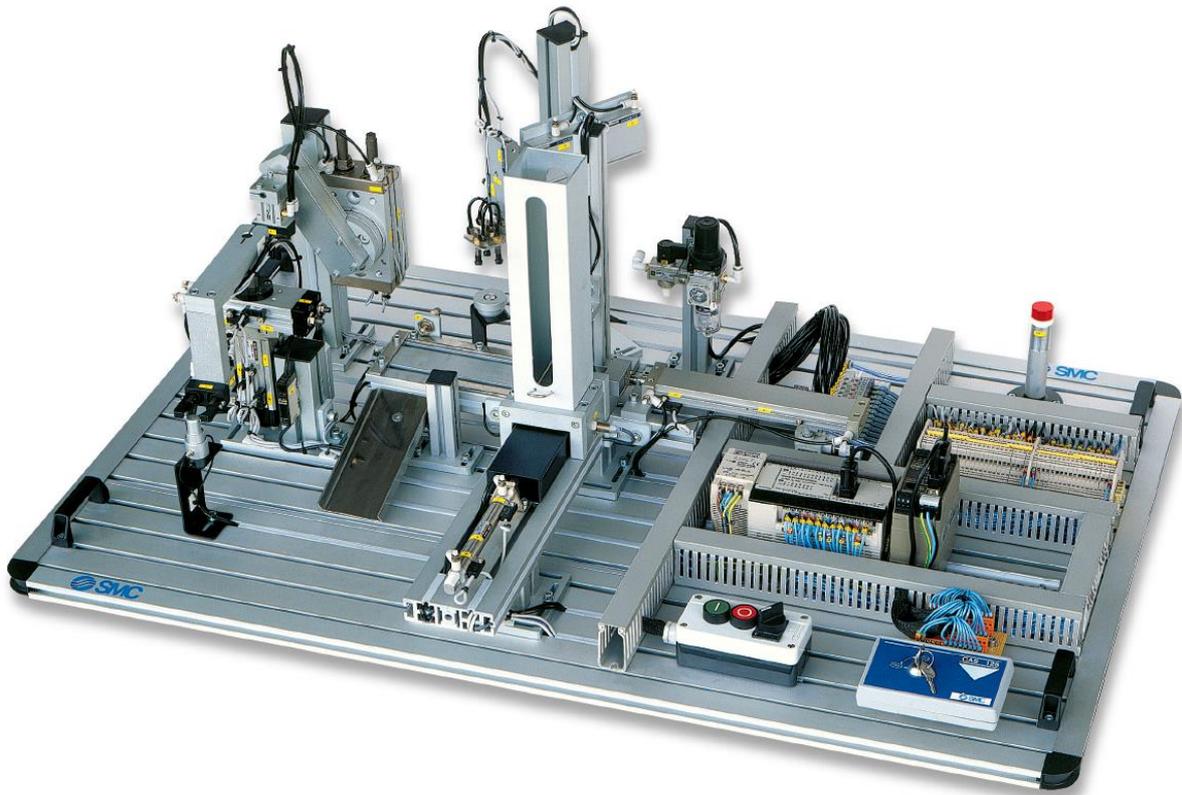


Figura 4. Célula de ensamblaje SMC MAP-205

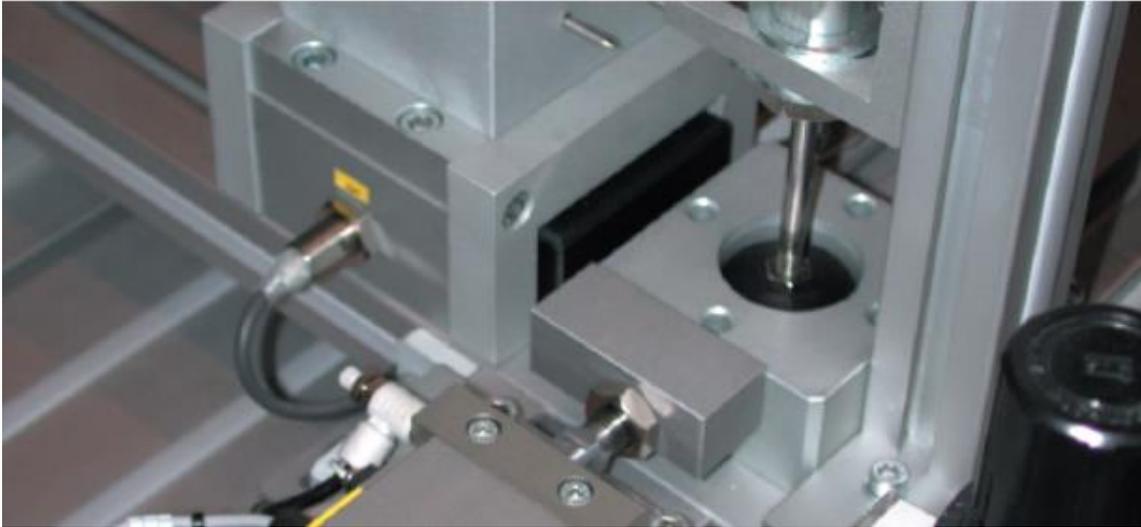


Figura 5. Estación de alimentación de bases

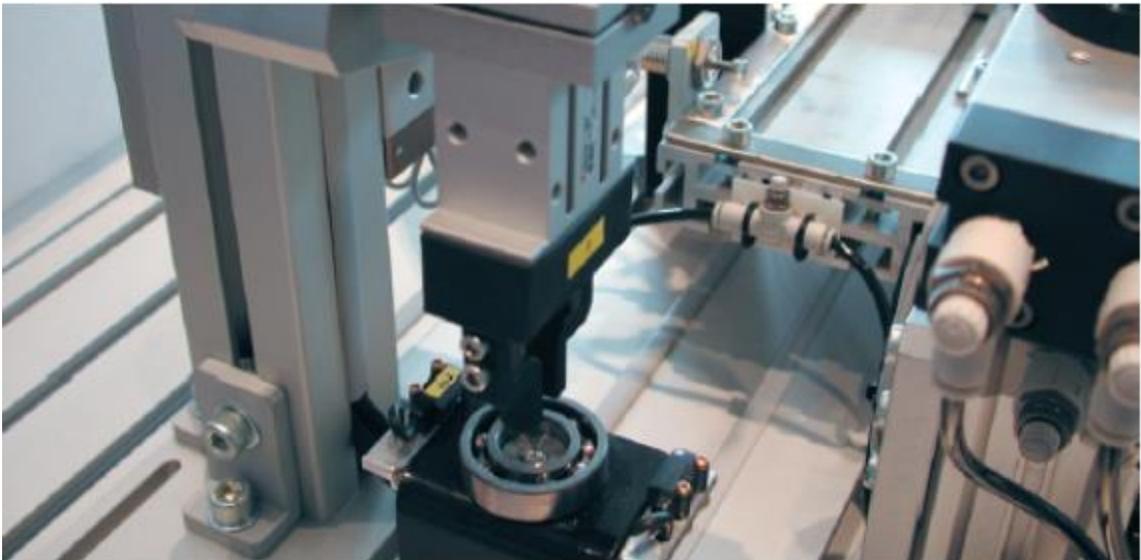


Figura 6. Estación de inserción del rodamiento



Figura 7. Estación de inserción del eje

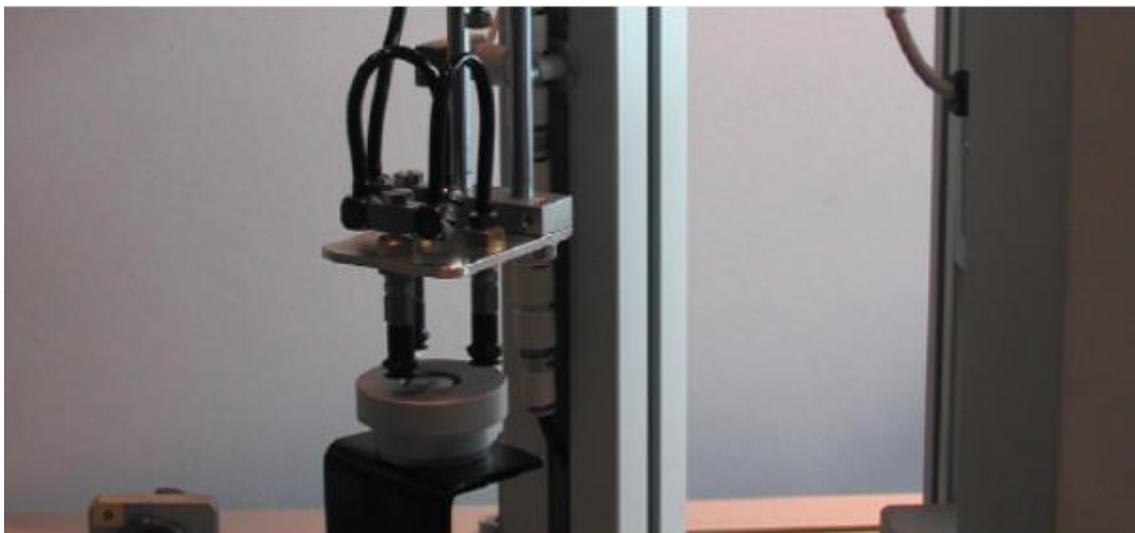


Figura 8. Estación de colocación de la tapa

5.3. Pieza a ensamblar

La pieza que se obtiene en este proceso (Figura 9) está formada por tres componentes que deben ser montados en el orden correspondiente sobre una base.



Figura 9. Pieza final obtenida en el proceso de montaje

Como ya se ha mencionado, en primer lugar se introduce la base al sistema, a continuación, se inserta el rodamiento, posteriormente se introduce en él el eje del mismo y, por último, se coloca la tapa (Figura 10).

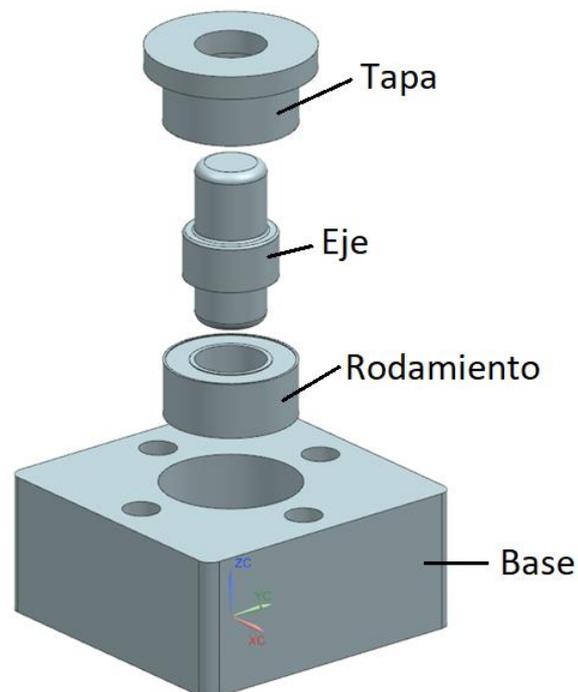


Figura 10. Componentes ensamblados durante el proceso

5.4. Alimentador de bases

El ensamblaje de la etapa inicial (Figura 11) se ha nombrado con el identificador 000, las piezas que lo componen se identifican con el número 00 (Tabla 1).

Esta etapa contiene un ensamblaje correspondiente a la parte fija por la que irán entrando las bases apiladas y la superficie por la que se desplazarán inicialmente, que a la vez sirve de soporte de la parte móvil del cilindro neumático.

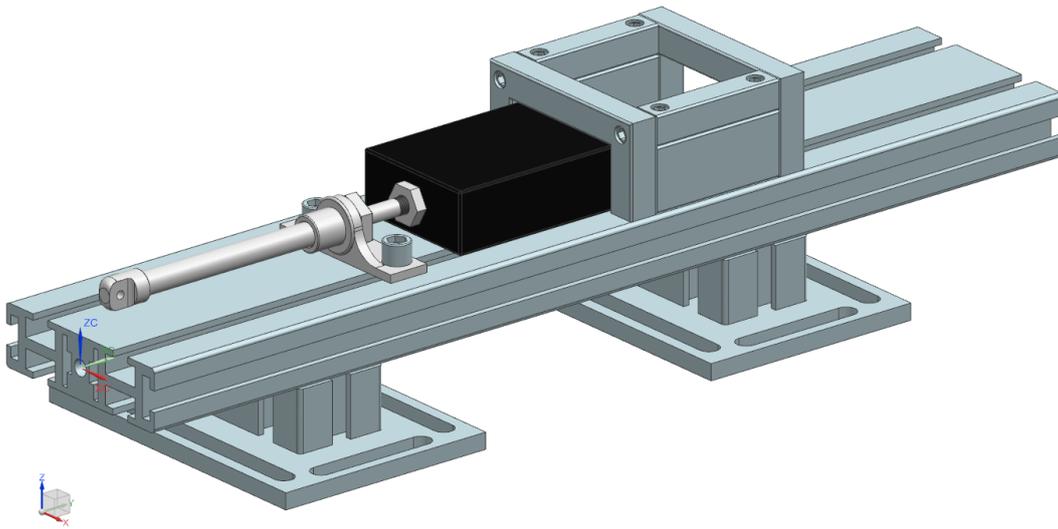


Figura 11. Ensamblaje del alimentador de bases

Tabla 1. Piezas que componen el alimentador

000_Alimentador_Soporte	Denominación de la pieza	Unidades en el ensamblaje
	00_base_alim	1
	00_marco	2
	00_marco_doble	2
	00_lateral	2
	00_tornillo8	8
	00_agarre_cilindro	1
	00_tornillo_13	2
	00_tuerca_piston	1
	00_pie_base	1

000_Alimentador_Piston	Denominación de la pieza	Unidades en el ensamblaje
	00_eje_alim	1
	00_tuerca_alim	1
	00_prisma_alim	1

5.5. Comprobador de orientación

La segunda etapa se ha identificado con el identificador 002, las piezas que componen este ensamblaje se identifican con el número 02 (Tabla 2).

Esta etapa (Figura 12) consta de un ensamblaje correspondiente a la parte fija que sirve de sujeción al pistón con comprobará que las bases están colocadas con la abertura en la parte superior, y una parte móvil, que es el propio pistón.

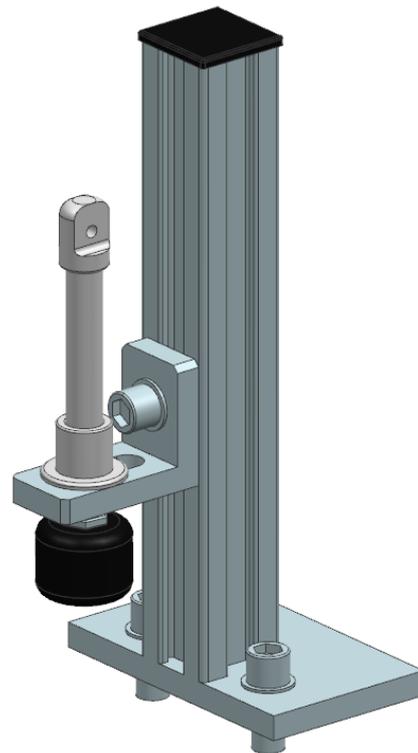
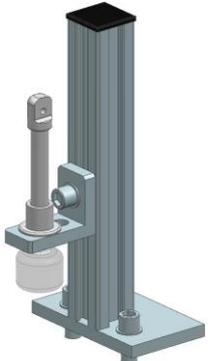


Figura 12. Ensamblaje del comprobador de orientación

Tabla 2. Piezas que componen el comprobador de orientación

002_Comprobador_Soporte	Denominación de la pieza	Unidades en el ensamblaje
	02_soporte_comprobador	1
	02_tapa_comprobador	1
	02_agarre_comprobador	1
	02_cilindro_compr	1
	02_tornillo13	3
	02_arandela	3

002_Comprobador_Piston	Denominación de la pieza	Unidades en el ensamblaje
	02_eje_compr	1
	02_tope_piston	1
	02_arandela_piston	1

5.6. Empujador a zona de montaje

La siguiente etapa (Figura 13) se ha identificado con el identificador 001, las piezas que componen este ensamblaje se identifican con el número 01 (Tabla 3).

Esta etapa consta de un ensamblaje correspondiente a la parte fija que sirve de soporte y sujeción al pistón que empujará las bases hacia la zona de montaje y el propio pistón.

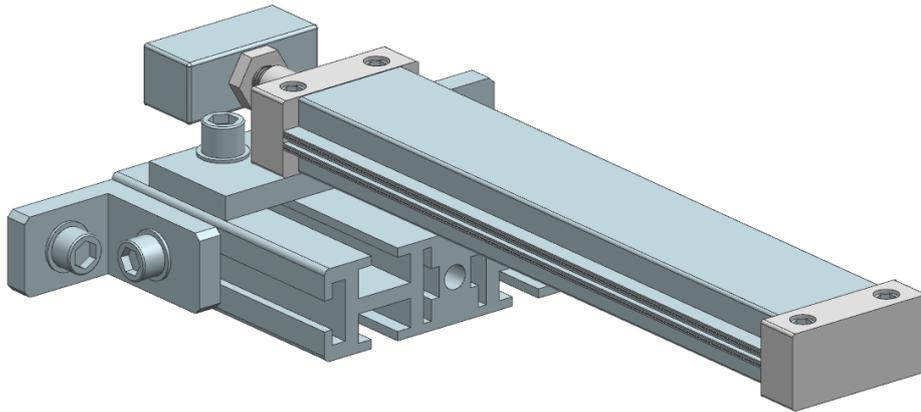


Figura 13. Ensamblaje del empujador

Tabla 3. Piezas que componen el empujador

001_Empujador_Soporte		Denominación de la pieza	Unidades en el ensamblaje
		<i>01_base_emp</i>	1
		<i>01_agarre</i>	2
		<i>01_tapa_base_emp</i>	1
		<i>01_cilindro</i>	1
		<i>01_tornillo8</i>	4
		<i>01_tornillo13</i>	2
		<i>01_arandela</i>	6
		<i>01_tornillo13_corto</i>	4
001_Comprobador_Piston		Denominación de la pieza	Unidades en el ensamblaje
		<i>01_eje_empujador</i>	1
		<i>01_tuerca</i>	1
		<i>01_prisma_empuj</i>	1

5.7. Superficie de desplazamiento y zona de montaje

Cuando el pistón empuja la pieza correspondiente a la base, ésta se desliza sobre una superficie hasta que alcanza el final, lo que se considera la zona de montaje donde el resto de los mecanismos colocarán las piezas correspondientes, posteriormente un mecanismo la empujará hacia una rampa una vez concluye el proceso.

Este ensamblaje (Figura 14) se identifica con 003 y las piezas que lo componen se identifican con el número 03 (Tabla 4).

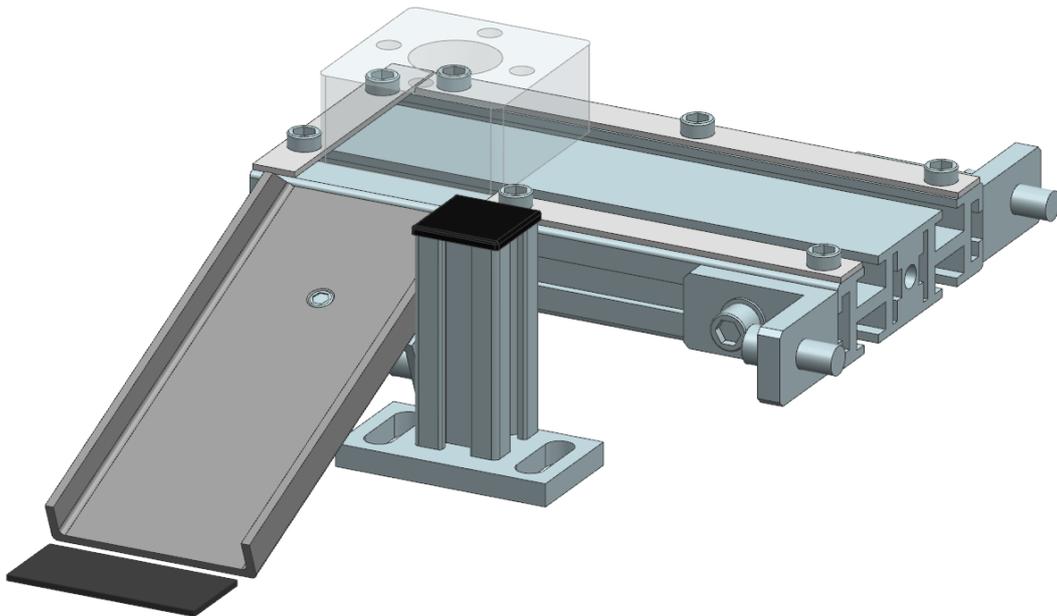


Figura 14. Ensamblaje de la zona de montaje

Tabla 4. Piezas que componen la zona de montaje y superficie de desplazamiento

003_Cinta_recorrido	Denominación de la pieza	Unidades en el ensamblaje
	03_cinta	1
	03_agarre_cinta	2
	03_tapa	1
	03_rampa	1
	03_final	1
	03_agarre_rampa	1
	03_tornillo8	1
	03_tornillo13_corto	1
	03_tope_finalrecorrido	1
	03_tornillo11_guia	2
	03_tornillo13	4
	03_arandela	5
	03_tope_trasero	1
	03_tornillo11_guia_corto	5
	03_tope_delantero	1

5.8. Montaje del rodamiento

Una vez que la base se encuentra en la zona de montaje, la primera pieza que se monta es el rodamiento. Para ello se emplea un actuador de giro con dos articulaciones y con una pinza de dos dedos (Figura 15), que serán los ensamblajes móviles; el ensamblaje correspondiente a la parte fija será la base del mecanismo.

La etapa se diferenciado con el identificador 004 y, por tanto, las piezas individuales que componen estos ensamblajes se identifican con 04 (Tabla 5).

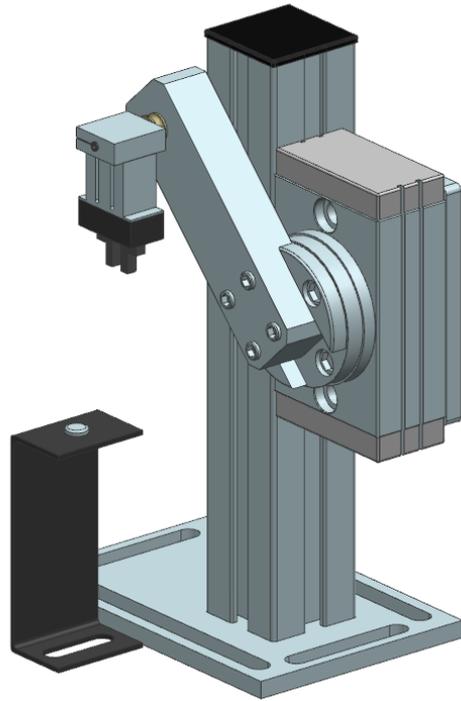
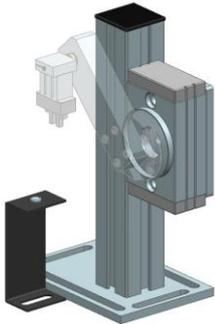
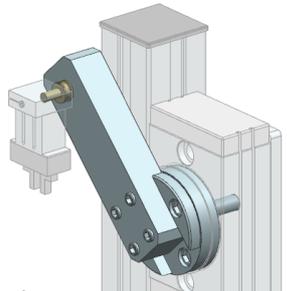
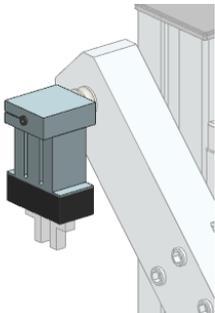
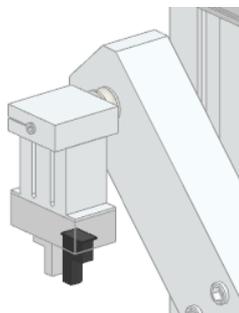


Figura 15. Ensamblaje del actuador que coloca el rodamiento

Tabla 5. Piezas que componen el manipulador del rodamiento

004_Rodamiento_soporte	Denominación de la pieza	Unidades en el ensamblaje
	04_soporte_brazo_rodamiento	1
	04_tapa_soporte_rodamiento	1
	04_motor_rodamientos	1
	04_tapa_trasera_motor	1
	04_apoyo_rodamiento	1

004_Rodamiento_Brazo	004_Rodamiento_Brazo2	004_Rodamiento_Pinza
----------------------	-----------------------	----------------------

					
Denominación	Unidades	Denominación	Unidades	Denominación	Unidades
04_brazo1	1	04_brazo2	1	04_pinza	2
04_tornillo13_corto	3	04_brazo_inferior	1		
04_eje_Brazo1	1				
04_tornillo8	4				

5.9. Montaje del eje

Tras colocar el rodamiento, se inserta el eje del mismo. Este proceso se realiza mediante un cilindro de movimiento lineal vertical y rotativo (Figura 16), con una pinza en su extremo; la base del mecanismo también será el ensamblaje fijo en este caso. Las piezas correspondientes a esta etapa se numeran con 005 (Tabla 6).

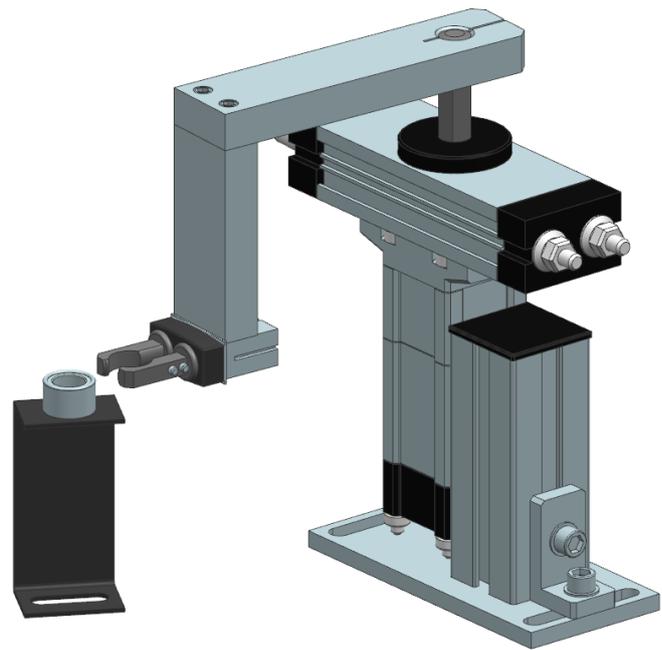
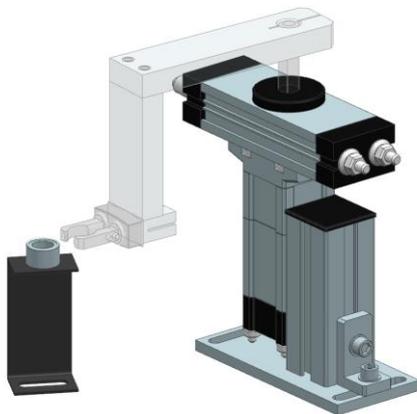


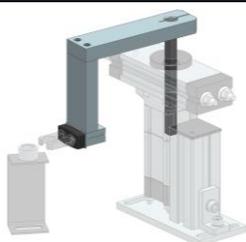
Figura 16. Ensamblaje del actuador que coloca el eje

Tabla 6. Piezas que componen el manipulador de ejes

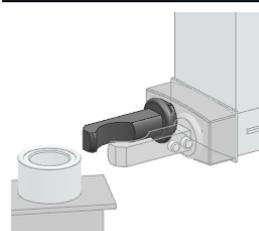
005_Eje_soporte	Denominación de la pieza	Unidades en el ensamblaje
	05_soporte_brajo_eje	1
	05_tapa_soporte_eje	1
	05_soporte_vertical	1
	05_soporte_vertical_corto	1
	05_soporte_vertical_cortoT	1
	05_tornillo13_largo	4
	05_soporte_vertical_inferior	1
	05_tuerca	4
	05_soporte_horizontal	1
	05_agarre	1
	05_tornillo13_corto	2
	05_arandela	2
	05_tuerca_horizontal	2
	05_barra_pasante	2
	05_arandela_horizontal	2
	05_embellecedor_agujero	1
	05_apoyo_eje	1



005_Eje_brazo	Denominación	Unidades
	05_brazo1	1
	05_tornillo8	2
	05_brazo2	1
	05_ejebrazo	1
	05_brazo2_pinza	1
	05_brazo2_inferior	1



005_Pinza_eje	Denominación	Unidades
	05_pinza_eje	2



5.10. Colocación de la tapa

La última pieza que se monta es la tapa, identificada como 006 y sus componentes con 06 (Tabla 7).

Sobre la parte fija o soporte de este mecanismo (Figura 17), se desliza un cilindro horizontalmente que permite desplazar desde la posición inicial a la zona de montaje la segunda articulación o cilindro, y éste permite el movimiento vertical para coger y soltar la tapa.

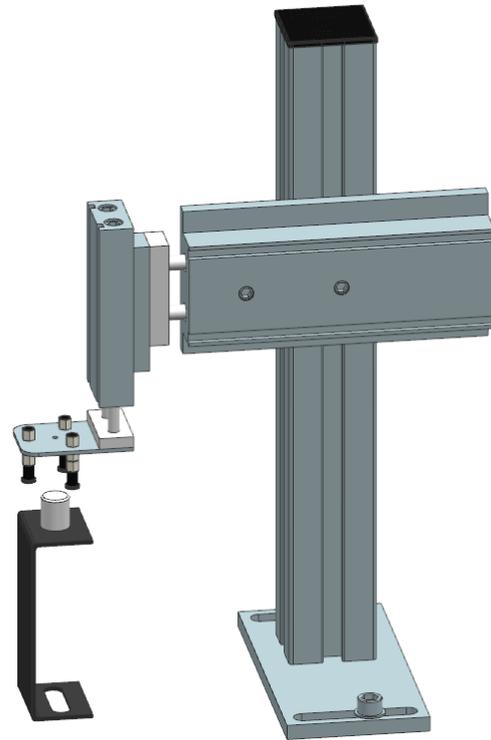


Figura 17. Ensamblaje del actuador que coloca la tapa

Tabla 7. Piezas que componen el manipulador de la tapa

006_Tapa_Soporte	Denominación de la pieza	Unidades en el ensamblaje
	06_soporte_brazp_tapa	1
	06_tapa_soporte_tapa	1
	06_arandela	2
	06_ornillo13_corto	2
	06_cilindro1	1
	06_tornillo8	2
	06_apoyo_tapa	1

006_Tapa_Piston1	Denominación	Unidades
	06_pisto1_eje	1
	06_prisma_piston	1
	06_cilintro	1
	06_tornillo10	2

006_Tapa_Piston2	Denominación	Unidades
	06_piston2_eje	1
	06_base_succion	1
	06_ventosa	3
	06_tuerca	9

5.11. Expulsión de la pieza

La última etapa es la expulsión de la pieza del sistema, para ello se emplea un cilindro de menor tamaño (Figura 18) que la empujará de la zona de montaje hacia la rampa por la que deslizará.

Esta última etapa se ha identificado con 007 y sus componentes con 07 (Tabla 8). Consta también de una parte fija y el pistón móvil.

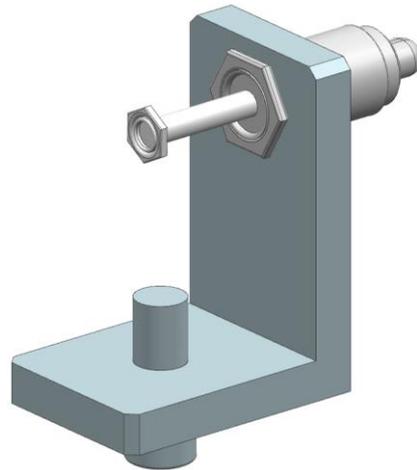
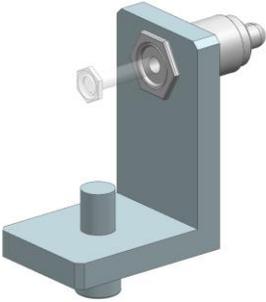


Figura 18. Ensamblaje del pistón de expulsión

Tabla 8. Piezas que componen el pistón de expulsión de piezas

007_Expulsión_Soporte	Denominación de la pieza	Unidades en el ensamblaje
	07_agarre_empujador	1
	07_cilindro_exp	1
	07_arandela_piston	1
	07_arandela	1
	07_tornillo13	1
000_Alimentador_Piston	Denominación de la pieza	Unidades en el ensamblaje
	07_piston	1
	07_tuerca_piston	1

5.12. Modelo completo

Todos los ensamblajes correspondientes a partes fijas se han agregado a su vez a un ensamblaje que se restringirá por completo, mientras que lo móviles serán independientes y sus restricciones se basarán en los desplazamientos que deben realizar para la correcta colocación de las piezas correspondientes (Figura 19).

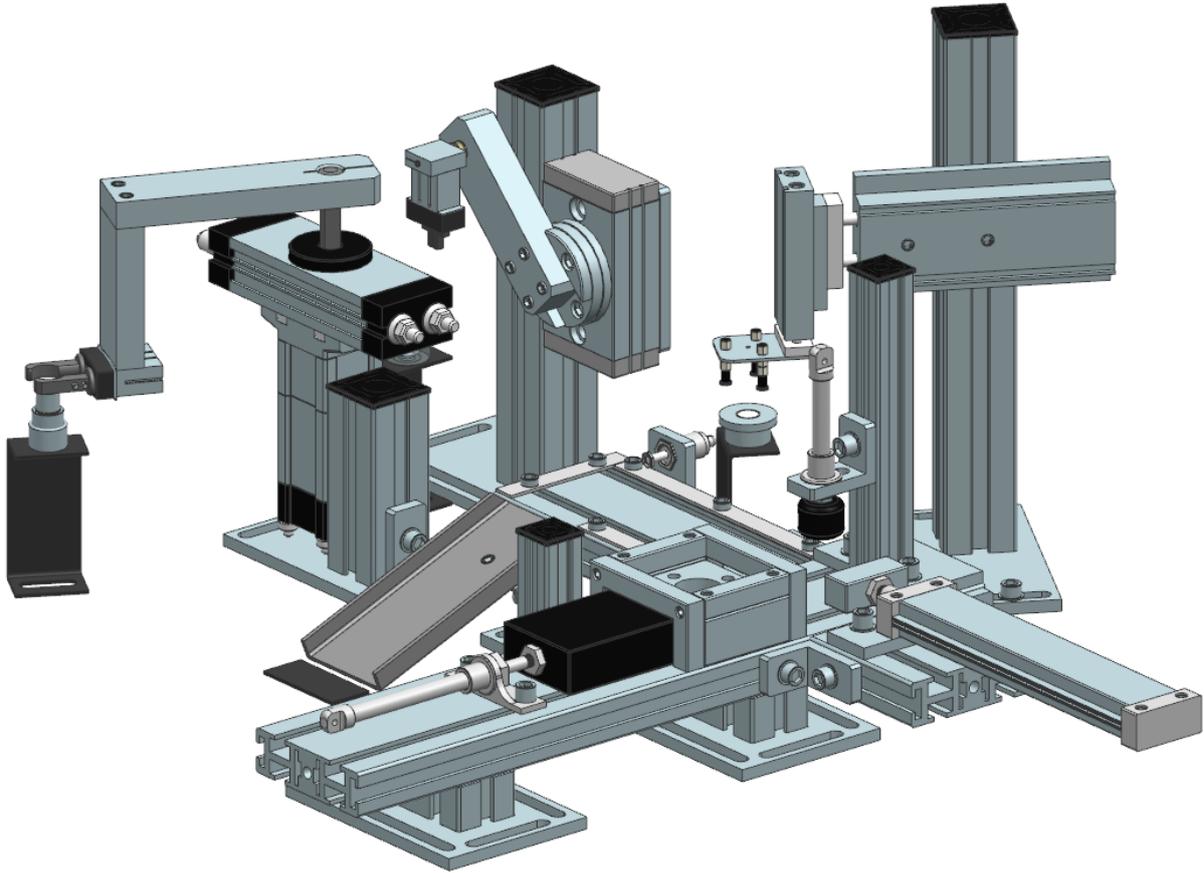


Figura 19. Ensamblaje del proceso completo

El sistema y los diferentes accionamientos que lo forman tienen un total de 179 piezas como se observa en la Tabla 9.

Tabla 9. Resumen del total de piezas que componen el proceso

Etapa	Número de piezas
<i>Alimentador de bases</i>	23
<i>Comprobador de posición</i>	13
<i>Superficies de deslizamiento y zona de montaje</i>	28
<i>Empujador a zona de montaje</i>	24
<i>Montaje del rodamiento</i>	18
<i>Montaje del eje</i>	37
<i>Colocación de la tapa</i>	29
<i>Expulsión</i>	7
PIEZAS TOTALES	179

6. Automatización del proceso

Una vez modelados todos los componentes del sistema de ensamblaje, se procede a simular su movimiento y realizar su automatización. En primer lugar, se definen los movimientos en Siemens® NX12, enlazados entre ellos y condicionados a los sensores que definirán las posiciones máximas y mínimas.

Tras lograr simular el movimiento, se automatizará vinculando CODESYS con Siemens® NX12 a través de un servidor TCP/IP.

6.1. Descripción del proceso

Como ya se ha descrito, el proceso completo consta de varias etapas o estaciones que, aunque realizan una función diferente, están condicionadas por los comienzos y finales de otras etapas. Por ello, a la hora de realizar los GRAFCET del sistema completo, se han agrupado por subprocesos que en algunos casos engloban más de una de las estaciones antes descritas.

Además de los sólidos que se han modelado, es importante conocer los actuadores y sensores del sistema, es decir, las entradas y salidas. Éstas son las señales que utilizará el controlador una vez se haya programado ^[11], recogidas en la Tabla 10 y la Tabla 11.

Tabla 10. Descripción de las entradas del sistema

Entrada	Descripción	Dirección	Proceso
Marcha	<i>Pulsador que inicia el sistema</i>		Inicio
a1	<i>Sensor de fin de recorrido del cilindro A</i>	%IX0.1	Alimentador y comprobador
b1	<i>Sensor de fin de recorrido del cilindro B</i>	%IX0.0	
c1	<i>Sensor de pieza correcta</i>	%IX0.2	
dpr1	<i>Sensor de base en posición de inicio</i>	%IX2.0	
j0	<i>Sensor del manipulador de tapas en posición retraída</i>	%IX1.3	Tapa y expulsión
j1	<i>Sensor del manipulador de tapas en zona de montaje</i>	%IX1.3	
k0	<i>Sensor del manipulador de tapas en posición elevada</i>	%IX1.1	
k1	<i>Sensor del manipulador de tapas en posición inferior</i>	%IX1.0	
e0	<i>Sensor de posición del manipulador E a la izquierda</i>	%IX0.3	Rodamiento
e1	<i>Sensor de posición de inicio del manipulador E</i>	%IX1.4	
e2	<i>Sensor de posición del manipulador E en zona montaje</i>	%IX1.5	
dpr2	<i>Sensor de rodamiento disponible</i>	%IX2.1	
g0	<i>Sensor de posición superior del manipulador de ejes</i>	%IX0.6	Eje del rodamiento
g1	<i>Sensor de posición inferior del manipulador de ejes</i>	%IX0.4	
h0	<i>Sensor de posición del manipulador de ejes (P) a la izq</i>	%IX0.7	
h1	<i>Sensor de posición del manipulador en zona montaje</i>	%IX0.5	
i1	<i>Sensor de pinzas en posición de cierre</i>	%IX1.6	
dpr3	<i>Sensor de eje disponible para montar</i>	%IX2.2	

Los sensores dpr2 y dpr3 toman el valor de 0 cuando detectan pieza y 1 cuando no detectan nada. Todos los demás sensores toman el valor 1 cuando detectan pieza o manipulador en la posición determinada.

Tabla 11. Descripción de las salidas del sistema

Salida	Descripción	Dirección	Proceso
A+	Extracción del cilindro alimentador A	%QX0.2	Alimentador y comprobador
B+	Activación del cilindro B a zona de montaje	%QX0.0	
C+	Extracción del cilindro comprobador C	%QX0.4	
D+	Extracción del cilindro expulsor D	%QX2.1	Tapa y expulsión
J+	Avance el manipulador de tapas a zona de montaje	%QX1.6	
J-	Retracción del manipulador de tapas	%QX1.7	
K+	Descenso del manipulador de tapa	%QX1.4	
V+	Succión de tapa	%QX2.7	Rodamiento
E+	Movimiento del manipulador E a la derecha	%QX2.0	
E-	Movimiento del manipulador E a la izquierda	%QX0.6	
F+	Apertura de pinzas	%QX2.3	Eje del rodamiento
H+	Giro del manipulador de ejes a la zona de montaje	%QX1.2	
I+	Cierre de pinzas	%QX2.5	
P+	Descenso del manipulador P	%QX1.0	

Excepto los actuadores J y E, todos los cilindros y manipuladores son monoestables, es decir, la retracción o vuelta a posición de inicio de ellos se produce de forma automática al deja de activarlos.

El diseño del control del proceso basa en la normativa IEC 61131-3 para Controladores Lógicos Programables (PLC), que emplea gráficos funcionales secuenciales (SFC) para dividir un proceso o subproceso en etapas en las que se realizan las acciones asociadas y transiciones que activan la etapa siguiente cuando se cumple la condición que se defina en ella. (ver Apartado 4).

6.1.1. Inicio del proceso

El sistema se inicia tras la pulsación del botón de marcha (M), tras lo que arranca el subsistema de alimentación de bases.

Permanecerá activado hasta que se haya activado el último accionamiento.

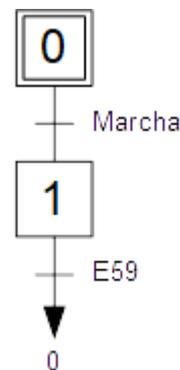
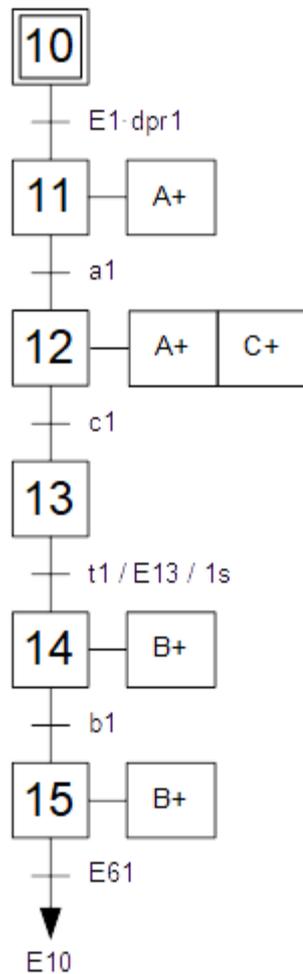


Figura 20. GRAFCET de arranque del sistema

6.1.2. Alimentación de bases y comprobador de posición



En el alimentador estarán apiladas las bases que caerán a la posición de inicio cuando la pieza anterior haya salido del alimentador.

Cuando el sistema se inicie y se detecte una pieza (dpr1), se extrae el cilindro A (A+) hasta que se active el sensor de fin de recorrido (a1).

En esta posición, manteniendo activado el cilindro A, se extrae el cilindro C (C+) hasta activar el sensor c1 si la pieza está en posición correcta. Tras esto, se espera un segundo por seguridad para que el cilindro C se retraiga y se activa el cilindro B (B+) hasta activar b1, que permanecerá activado hasta que finalice el montaje.

Figura 21. GRAFCET de la alimentación de bases

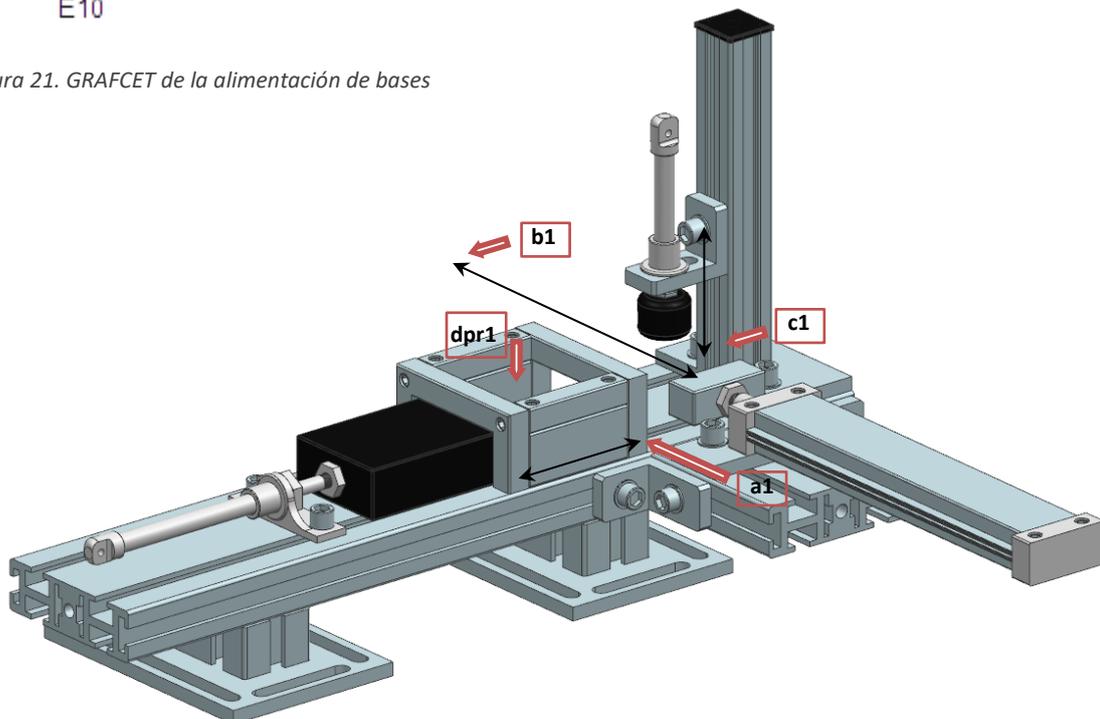


Figura 22. Ubicación de los sensores del primer subproceso

6.1.3. Montaje del rodamiento

El funcionamiento de este subproceso comenzará cuando haya finalizado el anterior y la base haya llegado a la posición de montaje.

Si existe un rodamiento disponible (not dpr2) y el manipulador está en la posición de inicio (e1), éste se desplazará a la izquierda (E-) hasta llegar a la posición del rodamiento (e0), tras esto se esperarán 0,5 segundos en esta posición para que se estabilice.

A continuación, se abre la pinza que sujeta el rodamiento (F+) durante 0,5 segundos y se desplaza el manipulador a la derecha (E+) hasta la posición de montaje (e2). Se mantiene esta posición durante 0.5 segundos para estabilizarlo y después se suelta el rodamiento (dejar de activar F+). Por último, se mueve el manipulador a la izquierda (E-) hasta la posición de inicio (e1).

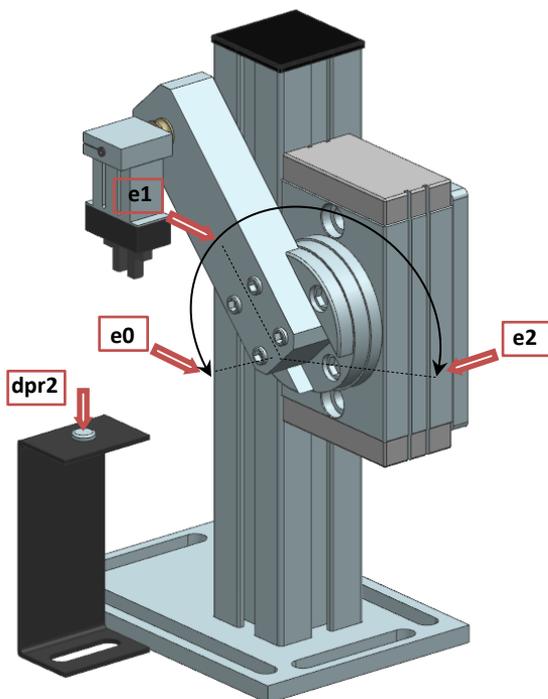


Figura 24. Ubicación de los sensores del segundo subproceso

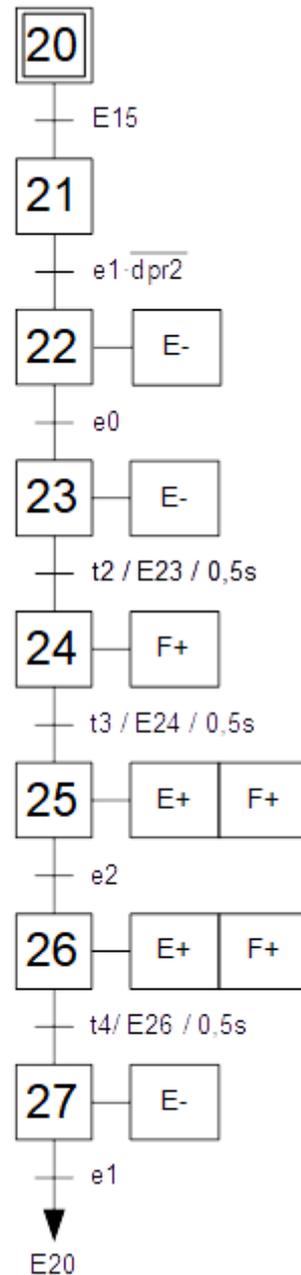


Figura 23. GRAFCET del montaje del rodamiento

6.1.4. Montaje del eje del eje

Cuando el subproceso de montaje del rodamiento haya finalizado, si hay un eje preparado y el manipulado se encuentra arriba (g0) y a la izquierda (h0), comenzará el montaje del eje.

En primer lugar desciende el manipulador (P+) hasta activar g1, a continuación se cierran las pinzas (I+) hasta activarse i1 y se esperan 0.5 segundos por seguridad.

Manteniendo activadas las pinzas, se sube el manipulador (desactivar P+) hasta la posición superior (g0), se gira (H+) hasta activar h1 y se desciende (P+) hasta la posición de montaje (h1), se espera un segundo y se desactiva P+ hasta g0 y se devuelve a la posición h0.

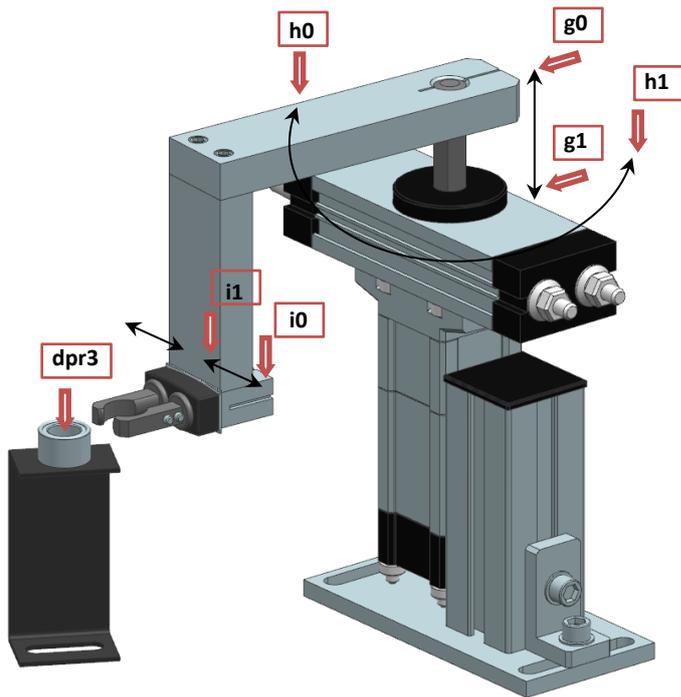
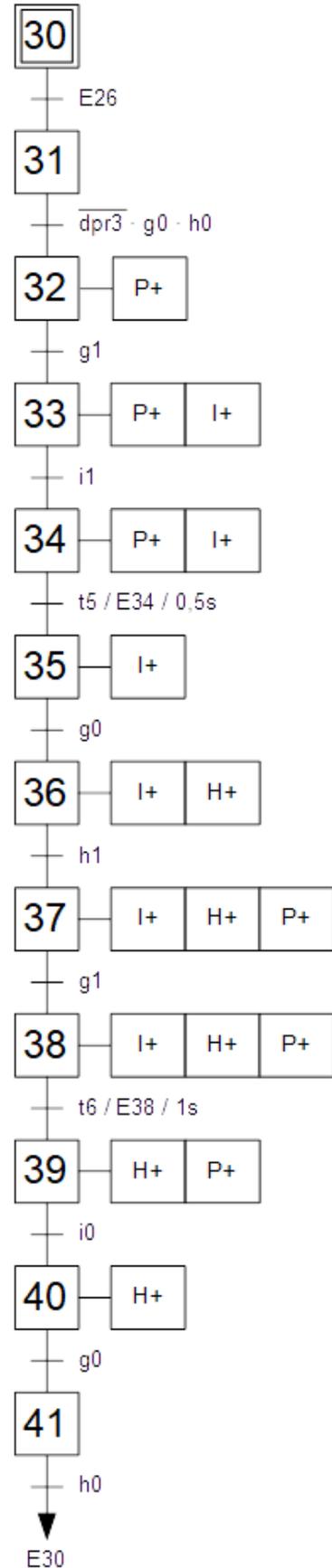


Figura 25. Ubicación de los sensores del tercer subproceso



6.1.5. Colocación de la tapa y expulsión

El último subproceso se iniciará cuando se haya colocado el eje siempre que el manipulador esté en la posición superior (k0) y se encuentre retraído (j0). Si esto se cumple, bajará el manipulador (K+) hasta activar k1, tras lo que se activa la succión (V+) durante 0.5 segundos.

Tras este tiempo y sin desactivar la succión, se eleva el manipulador (desactivando K+) hasta que se alcance k0 y a continuación se activa el avance (J+) hacia la zona de montaje (j1).

Con el manipulador sobre la base, se vuelve a bajar (K+) hasta k1 y se desactiva la succión de la tapa, tras esperar 0.5 seg se vuelve a elevar el manipulador y se retrae (J-) hasta j0.

Por último, se activa D+ durante un segundo para asegurar que expulsa la base, se espera otro segundo y se da por finalizado el proceso.

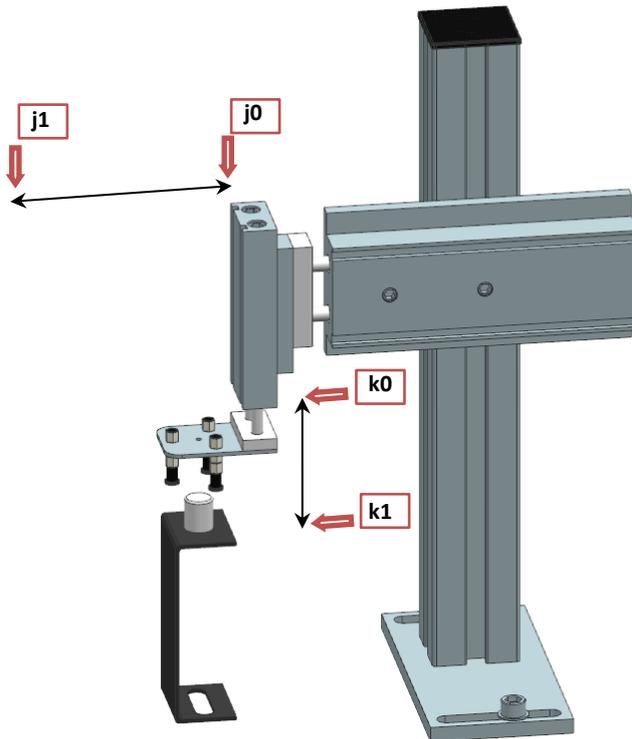


Figura 27. Ubicación de los sensores del último subproceso

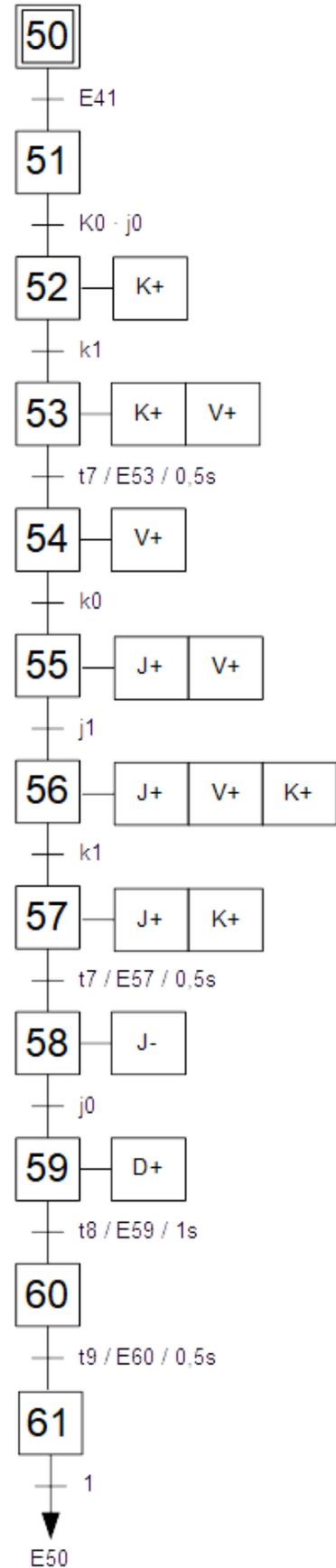


Figura 28. GRAFCET del montaje de tapa y expulsión

6.2. Definición de parámetros y componentes

Conociendo todos los componentes que forman cada subproceso y el funcionamiento y movimiento de estos, así como sus relaciones entre ellos, el siguiente paso es la definición de éstos en el módulo *Diseñador de Conceptos de Mecatrónica* de Siemens® NX12 como inicio de la preparación de la simulación.

Para definir cada movimiento, se debe partir de la definición de la física básica, los distintos tipos de juntas y restricciones que condicionarán los grados de libertad de cada actuador y los sensores correspondientes.

6.2.1. Componentes mecánicos

El módulo de NX utilizado permite distinguir entre distintos tipos de cuerpos y juntas que se consideran mecánicos (**¡Error! No se encuentra el origen de la referencia.**). En el caso de estudio, sólo se van a emplear los dos primeros tipos de la primera opción (cuerpos): *Cuerpo rígido* y *Fuente de objetos*; de la segunda (colisión): *Cuerpo de colisión* y *Superficie de transporte* y en cuanto a las juntas, se combinarán distintos tipos según el movimiento de los cuerpos.

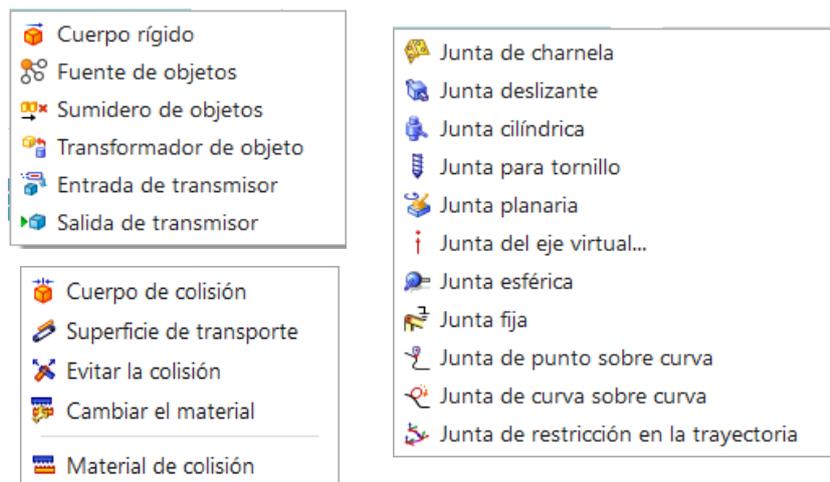


Figura 29. Componentes mecánicos que permite definir el programa

- **Cuerpo rígido:** Cada pieza o conjunto de piezas sin movimiento relativo entre ellas se definirán como un *cuerpo rígido*. Éste es el elemento principal en el modelo virtual, pues se considerarán como cuerpos rígidos todos los componentes cuya masa se quiera tener en cuenta en la simulación. Es imprescindible esta asignación para definir el movimiento de cualquier cuerpo.
En el caso de estudio, se ha considerado un mismo cuerpo rígido el ensamblaje formado por todas las partes fijas y, por otra parte, cada pieza a ensamblar o manipulador con movimiento independiente es un cuerpo rígido independiente.
- **Fuente de objetos:** Este comando se utiliza para controlar el flujo de piezas que entran al proceso, siendo en este caso las bases, que se considera que estarán apiladas en la zona de alimentación. El comando se encarga de generar una fuente de objetos en la ubicación del *cuerpo rígido* seleccionado, tantas veces como se indique y con sus mismas características cuando se cumpla una condición establecida.

- **Cuerpo de colisión:** Las superficies de los cuerpos rígidos que puedan chocar o vayan a estar en contacto con otro cuerpo, deben definirse como *cuerpos de colisión*. Este comando permite que se definan los límites de los cuerpos para que no se superponga ni atraviesen los cuerpos.

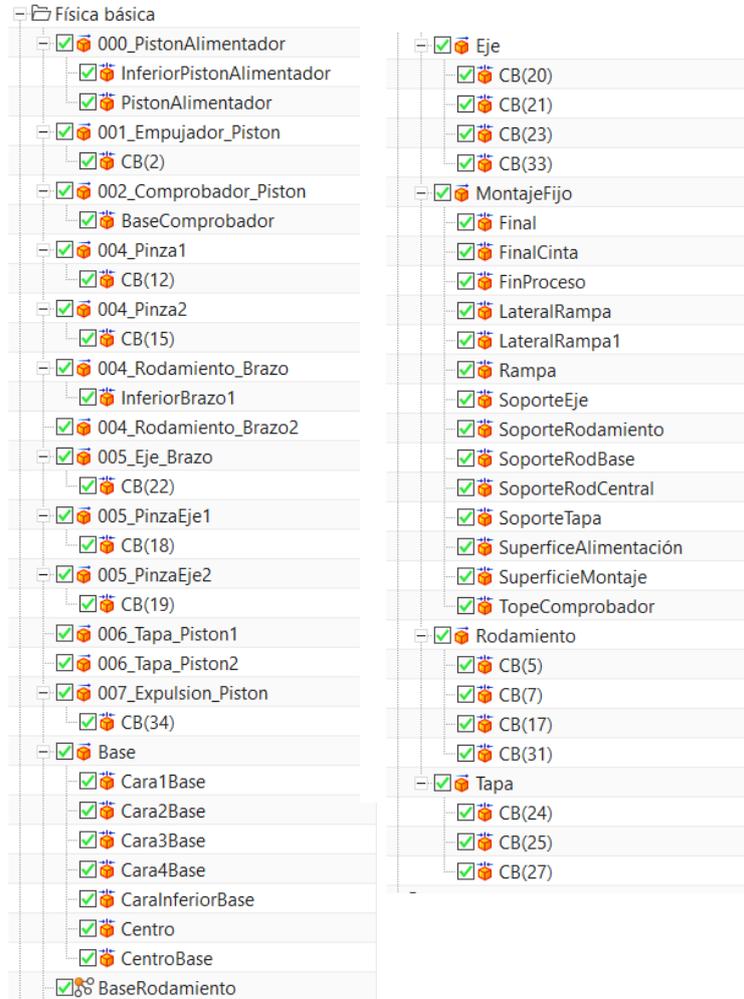


Figura 30. Física básica del proceso completo de montaje de rodamientos

Los distintos tipos de juntas permiten asociar o restringir los grados de libertad entre los *cuerpos rígidos* una vez se han definido. Es importante destacar que algunos movimientos se han simulado empleando una combinación de juntas para simplificar la complejidad de la simulación, aunque el modelo real no sea exactamente ese tipo de junta; también hay que considerar que al inicio y final de algunos movimientos se han tenido que añadir juntas fijas en los componentes para que la masa simulada de los mismos no afecte y se mantengan en su posición.

- **Junta fija:** Es el tipo de junta más simple (Figura 31). Suprime todos los grados de libertad de un *cuerpo fijo* respecto a otro o respecto al origen de coordenadas.

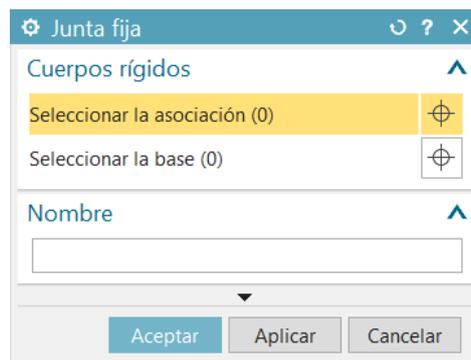


Figura 31. Ventana del comando junta fija

- **Junta de charnela:** este comando conecta los objetos a lo largo de un eje y restringe todos los grados de libertad excepto uno de rotación. Requiere definir un vector al que se asociará el movimiento de rotación y un punto de anclaje (Figura 32), que puede ser respecto a un cuerpo fijo o respecto a otro cuerpo que tenga movimiento.

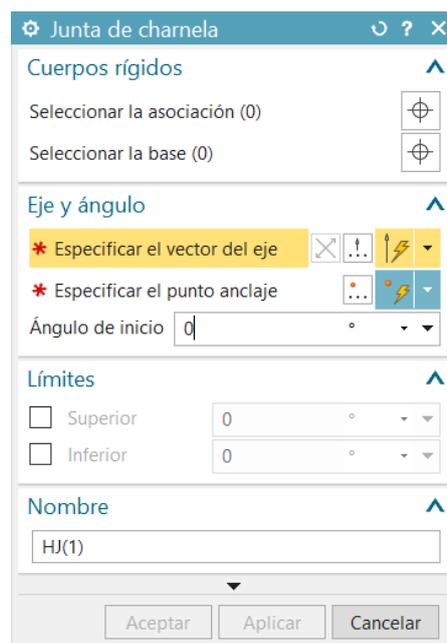


Figura 32. Ventana del comando junta de charnela

- **Junta deslizante:** conecta los objetos a lo largo de un eje lineal fijo, de modo que restringe todos los grados de libertad excepto uno de traslación. Requiere definir un vector al que se asociará el movimiento de traslación (Figura 33) que puede ser respecto a un cuerpo fijo o respecto a otro cuerpo que tenga movimiento.

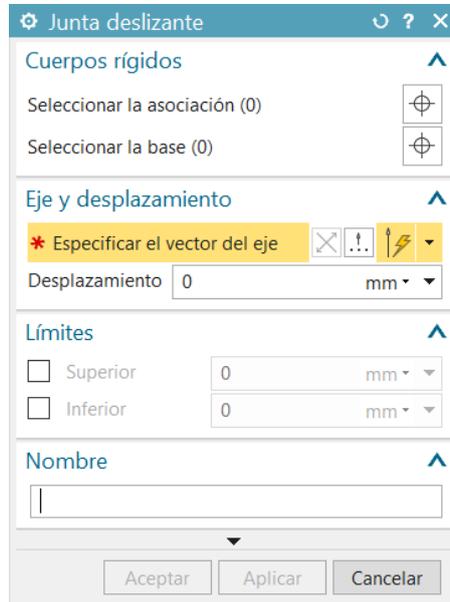


Figura 33. Ventana del comando junta deslizante

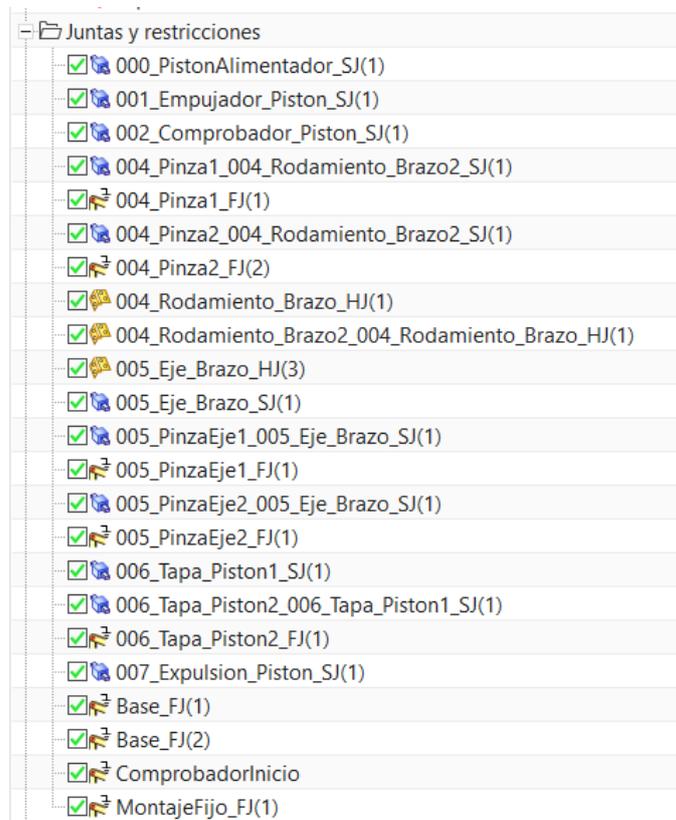


Figura 34. Juntas del proceso completo de montaje de rodamientos

6.2.2. Componentes eléctricos

Los componentes considerados eléctricos que permite definir el programa pueden ser sensores o accionadores (Figura 35). Los sensores utilizados serán: *Sensores de colisión*, *Sensor de distancia* y *Sensor de posición*; los accionadores empleados serán *Control de posición* y *Superficie de transporte*.

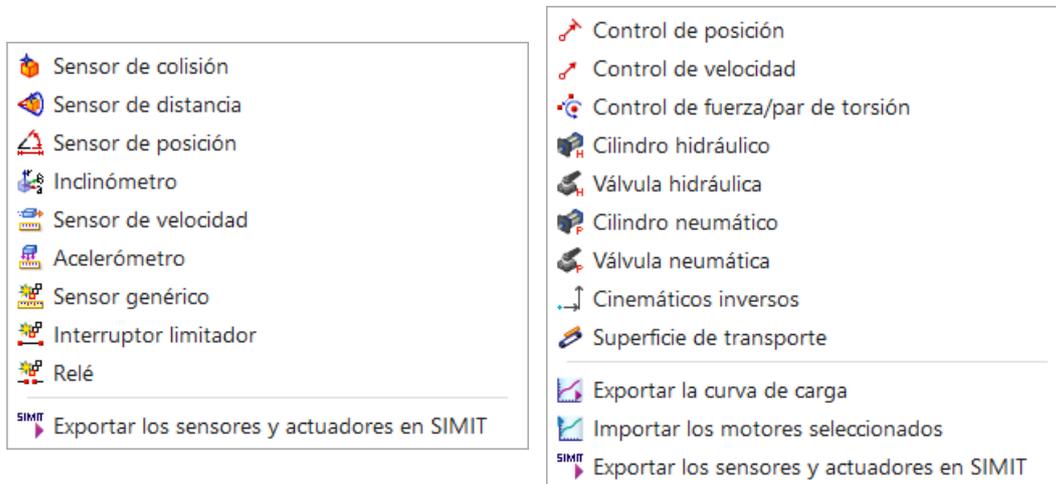


Figura 35. Componentes eléctricos que permite definir el programa

Al igual que ocurre con los movimientos, los sensores que permite definir NX no corresponden exactamente con los empleados en el modelo real, pero con los comandos disponibles es posible conseguir definir otros con la misma función.

- **Sensor de colisión:** Este tipo de sensor permite detectar cuándo un cuerpo entra en contacto con la superficie o plano de un *cuerpo rígido* sobre el que se ha definido este sensor. Pudiendo definir la forma de éste como (Figura 36).

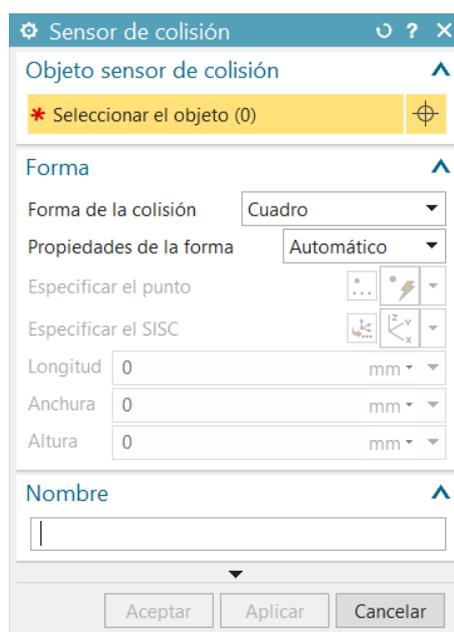


Figura 36. Ventana del comando sensor de colisión

- **Sensor de distancia:** Este sensor se emplea para detectar un objeto que pase por una barrera delimitada por una distancia (rango) definida sobre un vector especificado desde un punto y dentro del alguno de apertura establecido (Figura 37). Es decir, se indica un punto en un *cuerpo rígido* desde el cuál se pretende detectar un objeto a la distancia definida.

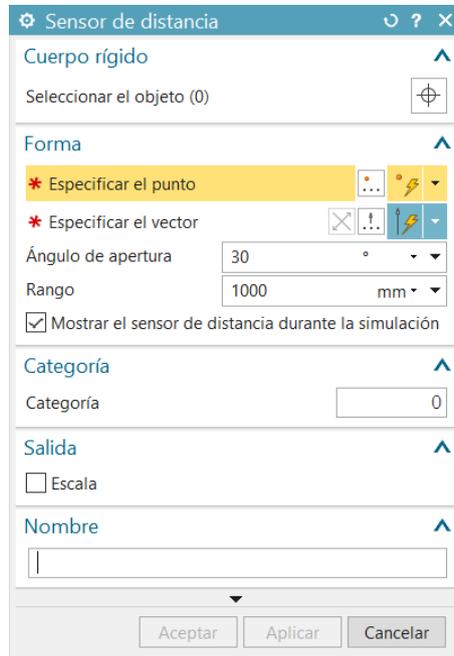


Figura 37. Ventana del comando sensor de distancia

- **Sensor de posición:** Este sensor (Figura 38) se establece sobre un accionador (concretamente *Control de posición o eje de éste*) y no sobre un *cuerpo rígido*. Cuando el accionador se ha desplazado una distancia establecida, proporciona una señal que permite condicionar un movimiento u operación.

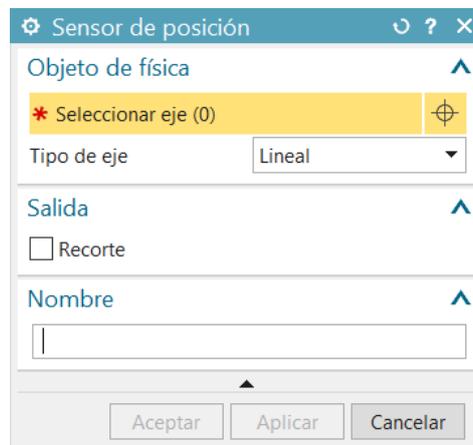


Figura 38. Ventana del comando sensor de posición

- **Control de posición:** Al asignar este comando sobre una *junta* de un accionador, permite controlar su recorrido linear o angular sobre un eje, indicando su posición de destino y la velocidad para alcanzar dicha posición (Figura 39).

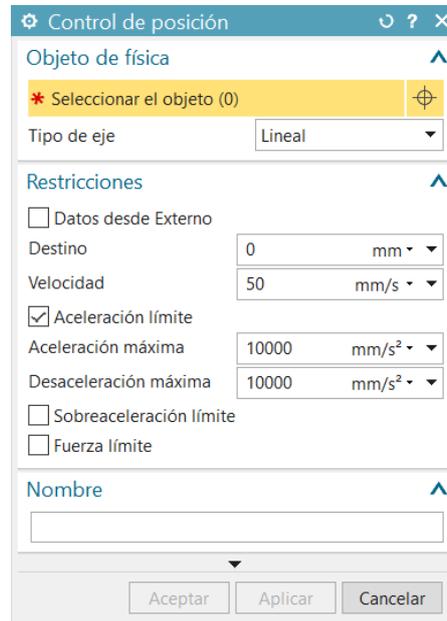


Figura 39. Ventana del comando Control de posición

- **Superficie de transporte:** Normalmente se emplea este comando para simular una cinta transportadora (Figura 40), pero se ha empleado para simular y controlar el desplazamiento de la base al empujarle el pistón. Asignando este comando sobre la superficie de un cuerpo rígido y uno de colisión, éste se desplazará con la velocidad indicada cuando se pongan en contacto.

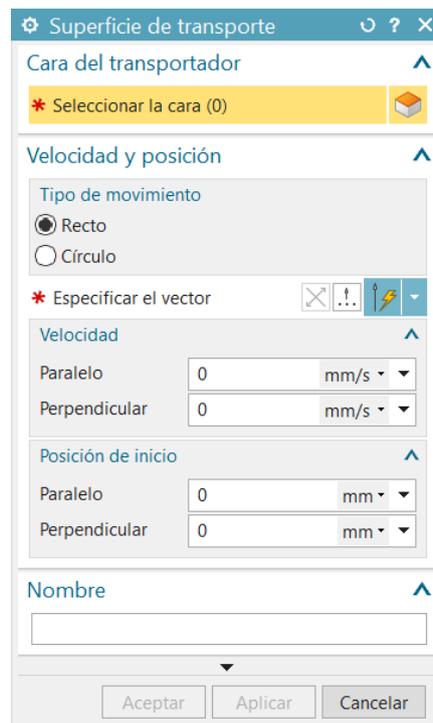


Figura 40. Ventana del comando Superficie de transporte

Algunos de los movimientos de los actuadores no se han condicionado y limitado respecto a los sensores, por lo que algunos de ellos no se han definido en este proceso como sensores sino como límites de un movimiento (Figura 41). Esto significa que los movimientos se han iniciado y finalizado respecto a posiciones y desplazamientos en los que se considera que se ubica el sensor en el modelo real.



Figura 41. Componentes eléctricos del proceso completo de montaje de rodamientos

6.3. Simulación del proceso en Siemens® NX12

Conociendo el funcionamiento y los parámetros de control ya definidos en el apartado anterior (6.1. Descripción del proceso), el siguiente paso en la simulación de la automatización del sistema es la definición de las operaciones que activan o desactivan las entradas, salidas u otras variables manipulables.

Para realizar este paso, el programa dispone de las siguientes herramientas:

- Operación:

Es el comando con el que se definen las etapas y transiciones del proceso. Permite modificar los diferentes parámetros o componentes a controlar, permitiendo establecer etapas de forma secuencial y/o condicionadas a otras etapas o variables. Los parámetros que permite manipular dependerán del objeto seleccionado y los grados de libertad de éste.

Cuando el objeto seleccionado en la operación controle un desplazamiento (Figura 42) los parámetros que se utilizarán serán la velocidad (lineal o angular según el caso), la posición y la activación del objeto. Cuando se trate de activar un sensor o junta, se variará únicamente la opción de activación; en caso de que la operación establezca detener un movimiento, la velocidad y posición serán nulas y la activación se establecerá como falsa.

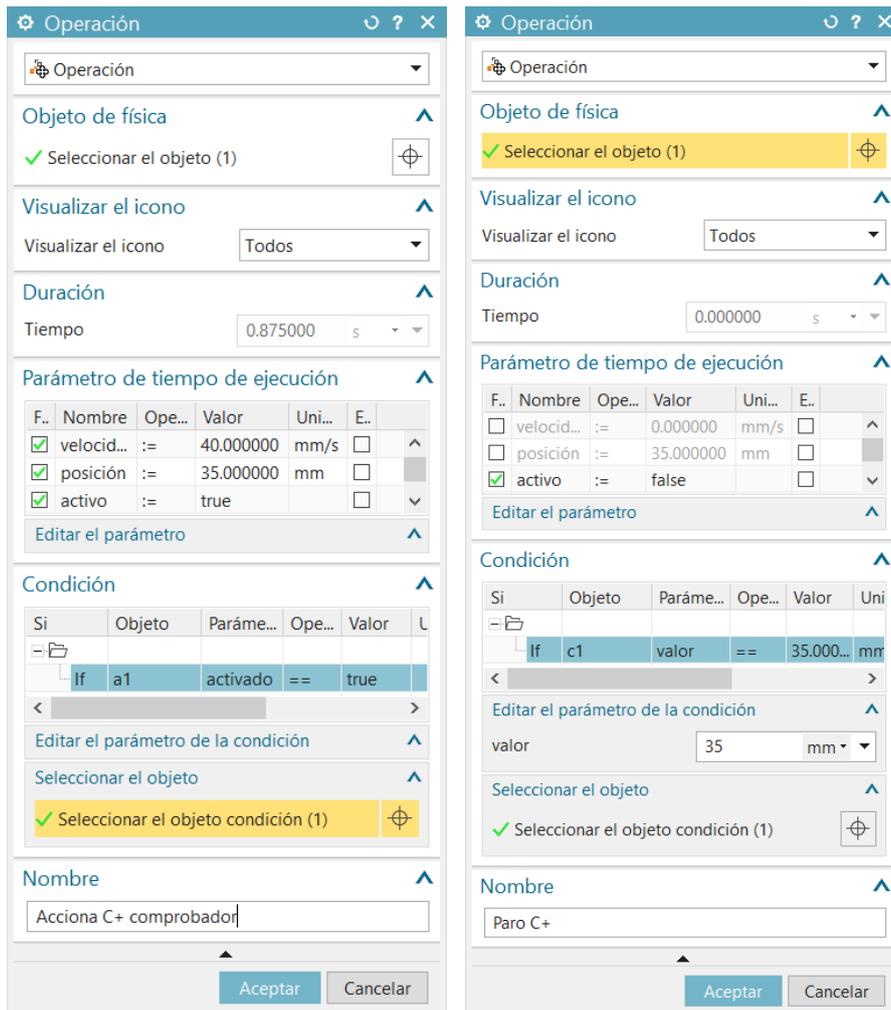


Figura 42. Ventana de operación en Siemens® NX12 para iniciar o detener un movimiento

- Editor de secuencias:

Una vez creadas las operaciones, es en la ventana gráfica *editor de secuencias* donde se aparece la lista completa de ellas y su duración (Figura 44). En esta ventana es posible modificar las operaciones ya creadas, activarlas o desactivarlas, y condicionarlas secuencialmente a la finalización de otra operación o activación de un sensor.

- Inspector tiempo ejecución:

Si se quiere conocer la posición o el estado en que se encuentra una variable o componente, la herramienta Inspector de tiempo de ejecución permite visualizar los distintos valores de los parámetros que lo describen (Figura 43). Para visualizar estos valores, se debe seleccionar y añadir al inspector el elemento en cuestión.

Inspector de tiempo de ejecución

Inspector						
Física	Gráf...	Exp...	Grabar	Valor	Unidad	
[-] a1						
[-] activado				true		
[-] valor	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0.560332	mm	
[-] activo				true		
[-] 000_PistonAlimentado...						
[-] eje				000_PistonA...		
[-] velocidad	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	14.168546	mm/s	
[-] posición	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	97.345475	mm	
[-] activo				true		
[-] aceleración límite				false		
[-] aceleración	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	33499.615614	mm/s ²	
[-] desaceleración	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	-33499.6156...	mm/s ²	
[-] sobreaceleración lí...				false		
[-] sobreaceleración	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	57043835.68...	mm/s ³	
[-] sobrecarga				false		
[-] fuerza						
[-] x	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0.771890	N	
[-] y	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1300.573579	N	
[-] z	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0.618164	N	

Figura 43. Ventana de visualización del Inspector de tiempo de ejecución

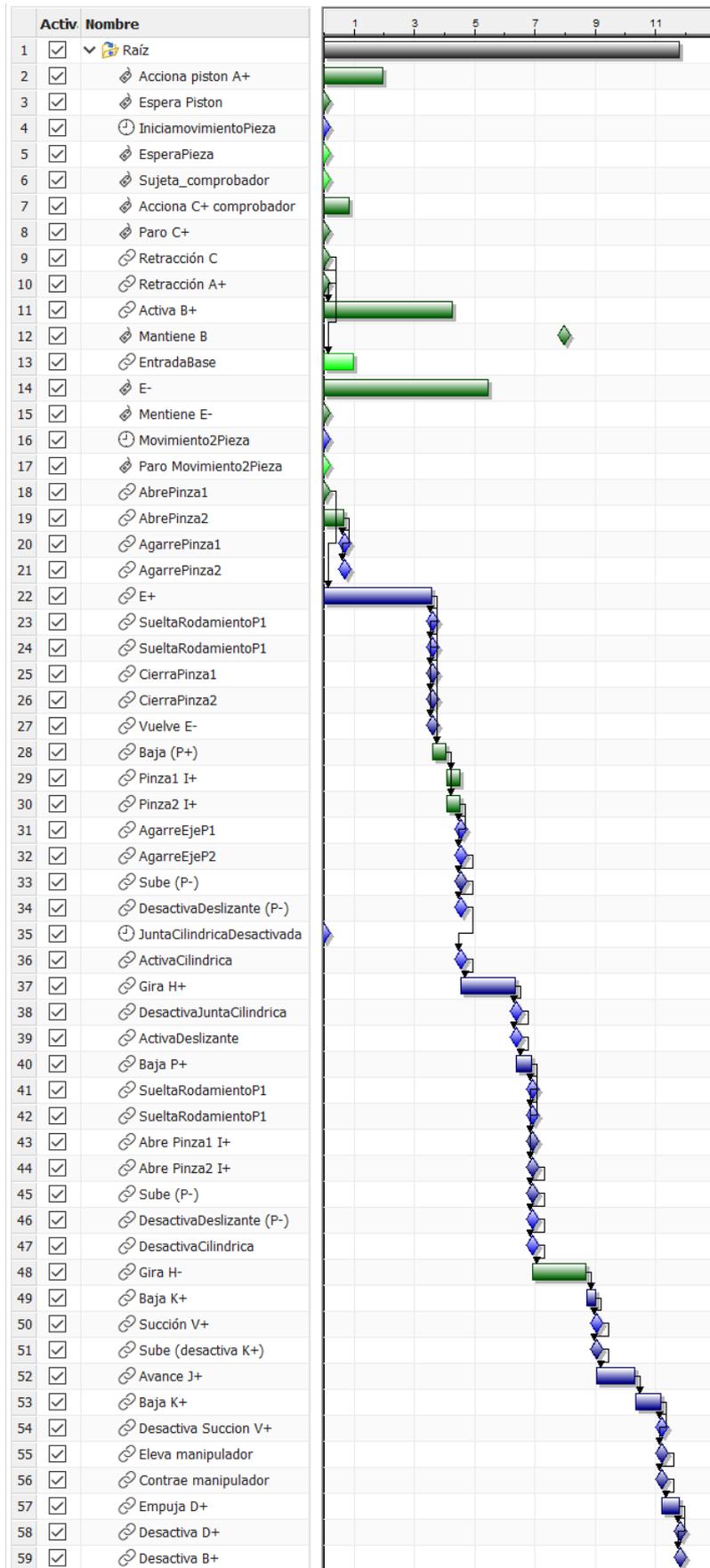


Figura 44. Editor de operaciones de la simulación del proceso de montaje de rodamientos

6.4. Integración de Siemens® NX12 en CODESYS® a través de TCP/IP

La comunicación entre el controlador y el gemelo digital se realiza a través de un enlace TCP/IP, lo que requiere que los valores de las entradas y salidas sean binarios, es decir, que las señales sean verdaderas o falsas según estén o no activadas.

Los sensores y movimientos definidos en Siemens NX no corresponden exactamente con los existentes en el modelo real, por lo que es necesario adaptar las señales disponibles para que las señales que se envíen o reciban el modelo digital correspondan con los valores binarios que proporcionarían los sensores y actuadores del modelo físico (*Ver anexo I*).

6.4.1. Controles de posición

Los actuadores que realizan los movimientos son los mismos que cuando se realiza una simulación únicamente en NX12 mediante operaciones, pero en este caso la condición que los inicia o desactiva no es otra operación ni sensor, sino una *señal externa* recibida del controlador.

Partiendo de las juntas ya definidas que determinarán los grados de libertad de cada componente móvil, se debe crear un control de posición para cada movimiento de avance y otro para cada movimiento de retroceso (Figura 48), sean o no monoestables, pues el programa requiere que se indique la posición en cada momento y no vuelve por defecto a la de inicio.

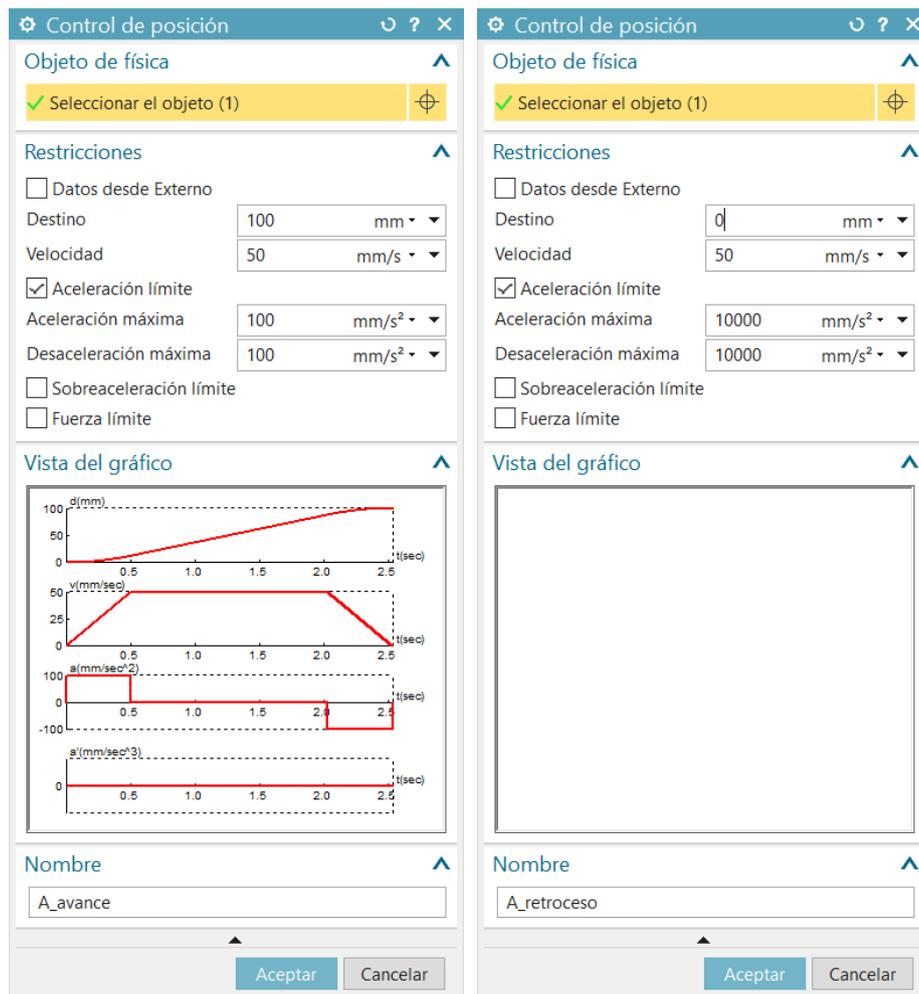
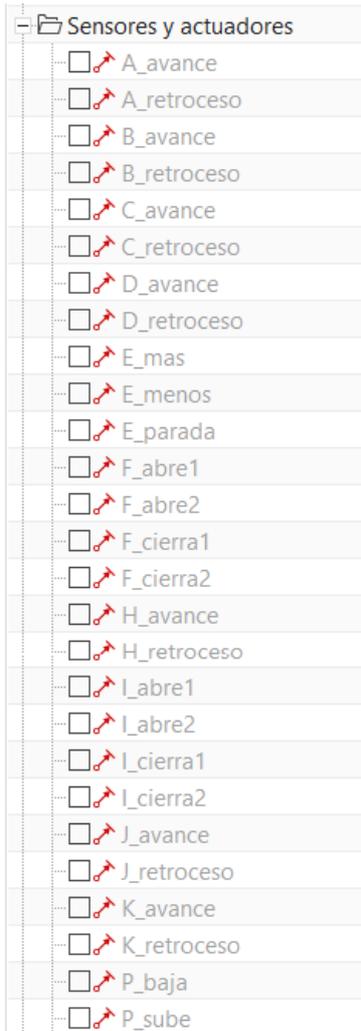


Figura 45. Controles de posición del movimiento de avance y contracción del alimentador



Los controles creados (Figura 47) permanecerán desactivados y serán las señales externas las que los activen cuando corresponda, por lo que es necesario crear una señal de entrada para cada acción y vincularla con el control de posición que determine dicha acción (Figura 46).

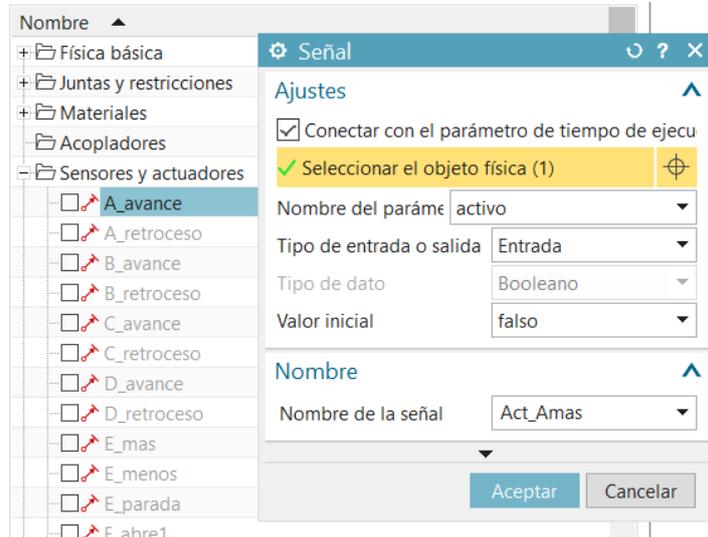


Figura 46. Señal de entrada del movimiento de avance del alimentador.

Figura 47. Controles de posición del control externo

En el caso de movimientos verticales o con componente en el eje Z, como sería el caso del comprobador de la orientación de la base, es necesario que cuando se inicie la simulación, permanezca activada por defecto la *señal* del *control de posición* que lo desplaza a la posición inicial (Figura 48), pues el programa tiene en cuenta la fuerza de gravedad y el pistón caería al vacío si no se indica ninguna posición.

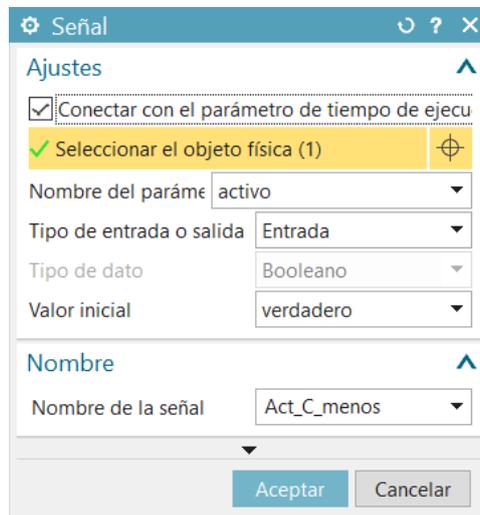


Figura 48. Definición de la señal que sujeta el comprobador de bases

6.4.2. Sensores de posición y distancia

El envío de señales de los sensores también debe realizarse mediante de valores binarios, por lo que se definen sensores que registren la posición de un actuador a lo largo de un eje (Figura 49) y se les asociará una señal de salida que se activará (valor de la señal = true) cuando la posición del actuador sea aquella en la que se ubicaría el sensor físicamente.

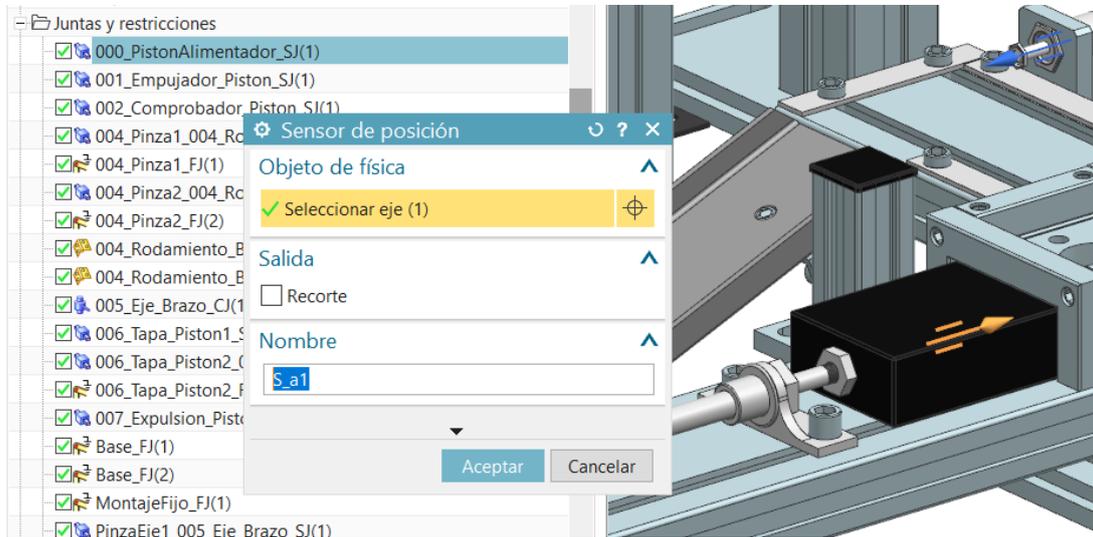


Figura 49. Sensor de posición del pistón alimentador

Para que las señales de salida de los sensores correspondan con valores binarios, se emplea el *adaptador de señales*. Este comando permite asociar el sensor con una señal de salida, que tomará el valor que se establezca cuando se cumpla la condición indicada (Figura 50).

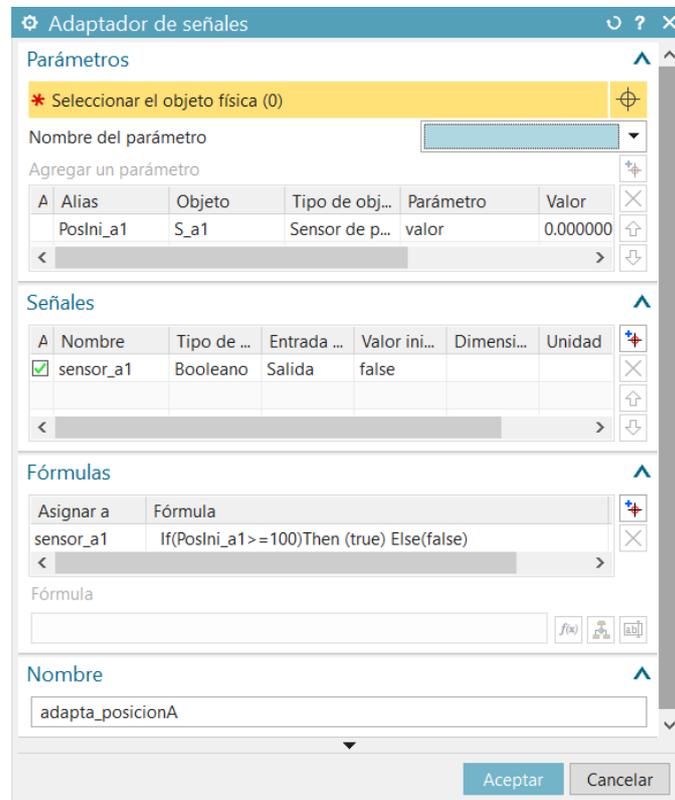


Figura 50. Adaptador de señales del sensor a1.

En el caso de los sensores que detectan las piezas (dpr1, dpr2 y dpr3), al no tratarse de movimientos, se han definido como sensores de distancia, que se activan o desactivan según detecten un cuerpo en la distancia que abarca su rango. Para asociar las señales de salida de estos sensores también se ha empleado el *adaptador de señales*, lo que permite además establecer dpr2 y dpr3 como en el modelo real, es decir, que la señal sea 0 cuando detecte pieza y 1 cuando no (Figura 51).

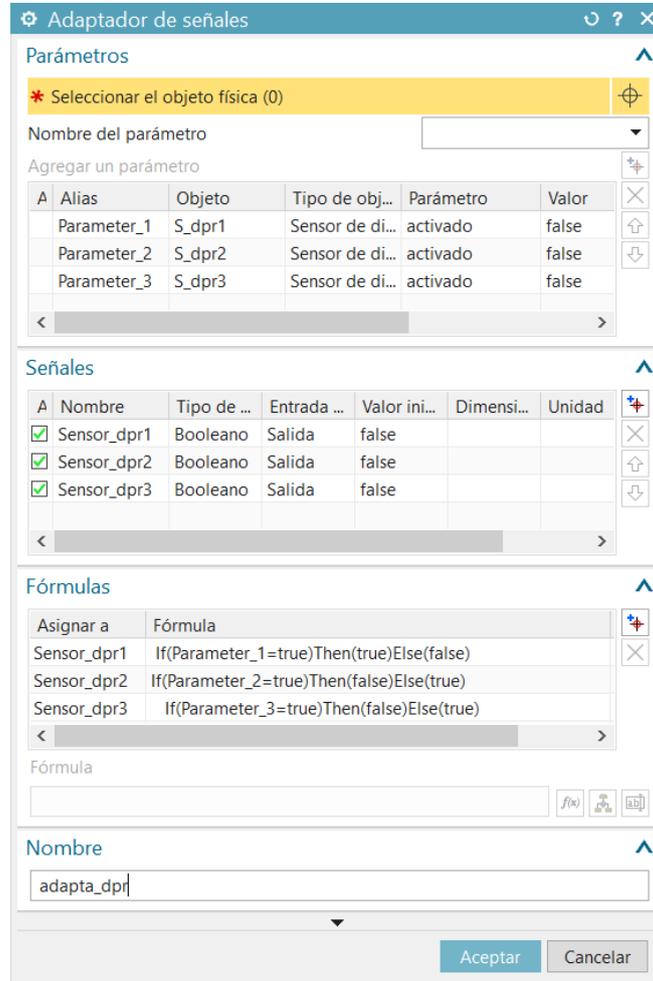


Figura 51. Adaptador de señales de los sensores de piezas

6.4.3. Agarre de piezas

Las acciones de agarre de piezas (pick & place), son movimientos que no incluye en controlador, pues en el modelo real la propia física de los cuerpos permite sujetarlos y soltarlos según la presión de los mecanismos de agarre como las pinzas. Sin embrago, para que la simulación del gemelo digital sea asimile visualmente al funcionamiento del sistema real, se han definido las operaciones (Figura 52) que permiten realizar esos movimientos.

1	▼ Raíz
2	🔗 V+
3	🔗 Desactiva V+
4	🔗 PinzaEje1
5	🔗 Suelta PinzaEje1
6	🔗 PinzaEje2
7	🔗 Suelta PinzaEje2
8	🔗 PinzaRod1
9	🔗 Suelta PinzaRod1
10	🔗 PinzaRod2
11	🔗 Suelta PinzaRod2

Figura 52. Operaciones creadas para el movimiento de piezas.

El agarre de las piezas se realiza activando una *junta fija* cuando un sensor de colisión detecta un cuerpo, es decir, se define una operación que active dicha junta que fije al *sensor de colisión* el cuerpo detectado cuando se active la señal correspondiente (Figura 53).

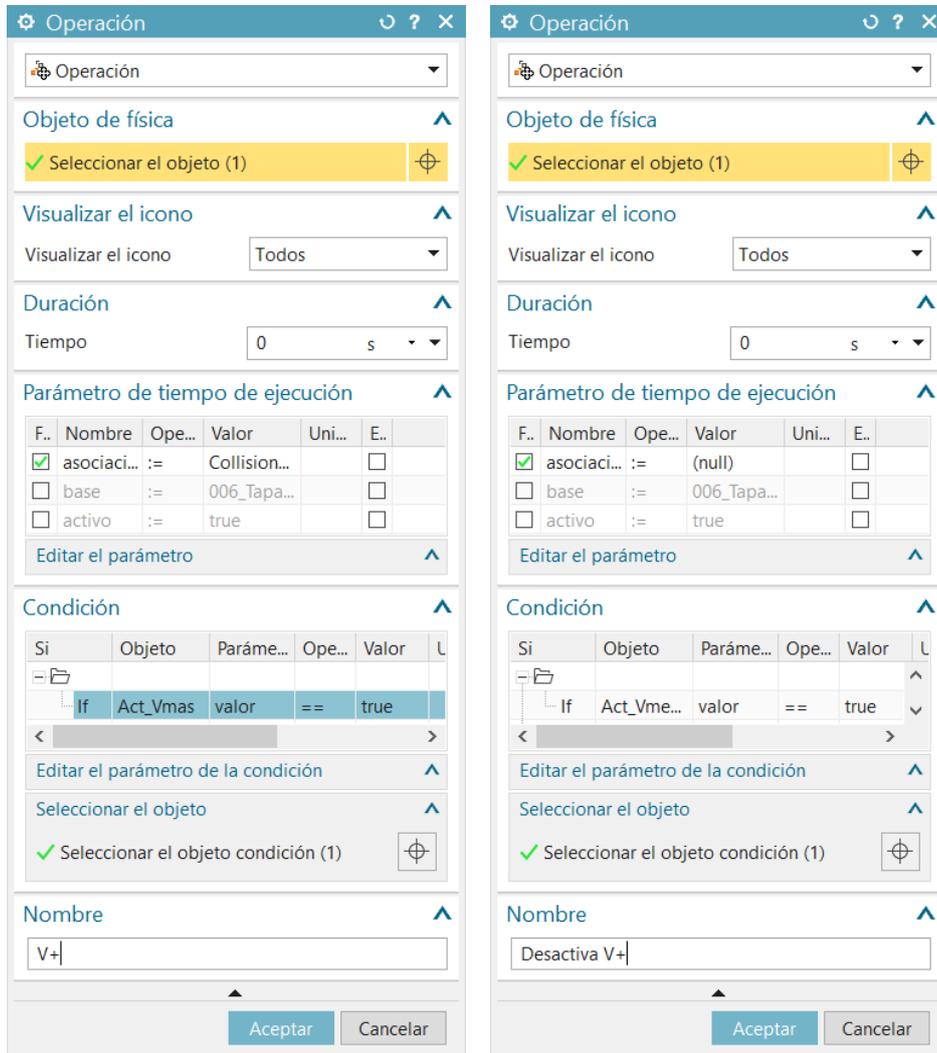


Figura 53. Operaciones de activación y desactivación de la succión de la tapa.

6.4.4. Comunicación de señales

Una vez se han creado las señales de entrada y salida correspondientes, se deben crear los mismos registros de señales en el controlador y en NX y enlazarlos, por lo que las direcciones de registro de modbus en el controlador deben coincidir con las comunica NX. (Ver anexo I).

El comando mapeo de señales (Figura 54) es la herramienta que permite relacionar la señal externa con las señales de los sensores y actuadores de NX.

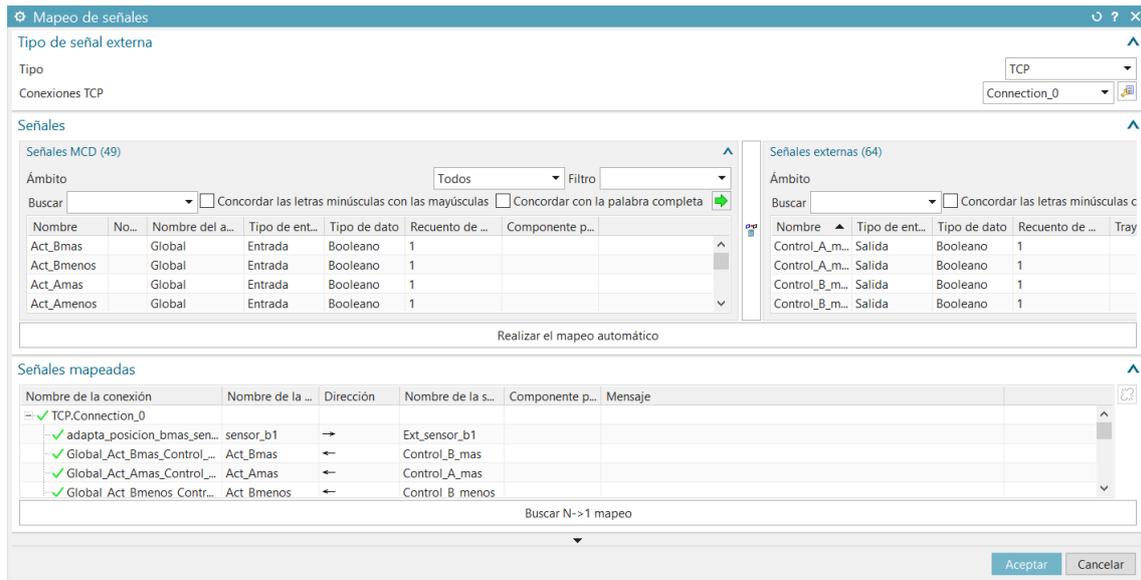


Figura 54. Mapeo de señales del sistema

El siguiente paso en la implementación del control es la introducción de los grafcet (6.1. Descripción del proceso) en CODESYS (Figura 55), que será quien transmita los estados de los GRAFCET al controlador y al enlace.

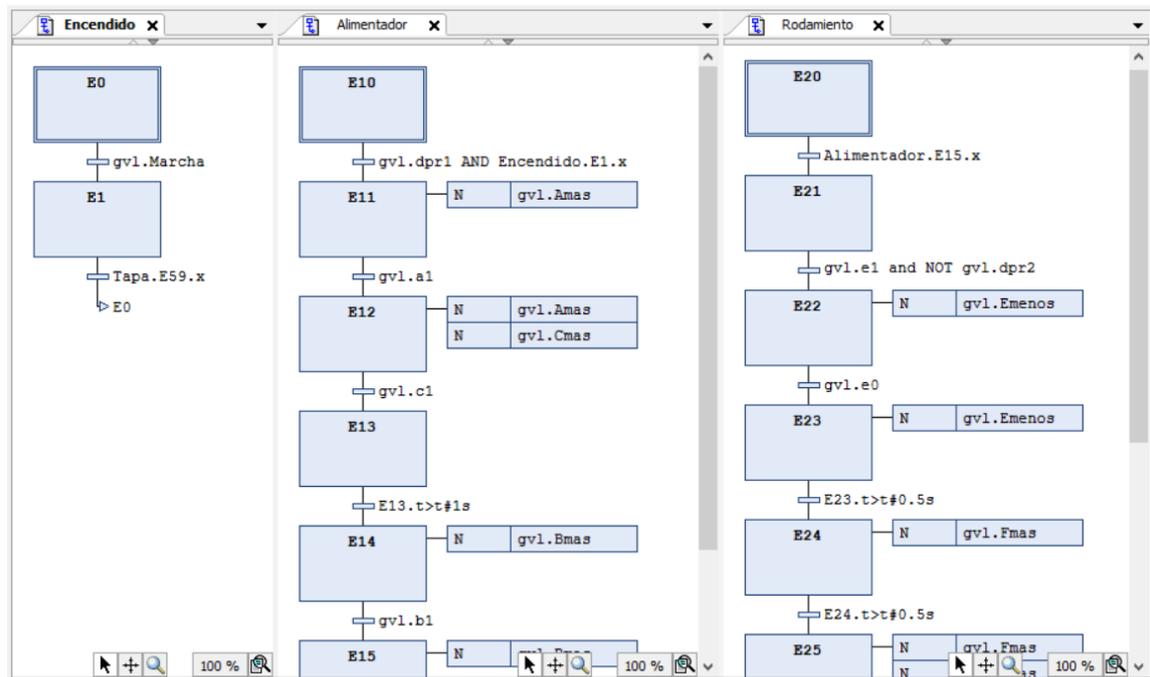


Figura 55. Visualización del control en CODESYS

El proceso se encuentra inicialmente en la etapa E0 y cuando se inicia el proceso activando la acción *Marcha*, éste cambiará de estado pasando a la etapa E1 en la que permanecerá hasta que finalicen todos los subprocessos; esta etapa a su vez inicia el proceso de alimentación de bases si se hay una base disponible.

Cuando el controlador activa una acción de control como podría ser *Amas*, almacena su valor binario en la dirección de registro correspondiente de modbus (QX0.2). En enlace copia los 4 registros de acciones de modbus (4 registros de 8 bits cada uno) y los transmite a NX por TCP, que según el bit que reciba para cada acción, la activará o desactivará. Por otra parte, NX transfiere los otros 4 registros correspondientes a los sensores por TCP, que los escribe por modbus sobre el PLC.

Si el controlador activa la acción de control *Amas*, a través de TCP llega una señal a NX que conecta con la propiedad que activa *Amas*, se pondrá en funcionamiento el control de posición que desplaza *A* hasta que se alcance el valor indicado en el sensor que se relaciona con dicho movimiento, en ese momento se activará la señal que produce ese sensor gracias al *adaptador de señales*. NX enviará los registros de los sensores por TCP que los escribirá por modbus sobre el PLC, cuando éste detecte el sensor indicado en la transición (*a1*), pasará a la siguiente etapa (Figura 56).

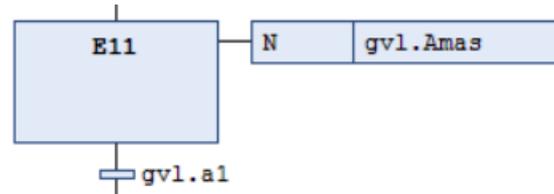


Figura 56. Ejemplo de etapa y transición

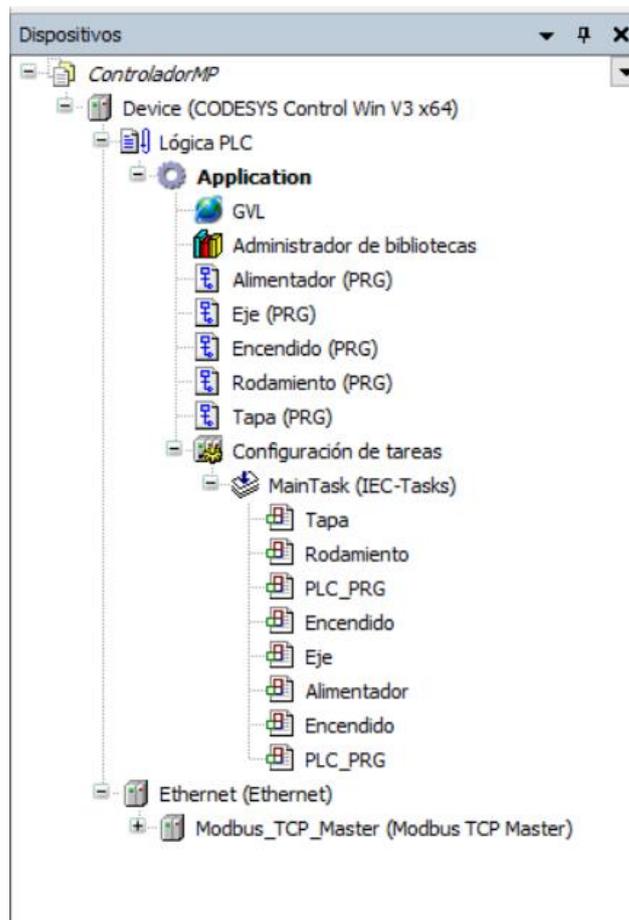


Figura 57. Estructura del controlador

6.5. Automatización del gemelo digital

Una vez se han enlazado todos los accionamientos y señales y se ha conseguido su correcta comunicación y funcionamiento, el gemelo digital estará completamente definido y programado su control.

La activación del pulsador de arranque iniciará el proceso de control, cuyo funcionamiento debe ser el mismo que se observó en la célula SMC real al implementar ese mismo programa de control.

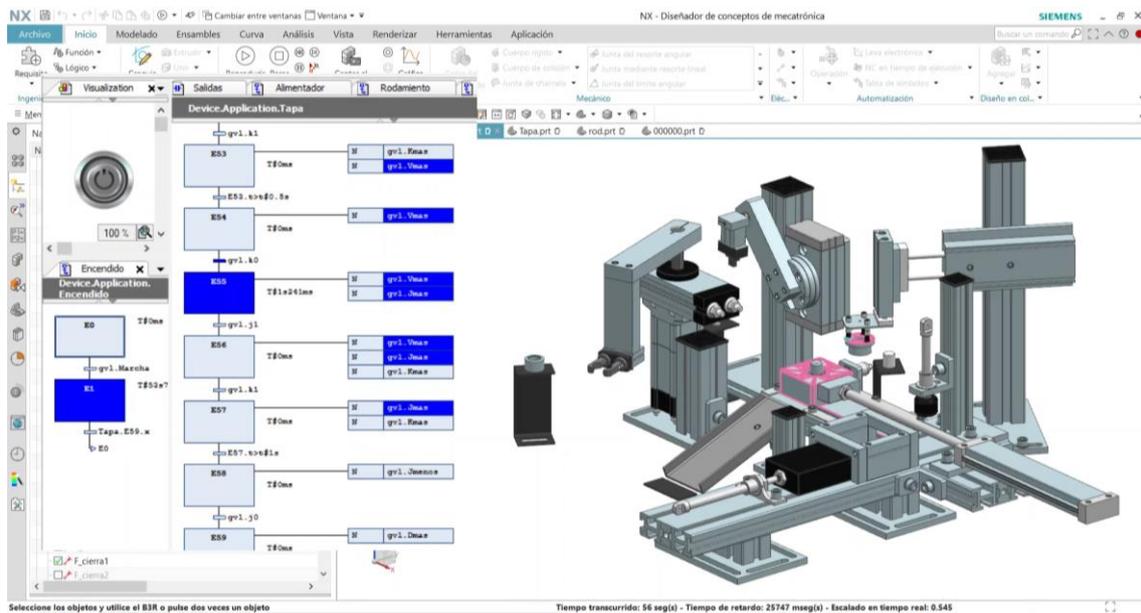


Figura 58. Colocación de la tapa durante el control con CODESYS

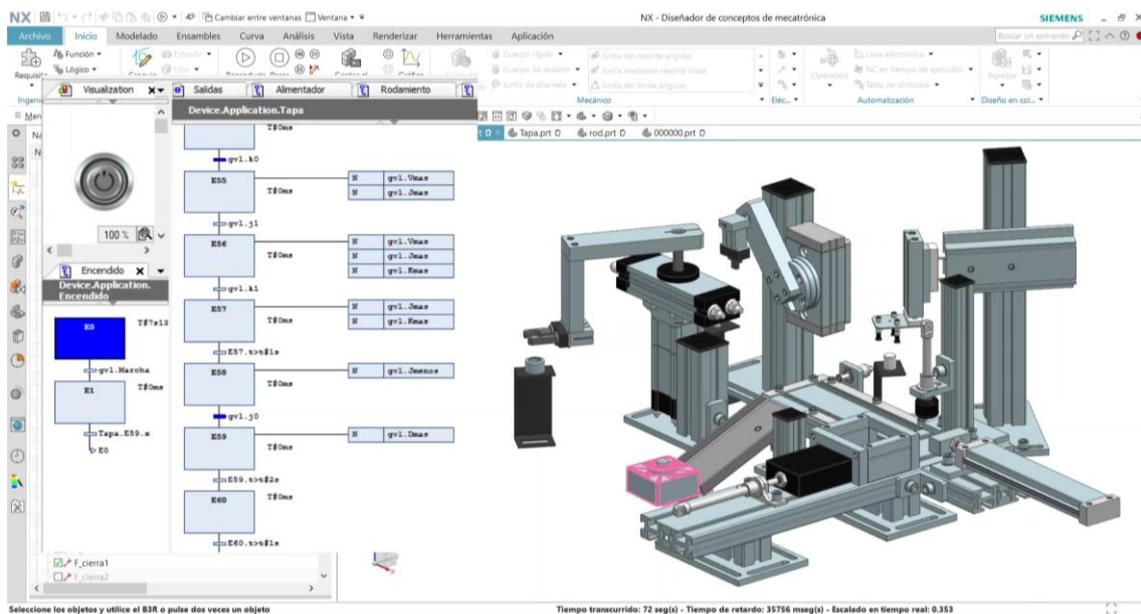


Figura 59. Final del proceso de montaje

7. Valoración económica

A continuación, se realizará un estudio económico para justificar la inversión en la realización del gemelo digital a partir de los costes totales del proyecto. La valoración económica descrita a continuación considera que los costes totales incluyen los costes directos, indirecto y generales (Ec. 1):

$$C_T = C_D + C_I + C_G \quad (\text{Ec. 1})$$

Siendo:

- C_T = Costes totales
- C_D = Costes directos
- C_I = Costes indirectos
- C_G = Costes generales

7.1. Costes directos

Se consideran costes directos a todos aquellos asociados directamente al diseño y fabricación de un producto individual. En estos gastos se incluye la mano de obra directa, el material, equipos y/o maquinaria.

7.1.1. Coste de mano de obra directa (C_{MOD})

Este coste se refiere a la actividad directamente relacionada con la fabricación de un producto realizada por el personal. Se ha calculado a partir del tiempo empleado y la retribución del operario (Tabla 12), considerando que se han empleado 450 horas y la retribución del personal es de 15 €/h.

Tabla 12. Coste de mano de obra directa

Operarios	Tiempo (horas)	Jornal (€/hora)	C_{MOD} (€)
1	450	15	6750

Las horas requeridas para el desarrollo del proyecto son la suma las horas empleadas en las distintas tareas (Figura 60), siendo la que más tiempo ha requerido la etapa de modelado, que además posteriormente ha necesitado tiempo extra para realizar los ajustes necesarios que permitan el correcto funcionamiento.

La etapa de diseño del control o automatización del sistema es la que menos tiempo ha requerido, por lo que una vez se haya definido el sistema completamente, será posible programarlo y modificar su control de forma rápida.

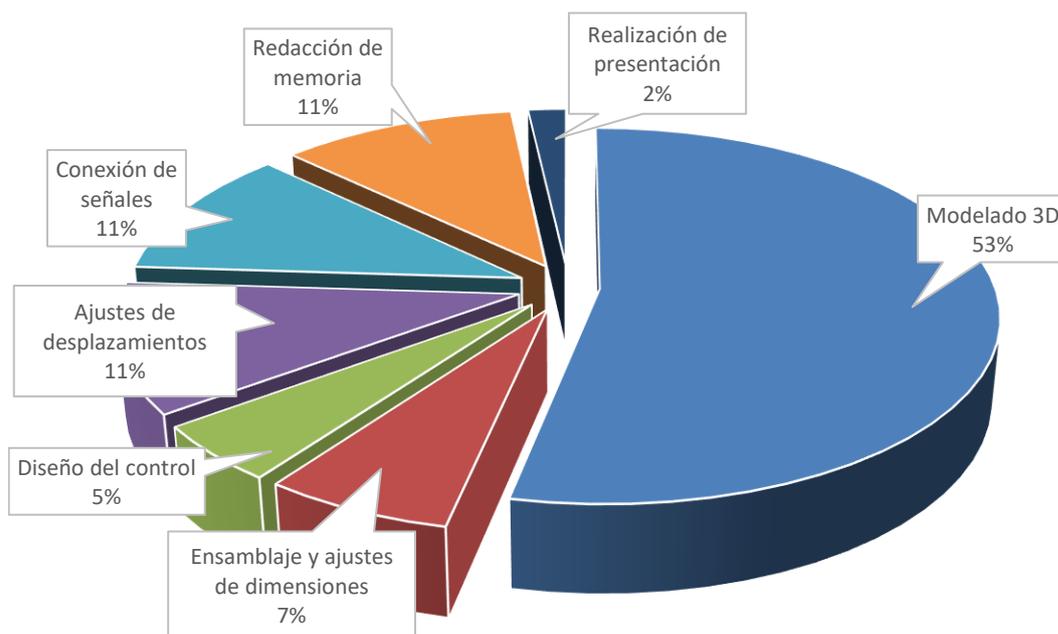


Figura 60. Reparto de los tiempos del proyecto

7.1.2. Coste de material (C_{MAT})

En el caso de un producto virtual, se considera coste de material al precio de las licencias del software utilizado durante el tiempo de realización del proyecto, proporcional al periodo de amortización de cada una de ellas (Tabla 13).

Tabla 13. Costes de material correspondientes a las licencias de software

Programa	Periodo de amortización (meses)	Periodo amortizado (meses)	Precio de licencia (€)	Coste de material (€)
Siemens® NX12	12	6	16000 ^[13]	8000
CODESYS®	12	1	50	0*
Office 365®	12	2	149	24,83
Coste total de material (C_{MOD})				8024,83

* Se ha empleado una versión gratuita de prueba de CODESYS®

7.1.3. Coste de equipos (C_{MAQ})

Estos costes incluyen el precio de los equipos utilizados durante la realización el proyecto (Tabla 14). En este caso ha sido un ordenador ASUS F509FB, con un precio de adquisición de 699 € y considerando que el periodo de amortización son 2 años.

Tabla 14. Costes de directos de equipos y maquinaria

Equipo	Periodo de amortización (meses)	Periodo amortizado (meses)	Precio (€)	C_{MAQ} (€)
ASUS F509FB	48	6	699	87,38

7.2. Costes indirectos y costes generales

Los costes indirectos y costes generales son aquellos que no son directamente atribuibles a una única unidad de proyecto, pero se relacionan con los recursos empleados en la actividad cotidiana de una empresa.

En este proyecto no se han considerado costes indirectos ni generales, pues no se ha requerido alquilar instalaciones, contratar personal auxiliar, etc.

7.3. Coste total del proyecto

El coste total de la realización del proyecto del gemelo digital corresponde, por tanto, a los costes directos ya descritos, que como se observa en la (Tabla 15), ascienden a 14862,21 €. A este importe se le aplicaría el porcentaje de beneficios que se consideren para obtener un precio de venta y distribución si se fuera a comercializar.

Tabla 15. Costes totales del proyecto

Proyecto	C_{MOD} (€)	C_{MAT} (€)	C_{MAQ} (€)	C_T (€)
<i>Gemelo digital</i>	6750	8024,83	87,38	14862,21

8. Conclusiones

Tras modelar el sistema y realizar las diferentes simulaciones hasta lograr el correcto funcionamiento del gemelo digital para poderlo contemplar como tal, se considera que se ha logrado cumplir el objetivo principal de esta propuesta de trabajo: integrar los distintos conocimientos adquiridos durante el *Máster en Diseño y Fabricación Integrada Asistidos por Computador* desarrollando un gemelo digital.

Por una parte, el modelado de los distintos cuerpos aplica directamente los conocimientos relacionados con el diseño asistido por ordenador y por otra, la programación del control de dicho modelo aplica los conocimientos adquiridos en el campo de la automatización industrial. Además, su posible aplicación es una herramienta para avanzar hacia la industria 4.0, en la que se basan las competencias generales que proporciona el máster.

El proyecto desarrollado se espera que sirva como tanto como complemento en la docencia de futuros estudiantes de la asignatura de *Automatización Industrial*, como referencia para futuros proyectos en este campo. En cuanto a la posibilidad de aplicarlo en la docencia, en este trabajo se ha creado un control del funcionamiento básico de la célula para comprobar que actúa como gemelo del proceso real, pero podría crearse otro programa de control para modificar el orden de los movimientos, optimizar tiempos, corrección de posiciones...

Las mayores dificultades que han surgido durante este proyecto han sido, por una parte, el modelado sin ninguna referencia dimensional, y, por otra, la definición de movimientos controlados y la comunicación con CODESYS, al existir muy pocas referencias por no haberse realizado trabajos de este tipo previamente.

La interfaz de la aplicación *Diseñador de Conceptos de Mecatrónica* de Siemens® NX12 también ha conllevado algunos inconvenientes debido a la aleatoriedad en los cálculos que producen las vibraciones de algunas piezas, provocando fallos irresolubles y choques ficticios que no ocurren realmente en el modelo físico, lo que implica tener que reiniciar la simulación frecuentemente.

Lograr resolver el proyecto planteado siendo lo más fiel posible a la realidad, tanto visualmente como en su funcionamiento, ha determinado que el trabajo en su conjunto haya supuesto un enorme reto que ha requerido mucho tiempo de dedicación y pruebas de simulación, pero su enorme utilidad y resultados fácilmente apreciables proporcionan gran satisfacción y motivación durante todo el proceso hasta el completo desarrollo del gemelo digital.

Anexo I. Conexión de señales

La comunicación entre CODESYS® y Siemens® NX12 a través de TCP se realiza mediante el envío de registros o “paquetes” de 8 bits, siendo necesarios 4 registros de entradas y 4 de salidas según el número de señales del proceso en cuestión.

El comando *configuración de la señal externa* (Figura 61) permite crear y configurar la conexión y las señales. Una vez creada la conexión, es importante que la dirección IP coincida con la red que se va a utilizar y las direcciones de las señales creadas coincidan con las señales del controlador (Figura 62). Para evitar errores en la comunicación, es necesario crear 32 señales de entrada externa y 32 de salida, aunque en la implementación no se utilicen todas ellas.

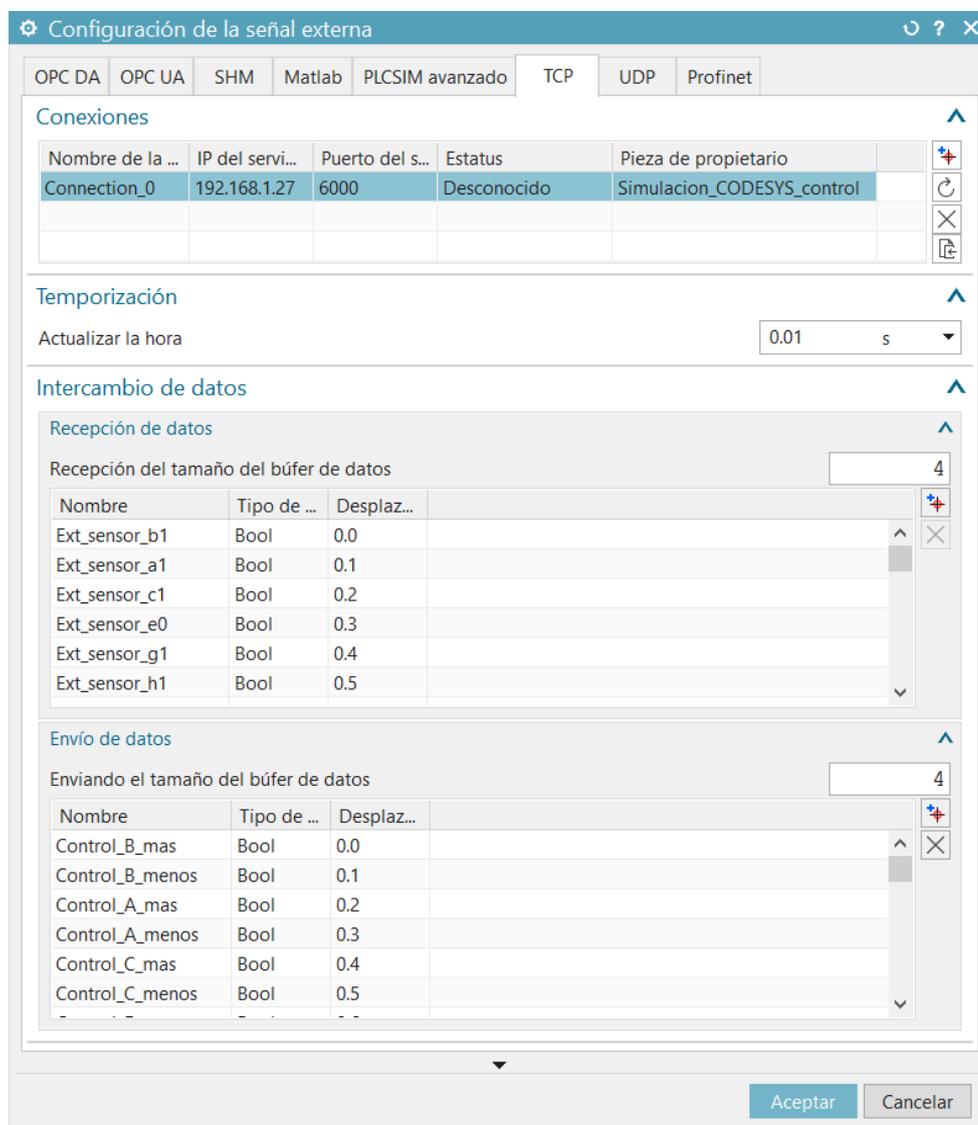


Figura 61. Configuración de las señales externas en Siemens® NX12

```

END_VAR
VAR_GLOBAL PERSISTENT
  b1 AT %IX0.0: BOOL;
  a1 AT %IX0.1: BOOL;
  c1 AT %IX0.2: BOOL;
  e0 AT %IX0.3: BOOL;
  g1 AT %IX0.4: BOOL;
  h1 AT %IX0.5: BOOL;
  g0 AT %IX0.6: BOOL;
  h0 AT %IX0.7: BOOL;
  k1 AT %IX1.0: BOOL;
  k0 AT %IX1.1: BOOL;
  j1 AT %IX1.2: BOOL;
  j0 AT %IX1.3: BOOL;
  e1 AT %IX1.4: BOOL;
  e2 AT %IX1.5: BOOL;
  i1 AT %IX1.6: BOOL;
  i0 AT %IX1.7: BOOL;
  dpr1 AT %IX2.0: BOOL;
  dpr2 AT %IX2.1: BOOL;
  dpr3 AT %IX2.2: BOOL;

  Marcha : BOOL;

END_VAR

VAR_GLOBAL
  Bmas AT %QX0.0: BOOL;
  Bmenos AT %QX0.1: BOOL;
  Amas AT %QX0.2: BOOL;
  Amenos AT %QX0.3: BOOL;
  Cmas AT %QX0.4: BOOL;
  Cmenos AT %QX0.5: BOOL;
  Emenos AT %QX0.6: BOOL;
  Eparada AT %QX0.7: BOOL;
  Fmas AT %QX1.0: BOOL;
  Fmenos AT %QX1.1: BOOL;
  Hmas AT %QX1.2: BOOL;
  Hmenos AT %QX1.3: BOOL;
  Kmas AT %QX1.4: BOOL;
  Kmenos AT %QX1.5: BOOL;
  Jmas AT %QX1.6: BOOL;
  Jmenos AT %QX1.7: BOOL;
  Emas AT %QX2.0: BOOL;
  Dmas AT %QX2.1: BOOL;
  Dmenos AT %QX2.2: BOOL;
  Fmas AT %QX2.3: BOOL;
  Fmenos AT %QX2.4: BOOL;
  Imas AT %QX2.5: BOOL;
  Imenos AT %QX2.6: BOOL;
  Vmas AT %QX2.7: BOOL;
  Vmenos AT %QX3.0: BOOL;

END_VAR

```

Figura 62. Variables definidas en CODESYS

■ Mapeo de señales

Las señales externas tienen que estar a su vez enlazadas con las señales que activan o desactivan una acción y las que proporcionan los sensores; la vinculación de las internas de la simulación con las externas se realiza en la opción *mapeo de señales* (Figura 63).

Esta herramienta muestra a la izquierda las señales internas del modelo digital y a la derecha, las externas con las que comunicará con el controlador. Pulsando el botón central se conectan las dos seleccionadas.

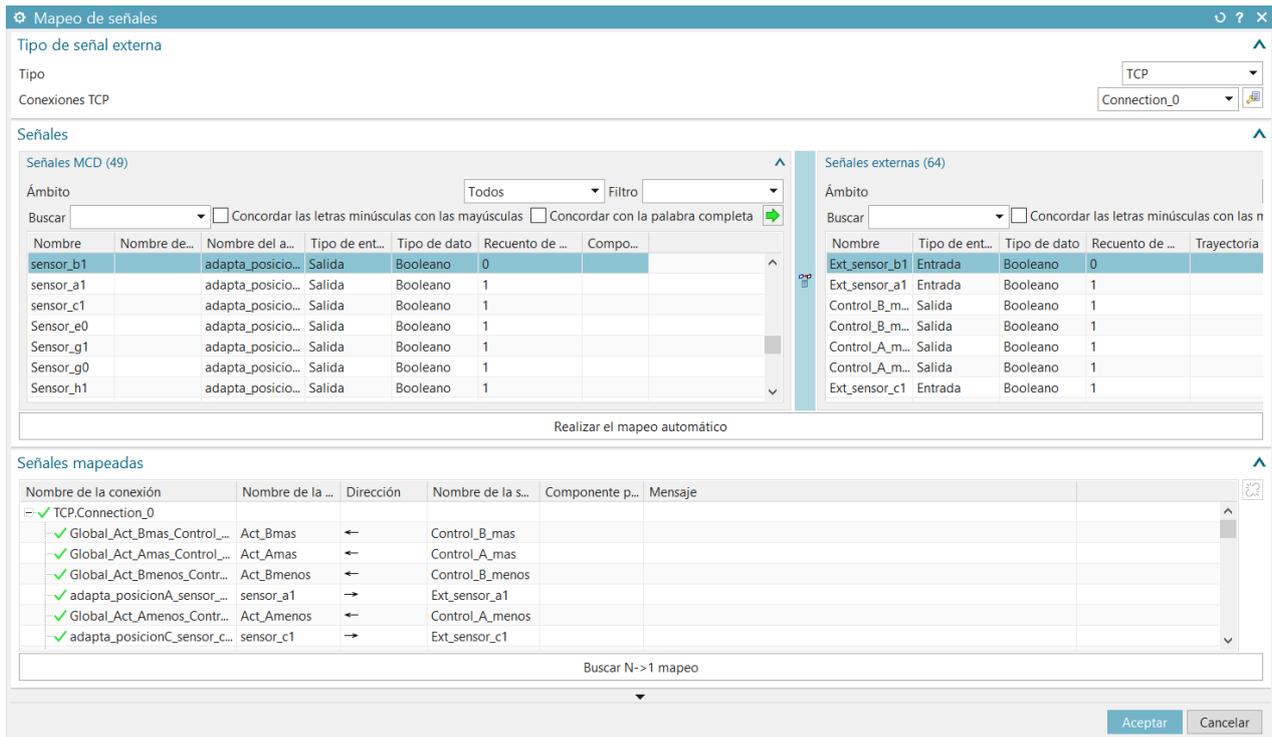


Figura 63. Ventana de mapeo de señales en Siemens® NX12

En algunos casos como los movimientos de las pinzas, una misma señal de entrada externa se enlazará con dos señales de activación interna, correspondientes a cada uno de los dedos de la pinza que se moverán a la vez.

Señales mapeadas

Nombre de la conexión ▲	Nombre de la ...	Dirección	Nombre de la s...
✓ TCP.Connection_0			
✓ adapta_dpr_Sensor_dpr1_Ext_sensor_dpr1	Sensor_dpr1	→	Ext_sensor_dpr1
✓ adapta_dpr_Sensor_dpr2_Ext_sensor_dpr2	Sensor_dpr2	→	Ext_sensor_dpr2
✓ adapta_dpr_Sensor_dpr3_Ext_sensor_dpr3	Sensor_dpr3	→	Ext_sensor_dpr3
✓ adapta_posicion_PinzaEje_Sensor_i0_Ext_sensor_i0	Sensor_i0	→	Ext_sensor_i0
✓ adapta_posicion_PinzaEje_Sensor_i1_Ext_sensor_i1	Sensor_i1	→	Ext_sensor_i1
✓ adapta_posicionA_sensor_a1_Ext_sensor_a1	sensor_a1	→	Ext_sensor_a1
✓ adapta_posicionC_sensor_c1_Ext_sensor_c1	sensor_c1	→	Ext_sensor_c1
✓ adapta_posicionE_Sensor_e0_Ext_sensor_e0	Sensor_e0	→	Ext_sensor_e0
✓ adapta_posicionE_Sensor_e1_Ext_sensor_e1	Sensor_e1	→	Ext_sensor_e1
✓ adapta_posicionE_Sensor_e2_Ext_sensor_e2	Sensor_e2	→	Ext_sensor_e2
✓ adapta_posicionEje_Sensor_g0_Ext_sensor_g0	Sensor_g0	→	Ext_sensor_g0
✓ adapta_posicionEje_Sensor_g1_Ext_sensor_g1	Sensor_g1	→	Ext_sensor_g1
✓ adapta_posicionEje_Sensor_h0_Ext_sensor_h0	Sensor_h0	→	Ext_sensor_h0
✓ adapta_posicionEje_Sensor_h1_Ext_sensor_h1	Sensor_h1	→	Ext_sensor_h1
✓ adapta_posicionTapa_Sensor_j0_Ext_sensor_j0	Sensor_j0	→	Ext_sensor_j0
✓ adapta_posicionTapa_Sensor_j1_Ext_sensor_j1	Sensor_j1	→	Ext_sensor_j1
✓ adapta_posicionTapa_Sensor_k0_Ext_sensor_k0	Sensor_k0	→	Ext_sensor_k0
✓ adapta_posicionTapa_Sensor_k1_Ext_sensor_k1	Sensor_k1	→	Ext_sensor_k1
✓ Global_Act_Amas_Control_A_mas	Act_Amas	←	Control_A_mas
✓ Global_Act_Amenos_Control_A_menos	Act_Amenos	←	Control_A_menos
✓ Global_Act_Bmas_Control_B_mas	Act_Bmas	←	Control_B_mas
✓ Global_Act_Bmenos_Control_B_menos	Act_Bmenos	←	Control_B_menos
✓ Global_Act_C_mas_Control_C_mas	Act_C_mas	←	Control_C_mas
✓ Global_Act_C_menos_Control_C_menos	Act_C_menos	←	Control_C_menos
✓ Global_Act_D_mas_Control_D_mas	Act_D_mas	←	Control_D_mas
✓ Global_Act_D_menos_Control_D_menos	Act_D_menos	←	Control_D_menos
✓ Global_Act_E_mas_Control_E_mas	Act_E_mas	←	Control_E_mas
✓ Global_Act_E_menos_Control_E_menos	Act_E_menos	←	Control_E_menos
✓ Global_Act_E_parada_Control_E_inicio	Act_E_parada	←	Control_E_inicio
✓ Global_Act_F1_mas_Control_F_mas	Act_F1_mas	←	Control_F_mas
✓ Global_Act_F1_menos_Control_F_menos	Act_F1_menos	←	Control_F_menos
✓ Global_Act_F2_mas_Control_F_mas	Act_F2_mas	←	Control_F_mas
✓ Global_Act_F2_menos_Control_F_menos	Act_F2_menos	←	Control_F_menos
✓ Global_Act_H_mas_Control_Hmas	Act_H_mas	←	Control_Hmas
✓ Global_Act_H_menos_Control_Hmenos	Act_H_menos	←	Control_Hmenos
✓ Global_Act_I1_mas_Control_I_mas	Act_I1_mas	←	Control_I_mas
✓ Global_Act_I1_menos_Control_I_menos	Act_I1_menos	←	Control_I_menos
✓ Global_Act_I2_mas_Control_I_mas	Act_I2_mas	←	Control_I_mas
✓ Global_Act_I2_menos_Control_I_menos	Act_I2_menos	←	Control_I_menos
✓ Global_Act_J_mas_Control_Jmas	Act_J_mas	←	Control_Jmas
✓ Global_Act_J_menos_Control_Jmenos	Act_J_menos	←	Control_Jmenos
✓ Global_Act_K_mas_Control_Kmas	Act_K_mas	←	Control_Kmas
✓ Global_Act_K_menos_Control_Kmenos	Act_K_menos	←	Control_Kmenos
✓ Global_Act_P_mas_Control_P_mas	Act_P_mas	←	Control_P_mas
✓ Global_Act_P_menos_Control_P_menos	Act_P_menos	←	Control_P_menos
✓ Global_Act_Vmas_Control_V_mas	Act_Vmas	←	Control_V_mas
✓ Global_Act_Vmenos_Control_V_menos	Act_Vmenos	←	Control_V_menos

Figura 64. Mapeo de todas las señales del sistema

Anexo II. Accionamientos monoestables

La mayoría de accionadores del sistema real son de tipo monoestable, es decir, cuando se desactiva la señal que activa su movimiento, éstos regresan a su posición inicial automáticamente, pero para que en Siemens® NX12 un accionar vuelva a su posición inicial, es necesario que se reciba una señal que active ese movimiento de retroceso.

Manteniendo el mismo programa de control que en la célula real, es posible crear un diagrama de contactos (Figura 65) que active la señal que inicia el retroceso de los accionamientos monoestables cuando la acción de avance esté desactiva.

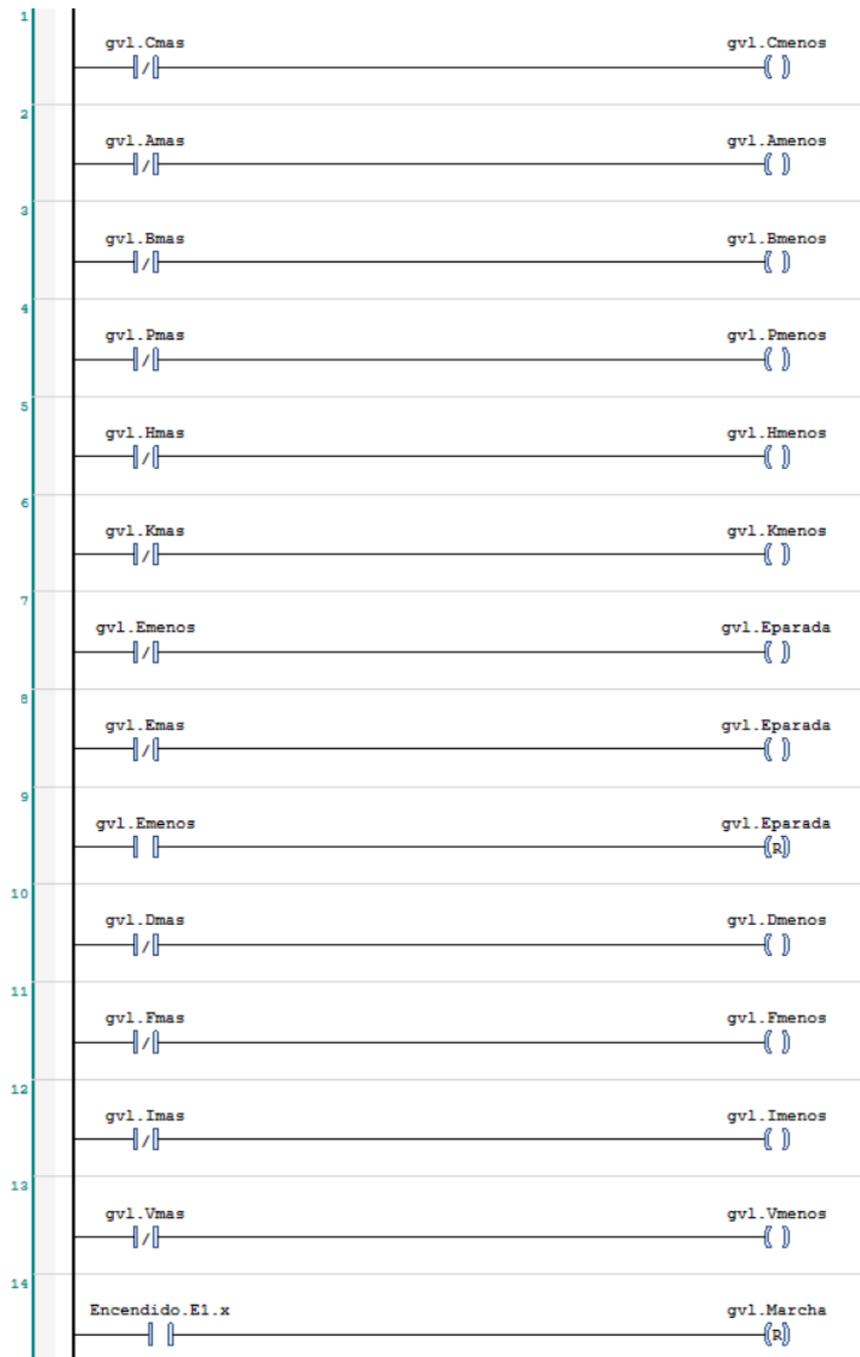


Figura 65. Activación de las señales de retroceso en CODESYS

Referencias

- [1] Universitat Politècnica de València, «UPV,» [En línea]. Available: <https://www.upv.es/titulaciones/MUDFIAC/>. [Último acceso: 24 noviembre 2020].
- [2] Siemens, «Siemens,» [En línea]. Available: <https://www.plm.automation.siemens.com/global/es/products/nx/>. [Último acceso: 20 octubre 2020].
- [3] CODESYS, <<CODESYS Development>>, [En línea]. Available: https://store.codesys.com/engineering/codesys.html?__store=en. [Último acceso: 16 noviembre 2020]
- [4] Universitat Oberta de Catalunya, <<UOC>>, [En línea]. Available: <https://www.uoc.edu/portal/es/news/actualitat/2020/371-combatir-pandemia-gemelos-digitales.html/>. [Último acceso: 12 noviembre 2020]
- [5] M. Gascón, <<20 minutos>>, [En línea]. Available: <https://www.20minutos.es/noticia/4424933/0/gemelos-digitales-investigadores-espanoles-han-logrado-crear-uno-capaz-de-aprender-y-corregirse-a-si-mismo/>. [Último acceso: 12 noviembre 2020]
- [6] Investigate to Innovate, <<ITI>>, [En línea]. Available: <https://www.iti.es/proyectosidi/proyecto-gemelos-digitales-industria-4-0/> [Último acceso: 15 noviembre 2020]
- [7] SAGE Group, <<SAGE Automation>>, [En línea]. Available: <https://www.sageautomation.com/blog/what-is-the-digital-twin-and-how-does-derive-value> [Último acceso: 12 noviembre 2020]
- [8] Siemens, <<Ciudades del futuro>>, [En línea]. Available: <https://ciudadesdefuturo.es/que-es-el-gemelo-digital.php> [Último acceso: 19 noviembre 2020]

- [9] Hightech, <<Wat is de waarde van een Digitale Tweeling?>>, [En línea]. Available: <https://hightechsoftwarecluster.nl/technologie-nl/wat-is-de-waarde-van-een-digitale-tweeling/> [Último acceso: 19 noviembre 2020]
- [10] ISA, Universidad de Oviedo, <<IEC 61131-3: un recurso de programación estándar >>, [En línea]. Available: http://isa.uniovi.es/docencia/ra_marina/cuatrim2/Temas/Intro_IEC_61131-3_Spanish.pdf [Último acceso: 25 noviembre 2020]
- [11] SMC Training, <<SMC>>, [En línea]. Available: <https://www.smctraining.com/es/webpage/indexpage/159> [Último acceso: 19 octubre 2020]
- [12] A. Correcher, <<Manual de prácticas de automatización industrial>>, 2019.
- [13] C. Herrero, “Simulación, control y análisis de sistemas automatizados mediante la integración de NX y Simulink” <<Trabajo Final de Máster>>, septiembre 2019 <http://hdl.handle.net/10251/129498>