

Trabajo de Fin de Máster  
Máster de Ingeniería en Automática, Robótica  
y Telemática

Control de un Robot SCARA Mediante Autómata  
Programable, Integración en Célula de Fabricación  
Flexible

Autor: Manuel Eduardo Monserrat González

Tutor: Luis Fernando Castaño Castaño

Dep. Ingeniería de Sistemas y Automática  
Escuela Técnica Superior de Ingeniería  
Universidad de Sevilla

Sevilla, 2018





Trabajo Fin de Máster  
Máster de Ingeniería en Automática, Robótica y Telemática

# **Control de un Robot SCARA Mediante Autómata Programable, Integración en Célula de Fabricación Flexible**

Autor:

Ing. Manuel Eduardo Monserrat González

Tutor:

Dr. Luis Fernando Castaño Castaño

Doctor Ingeniero Industrial

Dpto. de Ingeniería de Sistemas y Automática

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2018



Trabajo Fin de Máster: Control de un Robot SCARA Mediante Autómata Programable, Integración en Célula de Fabricación Flexible

Autor: Manuel Eduardo Monserrat González

Tutor: Dr. Luis Fernando Castaño Castaño

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2018

El Secretario del Tribunal



*A Jesús Iván Pacheco Moncada.*



# Agradecimientos

---

Me encuentro profundamente agradecido de Dios que escucha las oraciones de mis padres y de ellos por confiar en que cumpla mis metas.

Agradezco a la Escuela Técnica Superior de Ingeniería de la Universidad de Sevilla por abrirse a la formación de profesionales foráneos. A su personal administrativo y en mayor cuantía al personal docente formador, señalando entre todos ellos al Dr. Fernando Castaño quien me ha permitido emplear mis habilidades en este proyecto y ha sabido notar lo que necesitaba conocer para conseguir llevarlo a término.

Finalmente, a Carmen Monserrat González, ¡no hay palabras hermana!

*Manuel Monserrat*

*Sevilla, 2018*



# Resumen

---

En este proyecto se ha realizado la integración de la parte motorizada de un brazo robótico manipulador tipo SCARA a una célula de fabricación empleada para fines docentes.

Se parte de una estructura de un robot manipulador SONY SRX 4 CH en la que su unidad de control se encontraba averiada, así como el motor del eje Z, estando operativo el resto del equipo. Siendo inviable la reparación de los elementos mencionados, se plantea como objetivo del proyecto dar una solución alternativa para seguir operando con el robot e integrarlo en la célula de fabricación.

El proyecto se desarrolla en distintas fases:

- Estudio detallado del robot SONY SRX 4CH en lo que respecta a sus circuitos de potencia y seguridad.
- Planteamiento de soluciones de interconexión hardware entre el robot y un autómata programable (M340 de Schneider Electric).
- Soluciones hardware para sustituir el motor de eje Z averiado por un motor paso a paso.
- Implementación de las soluciones hardware planteadas y pruebas de funcionamiento.
- Desarrollo de un conjunto de funciones que permitan operar con el equipo desde el autómata programable y disponer de un interfaz hombre máquina.

Como detalles de las soluciones de hardware cabe mencionar:

- El uso de salidas analógicas del autómata para el control de los motores de corriente continua originales del robot SONY a partir de módulos de amplificación.
- La inclusión de un driver para el motor paso a paso añadido.
- El uso de tarjetas de contaje rápido (EHC200) para hacer la lectura de las señales de los encoder de los motores originales.

Respecto a las soluciones de software se han empleado las modalidades de programación agrupadas en la normativa IEC-61131-3, que son sustentadas por el software Unity Pro que es usado en la configuración de programas del autómata M340.

Se han creado funciones para el gobierno de las unidades que motorizan las articulaciones del robot. De las que se ha hecho una descripción detallada del proceso de identificación de la naturaleza del elemento motriz y la unidad de potencia que lo opera y como estas se han enlazado al autómata.

Adicionalmente, se ha incluido un conjunto de indicaciones para el uso de los bloques de funciones creados y la interfaz con la que el usuario puede manipular el robot.

# Abstract

---

In this project the integration of the motorized part of a robotic arm manipulator type SCARA to a manufacturing cell used for teaching purposes has been made.

It starts from a structure of a robot manipulator SONY SRX 4 CH in which its control unit was broken, as well as one of the motors of its Z axis, the rest of the equipment being operative. If the repair of the aforementioned elements is unfeasible, the objective of the project is to provide an alternative solution to continue operating with the robot and integrate it into the manufacturing cell.

The project is developed in different phases:

- Detailed study of the SONY SRX 4CH robot with regard to its power and safety circuits. Approach of solutions of hardware interconnection between the robot and a programmable automaton (M340 of Schneider Electric).
- Hardware solutions to replace the broken Z-axis motor with a stepper motor.
- Implementation of the proposed hardware solutions and performance tests.
- Development of a set of functions that allow to operate with the equipment from the programmable automaton and have a human machine interface.

Details of the hardware solutions include:

- The use of analog outputs from the PLC for the control of the original DC motors of the robot SONY from amplification modules.
- The inclusion of a driver for the stepper motor added.
- The use of fast counting cards (EHC200) to read the signals from the encoders of the original engines.

With regard to software solutions, the programming modalities grouped in the IEC-61131-3 standard have been used, which are supported by the Unity Pro software that is used in the configuration of the M340 PLC programs.

Functions Blocks have been created for the government of the units that motorize the joints of the robot. Of which a detailed description has been made of the process of identification of the nature of the motor element and the power unit that operates it and how they have been linked to the automaton.

Additionally, a set of instructions has been included for the use of the created function blocks and the interface with which the user can manipulate the robot.

# Índice

---

<b>Agradecimientos</b>	<b>ix</b>
<b>Resumen</b>	<b>xi</b>
<b>Abstract</b>	<b>xii</b>
<b>Índice</b>	<b>xiii</b>
<b>Índice de Tablas</b>	<b>xv</b>
<b>Índice de Figuras</b>	<b>xvii</b>
<b>1 Introducción</b>	<b>1</b>
<b>2 Proceso a Controlar</b>	<b>3</b>
<b>3 Especificaciones del Sony SRX 4CH</b>	<b>5</b>
3.1 <i>DIMENSIONES DEL SRX 4CH</i>	7
3.2 <i>EL LAZO DE CONTROL</i>	9
3.3 <i>LA PUESTA A HOME Y EL AVANCE DE LAS ARTICULACIONES</i>	9
<b>4 Descripción del Autómata</b>	<b>11</b>
4.1 <i>EL BASTIDOR BMX XBP 0800</i>	11
4.2 <i>EL MÓDULO DE ALIMENTACIÓN BMX CPS 2000</i>	12
4.3 <i>EL PROCESADOR BMX P34 2020</i>	12
4.4 <i>EL MÓDULO DE ENTRADAS / SALIDAS BINARIAS BMX DDM 16022</i>	13
4.5 <i>EL MÓDULO DE ENTRADAS / SALIDAS BINARIAS BMX DDM 3202 K</i>	14
4.6 <i>EL MÓDULO DE CONTAJE RÁPIDO BMX EHC 0200</i>	16
4.6.1 <i>LOS CONECTORES DE 16 PINES DEL BMX EHC 0200</i>	18
4.6.2 <i>EL CONECTOR DE 10 PINES DEL BMX EHC 0200</i>	19
4.6.3 <i>MODALIDADES DE OPERACIÓN DEL MÓDULO</i>	20
4.7 <i>EL MÓDULO DE SALIDAS ANALÓGICAS BMX AMO 0210</i>	22
<b>5 Variables y Señales del Sistema</b>	<b>25</b>
5.1 <i>VINCULACIÓN DE SEÑALES A LOS MÓDULOS DEL AUTÓMATA.</i>	27
5.2 <i>SEÑALES DE LA CÉLULA DE FABRICACIÓN FLEXIBLE.</i>	29
<b>6 Funcionamiento del Automatismo</b>	<b>39</b>
6.1 <i>LA PUESTA A HOME (HOMING PROCESS)</i>	40
6.2 <i>CONTROL DE POSICIÓN</i>	40
6.3 <i>CONTROL DE VELOCIDAD.</i>	40
6.4 <i>REGISTRO DE POSICIÓN</i>	41
6.5 <i>INICIO DE MOVIMIENTO</i>	41
6.6 <i>DESPLAZAMIENTO ARTICULAR</i>	41

6.7	<i>DESPLAZAMIENTO ESPACIAL</i>	41
6.8	<i>APERTURA Y/O CIERRE DE LA PINZA</i>	41
6.9	<i>FUNCIÓN COGE Y COLOCA (PICK AND PLACE)</i>	42
<b>7</b>	<b>Estructura del Programa</b>	<b>43</b>
7.1	<i>DESCRIPCIÓN DE LA GUÍA GEMMA.</i>	45
7.1.1	<i>LOS ESTADOS DE LA GUÍA GEMMA.</i>	46
<b>8</b>	<b>Conexión de Drivers de Potencia</b>	<b>49</b>
8.1	<i>LOS SERVOMOTORES.</i>	49
8.2	<i>EL MOTOR A PASOS.</i>	51
<b>9</b>	<b>Programa del Autómata</b>	<b>55</b>
<b>10</b>	<b>Resultados</b>	<b>59</b>
10.1	<i>MEJORAS A IMPLEMENTAR</i>	62
<b>11</b>	<b>Conclusiones</b>	<b>63</b>
<b>12</b>	<b>Anexos</b>	<b>65</b>
12.1.	<i>ANEXO 1: PROGRAMA PRINCIPAL, CONTROL DE PANTALLAS DE OPERADOR</i>	65
12.1.1	<i>ANEXO 1.1: CONTROL DE LA PANTALLA DE OPERADOR, SUP_PANT1</i>	66
12.1.2	<i>ANEXO 1.2: CONTROL DE LA PANTALLA DE OPERADOR, SUP_PANT2</i>	69
12.1.3	<i>ANEXO 1.3: PANTALLAS DE OPERADOR, VISUALIZACIÓN y COMANDO</i>	71
12.2.	<i>ANEXO 2: PROGRAMA PRINCIPAL, PUESTA A HOME</i>	76
12.2.1.	<i>ANEXO.2.1. ACCIÓN PST_HOME</i>	76
12.2.2.	<i>ANEXO.2.2. SECCIÓN GIR_mot</i>	83
12.2.3.	<i>ANEXO.2.3. SECCIÓN TOPE_AVANC</i>	84
12.2.4.	<i>ANEXO.2.4. SECCIÓN SEC</i>	85
12.3.	<i>ANEXO 3: PROGRAMA PRINCIPAL, MOVIMIENTO MANUAL</i>	86
12.3.1.	<i>ANEXO 3.1: CINEMÁTICA DIRECTA</i>	87
12.3.2.	<i>ANEXO 3.2: MOVIMIENTOS ARTICULAR Y CARTESIANO</i>	88
12.3.3.	<i>ANEXO 3.3: FUNCIÓN DF_LIM</i>	89
12.3.4.	<i>ANEXO 3.4: FUNCIÓN RGN_SPC</i>	90
12.3.5.	<i>ANEXO 3.5: CIN_INVERSA</i>	92
12.3.6.	<i>ANEXO 3.6: FATAN2</i>	97
12.3.7.	<i>ANEXO 3.7: FUNCIÓN PINZA</i>	100
12.4.	<i>ANEXO 4: PROGRAMA PRINCIPAL, REGISTRO DE POSICIONES</i>	102
12.5.	<i>ANEXO 5: PROGRAMA PRINCIPAL, ACTUADORES</i>	104
12.5.1.	<i>ANEXO 5.1: CONTROL POSICIÓN Y VELOCIDAD</i>	107
12.5.2.	<i>ANEXO.5.2. CONTROL POSICIÓN Y VELOCIDAD DE MOTOR DC</i>	108
12.5.3.	<i>ANEXO.5.3. CONTROL POSICIÓN Y VELOCIDAD DE MOTOR PASO A PASO</i>	115
12.6.	<i>ANEXO 6: PROGRAMA PRINCIPAL, MOVIMIENTO AUTOMÁTICO</i>	122
12.6.1.	<i>ANEXO 6.1: RUTINA RECOGER Y COLOCAR</i>	125
12.6.2.	<i>ANEXO 6.2: FUNCIÓN DESPLAZA</i>	128
12.7.	<i>ANEXO 7: ESQUEMAS ELÉCTRICOS</i>	132
	<b>Referencias</b>	<b>135</b>

# ÍNDICE DE TABLAS

---

Tabla 3-1. Especificaciones globales del Sony SRX 4CH.	6
Tabla 3-2. Especificaciones eléctricas del Sony SRX 4CH.	6
Tabla 3-3. Ganancias del algoritmo de control inicial del Sony SRX 4CH.	9
Tabla 4-1. Consumo de corriente del módulo de alimentación BMX CPS 2000.	12
Tabla 4-2. Especificaciones del módulo BMX P34 2020.	13
Tabla 4-3. Especificaciones del módulo BMX DDM 16022.	13
Tabla 4-4. Especificaciones del módulo BMX DDM 3202 K.	15
Tabla 4-5. Numeración de conductores bajo la norma DIN 47100.	16
Tabla 4-6. Especificaciones del módulo BMX EHC 0200.	17
Tabla 4-7. Relación de ciclo de servicio en comparación a la frecuencia.	22
Tabla 4-8. Especificaciones del módulo BMX AMO 0210.	22
Tabla 5-1. Lista de variables de entradas y salidas ubicadas en el robot.	27
Tabla 5-2. Lista de entradas empleadas en el módulo BMX DDM 16022.	27
Tabla 5-3. Lista de variables de salidas empleadas en el módulo BMX DDM 16022.	27
Tabla 5-4. Lista de variables de entradas empleadas en el módulo BMX DDM 3202 K.	28
Tabla 5-5. Lista de variables de salidas empleadas en el módulo BMX DDM 3202 K.	28
Tabla 5-6. Lista de variables de entradas y salidas del módulo BMX EHC 0200.	29
Tabla 5-7. Lista de variables de salidas del módulo BMX AMO 0210.	29
Tabla 5-8. Lista de elementos de la Célula de Fabricación Flexible.	31
Tabla 5-9. Lista N° 1 de variables del programa.	32
Tabla 5-10. Lista N° 2 de Variables de programa.	33
Tabla 5-11. Lista N° 3 de variables del programa.	34
Tabla 5-12. Lista N° 4 de variables del programa.	35
Tabla 5-13. Lista N° 5 de variables del programa.	36
Tabla 5-14. Lista N° 6 de variables del programa.	37
Tabla 7-1. Lista de pasos de la Red de Petri que describe el proceso.	43
Tabla 7-2. Lista de pasos y transiciones de la Red de Petri que describen el proceso.	44
Tabla 8-1. Conexiones PLC – Controladora, Controladora – ROBOT, primera articulación.	50
Tabla 8-2. Conexiones PLC – Controladora, Controladora – ROBOT, segunda articulación.	51
Tabla 8-3. Conexiones PLC – Controladora, Controladora – ROBOT, cuarta articulación.	51
Tabla 8-4. Especificaciones del Motor Paso a Paso.	51

Tabla 8-5. Especificaciones del Driver para motores a pasos HY-DIV268N-5.	53
Tabla 8-6. Conexiones PLC – Controladora, Controladora – ROBOT, tercera articulación.	53
Tabla 12-1. Lista de variables del bloque de función SUP_PANT1.	68
Tabla 12-2. Lista de variables del bloque de función SUP_PANT2.	70
Tabla 12-3. Lista de variables del bloque de función FNC_HOME	78
Tabla 12-4. Lista de variables del bloque de función MOV_MAN.	87
Tabla 12-5. Lista de variables del bloque de función DF_LIM.	90
Tabla 12-6. Lista de variables del bloque de función cinInvesa.	93
Tabla 12-7. Lista de variables del bloque de función Atan2.	98
Tabla 12-8. Lista de variables del bloque de función PINZA.	100
Tabla 12-9. Lista de variables del bloque de función REG_POS.	102
Tabla 12-10. Lista de variables del bloque de función ACTUADORES.	105
Tabla 12-11. Lista de variables del bloque de función MOT_RBT.	109
Tabla 12-12. Lista de variables del bloque de función Mv_Mt_Rap.	116
Tabla 12-13. Lista de variables del bloque de función MOV_AUTO.	123
Tabla 12-14. Lista de posiciones de la rutina automática.	125
Tabla 12-15. Lista de variables del bloque de función DESPLAZA.	128

# ÍNDICE DE FIGURAS

---

Figura 3-1. Configuración del Sony SRX 4CH.	5
Figura 3-2. Dimensiones articulares del Sony SRX 4CH.	7
Figura 3-3. Vista tridimensional del Sony SRX 4CH.	7
Figura 3-4. Vista de planta de las dimensiones del espacio de trabajo del Sony SRX 4CH.	8
Figura 3-5. Vista tridimensional de las dimensiones del espacio de trabajo del Sony SRX 4CH.	8
Figura 3-6. Configuración del lazo de control original del Sony SRX 4CH.	9
Figura 3-7. Asignación del sentido de giro de los servomotores del Sony SRX 4CH.	10
Figura 3-8. Configuración articular en la posición de HOME del Sony SRX 4CH.	10
Figura 4-1. PLC Modicon M340 de Schneider Electric empleado en el proyecto.	11
Figura 4-2. Bastidor BMX XBP 0800 para PLC Modicon M340 Schneider Electric.	11
Figura 4-3. Fuente BMX CPS 2000 para PLC Modicon M340 Schneider Electric.	12
Figura 4-4. Procesador BMX P34 2020 para PLC Modicon M340 Schneider Electric.	12
Figura 4-5. Diagrama de conexión del BMX DDM 16022 Modicon M340 Schneider Electric.	14
Figura 4-6. Módulo BMX DDM 16022 instalado en el Modicon M340 Schneider Electric.	14
Figura 4-7. Diagrama de conexión del módulo BMX DDM 3202 K Schneider Electric.	15
Figura 4-8. Módulo BMX DDM 3202 K instalado en el Modicon M340 Schneider Electric.	16
Figura 4-9. Módulos BMX EHC 0200 fijados en el PLC M340 Modicon Schneider Electric.	17
Figura 4-10. Diagrama de conexión del bloque de entradas del BMX EHC 0200.	18
Figura 4-11. Diagrama de conexión del bloque de entradas del BMX EHC 0200 con sensores.	18
Figura 4-12. Diagrama de conexión del bloque de entradas del BMX EHC 0200 con encoder.	19
Figura 4-13. Diagrama de conexión del bloque de salidas del BMX EHC 0200.	19
Figura 4-14. Diagrama de conexión del bloque de salidas del BMX EHC 0200 con actuadores.	19
Figura 4-15. Capturas de entradas rápidas en cuadratura.	20
Figura 4-16. Salida del bloque BMX EHC 0200 en modulación de ancho de pulsos.	22
Figura 4-17. Terminales del Módulo BMX AMO 0210.	23
Figura 4-18. Módulos BMX AMO 0210 fijados en el PLC M340 Modicon Schneider Electric.	23
Figura 5-1. Conectores de cables de la unidad de potencia al robot.	25
Figura 5-2. Esquema de conexión de la interfaz electrónica.	25
Figura 5-3. Disposición de amplificadores darlington en el PCB.	26
Figura 5-4. Ubicación de los sensores y actuadores del SONY SRX 4CH.	26
Figura 5-5. Ubicación de elementos de la Célula de Fabricación Flexible.	30

Figura 5-6. Proceso de Depuración de Variables del Programa.	32
Figura 6-1. Consola de programación original Teaching Pendant.	39
Figura 6-2. Dimensiones de la pinza de sujeción de piezas SMC MHQ2-20D.	42
Figura 6-3. Sistema neumático de la pinza de sujeción de piezas simulado en Festo Fluidsim.	42
Figura 7-1. Red de Petri del proceso a automatizar	43
Figura 7-2. Diagrama de estados de la Guía GEMMA.	45
Figura 8-1. Driver para motores DC con escobillas Advanced Motion Controls.	49
Figura 8-2. Diagrama de conexión de accionamiento de articulaciones rotacionales.	50
Figura 8-3. Dimensiones y bornes del motor paso a paso.	52
Figura 8-4. Driver para motores paso a paso HY-DIV268N-5.	52
Figura 8-5. Diagrama de conexión del accionamiento de la articulación prismática.	53
Figura 9-1. Árbol del proyecto creado para el Modicon M340.	55
Figura 10-1. Manipulador y Unidad de Potencia del SONY SRX 4CH.	59
Figura 10-2. Cableado del arreglo relés e interfaz PLC y Unidad de Potencia.	59
Figura 10-3. Cableado del arreglo de drivers para motores.	60
Figura 10-4. Cuadro eléctrico de la Unidad de Potencia.	60
Figura 10-5. Bloque de Función PWM de Unity Pro.	61
Figura 12-1 . Programa principal. Control de Pantallas de Operador.	65
Figura 12-2. Bloque SUP_PANT1. Sección de programa CONT_PANTALLA1.	66
Figura 12-3 . Bloque SUP_PANT1. Sección de programa CONT_PANTALLA1, segunda parte.	67
Figura 12-4 . Bloque SUP_PANT2. Sección de programa CONT_PANTALLA2.	69
Figura 12-5. Pantallas de operador VISUALIZACIÓN y COMANDO.	71
Figura 12-6. Pantalla de operador VISUALIZACIÓN.	72
Figura 12-7. Pantalla de operador COMANDO.	73
Figura 12-8. Renglón de la pantalla COMANDO para el registro de posiciones.	73
Figura 12-9. Renglón de la pantalla COMANDO para habilitar el método de trabajo.	74
Figura 12-10. Renglón de la pantalla COMANDO para puesta a home y movimientos manuales.	74
Figura 12-11. Renglón de la pantalla COMANDO para la ejecución de movimientos manuales.	74
Figura 12-12. Cambios de pantallas de operador de acuerdo a la guía GEMMA.	75
Figura 12-13. Sección del Programa Principal, Funcion Home.	76
Figura 12-14. Bloque FNC_HOME. Sección de programa PST_HOME.	77
Figura 12-15. Sección de programa PST_HOME, segunda parte.	80
Figura 12-16. Sección de programa PST_HOME, tercera parte.	81
Figura 12-17. Sección de programa PST_HOME, cuarta parte.	82
Figura 12-18 . Sección de programa PST_HOME, quinta parte.	83
Figura 12-19. Sección de programa GIR_mot.	84
Figura 12-20. Sección TOPE_AVANC.	84
Figura 12-21. Sección de programa de supervisión SEC.	85
Figura 12-22. Sección del Programa Principal, Movimiento Manual.	86

Figura 12-23. Bloque Cin_Directa. Sección de programa Cin_Directa.	87
Figura 12-24. Sección del programa S_MAN.	88
Figura 12-25. Bloque DF_LIM. Sección del programa CAL_LIM.	89
Figura 12-26. Bloque RNG_SPC. Sección de programa DEF_LIM1.	90
Figura 12-27. Sección del programa S_MAN, segunda parte.	91
Figura 12-28. Configuraciones articulares a partir de la cinemática inversa.	92
Figura 12-29. Bloque cinInversa. Sección de programa Fn_C_Inv.	94
Figura 12-30. Sección de programa Fn_C_Inv, segunda parte.	95
Figura 12-31. Sección de programa Fn_C_Inv, tercera parte.	96
Figura 12-32. Bloque Atan2. Sección de programa FATAN2.	97
Figura 12-33. Sección del programa S_MAN, tercera parte.	98
Figura 12-34. Sección del programa S_MAN, cuarta parte.	99
Figura 12-35. Sección del programa S_MAN, quinta parte.	100
Figura 12-36. Bloque PINZA. Sección de programa ACTUADORPINZA.	101
Figura 12-37 . Sección del Programa Principal, Registro de Posiciones.	102
Figura 12-38. Bloque de REG_POS y Sección del programa GUARDAPOSICION.	103
Figura 12-39 . Sección del Programa Principal, Actuadores.	104
Figura 12-40. Sección de programa ACTUADORES.	106
Figura 12-41 . Sección de programa ACTUADORES, segunda parte.	107
Figura 12-42. Sección de programa MOT_SCARA.	108
Figura 12-43. Sección de programa MOT_SCARA, segunda parte.	110
Figura 12-44. Sección de programa MOT_SCARA, tercera parte.	111
Figura 12-45. Perfil de velocidad a seguir por el motor.	112
Figura 12-46. Sección de programa MOT_SCARA, cuarta parte.	112
Figura 12-47. Sección de programa MOT_SCARA, cuarta parte.	113
Figura 12-48. Sección de programa MOT_SCARA, quinta parte.	114
Figura 12-49. Sección de programa MOT_SCARA, sexta parte.	115
Figura 12-50. Sección de programa ACTUADORES, tercera parte.	116
Figura 12-51. Sección de programa MV_MT_RP.	117
Figura 12-52. Sección del programa MV_MT_RP, segunda parte.	118
Figura 12-53 . Sección del programa MV_MT_RP, tercera parte.	119
Figura 12-54 . Sección del programa MV_MT_RP, cuarta parte.	120
Figura 12-55 . Sección de programa ACTUADORES, cuarta parte.	120
Figura 12-56 . Sección de programa ACTUADORES, quinta parte.	121
Figura 12-57. Sección del Programa Principal, MOV_AUTO.	122
Figura 12-58. Sección del Programa Principal, MOV_AUTO, segunda parte .	122
Figura 12-59. Sección de programa S_AUTO.	124
Figura 12-60. Configuración de posiciones de la secuencia automática	125
Figura 12-61. Sección de programa S_AUTO, segunda parte.	126

Figura 12-62 . Sección de programa S_AUTO, tercera parte.	127
Figura 12-63. Bloque DESPLAZA. Sección de programa MOVER.	129
Figura 12-64. Sección de Programa S_AUTO, cuarta parte.	129
Figura 12-65. . Sección de Programa S_AUTO, quinta parte.	130
Figura 12-66. Sección de programa de transiciones MODO2, TR29 y TR36 de rutina PCK_N_PLC.	131
Figura 12-67. Diagrama Unifilar Original del SONY SRX 4CH.	132
Figura 12-68. Diagrama Unifilar de Interfaz Electrónica con Darlington.	133

# 1 INTRODUCCIÓN

---

El laboratorio L1-PB-E2 perteneciente al Instituto de Robótica y Automática de la Escuela Superior de Ingenieros de la Universidad de Sevilla, cuenta con una estación para la enseñanza del manejo de sistemas integrados de manufactura que es denominada Célula de Fabricación Flexible, donde los estudiantes acumulan experiencia en las tareas de automatización industrial en un sistema que posee alta similitud a los procesos de un entorno productivo real.

Por esta razón sobre ella se han realizado varios proyectos de carrera con el propósito de integrar nuevos dispositivos, actualizar o habilitar los ya dispuestos en la estación y sustentar todo el proceso sobre la plataforma automatizada, motivo por el cual hayan sido utilizados para consulta durante la ejecución de este trabajo final de máster, en miras de seguir consecuentemente la metodología de trabajo que se ha empleado en la estación previamente.

En la misma célula de fabricación, específicamente en la estación de robots se encontraba un manipulador SONY modelo SRX - 4CH de configuración morfológica SCARA que había estado inoperante por un largo tiempo, debido a que su unidad de control se deterioró. La intención de este proyecto de automatización es proponer el empleo de un controlador lógico programable (PLC) Modicon M340 que actúe sobre el robot, para habilitar el movimiento de la cadena articular con instrucciones realizadas mediante el software UNITY PRO de Schneider Electric desarrollado para realizar la configuración de este modelo de autómatas.

Adicionalmente y sobre la misma tecnología, implementar una interfaz para el manejo desde las terminales de la red implementada bajo el protocolo TCP/IP que se emplea para la comunicación de las diferentes secciones de la planta. Esta interfaz ha de emular al mando remoto original que se enlazaba a la controladora del SCARA, como terminal remoto de configuración de tareas a realizar por el manipulador. De esta manera el robot, al final de este proyecto, puede emplearse en las operaciones de traslado de piezas entre las demás secciones de la estación que se encuentren dentro de su espacio de trabajo.

Sin embargo, a diferencia de la célula de fabricación flexible, sólo se tenía un trabajo previo realizado en el robot, realizado en el año 1997, fue llevado a cabo para enlazar la unidad de control al ordenador, para ejecutar instrucciones por medio del puerto serie. De esta manera el SCARA se coordinaba desde un programa creado en Lenguaje C, que era superior en potencialidades al programa LUNA que se empleaba en la programación del robot.

Indudablemente al ya no contar con la controladora original, el nuevo planteamiento se hace novedoso y resulta de un punto de partida para la creación de subsecuentes mejoras que puedan sustentarse sobre este proyecto como base.

Ha sido esta razón por la cual este proyecto orienta sus objetivos a:

- Realizar un programa en UNITY PRO para el manejo del robot SRX 4CH de Sony, que permita configurar tareas de manipulación de objetos en la Célula de Fabricación Flexible del laboratorio L1-PB-E2 de la Escuela Técnica Superior de Ingenieros de la Universidad de Sevilla.
- Crear secciones para el control clásico de la posición y el seguimiento de un perfil de velocidad en el movimiento de las articulaciones en el espacio de trabajo del robot SCARA.
- Crear secciones de programa para la realización de una puesta a punto inicial, Homing Process, para habilitar el uso del manipulador con el debido conocimiento de la ubicación en el espacio de la tarea.
- Crear secciones de programa para el registro de configuraciones articulares a adoptar por el robot.
- Desarrollar una Interfaz Hombre Máquina para permitir a los usuarios de la estación, la realización de ajustes de posicionamiento de los eslabones y articulaciones.
- Crear un proceso completo de recogida y colocación de objetos.



## 2 PROCESO A CONTROLAR

---

El proceso estudiado, como se ha mencionado anteriormente, se trata de un robot SCARA (Selective Compliant Articulated Robot Arm, brazo robótico articulado de respuesta selectiva) de cuatro grados de libertad, del tipo Rotacional, Rotacional, Prismático, Rotacional, que contiene una pinza neumática de dos dedos para asir piezas que tengan unos veinte (20) milímetros de diámetro aproximadamente.

Cada una de las articulaciones rotacionales se motoriza con un motor DC (corriente directa) que tiene un encoder acoplado al eje de rotación, para medir el avance del giro por medio del número de pulsos que este genera al desplazarse el eslabón. Por su parte la articulación prismática está adosada a un motor paso a paso que hace girar un tornillo sin fin y de esta manera realiza el desplazamiento longitudinal del efector, que está aritméticamente relacionado al número de pasos que el motor gira por vuelta y al número de vueltas que su eje complete.

La estrategia seguida en el proyecto se basa en el empleo de un Controlador Lógico Programable (PLC) que incluye módulos de contaje rápido en el autómeta y de generación de señales analógicas de 0 a 10 voltios, con la finalidad de referenciar en tiempo real desde los encoders, el espacio de trabajo del robot y emplear las salidas analógicas para controlar la magnitud de la tensión que se aplica a los servomotores y de la misma manera emplear la salida de alta frecuencia en la generación de las señales de actuación sobre el motor a pasos.

Con la intención de conocer con más detalle el proceso y de esta manera lograr sustentar eficientemente el procedimiento de automatización del robot, en los apartados subsecuentes se realizará una descripción exhaustiva de los elementos que conforman el sistema. Es importante señalar que este proyecto está orientado a la implementación de estrategias de control sobre dispositivos ya instalados, por lo tanto, la ruta a seguir es conocer las propiedades de lo ya dispuesto para el SCARA, en el afán de darle el mejor empleo posible, extraer su máxima utilidad y engranarlos para conseguir un alto performance en la utilización del manipulador.



### 3 ESPECIFICACIONES DEL SONY SRX 4CH

Al robot SCARA SRX 4CH desarrollado por la empresa SONY, construido a lo largo de la serie de fabricación del año 1987, le ha sido posible implementar las mejoras contempladas en este proyecto de automatización, gracias a la documentación técnica del manipulador, que está bajo el resguardo del personal técnico del laboratorio L1-PB-E2 de la Escuela Técnica Superior de Ingenieros.

Los documentos a los cuales se hace referencia están conformados por la serie de manuales que el fabricante ofreció al momento de su comercialización, que describen las condiciones de instalación de la unidad, los procedimientos de operación, el modo de empleo, las estrategias de corrección de problemas y la prestación de servicio recomendados para el conjunto del robot, y que se han seguido con el afán de conseguir la mayor similitud a los escenarios originales del proceso.

De las instrucciones de operación diseñadas para el correcto uso del SCARA, se ha aprovechado la buena descripción de las dimensiones del robot, para definir el tamaño del espacio de trabajo que el manipulador es capaz de contener en sus movimientos y desde dicho espacio de tarea, restringir las configuraciones articulares a adoptar en el posicionamiento del robot y en el seguimiento de trayectorias que este ha de trazar en cada uno de sus desplazamientos.

Asimismo, se toma el procedimiento de composición de tareas a ejecutar por el manipulador desde el terminal remoto de programación (teaching pendant). Cabe destacar que dicho procedimiento, dadas las condiciones de hardware en que ha sido encontrado el robot, no resultó posible de emplear en su operación. Sin embargo, el mismo ha permitido seguir una analogía entre la manera en la ejecución inicial de los comandos y lo que se perseguía configurar para el nuevo modo de empleo del SRX 4CH y que se ha de ejecutar desde la estación de supervisión del SCADA que rige el funcionamiento de la célula de fabricación flexible o desde uno de los terminales de operador ubicado en los cuadros eléctricos de potencia y control dispuestos en la cercanía del manipulador.

Dentro de las descripciones de uso se pudieron encontrar principalmente los atributos nominales que posee el robot, al mismo tiempo, en esta se detalla detenidamente la unidad controladora del manipulador.

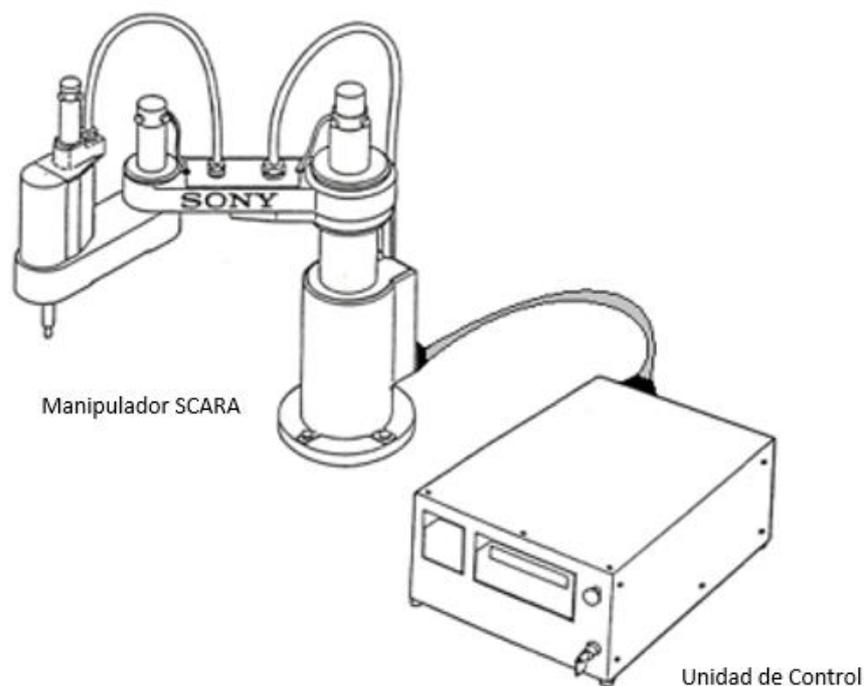


Figura 3-1. Configuración del Sony SRX 4CH.

La información además de señalar la configuración del sistema, que es mostrada en la figura anterior, contiene las características de desempeño del conjunto y las recoge en la siguiente tabla.

Tabla 3-1. Especificaciones globales del Sony SRX 4CH.

<b>Features</b>	
<b>High speed</b>	Maximun speed: 5.2 m/sec.
<b>Precise positioning</b>	Positioning repeatability: within $\pm 0.05$ mm.
<b>Compact desktop controller</b>	Dimensions (W*D*HD) = 430*450*235
<b>Large data storage capacity</b>	Up to 1600 points (variable length splitting into 16 types possible).
<b>16 – bit CPU</b>	High – speed arithmetic operation and versatile functions.
<b>Additional visual systems</b>	Such visual equipment using CCD (charge – coupled device) cameras can easily be conected to the robot.
<b>CP control capability</b>	Linear and circular interpolating functions are made avaliable

De la misma manera, ha sido incluida también las propiedades del hardware y del software que ofrecía el robot y se añaden los gráficos demostrativos de conexionado de elementos, se registran los valores nominales de alimentación, consumo de potencia, condiciones del entorno donde se emplearía el conjunto, la puesta a tierra y el aislamiento del sistema, entre otros. Estas magnitudes se pueden observar como en el caso anterior en la siguiente tabla.

Tabla 3-2. Especificaciones eléctricas del Sony SRX 4CH.

<b>Electrical Specifications</b>	
<b>Supply voltaje</b>	100/200 Vac $\pm 10\%$ 50/60Hz.
<b>Power consumption</b>	1.5 KVA.
<b>Instantaneous power failure, backup – time</b>	Operation is not interrupted when instanteneous power failure is recovered witjin 20 msec.
<b>Insulation resistance</b>	Max. 20 M $\Omega$ between the housing and primary power supply.
<b>Noise resistance</b>	1000 V <sub>p-p</sub> 1 $\mu$ sec (between supply and chassis when measured using a noise simulator).
<b>Operating temperature</b>	0° to 40° Celsius.
<b>Ambient conditions</b>	No corrosive gas nor condensation.
<b>Grounding</b>	Class AWG 14 grounding and higher.

Por otro lado, de la misma documentación técnica se ha hecho uso de la valiosa descripción de las funciones que se ejecutaban con ayuda del ordenador y la estación remota (teaching pendant) que conformaban el sistema original, con lo que se ha pretendido imitar en el automatismo la mayoría de estas operaciones.

Finalmente, se han seguido las instrucciones para instalar apropiadamente el conjunto de elementos del manipulador, se han observado las recomendaciones para realizar el mantenimiento de la unidad y como corregir los problemas que se presentaban en la operación del manipulador SCARA.

Ciertamente estas condiciones señaladas en ambos documentos han cambiado en el desarrollo del proyecto, sin embargo, la mejor manera de orientar un proceso de automatización centrado en el reemplazo de elementos, cuando el proyecto esté justificado por el deterioro o renovación de los componentes que conformaban el sistema inicial, radica en enfocar el proyecto en la búsqueda de la familiaridad del usuario en la operabilidad de dicho sistema anterior, en otras palabras, conseguir que se mantengan las mismas condiciones de manejo que se realizaban en los métodos de operación que antecedieron la puesta en ejecución de la mejora, sabiendo que estos métodos cumplían a cabalidad las tareas solicitadas.

Siendo en el procedimiento de mejora optimizadas aquellas acciones que se han observado como limitantes de lograr alcanzar la máxima eficiencia y en pro de asegurarla se han de añadir más cualidades, opciones y herramientas que al momento tecnológico de la implementación inicial no se tenían a disposición y que se tiene la certeza en que llegarán a mejorar el desempeño del sistema mediante el empleo de las nuevas potencialidades tecnológicas.

Para tal fin, lo primordial es continuar con la descripción de los aspectos recopilados en esta investigación que se consideran realmente significativos y por consiguiente se presentan en este documento.

### 3.1 DIMENSIONES DEL SRX 4CH

Al evaluar un manipulador las principales cualidades que se tienen a consideración son la capacidad de posicionar con exactitud el efector terminal, las características de los objetos a acarrear como sus dimensiones, las texturas, la cantidad de masa, entre otras. De igual manera se precisa la disponibilidad de manejar el volumen de trabajo en el tiempo en que este se produce y las dimensiones que han de tener la cadena de eslabones del robot para poder alcanzar la ubicación en el espacio en que los objetos se encuentran.

Esta definición del espacio de trabajo es propia de la morfología del robot. En la figura 2 tomada del libro de fundamentos de la manipulación mecánica robótica [5], se acota todas las dimensiones del SCARA dispuesto en la célula de fabricación flexible y a partir de esta descripción mecánica estructural se ha levantado una representación tridimensional que permita tener una referencia visual completa del robot y que está mostrada en la figura 3-2.

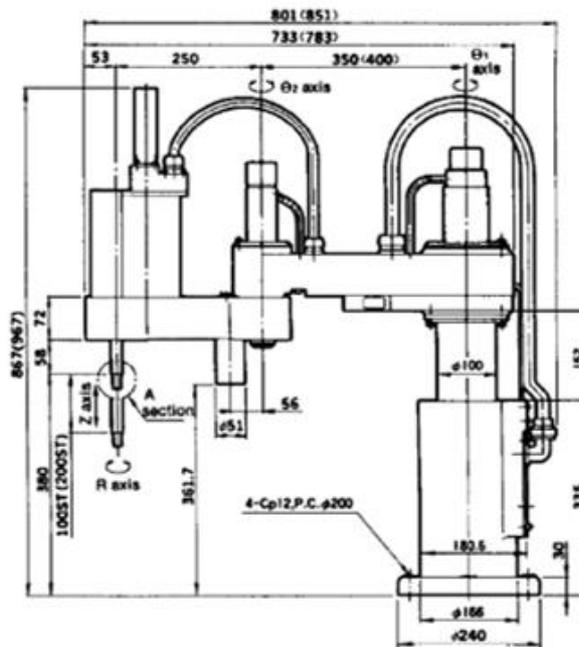


Figura 3-2. Dimensiones articulares del Sony SRX 4CH.

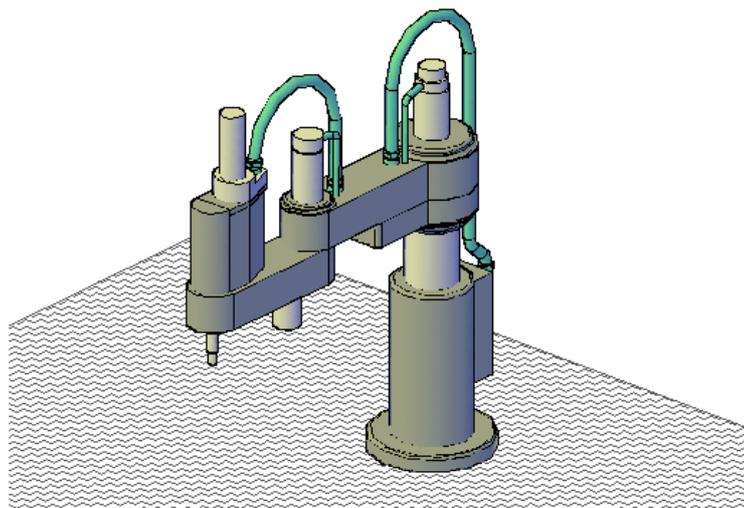


Figura 3-3. Vista tridimensional del Sony SRX 4CH.

Siguiendo el mismo planteamiento se han tomado los planos bidimensionales del espacio de trabajo del robot, para reflejarlo como en la sección anterior haciendo uso del levantamiento tridimensional. A continuación, ambos esquemas son presentados en la figura 3-4 y 3-5 respectivamente.

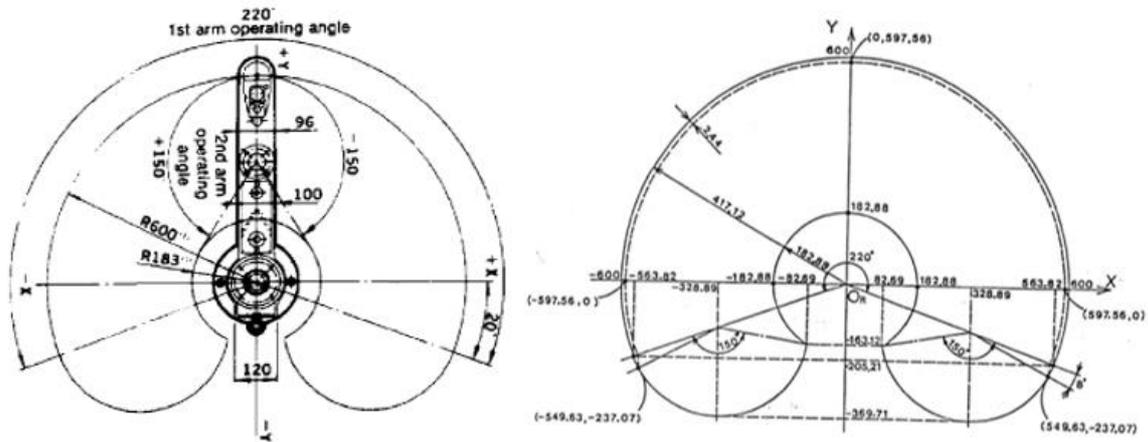


Figura 3-4. Vista de planta de las dimensiones del espacio de trabajo del Sony SRX 4CH.

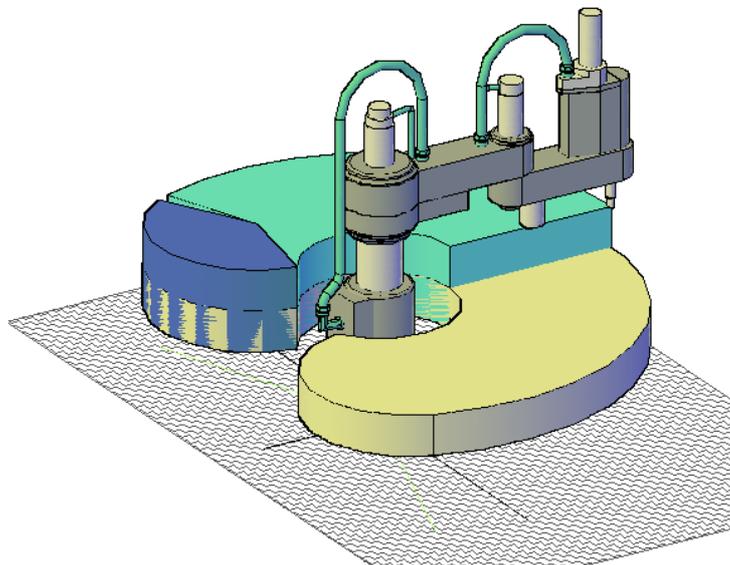


Figura 3-5. Vista tridimensional de las dimensiones del espacio de trabajo del Sony SRX 4CH.

En la figura anterior y que ya ha sido referida previamente, se puede apreciar como existen secciones tronizadas en el espacio. Y con ello se ha querido ilustrar el espacio de trabajo que es cubierto por cada uno de los eslabones del robot, donde el primer enlace que está adosado a la articulación  $\theta_1$ , abarca la sección del espacio de trabajo señalada en azul claro, es capaz de desplazarse en un arco de  $220^\circ$  (de  $70^\circ$  a  $290^\circ$ ) y posee una longitud de 350 milímetros. Por su parte, el segundo enlace que está adosado a la articulación  $\theta_2$ , envuelve la sección del espacio de trabajo señalada en azul oscuro, puede desplazarse sobre un eje dispuesto en el extremo del primer eslabón, abarcando  $300^\circ$  (de  $30^\circ$  a  $330^\circ$ ), con la posibilidad de trazar una circunferencia externa ante la extensión total del manipulador de 600 milímetros de radio y en su máxima retracción puede trazar una circunferencia interna de 183 milímetros de radio.

En ese mismo orden de ideas, la articulación prismática de los modelos del SONY SRX se han definido a 100 o 200 milímetros (sin incluir la dimensión del efector) de desplazamiento longitudinal vertical. El 4CH que se estudia se desplaza 100 milímetros. Ese conjunto de múltiples posiciones a alcanzar es señalado por el sólido tridimensional de color verde que se encuentra en la figura. De la misma manera la articulación rotacional  $\theta_4$ , permite orientar la pinza entorno al eje de desplazamiento de la articulación prismática, un ángulo total de  $180^\circ$ .

En términos de control de posicionamiento y velocidad el SCARA estaba provisto de un algoritmo de control que es descrito en la siguiente sección

### 3.2 EL LAZO DE CONTROL

El conocimiento de las cualidades espaciales del robot, aunado al hecho de motorizar las articulaciones mediante servomotores, a excepción de la prismática que se gobierna por un motor paso a paso, es posible realizar el planteamiento de diversas estrategias de control sobre las unidades de movimiento.

De partida se conoce que el algoritmo que el SRX 4CH implementaba en su configuración original, era el expuesto en la figura siguiente, que se ha reproducido a partir del manual de operación del robot [17].

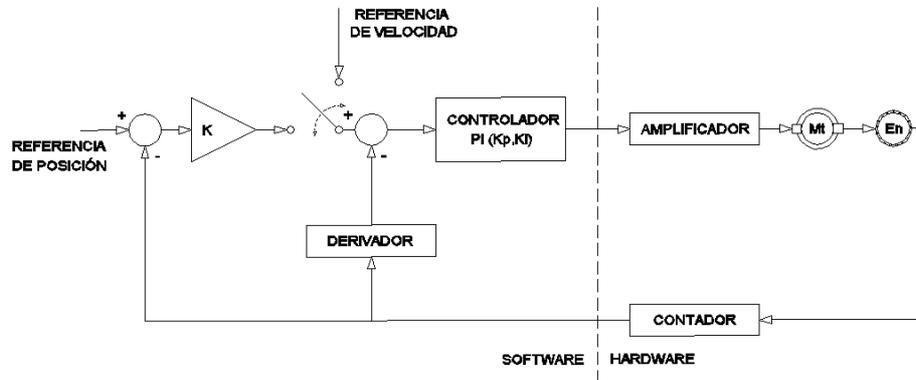


Figura 3-6. Configuración del lazo de control original del Sony SRX 4CH.

En ella se observa un controlador proporcional, integral, derivativo (PID) que conmuta entre los valores de consigna de posición y velocidad, donde además la retroalimentación del lazo depende del encoder que el servo tiene acoplado, cuyos pulsos son totalizados con el empleo de un contador de alta velocidad. El fabricante ha provisto los valores estándar para las ganancias de cada controlador presente en el lazo en la siguiente tabla.

Tabla 3-3. Ganancias del algoritmo de control inicial del Sony SRX 4CH.

	$K_i$ ( $K_1$ )	$K_p$ ( $K_2$ )	$K$ ( $K_3$ )
$\theta_1, X$	14	80	15
$\theta_2, Y$	8	40	17
$d_4, Z$	14	30	40
$\theta_3, R$	8	20	35

De esta última se indica a  $K_1$  como la ganancia  $k_p$  que posee el lazo y al ser ajustada se mejoraría el cumplimiento de la respuesta ante la consigna de velocidad del eslabón. Por su parte,  $K_2$  representa la ganancia integral  $k_i$  y estaría siendo empleada en la puesta a punto de la velocidad de respuesta del lazo de control.  $K_3$  por su lado es la ganancia  $K$  y su modificación repercute en la respuesta de posicionamiento del eslabón.

La documentación técnica del robot de la misma manera que ha descrito la estrategia de control incluye una amplia definición de las directrices en el accionar que en términos de operación de la unidad se debían ejecutar y en la sección siguiente se condensan por haber sido seguidas en el desarrollo del procedimiento de automatización del sistema.

### 3.3 LA PUESTA A HOME Y EL AVANCE DE LAS ARTICULACIONES

La realización de los movimientos de la articulación que amerita la energización del motor dependerá de la posición que se desee alcanzar con el efector y está supeditada a la dirección que la articulación deba adoptar para tal fin. Eso amerita la polarización de los terminales del motor de manera de hacer la corriente en un sentido u otro para coordinar el avance adecuado del eslabón. Para conseguirlo se ha asumido la convención que el fabricante acogió y dispuso en la documentación técnica para no apartarse de este señalamiento. El mismo se ilustra en la figura dispuesta a continuación y es empleado para cerciorarse de cumplir el con el test empleado en el proceso de puesta home.

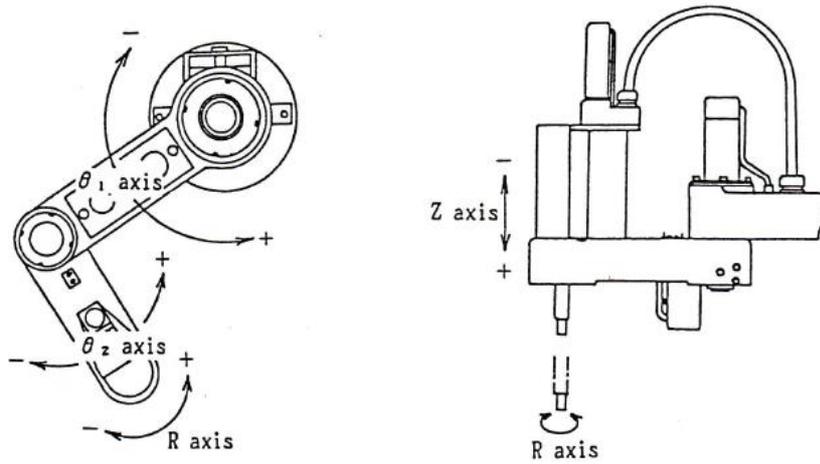


Figura 3-7. Asignación del sentido de giro de los servomotores del Sony SRX 4CH.

Este proceso ha de ser ejecutado al inicio y al final de cada sesión de trabajo que se realice con el SCARA, así se configuran los parámetros de trabajo del robot, se certifica que está operando bajo esos parámetros adecuadamente y al finalizar su empleo se deja en una configuración articular conocida, que le conduce a evitar iniciar desde una ubicación espacial no referenciada. Esto además conlleva al ahorro en el tiempo de puesta a punto, de consumo de energía y de los elementos que integran el manipulador. En la figura que es mostrada a continuación se expresa la configuración espacial que las articulaciones deben alcanzar en la posición de home, que se ha indicado previamente con el arreglo  $[\theta_1, \theta_2, d_3, \theta_4] = [0^\circ, 0^\circ, 0 \text{ mm}, 0^\circ]$ .

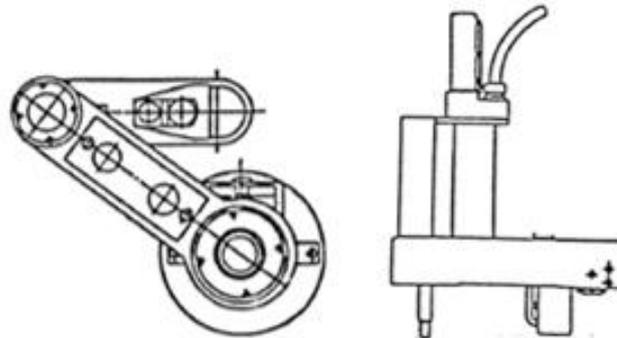


Figura 3-8. Configuración articular en la posición de HOME del Sony SRX 4CH.

## 4 DESCRIPCIÓN DEL AUTÓMATA

Como ya se mencionó anteriormente, en la estación de la Célula de Fabricación Flexible que contiene el SONY SRX 4CH, se encuentra instalado un Controlador Lógico Programable Modicon M340 de Schneider Electric para gobernarlo, este autómata es capaz de trabajar dentro del rango de temperatura comprendido entre  $0$  y  $60$  °C ( $32...140$  °F) en su versión estándar o en su versión reforzada o endurecida M340H que es apta para un mayor rango de temperatura de  $-25$  a  $70$  °C ( $-13...158$  °F) y soporta entornos químicos severos.

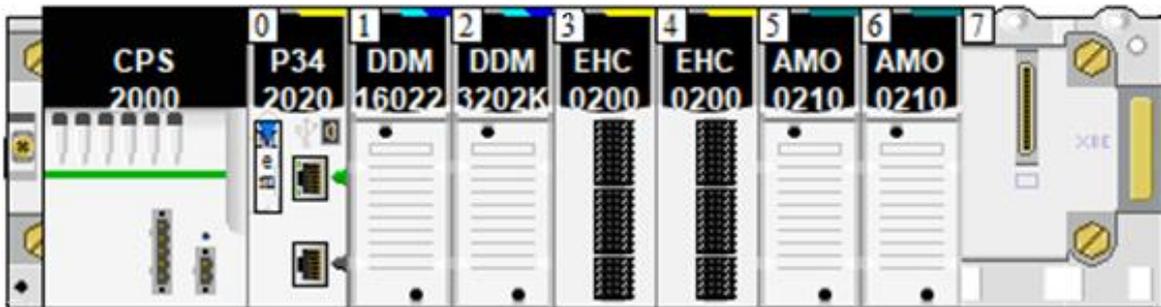


Figura 4-1. PLC Modicon M340 de Schneider Electric empleado en el proyecto.

El autómata dispuesto en la célula posee una configuración modular como la mostrada en el esquema de la figura anterior. Dicha configuración se compone de una fuente de alimentación **BMX CPS 2000**, una unidad de procesamiento **BMX P34 2020**, dos módulos de contaje rápido **BMX EHC 0200**, un módulo de entradas y salidas digitales **BMX DDM 16022** que se encarga de comandar el SCORBOT V5, un módulo binario mixto **BMX DDM 3202 K** y dos tarjetas de salidas analógicas **BMX AMO 0210** que se encuentran montados en un bastidor **BMX XBP 0800**, por lo que aún se cuenta con una posición disponible en el bastidor para posibles expansiones del arreglo.

A continuación, se listan y describen las principales especificaciones y funciones de cada uno de los elementos que conforman el autómata.

### 4.1 EL BASTIDOR BMX XBP 0800

El bastidor BMX XBP 0800 constituye el elemento fundamental de la estación del PLC Modicon M340 que se emplea en el proyecto. Está destinado para la fijación mecánica de todos los módulos de la estación del PLC (módulo de alimentación, procesador, módulos de E/S binarias/analógicas, de contaje rápido u otros módulos que son específicos de cada aplicación).



Figura 4-2. Bastidor BMX XBP 0800 para PLC Modicon M340 Schneider Electric.

Los bastidores, como el mostrado en la figura 4-2, se pueden fijar en armarios, carcasas o paneles. Eléctricamente los bastidores proporcionan la alimentación necesaria para cada uno de los módulos que son dispuestos sobre ellos, de la misma manera proveen las señales de servicio y datos para la totalidad de la estación del PLC. Para el modelo BMX XBP 0800 se admite la fijación de la fuente de alimentación, la unidad de procesamiento y 7

módulos que son del propósito propio de la aplicación a la cual se destina el autómata.

## 4.2 EL MÓDULO DE ALIMENTACIÓN BMX CPS 2000

El módulo de alimentación BMX CPS 2000 que se monta sobre el bastidor BMX XBP 0800 en la primera posición o slot, de izquierda a derecha, es empleado para energizar los demás módulos que son dispuestos en el bastidor a partir del segundo slot. La elección del módulo de alimentación dependerá de la red que alimenta al controlador (corriente alterna o continua) y de la alimentación que precisen los elementos del autómata. Los módulos de alimentación, como el mostrado en la siguiente figura, permiten además desarrollar funciones auxiliares al funcionar como bloques de visualización, como relés de alarma, elementos de restablecimiento y para la alimentación de sensores de 24 VCC (sólo para los módulos abastecidos por una red de corriente alterna).



Figura 4-3. Fuente BMX CPS 2000 para PLC Modicon M340 Schneider Electric.

Por esta razón es recomendable la instalación de un dispositivo de protección al comienzo de la línea de la red que suple de energía a la fuente, que bien pudiera ser un disyuntor, breaker o fusible de recambio. Considerando la corriente nominal de entrada, la corriente de señalización y las características de corriente de alimentación. Siendo para el módulo del PLC del proyecto las que son listadas en la siguiente tabla.

Tabla 4-1. Consumo de corriente del módulo de alimentación BMX CPS 2000.

Módulo de alimentación BMX CPS 2000	Irms de corriente nominal de entrada	Corriente de señalización I (1)	Característica de corriente It	Característica de corriente I2t
A 230 VCA	0,31 A	60 A	0,06 As	2 A2s

## 4.3 EL PROCESADOR BMX P34 2020

El procesador BMX P34 2020 que es instalado en el bastidor, se encarga de gestionar toda la estación del PLC, aloja el programa de usuario y los programas que controlan las interacciones con los módulos de entradas/salidas binarias, las tarjetas de salidas analógicas y los módulos de contaje rápido.

La siguiente figura muestra el arreglo fuente y procesador montados en el bastidor que conforma el autómata de este proyecto.



Figura 4-4. Procesador BMX P34 2020 para PLC Modicon M340 Schneider Electric.

Por otra parte, el procesador tiene la funcionalidad de establecer las comunicaciones del controlador con el entorno, que para este proyecto se encuentran soportadas en el protocolo TCP/IP de Ethernet y así es capaz de interactuar con el resto de los elementos de la planta. En la siguiente tabla se muestra las especificaciones del procesador que se ha encontrado en el arreglo modular.

Tabla 4-2. Especificaciones del módulo BMX P34 2020.

<b>Procesador BMX P34 2020</b>	
Número máximo de entradas y salidas binarias	1.024
Número global máximo de entradas y salidas analógicas	256
Tamaño máximo de memoria	4.096 Kb
Conexión Modbus	Si
Conexión Ethernet Integrada	Si

#### 4.4 EL MÓDULO DE ENTRADAS / SALIDAS BINARIAS BMX DDM 16022

El módulo binario mixto que se describe a continuación está destinado para conmutar las entradas y salidas digitales del SCORBOT V5 de la sección y aunque no tienen relación con el programa del SCARA, la rutina de manejo de esta unidad será compartida en las secciones del SONY, por estar coordinado por el mismo autómata, razón por la cual para este proyecto es importante el conocimiento de sus características.

Los módulos BMX DDM 16022, son tarjetas para el manejo de entradas y salidas booleanas (todo o nada) de 24 VCC conectado a través de un conector de 20 pines, que es configurado con lógica positiva, esto quiere decir, que los 8 canales de entrada reciben corriente de los sensores siendo común el terminal positivo y los 8 canales de salida proporcionan corriente a los preactuadores siendo común el negativo.

Los manuales de usuario que provee Schneider Electric ofrecen una descripción extensa de las especificaciones del módulo de entradas y salidas, sin embargo, para el desarrollo de este proyecto basta con listar las que permiten realizar las conexiones del autómata con su entorno y que son mostradas en la siguiente tabla de especificaciones.

Tabla 4-3. Especificaciones del módulo BMX DDM 16022.

<b>Módulo BMX DDM 16022</b>				
<i>Especificaciones de las entradas</i>		Entradas de lógica positiva de 24 V CC		
<b>Valores de entrada nominal</b>		<b>Tensión</b>	24 VCC	<b>Corriente</b> 3,5 mA
<b>Valores límite de entrada</b>	<b>En 1</b>	<b>Tensión</b>	$\geq 11$ V	<b>Corriente</b> $\geq 2$ mA para $U \geq 11$ V
	<b>En 0</b>	<b>Tensión</b>	5 V	<b>Corriente</b> $\leq 1,5$ mA
<b>Impedancia de entrada</b>		<b>En U nominal</b>	6.8 k $\Omega$	<b>Tipo</b> 3 según IEC 1131-2
<b>Tiempo de respuesta</b>		<b>Típico</b>	4 ms	<b>Máximo</b> 7 ms
<b>Consumo de alimentación de 3,3V</b>		<b>Típico</b>	79 mA	<b>Máximo</b> 111 mA
<b>Consumo del preactuador de 24V</b>		<b>Típico</b>	59 mA	<b>Máximo</b> 67 mA
(sin incluir la corriente de carga)		<b>Potencia disipada</b>	3.7 W sin descenso de temperatura	
<i>Especificaciones de las salidas</i>		Salidas estáticas de 24 V CC con lógica positiva		
<b>Valores nominales</b>		<b>Tensión</b>	24 VCC	<b>Corriente</b> 0,5 mA
<b>Valores de Tensión (ondulación incluida)</b>		De 19 a 30 V (posible hasta 34 V durante 1 hora/día)		
<b>Umbral</b>	<b>Corriente/canal</b>	0,625 A		<b>Corriente/módulo</b> 5 A
<b>Corriente de Fuga</b>	$< 0,5$ mA	<b>Caída de Tensión</b>	$< 1,2$ V	<b>Impedancia de carga</b> Mínima 48 $\Omega$
<b>Consumo de alimentación de 3,3V</b>		<b>Típico</b>	125 mA	<b>Máximo</b> 166 mA
<b>Consumo del preactuador de 24V</b>		<b>Típico</b>	69 mA	<b>Máximo</b> 104 mA
(sin incluir la corriente de carga)		<b>Potencia disipada</b>	3.7 W sin descenso de temperatura	

El módulo BMX DDM 16022 dispone de un bloque de terminales de 20 pines extraíble para la conexión rápida de los ocho (8) canales de entrada y los ocho (8) canales de salida. El diagrama que se ilustra en la siguiente figura permite observar el esquema de cableado que se ha de seguir para la conexión del módulo con sus respectivos sensores y actuadores.

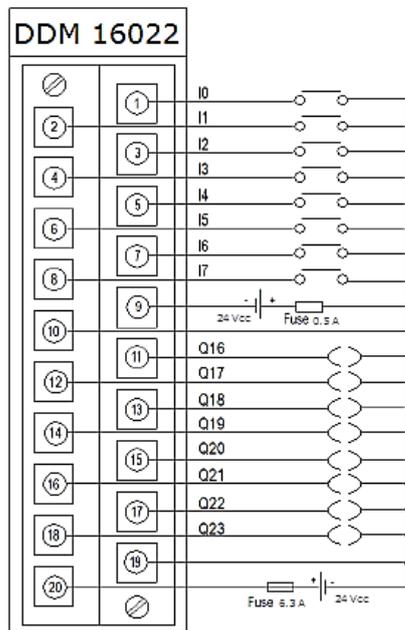


Figura 4-5. Diagrama de conexión del BMX DDM 16022 Modicon M340 Schneider Electric.

La siguiente figura muestra la evolución del arreglo al incluir la tarjeta de manipulación de entradas binarias en el autómata que se ha empleado para el manejo de ambos manipuladores en la sección de la célula de fabricación flexible objeto de este estudio.

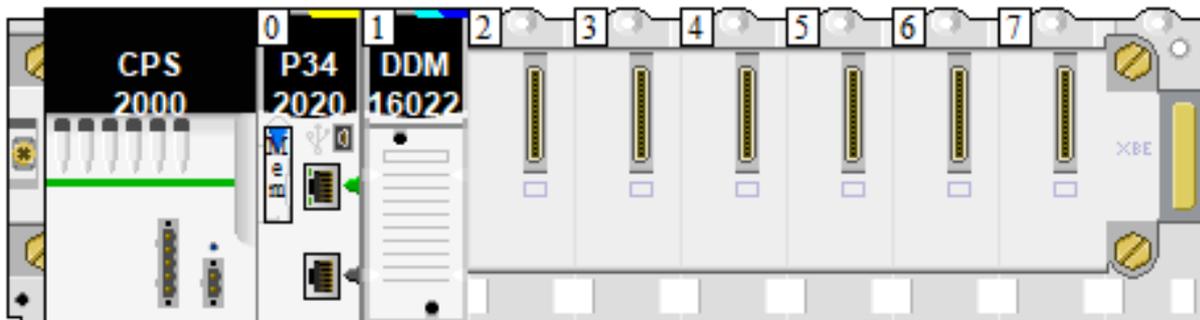


Figura 4-6. Módulo BMX DDM 16022 instalado en el Modicon M340Schneider Electric.

### 4.5 EL MÓDULO DE ENTRADAS / SALIDAS BINARIAS BMX DDM 3202 K

Los módulos **BMX DDM 3202 K** son módulos binarios de 24 VCC conectado a través de un conector de 40 pines, que poseen lógica positiva, esto quiere decir que los 16 canales de entrada reciben corriente de los sensores siendo común el terminal positivo y los 16 canales de salida proporcionan corriente a los preactuadores siendo común el negativo.

Los manuales de Schneider Electric ofrecen una amplia descripción de las especificaciones del módulo de entradas y salidas, sin embargo, es importante ofrecer en el proyecto listadas en la próxima tabla aquellas de naturaleza binaria y que permiten realizar las conexiones del autómata con su entorno.

Tabla 4-4. Especificaciones del módulo BMX DDM 3202 K.

<b>Módulo BMX DDM 3202 K</b>			
<b>Descripción de las entradas</b>		Entradas de lógica positiva de 24 V CC	
<b>Valores de entrada nominal</b>	<b>Tensión</b>	24 VCC	
	<b>Corriente</b>	2,5 mA	
<b>Valores límite de entrada</b>	<b>En 1</b>	<b>Tensión</b>	$\geq 11$ V
		<b>Corriente</b>	$\geq 2$ mA para $U \geq 11$ V
	<b>En 0</b>	<b>Tensión</b>	5 V
		<b>Corriente</b>	$\leq 1,5$ mA
<b>Impedancia de entrada</b>	<b>En U nominal</b>	9,6 k $\Omega$	
<b>Tiempo de respuesta</b>	<b>Típico</b>	4 ms	
	<b>Máximo</b>	7 ms	
<b>Consumo de alimentación de 3,3V</b>	<b>Típico</b>	125 mA	
	<b>Máximo</b>	166 mA	
<b>Consumo del preactuador de 24 V (sin incluir la corriente de carga)</b>	<b>Típico</b>	69 mA	
	<b>Máximo</b>	104 mA	
<b>Descripción de las salidas</b>		Salidas estáticas de 24 V CC con lógica positiva	
<b>Valores nominales</b>	<b>Tensión</b>	24 VCC	
	<b>Corriente</b>	0,1 mA	
<b>Valores de Umbral</b>	<b>Tensión (ondulación incluida)</b>	De 19 a 30 V (posibles 34 V durante 1 hora/día)	
	<b>Corriente/canal</b>	0,125 A	
	<b>Corriente/módulo</b>	3,2 A	
<b>Impedancia de carga</b>	<b>Mínima</b>	220 $\Omega$	
<b>Consumo de alimentación de 3,3V</b>	<b>Típico</b>	125 mA	
	<b>Máximo</b>	166 mA	
<b>Consumo del preactuador de 24 V (sin incluir la corriente de carga)</b>	<b>Típico</b>	69 mA	
	<b>Máximo</b>	104 mA	

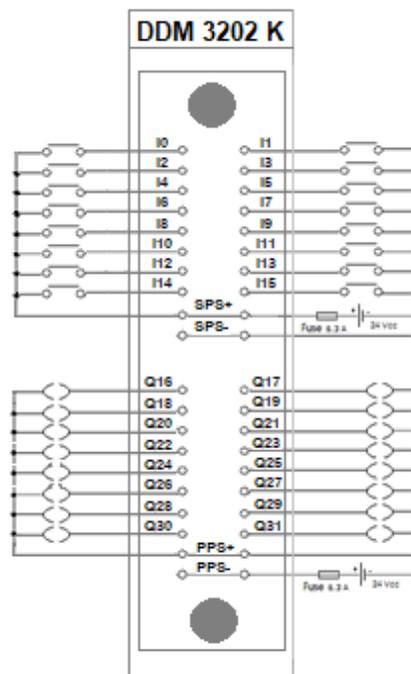


Figura 4-7. Diagrama de conexión del módulo BMX DDM 3202 K Schneider Electric.

De la misma manera que se hiciera en el apartado anterior, mediante el diagrama que se ilustra en la figura precedente se ha logrado observar el esquema de cableado que se ha seguido para la conexión del módulo con sus respectivos sensores y actuadores.

Además, la siguiente ilustración ofrece una visión del acoplamiento del módulo BMX DDM 3202 K con los primeros elementos ya previamente instalados en el autómata.

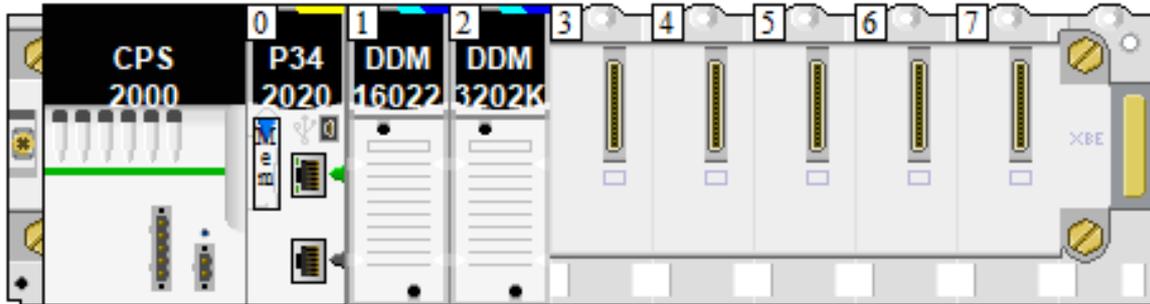


Figura 4-8. Módulo BMX DDM 3202 K instalado en el Modicon M340 Schneider Electric.

Finalmente, es importante señalar que el módulo se enlaza a través del conector BMX FCW 03 que el fabricante ofrece previamente cableado con dos mangueras, una para las entradas y otra para las salidas, que contienen 20 hilos conductores cada una y poseen la numeración según la codificación que se señala en la próxima tabla.

Tabla 4-5. Numeración de conductores bajo la norma DIN 47100.

Codificación de colores para la identificación de conductores DIN 47100			
1	Blanco (WH)	11	Gris-Rosa (GYPK)
2	Marrón (BN)	12	Rojo-Azul (RDBU)
3	Verde (GN)	13	Blanco-Verde (WHGN)
4	Amarillo (YE)	14	Marrón-Verde (BNGN)
5	Gris (GY)	15	Blanco-Amarillo (WHYE)
6	Rosa (PK)	16	Amarillo-Marrón (YEBN)
7	Azul (BU)	17	Blanco-Gris (WHGY)
8	Rojo (RD)	18	Gris-Marrón (GYBN)
9	Negro (BK)	19	Blanco-Rosa (WHPK)
10	Violeta (VT)	20	Rosa-Marrón (PKBN)

## 4.6 EL MÓDULO DE CONTAJE RÁPIDO BMX EHC 0200

Los módulos expertos de contaje rápido BMX EHC 0200 que están instalados en el slot 3 y 4 del bastidor del controlador utilizado para comandar el robot SCARA, que se puede observar cómo han sido instalados en la novena ilustración de esta sección, tienen la función primordial de acoplar mediante los canales de entrada las señales de retroalimentación que entregan los codificadores incrementales a medida que estos giran debido a la rotación de los ejes de los servos a los que han sido mecánicamente acoplados.

Además, se ha tomado provecho de los canales de salida para generar una señal de alta frecuencia para actuar sobre el motor a pasos que conduce la articulación prismática del manipulador.

Debido a todo el potencial de operación que ofrece el módulo, se ha hecho necesario explicar de manera detenida las cualidades que se han explotado en el BMX ECH 0200 en su aplicación en el controlador programable, entendiéndose que estas interfaces se ocupan de dotar al sistema de la capacidad esencial de referenciar espacialmente la configuración articular de la cadena de eslabones, permitiéndose así la capacidad de realizar control sobre los actuadores dispuestos en el robot SRX 4CH.

Además, la tarjeta de contaje resulta ser el componente novedoso de esta investigación y que ha de servir de referencia de consulta de las operaciones que se realicen sobre dicho robot en el futuro.

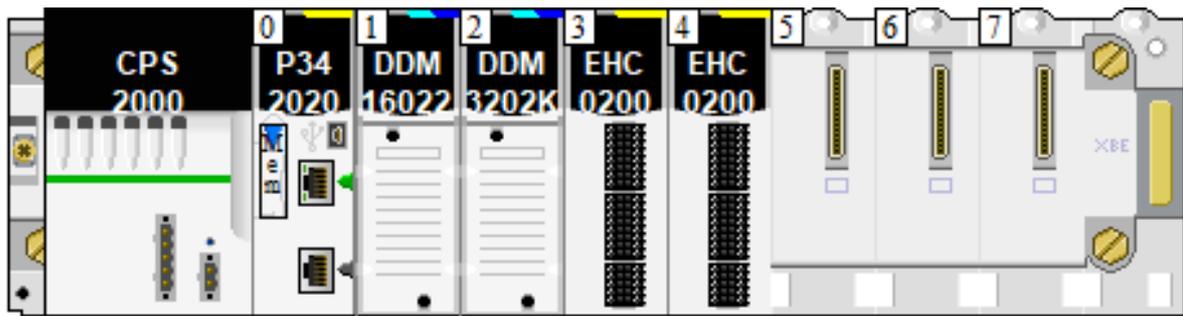


Figura 4-9. Módulos BMX EHC 0200 fijados en el PLC M340 Modicon Schneider Electric.

Al describir el BMX EHC 0200 de formato estándar, lo primero que se ha de mencionar es que el módulo permite contar pulsos a una frecuencia máxima de 60 KHz. Cada acoplador de la serie dispone de dos canales de entrada, cada uno con un bloque de terminales de 16 pines, que son capaces de vincular las señales provenientes de sensores de proximidad de dos o tres hilos de 24 Vcc de alta velocidad de respuesta. Y de la misma manera se pueden aplicar codificadores de señal incremental con salida de 10/30 Vcc, que inclusive pudieren poseer salidas con contrafase.

En ese mismo orden de ideas, el módulo contiene un bloque de terminales de 10 pines para la conexión de las salidas rápidas para el accionamiento de actuadores que posean esta naturaleza de operación. Además, los BMX EHC 0200 admiten todas las funcionalidades del software de Unity Pro, que es empleado para la configuración del autómata.

Las principales aplicaciones que se realizan con los módulos de la serie son el conteo de alta velocidad progresivo y/o regresivo, la medición de frecuencia, la supervisión del desplazamiento angular de ejes y de la misma manera se pueden generar señales frecuenciales para la alimentación de actuadores de diversa índole, que demanden ser energizados con una alta velocidad de respuesta. La tabla siguiente muestra las características de los bloques de terminales BMX EHC 0200.

Tabla 4-6. Especificaciones del módulo BMX EHC 0200.

<b>Características del Módulo BMX ECH 0200</b>					
<b>Tipo de módulo</b>	2 canales de entrada para conteo, 1 para salidas				
<b>Máxima frecuencia de las entradas de contaje</b>	60 KHz				
<b>Tensión de aislamiento de la base al bus</b>	1500 V RMS para 1 min				
<b>Tiempo de ciclo del módulo</b>	1 ms				
<b>Números de entradas/salidas por canal de conteo</b>	<b>Entradas</b>	6 entradas de 24 Vcc			
	<b>Salidas</b>	Dos salidas 24 Vcc			
<b>Conformidad del codificador</b>	10-30 Vcc codificador incremental con contrafase en las salidas				
<b>Fuente de Alimentación</b>	Para energizar los sensores desde 15 a 75 mA				
<b>Características de las entradas del canal</b>					
<b>Entradas</b>	Seis entradas de 24 Vcc				
	In_A, IN_B, IN_SYNC, IN_EN, IN_REF, IN_CAP. (IN_A, IN_B e IN_SYNC son rápidas)				
<b>Estado</b>	Encendida	11-30 Vcc con 2-5 mA (para el rango de tensión)			
	Apagada	< 5 Vcc con < 1.5 mA			
<b>Características de las salidas del módulo</b>					
<b>Tensión</b>	<b>Número de salidas por módulo</b>	2	<b>Tipo</b>	Común positivo 24 Vcc 0.5 A	
	<b>Tensión</b>	19.2-30 Vcc	<b>Caída de tensión en estado 1</b>	3 Vcc como máximo	
<b>Corriente</b>	<b>Corriente de carga máxima</b>	<b>Por punto</b>	0.5 A	<b>Por módulo</b>	2 A
	<b>Corriente de fuga en estado 0</b>	0.1 mA como máximo		<b>Capacidad máxima</b>	50 µf
<b>Características mecánicas del módulo</b>					
<b>Conexionado</b>	<b>Calibre mínimo</b>			<b>Calibre máximo</b>	
1 cable por terminal	AWG 24 (0.5 mm <sup>2</sup> )			AWG 17 (1 mm <sup>2</sup> )	

### 4.6.1 LOS CONECTORES DE 16 PINES DEL BMX EHC 0200

La figura que se ilustra a continuación describe la función de cada uno de los terminales de los dos conectores de 16 pines que son empleados para el acoplamiento rápido de las señales de entrada que son enlazadas a los canales del módulo BMX EHC 0200.

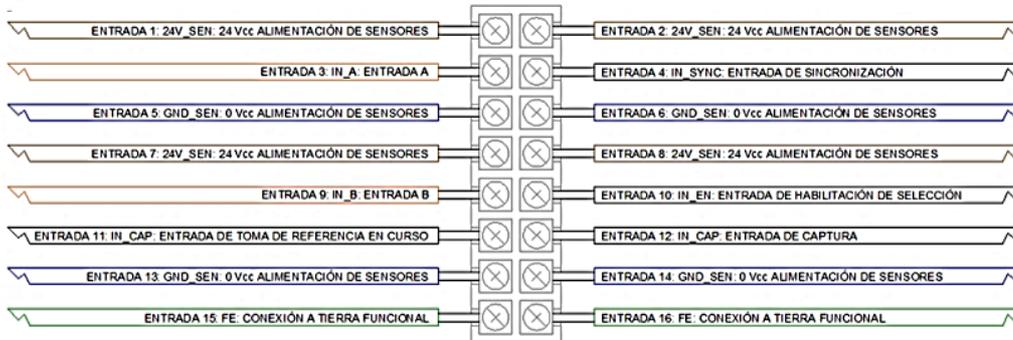


Figura 4-10. Diagrama de conexión del bloque de entradas del BMX EHC 0200.

Si se amerita realizar el manejo de sensores de respuesta de alta velocidad, se ha de seguir el esquema que la figura siguiente ofrece. Para alcanzar este objetivo, se han de alimentar con los terminales que proveen tensión (entradas 1,2,7,8 para +24 Vcc y 5,6,13,14 para -0 Vcc) y así mismo el diagrama referencia a las entradas *IN\_A* (entrada 3) e *IN\_B* (entrada 9) para conseguir el propósito, ya que son capaces de captar pulsos de hasta 5  $\mu$ s de duración.

Además, la figura indica el empleo de contactos externos para comandar el uso de registros internos del controlador y que están disponibles a través de las entradas *IN\_SYNC* (entrada 4 para sincronización) e *IN\_EN* (entrada 10 para habilitación) que posee el módulo en ambos canales. El esquema de la ilustración 4-11, también indica que las entradas *FE* (entradas 15 y 16) han de emplearse para realizar el aterramiento del apantallado de los conductores que se emplean para el enrutado de las señales, evitando de esta manera la aparición de errores en la medición por la incidencia de interferencia electromagnética en el canal.

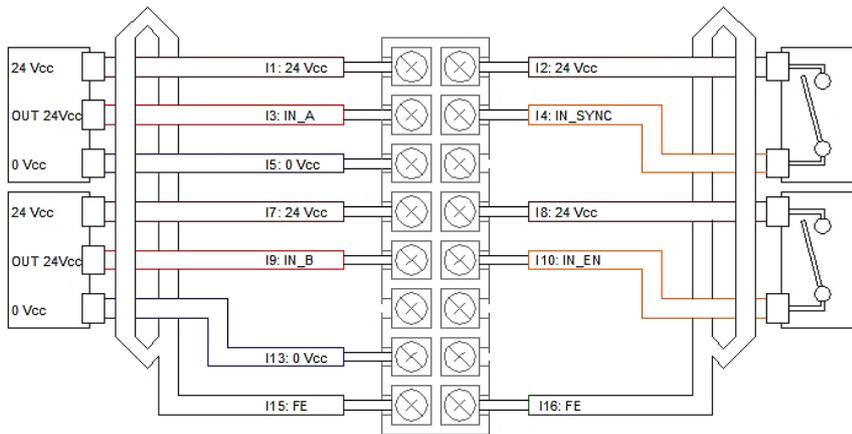


Figura 4-11. Diagrama de conexión del bloque de entradas del BMX EHC 0200 con sensores.

De la misma manera, en el caso de ameritarse el manejo de encoders para la referencia de contaje, se debe seguir el diagrama que la figura siguiente permite observar y que referencia a las entradas *IN\_A* (entrada 3) e *IN\_B* (entrada 9) para alcanzar el objetivo, siendo que cada una puede contar trenes de pulsos de hasta 60 KHz y se pueden emplear en conjunto para medir fracciones de dicha señal. Además, como fue hecho en la sección anterior, se muestra que el comando de registros internos adicionales del controlador se puede llevar a cabo con el uso de contactos externos y que son accesibles a través de las entradas *IN\_REF* (entrada 11 para la toma de referencia) e *IN\_CAP* (entrada 12 para la captura) que posee el módulo en ambos canales.

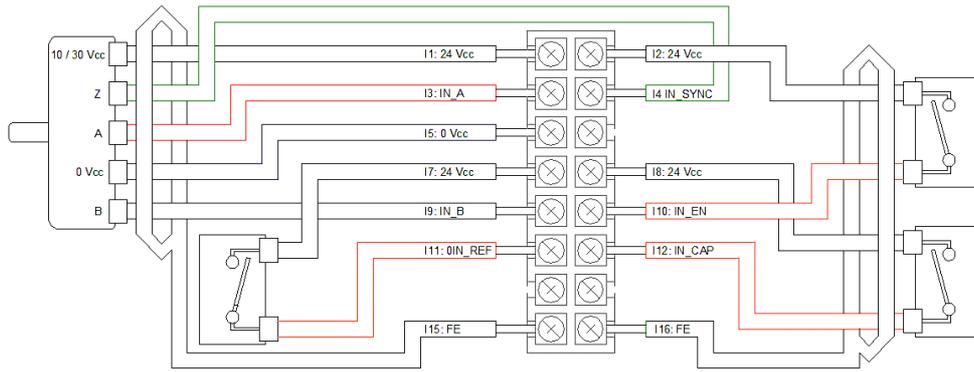


Figura 4-12. Diagrama de conexión del bloque de entradas del BMX EHC 0200 con encoder.

#### 4.6.2 EL CONECTOR DE 10 PINES DEL BMX EHC 0200

Para describir el bloque de terminales del conector de 10 pines que se emplea para el acoplamiento rápido de las señales de salida, se hace uso de la figura que se presenta a continuación, en ella se describe la función de cada uno de los bornes del único canal disponible en el módulo BMX EHC 0200 para conectar los actuadores del manipulador y así ejercer acciones de control sobre ellos.

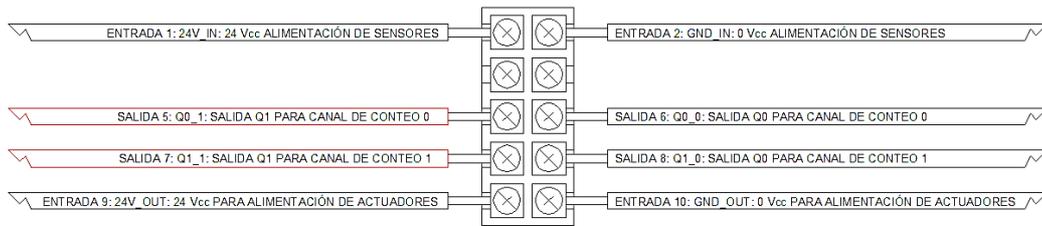


Figura 4-13. Diagrama de conexión del bloque de salidas del BMX EHC 0200.

En función de emplear el módulo para el gobierno de actuadores, se debe seguir el diagrama que la figura siguiente instruye desarrollar y que señala que se han de alimentar con 24 voltios de corriente continua los terminales que proveen energía a los sensores y a los actuadores que se enlazan al acoplador (entradas 1 y 9 para +24 Vcc y 2 y 10 para -0 Vcc), desde una fuente externa. Indica también el esquema que los terminales  $Q0_0$ ,  $Q0_1$ ,  $Q1_0$  y  $Q1_1$  (salidas 5,6,7 y 8) se han de utilizar para emitir al proceso las señales de actuación sobre los elementos de campo. Estas pudieren ser una señal todo o nada, un tren de pulsos que puedan conmutar intermitentemente un dispositivo o se destinen a modular una señal de empleo específico. Estas señales de salida sólo podrán poseer una frecuencia máxima de 4 KHz, teniendo en cuenta que solo podrán conducir 0.5 amperios por lo que se habrán de disponer de acondicionadores para acoplar elementos que disipen potencias superiores al consumo de corriente señalado. Para resguardo de la circuitería interna ante sobrecargas, las salidas del BMX EHC 0200 poseen protección térmica.

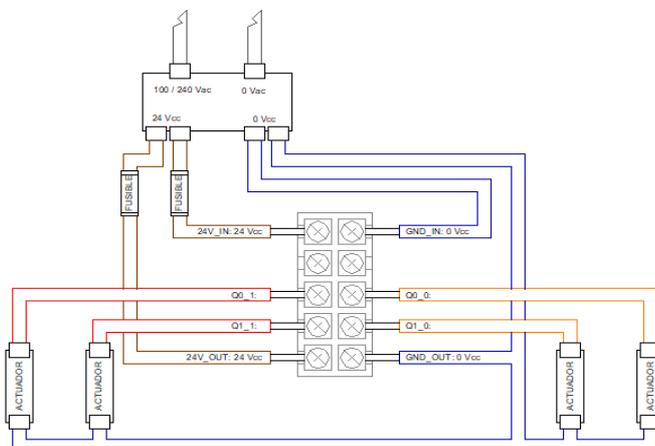


Figura 4-14. Diagrama de conexión del bloque de salidas del BMX EHC 0200 con actuadores.

### 4.6.3 MODALIDADES DE OPERACIÓN DEL MÓDULO

El módulo BMX EHC 0200 puede ser empleado en varias modalidades de funcionamiento, elegir una u otra dependerá de la utilidad física que aporte a la aplicación que tenga destinada la función de contaje de alta velocidad de la señal que se recibe en el canal del acoplador. Para el manipulador se han elegido dos modalidades de operación, debido a que cómo se ha mencionado previamente la articulación prismática está configurada mecátricamente de manera distinta a las rotacionales en el SCARA, que al ser medidas con el empleo de los codificadores ópticos, se han manejado en la modalidad de contador libre de 32 bits y para la regulación del avance longitudinal del efector con alta precisión y rapidez, se ha elegido la modalidad de modulación del ancho de pulso.

En las siguientes secciones del documento se ofrece una descripción de cada una de ambas modalidades para comprender las razones que llevaron a su elección y puesta en ejecución.

#### 4.6.3.1 EL MÓDULO BMX EHC 0200 EN MODALIDAD DE CONTADOR GRANDE LIBRE

Haber empleado esta modalidad obedece a las recomendaciones del fabricante del autómata que señala este modo de operación como ideal para ser aplicada en las tareas de supervisión de la posición de un eje motriz. Tareas como, por ejemplo, el contado de piezas sobre una banda de transporte y su presencia deviene en la ocurrencia de la modificación de estado de alguna de las entradas, con ello es posible referir la posición del eje mediante el registro de la posición que el módulo apunta en el instante de dicho cambio de estado y muchas otras posibilidades operativas con el modo de contador libre.

En términos operativos, habilitar el módulo requiere colocar en alto el bit *FORCE\_ENABLE* y de la misma manera establecer el bit *VALID\_ENABLE* cuando la entrada *IN\_EN* se encuentre en el nivel superior. Una vez realizada esta operación el canal de entrada queda disponible para realizar la lectura de las entradas *IN\_A* e *IN\_B* para la realización del contaje de alta velocidad y la palabra de *STATUS* del módulo indicará un número entero que está ligado a los bits de habilitación del canal que se han establecido como resultado de la operación. En este mismo orden de ideas, se puede manipular la activación o desacoplo del resto de señales disponibles en el canal, tanto internas de la unidad de procesamiento del autómata, así como externas y que provienen del proceso, en vías de realizar tantas aplicaciones como se hagan necesarias para el dispositivo que se monitoriza con el canal, entre las que se pueden mencionar, la captura de una posición, la medición de dimensiones del desplazamiento directo o indirecto del objeto asociado al eje, reiniciar el conteo, entre otras.

Para el caso del SRX 4CH, no se han destinado señales adicionales, más que los dos trenes de pulsos de ambos canales [*A* y *B*] del encoder, y para las dos primeras articulaciones se encuentran cableadas también el canal indicador de un giro completo del codificador óptico [*canal Z*].

En la modalidad de contador libre se puede configurar el contaje en alguna de las siguientes opciones de incidencia de las señales que se obtienen del encoder.

- *A* = arriba, *B* = abajo (configuración predeterminada)
- *A* = impulso, *B* = dirección
- Cuadratura normal o inversa *X1*, *X2* o *X4*.

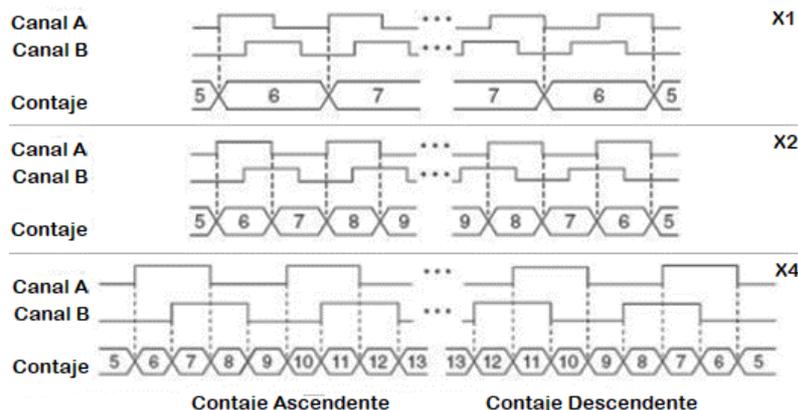


Figura 4-15. Capturas de entradas rápidas en cuadratura.

De la figura 4-15 se logra comprender a que se refiere las configuraciones de las cuadraturas del contaje rápido, en la misma se muestran los diferentes ciclos mencionados y como el módulo procesa el valor de contado en incrementos o decrementos de la magnitud según la percepción de los trenes de pulsos que son recibidos desde el codificador.

Como se observa en la primera sección de la imagen que refiere a la cuadratura  $X1$ , cuando el pulso en el canal  $A$  se adelanta al flanco de subida del  $B$ , el incremento se produce por el flanco de subida del canal  $A$ . De manera reciproca cuando el pulso en el canal  $B$  adelanta al canal  $A$ , el decremento se produce por el flanco de bajada del canal  $A$ . Esa es la valencia más significativa de los codificadores ópticos, la capacidad de autodetección del sentido de giro del eje motriz.

El mismo comportamiento se obtiene en la codificación  $X2$ , el desfase del canal  $B$  ofrece la dirección del contaje, más ahora son obtenidos los incrementos o decrementos del contador tanto en el flanco de subida, como en el de bajada del canal  $A$ , por lo tanto, la resolución del pulso se duplica, en otras palabras, para la misma magnitud de desplazamiento ahora se tienen el doble de impulsos, por ende la precisión del contado y la capacidad de cálculo derivada de este valor se incrementa en el mismo factor.

Ahora bien, de ser configurada la cuadratura  $X4$ , la estimación se hará con cada uno de los flancos de las señales emitidas por los dos canales. Por lo que el número de pulsos se cuadruplica y al igual que como en el apartado anterior la precisión de estimación del módulo. Y bajo el mismo principio, la cuenta se incrementa o decremента dependiendo de si los pulsos son percibidos primero en un canal antes que en el otro.

Finalmente, una vez fijado el tipo de codificación y ya habiéndose contado los impulsos, que son capturados en la carrera del eje, como ha sido mencionado, se realiza la conversión por relaciones matemáticas para presentar en el formato más apropiado, la posición que alcanza el mecanismo en función de los pulsos que se han registrado:

Cuando el desplazamiento es rotativo se emplea la ecuación:

$$(\#\_grados) = \frac{Fc}{xN} * 360 \quad (5 - 1)$$

La cantidad de rotación en grados es resultado de relacionar la incidencia de flancos ( $Fc$ ), el número de pulsos generados por el codificador por giro del eje ( $N$ ) y el tipo de codificación ( $x$ ).

Y para referenciar movimientos lineales se emplea:

$$(\#\_grados) = \frac{Fc}{xN} * \frac{1}{PPmm} \quad (5 - 2)$$

Siendo  $PPmm$  el número de pulsos por milímetro de recorrido.

#### 4.6.3.2 EL MÓDULO BMX EHC 0200 EN MODALIDAD DE MODULACIÓN DEL ANCHO DE PULSOS

En esta modalidad de servicio, el módulo utiliza un generador de reloj interno para ofrecer señales periódicas en la salida  $Q0$  del módulo. Este modo operacional es elegido con la habilitación del bit del comando `output_block_0_enable` en alto (1), siendo además posible con este ajuste la obtención de una modulación en la salida  $Q0$ .

La forma de onda de la señal de salida se configura haciendo que el valor de la frecuencia de modulación esté entre 1 (0.1 Hz) y 40.000 (4 KHz) unidades, donde es intuitivo comprender que los incrementos corresponden a 0,1 Hz. Y el valor porcentual del ciclo de servicio de la señal esté comprendido entre 1 (5%) hasta 19 (95%), y como en el caso anterior están definidos los incrementos al 5%.

La siguiente imagen muestra el funcionamiento del módulo en la modalidad de modulación del ancho de pulsos.

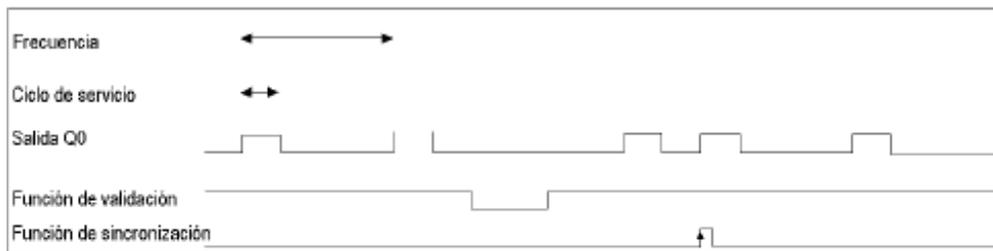


Figura 4-16. Salida del bloque BMX EHC 0200 en modulación de ancho de pulsos.

En esta configuración la frecuencia máxima de salida es 4 KHz. Al igual que la máxima aplicable en la entrada *IN\_SYNC* es un pulso cada 5 ms. Asimismo es necesario disponer entre bornes de la salida *Q0* a una resistencia de carga de 250  $\Omega$ , para lograr obtener una salida eficiente. Esto ajustará la señal de salida *Q0* a 0 V al no estar esta energizada, asegurando que se obtengan los cambios de la onda de salida y con la frecuencia correcta que ha sido configurada.

Con esta consideración se logra conseguir una relación del ciclo de servicio permitido de dicha onda, que puede obtener una variación, certificada por el fabricante, a medida que la frecuencia de la salida *Q0* se incrementa y que es mostrada en la siguiente tabla.

Tabla 4-7. Relación de ciclo de servicio en comparación a la frecuencia.

Frecuencia	Ciclo de Servicio
0.1-250 Hz	95 / 5%
251-500 Hz	90 / 10 %
501-1000 Hz	80 / 20 %
1001-1500 Hz	70 / 30 %
1501-2000 Hz	60 / 40 %
2001-2500 Hz	50 %
2501-4000 Hz	50 % no garantizado por el fabricante.

## 4.7 EL MÓDULO DE SALIDAS ANALÓGICAS BMX AMO 0210

Los módulos BMX AMO 0210 han sido elegidos para conducir los servomotores del SCARA por poseer las características eléctricas que se describen en este apartado. El módulo posee dos canales de salida analógica independientes que tienen un tiempo de respuesta inferior a los 2 ms, por lo tanto, se pueden destinar para el manejo de dos actuadores con señales de tensión de  $\pm 10$  V, o corriente desde 0 a 20 y 4 a 20 mA generadas por el mismo módulo, donde esta última configuración es capaz de detectar si el lazo de corriente se encuentra abierto o detectar cortocircuitos de ser configurados para manejar tensión. Además de las especificaciones que han sido mencionadas el módulo posee características más dedicadas que se listan en la siguiente tabla y que aportan valor al momento de la realización de la configuración de la señal de control que emitirán al proceso.

Tabla 4-8. Especificaciones del módulo BMX AMO 0210.

BMX AMO 0210				
<b>Tipo de salidas</b>	Salidas libres de potencial de alto nivel	<b>Resolución</b>	15 bits + signo	
<b>En configuración de tensión</b>	Mínima Impedancia de Carga		1 K $\Omega$	
	<b>Resolución analógica</b>	0.27 mV	Detección de Errores	Cortocircuito
<b>En configuración de corriente</b>	Máxima Impedancia de Carga		600 $\Omega$	
	<b>Resolución analógica</b>	0.74 $\mu$ A	Detección de Errores	Circuito abierto

Las salidas de la tarjeta analógica están normalizadas en el formato [-10.000 a +10.000 unidades] para el rango  $\pm 10$  V y [0 a +10.000] para 0 a 20 y/o 4 a 20 mA.

Usualmente la tarjeta de señales analógicas AMO 0210 conmuta su salida a 0 V o 0 mA en las acciones de parada, ante la incidencia de errores, la interrupción de la comunicación del autómata, así como también satura las salidas ante el desborde de la magnitud de estas ( $\pm 1125$  unidades en tensión y  $\pm 1600$  en corriente) y otras acciones preventivas consideradas como respuesta segura ante condiciones anómalas de funcionamiento.

El BMX AMO 0210 se enlaza a los actuadores del proceso por medio de un bloque de terminales de 20 puntos, y ellos son conectados a los terminales [3, 4] y [17, 18] del bloque para el primer y segundo actuador respectivamente y como lo indica la figura siguiente, sin ameritar ningún tipo de acondicionamiento adicional para la señal, sin embargo, se debe prever la incidencia de interferencias en el conductor con el empleo de blindaje de las líneas que son direccionadas desde el módulo hacia los actuadores y realizar el apropiado aterramiento del blindaje según las recomendaciones que el fabricante describe para las fuentes de alimentación y los bastidores del Modicon M340.

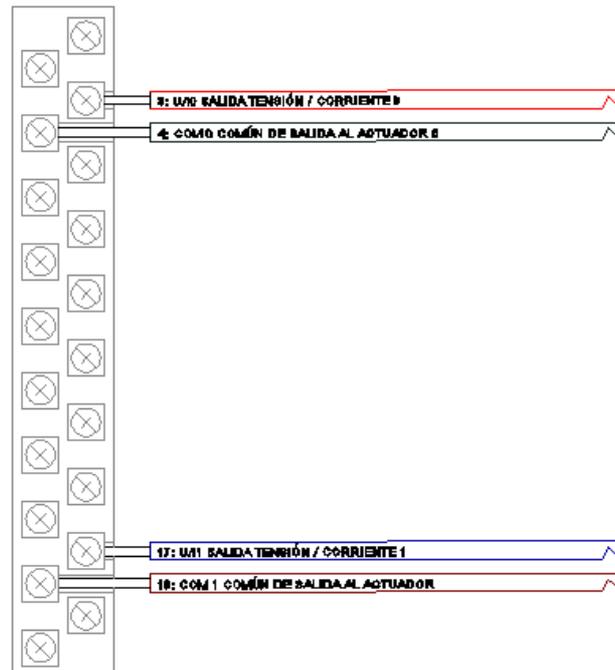


Figura 4-17. Terminales del Módulo BMX AMO 0210.

Finalmente, la figura 4-18 ilustra el módulo de salidas analógicas BMX AMO 0210 ya dispuesto sobre el bastidor del Modicon M340 de la sección de la célula de fabricación.

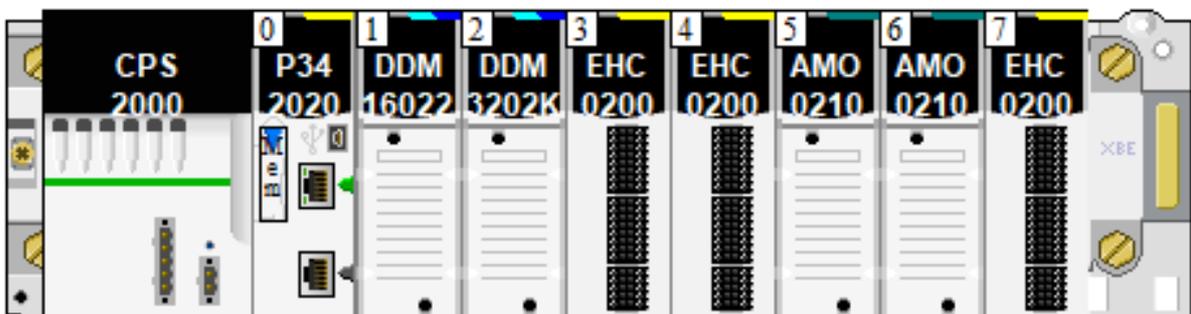


Figura 4-18. Módulos BMX AMO 0210 fijados en el PLC M340 Modicon Schneider Electric.

Gracias a la descripción de las características de cada módulo dispuesto en el Modicon M340 que en este apartado se ha llevado a cabo y a partir del conocimiento de la naturaleza de las variables que se le habrían de vincular a través de los terminales de cada uno de estos, se ha logrado plantear un esquema de conexión de las señales desde la unidad controladora con el manipulador y en dirección al autómata con el propósito de monitorizar y gobernar el robot SCARA.



## 5 VARIABLES Y SEÑALES DEL SISTEMA

El diagrama de conexiones de la figura 12-61, presentada en el octavo [8] anexo adjunto a este documento, ha conducido a realizar la identificación de las señales del sistema en conductores de potencia, de control y retroalimentación en todo el hardware del SCARA.

La unidad controladora se ha destinado para que desempeñe una nueva función como *Unidad de Potencia*, desde esta se envían al robot a través de una manguera con 18 hilos conductores de sección  $2.08 \text{ mm}^2$  (# 14 AWG: *American Wire Gauge*), como se observa en la próxima ilustración, las magnitudes eléctricas que actúan sobre los motores de las articulaciones. El servo para la motorización del eje Z se ha cambiado, por tal razón los conductores [T4OP, T4OM, BRKP, BRKM] se usan para alimentar los terminales del motor a pasos NEMA 17 [A+, A-, B+, B-], por haberse deteriorado el motor original en el pasado, por esa misma razón tampoco se utilizan los cables [T4PA, T4PB] del encoder que se empleaba en la misma articulación.

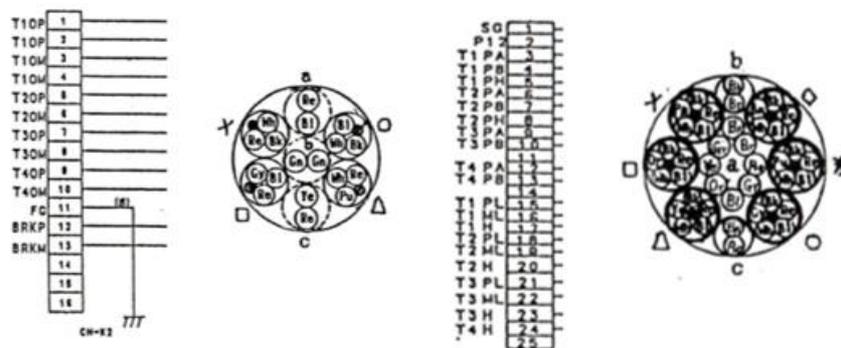


Figura 5-1. Conectores de cables de la unidad de potencia al robot.

De manera recíproca desde el SCARA se reciben la información de posición variable de los codificadores ópticos y la absoluta de los finales de carrera. Estas magnitudes viajan en una manguera de 25 hilos conductores que, por ser de menor consumo, su sección transversal también es de menor dimensión.

Ahora bien, enlazar la unidad de potencia de la controladora del robot con el PLC Modicon M340 de Schneider Electric ha sido una de las dificultades de este proyecto, debido a que el autómata está diseñado para operar a 24 voltios de corriente continua, que en las últimas décadas ha sido el estándar de alimentación del hardware empleado en los dispositivos destinados a la automatización de procesos, dicha tensión difiere de la alimentación del SRX 4CH que fue alimentado a 230 voltios de corriente alterna, y empleaba diferentes fuentes de tensión para energizar cada una de las tarjetas electrónicas que demandaba para ejecutar las operaciones de los sistemas de vídeo, el servo manejo, la unidad de procesamiento, entre otras.

Sin embargo, para salvar esta restricción, fue habilitada una interfaz que estaba dispuesta en el cuadro eléctrico, para la correspondiente adecuación de las diferencias de potencial de los distintos elementos de la controladora que comandan la motorización de los eslabones, los codificadores de posición que se asocian cada motor y el resto de dispositivos como relés, sensores de barrera y otras señales importantes para el gobierno del brazo SCARA.

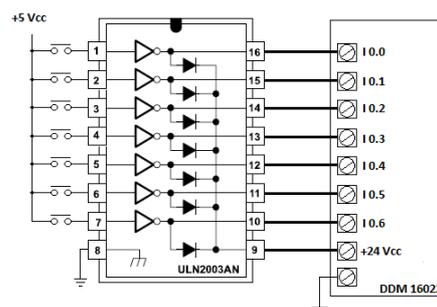


Figura 5-2. Esquema de conexión de la interfaz electrónica.

La tarjeta para el acoplo está construida a partir de Amplificadores Darlington, este arreglo se ilustra en la figura previa, y al darse las condiciones de alimentación necesarias para el funcionamiento del integrado, se cierra el circuito entre colector emisor de la pareja de transistores, consiguiendo que los 24 voltios estén reflejados a la salida con un tiempo de retardo bastante pequeño para cada conmutación de los siete (7) arreglos embebidos que conforman el circuito integrado (IC) *ULN2003AN*.

La tarjeta que se ha referido contiene cuatro (4) encapsulados, por lo tanto, se consiguen llevar al autómeta hasta 28 señales de diversa naturaleza, entre las que se listan las respuestas de alta velocidad de los encoders, las consignas analógicas con la que se regulan el giro de los motores DC y varias señales digitales de finales de carrera que se conmutan en la adopción de posiciones y señales de enclavamiento de relés para el accionar de los actuadores del robot.

Del arreglo que se empleó en el diseño de la tarjeta impresa que se observará en la figura siguiente, se tiene que los conductores son ligados a ella con el uso de conectores *DB50* y *DB25*, con ello se hacen llegar al autómeta y al manipulador respectivamente (50 conductores al autómeta, 25 al SCARA), las variables de tensión que estos vinculan entre el controlador programable, la unidad de potencia y el robot. El circuito interno de la tarjeta electrónica está reflejado en la figura 12-62 del numeral 8 de los anexos.

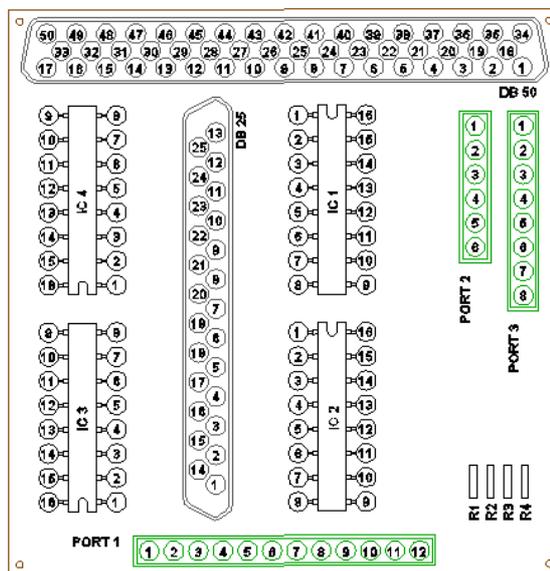


Figura 5-3. Disposición de amplificadores darlington en el PCB.

El siguiente paso es identificar sobre el SRX 4CH la localización física de los actuadores y sensores que están dispuestos en él, esto se ha hecho con el uso de la siguiente figura para ilustrarlo, comprendiendo que realizar una adecuada identificación es esencial para sustentar el programa que se configura, elegir adecuadamente la dimensión y las especificaciones del autómeta y para regir las actuaciones sobre el manipulador.

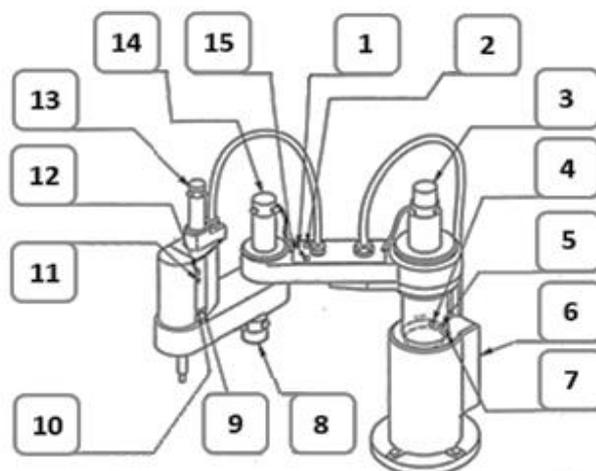


Figura 5-4. Ubicación de los sensores y actuadores del SONY SRX 4CH.

Esta descripción se completa gracias a la utilización de la tabla siguiente.

Tabla 5-1. Lista de variables de entradas y salidas ubicadas en el robot.

Nº	Símbolo	Descripción de la función
1	T2H	Sensor que detecta que se ha llegado a la posición de home de la articulación theta_2.
2	T2LI	Sensor que detecta que se ha llegado al límite inferior de la articulación theta_2.
3	T1M	Motor de la articulación theta_1.
4	T1LI	Sensor que detecta que se ha llegado al límite inferior de la articulación theta_1.
5	T1H	Sensor que detecta que se ha llegado a la posición de home de la articulación theta_1.
6	EV	Electroválvula para accionamiento de la pinza neumática.
7	T1LS	Sensor que indica que se ha llegado al límite superior de la articulación theta_1.
8	T4M	Motor de la articulación theta_4.
9	T4H	Sensor que detecta que se ha llegado a la posición de home de la articulación theta_4.
10	D3LS	Sensor que indica que se ha llegado al límite superior de la articulación delta_3.
11	D3H	Sensor que detecta que se ha llegado a la posición de home de la articulación delta_3.
12	D3LI	Sensor que detecta que se ha llegado al límite inferior de la articulación delta_3.
13	D3M	Motor de la articulación delta_3.
14	T2M	Motor de la articulación theta_2.
15	T2LS	Sensor que indica que se ha llegado al límite superior de la articulación theta_2.

Ciertamente la tabla 5-1 se entiende como una descripción previa y genérica, por tal razón la intención es expandir cada ítem del listado en la búsqueda de particularizar las funciones de cada hilo conductor y de cada módulo que ha sido instalado en el bastidor.

Por ello se inicia en el mismo orden que estos ocupan sobre él, una descripción agrupada en las siguientes tablas.

## 5.1 VINCULACIÓN DE SEÑALES A LOS MÓDULOS DEL AUTÓMATA.

La primera ranura de conexión del bastidor denominadas slot, está ocupada por la tarjeta para el manejo de entradas y salidas binarias BMX DDM 16022, esta se encuentra encargada exclusivamente de encauzar las instrucciones de la célula de fabricación flexible al SCORBOT V5.

A este módulo de entradas/salidas se le agrega una descripción en esta sección del proyecto, porque está incluido en el mismo controlador programable y al presentar el hardware es preciso listarlo con el conjunto de señales que el autómata maneja.

Sin embargo, las variables que esta tarjeta emplea no guardan relación con la programación elaborada para el SONY SRX 4CH y será un aspecto que el usuario deberá incluir en la programación de los elementos en conjunto de esta porción de la célula.

La próxima tabla contiene las entradas empleadas para el manejo del robot.

Tabla 5-2. Lista de entradas empleadas en el módulo BMX DDM 16022.

Dirección de Entrada	Color cable	Descripción de la función
%I0.1.0	VL	Salida a colector abierto del SCORBOT Q5
%I0.1.1	BK	Salida a colector abierto del SCORBOT Q6
%I0.1.2	WH	Salida a colector abierto del SCORBOT Q7
%I0.1.3	PK	Salida a colector abierto del SCORBOT Q8

De la misma manera se agrupan las salidas en la tabla que sigue.

Tabla 5-3. Lista de variables de salidas empleadas en el módulo BMX DDM 16022.

Dirección de Salida	Color cable	Descripción de la función
%Q0.1.8	BN	Entrada digital del SCORBOT I1
%Q0.1.9	PK	Entrada digital del SCORBOT I2
%Q0.1.10	GN	Entrada digital del SCORBOT I3
%Q0.1.11	YW	Entrada digital del SCORBOT I4
%Q0.1.12	PK	Motor a pasos alimentador de piezas 1 STEP
%Q0.1.13	YW	Motor a pasos alimentador de piezas 2 DIR
%Q0.1.14	BN	Motor a pasos alimentador de piezas 2 DIR
%Q0.1.15	GY	Motor a pasos alimentador de piezas 2 STEP

Las instrucciones enviadas al SCORBOT son muy básicas por el hecho de que el manipulador posee en funcionamiento su unidad de control, por lo cual el Modicon M340 cumple tareas de enlace con la planta, a diferencia de las acciones más complejas que el autómatas realiza en el SONY. Por tal razón se agrupan las variables binarias que se vinculan por medio de la tarjeta BMX DDM 3202 K instalada en el segundo slot del bastidor.

Tabla 5-4. Lista de variables de entradas empleadas en el módulo BMX DDM 3202 K.

Dirección de Entrada	Símbolo	Descripción de la función
%I0.2.0	ED_HOME_THETA_1	Final de carrera del HOME de theta 1
%I0.2.1	ED_LIM_SUP_THETA_1	Final de carrera del límite superior de theta 1
%I0.2.2	ED_LIM_INF_THETA_1	Final de carrera del límite inferior de theta 1
%I0.2.3	ED_HOME_THETA_2	Final de carrera del HOME de theta 2
%I0.2.4	ED_LIM_SUP_THETA_2	Final de carrera del límite superior de theta 2
%I0.2.5	ED_LIM_INF_THETA_2	Final de carrera del límite inferior de theta 2
%I0.2.6	ED_HOME_DELTA_3	Final de carrera del HOME de delta 3
%I0.2.7	ED_LIM_SUP_DELTA_3	Final de carrera del límite superior de delta 3
%I0.2.8	ED_LIM_INF_DELTA_3	Final de carrera del límite inferior de delta 3
%I0.2.9	ED_HOME_THETA_4	Final de carrera del HOME de theta 4
%I0.2.13	ED_SETA_EMERGENCIA	Pulsador Seta de Emergencia

La tabla anterior agrupa las entradas que se encauzan al módulo y la que es mostrada a continuación, contiene las salidas que son controladas a través del mismo.

Tabla 5-5. Lista de variables de salidas empleadas en el módulo BMX DDM 3202 K.

Dirección de Salida	Símbolo	Descripción de la función
%Q0.2.16	SD_PINZA	Electroválvula de la pinza
%Q0.2.18	SD_POWER_ON	Encendido de la unidad de control
%Q0.2.19	SD_SAFETY	Habilitación de la seguridad.
%Q0.2.20	SD_PIN_HAB_TH1	Habilitación del Servo Driver de theta1
%Q0.2.21	SD_PIN_HAB_TH2	Habilitación del Servo Driver de theta2
%Q0.2.22	SD_ENB_DT3	Habilitación del Stepper Driver de delta3
%Q0.2.23	SD_PIN_HAB_TH4	Habilitación del Servo Driver de theta4
%Q0.2.24	SD_DIR_THETA_1	Relé de consigna del Servo Driver de theta1
%Q0.2.25	SD_DIR_THETA_2	Relé de consigna del Servo Driver de theta2
%Q0.2.26	SD_DIR_THETA_4	Relé de consigna del Servo Driver de theta4
%Q0.2.27	SD_DIR_DELTA_3	Relé de consigna del Stepper Driver de delta3

Asimismo, para la cuantificación de los pulsos emitidos por los codificadores y actuación con señales rápidas pulsantes provistas, se han insertado dos módulos BMX EHC 0200 para que ambos estando conectados a la unidad de potencia, según la manera que indica la siguiente tabla, gobiernen los cuatro canales necesarios para

todas las articulaciones del SRX 4CH.

Tabla 5-6. Lista de variables de entradas y salidas del módulo BMX EHC 0200.

Canal de Entrada	Símbolo	Descripción de la función
%CH0.3.0	MODULO_1_2	Canal para contaje de pulsos de theta1
%I0.3.0.4	canal0.Servo1.ST_INPUT_A	Tren de pulsos leídos del canal A del encoder
%I0.3.0.5	canal0.Servo1.ST_INPUT_B	Tren de pulsos leídos del canal B del encoder
%I0.3.0.6	canal0.Servo1.ST_INPUT_SYNC	Pulso de vuelta completa del encoder
%CH0.3.1	canal1.Servo2.MODULO_1_2	Canal para contaje de pulsos de theta2
%I0.3.1.4	canal1.Servo2.ST_INPUT_A	Tren de pulsos leídos del canal A del encoder
%I0.3.1.5	canal1.Servo2.ST_INPUT_B	Tren de pulsos leídos del canal B del encoder
%I0.3.1.6	canal1.Servo2.ST_INPUT_SYNC	Pulso de vuelta completa del encoder
%CH0.4.0	MODULO_3_4	Canal para modulación de pulsos de delta3
%Q0.4.0.0	canal0.Pap3.OUTPUT_0	Tren de pulsos para actuar sobre paso a paso
%CH0.4.1	canal1.Servo4.MODULO_3_4	Canal para contaje de pulsos de theta4
%I0.4.0.4	canal1.Servo4.ST_INPUT_A	Tren de pulsos leídos del canal A del encoder
%I0.4.0.5	canal1.Servo4.ST_INPUT_B	Tren de pulsos leídos del canal B del encoder

Finalmente, para completar la descripción de las señales que maneja el autómata, se listan en la próxima tabla los canales de los módulos de salida analógica BMX AMO 0210 que han sido empleados para manipular las variables de consigna para actuar sobre los motores de corriente directa de las articulaciones rotacionales 1,2 y 4.

Tabla 5-7. Lista de variables de salidas del módulo BMX AMO 0210.

Canal de Entrada	Símbolo	Descripción de la función
%CH0.5	an_MODULO_1_2	Canal para alimentación de motor DC
%QW0.5.0	an_canal.Servo1.OUTPUT_0	Señal de control del motor de theta1
%QW 0.5.1	an_canal1.Servo2.OUTPUT_1	Señal de control del motor de theta2
%CH0.6	an_MODULO_3_4	Canal para alimentación de motor DC
%QW 0.6.0	an_canal0.Servo4.OUTPUT_0	Señal de control del motor de theta4

Estas variables se encuentran presentes en el programa que ha sido creado y que reside en la memoria del módulo procesador BMX P34 2020, además de variables definidas como marcas internas (variables que no interactúan con el entorno del autómata) y variables para la parametrización de todos los procesos de cómputo de la CPU, esto se traduce en un proyecto que demanda un número de variables denso, debido al alto procesamiento matemático que se ejecuta cíclicamente y que se registra a medida que cada acción es ejecutada.

Donde la mayoría de las magnitudes son dinámicas y subyace en cada una la necesidad de alta precisión, con el fin de completar la acción que se ejecuta en tiempo real, que depende de los valores que han sido registrados en el movimiento anterior y que a la vez son el punto de partida del desplazamiento que se ha de llevar a cabo inmediatamente después.

## 5.2 SEÑALES DE LA CÉLULA DE FABRICACIÓN FLEXIBLE.

Estas mismas consideraciones están presentes en el entorno automatizado que es la Célula de Fabricación Flexible, la planta interactúa bajo la acción de un complejo sistema de señales eléctricas que se emplean para su control. El volumen de dichas señales es tan grande que no puede ser contenido en un único controlador.

En ese mismo orden de ideas, tantos procesos siendo llevados a cabo en una única unidad de procesamiento computarizada, demandaría que esta sea muy potente en cálculo y por ende costosa. Además, el sistema se haría vulnerable a la dependencia de las garantías de funcionamiento de dicho dispositivo de control.

Estas premisas propias de los Entornos de Control Distribuido han sido implementadas en la filosofía de trabajo

de la célula. La planta cuenta con un grupo de autómatas ubicados en los armarios o cuadros de potencia y control, quienes comparten paquetes de tareas pequeñas, que al ser integradas en el proceso global satisfacen las necesidades del procedimiento de desarrollo del producto final.

La ubicación de los diferentes elementos se puede precisar en la figura siguiente hecha a escala del sistema real, y donde además los dispositivos principales son referenciados para ofrecer su descripción listada en la tabla número 5-8.

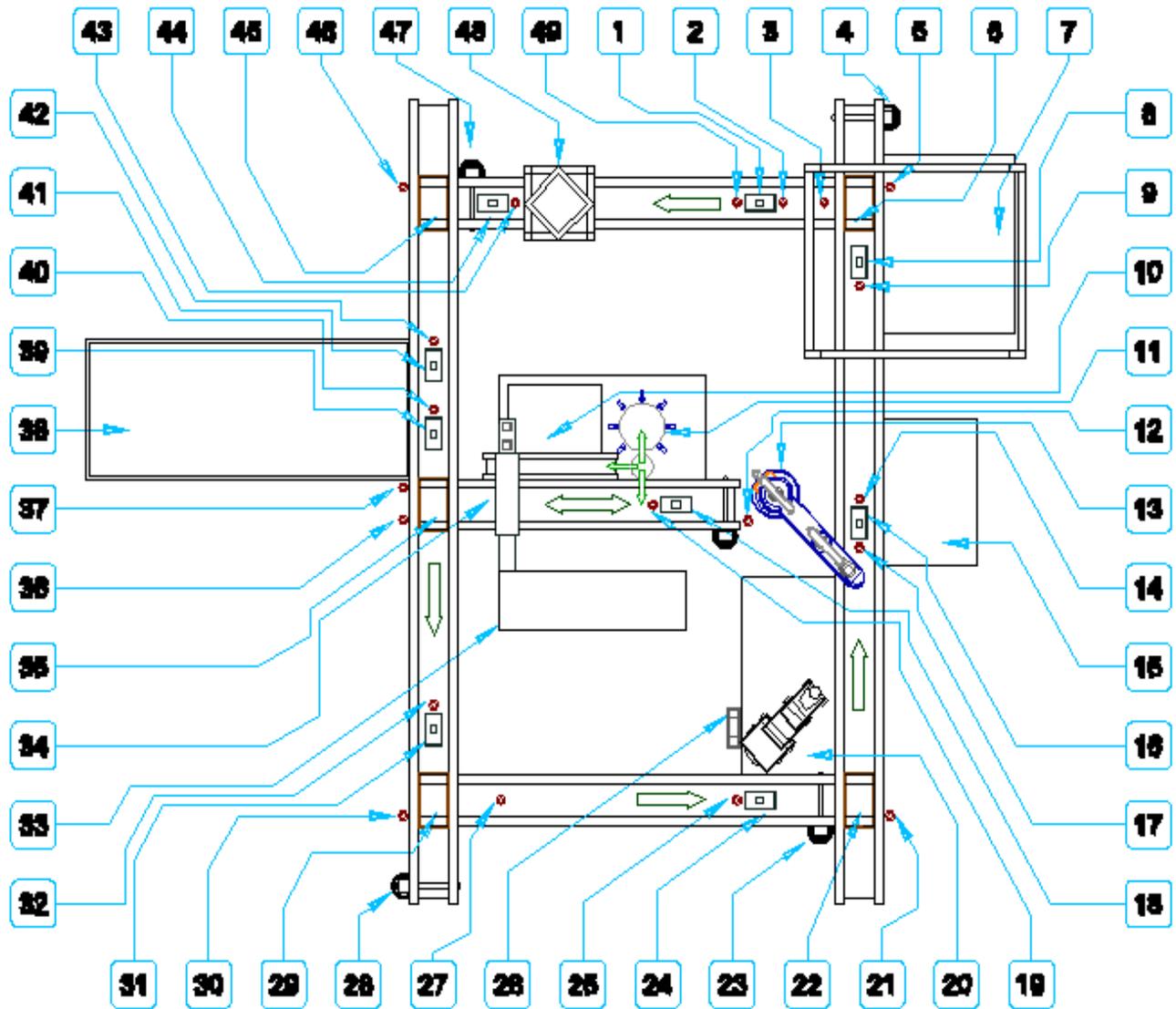


Figura 5-5. Ubicación de elementos de la Célula de Fabricación Flexible.

El sistema de control está sustentado en una red con protocolo TCP/IP y además es supervisado desde un SCADA (Vijeo CITEC) del mismo fabricante de los autómatas, que emplea también interfaces hombre máquina en los paneles principales para la configuración local de varias maniobras específicas de las estaciones a las que le ha sido asociado el tablero o cuadro de control.

De la misma manera se emplean ordenadores asociados a la red como estaciones de operador.

Desde estos se pueden optimizar el desempeño de los diferentes controladores lógicos programables de la red, para conseguir el manejo eficiente de la célula según las especificaciones que se instruyen en los ensayos a realizar en la planta y desarrollar desde los terminales todas las tareas de parametrización, programación, puesta a punto y supervisión de dichos ensayos.

En la tabla se muestran las señales que se establecen en la operación del nivel más básico del sistema.

Tabla 5-8. Lista de elementos de la Célula de Fabricación Flexible.

N°	Dirección	Descripción	N°	Dirección	Descripción
1	%Q0.2.13	SD_RETENEDOR_Puma	26	-	ALIMENTADOR DE PIEZAS SCORBOT
2	%I0.1.17	ED_POSICIÓN_PUMA_1	27	%I0.1.13	ED_FIN_DESVIO_1
3	%I0.1.8	ED_FIN_DESVIO_3	28	%Q0.2.16	SD_MOTOR_LINEA1
4	%Q0.2.18	SD_MOTOR_LINEA3	29	%Q0.2.2	SD_ELEVADOR_DESVIO 1
5	%I0.1.7	ED_POSICIÓN_DESVIO_3	30	%I0.1.2	ED_POSICIÓN_DESVIO_1
6	%Q0.2.6	SD_ELEVADOR_DESVIO 3	31	%Q0.2.1	SD_RETENEDOR_DESVIO 1
7		MESA 1	32	%I0.1.1	ED_PREVIO_DESVIO_1
8	%Q0.2.5	SD_RETENEDOR_DESVIO 3	33	-	CUADRO AUTÓMATA PRINCIPAL
9	%I0.1.6	ED_PREVIO_DESVIO_3	34	-	PORTICO ALIMENTADOR DE PIEZAS
10	-	CINTA	35	%Q0.2.10	SD_ELEVADOR_CRUCE
11	-	MESA GIRATORIA	36	%I0.1.14	ED_POSICIÓN_CRUCE
12	%I0.1.20	ED_POSICIÓN_AUX	37	%I0.1.13	ED_BARRA_CRUCE
13	-	ROBOT SONY	38	-	ALMACÉN DE PALETS ASR
14	%I0.1.16	ED_POSICIÓN_SONY_2	39	%Q0.2.9	SD_RETENEDOR_CRUCE
15	-	CUADRO ALIMENTADOR BANDEJAS	40	%I0.1.11	ED_PREVIO_CRUCE
16	%Q0.2.12	SD_RETENEDOR_Sony	41	%Q0.2.11	SD_RETENEDOR_ASR
17	%I0.1.15	ED_POSICIÓN_SONY_1	42	%I0.1.12	ED_POSICIÓN_ASR
18	%Q0.2.14	SD_RETENEDOR_Ventosa	43	%I0.1.9	ED_PREVIO_DESVIO_4
19	%I0.1.19	ED_POSICIÓN_VENTOSA	44	%Q0.2.7	SD_RETENEDOR_DESVIO 4
20	-	MESA 2 CUADRO SCORBOT Y SONY	45	%Q0.2.8	SD_ELEVADOR_DESVIO 4
21	%I0.1.5	ED_BARRA_DESVIO_2	46	%I0.1.10	ED_BARRA_DESVIO_4
22	%Q0.2.4	SD_ELEVADOR_DESVIO 2	47	%Q0.2.19	SD_MOTOR_LINEA4
23	%Q0.2.17	SD_MOTOR_LINEA2	48	-	ALIMENTADOR DE BANDEJAS
24	%Q0.2.3	SD_RETENEDOR_DESVIO 2	49	%I0.1.18	ED_POSICIÓN_PUMA_2
25	%I0.1.4	ED_PREVIO_DESVIO_2	50	%Q0.2.6	SD_MOTOR_LINEA_CENTRO

Para el manejo del SCARA es necesario que la célula al operar en modo automático indique el momento en que las piezas que este robot manipulará estén a su alcance. Esperando que estas instrucciones sean emitidas al bus, cuando se tenga una bandeja sujeta o bien por el retenedor que es propio del robot, que se señala con el ítem 16 del listado, que hará que sea detectada por los sensores de los numerales 14 y 17, o bien por el retenedor del manipulador por vacío del ordinal 18, en cuyo caso la detección se hará por el sensor 19. El espacio de trabajo también abarca la mesa giratoria del undécimo elemento de la lista en todas sus posiciones y todas aquellas nuevas interacciones que con la evolución de la célula de fabricación se agreguen posteriormente a este proyecto.

Interiormente en el Modicon M340 se llegó a contener una extensa lista de variables, que fue construida producto de la realización de ensayos con las diferentes estrategias que se fueron planteando para el control del sistema.

Por tal razón al final de la programación, al comprobar que el proceso se encontraba bajo un esquema de control que satisfizo las necesidades del escenario de trabajo, se completó un proceso de depuración de la lista, con el propósito de remover aquellas variables temporales y que en el programa final no cumplían alguna función. Teniendo la intención de optimizar la utilización de memoria de almacenamiento interno de la CPU. Además, esta lista se ha vuelto a depurar al momento de convertir las secciones en bloques de funciones derivadas y han pasado a pertenecer a dichos bloques. Esta particularidad es otra de las valencias importantes de la utilización de bloques DFB en el marco de la programación de autómatas bajo Unity Pro.

La figura siguiente permite mostrar cómo se ejecutó el proceso de depuración que se ha mencionado.

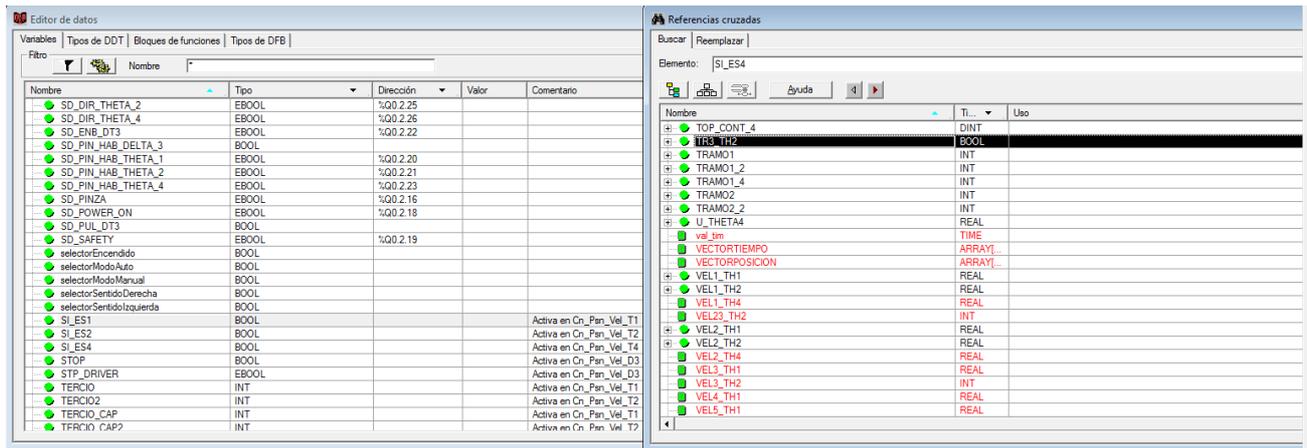


Figura 5-6. Proceso de Depuración de Variables del Programa.

De la misma manera se presentan las señales que se emplearon, que han quedado listadas en la agrupación de tablas que son ordenadas alfabéticamente a continuación.

Tabla 5-9. Lista N° 1 de variables del programa.

Lista de Variables de la A-B				
#	Nombre	tipo	Dirección	Ubicación
1	ACNM_D3	BOOL		Activa el motor 3 en ENCENDIDO y MOTORES
2	ACNM_T1	BOOL		Activa el motor 1 en ENCENDIDO y MOTORES
3	ACNM_T2	BOOL		Activa el motor 2 en ENCENDIDO y MOTORES
4	ACNM_T4	BOOL		Activa el motor 4 en ENCENDIDO y MOTORES
5	ACT_INIC	BOOL		Inicia la sección PCK_MOV_AUTO
6	ACT_INIC2	BOOL		Inicia la sección SEC_MOV_AUTO
7	AV	REAL		Incremento de avance en ENCENDIDO y COMANDO
8	AVANCE_ART1	BOOL		Activa el motor 1 en ENCENDIDO y MOTORES
9	AVANCE_ART2	BOOL		Activa el motor 2 en ENCENDIDO y MOTORES
10	AVANCE_ART3	BOOL		Activa el motor 3 en ENCENDIDO y MOTORES
11	AVANCE_ART4	BOOL		Activa el motor 4 en ENCENDIDO y MOTORES
12	botonActuador	BOOL		Actúa en COMANDO, MOTORES y ENCENDIDO
13	botonConfirma	BOOL		Actúa en COMANDO, MOTORES y ENCENDIDO
14	botonHome	BOOL		Actúa en COMANDO y ENCENDIDO
15	botonJoint	BOOL		Actúa en COMANDO y ENCENDIDO
16	botonMas1	BOOL		Actúa en COMANDO y ENCENDIDO
17	botonMas2	BOOL		Actúa en COMANDO y ENCENDIDO
18	botonMas3	BOOL		Actúa en COMANDO y ENCENDIDO
19	botonMas4	BOOL		Actúa en COMANDO y ENCENDIDO
20	botonMenos1	BOOL		Actúa en COMANDO y ENCENDIDO
21	botonMenos2	BOOL		Actúa en COMANDO y ENCENDIDO
22	botonMenos3	BOOL		Actúa en COMANDO y ENCENDIDO
23	botonMenos4	BOOL		Actúa en COMANDO y ENCENDIDO
24	botonParoRepPant	BOOL		Actúa en COMANDO y ENCENDIDO
25	botonRearme	BOOL		Actúa en las secuencias, COMANDO y ENCENDIDO
26	botonRegistro	BOOL		Actua en COMANDO y REGISTRO_POSICION
27	botonSeta	BOOL		Actúa en COMANDO y ENCENDIDO

Tabla 5-10. Lista N° 2 de Variables de programa.

<b>Lista de Variables de la B-M</b>				
#	Nombre	tipo	Dirección	Ubicación
1	botonStart	BOOL		Actúa en COMANDO, ENCENDIDO y M_AUTO
2	botonStop	BOOL		Actúa en COMANDO, ENCENDIDO y M_AUTO
3	botonWorld	BOOL		Actúa en COMANDO y ENCENDIDO
5	C_RX	REAL		Actúa en COMANDO y ENCENDIDO
6	C_RY	REAL		Actúa en COMANDO y ENCENDIDO
7	C_RZ	REAL		Actúa en COMANDO y ENCENDIDO
8	CNT_HM_MOSTRADO	INT		Actúa en COMANDO y ENCENDIDO
9	DEC_REV	BOOL		Actua en COMANDO y REGISTRO_POSICION
10	DELTA_3	BOOL		Actúa en COMANDO, VISUALIZACION, ENCENDIDO
11	DETENIDO	BOOL		Actúa en ENCENDIDO
12	DIF_TH3	INT		Actúa en secuencias, COMANDO y MOTORES
13	ED_HOME_DELTA_3	EBOOL	%I0.2.6	Sensor Home articulación 3
14	ED_HOME_THETA_1	EBOOL	%I0.2.0	Sensor Home articulación 1
17	ED_HOME_THETA_2	EBOOL	%I0.2.3	Sensor Home articulación 2
18	ED_HOME_THETA_4	EBOOL	%I0.2.9	Sensor Home articulación 4
19	ED_LIM_INF_DELTA3	EBOOL	%I0.2.8	Sensor límite inferior articulación 3
20	ED_LIM_INF_THETA1	EBOOL	%I0.2.2	Sensor límite inferior articulación 1
21	ED_LIM_INF_THETA2	EBOOL	%I0.2.5	Sensor límite inferior articulación 2
22	ED_LIM_SUP_DELTA3	EBOOL	%I0.2.7	Sensor límite superior articulación 3
23	ED_LIM_SUP_THETA1	EBOOL	%I0.2.4	Sensor límite superior articulación 1
24	ED_LIM_SUP_THETA2	EBOOL	%I0.2.1	Sensor límite superior articulación 2
25	ED_SET_EMERGENCIA	EBOOL	%I0.2.13	Pulsador seta de emergencia
26	EFECTOR	BOOL		Se activa en ENCENDIDO y VISUALIZACION
27	ESPACIO_TRABAJO	BOOL		Se activa en ENCENDIDO y COMANDO
28	FC_INF	BOOL		Finales de carrera inferiores de la articulación
29	FC_SUP	BOOL		Finales de carrera superiores de la articulación
30	H3_AUX	BOOL		Se activa en ENCENDIDO y MOTORES
31	HABILITA	BOOL		Se activa en ENCENDIDO y MOTORES
32	HM_FIN	BOOL		Se activa en ENCENDIDO
33	HM_PRC	BOOL		Se activa en ENCENDIDO y MOTORES
34	I	INT		Se activa en COMANDO y REGISTRO_POSICION
35	INC_REV	BOOL		Se activa en COMANDO y REGISTRO_POSICION
36	IND_VEC	INT		Índice del vector de posición
37	INICIADO	BOOL		Indica inici del proceso de puesta a HOME
38	MANUAL	BOOL		Elegido movimiento manual articular o cartesiano.
39	MARCHA_AUTO	BOOL		Modo de marcha automática ha sido iniciado.
40	MODO	BOOL		Transición de marcha del Grafcet Pick and Place
41	MODO2	BOOL		Transición de marcha del Grafcet Secuencia
42	MOT_AV1	BOOL		Activación en automático del motor 1
43	MOT_AV2	BOOL		Activación en automático del motor 2
44	MOT_AV3	BOOL		Activación en modo automático del motor 3

Tabla 5-11. Lista N° 3 de variables del programa.

<b>Lista de Variables de la M – P</b>				
<b>#</b>	<b>Nombre</b>	<b>tipo</b>	<b>Dirección</b>	<b>Descripción</b>
1	MOT_AV4	BOOL		Activación en modo automático del motor 4
2	PARO	BOOL		Paro de final del ciclo automático solicitada.
3	PAROFINAL	BOOL		Parada al final del ciclo.
5	PICK	BOOL		Inicia la rutina de manipulación de piezas
6	PIEZA_PRES	BOOL		Presencia de piezas en el retenedor de bandejas.
7	pilotoAuto	BOOL		En secuencias, COMANDO, ENCENDIDO y MOTORES
8	pilotoD3LI	BOOL		Se activa en ENCENDIDO y VISUALIZACION
9	pilotoD3LS	BOOL		Se activa en ENCENDIDO y VISUALIZACION
10	pilotoD3M	BOOL		Se activa en ENCENDIDO y VISUALIZACION
11	pilotoEMERGENCIA	BOOL		Se activa en ENCENDIDO, VISUALIZACION y COMANDO
12	pilotoEncendido	BOOL		Se activa en ENCENDIDO, VISUALIZACION y COMANDO
13	pilotoEV	BOOL		Se activa en ENCENDIDO y VISUALIZACION
14	pilotoFinHome	BOOL		Se activa en ENCENDIDO y COMANDO
17	pilotoHome	BOOL		Se activa en ENCENDIDO
18	pilotoJoint	BOOL		Se activa en ENCENDIDO y COMANDO
19	pilotoManual	BOOL		ENCENDIDO, COMANDO, REGISTRO_POSICION y MOTORES
20	pilotoMarcha	BOOL		En ENCENDIDO, COMANDO, MOTORES y VISUALIZACION
21	pilotoModoCentro	BOOL		Se activa en ENCENDIDO y COMANDO
22	pilotoD3H	BOOL		En ENCENDIDO y VISUALIZACION
23	pilotoREARME	BOOL		En ENCENDIDO, COMANDO y VISUALIZACION
24	pilotoRegistro	BOOL		En ENCENDIDO, COMANDO y REGISTRO_POSICION
25	pilotoRevision	BOOL		Se activa en ENCENDIDO
26	pilotoT1H	BOOL		Se activa en ENCENDIDO y VISUALIZACION
27	pilotoT1LI	BOOL		En ENCENDIDO, MOTORES y VISUALIZACION
28	pilotoT1LS	BOOL		En ENCENDIDO, MOTORES y VISUALIZACION
29	pilotoT1M	BOOL		Se activa en ENCENDIDO y VISUALIZACION
30	pilotoT2H	BOOL		Se activa en ENCENDIDO y VISUALIZACION
31	pilotoT2LI	BOOL		En ENCENDIDO, MOTORES y VISUALIZACION
32	pilotoT2LS	BOOL		En ENCENDIDO, MOTORES y VISUALIZACION
33	PilotoT2M	BOOL		Se activa en ENCENDIDO y VISUALIZACION
34	pilotoT4H	BOOL		En ENCENDIDO, MOTORES y VISUALIZACION
35	PilotoT4M	BOOL		Se activa en ENCENDIDO y VISUALIZACION
36	pilotoWorld	BOOL		Se activa en ENCENDIDO y COMANDO
37	PIN_HAB_MOT	BOOL		Se activa en M_AUTO
38	PINZA_AUTO	BOOL		Activación de pinza en modo automático
39	PINZA_HOME	BOOL		Activación de pinza en el proceso de puesta a HOME
40	PINZA_MAN	BOOL		Activación de pinza en el modo manual
41	POS_0	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
42	POS_1	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
43	POS_2	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
44	POS_3	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
45	POS_4	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
46	POS_5	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
47	POS_6	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.

Tabla 5-12. Lista N° 4 de variables del programa.

<b>Lista de Variables de la P – P</b>				
<b>#</b>	<b>Nombre</b>	<b>típo</b>	<b>Dirección</b>	<b>Descripción</b>
1	POS_7	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
2	POS_8	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
3	POS_9	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
5	POS_10	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
6	POS_11	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
7	POS_12	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
8	POS_13	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
9	POS_14	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
10	POS_15	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
11	POS_16	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
12	POS_17	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
13	POS_18	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
14	POS_19	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
17	POS_20	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
18	POS_21	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
19	POS_22	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
20	POS_23	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
21	POS_24	BOOL		Posición del vector alcanzada, se habilita el paso siguiente.
22	POS_DES_TH1	INT		Se activa en secuencias MOTORES Y COMANDO
23	POS_DES_TH2	INT		Se activa en secuencias MOTORES Y COMANDO
24	POS_DES_TH4	INT		Se activa en secuencias MOTORES Y COMANDO
25	POS_DEST_D3	EBOOL		Se activa en secuencias MOTORES Y COMANDO
26	POS_DT3	INT		En ENCENDIDO, MOTORES y REGISTRO_POSICION
27	POS_SOL_AUT1	REAL		Se activa en secuencias Y M_AUTO
28	POS_SOL_AUT2	REAL		Se activa en secuencias Y M_AUTO
29	POS_SOL_AUT3	REAL		Se activa en secuencias Y M_AUTO
30	POS_SOL_AUT4	REAL		Se activa en secuencias Y M_AUTO
31	POS_SOL_MAN1	REAL		Se activa en ENCENDIDO
32	POS_SOL_MAN2	REAL		Se activa en ENCENDIDO
33	POS_SOL_MAN3	REAL		Se activa en ENCENDIDO
34	POS_SOL_MAN4	REAL		Se activa en ENCENDIDO
35	POS_SOLIC_REAL_TH1	REAL		ENCENDIDO, COMANDO, VISUALIZACION, M_AUTO, MOTORES
36	POS_SOLIC_REAL_TH2	REAL		ENCENDIDO, COMANDO, M_AUTO, MOTORES y VISUALIZACION
37	POS_SOLIC_REAL_TH4	REAL		En ENCENDIDO, COMANDO, M_AUTO, MOTORES y VISUALIZACION
38	POS_VIR_PLS	INT		En ENCENDIDO, COMANDO y MOTORES
39	POSIC	BOOL		Se activa en secuencias automáticas
40	PS_MMTS	DINT		Se activa en secuencias Y MOTORES
41	PS_MMTS_REAL	REAL		En ENCENDIDO, MOTORES y VISUALIZACION
42	RECORRIDO	BOOL		Se activa en secuencias, M_AUTO y COMANDO.
43	REF_REAL	REAL		En ENCENDIDO, COMANDO, M_AUTO, MOTORES y VISUALIZACION

Tabla 5-13. Lista N° 5 de variables del programa.

<b>Lista de Variables de la R – T</b>				
<b>#</b>	<b>Nombre</b>	<b>tipo</b>	<b>Dirección</b>	<b>Descripción</b>
1	REVISION	BOOL		Actúa en COMANDO y REGISTRO_POSICION
2	SAL_MOT1	INT		Actúa en MOTORES.
3	SAL_MOT2	INT		Actúa en MOTORES
5	SAL_MOT3	INT		Actúa en MOTORES
6	SAL_MOT4	INT		Actúa en MOTORES
7	SD_DIR_DELTA_3	EBOOL	%Q0.2.27	Gobernada desde MOTORES, sentido del motor 3
8	SD_DIR_THETA_1	EBOOL	%Q0.2.24	Gobernada desde MOTORES, habilita el motor 1
9	SD_DIR_THETA_2	EBOOL	%Q0.2.25	Gobernada desde MOTORES, habilita el motor 2
10	SD_DIR_THETA_4	EBOOL	%Q0.2.26	Gobernada desde MOTORES, habilita el motor 4
11	SD_ENB_DT3	BOOL	%Q0.2.22	Gobernada desde MOTORES, habilita el motor 3
12	SD_PIN_HAB_DELTA_3	BOOL		En secuencias, ENCENDIDO, MOTORES y M_AUTO
13	SD_PIN_HAB_THETA_1	EBOOL	%Q0.2.20	En ENCENDIDO, M_AUTO y MOTORES
14	SD_PIN_HAB_THETA_2	EBOOL	%Q0.2.21	En ENCENDIDO, M_AUTO y MOTORES
17	SD_PIN_HAB_THETA_4	EBOOL	%Q0.2.23	En ENCENDIDO, M_AUTO y MOTORES
18	SD_PINZA	EBOOL	%Q0.2.16	En secuencias, ENCENDIDO, M_AUTO y MOTORES
19	SD_POWER_ON	EBOOL	%Q0.2.18	Gobernada desde MOTORES, energiza la controladora.
20	SEL_AV	INT		En ENCENDIDO, elección de magnitud del avance.
21	selectorEncendido	BOOL		En ENCENDIDO, COMANDO. Inicia el sistema.
22	selectorModoAuto	BOOL		En ENCENDIDO, COMANDO. Elige modo automático
23	selectorModoManual	BOOL		En ENCENDIDO, COMANDO. Elige modo manual
24	selectorRegistroHabilitado	BOOL		ENCENDIDO, COMANDO y REGISTRO_POSICION
25	seletorRevisión	BOOL		ENCENDIDO, COMANDO y REGISTRO_POSICION
26	STOP	BOOL		Se activa en MOTORES
27	THETA1_GRADOS	DINT		En secuencias, MOTORES, COMANDO, REGISTRO_POSICION
28	THETA1_GRADOS_REAL	REAL		En ENCENDIDO, MOTORES y VISUALIZACION
29	THETA2_GRADOS	DINT		En secuencias, MOTORES, COMANDO, REGISTRO_POSICION
30	THETA2_GRADOS_REAL	REAL		En ENCENDIDO, MOTORES y VISUALIZACION
31	THETA4_GRADOS	DINT		En secuencias, MOTORES COMANDO, REGISTRO_POSICION
32	THETA4_GRADOS_REAL	REAL		En ENCENDIDO, MOTORES y VISUALIZACION
33	THETA_1	BOOL		Se activa en Cn_Psn_Vel_T1
34	THETA_2	BOOL		Se activa en Cn_Psn_Vel_T2
35	THETA_4	BOOL		Se activa en Cn_Psn_Vel_T4
36	TR1	BOOL		Transición de marcha del Grafcet Secuencia
37	TR2	BOOL		Transición de marcha del Grafcet Secuencia
38	TR3	BOOL		Transición de marcha del Grafcet Secuencia
39	TR4	BOOL		Transición de marcha del Grafcet Secuencia
40	TR5	BOOL		Transición de marcha del Grafcet Secuencia
41	TR6	BOOL		Transición de marcha del Grafcet Secuencia
42	TR7	BOOL		Transición de marcha del Grafcet Secuencia
43	TR8	BOOL		Transición de marcha del Grafcet Secuencia
44	TR9	BOOL		Transición de marcha del Grafcet Secuencia

Tabla 5-14. Lista N° 6 de variables del programa.

<b>Lista de Variables de la T – V</b>				
<b>#</b>	<b>Nombre</b>	<b>tipo</b>	<b>Dirección</b>	<b>Descripción</b>
1	TR10	BOOL		Transición de marcha del Grafcet Secuencia
2	TR11	BOOL		Transición de marcha del Grafcet Secuencia
3	TR12	BOOL		Transición de marcha del Grafcet Secuencia
4	TR13	BOOL		Transición de marcha del Grafcet Secuencia
5	TR14	BOOL		Transición de marcha del Grafcet Secuencia
6	TR15	BOOL		Transición de marcha del Grafcet Secuencia
7	TR16	BOOL		Transición de marcha del Grafcet Secuencia
8	TR17	BOOL		Transición de marcha del Grafcet Secuencia
9	TR18	BOOL		Transición de marcha del Grafcet Secuencia
10	TR19	BOOL		Transición de marcha del Grafcet Secuencia
11	TR20	BOOL		Transición de marcha del Grafcet Secuencia
12	TR21	BOOL		Transición de marcha del Grafcet Secuencia
13	TR22	BOOL		Transición de marcha del Grafcet Secuencia
14	TR23	BOOL		Transición de marcha del Grafcet Secuencia
17	TR24	BOOL		Transición de marcha del Grafcet Secuencia
18	TR25	BOOL		Transición de marcha del Grafcet Secuencia
19	TR26	BOOL		Transición de marcha del Grafcet Secuencia
20	TR27	BOOL		Transición de marcha del Grafcet Secuencia
21	TR28	BOOL		Transición de marcha del Grafcet Pick and Place
22	TR29	BOOL		Transición de marcha del Grafcet Pick and Place
23	TR30	BOOL		Transición de marcha del Grafcet Pick and Place
24	TR31	BOOL		Transición de marcha del Grafcet Pick and Place
25	TR32	BOOL		Transición de marcha del Grafcet Pick and Place
26	TR33	BOOL		Transición de marcha del Grafcet Pick and Place
27	TR34	BOOL		Transición de marcha del Grafcet Pick and Place
28	TR35	BOOL		Transición de marcha del Grafcet Pick and Place
29	TR36	BOOL		Transición de marcha del Grafcet Pick and Place
30	TR37	BOOL		Transición de marcha del Grafcet Pick and Place
31	TR38	BOOL		Transición de marcha del Grafcet Pick and Place
32	TR39	BOOL		Transición de marcha del Grafcet Pick and Place
33	TR40	BOOL		Transición de marcha del Grafcet Pick and Place
34	U_THETA1	REAL		Señal del control del PID que regula el motor 1
35	U_THETA2	REAL		Señal del control del PID que regula el motor 2
36	U_THETA4	REAL		Señal del control del PID que regula el motor 4
37	VC_POSICIONES	DINT		Matriz de posiciones almacenadas.
38	VF1	DINT		Se activa en REGISTRO_POSICION
39	VF2	DINT		Se activa en REGISTRO_POSICION.
40	VF3	DINT		Se activa en REGISTRO_POSICION.
41	VF4	DINT		Se activa en REGISTRO_POSICION.
42	VR1	DINT		Se activa en COMANDO y REGISTRO_POSICION
43	VR2	DINT		Se activa en COMANDO y REGISTRO_POSICION
44	VR3	DINT		Se activa en COMANDO y REGISTRO_POSICION
45	VR4	DINT		Se activa en COMANDO y REGISTRO_POSICION



## 6 FUNCIONAMIENTO DEL AUTOMATISMO

Como bien se ha señalado en varias ocasiones en este documento, el control automático del robot SONY SRX 4CH que es objeto de estudio de este proyecto de automatización, está destinado a su empleo en un entorno educacional. Por ende, se espera que las configuraciones de trabajo habrán de ser modificadas con frecuencia y además el conjunto de herramientas que se empleen deben estar diseñadas con el propósito de que se puedan emplear de manera intuitiva y con verdadera precisión.

Por tal razón el automatismo a implementar para el manejo del robot está basado en la elaboración de una visualización en el lenguaje UNITY PRO, que se emplea como una Interfaz Hombre Maquina (HMI) que ha de utilizarse regularmente como la consola de programación del SONY, que es mostrada en la ilustración siguiente. De allí que operando sobre los elementos que conforman la consola (pantalla de operador), se han de poder ejecutar tareas básicas para realizar movimientos de las articulaciones del manipulador con miras a ubicarlo en la posición y orientación deseada de su efector y de cada una de sus articulaciones en el espacio de la tarea. Además, se podrá instruir el cierre y apertura del efector para asir objetos o simplemente comprobar su correcto funcionamiento.

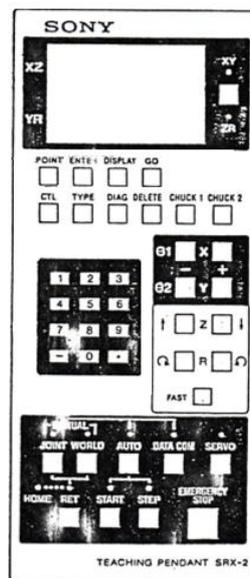


Figura 6-1. Consola de programación original Teaching Pendant.

De la misma manera, desde la pantalla de visualización se permitirá operar sobre comandos más complejos, entre los que se encuentran disponibles: la acción de puesta a home del manipulador, el registro de las posiciones que son alcanzadas por desplazamientos manuales, habilitar el desplazamiento por eje o por coordinación articular, entre otras acciones posibles que se precisen de manera posterior.

El automatismo será operado desde dos modos de funcionamiento: manual y automático. En el modo manual se han de realizar instrucciones de edición y registro de variables como posición y la actuación del efector. En el modo automático se realizan tareas secuenciales a partir de la información que se ha guardado en el modo manual.

Al elegir en la visualización el modo de funcionamiento automático, se podrán realizar el seguimiento de trayectorias y secuencias que sean diseñadas por el usuario, también se puede completar la ejecución de las tareas previamente configurada para recoger, transportar y colocar piezas (Pick and Place), que se han incluido en las secciones de programa creadas para demostrar el funcionamiento del robot.

Como se acostumbra en el entorno industrial, en la interfaz se han incluidos pulsadores de arranque (start) para darle marcha al sistema y de paro (stop) para detenerlo de manera controlada. En la unidad controladora de potencia del robot ya está incluido un pulsador seta de emergencia, además de los dispuestos a lo largo de la célula, en la estación de trabajo también se ha incluido uno, para detener de manera abrupta la ejecución de las

tareas del autómatas ante la observación de una condición que está incurriendo el manipulador robótico o que pudiere devenir en un estado de anomalía o de alto riesgo de deterioro de los equipos o en el peor de los casos, en el daño físico de las personas que interactúan con el entorno productivo.

Este orden procedimental descrito, obedece a lo planteado en las guías GEMMA diseñadas para el control de procesos automatizados y que se explicará brevemente más adelante. Para obtener una visión más amplia de lo comentado, a continuación, se describen los procedimientos básicos que se han empleado usualmente en el lenguaje ACL para los SCORBOTS V5 del laboratorio de robótica de la escuela y que son considerados para formar parte del menú de instrucciones que se deben incluir en la librería de funciones del robot SCARA del proyecto.

El conjunto de procedimientos elegidos puede ser discutido en razón a su elección, pudiesen haberse tomado otros con propósitos más específicos. Más lo pretendido fue desarrollar aquellos que se han de demandar en la solución de la necesidad de habilitar el robot y dotarlo de las prestaciones operativas necesarias para emplearlo en la configuración de procesos automáticos en la célula de fabricación.

## 6.1 LA PUESTA A HOME (HOMING PROCESS)

El proceso de puesta a home, que se realiza en el modo manual, es una secuencia que permite observar si el robot se encuentra operativo en todos sus elementos móviles. Además, el procedimiento conduce al manipulador a posicionarse en un punto de partida que el controlador ajusta como posición cero (0) y desde la cual todos los codificadores (encoders) de posición, comienzan a registrar los pulsos que se traducen a unidades de distancia y que a medida que se desplazan representarán un punto en el espacio de la tarea que la configuración del robot permite alcanzar. Para el proceso se consideran inicialmente las siguientes estrategias:

Realizar la prueba por articulación comprobando que el sentido de giro del motor se corresponde con los incrementos o decrementos de los pulsos del encoder en la medición del desplazamiento.

Avanzar hasta la detección del sensor que señala el límite de carrera inferior y ajustar en este punto el punto de partida de contaje del encoder.

## 6.2 CONTROL DE POSICIÓN

Este procedimiento es esencial en la implementación del proyecto. Se efectúa en el núcleo del controlador y es invisible para el usuario. En su construcción se ha partido de las condiciones electromecánicas actuales del SCARA y se consideran dos lineamientos generales a seguir.

Si el movimiento de la articulación está gobernado por un servomotor que tenga asociado un encoder. Se ha de ejecutar una rutina que estime las posiciones angulares a partir de las magnitudes que se registran en las variables incrementales del contaje rápido. Si el movimiento de la articulación está sujeto a un motor a pasos, el desplazamiento se estima a partir de la manipulación aritmética del número de divisiones o pasos que el motor posee para completar una revolución entera. En el manipulador es necesario considerar las relaciones de engranajes que el conjunto motriz de la articulación posee. Siendo que, para cada servo motor, el encoder está directamente acoplado al eje motriz, por ende, se amerita conocer las relaciones de las reductoras en función de precisar la magnitud del desplazamiento angular o longitudinal que obedece al giro del motor.

## 6.3 CONTROL DE VELOCIDAD.

El control de velocidad del desplazamiento, al igual que el control de posición no es editable por el usuario en su configuración, sin embargo, se realiza mediante el seguimiento de un perfil de incremento y decremento de la tensión que se le aplica al motor. La regulación de la velocidad se ha de someter a un más profundo desglose, pero se puede comentar llanamente que ha de cambiar en función de la distancia que ha de recorrer el enlace o eslabón para alcanzar la posición fijada como destino.

La distancia del recorrido se conoce en todo momento con el procedimiento del apartado anterior, mientras que

el avance articular se realiza, por ello es posible adaptar la magnitud de tensión a medida que se recorre la trayectoria y aprovechar la magnitud para potenciar las cualidades de posicionamiento.

Además de los ya mencionados procedimientos, ha sido necesaria la inclusión de un grupo de comandos que a diferencia de los procedimientos anteriores que pertenecen completamente a la unidad de control del SCARA, estos doten al usuario de herramientas de edición y registro de posiciones, trayectorias y secuencias de movimientos y acciones, que puedan ser concatenadas en la resolución de tareas de manipulación y transporte de objetos u otra que el usuario desee completar.

## **6.4 REGISTRO DE POSICIÓN**

Este comando que se ejecuta en modo manual permite registrar en la memoria del controlador las posiciones de la pinza que hayan sido alcanzadas por el avance de las articulaciones del manipulador. La posición creada es un arreglo matricial que contiene las variables que registran en tiempo real los encoders acoplados a las articulaciones del robot [ $\Phi_1$ ,  $\Phi_2$ ,  $d_3$ ,  $\Phi_4$ ]. Es de entender que para poder crear una trayectoria es necesario, registrar al menos una posición a la que el efector se desplace desde la posición cero (0) que ha sido definida como resultado del proceso de home.

## **6.5 INICIO DE MOVIMIENTO**

El movimiento del robot se ejecuta en ambos modos de operación, y en su ejecución se habilita el control de posición y velocidad. Si la instrucción es ejecutada en modo manual el movimiento se realiza a una velocidad limitada por el controlador, que ha de comprender que el movimiento que se realiza está siendo empleado en labores de configuración de posicionamiento. Por otro lado, se la instrucción es lanzada en modo automático, la velocidad del robot obedece a la elegida en la secuencia y a partir de la cual el controlador ha de interpolar en la ejecución de movimientos.

## **6.6 DESPLAZAMIENTO ARTICULAR**

Cuando se realiza un movimiento el controlador puede comprender que se trata de un desplazamiento simple entre dos coordenadas espaciales a visitar y estimará un trazo libre de restricciones por el cual se ha de desplazar para alcanzar la configuración de la posición destino mediante el avance individual de los enlaces.

## **6.7 DESPLAZAMIENTO ESPACIAL**

Por otro lado, si el movimiento se configura como cartesiano el desplazamiento se hace de manera rigurosa en término de precisión espacial. Para lograr realizar un movimiento del efector que sea resultado de la estimación de la ubicación en el espacio de la tarea que obedezca a la definición del sistema de coordenadas tridimensionales, con el avance en conjunto de la cadena de enlaces del SCARA.

## **6.8 APERTURA Y/O CIERRE DE LA PINZA**

La apertura del efector del robot se ejecuta en ambos modos de operación, manual y automático. El efector es accionado por una pinza neumática modelo MHQ2-20D, de la empresa SMC. Para accionar este actuador se ha dotado al robot de una electroválvula modelo VJ3143, que ha sido fabricada por la misma casa comercial.

La ilustración 6-2 muestra las dimensiones del efector colocado en el extremo del SCARA.

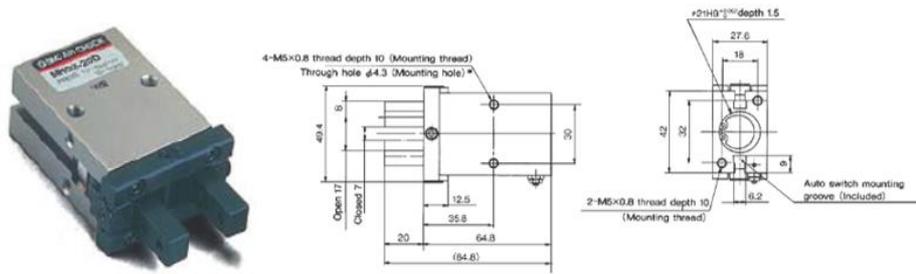


Figura 6-2. Dimensiones de la pinza de sujeción de piezas SMC MHQ2-20D.

Con la intención de conmutar eléctricamente la válvula se emplea una de las salidas digitales del módulo BMX DDM 3202 K, dicha señal al estar en alto (encendido) es llevada desde el autómatas al solenoide que desplaza el vástago de la válvula 5 vías dos posiciones EV1. Direccionando el avance del actuador doble efecto que cierra las articulaciones de la pinza y sujeta entre ellas la pieza a manipular.

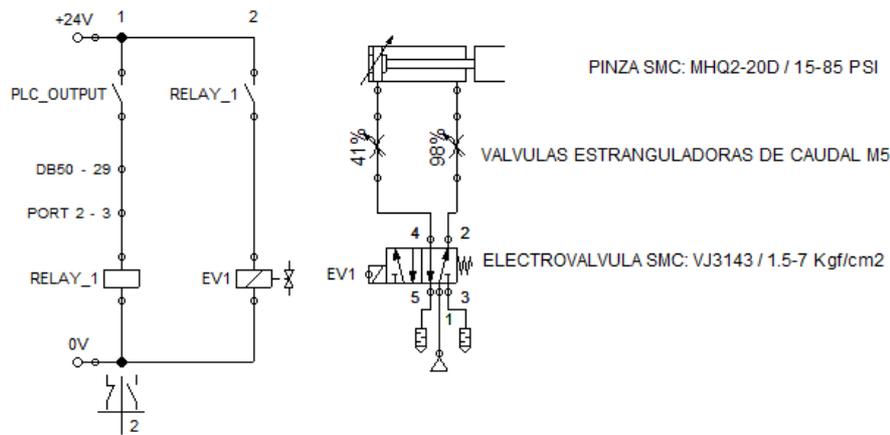


Figura 6-3. Sistema neumático de la pinza de sujeción de piezas simulado en Festo Fluidsim.

En la figura anterior además se puede apreciar que al ser puesta a bajo (apagado) nuevamente la señal que energiza el relé 1, el muelle hace retornar la electroválvula a su posición de relajación, llevando al fluido neumático a llenar las cavidades del actuador en sentido contrario. Esto obliga a abrir nuevamente la pinza a la rapidez que definen la apertura de las válvulas estranguladoras de caudal adosadas al efector.

### 6.9 FUNCIÓN COGE Y COLOCA (PICK AND PLACE)

En modo automático se podrá ejecutar secuencialmente con el SCARA la recogida de objetos desde una posición que es elegida por el usuario, trasladarlos a otra posición destino también de elección del operador y colocarlos apropiadamente según la configuración articular elegida. Este procedimiento se puede desencadenar como respuesta de una de las entradas digitales del autómatas o como resultado de las instrucciones de la red de control que posee la estación de trabajo.

## 7 ESTRUCTURA DEL PROGRAMA

Para exponer el programa del autómata que comandará el robot SCARA se ha hecho uso de una Red de Petri. Este diagrama usualmente empleado en sistemas informáticos permite exponer las acciones que realiza el controlador al estar dadas tanto las condiciones secuenciales como combinatoriales, que son necesarias para ejecutar una operación dentro del esquema global que se ha planteado para realizar la operación de la unidad manipuladora.

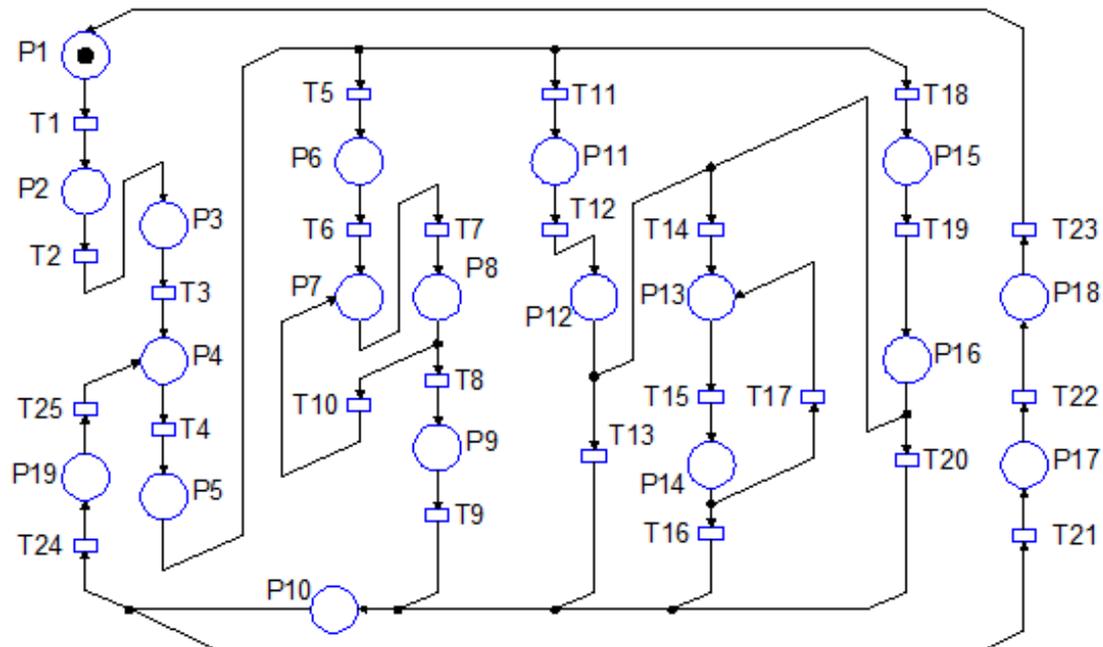


Figura 7-1. Red de Petri del proceso a automatizar

Las Redes de Petri están regidas por una extensa lista de relaciones lógico-aritméticas que aseguran el adecuado comportamiento del proceso que modelan, sin embargo, la secuencia que está descrita con la Red de Petri de la figura previa, y que ha sido diseñada para el automatismo del robot Sony del proyecto, solo ha de representar la ruta a seguir desde el momento que el usuario de la estación robótica enciende, comienza a hacer empleo y finalmente detiene la utilización del SRX 4CH.

La red encuentra en las siguientes tablas la enumeración y descripción de sus pasos y transiciones.

Tabla 7-1. Lista de pasos de la Red de Petri que describe el proceso.

Paso	Descripción	Paso	Descripción
<b>P1</b>	En espera de encendido.	<b>P11</b>	Configuración de una secuencia.
<b>P2</b>	Puesta a Home.	<b>P12</b>	Guardar la secuencia.
<b>P3</b>	En espera de operaciones.	<b>P13</b>	Modo automático habilitado.
<b>P4</b>	Sistema habilitado.	<b>P14</b>	Ejecución de marcha continua.
<b>P5</b>	Selección de modo de operación.	<b>P15</b>	Modo Pick & Place habilitado.
<b>P6</b>	Modo Manual habilitado.	<b>P16</b>	Guardar secuencia Pick & Place.
<b>P7</b>	Selección del eje o articulación del movimiento.	<b>P17</b>	Salir del sistema.
<b>P8</b>	Ejecución de movimiento en manual.	<b>P18</b>	Nueva puesta a Home de seguridad.
<b>P9</b>	Guardado de posición alcanzada.	<b>P19</b>	Salir al reajuste
<b>P10</b>	En espera del modo de salida del sistema.		

Tabla 7-2. Lista de pasos y transiciones de la Red de Petri que describen el proceso.

Transición	Descripción	Transición	Descripción
<b>T1</b>	Encendido	<b>T14</b>	Habilitado de marcha continua.
<b>T2</b>	Puesta a home completada.	<b>T15</b>	Desencadenador de marcha, pulsador o sensor.
<b>T3</b>	Inicio de maniobras.	<b>T16</b>	Modificar secuencias.
<b>T4</b>	Habilitación confirmada.	<b>T17</b>	Espera de nueva ejecución de secuencia.
<b>T5</b>	Selector a modo manual.	<b>T18</b>	Selector a modo Pick & Place.
<b>T6</b>	Selector de modo de movimiento del robot.	<b>T19</b>	Ajustes de la secuencia Pick & Place.
<b>T7</b>	Pulsador de avance o de retorno.	<b>T20</b>	Abortar secuencia.
<b>T8</b>	Posición alcanzada.	<b>T21</b>	Selección de salida del sistema.
<b>T9</b>	Registro de la posición completa.	<b>T22</b>	Finalizar operaciones con el robot.
<b>T10</b>	Cambio del eje de movimiento.	<b>T23</b>	Puesta a home completa.
<b>T11</b>	Selector a modo secuencial.	<b>T24</b>	Selección de salida a parametrizar.
<b>T12</b>	Secuencia configurada.	<b>T25</b>	Continuar operaciones.
<b>T13</b>	Abortar secuencia.		

Inicialmente se ha señalado que este proyecto de automatización está orientado a la creación de un paquete de herramientas para ser empleadas en la configuración de tareas dentro de la estación del robot SCARA. La investigación explota la potencialidad de creación de instancias de programación que la mayoría del software desarrollado bajo la norma IEC 61131, como UNITY PRO, incluye. Esta posibilidad se deriva de la reproducción de bloques de función definidos arbitrariamente por el usuario-programador que comprendiendo la naturaleza de su proyecto de automatización, procura realizar unidades inteligentes previamente configuradas bajo las prestaciones de desempeño en tareas específicas para cada bloque.

Estos bloques que el Unity Pro llama DFB (bloques de función derivados) se pueden agrupar en listas donde estos guardan relación entre sí, siendo estas listas comúnmente configuradas bajo la tipología de aplicación a la cual se han de destinar posteriormente. Estos conjuntos de bloques de usuario reciben el nombre de librerías y son de gran utilidad en las labores de ordenamiento con el empleo del software.

La librerías permiten empaquetar las configuraciones de programa que se utilizan con más regularidad, se llega a reducir en extensión los programas que se elaboran y además con el uso de librerías se puede disponer de los bloques de manera habitual, se evita así realizar tareas repetitivas y se consigue construir un set de herramientas para simplificar la labor, alcanzando inclusive estandarizar los procedimientos cuando la creación y supervisión de los esquemas de control es llevada a cabo desde más de una estación de trabajo y por más de un técnico programador, debido a que como en el caso de UNITY PRO, es posible compartir las librerías entre las diferentes unidades, terminales y estaciones.

Este planteamiento encauza a realizar de una manera más amigable, la configuración de operaciones dentro del entorno Unity Pro, llevando de un escenario de desarrollo en el muy extendido lenguaje netamente realizado en texto estructurado, como el ACL que se emplea en la programación de algunos elementos de la célula de fabricación y del *SCORBOT* o el antiguo *LUNA* que se usaba para el SRX que se estudia, hacia un entorno de desarrollo gráfico como lo es el lenguaje de programación bajo diagramas de bloque de función que es provisto para la configuración del autómeta.

Sin duda este modo de operación, repercutirá en la aplicabilidad de la parametrización del manipulador, que es lograda con el manejo de esta librería contentiva de las acciones descritas en los pasos de la red de Petri, que no es más que la representación de la secuencialidad del automatismo, donde el usuario en vías de crear una serie de pasos de trabajo para el traslado de piezas en la célula, se encargará de insertar e interconectar según desee los bloques funcionales que el robot ejecute en su accionar.

En Unity Pro tanto los bloques DFB, como las librerías y las familias de estas son dinámicas, por lo tanto, a medida que la utilización del SCARA demande la construcción de maniobras más específicas, más especializadas o la modificación en pro de la mejora de desempeño, estas se pueden incluir como bloques auxiliares que se adicione al proceso o que sustituyan por completo a los que se han ofrecido en este proyecto.

Lo que se entiende como cooperación parcial con los existentes o la eliminación total de estos. Teniendo cuidado en la reproducción de los bloques en instancias y en la utilización de palabras reservadas dentro del entorno de programación. Siendo un ejemplo de lo comentado el siguiente, la instrucción de desplazamiento del robot no puede llamarse *move*, por estar previamente definida en el software para el almacenamiento de los valores de una variable en una ubicación de memoria. Unity Pro estandariza un procedimiento para cada una de las operaciones que se han mencionado en este apartado y se han de referenciar al momento de ejecutar alguna de dichas operaciones.

## 7.1 DESCRIPCIÓN DE LA GUÍA GEMMA.

Por otro lado, la estructuración del programa debe seguir análogamente al modo que usualmente se emplea en la Célula de Fabricación Flexible que contiene al SRX 4CH. Esta rige su operación bajo la implementación de la guía GEMMA desarrollada por la agencia nacional francófona para el desarrollo de la productividad aplicadas a la industria ADEPA (de sus siglas en francés: Agence Nationale pour le Développement de la Productique Appliquée à L'Industrie), con miras a estandarizar los modos de arranque y de paradas de los procesos automatizados.

Por lo tanto, el sistema implementado en el SCARA sigue el concepto. El cual conduce a seguir un método gráfico para definir las condiciones que desencadenan los cambios de modos operativos y la transición entre los estados del proceso productivo automatizado.

A continuación, se ofrece una breve descripción de cada uno de los 17 estados que se pueden alcanzar en un proceso productivo industrial y que han sido contemplados en la guía GEMMA. Además, se señala cómo estos son aplicados en el automatismo del robot, por medio de la observación de la figura 7-2 que es presentada de manera ulterior.

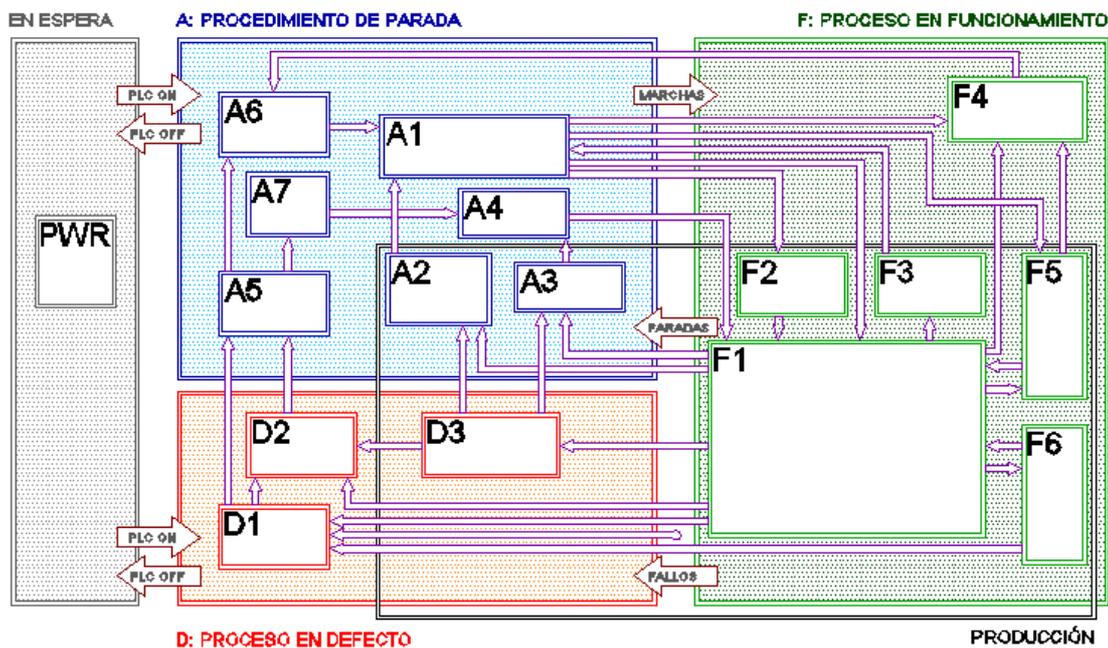


Figura 7-2. Diagrama de estados de la Guía GEMMA.

Es posible observar que la guía constituye una estrategia sistemática para integrar y entrelazar las interacciones que se han de llevar a cabo entre los operadores y el sistema de control con el propósito de dar respuesta, bien sea de manera reactiva o bajo las previsiones que se han incluido anticipadamente para ser ejecutadas por parte del control, empaquetadas en consignas de accionamiento manual, funcionamiento semiautomático y/o automático, paradas solicitadas o de emergencia, marcha condicionada o continua, enmarcadas todas dentro cuatro bloques principales que se denominan según su estado como: control sin alimentación o en espera, proceso en funcionamiento, procedimientos de parada y en proceso con defecto.

### 7.1.1 LOS ESTADOS DE LA GUÍA GEMMA.

El primero de los estados que se considera que ha de adoptar el autómatas dentro del proceso está definido por su energización o *PWR OFF*, de donde se encuentra el proceso *EN ESPERA*. Al ser este conmutado en alto, es decir, encender el autómatas, se podrán alcanzar alguno de los estados de los *PROCEDIMIENTOS DE PARADA* o algunos de los estados del *PROCESO CON DEFECTOS*.

En los *PROCEDIMIENTOS ANTE DEFECTOS D*, se agrupan en la necesidad de realizar una *PARADA DE EMERGENCIA D1*, esta detención abrupta y por consiguiente inmediata del proceso, conduce a un estado de interrupción total y segura del sistema. Es ejecutable por el accionamiento de instrumentos y/o sensores del equipo o por decisión del usuario mediante la pulsación de la seta de emergencia. Este estado es alcanzable desde cualquiera de los estados de la guía y para salir de él es preciso desbloquear mecánicamente la seta de emergencia o corregir la condición que llevó a la parada desde el automatismo.

Asimismo, abarca la realización de *DIAGNÓSTICO Y/O TRATAMIENTO DE FALLOS D2*, dicho estado se alcanza como resultado de la revisión de la máquina tras la ocurrencia del defecto que condujo a *D1*, es realizable con o sin intervención del operador para determinar la razón de la anomalía. El automatismo aguarda por el rearme de los elementos que han suscitado la detención que permitirán la habilitación del sistema para su nueva puesta en marcha.

En ocasiones se decide continuar la marcha del sistema a sabiendas del fallo que se ha observado, este estado también se encuentra dentro del grupo y se le denomina *PRODUCCIÓN A PESAR DE LOS DEFECTOS D3*, y se establece con la intención de completar la producción del recurso que se está transformando o de llevar el sistema a un estado que se considere más seguro y confiable para la realización de una parada más extensa en miras de subsanar el inconveniente que conduce al mal funcionamiento. En el SCARA los tres estados son lanzados por la pulsación de tres Setas de Emergencia con enclavamiento mecánico, ante la ausencia de presión neumática o por la consigna externa, estas dos últimas se le han de instruir desde la red informática de la célula al momento de integrarlo operativamente a ella.

A los *PROCEDIMIENTOS DE PARADA A*, pertenecen todos los estados a los que se ha llegado desde el encendido del autómatas, por finalización de tareas o por incidencia de errores y en ellos el automatismo se encuentra en la espera del comando de marcha para dar partida a cualquiera de los procedimientos de funcionamiento.

El primero de los estados del bloque es la *PARADA EN EL ESTADO INICIAL A1*, a este se ha llegado después de que el sistema se ha posicionado en la condición de partida segura. En el SRX 4CH este punto lo conformará la realización con éxito de la puesta a HOME.

Otro estado definido en la guía para este bloque es la *PARADA PEDIDA AL FINAL DE CICLO A2*. Este modo operacional es empleado para llevar al automatismo a completar el ciclo productivo actual, de manera que ante la solicitud de parada normal se logre llegar a la detención del sistema en su estado inicial A1.

Sin embargo, el proceso puede detenerse en un estado determinado que no coincide con el final de ciclo, este estado igualmente está agrupado en el bloque y se le conoce como *PARADA SOLICITADA EN UN ESTADO DETERMINADO A3*, y es alcanzable desde el procedimiento productivo normal o desde un proceso defectuoso. El estado *A3* es un procedimiento transitorio que evoluciona hacia el estado *A4* denominado *PARADA OBTENIDA*. Este es una puesta en reposo del proceso desde donde, al igual que en *A1*, se puede alcanzar la marcha funcional del sistema.

El bloque de estados también permite restablecer los estados de producción defectuosa mediante la *PREPARACIÓN PARA LA PUESTA EN MARCHA DESPUÉS DE UN DEFECTO A5*, y allí se enmarcan las actividades que se ameritan para poner la máquina nuevamente en funcionamiento posterior a la ocurrencia de un desperfecto.

Finalmente, posterior al encendido o restablecimiento de errores el sistema puede aguardar en reposo en dos condiciones distintas definidas por los estados de *PUESTA DEL SISTEMA EN EL ESTADO INICIAL A6* o una *PUESTA DEL SISTEMA EN UN ESTADO DETERMINADO A7*, en donde la diferencia entre ambos radica en la espera en el punto de partida del proceso o en aguardar a una puesta en ejecución en una situación concreta diferente de la inicial.

La guía es cumplimentada con el bloque de estados que agrupan los *PROCEDIMIENTOS DE*

*FUNCIONAMIENTO F*, en los que el sistema se encuentra en marcha. El primero de los estados lo compone la *PRODUCCIÓN NORMAL F1*, en donde el manipulador se encuentra en marcha continua y automática, siendo el modo típico de operación que desarrolla el automatismo diseñado para el robot.

El manipulador precisa del conocimiento de su ubicación antes de entrar en el modo automático, razón por la cual realiza una puesta a punto previa, este procedimiento en el SCARA se ha señalado como puesta a HOME del robot y ha sido considerado en la guía como el estado de *MARCHA DE PREPARACIÓN F2*, este modo garantiza el funcionamiento correcto y preciso del sistema durante la marcha normal.

Análogo a este estado la guía contiene la *MARCHA DE CIERRE F3*, un estado del sistema donde se ha solicitado la detención del sistema de manera planificada. Y se encarga de mantener el sistema en operación hasta que el último ciclo productivo es finalizado y el recurso a transformar es conformado hasta ser agotado o completado el proceso. Para el SONY este modo se empleará para realizar una nueva puesta a HOME final del manipulador posterior a su utilización.

El manipulador amerita la comprobación de funcionamiento de los actuadores y la precisión de sus desplazamientos, también demanda la enseñanza de configuraciones articulares para la configuración de trayectorias, por ello utiliza el control manual de la unidad. Este se denomina dentro de la guía como *MARCHAS DE VERIFICACIÓN SIN ORDEN F4*, permite al operario accionar directamente el sistema usando un interfaz hombre máquina o un terminal de operador desde un SCADA.

De manera similar las *MARCHAS DE VERIFICACIÓN CON ORDEN F5*, se usan para comprobaciones del automatismo a una velocidad de operación definida por el operador, a la cual es capaz de certificar que las condiciones de desempeño de la maquinaria alcanzan las especificaciones estipuladas en el diseño del procedimiento de marcha. La unidad controladora original consideró un procedimiento ejecutable desde la consola de programación con el pulsador STEP, en donde el robot ejecutaba el programa de ejecución a baja velocidad y a pasos, donde el avance a la próxima instrucción ameritaba la comprobación del operario.

Otro estado de funcionamiento de la guía GEMMA es la realización de *MARCHAS DE TEST F6*, estas son una prueba al vuelo de los elementos que están insertados en el sistema. Esto permite determinar si existen fallos en el automatismo durante su ejecución. En el manipulador este modo está empleado dentro de la tarea de puesta a HOME. Allí se comprueba, por ejemplo, si el sentido de giro corresponde con la consigna de movimiento elegida y si el número de pulsos que son leídos define correctamente las dimensiones del espacio de trabajo del efector.



## 8 CONEXIÓN DE DRIVERS DE POTENCIA

La unidad de potencia del SCARA emplea para los actuadores de las articulaciones tres convertidores DC-DC para energizar los motores de corriente continua y uno especialmente diseñado para el manejo de motores paso a paso. En la siguiente sección se describen con detalle.

### 8.1 LOS SERVOMOTORES.

El dispositivo controlador de motores de corriente directa con escobillas 20A14 fabricado por la empresa estadounidense Advance Motion Controls (AMC), se ha de alimentar por una fuente de corriente continua de 127 voltios, posee un arreglo en H formado por transistores de potencia para operar en la modulación de ancho de pulsos PWM en alta frecuencia, así logra aplicar una señal de actuación estable sobre el motor de naturaleza suave y sin detrimento de la aplicación de potencia por parte del motor. Los dispositivos están provistos de un diodo emisor de luz que se torna en verde para indicar que el elemento está habilitado para operar, o cambia a rojo si existiere un fallo. Además, el módulo está protegido ante disminuciones o excesos de la tensión de alimentación, excesos de corriente de consumo, ante el sobrecalentamiento y corto circuitos de los conductores del motor, del aterramiento y de las líneas de alimentación del driver. Incluye un bloque de entradas y salidas para su maniobra y supervisión remota.



Figura 8-1. Driver para motores DC con escobillas Advanced Motion Controls.

Cada servo amplificador posee un set de potenciómetros de múltiples vueltas (14) para el ajuste de las ganancias del lazo del controlador proporcional, integral y derivativo (PID) interno, de la misma manera emplea uno de ellos para limitar el máximo pico de la corriente de arranque del motor, emplea un tercer reóstato para el ajuste de la sensibilidad ante las referencias externas y también es incluido un potenciómetro para el ajuste de las respuestas de los transistores del puente H, de manera de lograr balancear el comportamiento del avance del motor en ambos sentidos de rotación de su eje.

El mismo se puede emplear con la intención de ejecutar pruebas de giro cuando el cuarto interruptor *dip* es conmutado en alto.

En resumen, el A2014K, como el mostrado en la figura 8-1, es un dispositivo de simple aplicación, robusto y confiable. Por otro lado, es un elemento que ha dejado de ser fabricado unas décadas atrás y por ello fue discontinuado por parte del fabricante, debido a que en los últimos años ha renovado su oferta de drivers. Razón por la cual la adquisición de buena información no fue una tarea trivial.

Y la manera más expresa de aclarar dudas se consiguió por medio de la consulta directa con la empresa y así no incurrir en errores que pudieran deteriorar al accionamiento completo (driver-motor). Por consiguiente, la configuración que se ha completado con éxito es el arreglo del conexionado, que como se ilustra en la figura

que sigue, ha quedado dispuesto en la unidad de potencia del robot.

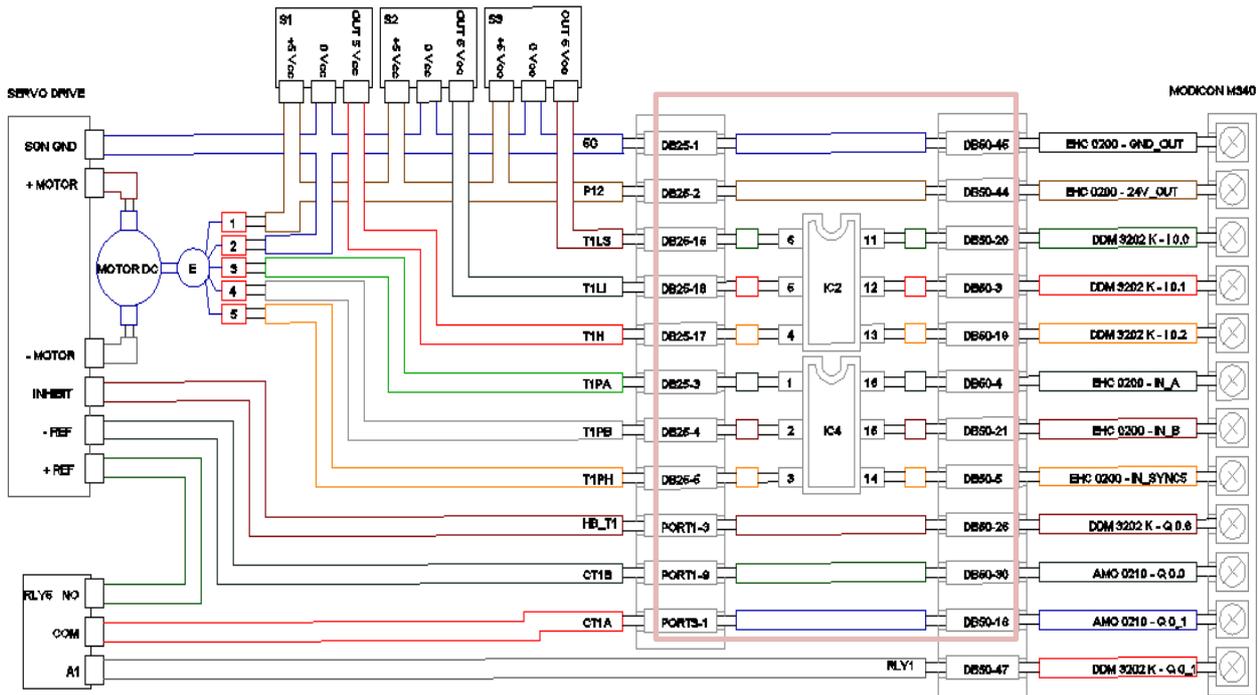


Figura 8-2. Diagrama de conexión de accionamiento de articulaciones rotacionales.

Es importante señalar que las entradas del autómata indicadas en el esquema son la referencia genérica de los terminales que se utilizan según el tipo de entrada o salida que demanda la aplicación, en otras palabras, el terminal indicado no apunta al borne exacto al cual han sido conectada las variables del servo-accionamiento de la primera articulación rotacional.

La asignación de entradas y salidas definitiva fue listada de manera previa en la sexta sección de este documento, con el apartado que define la vinculación de las señales al autómata.

Siendo de utilidad referir nuevamente en las tablas que se añaden a continuación cada una de las rutas que siguen los conductores desde el autómata en dirección a cada uno de los dispositivos que se emplean tanto en el robot, como en la unidad de potencia.

Tabla 8-1. Conexiones PLC – Controladora, Controladora – ROBOT, primera articulación.

Lista de Conexiones del Autómata al Robot						
DB25	IC-I	IC-O	DB50	Color	Símbolo	Descripción
DB25-15	IC2-6	IC2-11	DB50-20	OR	T1LS	Final de carrera del límite superior.
DB25-16	IC2-5	IC2-12	DB50-3	BK	T1LI	Final de carrera del límite inferior.
DB25-17	IC2-4	IC2-13	DB50-19	GN	T1H	Final de carrera del home.
DB25-3	IC4-1	IC4-16	DB50-4	WH	ET1	T1PA Encoder de theta 1 cable # 3.
DB25-4	IC4-2	IC4-15	DB50-21	GN	ET1	T1PB Encoder de theta 1 cable # 4.
DB25-5	IC4-3	IC4-14	DB50-5	BN	ET1	T1PH Encoder de theta 1 cable # 5.
Port3-1			DB50-16	GY	CT1A	Consigna de Theta 1.
Port3-5			DB50-47	RD	CT1B	Consigna de Theta 1.
Port1-9			DB50-30	BK	HB_T1	Inhibit 1 Servo Drive

Para la articulación theta 2 descrita en la tabla presentada a continuación, el esquema de conexión es el mismo

que el mostrado en la figura 8-2, que ha sido utilizada para describir la articulación theta 1.

Tabla 8-2. Conexiones PLC – Controladora, Controladora – ROBOT, segunda articulación.

Lista de Conexiones del Autómata al Robot						
DB25	IC-I	IC-O	DB50	Color	Símbolo	Descripción
DB25-18	IC2-3	IC2-14	DB50-2	BU	T2LS	Final de carrera del límite superior.
DB25-19	IC2-2	IC2-15	DB50-18	GN	T2LI	Final de carrera del límite inferior.
DB25-20	IC2-1	IC2-16	DB50-1	BK	T2H	Home de la articulación theta_2.
DB25-7	IC4-4	IC4-13	DB50-22	YL	ET2	T2PA Encoder de theta 2 cable # 3.
DB25-7	IC4-5	IC4-12	DB50-6	OR	ET2	T2PB Encoder de theta 2 cable # 4.
DB25-8	IC4-6	IC4-11	DB50-23	WH	ET2	T2PH Encoder de theta 2 cable # 5.
Port3-2			DB50-17	VT	CT2A	Consigna de Theta 2.
Port3-6			DB50-50	BK	CT2B	Consigna de Theta 2.
Port1-10			DB50-14	VT	HB_T2	Inhibit 2 Servo Drive

A diferencia de las dos primeras articulaciones rotacionales, la cuarta está configurada de manera distinta. En ella se utiliza un único final de carrera para realizar la puesta a Home y el terminal de sincronización del encoder tampoco se encuentra cableado, por tal razón la lista de rutado de los conductores queda como lo indica la próxima tabla.

Tabla 8-3. Conexiones PLC – Controladora, Controladora – ROBOT, cuarta articulación.

Lista de Conexiones del Autómata al Robot						
DB25	IC-I	IC-O	DB50	Color	Símbolo	Descripción
DB25-24	IC1-1	IC1-16	DB50-37	GN	T4H	Final de carrera del home.
DB25-12	IC3-2	IC2-15	DB50-8	GN	ET4	T4PA Encoder de theta 4 cable # 3.
DB25-13	IC3-3	IC3-14	DB50-25	GY	ET4	T4PB Encoder de theta 4 cable # 4.
Port3-4			DB50-33	VT	CT4A	Consigna de Theta 4.
Port3-8			DB50-48	BU	CT4B	Consigna de Theta 4.
Port1-12			DB50-15	BN	HB_T4	Inhibit 4 Servo Drive

## 8.2 EL MOTOR A PASOS.

En el caso de la articulación prismática Delta 3, ha resultado modificada de su configuración original, que correspondía a un servo-accionamiento como los anteriores, dicha modificación se precisó debido a que el motor original sufrió daños que lo inhabilitaron por completo. Por tal razón fue reemplazado por un motor a pasos NEMA 17 que posee las características eléctricas de la tabla, así como las dimensiones y conexiones de la figura que son mostradas a continuación.

Tabla 8-4. Especificaciones del Motor Paso a Paso.

Especificaciones del Motor Paso a Paso					
<b>Fabricante</b>	Mercury Motor		<b>Modelo</b>	SM42BYG01111-25	
<b>Tensión</b>	12 V	<b>Corriente</b>	0.33 A	<b># Fases</b>	2
<b>Angulo del Paso</b>	1.8° ± 5%		<b>Aislamiento</b>	Clase B	

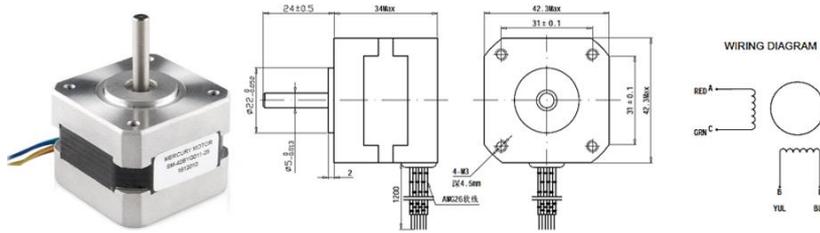


Figura 8-3. Dimensiones y bornes del motor paso a paso.

Ante la modificación ha sido preciso cambiar la disposición de los conductores que fueron previamente conseguidos en el arreglo anterior de la articulación e implementar un planteamiento diferente de manejo del motor.

La estrategia parte del diseño de la secuencia de excitación de los arrollados de los polos magnéticos. Sabiendo de la tabla de especificaciones del nuevo actuador que cada excitación individual de bobina permitirá obtener 1.8° de giro, lo que implica que para un giro de la flecha del motor se podrían conseguir 200 pasos vuelta. Y si en un momento se ameritase mayor precisión, por el gobierno de la tensión y la corriente que circula las espiras del estator se pueden obtener fracciones del paso, procedimiento que es conocido como *microstepping* (generación de micropasos).

Esta particularidad del motor se entiende como una mejora evidente de la precisión en el posicionamiento, sin contar que además las revoluciones del motor se ven afectadas por la relación de la transmisión que posee la articulación, llegando inclusive a fraccionar más aún el recorrido del efector.

Sin embargo, se ha perdido la habilidad de retroalimentar la posición por medio del encoder, por lo que se ha de confiar en las garantías del cálculo del programa que estima la relación de avance con el tren de pasos del motor y regula el desplazamiento longitudinal.

El motor a pasos bipolar utilizado en la articulación traslacional del SCARA, demandó la inserción de una interfaz electrónica de potencia que simplificase las acciones de control sobre el motor. Que de manera autónoma y efectiva se encargue de habilitar su marcha, de realizar la alimentación secuencial de las bobinas del paso a paso y que lo lleve a marchar de manera correcta, inclusive ante cambios del sentido de rotación.

Esa simplificación en el mando del motor a pasos se llevó a cabo incluyendo en la unidad de potencia al Stepper Driver (dispositivo controlador de motor a pasos) HY-DIV268N-5A, que usualmente se emplea en la célula de fabricación flexible para el manejo de este tipo de motores. Este encapsulado posee dos convertidores DC-DC trabajando en conjunto, mayormente conocidos como puentes H, para poder conmutar cada bobinado del motor paso a paso.



Figura 8-4. Driver para motores paso a paso HY-DIV268N-5.

Además, el driver, que se ilustra en la figura 8-4, cuenta con la configuración necesaria para que con solo tres señales de consigna se pueda realizar una adecuada manipulación del actuador, habilitación (ENABLE), dirección de giro (DIR), y la secuencia de pasos (STEP). En la siguiente tabla se listan las características más relevantes para la aplicación de este circuito integrado en la motorización de la articulación delta 3 del robot SRX 4CH.

Tabla 8-5. Especificaciones del Driver para motores a pasos HY-DIV268N-5.

Especificaciones del HY-DIV268N-5 <sup>a</sup>					
<b>Tensión</b>	12 – 48 V	<b>Corriente</b>	0.2 - 5 A	<b>Micropasos</b>	1 a 16 divisiones por paso
<b>Tensión de consigna</b>	24Vcc con resistencia de 2 KΩ	<b>Temp. Ambiente</b>	-10 a 45 °C		

De igual manera la figura 8-5, ilustra cómo se han vinculado las señales del autómata de la estación para conducir el desempeño del motor a pasos.

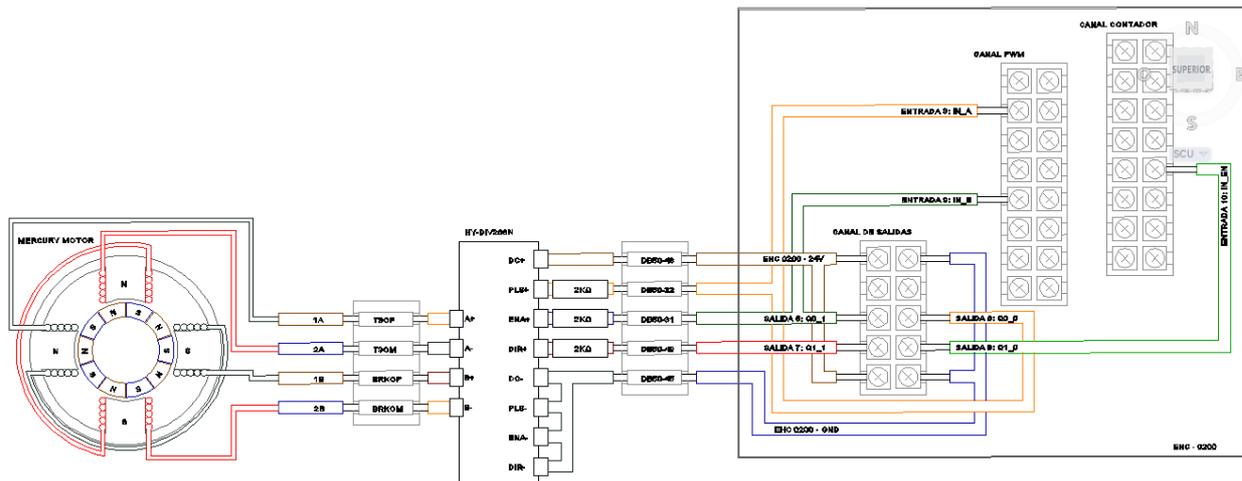


Figura 8-5. Diagrama de conexión del accionamiento de la articulación prismática.

Resultando que la nueva descripción de las señales involucradas en el automatismo que instruye la actuación de la articulación delta 3 sea la que se muestra en la tabla 8-6.

Tabla 8-6. Conexiones PLC – Controladora, Controladora – ROBOT, tercera articulación.

Lista de Conexiones del Automata al Robot						
DB25	IC-I	IC-O	DB50	Color	Símbolo	Descripción
DB25-21	IC2-4	IC2-13	DB50-34	Red	D3LS	Final de carrera del límite superior.
DB25-22	IC2-3	IC2-14	DB50-35	Red	D3LI	Final de carrera del límite inferior.
DB25-23	IC1-2	IC1-15	DB50-36	Org	D3H	Final de carrera del home.
DB25-9	IC4-7	IC4-10	DB50-7	Red	T3OP	Bobina 1A del motor a pasos
DB25-10	IC3-1	IC3-16	DB50-24	Gry	T3OM	Bobina 2A del motor a pasos
				Red	BRKOP	Bobina 1B del motor a pasos
				Blu	BRKOM	Bobina 2B del motor a pasos
Port3-2			DB50-17	Vlt	STEP	Tren de pasos
Port3-6			DB50-50	Blk	DIR	Dirección de giro
Port1-10			DB50-14	Vlt	EN	Habilitación del HY_DIV268N-5

De esta manera se finaliza la descripción electromecánica completa del manipulador, por lo tanto, basando la implementación en el apropiado conocimiento del hardware del robot, a partir de este punto se explica la solución del programa que se ha creado para la habilitación del SCARA.



## 9 PROGRAMA DEL AUTÓMATA

Previamente se ha comentado que el programa habría de ceñirse a la concepción de conseguir similitud entre los planteamientos del fabricante del manipulador, extraídos de la documentación de soporte, y basar la operabilidad en un entorno que resulte simple e intuitivo para los usuarios de la estación.

Como respuesta a la premisa anterior, en el controlador se ha alojado un programa que sigue una estructura basada en la estrategia de bloques derivados de función alojados en un único programa, que han sido creados para que estos logren completar de manera individual los diversos accionamientos que componen el robot y trabajando integrados en conjunto, completen las acciones desde el autómata, agreguen atributos que permitan al usuario observar la activación de sensores e instrumentos del robot y además puedan definirse a futuro para completar las tareas a las que se destine al SCARA ejecutar.

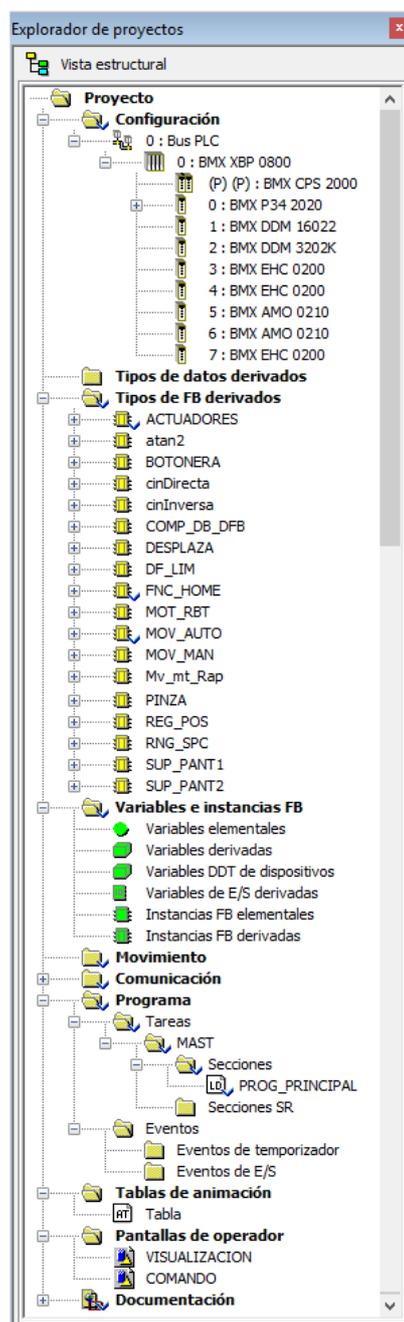


Figura 9-1. Árbol del proyecto creado para el Modicon M340.

El árbol del proyecto que se ha desplegado en la imagen anterior ilustra la estructura del programa. En primer lugar describe la configuración modular del Modicon M340, apreciándose en el tope del listado el modelo de bastidor y unidad de procesamiento, seguido inmediatamente por la tarjeta mixta de entradas/salidas digitales DDM 16022 para hacer interactuar el autómata con el puerto de entradas/salidas de la unidad controladora del *SCORBOT* que comparte ubicación en la célula con el SCARA, de la misma manera las señales binarias de la unidad de potencia del *SONY* se han enlazado con la tarjeta mixta DDM 3202K, los canales de contaje rápido de las dos tarjetas EHC 0200 se han empleado para medir por medio de los codificadores el desplazamiento angular de los eslabones y generar el tren de pulsos que ha de gobernar la marcha del motor a pasos. Finalmente se energizan los motores de corriente directa con el empleo de las dos tarjetas de salidas analógicas AMO 0210.

El árbol del proyecto también señala que han sido creados 12 bloques de función derivados para incluirlos en la programación del SCARA. Los bloques de función fueron considerados inicialmente para realizar la coordinación de los motores, al final del proyecto estos han derivado en el mando de gran parte de las acciones incluidas en el programa, concentrando su mayor aprovechamiento en las acciones de cálculo matemático inherente al desempeño del robot.

Para operar el SRX 4CH se ha creado una interfaz que es controlada por la función *SUP\_PANTI* (supervisión de pantallas), esta establece la lógica con la que los pulsadores y selectores dispuestos en la estación de mando rigen el comportamiento de la estación completa. Por ejemplo, desde la pantalla se pueden instruir movimientos de los eslabones y adoptar poses con el SCARA.

A partir de estas posiciones de brazo, puede concluirse que mover el SCARA es viable mediante la adopción sucesiva de configuraciones articulares, es necesario que el autómata cíclicamente estime la magnitud de desplazamiento de los eslabones, discrimine si es capaz de adoptarlas, luego les alcance y posteriormente logre mantenerse en ella hasta que el usuario le instruya una nueva ubicación desde la estación de trabajo.

Las operaciones descritas están incluidas en el bloque *MOV\_MAN*, creado para la realización de movimientos en modo manual de operación, tal y como lo describe la Guía GEMMA en el apartado 7.1 de este documento.

Esta sección se sustenta en otros bloques donde resaltan el cálculo de la cinemática inversa *cinInversa* y esta a su vez exigió la creación de la función *ATAN2* para poder obtener a partir de dos magnitudes de un plano, los ángulos de las articulaciones para alcanzar el posicionamiento acorde del efector. Tanto la función *cinInversa* como *ATAN2* han sido escritas en texto estructurado, por haber sido realizadas y simuladas en Matlab, para adecuarlas al Unity Pro posteriormente(ST).

*MOV\_MAN* recibe información proveniente del bloque *cinDirecta* que estima constantemente la cinemática directa y ambas comparten su resultado en la pantalla de explotación. Por su parte la cinemática inversa demanda más procesamiento, razón por la cual se ejecuta solo cuando la solución se prevé que pertenecerá al espacio de trabajo del robot. Bajo este concepto la realización de los movimientos obedece a la posibilidad de alcance de las posiciones demandadas para el efector. Para ello el bloque *DF\_LIM* (definición de límites), que emplea internamente a la función *RGN\_SPC* (región del espacio), habilita los desplazamientos si estos se pueden efectuar sin llegar a salirse del espacio de la tarea.

De ser permisible el desplazamiento el bloque de movimiento manual conduce a completar instrucciones de marcha, completadas con la actuación sobre el bloque de función *MOT\_RBT* (movimiento del robot), el cual coordina el movimiento del motor al que le sea asociada la función. Está realizada en escalera (LD) y es la más extensa de las funciones construidas, debido a que ha sido desarrollada de manera que pueda emplearse tanto para los motores de corriente directa, como al motor a pasos. Además, cuenta con la parametrización de los controladores de posición con el fin de garantizar el seguimiento de consignas, incluyendo también los factores de conversión de los encoders que los motores poseen acoplados. De esta manera el usuario con indicarle al bloque a cuál eslabón se vincula la función, son cargados estos parámetros sin necesidad de realizar modificación alguna.

De la misma manera se incluye un bloque para actuar sobre el efector del brazo, que se ha llamado *PINZA* y controla el accionamiento de la electroválvula que pudiere comandar al dispositivo neumático de sujeción de piezas.

Por su parte el bloque *FNC\_HOME*, de manera independiente gobierna el robot para configurar el punto de partida de todas las articulaciones, incide sobre las tarjetas de contaje rápido para habilitar su operación, por lo que se hace necesario iniciar las maniobras del manipulador con la puesta a HOME para conseguir operarlo.

Ya en el modo automático el robot se configura para reaccionar ante señales que se le envíen desde la célula de fabricación flexible y como respuesta a estos estímulos realizará movimientos en conjunto de todo el grupo de eslabones. Para ello el bloque de función *DESPLAZA* ha sido creado, para permitir el avance de una posición a otra que haya sido almacenada en la memoria del controlador mediante la función *REG\_POS* (registro de posiciones).

De la ilustración 9-1 también se extrae que existen un total de ocho (8) secciones básicas de programa que se emplean para hacer maniobras en el robot. Seis (6) de las cuales se han realizado en lenguaje escalera (LD) con la siguiente descripción según sus propósitos.

En primer lugar, la sección *ENCENDIDO* ha sido diseñada para incluir en ella todas las maniobras de puesta en marcha y operación del manipulador, la sección monitoriza los elementos de la unidad de potencia, los sistemas de seguridad e interactúa con los elementos de las pantallas de operador.

De la misma manera *REGISTRO\_POSICION* permite el guardado de posiciones definidas sobre el espacio de la tarea, además posibilita que sea realizada la revisión desde la pantalla de explotación de recursos del autómata, el registro que se ha realizado en el arreglo de memoria, en el momento en que el usuario precise editar alguno o varios de los elementos que lo componen.

Por su parte *MOTORES* se dedica únicamente al control de posición y velocidad del desplazamiento articular de los 4 enlaces del robot.

La sección de programa *M\_AUTO* concreta el uso del manipulador de manera autónoma, es de libre configuración y es específica a cada aplicación del robot. Por tal razón no pertenece al abanico de funciones que se incluyen en el árbol del proyecto como bloques de función derivados y que se refieren más adelante.

Las dos (2) secciones en lenguaje Grafset (SFC) han sido creadas para establecer secuencias de operación con el SRX 4CH y están vinculadas de manera directa a la sección *M\_AUTO*. *SECUENCIA* se ha desarrollado para recorrer de manera ordenada un total de 20 posiciones guardadas en un vector alojado en la memoria del autómata, específicamente se ha diseñado una rutina que sigue cuatro (4) aristas superficiales horizontales de un cubo de madera y al llegar a los vértices de la cara vista desde planta, desciende al centro de las aristas verticales de cada esquina.

*SECUENCIA* emplea la unidad de programación *SEC\_MOV\_AUTO*, con la intención de motorizar las articulaciones del robot a medida que el marcado del Grafset evoluciona.

Asimismo, la sección *PCK\_N\_PLC* emula un sistema de recogida, transporte y colocación de piezas (del inglés pick and place), con la inclusión de la electroválvula neumática descrita en la sección 6.8 de este trabajo como el accionador del elemento de agarre. Esta unidad de programación sustenta la motorización del manipulador que ha de desplazarse con la evolución del marcado de la sección *PICK\_MOV\_AUTO*.

Las últimas aplicaciones que se han desplegado en el árbol del proyecto constan de dos pantallas de operador de nombre *COMANDO* y *VISUALIZACION*, además de una tabla de animación denominada simplemente *tabla*. Estas se desarrollaron para servir de interfaz para que el usuario pueda además de visualizar las variables presentes en el programa del autómata, también puede accionarlas, modificarlas en magnitud y coordinar las actividades a ejecutar con el robot.

Finalmente, este documento en su sección de anexos incluye un manual descriptivo para la operación del robot. Sobre el mismo han sido incluidos todos los programas con una descripción expuesta a detalle y se incluye un modelo del empleo de cada función en la construcción de secuencias de trabajo del SCARA. Pero antes de llegar a ellos en la siguiente sección se mostrarán los aspectos de éxito alcanzados y los inconvenientes que se presentaron en la ejecución de este proyecto.



## 10 RESULTADOS

En esta sección se describen los aspectos más resaltantes en la realización de los diferentes planteamientos de automatización, los pasos recorridos con los cuales se ha llegado a la solución actual y que trabajos adicionales se desarrollaron para completar la tarea, que ha conducido a las condiciones en las que en la siguiente ilustración muestra la configuración final del hardware del robot SCARA SONY SRX 4CH. Ya habiendo sido instalados los elementos necesarios para completar los movimientos sobre sus eslabones coordinados desde la estación automática Modicon M340 que se ha descrito.

Sobre el brazo pocas modificaciones se precisaron, a razón de estar casi la totalidad de sus componentes en buenas condiciones, ameritando solamente conducir el cableado desde el autómata a la electroválvula para la actuación sobre el efector y de la misma manera dirigir los conductores dedicados a energizar el motor a pasos hasta la unidad de potencia.



Figura 10-1. Manipulador y Unidad de Potencia del SONY SRX 4CH.

Sobre esta última han sido llevadas a cabo la mayor parte de las tareas. Partiendo de la identificación de señales de fuerza y control que desde la unidad de control se envían y reciben hacia y desde el brazo.

En la imagen siguiente tomada de la controladora, se aprecia en el centro como ha sido fijada la interfaz descrita en la quinta sección de la memoria, que como se explicó en dicha sección, se encarga de la vinculación de señales de diferente naturaleza y distintos niveles de tensión.



Figura 10-2. Cableado del arreglo relés e interfaz PLC y Unidad de Potencia.

Mecánicamente la instalación de esta tarjeta electrónica también incluyó la sujeción de la manguera portacables que se acopla a la interfaz por medio de un conector DB-50, de manera que las fuerzas a las que están sometidas no incidieran sobre la integridad de la misma interfaz.

A la derecha de esta se ha dispuesto un riel para soportar las borneras de conexión de los conductores provenientes de los interruptores y los pulsadores de emergencia y se han fijado los relés de encendido y de emergencia de la unidad de potencia. De la misma manera del lado izquierdo se han colocado los relés para

actuar sobre los drivers de los servomotores y un grupo de borneras, que incluyen un fusible para actuar sobre el driver del motor a paso.

Estos drivers de potencia para el manejo de los motores se encuentran en el nivel inferior de la controladora, de esta manera los disipadores de calor drenan al chasis y además son refrigerados por medio de ventilación forzada, en la imagen 10-3 se puede observar la disposición.

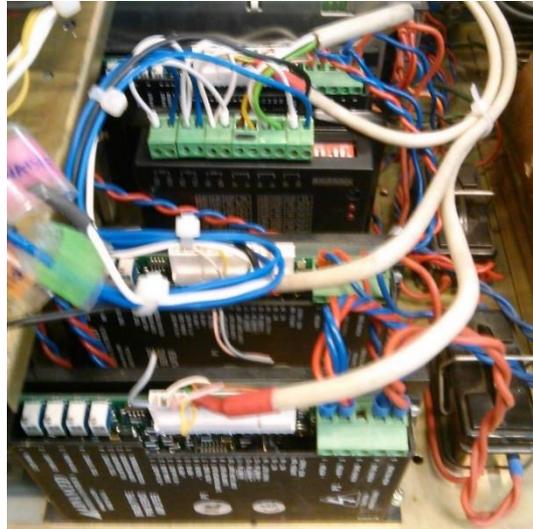


Figura 10-3. Cableado del arreglo de drivers para motores.

Finalmente, se ordenó el cableado con el uso de cintas de sujeción para evitar que por causa de vibraciones, por efecto del calor u otros factores se pudiera causar la liberación de los conductores de los puntos de conexión o en su deterioro, incurrir en fallas, inclusive en daños mas importantes. Además, se ha conseguido un acabado más estético de la unidad de potencia, quedando al final como se ilustra en la siguiente imagen.

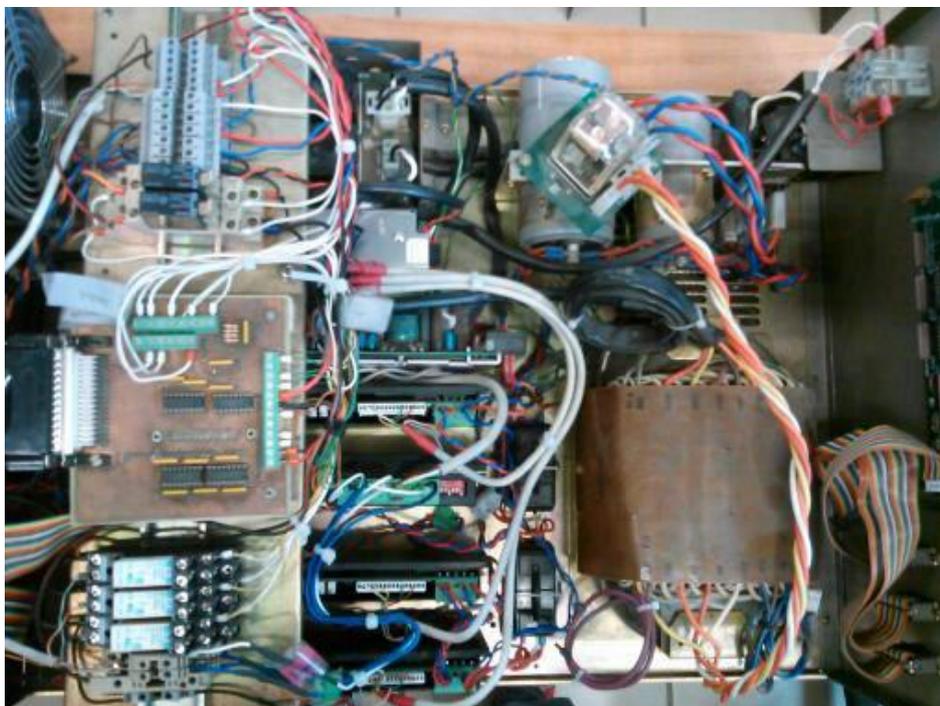


Figura 10-4. Cuadro eléctrico de la Unidad de Potencia.

Desde el punto de vista del software, se han ejecutado varios planteamientos que a medida que se han ido implementando, han evolucionado en las estrategias que se han completado finalmente en el automatismo. Inicialmente la intención estaba centrada en la utilización de los módulos de conteo rápido BMX EHC 0200 para controlar los motores de corriente directa y el motor a pasos.

Por ello se consideró destinar las salidas rápidas de los módulos para generar mediante la modulación de ancho de pulsos una señal analógica que guiara a los motores de manera eficiente.

Unity Pro posee la función ya desarrollada en un bloque de nombre PWM.

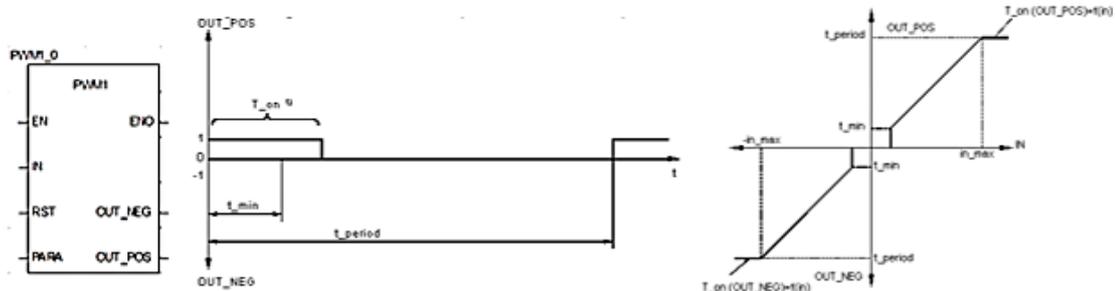


Figura 10-5. Bloque de Función PWM de Unity Pro.

Este bloque consigue que a partir de una señal de tensión de 24 voltios de corriente directa una salida digital sea capaz de emitir una señal analógica. Con el ajuste de tres parámetros como se aprecia en los esquemas incluidos en la figura anterior. El período total de la señal cuadrada, el tiempo mínimo que marca el tiempo de servicio (duty cycle) de la onda. Y el valor máximo de la entrada, 24 voltios.

Al medir con el voltímetro y modificar la duración del tiempo mínimo, se regula la magnitud de la señal de tensión desde 0 voltios a 24 voltios. Sin embargo, se obtuvieron algunos inconvenientes que llevo a desistir de este plan inicial.

En primer lugar, las salidas de las tarjetas de conteo en ausencia de consigna presentan una tensión por encima de 0 voltios, 0.79 voltios han sido medidos, producto de una pequeña corriente de fuga que en la tabla de especificaciones del módulo el fabricante refleja. Dicha tensión resulta suficiente para inducir una corriente en el rotor de los servos y mover el eslabón. Esta desviación se pudo corregir regulando los potenciómetros de ganancia de los drivers llevándolos a disminuir la sensibilidad a la tensión de referencia que se aplicaba, solución que va en detrimento de la calidad de respuesta del sistema.

Seguidamente se presentó la dificultad de cambiar el sentido de la corriente de polarización del motor. La señal que entregaba el módulo resultaba siempre positiva, por lo que invertir el sentido de giro de motor se planteaba con el uso de relés externos. Esto además del costo económico de los elementos, incluía un costo en señales digitales para actuar desde el autómatas sobre los relés, multiplicado por el número de motores del brazo.

Por otro lado, la magnitud de la señal tenía cambios abruptos. De 0 voltios se conseguía incrementar a 1.5, luego a 3, a 5.7, 8 y 9.9. La referencia que los drivers admiten está dentro del rango de  $\pm 10$  voltios máximos, y las velocidades de los motores en el eslabón son altas ante estos cambios de referencia que no se consideraban admisibles a razón de ser un riesgo potencial.

Finalmente, el fenómeno observado que condujo a desechar el planteamiento se produjo sobre la calidad del movimiento del eslabón. La naturaleza de la señal analógica con la que se gobierna el driver parte de una señal discreta. Interiormente el driver emplea también modulación por ancho de pulsos (PWM) para regir el motor. Entonces el motor marchaba con un movimiento percutado, indicando que la corriente le llegaba a ráfagas y no con un trazo suave constante.

La solución derivó en la instalación de tarjetas de salidas analógicas BMX AMO 0210 que permiten conseguir 0 voltios absolutos de tensión, nivel al cual el motor se detiene totalmente. ofrecen una señal de -10 a 10 voltios de corriente continua, con lo cual puede definirse sin la necesidad de elementos adicionales, el sentido de marcha del servo. Además, la señal de salida se puede regular en magnitud con una precisión de milésimas de voltios, que en esta aplicación no resulta ciertamente necesario, pero posibilita crear consignas que suavizan el comportamiento del avance del motor.

A partir de esta modificación se consideró que el hardware contaba con las prestaciones esperadas de desempeño, se completaron las actividades de limpieza y orden de los demás elementos, se colocaron las protecciones del cuadro de control, se colocó en el lugar de destino y se reorganizaron los conductores en la estación de manipuladores de la célula de fabricación flexible.

Existe una gran cantidad de modificaciones que pueden incluirse a medida que la utilización del SCARA se

impulse ameritando más habilidades, por ejemplo, en el siguiente apartado se listan algunas de ellas.

## 10.1 MEJORAS A IMPLEMENTAR

Una función a incluir y que resulta sencilla de implementar es conseguir la *DEFINICIÓN DE UNA POSICIÓN REFERENCIAL*, que al ya haber creado una posición o a partir de la posición cero (0), conduzca a referenciar en el espacio de trabajo del robot nuevas posiciones creadas a partir de la modificación de alguna de las cuatro magnitudes que componen el arreglo matricial que ha sido alojado en memoria y que refiere a una de las posiciones creadas para el robot.

Componer una función que completa para la generación de trayectorias más complejas, líneas precisas y movimientos circulares definidas a partir de la inteligencia del manipulador. Y así crear tantas herramientas como se hagan necesarias para promover el performance de la unidad y por ende de las aplicaciones a las que se destine la estación de robots.

# 11 CONCLUSIONES

---

Este proyecto ha sido llevado a cabo en un entorno real que demanda de una gran eficiencia de desempeño. Ha permitido encontrar diversas estrategias de trabajo con el autómata programable, debido a la variedad de subprocesos que trabajan en conjunto dentro del manipulador. Además, ha sido exigida la adecuación de diferentes consignas de operación, con lo cual, una buena parte de las situaciones que se pueden presentar en un entorno de mayor escala se encuentran presentes en este sistema.

Es en estas valencias donde el proyecto centra su alto valor educacional y práctico. Haberlo concluido permite que fruto del aprendizaje obtenido, se haga pertinente plantear algunas recomendaciones que resulten de provecho al momento de utilizar el manipulador o que de manera general, sean observadas en las tareas de programación de procesos automáticos, con la utilización del software Unity Pro en entornos reales donde es preciso tomar previsiones para no incurrir en errores, en funcionamiento anómalo, en posible deterioro de los equipos e inclusive impulse la prevención de accidentes. Considerando estas razones suficientes para listar algunas de ellas a continuación:

Programar en primer lugar un lazo donde la seta de emergencia sea habilitada para detener todos los procesos donde la utilización de energía tanto cinética como potencial sea empleada. Movimiento de motores, alimentación de tensión, descenso o incremento de presión, elevación o disminución de la temperatura, entre otras posibles manifestaciones de la energía en los sistemas a controlar.

Incluir en la medida de lo posible instrumentos para la vigilancia de estas magnitudes, con la intención de crear lazos de control y de atención de condiciones seguras desde el inicio de las actividades de automatización de los procesos.

Observar el nombre exacto de las variables con el fin de realizar su correcto direccionamiento, conseguir que las señales coincidan con las acciones del autómata y su conexión con los elementos de la planta, no repetir equívocamente los accionamientos desde diferentes ubicaciones del programa, que en lenguajes de secciones múltiples como Unity Pro, la probabilidad de que esta falla ocurra se incrementa.

Definir correctamente el tipo de variable que se maneja, BOOL, INT, REAL, BYTES, entre otros.

Comprender las limitaciones del entorno de programación y de igual manera explotar al máximo las valencias que este tenga.

Simplificar en cuanto sea posible las soluciones propuestas, siempre que satisfagan el planteamiento y las condiciones seguras. Entendiendo que existen casos en donde con menos complejidad se logra más rapidez de respuesta en la solución de la necesidad original.

Elegir el entorno de programación más amigable y sencillo de implementar, que sea de la comprensión de usuarios, personal técnico y cuantos pudieren en determinado momento hacer empleo de la plataforma que se diseña para automatizar el sistema propuesto a control.

Conocer los pasos de la creación de proyectos de automatización y su correcta documentación técnica.

Incluir todas las posibles acciones que tanto de manera correcta como errónea pudiera incurrirse en la operación del proceso, en pro de evitar eventos no deseados y de riesgo.

Estas previsiones aunadas al conocimiento del manejo de automatismos y control clásico industrial derivarán en la consecución de soluciones eficientes. En el caso del SCARA se ha conseguido crear un grupo de funciones capaces de la realización de maniobras de manipulación de objetos en la Célula de Fabricación Flexible, obteniendo a partir de ellas posicionamiento efectivo del espacio de trabajo del robot, el seguimiento de un perfil en la rapidez de desplazamiento que es acorde a la demanda de tiempo de la planta y la capacidad de crear tareas específicas de trabajo donde el manipulador es el centro de la funcionalidad del sistema, desde una estación de trabajo de la red de control.



# 12 ANEXOS

Este apartado expone cómo se han desarrollado las soluciones a los planteamientos propuestos y descritos en la sexta sección del proyecto, incluye la metodología operacional a seguir y obedece al mismo orden establecido en el programa y en la sección que se ha hecho referencia anteriormente. Del mismo modo sugiere emplearle como un manual técnico de operación, al cual consultar al realizar operaciones sobre el manipulador.

## 12.1. ANEXO 1: PROGRAMA PRINCIPAL, CONTROL DE PANTALLAS DE OPERADOR

El programa diseñado para el control del SCARA ha sido desarrollado en lenguaje escalera (LD), se sustenta en aprovechar la utilidad del software Unity Pro, que permite la creación de bloques derivados de función. Esta ventaja ha permitido que se ejecuten todas las acciones de control del manipulador, con el uso de una única sección de programa (POU) denominada *PROGRAMA PRINCIPAL*, que se ejecuta en la tarea principal [*MAST*].

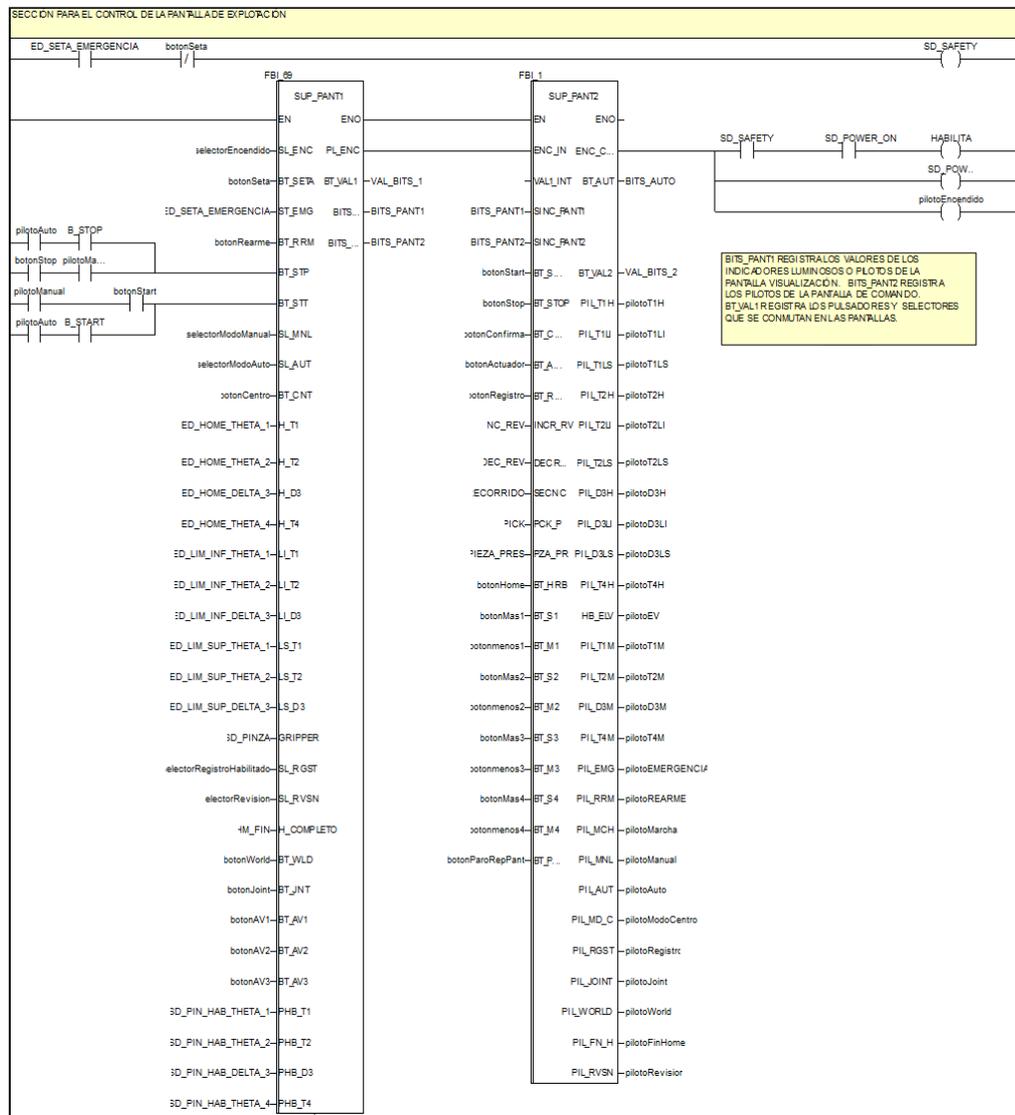


Figura 12-1 . Programa principal. Control de Pantallas de Operador.

Siguiendo este planteamiento, se han creado las funciones *SUP\_PANT1* y *SUP\_PANT2*, que se ilustran en la figura anterior. Estos en conjunto energizan, coordinan y protegen la unidad de potencia del SONY SRX 4CH.

### 12.1.1 ANEXO 1.1: CONTROL DE LA PANTALLA DE OPERADOR, SUP\_PANT1

El bloque de la izquierda de la figura 12-1, llamado *SUP\_PANT1*, se creó para recibir las variables digitales que se generan en la pantalla de operador y siguiendo una combinación de condiciones, como se muestra en la siguiente figura, conmuta entre la realización de operaciones en modo manual y/o la marcha automática.

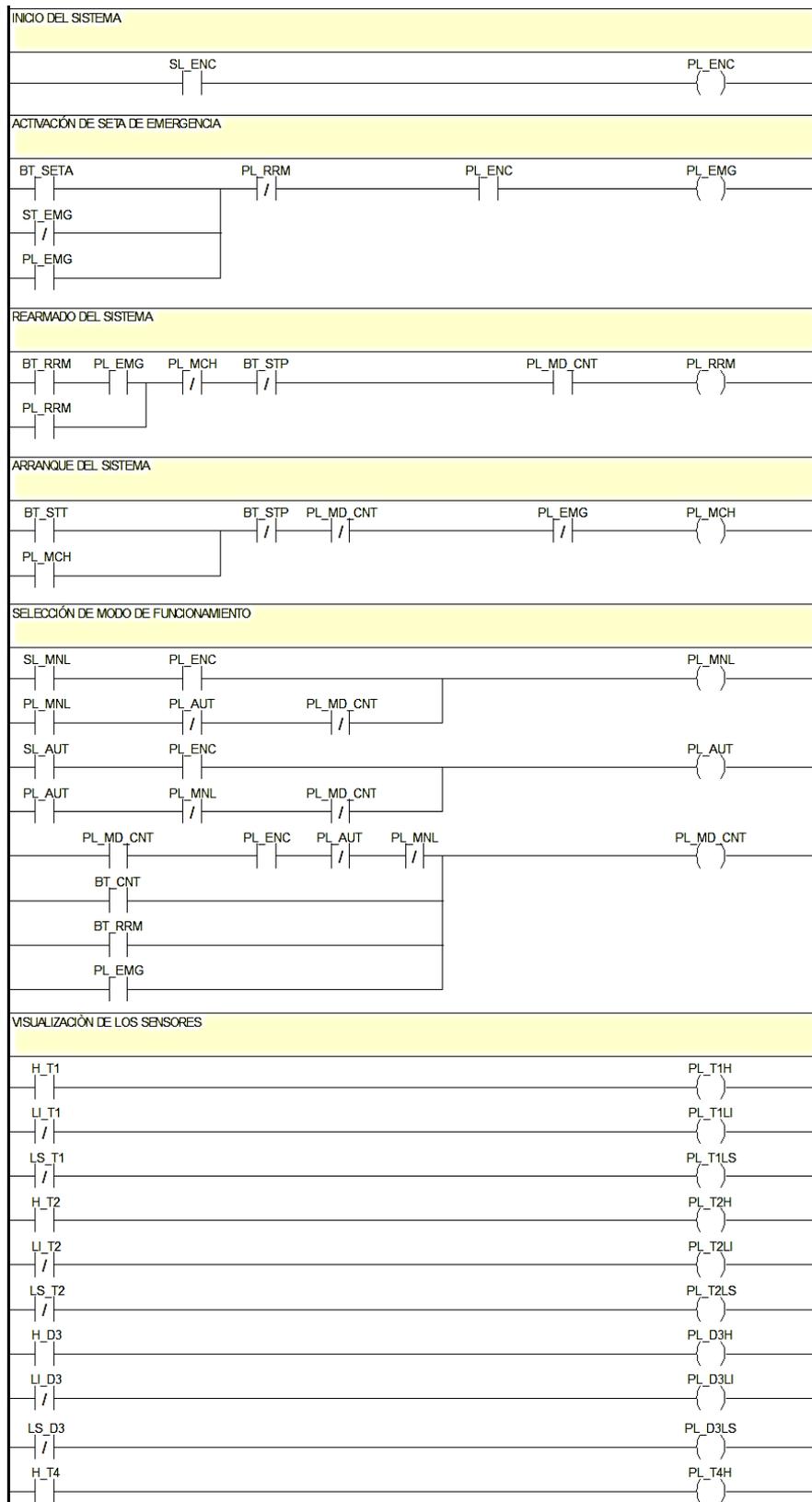


Figura 12-2. Bloque SUP\_PANT1. Sección de programa CONT\_PANTALLA1.

Es importante señalar que los bloques de función tienen, por configuración del software, una cantidad máxima de treinta y dos (32) entradas y el mismo número de salidas, las que pueden ser de diversa naturaleza, enteras, reales, digitales, entre otras. Al querer enlazar el gran número de señales que están disponibles para que el operador realice maniobras con el robot y de manera recíproca, visualice las que el entorno utiliza durante su evolución, se ha encontrado que estas superan en cantidad la restricción mencionada, y por ello ha sido preciso enrutar estas señales a través de funciones de programa.

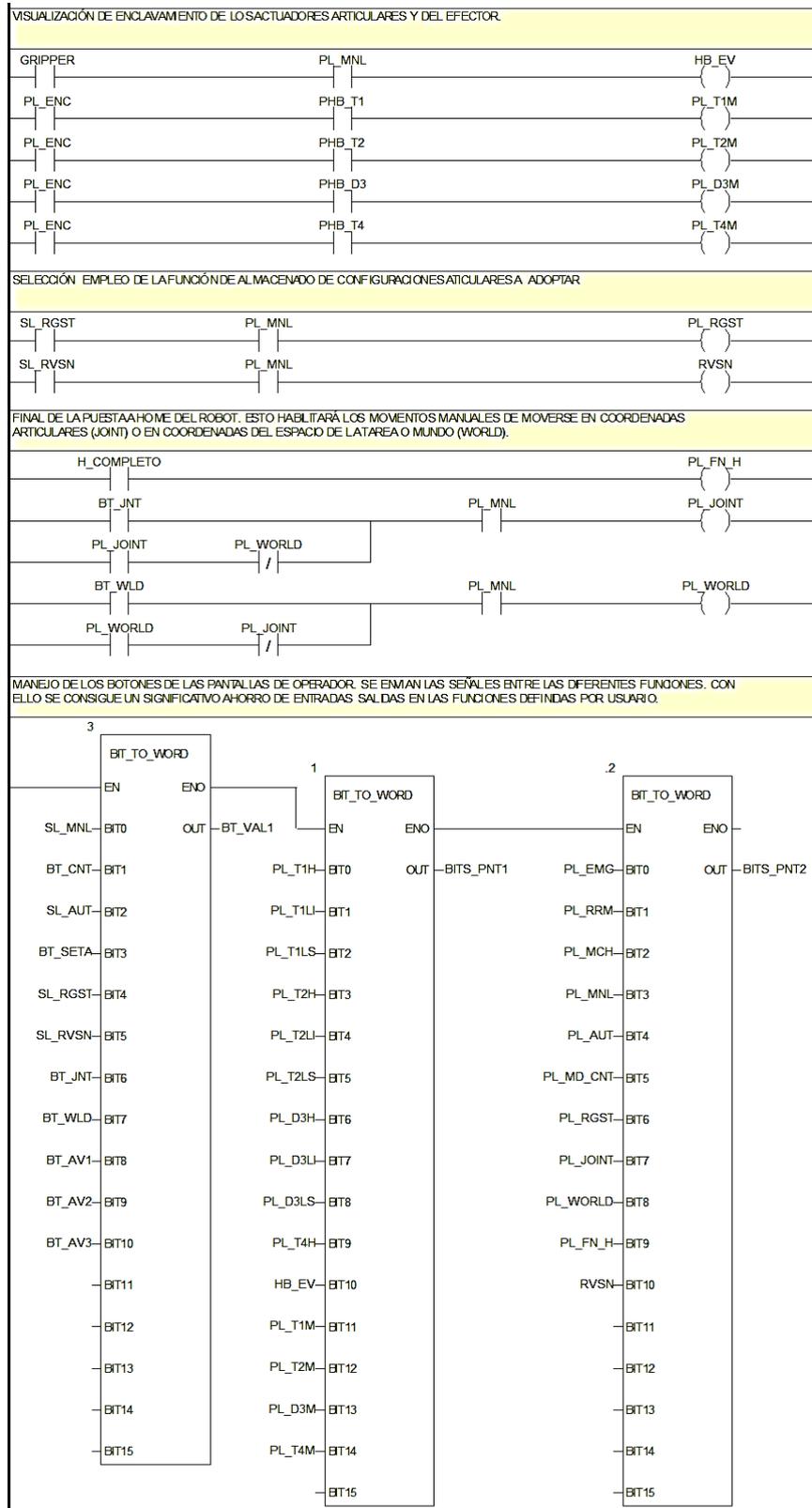


Figura 12-3 . Bloque SUP\_PANT1. Sección de programa CONT\_PANTALLA1, segunda parte.

En tal sentido, se han utilizado variables del tipo BYTE y WORD, logrando a través de cada una de ellas transmitir a otras instancias del programa, por medio de una sola entrada y/o salida del bloque de función, hasta ocho (8) o dieciséis (16) estados de variables distintas que se modifican dentro de la función. Es así, tal como se aprecia en la figura 12-3 y en la tabla 12-1, que se ha conseguido una significativa reducción del número de señales a vincular a las funciones del programa y se optimiza a la vez la interacción entre todas ellas.

Tabla 12-1. Lista de variables del bloque de función SUP\_PANT1.

<b>Lista de Variables del Bloque Derivado de Función SUP_PANT1</b>							
<b>ENTRADAS</b>			<b>SALIDAS</b>			<b>PRIVADO</b>	
<b>#</b>	<b>Nombre</b>	<b>Tipo</b>	<b>#</b>	<b>Nombre</b>	<b>Tipo</b>	<b>Nombre</b>	<b>Tipo</b>
1	SL_ENC	BOOL	1	PL_ENC	BOOL	INTERM1	INT
2	BT_SETA	BOOL	2	BT_VAL1	WORD	HB_EV	BOOL
3	ST_EMG	BOOL	3	BITS_PNT1	WORD	PL_T1H	BOOL
4	BT_RRM	BOOL	4	BITS_PNT2	WORD	PL_T2H	BOOL
5	BT_STP	BOOL				PL_D3H	BOOL
6	BT_STT	BOOL				PL_T4H	BOOL
7	SL_MNL	BOOL				PL_T1LI	BOOL
8	SL_AUT	BOOL				PL_T2LI	BOOL
9	BT_CNT	BOOL				PL_D3LI	BOOL
10	H_T1	BOOL				PL_T1LS	BOOL
11	H_T2	BOOL				PL_T2LS	BOOL
12	H_D3	BOOL				PL_D3LS	BOOL
13	H_T4	BOOL				PL_T1M	BOOL
14	LI_T1	BOOL				PL_T2M	BOOL
15	LI_T2	BOOL				PL_D3M	BOOL
16	LI_D3	BOOL				PL_T4M	BOOL
17	LS_T1	BOOL				PL_EMG	BOOL
18	LS_T2	BOOL				PL_RRM	BOOL
19	LS_D3	BOOL				PL_MCH	BOOL
20	GRIPPER	BOOL				PL_MNL	BOOL
21	SL_RGST	BOOL				PL_AUT	BOOL
22	SL_RVSN	BOOL				PL_MD_CNT	BOOL
23	H_COMPLETO	BOOL				PL_JOINT	BOOL
24	BT_WLD	BOOL				PL_WORLD	BOOL
25	BT_JNT	BOOL				PL_FN_H	BOOL
26	BT_AV1	BOOL				RVSN	BOOL
27	BT_AV2	BOOL					
28	BT_AV3	BOOL					
29	PHB_T1	BOOL					
30	PHB_T2	BOOL					
31	PHB_D3	BOOL					
32	PHB_T4	BOOL					

### 12.1.2 ANEXO 1.2: CONTROL DE LA PANTALLA DE OPERADOR, SUP\_PANT2

Analogamente se ha creado el bloque de la derecha de la figura 12-1, llamado *SUP\_PANT2*, este recibe un número importante de las variables digitales que se visualizan en la pantalla de operador y las señales procedentes del manipulador que monitoriza el autómata.

Asimismo, recibe los bits que se han producido en la función *SUP\_PANT1* y además se encarga de escribir en las pantallas de explotación los cambios de estado que se han detectado en la ejecución del programa. La siguiente figura muestra el programa *CONT\_PANTALLA2* que reside en el bloque *SUP\_PANT2*.

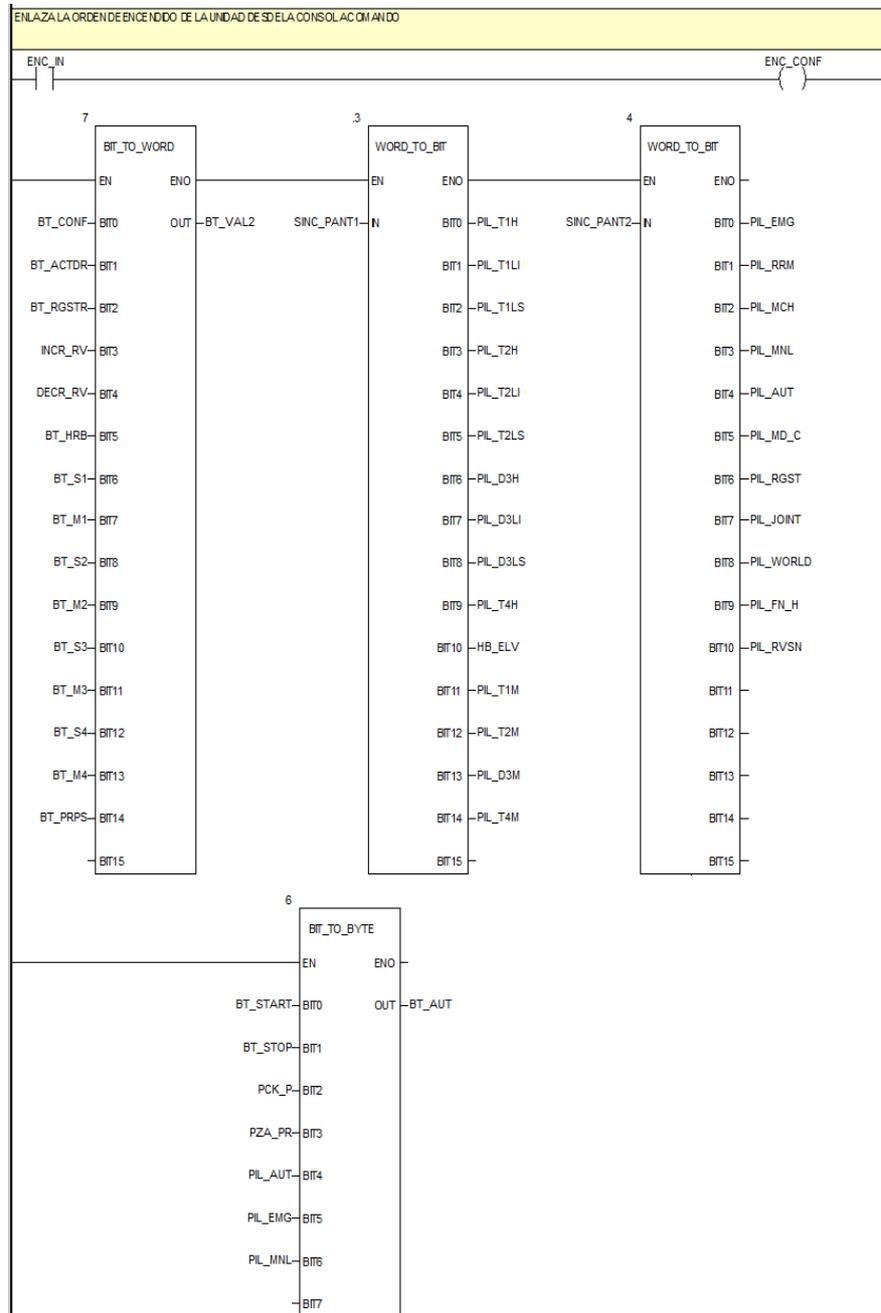


Figura 12-4 . Bloque SUP\_PANT2. Sección de programa CONT\_PANTALLA2.

Resumiendo, los bloques de supervisión de las pantallas de explotación de Unity Pro, *SUP\_PANT1* y *SUP\_PANT2*, cuya lista de variables de este último se agrupa en la tabla 12-2, se encargan de recibir y energizar los diferentes elementos mecatrónicos que conforman el conjunto unidad de potencia, robot manipulador y controlador del SONY SRX 4CH. Ambas funciones han sido construidas en lenguaje escalera y se encuentran vinculadas a las pantallas de operador *VISUALIZACION* y *COMANDO*, que se muestran en la ilustración 12-5.

Tabla 12-2. Lista de variables del bloque de función SUP\_PANT2.

Lista de Variables del Bloque Derivado de Función SUP_PANT2							
ENTRADAS			SALIDAS			PRIVADO	
#	Nombre	Tipo	#	Nombre	Tipo	Nombre	Tipo
1	ENC_IN	BOOL	1	ENC_CONF	BOOL	BT_VAL3	WORD
2	VALI_INT	INT	2	BT_AUT	BYTE	BT_REARME	BOOL
3	SINC_PANT1	WORD	5	BT_VAL2	WORD		
4	SINC_PANT2	WORD	6	PIL_T1H	BOOL		
5	BT_START	BOOL	7	PIL_T1LI	BOOL		
6	BT_STOP	BOOL	8	PIL_T1LS	BOOL		
7	BT_CONF	BOOL	9	PIL_T2H	BOOL		
8	BT_ACTDR	BOOL	10	PIL_T2LI	BOOL		
9	BT_RGSTR	BOOL	11	PIL_T2LS	BOOL		
10	INCR_RV	BOOL	12	PIL_D3H	BOOL		
11	DECR_RV	BOOL	13	PIL_D3LI	BOOL		
12	PCK_P	BOOL	14	PIL_D3LS	BOOL		
13	PZA_PR	BOOL	15	PIL_T4H	BOOL		
14	BT_HRB	BOOL	16	HB_ELV	BOOL		
15	BT_S1	BOOL	17	PIL_T1M	BOOL		
16	BT_M1	BOOL	18	PIL_T2M	BOOL		
17	BT_S2	BOOL	19	PIL_D3M	BOOL		
18	BT_M2	BOOL	20	PIL_T4M	BOOL		
19	BT_S3	BOOL	21	PIL_EMG	BOOL		
20	BT_M3	BOOL	22	PIL_RRM	BOOL		
21	BT_S4	BOOL	23	PIL_MCH	BOOL		
22	BT_M4	BOOL	24	PIL_MNL	BOOL		
23	BT_PRPS	BOOL	25	PIL_AUT	BOOL		
			26	PIL_MD_C	BOOL		
			27	PIL_RGST	BOOL		
			28	PIL_JOINT	BOOL		
			29	PIL_WORLD	BOOL		
			30	PIL_FN_H	BOOL		
			31	PIL_RVSN	BOOL		

En su mayoría, las variables residen dentro de ambos bloques de función, presentes en POU's de nombre *CONT\_PANTALLA1* y *CONT\_PANTALLA2*. Estas unidades de programación están diseñadas para que se cumplan las especificaciones de la Guía *GEMMA*, que fue descrita en el apartado 7.1 de este documento. Por este motivo se aprecia en la figura 12-2, como los modos de operación son enclavados en concordancia con la interacción que se ha establecido en el estándar ilustrado en la figura 7-2 dispuesta en el mismo apartado.

Además, en la figura se observa, a partir de la vigésima tercera (23) línea, como el autómatas interpreta las señales todo o nada de los sensores y las muestra en la interfaz *VISUALIZACION*, Por ejemplo, a través de la décima (10) entrada H\_T1 de la función *SUP\_PANT1* se detecta a la variable *ED\_HOME\_THETA\_1*, que el PLC recibe a través de la tarjeta de entradas digitales desde el sensor de *HOME* de la primera articulación del SCARA. Esta actúa directamente sobre la variable privada *PL\_T1H* de la función y esta a su vez modifica el estado de la variable piloto *T1H* en la pantalla de explotación, como resultado de haber transmitido *PL\_T1H*, a la salida número seis (6) del bloque *SUP\_PANT2* y en las demás ubicaciones del programa principal en donde sea utilizada. El resto de los sensores funciona de manera análoga.

Asimismo, los actuadores del SRX 4CH se señalan en la interfaz al encender o estar detenidos, como resultado

de habilitar la salida del motor de la articulación SD\_PIN\_HAB\_THETA\_1. En la línea 34 se conmuta el permiso de la vigésima novena entrada (29) PHB\_T1 de la función SUP\_PANTI y se acciona la decimo séptima (17) salida PL\_T1M de SUP\_PANT2, la cual incide sobre la variable pilotoT1M en la interfaz y en la totalidad del programa principal.

El resto de los actuadores siguen la misma estrategia, la que es posible de apreciar en la figura 12-5 y 12\_6, que se ofrece a continuación.

### 12.1.3 ANEXO 1.3: PANTALLAS DE OPERADOR, VISUALIZACIÓN y COMANDO

A la izquierda de la figura 12-5 se observa como los diferentes elementos dispuestos en el brazo cambian de estado, a medida que son accionados por el desplazamiento de los eslabones o cuando el controlador emite una consigna para que estos actúen.

La pantalla VISUALIZACION también incluye los trazos de la respuesta de los controladores de posición, tanto en seguimiento de consignas como en la regulación de desviaciones de cada uno de los motores de las articulaciones del SCARA.

A la derecha se tiene una interfaz que contiene un mayor número de elementos, tanto de manipulación como de observación de las condiciones de las diferentes variables que se utilizan dentro del autómata. El usuario le empleará para múltiples tareas en la operación del sistema, siendo esta la columna vertebral del empleo del SONY, en la estación que ocupa en la célula de fabricación flexible.

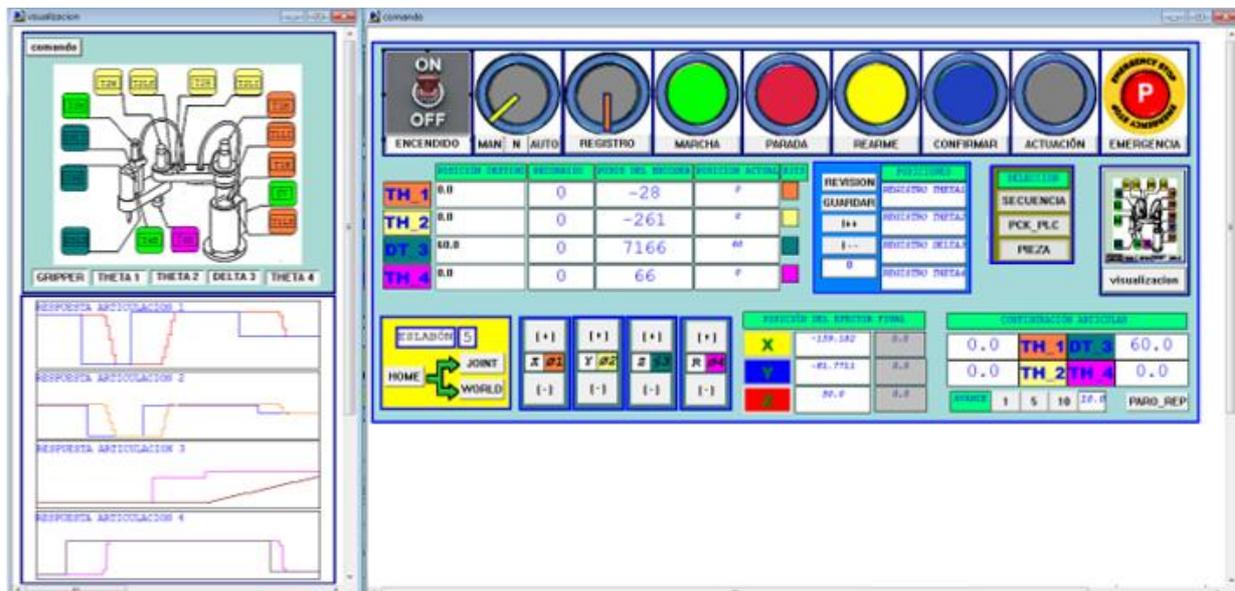


Figura 12-5. Pantallas de operador VISUALIZACIÓN y COMANDO.

Mas adelante se describe cada elemento de ambas pantallas, para conseguir familiarizarse con los distintos renglones que las conforman, y lograr aprovechar al máximo las potencialidades que en ella se han incluido y que superan a la consola de programación (teaching pendant) original del dispositivo.

En la siguiente figura se aprecia como la sección de programa hace la llamada a la función de control de la pantalla de operador.

Sobre ella los bloques de función activan los pilotos en las pantallas de explotación, entendiéndose que cada imagen desplegada en la interfaz se ha asociado a una variable presente en la programación del autómata. Cuando cambia de estado o magnitud se modifica su presentación como si se tratase de una lámpara o luz piloto, como las dispuestas usualmente en los cuadros o armarios eléctricos de potencia, distribución y control.

En la ilustración se evidencia este comportamiento, la pantalla de operador VISUALIZACIÓN, maneja quince (15) indicadores, agrupados de acuerdo a la articulación a la cual pertenecen y señalan los finales de carrera superior e inferior y la detección del punto de inicio. Cabe destacar que a cada articulación se le asignó un color para identificarla en toda la interfaz y se tornarán verdes cuando detecten la presencia de la leva que los acciona o se energice el actuador que la imagen refiere (motores o electroválvula).

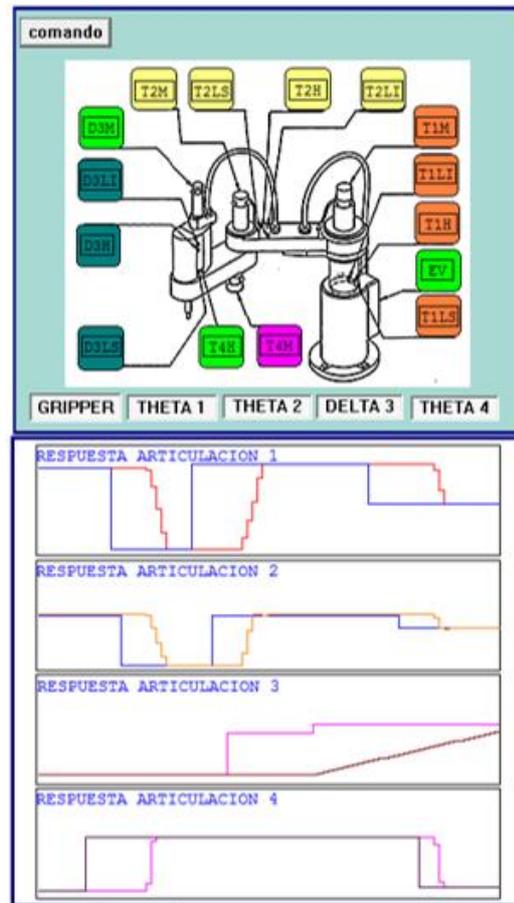


Figura 12-6. Pantalla de operador VISUALIZACIÓN.

De la misma manera la pantalla de explotación contiene cinco (5) interruptores en la parte inferior de la figura principal, que se conmutan para habilitar los actuadores y los mismos puedan ser accionados de manera segura y bajo el consentimiento del operador.

En el nivel inferior de la interfaz, han sido incluidos gráficos de tendencias, para hacer visibles las respuestas de las articulaciones ante las consignas de posición. Se nota que el bloque PID de la librería de Unity Pro permite controlar adecuadamente los motores, gracias a varias estrategias incluidas como el supresor del efecto windup que se ha habilitado. Es preciso obviar el retardo que resulta de la modificación de la referencia, cuyo valor se ingresa por pantalla y posteriormente se confirma con un pulsador para dar inicio a la actuación.

El resultado de esta sección es la activación de señales alojadas en la memoria del autómat, como la referencia de posición y la orden de movimiento, para que se pueden realizar por parte del usuario desde el ordenador con la utilización de pulsadores, interruptores y cuadros de modificación de variables dispuestos en la estación de trabajo.

Las demás líneas de programa de las funciones son dedicadas a la operación del manipulador mediante el uso de la pantalla de explotación de recursos nombrada *COMANDO*.

La interfaz *COMANDO* surge de la necesidad de iniciar el proceso de pruebas en el robot, sin embargo, se convirtió en un recurso valioso para el uso del SCARA y satisfizo la necesidad de realizar maniobras en el autómat, ante la inexistencia de elementos en la unidad de control que pudieren destinarse para este propósito.

Esa es la razón de tener en la parte superior de la interfaz un total de nueve (9) elementos de maniobra principales, un interruptor para el encendido de la unidad de potencia del robot, un (1) selector de tres posiciones para elegir el modo de operación manual o automático del manipulador, un (1) segundo selector, en este caso de dos posiciones, para habilitar el registro de las configuraciones articulares en la composición de trayectorias.

De la misma manera se cuenta con cinco (5) pulsadores para realizar las acciones de arranque de marcha del proceso, la parada controlada, el rearmado del sistema después de una situación atípica que haya sido motivo de aplicar la parada de emergencia, la confirmación de consignas y la puesta en ejecución de esas consignas.

Completa el conjunto de elementos de la línea superior la inclusión de un pulsador tipo seta, este está asociado al sistema de emergencia de la unidad de potencia del SCARA, que inhibe la energización de la unidad al darse su accionamiento, dicho sistema de emergencia posee otros dos pulsadores tipo seta de emergencia físicos, ubicados en la unidad de potencia y sobre la célula de fabricación flexible respectivamente.

En este punto es importante señalar, que la energización de los módulos del Modicon M340 proviene de la unidad de potencia, en otras palabras, la unidad de potencia brinda a las tarjetas 24 voltios de corriente continua para que estas operen, por lo cual después de iniciar el sistema y concluir la puesta punto inicial, no es recomendable conmutar el interruptor de ENCENDIDO a su estado apagado y volver a energizar la unidad, conmutándolo a *ON* nuevamente, porque el ajuste cambia de origen y referenciaría un nuevo espacio de trabajo, con lo cual se puede causar daños a la unidad robótica.

La recomendación es realizar el reinicio de los estados de los elementos de maniobra con la puesta del selector de tres posiciones en el botón *N*, así se rearmen los modos operativos, se detienen las actuaciones del brazo y son conservados los valores referenciales de la cadena articular.

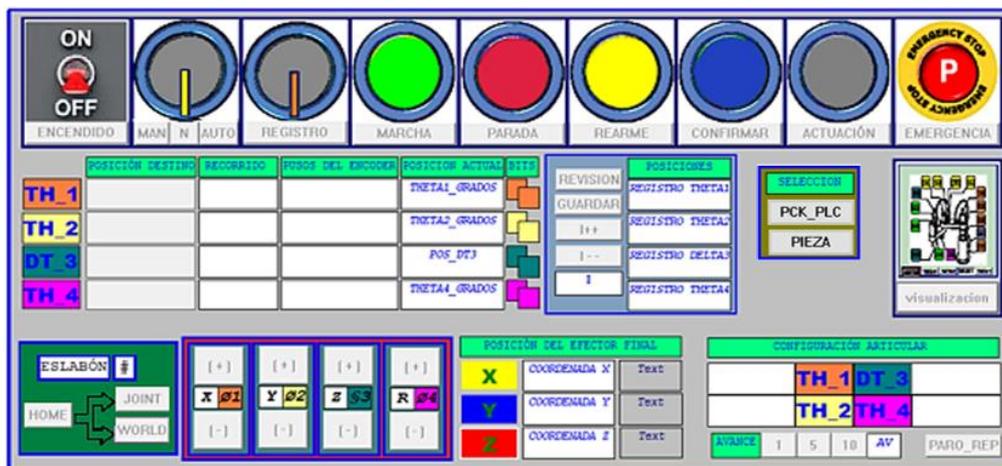


Figura 12-7. Pantalla de operador COMANDO.

De la figura 12-7 se conoce también, que en la región central de la pantalla de operador son ofrecidas las visualizaciones de las magnitudes de posición, estas se encuentran listadas de manera de distinguir el destino, la distancia angular que se ha de recorrer para alcanzarlo y la ubicación actual expresada tanto en pulsos del encoder como en grados, que han sido elegidos como la unidad de ingeniería correspondiente para representar esta magnitud en las articulaciones rotacionales y milímetros en la prismática.

A la derecha de las referencias de posición son encontrados unos pilotos que indican desde el momento de la puesta a HOME en el que son encendidos, cuando son habilitadas las tarjetas de conteo rápido (con el *BIT FORCE ENABLE*) y otros que se encienden cuando se ajustan a cero los valores de conteo (con el *BIT FORCE RESET*).

Siguiendo el recorrido se encuentra el renglón de registro de posiciones a emplear en las secuencias automáticas configurables según las necesidades de aplicación. Este renglón se habilita desde el selector dos posiciones *REGISTRO* de la línea superior. Dentro de la celda se encuentra el botón *REVISION* y *GUARDAR* usados para registrar la posición actual del efector en las variables articulares, con el propósito de referirlos al momento de la configuración de la secuencia de movimientos del robot en el modo de operación automático.

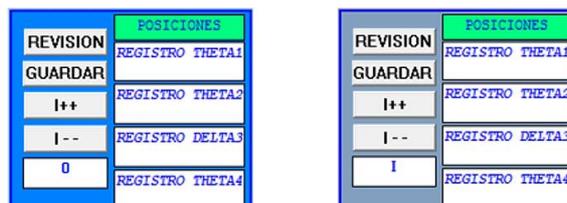


Figura 12-8. Renglón de la pantalla COMANDO para el registro de posiciones.

El conjunto de botones se emplea también para revisar y editar las magnitudes que se guardan en el arreglo matricial. Ambos modos de funcionamiento operables desde la misma sección de pantalla se indican con el

cambio de color del fondo del renglón que señala la figura anterior, pudiendo el usuario de esta manera identificar la tarea que realiza con simplicidad, estableciendo una interacción entre la función de supervisión de pantalla y la función de almacenamiento de datos desde la misma interfaz de operador.



Figura 12-9. Renglón de la pantalla COMANDO para habilitar el método de trabajo.

A la derecha de la región de registro, se encuentra un renglón dedicado a habilitar el tipo de trabajo que se realizará al operar el robot de modo autónomo, ilustrado en la figura 12-9. Para el proyecto se ha creado una secuencia o tarea de recogida y posterior colocación de piezas. Esta se señala desde este cuadro de la pantalla y al igual como en estaciones referidas anteriormente ofrece una herramienta visual para ver de manera sencilla cuando está en marcha.

La región central de la pantalla de operación finaliza con la inclusión de un botón de navegación entre ambas interfaces de navegación.

Al fijar la atención sobre la franja inferior de la estación, se ubica a la izquierda un pabellón dedicado a la realización de la puesta a home de las articulaciones del robot, y cuando esta función se completa se habilitan los movimientos manuales, o bien en coordenadas articulares o en el plano cartesiano, que se ha definido con la utilización de la regla de la mano derecha referenciado desde la base del manipulador. Señalando el eje Z positivo con el pulgar, el índice apuntando a la dirección del SCORBOT define la dirección del eje X positivo y la proyección ortogonal fija el sentido del eje Y positivo en dirección del cuadro eléctrico del alimentador de bandejas de la célula.

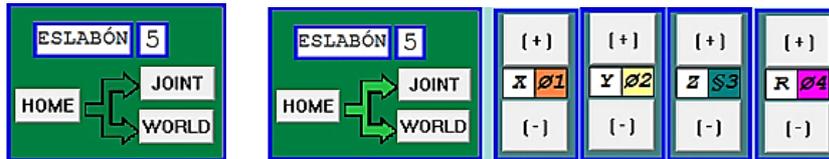


Figura 12-10. Renglón de la pantalla COMANDO para puesta a home y movimientos manuales.

Sobre el mismo renglón se aprecia que completar el HOME será indicado iluminando la flecha en verde, que a su vez dirige a continuar a alguno de los dos tipos de movimiento manual. Si la puesta a HOME fallare, se entraría en estado de emergencia y con la ayuda del número de eslabón se sabrá cual articulación ha incurrido en el fallo.

En la misma ilustración 12-10 y la 12-11 que sigue, se muestran los elementos para las maniobras de desplazamiento. Este se realizará siguiendo el sentido descrito al pulsar los botones [+] o [-] destinados a cada una de las articulaciones.

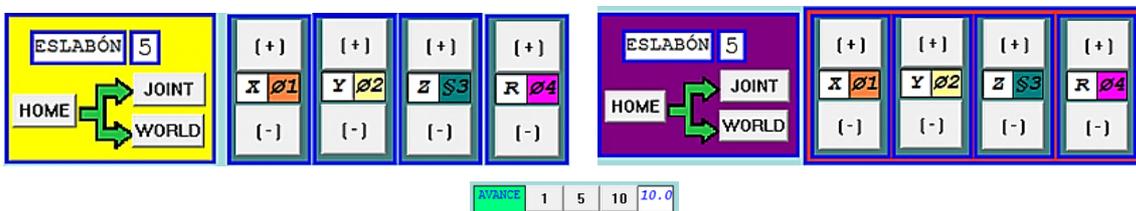


Figura 12-11. Renglón de la pantalla COMANDO para la ejecución de movimientos manuales.

Los pulsadores están sujetos a la selección de desplazamientos en coordenadas articulares, donde los eslabones se mueven de manera independiente para alcanzar el valor de consigna de avance fijada con los interruptores [1], [5] o [10]. O bien en coordenadas cartesianas o mundo como suele llamárseles, donde los eslabones se desplazan en conjunto producto del cálculo de la cinemática directa e inversa y que muestra su resultado de operación en las celdas de las coordenadas (X, Y, Z) del efector en milímetros y de las magnitudes articulares (TH1, TH2, DT\_3) de cada enlace en grados.

Los tipos de movimiento se indican con el cambio de color de fondo del pabellón que contiene los pulsadores

de selección de modalidad. De manera similar el fondo de los pulsadores incrementales cambiará de color, si el resultado de la estimación de la cinemática inversa derivaría en una singularidad del espacio de la tarea, en el caso en que el robot realizare el movimiento, que a modo de precaución se evita ejecutar. Para salir de esta última condición señalada, simplemente se ha de pulsar uno de los botones que encaucen al resultado a pertenecer al conjunto de posiciones solución, en otras palabras, solicitar un movimiento en sentido opuesto al que causó la singularidad.

Finalmente, en la figura siguiente se observa que el fondo de ambas visualizaciones cambia de color, en función del estado que el automatismo posea, siguiendo la normatividad de la guía *GEMMA*, que ha sido referida recientemente. Como ya se presentó en la figura 12-5, habrá de tornarse gris antes de ser energizada la unidad de potencia, para indicar que el *SCARA* está apagado.

Pasará a tornarse azul claro al encenderlo por medio del interruptor principal y volverá a este estado cada vez que la marcha del sistema se detenga. Siendo que al ser puesto en marcha el proceso, la interfaz se tornará de color verde.

Del mismo modo, el entrar al estado de emergencia, al no alcanzar el proceso de home o al quitar la energía de la unidad de potencia, se indicará con la interfaz en rojo y ameritará que se realice el rearmado del sistema con la pulsación del botón rearme, acción que derivará en llevar al fondo a ser de color amarillo, para indicarle al usuario que puede reanudar la marcha normal o salir del sistema mediante el uso del selector de tres posiciones a modo centro o ningún modo de operación pulsando el botón [N].

Estos cambios descritos, se muestran en la figura siguiente.



Figura 12-12. Cambios de pantallas de operador de acuerdo a la guía GEMMA.

## 12.2. ANEXO 2: PROGRAMA PRINCIPAL, PUESTA A HOME

En el programa principal se encuentra dispuesto el bloque de función para llevar el SRX 4CH a la posición de HOME. Este bloque de función se aprecia en la siguiente figura, que además permite observar cómo le han sido vinculadas desde el programa, las señales que intervienen sobre la codificación de la función, a través de las variables agrupadas en la tabla 12-3.

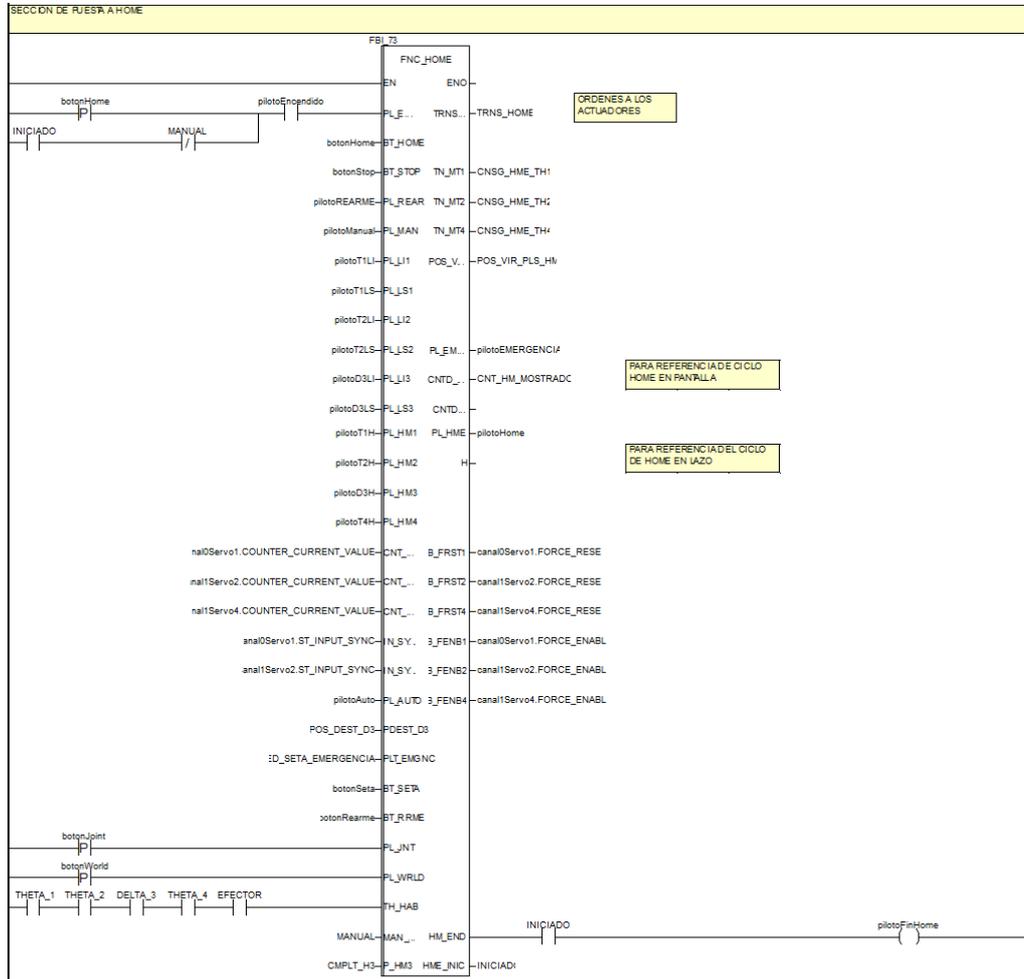


Figura 12-13. Sección del Programa Principal, Funcion Home.

La operación de puesta a Home del manipulador es un proceso de naturaleza secuencial, que se podría realizar de manera óptima a través del lenguaje SFC de Unity Pro, basado en el estándar *IEC-61131*, denominado *GRAFSET*.

Lamentablemente el software para programar el Modicon M340, no admite la utilización de Grafset en la creación de bloques derivados de función. Por lo que, para lograr aplicar esta alta operatividad en la ejecución de esta tarea en particular, el programa demandó considerar una estrategia para traducir secuencias cíclicas a lenguaje escalera. En la cual se emula la entrada a cada paso (*step*), con la inserción de una bobina tipo *SET* y el flanquear la transición, con la inserción de una bobina tipo *RESET*, que es vinculada a la misma variable.

Esto tiene como ventaja la inclusión de las acciones desde el mismo diagrama, sin la necesidad de secciones de código adicionales, con el coste de ver incrementada la extensión del programa, al necesitar ser incluidas en él, las condiciones de flanqueo década una de las transiciones.

### 12.2.1. ANEXO.2.1. ACCIÓN PST\_HOME

La sección de la unidad de programación que se muestra en la figura 12-14, pertenece al diagrama de contactos secuencial de nombre *PST\_HOME*, esta y otras secciones de programa o POU's adicionales, que se describen mas adelante, se ejecutan dentro de la función *FCN\_HOME*, y en conjunto complementan la ejecución del ciclo

total de ajuste de los eslabones del SONY SRX 4CH, con el fin de permitir delimitar el espacio de trabajo, a partir de la posición referencial inicial. Asimismo, son listadas en la tabla 12-3, como ya se ha mencionado, todas las variables que intervienen en la función de ajuste a home del manipulador.

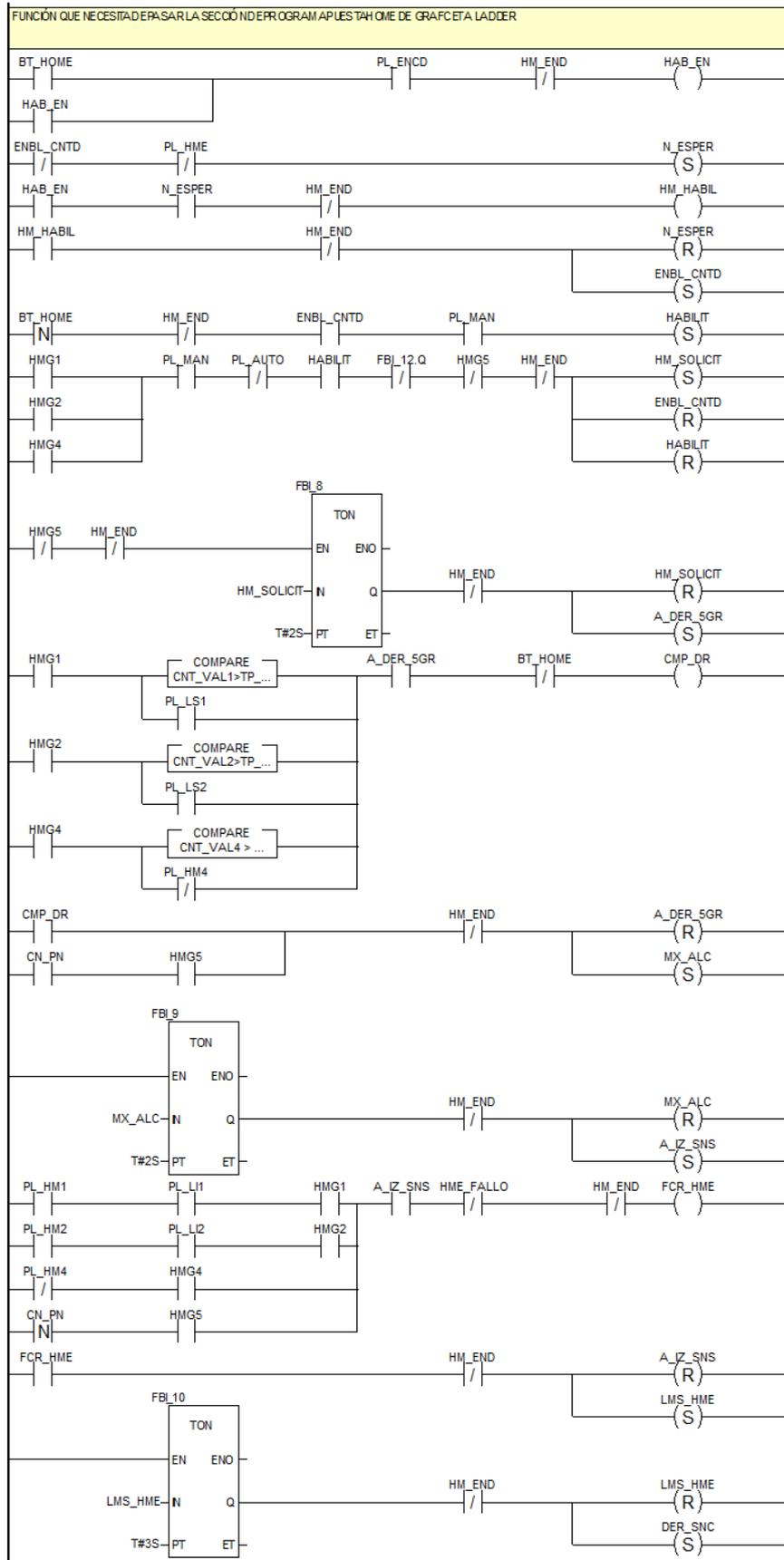


Figura 12-14. Bloque FNC\_HOME. Sección de programa PST\_HOME.

Tabla 12-3. Lista de variables del bloque de función FNC\_HOME

Lista de Variables del Bloque Derivado de Función FNC_HOME							
ENTRADAS			SALIDAS			PRIVADO	
#	Nombre	Tipo	#	Nombre	Tipo	Nombre	Tipo
1	PL_ENC	BOOL	1	TRNS_HOME_OUT	BYTE	SLD	BOOL
2	BT_HOME	BOOL	3	TN_MT1	INT	HAB_EN	BOOL
3	ST_STOP	BOOL	4	TN_MT2	INT	PNZ_H_OUT	BOOL
4	PL_REAR	BOOL	5	TN_MT4	INT	HMG1	BOOL
5	PL_MAN	BOOL	6	POS_VIR_pulsos	INT	HMG2	BOOL
6	PL_LI1	BOOL	9	PL_EMGRC	BOOL	HMG3	BOOL
7	PL_LS1	BOOL	10	CNTD_HMG_DSPL	INT	HMG4	BOOL
8	PL_LI2	BOOL	11	CNTD_HMG	INT	HMG5	BOOL
9	PL_LS2	BOOL	12	PL_HME	BOOL	PL_HM	BOOL
10	PL_LI3	BOOL	13	H	INT	TP_CNT1	DINT
11	PL_LS3	BOOL	16	B_FRST1	BOOL	TP_CNT2	DINT
12	PL_HM1	BOOL	17	B_FRST2	BOOL	TP_CNT4	DINT
13	PL_HM2	BOOL	18	B_FRST4	BOOL	N_ESPER	BOOL
14	PL_HM3	BOOL	19	B_FENB1	BOOL	HM_HABIL	BOOL
15	PL_HM4	BOOL	20	B_FENB2	BOOL	ENBL_CNTD	BOOL
16	CNT_VAL1	DINT	21	B_FENB4	BOOL	HABILIT	BOOL
17	CNT_VAL2	DINT	29	HME_END	BOOL	HM_SOLICIT	BOOL
18	CNT_VAL4	DINT	30	HME_INIC	BOOL	A_DER_5GR	BOOL
19	IN_SYNC1	EBOOL				CMP_DR	BOOL
20	IN_SYNC2	EBOOL				MX_ALC	BOOL
21	PL_AUTO	BOOL				A_IZ_SNS	BOOL
22	PDEST_D3	EBOOL				FCR_HME	BOOL
23	PLT_EMGNC	BOOL				DER_SNC	BOOL
24	BT_SETA	BOOL				SN_NMB	BOOL
25	BT_RRME	BOOL				HME_CMPLET	BOOL
26	PL_JNT	BOOL				ARTC_SGTE	BOOL
27	PL_WRLD	BOOL				RBT_HME	BOOL
28	TH_HAB	BOOL				EMERGENCIA	BOOL
29	MAN_ACT	BOOL				WD_TIM	BOOL
30	P_HM3	BOOL				WD_TIM_ON	BOOL
						AUX_SYNC	BOOL
						CN_PN	BOOL
						PL_EMGR	BOOL
						AC_T1	BOOL
						AC_T2	BOOL
						AC_D3	BOOL
						AC_T4	BOOL
						HM_PRCs	BOOL
						RF_RL	REAL
						INST	BOOL

La secuencia cíclica de este programa se ha de ejecutar para cada articulación por separado, para su puesta a HOME y está comprendida por el seguimiento de los pasos del siguiente algoritmo:

1. Aguardar a que la solicitud de ejecución del proceso se emita desde la estación de explotación. Siendo en este paso inmediatamente realizada, la habilitación de las tarjetas de conteo rápido del autómata, mediante la conmutación de los bits *FORCE\_RESET* y *FORCE\_ENABLE* de cada canal de conteo, de los módulos EHC-0200 incluidos en el PLC. Esta acción es llevada a cabo a partir de la cuarta sección del programa *PST\_HOME* que se ilustra en la figura 12-17.
2. Habiendo establecido la configuración de las tarjetas y ante la observación de la orden de marcha, se inicia el movimiento del eslabón a baja velocidad con sentido positivo (a derechas), hasta alcanzar 10 grados o 10 milímetros según el tipo de articulación. Esta acción se ejecuta desde la sección de programa *GIR\_mot* que pertenece a la función, y hace desplazar el eslabón una distancia preconcebida suficiente, para que esté fuera del umbral de detección del sensor límite inferior de la carrera, siendo el caso de encontrarse inicialmente ubicado el eslabón, dentro de ese rango de valores posicionales.
3. Dicho movimiento a derechas puede conducir a dos estados, el primero de ellos es el caso ideal en donde se alcanzan los 10 grados o milímetros definidos como límite de avance, en la sección *TOPE\_AVANC*. Se detendría el giro posteriormente al ser alcanzada la condición. El segundo caso, se da cuando se ha alcanzado el sensor del límite de carrera superior, con lo cual el motor ha de detenerse de inmediato para no incurrir en anomalías en el manipulador. La detención del motor también se ejecuta desde la unidad de programación *GIR\_mot*.
4. Seguidamente emplea la misma sección para iniciar la marcha con la polarización negativa del motor (a izquierdas), hasta encontrar el sensor de HOME. Esta acción precisa de una consideración especial en el manejo de la articulación prismática, esta está en un lazo abierto de posiciones virtuales para la generación de los pasos del motor, que resulta complejo de controlar desde la función de home, razón por la cual la acción sobre la articulación se coordina desde el mismo bloque de función *PST-HOME*, pero se realiza dentro del bloque *Mv\_Mt\_Rap*, que se creó para comandar el motor a pasos que acciona el eslabón.
5. Luego de ir al sensor de HOME, cada eslabón se desplaza nuevamente a derechas, en la búsqueda del bit de sincronía del codificador incremental que es leído por la tarjeta rápida mediante la entrada *IN\_SYNC*. De no manejarse esta señal por razones de hardware, el movimiento lleva a la articulación a abandonar el sensor de HOME y a partir del primer pulso en el que no lo perciba, comenzará a referenciar el espacio de trabajo de la articulación. Esta condición además detiene el temporizador que limita la duración del proceso de puesta a Home de cada enlace de la cadena articular, el cual inicia su marcha de manera individual, al momento en que comienza el movimiento de cada eslabón.
6. Para finalizar vuelve a forzar el bit que establece el inicio de conteo a cero, o hace la referencia de pulsos cero, como en el caso de tratarse del motor a pasos y luego avanza a la próxima articulación. La variable *ARTC\_SGTE* incrementa un contador destinado a totalizar las cuatro articulaciones del SONY SRX 4CH y así va evolucionando el proceso.
7. Posterior al haber completado los procedimientos de los cuatro (4) grados de libertad del SCARA, se realiza la comprobación de funcionamiento del efector adosado al robot, con lo cual el operador observa su actuación y certifica que el autómata ha completado el proceso, dejando al SCARA en la posición referencial "Home". La finalización se señala en la pantalla de operador tornando verde la flecha que notifica al usuario que puede continuar con la manipulación del robot, pasando a operaciones en automático o realizar movimientos manuales en orden articular o cartesiano del robot, con la certeza de ser obedecidas en el espacio real de la tarea, todas las indicaciones que emite desde la estación de operación.

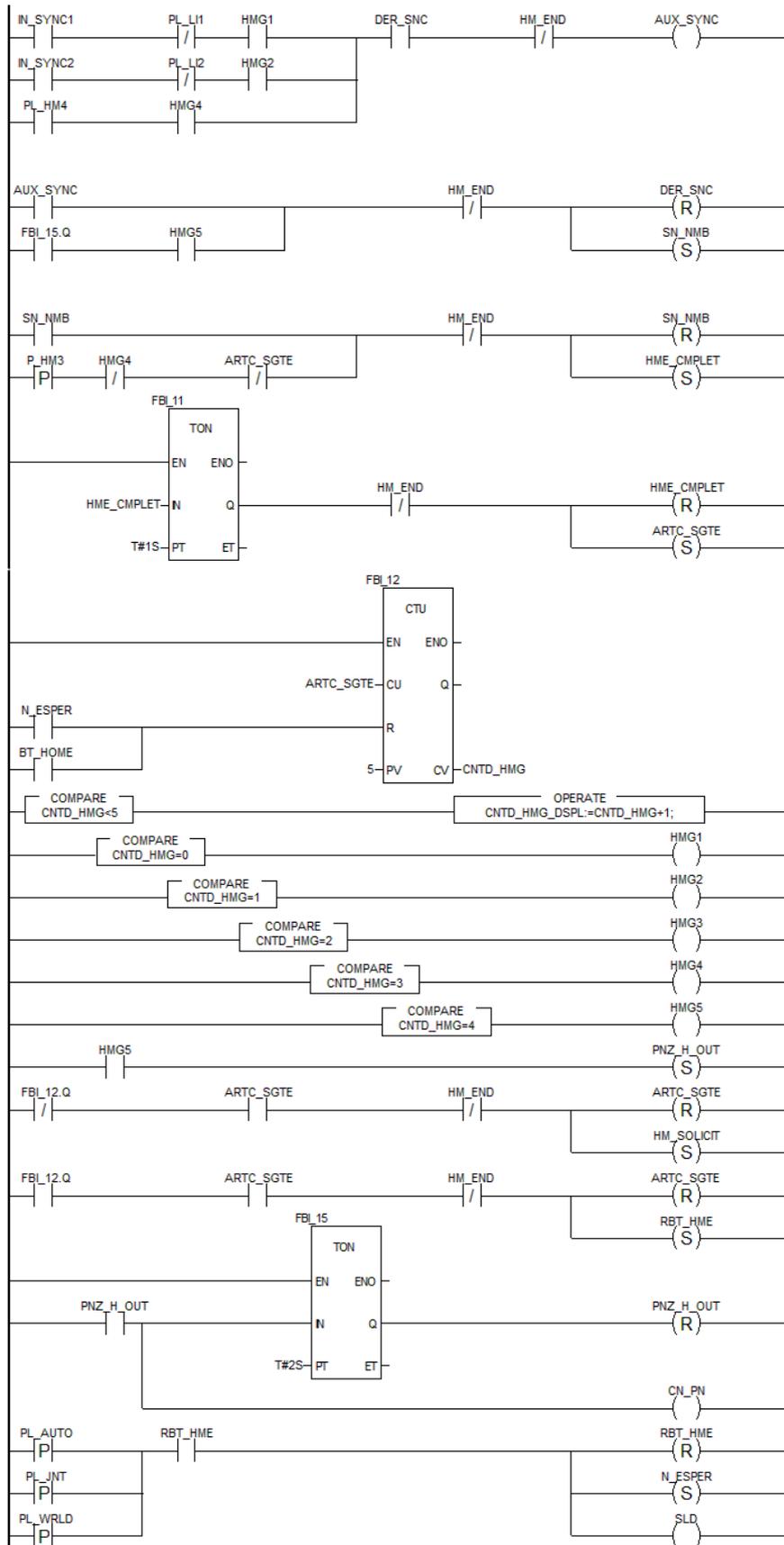


Figura 12-15. Sección de programa PST\_HOME, segunda parte.

En la figura 12-15 se puede apreciar que a partir de la línea 80 del diagrama, se comienza a coordinar acciones de supervisión de las condiciones del proceso, habiendo finalizado el ciclo en las líneas superiores.

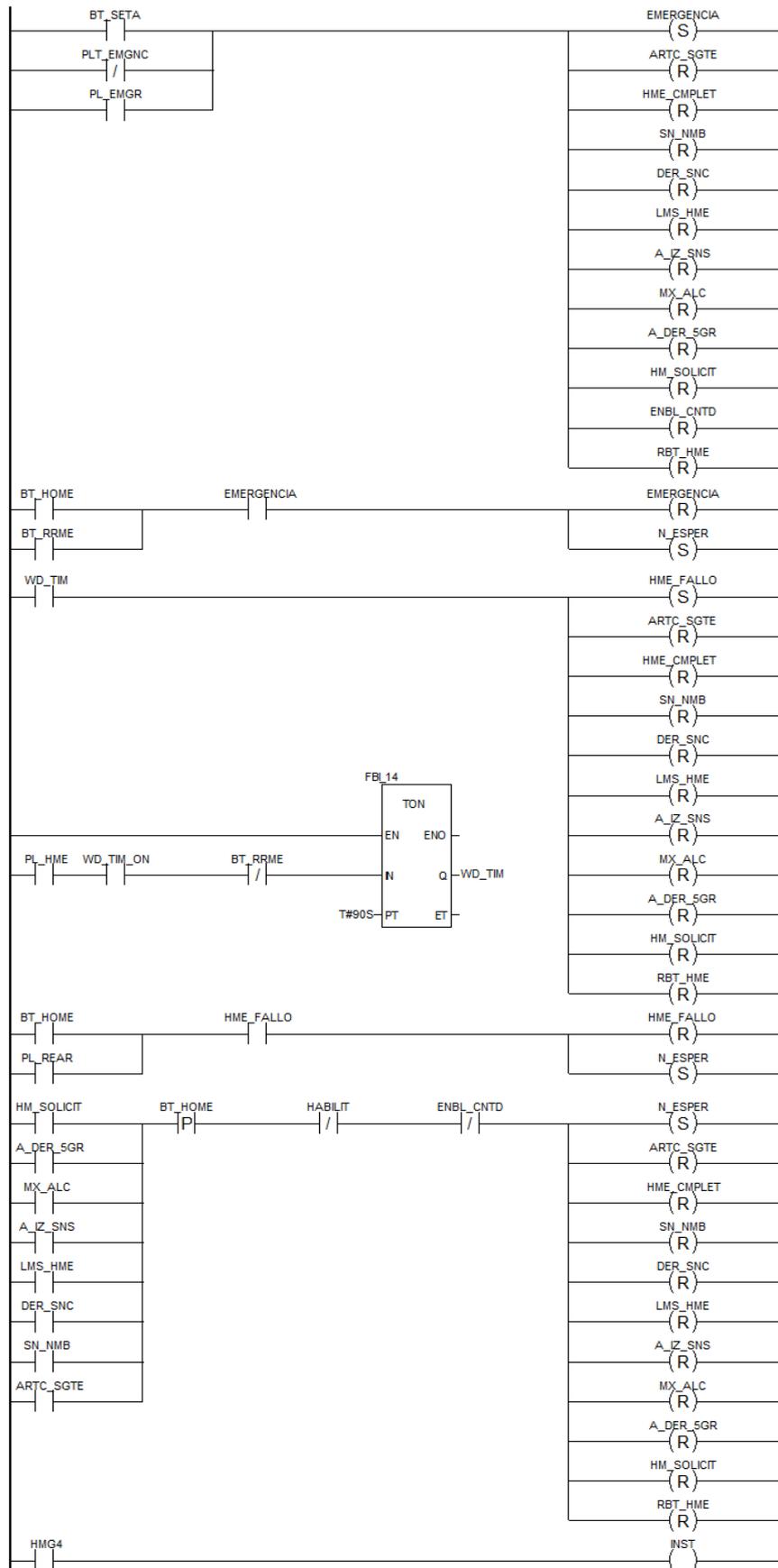


Figura 12-16. Sección de programa PST\_HOME, tercera parte.

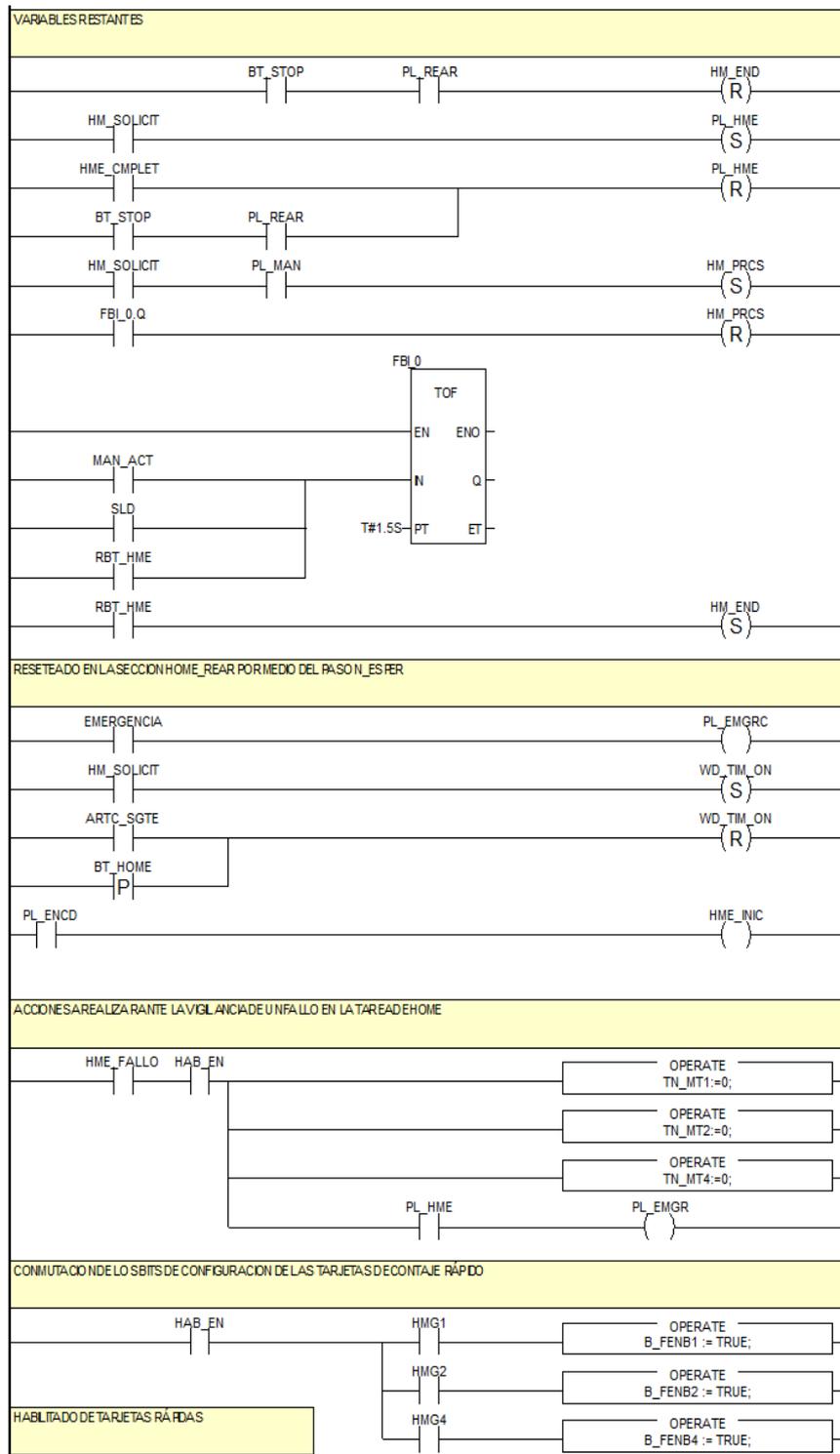


Figura 12-17. Sección de programa PST\_HOME, cuarta parte.

A partir de las líneas o redes de programa de la figura 12-17, la sección de programa se dedica a vigilar las condiciones que han de concluir el proceso y dejar el control de la unidad en otras funciones del programa.

Estas situaciones bien pudieran ser, el haber completado el proceso de manera exitosa y el operador pasa a realizar otras tareas con el manipulador, o bien pudieran tratarse de una situación no deseada que haya hecho preciso que la seta de emergencia se pulsase en algún momento, de igual manera pudiere ser producto de la vigilancia, sin la actuación del usuario, de aquellas condiciones que pudieren conducir al sistema a entrar en una condición anómala, mientras se evoluciona en el correcto funcionamiento del robot, también pudieran darse casos, en los cuales dicha situación no acorde ya haya ocurrido y por esta razón, se ha vencido el tiempo sujeto a un temporizador perro guardián, que se asegura de lanzar una condición de emergencia si alguna articulación

no culminase la rutina dentro del tiempo que se ha estimado, en el cual habría de cumplirse el proceso y por ello el sistema ha entrado en estado de emergencia o ha devenido en un fallo del robot.

Ante estas posibles situaciones se ha previsto las acciones de movimiento que se deben tomar (detener la marcha prioritariamente) y las respuestas de comportamiento que se han definido para el automatismo, según los lineamientos de la Guía GEMMA, que señala que se ha de conducir el estado actual, a un estado detenido “EMERGENCIA”, e indicar dicho estado en la pantalla del operador, pudiendo salir posteriormente de esta condición, con el botón de rearme dispuesto en la interfaz.

En estos casos el operador puede apoyarse en el contador asociado a la variable *CNT\_HMG\_DSPL* del programa, la cual permite conocer de inmediato cual articulación derivó en la falla del proceso y de esta manera tomar las acciones de corrección del error.

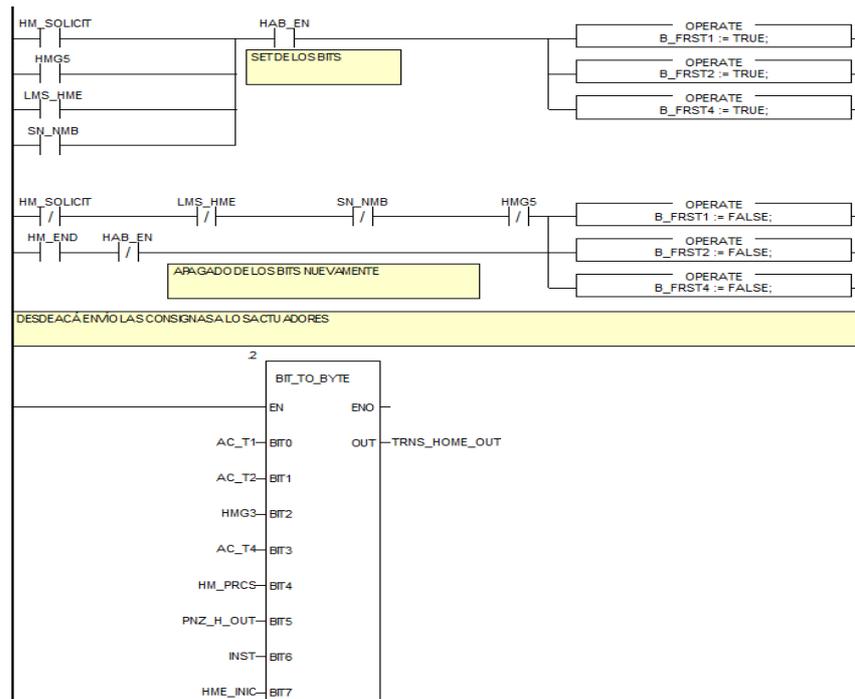


Figura 12-18 . Sección de programa *PST\_HOME*, quinta parte.

La sección final del diagrama *PST\_HOME* que se observa en la figura 12-18 se incluye, para permitirle modificar el estado de los bits de la tarjeta de contaje rápido que han sido dispuestas en el autómata y han de habilitarse para que se ejecute de manera confiable la supervisión del avance de los motores, mediante la monitorización de los pulsos de los encoders que están asociados a las articulaciones. Los módulos también son responsables de suplir al driver del stepper, los pulsos que conducen al avance del motor a pasos.

Como puede observarse también en la figura, los bits de habilitación del funcionamiento del canal (*FORCE ENABLE*) se encienden con la energización del sistema y guardan esa condición indefinidamente, hasta que se apague la unidad o se haga la detención del autómata y por consiguiente del programa que este contiene.

La ilustración muestra además, que los bits de reinicio (*FORCE RESET*) del canal de contaje rápido, han de ser colocados en alto un instante para habilitarlo en la puesta en funcionamiento inicial y cada vez que se desee referenciar el punto de partida del conteo de pulsos o inicio de la carrera, para lograr dimensionar el desplazamiento del eslabón. Asimismo, estos bits de la memoria se observan con la ayuda de la pantalla de explotación cuando se encuentran energizados.

Finalmente, se envían por medio de la salida *TRNS\_HOME\_OUT*, todos los bits que el bloque de función para gobernar el movimiento de los actuadores precisa, a razón de que el SCARA responda a las instrucciones del bloque *FCN\_HOME*.

## 12.2.2. ANEXO.2.2. SECCIÓN *GIR\_mot*

La sección de la ilustración siguiente permite gobernar los actuadores del robot desde la función de puesta a HOME. Desde la sección se modifica la variable correspondiente a cada uno de los canales analógicos de salida

y se consigue que los motores giren a la derecha o izquierda, según lo acordado inicialmente en el apartado 3.3 de la memoria, donde son definidos el sentido del avance de las articulaciones, con la polarización o bien de manera positiva o negativa del motor. Recíprocamente también se muestra en la figura, que al detectar la llegada al punto de consigna o bien a los micros de detención de home o extremos de recorrido, se detendrá la marcha del motor, al guardar en la variable el valor de consigna correspondiente a 0 voltios.

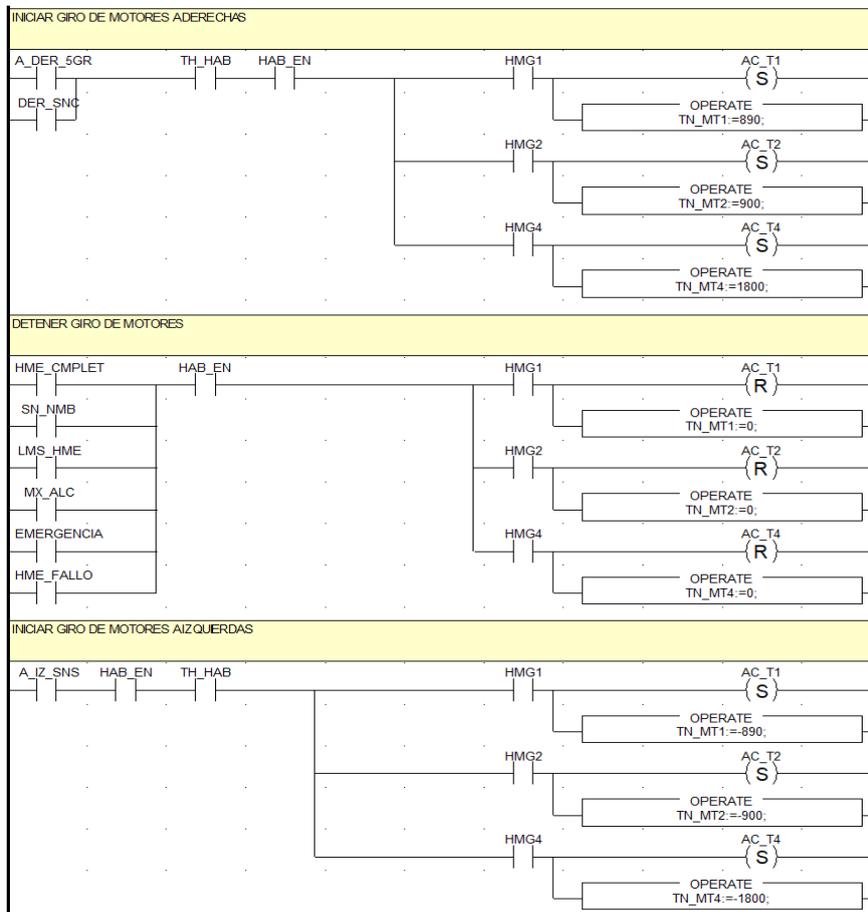


Figura 12-19. Sección de programa GIR\_mot.

### 12.2.3. ANEXO.2.3. SECCIÓN TOPE\_AVANC

Dentro de la función la sección *TOPE\_AVANC* que ha sido escrita en texto estructurado (ST), se busca asegurar la detención de todo movimiento que se realizase en el momento de la solicitud de ajuste a HOME y se encarga de fijar el límite del recorrido que se realizará a la derecha al inicio de los desplazamientos, esta es una distancia de 10 grados, como se puede observar en el código de la figura 12-20.

```
(*SECCION DE DETENCIÓN INICIAL DE LOS MOTORES DE LAS ARTICULACIONES DEL SCARA*)
IF (ENBL_CNTD = TRUE AND HMG2 = FALSE AND HMG3 = FALSE AND HMG4 = FALSE) THEN
  C_T1 := FALSE;
  AC_T2 := FALSE;
  TN_MT1 := 0;
  TN_MT2 := 0;
  POS_VR_pulsos := 0;
END_IF;
(*SECCION DE HABILITACION DE LA CAPTURA DEL VALOR DEL ENCODER EN LA POSICION CONSEGUIDA*)
IF (HM_SOLICIT = TRUE) THEN
  TP_CNT1 := 3340; (*EL ORDINAL 1710 CORRESPONDE A 10 GRADOS DE ROTACION*)
  TP_CNT2 := 2770; (*EL ORDINAL 2820 CORRESPONDE A 10 GRADOS DE ROTACION*)
  TP_CNT4 := 1000; (*EL ORDINAL 1000 CORRESPONDE A 10 GRADOS DE ROTACION*)
  (*TP_CNT3B := DINT_TO_REAL (TP_CNT3A);*)
END_IF;
```

Figura 12-20. Sección TOPE\_AVANC.

Con este planteamiento se logra evitar que se detecten erróneamente los finales de carrera de límite inferior o home, de estar iniciando el proceso de home desde esa ubicación inicial.

### 12.2.4. ANEXO.2.4. SECCIÓN SEC

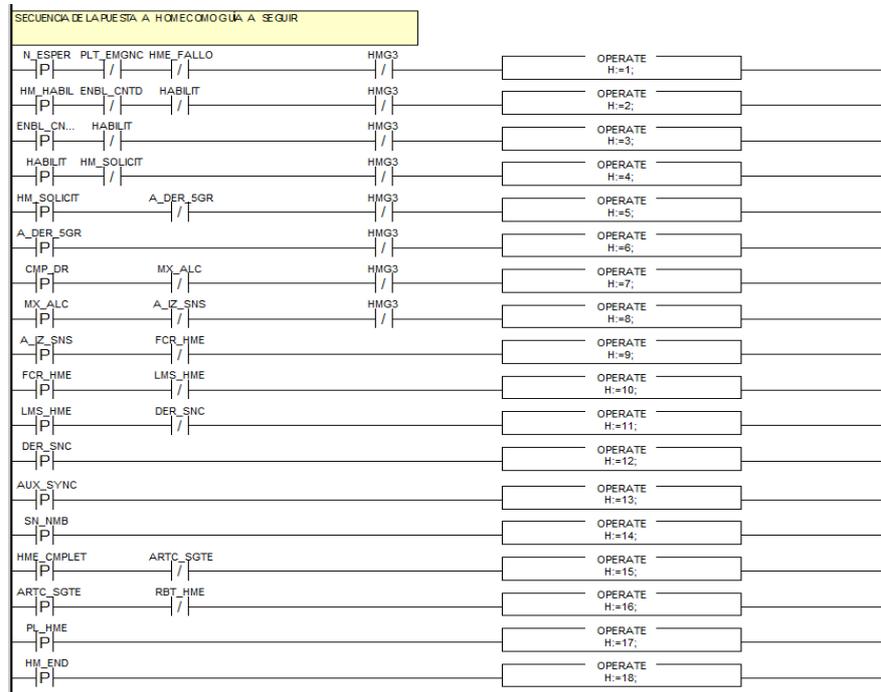


Figura 12-21. Sección de programa de supervisión SEC.

La sección de programa *SEC* se ofrece como una recomendación, no cumple acciones en el robot, pero sirve para visualizar a través de la salida del bloque de función *H*, en que parte del ciclo se encontraba la función marchando. En Unity Pro el diagrama dentro del bloque no se anima en línea (cambia de color los elementos y nombres de variable del diagrama), como lo hace el programa principal del autómata, por ello seguir la ejecución amerita un indicador externo adicional, para supervisar la evolución del programa desde la estación de programación. Esta es la intención de la sección *SEC* y se emplea esta estrategia en diferentes funciones del programa

### 12.3. ANEXO 3: PROGRAMA PRINCIPAL, MOVIMIENTO MANUAL

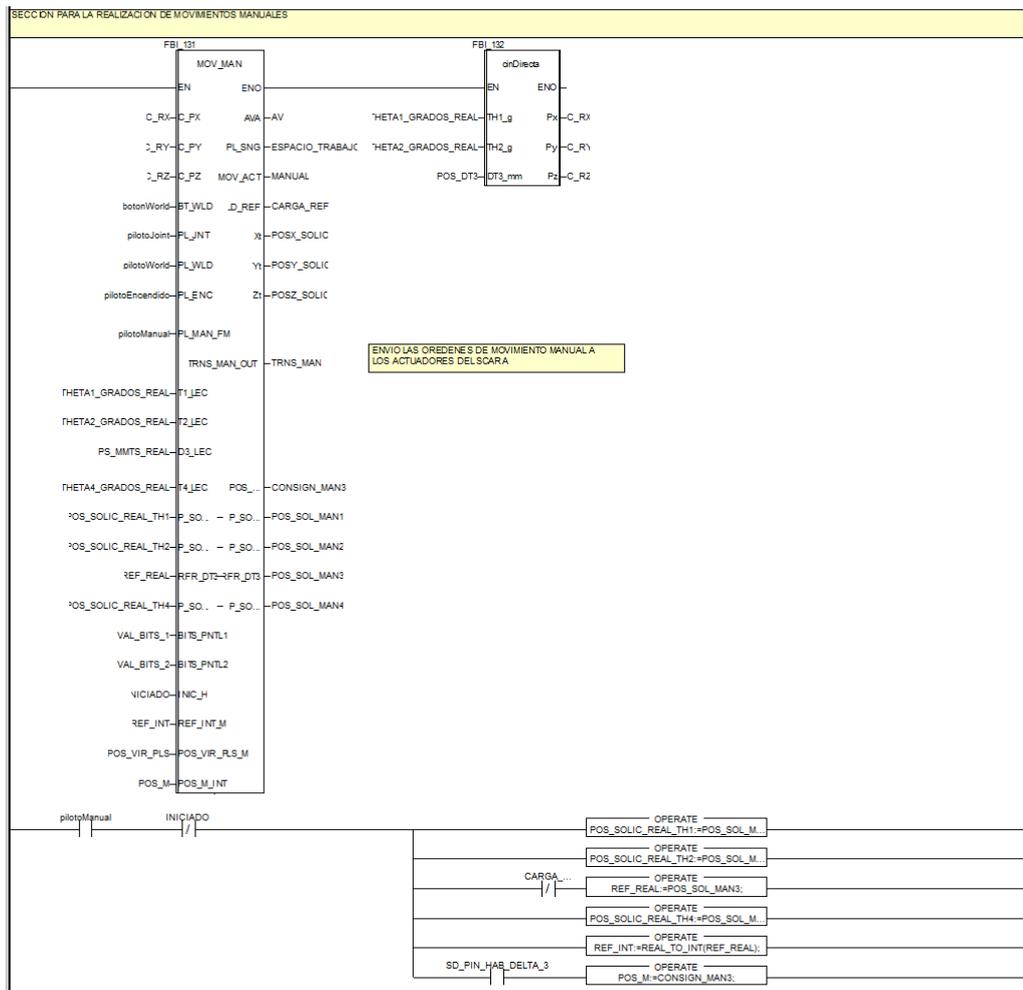


Figura 12-22. Sección del Programa Principal, Movimiento Manual.

En la figura 12-22 tomada de la sección del programa principal se observa, cómo se han vinculado las variables a las entradas y salidas del bloque de función, cuyas variables internas se listan en la tabla 12-4. La sección ha sido creada como respuesta, a la necesidad de permitirle al usuario controlar el brazo manipulador y tiene la finalidad de familiarizarle con su operación, además con ella podrá visualizar que puede conseguir con las cualidades que posee el dispositivo y finalmente podrá conformar el grupo de configuraciones articulares, dentro del arreglo matricial que habrá de emplear el programa secuencial de tareas automáticas, que el mismo usuario disponga a llevar a cabo.

También se logra apreciar una vinculación directa entre las funciones *MOV\_MAN* y *cinDirecta*, esta última le ofrece constantemente la ubicación del efector, estimándola desde la información que capta desde el bloque de función *ACTUADORES*. En ese mismo sentido, la misma ilustración expresa cómo el bloque derivado de función para la resolución de la cinemática directa utiliza la posición en grados de las dos primeras articulaciones rotacionales y de la articulación prismática como entradas para la realización del cálculo, estas se reciben por medio de las entradas y el resultado se ofrece en las salidas que se listan en la tabla 12-5.

Por otro lado, la función de movimiento manual escribe sobre el bloque que controla la actuación de los motores y el efector. Existe entonces una realimentación constante entre los diferentes elementos de la rutina principal. Esta relación precisó que se estableciese la separación de las solicitudes de posición a visitar, con la intención de evitar que estas instrucciones de desplazamiento de motor, se sobrescribiesen al ser invocadas desde diferentes secciones del programa. En el caso específico de las consignas que emite la función para llevar a cabo los movimientos manuales, se ha conseguido particularizar mediante la inclusión de bloques *OPERATE*, que se ubican en las líneas posteriores al bloque funcional, sobre el diagrama de la sección de la rutina programa principal.

Tabla 12-4. Lista de variables del bloque de función MOV\_MAN.

Lista de Variables del Bloque Derivado de Función MOV_MAN									
ENTRADAS			SALIDAS			PRIVADO			
#	Nombre	Tipo	#	Nombre	Tipo	Nombre	Tipo		
1	C_PX	REAL	2	AVA	REAL	CARTESIANO	BOOL		
2	C_PY	REAL	3	PL_SNGL	BOOL	ENBL	BOOL		
3	C_PZ	REAL	7	HB_ART1	BOOL	C_TH1_INT	INT		
4	BT_WLD	BOOL	8	HB_ART2	BOOL	C_TH2_INT	INT		
5	PL_JNT	BOOL	9	HB_ART3	BOOL	C_D3_INT	INT		
6	PL_WLD	BOOL	10	HB_ART4	BOOL	ARTICULAR	BOOL		
7	PL_ENC	BOOL	11	Xt	REAL	EQ_POS1	BOOL		
8	PL_MAN	BOOL	12	Yt	REAL	EQ_POS2	BOOL		
9	BT_MAS1	BOOL	13	Zt	REAL	EQ_POS3	BOOL		
10	BT_MNS1	BOOL				PERMITIDO	BOOL		
11	BT_MAS2	BOOL	<b>ENTRADAS – SALIDAS</b>						
12	BT_MNS2	BOOL	#	Nombre	Tipo				
13	BT_MAS3	BOOL	22	P_SOLR_TH1	REAL				
14	BT_MNS3	BOOL	23	P_SOLR_TH2	REAL				
15	BT_MAS4	BOOL	24	RFR_DT3	REAL				
16	BT_MNS4	BOOL	25	P_SOLR_TH4	REAL				
17	AVNC	INT							
18	T1_LEC	REAL							
19	T2_LEC	REAL							
20	D3_LEC	REAL							
21	T4_LEC	REAL							

### 12.3.1. ANEXO 3.1: CINEMÁTICA DIRECTA

La figura 12-23 permite observar que la ubicación del efector se estima constantemente, para desplegarla en la interfaz y ofrecerla a la función cinemática inversa como datos de entrada para definir las coordenadas articulares del manipulador. De modo similar la tabla 12-5 ofrece listadas las variables que emplea la función.

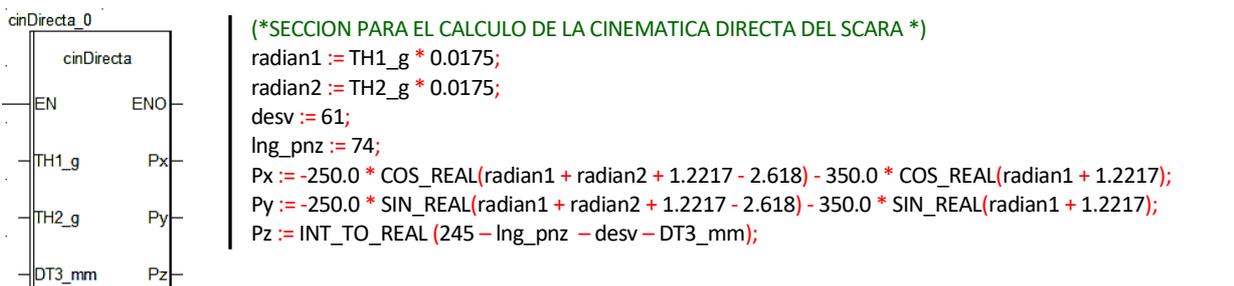


Figura 12-23. Bloque Cin\_Directa. Sección de programa Cin\_Directa.

Tabla 12-5. Lista de variables del bloque de función cinDirecta.

Lista de Variables del Bloque Derivado de Función cinDirecta							
ENTRADAS			SALIDAS			PRIVADO	
#	Nombre	Tipo	#	Nombre	Tipo	Nombre	Tipo
1	TH1_g	REAL	1	Px	REAL	radian1	REAL
2	TH2_g	REAL	2	Py	REAL	radian 2	REAL
3	DT3_mm	REAL	3	Pz	REAL	lng_pnz	INT
						Desv	INT

### 12.3.2. ANEXO 3.2: MOVIMIENTOS ARTICULAR Y CARTESIANO

El diagrama escalera *S\_MAN* del bloque derivado de función para la realización de movimientos manuales, cuenta con otros bloques que han sido creados con el propósito de sustentar la operación. Estos se ocupan de estimar los límites de los avances, prevenir errores de emisión de consignas, cuya ejecución no pertenecerá al conjunto de posiciones alcanzables en el entorno real, gracias a la capacidad de resolver el problema cinemático inverso.

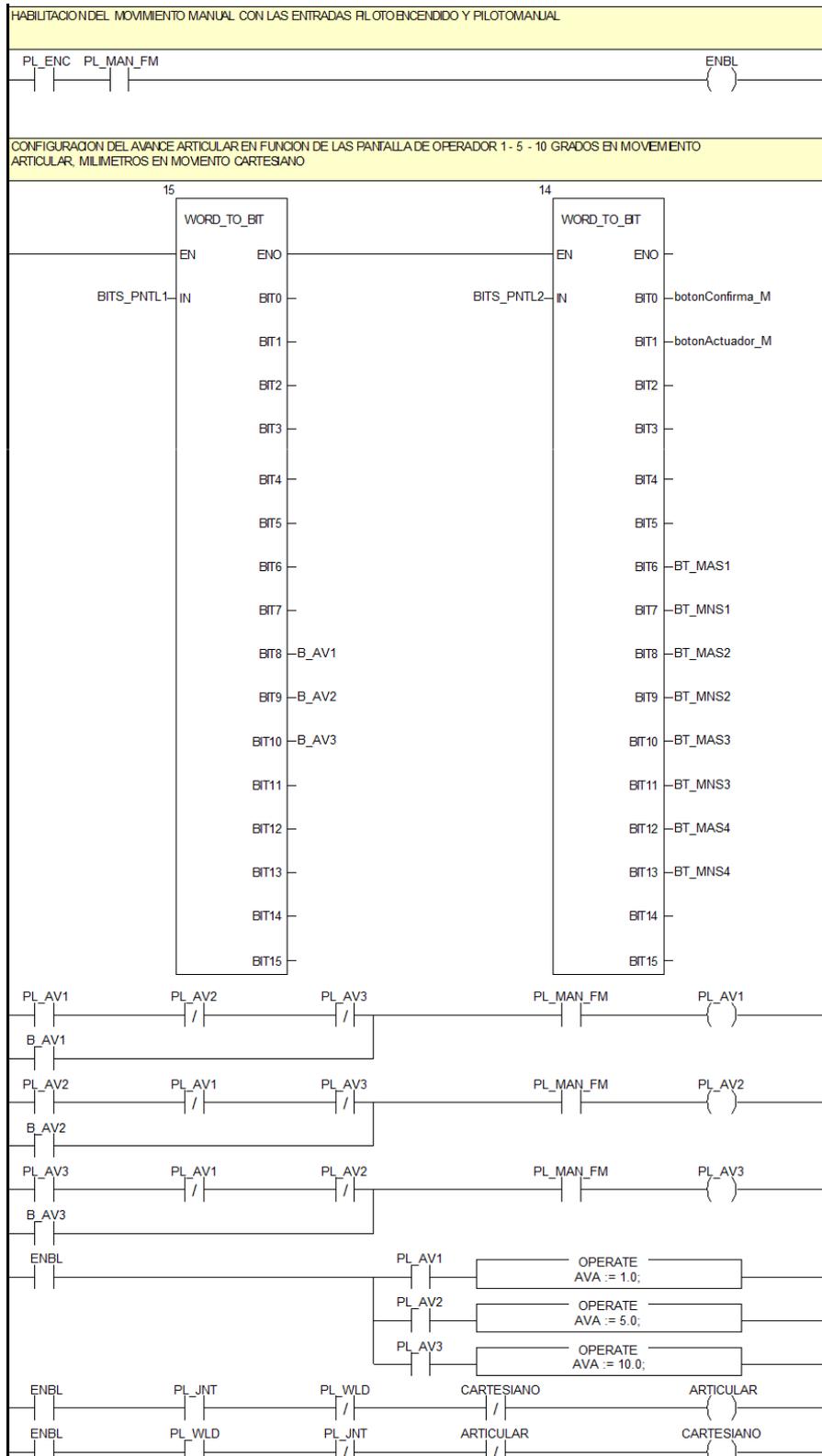


Figura 12-24. Sección del programa S\_MAN.

Con estas herramientas el bloque principal posee la configuración necesaria para que se ejecuten movimientos en coordenadas articulares, en las que se emiten consignas a cada eslabón por separado para que este se desplace a la posición deseada, o en coordenadas cartesianas o coordenadas mundo como suele llamarseles, en donde las articulaciones avanzan en conjunto para que el efector ejecute desplazamientos en correspondencia con los ejes que definen el espacio tridimensional. Es recomendable usar el movimiento articular, que es significativamente más sencillo en procesamiento computacional, para desplazar el efector con mayor rapidez en avances extensos. Al llegar a la proximidad de la posición deseada a referenciar, se ha de elegir el desplazamiento en coordenadas mundo con lo cual se ha de ejercer un control referencial mas alto.

### 12.3.3. ANEXO 3.3: FUNCIÓN DF\_LIM

Cuando es seleccionado el modo de movimiento en coordenadas articulares mediante el pulsador JOINT, el programa del SCARA habilita a la función encargada de definir los límites en los cuales se puede mover el eslabón, para que el operador haciendo uso de los pulsadores de incremento y decremento de la magnitud de posición de la interfaz *COMANDO*, adicione o sustraiga al valor previo de la variable que se modifica, el factor de aporte, definido por los botones *AVANCE*, que haya elegido para desplazarle.

La función *DF\_LIM*, descrita en la ilustración 12-25, ha sido alimentada con los límites definidos en la figura 3-4, que muestra el espacio de trabajo del SCARA. La función los emplea en un bloque derivado de función de un nivel inferior, que algebraicamente realiza la comparación en función de la magnitud del avance y la ubicación actual del eslabón. Por ejemplo, si se fijase un avance de 5° por cada pulsación y se encontrase la articulación 1 en 4° no se admitirían decrementos de la posición actual. Del mismo modo si se encontrase en 217° no se admitirían incrementos, ya que el resultado se encontrarían fuera de los 220° que comprende el rango total de la articulación mencionada.

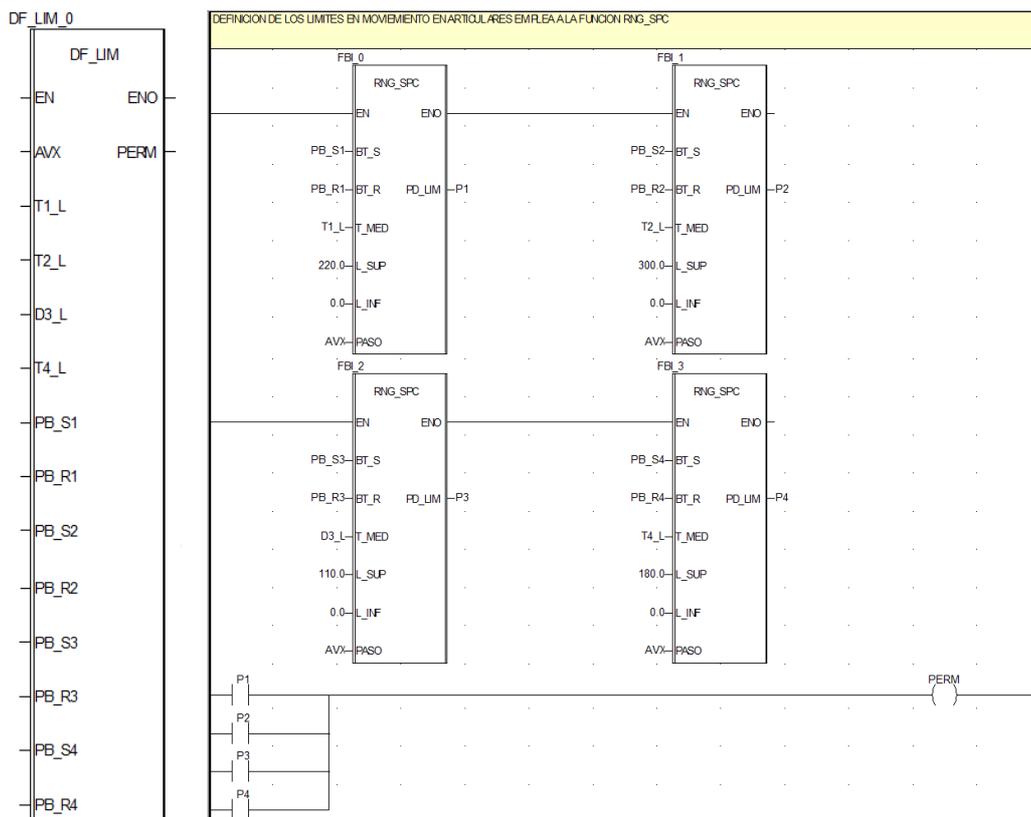


Figura 12-25. Bloque DF\_LIM. Sección del programa CAL\_LIM.

La función para el movimiento manual, maneja cada enlace bajo este planteamiento, siendo señalados como entrada los diferentes rangos de movimiento. Así se coloca en alto el permisivo de marcha de ser la condición verdadera. Para *DF\_LIM* en la tabla siguiente y tal como se ha hecho con los bloques derivados de función, se hace una lista de todas las variables de entrada, salida e internas, que se le vinculan desde la sección de programa *S\_MANUAL*.

Tabla 12-5. Lista de variables del bloque de función DF\_LIM.

Lista de Variables del Bloque Derivado de Función DF_LIM							
ENTRADAS			SALIDAS			PRIVADO	
#	Nombre	Tipo	#	Nombre	Tipo	Nombre	Tipo
1	AVX	REAL	1	PERM	BOOL	PL_AV1	BOOL
2	T1_L	REAL				PL_AV2	BOOL
3	T2_L	REAL				PL_AV3	BOOL
4	D3_L	REAL					
5	T4_L	REAL					
6	PB_S1	BOOL					
7	PB_R1	BOOL					
8	PB_S2	BOOL					
9	PB_R2	BOOL					
10	PB_S3	BOOL					
11	PB_R3	BOOL					
12	PB_S4	BOOL					
13	PB_R4	BOOL					

12.3.4. ANEXO 3.4: FUNCIÓN RGN\_SPC

En Unity Pro se posibilita insertar bloques derivados de función dentro de otros, admitiendo hasta ocho niveles embebidos desde la función principal. Esta condición facilitó el empleo de la función *RGN\_SPC* para asistir al programa alojado en *DF\_LIM*, en la omisión de posiciones destino que no perteneciesen al espacio articular. La siguiente ilustración muestra el diagrama del programa y la tabla 12-7 incluye las variables que emplea en su operación.

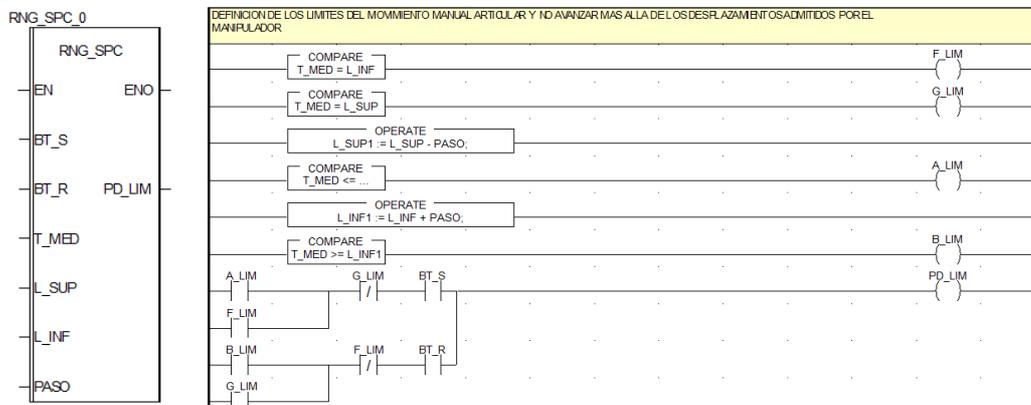


Figura 12-26. Bloque RGN\_SPC. Sección de programa DEF\_LIM1.

Tabla 12-7. Lista de variables del bloque de función.

Lista de Variables del Bloque Derivado de Función RGN_SPC							
ENTRADAS			SALIDAS			PRIVADO	
#	Nombre	Tipo	#	Nombre	Tipo	Nombre	Tipo
1	BT_S	BOOL	1	PD_LIM	BOOL	L_INF1	REAL
2	BT_R	BOOL	2			L_SUP1	BOOL
3	T_MED	REAL	3			F_LIM	BOOL
4	L_SUP	REAL	4			G_LIM	BOOL
5	L_INF	REAL	5			A_LIM	BOOL
6	PASO	REAL	6			B_LIM	BOOL

Por otro lado, cuando se elije el modo de movimiento en coordenadas cartesianas, con el uso del pulsador *WORLD*, el programa del autómata habilita a la función que resuelve el problema cinemático inverso del SCARA.

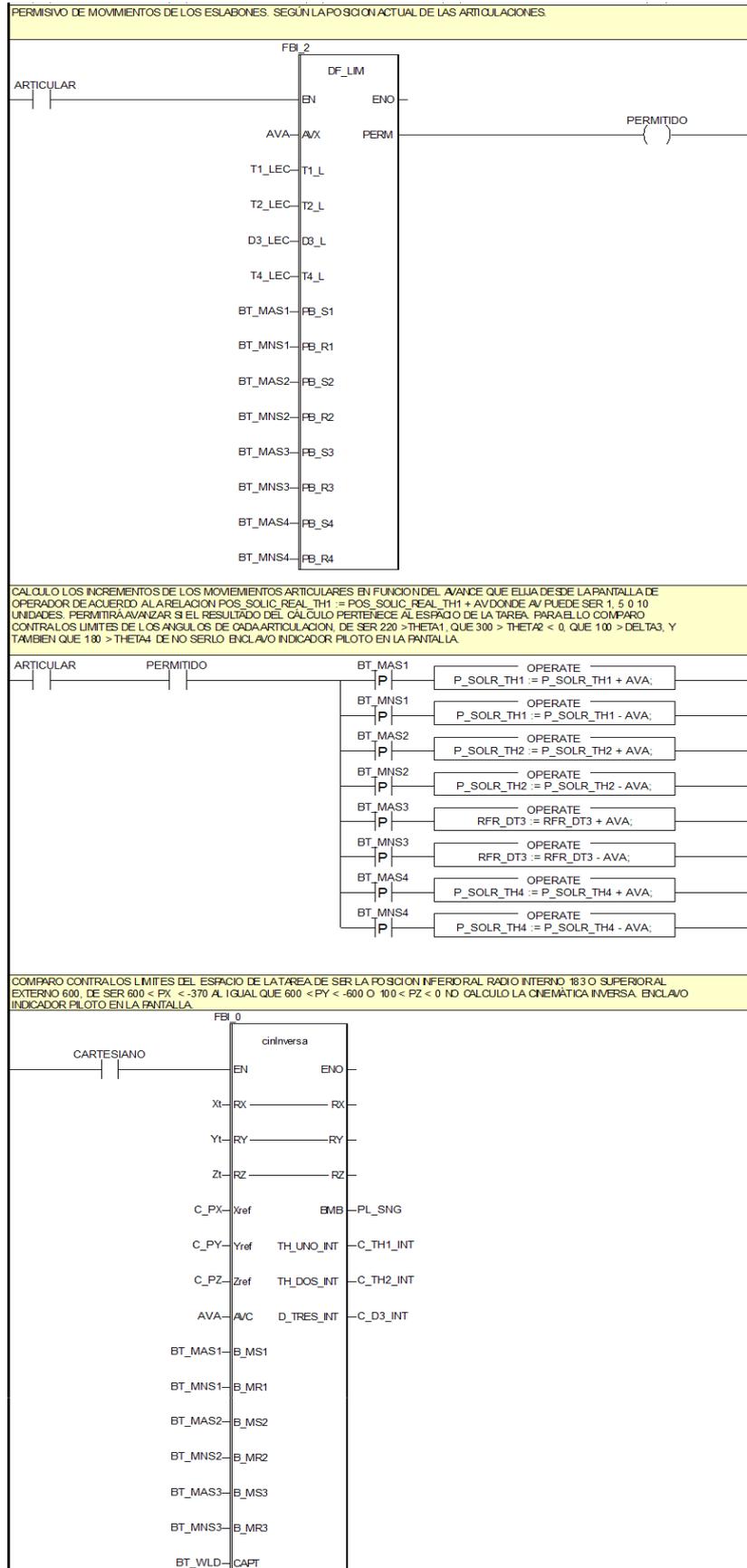


Figura 12-27. Sección del programa S\_MAN, segunda parte.

A partir de ese instante, el operador utiliza los pulsadores de incremento y decremento de la magnitud de posición de la interfaz *COMANDO*, adicionando o sustrayendo al valor previo de las posiciones del centro del efector final, a razón del factor de aporte definido por los botones *AVANCE* que haya elegido para desplazarle.

La figura anterior señala que esta estimación se realiza dentro de un bloque de función llamado *cinInversa* realizado en texto estructurado por la simplicidad de edición y su similitud con otros entornos de programación en los que usualmente se realiza estos desarrollos matemáticos.

### 12.3.5. ANEXO 3.5: CIN\_INVERSA

Se comprende entonces, que al emplear el bloque para movimientos manuales en coordenadas mundo, serán usadas las posiciones del efector y sobre ellas se aplicará el aporte de avance y el resultado del bloque derivará en las magnitudes en grados o milímetros, según el tipo de articulación, que esta se ha de mover para alcanzar dicha posición, obedeciendo a la resolución del siguiente conjunto de ecuaciones:

$$\theta_2 = \text{atan2}(\sqrt{1 - a^2}, a) \quad (12 - 1)$$

$$\text{Donde: } a = \frac{rx^2 + ry^2 - l1^2 - l2^2}{2 * l1 * l2} \quad (12 - 2)$$

$$\text{Donde: } \varphi = \text{atan2}(td, tc) - \text{atan2}(tb, ta) \quad (12 - 3)$$

$$\text{Siendo: } ta = l1 + l2 * \cos(\theta_2) \quad (12 - 4)$$

$$tb = l2 * \sin(\theta_2) \quad (12 - 5)$$

$$tc = rx \quad (12 - 6)$$

$$td = ry \quad (12 - 7)$$

Este set de ecuaciones fue simulado en repetidas oportunidades con el empleo de Matlab antes de implementarlo en el autómat, siguiendo el principio de la realización de modelos y simularlos antes de aplicarlos sobre el sistema real y así observar de manera segura su comportamiento. En el ejercicio fueron obtenidas las configuraciones mostradas en la figura siguiente, al solicitarle las posiciones más significativas observadas en la ilustración 3-4 que dimensiona el espacio de trabajo del manipulador.

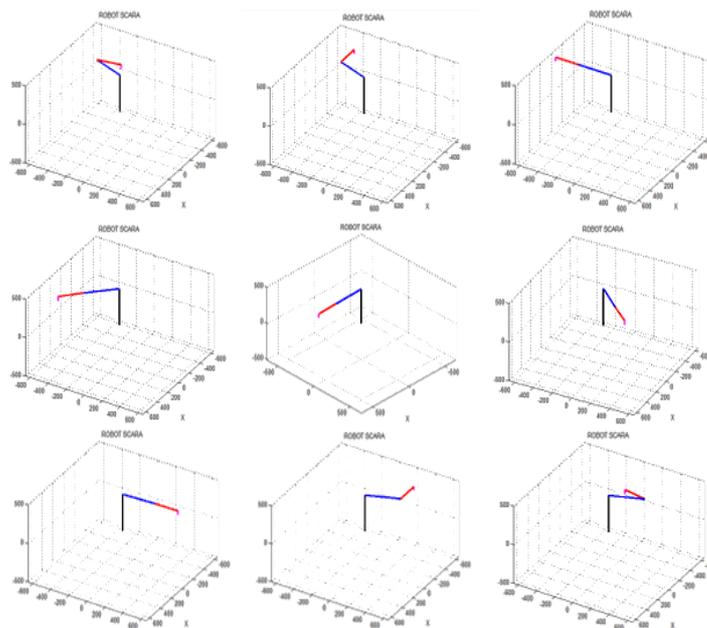


Figura 12-28. Configuraciones articulares a partir de la cinemática inversa.

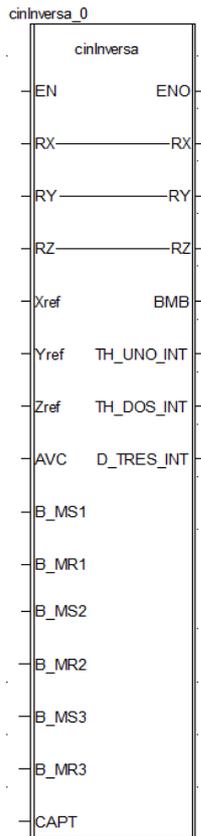
Ya habiendo realizado las simulaciones necesarias se traduce el código a la sección de programa  $F_n\_C\_Inv$  en donde se utilizan las variables de la siguiente tabla para completar el cálculo en el autómata.

Tabla 12-6. Lista de variables del bloque de función cinInvesa.

Lista de Variables del Bloque Derivado de Función cinInversa						
ENTRADAS			PRIVADO		PRIVADO	
#	Nombre	Tipo	Nombre	Tipo	Nombre	Tipo
1	Xref	REAL	flag1	BOOL	RYC	REAL
2	Yref	REAL	PATV	BOOL	h_3	REAL
3	Zref	REAL	PATV2	BOOL	h_3C	REAL
4	AVC	REAL	PATV3	BOOL	k_3	REAL
5	B_MS1	BOOL	PATV4	BOOL	k_3C	REAL
6	B_MR1	BOOL	PATV5	BOOL	h_4	REAL
7	B_MS2	BOOL	PATV6	BOOL	h_4C	REAL
8	B_MR2	BOOL	MDG	BOOL	k_4	REAL
9	B_MS3	BOOL	MDF	BOOL	k_4C	REAL
10	B_MR3	BOOL	MDG2	BOOL	r_3	REAL
11	CAPT	BOOL	MDF2	BOOL	r_4	REAL
			MDG3	BOOL	RADIO_PRINCIPAL	REAL
	<b>ENTRADAS – SALIDAS</b>		MDF3	BOOL	RAD1	
#	<b>Nombre</b>	<b>Tipo</b>	pulso	REAL	RAD2	REAL
1	RX	REAL	A1	REAL	RADIO_NEG	REAL
2	RY	REAL	A2	REAL	RADIO_POS	REAL
3	RZ	REAL	A1C	REAL	BND	INT
			A2C	REAL	BND2	INT
	<b>SALIDAS</b>		SM1	REAL	BND3	INT
#	<b>Nombre</b>	<b>Tipo</b>	SM2	REAL	BND4	INT
4	BMB	BOOL	NMDIV	REAL	Flag	BOOL
5	TH_UNO_INT	INT	M	REAL	theta2sal	REAL
6	TH_DOS_INT	INT	MC	REAL	theta2grad	REAL
7	D_TRES_INT	INT	RST1	REAL	Ta	REAL
			S	REAL	Tb	REAL
			PD1	REAL	Tc	REAL
			AT1	REAL	Td	REAL
			AT2	REAL	OFFSET	REAL
			phi	REAL	D3	REAL
			theta1sal	REAL	RST	REAL
			theta2	REAL	CAL_C	BOOL
			RXC	REAL	D_TRES	REAL

La función es capaz de determinar sin la necesidad de realizar la estimación, si la solución que se obtendría pertenecería o no al espacio de trabajo del robot, indicando visualmente que se ha alcanzado una singularidad. De esta manera se ahorra el recorrer las líneas del código que define las magnitudes, repercutiendo en la disminución de procesamiento computacional innecesario.

Esta función además es el centro de la aplicación de estrategias de control en el robot, de las cuales sobresale la necesidad de implementar un interpolador para la generación de trayectorias lineales y circulares que se ha propuesto como una de las mas significativas mejoras a implementar en futuros estudios desarrollados sobre el SCARA.



```

(*FUNCION PARA LA ESTIMACIÓN DE LA CINEMÁTICA INVERSA DEL SCARA*)
IF (CAPT = TRUE) THEN
  RX := Xref;
  RY := Yref;
  RZ := Zref;
ELSE
  RX := RX;
  RY := RY;
  RZ := RZ;
END_IF;

(*USO DE LOS PULSADORES DE MANEJO [+]/ [-]*)
TRIGGER_0 (CLK := B_MS1(*BOOL*), RISE => MDF (*BOOL*), FALL => PATV (*BOOL*));
TRIGGER_1 (CLK := B_MR1(*BOOL*), RISE => MDG (*BOOL*), FALL => PATV2(*BOOL*));
TRIGGER_2 (CLK := B_MR2(*BOOL*), RISE => MDG2 (*BOOL*), FALL => PATV3(*BOOL*));
TRIGGER_3 (CLK := B_MS2(*BOOL*), RISE => MDF2 (*BOOL*), FALL => PATV4(*BOOL*));
TRIGGER_4 (CLK := B_MS3(*BOOL*), RISE => MDF3 (*BOOL*), FALL => PATV5(*BOOL*));
TRIGGER_5 (CLK := B_MR3(*BOOL*), RISE => MDG3 (*BOOL*), FALL => PATV6(*BOOL*));

IF (MDF = TRUE) THEN
  pulso := 1.0;
  RX := RX + (AVC * pulso);
ELSE IF (MDG = TRUE) THEN
  pulso := -1.0;
  RX := RX + (AVC * pulso);
ELSE
  pulso := 0.0;
END_IF;
END_IF;

IF (MDF2 = TRUE) THEN
  pulso := 1.0;
  RY := RY + (AVC * pulso);
ELSE IF (MDG2 = TRUE) THEN
  pulso := -1.0;
  RY := RY + (AVC * pulso);
ELSE
  pulso := 0.0;
END_IF;
END_IF;

IF (MDF3 = TRUE) THEN
  pulso := 1.0;
  RZ := RZ + (AVC * pulso);
ELSE IF (MDG3 = TRUE) THEN
  pulso := -1.0;
  RZ := RZ + (AVC * pulso);
ELSE
  pulso := 0.0;
END_IF;
END_IF;

(* _____ *)

CAL_C := (PATV6 OR PATV5 OR PATV4 OR PATV3 OR PATV2 OR PATV OR CAL_C)
AND NOT MDF2 AND NOT MDG2 AND NOT MDF AND NOT MDG;

IF (CAL_C = TRUE) THEN
  (* _____ CALCULO DE LAS MAGNITUDES _____ *)
  A1 := 350.0; (*LONGITUD DEL ESLABÓN 1*)
  A2 := 250.0; (*LONGITUD DEL ESLABÓN 2*)
  D3 := 74.0; (*LONGITUD DE LA PINZA*)
  OFFSET := 61.0; (*MAGNITUD DE DIFERENCIA NO ALCANZADA POR LA PINZA*)
  A1C := EXPT_REAL_REAL(A1, 2.0);
  A2C := EXPT_REAL_REAL(A2, 2.0);
  RXC := EXPT_REAL_REAL(RX, 2.0); (*CUADRADO DE COORDENADAS X,Y SOLICITADAS*)
  RYC := EXPT_REAL_REAL(RY, 2.0);

```

Figura 12-29. Bloque cinInversa. Sección de programa Fn\_C\_Inv.

```

FBI_5 (x := RY(*REAL*), y := RX(*REAL*), ang => RST (*REAL*));
ANG_MED := 57.295 * RST;          (*DEFINICIÓN DE ZONAS DEL ESPACIO DE LA TAREA*)
h_3 := -119.71;
h_3C := EXPT_REAL_REAL(h_3, 2.0);
k_3 := 328.89;
k_3C := EXPT_REAL_REAL(k_3, 2.0);
  h_4 := h_3;
  h_4C := EXPT_REAL_REAL(h_4, 2.0);
  k_4 := -k_3;
  k_4C := EXPT_REAL_REAL(k_4, 2.0);
  r_3 := 250.0;
  r_4 := r_3;
SM1 := RXC + RYC;                (*CÁLCULO EN OPERACIONES SIMPLES*)
RADIO_PRINCIPAL := SQRT_REAL(SM1);
RAD1 := RXC + RYC - 2.0*h_3*RX - 2.0*k_3*RY + h_3C + k_3C;
RADIO_POS := SQRT_REAL(RAD1);
RAD2 := RXC + RYC - 2.0*h_4*RX - 2.0*k_4*RY + h_4C + k_4C;
RADIO_NEG := SQRT_REAL(RAD2);
          (*DEFINICIÓN DEL ESPACIO DE LA TAREA, SINO PERTENCE NO HAGO CÁLCULOS*)
IF (ANG_MED > -160.0 AND ANG_MED < -20.0) THEN
  BND := 1;                        (*PERTENECE A LOS ANGULOS NEGATIVOS*)
ELSE
  BND := 0;                        (*PERTENECE A LOS ANGULOS POSITIVOS*)
END_IF;
  IF (RADIO_PRINCIPAL <= 600.0 AND RADIO_PRINCIPAL >= 182.0) THEN
    BND1 := 1;                      (*PERTENECE AL PRINCIPAL*)
  ELSE
    BND1 := 0;                      (*FUERA DEL PRINCIPAL*)
  END_IF;
  IF (ANG_MED > -70.0 AND ANG_MED < -20.0) THEN
    BND2 := 1;                      (*ANGULOS NEGATIVOS MENORES A 90*)
  ELSE IF (ANG_MED > -160.0 AND ANG_MED < -110.0) THEN
    BND2 := 2;                      (*ANGULOS NEGATIVOS MAYORES A 90*)
  ELSE
    BND2 := 0;                      (*OTHERWISE NO PERTENECE A LOS SECUNDARIOS*)
  END_IF;
END_IF;
  IF (RADIO_NEG <= 250.0) THEN
    BND3 := 1;                      (*PERTENECE A PEQUEÑA NEGATIVA*)
  ELSE
    BND3 := 0;                      (*NO PERTENECE A PEQUEÑA NEGATIVA*)
  END_IF;
  IF (RADIO_POS <= 250.0) THEN
    BND4 := 1;                      (*PERTENECE A PEQUEÑA POSITIVA*)
  ELSE
    BND4 := 0;                      (*NO PERTENECE A PEQUEÑA POSITIVA*)
  END_IF;
  IF (BND1 = 1) THEN
    IF (BND = 0) THEN
      flag := 1;                    (*CALCULAR ESTÁ EN ZONA A*)
    ELSE IF (BND2 = 2 AND BND3 = 1) THEN
      flag := 1;                    (*CALCULAR ESTÁ EN B*)
    ELSE IF (BND4 = 1 AND BND2 = 1) THEN
      flag := 1;                    (*CALCULAR ESTÁ EN C*)
    ELSE
      flag := 0;                    (*FUERA DEL ESPACIO DE TRABAJO*)
    END_IF;
  END_IF;
END_IF;
ELSE
  flag := 0;
END_IF;

```

Figura 12-30. Sección de programa Fn\_C\_Inv, segunda parte.

```

IF (flag = 1) THEN                                     (*CÁLCULO DE LAS ARTICULACIONES*)
  SM2 := A1C + A2C;                                   (*PARA UTILIZAR MÁS RAPIDEZ DE PROCESAMIENTO*)
  PD1 := 2.0 * A1 * A2;
  NMDIV := SM1 - SM2;
  M := NMDIV / PD1;                                  (*COSENO DEL ANGULO THETA1*)
  MC := EXPT_REAL_REAL(M, 2.0);
  RST1 := 1.0 - MC;
  S := SQRT_REAL(RST1);                              (* ESTIMADO DE LA ARTICULACION THETA 2*)
  FBI_0 (x := M (*REAL*), y := S (*REAL*), ang => theta2 (*REAL*));
  theta2grad := theta2 * 57.295;
  ta := A1 + a2 * COS_REAL(theta2);                  (* ESTIMADO DE THETA 1*)
  tb := a2 * SIN_REAL(theta2);
  tc := RX;
  td := RY;
  FBI_1 (x := tc(*REAL*), y := td(*REAL*), ang => AT1(*REAL*)); (*CALCULAR THETA UNO*)
  FBI_2 (x := ta(*REAL*), y := tb(*REAL*), ang => AT2(*REAL*));
  phi := 57.295 * (AT1 - AT2);

  IF (phi >= 0.0) THEN
    theta1sal := phi + 110.0;
  END_IF;
  IF (theta2grad >= 0.0) THEN
    theta2sal := theta2grad + 150.0;
  END_IF;
  IF (phi < 0.0) THEN
    theta1sal := 110.0 + phi;
  END_IF;
  IF (theta2grad < 0.0) THEN
    theta2sal := 150.0 + theta2grad;
  END_IF;
  IF (theta1sal < 0.0) THEN
    td := -RY;
    FBI_3 (x := tc(*REAL*), y := td(*REAL*), ang => AT1(*REAL*));
    FBI_4 (x := ta(*REAL*), y := tb(*REAL*), ang => AT2(*REAL*));

    phi := 57.295 * (AT1 - AT2);
    theta2sal := 300.0 - theta2sal;
    theta1sal := phi + 110.0;
    theta1sal := 220.0 - theta1sal;
  END_IF;

  theta1sal(*REAL*) := ABS_REAL (IN := theta1sal(*REAL*));
  theta2sal(*REAL*) := ABS_REAL (IN := theta2sal(*REAL*));
END_IF;

TH_UNO_INT := REAL_TRUNC_INT (theta1sal);
TH_DOS_INT := REAL_TRUNC_INT (theta2sal);
END_IF;

D_TRES := 245.0 - D3 - OFFSET - RZ; (* _____ PARA CALCULAR DELTA TRES _____ *)
D_TRES_INT := REAL_TRUNC_INT (D_TRES);
IF (RZ < 0.0) THEN
  flag1 := 1;
ELSE IF (RZ > 110.0) THEN
  flag1 := 1;
ELSE
  flag1 := 0;
END_IF;
END_IF;
BMB := flag1 OR NOT flag; (* _____ *)

```

Figura 12-31. Sección de programa Fn\_C\_Inv, tercera parte.

### 12.3.6. ANEXO 3.6: FATAN2

Como se puede observar en las ecuaciones 12-1 y 12-3, la estimación de la cinemática inversa, está sustentada en la utilización de la función arcotangente especial *ATAN2*, que al operar permite obtener todas las soluciones posibles de los ángulos formados en la geometría de las configuraciones articulares que el manipulador puede adoptar, para posicionar efectivamente el efector sobre la posición que se le instruye en el plano bidimensional, sobre el cual se desplazan los dos primeros enlaces del robot. El lenguaje Unity Pro en su set de herramientas de cálculo original, cuenta solamente con la función arcotangente ordinaria, que ofrece las soluciones sobre el primer y cuarto cuadrante de la circunferencia. El SCARA define con los dos primeros enlaces, posiciones alcanzables en los cuatro cuadrantes del plano X, Y.

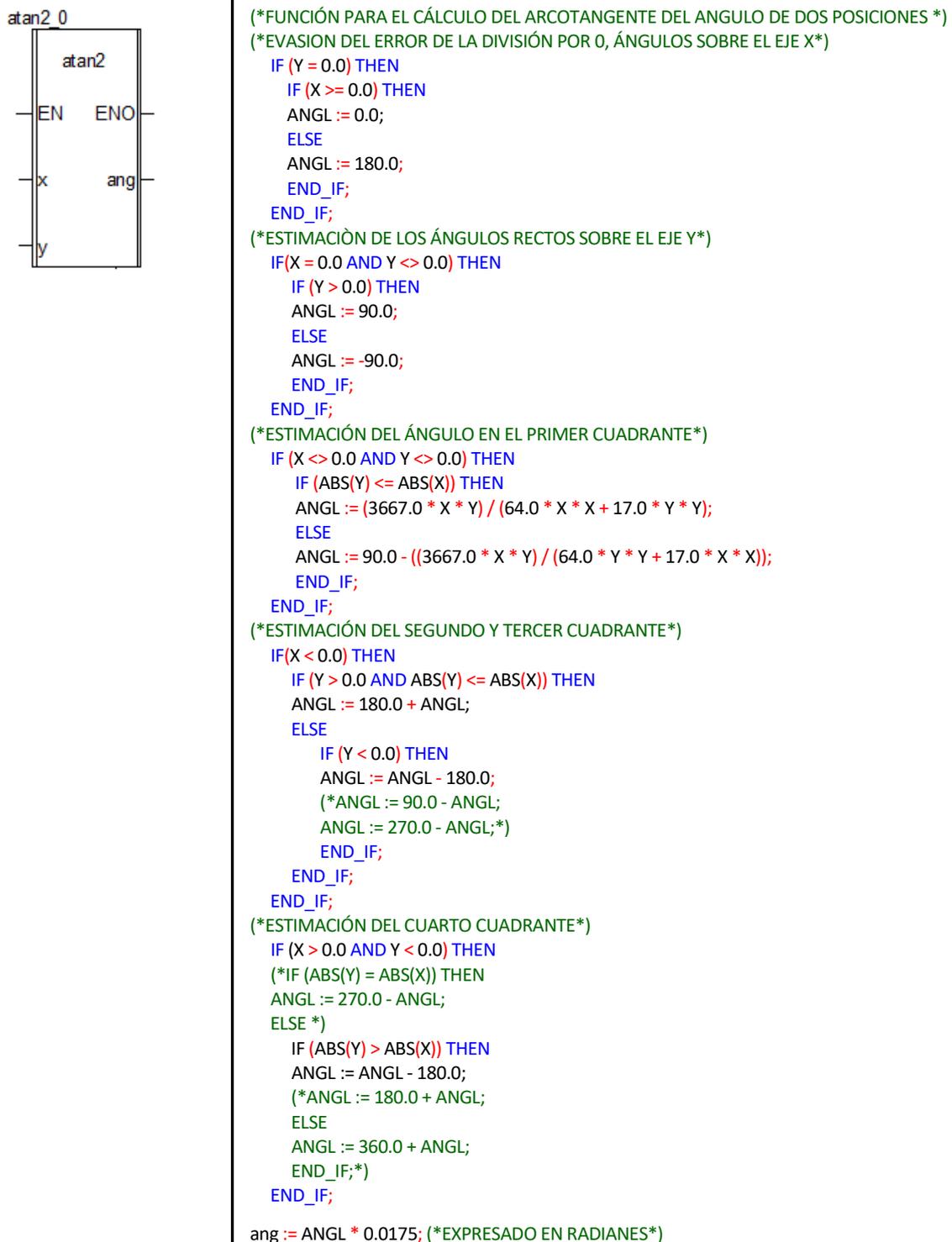


Figura 12-32. Bloque Atan2.Sección de programa FATAN2.

Por tal razón fue necesario crear el bloque de función derivado con el programa de nombre *FATAN2*, diseñado en texto estructurado y se muestra en el código de la figura anterior. En la tabla 12-9 esta función tiene sus variables operativas listadas. El código ha sido basado principalmente en operaciones básicas no demandantes de alta potencialidad de cálculo, siendo utilizado para encontrar las soluciones con la implementación de las instancias *FBI\_0*, *FBI\_1*, *FBI\_2*, *FBI\_3*, y *FBI\_4*, incluidos en la POU ilustrada en las figuras 12-29, 12-30 y 12-31. Esta función cuenta con una precisión de algunas centésimas de grados. Para la función *atan2* cabe acotar que dicho planteamiento inicialmente puede ser implementado para los fines de este proyecto, sin embargo, si se demandare mayores exigencias de prestaciones de servicio preciso al robot, el desarrollo de la función habrá de optimizarse a razón de la precisión esperada de cálculo

Tabla 12-7. Lista de variables del bloque de función *Atan2*.

Lista de Variables del Bloque Derivado de Función ATAN2							
Entradas			Salidas			Privado	
#	Nombre	Tipo	#	Nombre	Tipo	Nombre	Tipo
1	X	REAL	1	ang	REAL	ANGL	REAL
2	Y	REAL	2				

La sección de programa *S\_MANUAL* que se ha dedicado a la coordinación de los movimientos del robot desde la estación de trabajo, en las ultimas líneas del diagrama escalera que la controla, dispone de la actuación sobre cada uno de los motores de la unidad.

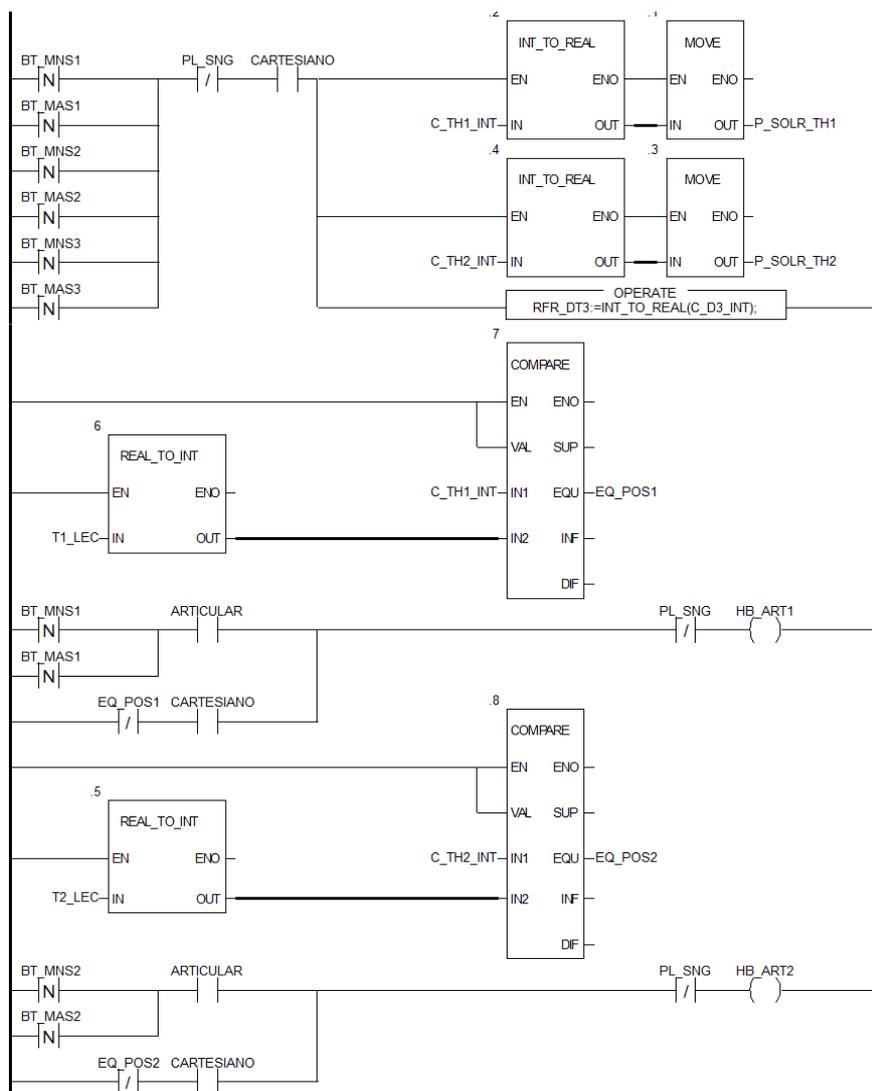


Figura 12-33. Sección del programa *S\_MAN*, tercera parte.

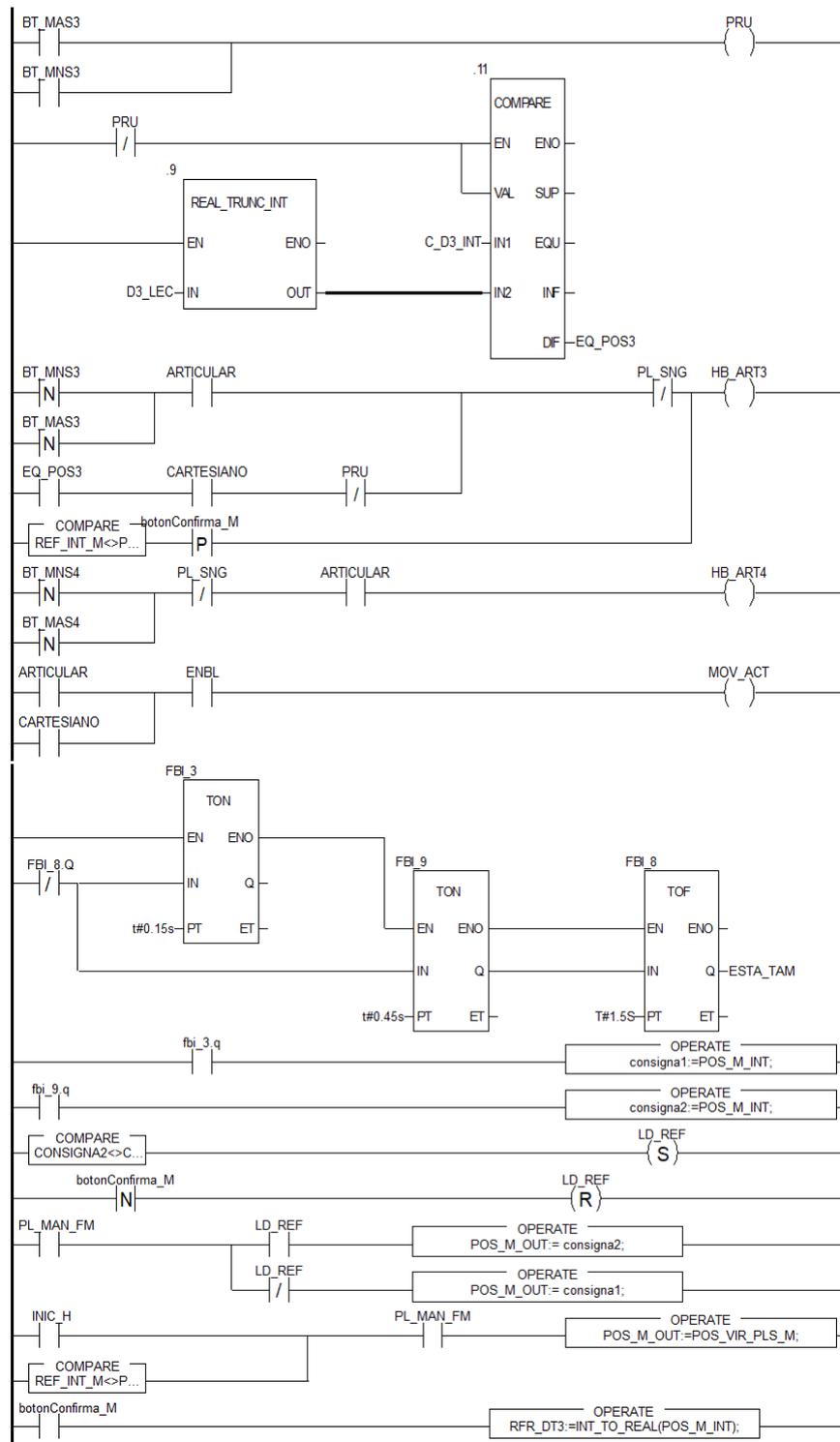


Figura 12-34. Sección del programa S\_MAN, cuarta parte.

Siguiendo el mismo principio de actuación que ha sido concebido desde le inicio, comparar continuamente la posición actual del enlace contra la posición fijada como destino. De esta manera inicia y detiene la marcha de los elementos motrices de manera autónoma, solamente ante la existencia de la diferencia existente.

Finalmente, de la misma manera que se puede accionar los motores asociados a los accionamientos de los eslabones, la función de movimiento manual es capaz de actuar sobre el efector adosado al robot, haciendo que el usuario tenga una herramienta para la realización de movimientos útiles en las tareas que el SRX 4CH ha de realizar de manera autónoma y que serán enmarcadas dentro de una tarea principal de recogida, traslado y colocación de piezas Es entonces que la función PINZA, incluida en la sección del programa de la figura siguiente; tiene entonces el propósito de utilizar el efector final mediante el arreglo neumático descrito en la sección 6.3 de este documento.

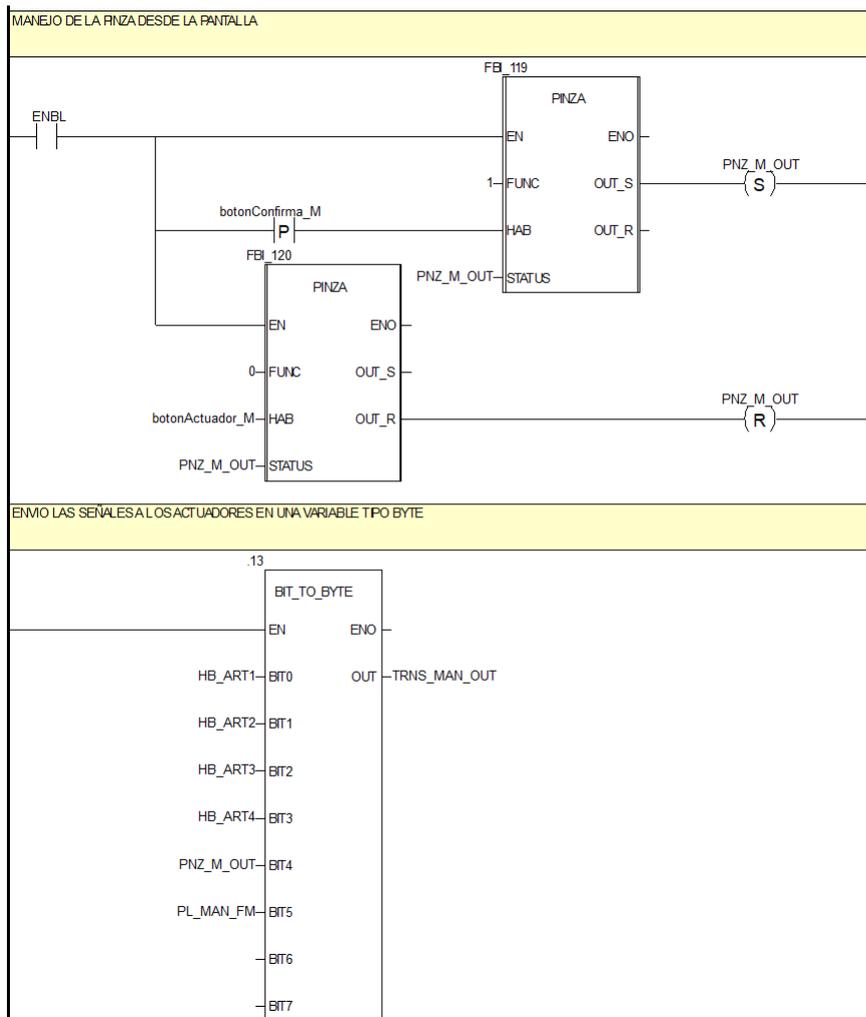


Figura 12-35. Sección del programa S\_MAN, quinta parte.

### 12.3.7. ANEXO 3.7: FUNCIÓN PINZA

La función *PINZA* es empleada para maniobrar sobre el efector que se encuentra dispuesto en el extremo del robot, con el cual el SCARA es capaz de asir piezas. Se ejecuta mediante el empleo de pinzas de dos o tres dedos, ventosas u otro accionamiento neumático de doble o simple efecto que facilite la sujeción de objetos. Finalmente, gracias a la tabla desplegada a continuación, se ha hecho posible agrupar las variables que se utilizan en la sección de programa que la maneja.

Tabla 12-8. Lista de variables del bloque de función PINZA.

Lista de Variables del Bloque Derivado de Función PINZA							
ENTRADAS			SALIDAS			PRIVADO	
#	Nombre	Tipo	#	Nombre	Tipo	Nombre	Tipo
1	FUNC	BOOL	1	OUT_S	BOOL		
2	HAB	BOOL	2	OUT_R	BOOL		
3	STATUS	BOOL					

Como se aprecia en la figura 12-36, la función posee una sección de programa en escalera de nombre *ACTUADORPINZA*, que permite por medio de la puesta en alto de la entrada de habilitación *HAB*, conmutar el arreglo electroneumático. La apertura está condicionada a que la entrada *FUNC* sea definida *FALSE* y el cierre

se realiza definiéndola *TRUE*.

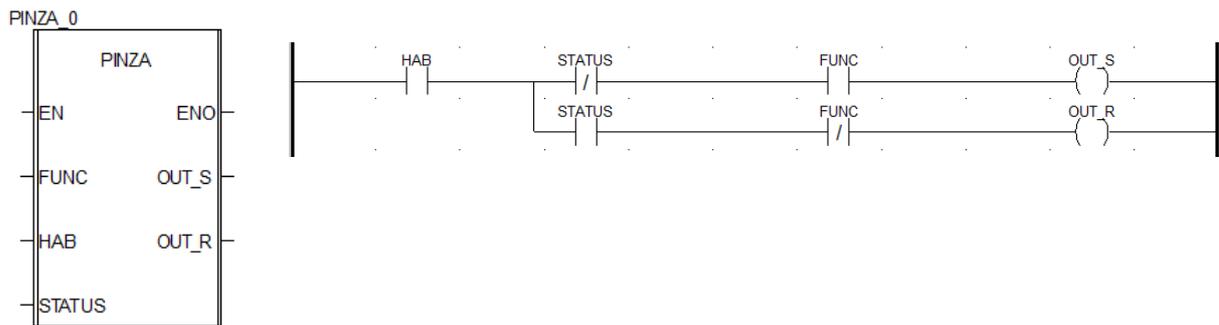


Figura 12-36. Bloque PINZA. Sección de programa ACTUADORPINZA.

La entrada *STATUS* se emplea para conocer si la garra se encuentra abierta o cerrada, de esta manera discriminar si el próximo movimiento colocará en alto a la salida *OUT\_S* para hacer el set de la electroválvula, sujetando así el objeto, o *OUT\_R* para rearmarla, lo que derivaría en la liberación de la pieza.

### 12.4. ANEXO 4: PROGRAMA PRINCIPAL, REGISTRO DE POSICIONES

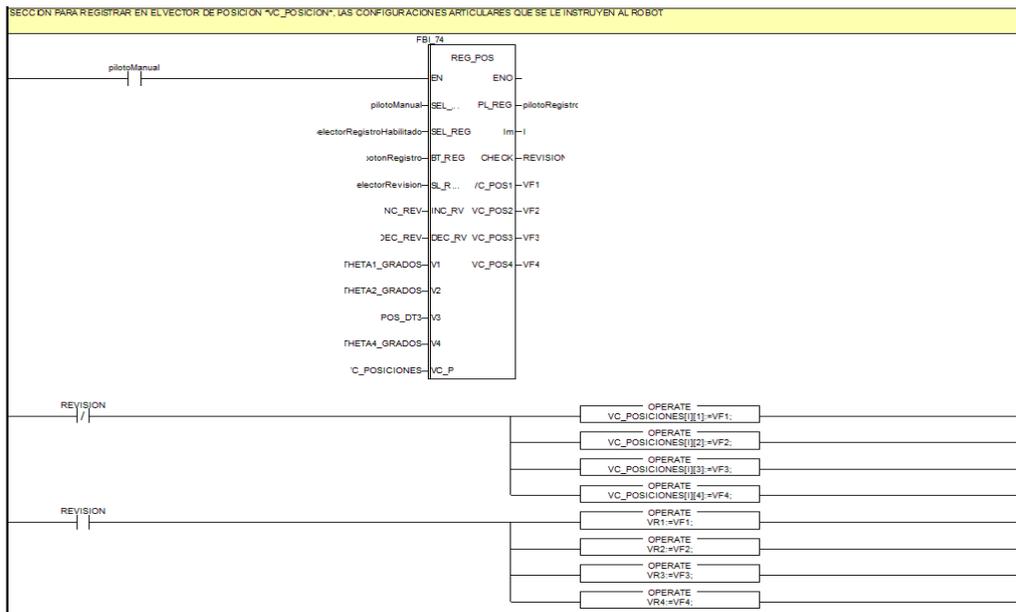


Figura 12-37 . Sección del Programa Principal, Registro de Posiciones.

En la figura anterior que fue tomada de la sección del programa principal, se muestra cómo se han vinculado las variables del programa principal a las entradas y salidas del bloque de función que han sido listadas en la tabla 12-9. El resultado es una función que contiene un proceso suficiente para completar el registro de configuraciones articulares a adoptar en el modo de operación automática del SRX 4CH.

Tabla 12-9. Lista de variables del bloque de función REG\_POS.

Lista de Variables del Bloque Derivado de Función REG_POS							
ENTRADAS			SALIDAS			PRIVADO	
#	Nombre	Tipo	#	Nombre	Tipo	Nombre	Tipo
1	SL_MOD_MAN	BOOL	1	PL_REG	BOOL	AUXILIAR	BOOL
2	SEL_REG	BOOL	2	Im	INT	PL_MANUAL	BOOL
3	BT_REG	BOOL	3	CHECK	BOOL	BLOQUEO	BOOL
4	SEL_REVISION	BOOL	4	VC_POS1	DINT	Jm	INT
5	INC_RV	BOOL	5	VC_POS2	DINT		
6	DEC_RV	BOOL	6	VC_POS3	DINT		
7	V1	DINT	7	VC_POS4	DINT		
8	V2	DINT					
9	V3	INT					
10	V4	DINT					
11	VC_P	ARRAY OF DINT					

Empleándole se puede construir una trayectoria a seguir con el SCARA, en otras palabras, usando el bloque de función se registran posiciones dentro de un arreglo matricial que se ha denominado *VC\_POSICIONES*, en el que se agrupan las cuatro variables articulares del brazo manipulador y este arreglo se ha definido para almacenar hasta 25 posiciones. Siendo editable por el usuario desde la pestaña *Variables Elementales* del árbol del proyecto el arreglo y el tope de avance del puntero (*+++*) en la sección del programa *REGISTRO\_POSICION*, además satisfaciendo que las variables que recibe sean congruentes y de esta manera la matriz podrá contener más configuraciones articulares a adoptar por el SCARA si la secuencia de movimientos que está configurando así lo demandase.

La posición inicial del vector está definida como [0,0,0,0], señalándole automáticamente como la posición cero (POS\_0) del robot y esta es coincidente con la posición de HOME. A partir del siguiente renglón del arreglo, es

decir la siguiente posición, se comienzan a registrar las deseadas por usuario con el uso del botón *GUARDAR* de la pantalla de operador *COMANDO*, después de haber conmutado el selector dos posiciones *REGISTRO* a encendido. Este funciona como permisivo de la función, que de la misma manera está habilitada solo en el modo manual. Estos elementos de maniobra mencionados han sido mostrados en la figura 12-7 y 12-8 en la descripción de la pantalla de explotación de recursos.

Mediante el uso del botón *REVISION* es posible realizar por medio de la misma función la visualización del contenido del arreglo almacenado, con la intención de corroborar que el contenido es el patrón necesario para la tarea a ejecutar, para ello en la pantalla se ha incluido un indicador de la posición que se está observando en las celdas numéricas.

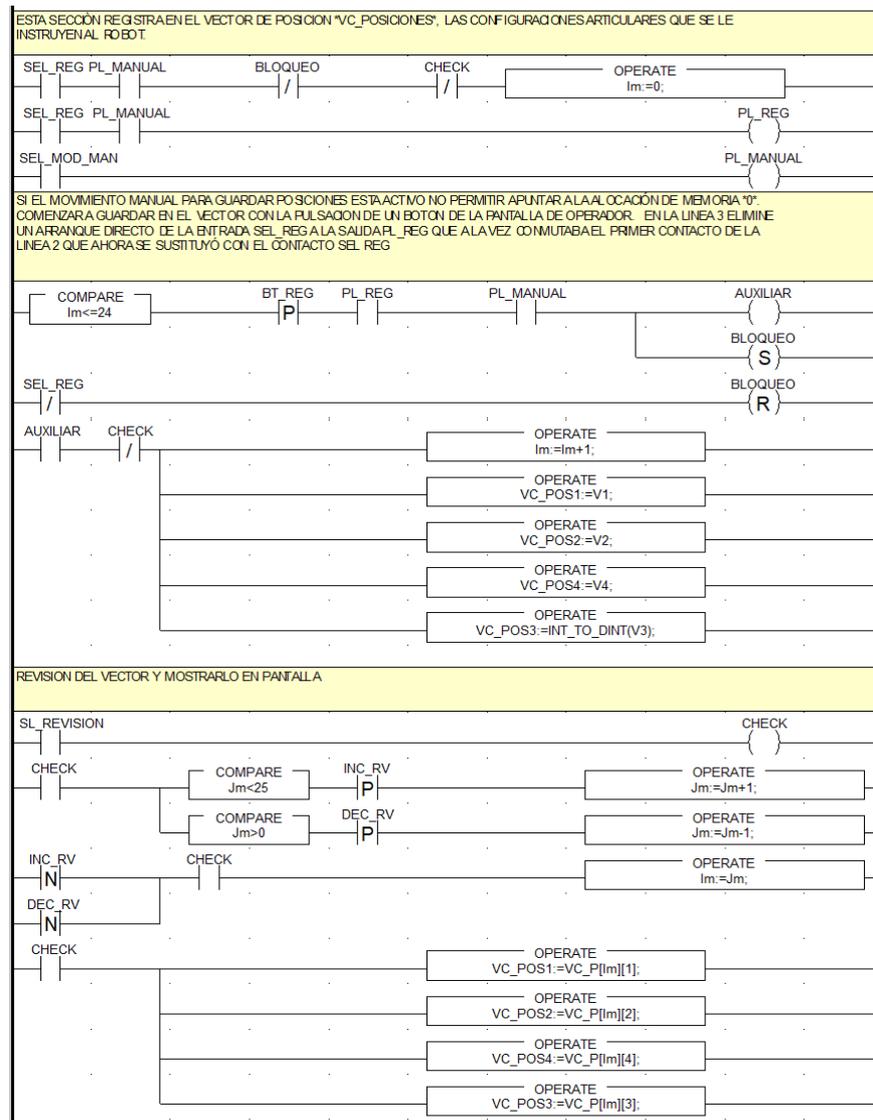


Figura 12-38. Bloque de REG\_POS y Sección del programa GUARDAPOSICION.

Es importante realizar un planteamiento previo para el registro de posiciones, de esta manera se logra simplificar el guardado de todas desde la pantalla de operador, que actúa sobre la sección de programa de la figura anterior, en un solo procedimiento. De lo contrario será necesario reiniciar el proceso nuevamente, hacer la edición de las posiciones erróneas una a una a través del proceso de revisión (posicionarse en la posición del vector anterior editar y guardar) o editarlas con la inclusión de una *tabla de edición de variables* en el árbol del proyecto para modificar sobre las celdas del arreglo de manera directa.

En la ilustración 12-34, en la cual es mostrada la programación del bloque de función, se indica que el bloque toma la posición a guardar en la memoria directamente del valor que se ingresa a través de las entradas V1 a V4 correspondientes a las posiciones de brazo. Entonces, editar un registro está ligado a llevar al robot a adoptar la pose y luego guardarla en el arreglo matricial. Por ello se insiste en considerar adecuadamente la trayectoria previamente.

## 12.5. ANEXO 5: PROGRAMA PRINCIPAL, ACTUADORES

Los actuadores del SONY SRX 4CH en su totalidad, son controlados desde el autómata con el empleo de una única función derivada de nombre *ACTUADORES*, la cual es mostrada en la figura 12-39 que se ilustra a continuación. Al crearse esta función se consideró que debía poseer la habilidad de adaptar su operación a cualquiera de las acciones que el brazo realizase, además habría de incluir los elementos necesarios de seguridad y de reconocimiento del entorno, para evitar que en la ejecución de movimientos se incurriese en daños de los elementos de la unidad.

De la misma manera que se ha hecho en las secciones previas, el bloque derivado de función ha sido vinculado al programa principal con la relación de entradas y salidas que se lista en la tabla 12-10 que sigue.

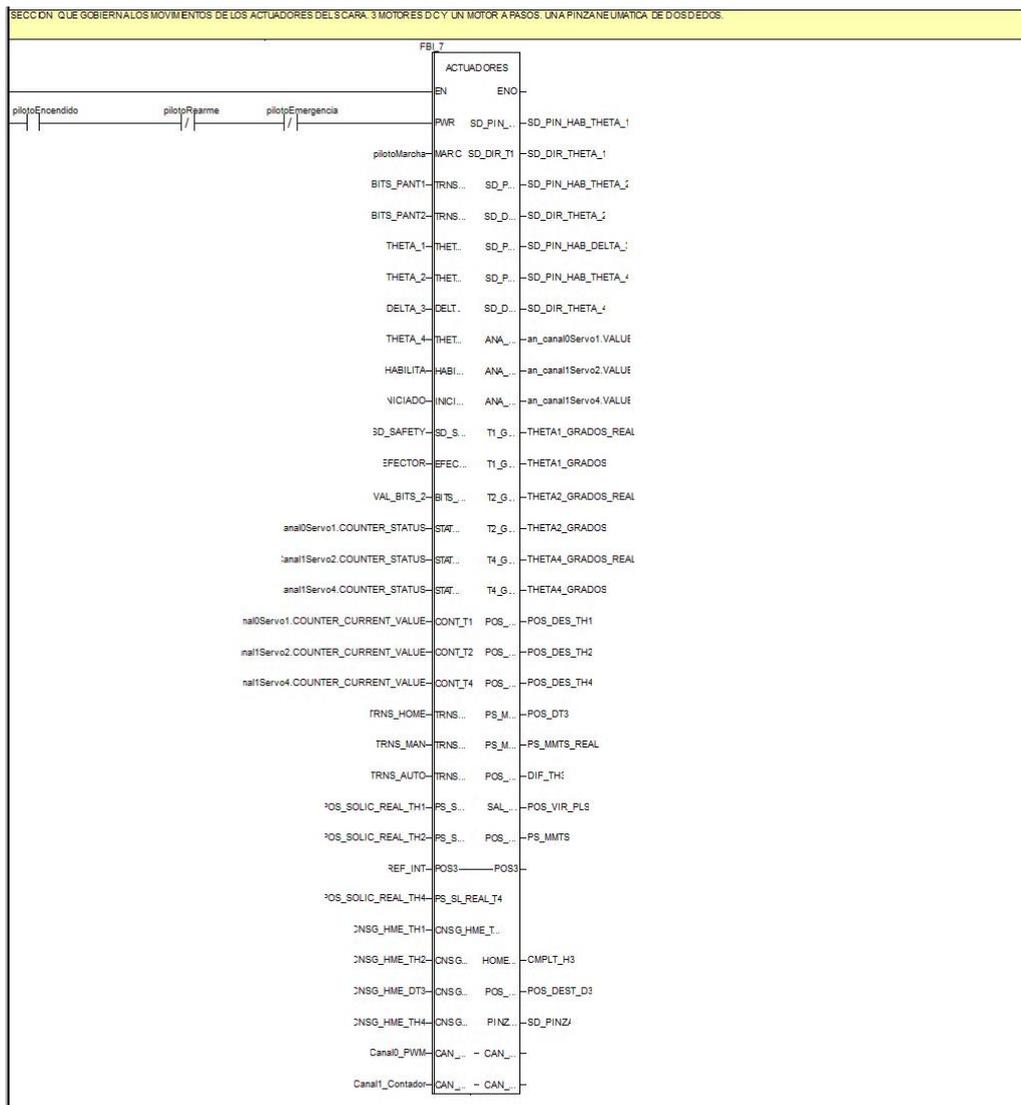


Figura 12-39 . Sección del Programa Principal, Actuadores.

El programa de la función de nombre *MUEVEACTUADOR*, hace uso de los bits que le son enviados desde el resto de funciones que se despliegan en el programa principal. Es de este modo que la unidad de potencia obedece a las instrucciones de la pantalla de operador con tres variables del tipo *WORD* y de las funciones *MOV\_MAN*, *MOV\_AUTO* y *FCN\_HOME*, cada una con una variable del tipo *BYTE*. Esto se aprecia en la figura 12-40 desplegada en este apartado.

El resto del programa está únicamente dedicado a invocar los bloques de función dedicados específicamente al gobierno de los tres motores de corriente continua que gobiernan las articulaciones rotacionales y al motor a pasos que gobierna la prismática.

Tabla 12-10. Lista de variables del bloque de función ACTUADORES.

<b>Lista de Variables del Bloque Derivado de Función Actuadores.</b>							
<b>ENTRADAS</b>			<b>ENTRADAS – SALIDAS</b>			<b>PRIVADO</b>	
#	Nombre	Tipo	#	Nombre	Tipo	Nombre	Tipo
1	PWR	BOOL	1	SD_PIN_HAB_T1	BOOL	ACNM_D3_FN	BOOL
2	MARCA	BOOL	2	SD_DIR_T1	BOOL	DETNC_D3_FN	BOOL
3	TRNS_PANT1_FN	WORD	3	SD_PIN_HAB_T2	BOOL	MOT_AV1_FN	BOOL
4	TRNS_PANT2_FN	WORD	4	SD_DIR_T2	BOOL	MOT_AV2_FN	BOOL
5	THETA_1_FN	BOOL	5	SD_PIN_HAB_D3	BOOL	MOT_AV3_FN	BOOL
6	THETA_2_FN	BOOL	6	SD_PIN_HAB_T4	BOOL	MOT_AV4_FN	BOOL
7	DELTA_3_FN	BOOL	7	SD_DIR_T4	BOOL	AVANCE_ART1_FN	BOOL
8	THETA_4_FN	BOOL	8	ANA_CAN_1	INT	AVANCE_ART2_FN	BOOL
9	HABILITA_FN	BOOL	9	ANA_CAN_1	INT	AVANCE_ART3_FN	BOOL
10	INICIADO_FN	BOOL	10	ANA_CAN_1	INT	AVANCE_ART4_FN	BOOL
11	SD_SAFETY_FN	BOOL	11	T1_GR_REAL_FN	REAL	ACNM_T4_FN	BOOL
12	EFECTOR_FN	BOOL	12	T1_GR_FN	DINT	U_THETA_1	REAL
13	BITS_ACTDRS	WORD	13	T2_GR_REAL_FN	REAL	U_THETA_2	REAL
14	STATUS_T1	INT	14	T2_GR_FN	DINT	U_THETA_4	REAL
15	STATUS_T2	INT	15	T4_GR_REAL_FN	REAL	SAL_MOT1_FN	INT
16	STATUS_T1	INT	16	T4_GR_FN	DINT	SAL_MOT2_FN	INT
17	CONT_T1	DINT	17	POS_DES_T1_FN	INT	SAL_MOT3_FN	INT
18	CONT_T2	DINT	18	POS_DES_T2_FN	INT	SAL_MOT4_FN	INT
19	CONT_T4	DINT	19	POS_DES_T4_FN	INT	PNZ_A_FN	BOOL
20	TRNS_HOME_FN	BYTE	20	POS_MM_D3	INT	PNZ_H_FN	BOOL
21	TRNS_MAN_FN	BYTE	21	POS_MM_RL_D3	REAL	PNZ_M_FN	BOOL
22	TRNS_AUTO_FN	BYTE	22	POS_DES_D3_FN	INT	piloto_Marcha_FN	BOOL
23	PS_SL_REAL_T1	REAL	23	SAL_PLS_FN	INT	piloto_Auto_FN	BOOL
24	PS_SL_REAL_T2	REAL	24	POS_DT3_FN	DINT	piloto_Manual_FN	BOOL
26	PS_SL_REAL_T4	REAL	28	HOME_TH3	BOOL	piloto_T1LI_FN	BOOL
27	CNSG_HME_TH1_FN	INT	29	POS_DEST_D3_FN	BOOL	piloto_T1LS_FN	BOOL
28	CNSG_HME_TH2_FN	INT	30	PINZA_FN	BOOL	piloto_T2LI_FN	BOOL
29	CNSG_HME_DT3_FN	REAL				piloto_T2LS_FN	BOOL
30	CNSG_HME_TH4_FN	INT				piloto_D3LI_FN	BOOL
						piloto_D3LS_FN	BOOL
						piloto_T4H_FN	BOOL
						piloto_D3H_FN	BOOL
						VISLZ	BOOL
						HOM_M	BOOL
						piloto_EMG_FN	BOOL
						PR_AUTO	BOOL
						botonParoRepPant_FN	BOOL
						botonActuador_FN	BOOL
						botonConfirma_FN	BOOL
<b>ENTRADAS – SALIDAS</b>			<b>PRIVADO</b>				
Nombre	Tipo		POS_HOM	INT			
POS3	INT		POS_SAL	INT			
CAN_PWM	T_UNSIGN_CPT_BMX		POS_REF	INT			
CAN_CONTAD	T_SIGN_CPT_BMX		RF_RL_D3	REAL			
			RF_RL_DT3	REAL			
			ACNM_T1_FN	BOOL			
			ACNM_T2_FN	BOOL			

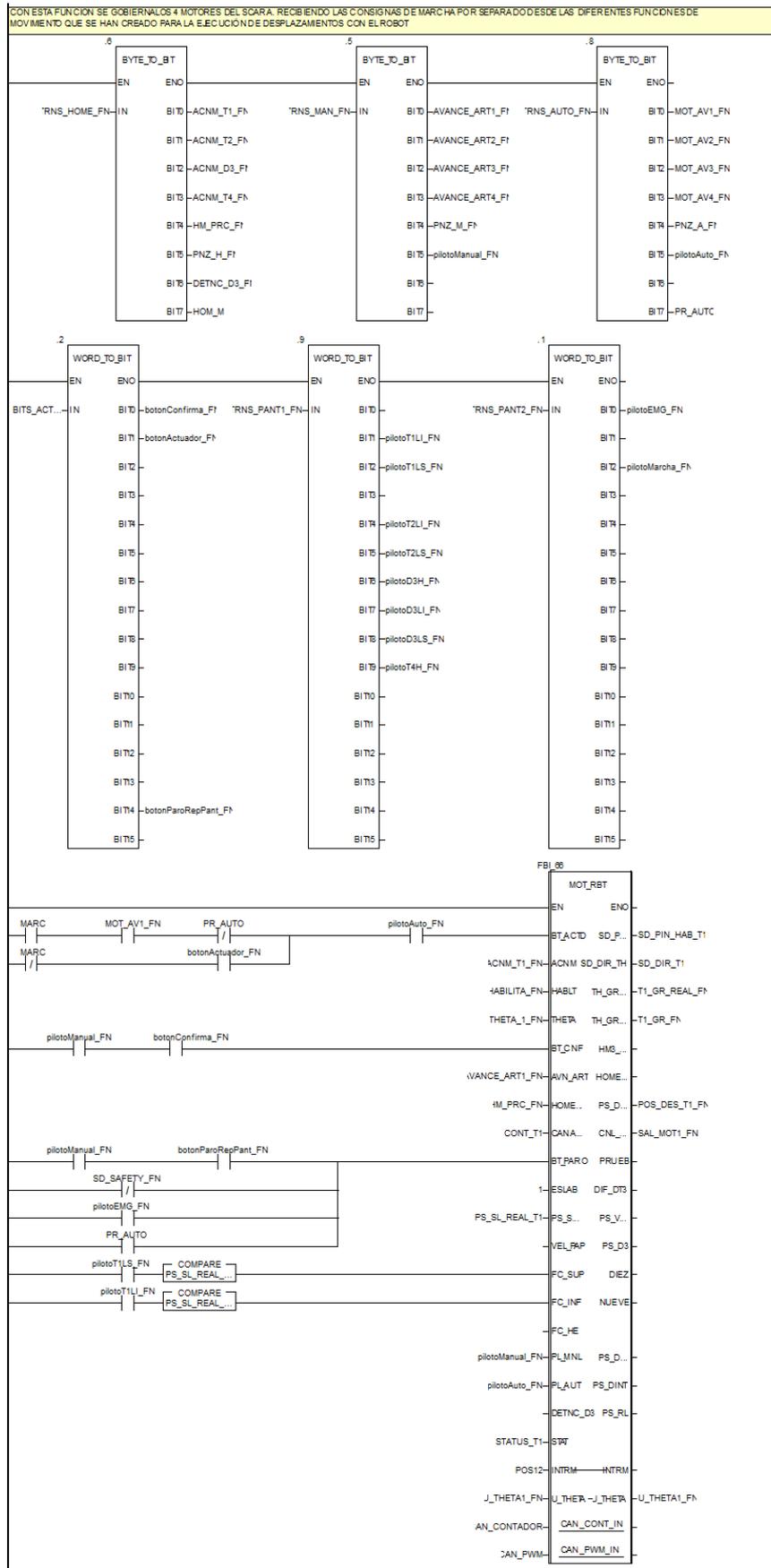


Figura 12-40. Sección de programa ACTUADORES.

En el manejo de los motores se emplean dos métodos de control, con el fin de optimizar el posicionamiento final, la respuesta en regulación de la posición y el seguimiento de la consigna de velocidad.

### 12.5.1. ANEXO 5.1: CONTROL POSICIÓN Y VELOCIDAD

El control de posicionamiento y la rapidez de las cuatro articulaciones se realiza actuando sobre los motores que accionan el desplazamiento del eslabón. En la sección de nombre *MUEVEACTUADOR* perteneciente al programa de la función, se encuentran incluidos los bloques para cada uno de los cuatro motores que se encuentran en el manipulador dada su morfología, de donde se ha tomado la figura siguiente para mostrar cómo se han vinculado las variables del programa principal a las entradas y salidas que se han listado en el sexto apartado de esta memoria, logrando de esta manera que la función pueda actuar sobre la articulación THETA 2.

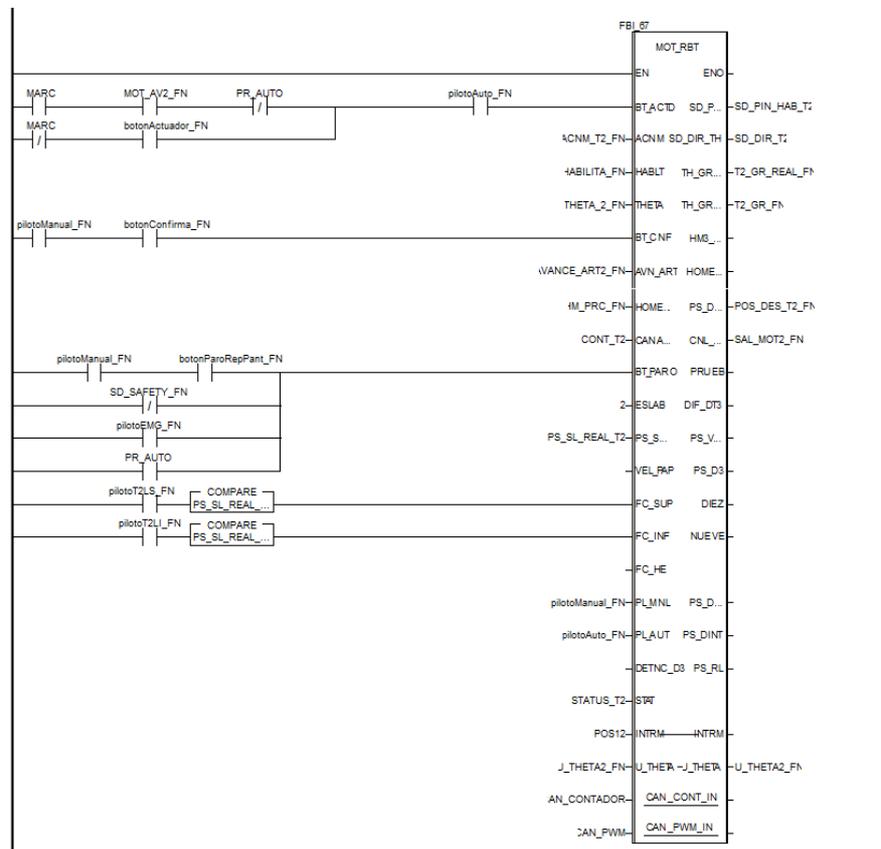


Figura 12-41 . Sección de programa ACTUADORES, segunda parte.

Sin embargo, existen diferencias para cada motor asociadas no solo a las características operacionales que posee (stepper o brushed), sino también a las particularidades mecánicas del elemento que gobierna.

Es este el motivo de incluir en el programa interno del bloque de función, además de los necesarios elementos de arranque y parada, elementos de seguridad y dimensionamiento de los desplazamientos inherentes a su actividad, también la función debía poseer la habilidad de adaptarse a la operación de cualquiera de los 4 accionamientos en donde el programador pudiera emplearle.

En la ilustración 12-42 que sigue, se logra apreciar el diagrama en escalera del bloque *MOT\_RBT*. En ella se observa que habiendo comprobado que el registro de estado principal *COUNTER\_STATUS* de los módulos EHC-0200 se encuentra habilitado, se podrá iniciar la operación, comenzando con la discriminación del eslabón al que pertenece el motor que el bloque ha de controlar por medio de la entrada denominada *ESLAB* que es expresada en enteros.

Esta tarea permite además diferenciar el modo de operación del motor, permite también realizar la carga de parámetros individuales como la conversión a unidades de ingeniería de los movimientos, la parametrización de los controladores proporcional + integral + derivativo que se emplean para el control de posición y fijar las magnitudes a alcanzar en el perfil de velocidad con la que el eslabón se le permite moverse según la distancia que ha de recorrer.

Lo que se ha pretendido es ahorrarle al usuario – programador la necesidad de estar configurando estos factores cuando haga empleo del robot. Así ha de centrar su actividad en el aprovechamiento del recurso y no en las tareas de ajuste y gobierno del SCARA.

### 12.5.2. ANEXO.5.2. CONTROL POSICIÓN Y VELOCIDAD DE MOTOR DC

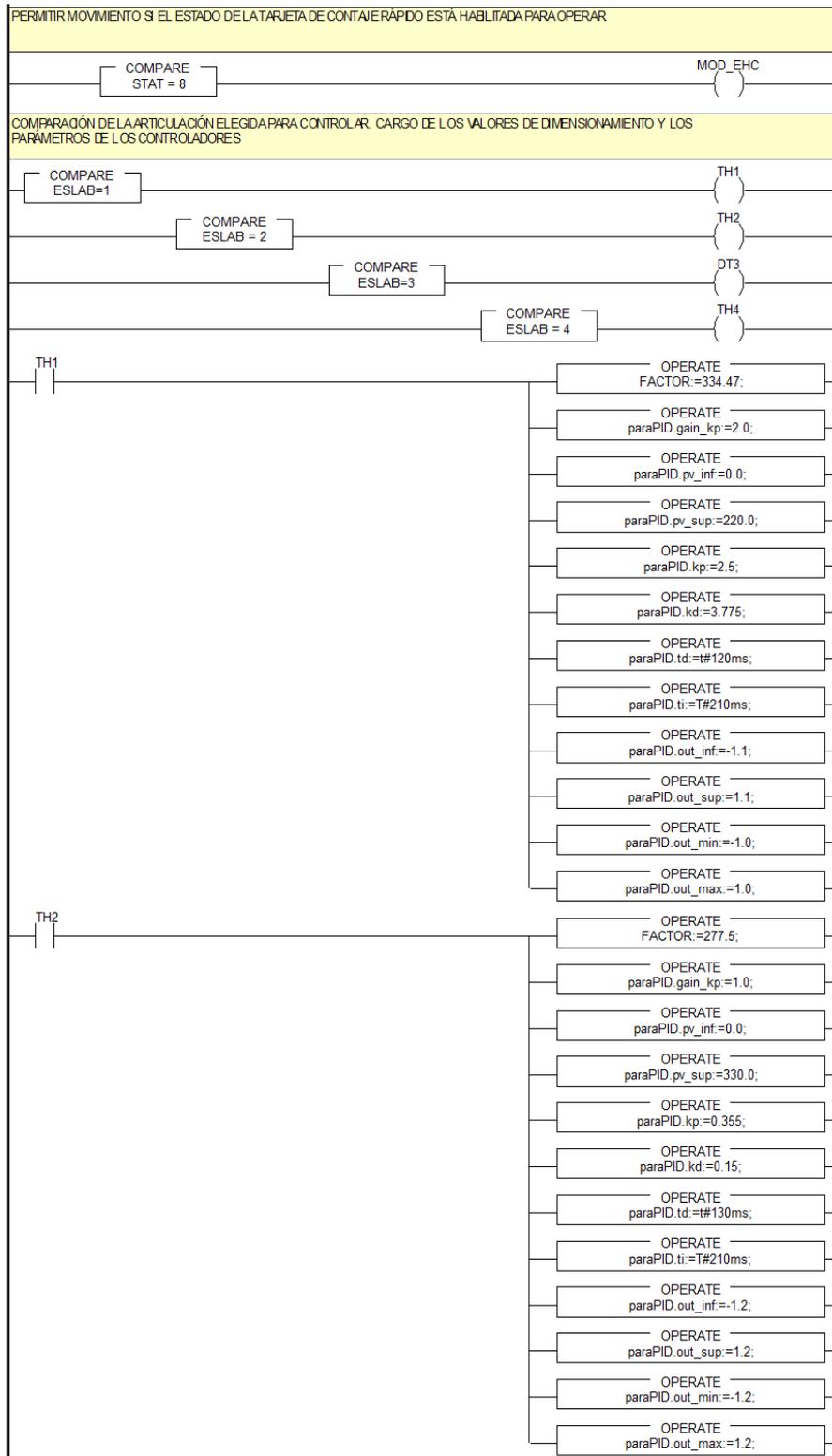


Figura 12-42.Sección de programa MOT\_SCARA.

Al bloque de función *MOT\_RBT*, le han sido vinculadas las variables del programa de la función *ACTUADORES*, siguiendo la relación de entradas y salidas, que se encuentran listas seguidamente en la tabla 12-11.



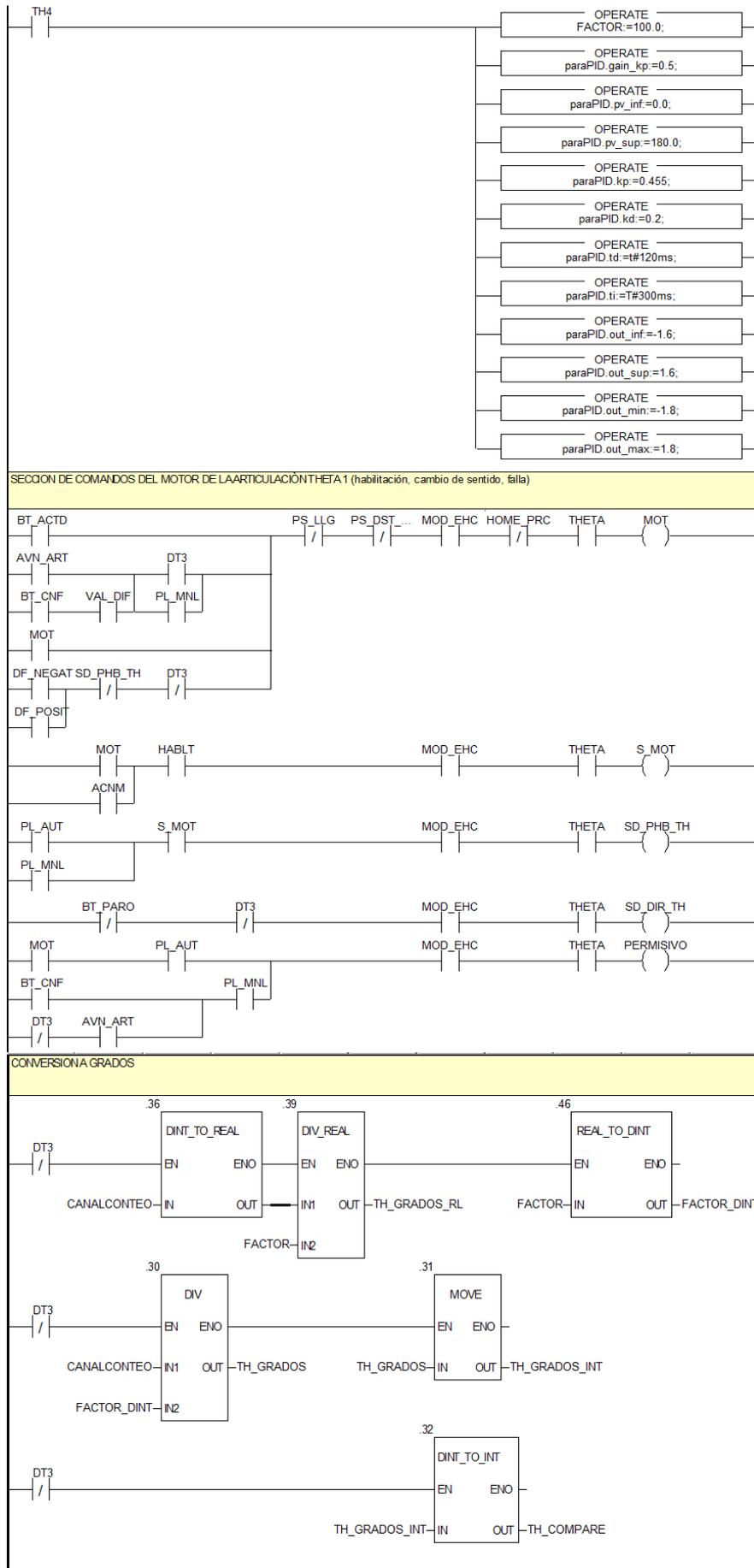


Figura 12-43. Sección de programa MOT\_SCARA, segunda parte.

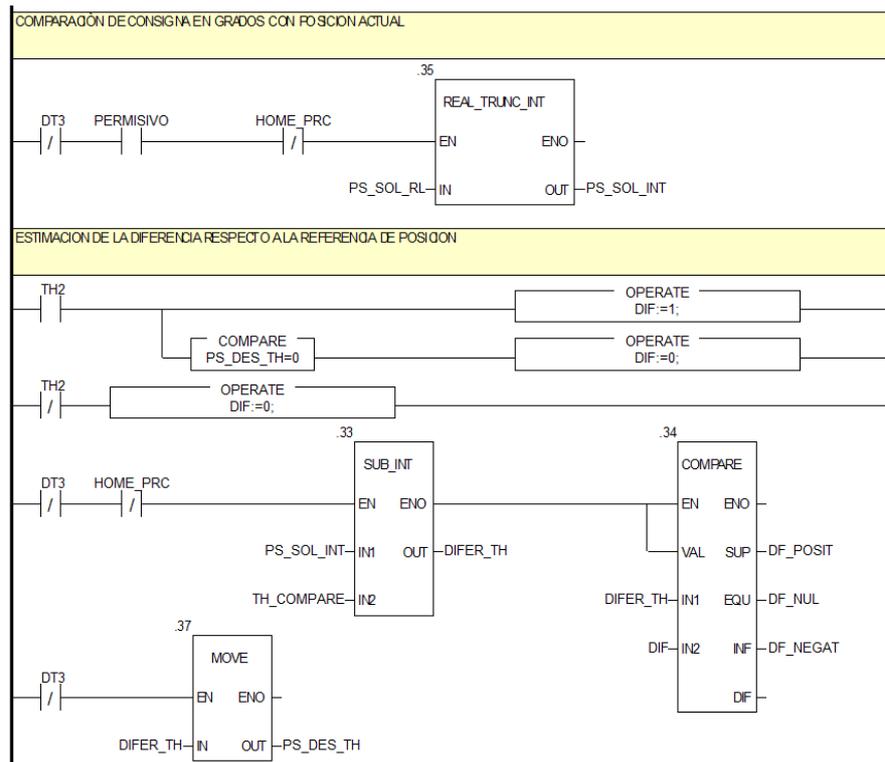


Figura 12-44. Sección de programa MOT\_SCARA, tercera parte.

La figura 12-44 muestra el control del avance de las articulaciones de naturaleza rotacional. Controlándoles desde el driver amplificador, fabricado por Advance Motion Controls, mediante modulación de ancho de pulsos sobre cada uno de los motores tipo brush (con escobillas) dispuestos en el robot.

En primer lugar, lo que realiza la rutina es dimensionar el recorrido, midiéndolo por medio del codificador incremental. Esta magnitud se adecúa matemáticamente para que referencie de manera acorde el movimiento y lo ofrezca en unidades que interprete el usuario al tomarlas desde la interfaz de interacción que emplea para la configuración del SCARA, (se han definido en grados).

Para ello se hizo necesario adaptar el tipo de variables ofrecidas por los diferentes elementos que intervienen en el servomecanismo. Por ejemplo, la tarjeta de contaje entrega la medición del codificador como un número *DINT* (entero doble), precisando la conversión a *REAL* para operar matemáticamente sobre la medición.

El principio de actuación de la unidad de organización de programas está centrado en determinar la ubicación actual del eslabón, luego ha de compararla con la que se desea alcanzar y cuando la variación se hace cero (0), habrá de detener el motor, quedando la articulación posicionada correctamente sobre el punto en el espacio que se le ha asignado como referencia.

Gracias a la acción de un controlador proporcional + integral + derivativo, se garantiza que en el permanente no exista variación de la salida de tensión, anulando el error existente contra la referencia de posición. El controlador ha dado excelentes resultados de posicionamiento, posee la limitación de efecto windup para disminuir la susceptibilidad a los sobrepasos de la referencia por efecto de la inercia del eslabón en el desplazamiento. Sin embargo, si se desea emplear el SCARA en otro tipo de actividades que demanden de un más alto performance que el objetivo de traslado de piezas, se han de implementar estrategias más complejas para atacar el efecto de las fricciones al inicio del movimiento y otros efectos que afectan el desempeño del robot.

Esta es la razón de encontrar un enfoque inicial simple en la supresión de los efectos que se han mencionado anteriormente. Planteando avanzar mediante el trazo de un perfil de velocidad, que se ajusta a la magnitud de la referencia angular. Que se estima a partir de la diferencia de distancia desde el punto donde se ha de iniciar el movimiento hasta el punto final definido en la consigna de posición. De este modo, el seguimiento del perfil habría de conducir al servomotor a realizar un arranque suave, incrementaría la velocidad en el tramo central y posteriormente descendería la velocidad al acercarse a la consigna de posición, con la pretención de disminuir los sobrepasos y las oscilaciones sobre la referencia.

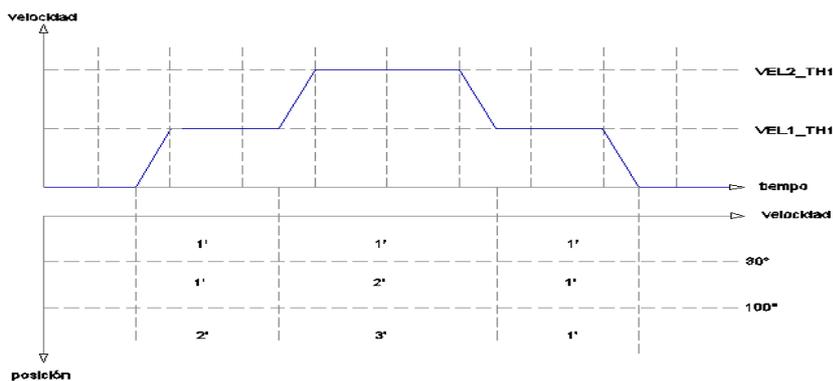


Figura 12-45. Perfil de velocidade a seguir por el motor.

La ilustración previa detalla cómo se seleccionan los cambios de referencia, por medio de la división del recorrido en tres tramos, donde la simetría no es obligatoria. Asimismo, las magnitudes son parametrizables para ajustar adecuadamente el desempeño. El primero de los tramos se emplea para la aceleración inicial, con la intención de suavizarla, el tramo central se permite ir a alta velocidad si las distancias a recorrer superan un arco mayor a 110°, y el tercer tramo está definido para el frenado dinámico con una marcha lenta, y como ya se ha mencionado el planteamiento es no permitir grandes sobrepasos producto de la inercia del eslabón, estos ajustes lo define la variable *VEL1\_TH* cuando se avanza en sentido positivo y *VEL2\_TH* en el negativo. Caso contrario se presenta cuando el recorrido angular es menor a 50°, frente a este evento se mantendrá la velocidad constante durante el recorrido total.

Es en entonces necesario para seguir el planteamiento del perfil de velocidad, particionar el espacio de trabajo articular en regiones o tramos como indica la curva. El diagrama mostrado en la figura dispuesta a continuación incluye líneas o redes de programación para definirlo. Ubica el eslabón cuando se genera la consigna de desplazamiento y compara el sector donde se encuentra respecto al espacio particionado de trabajo. Señalando que, si la comparación de distancia resultare negativa, entiende que el giro del motor se ha de realizar a izquierdas, es decir, polarizándole negativamente.

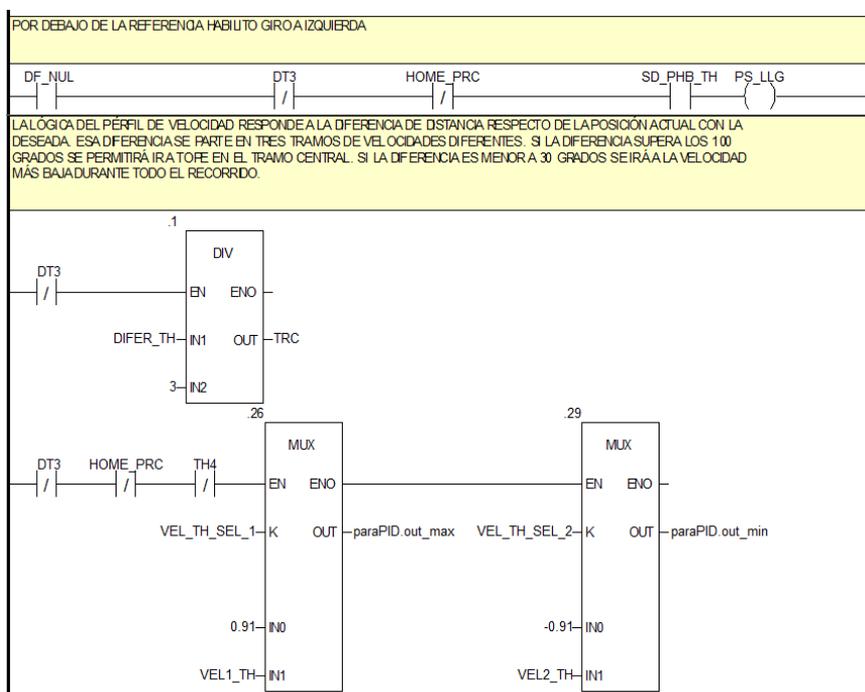


Figura 12-46. Sección de programa MOT\_SCARA, cuarta parte.

La figura 12-44 evidencia lo comentado. En esa sección del programa se compara contra cero (0) la partición del recorrido y si esta esta diferencia es mayor, lo que se entiende como positiva, se habilita la comparación del valor del encoder contra dicha partición, para el avance positivo y si esta no lo es, se habilita en sentido opuesto. En ese sentido, en la ilustración anterior se muestra la configuración de las dos consignas del perfil de velocidad.

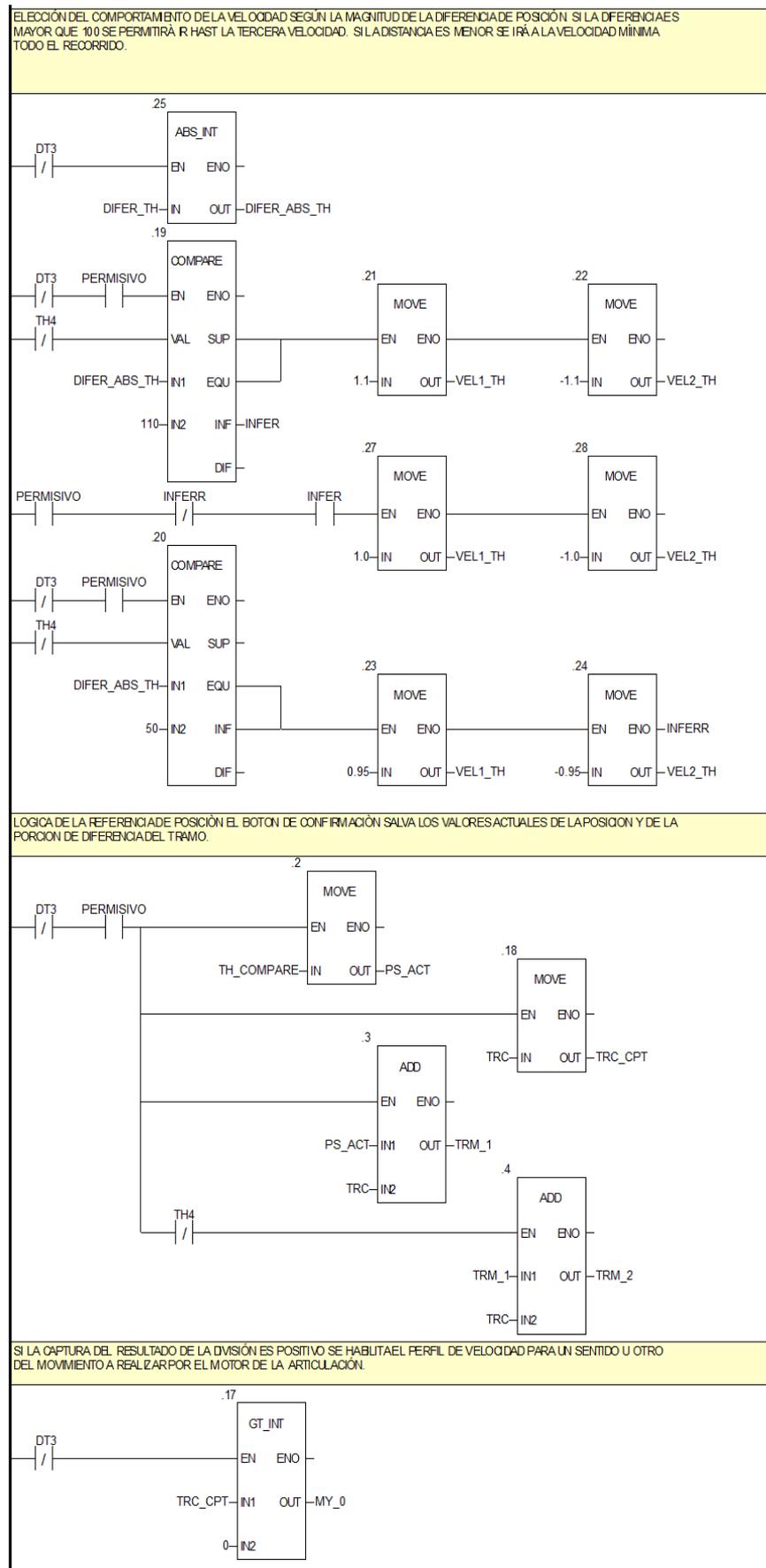


Figura 12-47. Sección de programa MOT\_SCARA, cuarta parte.

Las figuras 12-46 y 12-47 muestran la comparación de la distancia a recorrer, o como se mencionó anteriormente, la partición del espacio de trabajo.

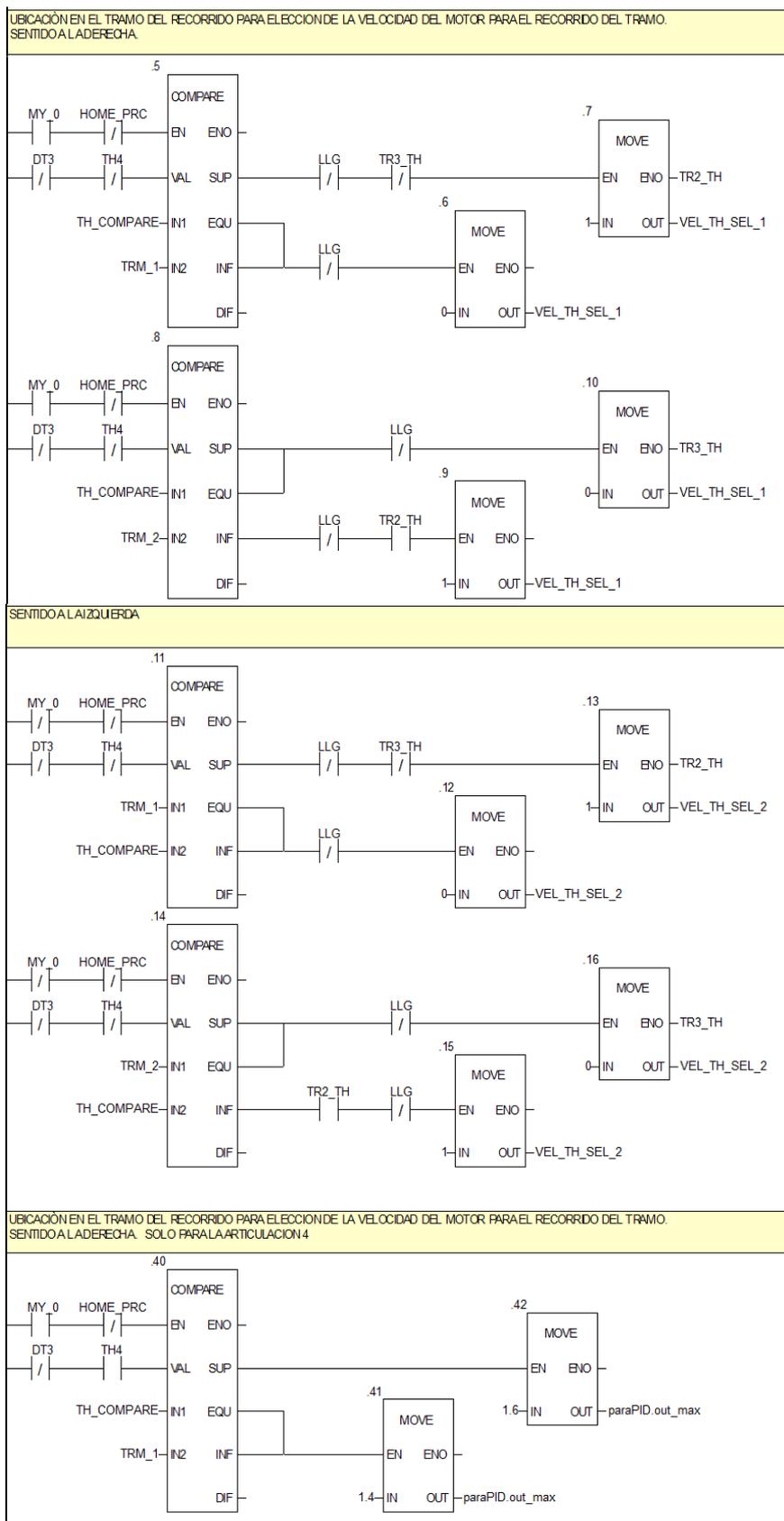


Figura 12-48. Sección de programa MOT\_SCARA, quinta parte.

La figura 12-49 muestra la inclusión del controlador PID. A este se le han definido los parámetros siguiendo el planteamiento de las oscilaciones sostenidas sobre la consigna propuesto por Ziegler & Nichols. En el que además de lo ya comentado previamente sobre su desempeño, se comenta que tiene un buen rechazo a las perturbaciones posicionales, producto de los esfuerzos del manipulador.

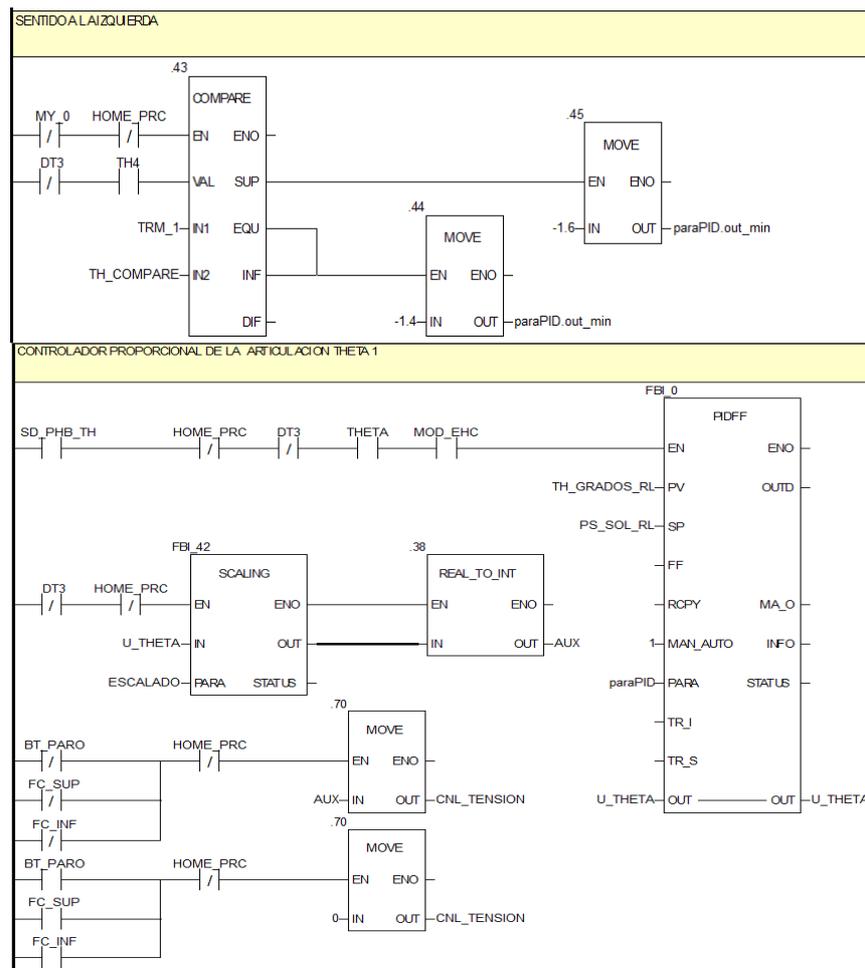


Figura 12-49. Sección de programa MOT\_SCARA, sexta parte.

En resumen, el programa para los motores de corriente directa prevé las diferencias pequeñas de dichos servomotores, modifica las magnitudes del espacio de trabajo que los eslabones acoplados recorrerán, condiciona las velocidades con las que se desplazan y parametriza el controlador particularizando cada motor.

### 12.5.3. ANEXO.5.3. CONTROL POSICIÓN Y VELOCIDAD DE MOTOR PASO A PASO

Tal como se hiciera con los motores de corriente directa, la sección de programa del bloque *MOT\_RBT*, debía comandar también el avance del motor a pasos que se instaló en la articulación prismática. El planteamiento inicial incluía para tal fin, un set de instrucciones implementadas que gobernarían, aunque siguiendo un planteamiento distinto, al motor a pasos. Todo ello debido a la marcada diferencia del modo de accionamiento que este presenta en comparación con los motores de corriente continua.

La figura siguiente tomada igualmente del programa *ACTUADORES*, muestra que en lugar del plan descrito, en su lugar fue preciso diseñar un nuevo bloque para conducir el desempeño del motor a pasos. El motivo se sustentó en que el autómata no era capaz de accionar todos los estados de las variables en el mismo ciclo de máquina, inclusive se observaba el caso de bobinas set y reset que estando conectadas en paralelo, se enclavaban alguna de ellas y las otras no.

Ciertamente la carga de procesamiento es alta en este programa, y se vinculaban señales de contaje rápido en funciones con otras funciones embebidas, con lo cual el rendimiento del encadenamiento de funciones producía el detrimento de la respuesta del autómata.

La solución derivó en subir de nivel al bloque de función *Mv\_Mt\_Rap* que maneja el motor paso a paso, en otras palabras, quitarlo del programa *MOT\_SCARA* que reside en *MOT\_RBT* y disponerlo directamente en *MUEVEACTUADOR*, vinculándole directamente las señales a las entradas y salidas como se muestra en la figura siguiente y obteniendo como resultado el adecuado desplazamiento del motor ante las consignas que se le envían al mismo, desde las diferentes funciones del programa principal.

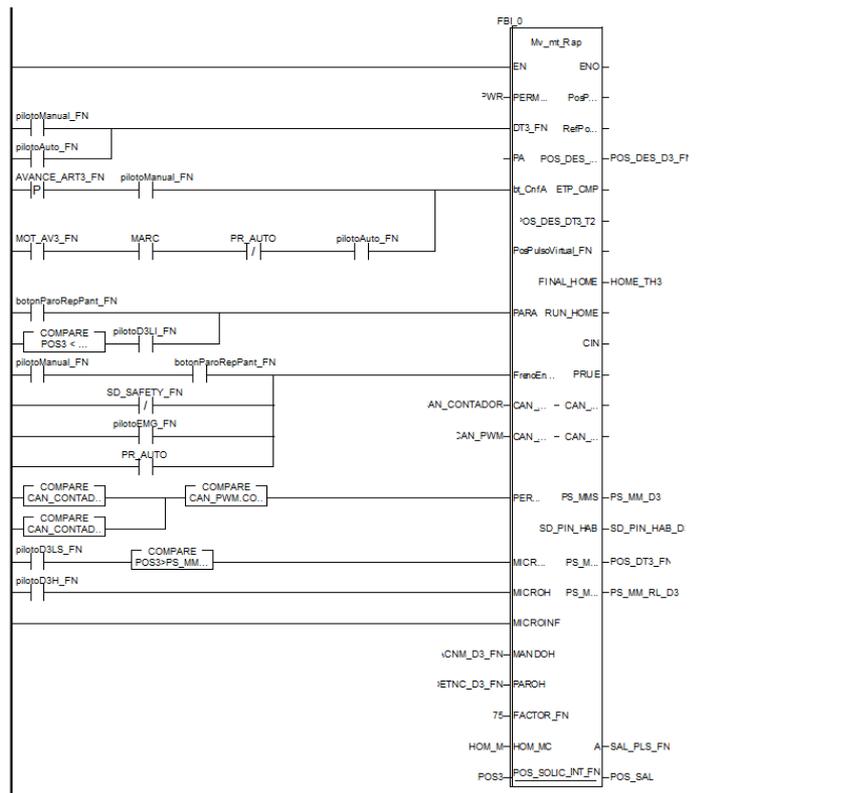


Figura 12-50. Sección de programa ACTUADORES, tercera parte.

En la tabla12-12 que se muestra seguidamente, se lista la configuración de entradas y salidas del bloque.

Tabla 12-12. Lista de variables del bloque de función Mv\_Mt\_Rap.

**Lista de Variables del Bloque Derivado de Función Mv\_Mt\_Rap.**

ENTRADAS			SALIDAS			PRIVADO	
#	Nombre	Tipo	#	Nombre	Tipo	Nombre	Tipo
1	PERMISIVO_FN	BOOL	1	PosPlsVirtual	DINT	AVZ_IZQ	BOOL
2	DT3_FN	BOOL	2	RfPsPlsVir_FN_B	DINT	LIM_DER	BOOL
4	Bt_Cnf	WORD	3	POS_DES_DT3_FN_T	INT	LIM_IZQ	BOOL
8	PARA	WORD	4	ETP_CMP	INT	PR_INTER	BOOL
10	FrenoEnReposo_FN	BOOL	5	POS_DES_DT3_T2	BOOL	DELTA3_MMTS	DINT
14	PERMISO_FN	BOOL	6	PosPulsoVirtual_FN	DINT	FRENOR	BOOL
16	MICROSP	BOOL	7	FINAL_HOME	BOOL	CONFIRMA	BOOL
17	MICROH	BOOL	8	RUN_HOME	BOOL	CN_CNT_DISABLE	BOOL
18	MICROINF	BOOL	14	PS_MMS	INT	JMPER_FN	REAL
19	MANDOH	BOOL	15	SD_PIN_HAB	BOOL	IGUAL_FN	REAL
20	PAROH	BOOL	16	PS_MMTS_DINT	DINT	ETAPA1	BOOL
21	FACTOR_FN	DINT	17	PS_MMT_R	REAL	ETAPA2	BOOL
22	HOME_MC	BOOL	22	A	INT	ETAPA3	BOOL
						ETAPA4	BOOL
						ETAPA5	BOOL
						ETAPA6	BOOL
						ETAPA7	BOOL
						ETAPA8	BOOL
						POS_SOLIC_HOME_FN	INT

ENTRADAS – SALIDAS		PRIVADO	
POS_SOLIC_INT_FN	INT	ENCLV_DT3_FN	BOOL
CAN_PWM_FN	T_UNSIGN_CPT _BMX	botonnuevo	BOOL
CAN_CONTAD_FN	T_SIGN_CPT_B	RESET_D3	BOOL
		AVZ_DER	BOOL

## MX

Para manejar el motor a pasos, se ha partido desde la virtualización de las referencias posicionales, porque a diferencia de los servomotores que poseen los encoders incrementales para retroalimentar la posición que posee el eslabón, sobre la articulación prismática esta información queda condicionada al registro de una magnitud en la memoria del autómata, es decir, en la articulación la dimensión del espacio es definida por la cantidad de pulsos que se envían al motor para completar el recorrido total, no existe una retroalimentación física del robot.

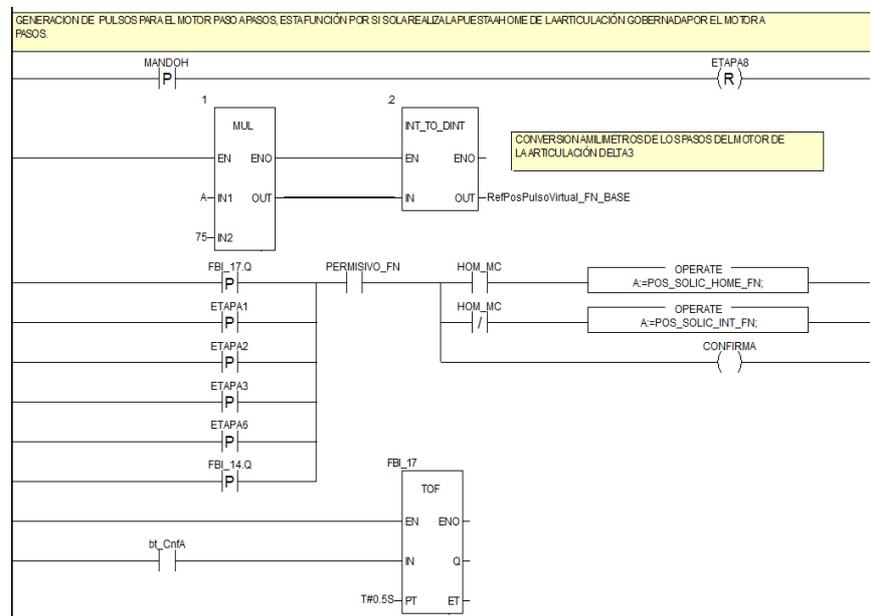


Figura 12-51. Sección de programa MV\_MT\_RP.

Entonces conociendo de la hoja de especificaciones del motor, que este utiliza 200 pulsos para completar una vuelta. Y habiendo medido unos 8458 pulsos para lograr recorrer 112.9 milímetros de carrera total, por lo tanto, se invierten alrededor de 74.92 pulsos para cada milímetro de recorrido, poco más de un tercio de revolución, y arroja como resultado una alta precisión de posicionamiento, siempre que no se presenten pérdida de pasos en el accionamiento.

Sin embargo, la verdadera dificultad de la aplicación radica en la rapidez del desplazamiento, que debido a la configuración modular del autómata fue difícil superar conmutaciones de pulsos inferiores a 8 milisegundos, unos 125 Hercios (Hz) al ritmo pulsante del ciclo de máquina, el cual se encuentra afectado por la demanda de procesamiento de la CPU del controlador. Se aplicó esta estrategia para el accionamiento y viendo su desempeño, se ha optado finalmente por la opción de incluir una nueva tarjeta de conteo rápido, para realizar un lazo PWM supervisado por el mismo módulo y garantizar que el avance longitudinal sea congruente con la rapidez de los otros eslabones del robot.

Con la aplicación PWM de la tarjeta rápida se pueden conseguir hasta 4000 hercios cuando el ciclo de servicio de la onda es menor del 50%. Y desde el canal para la modulación del ancho del pulso, se le envía la segunda salida como reflejo de los pulsos enviados al motor, hacia el canal de conteo configurado como contador grande libre.

Así el procesador interno del módulo EHC-0200, es el encargado de las estimaciones lógico-matemáticas que se emplean en el programa, por ello se reduce el procesamiento dentro de la CPU del autómata y el posicionamiento y la rapidez del eslabón se incrementa significativamente.

De la siguiente ilustración se aprecia que la configuración utilizada en el SCARA, que es de 3500 hercios a un ciclo de trabajo del 15%. También se muestra como se compara la ubicación del eslabón contra un umbral definido por las variables internas del módulo, en contra de la consigna que se le indica a seguir. Si esta es mayor que la ubicación (LOWER), se habilita el giro en un sentido para que la articulación se extienda hasta alcanzar la referencia. Recíprocamente al ser menor (UPPER), el sentido de giro será el opuesto con lo cual la articulación se retrae en busca de igualar la posición actual contra la nueva consigna.

En términos de calidad, después de haber aplicados las dos posibilidades de configuración, el emplear la salida

rápida optimiza el desempeño de la articulación prismática, sin embargo, en las condiciones que limita el autómatas, el accionamiento con servomotores supera en prestaciones al performance del motor a pasos.

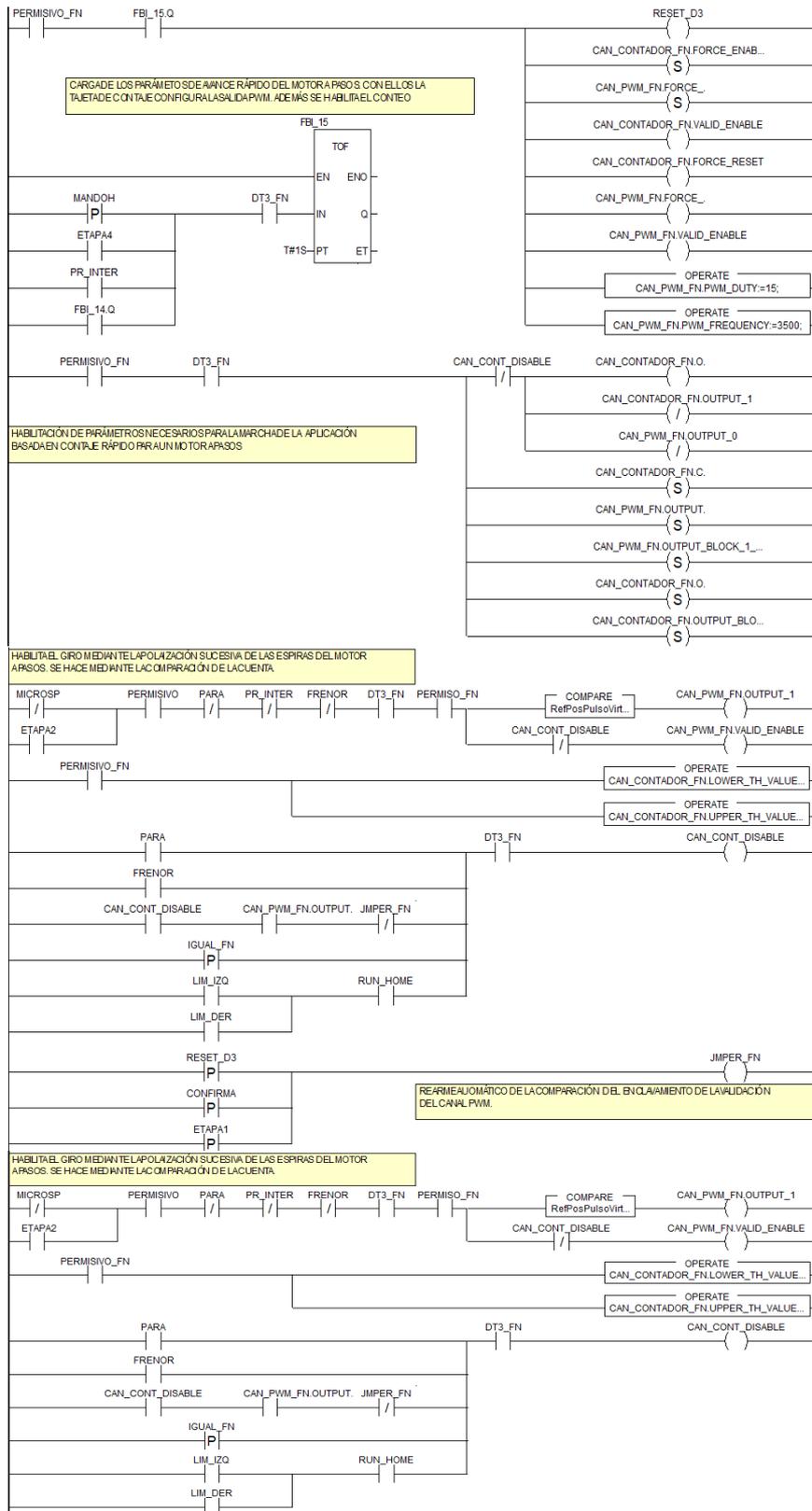


Figura 12-52. Sección del programa MV\_MT\_RP, segunda parte.

Como ya se comentó previamente, la configuración de *MV\_MT\_RAP*, dificultaba la puesta a home desde *FCN\_HOME*, por ello se decidió incluir dentro de la función del motor a pasos, una secuencia que fuese ejecutada y coordinada desde el proceso de puesta a home ubicado en el programa principal, evolutivamente

idéntica, pero que residiese en el bloque de función dedicado al motor. Esta solución ha obedecido a la necesidad de hacer el reset en varias oportunidades de la tarjeta de conteo y se hizo mas sencillo a realizarlo desde la misma sección de programa del bloque.

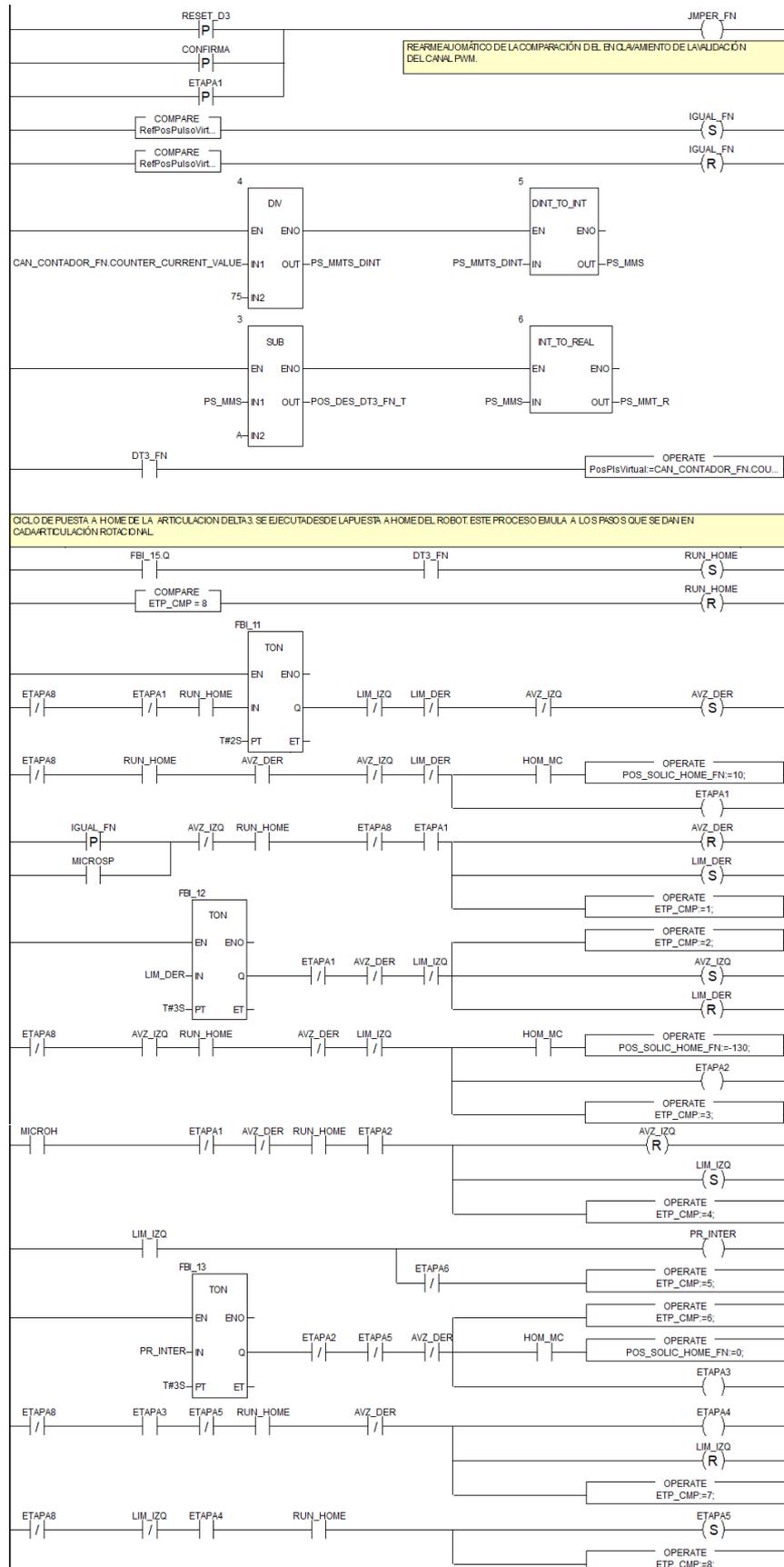


Figura 12-53 . Sección del programa MV\_MT\_RP, tercera parte.

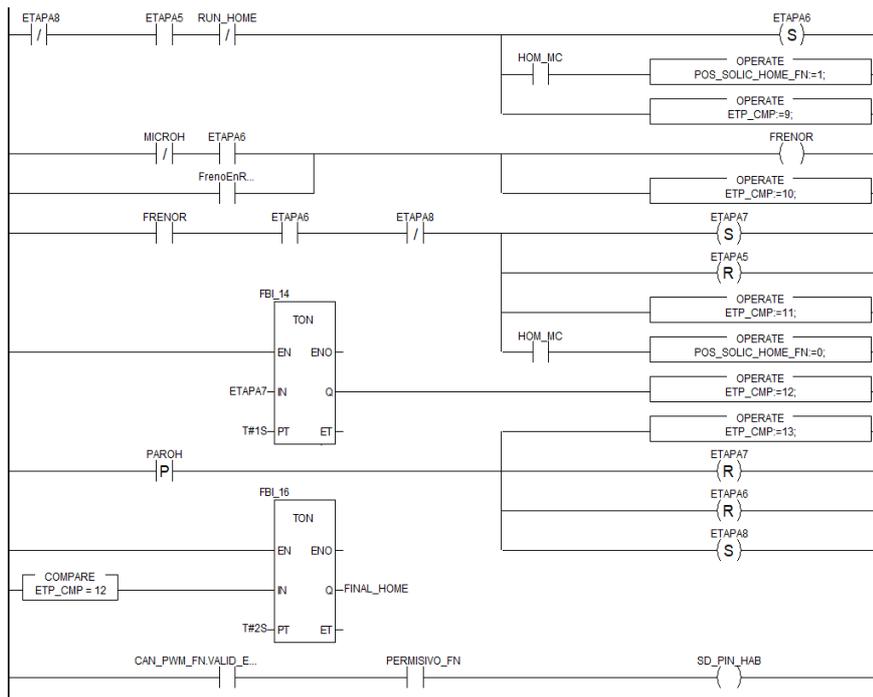


Figura 12-54 . Sección del programa MV\_MT\_RP, cuarta parte.

Además, es preciso comentar que en las figuras donde se aprecia el proceso de puesta a home de la articulación prismática, se ve incluido también la variable para dar seguimiento a la evolución de la secuencia de nombre *ETP\_CMP*. Esta estrategia ha sido de gran ayuda para el desarrollo de cada una de las funciones que son incluidas en el programa, recordando que al estar visualizando la evolución del programa desde el ordenador, la lógica presente en los bloques derivados de función creados por el usuario, no dinamizan el gráfico o el texto de la sección, como si se hace sobre las tareas del programa principal.

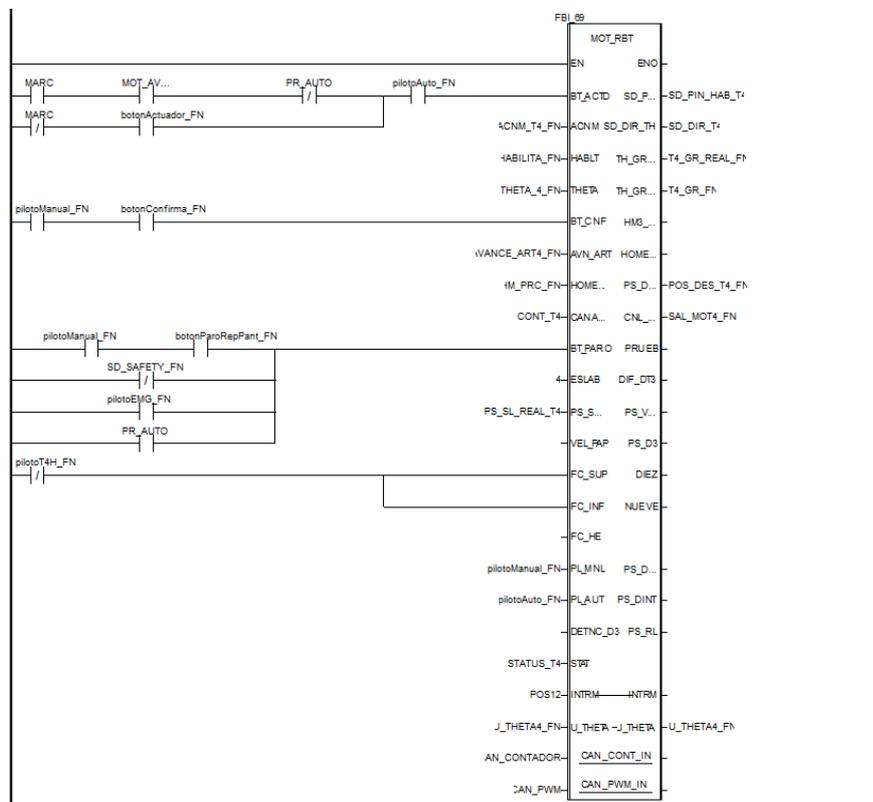


Figura 12-55 . Sección de programa ACTUADORES, cuarta parte.

Finalmente, el programa del bloque dedicado a los cuatro actuadores termina con la aplicación del bloque *MOT\_RBT*, al servo que gobierna la muñeca del robot. Esta tiene como diferencias la utilización de un solo sensor para detectar la posición de home de la articulación. No incluye los dos que limitan el extremo inferior y el superior de la carrera. Esta diferencia se puede observar en la figura anterior.

Por otro lado, en la ilustración siguiente, son separadas por un permisivo que se emite en la función de home, las consignas que se llevan a los motores de corriente directa, a través de las salidas de las tarjetas analógicas dispuestas en el bastidor del autómata.

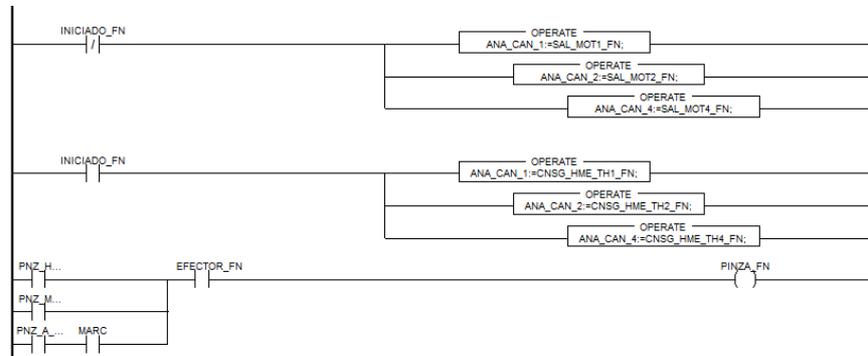


Figura 12-56 . Sección de programa ACTUADORES, quinta parte.

Así las consignas emitidas desde la tarea de puesta a home no sobrescriben las emitidas desde la operación manual o automática del SONY SRX 4CH.



Tabla 12-13. Lista de variables del bloque de función MOV\_AUTO.

Lista de Variables del Bloque Derivado de Función MOV_AUTO							
ENTRADAS			SALIDAS			PRIVADO	
#	Nombre	Tipo	#	Nombre	Tipo	Nombre	Tipo
1	PL_AUTO_FA	BOOL	1	PAROFINAL_FN	BOOL	POSA	BOOL
2	PVEC0	INT	2	POS_SOLIC_REAL_TH1_OUT	REAL	POSA1	BOOL
3	PVEC1	INT	3	POS_SOLIC_REAL_TH2_OUT	REAL	POSA2	BOOL
4	PVEC2	INT	4	REF_REAL_OUT	REAL	POSA3	BOOL
5	PVEC3	INT	5	POS_SOLIC_REAL_TH4_OUT	REAL	POSB	BOOL
6	PVEC4	INT	6	AUTO_OUT_FN	BYTE	POSB1	BOOL
7	PVEC5	INT	7	TRNS_AUTO_OUT	BYTE	POSC	BOOL
8	THETA1_GRADOS_FN	DINT	8	AUT_G	INT	POSC1	BOOL
9	THETA2_GRADOS_FN	DINT	9	PCK_PLC_S	BOOL	POSD	BOOL
10	PS_MMTS_FN	DINT	17	SAL_STR	BOOL	POSD1	BOOL
11	THETA4_GRADOS_FN	DINT	18	SAL_STOP	BOOL	POSN0	BOOL
12	POS_DES_TH1_FN	INT	19	CONT_PRD	BOOL	POS0PA	BOOL
13	POS_DES_TH2_FN	INT				MOT_AV1_OUT	BOOL
14	DIF_TH3_FN	INT				MOT_AV2_OUT	BOOL
15	POS_DES_TH4_FN	INT		<b>PRIVADO</b>		MOT_AV3_OUT	BOOL
16	VC_POSICIONES_FN	ARRAY		POS_SOL_AUT1_OUT	REAL	MOT_AV4_OUT	BOOL
17	SD_PIN_HAB_DLT3_FN	BOOL		POS_SOL_AUT2_OUT	REAL	S_DSPZ_RF1	REAL
18	BT_AUTO_FA	BYTE		POS_SOL_AUT3_OUT	REAL	S_DSPZ_RF2	REAL
19	R_FALLO	BOOL		POS_SOL_AUT4_OUT	REAL	S_DSPZ_RF3	REAL
20	PL_MRC_AUT	BOOL		pilotoEMGNC_FN	BOOL	S_DSPZ_RF4	REAL
21	PIEZA_REAL	BOOL		pilotoMANUAL_FN	BOOL	LL_POS0	BOOL
				BotonStart_FN	BOOL	LL_POS1	BOOL
				BotonStop_FN	BOOL	LL_POS2	BOOL
				ACT_INICIADO	BOOL	LL_POS3	BOOL
				POSICN	BOOL	LL_POS4	BOOL
				PILOTOAUTO_FA	DINT	LL_POS5	BOOL
				MARCHA_AUTO	DINT	TRNS1	BOOL
				CICLO	DINT	TRNS2	BOOL
				ARRANQUE	BOOL	TRNS3	BOOL
				PARADA	BOOL	TRNS4	BOOL
				PARAR	BOOL	TRNS5	BOOL
				PARO	BOOL	TRNS6	BOOL
				PCK_PLC	DINT	TRNS7	BOOL
				MODO2	DINT	TRNS8	BOOL
				POS0	DINT	TRNS9	BOOL
				PIEZA_SIMUL	BOOL	TRNS10	BOOL
				PNZ_A_OUT	BOOL	TRNS11	BOOL
				I_VECTOR	INT	TRNS12	BOOL

En el programa en primer lugar se reciben desde la pantalla de explotación las condiciones de operación del sistema. Estas son transmitidas al bloque de función empaquetadas dentro de una variable del tipo BYTE, vinculando así, por medio de una sola entrada al bloque hasta 8 estados digitales, que sirven para indicarle a la secuencia cuando iniciar, cuando detener el ciclo o cuando inhibir su ejecución ante estados de marcha manual o ante estados de emergencia.

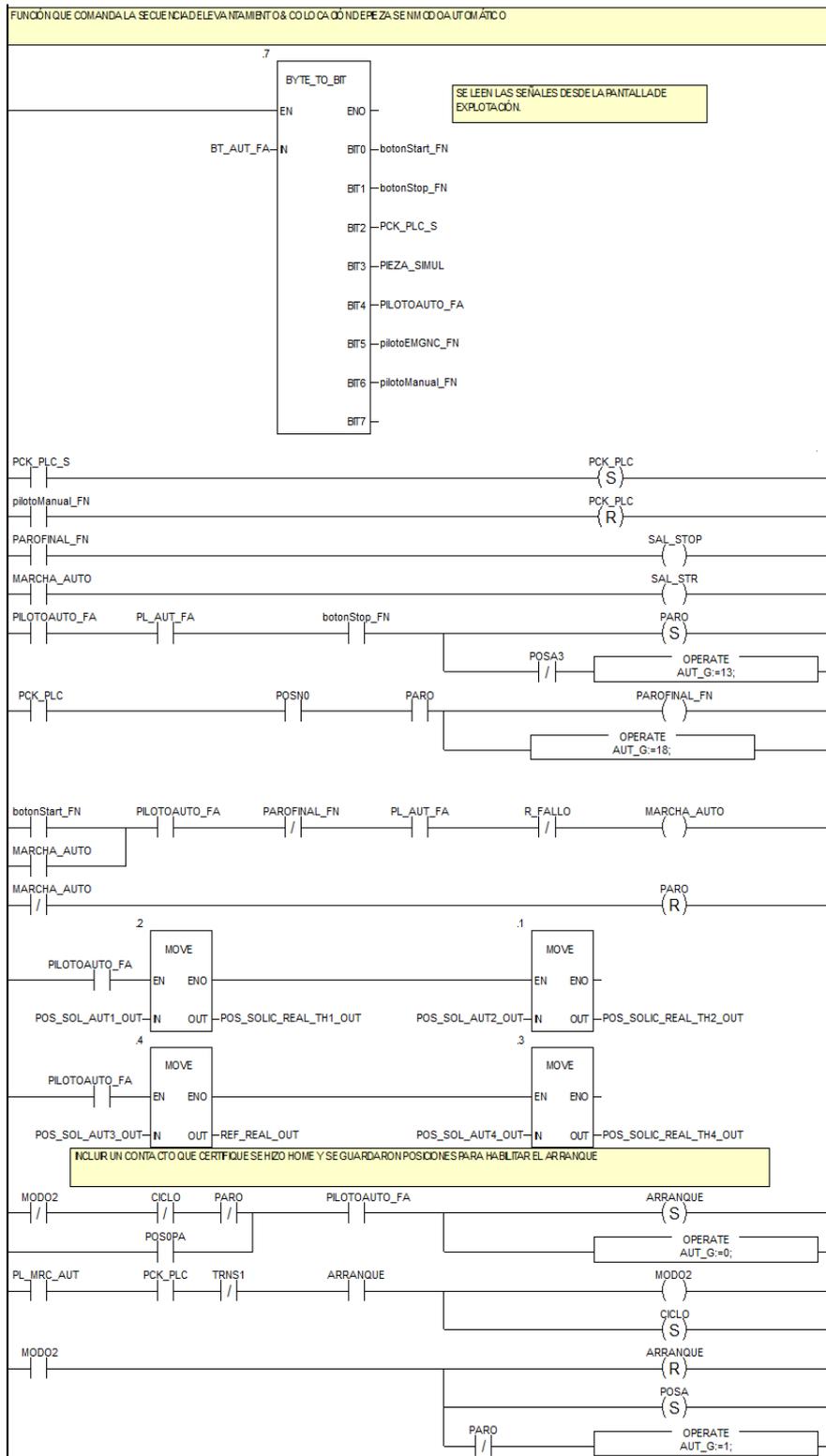


Figura 12-59. Sección de programa S\_AUTO.

Por otra parte, en la ilustración anterior se logra apreciar que toda vez que el usuario elija la operación en modo automático, le otorga el dominio de los estados del automatismo a la función *MOV\_AUTO*, en otras palabras, es la función la que le indica al programa principal, si el sistema se encuentra en marcha, ha comenzado a detenerse o si está detenido. Esto lo consigue mediante la vigilancia de las salidas SAL\_STR y SAL\_STOP, que se encuentran asociadas al pulsador start y al stop respectivamente, pulsadores que se hallan dispuestos en la pantalla de operador *COMANDO*.

### 12.6.1. ANEXO 6.1: RUTINA RECOGER Y COLOCAR

La rutina automática está centrada en la sujeción, transporte y colocación de objetos y la misma se aprecia en la ilustración 12-60. Dicha operación se aproxima a las tareas que regularmente se dan en las actividades de la célula de fabricación flexible, que en su gran mayoría se encargan de tomar elementos desde las bandejas que se trasladan sobre los transportadores y colocar en otra bandeja o hacia otras ubicaciones de la célula. En este caso, en la cercanía del lugar donde se tiene dispuesta la estación del SRX 4CH, se tiene la ubicación de un retenedor, para que las bandejas hagan una pausa en su avance y el robot ejecute acciones sobre ellas.

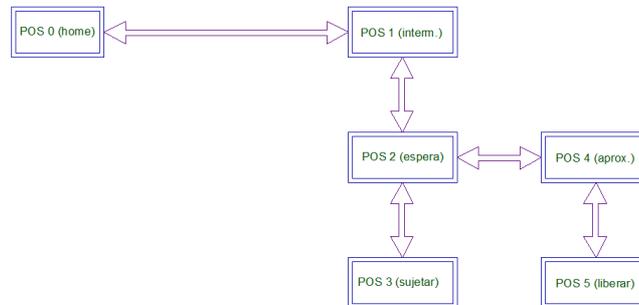


Figura 12-60. Configuración de posiciones de la secuencia automática

Por tal razón en la confección de la secuencia se tomó esta ubicación como la posición de recogida de piezas, que con las demás posiciones empleadas, se listan en la tabla 12-16 que se despliega a continuación y se presentan gráficamente en la figura anterior.

Tabla 12-14. Lista de posiciones de la rutina automática.

Lista de Posiciones y Acciones en la Función RECOGER Y COLOCAR					
POSICIONES			POSICIONES		
En vector	En secuencia	Descripción	En vector	En secuencia	Descripción
Pos. 0	Pos. 0	HOME	Pos. 3	Pos. B1	Cierre de pinza
Pos. 1	Pos. A	Posición Intermedia	Pos. 4	Pos. C, C1	Aproximación a soltar
Pos. 2	Pos. A1, A2, A3	Aproximación a pieza	Pos. 5	Pos. D	Liberación de pieza
Pos. 3	Pos. B	Recogida de pieza	Pos. 5	Pos. D1	Apertura de pinza

En la lista han sido incluidas como una posición adicional las acciones de apertura y cierre de la pinza, aunque en el entorno físico real estas se encuentran sobre las mismas posiciones de sujeción y liberación del objeto que se manipula.

Como resultado de la necesidad de asir las piezas, en el programa se han incluido dos acciones a ejecutar para tomarlo de la bandeja del transportador y liberarlo sobre una segunda posición que se elija de manera arbitraria y que satisfaga la necesidad de la implementación.

Siendo que en la búsqueda de completar de manera confiable la secuencia de avances del robot, se ha configurado y se recomienda seguir este planteamiento en la confección de futuras trayectorias, un plan de avances conservador, ante las limitaciones de rapidez de desplazamientos del SCARA. En otras palabras, sabiendo que el motor a pasos no es capaz de seguir en las condiciones actuales, la respuestas de los servoaccionamientos, se configuran posiciones de aproximación que tratan de acortar la distancia vertical y pretenden aislar el movimiento en ese sentido, para que el mismo se ejecute sin simultaneidad articular sobre el eje Z. Así se conseguirán mejores prestaciones en el desempeño de la estación.

Es entonces que al pulsar el botón de inicio, y siempre que se haya elegido la rutina con el botón *PCK PLC*, el manipulador se desplaza desde la posición de *HOME*, sobre el plano *XY* hasta la posición intermedia *POS1* con la garras abierta. Luego desciende a la posición de aproximación *POS2*, en donde aguarda por la percepción de la presencia de una pieza a tomar.

Esta condición es simulada con el botón *PIEZA* de la pantalla de operador *COMANDO*, al ser pulsado el botón es enclavada la variable *PIEZA\_SIMUL*. Esta en el programa, se encuentra conectada en paralelo con una entrada del bloque *MOV\_AUTO* de nombre *PIEZA\_REAL*. Dicha entrada el usuario le empleará para vincular por medio de la Ethernet presente en la célula, los estados de los sensores con numerales 14 y 17 de la figura 5-5 y que son

identificados en la tabla 5-8 como supervisores de la presencia de bandejas en el retenedor del SONY SRX 4CH.

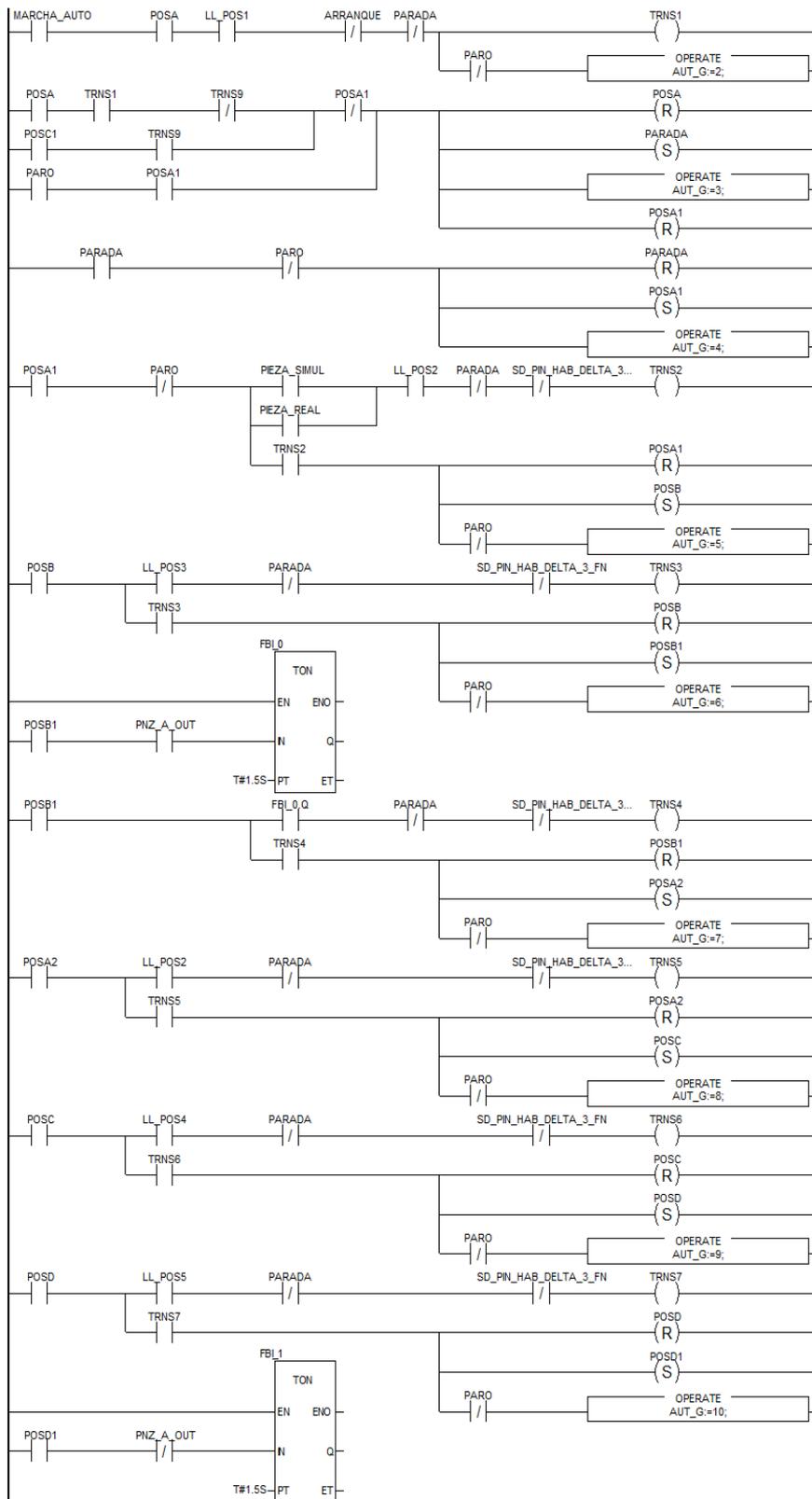


Figura 12-61. Sección de programa S\_AUTO, segunda parte.

Al ser verdadera esta condición, el efector desciende a la posición de recogida de piezas POS3, sujetará la pieza y regresará a la posición de aproximación, desde allí comenzará a desplazarse nuevamente sobre el plano XY hasta la posición de aproximación a la descarga POS4.

Desde esta locación desciende a depositar el objeto en la ubicación POS5. El efector se abre y regresa a la POS2,

pasando previamente por la posición de aproximación POS4.

En las ilustraciones que en esta sección se emplean para describir el ciclo automático, al igual que se hicieron en la sección de programa *PST\_HOME*, se muestra que también se ha incluido una salida entera *AUT\_G*, con la intención de monitorizar el avance del ciclo. Por ejemplo, en la evolución de la secuencia descrita, de la figura 12-60 se conoce que los valores irán de 3 a 11, desde el momento de simular la presencia de pieza, hasta liberarla en la posición destino.

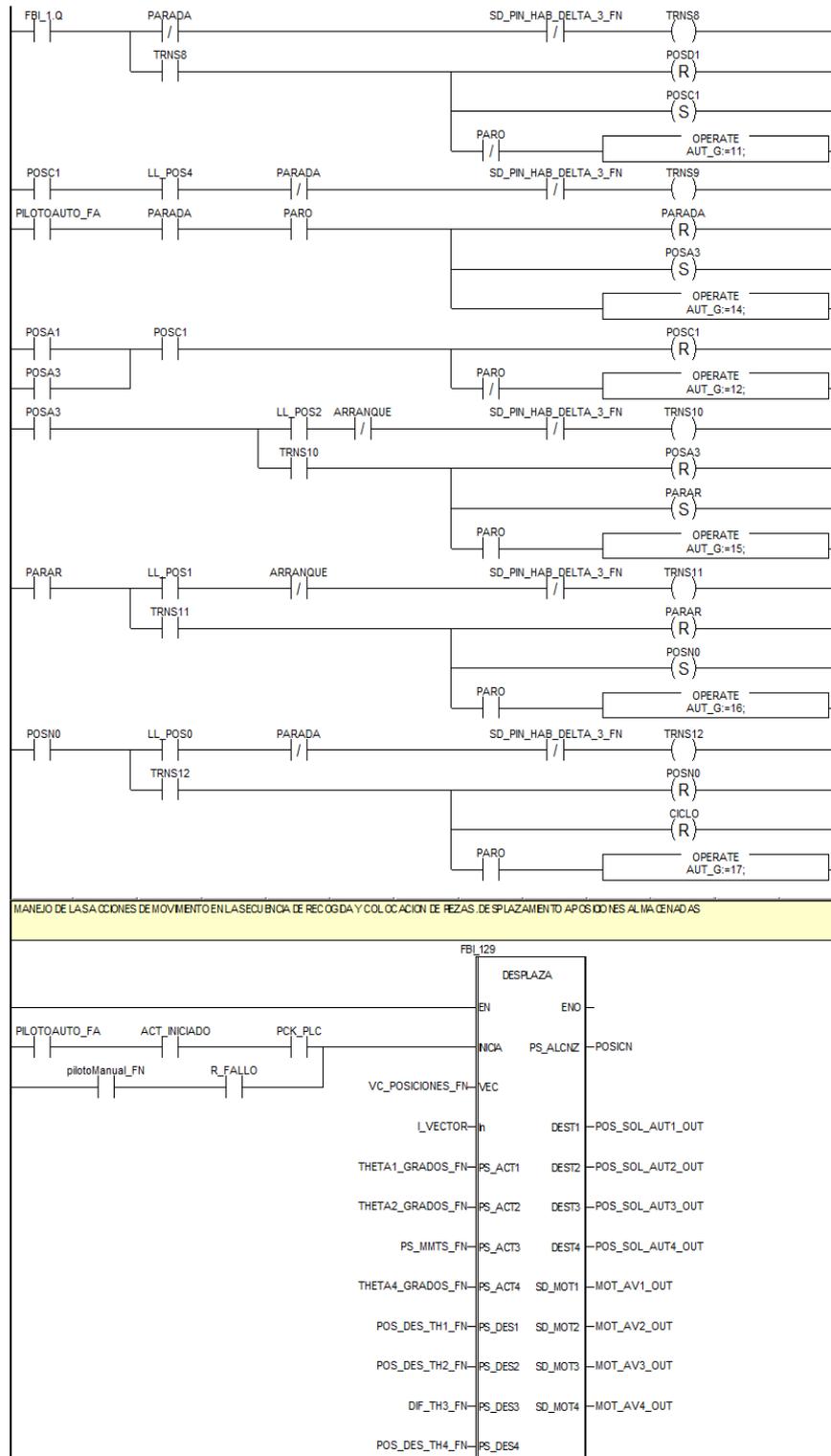


Figura 12-62 . Sección de programa S\_AUTO, tercera parte.

Esta evolución cíclica mantendrá ejecutándose hasta que se emite una orden de parada normal o controlada con la conmutación del botón Stop. Esto deviene en dos posibles respuestas, en la ilustración 12-62 se observa que

mediante el enclavamiento de la bobina *PARO*, el ciclo que actualmente se ejecuta, de estar en marcha, continua su marcha hasta que el manipulador complete la tarea, regrese a la *POS2* y desde esta se dirija a la posición de home pasando previamente por *POS1*, causando que todas las bobinas sean desenergizadas y se quede el sistema a la espera de otra orden de marcha automática al llegar a dicha ubicación.

La segunda respuesta posible se tendrá cuando el manipulador esté detenido en la espera de la presencia de pieza, al ser obtenida la orden de parada, el manipulador se dirige desde su ubicación actual a la posición de home pasando previamente por *POS1* y nuevamente se habrá de aguardar una nueva orden de marcha automática o por alguna otra instrucción que el operador emita o configure para el empleo del robot.

## 12.6.2. ANEXO 6.2: FUNCIÓN DESPLAZA

Otra herramienta creada para el programa *S\_AUTO*, es la función de desplazamiento combinado del SCARA. Que ha sido llamada por simplicidad *DESPLAZA* y es posible observarla al final de la ilustración anterior. La función posee una sección de programa en escalera nombrada *MOVER*, que emula la función *MOVE* del lenguaje *ATS*, pero este es un nombre reservado en Unity Pro por eso se ha marcado la diferencia de nombre.

La tabla 12-15 contiene las variables del bloque y gracias a la figura anterior es posible comprender como se han de enlazar las variables del programa principal al conjunto de entradas y salidas de la función.

Tabla 12-15. Lista de variables del bloque de función *DESPLAZA*.

Lista de Variables del Bloque Derivado de Función <i>DESPLAZA</i>							
ENTRADAS			SALIDAS			PRIVADO	
#	Nombre	Tipo	#	Nombre	Tipo	Nombre	Tipo
1	SL_ENC	BOOL	1	PS_ALCNZ	BOOL	DIFR_1	DINT
2	VEC	BOOL	2	DEST1	REAL	DIFR_2	DINT
3	In	INT	3	DEST2	REAL	DIFR_3	DINT
4	PS_ACT1	DINT	4	DEST3	REAL	DIFR_4	DINT
5	PS_ACT2	DINT	5	DEST4	REAL	RF_1	DINT
6	PS_ACT3	DINT	6	SD_MOT1	BOOL	RF_2	DINT
7	PS_ACT4	DINT	7	SD_MOT2	BOOL	RF_3	DINT
8	PS_DES1	INT	8	SD_MOT3	BOOL	RF_4	DINT
9	PS_DES2	INT	9	SD_MOT4	BOOL	DFR_REAL	BOOL
10	PS_DES3	INT	10	PL_D3H	BOOL	DFR_ENTERA	BOOL
11	PS_DES4	INT	11	PL_T4H	BOOL	SD_MOT	BOOL

La función de desplazamiento está destinada a ejecutar los movimientos del robot, a partir de las posiciones definidas en un vector de enteros dobles, que ha de insertársele a través de la entrada *VEC*, siendo este quien guarda el registro de configuraciones articulares a adoptar por el SRX 4CH y que ha sido alimentado como se ha descrito en el cuarto apartado de este manual operativo anexo.

La función *DESPLAZA* cuya rutina se despliega en la ilustración 12-63, ha de habilitarse con la entrada *INICIA*, obtendrá del vector de posiciones los valores de coordenadas articulares de destino que apunta el entero *In* y los ofrece a las funciones de movimiento del motor por medio de las salidas reales *DEST1*, *DEST2*, *DEST3* Y *DEST4*. Coloca en alto las salidas que habilitan el movimiento del motor *SD\_MOT1*, *SD\_MOT2*, *SD\_MOT3*, *SD\_MOT4*, para aquellas articulaciones en las cuales difiera la posición que demanda el vector a visitar y las posiciones actuales de los eslabones que son leídas por las entradas *PS\_ACT1*, *PS\_ACT2*, *PS\_ACT3*, *PS\_ACT4*. De manera redundante y para asegurar que se cumplan las condiciones de posicionamiento del efector se han de incluir por medios de las entradas enteras *PS\_DES1*, *PS\_DES2*, *PS\_DES3*, *PS\_DES4*, las diferencia a las posiciones destino, siendo que cuando esta magnitud se haga nula, es decir, se iguale a cero (0), se entiende que ha sido alcanzada la consigna de desplazamiento solicitada. De esta manera, el autómata comprende que ha completado el propósito, cuando las posiciones actuales del robot cargadas para cada articulación en sus respectivas entradas se han igualado a los registros del vector.

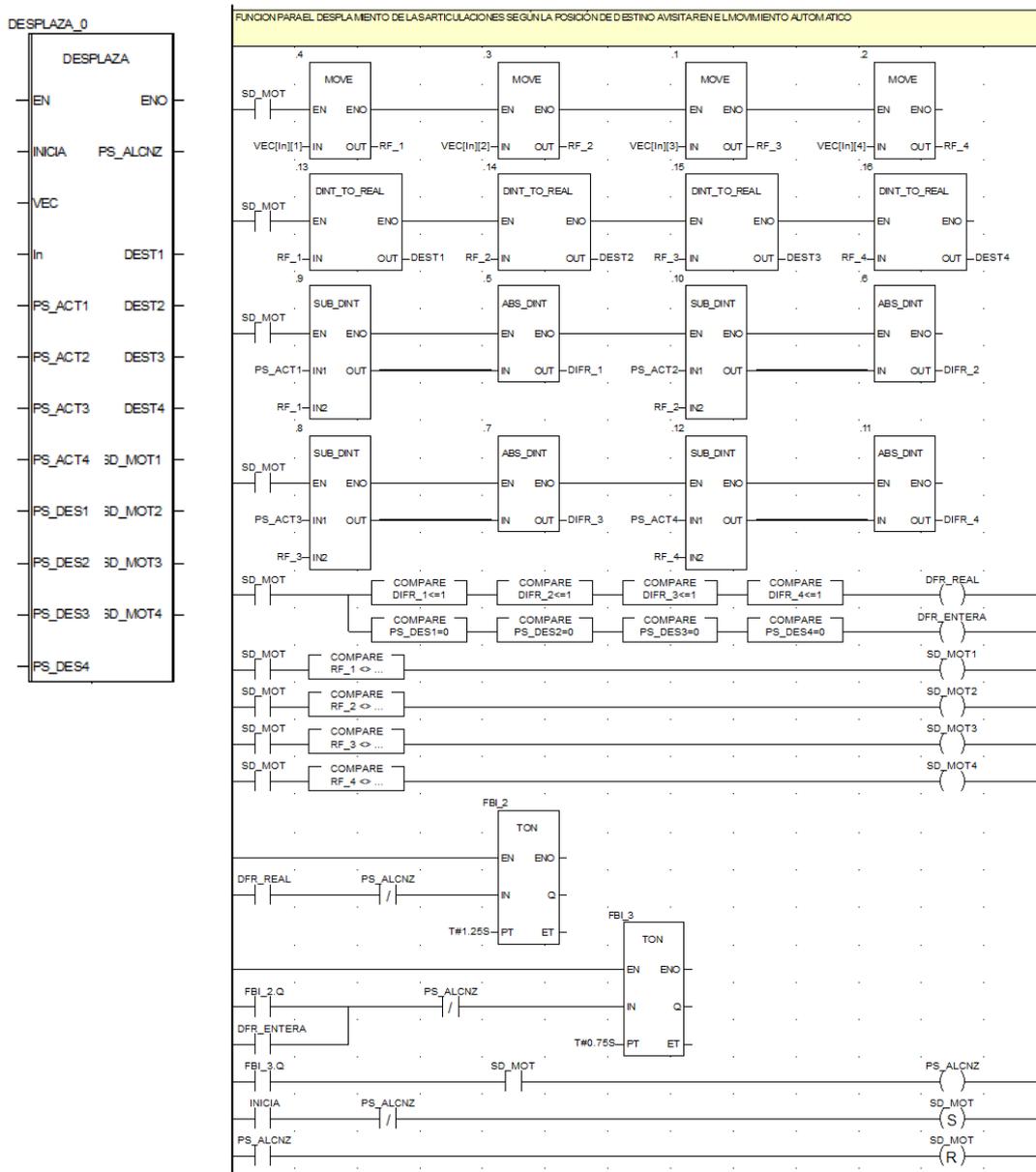


Figura 12-63. Bloque DESPLAZA. Sección de programa MOVER.

Para finalizar, la función coloca en alto la salida *PS\_ALCNZ* que indica que ha sido completada y esta condición se ha empleado en programa para flanquear la transición que habilita la próxima acción de la secuencia de movimiento.

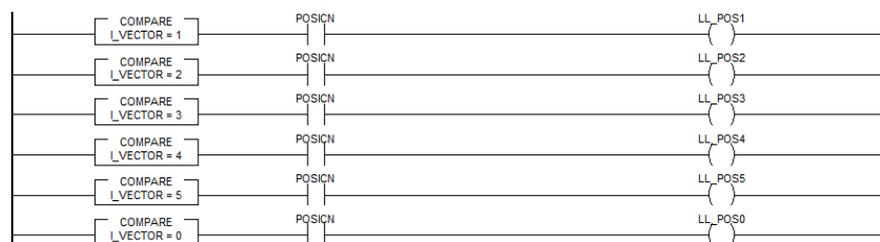


Figura 12-64. Sección de Programa S\_AUTO, cuarta parte.

Esta salida actua sobre la variable *POSICN* y como se observa en la figura anterior, se emplea para validar que se han alcanzado las posiciones que indica el vector que han de visitarse. En tal sentido, es la secuencia la que toma del arreglo matricial de posiciones, como se ilustra en la siguiente figura, según el ordinal que se le indique a las entradas *PVEC0*, *PVEC1*, *PVEC2*, *PVEC3*, *PVEC4* y *PVEC5* de la función *MOV-AUTO*, que configuración articular almacenada en el arreglo se ha de adoptar.

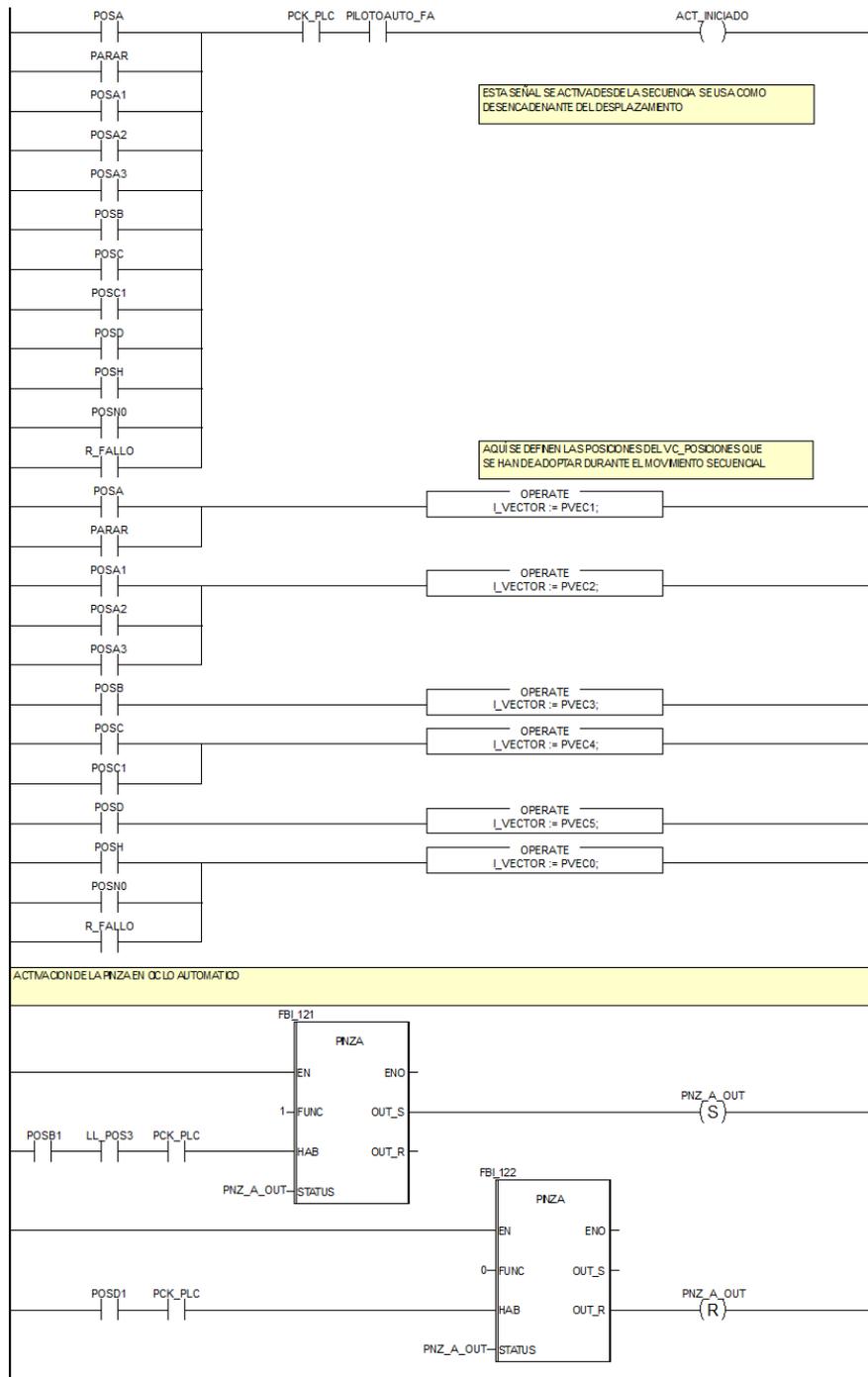


Figura 12-65. . Sección de Programa S\_AUTO, quinta parte.

En la figura anterior además se muestra la llamada para ejecutar la apertura o cierre del efector según la posición que se ha alcanzado y la evolución de la rutina.

Finalmente, la ilustración 12-66 que se muestra a continuación, son presentadas dos tareas importantes, la primera señala como son transmitidas al bloque de actuadores las ordenes de movimiento y en la segunda se indica como la función ha de reaccionar ante la ocurrencia de paradas abruptas de la secuencia. Siendo que se detendrán todos los movimientos del manipulador y se indicará mediante las interfaces de operador en rojo, será enclavada la variable *POSOPA*, con lo cual se abandona cualquier condición alcanzada. Posterior al rearme, al pulsar el botón de parada con las interfaces en amarillo, se ha de ir a modo manual y seleccionar la puesta a home, así será referenciado nuevamente el espacio de la tarea.

Cuando el *HOME* haya finalizado, se lleva *POSOPA* a su estado normal y se deja la función habilitada para iniciar nuevamente su operación, con lo cual la atención de emergencias finaliza con la elección del modo automático, con la elección de la rutina y el nuevo inicio de la marcha automática.

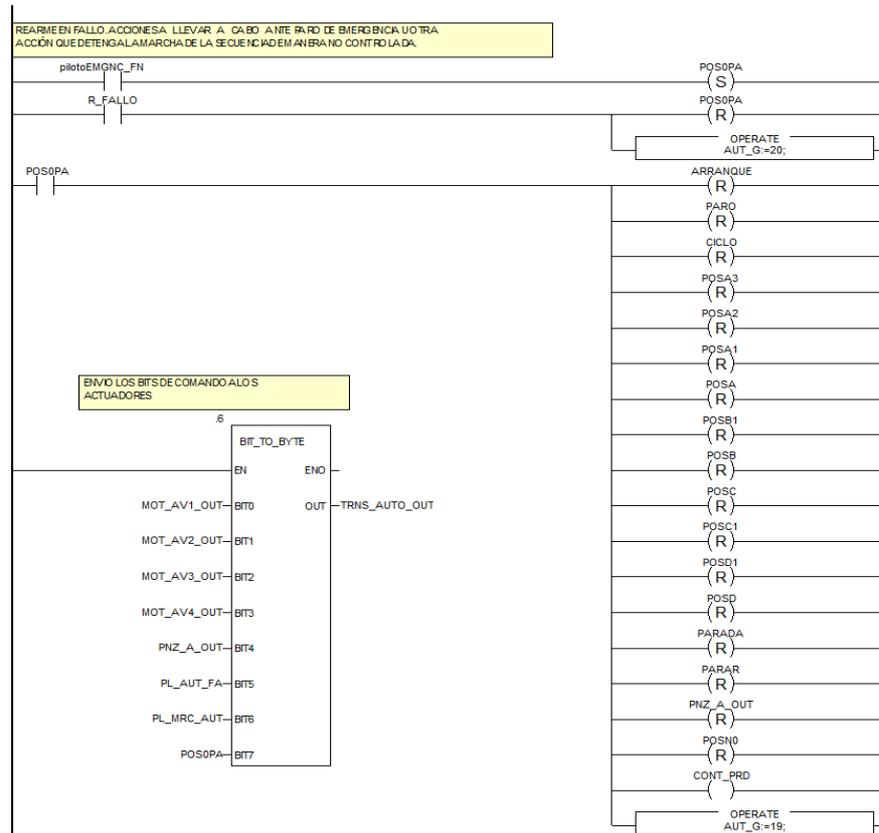


Figura 12-66. Sección de programa de transiciones MODO2, TR29 y TR36 de rutina PCK\_N\_PLC.

De esta manera se da por concluida la descripción del programa diseñado para operar el robot SCARA SONY SRX 4CH, mediante el empleo del autómata Modicon M340 de Schneider Electric que se emplea en este proyecto.

12.7. ANEXO 7: ESQUEMAS ELÉCTRICOS

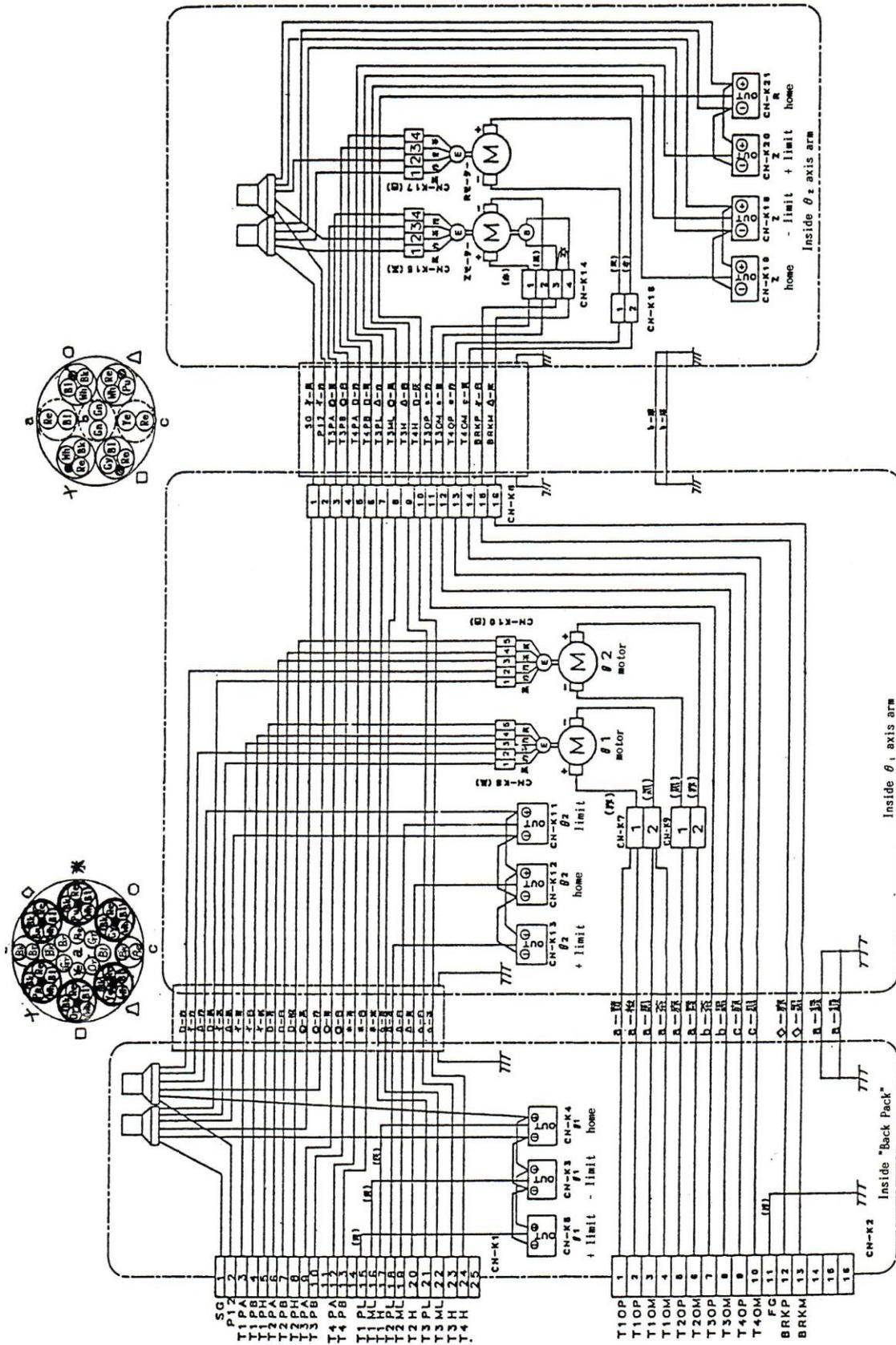


Figura 12-67. Diagrama Unifilar Original del SONY SRX 4CH.

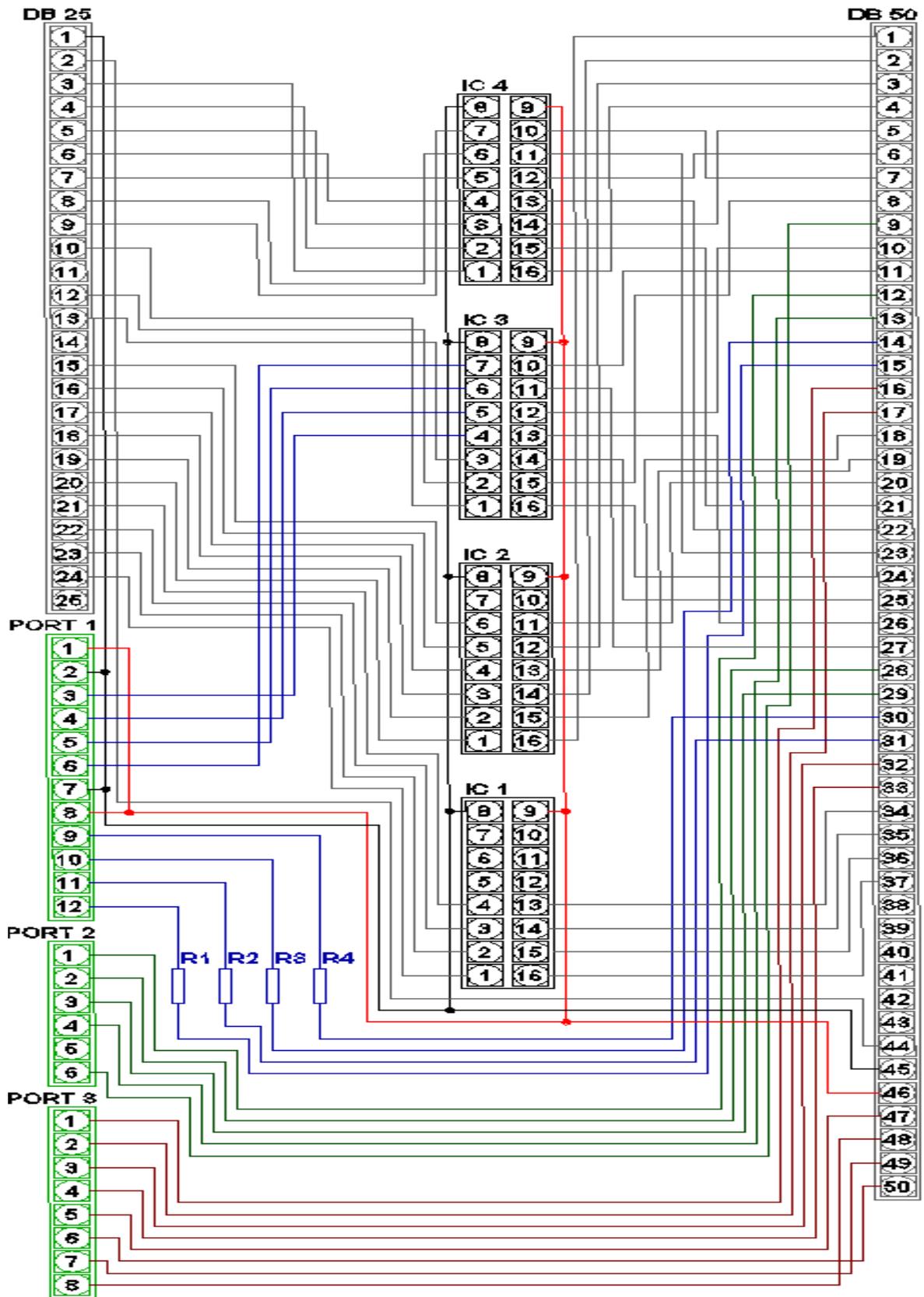


Figura 12-68. Diagrama Unifilar de Interfaz Electrónica con Darlingtons.



## REFERENCIAS

- [1] Advanced Motion Controls, «Series 25A Brush Type PWM Servo Amplifiers,» Manual de Configuración, Agosto, 1996.
- [2] Barrientos, A. Peñin, L. Balaguer, C. Aracil, R., «Fundamentos de Robótica,» Mc Graw-Hill, Segunda Edición, Madrid, 2007.
- [3] Candela Herías, F. A., «Introducción al software Unity Pro y a los PLCs M340 de Schneider,» Manual de prácticas, Universidad de Alicante, 2011.
- [4] Caso Solis, F. J., Control de Ejes mediante PLC Aplicación en Célula de Fabricación Flexible, Proyecto Final de Carrera, Universidad de Sevilla 2016.
- [5] Cecarelli, M., «Fundamentals of Mechanical Robotics Manipulation,» Springer Science, Vol. 27, p.22, University of Cassino, Italy, 2004.
- [6] Iglesias Bayo, R., Integración de equipos en célula de fabricación flexible, Trabajo Final de Grado, Universidad de Sevilla 2016.
- [7] López de Ahumada Gutierrez, J. L., «Realización de un Controlador para Célula de Fabricación – Interfase entre Ordenador y Robot SRX - 4CH de Sony,» Proyecto Fin de Carrera, Hogeschool Eindhoven (Países Bajos) – Universidad de Sevilla, 1997.
- [8] Ramos Ramírez, F. J., Control y Supervisión de Célula de Fabricación Flexible, Universidad Sevilla, 2008.
- [9] Schneider Electric, «Modicon M340 para Ethernet - Procesadores y módulos de comunicaciones,» 2009. [En línea]. Available: [http://www2.schneider-electric.com/resources/sites/SCHNEIDER\\_ELECTRIC/content/live/FAQS/86000/FA86427/es\\_ES/Modicon%20M340%20for%20Ethernet%20Communication%20Modules%20and%20Processors%20Unity%20Pro%2041.pdf](http://www2.schneider-electric.com/resources/sites/SCHNEIDER_ELECTRIC/content/live/FAQS/86000/FA86427/es_ES/Modicon%20M340%20for%20Ethernet%20Communication%20Modules%20and%20Processors%20Unity%20Pro%2041.pdf). [Último acceso: Junio 2016].
- [10] Schneider Electric. «Modicon M340 con Unity Pro. Módulos de Entradas/Salidas Analógicas,» Manual del Usuario, Febrero 2017. Available: [https://download.schneider-electric.com/files?p\\_enDocType=User+guide&p\\_File\\_Id=7433765549&p\\_File\\_Name=35011981\\_K01\\_000\\_13.pdf&p\\_Reference=35011981K01000](https://download.schneider-electric.com/files?p_enDocType=User+guide&p_File_Id=7433765549&p_File_Name=35011981_K01_000_13.pdf&p_Reference=35011981K01000) [Último acceso: 21 de Julio 2017]
- [11] Schneider Electric. Modicon M340 con Unity Pro. Módulos de Entradas/Salidas Binarias. Manual del Usuario, Julio 2012. Available: [http://www2.schneider-electric.com/resources/sites/SCHNEIDER\\_ELECTRIC/content/live/FAQS/169000/FA169573/es\\_ES/Unity%20v70%20-%20M340%20Modulos%20de%20entradas%20salidas%20digitales.pdf](http://www2.schneider-electric.com/resources/sites/SCHNEIDER_ELECTRIC/content/live/FAQS/169000/FA169573/es_ES/Unity%20v70%20-%20M340%20Modulos%20de%20entradas%20salidas%20digitales.pdf). [Último acceso:

15 de Julio 2017]

- [12] Schneider Electric. «Modicon M340 con Unity Pro. Procesadores, Bastidores, Módulos de Fuente de Alimentación,» Manual de Configuración, Julio 2016. Available: [http://www2.schneider-electric.com/resources/sites/SCHNEIDER\\_ELECTRIC/content/live/FAQS/28000/FA28444/es\\_ES/Unity\\_v4.1\\_M340\\_Procesadores\\_Bastidores\\_y\\_Fuentes\\_de\\_Alimentacion.pdf](http://www2.schneider-electric.com/resources/sites/SCHNEIDER_ELECTRIC/content/live/FAQS/28000/FA28444/es_ES/Unity_v4.1_M340_Procesadores_Bastidores_y_Fuentes_de_Alimentacion.pdf). [Último acceso: 15 de Julio 2017]
- [13] Schneider Electric, «Modicon M340 Automation Platform. Cordset colour codes in accordance with DIN 47100,» Manual de Configuración, Abril 2009. Available: [http://www2.schneider-electric.com/resources/sites/SCHNEIDER\\_ELECTRIC/content/live/FAQS/174000/FA174976/en\\_US/BMXFCW\\_01\\_03.pdf](http://www2.schneider-electric.com/resources/sites/SCHNEIDER_ELECTRIC/content/live/FAQS/174000/FA174976/en_US/BMXFCW_01_03.pdf). [Último acceso: 25 de Julio 2017]
- [14] Schneider Electric, «Unity Pro. Modalidades de Funcionamiento,» Manual de Configuración, Mayo 2010. Available: [http://www2.schneider-electric.com/resources/sites/SCHNEIDER\\_ELECTRIC/content/live/FAQS/33000/FA33733/es\\_ES/Unity%20v50%20-%20Modalidades%20de%20funcionamiento.pdf](http://www2.schneider-electric.com/resources/sites/SCHNEIDER_ELECTRIC/content/live/FAQS/33000/FA33733/es_ES/Unity%20v50%20-%20Modalidades%20de%20funcionamiento.pdf). [Último acceso: 30 de Junio 2017]
- [15] Schneider Electric. «Modicon M340 con Unity Pro. Módulo de conteo BMX EHC 0200. Manual de usuario,» Manual de Configuración, 2010. Available: [http://www2.schneider-electric.com/resources/sites/SCHNEIDER\\_ELECTRIC/content/live/FAQS/33000/FA33770/es\\_ES/Unity%20v50%20-%20Contaje%20BMXEHC0200.pdf](http://www2.schneider-electric.com/resources/sites/SCHNEIDER_ELECTRIC/content/live/FAQS/33000/FA33770/es_ES/Unity%20v50%20-%20Contaje%20BMXEHC0200.pdf). [Último acceso: 6 de Julio 2017]
- [16] SMC Corporation. «The Pneumatics Calalog,» Product Catalog, p. 1.221 – 2.8, 2001.
- [17] Sony Corporation, «Scara – Type High Speed Assembly Robot SRX – 4CH,» Operating Manual, Production Technolog Group, 1989.
- [18] Sony Corporation, «Scara – Type High Speed Assembly Robot SRX – 4CH,» User’s Manual (Electrical), Production Technolog Group, 1989
- [19] Sony Corporation, «Scara – Type High Speed Assembly Robot SRX – 4CH,» User’s Manual (Mechanical), Production Technolog Group, 1989
- [20] Velasco Martin Antonio Luis, «Integración de Alimentador de Bandejas a Célula de Fabricación Flexible Controlada por PLC,» Trabajo Fin de Grado, Universidad de Sevilla 2015.