



UNIVERSIDAD POLITÉCNICA SALESIANA SEDE GUAYAQUIL

CARRERA DE INGENIERÍA ELECTRÓNICA

PROYECTO TÉCNICO PREVIA A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO ELECTRÓNICO

TEMA:

**DISEÑO E IMPLEMENTACIÓN DE CONTROLADORES PID Y FUZZY
PARA LA PLANTA DE ALMACENAMIENTO AUTOMÁTICO DEL
SISTEMA FESTO MPS500 EN LA UNIVERSIDAD POLITÉCNICA
SALESIANA SEDE GUAYAQUIL.**

Autores:

**LUIGGI JEFFERSON CONSTANTE MARTÍNEZ
JEAN CARLO CALLE RODAS**

Tutor:

ING. BYRON LIMA CEDILLO MSc.

GUAYAQUIL – ECUADOR
2018

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA

Yo, Luigi Jefferson Constante Martínez y Jean Carlo Calle Rodas estudiantes de Ingeniería Electrónica de la Universidad Politécnica Salesiana, certificamos que los desarrollos, análisis y conclusiones de este trabajo son exclusiva responsabilidad de los autores.

Guayaquil, Julio del 2018

Luigi Jefferson Constante Martínez
C.I.: 0921690020

Jean Carlo Calle Rodas
C.I.:0930487020

CERTIFICADO DE CESIÓN DE DERECHOS

Mediante este certificado, se procede a ceder los derechos de propiedad a la Universidad Politécnica Salesiana, de acuerdo a su normativa institucional vigente.

Guayaquil, Julio del 2018

Luigi Jefferson Constante Martínez
C.I.: 0921690020

Jean Carlo Calle Rodas
C.I.: 0930487020

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Por medio de la presente doy constancia que los Sres. Luigi Jefferson Constante Martínez y Jean Carlo Calle Rodas han desarrollado y elaborado satisfactoriamente el proyecto final de titulación, que se ajusta a las normas establecidas por la Universidad Politécnica Salesiana, por tanto, autorizo su presentación para los fines legales pertinentes.

Ing. Byron Lima Cedillo MSc.

DIRECTOR DEL PROYECTO

DEDICATORIA

Quiero dedicar este proyecto de titulación primordialmente a Dios, porque siento que en todo momento está dentro de mí dándome las fuerzas necesarias de seguir adelante, de tener mucha perseverancia y decisión para poder alcanzar cada logro en mi vida, como lo es también de tener la oportunidad de culminar este gran proyecto.

A mi madre Ángela Martínez por seguir conmigo en cada paso que doy, brindándome sus consejos necesarios para así tomar las mejores decisiones en la vida.

A mi padre Luis Constante que durante mi trayectoria de profesión me ayudó muchísimo para tener la capacidad de culminar esta gran carrera con sus consejos y anécdotas contadas.

A mis abuelas por brindarme su cariño incondicional del estar a mi lado día y noche.

A mi novia Keila Bermudez, de tener siempre la predisposición al ayudarme con la culminación de este proyecto y permanecer junto a mi lado.

Al Ing. Byron Lima por brindarme todo su apoyo para la realización de este proyecto y formación profesional.

Este proyecto de titulación se lo quiero dedicar a todas y cada una de las personas que forman parte fundamental en mi vida que de alguna forma u otra están pendientes de este servidor, a aquellos que confiaron en mí ciegamente desde el momento en el que decidí cursar esta profesión, este proyecto también les pertenece.

Luigi Constante Martínez

DEDICATORIA

Dedico este proyecto de titulación en primer lugar a Dios por darme salud y el conocimiento necesario para la realización de este proyecto, por permitirme culminar esta etapa en mi formación profesional, a mis padres que han sido durante mi vida los pilares fundamentales tanto en principios, afecto y amor, a mi hermana por sus consejos y su apoyo incondicional, a mi tutor el Ing. Byron Lima por haberme guiado no solamente en este proyecto sino en mi formación académica y profesional, a don Martín y a Junior quienes amablemente nos ayudaron con la disponibilidad del laboratorio de fabricación flexible para la elaboración de este proyecto de titulación.

Jean Carlo Calle

AGRADECIMIENTO

Le doy gracias a Dios porque me permitió levantarme cada día con la motivación necesaria de poder presentar y culminar este proyecto de titulación a pesar de las grandes adversidades que tuvimos durante mi proceso profesional, hoy en día gracias a Dios se logró.

Gracias a mi madre Ángela Martínez por todos sus sacrificios que tuvo para yo poder haber ingresado y culminado esta gran profesión, te amo y gracias por todo madre.

Gracias a mi padre Luis Constante que a pesar de no estar siempre juntos de cualquier forma siempre me aportaste a esta profesión, te amo y te agradezco por seguir a mi lado también.

Gracias a mis abuelitas porque no dejaron que jamás deje esta gran profesión.

Gracias a ti mi hermana por todos tus deseos desde que inicie en esta profesión y hoy por hoy somos el gran orgullo de nuestros padres.

Gracias a mí amada Keila Bermudez por su consistencia en seguir adelante y a pesar de los altos y bajos siempre estás conmigo.

Agradezco también a mi tutor el Ingeniero Byron Lima Cedillo que me aportó muchos conocimientos para de esta forma poder realizar la culminación del proyecto con mucho éxito y por su experiencia brindada.

Gracias a la Universidad Politécnica Salesiana por permitir el desarrollo del proyecto, al Sr. Don Martin y Junior que nos dieron la habilitación del laboratorio para generar este dichoso proyecto.

Humildemente Gracias a Todos.

Luigi Constante Martínez

AGRADECIMIENTO

Agradezco a todas las personas que hicieron posible este logro personal, a mis padres, a mi hermana a mi tutor el ing. Byron Lima, a mis amigos y a los encargados de los laboratorios ya que si estuvieron presto para ayudarme en culminar este proyecto.

Jean Carlo Calle

RESUMEN

AÑO	ALUMNOS	DIRECTOR DEL PROYECTO TÉCNICO	TEMA DEL PROYECTO
2018	-CONSTANTE MARTÍNEZ LUIGGI JEFFERSON -CALLE RODAS JEAN CARLO	ING. BYRON LIMA CEDILLO MSc.	DISEÑO E IMPLEMENTACION DE CONTROLADORES PID Y FUZZY PARA LA PLANTA DE ALMACENAMIENTO AUTOMATICO DEL SISTEMA FESTO MPS500 EN LA UNIVERSIDAD POLITECNICA SALESIANA SEDE GUAYAQUIL.

Este presente proyecto de titulación, "DISEÑO E IMPLEMENTACIÓN DE CONTROLADORES PID Y FUZZY PARA LA PLANTA DE ALMACENAMIENTO AUTOMÁTICO DEL SISTEMA FESTO MPS500 EN LA UNIVERSIDAD POLITÉCNICA SALESIANA SEDE GUAYAQUIL", la idea principal de éste proyecto fue proporcionar controladores PID y FUZZY para el fácil procesamiento en las prácticas de laboratorios. Primero se logró obtener los pulsos del encoder en el motor M1 y en el motor M2 para poder transformar esos pulsos en distancia. Así se obtuvo los parámetros reales de cada motor, para con ello obtener la función de transferencia simulada y luego validar dicho modelo matemático con el modelo real del motor. Con el software MatLab en la herramienta sisotool se obtuvo el modelo simulado de los controladores PID y con ello se validó dicho modelo con el modelo real de la planta. Con la herramienta Toolbox Fuzzy Logic se procedió al diseño de los controladores Fuzzy de los motores M1 y M2.

Luego de ello se realizó la programación en el software Tia Portal V13, de los controladores PID y controladores FUZZY, finalmente se programó las prácticas propuestas en el Tia Portal V13.

PALABRAS CLAVES: Controladores, PID, FUZZY, Encoder, Motor, MatLab, Tia Portal V13

ABSTRACT

YEAR	STUDENTS	THECNICAL PROYECT MANEGER	PROYECT THEME
2018	-CONSTANTE MARTÍNEZ LUIGGI JEFFERSON -CALLE RODAS JEAN CARLO	ING. BYRON LIMA CEDILLO MSc.	DESING AND IMPLEMENTATION OF PID CONTROLLERS AND FUZZY FOR THE AUTOMATIC STORAGE PLANT OF THE FESTO MPS500 SYSTEM AT THE SALESIAN POLYTECHNIC UNIVERSITY GUAYAQUIL HEADQUEARTERS.

This present project of qualification, “DESIGN AND IMPLEMENTATION OF PID CONTROLLERS AND FUZZY FOR THE AUTOMATIC STORAGE PLANT OF THE FESTO MPS500 SYSTEM AT THE SALESIAN POLYTECHNIC UNIVERSITY GUAYAQUIL HEADQUEARTERS”, the main idea of this project was to provide PID and FUZZY controllers for easy processing in laboratory practices. First it was possible to obtain the pulses of the encoder in the M1 motor and the M2 motor in order to transform these pulses into distance. Thus, the real parameters of each motor were obtained, in order to obtain the simulated transfer function and then validate said mathematical model with the real model of the motor. With the MatLab software in the sisotool tool the simulated model of the PID controllers was obtained and with this the model was validated with the real model of the plant. With the Fuzzy Logic toolbox, the Fuzzy controllers of the M1 and M2 motors were designed. After that, the programming was done in the Tia Portal V13 software, the PID controllers and FUZZY controllers, finally the proposed practices were programmed in the Tia Portal V13.

KEY WORDS: Drivers, PID, FUZZY, Encoder, Motor, MatLab, Tia Portal V13

ÍNDICE GENERAL

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA	II
CERTIFICADO DE CESIÓN DE DERECHOS	III
CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN.....	IV
DEDICATORIA	V
AGRADECIMIENTO	VII
RESUMEN.....	IX
ABSTRACT	X
ÍNDICE GENERAL	XI
ÍNDICE FIGURAS.....	XIV
ÍNDICE TABLAS.....	XXI
INTRODUCCIÓN.....	1
1. EL PROBLEMA	2
1.1 Descripción del problema	2
1.2 Importancia y alcance	2
1.3 Delimitación del problema.....	2
1.3.1 Delimitación temporal:	2
1.3.2 Delimitación espacial:	2
1.3.3 Delimitación académica:	3
1.4 Objetivos.....	3
1.4.1 Objetivo general.....	3
1.4.2 Objetivos específicos.....	3
2. ESTADO DEL ARTE.....	3
2.1. Sabertooth 2x60	3
2.2. Motor DC SWMK tipo 404319.....	4
2.3. Encoder incremental.....	4
2.4. Sensor magnético FESTO	4
2.5. Electroválvula neumática	5
2.6. Fuente de alimentación.....	5
2.7. PLC Siemens simatic S7-300	6
2.8. Simatic NET CP343-1 Lean	6
2.9. MatLab	7
2.10. TIA Portal.....	7
2.11. NI LabView	7
2.12. NI ELVIS II.....	8
2.13. KEPServerEx.....	8
2.14. Controladores PID	8
2.15. Controladores fuzzy	9
2.16. Sistemas de almacenamiento automático en la industria.....	10
2.17. Descripción de la planta.....	10
3. MARCO METODOLÓGICO	11

3.1.	Modelo analítico de los motores DC M1 y M2	11
3.1.1.	Modelado del circuito eléctrico de la armadura	11
3.1.2.	Descripción matemática del sistema mecánico.....	12
3.1.3.	Análisis electromecánico del motor DC.....	13
3.1.3.1.	Parámetros experimentales en la planta.....	15
3.1.3.1.1.	Medición de la resistencia de armadura R_a	16
3.1.3.1.2.	Medición de la inductancia de armadura L_a	17
3.1.3.1.3.	Constante de la fuerza contra electromotriz k_b	18
3.1.3.1.4.	Constante del par electromagnético k_t	19
3.1.3.1.5.	Coeficiente de fricción viscosa B	20
3.1.3.1.6.	Movimiento de inercia equivalente del rotor J_m	20
3.2.	Conversión analógica de salida	22
3.3.	Conversión de la lectura del encoder.....	24
3.4.	Validación del modelo matemático de los motores DC M1 y M2.....	27
3.4.1.	Pruebas de validación de velocidad del motor DC M1	28
3.4.2.	Pruebas de validación de velocidad del motor DC M2	29
3.4.3.	Pruebas de validación de distancia del motor DC M1	31
3.4.4.	Pruebas de validación de distancia del motor DC M2	33
3.5.	Error absoluto y relativo de los motores DC M1 y M2	34
3.6.	Diseño de controladores PID de los motores DC M1 y M2.....	35
3.6.1.	Diseño del controlador P del motor DC M1	36
3.6.2.	Diseño del controlador PD del motor DC M1	39
3.6.3.	Diseño del controlador PI del motor DC M1	42
3.6.4.	Diseño del controlador PID del motor DC M1	44
3.6.5.	Diseño del controlador P del motor DC M2	47
3.6.6.	Diseño del controlador PD del motor DC M2	50
3.6.7.	Diseño del controlador PI del motor DC M2	52
3.6.8.	Diseño del controlador PID del motor DC M2	54
3.7.	Validación de controladores PID de los motores DC M1 y M2	57
3.7.1.	Prueba de validación del controlador P del motor DC M1	59
3.7.2.	Prueba de validación del controlador PD del motor DC M1.....	61
3.7.3.	Prueba de validación del controlador PI del motor DC M1	63
3.7.4.	Prueba de validación del controlador PID del motor DC M1.....	65
3.7.5.	Prueba de validación del controlador P del motor DC M2	67
3.7.6.	Prueba de validación del controlador PD del motor DC M2.....	68
3.7.7.	Prueba de validación del controlador PI del motor DC M2	70
3.7.8.	Prueba de validación del controlador PID del motor DC M2.....	72
3.8.	Diseño de controladores fuzzy de los motores DC M1 y M2	74
3.8.1.	Diseño del controlador fuzzy del motor DC M1	74
3.8.1.1.	Diseño de la entrada del controlador fuzzy del motor DC M1.....	74
3.8.1.2.	Diseño de la salida del controlador fuzzy del motor DC M1	75
3.8.2.	Diseño del controlador fuzzy del motor DC M2	77
3.8.2.1.	Diseño de la entrada del controlador fuzzy del motor DC M2.....	77
3.8.2.2.	Diseño de la salida del controlador fuzzy del motor DC M2	77
3.9.	Validación de controladores fuzzy de los motores DC M1 y M2.....	79
3.9.1.	Prueba de validación del controlador fuzzy del motor DC M1	80
3.9.2.	Prueba de validación del controlador fuzzy del motor DC M2	81
3.10.	Visualización de datos en tiempo real utilizando LabView	81
4.	GUÍA DE PRÁCTICAS DE LABORATORIO	102
4.1.	Práctica Nro. 1	102

4.2.	Práctica Nro. 2.....	109
4.3.	Práctica Nro. 3.....	113
4.4.	Práctica Nro. 4.....	124
4.5.	Práctica Nro. 5.....	129
4.6.	Práctica Nro. 6.....	134
4.7.	Práctica Nro. 7.....	143
4.8.	Práctica Nro. 8.....	152
5.	ANÁLISIS DE RESULTADOS.....	161
6.	CONCLUSIONES Y RECOMENDACIONES	165
7.	REFERENCIAS BIBLIOGRÁFICAS.....	166
8.	ANEXOS.....	168

ÍNDICE FIGURAS

Figura 1. Delimitación espacial del proyecto.....	2
Figura 2. Sabertooth 2x60.....	3
Figura 3. Motor DC de 24V	4
Figura 4. Encoder incremental	4
Figura 5. Sensor magnético	5
Figura 6. Electroválvula neumática de 5 vías, 3 posiciones	5
Figura 7. Fuente de alimentación sobremesa 24V DC	5
Figura 8. SIMATIC S7-300 CPU314C-2DP con entradas digitales y analógicas	6
Figura 9. Procesador de comunicaciones simatic NET CP 343-1 Lean.....	7
Figura 10. NI ELVIS II	8
Figura 11. Controladores PID.....	8
Figura 12. Diagrama de controladores Fuzzy	9
Figura 13. Proceso automático industrial con transelevadores	10
Figura 14. Elementos de la planta	11
Figura 15. Modelo de un circuito eléctrico de armadura del motor DC	12
Figura 16. Diagrama de bloques del motor DC.....	15
Figura 17. Motor DC M1 removido.....	15
Figura 18. Medición de resistencia de Armadura con multímetro.....	16
Figura 19. Medición de inductancia de armadura con NI Elvis II+.....	17
Figura 20. Medición de la frecuencia de los motores.	18
Figura 21. Medición del tiempo de freno en los motores.	21
Figura 22. Grafica del voltaje vs Porcentaje.	23
Figura 23. Ecuación de voltaje vs porcentaje en tia portal.	23
Figura 24. Des escalamiento de valores de la ecuación descrita.	24
Figura 25. Representación del des escalamiento final.	24
Figura 26. Función SFB COUNT	24
Figura 27. Configuración del canal 0 del encoder del motor DC M1.	25
Figura 28. Configuración del canal 1 del encoder del motor DC M1.	25
Figura 29. Impulsos vs Distancia del encoder motor DC M1.....	25
Figura 30. Conversión de datos enteros del encoder del motor DC M1.....	26
Figura 31. Ecuación de impulsos vs distancia del encoder del motor DC M1	26
Figura 32. Impulsos vs Distancia del encoder del motor DC M2.	26
Figura 33. Conversión de datos enteros del encoder del motor DC M2.....	27
Figura 34. Ecuación de impulsos vs distancia del encoder del motor DC M2.....	27
Figura 35. Generador de señales AFG-2112.....	27
Figura 36. Simulación del modelo de velocidad del motor DC M1	28
Figura 37. Señal cuadrada al motor DC M1.....	28
Figura 38. Comparación de velocidad con señal cuadrada del motor DC M1	28
Figura 39. Señal sinusoidal al motor DC M1.....	29
Figura 40. Comparación de velocidad con una señal sinusoidal del motor DC M1	29
Figura 41. Simulación del modelo de velocidad del motor DC M2	29
Figura 42. Señal cuadrada al motor DC M2.....	30
Figura 43. Comparación de velocidad con señal cuadrada del motor DC M2	30
Figura 44. Señal sinusoidal al motor DC M2.....	30
Figura 45. Comparación de velocidad con señal sinusoidal del motor DC M2	31
Figura 46. Simulación del modelo de distancia del motor DC M1	31
Figura 47. Señal cuadrada al motor DC M1.....	31
Figura 48. Comparación de distancia con señal cuadrada del motor DC M1	32
Figura 49. Señal sinusoidal al motor DC M1.....	32
Figura 50. Comparación de distancia con señal sinusoidal del motor DC M1	32
Figura 51. Simulación del modelo de distancia del motor DC M2	33
Figura 52. Señal cuadrada al motor DC M2.....	33

Figura 53. Comparación de distancia con señal cuadrada del motor DC M2	33
Figura 54. Señal sinusoidal al motor DC M2.....	34
Figura 55. Comparación de la distancia con señal sinusoidal del motor DC M2.....	34
Figura 56. Herramienta sisotool de Matlab	36
Figura 57. Esctructura de control P del motor DC M1	36
Figura 58. Lugar geométrico de las raíces del motor DC M1 con controlador P	37
Figura 59. Controlador P para el motor DC M1	37
Figura 60. Diagrama de bloques para el control P del motor DC M1	38
Figura 61. Configuración de la entrada paso a 10cm	38
Figura 62. Respuesta de entrada paso y salida del controlador P del motor M1	38
Figura 63. Esctructura de control PD del motor DC M1.....	39
Figura 64. Lugar geométrico de las raíces del motor DC M1 con controlador PD	39
Figura 65. Controlador PD para el motor DC M1.....	40
Figura 66. Diagrama de bloques para el control PD del motor DC M1	41
Figura 67. Configuración de la entrada paso a 10cm	41
Figura 68. Respuesta de entrada paso y salida del controlador PD del motor M1	41
Figura 69. Esctructura de control PI del motor DC M1	42
Figura 70. Lugar geométrico de las raíces del motor DC M1 con controlador PI.....	42
Figura 71. Controlador PI para el motor DC M1	43
Figura 72. Diagrama de bloques para el control PI del motor DC M1	44
Figura 73. Configuración de entrada paso a 10cm.....	44
Figura 74. Respuesta de entrada paso y salida del controlador PI del motor M1	44
Figura 75. Esctructura de control PID del motor DC M1.....	45
Figura 76. Lugar geométrico de las raíces del motor DC M1 con controlador PID	45
Figura 77. Controlador PID para el motor DC M1.....	45
Figura 78. Diagrama de bloques para el control PID del motor DC M1	47
Figura 79. Configuración de entrada paso a 10cm.....	47
Figura 80. Respuesta de entrada paso y salida PID del motor M1	47
Figura 81. Esctructura de control P del motor DC M2	48
Figura 82. Lugar geométrico de las raíces del motor DC M2 con controlador P.....	48
Figura 83. Controlador P para el motor DC M2	48
Figura 84. Diagrama de bloques para el control P del motor DC M2	49
Figura 85. Configuración de entrada paso a 10cm.....	49
Figura 86. Respuesta de entrada paso y salida del controlador P del motor M2	49
Figura 87. Esctructura de control PD del motor DC M2.....	50
Figura 88. Lugar geométrico de las raíces del motor DC M2 con controlador PD	50
Figura 89. Controlador PD para el motor DC M2.....	50
Figura 90. Diagrama de bloques para el control PD del motor DC M2	51
Figura 91. Configuración de entrada paso a 10cm.....	51
Figura 92. Respuesta de entrada paso y salida PD del motor DC M2	52
Figura 93. Esctructura de control PI del motor DC M2	52
Figura 94. Lugar geométrico de las raíces del motor DC M2 con controlador PI.....	52
Figura 95. Controlador PI para el motor DC M2	53
Figura 96. Diagrama de bloques para el control PI del motor DC M2	54
Figura 97. Configuración de entrada paso a 10cm.....	54
Figura 98. Respuesta de entrada paso y salida PI del motor DC M2.....	54
Figura 99. Esctructura de control PID del motor DC M2.....	55
Figura 100. Lugar geométrico de las raíces del motor DC M2 con controlador PID	55
Figura 101. Controlador PID para el motor DC M2.....	55
Figura 102. Diagrama de bloques para el control PID del motor DC M2	57
Figura 103. Configuración de entrada paso a 10cm.....	57
Figura 104. Respuesta de entrada paso y salida PID del motor DC M2	57
Figura 105. Función regulador continuo.	58
Figura 106. Configuración interna del controlador continuo (Siemens AG, 1996)	58
Figura 107. Variables de datos en KEPServerEX	58
Figura 108. Datos reales de controladores de motores DC M1 y M2.....	59

Figura 109. Configuración del controlador P del motor DC M1	59
Figura 110. Comparación del controlador P del motor DC M1	60
Figura 111. Diagrama de bloques del control P y zona muerta del motor DC M1	60
Figura 112. Comparación del controlador P para el motor DC M1	60
Figura 113. Configuración del controlador PD del motor DC M1	61
Figura 114. Diagrama de bloques del control PD del motor DC M1	62
Figura 115. Comparación del controlador PD del motor DC M1	62
Figura 116. Diagrama de bloques del control PD y zona muerta del motor DC M1	62
Figura 117. Comparación del controlador PD para el motor DC M1	63
Figura 118. Configuración del controlador PI del motor DC M1	64
Figura 119. Diagrama de bloques para el control PI del motor DC M1	64
Figura 120. Comparación del controlador PI del motor DC M1	64
Figura 121. Configuración del controlador PID del motor DC M1	66
Figura 122. Diagrama de bloques para el control PID del motor DC M1	66
Figura 123. Comparación del controlador PID del motor DC M1	66
Figura 124. Configuración del controlador P del motor DC M2	67
Figura 125. Comparación del controlador P del motor DC M2	67
Figura 126. Diagrama de bloques del control P y zona muerta del motor DC M2	68
Figura 127. Comparación del controlador P del motor DC M2	68
Figura 128. Configuración del controlador PD del motor DC M2	69
Figura 129. Diagrama de bloques para el control PD del motor DC M2	69
Figura 130. Comparación del controlador PD del motor DC M2	70
Figura 131. Diagrama de bloques del control PD y zona muerta del motor DC M2	70
Figura 132. Comparación del controlador PD para el motor DC M2	70
Figura 133. Configuración del controlador PI del motor DC M2	71
Figura 134. Diagrama de bloques del control PI del motor DC M2	71
Figura 135. Comparación del controlador PI del motor DC M2	72
Figura 136. Configuración del controlador PID del motor DC M2	73
Figura 137. Diagrama de bloques del control PID del motor DC M2	73
Figura 138. Comparación del controlador PID del motor DC M2	74
Figura 139. Herramienta fuzzy en MatLab	74
Figura 140. Diseño de entrada del controlador fuzzy del motor DC M1	75
Figura 141. Diseño de salida del controlador fuzzy del motor DC M1	75
Figura 142. Diagrama de bloques del control fuzzy del motor DC M1	76
Figura 143. Configuración de entrada paso a 60cm	76
Figura 144. Respuesta de entrada paso y salida fuzzy del motor DC M1	76
Figura 145. Diseño de entrada del controlador fuzzy del motor DC M2	77
Figura 146. Diseño de salida del controlador fuzzy del motor DC M2	78
Figura 147. Diagrama de bloques del control fuzzy del motor DC M2	78
Figura 148. Configuración de entrada paso a 41cm	79
Figura 149. Respuesta de entrada paso y salida fuzzy del motor DC M2	79
Figura 150. Bloque de función fuzzy	79
Figura 151. Variables de datos en KEPServerEX	80
Figura 152. Datos reales de controladores de motores DC M1 y M2	80
Figura 153. Configuración del error y fuzzy del motor DC M1	80
Figura 154. Comparación del controlador fuzzy del motor DC M1	80
Figura 155. Configuración del error y fuzzy del motor DC M2	81
Figura 156. Comparación del controlador fuzzy del motor DC M2	81
Figura 157. Inicio del proceso del sistema Scada en LabView	82
Figura 158. Pulsación de la botonera Start	82
Figura 159. Proceso iniciado de la planta en tia portal	82
Figura 160. Proceso de los motores M1 y M2 a su posición inicial	83
Figura 161. Motor M1 en la posición inicial en el sistema Scada	83
Figura 162. Motor M1 en la posición inicial en tia portal	84
Figura 163. Motor M1 detenido en la posición inicial en tia portal	84
Figura 164. Arranque del motor M2 en tia portal	84

Figura 165.	Posición inicial del motor M1	85
Figura 166.	Motor M2 en la posición inicial en el sistema Scada	85
Figura 167.	Motor M2 en la posición inicial en tia portal.....	86
Figura 168.	Frenado del motor M2 en la posición inicial en tia portal	86
Figura 169.	Desactivación del frenado del motor M1 en tia portal.....	86
Figura 170.	Arranque del motor M1 en tia portal	86
Figura 171.	Posición inicial motor M2 visualizado en la planta.....	87
Figura 172.	Motor M1 en posición de pieza de trabajo del sistema Scada	87
Figura 173.	Frenado del motor M1 en tia portal.....	87
Figura 174.	Condición de frenado del motor M1 en tia portal.....	88
Figura 175.	Arranque del motor M2 en tia portal	88
Figura 176.	Posición inicial del motor M2 visualizado en la planta.....	88
Figura 177.	Recolección del Motor M2 de la pieza en el sistema Scada	89
Figura 178.	Condición de frenado del motor M2 en tia portal.....	89
Figura 179.	Frenado del motor M2 en programa tia portal	89
Figura 180.	Tiempo de espera del solenoide del eje Y en tia portal.....	89
Figura 181.	Activación del solenoide del eje Y en tia portal	90
Figura 182.	Recolección de la pieza de trabajo y solenoide activado	90
Figura 183.	Motor M2 con el solenoide activado en el sistema Scada.....	91
Figura 184.	Motor M2 en la posición inicial en tia portal.....	91
Figura 185.	Frenado del motor M2 con la pieza de trabajo en tia portal	91
Figura 186.	Arranque del motor M1 a la posición inicial en tia portal	91
Figura 187.	Motor M2 con solenoide activado visualizado en la planta.	92
Figura 188.	Motor M2 con solenoide desactivado en el sistema Scada.....	92
Figura 189.	Detección del motor M1 en tia portal.....	93
Figura 190.	Desactivación del motor M1 con la pieza de trabajo en tia portal	93
Figura 191.	Desactivación del solenoide eje Y en tia portal	93
Figura 192.	Frenado del motor M1 en la posición inicial en tia portal	94
Figura 193.	Habilitador PID o Fuzzy en tia portal	94
Figura 194.	Motor M2 pieza de trabajo activa y solenoide desactivado	94
Figura 195.	Pulsado de la botonera Control Fuzzy en el sistema Scada	95
Figura 196.	Activación de la elección del controlador en tia portal.....	95
Figura 197.	Activación para el posicionamiento de la pieza de trabajo en tia portal.....	95
Figura 198.	Posición 1 para el almacenamiento en el sistema Scada	96
Figura 199.	Posición 1 para el almacenamiento de la pieza en tia portal	96
Figura 200.	Arranque del motor M1 con el control Fuzzy en el sistema Scada	97
Figura 201.	Activación del control Fuzzy del motor M1 en tia portal	97
Figura 202.	Motor M1 hacia la posición 1 visualizado en la planta.....	97
Figura 203.	Arranque del motor M2 hacia la posición 1 Sistema Scada	98
Figura 204.	Activación del control Fuzzy del motor M2 en tia portal	98
Figura 205.	Arranque inmediato del motor M2 para alcanzar el Set Point	98
Figura 206.	Almacenamiento de la pieza de trabajo en la posición 1 sistema Scada.....	99
Figura 207.	Almacenamiento de la pieza de trabajo en la posición 1 en tia portal.....	99
Figura 208.	Ubicación de la pieza de trabajo en la posición 1.....	100
Figura 209.	Pulsado de la botonera Control PID en el sistema Scada.....	100
Figura 210.	Activación y elección del controlador PID en tia portal.....	100
Figura 211.	Activación del controlador PID del Motor M1 en tia portal.....	101
Figura 212.	Activación del controlador PID del Motor M2 en tia portal.....	101
Figura 213.	PLC S7-314C-2 DP.....	102
Figura 214.	Sensores B10 y B11	102
Figura 215.	Sensores B14 y B15	103
Figura 216.	Sensores B12 y B13.	103
Figura 217.	Sensores B16 y B17.	103
Figura 218.	Motor M1 y encoder B1.....	103
Figura 219.	Motor M2 y encoder B2.....	104
Figura 220.	Variables del PLC inicial.	104

Figura 221. Variables del PLC final.....	104
Figura 222. Bloque de función conversión analógica.....	104
Figura 223. Inicio del programa activo.....	105
Figura 224. Entrada del sensor B10 activo.....	105
Figura 225. Sensor derecha B10 activado.....	105
Figura 226. Entrada del sensor B11 activo.....	105
Figura 227. Sensor derecha B11 activado.....	105
Figura 228. Entrada del sensor B12 activa en el programa.....	106
Figura 229. Sensor alto B12 activado.....	106
Figura 230. Entrada del sensor B13 activa en el programa.....	106
Figura 231. Sensor alto B13 activado.....	106
Figura 232. Salida del solenoide del gripper activa en el programa.....	107
Figura 233. Sensor alto B17 activado.....	107
Figura 234. Salida del solenoide del eje y activa en el programa.....	107
Figura 235. Sensor alto B15 activado.....	107
Figura 236. Conversión de encoder M1 en tia portal.....	107
Figura 237. Lectura del encoder B1 funcionando.....	108
Figura 238. Lectura del encoder B2 funcionando.....	108
Figura 239. Diagrama de bloques de los motores DC M1 y M2.....	109
Figura 240. Datos de los motores DC M1 y M2.....	111
Figura 241. Código de la función transferencia del motor DC M1.....	111
Figura 242. Función transferencia de velocidad del motor DC M1.....	111
Figura 243. Función transferencia de velocidad del motor DC M1.....	111
Figura 244. Código de la función transferencia del motor DC M2.....	112
Figura 245. Función transferencia de velocidad del motor DC M2.....	112
Figura 246. Función transferencia de velocidad del motor DC M2.....	112
Figura 247. Variables de los motores M1 y M2 con sus sensores.....	113
Figura 248. Conversión analógica del motor M1.....	114
Figura 249. Conversión analógica del motor M2.....	114
Figura 250. Conversión del encoder del motor M1.....	114
Figura 251. Conversión del encoder del motor M2.....	114
Figura 252. Bloque de función del contaje del encoder B1 del motor M1.....	115
Figura 253. Bloque de función del contaje del encoder B2 del motor M2.....	115
Figura 254. Conexión Ethernet al PC y PLC S7-314C 2DP.....	116
Figura 255. Asignación de la IP en el módulo de comunicación.....	116
Figura 256. Carga de la configuración hardware al PLC-314C 2DP.....	117
Figura 257. Carga del programa en Tia Portal vía Ethernet.....	117
Figura 258. Asignación de la IP al PLC-314C 2DP.....	117
Figura 259. Creación del programa en KEPServerEX.....	118
Figura 260. Creación del canal PRÁCTICA 3.....	118
Figura 261. Seleccionamos el PLC-314C 2DP.....	118
Figura 262. Selección de la IP para la comunicación con el PLC-314C 2DP.....	118
Figura 263. Sumario de la configuración del canal PRÁCTICA 3.....	119
Figura 264. Canal PRÁCTICA 3 creado.....	119
Figura 265. Creación del dispositivo MONITOREO.....	119
Figura 266. Modelo del PLC empleado.....	119
Figura 267. Colocación de la IP.....	120
Figura 268. Configuración del Puerto.....	120
Figura 269. Sumario de la configuración del dispositivo MONITOREO.....	120
Figura 270. Canal y dispositivo configurados.....	121
Figura 271. Configuración de las variables de la Práctica 3.....	121
Figura 272. Motor DC M1 en la posición inicial.....	121
Figura 273. Detección y frenado del motor DC M1 por parte del sensor B10.....	121
Figura 274. Motor DC M1 del Set Point en el sistema Scada.....	122
Figura 275. Detección y frenado del motor DC M1 del Set Point.....	122
Figura 276. Motor DC M2 en la posición inicial sistema Scada.....	122

Figura 277. Detección y frenado del motor DC M2 por parte del sensor B12.....	122
Figura 278. Motor DC M2 en el Set Point del sistema Scada.....	123
Figura 279. Detección y frenado del motor DC M2 por parte del Set Point.....	123
Figura 280. Función sisotool de Matlab en la practica 4.....	124
Figura 281. Control de arquitectura PID del motor DC M2.....	124
Figura 282. Configuración del compensador control PID en la práctica 4.....	125
Figura 283. Configuración de la gráfica PID en lazo cerrado en la práctica 4.....	125
Figura 284. Ajuste del controlador PID en la práctica 4.....	126
Figura 285. Gráfica del controlador PID motor DC M2 eje vertical.....	126
Figura 286. Diagrama de bloques del motor DC M2.....	127
Figura 287. Gráfica del motor DC M2.....	128
Figura 288. Función sisotool en la práctica 5.....	129
Figura 289. Control de arquitectura PID del motor DC M1 en la práctica 5.....	129
Figura 290. Configuración del compensador PID en la práctica 5.....	130
Figura 291. Configuración de la gráfica en lazo cerrado en la práctica 5.....	130
Figura 292. Ajuste del controlador PID en la práctica 5.....	131
Figura 293. Gráfica del controlador PID del motor DC M1 práctica 5.....	131
Figura 294. Diagrama de bloques del motor DC M1 práctica 5.....	132
Figura 295. Gráfica del motor DC M1 práctica 5.....	133
Figura 296. Función Fuzzy en la práctica 6.....	134
Figura 297. Creación de entrada para controlador difuso del motor DC M2.....	134
Figura 298. Función de membresía MN2 del motor DC M2.....	135
Figura 299. Función de membresía N2 del motor DC M2.....	135
Figura 300. Función de membresía C2 del motor DC M2.....	136
Figura 301. Función de membresía P2 del motor DC M2.....	136
Figura 302. Función de membresía MP2 del motor DC M2.....	137
Figura 303. Creación de la salida para el controlador difuso del motor DC M2.....	137
Figura 304. Función de membresía MNeg2 del motor DC M2.....	138
Figura 305. Función de membresía Neg2 del motor DC M2.....	138
Figura 306. Función de membresía Cero2 del motor DC M2.....	139
Figura 307. Función de membresía Pos2 del motor DC M2.....	139
Figura 308. Función de membresía MPos2 del motor DC M2.....	140
Figura 309. Reglas de correspondencia del motor DC M2.....	140
Figura 310. Archivo .fis del diseño del controlador Fuzzy del motor DC M2.....	141
Figura 311. Diagrama del control Fuzzy del motor DC M2.....	141
Figura 312. Función transferencia control Fuzzy del motor DC M2.....	141
Figura 313. Función Fuzzy en la práctica 7.....	143
Figura 314. Creación de la entrada para el controlador difuso del motor DC M1.....	143
Figura 315. Función de membresía MN del motor DC M1.....	144
Figura 316. Función de membresía N del DC M1.....	144
Figura 317. Función de membresía C del DC M1.....	145
Figura 318. Función de membresía P del DC M1.....	145
Figura 319. Función de membresía MP del motor DC M1.....	146
Figura 320. Creación de la salida para el controlador difuso del motor DC M1.....	146
Figura 321. Función de membresía MNeg del motor DC M1.....	147
Figura 322. Función de membresía Neg del motor DC M1.....	147
Figura 323. Función de membresía Cero del motor DC M1.....	148
Figura 324. Función de membresía Pos del motor DC M1.....	148
Figura 325. Función de membresía MPos del motor DC M1.....	149
Figura 326. Reglas de correspondencia del controlador difuso del motor DC M1.....	149
Figura 327. Archivo .fis del motor DC M1.....	150
Figura 328. Diagrama de bloques del control Fuzzy del motor DC M1.....	150
Figura 329. Función transferencia del motor DC M1.....	150
Figura 330. Bloque de función PID del motor M1 inicio.....	152
Figura 331. Bloque de función PID del motor M1 final.....	153
Figura 332. Bloque de función CONT_C para el controlador PID del motor M2.....	153

Figura 333. Error Fuzzy M1 y el controlador Fuzzy M1	154
Figura 334. Error Fuzzy M2 y el controlador Fuzzy M2	154
Figura 335. Elección del sistema PC para la aplicación HMI	154
Figura 336. Módulo de comunicación Profinet a seleccionar	155
Figura 337. Módulo de comunicación insertado en el sistema PC	155
Figura 338. Direcciones de red de cada módulo	156
Figura 339. Conexión y sincronización de los dispositivos.	156
Figura 340. Creación de la aplicación HMI en la práctica 8.....	157
Figura 341. Configuración pulsar del pulsador.	157
Figura 342. Configuración soltar del pulsador	158
Figura 343. Configuración del indicador de luz.....	158
Figura 344. Configuración del indicador de la ficha.....	158
Figura 345. Configuración del indicador de la ficha en el estante	159
Figura 346. Configuración de la variable para el movimiento eje X del motor M1	159
Figura 347. Configuración de los indicadores numéricos	160
Figura 348. Tabla de variables de la aplicación HMI en Tia Portal.....	160
Figura 349. Comparación del P vs fuzzy del motor DC M1	161
Figura 350. Comparación del PD vs fuzzy del motor DC M1	161
Figura 351. Comparación del PI vs fuzzy del motor DC M1	162
Figura 352. Comparación del PID vs fuzzy del motor DC M1	162
Figura 353. Comparación del P vs fuzzy del motor DC M2	163
Figura 354. Comparación del PD vs fuzzy del motor DC M2	163
Figura 355. Comparación del PI vs fuzzy del motor DC M2	164
Figura 356. Comparación del PID vs fuzzy del motor DC M2	164

ÍNDICE TABLAS

Tabla 1. Medición experimental de resistencia R_a del motor DC M1	16
Tabla 2. Medición experimental de resistencia R_a del motor DC M2	16
Tabla 3. Medición experimental de inductancia L_a del motor DC M1	17
Tabla 4. Medición experimental de inductancia L_a del motor DC M2	17
Tabla 5. Medición experimental de la constante de fuerza contra electromotriz k_b del motor DC M1	19
Tabla 6. Medición experimental de la constante de fuerza contra electromotriz k_b del motor DC M2	19
Tabla 7. Medición experimental de la constante mecánica del motor DC M1	21
Tabla 8. Medición experimental de la constante mecánica del motor DC M2	22
Tabla 9. Determinación del momento de inercia del motor DC M1	22
Tabla 10. Determinación del momento de inercia del motor DC M2	22
Tabla 11. Obtención del error absoluto, relativo con una señal cuadrada y seno del motor DC M1	34
Tabla 12. Obtención del error absoluto, relativo con una señal cuadrada y seno del motor DC M2	35
Tabla 13. Obtención del error absoluto, relativo con una señal cuadrada y seno del motor DC M1	35
Tabla 14. Obtención del error absoluto, relativo con una señal cuadrada y seno del motor DC M2	35

INTRODUCCIÓN

Este proyecto presenta la aplicación de controladores PID y Fuzzy a los motores DC M1 y M2 implementados de fábrica en la planta, con la idea de poder verificar que tipo de controlador es mejor tratándose del tiempo, es decir que tanto tiempo se tarda en alcanzar totalmente dicho controlador a cada uno de los motores mencionados, para esto se logró obtener una serie de datos tanto reales como simulados para que, mediante los Software utilizados generen las comparaciones respectivas de los controladores y así observar que tipo de controlador se comporta de la mejor manera.

Se logró implementar un visualizador en tiempo real de la planta para de esta manera poder verificar como actúan los controladores PID y Difusos en los motores de dicha planta, este visualizador se lo realizo en el Software LabView y contiene dichas variables en donde se observa mejor el valor del encoder de cada motor en sus respectivos ejes.

Este proyecto contiene una serie de prácticas para que los estudiantes de esta carrera puedan entender de una mejor manera el funcionamiento y aplicación que se le dio a este proyecto de titulación.

Generando motivación de seguir mejorando aquella planta Festo MPS500 de Almacenamiento Automático del Sistema, este proyecto impulsa a los estudiantes para que puedan diseñar e implementar otros tipos de controladores para un mejor funcionamiento de dicha planta.

1. EL PROBLEMA

1.1 Descripción del problema

La planta de Almacenamiento Automático del Sistema Festo MPS500 que se encuentra en el laboratorio de fabricación flexible, cuenta con una programación en SIMATIC V5.5 (bajo usuario Administrador), pero actualmente se está utilizando un software más actualizado como es TIA PORTAL V13. La programación Demo de la planta limita la capacidad de almacenamiento del equipo. El control actual del movimiento vertical y horizontal está basado en comparadores y no en controladores continuos, haciendo trabajar a los actuadores de forma intermitente (on/off).

1.2 Importancia y alcance

Por medio de este proyecto se dará a conocer la importancia de manejar la utilización de estos sistemas en los procesos industriales y la implementación de controladores en procesos de manufactura. Este proyecto de titulación impulsará el crecimiento de prácticas de las materias de especialización en Sistemas Industriales tales como Automatización Industrial II, Redes de Computadoras III, Comunicaciones, Electiva I e Informática Industrial.

1.3 Delimitación del problema

1.3.1 Delimitación temporal:

Este proyecto de titulación se lo logro finalizar en el año 2018, teniendo una duración de 16 meses, la cual dentro de ese tiempo se pudo ajustar parámetros antes de la sustentación.

1.3.2 Delimitación espacial:

Dicho proyecto de titulación estará implementado en el Laboratorio de Fabricación Flexible en la planta de Almacenamiento Automático del Sistema Festo en el bloque B de Eléctrica-Electrónica de la Universidad Politécnica Salesiana Sede Guayaquil Campus Centenario.



Figura 1. Delimitación espacial del proyecto.

1.3.3 Delimitación académica:

Académicamente en el proyecto de titulación se contribuyó con todo el aprendizaje obtenido durante la carrera de Ingeniería Electrónica especialmente con bases en Sistemas Industriales, ya que este proyecto nos conlleva a tener una idea fundamental de lo que sería implementar este tipo de controladores en plantas industriales.

La implementación de la tarjeta electrónica Sabertooth 2x60, nos permite realizar el enlace con los motores de la planta para poder generar los respectivos controladores PID y Fuzzy.

Este proyecto estará disponible para los estudiantes de la Universidad Politécnica Salesiana de la Sede de Guayaquil en las carreras de Ingenierías.

1.4 Objetivos

1.4.1 Objetivo general

Diseñar e Implementar controladores PID y Fuzzy a la planta de almacenamiento automático del Sistema Festo MPS500 para el control de posición.

1.4.2 Objetivos específicos

- Obtener la función de transferencia de los motores a utilizar.
- Diseñar los controladores PID y Fuzzy para ambos ejes (vertical y horizontal) de la planta.
- Comparar las características de los controladores PID con los controladores Fuzzy (PID vs Fuzzy) aplicados a la planta.
- Implementar un visualizador de datos en tiempo real de la planta utilizando LabView.
- Elaborar un manual de 8 prácticas para el uso correcto de la planta.

2. ESTADO DEL ARTE

2.1. Sabertooth 2x60

La Sabertooth 2x60, es una tarjeta electrónica de muy buena eficiencia para el uso de controladores de motores dual. En temas de robóticas son muy utilizados para robots de hasta 1000 libras. Esta tarjeta electrónica puede proporcionar la conexión de hasta dos motores DC con 60A en cada uno, también logra alcanzar corrientes de 120A pico por canal en cada motor durante pocos segundos. Contiene una protección de sobre corrientes la cual impide destruir al conductor de grandes conexiones en cuanto a los motores. (Robotica-dec, 2018)



Figura 2. Sabertooth 2x60. (Dimension Engineering LLC, 2004)

2.2. Motor DC SWMK tipo 404319

Es un motor de corriente continua con una tensión nominal de 24V, tiene una velocidad sin carga de 70 revoluciones por minuto [rpm], una relación de engranaje de 55:1, contiene una resistencia de armadura de dos tipos una con 2 cuchillas de 420mOhm y otra con 4 cuchillas de 350mOhm, en su interior conlleva material de engranaje plástico y tiene un peso de 1.2 [kg]. (Smolka-berlin.de, s.f.)



Figura 3. Motor DC de 24V (Smolka-berlin.de, s.f.)

2.3. Encoder incremental

Es un encoder que contiene una pequeña tarjeta electrónica en donde por medio de dos sistema de transducción incremental incluidos generan en uno el conteo de pulsos del motor y en el segundo sistema de transducción incremental determina la dirección del movimiento del motor por la cual trabaja con una tensión de 24V. Estos tipos de encoder incrementales tienen un recorrido por rotación de 63mm, llevando a tener una cantidad de pulsos por rotación de 30 impulsos y una relación de engranaje del 55:1. (Eckart, 2002)



Figura 4. Encoder incremental

2.4. Sensor magnético FESTO

Es un sensor que actúa de forma normalmente abierta por la cual dentro de su funcionamiento industrial se activa por medio de un imán que lo contiene internamente y para procesar su dato envía la información a la entrada del PLC. (Eckart, 2002)



Figura 5. Sensor magnético

2.5. Electroválvula neumática

Las electroválvulas neumáticas son válvulas electromecánicas accionadas generalmente por medio de aire comprimido la cual distribuye su flujo de aire en una planta industrial o circuito industrial cerrado llegando así a regular el paso del flujo como también frenar el recorrido del aire industrial.

Para la planta de accionamiento automático MPS500 tiene una electroválvula neumática de tipo 5 vías y 3 posiciones por lo cual contiene una posición central adicional que puede llegar a ser utilizada como centro abierto, cerrado o a presión. (INTOR, 2013)

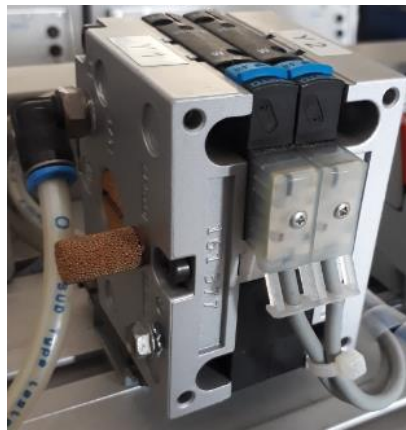


Figura 6. Electroválvula neumática de 5 vías, 3 posiciones

2.6. Fuente de alimentación

Es una fuente de alimentación usada normalmente sobremesa, tiene una tensión de entrada de 85 a 265 voltios AC en 47 a 63 Hercios y una tensión de salida de 24 voltios DC que lo protege de un cortocircuito generado. Su corriente máxima de salida de la fuente de alimentación es de 4.5 Amperios. (Festo, 2018)



Figura 7. Fuente de alimentación sobremesa 24V DC (Festo, 2018)

2.7. PLC Siemens simatic S7-300

Es un controlador lógico programable por sus siglas (PLC), en donde puede brindar soluciones a grande procesos industriales de manufacturas llevándolo así a la innovación de la tecnología y fabricación eficaz en cuanto a sistemas de automatización universal, tiene la capacidad de lograr varias soluciones ópticas en aplicaciones industriales con estructuras centralizadas.

Poseen la conexión interfaz de módulos Ethernet/Profinet con la integración de funciones tecnológicas.

Simatic S7-300 tiene una configuración en forma modular en donde no existe ninguna asignación "slots" en módulos periféricos.

Posee la opción de trabajar con una Micro Memory Card, la cual permite reducir costos en la parte económica a la hora de realizar un mantenimiento como en la parte de servicio técnico. De la misma forma la utilización de la Micro Memory Card, permite la actualización del programa o poder consultar datos técnicos mientras su funcionamiento.

Dentro la estandarización que presenta este Simatic S7-300 integra funciones de seguridad y control autómatas de movimiento.

Contiene capacidad de 192 Kbytes de memoria y por operación de bytes genera 0.06 microsegundos. (Siemens AG, 2018)



Figura 8. SIMATIC S7-300 CPU314C-2DP con entradas digitales y analógicas (Elektroskandia Sverige AB, 2018)

2.8. Simatic NET CP343-1 Lean

Este tipo de modulo nos permite realizar el enlace entre el Simatic S7-300 y el módulo de comunicaciones Simatic Net por medio del puerto Ethernet que tiene integrado en el módulo 2 Switch para el respectivo enlace.

También llamado módulo Profinet tiene la capacidad de poder enlazar más módulos de funciones dando así un mejor ahorro en ancho de tiempo para varias aplicaciones bastantes flexibles. Posee una capacidad muy segura al envío de datos por medio de conectores de tipo IE, FC y RJ45.

Por medio de la web puede generar diagnósticos, consultando varios ajustes desde una estación a otra, a nivel de aplicaciones de sistemas industriales o también verificar los estados de conexión en red de búfer en los módulos de rack en la cual esté conectado cierto CPU, cabe recalcar que la verificación de datos mediante la

web solo permite leer datos de lectura en las estaciones que se encuentren enlazadas por medio de la red misma. (Siemens AG, 2018)



Figura 9. Procesador de comunicaciones simatic NET CP 343-1 Lean (VENANDCOM, s.f.)

2.9. MatLab

Es un software de gran nivel en donde podemos generar programas con algoritmos teniendo la posibilidad de visualizar dentro de su estudio datos técnicos que podemos calcular, usa programaciones de lenguajes como los de C++ y Fortran. Este software posee una gran variedad de aplicaciones internas como lograr algún proceso de señales, lograr también el diseño mediante sistemas de control, modelado matemáticos, y en cuanto a las telecomunicaciones también puede generar procesos de señales en comunicaciones, como de la misma forma puede ser usada para ámbitos de administrativos.

MatLab logra enlazar los códigos de programación con otros tipos de lenguajes para poder ser distribuidos con otras aplicaciones que se desarrolló con este software. (Intercambios Virtuales, 2015)

2.10. TIA Portal

Es un software totalmente para la integración a la automatización de procesos industriales con grandes potenciales de programación, logra reducir todos los procesos industriales con un fácil procesamiento en operaciones de plantas.

Su forma de interactuar mediante el interfaz de usuario llega hacer totalmente fácil de usarlo por las funciones que presenta y su extensa transparencia de datos, ya que logran adaptarse sin ningún inconveniente en cuanto a los proyectos de sistemas automatizados. (Siemens AG, 2018)

2.11. NI LabView

Es un software de desarrollo básicamente para la utilización de desarrollos integrados con un lenguaje de programación en donde se puede generar flujos de datos en vez de varios códigos dentro de una línea secuencial, de esta manera LabView permite al usuario interactuar la visión del proceso mediante la supervisión de varios diseños, esto a su vez implica que el lenguaje que maneja este software impide preocuparnos por algún punto o coma dentro de su código de programación y ganando así tiempo para la resolución de procesos. (National Instruments Corporation, 2018)

2.12. NI ELVIS II

Es un instrumento de ingeniería para prácticas experimentales, en donde tiene la habilidad de interactuar con varios instrumentos de medición de laboratorio. Puede conectarse por medio de USB para el interfaz con la PC, la cual de la misma forma permite implementar circuitos electrónicos en su ProtoBoard que viene incluido de fábrica. (National Instruments Corporation, 2018)



Figura 10. NI ELVIS II (National Instruments Corporation, 2018)

2.13. KEPServerEx

Es un tipo de Software que posee la capacidad de generar y enviar datos únicos para aplicaciones de sistemas industriales, además puede darle toda la facilidad al usuario de conectarse mediante una interfaz intuitiva en donde el operador podrá administrar y monitorear varios dispositivos en un mismo control industrial.

La manera en que este tipo de Software envía datos es mediante la interoperabilidad en cuanto a los estándares de automatización OPC, que permiten así darle la respectiva comunicación de datos a dicho sistema industrial. (PTC Inc, 2018)

2.14. Controladores PID

Es un tipo de controladores proporcional, integral y derivativo, en donde normalmente son aplicados a mecanismo de retroalimentación a lazo cerrado muy usados para aplicaciones industriales. Internamente en algún proceso industrial el control PID actúa de forma en el que se introduce un error calculado en base a la salida requerida menos la salida a obtenerse y ese resultado de la salida es utilizado como la entrada al proceso que se requiere ejecutar.

En el siguiente esquema se observa la implementación del controlador PID sobre un proceso. (Tangent LLC, 2018)

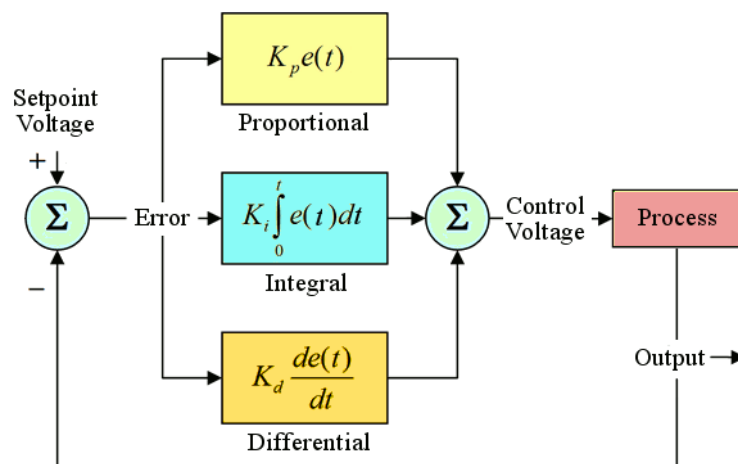


Figura 11. Controladores PID. (8760 Engineering, 2018)

Según Mazzone, Este tipo de controladores PID también tienen varias estructuras de manejo por la cual se describen de la siguiente manera:

Proporcional: Se basa en que es totalmente proporcional al error en cuanto al proceso industrial.

Integral: Se basa en que es totalmente proporcional al error que se va acumulando en un proceso industrial.

Proporcional e integral: En conjunto de los dos controladores, es una forma usualmente sencilla de usarse en procesos de industrias.

Proporcional y derivativa: Incluyendo la parte Derivativa genera que la acción del control sea más rápida en el proceso.

Proporcional, integral y derivativa: Permite que cierto control industrial muestre las ventajas de las anteriores descritas.

Mazzone indica que los tipos de estructura de estos controladores P, I, PI, PD y PID son muy simples ya que por ello también crea cierta debilidad en el proceso, dándole así limitaciones en cuanto a los rangos de aplicaciones donde se puede llegar a controlar. (Mazzone, 2002)

2.15. Controladores fuzzy

León explica que los controladores Fuzzy se definen como varios elementos por medio de un mismo conjunto de reglas, las cuales pueden combinarse con el lenguaje IF-THEN, que mediante el concepto lingüístico conlleva a una cuantificación de varios controladores difusos.

En cuanto a las aplicaciones industriales las reglas de este tipo de controladores llevan a ser procesados difusamente su dicha información ingresada.

Existen dos tipos de controladores difusos que son usados normalmente en aplicaciones o procesos industriales las cuales son:

Controladores fuzzy de takagi-sugeno: Se procesan con valores reales de acuerdo a sus entradas o salidas que contenga el proceso en sí.

Controladores fuzzy de mandani: Se procesa que a la entrada lleva un fusificador y a su salida dispone de un defusificador. (LEÓN PÉREZ, 2014)

Para este proyecto se utilizó el método del promedio ponderado que se procesa a partir de la salida de las funciones de membresía de nuestro controlador, la cual consiste en tomar varios puntos a las funciones de membresía para de esta forma obtener un promedio ponderado.

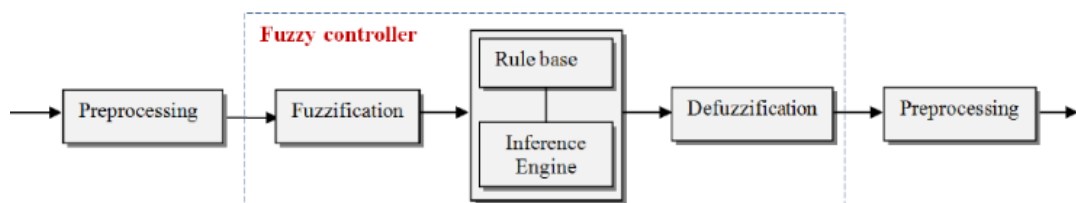


Figura 12. Diagrama de controladores Fuzzy (Shamshiri & Wan Ismail, 2012)

2.16. Sistemas de almacenamiento automático en la industria

En procesos industriales existen varios sistemas de almacenamiento automático como los Transelevadores que recogen y depositan sus unidades mediante una paleta de carga para ser ubicada en su posición, dentro de este tipo de procesos pueden haber también otras soluciones como:

- Transelevadores de una sola paleta
- Transelevadores de doble paleta
- Transelevadores de descanso sobre soportes (NOEGA System, 2016)



Figura 13. Proceso automático industrial con transelevadores. (NOEGA System, 2016)

2.17. Descripción de la planta

Ubicación de los elementos principales de la planta:

- PLC S7-314C 2DP
- Sensor B10 derecho del eje X
- Sensor B11 izquierdo del eje X
- Sensor B12 arriba del eje Y
- Sensor B13 abajo del eje Y
- Sensor B14 anterior del eje Y
- Sensor B15 delantero del eje Y
- Sensor B16 del gripper cerrado
- Sensor B17 del gripper abierto
- Motor M1 del eje X
- Encoder B1 del eje X
- Motor M2 del eje Y
- Encoder B2 del eje Y
- Botoneras de Star, Paro y Reset
- Electroválvula
- Flujo de aire comprimido

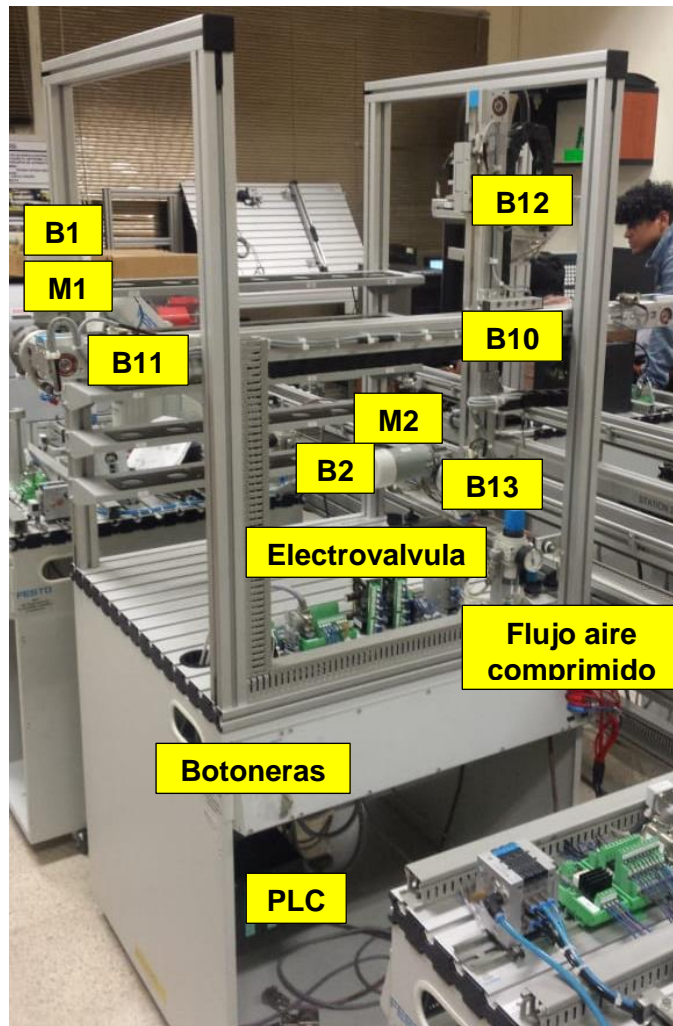


Figura 14. Elementos de la planta

3. MARCO METODOLÓGICO

Para realizar este proyecto de titulación se procedió a reconocer las variables que se tendrán presentes tanto de entrada como de salida.

Se dispuso de un controlador PID y un controlador Fuzzy para los dos motores existentes, tanto para el eje vertical como para el eje horizontal.

Se realizó las correspondientes graficas en el Software Matlab para comparar la efectividad de los controladores. También se realizó la obtención y visualización de datos de la planta en tiempo real a través del programa Labview.

3.1. Modelo analítico de los motores DC M1 y M2

Para obtener los modelos matemáticos de los motores se los realizó por medio de formulación matemática y tomando en cuenta los métodos experimentales para conocer los diferentes parámetros que integran a los motores.

3.1.1. Modelado del circuito eléctrico de la armadura

El modelo de un motor DC, donde se puede observar el circuito eléctrico que está conformado por la resistencia R_a y el inductor L_a .

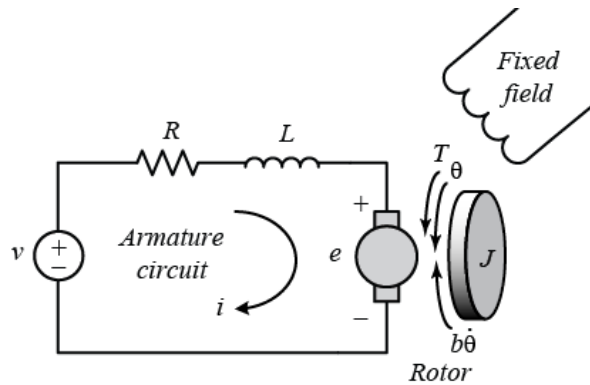


Figura 15. Modelo de un circuito eléctrico de armadura del motor DC (*CONTROL TUTORIALS FOR MATLAB & SIMULINK, s.f.*)

El circuito RL de la figura 16 está desde el arranque del motor hasta que el motor llega a su estado estacionario, en este punto es donde se efectúa el análisis transitorio del sistema.

Por medio de la ley de voltajes de Kirchhoff y teoría de circuitos eléctricos, las ecuaciones para la equivalencia de un motor DC son expresadas como:

$$V_a = V_{Ra} + V_{La} + e_a$$

$$V_a = L_a \left(\frac{di_a}{dt} \right) + i_a R_a + e_a \quad (3.1)$$

Donde:

V_a : Voltaje de armadura.

$L_a \left(\frac{di_a}{dt} \right)$: Caída de voltaje en la inductancia.

$i_a R_a$: Caída de voltaje en la resistencia.

e_a : Fuerza contra electromotriz desarrollada en la armadura.

La ecuación del voltaje de armadura (3.1) relaciona la fuerza contra electromotriz, debido a que al suministrar corriente produce una fuerza contra electromotriz en el inducido a causa del giro del motor y los cortes de líneas de flujo magnético. La polaridad de la fuerza contra electromotriz es contraria al voltaje del inducido.

3.1.2. Descripción matemática del sistema mecánico

Cuando a un mecanismo rotatorio se implementa como motor, entrega energía eléctrica a sus bobinados y a las fuerzas electromagnéticas crean un par mecánico.

Empleando las leyes de Newton se puede delinear este par mecánico, conociendo que el par equivalente que procede sobre un eje siempre es cero, explicando todas las componentes, de acuerdo a una dirección de referencia común.

“entonces el equilibrio o balance del par de la armadura que acelera una carga en la dirección del giro del rotor, sometida a la pérdida de par debido al rozamiento mecánico, por la fricción del aire y la pérdida en el hierro, es el siguiente”.

$$\Sigma T = T_{mec} - T_{acel} - T_{perd} - T_L = 0$$

$$T_{mec} = T_{acel} + T_{perd} + T_L$$

$$T_{mec} = J \left(\frac{d\omega}{dt} \right) + B\omega + T_L \quad (3.2)$$

Donde:

$J \left(\frac{d\omega}{dt} \right)$: Par mecánico producido por el momento de inercia J, éste término desaparece cuando la aceleración es cero y se presenta en condiciones de estado permanente.

$B\omega$: Par mecánico producido por el coeficiente de fricción viscosa B producido en los valeros o chumaceras, así como en cualquier otra zona donde existe algún rozamiento, este término permanece constante si ω lo es también.

T_L : Par mecánico producido por la carga externa aplicada al rotor del mecanismo.

El momento de inercia J está constituido por el momento de inercia J_m al hacer girar el inducido del motor y el momento de inercia de la carga J_L , ya que la carga gira a la misma velocidad que el motor.

$$J = J_m + J_L \quad (3.3)$$

De la misma forma el coeficiente de fricción viscosa B es el resultado del rozamiento de las partes mecánicas B_m y el rozamiento de la carga conectada al eje del motor B_L .

$$B = B_m + B_L \quad (3.4)$$

3.1.3. Análisis electromecánico del motor DC

Haciendo referencia a las ecuaciones mencionadas, se pueden efectuar las ecuaciones que conforman el modelado del sistema como:

$$T_{elec} = k_t i_a \quad (3.5)$$

$$e_a = k_b \omega \quad (3.6)$$

$$V_a = L_a \left(\frac{di_a}{dt} \right) + i_a R_a + e_a \quad (3.7)$$

$$T_{mec} = J \left(\frac{d\omega}{dt} \right) + B\omega + T_L \quad (3.8)$$

Donde:

T_{elec} : Par electromagnético generado (N.m)

i_a : Corriente de armadura (A).

e_a : Fuerza contra electromotriz inducida (V).

ω : Velocidad angular del motor (rad/s).

V_a : Voltaje aplicado en terminales de la armadura (V).

T_{mec} : Par mecánico producido en el motor (N.m).

T_L : Par de carga (N.m).

R_a : Resistencia de devanado de armadura (Ω).

En base a la ley de Newton, confirmamos que en cualquier mecanismo giratorio se presenta un balance del par electromagnético T_{elec} con el par mecánico T_{mec} con lo que se obtiene la siguiente igualdad:

$$T_{elec} = T_{mec} = T_m \quad (3.9)$$

Las ecuaciones explicadas para el comportamiento del sistema, se encuentran en función del tiempo para su análisis del sistema dinámico. Cabe anotar que los valores de la constante de par electromagnético k_t y la constante de la fuerza contra electromotriz k_b , se encuentran relacionadas. Una forma de demostrar dicha relación es sabiendo que la potencia en estado permanente generado en la armadura es:

$$P_{elec} = e_a i_a \quad (3.10)$$

La potencia mecánica en estado estacionario es dicha por:

$$P_{mec} = T_{elec} \omega \quad (3.11)$$

En las máquinas eléctricas debe haber un equilibrio entre la potencia eléctrica y la potencia mecánica:

$$P_{elec} = P_{mec} \quad (3.12)$$

$$e_a i_a = T_{elec} \omega \quad (3.13)$$

Reemplazando la ecuación (3.5) y (3.6) en la ecuación (3.13).

$$k_b \omega i_a = k_t i_a \omega$$

Donde se obtiene la siguiente igualdad:

$$k_b = k_t \quad (3.14)$$

Para llegar a demostrar una función de transferencia del motor, tenemos que llevar las ecuaciones obtenidas al dominio de la frecuencia "s" utilizando la transformada de Laplace.

$$T_m(s) = k_t I_a(s) \quad (3.15)$$

$$E_a(s) = k_b \Omega(s) \quad (3.16)$$

$$V_a(s) = L_a s I_a(s) + R_a I_a(s) + E_a(s) \quad (3.17)$$

$$T_m(s) = J s \Omega(s) + B \Omega(s) + T_L(s) \quad (3.18)$$

Dada las ecuaciones (3.15), (3.16), (3.17), (3.18) se obtuvo el respectivo diagrama de bloques:

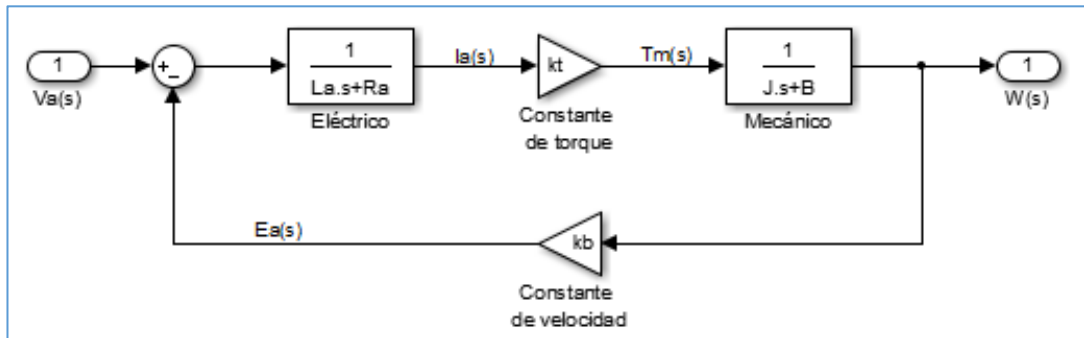


Figura 16. Diagrama de bloques del motor DC

Empleando algebra de bloques se pudo obtener la siguiente función de transferencia perteneciente a la velocidad angular $\omega(s)$:

$$\frac{\omega(s)}{V_a(s)} = \frac{kt}{JL_a s^2 + BL_a s + JR_a s + BR_a + k_t k_b}$$

$$\frac{\omega(s)}{V_a(s)} = \frac{k}{(L_a s + R_a)(J s + B) + k_t k_b} \quad (3.19)$$

Para obtener la posición del motor $\Theta(s)$ integramos la velocidad angular:

$$\frac{\theta(s)}{V_a(s)} = \frac{k}{(L_a s + R_a)(J s + B) + k_t k_b} * \frac{1}{s}$$

$$\frac{\theta(s)}{V_a(s)} = \frac{k}{(L_a s + R_a)(J s + B) s + k_t k_b s} \quad (3.20)$$

3.1.3.1. Parámetros experimentales en la planta

Para poder medir los parámetros del modelo matemático de los motores DC, se removió los motores de la planta.



Figura 17. Motor DC M1 removido

3.1.3.1.1. Medición de la resistencia de armadura R_a

Empleando el multímetro DT-5801, se conectó los puntales a los respectivos terminales de salida de cada motor se realizó la medición de resistencia de armadura. Se obtuvo un número de muestras con las cuales se realizó el promedio de las mismas, como se muestra en la figura 17.

Tabla 1.

Medición experimental de resistencia R_a del motor DC M1

Muestra	R_a M1 (ohm)
1	3.8
2	3.2
3	4
4	3.3
5	2.8
6	3.8
7	3.3
Promedio:	3.457142857

Nota: Se determinó que el valor de R_a del motor DC M1 = 3.457 Ω .

Tabla 2.

Medición experimental de resistencia R_a del motor DC M2

Muestra	R_a M2 (ohm)
1	6.7
2	4.8
3	5.6
4	5.3
5	4.1
6	5.1
7	5.1
Promedio:	5.242857143

Nota: Se determinó que el valor de R_a del motor DC M2 = 5.242 Ω .

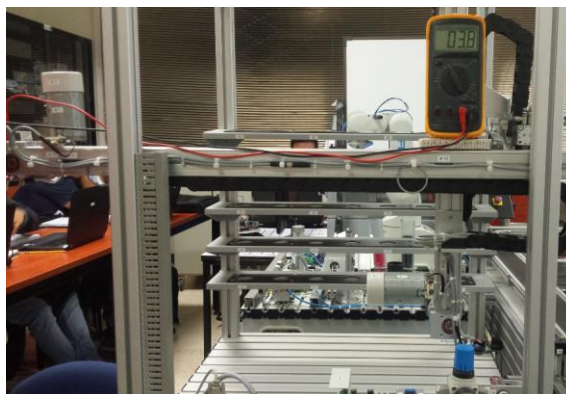


Figura 18. Medición de resistencia de Armadura con multímetro

3.1.3.1.2. Medición de la inductancia de armadura L_a

Para medir el valor de inductancia se utilizó el módulo NI ELVIS II+ de National Instruments, se conectó los terminales de salida de cada motor al módulo obteniendo la medición de inductancia de armadura.

Se obtuvo un número de muestras con las cuales se realizó el promedio de las mismas.

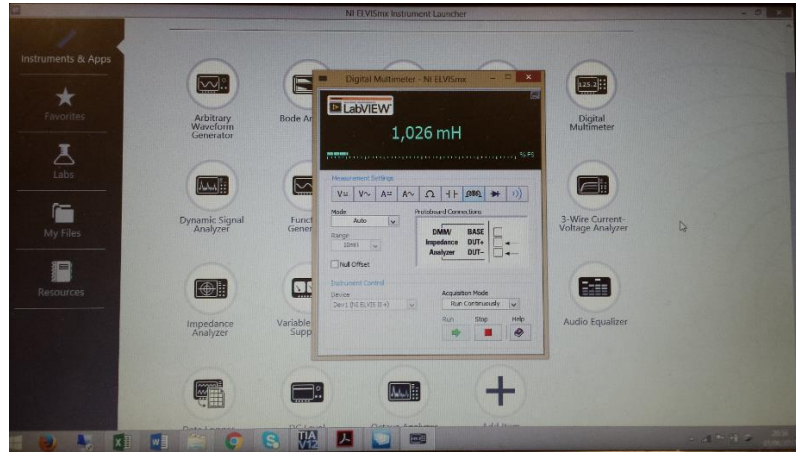


Figura 19. Medición de inductancia de armadura con NI Elvis II+

Tabla 3.

Medición experimental de inductancia L_a del motor DC M1

Muestra	L_a M1 (H)
1	0.001026
2	0.001007
3	0.0005604
4	0.0005422
5	0.001001
6	0.0005517
7	0.000981
8	0.001009
9	0.001024
10	0.001003
Promedio:	0.00087053

Nota: Se determinó que el valor de L_a del motor DC M1 = 0.00087053H.

Tabla 4.

Medición experimental de inductancia L_a del motor DC M2

Muestra	L_a M2 (H)
1	0.00101
2	0.000999
3	0.001018
4	0.00102
5	0.000983
6	0.001001

7	0.0005549
8	0.0005479
9	0.001007
10	0.00102
Promedio:	0.00091608

Nota: Se determinó que el valor de L_a del motor DC M2 = 0.00091608H.

3.1.3.1.3. Constante de la fuerza contra electromotriz kb

Para medir el valor de la constante de la fuerza contra electromotriz, se midió la velocidad angular (ω) de cada motor sin carga a diferentes valores de entrada de voltaje. Se utilizó el osciloscopio PeakTech 1200 para obtener la frecuencia de cada motor. Se utilizó el multímetro Fluke-175 para obtener la corriente de armadura de cada motor. Se utilizó la fuente de voltaje DC GPS-3303 para suministrar los diferentes valores de voltaje de entrada a cada motor.

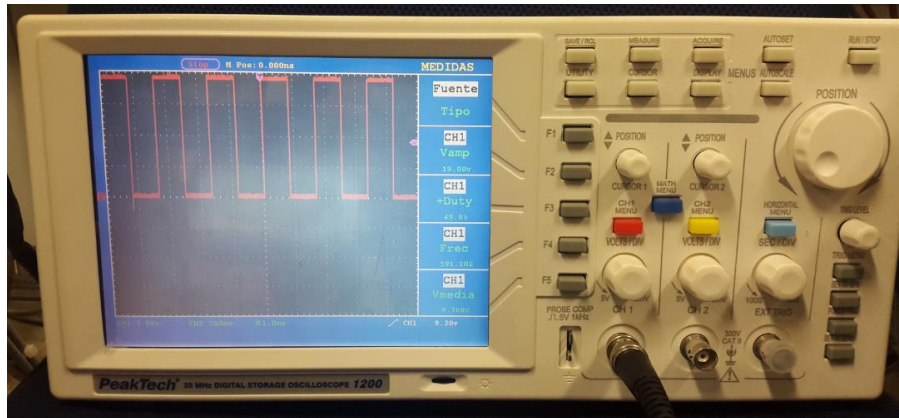


Figura 20. Medición de la frecuencia de los motores.

Para determinar la velocidad angular de cada motor, se midió la frecuencia de los motores, a partir de la siguiente ecuación:

$$\omega = 2\pi f \quad (3.21)$$

Donde f es la frecuencia en Hertz y la velocidad angular en rad/seg. Considerando que se midió la velocidad angular de cada motor en estado estacionario, se pudo calcular la constante de la fuerza contra electromotriz sustituyendo las ecuaciones (3.6), (3.21) en (3.1) y despejando k_b se obtiene:

$$k_b = \frac{V_a - R_a i_a}{\omega} \quad (3.22)$$

Se realizó el cálculo de los valores de la velocidad angular y se midió la corriente de armadura de cada motor, se obtuvo el valor de k_b .

Tabla 5.

Medición experimental de la constante de fuerza contra electromotriz kb del motor DC M1

Muestra	Voltaje (V)	Ra (ohm)	Ia (A)	F (Hz)	w (rad/s)	Kb (V.seg/rad)
1	2	3.457	0.317	60.74	381.640676	0.002369064
2	4	3.457	0.354	252.4	1585.87597	0.001750592
3	6	3.457	0.417	414.9	2606.89358	0.001748606
4	8	3.457	0.476	591.2	3714.61915	0.001710665
5	10	3.457	0.537	776.4	4878.26507	0.001669362
6	12	3.457	0.597	970.4	6097.20302	0.001629628
7	14	3.457	0.641	1162	7301.06133	0.001614021
8	16	3.457	0.69	1325	8325.22053	0.001635352
9	18	3.457	0.74	1536	9650.97263	0.001600027
10	20	3.457	0.783	1719	10800.7955	0.001601101
11	22	3.457	0.815	1920	12063.7158	0.001590103
Promedio:						0.001719866

Nota: Se determinó que el valor de kb del motor DC M1 = $0.001719866 \frac{V.seg}{rad}$.

Tabla 6.

Medición experimental de la constante de fuerza contra electromotriz kb del motor DC M2

Muestra	Voltaje (V)	Ra (ohm)	Ia (A)	F (Hz)	w (rad/s)	kb (V.seg/rad)
1	2	5.24285	0.361	94.75	595.331808	0.000180288
2	4	5.24285	0.455	268.9	1689.54853	0.000955583
3	6	5.24285	0.522	433.9	2726.27411	0.001196957
4	8	5.24285	0.581	603.7	3793.15897	0.00130601
5	10	5.24285	0.645	779.7	4898.99958	0.001350962
6	12	5.24285	0.696	962.6	6048.19418	0.001380739
7	14	5.24285	0.74	1154	7250.79585	0.001395749
8	16	5.24285	0.783	1319	8287.52142	0.001435272
9	18	5.24285	0.818	1506	9462.47707	0.001449023
10	20	5.24285	0.849	1696	10656.2823	0.001459122
11	22	5.24285	0.883	1882	11824.9547	0.001468975
Promedio						0.001234425

Nota: Se determinó que el valor de kb del motor DC M2 = $0.001234425 \frac{V.seg}{rad}$.

3.1.3.1.4. Constante del par electromagnético kt

Usando la relación de la ecuación (3.14) y expresando Kt en las respectivas unidades se determinó la constante del par electromagnético para cada motor.

Se determinó que el valor de kb del motor DC M1 = $0.001719866 \frac{N.m}{A}$.

Se determinó que el valor de kb del motor DC M2 = $0.001234425 \frac{N.m}{A}$.

3.1.3.1.5. Coeficiente de fricción viscosa B

Para determinar el coeficiente de fricción viscosa de cada motor, es imprescindible usar la ecuación diferencial de cada motor sin carga:

$$T_m = J_m \frac{d\omega}{dt} + B\omega \quad (3.23)$$

Donde $T_m = k_b \cdot i_a$ es el torque en el motor. Al encender el motor con un voltaje constante durante un tiempo, hasta que llegue a una velocidad constante, la aceleración angular desaparece obteniendo directamente la relación del coeficiente de fricción viscosa. Se elimina el término de la derivada de la velocidad angular y se despeja el coeficiente de fricción, se tiene que:

$$B = \frac{k_b i_a}{\omega} \quad (3.24)$$

Considerando los valores de k_b e i_a encontrados anteriormente podemos obtener que:

Se determinó que el valor de B del motor M1 = $0.000312892 \frac{N.m.seg}{rad}$.

Se determinó que el valor de B del motor M2 = $0.00031466 \frac{N.m.seg}{rad}$.

3.1.3.1.6. Movimiento de inercia equivalente del rotor J_m

Usando la ecuación (3.23) y resolviendo la misma ecuación es posible determinar los valores del momento de inercia de cada motor. Conociendo que la solución de esta ecuación diferencial puede ser encontrada como la suma de una solución particular más una respuesta transitoria.

$$y = y_p + y_t$$

Para conocer la solución particular de la ecuación, el término derivativo de la ecuación desaparece con lo cual la solución particular puede ser determinada como:

$$y_p = \omega = \frac{T_m}{B} \quad (3.25)$$

Obtenido la solución particular del sistema se tiene que calcular la respuesta transitoria, para ello se realiza la igualdad de $\omega = e^{mt}$, sustituyendo e igualando a cero:

$$J_m = \frac{de^{mt}}{dt} + Be^{mt} = 0$$

Resolviendo la derivada de e^{mt} en el primer término de la ecuación tenemos que:

$$e^{mt}(mJ_m + B) = 0$$

Para encontrar el valor de la variable m en la exponencial se realiza el cálculo de la ecuación anterior para un tiempo igual a cero, siendo el valor de la exponencial igual a uno. Despejando m se encuentra el valor como:

$$m = -\frac{B}{J_m}$$

Una vez encontrado el valor de la constante de tiempo en la exponencial la respuesta transitoria esta dada por:

$$y(t) = \frac{T_m}{B} e^{-\frac{B}{Jm}t} \quad (3.26)$$

Agregando la respuesta particular y la respuesta transitoria.

$$\omega(t) = \frac{T_m}{B} \left(1 - e^{-\frac{B}{Jm}t} \right)$$

Donde la constante de tiempo de esta ecuación es igual a:

$$\tau = \frac{Jm}{B} \quad (3.27)$$

Dicha constante es reconocida como la constante mecánica del motor, la cual por medio de experimentación se calcula como un tercio del tiempo transcurrido entre la desconexión del motor sin carga a una velocidad constante y su frenado.

Para hallar el tiempo que transcurrió entre la desconexión y el frenado de cada motor se empleó la fuente de voltaje DC GPS-3303 y el osciloscopio PeakTech 1200 como muestra la siguiente figura:

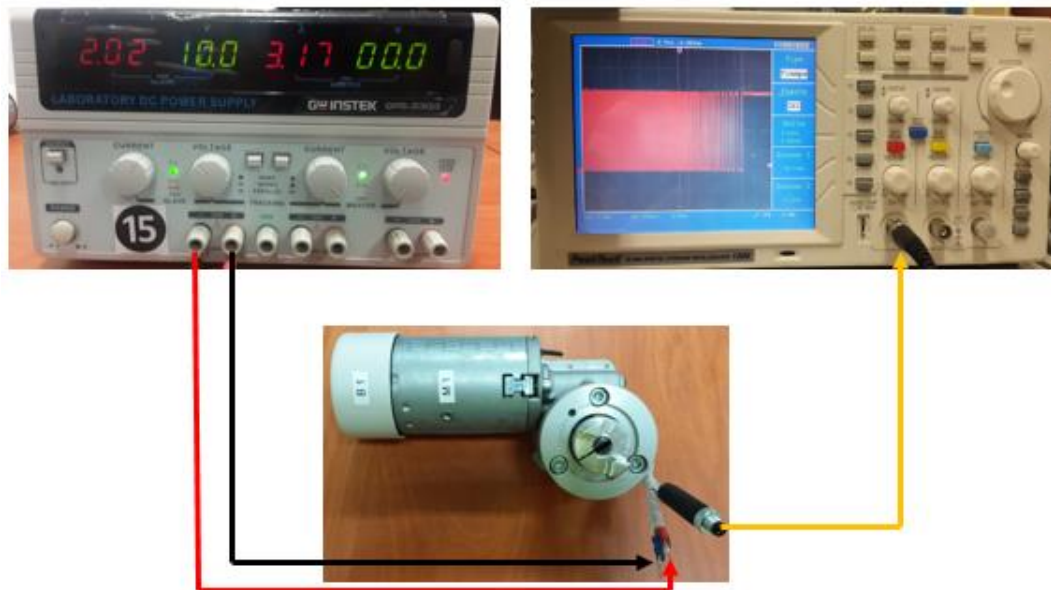


Figura 21. Medición del tiempo de freno en los motores.

Tabla 7.

Medición experimental de la constante mecánica del motor DC M1

Muestra	Tiempo de Frenado (seg)
1	0.144
2	0.144
3	0.198
4	0.152
5	0.306
6	0.268

7	0.23
8	0.222
9	0.228
Promedio:	0.210222222

Nota: Se determinó que el tiempo de frenado del motor DC M1 = 210 ms.

Tabla 8.

Medición experimental de la constante mecánica del motor DC M2

Muestra	Tiempo de Frenado (seg)
1	0.156
2	0.162
3	0.226
4	0.236
5	0.232
6	0.126
7	0.118
8	0.206
9	0.204
Promedio:	0.185111111

Nota: Se determinó que el tiempo de frenado del motor DC M2 = 185 ms.

Con esto se utilizó la ecuación (3.27), reemplazando los valores de la constante de tiempo y la constante de fricción equivalente a un tercio de la constante mecánica del motor se determina el momento de inercia de cada motor como:

Tabla 9.

Determinación del momento de inercia del motor DC M1

τ (seg)	B (N.m.seg/rad)	Jm (Kg.m²)
0.07007407	0.000312892	2.19256E-05

Nota: Se determinó que el momento de inercia del motor DC M1 = 2.19256E-05 kg.m².

Tabla 10.

Determinación del momento de inercia del motor DC M2

τ (seg)	B (N.m.seg/rad)	Jm (Kg.m²)
0.0617037	0.00031466	1.94157E-05

Nota: Se determinó que el momento de inercia del motor DC M2 = 1.94157E-05 kg.m².

3.2. Conversión analógica de salida

Para que los motores DC M1 y M2 puedan energizarse con los valores de voltaje de -24 voltios a 24 voltios se realizó una conversión analógica de salida del PLC S7-300 a través del programa TIA PORTAL.

La salida del PLC es de 0 voltios a 10 voltios representados analógicamente de 0 a 27648.

Se realizó la obtención de la ecuación de la relación del voltaje vs el porcentaje que corresponde al voltaje utilizando la ecuación de la pendiente.

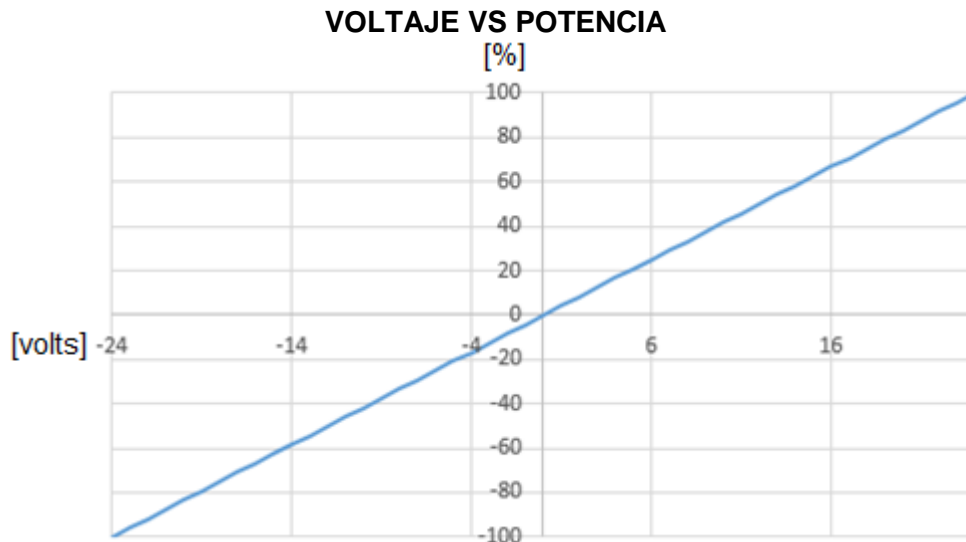


Figura 22. Grafica del voltaje vs Porcentaje.

$$m = \frac{100 + 100}{24 + 24}$$

$$m = 4.1666$$

Y luego de ello se utilizó la ecuación de la recta:

$$y + 100 = 4.1666(x + 24)$$

$$y = 4.1666x - 0.0016$$

Donde se pudo obviar el término negativo porque no afecta a la ecuación quedando la ecuación en:

$$y = 4.1666x$$

Con la ecuación determinada se la representó en el programa TIA PORTAL de la siguiente manera:

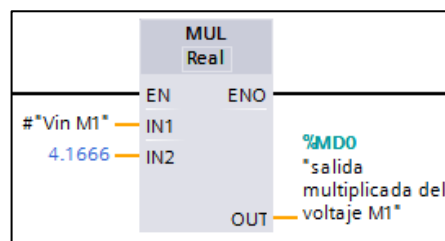


Figura 23. Ecuación de voltaje vs porcentaje en tia portal.

Obtenida la ecuación de relación voltaje vs porcentaje, se procedió a des escalar los valores en el rango de -27648 hasta 27648.

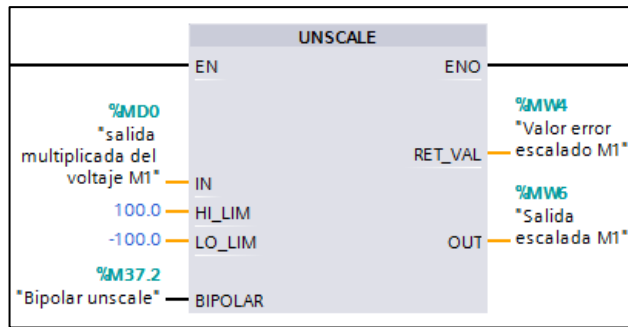


Figura 24. Des escalamiento de valores de la ecuación descrita.

Con el des escalamiento obtenido se realizó la división de estos valores a la mitad ya que la entrada del driver de los motores es de 0 voltios a 5 voltios, es decir en valores analógicos de -13829 a 13829.

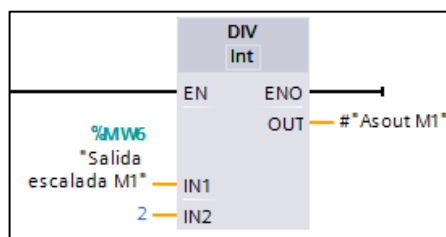


Figura 25. Representación del des escalamiento final.

3.3. Conversión de la lectura del encoder

Los datos proporcionados por los encoders incrementales de los motores DC M1 y M2 son los impulsos contados al momento en el que los motores se energicen con un voltaje determinado.

Estos impulsos se obtuvieron en el programa TIA PORTAL mediante la función SFB COUNT (SFB 47).

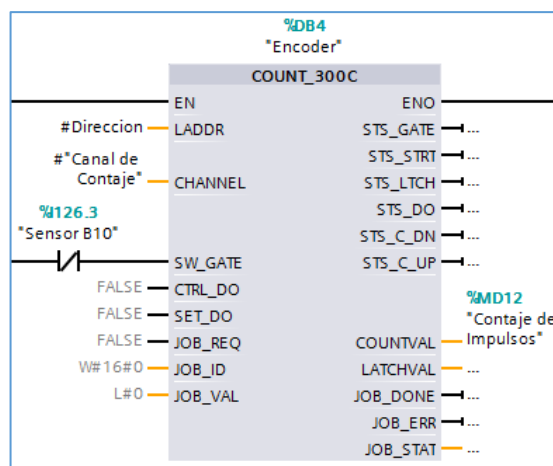


Figura 26. Función SFB COUNT (SFB 47).

Para que la función SFB COUNT (SFB 47) pueda leer los impulsos correctamente, se configuró el encoder en el programa TIA PORTAL en configuración de dispositivos, general, contaje y en canal 0 se configuró el modo de contaje para el encoder del motor DC M1.

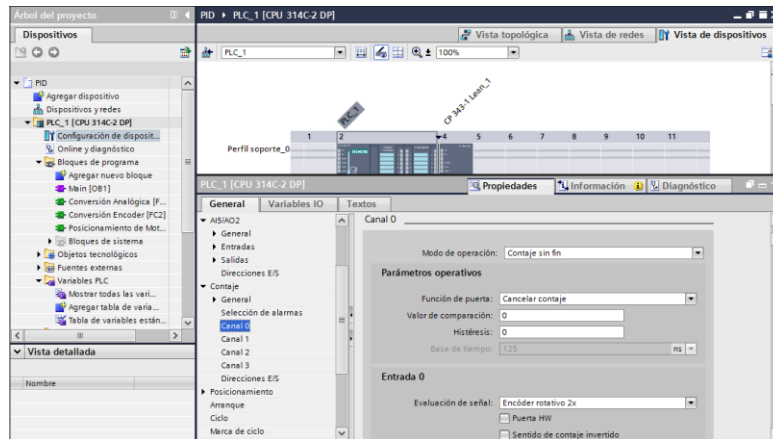


Figura 27. Configuración del canal 0 del encoder del motor DC M1.

De la misma forma se configuró el canal 1 para el encoder del motor DC M2.

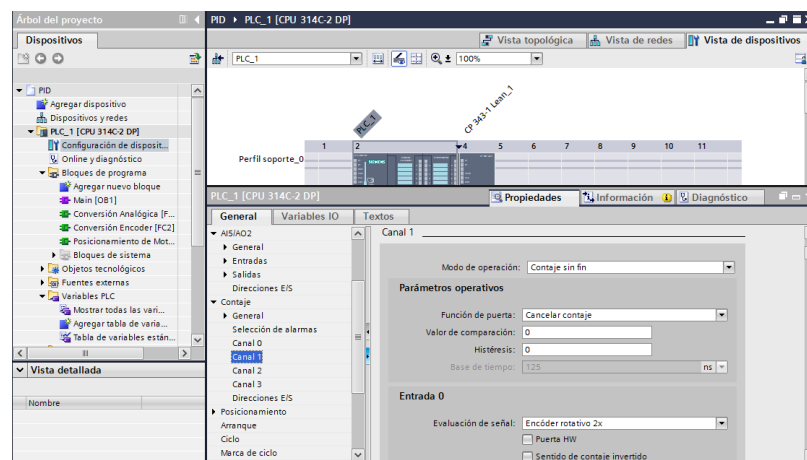


Figura 28. Configuración del canal 1 del encoder del motor DC M1.

Para obtener la distancia recorrida por los motores DC M1 y M2, se obtuvo las respectivas ecuaciones de relación.

Para el motor DC M1 la distancia total de desplazamiento desde el sensor B10 tiene 60 cm de área de trabajo con respecto al eje horizontal, quedando 1 cm antes del choque con la propia estructura.

Se realizó la obtención de la ecuación de la relación de los impulsos vs la distancia utilizando la ecuación de la pendiente.

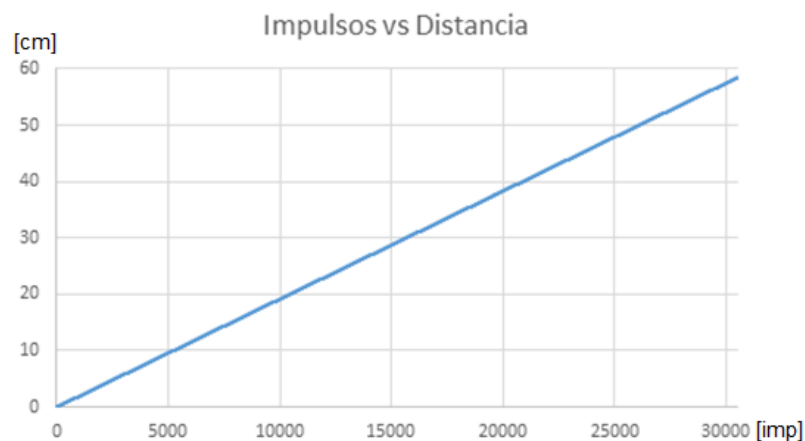


Figura 29. Impulsos vs Distancia del encoder motor DC M1.

$$m = \frac{60 + 0}{31325 + 0}$$

$$m = 0.001915403$$

Y luego de ello se utilizó la ecuación de la recta:

$$y - 0 = 0.001915403(x - 0)$$

$$y = 0.001915403x$$

Con la ecuación determinada se la representó en el programa TIA PORTAL de la siguiente manera:

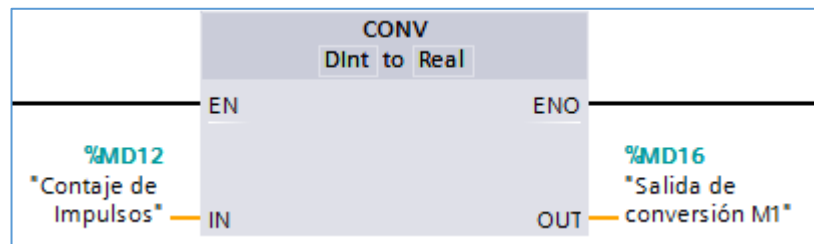


Figura 30. Conversión de datos enteros del encoder del motor DC M1.

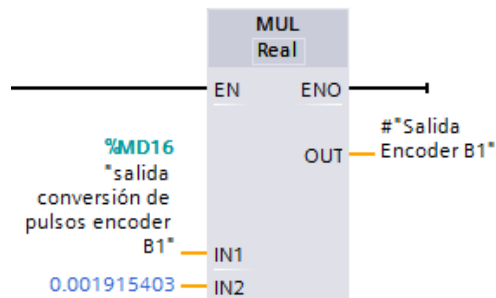


Figura 31. Ecuación de impulsos vs distancia del encoder del motor DC M1

Para el motor DC M2 la distancia total de desplazamiento desde el sensor B12 tiene 41 cm de área de trabajo con respecto al eje vertical. Se realizó la obtención de la ecuación de la relación de los impulsos vs la distancia utilizando la ecuación de la pendiente.

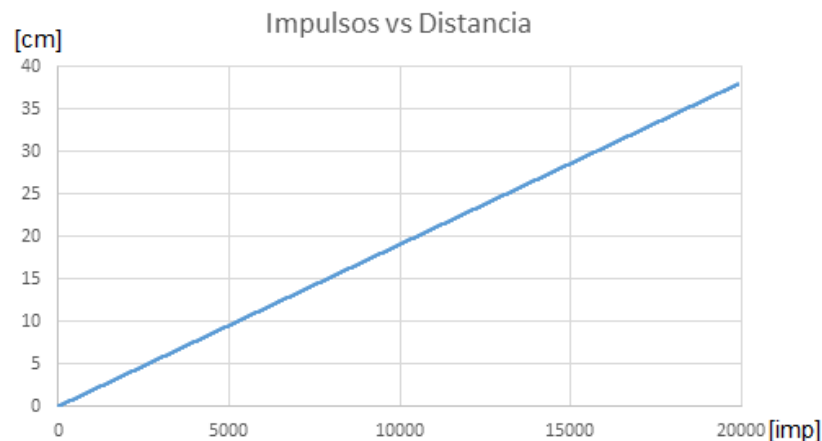


Figura 32. Impulsos vs Distancia del encoder del motor DC M2.

$$m = \frac{41 + 0}{21441 + 0}$$

$$m = 0.001912224$$

Y luego de ello se utilizó la ecuación de la recta:

$$y - 0 = 0.001912224(x - 0)$$

$$y = 0.001912224$$

Con la ecuación determinada se la representó en el programa TIA PORTAL de la siguiente manera:

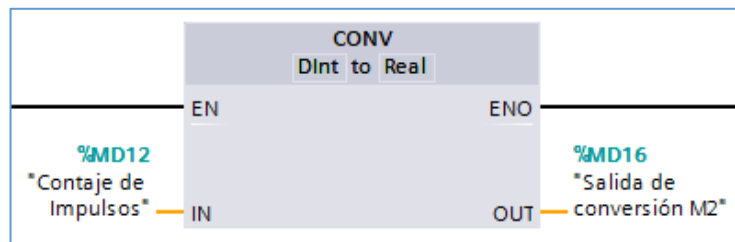


Figura 33. Conversión de datos enteros del encoder del motor DC M2

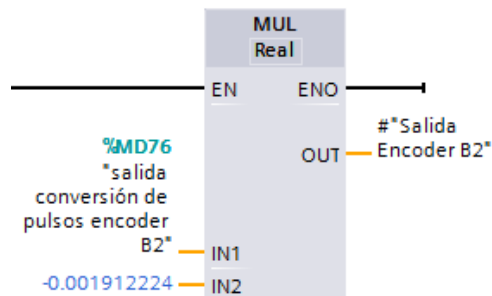


Figura 34. Ecuación de impulsos vs distancia del encoder del motor DC M2.

3.4. Validación del modelo matemático de los motores DC M1 y M2

Para las pruebas de validación de los modelos matemáticos obtenidos, se utilizó el generador de señales AFG-2112, el cual proporciona la señal cuadrada y la señal seno, las cuales energizaron a los motores DC M1 y M2.



Figura 35. Generador de señales AFG-2112.

3.4.1. Pruebas de validación de velocidad del motor DC M1

Para la validación de velocidad del motor DC M1, se procedió a generar dos señales de pruebas con la finalidad de apreciar la aproximación con los datos reales. Para las pruebas se utilizó dos señales de voltaje, la onda cuadrada y la onda sinusoidal, los cuales se realizaron en el software Simulink. Para esta prueba se desconectó el motor DC M1 de la planta.

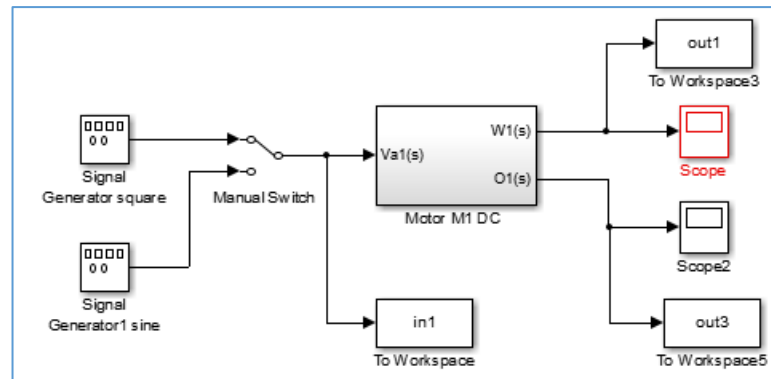


Figura 36. Simulación del modelo de velocidad del motor DC M1

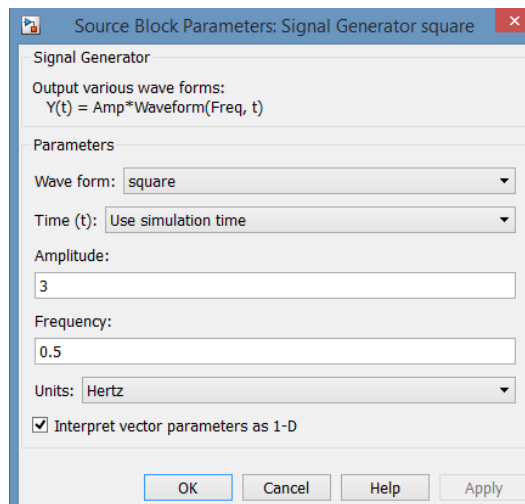


Figura 37. Señal cuadrada al motor DC M1

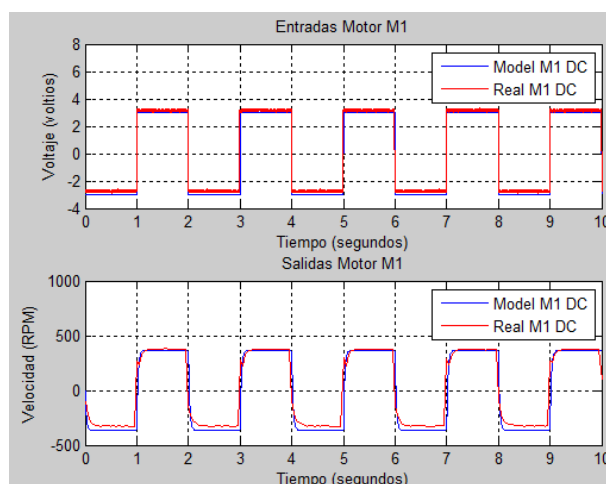


Figura 38. Comparación de velocidad con señal cuadrada del motor DC M1

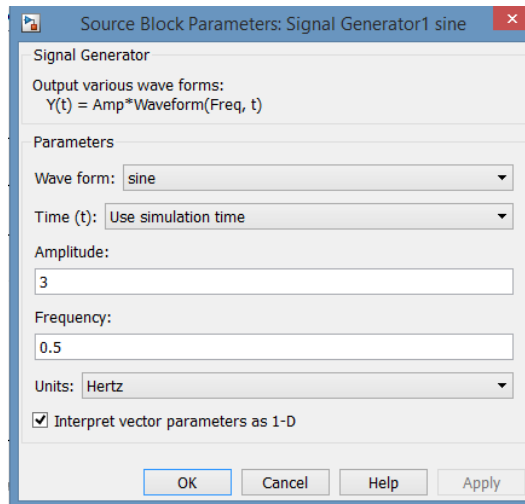


Figura 39. Señal sinusoidal al motor DC M1

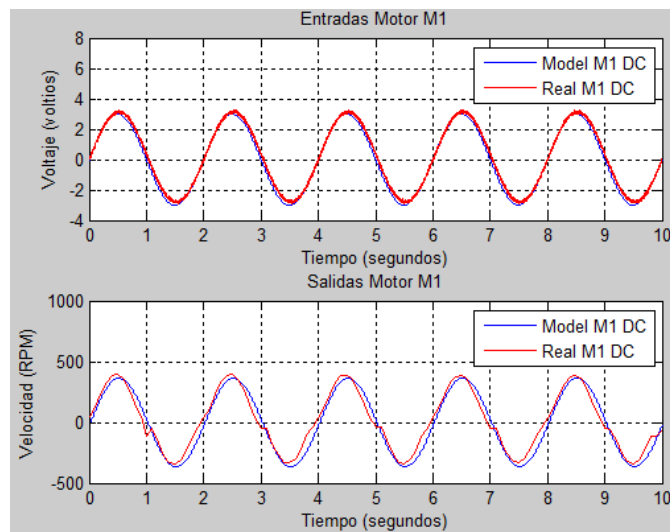


Figura 40. Comparación de velocidad con una señal sinusoidal del motor DC M1

3.4.2. Pruebas de validación de velocidad del motor DC M2

Para la validación de velocidad del motor DC M2, se procedió a generar dos señales de pruebas con la finalidad de apreciar la aproximación con los datos reales. Para esta prueba se desconectó el motor DC M2 de la planta.

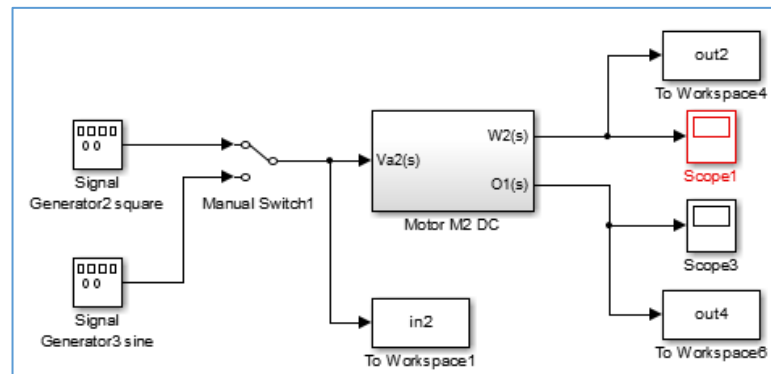


Figura 41. Simulación del modelo de velocidad del motor DC M2

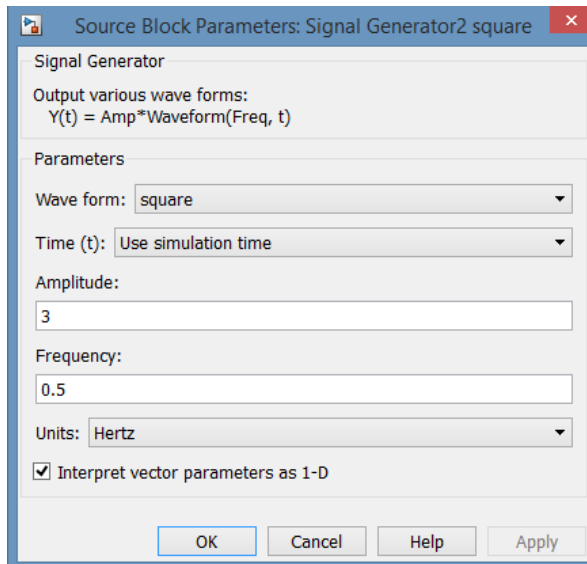


Figura 42. Señal cuadrada al motor DC M2

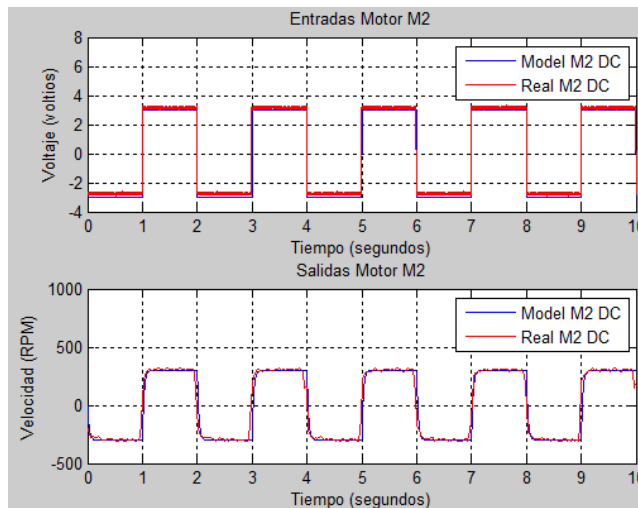


Figura 43. Comparación de velocidad con señal cuadrada del motor DC M2

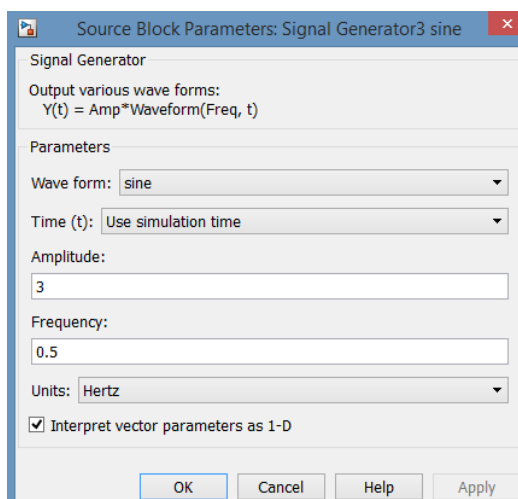


Figura 44. Señal sinusoidal al motor DC M2

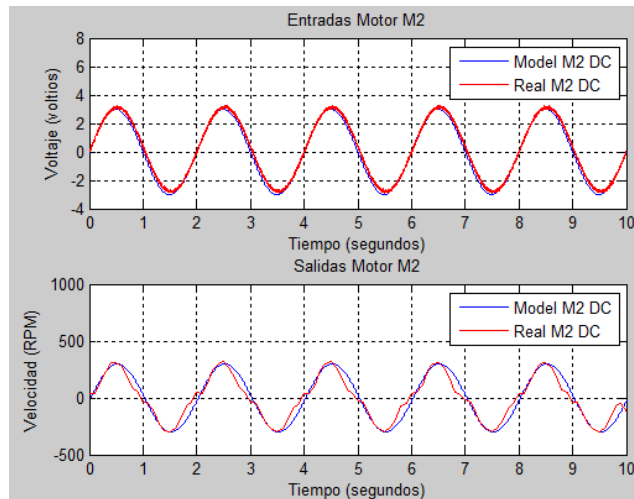


Figura 45. Comparación de velocidad con señal sinusoidal del motor DC M2

3.4.3. Pruebas de validación de distancia del motor DC M1

Para la validación de distancia del motor DC M1, se procedió a generar dos señales de pruebas con la finalidad de apreciar la aproximación con los datos reales. Para esta prueba se desconectó el motor DC M1 de la planta.

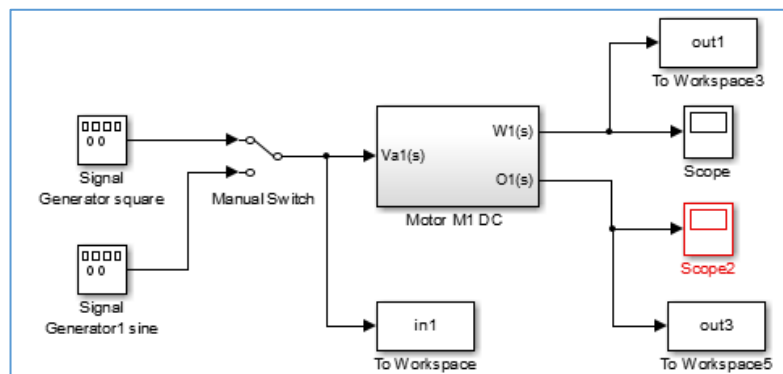


Figura 46. Simulación del modelo de distancia del motor DC M1

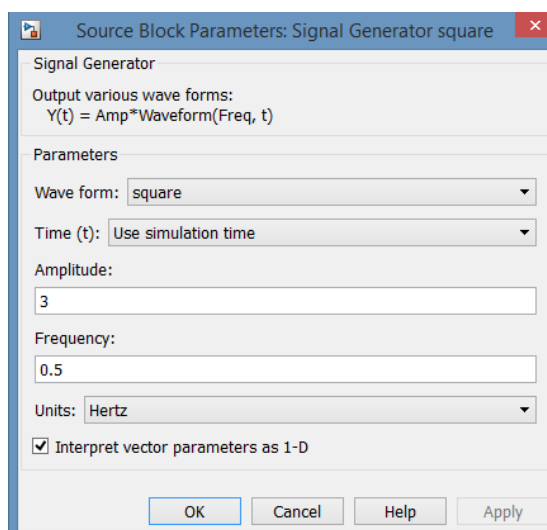


Figura 47. Señal cuadrada al motor DC M1

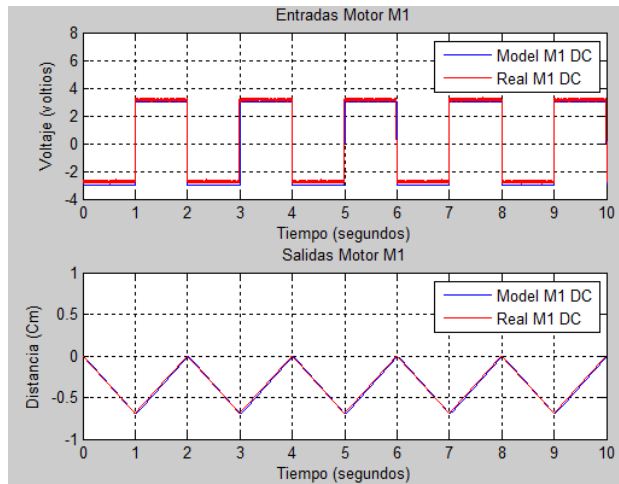


Figura 48. Comparación de distancia con señal cuadrada del motor DC M1

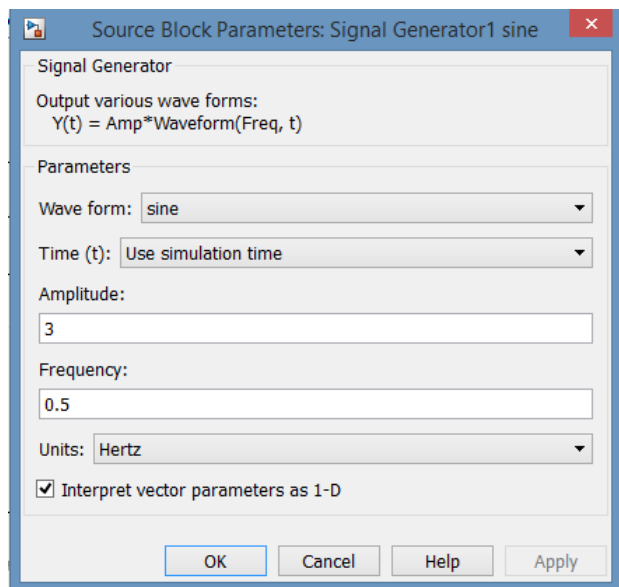


Figura 49. Señal sinusoidal al motor DC M1

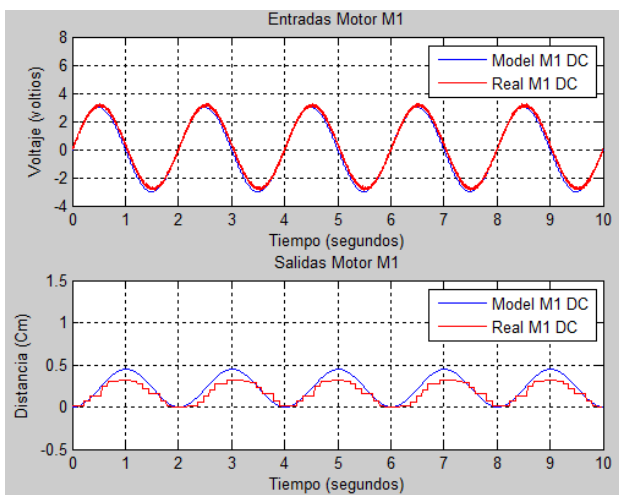


Figura 50. Comparación de distancia con señal sinusoidal del motor DC M1

3.4.4. Pruebas de validación de distancia del motor DC M2

Para la validación de distancia del motor DC M2, se procedió a generar dos señales de pruebas con la finalidad de apreciar la aproximación con los datos reales. Para esta prueba se desconectó el motor DC M2 de la planta.

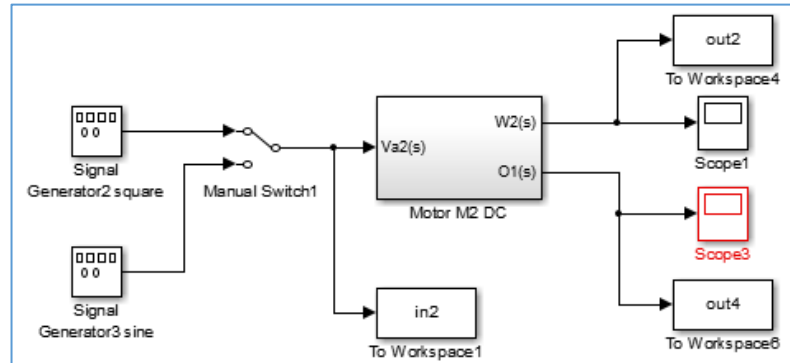


Figura 51. Simulación del modelo de distancia del motor DC M2

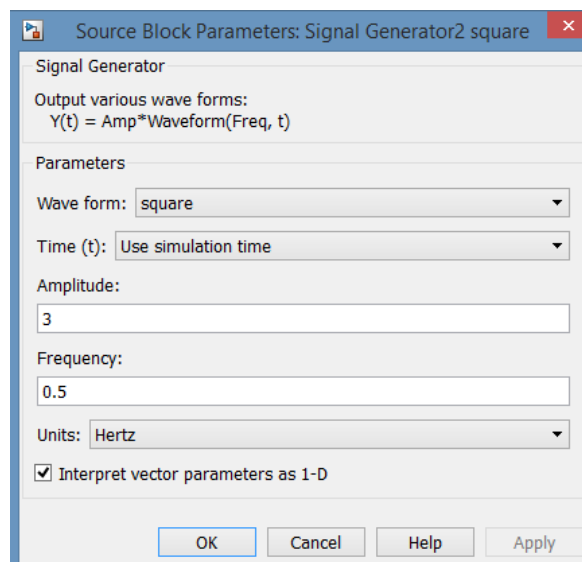


Figura 52. Señal cuadrada al motor DC M2

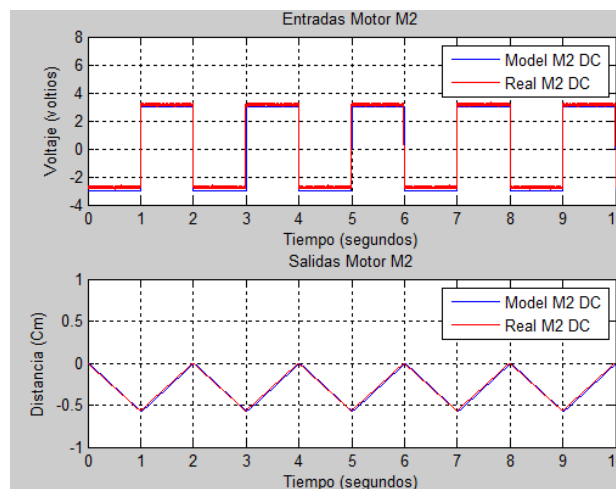


Figura 53. Comparación de distancia con señal cuadrada del motor DC M2

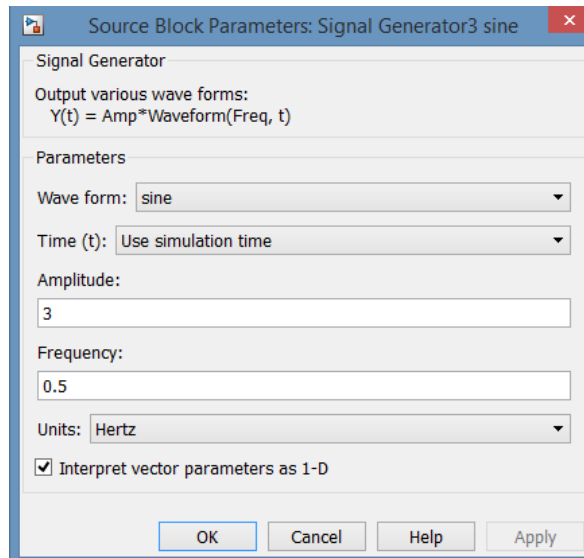


Figura 54. Señal sinusoidal al motor DC M2

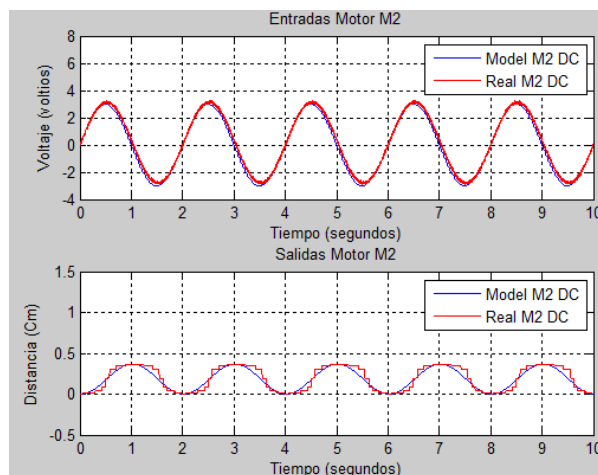


Figura 55. Comparación de la distancia con señal sinusoidal del motor DC M2

3.5. Error absoluto y relativo de los motores DC M1 y M2

Para la validación de los modelos matemáticos con los modelos reales obtenidos de la velocidad del motor DC M1, se obtuvieron los cálculos de los errores absolutos y errores relativos.

Tabla 11.

Obtención del error absoluto, relativo con señal cuadrada y seno del motor DC M1

Señal de Prueba Aplicada al Motor M1 (Velocidad)	Error Absoluto (RPM)	Error Relativo (%)
Onda Cuadrada	24.4052	7.382
Onda Seno	22.6869	10.8939

Para la validación de los modelos matemáticos con los modelos reales obtenidos de la velocidad del motor DC M2, se obtuvieron los cálculos de los errores absolutos y errores relativos.

Tabla 12.*Obtención del error absoluto, relativo con señal cuadrada y seno del motor DC M2*

Señal de Prueba Aplicada al Motor M2 (Velocidad)	Error Absoluto (RPM)	Error Relativo (%)
Onda Cuadrada	0.0567	0.0198
Onda Seno	29.3467	18.3175

Para la validación de los modelos matemáticos con los modelos reales obtenidos de la distancia del motor DC M1, se obtuvieron los cálculos de los errores absolutos y errores relativos.

Tabla 13.*Obtención del error absoluto, relativo con señal cuadrada y seno del motor DC M1*

Señal de Prueba Aplicada al Motor M1 (Distancia)	Error Absoluto (cm)	Error Relativo (%)
Onda Cuadrada	0.0061	1.7659
Onda Seno	0.0534	31.321

Para la validación de los modelos matemáticos con los modelos reales obtenidos de la distancia del motor DC M2, se obtuvieron los cálculos de los errores absolutos y errores relativos.

Tabla 14.*Obtención del error absoluto, relativo con señal cuadrada y seno del motor DC M2*

Señal de Prueba Aplicada al Motor M2 (Distancia)	Error Absoluto (cm)	Error Relativo (%)
Onda Cuadrada	0.0054	1.9027
Onda Seno	0.012	6.0835

3.6. Diseño de controladores PID de los motores DC M1 y M2

En esta parte se realizó el diseño de los controlados P, PD, PI, PID de distancia a los motores DC M1 y M2.

Para diseñar los controladores se tomó en cuenta la función transferencia en la lazo abierto mostrada (3.20), obteniendo la función transferencia de los motores DC M1 y M2, las cuales fueron ingresadas en el programa Matlab, en la herramienta Sisotool.

$$M1 = \frac{\theta_1(s)}{V_{a1}(s)} = \frac{0.0009512}{1.909e^{-8}s^3 + 9.8e^{-5}s^2 + 0.004057s} \quad (3.21)$$

$$M2 = \frac{\theta_2(s)}{V_{a2}(s)} = \frac{0.0009333}{1.782e^{-8}s^3 + 0.0001412s^2 + 0.004824s} \quad (3.22)$$

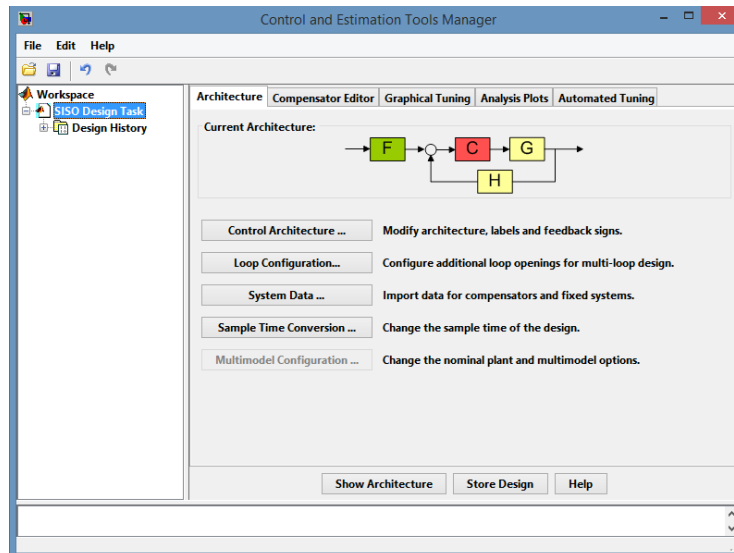


Figura 56. Herramienta sisotool de Matlab

Para prevenir que la salida del controlador no se sature, se fijó que la salida sea de 22 voltios ya que el voltaje nominal de la tarjeta Sabertooth 2X60 y de los motores DC M1 y M2 es de 24 voltios.

3.6.1. Diseño del controlador P del motor DC M1

Para diseñar el controlador P se tomó en cuenta la función transferencia en lazo abierto mostrada (3.21), la cual fue ingresada en el programa Matlab, en la herramienta sisotool, donde se prosedió a designar la estructura de control.

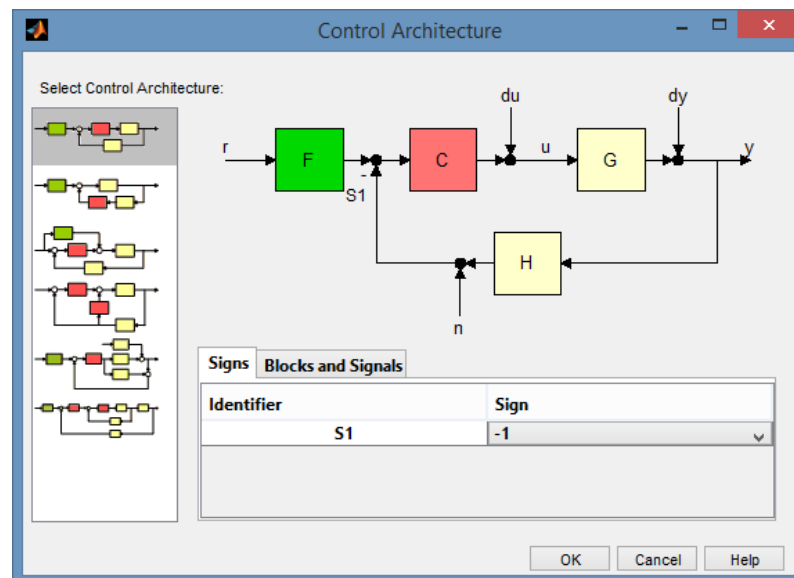


Figura 57. Estructura de control P del motor DC M1

El diseño del controlador estuvo basado en el análisis del lugar geométrico de las raíces como se muestra en las siguientes figuras.

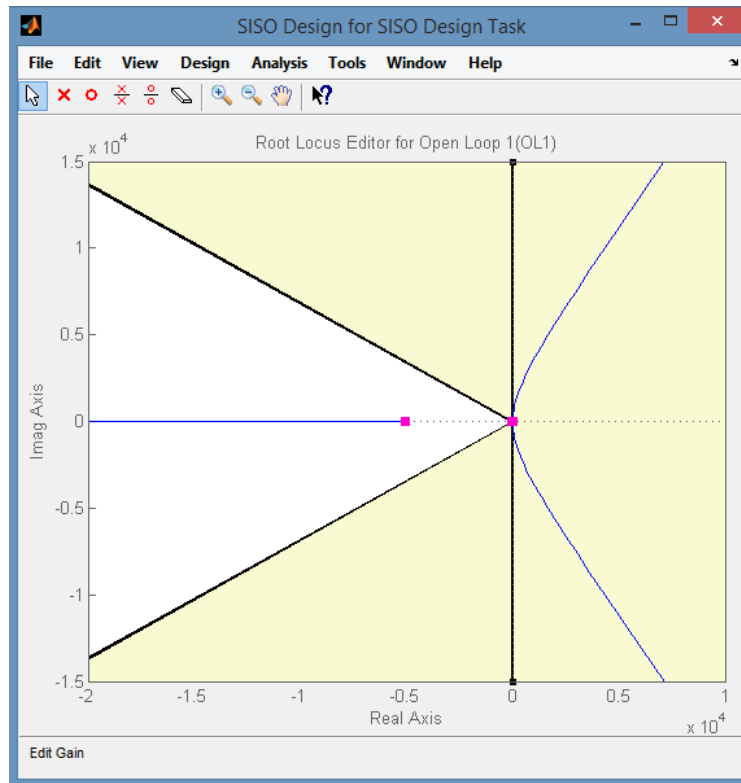


Figura 58. Lugar geométrico de las raíces del motor DC M1 con controlador P

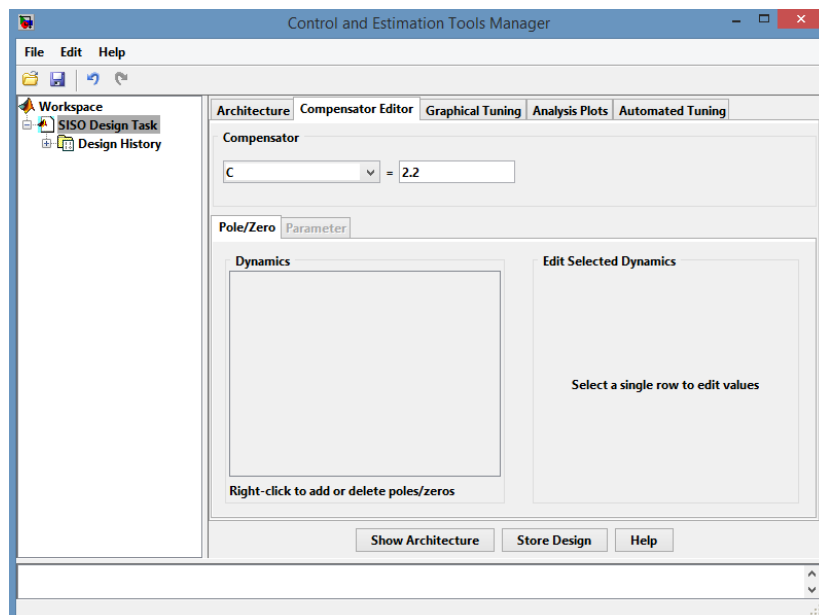


Figura 59. Controlador P para el motor DC M1

Tomando en cuenta el compensador de la figura anterior se tiene que:

$$k_p = 2.2$$

Usando Simulink se procedió a simular la función transferencia a lazo cerrado con el compensador diseñado, agregando una entrada paso de 10 (cm) y la salida de control del compensador.

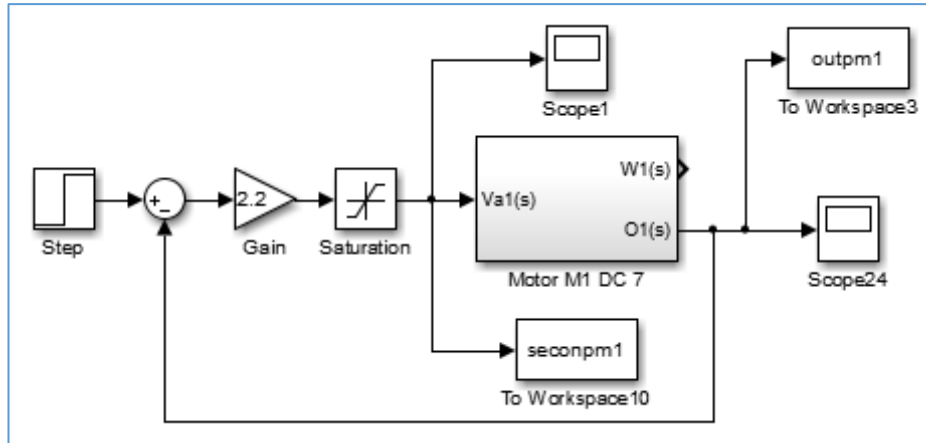


Figura 60. Diagrama de bloques para el control P del motor DC M1

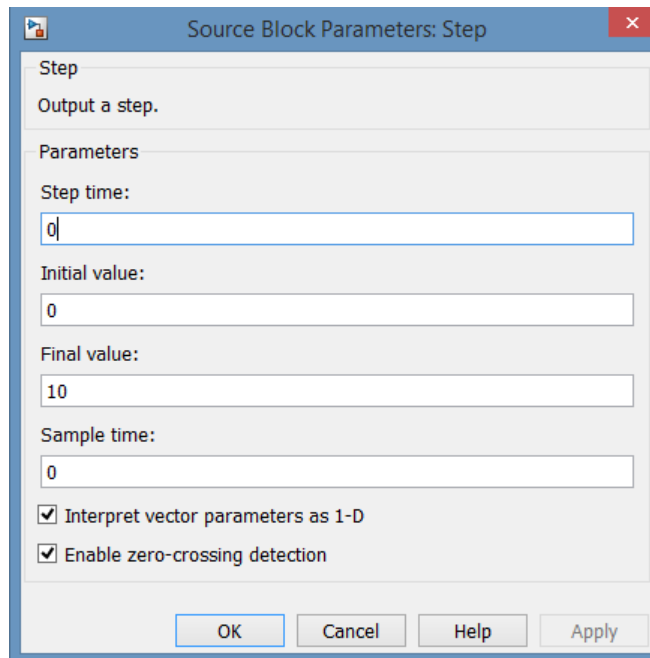


Figura 61. Configuración de la entrada paso a 10cm

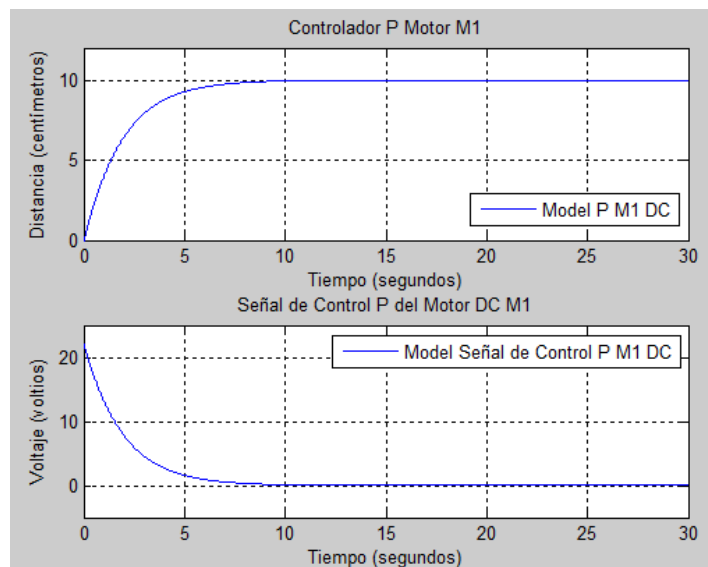


Figura 62. Respuesta de entrada paso y salida del controlador P del motor M1

3.6.2. Diseño del controlador PD del motor DC M1

Para diseñar el controlador PD se tomó en cuenta la función transferencia en lazo abierto mostrada (3.21), la cual fue ingresada en el programa Matlab, en la herramienta sisotool, donde se prosedió a designar la estructura de control.

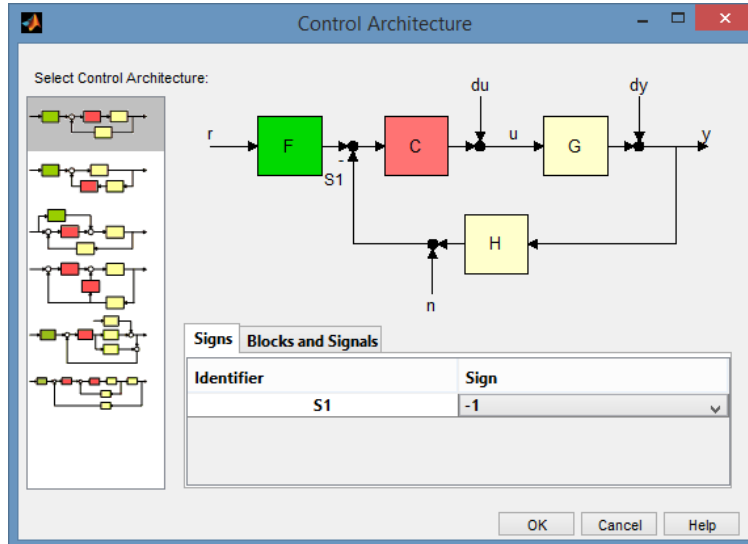


Figura 63. Estructura de control PD del motor DC M1

El diseño del controlador estuvo basado en el análisis del lugar geométrico de las raíces como se muestra en las siguientes figuras.

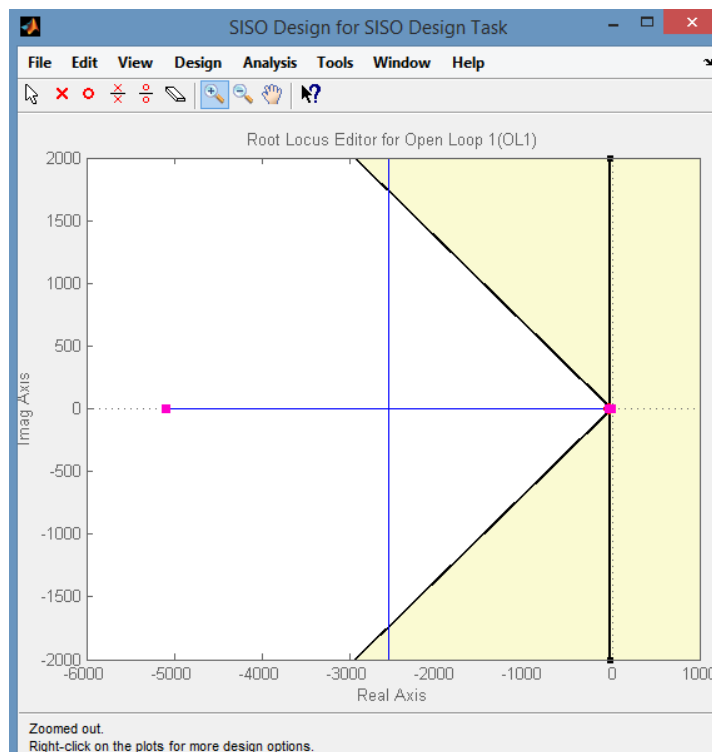


Figura 64. Lugar geométrico de las raíces del motor DC M1 con controlador PD

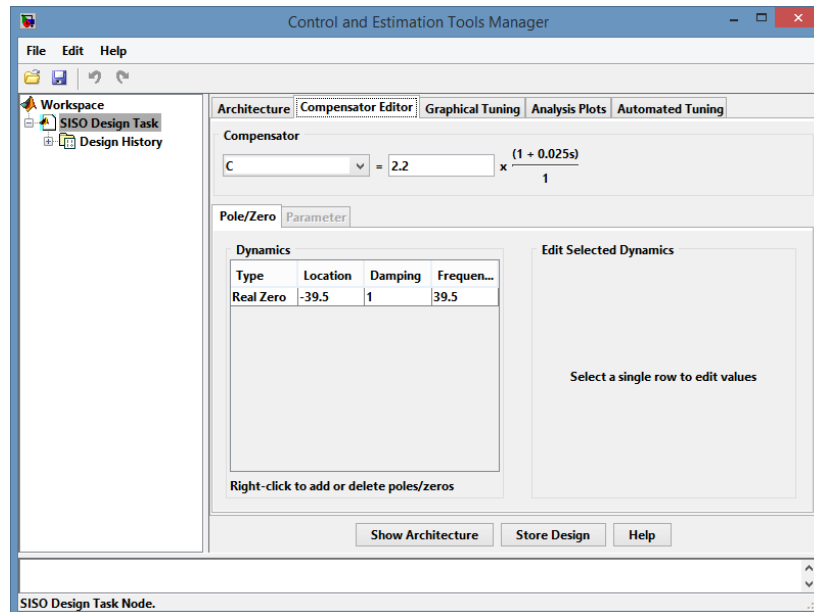


Figura 65. Controlador PD para el motor DC M1

Tomando en cuenta el compensador de la figura anterior se tiene que:

$$G_{M1}(s) = k_p(1 + \tau_d s)$$

$$G_{M1}(s) = k_p + k_p \tau_d s$$

$$G_{M1} = k_p + k_d s$$

$$C_{PD}(s) = 2.2(1 + 0.025s)$$

Donde:

$$C_{PD}(s) = 2.2 + 0.055s$$

Reemplazando:

$$k_p = 2.2$$

$$k_d = 0.055$$

Usando Simulink se procedió a simular la función transferencia a lazo cerrado con el compensador diseñado agregando una entrada paso de 10 (cm) y la salida de control del compensador.

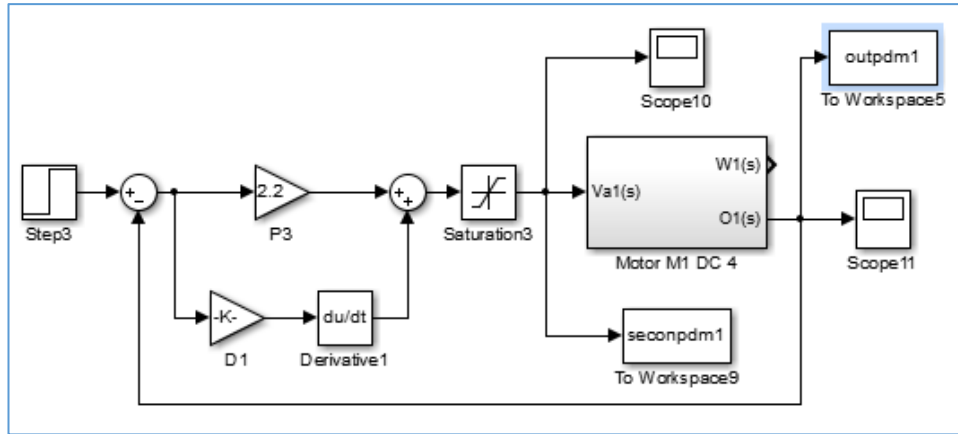


Figura 66. Diagrama de bloques para el control PD del motor DC M1

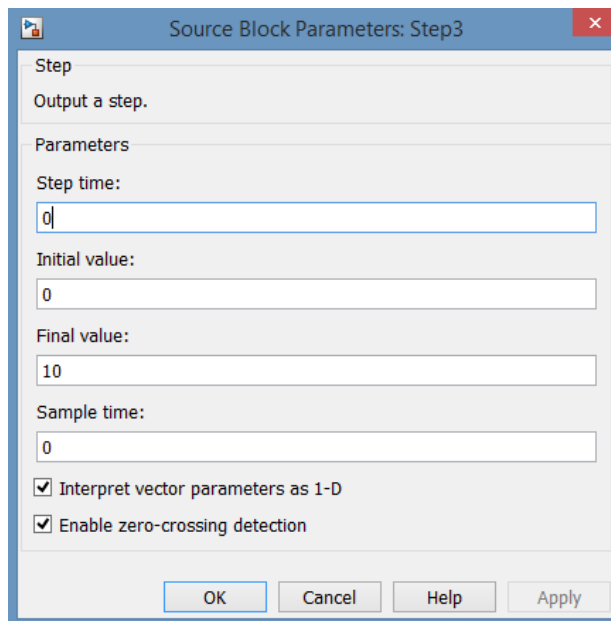


Figura 67. Configuración de la entrada paso a 10cm

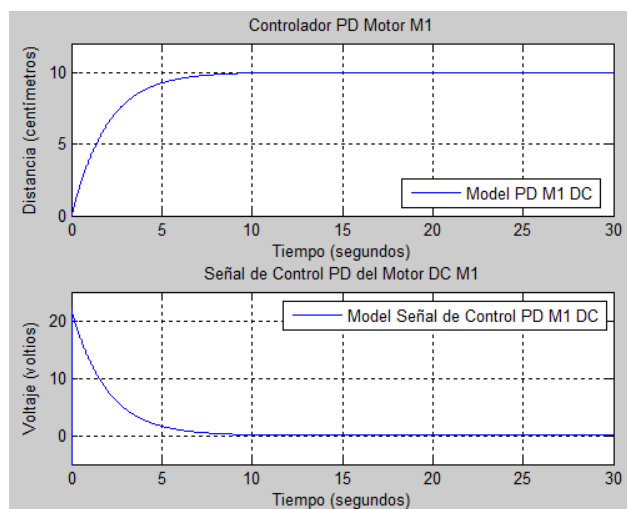


Figura 68. Respuesta de entrada paso y salida del controlador PD del motor M1

3.6.3. Diseño del controlador PI del motor DC M1

Para diseñar el controlador PI se tomó en cuenta la función transferencia en lazo abierto mostrada (3.21), la cual fue ingresada en el programa Matlab, en la herramienta sisotool, donde se prosedió a designar la estructura de control.

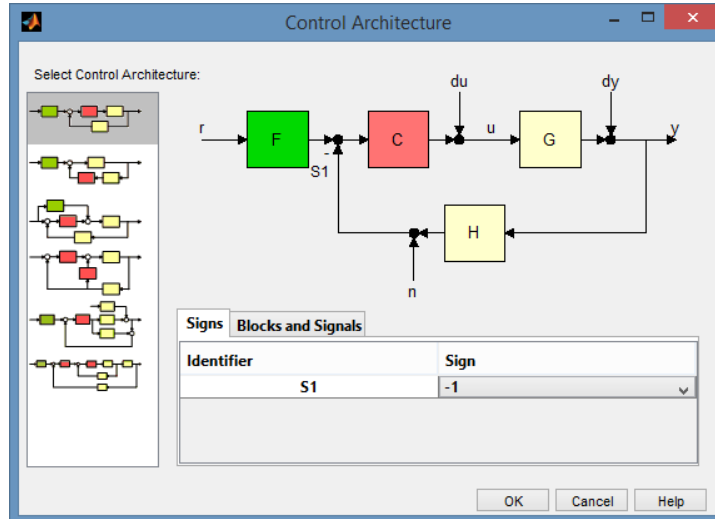


Figura 69. Estructura de control PI del motor DC M1

El diseño del controlador estuvo basado en el análisis del lugar geométrico de las raíces como se muestra en las siguientes figuras.

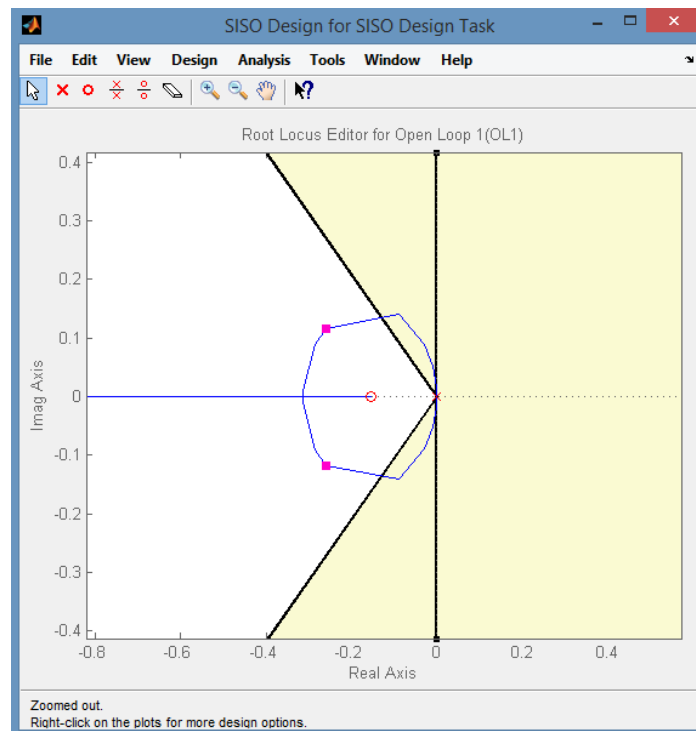


Figura 70. Lugar geométrico de las raíces del motor DC M1 con controlador PI

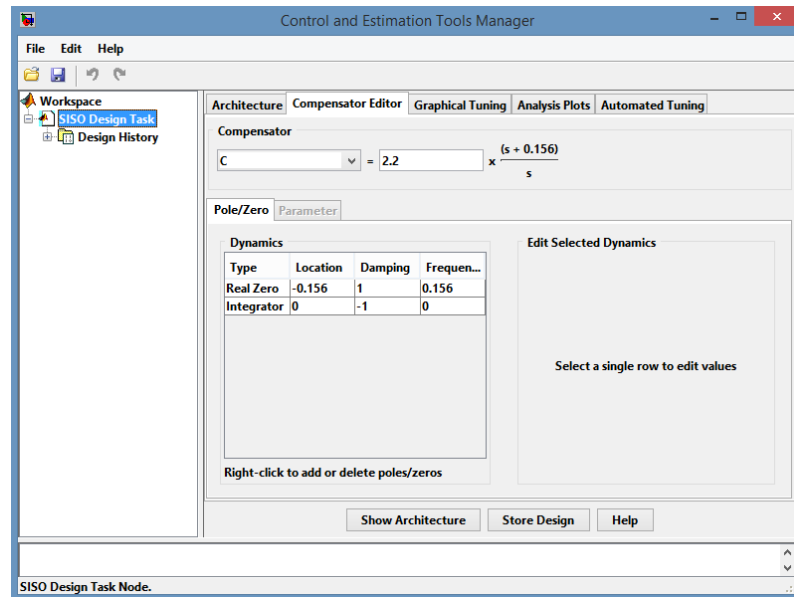


Figura 71. Controlador PI para el motor DC M1

Tomando en cuenta el compensador de la figura anterior se tiene que:

$$G_{M1}(s) = k_p \left(1 + \frac{1}{\tau_i s} \right)$$

$$G_{M1}(s) = k_p + \frac{k_p}{\tau_i s}$$

$$G_{M1} = k_p + \frac{k_i}{s}$$

$$G_{M1} = k_p \left(\frac{s + \frac{k_i}{k_p}}{s} \right)$$

$$C_{PI}(s) = 2.2 \left(\frac{s + 0.156}{s} \right)$$

Donde:

$$\frac{k_i}{k_p} = 0.156$$

Reemplazando:

$$k_p = 2.2$$

$$k_i = 0.3432$$

Usando Simulink se procedió a simular la función transferencia a lazo cerrado con el compensador diseñado agregando una entrada paso de 10 (cm) y la salida de control del compensador.

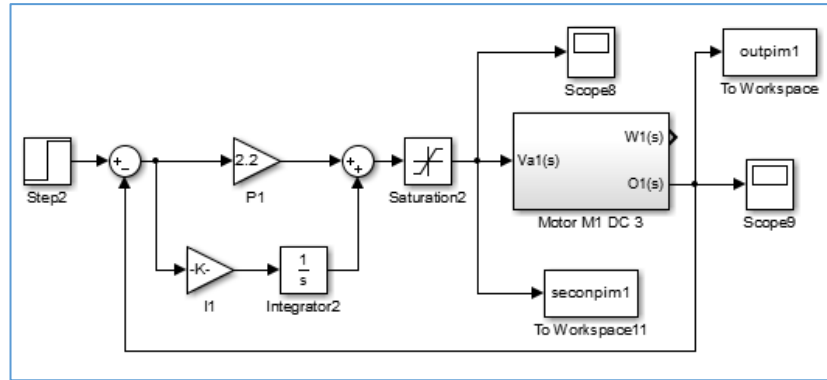


Figura 72. Diagrama de bloques para el control PI del motor DC M1

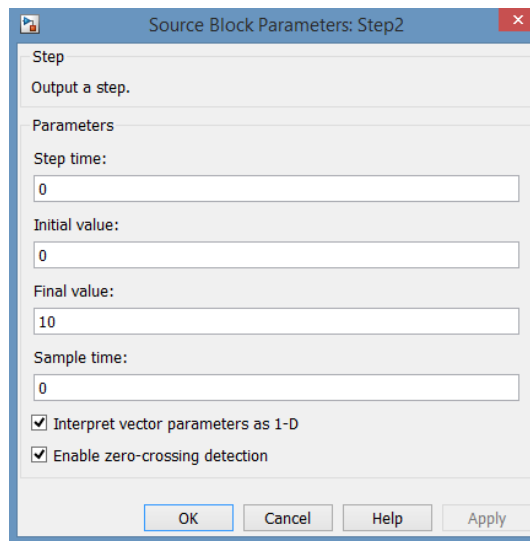


Figura 73. Configuración de entrada paso a 10cm

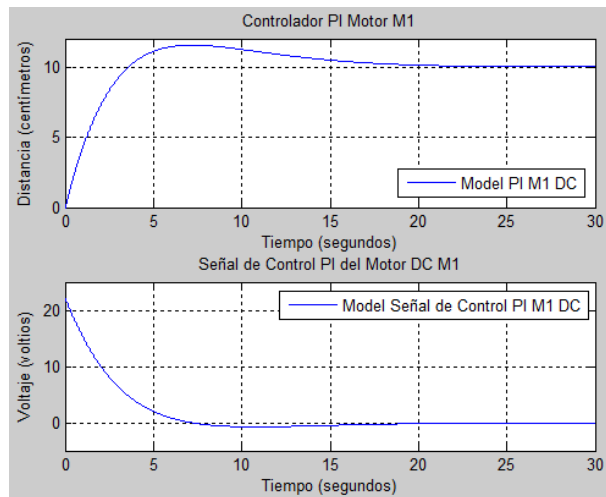


Figura 74. Respuesta de entrada paso y salida del controlador PI del motor M1

3.6.4. Diseño del controlador PID del motor DC M1

Para diseñar el controlador PID se tomó en cuenta la función transferencia en lazo abierto mostrada (3.21), la cual fue ingresada en el programa Matlab, en la herramienta sisotool, donde se prosedió a designar la estructura de control.

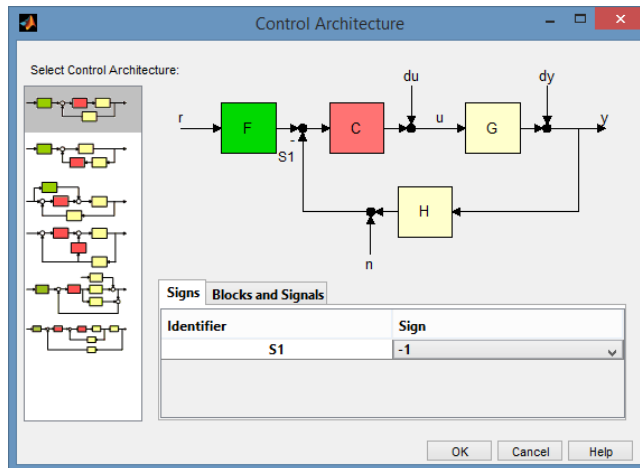


Figura 75. Estructura de control PID del motor DC M1

El diseño del controlador estuvo basado en el análisis del lugar geométrico de las raíces como se muestra en las siguientes figuras.

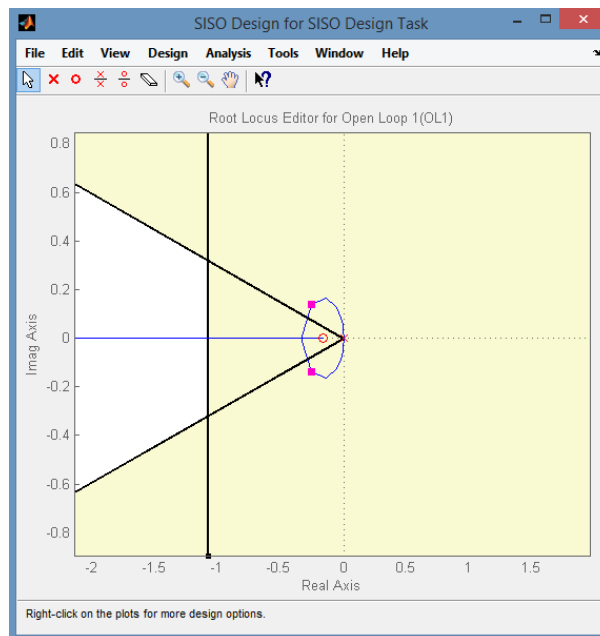


Figura 76. Lugar geométrico de las raíces del motor DC M1 con controlador PID

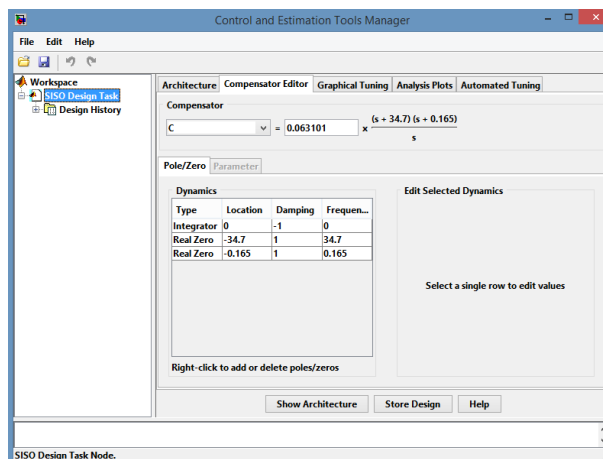


Figura 77. Controlador PID para el motor DC M1

Tomando en cuenta el compensador de la figura anterior se tiene que:

$$G_{M1}(s) = k_p \left(1 + \frac{1}{\tau_i s} + \tau_d s \right)$$

$$G_{M1}(s) = k_p + \frac{k_p}{\tau_i s} + k_p \tau_d s$$

$$G_{M1} = k_p + \frac{k_i}{s} + k_d s$$

$$G_{M1} = k_d \left(\frac{s^2 + \frac{k_p}{k_d} s + \frac{k_i}{k_d}}{s} \right)$$

$$C_{PID}(s) = 0.063101 \frac{(s + 34.7)(s + 0.165)}{s}$$

$$C_{PID}(s) = 0.063101 \left(\frac{s^2 + 34.865s + 5.7255}{s} \right)$$

Donde:

$$\frac{k_p}{k_d} = 34.865$$

$$\frac{k_i}{k_d} = 5.7255$$

Reemplazando:

$$k_d = 0.063101$$

$$k_p = 2.2$$

$$k_i = 0.3612847755$$

Usando Simulink se procedió a simular la función transferencia a lazo cerrado con el compensador diseñado agregando una entrada paso de 10 (cm) y la salida de control del compensador.

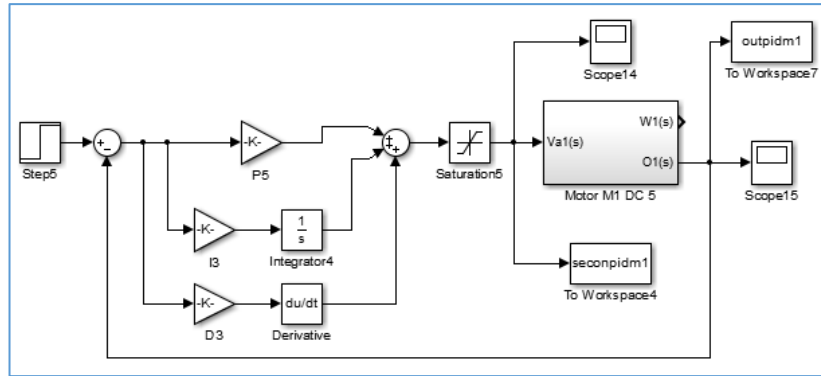


Figura 78. Diagrama de bloques para el control PID del motor DC M1

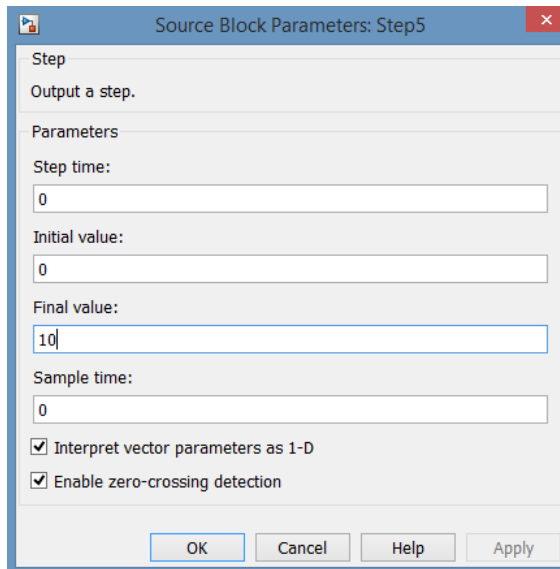


Figura 79. Configuración de entrada paso a 10cm

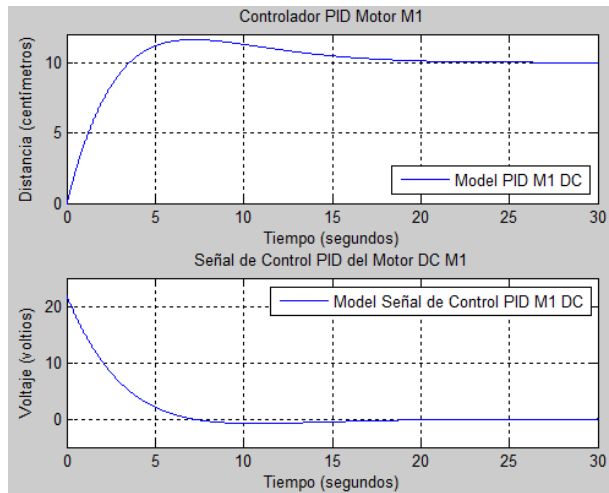


Figura 80. Respuesta de entrada paso y salida PID del motor M1

3.6.5. Diseño del controlador P del motor DC M2

Para diseñar el controlador P se tomó en cuenta la función transferencia en lazo abierto mostrada (3.22), la cual fue ingresada en el programa Matlab, en la herramienta sisotool, donde se prosedió a designar la estructura de control.

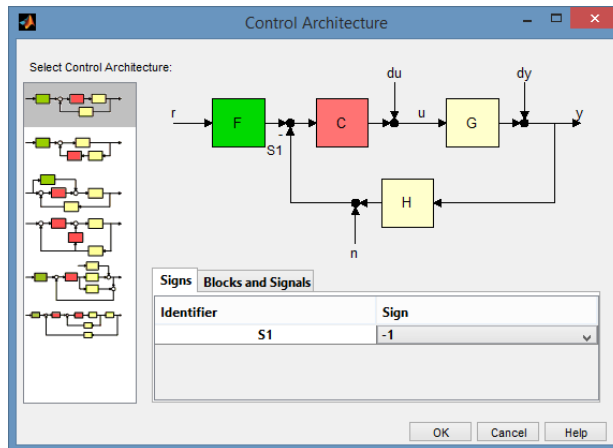


Figura 81. Estructura de control P del motor DC M2

El diseño del controlador estuvo basado en el análisis del lugar geométrico de las raíces como se muestra en las siguientes figuras.

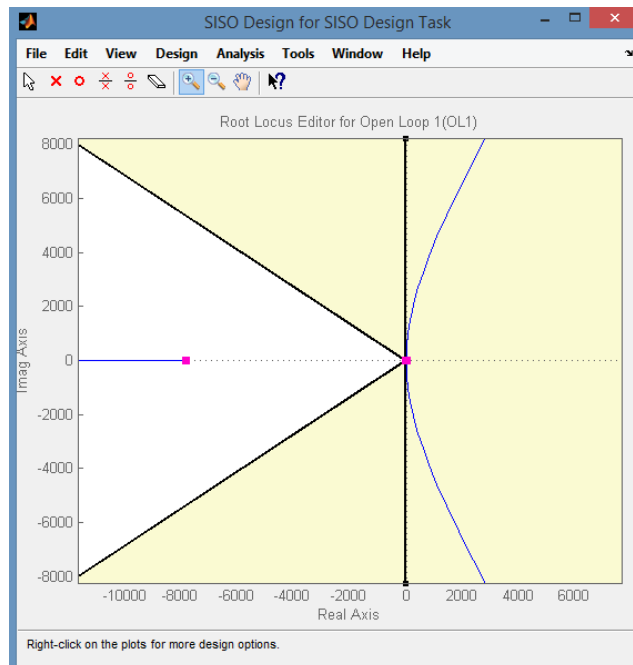


Figura 82. Lugar geométrico de las raíces del motor DC M2 con controlador P

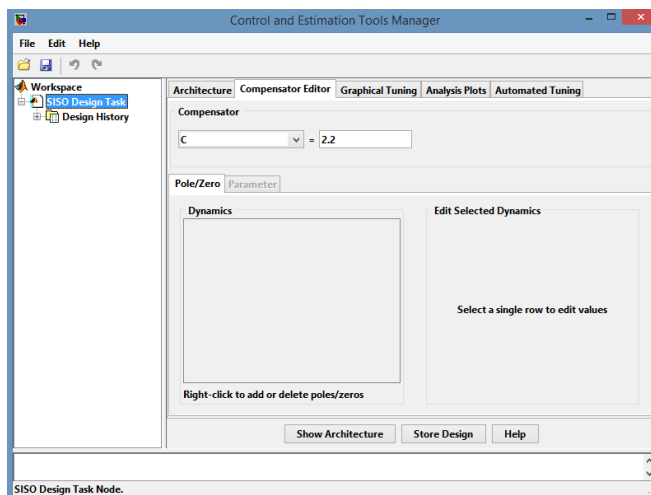


Figura 83. Controlador P para el motor DC M2

Tomando en cuenta el compensador de la figura anterior se tiene que:

$$k_p = 2.2$$

Usando Simulink se procedió a simular la función transferencia a lazo cerrado con el compensador diseñado agregando una entrada paso de 10 (cm) y la salida de control del compensador.

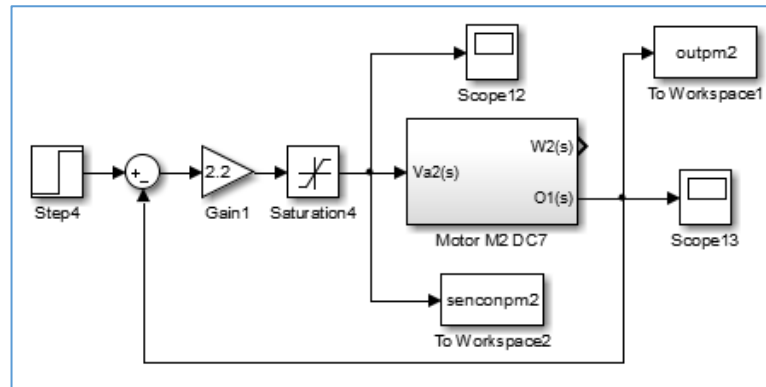


Figura 84. Diagrama de bloques para el control P del motor DC M2

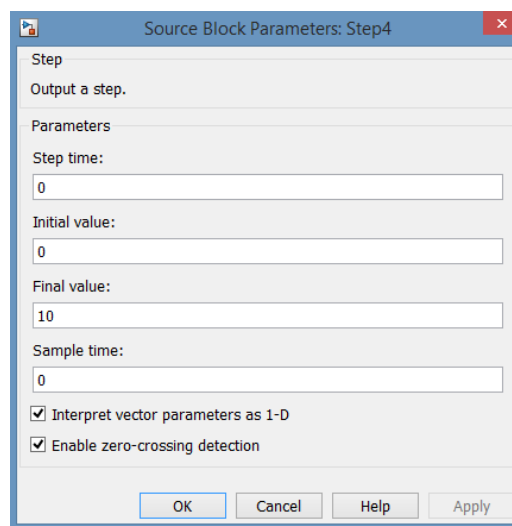


Figura 85. Configuración de entrada paso a 10cm

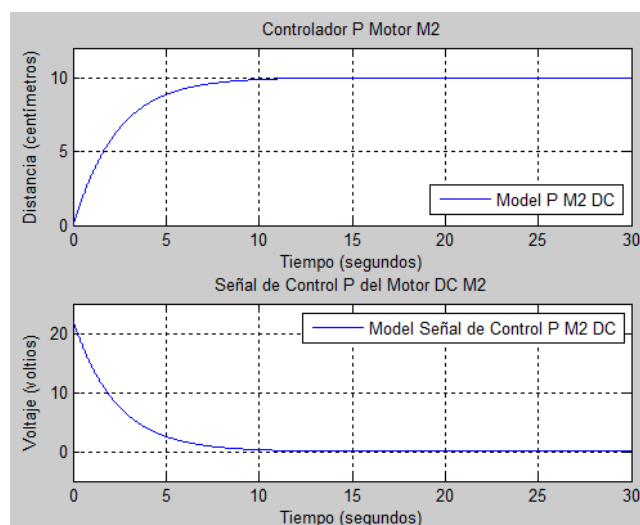


Figura 86. Respuesta de entrada paso y salida del controlador P del motor M2

3.6.6. Diseño del controlador PD del motor DC M2

Para diseñar el controlador PD se tomó en cuenta la función transferencia en lazo abierto mostrada (3.22), la cual fue ingresada en el programa Matlab, en la herramienta sisotool, donde se prosedió a designar la estructura de control.

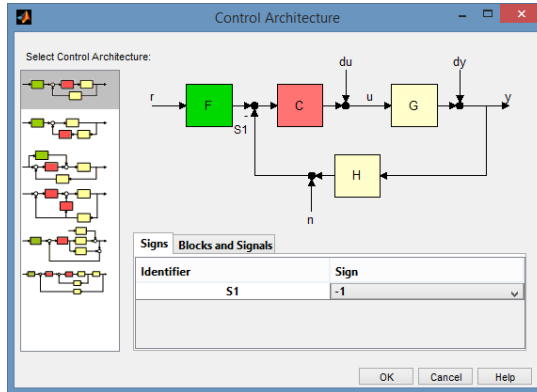


Figura 87. Estructura de control PD del motor DC M2

El diseño del controlador estuvo basado en el análisis del lugar geométrico de las raíces como se muestra en las siguientes figuras.

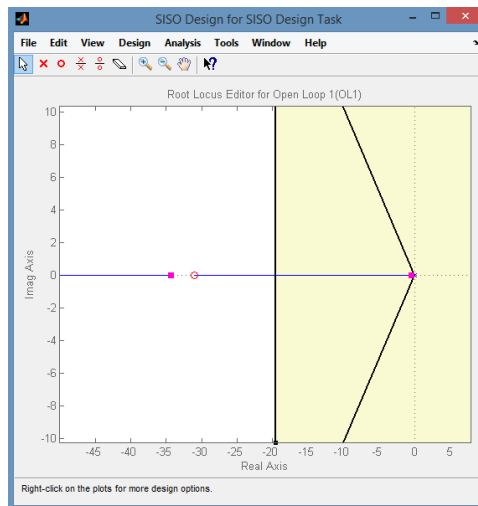


Figura 88. Lugar geométrico de las raíces del motor DC M2 con controlador PD

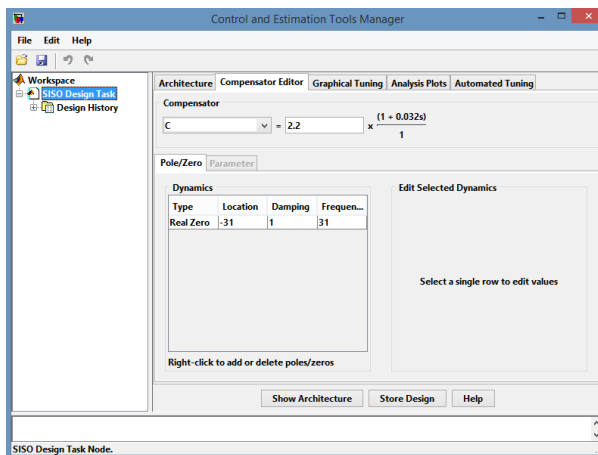


Figura 89. Controlador PD para el motor DC M2

Tomando en cuenta el compensador de la figura anterior se tiene que:

$$G_{M2}(s) = k_p(1 + \tau_d s)$$

$$G_{M2}(s) = k_p + k_p \tau_d s$$

$$G_{M2} = k_p + k_d s$$

$$C_{PD}(s) = 2.2(1 + 0.032s)$$

Donde:

$$C_{PD}(s) = 2.2 + 0.032s$$

Reemplazando:

$$k_p = 2.2$$

$$k_d = 0.0704$$

Usando Simulink se procedió a simular la función transferencia a lazo cerrado con el compensador diseñado agregando una entrada paso de 10 (cm) y la salida de control del compensador.

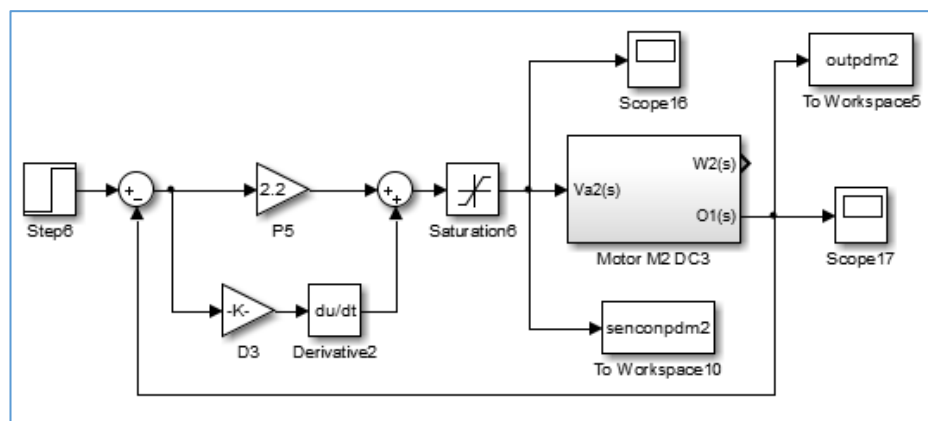


Figura 90. Diagrama de bloques para el control PD del motor DC M2

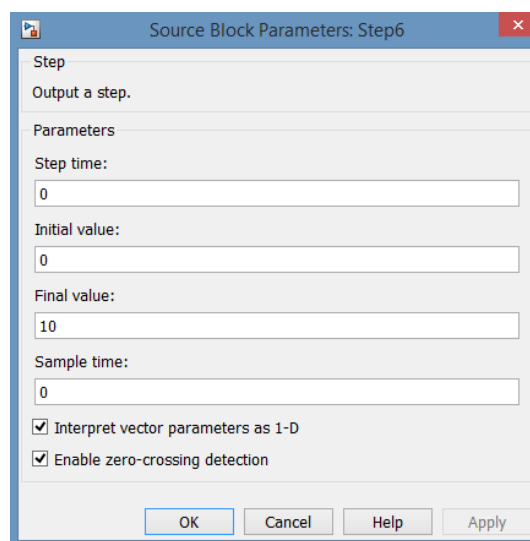


Figura 91. Configuración de entrada paso a 10cm

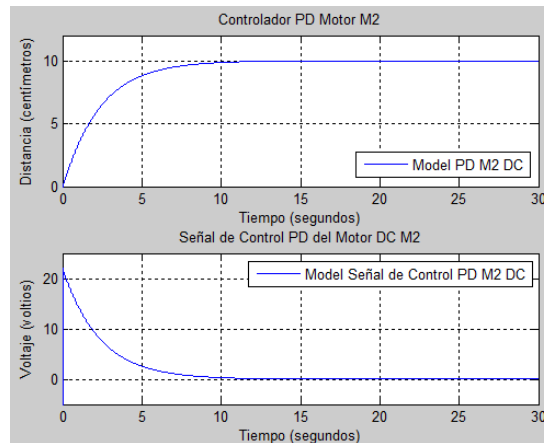


Figura 92. Respuesta de entrada paso y salida PD del motor DC M2

3.6.7. Diseño del controlador PI del motor DC M2

Para diseñar el controlador PI se tomó en cuenta la función transferencia en lazo abierto mostrada (3.22), la cual fue ingresada en el programa Matlab, en la herramienta sisotool, donde se prosedió a designar la estructura de control.

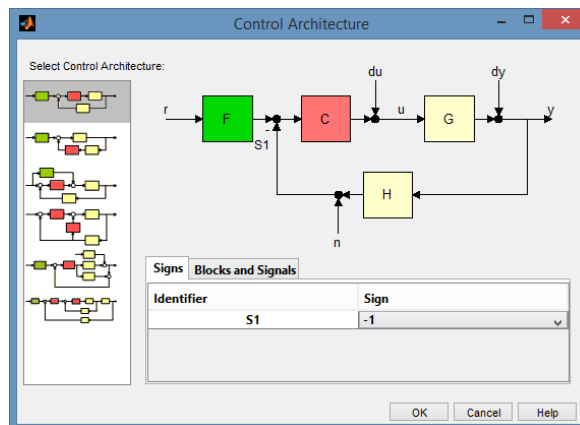


Figura 93. Estructura de control PI del motor DC M2

El diseño del controlador estuvo basado en el análisis del lugar geométrico de las raíces como se muestra en las siguientes figuras.

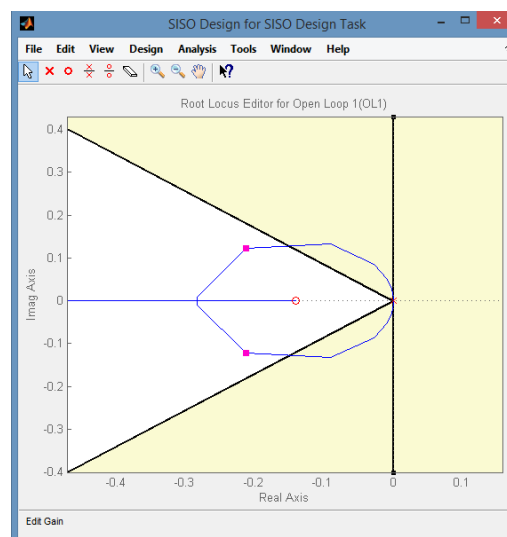


Figura 94. Lugar geométrico de las raíces del motor DC M2 con controlador PI

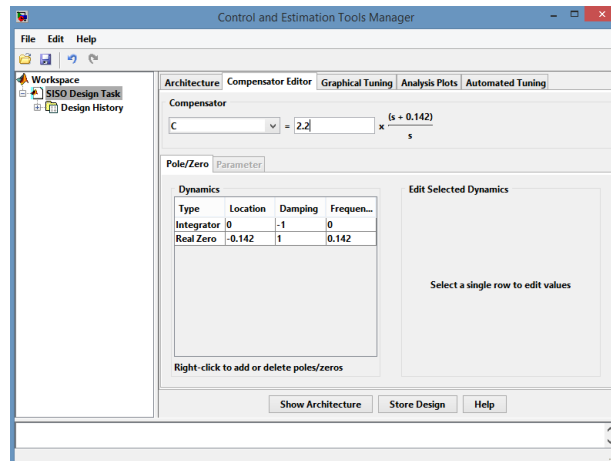


Figura 95. Controlador PI para el motor DC M2

Tomando en cuenta el compensador de la figura anterior se tiene que:

$$G_{M2}(s) = k_p \left(1 + \frac{1}{\tau_i s} \right)$$

$$G_{M2}(s) = k_p + \frac{k_p}{\tau_i s}$$

$$G_{M2} = k_p + \frac{k_i}{s}$$

$$G_{M2} = k_p \left(\frac{s + \frac{k_i}{k_p}}{s} \right)$$

$$C_{PI}(s) = 2.2 \left(\frac{s + 0.142}{s} \right)$$

Donde:

$$\frac{k_i}{k_p} = 0.142$$

Reemplazando:

$$k_p = 2.2$$

$$k_i = 0.3124$$

Usando Simulink se procedió a simular la función transferencia a lazo cerrado con el compensador diseñado agregando una entrada paso de 10 (cm) y la salida de control del compensador.

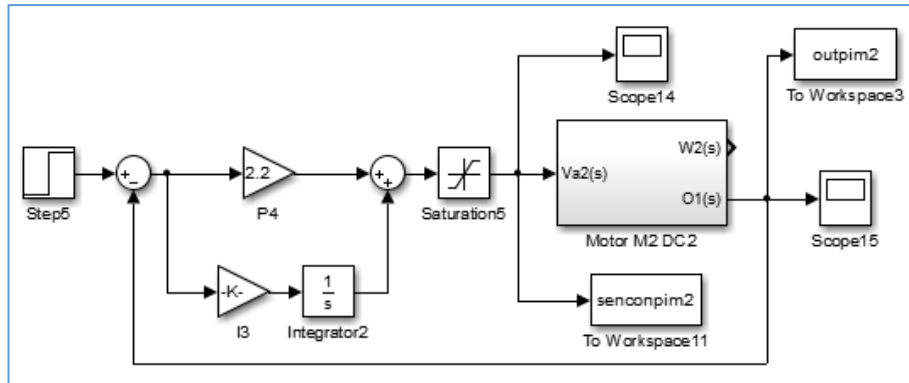


Figura 96. Diagrama de bloques para el control PI del motor DC M2

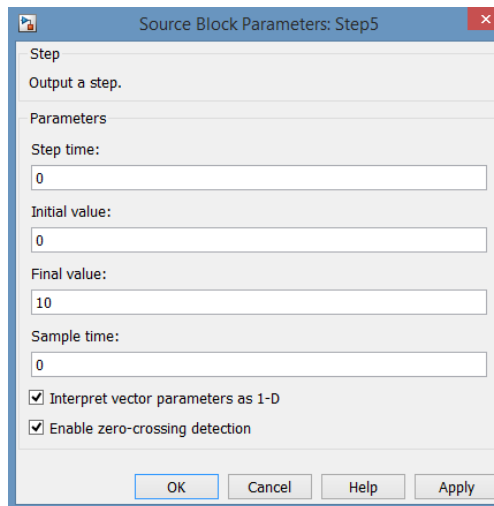


Figura 97. Configuración de entrada paso a 10cm

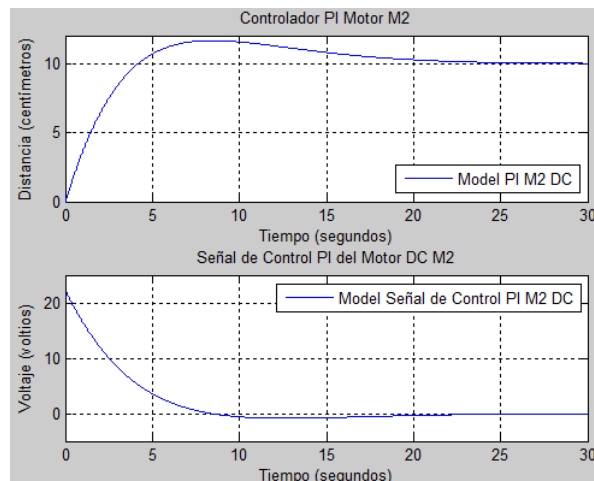


Figura 98. Respuesta de entrada paso y salida PI del motor DC M2

3.6.8. Diseño del controlador PID del motor DC M2

Para diseñar el controlador PID se tomó en cuenta la función transferencia en lazo abierto mostrada (3.22), la cual fue ingresada en el programa Matlab, en la herramienta sisotool, donde se procedió a designar la estructura de control.

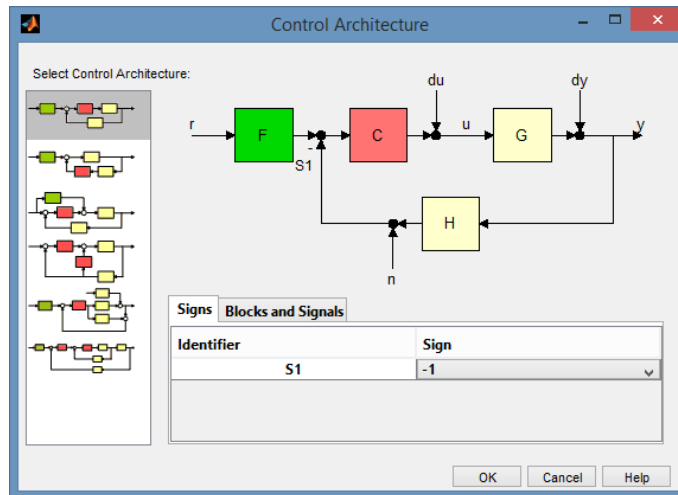


Figura 99. Estructura de control PID del motor DC M2

El diseño del controlador estuvo basado en el análisis del lugar geométrico de las raíces como se muestra en las siguientes figuras.

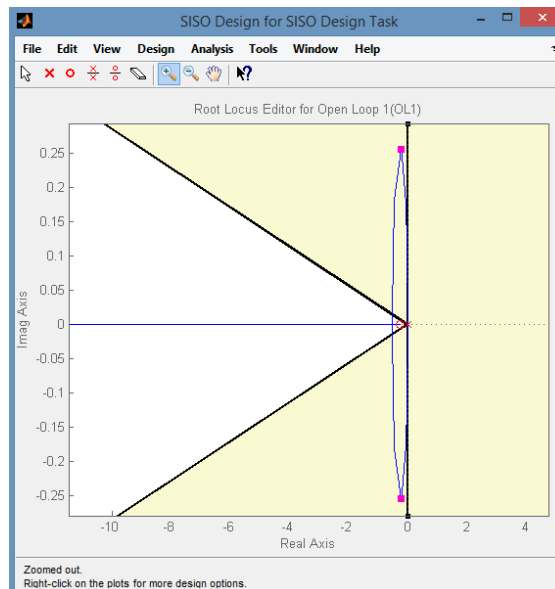


Figura 100. Lugar geométrico de las raíces del motor DC M2 con controlador PID

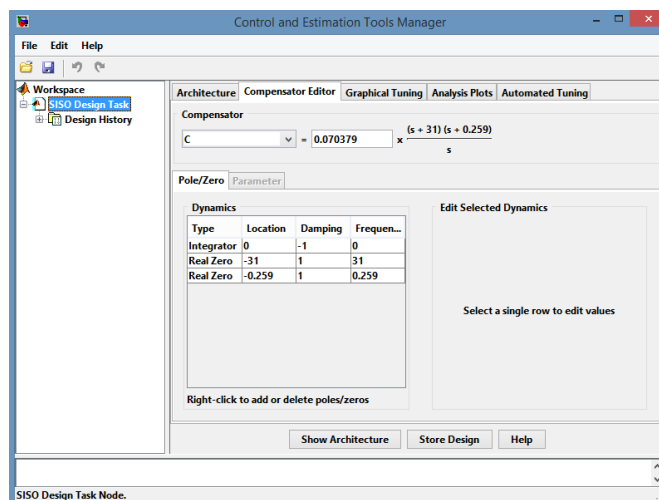


Figura 101. Controlador PID para el motor DC M2

Tomando en cuenta el compensador de la figura anterior se tiene que:

$$G_{M2}(s) = k_p \left(1 + \frac{1}{\tau_i s} + \tau_d s \right)$$

$$G_{M2}(s) = k_p + \frac{k_p}{\tau_i s} + k_p \tau_d s$$

$$G_{M2} = k_p + \frac{k_i}{s} + k_d s$$

$$G_{M2} = k_d \left(\frac{s^2 + \frac{k_p}{k_d} s + \frac{k_i}{k_d}}{s} \right)$$

$$C_{PID}(s) = 0.070379 \frac{(s + 31)(s + 0.259)}{s}$$

$$C_{PID}(s) = 0.070379 \left(\frac{s^2 + 31.259s + 8.029}{s} \right)$$

Donde:

$$\frac{k_p}{k_d} = 31.259$$

$$\frac{k_i}{k_d} = 8.029$$

Reemplazando:

$$k_d = 0.070379$$

$$k_p = 2.2$$

$$k_i = 0.565072991$$

Usando Simulink se procedió a simular la función transferencia a lazo cerrado con el compensador diseñado agregando una entrada paso de 10 (cm) y la salida de control del compensador.

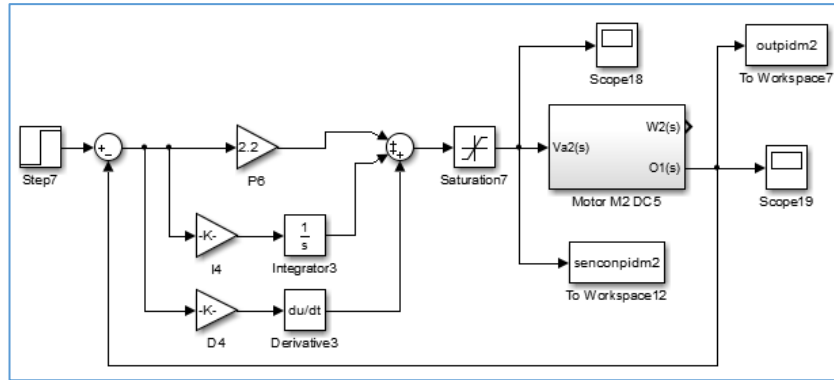


Figura 102. Diagrama de bloques para el control PID del motor DC M2

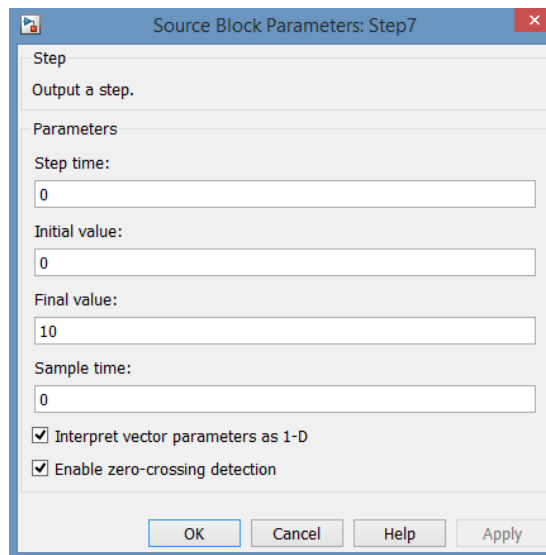


Figura 103. Configuración de entrada paso a 10cm

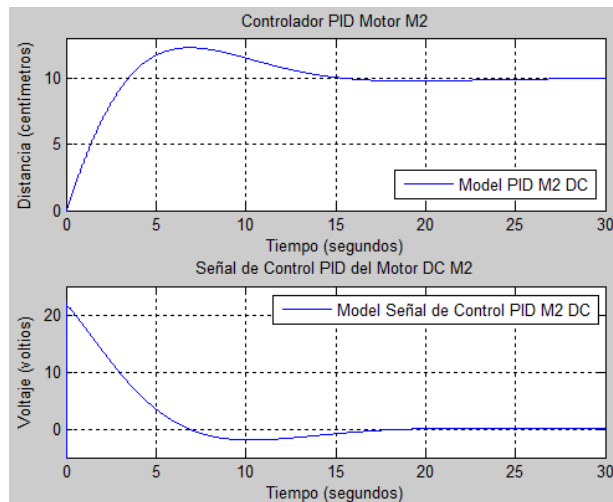


Figura 104. Respuesta de entrada paso y salida PID del motor DC M2

3.7. Validación de controladores PID de los motores DC M1 y M2

Para las pruebas de validación de los controladores PID de los motores DC M1 y M2, se utilizó el software TIA PORTAL donde por medio de la función CONT_C el cual es un regulador continuo. Se estableció una conexión ETHERNET con el PLC S7-300 mediante el módulo de comunicación CP 343-1 Lean.

También se utilizó el software LabView y el software KEPserverEX para la obtención de los datos reales de distancia y de la salida del controlador de los motores DC M1 y M2.

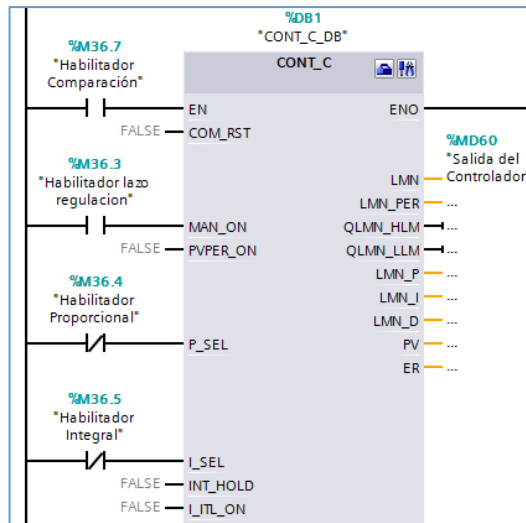


Figura 105. Función regulador continuo.

A partir de la configuración interna del controlador continuo CONT_C, se procedió a modificar en base a los diseños de los controladores realizados en el apartado anterior, los valores integrales y derivativos de los controladores.

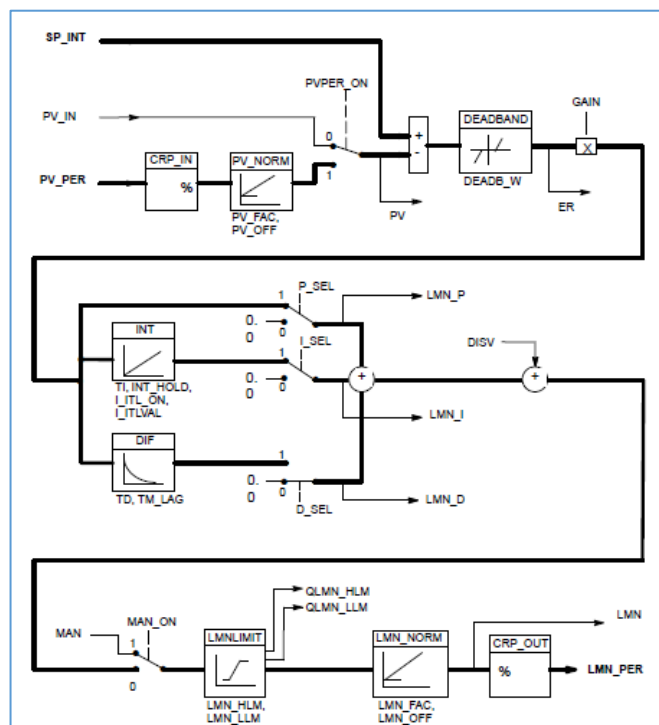


Figura 106. Configuración interna del controlador continuo (Siemens AG, 1996)

Tag Name	Address	Data Type	Scan Rate	Scaling	Description
CONTROL	MD60	Long	10	None	
ENCODER	MD28	Long	10	None	
INICIO	I125.0	Boolean	100	None	

Figura 107. Variables de datos en KEPServerEX

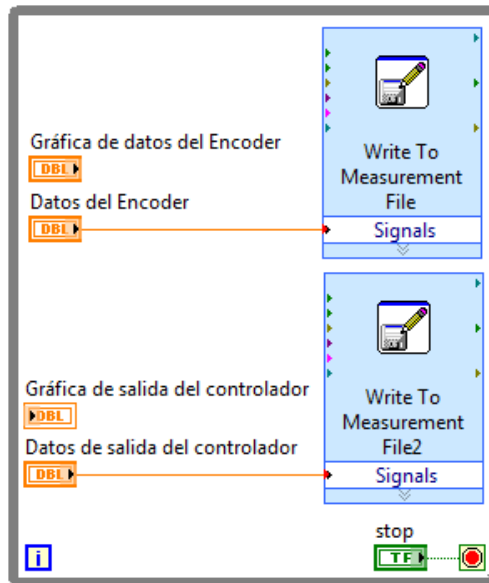


Figura 108. Datos reales de controladores de motores DC M1 y M2

3.7.1. Prueba de validación del controlador P del motor DC M1

Para la validación del controlador P del motor DC M1 se configuró la función CONT_C en TIA PORTAL desactivando la parte integral y derivativa. Se añadió un Set Point de 10 (cm) y el valor de la ganancia es de $K_p = 2.2$ tal cual como se diseñó en el capítulo anterior.

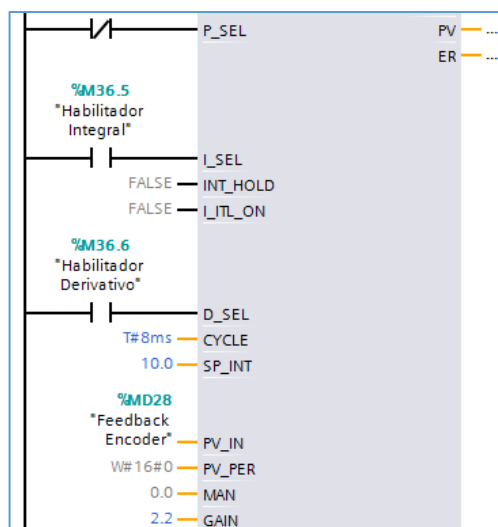


Figura 109. Configuración del controlador P del motor DC M1

Mediante el programa MatLab se procesaron los datos obtenidos del motor DC M1 con el controlador P y la salida del controlador obteniendo las gráficas correspondientes.

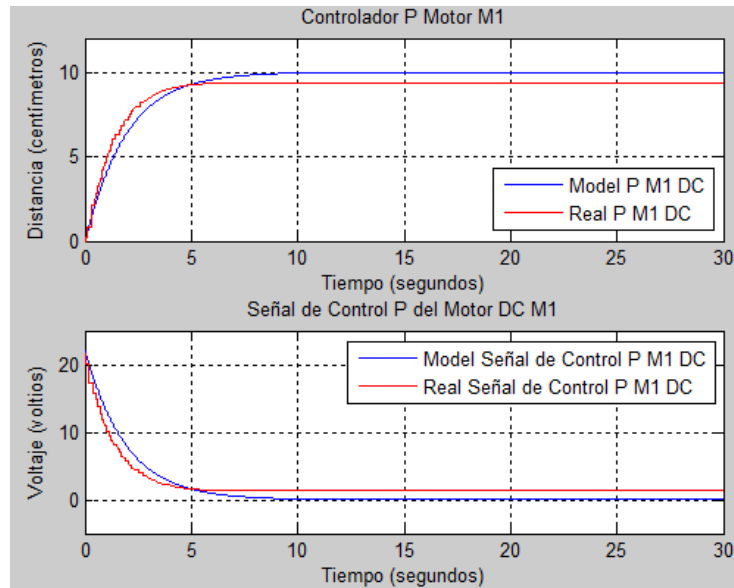


Figura 110. Comparación del controlador P del motor DC M1

Se pudo observar que hubo un error de estado estacionario, para poder simular dicho error se añadió un bloque llamado Dead Zone (zona muerta), en la simulación donde se obtuvo la siguiente gráfica.

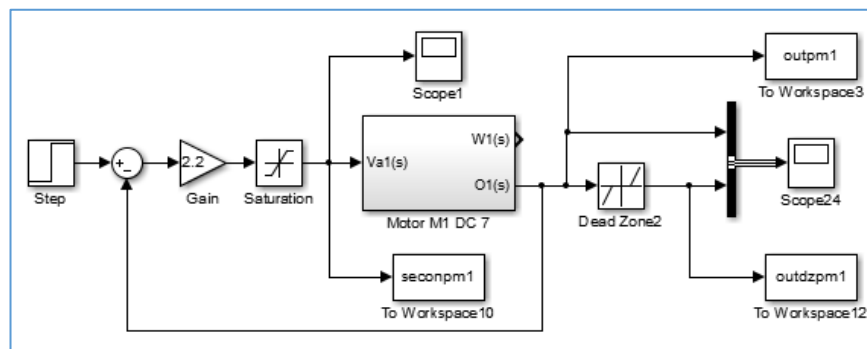


Figura 111. Diagrama de bloques del control P y zona muerta del motor DC M1

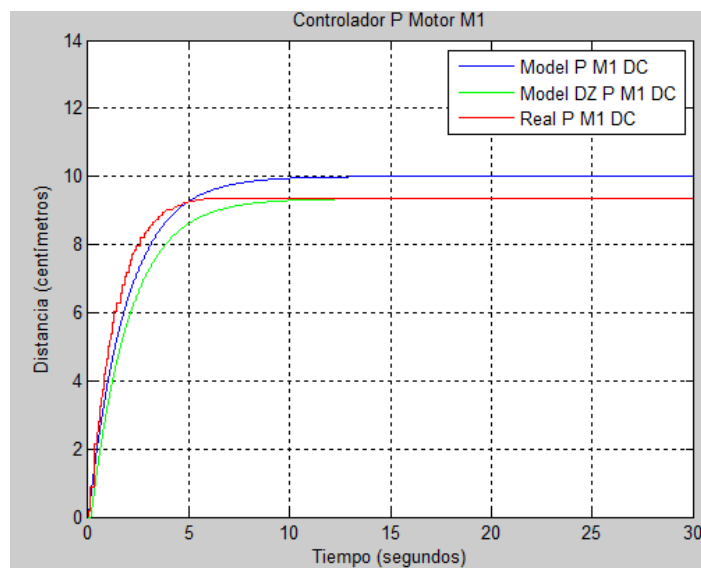


Figura 112. Comparación del controlador P para el motor DC M1

3.7.2. Prueba de validación del controlador PD del motor DC M1

Para la validación del controlador PD del motor DC M1 se configuró la función CONT_C en TIA PORTAL desactivando la parte integral.

Se añadió un Set Point de 10 (cm), el valor de la ganancia y de la parte derivativa se obtuvieron, en el apartado anterior el diseño del controlador PD del motor DC M1 y a partir de ello se obtuvo la constante de tiempo derivativo.

$$10(0.055) = 0.55$$

$$22k_d = 0.55$$

$$k_d = 0.025$$

Donde:

$$k_p = 2.2$$

$$k_d = 0.025$$

$$\tau_d = \frac{k_d}{k_p}$$

Reemplazando:

$$\tau_d = 0.013636$$

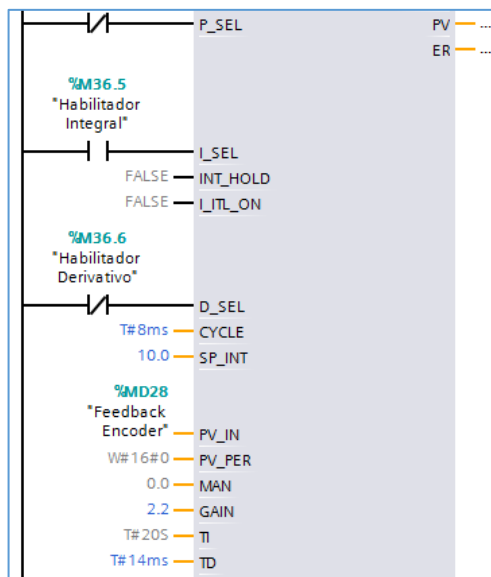


Figura 113. Configuración del controlador PD del motor DC M1

Se modificó el diagrama de bloques en Simulink de acuerdo a la configuración establecida del regulador continuo CONT_C del TIA PORTAL.

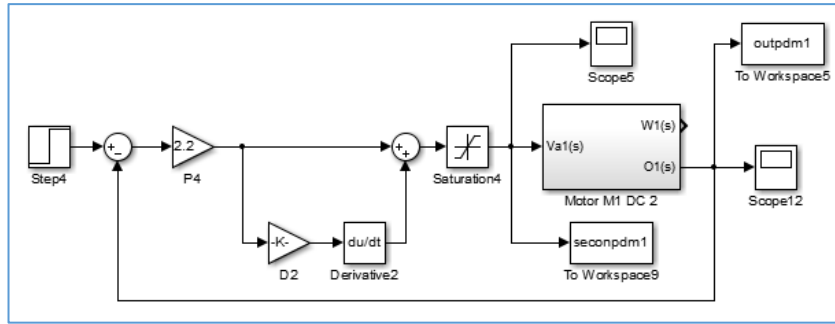


Figura 114. Diagrama de bloques del control PD del motor DC M1

Mediante el programa MATLAB se procesaron los datos obtenidos del motor DC M1 con el controlador PD y de la salida del controlador obteniendo las gráficas correspondientes.

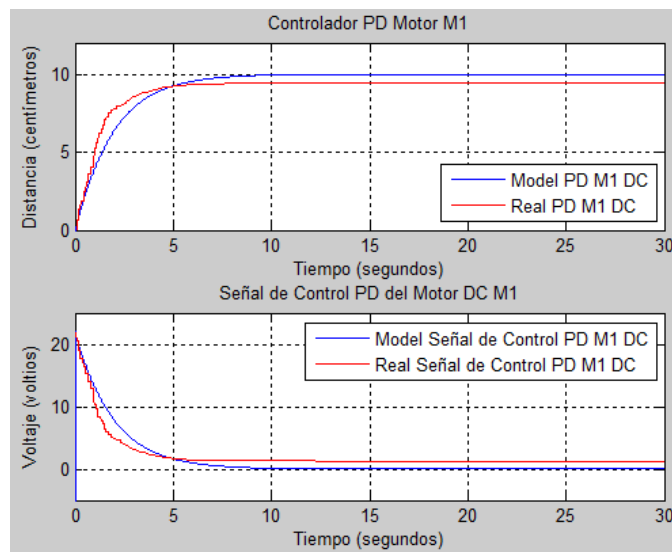


Figura 115. Comparación del controlador PD del motor DC M1

Se pudo observar que hubo un error de estado estacionario, para poder simular dicho error se añadió un bloque llamado Dead Zone (zona muerta) en la simulación, en donde se obtuvo la siguiente gráfica.

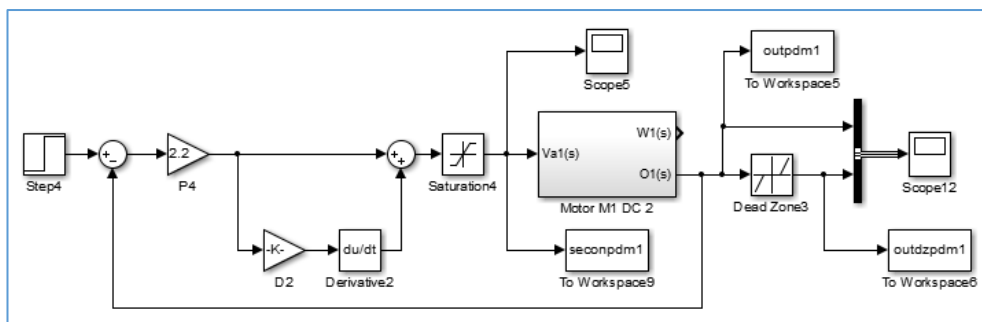


Figura 116. Diagrama de bloques del control PD y zona muerta del motor DC M1

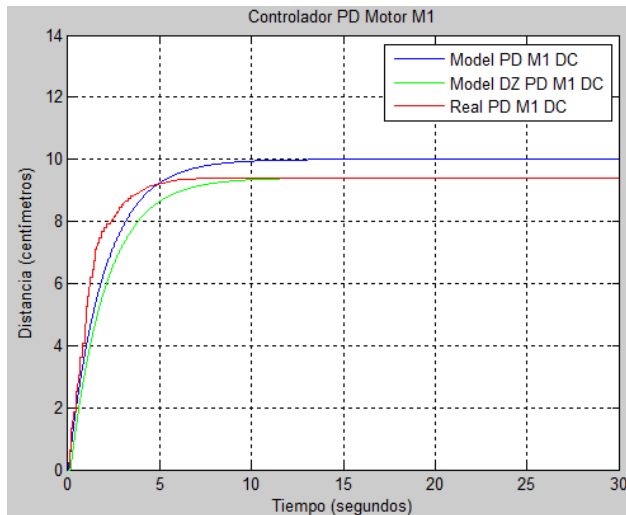


Figura 117. Comparación del controlador PD para el motor DC M1

3.7.3. Prueba de validación del controlador PI del motor DC M1

Para la validación del controlador PI del motor DC M1 se configuró la función CONT_C en TIA PORTAL desactivando la parte integral. Se añadió un Set Point de 10 (cm), el valor de la ganancia y de la parte integral se obtuvieron en el apartado anterior en el diseño del controlador PI del motor DC M1 y a partir de ello se obtuvo la constante de tiempo integral.

$$10(0.3432) = 3.432$$

$$22k_i = 3.432$$

$$k_i = 0.156$$

Donde:

$$k_p = 2.2$$

$$k_i = 0.156$$

$$\tau_i = \frac{k_p}{k_i}$$

Reemplazando:

$$\tau_i = 14.10256$$

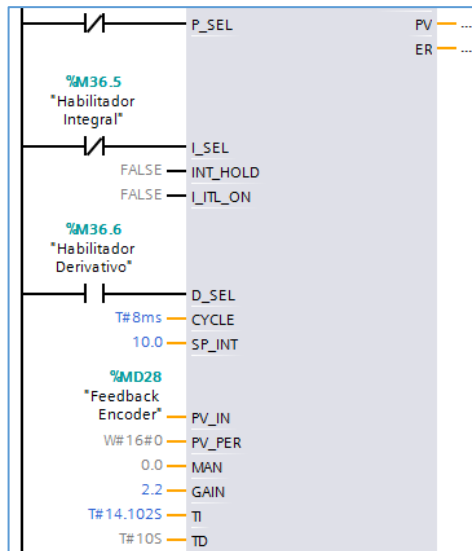


Figura 118. Configuración del controlador PI del motor DC M1

Se modificó el diagrama de bloques en Simulink de acuerdo a la configuración establecida del regulador continuo CONT_C de TIA PORTAL.

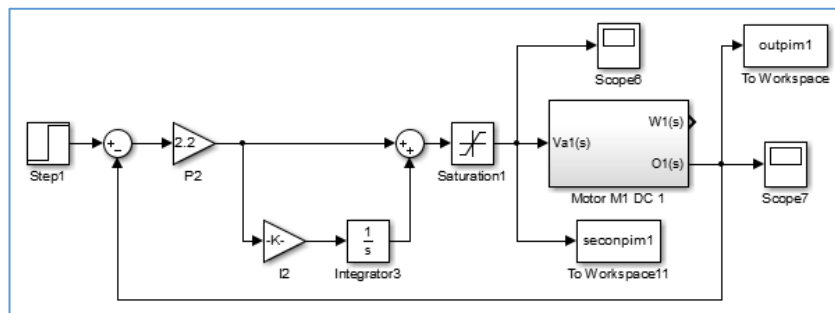


Figura 119. Diagrama de bloques para el control PI del motor DC M1

Mediante el programa MATLAB se procesaron los datos obtenidos del motor DC M1 con el controlador PI y de la salida del controlador obteniendo las gráficas correspondientes.

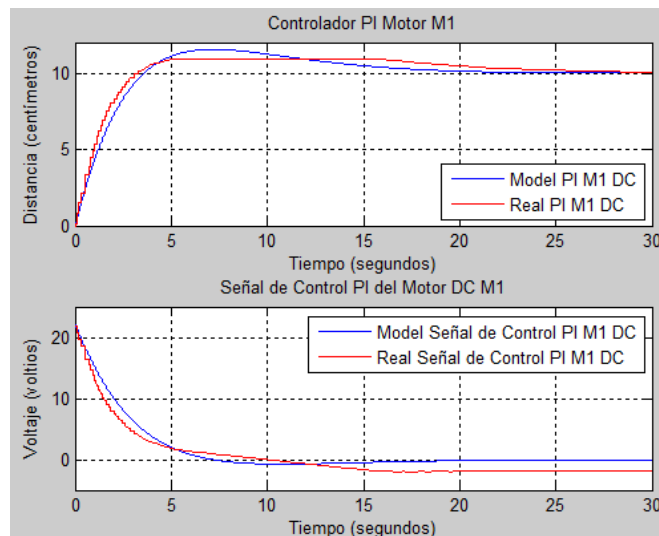


Figura 120. Comparación del controlador PI del motor DC M1

3.7.4. Prueba de validación del controlador PID del motor DC M1

Para la validación del controlador PID del motor DC M1 se configuró la función CONT_C en TIA PORTAL.

Se añadió un Set Point de 10 (cm), el valor de la ganancia, la parte integral y la parte derivativa que se obtuvieron en el apartado anterior en el diseño del controlador PID del motor DC M1 y a partir de ello se obtuvo la constante de tiempo derivativo y la constante de tiempo integral.

$$10(0.3612847755) = 3.612847755$$

$$22k_i = 3.612847755$$

$$k_i = 0.1642203525$$

$$10(0.063101) = 0.63101$$

$$22k_d = 0.63101$$

$$k_d = 0.0286822$$

Donde:

$$k_p = 2.2$$

$$k_i = 0.1642203525$$

$$k_d = 0.0286822$$

$$\tau_i = \frac{k_p}{k_i}$$

$$\tau_d = \frac{k_d}{k_p}$$

Reemplazando:

$$\tau_i = 13.39663$$

$$\tau_d = 0.0130373$$

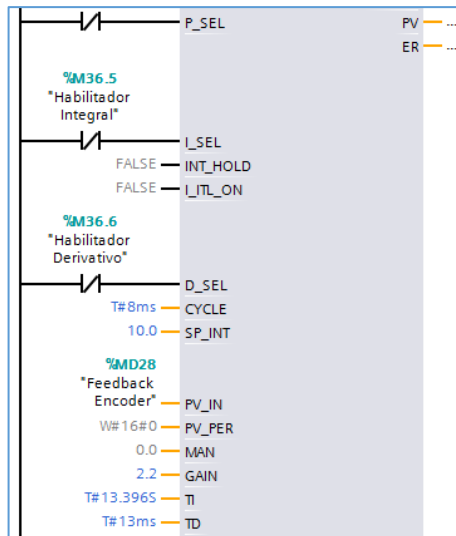


Figura 121. Configuración del controlador PID del motor DC M1

Se modificó el diagrama de bloques en Simulink de acuerdo a la configuración establecida del regulador continuo CONT_C de TIA PORTAL.

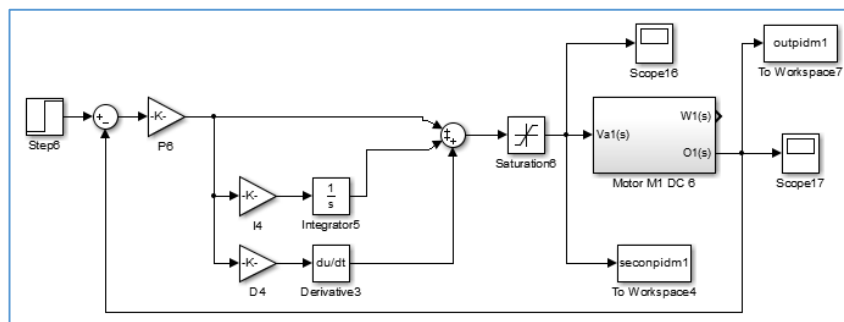


Figura 122. Diagrama de bloques para el control PID del motor DC M1

Mediante el programa MATLAB se procesaron los datos obtenidos del motor DC M1 con el controlador PID y de la salida del controlador obteniendo las gráficas correspondientes.

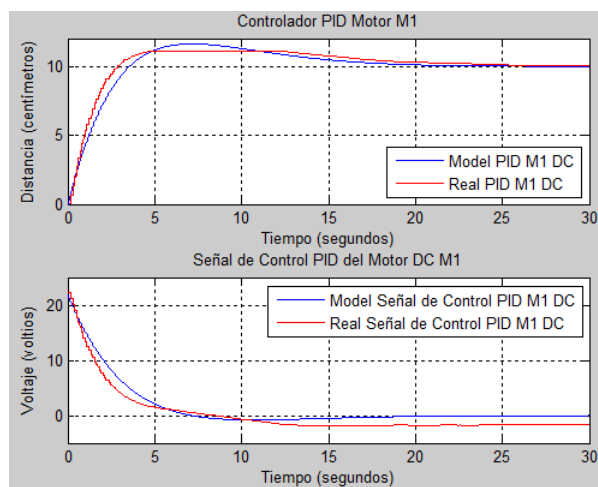


Figura 123. Comparación del controlador PID del motor DC M1

3.7.5. Prueba de validación del controlador P del motor DC M2

Para la validación del controlador P del motor DC M2 se configuró la función CONT_C en TIA PORTAL desactivando la parte integral y derivativa.

Se añadió un Set Point de 10 (cm) y el valor de la ganancia es de $K_p = 2.2$ tal cual como se diseñó en el capítulo anterior.

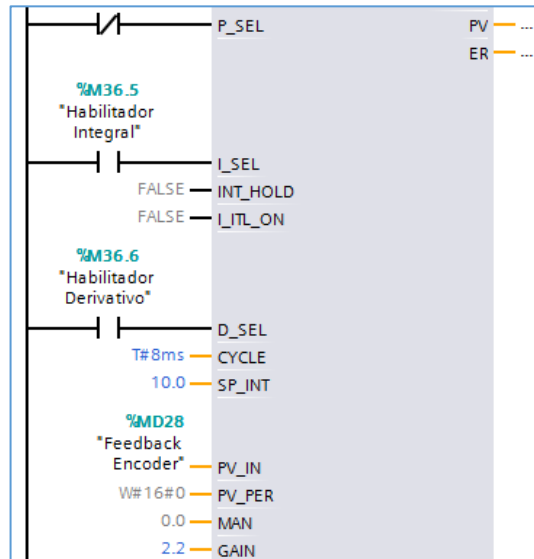


Figura 124. Configuración del controlador P del motor DC M2

Mediante el programa MATLAB se procesaron los datos obtenidos del motor DC M2 con el controlador P y de la salida del controlador obteniendo las gráficas correspondientes.

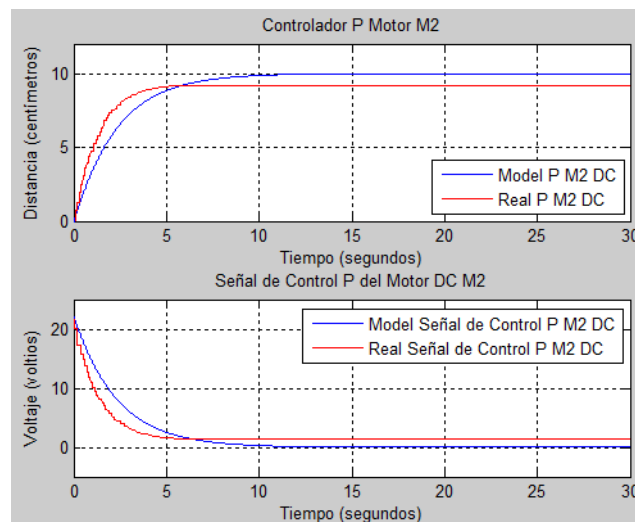


Figura 125. Comparación del controlador P del motor DC M2

Se pudo observar que hubo un error de estado estacionario, para poder simular dicho error se añadió un bloque llamado Dead Zone (zona muerta) en la simulación donde se obtuvo la siguiente gráfica.

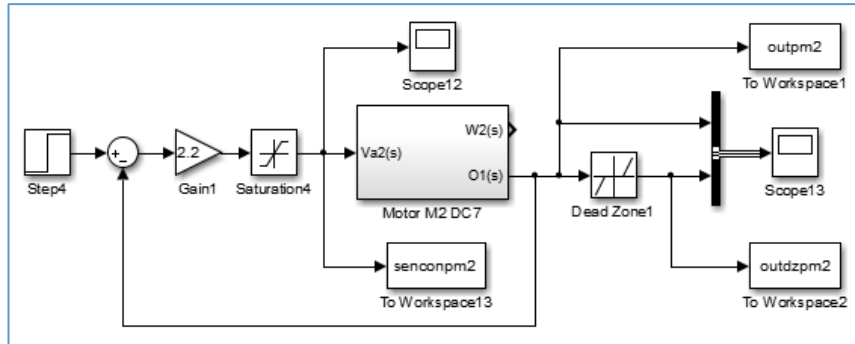


Figura 126. Diagrama de bloques del control P y zona muerta del motor DC M2

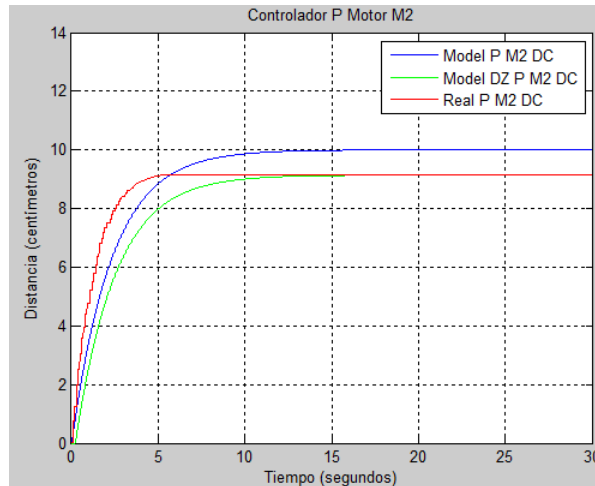


Figura 127. Comparación del controlador P del motor DC M2

3.7.6. Prueba de validación del controlador PD del motor DC M2

Para la validación del controlador PD del motor DC M2 se configuró la función CONT_C en TIA PORTAL desactivando la parte integral.

Se añadió un Set Point de 10 (cm), el valor de la ganancia y de la parte derivativa que se obtuvieron en el apartado anterior en el diseño del controlador PD del motor DC M2 y a partir de ello se obtuvo la constante de tiempo derivativo.

$$10(0.0704) = 0.704$$

$$22k_d = 0.704$$

$$k_d = 0.032$$

Donde:

$$k_p = 2.2$$

$$k_d = 0.032$$

$$\tau_d = \frac{k_d}{k_p}$$

Reemplazando:

$$\tau_d = 0.01454$$

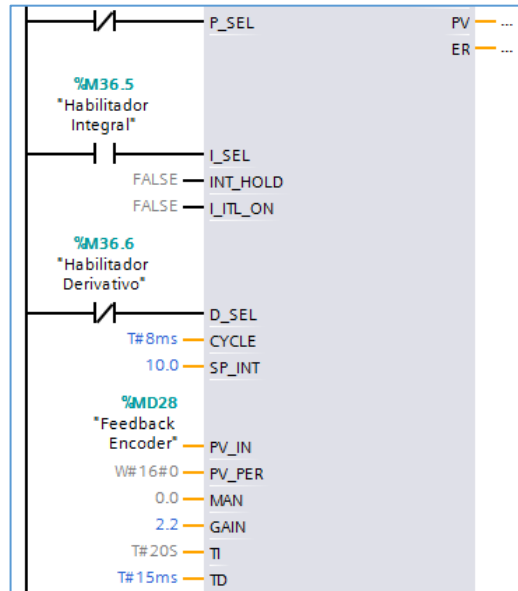


Figura 128. Configuración del controlador PD del motor DC M2

Se modificó el diagrama de bloques en Simulink de acuerdo a la configuración establecida del regulador continuo CONT_C de TIA PORTAL.

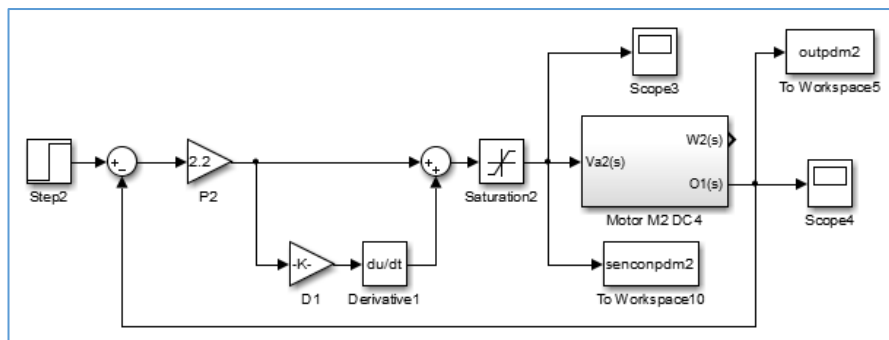


Figura 129. Diagrama de bloques para el control PD del motor DC M2

Mediante el programa MATLAB se procesaron los datos obtenidos del motor DC M2 con el controlador PD y de la salida del controlador obteniendo las gráficas correspondientes.

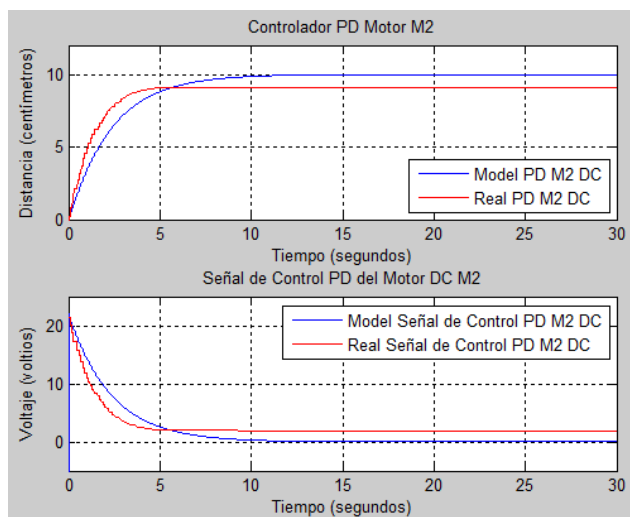


Figura 130. Comparación del controlador PD del motor DC M2

Se pudo observar que hubo un error de estado estacionario, para poder simular dicho error se añadió un bloque llamado Dead Zone (zona muerta), en la simulación donde se obtuvo la siguiente gráfica.

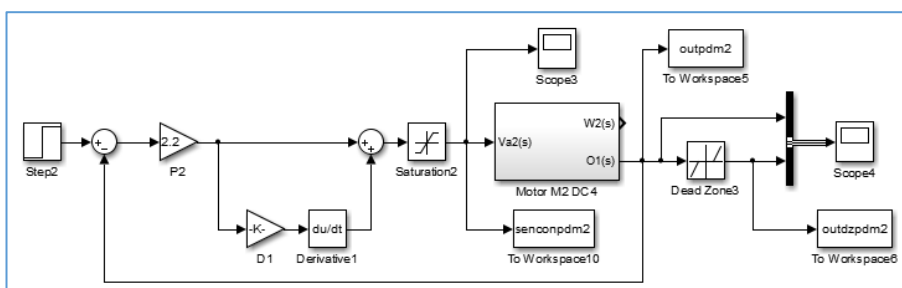


Figura 131. Diagrama de bloques del control PD y zona muerta del motor DC M2

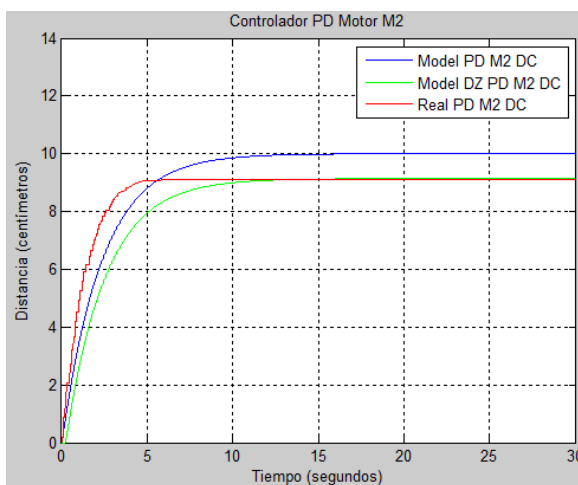


Figura 132. Comparación del controlador PD para el motor DC M2

3.7.7. Prueba de validación del controlador PI del motor DC M2

Para la validación del controlador PI del motor DC M2 se configuró la función CONT_C en TIA PORTAL desactivando la parte integral.

Se añadió un Set Point de 10 (cm), el valor de la ganancia y de la parte integral se obtuvieron en el apartado anterior en el diseño del controlador PI del motor DC M2 y a partir de ello se obtuvo la constante de tiempo integral.

$$10(0.3124) = 3.124$$

$$22k_i = 3.124$$

$$k_i = 0.142$$

Donde:

$$k_p = 2.2$$

$$k_i = 0.142$$

$$\tau_i = \frac{k_p}{k_i}$$

Reemplazando:

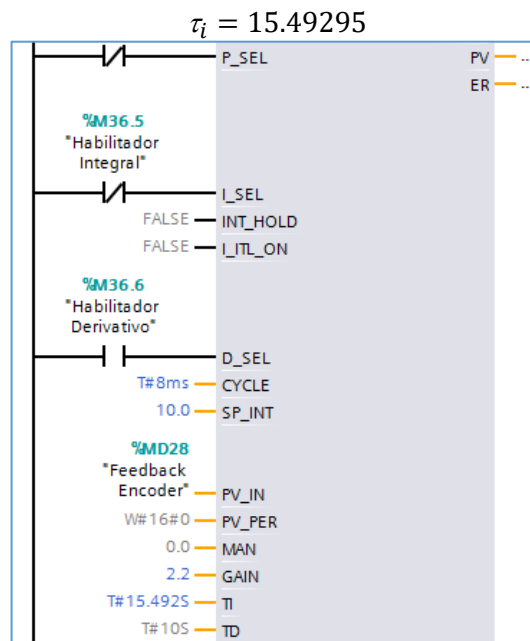


Figura 133. Configuración del controlador PI del motor DC M2

Se modificó el diagrama de bloques en Simulink de acuerdo a la configuración establecida del regulador continuo CONT_C de TIA PORTAL.

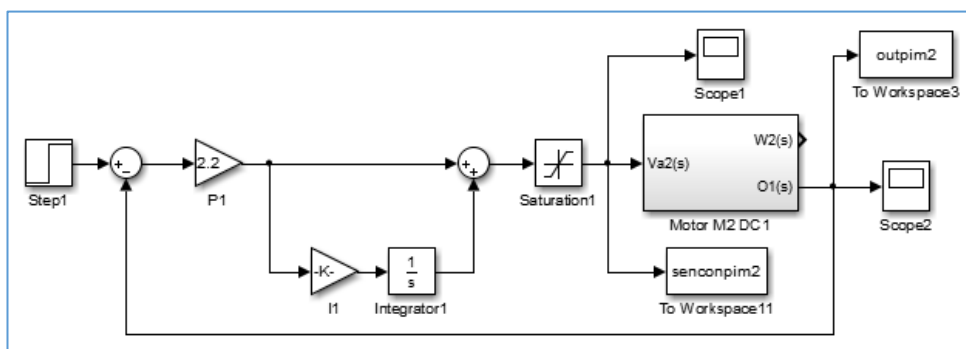


Figura 134. Diagrama de bloques del control PI del motor DC M2

Mediante el programa MATLAB se procesaron los datos obtenidos del motor DC M2 con el controlador PI y de la salida del controlador obteniendo las gráficas correspondientes.

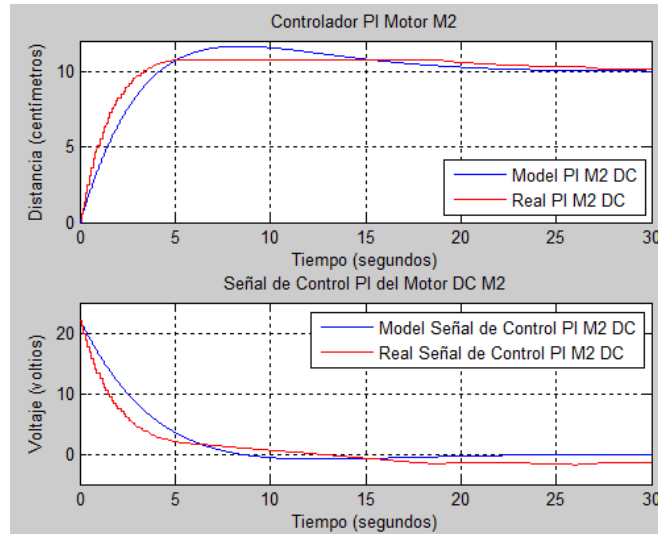


Figura 135. Comparación del controlador PI del motor DC M2

3.7.8. Prueba de validación del controlador PID del motor DC M2

Para la validación del controlador PID del motor DC M2 se configuró la función CONT_C en TIA PORTAL.

Se añadió un Set Point de 10 (cm), el valor de la ganancia, la parte integral y la parte derivativa que se obtuvieron en el apartado anterior en el diseño del controlador PID del motor DC M2 y a partir de ello se obtuvo la constante de tiempo derivativo y la constante de tiempo integral.

$$10(0.565072991) = 5.65072991$$

$$22k_i = 5.65072991$$

$$k_i = 0.2568513595$$

$$10(0.070379) = 0.70379$$

$$22k_d = 0.70379$$

$$k_d = 0.031990454$$

Donde:

$$k_p = 2.2$$

$$k_i = 0.2568513595$$

$$k_d = 0.031990454$$

$$\tau_i = \frac{k_p}{k_i}$$

$$\tau_d = \frac{k_d}{k_p}$$

Reemplazando:

$$\tau_i = 8.565265$$

$$\tau_d = 0.014541$$

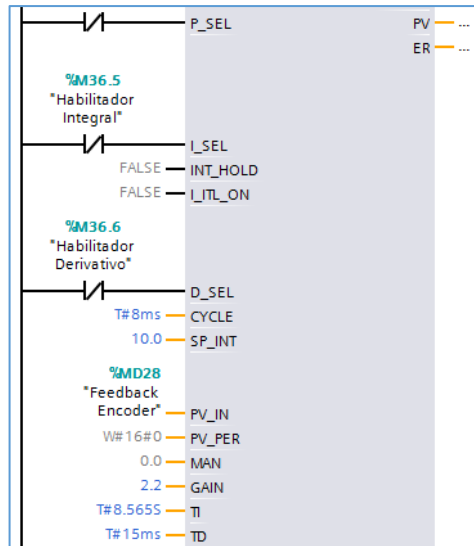


Figura 136. Configuración del controlador PID del motor DC M2

Se modificó el diagrama de bloques en Simulink de acuerdo a la configuración establecida del regulador continuo CONT_C de TIA PORTAL.

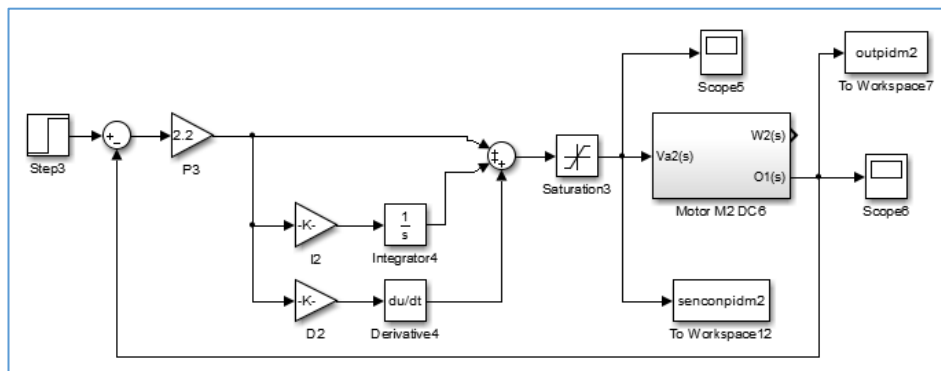


Figura 137. Diagrama de bloques del control PID del motor DC M2

Mediante el programa MATLAB se procesaron los datos obtenidos del motor DC M2 con el controlador PID y de la salida del controlador obteniendo las gráficas correspondientes.

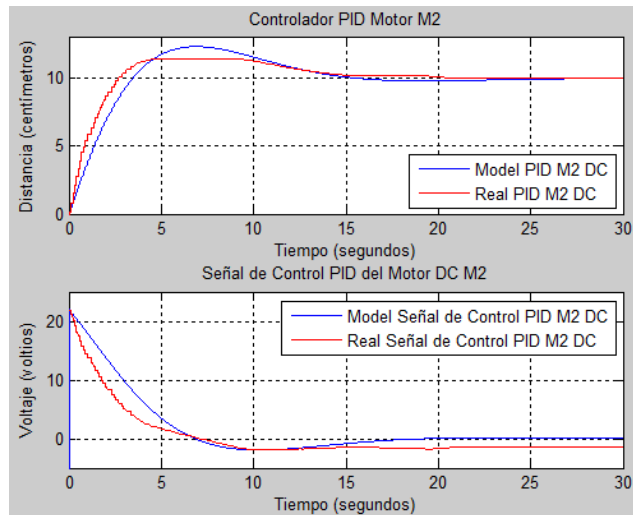


Figura 138. Comparación del controlador PID del motor DC M2

3.8. Diseño de controladores fuzzy de los motores DC M1 y M2

En este capítulo se realizó el diseño de los controlados Fuzzy de distancia a los motores DC M1 y M2.

Para diseñar los controladores se tomó en cuenta las distancias máximas en centímetros que los motores podían recorrer y el voltaje nominal de ambos motores, las cuales fueron ingresadas en el programa Matlab, en la herramienta Fuzzy.

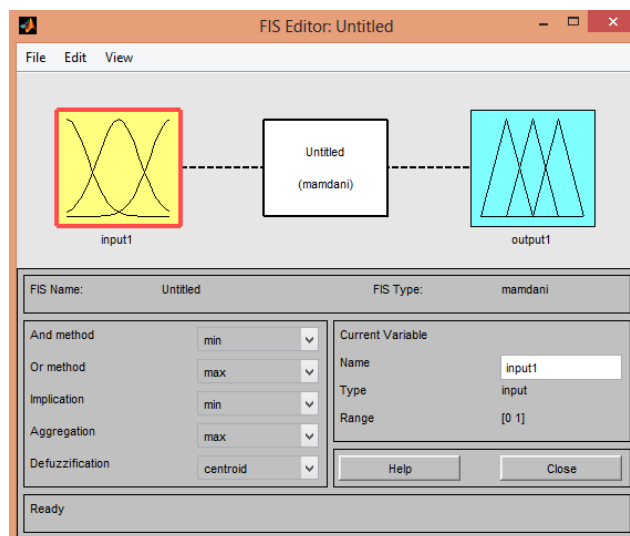


Figura 139. Herramienta fuzzy en MatLab.

Para el diseño de los controladores Fuzzy se fijó la salida de 23 voltios ya que el voltaje nominal de la tarjeta Sabertooth 2X60 y de los motores DC M1 y M2 es de 24 voltios.

3.8.1. Diseño del controlador fuzzy del motor DC M1

3.8.1.1. Diseño de la entrada del controlador fuzzy del motor DC M1

Para el diseño de la entrada del controlador Fuzzy del motor DC M1, se estableció como rango la distancia comprendida de -60 centímetros hasta 60 centímetros, el rango positivo es la máxima distancia que el motor DC M1 podía recorrer ya que al llegar a los 61 centímetros se impactaría con la misma estructura de la planta.

Se implementó cinco funciones de membresía para la entrada del controlador.

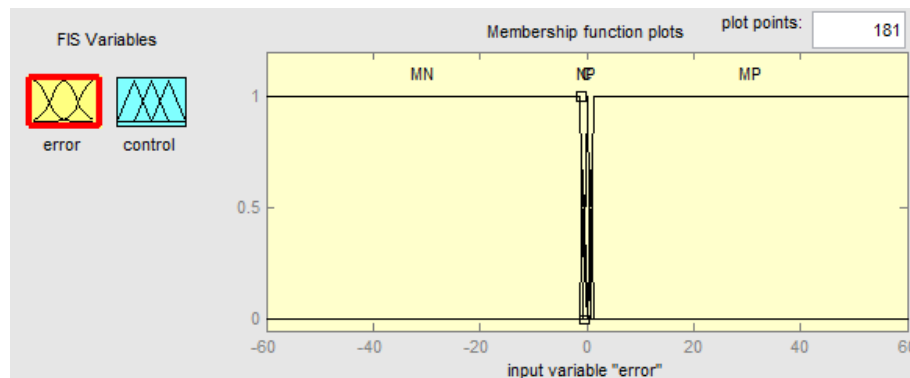


Figura 140. Diseño de entrada del controlador fuzzy del motor DC M1

La primera función de membresía considerada se la llamó MN, la cual es de forma trapezoidal que comprende el rango de -60 hasta -0.5.

La segunda función de membresía considerada se la llamó N, la cual es de forma triangular que comprende el rango de -1 hasta 0.

La tercera función de membresía considerada se la llamó C, la cual es de forma triangular que comprende el rango de -0.5 hasta 0.5.

La cuarta función de membresía considerada se la llamó P, la cual es de forma triangular que comprende el rango de 0 hasta 1.

La quinta función de membresía considerada se la llamó MP, la cual es de forma trapezoidal que comprende el rango de 0.5 hasta 60.

3.8.1.2. Diseño de la salida del controlador fuzzy del motor DC M1

Para el diseño de la salida del controlador Fuzzy del motor DC M1, se estableció como rango el voltaje comprendido de -23 voltios hasta 23 voltios, dicho rango es el voltaje máximo que se implementó al motor DC M1, ya que el voltaje nominal es de -24 voltios hasta 24 voltios, dejando 1 voltio de diferencia por seguridad del motor DC M1.

En este controlador no hay sobrepicos que puedan pasar el voltaje nominal del motor DC M1, con lo cual se puede trabajar con el voltaje cercano al nominal sin inconvenientes.

Se implementó cinco funciones de membresía para la salida del controlador.

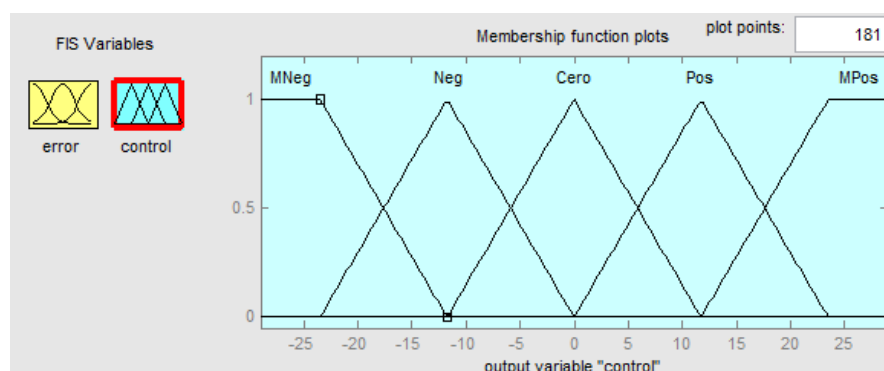


Figura 141. Diseño de salida del controlador fuzzy del motor DC M1

La primera función de membresía considerada se la llamó MNeg, la cual es de forma trapezoidal que comprende el rango de -29 hasta -11.75.

La segunda función de membresía considerada se la llamó Neg, la cual es de forma triangular que comprende el rango de -23.5 hasta 0.

La tercera función de membresía considerada se la llamó Cero, la cual es de forma triangular que comprende el rango de -11.75 hasta 11.75.

La cuarta función de membresía considerada se la llamó Pos, la cual es de forma triangular que comprende el rango de 0 hasta 23.5.

La quinta función de membresía considerada se la llamó MPos, la cual es de forma trapezoidal que comprende el rango de 11.75 hasta 29.

Usando Simulink se procedió a simular el controlador Fuzzy del motor DC M1 agregando una entrada paso de 60 (cm) y la salida de control del controlador.

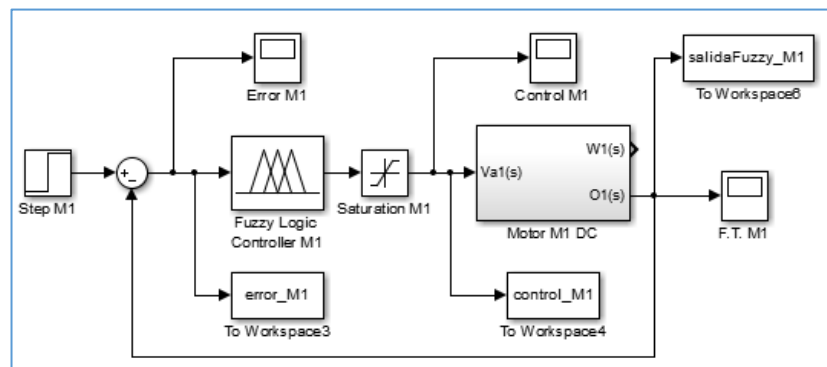


Figura 142. Diagrama de bloques del control fuzzy del motor DC M1

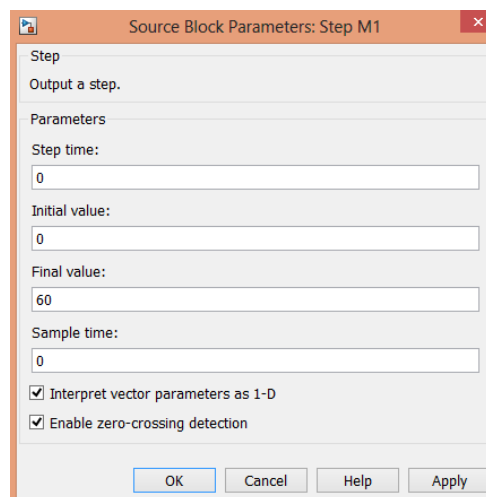


Figura 143. Configuración de entrada paso a 60cm

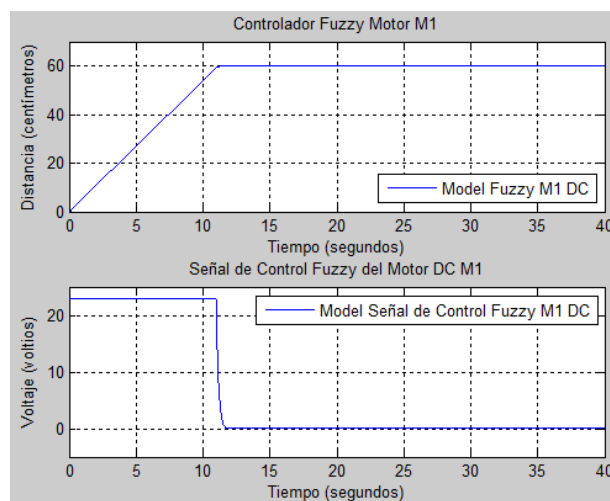


Figura 144. Respuesta de entrada paso y salida fuzzy del motor DC M1

3.8.2. Diseño del controlador fuzzy del motor DC M2

3.8.2.1. Diseño de la entrada del controlador fuzzy del motor DC M2

Para el diseño de la entrada del controlador Fuzzy del motor DC M2, se estableció como rango la distancia comprendida de -41 centímetros hasta 41 centímetros, el rango positivo es la máxima distancia que el motor DC M2 podía recorrer ya que la distancia máxima que podía recorrer fue de 42 centímetros.

Se implementó cinco funciones de membresía para la entrada del controlador.

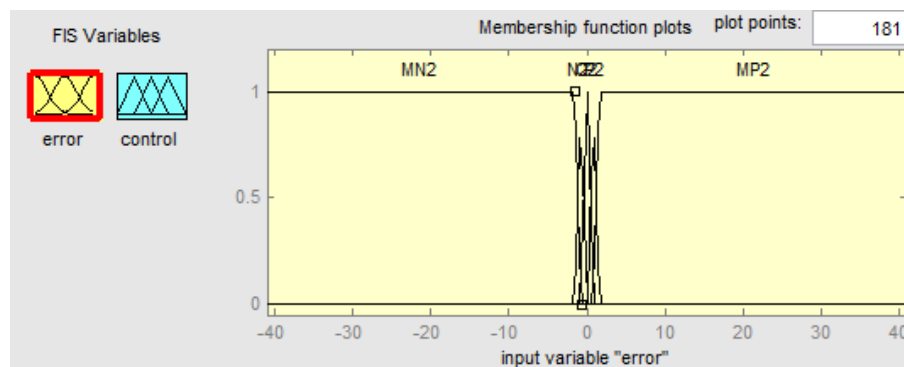


Figura 145. Diseño de entrada del controlador fuzzy del motor DC M2

La primera función de membresía considerada se la llamó MN2, la cual es de forma trapezoidal que comprende el rango de -41 hasta -0.75.

La segunda función de membresía considerada se la llamó N2, la cual es de forma triangular que comprende el rango de -1.5 hasta 0.

La tercera función de membresía considerada se la llamó C2, la cual es de forma triangular que comprende el rango de -0.75 hasta 0.75.

La cuarta función de membresía considerada se la llamó P2, la cual es de forma triangular que comprende el rango de 0 hasta 1.5.

La quinta función de membresía considerada se la llamó MP2, la cual es de forma trapezoidal que comprende el rango de 0.75 hasta 41.

3.8.2.2. Diseño de la salida del controlador fuzzy del motor DC M2

Para el diseño de la salida del controlador Fuzzy del motor DC M2, se estableció como rango el voltaje comprendido de -23 voltios hasta 23 voltios, dicho rango es el voltaje máximo que se implementó al motor DC M2, ya que el voltaje nominal es de -24 voltios hasta 24 voltios, dejando 1 voltio de diferencia por seguridad del motor DC M2.

En este controlador no hay sobrepicos que puedan pasar el voltaje nominal del motor DC M2, con lo cual se puede trabajar con el voltaje cercano al nominal sin inconvenientes.

Se implementó cinco funciones de membresía para la salida del controlador.

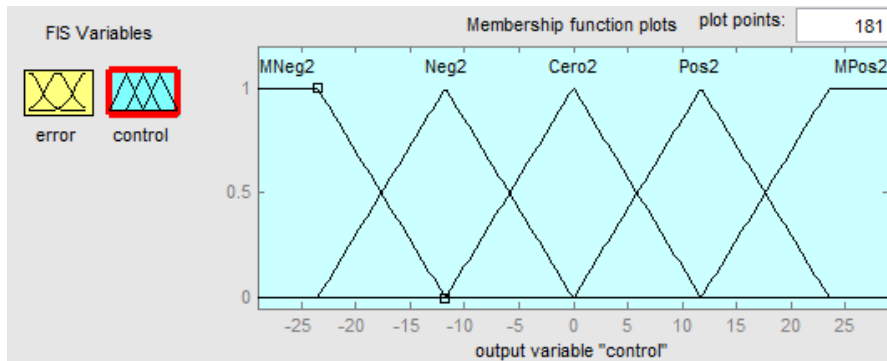


Figura 146. Diseño de salida del controlador fuzzy del motor DC M2

La primera función de membresía considerada se la llamó MNeg2, la cual es de forma trapezoidal que comprende el rango de -29 hasta -11.75.

La segunda función de membresía considerada se la llamó Neg2, la cual es de forma triangular que comprende el rango de -23.5 hasta 0.

La tercera función de membresía considerada se la llamó Cero2, la cual es de forma triangular que comprende el rango de -11.75 hasta 11.75.

La cuarta función de membresía considerada se la llamó Pos2, la cual es de forma triangular que comprende el rango de 0 hasta 23.5.

La quinta función de membresía considerada se la llamó MPos2, la cual es de forma trapezoidal que comprende el rango de 11.75 hasta 29.

Usando Simulink se procedió a simular el controlador Fuzzy del motor DC M2 agregando una entrada paso de 41 (cm) y la salida de control del controlador.

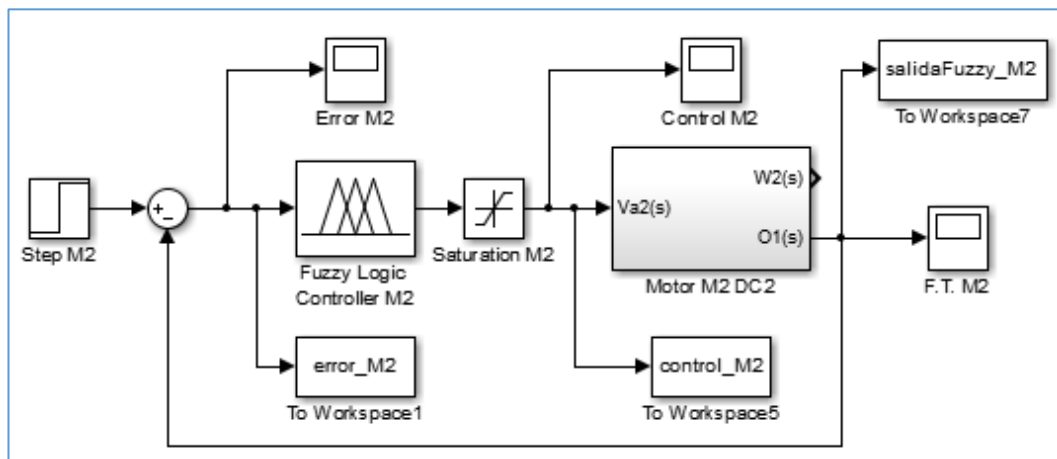


Figura 147. Diagrama de bloques del control fuzzy del motor DC M2

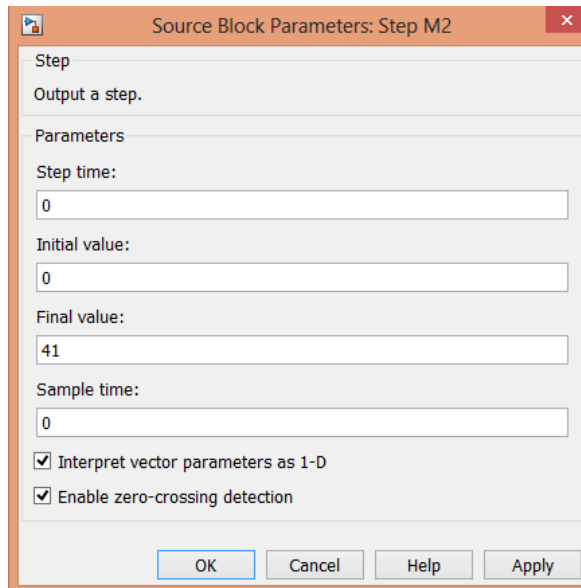


Figura 148. Configuración de entrada paso a 41cm

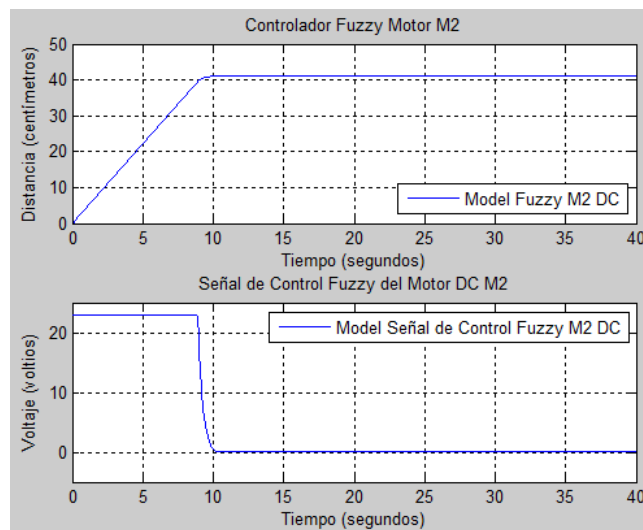


Figura 149. Respuesta de entrada paso y salida fuzzy del motor DC M2

3.9. Validación de controladores fuzzy de los motores DC M1 y M2

Para las pruebas de validación de los controladores Fuzzy de los motores DC M1 y M2, se utilizó el software TIA PORTAL donde se programó en lenguaje SCL. Se estableció una conexión ETHERNET con el PLC S7-300 mediante el módulo de comunicación CP 343-1 Lean.

También se utilizó el software LabView y el software KEPserverEX para la obtención de los datos reales de distancia y de la salida del controlador de los motores DC M1 y M2.

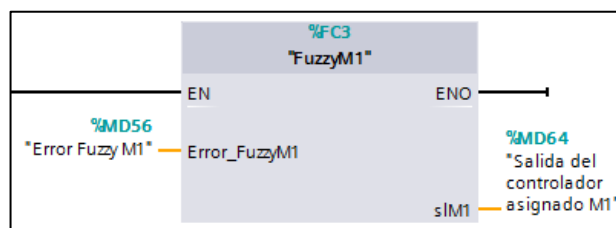


Figura 150. Bloque de función fuzzy

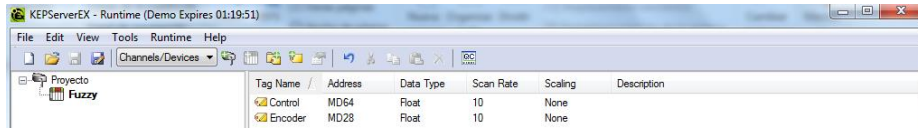


Figura 151. Variables de datos en KEPServerEX

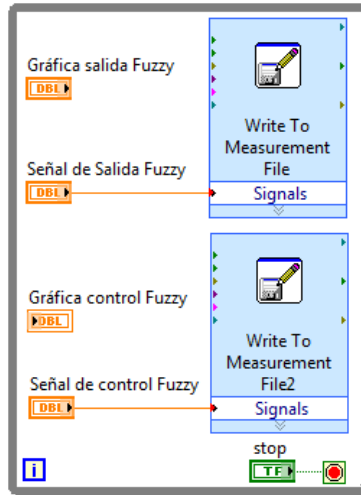


Figura 152. Datos reales de controladores de motores DC M1 y M2

3.9.1. Prueba de validación del controlador fuzzy del motor DC M1

Para la validación del controlador Fuzzy del motor DC M1 se programó el bloque de función Fuzzy M1 y el bloque de función Error Fuzzy M1 en lenguaje SCL en TIA PORTAL.

Se añadió un Set Point de 60 (cm).

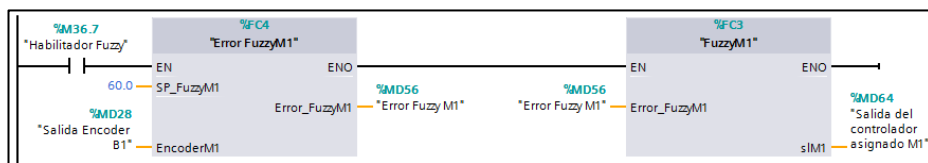


Figura 153. Configuración del error y fuzzy del motor DC M1

Mediante el programa MATLAB se procesaron los datos obtenidos del motor DC M1 con el controlador Fuzzy y de la salida del controlador obteniendo las gráficas correspondientes.

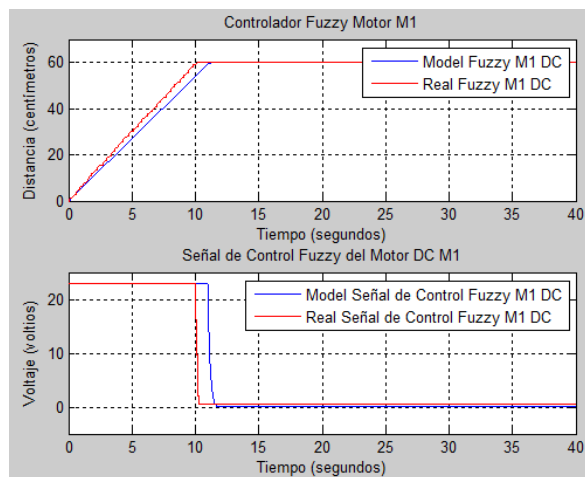


Figura 154. Comparación del controlador fuzzy del motor DC M1

3.9.2. Prueba de validación del controlador fuzzy del motor DC M2

Para la validación del controlador Fuzzy del motor DC M2 se programó el bloque de función Fuzzy M1 y el bloque de función Error Fuzzy M1 en lenguaje SCL en TIA PORTAL. Se añadió un Set Point de 41 (cm).

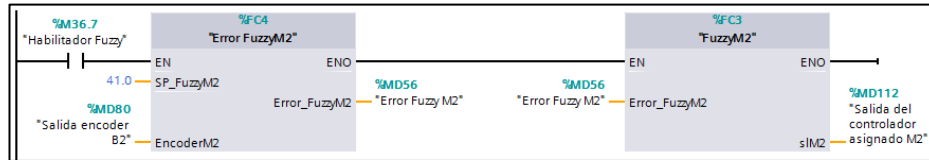


Figura 155. Configuración del error y fuzzy del motor DC M2

Mediante el programa MATLAB se procesaron los datos obtenidos del motor DC M2 con el controlador Fuzzy y de la salida del controlador obteniendo las gráficas correspondientes.

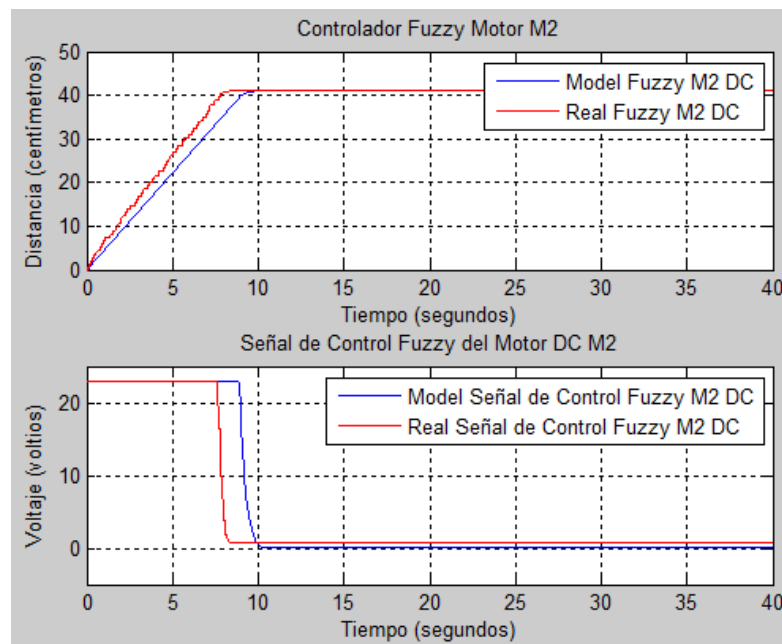


Figura 156. Comparación del controlador fuzzy del motor DC M2

3.10. Visualización de datos en tiempo real utilizando LabView

En esta parte del capítulo se procedió a diseñar la visualización en tiempo real del proceso de la planta, para que el usuario pueda ir verificando el funcionamiento respectivo de cada motor DC M1 y M2.

El proceso comienza pulsando la botonera Start con el cual se enciende el indicador de proceso iniciado, el cual indica que el proceso de almacenamiento está activo.

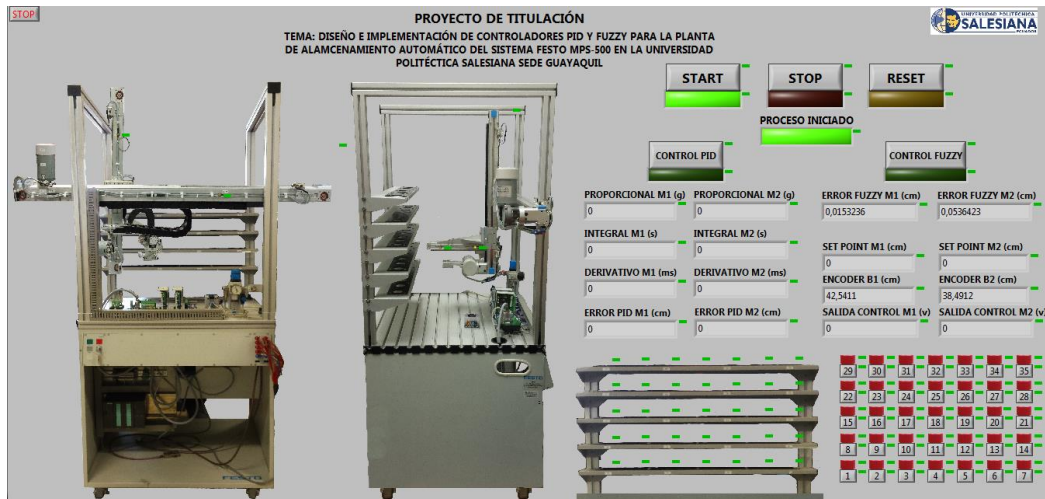


Figura 157. Inicio del proceso del sistema Scada en LabView

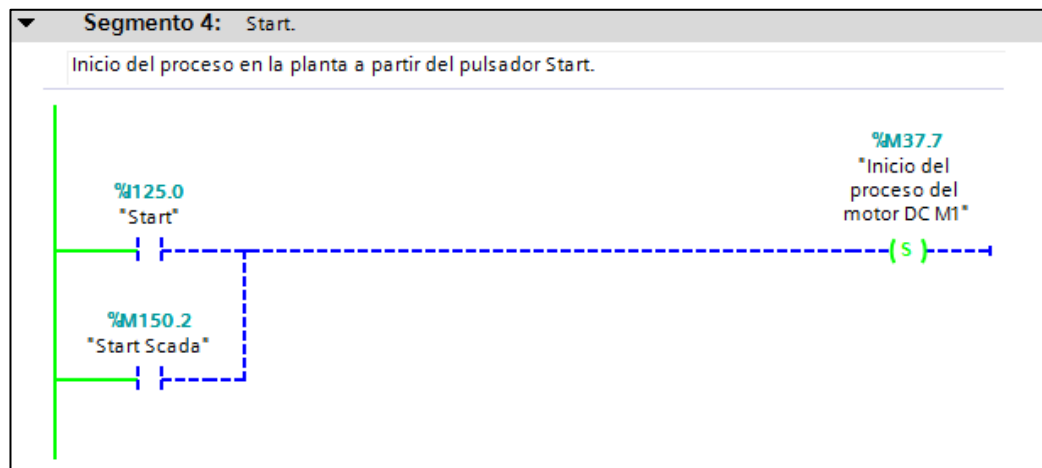


Figura 158. Pulsación de la botonera Start

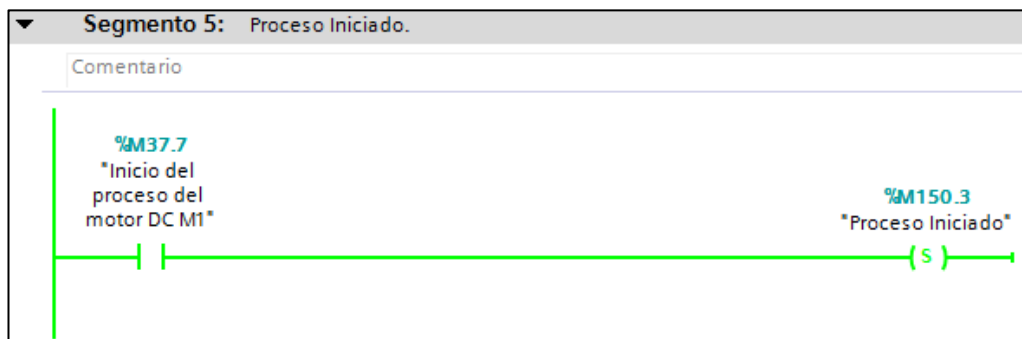


Figura 159. Proceso iniciado de la planta en tia portal

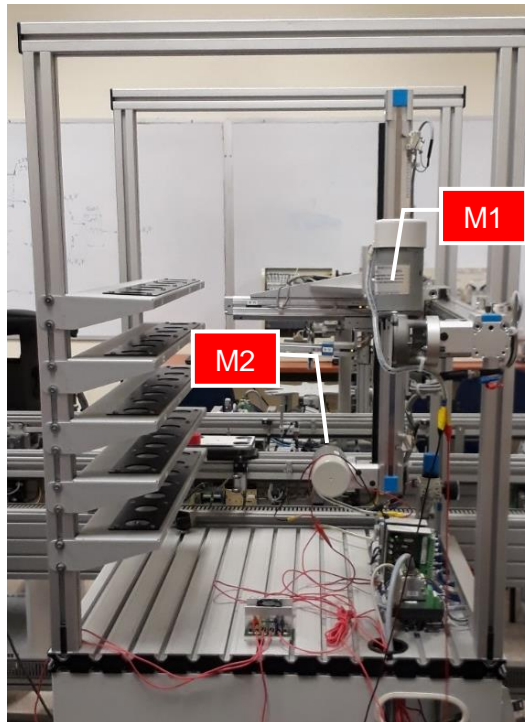


Figura 160. Proceso de los motores M1 y M2 a su posición inicial

Con haber iniciado el proceso con el botón Start, los motores hacen desplazar los brazos hacia la posición inicial para con ello ir a recolectar la pieza de trabajo.

Primero arranca el motor DC M1 desplazando el brazo a la posición inicial que se encuentra en el sensor B10, una vez detenido el motor M1 arranca el motor M2 haciendo desplazar su brazo hacia su posición inicial.

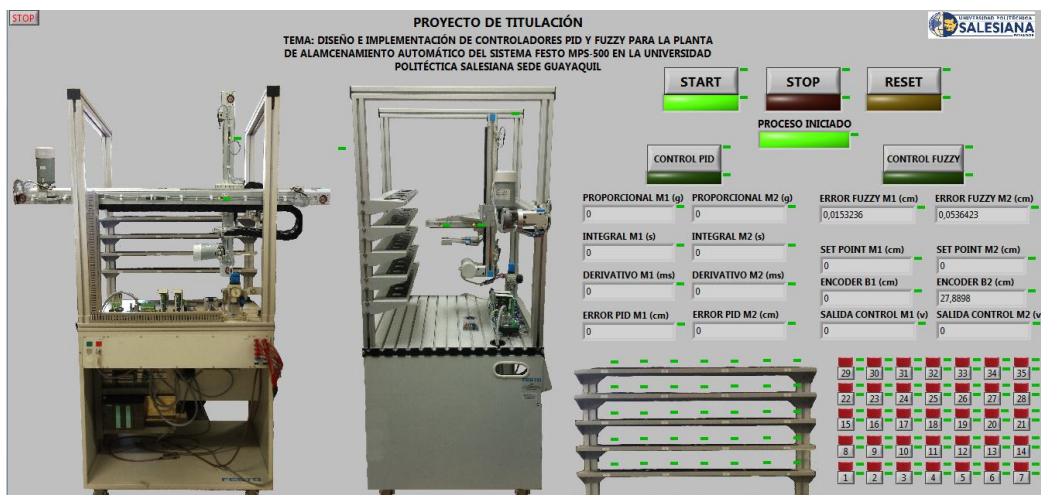


Figura 161. Motor M1 en la posición inicial en el sistema Scada

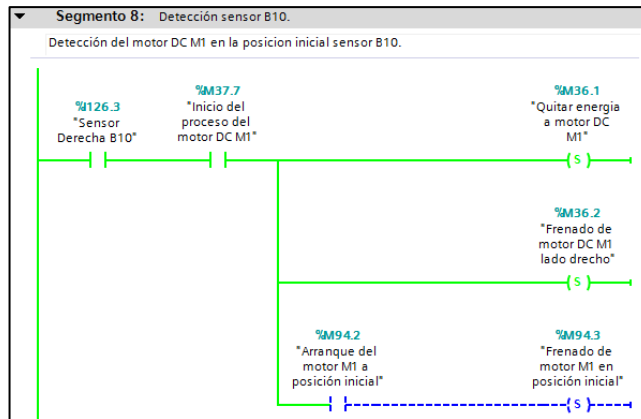


Figura 162. Motor M1 en la posición inicial en tía portal

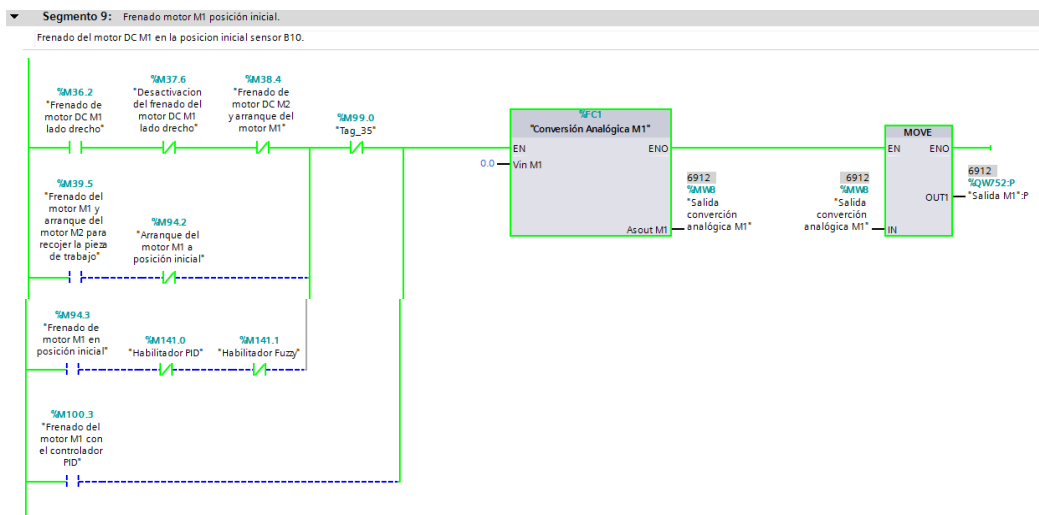


Figura 163. Motor M1 detenido en la posición inicial en tía portal

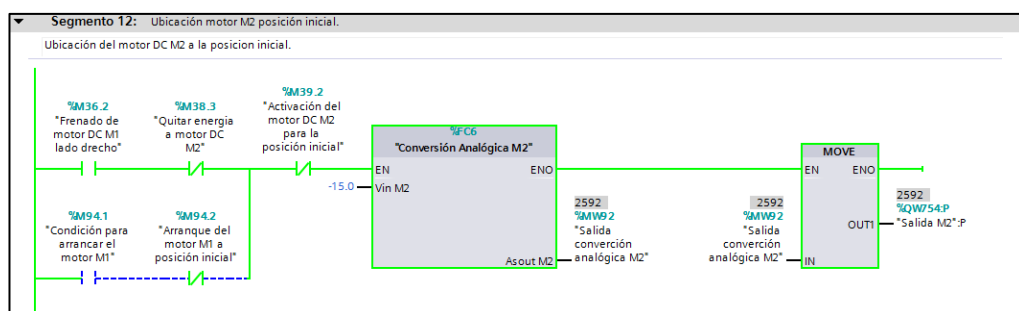


Figura 164. Arranque del motor M2 en tía portal

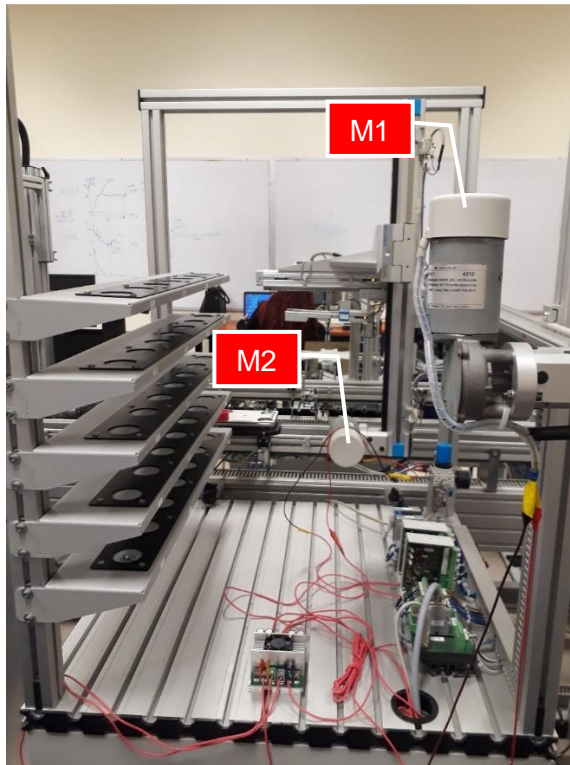


Figura 165. Posición inicial del motor M1

Con el motor M1 en la posición inicial, arranca el motor M2 haciendo desplazar su brazo a la posición inicial y este se detiene en su posición inicial a través del sensor B12, una vez detenido el motor M2 arranca nuevamente el motor M1 haciendo desplazar su brazo hacia la posición de la pieza de trabajo.

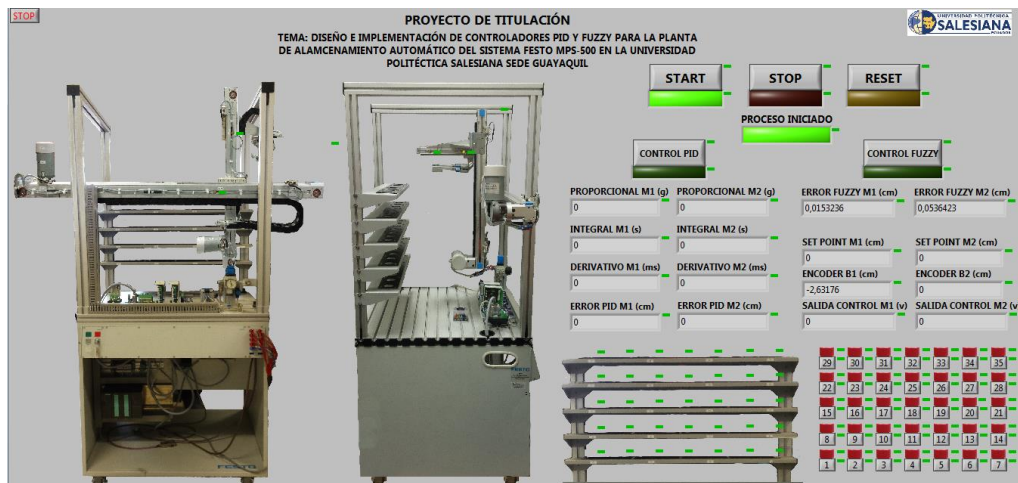


Figura 166. Motor M2 en la posición inicial en el sistema Scada

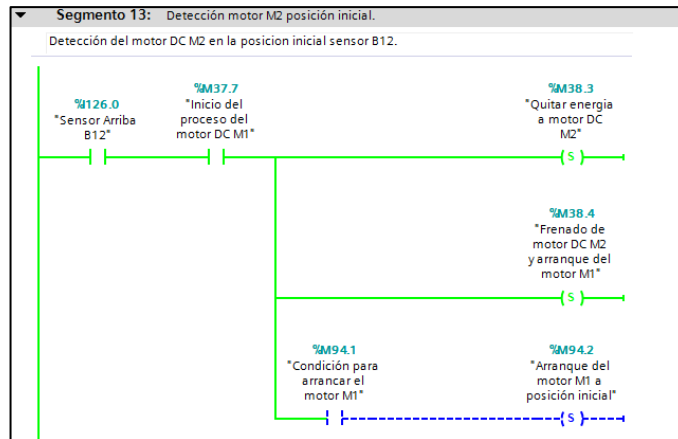


Figura 167. Motor M2 en la posición inicial en tía portal

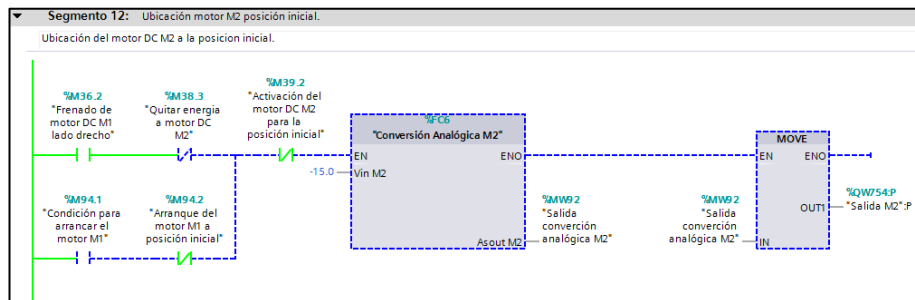


Figura 168. Frenado del motor M2 en la posición inicial en tía portal

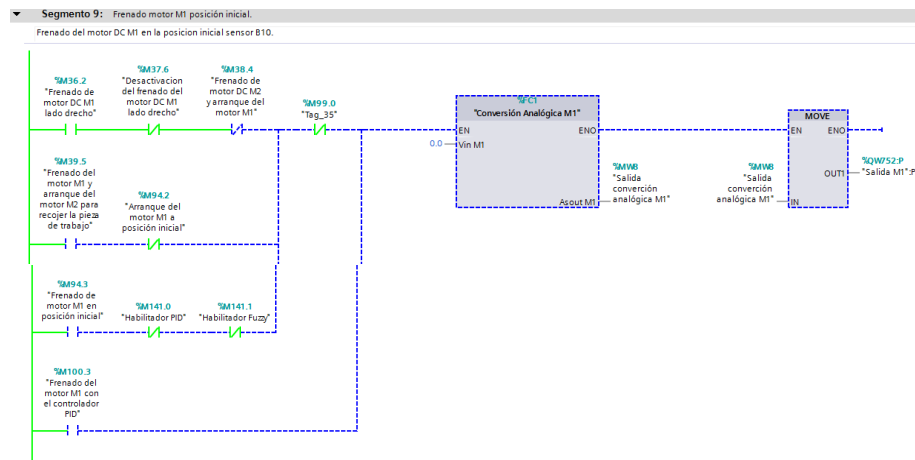


Figura 169. Desactivación del frenado del motor M1 en tía portal

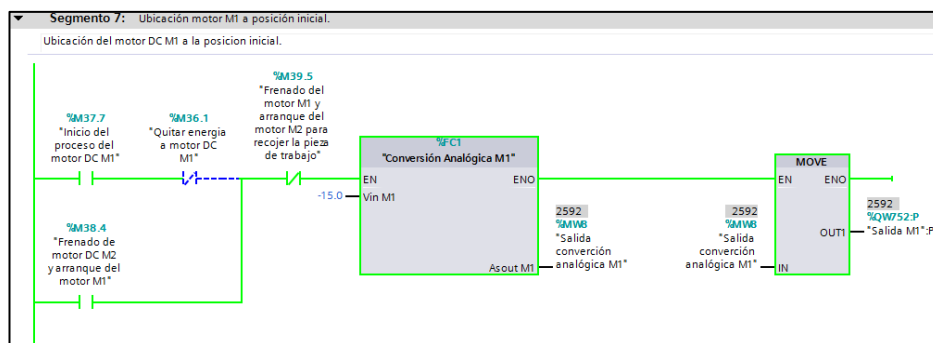


Figura 170. Arranque del motor M1 en tía portal

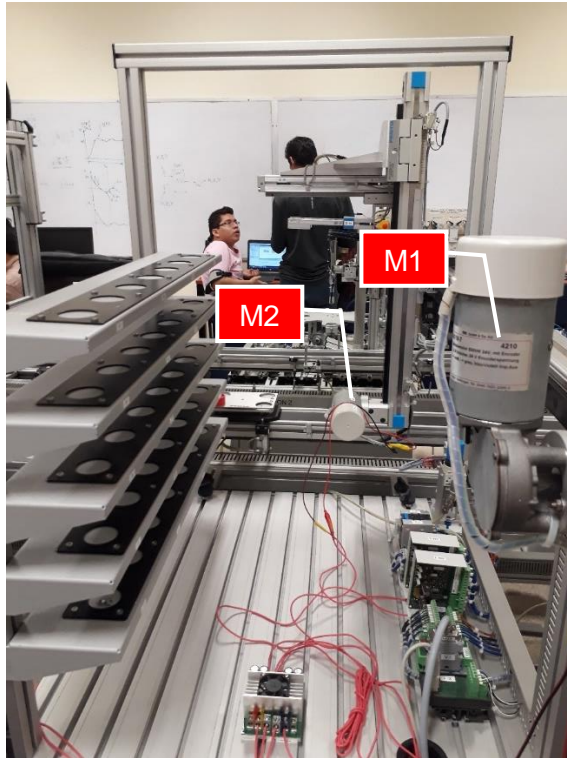


Figura 171. Posición inicial motor M2 visualizado en la planta

El motor M1 se detiene en la posición para recolectar la pieza con la condición de mayor o igual a -13.65 que es la distancia desde la posición inicial que debe recorrer el motor M1, una vez que el motor M1 se detiene en dicha posición, arranca el motor M2 haciendo desplazar su brazo hacia la pieza de trabajo.

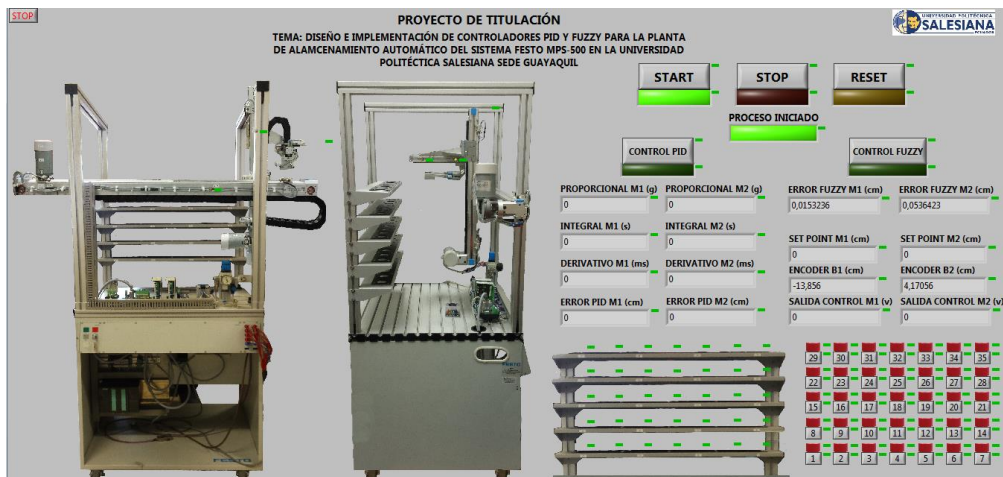


Figura 172. Motor M1 en posición de pieza de trabajo del sistema Scada

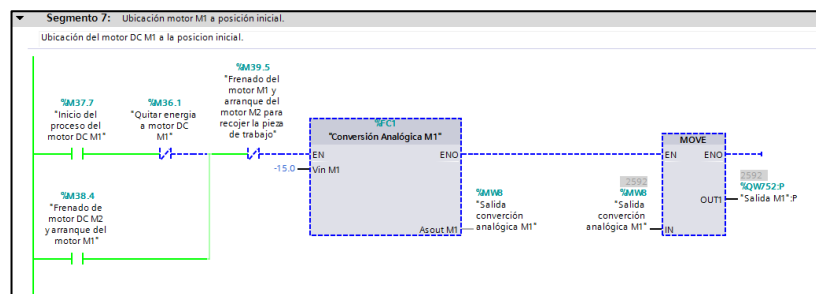


Figura 173. Frenado del motor M1 en tía portal

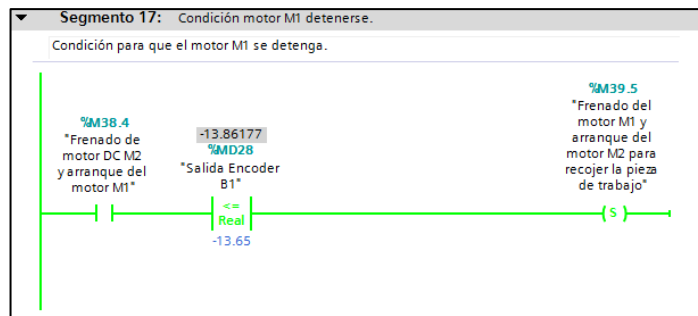


Figura 174. Condición de frenado del motor M1 en tía portal

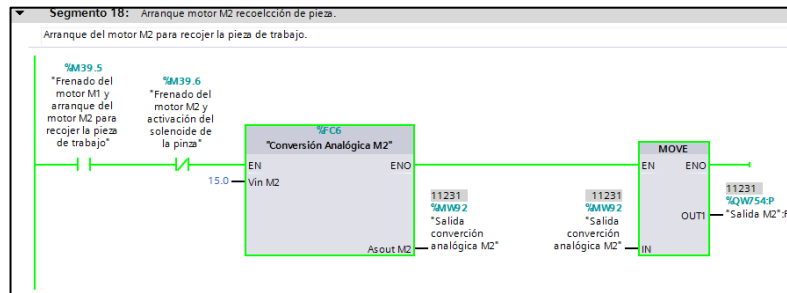


Figura 175. Arranque del motor M2 en tía portal

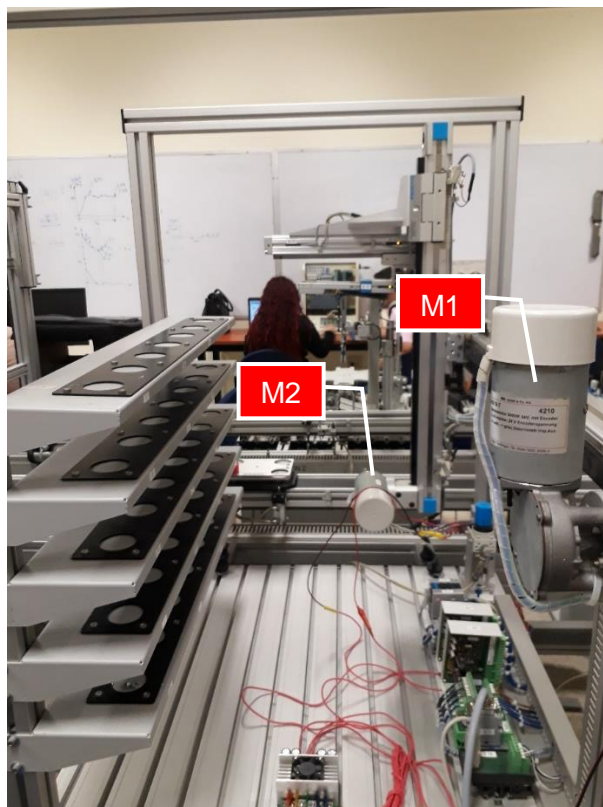


Figura 176. Posición inicial del motor M2 visualizado en la planta

El motor M2 se detiene en la posición para recolectar la pieza con la condición de mayor o igual a 40.6 que es la distancia desde la posición inicial que debe recorrer el motor M2, una vez que el motor M2 se detiene en dicha posición, se activa el solenoide de la pinza y el solenoide del eje Y, para que con ello se recoja la pieza de trabajo, una vez recogida la pieza de trabajo, arranca el motor M2 hacia su posición inicial.

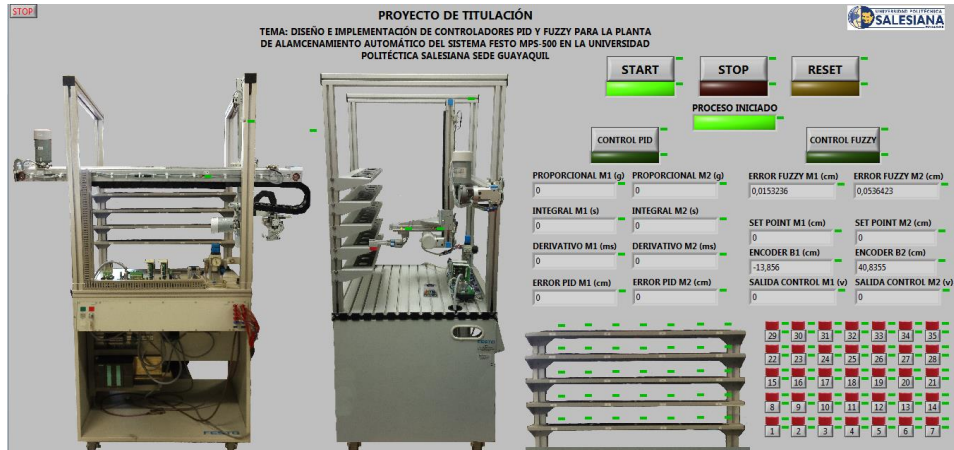


Figura 177. Recolección del Motor M2 de la pieza en el sistema Scada

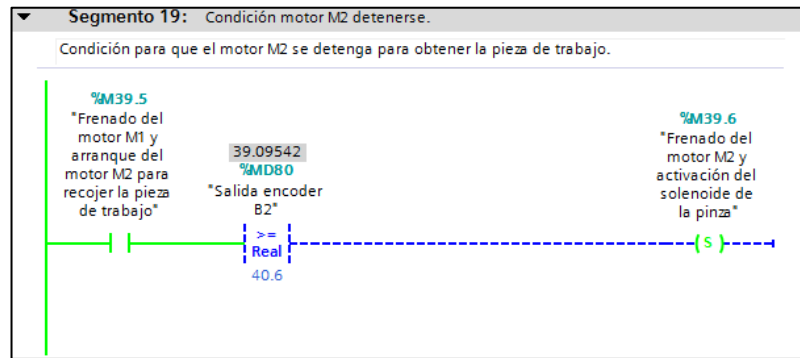


Figura 178. Condición de frenado del motor M2 en tia portal

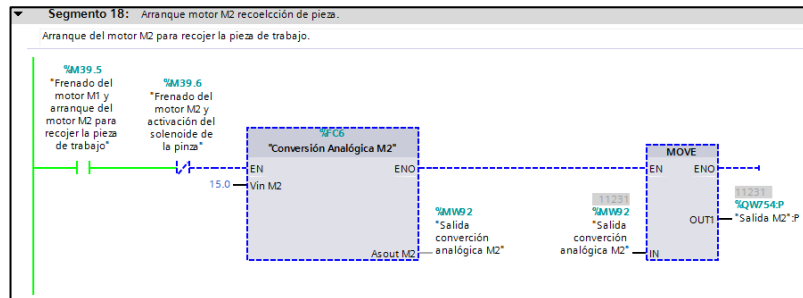


Figura 179. Frenado del motor M2 en programa tia portal

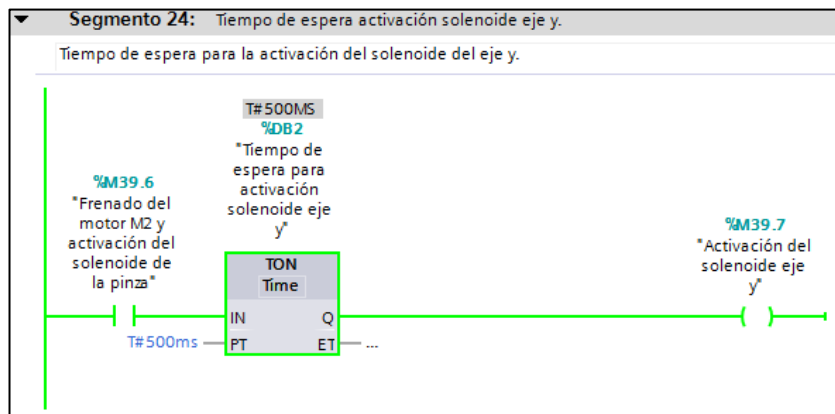


Figura 180. Tiempo de espera del solenoide del eje Y en tia portal

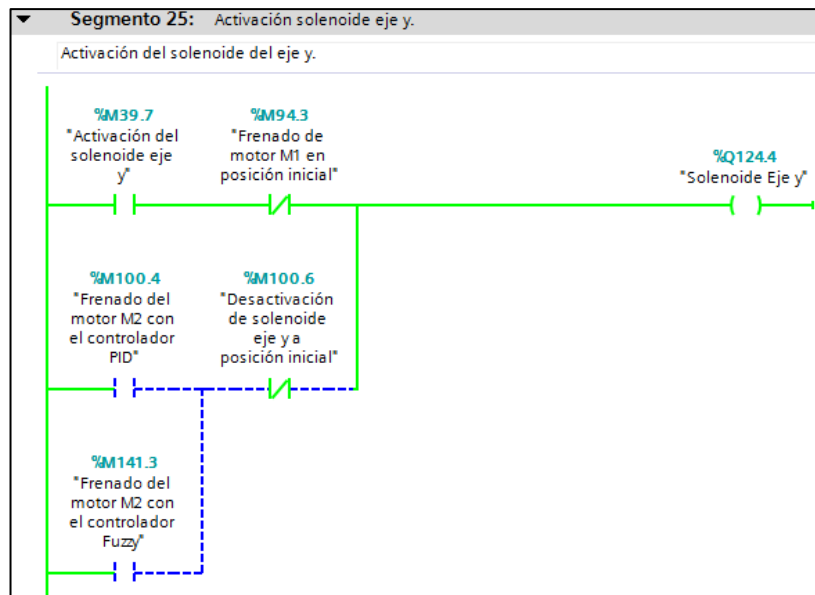


Figura 181. Activación del solenoide del eje Y en tia portal

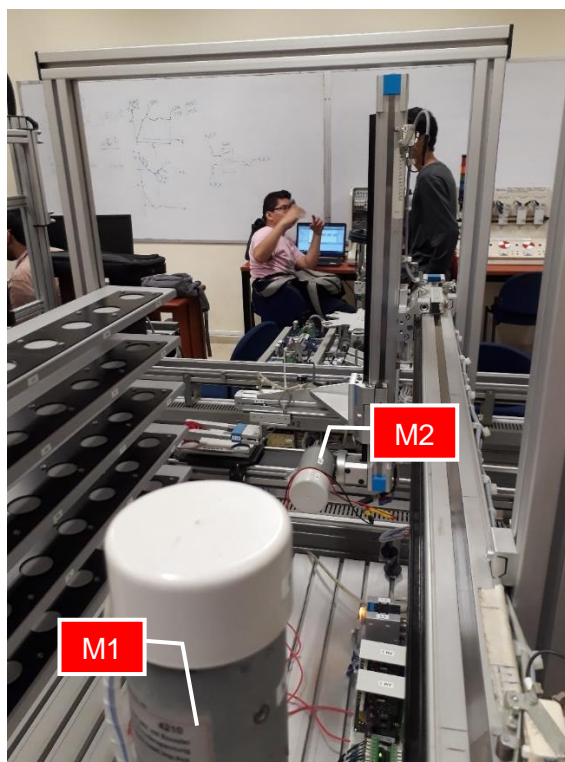


Figura 182. Recolección de la pieza de trabajo y solenoide activado

El motor M2 se detiene en su posición inicial e inmediatamente arranca el motor M1 hacia su posición inicial.

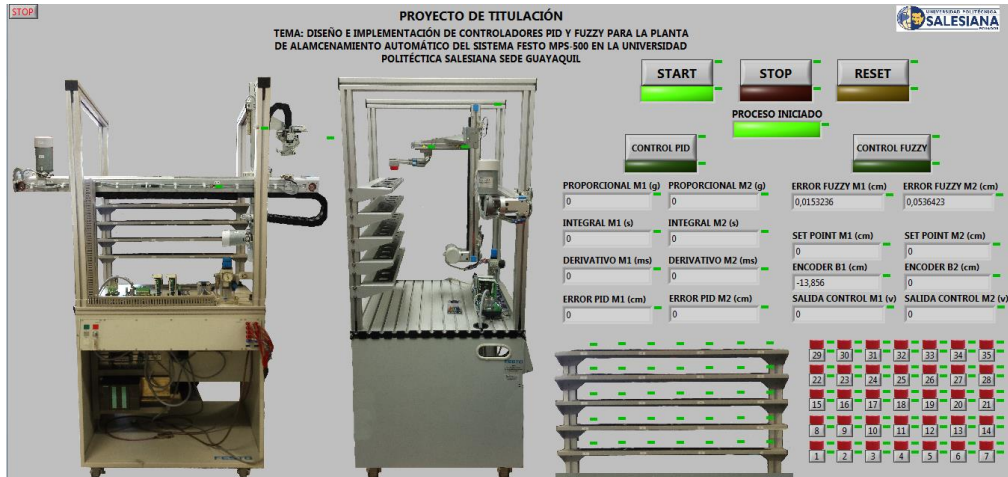


Figura 183. Motor M2 con el solenoide activado en el sistema Scada

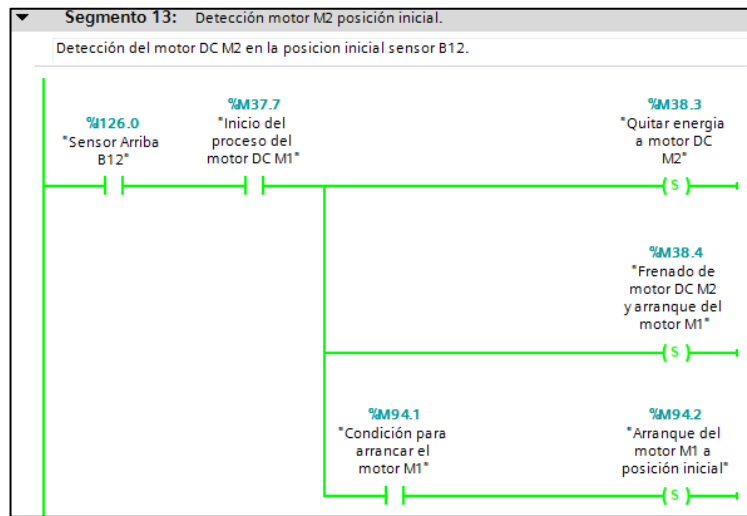


Figura 184. Motor M2 en la posición inicial en tía portal

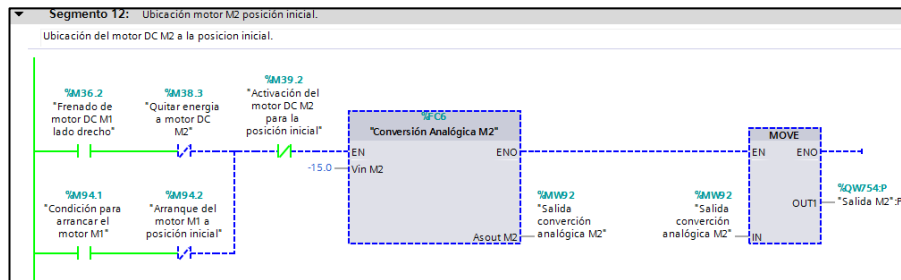


Figura 185. Frenado del motor M2 con la pieza de trabajo en tía portal

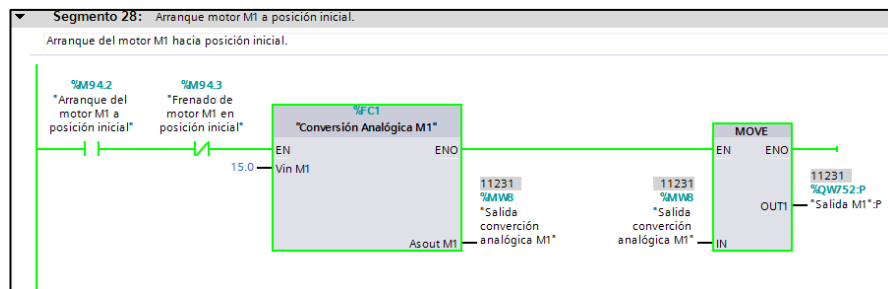


Figura 186. Arranque del motor M1 a la posición inicial en tía portal

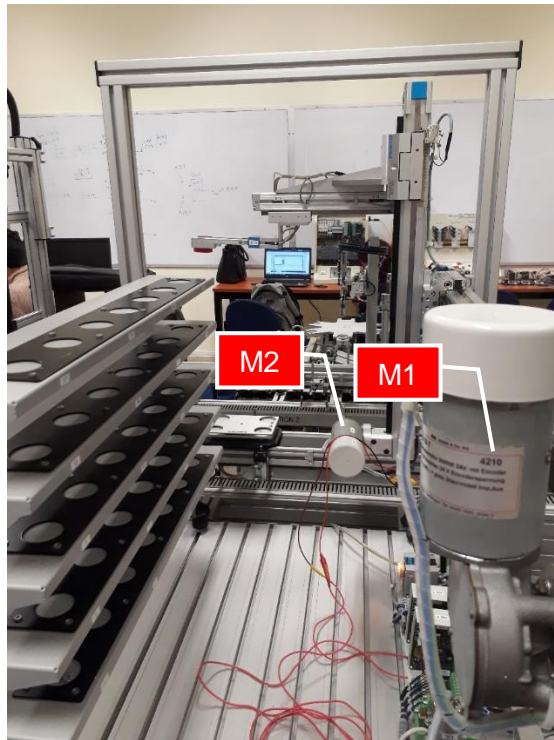


Figura 187. Motor M2 con solenoide activado visualizado en la planta.

Al llegar el motor M2 a su posición inicial con la pieza de trabajo, se desactiva el solenoide del eje Y, a partir de aquí el proceso está listo para elegir con que controlador trabajar para el posicionamiento y almacenamiento de la pieza de trabajo.

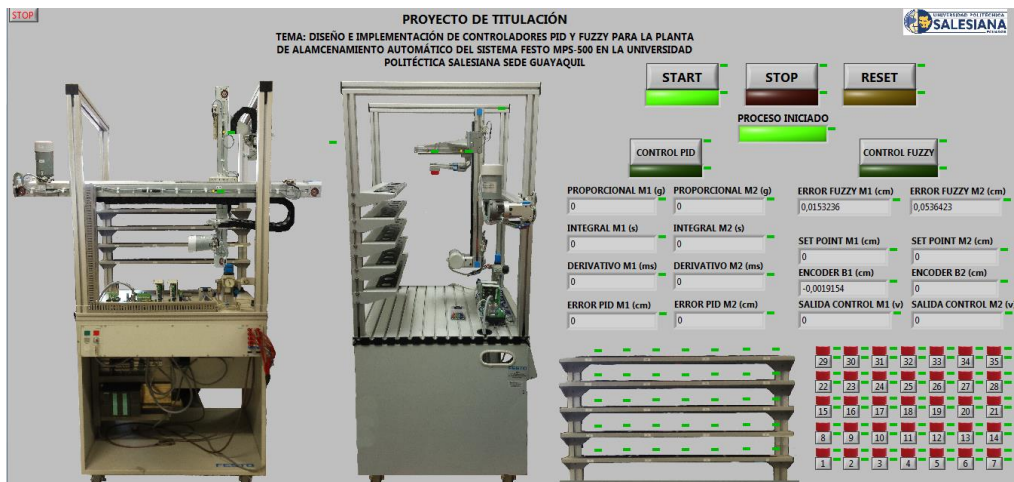


Figura 188. Motor M2 con solenoide desactivado en el sistema Scada

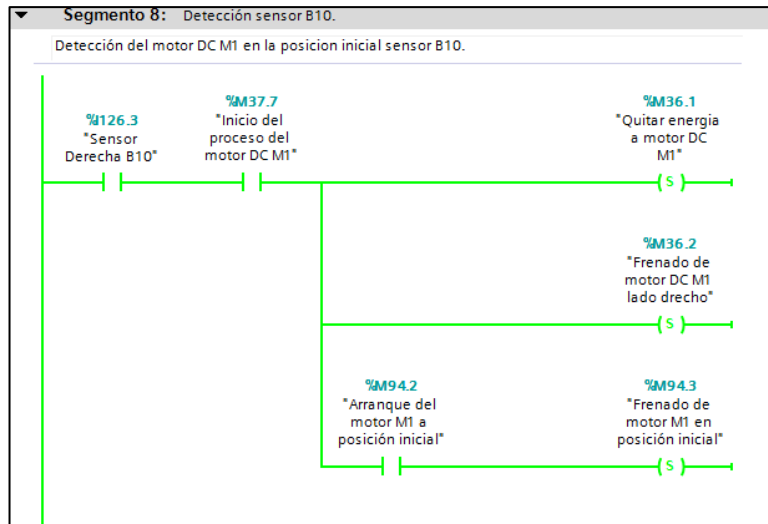


Figura 189. Detección del motor M1 en tía portal

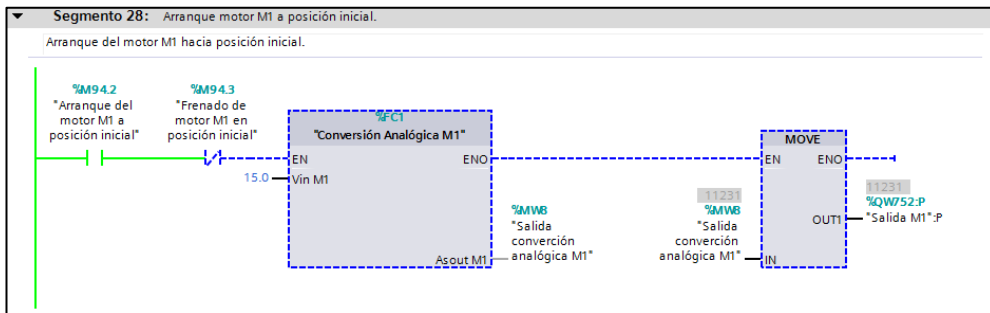


Figura 190. Desactivación del motor M1 con la pieza de trabajo en tía portal

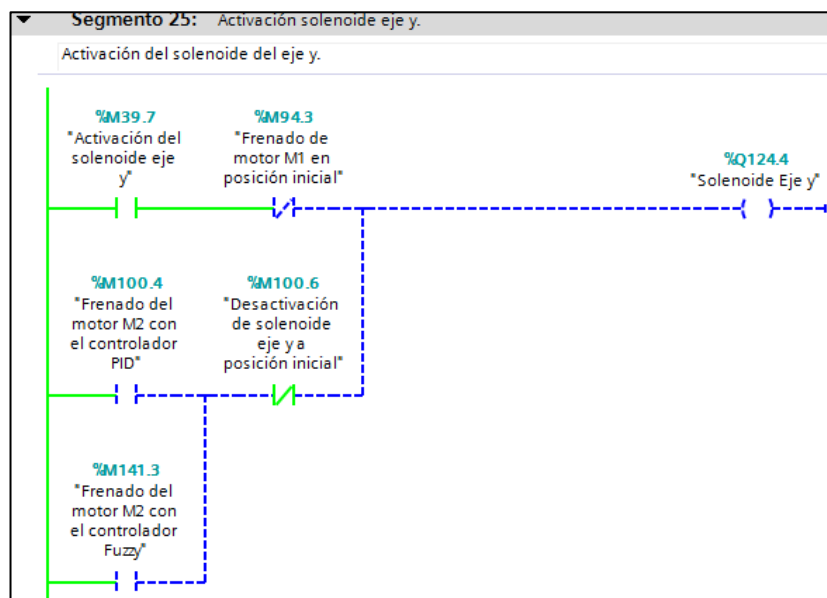


Figura 191. Desactivación del solenoide eje Y en tía portal

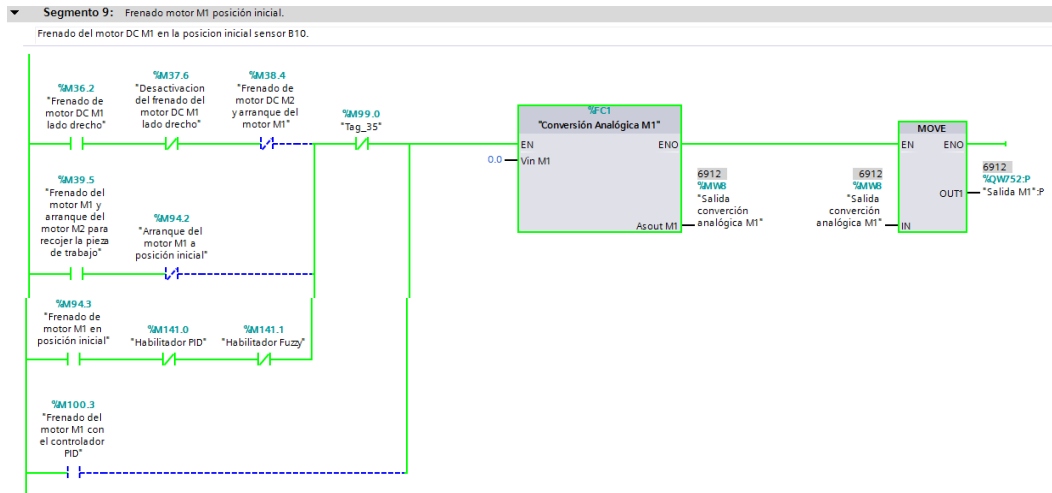


Figura 192. Frenado del motor M1 en la posición inicial en tia portal

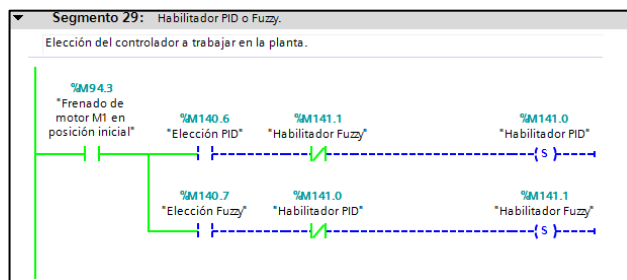


Figura 193. Habilitador PID o Fuzzy en tia portal

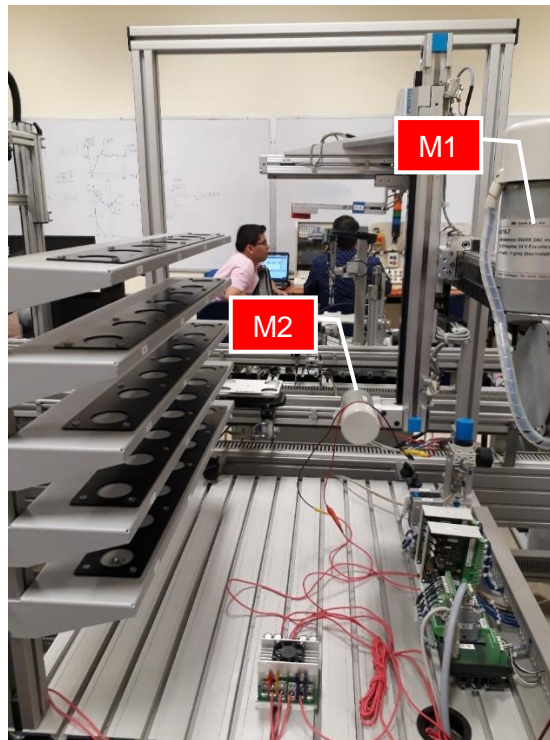


Figura 194. Motor M2 pieza de trabajo activa y solenoide desactivado

En el sistema Scada en el programa LabView pulsamos la botonera Control Fuzzy, con lo cual se activa la opción para el posicionamiento de la pieza de trabajo.



Figura 195. Pulsado de la botonera Control Fuzzy en el sistema Scada

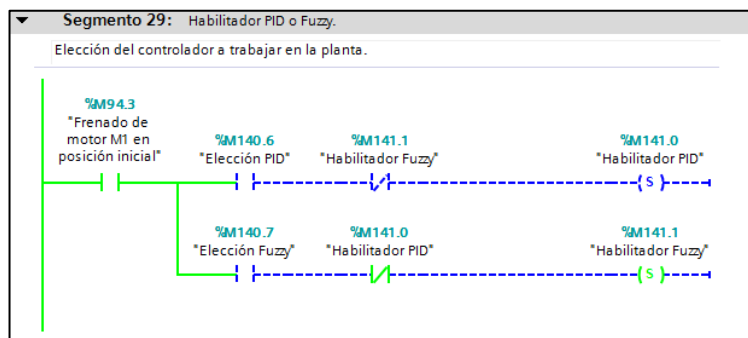


Figura 196. Activación de la elección del controlador en tía portal

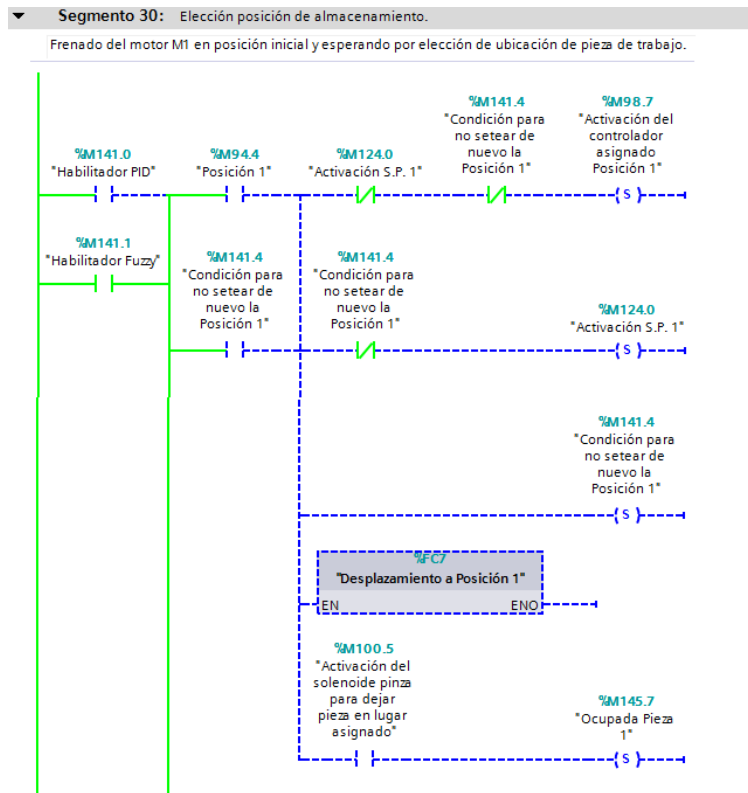


Figura 197. Activación para el posicionamiento de la pieza de trabajo en tía portal

Una vez elegido el controlador con el cual se va a trabajar, elegimos la posición en la cual se almacenará la pieza de trabajo, esta posición puede ser desde la 1 hasta la 35, se eligió la posición 1.



Figura 198. Posición 1 para el almacenamiento en el sistema Scada

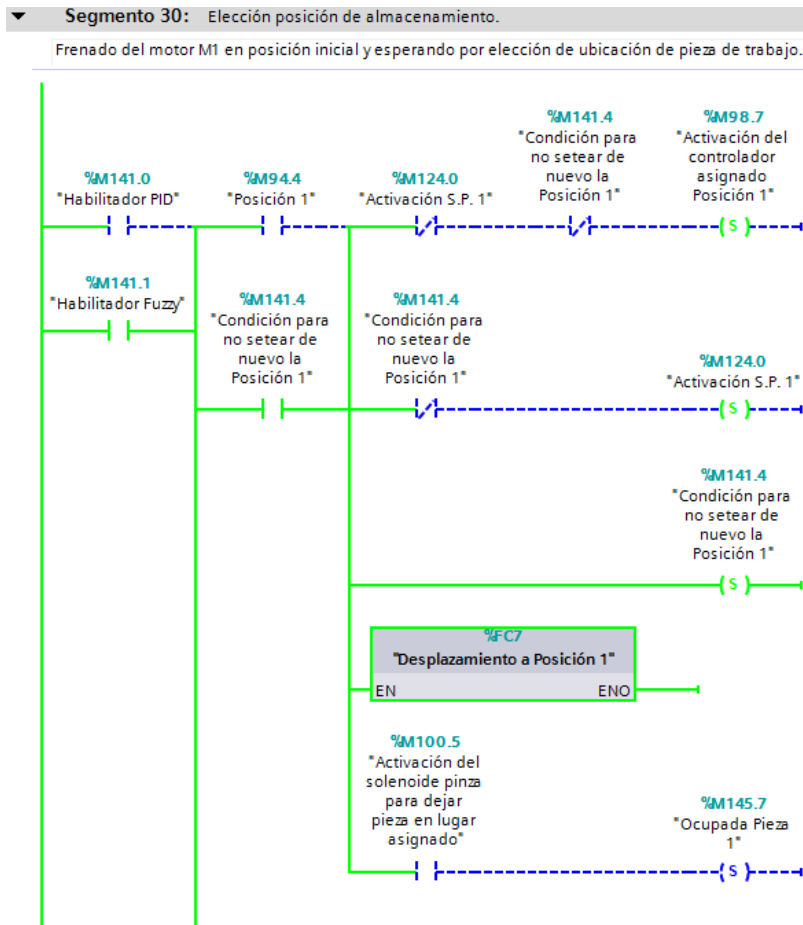


Figura 199. Posición 1 para el almacenamiento de la pieza en tía portal

Una vez elegida la posición, se activa el control elegido Fuzzy para el motor M1.



Figura 200. Arranque del motor M1 con el control Fuzzy en el sistema Scada

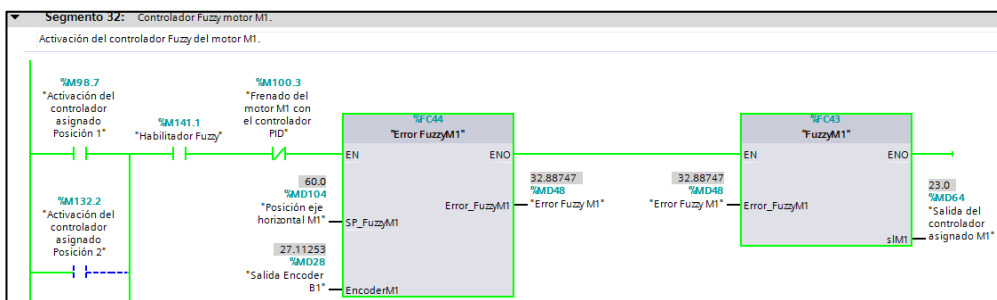


Figura 201. Activación del control Fuzzy del motor M1 en tia portal

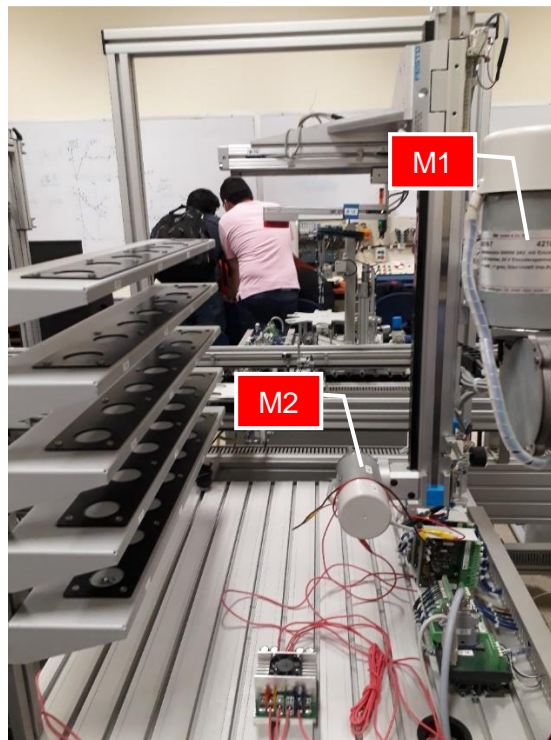


Figura 202. Motor M1 hacia la posición 1 visualizado en la planta.

Una vez posesionado el motor M1 en el set point de la posición 1, se activa el control fuzzy del motor M2 para el posicionamiento y almacenamiento de la pieza de trabajo.



Figura 203. Arranque del motor M2 hacia la posición 1 Sistema Scada

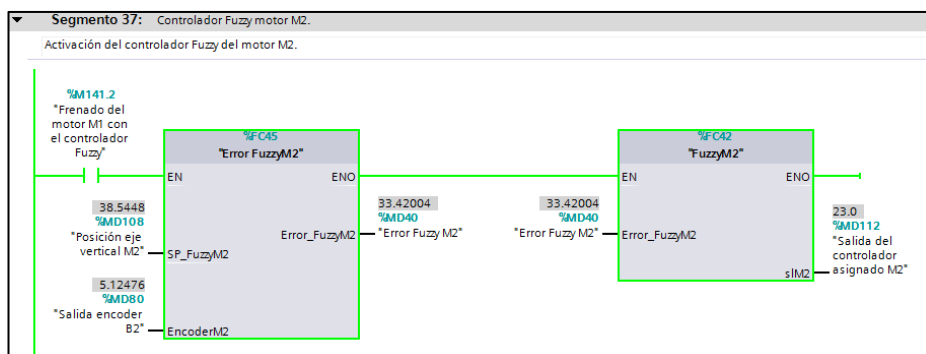


Figura 204. Activación del control Fuzzy del motor M2 en tia portal

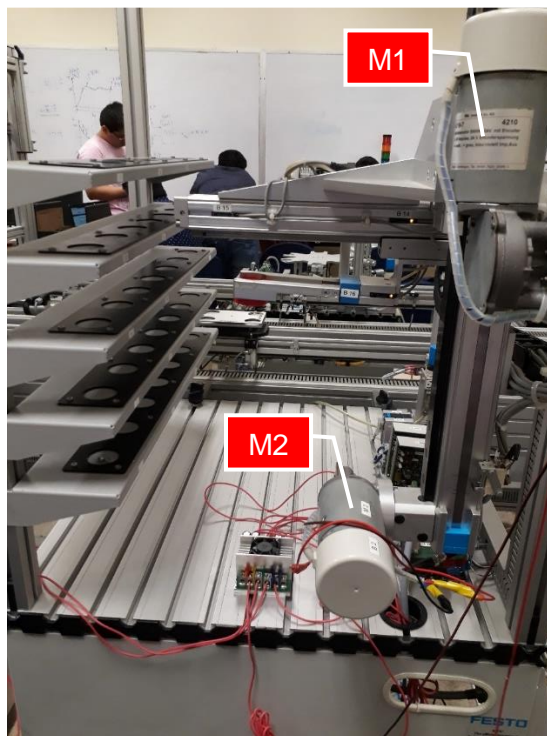


Figura 205. Arranque inmediato del motor M2 para alcanzar el Set Point

Una vez posesionado el motor M2 en el Set Point de la posición 1, se activa el solenoide del eje Y y de la pinza, quedando almacenada la pieza de trabajo en la posición deseada.

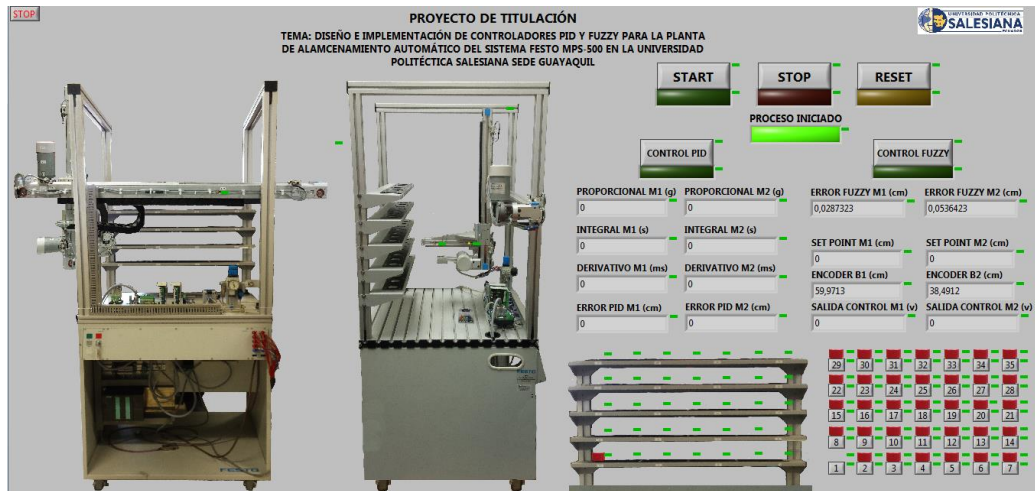


Figura 206. Almacenamiento de la pieza de trabajo en la posición 1 sistema Scada

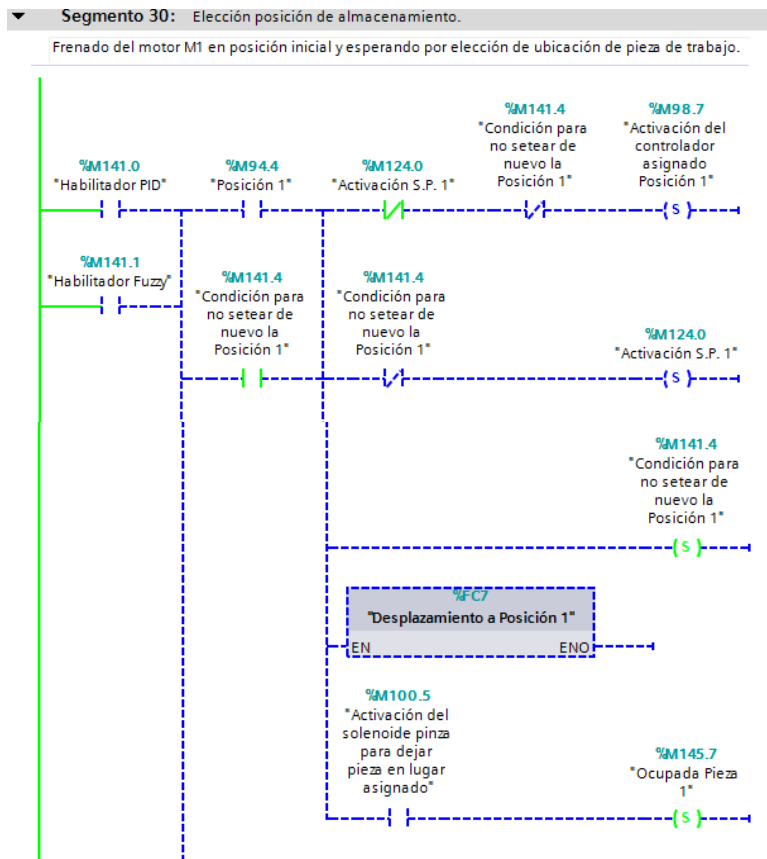


Figura 207. Almacenamiento de la pieza de trabajo en la posición 1 en tia portal

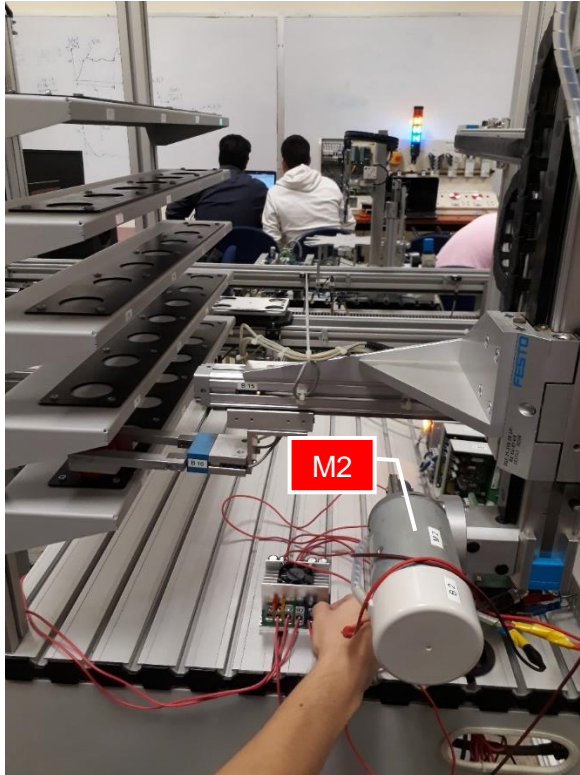


Figura 208. Ubicación de la pieza de trabajo en la posición 1.

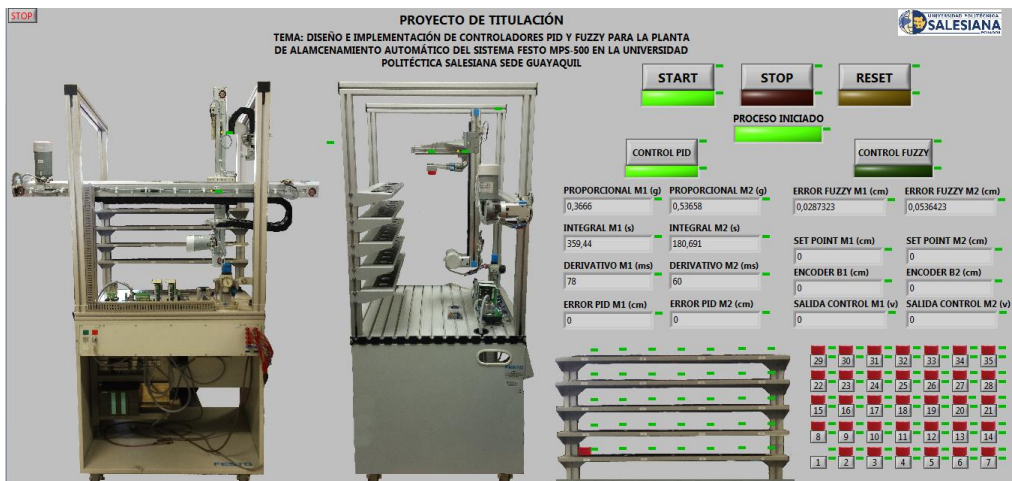


Figura 209. Pulsado de la botonera Control PID en el sistema Scada

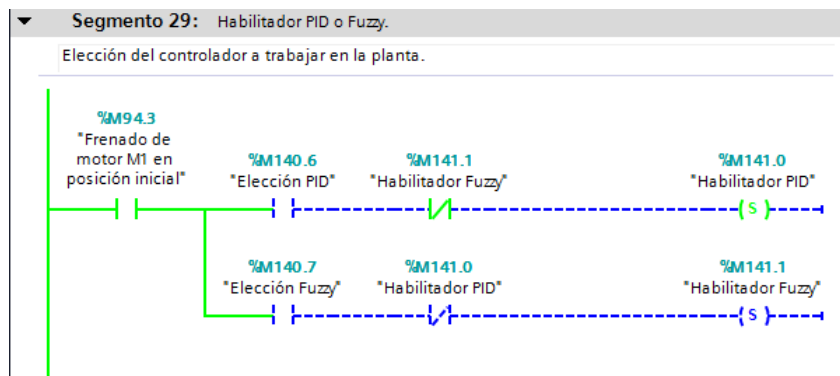


Figura 210. Activación y elección del controlador PID en tia portal

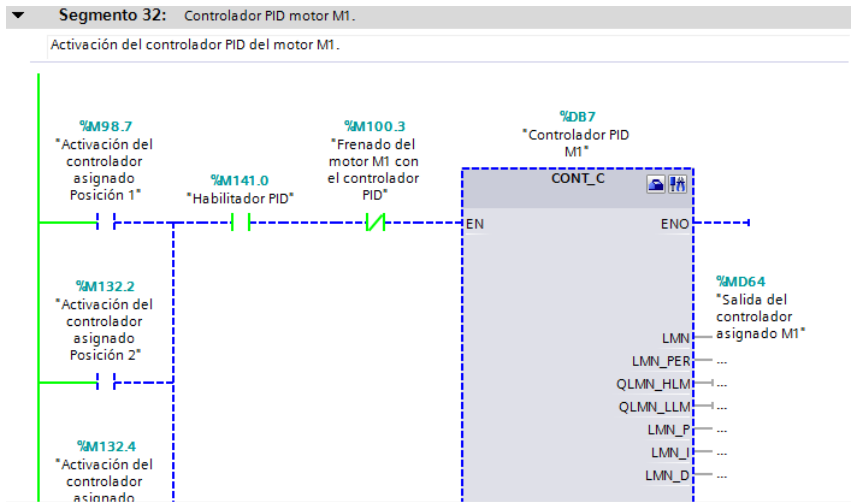


Figura 211. Activación del controlador PID del Motor M1 en tía portal

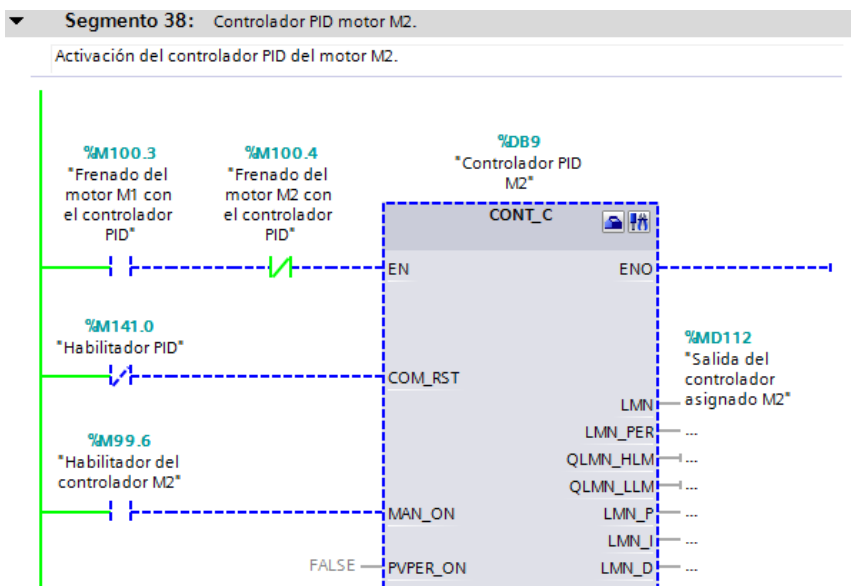





Figura 212. Activación del controlador PID del Motor M2 en tía portal

4. GUÍA DE PRÁCTICAS DE LABORATORIO

4.1. Práctica Nro. 1

	GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA: Ingeniería Electrónica	ASIGNATURA:
NRO. PRÁCTICA:	1 TÍTULO PRÁCTICA: Reconocimiento de elementos en la planta de almacenamiento automático del sistema Festo MPS-500.
Objetivos: <ul style="list-style-type: none"> • Reconocer cada elemento a emplear en el proceso de la planta. • Establecer las direcciones de entrada y salida de los elementos a utilizar en la planta en el programa TIA PORTAL. • Realizar un programa en TIA PORTAL para comprobar el funcionamiento de cada elemento. 	
INSTRUCCIONES:	<ol style="list-style-type: none"> 1. Verificación de la ubicación del PLC, sensores y actuadores de la planta. 2. Energización de la planta por medio de la fuente de voltaje de 24 voltios existente. 3. Creación de un nuevo programa en TIA PORTAL para la realización del programa de funcionamiento de los elementos de la planta.
<p style="text-align: center;">ACTIVIDADES POR DESARROLLAR</p> <p>Para la verificación del PLC, sensores y actuadores se procede a ver su ubicación en la planta:</p> <ul style="list-style-type: none"> • Autómata programable PLC S7-314C-2 DP. <div data-bbox="619 1294 1166 1641" data-label="Image">  </div> <p style="text-align: center;">Figura 213. PLC S7-314C-2 DP</p> <ul style="list-style-type: none"> • Sensores en el eje X <div data-bbox="470 1742 1315 1980" data-label="Image">  </div> <p style="text-align: center;">Figura 214. Sensores B10 y B11</p>	

- Sensores en el eje Y



Figura 215. Sensores B14 y B15

- Sensores en el eje Z

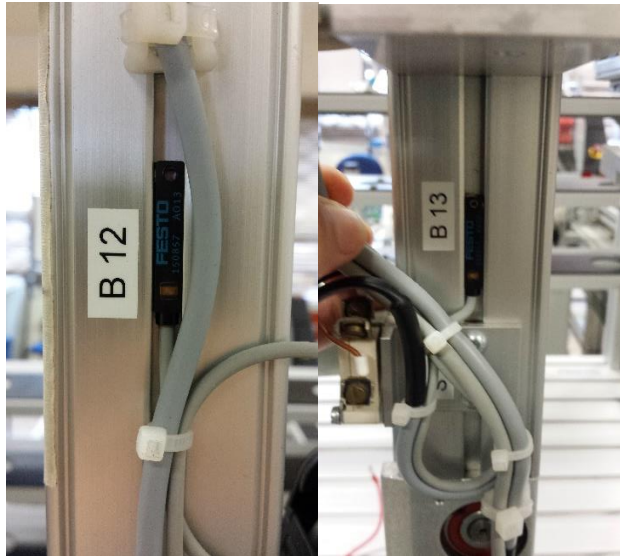


Figura 216. Sensores B12 y B13.

- Sensores del gripper.

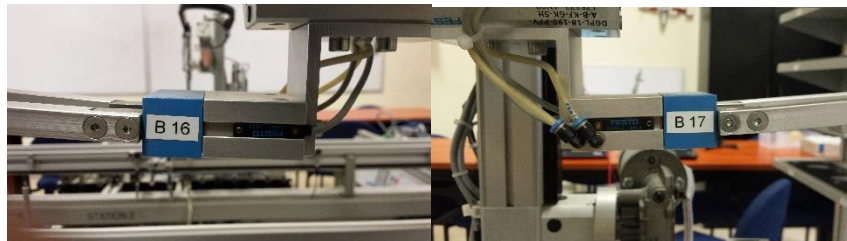


Figura 217. Sensores B16 y B17.

- Motores y encoders.



Figura 218. Motor M1 y encoder B1.



Figura 219. Motor M2 y encoder B2.

Después de la verificación de los elementos de la planta, se procede a crear un nuevo proyecto en el programa TIA PORTAL donde primero se ingresan las variables de entrada y salida de la planta.

Variables PLC								
	Nombre	Tabla de variables	Tipo de datos	Dirección	Rema...	Visibl...	Acces...	Coment
1	Inicio	Tabla de variables e..	Bool	%I125.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	Paro	Tabla de variables e..	Bool	%I125.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	Sensor Izquierda B11	Tabla de variables e..	Bool	%I126.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	Sensor Derecha B10	Tabla de variables e..	Bool	%I126.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	Sensor Anterior B14	Tabla de variables e..	Bool	%I126.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	Sensor Delantera B15	Tabla de variables e..	Bool	%I126.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	Pinza Cerrada B16	Tabla de variables e..	Bool	%I126.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	Pinza Abierta B17	Tabla de variables e..	Bool	%I126.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	Solenoido Eje y	Tabla de variables e..	Bool	%Q124.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	Solenoido Pinza	Tabla de variables e..	Bool	%Q124.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11	Entrada M1	Tabla de variables e..	Int	%QW752		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12	Entrada M2	Tabla de variables e..	Word	%QW754		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
13	salida multiplicada del voltaje M1	Tabla de variables e..	Real	%MD0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
14	Valor error escalado M1	Tabla de variables e..	Word	%MW4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
15	Salida escalada M1	Tabla de variables e..	Int	%MW6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
16	Salida conversión analógica M1	Tabla de variables e..	Int	%MW8		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
17	Salida pulsos encoder B1	Tabla de variables e..	DInt	%MD12		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
18	salida conversión de pulsos encod...	Tabla de variables e..	Real	%MD16		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
19	Salida Encoder B1	Tabla de variables e..	Real	%MD28		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
20	Marca del sensor B10	Tabla de variables e..	Bool	%M36.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
21	Quite de energia del motor M1 co...	Tabla de variables e..	Bool	%M36.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
22	Frenado del motor M1 con sensor...	Tabla de variables e..	Bool	%M36.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
23	Arranque de motor M1 desde el se...	Tabla de variables e..	Bool	%M36.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
24	Inicio del proceso del motor DC M1	Tabla de variables e..	Bool	%M37.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figura 220. Variables del PLC inicial.

	Nombre	Tabla de variables	Tipo de datos	Dirección	Rema...	Visibl...	Acces...	Coment
25	Bipolar unscable	Tabla de variables e..	Bool	%M37.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
26	Reseteo 2 del sw_gate del encoder	Tabla de variables e..	Bool	%M37.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
27	Reseteo 1 del sw_gate del encoder	Tabla de variables e..	Bool	%M37.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
28	Desactivación del quite de energia	Tabla de variables e..	Bool	%M37.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
29	Arranque del motor M1 despues d...	Tabla de variables e..	Bool	%M37.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
30	Marca del sensor B11	Tabla de variables e..	Bool	%M60.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
31	Frenado del motor M1 con el sens...	Tabla de variables e..	Bool	%M60.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
32	<Agregar>					<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figura 221. Variables del PLC final.

Procedemos a crear la función de conversión analógica la cual permite ingresar el valor de voltaje en voltios para energizar los motores DC M1 y M2.

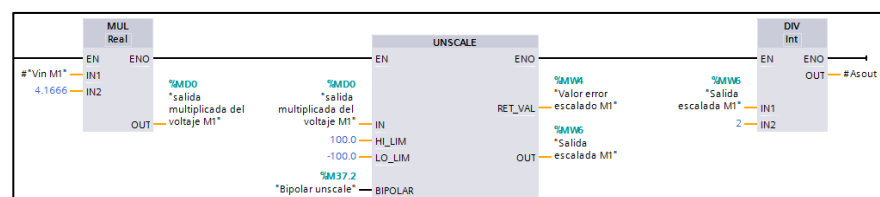


Figura 222. Bloque de función conversión analógica.

Al activar la entrada I125.0 se da inicio al programa.

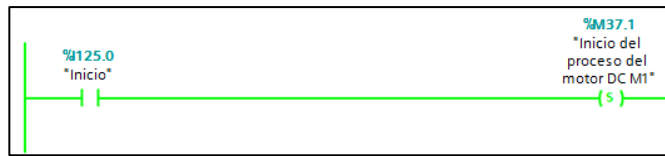


Figura 223. Inicio del programa activo.

Ingresamos un valor de voltaje para energizar el motor DC M1 y lo hacemos desplazar hasta el sensor B10 de lado derecho para comprobar su funcionamiento.

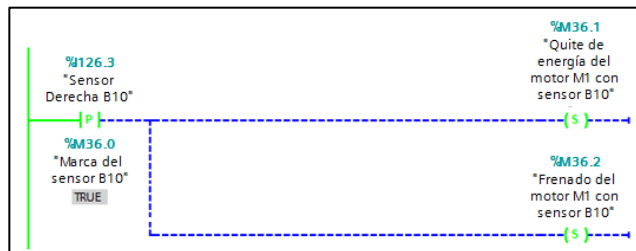


Figura 224. Entrada del sensor B10 activo.



Figura 225. Sensor derecha B10 activado.

De la misma manera ingresamos un voltaje inverso para que el motor DC M1 se desplace a lado izquierdo y comprobar el funcionamiento del sensor B11.

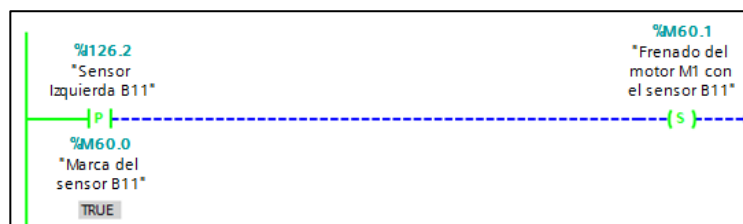


Figura 226. Entrada del sensor B11 activo.



Figura 227. Sensor derecha B11 activado.

Después ingresamos un valor de voltaje para energizar el motor DC M2 y lo

hacemos desplazar hasta el sensor B12.

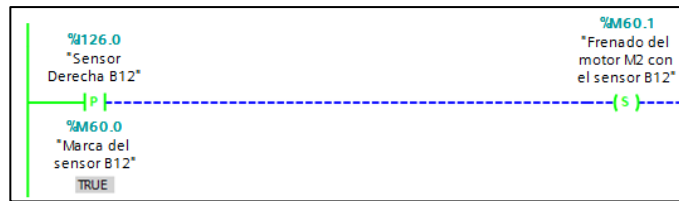


Figura 228. Entrada del sensor B12 activa en el programa.



Figura 229. Sensor alto B12 activado.

De la misma manera ingresamos un voltaje inverso para que el motor DC M2 se desplace a lado izquierdo y comprobar el funcionamiento del sensor B13.

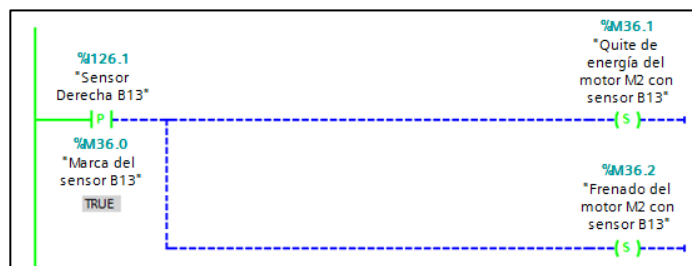


Figura 230. Entrada del sensor B13 activa en el programa.



Figura 231. Sensor alto B13 activado.

Después verificamos el estado del gripper, cuando está abierta, se encuentra activo el sensor B17.



Figura 232. Salida del solenoide del gripper activa en el programa.

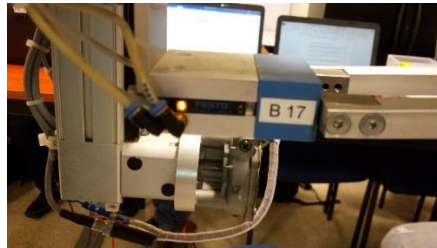


Figura 233. Sensor alto B17 activado.

Después verificamos el estado de la solenoide Y1 a través de la salida Q124.4, si esta desactivada está en la posición posterior, si Q124.4 está activa, la posición de Y1 es frontal.



Figura 234. Salida del solenoide del eje y activa en el programa.



Figura 235. Sensor alto B15 activado.

Procedemos a crear la función de conversión de datos del encoder la cual permite convertir los datos obtenidos en centímetros de ambos motores DC M1 y M2.

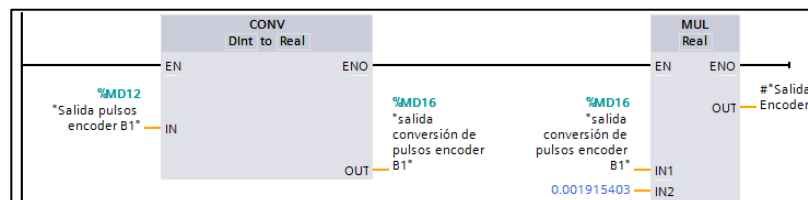


Figura 236. Conversión de encoder M1 en tía portal.

Comprobamos la lectura del encoder B1.

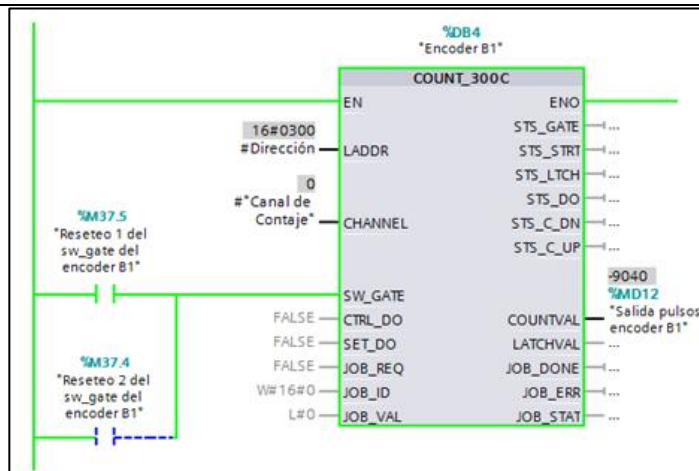


Figura 237. Lectura del encoder B1 funcionando.

Después comprobamos la lectura del encoder B2.

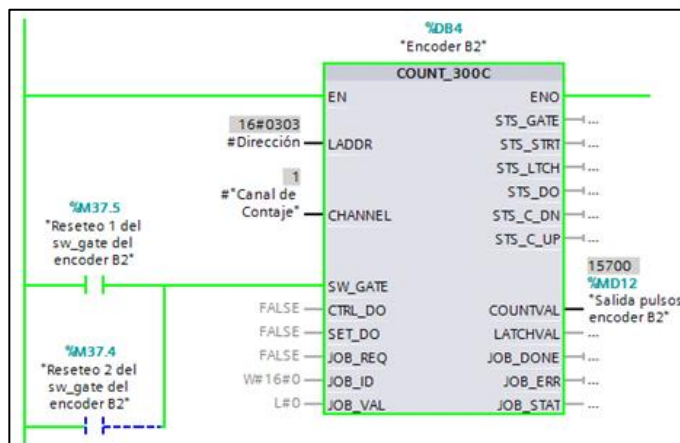

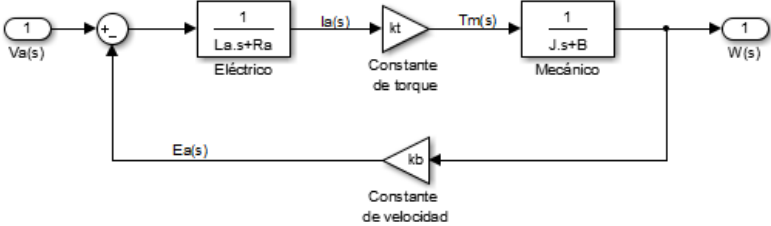


Figura 238. Lectura del encoder B2 funcionando.

Recomendaciones:

- Revisar la correcta parametrización de las entradas y salidas del PLC, ya que un error dejaría sin efecto algún sensor o actuador de la planta.
- Revisar la correcta parametrización de los canales de conteo, ya que si estos no están bien parametrizados los encoders no tendrán una correcta lectura.

4.2. Práctica Nro. 2

		GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA: Ingeniería Electrónica		ASIGNATURA:
NRO. PRÁCTICA:	2	TÍTULO PRÁCTICA: Obtención de la función de transferencia para el motor del eje vertical y horizontal.
Objetivos: <ul style="list-style-type: none"> • Obtener la función transferencia del motor DC M1 del eje horizontal. • Obtener la función transferencia del motor DC M2 del eje vertical. • Obtener la gráfica de la función transferencia del motor DC M1 y M2 correspondiente a la velocidad. 		
INSTRUCCIONES:		1. Disposición de los parámetros constructivos de los motores DC M1 y M2.
		2. Determinación de la función transferencia de los motores DC M1 y M2 por medio de los cálculos respectivos.
		3. Creación de un nuevo programa en MATLAB para la obtención de las gráficas de velocidad de los motores DC M1 y M2.
ACTIVIDADES POR DESARROLLAR		
Dada las expresiones resueltas anteriormente tales como: $T_m(s) = k_t I_a(s)$ $E_a(s) = k_b \Omega(s)$ $V_a(s) = L_a s I_a(s) + R_a I_a(s) + E_a(s)$ $T_m(s) = J s \Omega(s) + B \Omega(s) + T_L(s)$		
Se puede diseñar el siguiente diagrama de bloques.		
		
Figura 239. Diagrama de bloques de los motores DC M1 y M2.		
Del cual se obtiene la siguiente función transferencia correspondiente a la velocidad:		
$\frac{\omega(s)}{V_a(s)} = \frac{kt}{JL_a s^2 + BL_a s + JR_a s + BR_a + k_t k_b}$		
Los parámetros constructivos a utilizar para cada motor DC son los que se determinó por las pruebas que se realizó a los motores en la sección anterior. Los parámetros constructivos del motor DC M1 son:		

$$\begin{aligned}
R1 &= 4.4571 \\
L1_a &= 0.00087053 \\
k1_b &= 0.051595966 \\
k1_t &= 0.051595966 \\
B1 &= 0.000312892 \\
J1 &= 0.0000219256
\end{aligned}$$

Y los parámetros constructivos del motor DC M2 son :

$$\begin{aligned}
R2 &= 7.24285 \\
L2_a &= 0.00091608 \\
k2_b &= 0.050447731 \\
k2_t &= 0.050447731 \\
B2 &= 0.00031466 \\
J2 &= 0.000019457
\end{aligned}$$

Con los parámetros descritos de los motores DC M1 y M2 obtenemos las funciones de transferencia correspondientes.

Primero obtenemos la función transferencia del motor DC M1.

$$\begin{aligned}
\frac{\omega1(s)}{V1_a(s)} &= \frac{k1_t}{J1L1_a s^2 + B1L1_a s + J1R1_a s + B1R1_a + k1_t k1_b} \\
\frac{\omega1(s)}{V1_a(s)} &= \frac{0.051595966}{1.91e^{-8}s^2 + 2.724e^{-7}s + 9.77246e^{-5}s + 1.3945e^{-3} + 2.6621e^{-3}} \\
\frac{\omega1(s)}{V1_a(s)} &= \frac{0.051595966}{1.91e^{-8}s^2 + 9.7997e^{-5}s + 4.0566e^{-3}}
\end{aligned}$$

Luego obtenemos la función transferencia del motor DC M2.

$$\begin{aligned}
\frac{\omega2(s)}{V2_a(s)} &= \frac{k2_t}{J2L2_a s^2 + B2L2_a s + J2R2_a s + B2R2_a + k2_t k2_b} \\
\frac{\omega2(s)}{V2_a(s)} &= \frac{0.050447731}{1.78e^{-8}s^2 + 2.883e^{-7}s + 1.409e^{-4}s + 2.279e^{-3} + 0.002662} \\
\frac{\omega2(s)}{V2_a(s)} &= \frac{0.051595966}{1.78e^{-8}s^2 + 0.00014118s + 0.004941}
\end{aligned}$$

Obtenidas las funciones transferencia de los motores DC M1 y M2, se procede a realizar el programa en el software MATLAB.

Creamos un nuevo script y colocamos los parámetros de cada motor DC.

```

6 -   clc
7 -   clear all
8 -   close all
9 -   %Datos del Motor DC M1
10 -  R1 = 4.4571;           %Resistencia de Armadura
11 -  L1 = 0.00087053;     %Inductancia de Armadura
12 -  kb1 = 0.051595966;  %Constante de la fuerza contra electromotriz
13 -  kt1 = 0.051595966;  %Constante del par electromecánico
14 -  B1 = 0.000312892;   %Coeficiente de fricción viscosa
15 -  J1 = 0.0000219256;  %Momento de inercia del motor
16 -  %Datos del Motor DC M2
17 -  R2 = 7.24285;       %Resistencia de Armadura
18 -  L2 = 0.00091608;   %Inductancia de Armadura
19 -  kb2 = 0.050447731; %Constante de la fuerza contra electromotriz
20 -  kt2 = 0.050447731; %Constante del par electromecánico
21 -  B2 = 0.00031466;   %Coeficiente de fricción viscosa
22 -  J2 = 0.000019457;  %Momento de inercia del motor

```

Figura 240. Datos de los motores DC M1 y M2

Después se programa el código el cual obtenemos la función transferencia de velocidad del motor DC M1.

```

24 - %Función transferencia del motor DC M1 (velocidad)
25 - M1a = tf(kt1,[J1*L1 (J1*R1)+(B1*L1) (B1*R1)+(kt1*kb1)])
26 - figure(1)
27 - step(M1a)

```

Figura 241. Código de la función transferencia del motor DC M1

```

M1a =

          0.0516
-----
1.909e-08 s^2 + 9.8e-05 s + 0.004057

Continuous-time transfer function.

```

Figura 242. Función transferencia de velocidad del motor DC M1

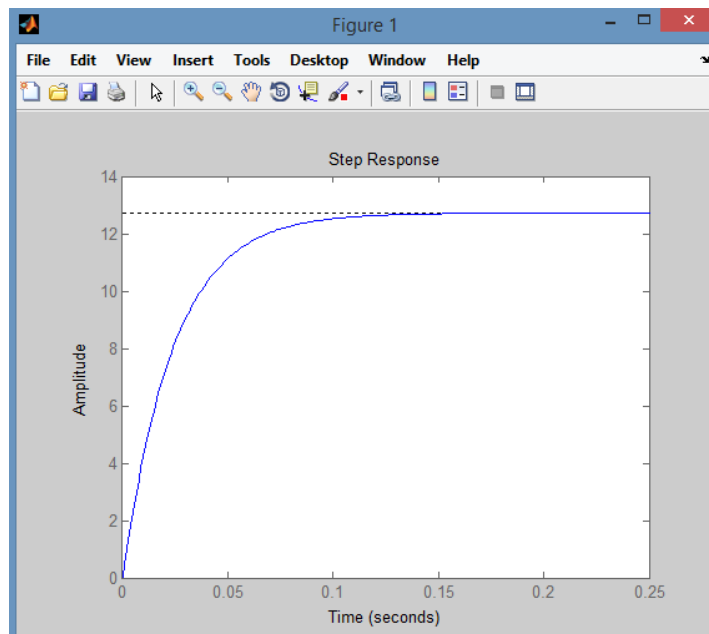


Figura 243. Función transferencia de velocidad del motor DC M1

Luego se programa el código el cual obtenemos la función transferencia de velocidad del motor DC M2.

```

28 %Función transferencia del motor DC M2 (velocidad)
29 M2a = tf(kt2, [(J2*L2 (J2*R2)+(B2*L2) (B2*R2)+(kt2*kb2)])
30 figure(2)
31 step(M2a)

```

Figura 244. Código de la función transferencia del motor DC M2

```

M2a =
      0.05045
-----
 1.782e-08 s^2 + 0.0001412 s + 0.004824
Continuous-time transfer function.

```

Figura 245. Función transferencia de velocidad del motor DC M2

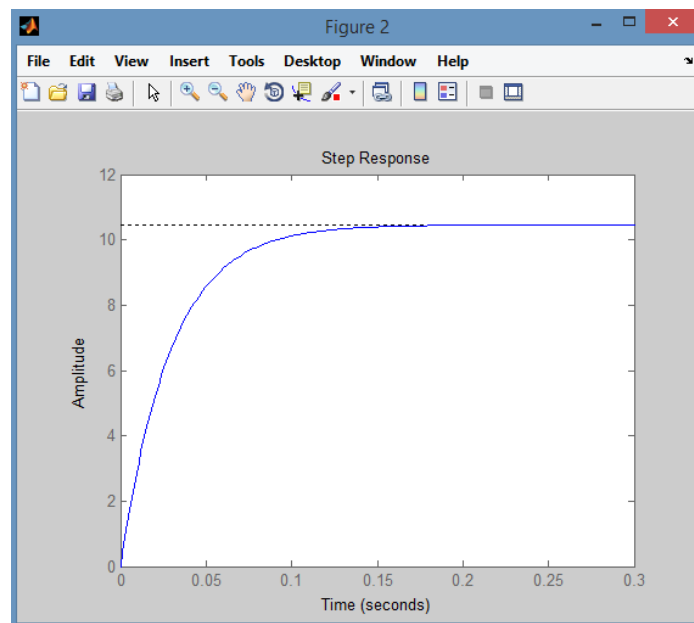

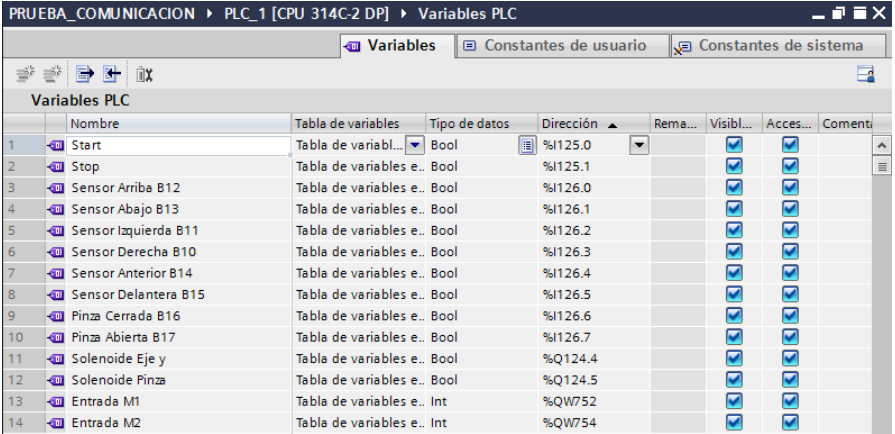


Figura 246. Función transferencia de velocidad del motor DC M2

Recomendaciones:

- Revisar el correcto reemplazo en las fórmulas de las funciones transferencia.
- Revisar la correcta codificación en el programa MATLAB para la obtención de la función transferencia y su respectiva gráfica.

4.3. Práctica Nro. 3

		GUÍA DE PRÁCTICA DE LABORATORIO																																																																																																																								
CARRERA: Ingeniería Electrónica		ASIGNATURA:																																																																																																																								
NRO. PRÁCTICA:	3	TÍTULO PRÁCTICA: Diseño del sistema SCADA en LabView para el monitoreo del eje vertical y horizontal.																																																																																																																								
Objetivos: <ul style="list-style-type: none"> Realizar un programa en TIA PORTAL para el desplazamiento de los motores DC M1 y M2 (eje vertical y eje horizontal). Realizar un programa en KEPServerEX para comunicar las variables del a utilizar en el programa en TIA PROTAL al Scada en el programa LabView. Realizar un Scada en el programa LabView que permita monitorear los motores DC M1 y M2 (eje vertical y eje horizontal). 																																																																																																																										
INSTRUCCIONES:		<ol style="list-style-type: none"> Creación de los bloques de funciones para energizar los motores DC M1 y M2. Creación de los bloques de funciones para los encoders para lectura de desplazamiento de los motores DC M1 y M2. Realización de la comunicación Ethernet para el monitoreo con el Scada en LabView. 																																																																																																																								
ACTIVIDADES POR DESARROLLAR																																																																																																																										
<p>Creamos un programa en el software TIA PORTAL donde ingresamos las variables de entrada y salida correspondientes a los motores y a los sensores:</p>																																																																																																																										
 <table border="1"> <thead> <tr> <th>Nombre</th> <th>Tabla de variables</th> <th>Tipo de datos</th> <th>Dirección</th> <th>Rema...</th> <th>Visibl...</th> <th>Acces...</th> <th>Comenti</th> </tr> </thead> <tbody> <tr><td>1 Start</td><td>Tabla de variabl...</td><td>Bool</td><td>%I125.0</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> <tr><td>2 Stop</td><td>Tabla de variables e..</td><td>Bool</td><td>%I125.1</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> <tr><td>3 Sensor Arriba B12</td><td>Tabla de variables e..</td><td>Bool</td><td>%I126.0</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> <tr><td>4 Sensor Abajo B13</td><td>Tabla de variables e..</td><td>Bool</td><td>%I126.1</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> <tr><td>5 Sensor Izquierda B11</td><td>Tabla de variables e..</td><td>Bool</td><td>%I126.2</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> <tr><td>6 Sensor Derecha B10</td><td>Tabla de variables e..</td><td>Bool</td><td>%I126.3</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> <tr><td>7 Sensor Anterior B14</td><td>Tabla de variables e..</td><td>Bool</td><td>%I126.4</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> <tr><td>8 Sensor Delantera B15</td><td>Tabla de variables e..</td><td>Bool</td><td>%I126.5</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> <tr><td>9 Pinza Cerrada B16</td><td>Tabla de variables e..</td><td>Bool</td><td>%I126.6</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> <tr><td>10 Pinza Abierta B17</td><td>Tabla de variables e..</td><td>Bool</td><td>%I126.7</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> <tr><td>11 Solenoide Eje y</td><td>Tabla de variables e..</td><td>Bool</td><td>%Q124.4</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> <tr><td>12 Solenoide Pinza</td><td>Tabla de variables e..</td><td>Bool</td><td>%Q124.5</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> <tr><td>13 Entrada M1</td><td>Tabla de variables e..</td><td>Int</td><td>%QW752</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> <tr><td>14 Entrada M2</td><td>Tabla de variables e..</td><td>Int</td><td>%QW754</td><td></td><td><input checked="" type="checkbox"/></td><td><input checked="" type="checkbox"/></td><td></td></tr> </tbody> </table>			Nombre	Tabla de variables	Tipo de datos	Dirección	Rema...	Visibl...	Acces...	Comenti	1 Start	Tabla de variabl...	Bool	%I125.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		2 Stop	Tabla de variables e..	Bool	%I125.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		3 Sensor Arriba B12	Tabla de variables e..	Bool	%I126.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		4 Sensor Abajo B13	Tabla de variables e..	Bool	%I126.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		5 Sensor Izquierda B11	Tabla de variables e..	Bool	%I126.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		6 Sensor Derecha B10	Tabla de variables e..	Bool	%I126.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		7 Sensor Anterior B14	Tabla de variables e..	Bool	%I126.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		8 Sensor Delantera B15	Tabla de variables e..	Bool	%I126.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		9 Pinza Cerrada B16	Tabla de variables e..	Bool	%I126.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		10 Pinza Abierta B17	Tabla de variables e..	Bool	%I126.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		11 Solenoide Eje y	Tabla de variables e..	Bool	%Q124.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		12 Solenoide Pinza	Tabla de variables e..	Bool	%Q124.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		13 Entrada M1	Tabla de variables e..	Int	%QW752		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		14 Entrada M2	Tabla de variables e..	Int	%QW754		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Nombre	Tabla de variables	Tipo de datos	Dirección	Rema...	Visibl...	Acces...	Comenti																																																																																																																			
1 Start	Tabla de variabl...	Bool	%I125.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
2 Stop	Tabla de variables e..	Bool	%I125.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
3 Sensor Arriba B12	Tabla de variables e..	Bool	%I126.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
4 Sensor Abajo B13	Tabla de variables e..	Bool	%I126.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
5 Sensor Izquierda B11	Tabla de variables e..	Bool	%I126.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
6 Sensor Derecha B10	Tabla de variables e..	Bool	%I126.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
7 Sensor Anterior B14	Tabla de variables e..	Bool	%I126.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
8 Sensor Delantera B15	Tabla de variables e..	Bool	%I126.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
9 Pinza Cerrada B16	Tabla de variables e..	Bool	%I126.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
10 Pinza Abierta B17	Tabla de variables e..	Bool	%I126.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
11 Solenoide Eje y	Tabla de variables e..	Bool	%Q124.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
12 Solenoide Pinza	Tabla de variables e..	Bool	%Q124.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
13 Entrada M1	Tabla de variables e..	Int	%QW752		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
14 Entrada M2	Tabla de variables e..	Int	%QW754		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																																																																																																				
Figura 247. Variables de los motores M1 y M2 con sus sensores.																																																																																																																										
<p>Después procedemos a crear los bloques de función llamados conversión analógica y conversión encoder.</p>																																																																																																																										

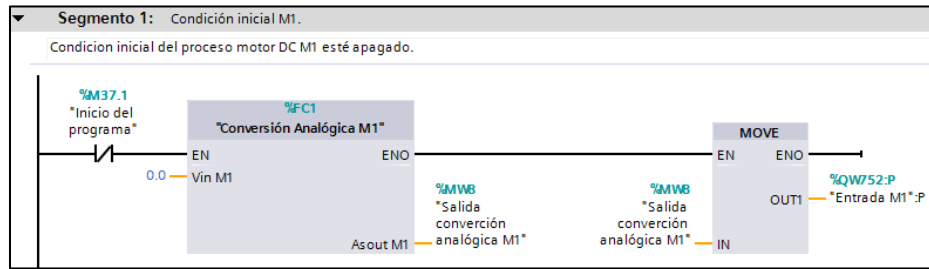


Figura 248. Conversión analógica del motor M1

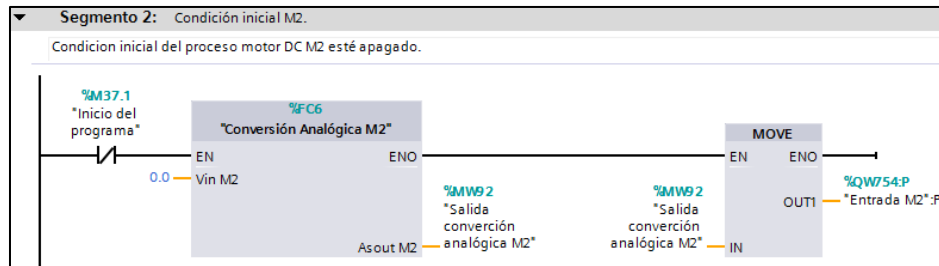


Figura 249. Conversión analógica del motor M2

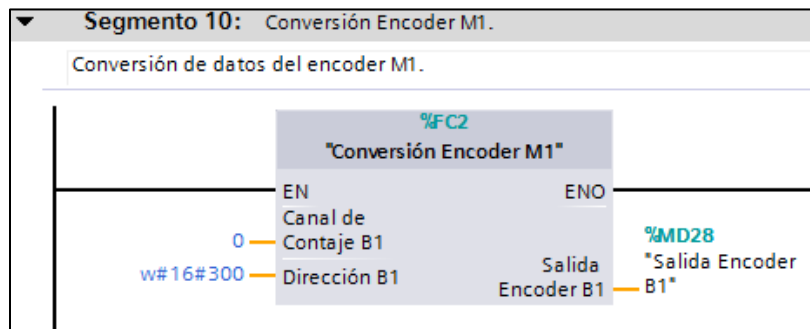


Figura 250. Conversión del encoder del motor M1

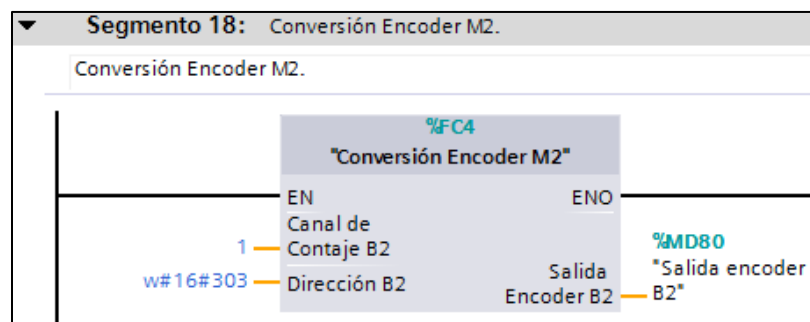


Figura 251. Conversión del encoder del motor M2

Para el bloque de función de conversión encoder, se configura el canal de conteo para cada encoder.

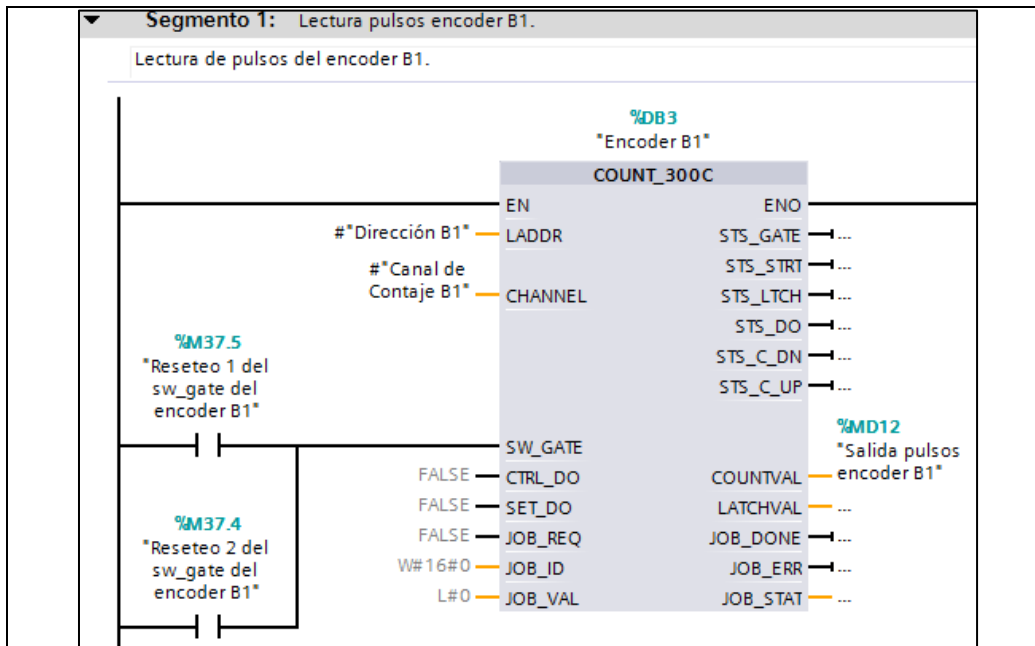


Figura 252. Bloque de función del contaje del encoder B1 del motor M1

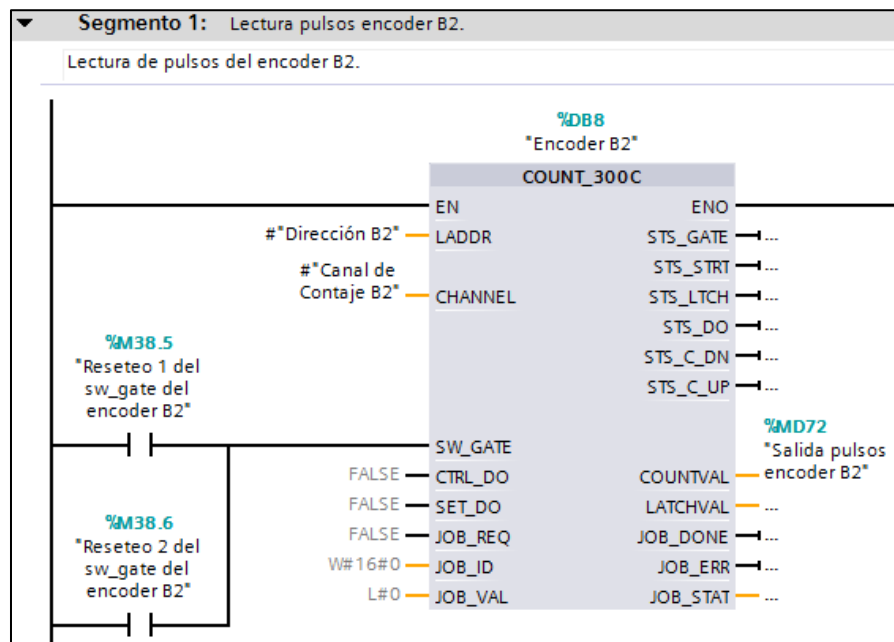


Figura 253. Bloque de función del contaje del encoder B2 del motor M2

Luego procedemos a realizar la comunicación Ethernet por medio del módulo de comunicaciones CP-343-1 Lean.

Conectamos el cable IP en el módulo y la laptop.



Figura 254. Conexión Ethernet al PC y PLC S7-314C 2DP

Después procedemos a configurar el módulo de comunicaciones asignándole una IP.

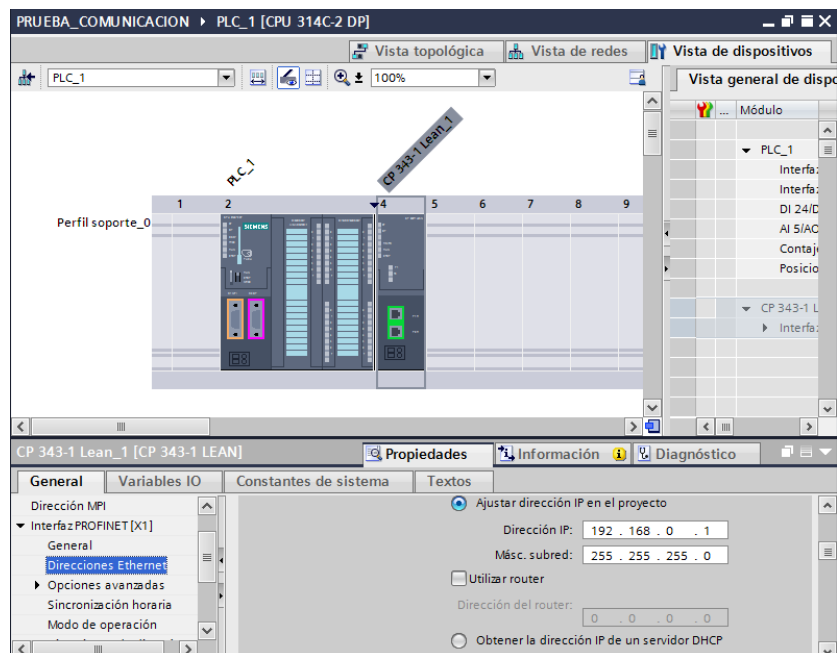


Figura 255. Asignación de la IP en el módulo de comunicación

De ahí cargamos el programa por medio de la opción de configuración de hardware para que se cargue la configuración del módulo de comunicaciones.

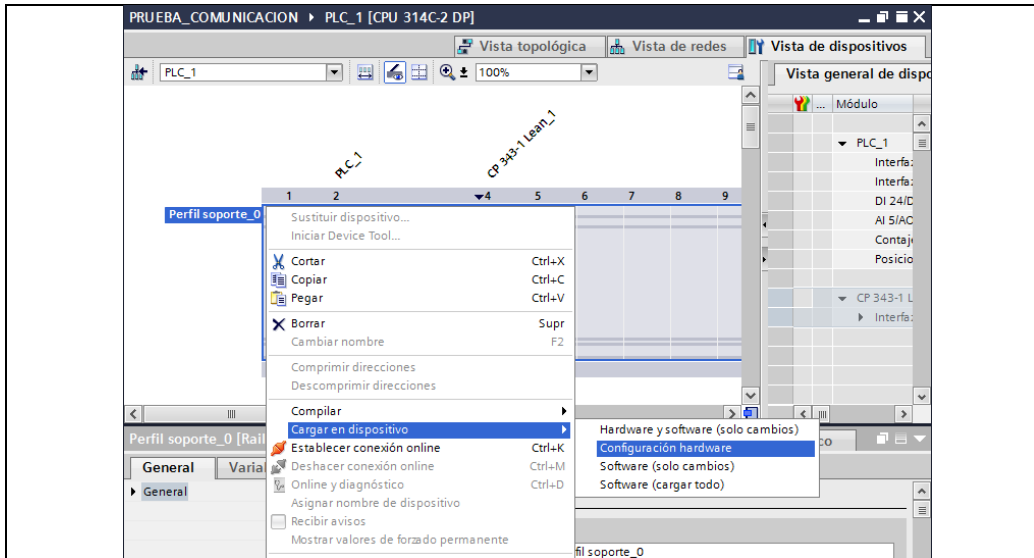


Figura 256. Carga de la configuración hardware al PLC-314C 2DP

Luego cargamos el programa cambiando el modo de carga de MPI a Ethernet.

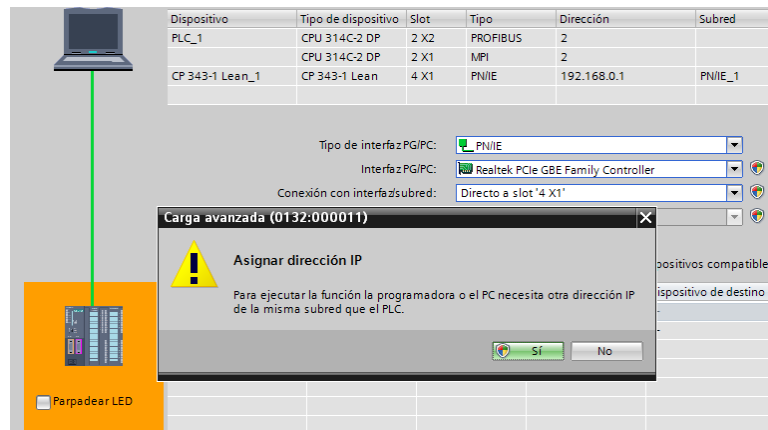


Figura 257. Carga del programa en Tia Portal vía Ethernet.

Asignamos una IP al PLC-314C 2DP para poder realizar la comunicación.

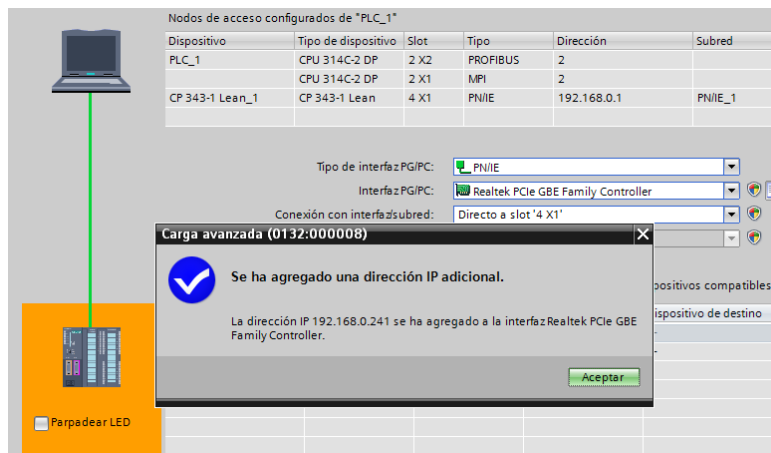


Figura 258. Asignación de la IP al PLC-314C 2DP.

Después procedemos a crear el programa en KEPServerEX.

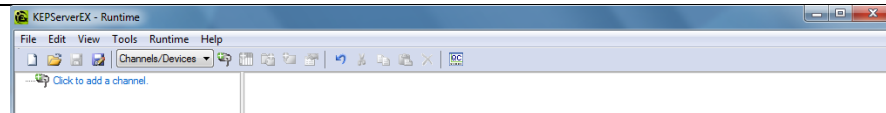


Figura 259. Creación del programa en KEPServerEX.

Procedemos a crear y configurar el canal de nombre PRÁCTICA 3, donde se podrá comunicar con el PLC-314C 2DP y donde estarán nuestras variables para el Scada con LABVIEW.

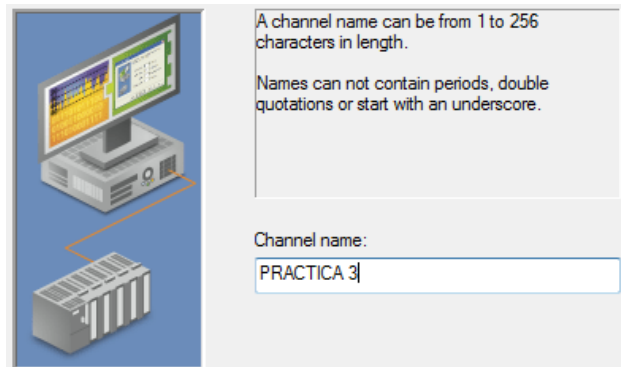


Figura 260. Creación del canal PRÁCTICA 3

Luego procedemos a configurar la vía de comunicación con el PLC-314C 2DP.

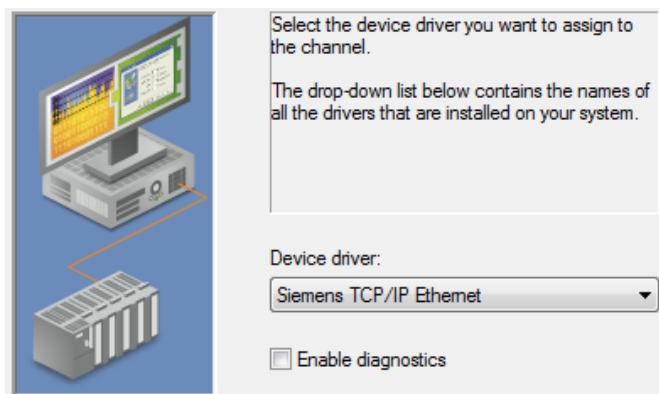


Figura 261. Seleccionamos el PLC-314C 2DP

Después procedemos a elegir la IP que se configuró en el PLC-314C 2DP.

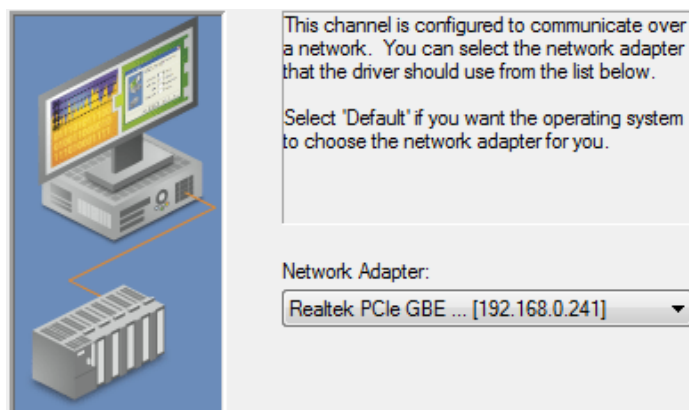


Figura 262. Selección de la IP para la comunicación con el PLC-314C 2DP.

Luego procedemos a dar finalizar para terminar la configuración del canal.

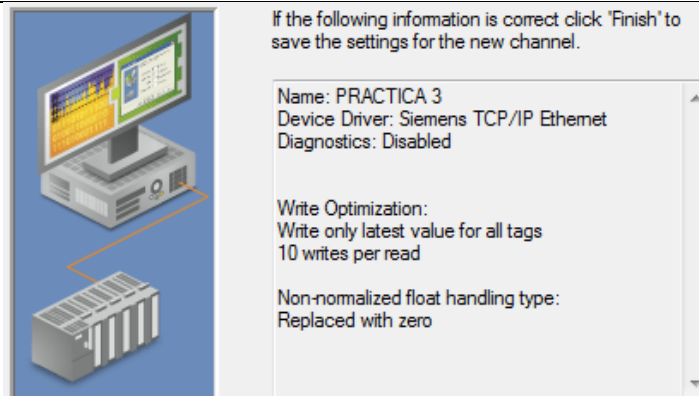


Figura 263. Sumario de la configuración del canal PRÁCTICA 3.

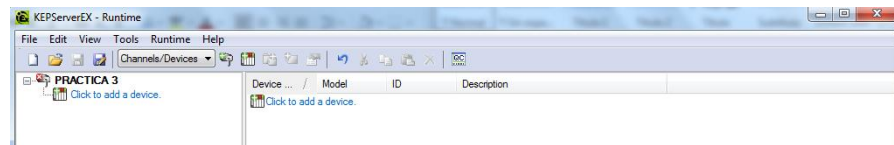


Figura 264. Canal PRÁCTICA 3 creado.

Luego procedemos a crear nuevo dispositivo con nombre MONITOREO.

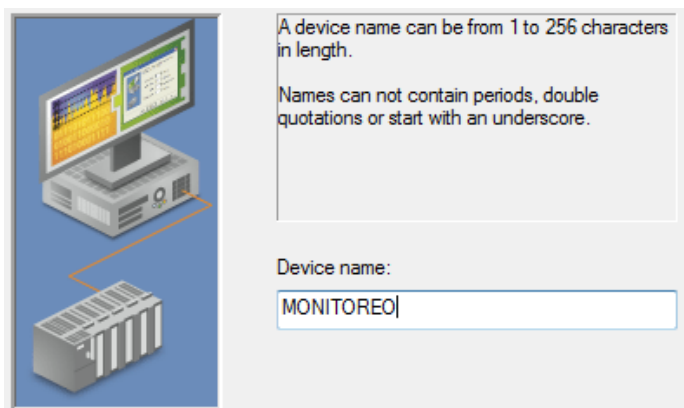


Figura 265. Creación del dispositivo MONITOREO.

Luego se procede a elegir el modelo del PLC con el cual se va a trabajar.

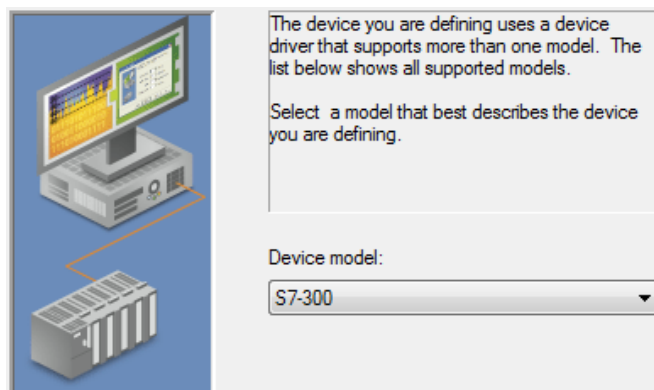


Figura 266. Modelo del PLC empleado.

Después procedemos a colocar la IP configurada en el modelo de comunicaciones CP-343-1 Lean.

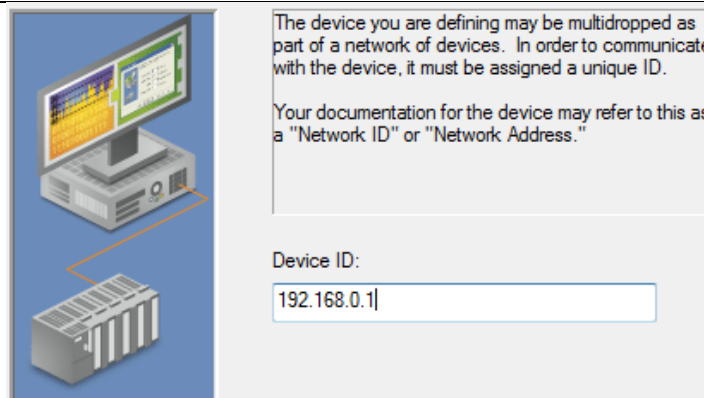


Figura 267. Colocación de la IP.

Luego damos siguiente y configuramos el número del puerto del módulo de comunicaciones donde estará conectado el cable IP.

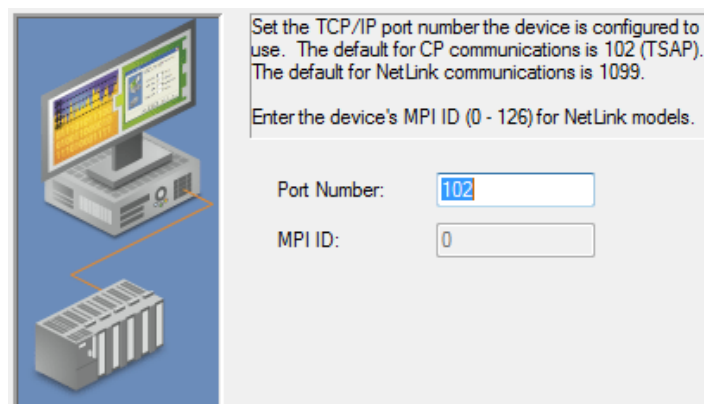


Figura 268. Configuración del Puerto .

Luego procedemos a dar siguiente y en el sumario damos por finalizar.

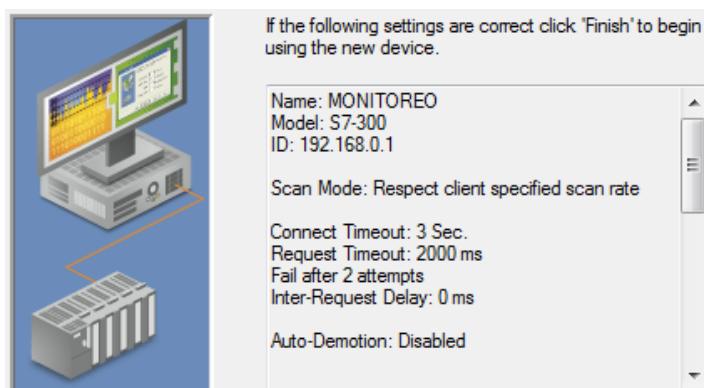


Figura 269. Sumario de la configuración del dispositivo MONITOREO.

Una vez terminado de configurar el dispositivo, se tiene listo para configurar las variables para el sistema Scada.

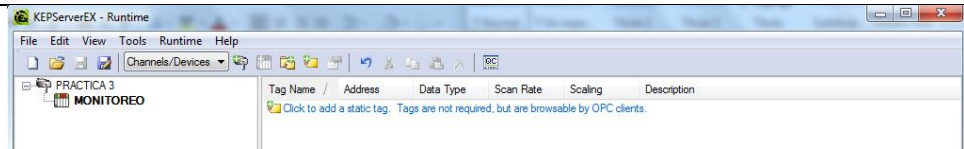


Figura 270. Canal y dispositivo configurados.

Las variables configuradas son las que se muestran en el siguiente gráfico, las cuales están siendo empleados en el programa Tia Portal.

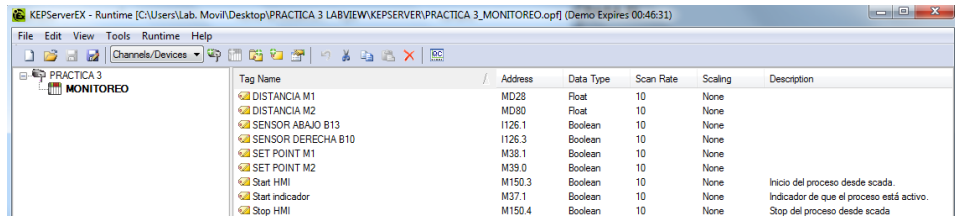


Figura 271. Configuración de las variables de la Práctica 3

Teniendo configurado las variables en el KEPServerEx para la comunicación con el sistema Scada, procedemos a realizar el programa en Tia Portal, el cual realiza el movimiento del eje horizontal y el eje vertical hacia un determinado Set Point.

Siguiendo con nuestro sistema Scada en Labview donde podremos visualizar el movimiento de los ejes de cada motor DC M1 y M2.

El programa comienza pulsando la botonera Start donde comienza el movimiento del eje horizontal por medio del motor DC M1 hasta llegar hasta su posición inicial el cual es hasta el sensor B10 donde el motor se detendrá.



Figura 272. Motor DC M1 en la posición inicial.

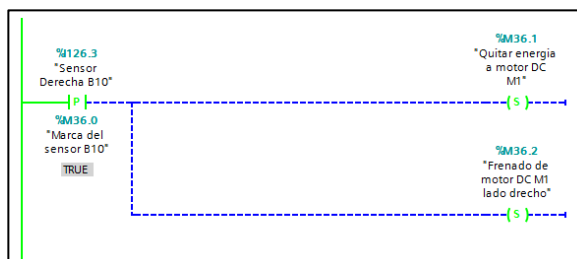


Figura 273. Detección y frenado del motor DC M1 por parte del sensor B10.

Luego se da un tiempo de 5 segundos, después el motor DC M1 arranca hacia el Set Point indicado en el Software LabView, este Set Point es de 15 centímetros.



Figura 274. Motor DC M1 del Set Point en el sistema Scada.

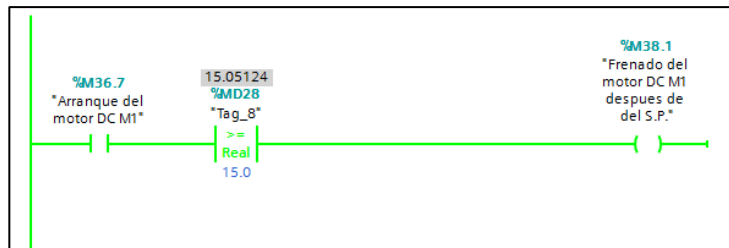


Figura 275. Detección y frenado del motor DC M1 del Set Point.

Una vez que el motor DC M1 se detiene en el Set Point asignado, arranca el motor DC M2 a la posición inicial hasta el sensor B12 el cual detiene al motor DC M2.



Figura 276. Motor DC M2 en la posición inicial sistema Scada.

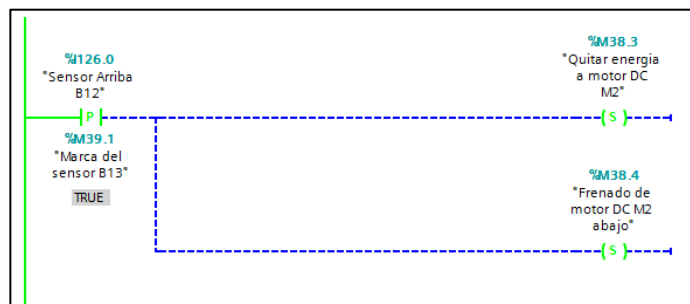


Figura 277. Detección y frenado del motor DC M2 por parte del sensor B12.

Luego se da un tiempo de 5 segundos, después el motor DC M2 arranca hacia

el Set Point indicado en el Software, este Set Point es de 15 centímetros.



Figura 278. Motor DC M2 en el Set Point del sistema Scada.

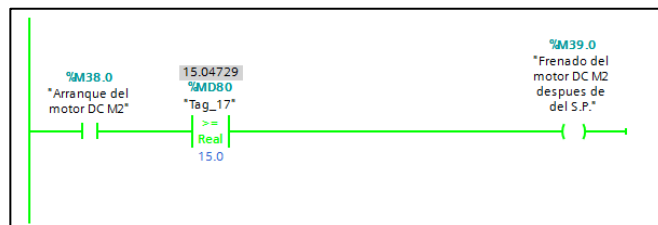

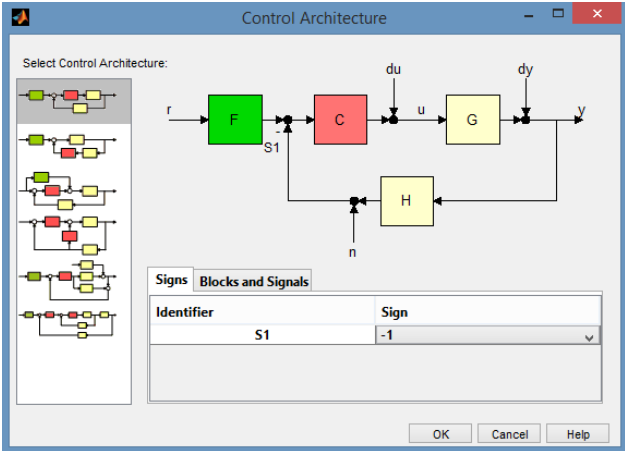


Figura 279. Detección y frenado del motor DC M2 por parte del Set Point.

Recomendaciones:

- Revisar la correcta configuración de entradas y salidas de los motores DC y los sensores.
- Revisar la correcta configuración de la IP en el módulo de comunicación.
- Tener cuidado en configurar la misma IP tanto en el Software KEPServerEX para el canal y el Device, estas deben ser las mismas IP de la PC y del módulo de comunicación respectivamente.

4.4. Práctica Nro. 4

		GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA: Ingeniería Electrónica		ASIGNATURA:
NRO. PRÁCTICA:	4	TÍTULO PRÁCTICA: Diseño del control PID para el eje en movimiento vertical.
Objetivos: <ul style="list-style-type: none"> • Diseñar el controlador PID para el motor DC M2 (eje vertical). • Obtener las constantes k_p, k_i y k_d para el controlador PID del motor DC M2 (eje vertical). • Obtener la gráfica de la función transferencia del motor DC M2 en lazo cerrado con el controlador PID. 		
INSTRUCCIONES:		1. Creación del controlador PID en la herramienta sisotool en el programa Matlab.
		2. Obtención de los parámetros PID.
		3. Simulación de la función transferencia con el controlador PID diseñado en Simulink.
ACTIVIDADES POR DESARROLLAR		
<p>Abrimos el sisotool con la función transferencia del motor DC M2 en la ventana de comando de Matlab.</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> <code>>> sisotool(M2)</code> </div> <p style="text-align: center;">Figura 280. Función sisotool de Matlab en la practica 4</p> <p>Una vez abierto la herramienta sisotool vamos a la pestaña que dice Arquitectura, seleccionamos el tipo de control.</p>		
		
Figura 281. Control de arquitectura PID del motor DC M2		
<p>Luego nos dirigimos a la pestaña de editor del compensador donde procedemos a agregar un integrador y dos ceros correspondientes al controlador PID.</p>		

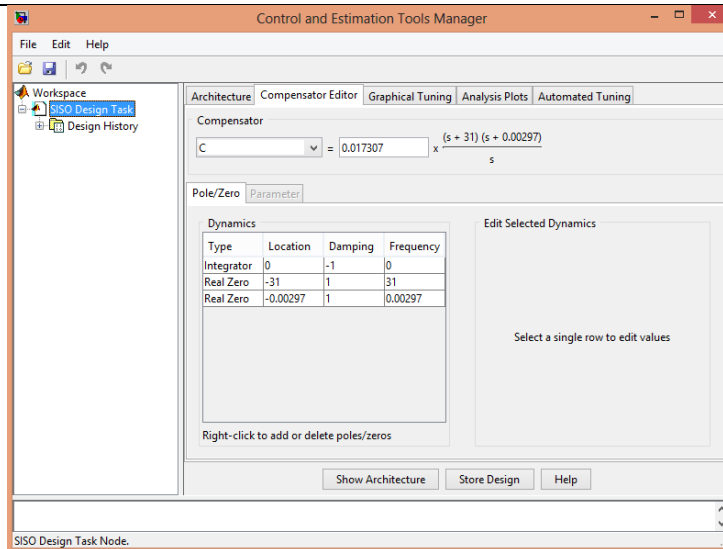


Figura 282. Configuración del compensador control PID en la práctica 4

Luego procedemos a ir a la pestaña de análisis de gráficos, donde seleccionamos en Step en el gráfico 1 y visualizamos la gráfica de lazo cerrado r to y .

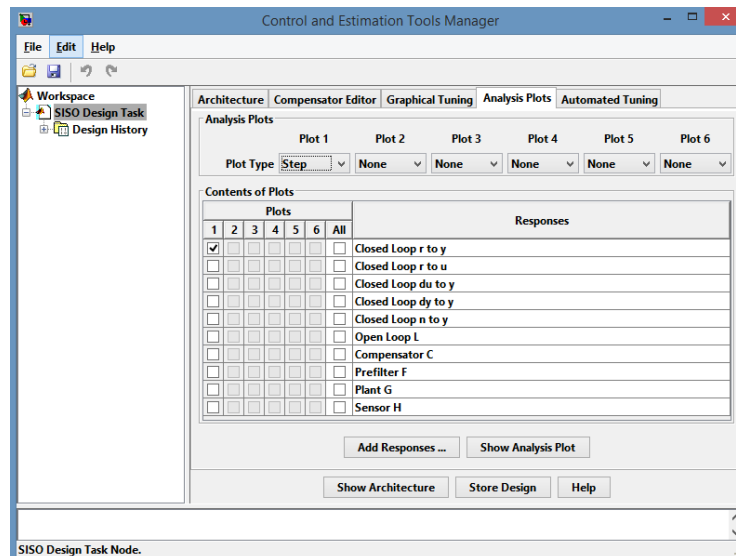


Figura 283. Configuración de la gráfica PID en lazo cerrado en la práctica 4

Luego procedemos a editar los ceros antes colocados en la ventana de diseño para ajustar la gráfica del controlador PID.

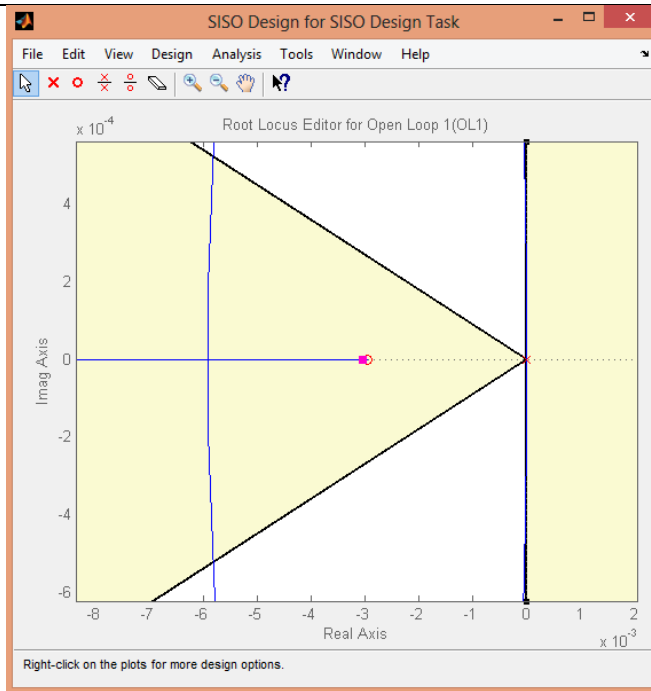


Figura 284. Ajuste del controlador PID en la práctica 4

Una vez ajustado los ceros en el lugar de las raíces, obtenemos la gráfica del controlador PID del motor DC M2 en lazo cerrado.

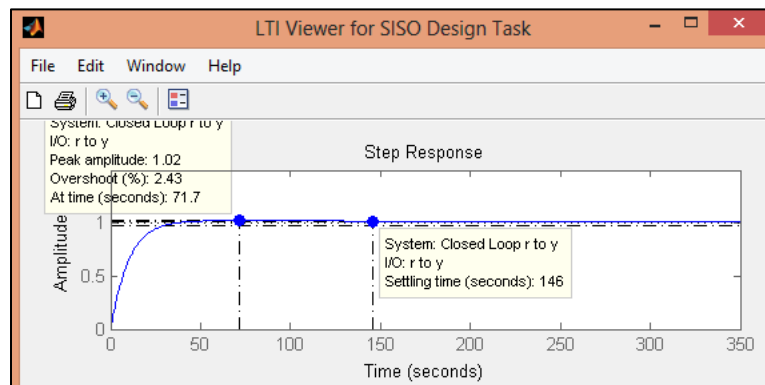


Figura 285. Gráfica del controlador PID motor DC M2 eje vertical

Una vez obtenido el controlador se procede a la obtención de los parámetros k_p , k_i y k_d .

$$G_{M2}(s) = k_p \left(1 + \frac{1}{\tau_i s} + \tau_d s \right)$$

$$G_{M2}(s) = k_p + \frac{k_p}{\tau_i s} + k_p \tau_d s$$

$$G_{M2} = k_p + \frac{k_i}{s} + k_d s$$

$$G_{M2} = k_d \left(\frac{s^2 + \frac{k_p}{k_d} s + \frac{k_i}{k_d}}{s} \right)$$

$$C_{PID}(s) = 0.017307 \frac{(s + 31)(s + 0.00297)}{s}$$

$$C_{PID}(s) = 0.017307 \left(\frac{s^2 + 31.00297s + 0.09207}{s} \right)$$

Donde:

$$\frac{k_p}{k_d} = 31.00297$$

$$\frac{k_i}{k_d} = 0.09207$$

Reemplazando:

$$k_d = 0.017307$$

$$k_p = 0.53658$$

$$k_i = 0.00159345$$

Con los parámetros obtenidos, creamos un programa en Simulink donde diseñamos el diagrama de bloques del controlador PID en lazo cerrado del motor DC M2 (eje vertical).

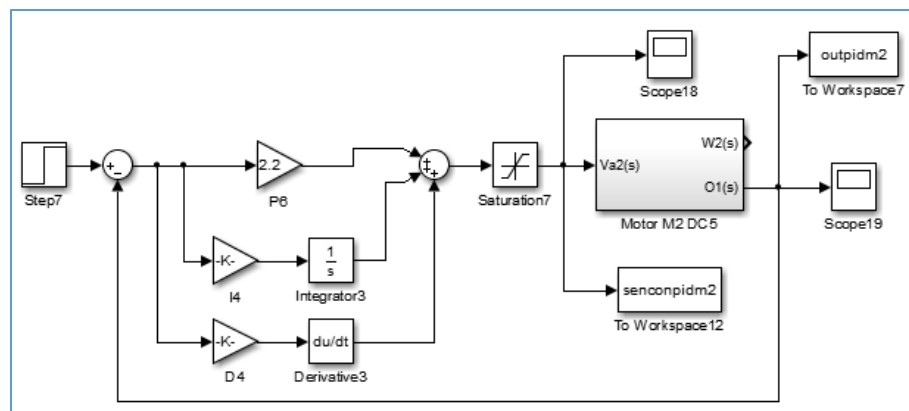


Figura 286. Diagrama de bloques del motor DC M2

Luego procedemos a simular el diagrama de bloques, obteniendo la gráfica de desplazamiento del motor DC M2 (eje vertical) con controlador PID.

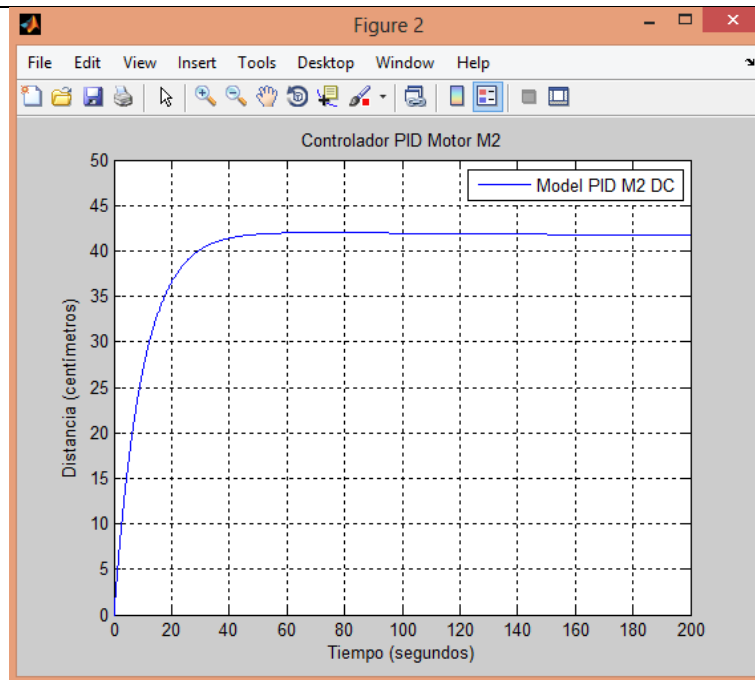



Figura 287. Gráfica del motor DC M2

Recomendaciones:

- Tener en cuenta el correcto diseño del controlador eligiendo bien los ceros a añadir al compensador.
- Tener en cuenta al momento de obtener los parámetros PID ya que de dichos valores depende el controlador PID.

4.5. Práctica Nro. 5

		GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA: Ingeniería Electrónica	ASIGNATURA:	
NRO. PRÁCTICA:	5	TÍTULO PRÁCTICA: Diseño del control PID para el eje en movimiento horizontal.
Objetivos: <ul style="list-style-type: none"> • Diseñar el controlador PID para el motor DC M1 (eje horizontal). • Obtener las constantes k_p, k_i y k_d para el controlador PID del motor DC M1 (eje horizontal). • Obtener la gráfica de la función transferencia del motor DC M1 en lazo cerrado con el controlador PID. 		
INSTRUCCIONES: <ol style="list-style-type: none"> 1. Creación del controlador PID en la herramienta sisotool en el programa Matlab. 2. Obtención de los parámetros PID. 3. Simulación de la función transferencia con el controlador PID diseñado en Simulink. 		

ACTIVIDADES POR DESARROLLAR

Abrimos el sisotool con la función transferencia del motor DC M1 en la ventana de comando de Matlab.

```
fx >> sisotool(M1)
```

Figura 288. Función sisotool en la práctica 5

Una vez abierto la herramienta sisotool vamos a la pestaña que dice Arquitectura y seleccionamos el tipo de control de arquitectura del controlador.

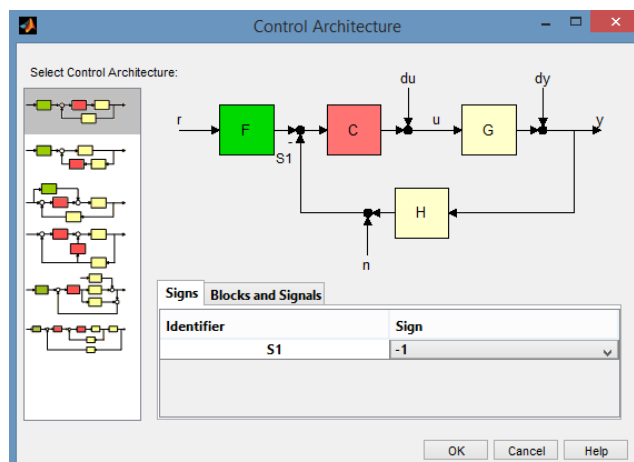


Figura 289. Control de arquitectura PID del motor DC M1 en la práctica 5

Luego nos dirigimos a la pestaña de editor del compensador donde procedemos a agregar un integrador y dos ceros correspondientes al controlador PID.

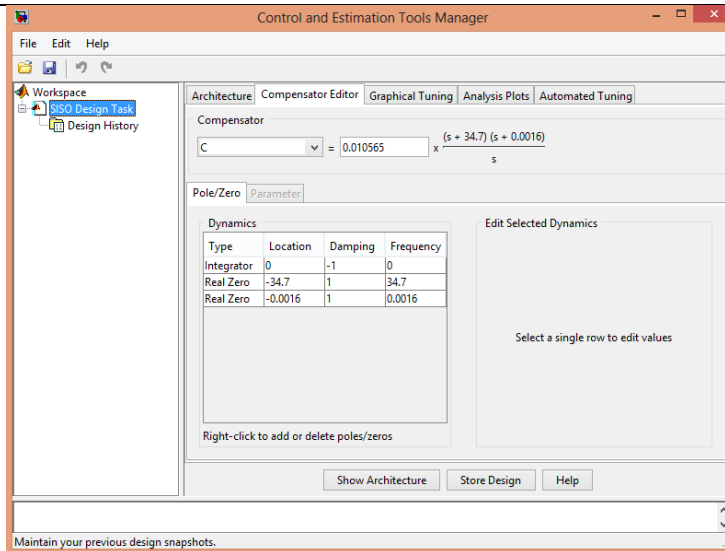


Figura 290. Configuración del compensador PID en la práctica 5

Luego procedemos a ir a la pestaña de análisis de gráficos, donde seleccionamos en Step en el gráfico 1 y visualizamos la gráfica de lazo cerrado r to y .

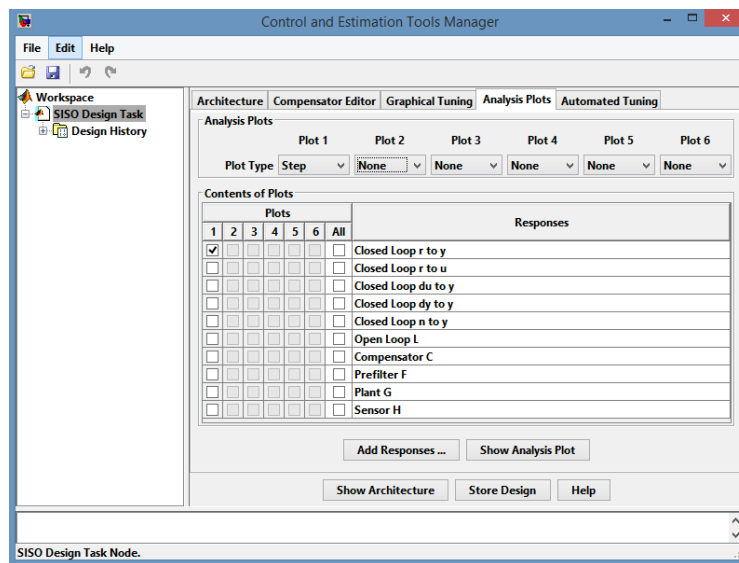


Figura 291. Configuración de la gráfica en lazo cerrado en la práctica 5

Luego procedemos a editar los ceros antes colocados en la ventana de diseño para ajustar la gráfica del controlador PID.

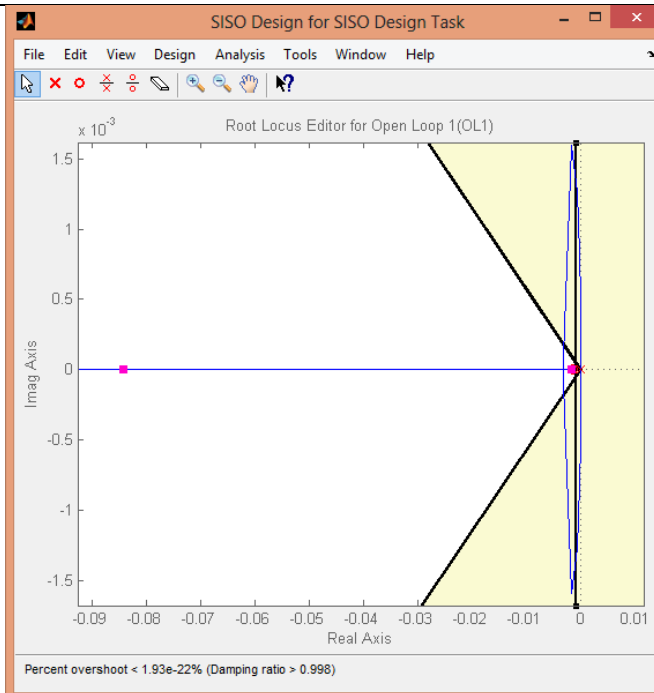


Figura 292. Ajuste del controlador PID en la práctica 5

Una vez ajustado los ceros en el lugar de las raíces, obtenemos la gráfica del controlador PID del motor DC M1 en lazo cerrado.

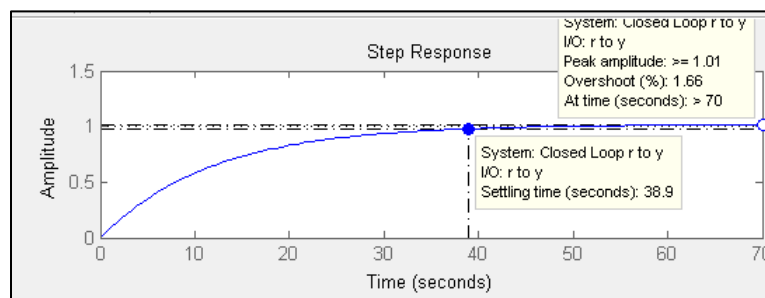


Figura 293. Gráfica del controlador PID del motor DC M1 práctica 5

Una vez obtenido el controlador se procede a la obtención de los parámetros k_p , k_i y k_d .

$$G_{M1}(s) = k_p \left(1 + \frac{1}{\tau_i s} + \tau_d s \right)$$

$$G_{M1}(s) = k_p + \frac{k_p}{\tau_i s} + k_p \tau_d s$$

$$G_{M1} = k_p + \frac{k_i}{s} + k_d s$$

$$G_{M1} = k_d \left(\frac{s^2 + \frac{k_p}{k_d} s + \frac{k_i}{k_d}}{s} \right)$$

$$C_{PID}(s) = 0.010565 \frac{(s + 34.7)(s + 0.0016)}{s}$$

$$C_{PID}(s) = 0.010565 \left(\frac{s^2 + 34.7016s + 0.05552}{s} \right)$$

Donde:

$$\frac{k_p}{k_d} = 34.7016$$

$$\frac{k_i}{k_d} = 0.05552$$

Reemplazando:

$$k_d = 0.010565$$

$$k_p = 0.3666$$

$$k_i = 0.0005865688$$

Con los parámetros obtenidos, creamos un programa en Simulink donde diseñamos el diagrama de bloques del controlador PID en lazo cerrado del motor DC M1 (eje horizontal).

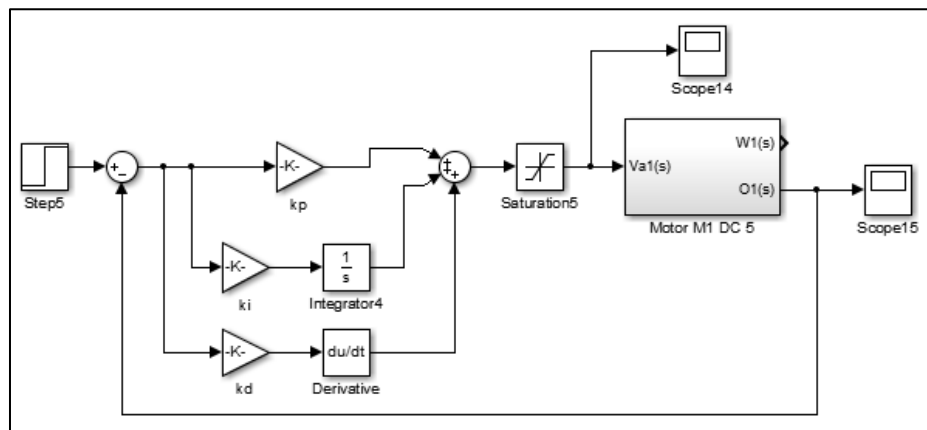


Figura 294. Diagrama de bloques del motor DC M1 práctica 5

Luego procedemos a simular el diagrama de bloques, obteniendo la gráfica de desplazamiento del motor DC M1 (eje horizontal) con controlador PID.

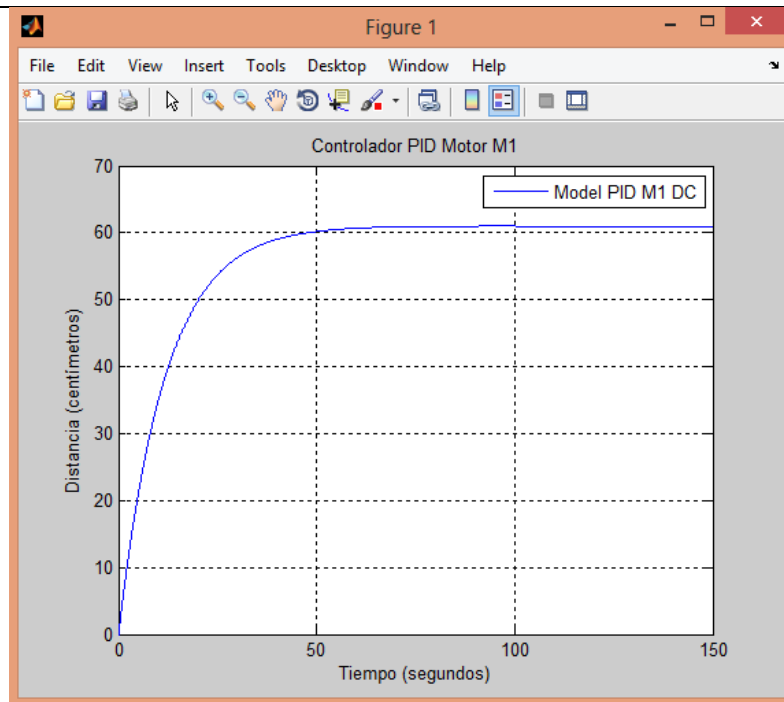



Figura 295. Gráfica del motor DC M1 práctica 5

Recomendaciones:

- Tener en cuenta el correcto diseño del controlador eligiendo bien los ceros a añadir al compensador.
- Tener en cuenta al momento de obtener los parámetros PID ya que de dichos valores depende el controlador PID.

4.6. Práctica Nro. 6

		GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA: Ingeniería Electrónica		ASIGNATURA:
NRO. PRÁCTICA:	6	TÍTULO PRÁCTICA: Diseño de un controlador difuso para el movimiento vertical.
Objetivos: <ul style="list-style-type: none"> • Diseñar el controlador Difuso para el motor DC M2 (eje vertical). • Obtener las funciones de membresía de entrada y salida del controlador Difuso para el motor DC M2 (eje vertical). • Obtener las reglas de correspondencia del controlador Difuso para el motor DC M2 (eje vertical). • Obtener la gráfica de la función transferencia del motor DC M2 en lazo cerrado con el controlador Difuso. 		
INSTRUCCIONES:		1. Creación del controlador Difuso en la herramienta Fuzzy en el programa Matlab.
		2. Obtención de las funciones de membresía de entrada y salida con sus respectivas reglas de correspondencias.
		3. Simulación de la función transferencia con el controlador Difuso diseñado en Simulink.

ACTIVIDADES POR DESARROLLAR

Abrimos el Fuzzy en la ventana de comando de Matlab.

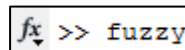


Figura 296. Función Fuzzy en la práctica 6

Una vez abierta la herramienta creamos la entrada ErrorM2 con rango de -41 hasta 41 y creamos la salida ControlM2 con rango de -29 hasta 29.

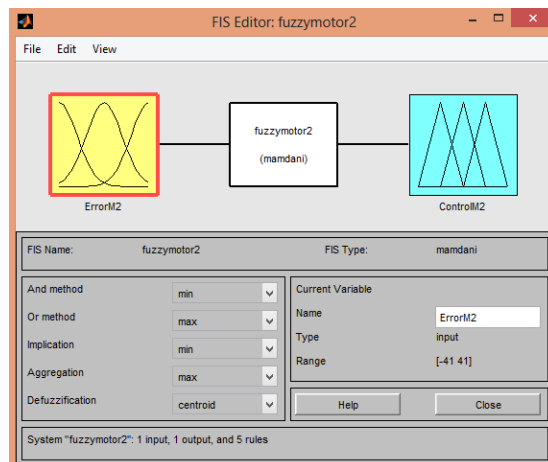


Figura 297. Creación de entrada para controlador difuso del motor DC M2

En la entrada creamos cinco funciones de membresía las cuales se llaman MN2, N2, C2, P2, MP2 para el controlador Fuzzy.

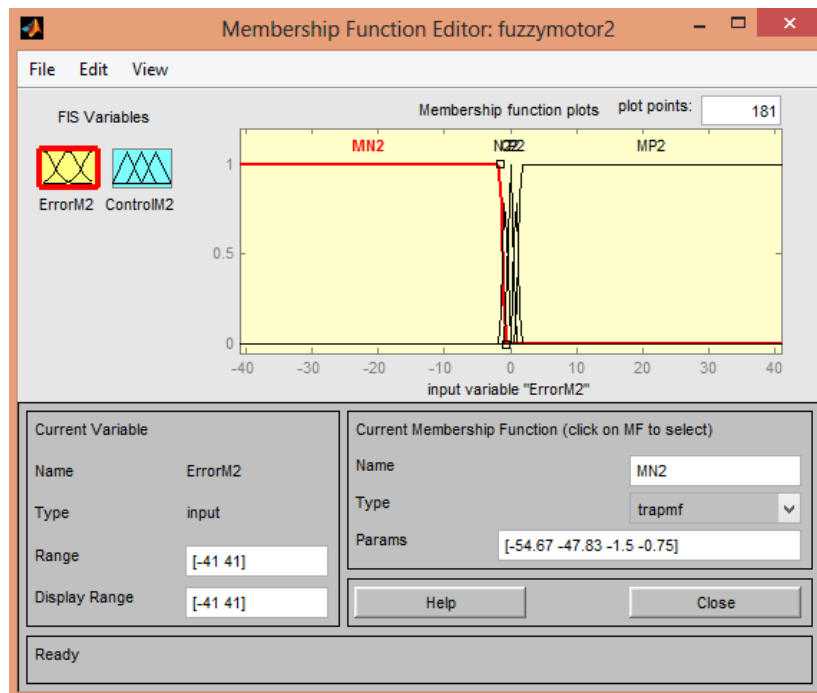


Figura 298. Función de membresía MN2 del motor DC M2

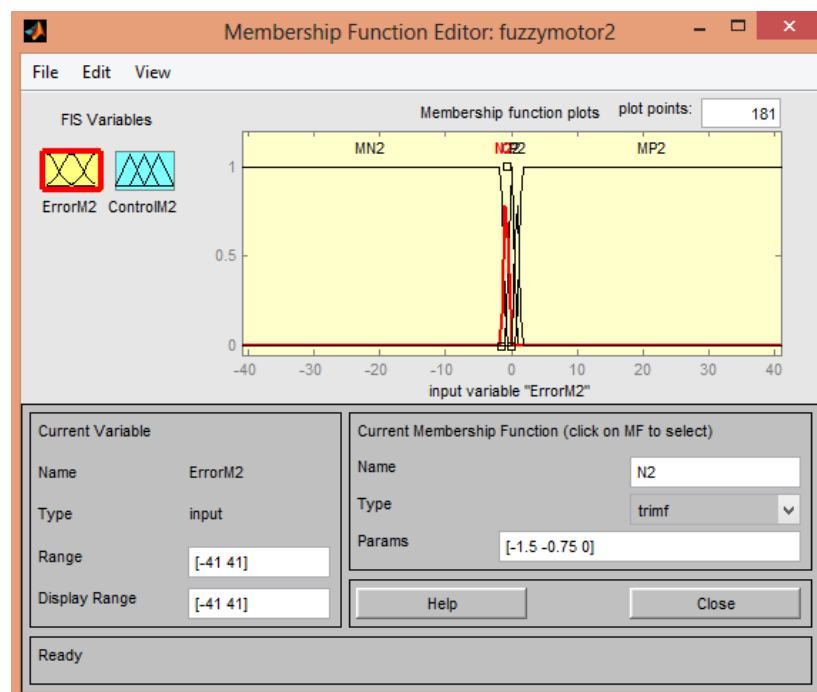


Figura 299. Función de membresía N2 del motor DC M2

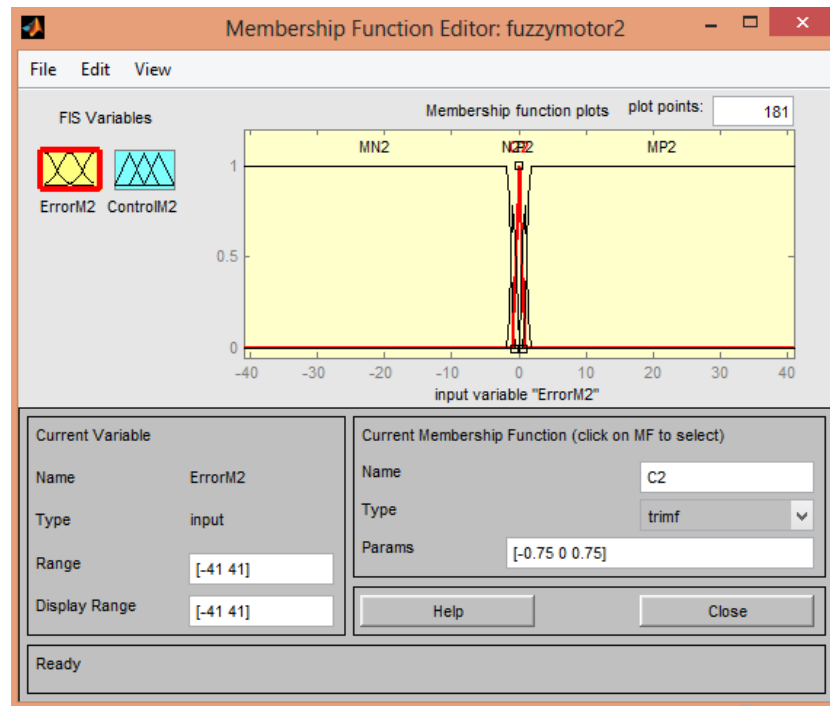


Figura 300. Función de membresía C2 del motor DC M2

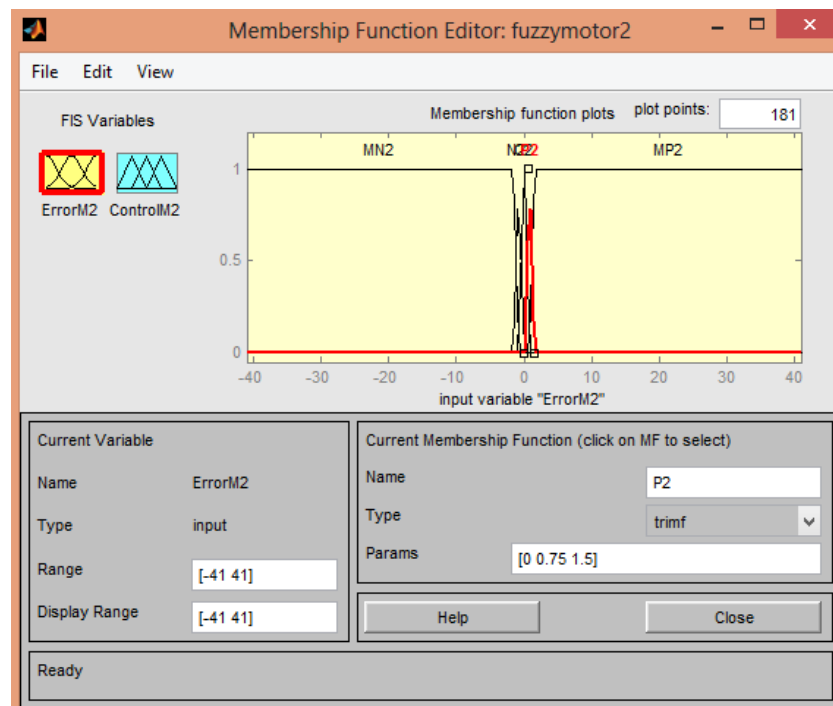


Figura 301. Función de membresía P2 del motor DC M2

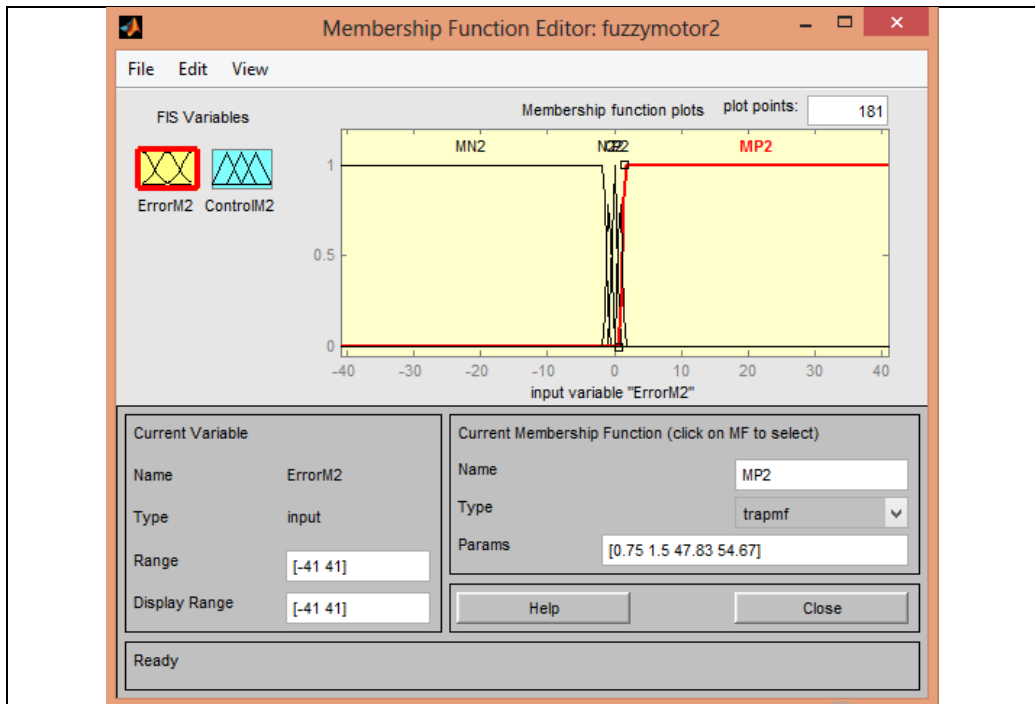


Figura 302. Función de membresía MP2 del motor DC M2

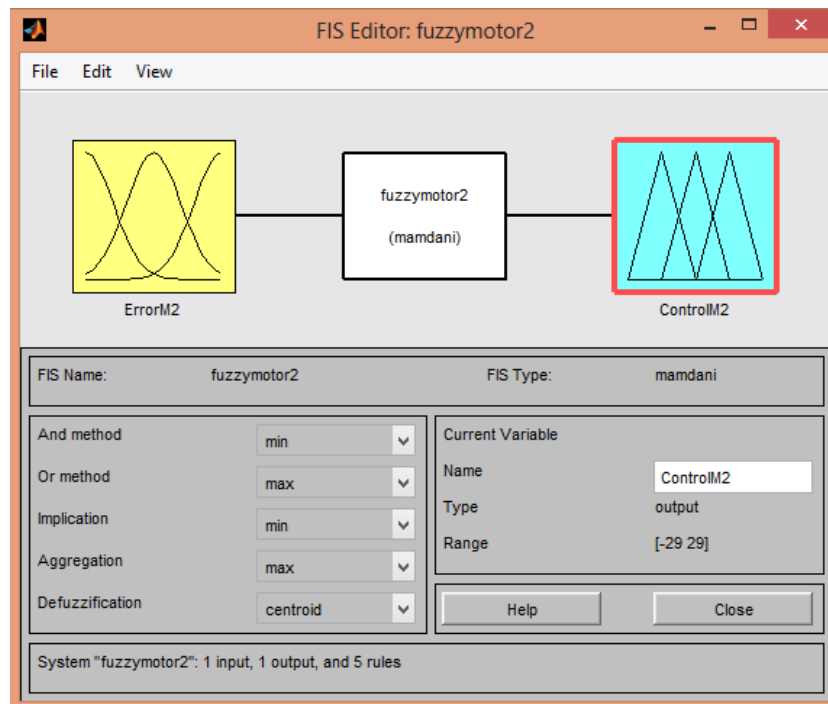


Figura 303. Creación de la salida para el controlador difuso del motor DC M2

Luego nos dirigimos a la salida donde creamos las funciones de membresía correspondientes las cuales se llaman MNeg2, Neg2, Cero2, Pos2, MPos2 para el controlador Fuzzy.

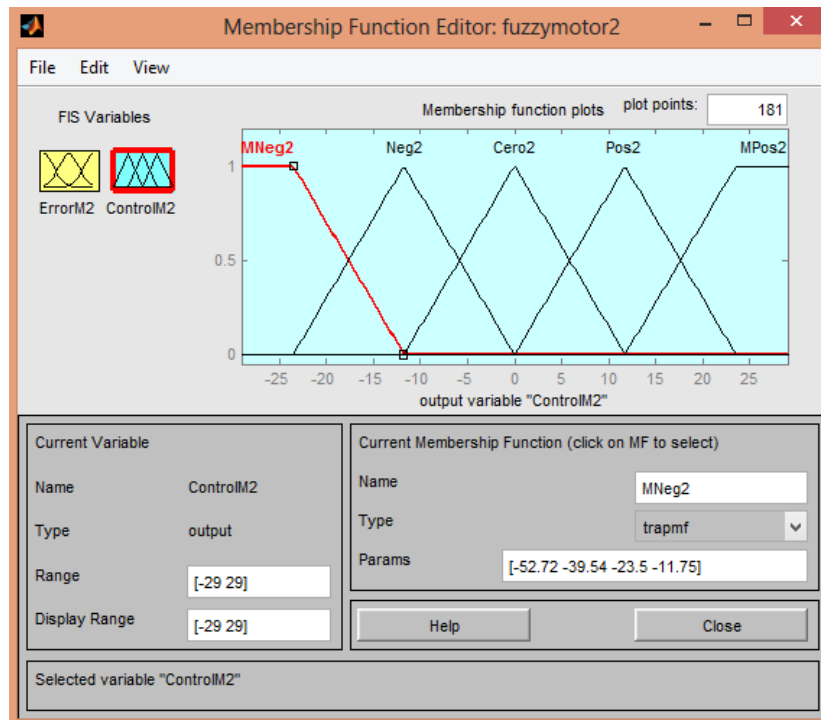


Figura 304. Función de membresía MNeg2 del motor DC M2

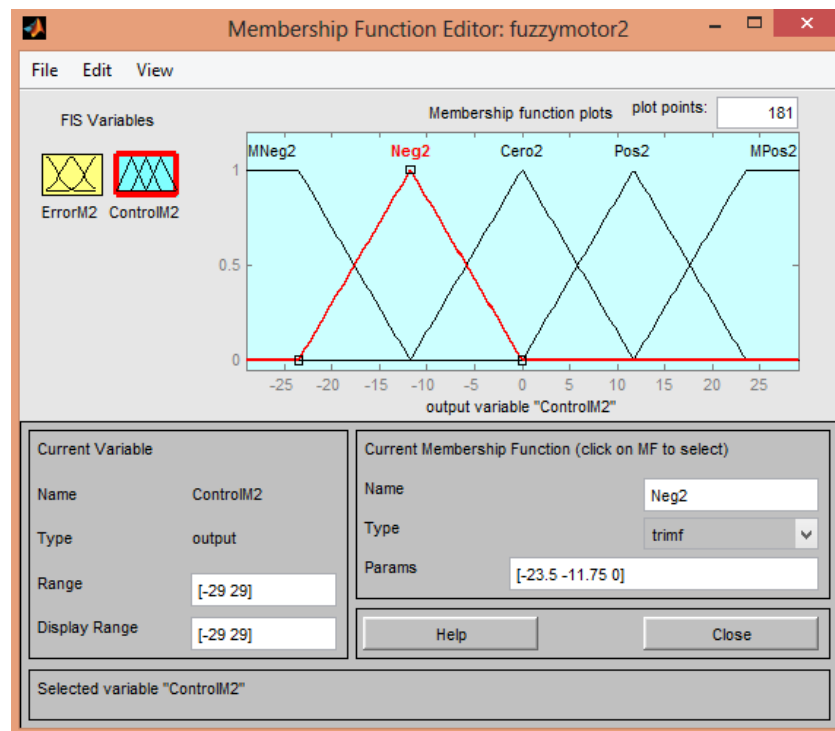


Figura 305. Función de membresía Neg2 del motor DC M2

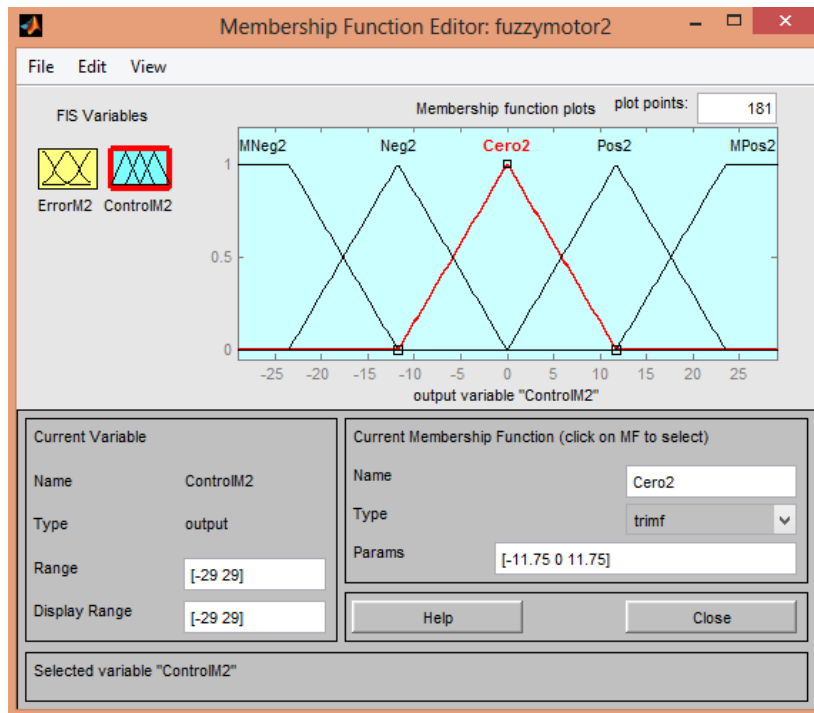


Figura 306. Función de membresía Cero2 del motor DC M2

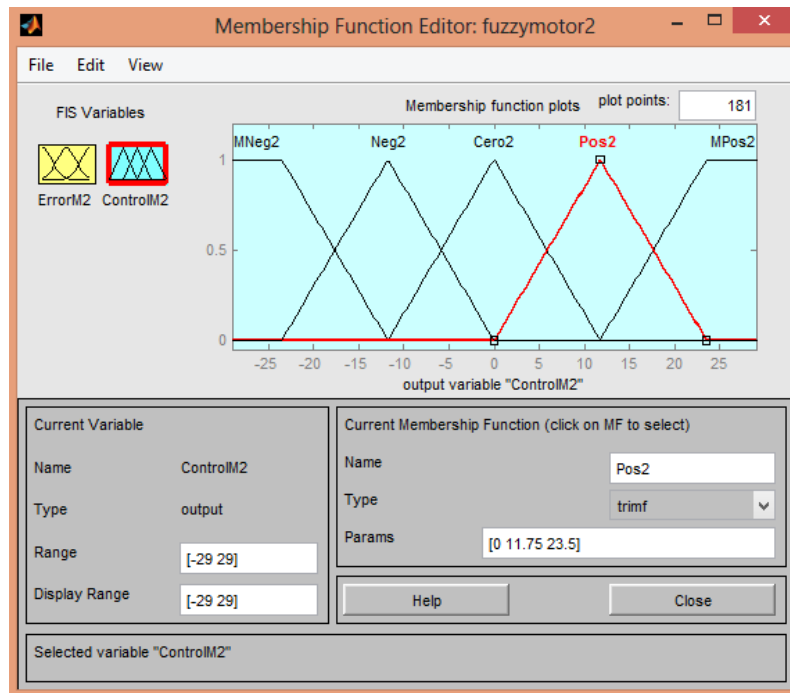


Figura 307. Función de membresía Pos2 del motor DC M2

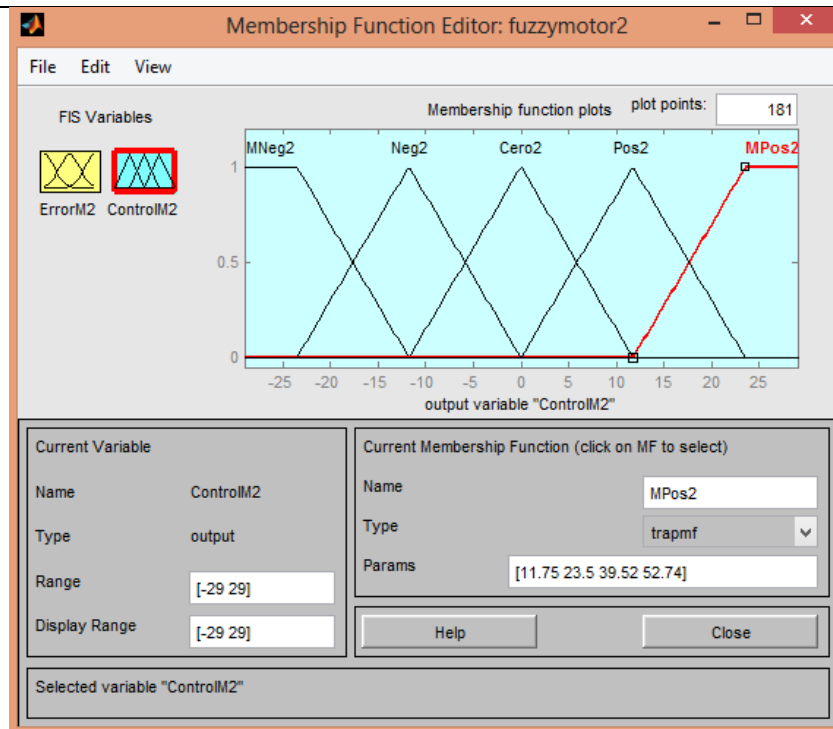


Figura 308. Función de membresía MPos2 del motor DC M2

Una vez obtenido la entrada y la salida del controlador difuso del motor DC M2 procedemos a obtener las reglas de correspondencia.

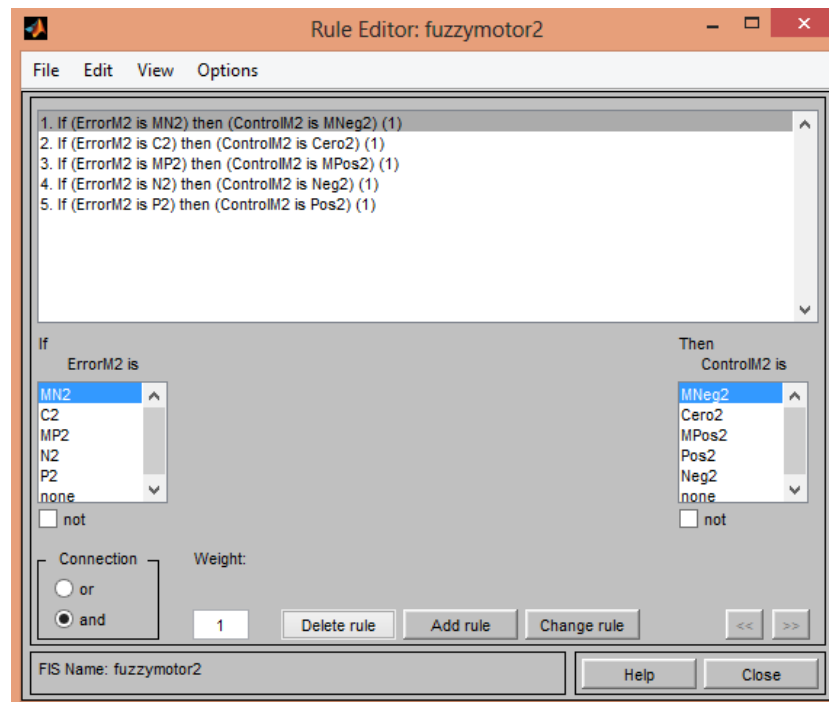


Figura 309. Reglas de correspondencia del motor DC M2

Con las reglas de correspondencia obtenidas procedemos a realizar el diagrama de bloques en Simulink en el cual importamos el archivo .fis donde se diseñó el controlador en el bloque llamado Fuzzy Logic Controller.

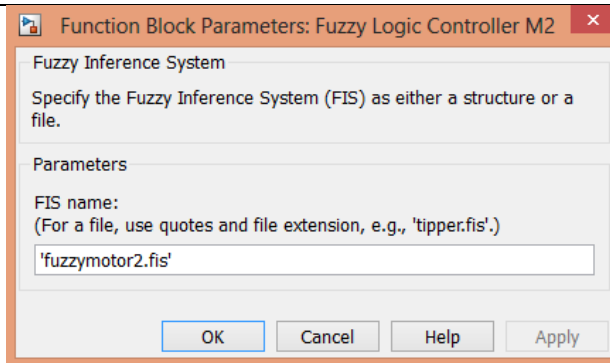


Figura 310. Archivo .fis del diseño del controlador Fuzzy del motor DC M2

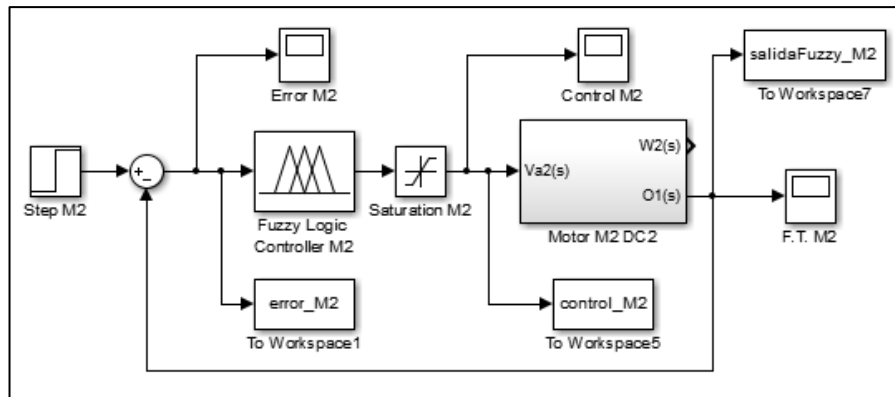


Figura 311. Diagrama del control Fuzzy del motor DC M2

Luego procedemos a obtener la función transferencia del motor DC M2 con el controlador difuso.

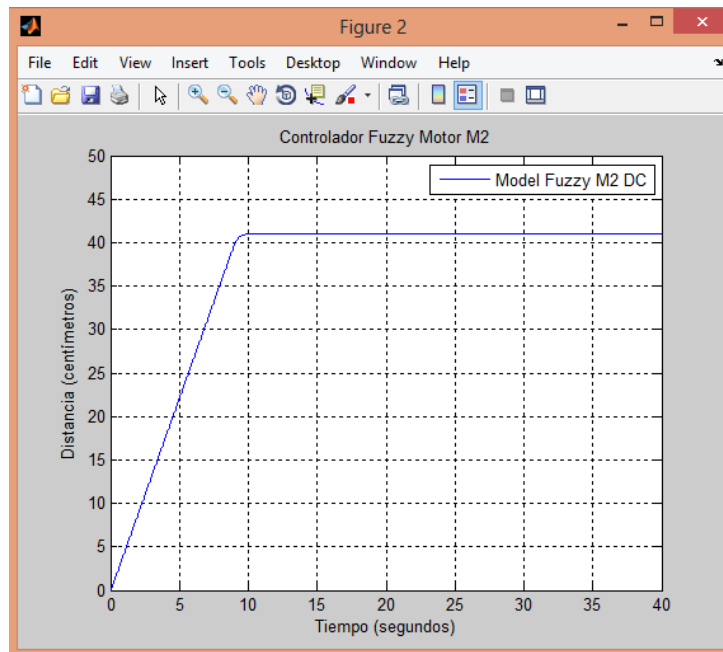



Figura 312. Función transferencia control Fuzzy del motor DC M2

Recomendaciones:

- Tener en cuenta el correcto diseño del controlador eligiendo correctamente los rangos de las funciones membresía de entrada y salida.
- Tener en consideración al momento de realizar las reglas de correspondencia ya que pueden ser descritas con varios operadores lógicos como AND y OR.

4.7. Práctica Nro. 7

		GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA: Ingeniería Electrónica		ASIGNATURA:
NRO. PRÁCTICA:	7	TÍTULO PRÁCTICA: Diseño de un controlador difuso para el movimiento horizontal.
Objetivos: <ul style="list-style-type: none"> • Diseñar el controlador Difuso para el motor DC M1 (eje horizontal). • Obtener las funciones de membresía de entrada y salida del controlador Difuso para el motor DC M1 (eje horizontal). • Obtener las reglas de correspondencia del controlador Difuso para el motor DC M1 (eje horizontal). • Obtener la gráfica de la función transferencia del motor DC M1 en lazo cerrado con el controlador Difuso. 		
INSTRUCCIONES:		1. Creación del controlador Difuso en la herramienta Fuzzy en el programa Matlab.
		2. Obtención de las funciones de membresía de entrada y salida con sus respectivas reglas de correspondencias.
		3. Simulación de la función transferencia con el controlador Difuso diseñado en Simulink.

ACTIVIDADES POR DESARROLLAR

Abrimos el Fuzzy en la ventana de comando de Matlab.

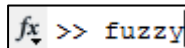


Figura 313. Función Fuzzy en la práctica 7

Una vez abierta la herramienta creamos la entrada ErrorM1 con rango de -60 hasta 60 y creamos la salida ControlM2 con rango de -29 hasta 29.

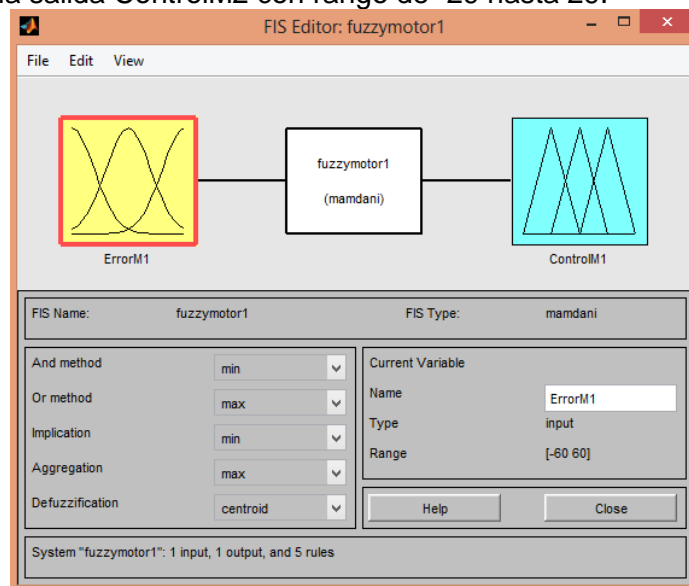


Figura 314. Creación de la entrada para el controlador difuso del motor DC M1

En la entrada creamos cinco funciones de membresía las cuales se llaman MN, N, C, P, MP para el controlador Fuzzy.

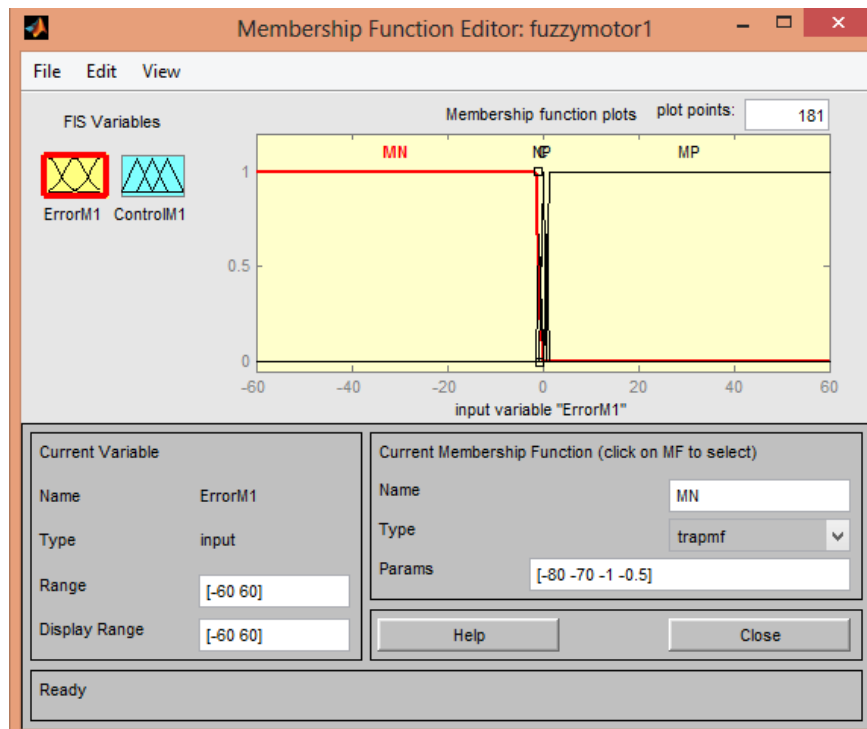


Figura 315. Función de membresía MN del motor DC M1

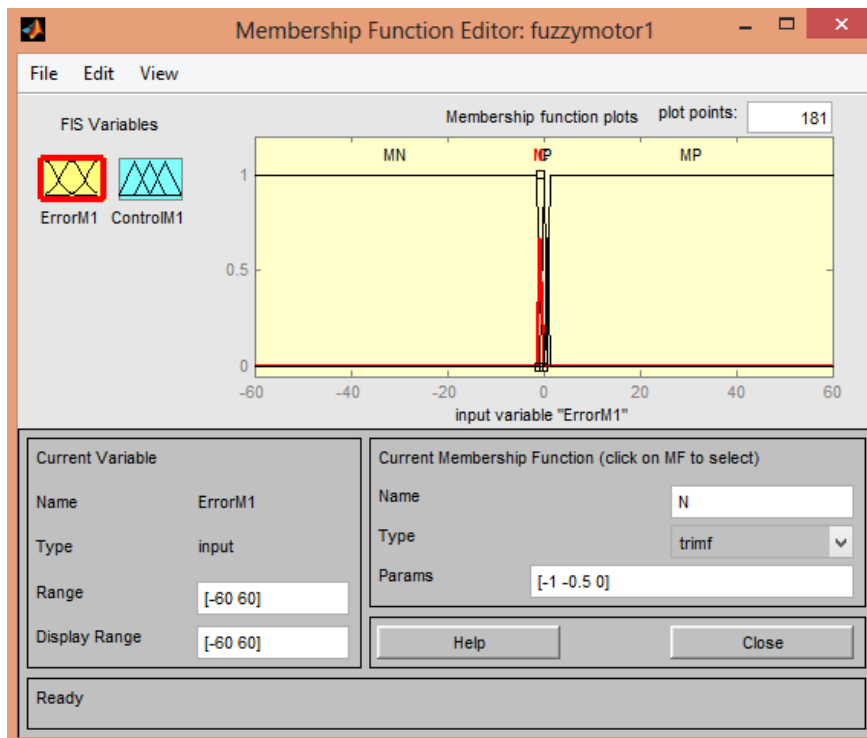


Figura 316. Función de membresía N del DC M1

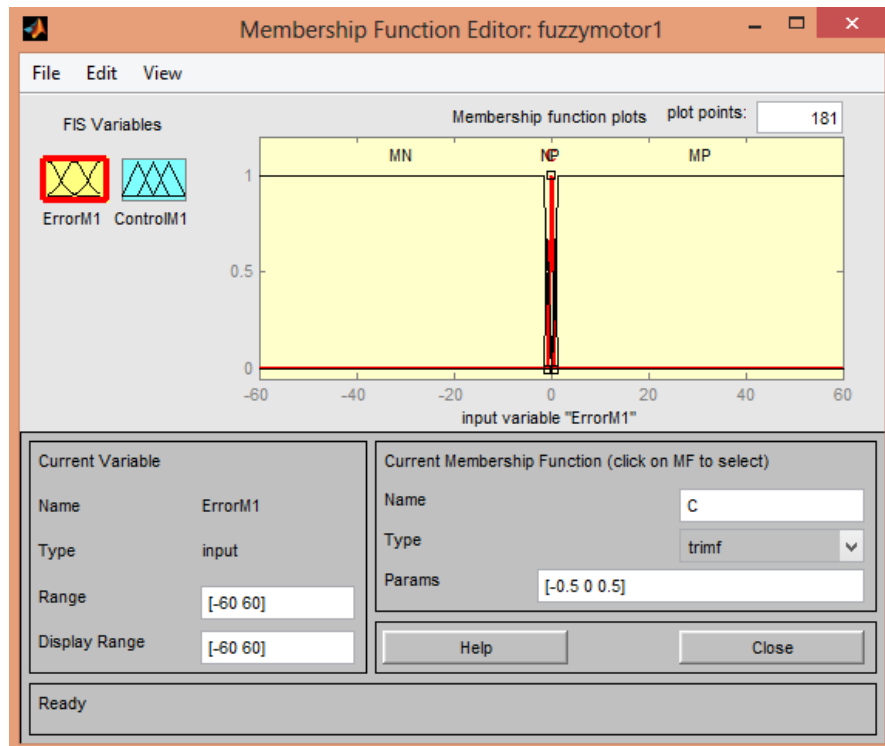


Figura 317. Función de membresía C del DC M1

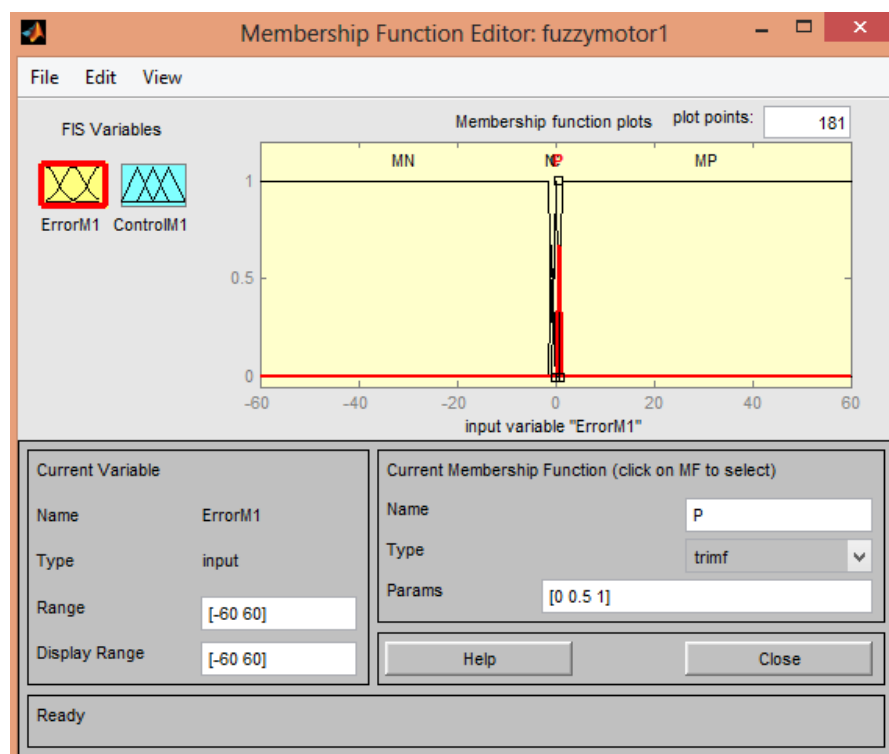


Figura 318. Función de membresía P del DC M1

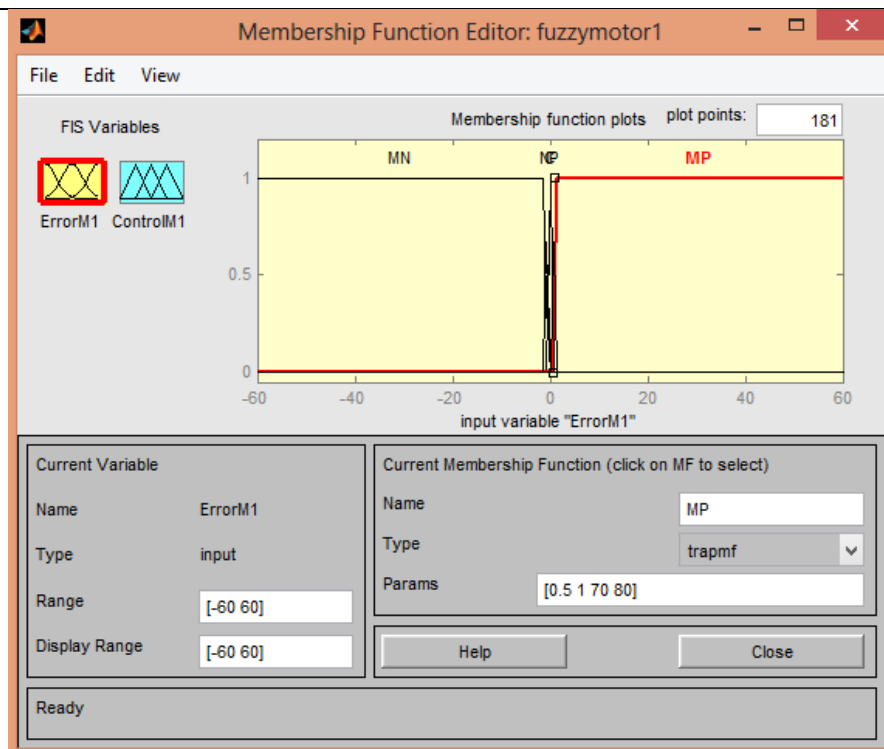


Figura 319. Función de membresía MP del motor DC M1

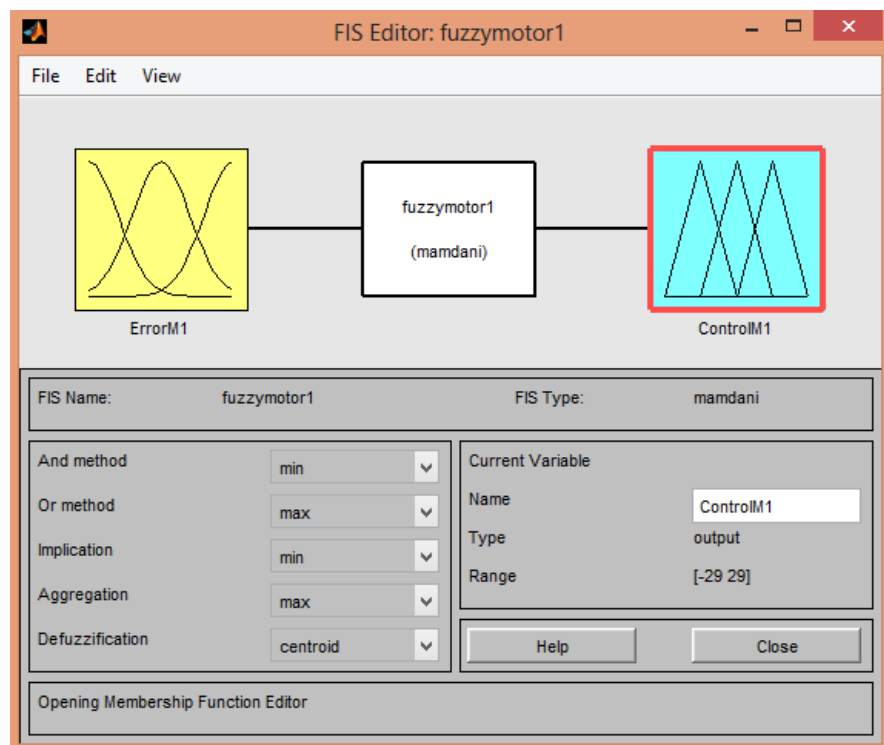


Figura 320. Creación de la salida para el controlador difuso del motor DC M1

Luego nos dirigimos a la salida donde creamos las funciones de membresía correspondientes las cuales se llaman MNeg, Neg, Cero, Pos, MPos para el controlador Fuzzy.

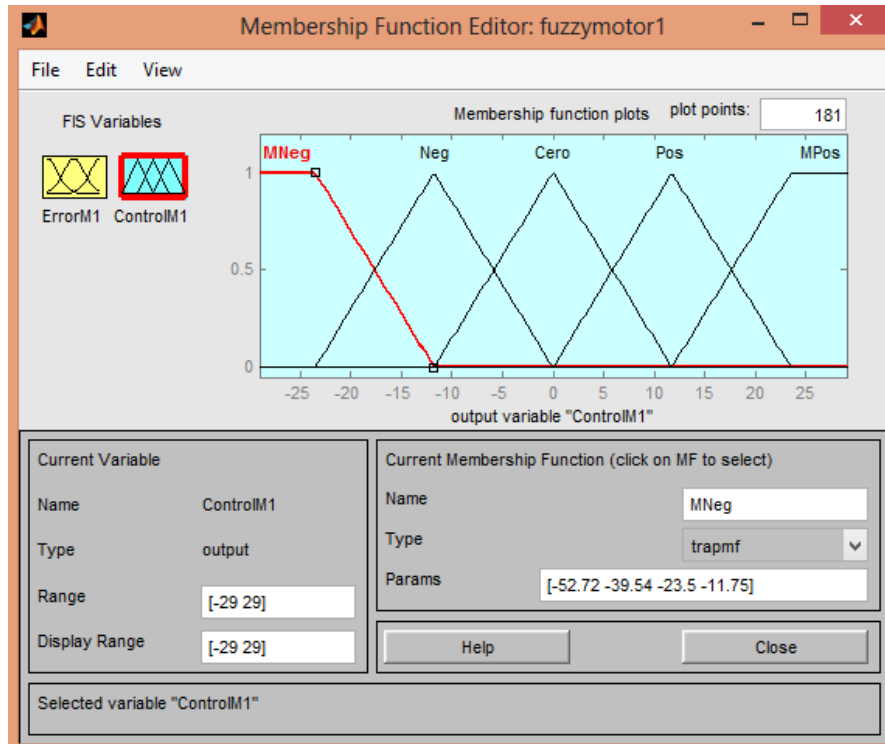


Figura 321. Función de membresía MNeg del motor DC M1

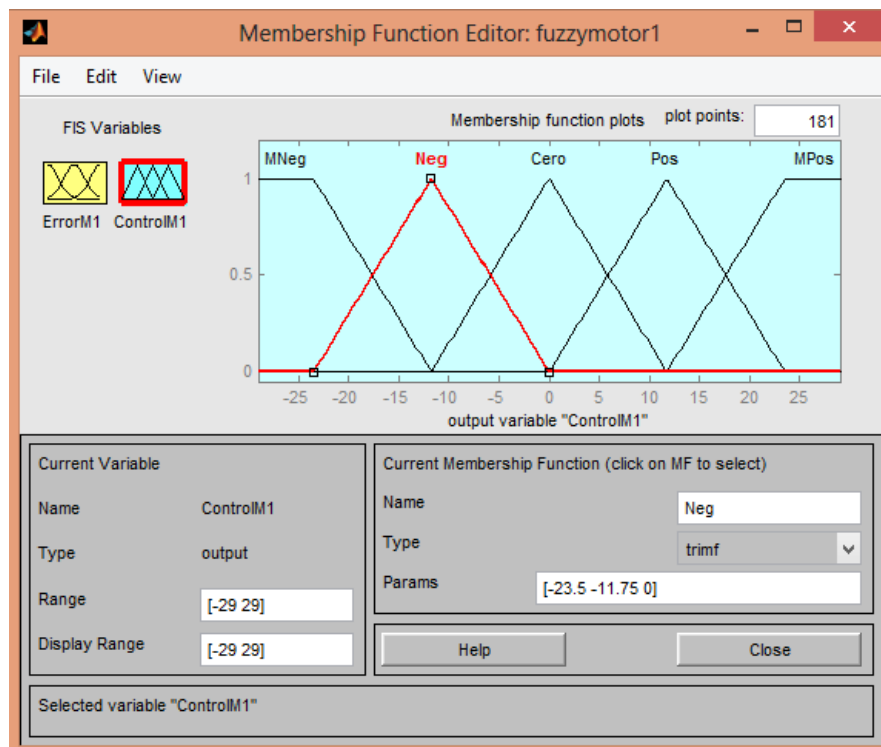


Figura 322. Función de membresía Neg del motor DC M1

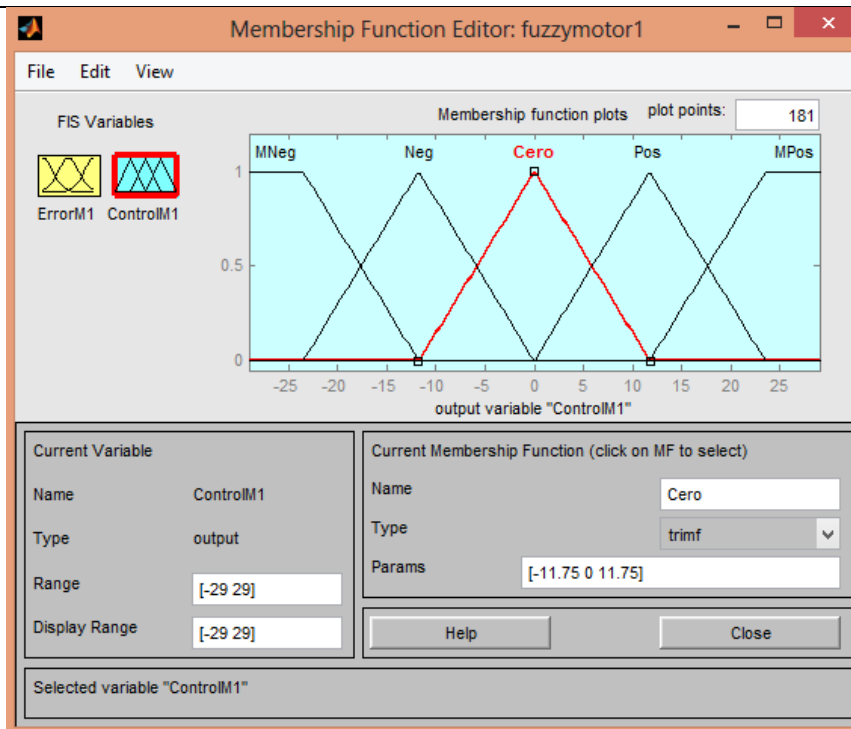


Figura 323. Función de membresía Cero del motor DC M1

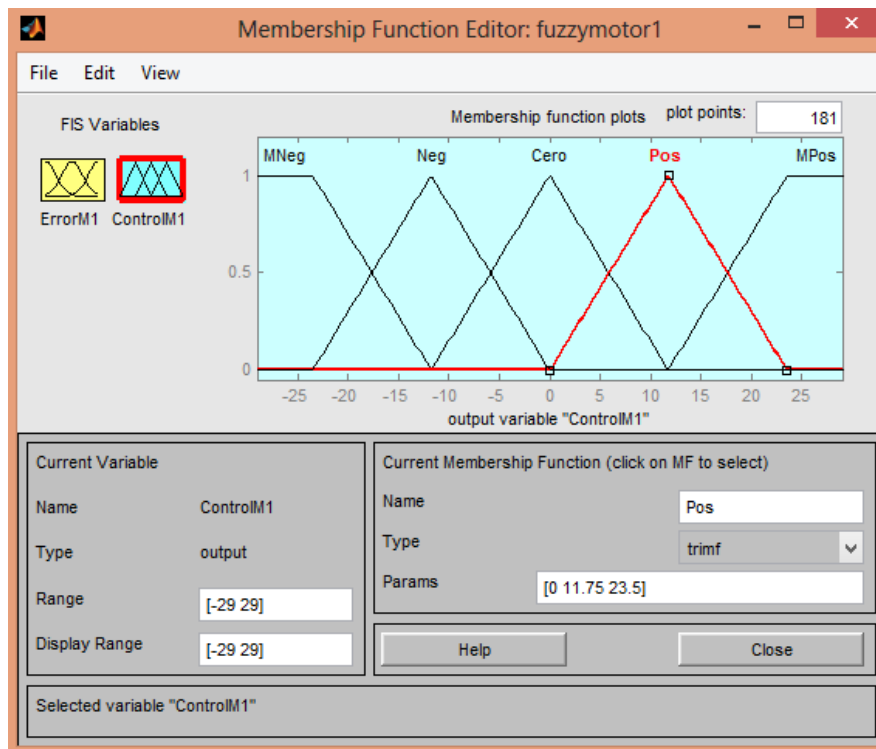


Figura 324. Función de membresía Pos del motor DC M1

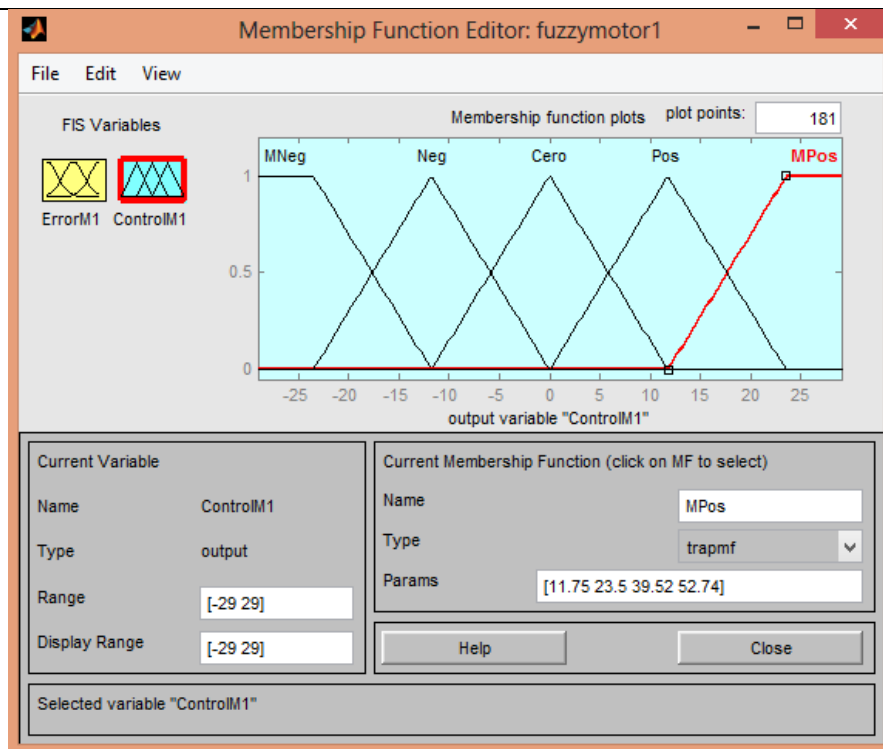


Figura 325. Función de membresía MPos del motor DC M1

Una vez obtenido la entrada y la salida del controlador difuso del motor DC M1 procedemos a obtener las reglas de correspondencia.

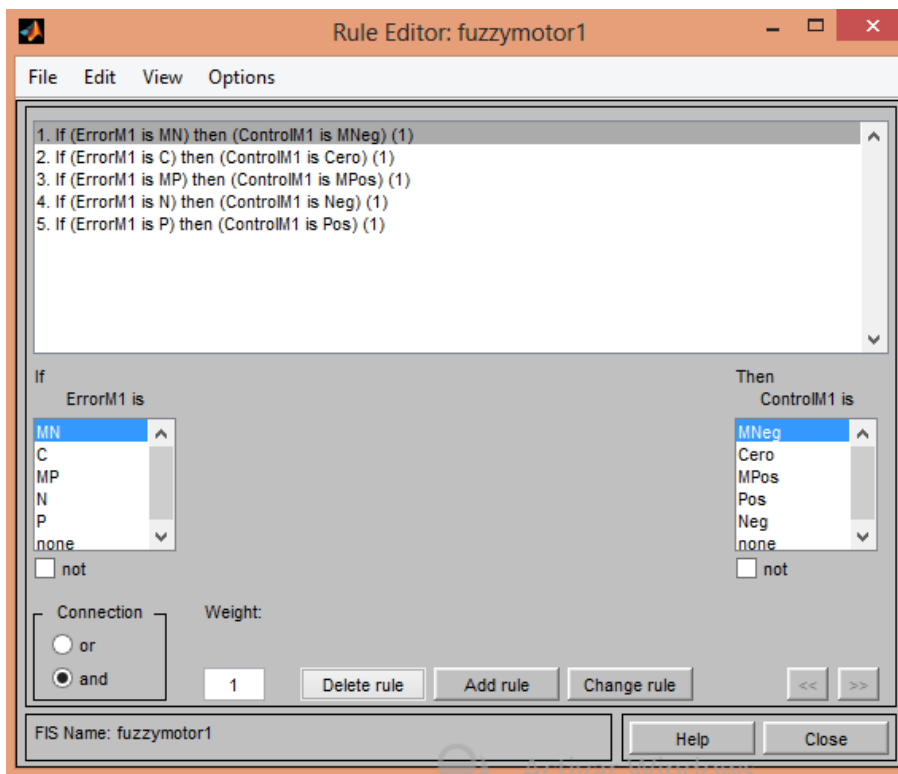


Figura 326. Reglas de correspondencia del controlador difuso del motor DC M1

Con las reglas de correspondencia obtenidas procedemos a realizar el diagrama de bloques en Simulink en el cual importamos el archivo .fis donde se diseñó el controlador en el bloque llamado Fuzzy Logic Controller.

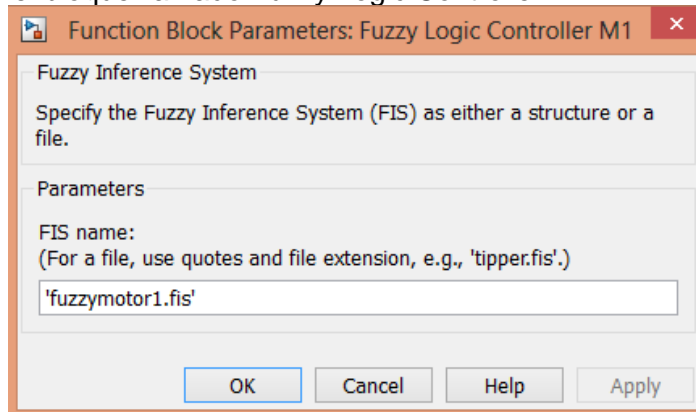


Figura 327. Archivo .fis del motor DC M1.

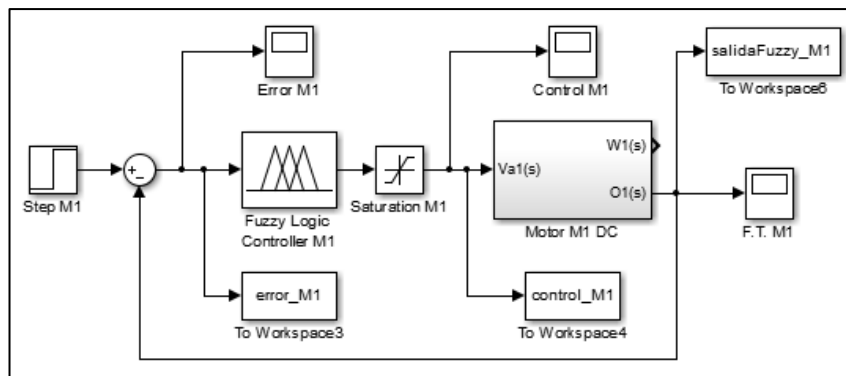


Figura 328. Diagrama de bloques del control Fuzzy del motor DC M1

Luego procedemos a obtener la función transferencia del motor DC M1 con el controlador difuso.

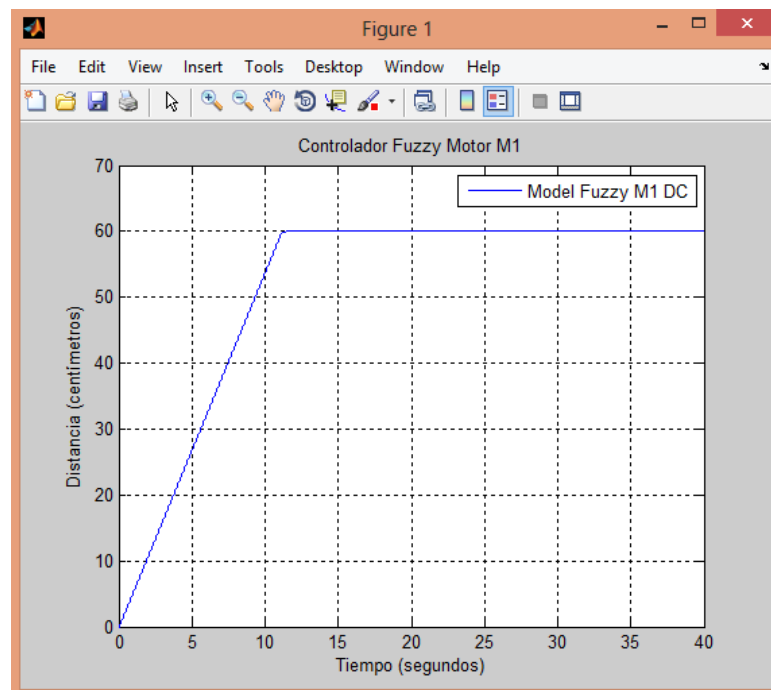

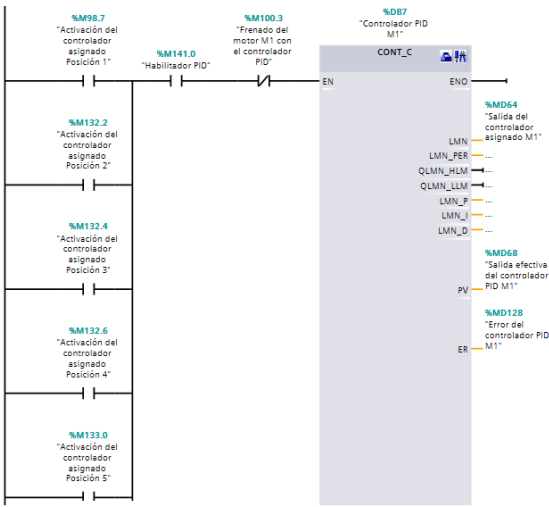


Figura 329. Función transferencia del motor DC M1

Recomendaciones:

- Tener en cuenta el correcto diseño del controlador eligiendo correctamente los rangos de las funciones membresía de entrada y salida.
- Tener en consideración al momento de realizar las reglas de correspondencia ya que pueden ser descritas con varios operadores lógicos como AND y OR.

4.8. Práctica Nro. 8

		GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA: Ingeniería Electrónica		ASIGNATURA:
NRO. PRÁCTICA:	8	TÍTULO PRÁCTICA: Diseño de aplicación HMI para el almacenamiento automático de objetos utilizando control PID y Difuso.
Objetivos: <ul style="list-style-type: none"> • Diseñar un programa para el almacenamiento de automático de piezas del sistema Festo MPS-500. • Diseñar una aplicación HMI para la visualización del almacenamiento automático del sistema Festo MPS-500. 		
INSTRUCCIONES:		1. Creación de los controladores PID y Difuso para el almacenamiento automático en el programa Tia Portal.
		2. Creación de la subred para la comunicación entre PLC S7-314C-2DP y WinCC RT Advanced.
		3. Creación de la aplicación HMI para la visualización de la planta en el programa Tia Portal.
ACTIVIDADES POR DESARROLLAR		
<p>Realizamos los bloques de los controladores, primero creamos el controlador PID con el bloque CONT_C donde procedemos a habilitar la parte proporcional, integral y derivativa con sus respectivos valores.</p> <p>Verificar el anexo 6 para más detalles.</p>		
		
Figura 330. Bloque de función PID del motor M1 inicio		

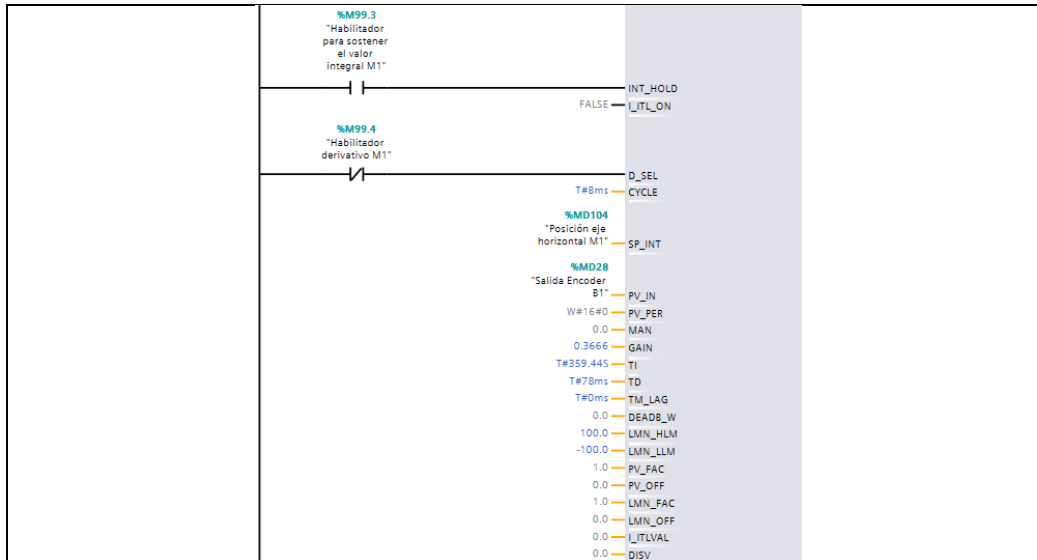


Figura 331. Bloque de función PID del motor M1 final

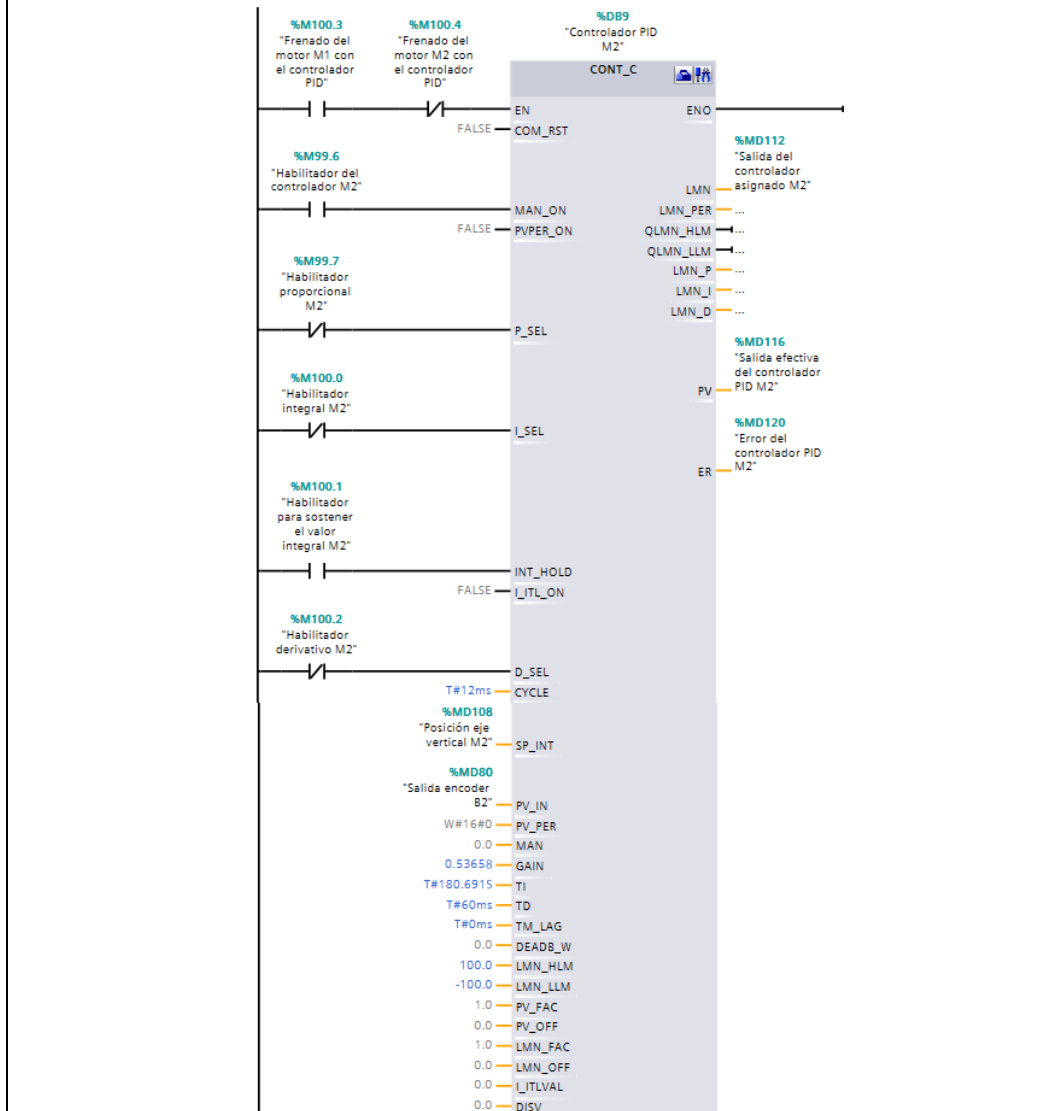


Figura 332. Bloque de función CONT_C para el controlador PID del motor M2

Después procedemos a crear el bloque Difuso, este bloque se lo realizo en un bloque de función y se lo diseñó por medio de programación SCL. Verificar para más detalles el anexo 6.

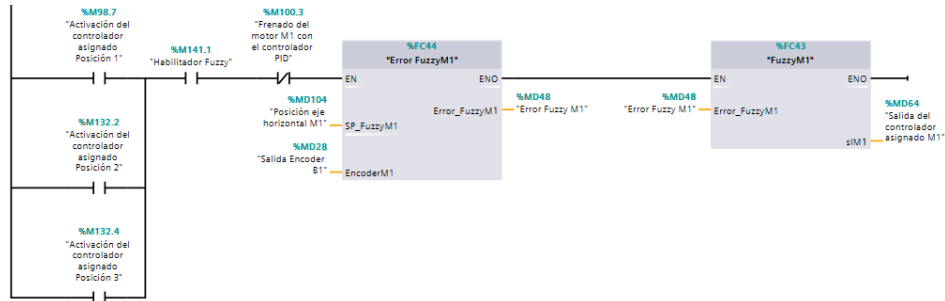


Figura 333. Error Fuzzy M1 y el controlador Fuzzy M1

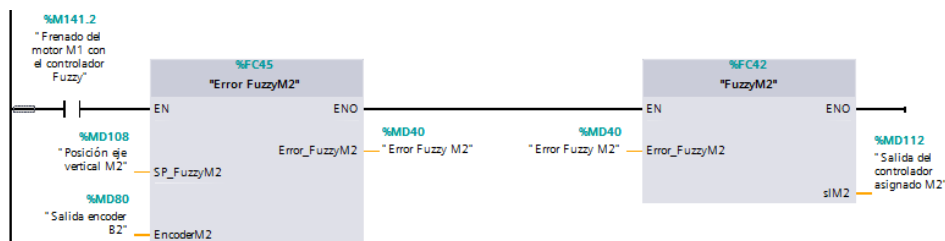


Figura 334. Error Fuzzy M2 y el controlador Fuzzy M2

Para la creación de la aplicación HMI procedemos a agregar dispositivo, en la parte de sistemas de PC seleccionamos SIMATIC HMI Application y elegimos WinCC RT Advanced.

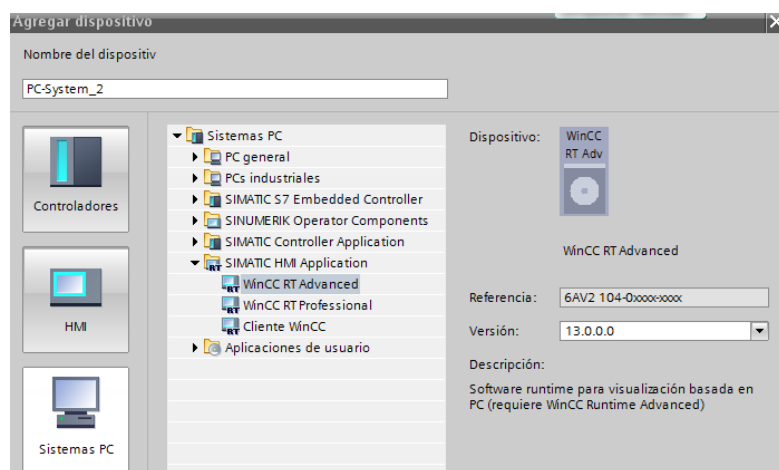


Figura 335. Elección del sistema PC para la aplicación HMI

Para realizar la comunicación procedemos a insertar un módulo de comunicación Profinet.

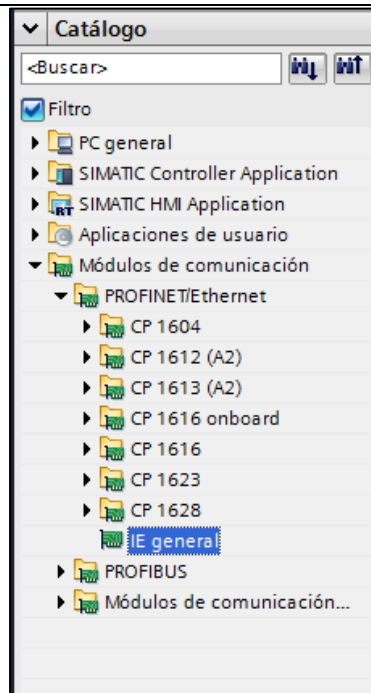


Figura 336. Módulo de comunicación Profinet a seleccionar

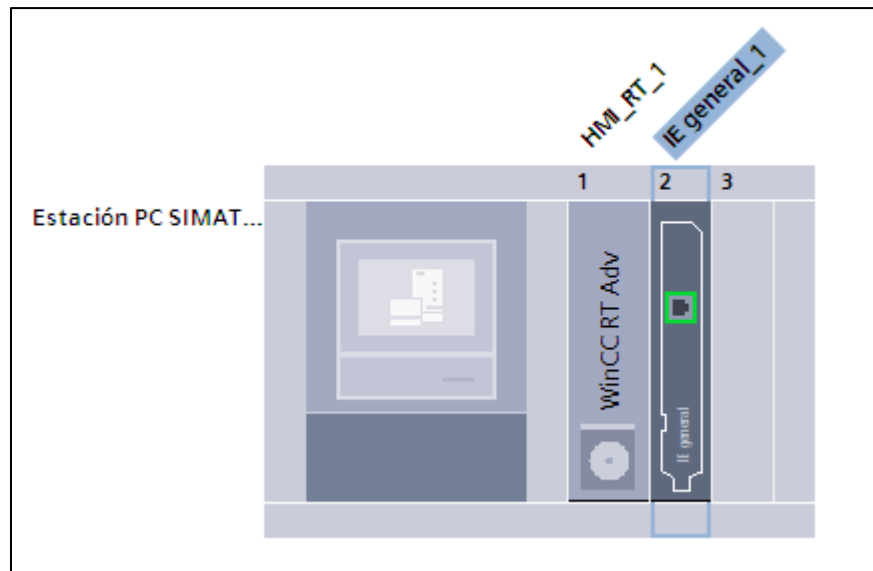


Figura 337. Módulo de comunicación insertado en el sistema PC

Luego nos dirigimos a la parte de topología de redes y vemos que direcciones tiene nuestros módulos de comunicación, en esta parte tenemos que configurar las direcciones para que se encuentren en la misma red, el módulo de comunicación insertado debe tener la dirección de la PC en uso.

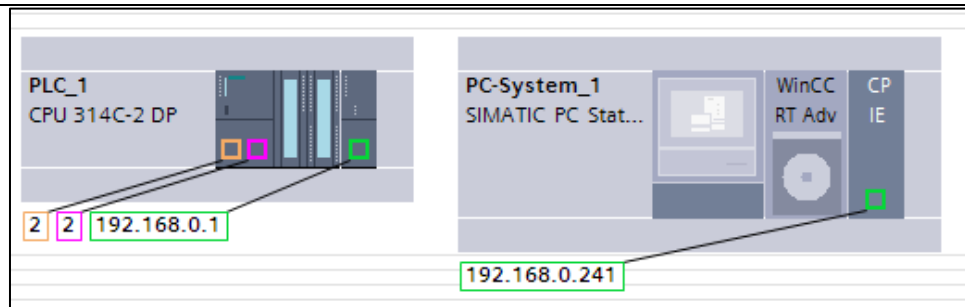


Figura 338. Direcciones de red de cada módulo

Una vez con las direcciones correctas para cada módulo establecemos la conexión HMI y la sincronización entre los dos dispositivos.

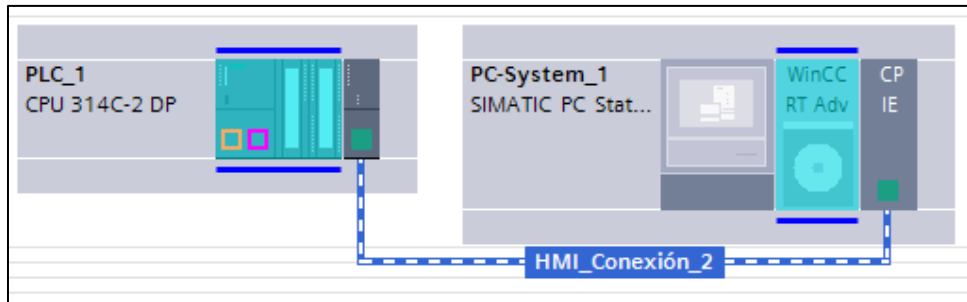


Figura 339. Conexión y sincronización de los dispositivos.

Una vez realizado la conexión creamos una imagen dentro del árbol de proyecto del WinCC RT Advanced donde procedemos a crear nuestra aplicación.

Para configurar los botones seleccionamos dicho botón, nos vamos a propiedades y nos dirigimos a eventos donde seleccionamos pulsar, en esa opción seleccionamos la función ActivarBit y colocamos la variable correspondiente.

Para desactivar el botón seleccionamos soltar, en esa opción seleccionamos la función DesactivarBit y colocamos la misma variable seleccionada. Como se observa en la figura 341.

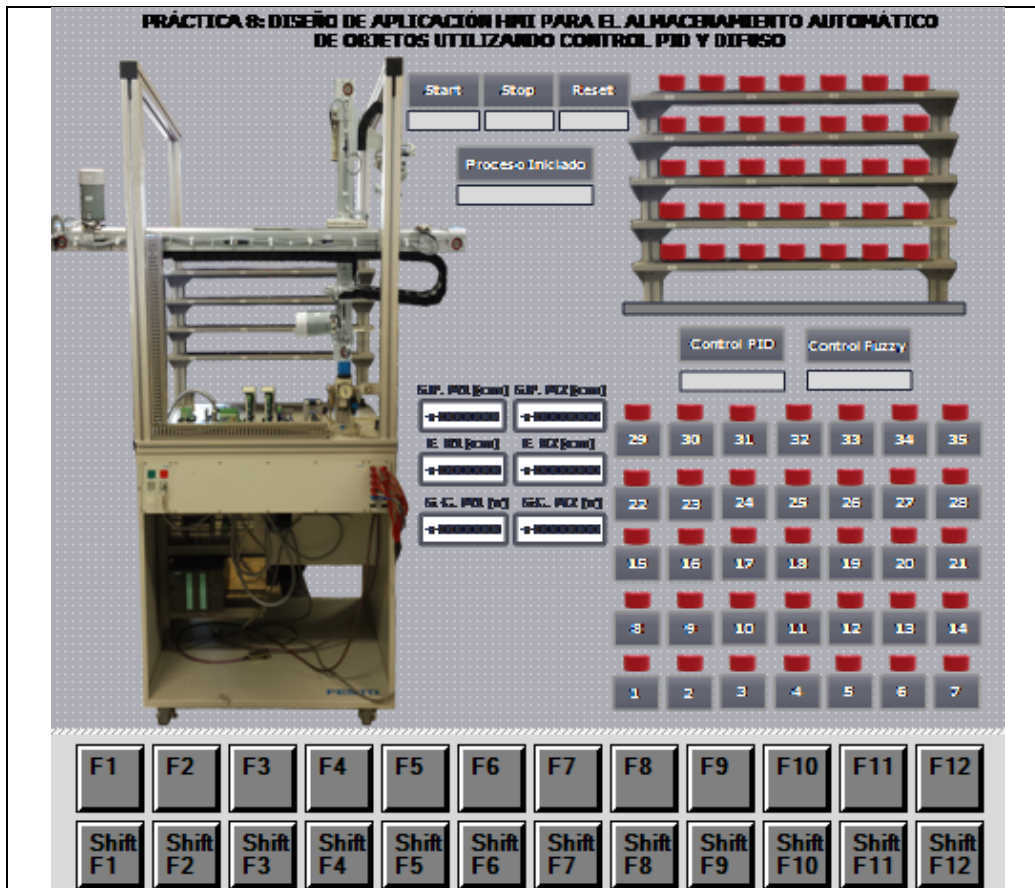


Figura 340. Creación de la aplicación HMI en la práctica 8

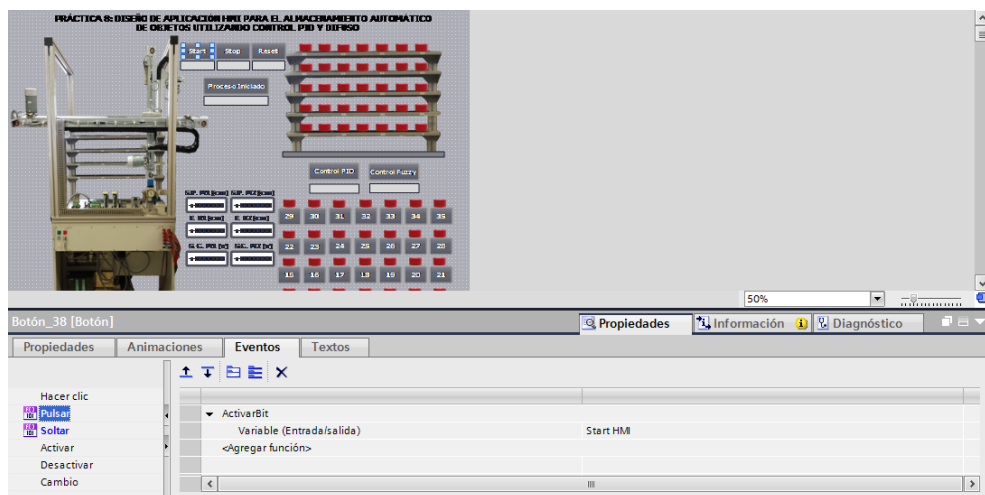


Figura 341. Configuración pulsar del pulsador.

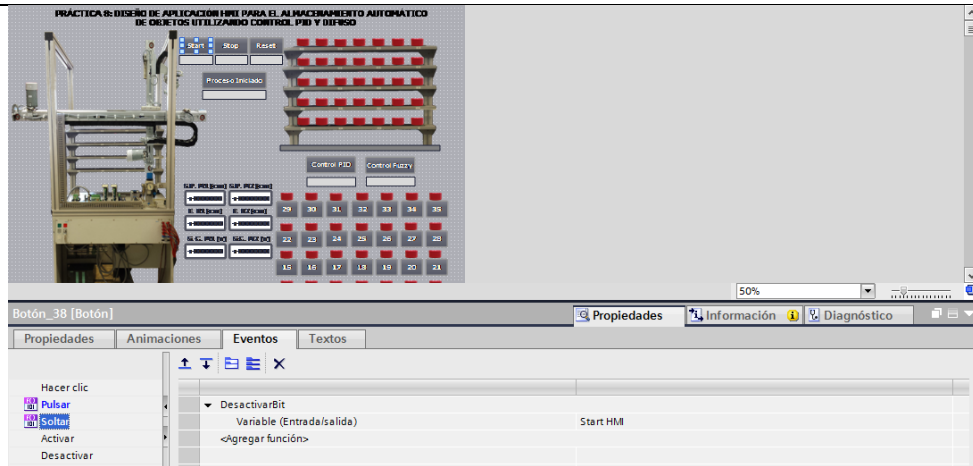


Figura 342. Configuración soltar del pulsador

Para configurar los indicadores de luz seleccionamos dicho indicador, nos vamos a propiedades y nos dirigimos a animaciones donde seleccionamos apariencia, en esa opción seleccionamos la variable correspondiente.

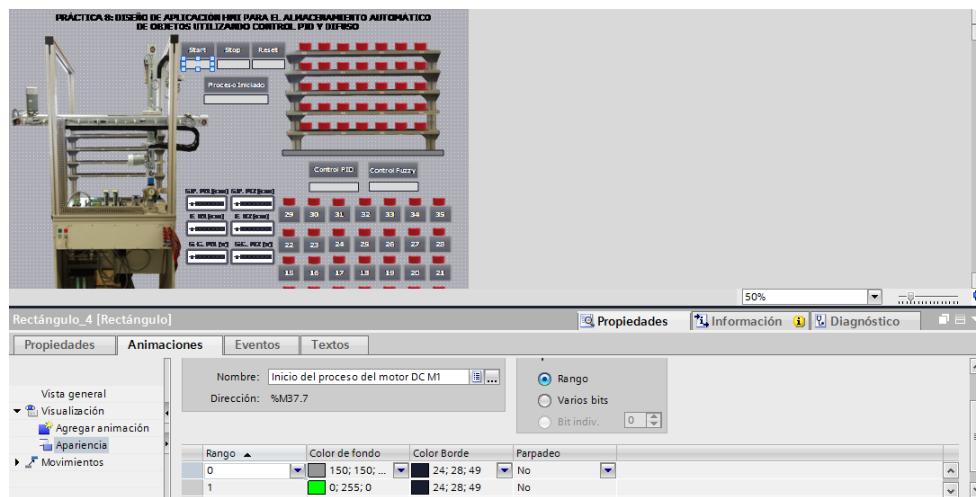


Figura 343. Configuración del indicador de luz

Para configurar los indicadores de la ficha a elegir nos vamos a propiedades, nos dirigimos a animaciones donde seleccionamos visibilidad, en esa opción seleccionamos la variable correspondiente y seleccionamos la opción invisible.

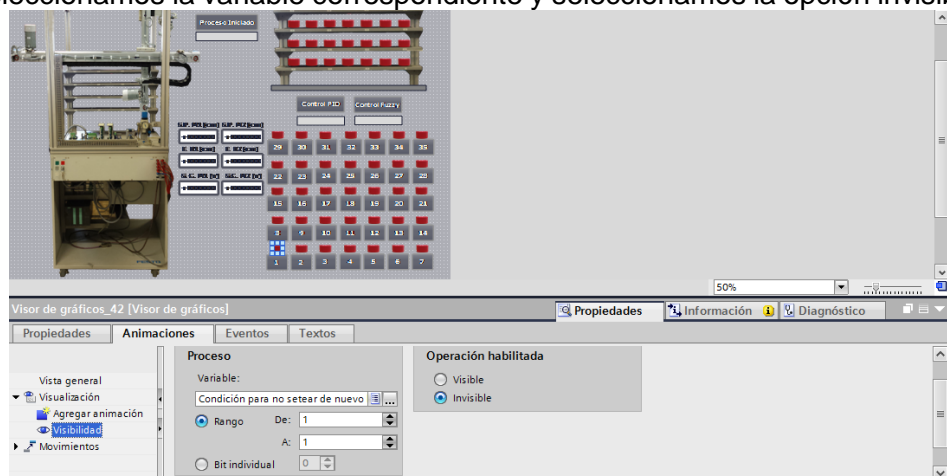


Figura 344. Configuración del indicador de la ficha

Para configurar los indicadores de la ficha nos vamos a propiedades y nos dirigimos a animaciones donde seleccionamos visibilidad, en esa opción seleccionamos la variable correspondiente y seleccionamos la opción visible.

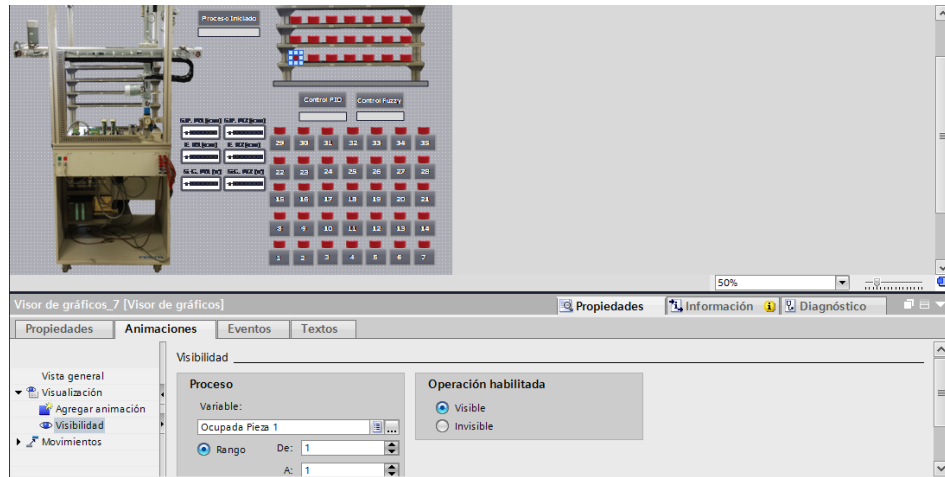


Figura 345. Configuración del indicador de la ficha en el estante

Para le movimiento del motor M1 seleccionamos la gráfica, nos vamos a propiedades y nos dirigimos a animaciones donde seleccionamos movimiento directo y colocamos la variable correspondiente.

Luego nos dirigimos a todas las variables del HMI y seleccionamos la variable seleccionada, vamos a propiedades, seleccionamos escalado lineal, activamos la opción, colocamos el punto final, punto inicial en el PLC y en el HMI.

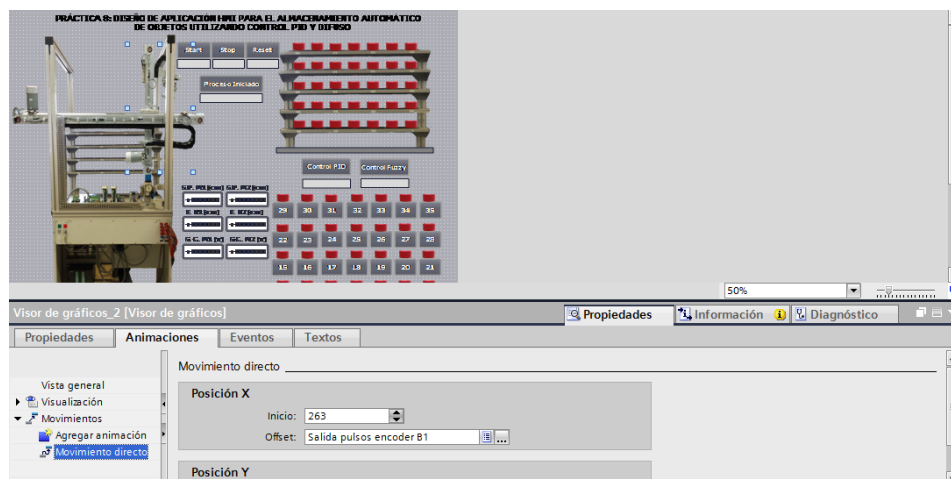


Figura 346. Configuración de la variable para el movimiento eje X del motor M1

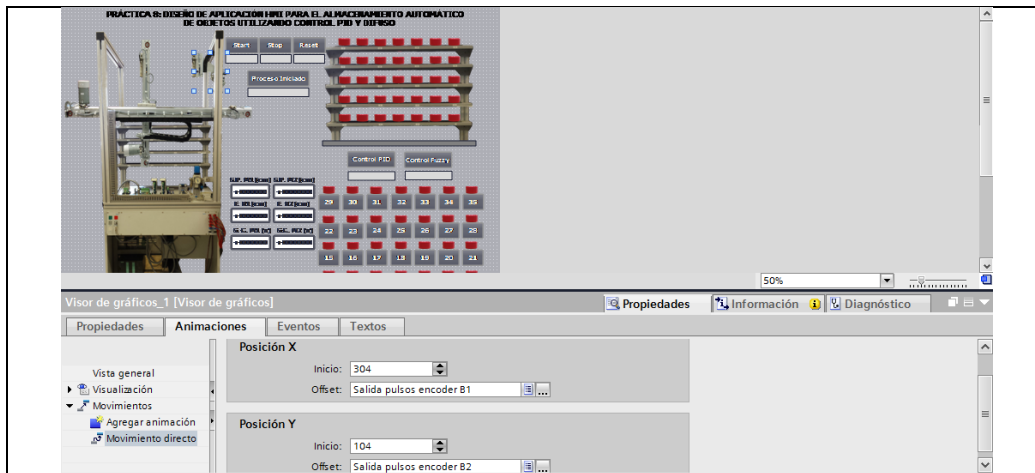


Figura 347. Configuración de los indicadores numéricos

Tabla de variables estándar					
Nombre	de acceso	Ciclo de adquisi..	Archi...	SourceComment	
Proceso Iniciado	eso absoluto>	100 ms	<input type="checkbox"/>		
Salida pulsos encoder B1	eso absoluto>	100 ms	<input type="checkbox"/>		
Salida pulsos encoder B2	eso absoluto>	100 ms	<input type="checkbox"/>		
Start	eso absoluto>	100 ms	<input type="checkbox"/>		
Start HMI	eso absoluto>	100 ms	<input type="checkbox"/>		
Stop	eso absoluto>	100 ms	<input type="checkbox"/>		
Sensor Derecha B10	eso absoluto>	100 ms	<input type="checkbox"/>		
Sensor Arriba B12	eso absoluto>	100 ms	<input type="checkbox"/>		
Stop HMI	eso absoluto>	100 ms	<input type="checkbox"/>		
Stop HMI luz	eso absoluto>	100 ms	<input type="checkbox"/>		
<Agregar>					

Parámetro Variable HMI			
Salida pulsos encoder B1 [Variable_HMI]			
Propiedades			
Escalado lineal <input checked="" type="checkbox"/>			
PLC	Valor final:	35000	HMI
	Valor inicial:	-2000	Valor final:
			Valor inicial:
			0

Figura 348. Tabla de variables de la aplicación HMI en Tia Portal

Recomendaciones:

- Tener en cuenta el correcto direccionamiento del módulo de comunicación del PLC y de la PC virtual para su correcto funcionamiento.
- Tener en consideración la correcta animación de los objetos en la aplicación HMI para observar la correcta visualización de la planta.

5. ANÁLISIS DE RESULTADOS

Se realizó la comparación de las características de los controladores PID versus los controladores Fuzzy de los motores DC M1 y M2.

Para el controlador P versus Fuzzy del motor DC M1, se pudo observar en los datos reales que el controlador P no alcanza el Set Point deseado dejando una diferencia de 4.55 centímetros, a diferencia del controlador Fuzzy que si alcanza el Set Point deseado y también teniendo menos tiempo de asentamiento con respecto al controlador P.

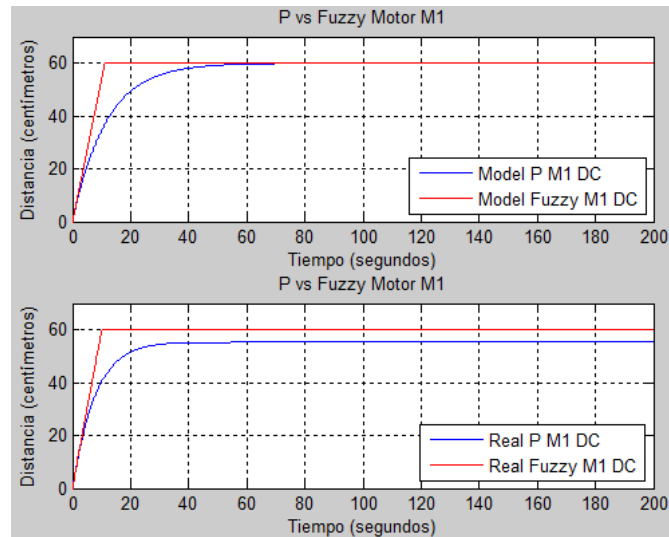


Figura 349. Comparación del P vs fuzzy del motor DC M1

Para el controlador PD versus Fuzzy del motor DC M1, se pudo observar en los datos reales que el controlador PD no alcanza el Set Point deseado dejando una diferencia de 4.55 centímetros, a diferencia del controlador Fuzzy que si alcanza el Set Point deseado y también teniendo menos tiempo de asentamiento con respecto al controlador PD.

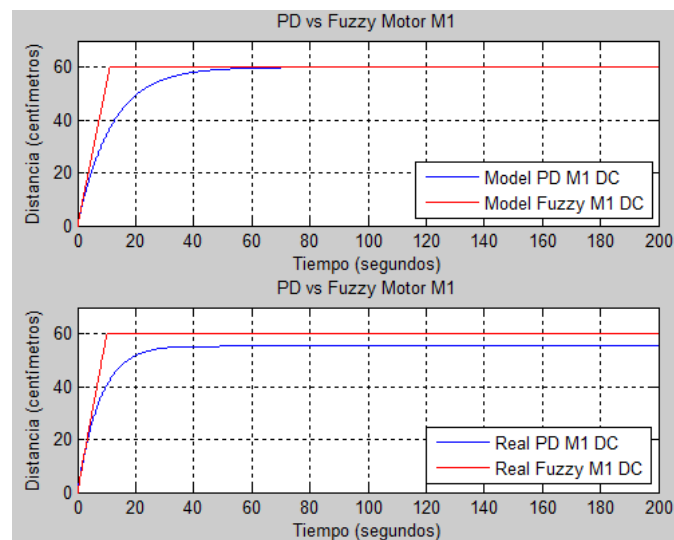


Figura 350. Comparación del PD vs fuzzy del motor DC M1

Para el controlador PI versus Fuzzy del motor DC M1, se pudo observar en los datos reales que el controlador PI tarda en alcanzar el Set Point deseado, a diferencia del controlador Fuzzy que si alcanza el Set Point deseado y también teniendo menos tiempo de asentamiento con respecto al controlador PI.

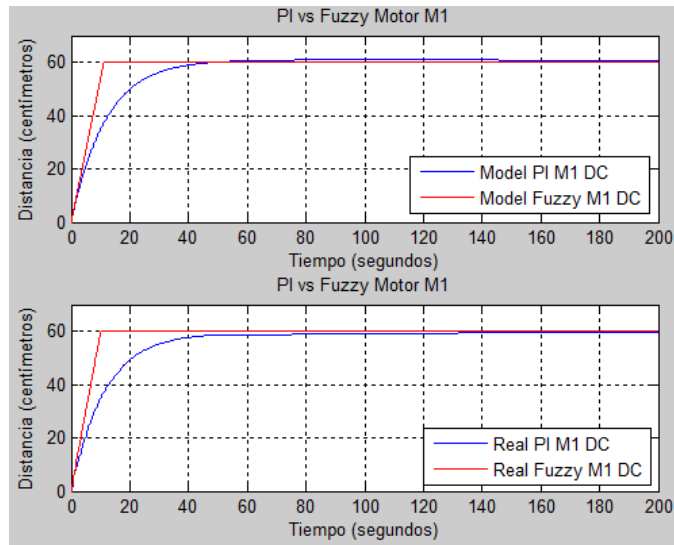


Figura 351. Comparación del PI vs fuzzy del motor DC M1

Para el controlador PID versus Fuzzy del motor DC M1, se pudo observar en los datos reales que el controlador PID tarda en alcanzar el Set Point deseado, a diferencia del controlador Fuzzy que si alcanza el Set Point deseado y también teniendo menos tiempo de asentamiento con respecto al controlador PID.

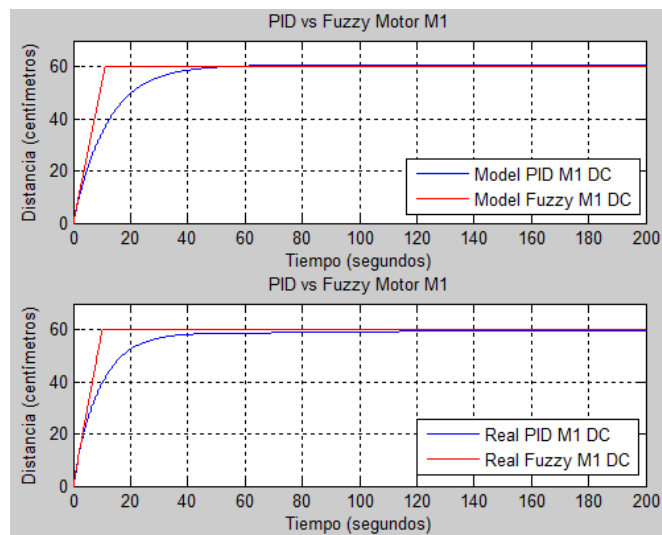


Figura 352. Comparación del PID vs fuzzy del motor DC M1

Para el controlador P versus Fuzzy del motor DC M2, se pudo observar en los datos reales que el controlador P no alcanza el Set Point deseado dejando una diferencia de 2.28 centímetros, a diferencia del controlador Fuzzy que si alcanza el Set Point

deseado y también teniendo menos tiempo de asentamiento con respecto al controlador P.

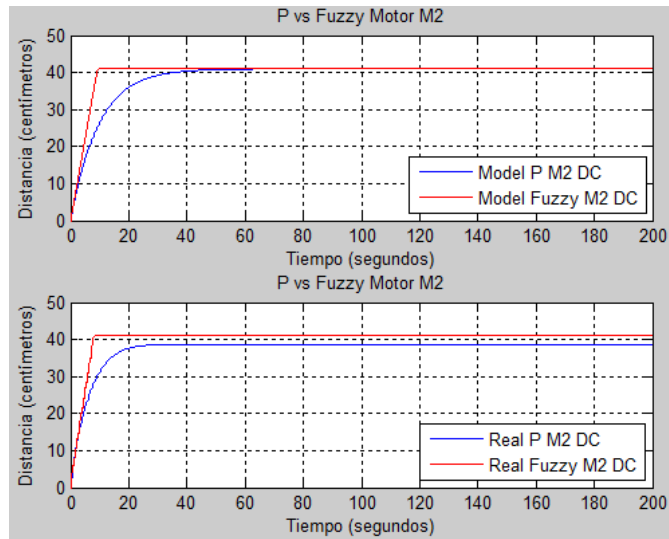


Figura 353. Comparación del P vs fuzzy del motor DC M2

Para el controlador PD versus Fuzzy del motor DC M2, se pudo observar en los datos reales que el controlador PD no alcanza el Set Point deseado dejando una diferencia de 2.28 centímetros, a diferencia del controlador Fuzzy que si alcanza el Set Point deseado y también teniendo menos tiempo de asentamiento con respecto al controlador PD.

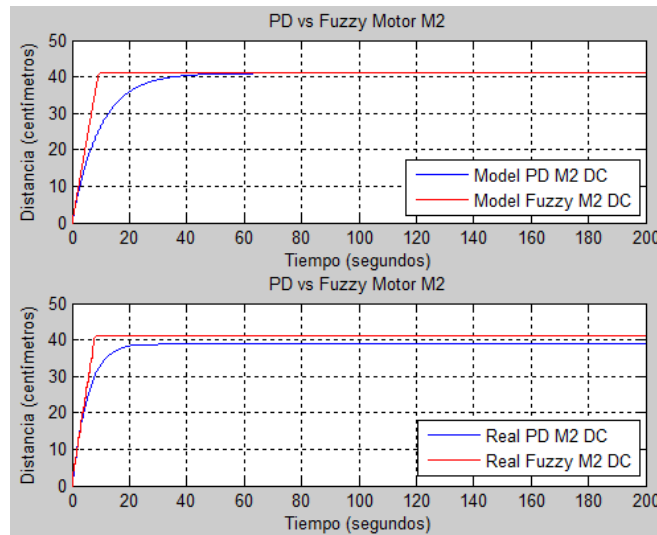


Figura 354. Comparación del PD vs fuzzy del motor DC M2

Para el controlador PI versus Fuzzy del motor DC M2, se pudo observar en los datos reales que el controlador PI tarda en alcanzar el Set Point deseado, a diferencia del controlador Fuzzy que si alcanza el Set Point deseado y también teniendo menos tiempo de asentamiento con respecto al controlador PI.

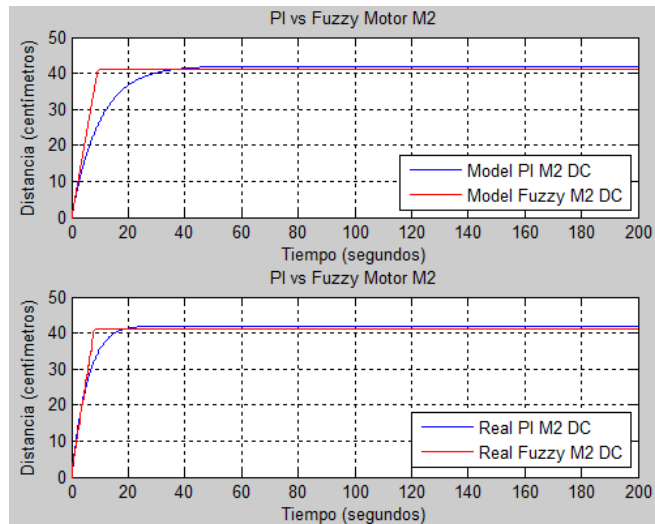


Figura 355. Comparación del PI vs fuzzy del motor DC M2

Para el controlador PID versus Fuzzy del motor DC M2, se pudo observar en los datos reales que el controlador PID tarda en alcanzar el Set Point deseado, a diferencia del controlador Fuzzy que si alcanza el Set Point deseado y también teniendo menos tiempo de asentamiento con respecto al controlador PID.

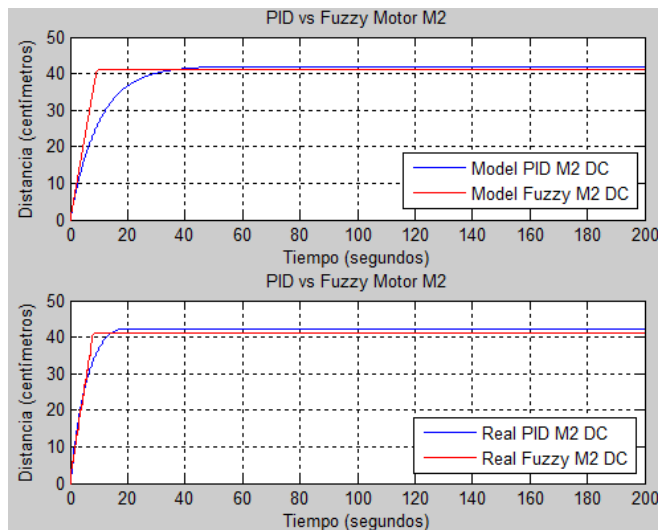


Figura 356. Comparación del PID vs fuzzy del motor DC M2

6. CONCLUSIONES Y RECOMENDACIONES

Conclusiones

- Se pudo corroborar que no se disponía de las funciones de transferencia de los motores DC M1 y M2 de manera experimental, por lo tanto se obtuvo los parámetros experimentales de ambos motores y así se obtuvo las funciones de transferencias experimentales de ambos motores.
- Se pudo constatar el tipo de control de los motores DC M1 y M2 el cual era on/off, por lo cual se diseñó e implementó el controlador PID y Fuzzy para los motores DC M1 y M2.
- Se pudo comparar las gráficas de las características de los controladores PID vs las gráficas de las características de los controladores Fuzzy, se pudo notar que el controlador fuzzy es más efectivo respecto al controlador PID, llegando a tener menos tiempo de asentamiento los motores DC M1 y M2 con el controlador fuzzy.
- La planta al no tener un sistema SCADA, se implementó un visualizador de datos en tiempo real (SCADA) en el software LabView, el cual permitió monitorear todos los datos referentes a la planta.
- La planta no contaba con una guía para el uso correcto de la misma, por lo tanto se elaboró un manual de prácticas para el correcto uso de la planta para los estudiantes de la carrera de ingeniería electrónica.
- Debido a que la planta dispone de las tarjetas DC-RS para el control on/off de los motores DC M1 y M2, se implementó la tarjeta Sabertooth 2x60 para el respectivo control PID y Fuzzy de ambos motores.

Recomendaciones

- Revisar detalladamente el reemplazo en las fórmulas de las funciones de transferencia de los motores y la correcta codificación en el software MatLab para generar las respectivas gráficas sin ningún tipo de error.
- Tomar en cuenta el correcto diseño de los controladores PID de los motores DC M1 y M2 por medio de la herramienta sisotool del software MatLab, eligiendo bien los ceros a añadir al compensador y de la misma forma tener en cuenta al momento de obtener los parámetros de los controladores PID.
- Tomar en cuenta el correcto diseño de los controladores Fuzzy de los motores DC M1 y M2 por medio de la herramienta fuzzy del software MatLab, eligiendo respectivamente los rangos de las funciones de membresía de entradas y salidas de ambos motores.
- Verificar detalladamente la codificación respectiva en MatLab en cuanto a las comparaciones del PID vs Fuzzy en ambos motores para obtener una mejor visualización de gráficas sin ningún tipo de error al momento de ejecutar.
- Revisar la correcta configuración de las variables de entradas y salidas para la visualización en tiempo real de la planta.
- Configurar de manera correcta la misma IP en el software KEPServerEX para el canal y el device, ya que deben ser las mismas IP de la laptop a trabajar y del módulo de comunicación del PLC respectivamente.
- Dejar bien fijado la planta en la que se implementó este proyecto para no generar ningún tipo de disturbio o incomodidad al momento de generar el proceso.
- Leer las instrucciones en la sección de las prácticas para poder llevar un excelente manejo al momento de ejecutar los procesos.
- Verificar en la planta que el breaker implementado este en modo OFF, ya que si está en modo ON podría estar en riesgo la estructura física de la planta.

7. REFERENCIAS BIBLIOGRÁFICAS

- PTC Inc. (2018). *KEPServerEX*. Obtenido de <https://www.kepware.com/en-us/products/kepserverex/>
- 8760 Engineering. (2018). *Linearizing PID Loop Control*. Obtenido de <http://www.8760engineering.com/blog/linearizing-pid-loop-control/>
- Cedillo, B. X. (2016). *ANÁLISIS DE ESTABILIDAD DE UN PÉNDULO INVERTIDO ROTATIVO UTILIZANDO TÉCNICAS DE CONTROL NO LINEAL*. Guayaquil: ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL.
- CONTROL TUTORIALS FOR MATLAB & SIMULINK. (s.f.). *DC Motor Position: System Modeling*. Obtenido de <http://ctms.engin.umich.edu/CTMS/index.php?example=MotorPosition§ion=SystemModeling>
- Dimension Engineering LLC. (2004). *Sabertooth dual 60A motor driver*. Obtenido de <https://www.dimensionengineering.com/products/sabertooth2x60>
- Eckart, W. (2002). *Mechatronics Teachware ASRS MPS 2000*. Denkendorf: Edición de Instructores.
- Elektroskandia Sverige AB. (2018). *CPU 314C-2DP 192KBYTE*. Obtenido de <https://www.elektroskandia.se/produkt?artnr=4500357>
- Festo. (2018). *Fuente de alimentación de sobremesa*. Obtenido de <http://www.festo-didactic.com/es-es/productos/mps-sistema-de-produccion-modular/accesorios/fuente-de-alimentacion-de-sobremesa.htm?fbid=ZXMuZXMuNTQ3LjE0LjE4LjcxMy4zNjM4>
- FOTRONIC CORPORATION. (2018). *Instek AFG-2112*. Obtenido de <http://www.testequipmentdepot.com/instek/signalgenerators/afg2112.htm>
- Intercambios Virtuales. (2015). *Mathworks Matlab R2013b, Software Matemático con Entorno Integrado*. Obtenido de <http://www.intercambiosvirtuales.org/software/mathworks-matlab-r2013a-software-matematico-con-entorno-integrado>
- INTOR. (2013). *Clasificación de válvulas neumáticas direccionales*. Obtenido de <http://www.intor.com.ar/clasificacion-de-valvulas-neumaticas-direccionales/>
- LEÓN PÉREZ, R. P. (2014). *DISEÑO, PROGRAMACIÓN E IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL DIFUSO PARA EL PROCESO DE PRODUCCIÓN DE HIDRÓGENO A PARTIR DE VAPOR SOBRECALENTADO DE AGUA*. Quito: ESCUELA POLITÉCNICA NACIONAL.
- Mazzone, V. (2002). *Controladores PID*. Quilmes: Universidad Nacional de Quilmes.
- National Instruments Corporation. (2018). *NI ELVIS II*. Obtenido de <http://www.ni.com/es-cr/support/model.ni-elvis-ii.html>
- National Instruments Corporation. (2018). *Software de Desarrollo de Sistemas NI LabVIEW*. Obtenido de <http://www.ni.com/labview/products/esa/>

- NOEGA System. (5 de MAYO de 2016). *SISTEMAS DE ALMACENAJE AUTOMÁTICOS*. Obtenido de <https://www.noegasystems.com/blog/almacenes-automaticos/sistemas-de-almacenaje-automaticos-2>
- PeakTech. (s.f.). *PeakTech 1200*. Obtenido de <https://www.peaktech.de/productdetail/kategorie/alte-modelle/produkt/p-1200.html>
- Premier Farnell Limited. (2018). *AFG-2112 - Function Generator, Arbitrary / Direct Digital Synthesis (DDS), 1 Channel, 12 MHz, AFG-2100 Series*. Obtenido de <http://uk.farnell.com/gw-instek/afg-2112/function-generator-1ch-arb-dds/dp/2100023>
- Premier Farnell Limited. (2018). *GPS-3303 - Fuente de Poder de Mesa, CD Lineal, Adjustable, Fixed, 3 Salidas, 0 V, 30 V, 0 A, 3 A*. Obtenido de <http://mexico.newark.com/gw-instek/gps-3303/power-supply-bench-adj-3o-p-30v/dp/86K0329>
- Robotica-dec. (2018). *SABERTOOTH 2X60*. Obtenido de http://www.robotica-dec.com.mx/index.php?route=product/product&product_id=363
- Shamshiri, R., & Wan Ismail, W. I. (2012). *CONTROL SYSTEM DESIGN FOR A FIELD SURVEY AGRICULTURAL MOBILE ROBOT*. Florida: University of Florida.
- Siemens AG. (1996). *Standard Software for S7-300 and S7-400 PID Control*. Obtenido de https://www.fer.unizg.hr/_download/repository/S7pidcob.pdf
- Siemens AG. (2018). *Procesador de comunicaciones para SIMATIC S7-300*. Obtenido de <https://w3.siemens.com/mcms/industrial-communication/es/ie/conexiones-de-sistema/simatic-s7-sinumerik-o/s7-300/pages/cp343-1lean.aspx>
- Siemens AG. (2018). *SIMATIC S7-300*. Obtenido de <https://w3.siemens.com/mcms/programmable-logic-controller/en/advanced-controller/s7-300/Pages/Default.aspx>
- Siemens AG. (2018). *Totally Integrated Automation Portal*. Obtenido de <https://w5.siemens.com/spain/web/es/industry/automatizacion/simatic/tia-portal/pages/tiaportal.aspx>
- Smolka-berlin.de. (s.f.). *SWF VALEO NIDEC ITT 404.319 gear motor 24 V DC Typ:SWMK*. Obtenido de https://www.smolka-berlin.de/onlineshop/en/artikel/SWF_VALEO_NIDEC_ITT_404.319_Motor,_Getriebemotor_24_V_DC/4068
- Tangient LLC. (2018). *CONTROL-PID*. Obtenido de <http://control-pid.wikispaces.com/>
- VENANDCOM. (s.f.). *Procesador de comunicaciones Siemens 6GK7 343-1CX10-OXEO*. Obtenido de <https://www.venandcom.es/es/p/procesador-de-comunicaciones-siemens-6gk7-343-1cx10-oxeo/1570>

ANEXO 1

**PROGRAMACIÓN PARA LAS GRÁFICAS
DE PRUEBAS DE LOS MOTORES M1-M2,
ERROR RELATIVO Y ABSOLUTO EN
MATLAB**

```

%Autores:
%Jean Calle
%Luiggi Constante
%Gráficas de pruebas de los Motores M1 y M2, error relativo y absoluto - Proyecto
de Titulación

```

```

clc
clear all
close all
%Datos del Motor M1
R1 = 4.4571;          %Resistencia de Armadura
L1 = 0.00087053;    %Inductancia de Armadura
kb1 = 0.051595966;  %Constante de la fuerza contra electromotriz
kt1 = 0.051595966;  %Constante del par electromecánico
B1 = 0.000312892;   %Coeficiente de fricción viscosa
J1 = 0.0000219256;  %Momento de inercia del motor
%Datos del Motor M2
R2 = 7.24285;        %Resistencia de Armadura
L2 = 0.00091608;    %Inductancia de Armadura
kb2 = 0.050447731;  %Constante de la fuerza contra electromotriz
kt2 = 0.050447731;  %Constante del par electromecánico
B2 = 0.00031466;    %Coeficiente de fricción viscosa
J2 = 0.000019457;   %Momento de inercia del motor

```

```

%Gráfica de entradas con señal cuadrada del motor M1 (velocidad)

```

```

figure(1)
subplot(2,1,1);
plot(ts1, in1,'b')
hold on;
plot(sinc(:,1), sinc(:,2),'r')
hold off;
axis([0 10 -4 8])
title('Entradas Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
legend('Model M1 DC','Real M1 DC');
grid;

```

```

%Gráfica de salidas con señal cuadrada del motor M1 (velocidad)

```

```

subplot(2,1,2);
plot(ts1, out1,'b')
hold on;
plot(tdc2m1(:,1), tdc2m1(:,2),'r')
hold off;
axis([0 10 -500 1000])
title('Salidas Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Velocidad (RPM)');
legend('Model M1 DC','Real M1 DC');
grid;
%Error Absoluto
Ea = abs(mean(sqrt((out1).^2) - sqrt((tdc2m1(:,2)).^2)))
%Error Relativo
Er = Ea / (abs(mean(sqrt((tdc2m1(:,2)).^2))))* 100

```

```

%Gráfica de entradas con señal cuadrada del motor M2 (velocidad)
figure(2)
subplot(2,1,1);
plot(ts1, in2,'b')
hold on;
plot(sinc(:,1), sinc(:,2),'r')
hold off;
axis([0 10 -4 8])
title('Entradas Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
legend('Model M2 DC','Real M2 DC');
grid;

```

```

%Gráfica de salidas con señal cuadrada del motor M2 (velocidad)
subplot(2,1,2);
plot(ts1, out2,'b')
hold on;
plot(tdc3m2(:,1), tdc3m2(:,2),'r')
hold off;
axis([0 10 -500 1000])
title('Salidas Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Velocidad (RPM)');
legend('Model M2 DC','Real M2 DC');
grid;

```

```

%Error Absoluto
Ea = abs(mean(sqrt((out2).^2) - sqrt((tdc3m2(:,2)).^2)))
%Error Relativo
Er = Ea / (abs(mean(sqrt((tdc3m2(:,2)).^2))))* 100

```

```

%Gráfica de entradas con señal seno del motor M1 (velocidad)
figure(3)
subplot(2,1,1);
plot(ts1, in1,'b')
hold on;
plot(sins(:,1), sins(:,2),'r')
hold off;
axis([0 10 -4 8])
title('Entradas Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
legend('Model M1 DC','Real M1 DC');
grid;

```

```

%Gráfica de salidas con señal seno del motor M1 (velocidad)
subplot(2,1,2);
plot(ts1, out1,'b')
hold on;
plot(tds1m1(:,1), tds1m1(:,2),'r')
hold off;
axis([0 10 -500 1000])
title('Salidas Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Velocidad (RPM)');

```

```

legend('Model M1 DC','Real M1 DC');
grid;
%Error Absoluto
Ea = abs(mean(sqrt((out1).^2) - sqrt((tds1m1(:,2)).^2)))
%Error Relativo
Er = Ea / (abs(mean(sqrt((tds1m1(:,2)).^2))))* 100

%Gráfica de entradas con señal seno del motor M2 (velocidad)
figure(4)
subplot(2,1,1);
plot(ts1, in2,'b')
hold on;
plot(sins(:,1), sins(:,2),'r')
hold off;
axis([0 10 -4 8])
title('Entradas Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
legend('Model M2 DC','Real M2 DC');
grid;

%Gráfica de salidas con señal seno del motor M2 (velocidad)
subplot(2,1,2);
plot(ts1, out2,'b')
hold on;
plot(tds1m2(:,1), tds1m2(:,2),'r')
hold off;
axis([0 10 -500 1000])
title('Salidas Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Velocidad (RPM)');
legend('Model M2 DC','Real M2 DC');
grid;
%Error Absoluto
Ea = abs(mean(sqrt((out2).^2) - sqrt((tds1m2(:,2)).^2)))
%Error Relativo
Er = Ea / (abs(mean(sqrt((tds1m2(:,2)).^2))))* 100

%Gráfica de entradas con señal cuadrada del motor M1 (distancia)
figure(5)
subplot(2,1,1);
plot(ts1, in1,'b')
hold on;
plot(sinc(:,1), sinc(:,2),'r')
hold off;
axis([0 10 -4 8])
title('Entradas Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
legend('Model M1 DC','Real M1 DC');
grid;

%Gráfica de salidas con señal cuadrada del motor M1 (distancia)
subplot(2,1,2);
plot(ts1, out3,'b')

```



```

hold on;
plot(discm1(:,1), discm1(:,2),'r')
hold off;
axis([0 10 -1 1])
title('Salidas Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (Cm)');
legend('Model M1 DC','Real M1 DC');
grid;
%Error Absoluto
Ea = abs(mean(out3 - discm1(:,2)))
%Error Relativo
Er = Ea / (abs(mean(discm1(:,2))))* 100

%Gráfica de entradas con señal cuadrada del motor M2 (distancia)
figure(6)
subplot(2,1,1);
plot(ts1, in2,'b')
hold on;
plot(sinc(:,1), sinc(:,2),'r')
hold off;
axis([0 10 -4 8])
title('Entradas Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
legend('Model M2 DC','Real M2 DC');
grid;

%Gráfica de salidas con señal cuadrada del motor M2 (distancia)
subplot(2,1,2);
plot(ts1, out4,'b')
hold on;
plot(discm2(:,1), discm2(:,2),'r')
hold off;
axis([0 10 -1 1])
title('Salidas Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Distancia (Cm)');
legend('Model M2 DC','Real M2 DC');
grid;
%Error Absoluto
Ea = abs(mean(out4 - discm2(:,2)))
%Error Relativo
Er = (Ea / (abs(mean(discm2(:,2)))))* 100

%Gráfica de entradas con señal seno del motor M1 (distancia)
figure(7)
subplot(2,1,1);
plot(ts1, in1,'b')
hold on;
plot(sins(:,1), sins(:,2),'r')
hold off;
axis([0 10 -4 8])
title('Entradas Motor M1')
xlabel('Tiempo (segundos)');

```

```

ylabel('Voltaje (voltios)');
legend('Model M1 DC','Real M1 DC');
grid;

%Gráfica de salidas con señal seno del motor M1 (distancia)
subplot(2,1,2);
plot(ts1, out3,'b')
hold on;
plot(dissm1(:,1), dissm1(:,2),'r')
hold off;
axis([0 10 -0.5 1.5])
title('Salidas Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (Cm)');
legend('Model M1 DC','Real M1 DC');
grid;
%Error Absoluto
Ea = abs(mean(out3 - dissm1(:,2)))
%Error Relativo
Er = (Ea / (abs(mean(dissm1(:,2))))) * 100

```

```

%Gráfica de entradas con señal seno del motor M2 (distancia)
figure(8)
subplot(2,1,1);
plot(ts1, in1,'b')
hold on;
plot(sins(:,1), sins(:,2),'r')
hold off;
axis([0 10 -4 8])
title('Entradas Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
legend('Model M2 DC','Real M2 DC');
grid;

```

```

%Gráfica de salidas con señal seno del motor M2 (distancia)
subplot(2,1,2);
plot(ts1, out4,'b')
hold on;
plot(dissm2(:,1), dissm2(:,2),'r')
hold off;
axis([0 10 -0.5 1.5])
title('Salidas Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Distancia (Cm)');
legend('Model M2 DC','Real M2 DC');
grid;
%Error Absoluto
Ea = abs(mean(out4 - dissm2(:,2)))
%Error Relativo
Er = (Ea / (abs(mean(dissm2(:,2))))) * 100

```

ANEXO 2

**PROGRAMACIÓN PARA LAS GRÁFICAS
DEL CONTROLADOR PID DE LOS
MOTORES M1 Y M2 EN MATLAB**

```

%Autores:
%Jean Calle
%Luiggi Constante
%Gráficas del Controlador PID Motores M1 y M2 - Proyecto de Titulación

```

```

clc
clear all
close all
%Datos del Motor M1
R1 = 4.4571;          %Resistencia de Armadura
L1 = 0.00087053;    %Inductancia de Armadura
kb1 = 0.051595966;  %Constante de la fuerza contra electromotriz
kt1 = 0.051595966;  %Constante del par electromecánico
B1 = 0.000312892;   %Coeficiente de fricción viscosa
J1 = 0.0000219256;  %Momento de inercia del motor
%Datos del Motor M2
R2 = 7.24285;        %Resistencia de Armadura
L2 = 0.00091608;    %Inductancia de Armadura
kb2 = 0.050447731;  %Constante de la fuerza contra electromotriz
kt2 = 0.050447731;  %Constante del par electromecánico
B2 = 0.00031466;    %Coeficiente de fricción viscosa
J2 = 0.000019457;   %Momento de inercia del motor

```

```

%Gráfica del controlador P del motor M1 (distancia)
figure(1)
%subplot(2,2,1);
plot(ts, outpm1,'b')
hold on;
plot(ts, outdzpm1,'g')
plot(P_M1(:,1), P_M1(:,2),'r')
hold off;
axis([0 150 0 70])
title('Controlador P Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model P M1 DC','Real P M1 DC');
legend('Model P M1 DC','Model DZ P M1 DC','Real P M1 DC');
grid;

```

```

%Gráfica del controlador PD del motor M1 (distancia)
figure(2)
%subplot(2,2,2);
plot(ts, outpdm1,'b')
hold on;
plot(ts, outdzpdm1,'g')
plot(PD_M1(:,1), PD_M1(:,2),'r')
hold off;
axis([0 150 0 70])
title('Controlador PD Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model PD M1 DC','Real PD M1 DC');
legend('Model PD M1 DC','Model DZ PD M1 DC','Real PD M1 DC');
grid;

```

```

%Gráfica del controlador PI del motor M1 (distancia)
figure(3)
%subplot(2,2,3);
plot(ts, outpim1,'b')
hold on;
plot(PI_M1(:,1), PI_M1(:,2),'r')
hold off;
axis([0 200 0 70])
title('Controlador PI Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
legend('Model PI M1 DC','Real PI M1 DC');
grid;

```

```

%Gráfica del controlador PID del motor M1 (distancia)
figure(4)
%subplot(2,2,4);
plot(ts, outpidm1,'b')
hold on;
plot(PID_M1(:,1), PID_M1(:,2),'r')
hold off;
axis([0 200 0 70])
title('Controlador PID Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
legend('Model PID M1 DC','Real PID M1 DC');
grid;

```

```

%Gráfica del controlador P del motor M2 (distancia)
figure(5)
%subplot(2,2,1);
plot(ts, outpm2,'b')
hold on;
%plot(ts, outdzpm2,'g')
plot(P_M2(:,1), P_M2(:,2),'r')
hold off;
axis([0 150 0 50])
title('Controlador P Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model P M2 DC');
legend('Model P M2 DC','Real P M2 DC');
%legend('Model P M2 DC','Model DZ P M2 DC','Real P M2 DC');
grid;

```

```

%Gráfica del controlador PD del motor M2 (distancia)
figure(6)
%subplot(2,2,2);
plot(ts, outpdm2,'b')
hold on;
plot(ts, outdzpdm2,'g')
plot(PD_M2(:,1), PD_M2(:,2),'r')
hold off;
axis([0 150 0 50])
title('Controlador PD Motor M2')

```

```

xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model PD M2 DC');
%legend('Model PD M2 DC','Real PD M2 DC');
legend('Model PD M2 DC','Model DZ PD M2 DC','Real PD M2 DC');
grid;

%Gráfica del controlador PI del motor M2 (distancia)
figure(7)
%subplot(2,2,3);
plot(ts, outpim2,'b')
hold on;
plot(PI_M2(:,1), PI_M2(:,2),'r')
%hold off;
axis([0 200 0 50])
title('Controlador PI Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model PI M2 DC');
legend('Model PI M2 DC','Real PI M2 DC');
grid;

%Gráfica del controlador PID del motor M2 (distancia)
figure(8)
%subplot(2,2,4);
plot(ts, outpidm2,'b')
hold on;
plot(PID_M2(:,1), PID_M2(:,2),'r')
hold off;
axis([0 200 0 50])
title('Controlador PID Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
legend('Model PID M2 DC','Real PID M2 DC');
%legend('Model PID M2 DC','Real PID M2 DC');
grid;

%Gráfica de la señal de control P del motor M1 (distancia)
figure(9)
plot(ts, seconpm1,'b')
hold on;
%plot(ts, outdzpim1,'g')
plot(P_secon_M1(:,1), P_secon_M1(:,2),'r')
hold off;
axis([0 150 -5 25])
title('Señal de Control P del Motor DC M1')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
legend('Model Señal de Control P M1 DC','Real Señal de Control P M1 DC');
grid;

%Gráfica de la señal de control PD del motor M1 (distancia)
figure(10)
plot(ts, seconpdm1,'b')
hold on;

```

```

%plot(ts, outdzipim1,'g')
plot(PD_secon_M1(:,1), PD_secon_M1(:,2),'r')
hold off;
axis([0 150 -5 25])
title('Señal de Control PD del Motor DC M1')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
legend('Model Señal de Control PD M1 DC','Real PD M1 DC');
grid;

%Gráfica de la señal de control PI del motor M1 (distancia)
figure(11)
plot(ts, seconpim1,'b')
hold on;
%plot(ts, outdzipim1,'g')
plot(PI_secon_M1(:,1), PI_secon_M1(:,2),'r')
hold off;
axis([0 200 -5 25])
title('Señal de Control PI del Motor DC M1')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
legend('Model Señal de Control PI M1 DC','Real Señal de Control PI M1 DC');
grid;

%Gráfica de la señal de control PID del motor M1 (distancia)
figure(12)
plot(ts, seconpidm1,'b')
hold on;
%plot(ts, outdzipim1,'g')
plot(PID_secon_M1(:,1), PID_secon_M1(:,2),'r')
hold off;
axis([0 200 -5 25])
title('Señal de Control PID del Motor DC M1')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
legend('Model Señal de Control PID M1 DC','Real Señal de Control PID M1 DC');
grid;

%Gráfica de la señal de control P del motor M2 (distancia)
figure(13)
plot(ts, senconpm2,'b')
%hold on;
%plot(ts, outdzipim1,'g')
%plot(P_secon_M2(:,1), P_secon_M2(:,2),'r')
%hold off;
axis([0 150 -5 25])
title('Señal de Control P del Motor DC M2')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
legend('Model Señal de Control P M2 DC');
%legend('Model Señal de Control P M2 DC','Real Señal de Control P M2 DC');
grid;

%Gráfica de la señal de control PD del motor M2 (distancia)
figure(14)

```

```

plot(ts, senconpdm2,'b')
hold on;
%plot(ts, outdzipim1,'g')
plot(PD_secon_M2(:,1), PD_secon_M2(:,2),'r')
hold off;
axis([0 150 -5 25])
title('Señal de Control PD del Motor DC M2')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
%legend('Model Señal de Control PD M2 DC');
legend('Model Señal de Control PD M2 DC','Real Señal de Control PD M2 DC');
grid;

%Gráfica de la señal de control PI del motor M2 (distancia)
figure(15)
plot(ts, senconpim2,'b')
hold on;
%plot(ts, outdzipim1,'g')
plot(PI_secon_M2(:,1), PI_secon_M2(:,2),'r')
hold off;
axis([0 200 -5 25])
title('Señal de Control PI del Motor DC M2')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
%legend('Model Señal de Control PI M2 DC');
legend('Model Señal de Control PI M2 DC','Real Señal de Control PI M2 DC');
grid;

%Gráfica de la señal de control PID del motor M2 (distancia)
figure(16)
plot(ts, senconpidm2,'b')
hold on;
%plot(ts, outdzipim1,'g')
plot(PID_secon_M2(:,1), PID_secon_M2(:,2),'r')
hold off;
axis([0 200 -5 25])
title('Señal de Control PID del Motor DC M2')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
%legend('Model Señal de Control PID M2 DC');
legend('Model Señal de Control PID M2 DC','Real Señal de Control PID M2 DC');
grid;

```


ANEXO 3

**PROGRAMACIÓN PARA LAS GRÁFICAS
DEL CONTROLADOR FUZZY DE LOS
MOTORES M1 Y M2 EN MATLAB**

```

%Autores:
%Jean Calle
%Luiggi Constante
%Gráficas del Controlador Fuzzy Motores M1 y M2 - Proyecto de Titulación

```

```

clc
clear all
close all
%Datos del Motor M1
R1 = 4.4571;      %Resistencia de Armadura
L1 = 0.00087053; %Inductancia de Armadura
kb1 = 0.051595966; %Constante de la fuerza contra electromotriz
kt1 = 0.051595966; %Constante del par electromecánico
B1 = 0.000312892; %Coeficiente de fricción viscosa
J1 = 0.0000219256; %Momento de inercia del motor
%Datos del Motor M2
R2 = 7.24285;      %Resistencia de Armadura
L2 = 0.00091608; %Inductancia de Armadura
kb2 = 0.050447731; %Constante de la fuerza contra electromotriz
kt2 = 0.050447731; %Constante del par electromecánico
B2 = 0.00031466; %Coeficiente de fricción viscosa
J2 = 0.000019457; %Momento de inercia del motor

```

```

%Gráfica del controlador Fuzzy del motor M1 (distancia)

```

```

figure(1)
%subplot(2,1,1);
plot(t, salidaFuzzy_M1,'b')
hold on;
plot(Fuzzy_M1(:,1), Fuzzy_M1(:,2),'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 40 0 70])
title('Controlador Fuzzy Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Model Fuzzy M1 DC','Real Fuzzy M1 DC');
grid;

```

```

%Gráfica del controlador Fuzzy del motor M2 (distancia)

```

```

figure(2)
%subplot(2,1,2);
plot(t, salidaFuzzy_M2,'b')
hold on;
%plot(p2(:,1), p2(:,2),'r')
plot(Fuzzy_M2(:,1), Fuzzy_M2(:,2),'r')
hold off;
axis([0 40 0 50])
title('Controlador Fuzzy Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M2 DC');
legend('Model Fuzzy M2 DC','Real Fuzzy M2 DC');
grid;
%,'Model DZ PD M1 DC'

```

```

%Gráfica de la señal de control del controlador Fuzzy del motor M1 (distancia)
figure(3)
%subplot(2,1,1);
plot(t, control_M1,'b')
hold on;
%plot(ts, outdzipim1,'g')
plot(Fuzzy_M1_secon(:,1), Fuzzy_M1_secon(:,2),'r')
hold off;
axis([0 40 -5 25])
title('Señal de Control Fuzzy del Motor DC M1')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
%legend('Model Señal de Control Fuzzy M1 DC');
legend('Model Señal de Control Fuzzy M1 DC','Real Señal de Control Fuzzy M1 DC');
grid;

```

```

%Gráfica de la señal de control del controlador Fuzzy del motor M2 (distancia)
figure(4)
%subplot(2,1,2);
plot(t, control_M2,'b')
hold on;
%plot(ts, outdzipim1,'g')
plot(Fuzzy_M2_secon(:,1), Fuzzy_M2_secon(:,2),'r')
hold off;
axis([0 40 -5 25])
title('Señal de Control Fuzzy del Motor DC M2')
xlabel('Tiempo (segundos)');
ylabel('Voltaje (voltios)');
%legend('Model Señal de Control Fuzzy M2 DC');
legend('Model Señal de Control Fuzzy M2 DC','Real Señal de Control Fuzzy M2 DC');
grid;

```

ANEXO 4

**PROGRAMACIÓN PARA LAS GRÁFICAS
DEL CONTROLADOR PID VS FUZZY DE
LOS MOTORES M1 Y M2 EN MATLAB**

```

%Autores:
%Jean Calle
%Luiggi Constante
%Gráficas del Controlador PID vs Fuzzy Motores M1 y M2 - Proyecto de Titulación

```

```

clc
clear all
close all
%Datos del Motor M1
R1 = 4.4571;      %Resistencia de Armadura
L1 = 0.00087053; %Inductancia de Armadura
kb1 = 0.051595966; %Constante de la fuerza contra electromotriz
kt1 = 0.051595966; %Constante del par electromecánico
B1 = 0.000312892; %Coeficiente de fricción viscosa
J1 = 0.0000219256; %Momento de inercia del motor
%Datos del Motor M2
R2 = 7.24285;      %Resistencia de Armadura
L2 = 0.00091608; %Inductancia de Armadura
kb2 = 0.050447731; %Constante de la fuerza contra electromotriz
kt2 = 0.050447731; %Constante del par electromecánico
B2 = 0.00031466; %Coeficiente de fricción viscosa
J2 = 0.000019457; %Momento de inercia del motor

%Gráfica del controlador P vs controlador Fuzzy del motor M1 (distancia)
figure(1)
%subplot(2,1,1);
plot(ts, outpm1,'b')
hold on;
plot(t, salidaFuzzy_M1,'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 70])
title('P vs Fuzzy Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Model P M1 DC','Model Fuzzy M1 DC');
grid;

%Gráfica del controlador P vs controlador Fuzzy del motor M1 (distancia)
figure(2)
%subplot(2,1,1);
plot(P_M1_200(:,1), P_M1_200(:,2),'b')
hold on;
plot(Fuzzy_M1_200(:,1), Fuzzy_M1_200(:,2),'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 70])
title('P vs Fuzzy Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Real P M1 DC','Real Fuzzy M1 DC');
grid;

```

```

%Gráfica del controlador PD vs controlador Fuzzy del motor M1 (distancia)
figure(3)
%subplot(2,1,1);
plot(ts, outpdm1,'b')
hold on;
plot(t, salidaFuzzy_M1,'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 70])
title('PD vs Fuzzy Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Model PD M1 DC','Model Fuzzy M1 DC');
grid;

```

```

%Gráfica del controlador PD vs controlador Fuzzy del motor M1 (distancia)
figure(4)
%subplot(2,1,1);
plot(PD_M1_200(:,1), PD_M1_200(:,2),'b')
hold on;
plot(Fuzzy_M1_200(:,1), Fuzzy_M1_200(:,2),'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 70])
title('PD vs Fuzzy Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Real PD M1 DC','Real Fuzzy M1 DC');
grid;

```

```

%Gráfica del controlador PI vs controlador Fuzzy del motor M1 (distancia)
figure(5)
%subplot(2,1,1);
plot(ts, outpim1,'b')
hold on;
plot(t, salidaFuzzy_M1,'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 70])
title('PI vs Fuzzy Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Model PI M1 DC','Model Fuzzy M1 DC');
grid;

```

```

%Gráfica del controlador PI vs controlador Fuzzy del motor M1 (distancia)
figure(6)
%subplot(2,1,1);
plot(PI_M1(:,1), PI_M1(:,2),'b')
hold on;
plot(Fuzzy_M1_200(:,1), Fuzzy_M1_200(:,2),'r')
%plot(p2(:,1), p2(:,2),'g')

```

```

hold off;
axis([0 200 0 70])
title('PI vs Fuzzy Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Real PI M1 DC','Real Fuzzy M1 DC');
grid;

%Gráfica del controlador PID vs controlador Fuzzy del motor M1 (distancia)
figure(7)
%subplot(2,1,1);
plot(ts, outpidm1,'b')
hold on;
plot(t, salidaFuzzy_M1,'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 70])
title('PID vs Fuzzy Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Model PID M1 DC','Model Fuzzy M1 DC');
grid;

%Gráfica del controlador PID vs controlador Fuzzy del motor M1 (distancia)
figure(8)
%subplot(2,1,1);
plot(PID_M1(:,1), PID_M1(:,2),'b')
hold on;
plot(Fuzzy_M1_200(:,1), Fuzzy_M1_200(:,2),'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 70])
title('PID vs Fuzzy Motor M1')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Real PID M1 DC','Real Fuzzy M1 DC');
grid;

%Gráfica del controlador P vs controlador Fuzzy del motor M2 (distancia)
figure(9)
%subplot(2,1,1);
plot(ts, outpm2,'b')
hold on;
plot(t, salidaFuzzy_M2,'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 50])
title('P vs Fuzzy Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Model P M2 DC','Model Fuzzy M2 DC');

```

```

grid;

%Gráfica del controlador P vs controlador Fuzzy del motor M2 (distancia)
figure(10)
%subplot(2,1,1);
plot(P_M2_200(:,1), P_M2_200(:,2),'b')
hold on;
plot(Fuzzy_M2_200(:,1), Fuzzy_M2_200(:,2),'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 50])
title('P vs Fuzzy Motor M2');
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Real P M2 DC','Real Fuzzy M2 DC');
grid;

%Gráfica del controlador PD vs controlador Fuzzy del motor M2 (distancia)
figure(11)
%subplot(2,1,1);
plot(ts, outpdm2,'b')
hold on;
plot(t, salidaFuzzy_M2,'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 50])
title('PD vs Fuzzy Motor M2');
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Model PD M2 DC','Model Fuzzy M2 DC');
grid;

%Gráfica del controlador PD vs controlador Fuzzy del motor M2 (distancia)
figure(12)
%subplot(2,1,1);
plot(PD_M2_200(:,1), PD_M2_200(:,2),'b')
hold on;
plot(Fuzzy_M2_200(:,1), Fuzzy_M2_200(:,2),'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 50])
title('PD vs Fuzzy Motor M2');
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Real PD M2 DC','Real Fuzzy M2 DC');
grid;

%Gráfica del controlador PI vs controlador Fuzzy del motor M2 (distancia)
figure(13)
%subplot(2,1,1);
plot(ts, outpim2,'b')
hold on;

```



```

plot(t, salidaFuzzy_M2,'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 50])
title('PI vs Fuzzy Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Model PI M2 DC','Model Fuzzy M2 DC');
grid;
%Gráfica del controlador PI vs controlador Fuzzy del motor M2 (distancia)
figure(14)
%subplot(2,1,1);
plot(PI_M2(:,1), PI_M2(:,2),'b')
hold on;
plot(Fuzzy_M2_200(:,1), Fuzzy_M2_200(:,2),'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 50])
title('PI vs Fuzzy Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Real PI M2 DC','Real Fuzzy M2 DC');
grid;
%Gráfica del controlador PID vs controlador Fuzzy del motor M2 (distancia)
figure(15)
%subplot(2,1,1);
plot(ts, outpidm2,'b')
hold on;
plot(t, salidaFuzzy_M2,'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 50])
title('PID vs Fuzzy Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Model PID M2 DC','Model Fuzzy M2 DC');
grid;
%Gráfica del controlador PID vs controlador Fuzzy del motor M2 (distancia)
figure(16)
%subplot(2,1,1);
plot(PID_M2(:,1), PID_M2(:,2),'b')
hold on;
plot(Fuzzy_M2_200(:,1), Fuzzy_M2_200(:,2),'r')
%plot(p2(:,1), p2(:,2),'g')
hold off;
axis([0 200 0 50])
title('PID vs Fuzzy Motor M2')
xlabel('Tiempo (segundos)');
ylabel('Distancia (centímetros)');
%legend('Model Fuzzy M1 DC');
legend('Real PID M2 DC','Real Fuzzy M2 DC');
grid;

```

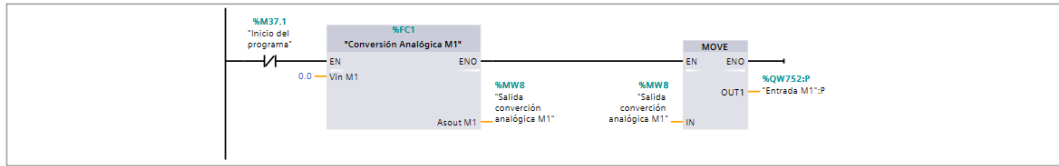
ANEXO 5

PROGRAMACIÓN DE LA PRÁCTICA 3

BLOQUE MAIN – OB1

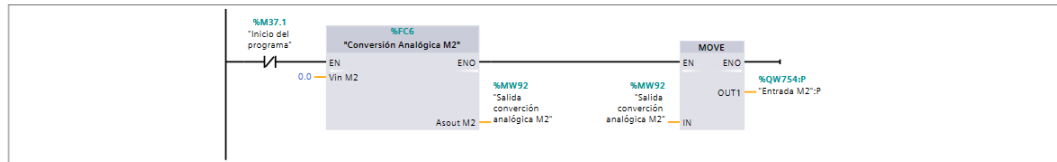
Segmento 1: Condición inicial M1.

Condición inicial del proceso motor DC M1 esté apagado.



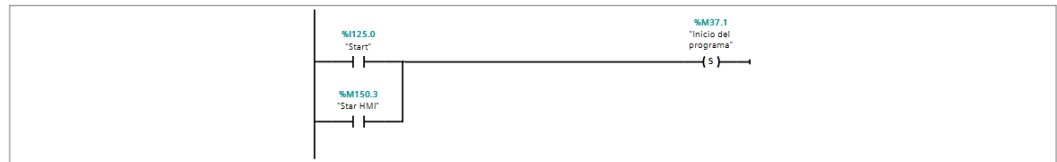
Segmento 2: Condición inicial M2.

Condición inicial del proceso motor DC M2 esté apagado.



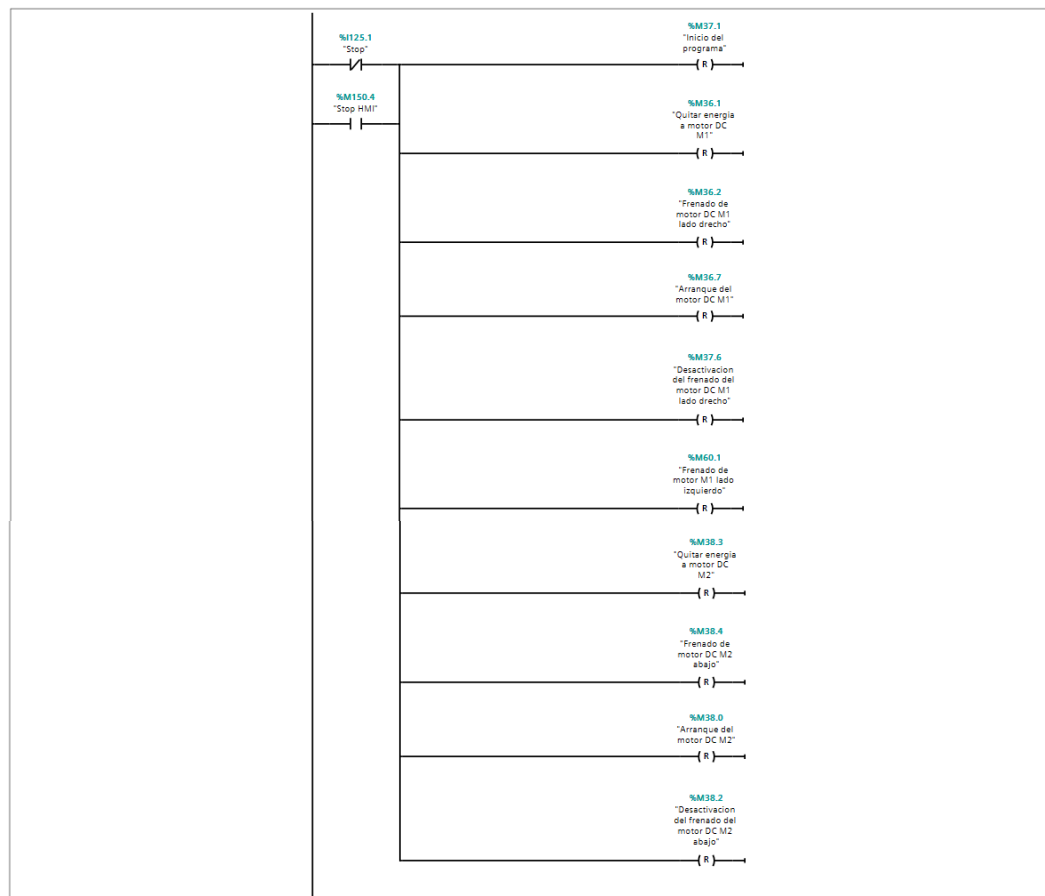
Segmento 3: Start.

Inicio del proceso en la planta a partir del pulsador Start.



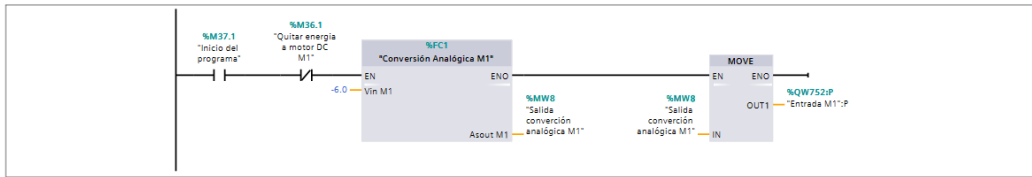
Segmento 4: Stop.

Paro del proceso en la planta a partir del pulsador Stop.



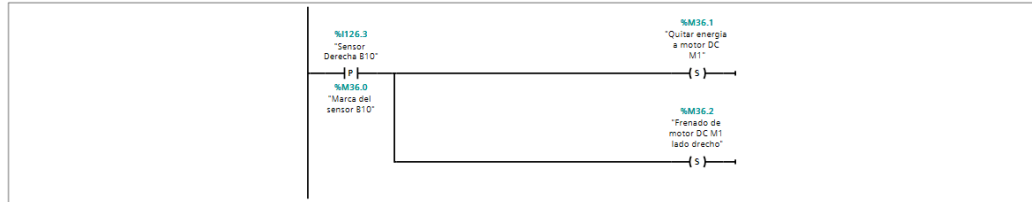
Segmento 5: Ubicación motor M1 a posición inicial.

Ubicación del motor DC M1 a la posición inicial.



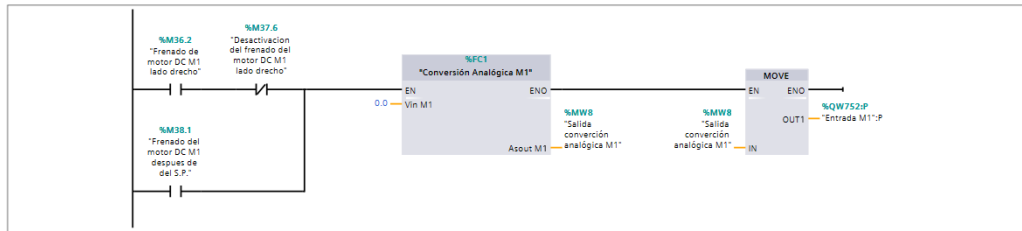
Segmento 6: Detección sensor B10.

Detección del motor DC M1 en la posición inicial sensor B10.



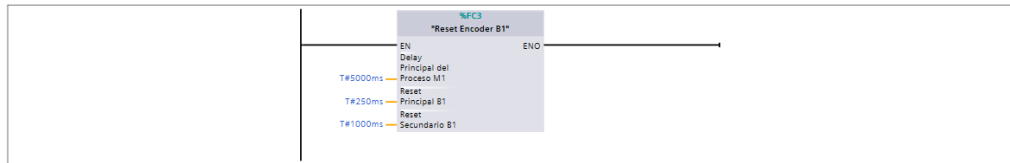
Segmento 7: Frenado motor M1 posición inicial.

Frenado del motor DC M1 en la posición inicial sensor B10.



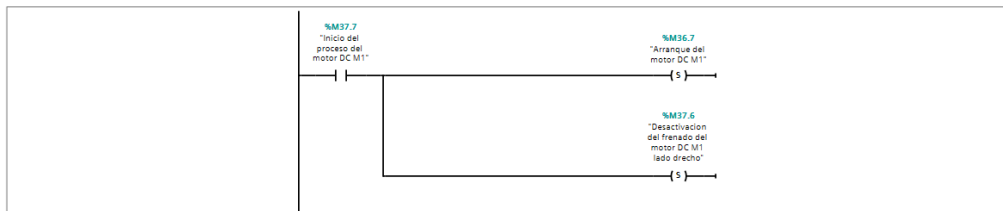
Segmento 8: Reset Encoder B1.

Tiempo de reseteo del encoder B1.



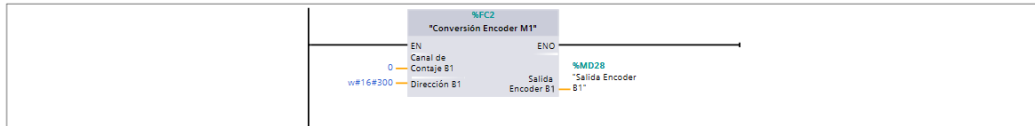
Segmento 9: Inicio del proceso del motor DC M1.

Inicio del proceso del motor DC M1.



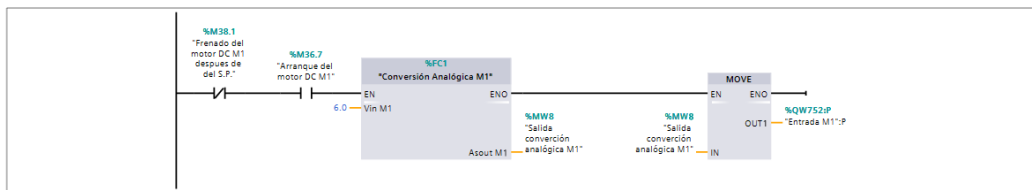
Segmento 10: Conversión Encoder M1.

Conversión de datos del encoder M1.



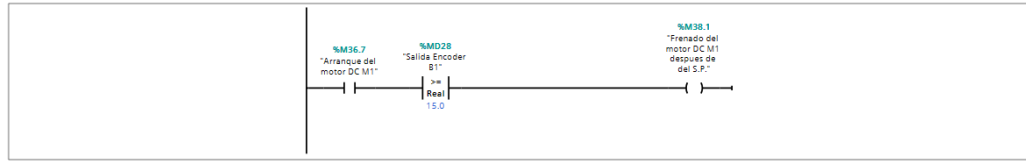
Segmento 11: Arranque del motor M1 hasta Set Point.

Arranque del motor M1 hasta Set Point.



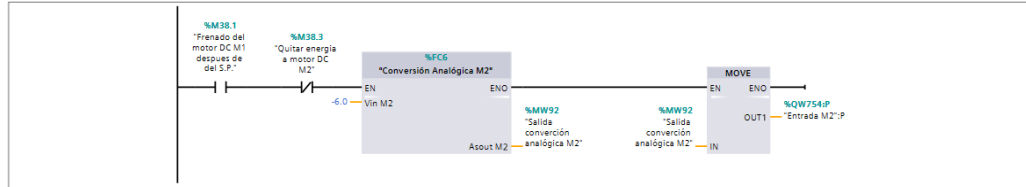
Segmento 12: Condición motor M1 detenerse.

Condición para que el motor M1 se detenga.



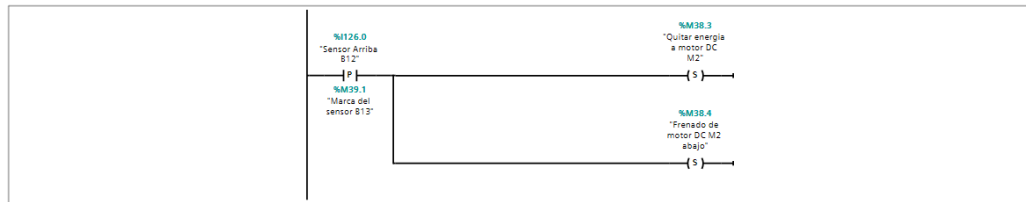
Segmento 13: Arranque del motor DC M2 después del frenado del motor DC M1.

Arranque del motor DC M2 después del frenado del motor DC M1.



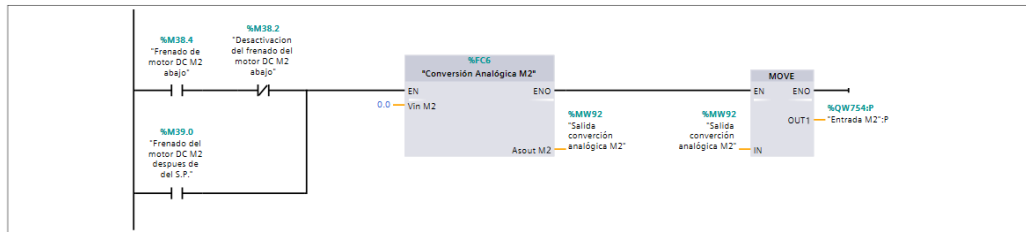
Segmento 14: Detección sensor B10.

Detección del motor DC M2 en la posición inicial sensor B12.



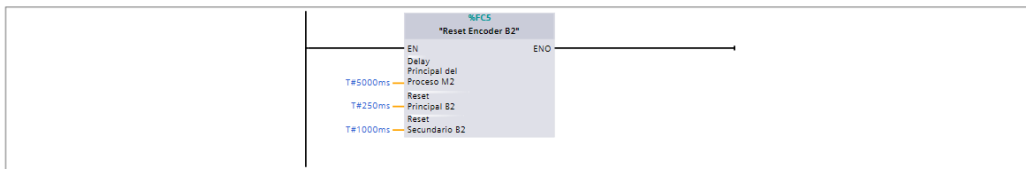
Segmento 15: Frenado del motor M2 en posición inicial.

Frenado del motor M2 en la posición inicial.



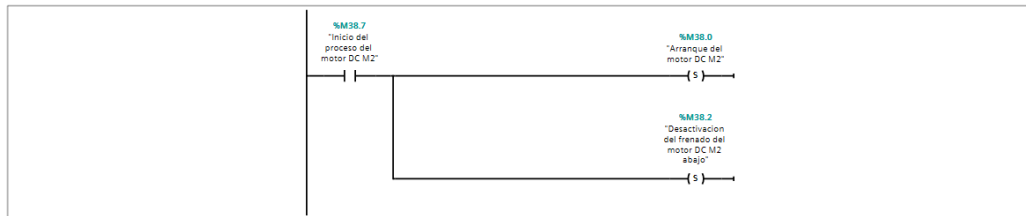
Segmento 16: Reset Encoder B2.

Tiempo de reseteo del encoder B2.



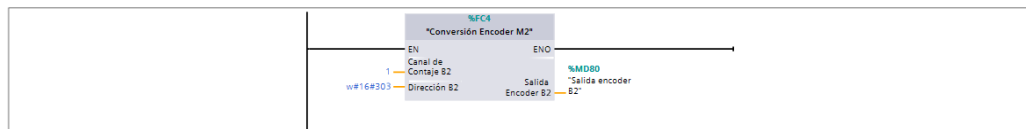
Segmento 17: Inicio del proceso del motor DC M2.

Inicio del proceso del motor DC M2.



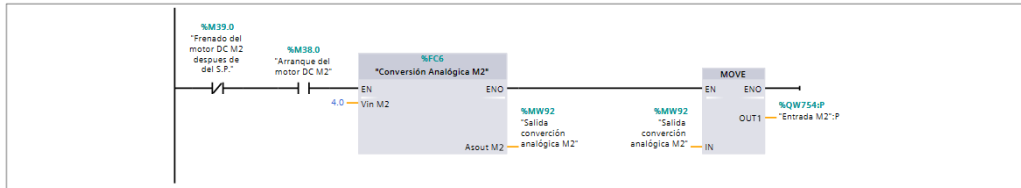
Segmento 18: Conversión Encoder M2.

Conversión Encoder M2.



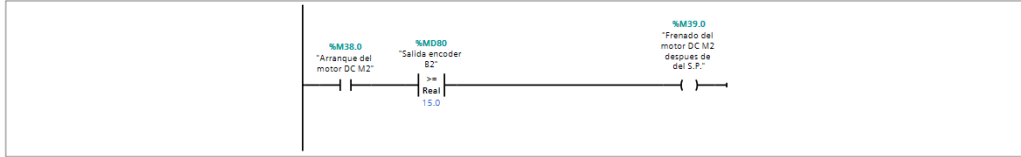
Segmento 19: Arranque del motor M2 hasta Set Point.

Arranque del motor M2 hasta Set Point.



Segmento 20: Condición motor M2 detenerse.

Condición motor M2 detenerse.

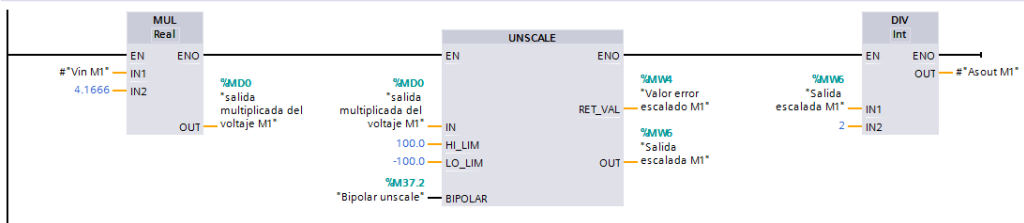


BLOQUES DE FUNCIONES (FC) - DEL MAIN

FC1.

Segmento 1: Conversión Analógica del motor M1.

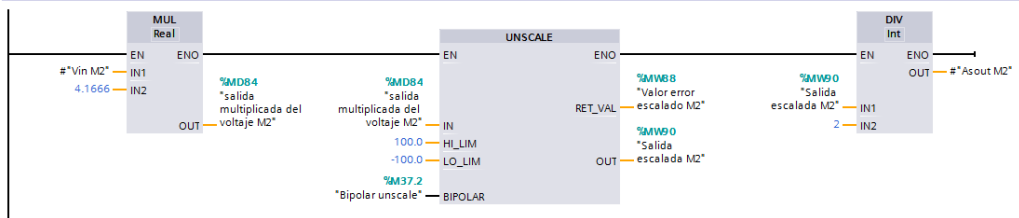
Conversión analógica del dato del voltaje hacia el dato del PLC del motor M1.



FC2.

Segmento 1: Conversión Analógica del motor M2.

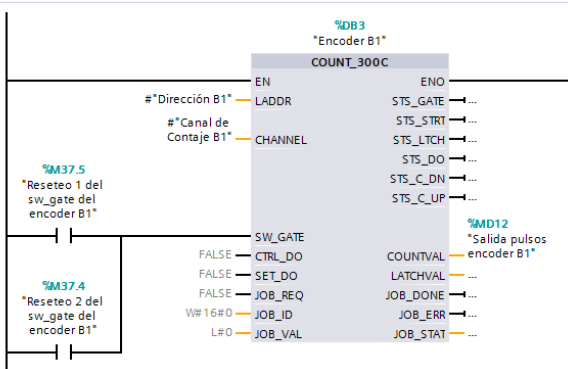
Conversión analógica del dato del voltaje hacia el dato del PLC del motor M2.



FC3.

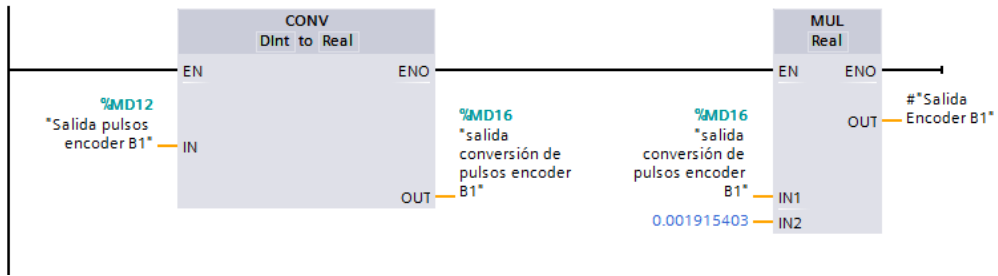
Segmento 1: Lectura pulsos encoder B1.

Lectura de pulsos del encoder B1.



Segmento 2: Conversión pulsos a centímetros encoder B1.

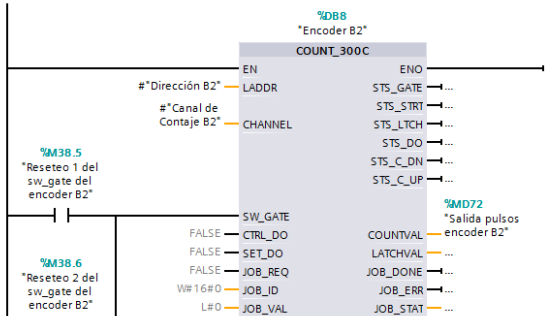
Conversión de pulsos obtenidos a centímetros del encoder B1.



FC4.

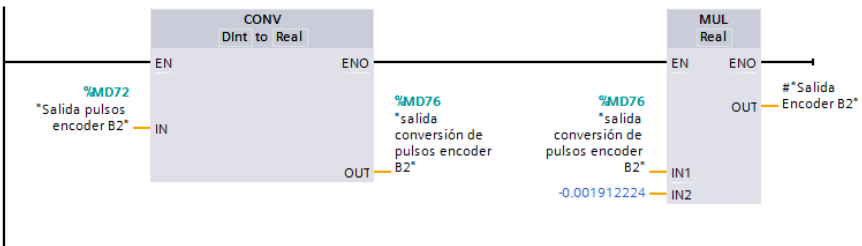
Segmento 1: Lectura pulsos encoder B2.

Lectura de pulsos del encoder B2.



Segmento 2: Conversión pulsos a centímetros encoder B2.

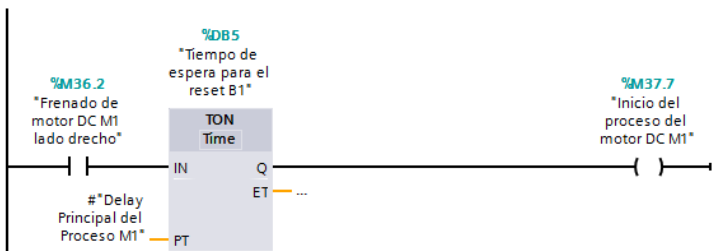
Conversión de pulsos obtenidos a centímetros del encoder B2.



FC5.

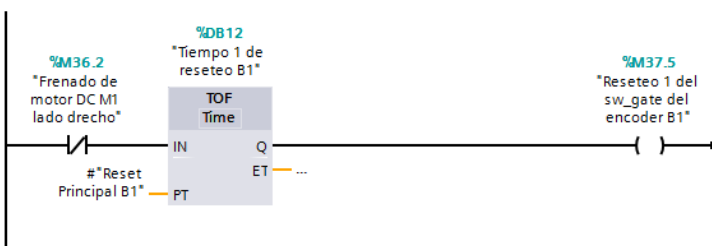
Segmento 1: Tiempo de espera B1 total para el proceso.

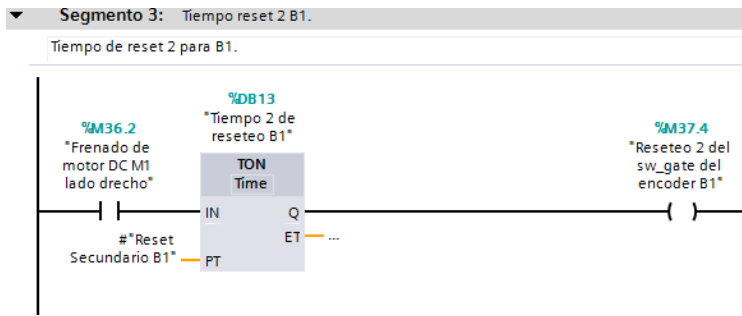
Tiempo de espera B1 total para continuar el proceso.



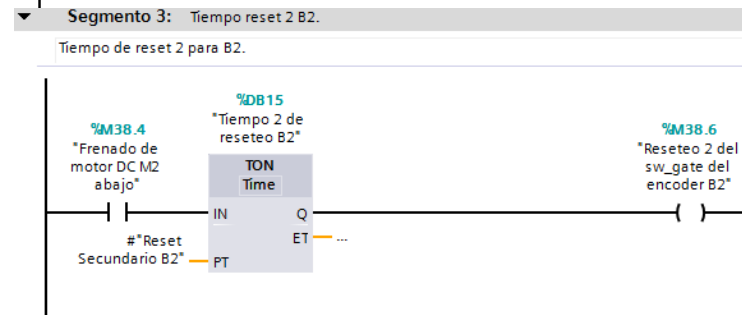
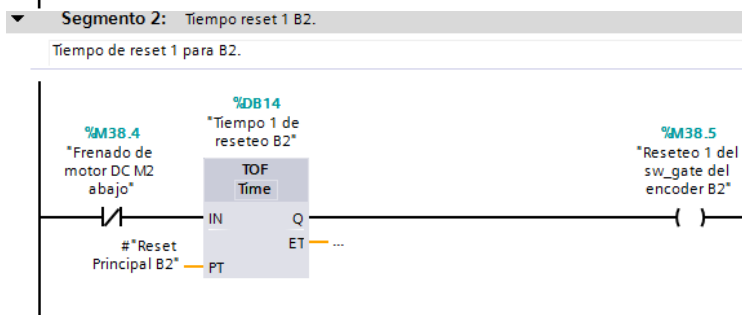
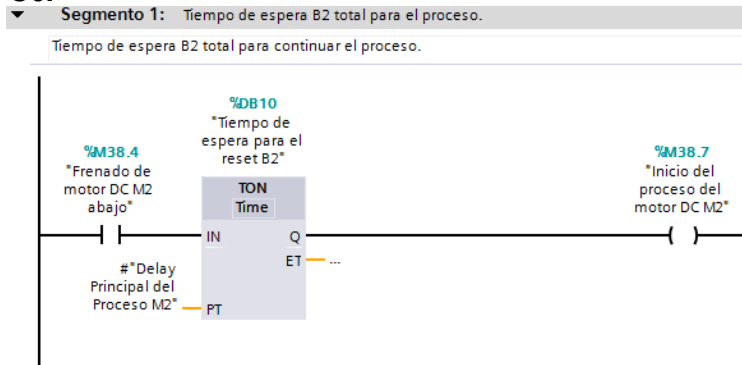
Segmento 2: Tiempo reset 1 B1.

Tiempo de reset 1 para B1.





FC6.



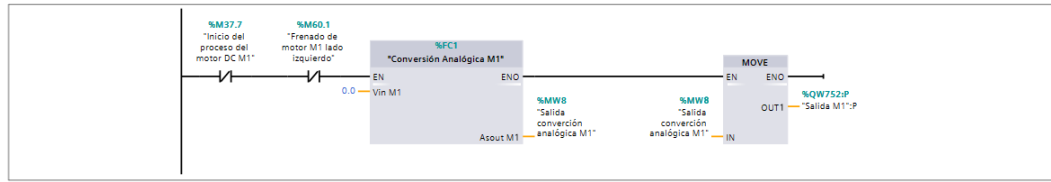
ANEXO 6

PROGRAMACIÓN DE LA PRÁCTICA 8

BLOQUE MAIN

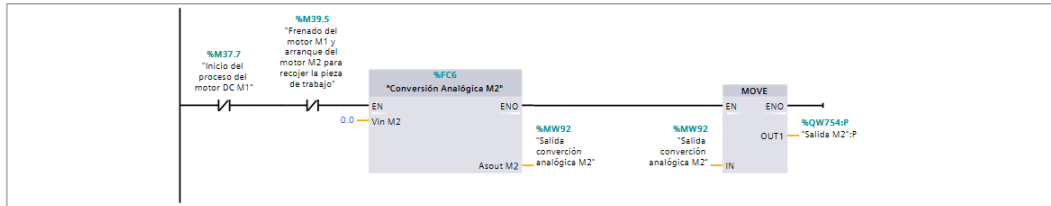
Segmento 1: Condición inicial M1.

Condición inicial del proceso motor DC M1 esté apagado.



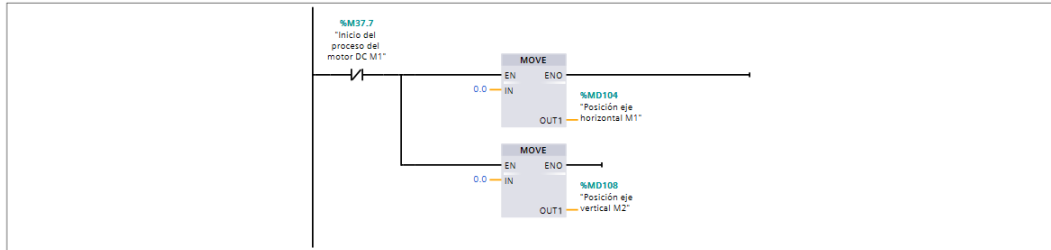
Segmento 2: Condición inicial M2.

Condición inicial del proceso motor DC M2 esté apagado.



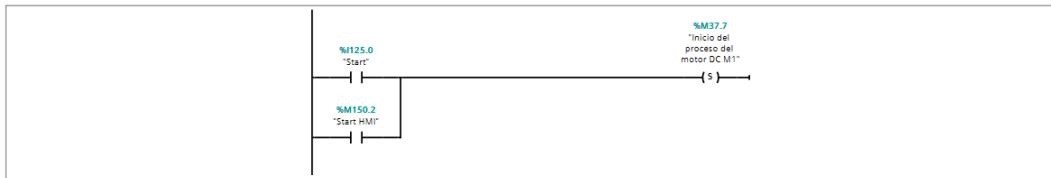
Segmento 3: Condición inicial Set Points.

Condición inicial del proceso Set Points en cero.



Segmento 4: Start.

Inicio del proceso en la planta a partir del pulsador Start.



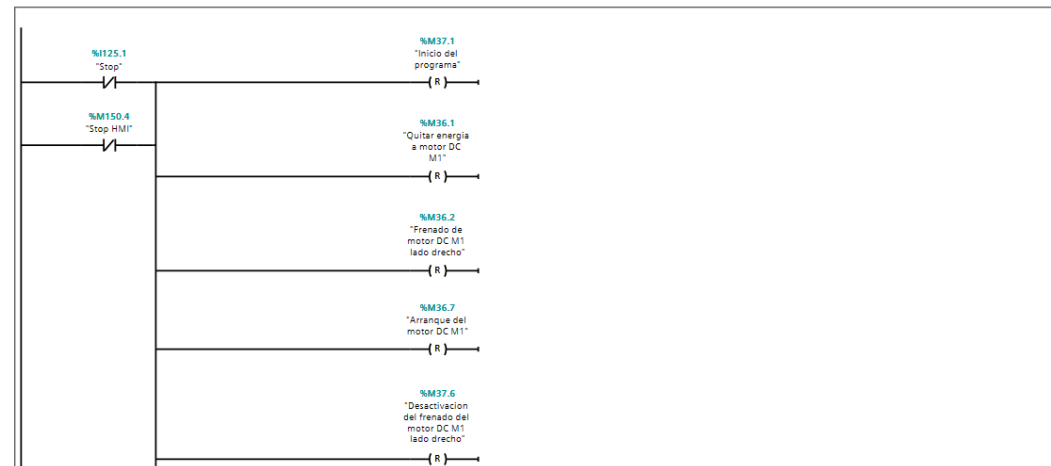
Segmento 5: Proceso Iniciado.

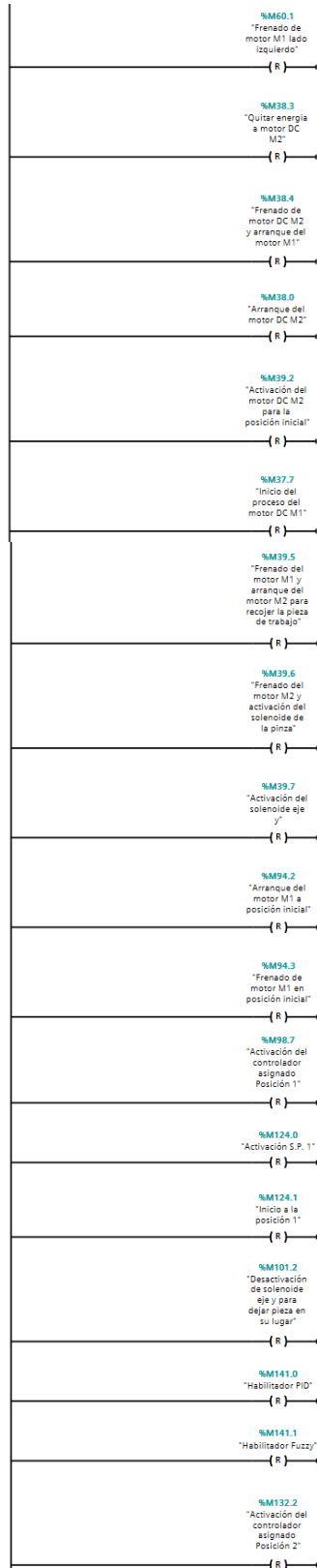
Indicación de que el proceso está activo.

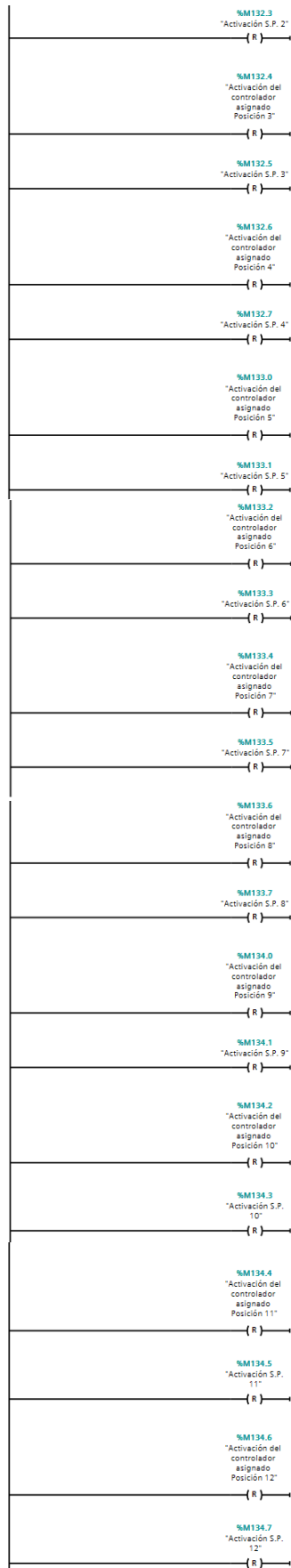


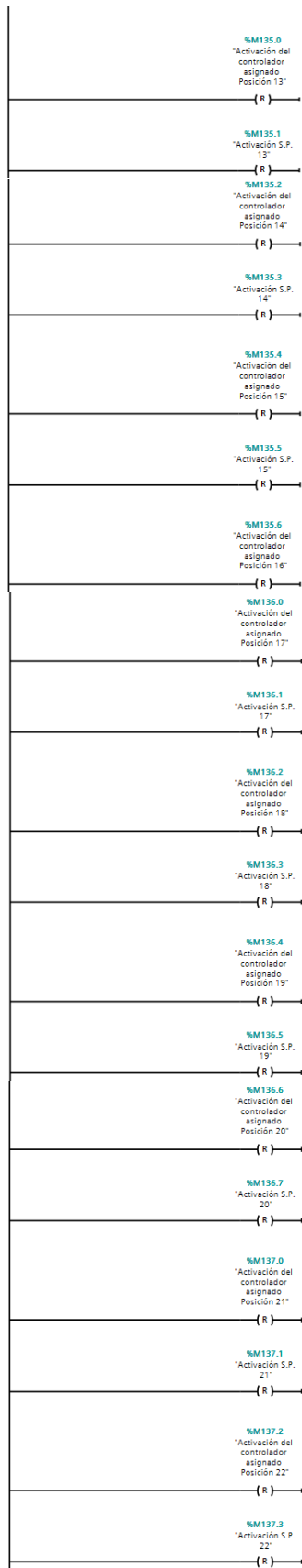
Segmento 6: Stop.

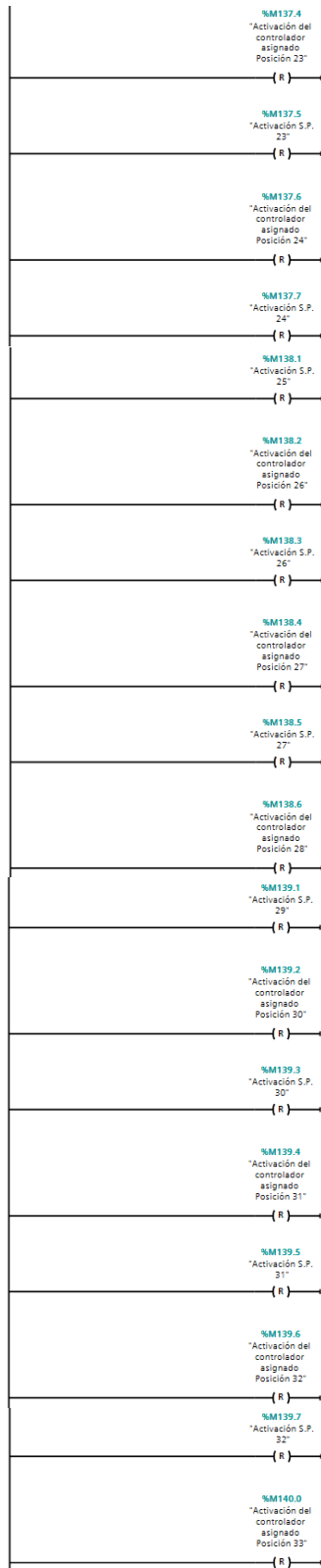
Paro del proceso en la planta a partir del pulsador Stop.

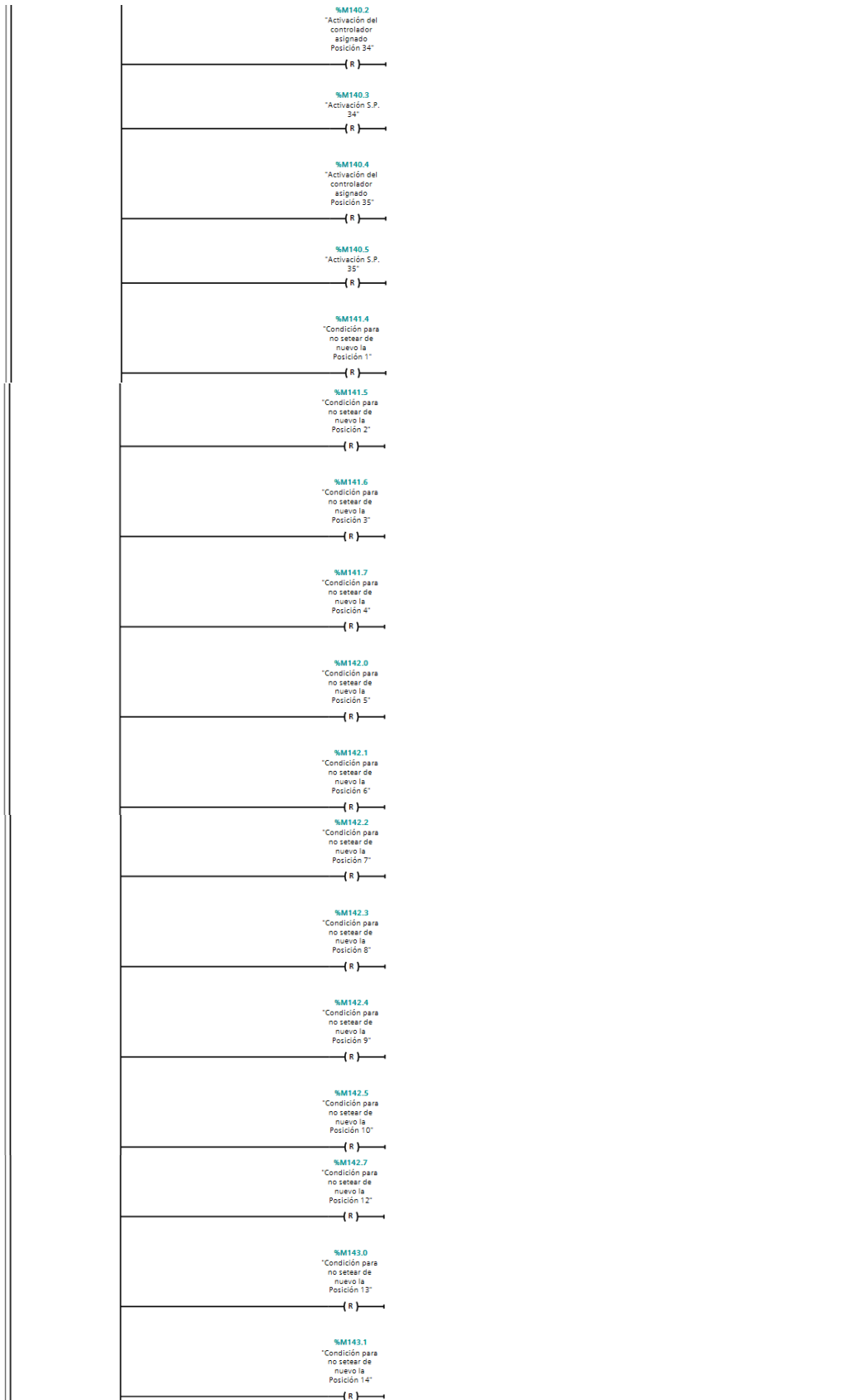




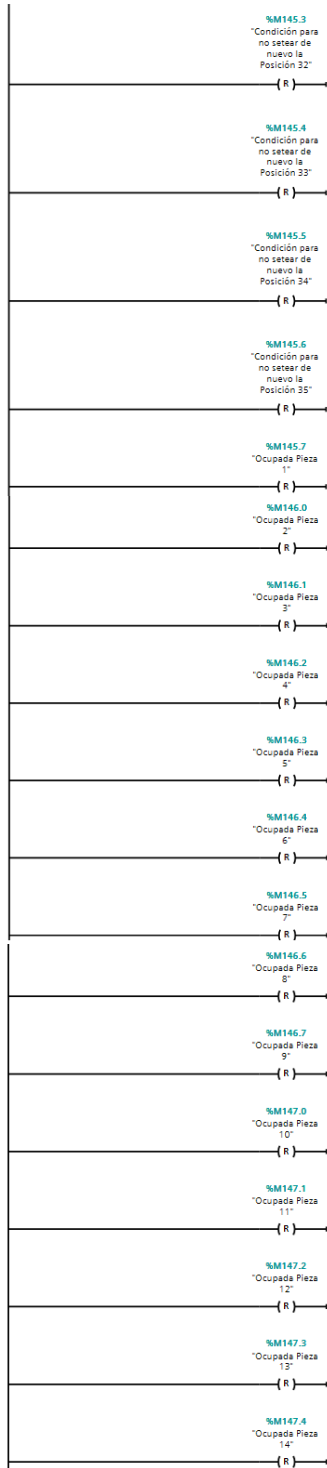


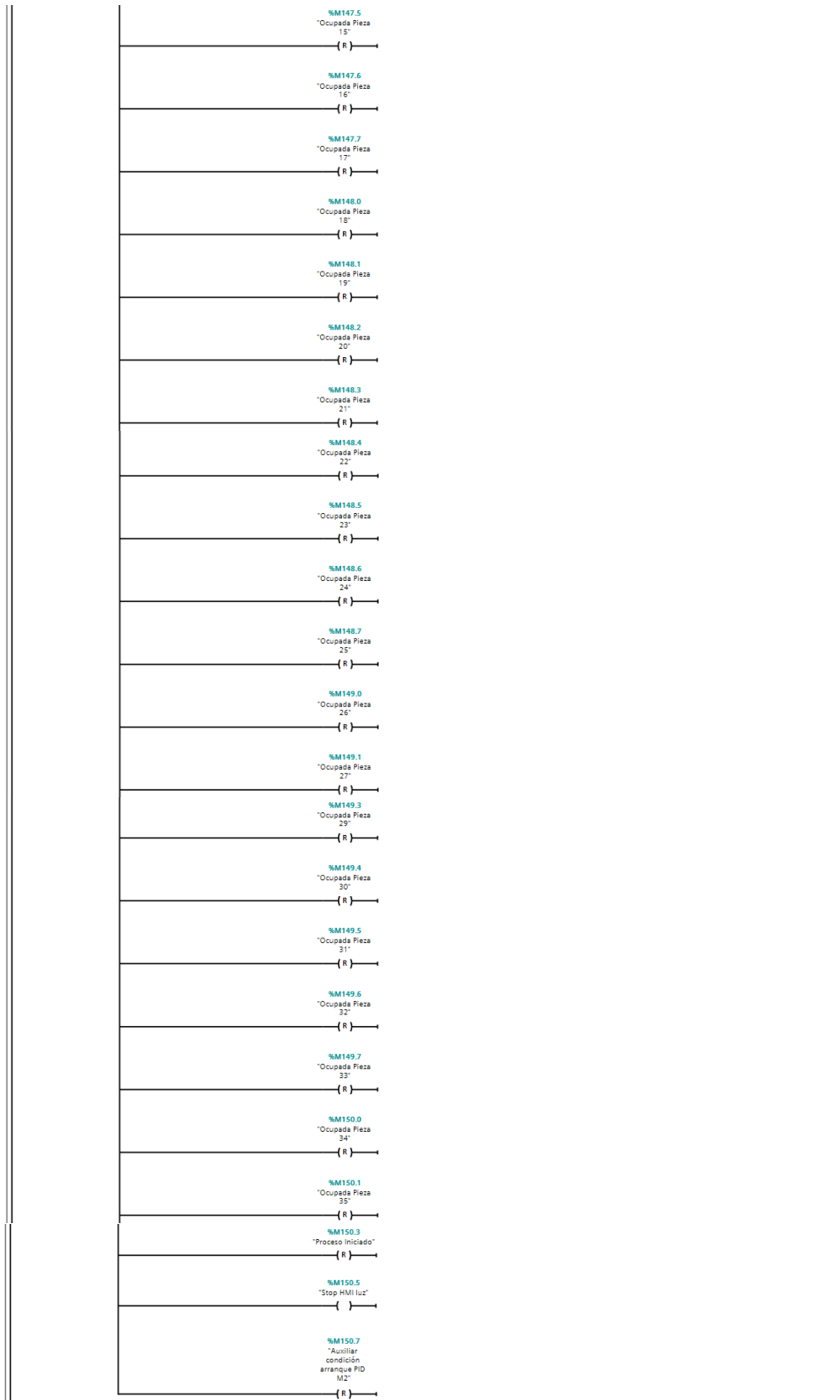






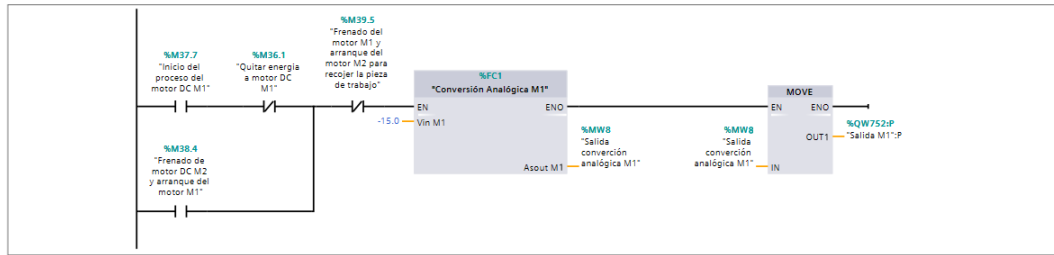
	<p>%M143.2 *Condición para no setear de nuevo la Posición 15*</p> <p>(R)</p>
	<p>%M143.3 *Condición para no setear de nuevo la Posición 16*</p> <p>(R)</p>
	<p>%M143.4 *Condición para no setear de nuevo la Posición 17*</p> <p>(R)</p>
	<p>%M143.5 *Condición para no setear de nuevo la Posición 18*</p> <p>(R)</p>
	<p>%M143.6 *Condición para no setear de nuevo la Posición 19*</p> <p>(R)</p>
	<p>%M143.7 *Condición para no setear de nuevo la Posición 20*</p> <p>(R)</p>
	<p>%M144.0 *Condición para no setear de nuevo la Posición 21*</p> <p>(R)</p>
	<p>%M144.1 *Condición para no setear de nuevo la Posición 22*</p> <p>(R)</p>
	<p>%M144.2 *Condición para no setear de nuevo la Posición 23*</p> <p>(R)</p>
	<p>%M144.3 *Condición para no setear de nuevo la Posición 24*</p> <p>(R)</p>
	<p>%M144.4 *Condición para no setear de nuevo la Posición 25*</p> <p>(R)</p>
	<p>%M144.6 *Condición para no setear de nuevo la Posición 27*</p> <p>(R)</p>
	<p>%M144.7 *Condición para no setear de nuevo la Posición 28*</p> <p>(R)</p>
	<p>%M145.0 *Condición para no setear de nuevo la Posición 29*</p> <p>(R)</p>
	<p>%M145.1 *Condición para no setear de nuevo la Posición 30*</p> <p>(R)</p>
	<p>%M145.2 *Condición para no setear de nuevo la Posición 31*</p> <p>(R)</p>





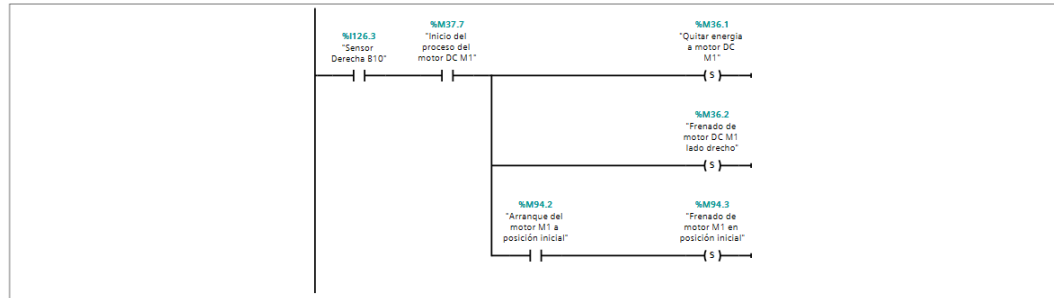
Segmento 7: Ubicación motor M1 a posición inicial.

Ubicación del motor DC M1 a la posición inicial.



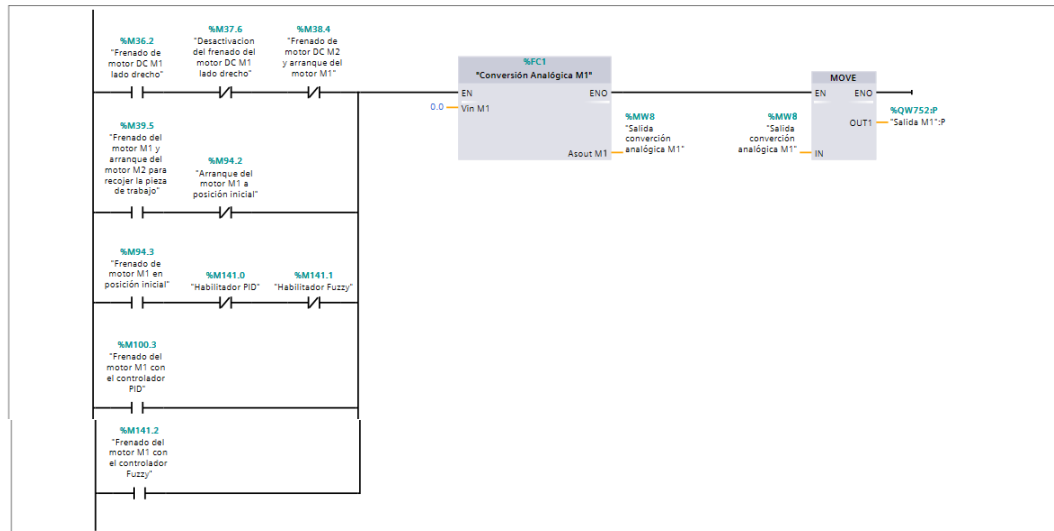
Segmento 8: Detección sensor B10.

Detección del motor DC M1 en la posición inicial sensor B10.



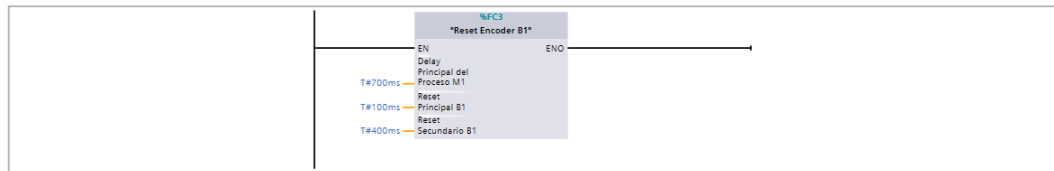
Segmento 9: Frenado motor M1 posición inicial.

Frenado del motor DC M1 en la posición inicial sensor B10.



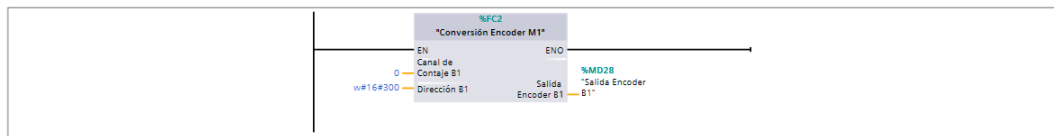
Segmento 10: Reset Encoder B1.

Tiempo de reseteo del encoder B1.



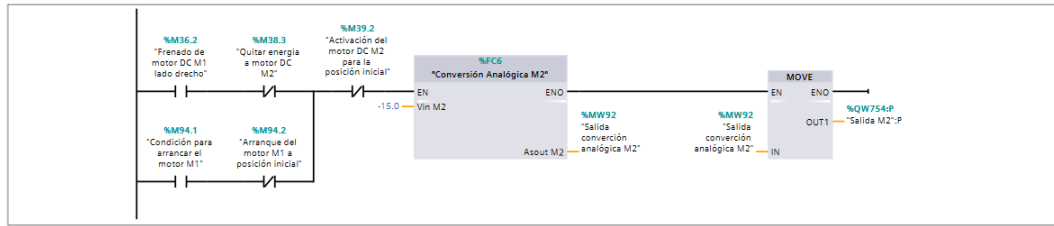
Segmento 11: Conversión Encoder M1.

Conversión de datos del encoder M1.



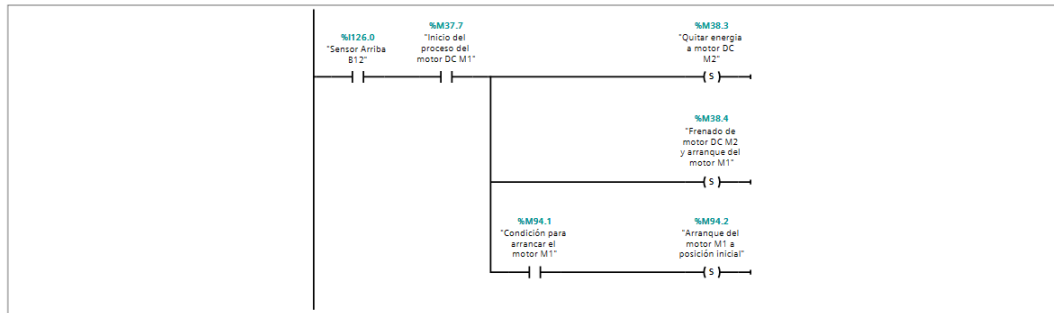
Segmento 12: Ubicación motor M2 posición inicial.

Ubicación del motor DC M2 a la posición inicial.



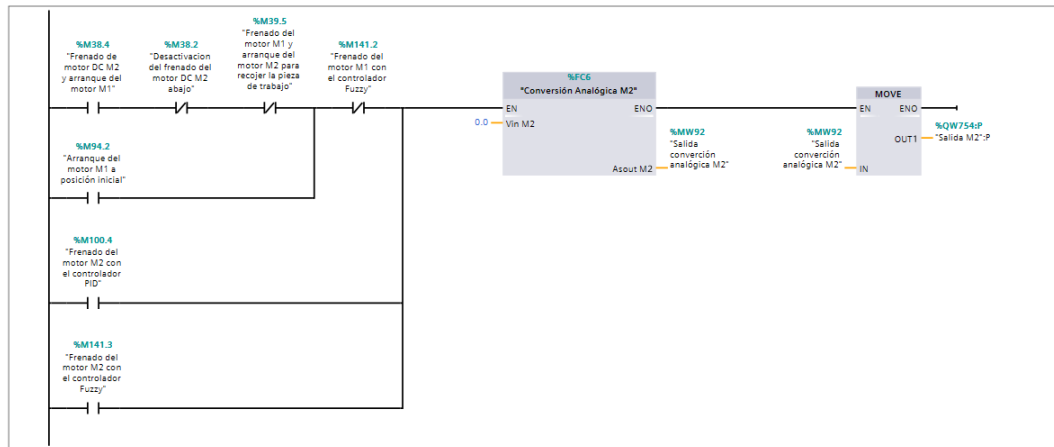
Segmento 13: Detección motor M2 posición inicial.

Detección del motor DC M2 en la posición inicial sensor B12.



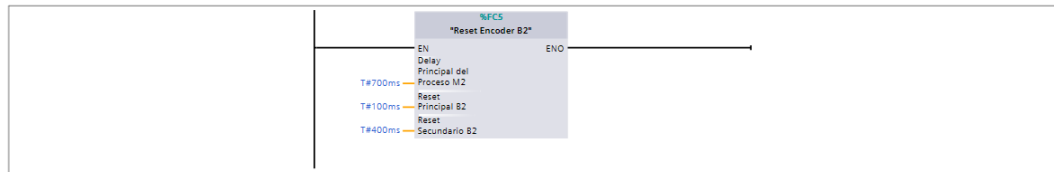
Segmento 14: Frenado motor M2 posición inicial.

Frenado del motor DC M2 en la posición inicial sensor B12.



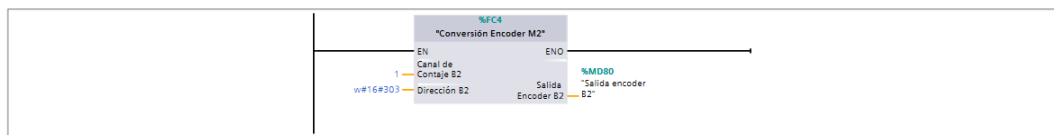
Segmento 15: Reset Encoder B2.

Tiempo de reseteo del encoder B2.



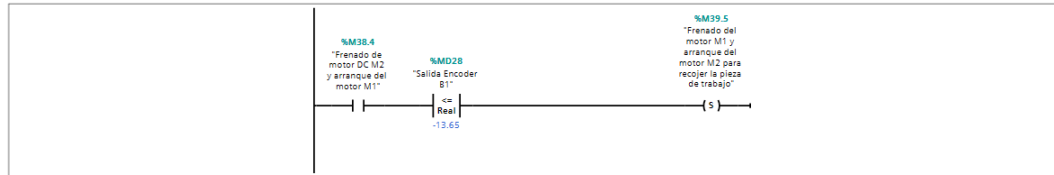
Segmento 16: Conversión Encoder M2.

Conversión de datos del encoder M2.



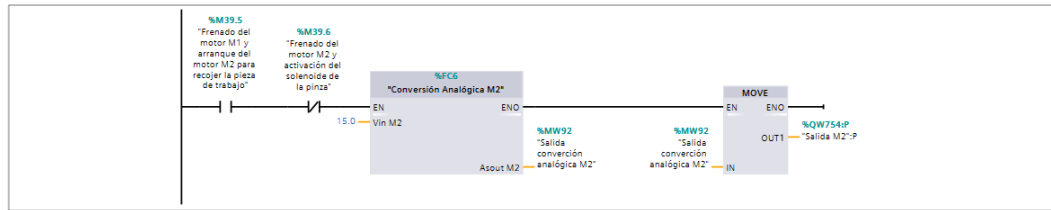
Segmento 17: Condición motor M1 detenerse.

Condición para que el motor M1 se detenga.



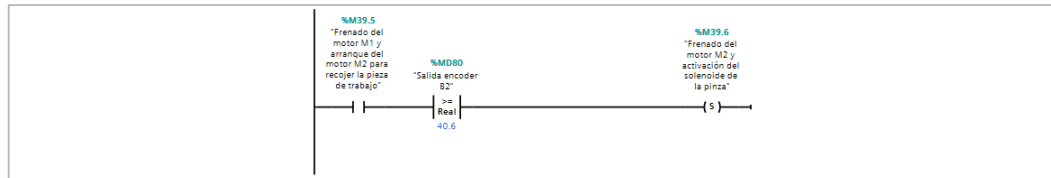
Segmento 18: Arranque motor M2 recolección de pieza.

Arranque del motor M2 para recoger la pieza de trabajo.



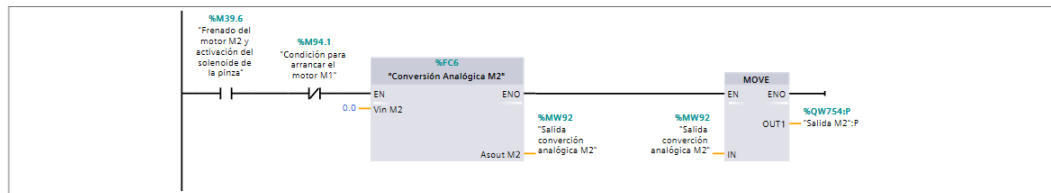
Segmento 19: Condición motor M2 detenerse.

Condición para que el motor M2 se detenga para obtener la pieza de trabajo.



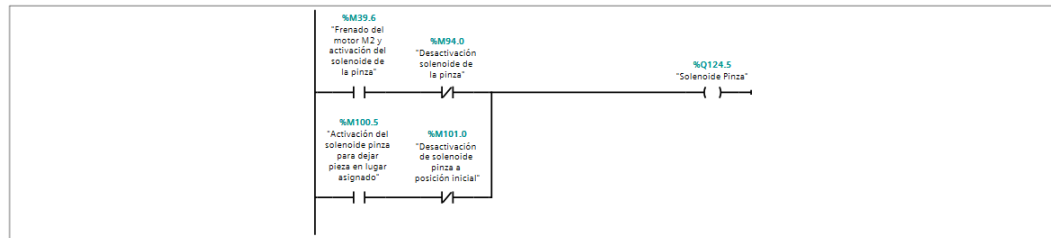
Segmento 20: Frenado motor M2 recolección pieza.

Frenado del motor M2 para recoger la pieza de trabajo.



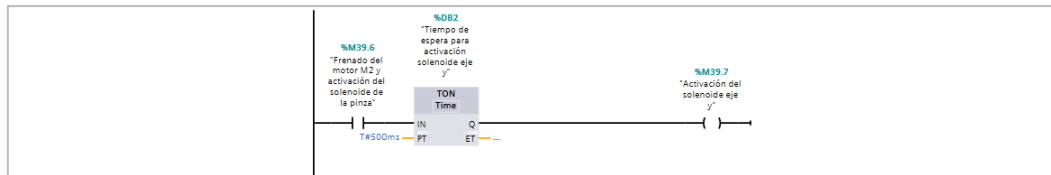
Segmento 21: Activación solenoide pinza.

Activación del solenoide de la pinza.



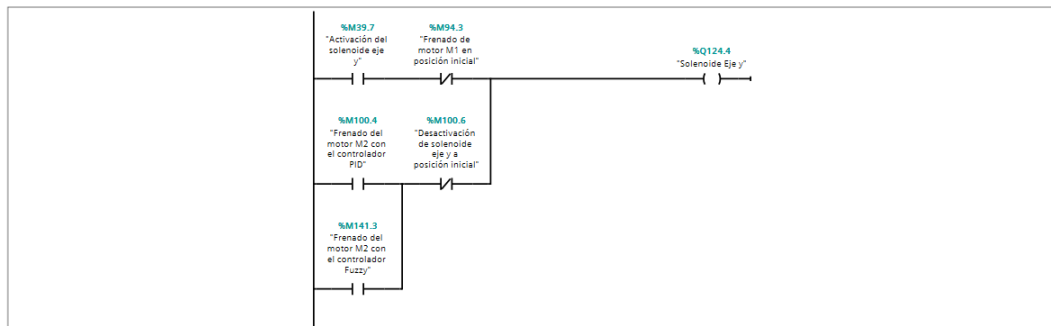
Segmento 22: Tiempo de espera activación solenoide eje y.

Tiempo de espera para la activación del solenoide del eje y.



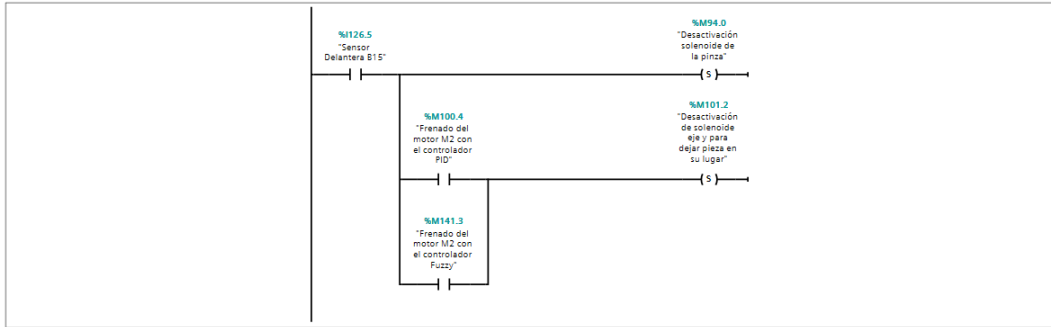
Segmento 23: Activación solenoide eje y.

Activación del solenoide del eje y.



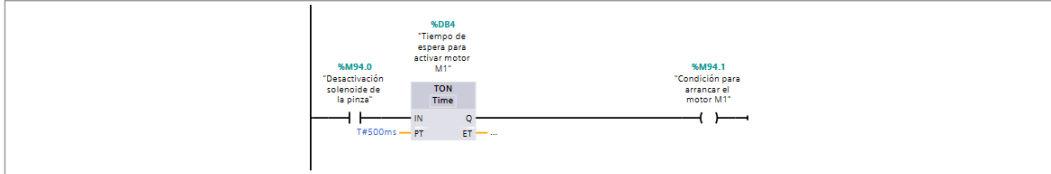
Segmento 24: Desactivación solenoide pinza.

Desactivación del solenoide de la pinza.



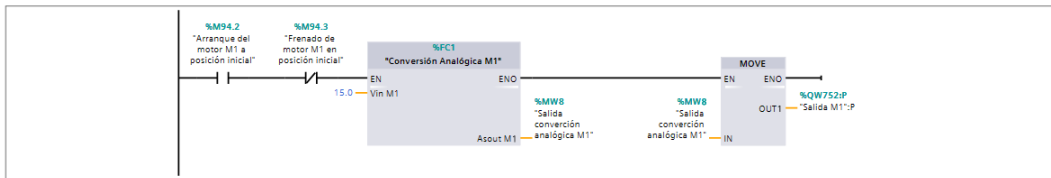
Segmento 25: Tiempo de espera activación M1.

Tiempo de espera para la activación del motor M1.



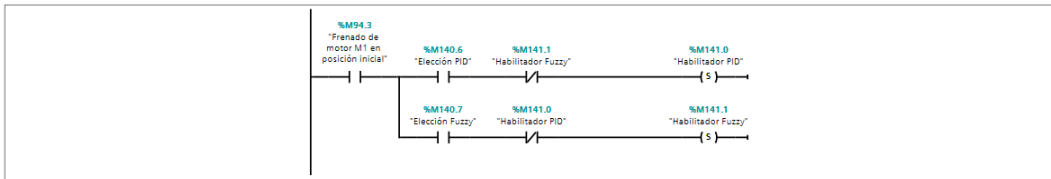
Segmento 26: Arranque motor M1 a posición inicial.

Arranque del motor M1 hacia posición inicial.



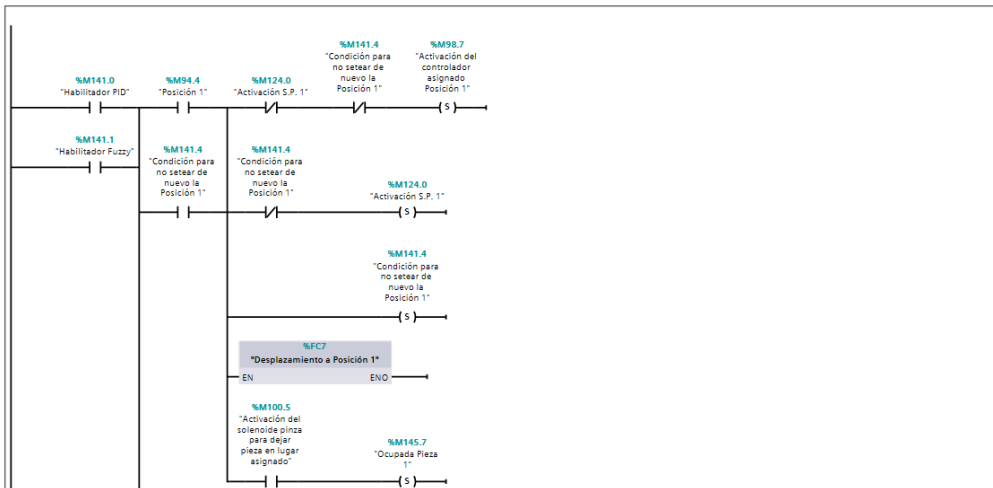
Segmento 27: Habilitador PID o Fuzzy.

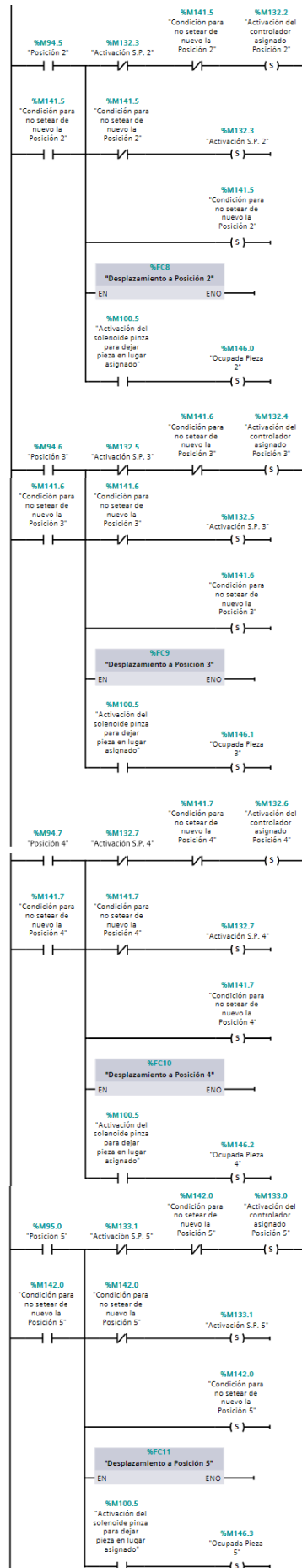
Elección del controlador a trabajar en la planta.

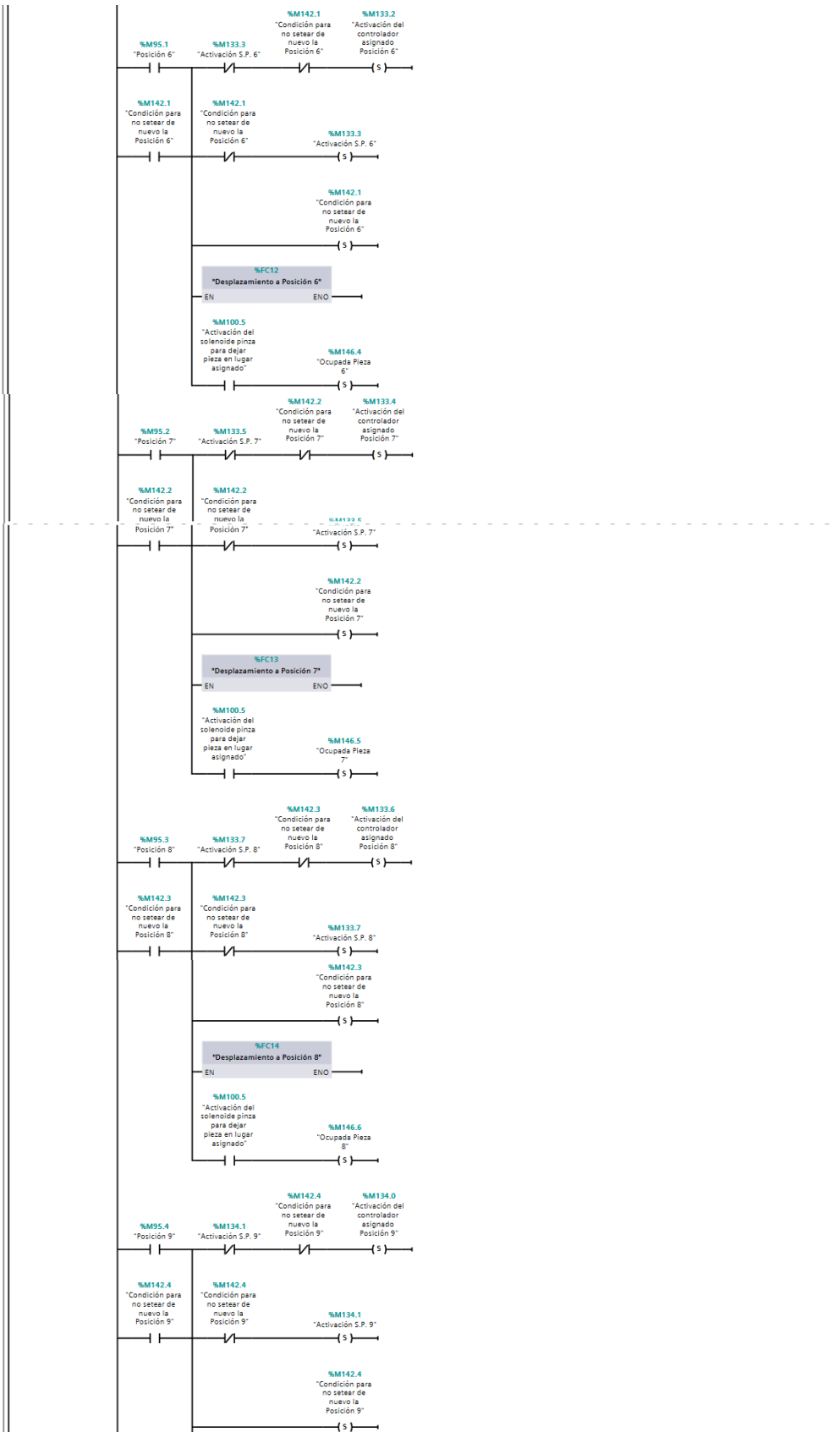


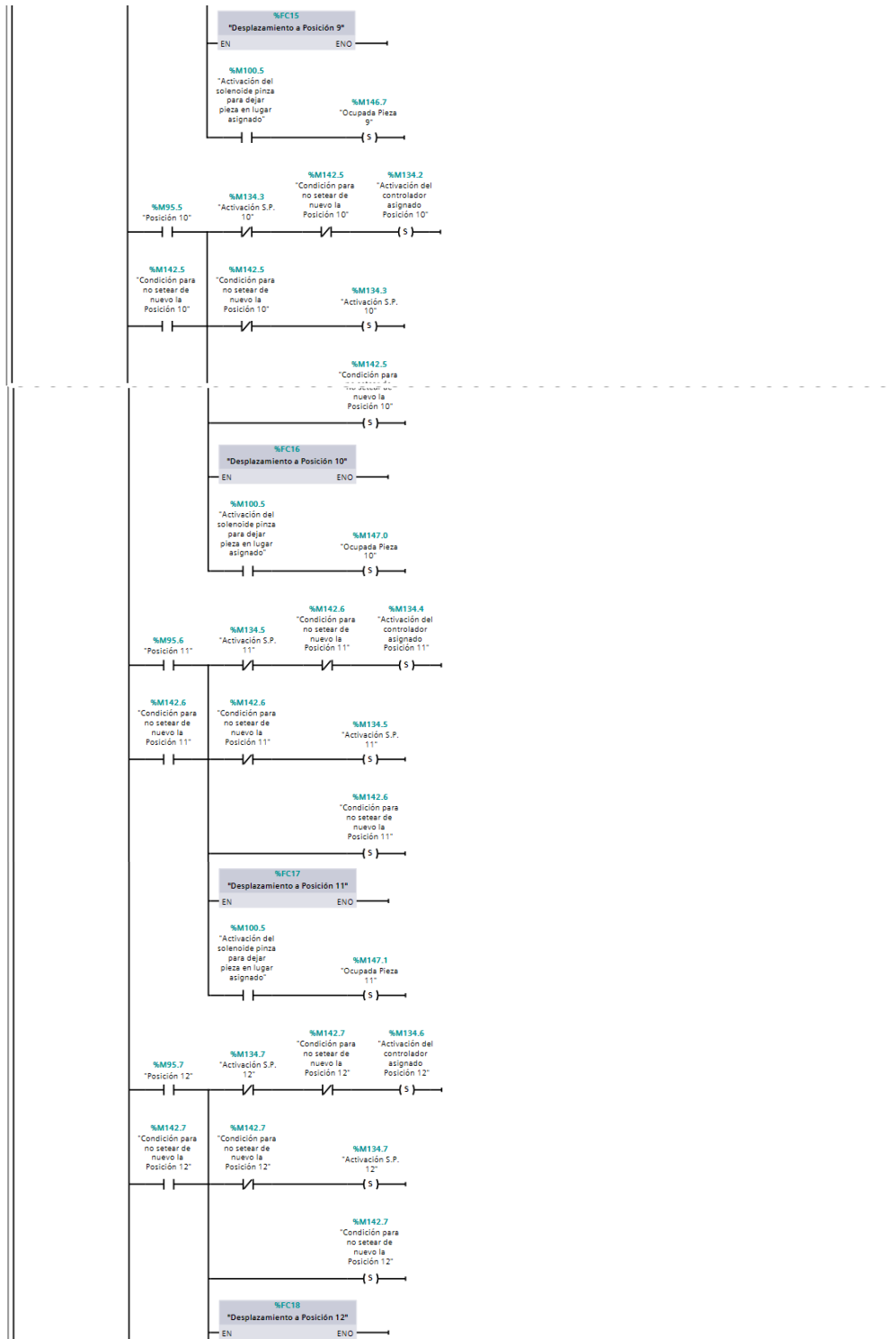
Segmento 28: Elección posición de almacenamiento.

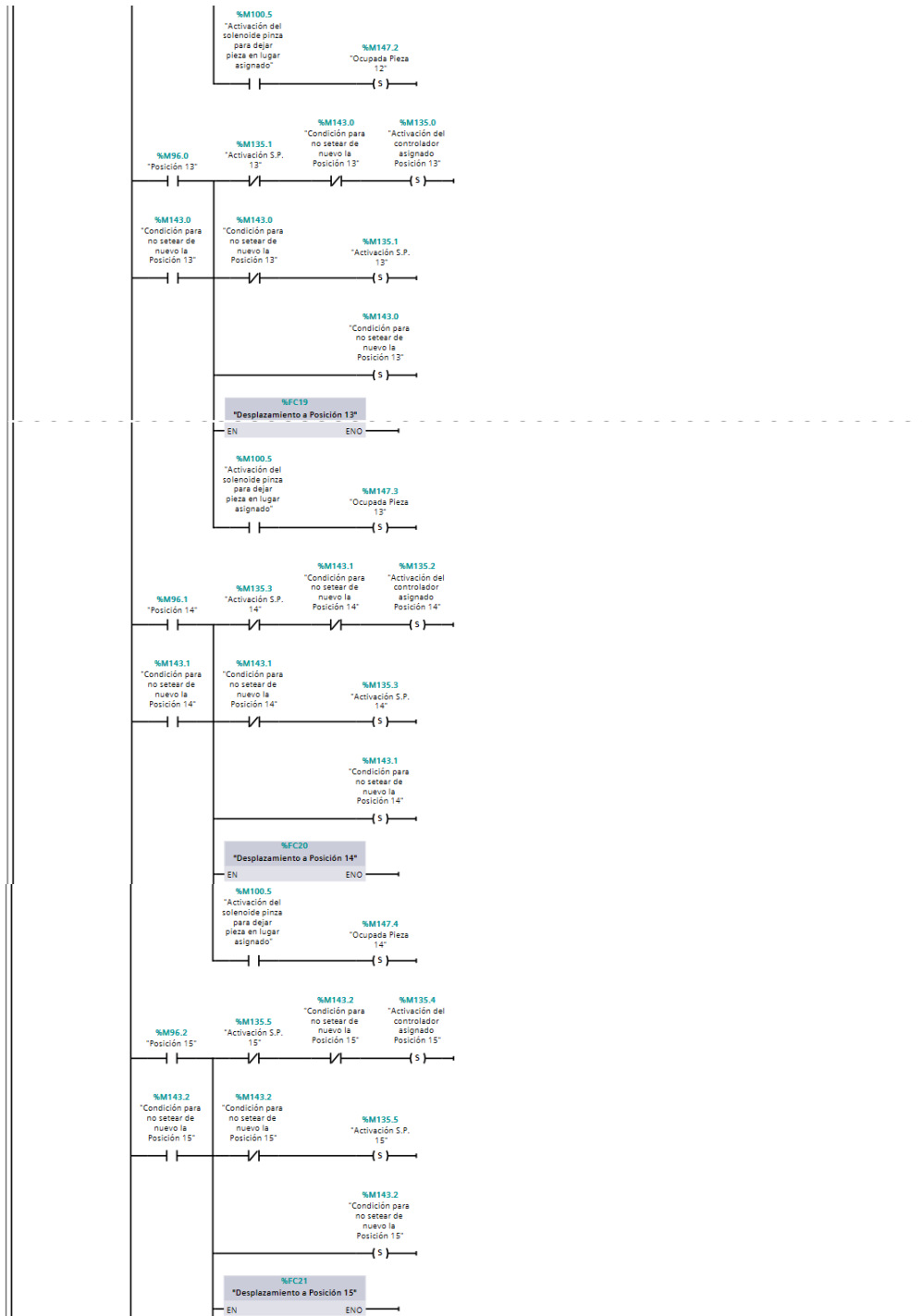
Elección de la posición a almacenar la pieza de trabajo.

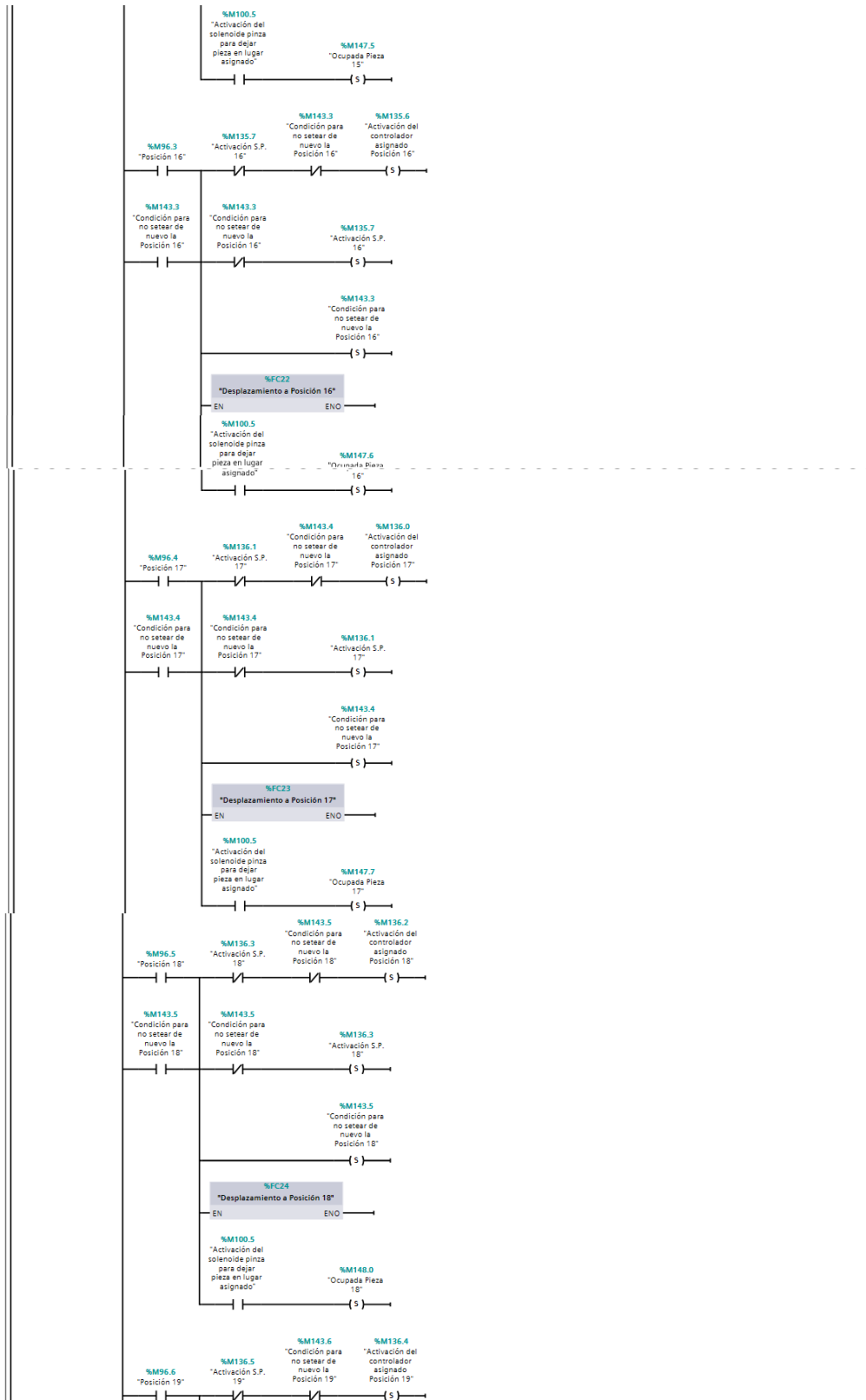


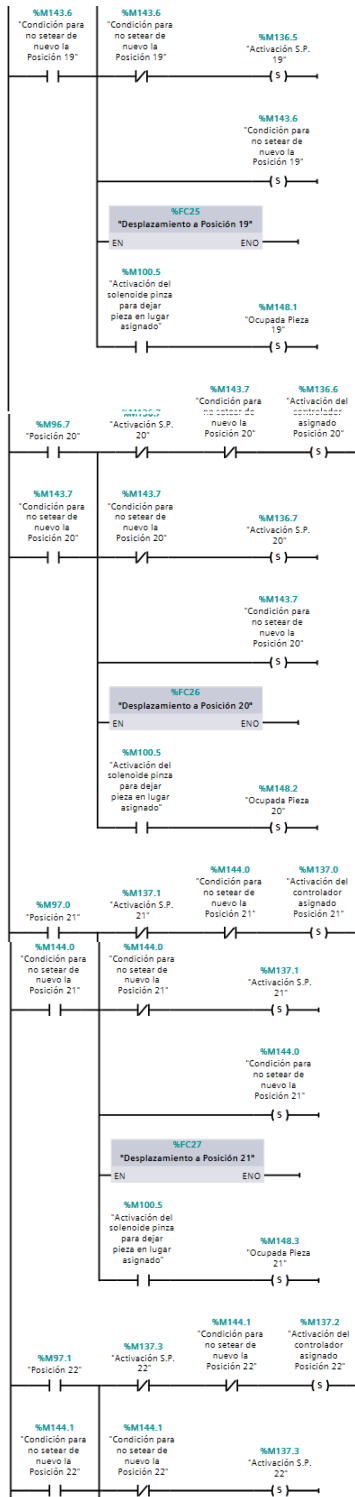


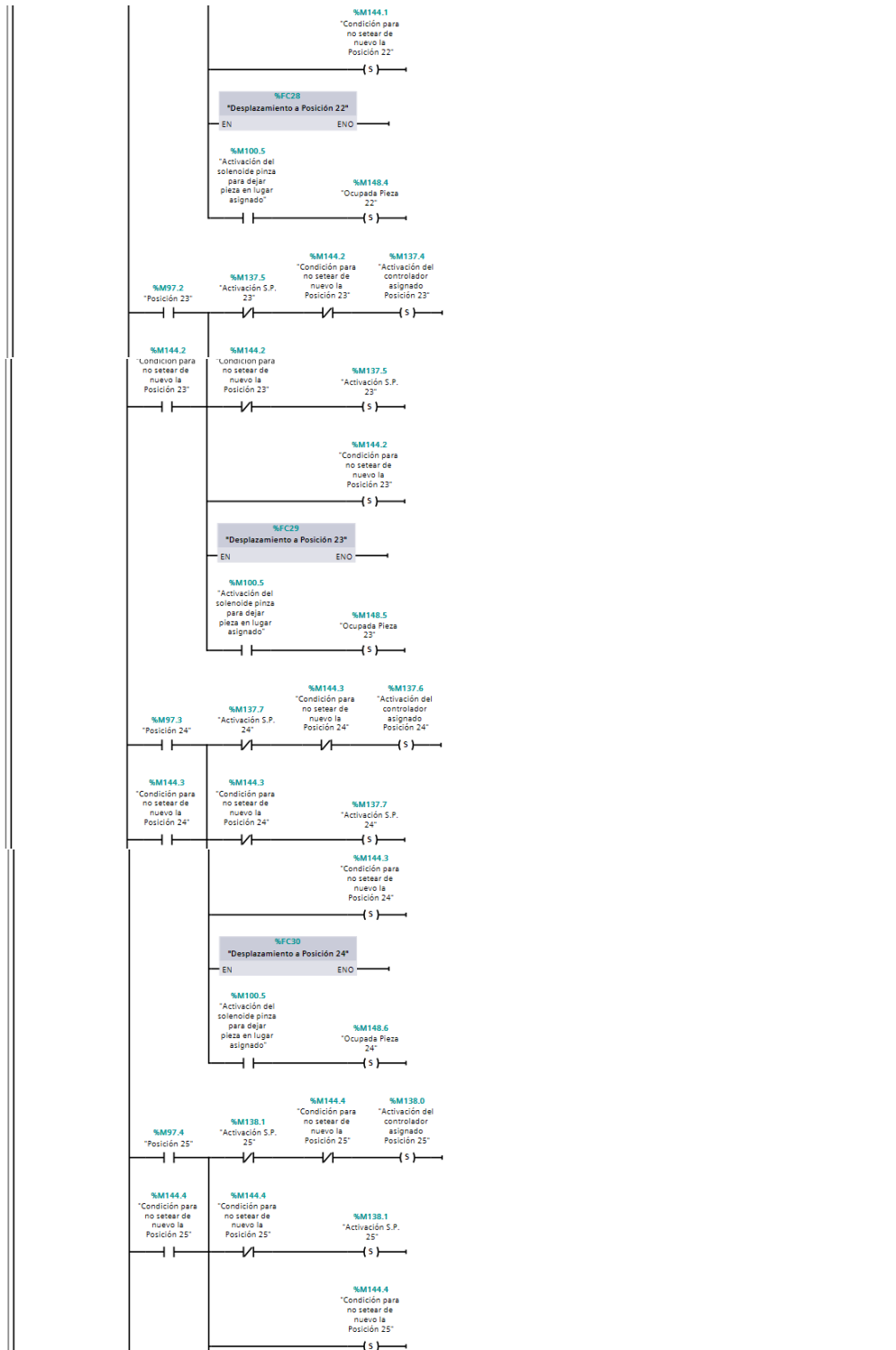


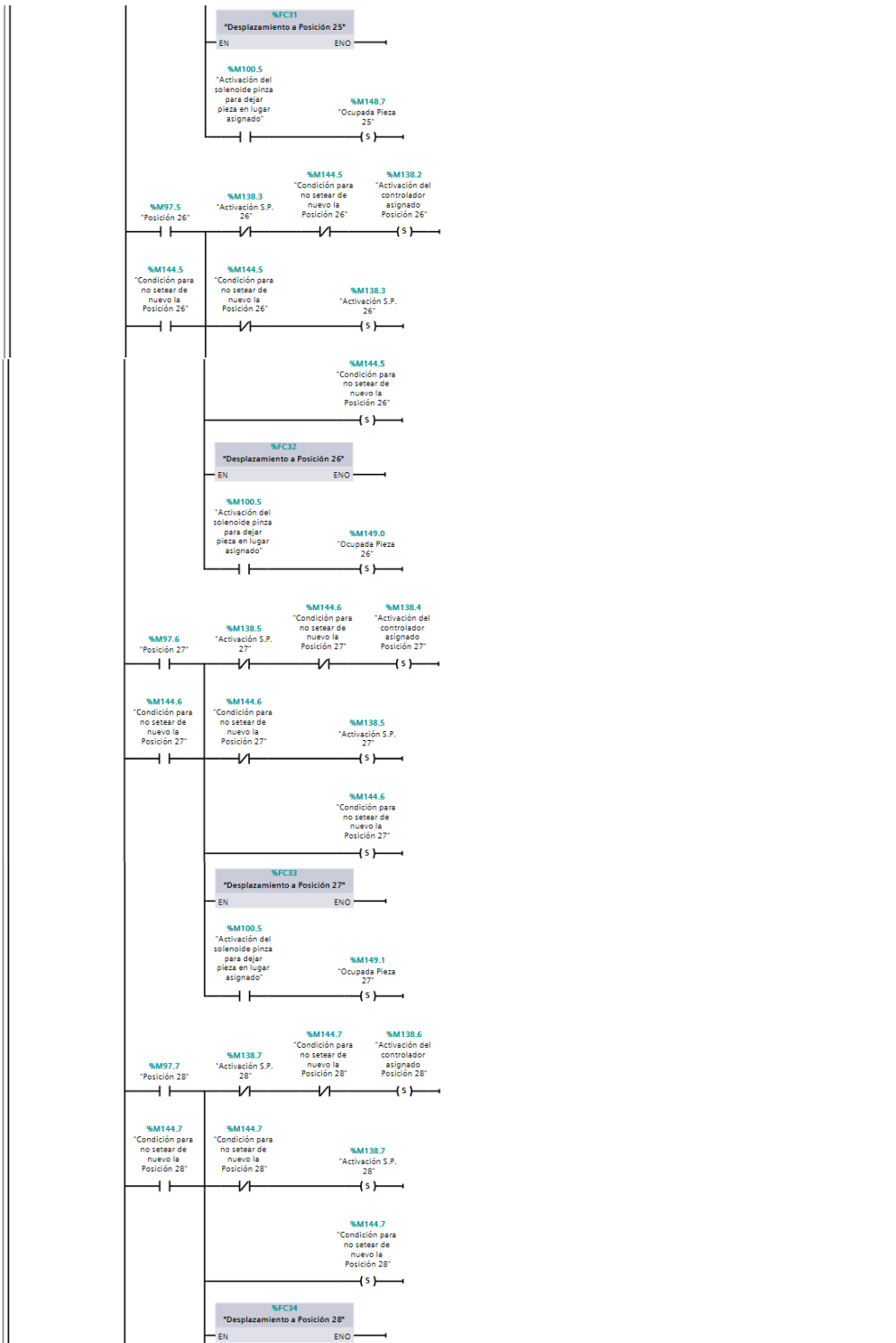


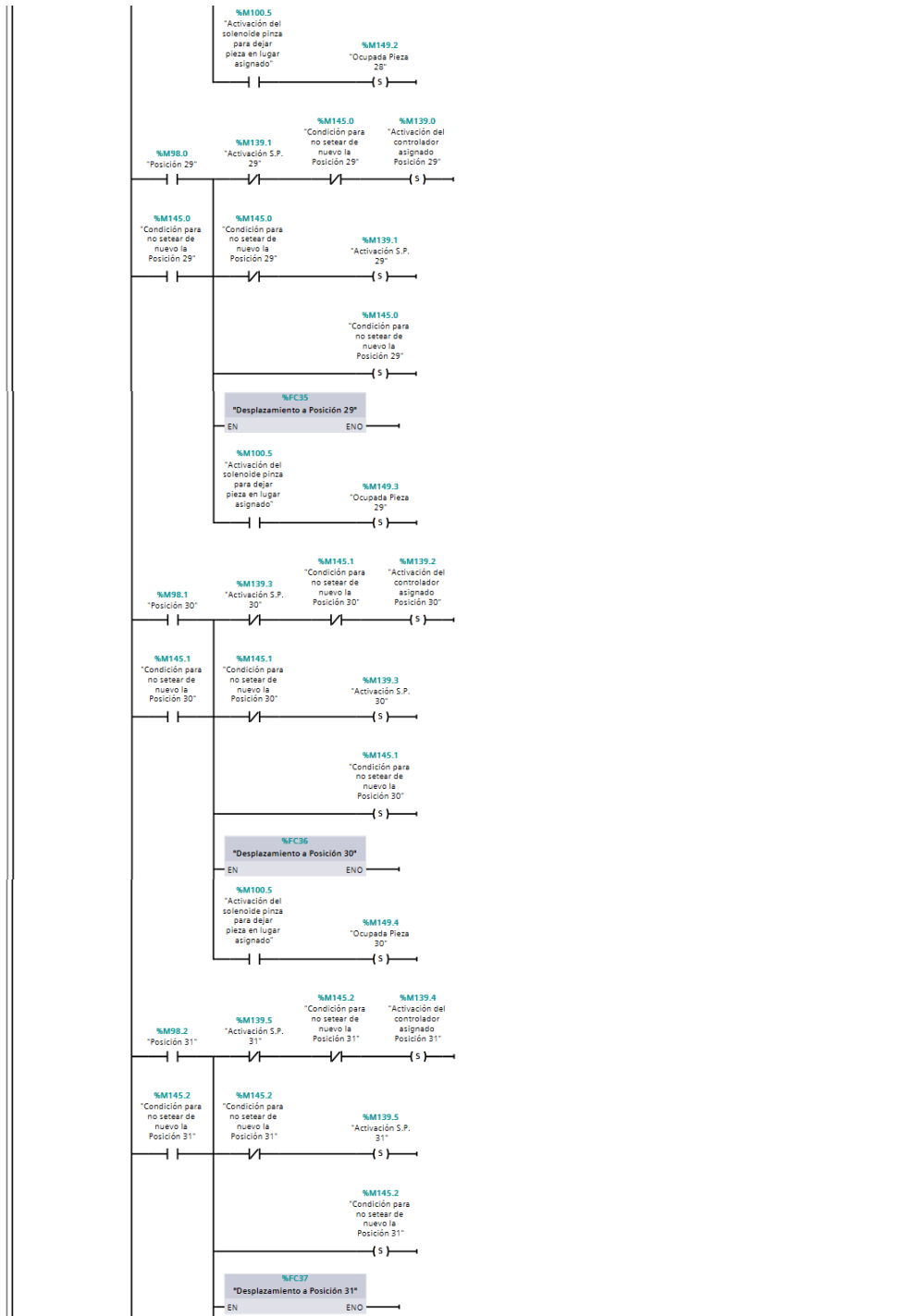


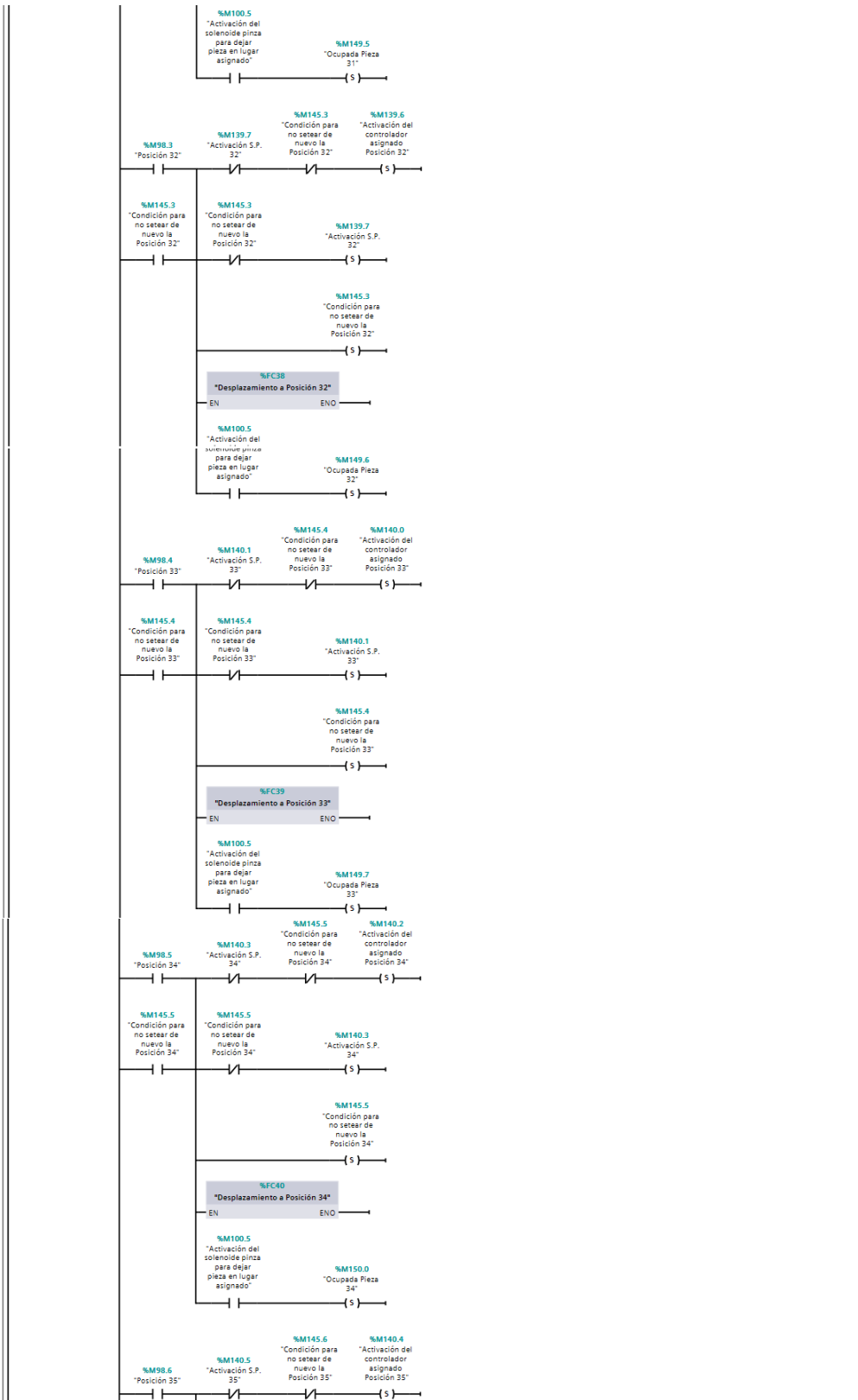


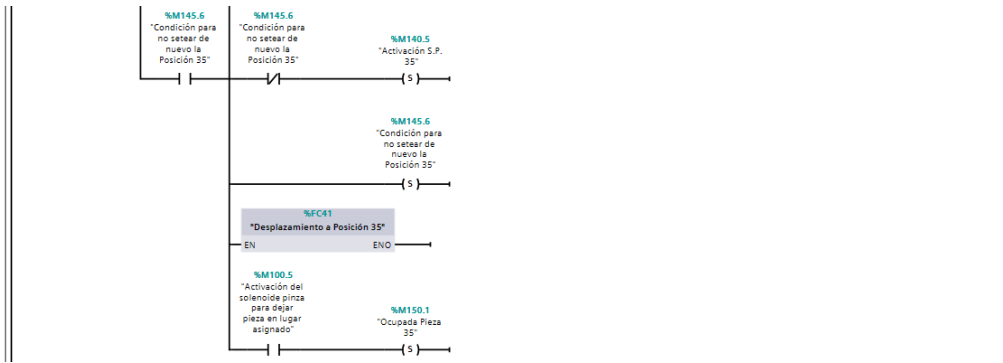






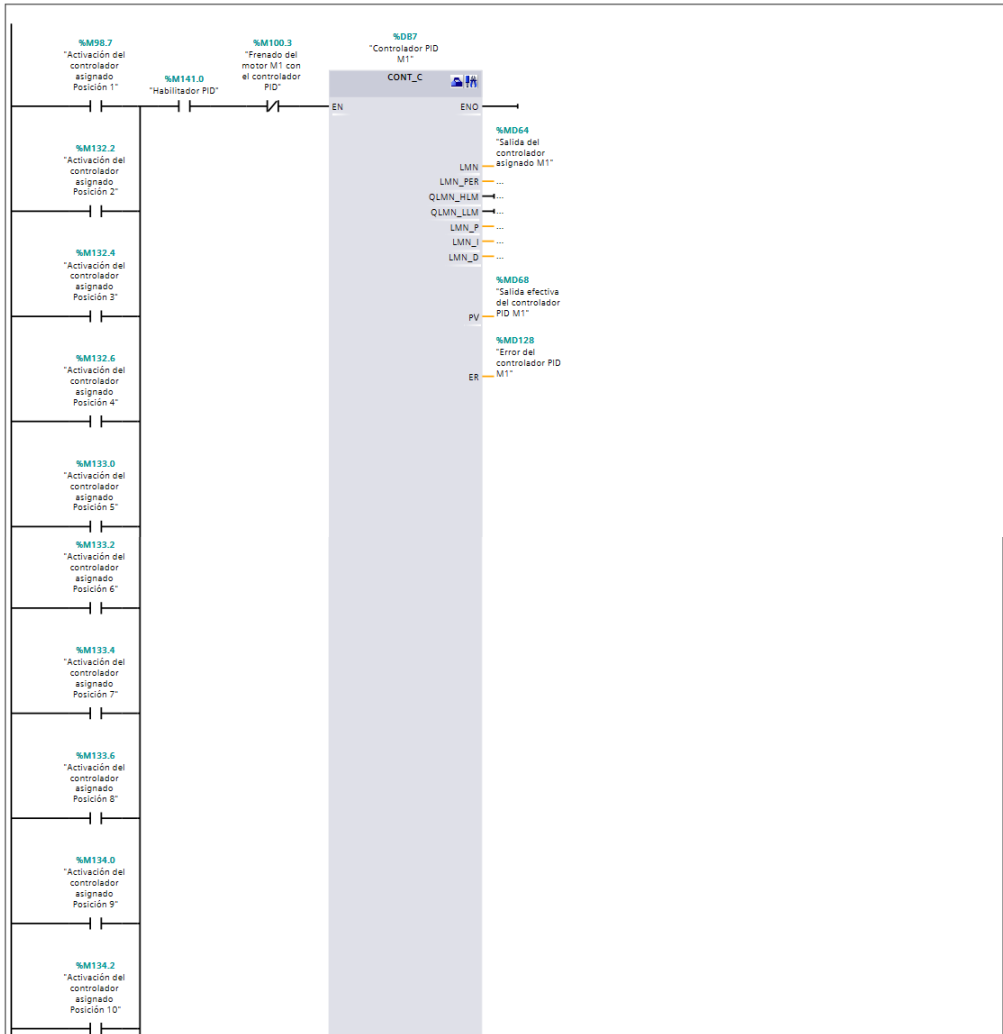




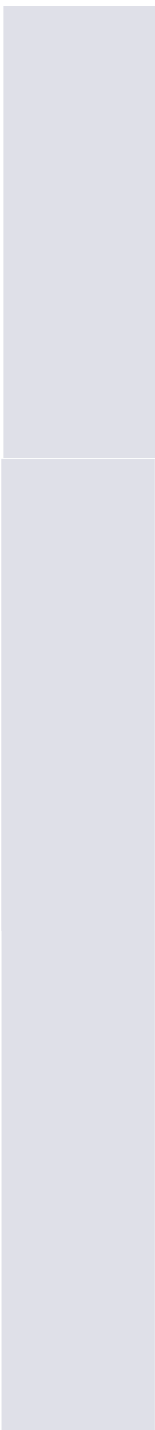


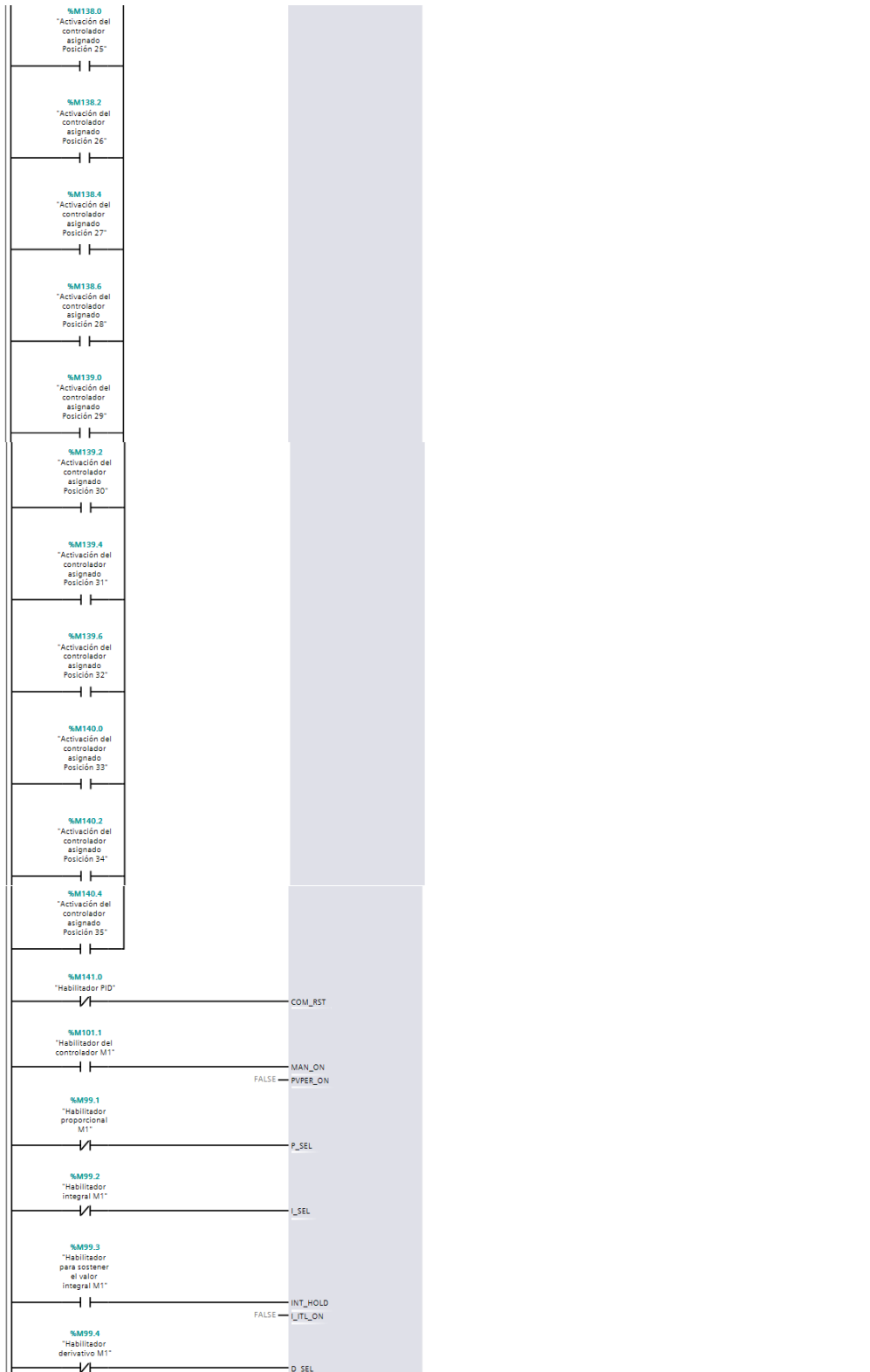
Segmento 29: Controlador PID motor M1.

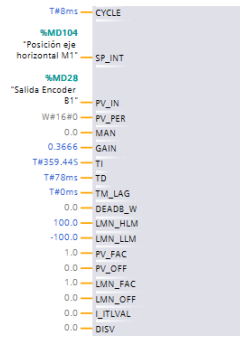
Activación del controlador PID del motor M1.



<p>%M134.4 "Activación del controlador asignado Posición 11"</p>
<p>%M134.6 "Activación del controlador asignado Posición 12"</p>
<p>%M135.0 "Activación del controlador asignado Posición 13"</p>
<p>%M135.2 "Activación del controlador asignado Posición 14"</p>
<p>%M135.4 "Activación del controlador asignado Posición 15"</p>
<p>%M135.6 "Activación del controlador asignado Posición 16"</p>
<p>%M136.0 "Activación del controlador asignado Posición 17"</p>
<p>%M136.2 "Activación del controlador asignado Posición 18"</p>
<p>%M136.4 "Activación del controlador asignado Posición 19"</p>
<p>%M136.6 "Activación del controlador asignado Posición 20"</p>
<p>%M137.0 "Activación del controlador asignado Posición 21"</p>
<p>%M137.2 "Activación del controlador asignado Posición 22"</p>
<p>%M137.4 "Activación del controlador asignado Posición 23"</p>
<p>%M137.6 "Activación del controlador asignado Posición 24"</p>

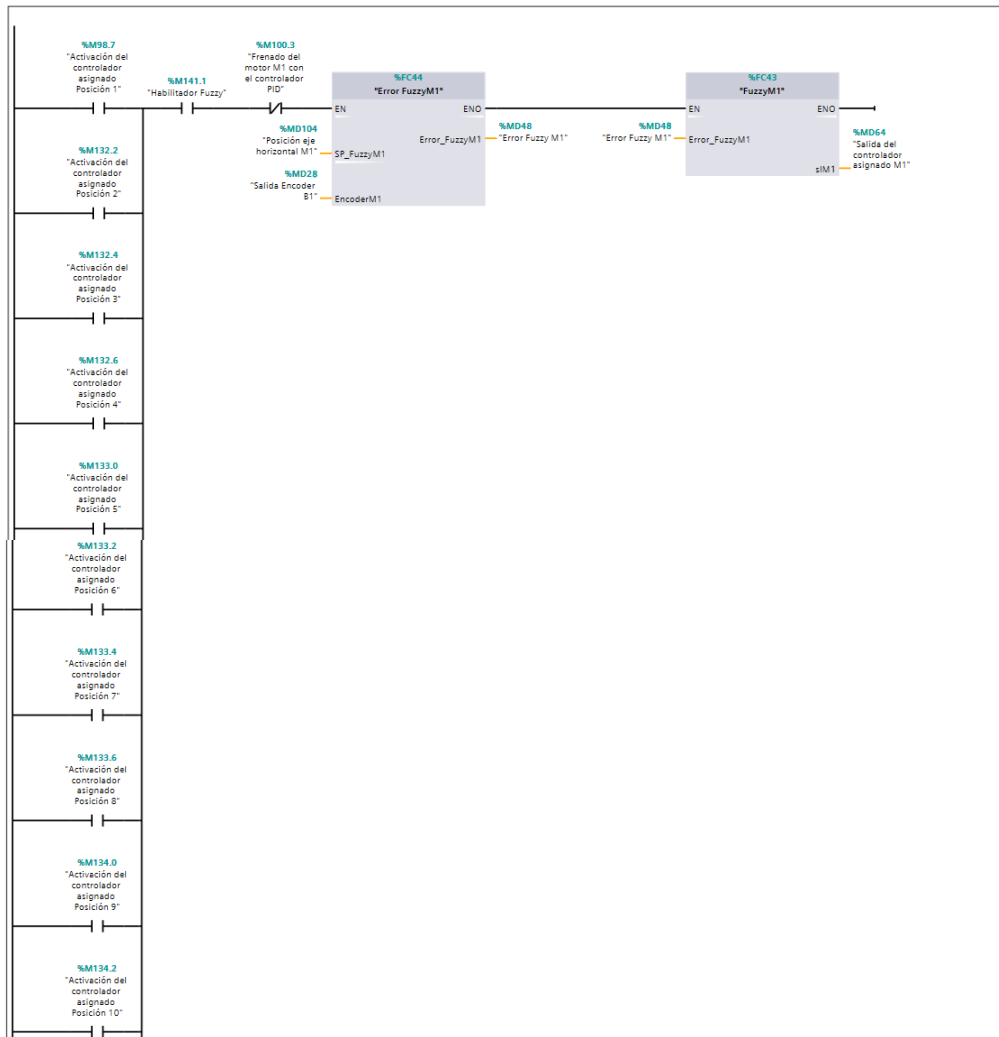






Segmento 30: Controlador Fuzzy motor M1.

Activación del controlador Fuzzy del motor M1.

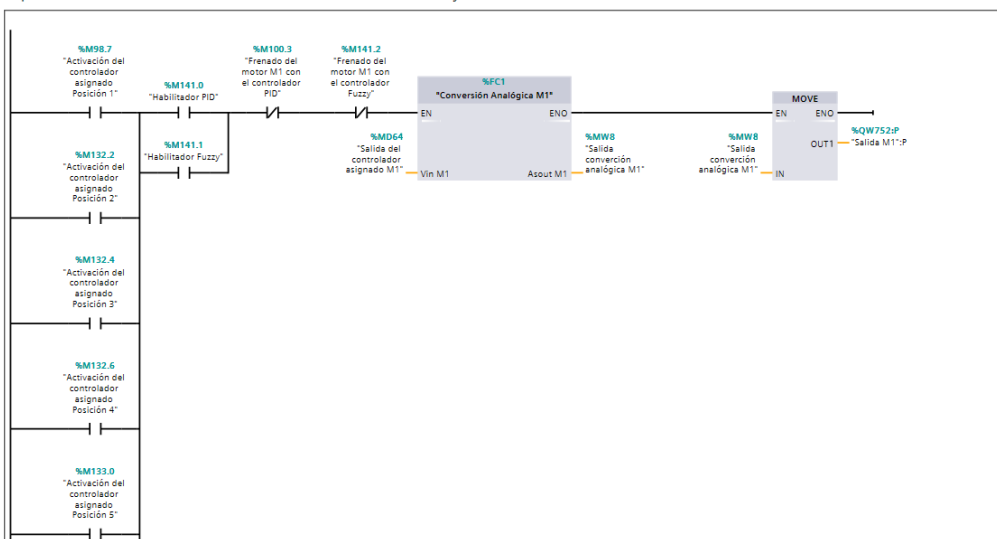


<p>%M134.4 *Activación del controlador asignado Posición 11*</p>
<p>%M134.6 *Activación del controlador asignado Posición 12*</p>
<p>%M135.0 *Activación del controlador asignado Posición 13*</p>
<p>%M135.2 *Activación del controlador asignado Posición 14*</p>
<p>%M135.4 *Activación del controlador asignado Posición 15*</p>
<p>%M135.6 *Activación del controlador asignado Posición 15*</p>
<p>%M136.0 *Activación del controlador asignado Posición 17*</p>
<p>%M136.2 *Activación del controlador asignado Posición 18*</p>
<p>%M136.4 *Activación del controlador asignado Posición 19*</p>
<p>%M136.6 *Activación del controlador asignado Posición 20*</p>
<p>%M137.0 *Activación del controlador asignado Posición 21*</p>
<p>%M137.2 *Activación del controlador asignado Posición 22*</p>
<p>%M137.4 *Activación del controlador asignado Posición 23*</p>
<p>%M137.6 *Activación del controlador asignado Posición 24*</p>

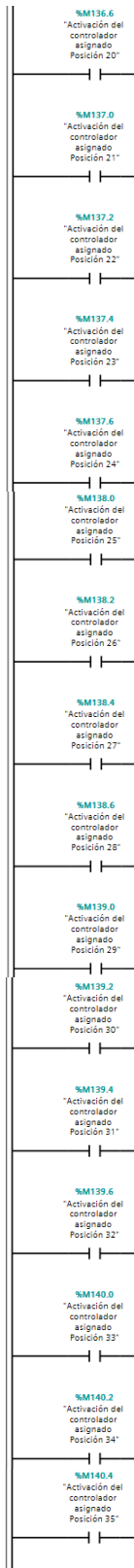


Segmento 31: Arranque motor M1 controlador PID o Fuzzy.

Arranque del motor M1 con el controlador PID o con el controlador Fuzzy.

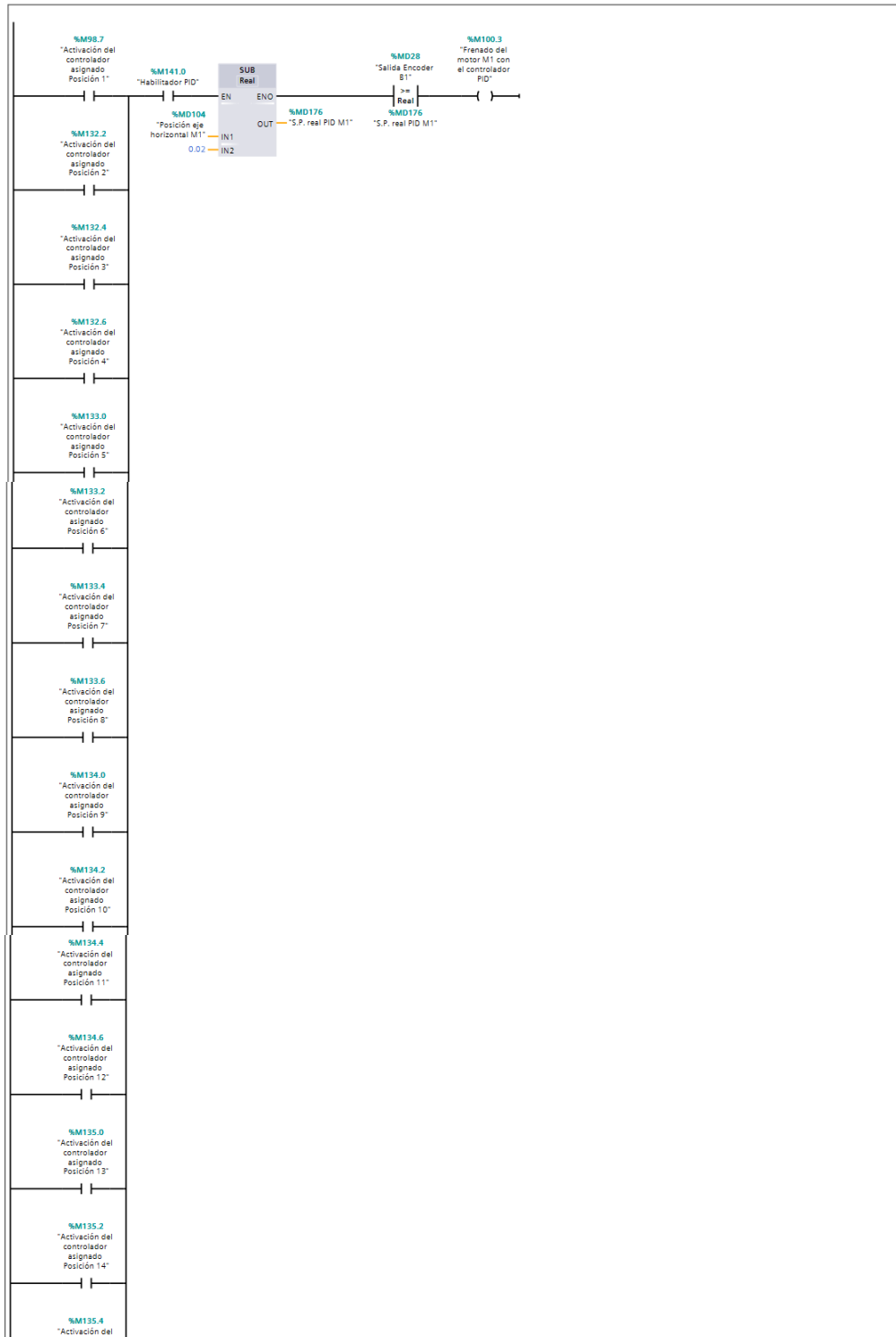


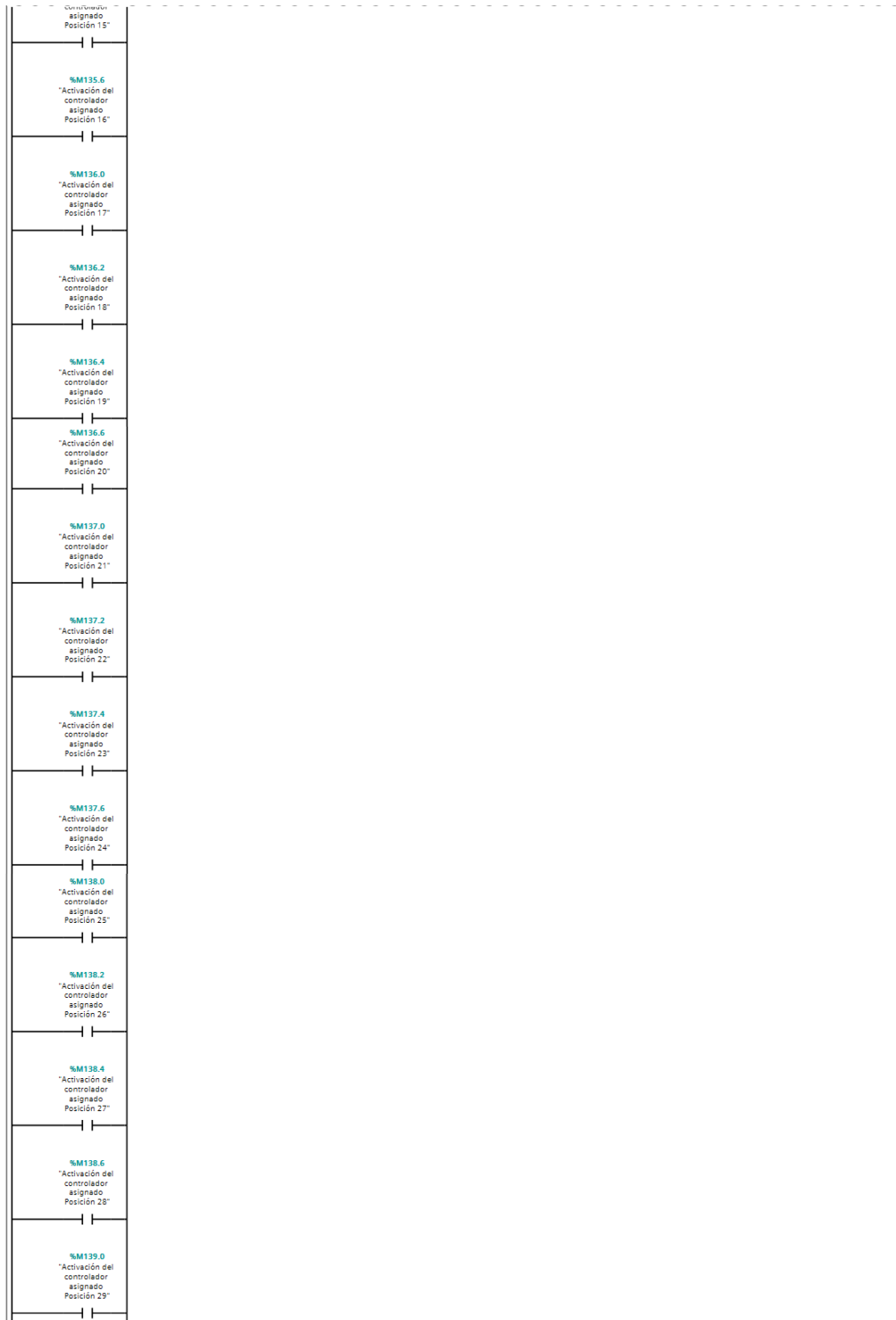
<p>%M133.2 *Activación del controlador asignado Posición 6*</p>
<p>%M133.4 *Activación del controlador asignado Posición 7*</p>
<p>%M133.6 *Activación del controlador asignado Posición 8*</p>
<p>%M134.0 *Activación del controlador asignado Posición 9*</p>
<p>%M134.2 *Activación del controlador asignado Posición 10*</p>
<p>%M134.4 *Activación del controlador asignado Posición 11*</p>
<p>%M134.6 *Activación del controlador asignado Posición 12*</p>
<p>%M135.0 *Activación del controlador asignado Posición 13*</p>
<p>%M135.2 *Activación del controlador asignado Posición 14*</p>
<p>%M135.4 *Activación del controlador asignado Posición 15*</p>
<p>%M135.6 *Activación del controlador asignado Posición 16*</p>
<p>%M136.0 *Activación del controlador asignado Posición 17*</p>
<p>%M136.2 *Activación del controlador asignado Posición 18*</p>
<p>%M136.4 *Activación del controlador asignado Posición 19*</p>

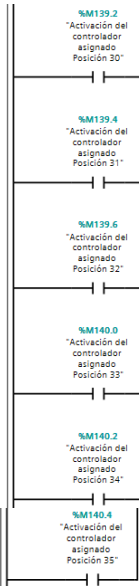


Segmento 32: Condición frenado motor M1 PID.

Condición de frenado de motor M1 con el controlador PID.

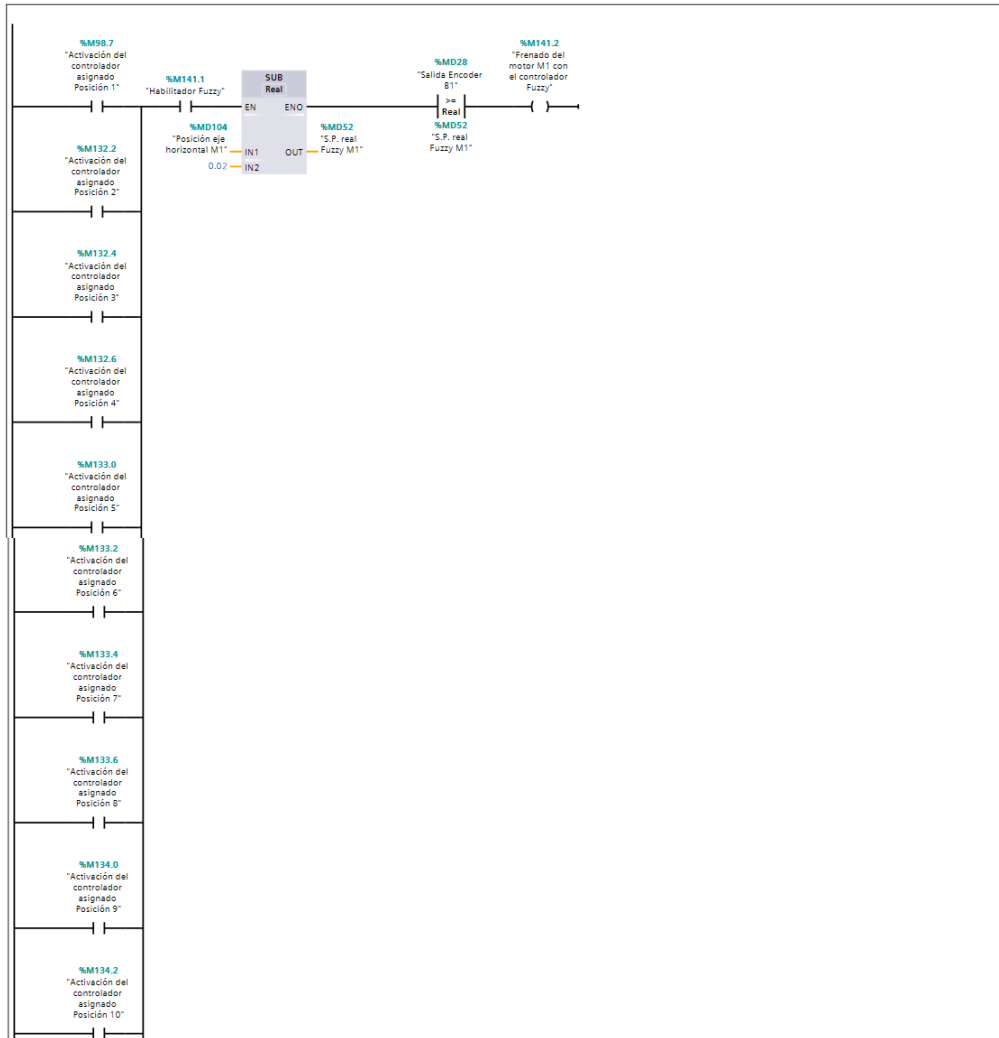


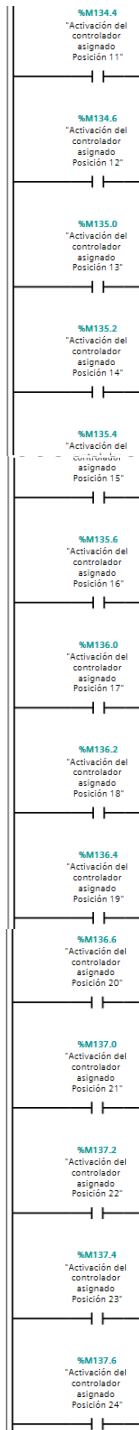




Segmento 33: Condición frenado motor M1 Fuzzy.

Condición de frenado de motor M1 con el controlador Fuzzy.

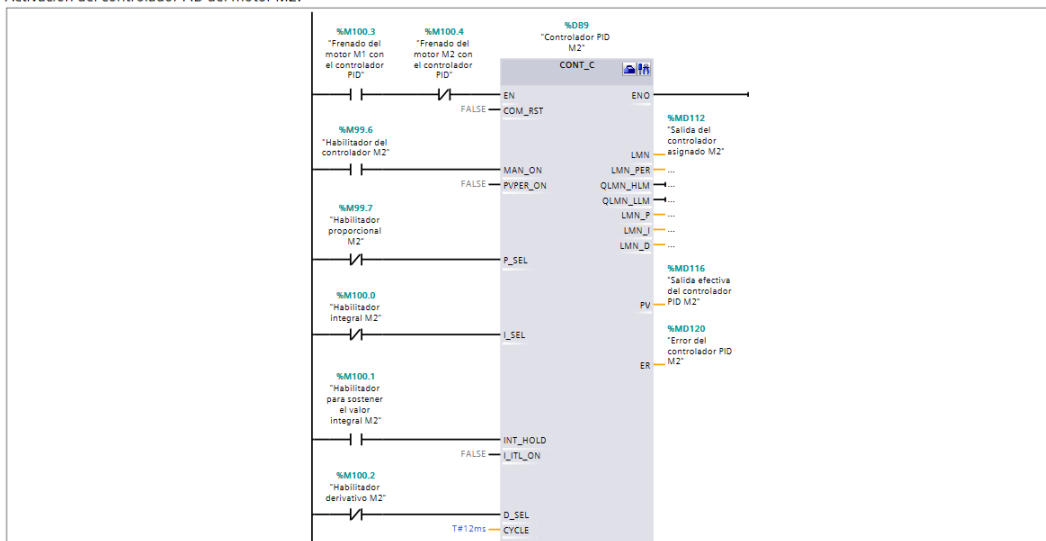


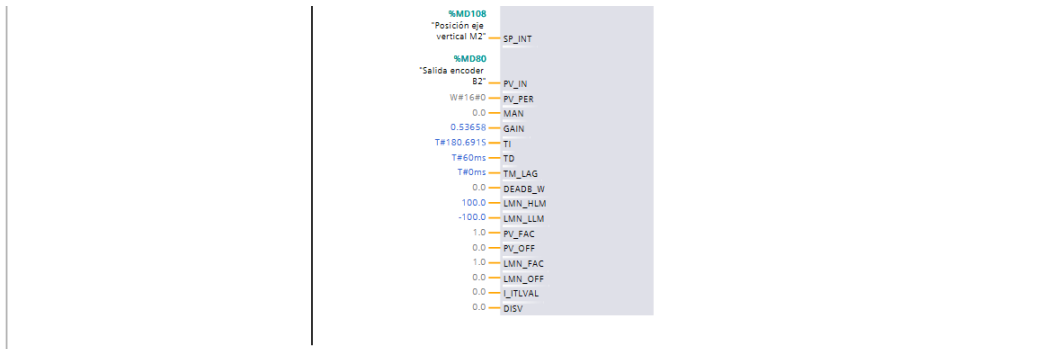




Segmento 34: Controlador PID motor M2.

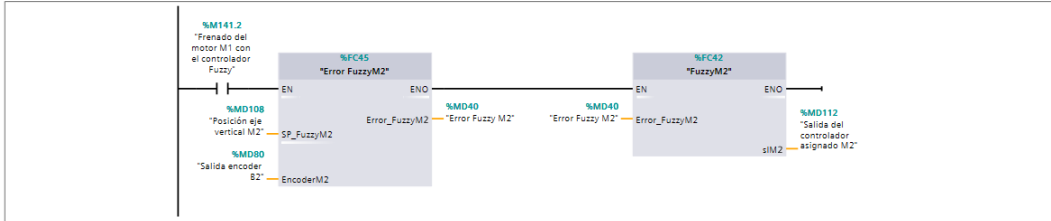
Activación del controlador PID del motor M2.





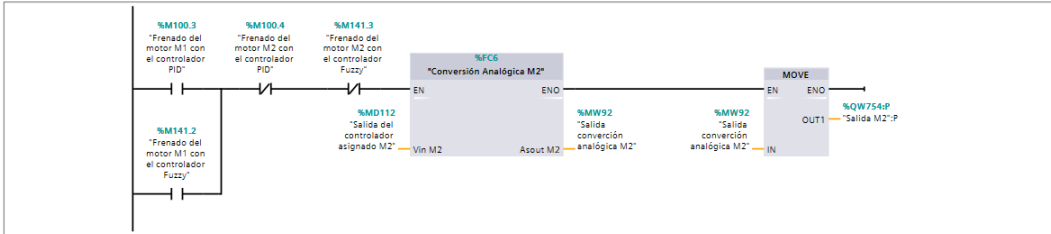
Segmento 35: Controlador Fuzzy motor M2.

Activación del controlador Fuzzy del motor M2.



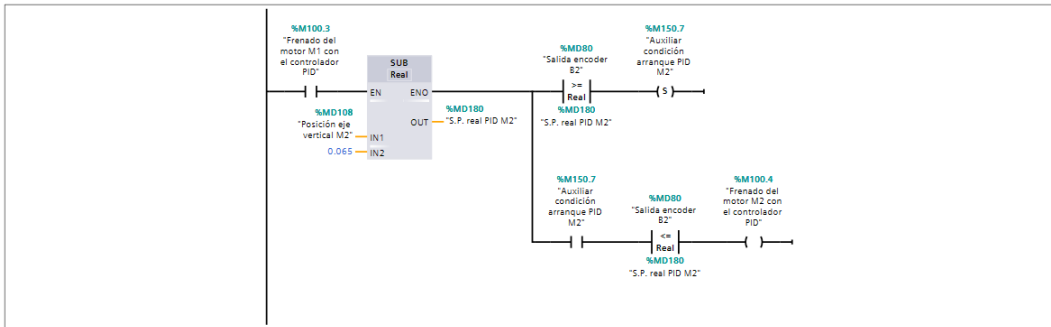
Segmento 36: Arranque motor M2 controlador PID o Fuzzy.

Arranque del motor M2 con el controlador PID.



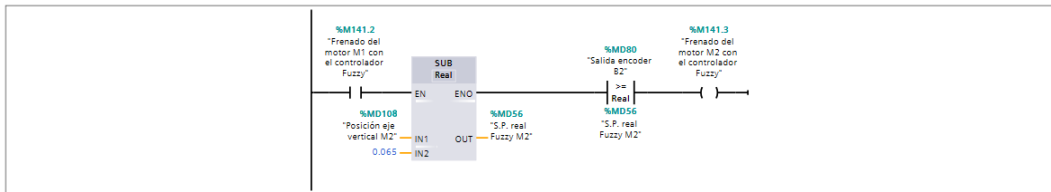
Segmento 37: Condición frenado motor M2 PID.

Condición de frenado de motor M2 con el controlador PID.



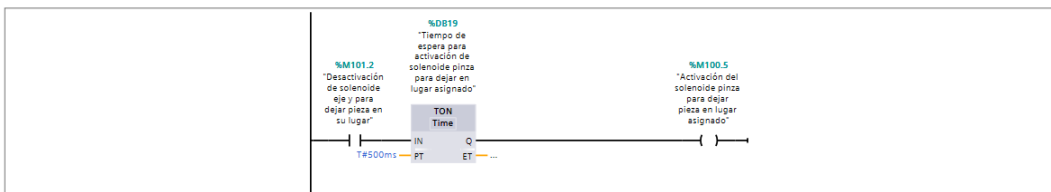
Segmento 38: Condición frenado motor M2 Fuzzy.

Condición de frenado de motor M2 con el controlador Fuzzy.



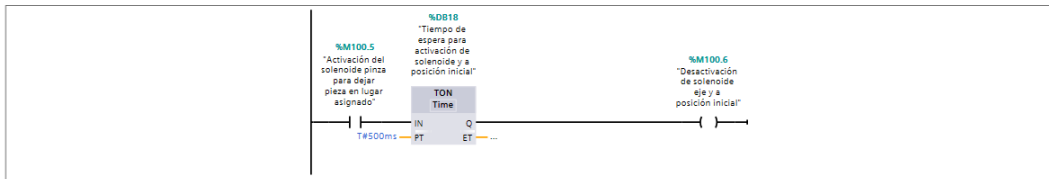
Segmento 39: Tiempo de espera activación solenoide pinza.

Tiempo de espera para activación de solenoide pinza para dejar en lugar asignado.



Segmento 40: Tiempo de espera activación solenoide a posición inicial.

Tiempo de espera para activación de solenoide y a posición inicial.



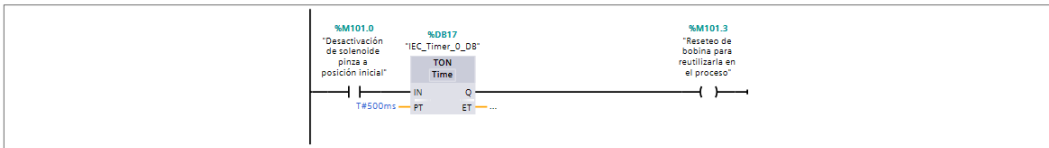
Segmento 41: Desactivación solenoide pinza a posición inicial.

Desactivación del solenoide de pinza a posición inicial.



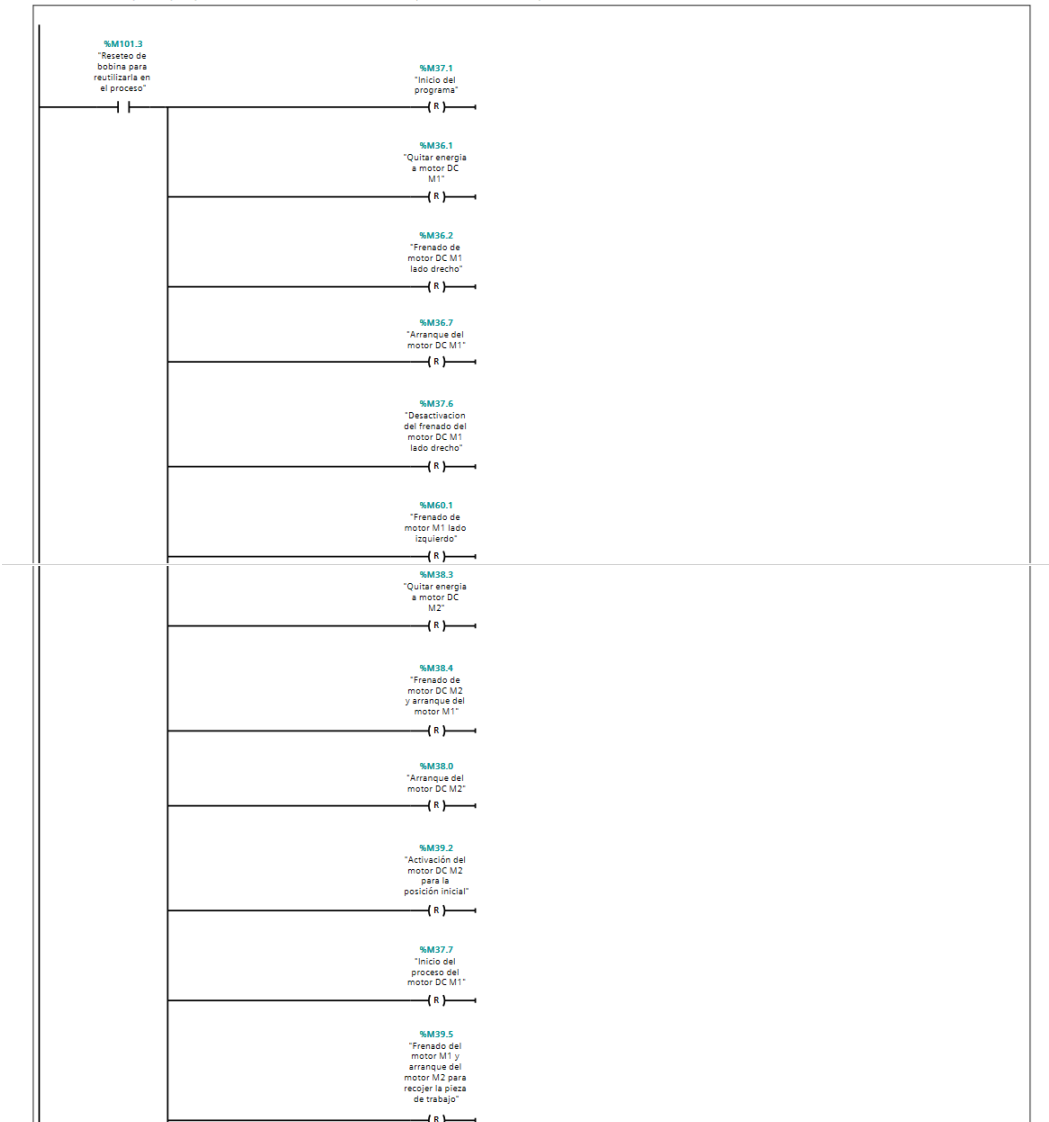
Segmento 42: Tiempo de espera reseteo de bobina solenoide eje y.

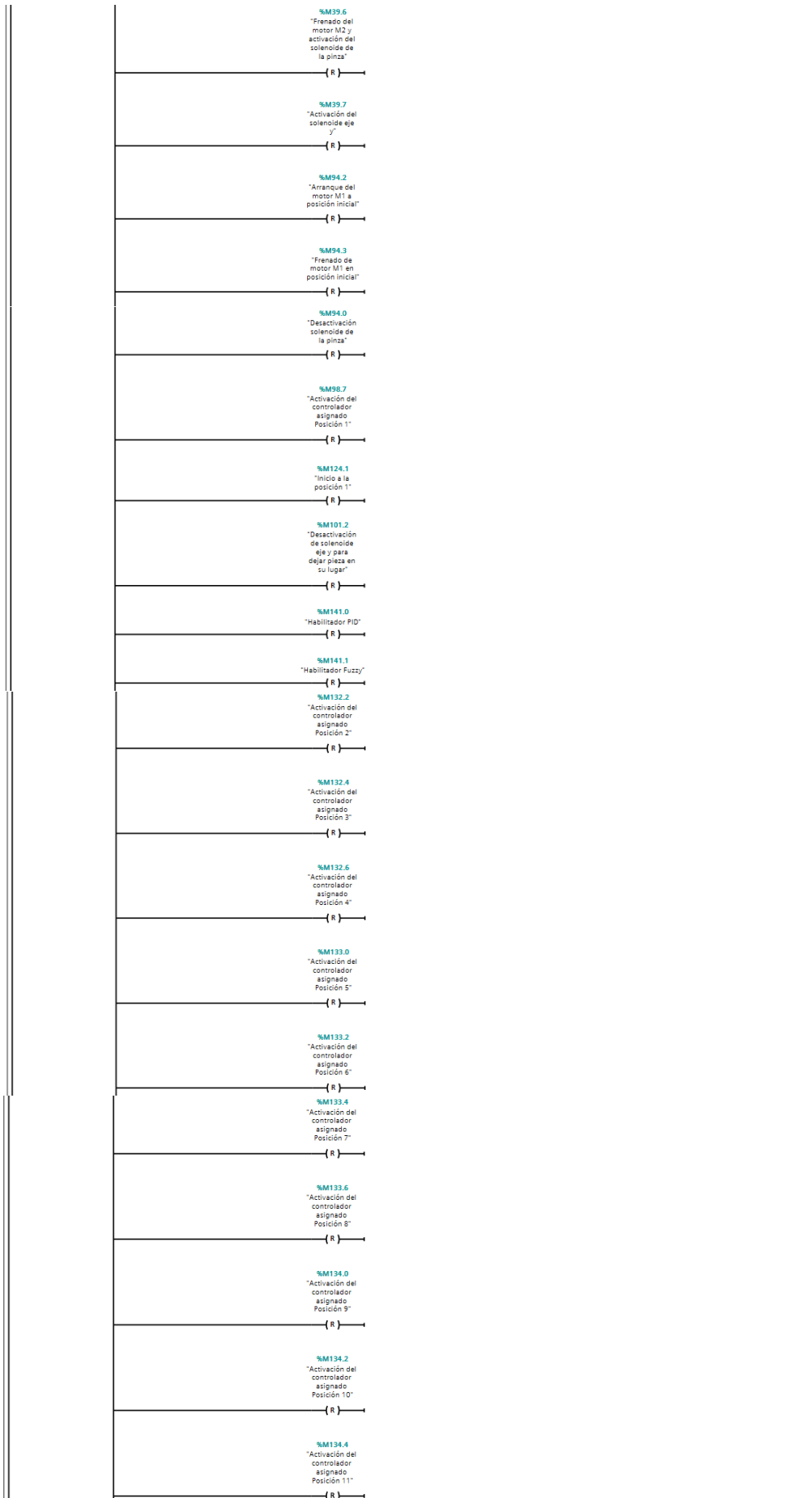
Tiempo de espera para resetear bobina del solenoide del eje y.

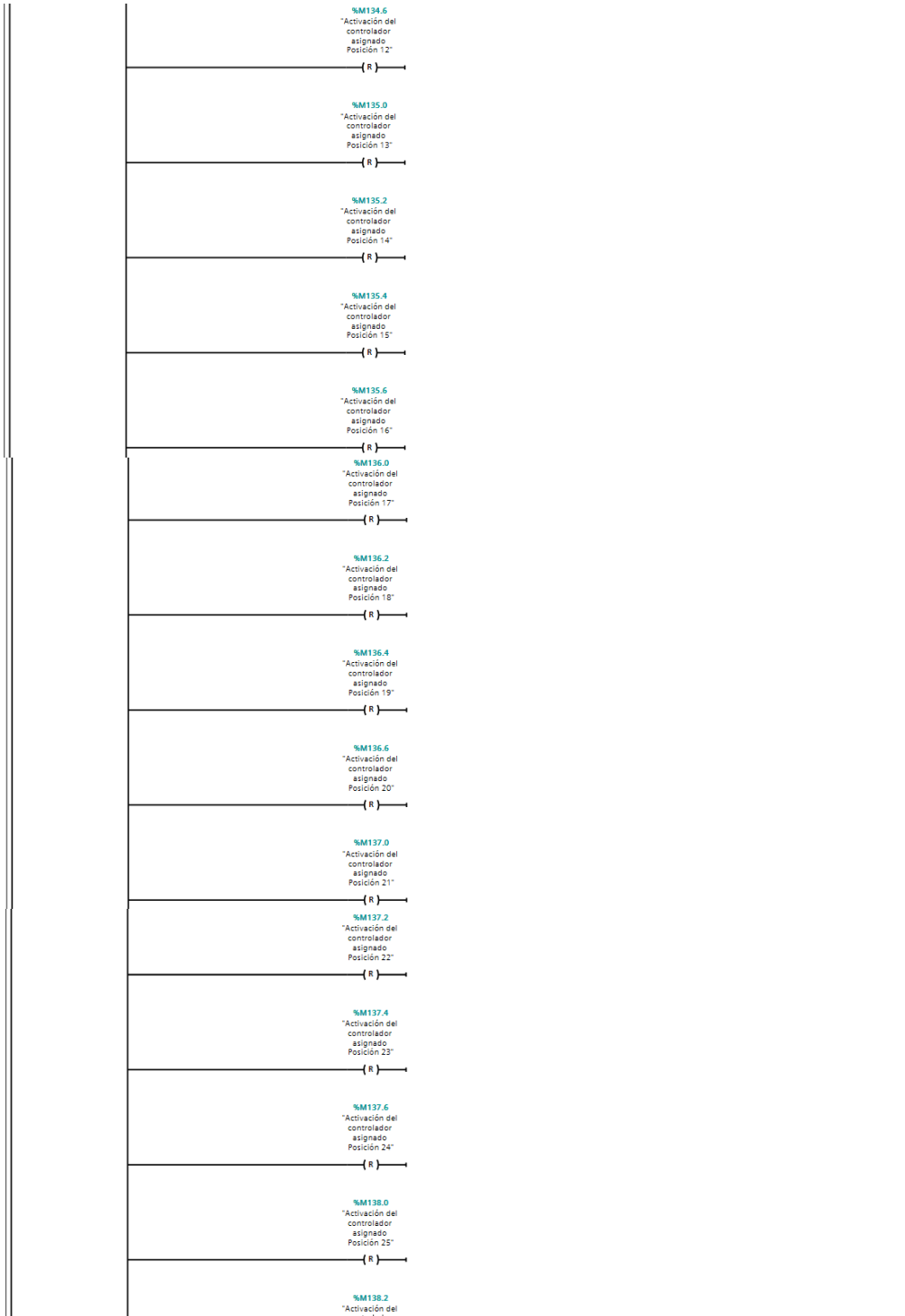


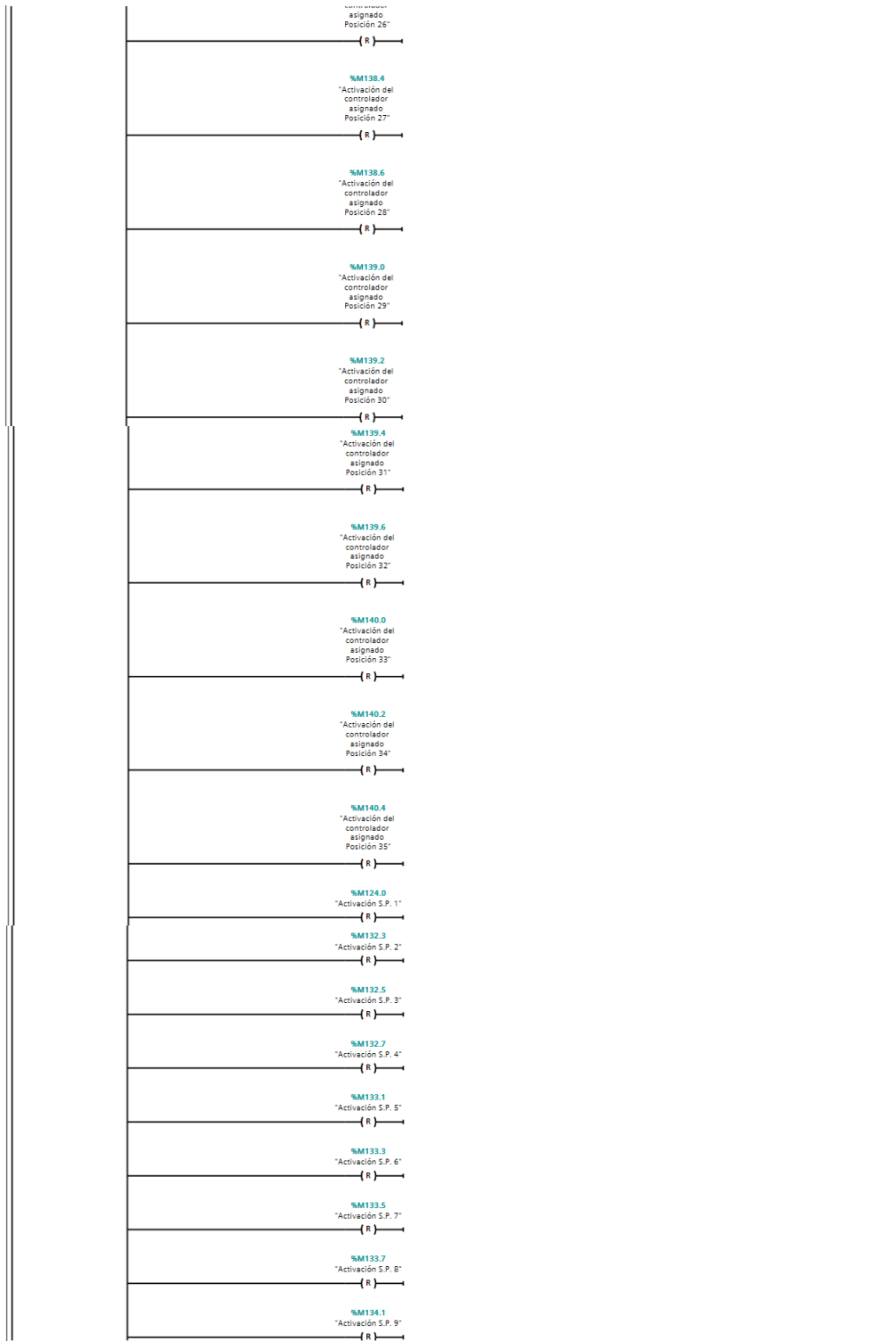
Segmento 43: Reseteo de bobinas para nuevo proceso.

Reseteo de bobina para que pueda ser utilizada nuevamente sin pulsar el botón de paro.

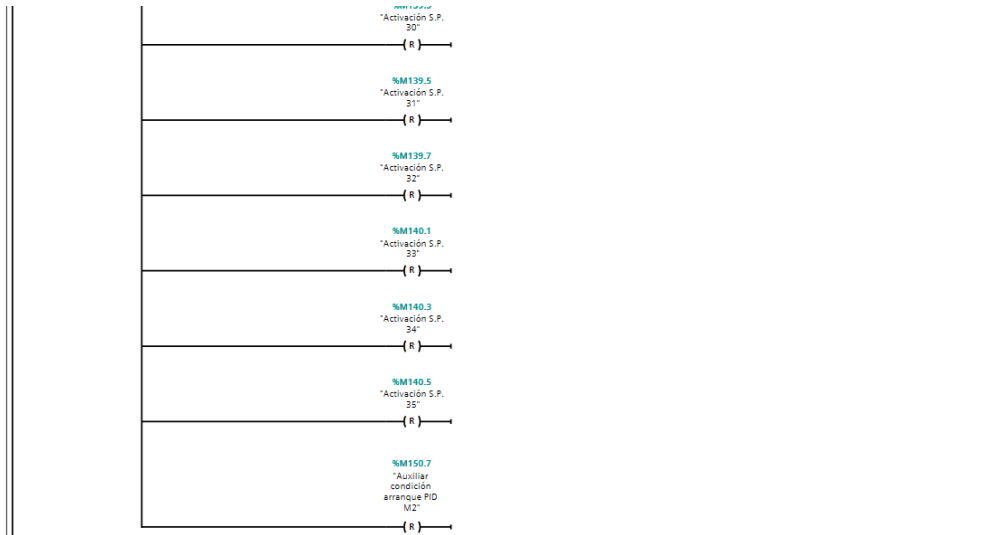






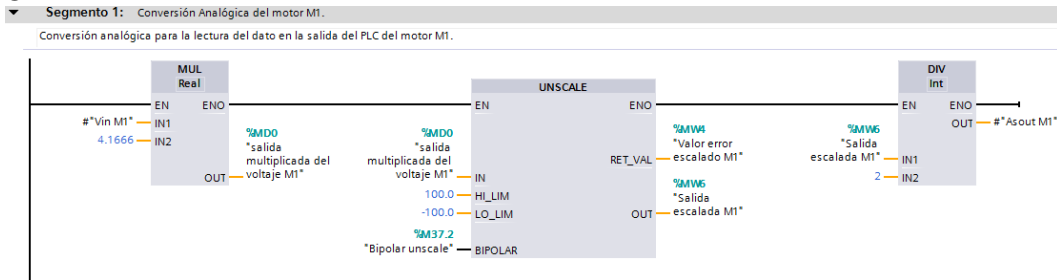


	%M134.3 "Activación S.P. 10"	(R)
	%M134.5 "Activación S.P. 11"	(R)
	%M134.7 "Activación S.P. 12"	(R)
	%M135.1 "Activación S.P. 13"	(R)
	%M135.3 "Activación S.P. 14"	(R)
	%M135.5 "Activación S.P. 15"	(R)
	%M135.7 "Activación S.P. 16"	(R)
	%M136.1 "Activación S.P. 17"	(R)
	%M136.3 "Activación S.P. 18"	(R)
	%M136.5 "Activación S.P. 19"	(R)
	%M136.7 "Activación S.P. 20"	(R)
	%M137.1 "Activación S.P. 21"	(R)
	%M137.3 "Activación S.P. 22"	(R)
	%M137.5 "Activación S.P. 23"	(R)
	%M137.7 "Activación S.P. 24"	(R)
	%M138.1 "Activación S.P. 25"	(R)
	%M138.3 "Activación S.P. 26"	(R)
	%M138.5 "Activación S.P. 27"	(R)
	%M138.7 "Activación S.P. 28"	(R)
	%M139.1 "Activación S.P. 29"	(R)

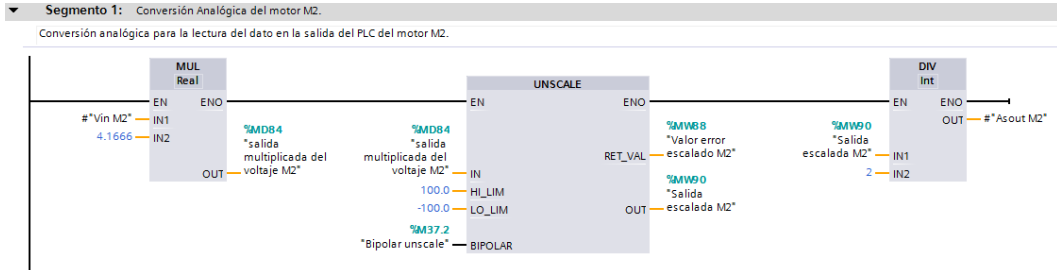


BLOQUES DE FUNCIONES (FC) – DEL MAIN

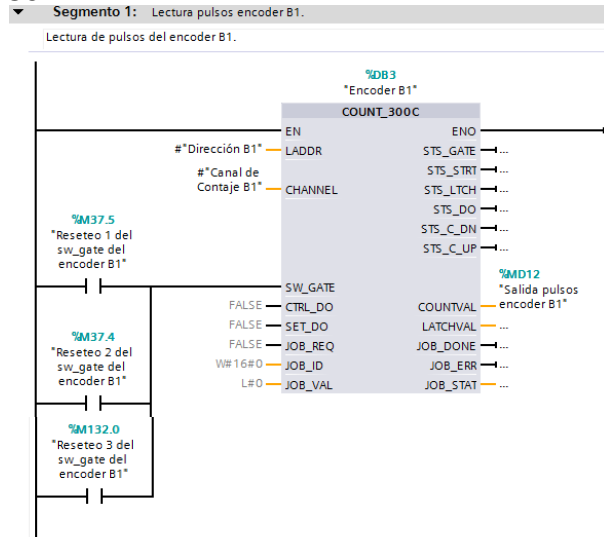
FC1.



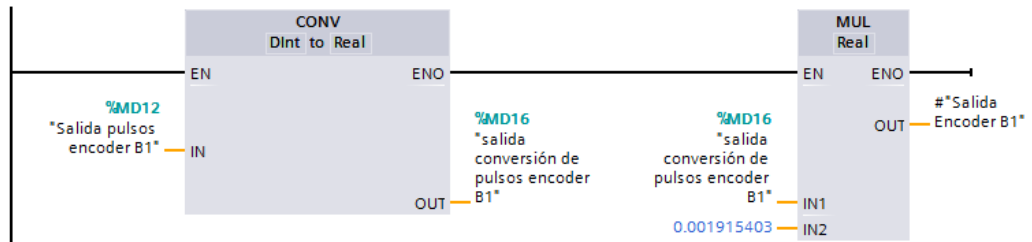
FC2.



FC3.

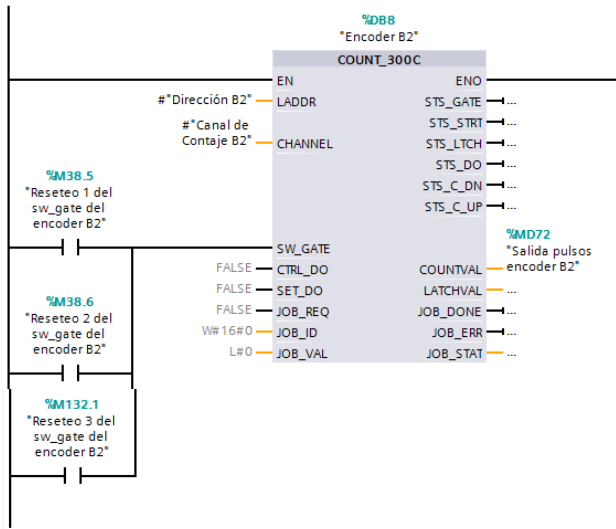


Segmento 2: Conversión pulsos a centímetros encoder B1.
 Conversión de pulsos obtenidos a centímetros del encoder B1.



FC4.

Segmento 1: Lectura pulsos encoder B2.
 Lectura de pulsos del encoder B2.



DefuzificacionFuzzyM1

//Cálculo de la salida defusificada del motor M1

$$\#sIM1 := ((\#alpha1M1 * \#u1M1) + (\#alpha2M1 * \#u2M1)) / (\#alpha1M1 + \#alpha2M1);$$

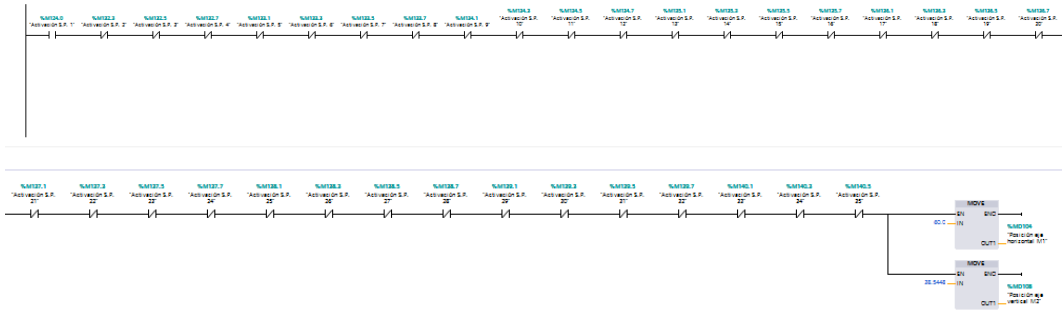
DefuzificacionFuzzyM2

//Cálculo de la salida defusificada del motor M2

$$\#sIM2 := ((\#alpha1M2 * \#u1M2) + (\#alpha2M2 * \#u2M2)) / (\#alpha1M2 + \#alpha2M2);$$

FC5.

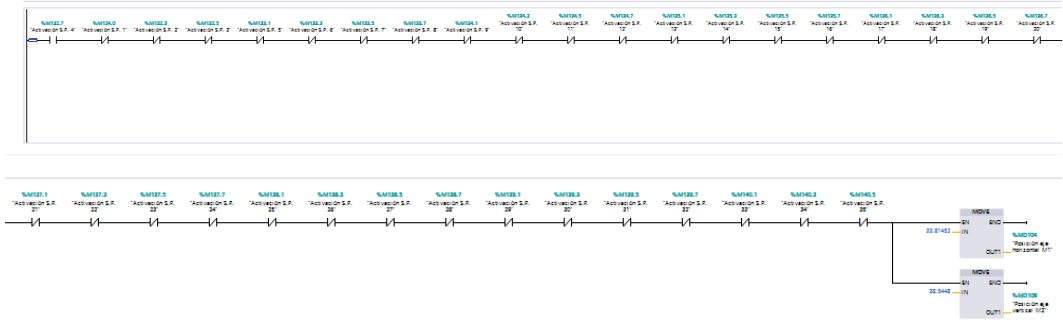
Segmento 1: Activación posición 1.
 Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 1.



FC6.

Segmento 1: Activación posición 4.

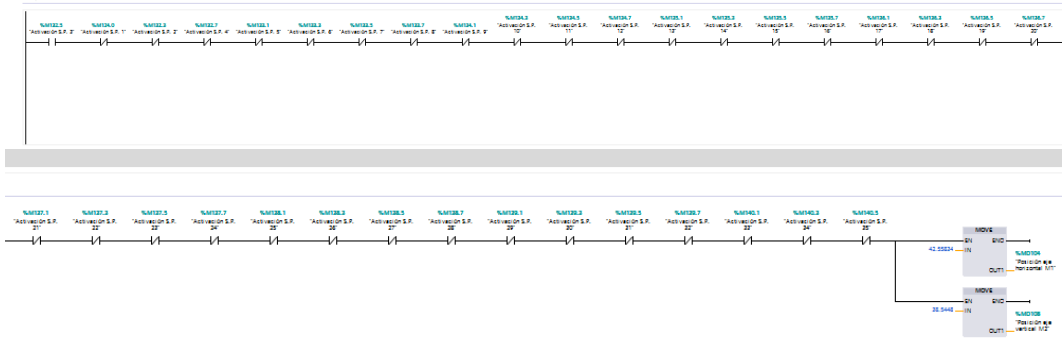
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 4.



FC7.

Segmento 1: Activación posición 3.

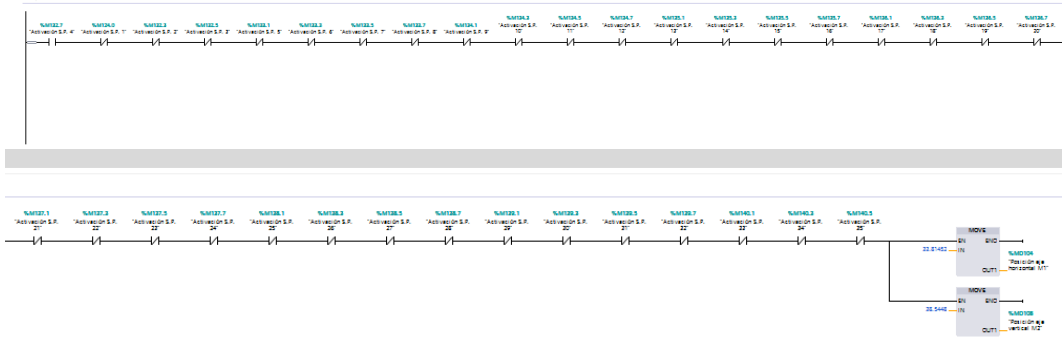
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 3.



FC8.

Segmento 1: Activación posición 4.

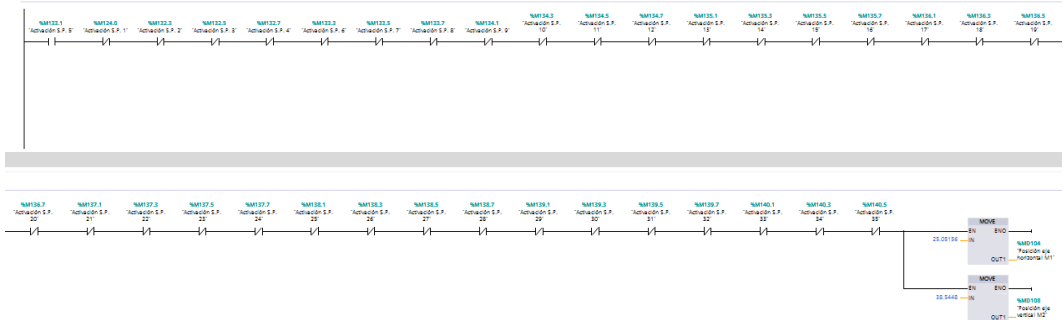
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 4.



FC9.

Segmento 1: Activación posición 5.

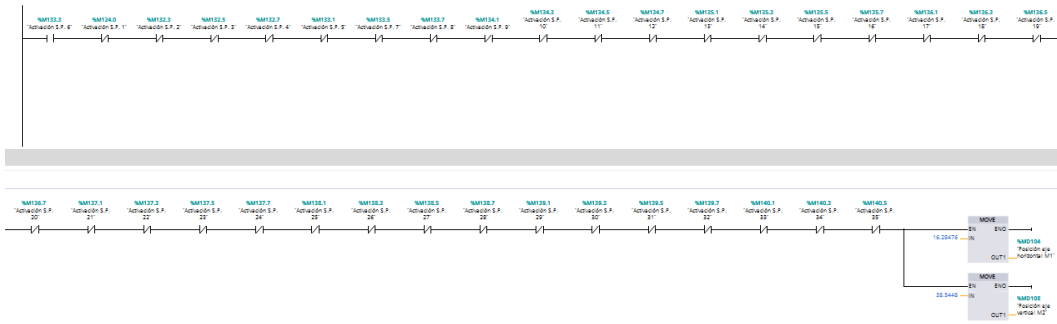
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 5.



FC10.

Segmento 1: Activación posición 6.

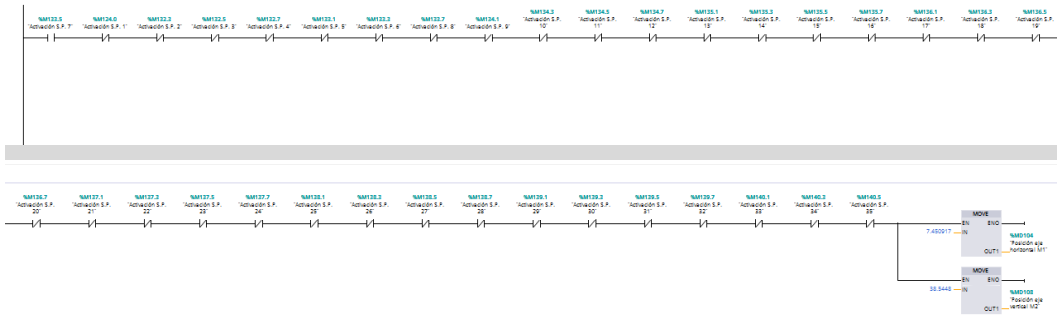
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 6.



FC11.

Segmento 1: Activación posición 7.

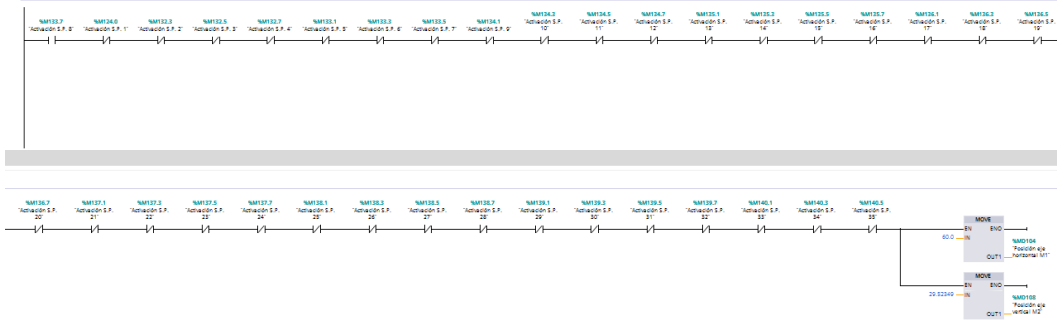
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 7.



FC12.

Segmento 1: Activación posición 8.

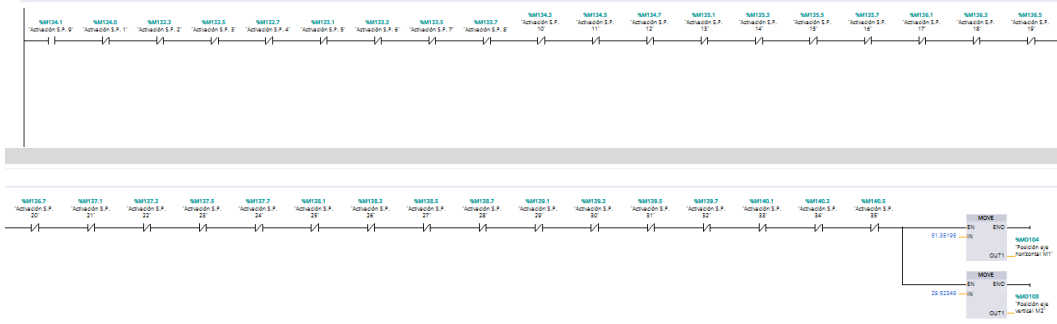
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 8.



FC13.

Segmento 1: Activación posición 9.

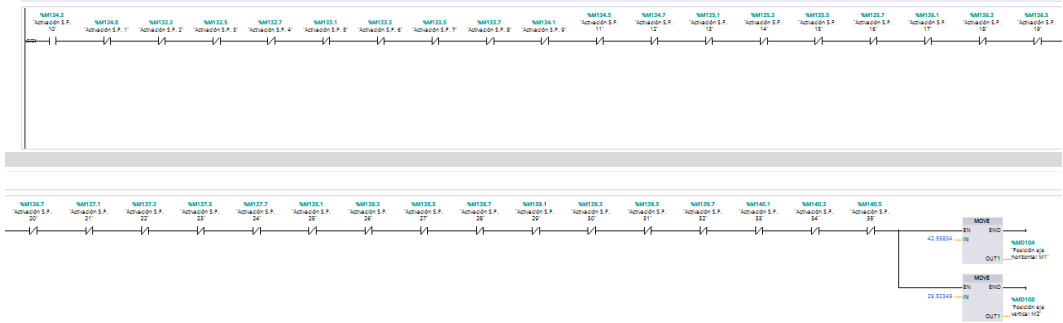
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 9.



FC14.

Segmento 1: Activación posición 10.

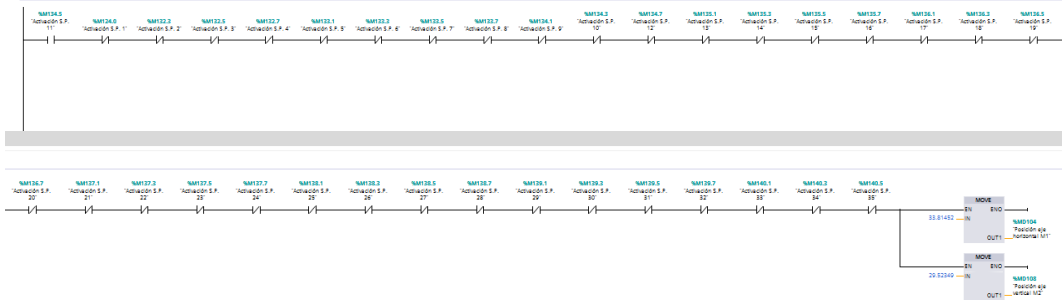
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 10.



FC15.

Segmento 1: Activación posición 11.

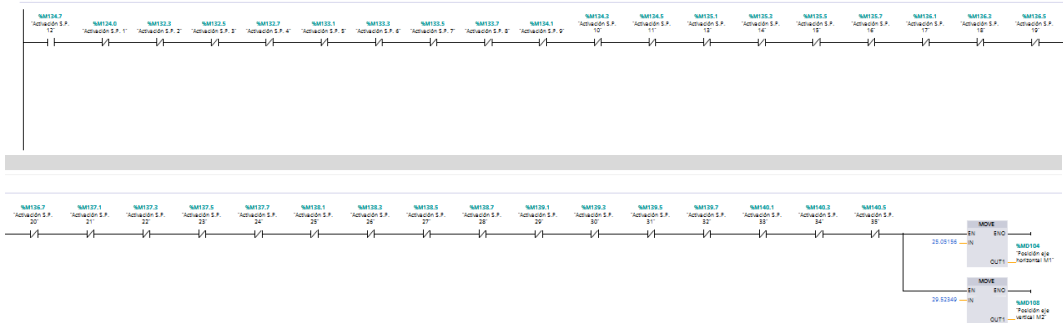
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 11.



FC16.

Segmento 1: Activación posición 12.

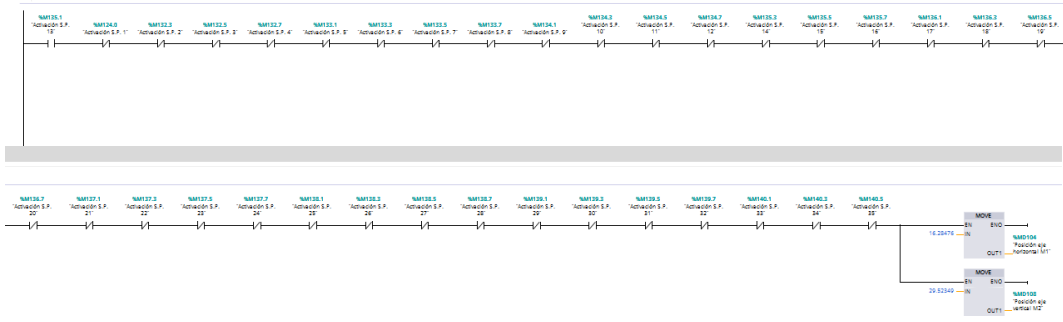
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 12.



FC17.

Segmento 1: Activación posición 13.

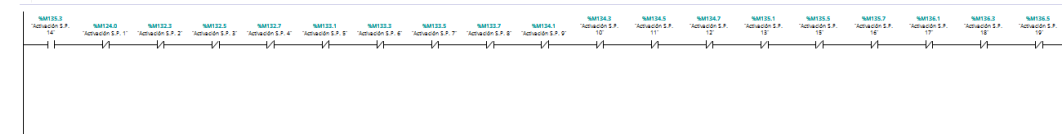
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 13.

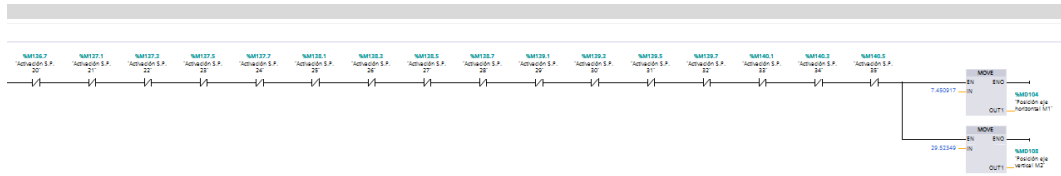


FC18.

Segmento 1: Activación posición 14.

Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 14.

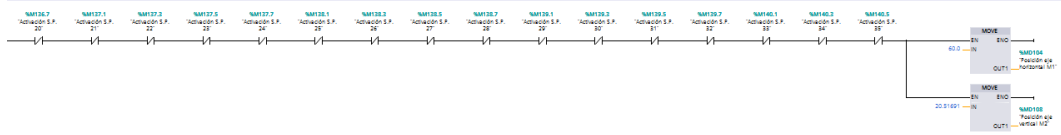
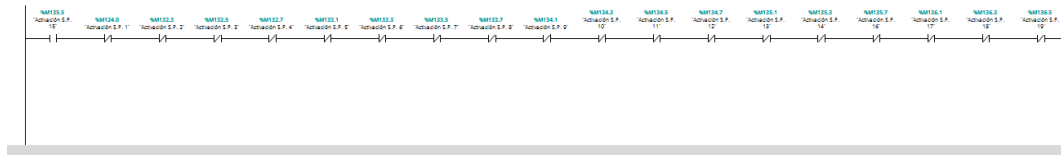




FC19.

Segmento 1: Activación posición 15.

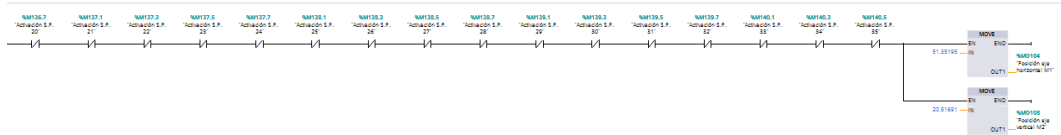
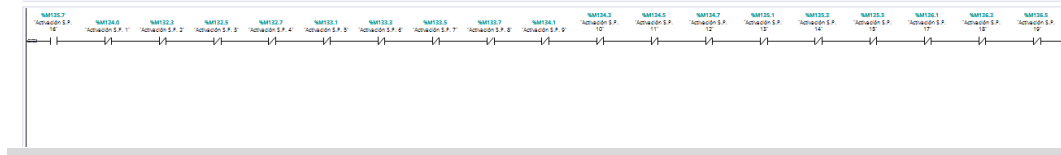
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 15.



FC20.

Segmento 1: Activación posición 16.

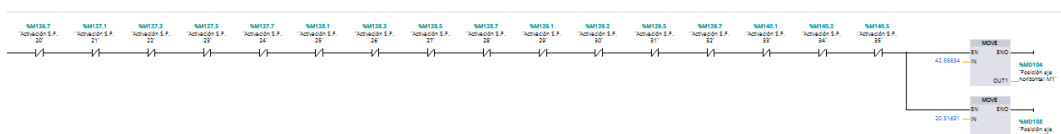
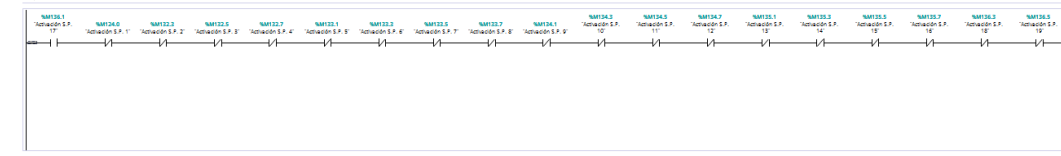
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 16.



FC21.

Segmento 1: Activación posición 17.

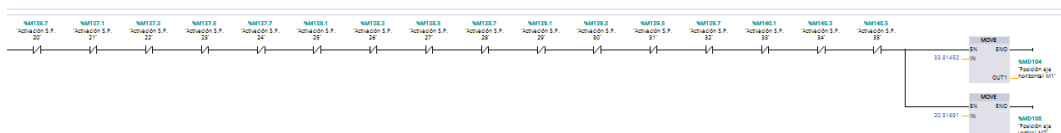
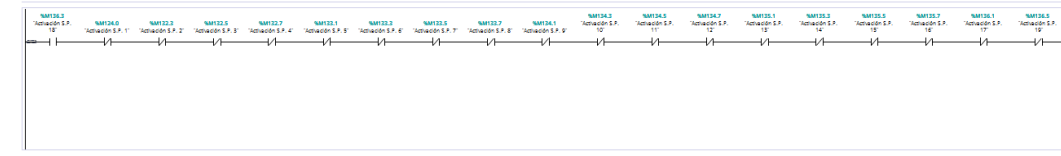
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 17.



FC22.

Segmento 1: Activación posición 18.

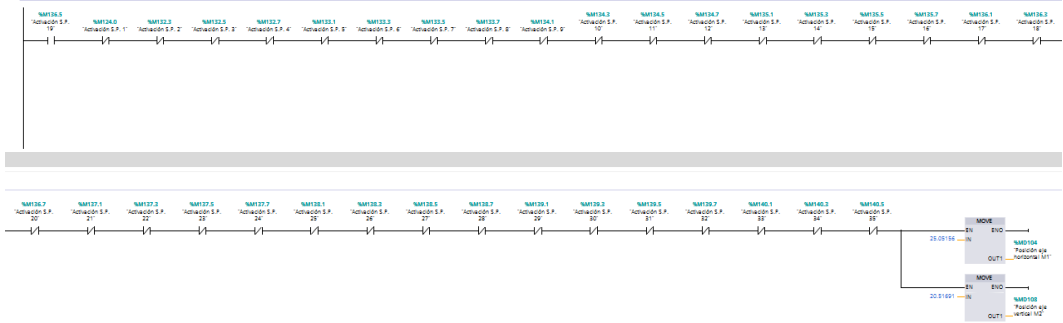
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 18.



FC23.

Segmento 1: Activación posición 19.

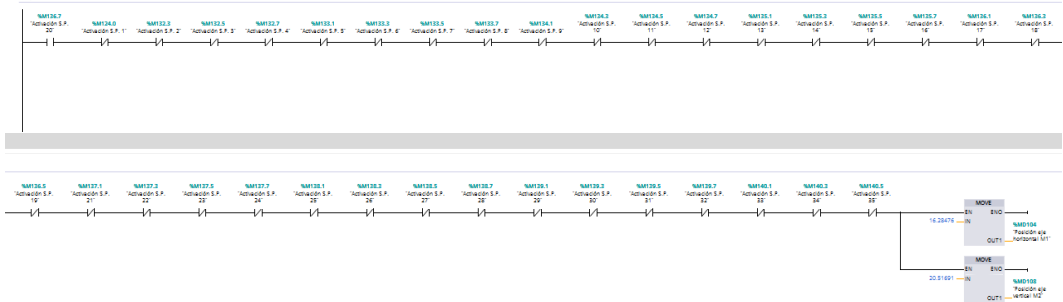
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 19.



FC24.

Segmento 1: Activación posición 20.

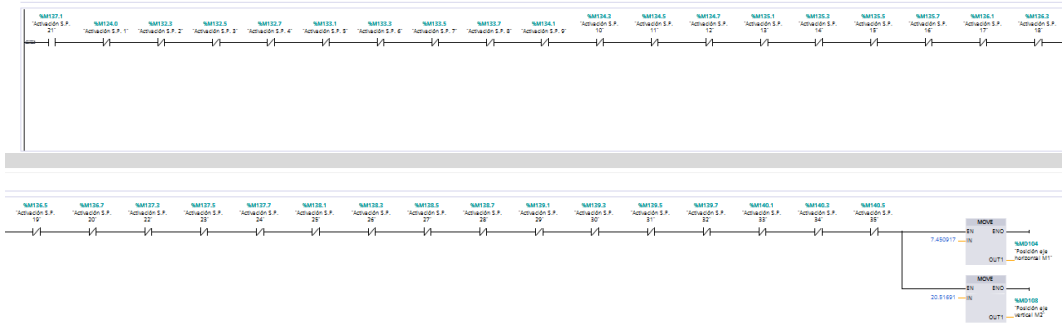
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 20.



FC25.

Segmento 1: Activación posición 21.

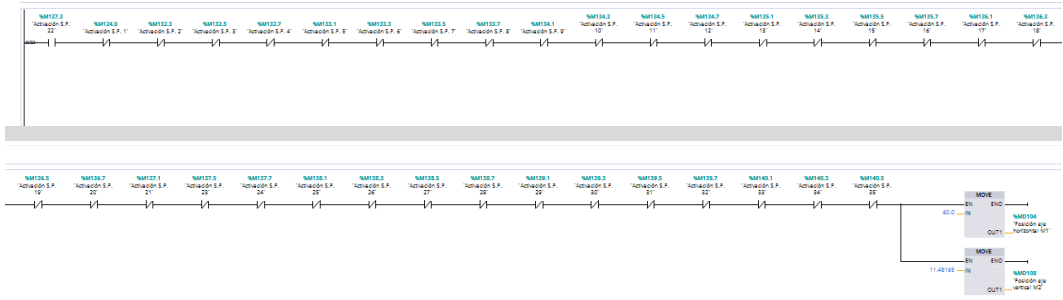
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 21.



FC26.

Segmento 1: Activación posición 22.

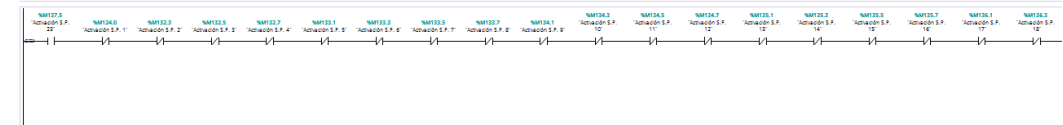
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 22.

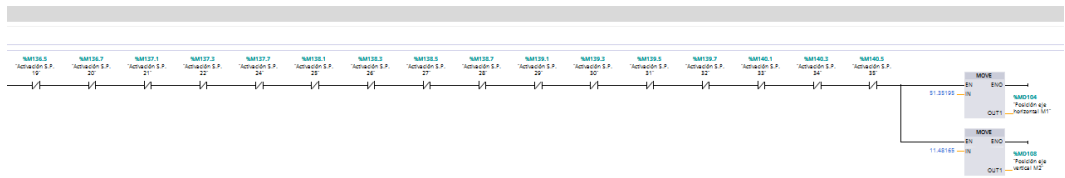


FC27.

Segmento 1: Activación posición 23.

Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 23.

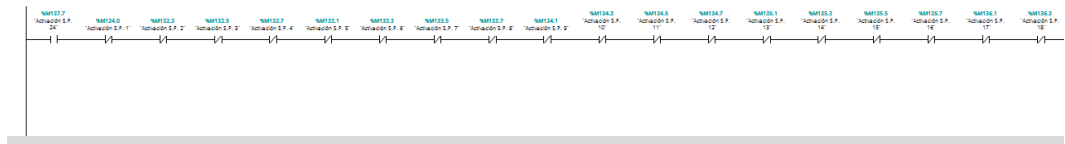




FC28.

Segmento 1: Activación posición 24.

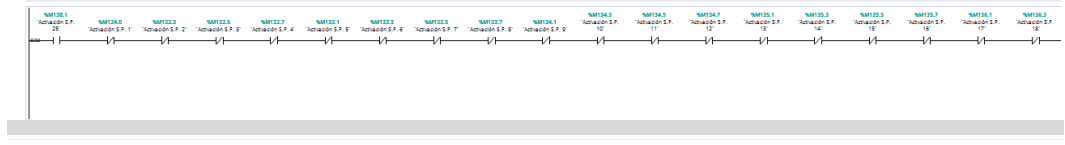
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 24.



FC29.

Segmento 1: Activación posición 25.

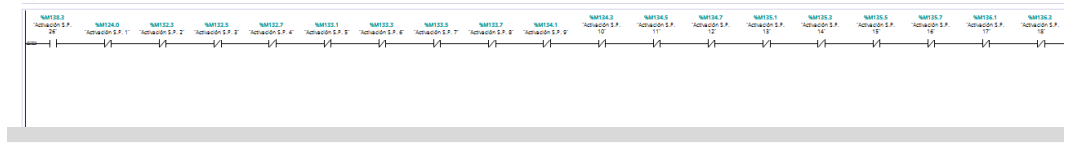
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 25.



FC30.

Segmento 1: Activación posición 26.

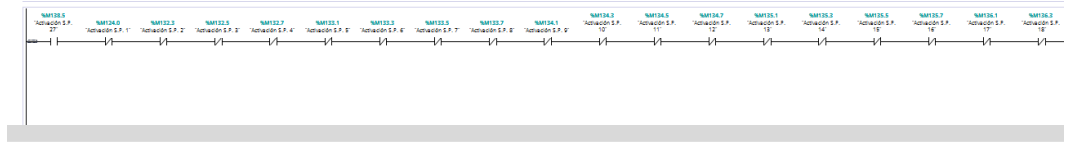
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 26.



FC31.

Segmento 1: Activación posición 27.

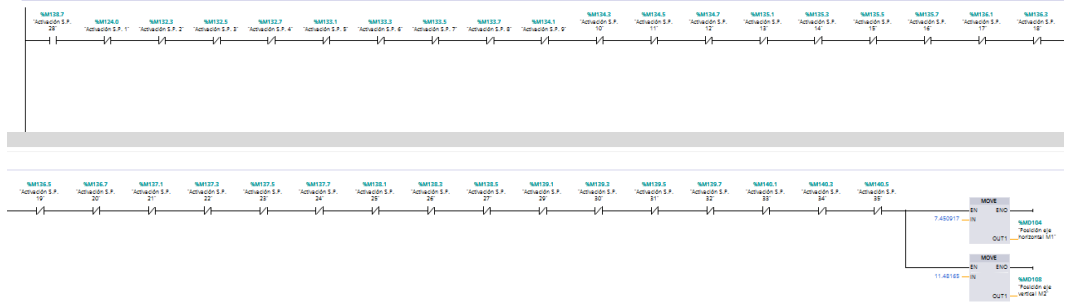
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 27.



FC32.

Segmento 1: Activación posición 28.

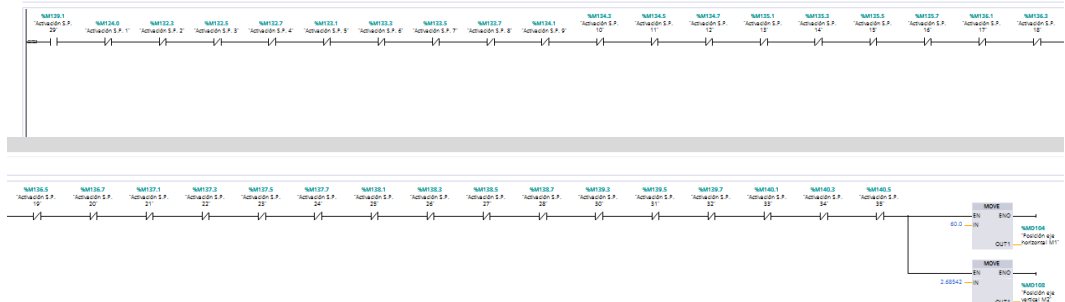
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 28.



FC32.

Segmento 1: Activación posición 29.

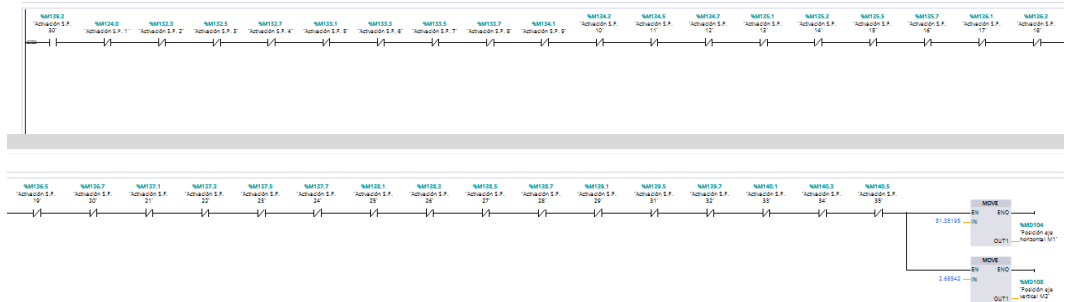
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 29.



FC33.

Segmento 1: Activación posición 30.

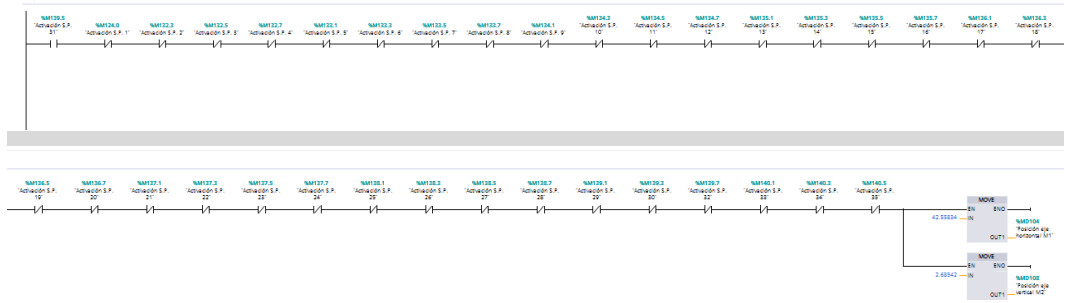
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 30.



FC34.

Segmento 1: Activación posición 31.

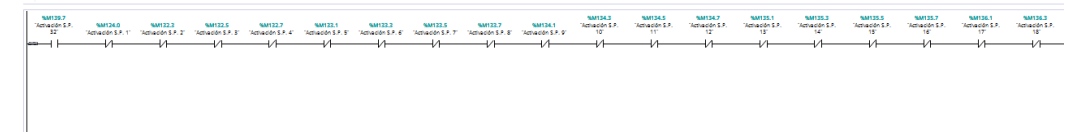
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 31.

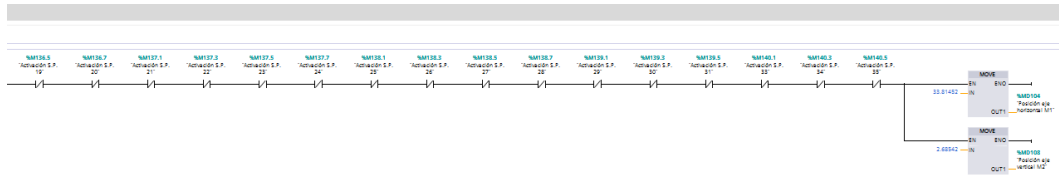


FC35.

Segmento 1: Activación posición 32.

Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 32.

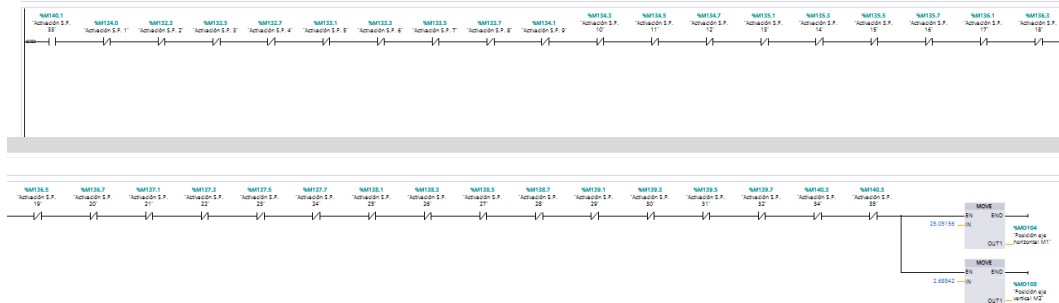




FC36.

Segmento 1: Activación posición 33.

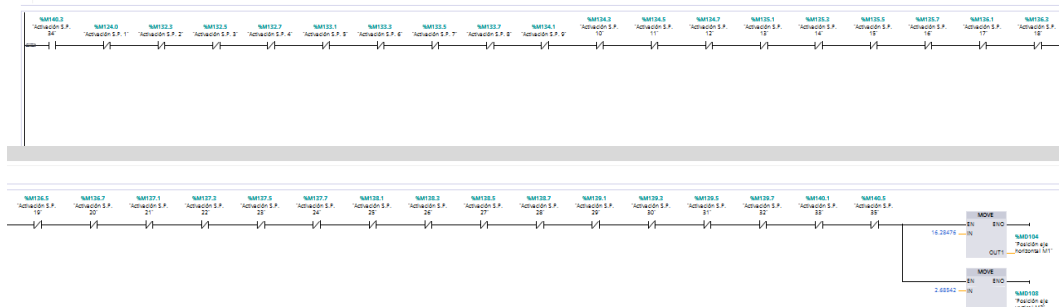
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 33.



FC37.

Segmento 1: Activación posición 34.

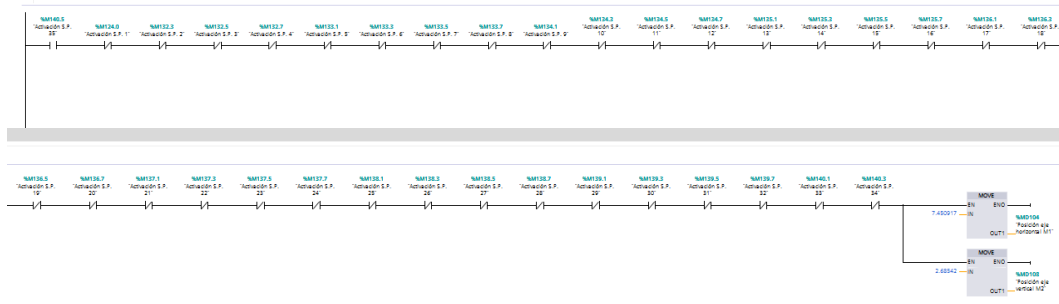
Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 34.



FC38.

Segmento 1: Activación posición 35.

Activación de los Set Points en los ejes horizontal M1 y vertical M2 de la posición 35.



Error Fuzzy M1

//Cálculo del error del control fuzzy del motor M2
 $\#Error_FuzzyM2 := \#SP_FuzzyM2 - \#EncoderM2;$

Error Fuzzy M2

//Cálculo del error del control fuzzy del motor M1
 $\#Error_FuzzyM1 := \#SP_FuzzyM1 - \#EncoderM1;$

Fuzzy M1

//Declaración de variables de salida del motor M1: centros de funciones de membresía
 $\#MNegM1 := -23;$
 $\#NegM1 := -11.75;$
 $\#CeroM1 := 0;$
 $\#PosM1 := 11.75;$

```

#MPosM1 := 23;

IF #Error_FuzzyM1 >= -60.0 AND #Error_FuzzyM1 <= -1.0 THEN
  // Control a partir del setpoint de -60cm hasta -1cm
  #aM1 := 1; //grado de permanencia del rango citado
  #sIM1 := #MNegM1; //salida desfuzificada
  ;
END_IF;

IF #Error_FuzzyM1 > -1.0 AND #Error_FuzzyM1 <= -0.5 THEN
  // Control a partir del setpoint de -1cm hasta -0.5cm
  #aM1 := 1; //grado de permanencia del rango citado
  "L2_FuzzyM1"(xM1 := #Error_FuzzyM1,yM1 => #bM1); //LLamada a la función
L2_FuzzyM1 para calcular el grado de pertenencia
  "L3_FuzzyM1"(xM1 := #Error_FuzzyM1,yM1 => #cM1); //LLamada a la función
L3_FuzzyM1 para calcular el grado de pertenencia
  #alpha1M1 := MIN(IN1 := #aM1, IN2 := #bM1); //Determinación del valor mínimo
de los grados de pertenencia de la función L2_FuzzyM1
  #alpha2M1 := MIN(IN1 := #aM1, IN2 := #cM1); //Determinación del valor mínimo
de los grados de pertenencia de la función L3_FuzzyM1
  "DefuzificaciónFuzzyM1"(alpha1M1 := #alpha1M1,alpha2M1 := #alpha2M1,sIM1
=> #sIM1,u1M1 := #MNegM1,u2M1 := #NegM1); //salida desfuzificada del motor M1
  ;
END_IF;

IF #Error_FuzzyM1 > -0.5 AND #Error_FuzzyM1 <= 0.0 THEN
  // Control a partir del setpoint de -0.5cm hasta 0cm
  #aM1 := 1; //grado de permanencia del rango citado
  "L4_FuzzyM1"(xM1 := #Error_FuzzyM1,yM1 => #bM1); //LLamada a la función
L4_FuzzyM1 para calcular el grado de pertenencia
  "L5_FuzzyM1"(xM1 := #Error_FuzzyM1,yM1 => #cM1); //LLamada a la función
L5_FuzzyM1 para calcular el grado de pertenencia
  #alpha1M1 := MIN(IN1 := #aM1, IN2 := #bM1); //Determinación del valor mínimo
de los grados de pertenencia de la función L4_FuzzyM1
  #alpha2M1 := MIN(IN1 := #aM1, IN2 := #cM1); //Determinación del valor mínimo
de los grados de pertenencia de la función L5_FuzzyM1
  "DefuzificaciónFuzzyM1"(alpha1M1 := #alpha1M1,alpha2M1 := #alpha2M1,sIM1
=> #sIM1,u1M1 := #NegM1,u2M1 := #CeroM1); //salida desfuzificada del motor M1
  ;
END_IF;

IF #Error_FuzzyM1 > 0.0 AND #Error_FuzzyM1 <= 0.5 THEN
  #aM1 := 1; //grado de permanencia del rango citado
  // Control a partir del setpoint de 0cm hasta 0.5cm
  "L6_FuzzyM1"(xM1 := #Error_FuzzyM1,yM1 => #bM1); //LLamada a la función
L6_FuzzyM1 para calcular el grado de pertenencia
  "L7_FuzzyM1"(xM1 := #Error_FuzzyM1,yM1 => #cM1); //LLamada a la función
L7_FuzzyM1 para calcular el grado de pertenencia
  #alpha1M1 := MIN(IN1 := #aM1, IN2 := #bM1); //Determinación del valor mínimo
de los grados de pertenencia de la función L6_FuzzyM1
  #alpha2M1 := MIN(IN1 := #aM1, IN2 := #cM1); //Determinación del valor mínimo
de los grados de pertenencia de la función L7_FuzzyM1
  "DefuzificaciónFuzzyM1"(alpha1M1 := #alpha1M1,alpha2M1 := #alpha2M1,sIM1
=> #sIM1,u1M1 := #CeroM1,u2M1 := #PosM1); //salida desfuzificada del motor M1
  ;

```

END_IF;

```
IF #Error_FuzzyM1 > 0.5 AND #Error_FuzzyM1 <= 1.0 THEN
  // Control a partir del setpoint de 0.5cm hasta 1cm
  #aM1 := 1; //grado de permanencia del rango citado
  "L8_FuzzyM1"(xM1 := #Error_FuzzyM1,yM1 => #bM1); //LLamada a la función
  L8_FuzzyM1 para calcular el grado de pertenencia
  "L9_FuzzyM1"(xM1 := #Error_FuzzyM1,yM1 => #cM1); //LLamada a la función
  L9_FuzzyM1 para calcular el grado de pertenencia
  #alpha1M1 := MIN(IN1 := #aM1, IN2 := #bM1); //Determinación del valor mínimo
  de los grados de pertenencia de la función L8_FuzzyM1
  #alpha2M1 := MIN(IN1 := #aM1, IN2 := #cM1); //Determinación del valor mínimo
  de los grados de pertenencia de la función L9_FuzzyM1
  "DefuzificaciónFuzzyM1"(alpha1M1 := #alpha1M1,alpha2M1 := #alpha2M1,sIM1
  => #sIM1,u1M1 := #PosM1,u2M1 := #MPosM1); //salida desfuzificada del motor M1
  ;
END_IF;
```

```
IF #Error_FuzzyM1 > 1.0 AND #Error_FuzzyM1 <= 60.1 THEN
  // Control a partir del setpoint de 1cm hasta 60.1cm
  #aM1 := 1; //grado de permanencia del rango citado
  #sIM1 := #MPosM1; //salida desfuzificada
  ;
END_IF;
```

Fuzzy M2

//Declaración de variables de salida del motor M2: centros de funciones de membresía

```
#MNegM2 := -23;
#NegM2 := -11.75;
#CeroM2 := 0;
#PosM2 := 11.75;
#MPosM2 := 23;
```

```
IF #Error_FuzzyM2 >= -41.0 AND #Error_FuzzyM2 <= -1.5 THEN
  // Control a partir del setpoint del motor M2 de -41cm hasta -1.5cm
  #aM2 := 1; //grado de permanencia del rango citado
  #sIM2 := #MNegM2; //salida desfuzificada del motor M2
  ;
END_IF;
```

```
IF #Error_FuzzyM2 > -1.5 AND #Error_FuzzyM2 <= -0.75 THEN
  // Control a partir del setpoint del motor M2 de -1.5cm hasta -0.75cm
  #aM2 := 1; //grado de permanencia del rango citado
  "L2_FuzzyM2"(xM2 := #Error_FuzzyM2,yM2 => #bM2); //LLamada a la función
  L2_FuzzyM2 para calcular el grado de pertenencia
  "L3_FuzzyM2"(xM2 := #Error_FuzzyM2,yM2 => #cM2); //LLamada a la función
  L3_FuzzyM2 para calcular el grado de pertenencia
  #alpha1M2 := MIN(IN1 := #aM2, IN2 := #bM2); //Determinación del valor mínimo
  de los grados de pertenencia de la función L2_FuzzyM2
  #alpha2M2 := MIN(IN1 := #aM2, IN2 := #cM2); //Determinación del valor mínimo
  de los grados de pertenencia de la función L3_FuzzyM2
  "DefuzificaciónFuzzyM2"(alpha1M2 := #alpha1M2,alpha2M2 := #alpha2M2,sIM2
  => #sIM2,u1M2 := #MNegM2,u2M2 := #NegM2); //salida desfuzificada del motor M2
  ;
```

```

END_IF;

IF #Error_FuzzyM2 > -0.75 AND #Error_FuzzyM2 <= 0.0 THEN
  // Control a partir del setpoint del motor M2 de -0.75cm hasta 0cm
  #aM2 := 1; //grado de permanencia del rango citado
  "L4_FuzzyM2"(xM2 := #Error_FuzzyM2,yM2 => #bM2); //LLamada a la función
  L4_FuzzyM2 para calcular el grado de pertenencia
  "L5_FuzzyM2"(xM2 := #Error_FuzzyM2,yM2 => #cM2); //LLamada a la función
  L5_FuzzyM2 para calcular el grado de pertenencia
  #alpha1M2 := MIN(IN1 := #aM2, IN2 := #bM2); //Determinación del valor mínimo
  de los grados de pertenencia de la función L4_FuzzyM2
  #alpha2M2 := MIN(IN1 := #aM2, IN2 := #cM2); //Determinación del valor mínimo
  de los grados de pertenencia de la función L5_FuzzyM2
  "DefuzificaciónFuzzyM2"(alpha1M2 := #alpha1M2,alpha2M2 := #alpha2M2,sIM2
  => #sIM2,u1M2 := #NegM2,u2M2 := #CeroM2); //salida desfuzificada del motor M2
  ;
END_IF;

IF #Error_FuzzyM2 > 0.0 AND #Error_FuzzyM2 <= 0.75 THEN
  // Control a partir del setpoint del motor M2 de 0cm hasta 0.75cm
  #aM2 := 1; //grado de permanencia del rango citado
  "L6_FuzzyM2"(xM2 := #Error_FuzzyM2,yM2 => #bM2); //LLamada a la función
  L6_FuzzyM2 para calcular el grado de pertenencia
  "L7_FuzzyM2"(xM2 := #Error_FuzzyM2,yM2 => #cM2); //LLamada a la función
  L7_FuzzyM1 para calcular el grado de pertenencia
  #alpha1M2 := MIN(IN1 := #aM2, IN2 := #bM2); //Determinación del valor mínimo
  de los grados de pertenencia de la función L6_FuzzyM2
  #alpha2M2 := MIN(IN1 := #aM2, IN2 := #cM2); //Determinación del valor mínimo
  de los grados de pertenencia de la función L7_FuzzyM2
  "DefuzificaciónFuzzyM2"(alpha1M2 := #alpha1M2,alpha2M2 := #alpha2M2,sIM2
  => #sIM2,u1M2 := #CeroM2,u2M2 := #PosM2); //salida desfuzificada del motor M2
  ;
END_IF;

IF #Error_FuzzyM2 > 0.75 AND #Error_FuzzyM2 <= 1.5 THEN
  // Control a partir del setpoint del motor M2 de 0.75cm hasta 1.5cm
  #aM2 := 1; //grado de permanencia del rango citado
  "L8_FuzzyM2"(xM2 := #Error_FuzzyM2,yM2 => #bM2); //LLamada a la función
  L8_FuzzyM2 para calcular el grado de pertenencia
  "L9_FuzzyM2"(xM2 := #Error_FuzzyM2,yM2 => #cM2); //LLamada a la función
  L9_FuzzyM2 para calcular el grado de pertenencia
  #alpha1M2 := MIN(IN1 := #aM2, IN2 := #bM2); //Determinación del valor mínimo
  de los grados de pertenencia de la función L8_FuzzyM2
  #alpha2M2 := MIN(IN1 := #aM2, IN2 := #cM2); //Determinación del valor mínimo
  de los grados de pertenencia de la función L9_FuzzyM2
  "DefuzificaciónFuzzyM2"(alpha1M2 := #alpha1M2,alpha2M2 := #alpha2M2,sIM2
  => #sIM2,u1M2 := #PosM2,u2M2 := #MPosM2); //salida desfuzificada del motor M2
  ;
END_IF;

IF #Error_FuzzyM2 > 1.5 AND #Error_FuzzyM2 <= 41.1 THEN
  // Control a partir del setpoint del motor M2 de 1.5cm hasta 41.1cm
  #aM2 := 1; //grado de permanencia del rango citado
  #sIM2 := #MPosM2; //salida desfuzificada del motor M2
  ;

```


END_IF;

L1_FuzzyM1

//Cálculo del grado de pertenencia de la función L1 del control fuzzy del motor M1
#yM1 := 1;

L1_FuzzyM2

//Cálculo del grado de pertenencia de la función L1 del control fuzzy del motor M2
#yM2 := 1;

L2_FuzzyM1

//Cálculo del grado de pertenencia de la función L2 del control fuzzy del motor M1
#yM1 := (-2 * #xM1) - 1;

L2_FuzzyM2

//Cálculo del grado de pertenencia de la función L2 del control fuzzy del motor M2
#yM2 := (- #xM2 - 1.5 + 0.75) / 0.75;

L3_FuzzyM1

//Cálculo del grado de pertenencia de la función L3 del control fuzzy del motor M1
#yM1 := (2 * #xM1) + 2;

L3_FuzzyM2

//Cálculo del grado de pertenencia de la función L3 del control fuzzy del motor M2
#yM2 := (#xM2 + 1.5) / 0.75;

L4_FuzzyM1

//Cálculo del grado de pertenencia de la función L4 del control fuzzy del motor M1
#yM1 := -2 * #xM1;

L4_FuzzyM2

//Cálculo del grado de pertenencia de la función L4 del control fuzzy del motor M2
#yM2 := - #xM2 / 0.75;

L5_FuzzyM1

//Cálculo del grado de pertenencia de la función L5 del control fuzzy del motor M1
#yM1 := (2 * #xM1) + 1;

L5_FuzzyM2

//Cálculo del grado de pertenencia de la función L5 del control fuzzy del motor M2
#yM2 := (#xM2 + 0.75) / 0.75;

L6_FuzzyM1

//Cálculo del grado de pertenencia de la función L6 del control fuzzy del motor M1
#yM1 := (-2 * #xM1) + 1;

L6_FuzzyM2

//Cálculo del grado de pertenencia de la función L6 del control fuzzy del motor M2
#yM2 := (- #xM2 + 0.75) / 0.75;

L7_FuzzyM1

//Cálculo del grado de pertenencia de la función L7 del control fuzzy del motor M1
#yM1 := 2 * #xM1;

L7_FuzzyM2

//Cálculo del grado de pertenencia de la función L7 del control fuzzy del motor M2
#yM2 := #xM2 / 0.75;

L8_FuzzyM1

//Cálculo del grado de pertenencia de la función L8 del control fuzzy del motor M1
#yM1 := (-2 * #xM1) + 2;

L8_FuzzyM2

//Cálculo del grado de pertenencia de la función L8 del control fuzzy del motor M2
#yM2 := (- #xM2 + 1.5) / 0.75;

L9_FuzzyM1

//Cálculo del grado de pertenencia de la función L9 del control fuzzy del motor M1
#yM1 := (2 * #xM1) - 1;

L9_FuzzyM2

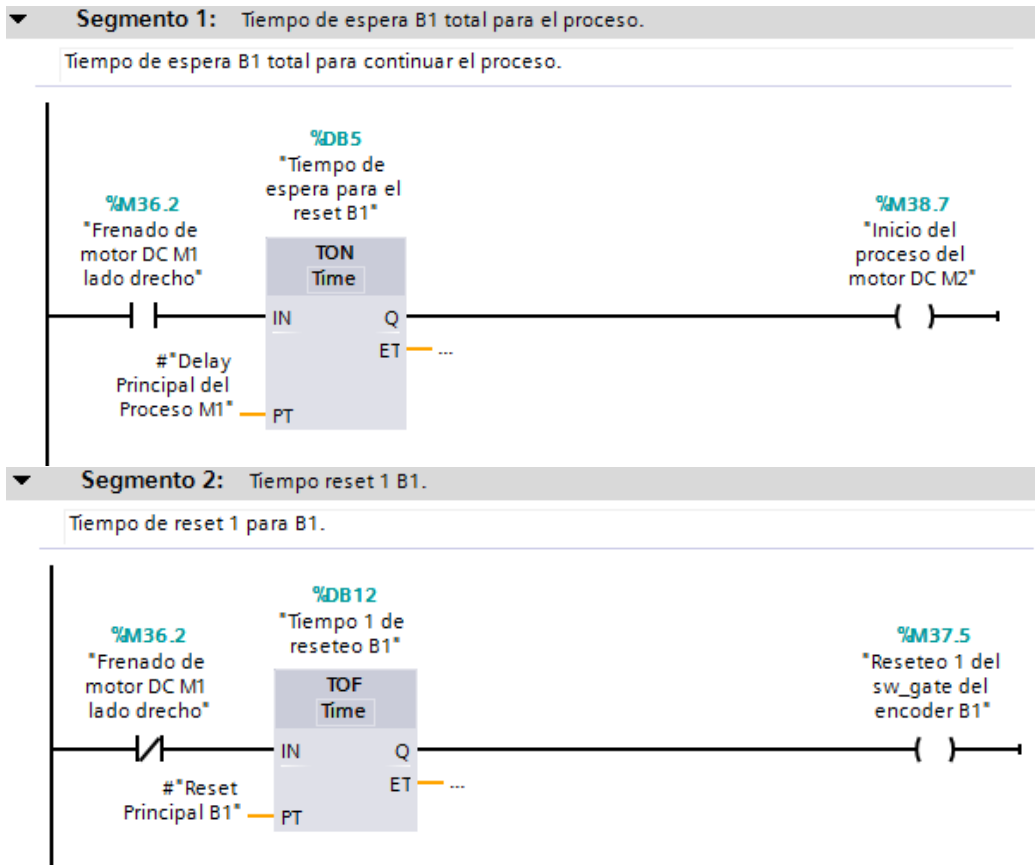
//Cálculo del grado de pertenencia de la función L9 del control fuzzy del motor M2
#yM2 := (#xM2 - 0.75) / 0.75;

L10_FuzzyM1

//Cálculo del grado de pertenencia de la función L10 del control fuzzy del motor M1
#yM1 := 1;

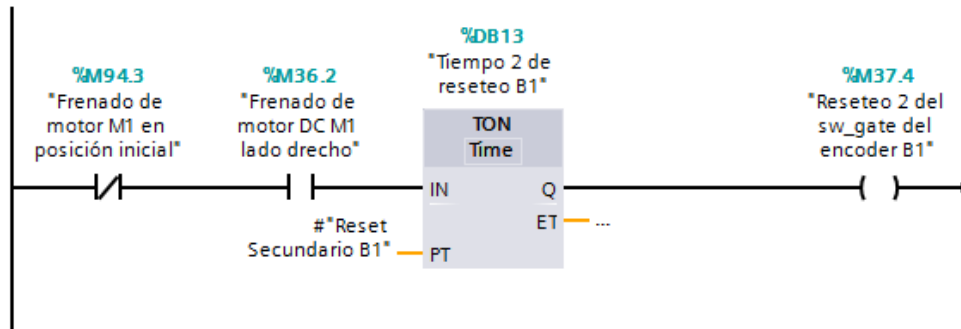
L10_FuzzyM2

//Cálculo del grado de pertenencia de la función L10 del control fuzzy del motor M2
#yM2 := 1;



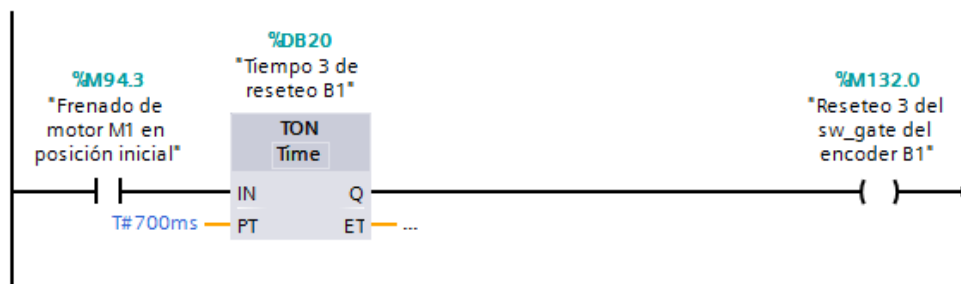
▼ **Segmento 3:** Tiempo reset 2 B1.

Tiempo de reset 2 para B1.



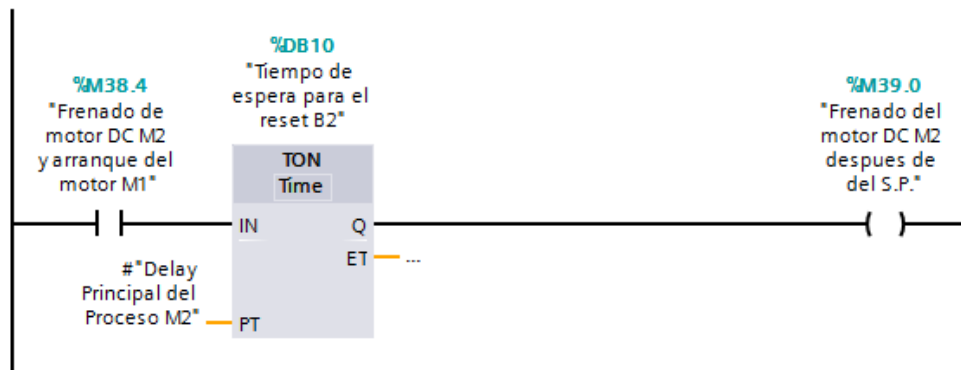
▼ **Segmento 4:** Tiempo reset 3 B1.

Tiempo de reset 3 para B1.



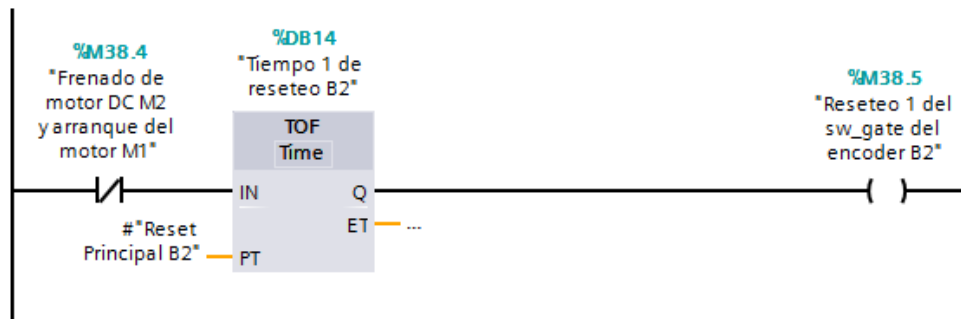
▼ **Segmento 1:** Tiempo de espera B2 total para el proceso.

Tiempo de espera B2 total para continuar el proceso.



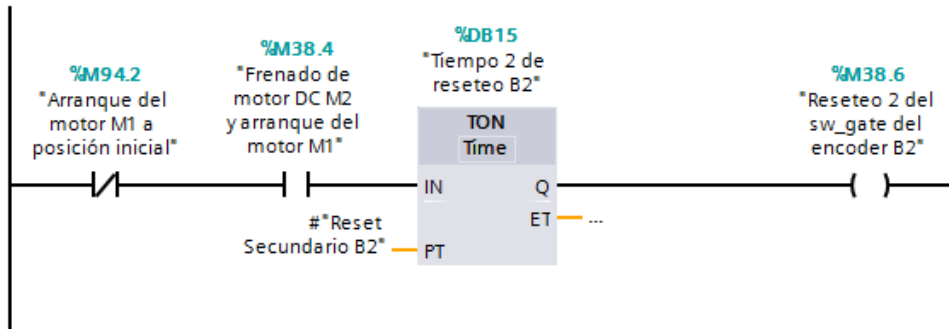
▼ **Segmento 2:** Tiempo reset 1 B2.

Tiempo de reset 1 para B2.



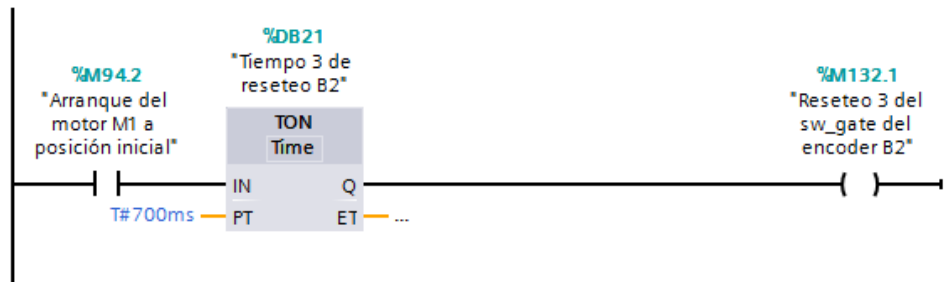
▼ **Segmento 3:** Tiempo reset 2 B2.

Tiempo de reset 2 para B2.



▼ **Segmento 4:** Tiempo reset 3 B2.

Tiempo de reset 3 para B2.



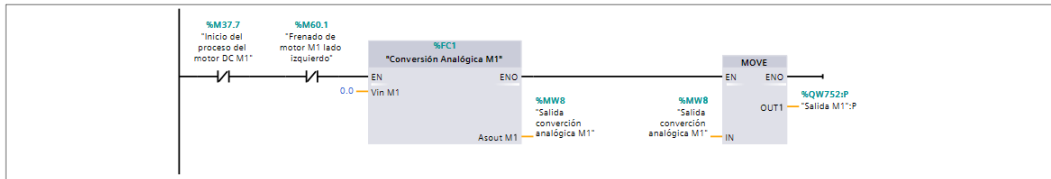
ANEXO 7

PROGRAMACIÓN PRINCIPAL

BLOQUE MAIN – OB1

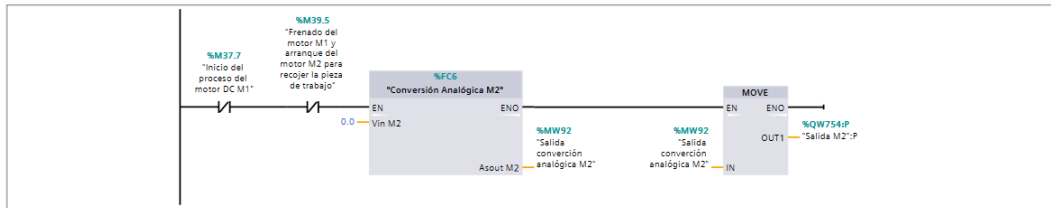
Segmento 1: Condición inicial M1.

Condicion inicial del proceso motor DC M1 esté apagado.



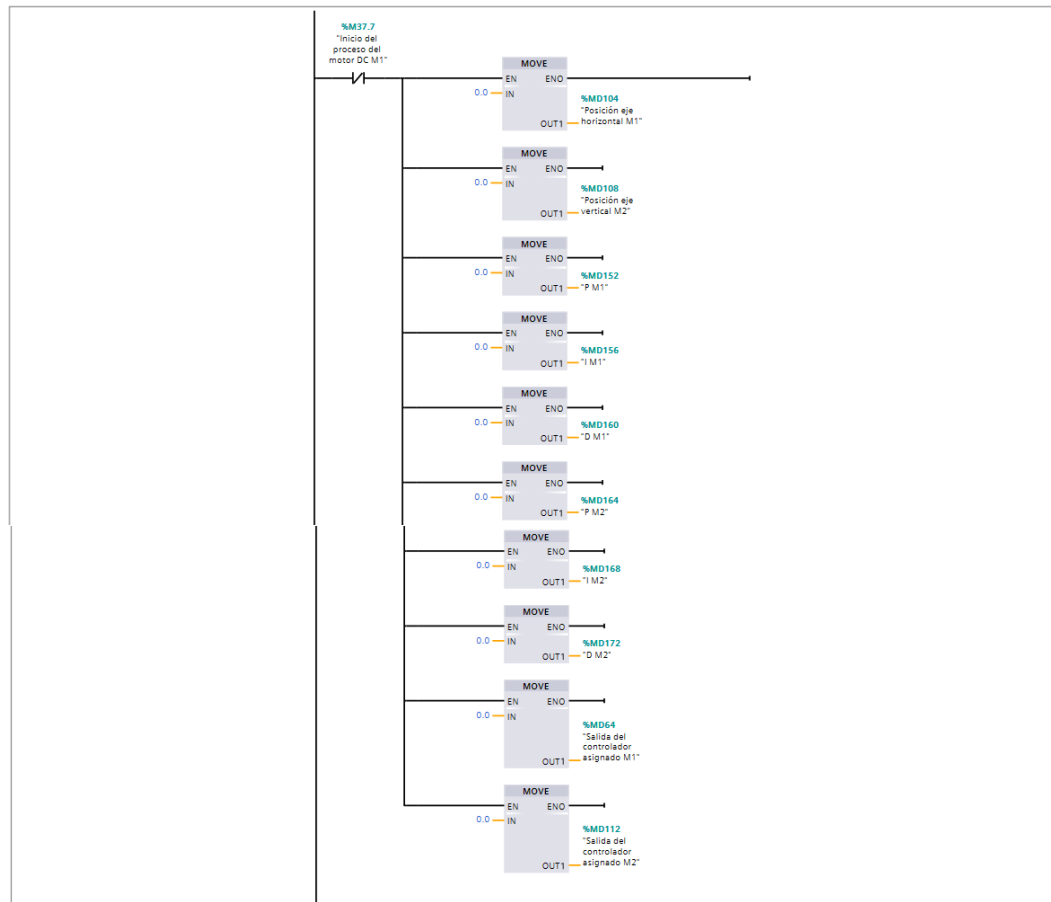
Segmento 2: Condición inicial M2.

Condicion inicial del proceso motor DC M2 esté apagado.



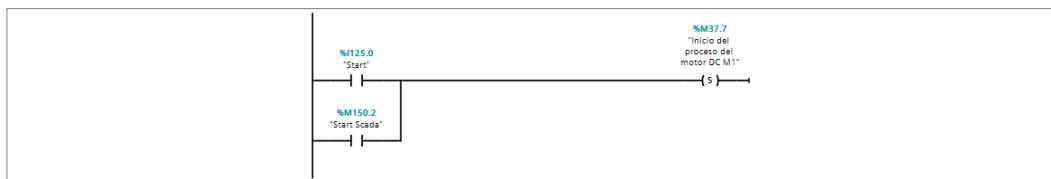
Segmento 3: Condición inicial Set Points.

Condicion inicial del proceso Set Points en cero.



Segmento 4: Start.

Inicio del proceso en la planta a partir del pulsador Start.

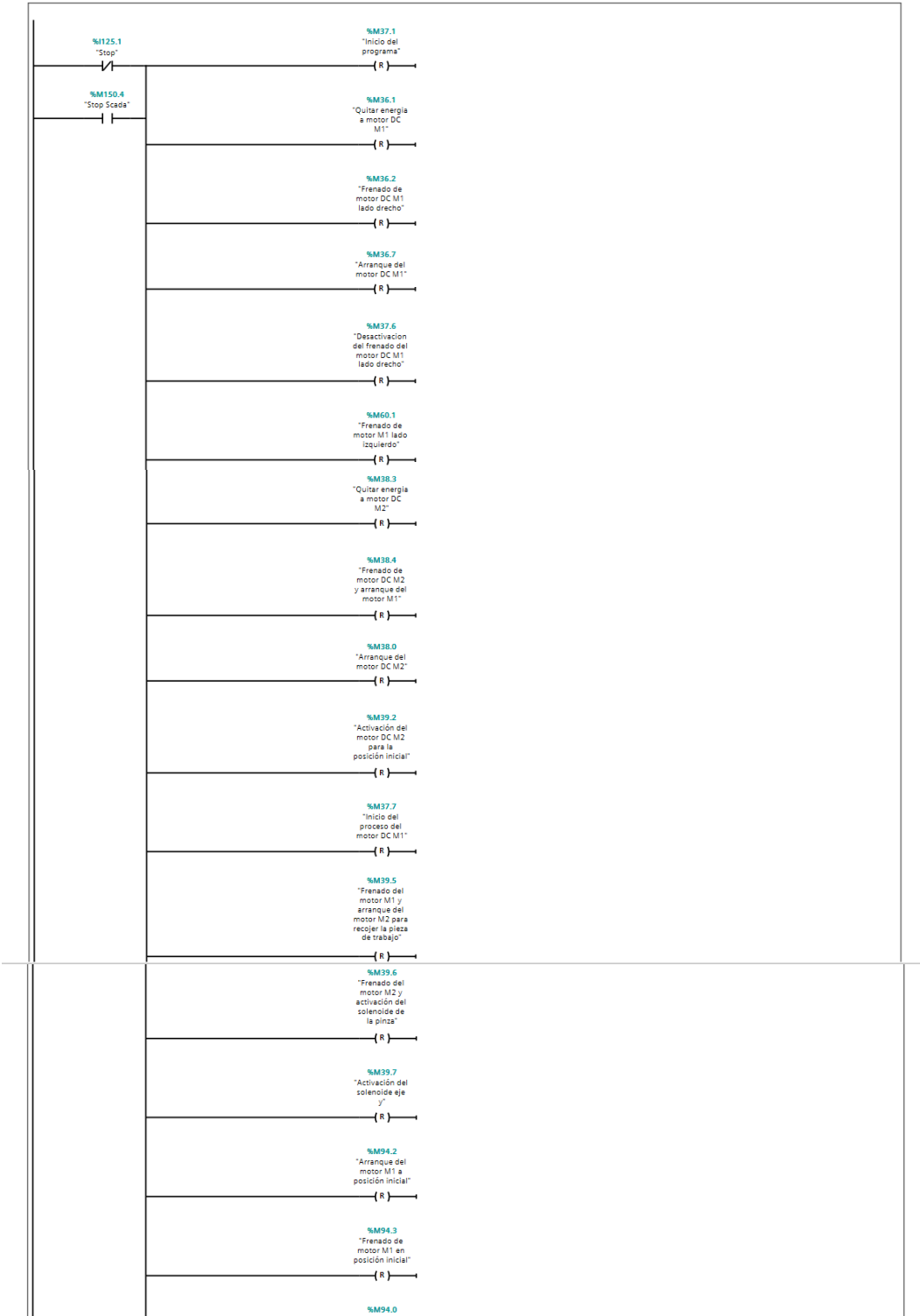


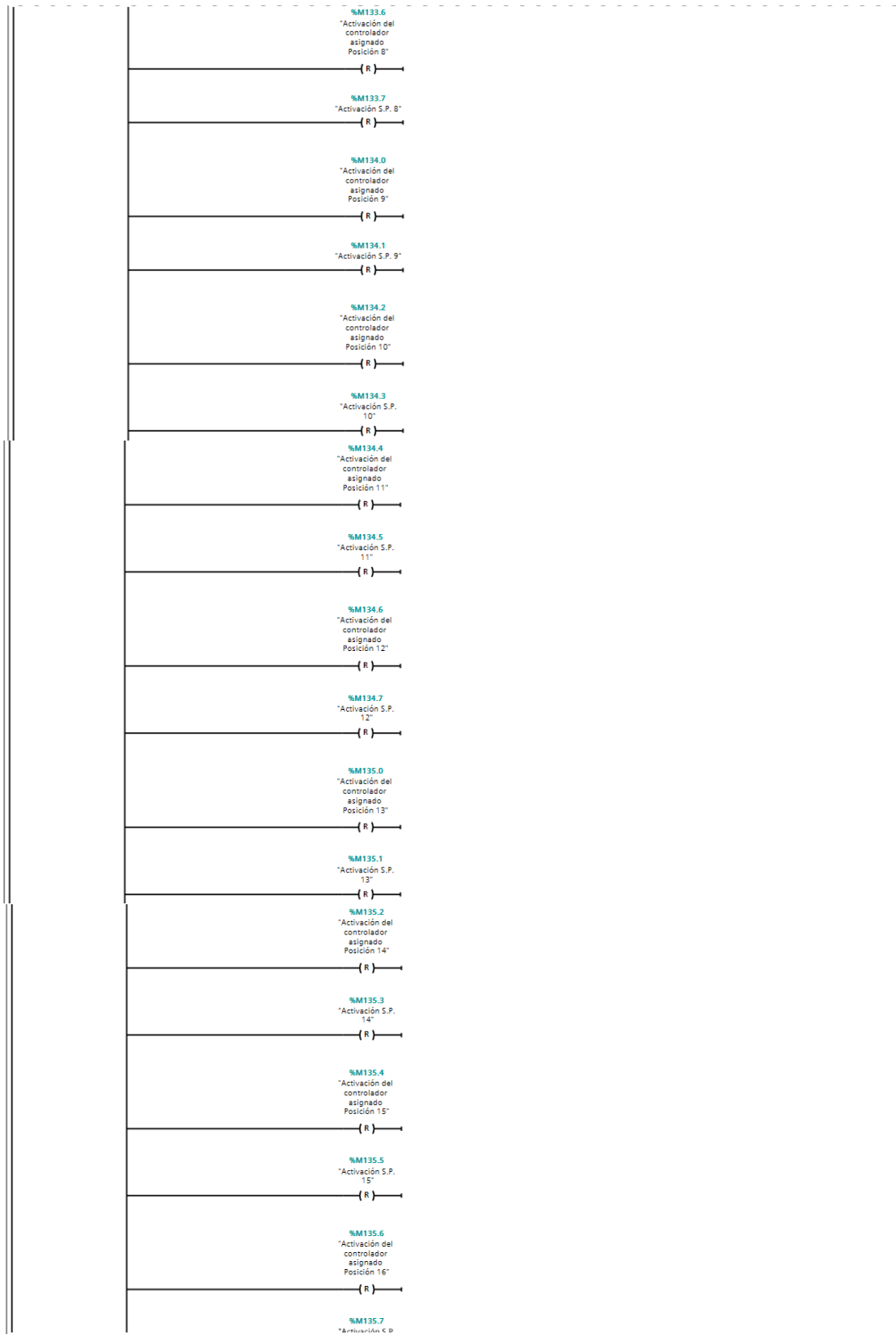
Segmento 5: Proceso Iniciado.

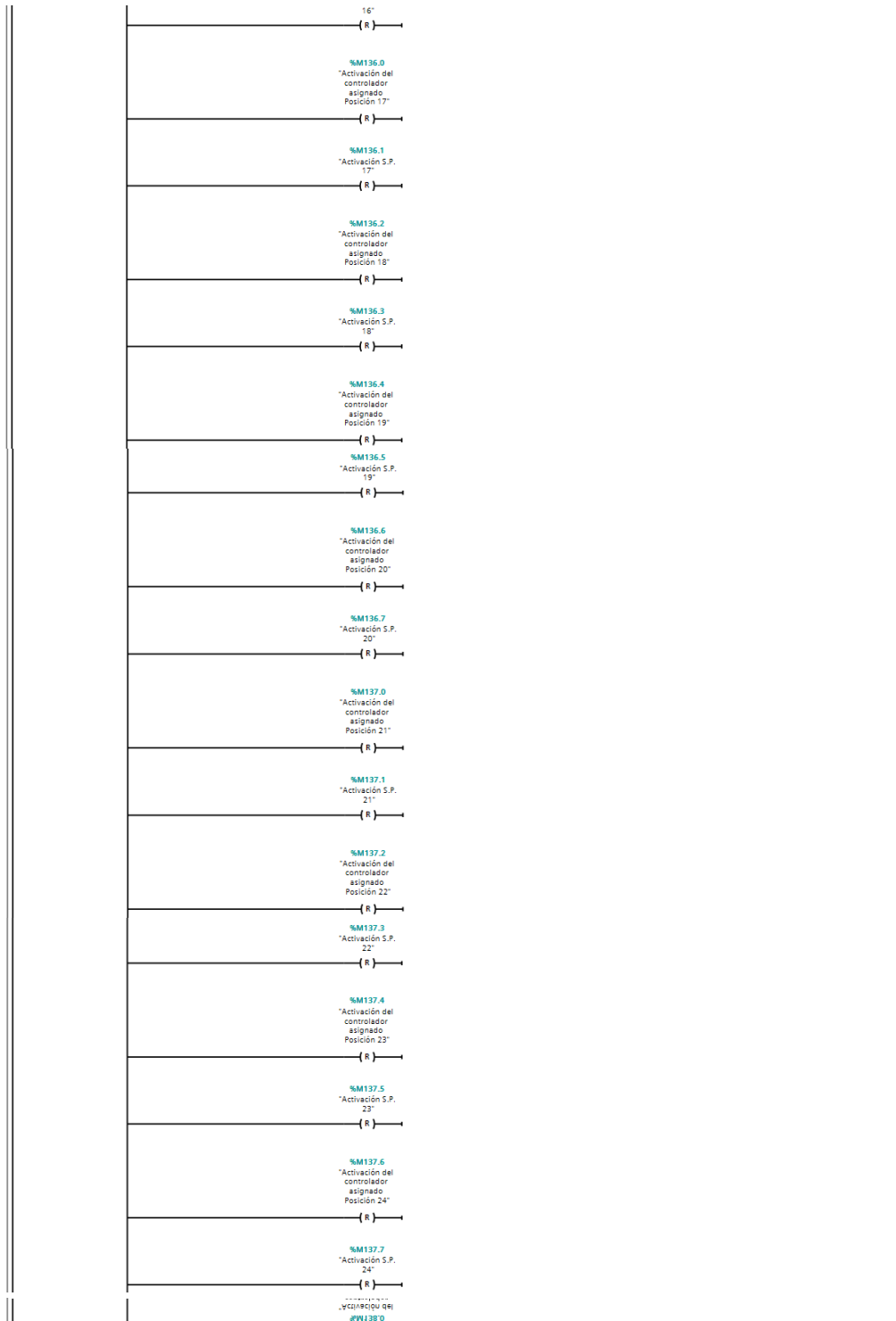


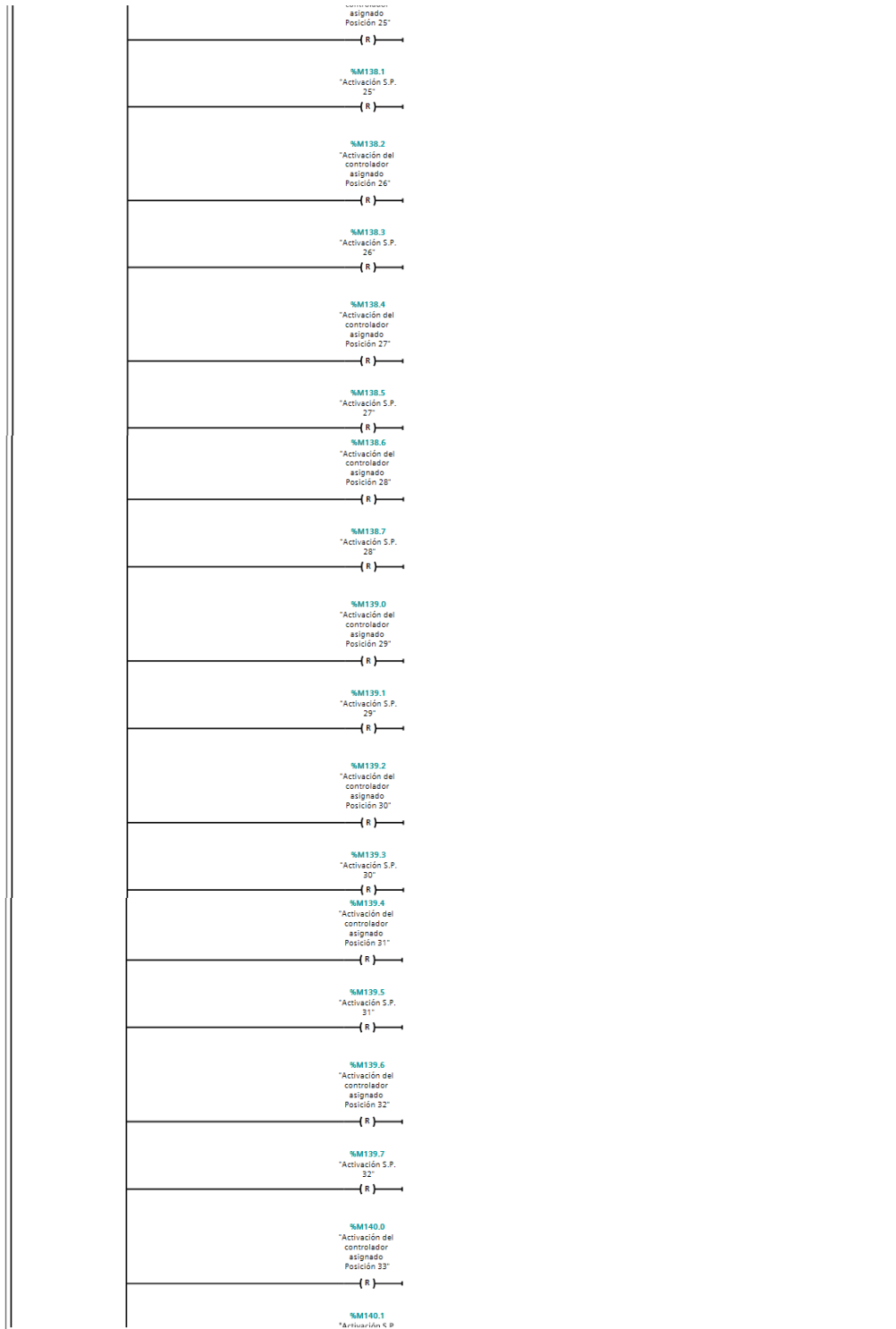
Segmento 6: Stop.

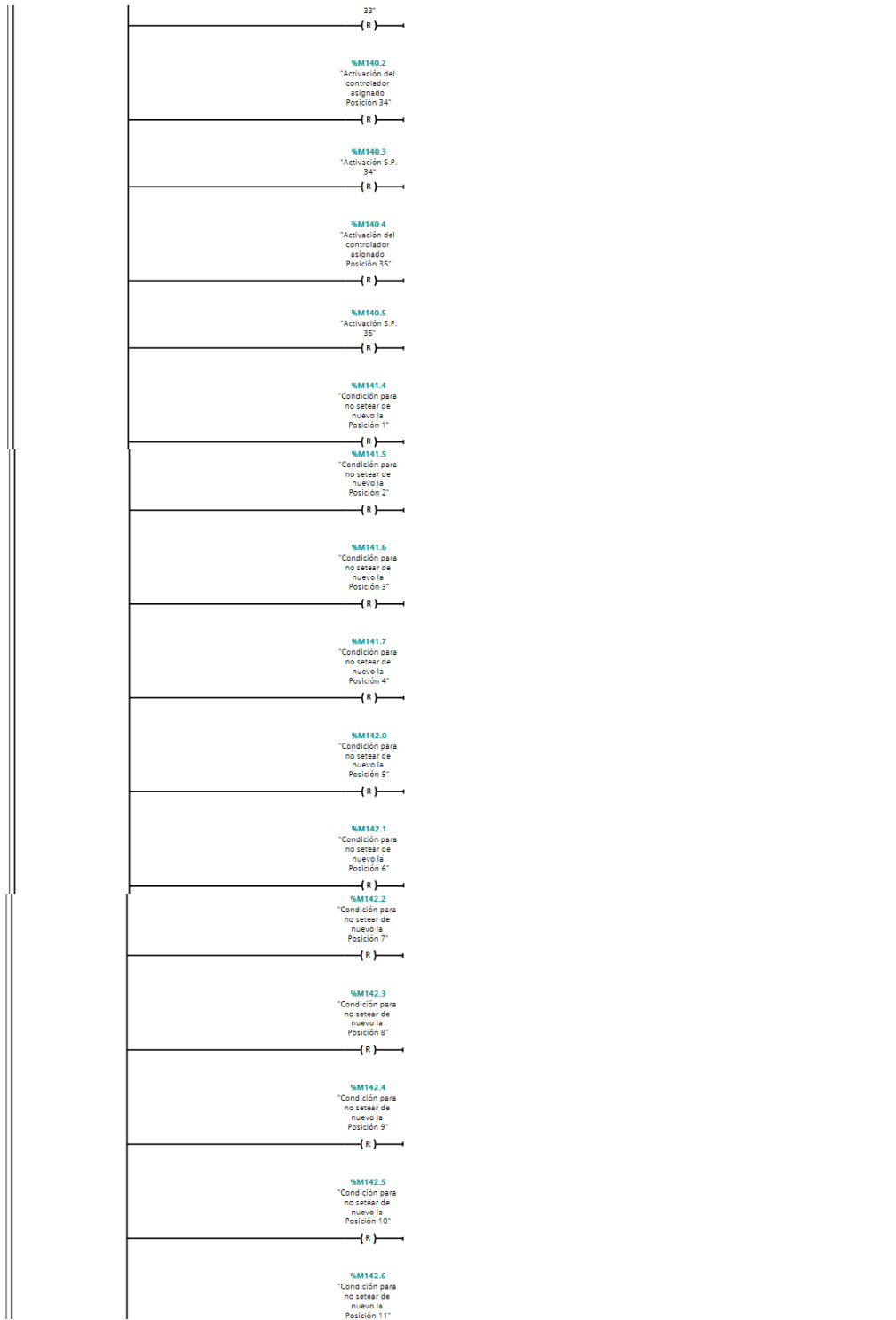
Paro del proceso en la planta a partir del pulsador Stop.







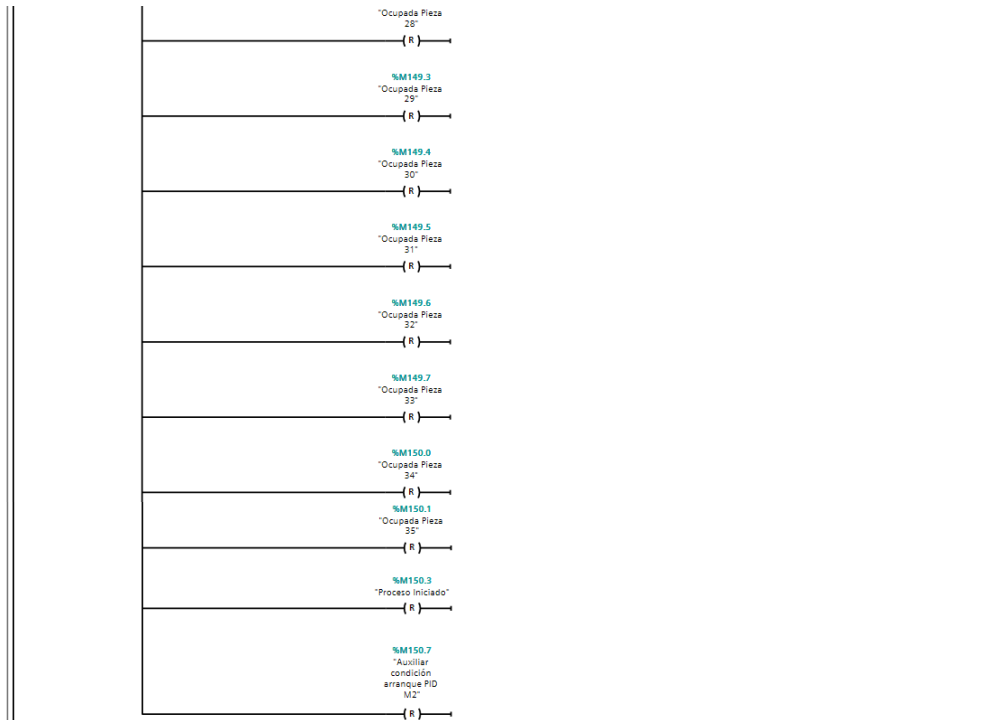




(R)
%M142.7 *Condición para no setear de nuevo la Posición 12*
(R)
%M143.0 *Condición para no setear de nuevo la Posición 13*
(R)
%M143.1 *Condición para no setear de nuevo la Posición 14*
(R)
%M143.2 *Condición para no setear de nuevo la Posición 15*
(R)
%M143.3 *Condición para no setear de nuevo la Posición 16*
(R)
%M143.4 *Condición para no setear de nuevo la Posición 17*
(R)
%M143.5 *Condición para no setear de nuevo la Posición 18*
(R)
%M143.6 *Condición para no setear de nuevo la Posición 19*
(R)
%M143.7 *Condición para no setear de nuevo la Posición 20*
(R)
%M144.0 *Condición para no setear de nuevo la Posición 21*
(R)
%M144.1 *Condición para no setear de nuevo la Posición 22*
(R)
%M144.2 *Condición para no setear de nuevo la Posición 23*
(R)
%M144.3 *Condición para no setear de nuevo la Posición 24*
(R)
%M144.4 *Condición para no setear de nuevo la Posición 25*
(R)
%M144.5

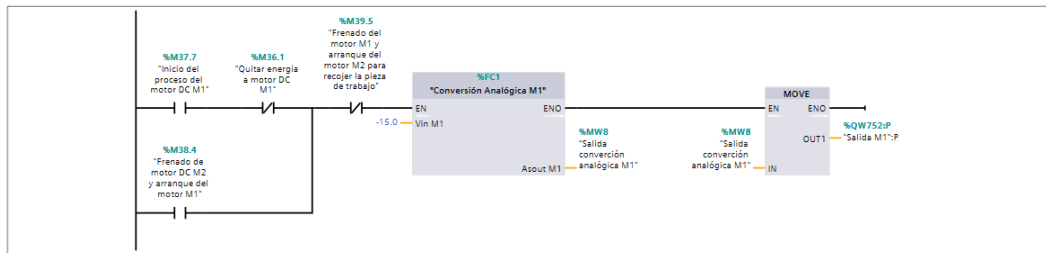
	Condición para no setear de nuevo la Posición 26 (R)
	%M144.6 *Condición para no setear de nuevo la Posición 27* (R)
	%M144.7 *Condición para no setear de nuevo la Posición 28* (R)
	%M145.0 *Condición para no setear de nuevo la Posición 29* (R)
	%M145.1 *Condición para no setear de nuevo la Posición 30* (R)
	%M145.2 *Condición para no setear de nuevo la Posición 31* (R)
	%M145.3 *Condición para no setear de nuevo la Posición 32* (R)
	%M145.4 *Condición para no setear de nuevo la Posición 33* (R)
	%M145.5 *Condición para no setear de nuevo la Posición 34* (R)
	%M145.6 *Condición para no setear de nuevo la Posición 35* (R)
	%M145.7 *Ocupada Pieza 1* (R)
	%M146.0 *Ocupada Pieza 2* (R)
	%M146.1 *Ocupada Pieza 3* (R)
	%M146.2 *Ocupada Pieza 4* (R)
	%M146.3 *Ocupada Pieza 5* (R)
	%M146.4 *Ocupada Pieza 6* (R)
	%M146.5 *Ocupada Pieza 7* (R)

	%M146.6 *Ocupada Pieza 8"	(R)
	%M146.7 *Ocupada Pieza 9"	(R)
	%M147.0 *Ocupada Pieza 10"	(R)
	%M147.1 *Ocupada Pieza 11"	(R)
	%M147.2 *Ocupada Pieza 12"	(R)
	%M147.3 *Ocupada Pieza 13"	(R)
	%M147.4 *Ocupada Pieza 14"	(R)
	%M147.5 *Ocupada Pieza 15"	(R)
	%M147.6 *Ocupada Pieza 16"	(R)
	%M147.7 *Ocupada Pieza 17"	(R)
	%M148.0 *Ocupada Pieza 18"	(R)
	%M148.1 *Ocupada Pieza 19"	(R)
	%M148.2 *Ocupada Pieza 20"	(R)
	%M148.3 *Ocupada Pieza 21"	(R)
	%M148.4 *Ocupada Pieza 22"	(R)
	%M148.5 *Ocupada Pieza 23"	(R)
	%M148.6 *Ocupada Pieza 24"	(R)
	%M148.7 *Ocupada Pieza 25"	(R)
	%M149.0 *Ocupada Pieza 26"	(R)
	%M149.1 *Ocupada Pieza 27"	(R)



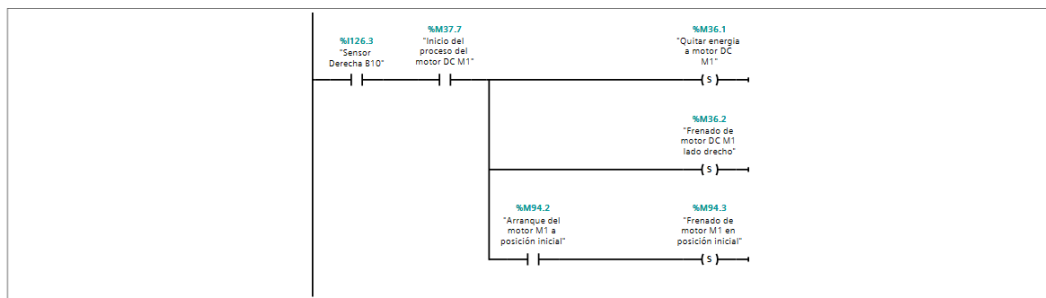
Segmento 7: Ubicación motor M1 a posición inicial.

Ubicación del motor DC M1 a la posición inicial.



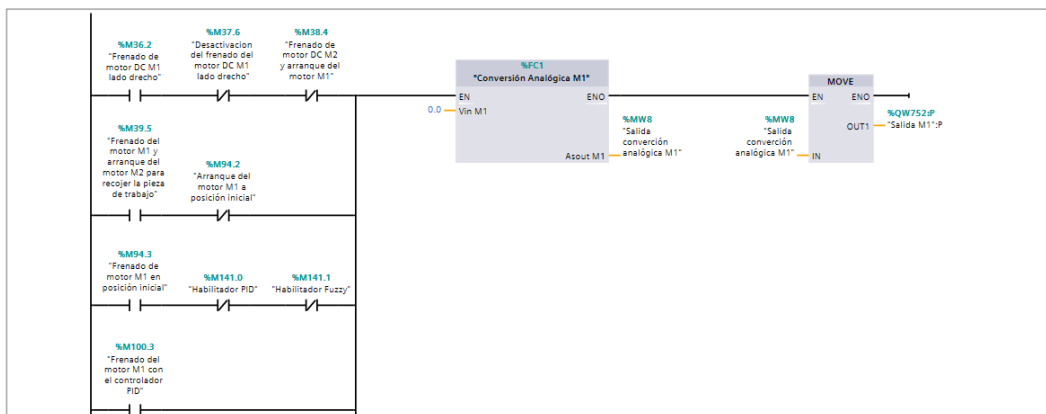
Segmento 8: Detección sensor B10.

Detección del motor DC M1 en la posición inicial sensor B10.



Segmento 9: Frenado motor M1 posición inicial.

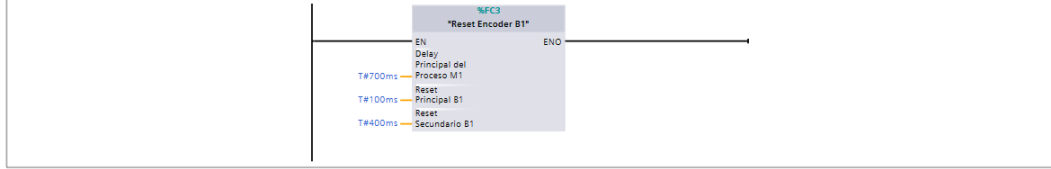
Frenado del motor DC M1 en la posición inicial sensor B10.





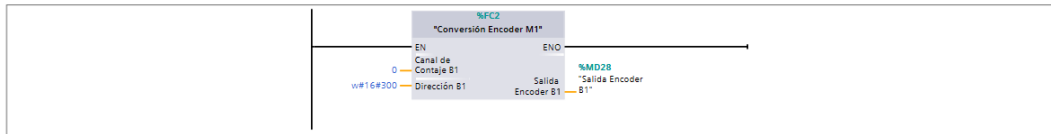
Segmento 10: Reset Encoder B1.

Tiempo de reseteo del encoder B1.



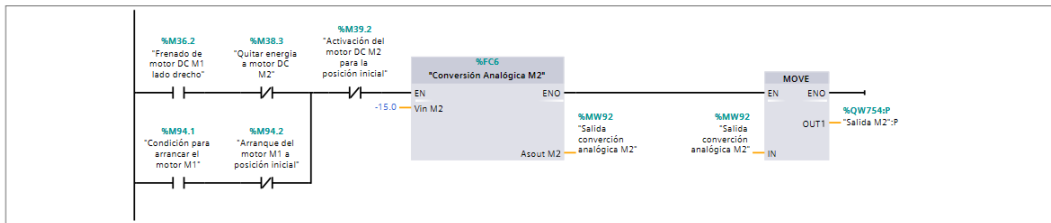
Segmento 11: Conversión Encoder M1.

Conversión de datos del encoder M1.



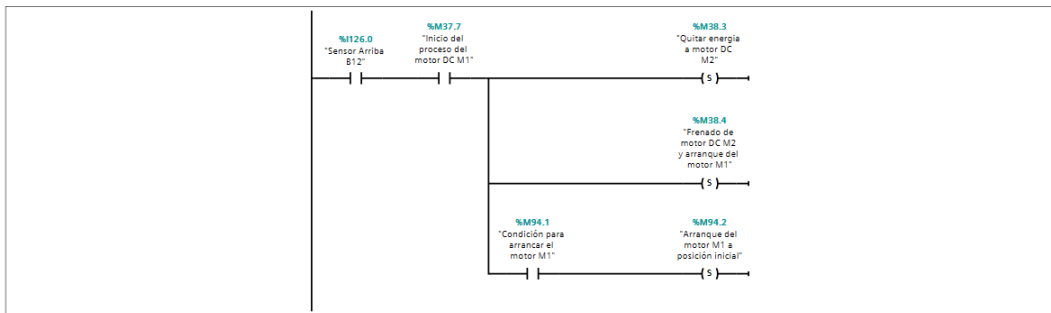
Segmento 12: Ubicación motor M2 posición inicial.

Ubicación del motor DC M2 a la posición inicial.



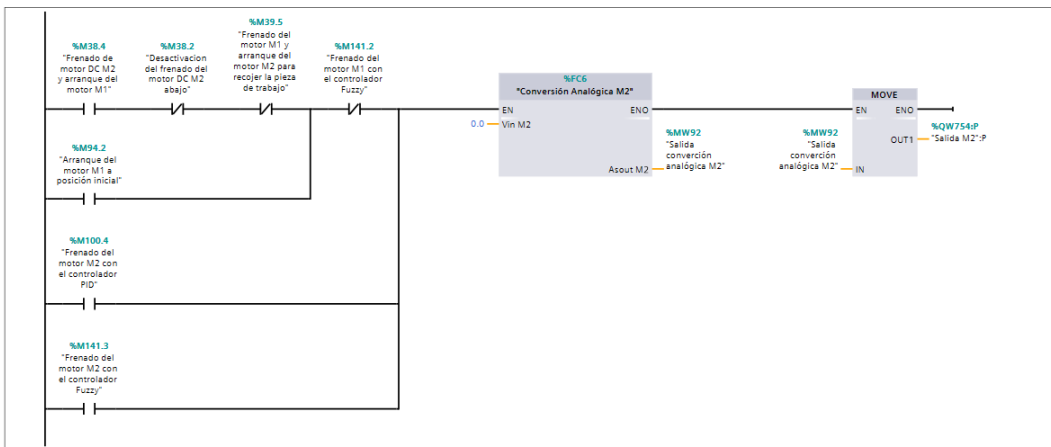
Segmento 13: Detección motor M2 posición inicial.

Detección del motor DC M2 en la posición inicial sensor B12.



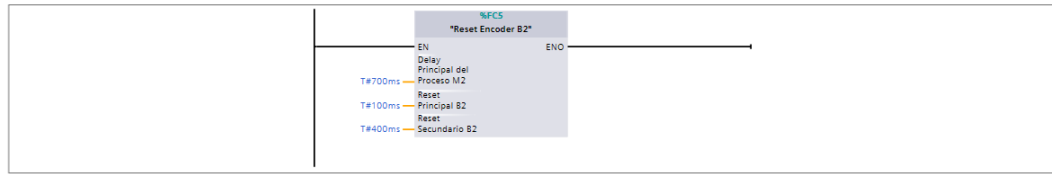
Segmento 14: Frenado motor M2 posición inicial.

Frenado del motor DC M2 en la posición inicial sensor B12.



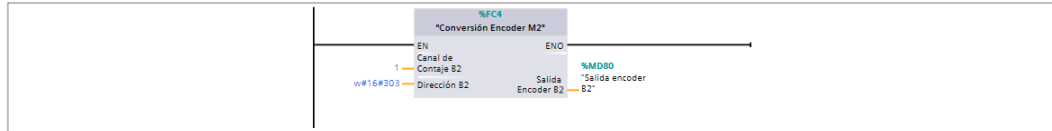
Segmento 15: Reset Encoder B2.

Tiempo de reseteo del encoder B2.



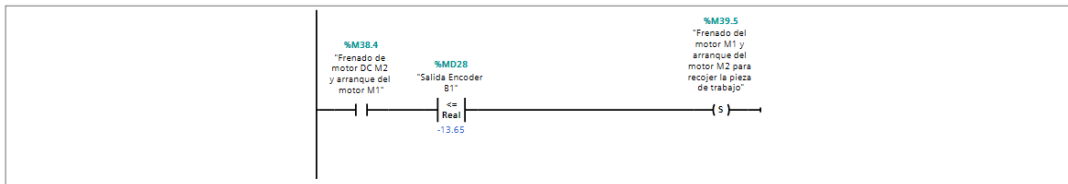
Segmento 16: Conversión Encoder M2.

Conversión de datos del encoder M2.



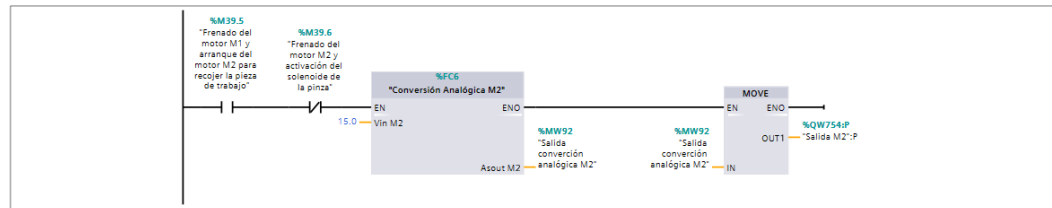
Segmento 17: Condición motor M1 detenerse.

Condición para que el motor M1 se detenga.



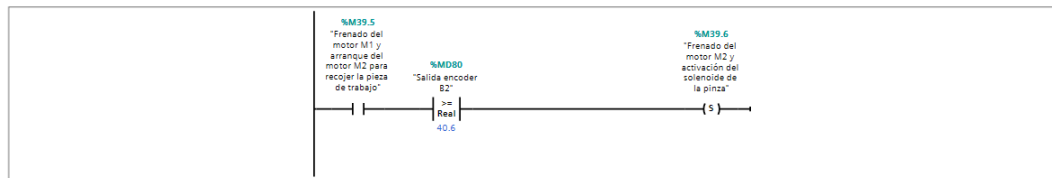
Segmento 18: Arranque motor M2 recolección de pieza.

Arranque del motor M2 para recoger la pieza de trabajo.



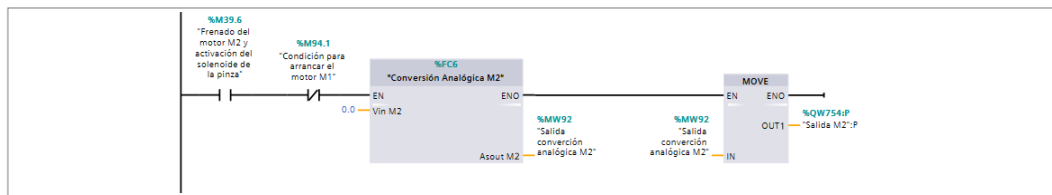
Segmento 19: Condición motor M2 detenerse.

Condición para que el motor M2 se detenga para obtener la pieza de trabajo.



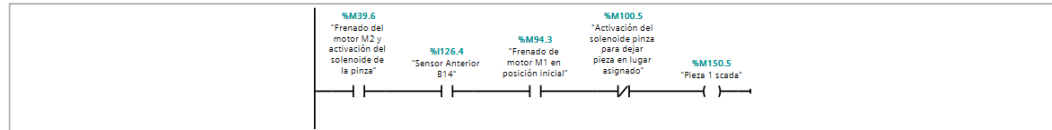
Segmento 20: Frenado motor M2 recolección pieza.

Frenado del motor M2 para recoger la pieza de trabajo.



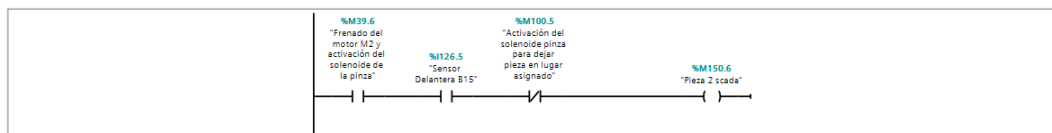
Segmento 21: Condición 1 para la visibilidad de la pieza en scada.

Condición 1 para la visibilidad de la pieza de trabajo en el scada.



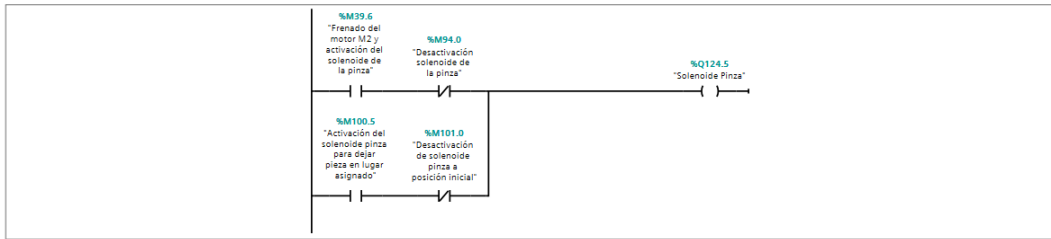
Segmento 22: Condición 2 para la visibilidad de la pieza en scada.

Condición 2 para la visibilidad de la pieza de trabajo en el scada.



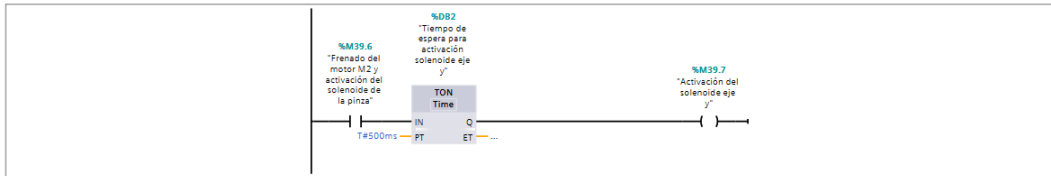
Segmento 23: Activación solenoide pinza.

Activación del solenoide de la pinza.



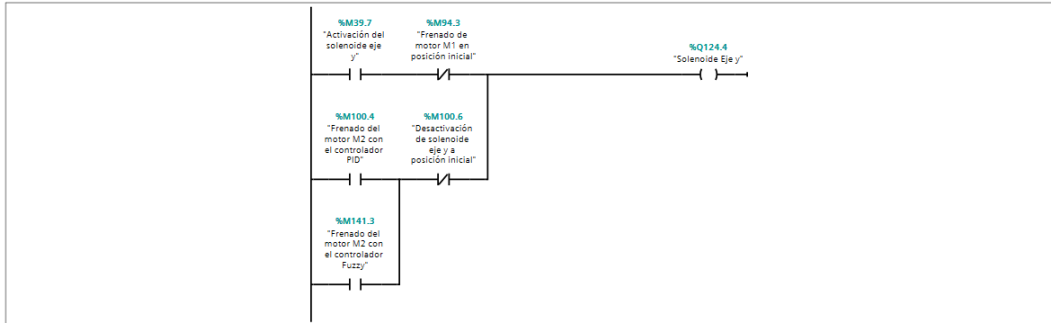
Segmento 24: Tiempo de espera activación solenoide eje y.

Tiempo de espera para la activación del solenoide del eje y.



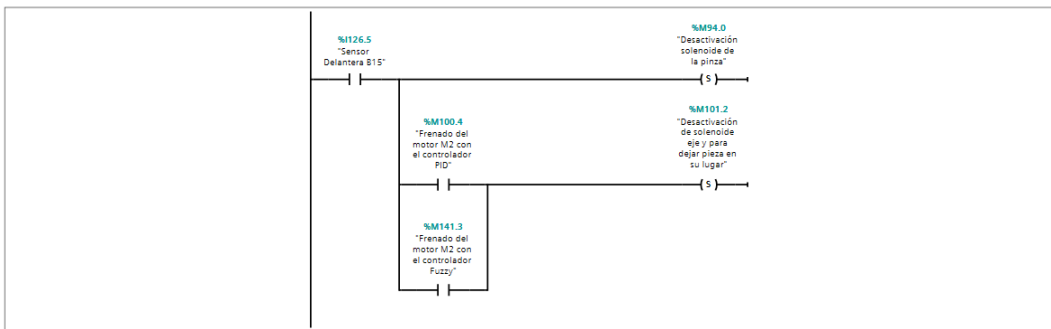
Segmento 25: Activación solenoide eje y.

Activación del solenoide del eje y.



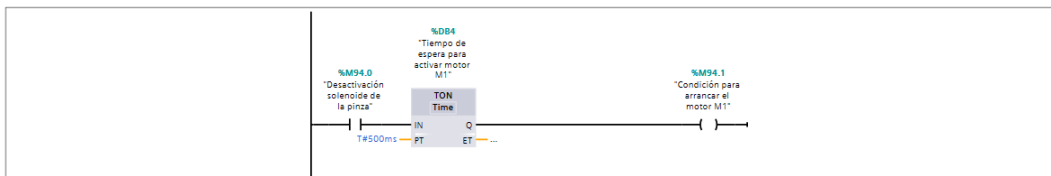
Segmento 26: Desactivación solenoide pinza.

Desactivación del solenoide de la pinza.



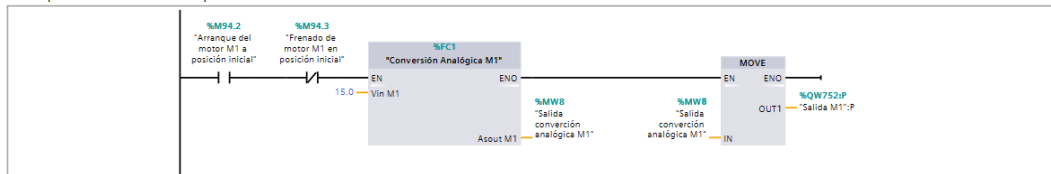
Segmento 27: Tiempo de espera activación M1.

Tiempo de espera para la activación del motor M1.



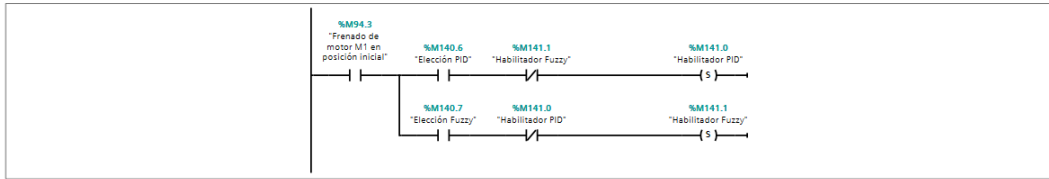
Segmento 28: Arranque motor M1 a posición inicial.

Arranque del motor M1 hacia posición inicial.



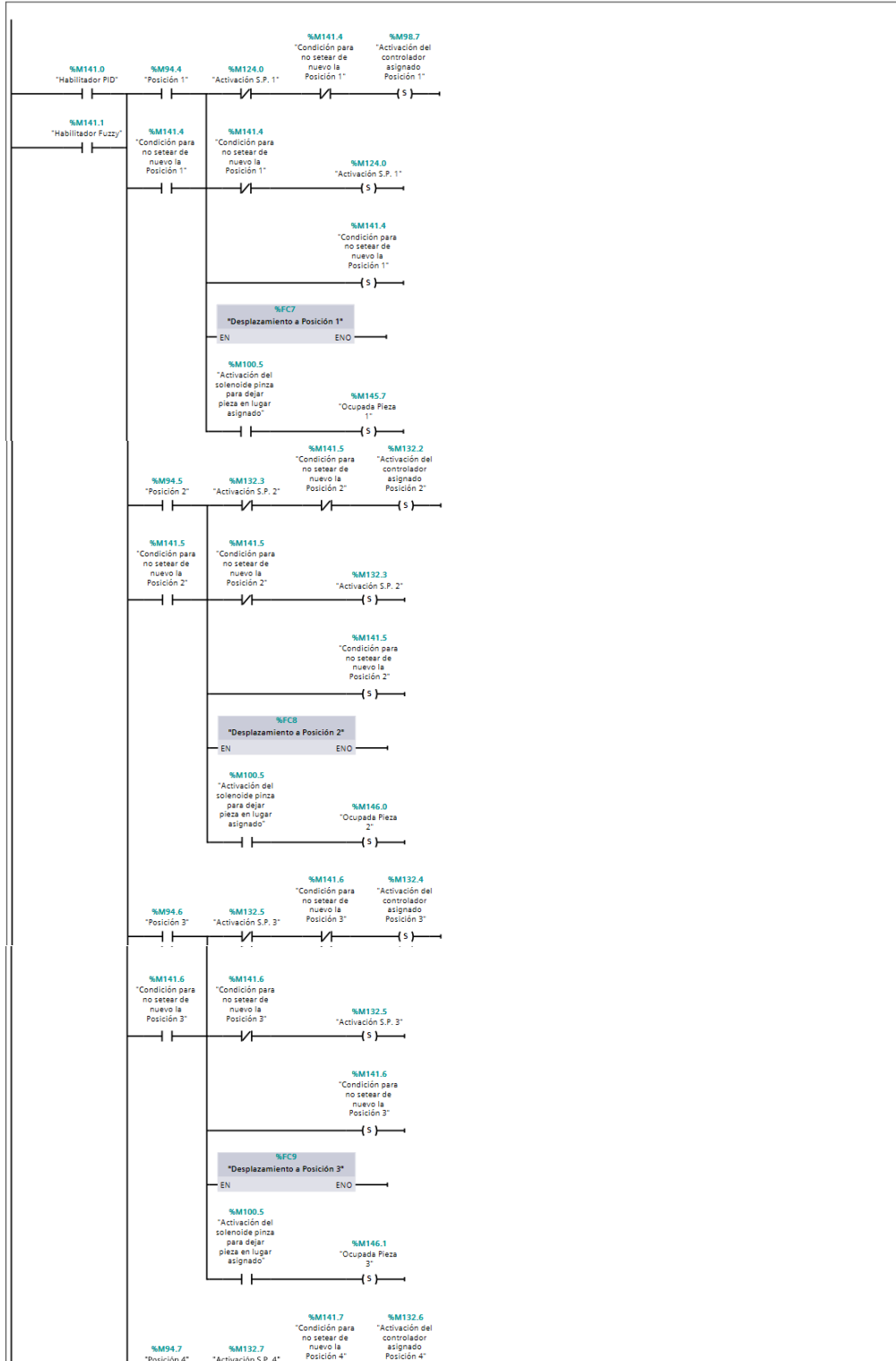
Segmento 29: Habilitador PID o Fuzzy.

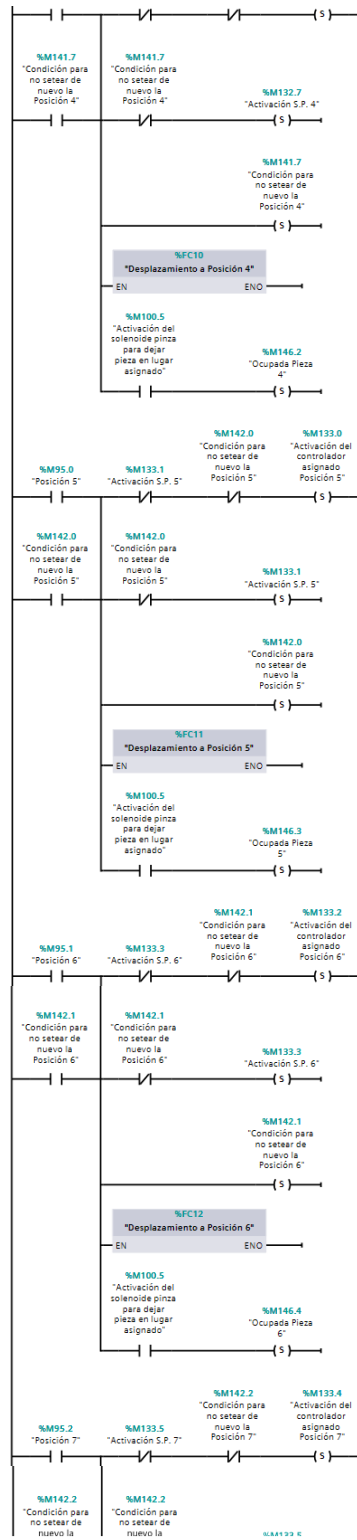
Elección del controlador a trabajar en la planta.

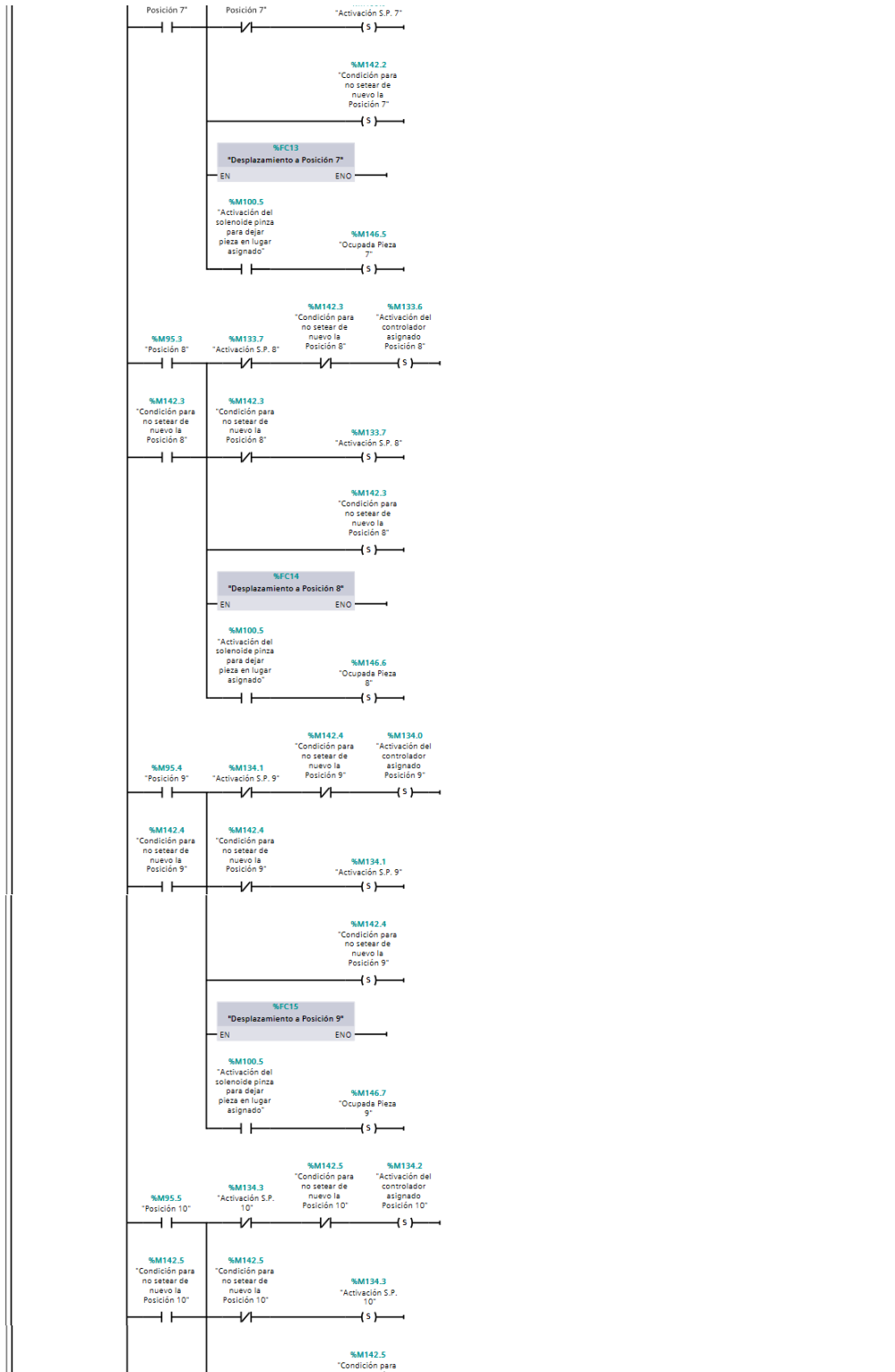


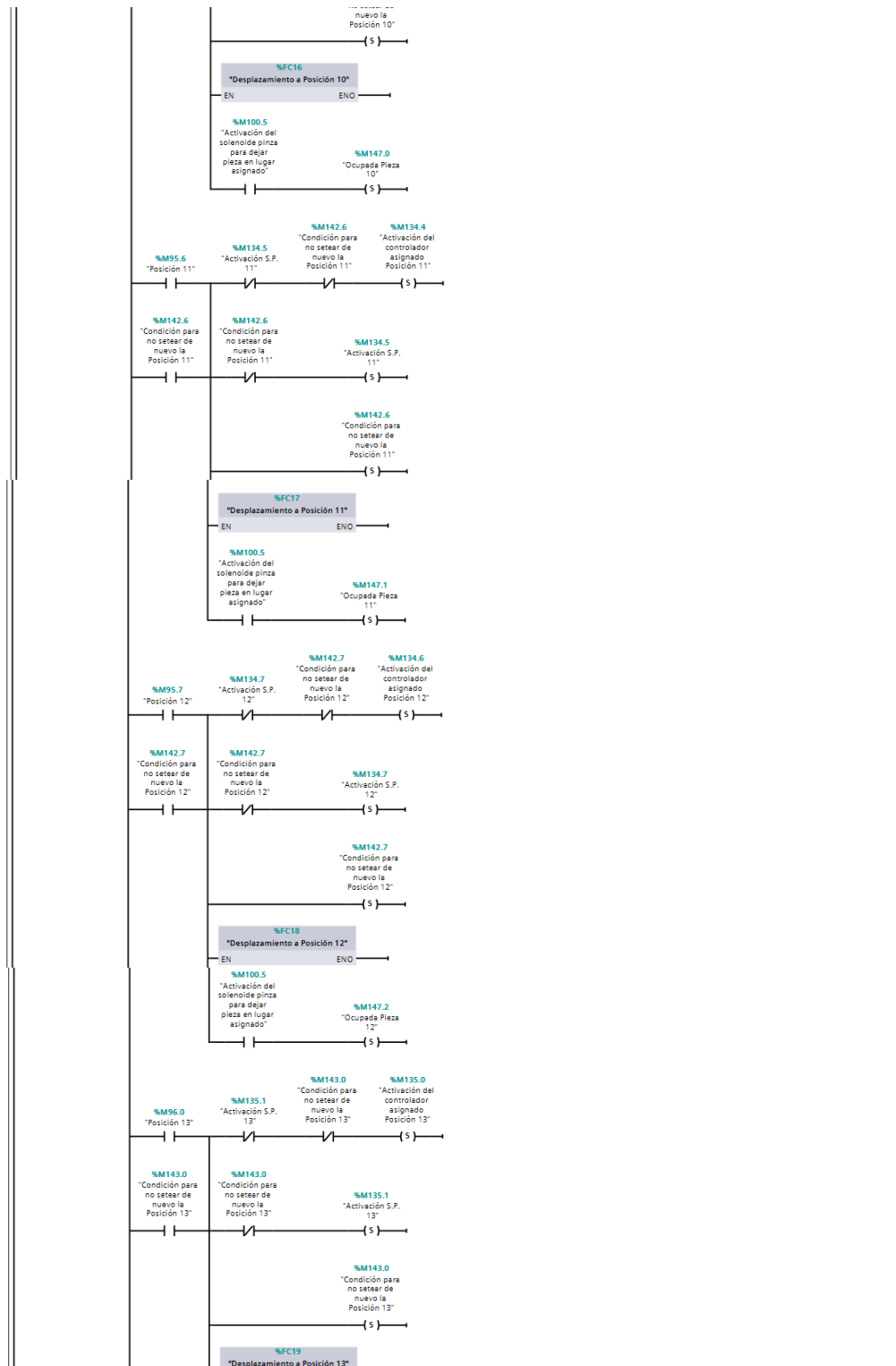
Segmento 30: Elección posición de almacenamiento.

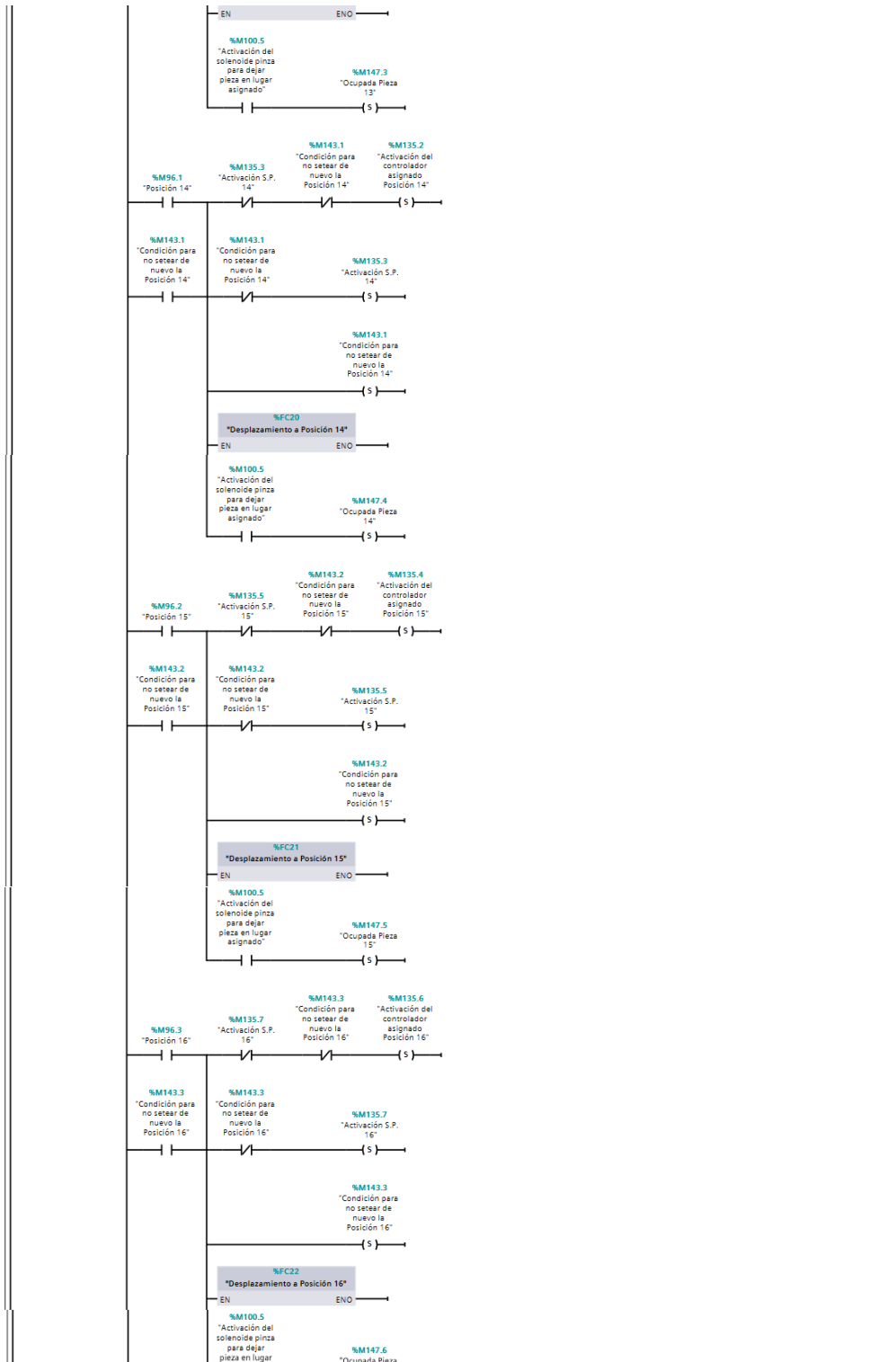
Frenado del motor M1 en posición inicial y esperando por elección de ubicación de pieza de trabajo.

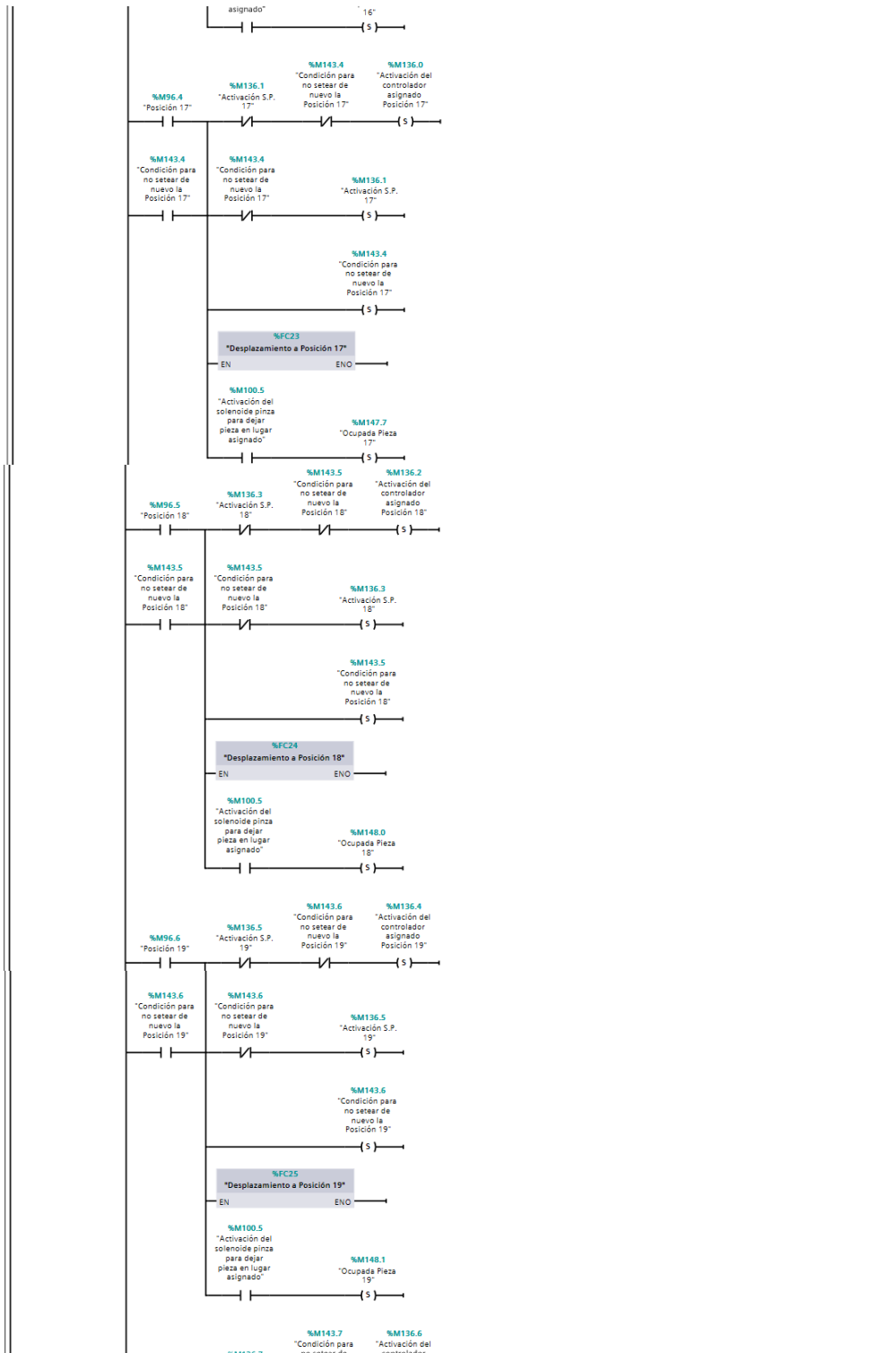


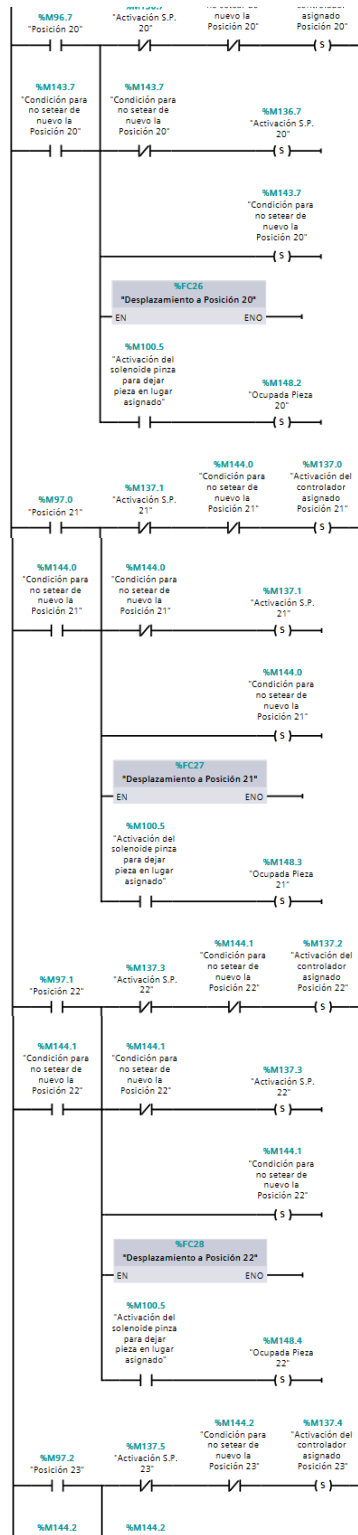


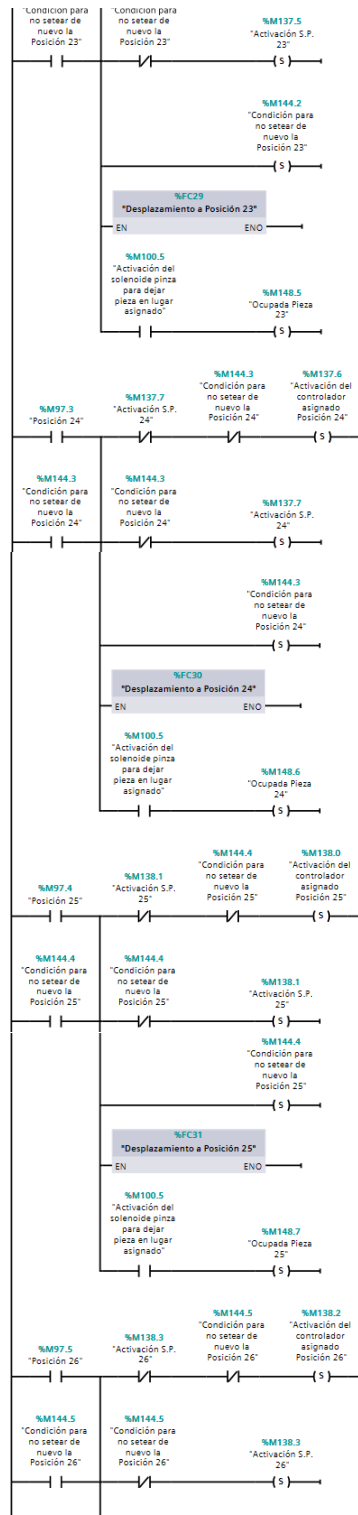


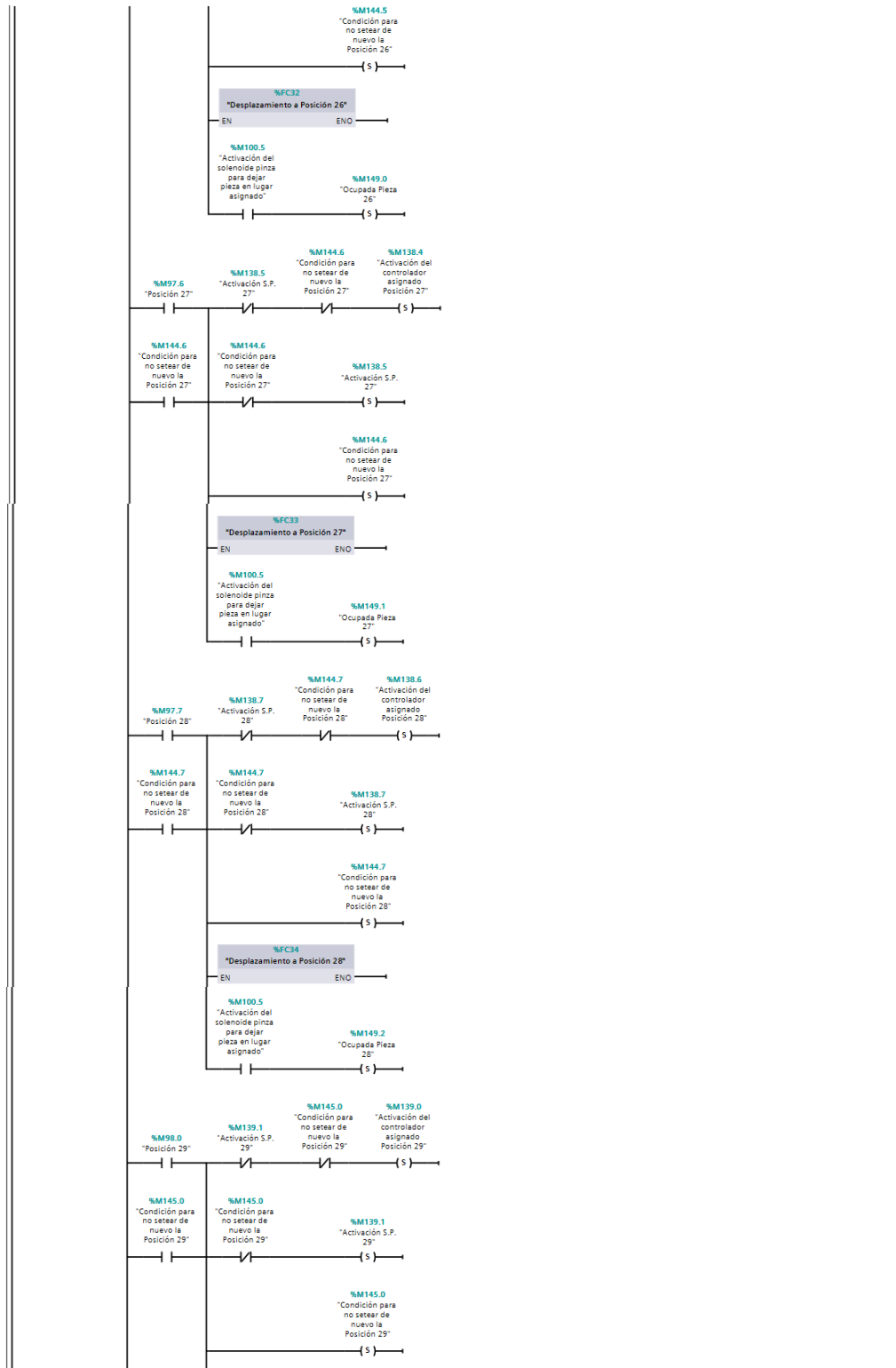


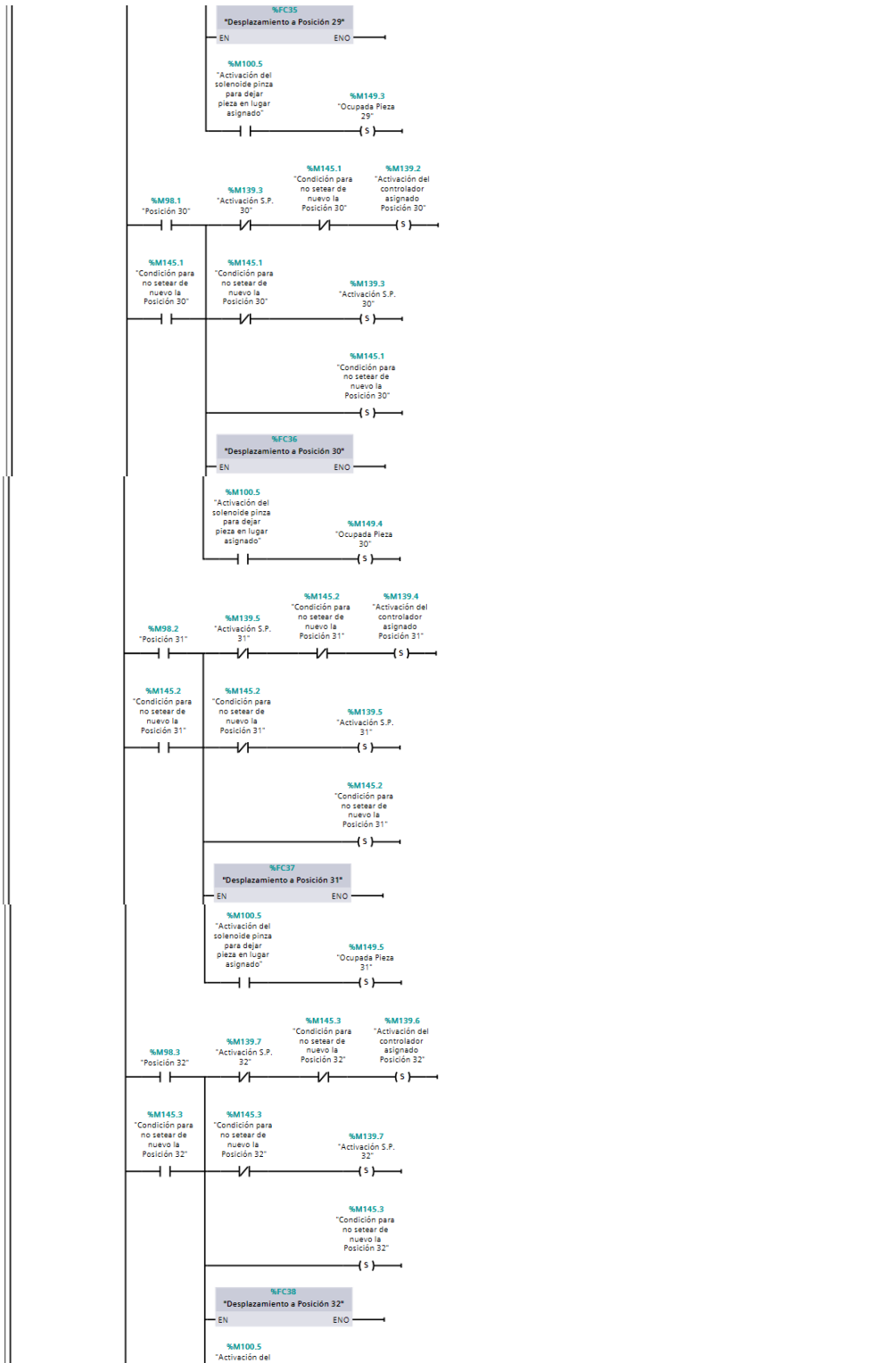


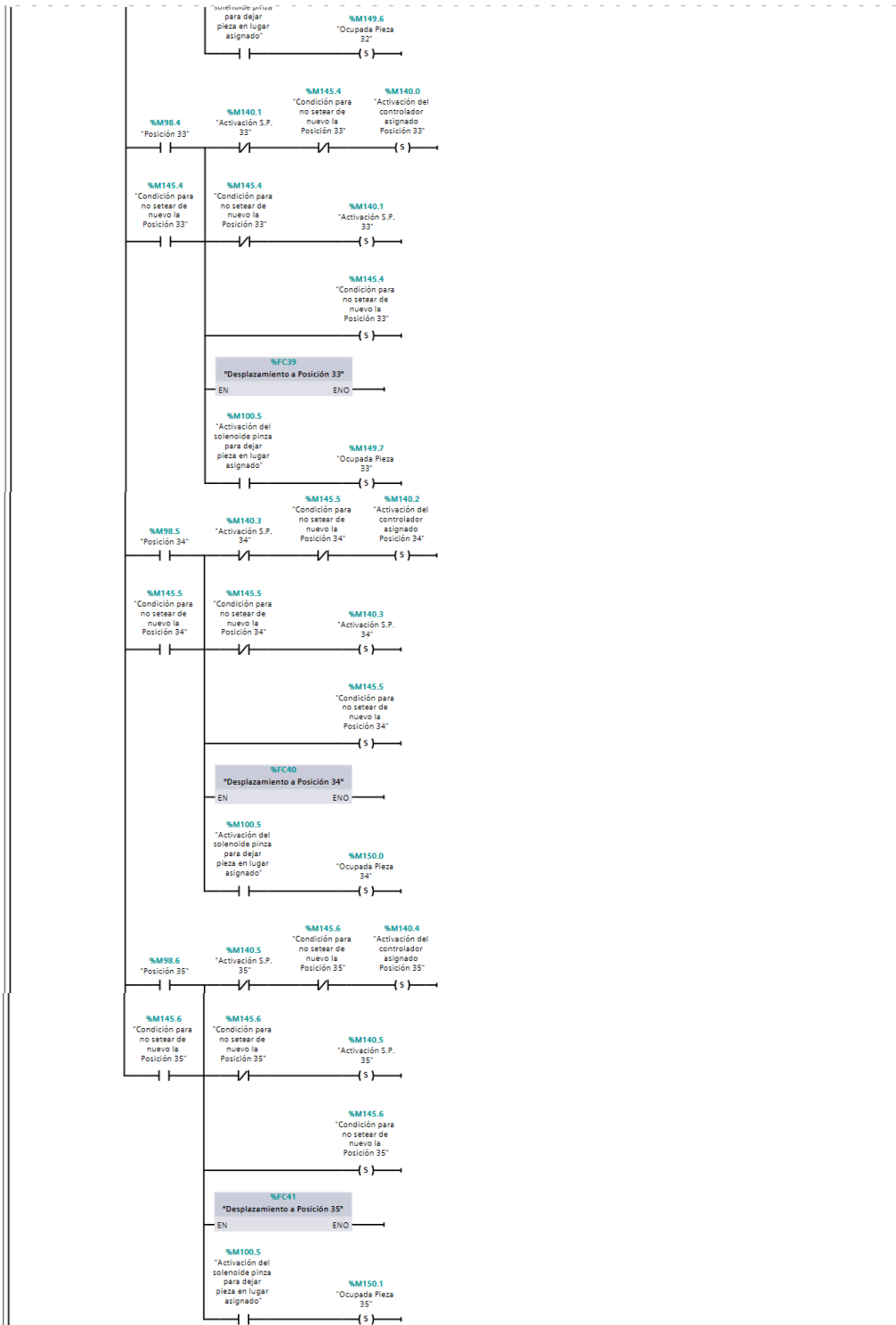






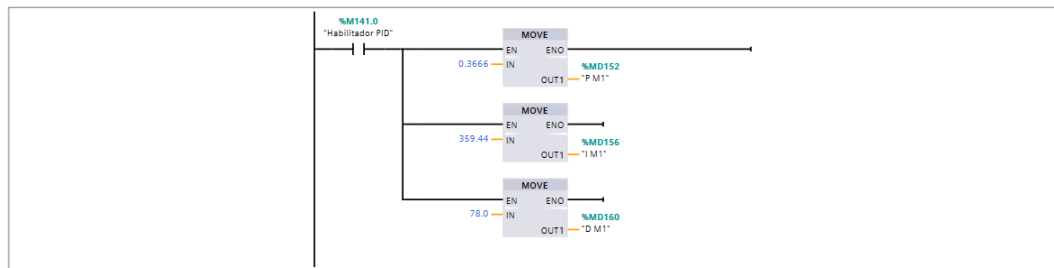






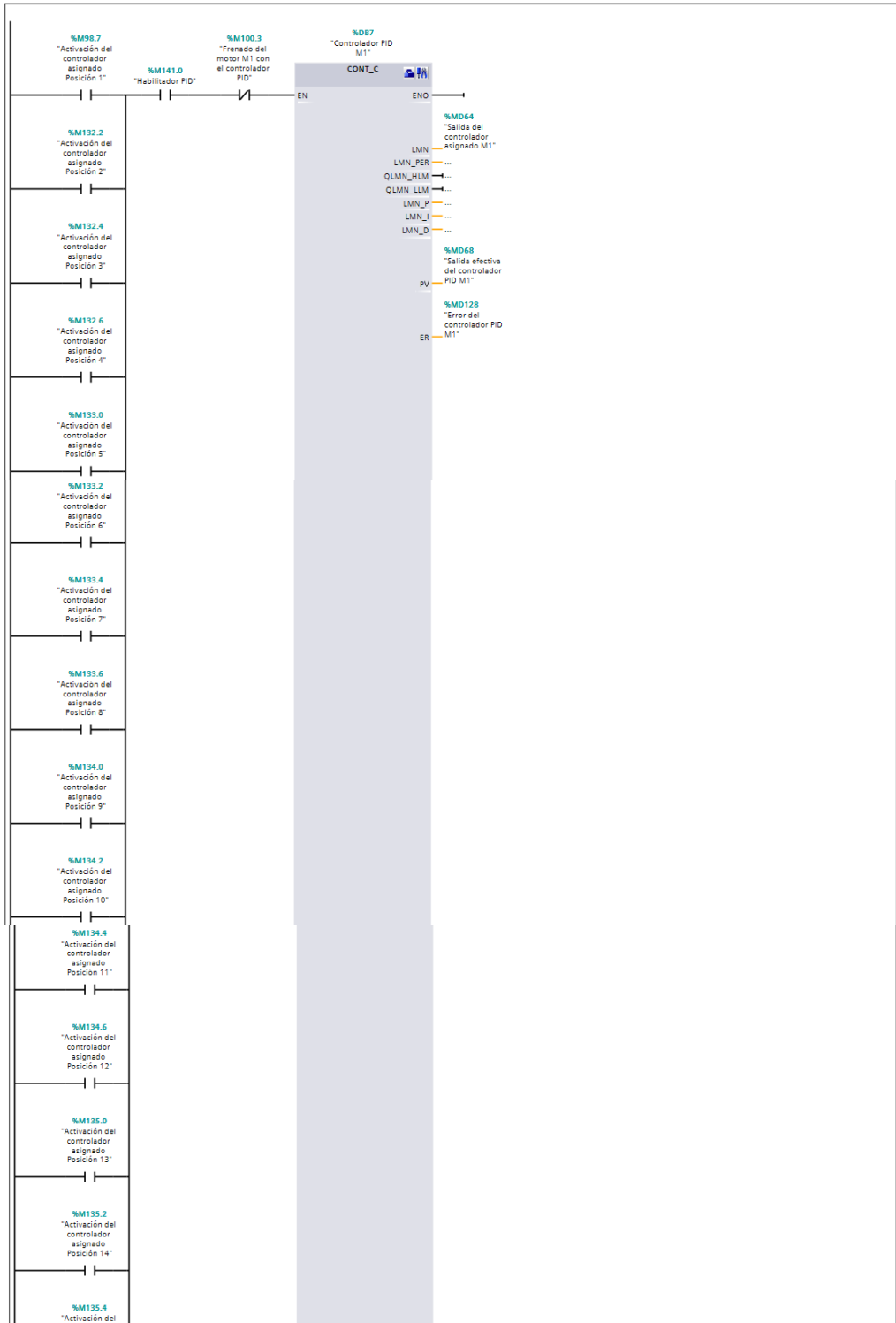
Segmento 31: Valores P,I y D del controlador PID motor M1.

Valores proporcional, integral y derivativo del controlador PID del motor M1.

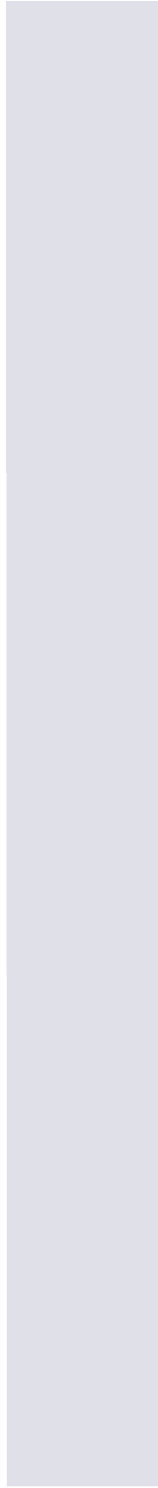


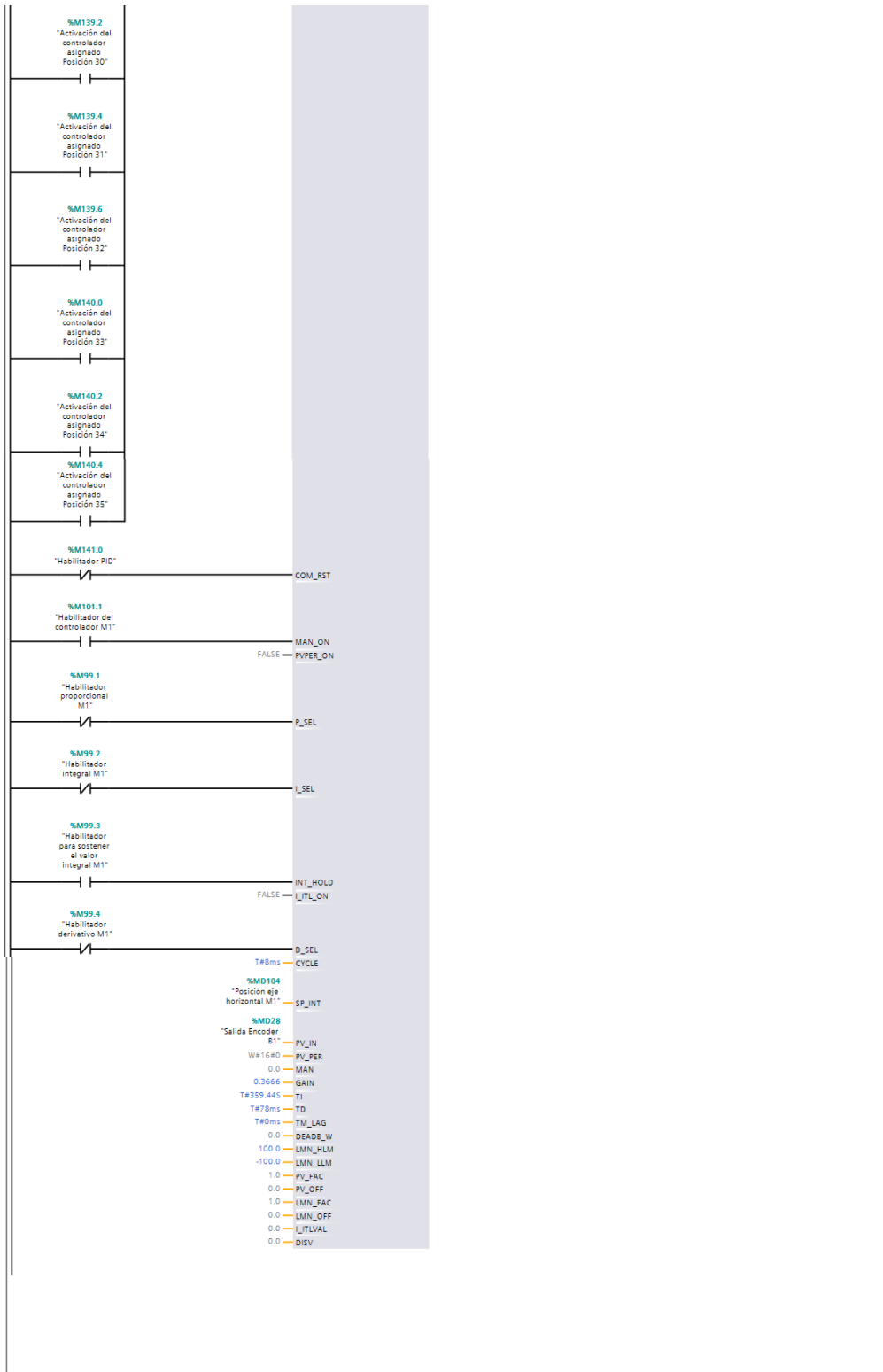
Segmento 32: Controlador PID motor M1.

Activación del controlador PID del motor M1.



<p>Controlador asignado Posición 15"</p>
<p>%M135.6 "Activación del controlador asignado Posición 16"</p>
<p>%M136.0 "Activación del controlador asignado Posición 17"</p>
<p>%M136.2 "Activación del controlador asignado Posición 18"</p>
<p>%M136.4 "Activación del controlador asignado Posición 19"</p>
<p>%M136.6 "Activación del controlador asignado Posición 20"</p>
<p>%M137.0 "Activación del controlador asignado Posición 21"</p>
<p>%M137.2 "Activación del controlador asignado Posición 22"</p>
<p>%M137.4 "Activación del controlador asignado Posición 23"</p>
<p>%M137.6 "Activación del controlador asignado Posición 24"</p>
<p>%M138.0 "Activación del controlador asignado Posición 25"</p>
<p>%M138.2 "Activación del controlador asignado Posición 26"</p>
<p>%M138.4 "Activación del controlador asignado Posición 27"</p>
<p>%M138.6 "Activación del controlador asignado Posición 28"</p>
<p>%M139.0 "Activación del controlador asignado Posición 29"</p>





Segmento 33: Controlador Fuzzy motor M1.

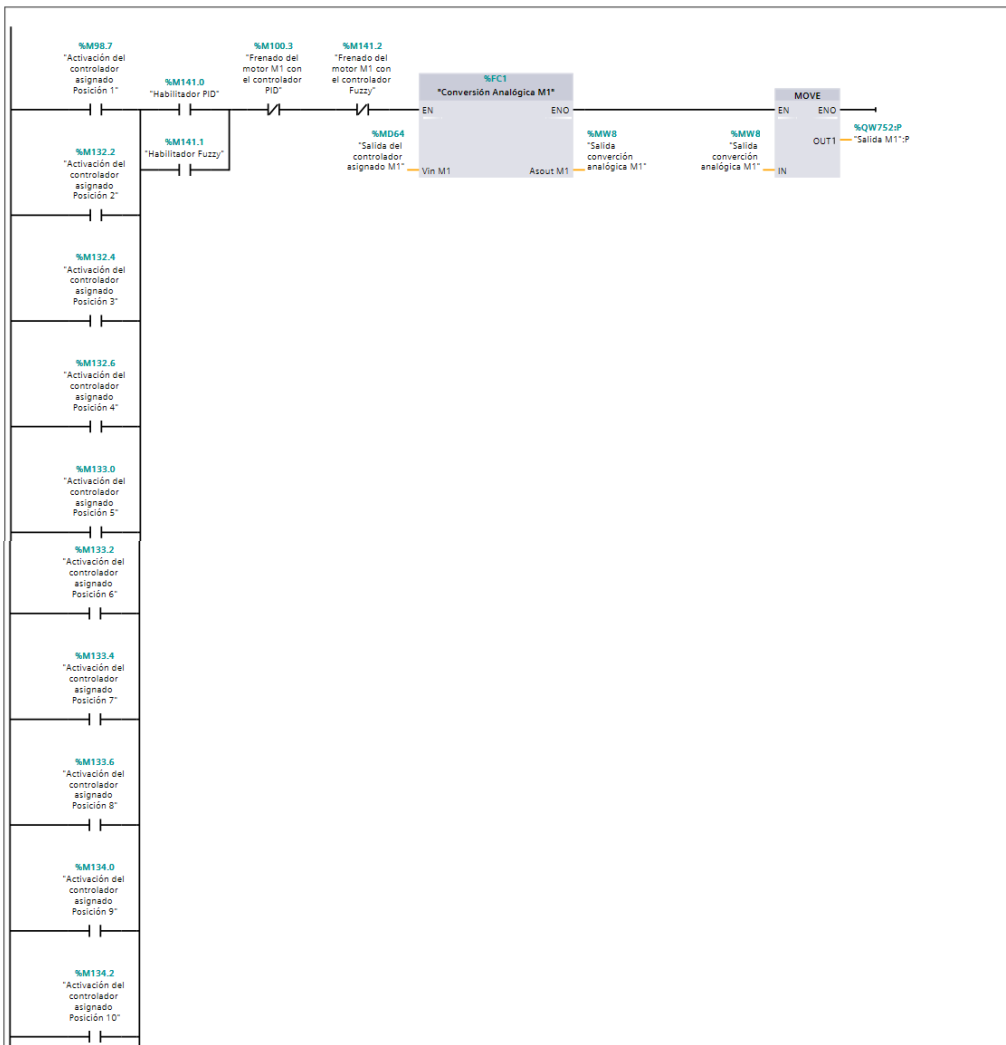
Activación del controlador Fuzzy del motor M1.

<p>Controlador asignado Posición 15"</p>
<p>%M135.6 "Activación del controlador asignado Posición 16"</p>
<p>%M136.0 "Activación del controlador asignado Posición 17"</p>
<p>%M136.2 "Activación del controlador asignado Posición 18"</p>
<p>%M136.4 "Activación del controlador asignado Posición 19"</p>
<p>%M136.6 "Activación del controlador asignado Posición 20"</p>
<p>%M137.0 "Activación del controlador asignado Posición 21"</p>
<p>%M137.2 "Activación del controlador asignado Posición 22"</p>
<p>%M137.4 "Activación del controlador asignado Posición 23"</p>
<p>%M137.6 "Activación del controlador asignado Posición 24"</p>
<p>%M138.0 "Activación del controlador asignado Posición 25"</p>
<p>%M138.2 "Activación del controlador asignado Posición 26"</p>
<p>%M138.4 "Activación del controlador asignado Posición 27"</p>
<p>%M138.6 "Activación del controlador asignado Posición 28"</p>
<p>%M139.0 "Activación del controlador asignado Posición 29"</p>



Segmento 34: Arranque motor M1 controlador PID o Fuzzy.

Arranque del motor M1 con el controlador PID o con el controlador Fuzzy.

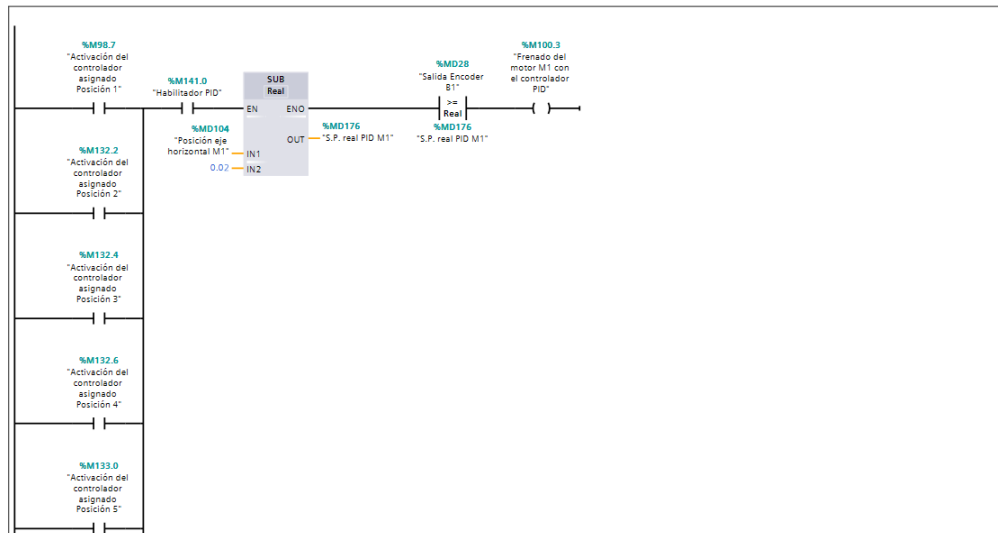


<p>%M134.4 "Activación del controlador asignado Posición 11"</p>
<p>%M134.6 "Activación del controlador asignado Posición 12"</p>
<p>%M135.0 "Activación del controlador asignado Posición 13"</p>
<p>%M135.2 "Activación del controlador asignado Posición 14"</p>
<p>%M135.4 "Activación del controlador asignado Posición 15"</p>
<p>%M135.6 "Activación del controlador asignado Posición 16"</p>
<p>%M136.0 "Activación del controlador asignado Posición 17"</p>
<p>%M136.2 "Activación del controlador asignado Posición 18"</p>
<p>%M136.4 "Activación del controlador asignado Posición 19"</p>
<p>%M136.6 "Activación del controlador asignado Posición 20"</p>
<p>%M137.0 "Activación del controlador asignado Posición 21"</p>
<p>%M137.2 "Activación del controlador asignado Posición 22"</p>
<p>%M137.4 "Activación del controlador asignado Posición 23"</p>
<p>%M137.6 "Activación del controlador asignado Posición 24"</p>



Segmento 35: Condición frenado motor M1 PID.

Condición de frenado de motor M1 con el controlador PID.



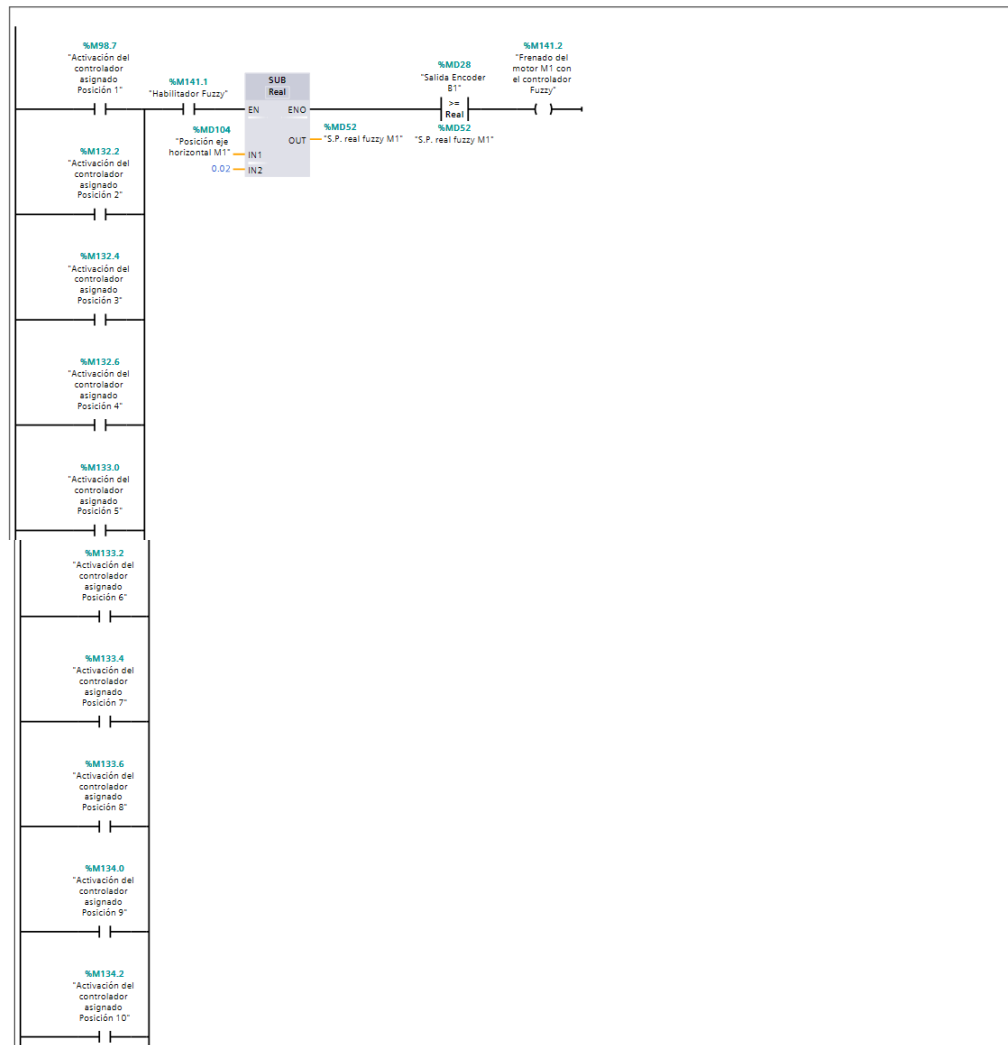


<p>..... asignado Posición 15"</p>
<p>%M135.6 "Activación del controlador asignado Posición 16"</p>
<p>%M136.0 "Activación del controlador asignado Posición 17"</p>
<p>%M136.2 "Activación del controlador asignado Posición 18"</p>
<p>%M136.4 "Activación del controlador asignado Posición 19"</p>
<p>%M136.6 "Activación del controlador asignado Posición 20"</p>
<p>%M137.0 "Activación del controlador asignado Posición 21"</p>
<p>%M137.2 "Activación del controlador asignado Posición 22"</p>
<p>%M137.4 "Activación del controlador asignado Posición 23"</p>
<p>%M137.6 "Activación del controlador asignado Posición 24"</p>
<p>%M138.0 "Activación del controlador asignado Posición 25"</p>
<p>%M138.2 "Activación del controlador asignado Posición 26"</p>
<p>%M138.4 "Activación del controlador asignado Posición 27"</p>
<p>%M138.6 "Activación del controlador asignado Posición 28"</p>
<p>%M139.0 "Activación del controlador asignado Posición 29"</p>



Segmento 36: Condición frenado motor M1 Fuzzy.

Condición de frenado de motor M1 con el controlador Fuzzy.

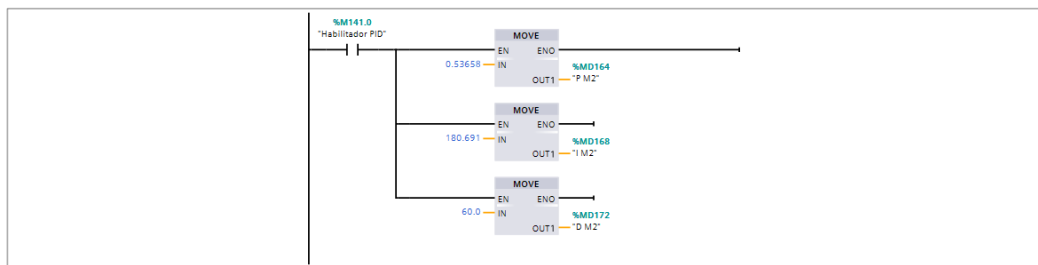


<p>%M134.4 "Activación del controlador asignado Posición 11"</p>
<p>%M134.6 "Activación del controlador asignado Posición 12"</p>
<p>%M135.0 "Activación del controlador asignado Posición 13"</p>
<p>%M135.2 "Activación del controlador asignado Posición 14"</p>
<p>%M135.4 "Activación del controlador asignado Posición 15"</p>
<p>%M135.6 "Activación del controlador asignado Posición 16"</p>
<p>%M136.0 "Activación del controlador asignado Posición 17"</p>
<p>%M136.2 "Activación del controlador asignado Posición 18"</p>
<p>%M136.4 "Activación del controlador asignado Posición 19"</p>
<p>%M136.6 "Activación del controlador asignado Posición 20"</p>
<p>%M137.0 "Activación del controlador asignado Posición 21"</p>
<p>%M137.2 "Activación del controlador asignado Posición 22"</p>
<p>%M137.4 "Activación del controlador asignado Posición 23"</p>
<p>%M137.6 "Activación del controlador asignado Posición 24"</p>



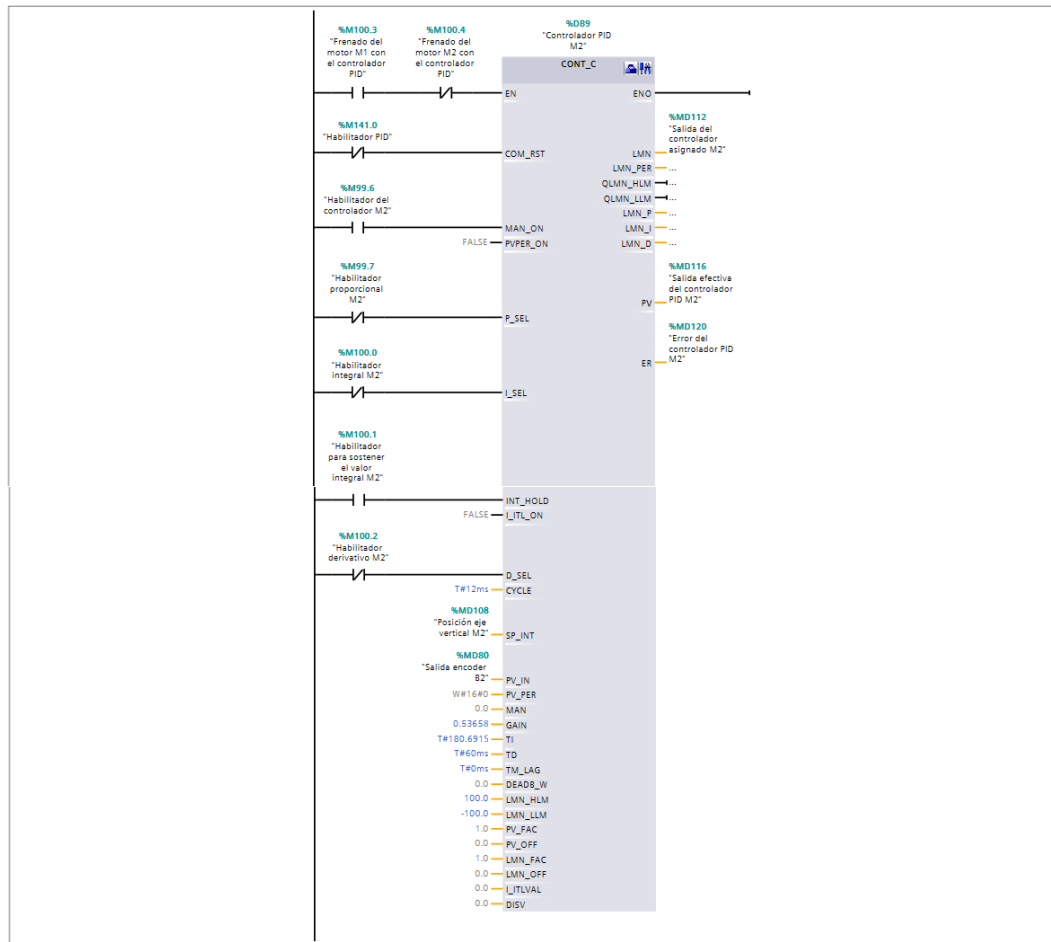
Segmento 37: Valores P,I y D del controlador PID motor M2.

Valores proporcional, integral y derivativo del controlador PID del motor M2.



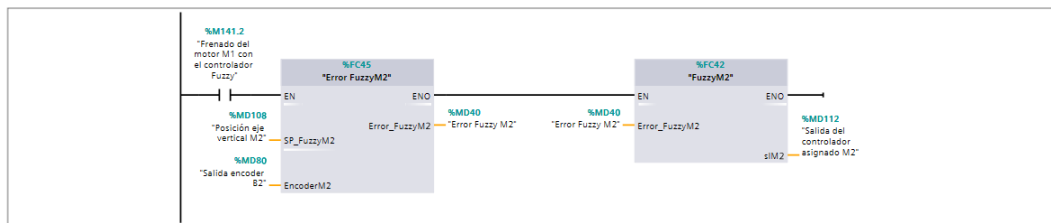
Segmento 38: Controlador PID motor M2.

Activación del controlador PID del motor M2.



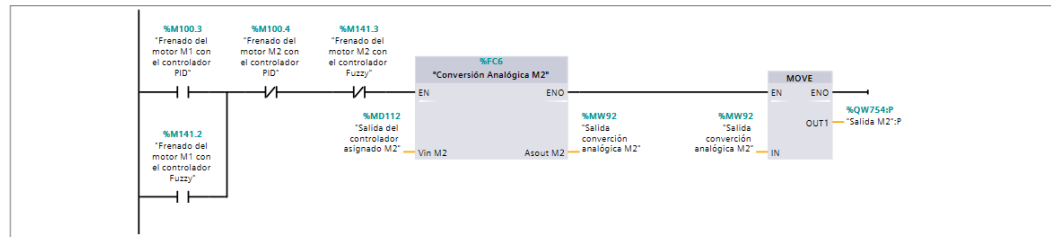
Segmento 39: Controlador Fuzzy motor M2.

Activación del controlador Fuzzy del motor M2.



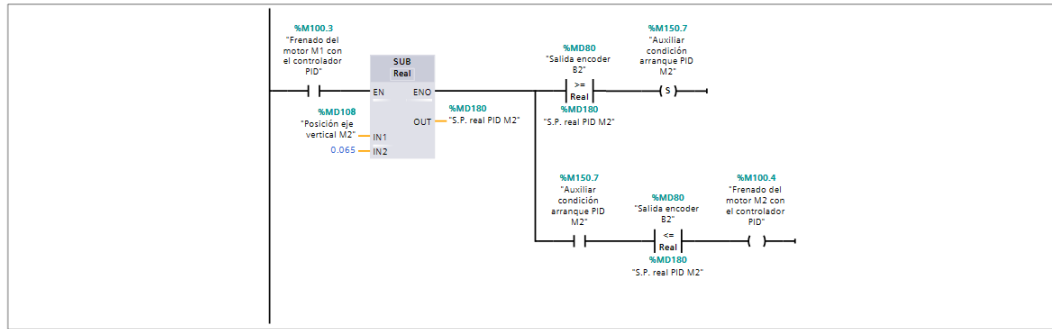
Segmento 40: Arranque motor M2 controlador PID o Fuzzy.

Arranque del motor M2 con el controlador PID.



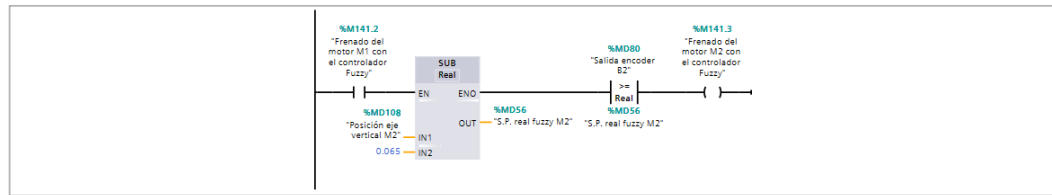
Segmento 41: Condición frenado motor M2 PID.

Condición de frenado de motor M2 con el controlador PID.



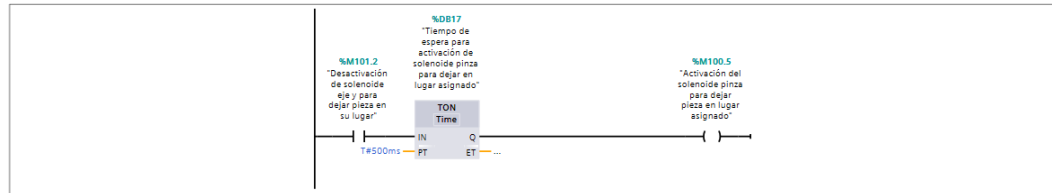
Segmento 42: Condición frenado motor M2 Fuzzy.

Condición de frenado de motor M2 con el controlador Fuzzy.



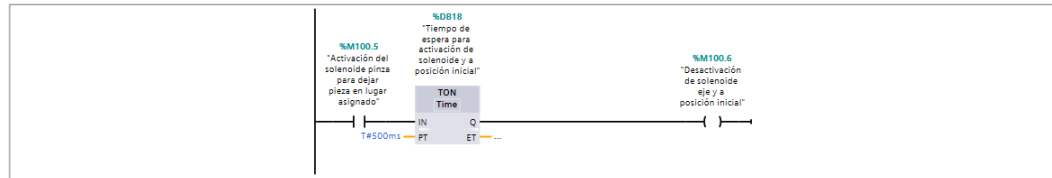
Segmento 43: Tiempo de espera activación solenoide pinza.

Tiempo de espera para activación de solenoide pinza para dejar en lugar asignado.



Segmento 44: Tiempo de espera activación solenoide a posición inicial.

Tiempo de espera para activación de solenoide y a posición inicial.



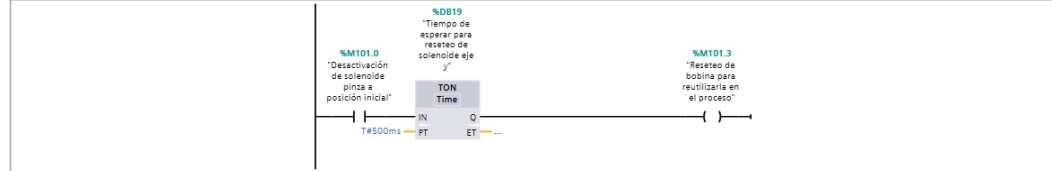
Segmento 45: Desactivación solenoide pinza a posición inicial.

Desactivación del solenoide de pinza a posición inicial.



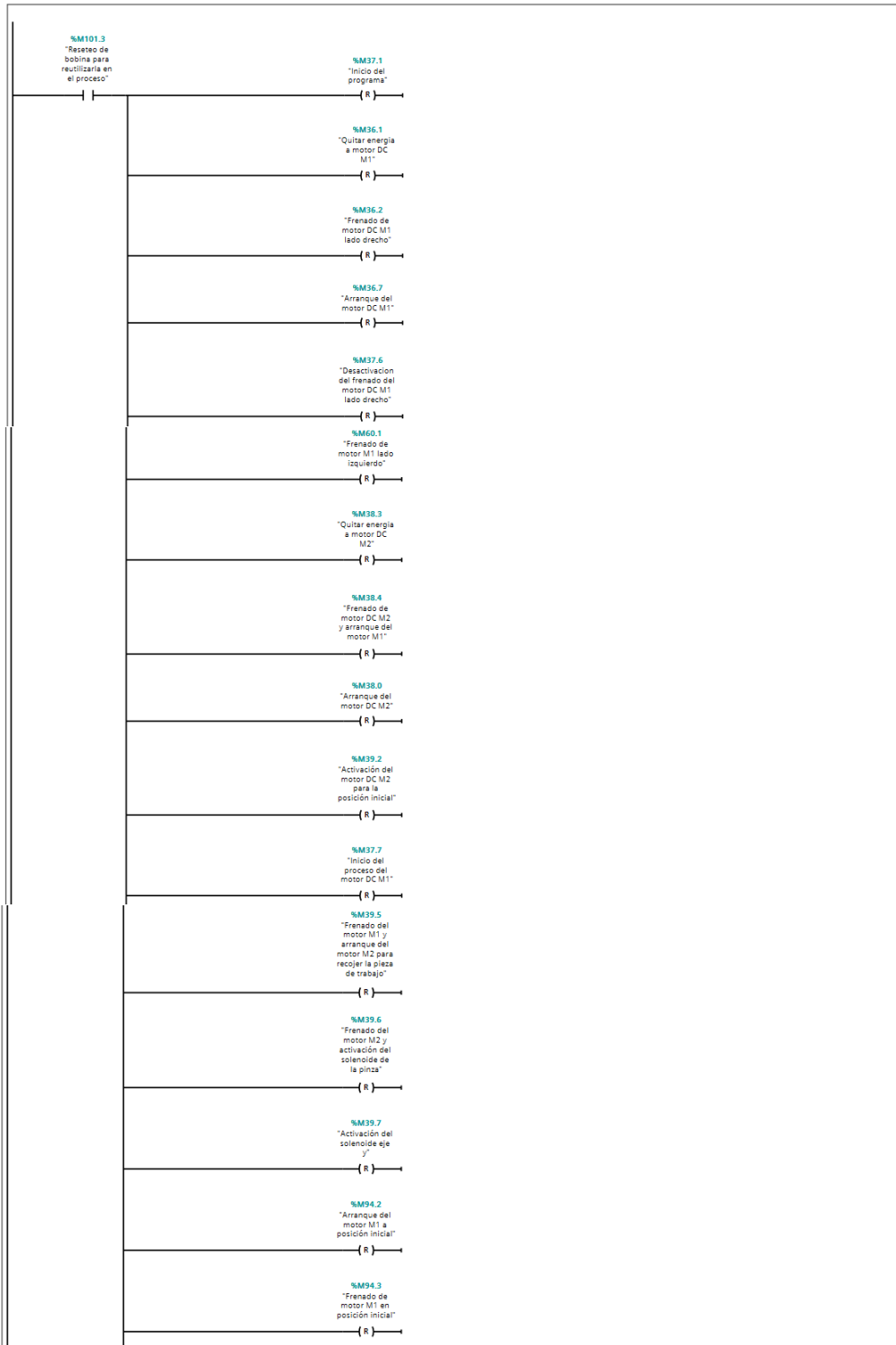
Segmento 46: Tiempo de espera reseteo de bobina solenoide eje y.

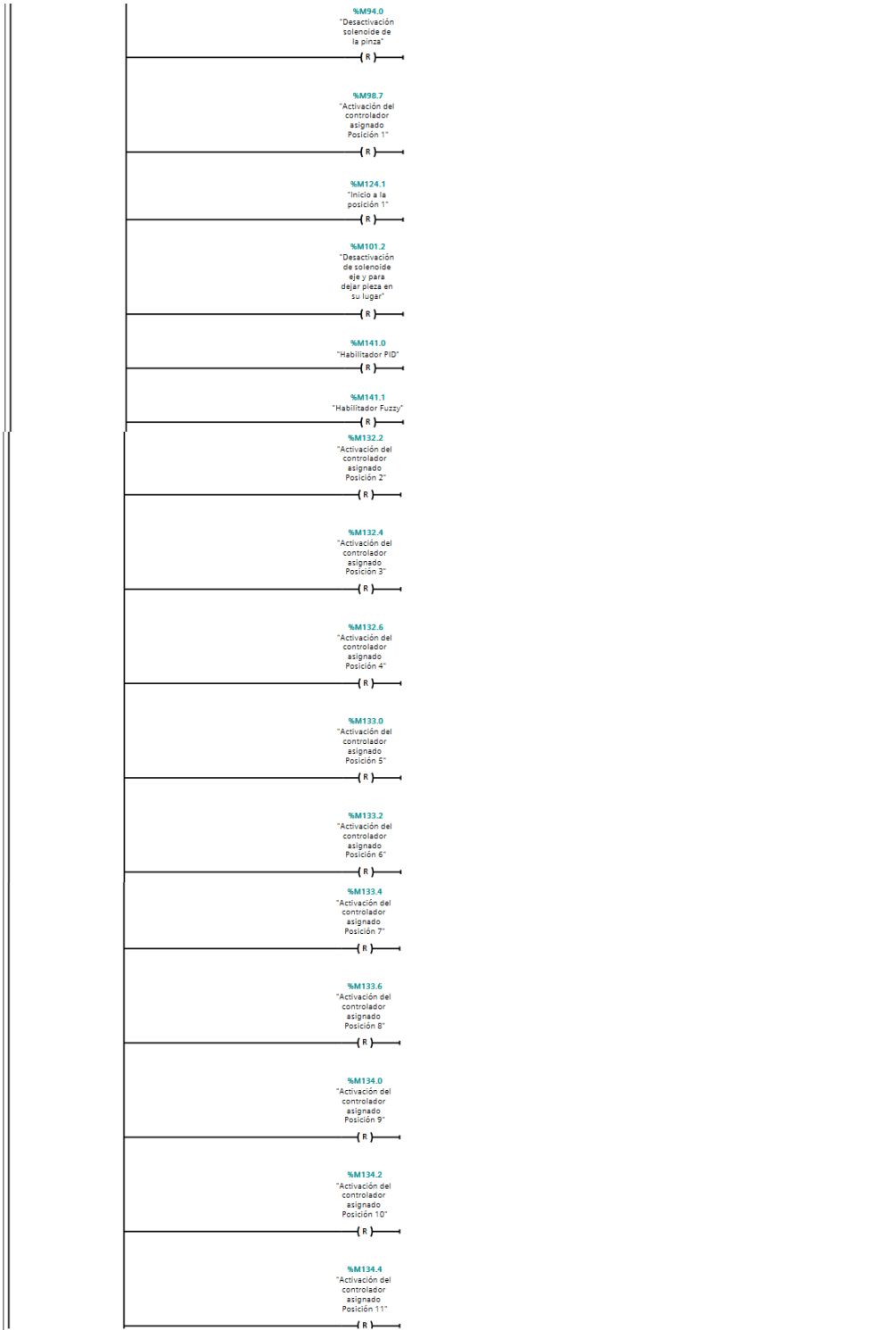
Tiempo de espera para resetear bobina del solenoide del eje y.

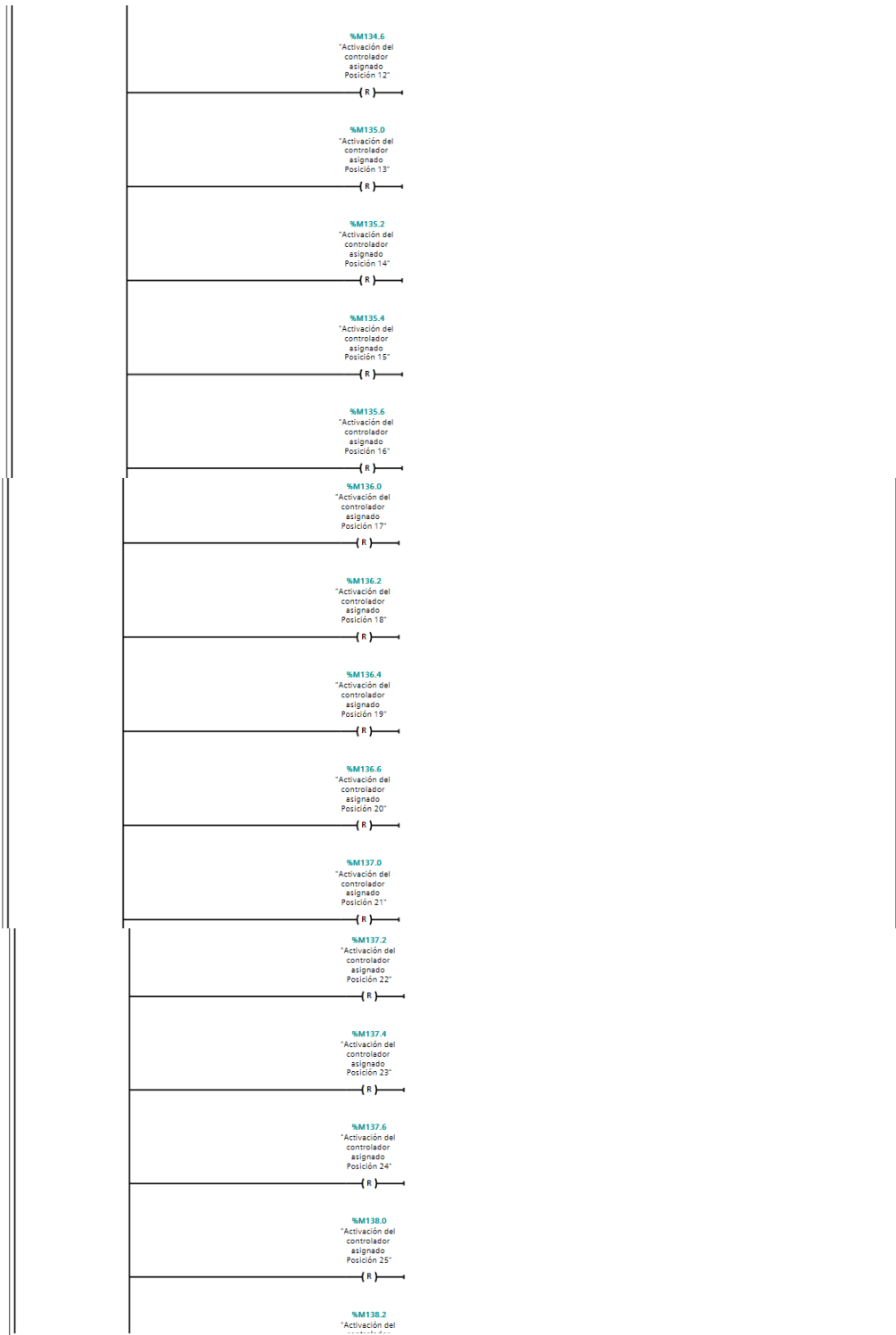


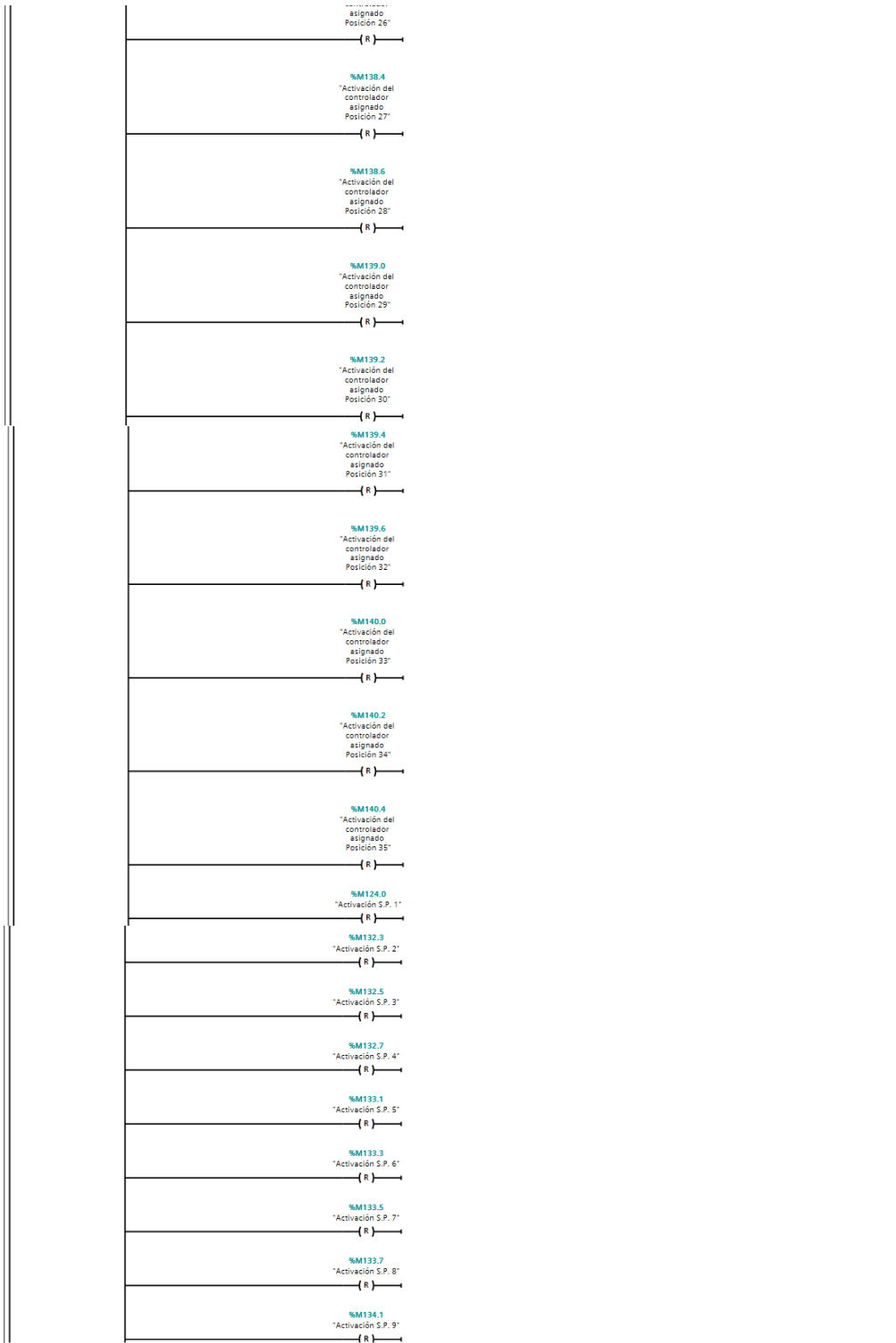
Segmento 47: Reseteo de bobinas para nuevo proceso.

Reseteo de bobina para que pueda ser utilizada nuevamente sin pulsar el botón de paro.









	%M134.3 "Activación S.P. 10"	(R)
	%M134.5 "Activación S.P. 11"	(R)
	%M134.7 "Activación S.P. 12"	(R)
	%M135.1 "Activación S.P. 13"	(R)
	%M135.3 "Activación S.P. 14"	(R)
	%M135.5 "Activación S.P. 15"	(R)
	%M135.7 "Activación S.P. 16"	(R)
	%M136.1 "Activación S.P. 17"	(R)
	%M136.3 "Activación S.P. 18"	(R)
	%M136.5 "Activación S.P. 19"	(R)
	%M136.7 "Activación S.P. 20"	(R)
	%M137.1 "Activación S.P. 21"	(R)
	%M137.3 "Activación S.P. 22"	(R)
	%M137.5 "Activación S.P. 23"	(R)
	%M137.7 "Activación S.P. 24"	(R)
	%M138.1 "Activación S.P. 25"	(R)
	%M138.3 "Activación S.P. 26"	(R)
	%M138.5 "Activación S.P. 27"	(R)
	%M138.7 "Activación S.P. 28"	(R)
	%M139.1 "Activación S.P. 29"	(R)

