

**ESCUELA TÉCNICA SUPERIOR DE INGENIEROS
INDUSTRIALES Y DE TELECOMUNICACIÓN**

UNIVERSIDAD DE CANTABRIA



Proyecto Fin de Grado

**CONTROL POR PLC Y SCADA PARA UN
MOTOR
DE CORRIENTE CONTINUA MEDIANTE
REGULADOR PID
PLC AND SCADA CONTROL FOR A MOTOR
DIRECT CURRENT BY PID REGULATOR**

Para acceder al Título de

**GRADUADO EN INGENIERÍA ELECTRÓNICA
INDUSTRIAL Y AUTOMÁTICA**

Autor: Marcos González Tresgallo

Julio - 2022

RESUMEN

El objetivo del presente trabajo es el de realizar el diseño y montaje de un sistema de control de velocidad de un motor de corriente continua a través de un autómata programable o Controlador Lógico Programable (a partir de ahora PLC) con un controlador S7-1200 CPU 1212C AC/DC/Rly y dos módulos añadidos, el módulo de salidas digitales SB 1222 DC 200 KHz DQ 4x5VDC 0.1A, el cual presenta 4 salida digitales de 5 V de continua y de 0.1 A, y el módulo de entradas y salidas analógicas SM 1234 AI/AQ, el cual tiene 4 entradas analógicas y 2 salidas analógicas. Con este PLC y sus módulos añadido se utilizará un regulador proporcional, integral y derivativo (a partir de ahora PID) creado en el programa TIA PORTAL para realizar el control de velocidad.

Se realizará un SCADA con WINCC, con el cual se podrá manipular el control de velocidad a través de una Interfaz Humano Máquina (a partir de ahora HMI) que es una pantalla interactiva por simulación, pudiendo modificar los valores de los parámetros del regulador PID y observar la evolución de la respuesta del sistema ante la entrada introducida. Además, también se implementará un registro de usuario para que solamente puedan acceder al control el personal autorizado y registrado.

También se ha creado una página web, la cual está vinculada con el PLC y con la cual se podrá realizar un mismo control de velocidad del motor al igual que se hacía en el SCADA con WINCC.

La finalidad del proyecto será para uso académico para aquellos alumnos de la universidad de Cantabria que tengan una asignatura de automática y control, para poner en práctica lo que se ha impartido en las clases de teoría.

ABSTRACT

The goal of this work is to design and assemble a speed control system of a direct current motor through a programmable automaton or Programmable Logic Controller (from now on PLC) with an S7-1200 CPU 1212C AC/DC/Rly controller and two added modules, the digital output module SB 1222 DC 200 KHz DQ 4x5VDC 0.1A, which has 4 digital outputs of 5 V continuous and 0.1 A, and the module of analog inputs and outputs SM 1234 AI / AQ, which has 4 analog inputs and 2 analog outputs. With this PLC and its modules added, a proportional, integral, and derivative regulator (from now on PID) created in the TIA PORTAL program will be used to perform the speed control.

A SCADA with WINCC will be conducted, with which the speed control can be manipulated through a Human Machine Interface (from now on HMI) that is an interactive screen by simulation, being able to change the values of the parameters of the PID regulator and see the evolution of the response of the system to the input entered. In addition, a user registry will also be implemented so that only authorized and registered personnel can access the control.

A web page has also been created, which is linked to the PLC and with which you can perform the same motor speed control as was done in the SCADA with WINCC.

The purpose of the project will be for academic use for those students at the University of Cantabria who have a subject of automatic and control, to put into practice what has been taught in the theory classes.

ÍNDICE

1. INTRODUCCION.	9
1.1 ANTECEDENTES.	9
1.1.1 Motor de Corriente Continua.	9
1.1.2 Sistemas de control de velocidad.	13
1.1.3 Controlador Lógico Programable.	14
1.2 OBJETIVO.	15
1.3 ESTRUCTURA DEL DOCUMENTO.	16
2. DESCRIPCIÓN DE LOS EQUIPOS FÍSICOS.	17
2.1 MOTOR DC UTILIZADO.	17
2.1.1 Dinamo Tacométrica.	18
2.1.2 Encoder.	20
2.2 CIRCUITO SENSOR OPTOACOPLADOR TCST2103.	21
2.2.1 Sensor TCST2103.	21
2.2.2 Obtención de los valores de resistencias.	23
2.2.3 Resultados y foto del circuito.	26
2.3 CIRCUITO PUENTE EN H.	28
2.3.1 Funcionamiento.	29
2.3.2 Resultado y conexión de circuito.	30
2.4 CIRCUITO DIVISOR DE TENSIÓN.	31
2.4.1 Diseño del filtro paso bajo.	31
2.4.2 Diseño del divisor de tensión.	32
2.4.3 Conexión del circuito físico.	33
3. DESCRIPCIÓN TIA PORTAL.	35
3.1 TIA PORTAL.	35
3.2 CREACIÓN DE PROYECTO.	35
3.3 BLOQUES DE PROGRAMA.	40
3.3.1 Main.	41
3.4 CYCLIC INTERRUPT.	65
3.4.1 Partes de un Regulador PID.	65

3.4.2 PID_Compact.	67
3.5 PANTALLA HMI.....	72
3.5.1 Plantilla.	73
3.5.2 Inicio de Sesión.	74
3.5.3 Menú.....	76
3.5.4 Control de velocidad.....	78
3.5.5 Control del error.....	84
3.5.6 Información.....	86
3.5.7 Avisos.	86
3.5.8 Usuarios.	89
3.6 EJEMPLOS DEL FUNCIONAMIENTO DEL HMI.	89
4. DESCRIPCIÓN PÁGINA WEB.....	93
4.1 CÓDIGO ARCHIVO CSS.	93
4.2 CÓDIGO ARCHIVO HTML.	95
4.2.1 Arranque y paro.....	98
4.2.2 Velocidad de referencia.....	99
4.2.3 Tipo de control.....	99
4.2.4 Sentido de giro.	100
4.2.5 Parámetros del PID.....	101
4.2.6 Resultado página web.....	102
4.3 ARCHIVO HTM SIGNAL.	104
4.3.1 Resultados página web.....	105
4.4 ARCHIVO HTM OUTPUT.....	106
4.4.1 Resultados página web.....	107
4.5 EJEMPLOS DEL FUNCIONAMIENTO DE LA PÁGINA WEB.....	107
5. CONCLUSIÓN.	110
6. PRESUPUESTO DE MATERIAL.	111
7. BIBLIOGRAFÍA.	113
8. ANEXO.....	115
8.1 CÓDIGO DE TIA PORTAL.	115

8.2 TABLA DE VARIABLES DEL TIA PORTAL.....	140
8.3 CÓDIGO HTML DE LA PÁGINA WEB.....	142
8.4 CÓDIGO HTM DE LA PÁGINA WEB.....	145
8.5 CÓDIGO CSS DE LA PÁGINA WEB.	147

INDICE DE FIGURAS.

Figura 1. Estator de un motor.	9
Figura 2. Rotor de un motor.	9
Figura 3. Escobillas de un motor.	10
Figura 4. Colector de un motor.	10
Figura 5. Carcasa de un motor.	10
Figura 6. Esquema de un motor de dos polos.	12
Figura 7. Gráfica comparativa del par de los motores AC y DC.	13
Figura 8. Señal PWM para distintos ciclos de trabajo.	14
Figura 9. Motor DC utilizado en el proyecto.	18
Figura 10. Esquema de la dinamo tacométrica del motor DC.	19
Figura 11. Encoder del motor DC.	20
Figura 12. Circuito esquemático del sensor TCST2103.	22
Figura 13. Posibles estructuras del circuito.	22
Figura 14. Estructura del circuito elegida.	23
Figura 15. Gráfica correspondiente a la figura 7 de la hoja de características para obtener If.	24
Figura 16. Gráfica correspondiente a la figura 4 de la hoja de características para obtener Vf.	24
Figura 17. Circuito final para la lectura de la velocidad por el Encoder.	26
Figura 18. Señal del circuito Sensor optoacoplador TCST2103 cuando no detecta.	26
Figura 19. Señal del circuito Sensor optoacoplador TCST2103 cuando detecta.	27
Figura 20. Circuito montado físicamente.	27
Figura 21. Conexión entre el sensor TCST2103 y los demás componentes del circuito.	28
Figura 22. Circuito Puente H señalando sus componentes.	28
Figura 23. Señales y salidas utilizadas del Puente H.	29
Figura 24. Señal PWM de la salida del PLC de 5 V de amplitud.	30
Figura 25. Puente H alimentado y funcionando.	30
Figura 26. Señal PWM de la salida del puente H.	31
Figura 27. Esquemático y gráfica de un filtro paso bajo.	31
Figura 28. Esquemático del circuito Divisor de Tensión.	32
Figura 29. Circuito esquemático del divisor de tensión con el filtro paso bajo.	33
Figura 30. Circuito del divisor de tensión y filtro paso bajo físico.	33
Figura 31. Iniciar proyecto nuevo.	35
Figura 32. Crear el proyecto.	36
Figura 33. Controlador SIMATIC S7-1200 CPU 1212C.	36
Figura 34. Módulo añadido de 4 salidas digitales.	37
Figura 35. Módulo añadido de 4 entradas y 2 salidas analógicas.	37
Figura 36. Conjunto de controlador y módulos añadidos.	37
Figura 37. Agregar dispositivo.	38
Figura 38. Seleccionar el dispositivo de controlador.	38
Figura 39. Añadir módulo de 4 salidas digitales.	39
Figura 40. Añadir módulo de 4 entradas y 2 salidas analógicas.	40
Figura 41. Comparación entre lo físico y virtual.	40
Figura 42. Segmento 1 del Main. Arranque y Paro.	41
Figura 43. Segmento 2 del Main. SetPoint.	42
Figura 44. Interfaz del FB SetPoint.	43
Figura 45. Segmento 1 del FB SetPoint.	44
Figura 46. Segmento 2 del FB SetPoint.	44
Figura 47. Segmento 3 del Main. Generador PWM.	45
Figura 48. Activar Generadores de impulsos.	46
Figura 49. Introducir salida de la señal PWM.	46
Figura 50. Habilitar la modificación del tiempo de ciclo.	46
Figura 51. Direcciones de entrada/salida del generador activado.	47
Figura 52. ID de hardware del generador activado.	47
Figura 53. Interfaz del FB Generador PWM.	47
Figura 54. Segmento 1 del FB Generador PWM.	48
Figura 55. Segmento 2 del FB Generador PWM.	48
Figura 56. Segmento 3 del FB Generador PWM.	49
Figura 57. Instrucciones para desactivar el sentido de giro cuando se cambia.	49

Figura 58. Segmento 4 del Main. Velocidad por el Tacogenerador.	50
Figura 59. Interfaz del FB Velocidad por el Tacogenerador.	50
Figura 60. Segmento 1 del FB Velocidad por el Tacogenerador.	51
Figura 61. Segmento 2 del FB Velocidad por el Tacogenerador.	51
Figura 62. Segmento 5 del Main. Velocidad por el Encoder.	52
Figura 63. Activar un contador rápido.	52
Figura 64. Función del contador rápido activado.	53
Figura 65. Direcciones de entrada/salida del contador activado.	53
Figura 66. ID de hardware del contador activado.	53
Figura 67. Filtro en la entrada del contador rápido I0.0.	54
Figura 68. Interfaz del FB Velocidad por el Encoder.	54
Figura 69. Segmento 1 del FB Velocidad por el Encoder.	54
Figura 70. Variables del DB Data_Block.	55
Figura 71. Segmento 2 del FB Velocidad por el Encoder.	55
Figura 72. Segmento 6 del Main. Tensión Vcc y Vout Motor.	56
Figura 73. Interfaz del FB Tensión Vcc y Vout Motor.	56
Figura 74. Segmento 1 del FB Tensión Vcc y Vout Motor.	56
Figura 75. Segmento 2 del FB Tensión Vcc y Vout Motor.	57
Figura 76. Segmento 7 del Main. Tipo de Control PID por Tacogenerador o Encoder.	58
Figura 77. Interfaz del FB Tipo de Control PID por Tacogenerador o Encoder.	58
Figura 78. Segmento 1 del FB Tipo de Control PID por Tacogenerador o Encoder.	59
Figura 79. Segmento 2 del FB Tipo de Control PID por Tacogenerador o Encoder.	59
Figura 80. Segmento 3 del FB Tipo de Control PID por Tacogenerador o Encoder.	59
Figura 81. Segmento 8 del Main. Error de las señales del Tacogenerador y del Encoder.	60
Figura 82. Interfaz del FB Error de las señales del Tacogenerador y del Encoder.	60
Figura 83. Segmento 1 del FB Error de las señales del Tacogenerador y del Encoder.	61
Figura 84. Segmento 2 del FB Error de las señales del Tacogenerador y del Encoder.	61
Figura 85. Segmento 9 del Main. Parámetros del PID.	62
Figura 86. Segmento 10 del Main. Conexión con el servidor web.	62
Figura 87. Activar Página web.	63
Figura 88. Establecer la página web creada.	63
Figura 89. Instrucciones para variables de la página web.	63
Figura 90. Segmento 11 del Main. Avisos para pantalla HMI.	64
Figura 91. Cambios en las propiedades del DB Avisos.	64
Figura 92. Variables del DB Avisos.	64
Figura 93. Segmento 1 del Cyclic Interrupt. PID_Compact.	67
Figura 94. Configuración del PID_Compact.	68
Figura 95. Ajustes básicos del PID_Compact.	68
Figura 96. Ajustes del valor real del PID_Compact.	69
Figura 97. Ajustes avanzados del PID_Compact.	70
Figura 98. Instrucciones para desactivar el PID_Compact.	71
Figura 99. Agregar pantalla HMI.	72
Figura 100. Plantilla HMI.	73
Figura 101. Botón para ir a la imagen Menú del HMI.	73
Figura 102. Imagen Inicio de Sesión del HMI.	74
Figura 103. Agregar usuarios y grupos.	74
Figura 104. Grupo de Administradores y sus permisos.	75
Figura 105. Grupo de Usuarios y sus permisos.	75
Figura 106. Ventana de Inicio de Sesión.	75
Figura 107. Botón para salir del HMI.	76
Figura 108. Imagen Menú del HMI.	76
Figura 109. Nombre del usuario en la imagen Menú.	76
Figura 110. Agregar tarea para visualizar el nombre de usuario.	77
Figura 111. Botón para cerrar sesión.	77
Figura 112. Botones para las distintas imágenes del HMI.	77
Figura 113. Imagen Control de Velocidad del HMI.	78
Figura 114. Cuadro de escritura para velocidad de referencia y botones para el sentido de giro.	79
Figura 115. Cuadros de lectura de las velocidades y botones del tipo de control.	80
Figura 116. Cuadros de escritura para dar valores a los parámetros PID.	81
Figura 117. Visor gráfico de las señales de velocidad.	82

Figura 118. Botones de arranque y paro del sistema.	82
Figura 119. Cuadros de lectura del estado del motor.....	83
Figura 120. Cuadro de escritura del tiempo de ciclo PWM.....	83
Figura 121. Botón para ir a la imagen Control del Error del HMI.....	84
Figura 122. Imagen Control del Error del HMI.....	84
Figura 123. Visor gráfico de las señales de error.....	85
Figura 124. Cuadros de lectura de los errores.....	85
Figura 125. Botón para ir a la imagen Control de Velocidad del HMI.....	86
Figura 126. Imagen Información del HMI.....	86
Figura 127. Ventana de avisos warnings del HMI.....	87
Figura 128. Propiedades de la variable Variable_HMI_Avisos.....	87
Figura 129. Estructura de una variable Word según la CPU.	87
Figura 130. Estructura de una variable Word según las alarmas del HMI.....	88
Figura 131. Categoría de los avisos.	88
Figura 132. Cambio de los avisos en la ventana de avisos pendientes.	88
Figura 133. Imagen Usuarios del HMI.	89
Figura 134. Dar permisos solo al Administrador.....	89
Figura 135. Respuesta a 250 rpm, $T_i=1$, $T_d=0.1$, $K=1$ y control Tacogenerador.	90
Figura 136. Respuesta a 500 rpm, $T_i=5$, $T_d=1$, $K=2$ y control Encoder.....	90
Figura 137. Respuesta a 500 rpm, $T_i=3$, $T_d=0.1$, $K=2$ y control Tacogenerador.	91
Figura 138. Respuesta a 750 rpm, $T_i=0.1$, $T_d=1.5$, $K=2.5$ y control Encoder.....	91
Figura 139. Dar forma al elemento tabla.	94
Figura 140. Modificar las columnas de las tablas.....	94
Figura 141. Modificar las filas pares de la tabla.....	94
Figura 142. Dar forma y modificar el elemento botón.	94
Figura 143. Conectar variables del PLC al archivo HTML.....	96
Figura 144. Cabeza del archivo HTML.....	96
Figura 145. Código JavaScript en el archivo HTML.....	97
Figura 146. Primeras líneas de código del cuerpo HTML.....	97
Figura 147. Fila de Arranque y Paro de la tabla.	98
Figura 148. Fila de Velocidad de referencia de la tabla.....	99
Figura 149. Fila de Tipo de Control de la tabla.	99
Figura 150. Fila de Sentido de giro de la tabla.	100
Figura 151. Filas de los Parámetros del PID de la tabla.	101
Figura 152. Tabla final de introducción de datos.	102
Figura 153. Ejemplo de las animaciones de los botones y cajas.....	102
Figura 154. Prueba de los límites de las cajas.	103
Figura 155. Llamar a los archivos HTM.....	103
Figura 156. Código del archivo signal HTM.....	104
Figura 157. Código para las imágenes.....	105
Figura 158. Tabla del archivo HTM signal en la página web.....	105
Figura 159. Código del archivo output HTM.....	106
Figura 160. Tabla del archivo HTM output en la página web.	107
Figura 161. Ejemplo N.º 1 del funcionamiento de la página web.....	108
Figura 162. Ejemplo N.º 2 del funcionamiento de la página web.....	109

ÍNDICE DE TABLAS

Tabla 1. Datos de la dinamo tacométrica.....	19
Tabla 2. Valores de los parámetros del sensor TCST2103.....	21
Tabla 3. Presupuesto de material.	112

1. INTRODUCCION.

1.1 ANTECEDENTES.

1.1.1 Motor de Corriente Continua.

Para entender cómo funciona un motor primero es necesario conocer las partes que constituye un motor de corriente continua [1]:

- ESTATOR: Es la parte fija que crea el campo magnético de excitación al ser alimentado por corriente continua. Está formado por unas bobinas de hilos de cobre que forman el devanado inductor y un núcleo de hierro por donde se alojan las bobinas por las ranuras de este.

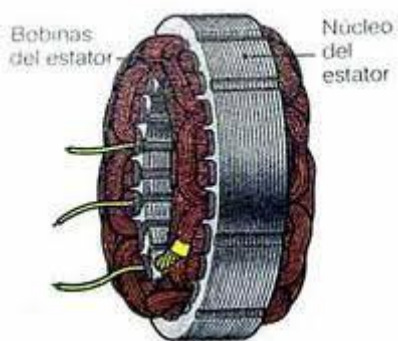


Figura 1. Estator de un motor.

Fuente: [1]

- ROTOR: Es la parte móvil de la máquina. El campo creado por sus bobinas, junto al del estator, crean el par de fuerzas para hacer girar al motor.



Figura 2. Rotor de un motor.

Fuente: [1]

- ESCOBILLAS: son dos tacos de grafito que están en contacto con el colector y suministran la corriente eléctrica a las bobinas del rotor.



Figura 3. Escobillas de un motor.

Fuente: [1]

- COLECTOR: es una especie de anillo de cobre que pone en contacto las escobillas con las bobinas del rotor.

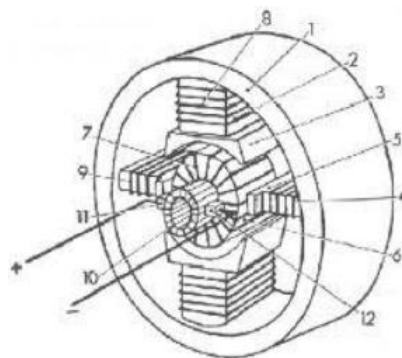


Figura 4. Colector de un motor.

Fuente: [1]

- ENTREHIERRO: Espacio de aire que separa el estator con el rotor. Por él pasa el flujo magnético de uno al otro. Además, permite el movimiento del motor y debe ser lo más reducido posible.

- CARCASA METÁLICA: Reside en su interior dos imanes permanentes con sus relativos polos norte y sur.



Figura 5. Carcasa de un motor.

Fuente: [1]

Los motores de corriente continua [2], o también conocidos por los nombres de motor de corriente directa, motor DC, que significa Direct Current en inglés, o motor CC, que en español sus siglas significan Corriente Continua, (de ahora en adelante se nombrará motor DC) generan un campo magnético que ejerce sobre el rotor convirtiendo la energía eléctrica que se le suministra en energía mecánica. Debido a que su velocidad se puede variar, los motores DC se pueden utilizar para numerosas aplicaciones industriales ya que son una de las mejores opciones en aplicaciones de control y automatización de procesos.

La principal característica es su regulación de velocidad, pudiéndose hacer tanto en el vacío como a plena carga. Y su principal desventaja es el mantenimiento, que es muy caro y laborioso.

Para conseguir el máximo rendimiento de un motor DC se necesita que la potencia de entrada sea igual a la potencia de salida, en vatios la potencia y en tanto por ciento el rendimiento. Sin embargo, muchas de las piezas y partes del motor DC influyen en la pérdida de rendimiento del motor, que se disipa por pérdidas de calor. Con el fin de conseguir el rendimiento óptimo, ya que el rendimiento del 100% es ideal, se necesitará alimentar al motor DC con una alimentación estable, además de tener un buen cableado desde la alimentación hasta el motor por ser de igual importancia, ya que la más mínima pérdida afectará en el rendimiento.

Su funcionamiento sigue la expresión de la ley de Lorentz, la fuerza que actúa en un alambre o cable eléctrico es igual a la intensidad que circula en el seno de un campo magnético.

$$\vec{F} = l \times \vec{I} \times \vec{B}$$

En donde:

- $\vec{B}$: inducción de campo magnético (en Teslas).
- l : la longitud del cable (en metros).
- $\vec{I}$: intensidad que recorre al conductor (en Amperios).
- $\vec{F}$: fuerza que se produce sobre el cable (en Newton).

Existen varios tipos de motores DC dependiendo de cómo estén conectadas las bobinas:

1. De conexión en serie: el devanado del rotor y el del estator están en serie.
2. Tipo shunt: cuyo bobinado inductor principal está conectado en paralelo con el circuito formado por los bobinados del inducido.
3. Compuesta: es combinación de las dos primeras.

Estos tres casos mencionados son con el estátor bobinado, pero también se pueden encontrar con que no lleven bobinado y en su lugar lleven incorporados imanes permanentes de ferrita o de tierras raras.

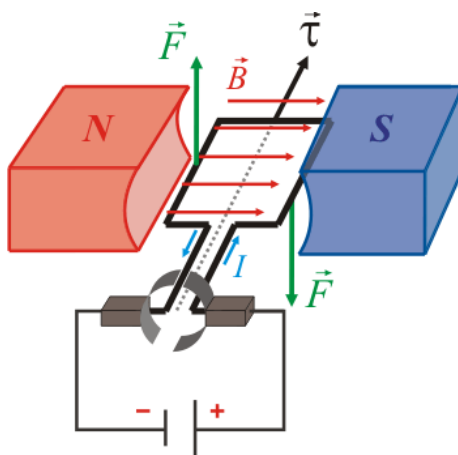


Figura 6. Esquema de un motor de dos polos.

Fuente: [2]

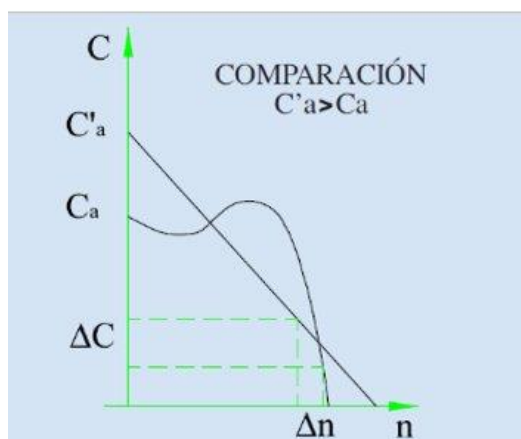
El sentido de giro del motor DC es sencillo de cambiar, ya que depende del campo magnético generado y éste de la corriente que lo genera. Es suficiente con cambiar la polaridad de la alimentación, el polo positivo se coloca en el negativo y viceversa, para cambiar el sentido de giro del motor DC.

La regulación de la velocidad se puede conseguir aplicando uno de los siguientes métodos;

1. Regulación de tensión por una resistencia variable o potenciómetro.
2. Regulación por modulación por ancho de pulso o en inglés Pulse Width Modulation (desde ahora PWM).

Este último método es la más usada por su eficiencia. En el caso de la resistencia variable, es un método que supone una pérdida de energía, en forma de disipación de calor, por la misma resistencia al variar la tensión.

Un dato importante que hay que tener en cuenta a la hora de la regulación de velocidad es la carga del motor, cuanto mayor carga se le proporcione menor será la velocidad de giro si la alimentación se mantiene constante, y el caso contrario, a menor carga mayor velocidad.



- $C'a$: par en el motor de continua.

- C_a : par del motor de alterna.

- N : revoluciones por minuto (a partir de ahora r.p.m.)

Fuente: [2]

Figura 7. Gráfica comparativa del par de los motores AC y DC.

Las ventajas que presentan estos motores son las siguientes:

- Al presentar una relación lineal entre el par de giro y velocidad, el par de arranque en motores DC es más alto que en motores de corriente alterna (AC).
- Tiene una regulación de la velocidad sencilla y económica.
- Es una máquina muy versátil, se puede aplicar para numerosas aplicación ya que se le puede añadir reductoras para conseguir más fuerza de giro.
- Se trabaja con tensiones que no suponen un peligro para el ser humano (normalmente de 24/12 V).

Y como toda máquina tiene unos inconvenientes:

- La construcción y mantenimiento es costoso.
- Limitada vida útil debido al desgaste por fricción de partes que lo componen como es el caso de las escobillas.
- No son tan buenos para producir energía en largas distancias.

1.1.2 Sistemas de control de velocidad.

Ya se ha comentado antes acerca de la regulación de velocidad de estos motores, que recordando había dos métodos principales, la regulación de tensión mediante una resistencia variable o potenciómetro y la regulación por PWM.

Gracias a la electrónica de hoy en día se puede realizar el control de la velocidad de forma muy sencilla. Para alcanzar la velocidad a la que se desea que el motor gire el valor de la señal se compara una y otra vez, de forma continua, con la velocidad real de giro. [3]

- Regulación de tensión por resistencia variable o potenciómetro:

Un potenciómetro no es más que una simple resistencia a la cual se la puede hacer variar su valor de impedancia girando un eje. Está formado con tres terminales y el eje para

modificar el valor. Usando una de las terminales laterales y el del medio se crea una resistencia variable para controlar o ajustar la tensión.

El problema que se da con este método es que está limitada la corriente y disipa la potencia, por lo tanto tenemos una alteración de las características de par de la máquina.

- Regulación por PWM:

Es el método más extendido para regular la velocidad de los motores DC por su fiabilidad y sencillez. Consiste en trabajar con una señal digital cuadrada de un valor de amplitud, en la cual podemos variar el ciclo de trabajo sin variar la frecuencia.

El ciclo de trabajo indica el tiempo, como porcentaje del tiempo total que tarda en completar un ciclo entero, en el que la señal PWM está en alto, es decir que aporta energía. Un ejemplo sería; si se tiene una señal PWM con una amplitud de 12 V y un ciclo de trabajo del 60%, la alimentación que le llegará a la maquina sería de 7.2 V ($12 \text{ (V)} * 0.6$) ya que se comporta como una señal continua.

La frecuencia o tiempo total determina con qué rapidez se completa un ciclo, y por ende, la velocidad entre los estados lógicos alto y bajo.

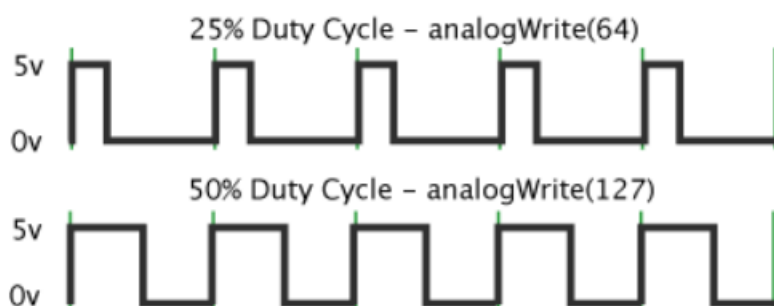


Figura 8. Señal PWM para distintos ciclos de trabajo.

Fuente: [3]

De los dos métodos mencionados, el usado en este proyecto es la regulación por medio de una señal PWM debido a su sencillez y porque el método del uso de un potenciómetro o resistencia variable presenta el inconveniente de la disipación de potencia.

1.1.3 Controlador Lógico Programable.

Con el avance de las nuevas tecnologías se reemplazaron los sistemas de lógica combinacional, con relés e interruptores. La lógica combinacional son sistemas en los que su salida es una función de algún valor de sus entradas en cualquier momento. [4]

A finales de los años 60 aparecieron los primeros autómatas programables, o también nombrados como PLCs, con los cuales se buscaba automatizar las industrias para poder

apaciguar las tareas repetitivas. Con esto se consiguió que el operario no realizase la tarea, sino que se encargaba de vigilar de que fuese correcta la ejecución de la máquina.

Cuando aún no se tenía muy clara la idea de que era un PLC, se incorporaron los microprocesadores en su construcción, y con ello se crearon PLCs capaces de comunicarse entre sí y controlar los procesadores de forma telemática. Además, se añadieron mejoras tecnológicas para la obtención de los datos.

No fue hasta los años 80 hasta que se estandarizó y se realizó mejoras en el sistema de programación para que se pudiera realizar con ordenadores particulares.

A partir de los 90 y hasta ahora, las características físicas de los dispositivos y los módulos de programación se modernizaron, pudiendo con ello acceder a las posibilidades de programación, control y los canales de comunicación.

Por tanto, una vez conocido la historia del surgimiento de los PLCs y para que se crearon, se puede definir un PLC como una ordenador industrial que usa la ingeniería para la automatización de procesos y tiene como finalidad que las máquinas desarrollen de forma eficaz las labores que se les han encargado.

Las partes principales de un PLC son: la CPU, un módulo de memorias y de entradas/salidas, la fuente de alimentación y la unidad de programación. De forma general, estos autómatas se clasifican en compactos, modular, banda estrecha, banda baja, montaje en rack, ordenador industrial, software y de ranura.

En el mercado actual los PLCs más vendidos son los de las marcas Siemens y Allen Bradley por su calidad y seguridad. Siemens abarca todo el mercado asiático, además del europeo, y Allen Bradley tiene la supremacía en Estados Unidos.

1.2 OBJETIVO.

El objetivo principal del proyecto es que los futuros alumnos de la universidad de Cantabria puedan estar en contacto con un PLC, viendo como es y cómo funciona, y puedan realizar pruebas de control gracias a la elaboración, diseño y montaje que ha llevado realizar este sistema de control de velocidad de un motor DC a través de un regulador PID que controlará que la velocidad de salida del motor sea la misma que señal de referencia dada.

La regulación se llevará a cabo por PWM, el cual se generará en una de las salidas digitales del PLC S7-1200 CPU 1212C por el regulador PID. Se creará un proyecto en la herramienta de Siemens conocida como TIA PORTAL y se irá agregando bloques de funciones para lograr dicho control de velocidad del motor.

A través de una pantalla HMI, también creada en TIA PORTAL, se podrá ver la evolución a tiempo real de la velocidad del motor alcanzando y estableciéndose a la velocidad de referencia. En esta pantalla se podrá introducir el valor de referencia de velocidad que se desea que el motor tenga, realizar los ajustes de los parámetros K , T_i y T_d del regulado PID, establecer el sentido de giro del motor y la duración del tiempo de la señal PWM.

Otro medio para realizar el control es a través de una página web diseñada y desarrollada para poder realizar este control de velocidad de forma remota. Se podrá realizar los mismos cambios que en la pantalla HMI.

1.3 ESTRUCTURA DEL DOCUMENTO.

El documento está compuesto por siete capítulos en los que se hablará de lo siguiente;

- Capítulo 1 Descripción de los equipos físicos: descripción de los pasos seguidos para el desarrollo e implementación de los circuitos creados con pruebas del correcto funcionamiento.
- Capítulo 2 Descripción TIA Portal: descripción de la programación de código detallada, diseño de pantalla HMI y comunicación con el PLC físico.
- Capítulo 3 Descripción Página web: diseño y funcionamiento de la página web propia del PLC.
- Capítulo 4 Conclusiones: análisis del proyecto general, con los objetivos alcanzados y problemas que han surgido durante su elaboración.
- Capítulo 5 Presupuesto de material: se dará los costes de cada parte del proyecto.
- Capítulo 6 Bibliografía: referencias utilizadas para la elaboración del proyecto.
- Capítulo 7 Anexo: todos los códigos creados para la obtención del proyecto. Se incluye el código de TIA Portal, además de la tabla de variables de este, y el código de la página web del PLC.

2. DESCRIPCIÓN DE LOS EQUIPOS FÍSICOS.

En este capítulo se hablará de todo aquello que no sea relacionado con la programación, en otras palabras, todo aquello que sea un equipo físico. A continuación se dará la lista de los materiales que se han usado:

- Impedancias.
- Cables para conexiones con los circuitos.
- Circuito impreso perforado.
- Soldador de estaño.
- Sensor optoacoplador TCST2103.
- Circuito integrado EL2244CN.
- Soporte para circuito integrado.
- Circuito integrado Puente H L298N Dual H-Bridge Motor Driver.
- Motor DC.
- PLC S7-1200 CPU 1212C AC/DC/Rly y módulos añadidos.
- Alimentaciones (baterías y señales de salida del PLC).
- Osciloscopio.
- Multímetro.

Ahora se dará más detalle de los circuitos que se han diseñado y usado, además del motor DC que se emplea y del PLC S7-1200 CPU 1212C AC/DC/Rly con sus módulos añadidos.

2.1 MOTOR DC UTILIZADO.

El motor que se utiliza es un motor DC de imán permanente del que se incluye la información extraída del manual de los transductores de velocidad y aceleración y acondicionador de señal Mod. G28/EV [5]. Se utiliza la hoja de características de este módulo, que utiliza el mismo motor, ya que no está disponible la documentación original del equipo. Se adjuntan los datos extraídos de la ficha técnica del módulo al que pertenece este motor [6], y la ficha técnica del módulo del que se saca la información en el manual que incluye la hoja de características [7].

Características del motor DC.

- Velocidad de rotación: 4 000 r.p.m.
- Resistencia de armadura R_a : 5.5 Ω
- Inductancia de armadura L_a : 2.8 mH
- Forma constructiva: B14

- Imán permanente: ferrita
- Aislamiento: clase F
- Peso: 5 kg

Estas son algunas de las características que se encontró en la ficha técnica que corresponde este motor [6], pero en el manual del otro modulo [5] se ha encontrado más información al respecto.

La velocidad del motor se obtiene variando desde una tensión de alimentación de 0 hasta 27.5 V, que es la máxima tensión del motor DC, consiguiendo unos valores de velocidad entre 0 y 4 000 r.p.m. máximo.

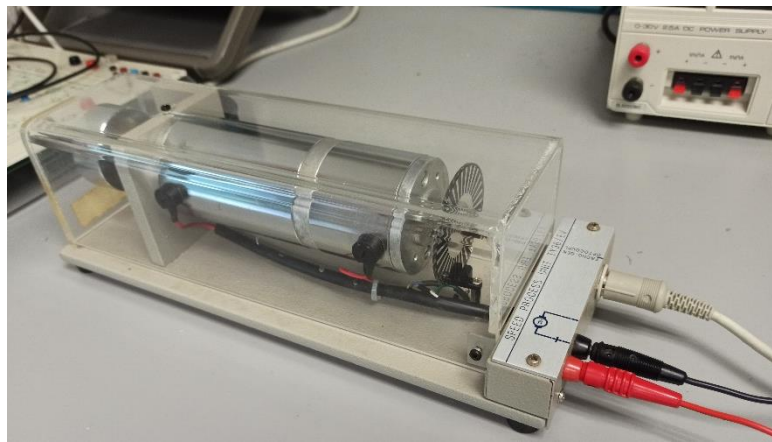


Figura 9. Motor DC utilizado en el proyecto.

2.1.1 Dinamo Tacométrica.

Este motor lleva incorporado una dinamo tacométrica, o también conocido como tacogenerador, el cual se utiliza para la medición de la velocidad angular de los motores DC. La forma en la que se construye una dinamos tacométrica es la siguiente; tiene un estator, en donde se aloja un imán, y un rotor, en el que se enrollan N espiras separadas entre sí por un ángulo $2\pi/N$. La señal de salida se obtiene por dos escobillas que lleva incorporadas.

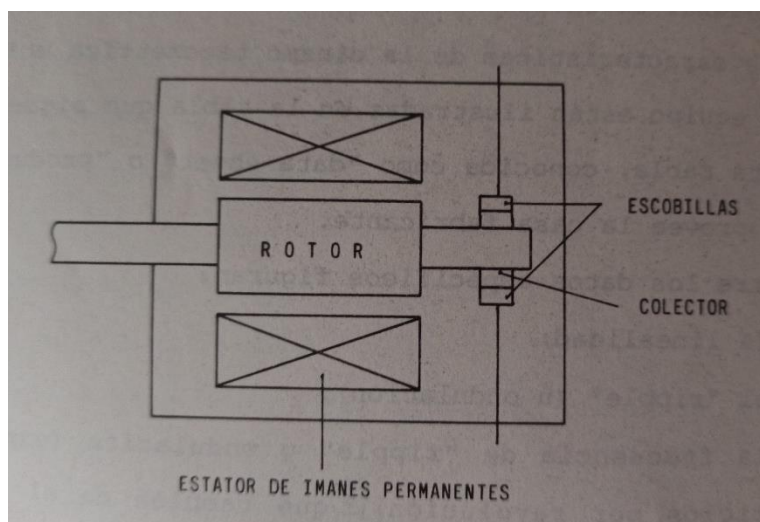


Figura 10. Esquema de la dinamo tacométrica del motor DC.

Fuente: [5]

Las características de este tacogenerador se recogen en la siguiente tabla:

LINEALIDAD (MAX. DESVIACION EN %)	0.2
“RIPPLE” U ONDULACION (MAX. COMPONENTE C.A. CRESTA A CRESTA EN %, A 500 r.p.m.)	2.0
FRECUENCIA DE “RIPPLE” U ONDULACION (CICLOS POR REVOLUCION)	21
COEFICIENTE DE TEMPERATURA (EN %/°C)	-0.01
RESISTENCIA DE AISLAMIENTO (EN MΩ)	10
FRECUENCIA DE RESONANCIA DE TORSION (MIN., EN KHz)	2
DURACION NOMINAL (EN HORAS A 4 000 r.p.m.)	10 000
CAMPO DE TEMPERATURA DE FUNCIONAMIENTO (EN °C)	0 – 155
CONSTANTE TACOMÉTRICA (EN mV/r.p.m.)	14.5
RESISTENCIA DE ARMADURA ($\Omega \pm 20\%$)	110
INDUCTANCIA DE ARMADURA (NOMINAL, EN mH)	8
INDUCTANCIA DE CARGA RECOMENDADA (EN Ω)	10 000

Tabla 1. Datos de la dinamo tacométrica.

Fuente: [5]

La constante tacométrica es la relación entre la tensión por el tacogenerador, en mili voltios, y la velocidad de giro, en r.p.m. . Es una constante muy importante ya que con ella se sacará la velocidad de giro del motor DC por el tacogenerador, que se explicará en el capítulo 2 Descripción TIA Portal la obtención de dicha velocidad de giro del motor DC.

El cálculo de este parámetro se consigue aplicando la fórmula que se muestra a continuación:

$$K_T = \frac{V}{n}$$

en donde:

K_T : constante tacométrica $\left(\frac{V}{r.p.m.}\right)$

V : tensión nominal (V)

n : velocidad nominal en revoluciones por minuto (r.p.m.)

También se puede obtener en radianes por segundo conociendo la relación con las revoluciones por minuto:

$$K_T = \frac{V}{n} \times \frac{60}{2\pi}$$

2.1.2 Encoder.

El encoder, o conocido también por el nombre de codificador, es un instrumento empleado para la obtención de velocidades y posiciones angulares de máquinas. El encoder usado en el motor del proyecto es del tipo fotoeléctrico. Consta de un disco dividido por franjas negras y transparentes, con una resolución de 30, que son el número de franjas negras que posee, y un sensor optoacoplador el cual está conectado a un circuito para poder detectar el giro. En el apartado 2.2 Circuito sensor optoacoplador TCST2103 se dará más información de este sensor optoacoplador.

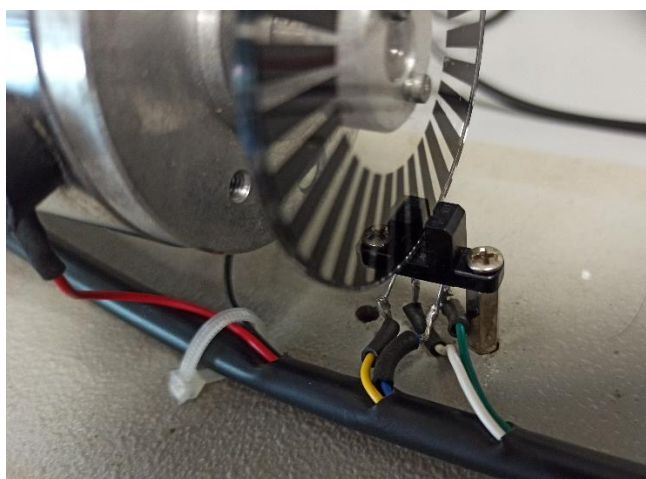


Figura 11. Encoder del motor DC.

El funcionamiento es muy sencillo, el sensor optoacoplador se encargará de realizar el conteo a partir de la rotación del disco, e ira detectando las franjas negras del disco cuando se esté produciendo ese movimiento rotacional.

2.2 CIRCUITO SENSOR OPTOACOPLADOR TCST2103.

Este circuito tiene el propósito de leer la velocidad del motor utilizando para ello el encoder que se comentó en el subapartado 2.1.2 Encoder. A través de lecturas de pulsos el sensor optoacoplador TCST2103 se encargará de detectar el movimiento y velocidad del encoder.

El circuito está compuesto por el sensor TCST2103 ya mencionado, un circuito integrado EL2244CN y unas resistencias R1 de 1 200 Ω y R2 de 6 000 Ω . Esta última se obtiene al colocar dos resistencias de 12 000 Ω en paralelo.

2.2.1 Sensor TCST2103.

Para utilizar el sensor de la mejor manera, primero se tiene que entender que hace y que necesita para funcionar. Viendo la hoja de características de este sensor [8] se ha sacado la información suficiente para entenderle, pero además, se han buscado ejemplos para entender aún más el funcionamiento.

De la información sacada de la hoja de características se puede decir que se trata de un sensor transmisivo, que incluye un emisor de infrarrojos y un fototransistor, situados frente a frente en los ejes ópticos en un paquete de plomo que bloquea la luz visible.

Estos son los valores a temperatura ambiente y un circuito esquemático de cómo se debe montar para hacer la lectura de pulsos:

ABSOLUTE MAXIMUM RATINGS ⁽¹⁾				
PARAMETER	TEST CONDITION	SYMBOL	VALUE	UNIT
INPUT (EMITTER)				
Reverse voltage		V_R	6	V
Forward current		I_F	60	mA
Forward surge current	$t_p \leq 10 \mu s$	I_{FSM}	3	A
Power dissipation	$T_{amb} \leq 25 ^\circ C$	P_V	100	mW
Junction temperature		T_J	100	$^\circ C$
OUTPUT (DETECTOR)				
Collector emitter voltage		V_{CEO}	70	V
Emitter collector voltage		V_{ECO}	7	V
Collector peak current	$t_p/T = 0.5, t_p \leq 10 ms$	I_{CM}	200	mA
Power dissipation	$T_{amb} \leq 25 ^\circ C$	P_V	150	mW
Junction temperature		T_J	100	$^\circ C$

Tabla 2. Valores de los parámetros del sensor TCST2103.

Fuente: [8]

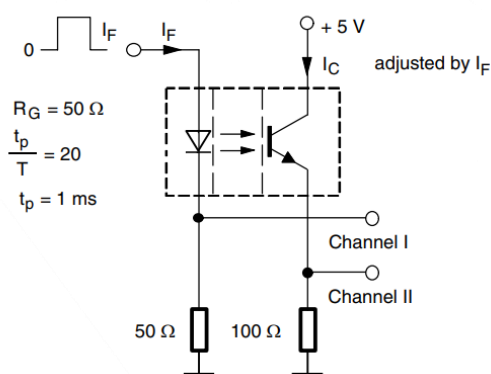


Figura 12. Circuito esquemático del sensor TCST2103.

Fuente: [8]

Sin embargo el circuito que se da en la hoja de características no es el indicado para la aplicación que se le va a dar para este proyecto. Las dos formas posibles que se encontraron para la aplicación del proyecto son las que se muestran en la Figura 13: [9]

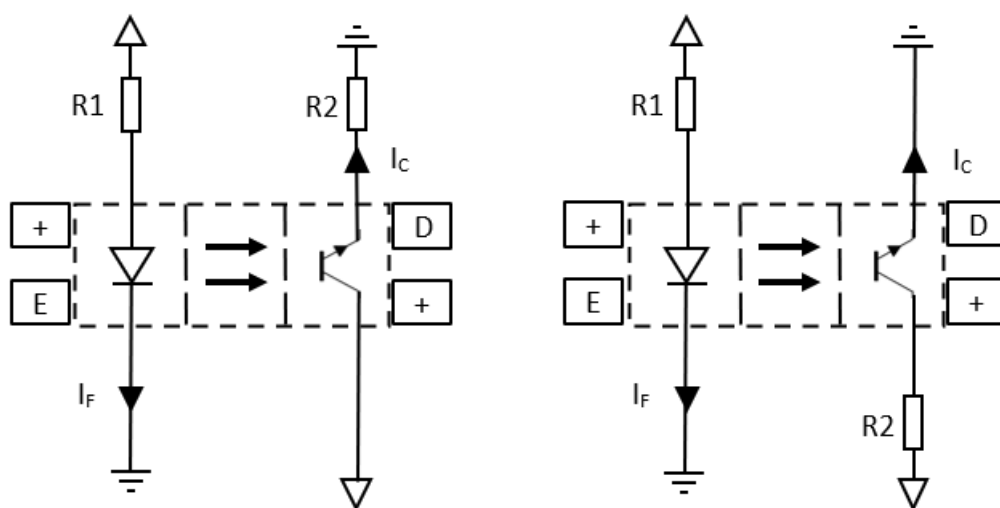


Figura 13. Posibles estructuras del circuito.

Estos dos circuitos se diferencian en que el de la izquierda cuando está en reposo da una señal lógica 1 y el de la derecha una señal lógica 0. Esto se debe por la distribución de las resistencias que se puede apreciar una clara diferencia.

La resistencia R2 del circuito de la derecha se encuentra posicionada entre la alimentación y el terminal positivo de la parte del transistor del sensor, y su terminal D va a tierra. En cambio, en el otro circuito, se encuentra dicha resistencia R2 entre la tierra y el terminal D de la parte del transistor del sensor, y posteriormente el terminal positivo se le alimenta a la tensión de alimentación directamente.

Para el caso de la aplicación del proyecto se ha optado usar el circuito de la derecha, debido a que en reposo, cuando en el encoder no se detecte la franja negra, dé una señal de 0 lógica.

2.2.2 Obtención de los valores de resistencias.

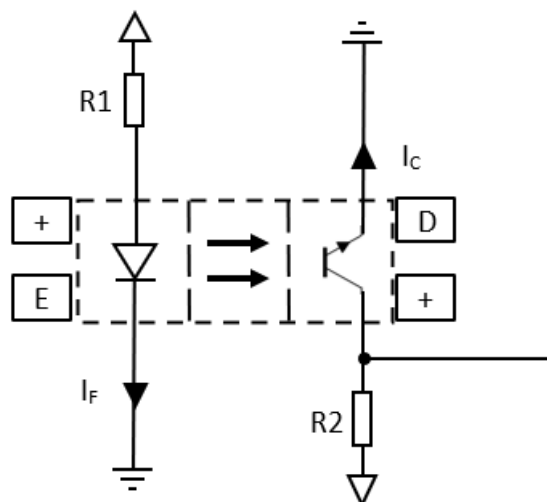


Figura 14. Estructura del circuito elegida.

En el proceso de adquisición de los valores de $R1$ y $R2$ se hicieron varios cálculos debido a que el circuito funciona para una multitud de valores de estas resistencias, ya que dependiendo de la tensión de alimentación que se le suministre y de las corrientes I_C e I_F que circule por el circuito los valores de $R1$ y $R2$ varían.

Sin embargo, para que funcione para la aplicación que se le da, se tiene que cumplir las condiciones de que la tensión de alimentación debe ser de 24 V de continua, porque para que la entrada digital del PLC detecte un 1 lógico debe ser de 24 V de amplitud, y además, que la corriente I_C sea por lo menos de 4 mA, ya que la salida de este circuito a la entrada digital del PLC será entre los terminales positivo de la parte del transistor y el terminal de la resistencia $R2$, como se muestra en la Figura 14.

Con estas dos condiciones se puede sacar los valores de las resistencias. A continuación se mostrará los pasos que se siguieron:

1. Con la condición de que I_C tenga que dar un valor de 4 mA, de la figura 7 de la hoja de características del TCST2103, la cual corresponde a la Figura 15 del documento y representa una gráfica donde se ve la relación entre las corrientes I_C e I_F , se saca el valor correspondiente de la corriente I_F .

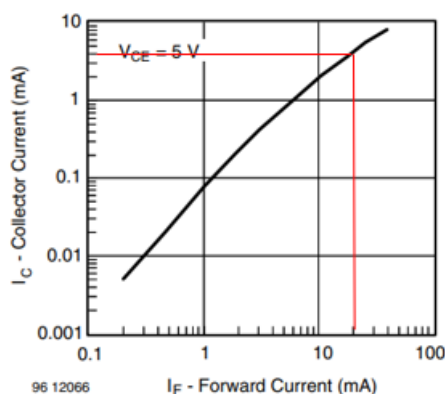


Figura 15. Gráfica correspondiente a la figura 7 de la hoja de características para obtener I_f .

Fuente: [8]

Para una I_C de 4 mA, el valor de I_F es de 20 mA.

- Una vez se tenga el valor de I_F , el siguiente paso de calcular es la tensión directa (V_F), también conocida en inglés como Forward Voltage, que es la tensión del diodo y se saca de la Figura 16, que corresponde a la figura 4 de la hoja de características.

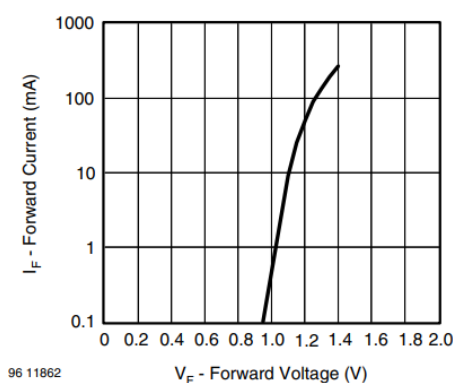


Figura 16. Gráfica correspondiente a la figura 4 de la hoja de características para obtener V_f .

Fuente: [8]

Como la escala de la corriente I_F no es buena para determinar el valor de V_F para un valor de 20 mA de I_F lo que se hizo fue linealizar como si fuera una recta entre los intervalos de 10 a 100 mA y así obtener un valor de referencia para la tensión V_F . Los valores de tensión para estas corrientes, sacadas de la gráfica que se muestra en la Figura 16, son:

$$10 \text{ mA} \rightarrow 1.1 \text{ V}$$

$$100 \text{ mA} \rightarrow 1.28 \text{ V}$$

$$20 \text{ mA} \rightarrow ?$$

$$10 = 1.1 \times m + n \quad (1)$$

$$100 = 1.28 \times m + n \quad (2)$$

$$(1) n = 10 - 1.1 \times m \rightarrow (2)$$

$$(2) 100 = 1.28 \times m + 10 - 1.1 \times m \rightarrow 90 = 0.18 \times m \rightarrow m = \frac{90}{0.18} \rightarrow m = 500$$

$$n = 10 - 1.1 \times 500 \rightarrow n = -540$$

La ecuación de la recta sería $\rightarrow I_F = 500 \times V_F - 540$

$$\text{Para } I_F = 20 \text{ mA} \rightarrow 20 = 500 \times V_F - 540 \rightarrow V_F = \frac{560}{500} \rightarrow V_F = 1.12 \text{ V}$$

3. Con estos valores calculados, el siguiente paso, que es el último, es obtener los valores de las resistencias R1 y R2 por medio de las siguientes expresiones que se sacan del circuito de la Figura 14.

$$R_1 = \frac{V_{cc} - V_F}{I_F} = \frac{24 - 1.12}{20 \times 10^{-3}} \rightarrow R_1 = 1\,144 \, \Omega$$

$$R_2 = \frac{V_{cc} - V_{+D}}{I_c} \rightarrow V_{+D} \text{ tiene un valor de } 0 \text{ V idealmente} \rightarrow \frac{24 - 0}{4 \times 10^{-3}} \rightarrow R_2 = 6\,000 \, \Omega$$

Finalmente se comprobó, que en efecto, el circuito funciona correctamente, donde los valores calculados seas los que se deben dar. Sin embargo, a la hora de conectar el circuito con la entrada digital se vio que no se hacia la lectura correcta por los siguientes motivos:

- Las resistencias calculadas no están normalizadas y los valores que se utilizan en el circuito son: R1 una resistencia de valor 1 200 Ω y R2 dos resistencias en paralelo de 12 000 Ω para obtener el valor de 6 000. Como las resistencias tienen una tolerancia, y no son exactamente las calculadas, las corrientes I_F e I_c no son las deducidas.
- Al conectar la salida del circuito con la entrada la señal digital no se llegaba a los 24 V en alta. Para solucionarlo se tuvo que colocar el circuito integrado EL2244CN, que actuase como seguidor de tensión. Se supuso que es problema del PLC que demandaba más corriente.

Por tanto, el circuito final para la lectura del encoder del motor, y con ello sacar la velocidad de giro con la lectura de la entrada digital del PLC, es el siguiente:

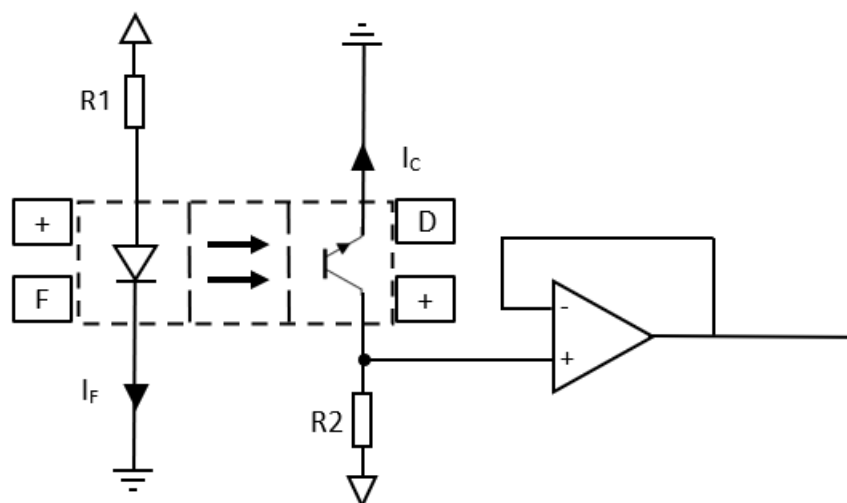


Figura 17. Circuito final para la lectura de la velocidad por el Encoder.

La razón la cual se escogió el circuito integrado EL2244CN fue porque podía ser alimentado a la misma tensión que se alimenta este circuito. En su hoja de características se puede ver cuál es su rango y como está construido. El funcionamiento del circuito integrado EL2244CN con una sola fuente de alimentación trabaja desde 36 V hasta tan sólo 2.5 V. [10]

2.2.3 Resultados y foto del circuito.

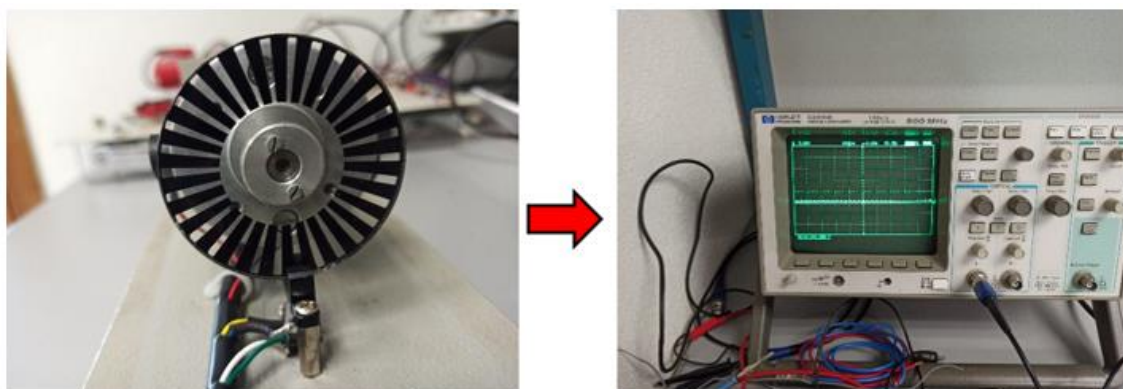


Figura 18. Señal del circuito Sensor optoacoplador TCST2103 cuando no detecta.

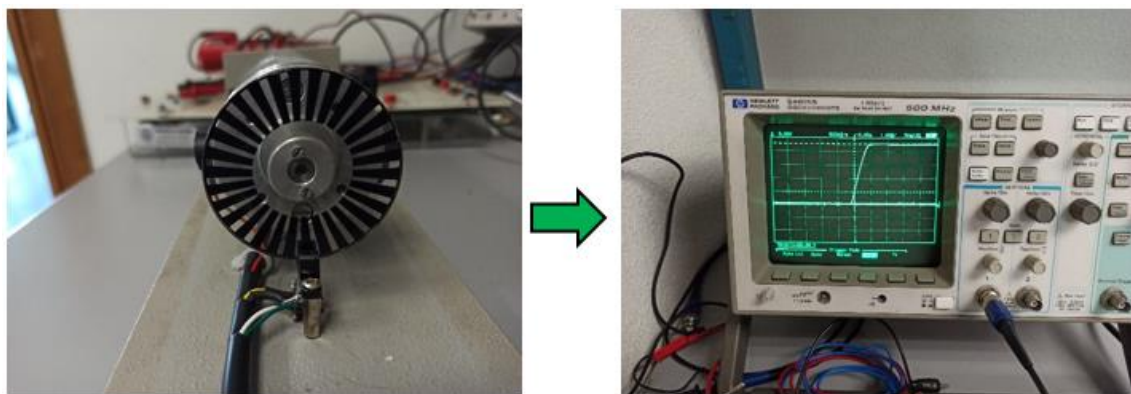


Figura 19. Señal del circuito Sensor optoacoplador TCST2103 cuando detecta.

Como se ve en las Figuras 18 y 19, al detectar un objeto en el espacio que hay entre el diodo y el transistor, el transistor conducirá y dará la señal de 24 V que se ve en el osciloscopio. En las figuras se ven los dos estados;

- El primero es cuando el sensor tiene en el espacio la franja transparente del encoder y por tanto la señal que devuelve es de 0 V.
- El segundo estado es cuando se detecta la franja negra en el sensor, que este dará una señal de 24 V.

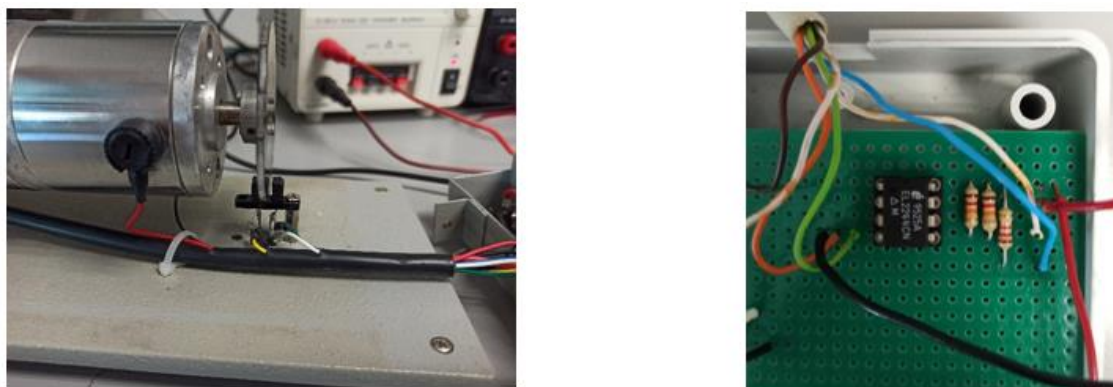


Figura 20. Circuito montado físicamente.

En la imagen de la izquierda de la Figura 20 se muestra dónde irá situado el sensor TCST2103, justo debajo del encoder, sujetado con un tornillo. Esto es así porque se necesita que este sensor esté leyendo el giro del encoder y así detectar el cambio entre franjas negras y transparentes, para generar una señal lógica de 0 y 1 con una amplitud de 24 V.

En la imagen de la derecha de la misma figura se encuentra el circuito montado físicamente en el circuito impreso perforado. Además de este circuito, también se montarán los demás circuitos que se explicarán más adelante. Todo esto irá guardado en una caja para tenerlo todo compacto. Para la unión se utilizó un soldador de estaño.

La conexión entre el circuito soldado en el circuito impreso con el sensor que está debajo del encoder se lleva a cabo con un cable cat 5e UTP 4x2x24, el cual uno de sus extremos se conectará al motor gracias a un conector DIN de 8 pines. Del cable cat 5e UTP 4x2x24 se usarán 6 de sus 8 cables, 4 para este circuito y los otros dos restantes para el circuito del divisor de tensión que se explicará en el subapartado 2.4 Circuito divisor de tensión.



Figura 21. Conexión entre el sensor TCST2103 y los demás componentes del circuito.

2.3 CIRCUITO PUENTE EN H.

Se utiliza un módulo comercial para esta etapa. Para su configuración se ha empleado la documentación encontrada en las referencias [11] y [12]

El circuito permite un control fácil, y de forma independiente, de dos motores de 2A cada uno en ambas direcciones. El uso que se le dará en el proyecto será solamente para el control de un motor, por lo que únicamente se utilizará una de sus dos salidas. La placa tiene incorporado unos indicadores LED de potencia, un regulador de +5 V y diodos de protección.

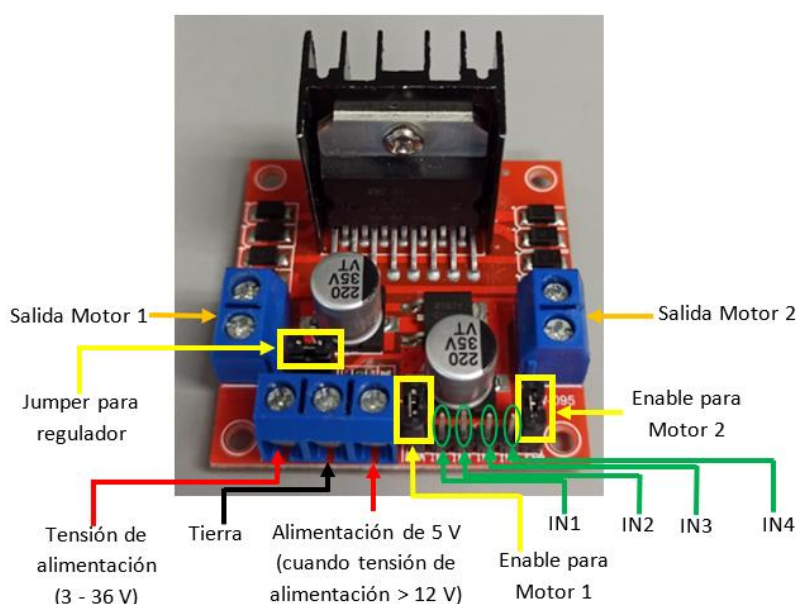


Figura 22. Circuito Puente H señalando sus componentes.

2.3.1 Funcionamiento.

La tensión de alimentación abarca desde los 3 V hasta los 36 V, pero hay unas condiciones que se debe tener en cuenta si se sobrepasa una tensión de 12 V, que no es el caso ya que se le alimentará a 12 V, las cuales tienen que ver con el regulador de 5 V que viene incorporado.

El jumper vinculado al regulador solo funciona hasta una alimentación de 12 V, una vez se sobrepase esta tensión se tendrá que quitar el jumper que esta justo detrás de los terminales de alimentación, y alimentar el terminal que se encuentra más a la derecha en la Figura 22 con una fuente de 5 V.

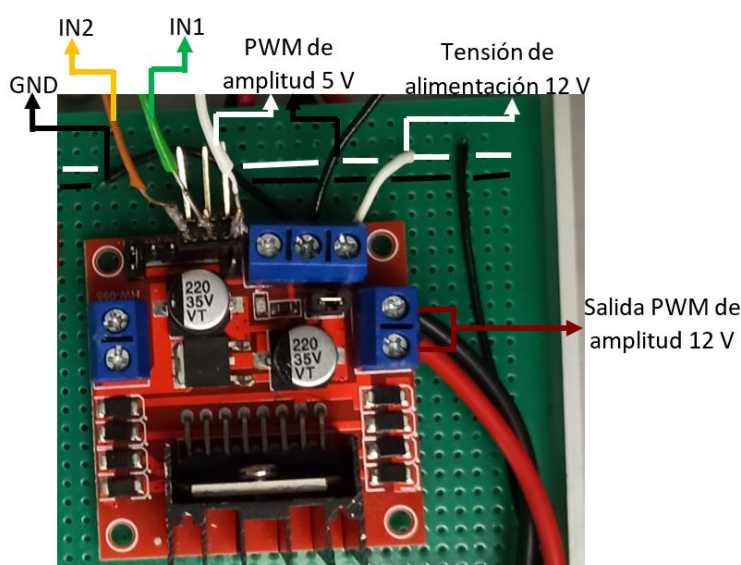


Figura 23. Señales y salidas utilizadas del Puente H.

Se utilizará el motor 1, y por ello se van a necesitar las entradas digitales IN1 e IN2 para indicar el sentido de giro. Si el pin IN1 está en alto, es decir, tiene el valor de 1 y el pin IN2 está bajo, en otras palabras, tiene el valor de 0, el motor girará en un sentido. Si es el caso contrario, el sentido de giro será el opuesto.

También se tiene que quitar el jumper que se localiza en la Figura 22 con el nombre de Enable para motor 1 para poder suministrar la señal PWM que se genera desde el PLC. Si no se quita dicho jumper no se podrá realizar un control de velocidad ya que con el jumper puesto la velocidad de giro del motor es constante.

Tanto el sentido de giro como la señal PWM se generan en el PLC, que posteriormente en el capítulo 2 Descripción TIA Portal se hablará de cómo se generan dichas señales. Hay que indicar que la señal PWM que se genera en el PLC es de amplitud 5 V, porque es lo máximo que puede dar el módulo añadido de salidas digitales SB 1222 DC 200 KHz, pero al introducir dicho PWM en el puente H la amplitud se elevará a 12 V para este caso por darle

una alimentación al circuito de 12 V. Con la señal PWM de 12 de amplitud se alimentará al motor DC, lo que significa que la máxima tensión que le va a suministrar será de 12 V si se trabaja al 100 % del ciclo de trabajo de la señal PWM.

2.3.2 Resultado y conexión de circuito.

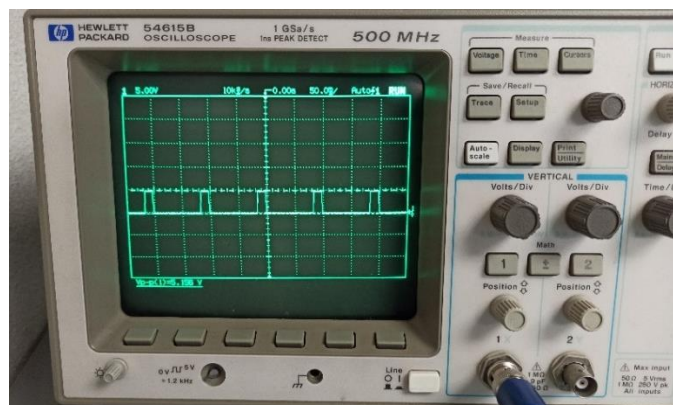


Figura 24. Señal PWM de la salida del PLC de 5 V de amplitud.

La señal que se muestra en el osciloscopio de la Figura 24, con una escala de 5 V por división en el eje Y, hace referencia a la señal PWM de amplitud de 5 V con un ciclo de trabajo del 20% que se genera desde el PLC.

Dicha señal se conecta; el terminal positivo al pin de Enable del motor 1 y el terminal negativo con el GND, como se puede apreciar en la Figura 23.

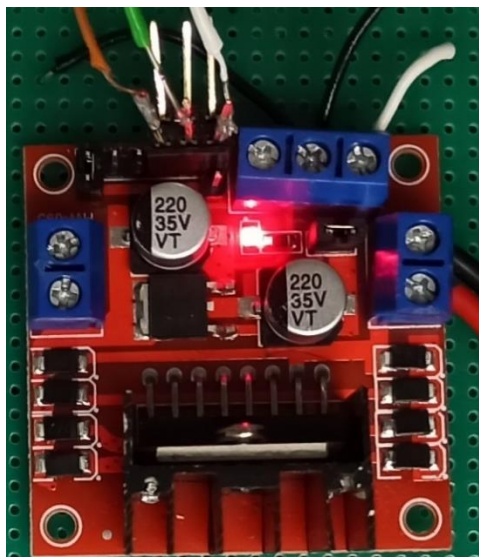


Figura 25. Puente H alimentado y funcionando.

Una vez que se alimenta al circuito del Puente H se encenderá un LED de color rojo indicando que esta alimentado, como se puede ver en la Figura 25. Como se viene

explicando, al alimentar al puente H con una tensión de 12 V, se generará una señal PWM de amplitud de 12 V, la cual se puede observar en la Figura 26.

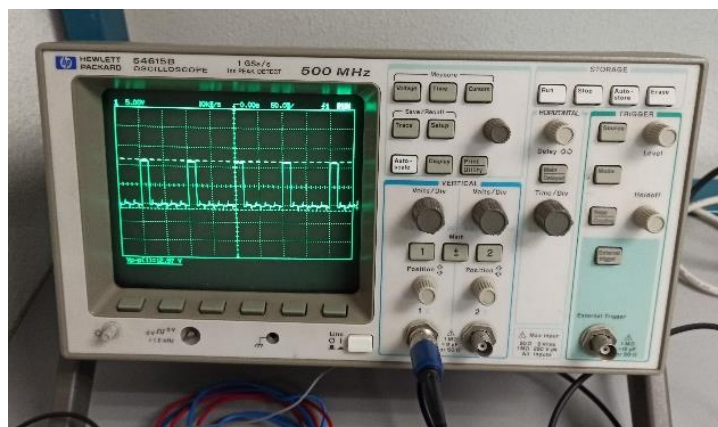


Figura 26. Señal PWM de la salida del puente H.

2.4 CIRCUITO DIVISOR DE TENSIÓN.

Este circuito se diseñó para tener controlada la tensión de salida del tacogenerador del motor DC entre los valores de ± 10 V, ya que sin este circuito la tensión de salida sería el doble de la tensión de alimentación del motor, por cómo está diseñado el tacogenerador. Se necesita este circuito debido a que para poder introducir la señal analógica como una de las entradas del PLC del módulo añadido SM 1234 AI/AQ se necesita que esté entre ± 10 V, que es el rango de tensión que acepta el módulo añadido.

Esta tensión de salida del tacogenerador se utiliza para obtener, por otro método, la velocidad de giro del motor, porque, como se ha explicado en apartado 2.1 Motor DC utilizado, hay una relación entre la tensión de salida y la velocidad de giro del motor en r.p.m. .

El circuito lo compone dos resistencias de valor de $2\,550\ \Omega$ para R1 y de $1\,800\ \Omega$ para R2, que harán de divisor de tensión. Además se le pone un filtro paso bajo después del divisor de tensión debido a que la señal del tacogenerador presenta ruido.

2.4.1 Diseño del filtro paso bajo.



Figura 27. Esquemático y gráfica de un filtro paso bajo.

De la tabla de características mencionada en el subapartado 2.1.1 Dinamo Tacométrica, se extrae la información de que la máxima ondulación se obtiene para la velocidad de 500 r.p.m., con una frecuencia de rizado de 21 ciclos por revolución.

$$f = 21 \left(\frac{\text{ciclos}}{\text{rpm}} \right) \times 500 \text{ (r.p.m.)} \times \frac{1}{60 \text{ (s)}} \rightarrow f = 175 \text{ Hz}$$

Con estos datos se obtiene una frecuencia que se tendrá como referencia para el cálculo de la frecuencia de corte del filtro paso bajo, la cual se obtiene de esta expresión:

$$f_c = \frac{1}{2 \times \pi \times C_f \times R_f}$$

El valor del condensador se ha escogido de los que se tiene disponible, que es de 100 nF

$$R_f = \frac{1}{2 \times \pi \times 100 \times 10^{-9} \times 175} = 9\,094 \, \Omega$$

Como ese valor de resistencia no es un valor normalizado se ha escogido una resistencia de 10 000 Ω . Entonces la frecuencia de corte sería de:

$$f_c = \frac{1}{2 \times \pi \times 100 \times 10^{-9} \times 10 \times 10^3} \rightarrow f_c = 159.15 \text{ Hz}$$

Se obtiene una frecuencia inferior, pero que sirve igualmente porque aún se está filtrando el ruido del motor.

2.4.2 Diseño del divisor de tensión.

Primero de todo se debe de conocer cuál va a ser la tensión máxima de salida por el tacogenerador. Como ya se mencionó, la salida del tacogenerador es el doble de la tensión de alimentación, y como se sabe que la tensión máxima de alimentación es de 12 V cuando se está trabajando al 100 % del ciclo de trabajo de la señal PWM que se genera en el puente H, la tensión que saldrá como máximo por el tacogenerador será de 24 V. Entonces se tiene que diseñar un divisor de tensión que reduzca esa tensión máxima de 24 V a una tensión como máximo de 10 V.

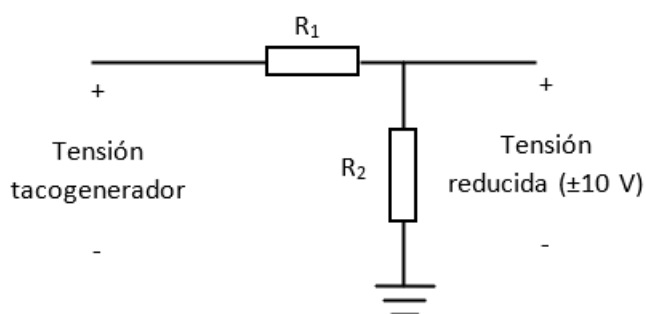


Figura 28. Esquemático del circuito Divisor de Tensión.

$$V_{reducida} = V_{Taco.} \times \frac{R_2}{R_1 + R_2} \rightarrow \frac{V_{reducida}}{V_{Taco.}} = \frac{R_2}{R_1 + R_2} \rightarrow \frac{10}{24} = \frac{R_2}{R_1 + R_2} \rightarrow$$

→ Se da un valor cualquiera para R_2 . Para un valor de $1\,800\,\Omega$ para R_2 , R_1 valdrá →

$$\rightarrow \frac{10}{24} = \frac{1\,800}{R_1 + 1\,800} \rightarrow R_1 = \frac{24}{10} \times \left(1\,800 - \frac{10}{24} \times 1\,800\right) \rightarrow R_1 = 2\,520\,\Omega$$

Todo el circuito, tanto el filtro paso bajo como el divisor de tensión, queda como se muestra en la Figura 29.

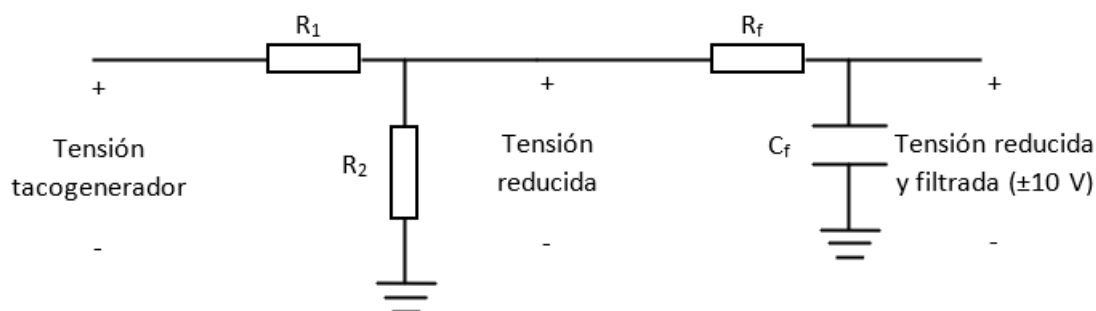


Figura 29. Circuito esquemático del divisor de tensión con el filtro paso bajo.

Se alimentará al circuito con la tensión de salida que proporciona el tacogenerador, se reducirá y filtrará a una tensión de $\pm 10\text{ V}$ para que se pueda introducir como entrada analógica del módulo SM 1234.

2.4.3 Conexión del circuito físico.

La conexión que tiene este circuito, situado en el mismo circuito impreso perforado que el circuito del sensor TCST2103, con la salida del tacogenerador es a través del mismo cable cat 5e UTP 4x2x24 que se usa para la conexión del sensor TCST2103 con su otra parte del circuito situada en el circuito impreso perforado.

Para este circuito se utilizará los dos cables sobrantes que se mencionaron.

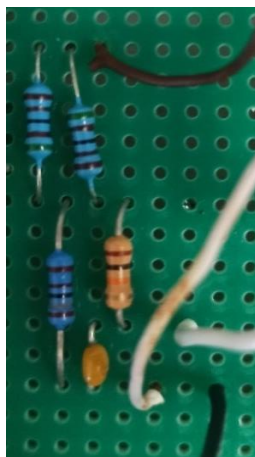


Figura 30. Circuito del divisor de tensión y filtro paso bajo físico.

El cableado y componentes que tiene este circuito es el siguiente:

- Los cables de color marrón y blanco con manchas marrones son la salida del tacogenerador, positivo y negativo respectivamente.
- El cable de color blanco, situado entre la resistencia de color marrón que corresponde a la resistencia del filtro de $10\,000\ \Omega$ y el condensador del filtro de valor $100\ \text{nF}$, y el cable negro de abajo son la señal de la tensión reducida y filtrada que se le introduce a la entrada del PLC.
- Las resistencias del divisor que se utilizaran son; para formar la R_2 se utiliza la resistencia normalizada de $1\,800\ \Omega$, y para la R_1 se obtiene del paralelo de dos resistencias del mismo valor de $5\,100\ \Omega$, que hace un valor de $2\,550\ \Omega$, un poco más del calculado, pero con las tolerancias de las resistencias el valor es muy parecido.

3. DESCRIPCIÓN TIA PORTAL.

3.1 TIA PORTAL.

TIA Portal, o lo que significa sus siglas en español, Portal de Automatización Totalmente Integrada, es un software que facilita la entrada, sin restricciones, a servicios de automatización digitalizados, desde la planificación digital y la ingeniería integrada hasta la operación transparente. Con varias versiones disponibles, haciendo cada vez más posible reducir los tiempo de comercialización y aumentar la productividad de empresas con diagnósticos.

Por lo tanto, TIA Portal ayuda a las empresas con dicho software a adentrarse a la automatización digital de la empresa. Como parte de la Digital Enterprise Suite, junto con PLM y MES, complementa la variedad integral de ofertas de Siemens para empresas que van en camino de la Industria 4.0. [13]

3.2 CREACIÓN DE PROYECTO.

Los pasos que seguir para crear un proyecto nuevo son:

1. Seleccionar el comando "Nuevo" del menú "Proyecto" que se encuentra en la esquina superior izquierda.

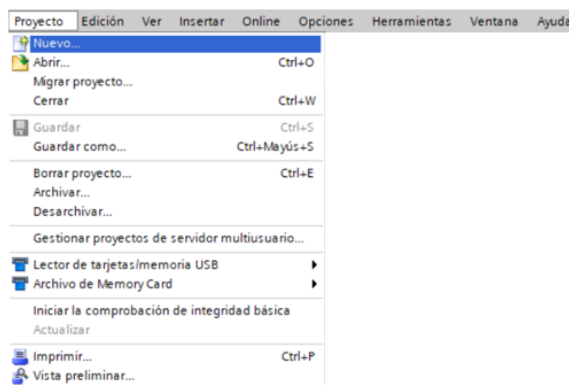


Figura 31. Iniciar proyecto nuevo.

2. Se abrirá una ventana flotante "Crear proyecto" donde se tendrá que poner un nombre al proyecto y una ruta para tenerlo guardado. Se podrá establecer la versión del proyecto en la que desea trabajar.

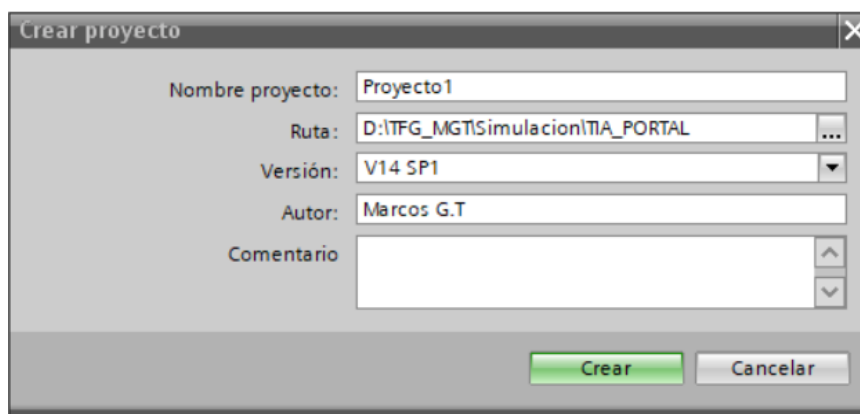


Figura 32. Crear el proyecto.

3. Finalmente se tendrá que pulsar el botón “Crear” situado en la esquina inferior derecha del cuadro anterior.

Se debe tener mucho cuidado con la versión que se establezca debido a la compatibilidad entre versiones. Los proyectos que se guardan en la versión V14 SP1 no son compatibles con versiones anteriores debido a la ampliación de funciones de la versión actual. Los proyectos de versión V14 SP1, que es la misma versión que se utiliza en este proyecto, solo pueden abrirse con el TIA Portal V14 SP1 o con versiones más recientes. [14]

Una vez creado el proyecto se deberá de añadir todos los dispositivos que se tienen, tanto la CPU del PLC como sus módulos añadidos, como se puede ver en las Figuras 33, 34 y 35. En la Figura 36 se muestran todo el equipo disponible junto.

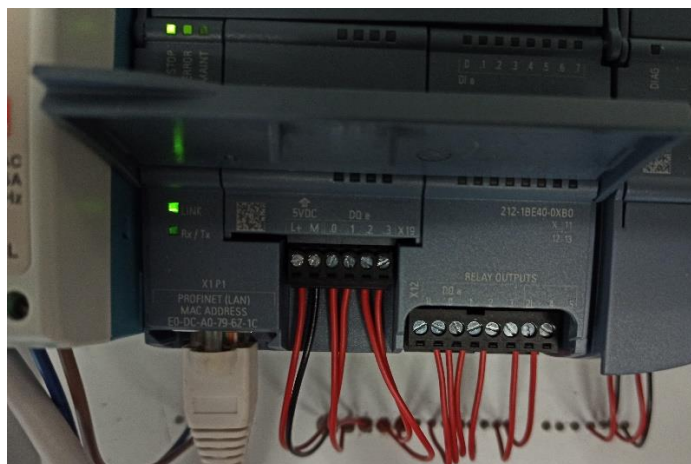


Figura 33. Controlador SIMATIC S7-1200 CPU 1212C.



Figura 34. Módulo añadido de 4 salidas digitales.



Figura 35. Módulo añadido de 4 entradas y 2 salidas analógicas.

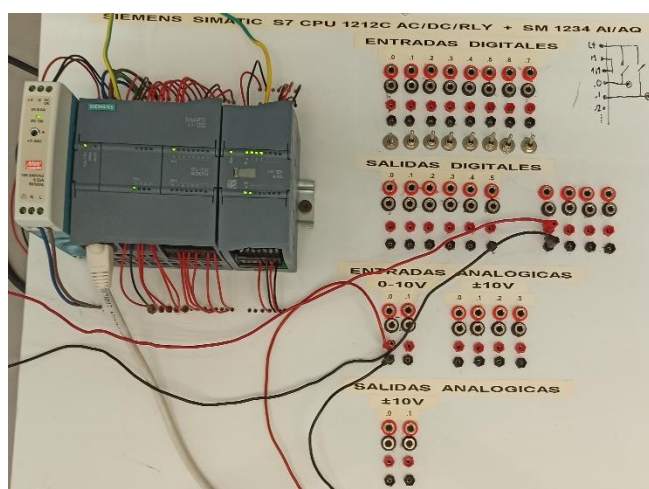


Figura 36. Conjunto de controlador y módulos añadidos.

Para añadir los dispositivos en TIA Portal se tiene que ir a la ventana que se sitúa a la izquierda y seguir estos pasos:

Árbol del proyectos → Dispositivos → Proyecto1 → “Agregar dispositivos”

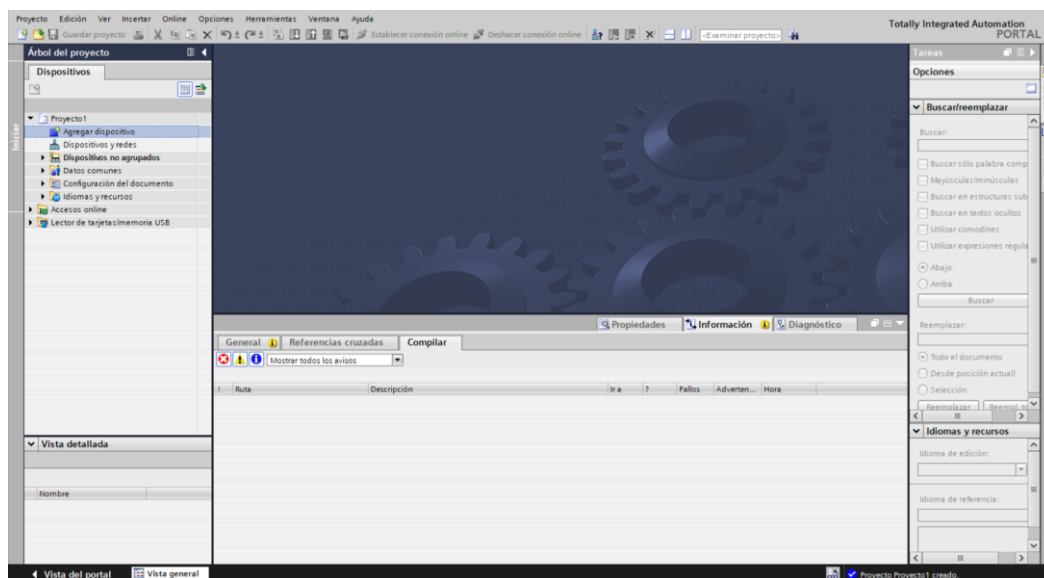


Figura 37. Agregar dispositivo.

Pulsando “Agregar dispositivos” saldrá una ventana flotante donde se muestran los distintos dispositivos que se pueden añadir en la versión que se está usando de TIA Portal. En el proyecto, como ya se ha mencionado durante el documento, se usa un controlador SIMATIC S7-1200 con una CPU 1212C DC/AC/Rly, de los tres disponibles, el que corresponde al que se tiene en disposición es el 212-1BE40-0XB0 como se puede ver en la Figura 33.

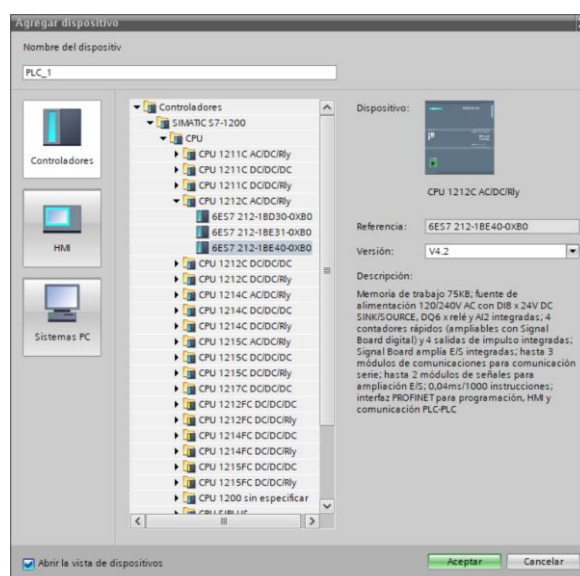


Figura 38. Seleccionar el dispositivo de controlador.

Una vez añadido el controlador queda añadir los dos módulos extras, que se explicarán el uso que se les dan en el proyecto. Para ello se tiene que ir a la ventana de la derecha y se verá el catálogo de hardware disponibles.

Para añadir el módulo de salidas digitales de 5 V se tiene que ir a:

Signal Boards → DQ → DQ 4x5VDC → 222-1AD30-0XB0 y se arrastra al recuadro marcado de color blanco.

Como se observa en la Figura 34, este modelo que se añade corresponde con el que se tiene físicamente.

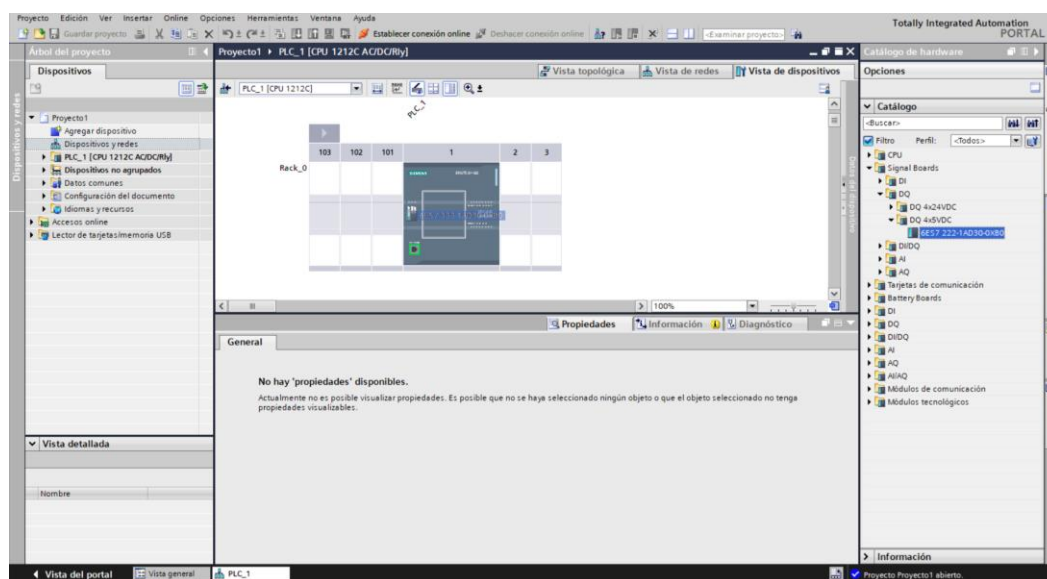


Figura 39. Añadir módulo de 4 salidas digitales.

Con este módulo se tendrá 4 salidas digitales, las cuales una de ellas se utilizará para la generación de la señal del PWM de amplitud de 5 V, ya que con las salidas digitales que la CPU del PLC tiene como predeterminadas no están capacitadas para generar pulsos.

El último módulo que queda por añadir corresponde al módulo de entradas y salidas analógicas de ± 10 V como se aprecia en la Figura 35:

AI/AQ → AI 4x13BIT/AQ 2x14BIT → 234-4HE32-0XB0 y se arrastra al espacio 2.

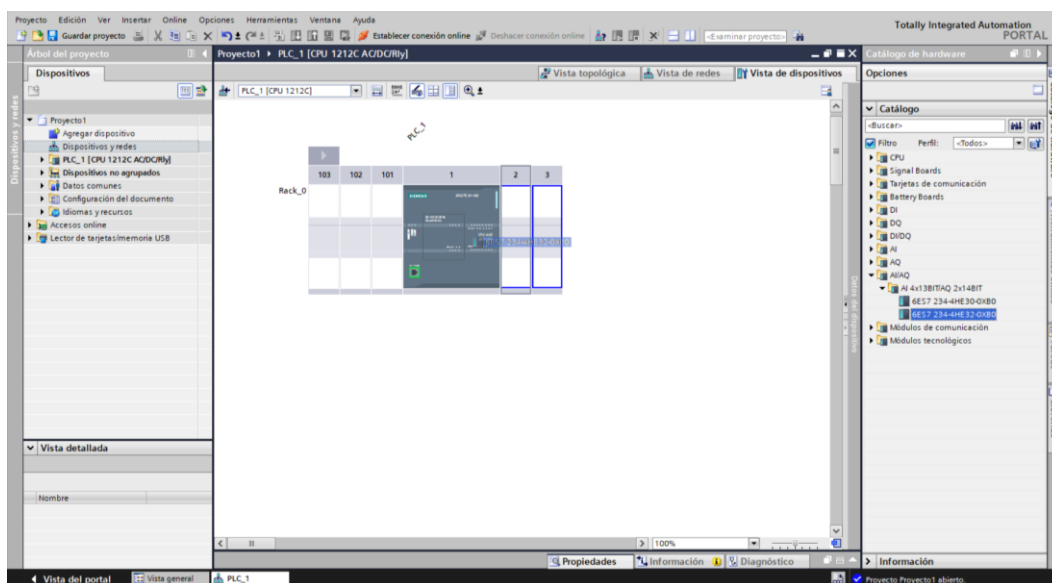


Figura 40. Añadir módulo de 4 entradas y 2 salidas analógicas.

Con este último módulo se está añadiendo 4 entradas analógicas de ± 10 y 2 salidas analógicas del mismo rango. Se utilizará una de las entradas para hacer la lectura de la tensión de salida del tacogenerador del motor DC.

Ya estructurado el proyecto de TIA Portal con todos los módulos que se tienen físicamente, el proyecto, con la comparación del real, queda de la siguiente manera:



Figura 41. Comparación entre lo físico y virtual.

3.3 BLOQUES DE PROGRAMA.

Los bloques de programa de TIA Portal sirven para desarrollar el código de tu programa, el cual se puede dividir en diversas partes para facilitar la comprensión.

En este apartado se comentará cada una de las partes del código de forma individual, así como las decisiones que se han llevado para su programación. En el capítulo 7 Anexo se incluirá todo el código y tabla de variables.

3.3.1 Main.

El bloque Main se trata de un bloque de organización (de ahora en adelante OB) que viene por defecto una vez se ha configurado el PLC. Este OB [14] se evalúa cíclicamente y es un bloque lógico de orden superior en el programa. En él se puede programar con las diferentes instrucciones que hay o, también, tiene la posibilidad de llamar a otros bloques.

Un ciclo de ejecución del programa abarca los siguientes pasos:

1. El sistema operativo inicia el tiempo de vigilancia del ciclo.
2. Escribe los valores de la memoria imagen de proceso de las salidas en los módulos de salidas.
3. Lee el estado de las entradas en los módulos de entradas y actualiza la memoria imagen de proceso de las entradas.
4. Procesa el programa de usuario y ejecuta las operaciones correspondientes.
5. Al final de un ciclo, el sistema operativo puede ejecutar otras tareas pendientes y a continuación, la CPU regresa al comienzo del ciclo y reinicia la vigilancia del tiempo de ciclo.

Dentro del Main se encuentran las distintas instrucciones separadas en segmentos que se han creado para obtener el código del programa.

3.1.1.1 Arranque y parada.

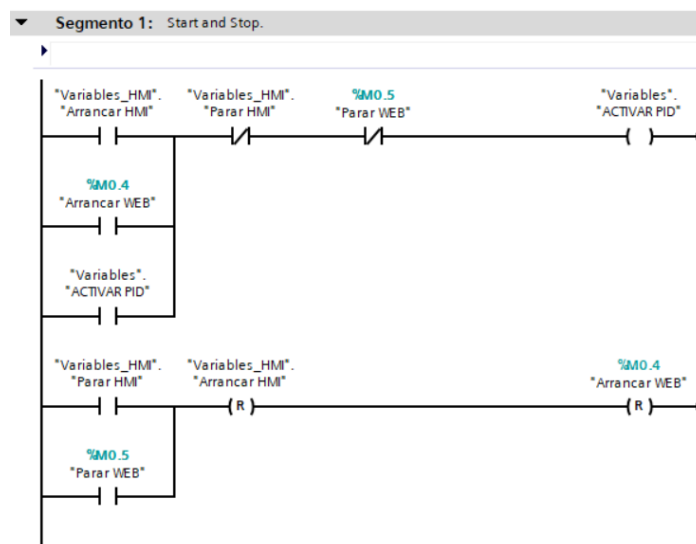


Figura 42. Segmento 1 del Main. Arranque y Paro.

En el primer segmento del bloque Main se encuentra el arranque y paro del PID, que será también del sistema global.

En él se cuenta con dos modalidades para el arranque y el paro, una a través de la pantalla HMI, y la otra por medio de la página web. En ambos modos se usan las siguientes

operaciones lógicas: una de tamaño bit de contacto normalmente abierto, para el caso del arranque, y otra de contacto normalmente cerrado para el paro.

El funcionamiento de este segmento se realiza de este modo:

- Si se pulsa uno de los dos arranques y ningún paro es apretado, se activará la operación lógica de tamaño bit de asignación que está asociado con la variable que activa el PID y por tanto se podrá realizar el control de velocidad.
- Si se da el caso de que se aprieta uno de los paro cuando el sistema está funcionando, el sistema se apagará y se reiniciarán las variables de arranque. Hasta que no se desactive cualquiera de los paros, el sistema no volverá a arrancar de nuevo.

3.1.1.2 SetPoint.

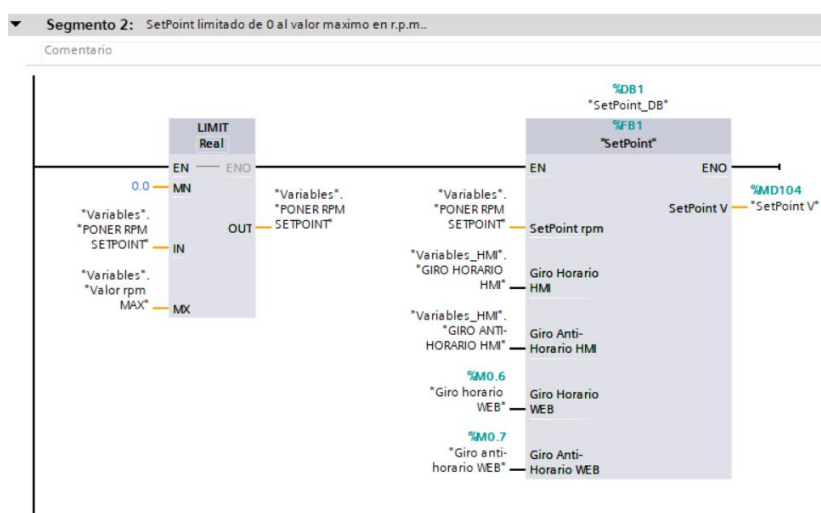


Figura 43. Segmento 2 del Main. SetPoint.

En el segundo segmento se establece el valor de referencia de velocidad a la que se desea que el motor esté girando.

Para prevenir que no se sobrepase de los límites de velocidad que el motor pueda soportar en las condiciones que esta alimentado, se ha puesto la instrucción de función matemática llamada LIMIT con la cual se limitará el valor de velocidad que el usuario pueda escribir tanto en la pantalla HMI como en la página web. Los valores límites son: el valor mínimo, que para este caso es de 0 r.p.m., y el valor máximo, el cual se ha sacado de la siguiente manera:

$$v_{max} = \frac{V_{taco.max}}{K_T} = \frac{12 (V) \times 2}{14.5 \times 10^{-3} \left(\frac{V}{r.p.m.} \right)} \rightarrow v_{max} \cong 1655 r.p.m.$$

De la tabla de características del motor DC se extrajo que la constante tacométrica tiene un valor de 14.5 mV/r.p.m. , y sabiendo que la máxima tensión de alimentación que se le suministra al motor por la señal PWM es de 12 V, con el ciclo de trabajo del 100%, y además, conociendo que la tensión por el tacogenerador es el doble que la alimentación, se calculará la velocidad máxima de giro del motor como se muestra en la ecuación anterior.

Teniendo ya controlado el valor de la velocidad que se introduce, ahora toca hablar del siguiente bloque que tiene este segmentos. Se trata de un bloque de programa que se debe de crear desde cero, pero en vez de ser un OB, como es el Main, se trata de bloque de función (de ahora en adelante FB) [14] los cuales son bloques lógicos que depositan sus valores de forma permanente en bloques de datos de instancia, de modo que siguen estando disponibles después de procesar el bloque.

Este FB se encargará de pasar la velocidad de referencia que se introduce en r.p.m. a voltios, es decir, a tensión. Se tiene que realizar este cambio de unidad porque la regulación se realiza mediante tensiones y se necesita tener la velocidad de referencia comprendida entre $\pm 10 \text{ V}$, lo cual se consigue de la siguiente manera:

SetPoint									
	Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible d...	Escrib...	Visible en ..	Valor de a..	Comentario
1	Input								
2	SetPoint rpm	Real	0.0	No remane...	☒	☒	☒	☐	
3	Giro Horario HMI	Bool	false	No remane...	☒	☒	☒	☐	
4	Giro Anti-Horario HMI	Bool	false	No remane...	☒	☒	☒	☐	
5	Giro Horario WEB	Bool	false	No remane...	☒	☒	☒	☐	
6	Giro Anti-Horario WEB	Bool	false	No remane...	☒	☒	☒	☐	
7	Output								
8	SetPoint V	Real	0.0	No remane...	☒	☒	☒	☐	
9	InOut								
10	<Agregar>								
11	Static								
12	Aux	Real	0.0	No remane...	☒	☒	☒	☐	
13	Aux2	Real	0.0	No remane...	☒	☒	☒	☐	
14	Temp								
15	<Agregar>								
16	Constant								
17	<Agregar>								

Figura 44. Interfaz del FB SetPoint.

Dentro de todos los FB se encuentran la interfaz del bloque, en donde se puede agregar tantas entradas y salidas como el creador quiera y crea necesarias para obtener el FB que haga la función que esté llevando a cabo.

Para este FB, se necesita: Cinco entradas, de las cuales una será para la velocidad de referencia en r.p.m., y las otras cuatro son el sentido de giro que se le esté introduciendo al motor, dos referidos en la pantalla HMI y las otras dos para la página web. Y una salida, que será la tensión comprendida entre $\pm 10 \text{ V}$. También se han agregado variables intermedias para ayudar en los cálculos.

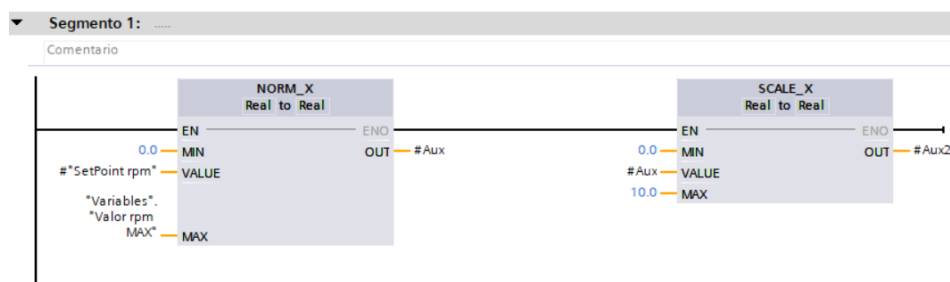


Figura 45. Segmento 1 del FB SetPoint.

En el primer segmento del FB SetPoint se encuentra el normalizado y escalado de la velocidad en r.p.m. a voltios. Esto se realiza con las instrucciones:

- NORM_X: normaliza la velocidad en r.p.m. en valores comprendidos entre 0 y 1. Se tiene que especificar los valores mínimo y máximo de la velocidad en r.p.m. .
- SCALE_X: escala ese valor que se ha normalizado entre los valores mínimo y máximo de tensión, que son de 0 y 10 V.

La tensión se encuentra comprendida entre estos valores debido a que la velocidad de referencia en r.p.m. que se le introduce solo está en valores positivos, por lo que de esta forma no se puede diferenciar el sentido de giro. Por ello, para solucionar eso, en el FB se introduce esas 4 entradas vinculadas con el sentido de giro del motor que se comentaron anteriormente, para poder tener la tensión entre ± 10 V, esto se consigue en el segundo segmento del FB:

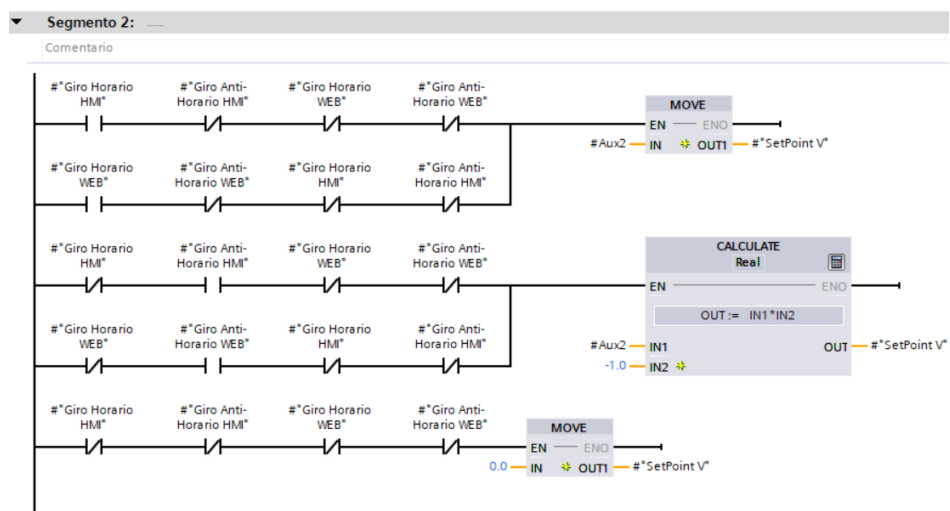


Figura 46. Segmento 2 del FB SetPoint.

Si se tiene que el sentido de giro que se establece es horario, tanto si se introduce en la pantalla HMI como en la página web, la tensión estará entre los valores de 0 y 10 V.

Si es el otro caso, que el sentido de giro es el antihorario, la tensión será entre -10 y 0 V. Esto se logra con la instrucción de función matemática CALCULATE, que al multiplicar por -1 al valor escalado entre 0 y 10 se cambia dichos valores al signo negativo.

Si se da el caso de que no se ha establecido ningún sentido de giro, el valor de tensión será de 0 V. Esta instrucción se puso para prevenir que cuando el motor se encontrase girando en uno de los sentidos se quede almacenado el valor de tensión anterior y al desactivar el sentido de giro el motor siga dando tensión, ya que al ser una entrada analógica puede que siga dando algún valor.

3.1.1.3 Generador PWM.

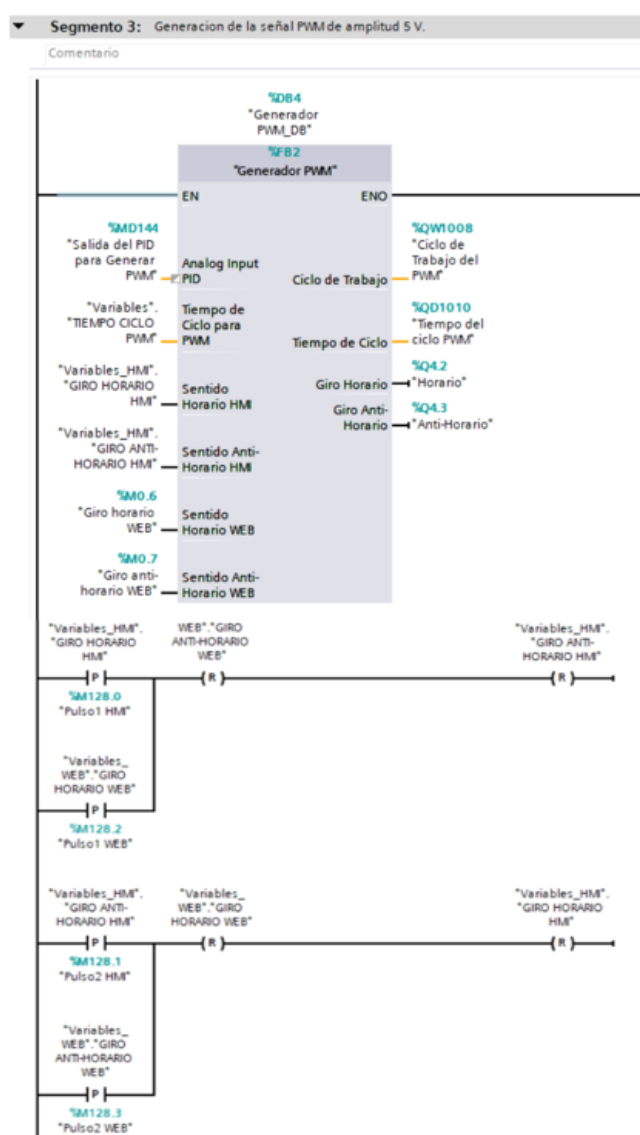


Figura 47. Segmento 3 del Main. Generador PWM.

Al igual que en el segmento 2, en los siguientes segmentos también se han creado distintos FB.

En este segmento numero 3 lo que se hace es la generación de los pulsos PWM de amplitud 5 V que se le introducen al circuito puente H para tener la señal a 12 V de amplitud.

Para poder generar pulsos PWM en el PLC se tiene que activar primero, en las propiedades del PLC, uno de los generadores de impulsos:

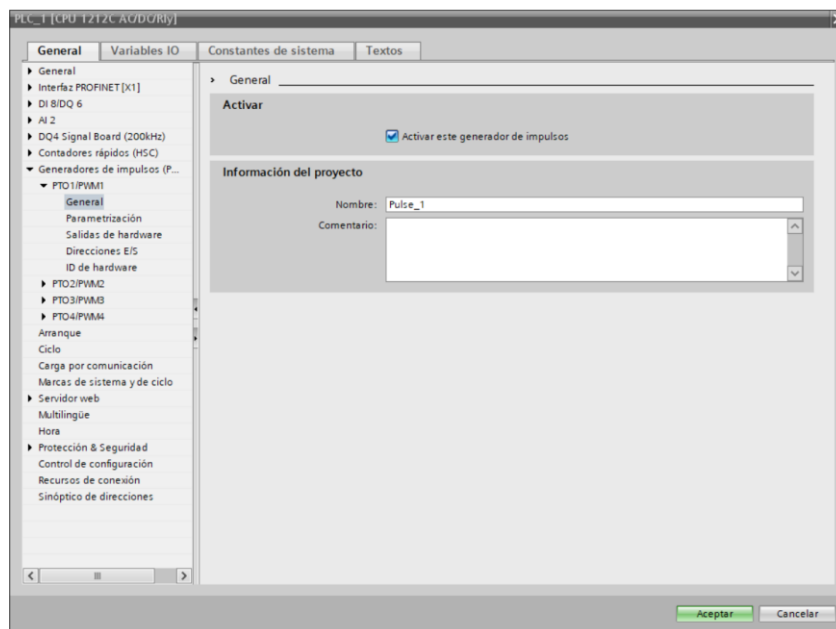


Figura 48. Activar Generadores de impulsos.

Una vez activado se deberá de indicar la salida por donde se van a generar dichos impulsos. Como ya se explicó, será una de las salida del módulo añadido SB 1222 DC 200 KHz DQ 4x5VDC 0.1A por estar capacitada para ello.

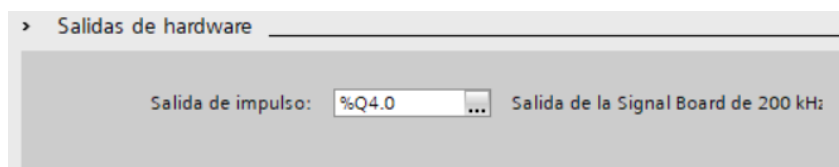


Figura 49. Introducir salida de la señal PWM.

También se ha habilitado la opción de poder cambiar, durante la ejecución de la señal PWM, el tiempo de duración del ciclo.

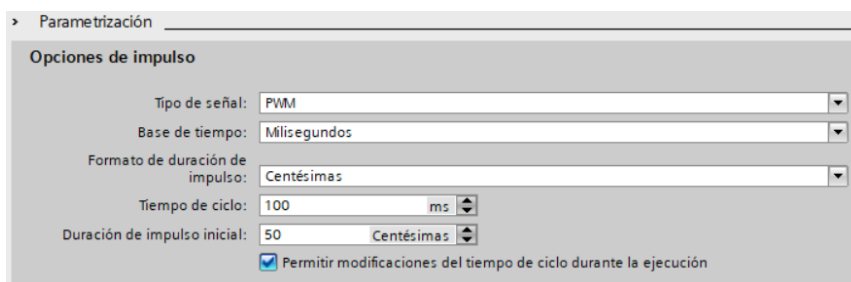


Figura 50. Habilitar la modificación del tiempo de ciclo.

Cada generador de impulsos tiene sus correspondientes direcciones de entrada y salida, y su ID de hardware. En las Figuras 51 y 52 se observa cuáles son las direcciones e ID del generador activado.

Figura 51. Direcciones de entrada/salida del generador activado.

Figura 52. ID de hardware del generador activado.

Al haber marcado la casilla de poder cambiar la duración del ciclo, Figura 50, se ha modificado las direcciones inicial y final que están por defecto en las direcciones de entrada y salida. Ahora los dos primeros bytes de salida contienen la duración de impulso y del tercero al sexto contienen el tiempo de ciclo, es decir, que la dirección QW1008 está asignada para indicar el ciclo de trabajo de la señal PWM y la dirección QD1010 para el tiempo de ciclo.

Una vez habilitado el generador de impulsos toca hablar de que se hace dentro del FB llamado Generador PWM.

Generador PWM									
	Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible d...	Escrib...	Visible en ...	Valor de a...	Comentario
1	Input								
2	Analog Input PID	Int	0	No remane...	☒	☒	☒	☐	
3	Tiempo de Ciclo para ...	DWord	16#0	No remane...	☒	☒	☒	☐	
4	Sentido Horario HMI	Bool	false	No remane...	☒	☒	☒	☐	
5	Sentido Anti-Horario ...	Bool	false	No remane...	☒	☒	☒	☐	
6	Sentido Horario WEB	Bool	false	No remane...	☒	☒	☒	☐	
7	Sentido Anti-Horario ...	Bool	false	No remane...	☒	☒	☒	☐	
8	Output								
9	Ciclo de Trabajo	Word	16#0	No remane...	☒	☒	☒	☐	
10	Tiempo de Ciclo	DWord	16#0	No remane...	☒	☒	☒	☐	
11	Giro Horario	Bool	false	No remane...	☒	☒	☒	☐	
12	Giro Anti-Horario	Bool	false	No remane...	☒	☒	☒	☐	
13	InOut								
14	<Agregar>								
15	Static								
16	Aux1	Real	0.0	No remane...	☒	☒	☒	☐	
17	Aux2	Int	0	No remane...	☒	☒	☒	☐	
18	Aux3	Real	0.0	No remane...	☒	☒	☒	☐	
19	G.H.	Bool	false	No remane...	☒	☒	☒	☐	
20	G.AH.	Bool	false	No remane...	☒	☒	☒	☐	
21	Pulso1 HMI	Bool	false	No remane...	☒	☒	☒	☐	
22	Pulso1 WEB	Bool	false	No remane...	☒	☒	☒	☐	
23	Pulso2 HMI	Bool	false	No remane...	☒	☒	☒	☐	
24	Pulso2 WEB	Bool	false	No remane...	☒	☒	☒	☐	
25	Temp								
26	<Agregar>								
27	Constant								
28	<Agregar>								

Figura 53. Interfaz del FB Generador PWM.

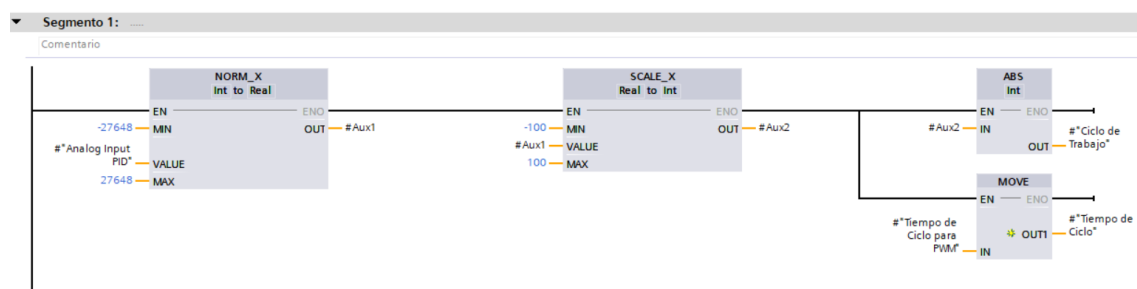


Figura 54. Segmento 1 del FB Generador PWM.

En el primer segmento del FB Generador PWM lo que se hace es un normalizado y escalado de la señal de entrada que proviene de la salida del bloque del PID, que se comentará más adelante, para obtener como una de las salidas del FB el valor del ciclo de trabajo entre 0 y 100 %.

Además, se introduce como una entrada el valor de la duración de la señal PWM que el usuario introduce para la duración de la señal PWM, dando dicho valor directamente a una de las salidas del FB y vinculando a la variable del tiempo de ciclo del PWM.

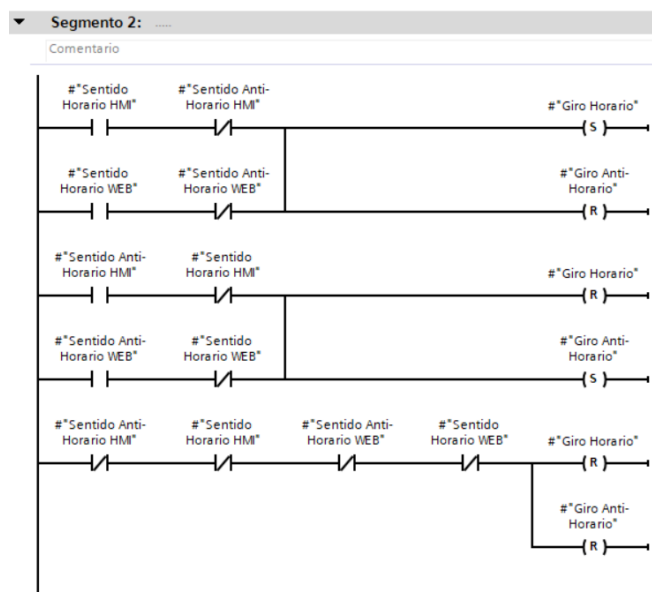


Figura 55. Segmento 2 del FB Generador PWM.

El siguiente segmento, Figura 55, establece el sentido de giro del motor con las entradas de sentido de giro horario y antihorario que el usuario pulse para que el motor gire en uno de los dos sentidos. Estas entradas de sentido de giro son tanto las que se encuentran en la pantalla HMI como las de la página web. Al pulsar una de las entradas de sentido de giro se activará y desactivará, dependiendo de que sentido se pulse, las salidas digital del PLC vinculadas al giro del motor, en el caso que se está explicando serían las salidas digitales Q4.2 y Q4.3.

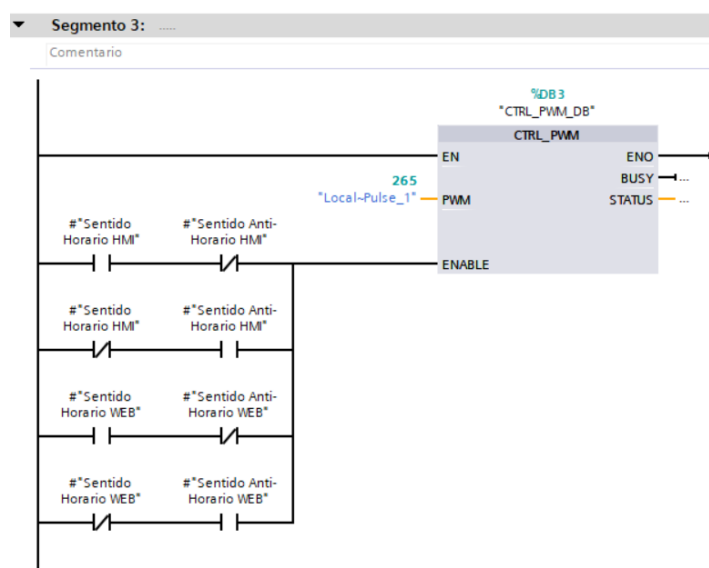


Figura 56. Segmento 3 del FB Generador PWM.

Por último, queda de explicar el tercer segmento del FB Generador PWM. En este segmento se activa el control PWM con la instrucción avanzada `CTRL_PWM`, en la cual se introduce la ID del hardware del generador de impulso, que se activó previamente, y cuando se debe de activar la señal PWM.

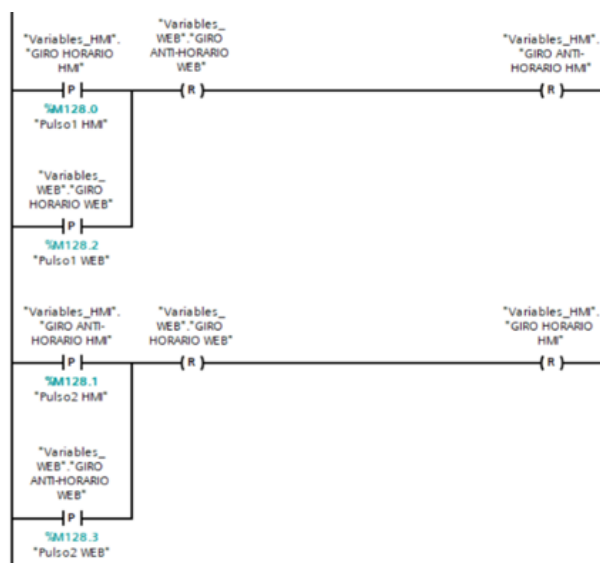


Figura 57. Instrucciones para desactivar el sentido de giro cuando se cambia.

Las últimas instrucciones que están dentro del segmento 3 y fuera del FB Generador PWM, Figura 57, sirven para desactivar las entradas vinculadas con el sentido de giro del motor.

Lo que se hace en esta parte es que cuando el usuario quiera cambiar el sentido de giro mientras el motor esté girando en uno de los sentidos, se desactive la señal vinculada al sentido de giro en el que se movía el motor para activar la señal que está vinculada al sentido

de giro que el usuario ha introducido como nuevo sentido. Esto se ha hecho con la instrucción de pulso positivo, la cual se activa solamente cuando se active dicha señal con el flanco positivo, desactivando el otro sentido de giro con la instrucción reset.

3.1.1.4 Velocidad por el Tacogenerador.

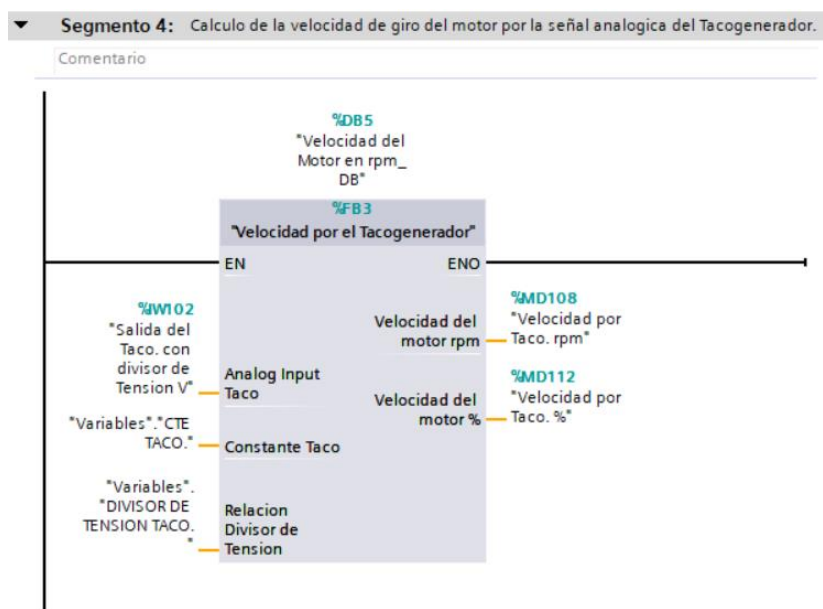


Figura 58. Segmento 4 del Main. Velocidad por el Tacogenerador.

El segmento 4 calcula la velocidad en r.p.m. usando la señal analógica que está leyendo del divisor de tensión que se ha comentado anteriormente. Dentro de este FB llamado Velocidad por el Tacogenerador se tiene:

Velocidad por el Tacogenerador									
	Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible d...	Escrib...	Visible en ..	Valor de a..	Comentario
1	Input								
2	Analog Input Taco	Word	16#0	No remane...	☒	☒	☒	☐	
3	Constante Taco	Real	0.0	No remane...	☒	☒	☒	☐	
4	Relacion Divisor de Te...	Real	0.0	No remane...	☒	☒	☒	☐	
5	Output								
6	Velocidad del motor r...	Real	0.0	No remane...	☒	☒	☒	☐	
7	Velocidad del motor %	Real	0.0	No remane...	☒	☒	☒	☐	
8	InOut								
9	<Agregar>								
10	Static								
11	Aux1	Real	0.0	No remane...	☒	☒	☒	☐	
12	Tension Taco Escalada	Real	0.0	No remane...	☒	☒	☒	☐	
13	Aux2	Real	0.0	No remane...	☒	☒	☒	☐	
14	Aux3	Real	0.0	No remane...	☒	☒	☒	☐	
15	Aux4	Real	0.0	No remane...	☒	☒	☒	☐	
16	Temp								
17	<Agregar>								
18	Constant								
19	<Agregar>								

Figura 59. Interfaz del FB Velocidad por el Tacogenerador.

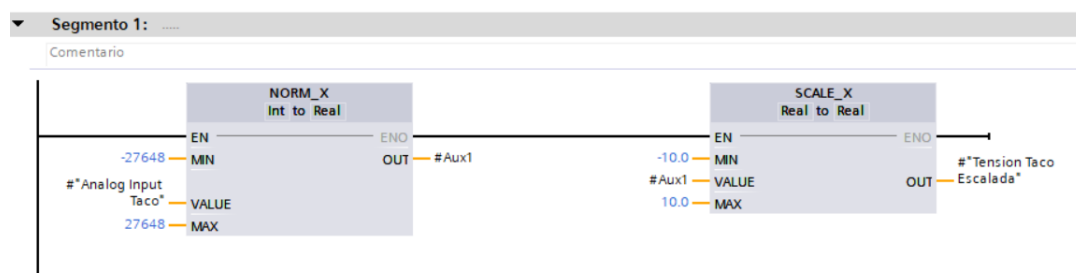


Figura 60. Segmento 1 del FB Velocidad por el Tacogenerador.

Como se viene haciendo en los demás FB que se han explicado, lo primero que se hace es un normalizado y escalado de la señal, en este caso de la lectura de la señal analógica que proviene del divisor de tensión de la salida del tacogenerador, entre los valores ± 10 V, ya que, aunque la salida del tacogenerador es el doble que la alimentación, el PLC no admite tanta tensión y por ello se diseñó el divisor de tensión del que se está tomando la lectura de la señal analógica. Se debe tener en cuenta que este divisor no afecta en el cálculo de la velocidad real de giro del motor porque se tiene en consideración.

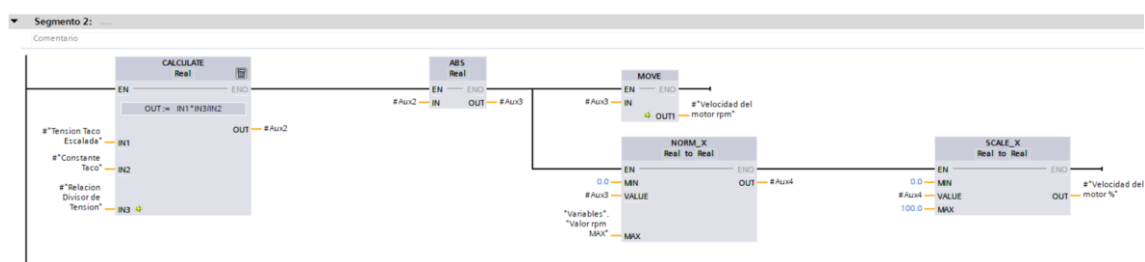


Figura 61. Segmento 2 del FB Velocidad por el Tacogenerador

Con la instrucción de función matemática CALCULATE se establecerá la velocidad en r.p.m. a la que el motor está girando con la siguiente expresión:

$$v_{giro} = \frac{Tension_{divisor} \times Relacion_{divisor}}{Constante_{Tacométrica}}$$

Siendo;

Tension_{divisor}: la tensión normalizada y escalada previamente.

Relacion_{divisor}: la relación entre el valor real de la tensión del

tacogenerador con la tensión del divisor de tensión (de valor $\frac{24 \text{ V (salida real)}}{10 \text{ V (divisor)}}$).

Constante_{Tacométrica}: relación entre tensión (V) y velocidad (r.p.m.)

(de valor $0.0145 \frac{\text{V}}{\text{r.p.m.}}$).

También se calcula el porcentaje de velocidad a la que se está girando el motor, haciendo un normalizado y escalado de la velocidad de giro en la que está girando en un

momento determinado el motor entre los valores mínimo y máximo de velocidad que ya se han explicado cuales eran en el subapartado 3.1.1.2 SetPoint.

3.1.1.5 Velocidad por el Encoder.

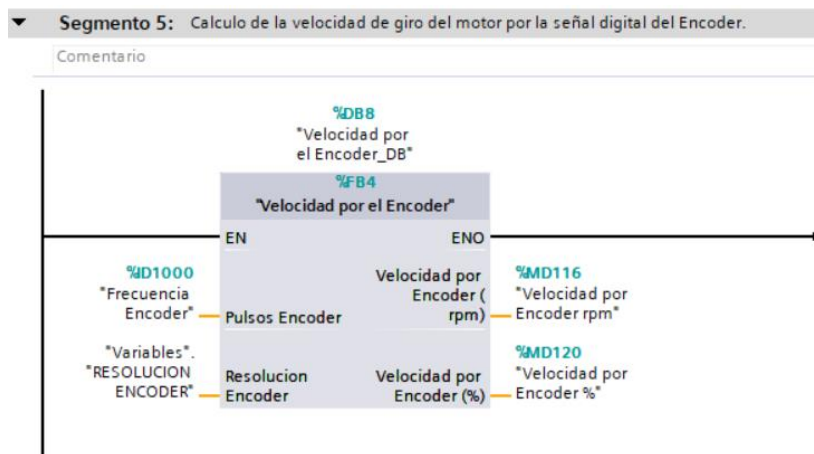


Figura 62. Segmento 5 del Main. Velocidad por el Encoder.

Al igual que en el anterior segmento, en el segmento 5 se calcula la velocidad de giro, pero esta vez con la señal digital de la lectura de los pulsos que se están leyendo por el encoder gracias al circuito del sensor optoacoplador que se ha descrito anteriormente en el subapartado 2.2 Circuito sensor optoacoplador TCST2103.

Para su cálculo, primero se tiene que ir a las propiedades del PLC y activar uno de los contadores rápidos para poder realizar una lectura más precisas de los pulsos del encoder.

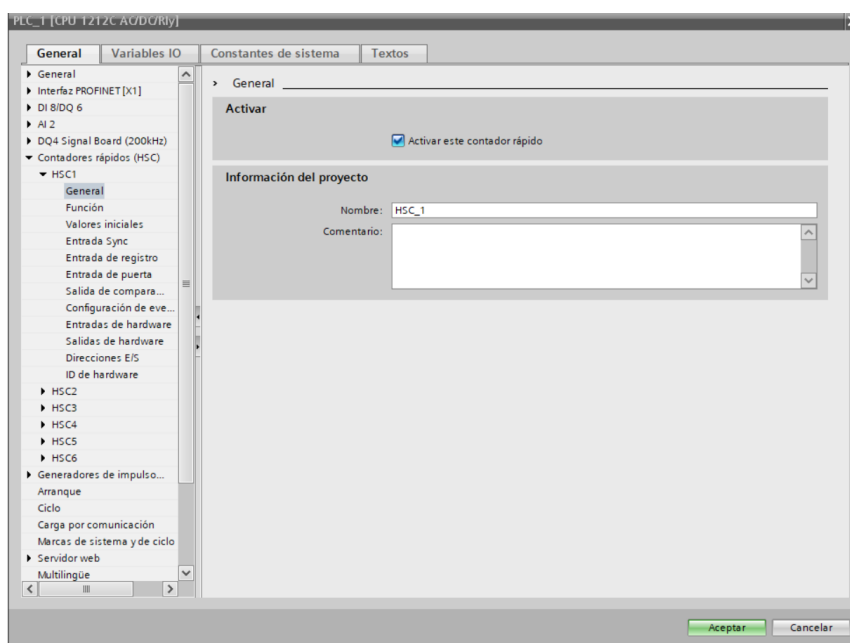


Figura 63. Activar un contador rápido.

Se tiene que indicar que tipo de conteo se va a realizar, en el caso del proyecto se realiza un conteo de frecuencia, la fase de servicio, que viene influida por el tipo de encoder disponible, es monofásico, y el periodo de medición de frecuencia, que por ser un conteo de frecuencia se ha activado esta opción, se establece el más pequeño para que sea más fina la toma de información.

The screenshot shows a configuration window titled 'Función'. It contains several dropdown menus and a text input field. The 'Modo de conteo' dropdown is set to 'Frecuencia'. The 'Fase servicio' dropdown is set to 'Monofásica'. The 'Sentido de conteo dado por:' dropdown is set to 'Programa de usuario (control interno de sentido)'. The 'Sentido de conteo inicial:' dropdown is set to 'Incrementar contador'. The 'Período de medición de frecuencia:' text input field is set to '0.01', and the unit dropdown is set to 'sec'.

Figura 64. Función del contador rápido activado.

También se debe de indicar cual va a ser la entrada por donde se realice la lectura de los pulsos del encoder, como es una entrada digital se usa la I0.0, ya que es una de las que tiene una entrada integrada de 100 kHz.

Al igual que el generador de impulsos, el contador rápido también tiene direcciones de entrada y salida y una ID de hardware.

The screenshot shows a configuration window titled 'Direcciones E/S'. It contains a section titled 'Direcciones de entrada' with four fields: 'Dirección inicial:' set to '1000', 'Dirección final:' set to '1003', 'Bloque de organización:' set to '--- (Actualización automática)', and 'Memoria imagen de proceso:' set to 'Actualización automática'. Each of the last three fields has a three-dot menu icon to its right.

Figura 65. Direcciones de entrada/salida del contador activado

The screenshot shows a configuration window titled 'ID de hardware'. It contains a section titled 'ID de hardware' with a single text input field labeled 'ID de hardware:' set to '257'.

Figura 66. ID de hardware del contador activado.

Además, se tiene que modificar la entrada digital I0.0 para que tenga un filtro de entrada de 0.1 milisegundo. Con esto se asegura de que se está tomando bien la información. Este valor de filtro se ha establecido tras realizar varias pruebas de tomas de información. La

mejor respuesta en cuanto a mejor información obtenida ha sido con el filtro de entrada que se muestra en la Figura 67.

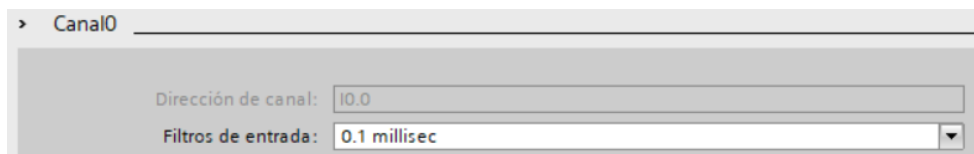


Figura 67. Filtro en la entrada del contador rápido I0.0.

Ya activado y explicado el contador rápido ahora se explicará las instrucciones que se llevan a cabo dentro del FB Velocidad por el Encoder:

Velocidad por el Encoder									
	Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible d...	Escrib...	Visible en ...	Valor de a...	Comentario
1	Input								
2	Pulsos Encoder	DInt	0	No remane...	☒	☒	☒	☐	
3	Resolucion Encoder	DInt	0	No remane...	☒	☒	☒	☐	
4	Output								
5	Velocidad por Encode...	Real	0.0	No remane...	☒	☒	☒	☐	
6	Velocidad por Encode...	Real	0.0	No remane...	☒	☒	☒	☐	
7	InOut								
8	<Agregar>								
9	Static								
10	Tiempo min/seg	Real	60.0	No rem...	☒	☒	☒	☐	
11	Aux1	Real	0.0	No remane...	☒	☒	☒	☐	
12	Aux2	Real	0.0	No remane...	☒	☒	☒	☐	
13	Temp								
14	<Agregar>								
15	Constant								
16	<Agregar>								

Figura 68. Interfaz del FB Velocidad por el Encoder.

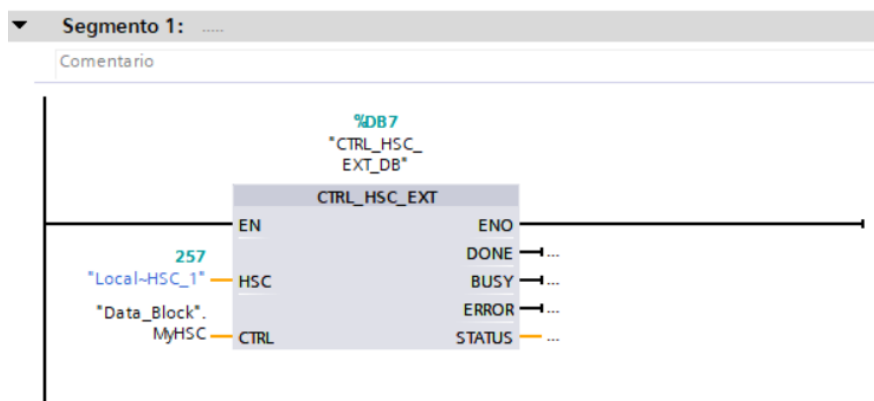


Figura 69. Segmento 1 del FB Velocidad por el Encoder.

En el primer segmento del FB Velocidad por el Encoder se está habilitando el contador rápido, que se ha activado antes, con la instrucción de categoría tecnológica CTRL_HSC_EXT, la cual necesita el ID de hardware del contador rápido y se debe de crear un bloque de datos (de ahora en adelante DB), donde se almacena los datos del contador rápido.

En este DB se debe de crear una variable estática de tipo de dato HSC_Frequency, al tratarse de un conteo por frecuencia. Al crearse la variable, aparece las siguientes

variables que aparecen en la Figura 70, de las cuales se tiene que poner a uno la variable con el nombre de EnHSC, que es la habilitación del contador rápido.

Data_Block									
	Nombre	Tipo de datos	Valor de arranq...	Remanen...	Accesible d...	Escrib...	Visible en ..	Valor de a...	Comentario
1	Static								
2	MyHSC	HSC_Frequency			☒	☒	☒		
3	Frequency	DInt	0		☒	☒	☒		
4	EnHSC	Bool	1		☒	☒	☒		
5	EnPeriod	Bool	false		☒	☒	☒		
6	NewPeriod	Int	0		☒	☒	☒		

Figura 70. Variables del DB Data_Block.

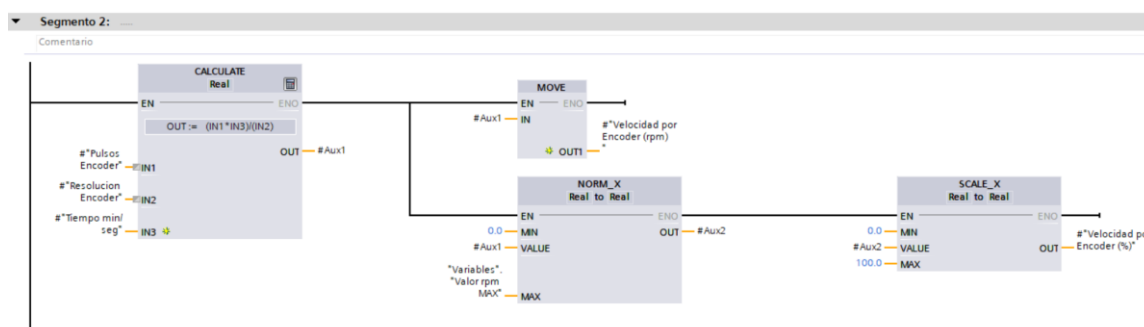


Figura 71. Segmento 2 del FB Velocidad por el Encoder.

En el último segmento de este FB se calcula la velocidad, en r.p.m. y en tanto por ciento, que nos proporciona la señal digital del encoder. El cálculo en porcentaje es igual que en el caso de la velocidad del tacogenerador, se normaliza y escala la velocidad de giro que se calcula.

Para el cálculo de la velocidad en r.p.m. se utiliza la siguiente expresión:

$$v_{giro} = \frac{\text{Pulsos}}{\text{segundo}} \times \left(\frac{1 \text{ revolucion}}{\text{Resolucion}_{encoder}} \right) \times \left(\frac{60 \text{ segundos}}{1 \text{ minuto}} \right)$$

Siendo;

Pulsos/segundo: la lectura de la entrada digital.

Resolucion_{encoder}: la divisiones de franjas negras que tiene el disco (de un valor de 30 Pulsos/vuelta completa).

3.1.1.6 Tensiones Vcc y Vout del motor.

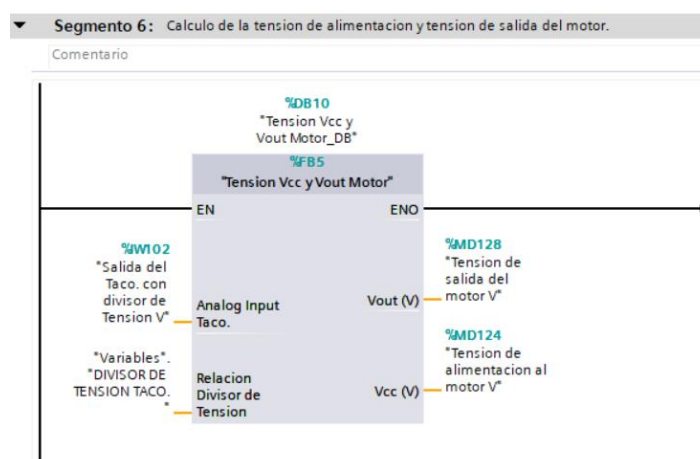


Figura 72. Segmento 6 del Main. Tensión Vcc y Vout Motor.

En este segmento se calcula el valor de la tensión de alimentación, ya que al ser una alimentación mediante una señal PWM no se conoce exactamente cuál es la tensión en continua con la que se está alimentado al motor, y la tensión de salida, que tampoco se conoce porque la señal que le llega al PLC esta reducida por el divisor de tensión.

Dentro de este FB se encuentra la interfaz con sus variables y los segmentos que se muestran en las siguientes figuras.

Tension Vcc y Vout Motor									
	Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible d...	Escrib...	Visible en ...	Valor de a...	Comentario
1	▼ Input								
2	■ Analog Input Tacho.	Word	16#0	No remane...	☒	☒	☒		
3	■ Relacion Divisor de Te...	Real	0.0	No remane...	☒	☒	☒		
4	▼ Output								
5	■ Vout (V)	Real	0.0	No remane...	☒	☒	☒		
6	■ Vcc (V)	Real	0.0	No remane...	☒	☒	☒		
7	▼ InOut								
8	■ <Agregar>								
9	▼ Static								
10	■ Aux1	Real	0.0	No remane...	☒	☒	☒		
11	■ Divisor de Tension (V)	Real	0.0	No remane...	☒	☒	☒		
12	■ Aux2	Real	0.0	No remane...	☒	☒	☒		
13	■ Relacion Salida/Entrad...	Real	2.0	No remane...	☒	☒	☒		
14	▼ Temp								
15	■ <Agregar>								
16	▼ Constant								
17	■ <Agregar>								

Figura 73. Interfaz del FB Tensión Vcc y Vout Motor.

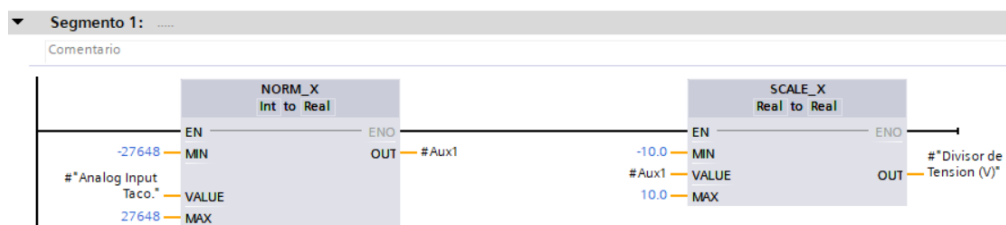


Figura 74. Segmento 1 del FB Tensión Vcc y Vout Motor.

Primero se normaliza y escala la señal analógica que proviene del divisor de tensión entre los valores de ± 10 V.

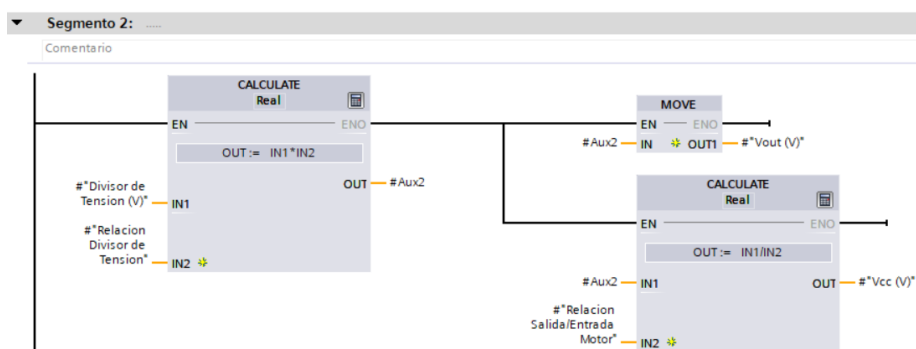


Figura 75. Segmento 2 del FB Tensión Vcc y Vout Motor.

Y en el segmento 2, se calcula la tensión de salida y de alimentación empleando las siguientes expresiones:

Tensión de salida del tacogenerador (V_{out}):

$$V_{out} = Tensión_{divisor} \times Relación_{divisor\ de\ tensión}$$

Siendo;

$Tensión_{divisor}$: la lectura de la entrada analógica del divisor de tensión.

$Relación_{divisor\ de\ tensión}$: la relación que hay entre la tensión real de salida

del motor y la tensión que sale del divisor de tensión (de valor $\frac{24\ V\ (salida\ real)}{10\ V\ (divisor)}$).

Tensión de alimentación del motor (V_{cc}):

$$V_{cc} = \frac{V_{out}}{Relación_{\frac{salida}{entrada}}}$$

Siendo;

V_{out} : tensión real de salida del motor por el tacogenerador.

$Relación_{salida/entrada}$: la relación que hay entre la tensión real de salida

del motor y la tensión de alimentación (tensión de salida es el doble que la entrada).

3.1.1.7 Selector del tipo de control del PID.

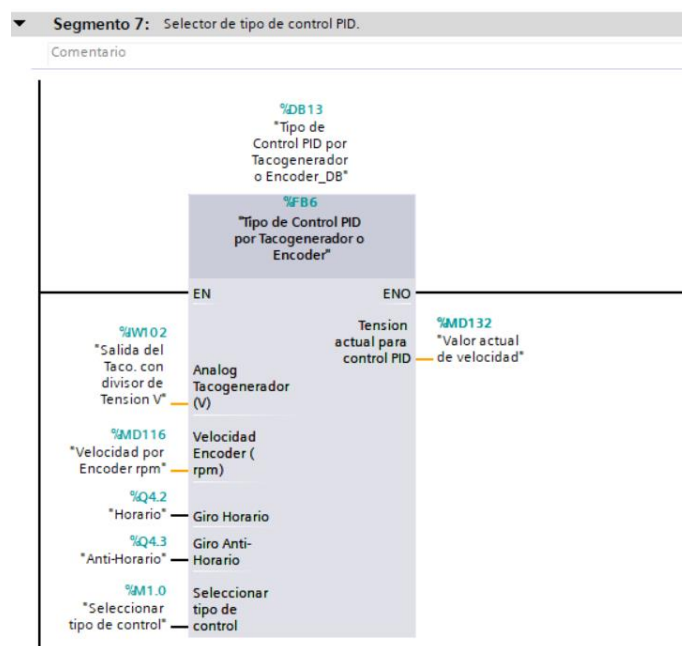


Figura 76. Segmento 7 del Main. Tipo de Control PID por Tacogenerador o Encoder.

En este segmento número 7 se establece cuál de las dos señales de lectura, la señal analógica del tacogenerador o la señal digital del encoder, se usa como una de las entradas del regulador PID para hacer el control de velocidad con dicha señal seleccionada.

Para conseguir esto, dentro de la FB están los siguientes segmentos que se muestran en la figuras de a continuación.

Tipo de Control PID por Tacogenerador o Encoder									
	Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible d...	Escrib...	Visible en ...	Valor de a...	Comentario
1	Input								
2	Analog Tacogenerado...	Word	16#0	No remane...	☒	☒	☒	☐	
3	Velocidad Encoder (rp...	Real	0.0	No remane...	☒	☒	☒	☐	
4	Giro Horario	Bool	false	No remane...	☒	☒	☒	☐	
5	Giro Anti-Horario	Bool	false	No remane...	☒	☒	☒	☐	
6	Seleccionar tipo de co...	Bool	false	No remane...	☒	☒	☒	☐	
7	Output								
8	Tension actual para c...	Real	0.0	No remane...	☒	☒	☒	☐	
9	InOut								
10	<Agregar>								
11	Static								
12	Aux1	Real	0.0	No remane...	☒	☒	☒	☐	
13	Aux1 Valor actual	Real	0.0	No remane...	☒	☒	☒	☐	
14	Aux2	Real	0.0	No remane...	☒	☒	☒	☐	
15	Velocidad Anti-Horaria	Real	0.0	No remane...	☒	☒	☒	☐	
16	Aux2 Valor actual	Real	0.0	No remane...	☒	☒	☒	☐	
17	Resolucion Encoder	Real	30.0	No remane...	☒	☒	☒	☐	
18	Tiempo min/seg	Real	60.0	No remane...	☒	☒	☒	☐	
19	Temp								
20	<Agregar>								
21	Constant								
22	<Agregar>								

Figura 77. Interfaz del FB Tipo de Control PID por Tacogenerador o Encoder.

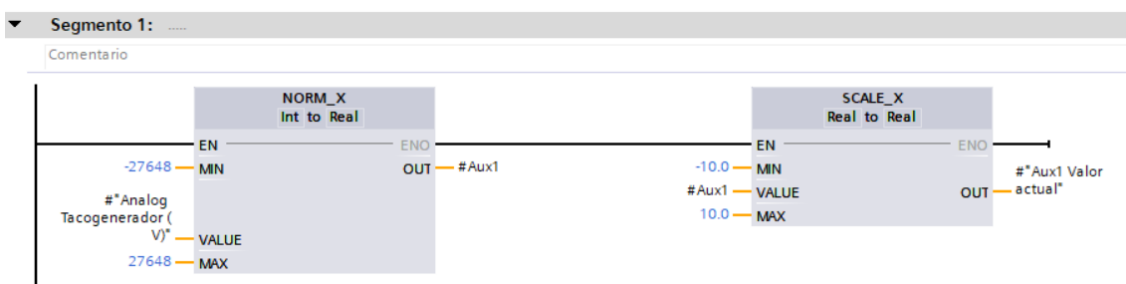


Figura 78. Segmento 1 del FB Tipo de Control PID por Tacogenerador o Encoder.

Con un normalizado y escalado de la tensión analógica del divisor de tensión del tacogenerador se obtiene una de las dos entradas posibles para el regulador PID.

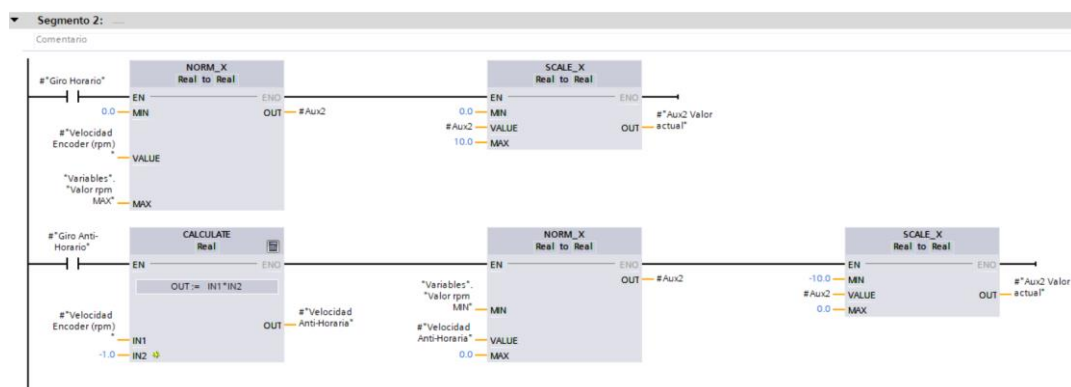


Figura 79. Segmento 2 del FB Tipo de Control PID por Tacogenerador o Encoder.

Como para el caso de la lectura de la señal digital del encoder no se puede distinguir el sentido de giro del motor, debido a que la lectura de pulsos no difiere de un sentido u otro, lo que se hizo fue introducir como entradas al FB las variables de giro que están vinculadas con las salidas digitales Q4.2 y Q4.3. Con esto se podrá determinar si la velocidad de giro es positiva o negativa y con ello saber si la tensión, tras realizar el normalizado y el escalado, es positiva o negativa como se muestra en el Figura 79.

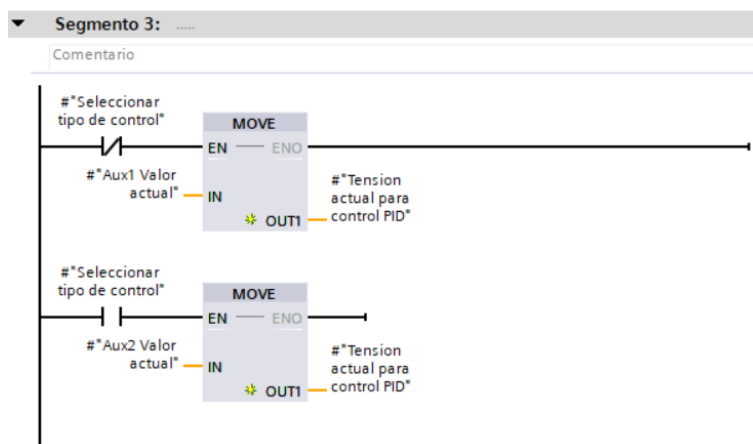


Figura 80. Segmento 3 del FB Tipo de Control PID por Tacogenerador o Encoder.

En el último segmento del FB Tipo de Control PID por Tacogenerador o Encoder mostrado en la Figura 80, se pasará la tensión normalizada y escalada, de una de las dos señales, a la variable de salida del FB. Para determinar cuál es la señal que será la salida del FB, y por tanto una de las entradas del regulador PID, se usa una marca de tamaño bit, que hará lo siguiente: Si tiene el valor de cero se trasladará la señal de tensión que se obtiene del escalado y normalizado de la señal del tacogenerador a la entrada del PID, utilizando la instrucción de categoría transferencia MOVE, pero, si está activada, la señal que se moverá a la entrada del PID será la del encoder.

3.1.1.8 Error de las señales.

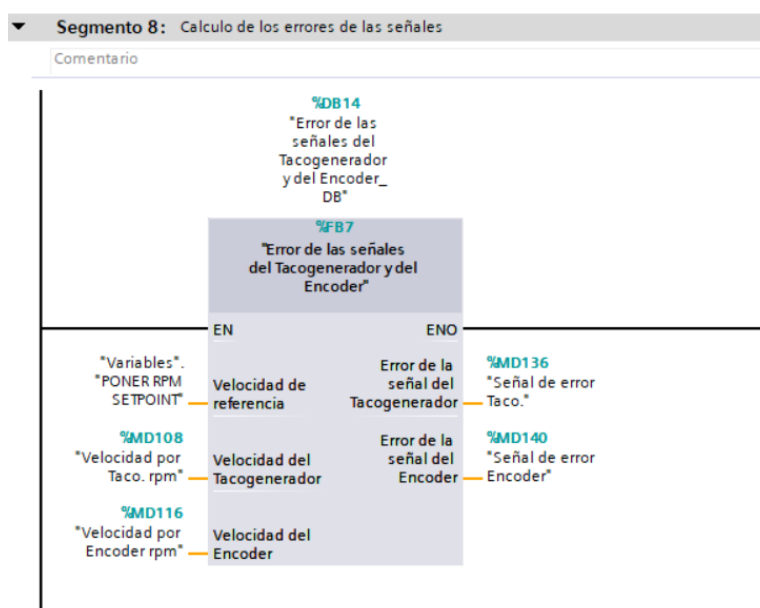


Figura 81. Segmento 8 del Main. Error de las señales del Tacogenerador y del Encoder.

En este segmento se lleva a cabo el cálculo del error que hay entre la velocidad de referencia y las velocidades que se obtienen del tacogenerador y del encoder. Dentro del FB se encuentran los siguientes segmentos que se ven en las figuras sucesivas.

Error de las señales del Tacogenerador y del Encoder									
	Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible d...	Escrib...	Visible en ...	Valor de a...	Comentario
1	Input								
2	Velocidad de referencia	Real	0.0	No rem...	☒	☒	☒	☐	
3	Velocidad del Tacoge...	Real	0.0	No remane...	☒	☒	☒	☐	
4	Velocidad del Encoder	Real	0.0	No remane...	☒	☒	☒	☐	
5	Output								
6	Error de la señal del Ta...	Real	0.0	No remane...	☒	☒	☒	☐	
7	Error de la señal del E...	Real	0.0	No remane...	☒	☒	☒	☐	
8	InOut								
9	<Agregar>								
10	Static								
11	<Agregar>								
12	Temp								
13	<Agregar>								
14	Constant								
15	<Agregar>								

Figura 82. Interfaz del FB Error de las señales del Tacogenerador y del Encoder.

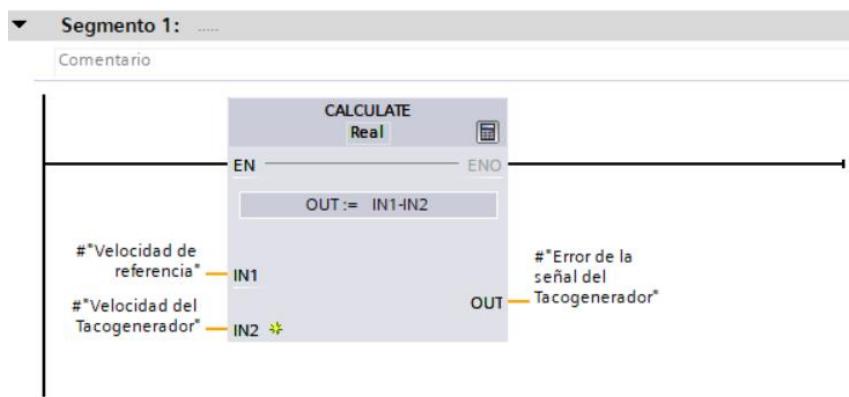


Figura 83. Segmento 1 del FB Error de las señales del Tacogenerador y del Encoder.

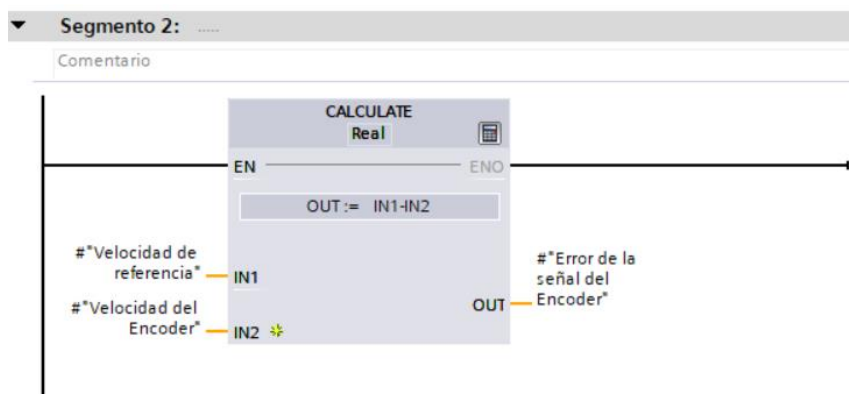


Figura 84. Segmento 2 del FB Error de las señales del Tacogenerador y del Encoder.

En ambos segmentos se realiza la misma operación, pero cambiando la velocidad de las señales del encoder y del tacogenerador.

Error de la señal del Tacogenerador ($Error_{Tacogenerador}$):

$$Error_{Tacogenerador} = V_{referencia} - V_{Tacogenerador}$$

Error de la señal del Encoder ($Error_{Encoder}$):

$$Error_{Encoder} = V_{referencia} - V_{Encoder}$$

3.1.1.9 Establecer parámetros PID.

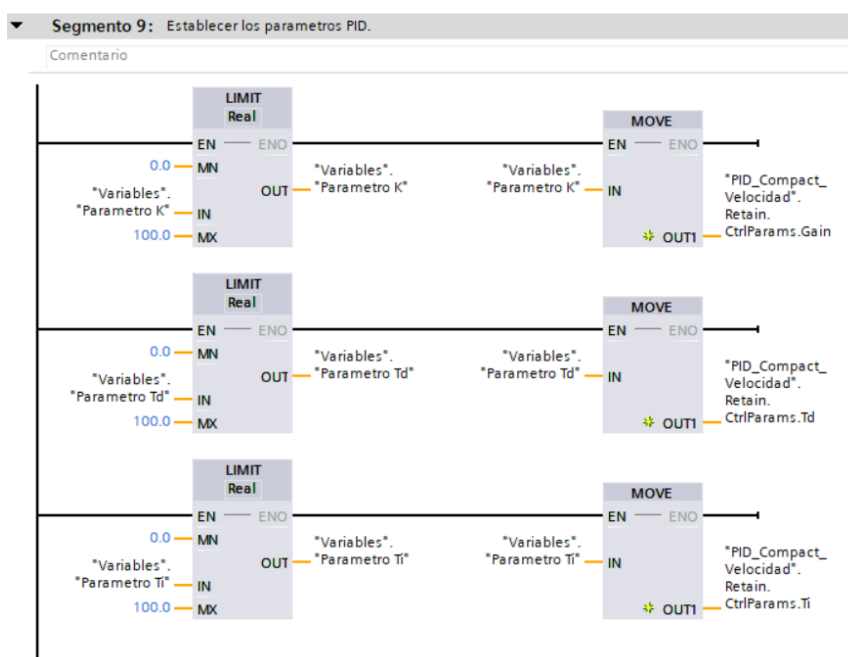


Figura 85. Segmento 9 del Main. Parámetros del PID.

En este segmento no hay ningún FB porque no es necesario. Lo que se está haciendo es limitar los valores que se pueden introducir a las variables de los parámetros del PID, en la pantalla HMI o la página web, y luego son movidos con la instrucción MOVE a los parámetros reales del regulador PID.

3.1.1.10 Conectar con el servidor web.

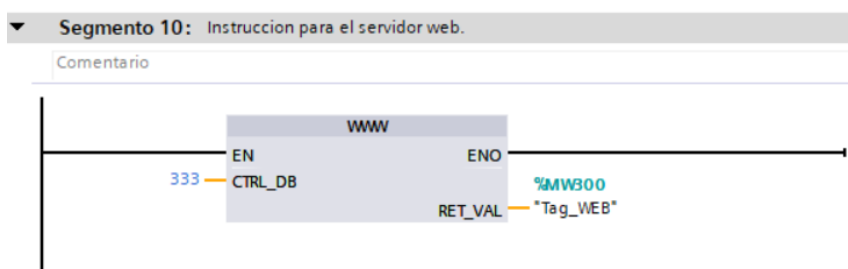


Figura 86. Segmento 10 del Main. Conexión con el servidor web.

En este segmento lo que se hace es conectar las variables del TIA Portal a la página web, pero primero se debe de activar el servidor web del PLC yendo a propiedades y activando la casilla de servidor web.

El servidor web permite administrar el acceso a los usuarios autorizados para vigilar y administrar la CPU a través de una red. Eso permite la evaluación y el diagnóstico a grandes distancias. Para ello solo se necesita un navegador web. Los avisos y la información sobre el estado se muestran en páginas HTML.

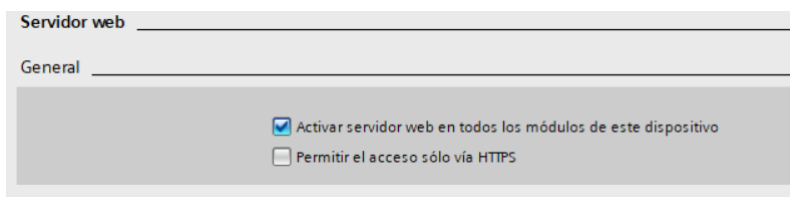


Figura 87. Activar Página web.

Una vez activado se tiene que ir a la pestaña de páginas de usuario y colocar la dirección donde se encuentra el código de la página web así como el archivo de la página web. Una vez hecho se debe de presionar el botón de generar bloques, los cuales, entre otras cosas, tiene la página web codificada.

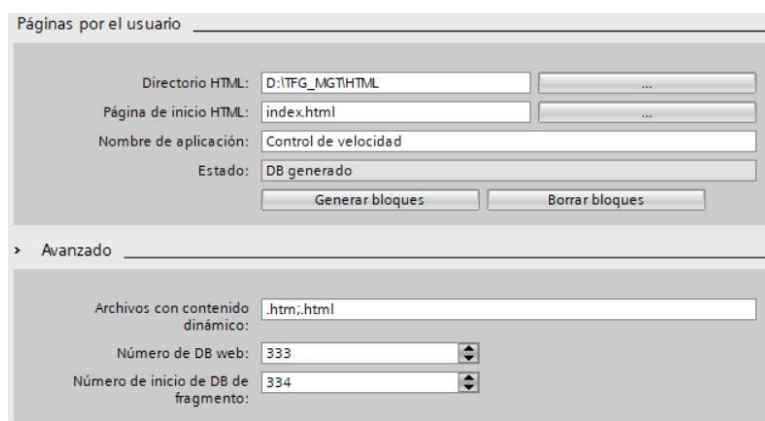


Figura 88. Establecer la página web creada.

Por ello, la instrucción WWW del OB Main tiene como entrada la dirección 333 porque representa el número de DB web. Además, se han añadido una serie de instrucciones que afectan a las variables que se utilizan en el código HTML para que se visualice de la mejor forma posible la página web.

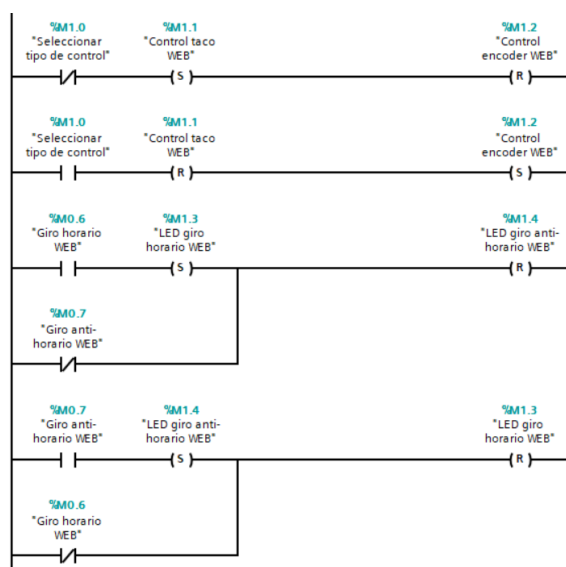


Figura 89. Instrucciones para variables de la página web.

Dependiendo de que se haya seleccionado para el tipo de control, en la página web se visualizará cual es el control que se está llevando a cabo. También, dependiendo de que sentido de giro se haya pulsado se verá dicho sentido de giro activado.

3.1.1.11 Variables de la imagen de avisos para el HMI.

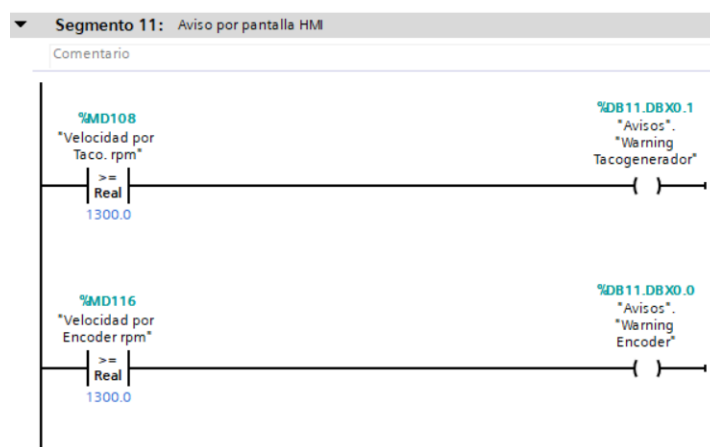


Figura 90. Segmento 11 del Main. Avisos para pantalla HMI.

Estas dos líneas de instrucción son usadas para la imagen de avisos de la pantalla HMI. Funciona del siguiente modo: cuando la velocidad que se está calculando por una de las señales, tanto la del encoder como la de tacogenerador, supere la velocidad de 1 300 r.p.m., se activará la variable tipo bit warning tacogenerador o warning encoder, dependiendo de cual supera dicha velocidad. Estas dos variables se encuentran alojadas en el DB creado y nombrado con el nombre de Avisos, en el cual se tuvo que hacer unos cambios en las propiedades para poder usarse en la ventana de avisos del HMI.

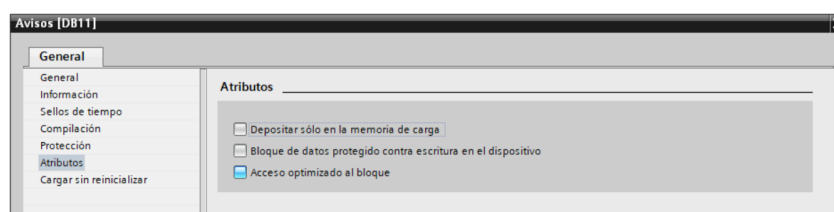


Figura 91. Cambios en las propiedades del DB Avisos.

Se debe de desactivar la casilla de Acceso optimizado al bloque como se muestra en la Figura 91. Esto se hace para que en el DB aparezca la columna Offset, la cual indica la dirección relativa de la variable en el bloque que usará la ventana de avisos del HMI para diferenciar cuál de los dos warnings ha saltado e indicarlo por pantalla.

Avisos											
	Nombre	Tipo de datos	Offset	Valor de arranq...	Remanen...	Accesible d...	Escrib...	Visible en ..	Valor de a..	Comentario	
1	Static				☐	☐	☐	☐	☐		
2	Warning Encoder	Bool	0.0	false	☐	☒	☒	☒	☐		
3	Warning Tacogenerador	Bool	0.1	false	☐	☒	☒	☒	☐		

Figura 92. Variables del DB Avisos.

3.4 CYCLIC INTERRUPT.

Al igual que el Main, el bloque de programa Cyclic Interrupto es un OB. Se trata de un OB [14] de alarma cíclica, las cuales sirven para iniciar programas en intervalos periódicos, independientemente de la ejecución cíclica del programa. Los intervalos se pueden definir por el usuario cuando se agrega el bloque o en sus propiedades.

3.4.1 Partes de un Regulador PID.

Antes de explicar cómo funciona y los ajustes que se han hecho en el OB Cyclic Interrupt se va a explicar que es un regulador PID.

Un controlador o regulador PID, cuyas siglas en español significan: Proporcional, Integrador y Derivativo, es un controlador de realimentación cuya función es mitigar el error que hay entre la señal de referencia y la señal de salida.

La forma en la que el regulador genera la señal de control se conoce como acción de control. Existen varias acciones de control, de las cuales el PID combina y aprovecha sus características, teniendo sus ventajas y desventajas. Como ventaja tiene que la respuesta es más rápida y tiene una compensación de la señal de error muy inmediata. Como desventaja tiene que los ajustes son más difíciles de regular, ya que al variar un parámetro, los efectos que tiene este en el sistema varían dependiendo de los otros valores de los demás parámetros, que se tendrán que ajustar para conseguir la regulación correcta del sistema. [15]

La ecuación que acatan los reguladores PID es la siguiente:

$$u(t) = K_p \times e(t) + \frac{K_p}{T_i} \times \int_0^t e(t) dt + K_p \times T_d \times \frac{de(t)}{dt}$$

o en función de transferencia

$$\frac{U(s)}{E(s)} = K_p \times \left(1 + \frac{1}{T_i \times s} + T_d \times s \right) = K_p + \frac{K_i}{s} + K_d \times s$$

3.4.1.1 Acción de control Proporcional.

Este tipo de acción de control es proporcional a la señal de error del sistema. Interiormente, la señal de error es multiplicada por una constante K_p , la cual cuanto más grande sea su valor mayor será su error, y viceversa. [15]

Es el más simple de los tipos de acción de control, ya que consiste en un amplificador que incrementa la señal de error antes de entrar a la planta.

Los efectos que se tiene al aumentar la acción proporcional K_p son:

- Aumento de la velocidad de respuesta del sistema.
- Disminución del error en régimen permanente.
- Aumento de la inestabilidad del sistema.

De los tres efectos, el ultimo es el indeseable y por ello se debe tener cuidado en el aumento de la K_p . Hay que buscar un equilibrio entre conseguir una respuesta rápida con un error mínimo y que el sistema no sea inestable.

3.4.1.2 Acción de control Integral.

En esta acción de control se calcula la integral de la señal de error. A medida que el tiempo va pasando los errores se van a ir sumando para ir aumentando la acción integral, consiguiendo reducir el error en régimen permanente. [15]

Al aumentar la acción integral K_i se obtiene los siguientes efectos:

- Disminución del error en régimen permanente.
- Aumento de la inestabilidad del sistema.
- Aumento, muy poco, de la velocidad de respuesta.

Sin embargo, el regulador de TIA Portal usa el parámetro T_i , por tanto los efectos anteriores se obtienen al disminuir el parámetro T_i , por ser $K_i = K_p/T_i$.

Esta acción se suele usar para disminuir el error en régimen permanente quitando velocidad y estabilidad al sistema.

3.4.1.3 Acción de control Derivativa.

Esta acción de control es proporcional a la derivada de la señal de error. Se la suele denominar acción de control de velocidad porque T_d es el intervalo de tiempo en el que la acción de control derivativa se adelanta al efecto de la acción de control proporcional. Por ello se debe de tener cuidado con los valores de este parámetro, ya que si el sistema se mueve a velocidades grandes hacia el valor de referencia, la respuesta del sistema se pasará por la inercia, produciendo un sobre impulso y oscilaciones en la respuesta. [15]

Al aumentar la acción derivativa K_d o el parámetro T_d que utiliza en el TIA Portal, que viene a ser $K_d = K_p \times T_d$, se obtiene los siguientes efectos:

- Aumento de la estabilidad del sistema.
- Disminución, muy poco, de la velocidad de respuesta.
- El error en régimen permanente no se ve afectado.

Esta acción se usa para estabilizar la respuesta quitando velocidad al sistema.

3.4.2 PID_Compact.

Una vez explicado que es un regulador PID y como afecta las modificaciones de sus parámetros a la respuesta del sistema toca comentar el regulador PID del TIA Portal.

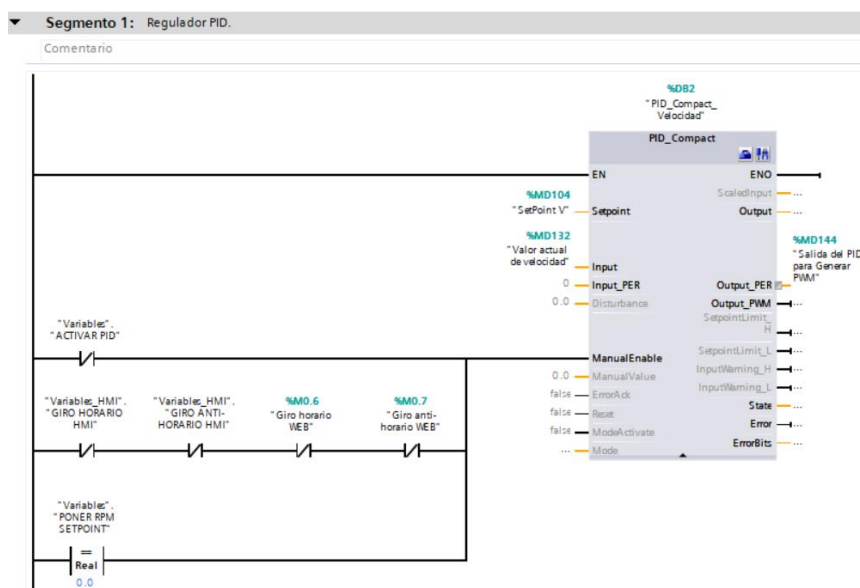


Figura 93. Segmento 1 del Cyclic Interrupt. PID_Compact.

Como se observa en la Figura 93, el regulador PID que se utiliza es una instrucción de la clase Tecnología que se encuentra en la carpeta PID Control, Compact PID. A parte de este regulador hay otros dos más que se comentarán, al igual que el que se utiliza, explicando sus funciones y en que ámbito se utilizan normalmente cada uno:

- PID_3Step (Versión 2.3): Este objeto tecnológico proporciona un regulador PID con optimización para válvulas o actuadores con comportamiento integrador. [14] Con él es posible configurar los siguientes reguladores:
 - Regulador paso a paso de tres puntos con realimentación de posición.
 - Regulador paso a paso de tres puntos sin realimentación de posición.
 - Regulador de válvula con valor de salida analógico.
- PID_Temp (Versión 1.1): ofrece un regulador PID continuo con optimización integrada. Está diseñado especialmente para regular la temperatura y es apto para aplicaciones de calefacción o refrigeración. Para ello se dispone de dos salidas, una para calefacción y otra para refrigeración. [14]
- PID_Compact (Versión 2.3): ofrece un regulador PID continuo con optimización integrada. Asimismo, también es posible configurar un regulador de impulsos. Es posible elegir entre el modo manual y el automático. [14]

En el proyecto se usa el PID_Compact por ser un regulador más universal, no como los otros dos que están diseñados específicamente para ciertos casos.

3.4.2.1 Configuración PID_Compact.

Una vez se tiene el PID_Compact en el segmento, para poder configurar los ajustes se tiene que pinchar en el icono que se muestra en la Figura 94.

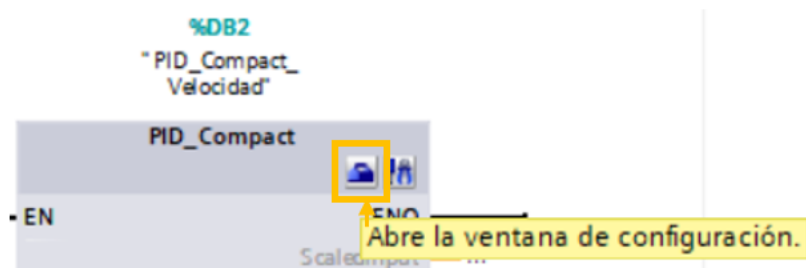


Figura 94. Configuración del PID_Compact.

Dentro de la configuración se encuentra una vista general de los ajustes que se le puede modificar al regulador, entre los que nos encontramos con:

Ajustes básicos.

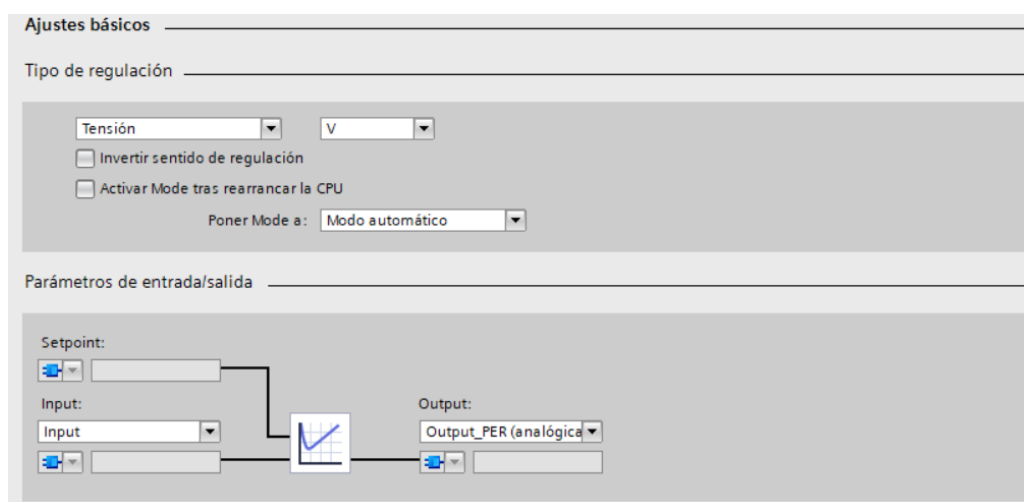


Figura 95. Ajustes básicos del PID_Compact.

En estos ajustes se indica el tipo de regulación que se quiera realizar, en el caso del proyecto, como ya se ha mencionado, se hará una regulación de tensión, en voltios.

Los parámetros de entrada del PID_Compact son, como se ve en la Figura 93, el SetPoint, es decir, el valor de referencia que teclea el usuario, pero normalizado y escalado entre ± 10 V, y el valor de una de las señales que se leen, la señal analógica del tacogenerador o bien la señal digital del encoder, dependiendo de cual se seleccione, pero ambas están normalizadas y escaladas entre ± 10 V.

La salida del PID_Compact puede ser de tres valores:

- Output_PER: utiliza una salida analógica como valor de salida.
- Output: utiliza una variable del programa como valor de salida.
- Output_PWM: utiliza una salida digital y la controla mediante PWM. El valor de salida se obtiene mediante tiempos de conexión y desconexión de la variable.

La salida del PID_Compact tiene que alimentar a la entrada del motor DC, y como se ha mencionado varias veces, la alimentación del motor DC se lleva a cabo mediante una señal PWM, por lo tanto la salida del PID_Compact deberá de ser del valor Output_PWM. Sin embargo, tras la puesta en marcha con esta salida, se observó que el control de velocidad no se realizaba correctamente y se decidió cambiar la salida por el valor Output_PER. Esta salida de PLC se utilizará como una entrada del FB Generador PWM, para crear la señal PWM que se ha explicada en el subapartado 3.1.1.3 Generador PWM.

Ajustes del valor real.

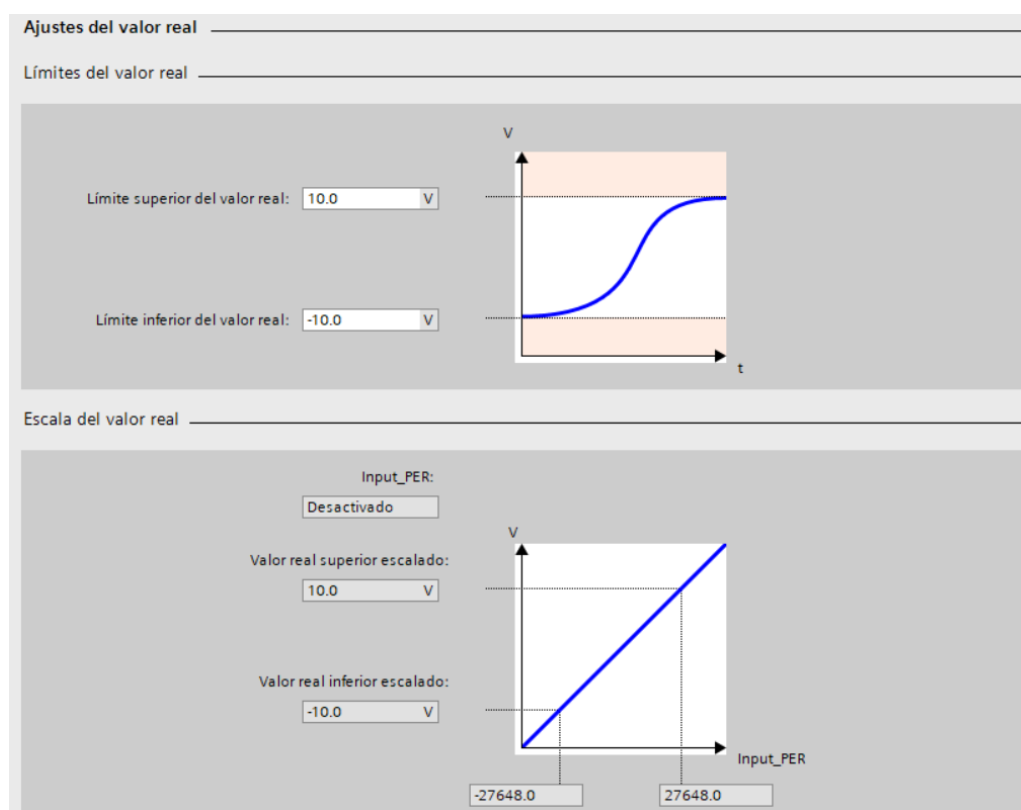


Figura 96. Ajustes del valor real del PID_Compact

En esta sección se determina cuáles son los límites del valor real, que como se lleva diciendo, los valores están comprendidos entre ± 10 V. El escaldado se encuentra desactivado debido a que hace referencia al valor de la entrada de una de las dos lecturas del motor, es decir, la señal analógica del tacogenerador o la señal digital del encoder, que ya están escaladas.

Ajustes avanzados.

En los ajustes avanzados solamente se modificó los límites del valor de salida, que por defecto viene como límite inferior 0 %, pero para el control es necesario que se den valores negativos para poder determinar el sentido de giro del motor DC.

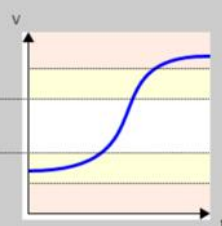
En el apartado de los parámetros PID son los que vienen por defecto, sin embargo, para poder ser modificables se necesita activar la casilla “Activar entrada manual”, para que durante la ejecución del programa los valores que se les den a los parámetros del PID pueden ser modificados y ver la respuesta del sistema con esos parámetros cambiados.

Ajustes avanzados

Monitorización del valor real

Lim. sup. advertencia: 3.402822E+38 V

Lim. inf. advertencia: -3.402822E+38 V



Limitaciones PWM

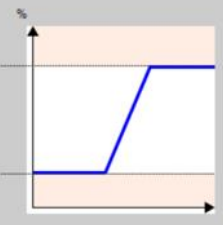
Tiempo conexión mín: 0.0 s

Tiempo desconex. mín: 0.0 s

Límites del valor de salida

Lim. sup. valor de salida: 100.0 %

Lim. inf. valor de salida: -100.0 %



Comportamiento en caso de error

Poner Output a: Valor de salida sustitutivo mientras dure el error

Valor de salida sustitutivo: 0.0 %

Parámetros PID

☒ Activar entrada manual

Ganancia proporcional: 1.9725

Tiempo de integración: 0.491438 s

Tiempo derivativo: 0.1258 s

Coefficiente retardo derivativo: 0.1

Ponderación de la acción P: 1.0

Ponderación de la acción D: 1.0

Tiempo muestreo algoritmo PID: 0.1 s

Regla para la optimización

Estructura del regulador: PID

Figura 97. Ajustes avanzados del PID_Compact.

3.4.2.2 Activación del Regulado PID_Compact.

Ya se ha explicado cómo funciona el regulador PID_Compact, que hace lo mismo que un regulado PID, mitigar el error que hay entre la señal de referencia y la señal del motor DC. Lo que falta por explicar de este OB son las instrucciones que están en la entrada ManualEnable del PID_Compact.

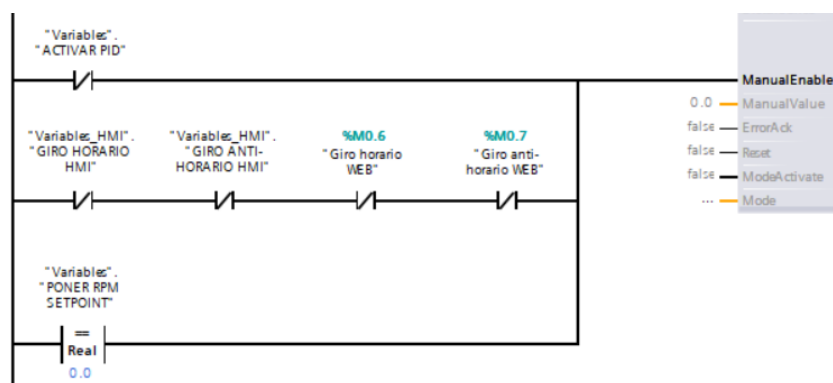


Figura 98. Instrucciones para desactivar el PID_Compact.

La finalidad de estas instrucciones es reiniciar a cero el valor de salida del PID por tres razones:

1. Desactivar el PID si se ha parado el sistema.
2. Poner a cero la salida si no se ha establecido el sentido de giro del motor.
3. Si el valor de la velocidad de referencia es de 0 r.p.m. la salida del PID debe ser de 0.

La primera de las tres razones es obvia, si el sistema no se arranca no tiene sentido que el PID este dando ningún valor de salida, o si el sistema se para cuando esté en funcionamiento, la salida del PID debe de ser nula.

La segunda y tercera razón están conectadas. Se pusieron estas dos condiciones porque durante la prueba de funcionamiento del control de velocidad del motor se observó que cuando el motor se encontraba girando a una velocidad y se le daba la orden de que no girase en ninguno de los sentido, el motor sí dejaba de girar, sin embargo, la salida del PID permanecía con el último valor que quedaba cuando estaba girando, y cuando se le volvía a dar un sentido de giro, el motor empezaba a girar a la máxima velocidad y el valor de salida del PID llegaba a su valor máximo. Lo mismo sucedía con un cambio brusco de velocidad de referencia, si se le daba un valor de 800 r.p.m., por ejemplo, y luego se le daba el valor de 0 r.p.m., la salida del PID se queda con el valor de las 800 r.p.m. y cuando se le volvía a dar otra velocidad, la salida del PID daba su valor máximo y el motor empezaba a girar a su máxima velocidad. Por estas razones, cuando se den estos dos casos, el valor de salida del PID se reiniciará.

3.5 PANTALLA HMI.

Para agregar una pantalla HMI se tiene que hacer el mismo proceso que cuando se añadió el controlador y los módulos añadidos, pero esta vez en la sección de HMI y escogiendo la pantalla que se desea. Para el proyecto, se escogió una pantalla de 12 pulgadas, pero se podía haber escogido otra cualquiera, ya que en esta parte no influye una pantalla u otra por la simple razón de que se va a simular dicha pantalla.

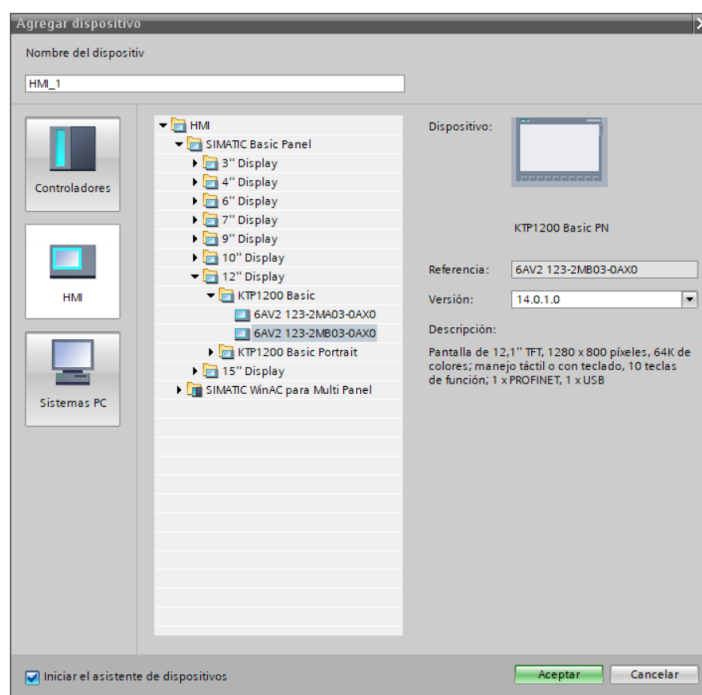


Figura 99. Agregar pantalla HMI.

Ahora queda hablar de las distintas imágenes de pantalla que tiene la pantalla HMI. Con el fin de obtener un buen resultado estético las imágenes que se utilizan en la pantalla HMI se han extraído de las referencias [17] y [18].

3.5.1 Plantilla.



Figura 100. Plantilla HMI.

La plantilla viene a ser el fondo que van a tener todas las imágenes del HMI. En ella se encuentra un botón, el cual al interactuar con él pulsándolo la imagen de la pantalla cambiará a la imagen Menú por tener un evento al ser soltado.

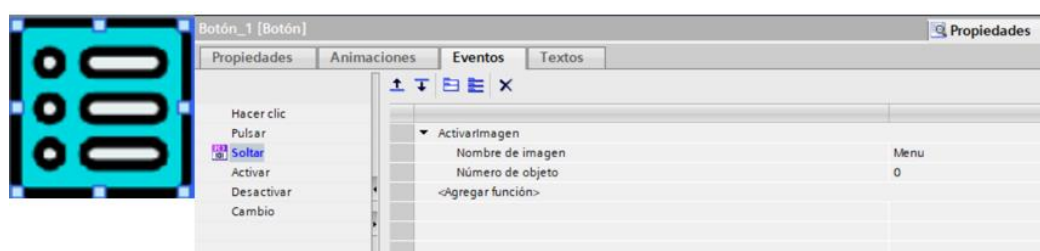


Figura 101. Botón para ir a la imagen Menú del HMI.

3.5.2 Inicio de Sesión.

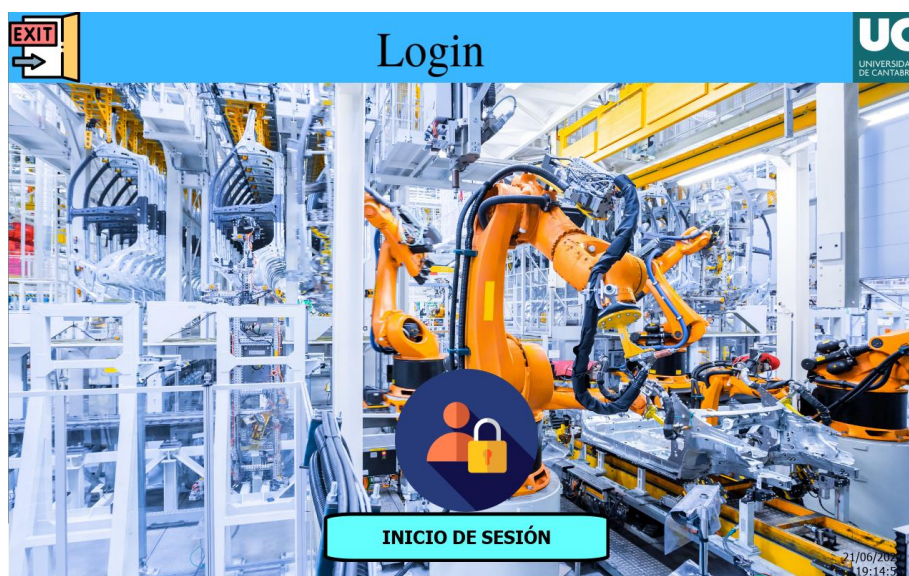


Figura 102. Imagen Inicio de Sesión del HMI.

La simulación de pantalla HMI viene con un registro de usuario, con el cual solo podrán acceder al control de velocidad del motor aquellos que estén registrados con un usuario y contraseña.

Para administrar a los usuario se tiene que ir a la pestaña de “Administración de usuarios” que se encuentra en la última fila del dispositivo HMI de la ventana de la izquierda. Una vez dentro se podrá agregar, dar grupos y permisos a los usuarios.

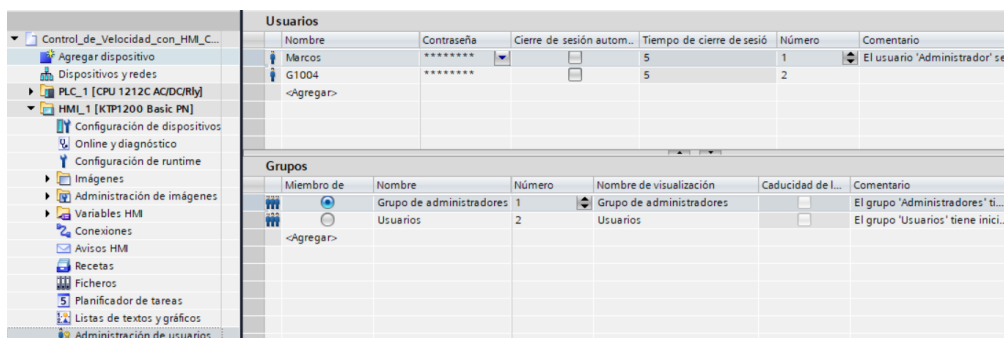


Figura 103. Agregar usuarios y grupos.

Grupos					
	Nombre	Número	Nombre de visualización	Caducidad de l...	Comentario
	Grupo de administradores	1	Grupo de administradores	☐	El grupo 'Administradores' ti...
	Usuarios	2	Usuarios	☐	El grupo 'Usuarios' tiene inici...
	<Agregar>				
Permisos					
	Activo	Nombre	Nombre de visualización	Número	Comentario
	☒	Administración de usuari...	Administración de usuarios	1	Permiso 'Administración de ...
	☒	Monitorización	Monitorización	2	Permiso 'Supervisar'.
	☒	Operación	Operación	3	Permiso 'Operación'.
	☒	Inicio de Sesion	Inicio de Sesion	4	

Figura 104. Grupo de Administradores y sus permisos.

Grupos					
	Nombre	Número	Nombre de visualización	Caducidad de l...	Comentario
	Grupo de administradores	1	Grupo de administradores	☐	El grupo 'Administradores' ti...
	Usuarios	2	Usuarios	☐	El grupo 'Usuarios' tiene inici...
	<Agregar>				
Permisos					
	Activo	Nombre	Nombre de visualización	Número	Comentario
	☐	Administración de usuari...	Administración de usuarios	1	Permiso 'Administración de ...
	☐	Monitorización	Monitorización	2	Permiso 'Supervisar'.
	☒	Operación	Operación	3	Permiso 'Operación'.
	☒	Inicio de Sesion	Inicio de Sesion	4	

Figura 105. Grupo de Usuarios y sus permisos.

Para iniciar sesión se tiene que pulsar el botón que pone “Inicio de sesión” y saldrá una ventana en donde se debe de introducir el usuario y contraseña. Una vez introducido se dará a aceptar y se volverá a pulsar de nuevo el botón de “Inicio de sesión”. Una vez hecho esto la imagen cambiará de la imagen de Inicio de Sesión a la imagen de Menú.

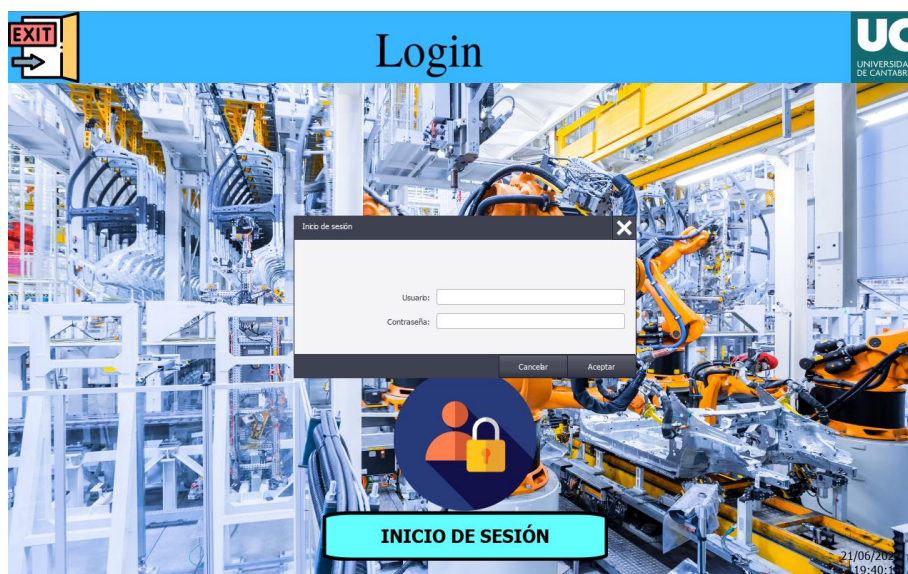


Figura 106. Ventana de Inicio de Sesión.

Además, en esta imagen se encuentra un botón para cerrar y salir de la pantalla HMI.

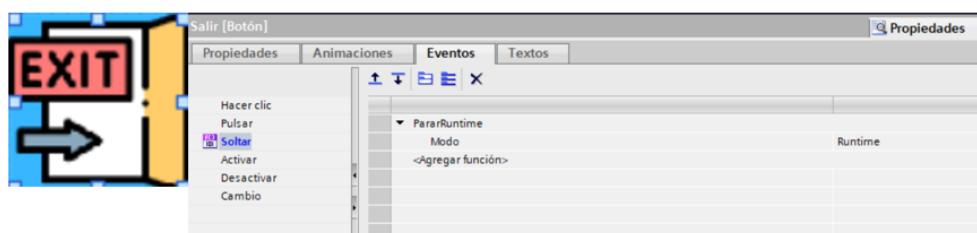


Figura 107. Botón para salir del HMI.

3.5.3 Menú.

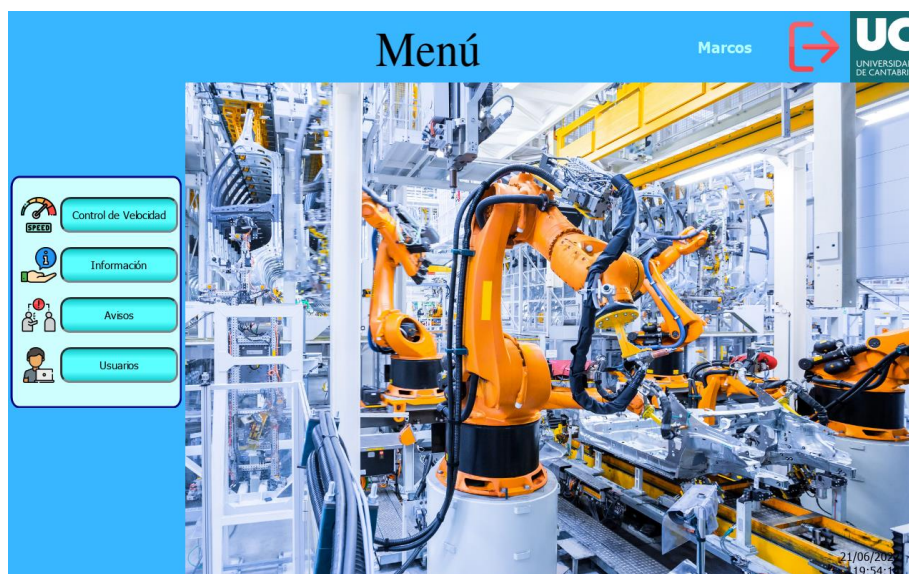


Figura 108. Imagen Menú del HMI.

A esta imagen se llega una vez se ha iniciado sesión. En la parte superior de la derecha se puede observar el nombre del usuario con el que se ha iniciado sesión, esto se consigue de este modo:



Figura 109. Nombre del usuario en la imagen Menú.

Se tiene que crear una variable de formato cadena de caracteres (WString), de la longitud que se desee, para el caso se escogió de 18 de longitud de campo para que sea lo suficientemente largo y no se quede corto por tener un nombre de usuario demasiado largo.

Después se tiene que crear una tarea en la sección de “Planificación de tareas”, que se encuentra en la ventana de la izquierda es el dispositivo HMI. Una vez dentro se debe de

crear la tarea que muestre el nombre del usuario con el que se ha iniciado sesión, dándole un evento de “LeerNombreDeUsuario” y asociando con la variable WString que se creó.

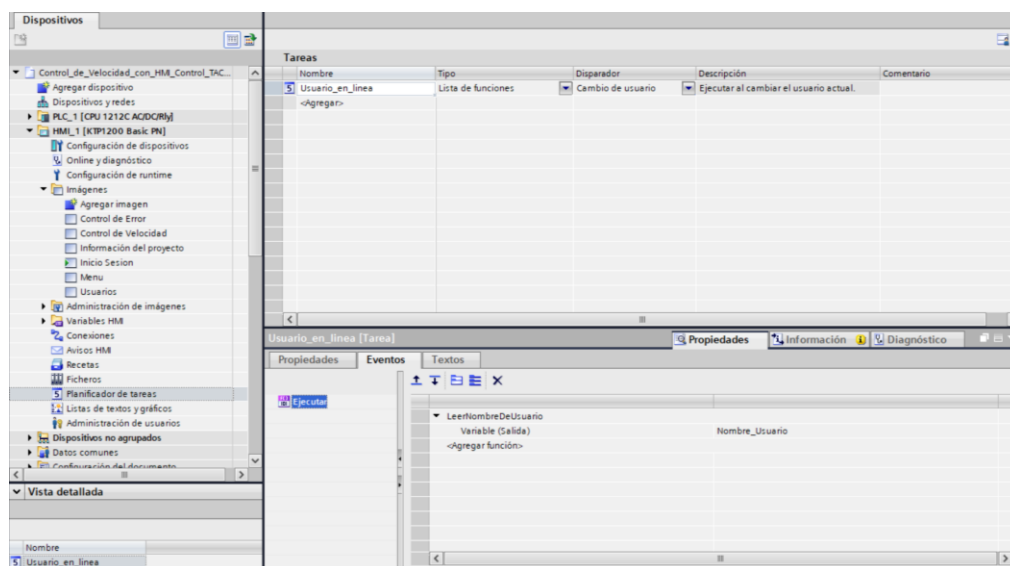


Figura 110. Agregar tarea para visualizar el nombre de usuario.

Al lado del nombre de usuario se encuentra un botón con el cual se cierra la sesión y se vuelve a la imagen de Inicio de Sesión.

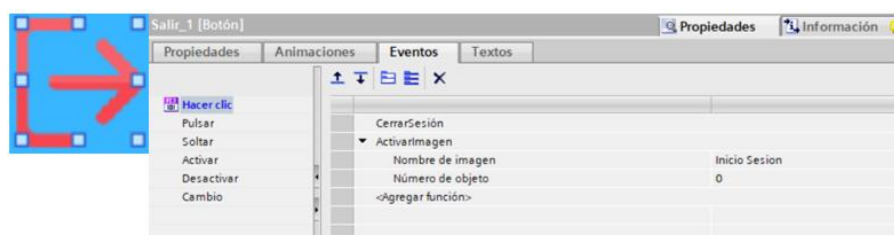


Figura 111. Botón para cerrar sesión.

En la parte izquierda se encuentran varios botones, con los cuales se pueden ir a distintas imágenes. Cada botón tiene asociado el nombre de la imagen a la que te lleva.

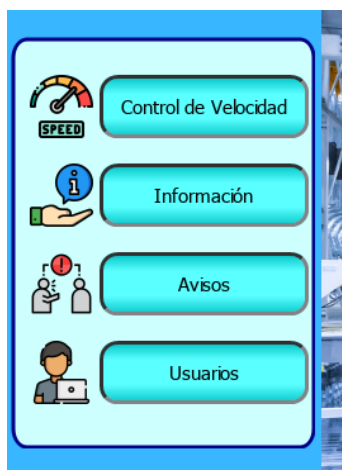


Figura 112. Botones para las distintas imágenes del HMI.

3.5.4 Control de velocidad.

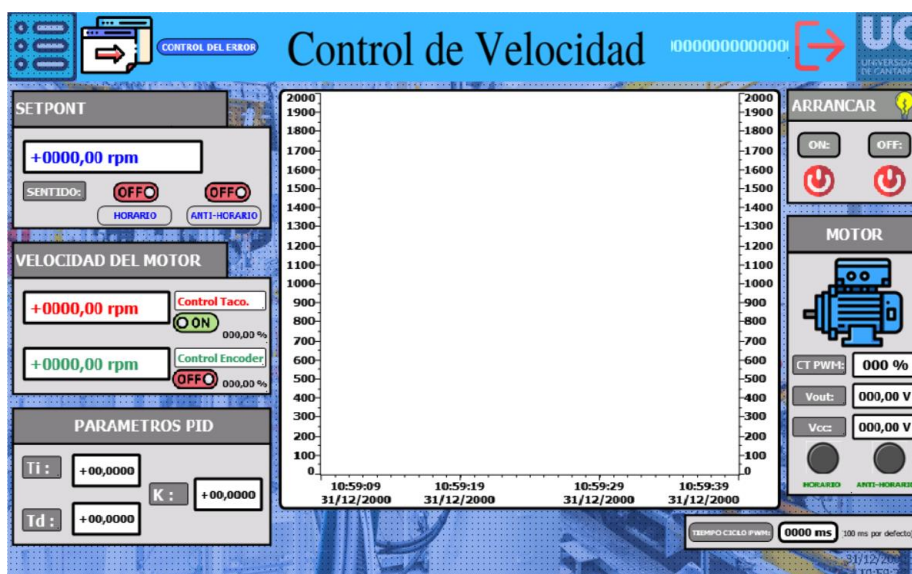


Figura 113. Imagen Control de Velocidad del HMI.

En esta imagen, como su nombre indica, se lleva a cabo el control de velocidad. En ella se puede establecer la velocidad de giro del motor, introduciéndola por teclado en el cuadro donde viene el nombre de SetPoint, que está asociada con la variable “Variables”. “PONER RPM SETPOINT” del PLC. Además se puede establecer, en el mismo cuadro, el sentido de giro con los botones Horario y Anti-Horario, asociados a las variables “Variables_HMI”. “GIRO HORARIO HMI” y “Variables_HMI”. “GIRO ANTI-HORARIO HMI” respectivamente. Se puede volver a la imagen Menú pulsando el botón que ya se ha comentado en el apartado 3.3.1 Plantilla.

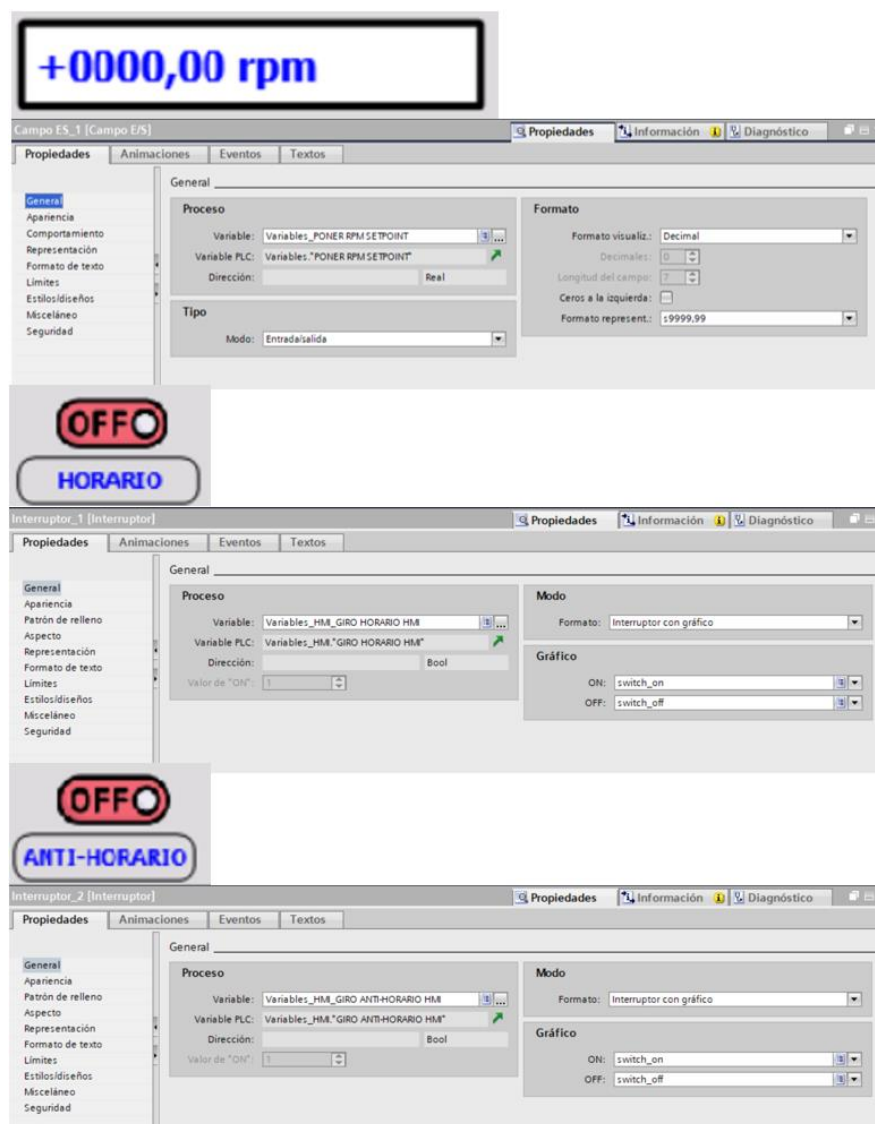


Figura 114. Cuadro de escritura para velocidad de referencia y botones para el sentido de giro.

Otra cosa que se puede hacer en esta imagen es observar las señales de lectura del motor, es decir, la señal analógica del tacogenerador y la señal digital del encoder, pero en las unidades de r.p.m. para que se puede apreciar cual es la velocidad de giro por esas dos señales y poder comparar con el valor de referencia. De igual forma está la opción de establecer el tipo de control de velocidad, si con la señal del tacogenerador o con la del encoder, con unos botones para visualizar que tipo de control se está ejecutando. Para saber cuál es el tipo de control que se está utilizando se juega con una única variable, cuando tenga el valor de 1 significará que está con la señal de control del encoder, por tanto el interruptor estará en ON. Para visualizar que el control tacogenerador está en OFF lo que se hizo fue invertir las imágenes de ON y OFF para que se visualicen la imagen contraria al otro tipo de control. Esto último se puede observar en la sección grafico de los botones de la imagen 115.

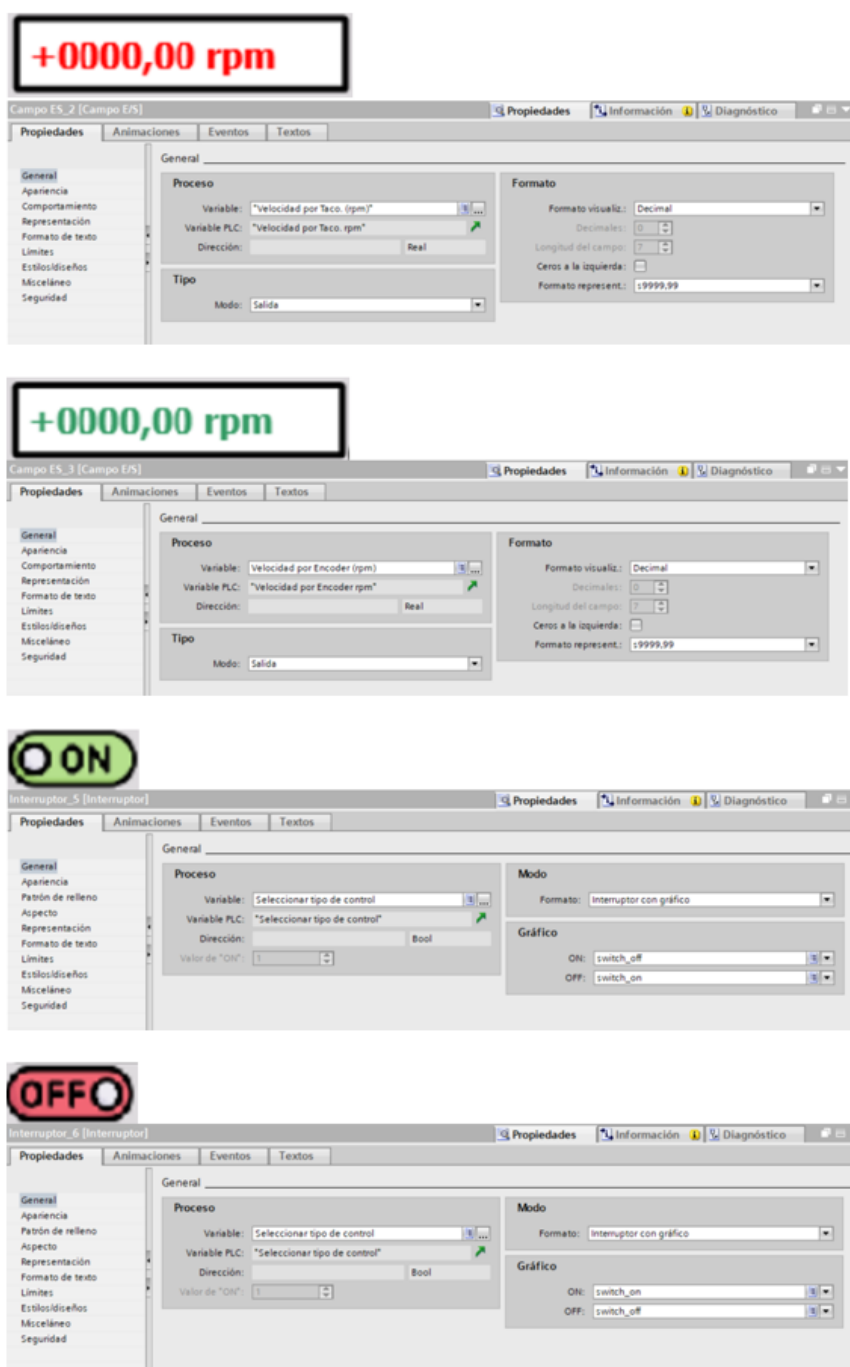


Figura 115. Cuadros de lectura de las velocidades y botones del tipo de control.

También se puede modificar el valor de los parámetros del PID escribiendo un valor con el teclado. Como se explicó en el subapartado 3.1.1.9 Establecer parámetros PID, estos valores se encuentran limitados y no pueden ser ni inferiores ni superiores a los límites establecidos. Si se escribe un número que esté fuera de los límites, el valor que se le da a ese parámetro será el del límite más próximo a ese valor escrito.

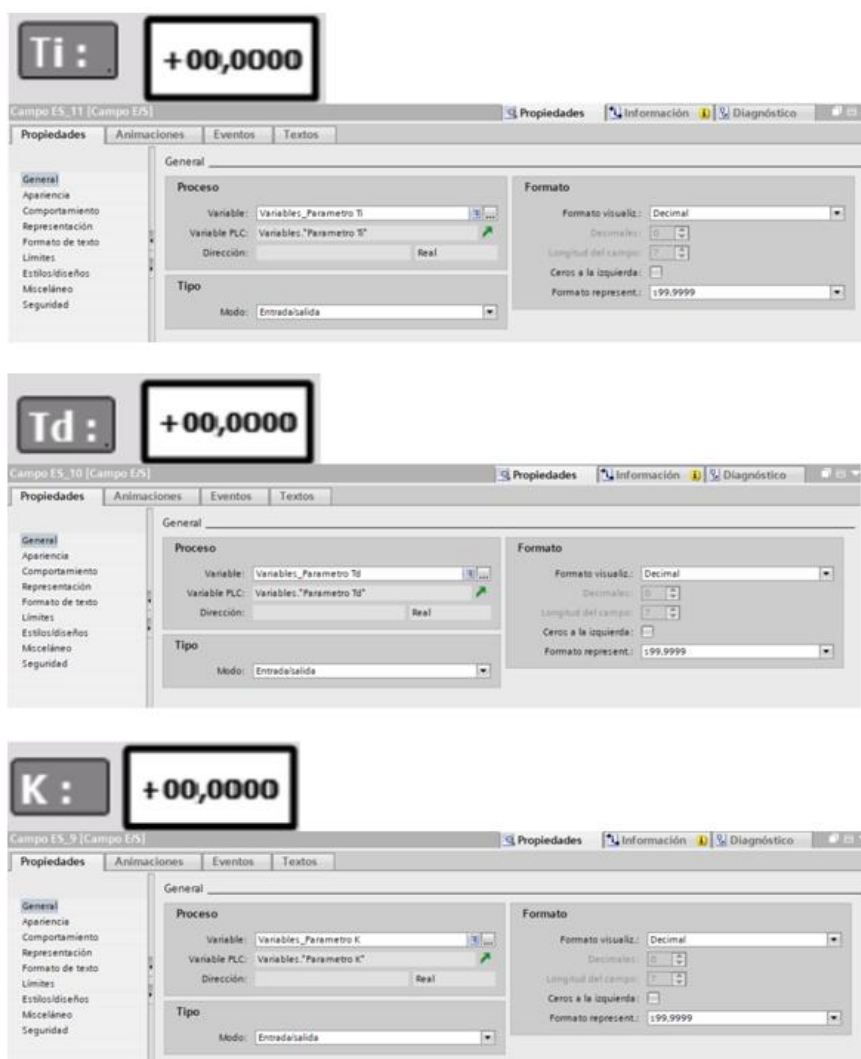


Figura 116. Cuadros de escritura para dar valores a los parámetros PID.

En esta imagen Control de Velocidad se puede observar la evolución de las respuesta de forma visual a través de un visor gráfico. Este visor grafico de la Figura 117 se saca de la ventana de la derecha en Herramientas de Control. Las curvas que se muestran son la velocidad de referencia con la variable “Variables.”PONER RPM SETPOINT”, de color azul, la variable de velocidad “Velocidad por Taco. (rpm)”, de color rojo, y la variable velocidad “Velocidad por Encoder (rpm)”, de color verde. Además se puso los límites de los ejes y las unidades que representan.

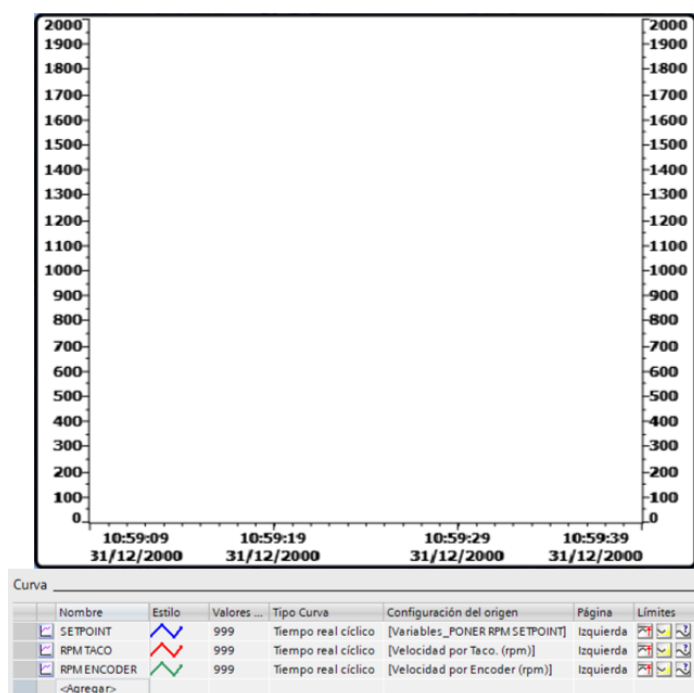


Figura 117. Visor gráfico de las señales de velocidad.

También en esta imagen se activa y desactiva el PID con los botones que se encuentran el cuadro de arriba a la derecha vinculado a las variables “Variables_HMI”. “Arrancar HMI” y “Variables_HMI”. “Parar HMI”. Si se arranca el sistema, tanto la imagen de la bombilla como el botón ON de la Figura 113 se iluminarán, y si se desactiva entonces la bombilla se apagará y el botón de OFF se encenderá.

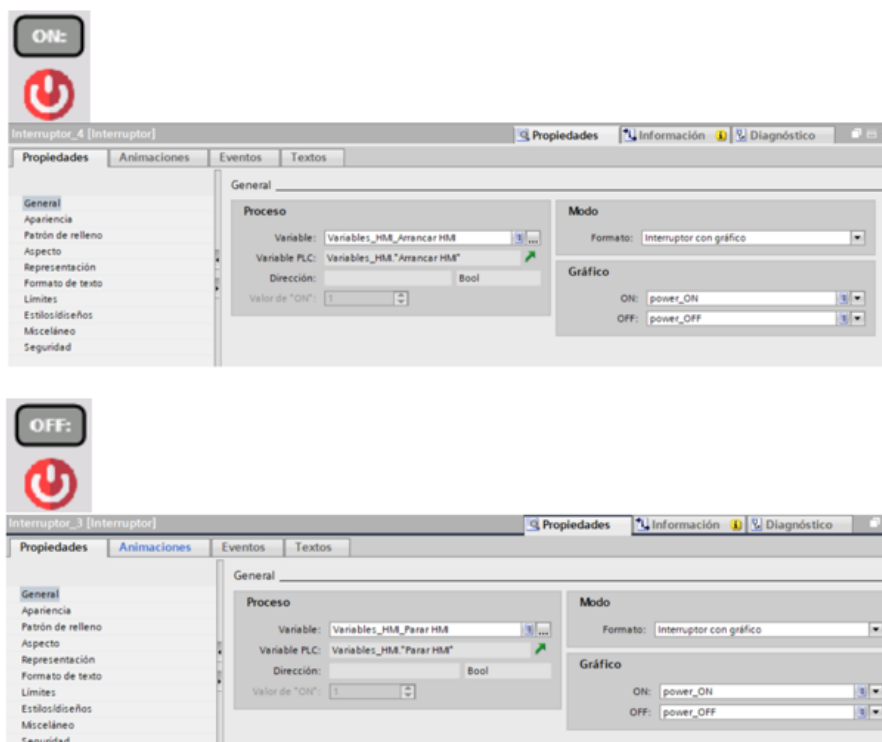


Figura 118. Botones de arranque y paro del sistema.

Como se puede observar en el cuadro situado en el lateral derecho de la Figura 113, se visualizan tanto el ciclo de trabajo del PWM como las tensiones de salida y de alimentación del motor, al igual que se verifica si el sentido de giro es el mismo al que se ha establecido, pudiéndolo ver si se pone de color verde el LED asociado al sentido de giro.



Figura 119. Cuadros de lectura del estado del motor.

La última modificación que se puede realizar en esta imagen es el poder establecer que tiempo de ciclo se quiere tener para la señal del PWM, Figura 120.

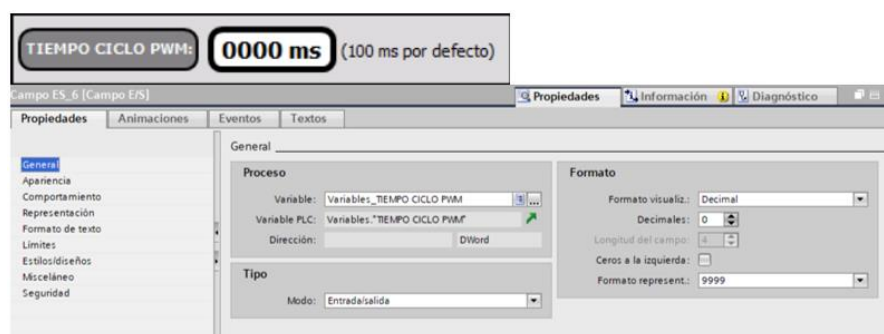


Figura 120. Cuadro de escritura del tiempo de ciclo PWM.

Además de poder manipular la velocidad de giro y los parámetros del PID para ver la evolución de las respuestas del tacogenerador y encoder se puede ver el error que tienen estas entradas respecto a la velocidad de referencia.

Para poder acceder a la imagen Control del Error se tiene que pulsar el botón que se encuentra justo al lado del botón para volver a la imagen Menú.

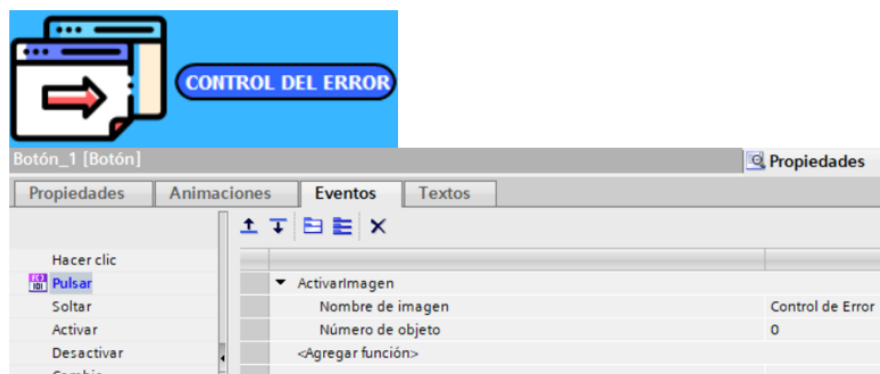


Figura 121. Botón para ir a la imagen Control del Error del HMI.

3.5.5 Control del error.

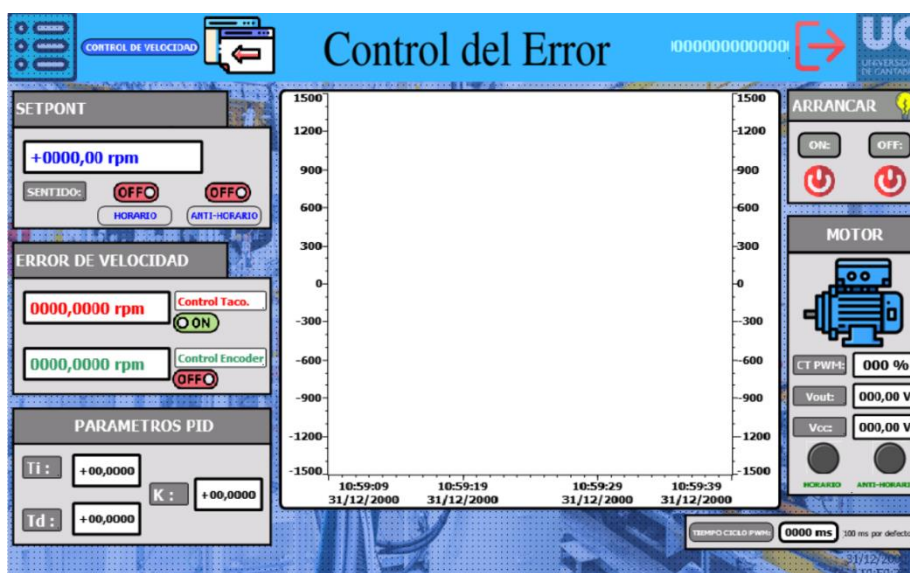


Figura 122. Imagen Control del Error del HMI.

Esta imagen presenta la misma estructura que la imagen Control de Velocidad de la Figura 113, lo que cambia son las curvas que se representan en la gráfica, que ahora serán las señales de error, y la visualización del valor en el instante actual de estas señales. Lo demás es igual, se puede seguir modificando el valor de referencia de velocidad, los parámetros PID, arrancar o parar el motor y el tiempo de ciclo.

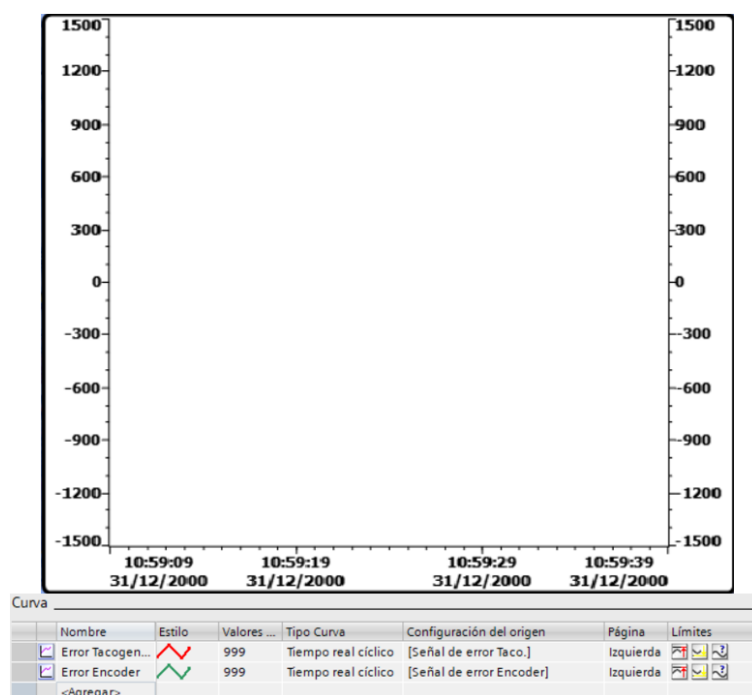


Figura 123. Visor gráfico de las señales de error.

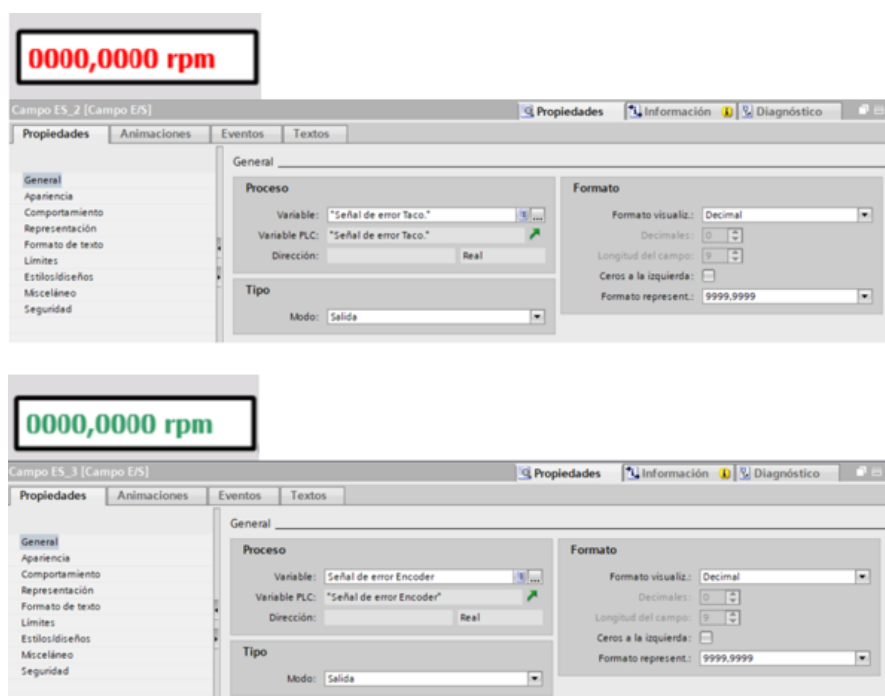


Figura 124. Cuadros de lectura de los errores.

Se puede volver a la imagen de Control de Velocidad pulsando el botón situado arriba a la izquierda, donde pone control de velocidad.

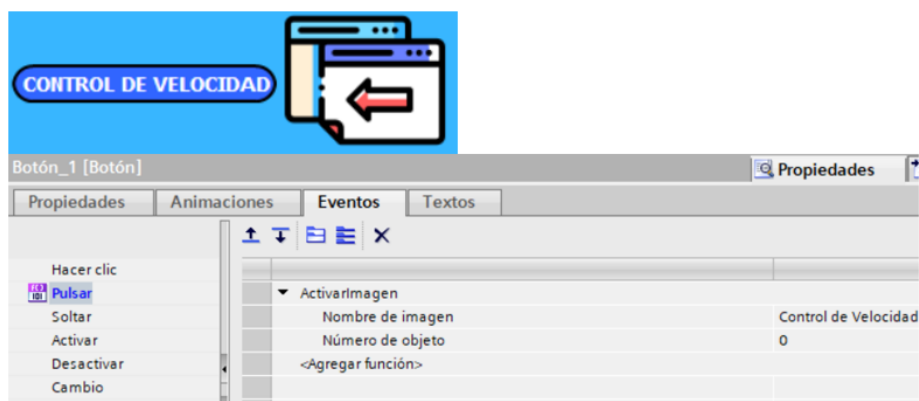


Figura 125. Botón para ir a la imagen Control de Velocidad del HMI.

3.5.6 Información.

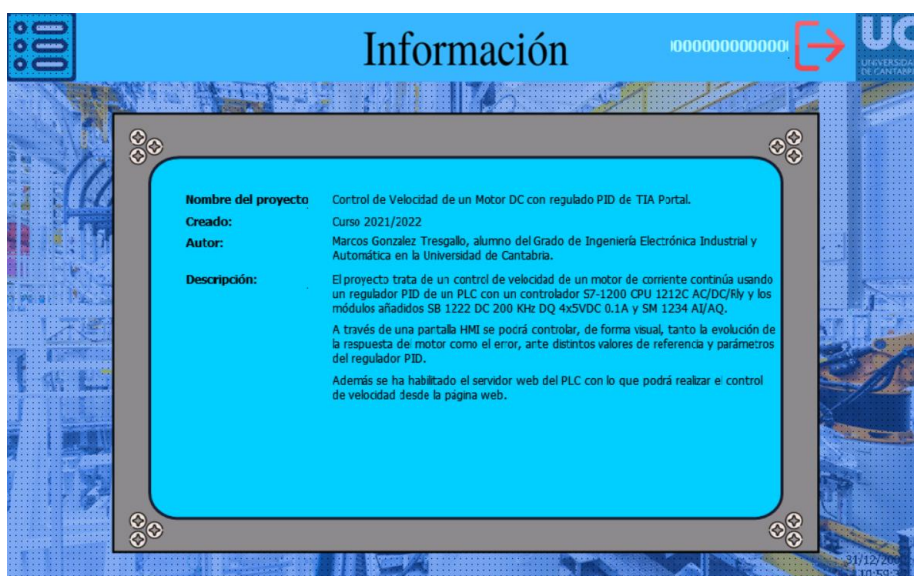


Figura 126. Imagen Información del HMI.

En esta imagen está redactado una breve descripción del proyecto, su nombre y el autor de dicho proyecto.

3.5.7 Avisos.

Como ya se señaló en el apartado 3.1.1.11 Variables de la imagen de avisos para el HMI, para la imagen de avisos se utilizan las dos variables del DB Avisos, “Warning Tacogenerador” y “Warning Encoder”.

En este apartado se comenta como se vincula estas dos variables con la ventana de avisos [16].

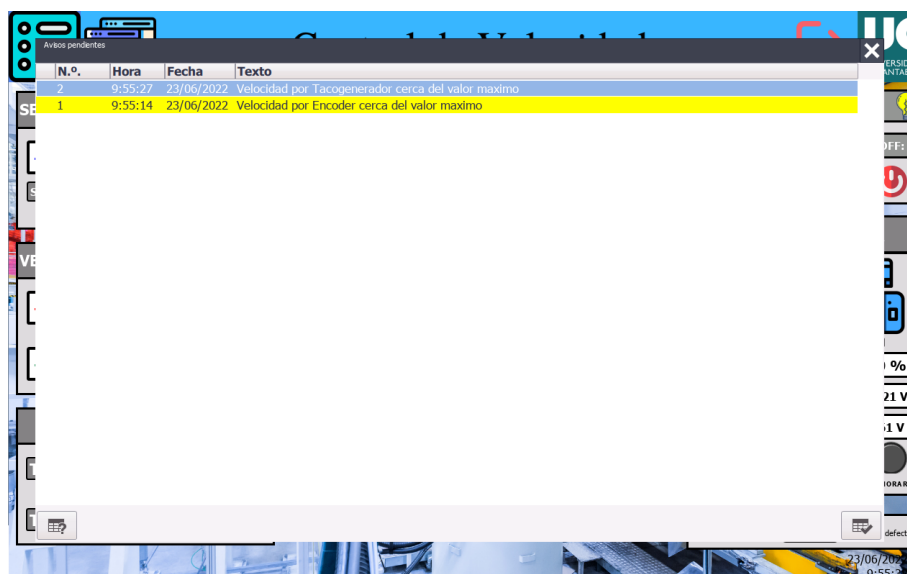


Figura 127. Ventana de avisos warnings del HMI.

Para conseguir que salgan estos avisos por pantalla se tuvo que hacer los siguientes pasos:

1. Ir a las variables del dispositivo HMI y añadir la siguiente variable:

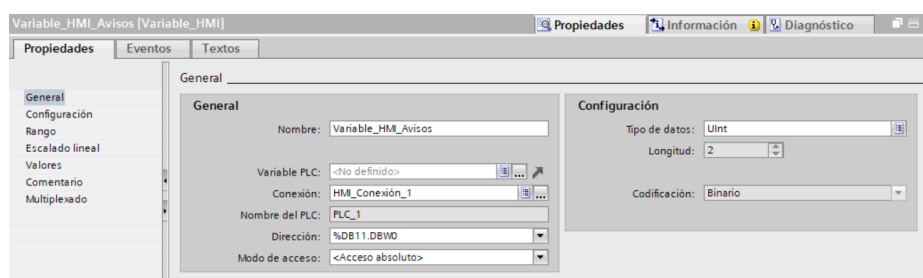


Figura 128. Propiedades de la variable Variable_HMI_Avisos.

Se creará una variable dónde la conexión se pondrá la conexión que hay entre el PLC y la pantalla HMI, que en este caso será la conexión “HMI_Conexión_1”, y se modificará el modo de acceso a “Acceso absoluto”.

2. Cambiar la dirección y el bit de llamada.

Ya asociadas las dos variables del DB Avisos a la pantalla HMI, se tiene que modificar la dirección, ya que por defecto lee el DB1 y en este caso es el DB11, y también, una cosa muy importante es el cambio del bit de disparo, debido a que en las alarmas del HMI, a diferencia que en las variables del PLC, se ordena de diferente forma.

Por normal general, una variable tamaño Word se cataloga de la siguiente manera:

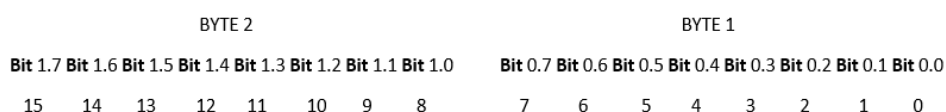


Figura 129. Estructura de una variable Word según la CPU.

Y en cambio en las alarmas de la pantalla HMI es la siguiente estructura:

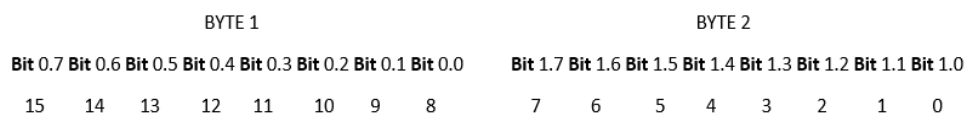


Figura 130. Estructura de una variable Word según las alarmas del HMI.

Por tanto, según la lógica de las alarmas, el bit 0.0 corresponde al bit de disparo 8.

3. Establecer categoría y texto de aviso.

Avisos de bit		Avisos analógicos	Variables de fichero			
ID	Nombre	Texto de aviso	Categoría	Variable de disparo	Bit de disparo	Dirección de disparo
1	Aviso de Velocidad Encoder Alta	Velocidad por Encoder cerca del valor máximo	Warnings	Variable_HMI_Avisos	8	%D811.DBX0.0
2	Aviso de Velocidad Tacogenerador Alta	Velocidad por Tacogenerador cerca del valor máximo	Warnings	Variable_HMI_Avisos	9	%D811.DBX0.1
<Agregar>						

Figura 131. Categoría de los avisos.

Estas variables son para dar una advertencia de que se está aproximando al límite de velocidad, por esa razón se categoriza como warnings y se pondrá como texto de aviso que la velocidad, por el encoder y tacogenerador, se aproxima a la velocidad máxima de giro del motor.

4. Ir a Administración de imagen, imagen general.

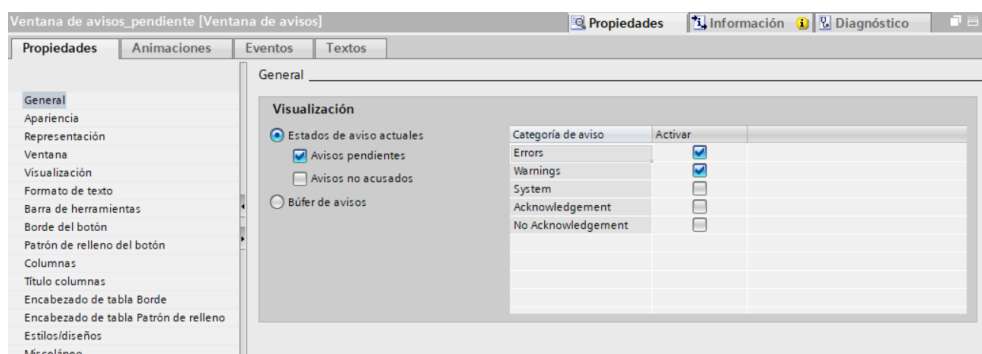


Figura 132. Cambio de los avisos en la ventana de avisos pendientes.

Por último, se tiene que activar en la ventana de avisos pendientes la categoría de avisos Warnings para que se puedan visualizar en dicha pantalla.

Una vez concluido el seguimiento de los pasos se obtendrá la pantalla de avisos que se ve en la Figura 127.

3.5.8 Usuarios.



Figura 133. Imagen Usuarios del HMI.

A esta imagen solo puede acceder el administrador, el cual podrá añadir y eliminar usuarios, poner las contraseñas y darles grupos para poder acceder a las distintas imágenes de la pantalla HMI.

Para dejar entrar a esta pantalla solamente al administrador lo que se hizo fue dar permisos de seguridad al botón correspondiente a la imagen Usuarios.

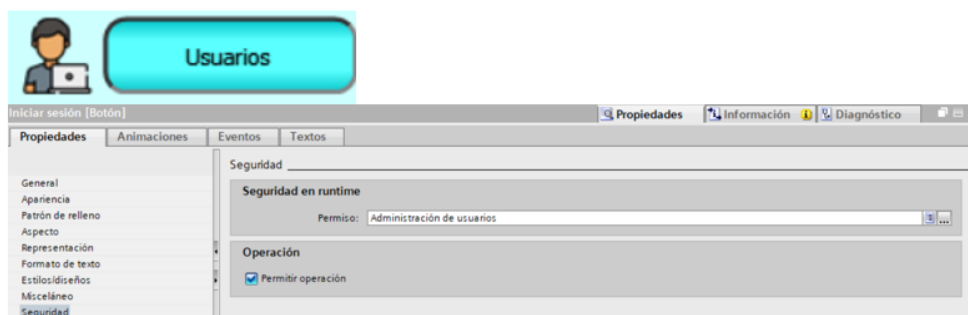


Figura 134. Dar permisos solo al Administrador.

3.6 EJEMPLOS DEL FUNCIONAMIENTO DEL HMI.

Aquí se muestran algunas imágenes de los resultados del sistema que se han ido obteniendo, dando valores de velocidad de referencia y variando los parámetros del PID.

Se mostrarán capturas tanto de la velocidad de las señales del tacogenerador y encoder estabilizándose a la velocidad de referencia como la señal de error de las mismas señales.

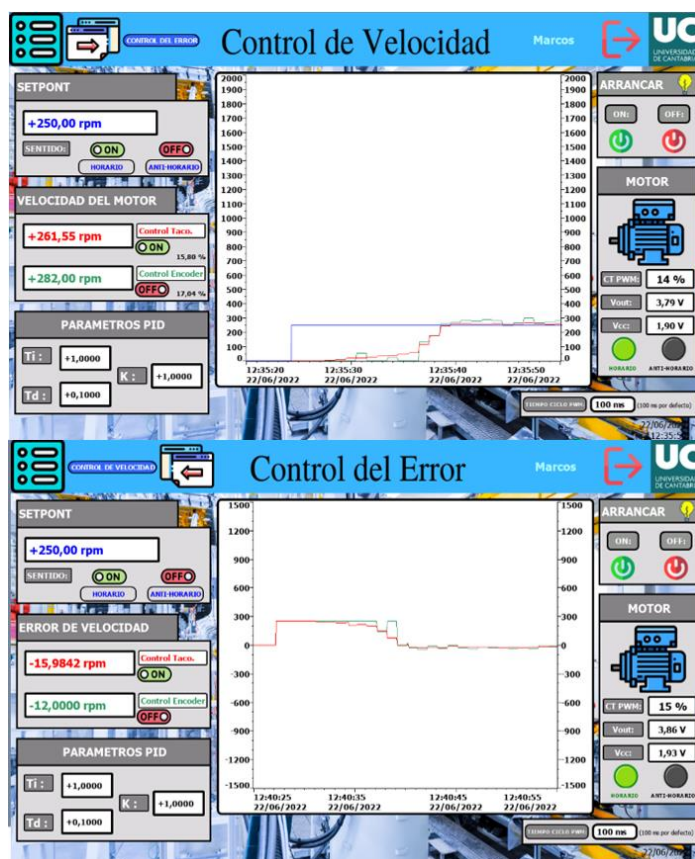


Figura 135. Respuesta a 250 rpm, $T_i=1$, $T_d=0.1$, $K=1$ y control Tacogenerador.

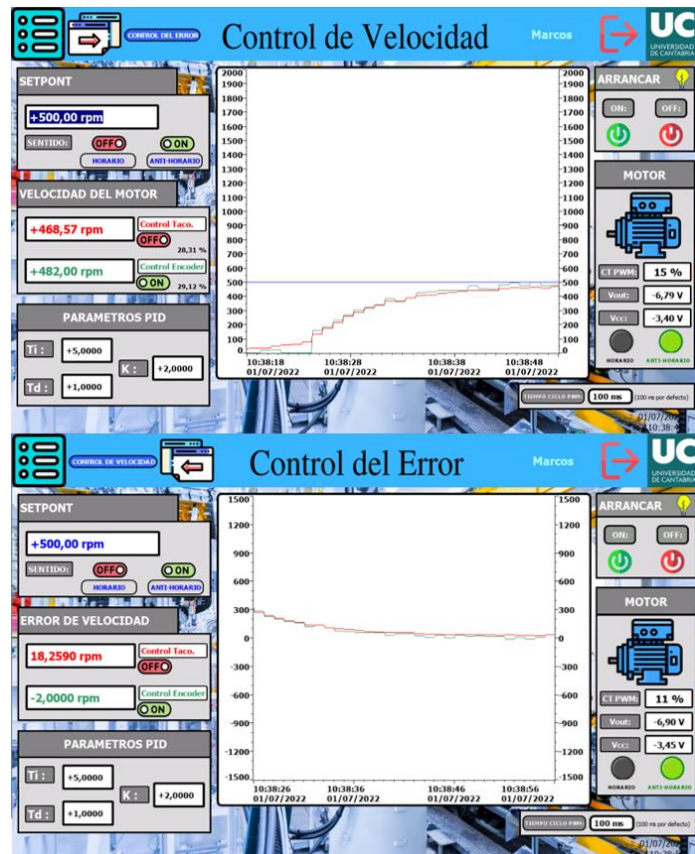


Figura 136. Respuesta a 500 rpm, $T_i=5$, $T_d=1$, $K=2$ y control Encoder.

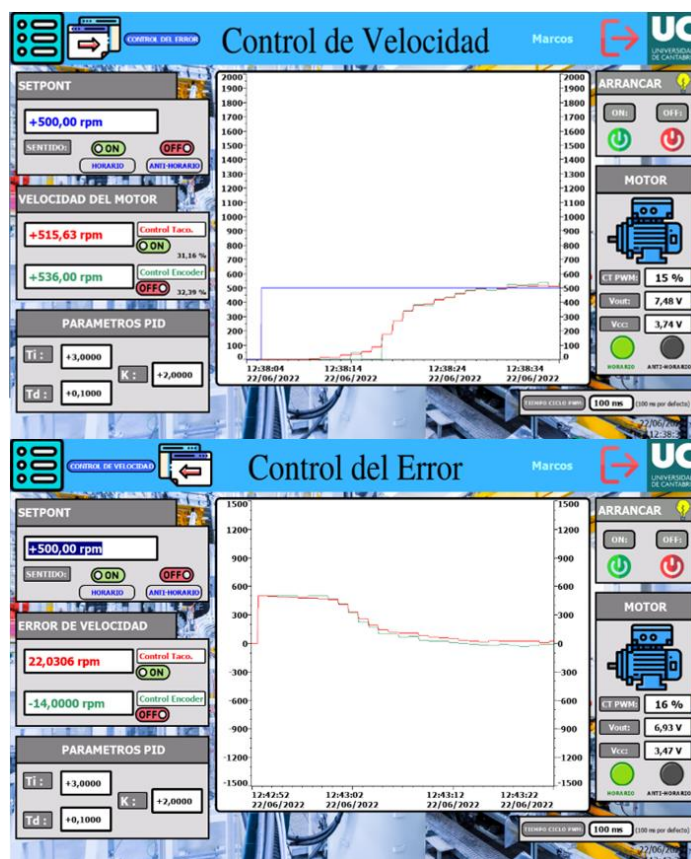


Figura 137. Respuesta a 500 rpm, $T_i=3$, $T_d=0.1$, $K=2$ y control Tacogenerador.

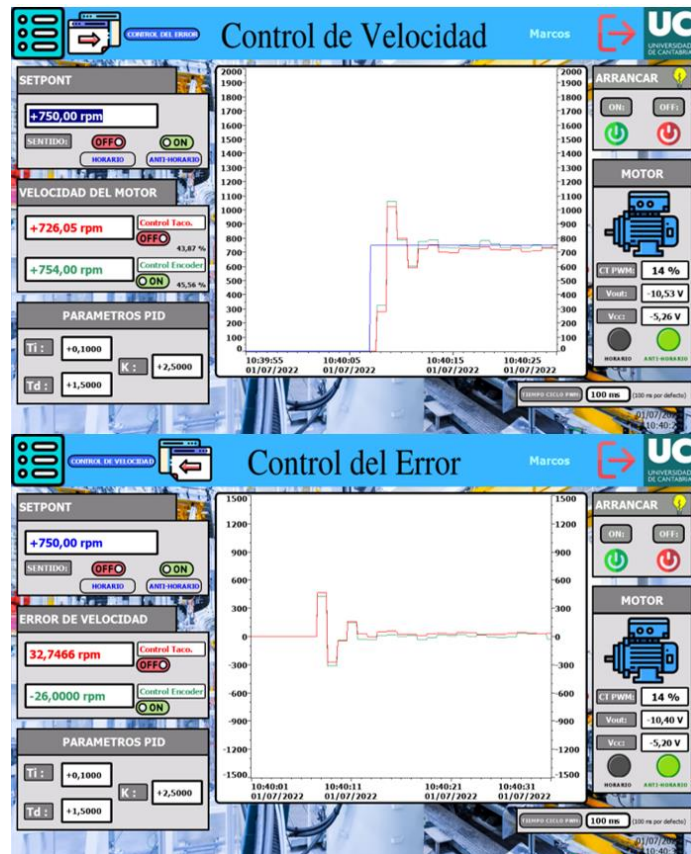


Figura 138. Respuesta a 750 rpm, $T_i=0.1$, $T_d=1.5$, $K=2.5$ y control Encoder.

Para la conexión entre la simulación del HMI y la carga de software con el PLC es necesario cambiar la interface PG/PC por tener un PLC físico y realizar una simulación de la pantalla HMI. Si no se hace este cambio tras la simulación de la pantalla HMI saltará el error N.º 190011 en la pantalla de avisos de sistema pendientes y todas las entradas y salidas asociadas a las variables de la CPU no darán ninguna lectura ni darán ningún cambio en el código del TIA Portal. Para corregirlo se tiene que cambiar dicha interface yendo al panel de control del ordenador o portátil y luego ir a la configuración de ajustar interface PG/PC y cambiar el adaptador de red a uno adecuado. El adaptador puede ser cualquiera, pero se debe de seleccionar el que pongo al final del nombre TCPIP.Auto. [19]

4. DESCRIPCIÓN PÁGINA WEB.

Gracias al desarrollo y diseño de una página web se podrá acceder remotamente, sin la necesidad de estar conectado el ordenador directamente con el PLC, al control de velocidad del motor DC.

En este capítulo se declarará, en primer lugar, cómo se ha ido diseñando la página web usando el editor de texto y de código fuente libre Notepad++, el cual posee varios lenguajes de programación, como puede ser C++, Python, Fortran, ..., y muchos más. Para el uso que se le da en este proyecto se utilizará el lenguaje de programación HTML con JavaScript, aplicando los conocimientos que se tiene de estos dos lenguajes. También hay que mencionar que se utilizó el manual de sistema S7-1200 [20] para conocer la forma de conectar las variables del PLC con el código. Además, se buscó una manera sencilla de poder recargar de forma automática sin recurrir a recargar la página manualmente, en tiempo real, los cambios de los valores de las variables que se observan en la página web, para ello se utilizó el lenguaje JavaScript, precisamente la biblioteca JQuery que permite simplificar la interacción con el documento HTML con la técnica AJAX, que es una técnica de desarrollo web que da la posibilidad de interactuar con la página web sin la necesidad de recargar la página. [21]

La forma de explicación del código será yendo parte por parte y viendo el resultado que se obtiene, pero primero se comentará el archivo CSS, con el cual se le está dando el diseño y estilo de la página web.

4.1 CÓDIGO ARCHIVO CSS.

Estos archivos tienen la función de formatear el contenido de la página HTML que está asociado al mismo. Contiene las propiedades que definen cómo se mostrarán los elementos HTML, como color, fuente, tamaño, y otros elementos.

Se basa en una serie de reglas que cada usuario define a los grupos de estilos que se aplican a cada elemento. Por dentro está estructurado por diferentes módulos, ya que hay una infinidad de propiedades de cada elemento que se pueden diseñar al gusto del usuario.

El funcionamiento es simple, cuando el navegador web aplica el archivo CSS se modifica la presentación de visualización. Se puede realizar dos formas distintas: Una de ellas es que se puede realizar la modificación de un conjunto de propiedades de todos los elementos, y la otra es que se puede seleccionar unos elementos en concreto para que solo afecte las modificaciones a dichos elementos o incluso, dentro de un elemento que solo afecte a una parte de este.

Aquí se muestran algunos ejemplos de lo que se ha ido mencionando. Se puede ver el código completo en el capítulo 7 Anexo.

```
/* Estilo para las tablas */
table{
    width:100%;
    background-color:white;
    text-align:left;
    font-size:15px;
    border-collapse:collapse;
    font-weight:bold;
    border-bottom:solid 8px #052D79;
    border-top:solid 8px #052D79;
    border-left:solid 8px #052D79;
    border-right:solid 8px #052D79;
    margin: 5px 5px;
}
```

Figura 139. Dar forma al elemento tabla.

```
/* La primera columna de las tablas */
table td:first-child{
    width:30%;
    height:100px;
    font-size:17px;
}
```

Figura 140. Modificar las columnas de las tablas.

```
/* Las filas pares de distinto color */
tr:nth-child(even){
    background-color: #E5E5E5;
}
```

Figura 141. Modificar las filas pares de la tabla.

```
/* Boton para Arranque, Paro y Control*/
.boton{
    width:45%;
    height:35%;
    background-color:#6CE2CB;
    border-radius:0.5em;
    font-size:20px;
    font-weight:700;
    padding:20px 40px;
    color:white;
    border-color:black;
    text-align:center;
    -webkit-box-shadow: inset 0px 0px 100px 0.1px rgba(0,95,95,1);
    -moz-box-shadow: inset 500px 0px 50px 100px rgba(0,95,95,1);
    box-shadow: inset 4px 4px 50px 5px rgba(0,95,80,1);
}

/* Cambio de color cuando se pasa por encima del boton */
.boton:hover{
    background-color:#7FFFFD;
    color:black;
    border-color:black;
    cursor: pointer;
    -webkit-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
    -moz-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
    box-shadow: inset 0px 0px 21px 1px rgba(0,143,100,1);
}

/* Cambio de color y tamaño cuando pulsa el boton */
.boton:active{
    width:46%;
    height:26%;
    padding: 22px 42px;
    font-size:22px;
    border-radius:0.5em;
    background-color:#D0FFFE;
    -webkit-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
    -moz-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
    box-shadow: inset 0px 0px 21px 1px rgba(0,143,100,1);
}
```

Figura 142. Dar forma y modificar el elemento botón.

Como se puede ver en la Figura 142, la clase “boton” tiene dos pseudo clases añadidas, “hover” y “active”.

La pseudo clase “hover” es utilizada para que cuando el usuario pase por encima del botón con el ratón, este cambie de color.

La pseudo clase “active” se usa con la finalidad de que cuando el usuario presiona el botón este se haga un poco más grande. Vuelve a la posición inicial una vez se deje de presionar.

Estas pseudo clases son usadas en los botones y en las cajas de texto que están repartidas por la página web para dar una sensación de animación.

4.2 CÓDIGO ARCHIVO HTML.

Es la parte más básica de la página web, define la estructura principal de la página web y utiliza marcas para etiquetar textos e imágenes, entre otras cosas, y son mostrados en la página web [21]. Estas marcas incluyen los elementos que ya se han podido visualizar en el anterior apartado, los cuales algunos son; “body”, “table”, “title”, “img”, “div”, “script”, “head”, entre otros. Todos los elementos se encuentran separados entre sí por los símbolos “<” y “>” y para cerrar un elemento se tendrá que escribir una barra de cierre como el siguiente ejemplo, </body>. Además, en cada elemento se pueden añadir distintas características o atributos de los cuales algunos que se han usado son; “method”, “class”, “type”, “name” y “value”.

Además del archivo HTML principal se encuentra presente dos llamadas a dos archivos HTM, con la finalidad de reducir la memoria de carga del PLC, ya que si toda la página se escribiera en un mismo archivo HTML, la memoria de carga del PLC sería superior al 100% y no se podría trabajar porque TIA Portal no cargaría los datos al PLC físico. Por esa razón se decidió buscar esta solución de utilizar archivos HTM, que viene a ser una extensión del HTML.

A continuación se explicará el código y se mostrará el resultado que se obtiene en la página web.

```

<!------- INTRODUCIR LOS DATOS PARA EL MOTOR DC ----->
<!-- AWP_In_Variable Name="Arrancar WEB" ----->
<!-- AWP_In_Variable Name="Parar WEB" ----->
<!-- AWP_In_Variable Name="Variables"."PONER RPM SETPOINT" -->
<!-- AWP_In_Variable Name="Seleccionar tipo de control" ----->
<!-- AWP_In_Variable Name="Giro horario WEB" ----->
<!-- AWP_In_Variable Name="Giro anti-horario WEB" ----->
<!-- AWP_In_Variable Name="Variables"."Parametro K" ----->
<!-- AWP_In_Variable Name="Variables"."Parametro Td" ----->
<!-- AWP_In_Variable Name="Variables"."Parametro Ti" ----->
<!------->

<!------- ESTADO DE LOS DATOS INTRODUCIDOS AL MOTOR DC ----->
<!-- AWP_In_Variable Name="Variables"."ACTIVAR PID" ----->
<!-- AWP_In_Variable Name="Control taco WEB" ----->
<!-- AWP_In_Variable Name="Control encoder WEB" ----->
<!-- AWP_In_Variable Name="LED giro anti-horario WEB" -->
<!-- AWP_In_Variable Name="LED giro horario WEB" ----->
<!------->

<!------- ESTADO DE LAS SEÑALES DEL MOTOR DC ----->
<!-- AWP_In_Variable Name="Velocidad por Encoder rpm" ----->
<!-- AWP_In_Variable Name="Velocidad por Encoder %" ----->
<!-- AWP_In_Variable Name="Velocidad por Taco. rpm" ----->
<!-- AWP_In_Variable Name="Velocidad por Taco. %" ----->
<!-- AWP_In_Variable Name="Señal de error Taco." ----->
<!-- AWP_In_Variable Name="Señal de error Encoder" ----->
<!-- AWP_In_Variable Name="Tension de alimentacion al motor V" -->
<!-- AWP_In_Variable Name="Tension de salida del motor V" ----->
<!------->

<!------- LISTADOS PARA OBSERVAR LAS IMAGENES DE ACTIVADO ----->
<!-- AWP_Enum_Def Name="listado1" Values='0:"Desactivado", 1:"Activado"' -->
<!-- AWP_Enum_Def Name="listado2" Values='0:"Desactivado", 1:"Activado"' -->
<!-- AWP_Enum_Def Name="listado3" Values='0:"Desactivado", 1:"Activado"' -->
<!-- AWP_Enum_Def Name="listado4" Values='0:"Desactivado", 1:"Activado"' -->
<!-- AWP_Enum_Def Name="listado5" Values='0:"Desactivado", 1:"Activado"' -->
<!------->

```

Figura 143. Conectar variables del PLC al archivo HTML.

Lo primero de todo es conectar las variables del PLC con las que se van a trabajar en el página web que se está desarrollando. Esto se realiza utilizando el comando AWP para identificar una variable en la CPU que sea escribible desde la página HTML. La variable debe especificarse por el nombre de la variable PLC o el nombre de variable de bloque de datos. En el código se utiliza el comando AWP_In_Variable, para poder escribir y leer dicha variable, y el comando AWP_Enum_Def, para asignar a una variable del PLC a un tipo de enumeración para realizar tanto operación de lecturas como de escritura. El uso que se le da en el proyecto es para la operación de lectura, donde en la página web se remplazará el valor numérico que se lea de la variable del PLC por el valor de texto de enumeración con el nombre de “listado” y el número de orden que se ha escrito. [20]

```

<html>
<head>
  <!-- Establecer todos los archivos que se va a usar -->
  <!-- Idioma y tipo de texto -->
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <!-- llamar al .css estilos -->
  <link rel="stylesheet" href="css/estilos.css"/>

  <!-- Habilitar JavaScript -->
  <meta http-equiv="content-script-type" content="text/javascript">
  <script type="text/javascript" src="jquery_min.js"></script>
  <!-- Otra alternativa es poniendo:
  <script type="text/javascript" src="http://code.jquery.com/jquery-latest.min.js"></script> -->

```

Figura 144. Cabeza del archivo HTML.

Ahora viene crear el archivo HTML. Se empieza estableciendo la cabeza, añadiendo los archivos que se van a usar como la llamada del archivo CSS y la habilitación del JavaScript.

```
<!-- Script para tiempo real -->
<script type="text/javascript">

    $(document).ready(function() {

        // Actualizar el contador a tiempos fijos
        setInterval(actualizar,500);

        // Enviar informacion al pulsar los botones sin recargar pagina.
        $.ajaxStart(function() {
            $('#loading').show();
            $('#result').hide();
        }).ajaxStop(function() {
            $('#loading').hide();
            $('#result').fadeIn('fast');
        });

        $('#T1, #T2, #T3, #T4, #T5, #T6, #T7, #T8, #T9').submit(function() {
            $.ajax({
                type: 'POST',
                url: $(this).attr('action'),
                data: $(this).serialize(),
                success: function(data) {
                    $('#result').html(data);
                }
            })
            return false;
        });

        function actualizar(){
            $("#signal").load("signal.htm");
            $("#output").load("output.htm");
        }

    });

</script>
```

Figura 145. Código JavaScript en el archivo HTML.

El habilitar el JavaScript se hizo con la intención de poder refrescar y actualizar las variables que se usan en la página web de forma automática. Por ello se tuvo que escribir este código, donde se actualiza el contador a tiempos fijos, el envío de la información al pulsar los botones y la actualización a tiempo real de las variables. [22]

```
<body background="FondoPLC.jpg" style="background-size:cover;background-position:left top">

    <!-- Titulo del proyecto -->
    <div id="header">
        <h1><b>CONTROL DE VELOCIDAD DE MOTOR DC CON PLC</b></h1>
    </div>

    <br>
    <br>

    <!-- CREACION DE LA TABLA DE CONTROL -->
    <table>
        <!-- Titulo -->
        <thead>
            <tr>
                <th colspan="3">INTRODUCIR LOS DATOS PARA EL MOTOR DC</th>
            </tr>
        </thead>
```

Figura 146. Primeras líneas de código del cuerpo HTML.

Una vez terminado con la cabeza toca escribir el cuerpo de la página. Con el atributo “background” del elemento “body” se modifica el fondo de la página, el cual se puso una imagen JPG [24].

Luego viene nombrar el título del proyecto que está vinculado con el estilo “header” del archivo CSS. Para ello se escribe dentro del elemento “div”, que sirve para crear secciones o, como para este caso, agrupar contenidos, y el atributo “id” para seleccionar el estilo.

Posteriormente se crea una tabla en donde se introducirán los datos que se necesitan para poner en marcha el control de velocidad, los cuales son; el arranque y paro del sistema, la velocidad de referencia, el tipo de control, el sentido de giro y los parámetros del PID.

En esta tabla habrá tres columnas, eso se establece con el atributo “colspan” del elemento “th”, que es el encabezado de la tabla creada.

4.2.1 Arranque y paro.

```
<!-- Pulsador de Start y Stop -->
<tr>
  <td>Arrancar y Parar:</td>
  <td>
    <form method="post" action="" id="T1" name="T1">
      <center>
        <input type="submit" name="Arrancar WEB" value="Start" class="boton">
      </center>
      <input type="hidden" name="Arrancar WEB" value="1">
      <div id="delay"></div>
      <input type="hidden" name="Arrancar WEB" value="0">
    </form>
  </td>
  <td>
    <form method="post" action="" id="T2" name="T2">
      <center>
        <input type="submit" name="Parar WEB" value="Stop" class="boton">
      </center>
      <input type="hidden" name="Parar WEB" value="1">
      <div id="delay"></div>
      <input type="hidden" name="Parar WEB" value="0">
    </form>
  </td>
</tr>
```

Figura 147. Fila de Arranque y Paro de la tabla.

Con el elemento “tr” se crean las filas de la tabla y para escribir en cada columna se escribe el elemento “td”.

En el arranque y paro se van a usar dos botones, que se consiguen con el atributo “submit”, y se llama a la clase “boton” del archivo CSS. Estos botones están vinculados a las variable “Arrancar WEB” y “Parar WEB” del PLC, usando para ello el atributo “name”.

Para que sea a tiempo real y se envíe la información al pulsar los botones se tiene que llamar al elemento “form”. Este elemento contiene los controles interactivos que permiten enviar la información y con el atributo “id” se vincula con uno de los tiempos reales que se crearon en el código de JavaScript mencionado antes. Para el botón de arranque se le vincula con el tiempo #T1, y lo mismo con el botón de Parar WEB, pero este con el tiempo #T2.

Estos botones actúan como pulsadores, es decir, una vez se dejen de presionar van a volver a la posición inicial, eso se consigue llamado al “delay” creado en el archivo CSS a través del atributo “id”.

4.2.2 Velocidad de referencia.

```

<!-- Establecer SetPoint rpm -->
<tr>
  <td>SetPoint (rpm):</td>
  <td>
    <form method="post" action="" id="T3" name="T3">
      <center>
        <input type="number" max="1655" min="0" step="0.001" value="0" name="Variables"."PONER RPM SETPOINT" class="cajaTexto">
        <center>
          <button type="submit" class="botonSet">Set</button>
        </center>
      </td>
    </center>
  </form>
</td>
</tr>

```

Figura 148. Fila de Velocidad de referencia de la tabla.

En esta parte del código se escribirá la velocidad a la que el motor debe girar. Para poder escribir se llama al atributo “type”, y para poner números se debe de poner seguido “number”. En el archivo CSS se creó una caja de texto para darle la forma. Además, se le añadieron los atributos “max”, “min” y “step” para tener la velocidad entre unos valores máximo y mínimo, y con un paso pequeño para poner valores decimales.

También se encuentra un botón, en este caso de la clase “botonSet” del archivo CSS, con el cual se mandará el valor de velocidad escrito al PLC cuando este sea pulsado.

4.2.3 Tipo de control.

```

<!-- Establecer Tipo de Control -->
<tr>
  <td>Tipo de Control:</td>
  <td>
    <form method="post" action="" id="T4" name="T4">
      <center>
        <input type="submit" name="Seleccionar tipo de control" value="Encoder" class="boton">
      </center>
      <input type="hidden" name="Seleccionar tipo de control" value="1">
    </form>
  </td>
  <td>
    <form method="post" action="" id="T4" name="T4">
      <center>
        <input type="submit" name="Seleccionar tipo de control" value="Tacogenerador" class="boton">
      </center>
      <input type="hidden" name="Seleccionar tipo de control" value="0">
    </form>
  </td>
</tr>

```

Figura 149. Fila de Tipo de Control de la tabla.

Para establecer el tipo de control se insertó dos botones, los cuales se encuentran vinculados a la misma variable del PLC “Seleccionar tipo de control”, pero un botón dará el valor de 1 al ser pulsado y el otro el valor de 0. Esto se hace dándole el valor de 1 o 0 con el atributo “value”, que define el valor predeterminado, en el elemento “input”, el cual es usado normalmente para crear controles interactivos con el fin de recibir datos del usuario.

4.2.4 Sentido de giro.

```

<!-- Boton horario -->
<tr>
  <td>Giro del Motor Horario:</td>
  <td>
    <form method="post" action="" id="T5" name="T5">
      <center>
        <input type="submit" name="Giro horario WEB" value="ON" class="botonMotor">
      </center>
      <input type="hidden" name="Giro horario WEB" value="1">
    </form>
  </td>
  <td>
    <form method="post" action="" id="T5" name="T5">
      <center>
        <input type="submit" name="Giro horario WEB" value="OFF" class="botonMotor">
      </center>
      <input type="hidden" name="Giro horario WEB" value="0">
    </form>
  </td>
</tr>

<!-- Boton anti-horario -->
<tr>
  <td>Giro del Motor Anti-Horario:</td>
  <td>
    <form method="post" action="" id="T6" name="T6">
      <center>
        <input type="submit" name="Giro anti-horario WEB" value="ON" class="botonMotor">
      </center>
      <input type="hidden" name="Giro anti-horario WEB" value="1">
    </form>
  </td>
  <td>
    <form method="post" action="" id="T6" name="T6">
      <center>
        <input type="submit" name="Giro anti-horario WEB" value="OFF" class="botonMotor">
      </center>
      <input type="hidden" name="Giro anti-horario WEB" value="0">
    </form>
  </td>
</tr>

```

Figura 150. Fila de Sentido de giro de la tabla.

Al igual que con el anterior apartado, en el sentido de giro se tiene la misma estructura de código, dos botones para activar o desactivar la variable del PLC del sentido de giro. Lo que difiere de lo anterior comentado es que, aparte del tiempo real y las variables vinculadas, cada botón tiene una clase distinta. En este apartado se usa el botón de clase “botonMotor” del archivo CSS.

Se han creado tres tipos de botones en el archivo CSS simplemente para tener botones de distintos tamaños, ya que se pudo haber creado solamente un único diseño.

4.2.5 Parámetros del PID

```

<!-- Establecer parametros PID-->
<tr>
<td>Parámetro K:</td>
<td>
<form method="post" action="" id="T7" name="T7">
<center>
<input type="number" max="100" min="0" step="0.001" value="0" name="Variables"."Parametro K" class="cajaTexto">
<td>
<center>
<button type="submit" class="botonSet">Set</button>
</center>
</td>
</center>
</form>
</td>
</tr>

<tr>
<td>Parámetro Td:</td>
<td>
<form method="post" action="" id="T8" name="T8">
<center>
<input type="number" max="100" min="0" step="0.001" value="0" name="Variables"."Parametro Td" class="cajaTexto">
<td>
<center>
<button type="submit" class="botonSet">Set</button>
</center>
</td>
</center>
</form>
</td>
</tr>

<tr>
<td>Parámetro Ti:</td>
<td>
<form method="post" action="" id="T9" name="T9">
<center>
<input type="number" max="100" min="0" step="0.001" value="0" name="Variables"."Parametro Ti" class="cajaTexto">
<td>
<center>
<button type="submit" class="botonSet">Set</button>
</center>
</td>
</center>
</form>
</td>
</tr>

```

Figura 151. Filas de los Parámetros del PID de la tabla.

Tiene la misma estructura que en lo comentado del apartado 4.2.2 Velocidad de referencia. Lo que cambia son las variables vinculadas, que serán los parámetros del PID, los límites de estas variables y el tiempo real.

4.2.6 Resultado página web.

Header → CONTROL DE VELOCIDAD DE MOTOR DC CON PLC		
Thead → INTRODUCIR LOS DATOS PARA EL MOTOR DC		
Arranque y paro	Arrancar y Parar:	Start Stop
Velocidad de referencia	SetPoint (rpm):	0 Set
Tipo de control	Tipo de Control:	Encoder Tacogenerador
Sentido de giro	Giro del Motor Horario:	ON OFF
	Giro del Motor Anti-Horario:	ON OFF
Parámetros del PID	Parámetro K:	0 Set
	Parámetro Td:	0 Set
	Parámetro Ti:	0 Set

Figura 152. Tabla final de introducción de datos.

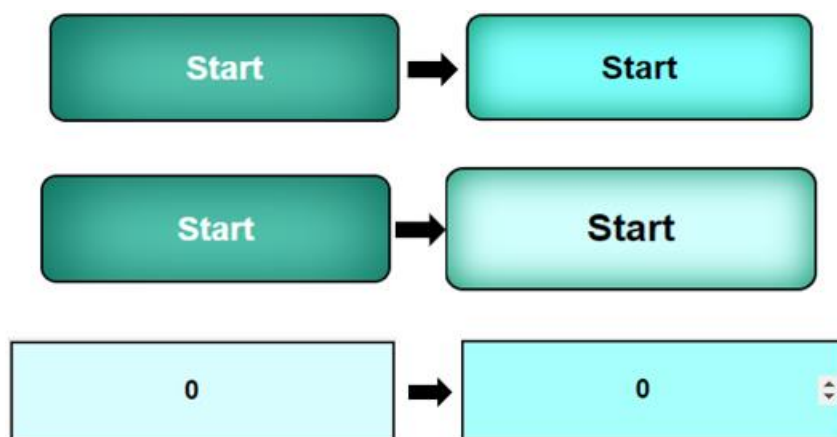


Figura 153. Ejemplo de las animaciones de los botones y cajas.

The image shows a web form with three input fields. The first field contains '-200' and has a validation error message: 'El valor debe ser superior o igual a 0'. The second field contains '1800' and has a validation error message: 'El valor debe ser inferior o igual a 1655'. The third field contains '0.00005' and has a validation error message: 'Introduce un valor válido. Los dos valores válidos más aproximados son 0 y 0,001.'

Figura 154. Prueba de los límites de las cajas.

Lo que se obtiene como resultado de lo ya comentado es la tabla que se muestra en la Figura 152. Al pasar por encima de un botón, este cambiará de color como se puede apreciar en la primera imagen de la Figura 153. Cuando sea pulsado, el botón de agrandará como se ve en la segunda imagen de la misma figura. Algo similar ocurre con las cajas de texto, al pasar por encima esta cambiará de color.

Además, en la Figura 154 se puede ver que ocurre cuando se introducen uno valores fuera de los rangos máximo y mínimo que se establecieron. Tampoco se puede escribir ninguna palabra por tener el atributo “type=“number””.

Como ya se ha indicado, se tiene dos archivos HTM para recudir la memoria de cargar del PLC. La forma de redactar el código es la misma que se ha estado comentando, lo que se tiene que hacer es una llamada de los archivos HTM en el código del HTML, como se muestra a continuación en la Figura 155. Usando el atributo “id”, como se lleva haciendo para llamar a un elemento o, en este caso a otro archivo, en el archivo HTML.

```
<div id="signal"></div>

<br>

<div id="output"></div>
```

Figura 155. Llamar a los archivos HTM.

4.3 ARCHIVO HTM SIGNAL.

```

<!-- AWP_Enum_Ref Name='"Control taco WEB"' Enum="listado1" -->
<!-- AWP_Enum_Ref Name='"Control encoder WEB"' Enum="listado2" -->
<!-- AWP_Enum_Ref Name='"Variables"."ACTIVAR PID"' Enum="listado3" -->
<!-- AWP_Enum_Ref Name='"LED giro anti-horario WEB"' Enum="listado4" -->
<!-- AWP_Enum_Ref Name='"LED giro horario WEB"' Enum="listado5" -->

<table>
  <thead>
    <tr>
      <th colspan="3">ESTADO DE LOS DATOS INTRODUCIDOS AL MOTOR DC</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Estado del Motor DC:</td>
      <td align="right">:=Variables".ACTIVAR PID":</td>
      <td>
        <center>
          
        </center>
      </td>
    </tr>
    <tr>
      <td>Velocidad de referencia:</td>
      <td align="right">:=Variables".PONER RPM SETPOINT": <b>r.p.m.</b></td>
      <td></td>
    </tr>
    <tr>
      <td>Tipo de Control establecido:</td>
      <td>Tacogenerador
        <center>
          
        </center>
      </td>
      <td>Encoder
        <center>
          
        </center>
      </td>
    </tr>
    <tr>
      <td>Giro del Motor Horario:</td>
      <td align="right">:=LED giro horario WEB:</td>
      <td>
        <center>
          
        </center>
      </td>
    </tr>
    <tr>
      <td>Giro del Motor Anti-Horario:</td>
      <td align="right">:=LED giro anti-horario WEB:</td>
      <td>
        <center>
          
        </center>
      </td>
    </tr>
  </tbody>
</table>

```

Figura 156. Código del archivo signal HTM.

En el archivo signal, de extensión HTM, se mostrarán los valores, y si han sido activadas o no, de las variables que se han explicada en el archivo HTML. Además, con el comando AWP_Enum_Def, que se utilizó en el archivo HTML para cambiar el valor numérico de una variable a un texto que se quiera, ahora en esta sección se utilizará el comando AWP_Enum_Ref para vincular ese texto con la variable. Haciendo esto se realizará un juego para mostrar unas imagen.

Las imágenes serán dos, una de ellas se mostrará cuando la variable esté activada y la otra cuando esté desactivada. Para ello se hizo lo siguiente:

- Se crearon cinco listados con el comando AWP_Enum_Def.
- Cuando la variable que se vincule con cada listado tenga el valor de 0, en vez de dar el valor numérico saldrá el valor de texto “Desactivado”. Y cuando tenga el valor de 1 saldrá el texto “Activado”.
- Se descargaron dos imágenes de botones Leds, una verde claro y otra oscuro, para simular que esté encendido o apagado. El nombre con el que se ha guardado la imagen de verde claro es “VerdeActivado” y la de la imagen de verde oscuro “VerdeDesactivado”.
- Para mostrar la imagen de “VerdeActivado” o de “VerdeDesactivado” se puso el código siguiente:

```

```

Figura 157. Código para las imágenes.

- Cuando la variable sea 1 dará el valor de texto Activado y la imagen saldrá la VerdeActivado, sino será la otra imagen.

Para mostrar el valor de la variable se tiene que escribir, donde se quiera que se muestre el valor, el siguiente comando de texto: **:=”Nombre_de_la_variable”**: .

4.3.1 Resultados página web.

ESTADO DE LOS DATOS INTRODUCIDOS AL MOTOR DC			
Estado del Motor DC:	Desactivado		
Velocidad de referencia:	0 r.p.m.		
Tipo de Control establecido:	Tacogenerador		Encoder 
Giro del Motor Horario:	Desactivado		
Giro del Motor Anti-Horario:	Desactivado		

Figura 158. Tabla del archivo HTM signal en la página web.

En la Figura 158 se puede ver el resultado que se obtiene en forma de tabla del código escrito en el archivo HTM con el nombre de signal. Esta imagen se tomó como captura durante la prueba de funcionamiento de la página web.

4.4 ARCHIVO HTM OUTPUT.

```

<table>
  <thead>
    <tr>
      <th colspan="3">ESTADO DE LAS SEÑALES DEL MOTOR DC</th>
    </tr>
  </thead>

  <tr>
    <td>Velocidad del Motor por la señal del Encoder:</td>
    <td align="right">:= "Velocidad por Encoder rpm": <b>r.p.m.</b></td>
    <td align="right">:= "Velocidad por Encoder %": <b>%</b></td>
  </tr>

  <tr>
    <td>Velocidad del Motor por la señal del Tacogenerador:</td>
    <td align="right">:= "Velocidad por Taco. rpm": <b>r.p.m.</b></td>
    <td align="right">:= "Velocidad por Taco. %": <b>%</b></td>
  </tr>

  <tr>
    <td>Error de la señal del Encoder:</td>
    <td align="right">:= "Señal de error Encoder": <b>r.p.m.</b></td>
    <td></td>
  </tr>

  <tr>
    <td>Error de la señal del Tacogenerador:</td>
    <td align="right">:= "Señal de error Taco.": <b>r.p.m.</b></td>
    <td></td>
  </tr>

  <tr>
    <td>Tensión de alimentacion del Motor:</td>
    <td align="right">:= "Tension de alimentacion al motor V": <b>V</b></td>
    <td></td>
  </tr>

  <tr>
    <td>Tensión de salida del Motor:</td>
    <td align="right">:= "Tension de salida del motor V": <b>V</b></td>
    <td></td>
  </tr>
</table>

```

Figura 159. Código del archivo output HTM.

Este archivo de señales de salida del motor se verá en una tabla las variables más interesante y que más información puedan aportar acerca del estado del motor. Estas variables serán; la velocidad, tanto en r.p.m. cómo en porcentaje, de la señal analógica del tacogenerador y la señal digital del encoder, los errores del tacogenerador y encoder, y las tensiones de alimentación al motor y de salida del motor.

4.4.1 Resultados página web.

ESTADO DE LAS SEÑALES DEL MOTOR DC			
Velocidad del Motor por la señal del Encoder:	0 r.p.m.	0 %	
Velocidad del Motor por la señal del Tacogenerador:	0 r.p.m.	0 %	
Error de la señal del Encoder:	0 r.p.m.		
Error de la señal del Tacogenerador:	0 r.p.m.		
Tensión de alimentación del Motor:	0 V		
Tensión de salida del Motor:	0 V		

Figura 160. Tabla del archivo HTM output en la página web.

Al igual que en la Figura 158, la Figura 160 fue tomada una vez estaban vinculadas el código de la página web con el PLC físico.

4.5 EJEMPLOS DEL FUNCIONAMIENTO DE LA PÁGINA WEB.

Aquí se muestran algunas capturas de la página web donde se muestran los valores que se han introducido y los valores de las señales del motor en un determinado instante.

ESTADO DE LOS DATOS INTRODUCIDOS AL MOTOR DC		
Estado del Motor DC:	Activado	
Velocidad de referencia:	800 r.p.m.	
Tipo de Control establecido:	Tacogenerador 	Encoder 
Giro del Motor Horario:	Desactivado	
Giro del Motor Anti-Horario:	Activado	

ESTADO DE LAS SEÑALES DEL MOTOR DC		
Velocidad del Motor por la señal del Encoder:	790 r.p.m.	46.76737 %
Velocidad del Motor por la señal del Tacogenerador:	778.9153 r.p.m.	46.45305 %
Error de la señal del Encoder:	40 r.p.m.	
Error de la señal del Tacogenerador:	37.12878 r.p.m.	
Tensión de alimentación del Motor:	-5.551649 V	
Tensión de salida del Motor:	-11.28993 V	

Figura 161. Ejemplo N.º 1 del funcionamiento de la página web.

ESTADO DE LOS DATOS INTRODUCIDOS AL MOTOR DC		
Estado del Motor DC:	Activado	
Velocidad de referencia:	500 r.p.m.	
Tipo de Control establecido:	Tacogenerador 	Encoder 
Giro del Motor Horario:	Activado	
Giro del Motor Anti-Horario:	Desactivado	

ESTADO DE LAS SEÑALES DEL MOTOR DC		
Velocidad del Motor por la señal del Encoder:	558 r.p.m.	33.71601 %
Velocidad del Motor por la señal del Tacogenerador:	544.8995 r.p.m.	32.84486 %
Error de la señal del Encoder:	-44 r.p.m.	
Error de la señal del Tacogenerador:	-34.48267 r.p.m.	
Tensión de alimentación del Motor:	3.844619 V	
Tensión de salida del Motor:	7.821181 V	

Figura 162. Ejemplo N.º 2 del funcionamiento de la página web.

5. CONCLUSIÓN.

Se ha desarrollado una aplicación para el control de velocidad de un motor de corriente continua mediante la utilización de un PLC para una planta de control de velocidad de un motor DC existente en el laboratorio de docencia del grupo de Ingeniería de Control para su uso en la impartición de prácticas.

A lo largo del documento se han ido explicando los pasos que se siguieron para su realización indicando los resultados obtenidos, que se pueden ver en las imágenes de las figuras, que muestran el funcionamiento de la planta de control mediante la utilización del PLC usando un regulador PID implementado en él.

Se ha diseñado también una pantalla HMI y una página web para dicha aplicación, lo que permite que los alumnos puedan utilizar, modificar, regular y, en general, ver la evolución de la respuesta del motor DC controlando su velocidad.

Esta aplicación permitirá a los alumnos manejar un proceso de regulación y poder analizar los efectos de las acciones de control mediante la modificación de los parámetros del regulador PID.

La aplicación está diseñada para interaccionar con ella mediante un panel de operador HMI y con posibilidad de utilizarlo también de forma remota mediante una página web, lo que permite ampliar la funcionalidad de dicha planta al poder utilizarse como laboratorio remoto.

Esta aplicación ha permitido complementar una planta que disponía únicamente de un sistema de regulación PID analógico con un regulador PID discreto controlador mediante un autómatas, lo cual es muy útil para que los alumnos puedan analizar diferentes estrategias de regulación. Además, el software de TIA Portal es una gran ayuda para acercar a los alumnos a sistemas de automatización de empresas industriales, ya que está muy extendido en la industria 4.0.

Además, el sistema desarrollado es abierto lo cual también permitirá en un futuro, únicamente mediante la modificación del código del PLC, implementar nuevas estrategias de control para dicha planta.

6. PRESUPUESTO DE MATERIAL.

UNIDAD	DESCRIPCION	PRECIO UNIDAD	PRECIO TOTAL IVA INCLUIDO
1 ud	Software de programación PLC Siemens, V16, para SIMATIC STEP 7, para Windows 7 (64 bits), Windows 10 (64-) Código RS: 200-9092 N.º ref. fábrica.: 6ES7822-0AA06-0YE5 Fabricante: Siemens	152.69 €	184.75 €
1 ud	Controlador lógico Siemens S7-1200, 8 (Digital Input, 2 switch as Analógicas Input) entradas tipo Analógico, Digital, 6 Código RS: 862-4461 N.º ref. fábrica.: 6ES7212-1BE40-0XB0 Fabricante: Siemens	240.73 €	291.28 €
1 ud	Módulo de E/S PLC Siemens para usar con SIMATIC S7-1200 Series, 4 entradas tipo Analógico, 2 salidas tipo Analógico, 24 Código RS: 787-7986 N.º ref. fábrica.: 6ES7234-4HE32-0XB0 Fabricante: Siemens	314.58 €	380.64 €
1 ud	Módulo de E/S PLC Siemens para usar con Serie S7-1200, 4 salidas tipo Digital, transistor, 5 V DC Código RS: 46-5779 N.º ref. fábrica.: 6ES7222-1AD30-0XB0 Fabricante: Siemens	61.60 €	74.54 €
7 ud	Resistencias de 1.2, 1.8, 5.1, 10, 12 kΩ	0.007 €	0.06 €
1 ud	Condensador cerámico de 100 nF	0.19 €	0.23 €
1 ud	Conector DIN macho de 8 vías	4.78 €	5.78 €
1 ud	Controlador de motor L298N Dual H-Bridge	9.99 €	9.99 €

1 ud	Interruptor Óptico Ranurado Vishay TCST2103, Fototransistor, Montaje en orificio pasante	2.146 €	2.60 €
1 ud	Amplificadores operacionales de alta velocidad EL2244CN	1.97 €	2.24 €
1 ud	Placa circuito impreso perforada de una cara	4.58 €	5.54 €
1 ud	Caja de ABS para protección y guardado de los circuitos	14.70 €	17.79 €
0.7 m	Cable Ethernet Cat5e UTP	0.72 €	0.61 €
PRECIO TOTAL:			976.05 €

Tabla 3. Presupuesto de material.

El coste total del proyecto es de novecientos setenta y seis euros y cinco céntimos [23].

7. BIBLIOGRAFÍA.

- [1] <https://nachoaguero-blog.tumblr.com/post/43640743894/partes-y-funcionamiento-de-un-motor-de-corriente>
- [2] https://www.zuendo.com/smartblog/26_Motores-cc-ventajas-inconvenientes.html
- [3] [Regulación de velocidad en motor de corriente continua \(mootio-components.com\)](#)
- [4] [¿Qué es un PLC? ¿Cómo funciona? ¿Para qué sirve? \(srcsl.com\)](#)
- [5] Manual del motor DC. Elettronica Veneta.
- [6] Data sheet TRANSDUCTOR Y CONTROL DE VELOCIDAD Y POSICIÓN Mod. G36A/EV. [Transductores y control de velocidad y posición - Elettronica Veneta S.p.A.](#)
- [7] Data sheet TRANSDUCTORES DE VELOCIDAD Y ACELERACIÓN Y ACONDICIONADOR DE SEÑAL Mod. G28/EV. [Transductores de velocidad y aceleración y acondicionadores de señal - Elettronica Veneta S.p.A.](#)
- [8] Data sheet del sensor TCST2103. [TCST2103 Datasheet\(PDF\) - Vishay Siliconix \(alldatasheet.com\)](#)
- [9] [Proyecto tutoriales 1 1 - YouTube](#)
- [10] Data sheet del circuito integrado EL2244CN. [EL2244CN Datasheet\(PDF\) - Elantec Semiconductor \(alldatasheet.com\)](#)
- [11] [Puente H doble L298N — Talos Electronics](#)
- [12] Data sheet L298N Dual H-Bridge Motor Driver. Handson Technology. [L298N Motor Driver.pdf \(handsontec.com\)](#)
- [13] [Portal de automatización totalmente integrada | Software de automatización](#)
- [14] Ventana de ayuda del programa TIA PORTAL.
- [15] [Controlador PID - Control Automático - Picuino](#)
- [16] [CURSO PANTALLA HMI TIA PORTAL 9 / 10 ! ! AVISOS HMI TIA PORTAL 🔔 🔔 🔔 - YouTube](#)
- [17] [Herramienta de diseño gratuita: presentaciones, vídeos y redes sociales| Canva](#)
- [18] [Iconos vectoriales y stickers - PNG, SVG, EPS, PSD y CSS \(flaticon.es\)](#)
- [19] [SIMULAR UN HMI DE SIEMENS Conectado a un PLC REAL en TIA PORTAL - YouTube](#)

-
- [20] S7 Controlador programable S7-1200. Manual de Sistema. [s71200 system manual es-ES es-ES.pdf - SIMATIC S7 Controlador programable S7-1200 - ID: 109759862 - Industry Support Siemens](#)
- [21] [MDN Web Docs \(mozilla.org\)](#)
- [22] [Web Server en Tia Portal para Control PID motor DC \(TIA PORTAL S7-1200\) - YouTube](#)
- [23] [RS Components | Su partner digital de soluciones industriales \(rs-online.com\)](#)
- [24] Fondo de la página web. [215-2159265 siemens-plc.jpg \(500×281\) \(itl.cat\)](#)
- [25] Fondo del HMI. [ROBOT.jpg \(4800×3199\) \(newsbook.es\)](#)

8. ANEXO.

8.1 CÓDIGO DE TIA PORTAL.

Totally Integrated Automation Portal

Control_de_Velocidad_con_HMI_Control_TACO_ENCODER / PLC_1 [CPU 1212C AC/DC/Rly] / Bloques de programa

Main [OB1]

Main Propiedades

General

Nombre	Main	Número	1	Tipo	OB
Idioma	KOP	Numeración	Automático		

Información

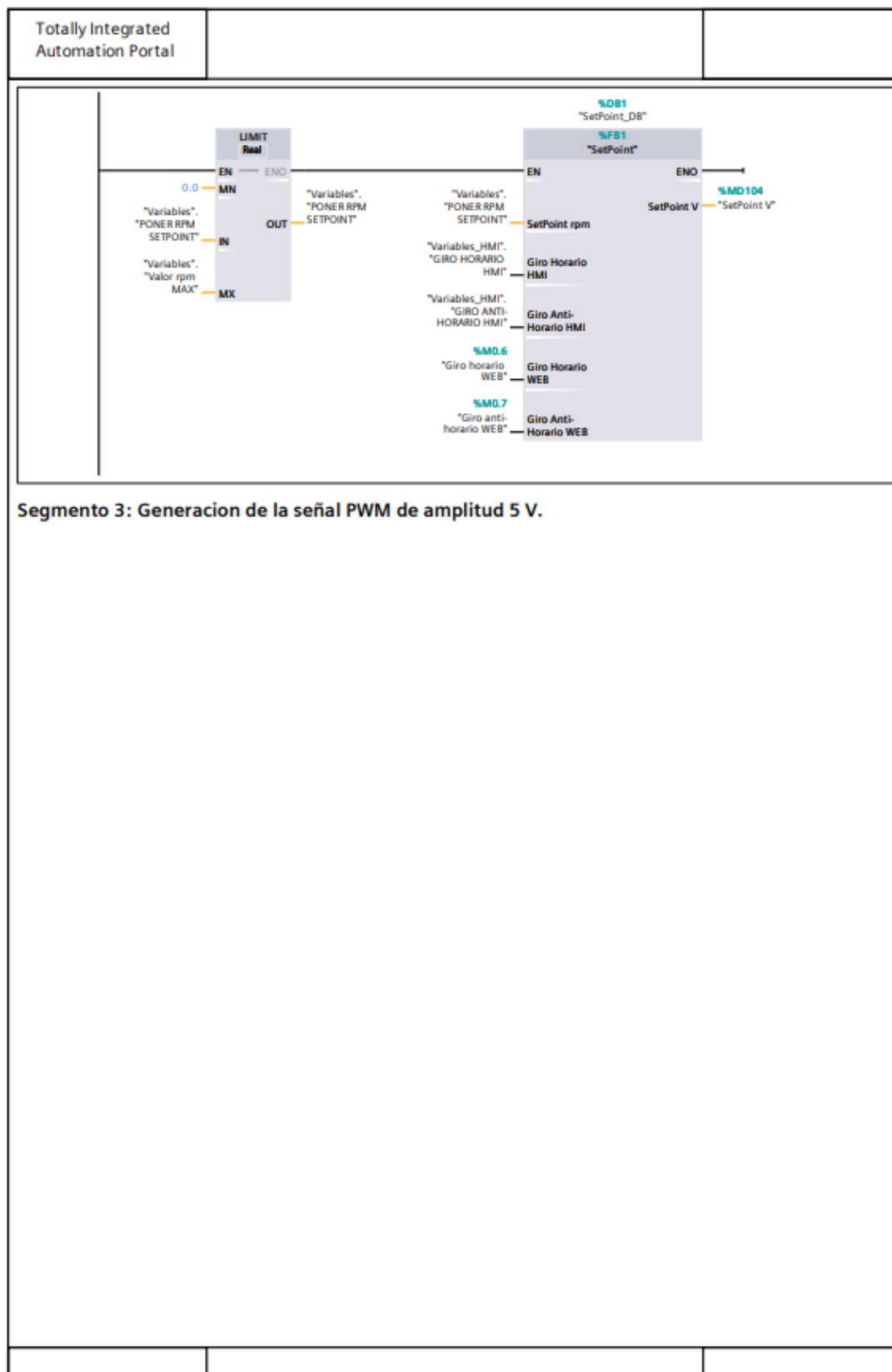
Título	"Main Program Sweep (Cycle)"	Autor		Comentario	
Familia		Versión	0.1	ID personalizada	

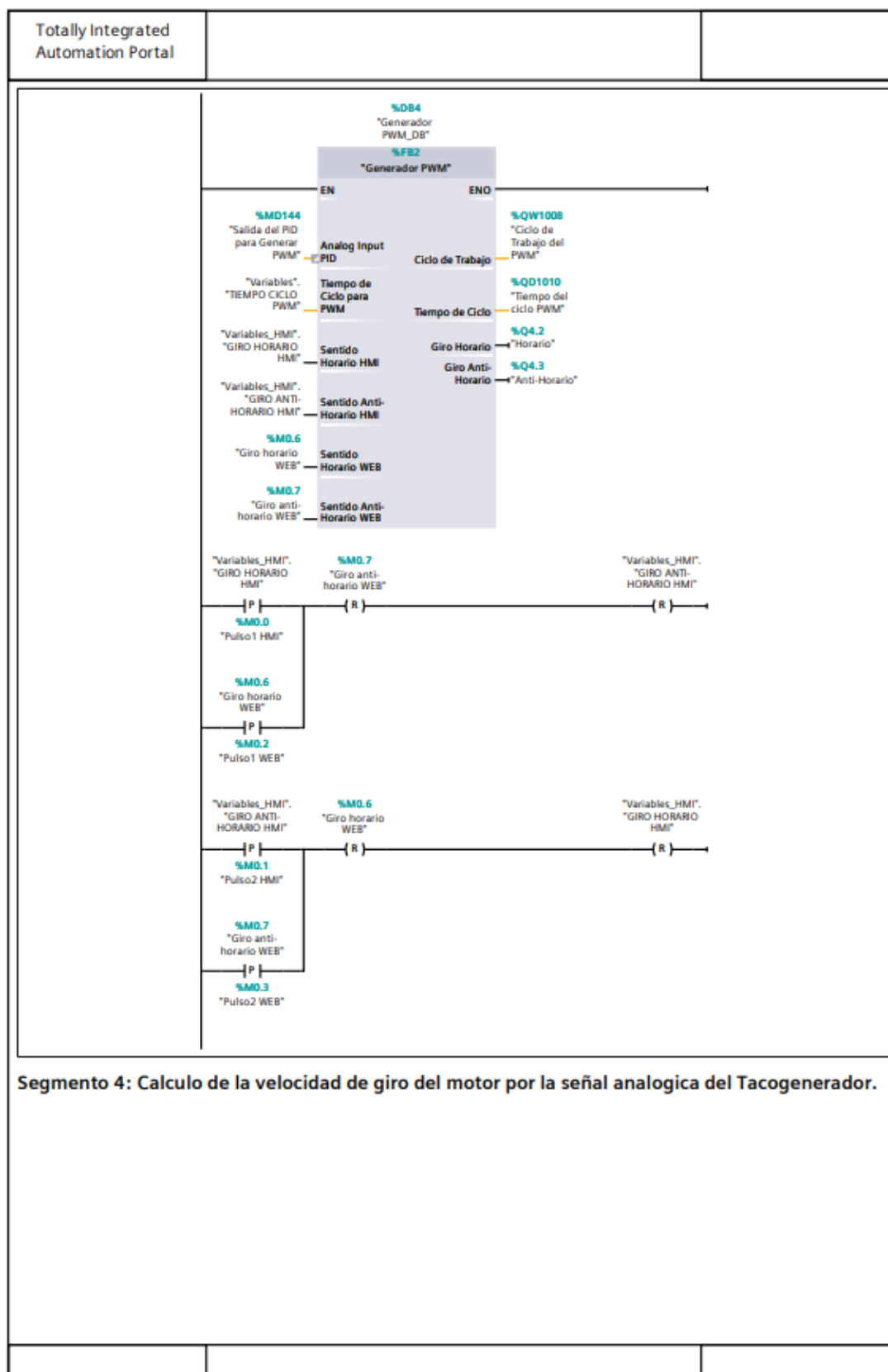
Main

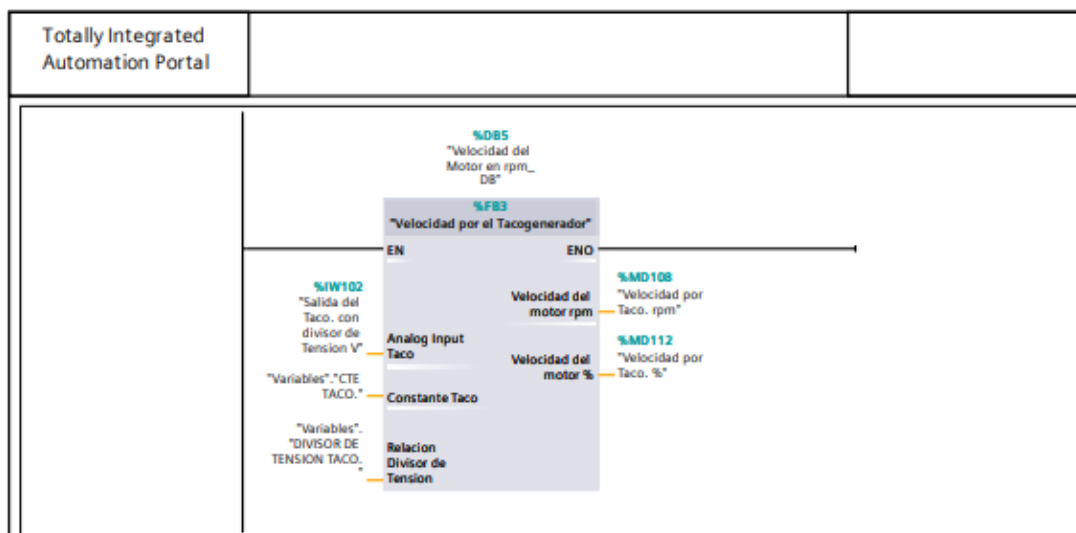
Nombre	Tipo de datos	Valor predet.	Supervisión	Comentario
▼ Input				
Initial_Call	Bool			Initial call of this OB
Remanence	Bool			=True, if remanent data are available
Temp				
Constant				

Segmento 1: Start and Stop.

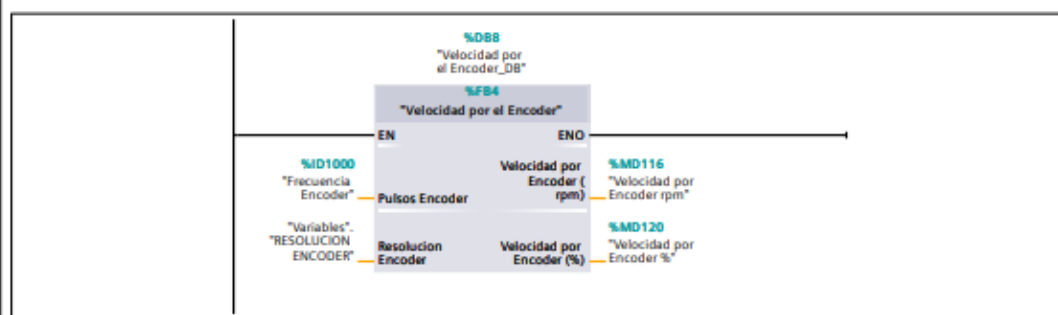
Segmento 2: SetPoint limitado de 0 al valor maximo en r.p.m..



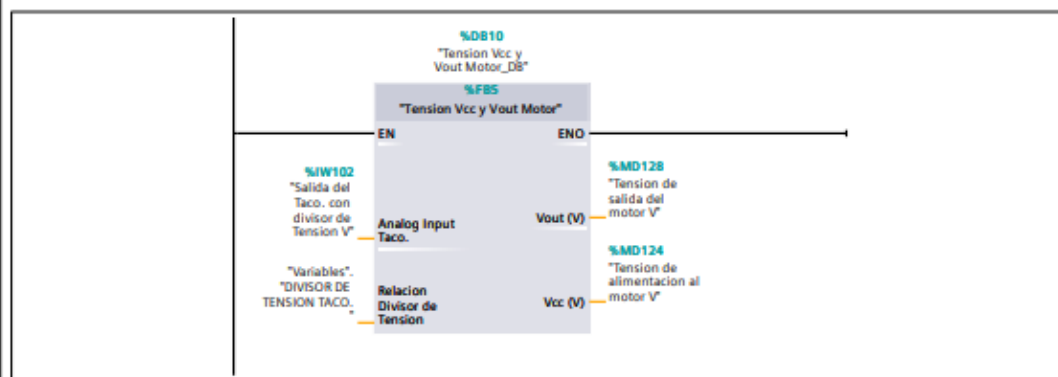




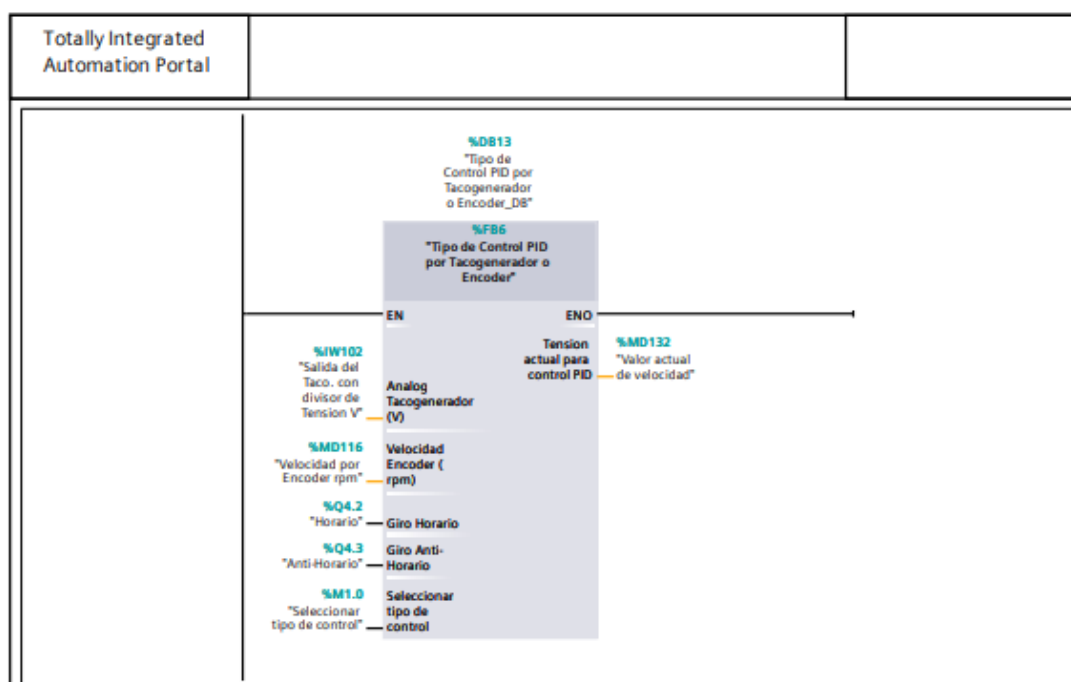
Segmento 5: Calculo de la velocidad de giro del motor por la señal digital del Encoder.



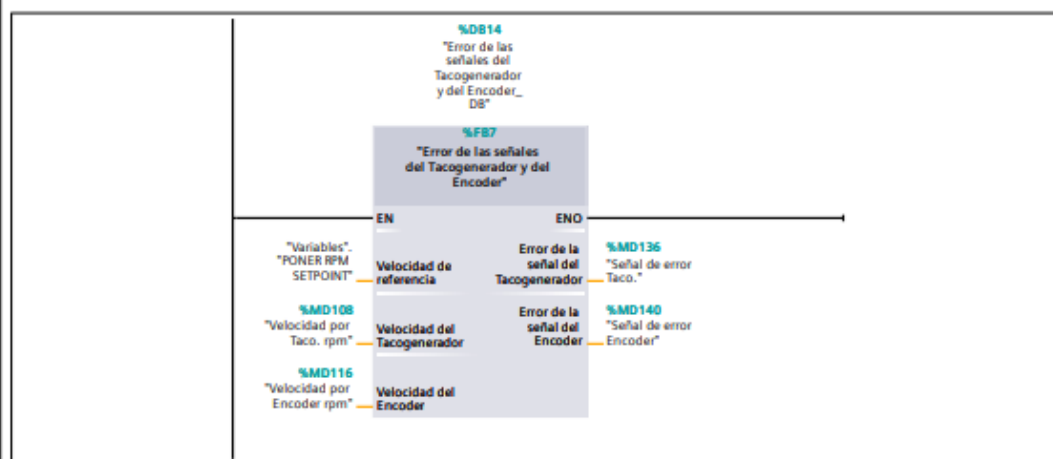
Segmento 6: Calculo de la tension de alimentacion y tension de salida del motor.



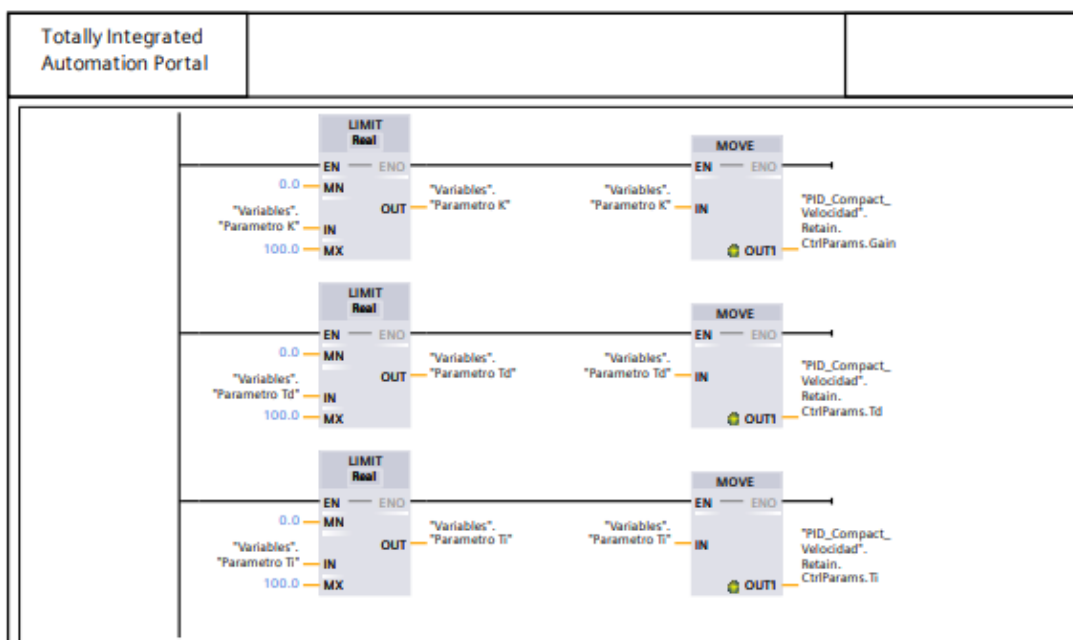
Segmento 7: Selector de tipo de control PID.



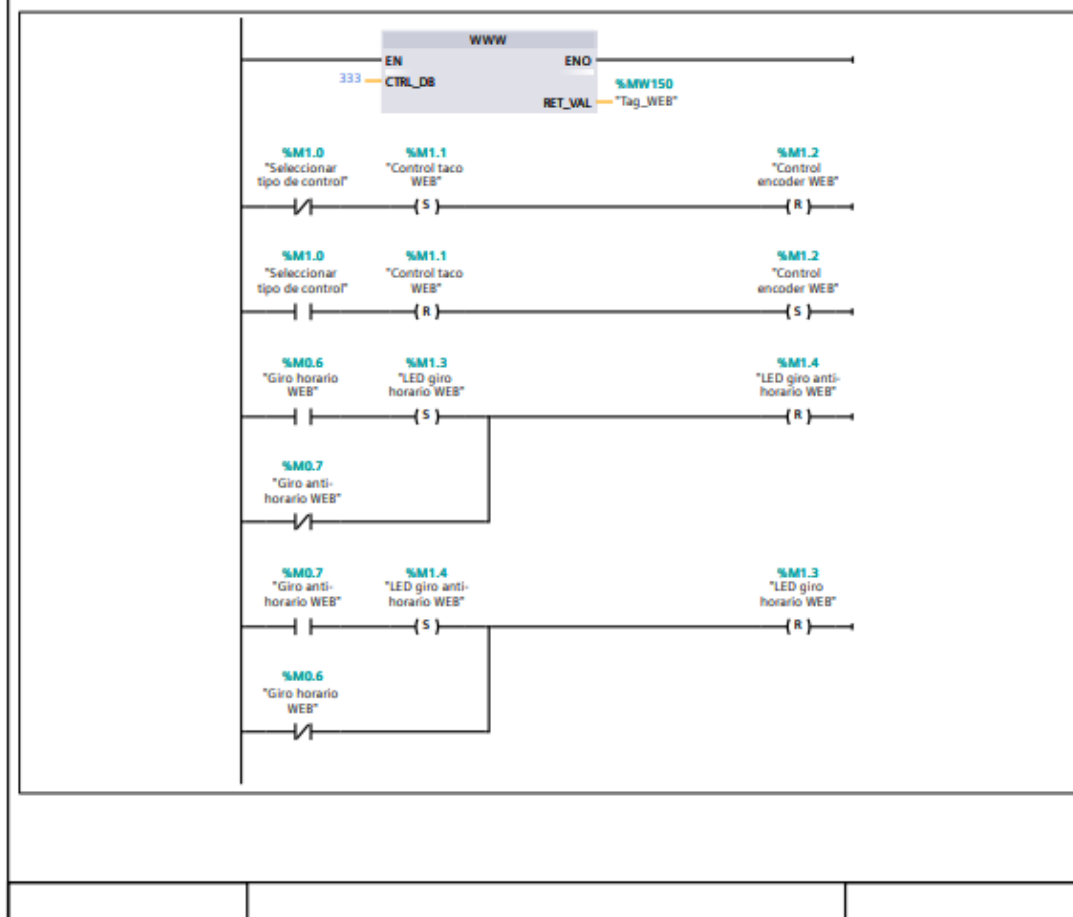
Segmento 8: Calculo de los errores de las señales



Segmento 9: Establecer los parametros PID.

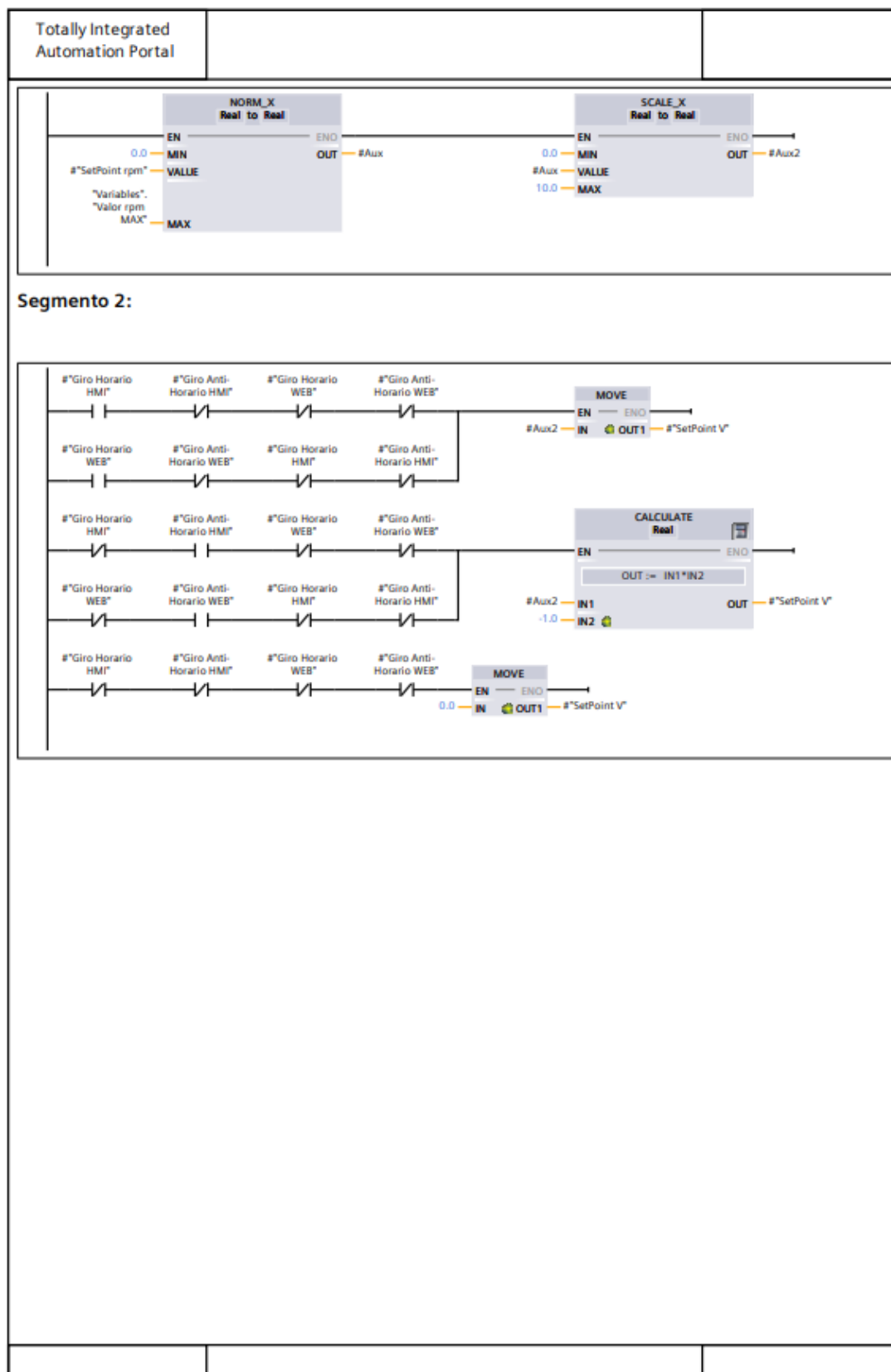


Segmento 10: Instruccion para el servidor web.



Totally Integrated Automation Portal		
Segmento 11: Aviso por pantalla HMI		
	<pre> graph LR MD108["%MD108 "Velocidad por Taco. rpm" 1300.0"] --> DB11_01["%DB11.DBX0.1 "Avisos". "Warning Tacogenerador""] MD116["%MD116 "Velocidad por Encoder rpm" 1300.0"] --> DB11_00["%DB11.DBX0.0 "Avisos". "Warning Encoder""] </pre>	

Totally Integrated Automation Portal									
Control_de_Velocidad_con_HMI_Control_TACO_ENCODER / PLC_1 [CPU 1212C AC/DC/Rly] / Bloques de programa									
SetPoint [FB1]									
SetPoint Propiedades									
General									
Nombre	SetPoint	Número	1	Tipo	FB				
Idioma	KOP	Numeración	Automático						
Información									
Título		Autor		Comentario					
Familia		Versión	0.1	ID personalizada					
SetPoint									
Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible desde HMI/OPC UA	Escribible desde HMI/OPC UA	Visible en HMI Engineering	Valor de ajuste	Supervisión	Comentario
▼ Input									
SetPoint rpm	Real	0.0	No remanente	True	True	True	False		
Giro Horario HMI	Bool	false	No remanente	True	True	True	False		
Giro Anti-Horario HMI	Bool	false	No remanente	True	True	True	False		
Giro Horario WEB	Bool	false	No remanente	True	True	True	False		
Giro Anti-Horario WEB	Bool	false	No remanente	True	True	True	False		
▼ Output									
SetPoint V	Real	0.0	No remanente	True	True	True	False		
InOut									
▼ Static									
Aux	Real	0.0	No remanente	True	True	True	False		
Aux2	Real	0.0	No remanente	True	True	True	False		
Temp									
Constant									
Segmento 1:									

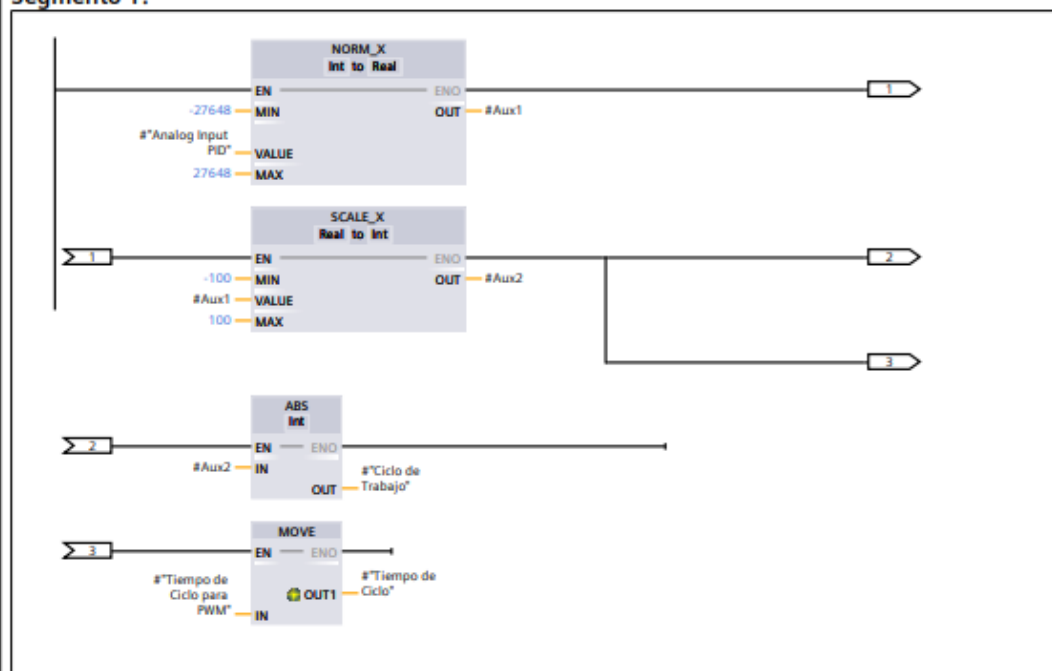


Totally Integrated Automation Portal									
Control de Velocidad con HMI Control TACO_ENCODER / PLC_1 [CPU 1212C AC/DC/Rly] / Bloques de programa									
Generador PWM [FB2]									
Generador PWM Propiedades									
General									
Nombre	Generador PWM	Número	2	Tipo	FB				
Idioma	KOP	Numeración	Automático						
Información									
Título		Autor		Comentario					
Familia		Versión	0.1	ID personalizada					
Generador PWM									
Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible desde HMI/OPC UA	Escriben desde HMI/OPC UA	Visible en HMI/Engineering	Valor de ajuste	Supervisión	Comentario
▼ Input									
Analog Input PID	Int	0	No remanente	True	True	True	False		
Tiempo de Ciclo para PWM	DWord	16#0	No remanente	True	True	True	False		
Sentido Horario HMI	Bool	false	No remanente	True	True	True	False		
Sentido Anti-Horario HMI	Bool	false	No remanente	True	True	True	False		
Sentido Horario WEB	Bool	false	No remanente	True	True	True	False		
Sentido Anti-Horario WEB	Bool	false	No remanente	True	True	True	False		
▼ Output									
Ciclo de Trabajo	Word	16#0	No remanente	True	True	True	False		
Tiempo de Ciclo	DWord	16#0	No remanente	True	True	True	False		
Giro Horario	Bool	false	No remanente	True	True	True	False		
Giro Anti-Horario	Bool	false	No remanente	True	True	True	False		
InOut									
▼ Static									
Aux1	Real	0.0	No remanente	True	True	True	False		
Aux2	Int	0	No remanente	True	True	True	False		
Aux3	Real	0.0	No remanente	True	True	True	False		
G.H.	Bool	false	No remanente	True	True	True	False		
G.AH.	Bool	false	No remanente	True	True	True	False		

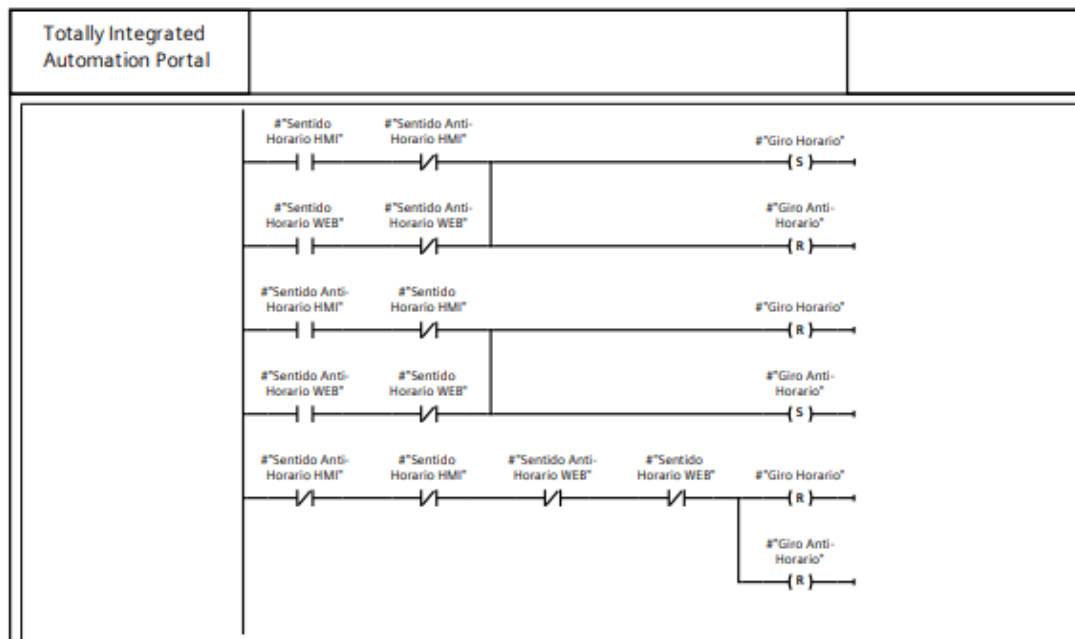
Totally Integrated Automation Portal										
Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible desde HMI/OPC UA	Escribible desde HMI/OPC UA	Visible desde HMI/Engineering	Valor de ajuste	Supervisión	Comentario	
Pulso1 HMI	Bool	false	No remanente	True	True	True	False			
Pulso1 WEB	Bool	false	No remanente	True	True	True	False			
Pulso2 HMI	Bool	false	No remanente	True	True	True	False			
Pulso2 WEB	Bool	false	No remanente	True	True	True	False			
Temp										
Constant										

Segmento 1:

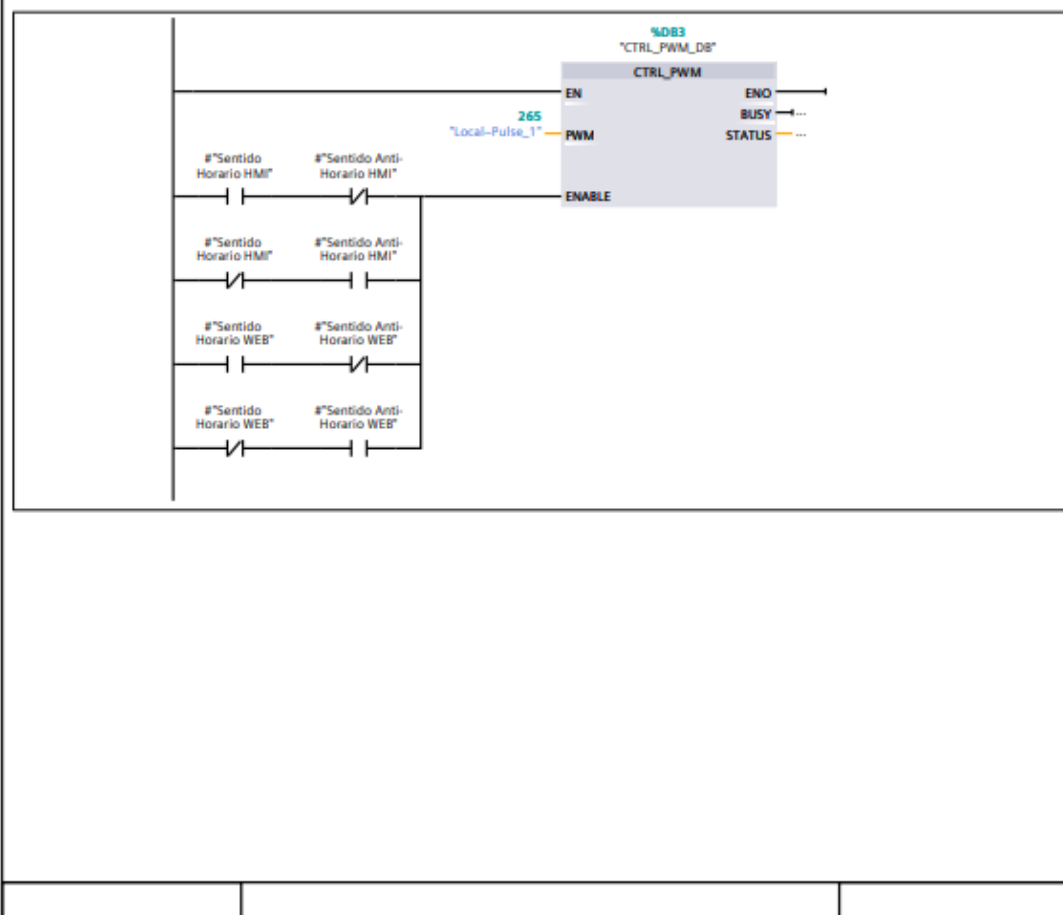
Segmento 1:



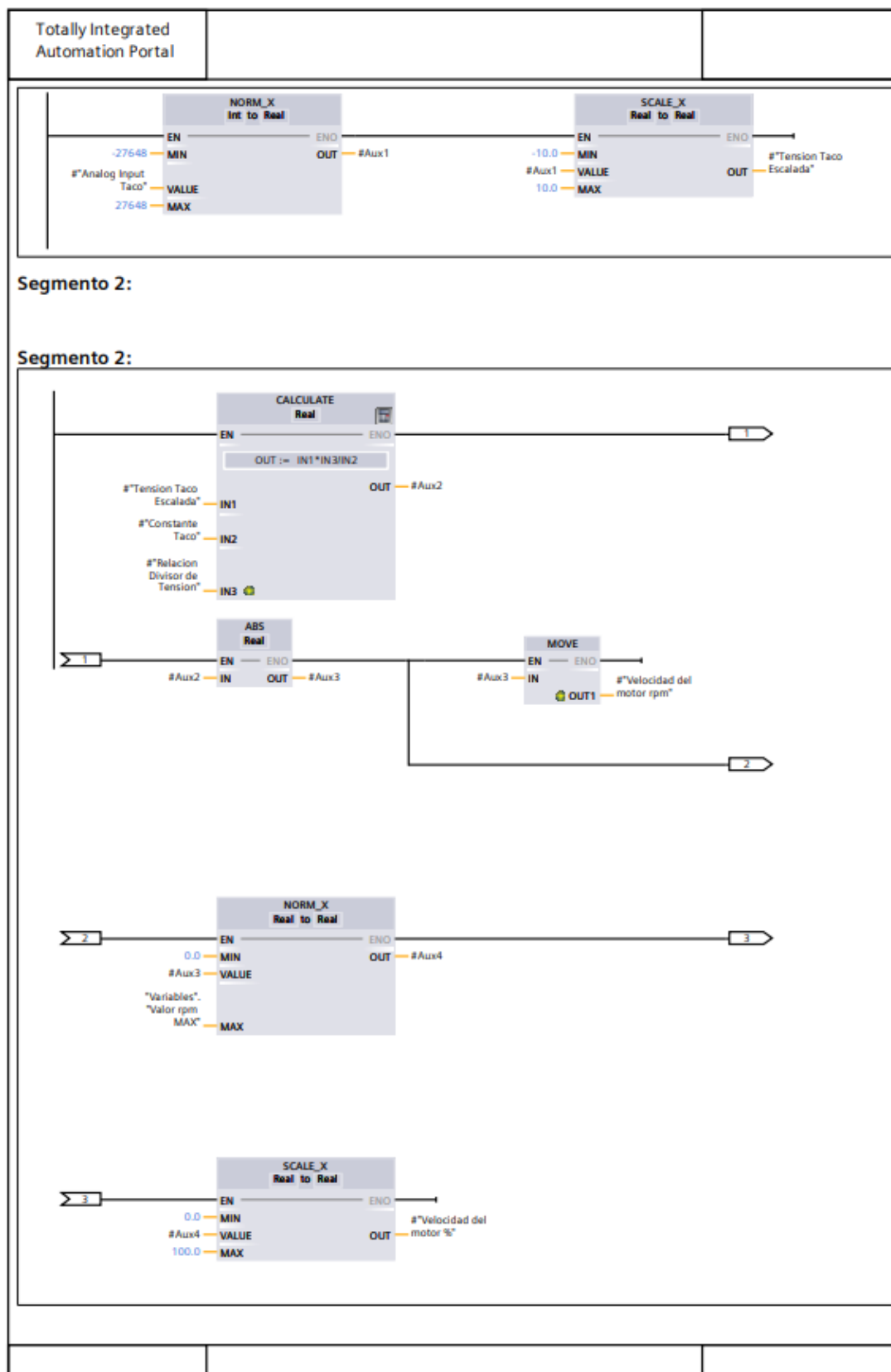
Segmento 2:



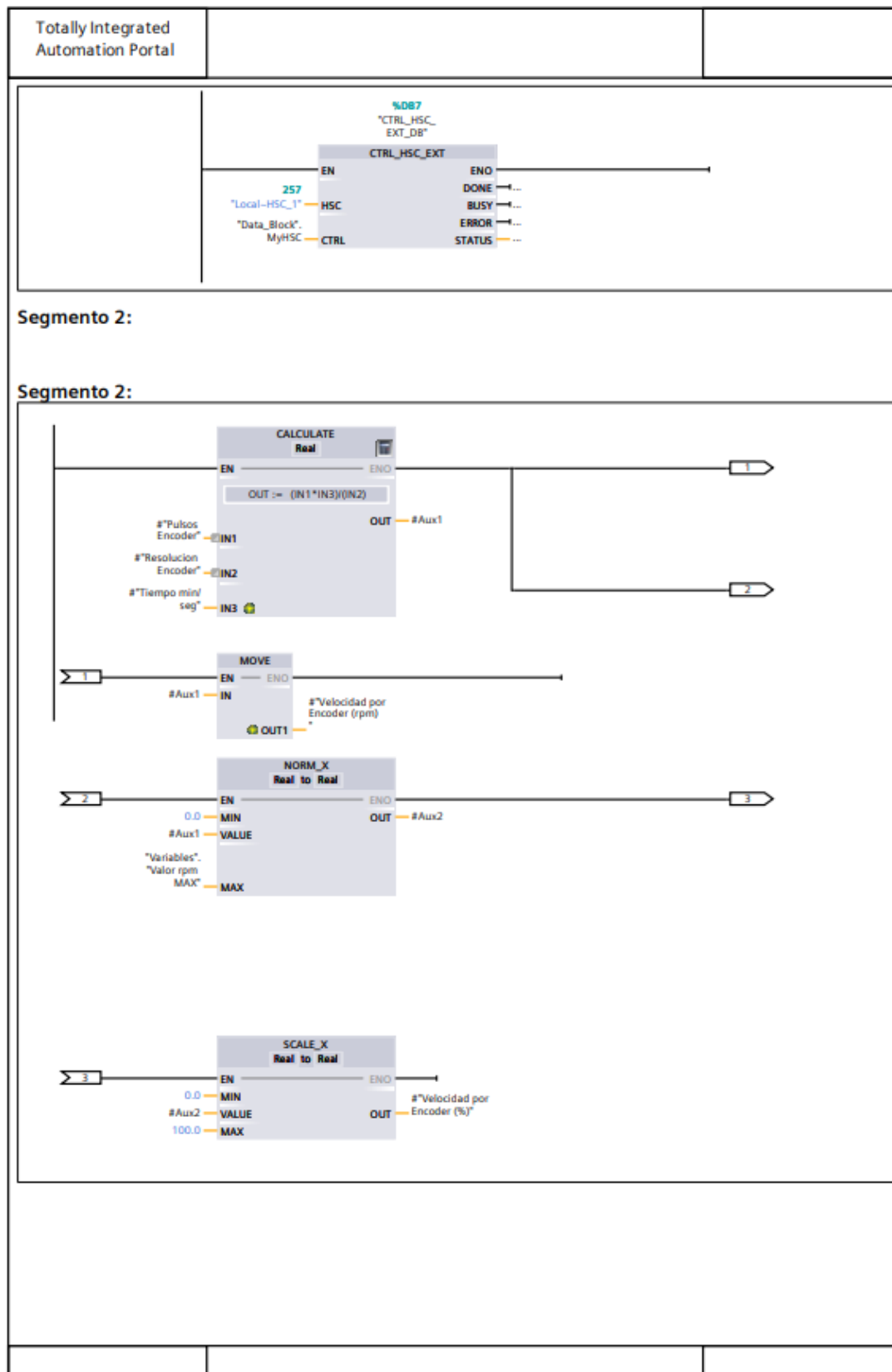
Segmento 3:



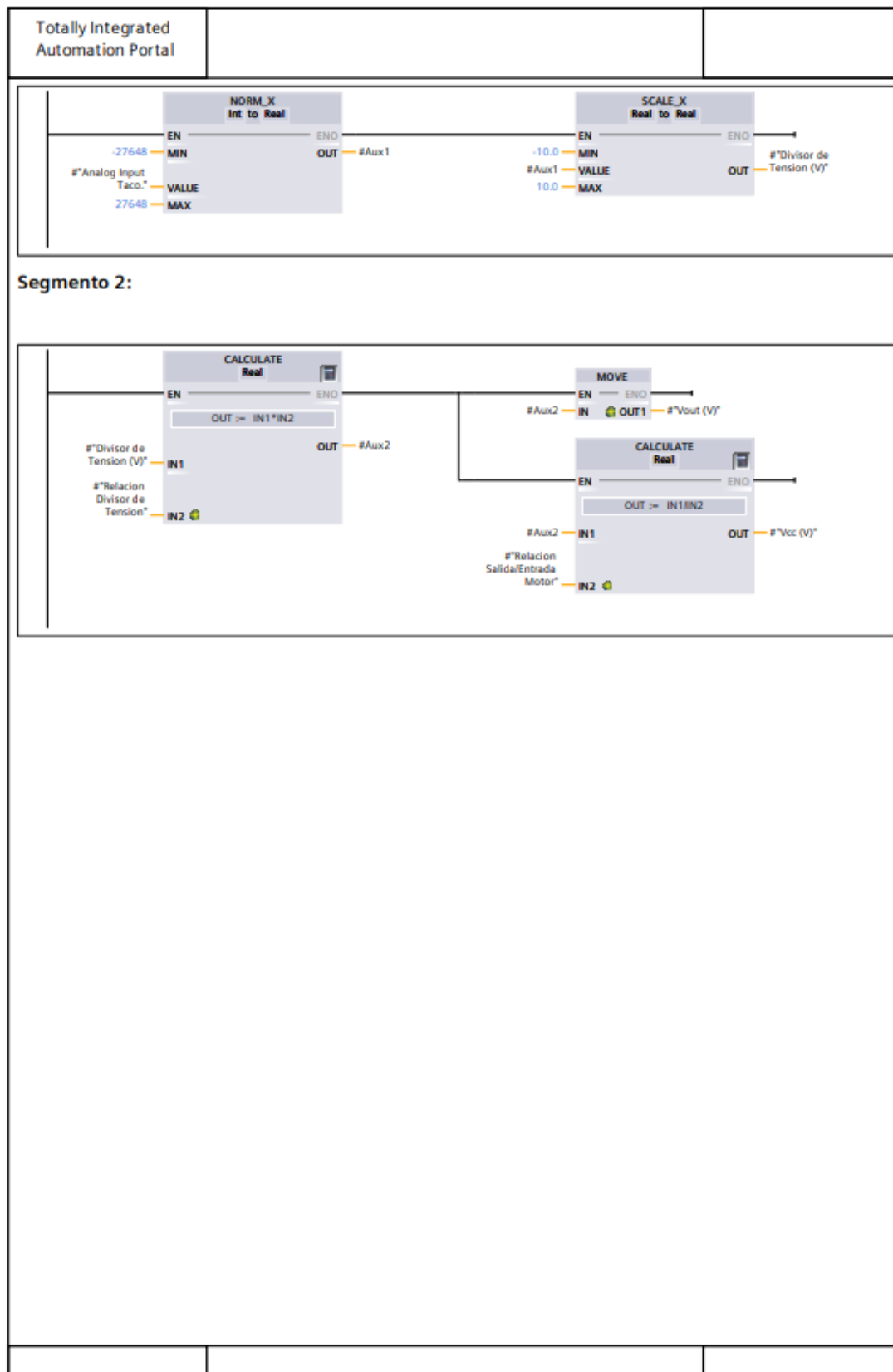
Totally Integrated Automation Portal									
Control de Velocidad con HMI Control TACO_ENCODER / PLC_1 [CPU 1212C AC/DC/Rly] / Bloques de programa									
Velocidad por el Tacogenerador [FB3]									
Velocidad por el Tacogenerador Propiedades									
General									
Nombre	Velocidad por el Tacogenerador	Número	3	Tipo	FB				
Idioma	KOP	Numeración	Automático						
Información									
Título		Autor		Comentario					
Familia		Versión	0.1	ID personalizada					
Velocidad por el Tacogenerador									
Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible desde HMI/OPC UA	Escribible desde HMI/OPC UA	Visible desde HMI Engineering	Valor de ajuste	Supervisión	Comentario
▼ Input									
Analog Input Taco	Word	16#0	No remanente	True	True	True	False		
Constante Taco	Real	0.0	No remanente	True	True	True	False		
Relacion Divisor de Tension	Real	0.0	No remanente	True	True	True	False		
▼ Output									
Velocidad del motor rpm	Real	0.0	No remanente	True	True	True	False		
Velocidad del motor %	Real	0.0	No remanente	True	True	True	False		
InOut									
▼ Static									
Aux1	Real	0.0	No remanente	True	True	True	False		
Tension Taco Escalada	Real	0.0	No remanente	True	True	True	False		
Aux2	Real	0.0	No remanente	True	True	True	False		
Aux3	Real	0.0	No remanente	True	True	True	False		
Aux4	Real	0.0	No remanente	True	True	True	False		
Temp									
Constant									
Segmento 1:									



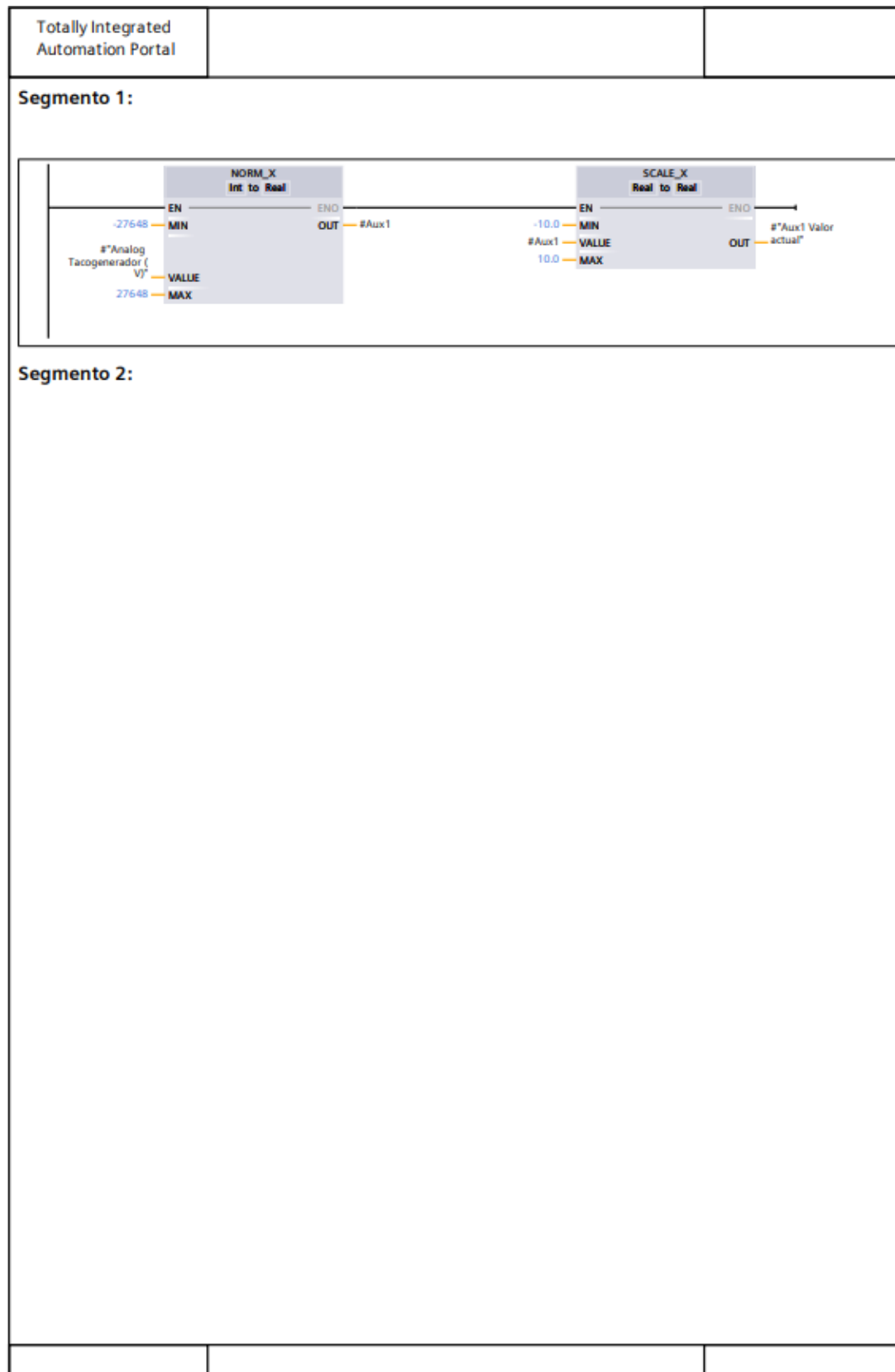
Totally Integrated Automation Portal									
Control_de_Velocidad_con_HMI_Control_TACO_ENCODER / PLC_1 [CPU 1212C AC/DC/Rly] / Bloques de programa									
Velocidad por el Encoder [FB4]									
Velocidad por el Encoder Propiedades									
General									
Nombre	Velocidad por el Encoder	Número	4	Tipo	FB				
Idioma	KOP	Numeración	Automático						
Información									
Título		Autor		Comentario					
Familia		Versión	0.1	ID personalizada					
Velocidad por el Encoder									
Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible desde HMI/OPC UA	Escribible desde HMI/OPC UA	Visible en HMI Engineering	Valor de ajuste	Supervisión	Comentario
▼ Input									
Pulsos Encoder	DInt	0	No remanente	True	True	True	False		
Resolucion Encoder	DInt	0	No remanente	True	True	True	False		
▼ Output									
Velocidad por Encoder (rpm)	Real	0.0	No remanente	True	True	True	False		
Velocidad por Encoder (%)	Real	0.0	No remanente	True	True	True	False		
InOut									
▼ Static									
Tiempo min/seg	Real	60.0	No remanente	True	True	True	False		
Aux1	Real	0.0	No remanente	True	True	True	False		
Aux2	Real	0.0	No remanente	True	True	True	False		
Temp									
Constant									
Segmento 1:									

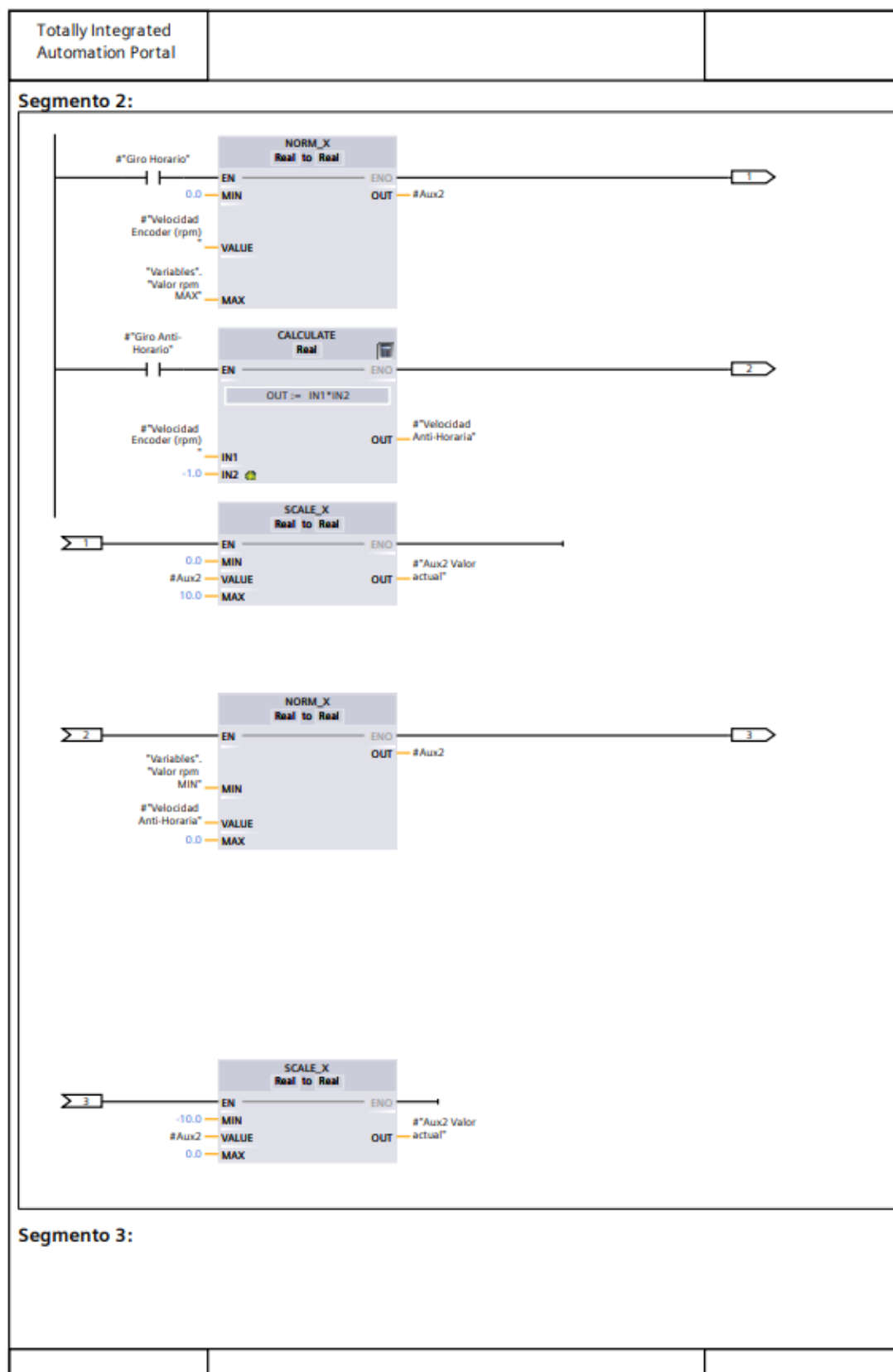


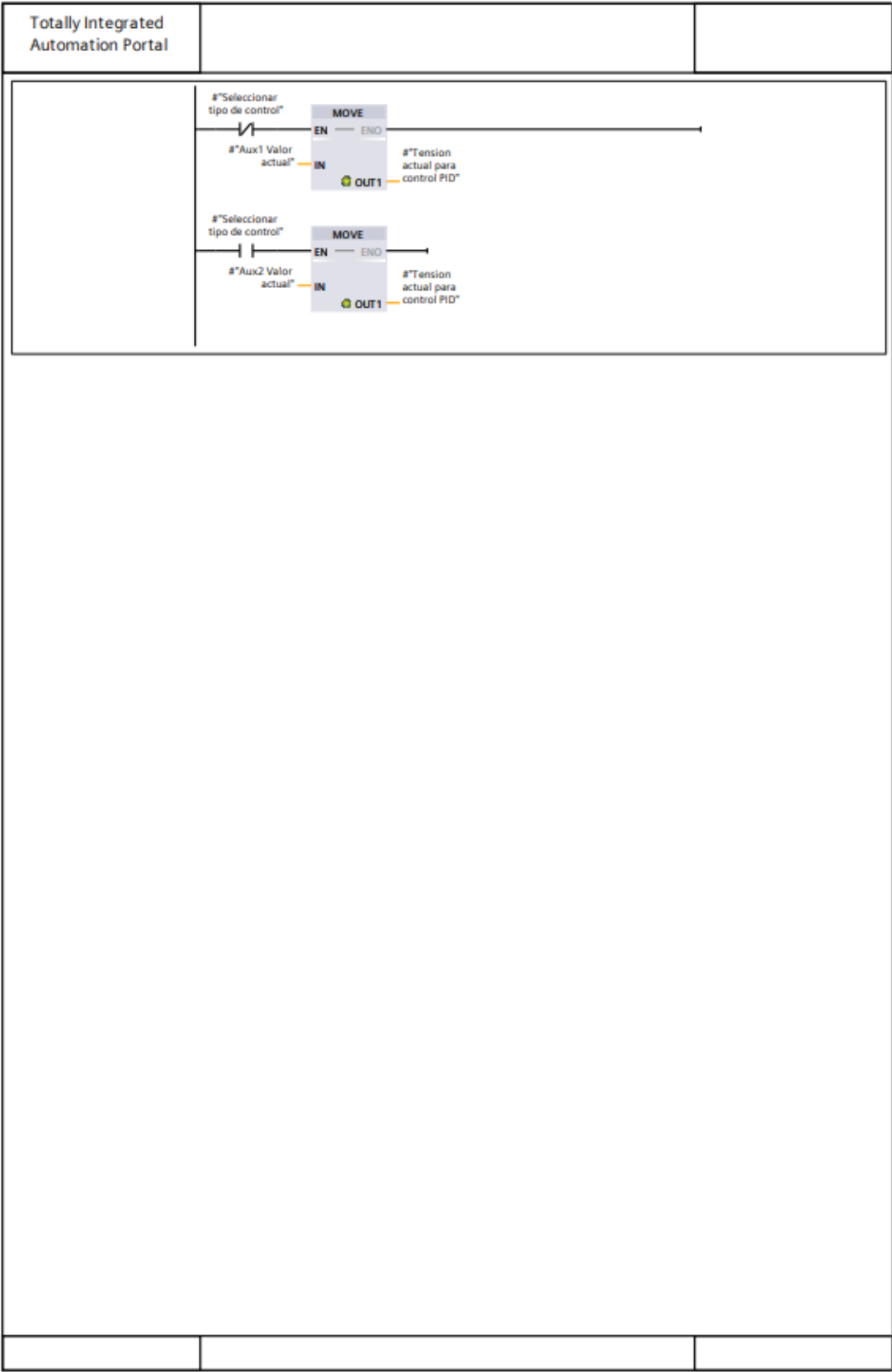
Totally Integrated Automation Portal									
Control_de_Velocidad_con_HMI_Control_TACO_ENCODER / PLC_1 [CPU 1212C AC/DC/Rly] / Bloques de programa									
Tension Vcc y Vout Motor [FB5]									
Tension Vcc y Vout Motor Propiedades									
General									
Nombre	Tension Vcc y Vout Motor	Número	5	Tipo	FB				
Idioma	KOP	Numeración	Automático						
Información									
Título		Autor		Comentario					
Familia		Versión	0.1	ID personalizada					
Tension Vcc y Vout Motor									
Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible desde HMI/OPC UA	Escribible desde HMI/OPC UA	Visible en HMI Engineering	Valor de ajuste	Supervisión	Comentario
▼ Input									
Analog Input Tacho.	Word	16#0	No remanente	True	True	True	False		
Relacion Divisor de Tension	Real	0.0	No remanente	True	True	True	False		
▼ Output									
Vout (V)	Real	0.0	No remanente	True	True	True	False		
Vcc (V)	Real	0.0	No remanente	True	True	True	False		
InOut									
▼ Static									
Aux1	Real	0.0	No remanente	True	True	True	False		
Divisor de Tension (V)	Real	0.0	No remanente	True	True	True	False		
Aux2	Real	0.0	No remanente	True	True	True	False		
Relacion Salida/Entrada Motor	Real	2.0	No remanente	True	True	True	False		
Temp									
Constant									
Segmento 1:									



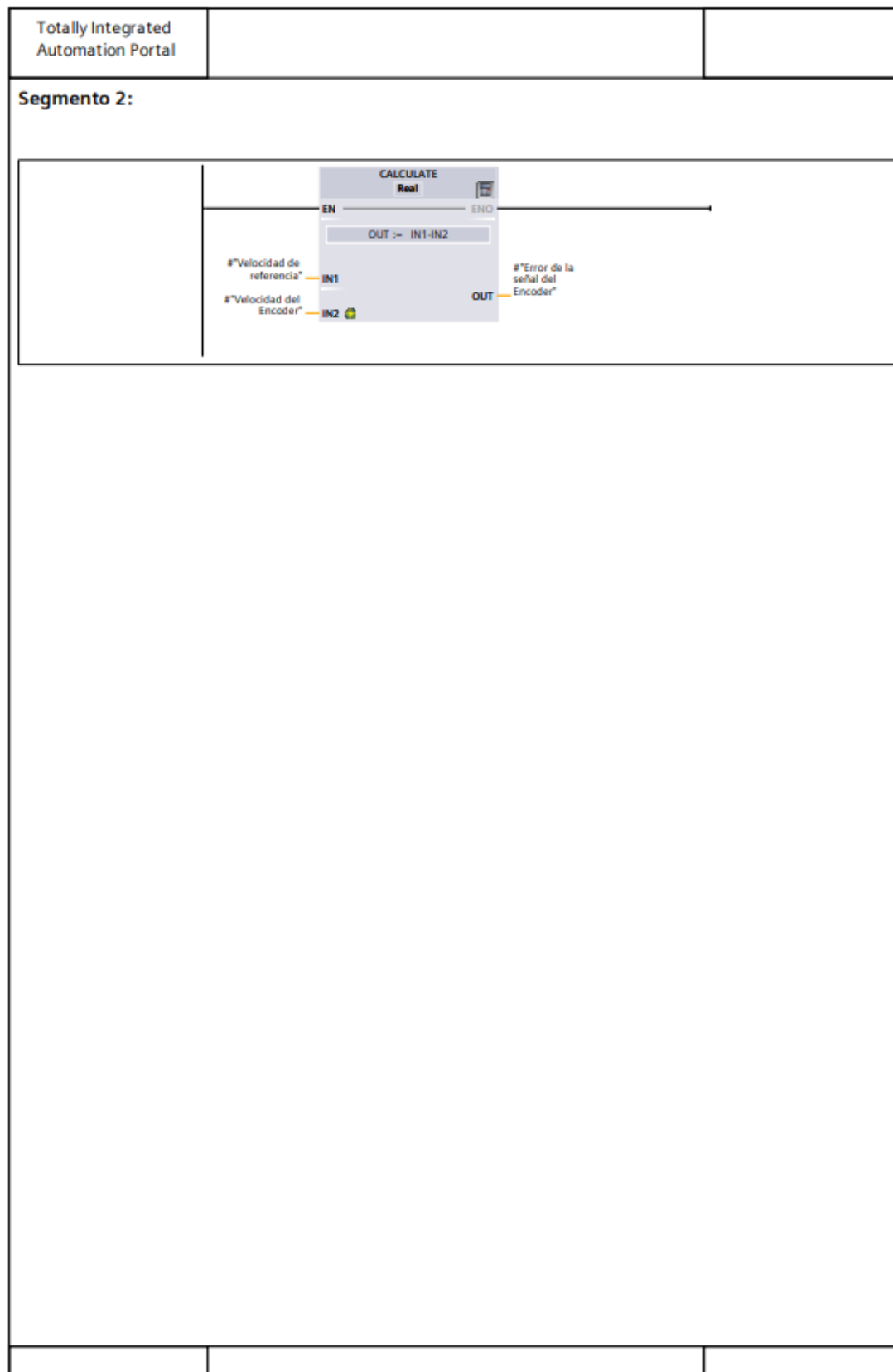
Totally Integrated Automation Portal									
Control_de_Velocidad_con_HMI_Control_TACO_ENCODER / PLC_1 [CPU 1212C AC/DC/Rly] / Bloques de programa									
Tipo de Control PID por Tacogenerador o Encoder [FB6]									
Tipo de Control PID por Tacogenerador o Encoder Propiedades									
General									
Nombre	Tipo de Control PID por Tacogenerador o Encoder	Número	6	Tipo	FB				
Idioma	KOP	Numeración	Automático						
Información									
Título		Autor		Comentario					
Familia		Versión	0.1	ID personalizada					
Tipo de Control PID por Tacogenerador o Encoder									
Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible desde HMI/OPC UA	Escribible desde HMI/OPC UA	Visible desde HMI/OPC UA	Valor de ajuste	Supervisión	Comentario
▼ Input									
Analog Tacogenerador (V)	Word	16#0	No remanente	True	True	True	False		
Velocidad Encoder (rpm)	Real	0.0	No remanente	True	True	True	False		
Giro Horario	Bool	false	No remanente	True	True	True	False		
Giro Anti-Horario	Bool	false	No remanente	True	True	True	False		
Seleccionar tipo de control	Bool	false	No remanente	True	True	True	False		
▼ Output									
Tension actual para control PID	Real	0.0	No remanente	True	True	True	False		
InOut									
▼ Static									
Aux1	Real	0.0	No remanente	True	True	True	False		
Aux1 Valor actual	Real	0.0	No remanente	True	True	True	False		
Aux2	Real	0.0	No remanente	True	True	True	False		
Velocidad Anti-Horaria	Real	0.0	No remanente	True	True	True	False		
Aux2 Valor actual	Real	0.0	No remanente	True	True	True	False		
Resolucion Encoder	Real	30.0	No remanente	True	True	True	False		
Tiempo min/seg	Real	60.0	No remanente	True	True	True	False		
Temp									
Constant									







Totally Integrated Automation Portal									
Control_de_Velocidad_con_HMI_Control_TACO_ENCODER / PLC_1 [CPU 1212C AC/DC/Rly] / Bloques de programa									
Error de las señales del Tacogenerador y del Encoder [FB7]									
Error de las señales del Tacogenerador y del Encoder Propiedades									
General									
Nombre	Error de las señales del Tacogenerador y del Encoder	Número	7	Tipo	FB				
Idioma	KOP	Numeración	Automático						
Información									
Título		Autor		Comentario					
Familia		Versión	0.1	ID personalizada					
Error de las señales del Tacogenerador y del Encoder									
Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible desde HMI/OPC UA	Escribible desde HMI/OPC UA	Visible en HMI Engineering	Valor de ajuste	Supervisión	Comentario
▼ Input									
Velocidad de referencia	Real	0.0	No remanente	True	True	True	False		
Velocidad del Tacogenerador	Real	0.0	No remanente	True	True	True	False		
Velocidad del Encoder	Real	0.0	No remanente	True	True	True	False		
▼ Output									
Error de la señal del Tacogenerador	Real	0.0	No remanente	True	True	True	False		
Error de la señal del Encoder	Real	0.0	No remanente	True	True	True	False		
InOut									
Static									
Temp									
Constant									
Segmento 1:									
<pre> graph LR EN[EN] --> CALCULATE[CALCULATE Real] IN1[IN1] --> CALCULATE IN2[IN2] --> CALCULATE CALCULATE --> ENO[ENO] OUT[OUT] --> OUT_VAR[OUT] IN1_VAR[IN1] --> IN1_LABEL["#Velocidad de referencia"] IN2_VAR[IN2] --> IN2_LABEL["#Velocidad del Tacogenerador"] OUT_VAR --> OUT_LABEL["#Error de la señal del Tacogenerador"] </pre>									



Totally Integrated Automation Portal					
--------------------------------------	--	--	--	--	--

Control de Velocidad con HMI_Control_TACO_ENCODER / PLC_1 [CPU 1212C AC/DC/Rly] / Bloques de programa

Cyclic interrupt PID [OB30]

Cyclic interrupt PID Propiedades					
General					
Nombre	Cyclic interrupt PID	Número	30	Tipo	OB
Idioma	KOP	Numeración	Automático		
Información					
Título		Autor		Comentario	
Familia		Versión	0.1	ID personalizada	

Cyclic interrupt PID				
Nombre	Tipo de datos	Valor predet.	Supervisión	Comentario
▼ Input				
Initial_Call	Bool			Initial call of this OB
Event_Count	Int			Events discarded
Temp				
Constant				

Segmento 1: Regulador PID.

8.2 TABLA DE VARIABLES DEL TIA PORTAL.

Totally Integrated Automation Portal									
Control_de_Velocidad_con_HMI_Control_TACO_ENCODER / PLC_1 [CPU 1212C AC/DC/Rly]									
Variables PLC									
	Nombre	Tipo de datos	Dirección	Remanencia	Accesible desde HMI/O PC UA	Escribible desde HMI/O PC UA	Visible en HMI Engineering	Supervisión	Comentario
	SetPoint V	Real	%MD104	False	True	True	True		
	Salida del Taco. con divisor de Tension V	Word	%IW102	False	True	True	True		
	Salida del PID para Generar PWM	Real	%MD144	False	True	True	True		
	Ciclo de Trabajo del PWM	Word	%QW1008	False	True	True	True		
	Tiempo del ciclo PWM	DWord	%QD1010	False	True	True	True		
	Horario	Bool	%Q4.2	False	True	True	True		
	Anti-Horario	Bool	%Q4.3	False	True	True	True		
	Velocidad por Taco. rpm	Real	%MD108	False	True	True	True		
	Velocidad por Taco. %	Real	%MD112	False	True	True	True		
	Frecuencia Encoder	DWord	%ID1000	False	True	True	True		
	Velocidad por Encoder rpm	Real	%MD116	False	True	True	True		
	Velocidad por Encoder %	Real	%MD120	False	True	True	True		
	Tension de alimentacion al motor V	Real	%MD124	False	True	True	True		
	Pulso1 HMI	Bool	%M0.0	False	True	True	True		
	Pulso2 HMI	Bool	%M0.1	False	True	True	True		
	Tension de salida del motor V	Real	%MD128	False	True	True	True		
	Pulso1 WEB	Bool	%M0.2	False	True	True	True		
	Pulso2 WEB	Bool	%M0.3	False	True	True	True		
	Valor actual de velocidad	Real	%MD132	False	True	True	True		
	Seleccionar tipo de control	Bool	%M1.0	False	True	True	True		
	Señal de error Taco.	Real	%MD136	False	True	True	True		
	Señal de error Encoder	Real	%MD140	False	True	True	True		
	Tag_WEB	Word	%MW150	False	True	True	True		
	Control taco WEB	Bool	%M1.1	False	True	True	True		
	Control encoder WEB	Bool	%M1.2	False	True	True	True		
	Arrancar WEB	Bool	%M0.4	False	True	True	True		
	Parar WEB	Bool	%M0.5	False	True	True	True		
	Giro horario WEB	Bool	%M0.6	False	True	True	True		
	Giro anti-horario WEB	Bool	%M0.7	False	True	True	True		

Nombre		Tipo de datos	Dirección	Remanencia	Accesible desde HMI/O PC UA	Escribible desde HMI/O PC UA	Visible en HMI Engineering	Supervisión	Comentario
3	LED giro horario WEB	Bool	%M1.3	False	True	True	True		
3	LED giro anti-horario WEB	Bool	%M1.4	False	True	True	True		

8.3 CÓDIGO HTML DE LA PÁGINA WEB.

```

1  <!DOCTYPE html>
2  <!-- INTRODUCCION DE LOS DATOS PARA EL MOTOR DC -->
3  <!-- INTRODUCCION DE LOS DATOS PARA EL MOTOR DC -->
4  <!-- INTRODUCCION DE LOS DATOS PARA EL MOTOR DC -->
5  <!-- AWP_In_Variable Name="Arrancar WEB" -->
6  <!-- AWP_In_Variable Name="Parar WEB" -->
7  <!-- AWP_In_Variable Name="Variables"."PONER RPM SETPOINT" -->
8  <!-- AWP_In_Variable Name="Seleccionar tipo de control" -->
9  <!-- AWP_In_Variable Name="Giro horario WEB" -->
10 <!-- AWP_In_Variable Name="Giro anti-horario WEB" -->
11 <!-- AWP_In_Variable Name="Variables"."Parametro K" -->
12 <!-- AWP_In_Variable Name="Variables"."Parametro Td" -->
13 <!-- AWP_In_Variable Name="Variables"."Parametro Ti" -->
14 <!-- AWP_In_Variable Name="Variables"."Parametro Ti" -->
15
16 <!-- ESTADO DE LOS DATOS INTRODUCIDOS AL MOTOR DC -->
17 <!-- ESTADO DE LOS DATOS INTRODUCIDOS AL MOTOR DC -->
18 <!-- ESTADO DE LOS DATOS INTRODUCIDOS AL MOTOR DC -->
19 <!-- AWP_In_Variable Name="Variables"."ACTIVAR PID" -->
20 <!-- AWP_In_Variable Name="Control taco WEB" -->
21 <!-- AWP_In_Variable Name="Control encoder WEB" -->
22 <!-- AWP_In_Variable Name="LED giro anti-horario WEB" -->
23 <!-- AWP_In_Variable Name="LED giro horario WEB" -->
24 <!-- AWP_In_Variable Name="LED giro horario WEB" -->
25
26 <!-- ESTADO DE LAS SEÑALES DEL MOTOR DC -->
27 <!-- ESTADO DE LAS SEÑALES DEL MOTOR DC -->
28 <!-- ESTADO DE LAS SEÑALES DEL MOTOR DC -->
29 <!-- AWP_In_Variable Name="Velocidad por Encoder rpm" -->
30 <!-- AWP_In_Variable Name="Velocidad por Encoder rpm" -->
31 <!-- AWP_In_Variable Name="Velocidad por Taco. rpm" -->
32 <!-- AWP_In_Variable Name="Velocidad por Taco. rpm" -->
33 <!-- AWP_In_Variable Name="Señal de error Taco." -->
34 <!-- AWP_In_Variable Name="Señal de error Encoder" -->
35 <!-- AWP_In_Variable Name="Tension de alimentacion al motor V" -->
36 <!-- AWP_In_Variable Name="Tension de salida del motor V" -->
37 <!-- AWP_In_Variable Name="Tension de salida del motor V" -->
38
39 <!-- LISTADOS PARA OBSERVAR LAS IMAGENES DE ACTIVADO -->
40 <!-- LISTADOS PARA OBSERVAR LAS IMAGENES DE ACTIVADO -->
41 <!-- LISTADOS PARA OBSERVAR LAS IMAGENES DE ACTIVADO -->
42 <!-- AWP_Enum_Def Name="listado1" Values="0:"Desactivado", 1:"Activado" -->
43 <!-- AWP_Enum_Def Name="listado2" Values="0:"Desactivado", 1:"Activado" -->
44 <!-- AWP_Enum_Def Name="listado3" Values="0:"Desactivado", 1:"Activado" -->
45 <!-- AWP_Enum_Def Name="listado4" Values="0:"Desactivado", 1:"Activado" -->
46 <!-- AWP_Enum_Def Name="listado5" Values="0:"Desactivado", 1:"Activado" -->
47 <!-- AWP_Enum_Def Name="listado5" Values="0:"Desactivado", 1:"Activado" -->
48
49 <html>
50 <head>
51 <!-- Establecer todos los archivos que se va a usar -->
52 <!-- Idioma y tipo de texto -->
53 <meta charset="UTF-8">
54 <meta name="viewport" content="width=device-width, initial-scale=1.0">
55
56 <!-- llamar al .css estilos -->
57 <link rel="stylesheet" href="css/estilos.css"/>
58
59 <!-- Habilitar JavaScript -->
60 <meta http-equiv="content-script-type" content="text/javascript">
61 <script type="text/javascript" src="jQuery_min.js"></script>
62 <!-- Otra alternativa es poniendo: -->
63 <script type="text/javascript" src="http://code.jquery.com/jquery-latest.min.js"></script> -->
64
65 <!-- Script para tiempo real -->
66 <script type="text/javascript">
67
68 $(document).ready(function() {
69
70     // Actualizar el contador a tiempos fijos
71     setInterval(actualizar,500);
72
73     // Enviar informacion al pulsar los botones sin recargar pagina.
74     $.ajaxStart(function() {
75         $('#loading').show();
76         $('#result').hide();
77     }).ajaxStop(function() {
78         $('#loading').hide();
79         $('#result').fadeIn('fast');
80     });
81
82     $('#T1, #T2, #T3, #T4, #T5, #T6, #T7, #T8, #T9').submit(function() {
83         $.ajax({
84             type: 'POST',
85             url: $(this).attr('action'),
86             data: $(this).serialize(),
87             success: function(data) {
88                 $('#result').html(data);
89             }
90         })
91         return false;
92     });
93
94 });

```

```

95     function actualizar() {
96         $("#signal").load("signal.htm");
97         $("#output").load("output.htm");
98     }
99
100
101
102     <title>Control de Velocidad de Motor DC</title>
103 </head>
104
105 <body background="FondoPLC.jpg" style="background-size:cover;background-position:left top">
106
107     <!-- Titulo del proyecto -->
108     <div id="header">
109         <h1><b>CONTROL DE VELOCIDAD DE MOTOR DC CON PLC</b></h1>
110     </div>
111
112     <br>
113     <br>
114
115     <!-- CREACION DE LA TABLA DE CONTROL -->
116     <table>
117         <!-- Titulo -->
118         <thead>
119             <tr>
120                 <th colspan="3">INTRODUCIR LOS DATOS PARA EL MOTOR DC</th>
121             </tr>
122         </thead>
123
124         <!-- Pulsador de Start y Stop -->
125         <tr>
126             <td>Arrancar y Parar:</td>
127             <td>
128                 <form method="post" action="" id="T1" name="T1">
129                     <center>
130                         <input type="submit" name="Arrancar WEB" value="Start" class="boton">
131                     </center>
132                     <input type="hidden" name="Arrancar WEB" value="1">
133                     <div id="delay"></div>
134                     <input type="hidden" name="Arrancar WEB" value="0">
135                 </form>
136             </td>
137             <td>
138                 <form method="post" action="" id="T2" name="T2">
139                     <center>
140                         <input type="submit" name="Parar WEB" value="Stop" class="boton">
141                     </center>
142
143                     <input type="hidden" name="Parar WEB" value="1">
144                     <div id="delay"></div>
145                     <input type="hidden" name="Parar WEB" value="0">
146                 </form>
147             </td>
148         </tr>
149
150         <!-- Establecer SetPoint rpm -->
151         <tr>
152             <td>SetPoint (rpm):</td>
153             <td>
154                 <form method="post" action="" id="T3" name="T3">
155                     <center>
156                         <input type="number" max="1655" min="0" step="0.001" value="0" name="Variables"."PONER RPM SETPOINT" class="cajaTexto">
157                     </center>
158                     <button type="submit" class="botonSet">Set</button>
159                     </center>
160                 </form>
161             </td>
162         </tr>
163
164         <!-- Establecer Tipo de Control -->
165         <tr>
166             <td>Tipo de Control:</td>
167             <td>
168                 <form method="post" action="" id="T4" name="T4">
169                     <center>
170                         <input type="submit" name="Seleccionar tipo de control" value="Encoder" class="boton">
171                     </center>
172                     <input type="hidden" name="Seleccionar tipo de control" value="1">
173                 </form>
174             </td>
175             <td>
176                 <form method="post" action="" id="T4" name="T4">
177                     <center>
178                         <input type="submit" name="Seleccionar tipo de control" value="Tacogenerador" class="boton">
179                     </center>
180                     <input type="hidden" name="Seleccionar tipo de control" value="0">
181                 </form>
182             </td>
183         </tr>
184
185         <!-- Boton horario -->
186         <tr>
187             <td>
188                 <form method="post" action="" id="T4" name="T4">
189                     <center>
190                         <input type="submit" name="Seleccionar tipo de control" value="Tacogenerador" class="boton">
191                     </center>
192                     <input type="hidden" name="Seleccionar tipo de control" value="0">
193                 </form>
194             </td>
195             <td>
196                 <form method="post" action="" id="T4" name="T4">
197                     <center>
198                         <input type="submit" name="Seleccionar tipo de control" value="Tacogenerador" class="boton">
199                     </center>
200                     <input type="hidden" name="Seleccionar tipo de control" value="0">
201                 </form>
202             </td>
203         </tr>
204     </table>

```

```

189 <td>Giro del Motor Horario:</td>
190 <td>
191     <form method="post" action="" id="T5" name="T5">
192         <center>
193             <input type="submit" name="Giro horario WEB" value="ON" class="botonMotor">
194         </center>
195         <input type="hidden" name="Giro horario WEB" value="1">
196     </form>
197 </td>
198 <td>
199     <form method="post" action="" id="T5" name="T5">
200         <center>
201             <input type="submit" name="Giro horario WEB" value="OFF" class="botonMotor">
202         </center>
203         <input type="hidden" name="Giro horario WEB" value="0">
204     </form>
205 </td>
206 </tr>
207
208 <!-- Boton anti-horario -->
209 <tr>
210 <td>Giro del Motor Anti-Horario:</td>
211 <td>
212     <form method="post" action="" id="T6" name="T6">
213         <center>
214             <input type="submit" name="Giro anti-horario WEB" value="ON" class="botonMotor">
215         </center>
216         <input type="hidden" name="Giro anti-horario WEB" value="1">
217     </form>
218 </td>
219 <td>
220     <form method="post" action="" id="T6" name="T6">
221         <center>
222             <input type="submit" name="Giro anti-horario WEB" value="OFF" class="botonMotor">
223         </center>
224         <input type="hidden" name="Giro anti-horario WEB" value="0">
225     </form>
226 </td>
227 </tr>
228
229 <!-- Establecer parametros PID-->
230 <tr>
231 <td>Parametro K:</td>
232 <td>
233     <form method="post" action="" id="T7" name="T7">
234         <center>
235             <input type="number" max="100" min="0" step="0.001" value="0" name="Variables"."Parametro K" class="cajaTexto">
236
237             <center>
238                 <button type="submit" class="botonSet">Set</button>
239             </center>
240         </center>
241     </form>
242 </td>
243 </tr>
244
245
246 <tr>
247 <td>Parametro Td:</td>
248 <td>
249     <form method="post" action="" id="T8" name="T8">
250         <center>
251             <input type="number" max="100" min="0" step="0.001" value="0" name="Variables"."Parametro Td" class="cajaTexto">
252         </center>
253         <center>
254             <button type="submit" class="botonSet">Set</button>
255         </center>
256     </form>
257 </td>
258 </tr>
259
260
261 <tr>
262 <td>Parametro Ti:</td>
263 <td>
264     <form method="post" action="" id="T9" name="T9">
265         <center>
266             <input type="number" max="100" min="0" step="0.001" value="0" name="Variables"."Parametro Ti" class="cajaTexto">
267         </center>
268         <center>
269             <button type="submit" class="botonSet">Set</button>
270         </center>
271     </form>
272 </td>
273 </tr>
274
275 </table>
276
277 </table>
278
279 <br>
280 <div id="signal"></div>
281 <br>
282 <div id="output"></div>
283 <br>
284 </body>
285 </html>

```

8.4 CÓDIGO HTM DE LA PÁGINA WEB.

```

1  <!-- AWP_Enum_Ref Name="Control taco WEB" Enum="listado1" -->
2  <!-- AWP_Enum_Ref Name="Control encoder WEB" Enum="listado2" -->
3  <!-- AWP_Enum_Ref Name="Variables"."ACTIVAR PID" Enum="listado3" -->
4  <!-- AWP_Enum_Ref Name="LED giro anti-horario WEB" Enum="listado4" -->
5  <!-- AWP_Enum_Ref Name="LED giro horario WEB" Enum="listado5" -->
6
7  <table>
8  <thead>
9
10     <tr>
11         <th colspan="3">ESTADO DE LOS DATOS INTRODUCIDOS AL MOTOR DC</th>
12     </tr>
13 </thead>
14
15     <tr>
16         <td>Estado del Motor DC:</td>
17         <td align="right">:=Variables"."ACTIVAR PID":</td>
18         <td>
19             <center>
20                 
21             </center>
22         </td>
23     </tr>
24
25     <tr>
26         <td>Velocidad de referencia:</td>
27         <td align="right">:=Variables"."PONER RPM SETPOINT": <b>r.p.m.</b></td>
28         <td></td>
29     </tr>
30
31     <tr>
32         <td>Tipo de Control establecido:</td>
33         <td>Tacogenerador
34             <center>
35                 
36             </center>
37         </td>
38         <td>Encoder
39             <center>
40                 
41             </center>
42         </td>
43     </tr>
44
45     <tr>
46         <td>Giro del Motor Horario:</td>
47         <td align="right">:=LED giro horario WEB:</td>
48         <td>
49             <center>
50                 
51             </center>
52         </td>
53     </tr>
54
55     <tr>
56         <td>Giro del Motor Anti-Horario:</td>
57         <td align="right">:=LED giro anti-horario WEB:</td>
58         <td>
59             <center>
60                 
61             </center>
62         </td>
63     </tr>
64 </table>

```

```

1  <table>
2      <thead>
3          <tr>
4              <th colspan="3">ESTADO DE LAS SEÑALES DEL MOTOR DC</th>
5          </tr>
6      </thead>
7
8      <tr>
9          <td>Velocidad del Motor por la señal del Encoder:</td>
10         <td align="right">="Velocidad por Encoder rpm": <b>r.p.m.</td>
11         <td align="right">="Velocidad por Encoder %": <b>%</td>
12     </tr>
13
14     <tr>
15         <td>Velocidad del Motor por la señal del Tacogenerador:</td>
16         <td align="right">="Velocidad por Taco. rpm": <b>r.p.m.</td>
17         <td align="right">="Velocidad por Taco. %": <b>%</td>
18     </tr>
19
20     <tr>
21         <td>Error de la señal del Encoder:</td>
22         <td align="right">="Señal de error Encoder": <b>r.p.m.</td>
23         <td></td>
24     </tr>
25
26     <tr>
27         <td>Error de la señal del Tacogenerador:</td>
28         <td align="right">="Señal de error Taco.": <b>r.p.m.</td>
29         <td></td>
30     </tr>
31
32     <tr>
33         <td>Tensión de alimentacion del Motor:</td>
34         <td align="right">="Tension de alimentacion al motor V": <b>V</td>
35         <td></td>
36     </tr>
37
38     <tr>
39         <td>Tensión de salida del Motor:</td>
40         <td align="right">="Tension de salida del motor V": <b>V</td>
41         <td></td>
42     </tr>
43 </table>

```

8.5 CÓDIGO CSS DE LA PÁGINA WEB.

```

1  /* Estilos para las imagenes */
2  #imagenes{
3      float:top;
4      width:0%;
5      position:relative;
6      left:3px;
7      top:15px;
8  }
9
10 /* Delay para tiempo de transicion */
11 #delay{
12     transition-duration:5000ms;
13 }
14
15 /* Estilo del titulo del proyecto */
16 #header{
17     color:white;
18     position:relative;
19     font-family:Georgia, fantasy;
20     font-size:100%;
21     width:100%;
22     height:15%;
23     display:flex;
24     flex-direction:column;
25     align-items:center;
26     justify-content:center;
27     text-align:center;
28     text-shadow: 3px 3px #000000;
29     border: 4px solid #000000;
30     background: #052D79;
31     border-radius: 20px;
32 }
33
34 /* Estilo para el cuerpo del proyecto */
35 body{
36     font-family:Times New Roman, serif;
37     width:95%;
38     padding:10px 20px;
39     background-position:left center;
40     background-repeat:no-repeat;
41     background-size:cover;
42     background-attachment:fixed;
43 }
44
45 /* Estilo para las tablas */
46 table{
47     width:100%;
48     background-color:white;
49     text-align:left;
50     font-size:15px;
51     border-collapse:collapse;
52     font-weight:bold;
53     border-bottom:solid 8px #052D79;
54     border-top:solid 8px #052D79;
55     border-left:solid 8px #052D79;
56     border-right:solid 8px #052D79;
57     margin: 5px 5px;
58 }
59 /* La primera columna de las tablas */
60 table td:first-child{
61     width:30%;
62     height:100px;
63     font-size:17px;
64 }
65 /* Modificacion del numero de columns y estructura */
66 th{
67     border:solid 1px #868686;
68     padding:10px;
69 }

```



```

70  /* Cada una de las columnas de la fila */
71  td{
72      width:35%;
73      height:100px;
74      border:solid 1px #868686;
75      padding:20px;
76  }
77  /* Titulo de la tabla */
78  thead{
79      background-color:#052D79;
80      border-top:solid 5px 215E50;
81      color:white;
82      text-align:center;
83      font-size:26px;
84      text-shadow: 3px 3px #000000;
85  }
86  /* Las filas pares de distinto color */
87  tr:nth-child(even){
88      background-color: #E5E5E5;
89  }
90
91  /* Boton para Arranque, Paro y Control*/
92  .boton{
93      width:45%;
94      height:35%;
95      background-color:#6CE2CB;
96      border-radius:0.5em;
97      font-size:20px;
98      font-weight:700;
99      padding:20px 40px;
100     color:white;
101     border-color:black;
102     text-align:center;
103     -webkit-box-shadow: inset 0px 0px 100px 0.1px rgba(0,95,95,1);
104     -moz-box-shadow: inset 500px 0px 50px 100px rgba(0,95,95,1);
105     box-shadow: inset 4px 4px 50px 5px rgba(0,95,80,1);
106 }
107 /* Cambio de color cuando se pasa por encima del boton */
108 .boton:hover{
109     background-color:#7FFFFD;
110     color:black;
111     border-color:black;
112     cursor: pointer;
113     -webkit-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
114     -moz-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
115     box-shadow: inset 0px 0px 21px 1px rgba(0,143,100,1);
116 }
117
118 /* Cambio de color y tamaño cuando pulsa el boton */
119 .boton:active{
120     width:46%;
121     height:26%;
122     padding: 22px 42px;
123     font-size:22px;
124     border-radius:0.5em;
125     background-color:#D0FFFE;
126     -webkit-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
127     -moz-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
128     box-shadow: inset 0px 0px 21px 1px rgba(0,143,100,1);
129 }
130
131 /* Botones usados para sentido de giro del motor */
132 .botonMotor{
133     width:25%;
134     height:45%;
135     background-color:#6CE2CB;
136     border-radius:0.5em;
137     transition:background-color 0.15s;
138     font-size:21px;
139     font-weight:700;
140     padding:20px 40px;
141     color:white;
142     border-color:black;
143     text-align:center;
144     -webkit-box-shadow: inset 0px 0px 100px 0.1px rgba(0,95,95,1);
145     -moz-box-shadow: inset 500px 0px 50px 100px rgba(0,95,95,1);
146     box-shadow: inset 4px 4px 50px 5px rgba(0,95,80,1);

```

```

147  /* Cambio de color cuando se pasa por encima del boton */
148  .botonMotor:hover{
149      background-color:#7FFFFD;
150      color:black;
151      border-color:black;
152      cursor: pointer;
153      -webkit-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
154      -moz-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
155      box-shadow: inset 0px 0px 21px 1px rgba(0,143,100,1);
156  }
157  /* Cambio de color y tamaño cuando pulsa el boton */
158  .botonMotor:active{
159      width:26%;
160      height:46%;
161      padding: 22px 42px;
162      font-size:23px;
163      border-radius:0.5em;
164      background-color:#D0FFFE;
165      -webkit-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
166      -moz-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
167      box-shadow: inset 0px 0px 21px 1px rgba(0,143,100,1);
168  }
169
170  /* Boton usado para el SetPoint y Parametros PID */
171  .botonSet{
172      width:35%;
173      height:75%;
174      background-color:#6CE2CB;
175      border-radius:0.5em;
176      transition:background-color 0.15s;
177      font-size:19px;
178      font-weight:700;
179      padding:20px 40px;
180      color:white;
181      border-color:black;
182      text-align:center;
183      -webkit-box-shadow: inset 0px 0px 100px 0.1px rgba(0,95,95,1);
184      -moz-box-shadow: inset 500px 0px 50px 100px rgba(0,95,95,1);
185      box-shadow: inset 4px 4px 50px 5px rgba(0,95,80,1);
186  }
187  /* Cambio de color cuando se pasa por encima del boton */
188  .botonSet:hover{
189      background-color:#7FFFFD;
190      color:black;
191      border-color:black;
192      cursor: pointer;
193      -webkit-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
194      -moz-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
195      box-shadow: inset 0px 0px 21px 1px rgba(0,143,100,1);
196  }
197
198  /* Cambio de color y tamaño cuando pulsa el boton */
199  .botonSet:active{
200      width:37%;
201      height:77%;
202      padding: 22px 42px;
203      font-size:22px;
204      border-radius:0.5em;
205      background-color:#D0FFFE;
206      -webkit-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
207      -moz-box-shadow: inset 10px 10px 21px 1px rgba(0,143,143,1);
208      box-shadow: inset 0px 0px 21px 1px rgba(0,143,100,1);
209  }
210  /* Cajas para escribir el SetPoint y los Parametros PID */
211  .cajaTexto{
212      width:55%;
213      height:65px;
214      font-size:18px;
215      font-weight:bold;
216      border-color:black;
217      text-align:center;
218      border-bottom-width:2px;
219      border-right-width:2px;
220      border-top-width:2px;
221      border-left-width:2px;
222      background-color:#D8FDFF;
223  }
224  /* Cambio de color cuando se pasa por encima del boton */
225  .cajaTexto:hover{
226      background-color:#A6FFFA;
227  }

```