

FICHA DE ENTREGA DE ACTIVIDADES EVALUABLES.

Nombre: Joel Sanz Martí	Fecha: 30/11/23
Unidad didáctica: 3	Nota:

ACTIVIDAD EVALUABLE nº 14-16**1. Enunciado y características de la actividad:**

Se desea realizar un banco de pruebas de una máquina que utiliza un husillo, en este banco de pruebas se desea determinar la velocidad real de avance del husillo según unas frecuencias aplicadas al motor, para ello se debe implementar un programa que mida la velocidad en RPM del husillo cuando se aplican las frecuencias de 30Hz, 50Hz y 80Hz.

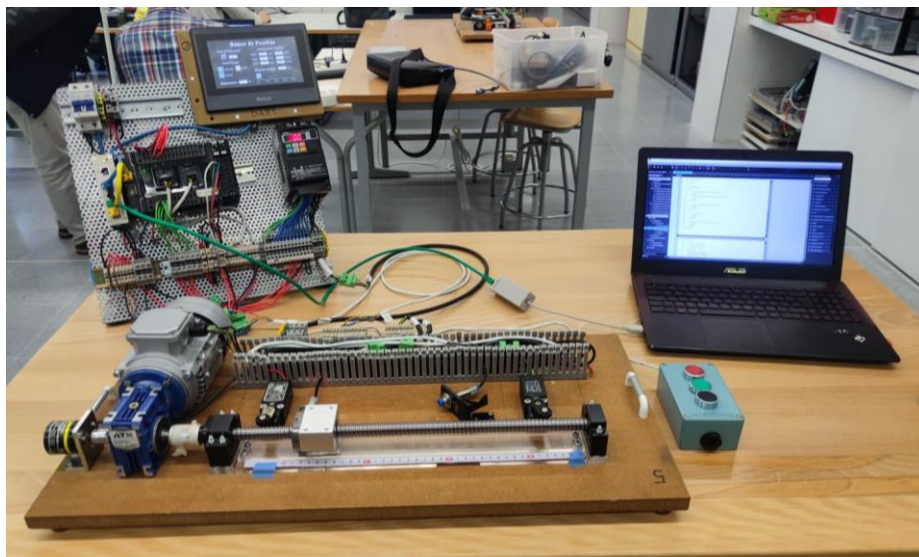
Se debe tener en cuenta que se debe posicionar en el origen (detector inductivo) para iniciar el test de RPM y mm/s. Al pulsar marcha, si está posicionado empezará a avanzar, de lo contrario, se tendrá que pulsar reset para que retroceda hasta estar posicionado. Cuando llegue a 100mm de desplazamiento, parará y esperará a que se pulse reset para retroceder y así volver a iniciar una prueba. Si se pulsa paro en cualquier momento, el husillo parará y obligará a devolverlo a su posición inicial después de pulsar reset.

La selección de velocidad de prueba se realizará desde el HMI. Desde el HMI también se podrá visualizar el movimiento del husillo.

Este proyecto se ha realizado con Raúl Fornes.

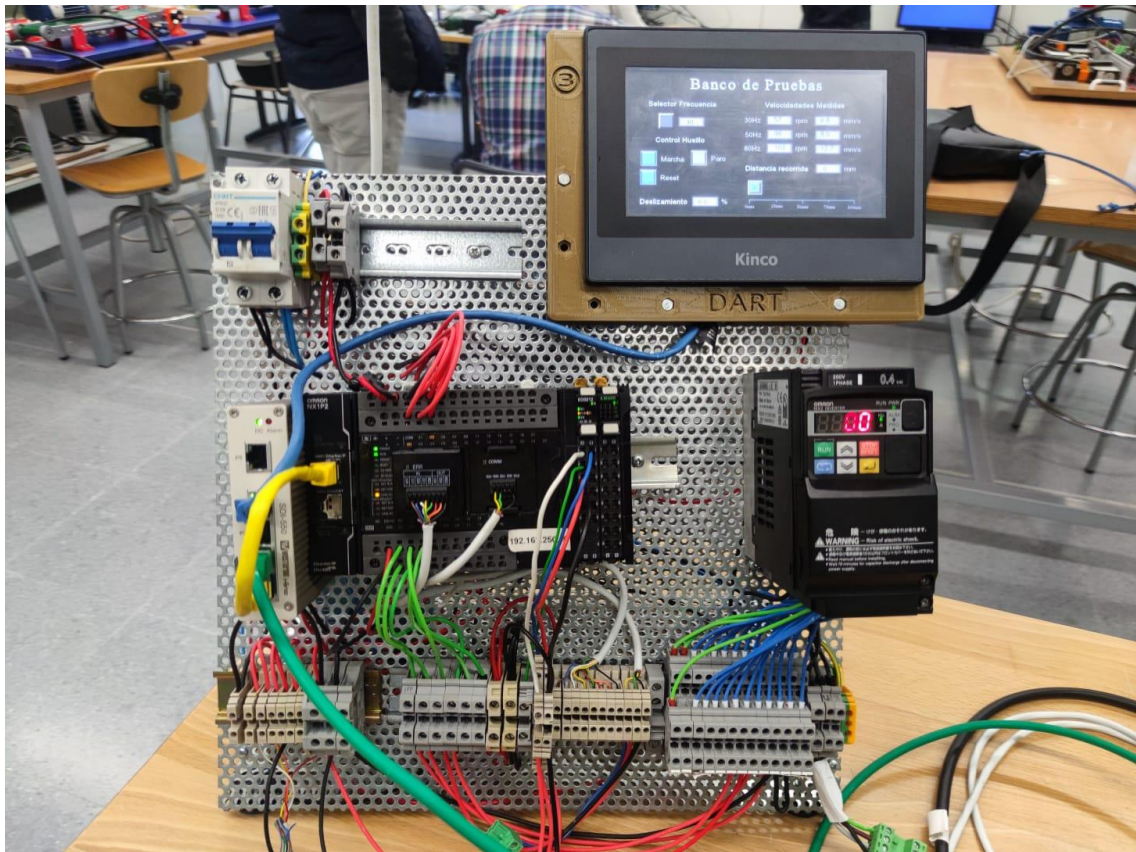
2. Imagen del montaje realizado:

Imagen del montaje completo:



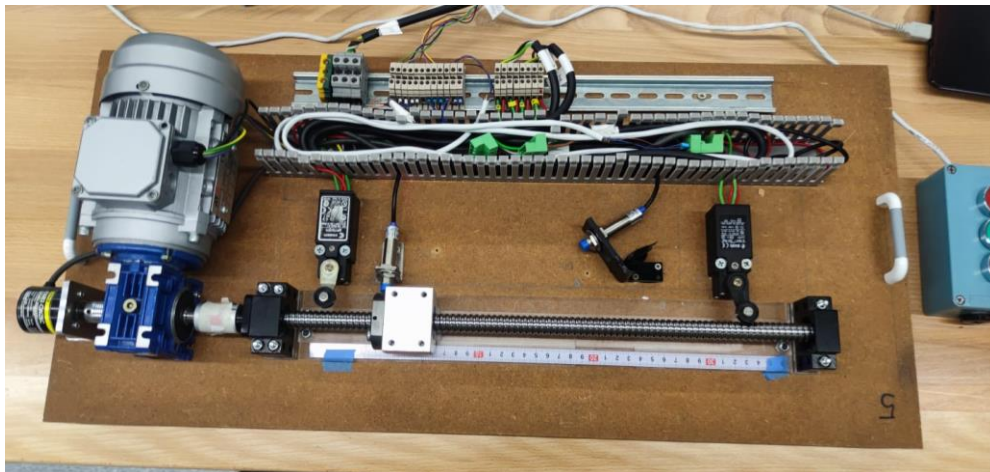
El montaje está formado por el PLC que controla el sistema, el HMI para monitorizar y controlar el sistema, una botonera física para controlar el sistema y un variador de frecuencia controlado por el PLC y que actúa sobre el motor que mueve el husillo. Además, desde el PC se programan el PLC y el HMI.

Maqueta de control:



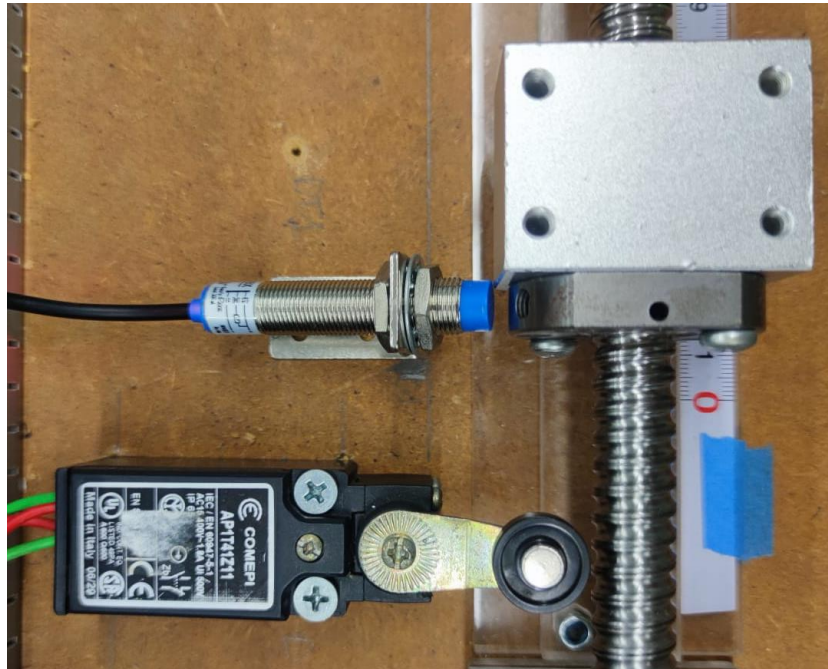
En esta maqueta están el PLC, el HMI y el variador de frecuencia.

Maqueta del husillo:



Maqueta donde está el husillo, movido por un motor asíncrono trifásico. Además, hay dos finales de carrera de protección, dos detectores inductivos y un encoder asociado al reductor 1:15 del motor.

Detalle del final de carrera y el detector inductivo:



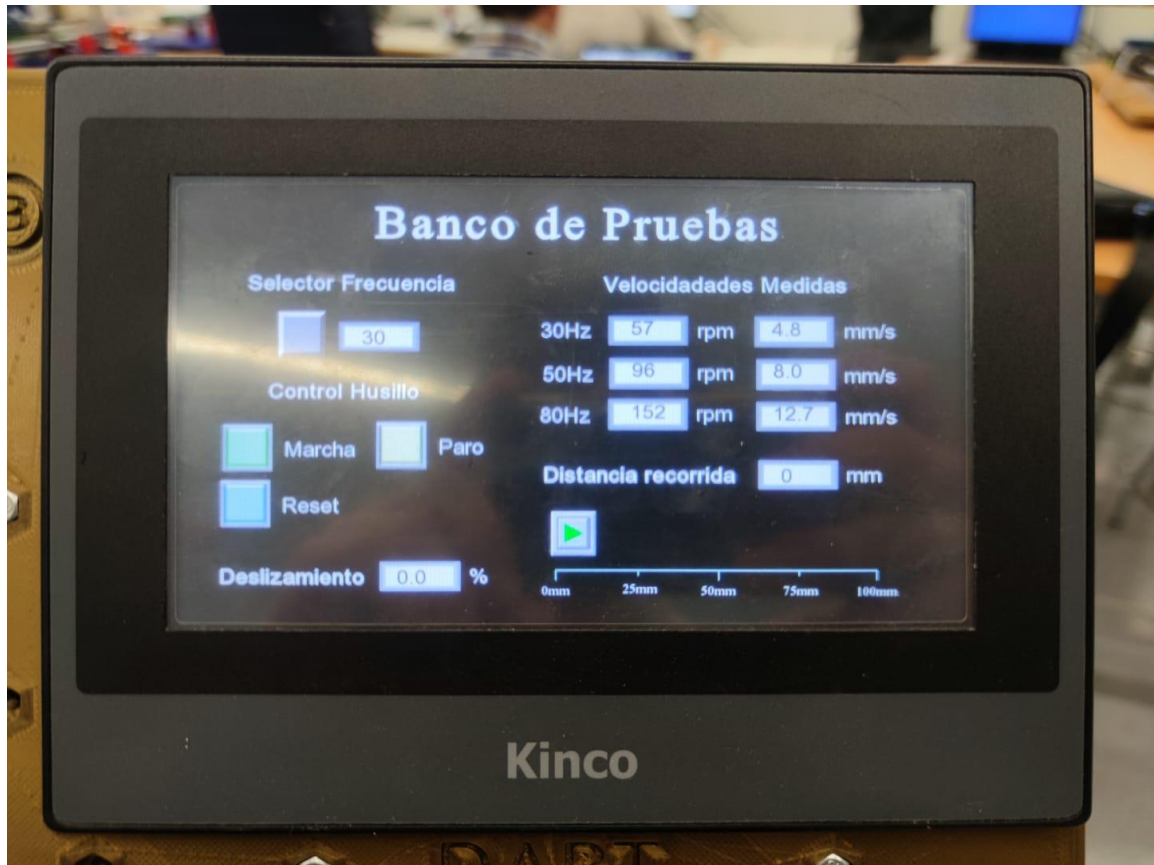
Estos son el final de carrera y el detector inductivo más próximos al motor. El final de carrera lo hemos usado como protección para que el carro del husillo no se salga de su carrera, mientras que el detector inductivo define la posición inicial del carro en el programa.

Botonera:



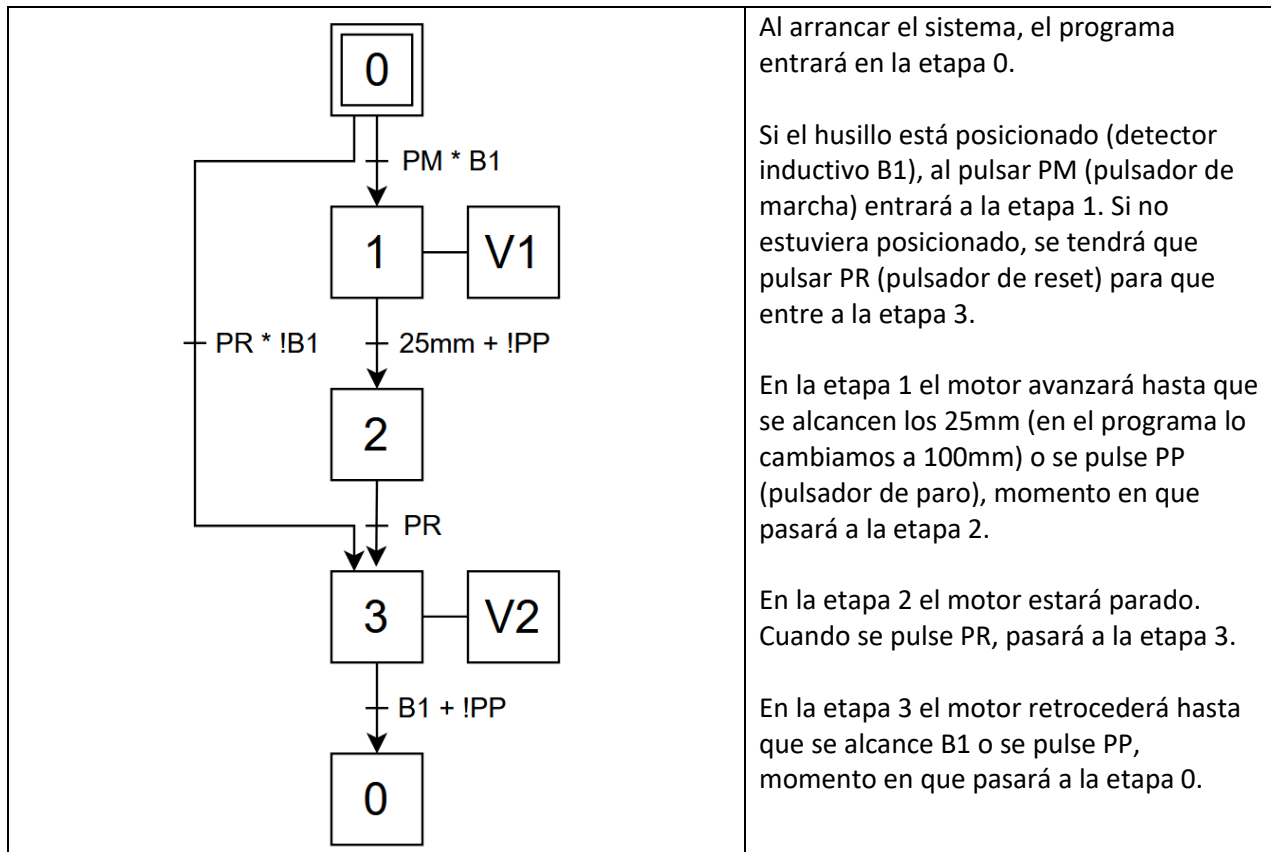
Esta botonera tiene los botones de reset (negro), marcha (verde) y paro (rojo) que tienen el funcionamiento ya explicado en el enunciado del problema y que se volverán a explicar más adelante.

Pantalla HMI:



Pantalla que servirá para variar la frecuencia a aplicar al motor, ver los valores obtenidos de las medidas, el desplazamiento del husillo y controlar su movimiento. Además, se ha añadido la visualización del deslizamiento del motor.

3. Grafcet de control implementado y explicación:



4. Elementos físicos no programables utilizados, cableado y función en el montaje.

- Pulsadores de paro, marcha:

Los dos pulsadores se encuentran en una única botonera y sirven para dar la señal de comienzo del proceso, pararlo y reanudarlo.

- Motor asíncrono:

Es el encargado de mover el carro del husillo. Lleva un reductor 1:15 (por cada 15 vueltas del motor, el eje del reductor da una vuelta). Es controlado por el variador de frecuencia.

- Detector inductivo:

Detector inductivo NPN. Define la posición de inicio del carro del husillo.

- Final de carrera:

Sirve como protección para evitar que se el carro del husillo se salga de su carrera.

- Encoder:

Asociado al reductor del motor que mueve el husillo. Se encarga de enviar pulsos al PLC para que pueda calcular el desplazamiento del husillo, su velocidad, etc.

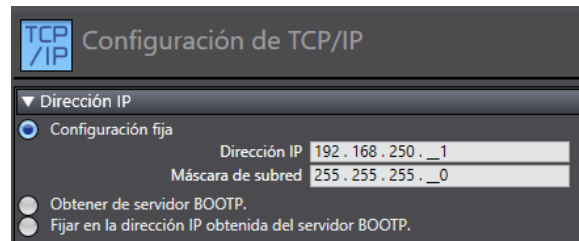
A continuación, se muestran los esquemas eléctricos. En dichos esquemas se puede observar que la alimentación del variador de frecuencia es trifásica, pero se ha cableado en monofásica. Esto es porque el variador de frecuencia usado en clase es monofásico. Solo hay macro de este variador con alimentación trifásica, pero he decidido ser fiel al cableado realizado en clase.

Además, la referencia del HMI tampoco coincide con el de clase porque no se ha encontrado la referencia correcta, pero las conexiones son las mismas.

5. Configuraciones de elementos programables utilizados y su función en el proyecto.

- OMRON NX1P2:

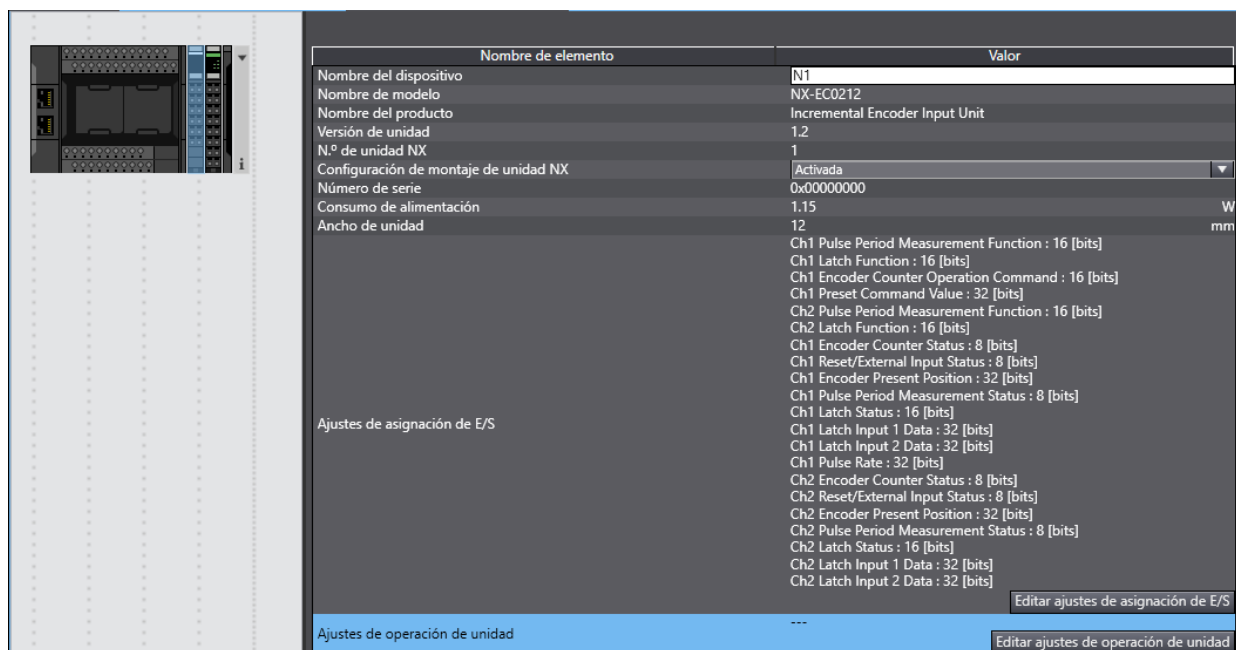
Es el PLC que controla el sistema. Recibe señales de sus entradas y, en función del programa que tiene cargado, actúa sobre las salidas correspondientes. Su puerto de salidas tiene una configuración PNP fija, pero su puerto de entradas puede ser tanto PNP como NPN. En nuestro caso, dicho puerto estará configurado en NPN. Este PLC tendrá la dirección IP 192.168.250.1.



- Módulo de encoder incremental NX-EC0212:

Es un módulo lateral del PLC que añade entradas para recibir los pulsos de dos encoders incrementales. Dispone de una entrada A, B y Z por cada canal y salidas de alimentación a 24Vdc.

Lo agregamos al bastidor de CPU en el programa del PLC:



Una vez agregado, entraremos a “Editar ajustes de asignación de E/S” para asignar las entradas y salidas que usaremos en el canal 1:

A las salidas agregaremos “Ch1 Encoder counter operation” y “Ch1 Preset Command Value”.

Entradas de E/S incluidas en Output Data Set 1

Índice	Tamaño	Tipo de datos	Nombre de entrada de E/S	Comentario
0x7008:01	16[bit]	WORD	Ch1 Pulse Period Measurement Function	Aggregated byte data of the Pulse Period Measurement Function for encoder input 1
0x7004:01	16[bit]	WORD	Ch1 Latch Function	Aggregated data of latch function for encoder input 1
0x7000:01	16[bit]	WORD	Ch1 Encoder Counter Operation Command	Aggregated byte data of counter operation command for encoder input 1
0x7002:01	32[bit]	DINT	Ch1 Preset Command Value	Counter preset command value for encoder input 1

A las entradas añadiremos “Ch1 Pulse Rate”.

Entradas de E/S incluidas en Input Data Set 1

Índice	Tamaño	Tipo de datos	Nombre de entrada de E/S	Comentario
0x6000:01	8[bit]	BYTE	Ch1 Encoder Counter Status	Aggregated status of Encoder Counter 1
0x6001:01	8[bit]	BYTE	Ch1 Reset/External Input Status	Aggregated status of reset/external inputs for encoder input 1
0x6002:01	32[bit]	DINT	Ch1 Encoder Present Position	Encoder 1 current position
0x6008:01	8[bit]	BYTE	Ch1 Pulse Period Measurement Status	Aggregated status of pulse period measurements of encoder input 1
---	8[bit]	---	Padding	---
0x6004:01	16[bit]	WORD	Ch1 Latch Status	Aggregated latch status of encoder input 1
0x6005:01	32[bit]	DINT	Ch1 Latch Input 1 Data	The value latched by latch input 1 of external input/phase Z input of encoder input 1
0x6006:01	32[bit]	DINT	Ch1 Latch Input 2 Data	The value latched by latch input 2 of external input/phase Z input of encoder input 1
0x6003:01	32[bit]	UDINT	Ch1 Pulse Rate	Counter 1 pulse rate

A continuación iremos al módulo dentro de “Bastidores de CPU” y lo configuraremos para el uso que le vamos a dar:

Todos los parámetros	
Nombre de elemento	Valor
Counter Type/Ch1 Counter Type	Ring Counter
Counter Type/Ch2 Counter Type	Ring Counter
Maximum Counter Value/Ch1 Maximum Counter Value	2147483647 pulse
Maximum Counter Value/Ch2 Maximum Counter Value	2147483647 pulse
Minimum Counter Value/Ch1 Minimum Counter Value	-2147483648 pulse
Minimum Counter Value/Ch2 Minimum Counter Value	-2147483648 pulse
Pulse Input Method/Ch1 Pulse Input Method	Phase Differential Pulse x4
Pulse Input Method/Ch2 Pulse Input Method	Phase Differential Pulse x4
Time Window/Ch1 Time Window	500 ms
Time Window/Ch2 Time Window	0 ms
Average Processing Times/Ch1 Average Processing Times	1 times
Average Processing Times/Ch2 Average Processing Times	0 times
Edge Detection Method/Ch1 Edge Detection Method	Disable the Function
Edge Detection Method/Ch2 Edge Detection Method	Disable the Function
Encoder Count Direction/Ch1 Encoder Count Direction	Positive Direction of Phase B
Encoder Count Direction/Ch2 Encoder Count Direction	Positive Direction of Phase B

Pondremos el tipo de contador en “Ring Counter” para que si los pulsos llegan al valor máximo vuelva al valor inicial.

El método de entrada será en cuadratura (“Phase Differential Pulse x4”).

“Time Window” será de 500ms y “Average Processing Times” será 1. De esta forma, el modulo guardará el incremento de pulsos cada 500ms y así calcular el ratio de pulsos.

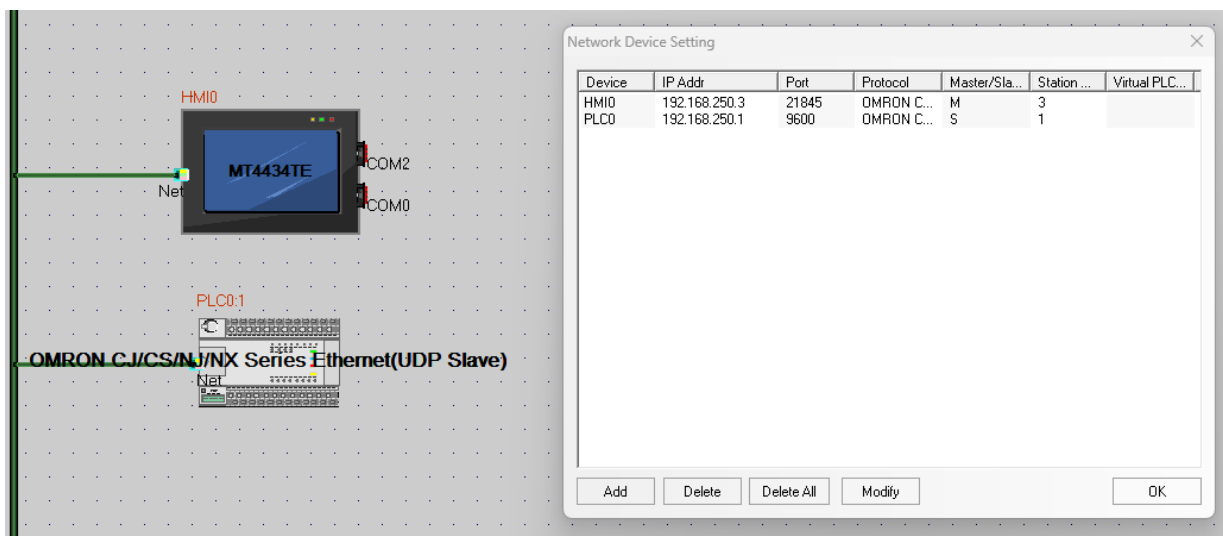
- Variador de frecuencia OMRON MX2:

Es el variador de frecuencia que controla el motor. Este, a su vez, es controlado por el PLC mediante sus entradas digitales. Para conseguir el funcionamiento deseado, hay que parametrizarlo. Esta es la parametrización necesaria para este proyecto:

A001 – Configuración de origen de frecuencia	-> 2: Operador Digital
A002 – Configuración de la señal de Run	-> 1: Terminales de Control
A004 – Configuración de máxima frecuencia	-> 80
A019 – Selección operación de multivelocidad	-> 0: Binaria
A020 – Referencia de multivelocidad 0	-> 30
A021 – Referencia de multivelocidad 1	-> 50
A022 – Referencia de multivelocidad 2	-> 80
C001 – Función de la entrada [1]	-> 00: FW marcha directa
C002 – Función de la entrada [2]	-> 01: BW marcha inversa
C003 – Función de la entrada [3]	-> 02: CF1
C004 – Función de la entrada [4]	-> 03: CF2
F001 – Configuración de la referencia de frecuencia	-> 80

- HMI Kinco MT4434TE:

Es el HMI que hemos utilizado en este proyecto. Su uso es el de monitorizar el proceso, además de poder comandarlo igual que desde la botonera. También se podrán variar los valores de entrada a cada zona. Para que pueda establecer conexión con el PLC, necesitamos configurarlo en la misma red:



El HMI tendrá la dirección IP 192.168.250.3.

6. Asignación de memoria y lista de IO.

Lista de entradas y salidas físicas:

Input Bit 00	Input Bit 00	R	BOOL	PP	Pulsador Paro	Variables globales
Input Bit 01	Input Bit 01	R	BOOL	PM	Pulsador Marcha	Variables globales
Input Bit 02	Input Bit 02	R	BOOL	PR	Pulsador Reset	Variables globales
Input Bit 03	Input Bit 03	R	BOOL	detector_inicial	Detector Inductivo B1	Variables globales
Output Bit 00	Output Bit 00	RW	BOOL	marcha_atras	Señal Retroceso Motor	Variables globales
Output Bit 01	Output Bit 01	RW	BOOL	marcha_adelante	Señal Avance Motor	Variables globales
Output Bit 02	Output Bit 02	RW	BOOL	multivelocidad_1	Señal Multivelocidad 1	Variables globales
Output Bit 03	Output Bit 03	RW	BOOL	multivelocidad_2	Señal Multivelocidad 2	Variables globales

Activación de las áreas de memoria usadas:

Configuración de memoria para unidades de la serie CJ				
Activar	Tipo de área	Tamaño (número de canales)	Retener	
<input type="checkbox"/>	CIO	6144	Sin retención	
<input checked="" type="checkbox"/>	WR	512	Sin retención	
<input type="checkbox"/>	HR	512	Con retención	
<input checked="" type="checkbox"/>	DM	4096	Con retención	

Variables internas:

Nombre	Tipo de datos	Valor inicial	AT	Retentiva	Constante	Comentario
time_window	INT	500		<input type="checkbox"/>	<input checked="" type="checkbox"/>	
encoder_resolution	INT	8000		<input type="checkbox"/>	<input checked="" type="checkbox"/>	Resolución del Encoder (x4)
state	INT			<input type="checkbox"/>	<input type="checkbox"/>	Contador de Etapa
velocidad_sincrona	REAL			<input type="checkbox"/>	<input type="checkbox"/>	Velocidad Sincrona del Motor

Variables externas:

Nombre	Tipo de datos	Constante
Ch1Enable	BOOL	<input type="checkbox"/>
Ch1Pulses	DINT	<input type="checkbox"/>
rpm	DINT	<input type="checkbox"/>
Ch1PulseRate	UDINT	<input type="checkbox"/>
PP	BOOL	<input type="checkbox"/>
PM	BOOL	<input type="checkbox"/>
detector_inicial	BOOL	<input type="checkbox"/>
marcha_adelante	BOOL	<input type="checkbox"/>
marcha_atras	BOOL	<input type="checkbox"/>
Ch1Reset	BOOL	<input type="checkbox"/>
PR	BOOL	<input type="checkbox"/>
mm	DINT	<input type="checkbox"/>
multivelocidad_1	BOOL	<input type="checkbox"/>
multivelocidad_2	BOOL	<input type="checkbox"/>
frecuencia_aplicada	INT	<input type="checkbox"/>
rpm_30hz	DINT	<input type="checkbox"/>
mm_s_30hz	REAL	<input type="checkbox"/>
rpm_50hz	DINT	<input type="checkbox"/>
mm_s_50hz	REAL	<input type="checkbox"/>
rpm_80hz	DINT	<input type="checkbox"/>
mm_s_80hz	REAL	<input type="checkbox"/>

selector_velocidad	INT	<input type="checkbox"/>
mm_s	REAL	<input type="checkbox"/>
PM_VAR	BOOL	<input type="checkbox"/>
PM_HMI	BOOL	<input type="checkbox"/>
PP_HMI	BOOL	<input type="checkbox"/>
PP_VAR	BOOL	<input type="checkbox"/>
PR_HMI	BOOL	<input type="checkbox"/>
PR_VAR	BOOL	<input type="checkbox"/>
deslizamiento	REAL	<input type="checkbox"/>

Variables globales:

Nombre	Tipo de datos	Valor inicial	AT	Retentiva	Constante	Publicación en red	Comentario
PP	BOOL		BuiltInIO//cpu/#0/Input_Bit_00	<input type="checkbox"/>	<input type="checkbox"/>	No publica	Pulsador Paro
PM	BOOL		BuiltInIO//cpu/#0/Input_Bit_01	<input type="checkbox"/>	<input type="checkbox"/>	No publica	Pulsador Marcha
PR	BOOL		BuiltInIO//cpu/#0/Input_Bit_02	<input type="checkbox"/>	<input type="checkbox"/>	No publica	Pulsador Reset
marcha_atras	BOOL		BuiltInIO//cpu/#0/Output_Bit_00	<input type="checkbox"/>	<input type="checkbox"/>	No publica	Señal Retroceso Motor
state	INT			<input type="checkbox"/>	<input type="checkbox"/>	No publica	Contador de Etapas
rpm	DINT		%D0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Velocidad del Husillo en RPM
mm_s	REAL		%D4	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Velocidad del Husillo en mm/s
Ch1Enable	BOOL		IOBus//unit#1/Ch1 Encoder Counter Operation Command/Ch1 Counter Enable	<input type="checkbox"/>	<input type="checkbox"/>	No publica	Activación del Contaje Canal 1
Ch1Pulses	DINT		IOBus//unit#1/Ch1 Encoder Present Position	<input type="checkbox"/>	<input type="checkbox"/>	No publica	Contaje Pulsos Actuales Canal 1
Ch1PulseRate	UDINT		IOBus//unit#1/Ch1 Pulse Rate	<input type="checkbox"/>	<input type="checkbox"/>	No publica	Ratio de Pulsos Canal 1
detector_inicial	BOOL		BuiltInIO//cpu/#0/Input_Bit_03	<input type="checkbox"/>	<input type="checkbox"/>	No publica	Detector Inductivo B1
marcha_adelante	BOOL		BuiltInIO//cpu/#0/Output_Bit_01	<input type="checkbox"/>	<input type="checkbox"/>	No publica	Señal Avance Motor
mm	DINT		%D8	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Desplazamiento del Husillo en mm
Ch1Reset	BOOL		IOBus//unit#1/Ch1 Encoder Counter Operation Command/Ch1 Internal Reset Execution	<input type="checkbox"/>	<input type="checkbox"/>	No publica	Reinicio del Contaje Canal 1
multivelocidad_1	BOOL		BuiltInIO//cpu/#0/Output_Bit_02	<input type="checkbox"/>	<input type="checkbox"/>	No publica	Señal Multivelocidad 1
multivelocidad_2	BOOL		BuiltInIO//cpu/#0/Output_Bit_03	<input type="checkbox"/>	<input type="checkbox"/>	No publica	Señal Multivelocidad 2
selector_velocidad	INT		%D12	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Selector de Multivelocidad HMI
frecuencia_aplicada	INT		%D16	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Frecuencia Aplicada al Motor
rpm_30hz	DINT		%D20	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Velocidad a 30Hz en RPM
rpm_50hz	DINT		%D24	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Velocidad a 50Hz en RPM
rpm_80hz	DINT		%D28	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Velocidad a 80Hz en RPM
mm_s_30hz	REAL		%D32	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Velocidad a 30Hz en mm/s
mm_s_50hz	REAL		%D36	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Velocidad a 50Hz en mm/s
mm_s_80hz	REAL		%D40	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Velocidad a 80Hz en mm/s
MC_state	INT	0	%D7	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Símbolo del Moving Component HMI
PM_HMI	BOOL		%W0.00	<input type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Pulsador Marcha HMI
PP_HMI	BOOL		%W0.01	<input type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Pulsador Paro HMI
PR_HMI	BOOL		%W0.02	<input type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Pulsador Reset HMI
PM_VAR	BOOL			<input type="checkbox"/>	<input type="checkbox"/>	No publica	Variable Pulsador Marcha
PP_VAR	BOOL			<input type="checkbox"/>	<input type="checkbox"/>	No publica	Variable Pulsador Paro
PR_VAR	BOOL			<input type="checkbox"/>	<input type="checkbox"/>	No publica	Variable Pulsador Reset
deslizamiento	REAL		%D44	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Solo publicar	Deslizamiento del Motor en Hz

Asignación de variables globales al módulo EXNX-EC0212:

▼ NX-EC0212						
► Ch1 Encoder Counter Status	Aggregated status of Encoder Counter 1	R	BYTE			
► Ch1 Reset/External Input Status	Aggregated status of reset/external inputs for encoder input 1	R	BYTE			
Ch1 Encoder Present Position	Encoder 1 current position	R	DINT	Ch1Pulses	Contaje Pulsos Actuales Canal 1	Variables globales
Ch1 Pulse Rate	Counter 1 pulse rate	R	UDINT	Ch1PulseRate	Ratio de Pulsos Canal 1	Variables globales
► Ch1 Latch Status	Aggregated latch status of encoder input 1	R	WORD			
Ch1 Latch Input 1 Data	The value latched by latch input 1 of external input/phase Z input of encoder input 1	R	DINT			
Ch1 Latch Input 2 Data	The value latched by latch input 2 of external input/phase Z input of encoder input 1	R	DINT			
► Ch1 Pulse Period Measurement Status	Aggregated status of pulse period measurements of encoder input 1	R	BYTE			
▼ Ch1 Encoder Counter Operation Command	Aggregated byte data of counter operation command for encoder input 1	W	WORD			
Ch1 Counter Enable	Enable or disable the counter for encoder input 1	W	BOOL	Ch1Enable	Activación del Contaje Canal 1	Variables globales
Ch1 Internal Reset Execution	Execute the internal reset for encoder input 1	W	BOOL	Ch1Reset	Reinicio del Contaje Canal 1	Variables globales
Ch1 Internal Latch Execution	Execute the internal latch for encoder input 1	W	BOOL			
Ch1 Preset Execution	Execute presetting for encoder input 1	W	BOOL			
Ch1 External Reset Enable	Enable the external reset for encoder input 1	W	BOOL			
Ch1 Phase Z Reset Enable	Enable the phase Z reset for encoder input 1	W	BOOL			
Ch1 External Reset Completed Flag Clear	Clear the external reset completed flag for encoder input 1	W	BOOL			
Ch1 Phase Z Reset Completed Flag Clear	Clear the phase Z reset completed flag for encoder input 1	W	BOOL			
Ch1 Preset Command Value	Counter preset command value for encoder input 1	W	DINT			

7. Programa realizado:

- Programa PLC:

El programa del PLC se divide en 4 segmentos:

calculos_encoder:

Realización de los cálculos de velocidad y desplazamiento a partir de los datos obtenidos del encoder.

```

0
1 // Cálculo de la velocidad del husillo en RPM y mm/s
2 rpm := (Ch1PulseRate * 60000) / (encoder_resolution * time_window);
3 mm_s := (TO_REAL(rpm) / 60 * 500.0) / 100.0;
4
5 // Cálculo del desplazamiento lineal del husillo en mm
6 mm := Ch1Pulses * 5 / encoder_resolution;

```

principal:

En esta sección está el funcionamiento principal del programa (el descrito por el grafcet):

```

0 Grafcet Principal
1 IF P_First_Run THEN
2   state := 0;
3 END_IF;
4
5 CASE state OF
6   0: // Si el husillo está posicionado y se pulsa marcha, entrar a la etapa 1
7     // Si no está posicionado y se pulsa reset, entrar a la etapa 2
8     IF PM_VAR AND detector_inicial THEN
9       state := 1;
10    ELSIF PR_VAR AND NOT detector_inicial THEN
11      state := 3;
12    END_IF;
13  1: // Si el husillo se desplaza 100mm o si se pulsa paro, entrar a la etapa 2
14    IF mm >= 100 OR NOT PP_VAR THEN
15      state := 2;
16    END_IF;
17  2: // Si se pulsa reset, entrar a la etapa 3
18    IF PR_VAR THEN
19      state := 3;
20    END_IF;
21  3: // Si el husillo llega al detector o se pulsa paro, entrar a la etapa 0
22    IF detector_inicial OR NOT PP_VAR THEN
23      state := 0;
24    END_IF;
25 END_CASE;

1 Tratamiento Posterior
1 CASE state OF
2   0: // En la etapa 0 parar el motor y resetear el conteo si está posicionado
3     Ch1Enable := FALSE;
4     IF detector_inicial THEN
5       Ch1Reset := TRUE;
6     END_IF;
7     marcha_atras := FALSE;
8     marcha_adelante := FALSE;
9   1: // En la etapa 1, el motor avanza y se activa el conteo
10    Ch1Reset := FALSE;
11    Ch1Enable := TRUE;
12    marcha_adelante := TRUE;
13    marcha_atras := FALSE;
14   3: // En la etapa 3, el motor retrocede y se activa el conteo
15    Ch1Enable := TRUE;
16    Ch1Reset := FALSE;
17    marcha_atras := TRUE;
18    marcha_adelante := FALSE;
19 ELSE // En cualquier otro caso, para el motor y se activa el conteo
20   Ch1Enable := TRUE;
21   Ch1Reset := FALSE;
22   marcha_atras := FALSE;
23   marcha_adelante := FALSE;
24 END_CASE;

```

calculos_velocidad:

En esta sección se realizan la selección de la frecuencia a aplicar en función de la selección desde el HMI, la toma de medidas durante la prueba realizada y el cálculo del deslizamiento del motor:

```

0  Gestión de la frecuencia a aplicar
1  // Según el selector de velocidad (de 1 a 3), activar su correspondiente
2  // multivelocidad e indicar la frecuencia que se va a aplicar al motor
3  CASE selector_velocidad OF
4      1:
5          multivelocidad_1 := TRUE;
6          multivelocidad_2 := FALSE;
7          frecuencia_aplicada := 30;
8      2:
9          multivelocidad_1 := FALSE;
10         multivelocidad_2 := TRUE;
11         frecuencia_aplicada := 50;
12      3:
13         multivelocidad_1 := TRUE;
14         multivelocidad_2 := TRUE;
15         frecuencia_aplicada := 80;
16  END_CASE;

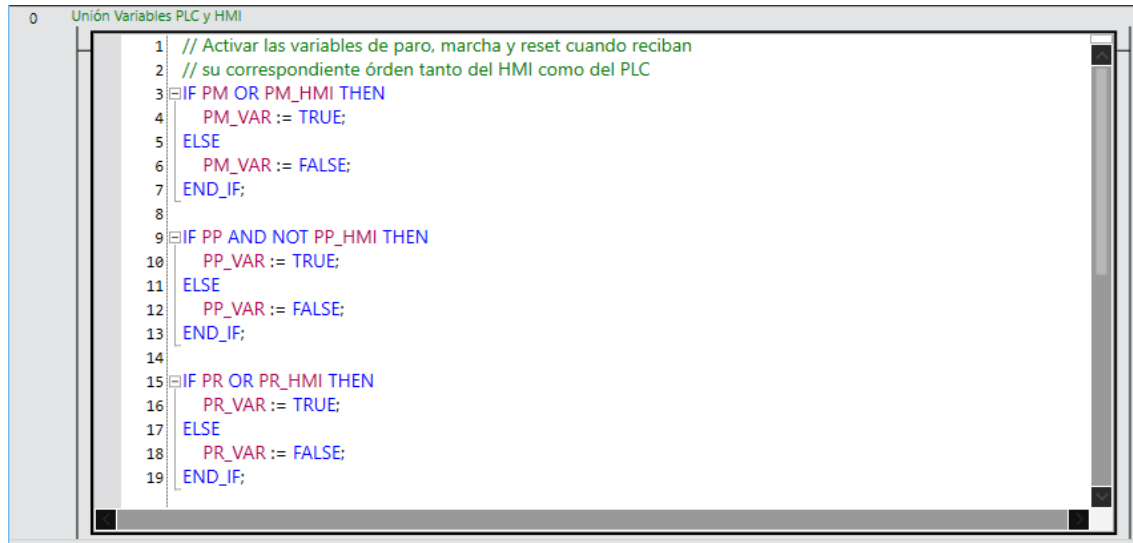
1  Almacenaje de medidas
1  // Según el selector de velocidad (de 1 a 3), almacenar la medida
2  // de velocidad en mm/s y RPM en sus correspondientes variables
3  IF state = 1 THEN
4      CASE selector_velocidad OF
5          1:
6              rpm_30hz := rpm;
7              mm_s_30hz := mm_s;
8          2:
9              rpm_50hz := rpm;
10             mm_s_50hz := mm_s;
11          3:
12             rpm_80hz := rpm;
13             mm_s_80hz := mm_s;
14      END_CASE;
15  END_IF;

2  Cálculo del deslizamiento del motor
1  // Cálculo de la velocidad síncrona del motor
2  velocidad_sincrona := TO_REAL((60 * frecuencia_aplicada) / 2);
3
4  // Cuando el motor se esté moviendo, calcular su deslizamiento
5  IF state = 1 OR state = 3 THEN
6      deslizamiento := ((velocidad_sincrona - (TO_REAL(rpm) * 15)) / velocidad_sincrona) * 100;
7  ELSE // Si el motor está parado, su deslizamiento será 0
8      deslizamiento := 0;
9  END_IF;

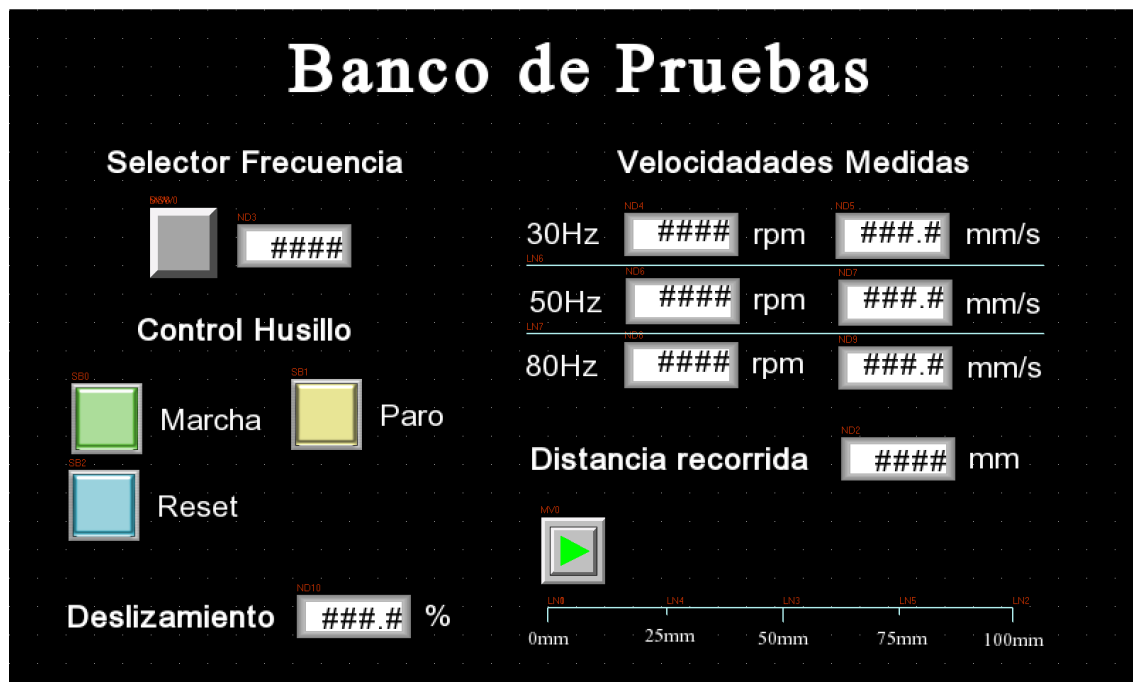
```

HMI_PLC:

En este bloque se unen las órdenes duplicadas provenientes de la botonera y del HMI en una sola variable:

**- Programa HMI:**

Este programa solo tiene una pantalla, que es la siguiente:



El selector de frecuencia es un selector de múltiples estados para seleccionar la frecuencia que se le aplicará al motor.

Los botones debajo de “Control Husillo” realizan la misma función de marcha, paro y reset que la botonera.

El deslizamiento es un visor numérico que muestra el deslizamiento del motor en %.

Las velocidades medidas son las medidas de RPM y mm/s realizadas durante la prueba para cada una de las 3 frecuencias.

Por último se muestra la distancia recorrida en mm con un visor numérico y gráficamente con un “moving component”.

8. Tabla de medidas.

Después de haber probado el programa, realizamos la siguiente tabla con los resultados obtenidos:

Frecuencia aplicada	Velocidad giro del husillo (RPM)	mm/s medidos
30 Hz	57 rpm	4,8 mm/s
50 Hz	96 rpm	8,0 mm/s
80 Hz	152 rpm	12,7 mm/s

Además, también comprobamos con un tacómetro que las medidas eran correctas.

9. Problemas encontrados y solución implementada.

El primero de los problemas fue que no contaba el encoder. Tras varias pruebas, vimos que una de las fases no estaba contando correctamente. Por tanto, lo desmontamos y pusimos otro encoder. También tuvimos que cambiar algunas variables en el programa ya que el encoder nuevo no tenía la misma resolución que el anterior.

Otro problema fue que la variable los cálculos de velocidad estaban dando error. Esto sucedía porque en cierto momento del cálculo se salían del máximo valor que admite su tipo de datos. La solución fue cambiar el orden de cálculo para que el valor nunca se salga de los valores permitidos.

También comprobamos que 25mm de desplazamiento (como inicialmente dice el enunciado del proyecto) era poco para tomar una buena medida. Por tanto, lo cambiamos a 100mm.

10. Mejoras propuestas.

Se podrían añadir más velocidades de prueba para así tener más medidas del funcionamiento del motor.

También se podría añadir un gráfico en el HMI donde se muestre la variación de las velocidades y el deslizamiento del motor en función de las frecuencias aplicadas.

Otra posible mejora sería la de almacenar todas las medidas en una base de datos (por ejemplo, en un Excel o un archivo .csv). Así, por ejemplo, se podrían comparar las medidas de diferentes años para comprobar el desgaste que está teniendo el motor.