

Lectura de Señales Analógicas

1. Objetivos.

- Realizar la configuración de las mediciones analógicas utilizando el PLC S7-300.
- Realizar el escalamiento de las señales analógicas

2. Introducción.

Para manejar señales análogas hay que tener en cuenta las direcciones de las tarjetas análogas, estas tarjetas digitalizan las señales analógicas del proceso para que puedan ser procesadas por el CPU. El valor medido ya digitalizado se almacena en una memoria contenida en la tarjeta, de allí se transfiere al CPU, donde prosigue su procesamiento.

Para direccionar las señales análogas hay que tener en cuenta que **son palabras** las que se van a utilizar, por lo tanto hay que **direccionarlos como palabras**, además, hay que tener en cuenta siempre la letra **P** de **periferia para direccionarlos** de manera adecuada. Por ejemplo si tengo una tarjeta análoga en la dirección **100**, entonces su correcta dirección sería: **PEW100**

Si desea el contenido de **PEWxx** puede ser copiada hacia una **marca** por medio de la instrucción **MOVE**.



3. Verificación en la configuración del Hardware

Configure los módulos de entrada y salida de las tarjetas análogas de su PLC en el HW

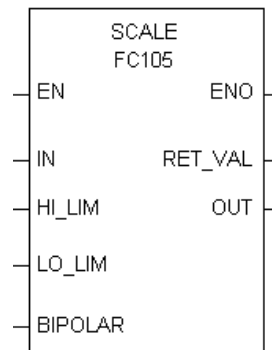
Slot	Módulo	Referencia	Firmware	Dirección MPI	Dirección E	Dirección S	Com
1							
2	CPU 313C-2 DP(1)	6ES7 313-6CF03-0AB0	V2.0				
X2	DP				1023*	124...	
2.2	DI16/DO16				125...	124...	
2.4	Contaje				768...	768...	
3							
4	AI8x12Bit	6ES7 331-7KF02-0AB0			256...271		
5	AO4x12Bit	6ES7 332-5HD01-0AB0				272...279	

Por ejemplo, para este caso según lo observado en el Hardware el direccionamiento de los canales de entrada y salida análogo sería los siguientes:

Entradas Análogas	Salidas Análogas
Ch-0 PEW256	Ch-0 PAW272
Ch-1 PEW258	Ch-1 PAW274
Ch-2 PEW260	Ch-2 PAW276
Ch-3 PEW262	Ch-3 PAW278
Ch-4 PEW264	
Ch-5 PEW266	
Ch-6 PEW268	
Ch-7 PEW270	

4. Escalamiento de la señal de analógica de entrada.

Como la señal de entrada analógica esta convertida en bits, si queremos escalarlo a unidades de ingeniería o valores por unidad o quizás porcentuales, tenemos que utilizar la función de **escalamiento** del programa:



La función de escalamiento, **SCALE**, toma un valor entero en la entrada **IN** y lo convierte en un valor real, convirtiéndolo a escala en un rango comprendido entre un límite inferior (**LO_LIM**) y un límite superior (**HI_LIM**). El resultado se escribe en la salida **OUT**. La función **SCALE** aplica la fórmula siguiente:

$$OUT = [((float(IN) - K1) / (K2 - K1)) * (HI_LIM - LO_LIM)] + LO_LIM$$

Las constantes K1 y K2 se aplican de forma diferente, dependiendo de si el valor de entrada es **BIPOLAR** o **UNIPOLAR**.

BIPOLAR	El valor entero de entrada debe estar entre -27648 y 27648, por lo tanto, K1 = -27648.0 y K2 = +27648.0
UNIPOLAR	El valor entero de entrada debe estar entre 0 y 27648, por lo tanto, K1 = 0.0 y K2 = +27648.0

Si el valor entero de entrada es mayor que K2, la salida (OUT) se une a HI_LIM y se indica un error. Si el valor entero de entrada es menor que K1, la salida se une a LO_LIM y se indica un error.

Se puede efectuar la **conversión escalar inversa** programando los límites de tal forma que el **valor en el campo** límite inferior sea mayor que el **valor en el campo** límite superior (LO_LIM > HI_LIM).

En esta conversión escalar inversa, el valor de la salida disminuye cuando aumenta el valor de la entrada.

Parámetros de la función SCALE:

Parámetro	Declaración	Tipo de datos	Area de memoria	Descripción
EN	Entrada	BOOL	E, A, M, D, L	La entrada de habilitación con estado de señal 1 activa el cuadro.
ENO	Salida	BOOL	E, A, M, D, L	La salida de habilitación tiene el estado de señal 1 si la función se ejecuta sin errores.
IN	Entrada	INT	E, A, M, D, L, P, o constante	Valor de entrada a convertir a escala en valor REAL.
HI_LIM	Entrada	REAL	E, A, M, D, L, P, o constante	Límite superior del rango escalar.
LO_LIM	Entrada	REAL	E, A, M, D, L, P, o constante	Límite inferior del rango escalar.
BIPOlar	Entrada	BOOL	E, A, M, D, L	El estado de señal 1 indica que el valor de entrada es bipolar; con el estado de señal 0 indica que es unipolar.
OUT	Salida	REAL	E, A, M, D, L, P	Resultado de la conversión a escala.
RET_VAL	Salida	WORD	E, A, M, D, L, P	Da el valor W#16#0000 cuando la función se ejecuta sin errores; si los valores son distintos de W#16#0000, véase la información sobre errores.

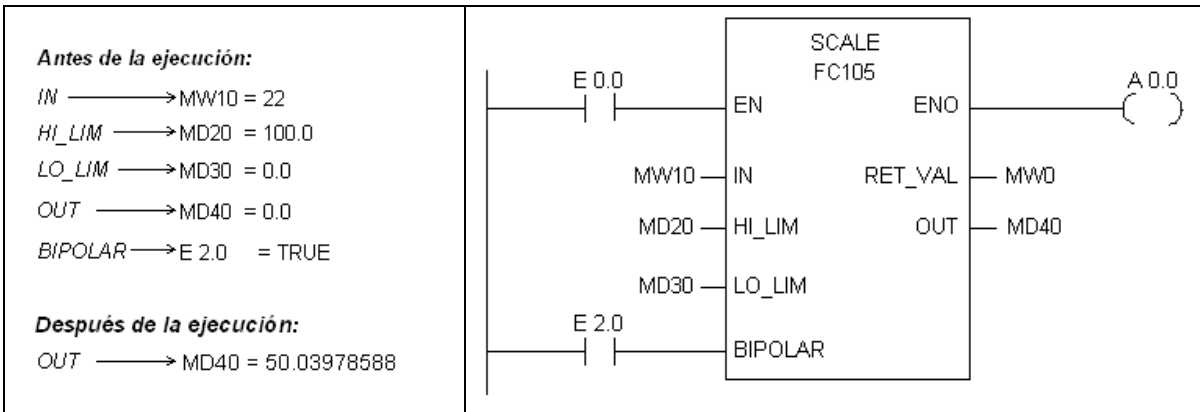
5. Ejemplo de Escalamiento a la Entrada y a la Salida.

5.1 Leer y normalizar un valor analógico a la entrada (FC105)

La función SCALE toma un valor entero en la entrada **IN** y lo convierte en un valor real, convirtiéndolo a escala en un rango comprendido entre un límite inferior y un límite superior (**LO_LIM** y **HI_LIM**). El resultado se escribe en la salida **OUT**.

La figura muestra un ejemplo: La función se ejecuta cuando el estado de señal de la entrada E0.0 es 1 (activada). En este ejemplo, el valor entero 22 se convierte en un valor **REAL escalado** entre 0.0 y 100.0, y éste se escribe en la salida OUT. El valor de entrada es BIPOlar, tal como lo indica el estado de señal de la entrada E2.0.

Si la función se ejecuta sin errores, los estados de señal de la salida de habilitación (ENO) y de la salida A0.0 se ponen a 1 (activadas).

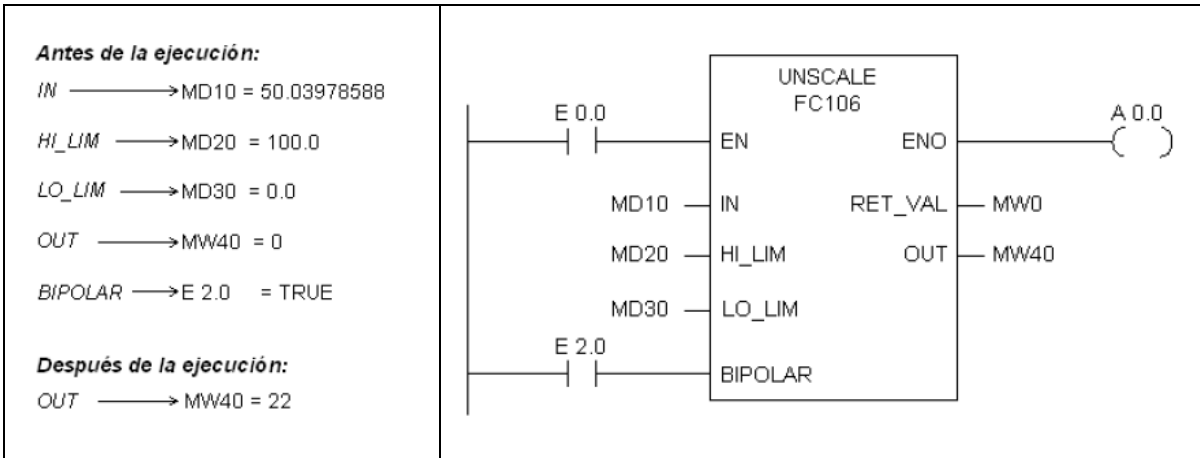


Tarea:

- Verifique con la ecuación de conversión dada que el resultado del valor de entrada escalado es el correcto.

5.2 Escalar un valor analógico a la salida (FC106)

La función UNSCALE toma en la entrada **IN** un valor real que está ajustado a escala en un rango comprendido entre un límite inferior y un límite superior (LO_LIM y HI_LIM), y lo convierte en un valor entero. El resultado se escribe en la salida **OUT**.



Escalamiento de Señales de Entrada y Salida Analógicas

1. Descripción.

Se tiene un tanque de agua, el cual se desea mantenerlo siempre lleno.

El ingreso de agua al tanque se controla mediante una válvula eléctrica proporcional de 0...10V, cuya apertura y cierre es inversamente proporcional al porcentaje del nivel de agua faltante, así por ejemplo, si el nivel de agua está en 80%, la válvula se abrirá en 20%.

El PLC recibe la información del nivel de agua a través de un sensor de nivel con transductor de presión incorporado de 0...10V proporcional al nivel de agua 0...100%

Además, la salida del agua se gobierna por medio de otra válvula y de manera manual.

2. Bloques disponibles:

En el programa emplearemos dos bloques de la librería del STEP 7:

- FC 105: SCALE
- FC 106: UNSCALE

Estos módulos funcionales ejecutan conversiones entre el margen nominal de una tarjeta analógica y un margen normalizado, que el usuario puede especificar.

3. Procedimiento.

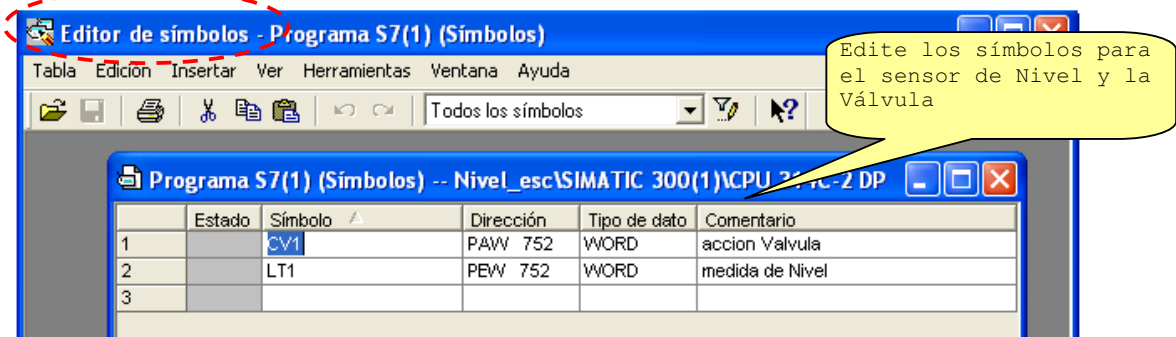
Creación de un proyecto

1. Primero creamos un proyecto con un equipo SIMATIC (S7 300). Si desea use el asistente para crear el proyecto o cree su proyecto manualmente. A nuestro proyecto lo llamaremos **Nivel_esc**
2. En la configuración Hardware verifique la dirección de la entrada y salida analógica

Slot	Módulo	Referencia	Firmware	Dirección MPI	Dirección E	Dirección S
1						
2	CPU 314C-2 DP	6ES7 314-6CG03-0AB0	V2.6	2		
X2	DP				1023*	
2.2	DI24/DO16				124...126	124...125
2.3	AI5/AO2				782...787	782...785
2.4	Contaje				788...789	788...789
2.5	Posicionamiento				784...799	784...799
3						

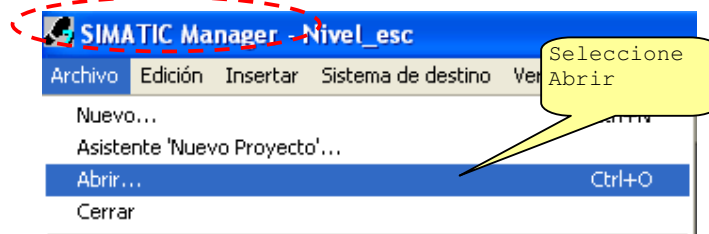
Creación de la Tabla de símbolos:

3. Luego creamos la tabla de símbolos desde el menú **Herramientas | Tabla de símbolos**, como se muestra en la figura:

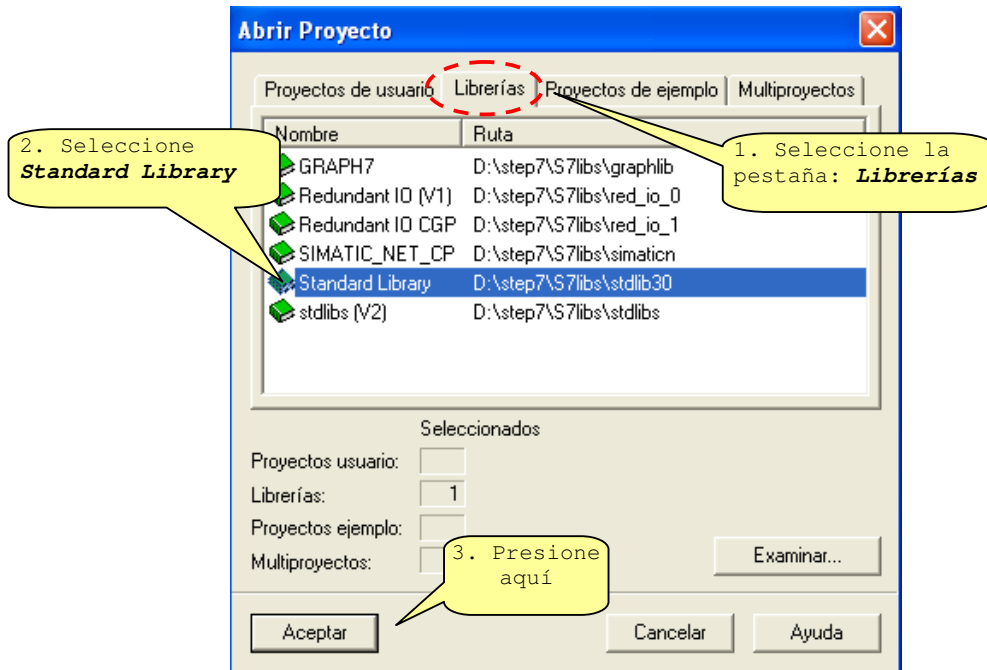


Cargar los bloques de escalamiento

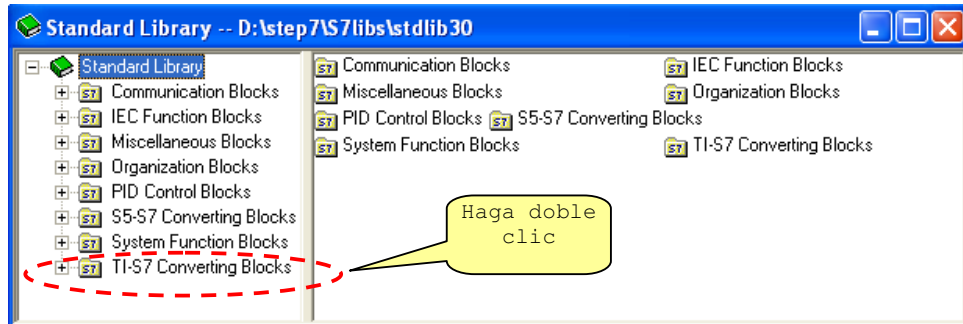
4. Desde el SIMATIC Manager, seleccionamos en el menú **Archivo | Abrir**



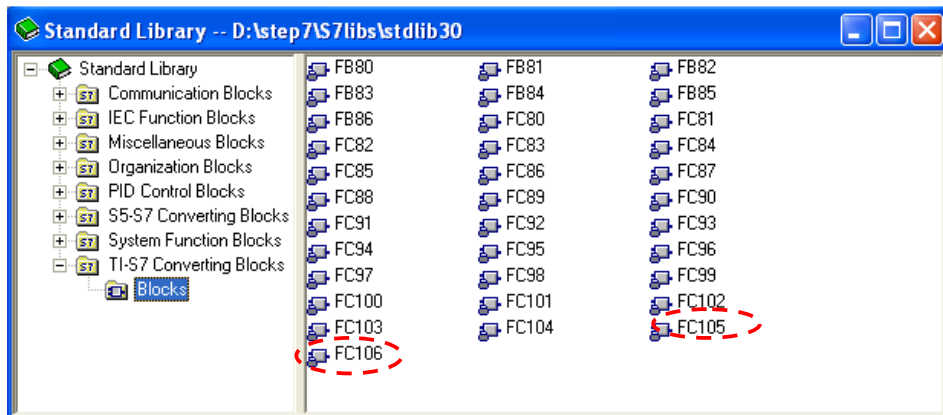
Se presenta la siguiente ventana, donde permite elegir la librería adecuada:



5. Luego seleccionamos **TI-S7 converting blocks**



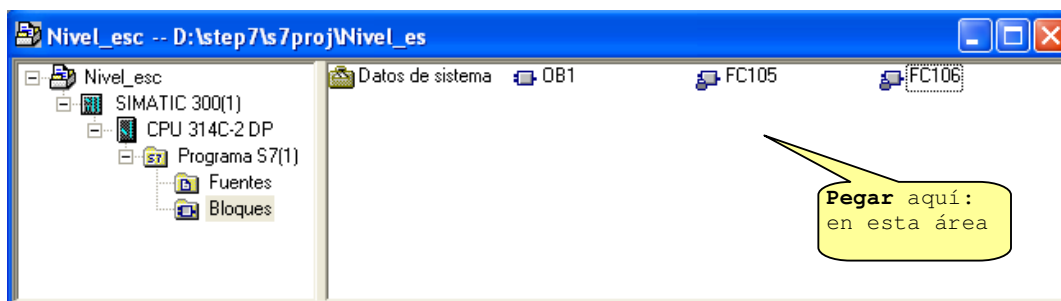
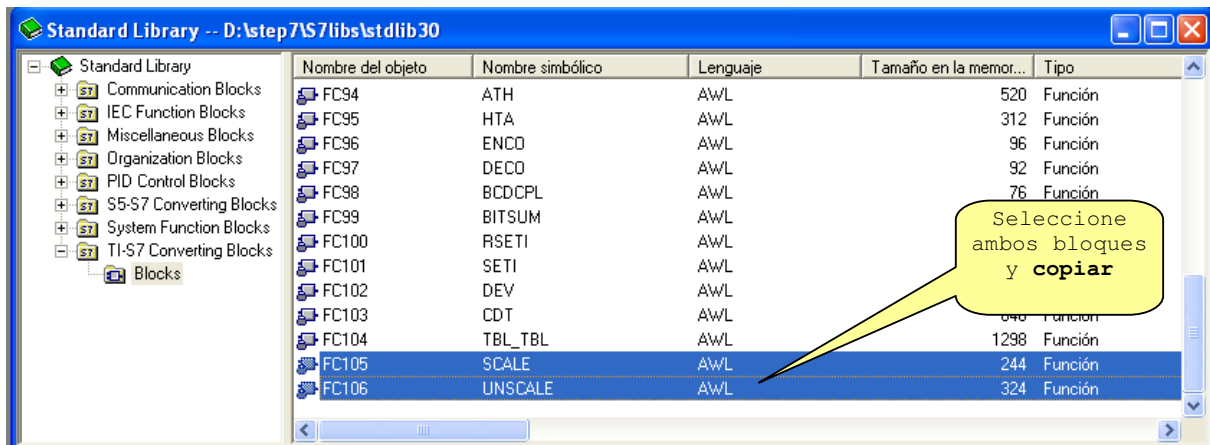
Aquí podemos observar los Bloques **FC 105 (SCALE)** y **FC 106 (UNSCALE)**



Si desea ver los detalles de los Bloques, active desde el menú la opción **ver detalles**. Se vera información adicional de cada bloque como se muestra en la figura de abajo.

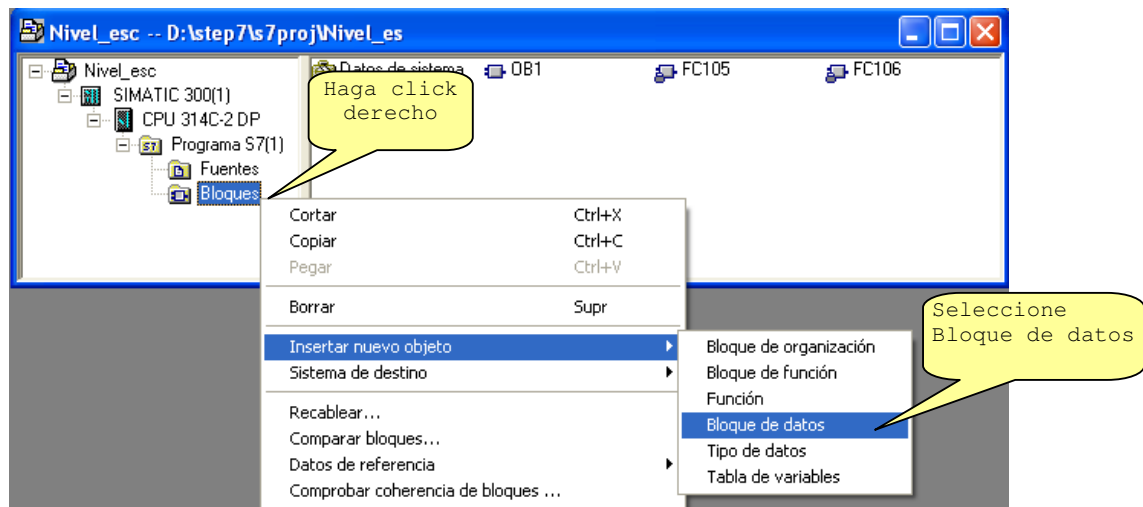
Nombre del objeto	Nombre simbólico	Lenguaje	Tamaño en la memor...	Tipo
FC94	ATH	AWL	520	Función
FC95	HTA	AWL	312	Función
FC96	ENCO	AWL	96	Función
FC97	DECO	AWL	92	Función
FC98	BCDCPL	AWL	76	Función
FC99	BITSUM	AWL	100	Función
FC100	RSETI	AWL	178	Función
FC101	SETI	AWL	178	Función
FC102	DEV	AWL	656	Función
FC103	CDT	AWL	640	Función
FC104	TBL_TBL	AWL	1298	Función
FC105	SCALE	AWL	244	Función
FC106	UNSCALE	AWL	324	Función

6. Luego seleccione los dos bloques **FC 105 (SCALE)** y **FC 106 (UNSCALE)** y debemos copiar (**Ctrl + C**) y pegar (**Ctrl + V**) en nuestro proyecto (en la ventana de abajo)

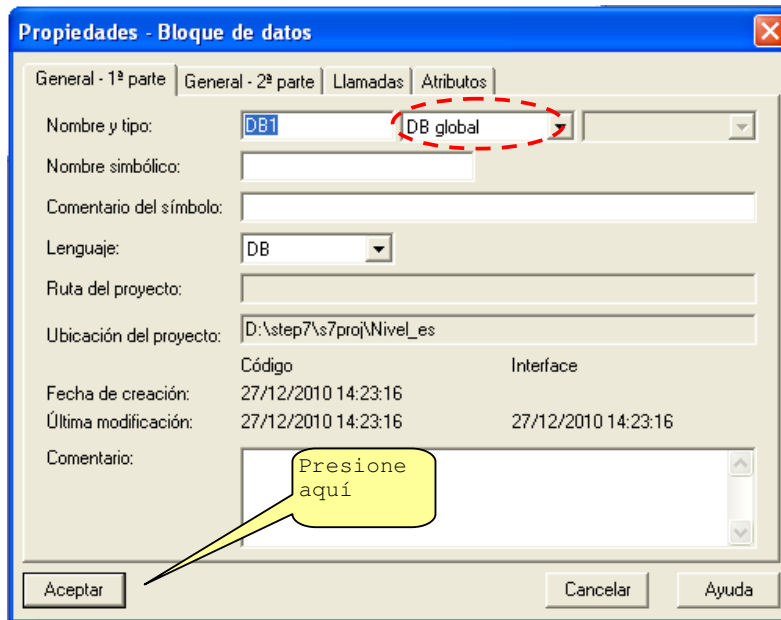


Crear el bloque de datos DB1

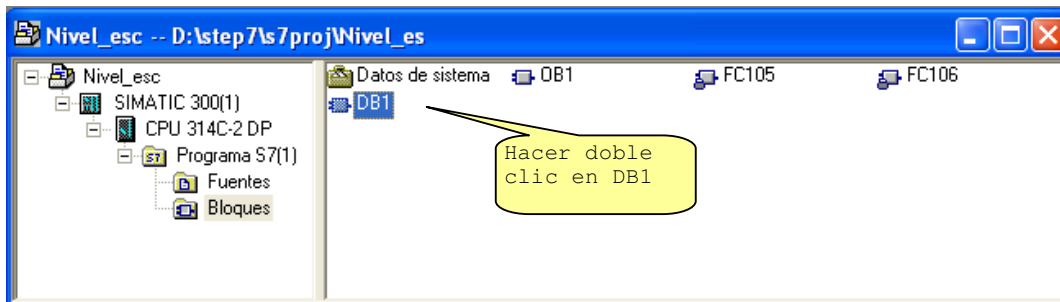
7. Estando en **Bloques**, damos click derecho con el mouse e insertamos el bloque de datos DB1.



8. Este bloque de datos nos permitirá manejar los bloques escalados. Parametrizamos como DB1, tipo **global** y aceptamos.



9. Luego hacemos Doble click sobre el DB1 creado para configurarlo.

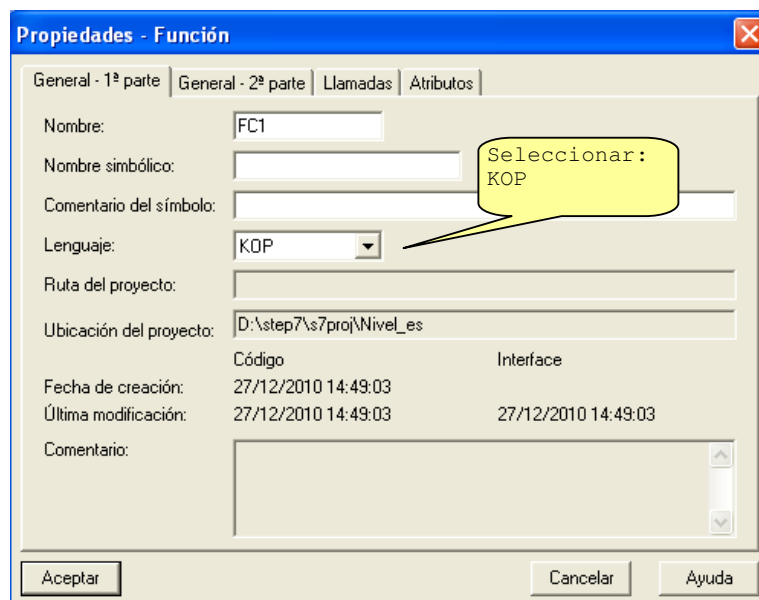
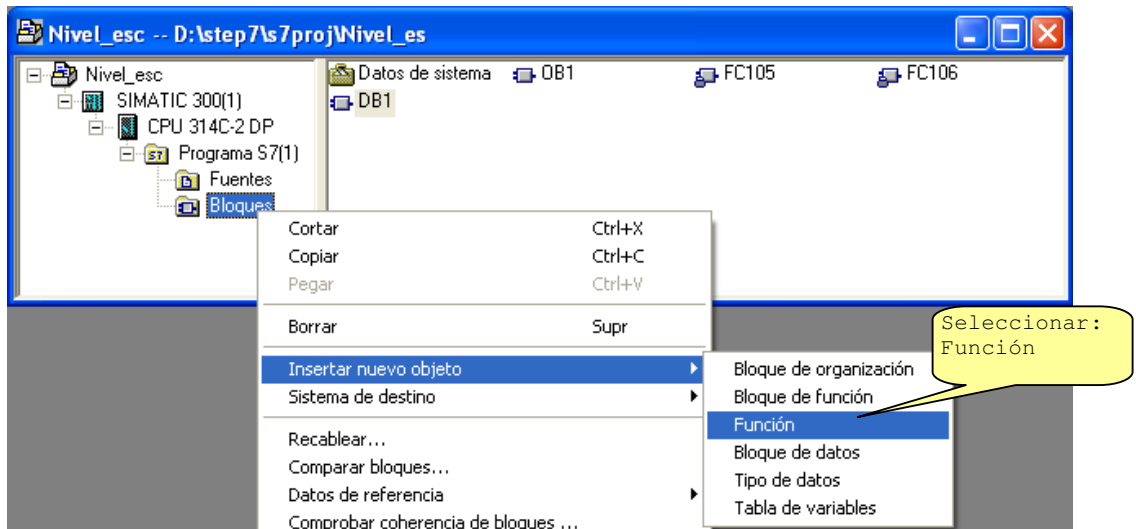


10. Ingresamos los siguientes datos en nuestro DB1. Para este ejemplo, consideramos todas las variables de los bloques FC105 y FC 106.

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	Sensor	WORD	W#16#0	Variable provisional
+2.0	Hi_lim	REAL	1.000000e+002	
+6.0	Lo_lim	REAL	0.000000e+000	
+10.0	IN_escalado	REAL	0.000000e+000	
+14.0	Error	WORD	W#16#0	
+16.0	Out_escalado	REAL	0.000000e+000	
+20.0	Hi_lim1	REAL	1.000000e+002	
+24.0	Lo_lim1	REAL	0.000000e+000	
+28.0	Valvula	WORD	W#16#0	
+30.0	Error1	WORD	W#16#0	
=32.0		END_STRUCT		

Programar el bloque de programa FC1

De la misma que se creó el bloque de datos, crear el bloque de programa FC1.



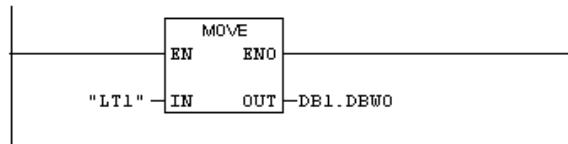
Implemente el siguiente programa dentro de la función FC1:

FCl : Título:

Comentario:

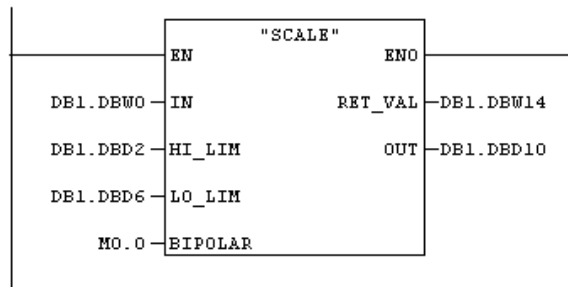
Segm. 1 : Título:

Mover sensor de Nivel a Bloque de datos



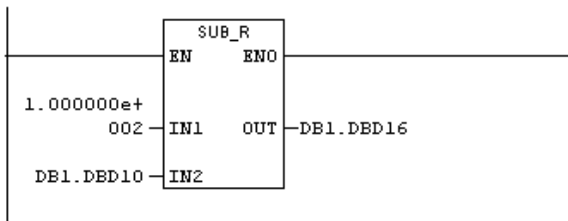
Segm. 2 : Título:

Escalar Sensor de Nivel



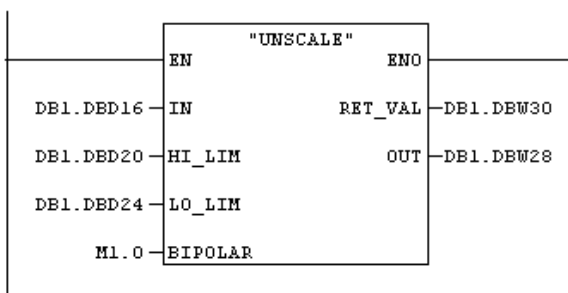
Segm. 3 : Título:

manejo de la valvula



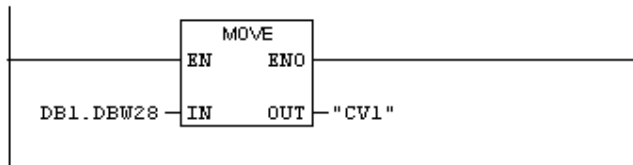
Segm. 4 : Título:

Desescala valvula de control

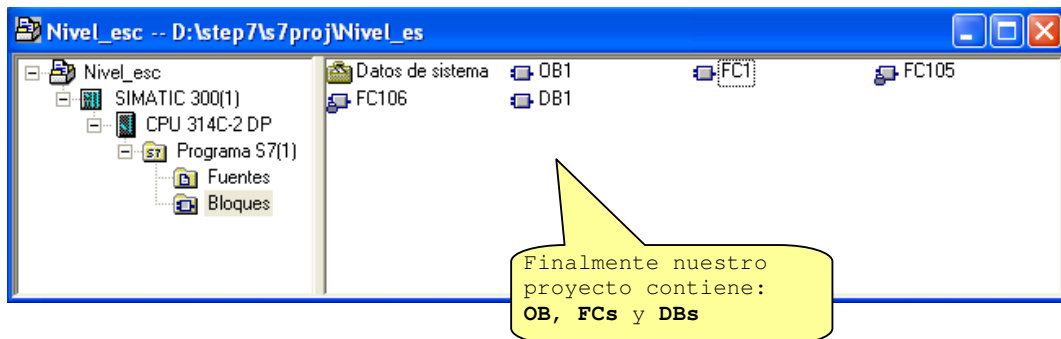


Segm. 5 : Título:

mover Valor desde bloque de datos a valvula de control

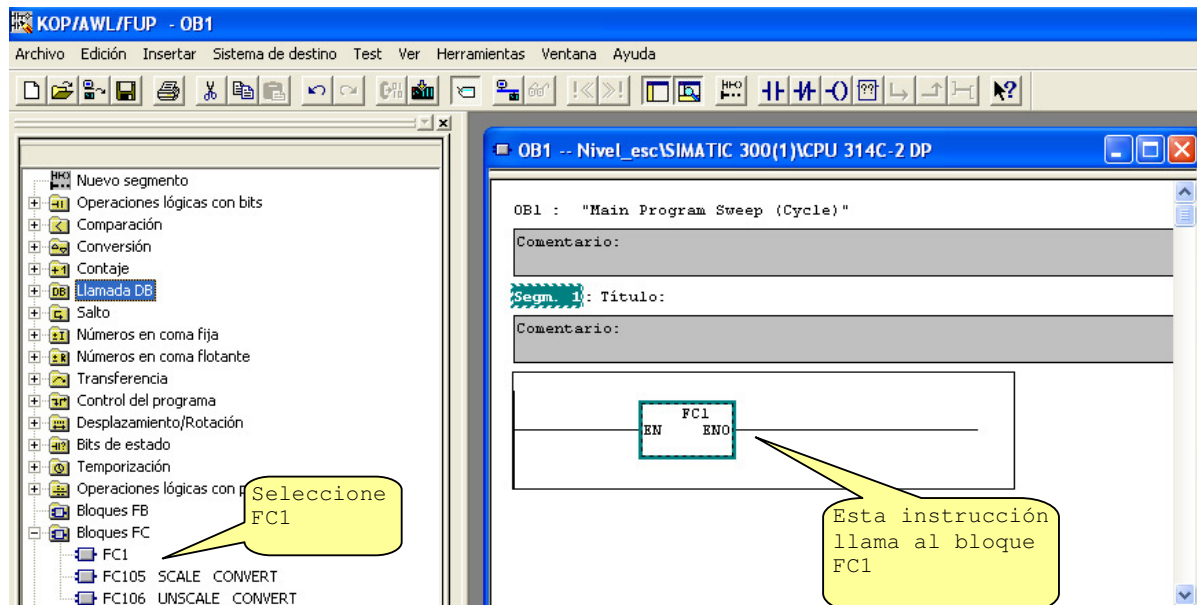


Grábelo y cierre:



Programar el bloque principal OB1

En el **OB1**, hacemos el llamado del bloque de programa FC1:



Listo para transferir el **DB1**, **FC1**, **FC105**, **FC106** y **OB1** creados.