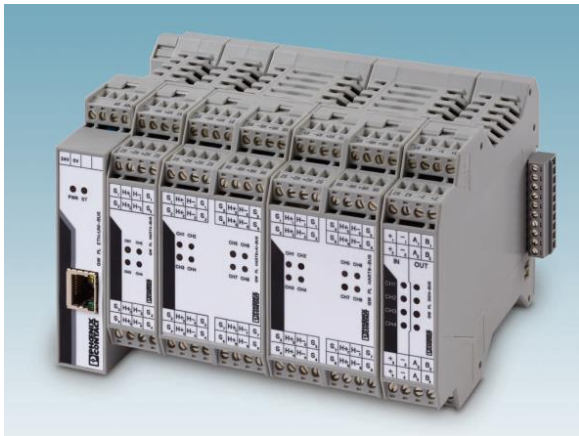


Guía de Inicio Rápido

Gateway HART con

PLCnext



©PHOENIX CONTACT 2021-02-04

1 Descripción

El presente documento describe un ejemplo de aplicación para leer los valores de dispositivos 4..20 mA HART, a través de Modbus TCP. Esto se logra por medio de la utilización de Gateway y concentradores ethernet. La ventaja de ésta configuración, es que no es necesario modificar el cableado de los dispositivos analógicos hacia el PLC para obtener información adicional.

Tabla de contenido

1	Descripción	1
2	Requisitos de hardware	3
3	Requisitos de software	3
4	Instalación del hardware	3
5	Configuración del Gateway	4
	Configuración General	5
	Configuración de LAN	6
	Configuración de protocolo	6
	Diagnósticos	6
	Mantenimiento del dispositivo	7
6	Proyecto de PCWorx	7
	Configuración del hardware	7
	Creación de las variables	8
	Configuración del enlace Modbus TCP	9
	Adquisición de datos	9

2 Requisitos de hardware

Mando:

Uno de los siguientes:

- PLCNext AXL F 1152/2152/4152
- AXL F AI2 AO2 1H

Multiplexor ethernet-HART:

Uno de los siguientes

- GW PL ETH/BASIC-BUS
- GW PL ETH/UNI-BUS

Convertidor de protocolos:

Uno o más de los siguientes

- GW PL HART4-BUS
- GW PL HART8-BUS
- GW PL HART8+AI-BUS

Para esta demo en particular, se utilizó un AXC F 1152, con un multiplexor GW PL ETH/BASIC-BUS, y un convertidor, GW PL HART4-BUS.

3 Requisitos de software

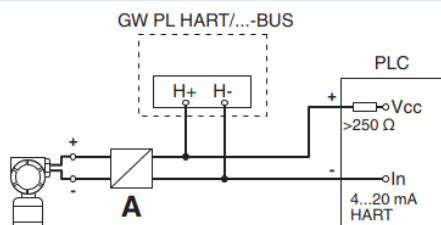
PLCNext Engineer, versión 2021 o superior

Navegador web compatible (Chrome/Firefox/Edge/Opera)

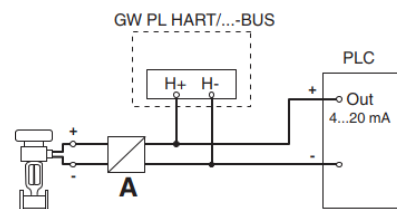
4 Instalación del hardware

Partiendo de una conexión estándar de un dispositivo HART, se realiza el cableado hacia el convertidor, siguiendo las instrucciones correspondientes al tipo de convertidor y dispositivo en particular:

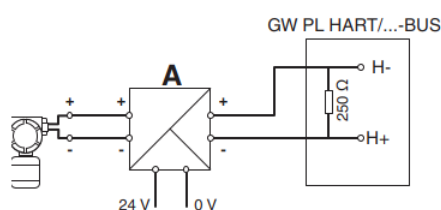
A – Entrada analógica de PLC *



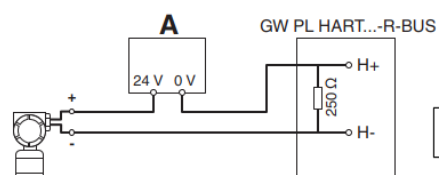
B Salida analógica de PLC



C Entrada analógica a HART...-BUS



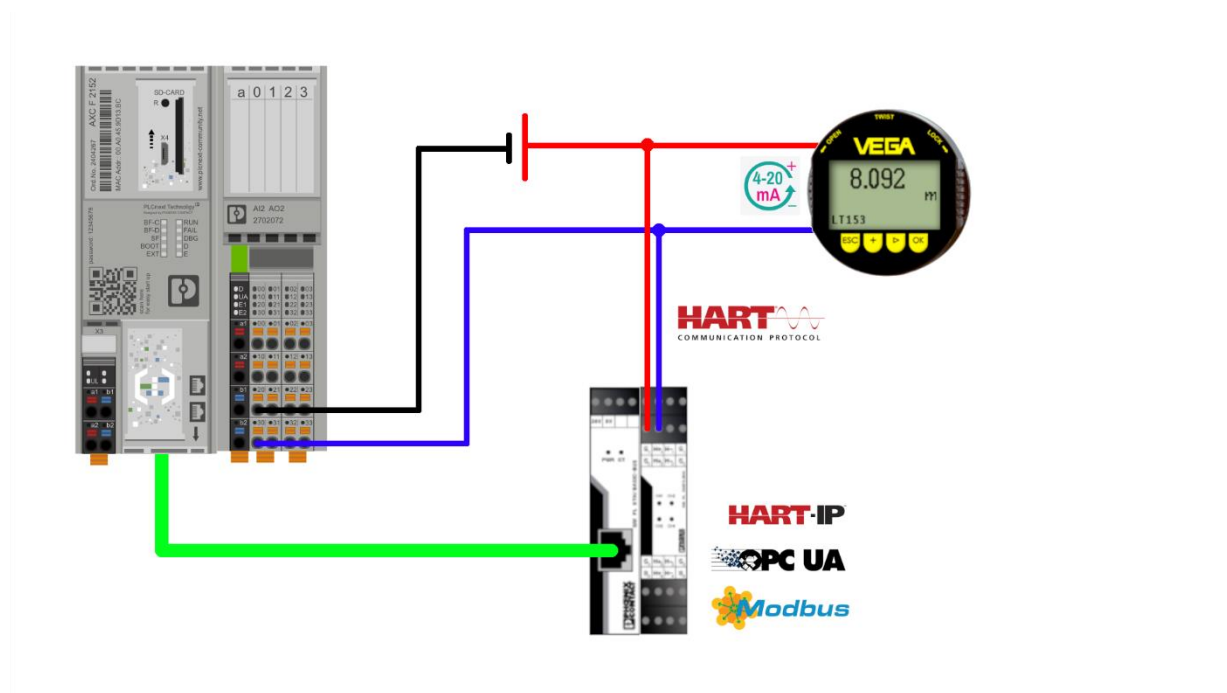
D Entrada analógica a HART...-R-BUS



Si se utiliza el módulo **GW PL HART8+AI-BUS**, puede alimentarse directamente un sensor analógico de 2 hilos pasivo, ya que el pin H+ suministra la corriente de lazo necesaria. En caso de que el sensor sea de 4 hilos o se utilice un PLC con salida de alimentación de lazo incorporada, debe utilizarse un convertidor/aislador.

* Si se utiliza un sensor de 2 hilos pasivo, junto con un módulo GW PL HART...-BUS, y en adición la lectura analógica del PLC no posee la corriente de lazo, debe añadirse una fuente de alimentación o un convertidor en el lazo para suministrar la corriente de trabajo (indicados como **A** en el diagrama).

Para ésta aplicación, se utilizó la siguiente configuración de dispositivos



5 Configuración del Gateway

Al energizarse por primera vez, será necesario configurar la dirección IP del Gateway para que coincida con el resto de los dispositivos de la red. Además, tendremos que activar la comunicación via Modbus TCP para éste proyecto en cuestión.

De fábrica, el dispositivo viene con la siguiente dirección asignada:

- Dirección IP: **192.168.254.254**
- Máscara de subred: **255.255.255.0**
- Gateway: **0.0.0.0**

Gateway HART con PLCnext

Credenciales de acceso:

- Usuario: **Admin** (debe respetarse la A mayúscula)
- Contraseña: **admin**

Los pasos a seguir son:

1. Conectar la PC al Gateway de forma directa utilizando un cable Ethernet
2. Asignarle a la PC una configuración IP **dentro de la misma subred** que la IP de fábrica del dispositivo. Por ej. 192.168.254.**210**, Máscara 255.255.255.0
3. A través de un navegador web, ingresar a la dirección de fábrica del Gateway (192.168.254.254)
4. Se visualizará la siguiente pantalla, donde debemos ingresar las credenciales de acceso

5. Habiendo iniciado sesión, podemos observar el menú de bienvenida, donde tendremos acceso a las configuraciones básicas y vistas de diagnóstico del dispositivo

6. A partir de aquí, podemos ingresar a cada una de las pestañas:

Configuración General

Aquí podemos modificar el nombre del dispositivo, así como la fecha y hora en forma manual o automática utilizando distintos métodos.

Configuración de LAN

General Settings | **LAN Settings** | Protocol Settings | Diagnostics | Device Maintenance

LAN IP Address

☐ Automatic address assignment (DHCP)

☒ Manual address assignment

IP Address: 192.168.1.103

Subnet Mask: 255.255.255.0

Default Gateway:

Hardware Address

MAC Address: A8:74:1D:9D:BB:D6

Apply

Desde aquí podemos modificar la dirección IP del dispositivo. Ante cualquier cambio, el dispositivo se reiniciará por completo para poder aplicarlos.

Es importante tener en cuenta que si se modifica la dirección IP del Gateway a otra subred (por ej, de 192.168.254.xxx a 192.168.1.yyy), se perderá conexión con el dispositivo hasta que se le asigne a la PC una IP en la misma subred.

Configuración de protocolo

Desde aquí se pueden activar o desactivar los distintos protocolos de comunicación que soporte el dispositivo.

General Settings | LAN Settings | **Protocol Settings** | Diagnostics | Device Maintenance

Check the protocols you wish to enable.

HART IP

☒ HART IP

HART IP listens on TCP/UDP port 5094.

ARCOM

☐ ARCOM

TCP Port: 3001

Modbus TCP

☒ Modbus TCP

Modbus TCP listens on TCP port 502.

Default configuration is active.

Apply

Modbus TCP está desactivado de fábrica, por lo cual debe activarse para éste proyecto en cuestión

Diagnósticos

General Settings | LAN Settings | Protocol Settings | **Diagnostics** | Device Maintenance

This is your current station configuration.

Slot Number 1

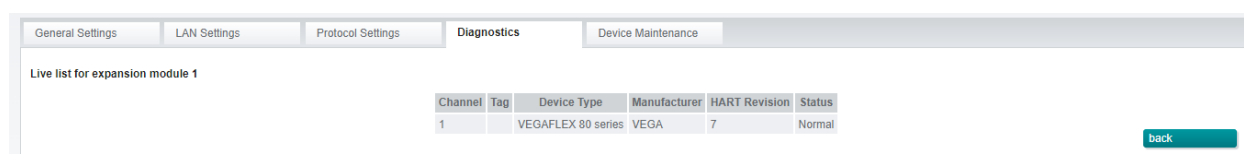
Module GW PL ETH/BASIC-BUS

Serial Number 3022027893

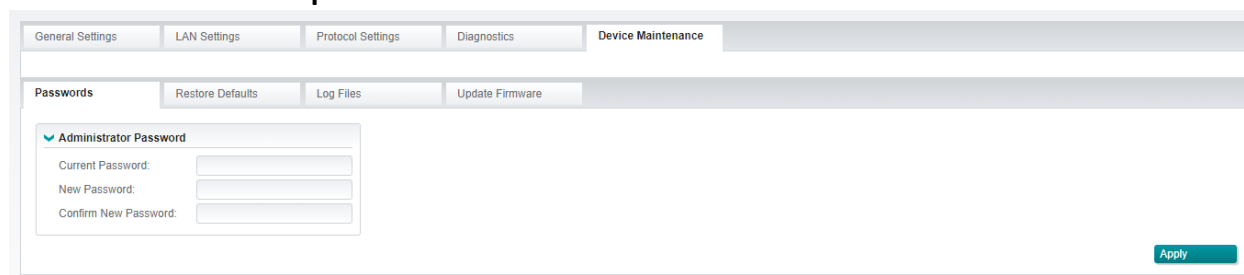
[View Live List](#)

Aquí podemos ver los dispositivos conectados. Si hacemos click en “**View Live List**” podremos ver qué dispositivos HART están conectados en un módulo en particular.

Gateway HART con PLCnext



Mantenimiento del dispositivo

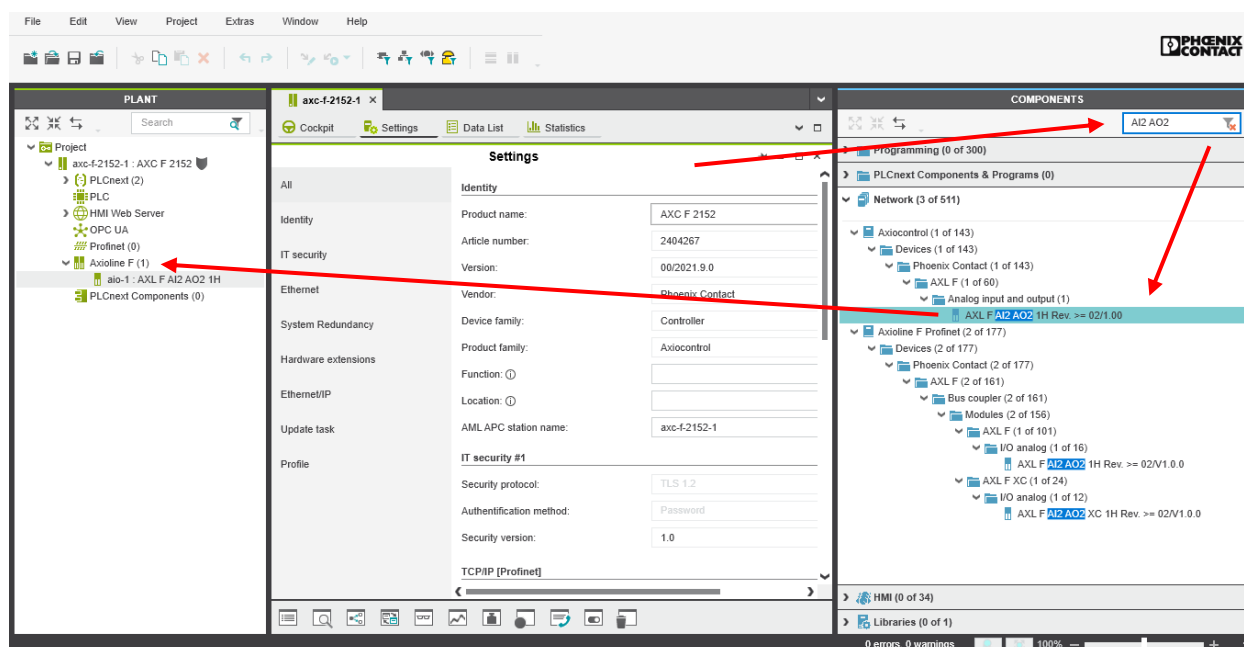


Desde aquí podemos realizar varias tareas administrativas como cambiar la contraseña de acceso, reestablecer las configuraciones de fábrica, visualizar los archivos de registro del dispositivo, o actualizar el firmware del mismo.

6 Proyecto de PCWorx

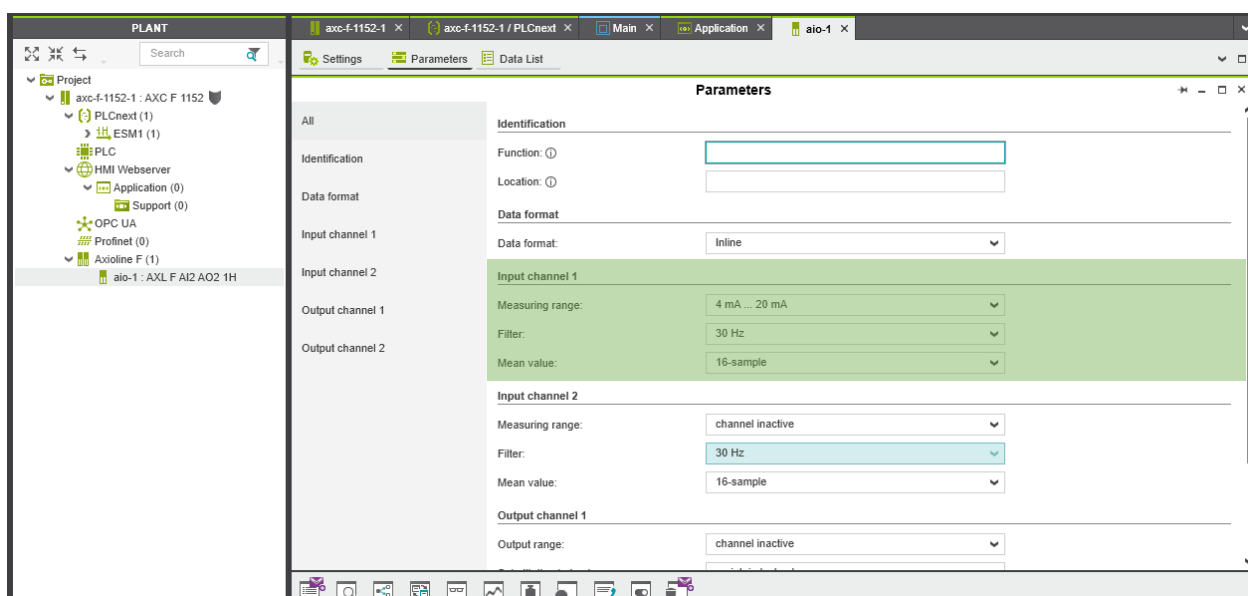
Configuración del hardware

1. Se debe crear un nuevo proyecto, con un PLC AXC F 2152 de firmware 2021.9.0 o superior, y asignarle una dirección IP, dentro de la misma subred que el gateway
2. Bajo **COMPONENTES**, buscamos el módulo “AXL F AI2 AO2” (dentro de la carpeta **Analog input and output**) y lo arrastramos a la carpeta “Axioline F” bajo la sección **PLANTA**



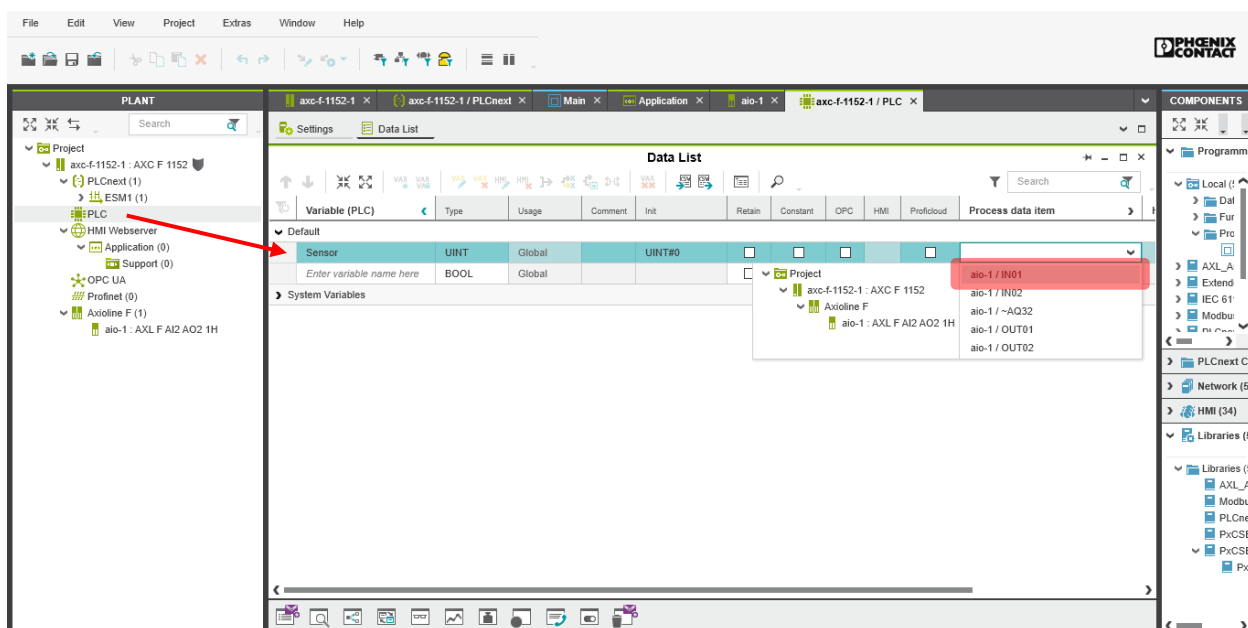
3. Por último, hacemos doble click en el módulo, y dentro de la sección “Parameters” nos aseguramos de que la lectura sea 4..20 mA en el **canal 1**. Si el resto de los canales no se usan, es preferible desactivarlos.

Gateway HART con PLCnext

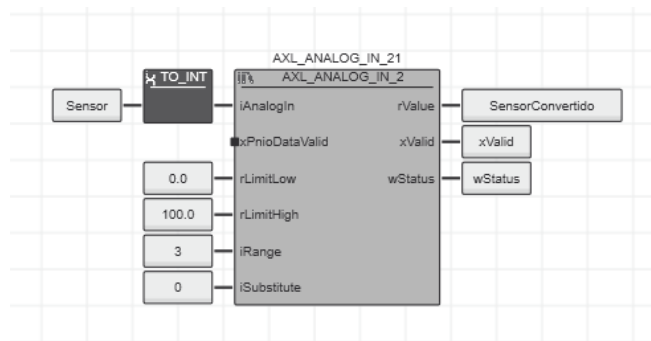


Creación de las variables

Bajo la sección **PLC** (o IEC 61131-3, dependiendo de la versión de firmware), crearemos una nueva variable llamada “Sensor”, del tipo **UINT**. En la columna “Process data item” la asociaremos con el canal de entrada analógica número 1

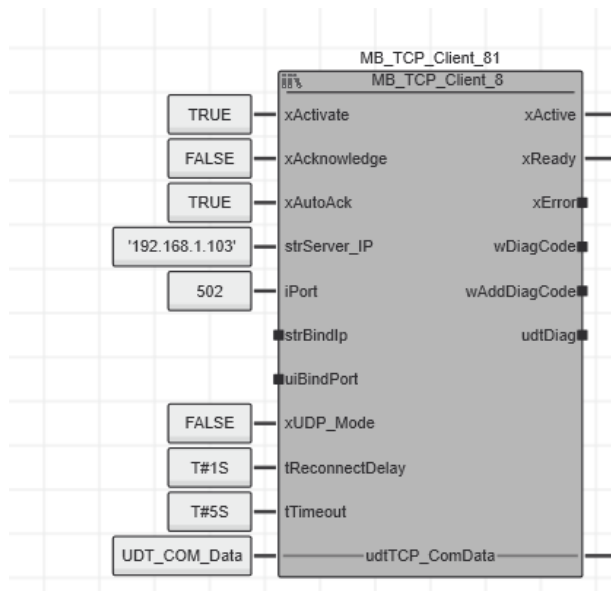


Finalmente, utilizamos cualquier método para escalar la señal analógica. Para ésta ocasión, se utilizó la biblioteca **AXL_Analog**



Configuración del enlace Modbus TCP

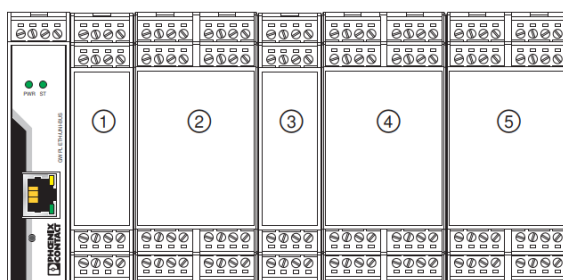
Descargaremos e insertamos la biblioteca MODBUS_TCP desde la página de descargas del PLCnext Engineer. De aquí, creamos la función Modbus_Client, la cual se encargará de conectar al Gateway y gestionar los llamados al FC que corresponda. Una vez generada una instancia, la parametrizaremos.



Adquisición de datos

Para adquirir los datos, debemos llamar a los registros de la dirección 30000, lo que quiere decir que tenemos que usar el FC4 dentro de las funciones Modbus. En el manual del Gateway, podemos observar los detalles de las direcciones a las cuales queremos leer.

Para cada módulo, tenemos que apuntar a una dirección de **esclavo** en particular, siendo el primer módulo el ID de esclavo 1, el segundo módulo es el esclavo 2, y así sucesivamente, como se observa en la siguiente imagen.



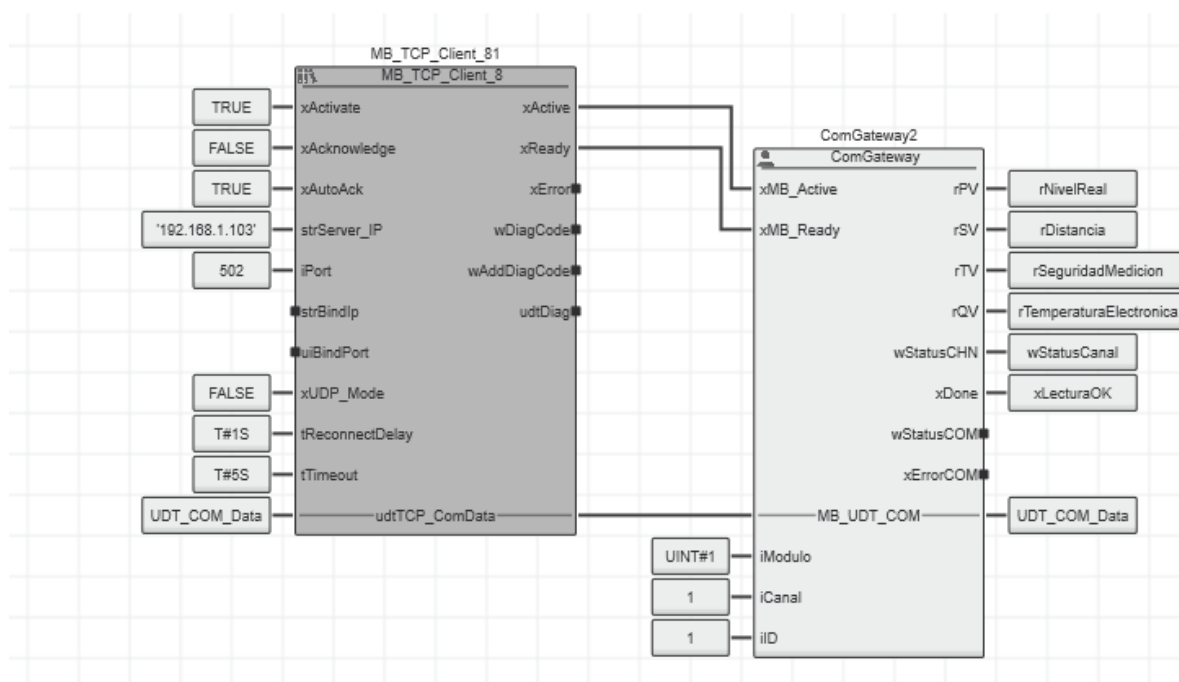
Gateway HART con PLCnext

Una vez seleccionado el número de **esclavo** modbus, debemos apuntar a qué **canal** en particular queremos apuntar, usando la tabla proporcionada en el manual. A continuación, podemos ver un fragmento de la misma

Module channel no.	HART parameter	Read/Write	Type	Modbus input register Base 1 addressing
1	Primary Variable (PV)	Read	Float	30257 (MSW) 30258 (LSW)
	Secondary Variable (SV)	Read	Float	30259 (MSW) 30260 (LSW)
	Tertiary Variable (TV)	Read	Float	30261 (MSW) 30262 (LSW)
	Quaternary Variable (QV)	Read	Float	30263 (MSW) 30264 (LSW)
	Status*	Read	Word	30265
	†Loop Current	Read	Float	33207 (MSW) 33208 (LSW)

Por ejemplo, para leer el **valor primario (PV)** de un dispositivo HART conectado al **módulo 1**, configure el cliente Modbus para solicitar los registros **30256 y 30257** del **esclavo ID 1**, con la dirección IP de la estación principal.

Para simplificar ésta tarea, se dejó preparado un bloque el cual permite seleccionar a qué módulo queremos acceder, y a qué canal en particular, además de encargarse de la conversión de los registros recibidos a un valor real. Éste bloque se interconecta con el bloque Modbus_Client, como se ve en la siguiente imagen



Donde la entrada **iModulo** es en número de módulo, la entrada **iCanal** es el canal seleccionado dentro del multiplexor, y el **iID** es un identificador único para la conexión.

Código de bloque ComGateway

```
// Primero lo primero: Establecemos comunicación con el módulo
// Para ello, tenemos que asegurarnos de que el cliente modbus conectó y está listo

xOK_Oper := xMB_Active AND xMB_Ready;

// Calculamos a qué registro tenemos que apuntar
// Fórmula:
wDir := TO_WORD(((iModulo - UINT#1)*9)+256);

// Configuramos el FC04 que necesitamos para leer el módulo

MB_TCP_FC4_31(
  xActivate := xOK_Oper,
  xAcknowledge := FALSE,
  iMT_ID := iID,
  tUpdateTime := T#500ms,
  uiUnitIdentifier := iModulo,
  wStartRegister := wDir,
  uiQuantityOfRegisters := UINT#10,

  xNDR => xDone,
  arrInputRegisters => arrDatos,
  xError => xErrorCOM,
  wDiagCode => wStatusCOM,
  wAddDiagCode => wStatusCOM,
  udtTCP_ComData := MB_UDT_COM);

// WIP:señal dummy para ver si estamos leyendo
wTest1 := arrDatos[1];
wTest2 := arrDatos[2];
wTest3 := arrDatos[3];
wTest4 := arrDatos[4];
wTest5 := arrDatos[5];
wTest6 := arrDatos[6];
wTest7 := arrDatos[7];
wTest8 := arrDatos[8];
wTest9 := arrDatos[9];

FOR indice:=1 TO 4 BY 1 DO (* 12 registros, almacenados en un array que va de 0 a 11 *)
  dwHigh := arrDatos[(indice*2)-1];
  dwLow := arrDatos[indice*2];
  dwReal := dwLow OR SHL(dwHigh,16);

  conv_r := TO_REAL(dwReal);

  CASE ((indice)) of (* consulto qué par de registros eran. El primero, el segundo,
tercero o cuarto *)
    1: rPV:=conv_r; (* Valor primario *)
    2: rSV:=conv_r; (* Valor secundario *)
    3: rTV:=conv_r; (* Valor terciario *)
    4: rQV:=conv_r; (* Valor cuaternario *)
  ELSE conv_r:=conv_r; (* Si no corresponde a ninguno, no lo guardo en
ningun lugar *)
  END_CASE;
  wStatusCHN:=arrDatos[9]; (* por ultimo, el noveno valor, es el estado de
la comunicacion. Se guarda directo como word *)
END_FOR;
```