



radwag.com

Escanee el código QR para ver materiales de investigación adicionales que podrían interesarle.
Allí encontrará más información útil de forma accesible.

Instrucciones de software

ITKP-20-01-04-20-ES

PROFINET

Protocolo de comunicación:

Módulo de pesaje MWMH

Módulo de pesaje MWLH

Módulo de pesaje MWSH

ABRIL 2020

INDICE

1. CONFIGURACIÓN DE AJUSTES DE MÓDULOS INDUSTRIALES	4
2. ESTRUCTURA DE DATOS	4
2.1. Registros de entrada	4
2.2. Registros de salida	6
3. CONFIGURACIÓN DEL MÓDULO PROFINET EN EL ENTORNO TIA PORTAL V14	9
3.1. Importación GSD	9
3.2. Configuración del módulo	11
4. APLICACIÓN DE DIAGNÓSTICO	16

1. CONFIGURACIÓN DE AJUSTES DE MÓDULOS INDUSTRIALES

La configuración de los ajustes de los módulos industriales MWMH, MWLH, MWSH para la comunicación mediante el protocolo PROFINET se realiza mediante el software MWMH MANAGER disponible en el sitio web www.radwag.com. En la pestaña COMUNICACIÓN → DISPOSITIVOS, seleccione el protocolo Profinet y luego ingrese la dirección IP, la máscara de subred y el nombre del dispositivo en la red Profinet. Para obtener más información, consulte el manual del usuario del dispositivo en www.radwag.com.

2. ESTRUCTURA DE DATOS

2.1. Registros de entrada

Lista de la variables de entrada:

Variable	Offset	Longitud [WORD]	Tipo de datos
Masa	0	2	float
Tara	4	2	float
Unidad	8	1	word
Estado de la plataforma	10	1	word
Umbral Lo	12	2	float
Estado entradas	66	1	word
MIN	68	2	float
Max	72	2	float
Umbral de dosificación rápida	76	2	float
Umbral de dosificación lenta	80	2	float
Estado de calibración	100	1	word
Estado de proceso de dosificación	102	1	word

Masa de la plataforma – el valor de la masa de la carga se devuelve en la unidad actual.

Tara de plataforma – el valor de tara de la plataforma se devuelve en la unidad de calibración.

Unidad de la plataforma – determina la unidad de masa de la plataforma actual (visualizada).

Bit del registro	
0	- gramo [g]
1	- kilogramo [kg]
2	- libra [lb]
3	- uncia [oz]
4	- quilates [ct]
5	- Newton [N]

Ejemplo:

Valor de lectura HEX 0x02. Forma binaria:

B1/7	B1/6	B1/5	B1/4	B1/3	B1/2	B1/1	B1/0	B0/7	B0/6	B0/5	B0/4	B0/3	B0/2	B0/1	B0/0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

La unidad de peso es kilogramo [kg].

Estado de la plataforma – determina el estado de una plataforma de pesaje dada.

Bit del registro	
0	medición correcta (la balanza no informa un error)
1	medición estable
2	balanza está en cero
3	balanza está tarado
4	balanza está en el segundo rango
5	balanza está en el tercer rango
6	balanza informa un error NULL
7	balanza informa un error LH
8	balanza informa un error FULL
9	necesidad_de calibración_de tiempo / necesidad_de calibración_de temperatura

Ejemplo:

Valor de lectura HEX: 0x13

B1/7	B1/6	B1/5	B1/4	B1/3	B1/2	B1/1	B1/0	B0/7	B0/6	B0/5	B0/4	B0/3	B0/2	B0/1	B0/0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1

La balanza no informa un error, medición estable en el segundo rango.

Umbral LO - devuelve el valor umbral **LO** en la unidad de calibración de la plataforma dada.

Estado de entradas-máscara de bits de las entradas de la plataforma

Ejemplo:

Valor de lectura HEX: 0x02

B1/7	B1/6	B1/5	B1/4	B1/3	B1/2	B1/1	B1/0	B0/7	B0/6	B0/5	B0/4	B0/3	B0/2	B0/1	B0/0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Las entradas 1 y 2 de la plataforma están en estado alto.

MIN – devuelve el valor ajustado en umbral **MIN**(en la unidad actual).

MÁX – devuelve el valor ajustado en umbral **MÁX**(en la unidad actual).

Umbral de dosificación rápida – devuelve el valor del umbral de dosificación rápida configurado (gruesa)

Umbral de dosificación lenta – devuelve el valor del umbral de dosificación lenta configurado (precisa)

Estado de calibración

0x00 – calibración completada correctamente

0x01 – proceso activo

0x02 – rango excedido

0x03 – tiempo excedido

0x04 – proceso interrumpido

Estado del proceso de dosificación – determina el estado del proceso

0x00 – proceso inactivo

0x01 – tara en curso

0x02 – proceso en ejecución

0x03 – proceso detenido

0x05 – proceso completo

2.2. Registros de salida

Lista de la variables de entrada:

Variable	Offset	Longitud [WORD]	Tipo de datos
Comando	0	1	word
Comando con parámetro	2	1	word
Tara	6	2	float
Umbral Lo	10	2	float
Estado de salidas	14	1	word
MIN	16	2	float
Max	20	2	float
Umbral de dosificación rápida	24	2	dword
Umbral de dosificación lenta	28	1	word
Masa peso de calibración	48	1	word

Comando básico – establecer el bit de comando apropiado realiza la tarea directamente de acuerdo con la tabla:

Bit del registro	Comando
0	Puesta a cero de la plataforma
1	Tara la plataforma
5	Inicio de dosificación
6	Detener la dosificación
7	inicio de la calibración interna
8	Determinación de masa inicial
9	Determinación del factor de calibración
10	Registro de parámetros de calibración (masa inicial/coeficiente de calibración)

Ejemplo:

Guardar el registro con el valor 0x02

B1/7	B1/6	B1/5	B1/4	B1/3	B1/2	B1/1	B1/0	B0/7	B0/6	B0/5	B0/4	B0/3	B0/2	B0/1	B0/0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Esto tara la plataforma.

	<i>El comando se ejecuta una vez, después de detectar el ajuste de su bit. Si es necesario volver a ejecutar el comando con el mismo conjunto de bits, primero se debe borrar y luego volver a establecer el valor deseado.</i>
---	--

Comando complejo – establecer el bit de comando apropiado realiza la tarea directamente de acuerdo con la tabla:

Bit del registro	Comando
0	Ajustar el valor de tara para la plataforma dada
1	Establecer el valor umbral de LO para una plataforma dada
2	Ajuste del estado de la salida
3	Configuración el umbral MIN
4	Configuración el umbral MÁX
5	Establecer el umbral de dosificación rápida
6	Configuración el umbral de dosificación lenta
7	Configuración del valor del peso de calibración

	<i>El comando compuesto requiere la configuración de parámetros en la dirección de 2 a 24. la tabla Lista de los parámetros del comando compuesto.</i>
	<i>Un comando complejo se ejecuta una vez, después de que se detecta la configuración de un bit dado. Si es necesario volver a ejecutar el comando con el mismo conjunto de bits, primero se debe borrar y luego volver a establecer el valor deseado.</i>

Ejemplo:

Enviar a la balanza tara del valor de 1.0

La ejecución del comando requiere guardar 2 registros:

1. Comando complejo - valor 0x01 – es decir, establecer la tara.
2. Tara – valor 1.0 (0x3F800000).

Tara – parámetro de comando compuesto: valor de tara (en la unidad de calibración)

Umbral LO – parámetro de comando compuesto valor de umbral LO (en la unidad de calibración)

Estado de salidas – parámetro de comando compuesto: especificando el estado de las salidas de la plataforma.

Ejemplo:

Configuración de la salida de la plataforma 1 en un estado alto.

La máscara de las salidas será:

B1/7	B1/6	B1/5	B1/4	B1/3	B1/2	B1/1	B1/0	B0/7	B0/6	B0/5	B0/4	B0/3	B0/2	B0/1	B0/0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1

Después de convertir a HEX, obtenemos 0x01

La ejecución del comando requiere guardar 2 registros

1. Comando complejo - valor 0x02 - que establece el estado de las salidas.
2. Máscara de salida – valor 0x01.

Min – parámetro de comando compuesto: el valor del umbral MIN(en la unidad del modo de trabajo actual usado).

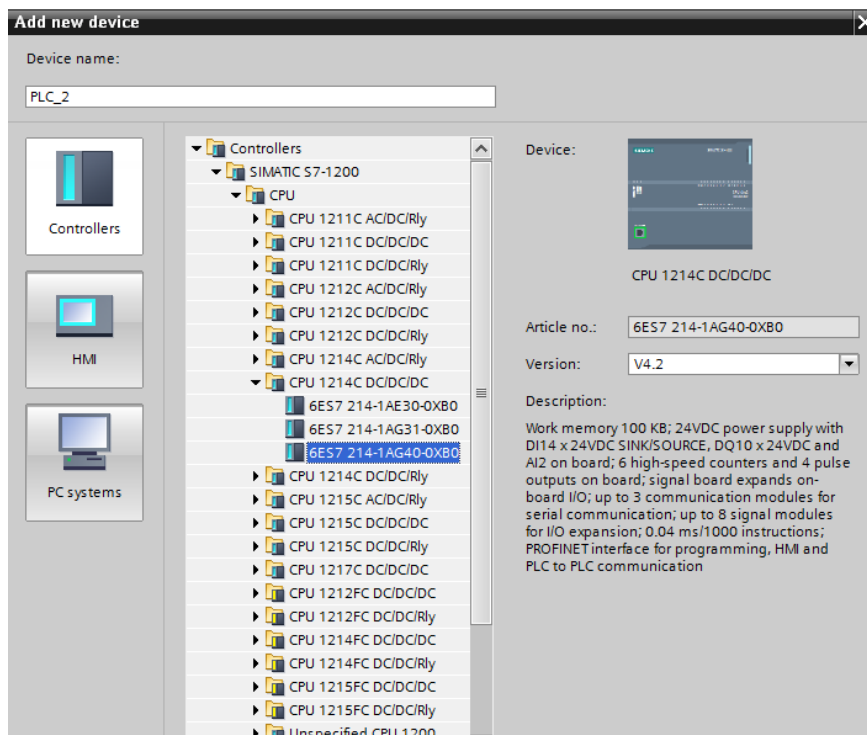
MÁX – parámetro de comando compuesto: el valor del umbral MÁX(en la unidad del modo de trabajo actual usado).

Umbral de dosificación rápida – parámetro de comando complejo - valor de umbral de dosificación rápido (grueso)

Umbral de dosificación lenta – parámetro de comando complejo - valor de umbral de dosificación lento (precisa)

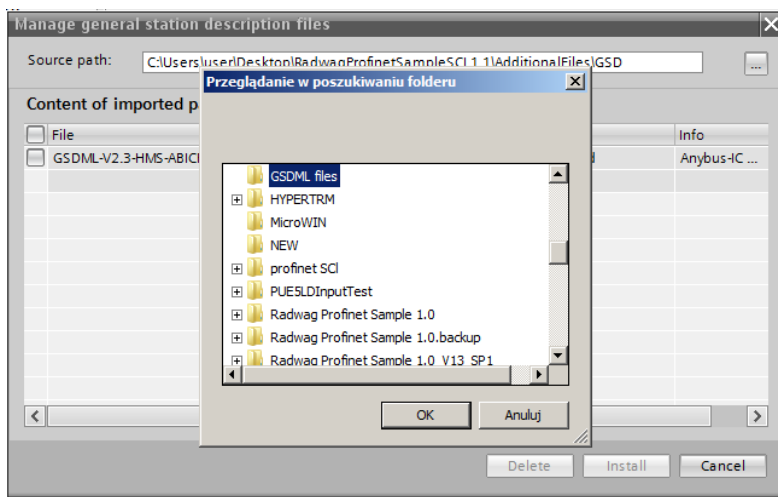
3. CONFIGURACIÓN DEL MÓDULO PROFINET EN EL ENTORNO TIA PORTAL V14

El trabajo en el entorno debe comenzar con la creación de un nuevo proyecto en el que se determinará la topología de la red PROFINET con el controlador MASTER, que en este ejemplo será el controlador de la serie SIEMENS S7-1200.

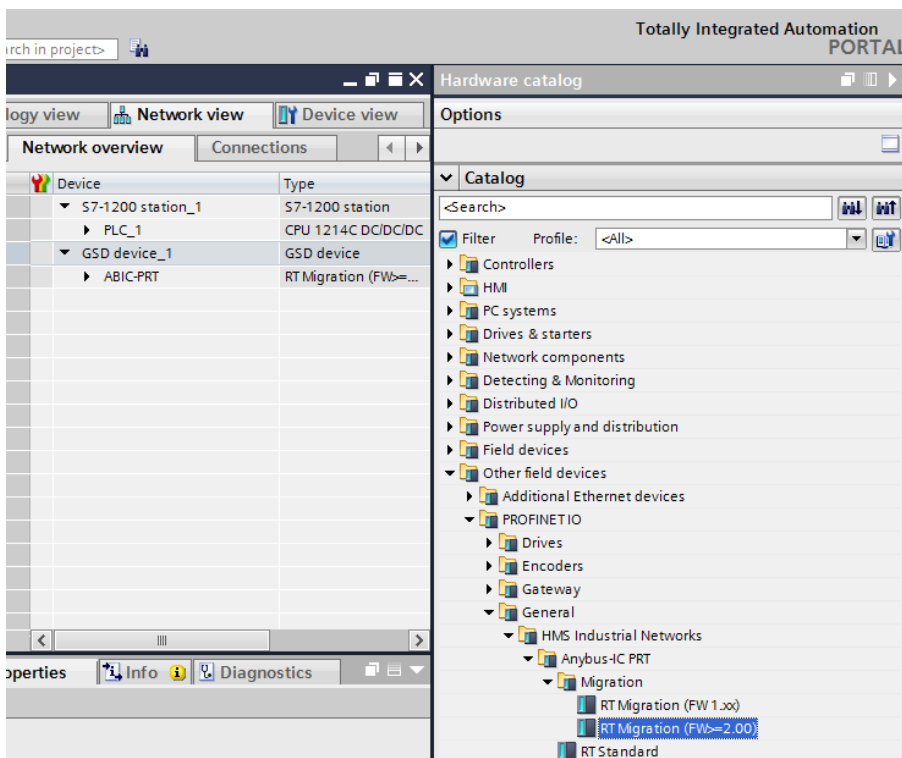


3.1. Importación GSD

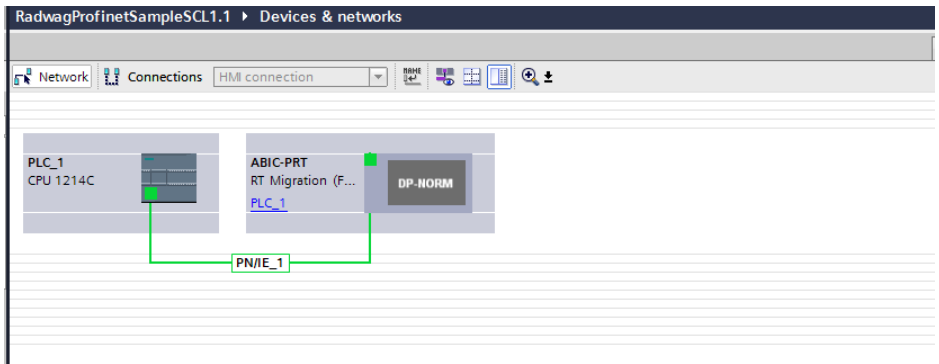
Usando el archivo de configuración GSD adjunto, se debe agregar un nuevo dispositivo al entorno. Para hacer esto, use la pestaña OPCIONES y luego GESTIONAR ARCHIVOS DE DESCRIPCIÓN DE LA ESTACIÓN GENERAL (GSD) e indique la ruta al archivo GSD.



Después de agregar con éxito el archivo en la lista de dispositivos, podemos encontrar el módulo ABIC-PRT que nos interesa:



Ya puede crear una red que consta de un controlador MASTER y un módulo SLAVE agregado:



3.2. Configuración del módulo

En esta etapa, debe construir una red compuesta por el controlador MASTER y los dispositivos SLAVE (balanza). Después de conectar la fuente de alimentación en el entorno, puede buscar dispositivos utilizando la función **DISPOSITIVOS ACCESIBLES**. Como resultado, deberíamos encontrar MASTER y SLAVE en la lista:

The screenshot shows the 'Accessible devices' window. At the top, a laptop icon is connected to a network of devices represented by a blue box. Below this, a table lists the accessible nodes of the selected interface:

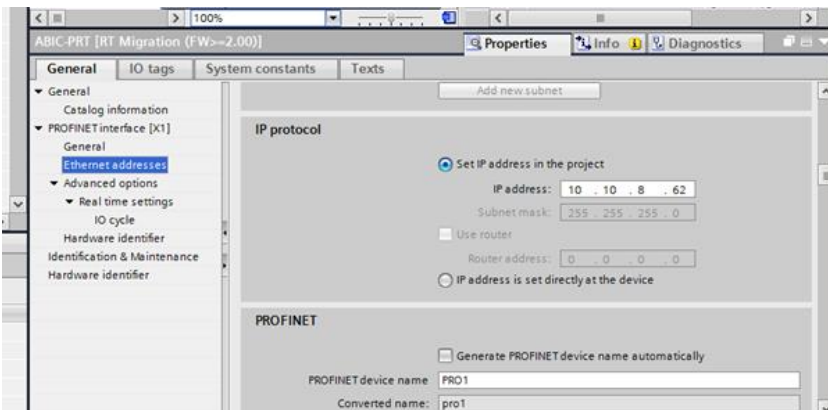
Device	Device type
Accessible device	S7-PC
pro2	RT Migration (F...)
plc_1	CPU 1214C D...

Below the table, there is a checkbox labeled 'Flash LED'.

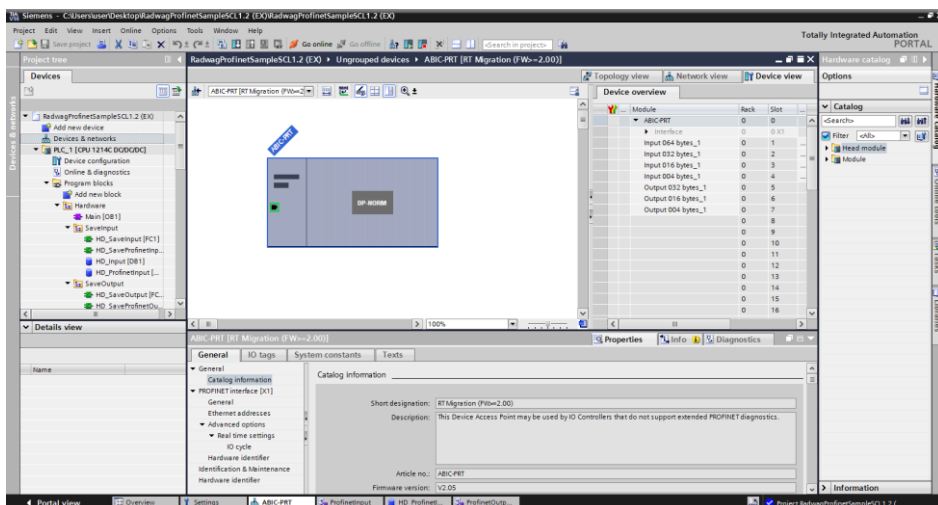
Online status information:

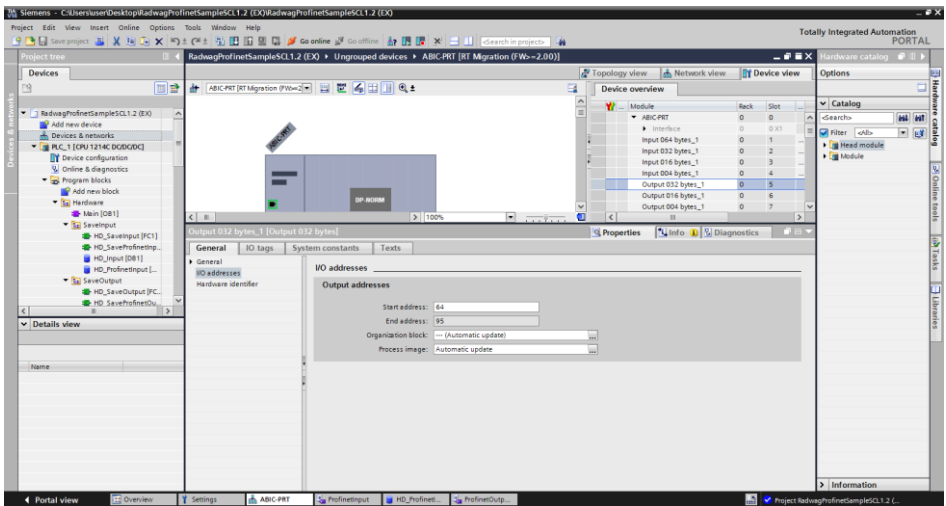
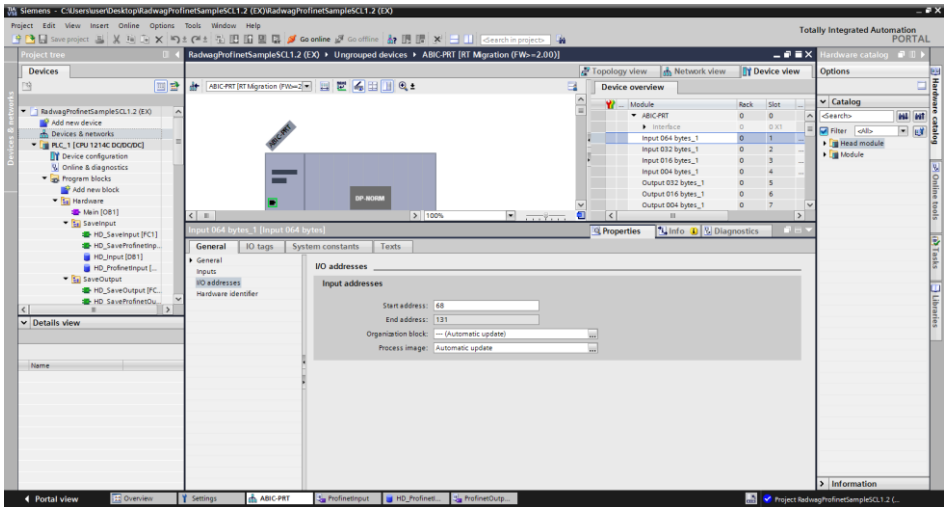
- Found accessible device Accessible device [00-16-76-25-13-51]
- Scan completed. 3 devices found.
- Retrieving device information...
- Scan and information retrieval completed.

A continuación, debe definir la dirección IP del módulo y su nombre en la red PROFINET. Después de seleccionar el módulo en la pestaña PROPIEDADES, busque el submenú PROFINET INTERFACE donde ingrese la dirección IP y dé un nombre. Estos ajustes deben ser compatibles con los parámetros establecidos en el menú de la balanza. Recuerde que la dirección IP SLAVE debe estar en la misma subred que la dirección MASTER.

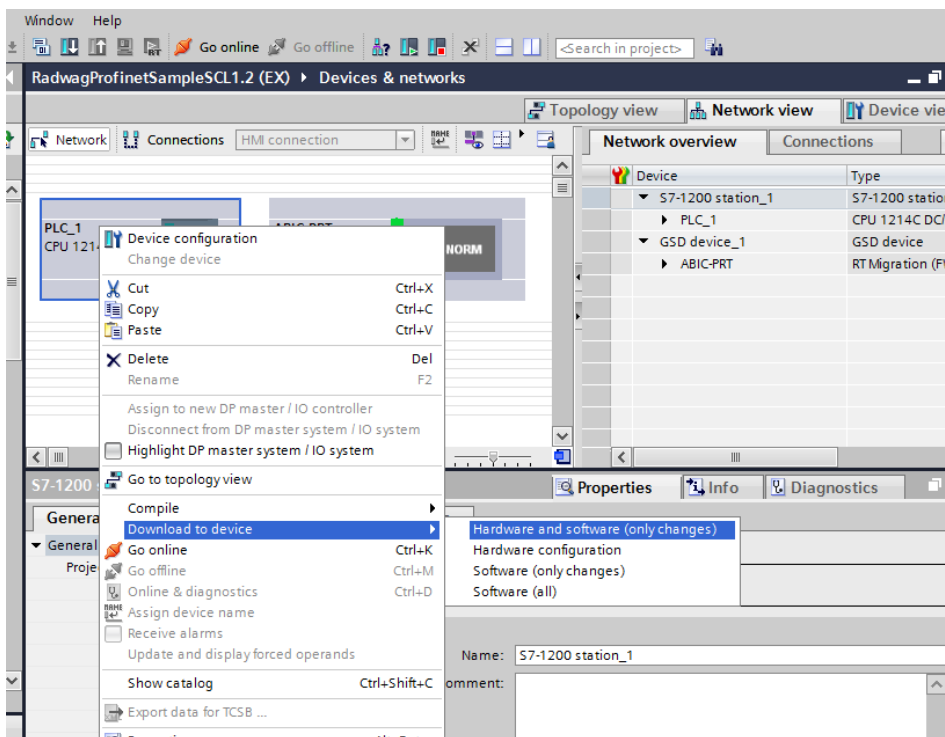


Podemos ir a la configuración del módulo. Al principio, definimos el tamaño de los registros de entrada y salida y definimos sus direcciones de inicio. Para ello, de la lista de módulos de ENTRADA y SALIDA disponibles, seleccione los que se muestran en la siguiente imagen. El tamaño máximo de los datos de entrada es 110 bytes y los datos de salida son 52 bytes. El proyecto utiliza las direcciones de inicio predeterminadas: 68 para el módulo ENTRADA y 64 para SALIDA:

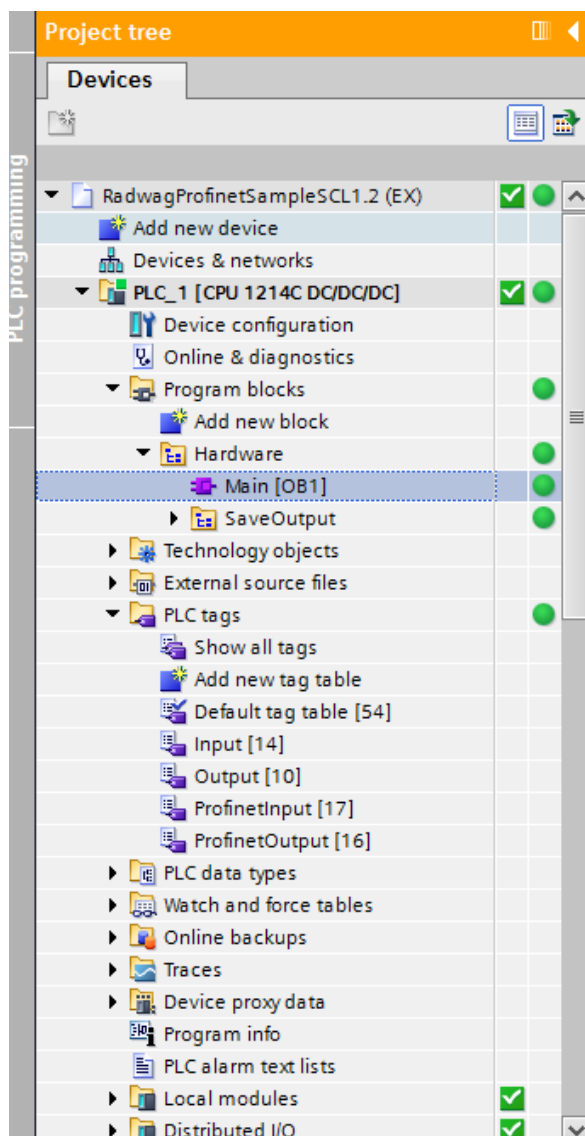




En esta etapa, puede cargar la configuración de hardware en el controlador y puede comenzar a cargar los datos en el controlador:



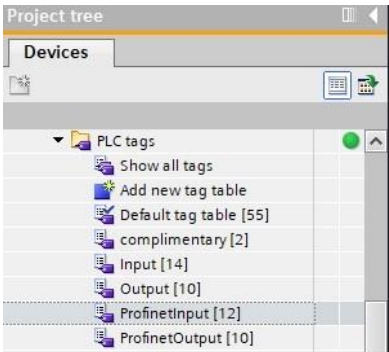
Después de compilar y cargar el código con éxito, MASTER y SLAVE deberían establecer una conexión. Esto se puede verificar yendo a la conexión ONLINE. Deberíamos obtener el resultado como se muestra a continuación.



El siguiente paso será crear el código del programa.

4. APLICACIÓN DE DIAGNÓSTICO

Es mejor comenzar a crear una aplicación definiendo los nombres de los registros simbólicos de entrada y salida. Para ello, utilizamos la rama PLC TAGS del árbol del proyecto. Para este ejemplo, las matrices de etiquetas se crearon como se muestra en la siguiente figura:

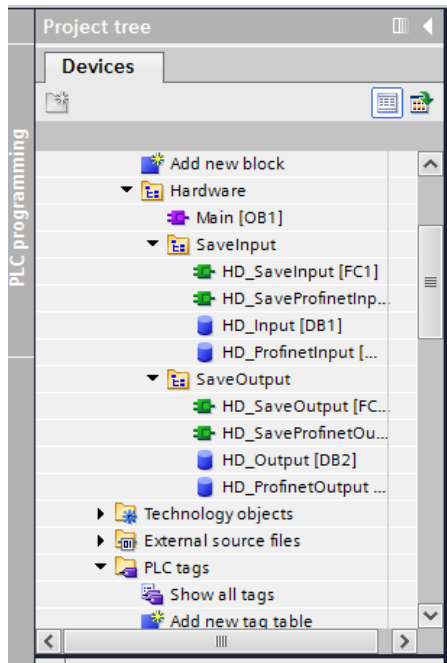


Las tablas ENTRADA y SALIDA se refieren a las entradas / salidas físicas del controlador MASTER y no tienen ningún significado en esta aplicación. Los registros de entrada y salida del módulo PROFINET se definen en las tablas ProfinetInput y ProfinetOutput. Las figuras siguientes muestran los nombres y direcciones simbólicos asignados:

RadwagProfinet#RP1.0 ang ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ PLC tags ▶ ProfinetInput [12]									
Tags User constants									
ProfinetInput									
	Name	Data type	Address	Retain	Access...	Write...	Visibl...	Monitor value	Comment
1	mass	Real	%ID68		☒	☒	☒	-125.3	
2	tare	Real	%ID72		☒	☒	☒	125.3	
3	unit	Word	%IW76		☒	☒	☒	16#0001	
4	status	Word	%IW78		☒	☒	☒	16#020F	
5	LO	Real	%ID80		☒	☒	☒	0.0	
6	inputs	Word	%IW134		☒	☒	☒	16#0000	
7	min	Real	%ID136		☒	☒	☒	0.0	
8	max	Real	%ID140		☒	☒	☒	0.0	
9	threshold_dose_coarse	Real	%ID144		☒	☒	☒	500.0	
10	threshold_dose_fine	Real	%ID148		☒	☒	☒	990.0	
11	calibr_stat	Word	%IW168		☒	☒	☒	16#0000	
12	dose_stat	Word	%IW170		☒	☒	☒	16#0000	
13	<Add new>				☒	☒	☒		

RadwagProfinet#RP1.0 ang ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ PLC tags ▶ ProfinetOutput [10]									
Tags User constants									
ProfinetOutput									
	Name	Data type	Address	Retain	Access...	Write...	Visibl...	Monitor value	Comment
1	cammand	Word	%QW64		☒	☒	☒	16#0000	
2	complex command	Word	%QW66		☒	☒	☒	16#0000	
3	set tare	DWord	%QD70		☒	☒	☒	16#0000_0001	
4	set lo	Real	%QD74		☒	☒	☒	4.0	
5	outputs	Word	%QW78		☒	☒	☒	16#0002	
6	set min	Real	%QD80		☒	☒	☒	200.0	
7	set max	Real	%QD84		☒	☒	☒	122.34	
8	set threshold_dose_coarse	Real	%QD88		☒	☒	☒	444.44	
9	set threshold_dose_fine	Real	%QD92		☒	☒	☒	888.88	
10	set_calibr_mass	Real	%QD112		☒	☒	☒	2000.0	
11	<Add new>				☒	☒	☒		

Para no trabajar directamente sobre las entradas / salidas físicas del módulo, conviene crear bloques de datos que contengan representaciones de estos registros y crear funciones "reescribiendo" los valores entre ellos. Para ello, cree el grupo **HARDWARE** en la rama **PROGRAMA BLOCKS** y defina los bloques de datos de la siguiente manera:



Los bloques **HD_OUTPUT** y **HD_INPUT** se refieren a entradas / salidas **MASTER** físicas y no son relevantes en términos de este proyecto. Los bloques **HD_ProfinetOutput** y **HD_ProfinetInput** se refieren a los registros de entrada / salida del módulo **PROFINET** en un instrumento de pesaje. Tienen el siguiente aspecto:

RadwagProfinetHRP1.0 ang ▶ PLC_1 [CPU 1214C DC/DC] ▶ Program blocks ▶ Hardware ▶ SaveInput ▶ HD_ProfinetInput [DB3]

Keep actual values Snapshot Copy snapshots to start values Load start values as actual values

	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	▼ Static								
2	mass	Real	0.0	☐	☒	☒	☒	☐	
3	tare	Real	0.0	☐	☒	☒	☒	☒	
4	unit	Word	16#0	☐	☒	☒	☒	☐	
5	status	Word	16#0	☐	☒	☒	☒	☐	
6	lo	Real	0.0	☐	☒	☒	☒	☐	
7	inputs	Word	16#0	☐	☒	☒	☒	☐	
8	min	Real	16#0	☐	☒	☒	☒	☐	
9	max	Real	0.0	☐	☒	☒	☒	☐	
10	threshold_dose_coarse	Real	0.0	☐	☒	☒	☒	☐	
11	threshold_dose_fine	Real	0.0	☐	☒	☒	☒	☐	
12	calibr_status	Word	16#0	☐	☒	☒	☒	☐	
13	dose_status	Word	16#0	☐	☒	☒	☒	☐	

RadwagProfinetHRP1.0 ang ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ Program blocks ▶ Hardware ▶ SaveOutput ▶ HD_ProfinetOutput [DB4]

Keep actual values Snapshot Copy snapshots to start values Load start values as actual values

	Name	Data type	Start value	Retain	Accessible f...	Write...	Visible in ...	Setpoint	Comment
1	Static								
2	command	Word	16#0000						
3	complex command	Word	16#0000						
4	set tare	Real	1.0						
5	set lo	Real	4.0						
6	outputs	Word	16#0002						
7	set min	Real	200.0						
8	set max	Real	122.34						
9	threshold_dose_coarse	Real	444.44						
10	threshold_dose_fine	Real	888.88						
11	calibr_mass	Real	2000.0						

Las funciones que reescriben valores entre las entradas / salidas físicas del módulo pueden tener este aspecto:

RadwagProfinetHRP1.0 ang ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ Program blocks ▶ Hardware ▶ SaveInput ▶ HD_SaveProfinetInput [FC3]

	Name	Data type	Default value	Comment
1	Input			
2	<Add new>			
3	Output			

```

IF... CASE... FOR... WHILE... (V...) REGION
OF... TO DO... DO...

1 "HD_ProfinetInput".mass := "mass";
2 "HD_ProfinetInput".tare := "tare";
3 "HD_ProfinetInput".unit := "unit";
4 "HD_ProfinetInput".status := "status";
5 "HD_ProfinetInput".lo := "LO";
6 "HD_ProfinetInput".inputs := "inputs";
7 "HD_ProfinetInput".min := "min";
8 "HD_ProfinetInput".max := "max";
9 "HD_ProfinetInput".threshold_dose_coarse := "threshold_dose_coarse";
10 "HD_ProfinetInput".threshold_dose_fine := "threshold_dose_fine";
11 "HD_ProfinetInput".calibr_status := "calibr_stat";
12 "HD_ProfinetInput".dose_status := "dose_stat";
13
14

```

RadwagProfinetHRP1.0 ang ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ Program blocks ▶ Hardware ▶ SaveOutput ▶ HD_SaveProfinetOutput [FC4]

	Name	Data type	Default value	Comment
1	Input			
2	<Add new>			
3	Output			

```

IF... CASE... FOR... WHILE... (V...) REGION
OF... TO DO... DO...

1 "command" := "HD_ProfinetOutput".command;
2 "complex command" := "HD_ProfinetOutput".complex command;
3 "set tare" := "HD_ProfinetOutput".set tare;
4 "set lo" := "HD_ProfinetOutput".set lo;
5 "outputs" := "HD_ProfinetOutput".outputs;
6 "set min" := "HD_ProfinetOutput".set min;
7 "set max" := "HD_ProfinetOutput".set max;
8 "set threshold_dose_coarse" := "HD_ProfinetOutput".threshold_dose_coarse;
9 "set threshold_dose_fine" := "HD_ProfinetOutput".threshold_dose_fine;
10 "set_calibr_mass" := "HD_ProfinetOutput".calibr_mass;
11
12
13

```

Invocar las funciones en el bucle principal del programa.

RadwagProfinetSampleSCL1.2 (EX) ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ Program blocks ▶ Hardware ▶ Main [OB1]

Main

	Name	Data type	Default value	Comment
1	Input			
2	Initial_Call	Bool		Initial call of this OB
3	Remanence	Bool		=True, if remanent data are available
4	Temp			
5	<Add new>			
6	Constant			

IF... CASE... FOR... WHILE...
OF... TO DO... DO... (*...*) REGION

```
1 "HD_SaveInput"();
2 "HD_SaveOutput"();
3 "HD_SaveProfinetInput"();
4 "HD_SaveProfinetOutput"();
5
```

	"HD_SaveInput"	%FC1
	"HD_SaveOutput"	%FC2
	"HD_SaveProfinetInput"	%FC3
	"HD_SaveProfinetOutput"	%FC4

Al compilar y cargar el programa en el dispositivo en el bloque de datos, puede leer registros de salida interesantes (MONITOR ALL) y guardar registros de salida (por ejemplo, cambiando el START VALUE i LOAD START VALUES AS ACTUAL) del modo SLAVE.

Una función ejemplar de guardar el registro con el valor 0x01(restablecimiento de la plataforma):

RadwagProfinetHRP1.0 ang ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ Program blocks ▶ Hardware ▶ Main [OB1]

Main

	Name	Data type	Default value	Comment
1	Input			
2	Initial_Call	Bool		Initial call of this OB
3	Remanence	Bool		=True, if remanent data are available

IF... CASE... FOR... WHILE...
OF... TO DO... DO... (*...*) REGION

Network 2:

Comment

```
1 "imp" := "Input0_0" AND "memo";
2 "memo" := NOT "Input0_0";
3
4 IF ("Input0_0" = true) THEN
5   "HD_ProfinetOutput".command := 16#0001;
6 ELSE
7   "HD_ProfinetOutput".command := 16#0000;
8
9 END_IF;
10
```

	"imp"	%M100.0
	"memo"	%M100.1
	"Input0_0"	%I0.0
	"HD_ProfinetOutput"	%DB4
	"HD_ProfinetOutput".com...	
	"HD_ProfinetOutput"	%DB4

