

Communication protocol:

PUE HX5.EX Weighing Indicator

SOFTWARE MANUAL

ITKP-02-04-04-23-EN



APRIL 2023

CONTENTS

1.	WEIGHING INDICATOR SETTINGS - CONFIGURATION	4
2.	DATA STRUCTURE	4
	2.1. Input Address	4
	2.2. Input Registers	5
	2.3. Output Address	7
	2.4. Output Registers	7
3.	CONFIGURATION OF PROFINET MODULE IN TIA PORTAL V14	11
	3.1. GSD Import	11
	3.2. Module Configuration	13
4.	PLC SOFTWARE SAMPLE	18

1. WEIGHING INDICATOR SETTINGS - CONFIGURATION

To set indicator communication via PROFINET protocol go to **SETUP / IM01.EX communication module / Additional modules / Anybus module>**. For detailed description of settings configuration read **"PUE HX5.EX -Software manual**" user manual.

2. DATA STRUCTURE

2.1. Input Address

Input variables list:

Variable	Offset	Length [WORD]	Data type
Mass	0	2	float
Tare	4	2	float
Unit	8	1	word
Platform status	10	1	word
LO threshold	12	2	float
Process status (Stop, Start)	16	1	word
Inputs status	66	1	word
Min	68	2	float
Max	72	2	float
Fast dosing threshold	76	2	float
Slow dosing threshold	80	2	float
Lot number	84	2	dword
Operator	88	1	word
Product	90	1	word
Customer	92	1	word
Packaging	94	1	word
Formulation	100	1	word
Dosing process	102	1	word

2.2. Input Registers

Platform mass – returns platform mass in current unit.

Platform tare – returns platform tare in an adjustment unit.

<u>Platform unit</u> – determines current mass unit of a given platform.

Unit bits						
0	gram [g]					
1	kilogram [kg]					
2	carat [ct]					
3	pound [lb]					
4	ounce [oz]					
5	Newton [N]					

Example:

Read HEX value: 0x02. Binary form:

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

The unit of the weighing instrument is kilogram [kg].

Platform status – determines state of a given weighing platform.

Status	Status bits								
0	Measurement correct (the weighing instrument does not report any error)								
1	Measurement stable								
2	Weighing instrument indicates zero								
3	Weighing instrument tared								
4	Weighing instrument in II weighing range								
5	Weighing instrument in III weighing range								
6	Weighing instrument reports NULL error								
7	Weighing instrument reports LH error								
8	Weighing instrument reports FULL error								

Example:

Read HEX value: 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

The weighing instrument does not report any error, measurement stable in II weighing range.

LO threshold – returns value of platform's LO threshold in an adjustment unit.

Process status – determines dosing or formulations process status:

0x00 – process disabled 0x01 – process stopped 0x02 – process aborted 0x03 – process finished

Input state – bitmask of indicator inputs. The first 4 least significant bits represent weighing terminal inputs.

Example:

Read HEX value: 0x000B

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	1	0	1	1

Inputs 1, 2 and 3 of the weighing indicator take HI state.

MIN - response: MIN threshold value (in current unit).

MAX - response: MAX threshold value (in current unit).

<u>Fast dosing threshold</u> - returns fast dosing threshold value in an adjustment unit.

<u>Slow dosing threshold</u> - returns slow dosing threshold value in an adjustment unit.

Lot number – response: lot number. Only numerical values are accepted! All other characters are skipped.

Operator - response: code of logged in operator.

Product – response: code of selected product.

Customer – response: code of selected customer.

Packaging - response: code of selected packaging.

Formulation – response: code of selected formulation.

Dosing process – response: code of selected dosing process.

2.3. Output Address

Input variables:

Variable	Offset	Length [WORD]	Data type
Command	0	1	word
Command with parameter	2	1	word
Platform	4	1	word
Tare	6	2	float
LO threshold	10	2	float
Output state	14	1	word
Min	16	2	float
Мах	20	2	float
Lot number	32	2	dword
Operator	36	1	word
Product	38	1	word
Customer	40	1	word
Packaging	42	1	word
Formulation	48	1	word
Dosing process	50	1	word

2.4. Output Registers

<u>Basic command</u> – writing the register with respective value triggers the following actions:

Bit No.	Operation
0	Zero the platform
1	Tare the platform
2	Delete statistics
3	Save/Print
4	Start

5	Breakdown (STOP without confirmation)
6	Tare/Zero the platform
7	Lock keypad
8	Unlock keypad



A command is executed once upon detecting that its bit has been set. If the command is to be executed more than once, it is necessary to zero the bit first, and reset it to the required value next.

Example:

Writing the register with value 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

This causes scale taring.

<u>**Complex command**</u> – setting a respective value results with performance of a given task, see the table:

Bit No.	Operation
0	Setting tare value for a given platform
1	Setting LO threshold value for a given platform
2	Setting lot number
3	Setting outputs status
4	Operator selection
5	Product selection
6	Packaging selection
7	Setting MIN threshold value
8	Customer selection
11	Dosing process selection
12	Setting MAX threshold value



Complex command requires setting a respective parameter (offset from 4 to 50 – refer to output registers table)



A command with a parameter is executed once upon detecting that its bit has been set. If the command is to be executed more than once, it is necessary to zero the bit first, and reset it to the required value next.

Example:

Sending tare of 1.0 value for the 1st platform.

Carrying out the command requires writing 3 registers:

offset 2 - command with parameter - value 0x01 - i.e. tare setting.

offset 4 – number of a weighing platform to which the tare is to be written - 0x01 value for the 1st first platform.

offset 6 – tare value in float format - 1.0.

<u>Platform</u> – complex command parameter: weighing platform number.

Tare – complex command parameter: tare value (in an adjustment unit).

LO threshold – complex command parameter: LO threshold value (in an adjustment unit).

<u>**Output state**</u> – complex command parameter: determines state of the weighing indicator and communication module outputs.

Example:

Setting high state to outputs 1 and 3 of the weighing indicator.

Output mask has the following format:

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	1	0	1

After conversion to HEX the result is 0x05

Carrying out the command requires writing 2 registers:

offset 2 - command with parameter - value 0x08 - i.e. record of outputs state.

offset 14 – outputs mask 0x05

As a result, outputs number 1 and 3 take high state.

<u>MIN</u> – complex command parameter: MIN threshold value (in a unit of current working mode).

<u>MAX</u> – complex command parameter: MAX threshold value (in a unit of current working mode).

<u>Lot number</u> – complex command parameter: lot number value. Only numerical values are accepted! All other characters are skipped.

Operator – complex command parameter: operator code (digits only).

<u>Product</u> – complex command parameter: product code (digits only).

<u>Customer</u> – complex command parameter: customer code (digits only).

Packaging – complex command parameter: packaging code (digits only)

Formulation – complex command parameter: formulation code (digits only).

Dosing process – complex command parameter: dosing process code (digits only).

3. CONFIGURATION OF PROFINET MODULE IN TIA PORTAL V14

Operating the environment has to be preceded with creating a new project in which the topology of the PROFINET network with MASTER PLC is determined (in this example: SIEMENS S7-1200).



3.1. GSD Import

Using the included GSD configuration file add new device to the environment. Use OPTIONS tab first, MANAGE GENERAL STATION DESCRIPTION FILES (GSD) next and indicate the path to GSD file.

Manage general station	description files	×
Source path: C:\Users	user/Deskton/RadwanProfinetSamnleSCI11\AdditionalFiles\GSD Przegladanie w poszukiwaniu folderu	
Content of imported p		
File GSDML-V2.3-HMS-ABICI	Info Anybus B Anybus An	-IC
<	OK Anuluj	>
	Delete Install Can	cel

Upon successful adding of the file using list of devices, find ABIC-PRT module:

ırch in project>		Totally Integrated Auto	matio POR	n TAL
	_ = = ×	Hardware catalog	70	
logy view 🔒 Network view	Device view	Options		
Network overview Connect	ions 🛛 🖣 🕨			
Y Device	Туре	✓ Catalog		
 \$7-1200 station_1 	S7-1200 station	<search></search>	itil	itit
PLC_1	CPU 1214C DC/DC/DC	Filter Profile: <all></all>		
 GSD device_1 	GSD device	Controller:		
ABIC-PRT	RT Migration (FW>=			
		PC systems		
		Drives & starters		
		Network components		
		Detecting & Monitoring		
		Distributed I/O		
		Power supply and distribution		
		Field devices		
		Additional Ethernet devices		
		▼ Image: PROFINETIO		
		Drives		
		Encoders		
		▶ 🛄 Gateway		
		✓ Im General		
<	>	HMS Industrial Networks		
perties	stics	✓ Im Anybus-IC PRT		
		✓ Im Migration		
		RI Migration (FW 1.xx)		
		RT Standard		

You can now create a network consisting of one MASTER PLC and added SLAVE module:

RadwagProfinetSampleSCL1.1 > Devices & networks
💦 Network 🔢 Connections HMI connection 💌 📅 🖫 💷 💷 🔍 生
PLC_1 ABIC-PRT
CPU 1214C RT Migration (F DP-NORM
PIN/E_1
1

3.2. Module Configuration

At this stage, create a network consisting of MASTER device and SLAVE device (weighing instrument). Upon connecting the power supply, search for device using ACCESSIBLE DEVICES function. The list should contain MASTER and SLAVE devices:

Accessible devices		Type of the PG/PC interface: PG/PC interface:	PN/IE	GBE Family Controll	► er ► € 3
	Accessible nodes of the	e selected interface:			
	Device	Device type	Interface type	Address	MAC address
	Accessible device	S7-PC	ISO	-	00-16-76-25-13-51
	pro2	RT Migration (FW 1.xx)	PN/IE	10.10.8.64	00-30-11-0D-EE-17
	plc_1	CPU 1214C DC/DC/DC	PN/IE	10.10.8.244	28-63-36-9C-D1-12
Flash LED					
Online status informatio	n:			Display only err	<u>S</u> tart search or messages
🔒 Found accessible d	evice Accessible device [0	00-16-76-25-13-51]			^
🚹 Scan completed. 3 (devices found.				
· Retrieving device in	formation				
Scan and informatio	on retrieval completed.				~
				<u></u>	now <u>C</u> ancel

Next, specify the IP address of the module and its name in PROFINET network. Upon selecting the module in PROPERTIES tab, find PROFINET INTERFACE and enter IP address and name. Those settings have to be the same as the ones set in the weighing instrument menu. IP SLAVE address has to be part of the same subnet as the MASTER address.

< E > 100%	
ABIC-PRT [RT Migration (FW>=2.00)]	Properties 🚺 Info 🔒 😨 Diagnostics 💷 🖃
General IO tags System constants Texts	
▼ General Catalog information	Add new subnet
PROFINET interface [X1] General <u>Ethemet addresses</u> · Advanced options v Real time settings IO cycle Indraware identifier Identification & Maintenance Hardware identifier	Set IP address in the project IP address: 10 , 10 , 8 , 62 Subnet mask: 255, 255, 255, 0 Use router Router address: 0 , 0 , 0 , 0 IP address is set directly at the device
PROFINET PROFINET device name	Generate PROFINET device name automatically

Proceed to module configuration. Start by determining the size and the starting address of input and output registers. To do this, select modules from the list of INPUT and OUTPUT modules as in the picture below. The maximum size of the input and output data is 116 bytes each. Default starting addresses were used in the project - 68 for INPUT module and 64 for OUTPUT module:

🖓 Siemens - C:IUsers\user\Desktop\RadwagProfi	netSampleSCL1.	2 (EX)\RadwagProf	inetSampleSCL1.2	(EX)								_ # ×
Project Edit View Insert Online Options	Tools Window	Help								To	tally integrated Aut	omation
🕒 🕒 🔄 Save project 🚢 🗶 ங 🕞 🗙 🍋	: (= ± 🔂 🖪	🖬 🖳 📮 💋 Go	online 🖉 Go offine	år 🖪 🖉 🛪	🚍 💷 🛛 🕹 earch in project> 📲					10	tany integrated Aut	PORTAL
Project tree 🛛 🕄 📢	RadwagProfine	etSampleSCL1.2 (E	X) • Ungrouped	devices + ABIC	PRT [RT Migration (FW>=2.00)]					_ # =>	Hardware catalog	• • • • •
Devices						21	Topology view	A Network view	D 10	evice view	Options	
e 🔤 🗎	ABIC-PRT [R	T Migration (FW)=2		🔟 🔍 ±		a 11	Device overvi	ew				🗆 Hər
**						^	. Nodula		Park	Clar	✓ Catalog	dw.
🗧 💌 🔄 RadwagProfinetSampleSCL1.2 (EX) 📃 🔨						-	- mooon	C-PRT	0	0	o Searchy	AND AND
Add new device								Interface	0	0 X1		
📩 Devices & networks		CT					Inc	ut 064 bytes 1	0	1	Filter Alls	. 🖬 🗧
PLC_1 [CPU 1214C DC/DC/DC]		V				10.5	Inp	ut 032 bytes 1	0	2	Head module	~
Device configuration						115	Inp	ut 016 bytes_1	0	3	Module	-
S Online & diagnostics						101	Inp	ut 004 bytes_1	0	4		8
 Trogram blocks 							Ou	tput 032 bytes_1	0	5		9
Add new block							Ov	tput 016 bytes_1	0	6		
 Hardware 						1	Ou	tput 004 bytes_1	0	7		5
Mein (061)						1			0	8		20
Saveinput									0	9		
HD_SaveInpot (PCT)									0	10		
ND Insue (081)									0	11		1
ND Profestional									0	12		5
The from the part of the									0	13		
HD SaveOutput IFC									0	14		
HD SaveProfinetOu.									0	15		5
< = >						× .			0	16 ,	× .	ĩ
✓ Details view	< =			> 1001	· · · · · · · · · · · · · · · · · · ·		<			>	-	ies
	ABIC-PRT [RT N	Migration (FW>-2.	.00)]				Properties	🚺 Info 👔 了 Dia	gnostics			
	General	IO tags Syste	em constants	Texts								
Name	 General 		Catalog informatio	n						-	<u>^</u>	
	Catalog into	ormation									-	
	· PROPINET inter	tace [X1]										
	General		Shi	ort designation: R	Migration (FVD=2.00)							
	T Advanced o	ioticos .		Description: Th	is Device Access Point may be used by	IO Contro	ollers that do not s	upport extended PROFI	NET diagno	istics.		
	· Paral time	a rettioor										
	10 cm	le le										
	Hardware in	dentifier										
	Identification 8	Maintenance										
	Hardware iden	tifer		Arbcie no.: A	SIC-PRI							_
			Fir	mware version: V	2.05						Information	_
Portal view Direction	Settings	ABIC-PRT	Sa Profinetinput	HD_Profinet	Sa ProfinetOutp					Y Project Rad	wagProfinetSampleSCL1.	

NadwagProf	inetSampleSCL1.2 (EX)\RadwagPr	ofinetSampleSCL1.2 (EX)					_ # X
Project Edit View Insert Online Options	Tools Window Help				Tota	ally Integrated Auto	mation
🕒 🕒 🗟 Save project 🚢 🐰 🖲 🕞 🗙 🍋	± (* ± 🗄 🗄 🖬 🖬 📮 💋 🛛	ia anline 🖉 Go offine 🛛 🛔 📗	📱 🗶 📑 🛄 🛹 Search in project>	- An		,	PORTAL
Project tree 🛛 🛛 🖣	RadwagProfinetSampleSCL1.2	(EX) → Ungrouped devices →	ABIC-PRT [RT Migration (FW>=2.00)		_ # = ×	Hardware catalog	
Devices				Topology view 🔥 Network view	Device view	Options	63
(H) (H)	ABIC-PRT (RT Migration (FV0=2)	- 🗉 📰 📈 🗄 🗍 🖲 ±		Device overview			
-					Pauls I fine	✓ Catalog	dw
RadwagProfinetSampleSCL1.2 (EX)	1			I ARIC PUT	0 0 0	Searchs	AND AND
Add new device				Interface	0 0 X1 =	C flux	
📩 Devices & networks	ALC:N			Input 064 bytes_1	0 1 _	Pilter Quis	
 PLC_1 [CPU 1214C DC/DC/DC] 	*		_	Input 032 bytes_1	0 2	Head module	
Device congulation				 Input 016 bytes_1 	0 3	P	10.1
The Program blacks				Input 004 bytes_1	0 4		8
Add new block				Output 032 bytes_1	0 5		n in
- Ta Hardware	-	DP-NORM		Output 016 bytes_1	0 6		0.0
Main [081]			1000	Output bok bytes_1	0 / 0		00
 SaveInput 			100.0			1	5
HD_SaveInput [FC1]	Input oos bytes_1 [input oos i	pytesj		Sinto S Diag	nostics		
HD_SaveProfinetinp	General IO tags Sys	tem constants Texts					8
HD_Input [D61]	General Inputs	I/O addresses					isks
SaveOutput HD_SaveOutput [FC	NO addresses Hardware identifier	Input addresses					U.
HD SaveProfinetOu		Start addres	ss: 68				6
< 2	-	End addres	ss: 131				an
Details view		Oceanin tion bloc	ek: (Automatic undate)				i i i i i i i i i i i i i i i i i i i
		Orgeniacioni ero	ce				
		increase manage	Per Controlline of Conte				
Name							
							_
							_
							_
						> Information	
Portal view Dverview	Settings 📩 ABIC-PRT	🛬 ProfinetInput 🧧 HD_Pro	fineti 🛬 ProfinetOutp		📑 😪 Project Radw	agProfinetSampleSCL1.2	(



On this stage you can download hardware and software configuration to the device and download data to the device.

	Window H	ielp 3 I	🔲 🖸 🖉 Co oplino 🖑	Co offling							
Ť	RadwagP	rof	inetSampleSCL1.2 (EX)	Devices & netwo	orks		ren in proje			-	
						📲 Topol	logy view	📥 Netw	ork view	Device	vie
ł	Networ	k	Connections HMI con	nection 💌 🕅	Ľ 🖫 🖽	• ⊒_ [Network	c overview	Conne	tions	[
						^	🛛 🍟 Dev	ice		Туре	
		_					-	S7-1200 statio	n_1	S7-1200 sta	tio
	DIC 1				_			PLC_1		CPU 1214C	DCI
	CPU 1214	Ľĭ	Device configuration		NORM		•	GSD device_1		GSD device	
			Change device		monam	-		ABIC-PRT		RT Migration	n (F\
≡		Х	Cut	Ctrl+X		· ·					
		Ē	Сору	Ctrl+C							
		Û	Paste	Ctrl+V		- H					
		×	Delete	Del		_					
			Rename	F2		_					
			Assign to new DP master /	IO controller		_					
			Disconnect from DP maste	r system / IO system							
	<		Highlight DP master syster	n / IO system		- 📋	<]	
	\$7-1200	đ	Go to topology view			Pi	operties	i Info	% Diagr	ostics	
	Comm		Compile	•				1			
	Genera		Download to device	•	Hardw	/are and soft	ware (only (changes)			
	 General 	ø	Go online	Ctrl+K	Hardw	/are configur	ation				
	Proje		Go offline	Ctrl+M	Softwa	are (only cha	nges)				
		Ų,	Online & diagnostics	Ctrl+D	Softwa	are (all)					
			Assign device name								
			Receive alarms								
			Update and display forced	operands	Name:	\$7-1200 st	ation_1				
~			Show catalog	Ctrl+Shift+C	omment:						^
-			Export data for TCSB								
		10	Proportion	Alt - Entor							

Upon successful compilation and loading of the code, MASTER and SLAVE modules should establish communication. You can check this by pressing GO ONLINE field. The result has to be similar to the result presented below.



The next step will be to create program code.

4. PLC SOFTWARE SAMPLE

Start creating the application by determining symbolic names of input and output registers. For this purpose, use the branch of the project tree: PLC TAGS. For the purpose of this example, the figure below contains exemplary tags tables:



INPUT and OUTPUT tables refer to the physical inputs/outputs of the MASTER device and are not relevant in terms of this application. The PROFINET module input and output registers are specified in ProfinetInput and ProfinetOutput tables.

RadwagProfinetSampleSCL1.2 ang(EX) → PLC_1 [CPU 1214C DC/DC/DC] → PLC tags → ProfinetInput [17]											
# :	🛫 🛫 📴 😤 🛍										
P	ProfinetInput										
		Name	Data type	Address	Retain	Acces	Writa	Visibl	Comment		
1		mass	Real 🔳	%ID68 💌							
2	-00	tare	Real	%ID72							
3	-00	unit	Word	%IW76			~	\checkmark			
4		status	Word	%IW78			~	\checkmark			
5	-	LO	Real	%ID80							
6	-	process status	Word	%IW84			~				
7		inputs	Word	%IW134							
8		min	Real	%ID136							
9	-00	max	Real	%ID140							
10	-00	lot number	DWord	%ID152							
11		operator	Word	%IW156							
12	-	article	Word	%IW158							
13	-00	customer	Word	%IW160							
14		packaging	Word	%IW162							
15		source warehuse	Word	%IW164							
16	-00	target warehouse	Word	%IW166							
17		formulation/dosing process	Word	%IW168							
18		<add new=""></add>				V	~	\checkmark			

The pictures below present determined symbolic names and addresses:

RadwagProfinetSampleSCL1.2 ang(EX) + PLC_1 [CPU 1214C DC/DC/DC] + PLC tags + ProfinetOutput [16]

# ;	2 2 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1										
Pr	ProfinetOutput										
	Na	ime	Data type	Address	Retain	Acces	Writa	Visibl	Comment		
1		cammand	Word 🔳	%QW64 💌		~					
2	-00	complex command	Word	%QW66							
З		platform	Word	%QW68		~					
4		set tare	Real	%QD70		\checkmark					
5		set lo	Real	%QD74							
6		outputs	Word	%QW78							
7		set min	Real	%QD80		~					
8		set max	Real	%QD84							
9		set lot	DWord	%QD96							
10		set operator	Word	%QW100		\checkmark					
11		set article	Word	%QW102							
12		set customer	Word	%QW104		\checkmark					
13		set source warehouse	Word	%QW108							
14		set target warehouse	Word	%QW110		\checkmark	\checkmark				
15		set formulation/dosing	Word	%QW112							
16		set packaging	Word	%QW106							
17		<add new=""></add>				V	 Image: A start of the start of	V			

In order to avoid working directly on module physical inputs/outputs, create data blocks containing the representations of those registers and create function 'rewriting' the values between them. Create HARDWARE group in PROGRAM BLOCKS branch and determine data blocks in the same way as presented below:



HD_OUTPUT and HD_INPUT blocks refer to physical MASTER inputs/outputs and are not relevant in terms of this project. HD_ProfinetOutput and HD_ProfinetInput blocks refer to the PROFINET module input/output registers on a weighing instrument. They look as follows:

RadwagProfinetSampleSCL1.2 ang(EX) > PLC_1 [CPU 1214C DC/DC/DC] > Program blocks > Hardware > SaveInput > HD_ProfinetInput [DB3]

🛫 👻 🔩 😹 📴 😤 Keep actual values 🔒 Snapshot 🧌 & Copysnapshots to start values 😹 🐼 Load start values as actual values 🖏 🖏

		Name		Data type	Start value	Retain	Accessible f	Writa	Visible in	Setpoint	Comment
1	-	▼ St	atic								
2	-	•	mass	Real 🔳	0.0						
З		•	tare	Real	0.0						
4		•	unit	Word	16#0						
5	-	•	status	Word	16#0						
6	-	•	lo	Real	0.0						
7	-	•	process status	Word	16#0						
8		•	inputs	Word	16#0						
9		•	min	Real	0.0						
10		•	max	Real	0.0						
11	-	•	lot number	DWord	16#0						
12	-	•	operator	Word	16#0						
13	-	•	article	Word	16#0						
14	-	•	customer	Word	16#0						
15		•	packaging	Word	16#0						
16		•	source warehouse	Word	16#0						
17	-	•	target warehouse	Word	16#0						
18	-	•	formulation/dosing pr	Word	16#0						

RadwagProfinetSampleSCL1.2 ang(EX) > 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 🔹 🖳											
HD_ProfinetOutput											
		Na	me	Data type	Start value	Retain	Accessible f	Writa	Visible in	Setpoint	Comment
1	-	•	Static								
2	-	•	command	Word 🔳	16#02						
З	-	•	complex command	Word	16#0008						
4	-	•	platform	Word	16#0001						
5	-	•	set tare	Real	2.0						
6	-	•	set lo	Real	1.5						
7	-	•	outputs	Word	16#0000					\sim	
8	-	•	set min	Real	2.1						
9	-	•	set max	Real	2.2						
10	-	•	set lot number	DWord	16#0000						
11	-	•	set operator	Word	16#0004						
12	-	•	set article	Word	16#0001						
13	-	•	set customer	Word	16#01						
14	-00	•	set packaging	Word	16#0004						
15	-	•	set source warehouse	Word	16#0						
16		•	set target warehouse	Word	16#0						
17	-	•	set formulation/dosin	Word	16#0						

The functions that rewrite values between physical inputs/outputs of the module may look like this:

wagProfinetSampleSCL1.2 an	g(EX) > PLC_1 [CPU 1214C DC/DC/DC] → Program blocks → Hardware → SaveOutput →	HD_SaveProfinetOutput [FC4]	_ # # X
# # ₽ ■ E 2 2 ± B HD SaveProfinetOutput	≱ ୯° ६₀ ८≣ २≣ २३ ६= र⊒ र⊒	₩ '= `= ⊯ % & @ % % % @		
Name	Data type Default value	Comment		
1 🕣 💌 Input				^
2 Add new>				
3 🕣 🔻 Output				~
	<pre>CHE_NOMEL (*.*) REGON</pre>	<pre>command; Dumput": "complex command"; platform; "set tare"; lo?; Dutputs; set min"; "set max"; set lot number"; put"."set clustomer"; put"."set clustomer"; put"."set clustomer"; funetOutput"."set target warehouse"; funetOutput".set formulation/dosing process";</pre>		
<			> 100%	
Padua @ProfinatSampleSCI 1.2 a			HD SaveProfinationut [EC2]	a = V
nadwaghormetsampleStL1.2 a		Sej - Hogran blocks - Hardware - Savemput -	nb_saverionneunput[rCs]	
P P P L E 2 2 ± 1 HD SaveProfinetInput	¢° 6₀ 68 18 19 64 54 54 54	# 노 노 🕅 위 선 🖉 🕾 🕮 🔒		

HD_SaveProfinetInput										
Name		Data type	Default value	Comment						
1 🕣 🔻 Input						~				
2 Add new>										
3 🕣 🔻 Output			-			~				
						-				
-⊒ == ₩	IF OF	FOR WHILE (**)	REGION							
	o					_				
	1 "HI	ProfinetInput	".mass := "mass"	;		^				
	2 "HI	ProfinetInput	<pre>iput".tare := "tare";</pre>							
	3 "HI	Profinetinput	".unit := "unit"							
	4 "HI	Profinetinput	".status := "sta	tus";						
	2 "UT	a" - "process status".								
8 "HD DrofinerTourt" min - "min".										
9 "HD ProfinetInput" max := "max":										
10 "HD_ProfinetIngut"."lot number" := "lot number"; 11 "HD_ProfinetIngut".operator := "operator"; 12 "HD ProfinetIngut".sticle := "article";										
						=				
 13 "HD ProfinetInput".customer := "customer"; 										
14 "HD_ProfinetInput".packaging := "packaging";										
15 "HD_ProfinetInput"."source warehouse" := "source warehuse";										
16 "HD_ProfinetInput"."target warehouse" := "target warehouse";										
	;									
	18									
	19									
	20									
	21									
	22					Ľ.,				
	23									
	24									

Invoke the functions in the main program loop.

RadwagProfinetSampleSCL1.2 (EX) → PLC_1 [CPU 1214C DC/DC/DC] → Program blocks → Hardware → Main [OB1]									
(2) 2) 20 10 10 10 10 10 10 10 10 10 10 10 10 10									
Main									
Name	Data type	Default value	Comment						
1 📶 🔻 Input									
2 🕣 🗉 Initial_Call	l Bool		Initial call of this OB						
3 🕣 = Remanend	ce Bool		=True, if remanent data ar	e available					
4 4 राष्ट्र Тетр 									
5 < <add new:<="" th=""><th>></th><th></th><th></th><th></th><th></th><th></th></add>	>								
6 🤕 🔻 Constant									
IF CASE FOR WHILE (*									
1 "HD_SaveIr	nput"();			"HD SaveInput"		%FC1			
2 "HD_SaveOu	<pre>2 "HD_SaveOutput"();</pre>				"HD_SaveOutput"				
3 "HD_SavePi	rofinetInput"();	"HD_SaveProfinetInput"		%FC3					
4 "HD_SavePi 5	rofinetOutput"();		"HD_SaveProfinetOu	tput"	%FC4				

Upon compiling and loading the program to the device in the data block you can read interesting output registers (MONITOR ALL) and save output registers (e.g. by changing START VALUE and LOAD START VALUES AS ACTUAL) of the SLAVE mode.



