

BALLUFF

Software-Description

BAE PD SmartVision Controller

Sample Program S7-1200/1500



CONTENT

1	SVC SAMPLE PROGRAM	3
1.1	Sample Functions	3
1.2	General Data	3
1.3	Description	3
1.4	PLC program overview	4
1.5	Assignment process data modules und camera instances	5
1.6	Camera system settings.....	5
1.7	Configuration of result container and PLC-receive data block.....	6
1.8	Control and watch options with the variable table.....	7
1.8.1	Function block input parameter	7
1.8.2	Function block output parameter	8
1.8.3	Sequene diagram of sample inspection with Smart Camera Function Block	9
1.8.4	Sequene diagram switch Application with Smart Camera Function Block.....	10
1.9	Disclaimer of Liability sample program.....	11

1 SVC SAMPLE PROGRAM

The demo program **SVC_Sample** allows a communication between a Balluff Smart Vision Controller BAE-PD-VS-* and a Simatic® S7-1200/1500 PLC.

1.1 Sample Functions

The following commandos are supported in the demo program:

Switch Application	Switches the application
Get Application ID	Gets the ID of the currently active application
Get Results	Gets the result container
Send Data	Sets the input data of the application
Get Date Time	Get the time stamp
Set Sequence Number	Sets the sequence number

Please, examine which of the given functions are supported by the SmartVisionController! The maximum read/write length of the FB is 32.767 byte.

1.2 General Data

Program name:	SVC_Sample
Invoked blocks:	FB10, FB50
Reserved memory bits:	MB0 Clock Memory, MB1 System Memory
Reserved Timers:	none
Reserved Counters:	none
Configured I/O Range	64 Byte
Invoke:	absolute
Device compatibility:	Siemens Simatic® S7-1200 CPU1214C Siemens Simatic® S7-1500 CPU1513F
Software version:	TIA-Portal V13 SP1

1.3 Description

The Simatic® S7-project „ **SVC_Sample** “ contains in FB10 an example invoke of the BVS_SC function block FB50. The used PLC type is a Simatic® S7-1200 CPU1214C and a S7-1500 CPU 1513F. The same program is used for both controllers. Configured I/O length 64 byte, peripheral I/O start address 256. The FB50 parameters are set suitable to the hardware configuration. The FB is initialized automatically by the program. The memory bit „DB_BVS_SC“.Init is set in the OB100 at PLC startup. For controlling the example, the variable table „VAT_BVS-SC_IBN“ is available.

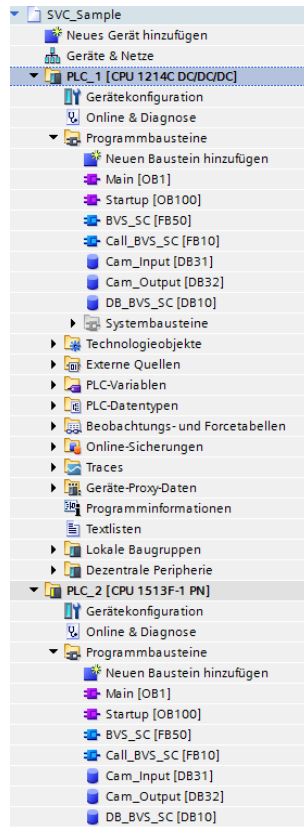
If the memory bit "DB_BVS_SC".Cam_Start_Run is set, the running mode starts in the camera and an inspection is executed. The trigger input is used for this example. The inspection starts if a positive pulse is detected at the FB Input "Cam_Trigger". If the result is ready, the memory bit "DB_BVS_SC". Result_Ready is set. With bit "DB_BVS_SC". Get_Results the result container can be transmitted to the PLC.

If the results are not read from the PLC the camera overwrites the results and the bit "DB_BVS_SC".WarnOutBufferErr is set.

Sample Program description for SmartVision Controller

1 SVC SAMPLE PROGRAM

1.4 PLC program overview

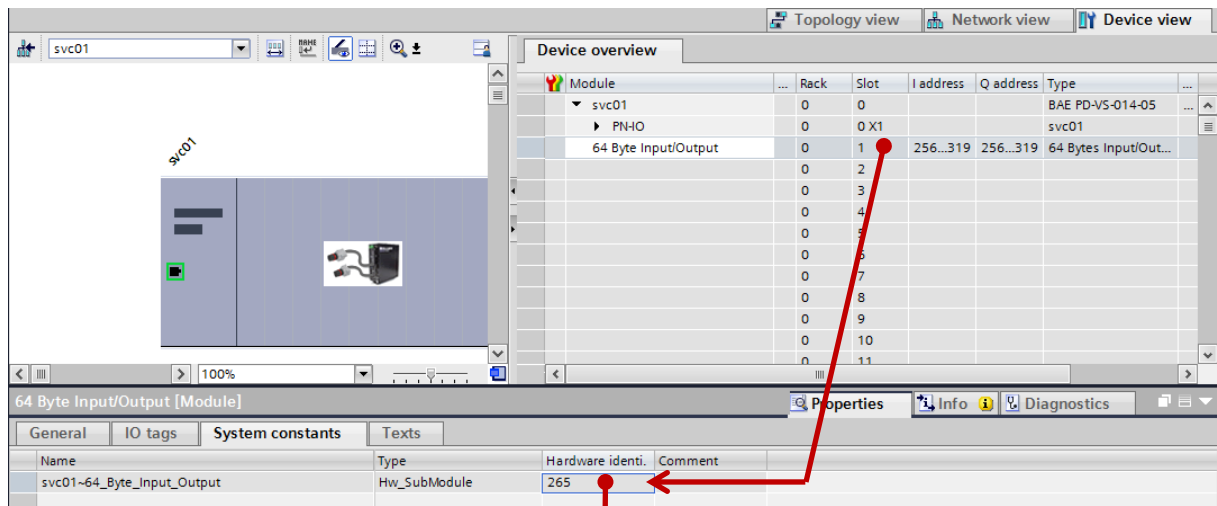


(Figure TIA Portal, Sample project)

Sample Program description for SmartVision Controller

1 SVC SAMPLE PROGRAM

1.5 Assignment process data modules und camera instances



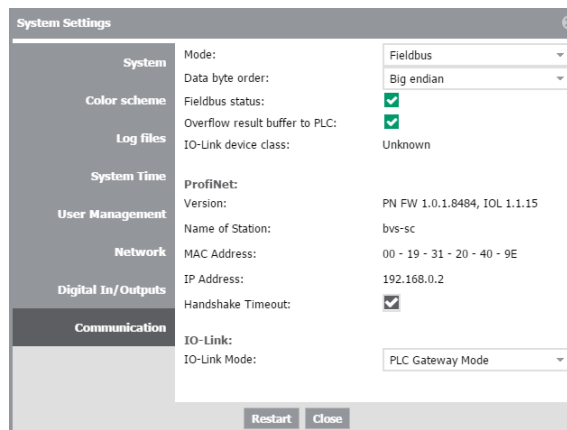
Slot	HW-Identifier (for example)	IO Adress (for example)	Camera instance
1	265	256...271	http://<SVC name>:9000

(Figure. TIA-Portal: Devices & networks, properties of SmartVisionController Input/Output module)

Each slot in the device configuration is firmly assigned to a camera instance. The function block gets access to the process data by the hardware-identifier of the slot for each camera instance. For each slot is a separate instance call of the function block necessary. The hardware-identifier is assigned by TIA Portal software. The hardware-identifier is displayed in the module properties of each slot. For the SmartVision Cpontroller svc01_1, connected to S7-1200, the HW-Identifier 277 is used.

1.6 Camera system settings

The "Fieldbus" Mode has to be activated in the system settings of the camera. The byte order "Big Endian" and the IO-Link mode "PLC Gateway" is set.



(Figure BVS Cockpit, system settings) The PLC settings are made according to the camera settings.

Sample Program description for SmartVision Controller

1 SVC SAMPLE PROGRAM

1.7 Configuration of result container and PLC-receive data block

The SmartCamera send result settings and the PLC receive data block using same data types with the same length. The folder sample „Sample.zip“ contains the SmartCamera inspection.

The screenshot shows the 'Configuration' tab of the BVS Cockpit. Below the 'Send results' button, there is a 'Parameter' section. A checkbox 'Use for inspection processing:' is checked. Below this, there is a table with four rows of parameters:

Parameter	Value	Variable
int16	2	Check_brightness.Brightness_va
string	256	Code_lesen.Included_text
real32	4	Code_lesen._Position_in_X_dir
bool	1	Code_lesen.Tool_processing

(Figure BVS Cockpit, result data)

The screenshot shows the TIA Portal interface for the 'Cam_Output [DB32]' data block. The table below represents the structure of the data block:

	Name	Data type	Offset	Start value
1	Static			
2	n.u.	Byte	0.0	16#0
3	AppID	Byte	1.0	16#0
4	Brightness	Int	2.0	0
5	ReadBarcode	Array[0..255] of Char	4.0	
6	CodePositionX	Real	260.0	0.0
7	BarcodeLocated	Byte	264.0	16#0

(Figure TIA Portal, DB32Cam_Output, SmartCamera output data)

The FB input parameter Offset_DBReceive is set to 1, therefor word variables from SmartCamera are sent to even byte addresses in the PLC. In the first byte, the camera transfers the AppID of the Inspection. Data types >1 starting in the Simatic® PLC at an even byte address. For processing of camera variables from data type string the datatype „Array[..] of Char“ must be used in the data block. String variables in the Simatic® PLC containing an additional length information. The „Array of Char“ can be formatted in data type String.

1 SVC SAMPLE PROGRAM

1.8 Control and watch options with the variable table

1.8.1 Function block input parameter

- "DB_BVS_SC". Init – function block is initialized
- "DB_BVS_SC". CommandStart – starts the commando
- "DB_BVS_SC". Cam_ResetWarn – resets the SmartCamera
- "DB_BVS_SC". Cam_Stop_Run – SmartCamera is stopped
- "DB_BVS_SC". Cam_Start_Run – SmartCamera is started
- "DB_BVS_SC". Cam_Trigger – SmartCamera is triggered
- "DB_BVS_SC". Get_Results – gets the results from SmartCamera
- "DB_BVS_SC". Cam_Default – set the FB input values to default
- "DB_BVS_SC". Command – FB commando input
- "DB_BVS_SC". Offset_DBSend – data offset in send data block
- "DB_BVS_SC". Offset_DBReceive – data offset in receive data block
- "DB_BVS_SC". Data_Length – amount of data, that is send to camera

...14C DC/DC] Watch and force tables VAT_BVS-SC_IBN

	Name	Address	Display format	Monitor ...	Modif...
1	// FB input parameter bit				
2	"DB_BVS_SC".Init		Bool		FALSE
3	"DB_BVS_SC".CommandStart		Bool		FALSE
4	"DB_BVS_SC".Cam_ResetWarn		Bool		FALSE
5	"DB_BVS_SC".Cam_Stop_Run		Bool		FALSE
6	"DB_BVS_SC".Cam_Start_Run		Bool		FALSE
7	"DB_BVS_SC".Cam_Trigger		Bool		FALSE
8	"DB_BVS_SC".Get_Results		Bool		FALSE
9	"DB_BVS_SC".Cam_Default		Bool		FALSE
10	// FB input parameter int				
11	"DB_BVS_SC".Offset_DBSend		DEC+/-		0
12	"DB_BVS_SC".Offset_DBReceive		DEC+/-		1
13	"DB_BVS_SC".Command		DEC+/-		3
14	"DB_BVS_SC".Data_Length		DEC+/-		1
15	// FB output parameter bit				
16	"DB_BVS_SC".FB_Ready		Bool		FALSE
17	"DB_BVS_SC".Result_Ready		Bool		FALSE
18	"DB_BVS_SC".Cam_WaitForTri...		Bool		
19	"DB_BVS_SC".Cam_Busy		Bool		
20	"DB_BVS_SC".Cam_Running		Bool		
21	"DB_BVS_SC".Cam_Ready		Bool		
22	"DB_BVS_SC".WarnOvertrigge...		Bool		
23	"DB_BVS_SC".WarnOutBufferErr		Bool		FALSE
24	"DB_BVS_SC".WarnInBufferErr		Bool		FALSE
25	"DB_BVS_SC".WarnOverheat		Bool		
26	"DB_BVS_SC".WarnSystemError		Bool		
27	"DB_BVS_SC".FB_Error		Bool		
28	"DB_BVS_SC".ErrorNumber		Hex		
29					
30	// Camera Inputcontainer				
31	"Cam_Input".SendByte[0]	%D831.D8B0	Hex		16#01
32	"Cam_Input".SendByte[1]	%D831.D8B1	Hex		16#02
33	"Cam_Input".SendByte[2]	%D831.D8B2	Hex		16#03
34	"Cam_Input".SendByte[3]	%D831.D8B3	Hex		16#04
35					
36	// Camera Outputcontainer				
37	"Cam_Output".n.u."	%D832.D8B0	Hex		
38	"Cam_Output".AppID	%D832.D8B1	Hex		16#00
39	"Cam_Output".Brightness	%D832.D8B2	DEC+/-		0
40	"Cam_Output".CodePositionX	%D832.D8D2...	Floating-point ...		
41	"Cam_Output".BarcodeLocated	%D832.D8B264	Hex		16#00
42	"Cam_Output".ReadBarcode[0]	%D832.D8B4	Character		
43	"Cam_Output".ReadBarcode[1]	%D832.D8B5	Character		
44	"Cam_Output".ReadBarcode[2]	%D832.D8B6	Character		
45	"Cam_Output".ReadBarcode[3]	%D832.D8B7	Character		
46	"Cam_Output".ReadBarcode[4]	%D832.D8B8	Character		

(Figure TIA Portal, watch table)

1 SVC SAMPLE PROGRAM

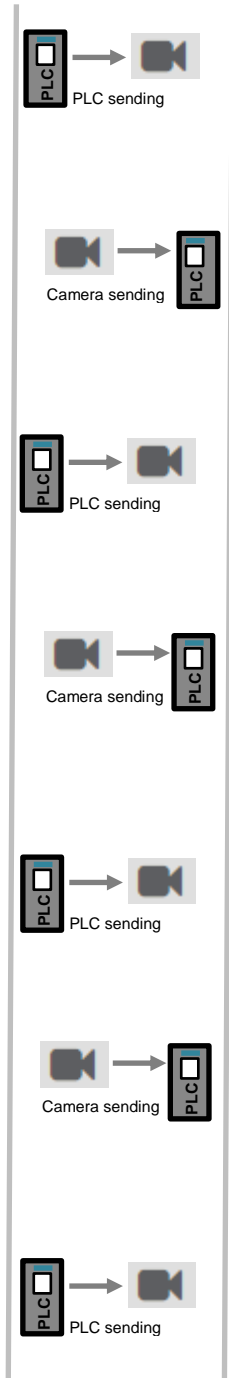
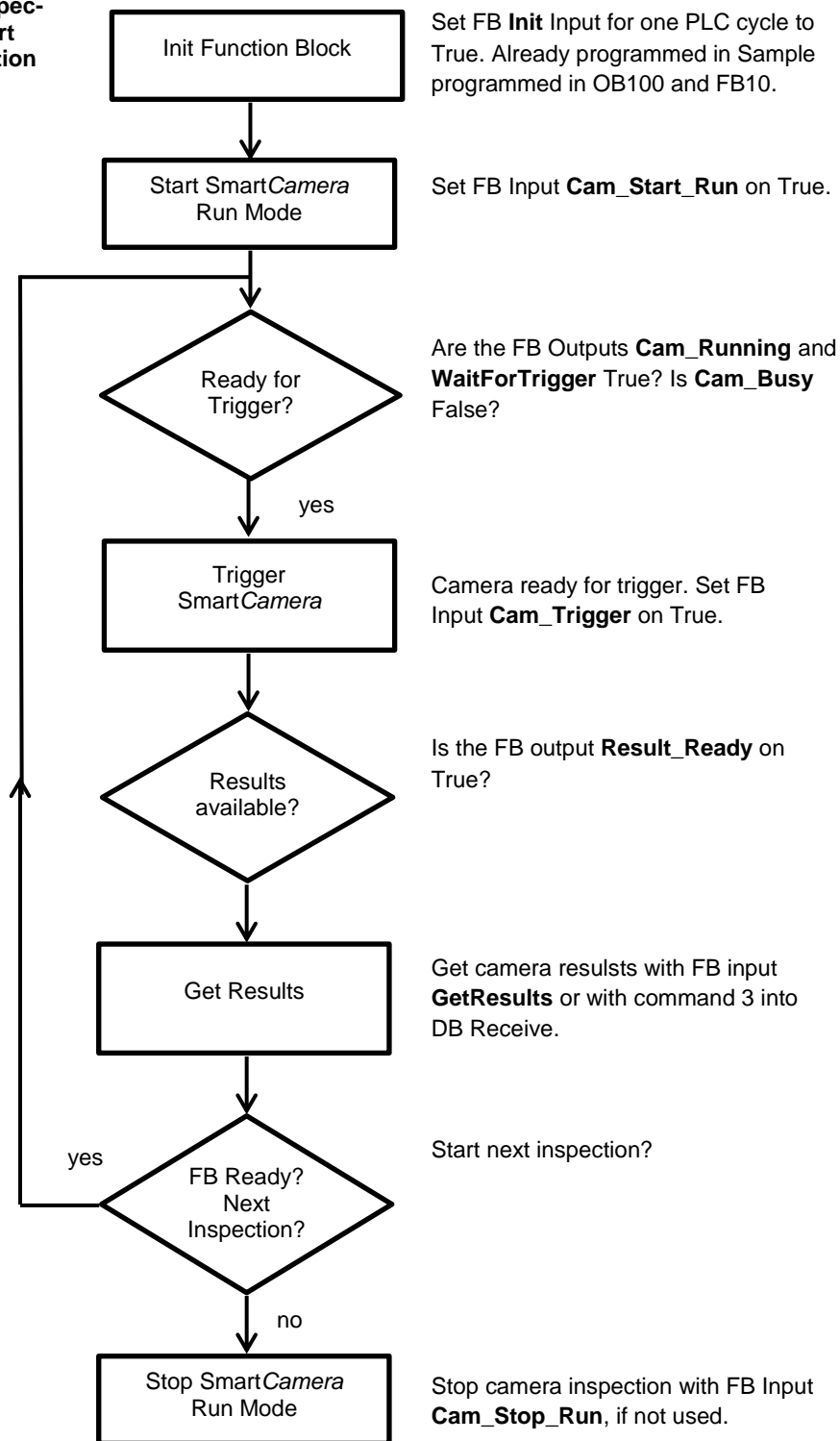
1.8.2 Function block output parameter

- "DB_BVS_SC".FB_Ready – job done
- "DB_BVS_SC".Result_Ready – inspection is done, data can be received from PLC
- "DB_BVS_SC".Cam_WaitForTrig – SmartCamera is ready for a trigger impulse
- "DB_BVS_SC".Cam_Busy – SmartCamera is busy at the moment
- "DB_BVS_SC".Cam_Running – SmartCamera is in running mode
- "DB_BVS_SC".Cam_Ready – SmartCamera is ready
- "DB_BVS_SC".WarnOvertriggerd – trigger was discarded
- "DB_BVS_SC".WarnOutBufferErr – output container overwritten
- "DB_BVS_SC".WarnInBufferErr - – input container error
- "DB_BVS_SC".WarnOverheat – temperature over limit
- "DB_BVS_SC".WarnSystemError - SmartCamera system errorr
- "DB_BVS_SC".FB_Error – job completed with error
- "DB_BVS_SC".ErrorNumber – FB or SmartCamera error number

Sample Program description for SmartVision Controller

1 SVC SAMPLE PROGRAM

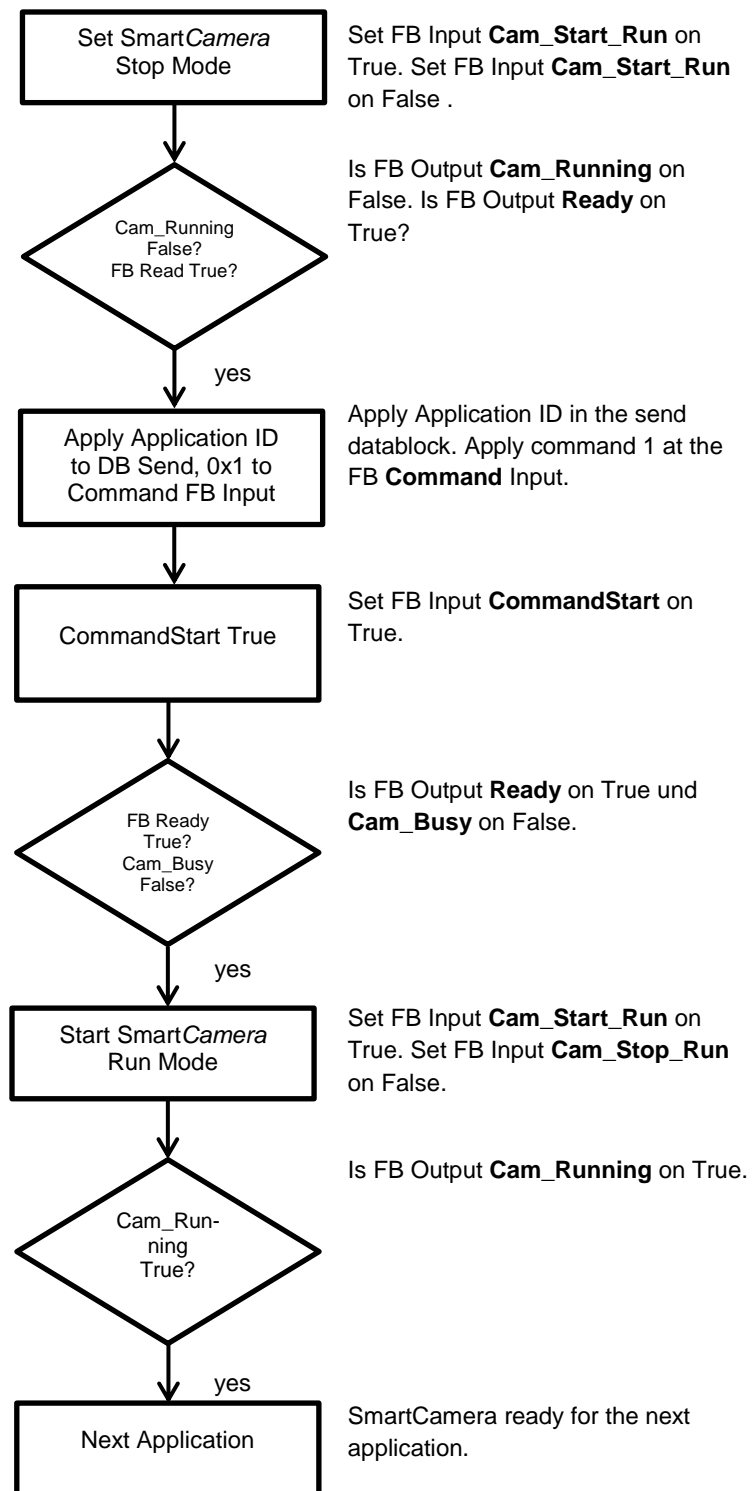
1.8.3 Sequence diagram of sample inspection with Smart Camera Function Block



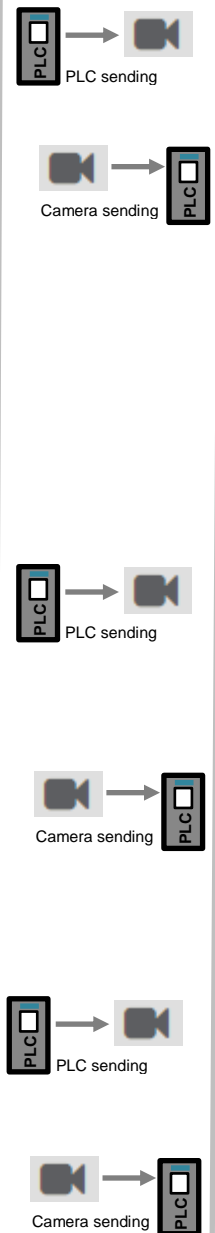
FB = SmartCamera Function Block

1 SVC SAMPLE PROGRAM

1.8.4 Sequence diagram switch Application with Smart Camera Function Block



FB = SmartCamera Funktionblock



1 SVC SAMPLE PROGRAM

1.9 Disclaimer of Liability sample program

This demo program is free of charge and is a universal application example. This demo program shall help program and configure PLC applications and shall provide possible solutions.

The user is not entitled to claim for warranty, error correction and updates. In particular there is excluded any claims against Balluff GmbH for damages that might result from the use of this demo program. Excluded from this limitation of liability shall be (a) those damages that are based on injury to life, limb or health, (b) a liability according to the Produkthaftungsgesetz (German Product Liability Law) and (c) cases of willful intent.

Please check if the demo program is intended for your application before adapting it in plants and machineries.

By using the S7 sample, made available free of charge you accept the limitation of warranty and liability!

Balluff GmbH
Schurwaldstraße 9
73765 Neuhausen a.d.F.
Germany
Tel. +49 7158 173-0
Fax +49 7158 5010
balluff@balluff.de
www.balluff.com

Valid from function block version 1.2 • D18; Subject to change.