Application Note



SBSI Vision Sensor integration with S7-1500 PLC USING TIA PORTAL 15

This Application Note helps the user to integrate SBSI Vision Sensor with Siemens S7-1500 PLC using TIA Portal 15. Also this application note briefly describes the Function Blocks which are used for controlling the functionality of SBSI Vision Sensor. Type(s)

Title	SBSI Vision Sensor Integration with S7-1500 PLC Using TIA Portal 15
Version	
Document no	
Original	en
Author	Festo
Last saved	

Copyright Notice

This documentation is the intellectual property of Festo AG & Co. KG, which also has the exclusive copyright. Any modification of the content, duplication or reprinting of this documentation as well as distribution to third parties can only be made with the express consent of Festo AG & Co. KG.

Festo AG & Co KG reserves the right to make modifications to this document in whole or in part. All brand and product names are trademarks or registered trademarks of their respective owners.

Legal Notice

Hardware, software, operating systems and drivers may only be used for the applications described and only in conjunction with components recommended by Festo AG & Co. KG.

Festo AG & Co. KG does not accept any liability for damages arising from the use of any incorrect or incomplete information contained in this documentation or any information missing therefrom.

Defects resulting from the improper handling of devices and modules are excluded from the warranty.

The data and information specified in this document should not be used for the implementation of safety functions relating to the protection of personnel and machinery.

No liability is accepted for claims for damages arising from a failure or functional defect. In other respects, the regulations with regard to liability from the terms and conditions of delivery, payment and use of software of Festo AG & Co. KG, which can be found at www.festo.com and can be supplied on request, shall apply.

All data contained in this document do not represent guaranteed specifications, particularly with regard to functionality, condition or quality, in the legal sense.

The information in this document serves only as basic information for the implementation of a specific, hypothetical application and is in no way intended as a substitute for the operating instructions of the respective manufacturers and the design and testing of the respective application by the user.

The operating instructions for Festo products can be found at www.festo.com.

Users of this document (application note) must verify that all functions described here also work correctly in the application. By reading this document and adhering to the specifications contained therein, users are also solely responsible for their own application.

Table of contents

1	Components/Software used	. 5
2	APPLICATION DESCRIPTION	. 6
2.1	Topology	. 6
3	Configuration of SBS Vision Sensor via Vision Sensor Configuration Studio for the use with PROFINET.	. 7
3.1	Settings in Vision Sensor Device Manager.	. 7
3.2	Setting of IP address and Name of SBS Vision Sensor	. 7
3.3	Open Vision Sensor Configuration Studio.	. 8
3.4	Select interface "PROFINET"	. 8
3.5	Definition of the telegram	. 9
4	S7-1500 PLC CONFIGURATION IN TIA PORTAL	10
4.1	Creating a new project in TIA Portal	10
4.2	Detecting the actual hardware configuration of the PLC connected in the network	14
4.3	Configuration of the IP parameters of the profinet interface of the PLC.	16
5	Adding GSDML File of SBS Vision sensor to TIA Portal.	17
5.1	GSDML file location in the installation folder	17
5.2	Adding the GSD File to TIA Portal	17
6	Configuration of SBS Vision Sensor in TIA Portal.	20
6.1	Adding the installed SBS Vision Sensor to Network View	20
6.2	Network Configuration of Profinet Interface SBSI Vision Sensor in TIA Portal.	22
6.3	Assigning Profinet name to SBSI Vision Sensor	24
6.4	Adding Data Module to SBSI	26
6.5	Identifying Hardware ID of the Sub modules of SBSI Vision Sensor.	28
7	Linking VisionSensor library to the project	30
8	DESCRIPTION OF SBSIVISION-Siemens Library FUNCTION BLOCKS	33
8.1	FB_CheckSBSI	33
9	INTEGRATION OF FUNCTION BLOCKS INTO PROJECT	36
9.1	FB_CheckSBSI	36
10	FUNCTION BLOCK EXECUTION WITH AN EXAMPLE	38
10.1	Example configuration in Vision Configuration Studio.	38
10.2	Example description in TIA Portal	43
10.3	Payload Data Mapping between TIA Portal and Vision Configuration Studio.	47
10.4	Payload Data conversion in TIA Portal 10.4.1 Converting floating values of payload data into a single Double Word	49 49

Table of contents

1 Components/Software used

Type/Name	Version Software/Firmware
GSDML File for SBSI Vision Sensor	V1_17_129
Siemens TIA Portal	V 15.1
Vision Sensor Configuration Studio	V 123.2.2
SBS Sensor Type description	

Table 1.1:Components/Software used

2 APPLICATION DESCRIPTION

This document explain how you will integrate the host function blocks of SBS Vision Sensor into SIEMENS TIA Portal V15.

The supported systems are:

- S71500
- S71200

Supported Field Bus :

• Profinet IO

The application note has the description for the following:

- SBS vision sensor configuration in vision sensor configuration studio.
- Configuring the needed payload data of the vision sensor in vision configuration studio.
- SBS Vision sensor Setup in SIEMENS TIA Portal.
- Installing the GSDML File for SBS Vision Sensor.
- Adding the SBS Vision sensor to Devices and Networks in TIA Portal.
- Description of the function blocks of SBS Vision sensor.
- Integrating the Function Blocks within the programming environment of TIA Portal.

2.1 Topology

When we want to communicate between S7-1500/1200 PLC and SBS Vision sensor the following topology must be followed.



→

- NOTE
- The IP address of SBS Vision Sensor, Ethernet Port of S7-1500/1200 and the PC used for programming the PLC must be in the same IP range.

3 Configuration of SBS Vision Sensor via Vision Sensor Configuration Studio for the use with PROFINET.

3.1 Settings in Vision Sensor Device Manager.

- Open Vision Sensor Configuration Studio .
- The following screen will be displayed in Vision Sensor device manager.

se	ensors											
	IP address	Sensor name	Hardv	vare Type	Varia Fi	irmwar M	Iode Manufact	turer Mac add	ress S	ubnet mask	Gateway	DH
	192.168.4.5	SBSI	R3B	Universal	Adv 1.	.23.2.2 R	un Festo	00-0E-F0	01-28-F5 2	55.255.255.0		Disa
	IP address	Sensor name	Hardware	Type Va	aria Firmwa	r Mod(M	anufacturer	Mac address	Subnet mas	k Gateway	DHCP	
		CDCT	R3B	Universal Ad	v 1.23.2.2	Run Fe	sto	00-0E-F0-01-28-F5	255.255.255.	0	Disabled	

• If it is not listed the sensor which is connected in the network can be found out by clicking the button "Find" as shown below.

Add active sensor			Favorites
IP address 🔽 🖌 🖌 Add			Options 🗢
Find	Config	View	Set
Click to find sensors in network			

3.2 Setting of IP address and Name of SBS Vision Sensor.

• Select the desired sensor and Click on **SET** the to get the network settings tab as shown below.





• The network settings tab will be as shown below.

Vision	Sensor Device M ? $ imes$
IPAddress	192.168.4 .2
Mask	24 255.255.255.000
Gateway	192.168.4 .5
DHCP	
Name	SBSI
	Get Cancel



NOTE

These settings are not active before a reboot of the sensor.

Important Conditions for properly working PROFINET communication

- 1. The SBS name must be identical in PLC and sensor.
- 2. The IP address of SBS and PLC must correspond (must be in same address range).
- **3.** IP address and name can be set in different ways:
 - Via SBS software (Vision Sensor Device Manager)
 - **Via PLC interface** (TIA portal). Refer **Chapter 6** to get detailed description on how to set SBS IP address and device name from TIA portal.

3.3 Open Vision Sensor Configuration Studio.

- Select the desired SBS Vision sensor. Click on Config.
- The Following dialog box appears.

×
inue?

- Confirm the dialog box by clicking **OK** to stop the Vision Sensor Device Manager and to start the configuration in Vision Sensor Configuration Studio.
- The following display can be seen.



NOTE

• In this chapter only settings needed for Profinet communication is explained. The step by step procedure to configure the vision sensor with an example is given in Chapter 10.

3.4 Select interface "PROFINET".

• Go to **Output** tab as shown below.

	Vision Sensor Configuration	n Studio - Universal	
	File View Options Help		
		🖌 🚺 🛄 🖉 💲	
	Setup	1.1	12 12
	Alignment	. <u>897.9</u>	
Select	Detector	QR - Hello World	QR - FESTO SE
output 🛑	Output		
tab	Result		
	Start sensor		

• Go to Interface >>> PROFINET as shown below to select Profinet interface.

		Selec	ct Interfa	ces						
										Configure output
			•							
	I/O) mapping Digital output	Interfaces	Timing	Telegram	Image transmission	Archiving			
			1		j	J				
		Name	Setting 1		Setting 2	Setting	3	Logical outputs	Enable	
	1	Internal I/O	PNP		\$				~	
	2	Serial	RS422		\$ 19200 Bd	🗢 🗍 🕏 🗍 🕏	÷ (0		
	4	Ethernet	(IN)2006	* *	(Out)2005		(0		
	5	EtherNet/IP					(<u>0</u>		
	6	PROFINET							✓	Select Profinet
	7	Vision Sensor Visualisation Stu	dio (Image an	d overlay	1 🜩 🛛					
	8	SBSWebViewer								
T L										

3.5 Definition of the telegram.

• Go to **Output >>>> Telegram** to define the data which has to be transferred via PROFINET interface.

		Select telegra	am			
						Configure output
		•				
I/O mapping	Digital output Interfaces	Timing Telegram	Image transmission	Archiving		
Binary 🗘	Start	C				
	Trailer					
	Separator)	
	End of Telegram				ANSI	•
Save to file	Selected fields		Data length		Status	
Reset	Detector result		Digital outputs		Logical outputs	
	Execution time		Active job no.		Checksum	

• For use with PROFINET interface the telegrams should be defined with **BINARY** format.

	Select telegram	
		Configure output
Calast	I/O mapping Digital output Interfaces Timing Telegram Image transmission Archiving	
BINARY	Binary 🗘 Start	
Iomat	Separator	
	End of Telegram	\$
	Save to file Selected fields Data length Status	
	Reset Detector result Digital outputs Logical outputs	
	Execution time Active job no. Checksum	

4 S7-1500 PLC CONFIGURATION IN TIA PORTAL

4.1 Creating a new project in TIA Portal

- 1. Start the TIA Portal V15.1 software.
- 2. Double click on Create New Project to create a new project.



3. Enter the Project name and select the path where the project must be saved in your system. Then click on **Create** to create the project.

	Enter the Project name
Create new project	
Project name:	SBSIVisionsensor
Path:	C:IPRAMODIPROJECTS\YJKP Servo Press\OPCUA Client
Version:	V15.1
Author:	inOpnss
Comment:	
	Click to create project
Comment:	Click to create project

Application Note-SBSI Vision Sensor- Integration of Host Function Blocks in SIEMENS TIA Portal V15 Seite 10 von 54

4. Double Click on **Configure a device** to configure the PLC needed.



5. Double Click on **Add New Device** to add a new PLC to the project.



6. Click on the PLC needed . In the below example S7-1500 is selected. All the available CPU's under S7-1500 will be displayed.



7. Click on Unspecified CPU S7-1500 and select the PLC below it and click on Add Device.

NOTE

If we select Unspecified CPU S7-1500, then the actual hardware configurations of the PLC can be read by using the detect hardware configurations option.

This saves the time needed to do the hardware configurations from the hardware catalog.



8. The project view will be as shown below.



4.2 Detecting the actual hardware configuration of the PLC connected in the network

1. Click on **Detect the configuration of the connected PLC** option to retrieve the PLC Configuration.



2. Once the hardware detection option is selected the following will be displayed.

-		PG/PC in	terface: Intel(R) Ethernet Connect	tion (4) I219-LM 💌 🐑
	Compatible acce	essible nodes of the selec	:ted interface:		
	Device	Device type	Interface type	Address	MAC address
				Oliakt	a access the available
				Click t	PLC's
Flash LED					↓
nline status informa	tion:			📃 Display	only error messages

3. Click **Start Search** to find the PLC's which are available in the network as shown in the above image.

4. Once the search is completed the PLC in the network will be showed as shown below:

Click **Detect** to retrieve the hardware configurations of the PLC connected in the network.

						•••
ardware detection for	PLC_1					
	Тур	e of the PG/PC inter	face: 🖳 PN/IE		-	
		PG/PC inter	face: 🔝 Intel(R)	Ethernet Connection (4) I219-LM	0
	Compatible accessible n	odes of the selected	l interface:			
	Device	Device type	Interface type	Address	MAC address	
	plc_1.profinet interface	CPU 1516F-3 PN/	PN/IE	192.168.4.1	AC-64-17-3D-EE-7B	
	PLC connected	to the networ	k			
Flash LED						
					<u>S</u> tart se	arch
Online status information	:			Display only	error messages	
🚹 Scan completed. 1 co	ompatible devices of 1 acc	essible devices four	nd.			4
Scan and information	retrieval completed.	Click to dete	ect the actual I	hardware config	of the PLC	
Retrieving device info	ormation			iaraware comig		
					Detect Cano	el
					<u>c</u> ane	~

- 5. Select the PLC which is communicating with the SBSI Vision sensor from the PLC's found in the search.
- 6. Click **Detect** to retrieve the hardware configuration of the PLC.
- 7. Once the hardware configuration is retrieved from PLC, go to Device configuration .



4.3 Configuration of the IP parameters of the profinet interface of the PLC.



NOTE

- The Profinet Interface address of PLC, SBS Vision Sensor and the PC used for programming the PLC must be in the same range.
- 1. Double click on **the Profinet interface_1**(refer above image) to view the properties of the Profinet interface.



2. Go to **General > Ethernet address** to change the IP parameters of Profinet interface_1.

Π	Ethernet addresses						
	Interface networked with						
	Subnet:	Not networked					
		Add new subnet					
	IP protocol						
		Set IP address in the project					
		IP address: 192.168.4.1 Enter IP and Subnet					
4		Subnet mask: 255 . 255 . 0 mask					
		Use router					
-		Router address: 0.0.0.0					
		O IP address is set directly at the device					
	PROFINET						
		PROFINET device name is set directly at the device					
		Generate PROFINET device name automatically					
	PROFINET device name:	plc_1.profinet interface_1					
	Converted name:	plcxb1.profinetxainterfacexb1036c					
	Device number:	0					

5 Adding GSDML File of SBS Vision sensor to TIA Portal.

5.1 GSDML file location in the installation folder

1. The Profinet GSDML file can be found in the location where the SBS Vision sensor software in the system.

It can be found in the below location:

C:\Program Files (x86)\Festo\SBS Vision Sensor\Tools\Profinet\GSD_V1_17_129



5.2 Adding the GSD File to TIA Portal

The GSDML file of SBS Vision Sensor must be added to the TIA portal.

1. Click on **Options >> Manage General Station Description Files (GSD).**



2. Click on Manage general station description files(GSD) as shown in above image.

3. The following screen will be displayed:

N	/lanage general	station description	n files				×
	Installed GSD	s GSDs in the	project				
	Source path:	C:\PRAMOD\PROJECTS	WisionSensorFE	8\SiemensFB\Vi	sionSensorFB\Ad	ditionalFiles\	GSD
	Content of imp	oorted path					
	File		Version	Language	Status		Info
		Sele	ect the path	where GS	DML file of		
		V15	ION SENSOR	is present	in system		
	<						>
					Delete	Install	Cancel

Application Note-SBSI Vision Sensor- Integration of Host Function Blocks in SIEMENS TIA Portal V15 Seite 17 von 54

- 4. As shown in the above image select the path where the GSDML file is saved in your system.
- 5. Once the appropriate file has been chosen the following screen will be displayed:

Manage general station description	ı files		×				
Installed GSDs GSDs in the p	project						
Source path: C:\PRAMOD\PROJECTS\VisionSensorFB\SiemensFB\VisionSensorFB\AdditionalFiles\GSD							
Content of imported path	Content of imported path						
File	Version	Language					
GSDML-V2.3-Festo-SBSI-2015030	V2.3	English					
1							
Select the GSDML file							
<			>				
		Delete Install Car	ncel				

6. Once the GSDML file has been selected, click on **install** to start installing the GSDML file.

Manage general station description	n files		×		
Installed GSDs GSDs in the	project				
Source path: C:\PRAMOD\PROJECTS\VisionSensorFB\SiemensFB\VisionSensorFB\AdditionalFiles\GSD					
Content of imported path					
File	Version	Language			
GSDML-V2.3-Festo-SBSI-2015030	V2.3	English			
		Click to install CSDM			
			>		
		Delete Install Can	col		
		Delete Install Can	lei		

7. Once the GSDM file is installed the following pop up will be displayed.

Manage general station description files			×
Installation result			
! Message			
Installation was completed successfully			
Save log Install	additional files	Close	

8. Click on **CLOSE** button to finish the installation and update the hardware catalogue.

6 Configuration of SBS Vision Sensor in TIA Portal.

6.1 Adding the installed SBS Vision Sensor to Network View.



2. The network view will be displayed as shown below.

s71500 > Devices & networks			_ !
	📲 Topology view	h Network view	Device v
💦 Network 🔡 Connections HMI connection 💌 🗛 Relations 🕎 🖏 🔛 🛄 🍳 ±			E
PLC_1 CPU 1516F-3 PN			
PN/IE_1			

3. Go to Hardware Catalog as shown below.



4. Under Hardware Catalog, Go to Other Field Devices >>>Profinet IO >>> Sensors >>> Festo >>> SBSI Vision Sensor >>> SBSI.

~	Catalog
	fiel Lei
	Filter Profile: <all></all>
	Controllers
	HMI
•	PC systems
•	🛅 Drives & starters
•	Metwork components
•	🛅 Detecting & Monitoring
•	Distributed I/O
•	The Power supply and distribution
•	Tield devices
•	Ther field devices
	Additional Ethernet devices
	▼ I PROFINET IO
	Drives
	Encoders
	Gateway
	▼ JII Sensors
	▼ _ Festo
	▼ SBSI Vision Sensor
	SBSI SBSI Vision
	Sensor
	Valves
	PROFIBUS DP

5. Drag and drop the SBSI Vision sensor to the network view as shown below.

SBSIVisionsensor > Devices & networks	K Hardware catalog 🛛 🗐 🔳 🕨
🛃 Topology view 🛛 🛔 Network view 🛛 🔐 Device view	Options
💦 Network 🔢 Connections HMI connection 🔽 📅 🖽 🔟 🔍 🛳	
	✓ Catalog
	tia lia
PLC_1 5653 PL	Filter Profile: All>
Not assigned	
	BC systems
	Drives & starters
	Network components
Drag and drop the	Detecting & Monitoring
SBSI Vision sensor to	Distributed I/O
network view	Power supply and distribution
	Field devices
V	 Other field devices
< Ⅲ > 100% ·····	Additional Ethernet devices
GSD device_1 [Device] 🔹 🖪 🖸 🔄 🖬 🔂 Diagnostate 🗇 🖃 🔿	PROFINET IO
General 10 tags System constants Texts	Drives
	Encoders
General	Gateway
	- Sensors
Name: GSD device 1	SBSI Vision Sensor
	SBSI VISION SENSOR
Autor: inopass	SIEMENS AG
Comment:	PROFIBUS DP
P	_

- 6. Connect the S7-1500 PLC to SBSI Vision Sensor.
- 7. Double click on **Not Assigned** as shown below.



8. If the SBSI Vision sensor is connected to Profinet Interface _2(X2), then select PLC_1.PROFINET interface_2.

If the SBSI Vision sensor is connected to Profinet Interface $_1(X1)$, then select PLC_1.PROFINET inter face_1.

After the profinet interface is selected the connection in the device view will look as shown below:



6.2 Network Configuration of Profinet Interface SBSI Vision Sensor in TIA Portal.

1. Go to Devices and Networks and Double click SBSI Vision Sensor.



2. Once you double click the following display will be seen.



3. Click on SBSI to view the properties .



4. The properties will be viewed as shown below.

SBSI [SBSI]		Roperties
General IO tags Sy	stem constants Texts	
▼ General	General	
✓ PROFINET interface [X1]		
General	Name: SBSI	
Ethernet addresses	Author: inOppss	
 Advanced options 		
Interface options	Comment:	
 Real time settings 		
IO cycle		
RJ45 10/100 MBit/s [X1 P1]	-	
	Rack: 0	

5. The Network parameters can be viewed under Ethernet Addresses tab as shown below.





NOTE

• The IP address of the Profinet interface **SBS Vision Sensor**, **S7-1500 PLC and** IP address of your PC must be in the same range.

6.3 Assigning Profinet name to SBSI Vision Sensor.

1. Go to Devices and Networks . Double click SBSI to go to device view.

VisionSensorFB > Devices & networks					
Network Connections HMI connection	🕂 📰 🛄 🔍 ±				
PLC_1 CPU 1516F-3 PN	SBSI B				
	PLC_1				
PN	/IE_1				
	Double click on SBSI				

2. Right click on **MSE** » Click **Assign Device Name** as shown below.

	EBSI			
	-			
Right Click		Change device		
		Write IO-Device name to Micro Me	mory Card	
	_	Start device tool		
	2	Cut	Ctrl+X	
	10	Сору	Ctrl+C	
		Paste	Ctrl+V	
	>	C Delete	Del	
	5	Go to topology view Go to network view		
		Compile	•	
		Download to device	•	
	ags 🛛 💆	🗸 Go online	Ctrl+K	
	on	Go offline	Ctrl+M	
	×11 m	Online & diagnostics	lick to assign Profin	et
	<u></u>	Update and display torced operan	ds device name	υı
	es	B Cross references	E11	
	5 3	Cross-reference information	Shift+F11	
	ns 🖻	Chow catalog	Ctrl Shift C	
	ngs	Show catalog	Curi+Shint+C	
		Export module labeling strips		
	>	Properties	Alt+Enter	

3. The wizard for assigning Profinet Device name looks as shown below.

		Configured PRC	OFINET de	vice		
		PROFINET devi	ice name:	sbsi		
		De	evice type:	SBSI		
		Online access				
		Type of the PG/PC	interface:	PN/IE		-
		PG/PC	interface:	Intel(R) Ethernet Connec	tion (4) I219-LM	• 🖲 💁
		Device filter				
8						
		Only show	w devices of	the same type		
		Only show	w devices of	the same type		
		Only shov	w devices of w devices wi	the same type th bad parameter settings		
		Only shov	w devices of w devices wi w devices wi	the same type th bad parameter settings thout names		
	Accessible dev	Only shov	w devices of w devices wi w devices wi	the same type th bad parameter settings thout names		
	Accessible dev IP address	Only shov Only shov Only shov Only shov ices in the network: MAC address	w devices of w devices wi w devices wi Device	the same type th bad parameter settings thout names PROFINET device name	Status	
	Accessible dev IP address	Only shov Only shov Only shov ices in the network: MAC address	w devices of w devices wi w devices wi Device	the same type th bad parameter settings thout names PROFINET device name	Status	
	Accessible dev IP address	Only shov Only shov Only shov ices in the network: MAC address	w devices of w devices wi w devices wi Device	the same type th bad parameter settings thout names PROFINET device name	Status	
I	Accessible dev IP address	Only shov Only shov Only shov ices in the network: MAC address	w devices of w devices wi w devices wi Device	the same type th bad parameter settings thout names PROFINET device name	Status	
.	Accessible dev IP address	Only shov Only shov Only shov ices in the network: MAC address	w devices of w devices wi w devices wi Device	the same type th bad parameter settings thout names PROFINET device name	Status	
Flash LED	Accessible dev IP address	Only shov Only shov Only shov ices in the network: MAC address	w devices of w devices wi w devices wi Device	the same type th bad parameter settings thout names PROFINET device name	Status	
Flash LED	Accessible dev IP address	Only shov	w devices of w devices wi w devices wi Device	the same type th bad parameter settings thout names PROFINET device name	Status	

4. Search the available devices in the Profinet network by clicking **Update List.**

and the second se		PROFINET de	evice name:	sbsi		-
		C	Device type:	SBSI		
		Online access	;			
		Type of the PG/F	PC interface :	PN/IE		•
		PG/F	PC interface:	Intel(R) Ethernet Connec	tion (4) I219-LM	• 🖲 🖸
5		Device filter				
8						
		Uniy shi	ow devices of	the same type		
		Only sh	ow devices wi	th bad parameter settings		
		Only she	ow devices wi	thout names		
	Accessible de	vices in the network:				
	IP address	MAC address	Device	PROFINET device name	Status	
				Click to set	arch the availa	ble
				Click to sea	arch the availa evices in netwo	ble ork
Flash LED				Click to sea profinet de	arch the availa evices in netwo	ble ork

Application Note-SBSI Vision Sensor- Integration of Host Function Blocks in SIEMENS TIA Portal V15 Seite 25 von 54 5. Once the search is completed the available profinet devices in the network will be displayed.

Accessible device	es in the network:						
IP address	MAC address	Device	PROFINET device name		Status		
192.168.4.2	00-0E-F0-01-28-F5	Festo	sbsi	$\mathbf{\otimes}$	ОК		
Available	profinet devices	in netwo	rk				
<			111				>
			U	pda	te list	Assign name	

6. Click **Assign Name** to assign the profinet device name to the project.

IP address	MAC address	Device	PROFINET device name		Status	
192.168.4.2	00-0E-F0-01-28-F5	Festo	sbsi	V	ОК	
<)
			U	lpda	ate list Assign name	
					•	
					•	

6.4 Adding Data Module to SBSI.

1. Go to Devices and Networks . Double click SBSI to go to device view.

SBSIVisionsensor > Devices & networks			- i
	📲 Topology v	view 🔥 Network view	Device vi
💦 Network 🔛 Connections 🔣 HMI connection 💌 🕎 🖽 🛄 🔍	±		E
	д	IO system: PLC_1.PROFINE	T IO-System (100)
PLC_1 CPU 1516F-3 PN	SBSI SBSI PLC_1	Double click S	BSI

• Device view appears as shown below.

(3 bytes)
(6 bytes)
0

• By default CTRL and STAT bytes will be there in the SBSI configuration as shown below.

Device	overview					
- *	Module	Rack	Slot	I address	Q address	Туре
	▼ SBSI	0	0			SBSI
	Interface	0	0 X1			SBSI
	CTRL (3 bytes)_1	0	1		68	CTRL (3 bytes)
	STAT (6 bytes)_1	0	2	1015		STAT (6 bytes)



- Based on the payload selection the telegram in Vision sensor configuration studio, Select the appropriate data module.
- The available options are :



• Drag and drop the needed data module to the Device Overview as shown below. In our example we have considered DATA(2+ 256 bytes).

Module	Rack	Slot	I address	Q address	Туре	✓ Catalog
▼ SBSI	0	0			SBSI	<search></search>
Interface	0	0 X1			SBSI	Filter Profile: <all></all>
CTRL (3 bytes)_1	0	1		68	CTRL (3 bytes)	head module
STAT (6 bytes)_1	0	2	1015		STAT (6 bytes)	
DATA (2 + 256 bytes)_1	0	3	16273		DATA (2 + 256	Control Module (CTPL)
	0	4	A COLOR			Data Module (DATA)
	0	5				DATA (2 + 128 bytes)
	C)rag ar	nd drop th	ne neede	ed	DATA (2 + 16 bytes)
		(data mod	lule		DATA (2 + 256 bytes)
						DATA (2 + 32 bytes)
						DATA (2 + 64 bytes)
						DATA (2 + 8 bytes)

Identifying Hardware ID of the Sub modules of SBSI Vision Sensor. 6.5

Mo	odule	Rack	Slot	I address	Q address	Туре	
•	SBSI	0	0			SBSI	
	Interface	0	0 X1			SBSI	Select CTPI
	CTRL (3 bytes)_1	0	1		68	CTRL (3 bytes)	module
-	STAT (6 bytes)_1	0	2	1015		STAT (6 bytes)	module
	DATA (2 + 256 bytes)_1	0	3	16273		DATA (2 + 256 byt	
		0	4				
		0	5				

In device view, select CTRL module as shown below. ٠

Select **System Constants** as shown below from the Properties tab of **CTRL** module. .

		Go to	system c	onstants				
CTRL (3 bytes)_1	[CTRL (3 b	ytes)]					🗟 Properties	1.1
General IO	tags	System	constants	Texts				
Show hardware sys	stem consta	nt 🔻						
Name				Туре	•	Hardware identifie	r Used by	
SBSI~CTRL_(3_bytes)_1			Hw_	SubModule	263	PLC_1	
• In device view , sel	ect STAT r	nodule a	as shown be	elow.	Ch	RL module		
Module	Rack	Slot	I address	Q address	Туре			
 SBSI 	0	0			SBSI			
Interface	0	0 X1			SBSI			
CTRL (3 bytes)_1	0	1		68	CTRL (3 bytes)	Sele	ct STAT	
STAT (6 bytes)_1	0	2	1015		STAT (6 bytes)	m m	odule	
DATA (2 + 256 bytes)_1	0	3	16273		DATA (2 + 256 b	oyt		

• Select **System Constants** as shown below from the Properties tab of **CTRL** module.



Hardware ID of STAT module

• In **device view**, select **DATA module** as shown below.

Device o	overview						
📲 N	Module	Rack	Slot	I address	Q address	Туре	Art
	 SBSI 	0	0			SBSI	
	Interface	0	0 X1			SBSI	
	CTRL (3 bytes)_1	0	1		68	CTRL (3 bytes)	
	STAT (6 bytes)_1	0	2	1015		STAT (6 bytes)	
	DATA (2 + 256 bytes)_1	0	3	16273		DATA (2 + 256 byt	

Select DATA module

• Select **System Constants** as shown below from the Properties tab of **DATA** module.

eneral	IO tags	System constants	Texts		
ow hardwa	are system con	stant 🔻			
Name			Туре	Hardware identifier	Used by
SBSI~D)ATA (2 + 256	bytes) 1	Hw SubModule	268	PLC 1

Hardware ID of DATA module

7 Linking VisionSensor library to the project.

- 1. Unzip the library enclosed in the application note into a folder of your choice.
- 2. Click on Options > Global Libraries > Open Library as shown below.



3. Choose the path where the Library is saved in your system. Select the library and click OPEN.

M Open global	library			×	
Look in:	SBSI-Siemen	\$	⊻ 🧿 🎓 📂 🛄▼		
Quick access Desktop Libraries	Name AdditionalFi IM System TMP UserFiles XRef SBSI-Siemer Select t	s.al15_1 he library	Date modified 16-08-2019 16:11 16-08-2019 16:11 16-08-2019 16:16 16-08-2019 16:11 16-08-2019 16:11 16-08-2019 16:11 16-08-2019 16:16	Type File folder File folder File folder File folder File folder Siemens T	
Network	< File name: Files of type:	SBSI-Siemens.al15_1 Global library ☑ Open as read-only	~	> Open Cancel	Click after selecting library

4. The opened library is now available in " Libraries > Global Libraries".

Libraries	ת ₪ ו		
Options			
🛨 Library view 🙆		Har	
✓ Project library		dwa	
📸 🗄 All	- 📑 '	reo	
Image: Project library		ata	
		log	
		۷.	
		On	
		line	
 Global libraries 		to	
률₫чьъ∋⊮ 🗉	ž 🗉 '	slo	
U Buttons-and-Switches			
Drive_Lib_S7_1200_1500			
Drive_Lib_\$7_300_400		Tas	
Long Functions		sks	
Monitoring-and-control-obje	cts		
Documentation templates			
• 🛄 SBSI-Siemens		F	
•		ario	
New data data 1		s	Click to view
Newly added vision			Global Libraries
sensor Library			

5. Open the SBSI-Siemens Library to view the content.



- 6. The Library has the following sub folders:
- Function Blocks : It has the Function Block named **FB_CheckSBSI** which is used to control the SBSI Vision Sensor functions.
- 7. Drag and drop the folder **Function Blocks** to the folder **Program Blocks** in your project. Drag and drop the folder Function Blocks to the project as shown below.



8. Once the folders from the library has been copied to the project, the **Project Tree** will look as shown below.



8 DESCRIPTION OF SBSIVISION-Siemens Library FUNCTION BLOCKS

The library has 1 Function Blocks within it:

1. FB_CheckSBSI

8.1 FB_CheckSBSI

The Functions block has the following features:

- It allows the user to do Vision Sensor control operations. Using this Function Block the user can control the triggering of the SBSI Vision Sensor.
- Using the Function Block the job number of the vision sensor can be changed.
- It gives the various status information of the vision sensor operation like **Active job number**, **Error information** occurred during the operation, **Job change acknowledgement**, **Trigger acknowledgement**.
- It allows outputs the payload result data of the active job.

The below image shows the schematic view of the FB_CheckSBSI block.



The following tables gives a detailed explanation of the inputs and outputs of the function block.

INPUT DATA

NAME	DATA TYPE	DESCRIPTION
iSBSICtrl_HWID	HW_ANY	Hardware ID of the Ctrl Module of SBSI Vision Sensor.
iSBSIStat_HWID	HW_ANY	Hardware ID of the Stat Module of SBSI Vision Sensor.
iSBSIData_HWID	HW_ANY	Hardware ID of the Data Module of SBSI Vision Sensor.
xEnable	BOOL	FALSE – Disable the Function Block. TRUE – Enable the Function Block.
xTrigger	BOOL	TRUE – Rising Edge (Low ==> High) triggers the Vision Sensor.
xTriggerExt	BOOL	TRUE – Hardware Trigger or free run enabled.
		FALSE – Hardware Trigger or free run disabled.

Application Note-SBSI Vision Sensor- Integration of Host Function Blocks in SIEMENS TIA Portal V15 Seite 33 von 54

NAME	DATA TYPE	DESCRIPTION
uiJobNumber	BOOL	Gives the active job number of the vision sensor. If the value doesn't equal with the active job number ,then the request for job change will be triggered automatically.

Table 5.1: FB_CheckSBSI Input Data

OUTPUT DATA

NAME	DATA TYPE	DESCRIPTION
xEnabled	BOOL	TRUE – Function Block is enabled FALSE - Function Block is disabled
xReady	BOOL	FALSE – SBSI Vision Sensor is not ready for next evaluation cycle.TRUE – SBSI Vision Sensor is ready for next evaluation cycle.
xTriggerAck	BOOL	 FALSE – No acknowledge for a successful trigger to SBSI Vision Sensor. TRUE – Acknowledge for a successful trigger to SBSI Vision Sensor.
xPartChecked	BOOL	FALSE – SBSI VisionSensor is busy in operation. TRUE – SBSI VisionSensor is wait- ing for next command.
aData	ARRAY[0257] OF BYTES	 BYTE 0 – Gives the Image ID of the job being exe- cuted. Image ID is incre- mented with each job exe- cution independent from trigger source. BYTE 1 – Bit0 of this byte has following values. Bit0 is 1 means Data Overrun = Data truncated. Bit0 is 0 means No Data overrun. BYTE 2 to BYTE 257 - Data as defined in Vision Config- uration Studio in Out- put/Telegram/Payload.
xError	BOOL	FALSE – No Error during SBSI Vision Sensor operation. TRUE – Error during SBSI Vision Sensor operation.

NAME	DATA TYPE	DESCRIPTION
uiErrorld	BOOL	Gives the ID of the error occured. It has following values.
		0 – No Error.
		1 – Failure Trigger request
		2 – Failure Change Job.
		3 – Failure switch to run.
		5 – Failure Profinet Not active in Job.
		15 – System Error
tTimeTotal	TIME	Time period for which the Vision Sensor is busy.

Table 5.2: FB_CheckSBSI Output Data

9 INTEGRATION OF FUNCTION BLOCKS INTO PROJECT

9.1 FB_CheckSBSI

Drag and drop the FB_CheckSBSI function block to the TIA portal to the Main(OB1) as shown below.



→

 \rightarrow

 \rightarrow

NOTE

Refer Table 5.2 in Chapter 8.1 to get detailed description of the Inputs and Outputs to be configured for the Function Block.

The following are the inputs the user has to configure properly :

1. iSBSICtrl_HWID : The hardware ID of the Ctrl module must be given to this input .

NOTE

• Refer Chapter 6.5 to get detailed description on how to identify the Hardware ID of iSBSICtrl_HWID module in TIA Portal.

2. iSBSIStat_HWID: The hardware ID of the Stat module must be given to this input.

NOTE

- Refer Chapter 6.5 to get detailed description on how to identify the Hardware ID of iSBSIStat_HWID module in TIA Portal.
- 3. iSBSIData_HWID: The hardware ID of the Data module must be given to this input.

NOTE

• Refer Chapter 6.5 to get detailed description on how to identify the Hardware ID of iSBSI-Data_HWID module in TIA Portal.

4. xEnable :

In order to make the Function Block operational make this input **TRUE**. In order to disable the operations of this Function Block make this input **FALSE**.

5. uiJobNumber:

The job which the Vision Sensor has to execute has to be given in this input. If this value doesn't equal with the Active job number of the vision sensor then the request for changing job number will be triggered automatically.

6. After the Hardware ID's of the modules are added the Function Block overview will appear as shown below.



10 FUNCTION BLOCK EXECUTION WITH AN EXAMPLE

This chapter will explain the following in detail:

- 1. To demonstrate the FB execution with an example where we have to detect a barcode using a SBS vision sensor.
- 2. Configuration of the Vision Configuration Studio to detect a sample of barcode.
- 3. Configuration of telegram in Vision Configuration Studio.
- 4. Configuration of data to be communicated with the Siemens PLC in the telegram.
- 5. Creating watch tables in TIA portal to monitor the following :
 - Data sent from the SBS Vision Sensor in the telegram.
 - Status of the SBS Vision Sensor.
- 6. Mapping the result data configured in vision configuration studio with the array of data obtained as an output of the Function Block **FB_CheckSBSI**.

10.1 Example configuration in Vision Configuration Studio.

1. Creating a new Job.

- Create a new Job named Job1. Go to Job >> New.
- Refer the image below to create a new job.

Setup				
Job				
Alignme	ent			
Detect	or			
Outpu	t			
Resul	t			
Start ser	nsor			
Trigger/Image u Trigger	pdate Single continuous	- Fit		
[]	1			•
Name	Description	Author	Created Chang	jed
Click to cre	ate a new j	job	22-08-20 22-08-	20
New	Load	Save	Delete	Delete all

2. Image Acquisition settings.



• Click Auto Shutter under image acquisition tab.

			Cor	nfigure job	
				C	Click on Auto
Image acquisition	Pre-processing	Calibration	Cycle time		Shutter
Resolution WVGA (736x480 🗢	Shutter speed	75.000 Auto s	Qua hutter		
Dynamic Linear 🔷 Trigger mode Trigger Free run	Gain	1.00	Inte	rnal illumination	

• In the image window the bar code sample which the vision sensor focuses on will be displayed as shown below.

1.1			1.2
QR - Hello World	•		QR - FESTO S
	• . ·	•	
•			

- 3. Detector Selection.
 - Go to **Detector** . Select **Data Code** as shown below.

	🕎 Vision Sensor Configuratio	n Stı	udio	o - Ur	niversal				 -	
	File View Options Help									
		P		 [6 6 9	Ş				
	Setup Job Alianment	A	vail	lew d able d	letector etector types		?	×		
Go to Detector	Detector				Detector type	Description				
	Output		1		Barcode	Barcodes detection (1D co	odes)			
	Result		2	<u>18</u>	Datacode	Datacode detection (2D c	odes)	-	Select	Datacode
	Start sensor		3	A	OCR	Read printed characters				
			4	÷	Pattern matching	Locate object by grayscal	e pattern	1		
			5	0	Contour	Locate and count objects	by objec	t conto		
			6	•	Contrast	Verify contrast in specifie	d region			
			7	*	Brightness	Verify brightness in specif	fied regio	on		
			8	•	Gray	Verify gray level in specifi	ied regio	n		
	- Trigger/Image undate		9	F	Caliper	Distance between edges				
	Single		10	C	BLOB	Count and evaluate object	ts			
	Trigger Continuous		4					Þ		
	Connection mode				ОК	Cancel				
	Online Offline	-][Fit	+	< > > > > > > > > > > > > > > > > > > >		_		

4. Interface Settings.

• Go to **Output** tab.

	Vision Sensor Configuration File View Options Help Image: Configuration Image: Configuration Image: Confi	on Studio - Universal	
Go to Output tab	Setup Job Alignment Detector Output Result Start sensor	1.1 QR - Hello World	1.2 QR - FESTO SI

• Under Output tab, Go to Interfaces as shown below.

			G	o to miler	laces								
I	/0	mapping	Digital output	Interfaces	Timing	1	Telegram	Image tra	nsı	mission	Archiving		
											,		
		Name		Setting 1			Setting 2			Setting 3		Logical outputs	Enable
	1	Internal I/O)	PNP		\$)						4
	2	Serial		RS422		\$	19200 Bd		•	8N1	\$	0	
	4	Ethernet		(IN)2006	*		(Out)2005					0	
	5	EtherNet/IP										0	
	6	PROFINET											√
	7	Vision Sens	or Visualisation St	udio Image an	d overlay	\$)						
	8	SBSWebVie	ewer										

Go to interfaces

• Enable **PROFINET** under interfaces tab as shown below.

F	I/C	D mapping Digital output	Interfaces Timing	Telegram Image trans	mission Archiving			_
		Name	Setting 1	Setting 2	Setting 3	Logical outputs	Enable	
	1	Internal I/O	PNP 🗘				4	
	2	Serial	RS422 \$	19200 Bd 🗘	8N1 🗧	0		
	4	Ethernet	(IN)2006	(Out)2005 🚔		0		Oplast
	E	EtherNet/IP						Select
	6	PROFINET		_			~	PROFINET
-	1	vision sensor visualisation studi	o (image and overlay 👘 🗘)				
	8	SBSWebViewer						

5. Telegram settings.

• Under Output , Go to Telegram.

		Go to Tele	egram			
		1				
I/O mapping D	igital output Interfaces	Timing Telegra	Image transmission	Archiving		
Binary 🗘	Start					5
	Trailer)
	Separator]	
	End of Telegram				ANSI	0
Save to file	Selected fields		Data length		Status	
Reset	Detector result		Digital outputs		Logical outputs	
	Execution time		Active job no.		Checksum	

	• Ensur								
Enable	I/O mapping	Digital output	Interfaces	Timing	Telegram	Image transmission	Archiving		
Binary Format	Binary	Start Trailer							
		Separator End of Tele	gram					ANSI	•
	Save to file	Selecte	d fields			Data length		Status	
	Keset	Executi	or result on time			Active job no.		Checksum	

6. Define the telegram to send the data needed to the PLC.

• Go to **Output >>> Telegram**. The following tab will be visualised.

I/O mapping	Digital output Interfaces	Timing Telegram	Image transmission	Archiving	
Binary 🗘	Start				
	Trailer				
	Separator]
	End of Telegram				ANSI
Save to file	Selected fields		Data length		Status
Reset	Detector result		Digital outputs		Logical outputs
	Execution time		Active job no.		Checksum
Reset	Detector result Execution time		Digital outputs Active job no.		Logical outputs Checksum

7. Configure the payload to be sent to the PLC.

• Click + button to add a new payload data as shown below.

ay	Activ	Detector	Value	Min. lengt	No. of result	To add new		+		
1	✓	Select				item to		-		
		1	+			payload		Jp		
New item added to payload										

- Select the detector whose data has to be added to the payload.
- Select the value of the detector which has to be added to the payload.

ay	load				
	Activ	Detector	Value	Min. lengt	No. of result
1 Detector1			DataCode-1: Position X	0	
	Sele det	ect the ector	Select the v	/alue :or	

- Repeat the process to add other Detector values also to the telegram.
- In our example we have considered the following data to the telegram.

ayl	load				
	Activ	Detector	Value	Min. lengt	No. of result
1	✓	Detector1	DataCode-1: Position X	0	
2	✓	Detector1	DataCode-1: Position Y	0	
3	✓	Detector1	DataCode-1: String	0	
4	¥	Detector1	DataCode-2: String	0	

8. Go to the Result tab to check the result data.

Dec	Jecoded results																			
	Decoded string	Trunca	String le	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Contrast	Decoding	Module he	Module w	Position X	Position Y	Angle	Compare result
1.	L Hello World		11	n/a	63	0	6	6	192.6	65.5	88.0	٥								
1.3	2 FESTO SBSI		10	n/a	60	0	6	6	667.8	88.3	88.2	٥								

9. After the configurations are done ensure that the Vision Sensor is in Trigger Mode and not in Free Run mode.

10. Go to Job >> Image Acquisition >>> Trigger Mode.

		Co	onfigure job
Image acquisition	Pre-processing Calibra	ition Cycle time	
Resolution WVGA (736x480 🗢	Shutter speed	Qu 75.000 m: 🗢 Auto shutter	uadrants
Dynamic Linear 🗢 Trigger mode	Gain	In 1.00 🗢 C Ex	ternal illumination on ¢ ternal illumination
 Trigger Free run 	Enable trigger mode	C	off 🔷 🗢

11. Click on **Start sensor** to download the configurations to the SBS vision sensor.

Job		1.2
Alignment	. Martine	S-MA
Detector	QR - Hello World	QR - FESTO SE
Output		
Result		
Start sensor		

Click Start sensor

10.2 Example description in TIA Portal

- 1. Integrate the Function Block FB_CheckSBSI as explained in Chapter 9.
- 2. Create a watch table.



3. Give a name to the watch table. Here in our example we have named the watch table as PayLoad Data.

		i	Name		Address	Display format	Monitor value	Modif
▼ 🔄 VisionSensorFB	^ 1				<add new=""></add>			
🚔 Add new device								
Devices & networks								
PLC_1 [CPU 1516F-3 PN/DP]								
Device configuration								
🖳 Online & diagnostics	J.							
Software units	L.							
Program blocks								
Technology objects								
External source files								
🕨 🚂 PLC tags	н							
PLC data types	L.							
 Watch and force tables 	н			Newly	added watch	table		
💕 Add new watch table								
Fill Force table								
BayLoadData	T							
Online backups								

- 4. Add the **aData** output of the **FBCheckSBSI.** Add all the 258 array elements of aData to the watch table for testing purpose.
- 5. Add the first element of the array aData to the watch table as shown below.



6. Click the corner of the 1st element and when a "+" mark appears drag the cursor down until all 256 elements of the array are added to the watch table.

🦻 👻 1	# 😼 🗓 🕫 16 🌮 😤 💁 🖤 🕯				
i	Name	Address	Display format	Monitor value	Modify value
1	"FB_CheckSBSI_DB".aData[0]		Hex		
	Click the o mark appea down until a array	corner and afte ars scroll the cu all 256 element y are added	r + Irsor s of		

7. Once the array elements are added the watch table looks as shown below.

🏥 🔰 🗓 🍠 🧏 🖤 🖤				
Name	Address	Display format	Monitor value	Modify value
"FB_CheckSBSI_DB".aData[0]		Hex		
"FB_CheckSBSI_DB".aData[1]		Hex		
"FB_CheckSBSI_DB".aData[2]		Hex		
"FB_CheckSBSI_DB".aData[3]		Hex		
"FB_CheckSBSI_DB".aData[4]		Hex		
"FB_CheckSBSI_DB".aData[5]		Hex		
"FB_CheckSBSI_DB".aData[6]		Hex		
"FB_CheckSBSI_DB".aData[7]		Hex		
"FB_CheckSBSI_DB".aData[8]		Hex		
"FB_CheckSBSI_DB".aData[9]	1	Hex	-	
"FB_CheckSBSI_DB".aData[10]		Hex		
"FB_CheckSBSI_DB".aData[11]		Hex		
"FB_CheckSBSI_DB".aData[12]		Hex		
"FB_CheckSBSI_DB".aData[13]		Hex		
"FB_CheckSBSI_DB".aData[14]		Hex		
"FB_CheckSBSI_DB".aData[15]		Hex		
"FB_CheckSBSI_DB".aData[16]		Hex		
"FB_CheckSBSI_DB".aData[17]		Hex		
"FB_CheckSBSI_DB".aData[18]		Hex		
"FR_CheckSRSL_DR" aData[19]		Hex		

8. The first 2 data in the payload are of the **Floating data type** (Double word). So configure the display format of Byte2 to Byte9 as **Hex**.

The next 2 data in the payload are of **String** type . So configure the display format in Watch Table as **Char** from Byte10 onwards.

"FB_CheckSBSI_DB".aData[0]	Hex 💌
"FB_CheckSBSI_DB".aData[1]	Hex
"FB_CheckSBSI_DB".aData[2]	Hex
"FB_CheckSBSI_DB".aData[3]	Hex
"FB_CheckSBSI_DB".aData[4]	Hex
"FB_CheckSBSI_DB".aData[5]	Hex
"FB_CheckSBSI_DB".aData[6]	Hex
"FB_CheckSBSI_DB".aData[7]	Hex
"FB_CheckSBSI_DB".aData[8]	Hex
"FB_CheckSBSI_DB".aData[9]	Hex
"FB_CheckSBSI_DB".aData[10]	Character
"FB_CheckSBSI_DB".aData[11]	Character
"FB_CheckSBSI_DB".aData[12]	Character
"FB_CheckSBSI_DB".aData[13]	Character
"FB_CheckSBSI_DB".aData[14]	Character
"FB_CheckSBSI_DB".aData[15]	Character
"FB_CheckSBSI_DB".aData[16]	Character
"FB_CheckSBSI_DB".aData[17]	Character
"FB_CheckSBSI_DB".aData[18]	Character
"FB_CheckSBSI_DB".aData[19]	Character
"FB_CheckSBSI_DB".aData[20]	Character
"FB_CheckSBSI_DB".aData[21]	Character
"FB_CheckSBSI_DB".aData[22]	Character
"FB_CheckSBSI_DB".aData[23]	Character
"FB_CheckSBSI_DB".aData[24]	Character
"FB_CheckSBSI_DB".aData[25]	Character
"FB_CheckSBSI_DB".aData[26]	Character
"FB_CheckSBSI_DB".aData[27]	Character
"FB_CheckSBSI_DB".aData[28]	Character
"FB_CheckSBSI_DB".aData[29]	Character
"FB_CheckSBSI_DB".aData[30]	Character

Application Note-SBSI Vision Sensor- Integration of Host Function Blocks in SIEMENS TIA Portal V15 Seite 45 von 54

- 9. Enter **iSBSICtrl_HWID**, **iSBSIStat_HWID**, **iSBSIData_HWID**. Refer **Chapter 6.5** to identify the hardware ID's of Ctrl , Stat and Data modules of SBSI.
- 10. Enable the Function Block. This is done by making **xEnable = TRUE**. Show how to toggle at least one variable. Through Watch table, for example.

11. Toggle input **xTrigger = 1**.

.

Observe the data received in the aData output of the FB. Monitor the watch table output.



The online payload data values of TIA portal match with the Results of the vision configuration studio (**Refer Step 8 in Section 10.1**).

10.3 Payload Data Mapping between TIA Portal and Vision Configuration Studio.

This chapter explains how the payload configured in Vision Configuration Studio and the data appearing in TIA portal(**aData output of the FB**) are mapped with each other.

1. Consider we have the following payload configuration in Vision Sensor Configuration Studio.

Pay	Payload												
	Activ	Detector	Value	Min. lengt	No. of result								
1	✓	Detector1	DataCode-1: Position X	0									
2	✓	Detector1	DataCode-1: Position Y	0									
3	✓	Detector1	DataCode-1: String	0									
4	∢	Detector1	DataCode-2: String	0									

2. TIA Portal online values of the payload data received from SBS Vision sensor.

Name	Address	Display format		Monitor value		
"FB_CheckSBSI_DB".aData[0]		Hex	•	16#02 🧲		Image ID
"FB_CheckSBSI_DB".aData[1]		Hex		16#00		
"FB_CheckSBSI_DB".aData[2]		Hex		16#00		
"FB_CheckSBSI_DB".aData[3]		Hex		16#02		
"FB_CheckSBSI_DB".aData[4]		Hex		16#F1	_	Datacode 1 Pos X
"FB_CheckSBSI_DB".aData[5]		Hex		16#C5		
"FB_CheckSBSI_DB".aData[6]		Hex	1	16#00		
"FB_CheckSBSI_DB".aData[7]		Hex		16#00		Deterred A Dee V
"FB_CheckSBSI_DB".aData[8]		Hex		16#FB	~	Datacode1 Pos r
"FB_CheckSBSI_DB".aData[9]		Hex		16#A4		
"FB_CheckSBSI_DB".aData[10]		Character	-	°H'	1	
"FB_CheckSBSI_DB".aData[11]		Character		'e'		
"FB_CheckSBSI_DB".aData[12]		Character		T		
"FB_CheckSBSI_DB".aData[13]		Character		т		
"FB_CheckSBSI_DB".aData[14]		Character		'o'		Datcode1
"FB_CheckSBSI_DB".aData[15]		Character				String
"FB_CheckSBSI_DB".aData[16]		Character		'W'		oung
"FB_CheckSBSI_DB".aData[17]		Character		'o'		
"FB_CheckSBSI_DB".aData[18]		Character		'r'		
"FB_CheckSBSI_DB".aData[19]		Character		т		
"FB_CheckSBSI_DB".aData[20]		Character		'd'	J	
"FB_CheckSBSI_DB".aData[21]		Character		'F'	r	
"FB_CheckSBSI_DB".aData[22]		Character		'E'		
"FB_CheckSBSI_DB".aData[23]		Character		'S'		
"FB_CheckSBSI_DB".aData[24]		Character		Т		Datacode?
"FB_CheckSBSI_DB".aData[25]		Character		'0'		Datacouez
"FB_CheckSBSI_DB".aData[26]		Character				sung
"FB_CheckSBSI_DB".aData[27]		Character		'S'		
"FB_CheckSBSI_DB".aData[28]		Character		'B'		
"FB_CheckSBSI_DB".aData[29]		Character		'S'		
"FB_CheckSBSI_DB".aData[30]		Character		T		
"FB_CheckSBSI_DB".aData[31]		Hex		16#00		
"FB_CheckSBSI_DB".aData[32]		Hex		16#00		

3. Setup of payload data in Vision Configuration studio as .csv file.

• Go to Output >>> Telegram >>> Save to File.

I/O mapping	Digital output	Interfaces	Timing	Telegram	Image transmission	Archiving
Binary 🔷	Start					
	Trailer					
	Separator					
	End of Tele	gram				
Save to file	Selecte	ed fields			Data length	
Reset	Detecto	or result			Digital outputs	
_	Executi	on time			Active job no.	

Generates .csv

•

Select the location where the .csv file has to be saved.

→	~ Ü	Search VisionSen	sorFB	٦
rganize 🔹 New folder			• •	(
🍤 (C:) OSDisk 🔨 Name	Date modified	Туре	Size	
Appl BerndDocs	09-07-2019 13:54	File folder		
Intel CodesysFB	08-07-2019 19:14	File folder		
MININT	23-08-2019 11:18	File folder		
PerfLogs DocsImages	23-08-2019 11:28	File folder		
PRAMOD GermanVisit	08-07-2019 19:22	File folder		
Program Files	11-07-2019 18:17	File folder		
Program Files	08-07-2019 18:46	File folder		
Quarantine	16-08-2019 16:18	File folder		
Emp				
Licers V K				
File name: Payload				
Save as type: *.csv				

- ∧ Hide Folders
 - Click on **Save** to save the file. •
 - Right click on the .csv file and edit with notepad. This will open a notepad file as shown below. Payload.csv - Notepad

```
File Edit Format View Help
Byte position;Data type;Field;Detector name;Value;Length;Detector number;Detector type
1;Integer;Detector;Detector1;Position X;4;1;DataCode;
5; Integer; Detector; Detector1; Position Y; 4; 1; DataCode;
9;String (or: Array of char);Detector;Detector1;String;Runtime;1;DataCode;
```

Edit the Notepad file by adding **sep=;**. • Now the edited notepad file will look as shown below.

	Add this line
	T Company of the second s
	ayload.csv - Notepad
F	ile Edit Format View Help
s	sep=;
E	Byte position;Data type;Field;Detector name;Value;Length;Detector number;Detector type
1	l;Integer;Detector;Detector1;Position X;4;1;DataCode;
5	;;Integer;Detector;Detector1;Position Y;4;1;DataCode;
9	;String (or: Array of char);Detector;Detector1;String;Runtime;1;DataCode;
	-

- **SAVE** and close the file.
- Now open the excel sheet. It will have the following view.

• The payload data can be seen in the above excel sheet. The order in which the data was configured in Vision sensor configuration studio in the same order data is displayed in the excel sheet.

Byte position	Data type	Field	Detector name	Value	Length	Detector number	Detector type
1	Integer	Detector	Detector1	Position X	4	1	DataCode
5	Integer	Detector	Detector1	Position Y	4	1	DataCode
9	String (or: Array o	Detector	Detector1	String	Runtime	1	DataCode
9	String (or: Array o	Detector	Detector1	String	Runtime	1	DataCode

• The following image shows how the payload data between Vision Sensor configuration studio and TIA portal.

	Address	Display format	Monitor value					
3_CheckSBSI_DB*.aData[0]	(Hex	 16#02 	Image ID				
_CheckSBSI_DB*.aData[1]		Hex	16#00					
_CheckSBSI_DB*.aData[2]		Hex	16#00					
_CheckSBSI_DB".aData[3]		Hex	16#02					
_CheckSBSI_DB*.aData[4]		Hex	16#F1	Datacode 1 Pos X				
3_CheckSBSI_DB*.aData[5]		Hex	16#C5					_
B_CheckSBSI_DB*.aData[6]		Hex	16#00	Byte position Data type Field	Detector name	Value	Length	De
B_CheckSBSI_DB*.aData[7]		Hex	16#00	1 Integer Deter	tor Detector1	Position)	(4
B_CheckSBSI_DB*.aData[8]		Hex	16#FB	Datacode1 Pos Y		P Usicion /		
B_CheckSBSI_DB*.aData[9]		Hex	16#A4	5 Integer Detec	tor Detector1	Position Y	1 1	4
B_CheckSBSI_DB".aData[10]		Character	°H'	9 String (or: Array o Detec	tor Detector1	String	Runtime	
B_CheckSBSI_DB".aData[11]		Character	'e'	0 String (our Arrows) Date	ter Detector1	Chring	Dunting	
B_CheckSBSI_DB".aData[12]		Character	T.	9 String (or: Array o beled	tor Detector1	String	Runume	
B_CheckSBSI_DB".aData[13]		Character	т					
3 CheckSBSI DB".aData[14]		Character	'o'	 Datcode1 				
CheckSBSI_DB".aData[15]		Character	10 C	String				
B_CheckSBSI_DB*.aData[16]		Character	'W	Guilig				
B_CheckSBSI_DB*.aData[17]		Character	'o'					
FB_CheckSBSI_DB".aData[18]		Character	19 M					
B_CheckSBSI_DB".aData[18] B_CheckSBSI_DB".aData[19]		Character Character	97 17					
FB_CheckSBSI_DB".aData[18] FB_CheckSBSI_DB".aData[19] FB_CheckSBSI_DB".aData[20]		Character Character Character	171 111 112					
FB_CheckSBSI_DB".aData[18] FB_CheckSBSI_DB".aData[19] FB_CheckSBSI_DB".aData[20] FB_CheckSBSI_DB".aData[21]		Character Character Character Character	9 11 16 17					
B_CheckSBSI_DB*.aData[18] B_CheckSBSI_DB*.aData[19] B_CheckSBSI_DB*.aData[20] B_CheckSBSI_DB*.aData[21] B_CheckSBSI_DB*.aData[22]		Character Character Character Character Character	17 11 18 19 19					
B_CheckSBS_DB*_aData[18] B_CheckSBS_DB*_aData[19] B_CheckSBS_DB*_aData[20] B_CheckSBS_DB*_aData[21] B_CheckSBS[_DB*_aData[22] B_CheckSBS[_DB*_aData[23]		Character Character Character Character Character Character	12. 14. 14. 14.					
B_CheckSBS_DB*aData118] #2_CheckSBS_DB*aData199] #2_CheckSBS_DB*aData120] #2_CheckSBS_DB*aData121] #2_CheckSBS_DB*aData122] #2_CheckSBS_DB*aData123] #3_CheckSBS_DB*aData124]		Character Character Character Character Character Character Character	17 17 18 19 19 19 19 19 19 19 17					
B_CheckSS_DB* abba[18] B_CheckSS_DB* abba[19] B_CheckSS_DB* abba[20] B_CheckSS_DB* abba[21] B_CheckSS_DB* abba[22] B_CheckSS_DB* abba[23] B_CheckSS_DB* abba[24] B_CheckSS_DB* abba[25]		Character Character Character Character Character Character Character Character	γ Τ' 'δ' 'F' 'E' 'S' 'T 'O'	Datacode2				
B_CheckSS5_DB*ADba[18] B_CheckSS5_DB*ADba[20] B_CheckSS5_DB*ADba[21] B_CheckSS5_DB*ADba[21] B_CheckSS5_DB*ADba[22] B_CheckSS5_DB*ADba[23] B_CheckSS5_DB*ADba[24] B_CheckSS5_DB*ADba[25] B_CheckSS5_DB*ADba[26]		Character Character Character Character Character Character Character Character Character	۰٬۰ ۲۰ ۶۰ ۴۶ ۲۶ ۲۰ ۲۰	Datacode2 string				
#_check55_D#*aba118] #_check55_D#*aba120] #_check55_D#*aba120] #_check55_D#*aba121] #_check55_D#*aba122] #_check55_D#*aba123] #_check55_D#*aba124] #_check55_D#*aba125] #_check55_D#*aba126] #_check55_D#*aba126]		Character Character Character Character Character Character Character Character Character Character	'Y' 'H' 'B' 'F' 'S' 'T 'O' 'S'	Datacode2 string				
S_check351_06*_abea[18] S_check351_06*_abea[19] S_check351_06*_abea[20] S_check351_06*_abea[21] S_check351_06*_abea[22] S_check351_06*_abea[23] S_check351_06*_abea[24] S_check351_06*_abea[25] S_check351_06*_abea[26] S_check351_06*_abea[26] S_check351_06*_abea[26] S_check351_06*_abea[26]		Character Character Character Character Character Character Character Character Character Character Character Character Character	'r' 'r' 'd' 'F' '5' '7' '0' '5' '5' '5'	Datacode2 string				
E_Deck55_DP*Abba[18] E_Deck55_DP*Abba[19] E_Deck55_DP*Abba[20] E_Deck55_DP*Abba[21] E_Deck55_DP*Abba[23] E_Deck55_DP*Abba[23] E_Deck55_DP*Abba[25] E_Deck55_DP*Abba[25] E_Deck55_DP*Abba[25] E_Deck55_DP*Abba[26] E_Deck55_DP*Abba[28] E_Deck55_DP*Abba[28]		Character Character Character Character Character Character Character Character Character Character Character Character Character Character	'r' 'r' 'd' 'F' '5' 'T '0' '' '8' '8' '8'	Datacode2 string				
P2_Check85_D0*Abba118] P2_Check85_D0*Abba199 P2_Check85_D0*Abba120] P2_Check85_D0*Abba120] P2_Check85_D0*Abba121] P2_Check85_D0*Abba123] P2_Check85_D0*Abba126] P2_Check85_D0*Abba126] P2_Check85_D0*Abba126] P2_Check85_D0*Abba126] P2_Check85_D0*Abba128] P2_Check85_D0*Abba128] P2_Check85_D0*Abba128] P2_Check85_D0*Abba128] P2_Check85_D0*Abba128] P2_Check85_D0*Abba128]		Character Character Character Character Character Character Character Character Character Character Character Character Character Character Character	Y T 'd' 'F' 'S' 'T 'O' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S'	Datacode2 string				
P2_Check85_DP3_abba[18] P2_Check85_DP3_abba[19] P2_Check85_DP3_abba[20] P2_Check85_DP3_abba[20] P2_Check85_DP3_abba[21] P2_Check85_DP3_abba[23] P2_Check85_DP3_abba[23] P2_Check85_DP3_abba[25] P2_Check85_DP3_abba[26] P2_Check85_DP3_abba[28] P2_Check85_DP3_abba[29] P2_Check85_DP3_abba[29] P2_Check85_DP3_abba[20] P2_Check85_DP3_abba[20] P2_Check85_DP3_abba[20] P2_Check85_DP3_abba[20] P2_Check85_DP3_abba[20] P2_Check85_DP3_abba[20] P3_Check85_DP3_abba[2		Character Character Character Character Character Character Character Character Character Character Character Character Character Character Character Character Character Character	'r' 'd' 'F' 'S' 'S' 'O' 'S' 'B' 'S' 'S' '16#00	Datacode2 string				

• Wherever string data is configured in the payload its always better to add them at the end. This is because the string length can be variable and if its put in the middle it can create difficulty in mapping.

It is recommended when transferring multiple strings to add the String Length into the Payload.

10.4 Payload Data conversion in TIA Portal.

- The payload data which is of the **Floating data type** appears as an array of 4 bytes in the Siemens Profinet data. So these array of bytes must be converted into a single variable of the type **DWORD** type.
- The below section explains how these array of bytes are converted into a single **DWORD** value.

10.4.1 Converting floating values of payload data into a single Double Word.

1. In our example following payload data will appear in the Profinet Data as shown below.

"FB_CheckSBSI_DB".aData[0]	Hex	16#03	3
"FB_CheckSBSI_DB".aData[1]	Hex	16#00)
"FB_CheckSBSI_DB".aData[2]	Hex	16#00	
"FB_CheckSBSI_DB".aData[3]	Hex	16#02	2 DataCode
"FB_CheckSBSI_DB".aData[4]	Hex	16#EF	PosX
"FB_CheckSBSI_DB".aData[5]	Hex	▼ 16#BF	
"FB_CheckSBSI_DB".aData[6]	Hex	16#00	
"FB_CheckSBSI_DB".aData[7]	Hex	16#00	
"FB_CheckSBSI_DB".aData[8]	Hex	16#FC	DataCoue
"FB_CheckSBSI_DB".aData[9]	Hex	16#AC	POST
"FB_CheckSBSI_DB".aData[10]	Char	racter 'H'	
"FB_CheckSBSI_DB".aData[11]	Char	racter 'e'	
"FB_CheckSBSI_DB".aData[12]	Char	racter 'l'	
"FB_CheckSBSI_DB".aData[13]	Char	racter 'l'	
"FB_CheckSBSI_DB".aData[14]	Char	racter 'o'	
"FB_CheckSBSI_DB".aData[15]	Char	racter ''	
"FB_CheckSBSI_DB".aData[16]	Char	racter 'W	
"FB_CheckSBSI_DB".aData[17]	Char	racter 'o'	
"FB_CheckSBSI_DB".aData[18]	Char	racter 'r'	
"FB_CheckSBSI_DB".aData[19]	Char	racter 'l'	
"FB_CheckSBSI_DB".aData[20]	Char	racter 'd'	
"FB_CheckSBSI_DB".aData[21]	Char	racter 'F'	
"FB_CheckSBSI_DB".aData[22]	Char	racter 'E'	
"FB_CheckSBSI_DB".aData[23]	Char	racter 'S'	
"FB_CheckSBSI_DB".aData[24]	Char	racter 'T'	
"FB_CheckSBSI_DB".aData[25]	Char	racter 'O'	
"FB_CheckSBSI_DB".aData[26]	Char	racter ''	
"FB_CheckSBSI_DB".aData[27]	Char	racter 'S'	
"FB_CheckSBSI_DB".aData[28]	Char	racter 'B'	
"FB_CheckSBSI_DB".aData[29]	Char	racter 'S'	
"FB_CheckSBSI_DB".aData[30]	Char	racter 'l'	

- 2. Here in our example 2 data's namely **DataCode1 Pos X and DataCode1 Pos Y** are of the Floating Data type.
- 3. **DataCode1 Pos X** value appears as an array of 4 Bytes starting from aData[2] to aData[5]. So this array of bytes has to be converted into a single DWORD.
- 4. **DataCode1 Pos Y** value appears as an array of 4 Bytes starting from aData[6] to aData[9]. So this array of bytes has to be converted into a single DWORD.
- 5. Under Global variables create 2 variables namely **DataCode1.PosX** and **DataCode1.PosY** of the type **DINT**.

Data_block_1

1		Name	Data type Start value		Retain Accessible		Writa	Visible in	Setpoint
I		▼ Static	I						
I	-00	DataCode1.PosX	DInt	0			~		
		DataCode1.PosY	DInt	0			~		

- 6. The array of bytes in the payload data (aData[2] to aData[5] will be mapped to the DINT variable **DataCode1.PosX** as shown below.
 - aData[2] ---- Byte 3 of DataCode1.PosX
 - aData[3] ---- Byte 2 of DataCode1.PosX
 - aData[4] ---- Byte 1 of DataCode1.PosX
 - aData[5] ---- Byte 0 of DataCode1.PosX
- 7. The array of bytes in the payload data (aData[2] to aData[5] will be mapped to the DINT variable **DataCode1.PosX** as shown below.
 - aData[6] ---- Byte 3 of DataCode1.PosY
 - aData[7] ---- Byte 2 of DataCode1.PosY
 - aData[8] ---- Byte 1 of DataCode1.PosY
 - aData[9] ---- Byte 0 of DataCode1.PosY

8. Write a logic to do the mapping as shown in **Step 6** and **Step 7**.

Mapping of DataCode1.PosX



Mapping of DataCode1.PosX



The payload data values are multiplied by a x1000 factor. So if we have to get the value which is exact to Vision Configuration studio we have to divide by 1000.
 Write a logic for this scaling as shown below.



Application Note-SBSI Vision Sensor- Integration of Host Function Blocks in SIEMENS TIA Portal V15 Seite 52 von 54 10. Add the variables DataCode1.PosX and DataCode1.PosY to Watch table.

"Data_block_1"."DataCode1.PosX"	DEC	
"Data_block_1"."DataCode1.PosY"	DEC	-

- 11. Download the project and Go Online.
- 12. Once the Function Block is executed observe the values of the watch table to monitor the payload data.
- 13. In our example the payload data appears in the watch table as shown below.

"FB_CheckSBSI_DB".aData[0]	Hex 💌	16#06	
"FB_CheckSBSI_DB".aData[1]	 Hex	16#00	
"FB_CheckSBSI_DB".aData[2]	Hex	16#00	
"FB_CheckSBSI_DB".aData[3]	Hex	16#02	DataCode1
"FB_CheckSBSI_DB".aData[4]	Hex	16#F1	PosX
"FB_CheckSBSI_DB".aData[5]	Hex	16#18	
"FB_CheckSBSI_DB".aData[6]	Hex	16#00	
"FB_CheckSBSI_DB".aData[7]	Hex	16#00	
"FB_CheckSBSI_DB".aData[8]	Hex	16#FC	PosV
"FB_CheckSBSI_DB".aData[9]	Hex	16#57	1 031
"FB_CheckSBSI_DB".aData[10]	Character	'H'	
"FB_CheckSBSI_DB".aData[11]	Character	'e'	
"FB_CheckSBSI_DB".aData[12]	Character	Т	
"FB_CheckSBSI_DB".aData[13]	Character	T	
"FB_CheckSBSI_DB".aData[14]	Character	'o'	
"FB_CheckSBSI_DB".aData[15]	Character	11 A.	
"FB_CheckSBSI_DB".aData[16]	Character	'W'	
"FB_CheckSBSI_DB".aData[17]	Character	'o'	
"FB_CheckSBSI_DB".aData[18]	Character	1 1 1	
"FB_CheckSBSI_DB".aData[19]	Character	т	
"FB_CheckSBSI_DB".aData[20]	Character	.q.	
"FB_CheckSBSI_DB".aData[21]	Character	'F'	
"FB_CheckSBSI_DB".aData[22]	Character	'E'	
"FB_CheckSBSI_DB".aData[23]	Character	'S'	
"FB_CheckSBSI_DB".aData[24]	Character	Τ	
"FB_CheckSBSI_DB".aData[25]	Character	'0'	
"FB_CheckSBSI_DB".aData[26]	Character		
"FB_CheckSBSI_DB".aData[27]	Character	'S'	
"FB_CheckSBSI_DB".aData[28]	Character	'B'	
"FB_CheckSBSI_DB".aData[29]	Character	'S'	
"FB_CheckSBSI_DB".aData[30]	Character	T.	

14. The PosX and PosY value of Datacode1 coming as an array of bytes is grouped to a DINT variable as explained before. This value in Watch table will appear as shown below.

"Data_block_1"."DataCode1.PosX"	DEC	192792
"Data_block_1"."DataCode1.PosY"	DEC	64599

15. To get the same value as in Vision Configuration studio we have to divide the above value by 1000. The scaled values will be displayed in watch table as shown below.

"Data_block_1".PosX	Floating-point nu	192.792
"Data_block_1".PosY	Floating-point nu	64.599

16. This value matches with the PosX and PosY value obtained in Vision Configuration studio.

DataCode1.PosX											DataCode1.Pos		1.PosY						
Deco	ecoded results																		
	Decoded string	Truncated	String length	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Contrast	Decoding error	Module height	Module widu.	Position X	Position Y	
1.1	Hello World		11	n/a	n/a	56	0	6	6	192.8	64.6								
1.2	FESTO SBSI		10	n/a	n/a	45	0	6	6	667.5	87.1								