

**SBSx**  
**Short explanation of Codesys Library**



Title ..... Short explanation of Codesys library  
Version ..... 1.00  
Document no. .... 12356  
Original ..... de  
Author ..... Bernd Schmid SG-OT  
  
Last update ..... 22.11.2019

## 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](http://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](http://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.

# Content

1	Hardware/Software .....	4
1.1	Manuals .....	
2	Description of the Application note .....	6
3	Description of the Function block in SBSI-library .....	7
4	Prepare Codesys .....	8
5	Prepare the vision sensor for communication.....	9
6	Test of communication.....	13

# 1 Hardware/Software

Typ/Name	Version Software/Firmware	Herstellungsdatum
SBSI	2930236	
Vision Sensor Configuration-software	1.23.2.2	
Codesys	V3.5 SP 12	

Table 1.1: Hardware/Software

## 1.1 Available manuals

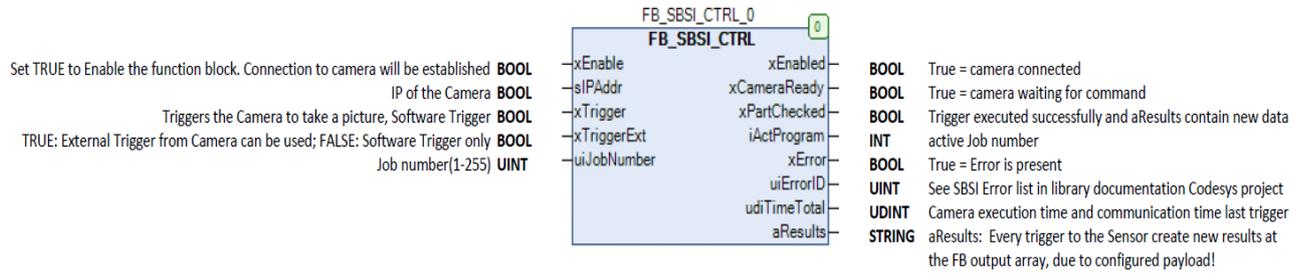
# 2 Description of the Application note

This short introduction explain how to configure a Festo vision sensor to communicate result data to a Codesys controller.

The example show to transfer the results of a barcode. The results are the x-,y-coordinates, angle and the string of the barcode.

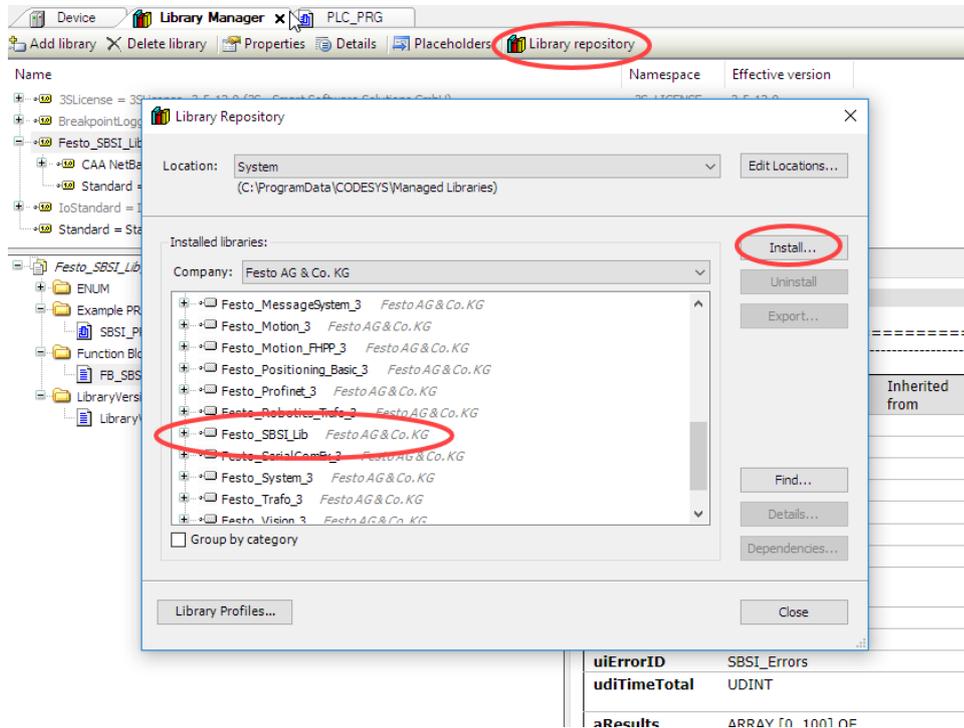
### 3 Description of the Function block in the SBSI-library

The fb support to establish TCP/IP connection, change active job no., send a trigger signal and receive result string from SBS-vision sensors. Independent from the different available model. Based on firmware 1.23.

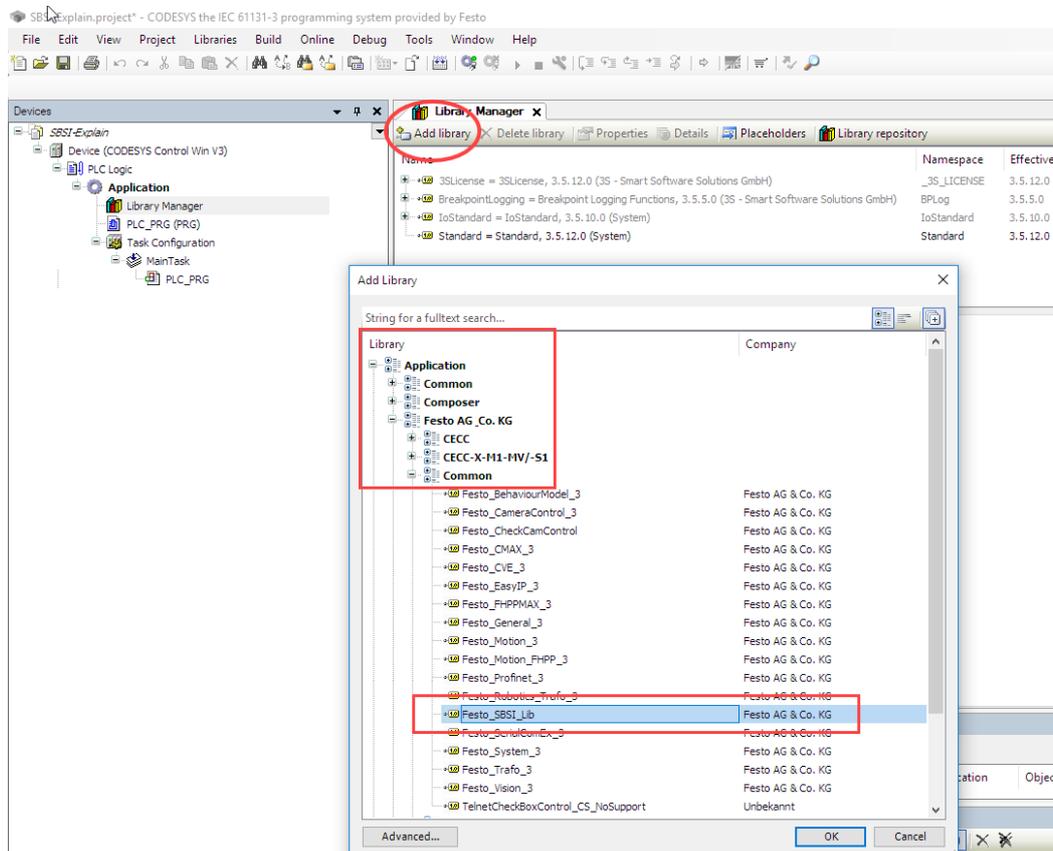


## 4 Prepare Codesys

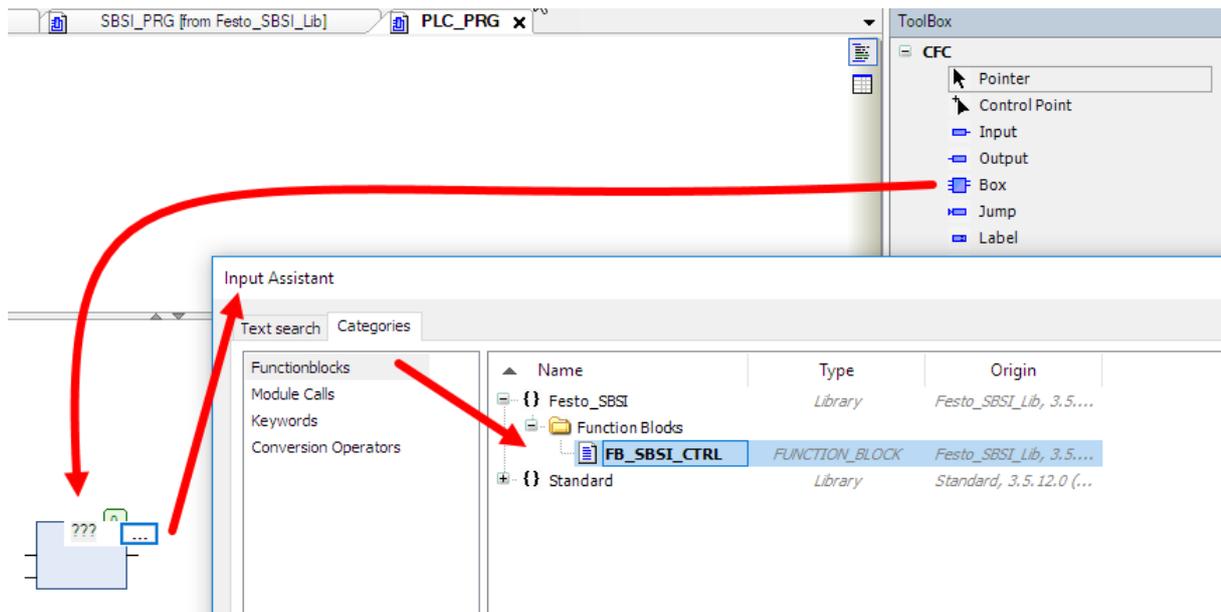
First you have to install the library to the repository of your Codesys. The Library is delivered with the Application note or in SPIMS database or by TSHQ.



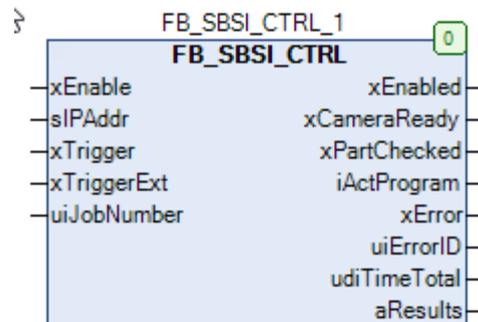
Then add the library to your project.



Then you can use the FB as a usual FB in Codesys. In CFC you can insert a box and use the input assistant with auto declaration and instance no.



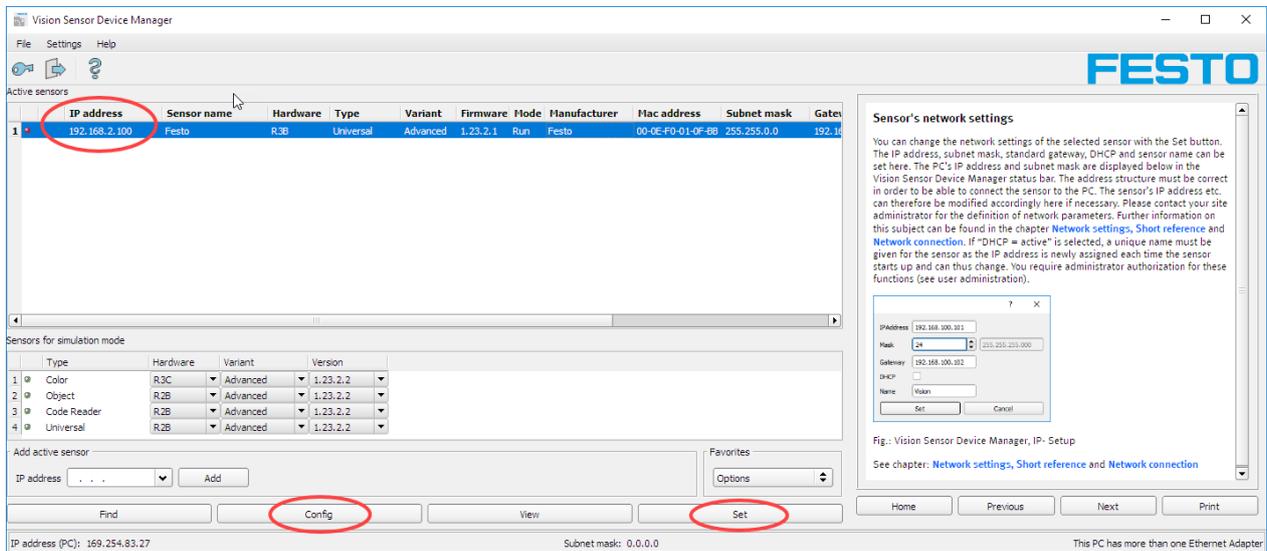
So the fb is applied:



## 5 Prepare the vision sensor for communication

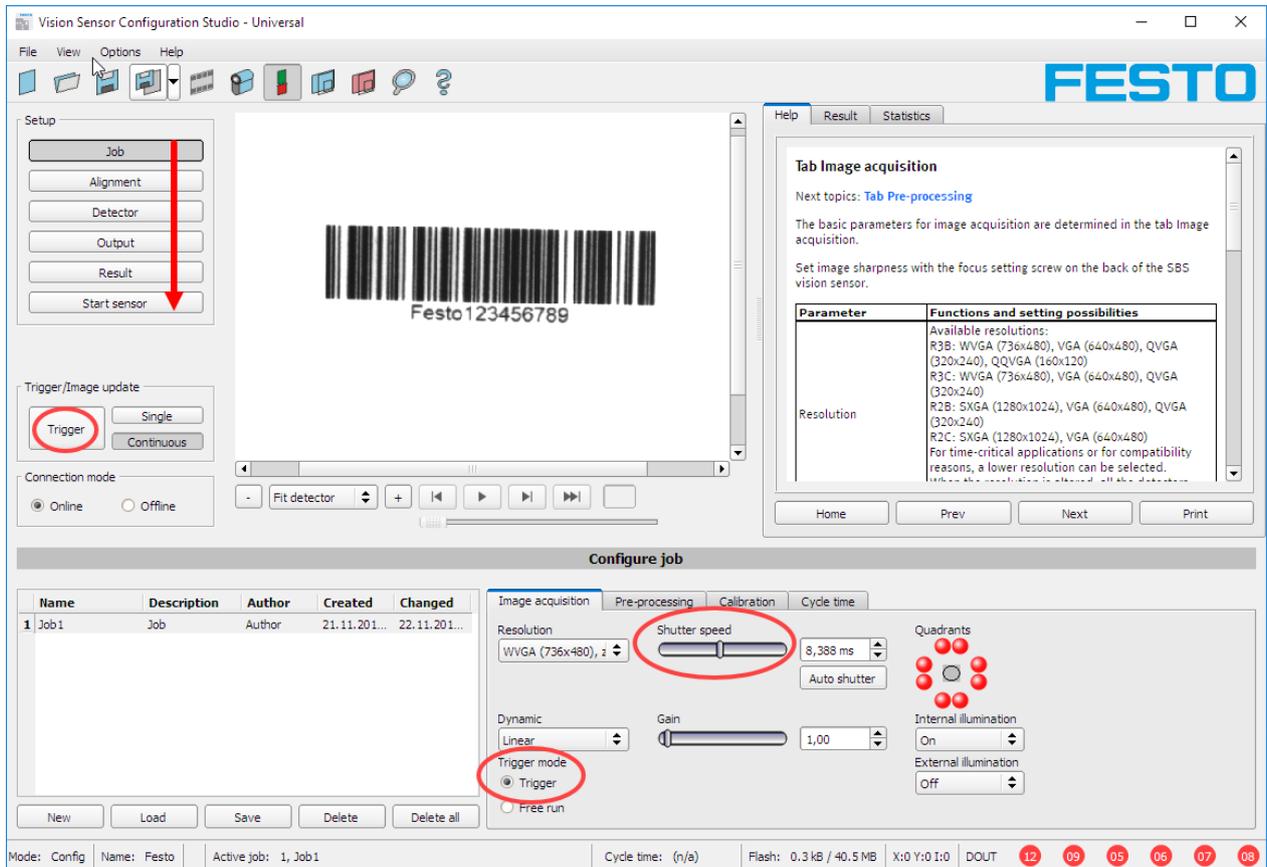
Start the vision sensor device manager.

Then the connected sensor will be shown in the list:



To change the IP-address press the **Set** button. If not necessary continue with the **Config** button.

After that, the Vision Sensor Configuration Studio will be started.

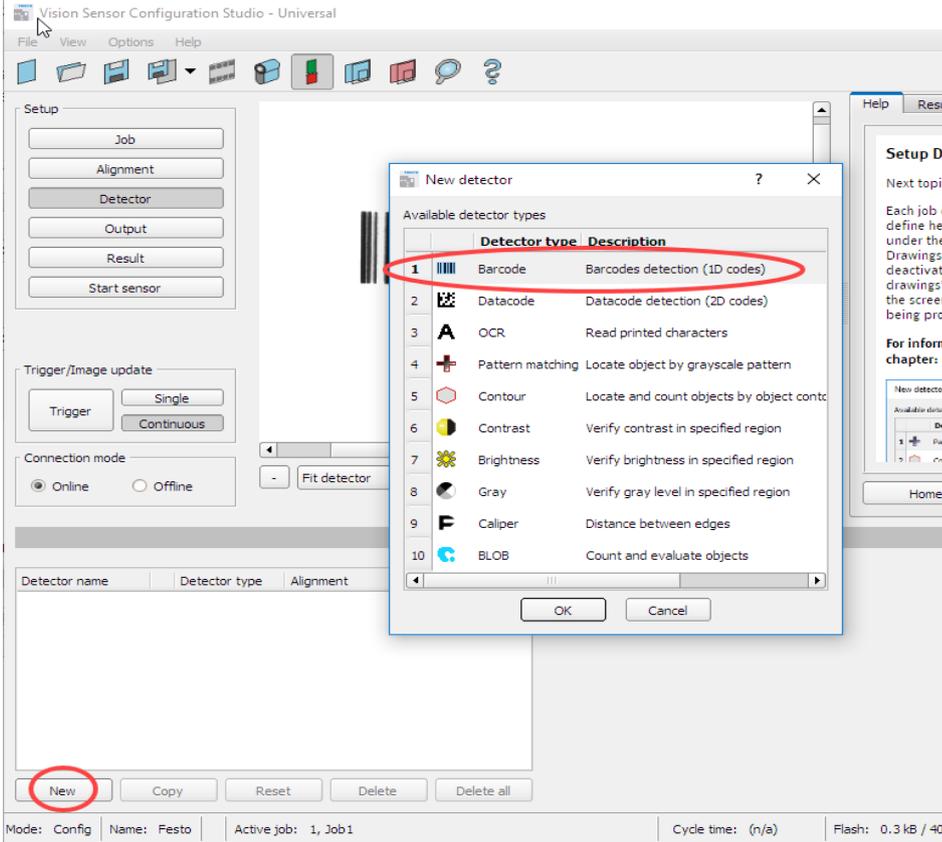


Follow the steps one after the other.

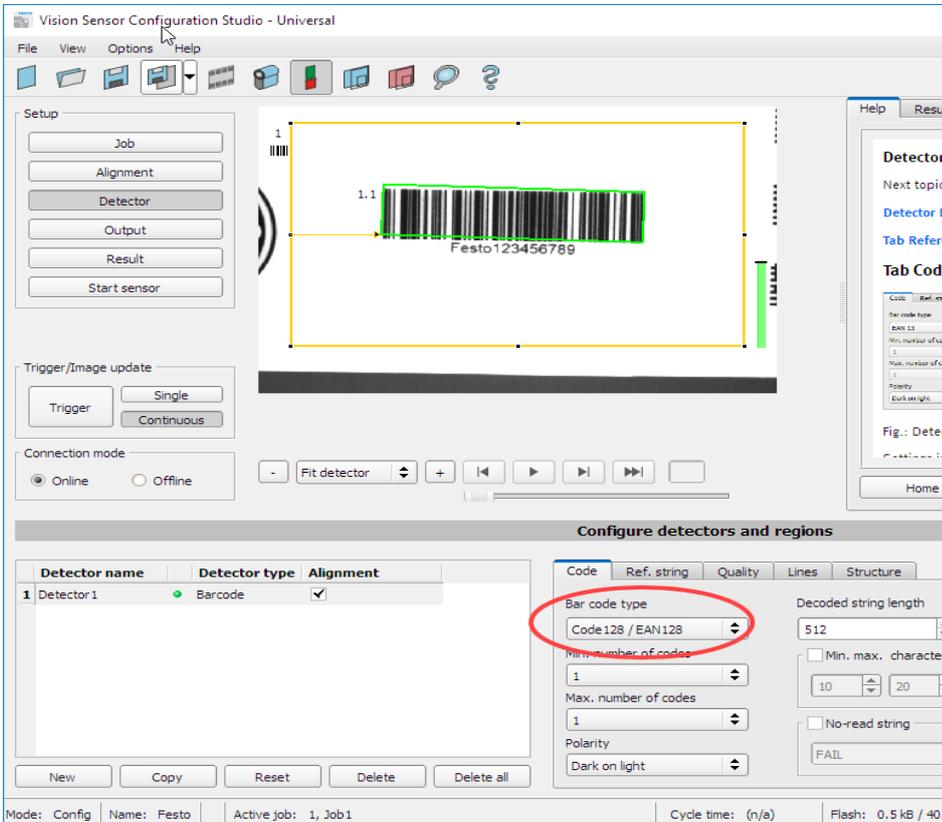
In the Job you adjust the brightness by the shutter speed. Set up the image that you get a good contrast of the code. Set the trigger mode to Trigger. This will be necessary for the data transfer.

With the Trigger button you can initiate a new image.

Next step Detector. (Alignment is not used for this example)

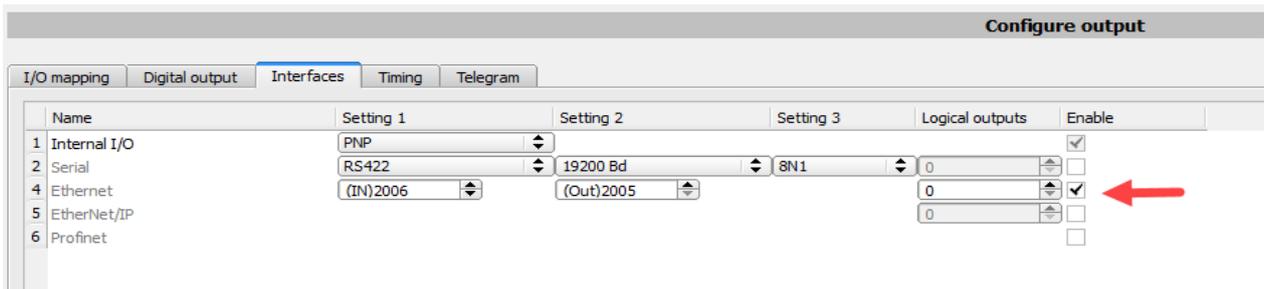


Select the correct code type:



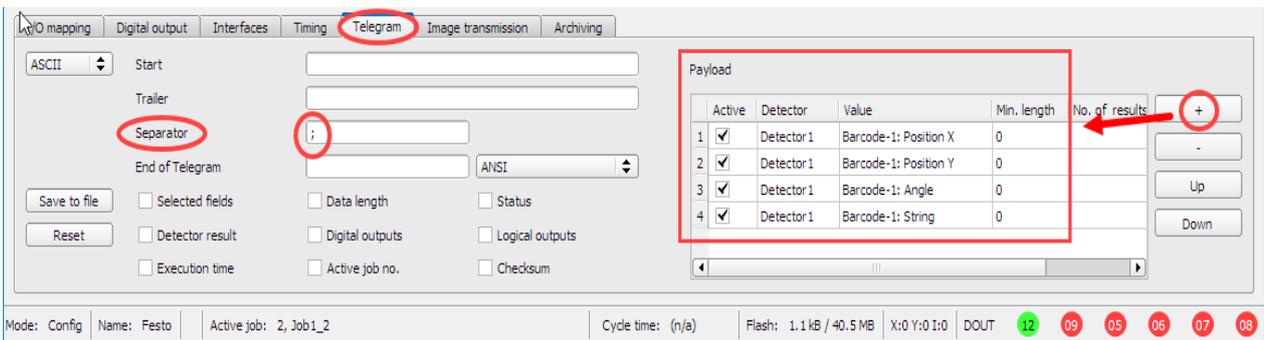
If the code is readable, it will be bounded with a green box.

Next step: Output > tab Interfaces



Enable the Ethernet interface.

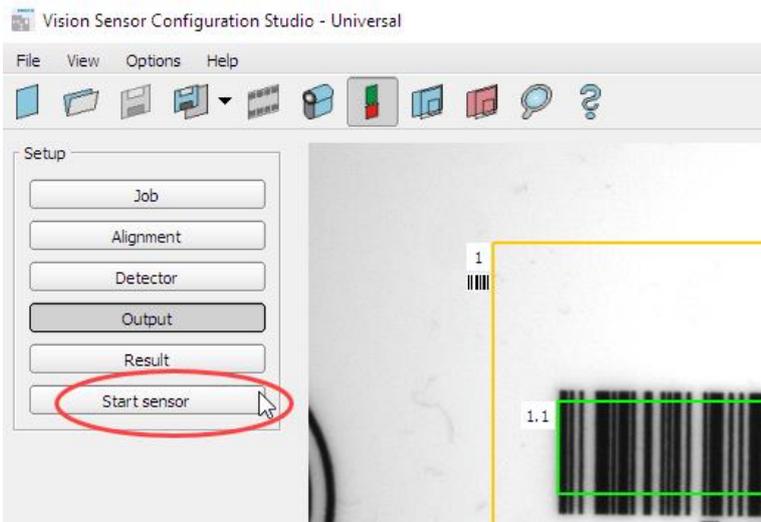
Change to tab Telegram



The FB is made for max. 100 result string. In this demo we transfer 4 values. Position coordinate of the Bar code x, y, angle and the detected string. Important is to insert the semicolon as a separator sign. By clicking the + it is possible to add more entry in the payload list.

Finish the project.

To store and activate the setting in the sensor you have to press "START sensor"



After a trigger by button in software, the following result will be calculated:

The screenshot displays the Vision Sensor Configuration Studio - Universal software interface. The main window shows a camera view of a barcode with the text "Festo123456789" below it. A yellow bounding box is drawn around the barcode, and a green bounding box is drawn around the text. The software is in the "Results/statistics" view. The "Trigger" button is circled in red. The "Decoded results" table shows the following data:

Detector	Score	Time	Detector type
1   Detector1	100.0	12ms	Barcode

Q9	Q10	Q11	Q12	Q13	Q14	Q15	Q16	Q17	Q18	Q19	Q20	Q21	Q22	Q23	Q24	Position X	Position Y	Angle
1.1	n/a	354.0	207.5	0.0														

Count	Pass	Fail	Minimum execution time	Maximum execution time	Average execution time
5	5	0	35ms	36ms	36ms

The "Statistics" panel on the right shows the following values:

- Count: 5
- Pass: 5 (100.00%)
- Fail: 0 (0.00%)
- Minimum execution time: 35ms
- Maximum execution time: 36ms
- Average execution time: 36ms

The status bar at the bottom shows: Mode: Run | Name: Festo | Active job: 1, Job 1 | Cycle time: 36 ms | Flash: 0.6 kB / 40.5 MB | X:0 Y:0 I:0 | DOUT: 10 09 05 06 07 08

Now the sensor is prepared for the next step and connection to the Codesys controller.

## 6 Test of communication

In the example project the Codesys Control Win V3 is used. But any other Festo Codesys controller should work similar.

Login to the controller and download the project to the controller. Start the PLC.

Force (ctrl+F7) the ip address of the sensor and then force "True" to the xEnable input. After that the connection will be established, and if all is o.k, the FB set the output xEnabled and xCameraReady.

If not then the Error output occur. Error Id could be checked in the library documentation in Codesys.

Expression	Type	Value	Prepared value
FB_SBSI_CTRL_1	FB_SBSI_CTRL		
xEnable	BOOL	TRUE	
sIPAddr	STRING	"192.168.2.100"	
xTrigger	BOOL	FALSE	
xTriggerExt	BOOL	FALSE	
uJobNumber	INT	0	
xEnabled	BOOL	TRUE	
xCameraReady	BOOL	TRUE	
xPartChecked	BOOL	FALSE	
iActProgram	INT	0	
xError	BOOL	FALSE	
uErrorID	SBSI_ERRORS	NO_ERROR	
uTimeTotal	UDINT	0	
aResults	ARRAY [0..100] OF ...		
iSTEP	INT	30	

Now the FB is ready for trigger input.

Force True to xTrigger.

After the image processing time the output xPartChecked will become true. Then the results are available in the aResult array.

Expression	Type	Value	Prepared value	A...	Comment
FB_SBSI_CTRL_0	FB_SBSI_CTRL				
FB_SBSI_CTRL_1	FB_SBSI_CTRL				
xEnable	BOOL	TRUE			TRUE = Enable Function Block
sIPAddr	STRING	"192.168.2.100"			IP address of the SBSI Vision Sensor
xTrigger	BOOL	TRUE			Rising Edge: Trigger SBSI
xTriggerExt	BOOL	FALSE			
uJobNumber	INT	0			Set SBSI active Job 1 - 255
xEnabled	BOOL	TRUE			TRUE = Camera connected
xCameraReady	BOOL	TRUE			TRUE = Camera waiting for Command
xPartChecked	BOOL	TRUE			TRUE = Trigger executed ... cessfully and aResults c.
xError	BOOL	FALSE			Active Camera Program.
iActProgram	INT	0			TRUE = Error is present
xError	BOOL	FALSE			See SBSI_Errors
uErrorID	SBSI_ERRORS	NO_ERROR			Camera Execution time + ... IP communication time.
uTimeTotal	UDINT	42			Array of results configure... the Output Tab of Visio.
aResults	ARRAY [0..100] OF ...				
aResults[0]	STRING	"382000"			
aResults[1]	STRING	"233502"			
aResults[2]	STRING	"-8142"			
aResults[3]	STRING	"Festo123456789"			
aResults[4]	STRING				

The sensor transmit the results multiplied by 1000! So for further processing in Codesys the values have to be divided by 1000!

Compare the result in sensor software

Detector	Score	Time	Detector type	Decoded results													Position X	Position Y	Angle	Com	
1	100.0	14ms	Barcode	Q11	Q12	Q13	Q14	Q15	Q16	Q17	Q18	Q19	Q20	Q21	Q22	Q23	Q24	382.0	233.5	-8.1	

If result is not possible or calculated. The String is empty and the value is 0.

Vision Sensor Configuration Studio - Universal

File View Options Help

Setup

- Job
- Alignment
- Detector
- Output
- Result
- Stop sensor

Trigger/Image update

Trigger

Connection mode

Online  Offline

100%

Results/statistics

Detector	Score	Time	Detector type	Decoded results													Position X	Position Y	Angle	Com		
1	0.0	48ms	Barcode	Q10	Q11	Q12	Q13	Q14	Q15	Q16	Q17	Q18	Q19	Q20	Q21	Q22	Q23	Q24				

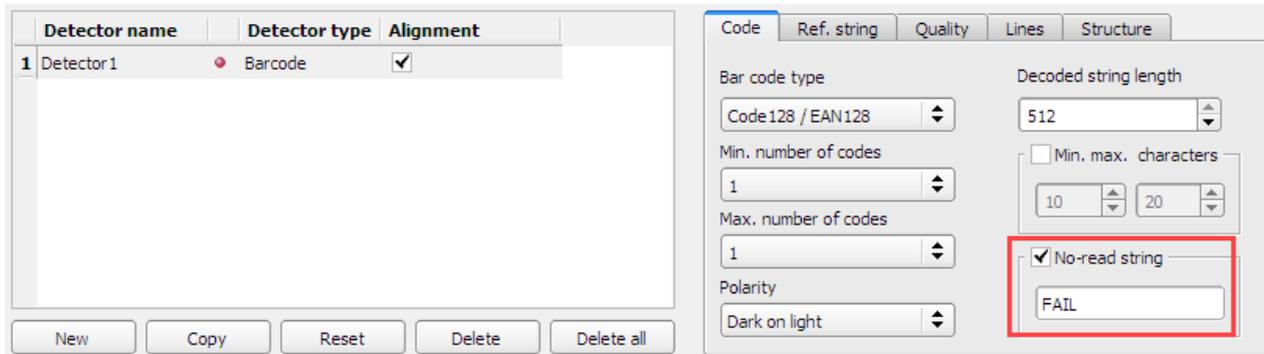
LibraryVersion [from Festo\_SBSI\_Lib] Library Manager SBSI\_PRG [from Festo\_SBSI\_Lib] PLC\_PRG x Device

Device.Application.PLC\_PRG

Expression	Type	Value	Prepa
FB_SBSI_CTRL_0	FB_SBSI_CTRL		
FB_SBSI_CTRL_1	FB_SBSI_CTRL		
xEnable	BOOL	TRUE	
sIPAddr	STRING	'192.168.2.100'	
xTrigger	BOOL	TRUE	
xTriggerExt	BOOL	FALSE	
uiJobNumber	INT	0	
xEnabled	BOOL	TRUE	
xCameraReady	BOOL	TRUE	
xPartChecked	BOOL	TRUE	
iActProgram	INT	0	
xError	BOOL	FALSE	
uiErrorID	SBSI_ERRORS	NO_ERROR	
udiTimeTotal	UDINT	60	
aResults	ARRAY [0..100] OF ...		
aResults[0]	STRING	'0'	
aResults[1]	STRING	'0'	
aResults[2]	STRING	'0'	
aResults[3]	STRING	'0'	
aResults[4]	STRING	'0'	
aResults[5]	STRING	'0'	

To get also a string for a "no read", it is possible to give an own text.

Therefor adjust in the detector setting a text for a no read result. Do not forget to start the sensor when you leave the config software!



To test again in Codesys, you have to disable the fb and enable again.

**Remark:** The Config software of the sensor interrupt existing TCP/IP connection to codesys controller.

In the result array the given text is shown:

ARRAY [0..100] OF ...	
STRING	'0'
STRING	'0'
STRING	'0'
STRING	'FAIL'
STRING	"
STRING	"

End of document