PERFECTION IN AUTOMATION
A MEMBER OF THE ABB GROUP

Vision Demo Application

Version 5.26
Date: 7/6/2024

Contents

Copyright © B&R - Änderungen vorbehalten

1 Introduction

This sample application can be used as a starting point for vision applications. Google Chrome browser must be used for the mappView project; other browser may not show the correct crosshair position.

1.1 System requirements

This sample was developed and tested with Automation Studio 4.12

- PLC OS system I4.93 or higher
- mappView 5.26
- Hardware files for camera (VSS112Q22.041P-000 was used in this sample)
- Chrome Browser

2 Project Files

The following project files are vision sensor related.

2.1 Logical View

All tasks starting with Vi_ should **not** be changed to make updating easier.

Vision	Package with vision tasks
Vi_main	This task handles functions that are sensor related
Vi_visu	This is a helper task for the visualization
Vi_light	This is a helper task for the backlights and bar lights
Vi_nettime	This task handles the nettime calculation
Vi_image	This task handles everything that is image related
Axis	Sample task for an axis, needed for nettime handling
YourTask	Customer specific task
setRouteToCamera	Make sure to adjust the IP address in the file “\Vision_1\Logical\Vision\setRouteToCamera.bat” and execute the batch file in Windows with right click (Run as administrator). Otherwise, the sensor image does not work in the Vision Cockpit or the demo visualization.
mappView	mappView demo visualization for vision
mappRecipe	Stores the camera configuration

2.2 Configuration View

mappView	mappView visualization for vision
mappVision	mappVision configuration for vision functions
mappService	Configuration for recipe management

2.3 Physical View

Blob	Sensor for the blob function. Powerlink Node 1.
Measurement	Sensor for the edge measurement function. Powerlink Node 2.
CodeRead	Sensor for the code reader function. Powerlink Node 3.
Match	Sensor for the match function. Powerlink Node 4.
OCR	Sensor for the text recognition function. Powerlink Node 5.
PixelCounter	Sensor for the text pixel counter function. Powerlink Node 6.

This is an example configuration with one camera for each vision function. You can quickly switch between the different vision functions by changing the node number.

3 Constants

The project provides several constants to adjust the configuration.

Name	Default	Description
MAX_NUM_CAMS	6	Maximum number of camera's
MAX_NUM_LIGHTS	5	Maximum number of light's
MAX_NUM_RESULTS	10	Maximum number of results
MAX_NUM_PRODUCTS	8	Maximum number of products for color detection
MAX_IDX_VA_LIST	19	Maximum number of entries in the vision application list
DEVICE_NAME	VisionFileDevice	File device where the images and recipes are stored
NETTIME_DEFAULT_DELAY	30000	This delay is used when software trigger is used and nettime is enabled
NETTIME_ENABLE	FALSE	Enable nettime code, this generates the nettime when using manual triggers
MAX_NUM_CODETYPES	69	Maximum number of code types
CodeTypes		Codes types for code reading

4 Parameter structures

The sample supports multiple cameras but only one is displayed at a time. The global structures begin with a “g” for (gVisionSensor, gVisionLight, gBlob, gMT, gCodeReader, gMatch, gOCR). The global structures are arrays where the index represents the camera index. The variable “gSelectedSensor” and “gSelectedLight” map one of the global structures to the dynamic local variable in the task “Vi_visu”. If the task “Vi_visu” is not used the global select variables must be set manually.

4.1 Vision sensor structure (gVisionSensor)

The vision structure handles all functions and parameters that are sensor related and are independent from the vision function used.

CMD		Command structure to trigger an action
	ImageTrigger	Start a new image aquisition
	ImageTriggerReset	Abort future image acquisition (with TriggerDelay)
	AutoSetupStartStop	Start and stop automatic camera setup
CFG		Sensor configuration
	VisionFunction	The type of vision function that is used with the sensor. This information is used to add the correct detail information on the main page.
	PowerlinkNode	The Powerlink node number for this camera. This information is used to generate the correct IP address for the camera.
	DataStructure	Pointer to the vision speccific function. (gBlob, gMT, gCodeReader, gMatch, gOCR)
	ComponentLink	Vision component name defined under mappVision in the configuration view
	…	All other parameters, see camera manual for details
DAT		Sensor data, see manual for details
HW		Sensor hardware information, see manual for details

4.2 Functional structure

Each vision function has its own structure (gBlob, gMT, gCodeReader, gMatch, gOCR).

4.3 Vision light structure (gVisionLight)

The light structure handles all functions and parameters that are light related..

CMD		Command structure to trigger an action
	FlashTrigger	Trigger flash light
	FlashTriggerReset	Abort flash light
CFG		Sensor configuration
	LightType	The type of light used (None, Backlight, Lightbar).
	PowerlinkNode	The Powerlink node number for this light.
	…	All other parameters, see light manual for details
DAT		Light data, see manual for details
HW		Light hardware information, see manual for details

4.4 Vision image structure

The image structure handles all functions and parameters that are related to the image archive and crosshair drawing in mappView.

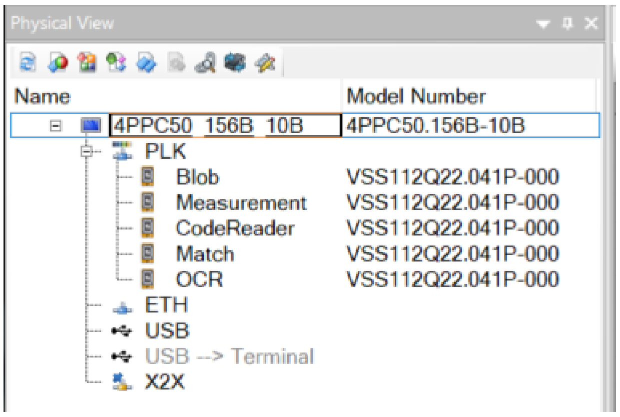
CMD		Command structure to trigger an action
	CreateDir	Creates an empty folder for images
	DeleteDir	Deletes the complete folder with all images
	DeleteImage	Delete selected image
	GetImage	Get image from camera
	SaveImage	Upload an image from the sensor and store it on the flash card
	Refresh	Reload image list from flash card
	ResetError	Acknowledge
CFG		Image Archiv configuration
	FileDevice	File device name where the images are stored
	FileDevice	File name for save image command
	DirName	Name of an automatically created folder on the FileDevice
	ComponentLink	Vision component name defined under mappVision in the configuration view

with data will be embedded in the new SVG file

DAT		Sensor status information, see manual for details
	ImageList	Images list as data provider for connection to mappView
	ImageSelected	Selected image from list
	ImagePath	Full path to selected image
	ImageLast	Full path to last image
	Status	Status of the image operation
	Croshair	Crosshairdata, will be copied from VisionMain

5 Description

5.1 Hardware configuration



Name	Model Number
4PPC50_156B_10B	4PPC50.156B-10B
PLK	
Blob	VSS112Q22.041P-000
Measurement	VSS112Q22.041P-000
CodeReader	VSS112Q22.041P-000
Match	VSS112Q22.041P-000
OCR	VSS112Q22.041P-000
ETH	
USB	
USB --> Terminal	
X2X	

The sensor used in this sample is VSS112Q22.041P-000. If this is not the sensor available right click on the hardware and choose “Replace Hardware Module” to select the correct hardware.

In the demo application, each sensor represents one vision function. By changing the node number, it is possible to quickly switch between different functions.

The hardware configuration uses the following Powerlink node numbers:

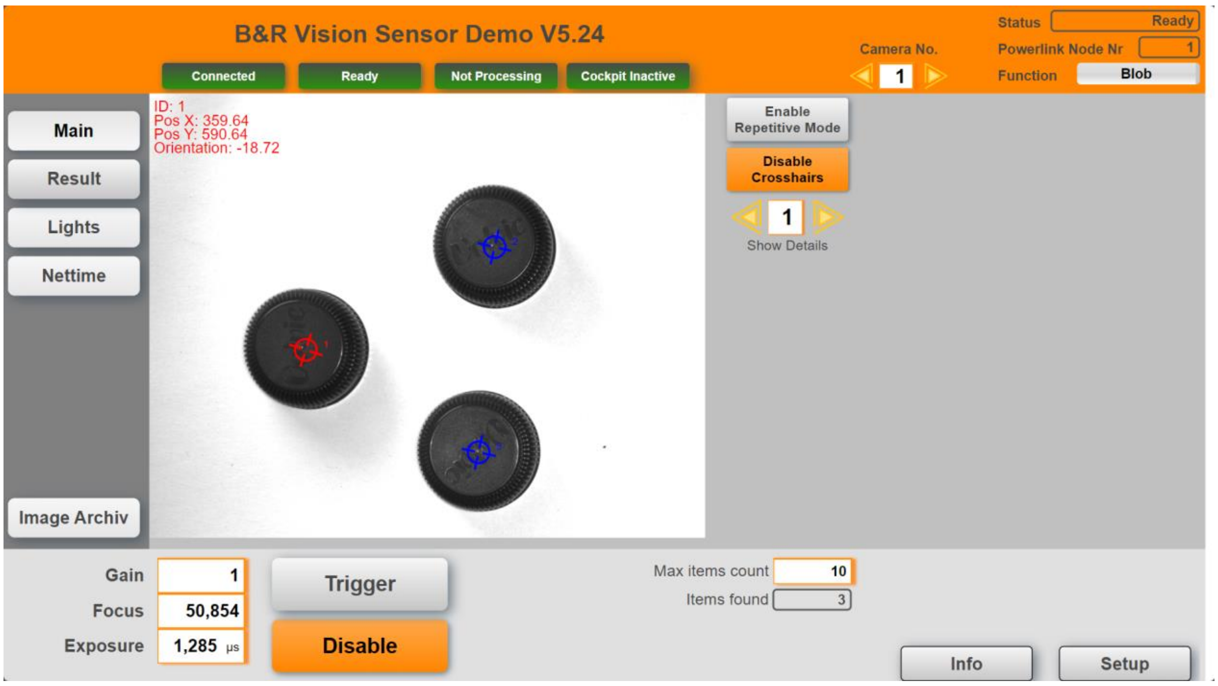
1: Blob
2: Measurement
3: Code Reader
4: Match
5: OCR
6: PixelCounter

5.2 Camera image

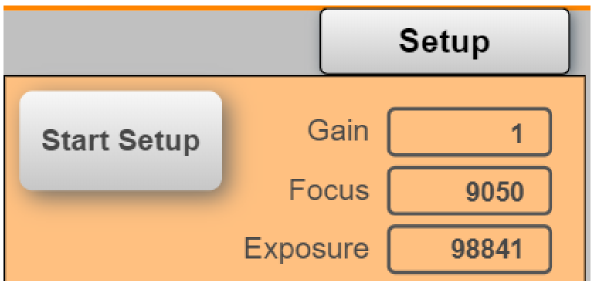
The mappView visualization shows the image of the camera on the main page. This is done through the function block ViBaseGetImage.

5.3 Demo application

The demo application consists of multiple pages to demonstrate the vision function. The main page is used to set up the sensor image. The bottom window shows the most important parameters and status information. The first step is to make sure that the sensor is connected and ready. All four elements at the top should be green.



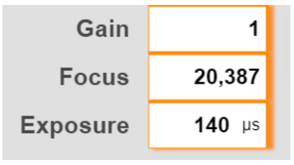
To start the auto setup process click on Setup on the bottom right corner. Click on “Start Setup” to initiate the auto setup that determines the values for gain, focus and exposure.



The sensor light should flash for about 20 seconds.

If the object is not aligned correctly use the Repetitive Mode to make continues images and allign the object.

Click on Trigger to generate a new image. In some cases it may be necessary to adjust the automatically generated values.



Use the crosshair toggle button to show additional information. Images are stored automatically when the checkbox “Auto Archiv Image” is set (see 0). Details results can also be viewed on the “Results” page. Google Chrome browser must be used for the mappView project; other browser may not show the correct crosshair position.

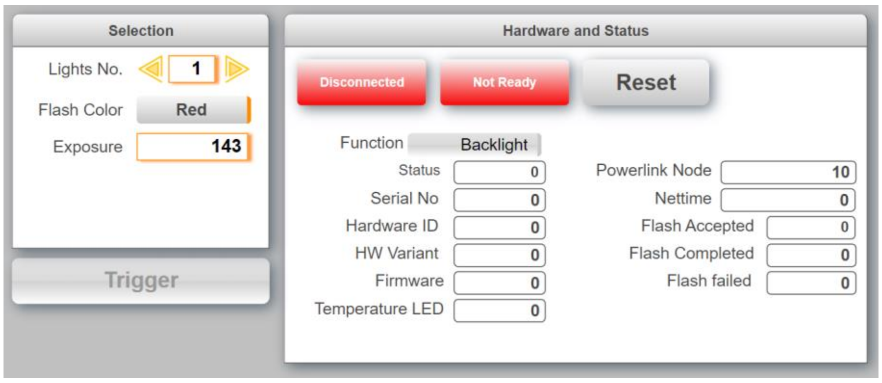
5.4 Changing the demo

The different tasks are designed to easily integrate into an existing application. All interactions are handle by variable structures. To allow easy update of these task they should not be changed. The demo includes a task “YourTask” that can be used to write own code.

The task includes the sample configuration the different vision functions and lights. It also includes a basic state machine to connect a drive and use the nettime functionality.

5.5 Lights

On the lights page the backlights or lightbars can be tested.



Selection

Lights No.	1
Flash Color	Red
Exposure	143

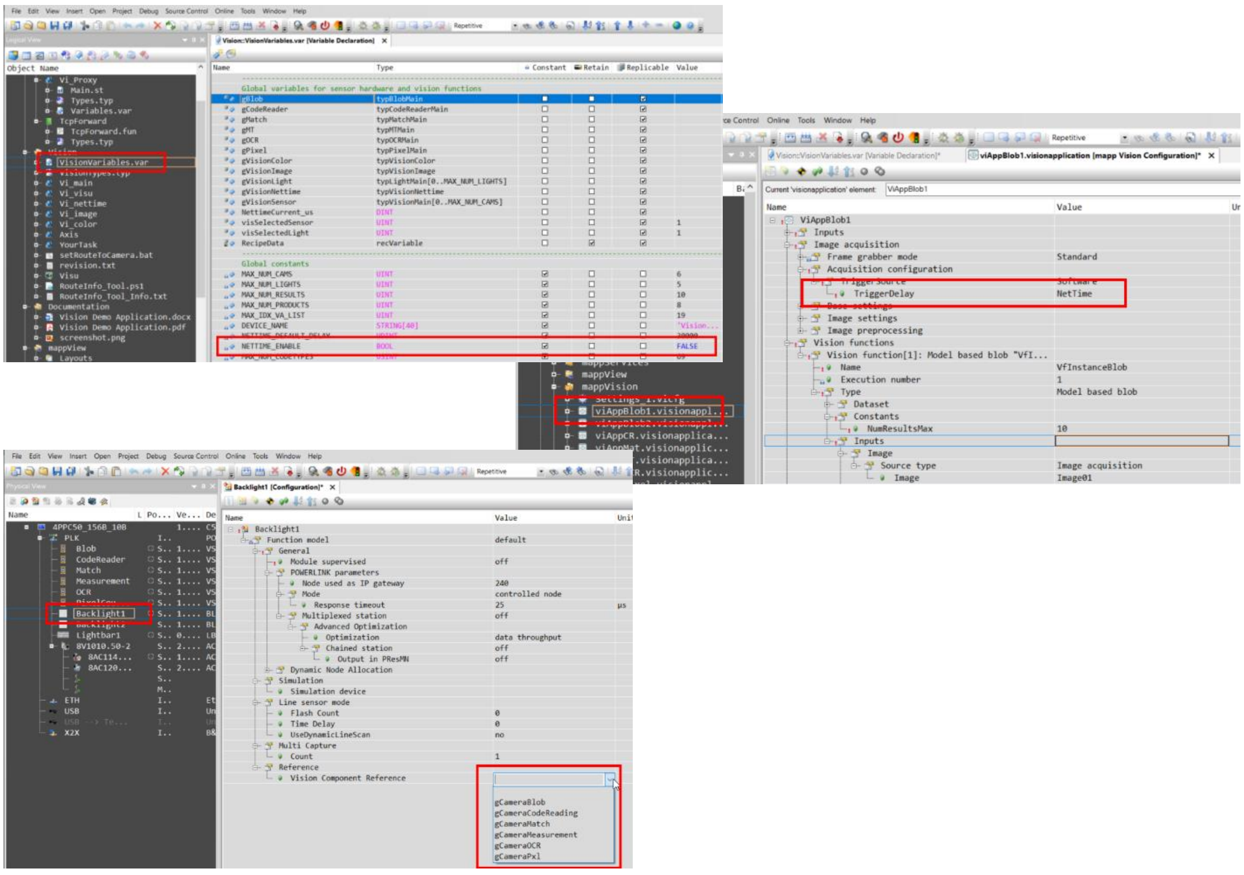
Trigger

Hardware and Status

Disconnected · Not Ready · Reset

Function	Backlight		
Status	0	Powerlink Node	10
Serial No	0	Nettime	0
Hardware ID	0	Flash Accepted	0
HW Variant	0	Flash Completed	0
Firmware	0	Flash failed	0
Temperature LED	0		

When using external lights make sure to set the constant NETTIME_ENABLE to TRUE, change the TriggerDelay to Nettime in the VisionApplication and connect the vision component under the light hardware configuration.



Vision::VisionVariables.var [Variable Declaration]

Name	Type	Constant	Retain	Replicable	Value
Global variables for sensor hardware and vision functions					
gBlob	typBlobMain	☐	☐	☑	
gCodeReader	typCodeReaderMain	☐	☐	☑	
gMatch	typMatchMain	☐	☐	☑	
gMT	typMTMain	☐	☐	☑	
gOCR	typOCRMain	☐	☐	☑	
gPixel	typPixelMain	☐	☐	☑	
gVisionColor	typVisionColor	☐	☐	☑	
gVisionImage	typVisionImage	☐	☐	☑	
gVisionLight	typLightMain[0..MAX_NUM_LIGHTS]	☐	☐	☑	
gVisionNettime	typVisionNettime	☐	☐	☑	
gVisionSensor	typVisionMain[0..MAX_NUM_CAMS]	☐	☐	☑	
NettimeCurrent_us	DINT	☐	☐	☑	
visSelectedSensor	UINT	☐	☐	☑	1
visSelectedLight	UINT	☐	☐	☑	1
RecipeData	recVariable	☐	☑	☑	
Global constants					
MAX_NUM_CAMS	UINT	☑	☐	☐	6
MAX_NUM_LIGHTS	UINT	☑	☐	☐	5
MAX_NUM_RESULTS	UINT	☑	☐	☐	10
MAX_NUM_PRODUCTS	UINT	☑	☐	☐	8
MAX_IDX_VA_LIST	UINT	☑	☐	☐	19
DEVICE_NAME	STRING[40]	☑	☐	☐	'Vision...
NETTIME_DEFAULT_DELAY	UDINT	☑	☐	☐	20000
NETTIME_ENABLE	BOOL	☑	☐	☐	FALSE

viAppBlob1.visionapplication [mapp Vision Configuration]

Current 'visionapplication' element: ViAppBlob1

Name	Value
ViAppBlob1	
Inputs	
Image acquisition	
Frame grabber mode	Standard
Acquisition configuration	
TriggerSource	Software
TriggerDelay	NetTime
Base settings	
Image settings	
Image preprocessing	
Vision functions	
Vision function[1]: Model based blob "VfI...	
Name	VfInstanceBlob
Execution number	1
Type	Model based blob
Dataset	
Constants	
NumResultsMax	10
Inputs	
Image	
Source type	Image acquisition
Image	Image01

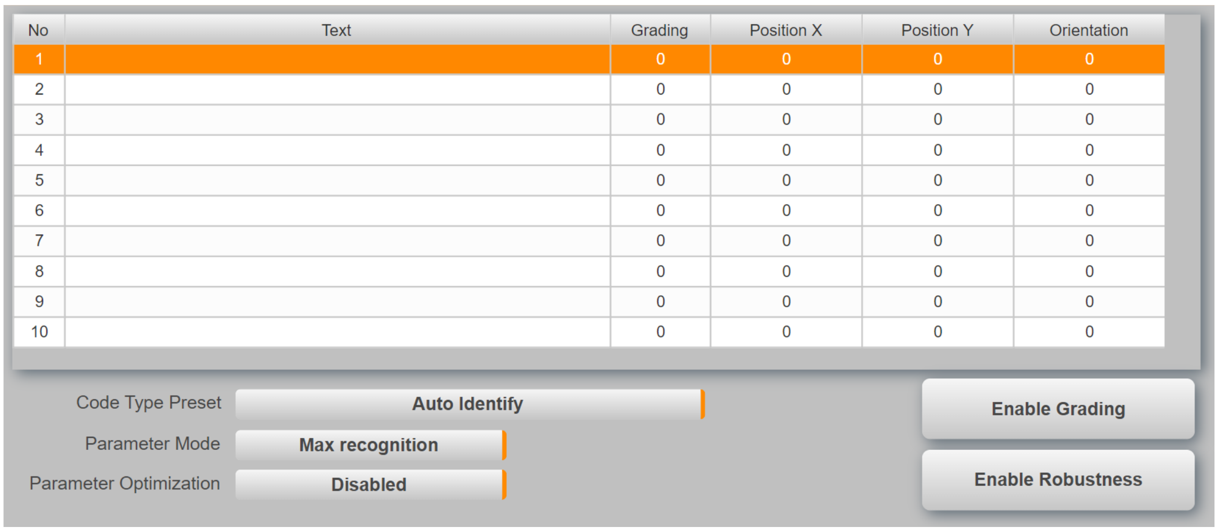
Backlight1 [Configuration]

Name	Value	Unit
Backlight1		
Function model	default	
General		
Module supervised	off	
POWERLINK parameters		
Node used as IP gateway	240	
Mode	controlled node	
Response timeout	25	µs
Multiplexed station	off	
Advanced Optimization		
Optimization	data throughput	
Chained station	off	
Output in PResMN	off	
Dynamic Node Allocation		
Simulation		
Simulation device		
Line sensor mode		
Flash Count	0	
Time Delay	0	
UseDynamicLineScan	no	
Multi Capture		
Count	1	
Reference		
Vision Component Reference		

gCameraBlob
gCameraCodeReading
gCameraMatch
gCameraMeasurement
gCameraOCR
gCameraPxl

5.6 Code Reader

The code reader page provides the information that are specific for the code reader functions. Select the code type from the drop down menu or use “Auto Identify” to start the process that tries to identify the code automatically. The identification process can run for up to 20s.



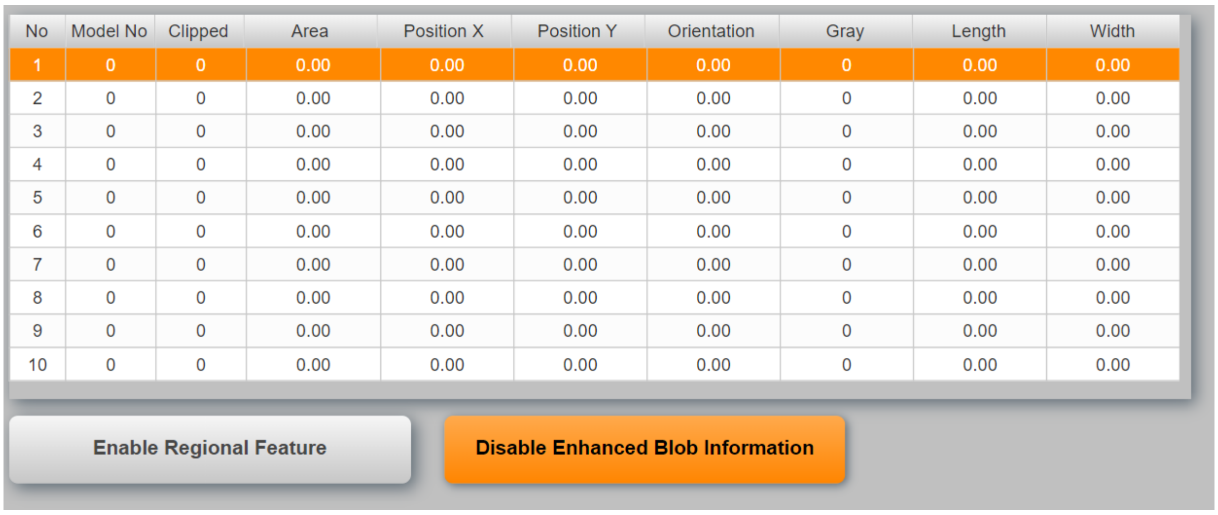
No	Text	Grading	Position X	Position Y	Orientation
1		0	0	0	0
2		0	0	0	0
3		0	0	0	0
4		0	0	0	0
5		0	0	0	0
6		0	0	0	0
7		0	0	0	0
8		0	0	0	0
9		0	0	0	0
10		0	0	0	0

Code Type Preset	Auto Identify
Parameter Mode	Max recognition
Parameter Optimization	Disabled

It is possible to read multiple codes at the same time but all codes must be of the same code type.

5.7 Blob

The blob page provides the information that are specific for the blob functions. The table shows the details for each blob that was detected by the sensor. Teaching must be done in the Vision Cockpit.

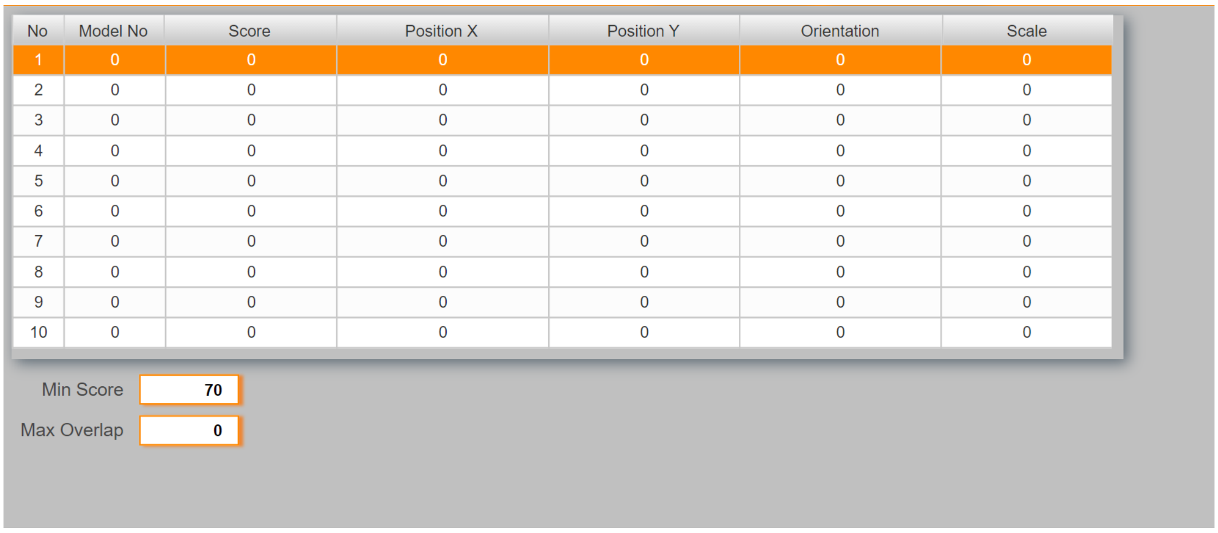


No	Model No	Clipped	Area	Position X	Position Y	Orientation	Gray	Length	Width
1	0	0	0.00	0.00	0.00	0.00	0	0.00	0.00
2	0	0	0.00	0.00	0.00	0.00	0	0.00	0.00
3	0	0	0.00	0.00	0.00	0.00	0	0.00	0.00
4	0	0	0.00	0.00	0.00	0.00	0	0.00	0.00
5	0	0	0.00	0.00	0.00	0.00	0	0.00	0.00
6	0	0	0.00	0.00	0.00	0.00	0	0.00	0.00
7	0	0	0.00	0.00	0.00	0.00	0	0.00	0.00
8	0	0	0.00	0.00	0.00	0.00	0	0.00	0.00
9	0	0	0.00	0.00	0.00	0.00	0	0.00	0.00
10	0	0	0.00	0.00	0.00	0.00	0	0.00	0.00

The blob sends only basic information like Position, Model No by default even if more data is configured. To receive all data the parameter “Enhanced Blob Information” must be enabled.

5.8 Match

The match page provides the information that are specific for the match functions. The table shows the details for each item that was detected by the sensor. Teaching must be done in the Vision Cockpit.



No	Model No	Score	Position X	Position Y	Orientation	Scale
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	0	0	0	0
10	0	0	0	0	0	0

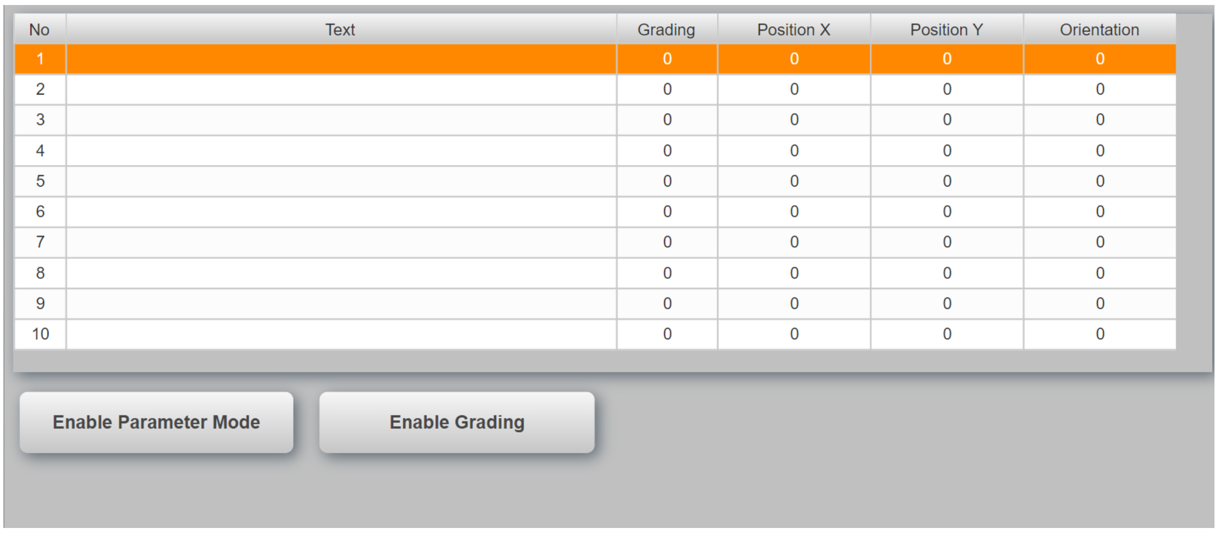
Min Score 70

Max Overlap 0

Use the value “Min Score” to adjust the detection rate. A lower value is more tolerant but can also cause fault detections.

5.9 OCR

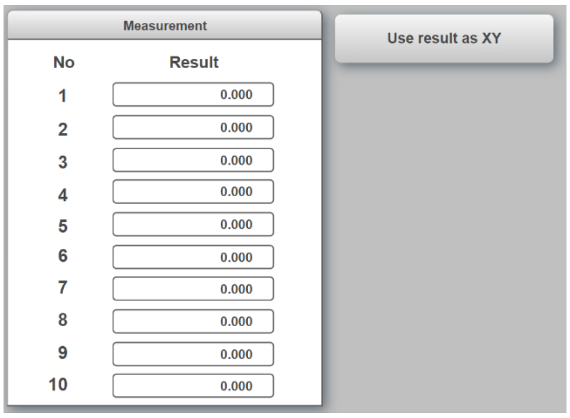
The OCR page provides the information that are specific for the OCR functions. The table shows the details for each text that was detected by the sensor.



No	Text	Grading	Position X	Position Y	Orientation
1		0	0	0	0
2		0	0	0	0
3		0	0	0	0
4		0	0	0	0
5		0	0	0	0
6		0	0	0	0
7		0	0	0	0
8		0	0	0	0
9		0	0	0	0
10		0	0	0	0

5.10 Measurement

The measurement page provides the information that are specific for the edge measurement functions. This page shows the results for the different measurement functions. What is measured must be configured in the vision cockpit.



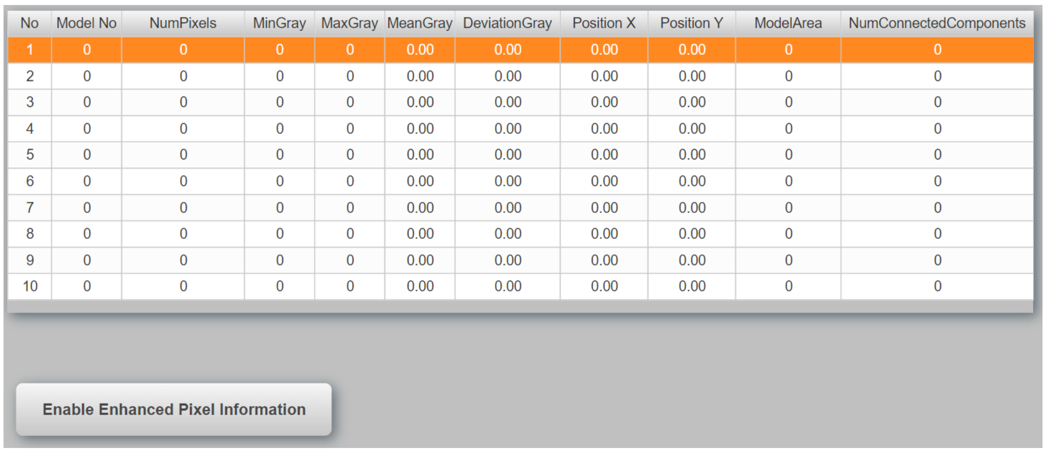
Measurement

No	Result
1	0.000
2	0.000
3	0.000
4	0.000
5	0.000
6	0.000
7	0.000
8	0.000
9	0.000
10	0.000

For edge detection it can be helpful to also draw crosshairs at the position where the edge was found. This can be enabled with the toggle button “Use result as XY”. In this case the first result must be defined as the X position and the second as the Y position. Repeat this pattern for all edges.

5.11 Pixel Counter

Vision function ModelBasedPixelCounter is a function for counting pixels and extracting features from them. ModelBasedPixelCounter makes it possible to define regions within which the pixels corresponding to a predefined grayscale value interval (ThresholdMin/ThresholdMax) are counted.



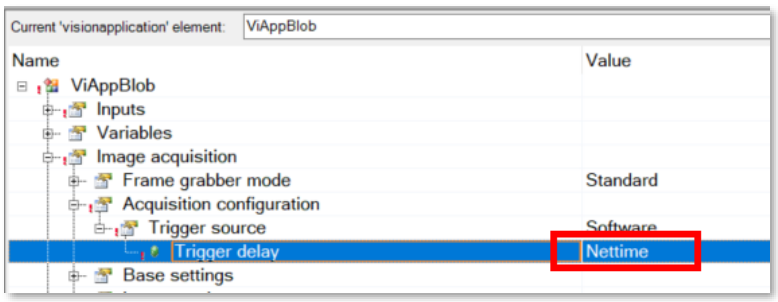
No	Model No	NumPixels	MinGray	MaxGray	MeanGray	DeviationGray	Position X	Position Y	ModelArea	NumConnectedComponents
1	0	0	0	0	0.00	0.00	0.00	0.00	0	0
2	0	0	0	0	0.00	0.00	0.00	0.00	0	0
3	0	0	0	0	0.00	0.00	0.00	0.00	0	0
4	0	0	0	0	0.00	0.00	0.00	0.00	0	0
5	0	0	0	0	0.00	0.00	0.00	0.00	0	0
6	0	0	0	0	0.00	0.00	0.00	0.00	0	0
7	0	0	0	0	0.00	0.00	0.00	0.00	0	0
8	0	0	0	0	0.00	0.00	0.00	0.00	0	0
9	0	0	0	0	0.00	0.00	0.00	0.00	0	0
10	0	0	0	0	0.00	0.00	0.00	0.00	0	0

If EnhancedPixelCounterInformation is disabled, only parameters ModelNumber, NumPixels and ModelArea are cyclically filled with data. If outputs are defined that do not correspond to those described above, they will be filled with 0.

Enabling this parameter increases the size of the POWERLINK frame since the remaining cyclically reading parameters are additionally transferred.

5.12 Using Nettime

In some applications it can be necessary to trigger the sensor periodicly depending on a drive position. This can be accomplished with nettime. To enable nettime the trigger delay has to be changed to nettime in the vision application settings.

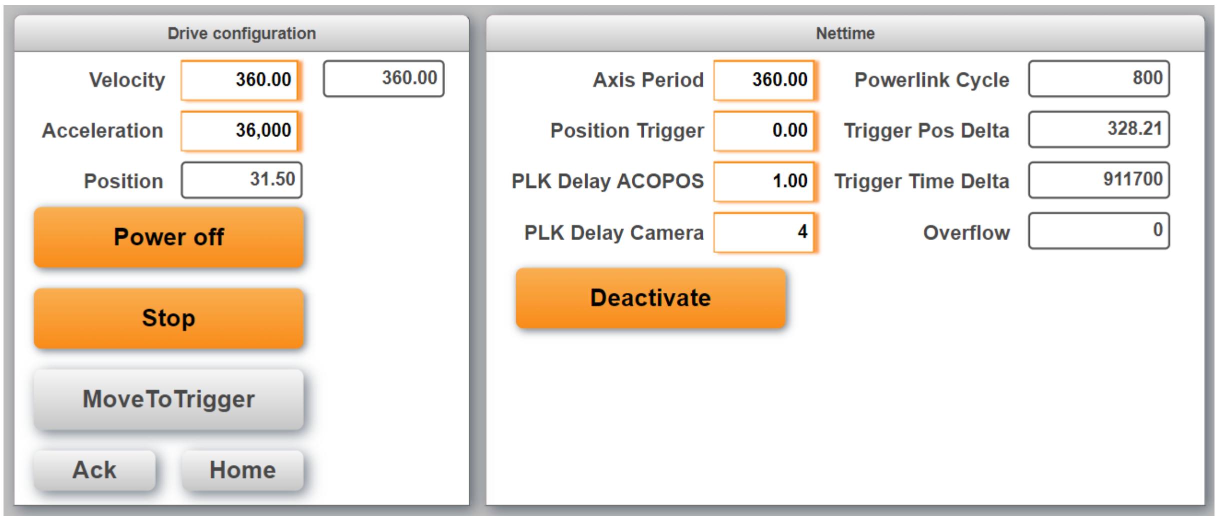


Current 'visionapplication' element: ViAppBlob

Name	Value
ViAppBlob	
Inputs	
Variables	
Image acquisition	
Frame grabber mode	Standard
Acquisition configuration	
Trigger source	Software
Trigger delay	Nettime
Base settings	

When this feature is enabled the manual trigger gVisionSensor[].CMD.ImageTrigger only works when the parameter gVisionSensor[].CFG.NettimeDelay is set correct (Current nettime value plus offset, ex. 10ms).

The task Vi_nettime provides the necessary calculation for a motion application. It is curcial that this task runs in sync and at the same cycle as the Powerlink bus. The following page allows the configuration of the nettime function.



Drive configuration

Velocity	360.00	360.00
Acceleration	36,000	
Position	31.50	

Nettime

Axis Period	360.00	Powerlink Cycle	800
Position Trigger	0.00	Trigger Pos Delta	328.21
PLK Delay ACOPOS	1.00	Trigger Time Delta	911700
PLK Delay Camera	4	Overflow	0

On the left hand side are the basic drive settings.

- Power: Switches the axis on and off. In the task “Axis” all the Axis-Handling is done. By default the setting is to use the encoder reference pulse. So when the axis is switched on and not homed it automatically searches the reference pulse
- Run: A continous movement will start with the set velocity and acceleration
- MoveToTrigger: Moves the axis to the “Position Trigger” (Nettime-settings)
- Ack: Acknowledges errors, if there are any errors
- Home: Makes again a homing, also if it was already done automatically while powering on.

On the right hand side are the nettime settings:

- Axis period: This is the number of units for one cycle (360 for a rotating axis). It should be the same value as in the axis settings
- Position Trigger: This is the position where the image will be made. Should be in the period

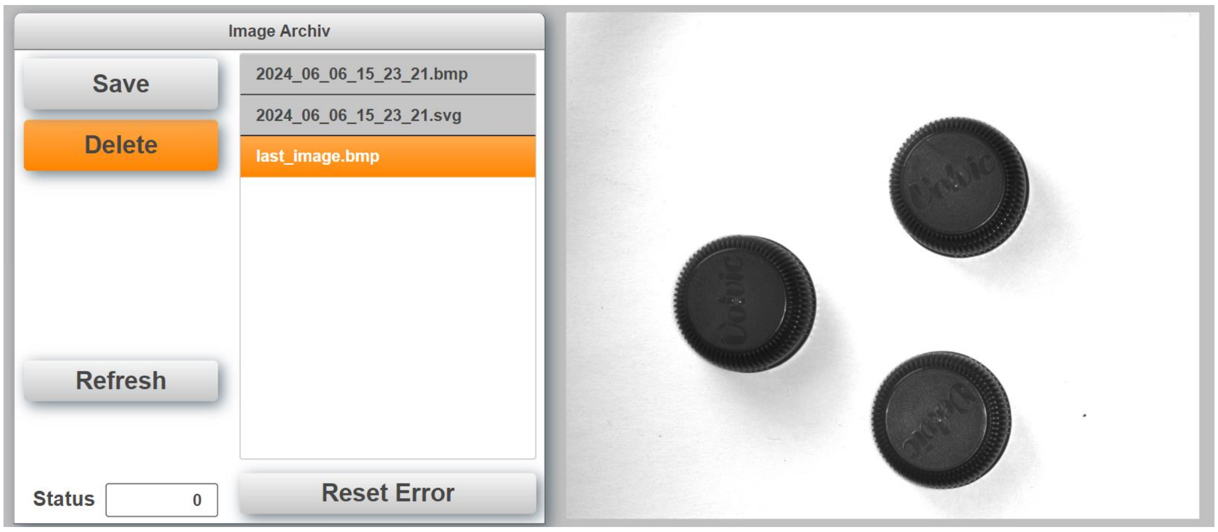
- PLK Delay ACOPOS (number of PLK cycles): When the PLC reads the motor position, this position is some time old, e.g. 1 oder 2 PLK cycles. This delay will be compensated. The value is the number of PLK cycles for the “age” of the motorposition. It is possible to use a value with fraction digits. This makes sense because of not only the Powerlink has a delay. E.g. also the encoder could have a small delay. So it is possible to adjust the value very precise
- PLK Delay Camera (number of PLK cycles): This is used to calculate the time when the nettime value must be set at the latest to make it to the sensor in time.The camera needs to get the nettime for the trigger some time before the trigger. A good value is 4. If the value is too small, the camera gets the nettime too late and can’t make the image any more. If the value is too high, the camera gets the nettime earlier. A speed-change will then no more be calculated.
- Powerlink Cycle: Powerlink cycletime (in microseconds)
- Trigger Pos Delta: This is the remaining position delta to the next trigger (in units)
- Trigger Time Delta: This is the remaining time delta to the next trigger (in microseconds)
- Overflow: If the nettime handling wants to send the next trigger to the camera, but the camera is not ready, this value will be increased by 1.

5.12.1 Precision

If the nettime seems not to be precise, it makes sense to check the lag error of the axis. To get very good results, a well tuned controller is necessaray.

5.13 Image Archive

The image archive is used to store sensor images on the PLC flash card. This can be necessary to inspect ‘bad’ products later in the process. The image archive is controlled by its own task “VisionImage” and structure (see 0).



2024_06_06_15_23_21.bmp
2024_06_06_15_23_21.svg
last_image.bmp

Status 0

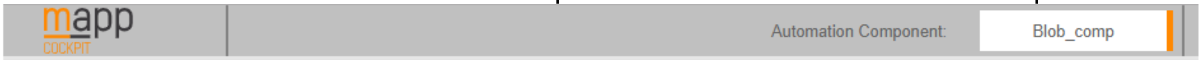
The number of images that are stored depends on the size of array *VisionImage.DATA.Images*. The default size is 20. When the list is full and a new images are uploaded the oldest images will automatically be deleted. The task will also highlight and load the newest image after upload or refresh.

6 Tips and Hints

6.1 The Vision Cockpit does not work correct and/or does not show the sensor image when the sensor is connected and ready.

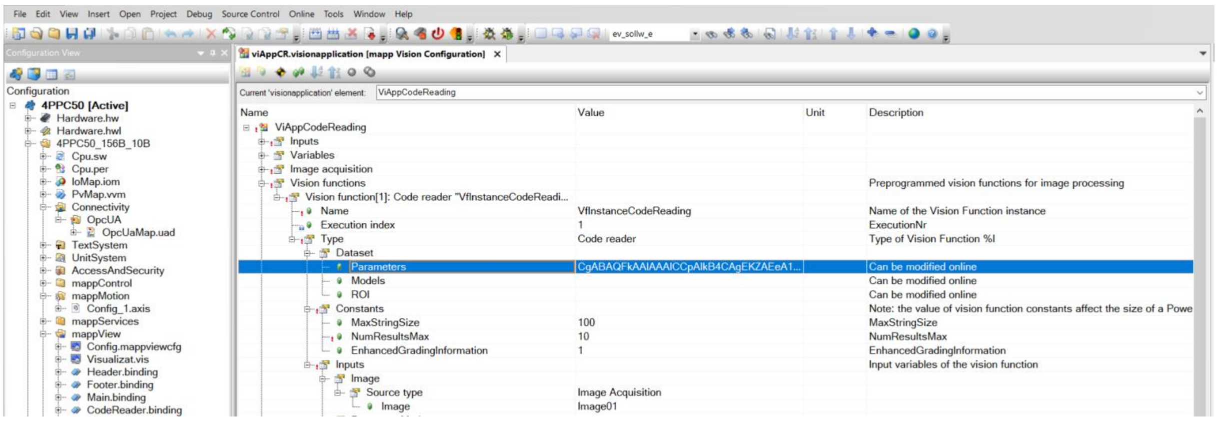
Make sure to adjust the IP address in the file “\Vision_1\Logical\Vision\setRouteToCamera.bat” and execute the batch file in Windows with right click (Run as administrator).

Make sure that the correct Automation Component is selected in the Vision Cockpit



6.2 How is the sensor configuration selected?

The default configuration is defined in the Automation Studio project under mappVision->…visionapplication.

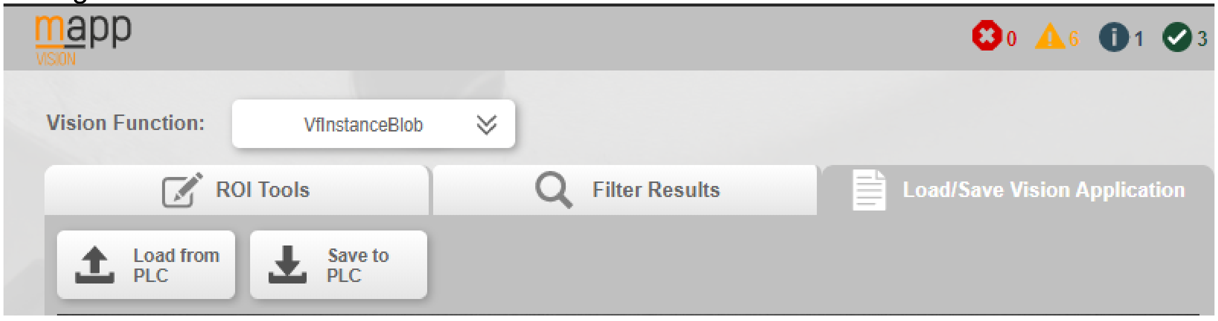


Current 'visionapplication' element: ViAppCodeReading

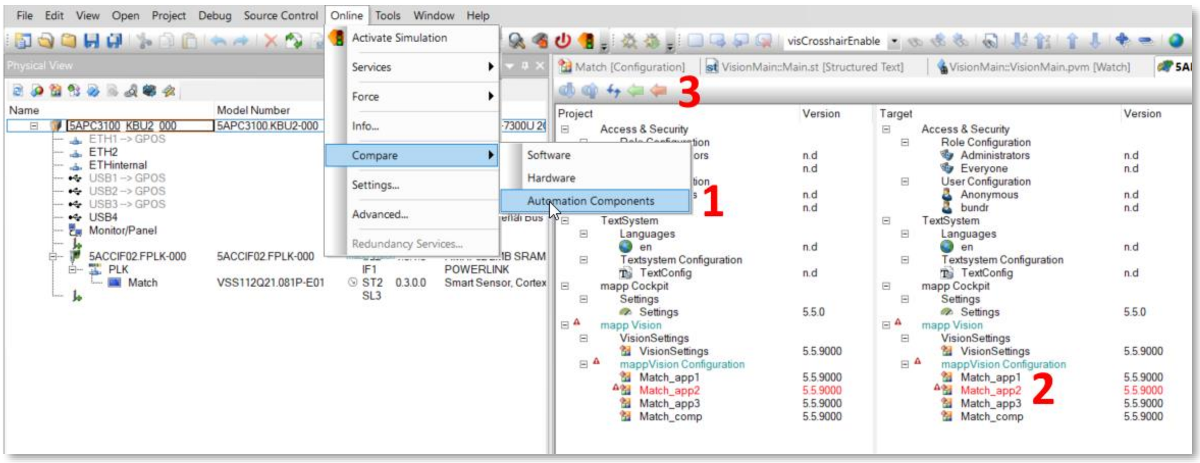
Name	Value	Unit	Description
ViAppCodeReading			
Inputs			
Variables			
Image acquisition			
Vision functions			Preprogrammed vision functions for image processing
Vision function[1]: Code reader "VfInstanceCodeReadi...			
Name	VfInstanceCodeReading		Name of the Vision Function instance
Execution index	1		ExecutionNr
Type	Code reader		Type of Vision Function %I
Dataset			
Parameters	CgABAQFkAAIAAAICCpAlkB4CAgEKZAEeA1...		Can be modified online
Models			Can be modified online
ROI			Can be modified online
Constants			Note: the value of vision function constants affect the size of a Powe
MaxStringSize	100		MaxStringSize
NumResultsMax	10		NumResultsMax
EnhancedGradingInformation	1		EnhancedGradingInformation
Inputs			Input variables of the vision function
Image			
Source type	Image Acquisition		
Image	Image01		

6.3 How to store a configuration taught in the vision Cockpit in the Automation Studio project.

Teach the configuration in the vision Cockpit and use the button “Save Vision Function Configuration”.



Go back into Automation Studio and select Online->Compare->Automation Components (1).



Select the vision application highlighted in red (2). Select orange arrow (3) at the top to transfer the sensor configuration back to Automation Studio.

7 Error numbers

Error number	Task	Description
3000	Vi_main	Number of applications exceeds array size. Increase MAX_IDX_VA_LIST
50001	Vi_image	Buffer for creating image file is too small

8 Revision History

➔ You can find the revision history also in the project (folder “Vision”/revision.txt)