相机在伺服设定位置触发拍照使用方法

基于 mappMotion

Date: September 20, 2023

I Versions

Version	Date	Comment	Edited by
1.0	Sep 18, 2023	First Edition	邱赵煜

Table 1: Versions

II Distribution

Name	Company, Department	Amount	Remarks

Table 2: Distribution

III Safety Notices

Safety notices in this document are organized as follows:

Safety notice	Description
Danger!	Disregarding the safety regulations and guidelines can be life-threatening.
Warning!	Disregarding the safety regulations and guidelines can result in severe injury or heavy damage to material.
Caution!	Disregarding the safety regulations and guidelines can result in injury or damage to material.
Information:	Important information used to prevent errors.

Table 3: Safety notices

IV Table of Contents

1 应用场景介绍 4

2 软硬件配置 5

2.1 软件版本 5

2.2 硬件拓扑 5

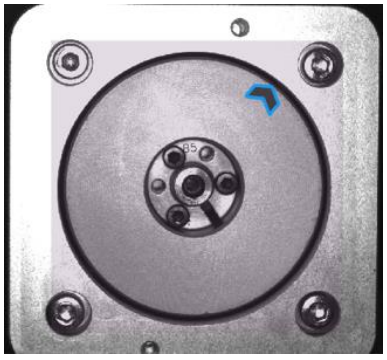
3 ViBaseAxisBasedAcquisition 功能块说明 6

4 AS 程序配置方法 8

1 应用场景介绍

贝加莱视觉相机的优势在于可与 powerlink 网络同步，可采用时间戳触发模式以达到微秒级响应精度。在较多印刷包装等应用场合中，需要使用相机与伺服电机或 Trak 进行位置同步以实现飞拍，在过去此类应用中，需要根据设定的拍照位置计算网络延时，提前计算触发拍照的时间戳，其中需要获取到伺服循环周期，powerlink 循环周期，task 循环周期，x2x 周期，相机触发信号处理周期，NettimeSoc 等参数用以计算 DelayTime，还需要根据实际照片的效果，对设定位置和延时时间等参数手动调整，整个过程较复杂。

在 mapp Vision 5.22 版本更新后，配合 mapp Motion，仅需简单配置和调用功能块的方式实现设定位置自动拍照，直线相机与伺服位置同步。其优势在于可根据网络拓扑和程序所在任务周期自动计算延时时间，且可根据实际情况进行上述参数微调，对于应用来说配置方便，也具有灵活性。劣势在于必须配合 mapp Motion 系统，老项目中如果使用了 ACP10 或 ncACTION，要添加视觉系统，该功能还是有一定局限性。根据实际应用案例数量和 mapp Vision 软件功能更新侧重点推测，此功能应用场合可能更多地会与 ACOPOStrak 配合使用。

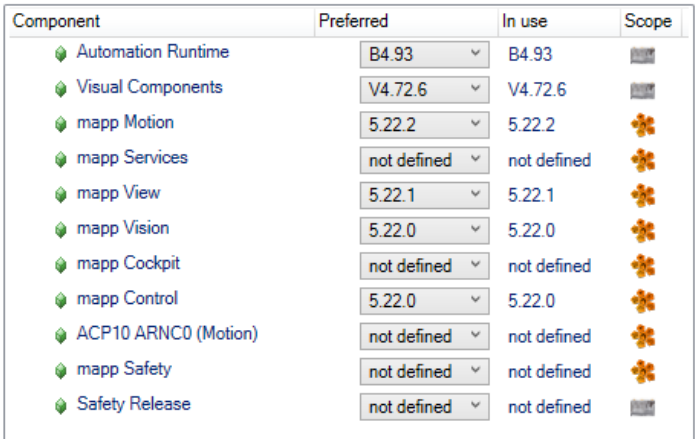


办公室测试使用 Smart Sensor 相机，在设定位置拍摄旋转运动中 Demo 设备上的箭头，并在每次拍摄后使用 Matching 功能读取到箭头所在的像素坐标位置，在一段时间内累计记录坐标位置，计算最大偏移量来反映拍照触发精度。本文档总结测试过程的经验，提供配置方式和功能块使用方法供参考，对 mapp Motion 和相机的基本使用配置不作详述，请参考相关培训材料。如果测试过程中遇到问题或有其它建议欢迎指正交流！

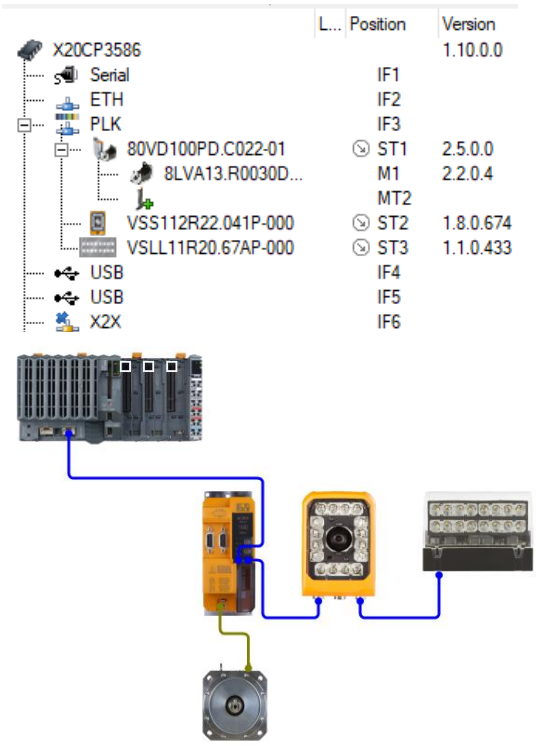
2 软硬件配置

2.1 软件版本

注意，使用该功能必须将 mapp Vision 版本升级到 V5.22 及以上，mapp Motion 和 mapp View 也需同步更新到对应版本。由于测试程序中有识别位置偏差量计算的功能块，因此额外加了 mapp Control，根据实际情况 mapp Control 可加可不加。



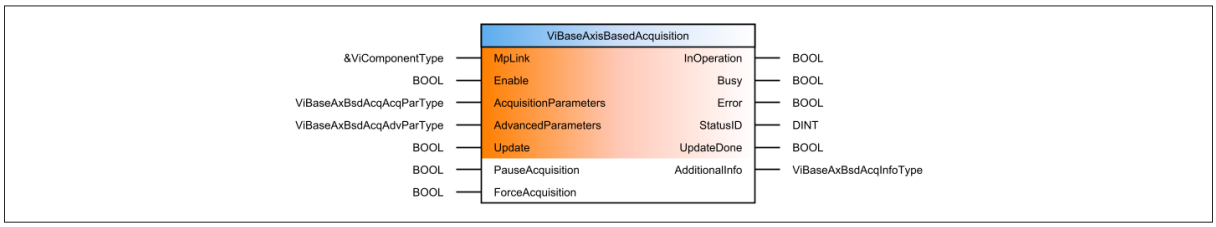
2.2 硬件拓扑



3 ViBaseAxisBasedAcquisition 功能块说明

ViBaseAxisBasedAcquisition 功能块基于轴位置计算机器视觉相机的图像采集时间戳，并在这些位置采集图像。需要与相机同步的轴可直接在 Vision 应用组件中选择对应的 mapp Motion 轴名称进行配置。其最大特点是在设定位置到达前，自动计算延迟时间，提前计算输出时间戳到相机，自动计算包含 CPU 获取伺服编码器信号的延时、程序处理延时、通讯输出延时和相机数据处理延时，功能块也提供了开放参数可手动校准输入输出延时。

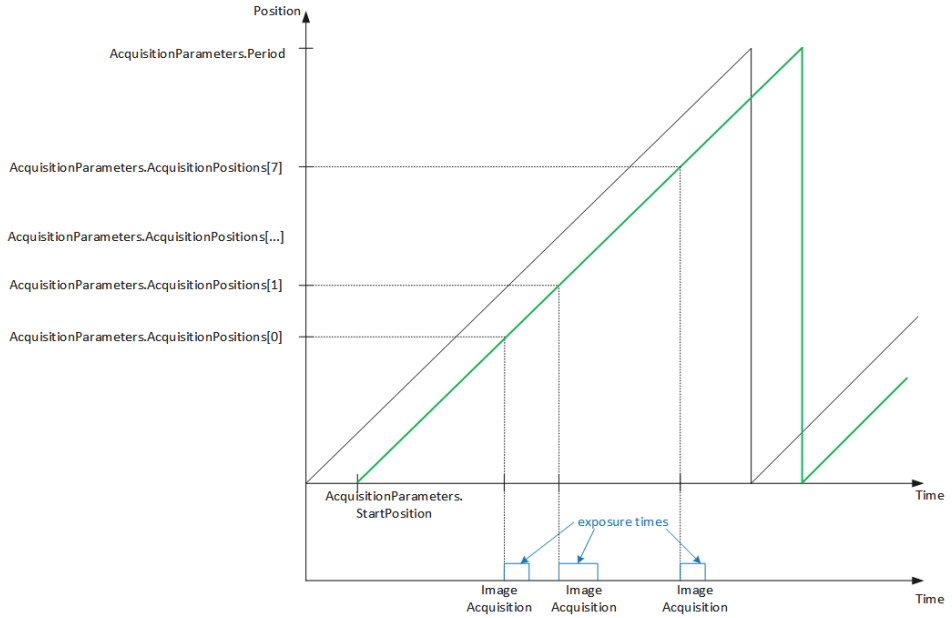
Function block



Interface

I/O	Name	Data type	Description
IN	MpLink	Pointer to ViComponentType	Pointer to the vision component from which images should be acquired.
IN	Enable	BOOL	Enables the function block and starts the calculations.
IN	AcquisitionParameters	ViBaseAxBsdAcqAcqParType	Parameters that define when the acquisition is triggered depending on the axis position.
IN	AdvancedParameters	ViBaseAxBsdAcqAdvParType	Structure for using advanced functions.
IN	Update	BOOL	Updates the acquisition parameters. See Function description.
IN	PauseAcquisition	BOOL	If TRUE, the internal calculations are still active, but no image is acquired.
IN	ForceAcquisition	BOOL	A rising edge on this input will force an image acquisition as quickly as possible. Note: • By setting this input, an image is acquired even if PauseAcquisition is set.
OUT	InOperation	BOOL	The calculations are active, and the timestamps are forwarded to the vision component.
OUT	Busy	BOOL	The function block is active and must continue to be called.
OUT	Error	BOOL	Error during execution.
OUT	StatusID	DINT	Status information.
OUT	UpdateDone	BOOL	Indicates that new acquisition parameters have been initialized.
OUT	AdditionalInfo	ViBaseAxBsdAcqAddInfoType	Additional useful data, for example for system monitoring or error analysis.

当相机拍照设定为伺服位置触发时，<ImageAcquisition> 和<DelayNetTime(n)>将不再生效，必须调用此功能块，设定伺服位置点后自动触发拍照。如果需要强制手动触发拍照或暂停拍照，需要使用此功能块中的指令。需要注意的是，为保证同步性，调用 mapp Motion 功能块的程序任务 **tolerance 必须设 0。**



在 ViBaseAxBsdAcqParType 结构体变量中，可以输入拍摄位置、轴周期和启动位置。在一个周期内最多可以输入 8 个拍摄位置，StartPosition 可以理解为设定拍摄位置的偏移量，在实际应用中可以用于微调
拍摄位置。

ViBaseAxBsdAcqAcqParType

Parameter	Data type	Description
AcquisitionPositions	LREAL[8]	Positions in [measurement units] within the defined "Period" or axis period in which images should be acquired. Notes: • For index > 0, value 0.0 defines an inactive position. • Starting from the first inactive position, all following array entries must also be defined as inactive. • All active positions must be strictly monotonically increasing.
Period	LREAL	Period in [measurement units] if a non-periodic axis is used or a different axis period than the one defined should be used. Defines the period within which the acquisition positions are defined or after which they are repeated. Note: • Only values greater than 0 and less than 2^{31} * the "measuring resolution" of the axis are permitted.
StartPosition	LREAL	Start position in [measurement units] of the period. Notes: • Non-periodic axis: The value corresponds to an absolute position value at which the period for the acquisition positions is started. If this position has already been exceeded, the function is started at the next multiple of "StartPosition" + "Period". • Periodic axis: The value corresponds to a position value within the axis period at which the function should be started. If a value other than "0" is specified, the start of "Period" is shifted with respect to the start of the axis period. Range of possible values: [0, axis period].

在 ViBaseAxBsdAcqAdvParType 结构体变量中，可以限定拍照时伺服的运动方向，也可以手动设定位置输入延迟的补偿量和输出延时的补偿量。在系统自动计算出的输入位置延迟时间和输出到相机的延时时间基础上再进行补偿。

ViBaseAxBsdAcqAdvParType

Parameter	Data type	Description
DisableNegativeDirection	BOOL	If TRUE, the image acquisition is only triggered when the position value of the axis increases.

AddPositionCompensation	REAL	The time value [s] configured here is added to the automatically determined position compensation time (see DefaultPositionCompensation for ViBasedAxBsdAcqAddInfoType).
		If necessary, this can be used to compensate for the duration between the position evaluation and the reception of the position on the PLC.
		Negative values are also permitted.
		Note:
		• If a position setpoint of an axis is used, the correct position compensation time is automatically determined internally. If actual position values are used (e.g. from an external encoder axis), the following times must also be taken into account:
		  • Evaluation time of the encoder
		  • Evaluation time of the counter module
		  • Network cycle time (X2X, POWERLINK, etc.)
		• See also Automation Help:
		  • X2X Link - Latency times
		  • POWERLINK - Input latency

AddOutputCompensation	LREAL	The time value [s] configured here is added to the automatically determined output compensation time (see DefaultOutputCompensation for ViBasedAxBsdAcqAddInfoType). If necessary, this can be used to compensate for the duration required to transfer the timestamp to the camera module. Negative values are also permitted. Note: • With a default setting, the correct output compensation time is automatically determined internally.

Table: ViBaseAxBsdAcqAdvParType

ViBaseAxBsdAcqAddInfoType 结构体变量中，功能块输出自动计算出的输入位置延迟时间、输出到相机的延时时间和输入到相机的时间戳。

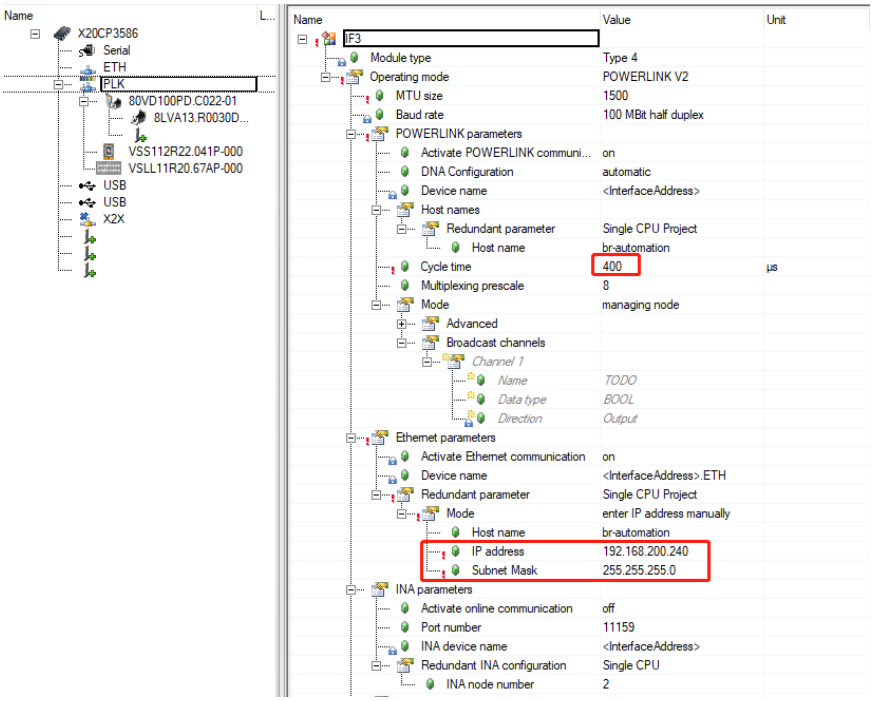
ViBaseAxBsdAcqAddInfoType

Parameter	Data type	Description
DefaultPositionCompensation	REAL	Automatically determined position compensation time [s].
DefaultOutputCompensation	REAL	Automatically determined output compensation time [s].
TimestampCount	SINT	The value is changed when a new timestamp value is calculated for the image acquisition. Note: • The value is not set if "PauseAcquisition" is set.
CalculatedTimestamp	DINT	Last calculated timestamp value forwarded to the vision component. Note: • The value is not set if "PauseAcquisition" is set.

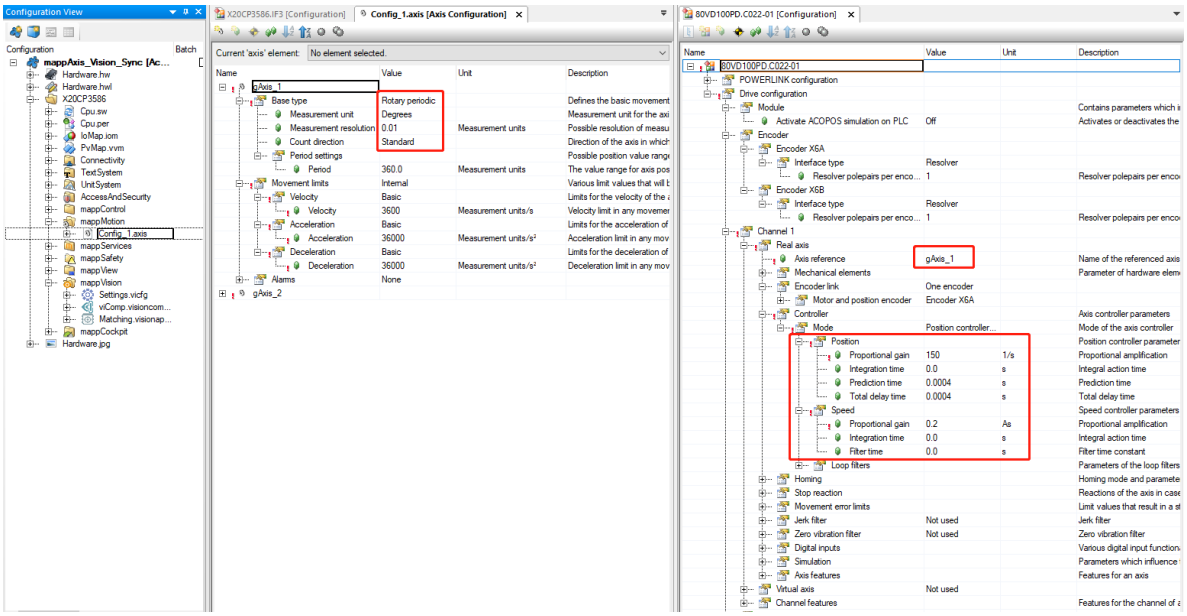
Table: ViBaseAxBsdAcqAddInfoType

4 AS 程序配置方法

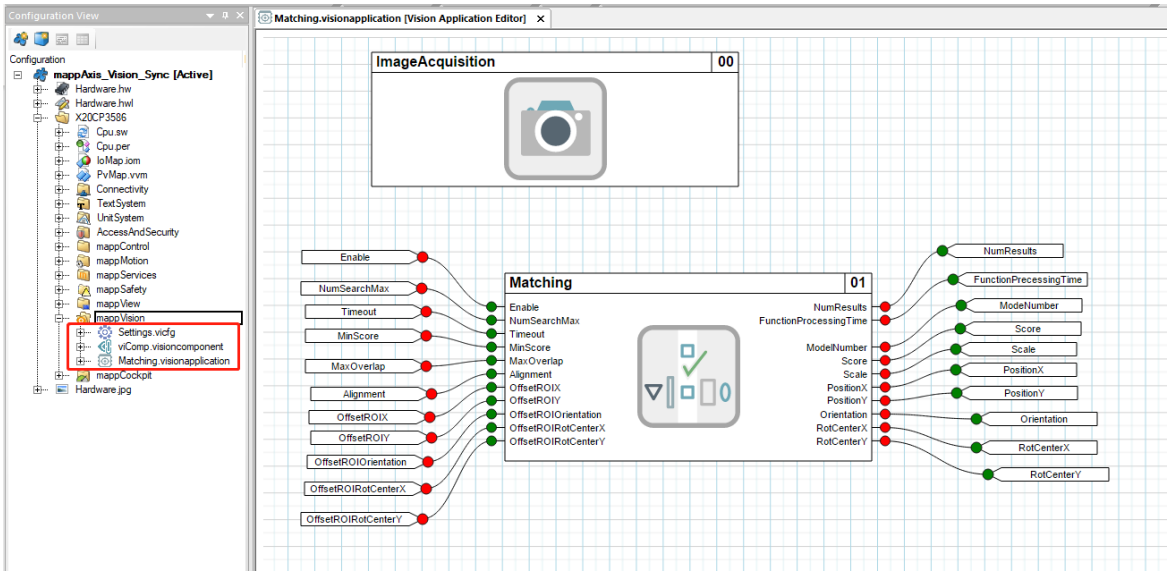
1. 设置 PLK 循环周期和 IP 地址，注意这里的 IP 不要与 CPU 的 IP 地址冲突。



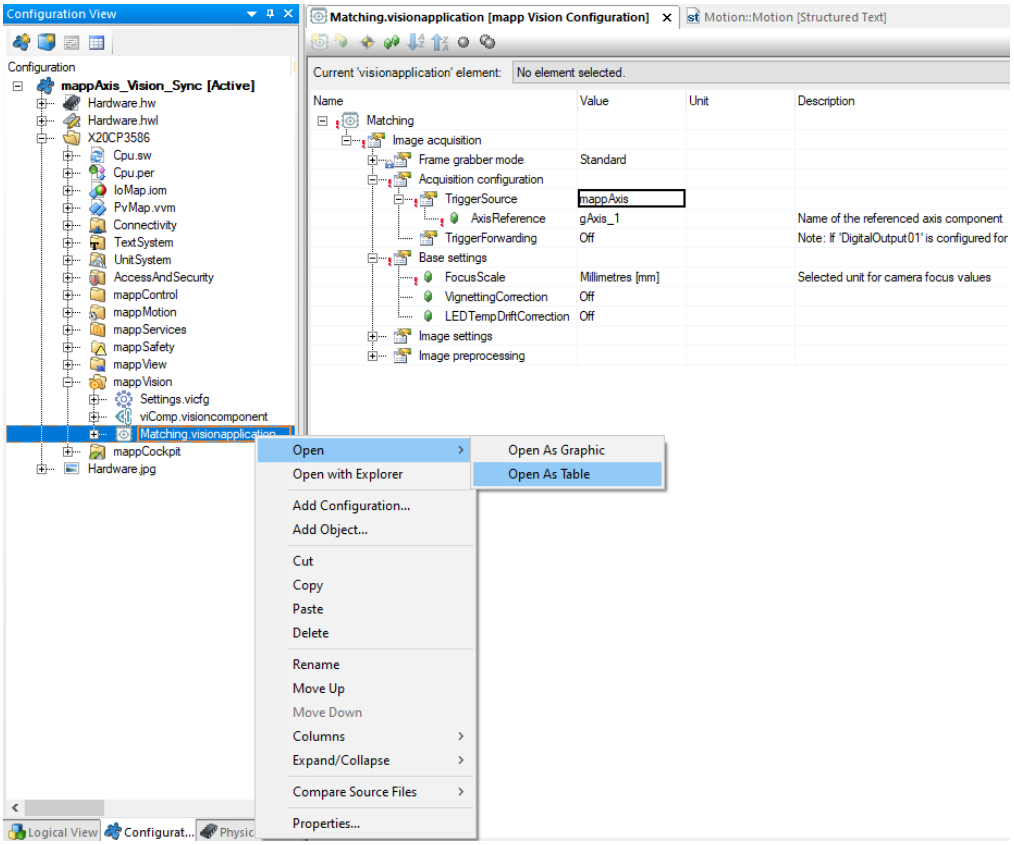
2. 添加伺服硬件配置及 mapp Motion 配置，测试选择旋转周期性轴，单位度，精度 0.01 度。三环参数根据实际情况使用 Cockpit 整定。



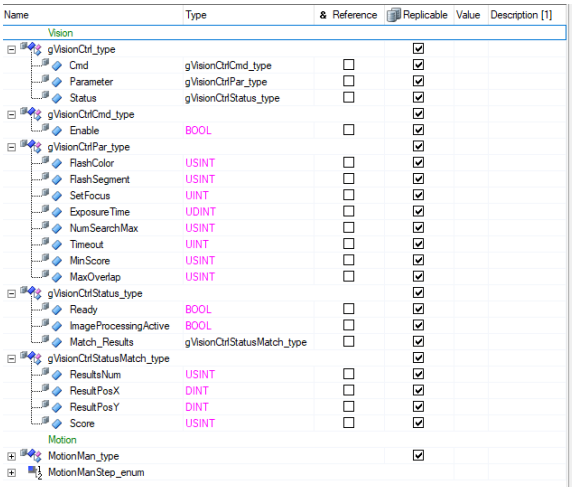
3. 在 mapp Vision 中添加配置文件，测试使用了<Matching>功能，配置接口变量。



4. 在 mapp Vision 5.22 版本中，编辑功能需要手动以 Table 视窗打开，在<TriggerSource>中选择，<mappAxis>，并根据实际情况配置焦距单位等参数。保存后，相机 IO Mapping 中的<ImageAcquisition>变量将不再开放，默认自动触发。



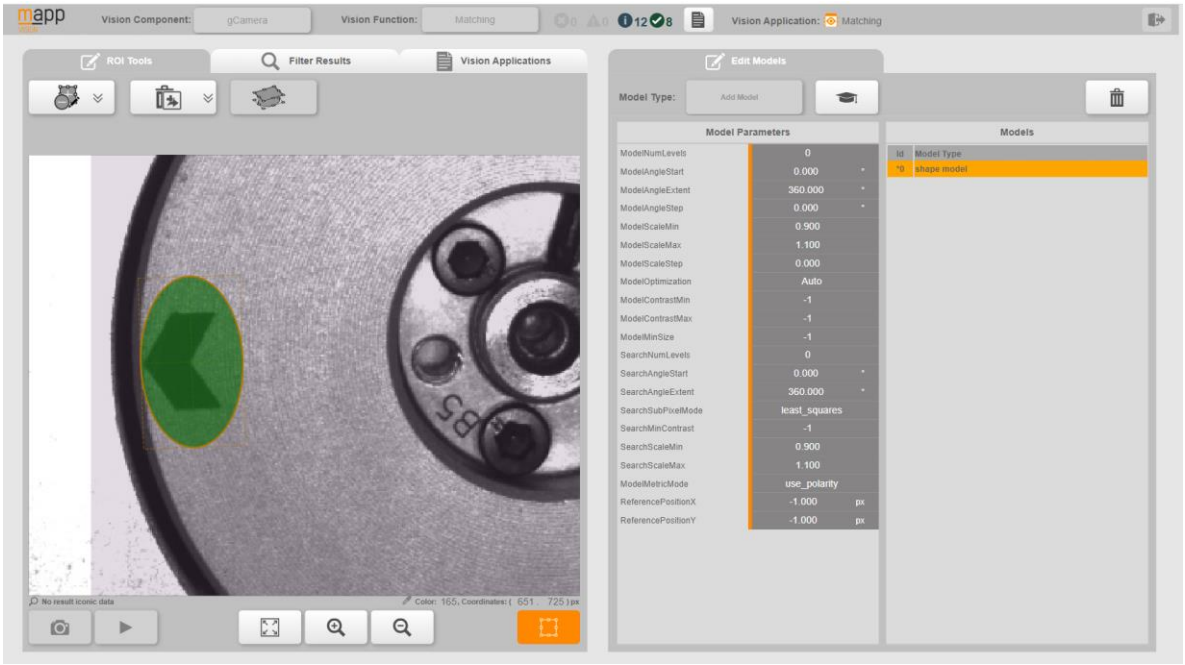
5. 在相机 IO Mapping 中关联相关结构体变量。

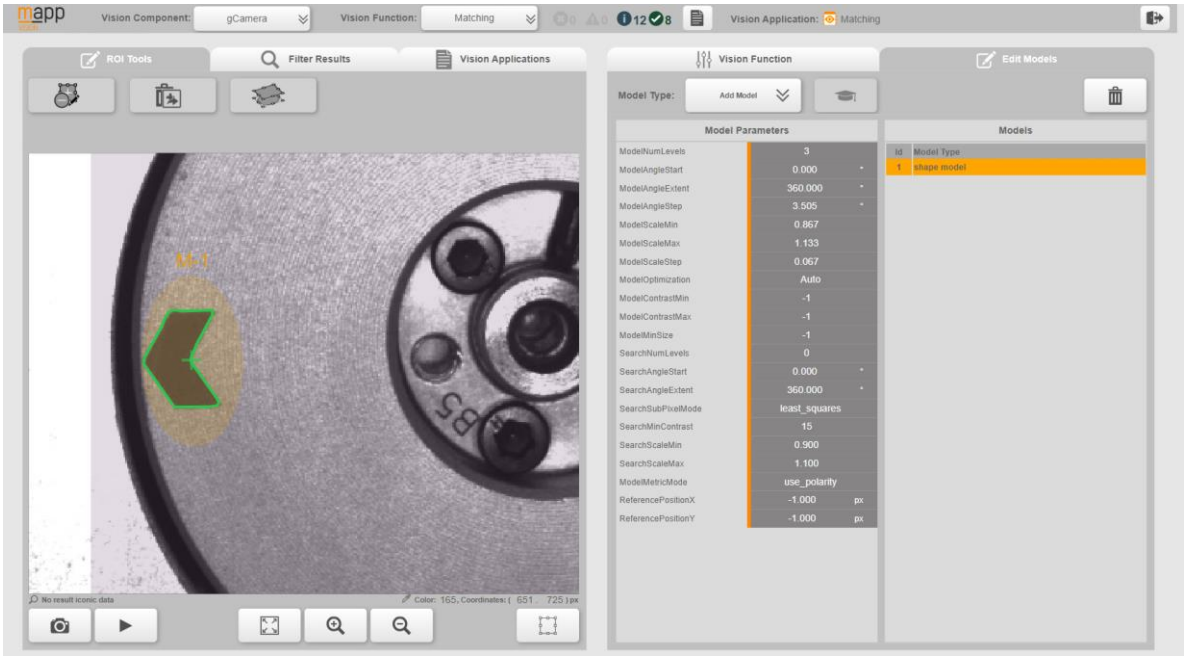


Name	Type	Reference	Replicable	Value	Description [1]
Vision					
gVisionCtrl_type			☑		
Cmd	gVisionCtrlCmd_type	☐	☑		
Parameter	gVisionCtrlPar_type	☐	☑		
Status	gVisionCtrlStatus_type	☐	☑		
gVisionCtrlCmd_type			☑		
Enable	BOOL	☐	☑		
gVisionCtrlPar_type			☑		
FlashColor	USINT	☐	☑		
FlashSegment	USINT	☐	☑		
SetFocus	UINT	☐	☑		
ExposureTime	UDINT	☐	☑		
NumSearchMax	USINT	☐	☑		
Timeout	UINT	☐	☑		
MinScore	USINT	☐	☑		
MaxOverlap	USINT	☐	☑		
gVisionCtrlStatus_type			☑		
Ready	BOOL	☐	☑		
ImageProcessingActive	BOOL	☐	☑		
Match_Results	gVisionCtrlStatusMatch_type	☐	☑		
gVisionCtrlStatusMatch_type			☑		
ResultsNum	USINT	☐	☑		
ResultPosX	DINT	☐	☑		
ResultPosY	DINT	☐	☑		
Score	USINT	☐	☑		
Motion					
MotionMan_type			☑		
MotionManStep_enum					

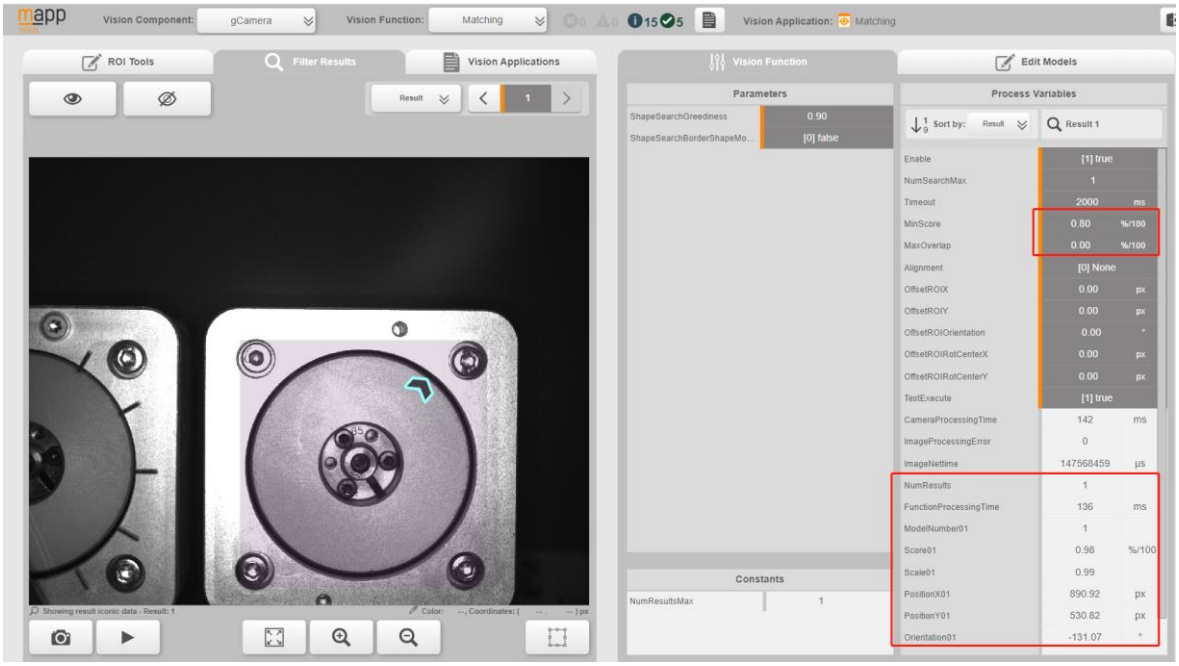
Channel Name	Process Variable	Data Type	Task Class
ReadFocus		UINT	
ReadExposureTime		UDINT	
GainLevel		USINT	
SetFocus	::gVisionCtrl.Parameter.SetFocus	UINT	Automatic
ExposureTime01	::gVisionCtrl.Parameter.ExposureTime	UDINT	Automatic
FlashColor01	::gVisionCtrl.Parameter.FlashColor	USINT	Automatic
FlashSegment01	::gVisionCtrl.Parameter.FlashSegment	USINT	Automatic
UndervoltageError		BOOL	
DigitalInput01		BOOL	
DigitalOutput01		BOOL	
DigitalOutputStatus01		BOOL	
ImageAcquisitionReady	::gVisionCtrl.Status.Ready	BOOL	Automatic
ImageProcessingActive	::gVisionCtrl.Status.ImageProcessingActive	BOOL	Automatic
SearchAcquisitionSettings		BOOL	
IRFilter		BOOL	
ChromaticLock		BOOL	
CameraProcessingTime		UINT	
ImageProcessingError		UINT	
ImageNettime		DINT	
Enable	::gVisionCtrl.Cmd.Enable	BOOL	Automatic
NumSearchMax	::gVisionCtrl.Parameter.NumSearchMax	USINT	Automatic
Timeout	::gVisionCtrl.Parameter.Timeout	UINT	Automatic
MinScore	::gVisionCtrl.Parameter.MinScore	USINT	Automatic
MaxOverlap	::gVisionCtrl.Parameter.MaxOverlap	USINT	Automatic
Alignment		USINT	
OffsetROIX		DINT	
OffsetROIY		DINT	
OffsetROIOrientation		INT	
OffsetROIRotCenterX		DINT	
OffsetROIRotCenterY		DINT	
NumResults	::gVisionCtrl.Status.Match_Results.ResultsNum	USINT	Automatic
FunctionPrecessingTime		UINT	
ModeNumber01		USINT	
Score01	::gVisionCtrl.Status.Match_Results.Score	USINT	Automatic
Scale01		USINT	
PositionX01	::gVisionCtrl.Status.Match_Results.ResultPosX	DINT	Automatic
PositionY01	::gVisionCtrl.Status.Match_Results.ResultPosY	DINT	Automatic
Orientation01		INT	

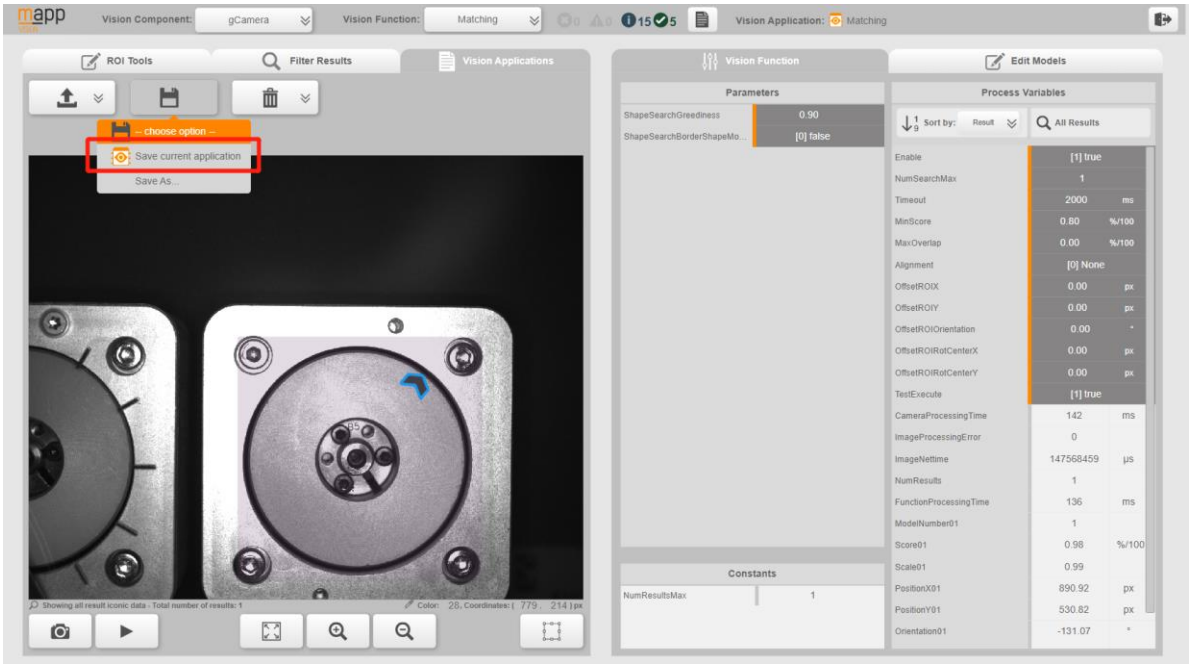
6. 下载程序，打开相机示教页面<xxx.xxx.xxx.xxx:81/index.html?visuid=visVision>，调试好光源、焦距和曝光时间等参数，获取清晰照片后，在 Matching 模式下添加模型，将标记框在 ROI 区域内，识别出箭头标志的模型。

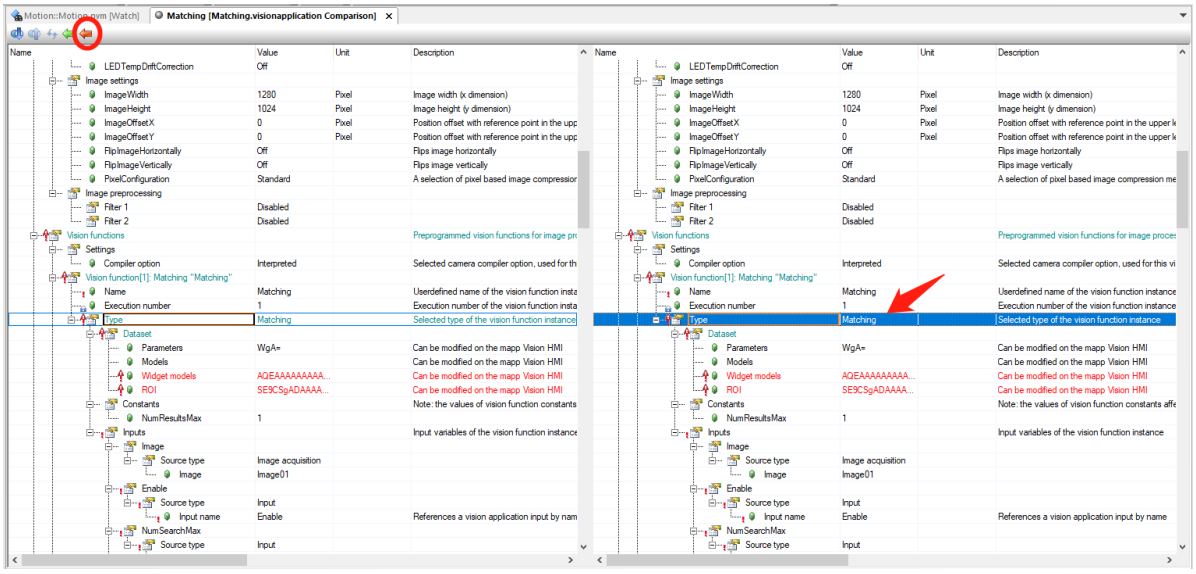


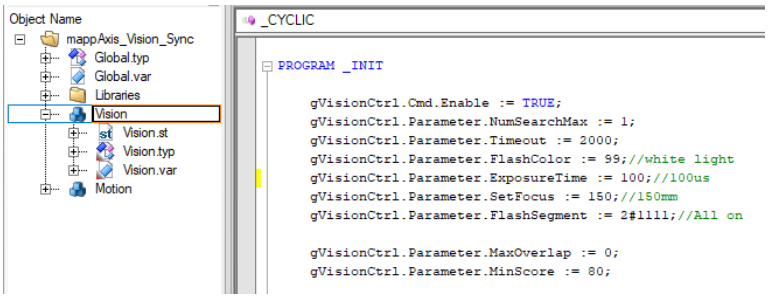


7. 转动电机将箭头移动到其它几个不同位置，在示教页面手动拍照进行识别，测试是否能成功识别出模型位置坐标，确认无误后保存当前应用配置参数。完成后在 AS 中打开<Matching.visionapplication→Compare Online>，在线将示教的 ROI 与模型同步到程序配置中，同时测试得到的光源、焦距、曝光时间等参数写在程序初始化段。




mapp
Vision Component:
gCamera
Vision Function:
Matching
Vision Application:
Matching
ROI Tools
Filter Results
Vision Applications
Vision Function
Edit Models
Parameters
Process Variables
-- choose option --
Save current application
Save As...
Constants


Matching [Matching.visionapplication Comparison]
Name
Value
Unit
Description


Object Name
mappAxis_Vision_Sync
Global.typ
Global.var
Libraries
Vision
Vision.st
Vision.typ
Vision.var
Motion
_CYCLIC
PROGRAM _INIT
gVisionCtrl.Cmd.Enable := TRUE;
gVisionCtrl.Parameter.NumSearchMax := 1;
gVisionCtrl.Parameter.Timeout := 2000;
gVisionCtrl.Parameter.FlashColor := 99;//white light
gVisionCtrl.Parameter.ExposureTime := 100;//100us
gVisionCtrl.Parameter.SetFocus := 150;//150mm
gVisionCtrl.Parameter.FlashSegment := 2#1111;//All on
gVisionCtrl.Parameter.MaxOverlap := 0;
gVisionCtrl.Parameter.MinScore := 80;

8. 以下程序用于计算每个周期设定位置拍摄的照片中，箭头图标在 X 和 Y 坐标上最大偏差值。<ErrMax_PosX>和<ErrMax_PosY>可反映拍摄时间精度和同步性，单位为像素。

Name	Type	Reference	Constant	Retain	Replicable
MTDataMinMax_X	MTDataMinMax	☐	☐	☐	☑
MTDataMinMax_Y	MTDataMinMax	☐	☐	☐	☑
ErrMax_PosX	REAL	☐	☐	☐	☑
ErrMax_PosY	REAL	☐	☐	☐	☑

```
PROGRAM _CYCLIC

    gVisionCtrl;

    //Record the MaxError of acquired label position
    MTDataMinMax_X.Enable := USINT_TO_BOOL(gVisionCtrl.Status.Match_Results.ResultsNum);
    MTDataMinMax_X.In := DINT_TO_REAL(gVisionCtrl.Status.Match_Results.ResultPosX);
    MTDataMinMax_X();
    MTDataMinMax_Y.Enable := USINT_TO_BOOL(gVisionCtrl.Status.Match_Results.ResultsNum);
    MTDataMinMax_Y.In := DINT_TO_REAL(gVisionCtrl.Status.Match_Results.ResultPosY);
    MTDataMinMax_Y();

    ErrMax_PosX := ABS(MTDataMinMax_X.MaxValue - MTDataMinMax_X.MinValue);
    ErrMax_PosY := ABS(MTDataMinMax_Y.MaxValue - MTDataMinMax_Y.MinValue);

END_PROGRAM
```

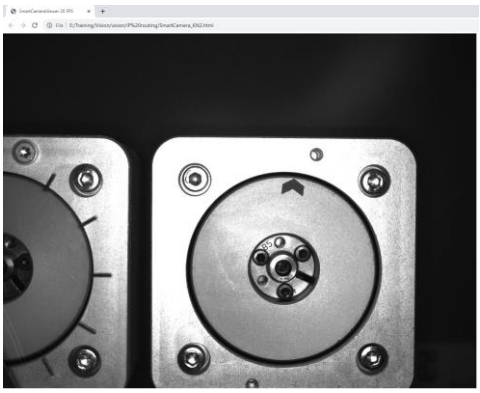
9. Motion 程序中添加<ViBaseAxisBasedAcquisition>功能块，在程序初始化段定义了拍摄位置 180°，周期与轴周期相同 360°，启动位置设定为 0。循环程序中定义好 Enable 条件和其它输入变量并调用改功能块。下载程序，注意该程序所在的 Task，Tolerance 需要设定为 0。

Name	Type	Reference	Constant	Retain	Replicable	Value
ViBaseAxisBasedAcquisition_0	ViBaseAxisBasedAcquisition	☐	☐	☐	☑	
ViBaseAxBsdAcqAcqParTyp	ViBaseAxBsdAcqAcqParType	☐	☐	☐	☑	
ViBaseAxBsdAcqAdvParTyp	ViBaseAxBsdAcqAdvParType	☐	☐	☐	☑	

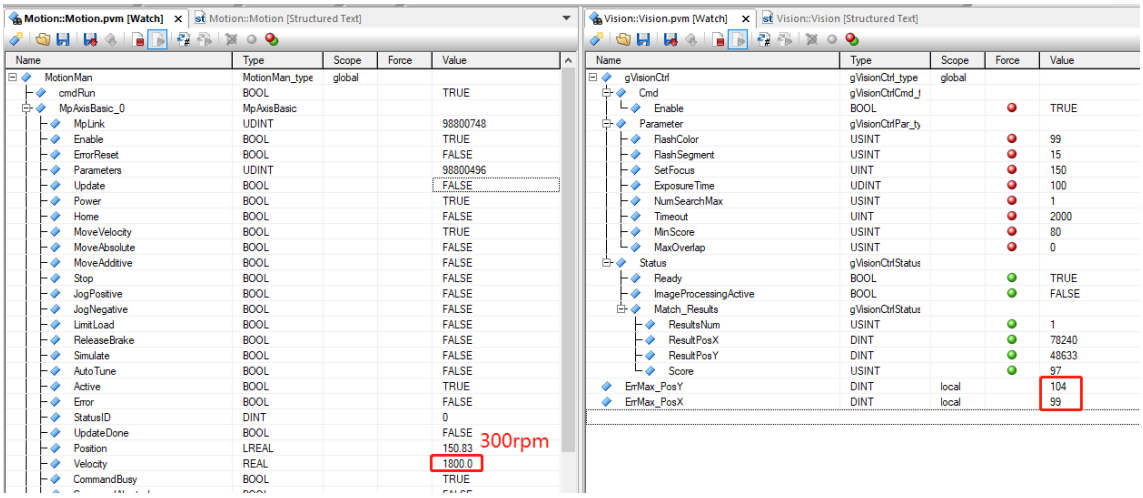
```
PROGRAM _INIT
    (* Insert code here *)
//To be done:
 // Motion function block initialization

    MotionMan.MpAxisBasicPar_0.Velocity := 360;//60rpm as default
    MotionMan.MpAxisBasicPar_0.Acceleration := 1800;
    MotionMan.MpAxisBasicPar_0.Deceleration := 1800;

    MotionMan.MpAxisBasic_0.Enable := TRUE;
    MotionMan.MpAxisBasic_0.MpLink := ADR(gAxis_1);
    MotionMan.MpAxisBasic_0.Parameters := ADR(MotionMan.MpAxisBasicPar_0);
    MotionMan.MpAxisBasic_0();

    ViBaseAxBsdAcqAcqParTyp.AcquisitionPositions[0] := 180;//Set auto imageAcquisition position 1(180°) in one set period
//  ViBaseAxBsdAcqAcqParTyp.AcquisitionPositions[1] := 270;//Set auto imageAcquisition position 2(270°) in one set period
    ViBaseAxBsdAcqAcqParTyp.Period := 360;
    ViBaseAxBsdAcqAcqParTyp.StartPosition := 0;

END_PROGRAM
```

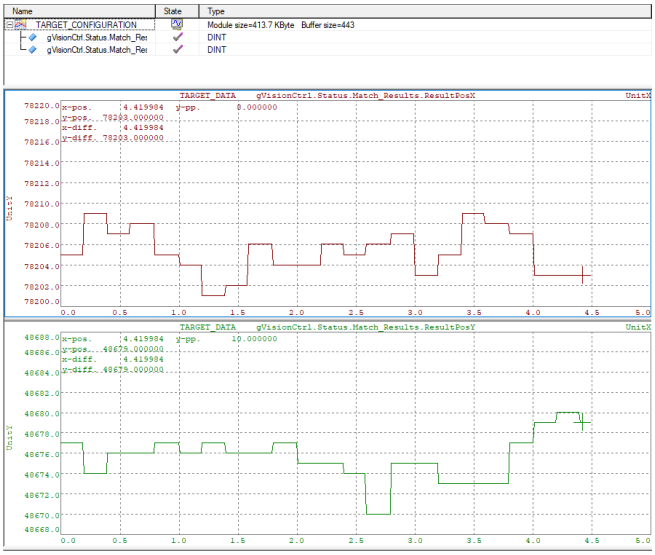
```
PROGRAM _CYCLIC
    // Main Motion state machine
    CASE MotionMan.Step OF . . .


    // Motion function block call
    MotionMan.MpAxisBasic_0();


    (************************Relate mappAxis to vision components*******************************)
    ViBaseAxisBasedAcquisition_0.MpLink := ADR(gCamera);
    ViBaseAxisBasedAcquisition_0.Enable := MotionMan.cmdRun AND gVisionCtrl.Status.Ready;
    ViBaseAxisBasedAcquisition_0.AcquisitionParameters := ViBaseAxBsdAcqAcqParTyp;
    ViBaseAxisBasedAcquisition_0.AdvancedParameters := ViBaseAxBsdAcqAdvParTyp;
    ViBaseAxisBasedAcquisition_0();

END_PROGRAM
```

10. CPU 运行后，给电机 MoveVelocity 指令旋转，这时每经过 180°时相机会自动拍照。打开 SmartCameraViewer，可以看到静止的图片。尝试用遮挡物进行遮挡镜头，页面可实时显示，并可在 watch 窗口中监控到<gVisionCtrl.Status.Match_Results.ResultsNum>为 0，拿开遮挡物后又恢复为 1。

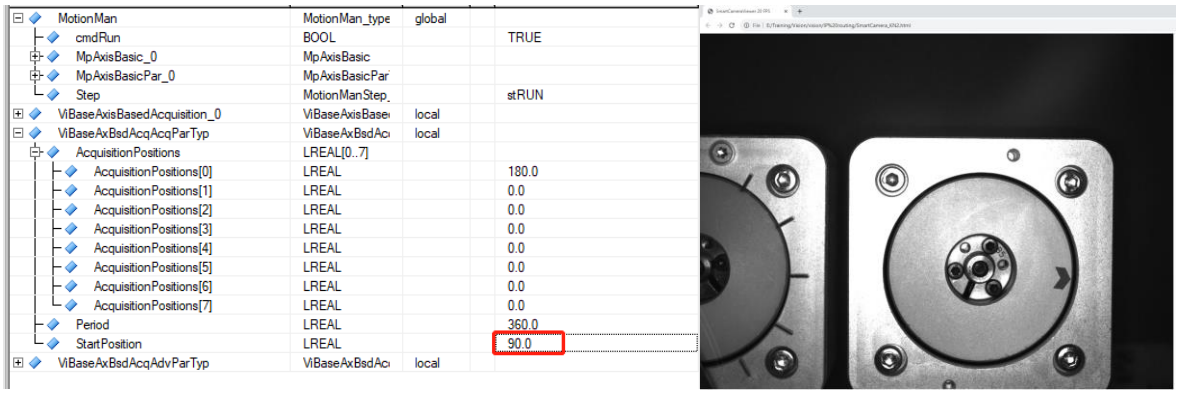


11. 测试中负载轮以 300rpm 转动，经过约 10 分钟后，在 Watch 中查看 X 与 Y 方向的最大像素偏差值。同时在 5s 内 Trace 了<ResultPosX>和<ResultPosY>，偏差量 PP 值约 10 个像素。





12. 更改 StartPosition 为 90°，即在设定拍照位置上加 90°，可以看到照片中箭头位置相对之前顺时针转了 90°。



MotionMan	MotionMan_type	global	
cmdRun	BOOL		TRUE
MpAxisBasic_0	MpAxisBasic		
MpAxisBasicPar_0	MpAxisBasicPar		
Step	MotionManStep_		stRUN
ViBaseAxisBasedAcquisition_0	ViBaseAxisBase	local	
ViBaseAxBsdAcqAcqParTyp	ViBaseAxBsdAc	local	
AcquisitionPositions	LREAL[0..7]		
AcquisitionPositions[0]	LREAL		180.0
AcquisitionPositions[1]	LREAL		0.0
AcquisitionPositions[2]	LREAL		0.0
AcquisitionPositions[3]	LREAL		0.0
AcquisitionPositions[4]	LREAL		0.0
AcquisitionPositions[5]	LREAL		0.0
AcquisitionPositions[6]	LREAL		0.0
AcquisitionPositions[7]	LREAL		0.0
Period	LREAL		360.0
StartPosition	LREAL		90.0
ViBaseAxBsdAcqAdvParTyp	ViBaseAxBsdAc	local	

13. 监控 AdditionalInfo，可以看到当前的延时补偿时间。

输入位置延时时间为 0.8ms，由 1 个伺服运算周期 0.4ms 和一个 PLK 周期 0.4ms 组成。

输出延时补偿时间为 12.15ms，由任务循环 10ms，PLK 周期 0.4ms，其余 2.11ms 个人推测是包含了相机内部 Buffer 处理到执行拍照的延时时间，和 HUB 延时时间。此处运算机制帮助中没有特别清晰的解释，此处如有问题请及时指正。

ViBaseAxisBasedAcquisition_0	ViBaseAxisBase	local	
MpLink	UDINT		98800764
Enable	BOOL		TRUE
AcquisitionParameters	ViBaseAxBsdAc		
AdvancedParameters	ViBaseAxBsdAc		
Update	BOOL		FALSE
PauseAcquisition	BOOL		FALSE
ForceAcquisition	BOOL		FALSE
InOperation	BOOL		TRUE
Busy	BOOL		TRUE
Error	BOOL		FALSE
StatusID	DINT		0
UpdateDone	BOOL		FALSE
AdditionalInfo	ViBaseAxBsdAc		
DefaultPositionCompensation	REAL		0.0008
DefaultOutputCompensation	REAL		0.01215
TimestampCount	SINT		28
CalculatedTimestamp	DINT		593753324
Internal	ViBaseInternalTy		

当将任务周期改为 2ms 后，相较 10ms 时输出延时，正好减少了 8ms，说明自动计算了当前任务周期。

Motion::Motion.pvm [Watch] | Motion::Motion [Structured Text] | X20CP3586 [Software]

Name	Type	Scope	Value
MotionMan	MotionMan_type	global	
cmdRun	BOOL		TRUE
MpAxisBasic_0	MpAxisBasic		
MpAxisBasicPar_0	MpAxisBasicPar		
Step	MotionManStep_		stRUN
ViBaseAxisBasedAcquisition_0	ViBaseAxisBase	local	
MpLink	UDINT		98781524
Enable	BOOL		TRUE
AcquisitionParameters	ViBaseAxBsdAc		
AdvancedParameters	ViBaseAxBsdAc		
Update	BOOL		FALSE
PauseAcquisition	BOOL		FALSE
ForceAcquisition	BOOL		FALSE
InOperation	BOOL		TRUE
Busy	BOOL		TRUE
Error	BOOL		FALSE
StatusID	DINT		0
UpdateDone	BOOL		FALSE
AdditionalInfo	ViBaseAxBsdAc		
DefaultPositionCompensation	REAL		0.0008
DefaultOutputCompensation	REAL		0.00415
TimestampCount	SINT		0
CalculatedTimestamp	DINT		0
Internal	ViBaseInternalTy		