Vision Sensor
FH Series
Vision System

3D Robot Vision
AOS Camera Calibration
Operation Guide

FH-505□

FH-SMDA-GS050B

Z451-E1-05

NOTE

- All rights reserved.
- No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, mechanical, electronic, photocopying, recording, or otherwise, without the prior written permission of OMRON.
- No patent liability is assumed with respect to the use of the information contained herein. Moreover, because OMRON is constantly striving to improve its high-quality products, the information contained in this manual is subject to change without notice. Every precaution has been taken in the preparation of this manual. Nevertheless, OMRON assumes no responsibility for errors or omissions.
  Neither is any liability assumed for damages resulting from the use of the information contained in this publication.

Trademarks

- Sysmac and SYSMAC are trademarks or registered trademarks of OMRON Corporation in Japan and other countries for OMRON factory automation products.
- This software is based in part on the work of the Independent JPEG Group.
- Microsoft, Windows, Windows Vista, Excel, and Visual Basic are either registered trademarks or trademarks of Microsoft Corporation in the United States and other countries.
- Intel, Core and Pentium are trademarks of Intel Corporation in the U.S. and/or other countries.
- EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.
- ODVA, CIP, CompoNet, DeviceNet, and EtherNet/IP are trademarks of ODVA.
- The SD, SDHC, microSD, and microSDHC logos are trademarks of SD-3C, LLC.

- QR Code is a registered trademark of DENSO WAVE INCORPORATED.
- MELSEC is a registered trademarks of Mitsubishi Electric Corporation.

Other company names and product names in this document are the trademarks or registered trade-

Copyrights

Microsoft product screen shots used with permission from Microsoft.

CONTENTS

Introduction

Thank you for purchasing the FH Series.
This manual contains information that is necessary to use the FH Series.
Please read this manual and make sure you understand the functionality and performance of the FH Series before you attempt to use it in a control system.
Keep this manual in a safe place where it will be available for reference during operation.

Terms and Conditions Agreement

Warranty, Limitations of Liability

Warranties

● Exclusive Warranty

Omron's exclusive warranty is that the Products will be free from defects in materials and workmanship for a period of twelve months from the date of sale by Omron (or such other period expressed in writing by Omron). Omron disclaims all other warranties, express or implied.

● Limitations

OMRON MAKES NO WARRANTY OR REPRESENTATION, EXPRESS OR IMPLIED, ABOUT NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE OF THE PRODUCTS. BUYER ACKNOWLEDGES THAT IT ALONE HAS DETERMINED THAT THE PRODUCTS WILL SUITABLY MEET THE REQUIREMENTS OF THEIR INTENDED USE.

Omron further disclaims all warranties and responsibility of any type for claims or expenses based on infringement by the Products or otherwise of any intellectual property right.

● Buyer Remedy

Omron's sole obligation hereunder shall be, at Omron's election, to (i) replace (in the form originally shipped with Buyer responsible for labor charges for removal or replacement thereof) the non-complying Product, (ii) repair the non-complying Product, or (iii) repay or credit Buyer an amount equal to the purchase price of the non-complying Product; provided that in no event shall Omron be responsible for warranty, repair, indemnity or any other claims or expenses regarding the Products unless Omron's analysis confirms that the Products were properly handled, stored, installed and maintained and not subject to contamination, abuse, misuse or inappropriate modification. Return of any Products by Buyer must be approved in writing by Omron before shipment. Omron Companies shall not be liable for the suitability or unsuitability or the results from the use of Products in combination with any electrical or electronic components, circuits, system assemblies or any other materials or substances or environments. Any advice, recommendations or information given orally or in writing, are not to be construed as an amendment or addition to the above warranty.

See http://www.omron.com/global/ or contact your Omron representative for published information.

Limitation on Liability; Etc

OMRON COMPANIES SHALL NOT BE LIABLE FOR SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, LOSS OF PROFITS OR PRODUCTION OR COMMERCIAL LOSS IN ANYWAY CONNECTED WITH THE PRODUCTS, WHETHER SUCH CLAIM IS BASED IN CONTRACT, WARRANTY, NEGLIGENCE OR STRICT LIABILITY.

Further, in no event shall liability of Omron Companies exceed the individual price of the Product on which liability is asserted.

Application Considerations

Suitability of Use

Omron Companies shall not be responsible for conformity with any standards, codes or regulations which apply to the combination of the Product in the Buyer's application or use of the Product. At Buyer's request, Omron will provide applicable third party certification documents identifying ratings and limitations of use which apply to the Product. This information by itself is not sufficient for a complete determination of the suitability of the Product in combination with the end product, machine, system, or other application or use. Buyer shall be solely responsible for determining appropriateness of the particular Product with respect to Buyer's application, product or system. Buyer shall take application responsibility in all cases.

NEVER USE THE PRODUCT FOR AN APPLICATION INVOLVING SERIOUS RISK TO LIFE OR PROPERTY OR IN LARGE QUANTITIES WITHOUT ENSURING THAT THE SYSTEM AS A WHOLE HAS BEEN DESIGNED TO ADDRESS THE RISKS, AND THAT THE OMRON PRODUCT(S) IS PROPERLY RATED AND INSTALLED FOR THE INTENDED USE WITHIN THE OVERALL EQUIPMENT OR SYSTEM.

Programmable Products

Omron Companies shall not be responsible for the user's programming of a programmable Product, or any consequence thereof.

Disclaimers

Performance Data

Data presented in Omron Company websites, catalogs and other materials is provided as a guide for the user in determining suitability and does not constitute a warranty. It may represent the result of Omron's test conditions, and the user must correlate it to actual application requirements. Actual performance is subject to the Omron's Warranty and Limitations of Liability.

Change in Specifications

Product specifications and accessories may be changed at any time based on improvements and other reasons. It is our practice to change part numbers when published ratings or features are changed, or when significant construction changes are made. However, some specifications of the Product may be changed without any notice. When in doubt, special part numbers may be assigned to fix or establish key specifications for your application. Please consult with your Omron's representative at any time to confirm actual specifications of purchased Product.

Errors and Omissions

Information presented by Omron Companies has been checked and is believed to be accurate; however, no responsibility is assumed for clerical, typographical or proofreading errors or omissions.

Safety Precautions

For details on Safety Precautions, refer to Safety Precautions in the *Vision System FH Series 3D Robot Vision Application Construction Guide (Cat. No. Z446)*.

Warning

For details on Warning, refer to Warning in the *Vision System FH Series 3D Robot Vision Application Construction Guide (Cat. No. Z446)*.

Precautions for Safe Use

For details on Precautions for Safe Use, refer to Precautions for Safe Use in the *Vision System FH Series 3D Robot Vision Application Construction Guide (Cat. No. Z446)*.

Precautions for Correct Use

For details on Precautions for Correct Use, refer to Precautions for Correct Use in the *Vision System FH Series 3D Robot Vision Application Construction Guide (Cat. No. Z446)*.

Regulations and Standards

For details on Regulations and Standards, refer to Regulations and Standards in the *Vision System FH Series 3D Robot Vision Application Construction Guide (Cat. No. Z446)*.

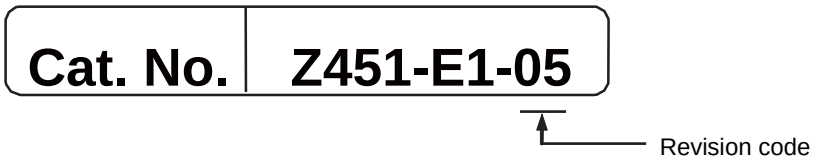
Related Manuals

Name of Manual	Cat. No.	Model	Purpose	Contents
Vision System FH Instruction Sheet	3102269-4	FH-2□□□ FH-2□□□-□□ FH-5□□□ FH-5□□□-□□	To confirm the safety and usage precautions of the Vision System FH series Sensor Controller.	Describes the definitions of basic terms, meaning of signal words, and precautions for correct use of FH series in the manual.
Vision System FH Instruction Sheet	3648743-1	FH-2□□2 FH-2□□2-□□ FH-5□□2 FH-5□□2-□□	To confirm the safety and usage precautions of the Vision System FH series Sensor Controller.	Describes the definitions of basic terms, meaning of signal words, and precautions for correct use of FH series in the manual.
3D Vision Sensor FH-SMDA Instruction Sheet	3290410-0	FH-SMDA-GS050B	To confirm the safety and usage precautions of the 3D Vision Sensor FH-SMDA.	Describes the definitions of basic terms, the meaning of signal words, and precautions for correct use of 3D Vision Sensor FH-SMDA in the manual.
Vision System FH series 3D Robot Vision Application Construction Guide	Z446	FH-505□ FH-SMDA-GS050B	When User want to know about the FH series 3D robot vision system.	Describes the soft functions, setup, and operations to use FH series 3D robot vision system.
Vision System FH series Hardware Setup Manual for 3D Robot Vision	Z436		When User want to know about the Hard-ware specifications or to setup the Sensor Controller of the FH series 3D robot vision system.	Describes FH series 3D robot vision system specifications, dimensions, part names, I/O information, installation information, and wiring information.
Vision System FH series Processing Item Function Reference Manual for 3D Robot Vision	Z445		When User confirm the details of each processing items at the create the measurement flow or operate it.	Describes the software functions, settings, and operations for using FH series 3D robot vision system.
Vision System FH series Robot Connection Guide OMRON TM Series Edition	Z447		When connecting FH series 3D robot vision system to the robot	Describes communication settings and sample programs for picking applications that connect FH series 3D robot vision system to the robot.
Vision System FH series Robot Connection Guide OMRON Viper Series Edition	Z448			
Vision System FH series Robot Connection Guide FANUC Corporation Edition	Z449			

Vision System FH series Robot Connection Guide DENSO WAVE Incorporated Edition	Z458			
Vision System FH series Robot Connection Guide ABB Edition	Z459			
Vision System FH series Robot Connection Guide YASKAWA Electric Corporation Edition	Z460			
Vision System FH series Robot Connection Guide UNIVERSAL ROBOTS Edition	Z463			
Vision System FH series Robot Connection Guide NACHI-FUJIKOSHI CORP. Edition	Z464			

Revision History

A manual revision code appears as a suffix to the catalog number on the front and back covers of the manual.



Rev. Code	Rev. Date	Revision Contents
01	Feb. 2021	Original product
02	Jun. 2021	Add the instruction of automatic calibration which manufactured by DENSO, ABB and YASKAWA.
03	Aug. 2021	Add the instruction of automatic calibration which manufactured by UNIVERSAL ROBOTS.
04	Dec. 2021	Add the instruction of automatic calibration which manufactured by NACHI-FUJIKOSHI CORP. Nachi.
05	Jun. 2024	Add the instruction of automatic calibration which manufactured by OMRON TM S series. Additional support for FH-5051 and FH-5052. Revisions for update camera cables model. Added FZ-MEM16G.

1. Overview

This manual describes how to perform camera calibration of the FH series vision system FH-SMDA-GS050B 3D vision sensor (hereinafter referred to as the 3D vision sensor). Some descriptions in this manual assume that the FH series vision system (hereinafter referred to as the vision system) is connected to the robot controller. Refer to the *Vision System FH series 3D Robot Vision Application Construction Guide* (Cat. No. Z446) and *Vision System FH series Robot Connection Guide* listed in *Related Manuals* in this manual.

2. System Configuration

For the system configuration, refer to the following manual.
Please refer to *System Configuration* in the *Vision System FH series 3D Robot Vision Application Construction Guide* (Cat. No. Z446) for details.

3. AOS Camera Calibration

This section describes when and how to execute AOS camera calibration.

3.1. Overview

The geometric positional relationship inside the 3D vision sensor changes due to temperature changes inside and outside the sensor, which may cause an error in the measurement results. To maintain stable measurement, it is necessary to implement regular calibration, and calibrate the geometric positional relationship between the 3D measurement lighting and imaging units of the 3D vision sensor. In this manual, this calibration is called “AOS camera calibration.”

You can execute AOS camera calibration by using the 3D vision sensor to capture the images of the camera calibration target from two viewpoints, i.e., the near imaging position and the far imaging position.

3.2. Executing Timing

Execute AOS camera calibration in the following timing.

- When you install the 3D vision sensor for the first time
- When you change the installation location of the 3D vision sensor
- When one month has passed since you executed AOS camera calibration last time

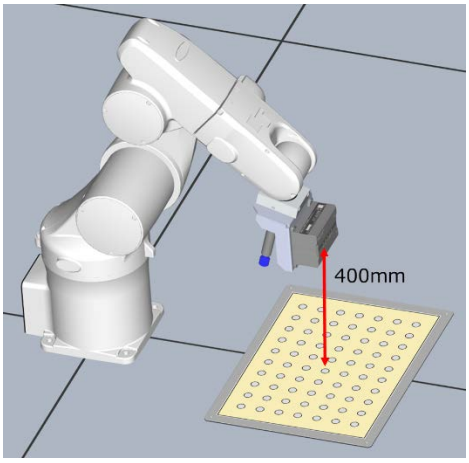
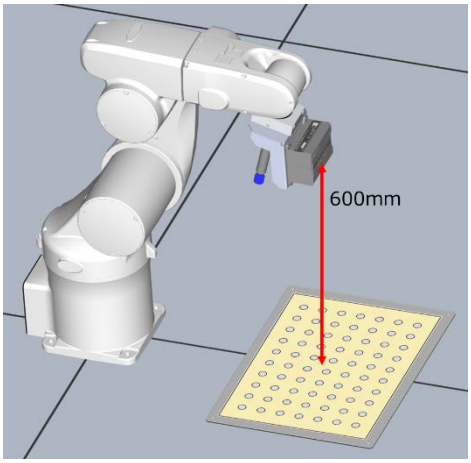
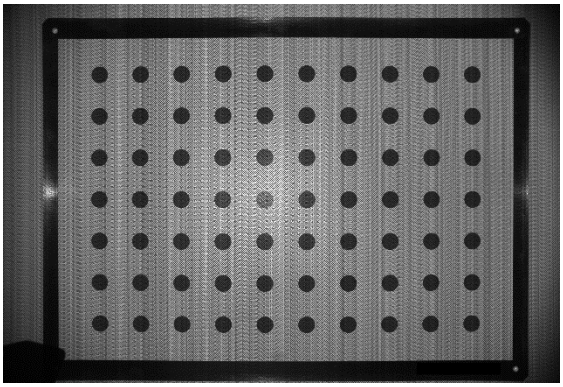
3.3. Calibration Methods

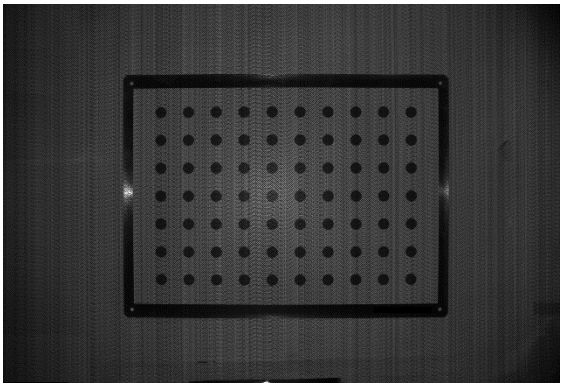
You can execute AOS camera calibration in the following three ways.

Chapter	Method	Description
3.5	Manual Calibration	The 3D vision sensor or the camera calibration target.is moved manually in order to execute AOS camera calibration. If you use the 3D vision sensor alone for workpiece evaluation, etc., use this method to execute AOS camera calibration.
3.6	Automatic Calibration (Vision Master)	The vision system issues motion instructions to the robot to move the 3D vision sensor attached to the robot in order to execute AOS camera calibration.
3.7	Automatic Calibration (Robot Master)	The robot issues motion instructions to the vision system to move the 3D vision sensor attached to the robot in order to execute AOS camera calibration. This method allows AOS camera calibration to be executed automatically from the robot program.

3.4. Terminology

Terms related to AOS camera calibration are described below.

Term	Description
Nearest imaging position	The camera position where images of the camera calibration target can be captured at a distance of +400 mm in the direction of the optical axis of the 3D Vision Sensor (i.e., Z axis). 
Far imaging position	The camera position where images of the camera calibration target can be captured at a distance of +600 mm in the direction of the optical axis of the 3D Vision Sensor (i.e., Z axis). 
Near image	A measurement image at the near imaging position Image example: 

Far image	A measurement image at the far imaging position Image example: 

3.5. Manual Calibration

Set up and execute a calibration by manually moving the 3D vision sensor. For the input image, a dedicated 3D vision sensor and a measurement image from the **Camera Image Input AOS** processing item are required. To execute a calibration, you need two measurement images: the near image that is captured at a distance of 400 mm and the far image captured at a distance of 600 mm from the 3D vision sensor.
It is a requirement that you complete up to Chapter 7 *3D Sensor Software Installation* of the *Vision System FH series 3D Robot Vision Application Construction Guide* (Cat. No. Z446).
This section assumes that the Layout 0 window is used.

Precautions for Correct Use

- Set up the 3D vision sensor and the camera calibration target so that the distance between them can be changed within a range of 400 to 600 mm.

Use the following work flow to configure the settings.

3.5.1	Preparation	Switch to the scene to configure, and then install the 3D vision sensor and the camera calibration target.
▼		
3.5.2	Configuring the Camera Image Input AOS	Set up the 3D vision sensor so that it holds the latest calibration data. Or adjust the Camera Image Input AOS so that the 3D vision sensor can measure the camera calibration target properly.
▼		
3.5.3	Executing a Geometric Variation Check	Diagnose the current calibration data to check whether the 3D vision sensor requires a calibration.
▼		
3.5.4	Executing a Calibration	If the check result shows *Need calib.* in *3.5.3 Executing a Geometric Variation Check*, execute a calibration. Register the near and far images in the processing item and execute a calibration.
▼		
3.5.5	Reflecting the Results in the Camera Calibration AOS	The image sensor, have the modified calibration data reflected in the 3D vision sensor.

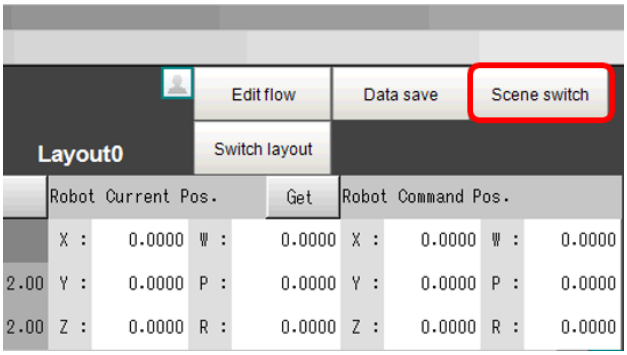
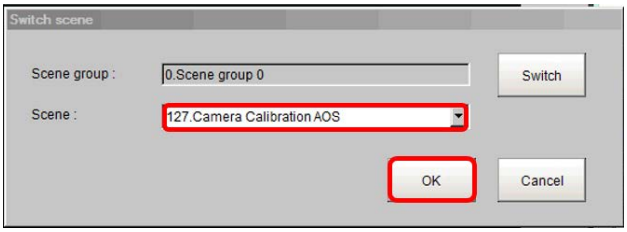
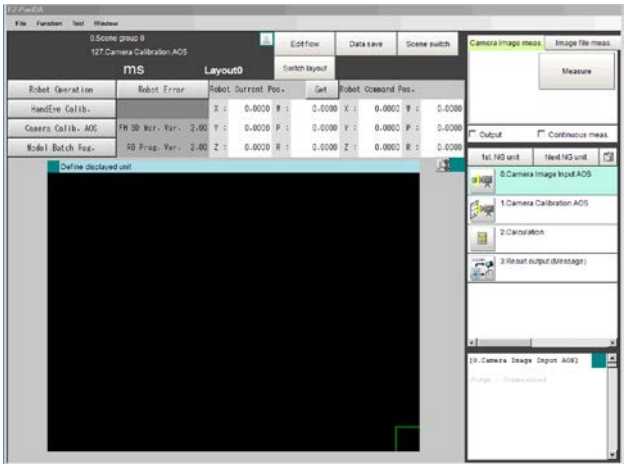
3.5.1. Preparation

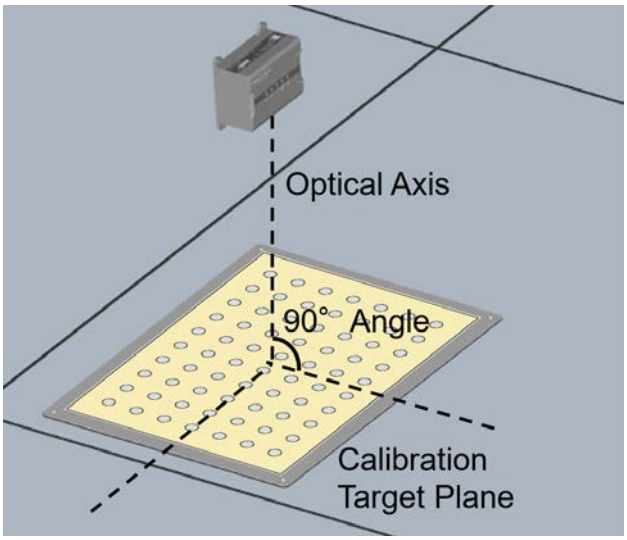
To reliably detect the camera calibration target, make the following software and hardware preparation in advance.

Software preparation: Switch to the scene to configure.

Loading a sensor controller project sets the scene named *Pick on hand_6 axis_XXX* for Scene No. 0 as a sample scene for the pick application.

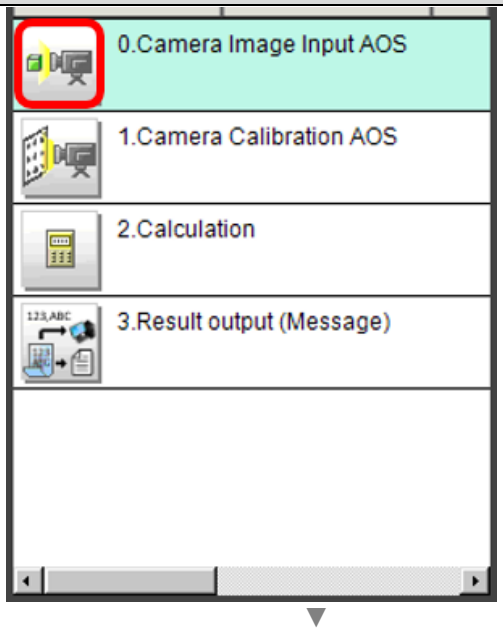
Hardware preparation: Install the 3D vision sensor and the camera calibration target.

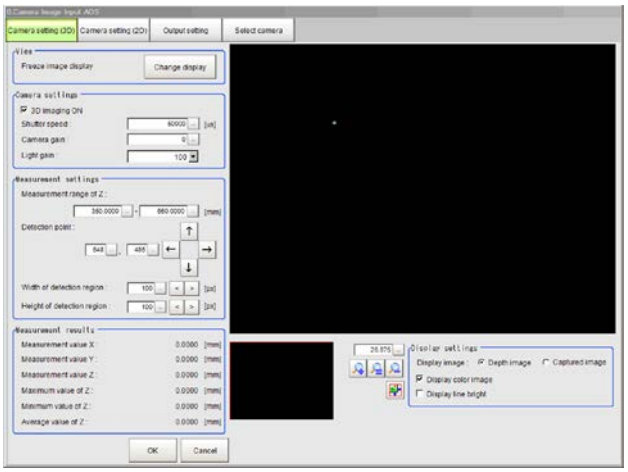
Step	Description	Window image, diagram
1	Software preparation: In the main window of the vision system, click the **Scene switch** button.	
	Select the scene No. 127 Camera Calibration AOS. Click **OK** to switch the scene.	 ▼ 

2	Hardware preparation: Place the camera calibration target so that the following conditions are met. • It is seen that the 3D vision sensor is placed directly above the center of the camera calibration target. • It is seen that the optical axis of the 3D vision sensor forms an angle of 90 ± 5 degrees with the plane of the camera calibration target. There are two methods of movement, for example. 1. Move the 3D vision sensor attached to the camera stand. 2. Move the camera calibration target with the 3D vision sensor fixed.	

3.5.2. Configuring the Camera Image Input AOS

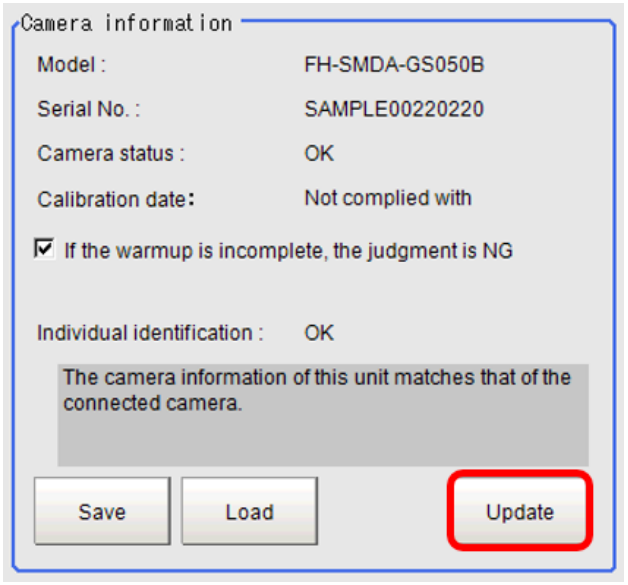
Set the imaging position, shutter speed, camera gain, and measurement range of Z so that the camera calibration target can be measured properly.
Make sure that the warmup of the 3D vision sensor is completed before you perform the procedure below.

Step	Description	Window image, diagram
1	In the main window, click the **0. Camera Image Input AOS** icon to open the setting window.	



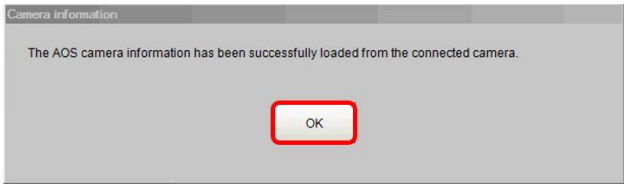
2 Select the **Select camera** tab.

Click the **Update** button to update the calibration data held in the **Camera Image Input AOS** processing item.

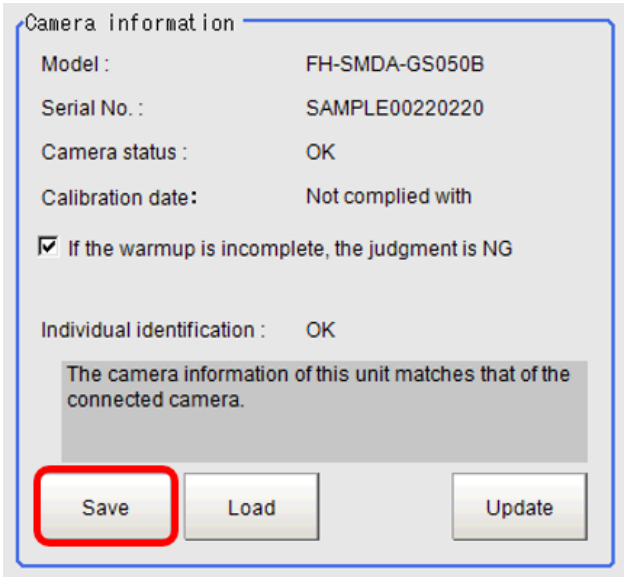


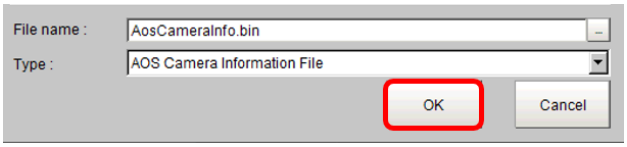
▼

▼



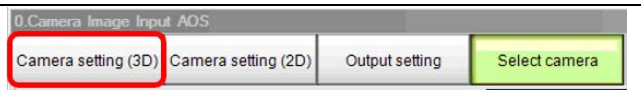
To back up the current calibration data, click the **Save** button to save the AOS camera information.

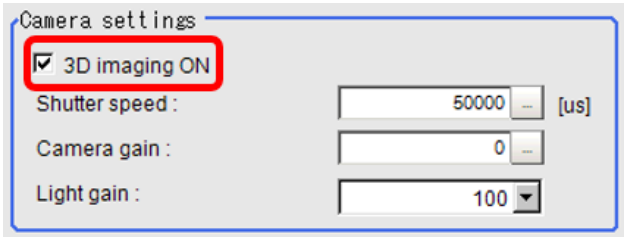




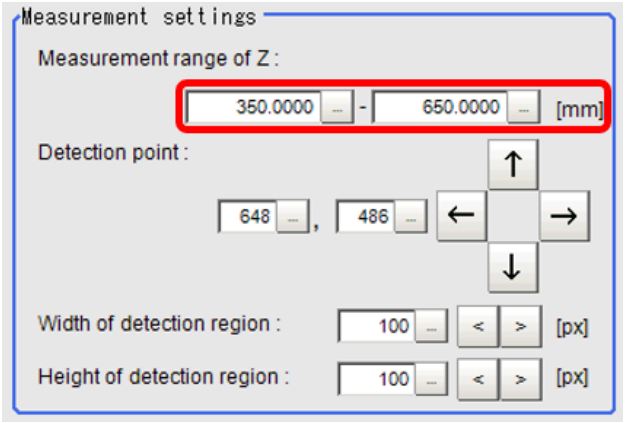
3 Select the **Camera setting (3D)** tab page.

Confirm that the **3D imaging ON** check box is selected.

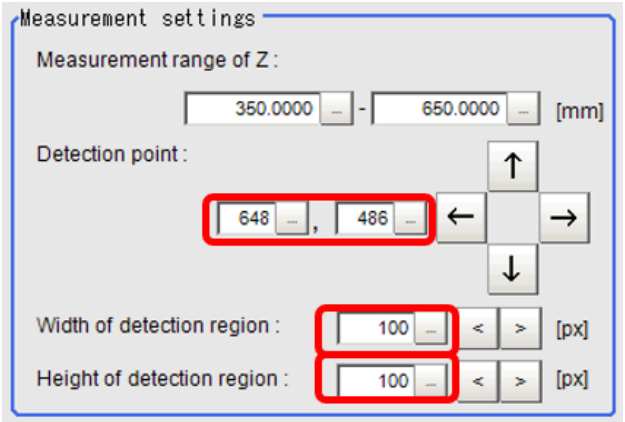




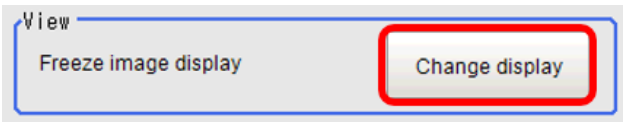
Confirm that **Measurement range of Z** is set to 350 to 650 mm.



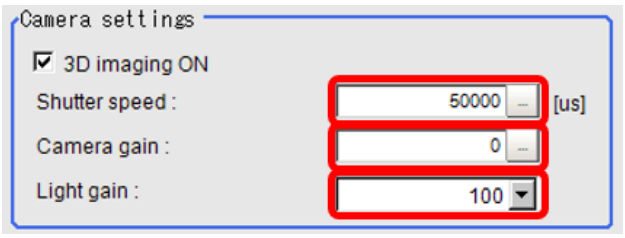
Confirm that **Detection point** is set to the center of the image.
Confirm that **Width of detection region** and **Height of detection region** are set to *100*.



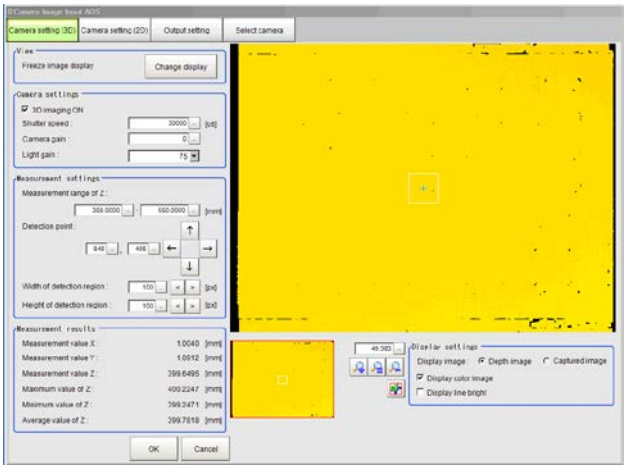
4 Click the **Change display** button to display the through image.



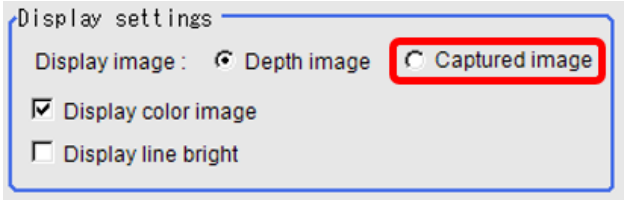
While viewing the through image, set **Shutter speed**, **Camera gain**, and **Light gain** so that the exposure is appropriate.

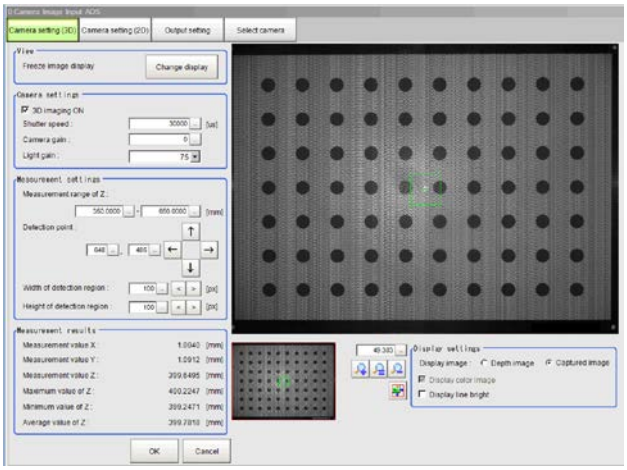


If the exposure is appropriate, the distance image of the camera calibration target is displayed as shown in the figure on the right.
On the camera calibration target, the boundaries of dots may have some lost point clouds.

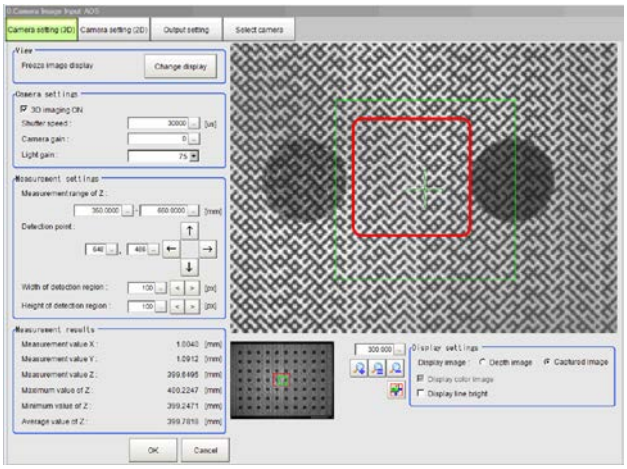


Select the **Captured image** option to display the captured image with a pattern.
Confirm that the calibration target is captured.

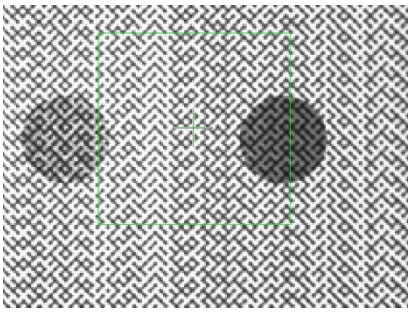




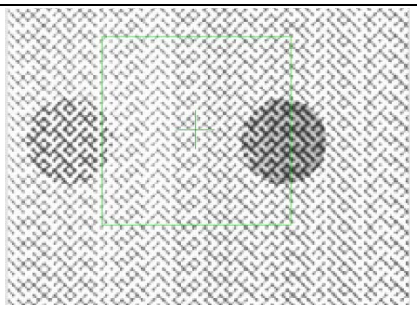
With the detection point placed in a white area on the plate as shown in the figure on the right, adjust the settings so that the contrast between the pattern and the background is high.



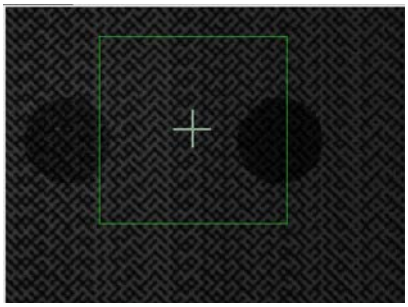
Good image example:



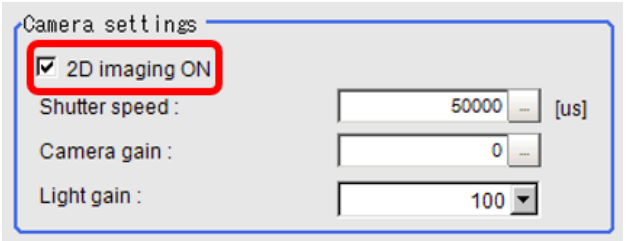
Bad image example 1 (Too bright):



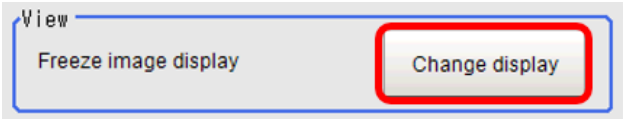
Bad image example 2 (Too dark):



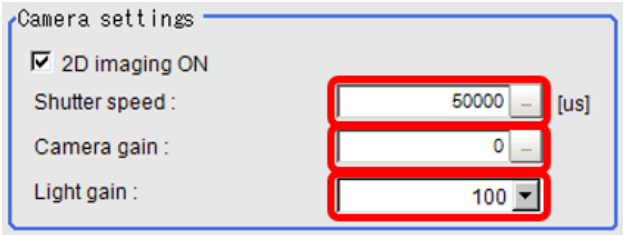
5 Select the **Camera setting (2D)** tab page.
Confirm that the **2D imaging ON** check box is selected.



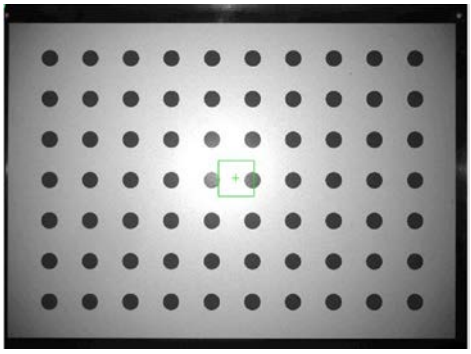
Click the **Change display** button to display the through image.



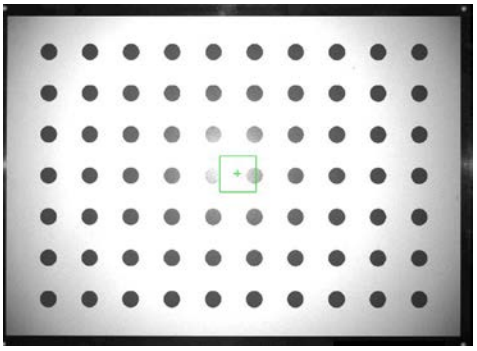
While viewing the through image, set **Shutter speed**, **Camera gain**, and **Light gain** so that the dot pattern is visible without halation on the camera calibration target image.



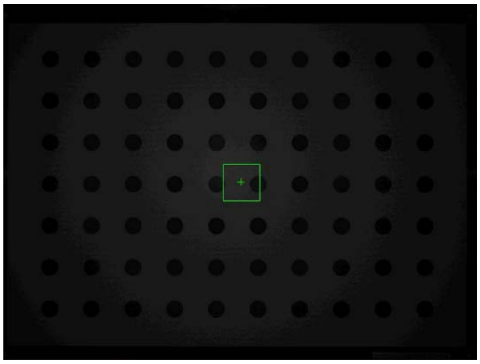
Good image example:



Bad image example 1 (Too bright):

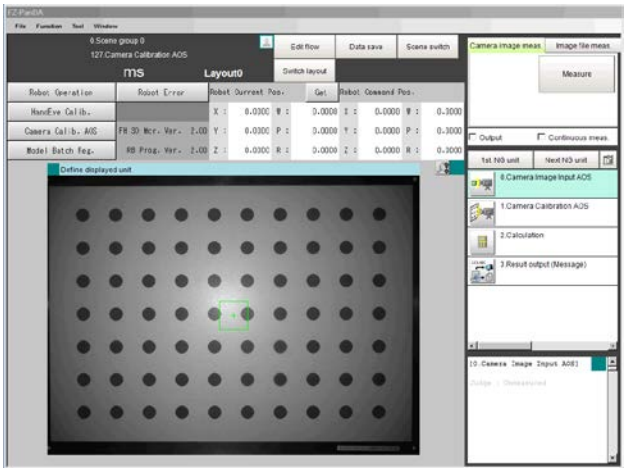


Bad image example 2 (Too dark):



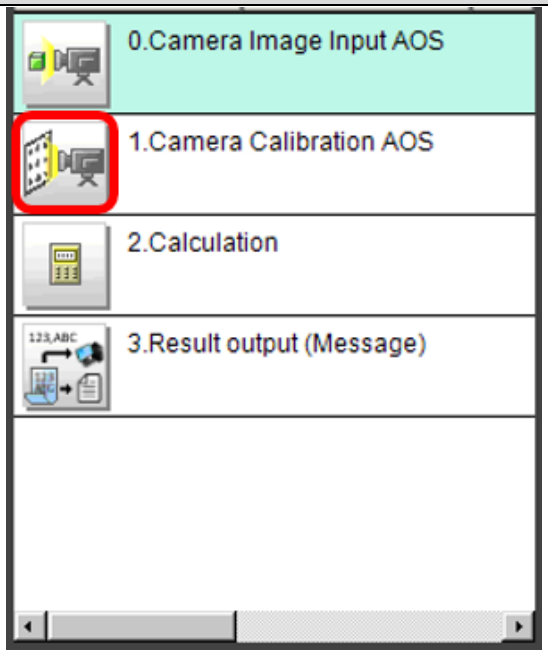
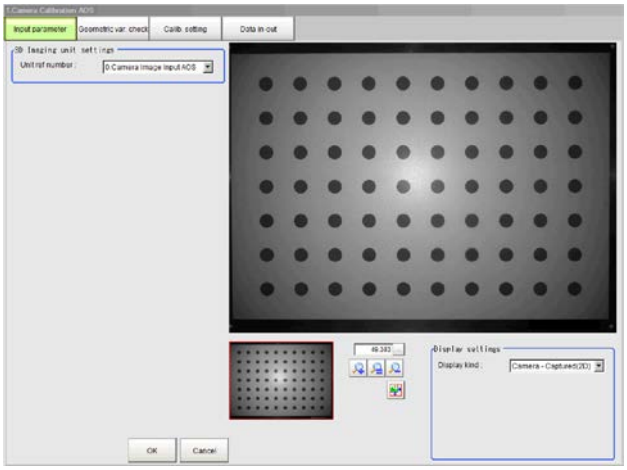
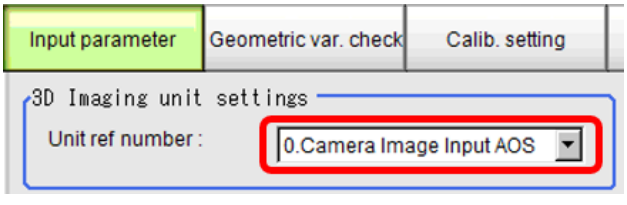
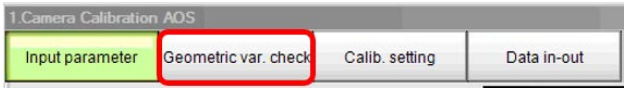
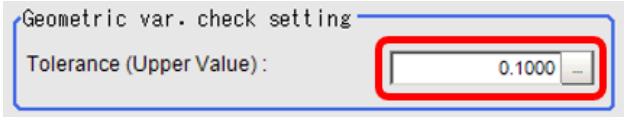
6 If there are no more changes, click the **OK** button to close the setting window.

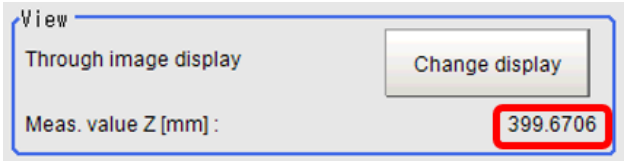
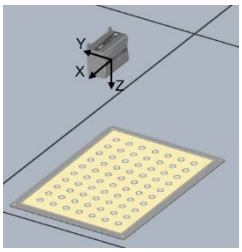
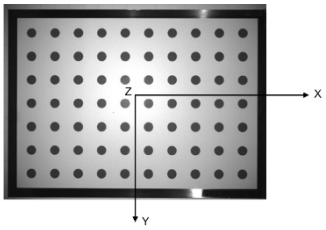
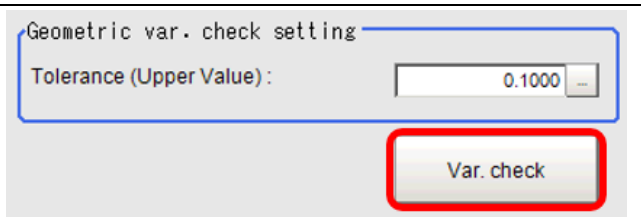
▼



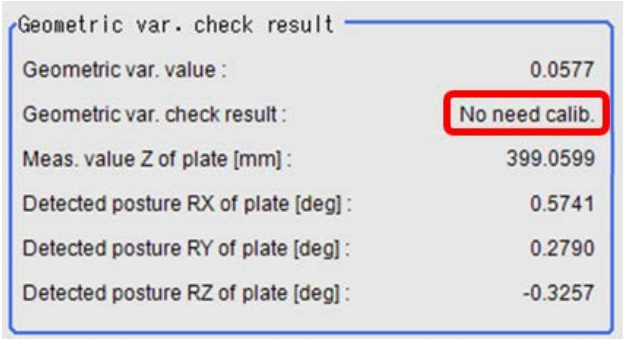
3.5.3. Executing a Geometric Variation Check

Execute a geometric variation check. If the geometric variation value is within the allowable range, there is no need to perform the calibration steps in 3.5.4 and later.

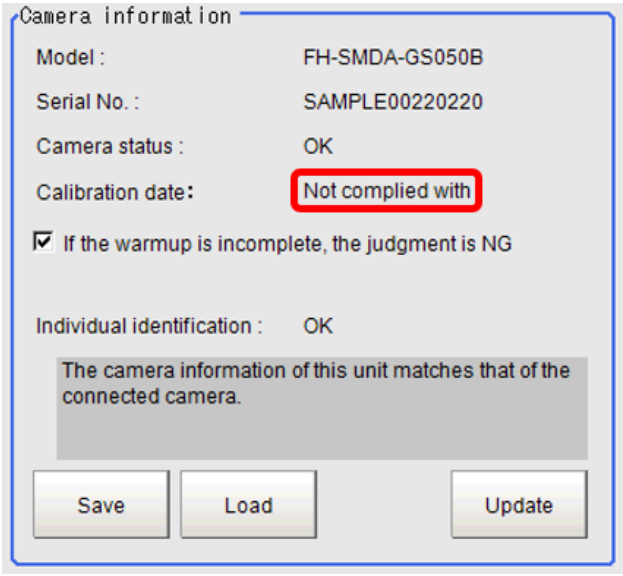
Step	Description	Window image, diagram
1	Open the **Camera Image Input AOS** processing item.	
		
	In the **Input parameter** tab page, confirm that **Unit ref number** matches the unit number of the **Camera Image Input AOS** processing item set in 3.5.2.	
2	Open the **Geometric var. check** tab page.	
	Confirm that **Tolerance (Upper Value)** is set to the recommended value (0.1).	

3	Click the **Change display** button to display the through image. Confirm that **Meas. value Z** is within 400 ± 5 mm and visually check that the optical axis of the 3D vision sensor is perpendicular to the plane of the plate. Refer to the figure on the right for the posture of the camera calibration target. After you check the above, click the **Change display** button once again to return to the freeze image display.	    
4	Click the **Var. check** button to display the results of the geometric variation check in the **Geometric var. check result** area.	

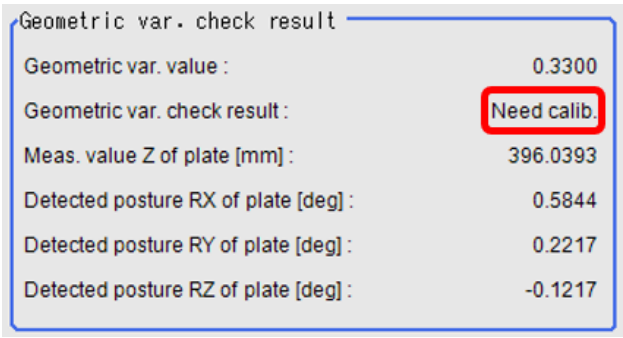
If **Geometric var. check result** shows *No need calib.*, AOS camera calibration is unnecessary. This means that there is no need to perform *3.5.4 Executing a Calibration* and later procedures.



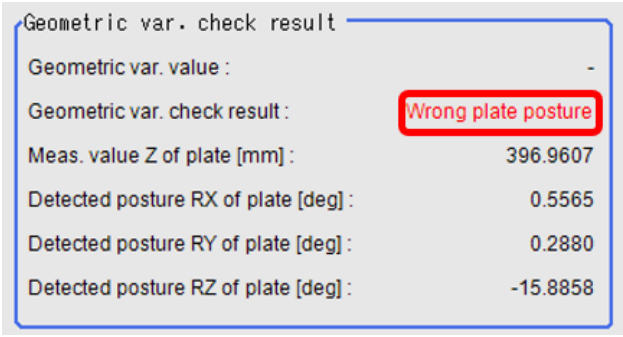
In this case, in the **Select camera** tab page of the **Camera Image Input AOS** processing item, **Calibration date** remains *Not complied with*. However, it is not a problem.



If **Geometric var. check result** shows *Need calib.*, the 3D vision sensor requires a calibration. Follow the steps in *3.5.4 Executing a Calibration* to set up the vision system.

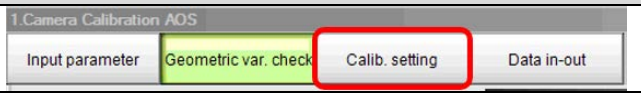
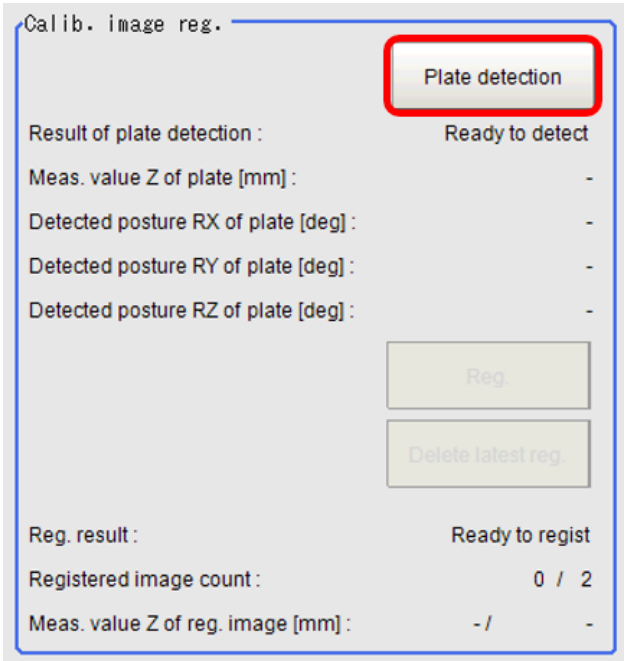


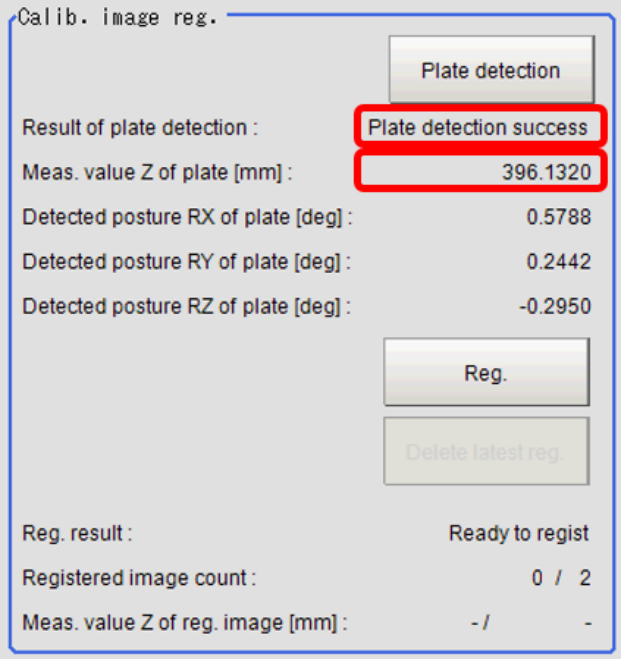
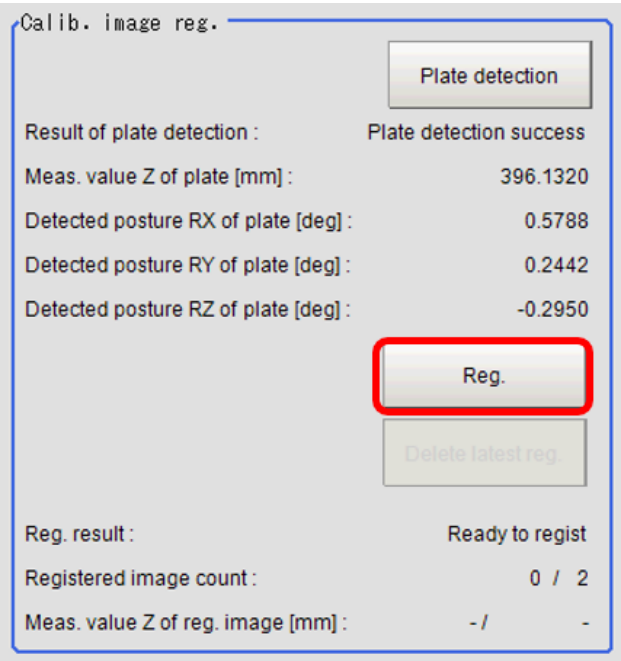
If **Geometric var. check result** shows *Plate detection failure*, *Wrong plate posture*, etc., and the variation check fails, adjust the position of the camera calibration target or the image input settings in the Camera Image Input AOS.



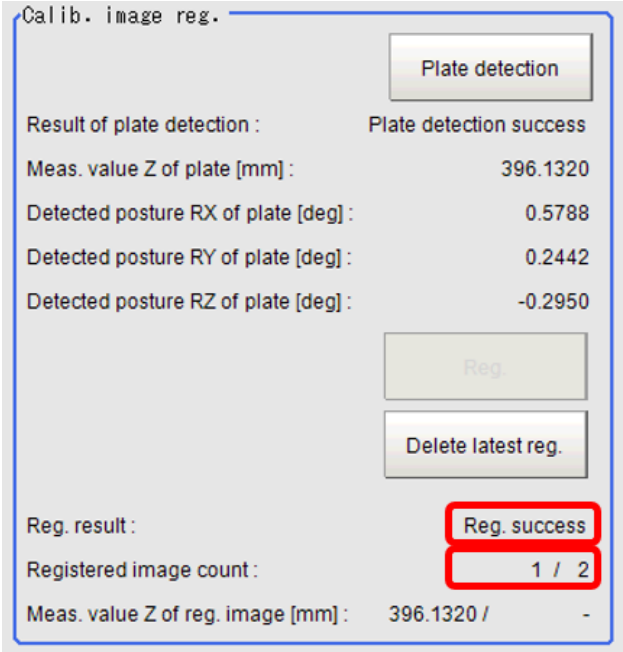
3.5.4. Executing a Calibration

If the check result shows *Need calib.* in *3.5.3 Executing a Geometric Variation Check*, execute a calibration. Follow the steps below to register the near and far images, and execute a calibration.

Step	Description	Window image, diagram
1	Select the **Calib. setting** tab page.	
2	Click the **Change display** button to display the through image. Confirm that **Meas. value Z** is within 400 ± 5 mm and visually check that the optical axis of the 3D vision sensor is perpendicular to the plane of the plate. After you check the above, click the **Change display** button once again to return to the freeze image display.	  
3	Click the **Plate detection** button to detect the camera calibration target.	

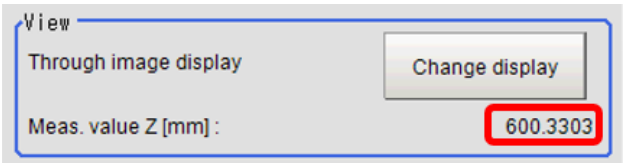
	Confirm that **Result of plate detection** shows *Plate detection success*. Confirm that **Meas. value Z of plate** shows a value around 400 mm.	
4	Click the **Reg.** button to register the captured image.	

Confirm that **Reg. result** shows *Reg. success* and **Registered image count** shows *1 / 2*.



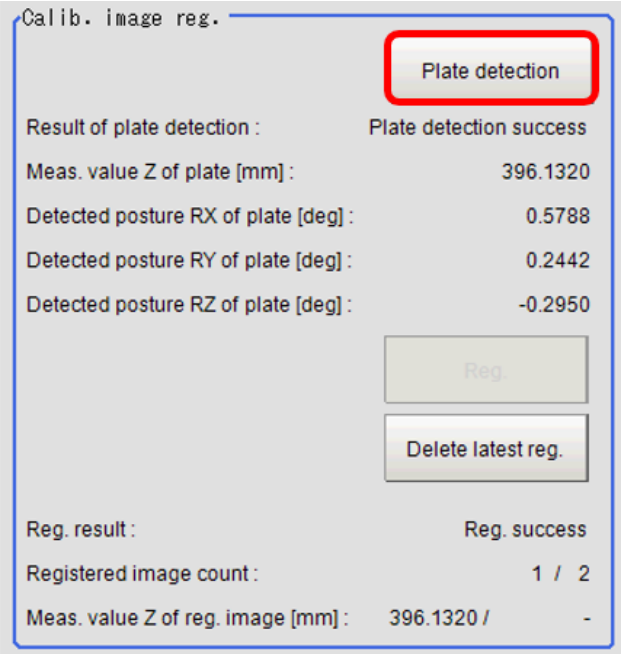
5 Click the **Change display** button to display the through image. Move the 3D vision sensor or the camera calibration target so that the distance from the 3D vision sensor to the camera calibration target is 600 mm.

Confirm that **Meas. value Z** is around 600 mm and visually check that the optical axis of the 3D vision sensor is perpendicular to the plane of the plate.

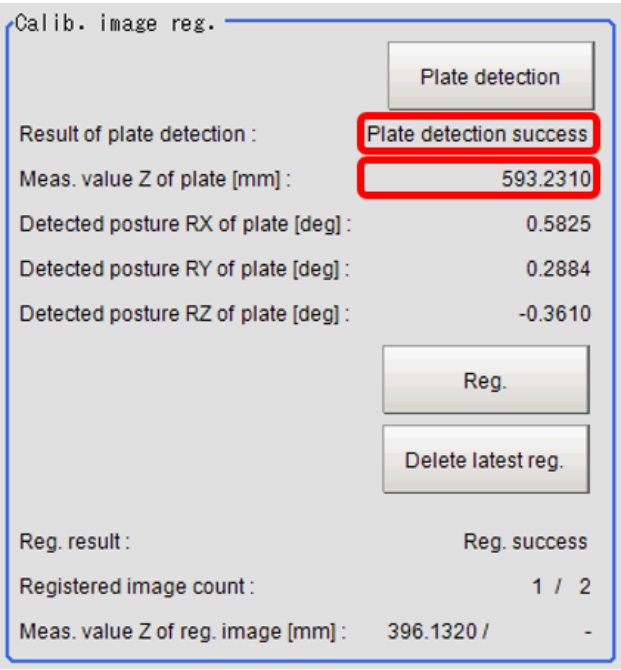


After you check the above, click the **Change display** button once again to return to the freeze image display.

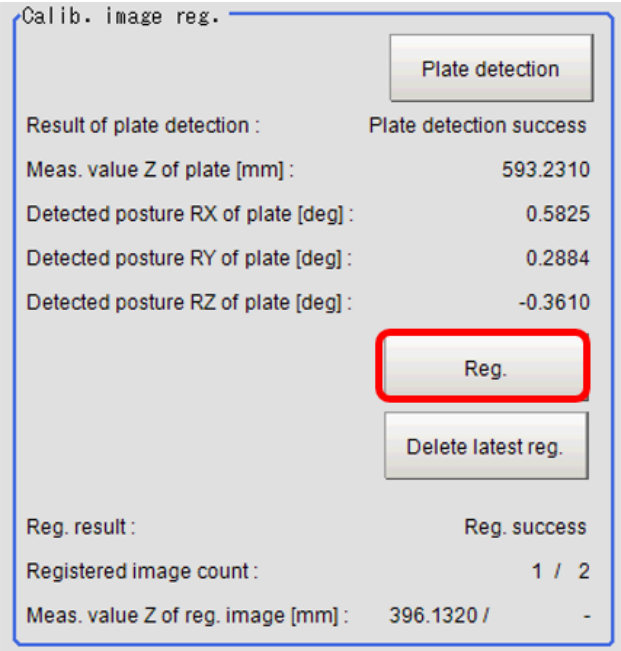
6 Click the **Plate detection** button to detect the camera calibration target.



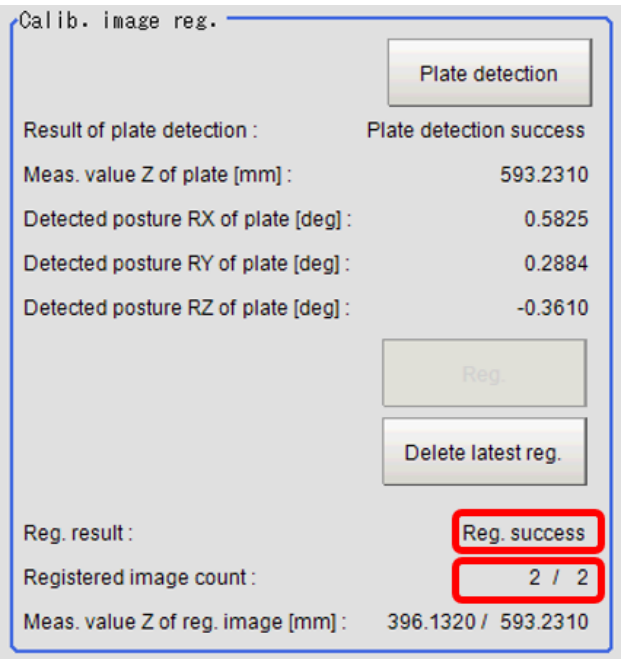
Confirm that the detection result of the camera calibration target shows *Plate detection success*. Confirm that **Meas. value Z of plate** shows a value around 600 mm.



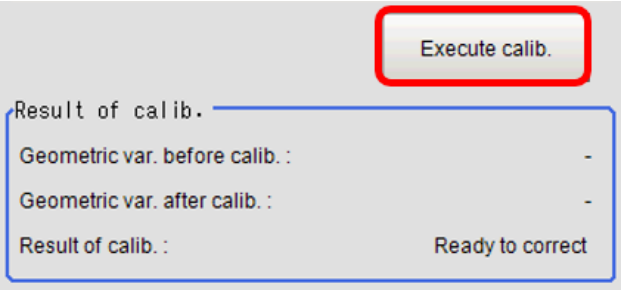
7 Click the **Reg.** button.

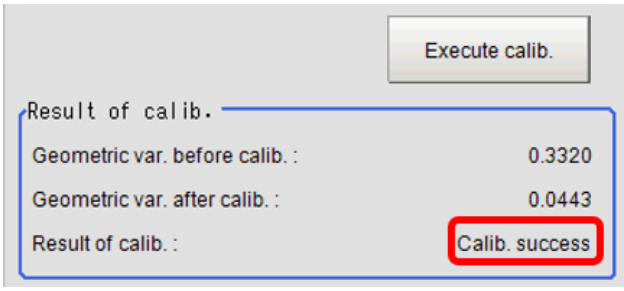


Confirm that **Reg. result** shows *Reg. success* and **Registered image count** shows *2 / 2*.



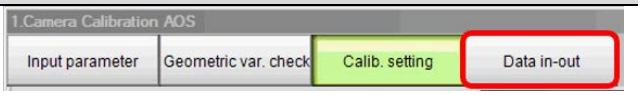
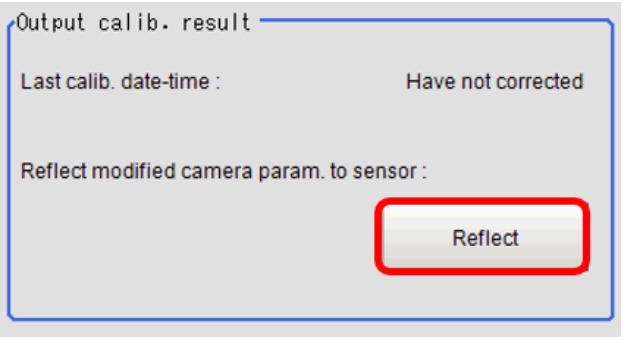
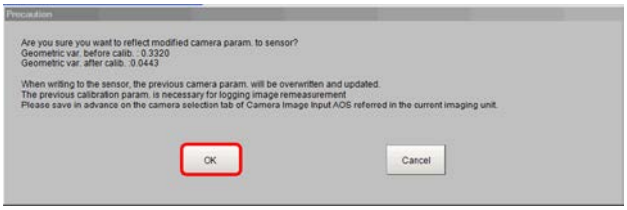
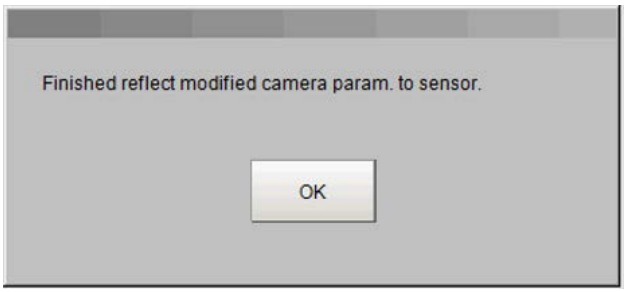
8 Click the **Execute calib.** button to execute calibration with the registered images.



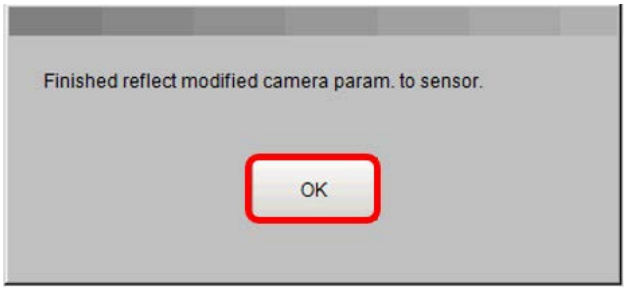
	Confirm that **Result of calib.** shows *Calib. success*.	

3.5.5. Reflecting the Results in the Camera Calibration AOS

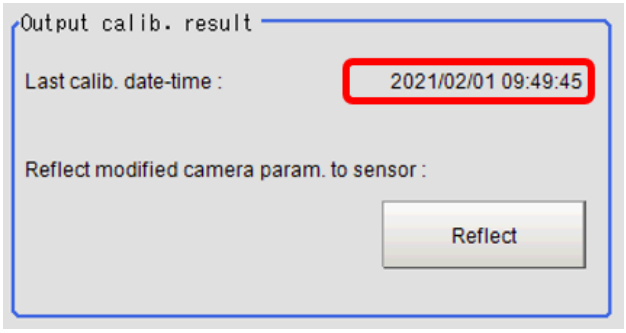
Follow the steps below to have the modified calibration data reflected in the 3D vision sensor.

Step	Description	Window image, diagram
1	Select the **Data in-out** tab page.	
2	Click the **Reflect** button.	
	Click the **OK** button in the writing warning dialog box. The message **Finished reflect modified camera param. to sensor** is displayed to indicate that the writing is successful.	 ▼ 

Click the **OK** button to close the writing successful message dialog box.

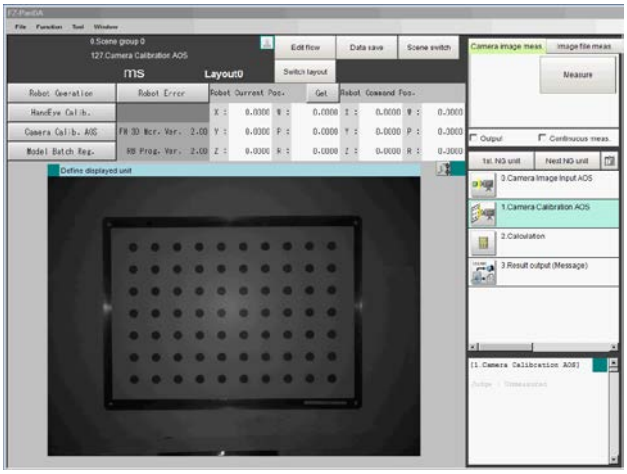


Confirm that the time shown in **Last calib. date-time** matches the time when the calibration was executed.
The execution date and time of calibration is the internal date and time of the vision system that indicates when the **Execute calib.** button was clicked.

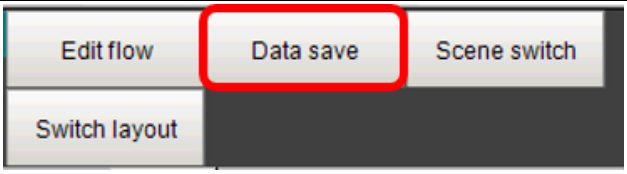


Click the **OK** button to close the **Camera Image Input AOS** processing item setting window.

▼



3 In the main window, click the **Data save** button.



In the confirmation dialog box, click the **OK** button.
This completes the AOS camera calibration.

3.5.6. Troubleshooting for AOS Camera Calibration

If AOS camera calibration fails, refer to *Key Points for Test Measurement and Adjustment* for each processing item in the *Vision System FH series Processing Item Function Reference Manual for 3D Robot Vision* (Cat. No. Z445).

3.6. Automatic Calibration (Vision Master)

With the 3D vision sensor attached to the robot hand, send control instructions from the vision system to the robot. As the robot moves, the 3D vision sensor moves and automatically sets up and execute a calibration.

It is a requirement that you complete up to Chapter 7 *3D Sensor Software Installation* of the *Vision System FH series 3D Robot Vision Application Construction Guide* (Cat. No. Z446).

This section assumes that the Layout 2 window is used.

⚠Warning

- Make sure that you always use a sensor controller project that matches the robot type. If the type is mismatched, the robot may operate in an unexpected manner.
- Make sure that the robot is operated by personnel who have completed special training for safety and health.
- Make sure that you can stop the operation of the robot at any time by pressing the emergency stop button.

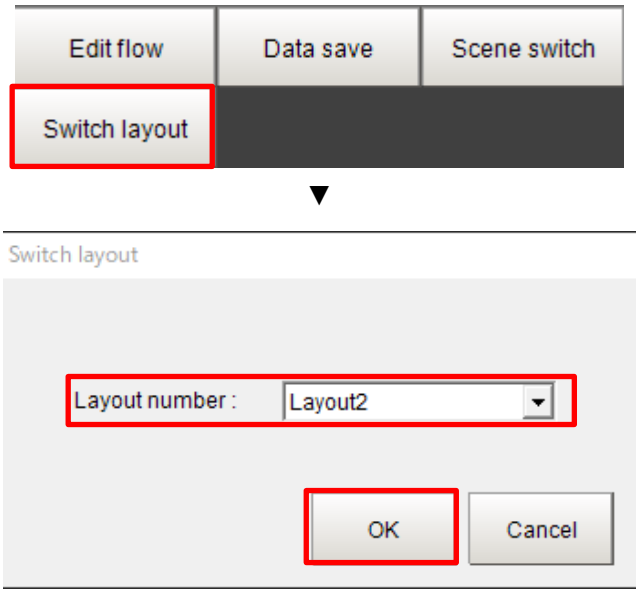
Precautions for Correct Use

- For robot-related settings, refer to the *Vision System FH series Robot Connection Guide* corresponding to each robot in *Related Manuals*.
- Set the position of the robot hand so that the distance between the 3D vision sensor and the camera calibration target can be changed within a range of 400 to 600 mm.
- This wizard configures only the minimum required settings. To configure advanced settings, refer to the following document listed in *Related Manuals* in this manual. *Vision System FH series Processing Item Function Reference Manual for 3D Robot Vision* (Cat. No. Z445)

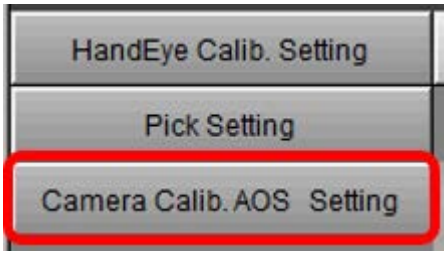
Additional Information

If the wizard does not work as described, you may not have followed the procedure correctly. Redo from the beginning of the wizard.

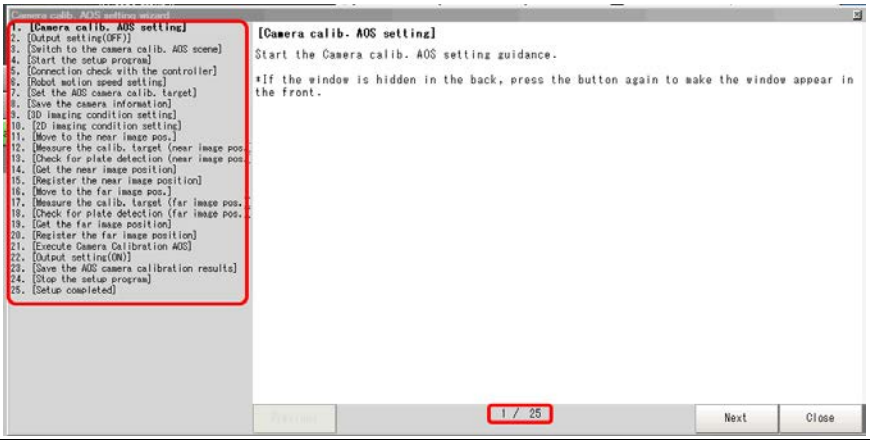
1 In the main window, click the **Switch layout** button and switch to *Layout2*.



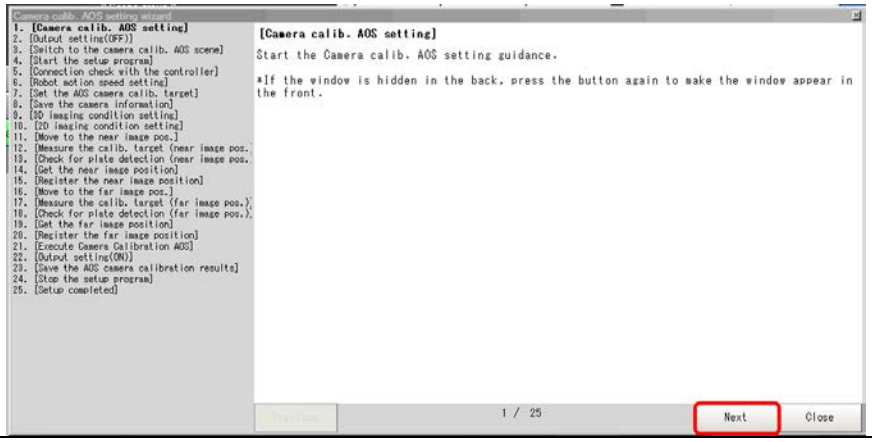
2 In the main window, click the **Camera Calib. AOS Setting** button to start the wizard.



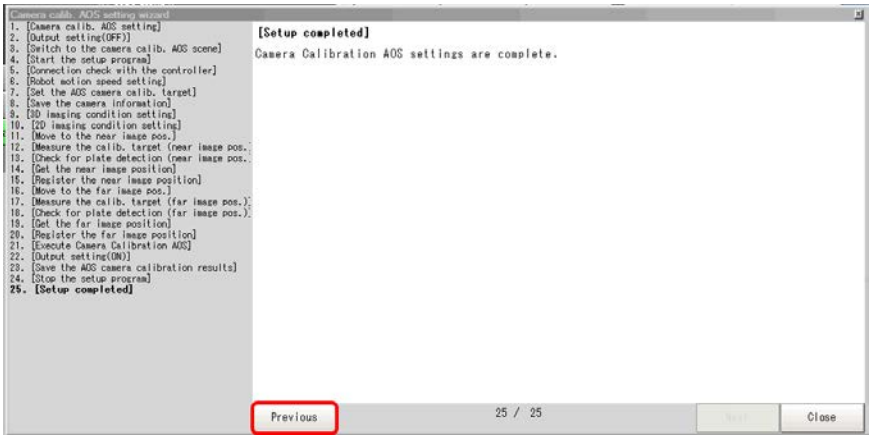
3 Read the wizard instructions and configure the settings on pages 1/25 to 25/25 of the wizard.



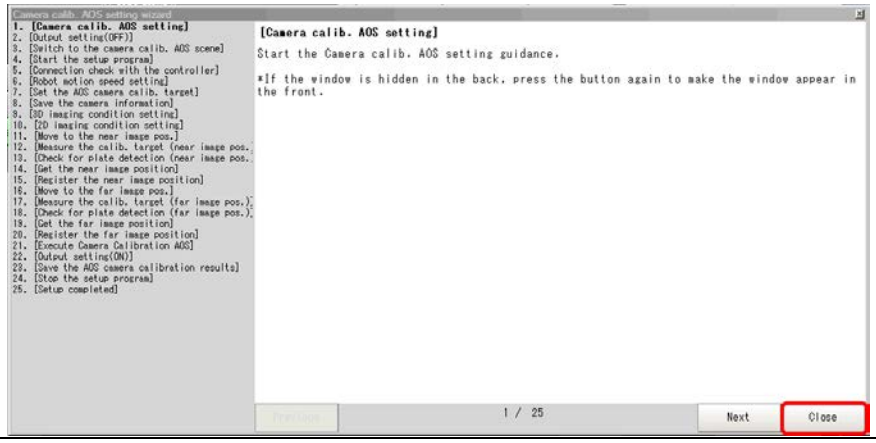
To configure the settings on the next page, click the **Next** button.



To return to the settings on the previous page, click the **Previous** button.



To close the wizard, click the **Close** button.



4 This completes the Camera Calibration AOS setting.

Click the **Close** button to close the wizard.

3.7. Automatic Calibration (Robot Master)

This section describes an example of designing a robot program that automatically executes AOS camera calibration by control instructions from the robot, using a sample program for AOS camera calibration (fhsample_calaos).
To execute AOS camera calibration with the method described in this section, it is required that you set up the vision sensor according to the method described in *3.6. Automatic Calibration (Vision Master)* and that you place the camera calibration target in the position where you placed it during the setup.
The robot program consists of the following steps. If you need detailed information on each function, refer to the *Vision System FH series Robot Connection Guide* corresponding to each robot manufacturer.

1	Connecting the Vision Sensor to the Robot Controller
▼	
2	Switching Scenes on the Vision Sensor
▼	
3	Acquiring the Camera Status from the 3D Vision Sensor
▼	
4	Acquiring the Calibration Date/Time Comparison Result from the 3D Vision Sensor
▼	
5	Changing the Calibration Mode of AOS Camera Calibration
▼	
6	Acquiring the Near Imaging Position
▼	
7	Moving the Robot to the Near Imaging Position
▼	
8	Executing a Measurement (Near Imaging Position)
▼	
9	Acquiring the Far Imaging Position
▼	
10	Moving the Robot to the Far Imaging Position
▼	
11	Executing a Measurement (Far Imaging Position)
▼	
12	Ending the Program (Normal End)
▼	
13	Ending the Program (Error End)

Precautions for Correct Use

• The implementation procedure of the robot program described in this section is for reference only. Design, implement, and test the robot program for actual operation according to your environment.
• In the main window of the vision system, or in the Layout setup, make sure that **Output** is ON for the current layout. If it is OFF, the vision system does not output measurement values.

3.7.1. For OMRON Viper Series

Connecting Vision Sensor to Robot Controller

The following describes a program that connects the vision system to the robot controller.

1 Declare internal variables.

* Description omitted (Refer to the source code.)

2 Execute the global variable initialization function (fhdefglobal).

```
CALL fhdefglobal()
socket_no = 0
err_no = success

cur_local_coord = 0
cur_tool_coord = 0

cmd_res = 0
warmup_flg = 0

TYPE "Initialization Done."
```

Initialization function for external variables

3 Set the IP address and port number of the vision system as variables (if the default values need to be changed).

```
;Set the network configuration
$ip_address = "10.5.5.100"
$port_no = "9876"
retries_connect = 2 ;times
timeout_connect = 4 ;sec
retry_count = 2     ;times
time_out = 4        ;sec
```

IP address

Port number

4 With the variables set as arguments, execute the connection function (fhconnect).

```
;Connect to the FH server
WHILE bconnected == 0 DO
    CALL fhconnect($ip_address, $port_no, retries_connect, timeout_connect,
socket_no, err_no)
    ; Error check
    IF err_no <> success THEN
        TYPE "ERROR: fhsample_main(): Connection failed. Exit:", err_no
        GOTO 11
    END
END
TYPE "Connection Done."
```

Connection function with the Vision Sensor (FH server)

Switching Scenes on the Vision Sensor

The following describes a program that switches the scene to a Camera Calibration AOS scene.

1 Set the scene number (127) of the Camera Calibration AOS as a variable.
With the variable set as an argument, execute the scene switching function (fhsample_chgscn).

```
scene_no = 127
```

Scene number

```
CALL fhsample_chgscn(socket_no, retry_count, time_out, scene_no, err_no)
```

Scene switching command execution sample function

```
; Error check
IF err_no <> success THEN
    TYPE "ERROR:  fhsample_main():Change scene No. failed. Exit :", err_no
    GOTO 10
END
```

Acquiring the Camera Status from the 3D Vision Sensor

The following describes a program that acquires the camera status of the 3D vision sensor and checks if the warmup of the camera is completed.

1 To send the camera status acquisition command to the vision system, set the nonprocedural command name to *RBCOM_GET_CAMERA_STATUS* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
;Get Camera Status
WHILE warmup_flg == 0 DO

    $cmd_name = "RBCOM_GET_CAMERA_STATUS"
    $cmd_arg[0] = ""
    $cmd_arg[1] = ""
    $cmd_arg[2] = ""
    $cmd_arg[3] = ""
    $cmd_arg[4] = ""
    cmd_arg_num = 0
    CALL fhrunsendcmd(socket_no, cmd_arg_num, $cmd_name, $cmd_arg[], err_no)
    IF err_no <> success THEN
        GOTO 10
    END
```

nonprocedural command transmission function

2 To receive the response to the camera status acquisition command from the vision system, execute the numerical sequence receiving function (fhrunrecvval).

```
CALL fhrunrecvval(retry_count, time_out, socket_no, param[], err_no)
IF err_no <> success THEN
   GOTO 10
END
IF param[0] <> 0 THEN
   GOTO 10
END
```

numerical sequence receiving function

3 Keep sending the camera status acquisition command to the vision system until the camera status changes to warmup complete.

```
CASE param[1] OF
 VALUE -1:
   warmup_flg = 0
   WAIT 1000
 VALUE 1:
   warmup_flg = 1
 ANY
   GOTO 10
END
END
```

If the camera status is warmup incomplete, send the camera status acquisition command again.

If the camera status is warmup complete, proceed to the next processing.

Acquiring the Calibration Date/Time Comparison Result the from 3D Vision Sensor

The following describes a program that checks if the date of calibration of the 3D vision sensor matches the internal date of the vision system and determines whether the AOS camera calibration must be executed.

1 To send the date/time comparison result acquisition command to the vision system, set the nonprocedural command name to *RBCOM_GET_CALIBTIME_COMP* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
;Get Calib time
$cmd_name = "RBCOM_GET_CALIBTIME_COMP"
$cmd_arg[0] = ""
$cmd_arg[1] = ""
$cmd_arg[2] = ""
$cmd_arg[3] = ""
$cmd_arg[4] = ""
cmd_arg_num = 0
CALL fhrunsendcmd(socket_no, cmd_arg_num, $cmd_name, $cmd_arg[], err_no)
IF err_no <> success THEN
   GOTO 10
END
```

nonprocedural command transmission function

2 To receive the response to the calibration date/time comparison result acquisition command from the vision system, execute the numerical sequence receiving function (fhrunrecvval).
If the date of calibration of the camera matches the internal date of the FH series, execute the sample program (fhsample_main) without executing camera calibration.

```
CALL fhrunrecvval(retry_count, time_out, socket_no, param[], err_no)
IF err_no <> success THEN
    GOTO 10
END
IF param[0] <> 0 THEN
    GOTO 10
END

IF param[1] == 1 THEN
    TYPE "The calibration date and the current date have been matched"
    TYPE "Calibration is not necessary"
    GOTO 11
END
```

numerical sequence receiving function

If calibration is not required, execute the sample program (fhsample_main).

Changing the Calibration Mode of AOS Camera Calibration

The following describes a program that changes the calibration mode of AOS camera calibration to automatic calibration so that AOS camera calibration can be executed during measurement.

1 To change the calibration mode of the AOS camera calibration to automatic calibration, set the first nonprocedural command argument $cmd_arg[0] to *1*.
To send the camera calibration mode change command to the vision system, set the nonprocedural command name to *RBCOM_SET_CALIB_MODE* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
;Set Calib mode
$cmd_name = "RBCOM_SET_CALIB_MODE"
$cmd_arg[0] = "1"
$cmd_arg[1] = ""
$cmd_arg[2] = ""
$cmd_arg[3] = ""
$cmd_arg[4] = ""
cmd_arg_num = 1
CALL fhrunsendcmd(socket_no, cmd_arg_num, $cmd_name, $cmd_arg[], err_no)
IF err_no <> success THEN
    GOTO 10
END
```

Set the camera calibration mode to automatic calibration.

nonprocedural command transmission function

2 To receive the response to the camera calibration mode change command from the vision system, execute the numerical sequence receiving function (fhrunrecvval).

```
CALL fhrunrecvval(retry_count, time_out, socket_no, param[], err_no)
IF err_no <> success THEN
    GOTO 10
END
IF param[0] <> 0 THEN
    GOTO 10
END
```

numerical sequence receiving function

Acquiring the Near Imaging Position

The following describes a program that acquires the near imaging position registered in *Automatic Calibration (Vision Master)*.

1 To acquire the near imaging position, set the first nonprocedural command argument $cmd_arg[0] to *0*. To send the camera calibration imaging position acquisition command to the vision system, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
;Get Calib Position Near
$cmd_name = "RBCOM_GET_CALIB_POS"
$cmd_arg[0] = "0"
$cmd_arg[1] = ""
$cmd_arg[2] = ""
$cmd_arg[3] = ""
$cmd_arg[4] = ""
cmd_arg_num = 1
CALL fhrunsendcmd(socket_no, cmd_arg_num, $cmd_name, $cmd_arg[], err_no)
IF err_no <> success THEN
    GOTO 10
END
```

Set this argument to *0* to acquire the near imaging position.

nonprocedural command transmission function

2 To receive the near imaging position from the vision system, execute the numerical sequence receiving function (fhrunrecvval). Store the values of the received near imaging position in variables.

```
CALL fhrunrecvval(retry_count, time_out, socket_no, param[], err_no)
IF err_no <> success THEN
    GOTO 10
END
IF param[0] <> 0 THEN
    GOTO 10
END
```

numerical sequence receiving function

```
cal_pos_near_x = param[1]
cal_pos_near_y = param[2]
cal_pos_near_z = param[3]
cal_pos_near_w = param[4]
cal_pos_near_p = param[5]
cal_pos_near_r = param[6]
```

Assign the values of the received near imaging position to variables.

Moving the Robot to the Near Imaging Position

The following describes a program that moves the robot to the near imaging position.

1 Execute the robot motion function (fhsample_move) to move the robot to the acquired near imaging position.

```
;Move the calib position near
CALL fhsample_move(cal_pos_near_x, cal_pos_near_y, cal_pos_near_z,
cal_pos_near_w, cal_pos_near_p, cal_pos_near_r, err_no)
IF err_no <> success THEN
   GOTO 10
END
```

Robot motion sample function

⚠ WARNING

- These operations drive the robot.
- Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

Executing a Measurement (Near Imaging Position)

The following describes a program that executes a measurement at the near imaging position.

1 To send the measurement command to the vision system, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
;Send Measure Command
$cmd_name = "MEASURE"
$cmd_arg[0] = ""
$cmd_arg[1] = ""
$cmd_arg[2] = ""
$cmd_arg[3] = ""
$cmd_arg[4] = ""
cmd_arg_num = 0
CALL fhrunsendcmd(socket_no, cmd_arg_num, $cmd_name, $cmd_arg[], err_no)
IF err_no <> success THEN
   GOTO 10
END
```

nonprocedural command transmission function

2 To receive the response to the measurement command from the vision system, execute the command response receiving function (fhrunrecvres).
If the response is not OK, execute the processing to end the program.

```
;Recv Command Response
CALL fhrunrecvres(retry_count, time_out, socket_no, cmd_res, err_no)
IF err_no <> success THEN
    GOTO 10
END
;Command Response Check
IF cmd_res <> 1 THEN
    GOTO 10
END
```

command response receiving function

If the response is not OK, exit the program.

3 To receive the measurement result from the vision system, execute the numerical sequence receiving function (fhrunrecvval) and check the value of the received overall judgment.
This program is created on the assumption that the measurement result from the vision system is sent as “ TJG.”

```
;Get the Measurement Result
CALL fhrunrecvval(retry_count, time_out, socket_no, param[], err_no)
IF err_no <> success THEN
    GOTO 10
END
;Total Judge Check
IF param[0] <> 1 THEN
    GOTO 10
END
```

numerical sequence receiving function

If the overall judgment is not OK, exit the program

Acquiring the Far Imaging Position

The following describes a program that acquires the far imaging position registered in *Automatic Calibration (Vision Master)*.

1 To acquire the far imaging position, set the first nonprocedural command argument $cmd_arg[0] to *1*. To send the camera calibration imaging position acquisition command to the vision system, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
;Get Calib Position Far
$cmd_name = "RBCOM_GET_CALIB_POS"
$cmd_arg[0] = "1"
$cmd_arg[1] = ""
$cmd_arg[2] = ""
$cmd_arg[3] = ""
$cmd_arg[4] = ""
cmd_arg_num = 1
CALL fhrunsendcmd(socket_no, cmd_arg_num, $cmd_name, $cmd_arg[], err_no)
IF err_no <> success THEN
   GOTO 10
END
```

Set this argument to *1* to acquire the far imaging position.

nonprocedural command transmission function

2 To receive the far imaging position from the vision system, execute the numerical sequence receiving function (fhrunrecvval). Store the values of the received far imaging position in variables.

```
CALL fhrunrecvval(retry_count, time_out, socket_no, param[], err_no)
IF err_no <> success THEN
   GOTO 10
END
IF param[0] <> 0 THEN
   GOTO 10
END
cal_pos_far_x = param[1]
cal_pos_far_y = param[2]
cal_pos_far_z = param[3]
cal_pos_far_w = param[4]
cal_pos_far_p = param[5]
cal_pos_far_r = param[6]
```

nonprocedural command transmission function

Assign the values of the received far imaging position to variables.

Moving the Robot to the Far Imaging Position

The following describes a program that moves the robot to the far imaging position.

1 Execute the robot motion function (fhsample_move) to move the robot to the acquired far imaging position.

```
;Move the calib position Far
CALL fhsample_move(cal_pos_far_x, cal_pos_far_y, cal_pos_far_z,
cal_pos_far_w, cal_pos_far_p, cal_pos_far_r, err_no)
IF err_no <> success THEN
   GOTO 10
END
```

Robot motion sample function

⚠ WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

Executing a Measurement (Far Imaging Position)

The following describes a program that executes a measurement at the far imaging position.

1 To send the measurement command to the vision system, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
;Send Measure Command
$cmd_name = "MEASURE"
$cmd_arg[0] = ""
$cmd_arg[1] = ""
$cmd_arg[2] = ""
$cmd_arg[3] = ""
$cmd_arg[4] = ""
cmd_arg_num = 0
CALL fhrunsendcmd(socket_no, cmd_arg_num, $cmd_name, $cmd_arg[], err_no)
IF err_no <> success THEN
   GOTO 10
END
```

nonprocedural command transmission function

2 To receive the response to the measurement command from the vision system, execute the command response receiving function (fhrunrecvres).
If the response is not OK, execute the processing to end the program.

```
;Recv Command Response
CALL fhrunrecvres(retry_count, time_out, socket_no, cmd_res, err_no)
IF err_no <> success THEN
   GOTO 10
END
```

Command response receiving function

```
;Command Response Check
IF cmd_res <> 1 THEN
    GOTO 10
END
```

If the response is not OK, exit the program.

3 To receive the measurement result from the vision system, execute the numerical sequence receiving function (fhrunrecvval) and check the value of the received overall judgment.
This program is created on the assumption that the measurement result from the vision system is sent as " TJG."

```
;Get the Measurement Result
CALL fhrunrecvval(retry_count, time_out, socket_no, param[], err_no)
IF err_no <> success THEN
    GOTO 10
END
;Total Judge Check
IF param[0] <> 1 THEN
    GOTO 10
END
```

nonprocedural command transmission function

If the overall judgment is not OK, exit the program

Ending the Program (Normal End)

The following describes processing that is carried out when AOS camera calibration is normally ended.

1 If this processing is successful, the program sends the camera calibration mode change command to manual calibration by the nonprocedural command transmission function (fhrunsendcmd), and then executes the sample program (fhsample_main).

```
11
;Set Calib mode
$cmd_name = "RBCOM_SET_CALIB_MODE"
$cmd_arg[0] = "0"
$cmd_arg[1] = ""
$cmd_arg[2] = ""
$cmd_arg[3] = ""
$cmd_arg[4] = ""
cmd_arg_num = 1
CALL fhrunsendcmd(socket_no, cmd_arg_num, $cmd_name, $cmd_arg[], err_no)
IF err_no <> success THEN
    GOTO 10
END
CALL fhrunrecvval(retry_count, time_out, socket_no, param[], err_no)
IF err_no <> success THEN
    GOTO 10
END
```

Set the camera calibration mode to manual calibration.

nonprocedural command transmission function

```
IF param[0] <> 0 THEN
    GOTO 10
END
IF bconnected == 1 THEN
    CALL fhclose(socket_no, err_no)
END
TYPE "AOS Camera Calibration Successful"
CALL fhsample_main()
```

Execute the sample program (fhsample_main).

Ending the Program (Error End)

The following describes processing that is carried out if an error occurs during execution of AOS camera calibration.

1 If an error occurs during execution, the program sends the camera calibration mode change command by the nonprocedural command transmission function (fhrunsendcmd) to set the manual calibration mode, and then stops the program.

```
10
    ;Set Calib mode
    $cmd_name = "RBCOM_SET_CALIB_MODE"
    $cmd_arg[0] = "0"
    $cmd_arg[1] = ""
    $cmd_arg[2] = ""
    $cmd_arg[3] = ""
    $cmd_arg[4] = ""
    cmd_arg_num = 1
    CALL fhrunsendcmd(socket_no, cmd_arg_num, $cmd_name, $cmd_arg[], err_no)
    CALL fhrunrecvval(retry_count, time_out, socket_no, param[], err_no)

    ;Error
    IF bconnected == 1 THEN
        CALL fhclose(socket_no, err_no)
    END
    TYPE "AOS Camera Calibration Failed"
    TYPE "Disconnection Done"
    STOP
```

Set the camera calibration mode to manual calibration.

nonprocedural command transmission function

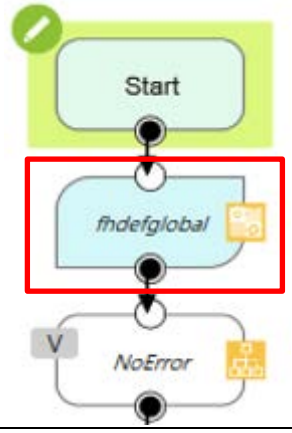
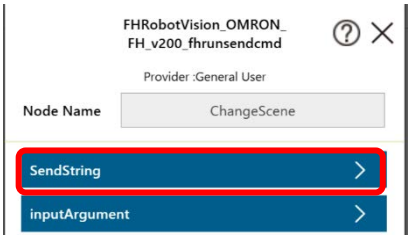
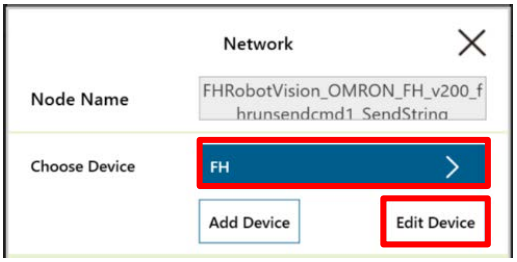
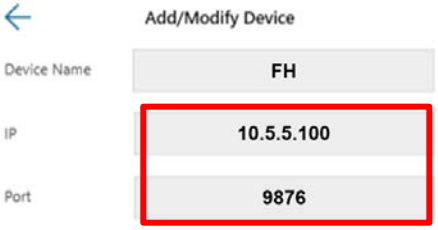
3.7.2. For OMRON TM Series

The explanation of the sample program is explained using the node name (component) / device name (network) of the TM robot. For the TM S robots, see the table below.

	TM	TM S
Node name (Component)	fhdefglobal	FH_v200_fhdefglobal1
	ChangeScene	FH_v200_fhrunsendcmd1
	GetCameraStatus	FH_v200_fhrunsendcmd2
	CompCalibTime	FH_v200_fhrunsendcmd3
	GetImgPos	FH_v200_fhrunsendcmd4
	MEASURE	FH_v200_fhrunsendcmd5
	SetCalibMode	FH_v200_fhrunsendcmd6
	RecvResponse	FH_v200_fhrunrecvres1-fhrunrecvres2
	RecvVal	FH_v200_fhrunrecvval1-fhrunrecvval5
Device name (Network)	FH	ntd_FH

Initialization of the Sample Program

The following describes a program that is required to initialize variables and connect the vision system to the robot controller.

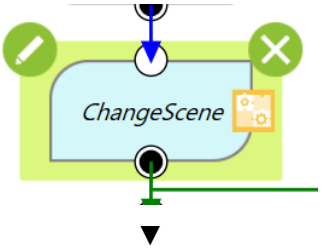
1	In the Edit project window, select the FHSAMPLE_CALIBAOS subflow.	
2	The **fhdefglobal** component is placed at the top of the flow. This component initializes global variables that are required for the robot controller to communicate with the vision system.	 fhdefglobal Component
3	Select the **ChangeScene** node on the flow, and click the pencil icon to open the setting window. Click **SendString** and confirm that *FH* is selected in **Choose Device**. Click **Edit Device**, and set the IP address and port number of the vision system. After you set them, click the **Done** button to close the dialog box. There is no need to configure the communication settings for each node since the sample program uses the FH series as a common communication device.	 fhrunsendcmd Component   

Switching Scenes on the Vision Sensor

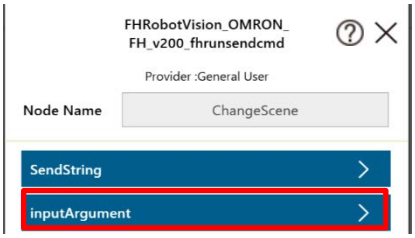
The following describes a program that switches the scene to a Camera Calibration AOS scene.

1 In the Edit project window, select the FHSAMPLE_CALIBAOS subflow.

2 Select the **ChangeScene** node on the flow, and click the pencil icon to open the setting window.



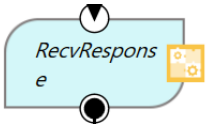
fhrunsendcmd Component



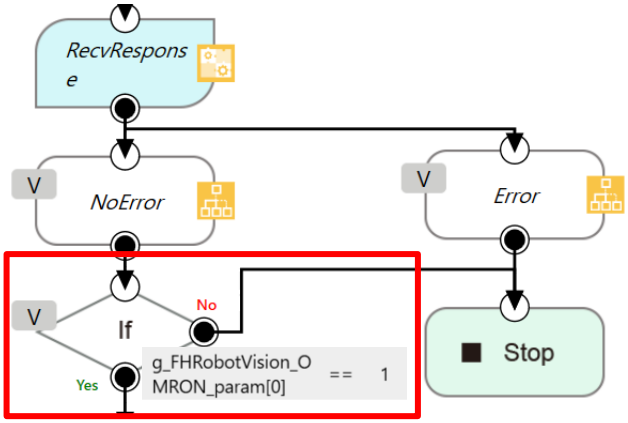
3 Set the scene number (127) of the Camera Calibration AOS as an argument to the fhrunsendcmd component.
Execute the fhrunsendcmd component with the set value to send a scene switching command to the vision system.
* Click **InputArgument** > **Variables[7]** to open the setting window.

int	FHRobotVision_OMRON_FH_v200_fhrunsendcmd1_var_cmdArgNum	=	1	Number of command arguments
string	FHRobotVision_OMRON_FH_v200_fhrunsendcmd1_var_cmdName	=	"SCENE"	Scene switching command
string[]	FHRobotVision_OMRON_FH_v200_fhrunsendcmd1_var_cmdArg[0]	=	"127"	Scene number

4 Execute the **RecvResponse** node to receive the response to the scene switching command.



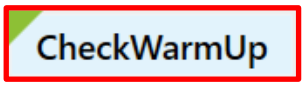
fhrunrecvres Component

5	Check the execution result (command response) of the **RecvResponse** node. The execution result is stored in the global variable *g_FHRobotVision_OMRON_param[0]*. If the execution result is 1 (OK), proceed to the next node. 

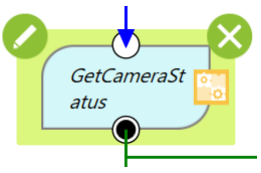
Acquiring the Camera Status from the 3D Vision Sensor

The following describes a program that acquires the camera status of the 3D vision sensor and checks if the warmup of the camera is completed.

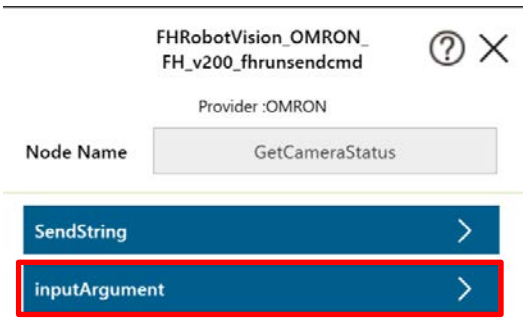
1 In the Edit project window, select the CheckWarmUp subflow . The CheckWarmUp subflow is called from the FHSAMPLE_CALIBAOS subflow.



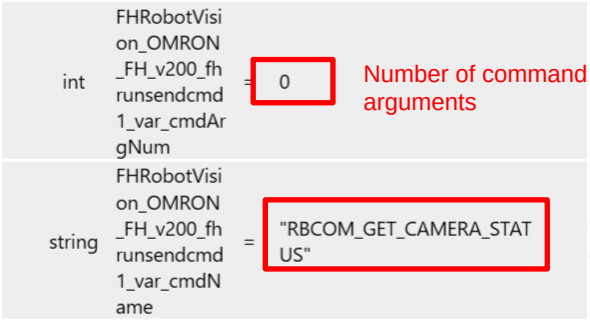
2 Select the **GetCameraStatus** node on the flow, and click the pencil icon to open the setting window.



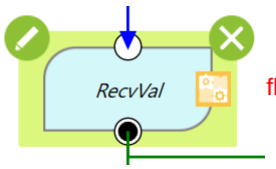
fhrunsendcmd Component



3 To send the camera status acquisition command to the vision system, set the nonprocedural command name to *RBCOM_GET_CAMERA_STATUS* and execute the fhrunsendcmd component.

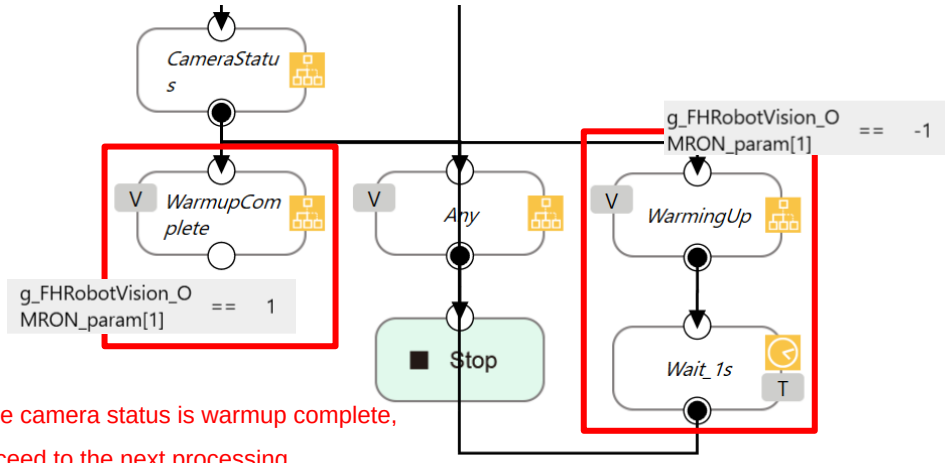


4 Execute the **RecvVal** node to receive the response to the camera status acquisition command from the vision system.



fhrunrecvval Component

5 Keep sending the camera status acquisition command to the vision system until the camera status changes to warmup complete.



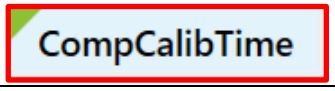
If the camera status is warmup complete, proceed to the next processing.

If the camera status is warmup incomplete, send the camera status acquisition command again.

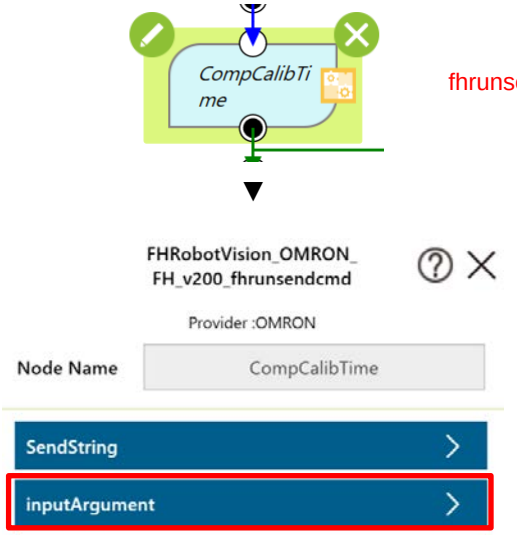
Acquiring the Calibration Date/Time Comparison Result the from 3D Vision Sensor

The following describes a program that checks if the date of calibration of the 3D vision sensor matches the internal date of the vision system and determines whether the AOS camera calibration must be executed.

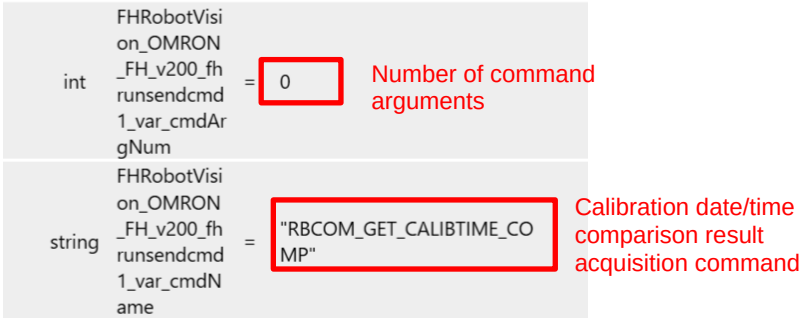
1 In the Edit project window, select the CompCalibTime subflow.
The CompCalibTime subflow is called from the FHSAMPLE_CALIBAOS subflow.



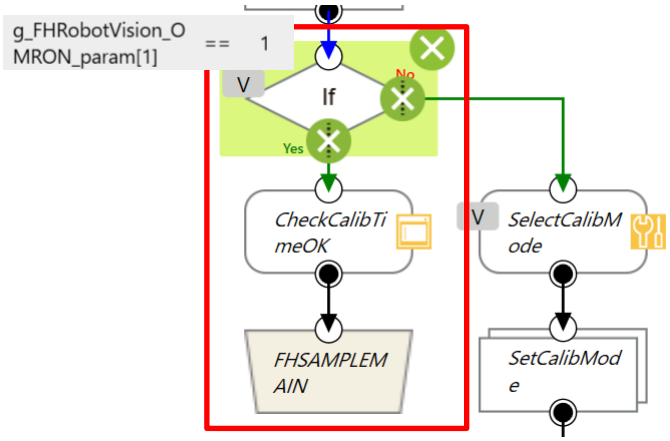
2 Select the **CompCalibTime** node on the flow, and click the pencil icon to open the setting window.



3 To send the date/time comparison result acquisition command to the vision system, set the nonprocedural command name to *RBCOM_GET_CALIBTIME_COMP* and execute the fhrunsendcmd component.

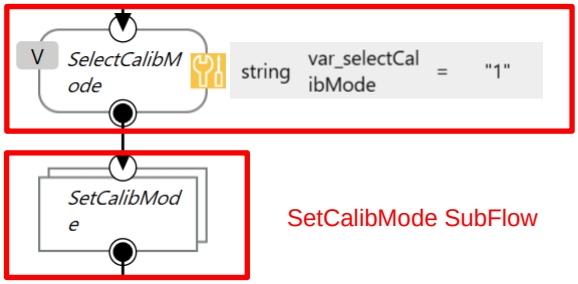
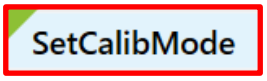


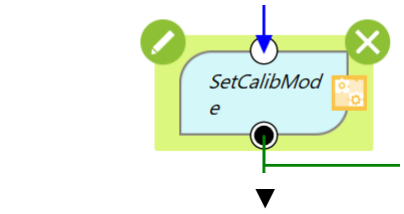
4 To receive the response to the calibration date/time comparison result acquisition command from the vision system, execute the numerical sequence receiving function (FHRUNRCVVAL).

5	If the date of calibration of the camera matches the internal date of the FH series, execute the sample program (FHSMPLMAIN) without executing camera calibration. * This processing is included in the FHSAMPLE_CALIBAOS subflow.  If calibration is not required, execute the sample program (FHSAMPLEMAIN).

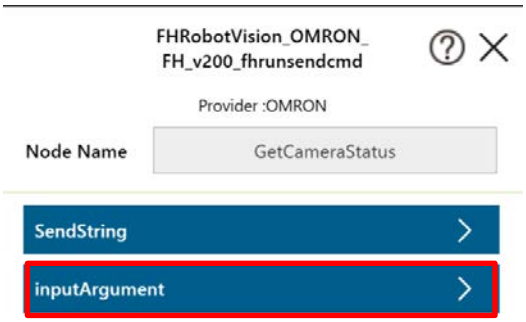
Changing the Calibration Mode of AOS Camera Calibration

The following describes a program that changes the calibration mode of AOS camera calibration to automatic calibration so that AOS camera calibration can be executed during measurement.

1	In the Edit project window, select the FHSAMPLE_CALIBAOS subflow. 
2	To change the camera calibration mode to automatic calibration, set the variable to determine the operation of SetCalibMode to *1*, and execute the SetCalibMode subflow.  SetCalibMode SubFlow
3	In the Edit project window, select the SetCalibMode subflow. 
4	Select the **SetCalibMode** node on the flow, and click the pencil icon to open the setting window.



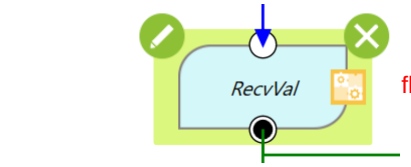
fhrunsendcmd Component



5 To send the camera calibration mode change command to the vision system, set the nonprocedural command name to *RBCOM_SET_CALIB_MODE* and execute the fhrunsendcmd component.

int	FHRobotVision_OMRON_FH_v200_fhrunsendcmd1_var_cmdArgNum	=	1	Number of command arguments
string	FHRobotVision_OMRON_FH_v200_fhrunsendcmd1_var_cmdName	=	"RBCOM_SET_CALIB_MODE"	Camera calibration mode change command
string[]	FHRobotVision_OMRON_FH_v200_fhrunsendcmd1_var_cmdArg[0]	=	var_selectCalibMode	Variable set in step 2

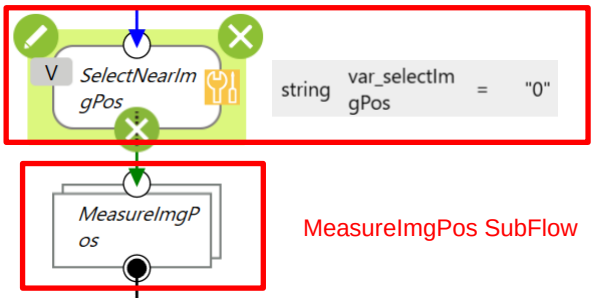
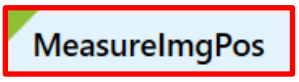
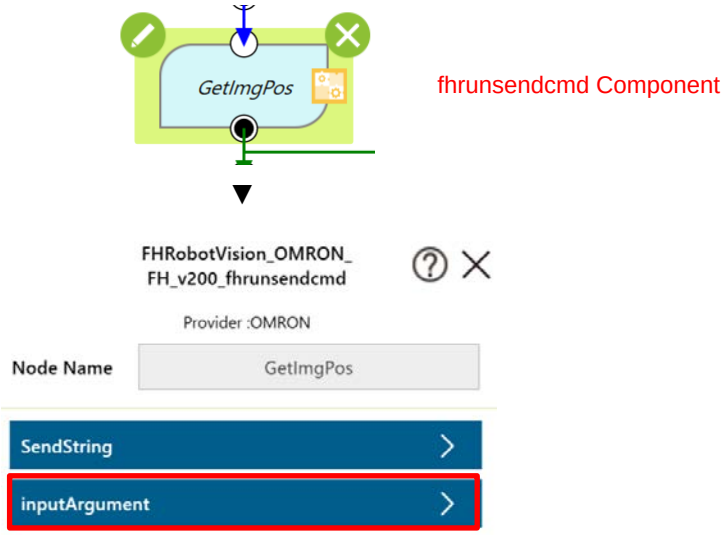
6 To receive the response to the camera calibration mode change command from the vision system, execute the numerical sequence receiving function (FHRUNRCVVAL).

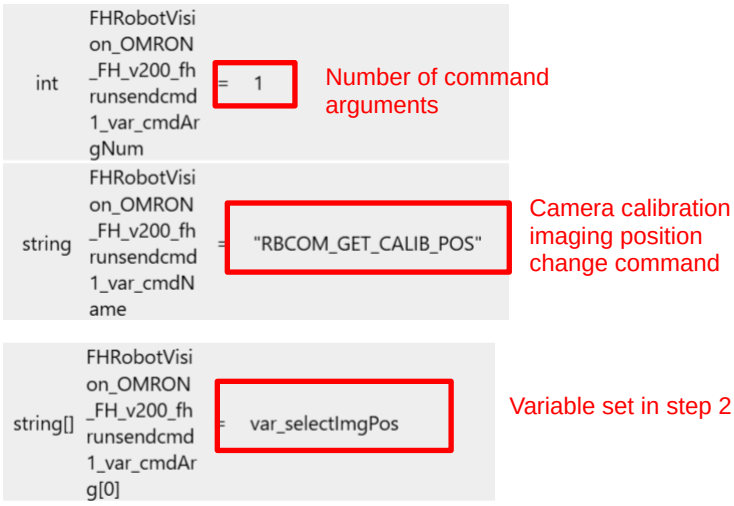


fhrunrecvval Component

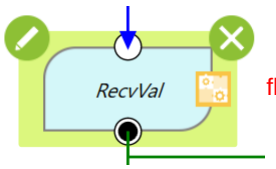
Acquiring the Near Imaging Position

The following describes a program that acquires the near imaging position registered in *Automatic Calibration (Vision Master)*.

1	In the Edit project window, select the FHSAMPLE_CALIBAOS subflow. 
2	To acquire the near imaging position, set the variable to determine the operation of MeasureImgPos to *0* and execute the MeasureImgPos subflow. 
3	In the Edit project window, select the MeasureImgPos subflow. 
4	Select the **GetImgPos** node on the flow, and click the pencil icon to open the setting window. 
5	To send the camera calibration imaging position acquisition command to the vision system, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the fhrunsendcmd component.



6 To receive the response to the camera calibration imaging position acquisition command from the vision system, execute the numerical sequence receiving function (FHRUNRCVVAL).

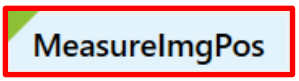


fhrunrecvval Component

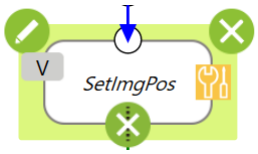
Moving the Robot to the Near Imaging Position

The following describes a program that moves the robot to the near imaging position.

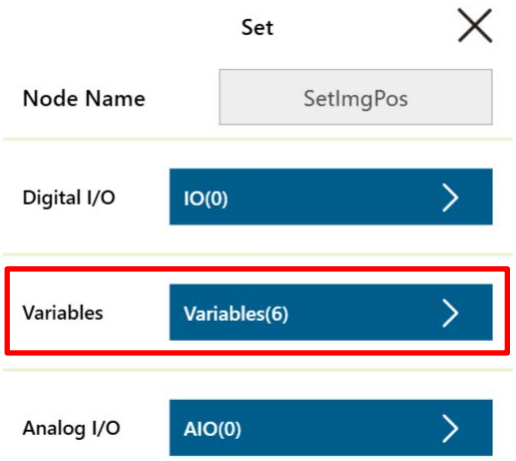
1 In the Edit project window, select the MeasureImgPos subflow.



2 Select the **SetImgPos** node on the flow, and click the pencil icon to open the setting window.



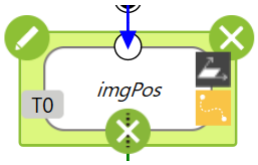
Set Node



3 Set the values of the acquired near imaging position as imgPos point values.

Point ["imgPos"].Value[0]	=	g_FHRobotVision_OMRON_param[1]
Point ["imgPos"].Value[1]	=	g_FHRobotVision_OMRON_param[2]
Point ["imgPos"].Value[2]	=	g_FHRobotVision_OMRON_param[3]
Point ["imgPos"].Value[3]	=	g_FHRobotVision_OMRON_param[4]
Point ["imgPos"].Value[4]	=	g_FHRobotVision_OMRON_param[5]
Point ["imgPos"].Value[5]	=	g_FHRobotVision_OMRON_param[6]

4 Execute the Point node to move to the near imaging position.



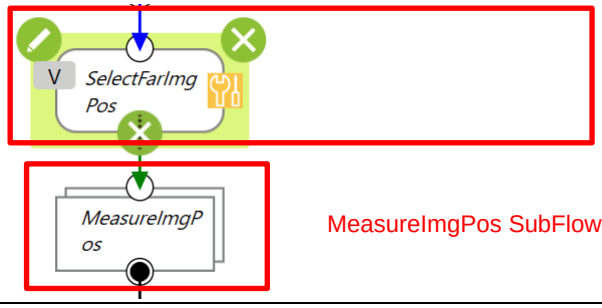
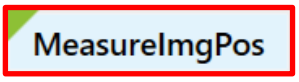
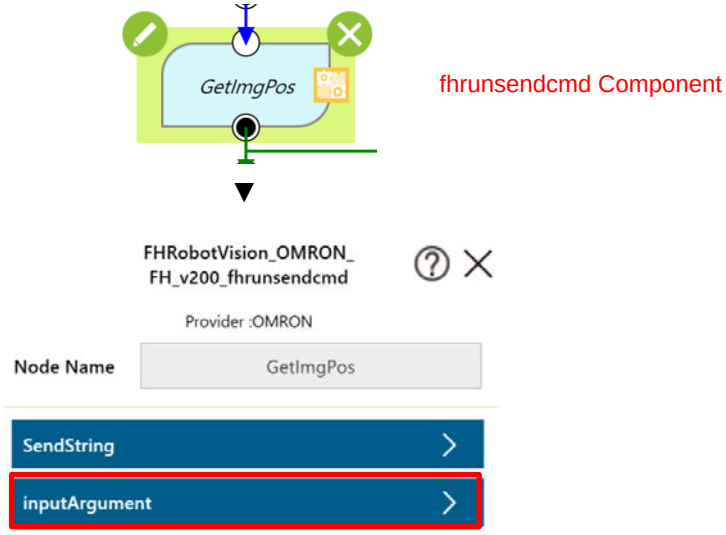
Point Node

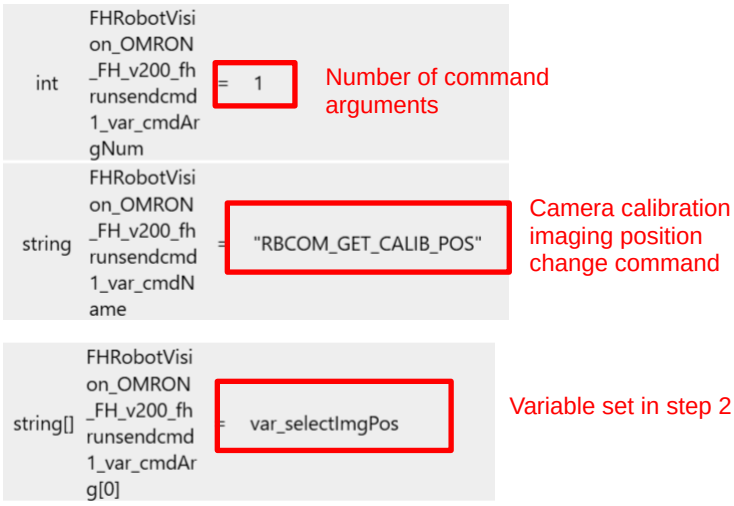
⚠ WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

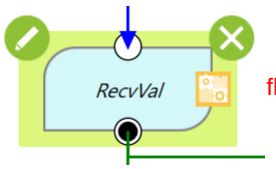
Acquiring the Far Imaging Position

The following describes a program that acquires the far imaging position registered in *Automatic Calibration (Vision Master)*.

1	In the Edit project window, select the FHSAMPLE_CALIBAOS subflow. 
2	To acquire the far imaging position, set the variable to determine the operation of MeasureImgPos to *0* and execute the MeasureImgPos subflow. 
3	In the Edit project window, select the MeasureImgPos subflow. 
4	Select the **GetImgPos** node on the flow, and click the pencil icon to open the setting window. 
5	To send the camera calibration imaging position acquisition command to the vision system, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the fhrunsendcmd component.



6 To receive the response to the camera calibration imaging position acquisition command from the vision system, execute the numerical sequence receiving function (FHRUNRCVVAL).

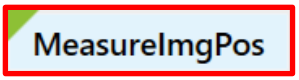


fhrunrecvval Component

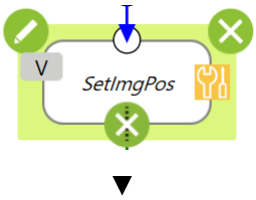
Moving the Robot to the Far Imaging Position

The following describes a program that moves the robot to the far imaging position.

1 In the Edit project window, select the MeasureImgPos subflow.

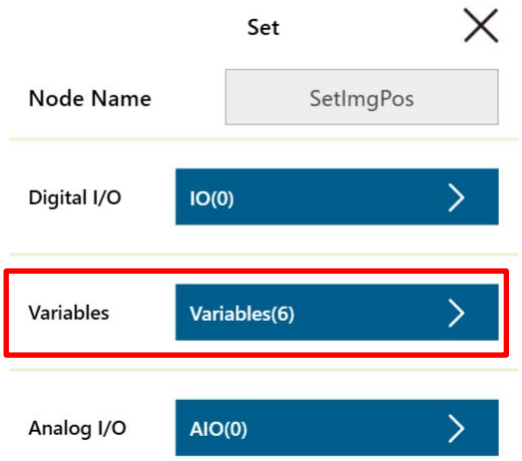


2 Select the **SetImgPos** node on the flow, and click the pencil icon to open the setting window.



Set Node

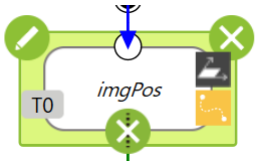
▼



3 Set the values of the acquired far imaging position as imgPos point values.

Point ["imgPos"].Value[0]	=	g_FHRobotVision_OMRON_param[1]
Point ["imgPos"].Value[1]	=	g_FHRobotVision_OMRON_param[2]
Point ["imgPos"].Value[2]	=	g_FHRobotVision_OMRON_param[3]
Point ["imgPos"].Value[3]	=	g_FHRobotVision_OMRON_param[4]
Point ["imgPos"].Value[4]	=	g_FHRobotVision_OMRON_param[5]
Point ["imgPos"].Value[5]	=	g_FHRobotVision_OMRON_param[6]

4 Execute the Point node to move to the far imaging position.



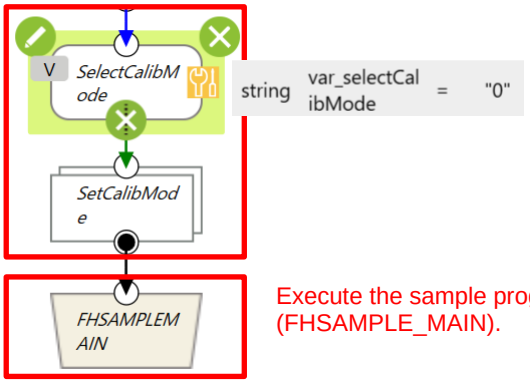
Point Node

⚠ WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

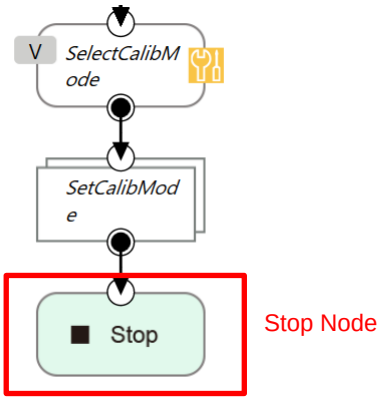
Ending the Program (Normal End)

The following describes processing that is carried out when AOS camera calibration is normally ended.

1	In the Edit project window, select the FHSAMPLE_CALIBAOS subflow. 
2	If this processing is successful, the program changes the camera calibration mode to manual calibration, and then executes the sample program (fhsample_main). Change the camera calibration mode to manual calibration.  Execute the sample program (FHSAMPLE_MAIN).

Ending the Program (Error End)

The following describes processing that is carried out if an error occurs during execution of AOS camera calibration.

1	If an error occurs, the program execution is ended in the Stop node. The Stop node is placed so that it is executed during error processing in each subflow.  Stop Node

3.7.3. For Robots Manufactured by FANUC Corporation

Connecting Vision Sensor to Robot Controller

The following describes a program that connects the vision system to the robot controller.

1	In **Tag C1:**, set the IP address of the vision system. Also, set the port number of the vision system.
2	Call the external variable initialization function (FHDEFGLOBAL) to initialize external variables. The external variables include variables for managing the communications status. At the time of execution, be sure to call the external variable initialization function (FHDEFGLOBAL) to initialize them. **1: CALL FHDEFGLOBAL ;** Initialization function for external variables
3	Set the local coordinate system number and the tool coordinate system number to *0*. **2: UFRAME_NUM=0 ;** Local Coordinate System No. **3: UTOOL_NUM=0 ;** Tool Coordinate System No
4	For Registers 196 and 197, set the number of retry times at connection and the timeout period (in seconds). For Registers 198 and 199, set the number of retry times at communications and the timeout period (in seconds). **5: R[196]=2 ;** The number of retry times at server connection **6: R[197]=10 ;** The timeout period at server connection [sec.] **7: R[198]=2 ;** The number of retry times at communications **8: R[199]=10 ;** The timeout period at communications [sec.]
5	With the set register variables as arguments, call the connection function (FHCONNECT) to connect to the vision system (FH server). **10: CALL FHCONNECT('C1:',R[196],R[197]) ;** Connection function with the Vision Sensor (FH server) **11: IF R[200]<>0,JMP LBL[999] ;**

Switching Scenes on the Vision Sensor

The following describes a program that switches the scene to a Camera Calibration AOS scene.

1	Set the scene number (127) of the Camera Calibration AOS as a variable. With the set variable as an argument, execute the scene switching function (FHSMPLCHGSN). Scene number **13: CALL FHSMPLCHGSN(R[198],R[199],127) ;** Scene switching command execution sample function **14: IF R[200]<>0,JMP LBL[998] ;**

Acquiring the Camera Status from the 3D Vision Sensor

The following describes a program that acquires the camera status of the 3D vision sensor and checks if the warmup of the camera is completed.

1 To send the camera status acquisition command to the vision system, set the nonprocedural command name to *RBCOM_GET_CAMERA_STATUS* and execute the nonprocedural command transmission function (FHRUNSNDCMD).

```
16:  LBL[990] ;
17:  WAIT    1.00(sec);
18:  CALL
FHRUNSNDCMD(0,'RBCOM_GET_CAMERA_STATUS','...','...','...','...');
19:  IF R[200]<>0,JMP LBL[998] ;
```

nonprocedural command transmission function

2 To receive the response to the camera status acquisition command from the vision system, execute the numerical sequence receiving function (FHRUNRCVRES).

```
21:  CALL FHRUNRCVVAL(R[198],R[199],170) ;
22:  IF R[200]<>0,JMP LBL[998] ;
23:  IF R[170]<>0,JMP LBL[998] ;
```

numerical sequence receiving function

3 Keep sending the camera status acquisition command to the vision system until the camera status changes to warmup complete.

```
25:  IF  R[171]=(-1),JMP  LBL[990]  ;
26:    ;
27:  IF  R[171]<>(-1)  AND  R[171]<>1,JMP  LBL[998]  ;
```

If the camera status is warmup incomplete, send the camera status acquisition command again.

If the camera status is not warmup complete or warmup incomplete, end the processing.

Acquiring the Calibration Date/Time Comparison Result the from 3D Vision Sensor

The following describes a program that checks if the date of calibration of the 3D vision sensor matches the internal date of the vision system and determines whether the AOS camera calibration must be executed.

1	To send the date/time comparison result acquisition command to the vision system, set the nonprocedural command name to *RBCOM_GET_CALIBTIME_COMP* and execute the nonprocedural command transmission function (FHRUNSNDCMD). **29: CALL FHRUNSNDCMD(0,'RBCOM_GET_CALIBTIME_COMP','...','...','...','...') ;** **30: IF R[200]<>0,JMP LBL[998] ;** nonprocedural command transmission function
2	To receive the response to the calibration date/time comparison result acquisition command from the vision system, execute the numerical sequence receiving function (FHRUNRCVVAL). If the date of calibration of the camera matches the internal date of the FH series, execute the sample program (FHSMPLMAIN) without executing camera calibration. **32: CALL FHRUNRCVVAL(R[198],R[199],170) ;** numerical sequence receiving function **33: IF R[200]<>0,JMP LBL[998] ;** **34: IF R[170]<>0,JMP LBL[998] ;** **35: ;** If calibration is not required, execute the sample program (FHSMPLMAIN). **36: IF R[171]=1,JMP LBL[991] ;**

Changing the Calibration Mode of AOS Camera Calibration

The following describes a program that changes the calibration mode of AOS camera calibration to automatic calibration so that AOS camera calibration can be executed during measurement.

1 To change the calibration mode of the AOS camera calibration to automatic calibration, set the first nonprocedural command argument to *1*.
To send the camera calibration mode change command to the vision system, set the nonprocedural command name to *RBCOM_SET_CALIB_MODE* and execute the nonprocedural command transmission function (FHRUNSNDCMD).

Set the camera calibration mode to automatic calibration.

```
38:  CALL FHRUNSNDCMD(1,'RBCOM_SET_CALIB_MODE','1','...','...','...','...') ;
39:  IF R[200]<>0,JMP LBL[998] ;
```

nonprocedural command transmission function

2 To receive the response to the AOS camera calibration mode change command from the vision system, execute the numerical sequence receiving function (FHRUNRCVVAL).

```
41:  CALL FHRUNRCVVAL(R[198],R[199],170) ;
42:  IF R[200]<>0,JMP LBL[998] ;
43:  IF R[170]<>0,JMP LBL[998] ;
```

numerical sequence receiving function

Acquiring the Near Imaging Position

The following describes a program that acquires the near imaging position registered in *Automatic Calibration (Vision Master)*.

1 To acquire the near imaging position, set the first nonprocedural command argument to *0*. To send the camera calibration imaging position acquisition command to the vision system, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (FHRUNSNDCMD).

Set this argument to *0* to acquire the near imaging position.

```
45:  CALL FHRUNSNDCMD(1,'RBCOM_GET_CALIB_POS','0','...','...','...','...') ;
46:  IF R[200]<>0,JMP LBL[998] ;
```

nonprocedural command transmission function

2 To receive the near imaging position from the vision system, execute the numerical sequence receiving function (FHRUNRCVVAL). Store the values of the received near imaging position in variables.

```
48:  CALL FHRUNRCVVAL(R[198],R[199],170) ;
49:  IF R[200]<>0,JMP LBL[998] ;
50:  IF R[170]<>0,JMP LBL[998] ;
51:   ;
52:  PR[99,1]=R[171]      ;
53:  PR[99,2]=R[172]      ;
54:  PR[99,3]=R[173]      ;
55:  PR[99,4]=R[174]      ;
56:  PR[99,5]=R[175]      ;
57:  PR[99,6]=R[176]      ;
```

numerical sequence receiving function

Assign the values of the received near imaging position to variables.

Moving the Robot to the Near Imaging Position

The following describes a program that moves the robot to the near imaging position.

1 Execute the robot motion function (FHSMPLMOVE) to move the robot to the acquired near imaging position.

```
59:  CALL FHSMPLMOVE(99) ;
60:  IF R[200]<>0,JMP LBL[998] ;
```

Robot motion sample function

⚠ WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

Executing a Measurement (Near Imaging Position)

The following describes a program that executes a measurement at the near imaging position.

1 To send the measurement command to the vision system, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (FHRUNSNDCMD).

```
62:  CALL FHRUNSNDCMD(0,'MEASURE','...','...','...','...','...') ;
63:  IF R[200]<>0,JMP LBL[998] ;
```

nonprocedural command transmission function

2 To receive the response to the measurement command from the vision system, execute the command response receiving function (FHRUNRCVRES).
If the response is not OK, execute the processing to end the program.

```
65:  CALL FHRUNRCVRES(R[198],R[199],170) ;
66:  IF R[200]<>0,JMP LBL[998] ;
67:  IF R[170]<>1,JMP LBL[998] ;
```

command response receiving function

If the response is not OK, exit the program.

3 To receive the measurement result from the vision system, execute the numerical sequence receiving function (FHRUNRCVVAL) and check the value of the received overall judgment.
This program is created on the assumption that the measurement result from the vision system is sent as “ TJG.”

```
69:  CALL FHRUNRCVVAL(R[198],R[199],170) ;
70:  IF R[200]<>0,JMP LBL[998] ;
71:  IF R[170]<>1,JMP LBL[998] ;
```

numerical sequence receiving function

If the response is not OK, exit the program.

Acquiring the Far Imaging Position

1 To acquire the far imaging position, set the first nonprocedural command argument to *1*. To send the camera calibration imaging position acquisition command to the vision system, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (FHRUNSNDCMD).

Set this argument to *1* to acquire the far imaging position.

```
73:  CALL FHRUNSNDCMD(1,'RBCOM_GET_CALIB_POS','1','...','...','...','...') ;
74:  IF R[200]<>0,JMP LBL[998] ;
```

nonprocedural command transmission function

2 To receive the far imaging position from the vision system, execute the numerical sequence receiving function (FHRUNRCVVAL). Store the values of the received far imaging position in variables.

```
76:  CALL FHRUNRCVVAL(R[198],R[199],170) ;
77:  IF R[200]<>0,JMP LBL[998] ;
78:  IF R[170]<>0,JMP LBL[998] ;
79:   ;
80:  PR[99,1]=R[171]    ;
81:  PR[99,2]=R[172]    ;
82:  PR[99,3]=R[173]    ;
83:  PR[99,4]=R[174]    ;
84:  PR[99,5]=R[175]    ;
85:  PR[99,6]=R[176]    ;
```

numerical sequence receiving function

Assign the values of the received far imaging position to variables.

Moving the Robot to the Far Imaging Position

The following describes a program that acquires the far imaging position registered in *Automatic Calibration (Vision Master)*.

1 Execute the robot motion function (FHSMPLMOVE) to move the robot to the acquired far imaging position.

```
87:  CALL FHSMPLMOVE(99) ;
88:  IF R[200]<>0,JMP LBL[998] ;
```

Robot motion sample function

⚠WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

Executing a Measurement (Far Imaging Position)

The following describes a program that executes a measurement at the far imaging position.

1	To send the measurement command to the vision system, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (FHRUNSNDCMD). **90: CALL FHRUNSNDCMD(0,'MEASURE','...','...','...','...') ;** **91: IF R[200]<>0,JMP LBL[998] ;** nonprocedural command transmission function
2	To receive the response to the measurement command from the vision system, execute the command response receiving function (FHRUNRCVRES). If the response is not OK, execute the processing to end the program. **93: CALL FHRUNRCVRES(R[198],R[199],170) ;** command response receiving function **94: IF R[200]<>0,JMP LBL[998] ;** **95: IF R[170]<>1,JMP LBL[998] ;** If the response is not OK, exit the program.
3	To receive the measurement result from the vision system, execute the numerical sequence receiving function (FHRUNRCVVAL) and check the value of the received overall judgment. This program is created on the assumption that the measurement result from the vision system is sent as “ TJG.” **97: CALL FHRUNRCVVAL(R[198],R[199],170) ;** numerical sequence receiving function **IF R[200]<>0,JMP LBL[998] ;** **99: IF R[170]<>1,JMP LBL[998] ;** If the overall judgment is not OK, exit the program

Ending the Program (Normal End)

The following describes processing that is carried out when AOS camera calibration is normally ended.

1 If this processing is successful, the program sends the camera calibration mode change command to manual calibration by the nonprocedural command (FHRUNSNDCMD), and then executes the sample program (FHSMPLMAIN).

Set the camera calibration mode to manual calibration.

```
112:  LBL[991] ;
113:  CALL FHRUNSNDCMD(1,'RBCOM_SET_CALIB_MODE','0','...','...','...','...') ;
114:  IF R[200]<>0,JMP LBL[998] ;
115:   ;
116:  CALL FHRUNRCVVAL(R[198],R[199],170) ;
117:  IF R[200]<>0,JMP LBL[998] ;
118:  IF R[170]<>0,JMP LBL[998] ;
119:  CALL FHCLOSE      ;
120:  CALL FHSMPLMAIN      ;
```

nonprocedural command transmission function

Execute the sample program (fhsample_main).

Ending the Program (Error End)

The following describes processing that is carried out if an error occurs during execution of AOS camera calibration.

1 If an error occurs during execution, the program sends the camera calibration mode change command to manual calibration by the nonprocedural command (FHRUNSNDCMD), and then stops the program.

Set the camera calibration mode to manual calibration.

```
103:  LBL[998] ;
104:  CALL FHRUNSNDCMD(1,'RBCOM_SET_CALIB_MODE','0','...','...','...','...') ;
105:  CALL FHRUNRCVVAL(R[198],R[199],170) ;
106:  CALL FHCLOSE      ;
107:  END ;
108:   ;
109:  LBL[999] ;
110:  END ;
```

nonprocedural command transmission function

3.7.4. For Robots Manufactured by DENSO WAVE Incorporated

Connecting Vision Sensor to Robot Controller

The following describes a program that connects the Vision Sensor to the robot controller.

1	Declare internal variables. * Description omitted (Refer to the source code.)
2	Execute the global variable initialization function (fhdefglobal). Initialization function for external variables
3	Set the client number as same as the communication setting in the robot controller. Ethernet client number
4	With the variables set as arguments, execute the connection function (fhconnect). Connection function with the Vision Sensor (FH server)

```
'!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
' (2)Initialaize global variables
'!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Call fhdefglobal()

err_no = success

cur_local_coord_no = 0
cur_tool_coord_no = 0

PrintMsg "Initialization Done.",-1
```

```
'!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
' (1)Set the network configuration
'   You have to configure the following communication settings.
'!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
fh_socket = 8
retries_connect = 2
timeout_connect = 4
retries_recv = 2
timeout_recv = 4
cmd_res = 0
```

```
'!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
' (2)Connect to the FH server
'!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Do While bconnected = 0
    Call fhconnect(fh_socket, retries_connect, timeout_connect, err_no)

    'Error check
    If err_no <> success Then
        GoTo exitProgram
    End If
Loop

PrintMsg "Connection Done.",-1
```

Switching Scenes on the Vision Sensor

The following describes a program that switches the scene to a Camera Calibration AOS scene.

1 Set the scene number (127) of the Camera Calibration AOS as a variable.
With the variable set as an argument, execute the scene switching function (fhsample_chgscn).

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' (1)Change the scene of the FH
'    You have to select a scene No. for your application.
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
scene_no = 127
Call fhsample_chgscn(fh_socket, retries_recv, timeout_recv, scene_no, err_no)
'Error check
If err_no <> success Then
    GoTo disconnect
End If

PrintMsg "Change scene No. Done.",-1
```

Scene number

Scene switching command execution sample function

Acquiring the Camera Status from the 3D Vision Sensor

The following describes a program that acquires the camera status of the 3D vision sensor and checks if the warmup of the camera is completed.

1 To send the camera status acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CAMERA_STATUS* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' (2)The warming status of camera
'    Check the warming status of camera,When warming 1 second until status is OK
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
warmup_flg = 0

cmd_name = "RBCOM_GET_CAMERA_STATUS"
cmd_arg(0) = ""
cmd_arg(1) = ""
cmd_arg(2) = ""
cmd_arg(3) = ""
cmd_arg(4) = ""
cmd_arg_num = 0

Do While warmup_flg = 0
    Call fhrunsendcmd(fh_socket, cmd_arg_num, cmd_name, cmd_arg(), err_no)
    'Error check
    If err_no <> success Then
        GoTo disconnect
    End If
```

nonprocedural command transmission function

2 To receive the response to the camera status acquisition command from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

```
    'Recv Command Response
    Call fhrunrecvval(fh_socket, retries_recv, timeout_recv, param(), err_no)
    'Error check
    If err_no <> success Then
        GoTo disconnect
    End If
    IF param(0) <> 0 Then
        GoTo disconnect
    End If
```

numerical sequence receiving function

3 Keep sending the camera status acquisition command to the Vision Sensor until the camera status changes to warmup complete.

```
    IF param(1) = -1 Then
        warmup_flg = 0
        Delay 1000
    ElseIf param(1) = 1 Then
        warmup_flg = 1
    Else
        GoTo disconnect
    End If
Loop
PrintMsg "Check the camera status done.",-1
```

If the camera status is warmup incomplete, send the camera status acquisition command again.

If the camera status is warmup complete, proceed to the next processing.

Acquiring the Calibration Date/Time Comparison Result the from 3D Vision Sensor

The following describes a program that checks if the date of calibration of the 3D vision sensor matches the internal date of the Vision Sensor and determines whether the AOS camera calibration must be executed.

1 To send the date/time comparison result acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIBTIME_COMP* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' (3)Get the Calib time to compare
'         Compare the Galib time to the current time
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
cmd_name = "RBCOM_GET_CALIBTIME_COMP"
cmd_arg(0) = ""
cmd_arg(1) = ""
cmd_arg(2) = ""
cmd_arg(3) = ""
cmd_arg(4) = ""
cmd_arg_num = 0
Call fhrunsendcmd(fh_socket, cmd_arg_num, cmd_name, cmd_arg(), err_no)
'Error check
If err_no <> success Then
    GoTo disconnect
End If
```

nonprocedural command transmission function

2 To receive the response to the calibration date/time comparison result acquisition command from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

If the date of calibration of the camera matches the internal date of the FH series, execute the sample program (fhsample_main) without executing camera calibration.

```
'Recv Command Response
Call fhrunrecvval(fh_socket, retries_recv, timeout_recv, param(), err_no)
'Error check
If err_no <> success Then
    GoTo disconnect
End If

IF param(0) <> 0 Then
    GoTo disconnect
End If
IF param(1) = 1 Then
    PrintMsg "The calibration date and the current date have been matched",-1
    PrintMsg "Calibration is not necessary",-1

    If bconnected = 1 Then
        Call fhclose(fh_socket, err_no)
    End If

    GoTo callfhsample
End If
PrintMsg "Check the calibration date done.",-1
```

numerical sequence receiving function

If calibration is not required,
execute the sample program (fhsample_main).

Changing the Calibration Mode of AOS Camera Calibration

The following describes a program that changes the calibration mode of AOS camera calibration to automatic calibration so that AOS camera calibration can be executed during measurement.

1 To change the calibration mode of the AOS camera calibration to automatic calibration, set the first nonprocedural command argument cmd_arg(0) to 1.
To send the camera calibration mode change command to the Vision Sensor, set the nonprocedural command name to *RBCOM_SET_CALIB_MODE* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' (4)Send the execute mode command
'    Set the Calib mode to the auto mode
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
cmd_name = "RBCOM_SET_CALIB_MODE"
cmd_arg(0) = "1"
cmd_arg(1) = ""
cmd_arg(2) = ""
cmd_arg(3) = ""
cmd_arg(4) = ""
cmd_arg_num = 1
Call fhrunsendcmd(fh_socket, cmd_arg_num, cmd_name, cmd_arg(), err_no)
'Error check
If err_no <> success Then
    GoTo disconnect
End If
```

Set the camera calibration mode to automatic calibration.

nonprocedural command transmission function

2 To receive the response to the camera calibration mode change command from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

```
'Recv Command Response
Call fhrunrecvval(fh_socket, retries_recv, timeout_recv, param(), err_no)
'Error check
If err_no <> success Then
    GoTo disconnect
End If

IF param(0) <> 0 Then
    GoTo disconnect
End If

PrintMsg "Change the calibration to auto mode done.",-1
```

numerical sequence receiving function

Acquiring the Near Imaging Position

The following describes a program that acquires the near imaging position registered in Automatic Calibration (Vision Master).

1 To acquire the near imaging position, set the first nonprocedural command argument cmd_arg(0) to 0. To send the camera calibration imaging position acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' (5)Calibration position near
'    Get the calib position near,Meausure at calib position near
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
'Get calib position near
cmd_name = "RBCOM_GET_CALIB_POS"
cmd_arg(0) = "0"
cmd_arg(1) = ""
cmd_arg(2) = ""
cmd_arg(3) = ""
cmd_arg(4) = ""
cmd_arg_num = 1
Call fhrunsendcmd(fh_socket, cmd_arg_num, cmd_name, cmd_arg(), err_no)
'Error check
If err_no <> success Then
    GoTo setcalibmode
End If
```

Set this argument to *0* to acquire the near imaging position.

nonprocedural command transmission function

2 To receive the near imaging position from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

```
'Recv Command Response
Call fhrunrecvval(fh_socket, retries_recv, timeout_recv, param(), err_no)
'Error check
If err_no <> success Then
    GoTo setcalibmode
End If
IF param(0) <> 0 Then
    GoTo setcalibmode
End If
```

numerical sequence receiving function

Moving the Robot to the Near Imaging Position

The following describes a program that moves the robot to the near imaging position.

1 Store the values of the received near imaging position in variables, then execute the robot motion function (fhsample_move) to move the robot to the acquired near imaging position.

```
'Move to calib position near
cal_pos_near_x = param(1)
cal_pos_near_y = param(2)
cal_pos_near_z = param(3)
cal_pos_near_w = param(4)
cal_pos_near_p = param(5)
cal_pos_near_r = param(6)
CALL fhsample_move(cal_pos_near_x, cal_pos_near_y, cal_pos_near_z, cal_pos_near_w,
                   cal_pos_near_p, cal_pos_near_r, err_no)
If err_no <> success Then
    GoTo setcalibmode
End If
```

Assign the values of the received near imaging position to variables.

Robot motion sample function

⚠WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

Executing a Measurement (Near Imaging Position)

The following describes a program that executes a measurement at the near imaging position.

1 To send the measurement command to the Vision Sensor, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
'Send Measure Command
cmd_name = "MEASURE"
cmd_arg(0) = ""
cmd_arg(1) = ""
cmd_arg(2) = ""
cmd_arg(3) = ""
cmd_arg(4) = ""
cmd_arg_num = 0
Call fhrunsendcmd(fh_socket, cmd_arg_num, cmd_name, cmd_arg(), err_no)
'Error check
If err_no <> success Then
    GoTo setcalibmode
End If
```

nonprocedural command transmission function

2 To receive the response to the measurement command from the Vision Sensor, execute the command response receiving function (fhrunrecvres).
If the response is not OK, execute the processing to end the program.

```
'Recv Command Response
Call fhrunrecvres(fh_socket, retries_recv, timeout_recv, cmd_res, err_no)

'Error check
If err_no <> success Then
    GoTo setcalibmode
End If

'Command Response Check
If cmd_res <> 1 Then
    GoTo setcalibmode
End If
```

command response receiving function

If the response is not OK, exit the program.

3 To receive the measurement result from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval) and check the value of the received overall judgment.
This program is created on the assumption that the measurement result from the Vision Sensor is sent as “TJG”

```
'Get the Measurement Result
Call fhrunrecvval(fh_socket, retries_recv, timeout_recv, param(), err_no)

'Error check
If err_no <> success Then
    GoTo setcalibmode
End If

'Total Judge Check
IF param(0) <> 1 THEN
    GoTo setcalibmode
End If

PrintMsg "Meausure at calib position near done.",-1
```

numerical sequence receiving function

If the overall judgment is not OK, exit the program.

Acquiring the Far Imaging Position

The following describes a program that acquires the far imaging position registered in Automatic Calibration (Vision Master).

1 To acquire the far imaging position, set the first nonprocedural command argument cmd_arg(0) to 1. To send the camera calibration imaging position acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' (6)Calibration position far
'     Get the calib position Far,Meausure at calib position far
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
'Get calib position far
cmd_name = "RBCOM_GET_CALIB_POS"
cmd_arg(0) = "1"
cmd_arg(1) = ""
cmd_arg(2) = ""
cmd_arg(3) = ""
cmd_arg(4) = ""
cmd_arg_num = 1
Call fhrunsendcmd(fh_socket, cmd_arg_num, cmd_name, cmd_arg(), err_no)
'Error check
If err_no <> success Then
    GoTo setcalibmode
End If
```

Set this argument to *1* to acquire the far imaging position.

nonprocedural command transmission function

2 To receive the far imaging position from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

```
'Recv Command Response
Call fhrunrecvval(fh_socket, retries_recv, timeout_recv, param(), err_no)
'Error check
If err_no <> success Then
    GoTo setcalibmode
End If
IF param(0) <> 0 Then
    GoTo setcalibmode
End If
```

numerical sequence receiving function

Moving the Robot to the Far Imaging Position

The following describes a program that moves the robot to the far imaging position.

1 Store the values of the received far imaging position in variables, then execute the robot motion function (fhsample_move) to move the robot to the acquired far imaging position.

```
'Move to calib position far
cal_pos_far_x = param(1)
cal_pos_far_y = param(2)
cal_pos_far_z = param(3)
cal_pos_far_w = param(4)
cal_pos_far_p = param(5)
cal_pos_far_r = param(6)
CALL fhsample_move(cal_pos_far_x, cal_pos_far_y, cal_pos_far_z, cal_pos_far_w,
                   cal_pos_far_p, cal_pos_far_r, err_no)
If err_no <> success Then
    GoTo setcalibmode
End If
```

Assign the values of the received far imaging position to variables.

Robot motion sample function

⚠WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

Executing a Measurement (Far Imaging Position)

The following describes a program that executes a measurement at the far imaging position.

1 To send the measurement command to the Vision Sensor, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
'Send Measure Command
cmd_name = "MEASURE"
cmd_arg(0) = ""
cmd_arg(1) = ""
cmd_arg(2) = ""
cmd_arg(3) = ""
cmd_arg(4) = ""
cmd_arg_num = 0
Call fhrunsendcmd(fh_socket, cmd_arg_num, cmd_name, cmd_arg(), err_no)

'Error check
If err_no <> success Then
    GoTo setcalibmode
End If
```

nonprocedural command transmission function

2 To receive the response to the measurement command from the Vision Sensor, execute the command response receiving function (fhrunrecvres).
If the response is not OK, execute the processing to end the program.

```
'Recv Command Response
Call fhrunrecvres(fh_socket, retries_recv, timeout_recv, cmd_res, err_no)

'Error check
If err_no <> success Then
    GoTo setcalibmode
End If

'Command Response Check
If cmd_res <> 1 Then
    GoTo setcalibmode
End If
```

Command response receiving function

If the response is not OK, exit the program.

3 To receive the measurement result from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval) and check the value of the received overall judgment.
This program is created on the assumption that the measurement result from the Vision Sensor is sent as “TJG”

```
'Get the Measurement Result
Call fhrunrecvval(fh_socket, retries_recv, timeout_recv, param(), err_no)

'Error check
If err_no <> success Then
    GoTo setcalibmode
End If

'Total Judge Check
IF param(0) <> 1 THEN
    GoTo setcalibmode
End If

PrintMsg "Meausure at calib position far done.",-1
```

numerical sequence receiving function

If the overall judgment is not OK, exit the program.

Ending the Program (Normal End)

The following describes processing that is carried out when AOS camera calibration is normally ended.

1 If this processing is successful, the program sends the camera calibration mode change command to manual calibration by the nonprocedural command transmission function (fhrunsendcmd), and then executes the sample program (fhsample_main).

```
    ';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
    ' (7)Set the calibration manual mode
    ';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
    'Set Calib manual mode
    cmd_name = "RBCOM_SET_CALIB_MODE"
    cmd_arg(0) = "0"
    cmd_arg(1) = ""
    cmd_arg(2) = ""
    cmd_arg(3) = ""
    cmd_arg(4) = ""
    cmd_arg_num = 1
    Call fhrunsendcmd(fh_socket, cmd_arg_num, cmd_name, cmd_arg(), err_no)
    'Error check
    If err_no <> success Then
        GoTo disconnect
    End If

    'Recv Command Response
    Call fhrunrecvval(fh_socket, retries_recv, timeout_recv, param(), err_no)
    'Error check
    If err_no <> success Then
        GoTo disconnect
    End If

    IF param(0) <> 0 Then
        GoTo disconnect
    End If

    If bconnected = 1 Then
        Call fhclose(fh_socket, err_no)
    End If

callfhsample:
    PrintMsg "AOS Camera Calibration Successful",-1

    CALL fhsample_main()
    GoTo exitProgram
```

Set the camera calibration mode to manual calibration.

nonprocedural command transmission function

Execute the sample program (fhsample_main).

Ending the Program (Error End)

The following describes processing that is carried out if an error occurs during execution of AOS camera calibration.

1 If an error occurs during execution, the program sends the camera calibration mode change command by the nonprocedural command transmission function (fhrunsendcmd) to set the manual calibration mode, and then stops the program.

```
    ';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
    ' 4. Finalization sequence
    '    Disconnect to the FH server
    '    Send Calib manual mode(manual mode)
    ';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

setcalibmode:
    'Send Calib manual mode Command
    cmd_name = "RBCOM_SET_CALIB_MODE"
    cmd_arg(0) = "0"
    cmd_arg(1) = ""
    cmd_arg(2) = ""
    cmd_arg(3) = ""
    cmd_arg(4) = ""
    cmd_arg_num = 1
    Call fhrunsendcmd(fh_socket, cmd_arg_num, cmd_name, cmd_arg(), err_no)
    Call fhrunrecvval(fh_socket, retries_recv, timeout_recv, param(), err_no)

    PrintMsg "Error in process of Calibration .",-1
    PrintMsg "Set the Calib back to manual mode.",-1

disconnect:
    If bconnected = 1 Then
        Call fhclose(fh_socket, err_no)
    End If

    PrintMsg "Disconnection Done.",-1

exitProgram:
    PrintMsg "EXIT AOS Callibration Program.",-1
```

Set the camera calibration mode to manual calibration.

nonprocedural command transmission function

3.7.5. For Robots Manufactured by ABB Corporation

Connecting Vision Sensor to Robot Controller

The following describes a program that connects the Vision Sensor to the robot controller.

1 Declare internal variables.

* Description omitted (Refer to the source code.)

2 Execute the global variable initialization.

```
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
! (2)Initialaize global variables
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
err_no := success;
! Set Current Coord No
cur_local_coord_no := 0;
cur_tool_coord_no := 0;
```

3 Set the IP address and port number of the Vision Sensor as variables (if the default values need to be changed).

```
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
! 2. Example: Network connection sequence
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
! (1)Set the network configuration
!   You have to configure the following communication settings.
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
ip_address := "10.5.5.100";
port_no := 9876;
retries_connect := 2;
timeout_connect := 4;
retries_recv := 2;
timeout_recv := 4;
```

IP address

Port number

4 With the variables set as arguments, execute the connection function (fhconnect).

```
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
! (2)Connect to the FH server
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
WHILE bconnected = FALSE DO
    fhconnect fh_socket,ip_address, port_no, retries_connect, timeout_connect,
err_no;
    !Error check
    IF err_no <> success THEN
        GOTO exit_program;
    ENDIF
ENDWHILE
```

Connection function with the Vision Sensor (FH server)

Switching Scenes on the Vision Sensor

The following describes a program that switches the scene to a Camera Calibration AOS scene.

1 Set the scene number (127) of the Camera Calibration AOS as a variable.
With the variable set as an argument, execute the scene switching function (fhsample_chgscn).

```
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
! (1)Change the scene of the FH
!    You have to select a scene No. for your application.
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
scene_no := 127;

fhsample_chgscn fh_socket, retries_recv, timeout_recv, scene_no, err_no;

!Error check
IF err_no <> success THEN
   GOTO exit_program;
ENDIF
```

Scene number

Scene switching command execution sample function

Acquiring the Camera Status from the 3D Vision Sensor

The following describes a program that acquires the camera status of the 3D vision sensor and checks if the warmup of the camera is completed.

1 To send the camera status acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CAMERA_STATUS* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
! (2)Get the camera status
!   Check the camera status,When warming wait 1 second until status is OK
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
warmup_flg := 0;

!Send the command of get the camera status
cmd_name := "RBCOM_GET_CAMERA_STATUS";
cmd_arg{1} := "";
cmd_arg{2} := "";
cmd_arg{3} := "";
cmd_arg{4} := "";
cmd_arg{5} := "";
cmd_arg_num := 0;

WHILE warmup_flg = 0 DO
    fhrunsendcmd fh_socket, cmd_name, cmd_arg, cmd_arg_num, err_no;
    IF err_no <> success THEN
        GOTO disconnect;
    ENDIF
```

nonprocedural command transmission function

2 To receive the response to the camera status acquisition command from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

```
    !Get the Camera status Result
    fhrunrecvval fh_socket, retries_recv, timeout_recv, outputdata, err_no;
    IF err_no <> success THEN
        GOTO disconnect;
    ENDIF

    IF outputdata{1} <> 0 THEN
        GOTO disconnect;
    ENDIF
```

numerical sequence receiving function

3 Keep sending the camera status acquisition command to the Vision Sensor until the camera status changes to warmup complete.

```
    IF outputdata{2} = -1 THEN
        warmup_flg := 0;
        WaitTime 1;
```

If the camera status is warmup incomplete, send the camera status acquisition command again.

```
    ELSEIF outputdata{2} = 1 THEN
        warmup_flg := 1;
    ELSE
        GOTO disconnect;
    ENDIF
ENDWHILE
```

If the camera status is warmup complete, proceed to the next processing.

Acquiring the Calibration Date/Time Comparison Result the from 3D Vision Sensor

The following describes a program that checks if the date of calibration of the 3D vision sensor matches the internal date of the Vision Sensor and determines whether the AOS camera calibration must be executed.

1 To send the date/time comparison result acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIBTIME_COMP* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
! (3)Get the Calib time to compare
!   Compare the Galib time to the current time
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

!Send the command of Calib time
cmd_name := "RBCOM_GET_CALIBTIME_COMP";
cmd_arg{1} := "";
cmd_arg{2} := "";
cmd_arg{3} := "";
cmd_arg{4} := "";
cmd_arg{5} := "";
cmd_arg_num := 0;

fhrunsendcmd fh_socket, cmd_name, cmd_arg, cmd_arg_num, err_no;
IF err_no <> success THEN
    GOTO disconnect;
ENDIF
```

nonprocedural command transmission function

2 To receive the response to the calibration date/time comparison result acquisition command from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).
If the date of calibration of the camera matches the internal date of the FH series, execute the sample program (fhsample_main) without executing camera calibration.

```
!Recv Command Response
fhrunrecvval fh_socket, retries_recv, timeout_recv, outputdata, err_no;
IF err_no <> success THEN
```

numerical sequence receiving function

```
    GOTO disconnect;
ENDIF

IF outputdata{1} <> 0 THEN
    GOTO disconnect;
ENDIF
```

If calibration is not required, execute the sample program (fhsample_main).

```
IF outputdata{2} = 1 THEN
    !The calibration date and the current date have been matched
    !Calibration is not necessary

    IF bconnected = TRUE THEN
        fhclose fh_socket, err_no;
    ENDIF

    GOTO call_fhsample;
ENDIF
```

Changing the Calibration Mode of AOS Camera Calibration

The following describes a program that changes the calibration mode of AOS camera calibration to automatic calibration so that AOS camera calibration can be executed during measurement.

1 To change the calibration mode of the AOS camera calibration to automatic calibration, set the first nonprocedural command argument cmd_arg{1}to *1*.
To send the camera calibration mode change command to the Vision Sensor, set the nonprocedural command name to *RBCOM_SET_CALIB_MODE* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
! (4)Send the execute mode command
!   Set the Calib mode to the auto mode
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
!Set Calib auto mode
cmd_name := "RBCOM_SET_CALIB_MODE";
cmd_arg{1} := "1";
cmd_arg{2} := "";
cmd_arg{3} := "";
cmd_arg{4} := "";
cmd_arg{5} := "";
cmd_arg_num := 1;
fhrunsendcmd fh_socket, cmd_name, cmd_arg, cmd_arg_num, err_no;
IF err_no <> success THEN
    GOTO disconnect;
ENDIF
```

Set the camera calibration mode to automatic calibration.

nonprocedural command transmission function

2 To receive the response to the camera calibration mode change command from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

```
!Recv Command Response
fhrunrecvval fh_socket, retries_recv, timeout_recv, outputdata, err_no;
IF err_no <> success THEN
   GOTO disconnect;
ENDIF

IF outputdata{1} <> 0 THEN
   GOTO disconnect;
ENDIF
```

numerical sequence receiving function

Acquiring the Near Imaging Position

The following describes a program that acquires the near imaging position registered in *Automatic Calibration (Vision Master)*.

1 To acquire the near imaging position, set the first nonprocedural command argument cmd_arg{1} to *0*. To send the camera calibration imaging position acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
 ! (5)Calibration position near
 !   Get the calib position near,Meausure at calib position near
 !;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
 !Get calib position near
 cmd_name := "RBCOM_GET_CALIB_POS";
 cmd_arg{1} := "0";
 cmd_arg{2} := "";
 cmd_arg{3} := "";
 cmd_arg{4} := "";
 cmd_arg{5} := "";
 cmd_arg_num := 1;

 fhrunsendcmd fh_socket, cmd_name, cmd_arg, cmd_arg_num, err_no;
 IF err_no <> success THEN
    GOTO set_calibmode;
 ENDIF
```

Set this argument to *0* to acquire the near imaging position.

nonprocedural command transmission function

2 To receive the near imaging position from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

```
 !Recv Command Response
 fhrunrecvval fh_socket, retries_recv, timeout_recv, outputdata, err_no;
 IF err_no <> success THEN
    GOTO set_calibmode;
 ENDIF

 IF outputdata{1} <> 0 THEN
    GOTO set_calibmode;
 ENDIF
```

numerical sequence receiving function

Moving the Robot to the Near Imaging Position

The following describes a program that moves the robot to the near imaging position.

1 Store the values of the received near imaging position in variables. then execute the robot motion function (fhsample_move) to move the robot to the acquired near imaging position.

```
!Move to calib position near
cal_pos_near_x := outputdata{2};
cal_pos_near_y := outputdata{3};
cal_pos_near_z := outputdata{4};
cal_pos_near_w := outputdata{5};
cal_pos_near_p := outputdata{6};
cal_pos_near_r := outputdata{7};
fhsample_move cal_pos_near_x, cal_pos_near_y, cal_pos_near_z, cal_pos_near_w,
cal_pos_near_p, cal_pos_near_r, err_no;
IF err_no <> success THEN
    GOTO set_calibmode;
ENDIF
```

Assign the values of the received near imaging position to variables.

Robot motion sample function

⚠WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

Executing a Measurement (Near Imaging Position)

The following describes a program that executes a measurement at the near imaging position.

1 To send the measurement command to the Vision Sensor, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
cmd_name := "MEASURE";
cmd_arg{1} := "";
cmd_arg{2} := "";
cmd_arg{3} := "";
cmd_arg{4} := "";
cmd_arg{5} := "";
cmd_arg_num := 0;
fhrunsendcmd fh_socket, cmd_name, cmd_arg, cmd_arg_num, err_no;
IF err_no <> success THEN
    GOTO set_calibmode;
ENDIF
```

nonprocedural command transmission function

2 To receive the response to the measurement command from the Vision Sensor, execute the command response receiving function (fhrunrecvres).
If the response is not OK, execute the processing to end the program.

```
!Recv Command Response
fhrunrecvres fh_socket, retries_recv, timeout_recv, cmd_res, err_no;
IF err_no <> success THEN
    GOTO set_calibmode;
ENDIF
!Command Response Check
IF cmd_res <> 1 THEN
    GOTO set_calibmode;
ENDIF
```

command response receiving function

If the response is not OK, exit the program.

3 To receive the measurement result from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval) and check the value of the received overall judgment.
This program is created on the assumption that the measurement result from the Vision Sensor is sent as “TJG”

```
!Get the Measurement Result
fhrunrecvval fh_socket, retries_recv, timeout_recv, outputdata, err_no;
IF err_no <> success THEN
    GOTO set_calibmode;
ENDIF
!Total Judge Check
IF outputdata{1} <> 1 THEN
    GOTO set_calibmode;
ENDIF
```

numerical sequence receiving function

If the overall judgment is not OK, exit the program

Acquiring the Far Imaging Position

The following describes a program that acquires the far imaging position registered in *Automatic Calibration (Vision Master)*.

1 To acquire the far imaging position, set the first nonprocedural command argument cmd_arg{1} to *1*. To send the camera calibration imaging position acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
! (6)Calibration position far
!   Get the calib position Far,Meausure at calib position far
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
!Get Calib Position Far
cmd_name := "RBCOM_GET_CALIB_POS";
cmd_arg{1} := "1";
cmd_arg{2} := "";
cmd_arg{3} := "";
cmd_arg{4} := "";
cmd_arg{5} := "";
cmd_arg_num := 1;
fhrunsendcmd fh_socket, cmd_name, cmd_arg, cmd_arg_num, err_no;
IF err_no <> success THEN
   GOTO set_calibmode;
ENDIF
```

Set this argument to *1* to acquire the far imaging position.

nonprocedural command transmission function

2 To receive the far imaging position from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

```
!Recv Command Response
fhrunrecvval fh_socket, retries_recv, timeout_recv, outputdata, err_no;
IF err_no <> success THEN
   GOTO set_calibmode;
ENDIF

IF outputdata{1} <> 0 THEN
   GOTO set_calibmode;
ENDIF
```

nonprocedural command transmission function

Moving the Robot to the Far Imaging Position

The following describes a program that moves the robot to the far imaging position.

1 Store the values of the received far imaging position in variables. Then execute the robot motion function (fhsample_move) to move the robot to the acquired far imaging position.

```
!Move to calib position far
cal_pos_far_x := outputdata{2};
cal_pos_far_y := outputdata{3};
cal_pos_far_z := outputdata{4};
cal_pos_far_w := outputdata{5};
cal_pos_far_p := outputdata{6};
cal_pos_far_r := outputdata{7};
fhsample_move cal_pos_far_x, cal_pos_far_y, cal_pos_far_z, cal_pos_far_w,
cal_pos_far_p, cal_pos_far_r, err_no;
IF err_no <> success THEN
    GOTO set_calibmode;
ENDIF
```

Assign the values of the received far imaging position to variables.

Robot motion sample function

⚠WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

Executing a Measurement (Far Imaging Position)

The following describes a program that executes a measurement at the far imaging position.

1 To send the measurement command to the Vision Sensor, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
cmd_name := "MEASURE";
cmd_arg{1} := "";
cmd_arg{2} := "";
cmd_arg{3} := "";
cmd_arg{4} := "";
cmd_arg{5} := "";
cmd_arg_num := 0;
fhrunsendcmd fh_socket, cmd_name, cmd_arg, cmd_arg_num, err_no;
IF err_no <> success THEN
    GOTO set_calibmode;
ENDIF
```

nonprocedural command transmission function

2 To receive the response to the measurement command from the Vision Sensor, execute the command response receiving function (fhrunrecvres).
If the response is not OK, execute the processing to end the program.

```
!Recv Command Response
fhrunrecvres fh_socket, retries_recv, timeout_recv, cmd_res, err_no;
IF err_no <> success THEN
    GOTO set_calibmode;
ENDIF
!Command Response Check
IF cmd_res <> 1 THEN
    GOTO set_calibmode;
ENDIF
```

Command response receiving function

If the response is not OK, exit the program.

3 To receive the measurement result from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval) and check the value of the received overall judgment.
This program is created on the assumption that the measurement result from the Vision Sensor is sent as “TJG”

```
!Get the Measurement Result
fhrunrecvval fh_socket, retries_recv, timeout_recv, outputdata, err_no;
IF err_no <> success THEN
    GOTO set_calibmode;
ENDIF
!Total Judge Check
IF outputdata{1} <> 1 THEN
    GOTO set_calibmode;
ENDIF
```

nonprocedural command transmission function

If the overall judgment is not OK, exit the program

Ending the Program (Normal End)

The following describes processing that is carried out when AOS camera calibration is normally ended.

1 If this processing is successful, the program sends the camera calibration mode change command to manual calibration by the nonprocedural command transmission function (fhrunsendcmd), and then executes the sample program (fhsample_main).

```
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
! (7)Set the calibration mode to manual
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
!Set Calib manual mode
cmd_name := "RBCOM_SET_CALIB_MODE";
cmd_arg{1} := "0";
cmd_arg{2} := "";
cmd_arg{3} := "";
cmd_arg{4} := "";
cmd_arg{5} := "";
cmd_arg_num := 1;

fhrunsendcmd fh_socket, cmd_name, cmd_arg, cmd_arg_num, err_no;
IF err_no <> success THEN
   GOTO disconnect;
ENDIF

!Recv Command Response
fhrunrecvval fh_socket, retries_recv, timeout_recv, outputdata, err_no;
IF err_no <> success THEN
   GOTO disconnect;
ENDIF

IF outputdata{1} <> 0 THEN
   GOTO disconnect;
ENDIF

IF bconnected = TRUE THEN
   fhclose fh_socket, err_no;
ENDIF

call_fhsample:
!AOS Camera Calibration Successful
fhsample_main;
GOTO exit_program;
```

Set the camera calibration mode to manual calibration.

nonprocedural command transmission function

Execute the sample program (fhsample_main).

Ending the Program (Error End)

The following describes processing that is carried out if an error occurs during execution of AOS camera calibration.

1 If an error occurs during execution, the program sends the camera calibration mode change command by the nonprocedural command transmission function (fhrunsendcmd) to set the manual calibration mode, and then stops the program.

```
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
! 4. Finalization sequence
!    Disconnect to the FH server
!    Send Calib manual mode(manual mode)
!;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
set_calibmode:
!Send Calib manual mode Command
cmd_name := "RBCOM_SET_CALIB_MODE";
cmd_arg{1} := "0";
cmd_arg{2} := "";
cmd_arg{3} := "";
cmd_arg{4} := "";
cmd_arg{5} := "";
cmd_arg_num := 1;
fhrunsendcmd fh_socket, cmd_name, cmd_arg, cmd_arg_num, err_no;
fhrunrecvval fh_socket, retries_recv, timeout_recv, outputdata, err_no;

disconnect:
IF bconnected = TRUE THEN
   fhclose fh_socket, err_no;
ENDIF

exit_program:
!Exit the program
```

Set the camera calibration mode to manual calibration.

nonprocedural command transmission function

3.7.6. For Robots Manufactured by YASKAWA Electric Corporation

Connecting Vision Sensor to Robot Controller

The following describes a program that connects the Vision Sensor to the robot controller.

1 Set internal variables with the programming pendant.

User variable name	Description	Setting value
I093	Local Coordinate System No.(only 0)	0
I094	Tool Coordinate System No.(0 to 63)	0 - 63
I095	Number of connection retries(0 to 60)	0 - 60
I096	Time of connection timeout(1 to 60[sec])	1 - 60
I097	Number of received retries(0 to 99)	0 - 99
I098	Time of Receive timeout(0 to 99[sec])	0 - 99

2 Set the operating mode variable “B099” to 2.

3 Use the set variable as an argument of the connection function (FHCONNECT) to the Vision Sensor (FH server) and execute the function.

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' Connect to the FH server
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
SET S089 "FHCONNECT"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *EXIT
ENDIF
```

Connection function with vision sensor (FH server)

Switching Scenes on the Vision Sensor

The following describes a program that switches the scene to a Camera Calibration AOS scene.

1 Set the scene number (127) of the Camera Calibration AOS as a variable.
Set the scene switching command in the argument of the nonprocedural command transmission function (FHRUNSENDCMD) for the vision sensor, execute the function.

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' (1) Change the scene of the FH
'  You have to select a scene No.
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
SET S089 "FHRUNSENDCMD"
SET B097 1
SET S090 "SCENE"
SET S091 "127"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *EXIT
ENDIF
```

nonprocedural command transmission function

Scene switching command

Scene number

2 Executes the command response receiving function (FHRUNRECVRES) for the vision sensor.

```
SET S089 "FHRUNRECVRES"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *EXIT
ENDIF
IFTHENEXP R090<>1
      JUMP *EXIT
ENDIF
```

command response receiving function

Acquiring the Camera Status from the 3D Vision Sensor

The following describes a program that acquires the camera status of the 3D vision sensor and checks if the warmup of the camera is completed.

1 To send the camera status acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CAMERA_STATUS* and execute the nonprocedural command transmission function (FHRUNSENDCMD).

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' (2) Check the camera status.
'  When warming wait 1 second until status is OK.
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
SET I080 0
WHILEEXP I080=0
      SET S090 "RBCOM_GET_CAMERA_STATUS"
      SET B097 0
      SET S089 "FHRUNSENDCMD"
      SET B098 1
      WAIT B098=0
      IFTHENEXP I099<>0
            JUMP *DISC
      ENDIF
```

nonprocedural command transmission function

2 To receive the response to the camera status acquisition command from the Vision Sensor, execute the numerical sequence receiving function (FHRUNRECVVAL).

```
      SET S089 "FHRUNRECVVAL"
      SET B098 1
      WAIT B098=0
      IFTHENEXP I099<>0
            JUMP *DISC
      ENDIF
      IFTHENEXP R090<>0
            JUMP *DISC
      ENDIF
```

numerical sequence receiving function

3 Keep sending the camera status acquisition command to the Vision Sensor until the camera status changes to warmup complete.

```
      IFTHENEXP R091=-1
            SET I080 0
            TIMER T=1
      ELSEIFEXP R091=1
            SET I080 1
      ELSE
            JUMP *DISC
      ENDIF
ENDWHILE
```

If the camera status is warmup incomplete, send the camera status acquisition command again.

If the camera status is warmup complete, proceed to the next processing.

Acquiring the Calibration Date/Time Comparison Result the from 3D Vision Sensor

The following describes a program that checks if the date of calibration of the 3D vision sensor matches the internal date of the Vision Sensor and determines whether the AOS camera calibration must be executed.

1 To send the date/time comparison result acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIBTIME_COMP* and execute the nonprocedural command transmission function (FHRUNSENDCMD).

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' (3) Get the Calib time compare
'  Compare the Calib time
'  to the current time
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
SET S090 "RBCOM_GET_CALIBTIME_COMP"
SET B097 0
SET S089 "FHRUNSENDCMD"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *DISC
ENDIF
```

nonprocedural command transmission function

2 To receive the response to the calibration date/time comparison result acquisition command from the Vision Sensor, execute the numerical sequence receiving function (FHRUNRECVVAL).
If the date of calibration of the camera matches the internal date of the FH series, execute the sample job (FHSMPLMAIN) without executing camera calibration.

```
SET S089 "FHRUNRECVVAL"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *DISC
ENDIF
IFTHENEXP R090<>0
      JUMP *DISC
ENDIF
IFTHENEXP R091=1
      SET S089 "FHCLOSE"
      SET B098 1
      WAIT B098=0
      JUMP *CSML
ENDIF
```

numerical sequence receiving function

If calibration is not required,
execute the sample job (FHSMPLMAIN).

Changing the Calibration Mode of AOS Camera Calibration

The following describes a program that changes the calibration mode of AOS camera calibration to automatic calibration so that AOS camera calibration can be executed during measurement.

1 To change the calibration mode of the AOS camera calibration to automatic calibration, set the first nonprocedural command argument “S091” to 1.
To send the camera calibration mode change command to the Vision Sensor, set the nonprocedural command name to *RBCOM_SET_CALIB_MODE* and execute the nonprocedural command transmission function (FHRUNSENDCMD).

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' (4) Send the auto mode command
'  Set the Calib mode
'  to the auto mode
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
SET S090 "RBCOM_SET_CALIB_MODE"
SET S091 "1"
SET B097 1
SET S089 "FHRUNSENDCMD"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *DISC
ENDIF
```

Set the camera calibration mode to automatic calibration.

nonprocedural command transmission function

2 To receive the response to the camera calibration mode change command from the Vision Sensor, execute the numerical sequence receiving function (FHRUNRECVVAL).

numerical sequence receiving function

```
SET S089 "FHRUNRECVVAL"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *DISC
ENDIF
IFTHENEXP R090<>0
      JUMP *DISC
ENDIF
```

Acquiring the Near Imaging Position

The following describes a program that acquires the near imaging position registered in *Automatic Calibration (Vision Master)*.

1 To acquire the near imaging position, set the first nonprocedural command argument “S091” to 0. To send the camera calibration imaging position acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (FHRUNSENDCMD).

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' (5) Calibration position near
'  Get the calib position near
'  Meausure calib position near
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
SET S090 "RBCOM_GET_CALIB_POS"
SET S091 "0"
SET B097 1
SET S089 "FHRUNSENDCMD"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *MMOD
ENDIF
```

Set this argument to 0 to acquire the near imaging position.

nonprocedural command transmission function

2 To receive the near imaging position from the Vision Sensor, execute the numerical sequence receiving function (FHRUNRECVVAL).

```
SET S089 "FHRUNRECVVAL"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *MMOD
ENDIF
IFTHENEXP R090<>0
      JUMP *MMOD
ENDIF
```

numerical sequence receiving function

Moving the Robot to the Near Imaging Position

The following describes a program that moves the robot to the near imaging position.

1 Store the values of the received near imaging position in variables, execute the robot motion command (MOVJ) to move the robot to the acquired near imaging position.

```
'Move to calib position near
SET D099  EXPRESS R091 * 1000
SETE P099 (1) D099
SET D099  EXPRESS R092 * 1000
SETE P099 (2) D099
SET D099  EXPRESS R093 * 1000
SETE P099 (3) D099
SET D099  EXPRESS R094 * 10000
SETE P099 (4) D099
SET D099  EXPRESS R095 * 10000
SETE P099 (5) D099
SET D099  EXPRESS R096 * 10000
SETE P099 (6) D099
SETE P099 TL#(0)
'!!!!!!!!!!! CAUTION !!!!!!!!!!
'!The following function drives
'!a robot motion immediately.
'!Confirm the settings
'!before execution.
'!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
MOVJ P099 VJ=10.00
```

Assign the values of the received near imaging position to variables.

Robot motion command

⚠ WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

Executing a Measurement (Near Imaging Position)

The following describes a program that executes a measurement at the near imaging position.

1 To send the measurement command to the Vision Sensor, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (FHRUNSENDCMD).

```
'Send Measure Command
SET S090 "MEASURE"
SET B097 0
SET S089 "FHRUNSENDCMD"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *MMOD
ENDIF
```

nonprocedural command transmission function

2 To receive the response to the measurement command from the Vision Sensor, execute the command response receiving function (FHRUNRECVRES).
If the response is not OK, execute the processing to end the program.

```
SET S089 "FHRUNRECVRES"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *MMOD
ENDIF
IFTHENEXP R090<>1
      JUMP *MMOD
ENDIF
```

command response receiving function

If the response is not OK, exit the program.

3 To receive the measurement result from the Vision Sensor, execute the numerical sequence receiving function (FHRUNRECVVAL) and check the value of the received overall judgment.
This program is created on the assumption that the measurement result from the Vision Sensor is sent as “TJG”

```
SET S089 "FHRUNRECVVAL"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *MMOD
ENDIF
IFTHENEXP R090<>1
      JUMP *MMOD
ENDIF
```

numerical sequence receiving function

If the overall judgment is not OK, exit the program.

Acquiring the Far Imaging Position

The following describes a program that acquires the far imaging position registered in *Automatic Calibration (Vision Master)*.

1 To acquire the far imaging position, set the first nonprocedural command argument “S091” to 1. To send the camera calibration imaging position acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (FHRUNSENDCMD).

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' (6) Calibration position far
'  Get the calib position far
'  Meausure at calib position far
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
SET S090 "RBCOM_GET_CALIB_POS"
SET S091 "1"
SET B097 1
SET S089 "FHRUNSENDCMD"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *MMOD
ENDIF
```

Set this argument to *1* to acquire the far imaging position.

nonprocedural command transmission function

2 To receive the far imaging position from the Vision Sensor, execute the numerical sequence receiving function (FHRUNRECVVAL).

```
SET S089 "FHRUNRECVVAL"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *MMOD
ENDIF
IFTHENEXP R090<>0
      JUMP *MMOD
ENDIF
```

numerical sequence receiving function

If the overall judgment is not OK, exit the program

Moving the Robot to the Far Imaging Position

The following describes a program that moves the robot to the far imaging position.

1 Store the values of the received far imaging position in variables, execute the robot motion command (MOVJ) to move the robot to the acquired far imaging position.

```
'Move to calib position far
SET D099  EXPRESS R091 * 1000
SETE P099 (1) D099
SET D099  EXPRESS R092 * 1000
SETE P099 (2) D099
SET D099  EXPRESS R093 * 1000
SETE P099 (3) D099
SET D099  EXPRESS R094 * 10000
SETE P099 (4) D099
SET D099  EXPRESS R095 * 10000
SETE P099 (5) D099
SET D099  EXPRESS R096 * 10000
SETE P099 (6) D099
SETE P099 TL#(0)
'!!!!!!!!!!! CAUTION !!!!!!!!!!!
'!The following function drives
'!a robot motion immediately.
'!Confirm the settings
'!before execution.
'!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
MOVJ P099 VJ=10.00
```

Assign the values of the received far imaging position to variables.

Robot motion command

⚠ WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

Executing a Measurement (Far Imaging Position)

The following describes a program that executes a measurement at the far imaging position.

1 To send the measurement command to the Vision Sensor, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (FHRUNSENDCMD).

```
'Send Measure Command
SET S090 "MEASURE"
SET B097 0
SET S089 "FHRUNSENDCMD"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *MMOD
ENDIF
```

nonprocedural command transmission function

2 To receive the response to the measurement command from the Vision Sensor, execute the command response receiving function (FHRUNRECVERS).
If the response is not OK, execute the processing to end the program.

```
SET S089 "FHRUNRECVRES"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *MMOD
ENDIF
IFTHENEXP R090<>1
      JUMP *MMOD
ENDIF
```

Command response receiving function

If the response is not OK, exit the program.

3 To receive the measurement result from the Vision Sensor, execute the numerical sequence receiving function (FHRUNRECVVAL) and check the value of the received overall judgment.
This program is created on the assumption that the measurement result from the Vision Sensor is sent as “TJG”

```
SET S089 "FHRUNRECVVAL"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *MMOD
ENDIF
IFTHENEXP R090<>1
      JUMP *MMOD
ENDIF
```

numerical sequence receiving function

If the overall judgment is not OK, exit the program.

Ending the Program (Normal End)

The following describes processing that is carried out when AOS camera calibration is normally ended.

1 If this processing is successful, the program sends the camera calibration mode change command to manual calibration by the nonprocedural command transmission function (FHRUNSENDCMD), and then executes the sample job (FHSMPLMAIN).

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' (7) Set the calibration
'  manual mode
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
SET S090 "RBCOM_SET_CALIB_MODE"
SET S091 "0"
SET B097 1
SET S089 "FHRUNSENDCMD"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *DISC
ENDIF
SET S089 "FHRUNRECVVAL"
SET B098 1
WAIT B098=0
IFTHENEXP I099<>0
      JUMP *DISC
ENDIF
IFTHENEXP R090<>0
      JUMP *DISC
ENDIF
SET S089 "FHCLOSE"
SET B098 1
WAIT B098=0
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' AOS successful call FHSMPLMAIN
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
*CSML
CALL JOB:FHSMPLMAIN
JUMP *EXIT
```

Set the camera calibration mode to manual calibration.

nonprocedural command transmission function

Execute the sample job (FHSMPLMAIN).

Ending the Program (Error End)

The following describes processing that is carried out if an error occurs during execution of AOS camera calibration.

1 If an error occurs during execution, the program sends the camera calibration mode change command by the nonprocedural command transmission function (FHRUNSENDCMD) to set the manual calibration mode, and then stops the program.

```
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
'4. Example: Finalization sequene
'  of a finalization.
'  Set Execute manual mode
'  Disconnect to the FH server
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
*MMOD
SET S090 "RBCOM_SET_CALIB_MODE"
SET S091 "0"
SET B097 1
SET S089 "FHRUNSENDCMD"
SET B098 1
WAIT B098=0
SET S089 "FHRUNRECVVAL"
SET B098 1
WAIT B098=0
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' Disconnect to the FH server
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
*DISC
SET S089 "FHCLOSE"
SET B098 1
WAIT B098=0
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
' Exit the Job
';;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
*EXIT
'EXIT AOS Callibration Job
END
```

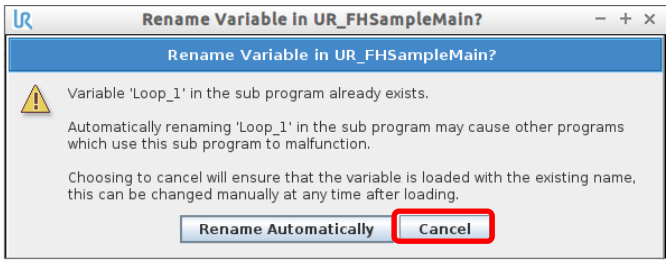
Set the camera calibration mode to manual calibration.

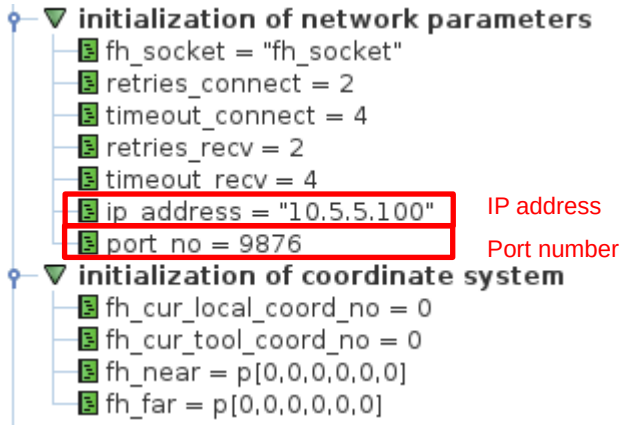
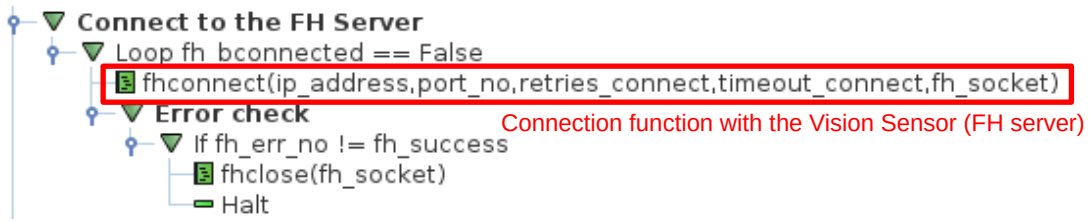
nonprocedural command transmission function

3.7.7. For Robots Manufactured by UNIVERSAL ROBOTS

Connecting Vision Sensor to Robot Controller

The following describes a program that connects the Vision Sensor to the robot controller.
When loading the robot program (“UR_FHSampleCalaos.urp”) into the robot controller, the following display may appear, but press [Cancel].

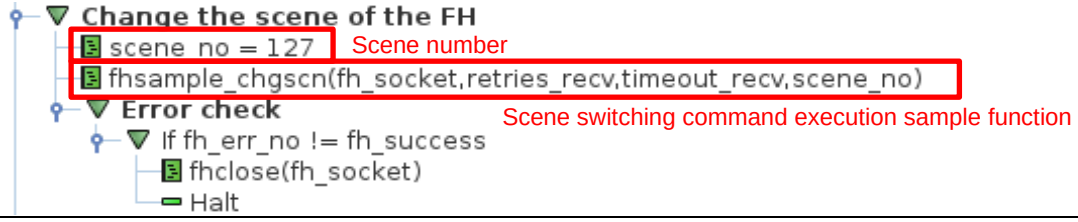


1	Declare internal variables. * Description omitted (Refer to the source code.)
2	Execute the global variable initialization function (fhdefglobal). 
3	Set the IP address and port number of the Vision Sensor as variables (if the default values need to be changed). 
4	With the variables set as arguments, execute the connection function (fhconnect). 

Switching Scenes on the Vision Sensor

The following describes a program that switches the scene to a Camera Calibration AOS scene.

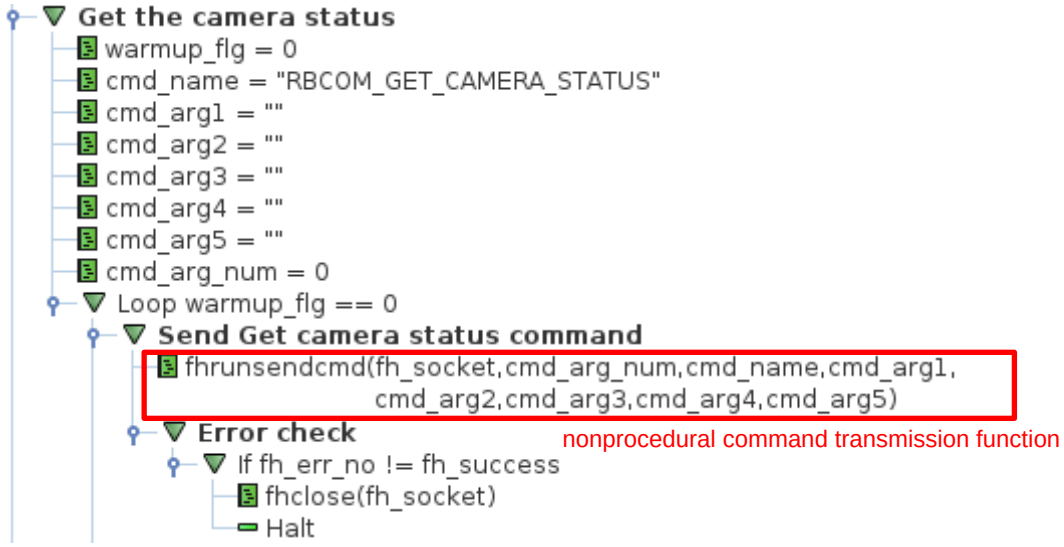
1 Set the scene number (127) of the Camera Calibration AOS as a variable.
With the variable set as an argument, execute the scene switching function (fhsample_chgscn).



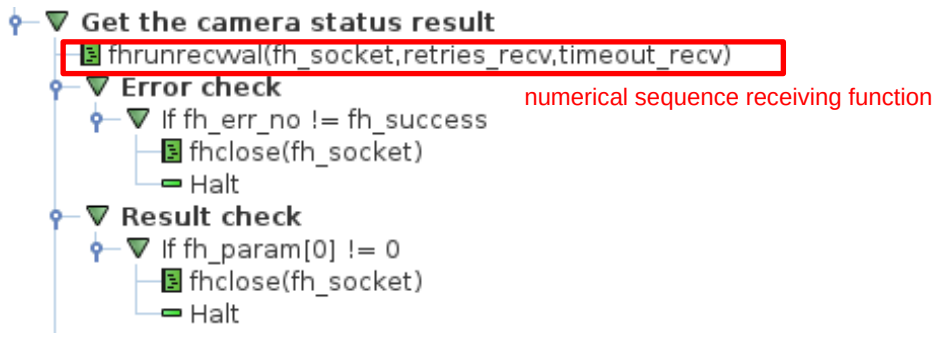
Acquiring the Camera Status from the 3D Vision Sensor

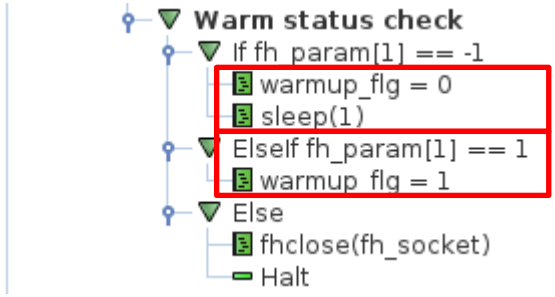
The following describes a program that acquires the camera status of the 3D vision sensor and checks if the warmup of the camera is completed.

1 To send the camera status acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CAMERA_STATUS* and execute the nonprocedural command transmission function (fhrunsendcmd).



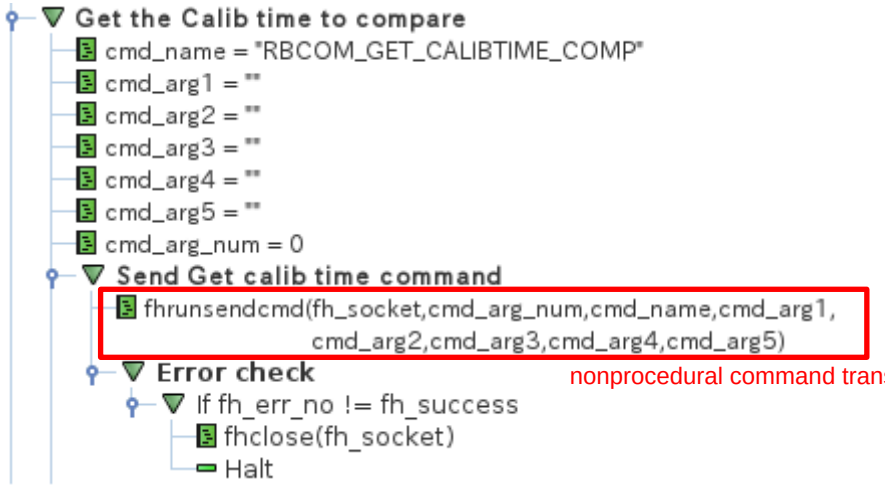
2 To receive the response to the camera status acquisition command from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

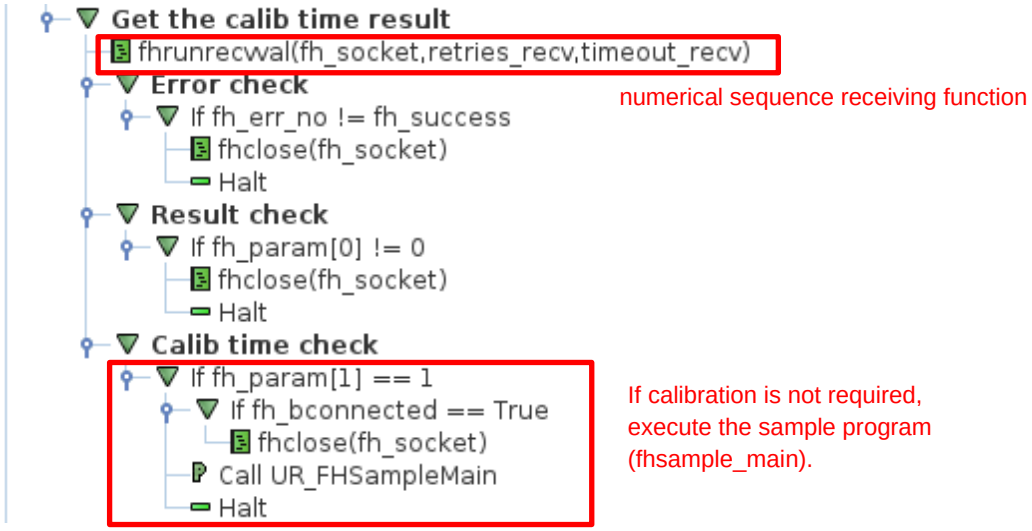


3	Keep sending the camera status acquisition command to the Vision Sensor until the camera status changes to warmup complete.  If the camera status is warmup incomplete, send the camera status acquisition command again. If the camera status is warmup complete, proceed to the next processing.

Acquiring the Calibration Date/Time Comparison Result the from 3D Vision Sensor

The following describes a program that checks if the date of calibration of the 3D vision sensor matches the internal date of the Vision Sensor and determines whether the AOS camera calibration must be executed.

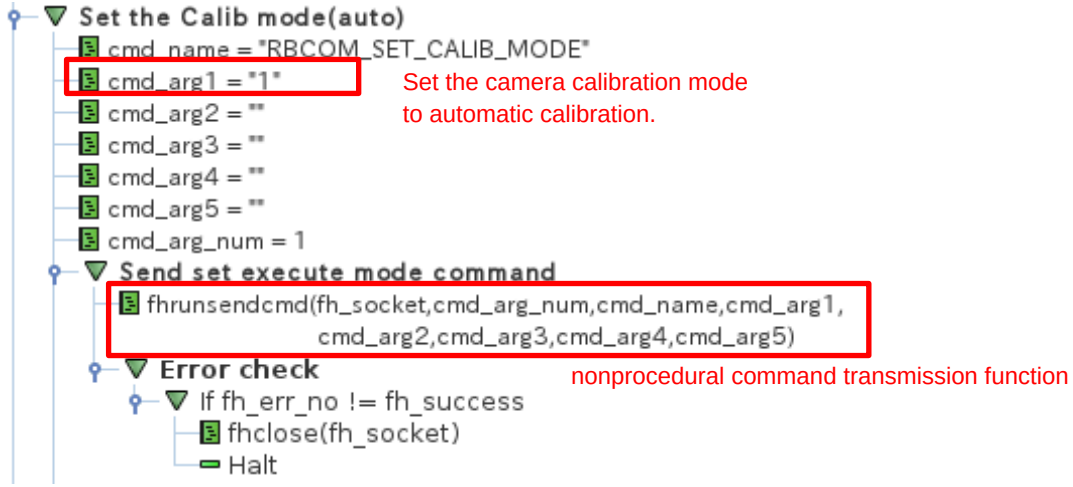
1	To send the date/time comparison result acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIBTIME_COMP* and execute the nonprocedural command transmission function (fhrunsendcmd).  nonprocedural command transmission function
2	To receive the response to the calibration date/time comparison result acquisition command from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval). If the date of calibration of the camera matches the internal date of the FH series, execute the sample program (fhsample_main) without executing camera calibration.



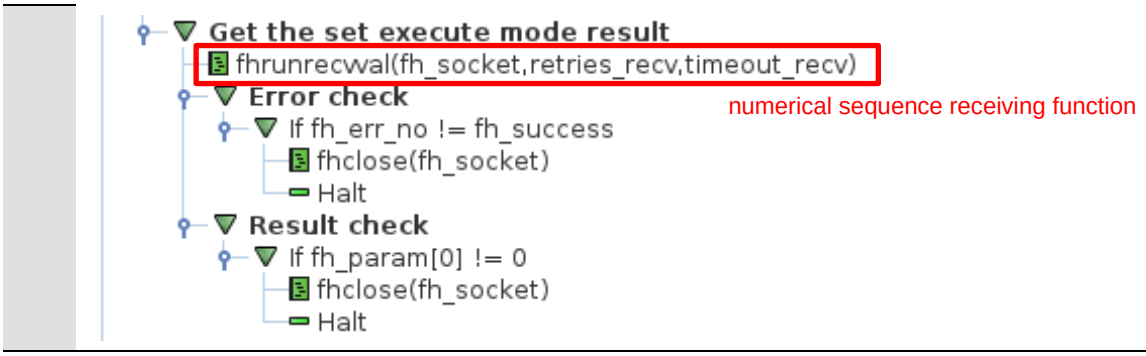
Changing the Calibration Mode of AOS Camera Calibration

The following describes a program that changes the calibration mode of AOS camera calibration to automatic calibration so that AOS camera calibration can be executed during measurement.

1 To change the calibration mode of the AOS camera calibration to automatic calibration, set the first nonprocedural command argument cmd_arg1 to 1.
To send the camera calibration mode change command to the Vision Sensor, set the nonprocedural command name to *RBCOM_SET_CALIB_MODE* and execute the nonprocedural command transmission function (fhrunsendcmd).



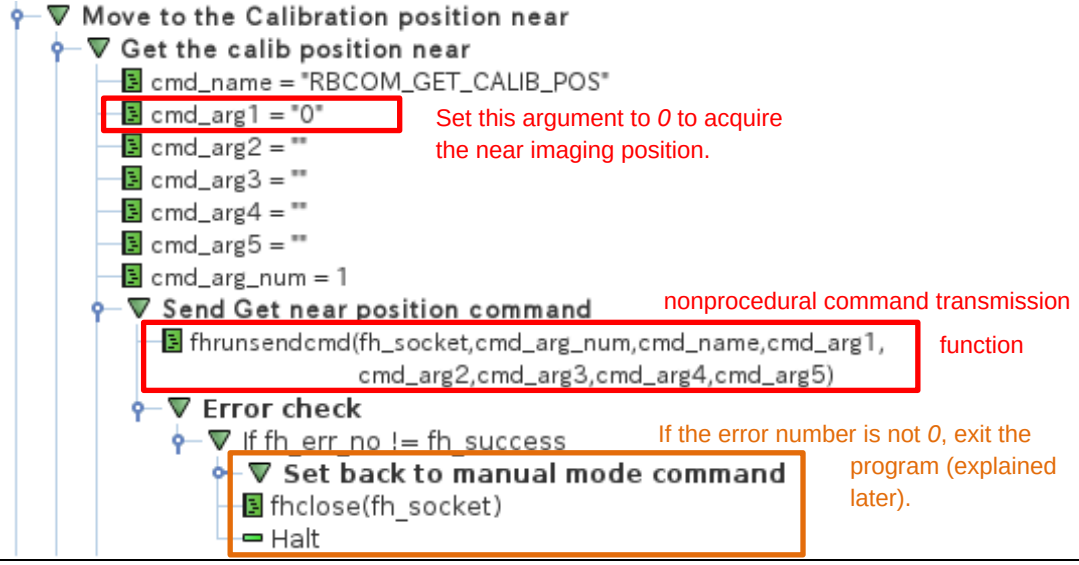
2 To receive the response to the camera calibration mode change command from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).



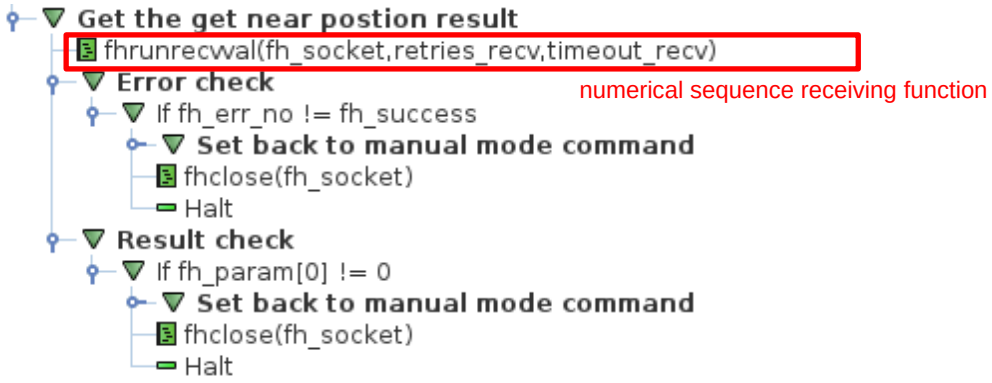
Acquiring the Near Imaging Position

The following describes a program that acquires the near imaging position registered in Automatic Calibration (Vision Master).

1 To acquire the near imaging position, set the first nonprocedural command argument cmd_arg1 to 0. To send the camera calibration imaging position acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (fhrunsendcmd).



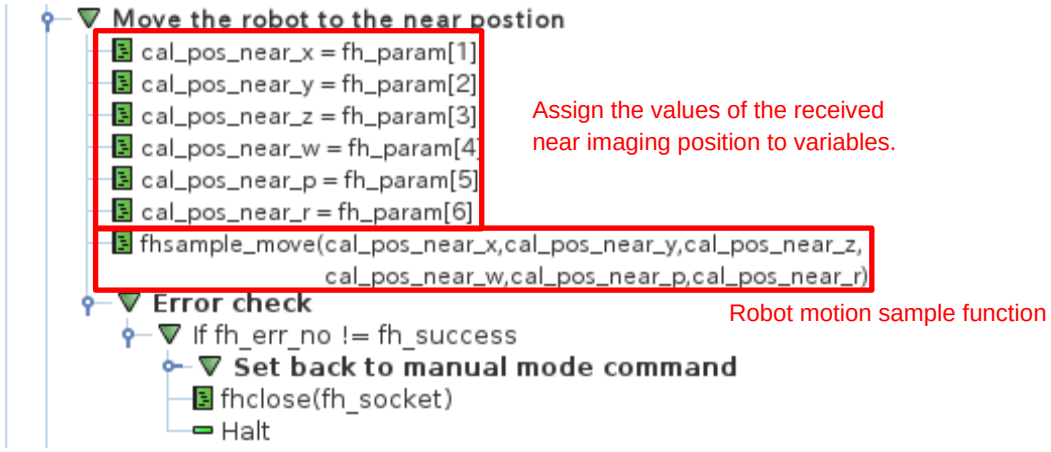
2 To receive the near imaging position from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).



Moving the Robot to the Near Imaging Position

The following describes a program that moves the robot to the near imaging position.

1 Store the values of the received near imaging position in variables, then execute the robot motion function (fhsample_move) to move the robot to the acquired near imaging position.



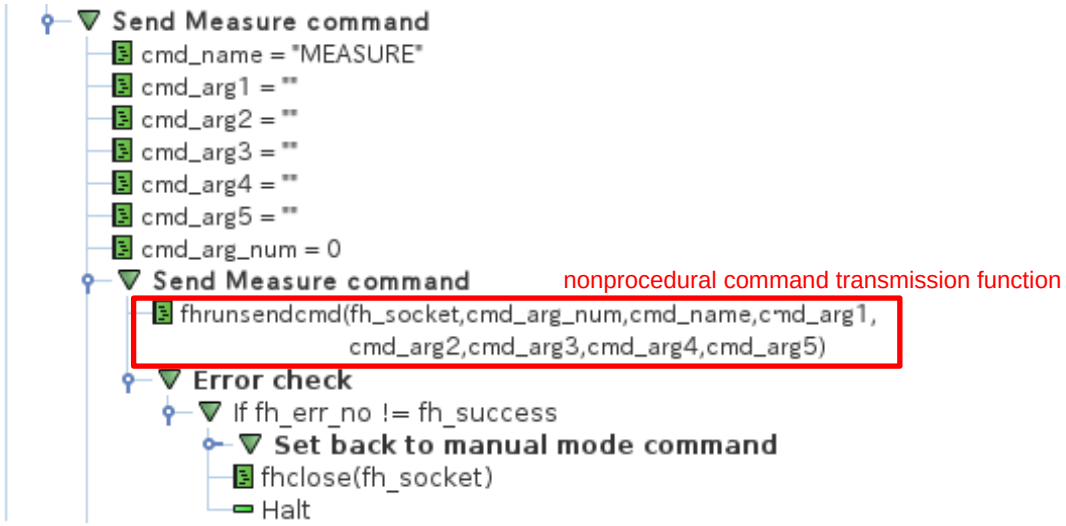
⚠WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

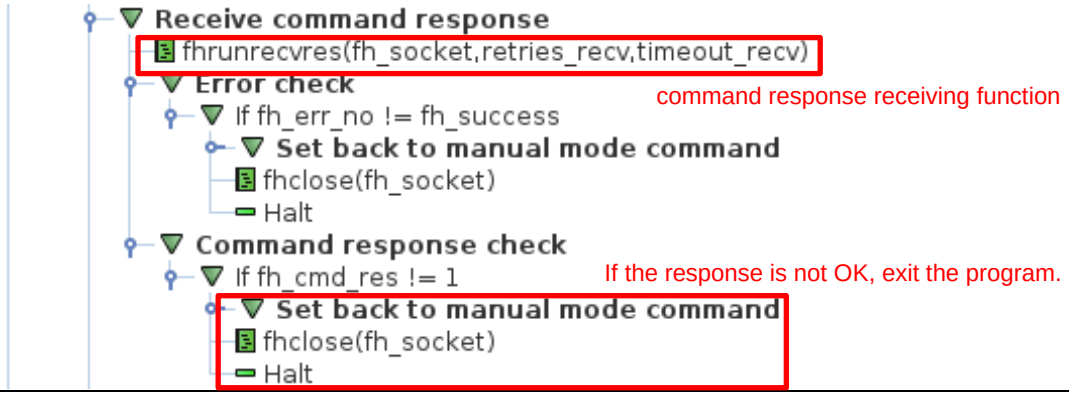
Executing a Measurement (Near Imaging Position)

The following describes a program that executes a measurement at the near imaging position.

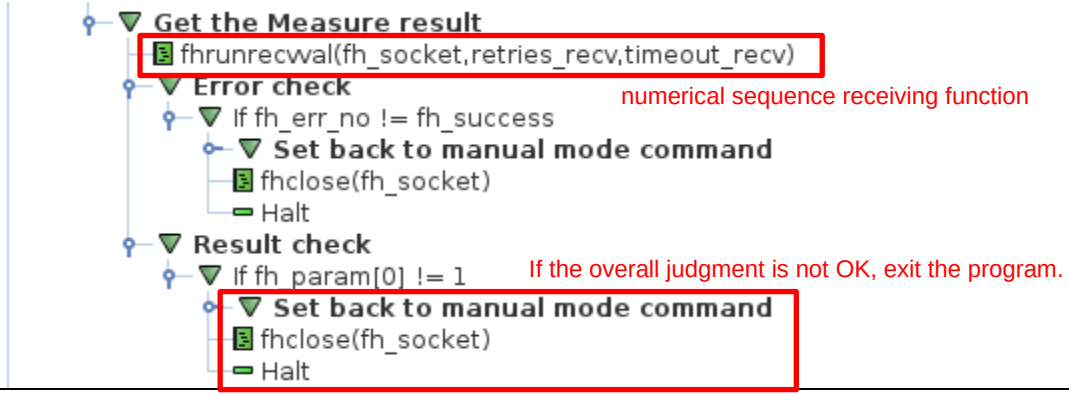
1 To send the measurement command to the Vision Sensor, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (fhrunsendcmd).



2 To receive the response to the measurement command from the Vision Sensor, execute the command response receiving function (fhrunrecvres).
If the response is not OK, execute the processing to end the program.



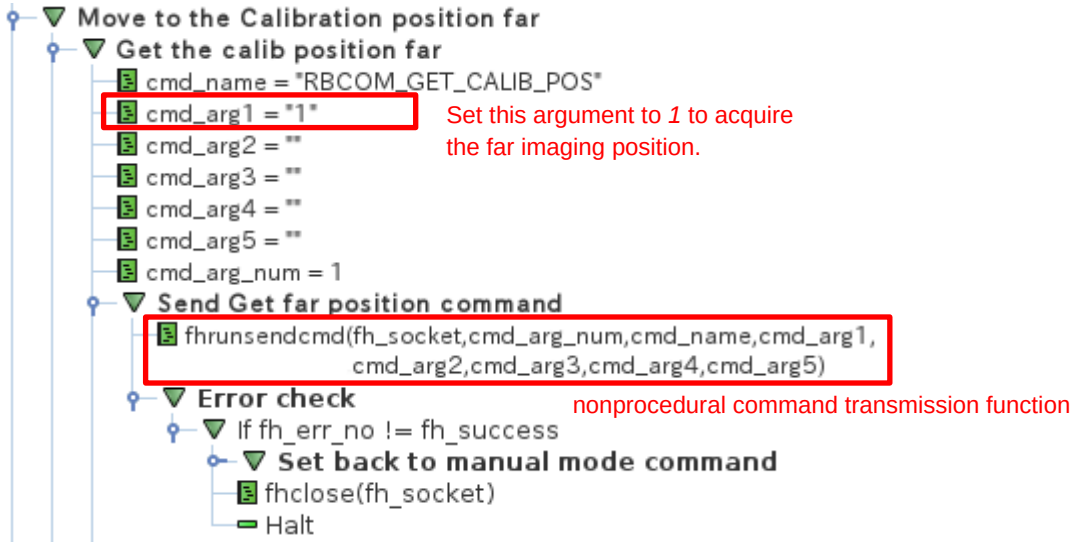
3 To receive the measurement result from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval) and check the value of the received overall judgment.
This program is created on the assumption that the measurement result from the Vision Sensor is sent as “TJG”



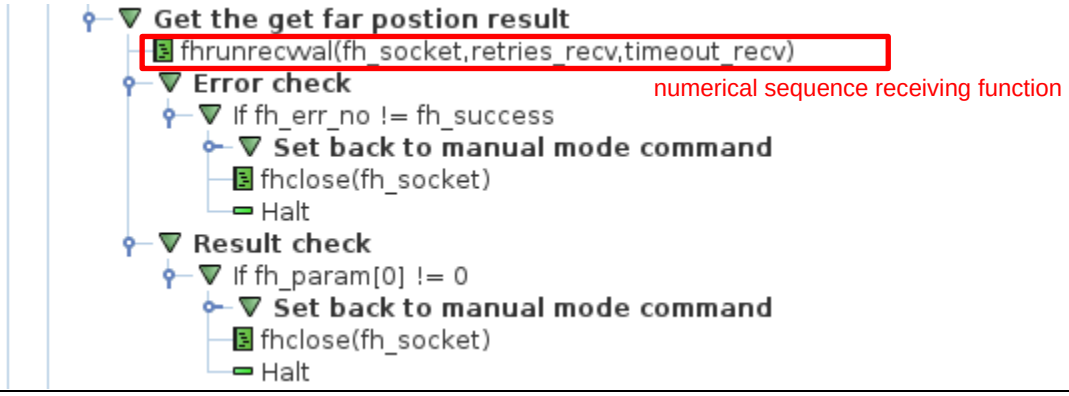
Acquiring the Far Imaging Position

The following describes a program that acquires the far imaging position registered in Automatic Calibration (Vision Master).

1 To acquire the far imaging position, set the first nonprocedural command argument cmd_arg1 to 1. To send the camera calibration imaging position acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (fhrunsendcmd).



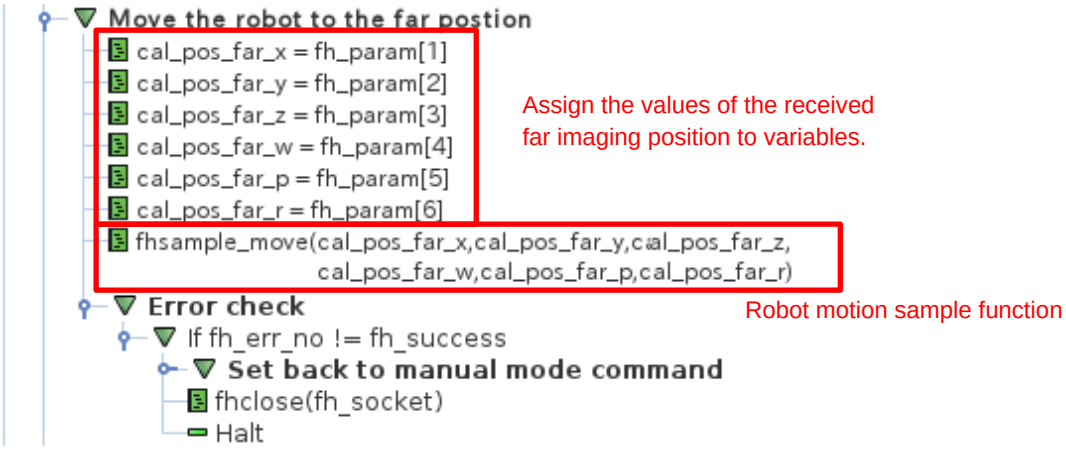
2 To receive the far imaging position from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).



Moving the Robot to the Far Imaging Position

The following describes a program that moves the robot to the far imaging position.

1 Store the values of the received far imaging position in variables, then execute the robot motion function (fhsample_move) to move the robot to the acquired far imaging position.



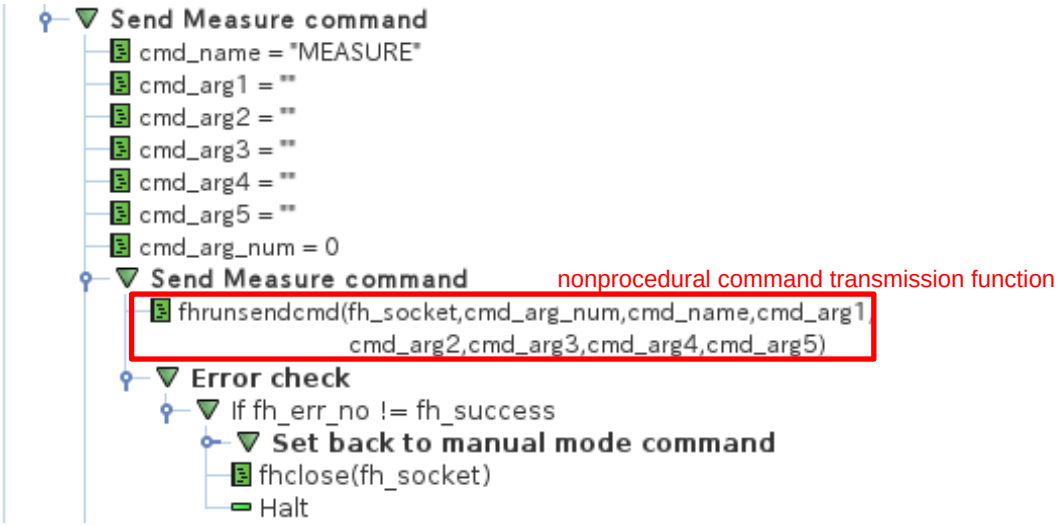
⚠WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

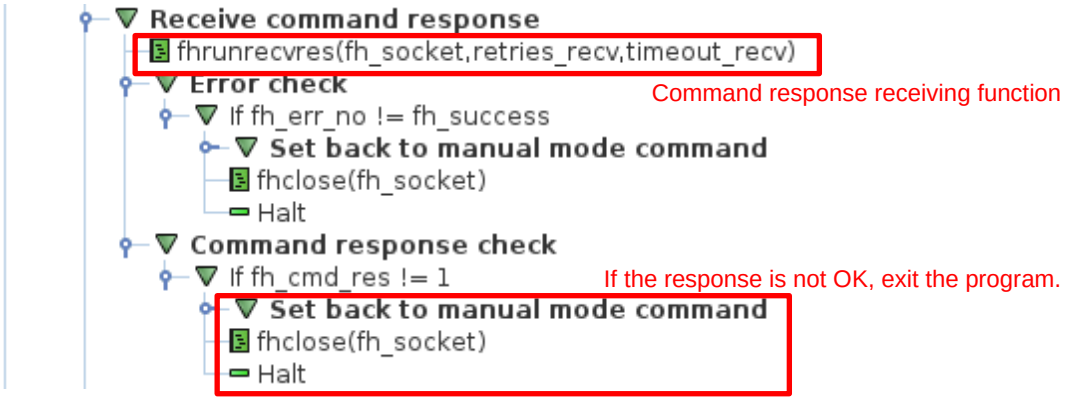
Executing a Measurement (Far Imaging Position)

The following describes a program that executes a measurement at the far imaging position.

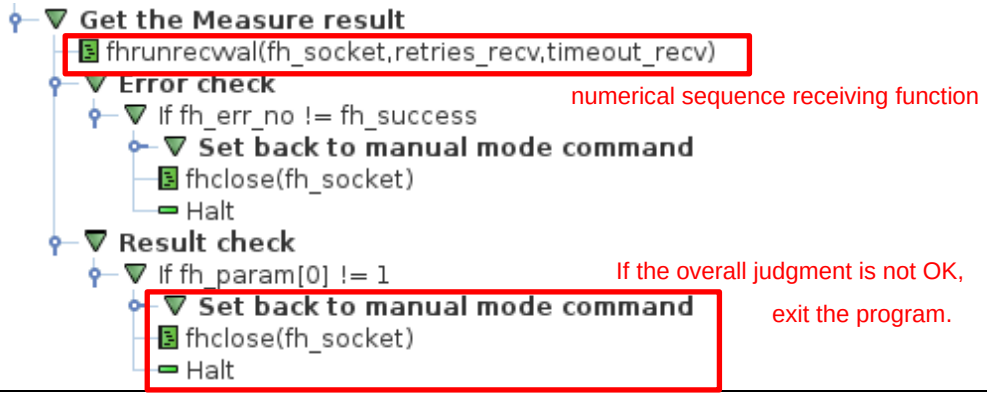
1 To send the measurement command to the Vision Sensor, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (fhrunsendcmd).



2 To receive the response to the measurement command from the Vision Sensor, execute the command response receiving function (fhrunrecvres).
If the response is not OK, execute the processing to end the program.



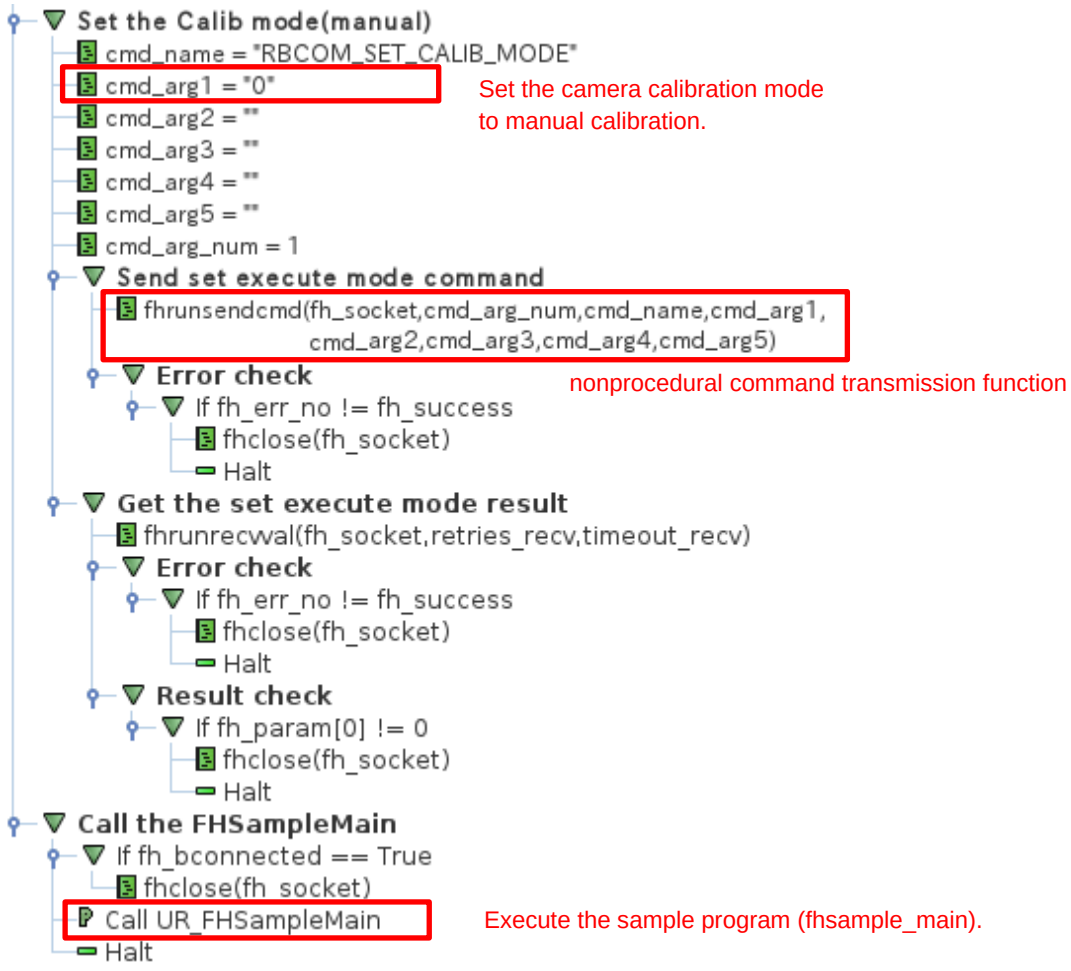
3 To receive the measurement result from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval) and check the value of the received overall judgment.
This program is created on the assumption that the measurement result from the Vision Sensor is sent as “TJG”



Ending the Program (Normal End)

The following describes processing that is carried out when AOS camera calibration is normally ended.

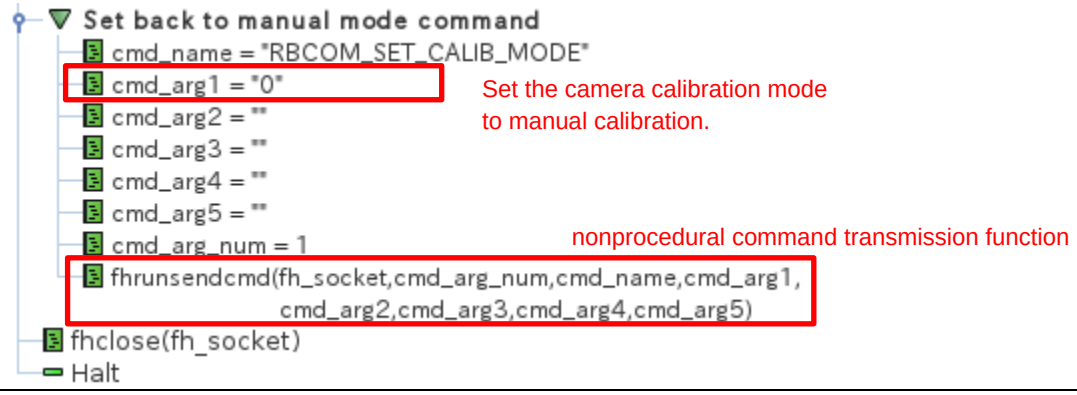
1 If this processing is successful, the program sends the camera calibration mode change command to manual calibration by the nonprocedural command transmission function (fhrunsendcmd), and then executes the sample program (fhsample_main).



Ending the Program (Error End)

The following describes processing that is carried out if an error occurs during execution of AOS camera calibration.

1 If an error occurs during execution, the program sends the camera calibration mode change command by the nonprocedural command transmission function (fhrunsendcmd) to set the manual calibration mode, and then stops the program.



3.7.8. For Robots Manufactured by NACHI-FUJIKOSHI CORP.

Connecting Vision Sensor to Robot Controller

The following describes a program that connects the Vision Sensor to the robot controller.

1 Declare internal variables.

* Description omitted (Refer to the source code.)

2 Execute the global variable initialization function (fhdefglobal).

```
'Set Parameter value
CallProc fhdefglobal()
```

Initialization function for external variables

3 Set the IP address and port number of the Vision Sensor as variables (if the default values need to be changed).

```
'Set network parameter
fh_ip_address = 100
fh_port_no = 9876
fh_retries_connect = 2
fh_timeout_connect = 4
```

IP address

Port number

4 Start the user task for Connect to FH server.

```
'Set the user task do nothing
fh_usertaskfunc_no = 0
'Start User Task
FORKMCR 990, 10000
CallProc fhsample_calaos()
```

Start the user task program

5 Set the variables as coordinate number. No.0 as local coordinate (Robot coordinate system), No.32 as tool coordinate (Flange coordinate system).
Execute the function(fhreflectcoord) to change the coordinate.

```
'Set the coordinate
set_local_coord_no = 0
set_tool_coord_no = 32
 'Change the coordinate
CallProc fhreflectcoord(set_local_coord_no,set_tool_coord_no)
'Error check
IF fh_err_number <> fh_success
 GoTo *CLOSE_AOS
ENDIF
```

6 Set the variables as arguments for the connection function (fhconnect) to the Vision Sensor (FH server) and execute it.

```
WHILE fh_bconnected = 0
 'Connect
 CallProc fhconnect()
 IF fh_err_number <> fh_success
  GoTo *CLOSE_AOS
 ENDIF
ENDW
```

Connection function with the Vision Sensor (FH server)

Switching Scenes on the Vision Sensor

The following describes a program that switches the scene to a Camera Calibration AOS scene.

1 Set the scene number (127) of the Camera Calibration AOS as a variable.
With the variable set as an argument, execute the scene switching function (fhsample_chgscn).

```
'(1)Change the scene of the FH
'   You have to select a scene No. for your application.

fh_sceneno_input = 127
CallProc fhsample_chgscn(fh_sceneno_input)
'Error check
IF fh_err_number <> fh_success
 GoTo *CLOSE_AOS
ENDIF
```

Scene number

Scene switching command execution sample function

Acquiring the Camera Status from the 3D Vision Sensor

The following describes a program that acquires the camera status of the 3D vision sensor and checks if the warmup of the camera is completed.

1 To send the camera status acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CAMERA_STATUS* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
'(2)Get the camera status
'   Check the camera status,When warming wait 1 second until status is OK
fh_res_string = "RBCOM_GET_CAMERA_STATUS"
fh_cmd_input_arg[1] = ""
fh_cmd_input_arg[2] = ""
fh_cmd_input_arg[3] = ""
fh_cmd_input_arg[4] = ""
fh_cmd_input_arg[5] = ""
fh_para_cnt= 0
WHILE L2% = 0
CallProc fhrunsendcmd(fh_para_cnt, fh_res_string, fh_cmd_input_arg[1-5])
 'Error check
 IF fh_err_number <> fh_success
  GoTo *CLOSE_AOS
 ENDIF
```

nonprocedural command transmission function

2 To receive the response to the camera status acquisition command from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

```
 'Recv Command Response
CallProc fhrunrecvval()
 'Error check
 IF fh_err_number <> fh_success
  GoTo *CLOSE_AOS
 ENDIF
 IF fh_param[1] <> 0
  GoTo *CLOSE_AOS
 ENDIF
```

numerical sequence receiving function

3 Keep sending the camera status acquisition command to the Vision Sensor until the camera status changes to warmup complete.

```
IF fh_param[2] = -1
 L2% = 0
 DELAY 1
```

If the camera status is warmup incomplete, send the camera status acquisition command again.

```
ELSEIF fh_param[2] = 1
 L2% = 1
ELSE
  GoTo *CLOSE_AOS
 ENDIF
```

If the camera status is warmup complete, proceed to the next processing.

Acquiring the Calibration Date/Time Comparison Result the from 3D Vision Sensor

The following describes a program that checks if the date of calibration of the 3D vision sensor matches the internal date of the Vision Sensor and determines whether the AOS camera calibration must be executed.

1 To send the date/time comparison result acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIBTIME_COMP* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
'(3)Get the Calib time to compare
 '   Compare the Calib time to the current time
 fh_res_string = "RBCOM_GET_CALIBTIME_COMP"
 fh_cmd_input_arg[1] = ""
 fh_cmd_input_arg[2] = ""
 fh_cmd_input_arg[3] = ""
 fh_cmd_input_arg[4] = ""
 fh_cmd_input_arg[5] = ""
 fh_para_cnt = 0
 CallProc fhrunsendcmd(fh_para_cnt, fh_res_string, fh_cmd_input_arg[1-5])
 'Error check
 IF fh_err_number <> fh_success
  GoTo *CLOSE_AOS
 ENDIF
```

nonprocedural command transmission function

2 To receive the response to the calibration date/time comparison result acquisition command from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).
If the date of calibration of the camera matches the internal date of the FH series, execute the sample program (fhsample_main) without executing camera calibration.

```
'Recv Command Response
 CallProc fhrunrecvval()
 'Error check
 IF fh_err_number <> fh_success
  GoTo *CLOSE_AOS
 ENDIF
 IF fh_param[1] <> 0
  GoTo *CLOSE_AOS
 ENDIF
 IF fh_param[2] = 1
  GoTo *CALL_FHSAMPLE
 ENDIF
```

numerical sequence receiving function

If calibration is not required,
execute the sample program (fhsample_main).

Changing the Calibration Mode of AOS Camera Calibration

The following describes a program that changes the calibration mode of AOS camera calibration to automatic calibration so that AOS camera calibration can be executed during measurement.

1 To change the calibration mode of the AOS camera calibration to automatic calibration, set the first nonprocedural command argument fh_cmd_input_arg[1]to *1*.
To send the camera calibration mode change command to the Vision Sensor, set the nonprocedural command name to *RBCOM_SET_CALIB_MODE* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
'(4)Send the execute mode command
'   Set the Calib mode to the auto mode
fh_res_string = "RBCOM_SET_CALIB_MODE"
fh_cmd_input_arg[1] = "1"
fh_cmd_input_arg[2] = ""
fh_cmd_input_arg[3] = ""
fh_cmd_input_arg[4] = ""
fh_cmd_input_arg[5] = ""
fh_para_cnt = 1
CallProc fhrunsendcmd(fh_para_cnt, fh_res_string, fh_cmd_input_arg[1-5])
'Error check
IF fh_err_number <> fh_success
 GoTo *CLOSE_AOS
ENDIF
```

Set the camera calibration mode to automatic calibration.

nonprocedural command transmission function

2 To receive the response to the camera calibration mode change command from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

```
'Recv Command Response
CallProc fhrunrecvval()
'Error check
IF fh_err_number <> fh_success
 GoTo *CLOSE_AOS
ENDIF
IF fh_param[1] <> 0
 GoTo *CLOSE_AOS
ENDIF
```

numerical sequence receiving function

Acquiring the Near Imaging Position

The following describes a program that acquires the near imaging position registered in *Automatic Calibration (Vision Master)*.

1 To acquire the near imaging position, set the first nonprocedural command argument fh_cmd_input_arg[1] to *0*. To send the camera calibration imaging position acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
'(5)Calibration position near
 '   Get the calib position near,Meausure at calib position near
 'Get calib position near

 fh_res_string = "RBCOM_GET_CALIB_POS"
 fh_cmd_input_arg[1] = "0"
 fh_cmd_input_arg[2] = ""
 fh_cmd_input_arg[3] = ""
 fh_cmd_input_arg[4] = ""
 fh_cmd_input_arg[5] = ""
 fh_para_cnt = 1
 CallProc fhrunsendcmd(fh_para_cnt, fh_res_string, fh_cmd_input_arg[1-5])
 'Error check
 IF fh_err_number <> fh_success
  GoTo *SET_MANUAL_MODE
 ENDIF
```

Set this argument to *0* to acquire the near imaging position.

nonprocedural command transmission function

2 To receive the near imaging position from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

```
 'Recv Command Response
 CallProc fhrunrecvval()
 'Error check
 IF fh_err_number <> fh_success
  GoTo *SET_MANUAL_MODE
 ENDIF
 IF fh_param[1] <> 0
  GoTo *SET_MANUAL_MODE
 ENDIF
```

numerical sequence receiving function

Moving the Robot to the Near Imaging Position

The following describes a program that moves the robot to the near imaging position.

1 Store the values of the received near imaging position in variables. then execute the robot motion function (fhsample_move) to move the robot to the acquired near imaging position.

```
'Move to calib position near
 fh_move_position[1] = fh_param[2]
 fh_move_position[2] = fh_param[3]
 fh_move_position[3] = fh_param[4]
 fh_move_position[4] = fh_param[5]
 fh_move_position[5] = fh_param[6]
 fh_move_position[6] = fh_param[7]
```

Assign the values of the received near imaging position to variables.

```
 CallProc
fhsample_move(fh_move_position[1],fh_move_position[2],fh_move_position[3],fh_move_p
osition[4],fh_move_position[5],fh_move_position[6])
 IF fh_err_number <> fh_success
  GoTo *SET_MANUAL_MODE
 ENDIF
```

Robot motion sample function

⚠WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

Executing a Measurement (Near Imaging Position)

The following describes a program that executes a measurement at the near imaging position.

1 To send the measurement command to the Vision Sensor, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
'Send Measure Command
 fh_res_string = "MEASURE"
 fh_cmd_input_arg[1] = ""
 fh_cmd_input_arg[2] = ""
 fh_cmd_input_arg[3] = ""
 fh_cmd_input_arg[4] = ""
 fh_cmd_input_arg[5] = ""
 fh_para_cnt = 0
 CallProc fhrunsendcmd(fh_para_cnt, fh_res_string, fh_cmd_input_arg[1-5])
 'Error check
 IF fh_err_number <> fh_success
  GoTo *SET_MANUAL_MODE
 ENDIF
```

nonprocedural command transmission function

2 To receive the response to the measurement command from the Vision Sensor, execute the command response receiving function (fhrunrecvres).
If the response is not OK, execute the processing to end the program.

```
 'Recv Command Response
 CallProc L4% = fhrunrecvres()
 'Error check
 IF fh_err_number <> fh_success
  GoTo *SET_MANUAL_MODE
 ENDIF

 'Command Response Check
 IF L4% <> 1
  GoTo *SET_MANUAL_MODE
 ENDIF
```

command response receiving function

If the response is not OK, exit the program.

3 To receive the measurement result from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval) and check the value of the received overall judgment.
This program is created on the assumption that the measurement result from the Vision Sensor is sent as “TJG”

```
'Get the Measurement Result
CallProc fhrunrecvval()
```

numerical sequence receiving function

```
'Error check
IF fh_err_number <> fh_success
 GoTo *SET_MANUAL_MODE
ENDIF

'Total Judge Check
IF fh_param[1] <> 1
 GoTo *SET_MANUAL_MODE
ENDIF
```

If the overall judgment is not OK, exit the program

Acquiring the Far Imaging Position

The following describes a program that acquires the far imaging position registered in *Automatic Calibration (Vision Master)*.

1 To acquire the far imaging position, set the first nonprocedural command argument fh_cmd_input_arg[1] to *1*. To send the camera calibration imaging position acquisition command to the Vision Sensor, set the nonprocedural command name to *RBCOM_GET_CALIB_POS* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
'(6)Calibration position far
'   Get the calib position Far,Meausure at calib position far
'Get calib position far
fh_res_string = "RBCOM_GET_CALIB_POS"
fh_cmd_input_arg[1] = "1"
fh_cmd_input_arg[2] = ""
fh_cmd_input_arg[3] = ""
fh_cmd_input_arg[4] = ""
fh_cmd_input_arg[5] = ""
fh_para_cnt = 1
CallProc fhrunsendcmd(fh_para_cnt, fh_res_string, fh_cmd_input_arg[1-5])
'Error check
IF fh_err_number <> fh_success
 GoTo *SET_MANUAL_MODE
ENDIF
```

Set this argument to *1* to acquire the far imaging position.

nonprocedural command transmission function

2 To receive the far imaging position from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval).

```
'Recv Command Response
CallProc fhrunrecvval()

'Error check
IF fh_err_number <> fh_success
 GoTo *SET_MANUAL_MODE
ENDIF
IF fh_param[1] <> 0
 GoTo *SET_MANUAL_MODE
ENDIF
```

nonprocedural command transmission function

Moving the Robot to the Far Imaging Position

The following describes a program that moves the robot to the far imaging position.

1 Store the values of the received far imaging position in variables. Then execute the robot motion function (fhsample_move) to move the robot to the acquired far imaging position.

```
'Move to calib position far
fh_move_position[1] = fh_param[2]
fh_move_position[2] = fh_param[3]
fh_move_position[3] = fh_param[4]
fh_move_position[4] = fh_param[5]
fh_move_position[5] = fh_param[6]
fh_move_position[6] = fh_param[7]
```

Assign the values of the received far imaging position to variables.

```
CallProc
fhsample_move(fh_move_position[1],fh_move_position[2],fh_move_position[3],fh_move_p
osition[4],fh_move_position[5],fh_move_position[6])
IF fh_err_number <> fh_success
 GoTo *SET_MANUAL_MODE
ENDIF
```

Robot motion sample function

⚠WARNING

These operations drive the robot.
Operate the robot in the state whereby pressing the [Emergency stop] button can stop its motion anytime.

Executing a Measurement (Far Imaging Position)

The following describes a program that executes a measurement at the far imaging position.

1 To send the measurement command to the Vision Sensor, set the nonprocedural command name to *MEASURE* and execute the nonprocedural command transmission function (fhrunsendcmd).

```
'Send Measure Command
fh_res_string = "MEASURE"
fh_cmd_input_arg[1] = ""
fh_cmd_input_arg[2] = ""
fh_cmd_input_arg[3] = ""
fh_cmd_input_arg[4] = ""
fh_cmd_input_arg[5] = ""
fh_para_cnt = 0
CallProc fhrunsendcmd(fh_para_cnt, fh_res_string, fh_cmd_input_arg[1-5])
'Error check
IF fh_err_number <> fh_success
 GoTo *SET_MANUAL_MODE
ENDIF
```

nonprocedural command transmission function

2 To receive the response to the measurement command from the Vision Sensor, execute the command response receiving function (fhrunrecvres).
If the response is not OK, execute the processing to end the program.

```
'Recv Command Response
CallProc L4% = fhrunrecvres()
'Error check
IF fh_err_number <> fh_success
 GoTo *SET_MANUAL_MODE
ENDIF

'Command Response Check
IF L4% <> 1
 GoTo *SET_MANUAL_MODE
ENDIF
```

Command response receiving function

If the response is not OK, exit the program.

3 To receive the measurement result from the Vision Sensor, execute the numerical sequence receiving function (fhrunrecvval) and check the value of the received overall judgment.
This program is created on the assumption that the measurement result from the Vision Sensor is sent as “TJG”

```
 'Get the Measurement Result
 CallProc fhrunrecvval()
```

nonprocedural command transmission function

```
'Error check
 IF fh_err_number <> fh_success
  GoTo *SET_MANUAL_MODE
 ENDIF

 'Total Judge Check
 IF fh_param[1] <> 1
  GoTo *SET_MANUAL_MODE
 ENDIF
```

If the overall judgment is not OK, exit the program

Ending the Program (Normal End)

The following describes processing that is carried out when AOS camera calibration is normally ended.

1 If this processing is successful, the program sends the camera calibration mode change command to manual calibration by the nonprocedural command transmission function (fhrunsendcmd), and then executes the sample program (fhsample_main).

```
'(7)Set the calibration manual mode
'Set Calib manual mode
fh_res_string = "RBCOM_SET_CALIB_MODE"
fh_cmd_input_arg[1] = "0"
fh_cmd_input_arg[2] = ""
fh_cmd_input_arg[3] = ""
fh_cmd_input_arg[4] = ""
fh_cmd_input_arg[5] = ""
fh_para_cnt = 1
CallProc fhrunsendcmd(fh_para_cnt, fh_res_string, fh_cmd_input_arg[1-5])
'Error check
IF fh_err_number <> fh_success
 GoTo *CLOSE_AOS
ENDIF

'Recv Command Response
CallProc fhrunrecvval()
'Error check
IF fh_err_number <> fh_success
 GoTo *CLOSE_AOS
ENDIF
IF fh_param[1] <> 0
 GoTo *CLOSE_AOS
ENDIF

*CALL_FHSAMPLE
'Call program of FH sample
CallProc fhsample_main()
GoTo *EXIT_AOS
```

Set the camera calibration mode to manual calibration.

nonprocedural command transmission function

Execute the sample program (fhsample_main).

Ending the Program (Error End)

The following describes processing that is carried out if an error occurs during execution of AOS camera calibration.

1 If an error occurs during execution, the program sends the camera calibration mode change command by the nonprocedural command transmission function (fhrunsendcmd) to set the manual calibration mode, and then stops the program.

```
 '3. Finalization sequence
'   Disconnect to the FH server
'   Send Calib manual mode (manual mode)

*SET_MANUAL_MODE
 'Send Calib manual mode Command
 fh_res_string = "RBCOM_SET_CALIB_MODE"
 fh_cmd_input_arg[1] = "0"
 fh_cmd_input_arg[2] = ""
 fh_cmd_input_arg[3] = ""
 fh_cmd_input_arg[4] = ""
 fh_cmd_input_arg[5] = ""
 fh_para_cnt = 1
 CallProc fhrunsendcmd(fh_para_cnt, fh_res_string, fh_cmd_input_arg[1-5])
 CallProc fhrunrecvval()

*CLOSE_AOS
 CallProc fhclose()

*EXIT_AOS
EndProc
```

Set the camera calibration mode to manual calibration.

nonprocedural command transmission function

OMRON Corporation **Industrial Automation Company**

Kyoto, JAPAN **Contact : www.ia.omron.com**

Regional Headquarters

OMRON EUROPE B.V.
Wegalaan 67-69, 2132 JD Hoofddorp
The Netherlands
Tel: (31) 2356-81-300 Fax: (31) 2356-81-388

OMRON ELECTRONICS LLC
2895 Greenspoint Parkway, Suite 200
Hoffman Estates, IL 60169 U.S.A.
Tel: (1) 847-843-7900 Fax: (1) 847-843-7787

OMRON ASIA PACIFIC PTE. LTD.
438B Alexandra Road, #08-01/02 Alexandra
Technopark, Singapore 119968
Tel: (65) 6835-3011 Fax: (65) 6835-3011

OMRON (CHINA) CO., LTD.
Room 2211, Bank of China Tower,
200 Yin Cheng Zhong Road,
PuDong New Area, Shanghai, 200120, China
Tel: (86) 21-6023-0333 Fax: (86) 21-5037-2388

Authorized Distributor:

Cat. No. Z451-E1-05 0624