

COGNEX

AS200 Alignment Sensor Communication Document Supplement

2018 September

Revision: Supplement to User Guide (HTLM5) VER 1.0.0

Legal Notices

The software described in this document is furnished under license, and may be used or copied only in accordance with the terms of such license and with the inclusion of the copyright notice shown on this page. Neither the software, this document, nor any copies thereof may be provided to, or otherwise made available to, anyone other than the licensee. Title to, and ownership of, this software remains with Cognex Corporation or its licensor. Cognex Corporation assumes no responsibility for the use or reliability of its software on equipment that is not supplied by Cognex Corporation. Cognex Corporation makes no warranties, either express or implied, regarding the described software, its merchantability, non-infringement or its fitness for any particular purpose.

The information in this document is subject to change without notice and should not be construed as a commitment by Cognex Corporation. Cognex Corporation is not responsible for any errors that may be present in either this document or the associated software.

Companies, names, and data used in examples herein are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, nor transferred to any other media or language without the written permission of Cognex Corporation.

2018 June 28 11:42 AM

Revision: 1.0.0.640

Copyright © 2018. Cognex Corporation. All Rights Reserved.

Portions of the hardware and software provided by Cognex may be covered by one or more U.S. and foreign patents, as well as pending U.S. and foreign patents listed on the Cognex web site at: <http://www.cognex.com/patents>.

The following are registered trademarks of Cognex Corporation:

Cognex, 2DMAX, Advantage, AlignPlus, Assemblyplus, Check it with Checker, Checker, Cognex Vision for Industry, Cognex VSOC, CVL, DataMan, DisplayInspect, DVT, EasyBuilder, Hotbars, IDMax, In-Sight, Laser Killer, MVS-8000, OmniView, PatFind, PatFlex, PatInspect, PatMax, PatQuick, SensorView, SmartView, SmartAdvisor, SmartLearn, UltraLight, Vision Solutions, VisionPro, VisionView

The following are trademarks of Cognex Corporation:

The Cognex logo, 1DMax, 3D-Locate, 3DMax, BGAll, CheckPoint, Cognex VSoC, CVC-1000, FFD, iLearn, In-Sight (design insignia with cross-hairs), In-Sight 2000, InspectEdge, Inspection Designer, MVS, NotchMax, OCRMax, PatMax RedLine, ProofRead, SmartSync, ProfilePlus, SmartDisplay, SmartSystem, SMD4, VisiFlex, Xpand

Portions copyright © Microsoft Corporation. All rights reserved.

Portions copyright © MadCap Software, Inc. All rights reserved.

Other product and company trademarks identified herein are the trademarks of their respective owners.

Copyright © 2018 | Cognex Corporation | All Rights Reserved.

Alignment Sensor User Guide | Revision: 1.0.0.640 | 2018 June 28

Table of Contents

1. Overview.....	5
1.1. TCP/IP Command Structure.....	5
1.2. SLMP Communication	6
1.2.1. Control Block Layout	6
1.2.2. Status Block Layout	7
1.2.3. Input Block Layout	7
1.2.4. Output Block Layout	8
2. AS200 Commands.....	9
2.1. Overview.....	9
Status & Version	9
2.2. Get State.....	9
GS.....	9
2.3. Get Version	10
GV	10
Hand-Eye Calibration.....	11
2.4. Hand Eye Begin.....	11
HEB	11
2.5. Hand Eye Execute Step	12
HE	12
2.6. Hand Eye End.....	15
HEE.....	15
AutoCalibration	16
2.7. Auto-Calibration Begin	16
ACB	16
2.8. Auto-Calibration Execute Step	18
AC	18
Custom Calibration.....	20
2.9. Custom Calibration Begin	20
CCB	20
2.10. Custom Calibration Execute Step	22
CC.....	22
2.11. Custom Calibration End	24
CCE.....	24

Golden Pose Alignment	25
2.12. Train Alignment Golden Pose	25
TA.....	25
2.13. Execute Alignment.....	26
XA.....	26
VGR commands (guided pick or place).....	28
2.14. Train Target Pose.....	28
TT	28
2.15. Train Target Robot Pose	30
TTR.....	30
2.16. Execute Target Alignment	32
XT	32
3. Loading a Recipe.....	34

1. Overview

This document describes the AS200 communication interface for both TCP/IP and SLMP connections.

1.1. TCP/IP Command Structure

With TCP/IP communication, a command is an ASCII string with the command and arguments, separated by comma:

`<Command>[,<Argument0>,<Argument1>,...,<Argument n>]`

Examples:

`HEB,0`

`XA,0,Off,100.0,140.123,0,15.36,0,0`

The response returned by the vision system is also a string containing the command this is the response for, a status value and result values, all separated by comma:

`<Command>,<Status>[,<Result 1>,<Result 2>,...,<Result n>]`

Examples:

`HEB,1`

`XA,1,0.035,0.056,0.0,0.002,0.0.0.0`

By default, command string and result string are terminated by CR/LF.

1.2. SLMP Communication

SLMP communication uses four memory blocks in the PLC's D-Register set to exchange data with the vision system. For each of the memory blocks an offset and a size must be specified. The four memory blocks are:

Name	Offset	Size (Words)	Purpose
Control Block	VCB_Offset	2	Contains Control Bits to control the Vision system
Status Block	VSB_Offset	2	Contains Status Bits providing status information of the Vision system
Input Block	VIB_Offset	25	Memory area to write user data to the Vision system
Output Block	VOB_Offset	29	Memory area to receive result data from the Vision system.

The size of the Control Block and Status Block is fixed to 2.

The Offset values must be specified on the SLMP configuration page while creating a sensor.

Example:

VCB_Offset = D300, Size = 2
VSB_Offset = D302, Size = 2
VIB_Offset = D304, Size = 25
VOB_Offset = D350, Size = 29

The screenshot shows the 'SLMP Scanner' configuration window. At the top, 'TCP/IP' is selected. The 'PLC Controller' is set to 'iQ-R/Q/L Series (3E Frame)'. The 'PLC IP Address' is '192.168.21.102', 'Host Port' is '7890', and 'Poll Interval' is '100'. Below these, there are four rows for block configuration: 'Control Data Block' (Offset: 300, Size: 2), 'Status Block' (Offset: 302, Size: 2), 'Input Block' (Offset: 304, Size: 25), and 'Output Block' (Offset: 350, Size: 29). The 'Input Block' and 'Output Block' size fields are highlighted with blue borders.

	Offset	Size
Control Data Block	300	2
Status Block	302	2
Input Block	304	25
Output Block	350	29

1.2.1. Control Block Layout

The Control Block has a size of 2 Words, i.e. 4 Bytes or 32 Bit.

It is used to initiate actions to be performed by the Vision system like

- Executing a Command
- Loading a Recipe

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Set Offline	Reserved		Execute Command	Reserved			
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Reserved							
Bit 23	Bit 22	Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit 16
Reserved							Set User Data
Bit 31	Bit 30	Bit 29	Bit 28	Bit 27	Bit 26	Bit 25	Bit 24
Reserved							Soft Event 0

1.2.2. Status Block Layout

Like the Control Block, the Status Block has a size of 2 Words.

It is used by the Vision System to provide Status information to the PLC and to acknowledge Control Block Bit status changes.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Online	Reserved			Reserved			
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Reserved	Command Failed	Command Completed	Command Executing	Reserved			
Bit 23	Bit 22	Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit 16
Reserved							Set User Data Ack
Bit 31	Bit 30	Bit 29	Bit 28	Bit 27	Bit 26	Bit 25	Bit 24
Reserved							Soft Event Ack 0

1.2.3. Input Block Layout

The Input Block is used by the PLC to write commands and arguments to the Vision System. It has a two-Word header (4 Bytes) preceding the actual command data:

Word Offset	Length (Word)	Name	Description
VIB_Offset + 0	1	Command	The ID (1-999) number of the Recipe to load when the “Execute Command” bit is set in the Control Block.
VIB_Offset + 1	1	Reserved	Unused
VIB_Offset + 2	1 (int)	Command	The ID of the command to be executed, if the “Set User Data” Bit was set in the Control Block.
VIB_Offset + 3	2 (float)	Argument0	The 1 st command argument.

VIB_Offset + 5	2 (float)	Argument1	The 2 nd command argument.
VIB_Offset + 7	2 (float)	Argument2	The 3 rd command argument.
....	...		
VIB_Offset + 23	2 (float)	Argument10	The 11 th command argument

1.2.4. Output Block Layout

The Output Block is used by the Vision System to send result data to the PLC. It has a header of 5 Words length preceding the actual data:

Word Offset	Length (Word)	Name	Description
VOB_Offset + 0	1	RecipeID	The ID of the currently running Recipe
VOB_Offset + 1	1		Reserved
VOB_Offset + 2	1		Reserved
VOB_Offset + 3	1		Reserved
VOB_Offset + 4	1		Reserved
VOB_Offset + 5	1 (int)	Command	The ID of the command this data is the result from
VOB_Offset + 6	1 (int)	Status	The result status (Success: 1, Fail: <=0)
VOB_Offset + 7	2 (float)	Result 0	Result data (e.g. X coordinate)
VOB_Offset + 9	2 (float)	Result 1	Result data (e.g. Y coordinate)
...	...	...	...
VOB_Offset +27	2 (float)	Result 10	Result data

2. AS200 Commands

2.1. Overview

This section describes the commands supported by the AS200 Alignment Sensor.

Status & Version

2.2. Get State

GS

Returns the current State.

TCP/IP:

Command String:

GS,<cam>

Field	Arguments
GS	The command
<cam>	The camera(s) to be read 0: All available cameras 1: Camera 1 only 2: Camera 2 only

Result String:

GS,<Status>

Field	Arguments
GS	The command
<Status>	The result status 1: Sensor ready <=0: Sensor not ready/Fail

SLMP

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	1010	Command: "GS" command ID
VIB_Offset + 3	<cam>	Argument0: The camera(s) to be read

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	1010	Command: "GS" command ID
VOB_Offset + 6	<Status>	The result Status: 1: Sensor ready <=0: Sensor not ready/Fail

2.3. Get Version

GV

Returns the current version of the loaded recipe.

TCP/IP:

Command String:

GV,<cam>

Field	Arguments
GV	The command
<cam>	The camera(s) to be read 0: All available cameras 1: Camera 1 only 2: Camera 2 only

Result String:

GV,<Status>,<Major>,<Minor>,<SubMinor>

Field	Arguments
GV	The command
<Status>	The result status 1: Success <=0: Fail
<Major>	The Major Version
<Minor>	The Minor Version
<SubMinor>	The Sub Minor Version

SLMP

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	1011	Command: "GV" command ID
VIB_Offset + 3	<cam>	Argument0: The camera(s) to be read

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	1011	Command: "GV" command ID
VOB_Offset + 6	<Status>	The result Status: 1: Success <=0: Fail
VOB_Offset + 7	<Major>	The Major Version
VOB_Offset + 9	<Minor>	The Minor Version
VOB_Offset + 11	<SubMinor>	The Sub Minor Version

Hand-Eye Calibration

2.4. Hand Eye Begin

HEB

Initiate a Hand-Eye calibration.

TCP/IP:

Command String:

HEB,<cam>

Field	Arguments
HEB	The command
<cam>	The camera(s) to calibrate 0: All available cameras 1: Camera 1 only 2: Camera 2 only

Result String:

HEB,<Status>

Field	Arguments
HEB	The command
<Status>	The result status 1: Success <=0: Fail

SLMP

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	2020	Command: "HEB" command ID
VIB_Offset + 3	<cam>	Argument0: The camera(s) to be calibrated

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	2020	Command: "HEB" command ID
VOB_Offset + 6	<Status>	The result Status: 1: Success <=0: Fail

2.5. Hand Eye Execute Step

HE

Execute a Hand-Eye calibration step.

TCP/IP:

Command String:

HE,<cam>,<targetID>,<X>,<Y>,<Z>,<A>,,<C>

Field	Description
HE	The command
<cam>	The camera(s) to be calibrated: 0: All available cameras 1: Camera 1 only 2: Camera 2 only
<targetID>	Should be 0. Ignored by AS200
<X>	The X-coordinate of the current position of the motion system
<Y>	The Y-coordinate of the current position of the motion system
<Z>	The Z-coordinate of the current position of the motion system
<A>	The Z-rotation of the current position of the motion system
	The Y-rotation of the current position of the motion system
<C>	The X-rotation of the current position of the motion system

Result String:

HE,<Status>

Filed	Arguments
HE	The command this is the result from
<Status>	The result status 1: Success <=0: Fail

SLMP

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	2021	Command: "HE" command ID
VIB_Offset + 3	<cam>	Argument0: The camera(s) to be calibrated
VIB_Offset + 5	<targetID>	Argument1: Should be 0 (ignored by AS200)
VIB_Offset + 7	<X>	Argument2: The X-coordinate of the current motion position
VIB_Offset + 9	<Y>	Argument3: The Y-coordinate of the current motion position
VIB_Offset + 11	<Z>	Argument4: The Z-coordinate of the current motion position
VIB_Offset + 13	<A>	Argument5: The Z-rotation of the current motion position
VIB_Offset + 15		Argument6: The Y- rotation of the current motion position
VIB_Offset + 17	<C>	Argument7: The X- rotation of the current motion position

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	2021	Command: "HE" command ID
VOB_Offset + 6	<Status>	The result Status: 1: Success <=0: Fail

2.6. Hand Eye End

HEE

Computes the Hand-Eye calibration from the data collected in the previous “HE” steps.

TCP/IP:

Command String:

HEE,<cam>

Field	Arguments
HEE	The command
<cam>	The camera(s) to compute the calibration for 0: All available cameras 1: Camera 1 only 2: Camera 2 only

Result String:

HEE,<Status>

Field	Arguments
HEE	The command
<Status>	The result status 1: Success <=0: Fail

SLMP

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	2022	Command: “HEE” command ID
VIB_Offset + 3	<cam>	Argument0: The camera(s) to be calibrated

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	2022	Command: “HEE” command ID
VOB_Offset + 6	<Status>	The result Status: 1: Success <=0: Fail

AutoCalibration

2.7. Auto-Calibration Begin

ACB

Initiate an AutoCalibration.

TCP/IP:

Command String:

ACB,<cam>,<targetID>,<X>,<Y>,<Z>,<A>,,<C>

Field	Arguments
ACB	The command
<cam>	The camera to calibrate 1: Camera 1 2: Camera 2
<targetID>	Any numeric value. Ignored by AS200
<X>	The X-coordinate of the current position of the motion system
<Y>	The Y-coordinate of the current position of the motion system
<Z>	The Z-coordinate of the current position of the motion system
<A>	The Z-rotation of the current position of the motion system
	The Y-rotation of the current position of the motion system
<C>	The X-rotation of the current position of the motion system

Result String:

ACB,<Status>,<X>,<Y>,<Z>,<A>,,<C>

Field	Description
ACB	The command
<Status>	The result status 2: Success, move to returned position and execute "AC" command <=0: Fail
<X>	The X coordinate of the next position to move to
<Y>	The Y coordinate of the next position to move to
<Z>	The Z coordinate of the next position to move to
<A>	The Z-rotation of the next position to move to
	The Y-rotation of the next position to move to
<C>	The X-rotation of the next position to move to

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	2030	Command: "ACB" command ID
VIB_Offset + 3	<cam>	Argument0: The camera to be calibrated
VIB_Offset + 5	<targetID>	Argument1: Should be 0 (ignored by AS200)
VIB_Offset + 7	<X>	Argument2: The X-coordinate of the current motion position
VIB_Offset + 9	<Y>	Argument3: The Y-coordinate of the current motion position
VIB_Offset + 11	<Z>	Argument4: The Z-coordinate of the current motion position
VIB_Offset + 13	<A>	Argument5: The Z-rotation of the current motion position
VIB_Offset + 15		Argument6: The Y- rotation of the current motion position
VIB_Offset + 17	<C>	Argument7: The X- rotation of the current motion position

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	2030	Command: "ACB" command ID
VOB_Offset + 6	<Status>	The result Status: 2: Success, move to returned position and execute "AC" command <=0: Fail
VOB_Offset + 7	<X>	The X coordinate of the next position to move to
VOB_Offset + 9	<Y>	The Y coordinate of the next position to move to
VOB_Offset + 11	<Z>	The Z coordinate of the next position to move to
VOB_Offset + 13	<A>	The Z-rotation of the next position to move to
VOB_Offset + 15		The Y-rotation of the next position to move to
VOB_Offset + 17	<C>	The X-rotation of the next position to move to

2.8. Auto-Calibration Execute Step

AC

Execute AutoCalibration step.

TCP/IP:

Command String:

AC,<cam>,<targetID>,<X>,<Y>,<Z>,<A>,,<C>

Field	Arguments
AC	The command
<cam>	The camera to calibrate 1: Camera 1 2: Camera 2
<targetID>	Should be 0. Ignored by AS200
<X>	The X-coordinate of the current position of the motion system
<Y>	The Y-coordinate of the current position of the motion system
<Z>	The Z-coordinate of the current position of the motion system
<A>	The Z-rotation of the current position of the motion system
	The Y-rotation of the current position of the motion system
<C>	The X-rotation of the current position of the motion system

Result String:

AC,<Status>,<X>,<Y>,<Z>,<A>,,<C>

Field	Description
AC	The command
<Status>	The result status 1: Success, calibration done. 2: Success, move to returned position and execute next "AC" command <=0: Fail
<X>	The X coordinate of the next position to move to
<Y>	The Y coordinate of the next position to move to
<Z>	The Z coordinate of the next position to move to
<A>	The Z-rotation of the next position to move to
	The Y-rotation of the next position to move to
<C>	The X-rotation of the next position to move to

SLMP

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	2031	Command: "AC" command ID
VIB_Offset + 3	<cam>	Argument0: The camera to be calibrated
VIB_Offset + 5	<targetID>	Argument1: Should be 0 (ignored by AS200)

VIB_Offset + 7	<X>	Argument2: The X-coordinate of the current motion position
VIB_Offset + 9	<Y>	Argument3: The Y-coordinate of the current motion position
VIB_Offset + 11	<Z>	Argument4: The Z-coordinate of the current motion position
VIB_Offset + 13	<A>	Argument5: The Z-rotation of the current motion position
VIB_Offset + 15		Argument6: The Y- rotation of the current motion position
VIB_Offset + 17	<C>	Argument7: The X- rotation of the current motion position

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	2031	Command: "AC" command ID
VOB_Offset + 6	<Status>	The result Status: 1: Success, calibration done 2: Success, move to returned position and execute "AC" command <=0: Fail
VOB_Offset + 7	<X>	The X coordinate of the next position to move to
VOB_Offset + 9	<Y>	The Y coordinate of the next position to move to
VOB_Offset + 11	<Z>	The Z coordinate of the next position to move to
VOB_Offset + 13	<A>	The Z-rotation of the next position to move to
VOB_Offset + 15		The Y-rotation of the next position to move to
VOB_Offset + 17	<C>	The X-rotation of the next position to move to

Custom Calibration

2.9. Custom Calibration Begin

CCB

Initiate Custom calibration.

TCP/IP:

Command String:

CCB,<cam>,<SwapHandedness>,<FeatureOffsetX>,<FeatureOffsetY>

Field	Arguments
CCB	The command
<cam>	The camera to calibrate 1: Camera 1 2: Camera 2
<SwapHandedness>	A value indicating whether the camera's handedness must be swapped 0: Don't swap 1: Swap
<FeatureOffsetX>	The X-Offset of the feature to be located from the rotation center of the motion system.
<FeatureOffsetY>	The Y-Offset of the feature to be located from the rotation center of the motion system.

Result String:

CCB,<Status>

Field	Description
CCB	The command
<Status>	The result status 1: Success <=0: Fail

SLMP

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	2040	Command: "CCB" command ID
VIB_Offset + 3	<cam>	Argument0: The camera to be calibrated
VIB_Offset + 5	<SwapHandedness>	Argument1
VIB_Offset + 7	<FeatureOffsetX>	Argument2
VIB_Offset + 9	<FeatureOffsetY>	Argument3

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	2040	Command: "CCB" command ID
VOB_Offset + 6	<Status>	The result Status: 1: Success <=0: Fail

2.10. Custom Calibration Execute Step

CC

Execute a CustomCalibration Step.

Note: CustomCalibration supports calibration with either 2 steps (motion only in X or Y) or 9 steps (motion in X and Y).

TCP/IP:

Command String:

CC,<cam>,<targetID>,<X>,<Y>,<Z>,<A>,,<C>

Field	Arguments
CC	The command
<cam>	The camera to calibrate 1: Camera 1 2: Camera 2
<targetID>	Should be 0. Ignored by AS200
<X>	The X-coordinate of the current position of the motion system
<Y>	The Y-coordinate of the current position of the motion system
<Z>	The Z-coordinate of the current position of the motion system
<A>	The Z-rotation of the current position of the motion system
	The Y-rotation of the current position of the motion system
<C>	The X-rotation of the current position of the motion system

Result String:

CC,<Status>

Field	Description
CC	The command
<Status>	The result status 1: Success, calibration done. 2: Success, move to returned position and execute next "AC" command <=0: Fail

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	2041	Command: "CC" command ID
VIB_Offset + 3	<cam>	Argument0: The camera to be calibrated
VIB_Offset + 5	<targetID>	Argument1: Should be 0 (ignored by AS200)
VIB_Offset + 7	<X>	Argument2: The X-coordinate of the current motion position
VIB_Offset + 9	<Y>	Argument3: The Y-coordinate of the current motion position
VIB_Offset + 11	<Z>	Argument4: The Z-coordinate of the current motion position
VIB_Offset + 13	<A>	Argument5: The Z-rotation of the current motion position
VIB_Offset + 15		Argument6: The Y- rotation of the current motion position
VIB_Offset + 17	<C>	Argument7: The X- rotation of the current motion position

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	2041	Command: "CC" command ID
VOB_Offset + 6	<Status>	The result Status: 1: Success <=0: Fail

2.11. Custom Calibration End

CCE

Computes the Hand-Eye calibration from the data collected in the (2 or 9) previous “CC” steps.

TCP/IP:

Command String:

CCE,<cam>

Field	Arguments
CCE	The command
<cam>	The camera to compute the calibration for 1: Camera 1 2: Camera 2

Result String:

CCE,<Status>

Field	Arguments
CCE	The command
<Status>	The result status 1: Success <=0: Fail

SLMP

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	2042	Command: “CCE” command ID
VIB_Offset + 3	<cam>	Argument0: The camera to be calibrated

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	2042	Command: “CCE” command ID
VOB_Offset + 6	<Status>	The result Status: 1: Success <=0: Fail

Golden Pose Alignment

2.12. Train Alignment Golden Pose

TA

Train the “Golden Pose” of the target.

TCP/IP:

Command String:

TA,<cam>

Field	Arguments
TA	The command
<cam>	The camera(s) used to locate the target features 0: All available cameras 1: Camera 1 only 2: Camera 2 only

Result String:

TA,<Status>

Field	Description
TA	The command
<Status>	The result status 1: Success <=0: Fail

SLMP

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	3010	Command: “TA” command ID
VIB_Offset + 3	<cam>	Argument0: The camera(s) used to locate target features.

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	3010	Command: “TA” command ID
VOB_Offset + 6	<Status>	The result Status: 1: Success <=0: Fail

2.13. Execute Alignment

XA

Execute “Golden Pose Alignment”

TCP/IP:

Command String:

XA,<cam>,<resultMode>,<X>,<Y>,<Z>,<A>,,<C>

Field	Arguments
XA	The command
<cam>	The camera(s) used to locate the target features 0: All available cameras 1: Camera 1 only 2: Camera 2 only
<resultMode>	Specifies how the result will be returned: “Abs”: Return the absolute motion position to get the target to the trained golden pose. “Off”: Return the Offset from the current motion position to get the target to the trained golden pose.
<X>	The X-coordinate of the current position of the motion system
<Y>	The Y-coordinate of the current position of the motion system
<Z>	The Z-coordinate of the current position of the motion system
<A>	The Z-rotation of the current position of the motion system
	The Y-rotation of the current position of the motion system
<C>	The X-rotation of the current position of the motion system

Result String:

XA,<Status>,<X>,<Y>,<Z>,<A>,,<C>

Field	Description
XA	The command
<Status>	The result status 1: Success <=0: Fail
<X>	The X-coordinate (absolute or offset) of the motion system’s destination pose.
<Y>	The Y-coordinate (absolute or offset) of the motion system’s destination pose.
<Z>	The Z-coordinate (absolute or offset) of the motion system’s destination pose.
<A>	The Z-rotation (absolute or offset) of the motion system’s destination pose.
	The Y-rotation (absolute or offset) of the motion system’s destination pose.
<C>	The X-rotation (absolute or offset) of the motion system’s destination pose.

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	3011	Command: "XA" command ID
VIB_Offset + 3	<cam>	Argument0: The camera(s) used to locate target features.
VIB_Offset + 5	<resultMode>	Argument1: The requested result mode: 1: Absolute 2: Offset
VIB_Offset + 7	<X>	The X-coordinate of the current position of the motion system
VIB_Offset + 9	<Y>	The Y-coordinate of the current position of the motion system
VIB_Offset + 11	<Z>	The Z-coordinate of the current position of the motion system
VIB_Offset + 13	<A>	The Z-rotation of the current position of the motion system
VIB_Offset + 15		The Y-rotation of the current position of the motion system
VIB_Offset + 17	<C>	The X-rotation of the current position of the motion system

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	3011	Command: "XA" command ID
VOB_Offset + 6	<Status>	The result Status: 1: Success <=0: Fail
VOB_Offset + 7	<X>	The X-coordinate of the result pose (absolute or offset)
VOB_Offset + 9	<Y>	The Y-coordinate of the result pose (absolute or offset)
VOB_Offset + 11	<Z>	The Z-coordinate of the result pose (absolute or offset)
VOB_Offset + 13	<A>	The Z-rotation of the result pose (absolute or offset)
VOB_Offset + 15		The Y-rotation of the result pose (absolute or offset)
VOB_Offset + 17	<C>	The X-rotation of the result pose (absolute or offset)

Robot commands (Vision Guided pick or place)

In order to do a guided pick or guided place, a target pose and a corresponding robot pick (or place) pose must be trained.

2.14. Train Target Pose

TT

Train the target pose.

TCP/IP:

Command String:

TT,<cam>,<X>,<Y>,<Z>,<A>,,<C>

Field	Arguments
TT	The command
<cam>	The camera(s) used to locate the target features 0: All available cameras 1: Camera 1 only 2: Camera 2 only
<X>	The X-coordinate of the current position of the motion system
<Y>	The Y-coordinate of the current position of the motion system
<Z>	The Z-coordinate of the current position of the motion system
<A>	The Z-rotation of the current position of the motion system
	The Y-rotation of the current position of the motion system
<C>	The X-rotation of the current position of the motion system

Result String:

TT,<Status>

Field	Description
TT	The command
<Status>	The result status 1: Success <=0: Fail

SLMP

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	4010	Command: "TT" command ID
VIB_Offset + 3	<cam>	Argument0: The camera(s) used to locate target features.
VIB_Offset + 5	<X>	The X-coordinate of the current position of the motion system

VIB_Offset + 7	<Y>	The Y-coordinate of the current position of the motion system
VIB_Offset + 9	<Z>	The Z-coordinate of the current position of the motion system
VIB_Offset + 11	<A>	The Z-rotation of the current position of the motion system
VIB_Offset + 13		The Y-rotation of the current position of the motion system
VIB_Offset + 15	<C>	The X-rotation of the current position of the motion system

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	4010	Command: "TT" command ID
VOB_Offset + 6	<Status>	The result Status: 1: Success <=0: Fail

2.15. Train Target Robot Pose

TTR

Train the robot pick (or place) pose.

TCP/IP:

Command String:

TTR,<cam>,<X>,<Y>,<Z>,<A>,,<C>

Field	Arguments
TTR	The command
<cam>	The camera(s) used to locate the target features 0: All available cameras 1: Camera 1 only 2: Camera 2 only
<X>	The X-coordinate of the robot's pick (or place) pose
<Y>	The Y-coordinate of the robot's pick (or place) pose
<Z>	The Z-coordinate of the robot's pick (or place) pose
<A>	The Z-rotation of the robot's pick (or place) pose
	The Y-rotation of the robot's pick (or place) pose
<C>	The X-rotation of the robot's pick (or place) pose

Result String:

TTR,<Status>

Field	Description
TTR	The command
<Status>	The result status 1: Success <=0: Fail

SLMP

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	4011	Command: "TTR" command ID
VIB_Offset + 3	<cam>	Argument0: The camera(s) used to locate target features.
VIB_Offset + 5	<X>	Argument1: The X-coordinate of the robot's pick (or place) pose
VIB_Offset + 7	<Y>	Argument2: The Y-coordinate of the robot's pick (or place) pose
VIB_Offset + 9	<Z>	Argument3: The Z-coordinate of the robot's pick (or place) pose
VIB_Offset + 11	<A>	Argument4: The Z-rotation of the robot's pick (or place) pose
VIB_Offset + 13		Argument5: The Y-rotation of the robot's pick (or place) pose
VIB_Offset + 15	<C>	Argument6: The X-rotation of the robot's pick (or place) pose

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	4011	Command: "TTR" command ID
VOB_Offset + 6	<Status>	The result Status: 1: Success <=0: Fail

2.16. Execute Target Alignment

XT

Locate the target and get the corrected robot pick (or place) pose.

TCP/IP:

Command String:

XT,<cam>,<resultMode>,<X>,<Y>,<Z>,<A>,,<C>

Field	Arguments
XT	The command
<cam>	The camera(s) used to locate the target features 0: All available cameras 1: Camera 1 only 2: Camera 2 only
<resultMode>	Specifies how the result will be returned: “Abs”: Return the absolute motion position to get the target to the trained golden pose. “Off”: Return the Offset from the current motion position to get the target to the trained golden pose.
<X>	The X-coordinate of the current position of the motion system
<Y>	The Y-coordinate of the current position of the motion system
<Z>	The Z-coordinate of the current position of the motion system
<A>	The Z-rotation of the current position of the motion system
	The Y-rotation of the current position of the motion system
<C>	The X-rotation of the current position of the motion system

Result String:

XT,<Status>,<X>,<Y>,<Z>,<A>,,<C>

Field	Description
XT	The command
<Status>	The result status 1: Success <=0: Fail
<X>	The X-coordinate (absolute or offset) of the corrected pick (or place) pose.
<Y>	The Y-coordinate (absolute or offset) of the corrected pick (or place) pose.
<Z>	The Z-coordinate (absolute or offset) of the corrected pick (or place) pose.
<A>	The Z-rotation (absolute or offset) of the corrected pick (or place) pose.
	The Y-rotation (absolute or offset) of the corrected pick (or place) pose.
<C>	The X-rotation (absolute or offset) of the corrected pick (or place) pose.

SLMP

Send command (Input Block)

Offset	Value	Description
VIB_Offset + 2	3012	Command: “XT” command ID

VIB_Offset + 3	<cam>	Argument0: The camera(s) used to locate target features.
VIB_Offset + 5	<resultMode>	Argument1: The requested result mode: 1: Absolute 2: Offset
VIB_Offset + 7	<X>	The X-coordinate of the current position of the motion system
VIB_Offset + 9	<Y>	The Y-coordinate of the current position of the motion system
VIB_Offset + 11	<Z>	The Z-coordinate of the current position of the motion system
VIB_Offset + 13	<A>	The Z-rotation of the current position of the motion system
VIB_Offset + 15		The Y-rotation of the current position of the motion system
VIB_Offset + 17	<C>	The X-rotation of the current position of the motion system

Receive Result (Output Block)

Offset	Value	Description
VOB_Offset + 5	3012	Command: "XT" command ID
VOB_Offset + 6	<Status>	The result Status: 1: Success <=0: Fail
VOB_Offset + 7	<X>	The X-coordinate of the result pose (absolute or offset)
VOB_Offset + 9	<Y>	The Y-coordinate of the result pose (absolute or offset)
VOB_Offset + 11	<Z>	The Z-coordinate of the result pose (absolute or offset)
VOB_Offset + 13	<A>	The Z-rotation of the result pose (absolute or offset)
VOB_Offset + 15		The Y-rotation of the result pose (absolute or offset)
VOB_Offset + 17	<C>	The X-rotation of the result pose (absolute or offset)

3. Loading a Recipe

A Recipe can only be loaded when the Sensor (Master) is in Offline mode. When in a two camera configuration the Master has finished loading the Recipe file and is set back to Online mode, it will automatically call the Slave camera to load the corresponding Recipe.

TCP

A Recipe change using TCP command is not supported by In-Sight. Therefore, a set of native commands must be sent via a telnet connection (Port 23) to the Master camera:

- Close TCP connection
- Open a Telnet connection to the Master In-Sight (user = "admin", Password = "").
- Execute native command: "SO0" (Set Offline)
- Execute native command: "LF<RecipeName>" (Load File)
- Execute native command: "SO1" (Set Online)
- Close Telnet connection
- Open TCP connection

Please note that the Recipe filename consists of the Recipe ID (3 digits) and the Recipe name, e.g.

000_DefaultRecipe.job

When the commands have been successfully executed, the client can close the Telnet connection and open the TCP connection, then send a TCP command to check the status of the sensor, i.e. whether the (optional) slave has also loaded the recipe job:

GS,0

The response will be:

GS,<status>

Status	
0	Sensor not ready
1	Sensor ready

SLMP

As opposed to TCP connection, the SLMP connection supports loading a Recipe. This is performed as follows:

- Set Master In-Sight offline by setting the "Set Offline" bit in the control block to 1
- Wait for the "Online" Bit in the Status Block to be 0
- Set the "Command" field to the desired recipe's ID
- Start the load process of the recipe with setting the "Execute Command" bit in the control block
- Wait for the "Command Executed" bit in the status block
- Set Master In-Sight online by setting the "Set Offline" bit in the control block to 0
- Wait for the "Online" Bit in the Status Block to be 1