見積番号:
ESTIMATE No.

日付:
DATE: Aug.01.2012

受注番号:
ORDER No.

御注文先:
CUSTOMER

御使用先:
USER

株式会社 安川電機

Standard Communication Specification for SR100

目次
CONTENTS

SER

備考 NOTE	配布先および部数 COPIES				担当部門 ENG.		
	御注文先 CUSTOMER	部 COPIES		部 COPIES	YASKAWA Electric Corporation Technology Section 2 Clean Robot Technology Department Robot Technology Department Robotics Division		
	御使用先 USER	部 COPIES		部 COPIES			
		部 COPIES		部 COPIES	承認 APPR.	照査 CHECK.	作成 DRAW.
		部 COPIES		部 COPIES			
		部 COPIES		部 COPIES			
		部 COPIES		部 COPIES	図番 DWG No.		
		部 COPIES		部 COPIES	HE1400615 1/135		

Revisions

Drawing No.	Item		Revised by	Approved by
HE1400615	Standard Communication Specification for SR100			
Date	Revision No.	Revision Contents		
Aug. 01 2012	0		-	-

CONTENTS

1 Introduction

This manual describes the specifications of the communications between your host controller (hereinafter called “host”) and the Yaskawa SR100 controller (hereinafter called “controller”).
Use this manual with the following:

- SR100 standard parameter list
- SR100 standard error code list

ATTENTION

- **Due to improvement to the product, specification changes, or to improve the usefulness of the manual, this manual may be changed without notice.**
- **We bear no responsibility for changes made to this product by the end user that are not warranted by Yaskawa.**
- **When performing teaching work by sending commands from the upper device, use the speeds according to ANSI standards at the time of teaching.**

2 System Configuration

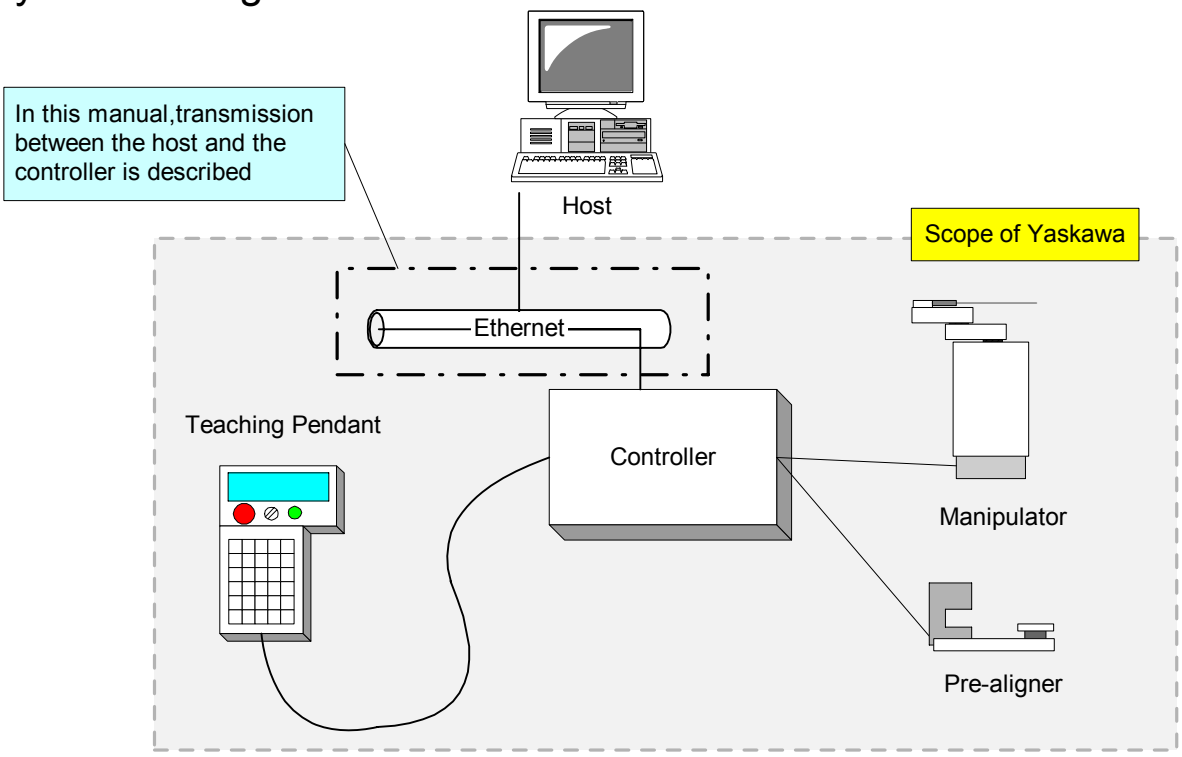


Figure 2.1 System configuration

2.1 Outline

The host and the controller are connected via Ethernet and the controller provides unified control of the manipulator(unit 1) and the pre-aligner(Unit 2).

2.2 Basic Operations of the Host and the Controller

The basic operations of the host and the controller are as follows:

2.2.1 Host

1) Sends command messages of various types to the controller and receives response message(s).
2) The commands can be motion requests, information requests, or status inquiries.
3) Retries sending command messages and handles errors when communication errors occur.

2.2.2 Controller

1) Operates the specified unit according to commands from the host and reports the results.
2) For the specified unit, sends reports in response to status report requests and configuration commands from the host.
3) Notifies the host as necessary of any errors when communication errors occur.

2.3 Controlled Units

This manual is applicable to the clean manipulator and pre-aligner.

2.3.1 Manipulator

With the clean manipulator provided in the motion sections for Arm 1, Arm 2, and Arm 3 (Blade 1, Blade 2), rotation, extension, and linear interpolation operations are available to the manipulator.

Adding the elevation axis (Z-axis), the manipulator can get and put a wafer by itself. (Refer to Figure 2.2.)

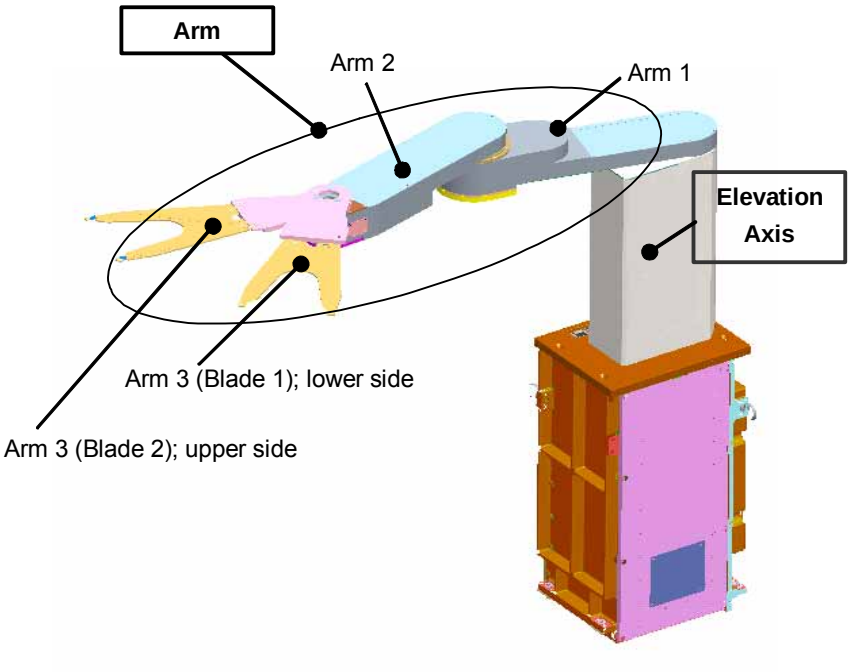


Figure 2.2 Basic configuration of the clean manipulator

(a) Manipulator Origin Posture

The origin posture of the manipulator arm(not including the elevation axis) is shown in Figure 2.3.
In the origin posture of the arm, Arm 1, Arm 2, and Arm 3 (Blade 1, Blade 2) fold up in a straight line, and, looking from the connector side, the blades are pointing directly to the right.
The elevation axis is in its lowest position.
The coordinates in the origin position are Rotation axis: 0.0 [deg]; Extension Axis: 0.0 [mm]; Wrist axis: 90.0 [deg]; Elevation axis: 0.0 [mm].

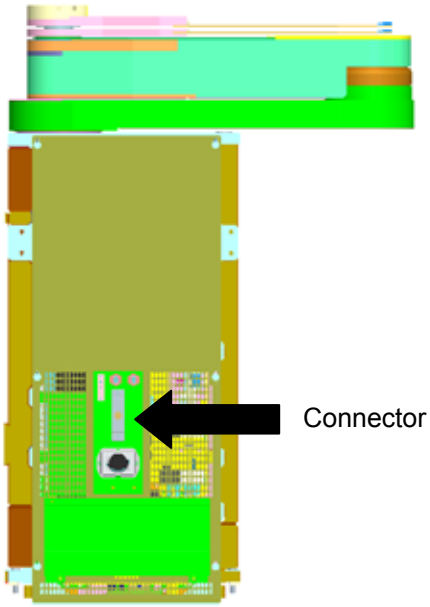


Figure 2.3 Arm Section Origin Posture

(b) Coordinates for the Manipulator's Axes

The coordinates for each of the manipulator's axes are explained below.

- Rotation axis : The angle of rotation [deg] from the origin (0.0 [deg]) of the line L (the direction where the center of rotation of Arm 2 is on the left side) that connects the center of rotation of Arm 1 and the center of rotation of the wrist axis
- Extension axis : The distance [mm] from the center of rotation of Arm 1 to the center of rotation of the wrist axis.
- Wrist axis : The angle of rotation [deg] from the line L (the direction where the center of rotation of Arm 2 is on the left side) that connects the center of rotation of Arm 1 and the center of rotation of the wrist axis.
- Elevation axis : The height [mm] from the origin (0.0 [mm]).

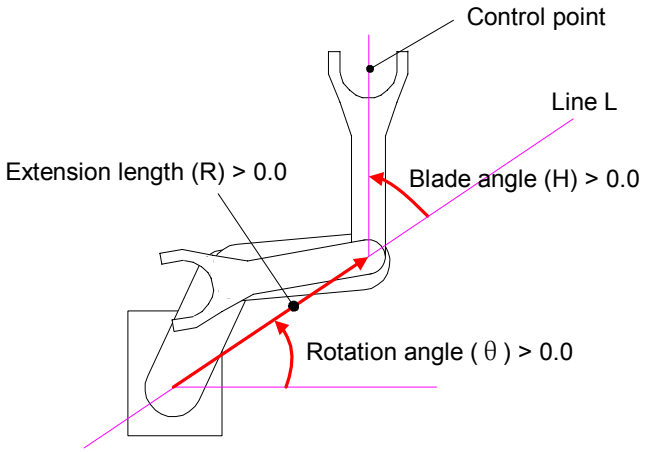


Figure 2.4 Coordinates for Arm's Axes

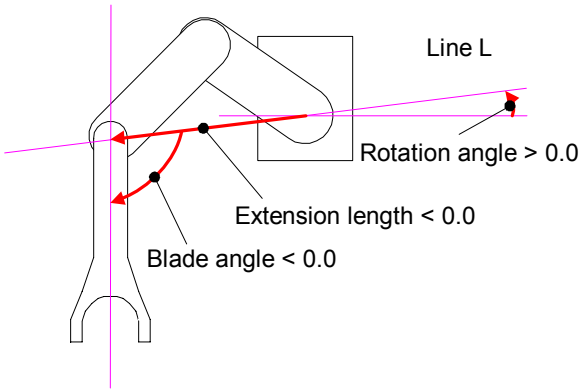


Figure 2.5 Arm posture and coordinates for left station

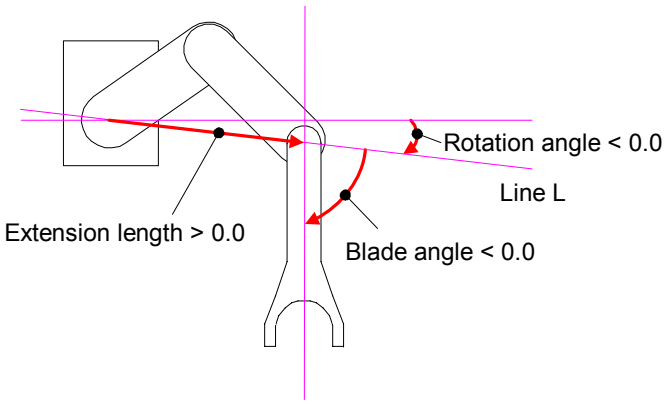


Figure 2.6 Arm posture and coordinates for right station

(c) Manipulator Axis Operations

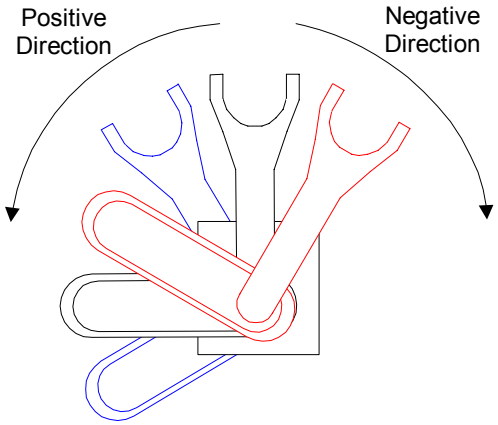
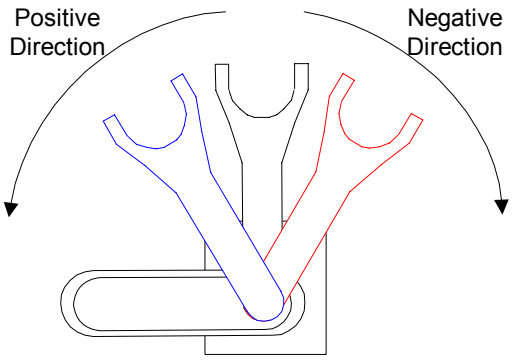
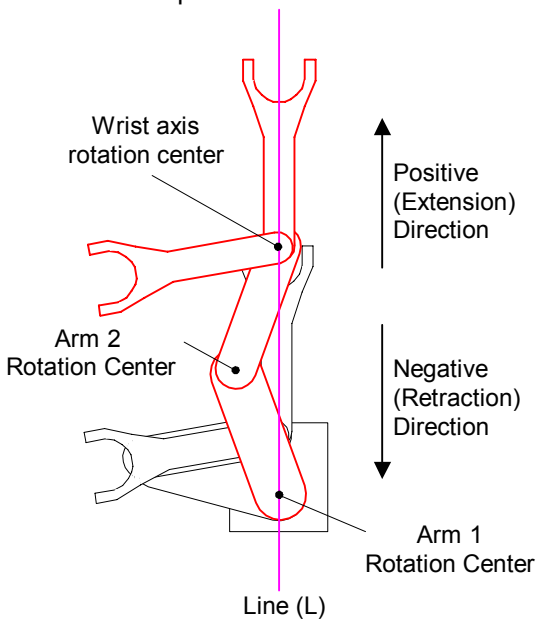
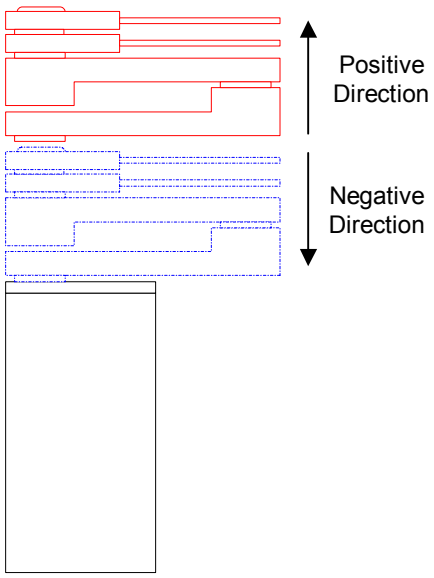
<table>
<tr><th>Rotation</th><th>Blade Rotation</th></tr>
<tr><td>Rotates the arm while maintaining its current posture.
Counterclockwise (CCW) rotation is the positive direction.
 
</td><td>Rotates the specified wrist axis while maintaining the arm's current posture. Counterclockwise (CCW) rotation is the positive direction.
 
</td></tr>
<tr><th>Extension: Link Operation</th><th>Elevation</th></tr>
<tr><td>Moves the arm forwards and backwards while maintaining the blade axis center of rotation on the line (L) that connects the Arm 1 center of rotation and the blade axis center of rotation. Moving the blade axis forward while the Arm 2 center of rotation is to the left of line L is movement in the positive direction.
 
</td><td>Moves the arm section up and down while maintaining the arm's posture. Up is the positive direction.
 
</td></tr>
</table>

Figure 2.7 Manipulator Operations

Linear Interpolation Operations (Linear Operations)

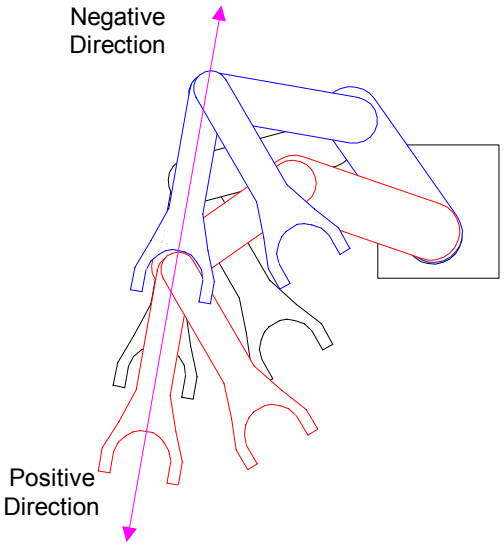
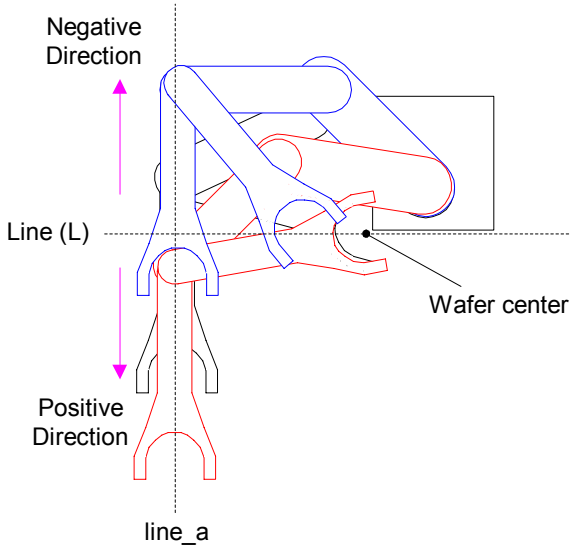
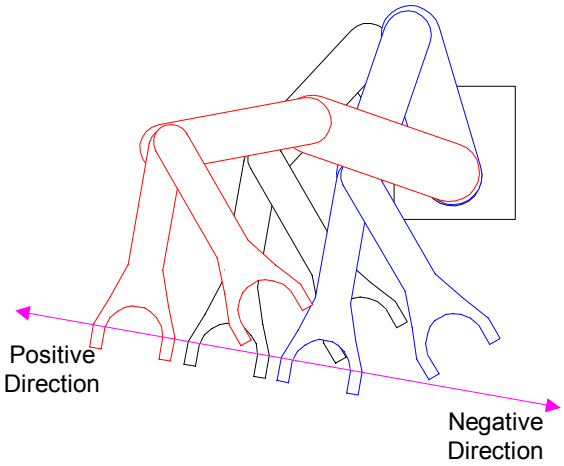
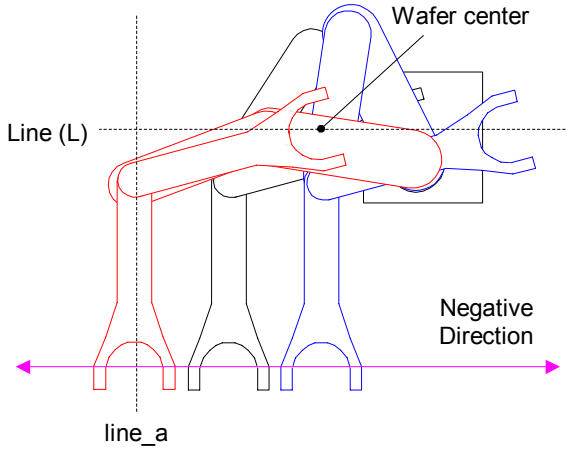
<table>
<tr><td>Front–Back Direction
(Keep Passive Blade Posture)</td><td>Front–Back Direction
(Passive Blade Fixed on Wafer Center)</td></tr>
<tr><td>While maintaining the posture of the non-specified wrist axis, the specified wrist axis moves forwards and backwards.
Forward motion of the blade is positive movement.
 
</td><td>The non-specified wrist axis stays fixed along a line L in Cartesian space centered on the wafer, and the specified wrist axis moves forwards and backwards.
Forward motion of the blade is positive movement.
 
</td></tr>
<tr><td>Horizontal Direction
(Keep Passive Blade Posture)</td><td>Horizontal Direction
(Passive Blade Fixed on Wafer Center)</td></tr>
<tr><td>While maintaining the posture of the non-specified wrist axis, the specified wrist axis moves horizontally.
Looking from the end of the blade, movement to the left is the positive direction.
 
</td><td>The non-specified wrist axis stays fixed along a line L in Cartesian space centered on the wafer, and the specified wrist axis moves horizontally.
Looking from the end of the blade, movement to the left is the positive direction.
 
</td></tr>
</table>

Figure 2.8 Linear Interpolation Operations (Linear Operations)

* The straight line L is a line drawn perpendicular from the control point of the passive blade to line_a(passing through the blade rotation center(the 2nd arm top) and the control point of the specified blade).

2.3.2 Pre-aligner

Determines the wafer's center and the notch (orientation flat) position and then aligns the notch.

It also converts the offset of the wafer center to the manipulator compensation value, which is used to adjust the manipulator when it is picking the wafer.

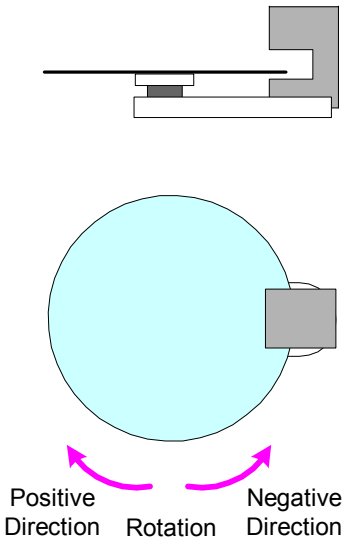


Figure 2.9 Pre-aligner

3 Communications Specifications

The host and controller can be connected either via Ethernet(TCP).

3.1 Ethernet Communication Specification (TCP)

IP address, subnet mask, default gateway and port number can be set by integer parameter. (Refer to the parameter list for the details.)

IP address:

Only the SR100 controller IP address can be configured by the controller and the host IP address is not taken set by a controller parameter.
SR100 responses to IP address when connection established.

Port number:

Port number can be set number of the top unitnumber and the port number for 2nd or following units should be the number incremented by 1 from previous unit.
(Port number can be set by an integer parameter. The number greater than “10110” can be used.)

Table 3.1 Ethernet Communication Settings

Item			Description
Communication type			Ethernet
Communication method			TCP/IP
Cable (communication speed)			10Base-T (10Mbps) / 100Base-TX (100Mbps)
Connector type			RJ-45
Transport layer protocol			TCP
IP Address	Host		10.0.0.*
	Controller		10.0.0.2
Subnet mask			255.255.255.0
Default gateway			0.0.0.0
Port number	Host		-
	Controller	Unit 1	10110
		Unit 2	10111

4 Message Transmission Specification

Communication between the host and the controller is done using ASCII-encoded messages.
Latin characters are all uppercase, with lowercase reserved for error messages.
Omitting or substituting commands and parameters (with the exception of certain commands and parameters) is not permitted and will result in an error.

4.1 Command Message (Host → Controller)

Commands sent from the host to the controller are broadly divided into the following categories:

(1)Action commands	:	Commands for moving the unit.
(2)Control commands	:	Commands, such as Servo on/off, that change the state of the unit.
(3)Setting commands	:	Commands that set parameters and position information necessary for movement, such as speed, and position registration.
(4)Reference commands	:	Commands for referencing set parameters, position information, etc., set in the unit.
(5)Acknowledgement command	:	Acknowledgement for execution completion from the controller.

A command is transmitted from the host in the following format.

Format: $,<UNo>(,<SeqNo>),<Command>,<Parameter>(,<Sum>)<CR>

1) $: Start mark (1 byte)
- Indicates the start of a message. (‘$’ = 24H)

2) , (comma): Delimiter mark (1 byte)
- Serves as a delimiter in a message. (‘,’ = 2CH)

3) UNo: Unit number (1 byte)
- Indicates the unit number
- ‘1’: Manipulator
- ‘2’: Pre-aligner

4) SeqNo: Sequence number (None / 2 bytes)
- The sequence number is used to forbid duplicate commands when a transmission error occurs.
- An integer parameter can specify the length of SeqNo (None / 2 bytes). Refer to the parameter table for more details.Default vavlue is None.
- The first SeqNo needs to be from “00” to “99” after power to SR100 controller is turned on. (Depends on length specification)
- If value is less than 2 digits, fill the higher digit with ‘0’ so that the field always has 2 digits.
- Increment SeqNo when sending next command.
- If same SeqNo was set, “Duplicated message error” occurs.
- As for the motion complete confirmation notification command (“ACKN”), increment SeqNo. (Not the same SeqNo of the motion command)
- As SeqNo is confirmed unit by unit, previous SeqNo for the other unit can be set.

5) Command: Command (4 bytes)
- Each command is defined as a fixed-length string of characters.

6) Parameter: Parameter (Differs depending on the command.)
- Sets the operation axis, the moving amount, etc. following a command.
- Multiple parameters exist, and within those some parameters may be omitted.

7) Sum: Checksum (2 bytes / None)
- This is a value for checking for malformed messages due to communication errors.
  Calculate the sum of the characters (in ASCII) between the start mark and the checksum, and then set the checksum as the lowest 2 digits in ASCII (‘0’–‘9’, ‘A’–‘F’).
- If value is less than 2 digits, fill the higher digit with ‘0’ so that the field always has 2 digits.

  Example) For the message “**$,1,INIT,1,1,G,<Sum><CR>**”, the underlined part is used to calculate the checksum.
  ‘,’ = 2CH, ‘1’ = 31H, ‘I’ = 49H, ‘N’ = 4eH, ‘T’ = 54H, ‘G’ = 47H, therefore:

  Sum = 2CH+31H+2CH+49H+4eH+49H+54H+2CH+31H+2CH+31H+2CH+47H+2CH= 316H
  (The ‘H’ following the number indicates that the number is in hexadecimal.)
  Thus, the checksum is the lowest 2 digits of 316H, 16H, and the final message is:
  “$,1,INIT,1,1,G,16<CR>”
- Checksum check function is enabled/disabled by a integer parameter. Refer to “Parameter Lists for Transfer Manipulator” for details. The default of the checksum function is enable.
  When the checksum is disabled, the two bytes for the checksum section are omitted.

8) CR: End mark (1 byte)
- Indicates the end of the message. (<CR> = 0DH)

The parentheses around <SeqNo> and <Sum> indicate that these can be omitted. Whether they are omitted can be selected in the integer parameters, and the <SeqNo> and <Sum> fields of the response message and end-of-execution message are set in the same manner.

4.2 Response Message (Controller → Host)

When a command from the host is received successfully, a response message is transmitted from the controller in the following format.

Format: $,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,<Command>,<Parameter>,
<Value>(,<Sum>)<CR>

1) $: Start mark (1 byte)
- Indicates the start of the message. (‘$’ = 24H)

2) , (comma) : Delimiter mark (1 byte)
- Serves as a delimiter in a message. (‘,’ = 2CH)

3) UNo: Unit number (1 byte)
- Indicates the unit number.

4) SeqNo : Sequence number (None / 2 bytes)
- Sequence number specified by command is responded.

5) Sts N: Status (2 byte)
- Responses the status of the unit specified by a command.
- Refer to the following table for the definition of each status bit. The value is the ASCII code of status (hexadecimal).
- Specified in the range between "00" and "FF"
- If value is less than 2 digits, fill the higher digit with ‘0’ so that the field always has 2 digits.

Table 4.1 Status

	Value	Status	Meaning
Sts1	1H	Manipulator Battery status	The battery voltage status of the specified absolute encoder 1: Low battery voltage, 0: Normal state
	2H	Unit status	The command execution status of a drive/control command of the specified unit 1: Ready, 0: Busy
	4H	Servo status	The servo status of the specified unit 1: Servo OFF, 0: Servo ON
	8H	Error status	Error status of the specified unit 1: Error occurrence, 0: No error occurrence
Sts2	1H	controller battery status	Low voltage of memory backup battery 1: Battery voltage dropped, 0: Normal status
	2H	Wafer presence Status 1	For a manipulator, shows whether there is a wafer on blade 1. For a pre-aligner, shows the wafer presence status from the vacuum sensor. (1: Has wafer, 0: No wafer)
	4H	Wafer presence Status 2	For a manipulator, shows whether there is a wafer on blade 2. For a pre-aligner, shows the wafer presence status from the CCD sensor. (1: Has wafer, 0: No wafer)
	8H	Reserve	

Example) If some error occurs, servo power is off, and both blades 1 and 2 have a wafer, the status will be (the ‘H’ after a number indicates the number is in hexadecimal):
Servo OFF = 4H, Error status = 8H, so 4H + 8H = CH, Sts1 is 43H (the ASCII code for ‘C’)
Wafer presence status 1 = 2H, Wafer presence status 2 = 4H
2H + 4H = 6H, Sts2 is 36H (the ASCII code for ‘6’)
Thus, “C6” is returned for Sts.

6) Ackcd: Response code (4 bytes)
- Verify the command message sent from the host and indicates it is executable or not.
- Ackcd is “0000” when normal reception. The response code consists of the detail code (3 bytes) and the error level (1 byte) that are described in “Error Code Lists for Transfer Manipulator when abnormal reception.” The error level is indicated by the following ASCII code and added to the most significant byte.
  ‘1’: Warning 1 (W1), ‘2’: Warning 2 (W2), ‘3’: Important alarm 1 (A1), ‘4’: Important alarm 2 (A2), ‘5’: Serious alarm (F)
  Example) If an “T.P Emergency Stop” error occurs, the error level is Important Alarm 1 (A1) and the error code is “A20”, so the response code is “3A20”.

7) Command: Command (4 bytes)
- The response contains contents added to the command message from the host in a fixed-length character string.

8) Parameter: Parameter (differs depending on the command)
- The response contains information such as the motion axis and amount of movement added to the command message.

9) Value: Response data (Definition depends on each command)
- Response value for the reference command sent from the host. Refer to the description of each command.

10) Sum: Checksum (2 bytes / None)
- Same as the checksum described for the command messages.

11) <CR>: End mark (1 byte)
- Indicates the end of the message. (<CR> = 0DH)

4.3 Response Message for Erroneous Reception (Controller → Host)

When a command from the host is *incorrectly received*, a response message is transmitted from the controller in the following format.

Possible cause of erroneous reception: (For details, refer to the Error Code Table)
- Checksum error
- Incorrect unit number
- Incorrect sequence number
- Duplicated message
- Parameter error
- Delimiter error
- Undefined command error

Format: ?,<Ackcd>(,<Sum>)<CR>

1) ?: Start mark (1 byte)
- Indicates the start of the message for erroneous reception. (‘?’ = 3FH)

2) , (comma) : Delimiter mark (1 byte)
- Serves as a delimiter in a message. (‘,’ = 2CH)

3) Ackcd: Response code (4 bytes)
- Same as the response code described for response messages.

4) Sum: Checksum (2 bytes / None)
- Same as the checksum described for the command messages.

5) CR: End mark (1 byte)
- Indicates the end of the message. (<CR> = 0DH)

4.4 End-of-execution Message (Controller → Host)

When the execution of the drive command or control command has been completed, message is transmitted from the controller in the following format.
The transmission of this message can be disabled/enabled by the integer parameter. (The default is to require the notification.)

Format: !,<UNo>(,<SeqNo>),<Sts>,<Errcd>,<Command>
,<ExeTime>,<PosData1>…,<PosDataN>,<Value>(,<Sum>)<CR>

1) !: Start mark (1 byte)
- Indicates the start of the message. (‘!’ = 21H)

2) , (comma) : Delimiter mark (1 byte)
- Serves as a delimiter in a message. (‘,’ = 2CH)

3) UNo: Unit number (1 byte)
- Indicates the unit number.

4) SeqNo: Sequence number (None / 2 bytes)
- Sequence number specified by command is responded.

5) Sts: Status (2 bytes)
- The same as in a response message.

6) Errcd: Error code (4 bytes)
- Indicates the error that occurred when the contents of the requested command from the host is executed.
- The contents of the error are the same as the response code in a response message.

7) Command: Command (4 bytes)
- Command name for which execution was completed

8) ExeTime : Execution time (6 bytes)
- The response contains the execution time. (Resolution: 1[msec])
- Specified in the range between "000000" and "999999"
- If value is less than 6 digits, fill the higher digit with ‘0’ so that the field always has 6 digits.

9) PosData* : Coordinate data (Each 8 bytes)
- Responds with the feedback position at the end of execution. (Resolution: 0.001 [deg] or 0.001 [mm])
- Specified in the range between "-9999999" and "99999999"
- If value is less than 8 digits, fill the higher digit with ‘0’ so that the field always has 8 digits.
- Responds with the specified unit’s axis count part.

Unit	Coordinate data	Axis	Unit
Manipulator	PosData1	Rotation axis	0.001 [deg]
	PosData2	Extension axis	0.001 [mm]
	PosData3	Wrist axis 1	0.001 [deg]
	PosData4	Wrist axis 2	0.001 [deg]
	PosData5	Elevation axis	0.001 [mm]
Pre-aligner	PosData1	Rotation axis (*1)	

(*1) If the specified unit is a pre-aligner, be sure to respond with “00000000”.

10) Sum: Checksum (2 bytes / None)
- Same as the checksum described for the command messages.

11) CR: End mark (1 byte)
- Indicates the end of the message. (<CR> = 0DH)

4.5 End-of-execution confirmation message (Host → Controller)

This is an *acknowledgement from the host that acknowledge the end-of-execution message from the controller.*

The controller waits this message with preset timeout value when it is enabled by the integer parameter. (The default is to require the notification.)

When the completion message (from the controller) is disabled by the integer parameter, check of the acknowledgement from the host is automatically disabled regardless of the parameter setting of acknowledgement check. The format is shown bellow.

If there is no end-of-execution confirmation from the host, the controller will send an end-of-execution message again.

After trying to send the end-of-execution message a certain number of times, if no confirmation message is received from the host, a “Receiving Time-out Error for Confirmation of Execution Completion” will occur.

Retry times can be set by the integer parameter.(The default is 2 times)

Format: $,<UNo>(,<SeqNo>),ACKN(,<Sum>)<CR>

1) $: Start mark (1 byte)
   - Indicates the start of the message. (‘$’ = 24H)

2) , (comma) : Delimiter mark (1 byte)
   - Serves as a delimiter in a message. (‘,’ = 2CH)

3) UNo: Unit number (1 byte)
   - Indicates the unit number.

4) SeqNo: Sequence number (1 byte)
   - Indicates the sequence number.

5) ACKN : Command (4 bytes)
   - Indicates the end-of-execution confirmation notification.

6) Sum: Checksum (None / 2 bytes)
   - Same as the checksum described for the command messages.

7) <CR>: End mark (1 byte)
   - Indicates the end of the message. (<CR> = 0DH)

4.6 Event Notification Message (Controller → Host)

Predefined event message is sent from controller to the host in following format.
The host does not need to reply to the controller regarding event notifications but will perform some type of processing, as necessary.

Even notifications can be enabled or disabled according to the type of event. (Selected in the integer parameters)

Format: >,<UNo>,EVNT,<EvNo>,<Date>,<EvData>(,<Sum>)<CR>

1) >: Start mark (1 byte)
   - Indicates the head of event message. (‘>’ = 3EH)

2) , (comma) : Delimiter mark (1 byte)
   - Serves as a delimiter in a message. (‘,’ = 2CH)

3) UNo: Unit number (1 byte)
   - Indicates unit number.

4) EVNT: Command(4 bytes)
   - Indicates event command.

5) EvNo: Event number (3 bytes)
   - Shows type of event.
   - Shows detail of event in the Table 4.2.

6) Date : The date and time the event occurred on (19 bytes)
   - Response is in the form <Year>/<Mon>/<Day>□<Hour>:<Min>:<Sec>. (□ indicates a blank space.)
   - Year: The year (4 bytes)
   - Mon: The month (2 bytes)
   - Day: The day (2 bytes)
   - Hour: The hour (2 bytes)
   - Min: The minute (2 bytes)
   - Sec: The second (2 bytes)
   - ‘/’ (2FH), ‘□’ (20H), ‘:’ (3AH) indicate delimiters in the date and time.

7) EvData: Event data (N bytes)
   - Shows event name. Data length depends on event number.
   - Event details are explained in the Table 4.2.

8) Sum: Checksum (2 bytes / Non)
   - Same as command message.

9) <CR>: End mark (1 byte)
   - Indicates the end of the message. (<CR> = 0DH)

Table 4.2 Event Notification Contents

EvNo	Event Contents(EvData)	Event Notification Timing	Default
100	Error notification event “,<ErrCd>” • ErrCd : Error code (4 bytes) • The same as the response code in a response message.	• Notified when error happened in host mode. • As for the error that is common between the units (e.g. emergency stop), event message is sent to all of the units.	Enabled
140	Alignment result event “,<ErrCd>,<Value1>…,<Value10>” • Errcd : Error code(4 Bytes) • Same as response code of response message • Value* : alignment result(each 8 Byte) • Same as **“MALN”**, **“RALN”**command	• If the controller receive **“MTRS”**,**”MPNT”**,**”MCTR”**command with alignment angle,The pre-aligner execute the alignment motion automatically when the manipulator put the wafer on the PA stage. The event is sent when the aligment motion complete.	Enabled

5 Communications Flow

This chapter describes the communications flow between the host and the controller.

5.1 Host Processing

(1) Sends a connection request to the controller port.
(2) For each new command, the previous sequence number is incremented by 1 to make a new sequence number. (When sequence numbers are being used)
(3) Receives the response message (received successfully, received with error) from the controller.
(4) Waiting for the response from the controller is limited by the response timer.
If the response timer expires, the probability of a malfunction in the communication medium is high, so communication is stopped. (No resending is done)
(5) The method of reception from the controller differs depending on the type of command as follows:
[Action command, Control command]
- Receives an end-of-execution message from the controller.
If an error is detected in the received message (e.g., a checksum error), that message is discarded.
- When an end-of-execution message is received from the controller, an end-of-execution confirmation message is sent to the controller.
The end-of-execution confirmation message notifies the controller that the host is aware that execution has stopped and that it is unnecessary for the controller to resend the end-of-execution message.
Also, when the host is waiting to receive an end-of-execution message from the controller and there is a malfunction in the communication medium, the host will be locked in a state (an infinite loop) waiting for the end-of-execution. Therefore, it is necessary to set an operation timer.

[Other commands]
- Receives the response for the command.

(6) The host receive the event notification, if necessary, the host shall perform any processing (response of event notification is not required).

The completion message and the acknowledgement for the completion message can be omitted by the integer parameters. Refer to “Parameter Lists for Transfer Manipulator” for details.

5.2 Controller Processing

(1) TCP requires connection establishment before starting the command transaction.

(2) For data received from the host, after receiving the '$' character (indicating the beginning of a command message), commands are received (and held in a buffer) **with between-character timeout monitoring** until it receives the <CR> character (indicating the end of the message). If a time-out between characters is detected while receiving the messages, the remainder of that entire message (the received characters and the following characters up to <CR>) will be ignored. The controller does not send any error notification to the host. The time-out time between characters can be set by the parameter.

(3) After the entire command message is received from the host, its checksum is verified. If a checksum error is detected, the controller informs the host of the error.

(4) Checks whether the received command's sequence number matches the previous number.
Prevent the double operation by saving the sequence number of the received command that the sequence number is not the same (When sequence numbers are being used)

(5) If received correctly, processing is performed as follows based on the type of command:
[Action commands, Control commands]
- The whether the command can be executed and parameters are checked, and if there are any errors the host is notified.
- When execution is finished, an end-of-execution message is sent to the host.
- Receives an end-of-execution confirmation message from the host. If the response timer expires, the message is sent again. (To deal with cases where the message cannot be successfully sent due to malfunctions in the communication medium, a maximum retry counter is set. If the counter expires without the message being successfully sent, no more communication is done.)

[Other commands]
- Whether the command can be accepted and parameters are checked, and if there are any errors the host is notified.

Response time-out time and the maximum number of retries can be set by the integer parameters.
The execution completion message and the acknowledgement message for the completion message can be enabled or disabled by the integer parameters. Refer to "Parameter Lists for Transfer Manipulator" for details.

5.3 Normal Communication Flow

5.3.1 Communications Flow Chart for the action commands

- Even while an action command is executing, a reference command can be used to confirm various statuses, the current position, etc.
- While the unit is executing (unit status: Busy), the next action command cannot be issued.
  To check the unit's status:
- Check the response status of the "RSTS" communication command.
- Through the IO board, notify the unit's status to the host. (Separate option support)

Host		Controller	Unit Status
			Ready
Send action or control command.	---------->		Ready
(SeqNo: "01")	<----------	Sends the response (normal).	Busy
		(SeqNo: "01")	Busy
		(Executing the command.)	Busy
			Busy
	<----------	Sends the end-of-execution message.	Busy
Sends an acknowledgement for the end-of-execution message(**"ACKN"**).	---------->	(SeqNo: "01")	Busy
(SeqNo: "02")			Ready*1
			Ready

*1 The maximum time to change the unit status from Busy to Ready is 2 msec.

5.3.2 Communications Flow Chart for except the action commands

There is no end-of-execution message in the status reference, the position reference/setting, and operating speed reference/setting commands. Therefore, the host identifies completion of the command by receiving the response message.

Host		Controller	Unit Status
			Ready
Sends a command.	---------->		Ready
(SeqNo: "03")	<----------	Sends the response (normal).	Ready
		(SeqNo: "03")	Ready

5.4 Communications Flow Chart for Interruption and Restart of Operations

Available the operation interruption (“CSTP” command) and restarting (“CRSM” command).

Host		Controller	Unit Status
			Ready
Send action command	---------->		Ready
(SeqNo: “04”)	<----------	Sends a response (normal).	Busy
		(SeqNo: “04”)	Busy
			Busy
			Busy
Sends an interruption command (**“CSTP”**).	---------->		Busy
(SeqNo: “05”)	<----------	Sends a response (normal).	Busy
		(SeqNo: “05”)	Busy
	<----------	Sends a end-of-execution message.	Busy
Sends an acknowledgement for the end-of-execution message (**“ACKN”**).	---------->	(SeqNo: “05”)	Ready
(SeqNo: “06”)			Ready*1
			Ready
Sends a restart command (**“CRSM”**).	---------->		Ready
(SeqNo: “07”)	<----------	Sends a response (normal).	Busy
		(SeqNo: “07”)	Busy
		(Executing the action command.)	Busy
			Busy
	<----------	Sends a end-of-execution message.	Busy
Sends an acknowledgement for the end-of-execution message (**“ACKN”**).	---------->	(SeqNo: “04”)	Ready
(SeqNo: “08”)			Ready*1
			Ready

*1 The maximum time to change the unit status from Busy to Ready is 2 msec.

When the action command was restarted by the resume command, the completion message of the action command has the original sequence number of the action command. (Not the resume command's sequence number)

5.5 Processing for Communications Error

This section describes the processing when a communications error (excluding an error in command message contents) occurs due to message transmission problems.

5.5.1 When an Error Occurs in a Message from the Host

The following causes are suspected for errors in a message from the host.

1) Garbage character received instead of '$' code (Missing character)
<CR> code is detected before '$' code.

Host		Controller
Sends a command. (SeqNo: "01")	---------->| Garbage character received instead of "$"	(Message head not identified) Cancels the message.
<**Response time-out**> Sends the command again. (SeqNo: "01")	----------> <----------	Sends the response (normal). (SeqNo: "01")

2) Garbage character received instead of <CR> Code (Missing character)
The end of the message is not detected.

Host		Controller
Sends a command. (SeqNo: '1')	---------->| Garbage character received instead of <CR>	Cancels the message. (**Time-out between characters**)
<**Response time-out**> Sends the command again. (SeqNo: '1')	----------> <----------	Sends the response (normal). (SeqNo: '1')

3) Garbage Character Other Than the Above Two (Missing Character)
The controller detects the checksum error if any errors are found with characters in a command message other than '$' and <CR> codes.

Host		Controller
Sends a command. (SeqNo: "01")	----------> Garbage character other than '$' and <CR> codes <----------	Cancels the message. (**Checksum error**) Sends the response (erroneous).
Sends the command again. (SeqNo: "01")	----------> <----------	Sends the response (normal). (SeqNo: "01")

5.5.2 When an Error Occurs in a Response Message from the Controller

The same communications error may occur in a response message from the controller as in a message from the host. Communications errors identified by the host are not informed to the controller.
If the command from the host are execution type commands, the controller will begin executing commands from the time the initial command is received. (The controller cannot recognize when a communication error occurs.)

If such cases when a communication error occurs, the controller may respond with an error ("cannot execute error") to the re-send command.
The end-of-execution may also be received before the response timeout occurs,
However, when the "sequence number" is enabled in command message, the controller will respond to the re-send command with a "duplicated message error".

1) Garbage character of '$' and <CR> codes (Missing character)

< No sequence number >

Host		Controller
Sends a command.	---------->	
	<----------	Sends the response (normal).
	Garbage character received instead of '$' or <CR>	
<Response time-out>		
Sends the message again.	---------->	
	<----------	Sends the response (normal).

< With sequence number >

Host		Controller
Sends a command. (SeqNo: "01")	---------->	
	<----------	Sends the response (normal). (SeqNo: "01")
	Garbage character received instead of '$' or <CR>	
<Response time-out>		
Sends the message again. (SeqNo: "01")	---------->	
	<----------	Sends the response (abnormal).

2) Garbage character other than above (Missing character)

< No sequence number >

Host		Controller
Sends a command. Cancels the message. (**Checksum error**)	---------->` <---------- Garbage character received instead of character other than '$' or <CR>	Sends the response (normal).
Sends the command again.	----------> <----------	Sends the response (normal).

< With sequence number >

Host		Controller
Sends a command. (SeqNo: "01") Cancels the message. (**Checksum error**)	----------> <---------- Garbage character received instead of character other than '$' or <CR>	Sends the response (normal). (SeqNo: "01")
Sends the command again. (SeqNo: "01")	----------> <----------	Sends the response (abnormal).

5.5.3 When an Error Occurs in a End-of-Execution message from the Controller

When a communications error occurs in the end-o-execution message from the controller to the host, the controller sends the end-of-execution message again after the response time-out of host acknowledgement for the end-of-execution message is detected.

Host		Controller
	<---------- Any garbage character	Sends a completion message. (SeqNo: “01”)
	<----------	(**Response time-out**) Sends a completion message again.
Sends an acknowledgement for the execution completion. (SeqNo: “02”)	---------->	(SeqNo: “01”)

5.5.4 When an Error Occurs in an Acknowledgement for the completion message

When a communications error occurs in a completion message acknowledgement from the host, the controller detects the response time-out and sends the completion message again.

Host		Controller
	<----------	Sends a message for the completion of execution.
Sends an acknowledgement for the completion message. (SeqNo: “02”)	---------->	(SeqNo: “01”)
	Garbage character other than ‘$’ or <CR>	
	<----------	(**Response time-out**) Sends the message for the completion of execution again.
Sends an acknowledgement for the completion message. (SeqNo: “02”)	---------->	(SeqNo: “01”)

6 Teaching Position and Basic Transfer Operations

Teaching only does station positioning for each wafer transfer destination.
The teaching box(hereafter T.P) or Host is used to the teaching operation.
The transfer position for wafer transfer and position for wafer mapping operation are automatically generated by the teaching position.
The transfer stations are classified into 3 station types (cassette stage, transfer stage, PA stage).

6.1 Cassette Stage

This type of station has multiple slots, such as open cassette, FOUP, and multi-stage, spaced evenly apart.
The cassette stage teaching spots are different depending on the slot position generation mode (the following 2 modes).

(1) Custom setting mode
This method is for teaching only the lowest slot of the cassette stage.
The slot position specified from the host is automatically generated from the lowest slot's teaching position and a custom set slot pitch.

(2) Automatic calculation mode (Z direction)
Teaching is done for 2 positions: the cassette stage's lowest slot and its highest slot.
The slot position specified from the host is automatically generated from the slot pitch derived by dividing the elevation coordinates of the highest and lowest slots' teaching position by (slot count – 1).
Each slot's rotating and extension axes use the lower slot's teaching position.

6.2 Transfer Stage

This type of station does not have multiple slots.

6.3 Pre-aligner Stage (PA stage)

This type of station does not have multiple slots.
The wafer center offset detected by the pre-aligner can be automatically compensated for when the manipulator is receiving the wafer.

6.4 Types of Registration Positions

(1) Lowest slot position
This position is the transfer position for any kind of the stations.
The cassette stations teach position as the lowest slot.

(2) Highest slot position
This position is for the cassette stage only and has to be taught when the slot pitch automatic calculation mode (described later) is selected.

6.5 Station Count

The number of stations, consisting of cassette stages, transfer stages, and PA stages, is shown in Table 6.1 Station Count.

Table 6.1 Station Count

Transfer Station		Type of Registered Position	
		Lowest Position	Highest Position
Cassette stage	8 stages	OK	OK
Transfer stage	12 stages	OK	-
PA stage	1 stage	OK	-

6.6 Positions Other than Transfer Station

Other than the transfer station previously mentioned, there are the following positions, which are generated automatically in the inside of the controller.

6.6.1 HOME Position

This is the position when the unit moves into the posture previously explained in “Manipulator Origin Posture”. On the “INIT” command, the unit moves to the origin position.

6.6.2 Ready Position

This is the position where the manipulator begins inserting the wafer into the transfer station and where the wafer is taken out of the transfer station, and is defined by the registered position and “**Transfer Offset**”, etc.

6.6.3 Intermediate Position

The intermedinate position can be set between the raedy position and the station position.

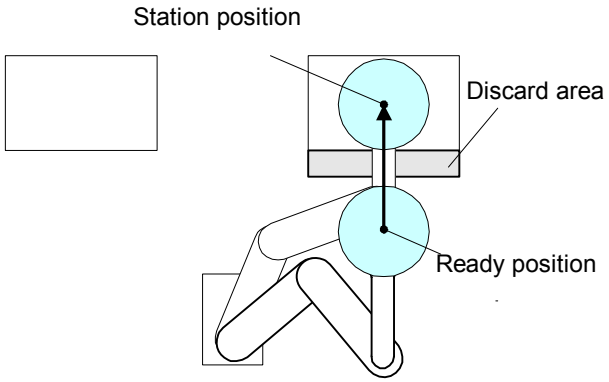


Figure 6.1 Ready Position When No Intermediate Position Exists

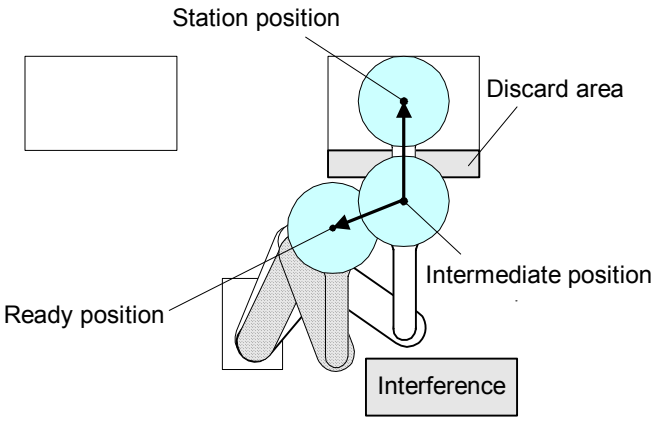


Figure 6.2 Ready Position and Intermediate Position When an Intermediate Position Exists

6.6.4 Positions Related to Mapping Operation

The positions for mapping operation are the mapping start position, the mapping end position, and the wafer protrusion detection start position.

These positions are automatically generated from the cassette stage's lowest slot's teaching position.

Figure 6.3 shows the teaching positions related to mapping operation.

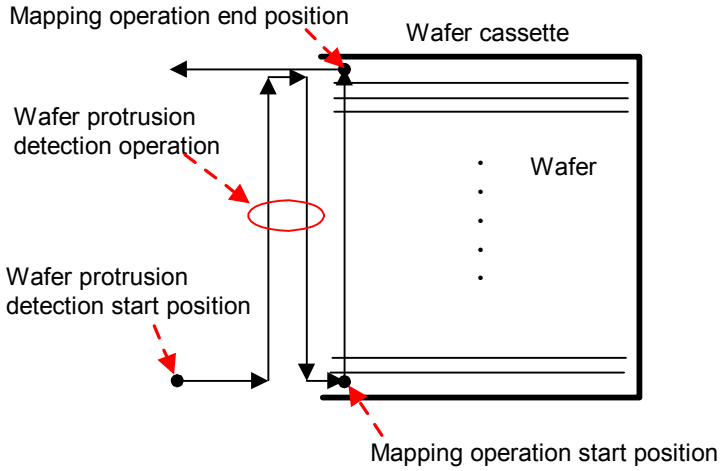


Figure 6.3 Positions Related to Mapping Operation

6.6.5 The registration posture of reference position

An absolute position encoder is loaded in the motor, so it is necessary to recognize an absolute value in a "specific arm posture".

The registration posture of reference position refers to this "specific arm posture" .

Refer to "manipulator instruction manual" about the registration posture of reference position.

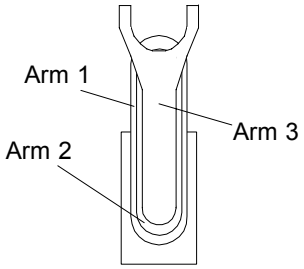


Figure 6.4 Reference Position Posture

6.7 Transfer Point Generation and Motion pass

The wafer Get operations, Put operations, and Exg (exchange) operations are realized by operating between the **transfer points** generated by the teaching positions and **transfer offset** (set in parameters).
The transfer offset is preset from controller parameters for each transfer station, and can be set as desired at the host or teaching pendant.

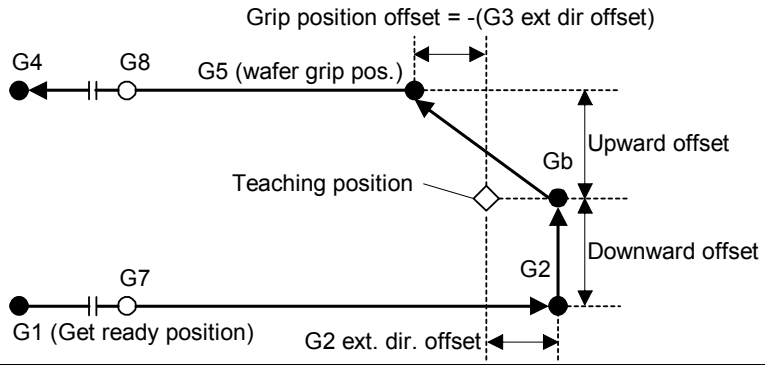
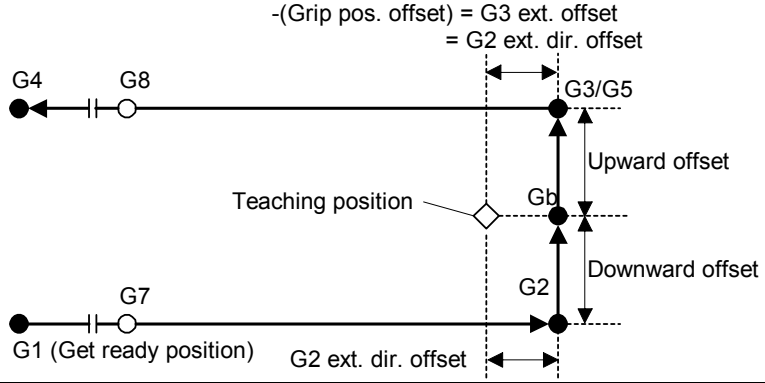
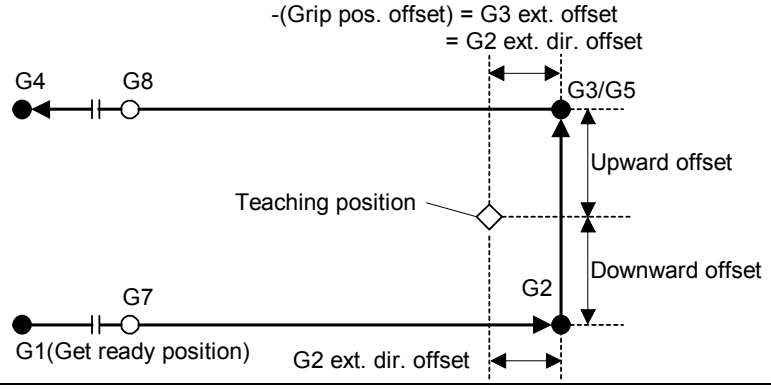
The manipulator supports the following 2 gripping methods, and the transfer points differ depending on the wafer grip method.

- Edge grip type: Blade that pushes against and holds the edge pad section
- Vacuum type : Blade that holds the wafer through vacuum

6.7.1 Edge Grip Type Transfer Point

(1) Get operation's transfer point and Get motion pass

The transfer point and motion pass for the Get operation (operation from Ready position to Get completion position) are shown in Figure 6.5 The Get motion pass differs depending on the station.

Station	Motion pass
Cassette stage	
PA stage	
Transfer stage	

●: Operation pause point

Figure 6.5 Edge Grip Type's Get Operation Transfer Point Motion pass

<Offset value>
The following parameters are available for the transfer offsets that defines the transfer point.

- Upward offset
  In the Get operation, this value is the elevation axis offset for lifting up the wafer from the teaching position.
- Downward offset
  In the Get operation, this value is the elevation axis offset from the teaching position for extending toward the station position.
- G2 (P3) extension direction offset parameter
  In the Get operation, this value is the offset in the extension direction from the teaching position that generates the G2 point.
- G3 extension direction offset parameter
  In the Get operation, this value is the offset in the extension direction from the teaching position that generates the G3 point.
- Gripping position offset parameter
  In the Get operation, this value is the offset in the extension direction that defines the holding position for the grip.

<Transfer points>

- Gb: Position obtained by adding the extension offset to the teachin position.
- G1: GET ready position.
- G2: Position obtained by adding the downward offset and G2 (P3) extension direction offset to the teaching position.
- G3: Wafer Get position obtained by adding the upward offset and the G3 extension direction offset to the teaching position .
- G4: Wafer Get end position.
- G5: Grip operation position obtained by adding the grip position direction offset to the teaching position.
- G7: Intermediate position. (Optional)
- G8: Intermediate position. (Optional)

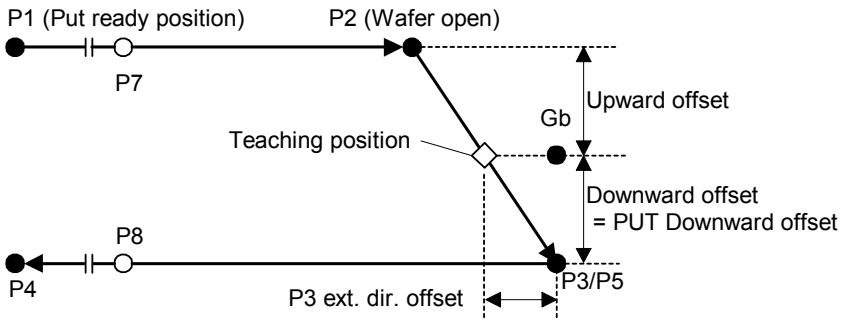
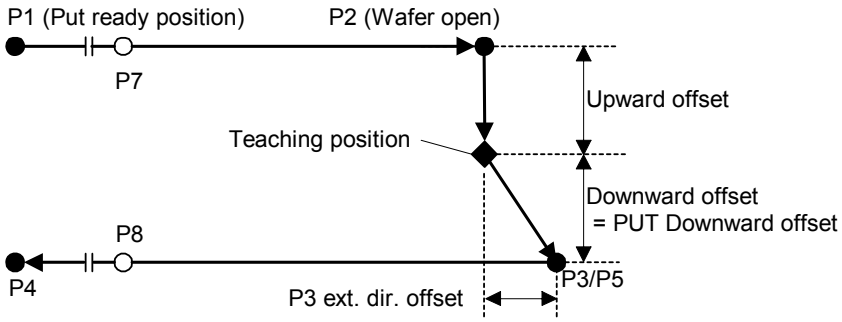
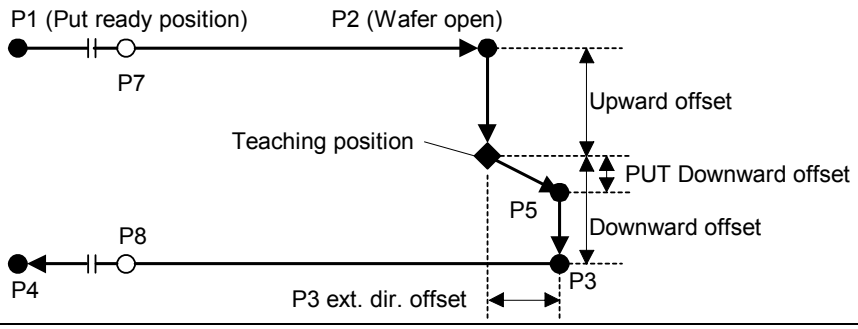
<Get Operation Sequence>
When the controller received the Get operation command from the host, the Get operation is performed in the following sequence.

(1) Confirm there is no wafer at the current position.
(2) Move to G1 (Get ready position).
(3) Move from G7 (if it exists) to G2.
(4) If at the cassette stage or PA stage, move to Gb. Pass by if at the transfer stage.
(5) Move to G5.
(6) The solenoid ON, and confirm whether a wafer is being held.
(7) If there is a wafer in (6): move from G8 (if it exists) to G4.
    If there is no wafer in (6): Try again, and if there is still no wafer, stop on an error.
    If you try again and there is a wafer, move from G8 to G4.
    Refer to 7.3 about retry motion.

(2) Put operation transfer points and Put motion pass

The transfer points for the Put operation (operation from Ready position to Put end position) are shown in Figure 6.6.

The Put motion pass differs depending on the station.

Stage	Motion pass
Cassette stage	
PA stage	
Transfer stage	

●, ◆: Operation pause point

Figure 6.6 Edge Grip Type's Put Operation Transfer Point and Motion pass

<Offset value>
The following parameters are available for the transfer offsets that define the transfer points.

- Upward offset (Shared parameter with Get operation)
  This value is the elevation axis offset from the teaching position for extending toward the station position.
- Downward offset (Shared parameter with Get operation)
  This value is the elevation axis offset from the teaching position for retracting toward the Put end position.
- (G2) P3 extension direction offset parameter (Shared parameter with Get operation)
  In the Put operation, this value is the offset in the extension direction from the teaching position that generates the P3 point.
- Gripping position offset parameter (Shared parameter with Get operation)
  In the Put operation on the cassette stage, this value is the offset in the extension direction that defines the holding position for the grip.
- Put downward offset (Put operation exclusive parameter)
  This value is the elevation axis offset for the sloped Put operation in order to prevent the wafer from being caught on the pad of the end of the blade.

<Transfer points>
- P1: Wafer put ready position.
- P2: The station extension position obtained by adding the upward offset from the teaching position and the P5 point slope from the teaching position.
- P3: The position obtained by adding the downward offset from the teaching position and the (G2) P3 extension direction offset.
- P4: Wafer handoff end position.
- P5: The sloped operation position obtained by adding the Put downward offset from the teaching position and the (G2) P3 extension direction offset.
- P7: Intermediate position. (Optional)
- P8: Intermediate position. (Optional)

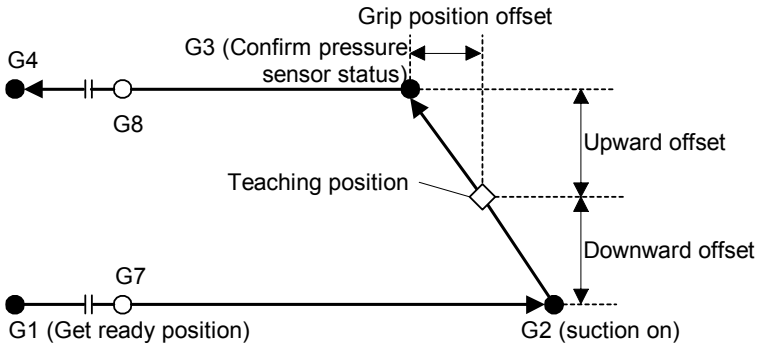
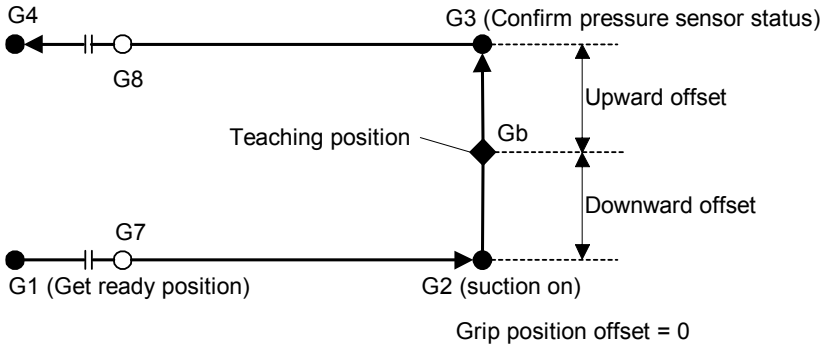
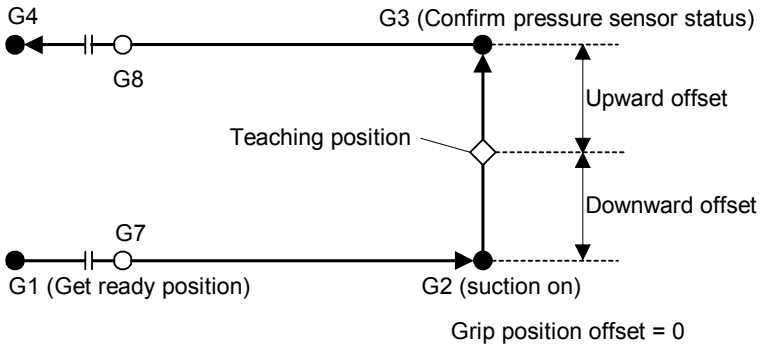
<Put operation sequence>
When a Put operation command is received from the host, the Put operation is performed in the following sequence.

(1) Confirm there is a wafer at the current position.
(2) Move to P1 (Put ready position).
(3) Move from P7 (if it exists) to P2.
(4) Turn the solenoid off, and do wafer open confirmation.
(5) For cassette stage: Move to P3 (= P5).
  For PA stage: Move from the teachin position to P3 (= P5).
  For transfer stage: Move from the teaching position to P5 to P3.
(6) Move to P4.

6.7.2 Vacuum Type Transfer Points

(1) Get operation's transfer points and Get motion pass

Figure 6.7 shows the Get operation's (operation from ready position to Get end position) transfer point. The Get motion pass differs depending on the station.

Station	Motion pass
Cassette stage	
PA stage	
Transfer stage	

●, ◆: Operation pause point

Figure 6.7 Vacuum Type's Get Operation Transfer Points and Motion pass

<Offset value>

The following parameters are available for the transfer offsets that define the transfer points.

- Upward offset
  This value is the elevation axis offset for lifting the wafer from the teaching position when the get motion is executed.
- Downward offset
  This value is the elevation axis offset from the teaching position for extending toward the station position when the get motion is executed.
- Gripping position offset parameter
  This value is the offset in the extension direction from the teaching position for the sloped Get operation.

<Transfer points>

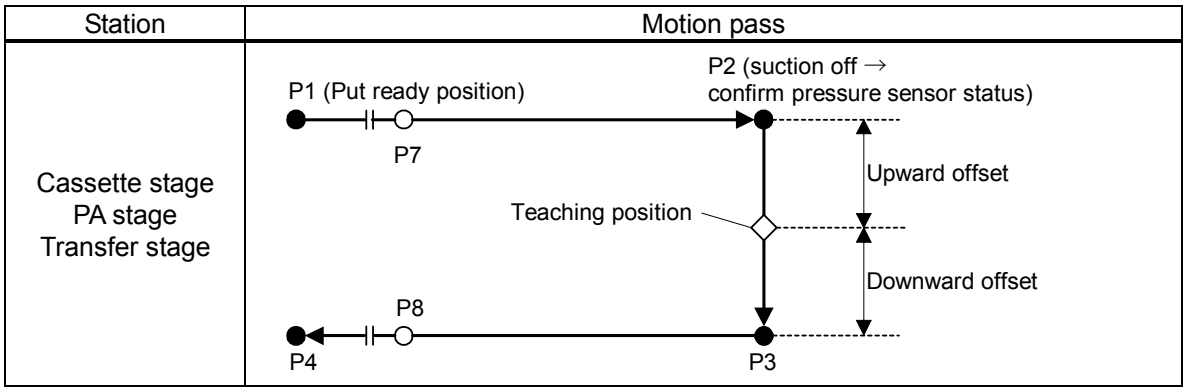
- G1: Wafer Get ready position.
- G2: Station extension position obtained by adding the downward offset from the teaching position, or station extention position obtained by adding the downward offset and G3 point's slope from the teaching position.
- G3: Wafer Get position obtained by adding the upward offset and holding position offset to the teaching position.
- G4: Wafer Get end position.
- G7: Intermediate position. (Optional)
- G8: Intermediate position. (Optional)

<Get Operation Sequence>

When a Get action command is received from the host, the Get operation is performed in the following sequence.

(1) Confirm there is no wafer at the current position.
(2) Move to G1 (Get ready position).
(3) Move from G7 (if it exists) to G2.
(4) Turn the solenoid on.
(5) Cassette stage: Move to G3.
PA stage: Move from the teaching position to G3.
Transfer stage: Move to G3.
(6) Confirm wafer presence (pressure sensor on).
(7) Move from G8 (if it exists) to G4.

(2) Put operation transfer points and Put motion pass
The transfer points for the Put operation (operation from Ready position to Put end position) are shown in Figure 6.8.



●, ◆: Operation pause point

Figure 6.8 Vacuum Type's Put Operation Transfer Points and Motion pass

<Offset value>
The following parameters are available for the transfer offsets that define the transfer points.

- Upward offset (Shared parameter with Get operation)
  This value is the elevation axis offset from the teaching position for extending toward the station position.
- Downward offset (Shared parameter with Get operation)
  This value is the elevation axis offset from the teaching position for retracting toward the Put end position.

<Transfer points>
- P1: Wafer put ready position.
- P2: The station extension position obtained by adding the upward offset to the teaching position.
- P3: The position obtained by adding the downward offset to the teaching position.
- P4: Wafer put end position.
- P7: Intermediate position. (Optional)
- P8: Intermediate position. (Optional)

<Put operation sequence>
When a Put operation command is received from the host, the Put operation is performed in the following sequence.

(1) Confirm there is a wafer at the current position.
(2) Move to P1 (Put ready position).
(3) Move from P7 (if it exists) to P2.
(4) Turn the solenoid off, and confirm there is no wafer (pressure sensor off).
(5) Move to P3.
(6) Move to P4.

6.8 Master Registration (Teaching) Function

The Master Registration Function is a function designed to reduce the amount of teaching work, and by selecting Master Registration during teaching, this function will automatically generate the other arm posture and blade teaching positions.

Note that the teaching positions generated by master registration are calculated based on the dimensions of the machine.

Therefore, because small variations from the correct teaching positions may occur, it is necessary to confirm the teaching positions.

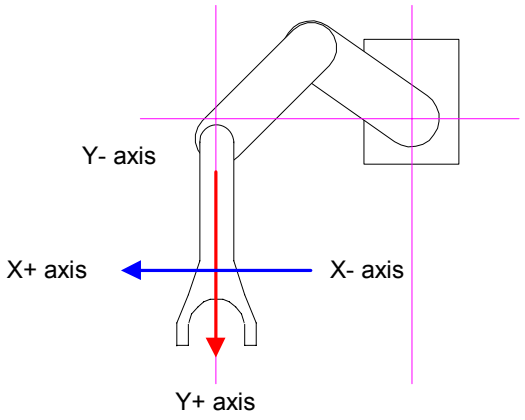
Table 6.2 Positions Registered Automatically by Master Registration

Registered blade/arm posture	Automatically generated position
Blade 1 with right elbow	Blade 1 left elbow teaching position (If accessible) Blade 2 right elbow teaching position Blade 2 left elbow teaching position (If accessible)
Blade 1 with left elbow	Blade 1 right elbow teaching position (If accessible) Blade 2 left elbow teaching position Blade 2 right elbow teaching position (If accessible)
Blade 2 with right elbow	Blade 2 left elbow teaching position (If accessible) Blade 1 right elbow teaching position Blade 1 left elbow teaching position (If accessible)
Blade 2 with left elbow	Blade 2 right elbow teaching position (If accessible) Blade 1 left elbow teaching position Blade 1 right elbow teaching position (If accessible)

6.9 Teaching Position Adjustment

It is possible to specify the correction values (X, Y, Z axes in Figure 6.9) in the transfer station teaching positions and then adjust the wafer get/put positions without changing the teaching positions. (The teaching positions are not changed.)

The commands for specifying correction values are the “MTRS” and “MCTR” commands.



- X+ axis: Left direction, X– axis: Right direction (*1 looking from the end of the blade)
- Y+ axis: Blade extension direction, Y– axis: Blade retraction direction
- Z+ axis: Up direction, Z– axis: Down direction

Figure 6.9 Teaching Position Correction Axes

6.10 Automatic Switching Function of Operation speed

Operation speed is set separately for action commands depending on wafer detection, wafer get/put area, etc.

The host can set these speeds to any value up to the maximum speed of each axis.
When the controller power is cycled, these speed setting is set back to the default setting (saved in parameter area).

- No-wafer-transfer-speed : Transfer speed when there is no wafer
- With-wafer-transfer-speed : Transfer speed when there is a wafer
- Low-speed-transfer-speed : The speed used for elevation during wafer get/put and operations returning to the reference position
- Home-return-speed : The speed when moving to the origin
- Low-speed-area-speed : Operation speed for the slow speed areas during wafer get/put

Changing the transfer speed is performed automatically in the controller according to the wafer grip state during operation (wafer presence detection through the vacuum type pressure sensor or wafer presence sensor) or the operation type.

Furthermore, for each speed there are 3 levels (speed level 1 to speed level 3), and the default for each level is set at the percentages set below.

The can set the speed level to use a the transfer speeds as appropriate.
When power is turned on, the speed level is initially level 1.

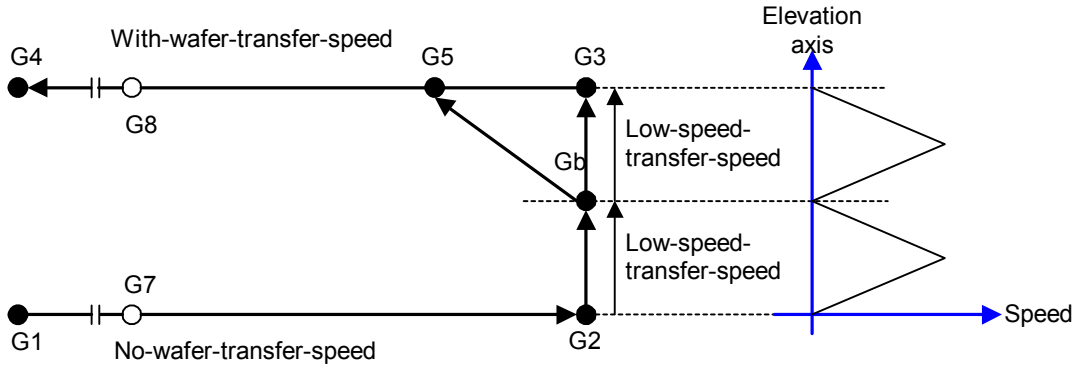
Refer to the parameter list for each speed.

6.10.1 Edge Grip Type Operation Speed

Figure 6.10 shows the speed for an edge grip type Get operation.

“Elevation low speed area offset” is a value that defines the area in the elevation operation of a Get/Put operation in which to operate at a constant speed, performing a wafer Get/Put at a constant low speed without slowing to a stop and therefore helping to improve throughput.

(a) Get operation speed in the cassette stage and PA stage



(b) Get operation speed in the transfer stage

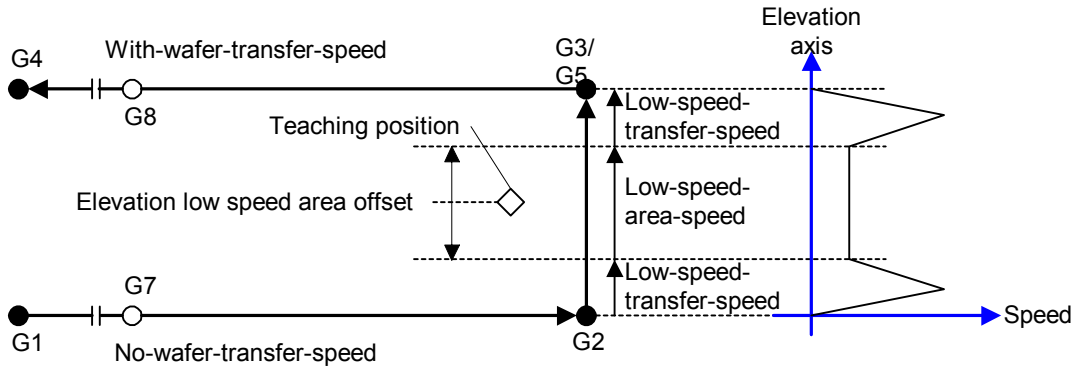


Figure 6.10 Edge Grip Type Get Operation Speed

shows the operation speed of an edge grip type Put operation

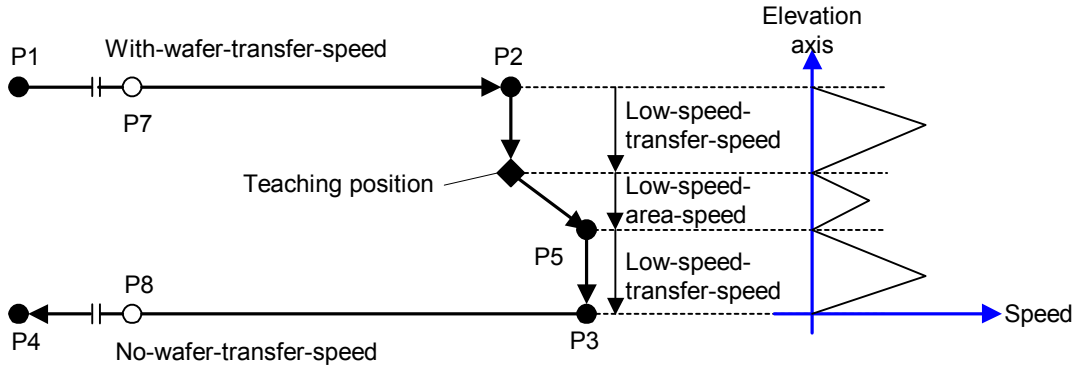
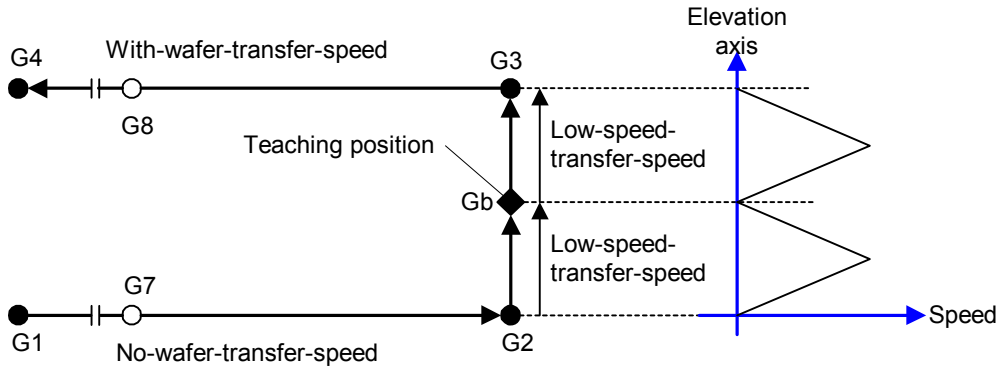


Figure 6.11 Edge Grip Type Put Operation Speed (all stages)

6.10.2 Vacuum Type Operation Speed

Figure 6.12 shows the speed for a vacuum type Get operation.

(a) Get operation speed in the cassette stage and PA stage



(b) Get operation speed in the transfer stage

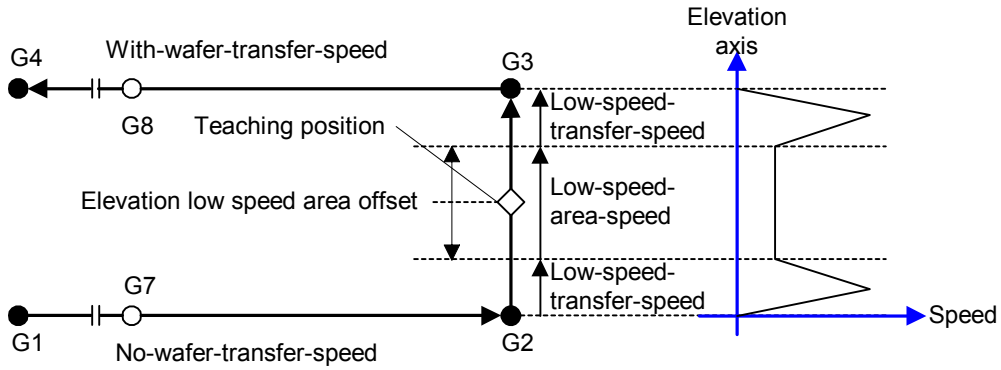


Figure 6.12 Vacuum Type Get Operation Speed

Figure 6.13 shows the speed for a vacuum type Put operation.

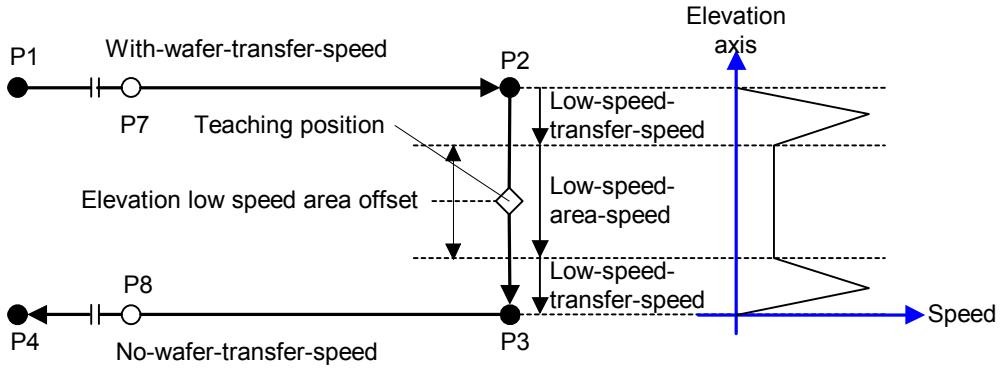


Figure 6.13 Vacuum Type Put Operation Speed (all stages)

7 Transfer Operation Expanded Functions

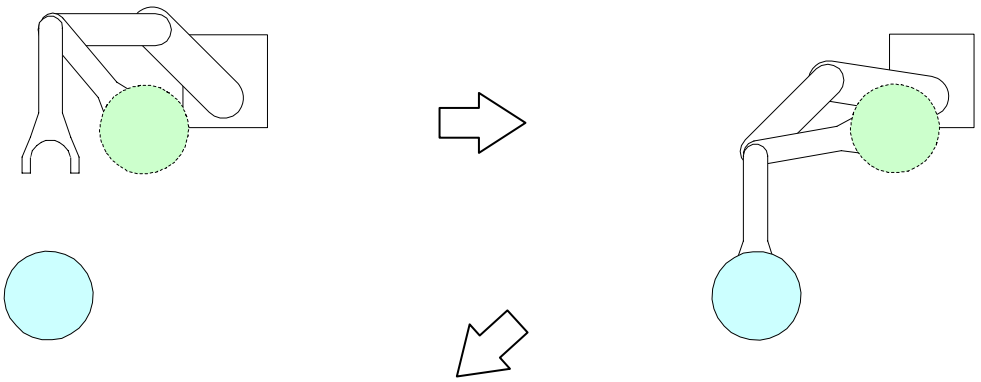
7.1 Wafer Exchange Operation

Wafer exchange operations (Get/Put, in that order) can be performed for all *transfer stages* with one operation command.

The wafer exchange operation movement order is shown in Figure 7.1 (In this case, a wafer is on Blade 1.)

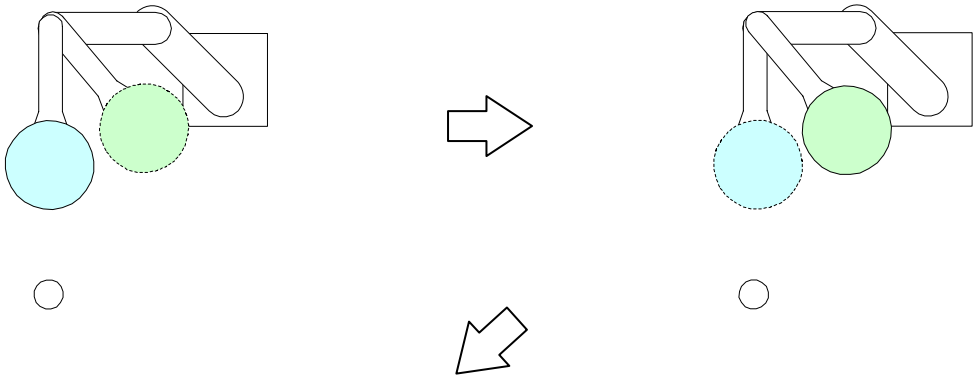
(1) Blade 2 moves to Get ready position.

(2) Blade 2 gets the wafer.



(3) Blade 2 moves to Get completion position.

(4) Blade 1 moves to Put ready position.
(Blade exchange)



(5) Blade 1 puts the wafer.

(6) Blade 1 moves to Put completion position.

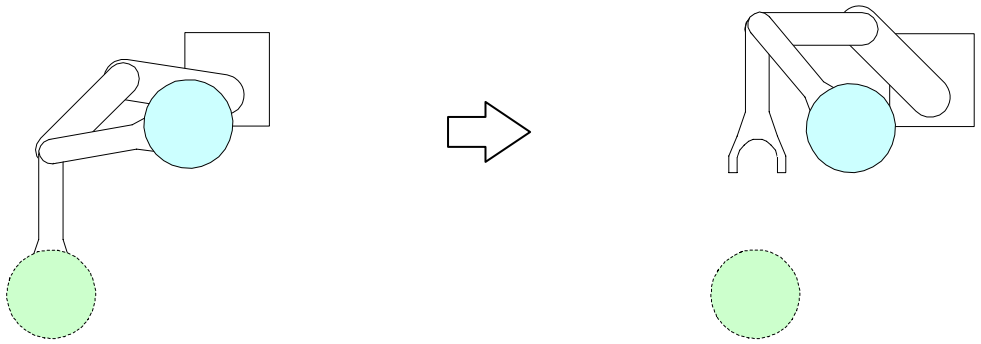


Figure 7.1 Wafer Exchange Operation

7.2 2-wafer Simultaneous Get Operation/Put Operation

It is possible to Get 2 wafers at the same time (WGet operation) or Put two wafers at the same time (WPut operation) for a given cassette stage.

By operating 2 blades together 2 wafers can be Get or Put in the same operation.

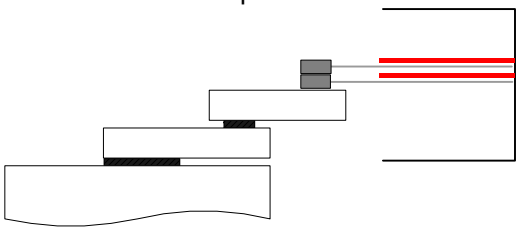
The WGet and WPut operations are intended for cassette stages with multiple slots, and as a condition both blades must be the same blade type.

With the blade teaching position based on Blade 1, Blade 2 uses Blade 1's teaching position.

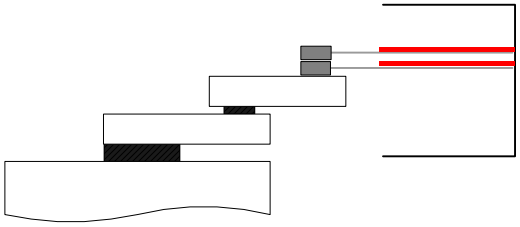
The WGet and WPut operation order is shown in Figure 7.2 and Figure 7.3.

(1) Both blades move to Get ready position

(2) Both blades move to position G2



(3) Both blades move to position G3



(4) Both blades move to position G4

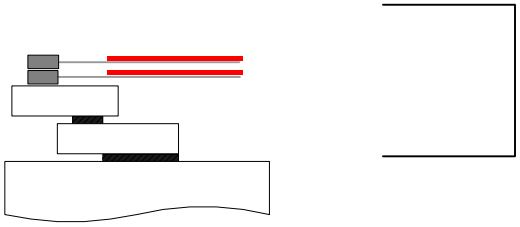
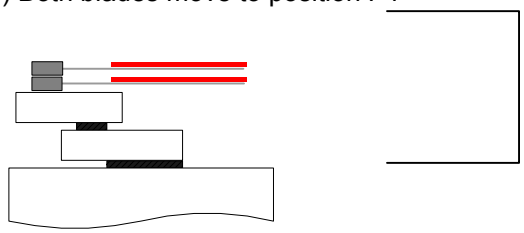


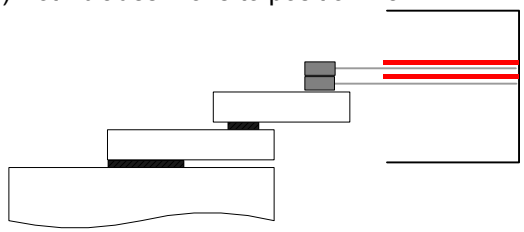
Figure 7.2 2-wafer Simultaneous Get Operation (WGet Operation)

(1) Both blades move to position P1



(2) Both blades move to position P2

(3) Both blades move to position P3



(4) Both blades move to position P4

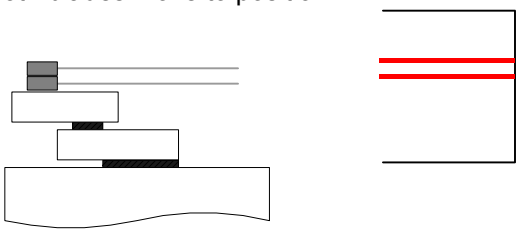


Figure 7.3 2-wafer Simultaneous Put Operation (WPut Operation)

7.3 Retry Operation

If a wafer's presence cannot be confirmed in a Get operation, a Retry operation can be performed. (By default, the Retry operation is disabled)

The retry operation differs for the edge grip type and vacuum type.

(1) Simple Retry Operation

- Edge grip type

  At position G5, if a wafer's presence cannot be confirmed, the solenoid is on at G5, and after the wait time, the solenoid is off again and wafer grip is confirmed.

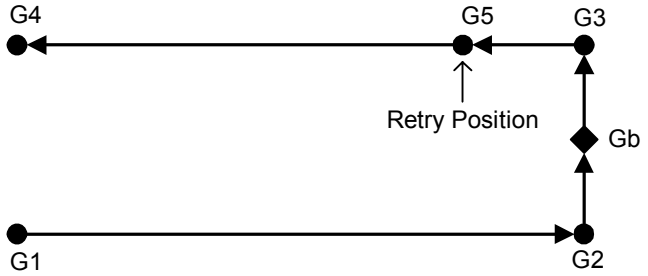


Figure 7.4 Simple Retry Operation (Edge grip type)

- Vacuum type

  At position G3, if a wafer's presence cannot be confirmed, the solenoid is off at G3, and after the wait time, the solenoid is on again and wafer grip is confirmed.

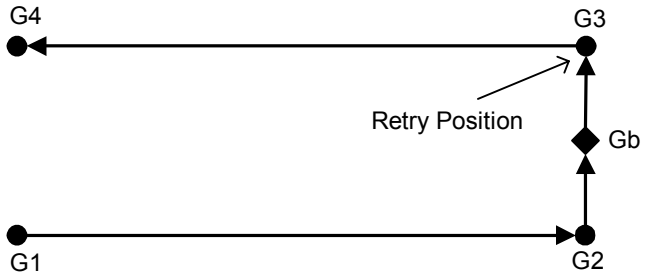


Figure 7.5 Simple Retry Operation (Vacuum type)

(2) Move Back Retry Operation

- Edge grip type

  Not supported.

- Vacuum type

  At position G3, if a wafer's presence cannot be confirmed, the solenoid is off at G3, and after the wait time, descends to G2 (passing through Gb), and the Get operation is performed again.

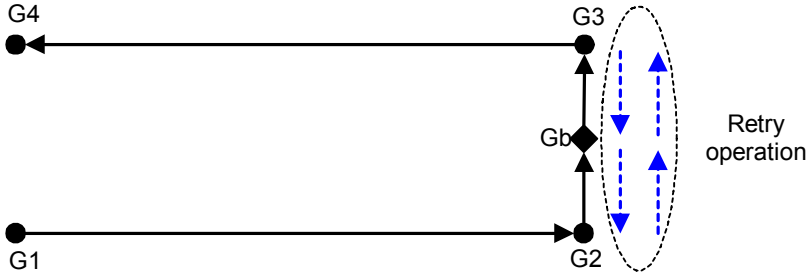


Figure 7.6 Move Back Retry Operation

8 Wafer Mapping Function

The wafer mapping function uses the mapping sensor (blade tip transmission-type sensor) to determine wafer insertion status in the cassette (which slots have wafers, whether insertion is correct, and if insertion is incorrect (inclined or double insertion), as well as wafer presence outside of the cassette stage.

Also, wafer protrusion detection, slot gap pitch detection, and teaching position automatic adjustment can also be performed.

8.1 Wafer Mapping Function Outline

The manipulator moves to the wafer cassette mapping start position of cassette stage, and the elevation axis is raised from that position, and sampling is taken of elevation axis positions where the mapping sensor turns on or off. Then, wafer insertion status (correct insertion, abnormal insertion) is determined by comparing that data to the wafer determination threshold value obtained in the calibration operation described below.

Figure 8.1 details the relationship between the mapping operation and the elevation axis sampling position according to wafer status.

As an expanded function of the mapping operation, wafer presence for certain cassette slots as well as for stations besides the wafer cassette can also be determined.

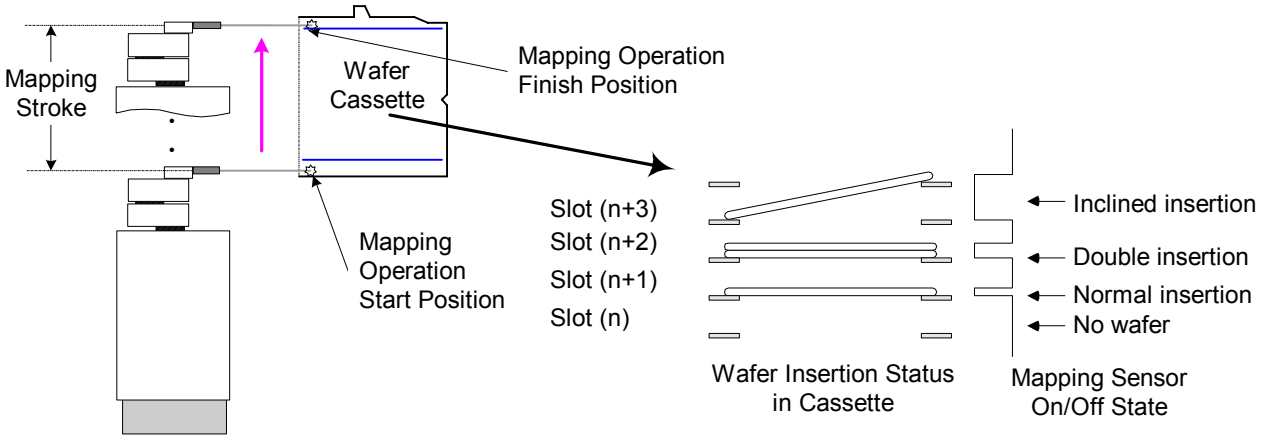


Figure 8.1 Mapping Operation and Wafer Status Detection

8.2 Wafer Protrusion Detection Function

In some rare cases the wafer may protrude towards the manipulator.

If a mapping operation is performed while a wafer is protruding, the wafer and blade will interfere with each other, and the wafer or blade may be damaged. This is why it is necessary to check for wafer protrusion.

This operation is performed using the wafer protrusion start position (a position defined with an offset value against the mapping operation start position) along with a distance and detection count (the count is set to 2 in Figure 8.2).

The wafer protrusion detection function can be enabled/disabled by a parameter selection for the mapping operation command.

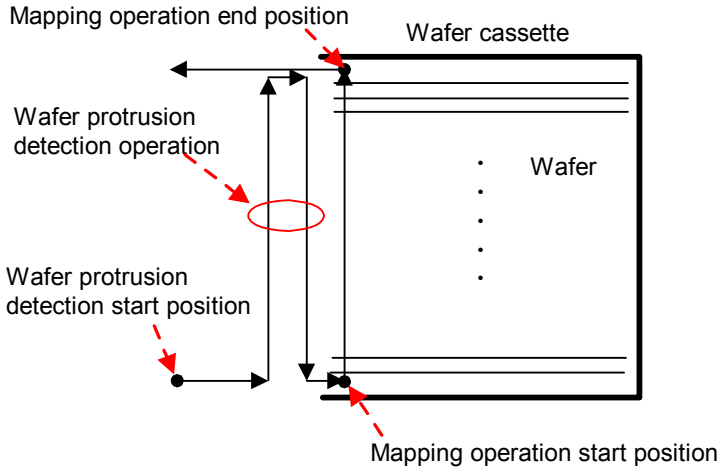


Figure 8.2 Wafer Protrusion Detection

8.3 Wafer Mapping Calibration Operation

The mapping function determines wafer presence and insertion errors using elevation axis detection positions sampled from the mapping operation combined with comparisons to wafer presence determination thresholds and irregular insertion determination thresholds set in advance.

The wafer mapping calibration operation is an operation to set wafer presence and insertion error determination thresholds, which are saved in the controller's parameters.

This operation is performed for the cassette stage only, and is not necessary for the transfer stage or the pre-aligner stage.

The mapping calibration operation is executed with wafers inserted in the highest-level and lowest-level slots.

The calibration operation is performed using the same motion as the mapping operation.

The mapping calibration operation must be performed when the robot is powered up, when cassette shapes are changed, when wafer properties (wafer size, wafer thickness, etc.) are changed, and when the teaching position is changed.

9 Wafer Alignment Function

The alignment function is for detecting the wafer center and matching up the notch (orientation flat) position. When the manipulator gets the wafer from the pre-aligner stage, the detected amount of wafer offset is compensated.

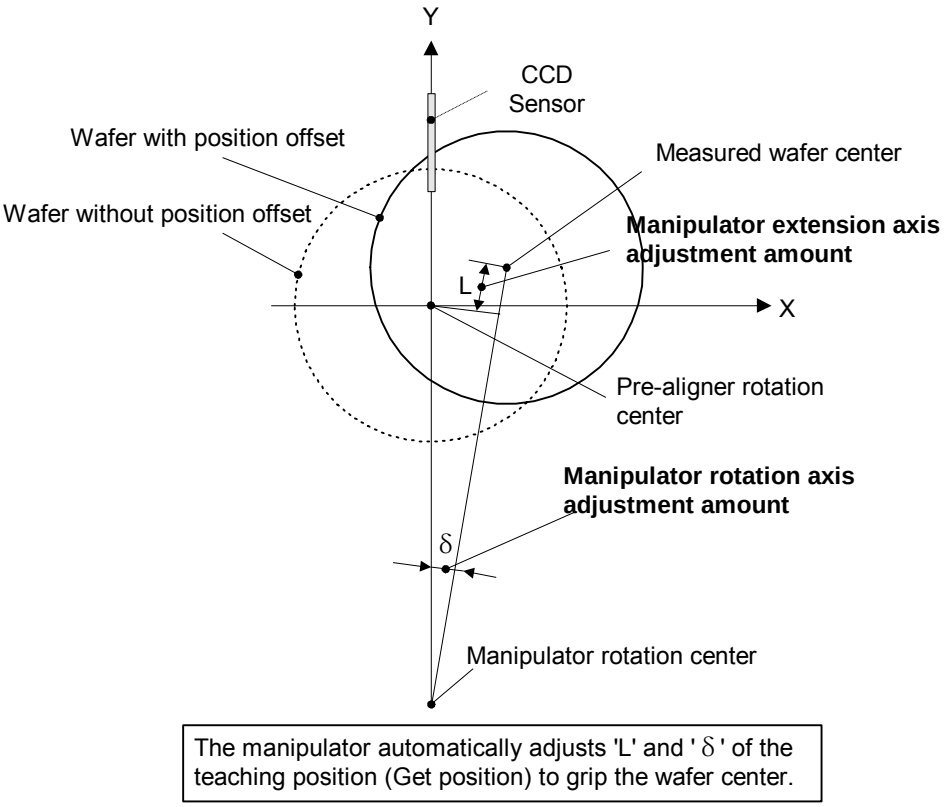


Figure 9.1 Alignment Adjustment Outline(When Pre-aligner Station Access is in Link Mode)

To perform the alignment operation, the two types of calibration detailed below must be performed in advance.

9.1 Arm Calibration

Arm calibration is an operation where a reference position is registered using the angle of the manipulator's advancement to the pre-aligner and the distance between Pre-aligner and the manipulator.

This operation must be performed when pre-aligner stage teaching is performed for the manipulator.

In its initialized state (without arm calibration having been executed), the CCD sensor direction will be at the reference position (0°).

After pre-aligner stage teaching is completed, the wafer is Put to the pre-aligner stage with the notch placed at the base of the blade as shown in Figure 9.2. When arm calibration is executed in this state, the reference position (0°) is updated to the position of the manipulator's advancement direction.

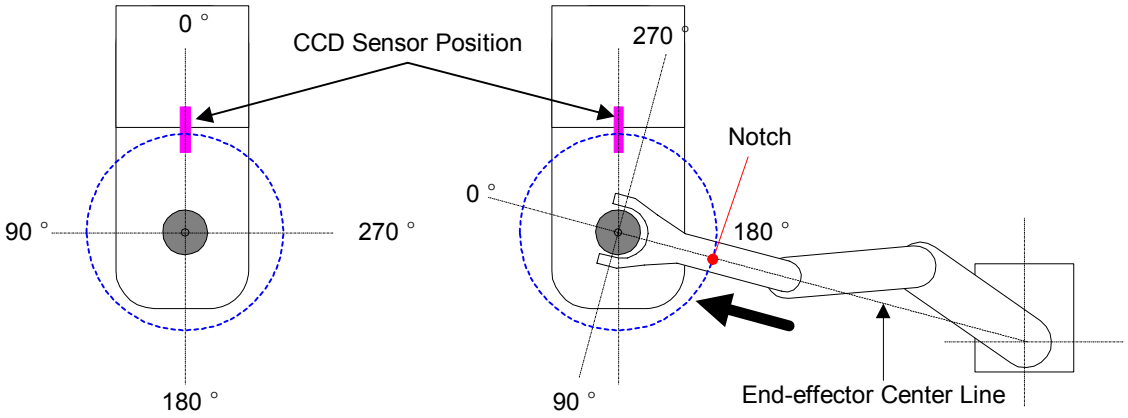


Figure 9.2 Arm Calibration

9.2 Notch Calibration

Notch calibration is an operation for registering the direction of the notch used for positioning during alignment.The notch direction is obtained in the sampling operation.

If notch calibration is not performed, in the alignment operation the notch position will be decided by the manipulator's advancement direction (direction from the blade base to the tip) set in arm calibration.

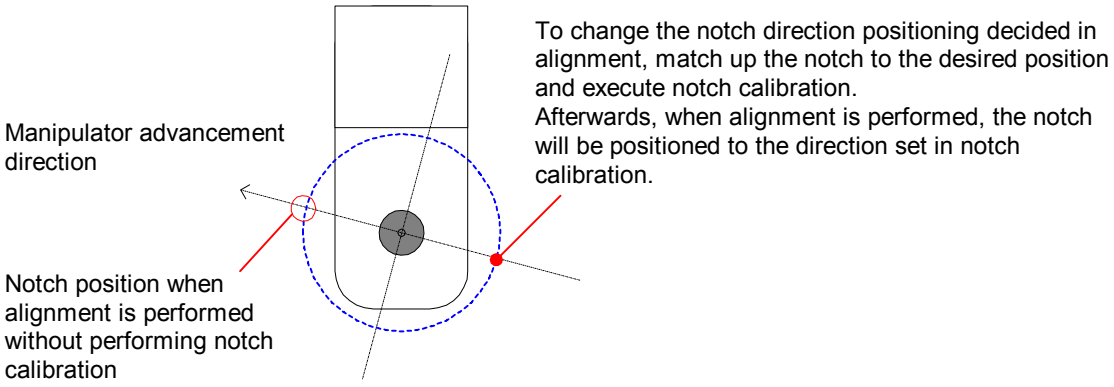


Figure 9.3 Notch Reference Position when Arm Calibration has not been Executed

10 Interlock Function

The manipulator has an interlock function to monitor various statuses to determine the validity of operations so that it can operate safely.

10.1 Wafer presence/absence status monitoring (manipulator and pre-aligner)

Controls operation depending on the presence or absence of a wafer on the blade or the pre-aligner.

<Manipulator>

- Wafer is present: Wafer get operation is prohibited. (Related commands: “MTRS”, “MCTR”)
  The mapping operation is prohibited. (Related commands: “MMAP”, “MMCA”)
- Wafer is absent:Wafe put operation is prohibited. (Related commands: “MTRS”, “MCTR”)

<Pre-aligner>

- Wafer is absent:Wafer alignment motion is prohibited. (Related command: “MALN”, “MACA”)

10.2 Pre-aligner status monitoring (manipulator)

The controller monitors the status of the pre-aligner and restricts manipulator motion for Get and Put operations to the pre-aligner.

(1) Pre-aligner motion monitoring:
When the pre-aligner is in motion (or executing motion or control command), manipulator access to the pre-aligner is prohibited.

(2) Pre-aligner wafer presence interlock 1:
If the pre-aligner is in the vacuum state, access operations to the pre-aligner stage are prohibited.

(3) Pre-aligner wafer presence interlock 2:
If a wafer is on the pre-aligner (determined by the CCD sensor), the Put operation to the pre-aligner stage is prohibited.

11 List of Commands

11.1 Action Commands

Name	Function	RB	PA
INIT	Initializes specified unit	√	√
MTRS	Performs wafer transfer operation (Get operations/Put operations/Exchange operations).	√	-
MPNT	Moves to the specified transfer point.	√	-
MCTR	Continued wafer transfer operation + Transfer operation.	√	-
MTCH	Moves to the specified position (registered position/ready position).	√	-
MABS	Moves the specified axis to a specified coordinate position.	√	-
MREL	Moves the specified axis to the specified relative position.	√	√
MMAP	Performs the wafer mapping.	√	-
MMCA	Performs the mapping calibration.	√	-
MALN	Aligns the wafer on the pre-aligner.	-	√
MACA	Performs alignment calibration.	-	√

11.2 Control Commands

Name	Function	RB	PA
CSTP	Applies deceleration/emergency stop to stop the motion.	√	√
CRSM	Restarts the motion interrupted by deceleration stop.	√	√
CSRV	Turns ON/OFF the servo power.	√	√
CCLR	Clears the current error or error history.	√	√
CSOL	Performs solenoid operation.	√	√

11.3 Setting Commands

Name	Function	RB	PA
SSPD	Sets the motion speed.	√	√
SSLV	Selects the transfer speed level.	√	√
SPOS	Registers the current position as the specified transfer station.	√	-
SABS	Registers the specified coordinate position as the specified transfer station.	√	-
SAPS	Modifies the specified transfer station's registered position by the adjustment offset.	√	-
SPDL	Deletes the specified transfer station's registered position.	√	-
SPSV	Registers the position data in the volatile memory to the non-volatile memory.	√	-
SPLD	Reads the position data in the non-volatile memory into the volatile memory.	√	-
SSTR	Sets the station's information parameters.	√	-
SPRM	Changes the parameter values.	√	√
SMSK	Enables or disables the interlock monitoring function.	√	√
SSTD	Registers the current position as the manipulator coordinate's reference position.	√	-
SSTN	Registers the specified number as a reference position.	√	-

11.4 Reference Commands

Name	Function	RB	PA
RSPD	References the motion speed.	√	√
RSLV	References the current transfer speed level.	√	√
RPOS	References the current position.	√	-
RSTP	References the registered position.	√	-
RSTR	References the station information.	√	-
RPRM	References the parameter value.	√	√
RSTS	References the units' statuses.	√	√
RERR	References the error history.	√	√
RMSK	References the current interlock monitor settings.	√	√
RVER	References the software version.	√	√
RMAP	References the specified transfer station's mapping results.	√	-
RMPD	References the mapping data (elevation axis coordinates during sensor edge startup/stopping).	√	-
RMCA	References the mapping caliblation result.	√	-
RALN	References the alignment result.	-	√
RACA	References the calibration results for alignment.	-	√
RCCD	References the pre-aligner's light amount and CCD data.	-	√
RLOG	References the log data.	√	√
RSTN	References the reference position record.	√	-

11.5 End-f-Execution Confirmation Notification Commands

Name	Function	RB	PA
ACKN	An acknowledgement from the host indicating the reception of an end-of-execution message from the controller.	√	√

12 Command Details

This is an explanation along with the common notations for message formats used for the command detail pages.

[Function]

Specifies the outline of each command's function.

[Conditions]

Specifies the conditions for executing each command.

[Command Message]

This describes the command message format used for each command. The underlined portion will differ for each command.

$,<UNo>(,<SeqNo>),<Command>,<Parameter1>…,<ParameterN>(,<Sum>)<CR>

- '$': Start mark (1 byte)
- ',' : Delimiter mark (1 byte)
- UNo: Unit number (1 byte)
- SeqNo: Sequence number (None / 2 bytes)
- Command: Command (4 bytes)
  - Differs according to the command.
- Parameter*: Parameter
  - The number, size, and contents of the parameter will differ according to the command.
- Sum: CheckSum (2 bytes / None)
- CR: End mark (1 byte)

[Response Message]

This describes the response message format used for each command. The underlined portion will differ for each command.

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,<Command>,<Parameter1>…,<ParameterN>,
<Value1>…,<ValueN>(,<Sum>)<CR>

- '$': Start mark (1 byte)
- ',': Delimiter mark (1 byte)
- UNo: Unit number (1 byte)
- SeqNo: Sequence number (None / 2 bytes)
- Sts: Status (2 bytes)
- Ackcd: Response code (4 bytes)
- Command: Command (4 bytes)
- Parameter*: Parameter
- Value*: Response data
  - The response data's number, size, and contents will differ according to the command.
- Sum: CheckSum (2 bytes / None)
- CR: End mark (1 byte)

[Message for the End-of-Execution]

Describes the end-of-execution message format for action commands and control commands. The underlined portion will differ for each command.

!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,<Command>,<ExeTime>,<PosData1>,…<PosDataN>
,<Value1>…,<ValueN>(,<Sum>)<CR>

- '!': Start mark (1 byte)
- ',': Delimiter mark (1 byte)
- UNo: Unit number (1 byte)
- SeqNo: Sequence number (None / 2 bytes)
- Sts: Status (2 bytes)
- Errcd: Error code (4 bytes)
- Command: Command (4 bytes)
- ExeTime: Execution time (6 bytes, Resolution: 1 [msec])
- PosData*: Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])
- Value*: Response data
  - The response data's number, size, and contents will differ according to the command.
- Sum: CheckSum (2 bytes / None)
- CR: End mark (1 byte)

[Usage Example]

Provides an example of the command's usage.

[Supplementary Explanation]

Provides additional related information.

INIT (Unit initialization)

[Function]

Initializes specified unit
Executes each function according to the specified initialization mode. However, only feasible processes for the specified unit can be executed.

- Error clear
- Servo ON
- Move to HOME Position

[Conditions]

- The specified unit is under ready state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),INIT,<ErrClr>,<SrvOn>,<Home>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None/2 bytes)
- ErrClr : Error clear Yes/No (1 byte)
  - '0' : No error clear.
  - '1' : Error clear.
- SrvOn : Servo ON Yes/No (1 byte)
  - '0' : No Servo ON.
  - '1' : Servo ON.
- Home : Axes that move to home position (1 byte)
  <Manipulator>
  - 'G' : All axes.
  - 'A' : Arm axes only.
  - 'N' : No axes move to home position.

  <Pre-aligner>
  - 'N' : Fixed value.
    (Vacuum type pre-aligners do not need to move to the home position)

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,INIT,<ErrClr>,<SrvOn>,<Axis>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- ErrClr : Error clear Yes/No (1 byte)
- SrvOn : Servo ON Yes/No (1 byte)
- Axis : Home-specified axes (1 byte)

[Message for the End-of-Execution]

!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,INIT,<ExeTime>,<PosData1>…,<PosDataN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])

[Usage Example]

(1) Error clear, servo ON and all axes move to home position:
$,1,INIT,1,1,G

(2) Servo ON and arm moves to home position:
$,1,INIT,0,1,A

MTRS (Perform wafer transfer operation (Get/Put/Exchange operations))

[Function]

Perform wafer transfer (Get/Put/Exchange) operations for the specified transfer station.
Each axis moves in the following order.

(1) Check the wafer presence.
(2) Moves the arm(s) to the minimum sweep posture through a safe path.
(3) Moves to the ready position with specified adjustment offset.
(4) Performs a wafer transfer operation (Get/Put/Exchange) with the specified adjustment offset

As for the operation sequence, refer to “(6 Transfer Point Generation and Motion pass”.
Note)

- If the alignment angle has been specified using a pre-aligner stage Put operation pre-aligner alignment (sampling operation + adjustment operation) is automatically performed.
- If alignment has already been performed at the time of a pre-aligner stage Get operation, pre-aligner adjustment and arm adjustment operations will be performed. At that time, even if an adjustment offset for XYZ directions has been input, the XYZ direction adjustment offset will not be applied.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.’s mode selector switch is set to Host mode (if T.P. is connected).
- The specified transfer station has been registered.

[Command Message]

$,<UNo>(,<SeqNo>),MTRS,<Mtn>,<TrsSt>,<Slot>,<Posture>,<Hand>,<TrsPnt>
(,<OfstX>,<OfstY>,<OfstZ>)(,<Angle>)(,<Sum>)<CR>

- UNo: Unit number (1 byte)
  - ‘1’ : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- Mtn : Motion mode (1 byte)
  - ‘G’ : Get motion.
  - ‘P’ : Put motion.
  - ‘E’ : Exchange motion.
- TrsSt : Transfer station (3 bytes)
  - “C01” - “C08” : Cassette stage.
  - “S01” - “S12” : Transfer stage.
  - “P01” : P/A stage.
- Slot : Slot number (2 bytes)
  <Cassette stage>
  - “01” - “30” : When cassette stage specified.

  <Transfer stage, Pre-aligner stage>
  - “00” : Fixed slot(because this type of station does not have multiple slots.).

  Note) If value is less than 2 digits, fill the higher digit with ‘0’ so that the field always has 2 digits.
  Note) If <Hand> is ‘F’(Blade 1 + Blade 2), specifies the slot accessed by Blade 1.
- Posture : Arm Posture (1 byte)
  - ‘L’ : Left elbow.
  - ‘R’ : Right elbow.
  - ‘A’ : Automatic (Automatically selected posture with the proper path).
- Hand : Blade (1 byte)
  - ‘1’ : Blade 1.
  - ‘2’ : Blade 2.
  - ‘F’ : Blade 1 + Blade 2 (WGet/WPut operation).

  Note) Except for <TrsPnt> is [C01-C08: Cassette stage], ‘F’(Blade 1 + Blade 2) cannot be specified.
  Note) If <Mtn> is ‘E’(Exchange motion), ‘F’(Blade 1 + Blade 2) cannot be specified.

- TrsPnt : Transfer point (2 bytes)
  - “G1” - “G8”, “Gb” : Get transfer point.
  - “P1” - “P8”, “Pb” : Put transfer point.

  Note) If <Mtn> is ‘G’(Get motion),”P1”-“P8”,”Pb”(Put transfer point) cannot be specified.

  Note) If <Mtn> is ‘P’(Put motion), “G1”-“G8”, “Gb”(Get transfer point) cannot be specified.
- Ofst* : XYZ direction offset (None / 8 bytes each, Resolution: 0.001 [mm])
  - OfstX : X direction offset.
  - OfstY : Y direction offset .
  - OfstZ : Z direction offset.

  Note) Specified in the range between “-0009999” and “00009999”.
  If a value is less than the specified digits, fill the higher digit(s) with ‘0’ so that the field always has specified digits.
  A sign is added to the highest digit.

  Note) Ignored during Exchange operations.

  Note) For the XYZ direction adjustment offset, <OfstX>,<OfstY>,<OfstZ> can be omitted together.
- Angle : Positioning angle (None / 8 bytes, Resolution: 0.001 [deg])

  Note) Relative angle from the position set by alignment calibration.

  Note) Specified in the range between “00000000” and “00359999”.
  If a value is less than the specified digits, fill the higher digit(s) with ‘0’ so that the field always has specified digits.

  Note) Only effective for a Put motion to the pre-aligner stage (Ignored otherwise).
  If omitted for a Put motion to the pre-aligner stage, the alignment operation will not be performed.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,MTRS,<Mtn>,<TrsSt>,<Slot>,<Posture>,<Hand>,<TrsPnt>
(,<OfstX>,<OfstY>,<OfstZ>)(,<Angle>)(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mtn : Motion mode (1 byte)
- TrsSt : Transfer station (3 bytes)
- Slot : Slot number (2 bytes)
- Posture : Arm Posture (1 byte)
- Hand : Blade (1 byte)
- TrsPnt : Transfer point (2 bytes)
- Ofst* : XYZ direction offset (None / 8 bytes each, Resolution: 0.001 [mm])
- Angle : Positioning angle (None / 8 bytes, Resolution: 0.001 [deg])

[Message for the End-of-Execution]

!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,MTRS,<ExeTime>,<PosData1>…,<PosDataN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])

[Usage Example]

(1) Moving cassette stage 2, slot 5 to the Get ready position (G1).
(No adjustment offset for Left elbow, Blade 1, or XYZ directions)
$,1,MTRS,G,C02,05,L,1,G1 : XYZ direction adjustment offset omitted, Positioning angle omitted
$,1,MTRS,G,C02,05,L,1,G1,00000000 : XYZ direction adjustment offset omitted
$,1,MTRS,G,C02,05,L,1,G1,00000000,00000000,00000000 : Positioning angle omitted
$,1,MTRS,G,C02,05,L,1,G1,00000000,00000000,00000000,00000000 : Nothing omitted

(2) Deliver a wafer to the pre-aligner stage, and perform alignment at an alignment angle of 90.0 degrees.(No adjustment offset for Right elbow, Blade 2, or XYZ directions)
$,1,MTRS,P,P01,00,R,2,P4,00090000 : XYZ direction adjustment offset omitted
$,1,MTRS,P,P01,00,R,2,P4,00000000,00000000,00000000,00090000 : Nothing omitted

(3) Perform a Get motion of transfer stage 1.
(Left elbow, Blade 2, X direction offset = 0.25 [mm], Y direction offset = -0.3 [mm], no Z direction offset)
$,1,MTRS,G,S01,00,L,2,G4,00000250,-0000300,00000000 : Positioning angle omitted
$,1,MTRS,G,S01,00,L,2,G4,00000250,-0000300,00000000,00000000 : Nothing omitted

MPNT (Motion between Transfer Points)

[Function]

Moves to the transfer point of the transfer station specified by “MTRS” or “MCTR” command executed immediately before this command.
As for the motion sequence, refer to “(6 Transfer Point Generation and Motion pass”.
Note)
If a pre-aligner stage wafer Get operation is performed after an alignment operation, the initial Get operation will be (G2→Gb→G3) only.
The alignment adjustment result is reflected.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.’s mode selector switch is set to Host mode (if T.P. is connected).
- The action command immediately before this command has to be successfully completed “MTRS” command or “MCTR command”/“MPNT command”.

[Command Message]

$,<UNo>(,<SeqNo>),MPNT,<TrsPnt>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - ‘1’ : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- TrsPnt : Transfer point (2 bytes)
  - “G1” - “G8”, “Gb” : Get transfer point.
  - “P1” - “P8”, “Pb” : Put transfer point.
  - “AL” : Final point (For Get motion: G4, for Put/Exg motion: P4).
  - “ST” : Step operation (Move to next transfer point).

  Note) If <Mtn> is ‘G’(Get motion), “P1”-“P8”,”Pb”(Put transfer point) cannot be specified.
  Note) If <Mtn> is ‘P’(Put motion), “G1”-“G8”,”Gb”(Get transfer point) cannot be specified.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,MPNT,<TrsPnt>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- TrsPnt : Transfer point (2 bytes)

[Message for the End-of-Execution]

!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,MPNT,<ExeTime>,<PosData1>…,<PosDataN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])

[Usage Example]

In the state where the Get ready position (G1) for cassette stage 3, slot 20 has already been reached,continue and move to the wafer Get operation transfer point (G3).
(No offset for Right elbow, Blade 2, or XYZ directions)

$,1,MTRS,G,C03,20,R,2,G1 : Move to Get Ready position (G1)
$,1,MPNT,G3 : Move to G3 point

[Supplementary Explanation]

[Vacuum-type transfer point specification]

The following tables show available destination point using the “Motion between Transfer Points” command (MPNT command).
The logic of the vacuum is controlled according to ” 6-7-2 Vacuum Type Transfer Points”.
(Even in a reversed path, the vacuum is controlled.)

1) When a Get operation is specified by “MTRS” command and “MCTR” command:

Current Position \ Destination	G1	G7	G2	Gb	G3	G8	G4
G1	G1→G1	G1→G7	G1→G7→G2	G1→G7→G2→Gb	G1→G7→G2→Gb→G3	G1→G7→G2→Gb→G3→G8	G1→G7→G2→Gb→G3→G8→G4
G7	G7→G1	G7→G7	G7→G2	G7→G2→Gb	G7→G2→Gb→G3	G7→G2→Gb→G3→G8	G7→G2→Gb→G3→G8→G4
G2	G2→G7→G1	G2→G7	G2→G2	G2→Gb	G2→Gb→G3	G2→Gb→G3→G8	G2→Gb→G3→G8→G4
Gb	Unacceptable	Unacceptable	Gb→G2	Gb→Gb	Gb→G3	Gb→G3→G8	Gb→G3→G8→G4
G3	Unacceptable	Unacceptable	G3→Gb→G2	G3→Gb	G3→G3	G3→G8	G3→G8→G4
G8	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	G8→G8	G8→G4
G4	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	G4→G4

2) When a Put motion is specified by “MTRS” command and “MCTR” command:

Current Position \ Destination	P1	P7	P2	Pb	P3	P8	P4
P1	P1→P1	P1→P7	P1→P7→P2	P1→P7→P2→Pb	P1→P7→P2→Pb→P3	P1→P7→P2→Pb→P3→P8	P1→P7→P2→Pb→P3→P8→P4
P7	P7→P1	P7→P7	P7→P2	P7→P2→Pb	P7→P2→Pb→P3	P7→P2→Pb→P3→P8	P7→P2→Pb→P3→P8→P4
P2	P2→P7→P1	P2→P7	P2→P2	P2→Pb	P2→Pb→P3	P2→Pb→P3→P8	P2→Pb→P3→P8→P4
Pb	Unacceptable	Unacceptable	Pb→P2	Pb→Pb	Pb→P3	Pb→P8	Pb→P3→P8→P4
P3	Unacceptable	Unacceptable	P3→Pb→P2	P3→Pb	P3→P3	P3→P8	P3→P8→P4
P8	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	P8→P8	P8→P4
P4	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	P4→P4

3) When wafer exchange motion is specified by “MTRS” command and ”MCTR” command, “MPNT” command performs a motion combining 1) and 2) above:

G1→G7→G2→Gb→G3→G8→G4→P1→P7→P2→Pb→P3→P8→P4

[Transfer point of Edge grip type]

Following table shows available destination point by “Motion between Transfer Points” command (MPNT command). The grip’s logic follows “6.10.1 Edge Grip Type Operation Speed” too.
(Even in a reversed path, grip is controlled)

1) When a Get motion is specified by ”MTRS” command and ”MCTR” command:

Current Position \ Destination	G1	G7	G2	Gb	G3	G5	G8	G4
G1	G1→G1	G1→G7	G1→G7→G2	G1→G7→G2→Gb	G1→G7→G2→Gb→G3	G1→G7→G2→Gb→G3→G5	G1→G7→G2→Gb→G3→G5→G8	G1→G7→G2→Gb→G3→G5→G8→G4
G7	G7→G1	G7→G7	G7→G2	G7→G2→Gb	G7→G2→Gb→G3	G7→G2→Gb→G3→G5	G7→G2→Gb→G3→G5→G8	G7→G2→Gb→G3→G5→G8→G4
G2	G2→G7→G1	G2→G7	G2→G2	G2→Gb	G2→Gb→G3	G2→Gb→G3→G5	G2→Gb→G3→G5→G8	G2→Gb→G3→G5→G8→G4
Gb	Unacceptable	Unacceptable	Gb→G2	Gb→Gb	Gb→G3	Gb→G3→G5	Gb→G3→G5→G8	Gb→G3→G5→G8→G4
G3	Unacceptable	Unacceptable	G3→Gb→G2	G3→Gb	G3→G3	G3→G5	G3→G5→G8	G3→G5→G8→G4
G5	Unacceptable	Unacceptable	Unacceptable	Unacceptable	G5→G3	G5→G5	G5→G8	G5→G8→G4
G8	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	G8→G8	G8→G4
G4	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	G4→G4

2) When a Put motion is specified by ”MTRS” command and ”MCTR” command:

Current Position \ Destination	P1	P7	P2	Pb	P5	P3	P8	P4
P1	P1→P1	P1→P7	P1→P7→P2	P1→P7→P2→Pb	P1→P7→P2→Pb→P5	P1→P7→P2→Pb→P5→P3	P1→P7→P2→Pb→P5→P3→P8	P1→P7→P2→Pb→P5→P3→P8→P4
P7	P7→P1	P7→P7	P7→P2	P7→P2→Pb	P7→P2→Pb→P5	P7→P2→Pb→P5→P3	P7→P2→Pb→P5→P3→P8	P7→P2→Pb→P5→P3→P8→P4
P2	P2→P7→P1	P2→P7	P2→P2	P2→Pb	P2→Pb→P5	P2→Pb→P5→P3	P2→Pb→P5→P3→P8	P2→Pb→P5→P3→P8→P4
Pb	Unacceptable	Unacceptable	Pb→P2	Pb→Pb	Pb→P5	Pb→P5→P3	Pb→P5→P3→P8	Pb→P5→P3→P8→P4
P5	Unacceptable	Unacceptable	Unacceptable	P5→Pb	P5→P5	P5→P3	P5→P3→P8	P5→P3→P8→P4
P3	Unacceptable	Unacceptable	P3→Pb→P2(*1)	P3→P5→Pb	P3→P5	P3→P3	P3→P8	P3→P8→P4
P8	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	P8→P8	P8→P4
P4	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	Unacceptable	P4→P4

(*1) “Unacceptable” when P5 exists.

3) In case of that wafer exchange motion is specified by “MTRS” command and “MCTR” command, “MPNT” command performs the motion combined 1) and 2) above:

G1→G7→G2→Gb→G3→G5→G8→G4→P1→P7→P2→Pb→P5→P3→P8→P4

MCTR (Continued wafer transfer operation + Transfer operation)

[Function]

Immediately following a “MTRS” or “MCTR” command, the specified wafer transfer operation is continued, and after the operation completes, a newly specified station’s wafer transfer operation is performed.
Each axis moves in the following order.

(1) Continues the transfer operation.
(2) Checks the wafer status.
(3) Moves the arm(s) to the minimum sweep posture through a safe path.
(4) Moves to the ready position with specified adjustment offset.
(5) Performs a wafer transfer operation (Get/Put/Exchange) with the adjustment offset applied.

As for the motion sequence, refer to “(6 Transfer Point Generation and Motion pass”.
Note)

- If the alignment angle has been specified for the pre-aligner stage Put motion, pre-aligner alignment (sampling operation + adjustment operation) is automatically performed.
- If alignment has already been performed at the time of a pre-aligner stage Get operation, pre-aligner adjustment and arm adjustment operations will be performed.
  At that time, even if an adjustment offset for XYZ directions have been input, the XYZ direction adjustment offset will not be applied.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.’s mode selector switch is set to Host mode (if T.P. is connected).
- The specified transfer station has been registered.
- The action command immediately before this command has to be a successfully completed “MTRS”, “MCTR”, or “MPNT” command.

[Command Message]

$,<UNo>(,<SeqNo>),MCTR,<Mtn>,<TrsSt>,<Slot>,<Posture>,<Hand>,<TrsPnt>
(,<OfstX>,<OfstY>,<OfstZ>)(,<Angle>)(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - ‘1’ : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- Mtn : Motion mode (1 byte)
  - ‘G’ : Get motion.
  - ‘P’ : Put motion.
  - ‘E’ : Exchange motion.
- TrsSt : Transfer station (3 bytes)
  - “C01” - “C08”: Cassette stage.
  - “S01” - “S12”: Transfer stage.
  - “P01” : P/A stage.
- Slot : Slot number (2 bytes)
  <Cassette stage>
  - “01” - “30” : When cassette stage specified.

  <Transfer stage, Pre-aligner stage>
  - “00” : Fixed slot(because this type of station does not have multiple slots.)

  Note) If value is less than 2 digits, fill the higher digit with ‘0’ so that the field always has 2 digits.
  Note) If <Hand> is ‘F’(Blade 1 + Blade 2), specifies the slot accessed by Blade 1.
- Posture : Arm Posture (1 byte)
  - ‘L’ : Left elbow.
  - ‘R’ : Right elbow.
  - ‘A’ : Automatic (Automatically selected posture with the proper path).

- Hand : Blade (1 byte)
  - '1' : Blade 1.
  - '2' : Blade 2.
  - 'F' : Blade 1 + Blade 2 (WGet/WPut motion).

  Note) Except for <TrsPnt> is [C01-C08: Cassette stage], 'F'(Blade 1 + Blade 2) cannot be specified.

  Note) If <Mtn> is 'E'(Exchange motion), 'F'(Blade 1 + Blade 2) cannot be specified.
- TrsPnt : Transfer point (2 bytes)
  - "G1" - "G8", "Gb" : Get transfer point.
  - "P1" - "P8", "Pb" : Put transfer point.

  Note) If <Mtn> is 'G'(Get motion),"P1"-"P8","Pb"(Put transfer point) cannot be specified.

  Note) If <Mtn> is 'P'(Put motion), "G1"-"G8", "Gb"(Get transfer point) cannot be specified.
- Ofst* : XYZ direction offset (None / 8 bytes each, Resolution: 0.001 [mm])
  - OfstX : X direction offset.
  - OfstY : Y direction offset.
  - OfstZ : Z direction offset.

  Note) Specified in the range between "-0009999" and "00009999".
  If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.
  A sign is added to the highest digit.

  Note) Ignored during Exchange operations.

  Note) For the XYZ direction adjustment offset, <OfstX>,<OfstY>,<OfstZ> can be omitted together.
- Angle : Positioning angle (None / 8 bytes, Resolution: 0.001 [deg])

  Note) Relative angle from the position set by alignment calibration as the reference point.

  Note) Specified in the range between "00000000" and "00359999".
  If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.

  Note) Only effective for a Put motion to the pre-aligner stage (Ignored otherwise).
  If omitted for a Put motion to the pre-aligner stage, the alignment operation will not be performed.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,MCTR,<Mtn>,<TrsSt>,<Slot>,<Posture>,<Hand>,<TrsPnt>
(,<OfstX>,<OfstY>,<OfstZ>)(,<Angle>)(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mtn : Motion mode (1 byte)
- TrsSt : Transfer station (3 bytes)
- Slot : Slot number (2 bytes)
- Posture : Arm Posture (1 byte)
- Hand : Blade (1 byte)
- TrsPnt : Transfer point (2 bytes)
- Ofst* : XYZ direction offset (None / 8 bytes each, Resolution: 0.001 [mm])
- Angle : Positioning angle (None / 8 bytes, Resolution: 0.001 [deg])

[Message for the End-of-Execution]

!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,MCTR,<ExeTime>,<PosData1>…,<PosDataN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])

[Usage Example]

(1) Move to the Get ready position (G1) of slot 25 of cassette stage 1 in advance using the “MTRS” command. (Left elbow, Blade 1, XYZ direction offsets omitted, Positioning angle omitted)
Next, use the “MCTR” command to continue from a Get operation for cassette stage 1 to perform a Put operation to the pre-aligner stage.(Left elbow, Blade 1, XYZ direction offset omitted, Positioning angle =90.0 [deg])
$,1,MTRS,G,C01,25,L,1,G1
$,1,MCTR,P,P01,00,L,1,P4,00090000

(2) Move to the Get ready position (G1) of the pre-aligner stage in advance using the “MTRS” command. (Left elbow, Blade 1, XYZ direction offsets omitted, Positioning angle omitted)
Continue the Get operation using the “MCTR” command to move to the P2 point of transfer stage 3. (Right elbow, Blade 1, XYZ direction offsets omitted, Positioning angle omitted)
$,1,MTRS,G,P01,00,L,1,G1
$,1,MCTR,P,S03,00,R,1,P2

MTCH (Move to Registered Position)

[Function]

Moves to the specified position for the specified transfer station.
Operations are performed in the following order.
(1) Moves the arm(s) to the minimum sweep posture through a safe path.
(2) Moves the arm(s) to the ready position. (For elevation axes, the position will have the Z direction adjustment offset applied.)
(3) Moves to the specified position (intermediate position,ready position,offset position,registered position,mapping start/mapping finish position).
(3)-1 Moves the arm to the specified position with the adjustment offsets in the XY directions applied.
(3)-2 Moves the arm and the elevation axis to their specified positions at the same time.
If the manipulator grip the wafer when motion start, the controller monitor the wafer status while moving.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).
- The specified transfer station has been registered.

[Command Message]

$,<UNo>(,<SeqNo>),MTCH,<TrsSt>,<Slot>,<Posture>,<Hand>,<PMode>
(,<OfstX>,<OfstY>,<OfstZ>)(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- TrsSt : Transfer station (3 bytes)
  - "C01" - "C08": Cassette stage (the lowest-layer position).
  - "H01" - "H08" : Cassette stage (the highest-layer position).
  - "S01" - "S12" : Transfer stage.
  - "P01" : P/A stage.
- Slot: Slot number (2 bytes)
  <Cassette stage>
  - "01" - "30" : If <TrsSt> is "C01"-"C08"(cassette stage), set the slot.
  - "00" : If <TrsSt> is "H01"-"H08"(cassette stage), recognize the highest-layer position.

  <Transfer stage, Pre-aligner stage>
  - "00" : Fixed slot(because this type of station does not have multiple slots.)

  Note) If value is less than 2 digits, fill the higher digit with '0' so that the field always has 2 digits.
  Note) If <Hand> is 'F'(Blade 1 + Blade 2), specifies the slot accessed by Blade 1.
- Posture : Arm Posture (1 byte)
  - 'L' : Left elbow.
  - 'R' : Right elbow.
  - 'A' : Automatic (Automatically selected posture with the proper path).
- Hand : Blade (1 byte)
  - '1' : Blade 1.
  - '2' : Blade 2.
  - 'F' : Blade 1 + Blade 2 (WGet/WPut motion).

  Note) Except for the <TrsSt> is "C01"-"C08"(cassette stage), 'F'(Blade 1 + Blade 2) cannot be specified.

- PMode : Position mode (1 byte)
  - ‘M’ : Intermediate position (position with XYZ direction offset value applied).
  - ‘R’ : Ready position (position with XYZ direction offset value applied).
  - ‘O’ : Offset position (position with XYZ direction offset values applied).
  - ‘S’ : Registered position.
  - ‘B’ : Mapping start position.
  - ‘E’ : Mapping finish position.

  Note) For mapping start/finish positions, slot number and blade specifications are ignored. Blades with mapping sensors equipped will operate.

  Note) The mapping start/finish position is shown in “Figure 6.3 Positions Related to Mapping Operation”
- Ofst* : XYZ direction offset value (None / 8 bytes each, Resolution: 0.001 [mm])
  - OfstX : X direction offset.
  - OfstY : Y direction offset.
  - OfstZ : Z direction offset.

  Note) Specified in the range between “-0009999” and “00009999”. If value is less than 8 digits, fill the higher digit(s) with ‘0’ so that the field always has 8 digits. A sign is added to the highest digit.

  Note) XYZ direction adjustment offsets <OfstX>,<OfstY>,<OfstZ> can be omitted together.

  Note) Ofst* values that are intended to avoid causing interference with the device or the wafer when accessing a transfer station will not have adjustment offsets applied.

  Note) the XYZ direction adjustment offset will not be applied in the registered position.

  Note) When mapping start/finish positions are specified, XYZ direction offset values are ignored.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,MTCH,<TrsSt>,<Slot>,<Posture>,<Hand>,<PMode>(,<OfstX>,<OfstY>,<OfstZ>)(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- TrsSt : Transfer station (3 bytes)
- Slot : Slot number (2 bytes)
- Posture : Arm Posture (1 byte)
- Hand : Blade (1 byte)
- PMode : Position mode (1 byte)
- Ofst* : XYZ direction offset (None / 8 bytes each, Resolution: 0.001 [mm])

[Message for the End-of-Execution]

!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,MTCH,<ExeTime>,<PosData1>…,<PosDataN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])

[Usage Example]

(1) Move to the ready position for slot 1 of cassette stage 4.
(No offset values for Right elbow, Blade 1, XYZ directions)
$,1,MTCH,C04,01,R,1,R,00000000,00000000,00000000 : XYZ direction offset values included
$,1,MTCH,C04,01,R,1,R : XYZ direction offset values omitted
$,1,MTCH,C04,00,R,1,R,00000000,00000000,00000000 : lowest-layer slot = move to slot 1

(2) Move to the offset position of the pre-aligner stage. (Left elbow, Blade 2)
Extend the blade at a height 4.0 [mm] higher than the registered position.
$,1,MTCH,P01,00,L,2,O,00000000,00000000,00004000

(3) Move to the transfer stage 2 registered position. (Left elbow, Blade 1)
Extend the blade 2.0 [mm] beyond the registered position at a height 3.5 [mm] lower than the registered position.
Afterwards, move the arm and elevation axis to the registered position at the same time.
$,1,MTCH,S02,00,L,1,S,00000000,00002000,-0003500

(4) Move to the mapping start position (Left elbow) of cassette stage 1.
$,1,MTCH,C01,01,L,1,B,00000000,00000000,00000000 : XYZ direction offset values included
$,1,MTCH,C01,01,L,1,B : XYZ direction offset values omitted

MABS (Move to Specified Coordinate Position)

[Function]

Moves the specified axis to a specified coordinate position.
For operation definitions and operation directions for each axis, refer to “2.3 Controlled Units”.
If the manipulator grip the wafer when motion start, the controller monitor the wafer status while moving.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.‘s mode selector switch is set to host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),MABS,<Axis>,<Hand>,<Mode>,<Value>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - ‘1’ : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- Axis : Axis (1 byte)
  - ‘S’ : Rotation axis.
  - ‘A’ : Extension axis.
  - ‘H’ : Wrist axis 1.
  - ‘I’ : Wrist axis 2.
  - ‘Z’ : Elevation axis.
- Hand : Blade (1 byte)
  - ‘1’ : Blade 1.
  - ‘2’ : Blade 2.

  Note) If the <Axis> specification is “A: Extension axis”, specify the access blade.
  If the <Axis> specification is not “A: Extension axis”, specify ‘1’
- Mode : Passive blade operation mode (1 byte)
  - ‘C’ : Maintain passive blade posture.
  - ‘H’ : Passive blade fixed to wafer center.

  Note) Valid if the <Axis> specification is “A: Extension axis”.
  If the <Axis> specification is not “A: Extension axis”, specify ‘C’.
- Value : Coordinate (8 bytes, Resolution: 0.001 [mm]/[deg])

  Note) Specified in the range between “-9999999” and “99999999”.
  If value is less than 8 digits, fill the higher digit(s) with ‘0’ so that the field always has 8 digits.
  A sign is added to the highest digit.

  Note) If the operation range is exceeded a stroke limit error will be notified.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,MABS,<Axis>,<Hand>,<Mode>,<Value>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Axis : Axis (1 byte)
- Hand : Blade (1 byte)
- Mode : Motion mode (1 byte)
- Value : Coordinate (8 bytes, Resolution: 0.001 [mm]/[deg])

[Message for the End-of-Execution]

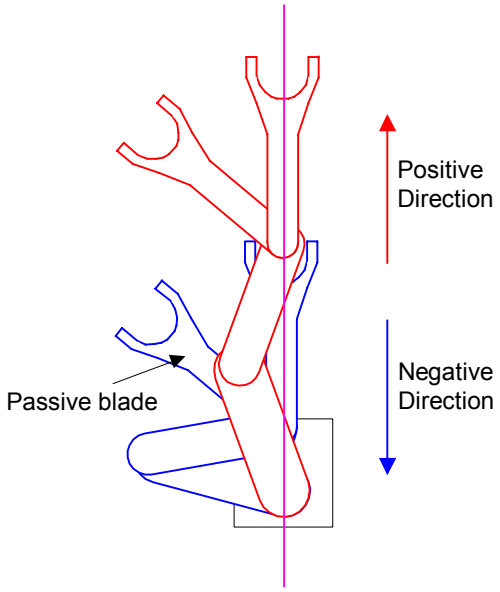
!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,MABS,<ExeTime>,<PosData1>…,<PosDataN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 byte)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])

[Supplementary Explanation]

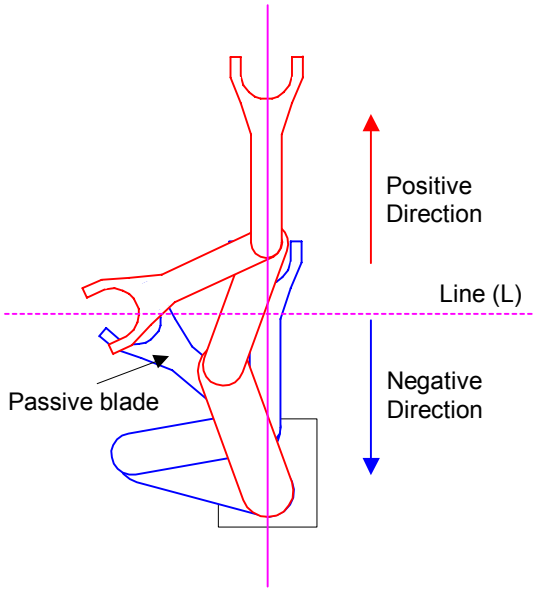
If <Mode> is "C: Maintain passive blade posture"

The postures of the blade specified in <Hand> and the other blade are maintained while the extension axis is moved forward or backward.



If <Mode> is "H: Passive blade fixed to wafer center"

The posture of the blade specified in <Hand> is maintained, while the other unspecified blade (passive blade) is fixed along a line (L) that is set at the wafer center at the time operation starts, and in this state the extension is moved forward or backward.



MREL (Moves to Specified Relative Position)

[Function]

Moves the specified axis from the current position to the specified relative position.
For operation definitions and operation directions for each axis, refer to “2.3 Controlled Units”.
If the manipulator grip the wafer when motion start, the controller monitor the wafer status while moving.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.’s mode selector switch is set to Host mode (if T.P. is connected).

[Command Format]

$,<UNo>(,<SeqNo>),MREL,<Axis>,<Hand>,<Mode>,<Value>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - ‘1’ : Manipulator.
  - ‘2’ : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)
- Axis : Axis (1 byte)
  - ‘S’ : Rotation axis.
  - ‘A’ : Extension axis
  - ‘H’ : Wrist axis1.
  - ‘I’ : Wrist axis2.
  - ‘Z’ : Elevation axis.
  - ‘X’ : Blade X direction (X direction linear motion).
  - ‘Y’ : Blade Y direction (Y direction linear motion).
- Hand : Blade (1 byte)
  - ‘1’ : Blade 1.
  - ‘2’ : Blade 2.

  Note) The blade that will be controlled when the <Axis> specification is “A: Extension axis”, “X: Blade X direction”, or “Y: Blade Y direction”.
  If the <Axis> specification is not those above, specify ‘1’.
- Mode : Passive blade operation mode (1 byte).
  - ‘C’ : Maintain passive blade posture.
  - ‘H’ : Passive blade fixed to wafer center.

  Note) Valid if the <Axis> specification is “A: Extension axis”, “X: blade X direction”, or “Y: blade Y direction”.
  If the <Axis> specification is not any of the above, specify ‘C’.
- Value : Movement amount (8 bytes, Resolution: 0.001 [mm]/[deg])

  Note) Specified in the range between “-9999999” and “99999999”.
  If value is less than 8 digits, fill the higher digit(s) with ‘0’ so that the field always has 8 digits.
  A sign is added to the highest digit.

  Note) If the operation range is exceeded a stroke limit error will be notified.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,MREL,<Axis>,<Hand>,<Mode>,<Value>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Axis : Axis (1 byte)
- Hand : Blade (1 byte)
- Mode : Motion mode (1 byte)
- Value : Movement amount (8 bytes, Resolution: 0.001 [mm]/[deg])

[Message for the End-of-Execution]

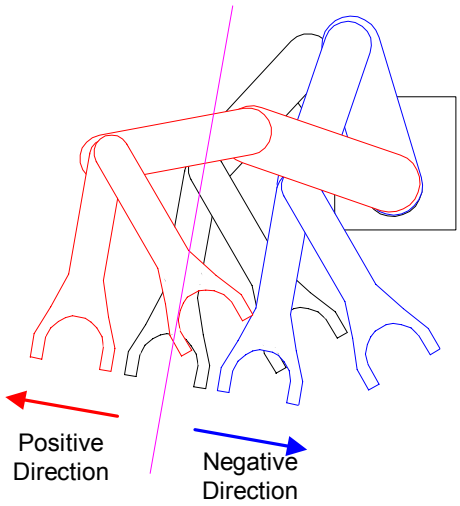
!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,MREL,<ExeTime>,<PosData1>…,<PosDataN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])

[Supplementary Explanation]

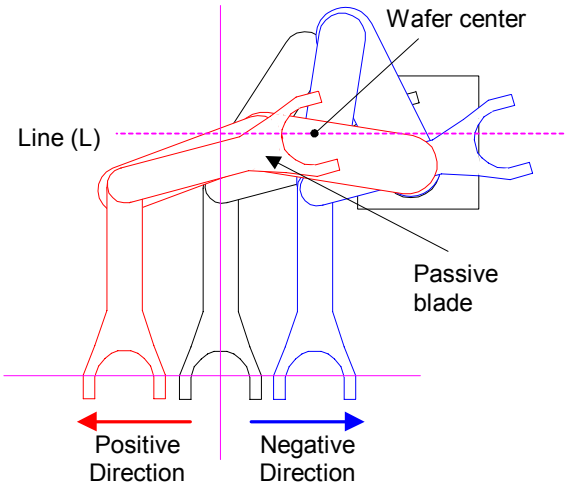
If <Axis> is “Blade X direction”, and <Mode> is “C: maintain passive blade posture”

The posture of the blade not specified in <Hand> (passive blade) is maintained, and the specified blade is moved left/right.



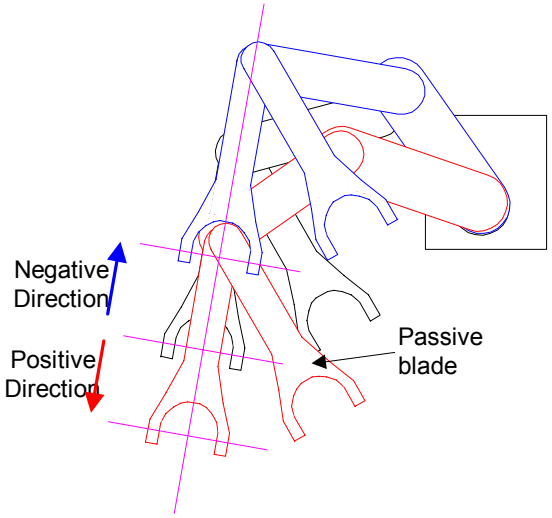
If <Axis> is “Blade X direction” and <Mode> is “H: Passive blade fixed to wafer center”

The unspecified blade (passive blade) is fixed along a line (L) that is set at the wafer center at the time operation starts, and the specified blade is moved left/right.



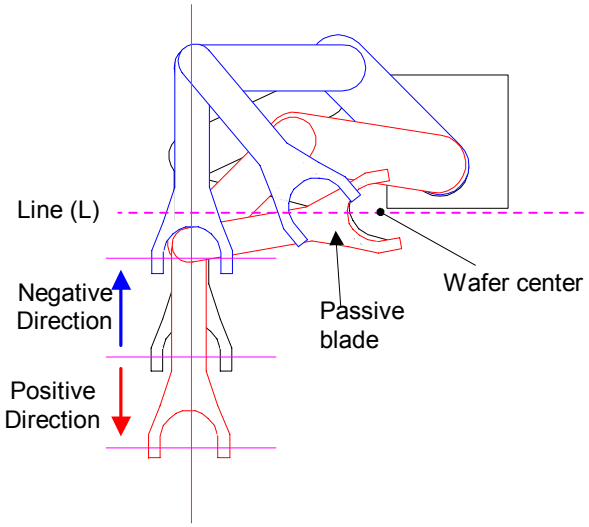
If <Axis> is “Blade Y direction”, and <Mode> is “C: maintain passive blade posture”

The posture of the blade not specified in <Hand> (passive blade) is maintained, and the specified blade is moved forward/backward.



If <Axis> is “Blade Y direction” and <Mode> is “H: Passive blade fixed to wafer center”

The unspecified blade (passive blade) is fixed along a line (L) that is set at the wafer center at the time operation starts, and the specified blade is moved forward/backward.



* For extension axis operation, refer to the “MABS” command.

MMAP (Wafer Mapping)

[Function]

Performs the wafer mapping at the specified transfer station.
Mapping at the specified transfer station is performed in the following order.
(1) Moves the arm(s) to the minimum sweep posture through a safe path.
(2) Moves the arm(s) to the mapping start position.
If wafer protrusion is detected, moves to the mapping start position after performing the protrusion detection operation.
(3) Moves the elevation axis to search for the wafer positions.
(4) Moves the arm(s) to a minimum sweep posture.
(5) Notifies the mapping result to the host.
The mapping result can also be referenced using the “RMAP” command mentioned below.
For mapping motion, refer to “8 Wafer Mapping Function”.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.’s mode selector switch is set to Host mode (if T.P. is connected).
- The specified transfer station has been registered.
- The specified transfer station’s calibration operation for mapping has been performed.

[Command Format]

$,<UNo>(,<SeqNo>),MMAP,<TrsSt>,<Slot>(,<Safe>)(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - ‘1’ : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- TrsSt : Transfer station (3 bytes)
  - “C01” - “C08”: Cassette stage.
  - “S01” - “S12”: Transfer stage.
  - “P01” : P/A stage.
- Slot : Slot number (2 bytes)
  <Cassette stage>
  - “01” - “30” : When specifying individual cassette stage slot.
  - “00” : When specifying all slots.

  <Transfer stage, P/A stage>
  - “00” : Fixed value t(because this type of station does not have multiple slots.)

  Note) If value is less than 2 digits, fill the higher digit with ‘0’ so that the field always has 8 digits.
- Safe : Specifies wafer protrusion detection operation yes/no (1 byte)
  - ‘0’ : No wafer protrusion detection operation.
  - ‘1’ : Wafer protrusion detection operation performed.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,MMAP,<TrsSt>,<Slot>(,<Safe>)(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- TrsSt : Transfer station (3 bytes)
- Slot : Slot number (2 bytes)
- Safe : Wafer protrusion detection operation yes/no (1 byte)

[Message for the End-of-Execution]

!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,MMAP,<ExeTime>,<PosData1>,…<PosDataN>
,01:<Result1>…,N:<ResultN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])
- Result* : Mapping result (2 bytes each)
  - "--" : No wafer detected.
  - "OK" : Wafer inserted correctly.
  - "CW" : Wafer inserted incorrectly (inclined).
  - "DW" : Wafer inserted incorrectly (duplicated).

  Note) Responds with the number of slots of the specified transfer station.

[Usage Example]

(1) Mapping motion (Cassette stage 1/ slot 3) without the wafer protrusion detection
$,1,MMAP,C01,03,0 : command format
$,1,00,0000,MMAP,C01,03,0 : response message
!,1,00,0000,MMAP,001500,00000450,-0757520,-0090450,00036200,00105500,03:OK
: End-of-Execution message

(2) Mapping motion(Cassette stage 2/all slot) with the wafer protrusion detection
(slot numbers: 10, slot2: duplicated, slot4/5: inclined, slot8: no wafer)
$,1,MMAP,C02,00,1 : command format
$,1,00,0000,MMAP,C02,00,1 : response message
!,1,00,0000,MMAP,006380,00051950,00531150,00038410,00167210,00405500,
01:OK,02:DW,03:OK,04:CW,05:CW,06:OK,07:OK,08:--,09:OK,10:OK
: End-of-Execution message

(3) Mapping motion(Transfer stage 1) without the wafer protrusion detection
$,1,MMAP,S01,00,0 : command format
$,1, 00,0000,MMAP,S01,00,0 : response message
!,1,00,0000,MMAP,005640,-0042300,-0618410,00132300,00003500,00300000,01:OK
: End-of-Execution message

MMCA (Mapping Calibration)

[Function]

Performs a calibration operation for mapping for the specified cassette station.
Acquires the base data for mapping via a calibration operation.
Also refreshes the threshold for detection of duplicated or inclined insertion detection.
The mapping calibration at the specified cassette station is performed in the following order.
(1) Moves the arm(s) to the minimum sweep posture through a safe path.
(2) Moves the arm(s) to the mapping start position.
If wafer protrusion detection is performed, moves to the mapping start position after the protrusion detection operation is performed.
(3) Moves the elevation axis to search for the wafer position.
(4) Moves the arm(s) to a minimum sweep posture.
(5) Notifies the calibration result for mapping to the host.
The calibration result for mapping can also be referenced using the “RMCA” command mentioned below.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.’s mode selector switch is set to Host mode (if T.P. is connected).
- The specified transfer station has been registered.
- The wafer needs to be inserted only in the lowest slot and the highest slots of cassette stage.

[Command Message]

$,<UNo>(,<SeqNo>),MMCA,<TrsSt>(,<Safe>)(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - ‘1’ : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- TrsSt : Transfer station (3 bytes)
  - “C01” - “C08” : Cassette stage.
- Safe : Specifies wafer protrusion detection operation yes/no (1 byte)
  - ‘0’ : No wafer protrusion detection operation.
  - ‘1’ : Wafer protrusion detection operation performed.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,MMCA,<TrsSt>(,<Safe>)(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- TrsSt : Transfer station (3 bytes)
- Safe : Wafer protrusion detection operation yes/no (1 byte)

[Message for the End-of-Execution]

!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,MMCA,<ExeTime>,<PosData1>,…<PosDataN>,
<Value1>…,<Value6>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])
- Value1 : Lowest-slot position (8 bytes, Resolution: 0.001 [mm])
- Value2 : highest-slot position (8 bytes, Resolution: 0.001 [mm])
- Value3 : Wafer width (8 bytes, Resolution: 0.001 [mm])
- Value4 : The threshold value of double insertion (8 bytes, Resolution: 0.001 [mm])
- Value5 : The threshold value of slanting insertion1 (8 bytes, Resolution: 0.001 [mm])
- Value6 : The threshold value of slanting insertion2 (8 bytes, Resolution: 0.001 [mm])

MALN (Wafer alignment)

[Function]

Aligns the wafer on the pre-aligner.
The wafer alignment on the pre-aligner is done in the following order.
(1) Rotates the rotation axis and samples the wafer edge profile data.
(2) Calculates the wafer center and notch (orientation flat) location and moves the wafer to the aligned position for the unit the wafer needs to be compensated for.
(3) Notifies the host of the alignment result.
The alignment result can also be referenced using the “RALN” command mentioned below.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.’s mode selector switch is set to Host mode (if T.P. is connected).
- In the case of edge grip pre-aligner, the action command immediately before this command has to be a successfully completed “MTRS” command.

[Command Format]

$,<UNo>(,<SeqNo>),MALN,<Mode>,<Angle>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - ‘2’ : Pre-aligner
- SeqNo : Sequence number (None / 2 bytes)
- Mode : Motion mode (1 byte)
  - ‘0’ : Sampling operation + Correction operation.
  - ‘1’ : Correction operation.
  - ‘2’ : Sampling operation.
- Angle: Positioning angle (None / 8 bytes each, Resolution: 0.001 [deg])
  Note) Relative angle from the position set by alignment calibration as the reference point.
  Note) Specified in the range between “00000000” and “00359999”.
  If a value is less than the specified digits, fill the higher digit(s) with ‘0’ so that the field always has specified digits.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,MALN,<Mode>,<Angle>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mode : Motion mode (1 byte)
- Angle : Positioning angle (None / 8 bytes each, Resolution: 0.001 [deg])

[Message for the End-of-Execution]

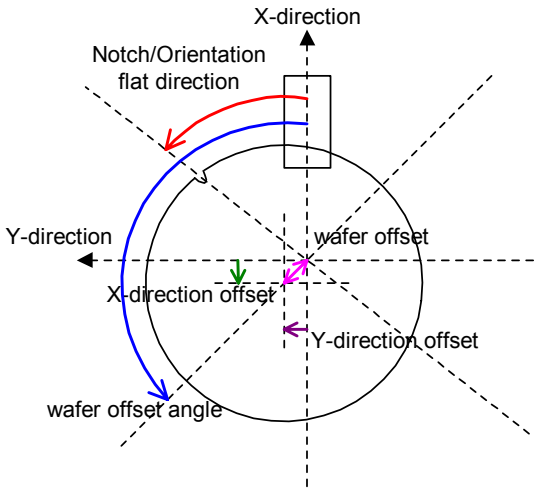
!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,MALN,<ExeTime>,<PosData1>…,<PosDataN>
,<Value1>…,<Value10>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])
- Value1 : Wafer eccentric amount before alignment operation (8 bytes, Resolution: 0.001 [mm])
- Value2 : Wafer eccentric angle direction before alignment operation (8 bytes, Resolution: 0.001 [deg])
- Value3 : Notch/Orientation Flat direction before alignment operation (8 bytes, Resolution: 0.001 [deg])
- Value4 : X direction offset amount before alignment operation (8 bytes, Resolution: 0.001 [mm])
- Value5 : Y direction offset amount before alignment operation (8 bytes, Resolution: 0.001 [mm])
- Value6 : Pre-aligner adjustment angle (8 bytes, Resolution: 0.001 [deg])
- Value7 : Manipulator adjustment amount (8 bytes, Resolution: 0.001 [mm])
- Value8 : Manipulator adjustment angle (8 bytes, Resolution: 0.001 [deg])
- Value9 : X direction offset amount after alignment operation (8 bytes, Resolution: 0.001 [mm])
- Value10 : Y direction offset amount after alignment operation (8 bytes, Resolution: 0.001 [mm])
  - If a value is less than the specified digits, fill the higher digit(s) with ‘0’ so that the field always has specified digits.

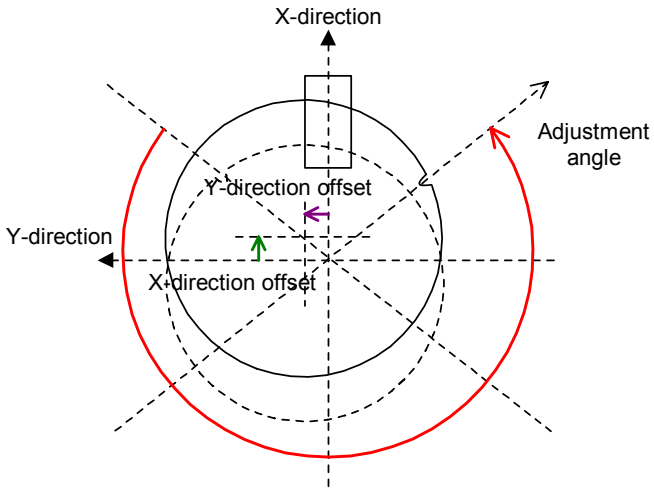
[Supplementary Explanation]

The alignment result are as follows:

Before alignment motion
(The state that the manipulator has put the wafer on the PA stage)



After alignment motion
(The state that the Pre-aligner has aligned the notch)



- Wafer offset amount: The distance between Pre-aligner rotation center and the wafer center
- Wafer offset angle: The wafer offset angle from CCD sensor
- Notch/Orentation flat direction: The Notch(Orientation flat) angle from CCD sensor
- X-direction offset amount: The X-direction offset amount between the Pre-aligner rotation center and the wafer center.
- Y-direction offset amount: The Y-directionoffset amount between the Pre-aligner rotation center and the wafer center.
- Pre-aligner adjustment angle: The angle between the angle before alignment operation and the angle after alignment operationafter alignment motion.
- Manipulator adjustment amount: The extension axis adjustment amout for centering the wafer when the manipulator get the wafer from the PA stage.
- Manipulator adjustment angle: The rotation axis adjustment angle for centering the wafer when the manipulator get the wafer from the PA stage.

For the manipulator offset amount and offset angle, refer to “Figure 9.1 Alignment Adjustment Outline(When Pre-aligner Station Access is in Link Mode)”

MACA (Alignment Calibration)

[Function]

Performs calibration for the wafer alignment.
The pre-aligner collect the data to align while rotating.
Performs the notch finding motion (rotate the wafer) and measure the wafer center and the notch (or the orientation flat) location. And then save the detected notch position as reference angle for wafer alignment that would be executed later.
Notifies the calibration result for alignment to the host.
The calibration result for alignment can also be referenced using the “RACA” command mentioned below.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.’s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),MACA,<Mode>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - ‘2’ : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)
- Mode: Motion mode (1 byte)
  - ‘0’ : Notch calibration.
  - ‘1’ : Arm calibration.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,MACA,<Mode>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mode : Motion mode (1 byte)

[Message for the End-of-Execution]

!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,MACA,<ExeTime>,<PosData1>…,<PosDataN>
,<Value1>,<Value2>,<Value3>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])
- Value1 : calibration angle (8 bytes, Resolution: 0.001 [deg])
- Value2 : Manipulator advance angle (8 bytes, Resolution: 0.001 [deg])
- Value3 : Distance between manipulator rotation center and pre-aligner rotation center distance (8 bytes, Resolution: 0.001 [mm])

CSTP (Deceleration/Emergency stop)

[Function]

Applies deceleration/emergency stop to stop the motion of the specified unit.
<If during operation>
- If a deceleration stop command is received, a deceleration stop is performed and the “CSTP” command sequence number will be responded for the end-of–execution message.
- If an emergency stop command is received, deceleration stop + servo off is performed, and the end-of-execution message for the current command will notify an emergency stop error.

< If during a stop >
- If a deceleration stop command is received, the “CSTP” command sequence number will be responded for the end-of–execution message.
- If an emergency stop command is received, servo off is performed, and an emergency stop error will be registered.

The next command received will be responded with an emergency stop error notification.

[Conditions]

- T.P.’s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),CSTP,<Sw>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - ‘1’ : Manipulator.
  - ‘2’ : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)
- Sw : Stop mode (1 byte)
  - ‘H’ : Deceleration to a stop.
  - ‘E’ : Emergency stop.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,CSTP,<Sw>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Sw : Stop mode (1 byte)

[Message for the End-of-Execution]

(Deceleration stop)

!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,CSTP,<ExeTime>,<PosData1>…,<PosDataN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])

(Emergency stop)

This command responds with the completion message of the interrupted action command.

CRSM (Resume the interrupted motion)

[Function]

Restarts the motion interrupted by deceleration stop.
After completion of the restarted motion, sends the completion message of interrupted action command.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).
- The last command is "CSTP" command(deceleration stop).

[Command Message]

$,<UNo>(,<SeqNo>),CRSM(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,CRSM(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)

[Message for the End-of-Execution]

This command responds with the completion message of the interrupted action command.

CSRV (Servo power command)

[Function]

Turns ON/OFF the servo power of the specified unit.

[Conditions]

< To turn the servo on >
• The specified unit is under ready state.
• T.P.'s mode selector switch is set to Host mode (if T.P. is connected).
• No alarm is occurring.
< To turn the servo off >
• The specified unit is under ready state.
• T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),CSRV,<Sw>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)
- Sw : Servo command (1 byte)
  - '0' : Servo OFF.
  - '1' : Servo ON.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,CSRV,<Sw>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Sw : Servo command (1 byte)

[Message for the End-of-Execution]

!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,CSRV,<ExeTime>,<PosData1>…,<PosDataN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])

CCLR (Clear the error)

[Function]

Clears the current error or error history of the specified unit.
Note: The cause of error itself cannot be cleared by this command (for example, the need to release the e-stop button before clearing the error).
Also, depending on its severity, some errors cannot be cleared.
The error include the warning and the alarm.

[Conditions]

< For an error clear >
• The specified unit is under ready state.
• T.P.'s mode selector switch is set to Host mode (if T.P. is connected).
< Clearing the error history >
• The specified unit is under ready state.
• T.P.'s mode selector switch is set to Host mode (if T.P. is connected).
• No error is occurring.

[Command Message]

$,<UNo>(,<SeqNo>),CCLR,<CMode>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner
- SeqNo : Sequence number (None / 2 bytes)
- CMode : Clear mode (1 byte)
  - 'E' : Clears the error status.
  - 'H' : Clears the error history (in the volatile memory).

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,CCLR,<CMode>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- CMode : Clear mode (1 byte)

[Message for the End-of-Execution]

!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,CCLR,<ExeTime>,<PosData1>…,<PosDataN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])

CSOL (Solenoid control command)

[Function]

Command wafer hold/release signal to the solenoid of the specified unit.
If a wait time is set, following a period of time after the solenoid control, check the wafer sensor status and the end-of-execution message is notified.
If a wait time is not set, immediately after the solenoid control, the end-of-execution message is notified. As a result, the command added to the end-of-execution message status does not confirm the wafer status.

[Conditions]

- The specified unit is under ready state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),CSOL,<Sol>,<Sw>,<Wait>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)
- Sol : Solenoid control specification (1 byte)
  - < Manipulator >
  - '1' : Blade 1.
  - '2' : Blade 2.
  - 'F' : Blade 1 + Blade 2.
  - < Pre-aligner >
  - '1' : Pre-aligner.
  - '2' : Edge-grip pre-aligner Lifter.
- Sw : Solenoid command (1 byte)
  - '0' : Wafer release.
  - '1' : Wafer hold.
- Wait : Wait time (1 byte)
  - '0' : No wait time.
  - '1' : With wait time.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,CSOL,<Sol>,<Sw>,<Wait>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Sol : Solenoid control specification (1 byte)
- Sw : Solenoid command (1 byte)
- Wait : Wait time (1 byte)

[Message for the End-of-Execution]

!,<UNo>(,<SeqNo>),<Sts>,<Errcd>,CSOL,<ExeTime>,<PosData1>…,<PosDataN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Errcd : Error code (4 bytes)
- ExeTime : Execution time (6 bytes, Resolution: 1 [msec])
- PosData* : Coordinate Data (8 bytes each, Resolution: 0.001 [mm]/[deg])

SSPD (Set Motion Speed)

[Function]

Sets the operation speed (No-wafer transfer speed, With-wafer transfer speed, Low-speed transfer speed, Home return speed, Low-speed-area speed) for the specified axis of the specified unit.
The speed set by this command is not saved in non-volatile memory.
When the controller power is cycled, the default speed setting in the non-volatile memory is used.

[Conditions]

- The specified unit is under ready state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),SSPD,<Level>,<Type>,<Axis>,<Speed>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)
- Level : Transfer speed level (1 byte)
  - '0' : Currently set speed level.
  - '1' : Speed level 1.
  - '2' : Speed level 2.
  - '3' : Speed level 3.
- Type : Speed type (1 byte)
  - 'H' : No-wafer transfer speed.
  - 'M' : With-wafer transfer speed.
  - 'L' : Low speed.
  - 'O' (Alphabet. Not zero.) : Home speed.
  - 'B' : Speed in low-speed-area.
- Axis : Axis (1 byte)
  - < Manipulator >
  - 'S' : Rotation axis.
  - 'A' : Extension axis.
  - 'H' : Wrist axis1.
  - 'I' : Wrist axis2.
  - 'Z' : Elevation axis.
  - 'R' : Linear access motion speed.
  - 'G' : All axes (Setting method is % specification only).
  - < Pre-aligner >
  - 'S' : Rotation axis.
  - 'G' : All axes (Setting method is % specification only).
- Speed : Speed data (8 bytes)
  - Note) Specified in the range between "00000001" and "99999999". (Resolution: 0.001 [mm/sec], [deg/sec])
  - Note) Adding '%' before the first digit sets to the ratio of the specified speed's maximum speed. When specifying a percentage, specify in the range between "%0000001" and "%0001000". (Resolution: 0.1 [%])
  - Note) If value is less than 8 digits, fill the higher digit(s) with '0' so that the field always has 8 digits.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SSPD,<Level>,<Type>,<Axis>,<Speed>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Level : Transfer speed level (1 byte)
- Type : Speed type (1 byte)
- Axis : Axis (1 byte)
- Speed : Speed data (8 bytes)

SSLV (Transfer speed level setting)

[Function]

Selects the speed level (No-wafer transfer speed, With-wafer transfer speed, Low-speed transfer speed, Home return speed, Low-speed-area speed) for the operation speed.
The speed set by this command is not saved in non-volatile memory.
If the controller power is cycled, speed level 1 is selected by default.

[Conditions]

- The specified unit is under ready state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),SSLV,<Level>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)
- Level : Transfer speed level (1 byte)
  - '1' : Speed level 1.
  - '2' : Speed level 2.
  - '3' : Speed level 3.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SSLV,<Level>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Level : Transfer speed level (1 byte)

SPOS (Registration of Current Position)

[Function]

Registers the current position of the specified unit as the specified transfer station.
The memory type (volatile or non-volatile) to register the position data can be specified.
The position data registered in the volatile memory is held until the power supply is turned off.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),SPOS,<Mem>,<RMode>,<TrsSt>,<Slot>,<Posture>,<Hand>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- Mem : Memory specification (1 byte)
  - 'V' : Volatile memory.
  - 'N' : Volatile memory and Non-volatile memory.
- RMode : Registration mode (1 byte)
  - 'N' : Specified position registration.
  - 'A' : Master registration.
- TrsSt : Transfer station (3 bytes)
  - "C01" - "C08" : Cassette stage (the lowest-layer position).
  - "H01" - "H08" : Cassette stage (the highest-layer position).
  - "S01" - "S12" : Transfer stage.
  - "P01" : P/A stage.
- Slot : Slot number (2 bytes)
  <Cassette stage>
  - "01" - "30" : If <TrsSt> is "C01"-"C08"(cassette stage), the slot specified.
  - "00" : If <TrsSt> is "C01"-"C08"(cassette stage), recognize the lowest-slot.
    If <TrsSt> is "H01"-"H08"(cassette stage), recognize the highest-slot.

  <Transfer stage, Pre-aligner stage>
  - "00" : Fixed value t(because this type of station does not have multiple slots.)

  Note) If value is less than 2 digits, fill the higher digit with '0' so that the field always has 2 digits.
  Note) If the slot number is specified, the lowest-slot position or the highest-slot position are calculated by the slot number and the slot pitch, and save it.
  Note) If the operation range is exceeded a stroke limit error will be notified.
- Posture : Arm Posture (1 byte)
  - 'L' : Left elbow.
  - 'R' : Right elbow.
- Hand : Blade (1 byte)
  - '1' : Blade 1.
  - '2' : Blade 2.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SPOS,<Mem>,<RMode>,<TrsSt>,<Slot>,<Posture>
<Hand>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mem : Memory specification (1 byte)
- RMode : Registration mode (1 byte)
- TrsSt : Transfer station (3 bytes)
- Slot : Slot number (2 bytes)
- Posture : Arm Posture (1 byte)
- Hand : Blade (1 byte)

SABS (Registration of Coordinate Position)

[Function]

Registers the coordinate position as the specified transfer station for the specified unit.
The memory type (volatile or non-volatile) to register the position data can be specified.
The position data registered in the volatile memory is held until the power supply is turned off.

[Conditions]

- The specified unit is under ready state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),SABS,<Mem>,<RMode>,<TrsSt>,<Posture>,<Hand>,
<Value1>…,<ValueN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- Mem : Memory specification (1 byte)
  - 'V' : Volatile memory.
  - 'N' : Volatile memory and Non-volatile memory.
- RMode : Registration mode (1 byte)
  - 'N' : Specified position registration.
  - 'A' : Master registration.
- TrsSt : Transfer station (3 bytes)
  - "C01" - "C08" : Cassette stage (the lowest-layer position).
  - "H01" - "H08" : Cassette stage (the highest-layer position).
  - "S01" - "S12" : Transfer stage.
  - "P01" : P/A stage.
- Posture : Arm Posture (1 byte)
  - 'L' : Left elbow.
  - 'R' : Right elbow.
- Hand : Blade (1 byte)
  - '1' : Blade 1.
  - '2' : Blade 2.
- Value* : Coordinate (8 bytes each, Resolution: 0.001 [mm]/[deg])
  Note) Specified in the range between "-9999999" and "99999999".
  If the operation range is exceeded, a stroke limit error is notified.
  If value is less than 8 digits, fill the higher digit(s) with '0' so that the field always has 8 digits.
  A sign is added to the highest digit.
  Note) If the operation range is exceeded a stroke limit error will be notified.
  Note) The number of "ValueN" depends on the unit type. Set as many axis numbers as the specified unit has.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SABS,<Mem>,<RMode>,<TrsSt>,<Posture>,<Hand>,
<Value1>…,<ValueN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mem : Memory specification (1 byte)
- RMode : Registration mode (1 byte)
- TrsSt : Transfer station (3 bytes)
- Posture : Arm Posture (1 byte)
- Hand : Blade (1 byte)
- Value* : Coordinate (8 bytes each, Resolution: 0.001 [mm]/[deg])

SAPS (Registered Position Adjustment)

[Function]

Adjusts the specified transfer station's teaching position.
The memory type (volatile or non-volatile) to register the position data can be specified.
The position data registered in the volatile memory is held until the power supply is turned off.

[Conditions]

- The specified unit is under ready state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),SAPS,<Mem>,<RMode>,<TrsSt>,<Posture>,<Hand>,
<OfstX>,<OfstY>,<OfstZ>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- Mem : Memory specification (1 byte)
  - 'V' : Volatile memory.
  - 'N' : Volatile memory and Non-volatile memory.
- RMode : Registration mode (1 byte)
  - 'N' : Specified position registration.
  - 'A' : Master registration.
- TrsSt : Transfer station (3 bytes)
  - "C01" - "C08" : Cassette stage (the lowest-layer position).
  - "H01" - "H08" : Cassette stage (the highest-layer position).
  - "S01" - "S12" : Transfer stage.
  - "P01" : P/A stage.
- Posture : Arm Posture (1 byte)
  - 'L' : Left elbow.
  - 'R' : Right elbow.
- Hand : Blade (1 byte)
  - '1' : Blade 1
  - '2' : Blade 2
- Ofst* : XYZ direction offset (8 bytes each, Resolution: 0.001 [mm])
  - OfstX : X direction offset
  - OfstY : Y direction offset
  - OfstZ : Z direction offset

  Note) Specified in the range between "-0009999" and "00009999".
  If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.
  A sign is added to the highest digit.

  Note) If the operation range is exceeded a stroke limit error will be notified.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SAPS,<Mem>,<RMode>,<TrsSt>,<Posture>,<Hand>,
<OfstX>,<OfstY>,<OfstZ>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mem : Memory specification (1 byte)
- RMode : Registration mode (1 byte)
- TrsSt : Transfer station (3 bytes)
- Posture : Arm Posture (1 byte)
- Hand : Blade (1 byte)
- Ofst* : XYZ direction offset (8 bytes each, Resolution: 0.001 [mm])

SPDL (Delete Registered Position)

[Function]

Deletes the registered position of the specified unit's specified transfer station.
The memory type (volatile or non-volatile) to register the position data can be specified.

[Conditions]

- The specified unit is under ready state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),SPDL,<Mem>,<TrsSt>,<Posture>,<Hand>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- Mem : Memory specification (1 byte)
  - 'V' : Only volatile memory deleted.
  - 'N' : Delete volatile memory and non-volatile memory.
- TrsSt : Transfer station (3 bytes)
  - "C01" - "C08" : Cassette stage (the lowest-layer position).
  - "H01" - "H08" : Cassette stage (the highest-layer position).
  - "S01" - "S12" : Transfer stage.
  - "P01" : P/A stage.
  - "FFF" : All transfer stations.
- Posture : Arm Posture (1 byte)
  - 'L' : Left elbow.
  - 'R' : Right elbow.
  - 'F' : Both elbow postures (Left elbow, Right elbow).
- Hand : Blade (1 byte)
  - '1' : Blade 1.
  - '2' : Blade 2.
  - 'F' : Both end-effectors (Blade 1, Blade 2).

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SPDL,<Mem>,<TrsSt>,<Posture>,<Hand>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mem : Memory specification (1 byte)
- TrsSt : Transfer station (3 bytes)
- Posture : Arm Posture (1 byte)
- Hand : Blade (1 byte)

SPSV (Registers the position data in the volatile memory to the non-volatile memory.)

[Function]

Registers the position data in the volatile memory to the non-volatile memory.

[Conditions]

- The specified unit is under ready state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),SPSV,<TrsSt>,<Posture>,<Hand>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- TrsSt : Transfer station (3 bytes)
  - "C01" - "C08" : Cassette stage (the lowest-layer position).
  - "H01" - "H08" : Cassette stage (the highest-layer position).
  - "S01" - "S12" : Transfer stage.
  - "P01" : P/A stage.
  - "FFF" : All transfer stations.
- Posture : Arm Posture (1 byte)
  - 'L' : Left elbow.
  - 'R' : Right elbow.
  - 'F' : Both elbow posture (Left elbow, Right elbow).
- Hand : Blade (1 byte)
  - '1' : Blade 1.
  - '2' : Blade 2.
  - 'F' : Both end-effector (Blade 1, Blade 2).

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SPSV,<TrsSt>,<Posture>,<Hand>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- TrsSt : Transfer station (3 bytes)
- Posture : Arm Posture (1 byte)
- Hand : Blade (1 byte)

SPLD (Reads the position data in the non-volatile memory into the volatile memory.)

[Function]

Reads the position data in the non-volatile memory into the volatile memory.

[Conditions]

- The specified unit is under ready state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),SPLD,<TrsSt>,<Posture>,<Hand>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- TrsSt : Transfer station (3 bytes)
  - "C01" - "C08" : Cassette stage (the lowest-layer position).
  - "H01" - "H08" : Cassette stage (the highest-layer position).
  - "S01" - "S12" : Transfer stage.
  - "P01" : P/A stage.
  - "FFF" : All transfer stations.
- Posture : Arm Posture (1 byte)
  - 'L' : Left elbow.
  - 'R' : Right elbow.
  - 'F' : Both elbow posture (Left elbow, Right elbow).
- Hand : Blade (1 byte)
  - '1' : Blade 1
  - '2' : Blade 2
  - 'F' : Both end-effector (Blade 1, Blade 2)

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SPLD,<TrsSt>,<Posture>,<Hand>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- TrsSt : Transfer station (3 bytes)
- Posture : Arm Posture (1 byte)
- Hand : Blade (1 byte)

SSTR (Set Transfer Station Information)

[Function]

Sets transfer station information.
The memory type (volatile or non-volatile) to register the position data can be specified.
The station's information parameters registered in the volatile memory is held until the power supply is turned off.

[Conditions]

- The specified unit is under ready state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),SSTR,<Mem>,<TrsSt>,<Item>,<Value>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- Mem : Memory specification (1 byte)
  - 'V' : Volatile memory.
  - 'N' : Volatile memory and Non-volatile memory.
- TrsSt : Transfer station (3 bytes)
  - "C01" - "C08" : Cassette stage.
  - "S01" - "S12" : Transfer stage.
  - "P01" : P/A stage.
- Item : Transfer station information(2 bytes)
  - See the supplementary explanation.
- Value : Parameter value (8 bytes)
  - See the supplementary explanation.

  Note) If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.
  A sign is added to the highest digit.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SSTR,<Mem>,<TrsSt>,<Item>,<Value>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mem : Memory specification (1 byte)
- TrsSt : Transfer station (3 bytes)
- Item : Transfer station information(2 bytes)
- Value : Parameter value (8 bytes)

[Supplementary Explanation]

Transfer station information contents are as follows.

Item	Contents	Unit	Setting Range
“00”	Downward offset	0.001 [mm]	“-9999999” - “99999999”
“01”	Upward offset	0.001 [mm]	“-9999999” - “99999999”
“02”	Grip position offset	0.001 [mm]	“-9999999” - “99999999”
“06”	G2/P3 Offset in the extending direction	0.001 [mm]	“-9999999” - “99999999”
“08”	Put downward offset	0.001 [mm]	“-9999999” - “99999999”
“30”	Get operation Move_grip function yes/no	[-]	“00000000”～“00000001”
“31”	Get operation rising pattern	[-]	“00000000”～“00000001”
“40”	Put operation Move_grip function yes/no	[-]	“00000000”～“00000001”
“41”	Put operation dropping pattern	[-]	“00000000”～“00000001”

SPRM (Setting of parameter)

[Function]

The specified parameter value of a specified unit is changed.
This command registers the position to the Non-Volatile Memory.

[Conditions]

- The specified unit is under ready state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).
- The specified unit is under servo OFF state.

[Command Message]

$,<UNo>(,<SeqNo>),SPRM,<Type>,<PrmNo>,<Value>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)
- Type : Parameter type (3 bytes)
  - "CRU" : Common real number parameter.
  - "CIU" : Common integer parameter.
  - "URU" : Unit real number parameter.
  - "UIU" : Unit integer parameter.
- PrmNo : Parameter number (4 bytes)
  Note) Specified in the range between "0000" and "9999".
  If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.
- Value : Parameter value (12 bytes, Resolution: 0.0001)
  Note) Specified in the range between "-02147483648" and "002147483647".
  If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.
  A sign is added to the highest digit.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SPRM,<Type>,<PrmNo>,<Value>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Type : Parameter type (3 bytes)
- PrmNo : Parameter number (4 bytes)
- Value : Parameter value (12 bytes, Resolution: 0.0001)

SMSK (Enable/Disable the Interlock monitoring)

[Function]

Enable or disable the interlock signal monitoring function.
This setting is registered in the volatile memory. All monitoring functions are enabled when the controller power is cycled.

[Conditions]

- The specified unit is under ready state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),SMSK,<Valid>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)
- Valid : Interlock information (4 bytes)
  - See the supplementary explanation.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SMSK,<Valid>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Valid : Interlock information (4 bytes)

[Supplementary Explanation]

The interlock information contents are as follows.

< Manipulator >

	Bit #	Contents	Setting
Valid1	Bit0	Wafer presence/absence check on Blade 1	0: Enabled, 1: Disabled
	Bit1	Wafer presence/absence check on Blade 2	
	Bit2	Reserved	
	Bit3	Reserved	
Valid2	Bit0	Pre-aligner operation interlock.	
	Bit1	Pre-aligner wafer status interlock.	
	Bit2	Pre-aligner wafer status interlock. Wafer status checked by CCD sensor.	
	Bit3	Reserved	
Valid3	Bit0	Reserved	
	Bit1	Reserved	
	Bit2	Reserved	
	Bit3	Reserved	
Valid4	Bit0	Reserved	
	Bit1	Reserved	
	Bit2	Reserved	
	Bit3	Reserved	

< Pre-aligner >

	Bit #	Contents	Setting
Valid1	Bit0	Wafer presence/absence check.	0: Enabled, 1: Disabled
	Bit1	Reserved.	
	Bit2	Reserved.	
	Bit3	Reserved.	
Valid2	Bit0	Reserved.	
	Bit1	Reserved.	
	Bit2	Reserved.	
	Bit3	Reserved.	
Valid3	Bit0	Reserved.	
	Bit1	Reserved.	
	Bit2	Reserved.	
	Bit3	Reserved.	
Valid4	Bit0	Manipulator operation interlock. Check if the manipulator is accessing pre-aligner.	
	Bit1	Reserved.	
	Bit2	Reserved.	
	Bit3	Reserved.	

The interlock information is set hexadecimal in order to <Valid1><Valid2><Valid3><Valid4>.
For the editing method of the interlock information, refer to “4.2 Response Message (Controller → Host)”

SSTD (Reference position record command)

[Function]

The current position is registered as a reference position of manipulator coordinates.
This command registers the position to the Non-Volatile Memory.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),SSTD,<Axis>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- Axis : Axis (1 byte)
  < Manipulator >
  - 'A' : Arm section only.
  - 'Z' : Elevation axis.
  - 'G' : All axes.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SSTD,<Axis>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Axis : Axis (1 byte)

SSTN (Reference position record command (for inputting numerical values))

[Function]

Specify the reference position encoder value and record the reference position.
This command registers the position to the Non-Volatile Memory.

[Conditions]

- The specified unit is under ready state.
- The specified unit is under servo ON state.
- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),SSTN,<Value1>…,<ValueN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- Value* : Reference position of the encoder value (12 bytes each)
  - Value1 : Arm1.
  - Value2 : Arm 2.
  - Value3 : Wrist axis 1.
  - Value4 : Wrist axis 2.
  - Value5 : Elevation axis.

  Note) Specified in the range between "-02147483648" and "002147483647".
  If value is less than 12 digits, fill the higher digit(s) with '0' so that the field always has 12 digits.
  A sign is added to the highest digit.

  Note) The number of "ValueN" depends on the unit type. Set as many axis numbers as the specified unit has.

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SSTN,<Value1>…,<ValueN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Value* : Reference position of the encoder value (12 bytes each)

RSPD (Motion speed reference)

[Function]

References the motion speed setting of the axis of the specified unit. (No-wafer speed, With-wafer speed, Low speed, home return speed, Low-speed-area speed).
This command refers the default speed setting in the volatile memory.

[Conditions]

• T.P.'s mode selector switch is set to Host mode (if T.P. is connected).

[Command Message]

$,<UNo>(,<SeqNo>),RSPD,<Level>,<Type>,<Axis>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)
- Level : Transfer speed level (1 byte)
  - '0' : Currently set speed level.
  - '1' : Speed level 1.
  - '2' : Speed level 2.
  - '3' : Speed level 3.
- Type : Speed type (1 byte)
  - 'H' : No-wafer speed.
  - 'M' : With-wafer speed.
  - 'L' : Low speed.
  - 'O' : (Alphabet. Not zero.): home return speed.
  - 'B' : Speed in low-speed-area.
- Axis : Axis (1 byte)

  < Manipulator >
  - 'S' : Rotation axis.
  - 'A' : Extension axis.
  - 'H' : Wrist axis 1.
  - 'I' : Wrist axis 2.
  - 'Z' : Elevation axis.
  - 'R' : Linear access motion speed.

  < Pre-aligner >
  - 'S' : Rotation axis.

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RSPD,<Level>,<Type>,<Axis>,
<Speed>,<MaxSpd>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Level : Transfer speed level (1 byte)
- Type : Speed type (1 byte)
- Axis : Axis (1 byte)
- Speed : Speed data (8 bytes)
  Note) Specified in the range between “00000001” and “99999999” (Resolution: 0.001 [mm/sec], [deg/sec]).
  If a value is less than the specified digits, fill the higher digit(s) with ‘0’ so that the field always has specified digits.
- MaxSpd : Maximum speed data of the specified speed type(8 bytes)
  Note) Specified in the range between “00000001” and “99999999” (Resolution: 0.001 [mm/sec], [deg/sec]).
  If a value is less than the specified digits, fill the higher digit(s) with ‘0’
  so that the field always has specified digits.

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RSPD,<Level>,<Type>,<Axis>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Level : Transfer speed level (1 byte)
- Type : Speed type (1 byte)
- Axis : Axis (1 byte)

RSLV (Transfer Speed Level Reference)

[Function]

References the speed level of the currently set transfer speed (No-wafer speed, With-wafer speed, Low speed, home return speed, Low-speed-area speed) for the specified unit.
If the controller power is cycled, high level speed profile is selected (default).

[Conditions]

None.

[Command Message]

$,<UNo>(,<SeqNo>),RSLV(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SSLV,<Level>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Level : Transfer speed level (1 byte)
  - '1' : Speed level 1.
  - '2' : Speed level 2.
  - '3' : Speed level 3.

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,SSLV(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)

RPOS (Reference Current Position)

[Function]

Reference the current position of the specified unit.

[Conditions]

The specified unit is under servo ON state.

[Command Message]

$,<UNo>(,<SeqNo>),RPOS,<PType>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- PType : Position data type (1 byte)
  - 'R' : Command position.
  - 'F' : Feedback position.

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RPOS,<PType>,<Value1>…,<ValueN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- PType : Position data type (1 byte)
- Value* : Coordinate (8 bytes each, Resolution: 0.001 [mm]/[deg])

  Note) Specified in the range between "-9999999" and "99999999".
  If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.
  A sign is added to the highest digit.

  Note) Response for all axes of specified unit.

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RPOS,<PType>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- PType : Position data type (1 byte)

RSTP (Reference Registered Position)

[Function]

Reference the registered position of the specified unit.

[Conditions]

- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).
- The specified transfer station has been registered.

[Command Message]

$,<UNo>(,<SeqNo>),RSTP,<Mem>,<TrsSt>,<Slot>,<Posture>,<Hand>,<PType>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- Mem : Memory specification (1 byte)
  - 'V' : Volatile memory.
  - 'N' : Non-volatile memory.
- TrsSt : Transfer station (3 bytes)
  - "C01" - "C08" : Cassette stage (the lowest-layer position).
  - "H01" - "H08" : Cassette stage (the highest-layer position).
  - "S01" - "S12" : Transfer stage.
  - "P01" : P/A stage.
- Slot : Slot number (2 bytes)
  <Cassette stage>
  - "01" - "30" : If <TrsSt> is "C01"-"C08"(cassette stage), set the slot.
  - "00" : If <TrsSt> is "H01"-"H08"(cassette stage), recognize the highest-layer position.

  <Transfer stage, Pre-aligner stage>
  - "00" : Fixed value t(because this type of station does not have multiple slots.)

  Note) If value is less than 2 digits, fill the higher digit with '0' so that the field always has 2 digits.
- Posture : Arm Posture (1 byte)
  - 'L' : Left elbow.
  - 'R' : Right elbow.
- Hand : Blade (1 byte)
  - '1' : Blade 1.
  - '2' : Blade 2.
- PType : Position type (1 byte)
  - 'S' : Registered position.
  - 'R' : Ready position.
  - 'M' : Intermediate position.
  - 'B' : Mapping start position.
  - 'E' : Mapping finish position.

  Note) For a mapping start/finish position, slot number and blade number specifications are ignored.

  Note) If <Mem> is N(Non-volatile memory), specified 'S'

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>, RSTP,<Mem>,<TrsSt>,<Slot>,<Posture>,<Hand>,<PType>,
<Value1>…,<ValueN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mem : Memory specification (1 byte)
- TrsSt : Transfer station (3 bytes)
- Slot : Slot number (2 bytes)
- Posture : Arm Posture (1 byte)
- Hand : Blade (1 byte)
- PType : Position type (1 byte)
- Value* : Coordinate (8 bytes each, Resolution: 0.001 [mm]/[deg])
  Note) Specified in the range between “-9999999” and “99999999”.
  If a value is less than the specified digits, fill the higher digit(s) with ‘0’ so that the field always has specified digits.
  A sign is added to the highest digit.
  Note) Response for all axes of the specified unit

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RSTP,<Mem>,<TrsSt>,<Slot>,<Posture>,<Hand>,
<PType>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mem : Memory specification (1 byte)
- TrsSt : Transfer station (3 bytes)
- Slot : Slot number (2 bytes)
- Posture : Arm Posture (1 byte)
- Hand : Blade (1 byte)
- PType : Position type (1 byte)

RSTR (Reference Transfer station information)

[Function]

Reference the transfers stations information.

[Conditions]

None.

[Command Message]

$,<UNo>(,<SeqNo>),RSTR,<Mem>,<TrsSt>,<Item>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- Mem : Memory specification (1 byte)
  - 'V' : Volatile memory.
  - 'N' : Non-volatile memory.
- TrsSt : Transfer station (3 bytes)
  - "C01" - "C08" : Cassette stage.
  - "S01" - "S12" : Transfer stage.
  - "P01" : P/A stage.
- Item : Transfer station information(2 bytes)
  - See the supplementary explanation.

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RSTR,<Mem>,<TrsSt>,<Item>,<Value>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mem : Memory specification (1 byte)
- TrsSt : Transfer station (3 bytes)
- Item : Transfer station information(2 bytes)
- Value : Parameter value (8 bytes)
  - See the supplementary explanation.

  Note) If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.
  A sign is added to the highest digit.

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RSTR,<Mem>,<TrsSt>,<Item>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mem : Memory specification (1 byte)
- TrsSt : Transfer station (3 bytes)
- Item : Transfer station information(2 bytes)

[Supplementary Explanation]

The transfer station information contents are as follows.

Item	Contents	Unit	Setting Range
“00”	Downward offset	0.001 [mm]	“-9999999” - “99999999”
“01”	Upward offset	0.001 [mm]	“-9999999” - “99999999”
“02”	Grip position offset	0.001 [mm]	“-9999999” - “99999999”
“06”	G2/P3 Offset in the extending direction	0.001 [mm]	“-9999999” - “99999999”
“08”	Put downward offset	0.001 [mm]	“-9999999” - “99999999”
“30”	Get operation Move_grip function yes/no	[-]	“00000000”～“00000001”
“31”	Get operation rising pattern	[-]	“00000000”～“00000001”
“40”	Put operation Move_grip function yes/no	[-]	“00000000”～“00000001”
“41”	Put operation dropping pattern	[-]	“00000000”～“00000001”
“10”	Slot numbers	[-]	“00000001”～“00000030”
“30”	Slot pitch(Left elbow, Blade1)	0.001[mm]	“-9999999”～“99999999”
“31”	Slot pitch(Left elbow, Blade2)	0.001[mm]	“-9999999”～“99999999”
“32”	Slot pitch(Right elbow, Blade1)	0.001[mm]	“-9999999”～“99999999”
“33”	Slot pitch(Right elbow, Blade2)	0.001[mm]	“-9999999”～“99999999”

RPRM (Reference Parameter)

[Function]

Reference the parameter values of the specified unit.

[Conditions]

None.

[Command Message]

$,<UNo>(,<SeqNo>),RPRM,<Type>,<PrmNo>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)
- Type : Parameter type (3 bytes)
  - "CRU" : Common real number parameter.
  - "CIU" : Common integer parameter.
  - "URU" : Unit real number parameter.
  - "UIU" : Unit integer parameter.
- PrmNo : Parameter number (4 bytes)

  Note) Specified in the range between "0000" and "9999".
  If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RPRM,<Type>,<PrmNo>,<Value>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Type : Parameter type (3 bytes)
- PrmNo : Parameter number (4 bytes)
- Value : Parameter value (12 bytes, Resolution: 0.0001)

  Note) The range is between "-02147483648" and "002147483647".
  If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.
  A sign is added to the highest digit.

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RPRM,<Type>,<PrmNo>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mem : Memory specification (1 byte)
- Type : Parameter type (3 bytes)
- PrmNo : Parameter number (4 bytes)

RSTS (Reference Status)

[Function]

Reference the various statuses.

[Conditions]

None.

[Command Message]

$,<UNo>(,<SeqNo>),RSTS(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RSTS,<Errcd>,<Status>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Errcd : Error code (4 bytes)
  Note) Responds with the currently occurring error for the specified unit.
  (When there is no error, responds "0000")
- Status : Status information (4 bytes)
  - Refer to the supplementary explanation.

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RSTS(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)

[Supplementary Explanation]

The status information contents are as follows.

< Manipulator >

	Bit #	Contents	Status	
			0	1
Status1	Bit0	Wafer presence status on Blade 1	No Wafer	Wafer present
	Bit1	Wafer presence status on Blade 2		
	Bit2	Blade 1 solenoid chucking command status	Wafer grip	Wafer release
	Bit3	Blade 2 solenoid chucking command status		
Status2	Bit0	Interlock 1 signal status (Optional)	Operation prohibited	Operation permitted
	Bit1	Interlock 2 signal status (Optional)		
	Bit2	Interlock 3 signal status (Optional)		
	Bit3	Interlock 4 signal status (Optional)		
Status3	Bit0	Interlock 5 signal status (Optional)		
	Bit1	Interlock 6 signal status (Optional)		
	Bit2	Interlock 7 signal status (Optional)		
	Bit3	Interlock 8 signal status (Optional)		
Status4	Bit0	Reserved	-	-
	Bit1	Reserved		
	Bit2	Reserved		
	Bit3	Reserved		

< Pre-aligner >

	Bit #	Contents	Status	
			0	1
Status1	Bit0	Vacuum status	No wafer	Wafer present
	Bit1	Wafer presence status (CCD sensor)		
	Bit2	Solenoid chucking command status	Wafer hold	Wafer release
	Bit3	Reserved		
Status2	Bit0	Reserved	-	-
	Bit1	Reserved		
	Bit2	Reserved		
	Bit3	Reserved		
Status3	Bit0	Reserved		
	Bit1	Reserved		
	Bit2	Reserved		
	Bit3	Reserved		
Status4	Bit0	Reserved		
	Bit1	Reserved		
	Bit2	Reserved		
	Bit3	Reserved		

The interlock information is set hexadecimal in order to <Valid1><Valid2><Valid3><Valid4>.
For the editing method of the interlock information, refer to “4.2 Response Message (Controller → Host)”

RERR (Reference Error History)

[Function]

Reference the error history of the specified unit.

[Conditions]

None.

[Command Message]

$,<UNo>(,<SeqNo>),RERR,<Mem>,<HNo>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)
- Mem : Memory (1 byte)
  - 'V' : Volatile memory.
  - 'N' : Non-volatile memory.
- HNo : Error history number (3 bytes)
  - "FFF" : Notify all error history (latest 32 records notified).
  - "000" - "127" : Notifies the details of the specified error history number ("000" is the latest error code).

[Response Message]

(Positive Response Message: Notify all error history)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RERR,<Mem>,<HNo>,<Errcd0>…,<Errcd32>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mem : Memory (1 byte)
- HNo : Error history number (3 bytes)
- Errcd* : Error code (4 bytes)

  Note) The smaller the value of N, the newer the error code.
  If no errors have occurred, "0000" is responded.

(Positive Response Message: Notify specified error history number details)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RERR,<Mem>,<HNo>,<Errcd>,<SvErr>,<SubCd>,

<Date>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mem : Memory (1 byte)
- HNo : Error history number (3 bytes)
- Errcd : Error code (4 bytes)
  - If no errors have occurred, "0000" is responded.
- SvErr: Servo error code (3 bytes)
  - If no errors have occurred, "0000" is responded.
- Subcd : Sub-error code (5 bytes)
  - If no errors have occurred, "0000" is responded.
- Date : Time occurred (12 bytes)

  Note) "Year/Month/Day/Hour/Minute/Second" format responded (2 bytes each).
  If no errors have occurred, "000000000000" is responded.

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RERR,<Mem>,<HNo>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Mem : Memory (1 byte)
- HNo : Error history number (3 bytes)

RMSK (Reference Interlock Monitoring Enable/Disable Information)

[Function]

Reference the interlock monitoring information.

[Conditions]

None.

[Command Message]

$,<UNo>(,<SeqNo>),RMSK(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RMSK,<Valid>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Valid : Interlock information (4 bytes)
  - The contents are described in "SMSK" command.

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RMSK(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)

RVER (Reference Software Version)

[Function]

Reference software version.

[Conditions]

None.

[Command Message]

$,<UNo>(,<SeqNo>),RVER(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RVER,<SysVer>,<SvVer>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- SysVer : System software version (8 bytes)
- SvVer : Servo software version(8 bytes)

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RVER(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)

RMAP (Reference Mapping Result)

[Function]

Reference the mapping result of the specified transfer station.

[Conditions]

- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).
- The mapping operation has completed normally.

[Command Format]

$,<UNo>(,<SeqNo>),RMAP,<TrsSt>,<Slot>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- TrsSt : Transfer station (3 bytes)
  - "C01" - "C08" : Cassette stage.
  - "S01" - "S12" : Transfer stage.
  - "P01" : P/A stage.
- Slot : Slot number (2 bytes)

  <Cassette stage>
  - "01" - "30" : When specifying individual cassette stage slots.
  - "00" : When specifying all slots.

  <Transfer stage, PA stage>
  - "00" : Fixed value

  Note) If value is less than 2 digits, fill the higher digit with '0' so that the field always has 2 digits.

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RMAP,<TrsSt>,<Slot>,
01:<Result1>…,N:<ResultN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- TrsSt : Transfer station (3 bytes)
- Slot : Slot number (2 bytes)
- Result* : Mapping result (2 bytes each)
  - "--" : No wafer detected.
  - "OK" : Wafer inserted correctly.
  - "CW" : Wafer inserted incorrectly (inclined).
  - "DW" : Wafer inserted incorrectly (duplicated).

  Note) Responds with the number of slots of the specified transfer station.

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RMAP,<TrsSt>,<Slot>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- TrsSt : Transfer station (3 bytes)
- Slot : Slot number (2 bytes)

[Usage Example]

(1) Reference the mapping result (Cassette stage 1/ slot 3)
$,1,RMAP,C01,03,0: command format
$,1,00,0000,RMAP,C01,03,03 :OK : response message

(2) Reference the mapping result for all slots(Cassette stage 2)
(slot numbers: 10, slot2: duplicated, slot4/5: inclined, slot8: no wafer)
$,1,RMAP,C02,00: command format
$,1,00,0000,RMAP,C02,00,
01:OK,02:DW,03:OK,04:CW,05:CW,06:OK,07:OK,08:--,09:OK,10:OK: response message

(3) Reference mapping result(Transfer stage 1)
$,1,RMAP,S01,00 : command format
$,1, 00,0000,RMAP,S01,00,01:OK : response message

RMPD (Reference Mapping Data)

[Function]

Reference the mapping data (elevation axis coordinate values during sensor edge startup/stopping) after mapping is executed for the specified transfer station.

[Conditions]

- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).
- Mapping operations have completed normally.

[Command Format]

$,<UNo>(,<SeqNo>),RMPD,<TrsSt>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
- SeqNo : Sequence number (None / 2 bytes)
- TrsSt : Transfer station (3 bytes)
  - "C01" - "C08" : Cassette stage.
  - "S01" - "S12" : Transfer stage.
  - "P01" : P/A stage.
  - "FFF" : The transfer station of the latest mapping motion.

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RMPD,<TrsSt>,
01:<Updata1>,<Dndata1>…,30:<Updata30>,<Dndata30>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- TrsSt : Transfer station (3 bytes)
- Updata* : Sensor edge startup elevation axis coordinates (8 bytes, Resolution: 0.001 [mm])
  Note) The range is between "-9999999" and "99999999".
  If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.
- Dndata* :Sensor edge stopping elevation axis coordinates (8 bytes, Resolution: 0.001[mm])
  Note) The range is between "-9999999" and "99999999".
  If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.
  Note) Updata and Dndata always responds with 30 sets of data.
  Data is responded as it is detected, and if 30 sets are not detected, "00000000" is responded.
  The slot no. and the data no. may not match.

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RMPD,<TrsSt>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- TrsSt : Transfer station (3 bytes)

[Usage Example]

(1) Referece the mapping data of Cassette stage 4(the wafer are slot No.1,15,20)

$,1,RMPD,C04 : command format

$,1,00,0000,RMPD,C04,01:00000000,00000000,02:00000000,00000000,03:00000000,00000000,
04:00000000,00000000,05:00000000,00000000,▪▪▪▪,29:00000000,00000000,30:00000000,00000000
: response message

(2) Reference the mapping data of the transfer station(in case of the latest mapping motion is S1 station)

$,1,RMPD,FFF

$,1,00,0000,RMPD,S01,01:00000000,00000000,02:00000000,00000000,03:00000000,00000000,
04:00000000,00000000,05:00000000,00000000, ▪▪▪▪,29:00000000,00000000,
30:00000000,00000000 : response message

RMCA (Reference Calibration Result for Mapping)

[Function]

Reference the mapping calibration result.

[Conditions]

• Calibration operation for mapping has completed normally.

[Command Message]

$,<UNo>(,<SeqNo>),RMCA,<TrsSt>(,<Sum>)<CR>

- UNo: Unit number (1 byte)
  - '1' : Manipulator
- SeqNo : Sequence number (None / 2 bytes)
- TrsSt : Transfer station (3 bytes)
  - "C01" - "C08" : Cassette stage

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RMCA,<TrsSt>,<Value1>…,<Value6>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- TrsSt : Transfer station (3 bytes)
- Value1 : Lowest-layer slot position (8 bytes, Resolution: 0.001 [mm])
- Value2 : Highest-layer slot position (8 bytes, Resolution: 0.001 [mm])
- Value3 : Wafer width (8 bytes, Resolution: 0.001 [mm])
- Value4 : The threshold value of double insertion (8 bytes, Resolution: 0.001 [mm])
- Value5 : The threshold value of slanting insertion1 (8 bytes, Resolution: 0.001 [mm])
- Value6 : The threshold value of slanting insertion2 (8 bytes, Resolution: 0.001 [mm])

  Note) The range is between "-9999999" and "99999999".
  If value is less than 8 digits, fill the higher digit(s) with '0' so that the field always has 8 digits.

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RMCA,<TrsSt>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- TrsSt : Transfer station (3 bytes)

RALN (Reference Alignment Result)

[Function]

Reference the alignment result.

[Conditions]

- T.P.'s mode selector switch is set to Host mode (if T.P. is connected).
- Alignment has been completed normally.

[Command Format]

$,<UNo>(,<SeqNo>),RALN(,<Posture>,<Hand>)(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '2' : Pre-aligner
- SeqNo : Sequence number (None / 2 bytes)
- Posture : Arm Posture (None / 1 byte)
  - 'L' : Left elbow.
  - 'R' : Right elbow.
- Hand : Blade (None / 1 byte)
  - '1' : Blade 1.
  - '2' : Blade 2.

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RALN(,<Posture>,<Hand>),<Value1>…,<Value10>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Posture : Arm Posture (None / 1 byte)
- Hand : Blade (None / 1 byte)
- Value1 : Wafer eccentricity before alignment operation (8 bytes, Resolution: 0.001 [mm])
- Value2 : Wafer eccentricity direction before alignment operation (8 bytes, Resolution: 0.001 [deg])
- Value3 : Notch/Orientation Flat direction before alignment operation (8 bytes, Resolution: 0.001 [deg])
- Value4 : X direction offset amount before alignment operation (8 bytes, Resolution: 0.001 [mm])
- Value5 : Y direction offset amount before alignment operation (8 bytes, Resolution: 0.001 [mm])
- Value6 : Pre-aligner adjustment angle (8 bytes, Resolution: 0.001 [deg])
- Value7 : Manipulator adjustment amount (8 bytes, Resolution: 0.001 [mm])
- Value8 : Manipulator adjustment angle (8 bytes, Resolution: 0.001 [deg])
- Value9 : X direction offset amount after alignment operation (8 bytes, Resolution: 0.001 [mm])
- Value10 : Y direction offset amount after alignment operation (8 bytes, Resolution: 0.001 [mm])
  - Refer to the Supplementary Explanation

  Note) If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.

(Negative Response Message)

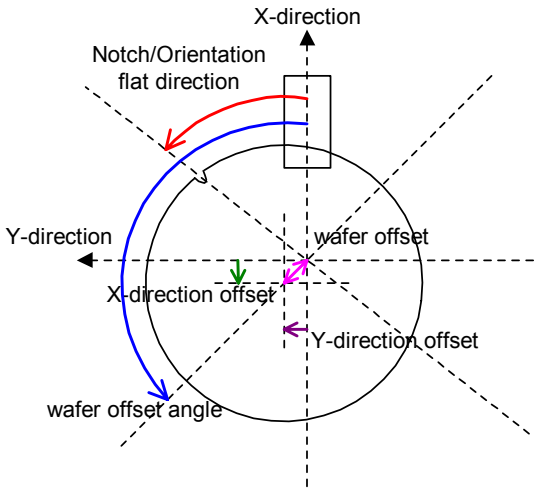
$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RALN(,<Posture>,<Hand>)(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Posture : Arm Posture (None / 1 byte)
- Hand : Blade (None / 1 byte)

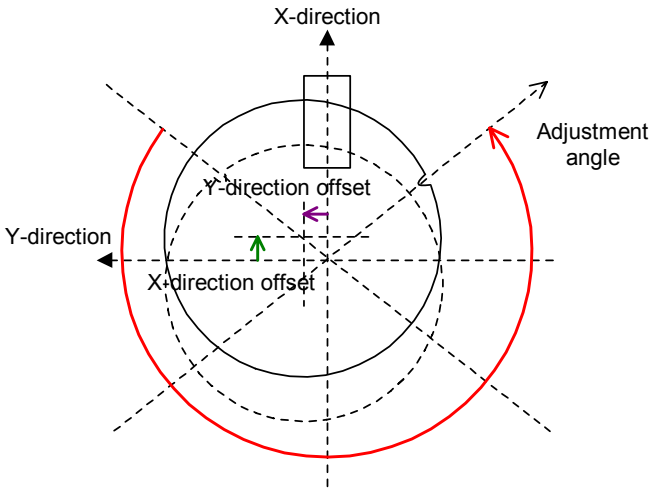
[Supplementary Explanation]

The alignment result are as follows:

Before alignment motion
(The state that the manipulator has put the wafer on the PA stage)



After alignment motion
(The state that the Pre-aligner has aligned the notch)



- Wafer offset amount: The distance between Pre-aligner rotation center and the wafer center
- Wafer offset angle: The wafer offset angle from CCD sensor
- Notch/Orentation flat direction: The Notch(Orientation flat) angle from CCD sensor
- X-direction offset amount: The X-direction offset amount between the Pre-aligner rotation center and the wafer center.
- Y-direction offset amount: The Y-directionoffset amount between the Pre-aligner rotation center and the wafer center.
- Pre-aligner adjustment angle: The angle between the angle before alignment operation and the angle after alignment operationafter alignment motion.
- Manipulator adjustment amount: The extension axis adjustment amout for centering the wafer when the manipulator get the wafer from the PA stage.
- Manipulator adjustment angle: The rotation axis adjustment angle for centering the wafer when the manipulator get the wafer from the PA stage.

For the manipulator offset amount and offset angle, refer to “Figure 9.1 Alignment Adjustment Outline(When Pre-aligner Station Access is in Link Mode)”

RACA (Reference Calibration Result for Alignment)

[Function]

Reference the alignment calibration result.

[Conditions]

Calibration for alignment has finished normally.

[Command Message]

$,<UNo>(,<SeqNo>),RACA(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RACA(,<Posture>,<Hand>),
<Value1>,<Value2>,<Value3>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Posture : Arm Posture (None / 1 byte)
- Hand : Blade (None / 1 byte)
- Value1 : Calibration angle (8 bytes, Resolution: 0.001 [deg])
- Value2 : Manipulator advance angle (8 bytes, Resolution: 0.001 [deg])
- Value3 : Manipulator swivel center to pre-aligner swivel center distance
  (8 bytes, Resolution: 0.001 [mm])

  Note) If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RACA(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)

RCCD (Reference Light Amount and CCD Sensor Values)

[Function]

Reference the pre-aligner's light amount and CCD sensor values.

[Conditions]

None.

[Command Message]

$,<UNo>(,<SeqNo>),RCCD(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)

[Response Message]

(Positive Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RCCD,<Light>,<Ccd>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Light : Light amount value (5 bytes)
  Note) Specified in the range between "00000" and "99999".
  If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.
- Ccd : CCD sensor value (5 bytes)
  Note) Specified in the range between "00000" and "99999".
  If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.

(Negative Response Message)

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RCCD(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)

RLOG (Reference Log Information)

[Function]

Reference the specified log information.

[Conditions]

None

[Command Message]

$,<UNo>(,<SeqNo>),RLOG,<LogNo>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)
- LogNo : Log number (2 bytes)
  - Refer to the supplementary explanation .

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RLOG,<LogNo>,<LogDat>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- LogNo : Log number (2 bytes)
- LogDat : Log data (Variable byte)
  - Refer to the supplementary explanation.

[Supplementary Explanation]

Each type of log's information is as shown below.

Log Number (LogNo)	Contents	Log Data (LogDat)
"00"	Power On times (6 bytes) (Accumulated no. of times)	• Specified in the range between "000000" and "999999".
"01"	Servo On times (6 bytes) (Accumulated no. of times)	• Specified in the range between "000000" and "999999".
"02"	Brake times (6 bytes) (Accumulated no. of times)	• Specified in the range between "000000" and "999999".
"10"	Powered time (13 bytes) (Accumulated time)	<Day>,<Hour>,<Min>,<Sec> • Specified in the range between "0000,00,00,00" and "9999,23,59,59".
"11"	Accumulated driving time (83 bytes) (Accumulated time × 6 axes)	<Ax1>,<Ax2>,<Ax3>,<Ax4>,<Ax5>,<Ax6>, Contents of <AxN> <Day>,<Hour>,<Min>,<Sec> • Specified in the range between "0000,00,00,00" and "9999,23,59,59".
"12"	Servo On time (83 bytes) (Accumulated time × 6 axes)	<Ax1>,<Ax2>,<Ax3>,<Ax4>,<Ax5>,<Ax6> Contents of <AxN> <Day>,<Hour>,<Min>,<Sec> • Specified in the range between "0000,00,00,00" and "9999,23,59,59".
"20"	Latest Power On time (14 bytes) (Time and date data)	<Mon>,<Day>,<Hour>,<Min>,<Sec> • Specified in the range between "00,00,00,00,00" and "99,99,23,59,59".
"21"	Power On time log (479 bytes) (Time and date data × 16records)	<Log1>,<Log2>,<Log3>,<Log4>…,<Log16> Contents of <LogN> <Mon>,<Day>,<Hour>,<Min>,<Sec> Specified in the range between "00,00,00,00,00" and "99,99,23,59,59".

RSTN (Reference the Reference Position)

[Function]

Reference the reference positions.

[Conditions]

None

[Command Message]

$,<UNo>(,<SeqNo>),RSTN(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner (Edge-grip pre-aligner).
- SeqNo : Sequence number (None / 2 bytes)

[Response Message]

$,<UNo>(,<SeqNo>),<Sts>,<Ackcd>,RSTN,<Value1>…,<ValueN>(,<Sum>)<CR>

- UNo : Unit number (1 byte)
- SeqNo : Sequence number (None / 2 bytes)
- Sts : Status (2 bytes)
- Ackcd : Response code (4 bytes)
- Value* : Reference position of the encoder value (12 bytes each)
  - Value1 : Arm 1.
  - Value2 : Arm 2.
  - Value3 : Wrist axis 1.
  - Value4 : Wrist axis 2.
  - Value5 : Elevation axis.

  Note) Specified in the range between "-02147483648" and "002147483647".
  If a value is less than the specified digits, fill the higher digit(s) with '0' so that the field always has specified digits.
  A sign is added to the highest digit.

  Note) The number of "ValueN" depends on the unit type. Set as many axis numbers as the specified unit has.

ACKN (Notify End-of-Execution Check)

[Function]

Notifies the controller's end-of-execution check.
For this command, the controller does not send a response.

[Conditions]

None

[Command Message]

$,<UNo>(,<SeqNo>),ACKN(,<Sum>)<CR>

- UNo : Unit number (1 byte)
  - '1' : Manipulator.
  - '2' : Pre-aligner.
- SeqNo : Sequence number (None / 2 bytes)

[Response Message]

None