



Machine Automation Controller

**Robot-supported NJ Series
CPU Unit**

User's Manual

NJ501-4□□□



© OMRON, 2012

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, mechanical, electronic, photocopying, recording, or otherwise, without the prior written permission of OMRON.

No patent liability is assumed with respect to the use of the information contained herein. Moreover, because OMRON is constantly striving to improve its high-quality products, the information contained in this manual is subject to change without notice. Every precaution has been taken in the preparation of this manual. Nevertheless, OMRON assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained in this publication.

Introduction

Thank you for purchasing an NJ-series CPU Unit.

This manual contains information that is necessary to use the NJ-series CPU Unit. Please read this manual and make sure you understand the functionality and performance of the NJ-series CPU Unit before you attempt to use it in a control system.

Keep this manual in a safe place where it will be available for reference during operation.

Also, this manual is mainly described about 4xxx's features, so please see this with NJ501-xxxx's manual indicated in "Related Manuals".

Intended Audience

This manual is intended for the following personnel, who must also have knowledge of electrical systems (an electrical engineer or the equivalent).

- Personnel in charge of introducing FA systems.
- Personnel in charge of designing FA systems.
- Personnel in charge of installing and maintaining FA systems.
- Personnel in charge of managing FA systems and facilities.

For programming, this manual is intended for personnel who understand the programming language specifications in international standard IEC 61131-3 or Japanese standard JIS B3503.

Applicable Products

This manual covers the following products.

- Robot-supported NJ-series CPU Units
 - NJ501-4500
 - NJ501-4400
 - NJ501-4310
 - NJ501-4300

CONTENTS

Introduction.....	3
Intended Audience.....	3
Applicable Products.....	3
CONTENTS.....	4
Read and Understand this Manual.....	7
Safety Precautions.....	10
Definition of Precautionary Information.....	10
Symbols.....	11
Precautions for Safe Use.....	15
Precautions for Correct Use.....	22
Regulations and Standards.....	25
Conformance to EC Directives.....	25
Conformance to Shipbuilding Standards.....	26
Usage in the robot system.....	26
Trademarks.....	26
Software Licenses and Copyrights.....	26
Unit Versions.....	27
Unit Versions and Notation.....	27
Unit Versions and Sysmac Studio Versions.....	29
Unit Version Notation.....	29
Related Manuals.....	30
Revision History.....	31
Terminology.....	32
1 Introduction to Robot-supported NJ Series.....	33
1.1 Features.....	33
1.2 System Configuration.....	33
1.3 Startup Procedure.....	36
1.4 Specifications.....	37
1.4.1 Function Specifications.....	37
1.4.2 Performance Specifications.....	37
2 Introduction to Robot Control.....	39
3 Functions.....	40
3.1 Controllable Robot Type.....	40
3.2 Coordinate Systems.....	40
3.2.1 Outline.....	40
3.2.2 Axis Coordinate System (ACS).....	45
3.2.3 Cartesian Coordinate Systems.....	45
3.3 Special Robot Functions.....	46
3.3.1 Kinematics Setting.....	46

3.3.2	Workspace Check.....	48
3.3.3	User Coordinate System.....	55
3.3.4	Monitoring.....	56
3.3.5	Time-specified Absolute Positioning Command.....	56
3.3.6	Synchronization with Conveyor.....	57
3.3.7	Inverse Kinematics.....	59
3.3.8	Robot Tool.....	60
3.3.9	Robot Jog.....	62
3.3.10	Robot maximum interpolation velocity and acceleration check.....	62
3.3.11	Multi-Execution (Buffered).....	66
3.3.12	Multi-Execution (Blending).....	66
3.3.13	Trajectory types of time-specified move.....	70
4	Robot Function Settings.....	72
4.1	Sysmac Studio Robot Option.....	72
4.1.1	Procedures of Registering Robot Option.....	72
4.1.2	Procedures of Unregistering Robot Option.....	73
4.2	Setting of Group to Register the Robot (Setting on Sysmac Studio).....	76
4.3	Setting of Link Length and Other Parameters for Robot (Setting on User Program).....	77
5	Homing.....	78
6	Wiring/Settings by Sysmac Studio.....	79
7	Instructions.....	80
7.1	Outline.....	80
7.1.1	Instruction List.....	80
7.1.2	Unavailable Instructions.....	81
7.1.3	Starting and Stopping a Robot.....	82
7.1.4	Axes Group Variables.....	82
7.2	How to specify a target position.....	83
7.2.1	Position input variable.....	83
7.2.2	Control of rotation.....	83
7.3	When using with Robot Ver. 1.01 a project which was created in Robot Ver. 1.00.....	85
7.4	Details.....	85
	MC_SetKinTransform <Robot Ver 1.00>.....	86
	MC_DefineCoordSystem.....	91
	MC_GroupMon <Robot Ver 1.00>.....	95
	MC_MoveTimeAbsolute <Robot Ver 1.00>.....	100
	MC_SyncLinearConveyor <Robot Ver 1.00>.....	105
	MC_SyncOut.....	116
	MC_InverseKin <Robot Ver 1.00>.....	122
	MC_GroupSyncMoveAbsolute.....	125
	MC_SetKinTransform <Robot Ver 1.01>.....	126
	MC_DefineToolTransform <Robot Ver 1.01>.....	133
	MC_GroupMon <Robot Ver 1.01>.....	137

MC_MoveTimeAbsolute <Robot Ver 1.01>.....	143
MC_SyncLinearConveyor <Robot Ver 1.01>	150
MC_InverseKin <Robot Ver 1.01>.....	163
MC_RobotJog <Robot Ver 1.01>	167
Multi-execution of Motion Control Instructions	173
■ Aborting	173
■ Buffered	174
■ Blending	174
State Transition of Robot Instructions	175
8 Sample Programming.....	176
9 Troubleshooting.....	178
9.1 Error Table	178
9.2 Error Code Details	183

Read and Understand this Manual

Please read and understand this manual before using the products. Please consult your OMRON representative if you have any questions or comments.

Warranty and Limitations of Liability

WARRANTY

OMRON's exclusive warranty is that the products are free from defects in materials and workmanship for a period of one year (or other period if specified) from date of sale by OMRON.

OMRON MAKES NO WARRANTY OR REPRESENTATION, EXPRESS OR IMPLIED, REGARDING NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR PARTICULAR PURPOSE OF THE PRODUCTS. ANY BUYER OR USER ACKNOWLEDGES THAT THE BUYER OR USER ALONE HAS DETERMINED THAT THE PRODUCTS WILL SUITABLY MEET THE REQUIREMENTS OF THEIR INTENDED USE. OMRON DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED.

LIMITATIONS OF LIABILITY

OMRON SHALL NOT BE RESPONSIBLE FOR SPECIAL, INDIRECT, OR CONSEQUENTIAL DAMAGES, LOSS OF PROFITS OR COMMERCIAL LOSS IN ANY WAY CONNECTED WITH THE PRODUCTS, WHETHER SUCH CLAIM IS BASED ON CONTRACT, WARRANTY, NEGLIGENCE, OR STRICT LIABILITY.

In no event shall the responsibility of OMRON for any act exceed the individual price of the product on which liability is asserted.

IN NO EVENT SHALL OMRON BE RESPONSIBLE FOR WARRANTY, REPAIR, OR OTHER CLAIMS REGARDING THE PRODUCTS UNLESS OMRON'S ANALYSIS CONFIRMS THAT THE PRODUCTS WERE PROPERLY HANDLED, STORED, INSTALLED, AND MAINTAINED AND NOT SUBJECT TO CONTAMINATION, ABUSE, MISUSE, OR INAPPROPRIATE MODIFICATION OR REPAIR.

Application Considerations

SUITABILITY FOR USE

OMRON shall not be responsible for conformity with any standards, codes, or regulations that apply to the combination of products in the customer's application or use of the products.

At the customer's request, OMRON will provide applicable third party certification documents identifying ratings and limitations of use that apply to the products. This information by itself is not sufficient for a complete determination of the suitability of the products in combination with the end product, machine, system, or other application or use.

The following are some examples of applications for which particular attention must be given. This is not

intended to be an exhaustive list of all possible uses of the products, nor is it intended to imply that the uses listed may be suitable for the products:

- Outdoor use, uses involving potential chemical contamination or electrical interference, or conditions or uses not described in this manual.
- Nuclear energy control systems, combustion systems, railroad systems, aviation systems, medical equipment, amusement machines, vehicles, safety equipment, and installations subject to separate industry or government regulations.
- Systems, machines, and equipment that could present a risk to life or property.

Please know and observe all prohibitions of use applicable to the products.

NEVER USE THE PRODUCTS FOR AN APPLICATION INVOLVING SERIOUS RISK TO LIFE OR PROPERTY WITHOUT ENSURING THAT THE SYSTEM AS A WHOLE HAS BEEN DESIGNED TO ADDRESS THE RISKS, AND THAT THE OMRON PRODUCTS ARE PROPERLY RATED AND INSTALLED FOR THE INTENDED USE WITHIN THE OVERALL EQUIPMENT OR SYSTEM.

PROGRAMMABLE PRODUCTS

OMRON shall not be responsible for the user's programming of a programmable product, or any consequence thereof.

Disclaimers

CHANGE IN SPECIFICATIONS

Product specifications and accessories may be changed at any time based on improvements and other reasons.

It is our practice to change model numbers when published ratings or features are changed, or when significant construction changes are made. However, some specifications of the products may be changed without any notice. When in doubt, special model numbers may be assigned to fix or establish key specifications for your application on your request. Please consult with your OMRON representative at any time to confirm actual specifications of purchased products.

DIMENSIONS AND WEIGHTS

Dimensions and weights are nominal and are not to be used for manufacturing purposes, even when tolerances are shown.

PERFORMANCE DATA

Performance data given in this manual is provided as a guide for the user in determining suitability and does not constitute a warranty. It may represent the result of OMRON's test conditions, and the users must correlate it to actual application requirements. Actual performance is subject to the OMRON Warranty and Limitations of Liability.

ERRORS AND OMISSIONS


The information in this manual has been carefully checked and is believed to be accurate; however, no responsibility is assumed for clerical, typographical, or proofreading errors, or omissions.


Safety Precautions

Definition of Precautionary Information

The following notation is used in this manual to provide precautions required to ensure safe usage of an NJ-series Controller. The safety precautions that are provided are extremely important to safety. Always read and heed the information provided in all safety precautions.

The following notation is used.

 WARNING	Indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury. Additionally, there may be severe property damage.
--	--

 Caution	Indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury, or property damage.
--	--



Precautions for Safe Use

Indicates precautions on what to do and what not to do to ensure safe usage of the product.



Precautions for Correct Use

Indicates precautions on what to do and what not to do to ensure proper operation and performance.

Symbols



The circle and slash symbol indicates operations that you must not do.
The specific operation is shown in the circle and explained in text.
This example indicates prohibiting disassembly.



The triangle symbol indicates precautions (including warnings).
The specific operation is shown in the triangle and explained in text.
This example indicates a precaution for electric shock.



The triangle symbol indicates precautions (including warnings).
The specific operation is shown in the triangle and explained in text.
This example indicates a general precaution.



The filled circle symbol indicates operations that you must do.
The specific operation is shown in the circle and explained in text.
This example shows a general precaution for something that you must do.

WARNING

During Power Supply

Do not touch any of the terminals or terminal blocks while the power is being supplied. Doing so may result in electric shock.



Do not attempt to take any Unit apart. In particular, high-voltage parts are present in the Power Supply Unit while power is supplied or immediately after power is turned OFF. Touching any of these parts may result in electric shock. There are sharp parts inside the Unit that may cause injury.



Fail-safe Measures

Provide safety measures in external circuits to ensure safety in the system if an abnormality occurs due to malfunction of the CPU Unit, other Units, or slaves or due to other external factors affecting operation. Not doing so may result in serious accidents due to incorrect operation.



Emergency stop circuits, interlock circuits, limit circuits, and similar safety measures must be provided in external control circuits.



The Controller outputs may remain ON or OFF due to deposition or burning of the output relays or destruction of the output transistors. As a countermeasure for such problems, external safety measures must be provided to ensure safe operation of the system.



The CPU Unit will turn OFF all outputs from Basic Output Units in the following cases.

- If an error occurs in the power supply
- If the power supply connection becomes faulty
- If a CPU watchdog timer error or CPU reset occurs
- If a major fault level Controller error occurs
- While the CPU Unit is on standby until RUN mode is entered after the power is turned ON



External safety measures must be provided to ensure safe operation of the system even if the outputs turn OFF.

If external power supplies for slaves or other devices are overloaded or short-circuited, the voltage will drop, outputs will turn OFF, and the system may be unable to read inputs. Provide external safety measures in controls with monitoring of external power supply voltage as required so that the system operates safely in such a case.





WARNING

Fail-safe Measures

Unintended outputs may occur when an error occurs in variable memory or in memory used for CJ-series Units. As a countermeasure for such problems, external safety measures must be provided to ensure safe operation of the system.



Provide measures in the communications system and user program to ensure safety in the overall system even if errors or malfunctions occur in data link communications or remote I/O communications.



If there is interference in remote I/O communications or if a major fault level error occurs, output status will depend on the products that are used. Confirm the operation that will occur when there is interference in communications or a major fault level error, and implement safety measures. Correctly set all of the EtherCAT slaves.



The NJ-series Controller continues normal operation for a certain period of time when a momentary power interruption occurs. This means that the NJ-series Controller may receive incorrect signals from external devices that are also affected by the power interruption. Accordingly, take suitable actions, such as external fail-safe measures and interlock conditions, to monitor the power supply voltage of the external device as required.



You must take fail-safe measures to ensure safety in the event of incorrect, missing, or abnormal signals caused by broken signal lines, momentary power interruptions, or other causes. Not doing so may result in serious accidents due to incorrect operation.



Voltage and Current Inputs

Make sure that the voltages and currents that are input to the Units and slaves are within the specified ranges. Inputting voltages or currents that are outside of the specified ranges may cause accidents or fire.



Downloading

Always confirm safety at the destination before you transfer a user program, configuration data, setup data, device variables, or values in memory used for CJ-series Units from the Sysmac Studio. The devices or machines may perform unexpected operation regardless of the operating mode of the CPU Unit.



Caution

Application

Do not touch any Unit when power is being supplied or immediately after the power supply is turned OFF. Doing so may result in burn injury.



Wiring

Be sure that all terminal screws and cable connector screws are tightened to the torque specified in the relevant manuals. The loose screws may result in fire or malfunction.



Online Editing

Execute online editing only after confirming that no adverse effects will be caused by deviations in the timing of I/O. If you perform online editing, the task execution time may exceed the task period, I/O may not be refreshed with external devices, input signals may not be read, and output timing may change.



Precaution on Error Message That Says an Instruction May Cause Unintended Operation

Instructions may result in unexpected operation and affect the system if you clear the selection of the *Detect an error when an in-out variable is passed to specific instruction argument* Check Box in the Program Check Area. Always confirm that the conditions for use that are given in the *NJ-series Instructions Reference Manual* (Cat. No. W502) are met before you clear the selection of this check box.



Version Information

This error message is displayed by and the above option setting is available on Sysmac Studio version 1.02.

Precautions for Safe Use

● Disassembly and Dropping

- Do not attempt to disassemble, repair, or modify any Units. Doing so may result in malfunction or fire.
- Do not drop any Unit or subject it to abnormal vibration or shock. Doing so may result in Unit malfunction or burning.

● Mounting

- The sliders on the tops and bottoms of the Power Supply Unit, CPU Unit, I/O Units, and other Units must be completely locked (until they click into place) after connecting the adjacent Unit connectors.

● Installation

- Always connect to a ground of 100 Ω or less when installing the Units. A ground of 100 Ω or less must be installed when shorting the GR and LG terminals on the Power Supply Unit.

● Wiring

- Follow the instructions in the NJ-series CPU Unit Hardware User's manual (Cat. No. W500) to correctly perform wiring.

Double-check all wiring and switch settings before turning ON the power supply.

- Use crimp terminals for wiring.

Do not connect bare stranded wires directly to terminals.

- Do not pull on the cables or bend the cables beyond their natural limit.

Do not place heavy objects on top of the cables or other wiring lines. Doing so may break the cables.

- Mount terminal blocks and connectors only after checking the mounting location carefully.

Be sure that the terminal blocks, expansion cables, and other items with locking devices are properly locked into place.

- Always remove any dustproof labels that are on the top of the Units when they are shipped before you turn ON the power supply. If the labels are not removed, heat will accumulate and malfunctions may occur.

- Before you connect a computer to the CPU Unit, disconnect the power supply plug of the computer from the AC outlet. Also, if the computer has an FG terminal, make the connections so that the FG terminal has the same electrical potential as the GR terminal on the Power Supply Unit. A difference in electrical potential between the computer and Controller may cause failure or malfunction.

- If the external power supply to an Output Unit or slave has polarity, connect it with the correct polarity. If the polarity is reversed, current may flow in the reverse direction and damage the connected devices regardless of the operation of the Controller.

● Power Supply Design

- Do not exceed the rated supply capacity of the Power Supply Units in the NJ-series Controller. The rated supply capacities are given in the NJ-series CPU Unit Hardware User's Manual (Cat. No.W500).

If the capacity is exceeded, operation may stop, malfunctions may occur, or data may not be backed up normally for power interruptions.

Use NJ-series Power Supply Units for both the NJ-series CPU Rack and Expansion Racks.

Operation is not possible if a CJ-series Power Supply Unit is used with an NJ-series CPU Unit or an NJ-series Power Supply Unit is used with a CJ-series CPU Unit.

- Do not apply voltages or connect loads to the Output Units or slaves in excess of the maximum ratings.
- Surge current occurs when the power supply is turned ON. When selecting fuses or breakers for external circuits, consider the above precaution and allow sufficient margin in shut-off performance. Refer to the relevant manuals for surge current specifications. Refer to the NJ-series CPU Unit Hardware User's Manual (Cat. No. W500) for surge current specifications.
- If the full dielectric strength voltage is applied or turned OFF using the switch on the tester, the generated impulse voltage may damage the Power Supply Unit. Use the adjustment on the tester to gradually increase and decrease the voltage.
- Apply the voltage between the Power Supply Unit's L1 or L2 terminal and the GR terminal when testing insulation and dielectric strength.
- Do not supply AC power from an inverter or other device with a square-wave output. Internal temperature rise may result in smoking or burning. Always input a sinusoidal wave with the frequency that is given in the NJ-series CPU Unit Hardware User's Manual (Cat. No. W500).
- Install external breakers and take other safety measures against short-circuiting in external wiring.

- **Turning ON the Power Supply**

- It takes up to approximately 10 to 20 s to enter RUN mode after the power is turned ON. During that time, outputs will be OFF or will be the values specified in the Unit or slave settings, and external communications cannot be performed. Use the RUN output on the Power Supply Unit, for example, to implement fail-safe circuits so that external devices do not operate incorrectly.
- Configure the external circuits so that the power supply to the control system turns ON only after the power supply to the Controller has turned ON. If the power supply to the Controller is turned ON after the control power supply, temporary errors may result in incorrect control system signals because the output terminals on Output Units may momentarily turn ON when power supply is turned ON to the Controller.

- **Actual Operation**

- Check the user program, data, and parameter settings for proper execution before you use them for actual operation.

- **Turning OFF the Power Supply**

- Never turn OFF the power supply to the Controller when the BUSY indicator is flashing. While the BUSY indicator is lit, the user program and settings in the CPU Unit are being backed up in the built-in non-volatile memory. This data will not be backed up correctly if the power supply is turned OFF. Also, a major fault level Controller error will occur the next time you start operation, and operation will stop.
- Do not turn OFF the power supply or remove the SD Memory Card while SD Memory Card access is in progress (i.e., while the SD BUSY indicator flashes). Data may become corrupted, and the Controller will not operate correctly if it uses corrupted data. To remove the SD Memory Card from the CPU Unit while the power supply is ON, press the SD Memory Card power supply switch and wait for the SD BUSY indicator to turn OFF before you remove the SD Memory Card.

- Do not disconnect the cable or turn OFF the power supply to the Controller when downloading data or the user program from Support Software.
- Always turn OFF the power supply to the Controller before you attempt any of the following.
 - Mounting or removing I/O Units or the CPU Unit
 - Assembling the Units
 - Setting DIP switches or rotary switches
 - Connecting cables or wiring the system
 - Connecting or disconnecting the connectors

The Power Supply Unit may continue to supply power to the rest of the Controller for a few seconds after the power supply turns OFF. The PWR indicator is lit during this time. Confirm that the PWR indicator is not lit before you perform any of the above.

● Operation

- Confirm that no adverse effect will occur in the system before you attempt any of the following.
 - Changing the operating mode of the CPU Unit (including changing the setting of the Operating Mode at Startup)
 - Changing the user program or settings
 - Changing set values or present values
 - Forced refreshing
- Always sufficiently check the safety at the connected devices before you change the settings of an EtherCAT slave or Special Unit.
- If two different function modules are used together, such as when you use CJ-series Basic Output Units and EtherCAT slave outputs, take suitable measures in the user program and external controls to ensure that safety is maintained in the controlled system if one of the function modules stops. The relevant outputs will stop if a partial fault level error occurs in one of the function modules.
- Always confirm safety at the connected equipment before you reset Controller errors with an event level of partial fault or higher for the EtherCAT Master Function Module.

When the error is reset, all slaves that were in any state other than Operational state due to a Controller error with an event level of partial fault or higher (in which outputs are disabled) will go to Operational state and the outputs will be enabled.

Before you reset all errors, confirm that no Controller errors with an event level of partial fault have occurred for the EtherCAT Master Function Module.

- Always confirm safety at the connected equipment before you reset Controller errors for a CJ-series Special Unit. When a Controller error is reset, the Unit where the Controller error with an event level of observation or higher will be restarted.

Before you reset all errors, confirm that no Controller errors with an event level of observation or higher have occurred for the CJ-series Special Unit. Observation level events do not appear on the Controller Error Tab Page, so it is possible that you may restart the CJ-series Special Unit without intending to do so.

You can check the status of the `_CJB_UnitErrSta[0,0]` to `_CJB_UnitErrSta[3,9]` error status variables on a Watch Tab Page to see if an observation level Controller error has occurred.

● Battery Backup

- The user program and initial values for the variables are stored in non-volatile memory in the CPU Unit. The present values of variables with the Retain attribute and the values of the Holding, DM, and EM Areas in the memory used for CJ-series Units are backed up by a Battery. If the Battery is not connected or the Battery is exhausted, the CPU Unit detects a Battery-backup Memory Check Error. If that error is detected, variables with a Retain attribute are set to their initial values and the Holding, DM, and EM Areas in memory used for CJ-series Units are cleared to all zeros. Perform thorough verifications and provide sufficient measures to ensure that the devices perform safe operation for the initial values of the variables with Retain attributes and the resulting operation.

● Debugging

- Forced refreshing ignores the results of user program execution and refreshes I/O with the specified values. If forced refreshing is used for inputs for which I/O refreshing is not supported, the inputs will first take the specified values, but they will then be overwritten by the user program. This operation differs from the force-set/reset functionality of the CJ-series PLCs.

- You cannot upload or download information for forced refreshing with the Sysmac Studio.

After downloading data that contains forced refreshing, change to RUN mode and then use the Sysmac Studio to perform the operation for forced refreshing.

Depending on the difference in the forced status, the control system may operate unexpectedly.

- Do not specify the same address for the AT specification for more than one variable.

Doing so would allow the same entity to be accessed with different variable names, which would make the user program more difficult to understand and possibly cause programming mistakes.

● General Communications

- When you use data link communications, check the error information that is given in `_ErrSta` (Controller Error Status) to make sure that no error has occurred in the source device. Create a user program that uses reception data only when there is no error in the source device. If there is an error in the source device, the data for the data link may contain incorrect values.

- Unexpected operation may result if inappropriate data link tables are set. Even if appropriate data link tables have been set, confirm that the controlled system will not be adversely affected before you transfer the data link tables. The data links start automatically after the data link tables are transferred.

- All CPU Bus Units are restarted when routing tables are transferred from Support Software to the CPU Unit. Confirm that the system will not be adversely affected by restarting before you transfer the routing tables.

- Tag data links will stop between related nodes while tag data link parameters are transferred during Controller operation. Confirm that the system will not be adversely affected before you transfer the tag data link parameters.

● EtherNet/IP Communications

- All related EtherNet/IP nodes are reset when you transfer settings for the built-in EtherNet/IP port (including IP addresses and tag data links settings). Confirm that the system will not be adversely

affected by resetting nodes before you transfer the settings.

- If EtherNet/IP tag data links (cyclic communications) are used with a repeating hub, the communications load on the network will increase. This will increase collisions and may prevent stable communications. Do not use repeating hubs on networks where tag data links are used. Use an Ethernet switch instead.

- EtherCAT Communications

- Make sure that the communications distance, number of nodes connected, and method of connection for EtherCAT are within specifications.

Do not connect EtherCAT communications to EtherNet/IP, a standard in-house LAN, or other networks. An overload may cause the network to fail or malfunction.

- Malfunctions or unexpected operation may occur for some combinations of EtherCAT revisions of the master and slaves. If you disable the revision check in the network settings, use the Sysmac Studio to check the slave revision settings in the master and the actual slave revisions, and then make sure that functionality is compatible in the slave manuals or other references. You can check the actual slave revisions from the Sysmac Studio or on slave nameplates.

- After you transfer the user program, the CPU Unit is restarted and communications with the EtherCAT slaves are cut off. During that period, the slave outputs behave according to the slave settings. The time that communications are cut off depends on the EtherCAT network configuration. If the EtherCAT network configuration contains only OMRON EtherCAT slaves, communications are cut off for a maximum of 45 seconds.

Before you transfer the user program, confirm that the system will not be adversely affected.

- If the Fail-soft Operation parameter is set to stop operation, process data communications will stop for all slaves when an EtherCAT communications error is detected in a slave. For this reason, if Servo Drives are connected, the Servos for all axes will be turned OFF. Make sure that the Fail-soft Operation parameter setting results in safe operation when a device error occurs.

- EtherCAT communications are not always established immediately after the power supply is turned ON. Use the system-defined variables in the user program to confirm that communications are established before attempting control operations.

- If frames sent to EtherCAT slaves are lost due to noise or other causes, slave I/O data is not communicated, and the intended operation is sometimes not achieved. If noise countermeasures are required, use the `_EC_InDataInvalid` (Input Data Disable) system-defined variable as an interlock condition in the user program. Refer to the NJ-series CPU Unit Built-in EtherCAT Port User's Manual (Cat. No. W505) for details. The slave outputs behave according to the slave settings. Refer to the manuals for the slaves for details.

- When an EtherCAT slave is disconnected, communications will stop and control of the outputs will be lost not only for the disconnected slave, but for all slaves connected after it. Confirm that the system will not be adversely affected before you disconnect a slave.

- If you disconnect the cable from an EtherCAT slave to disconnect it from the network, any current communications frames may be lost. If frames are lost, slave I/O data is not communicated, and the intended operation is sometimes not achieved. Perform the following processing for a slave that needs to be replaced.

Program the `_EC_InDataInvalid` (Input Data Disable) system-defined variable as an interlock condition. Set the Impermissible Number of Continuous Timeouts setting in the EtherCAT master to at least 2. Refer to the NJ-series CPU Unit Built-in EtherCAT Port User's Manual (Cat. No. W505) for details.

● Motion Control

- Confirm the axis number carefully before you perform an MC Test Run.
- The motor is stopped if communications are interrupted between the Sysmac Studio and the CPU Unit during an MC Test Run. Connect the communications cable between the computer and CPU Unit securely and confirm that the system will not be adversely affected before you perform an MC Test Run.
- The motor is stopped if communications are interrupted between the Sysmac Studio and the CPU Unit during an MC Test Run. Connect the communications cable between the computer and CPU Unit securely and confirm that the system will not be adversely affected before you perform an MC Test Run.
- Always execute the Save Cam Table instruction if you change any of the cam data from the user program in the CPU Unit or from the Sysmac Studio. If the cam data is not saved, the previous condition will be restored when the power is turned ON again, possibly causing unexpected machine operation.
- The positive drive prohibit input (POT), negative drive prohibit input (NOT), and home proximity input (DEC) of the Servo Drive are used by the MC Function Module as the positive limit input, negative limit input, and home proximity input. Make sure that the signal widths for all of these input signals are longer than the control period of the MC Function Module. If the input signal widths are shorter than the control period, the MC Function Module may not be able to detect the input signals, resulting in incorrect operation.
- Specify the setting of direction in the controller as same as the rotating direction of the Servo Drive. Otherwise, the robot may cause unexpected operation.
- Choose the correct combination of Kinematics type and Workspace type which are set by `MC_SetKinTransform`.
Confirm if the workspace check works as expected.
- After the adjustment of mechanical home of the robot, please move the robot.
- Do not use robot at out of the workspace, without workspace check function. If you use may cause damage to the robot.
- If you want to stop the movement of the running in the instruction `MC_SyncLinearConveyor`, should use the instruction `MC_SyncOut`.
- In Robot Ver. 1.01 or later, the velocity error (too fast) or the acceleration error (too much acceleration/deceleration) are not detected when `MaxVelocity` or `MaxAcceleration` are set to 0 or they are default. Set correct values to `MaxVelocity` and `MaxAcceleration` in order to avoid unexpected velocity and acceleration/deceleration.
- In Robot Ver. 1.01 or later, the velocity error or the acceleration error are not detected at the moment of starting this instruction when `ToolID` is set other than 0. Please pay attention to that.

- **Battery Replacement**

- The Battery may leak, rupture, heat, or ignite. Never short-circuit, charge, disassemble, heat, or incinerate the Battery or subject it to strong shock.
- Dispose of any Battery that has been dropped on the floor or otherwise subjected to excessive shock. Batteries that have been subjected to shock may leak if they are used.
- UL standards require that only an experienced engineer replace the Battery. Make sure that an experienced engineer is in charge of Battery replacement.
- Apply power for at least five minutes before changing the Battery. Install a new Battery within five minutes (at 25 ° C) of turning OFF the power supply. If power is not supplied for at least 5 minutes, the saved data may be lost.

- **Unit Replacement**

- We recommend replacing the Battery with the power turned OFF to prevent the CPU Unit's sensitive internal components from being damaged by static electricity and to prevent malfunctions. The Battery can be replaced without turning OFF the power supply. To do so, always touch a grounded piece of metal to discharge static electricity from your body before you start the procedure. After you replace the Battery, connect the Sysmac Studio and clear the Low Battery Voltage error.
 - Make sure that the required data, including the user program, configurations, settings, variables, and memory used for CJ-series Units, is transferred to a CPU Unit that was replaced and to externally connected devices before restarting operation.
- Be sure to include the routing tables, network parameters, and other CPU Bus Unit data, which are stored in the CPU Unit.

- **Disposal**

- Dispose of the product and Batteries according to local ordinances as they apply.



廢電池請回收

- The following information must be displayed for all products that contain primary lithium batteries with a perchlorate content of 6 ppb or higher when shipped to or transported through the State of California, USA.
 - Perchlorate Material - special handling may apply.
 - See www.dtsc.ca.gov/hazardouswaste/perchlorate.
- The CPU Unit contains a primary lithium battery with a perchlorate content of 6 ppb or higher. Place the above information on the individual boxes and shipping boxes when shipping finished products that contain a CPU Unit to the State of California, USA.

Precautions for Correct Use

- Storage, Mounting, and Wiring
 - Do not operate or store the Controller in the following locations. Operation may stop or malfunctions may occur.
 - Locations subject to direct sunlight
 - Locations subject to temperatures or humidity outside the range specified in the specifications
 - Locations subject to condensation as the result of severe changes in temperature
 - Locations subject to corrosive or flammable gases
 - Locations subject to dust (especially iron dust) or salts
 - Locations subject to exposure to water, oil, or chemicals
 - Locations subject to shock or vibration
 - Take appropriate and sufficient countermeasures when installing the Controller in the following locations.
 - Locations subject to strong, high-frequency noise
 - Locations subject to static electricity or other forms of noise
 - Locations subject to strong electromagnetic fields
 - Locations subject to possible exposure to radioactivity
 - Locations close to power lines
 - Before touching a Unit, be sure to first touch a grounded metallic object in order to discharge any static build-up.
 - Install the Controller away from sources of heat and ensure proper ventilation. Not doing so may result in malfunction, in operation stopping, or in burning.
 - An I/O bus check error will occur and the Controller will stop if an I/O Connecting Cable's connector is disconnected from the Rack. Be sure that the connectors are secure.
 - Do not allow foreign matter to enter the openings in the Unit. Doing so may result in Unit burning, electric shock, or failure.
 - Do not allow wire clippings, shavings, or other foreign material to enter any Unit. Otherwise, Unit burning, failure, or malfunction may occur. Cover the Units or take other suitable countermeasures, especially during wiring work.
 - For EtherCAT and EtherNet/IP, use the connection methods and cables that are specified in the NJ-series CPU Unit Built-in EtherCAT Port User's Manual (Cat. No. W505) and the NJ-series CPU Unit Built-in EtherNet/IP Port User's Manual (Cat. No. W506). Otherwise, communications may be faulty.
 - Use the rated power supply voltage for the Power Supply Units. Take appropriate measures to ensure that the specified power with the rated voltage and frequency is supplied in places where the power supply is unstable.
 - Make sure that the current capacity of the wire is sufficient. Otherwise, excessive heat may be generated. When cross-wiring terminals, the total current for all the terminals will flow in the wire. When wiring cross-overs, make sure that the current capacity of each of the wires is not exceeded.
 - Do not touch the terminals on the Power Supply Unit immediately after turning OFF the power

supply. Residual voltage may cause electrical shock.

- If you use reed switches for the input contacts for AC Input Units, use switches with a current capacity of 1 A or greater.

If the capacity of the reed switches is too low, surge current may fuse the contacts.

● Error Processing

- In applications that use the results of instructions that read the error status, consider the affect on the system when errors are detected and program error processing accordingly. For example, even the detection of a minor error, such as Battery replacement during operation, can affect the system depending on how the user program is written.

● Unit Replacement

- If you replace a CPU Bus Unit or Special I/O Unit, refer to operation manual for the Unit for information on the data required for individual Units and redo the necessary settings.

- The absolute encoder home offset is backed up with a Battery in the CPU Unit.

When you change the combination of the CPU Unit and Servomotor, e.g., when you add or replace a Servomotor, define home again.

To restore the information without changing the CPU Unit-Servomotor combination, remove the absolute encoder home offset from the data to restore.

● Task Settings

- If a Task Period Exceeded error occurs, shorten the programs to fit in the task period or increase the setting of the task period.

● Motion Control

- Use the system-defined variable in the user program to confirm that EtherCAT communications are established before you attempt to execute motion control instructions. Motion control instructions are not executed normally if EtherCAT communications are not established.

- Use the system-defined variables to monitor for errors in communications with the slaves that are controlled by the motion control function module. Motion control instructions are not executed normally if an error occur in slave communications.

- Before you start an MC Test Run, make sure that the operation parameters are set correctly.

- Do not download motion control settings during an MC Test Run.

● EtherCAT Communications

- Do not disconnect the EtherCAT slave cables during operation. The outputs will become unstable.

- Set the Servo Drives to stop operation if an error occurs in EtherCAT communications between the Controller and a Servo Drive.

● Battery Replacement

- Be sure to install a replacement Battery within two years of the production date shown on the Battery label.

- Turn ON the power after replacing the Battery for a CPU Unit that has been unused for a long time. Leaving the CPU Unit unused again without turning ON the power even once after the Battery is replaced may result in a shorter Battery life.
- When you replace the Battery, use the CJ1W-BAT01 Battery Set.

- **SD Memory Cards**

- Insert the SD Memory Card all the way.
- Do not turn OFF the power supply to the Controller during SD Memory Card access. The files may be corrupted.

If there is a corrupted file in the SD Memory Card, the file is automatically deleted by the restoration function when the power supply is turned ON.

Regulations and Standards

Conformance to EC Directives

- **Applicable Directives**
 - EMC Directives
 - Low Voltage Directive

- **Concepts**

- **EMC Directive**

OMRON devices that comply with EC Directives also conform to the related EMC standards so that they can be more easily built into other devices or the overall machine. The actual products have been checked for conformity to EMC standards.*

Whether the products conform to the standards in the system used by the customer, however, must be checked by the customer. EMC-related performance of the OMRON devices that comply with EC Directives will vary depending on the configuration, wiring, and other conditions of the equipment or control panel on which the OMRON devices are installed. The customer must, therefore, perform the final check to confirm that devices and the overall machine conform to EMC standards.

* Applicable EMC (Electromagnetic Compatibility) standards are as follows: EMS (Electromagnetic Susceptibility): EN 61131-2 and EN 61000-6-2 EMI (Electromagnetic Interference): EN 61131-2 and EN 61000-6-4 (Radiated emission: 10-m regulations)

- **Low Voltage Directive**

Always ensure that devices operating at voltages of 50 to 1,000 VAC and 75 to 1,500 VDC meet the required safety standards. The applicable directive is EN 61131-2.

- **Conformance to EC Directives**

The NJ-series Controllers comply with EC Directives. To ensure that the machine or device in which the NJ-series Controller is used complies with EC Directives, the Controller must be installed as follows:

- The NJ-series Controller must be installed within a control panel.
- You must use reinforced insulation or double insulation for the DC power supplies connected to DC Power Supply Units and I/O Units.
- NJ-series Controllers that comply with EC Directives also conform to the Common Emission Standard (EN 61000-6-4). Radiated emission characteristics (10-m regulations) may vary depending on the configuration of the control panel used, other devices connected to the control panel, wiring, and other conditions. You must therefore confirm that the overall machine or equipment complies with EC Directives.

Conformance to Shipbuilding Standards

The NJ-series Controllers comply with the following shipbuilding standards. Applicability to the shipbuilding standards is based on certain usage conditions. It may not be possible to use the product in some locations. Contact your OMRON representative before attempting to use a Controller on a ship.

- Usage Conditions for NK and LR Shipbuilding Standards

- The NJ-series Controller must be installed within a control panel.
- Gaps in the door to the control panel must be completely filled or covered with gaskets or other material.
- The following noise filter must be connected to the power supply line.


Noise Filter

Manufacturer	Model
Cosel Co., Ltd.	TAH-06-683

Usage in the robot system

NJ501-4300/4310/4400/4500, does not conform to safety regulations and standards for industrial robots. In this reason, if you use the NJ501-4300/4310/4400/4500 for robot systems including industrial robots, you ensure reliably conformance to safety regulations and standards of industrial robots ,and please implement the measures as necessary to ensure safety.

Trademarks

- Sysmac and SYSMAC are trademarks or registered trademarks of OMRON Corporation in Japan and other countries for OMRON factory automation products.
 - Windows, Windows Vista, are registered trademarks of Microsoft Corporation in the USA and other countries.
 - EtherCAT[®] is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.
 - The SD logo is a trademark of SD-3C, LLC. 
- Other company names and product names in this document are the trademarks or registered trademarks of their respective companies.
- Microsoft product screen shot(s) reprinted with permission from Microsoft Corporation.

Software Licenses and Copyrights

The license and copyright information associated with this software is available at http://www.fa.omron.co.jp/nj_info_e/ in Items related to NJ501-1300/1400/1500. This product incorporates certain third party software. The license and copyright information associated with this software is available at http://www.fa.omron.co.jp/nj_info_e/.

Unit Versions

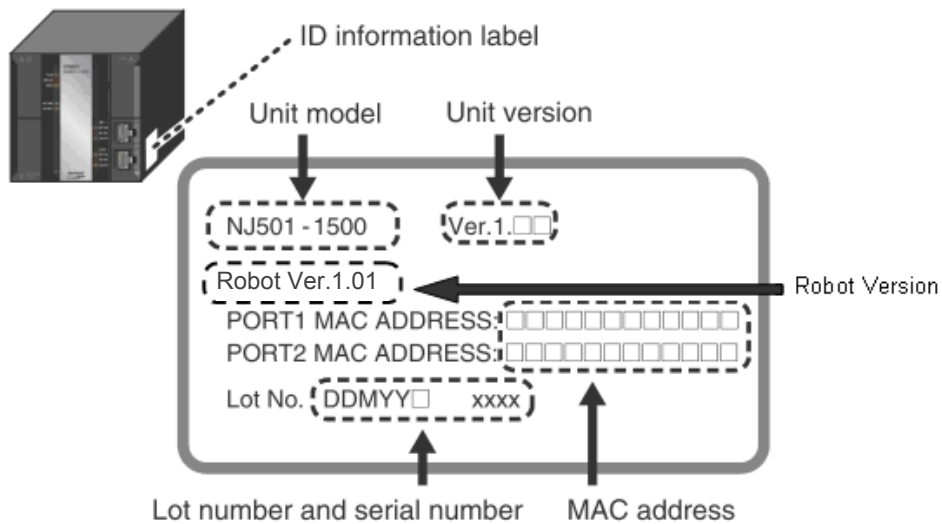
Unit Versions and Notation

A “unit version” has been introduced to manage CPU Units in the NJ Series according to differences in functionality accompanying Unit upgrades.

Notation of Unit Versions on Products

The unit version is given on the ID information label of the products for which unit versions are managed, as shown below.

Example for Robot-supported NJ-series NJ501-4300/4310/4400/4500 CPU Unit:



The following information is provided on the ID information label.

Item	Description
Unit Model	Gives the model of the Unit.
Unit Version	Given the unit version of NJ501-4□□□ that is added the robot function.
Robot Version	Given the version of robot function.
Lot number and serial number	Gives the lot number and serial number of the Unit. DDMY□□: Lot number, □: For use by OMRON, xxx□: Serial number “M” gives the month (1 to 9: January to September, X: October, Y: November, Z: December)
MAC address	Gives the MAC address of the built-in port on the Unit.

● Confirming Unit Versions with Sysmac Studio

You can use the Unit Production Information on the Sysmac Studio to check the unit version of the CPU Unit, CJ-series Special I/O Units, CJ-series CPU Bus Units, and EtherCAT slaves. The unit versions of CJ-series Basic I/O Units cannot be checked from the Sysmac Studio.

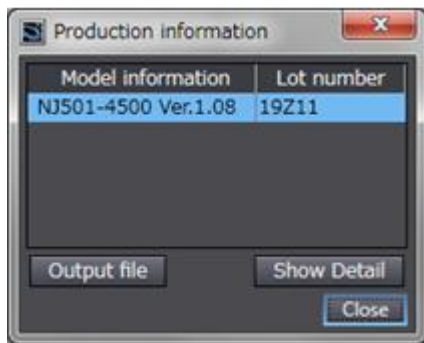
- CPU Unit and CJ-series Units

1 Double-click CPU/Expansion Racks under Configurations and Setup in the Multiview Explorer. Or, right-click CPU/Expansion Racks under Configurations and Setup and select Edit from the menu.

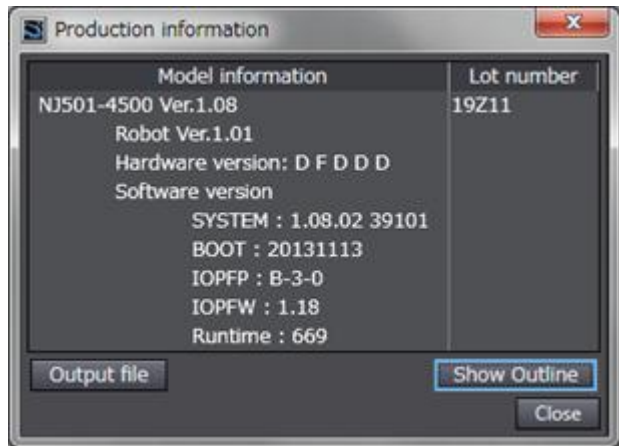
The Unit Editor is displayed for the Controller Configurations and Setup layer.

2 Right-click any open space in the Unit Editor and select Production Information.

The Production Information Dialog Box is displayed



Simple Display



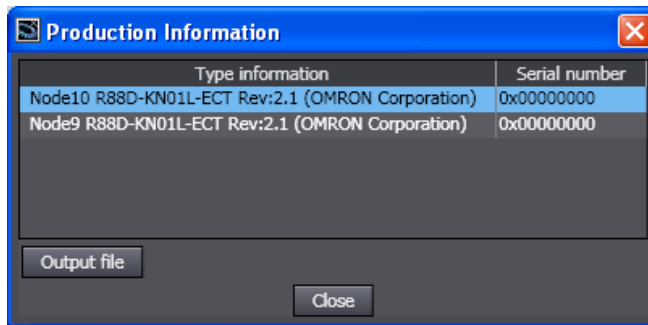
Detailed Display

The following items are displayed.

CPU Unit	CJ-series Units
Unit model	Unit model
Unit Version	Unit version
RobotVersion	Lot number
Lot number	Rack number, slot number, and unit number



- EtherCAT Slaves
 - 1 Double-click EtherCAT under Configurations and Setup in the Multiview Explorer. Or, right click EtherCAT under Configurations and Setup and select Edit from the menu. The EtherCAT Configuration Tab Page is displayed for the Controller Configurations and Setup layer.
 - 2 Right-click the master in the EtherCAT Configurations Editing Pane and select Display Production Information. The Production Information Dialog Box is displayed.



The following items are displayed.

Node address

Type information*

Serial number

* If the model number cannot be determined (such as when there is no ESI file), the vendor ID, product code, and revision number are displayed.

Unit Versions and Sysmac Studio Versions

The functions that are supported depend on the unit version of the Robot-supported NJ Series CPU Unit. The version of Sysmac Studio that supports the functions that were added for an upgrade is also required to use those functions.

Refer to NJ-series CPU Unit Software User's Manual(Cat. No. W501) Version Information for the relationship between the unit versions of the CPU Units and the Sysmac Studio versions, and for the functions that are supported by each unit version.

Unit Version Notation

In this manual, unit versions are specified as shown in the following table.

Product nameplate	Notation in this manual	Remarks
"Ver.1.0" or later to the right of the lot number	Unit version 1.0 or later	Unless unit versions are specified, the information in this manual applies to all unit versions.

Related Manuals

The following manuals are related to the NJ-series Controllers. Please read these manuals surely.

Manual name	Cat. No.	Model numbers	Application	Description
NJ-series CPU Unit Hardware User's Manual	W500	NJ501-□□□□ NJ301-□□□□	Learning the basic specifications of the NJ-series CPU Units, including introductory information, designing, installation, and maintenance. Mainly hardware information is provided.	An introduction to the entire NJ-series system is provided along with the following information on the CPU Unit. <ul style="list-style-type: none"> • Features and system configuration • Introduction • Part names and functions • General specifications • Installation and wiring • Maintenance and inspection Use this manual together with the <i>NJ-series CPU Unit Software User's Manual</i> (Cat. No. W501).
NJ-series CPU Unit Software User's Manual	W501	NJ501-□□□□ NJ301-□□□□	Learning how to program and set up an NJ-series CPU Unit. Mainly software information is provided.	The following information is provided on a Controller built with an NJ501 CPU Unit. <ul style="list-style-type: none"> • CPU Unit operation • CPU Unit features • Initial settings • Programming based on IEC 61131-3 language specifications Use this manual together with the <i>NJ-series CPU Unit Hardware User's Manual</i> (Cat. No. W500).
NJ-series Instructions Reference Manual	W502	NJ501-□□□□ NJ301-□□□□	Learning detailed specifications on the basic instructions of an NJ-series CPU Unit.	The instructions in the instruction set (IEC 61131-3 specifications) are described. When programming, use this manual together with the <i>NJ-series CPU Unit Hardware User's Manual</i> (Cat. No. W500) and <i>NJ-series CPU Unit Software User's Manual</i> (Cat. No. W501).
NJ-series CPU Unit Motion Control User's Manual	W507	NJ501-□□□□ NJ301-□□□□	Learning about motion control settings and programming concepts.	The settings and operation of the CPU Unit and programming concepts for motion control are described. Use this manual together with the <i>NJ-series CPU Unit Hardware User's Manual</i> (Cat. No. W500) and <i>NJ-series CPU Unit Software User's Manual</i> (Cat. No. W501).
NJ-series Motion Control Instructions Reference Manual	W508	NJ501-□□□□ NJ301-□□□□	Learning about the specifications of the motion control instructions that are provided by OMRON.	The motion control instructions are described. When programming, use this manual together with the <i>NJ-series CPU Unit Hardware User's Manual</i> (Cat. No. W500), <i>NJ-series CPU Unit Software User's Manual</i> (Cat. No. W501) and <i>NJ-series CPU Unit Motion Control User's Manual</i> (Cat. No. W507).
NJ-series CPU Unit Built-in EtherCAT® Port User's Manual	W505	NJ501-□□□□ NJ301-□□□□	Using the built-in EtherCAT port on an NJ-series CPU Unit.	Information on the built-in EtherCAT port is provided. This manual provides an introduction and provides information on the configuration, features, and setup. Use this manual together with the <i>NJ-series CPU Unit Hardware User's Manual</i> (Cat. No. W500) and <i>NJ-series CPU Unit Software User's Manual</i> (Cat. No. W501).
NJ-series CPU Unit Built-in EtherNet/IP™ Port User's Manual	W506	NJ501-□□□□ NJ301-□□□□	Using the built-in EtherNet/IP port on an NJ-series CPU Unit.	Information on the built-in EtherNet/IP port is provided. Information is provided on the basic setup, tag data links, and other features. Use this manual together with the <i>NJ-series CPU Unit Hardware User's Manual</i> (Cat. No. W500) and <i>NJ-series CPU Unit Software User's Manual</i> (Cat. No. W501).
NJ-series Database Connection CPU Units User's Manual	W527	NJ501-1□20	Using the database connection service with NJ-series Controllers	Describes the database connection service.

Manual name	Cat. No.	Model numbers	Application	Description
NJ-series Troubleshooting Manual	W503	NJ501-□□□□ NJ301-□□□□	Learning about the errors that may be detected in an NJ-series Controller.	Concepts on managing errors that may be detected in an NJ-series Controller and information on individual errors are described. Use this manual together with the <i>NJ-series CPU Unit Hardware User's Manual</i> (Cat. No. W500) and <i>NJ-series CPU Unit Software User's Manual</i> (Cat. No. W501).
Sysmac Studio Version 1 Operation Manual	W504	SYSMAC-SE2□□□	Learning about the operating procedures and functions of the Sysmac Studio.	Describes the operating procedures of the Sysmac Studio.
NX-series Communications Coupler Unit User's Manual	W519	NX-ECC□□□	Learning how to use an NX-series Communications Coupler Unit and Slave Terminals	Introduces the system, configuration methods, Unit hardware, setting methods, and functions of Slave Terminals that consist of a Communications Coupler Unit and NX Units.
NX-series NX Units User's Manuals	W521	NX-ID□□□□ NX-IA□□□□ NX-OC□□□□ NX-OD□□□□	Learning how to use NX Units	Describes the hardware, setup methods, and functions of the NX Units. Manuals are available for the following Units. Digital I/O Units, Analog I/O Units, System Units, and Position Interface Units.
	W522	NX-AD□□□□ NX-DA□□□□ NX-TS□□□□		
	W523	NX-PD1□□□ NX-PF0□□□ NX-PC0□□□ NX-TBX□□		
	W524	NX-EC0□□□ NX-ECS□□□ NX-PG0□□□		
NX-series Data Reference Manual	W525	NX-□□□□□□	Referring to the list of data required for NX-series unit system configuration.	Provides the list of data required for system configuration including the power consumption and weight of each NX-series unit.
GX-series EtherCAT Slave Units User's Manual	W488	GX-ID□□□□ GX-OD□□□□ GX-OC□□□□ GX-MD□□□□ GX-AD□□□□ GX-DA□□□□ GX-EC□□□□ XWT-ID□□ XWT-OD□□	Learning how to use the EtherCAT remote I/O terminals.	Describes the hardware, setup methods and functions of the EtherCAT remote I/O terminals.
G5-series AC Servomotors/Servo Drives User's Manuals	1573	R88M-K□ R88D-KN□-ECTR	Learning how to use the AC Servomotors/Servo Drives with built-in EtherCAT Communications.	Describes the hardware, setup methods and functions of the AC Servomotors/Servo Drives with built-in EtherCAT Communications. The linear motor type model and the model dedicated for position controls are available in G5-series.
	1576	R88M-K□ R88D-KN□-ECT		
	1577	R88L-EC-□ R88D-KN□-ECTL		

Revision History

RevisionCode	Date	Revised content
A	June 2012	Original production
B	July 2012	Add Information on the NJ501-4310
C	Nov 2012	Add Information on the NJ501-4310
D	Jan 2013	Updated the only revision code for synchronizing revision code of Japanese user's manual.
E	Feb 2014	Updated the manual by releasing Robot version 1.01.

Terminology

Term	Description
NJ501-1*00	NJ501-1300/1400/1500
Robot-supported NJ Series	NJ501-4300/4310/4400/4500. In some case, these models describe as NJ501-4□□□ in this manual.
SysmacStudio Robot Option	SysmacStudio Robot Option is SysmacStudio option to use a robotic function of NJ501-4□□□.It will be available when you in put the license No. for Sysmac Studio V1.03 or later.
Robot	Composition that consists of multiple links and axes (joints) such as Delta and SCARA.
Delta	Robot using the parallel link composition. This controller is possible to control two types of Delta robots, 2-parallel links type and 3-parallel links type. Further, there are robots with wrist rotation axis and robots without wrist.
Delta3	The 3-parallel links type of Delta robot is called "Delta3" in this manual.
Delta3R	The 3-parallel links type of Delta robot with wrist rotation is called "Delta3R" in this manual.
Delta2	The 2-parallel links type of Delta robot is called "Delta2" in this manual.
Kinematics	In this manual, the formulae required for transforming from axis coordinate system (ACS to be described later) of each link/axis composition of the robot to robot-specific Cartesian coordinate system (MCS to be described later) and vice versa are collectively called "kinematics".
Inverse Kinematics	Transforming the position on the robot-specific Cartesian coordinate system (MCS) to the position on the axis coordinate system (ACS) of the link/axis composition is called "inverse kinematics".
Direct Kinematics	Transforming the position on the axis coordinate system (ACS) of the link/axis composition to the position on the robot-specific Cartesian coordinate system (MCS) is called "direct kinematics".
Axis Coordinate System	Rotating coordinate system or linear coordinate system specific to each axis of the robot. Abbreviated as "ACS".
Machine Coordinate System	Cartesian coordinate system specific to the machine (i.e. specific to the robot). Abbreviated as "MCS".
User Coordinate System	Cartesian coordinate system that the user can define freely. Abbreviated as "UCS".
TCP (Tool Center Point)	Working point of the machine (robot). In the positioning on the Cartesian coordinate system, this TCP position is specified.
Fixed Frame	Frame to which each axis motor is mounted on a Delta robot.
Moving Frame	Frame driven by each axis motor of a Delta robot.
Robot Tool	A mechanical object attached to the robot flange used to perform operations on a workpiece.
TCP0	Default Tool Center Point of the Robot.
TCPi	Active Tool Center Point of the Robot.
Tool Coordinate System	Cartesian coordinate system that has the position of the Tool Center Point (TCP) at its origin. Abbreviated as "TCS".
TCS0	Default Tool Coordinate System. Its origin is located on the TCP0 of the robot.
TCSi	Active Tool Coordinate System.

1 Introduction to Robot-supported NJ Series

1.1 Features

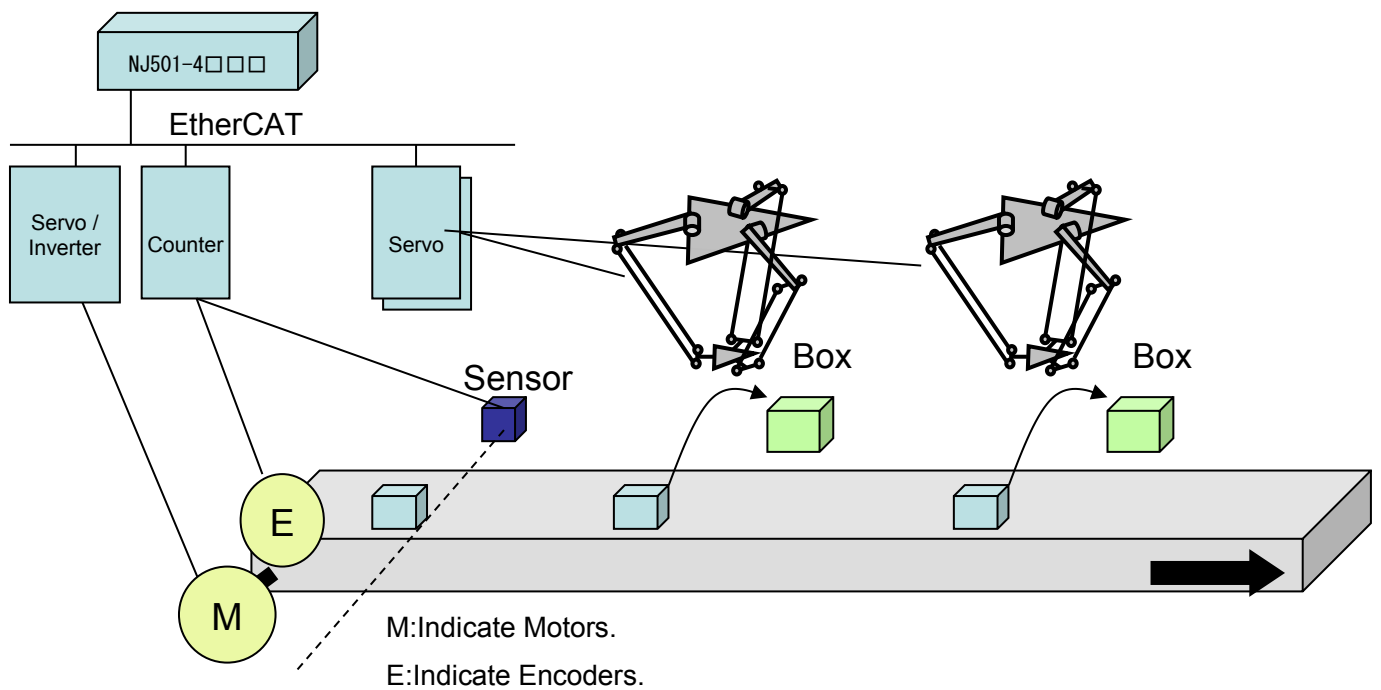
This robot package has the following features.

- Robot operations, single-axis operations, and synchronous operations can be all realized by Motion Control Function Module.
- The commands to the robot are realized by the function blocks specified in IEC 61131-1 (including the PLCopen-compliant instructions) in a unified manner.
- The instructions to facilitate application construction are supported as a standard feature.

1.2 System Configuration

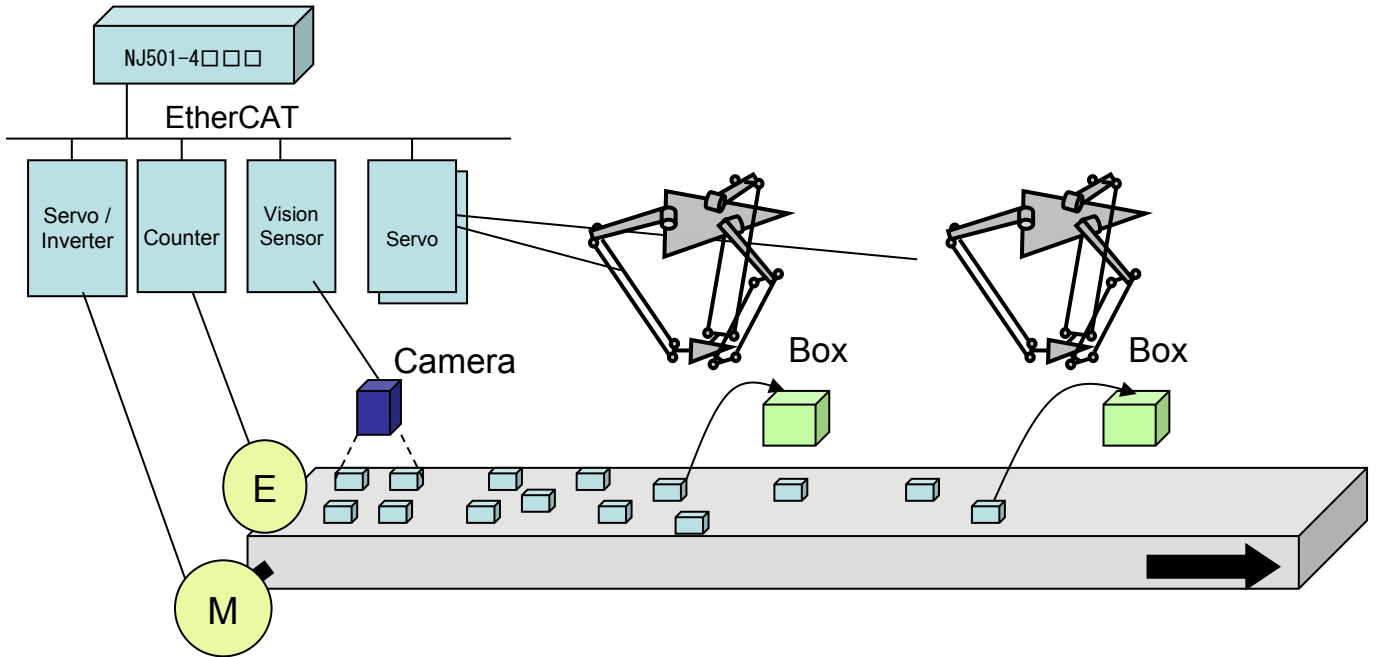
System configuration diagrams of this package are provided below.

System Configuration 1: Box Packing

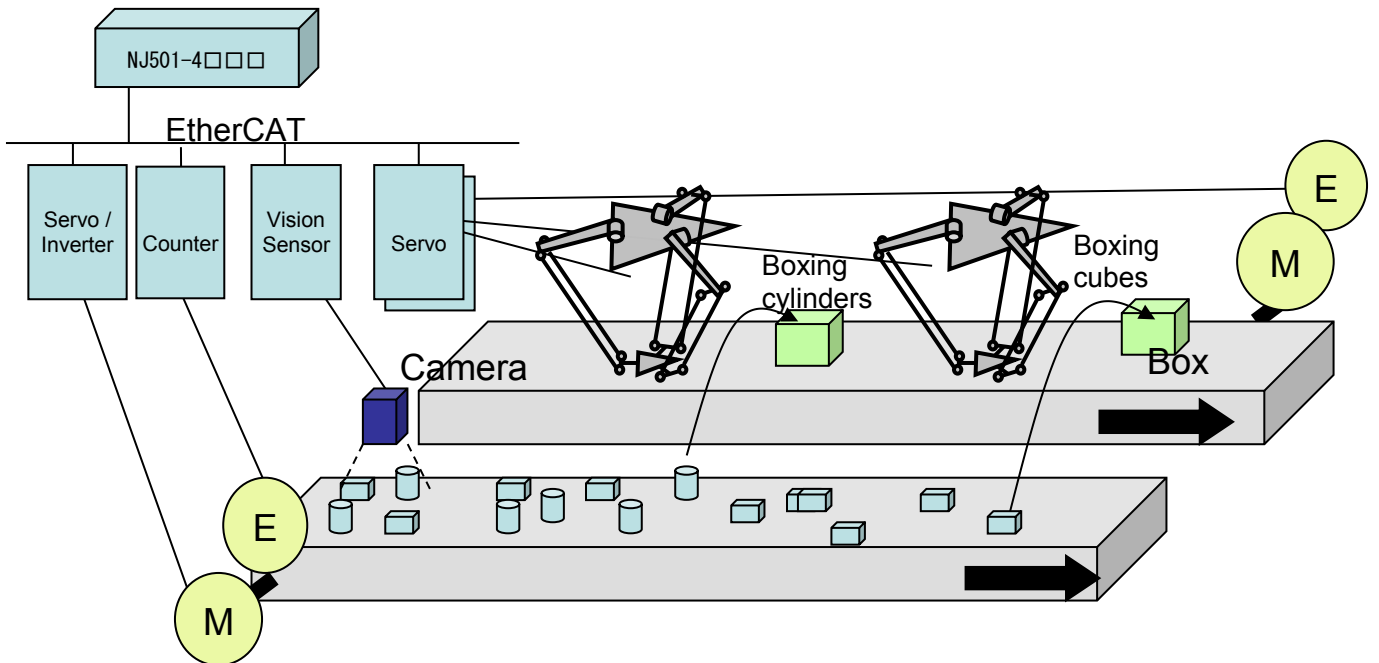


System Configuration 2: Box Packing

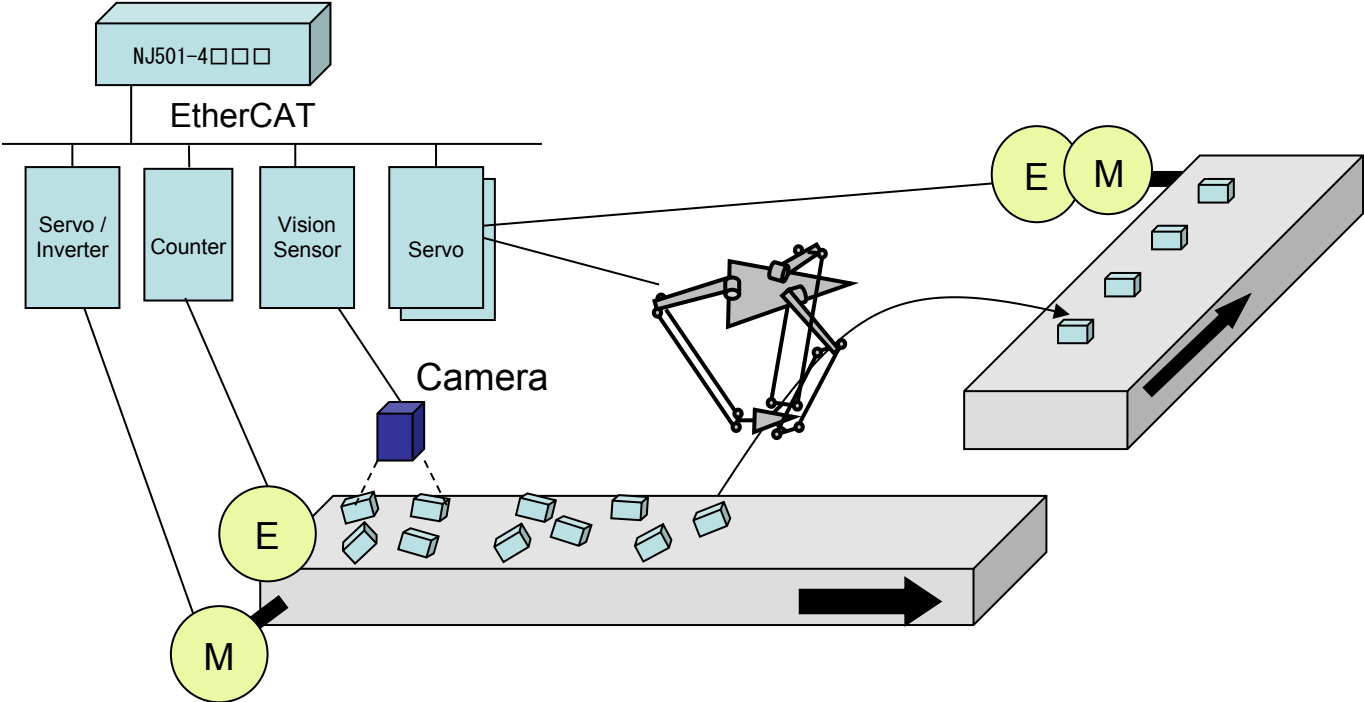
Using a camera



System Configuration 3: Box Packing

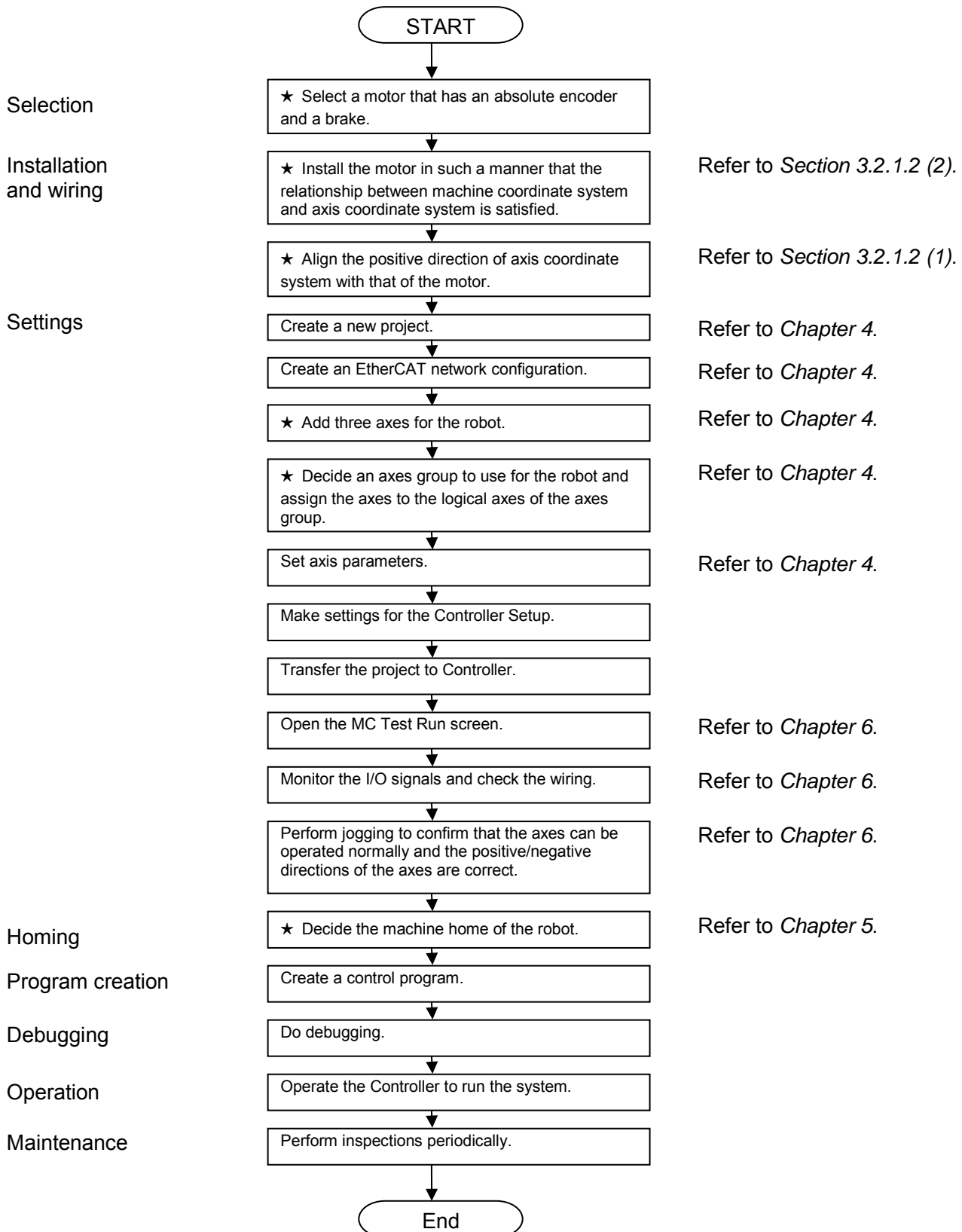


System Configuration 4: Alignment



1.3 Startup Procedure

When operating a Delta3, use the following procedure for startup. The steps marked with ★ are different from NJ501-1*00. Refer to NJ-series CPU Unit Motion Control User's Manual(Cat. No. W507) for other steps.



1.4 Specifications

1.4.1 Function Specifications

In the Robot-supported NJ Series, the following functions are added to the Motion Control Function Module of NJ501-1*00.

Item	Description
Coordinate system	Two types of coordinate systems, Machine Coordinate System (MCS) and User Coordinate System (UCS), can be used for robots.
Robot parameter settings	Sets various parameters (such as kinematics type and link length) for the robot.
Workspace check	Checks if the robot is moving within the movable volume (i.e. workspace).
User coordinate system	A specific coordinate system (user coordinate system) can be set for each robot.
Robot tool	Robot tools can be set for each robot.
Monitor	Reads the current position and current velocity of the robot.
Time-specified absolute positioning command	Moves the robot to a specified position in a specified time.
Synchronization with conveyor	Makes the active TCP follow a workpiece on the conveyor.
Inverse kinematics	Transforms the coordinate value (X, Y, Z) of the robot's TCP to the coordinate values of each axis.
Robot Jog	Jogs a robot defined by an Axes Group according the selected target velocity
Transition	Select a transition method to use between instructions in order to lessen this shock

1.4.2 Performance Specifications

Item	Specifications	
Controllable Robot Type	Delta3 (in Robot Ver. 1.00 or later) Delta3R, Delta2 (in Robot Ver. 1.01 or later)	
Controllable Servo Drives	OMRON G5-series Servo Drives with built-in EtherCAT communications ^{*1}	
Recommended encoder input terminals	OMRON GX-series GX-EC0211/EC0241 EtherCAT Remote I/O Terminals ^{*2} OMRON NX-series NX-EC0122/EC0222/EC0142 Incremental Encoder Input Units ^{*5} OMRON NX-series NX-ECS112/ECS212 SSI Input Units ^{*5}	
Control method	Control commands using EtherCAT communications	
Control modes	Position control, Velocity Control, and Torque control	
Number of controlled axes	Maximum number of controlled axes	64 axes (when using NJ501-4500) 32 axes (when using NJ501-4400) 16 axes (when using NJ501-4300/4310)
	Single-axis control	64 axes max. (when using NJ501-4500) 32 axes max. (when using NJ501-4400) 16 axes max. (when using NJ501-4300/4310)
	Linear interpolation axis control ^{*3}	4 axes max. per axes group
	Circular interpolation axis control ^{*3}	2 axes max. per axes group
Number of axes groups	32 axes groups max.	
Unit conversions	Display units	Pulse, mm, μm , nm, degree, and inch
	Electronic gear ratio	Pulse per motor rotation / travel distance per motor rotation
Positions that can be managed	Command positions and actual positions	
Position command values	Negative or positive long reals (LREAL) or 0 (command units ^{*4})	
Velocity command values	Negative or positive long reals (LREAL) or 0 (command units/s) 400 MHz max. (when converted to pulses) when using G5	
Acceleration command values and deceleration command values	Positive long reals (LREAL) or 0 (command units/s ²)	
Jerk command values	Positive long reals (LREAL) or 0 (command units/s ³)	
Override factors	0.00%, 0.01% to 500.00%	
Axis types	Servo axis, virtual servo axis, encoder axis, and virtual encoder axis	
Motion control period	The same control period as that is used for the process data communications cycle for EtherCAT. (Robot function to be use in 1ms, 2ms or 4 ms control period.)	
Cams	Number of cam data points	65,535 points per cam table 1,048,560 points max. for all cam tables
	Number of cam tables	640 tables max.
	Cam profile curves	To be created in the Cam Editor

Item		Specifications
	Rewriting of cam data	Cam data can be rewritten in the user program
Coordinate systems	Axis coordinate system (ACS)	1 axis coordinate system per axis
	Machine coordinate system (MCS)	1 machine coordinate system per axes group
	User coordinate system (UCS)	16 user coordinate systems per axes group
	Tool coordinate system (TCS)	17 tool coordinate systems per axes group (a default tool and user-defined robot tools)
Robot tool		16 tools per axes group
Number of robots	Maximum number of robots	8 robots max. (when using NJ501 - 4400/4500/4300)
		1 robots max. (when using NJ501 – 4310)

*1: Unit Ver. 2.1 or later is recommended.

*2: Unit Ver. 1.1 or later is recommended.

*3: Not possible to use for robotics group.

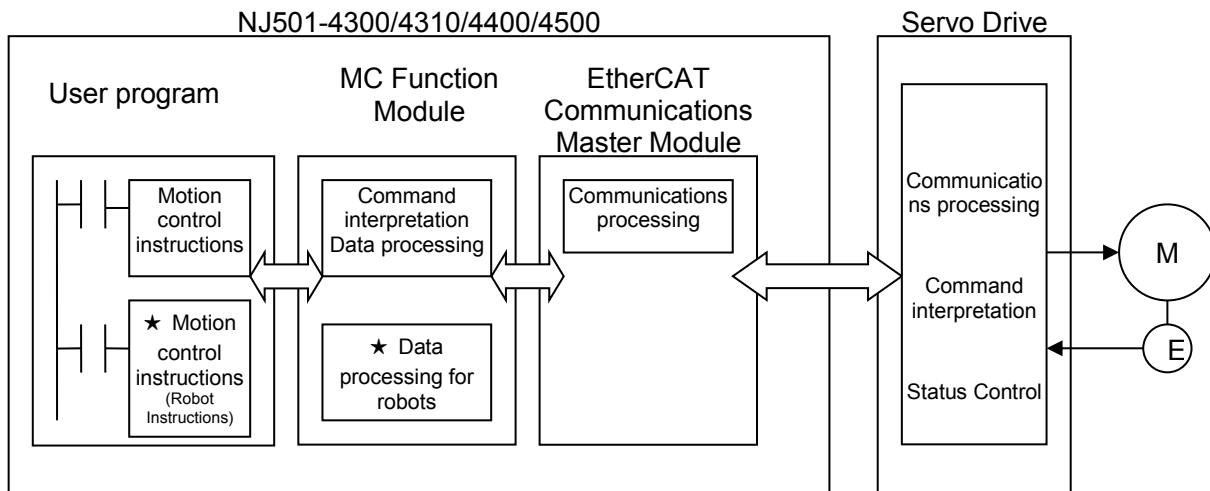
*4: Positions can be set within a 40-bit signed integer range when converted to pulses.

*5: Connect to an EtherCAT Coupler Unit (NX-ECC201).

2 Introduction to Robot Control

In the Robot-supported NJ Series, the motion control instructions for robots (hereinafter referred to as “robot instructions”) and data processing for robot function (Robot instructions and data processing for robots) marked with ★ are added to NJ501-1*00.

This realizes the robot control using IEC 61131-3 function blocks like NJ501-1*00.



3 Functions

3.1 Controllable Robot Type

In the Robot-supported NJ Series, the following three types of robots can be controlled: Delta3, Delta3R and Delta2.

3.2 Coordinate Systems

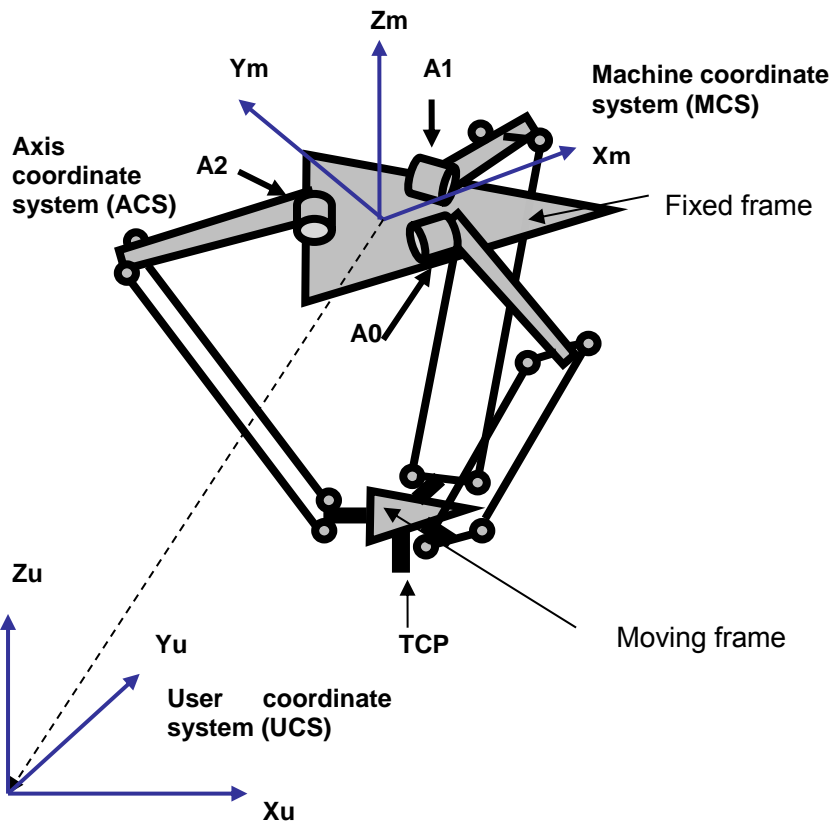
3.2.1 Outline

3.2.1.1 Types of Coordinate Systems

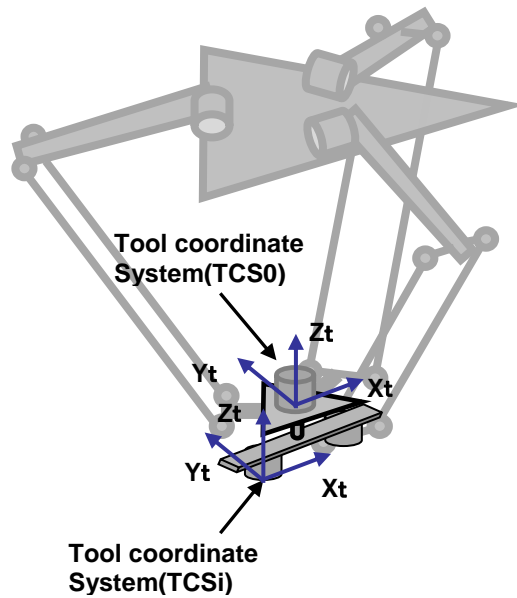
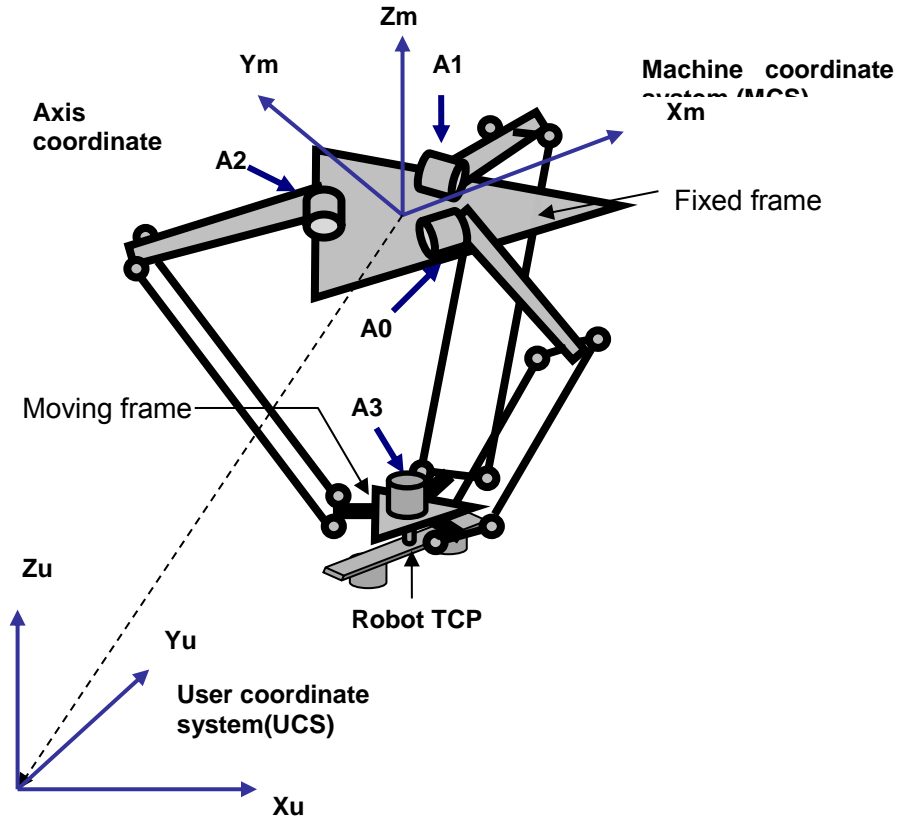
In the Robot-supported NJ Series, the following four types of coordinate systems are prepared to control the robots.

- Axis coordinate system (ACS): Coordinate system consisting of A0, A1 A2 in the figure below. (Refer to Section 3.2.2.)
- Machine coordinate system (MCS): Coordinate system consisting of X_m , Y_m , Z_m in the figure below. (Refer to Section 3.2.3.1)
- User coordinate system (UCS): Coordinate system consisting of X_u , Y_u , Z_u in the figure below. (Refer to Section 3.2.3.2)
- Tool coordinate system (TCS): Coordinate system consisting of X_t , Y_t , Z_t in the figure below. (Refer to Section 3.2.3.3)

(1) Delta3

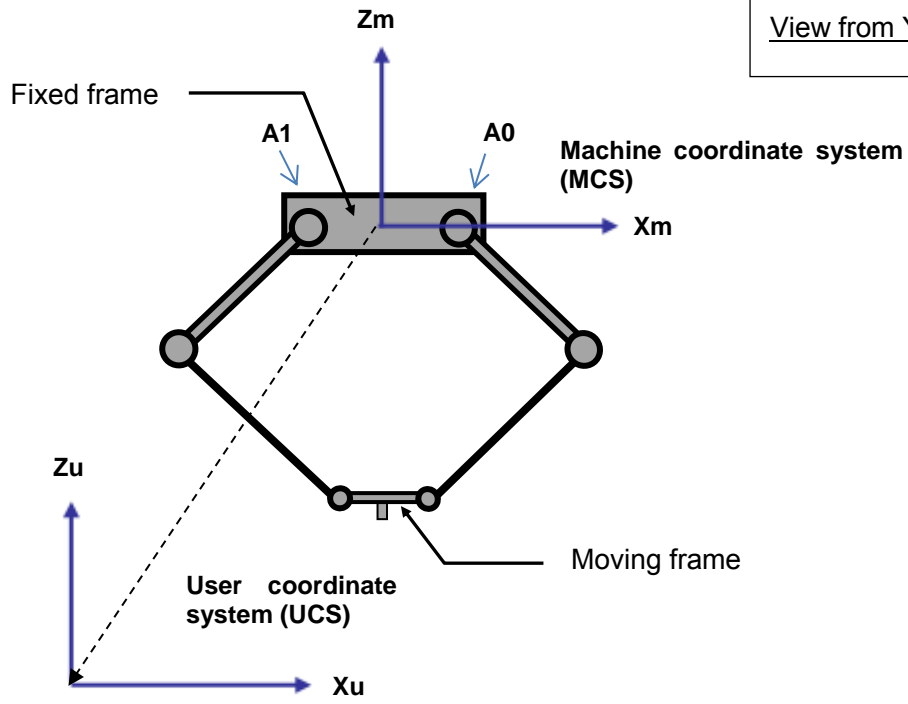


(2) Delta3R



(3) Delta2

View from Ym to Delta2

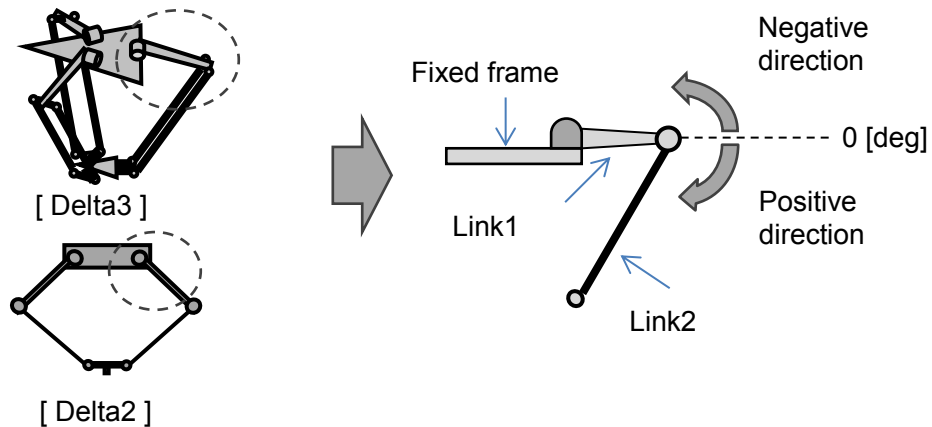


3.2.1.2 **Direction and Positional Relationship of Coordinate Systems**

(1) Direction of Axis Coordinate System (ACS)

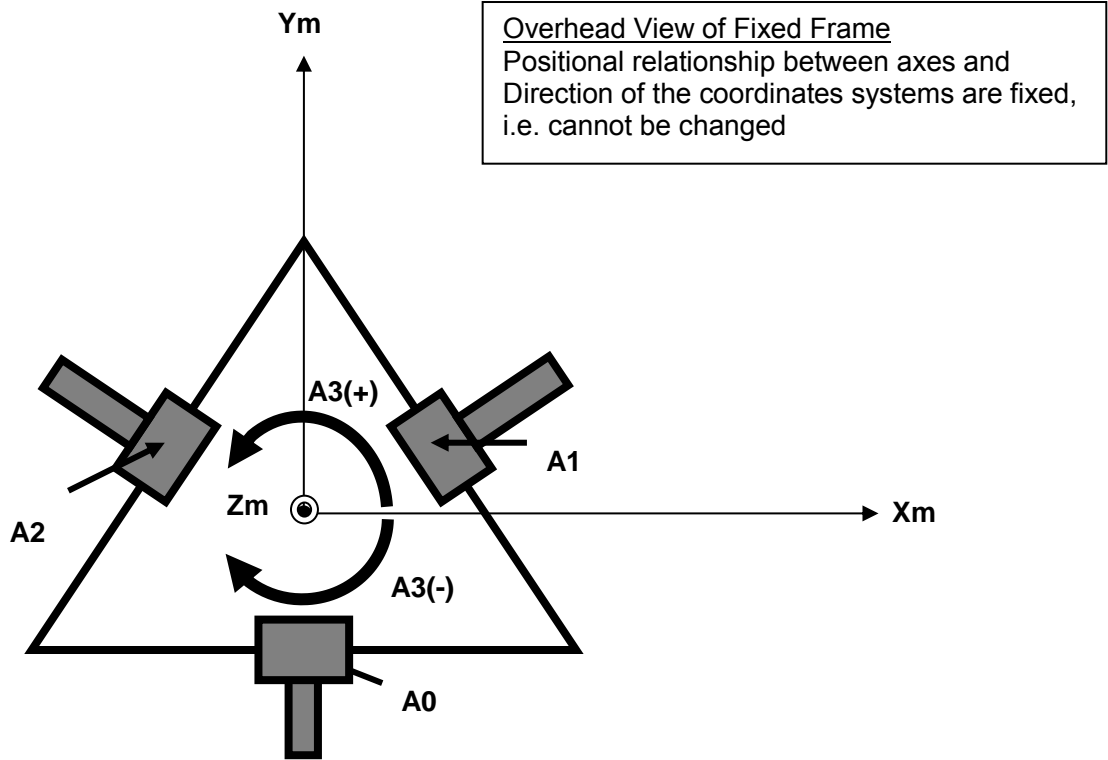
The position where the Link 1 is in alignment with the fixed frame is defined as “0 degree”. For all axes (in Delta3 and Delta3R from A0 to A2, and in Delta2 from A0 to A1), the lower side is positive direction*1 and the upper side is negative direction.

When installing the Servo Drives, make the rotating direction as shown below in the wiring. In addition, with all axes, set the (software limit) within the movable range of -180 degrees to 180 degrees.



*1: If an axis moves downward when a positive command current velocity is given to the axis (Servo Drive), the moving direction is called “positive direction”. If an axis moves upward when a negative command current velocity is given, the moving direction is called “negative direction”.

(2) Positional Relationship between Axis Coordinate System (ACS) and Machine Coordinate System (MCS) The directions of each axis (A0 to A2) and machine coordinate system (MCS) are as shown below.



Precautions for Safe Use

- Specify the setting of direction in the controller as same as the rotating direction of the Servo Drive.
Otherwise, the robot may cause unexpected operation.

3.2.2 Axis Coordinate System (ACS)

This is rotating coordinate system or linear coordinate system specific to each axis, which is also called “joint coordinate system” or “link coordinate system”.

One axis coordinate system is always created for each axis.

3.2.3 Cartesian Coordinate Systems

3.2.3.1 Machine Coordinate System (MCS)

This is a Cartesian coordinate system specific to each robot. The target position used for moving a robot is specified by the position of the working point of the robot (TCP) viewed from the home of this coordinate system. One machine coordinate system exist for each robot.

3.2.3.2 User Coordinate System (UCS)

This is a Cartesian coordinate system that can be freely defined by the user.

The home (T_x , T_y , T_z , R_x , R_y , R_z) of this coordinate system is specified by the position and rotation (called “pose”) from the home of machine coordinate system. Up to 16 user coordinate systems can be set for each robot.

3.2.3.3 Tool Coordinate System (TCS)

This is a Cartesian coordinate system that can be freely defined by the user.

The home of this coordinate system is specified by the position and rotation (called “pose”) from the robot TCP. Up to 16 tool coordinate systems can be set for each robot. With default tool coordinate system, up to 17 tool coordinate systems can be selected.

3.3 Special Robot Functions

It is required to set the kinematics type and transform parameters for the axes group to operate a robot. The MC_SetKinTransform instruction (described later) is used for the axes group settings.

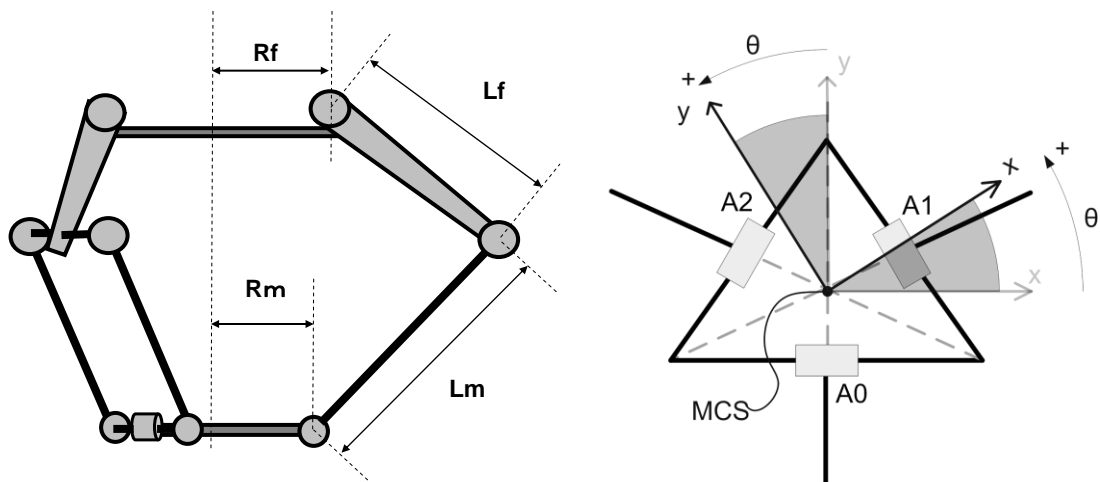
3.3.1 Kinematics Setting

Delta3, Delta3R and Delta2 are available.

3.3.1.1 Kinematics Setting for Delta3/Delta3R

For Delta3, the following parameters are set.

- Rf: stance (radius) from the center of the fixed frame to the motor of the axis [mm]
- Rm: Distance (radius) from the center of the moving frame to the connection point of Link 2 [mm]
- Lf: Length of Link 1 [mm]
- Lm: Length of Link 2 [mm]
- θ : Rotation around Z axis on the X-Y plane of the machine coordinate system (for axis adjustment) [deg]



The kinematics parameters of Delta3R are same as it of Delta3.

In case of the Delta3 kinematics type, if it exists, the wrist is controlled as single axis by the user. In case of the Delta-3R kinematics type, the wrist is controlled by robot instructions as part of the 4-axis group.

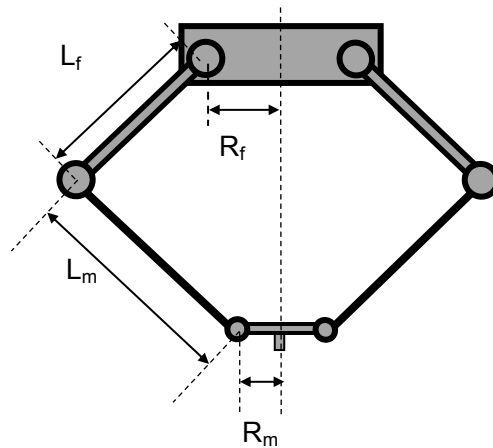
Note: In case of selecting Delta-3R, if a tool is used, wrist movement could occur unexpectedly for the user if not paying attention to the fact that the wrist is controlled as part of the robot group and not by the user.

When MCS is rotated by the parameter θ , the origin of the orientation around Z is aligned to the +X axis of the new MCS.

3.3.1.2 Kinematics Setting for Delta2

For Delta2, the following parameters are set.

- Rf: stance (radius) from the center of the fixed frame to the motor of the axis [mm]
- Rm: Distance (radius) from the center of the moving frame to the connection point of Link 2 [mm]
- Lf: Length of Link 1 [mm]
- Lm: Length of Link 2 [mm]



In case of Delta2, it can not use a wrist axis.

3.3.1.3 Constraints of the kinematics parameters

The Delta3, Delta3R and Delta2 that can be use with this product have to satisfy conditions all of the following.

1. The maximum length of the kinematics parameters
Must be the length of R_f is 2000[mm] less and L_m is 10000[mm] less.
2. Maximum range of the angle between two link1 and a link2
The angle of range between link1 and link2 have to be from 0 to 180[deg].
3. Relationship of the length of R_f and R_m
Must be the length of R_m is less than R_f .

3.3.2 Workspace Check

The workspace check is a function to check the movable volume of the robot's TCP.

If the position of the robot's TCP goes out of the workspace, it is considered as an error and the axis operation is stopped. When a robot instruction is executed, the TCP position is checked at the following four timing.

- Is the current stop position (i.e. actual current position) inside the workspace?
- Is the target position inside the workspace?
- Is the command current position created every control period inside the workspace?
- During stop of the robot, the current actual position is in the workspace?

To perform the workspace check, the user needs to set the necessary parameters for the axes group.

The MC_SetKinTransform instruction (described later) is used for the axes group settings and enabling the workspace check.

3.3.2.1 Setting of Workspace Type

Delta3Workspace and Delta2Workspace type is prepared.



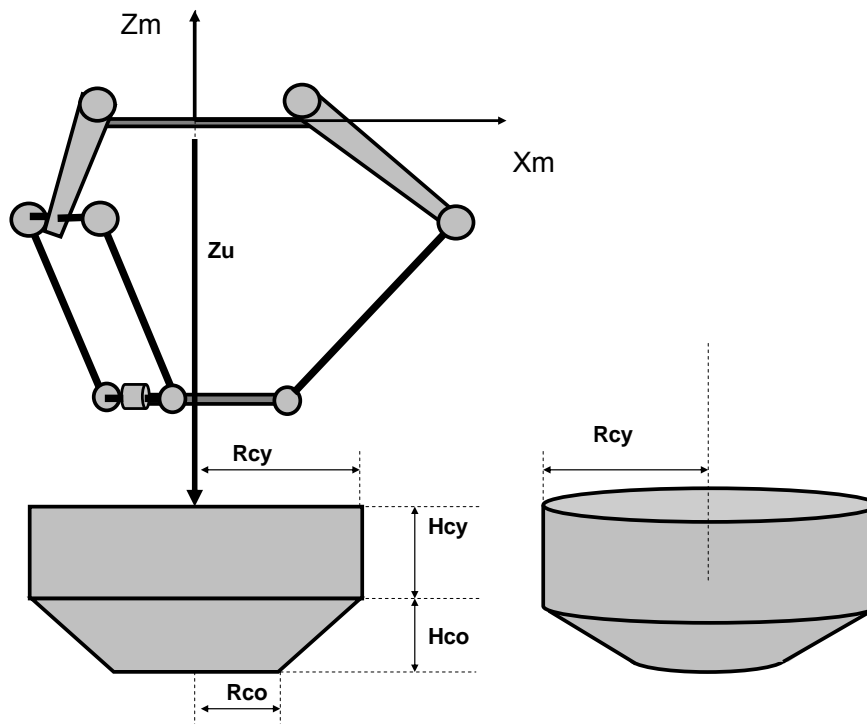
Precautions for Safe Use

- Choose the correct combination of Kinematics type and Workspace type which are set by MC_SetKinTransform.
Confirm if the workspace check works as expected.

3.3.2.2 Workspace parameters for Delta3 Robot

The Delta3 workspace is a combination of a cylinder and a frustum cone. The following parameters are set. Also this workspace type can be used for Delta3R robot.

- Zu: Front position of the cylinder (position from the home of Z axis) [mm]
- Rcy: Radius of the cylinder [mm]
- Hcy: Height of the cylinder [mm]
- Rco: Radius of the frustum cone of underside [mm]
- Hco: Height of the frustum cone [mm]



Note: In case of Delta 3R, when the 4th axis is attached with its motor placed on the fixed plate, the working volume normally becomes smaller comparing to the one of Delta 3 robot. In that case, change the parameters of the workspace by adapting them to the smaller volume.

3.3.2.3 Workspace parameters for Delta2 Robot

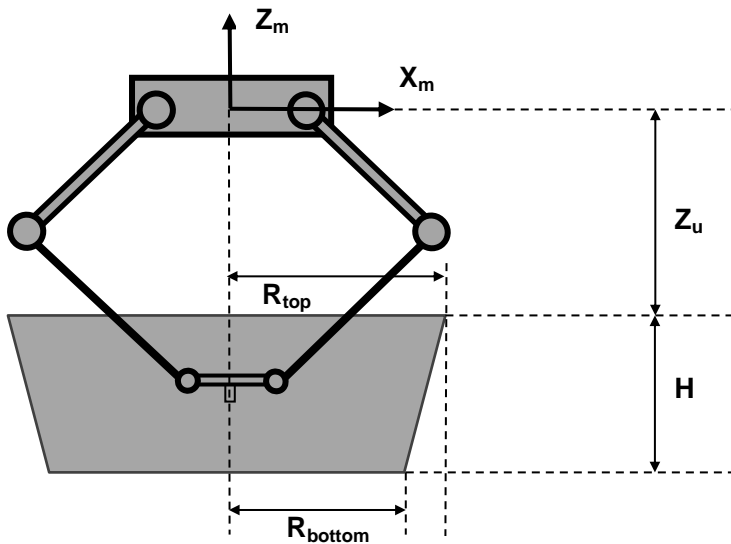
The Delta2 workspace is a trapezoid. The following parameters are set.

Z_u: Front position of the trapezoid (position from the home of Z axis) [mm]

H: Height of the trapezoid [mm]

R_{top}: Radius of the trapezoid of upper side [mm]

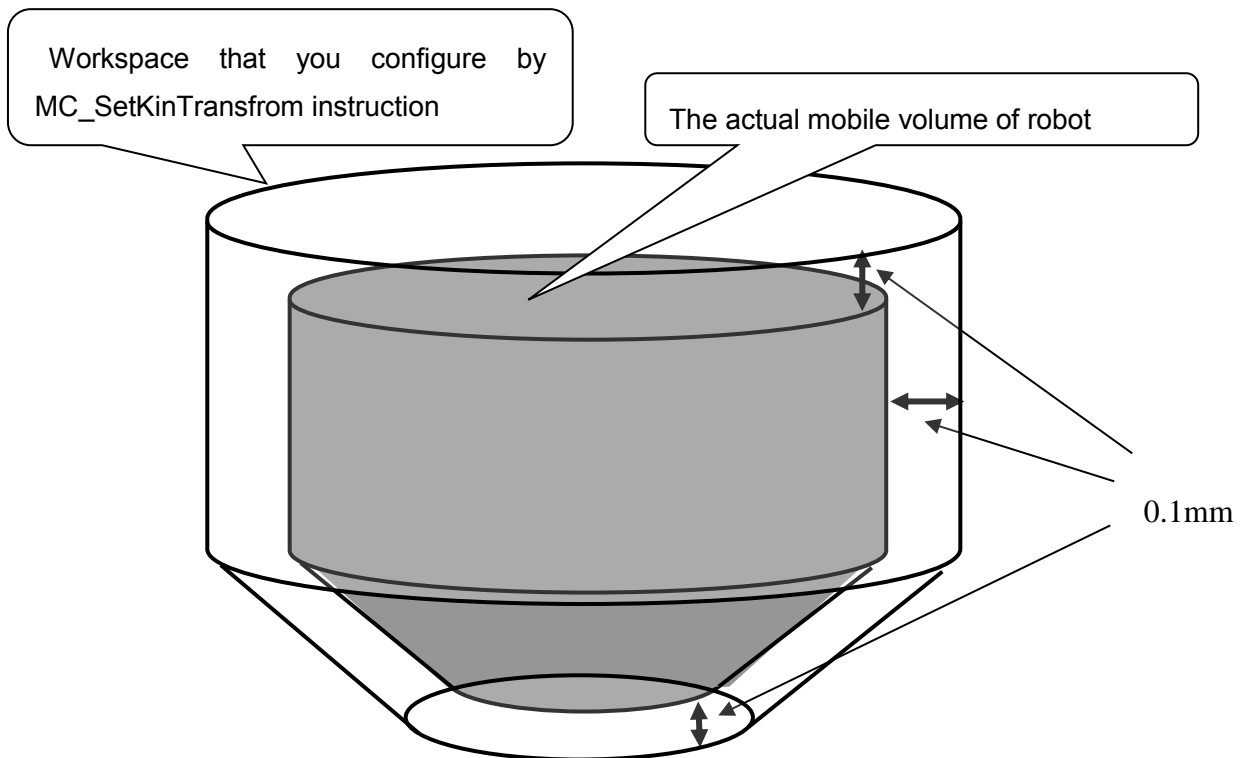
R_{bottom}: Radius of the trapezoid of underside [mm]



3.3.2.4 The actual mobile volume

The actual mobile volume is 0.1mm inside of the space that configure by MC_SetKinTransfrom instruction as indicated below figure. For this reason, the target position to move the robot, set the value of the workspace than 0.1mm inside the set. If the command current position or actual current position is out of “The actual mobile volume of robot” as indicated below figure , an error occurs and robot stop immediately .

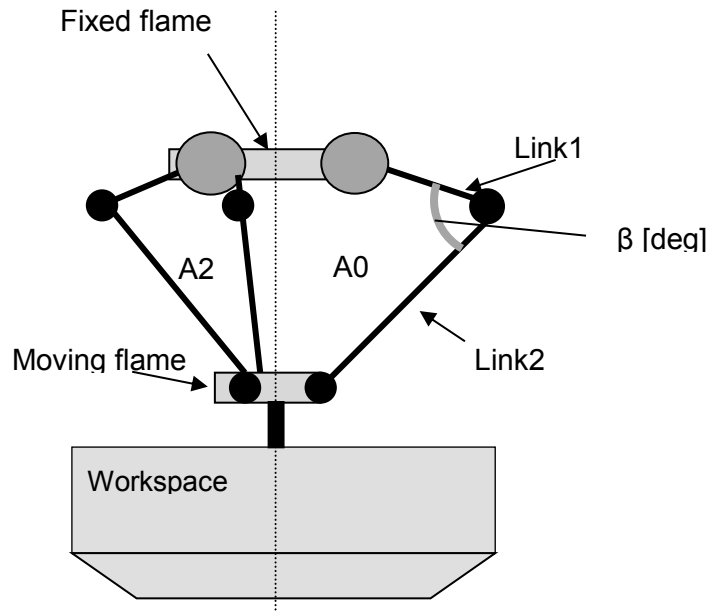
Example) Workspace for Delta3 Robot



3.3.2.5 Constraints of kinematics and workspace

1. Relationship between kinematics and parameters top workspace(Zu)

TCP of robot is necessary to be reachable to top of the workspace of the Z-axis((xm, ym, zm) = [0,0, Zu]), as shown in the figure below. In this case, angle of each motors are minimum under the conditions of moving TCP of the robot in the workspace, and the all angles(A0,A1 and A2) are same.



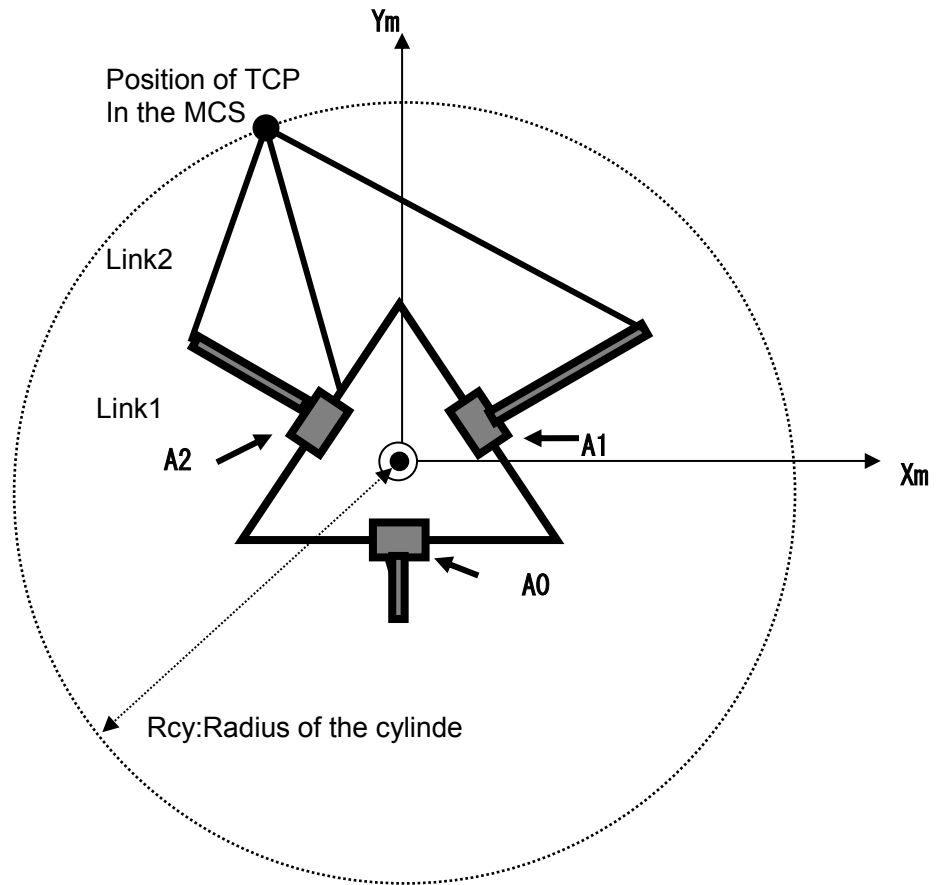
This condition can be judged by the following formula. In addition, β means the angle between the Link1 and Link 2.

$$L_m > 2 \cdot \cos(\beta) \cdot L_f$$

For example, the angle β is 30[deg] when the TCP of robot is reached to the top of the set workspace((xm, ym, zm) = [0,0, Zu]). In this case, the length of L1 and L2 must satisfy the following condition.

$$\therefore L_m/L_f > 1.732 \quad (\text{round off to three decimal places})$$

2. Relationship between kinematics and parameters cylinder radius workspace(Rcy)
 TCP of robot is necessary to reach on the cylinder Rcy is set of workspace, as shown in the figure below.



It can judge, by confirming the satisfy of the following relationship between the kinematics parameters (Rf, Rm, Lf, and Lm[mm]) and cylinder radius of the workspace(Rcy [mm]).

$$| R_f - R_m | \geq K_r * L_f$$

Parameters Kr and d means as below.

$$K_r = -\cos(\pi - \arctan(\tan(d) / \cos(60[\text{deg}])))$$

$$d = \arccos((L_m / R_{cy}) / \cos(30[\text{deg}])))$$

For example, Rcy = 1000[mm] and Le = 2000[mm] are set to kinematics parameters. In this case, the kinematics parameters Rf, Re and Lf must satisfy the following condition.

$$\therefore | R_f - R_m | \geq 0.333 * L_f$$

Intermediate calculation value is as below.

$$d = 0.955 \quad (\text{round off to three decimal places})$$

$$K_r = 0.333 \quad (\text{round off to three decimal places})$$

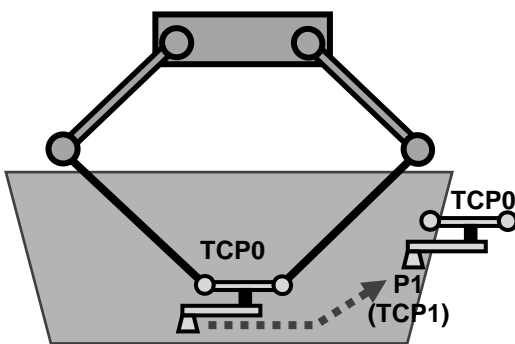
3.3.2.6 Workspace with Robot Tools

Workspace check is always performed for the robot TCP.

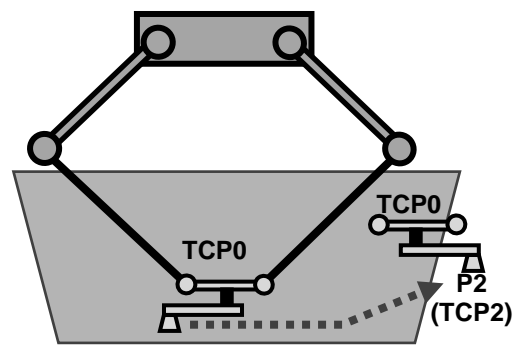
Precautions for Correct Use

- In case that a robot tool is selected, an error will occur when the robot TCP is outside the workspace (P1 in below figure), even if the active TCP is inside the workspace.

An error will not occur when the robot TCP is inside the workspace (P2 in below figure), even if the active TCP is outside the workspace.



Error will occur



No error

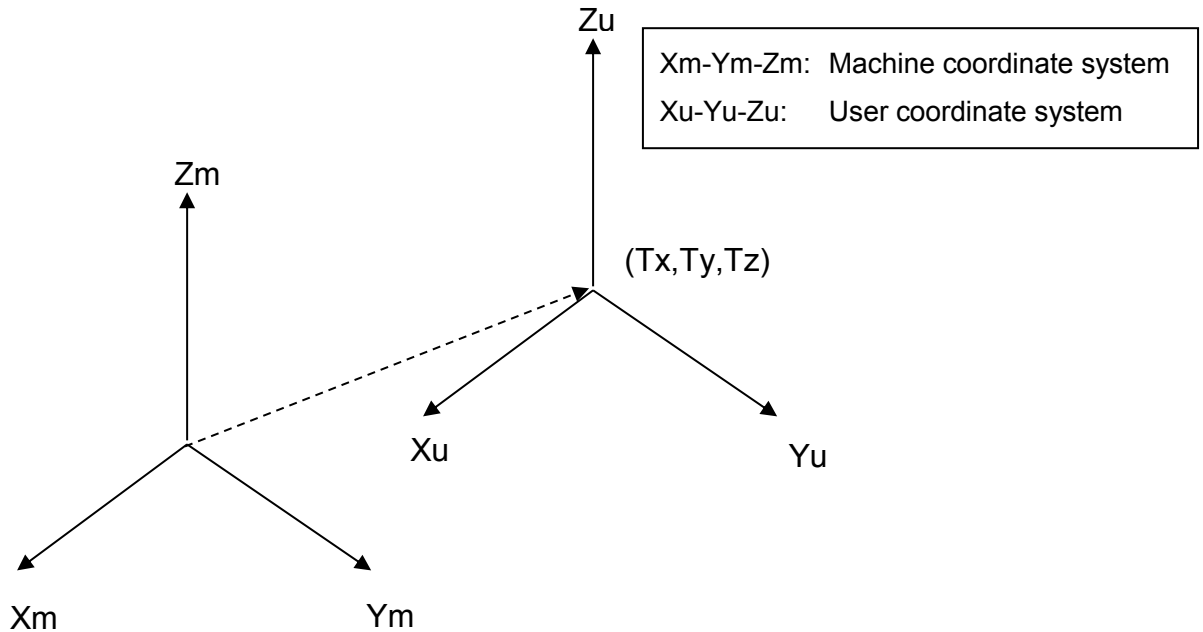
3.3.3 User Coordinate System

The home position (Tx, Ty, Tz, Rx, Ry, Rz) of the user coordinate system is set by the MC_DefineCoordSystem instruction (described later).

The home position to be set is described below.

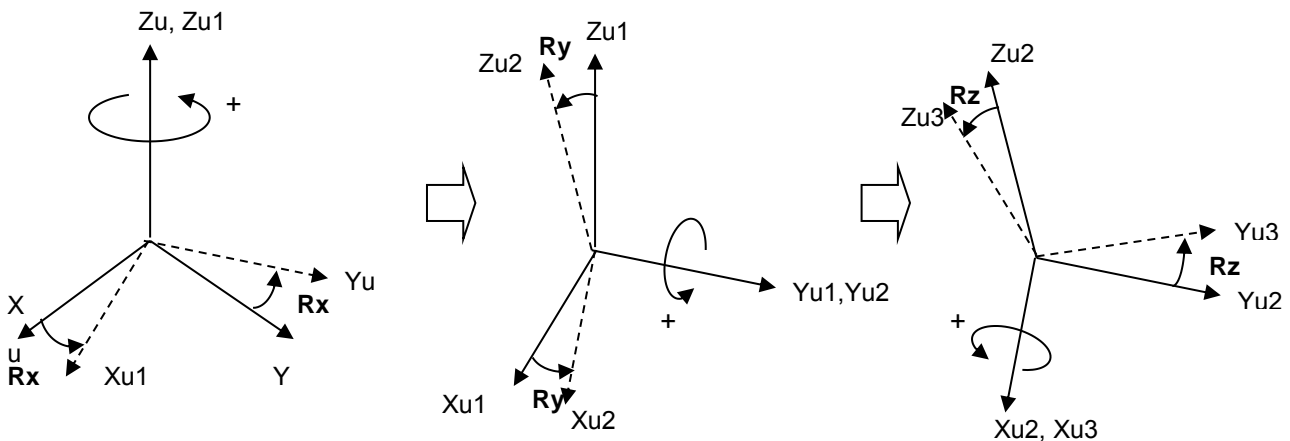
Setting of Position (Tx,Ty,Tz)

First, the home of the user coordinate system is set by the distance (Tx,Ty,Tz) from the home of the machine coordinate system.



- Setting of Rotation (Rx,Ry,Rz)

The rotation around Z axis Rz is set. The positive direction is the one in which the right hand is twisted in direction from Xu axis to Yu axis. In the Xu1-Yu1-Zu1 user coordinate system that is newly created by the above rotation, the rotation around Yu1 axis Ry is set. The positive direction is the one in which the right hand is twisted in direction from Zu1 axis to Xu1 axis. In the Xu2-Yu2-Zu2 user coordinate system that is newly created by the above rotation, the rotation around Xu2 axis Rx is set. The positive direction is the one in which the right hand is twisted in direction from Yu2 axis to Zu2 axis.





Precautions for Correct Use

- Each UCS is defined against a valid MCS.
It means that each UCS of one robot must be defined again by executing MC_DefineCoordSystem for each UCS, if the MC_SetKinTransform was executed again on that robot.

3.3.4 Monitoring

The following information can be monitored for the robot.

- Command current position and actual current position of robot's TCP on the machine/user coordinate systems
- Actual current velocity of TCP and actual current velocity of robot's TCP in X, Y, Z, Rx, Ry and Rz directions on the machine/user coordinate systems

The MC_GroupMon instruction (described later) is used for monitoring.

3.3.5 Time-specified Absolute Positioning Command

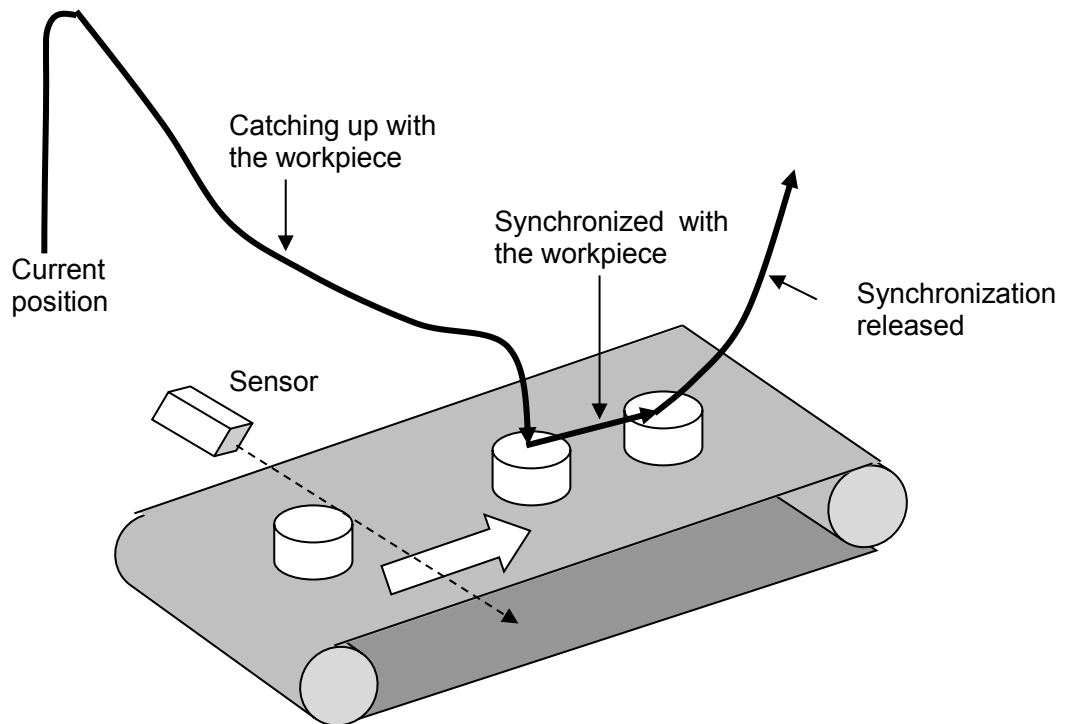
The command value to move the robot from the current position to the specified target position (X,Y,Z) in the specified time is created.

This command is used when the user wants to move the robot to a specific position.

The MC_MoveTimeAbsolute instruction (described later) is used to move the robot by the time-specified absolute positioning command.

3.3.6 Synchronization with Conveyor

The command value to make the robot catch up with a specified workpiece on a conveyor from the current position is created. Once the robot catches up with the workpiece, it moves in synchronization with the workpiece. When the robot moves to the next target position, the synchronization is released.



The MC_SyncLinearConveyor instruction (described later) is used to synchronize with the conveyor and MC_SyncOut instruction (described later) is used to release the synchronization.

3.3.6.1 Offset

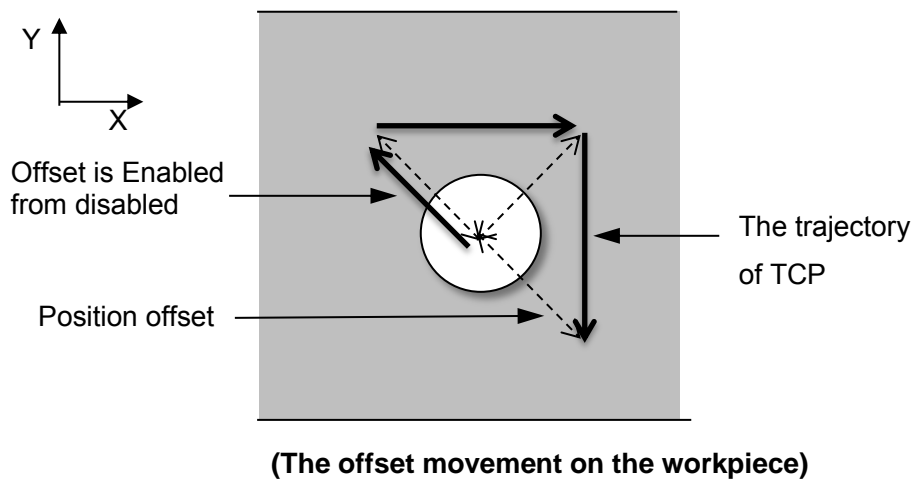
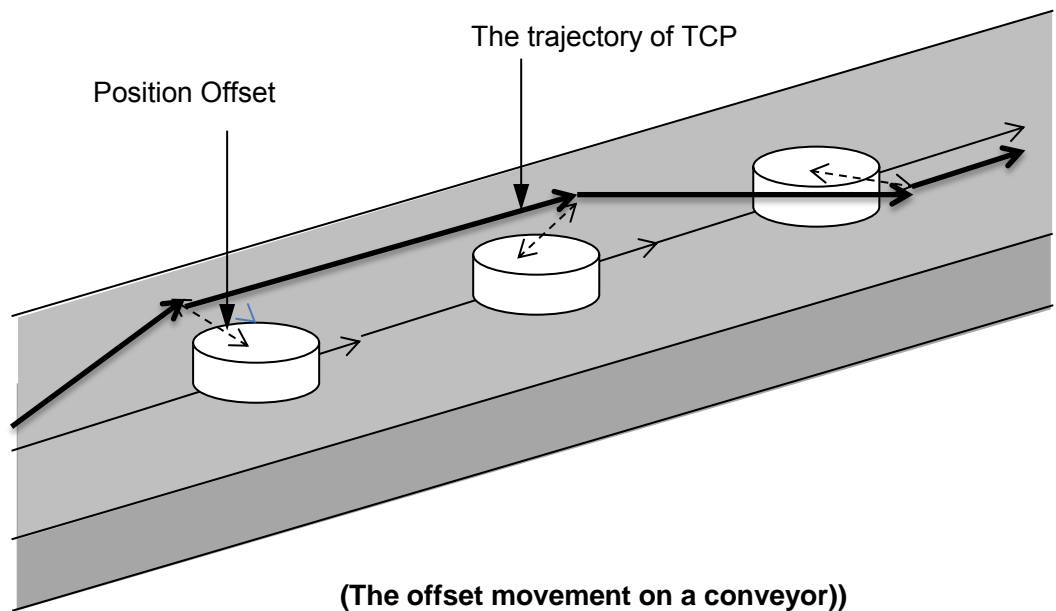
In Robot Ver. 1.01 or later, the offset functionality can be used.

The function adds offset positions to the robot position and orientation when it is synchronized with a conveyor.

Offsets are applied each control cycle to the output of the MC_SyncLinearConveyor. Therefore it is possible to keep synchronization with conveyor while other movements are performed.

Example)

Enable Offset	Disabled	Enabled	Enabled	Enabled	Disabled
Position offset X	-10	-10	10	10	0
Position offset Y	10	10	10	-10	0
Position offset Z	0	0 <td 0	0	0	





Precautions for Correct Use

- Offset functionality can only be used while the robot is synchronized with the workpiece.
If the offset is enabled before the synchronization has been achieved, an error occurs.
- When the offset is changed to enabled from disabled, the position offset at the moment is applied. Due to the value of position offset, TCP may move immediately. Pay attention that.
- When the offset is changed to disabled from enabled, the output is the same as the last position offset is kept.

To set position offset, a mixed format is used:

Cartesian coordinates (X,Y,Z) for Position and joint values (J4,J5 and J6) for the Orientation.

Name	Meaning	Units
PositionOffset[0]	X coordinate offset	mm
PositionOffset[1]	Y coordinate offset	mm
PositionOffset[2]	Z coordinate offset	mm
PositionOffset[3]	A3 coordinate offset (ACS)	deg
PositionOffset[4]	A4 coordinate offset (ACS)	deg
PositionOffset[5]	A5 coordinate offset (ACS)	deg

Position offset details depending on robot kinematics:

Kinematics type	Position offset					
	X:	Y:	Z:	J4:	J5:	J6:
	Position Offset[0]	Position Offset[1]	Position Offset [2]	PositionOf fset [3]	Position Offset[4]	Position Offset[5]
Delta3	O	O	O	X	X	X
Delta3R	O	O	O	O	X	X
Delta2	O	X	O	X	X	X

O: the parameter is enabled

X: the parameter is filtered: the value is ignored.

3.3.7 Inverse Kinematics

The machine coordinate system (MCS) position of the robot's TCP is transformed to the axis coordinate system (ACS) position of each axis of the robot.

This command is used when the user wants to move the robot on a specific path.

The MC_InverseKin instruction (described later) is used for inverse kinematics.

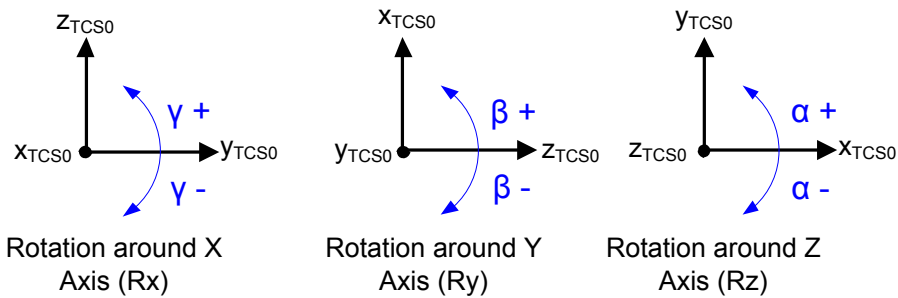
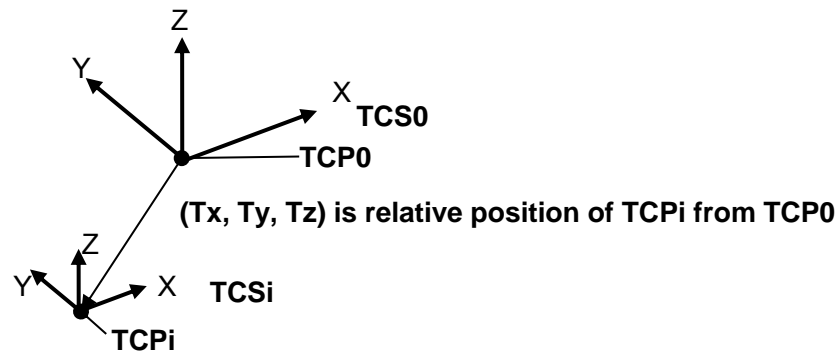
3.3.8 Robot Tool

In Robot Ver. 1.01 or later, the system provides Robot Tool support for a robot. It allows to define and to select robot tools to an existing robot. It allows to define and to select robot tools to an existing robot.

System allows definition up to 16 tools. The ID=0 is reserved to refer to the default TCP0. IDs from 1 to 16 are enabled to be defined for each robot. TCP with ID = i is described as TCPi.

Once a robot tool has been defined it is possible to select it.

To define a robot tool in the system, Tx, Ty, Tz, Rx, Ry and Rz should be defined with respect to the TCS0.



When a tool is defined and depending on type of kinematics some of the parameters are filtered.

Kinematics type	Parameters					
	Tx	Ty	Tz	Rx	Ry	Rz
Delta3	O	O	O	X	X	X
Delta3R	O	O	O	X	X	O
Delta2	O	X	O	X	X	X

O: the parameter is enabled

X: the parameter is filtered: the value is ignored.

When the robot tool is selected, the origin of its Tool Coordinate System will be used as the point of the robot that is commanded to move. Therefore robot will move from the current position to the target position referred to the selected robot tool.

Example)

Robot Tool definition:

TCP1= defined by (20.0mm, 0.0mm, -5.0mm, 0.0°, 0.0°, 0.0°)

Positions in MCS:

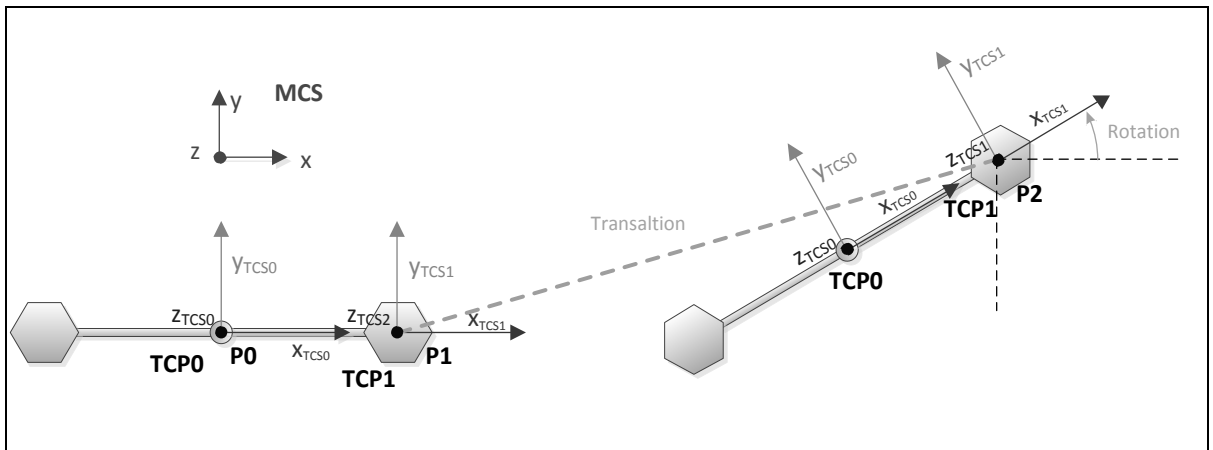
P0 (current position of TCP0) = (0.0mm, -20.0mm, -750.0mm, 0.0°, 0.0°, 0.0°)

P1 (current position of TCP1) = (20.0mm, -20.0mm, -755.0mm, 0.0°, 0.0°, 0.0°)

P2 (target position) = (70.0mm, 0.0mm, -750.0mm, 0.0°, 0.0°, 30.0°)

Operation:

Move to the target position P2, with Robot Tool ID1



Precautions for Correct Use

- Each Robot Tool is defined against a valid MCS.

It means that each Robot Tool of one robot must be defined again by executing MC_DefineToolTransform for each Robot Tool, if the MC_SetKinTransform was executed again on that robot.

3.3.9 Robot Jog

Robot Jog is supported in Robot Ver. 1.01 or later.

It operates jog movement along the specified direction.

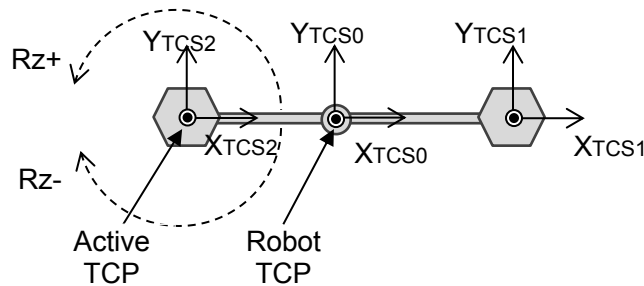
Jog Mode (MACHINE, USER, and TOOL) can be specified, that will set the point of the robot and the coordinate system used in to perform jogging.

The MC_RobotJog instruction (described later) is used for Robot Jog function.

- Rotations(Rx, Ry, Rz) in Robot Jog

Rotations are performed around the TCS of the selected robot tool. The position of the selected TCP will be kept.

Below figure shows the direction of rotation Rz around active TCP.



3.3.10 Robot maximum interpolation velocity and acceleration check

Maximum interpolation velocity, acceleration and deceleration of the robot TCP are checked as a background process in order to protect the robot.

Maximum interpolation velocity, acceleration and deceleration of the active TCP are checked in order to protect the product.

	<i>Robot TCP is selected</i>				<i>Active TCP is selected</i>			
	Max Velocity		Max Acc / Dec		Max Velocity		Max Acc / Dec	
	Robot TCP	Active TCP	Robot TCP	Active TCP	Robot TCP	Active TCP	Robot TCP	Active TCP
Pre-check in command	command	-	command	-	-	-	-	-
Every motion cycle	background command	-	background command	-	background	background command	background	background command

- background: the check is done as a background process.
- command: the check is done in the command



Precautions for Correct Use

- Maximum interpolation velocity/acceleration checks can be disabled by setting its maximum values to 0. This has been changed from in Robot Ver. 1.01.

3.3.10.1 Background maximum interpolation velocity, acceleration and deceleration check

The value of maximum interpolation velocity, maximum acceleration and maximum deceleration of the robot TCP can be set.

This check is activated when following conditions are true:

- 1) A kinematics is attached to a Group and the Group is enabled.
- 2) One of the configured maximum values is different than 0.

This check is performed for commanded values but not performed for feedback values. If the velocity/acceleration/deceleration exceeds the configured maximum values an error occurs for the Group. The Group status is set to ErrorStop and all movement commands associated with this Group are aborted.

The check is always performed for the TCP0, even if the movements are executed with a tool selected.

The value of maximum interpolation velocity, maximum interpolation acceleration and maximum interpolation deceleration are set by MC_SetKinTransform. In case a maximum value is set to 0, the check is disabled.

The default value of them are 0, therefore all checks are disabled. A different value should be set in order to enable the desired checks.

This check is only performed when there is a robotics command of movement type:

- MC_MoveTimeAbsolute
- MC_SyncLinearConveyor
- MC_SyncOut using sync-stop mode
- MC_RobotJog

And it is not done for robotics stopping commands neither non-robotics commands:

- MC_SyncOut using immediate-stop mode
- MC_GroupImmediateStop
- MC_GroupStop
- MC_GoupSyncMoveAbsolute

3.3.10.2 Command maximum interpolation velocity and acceleration check

There are two different checks of maximum interpolation velocity/acceleration/deceleration at a command level:

- Pre-check of the robot TCP velocity and acceleration only in case the Robot TCP is selected
- Every motion cycle check of the active TCP during the whole command execution

The commanded velocity/acceleration/deceleration of the active TCP is checked during the command execution against the configured value.

If the velocity/acceleration/deceleration exceeds the configured maximum values an error occurs for the Group. The Group status is set to ErrorStop and all movement commands associated with this Group are aborted.

This check is only performed when there is a robotics command of movement type:

- MC_MoveTimeAbsolute,
- MC_SyncLinearConveyor or
- MC_SyncOut using sync-stop mode

If the maximum interpolation velocity/acceleration values are set to 0.0 or after the command has finished, the corresponding check will be disabled.

■ Pre-check

Set the maximum interpolation velocity and acceleration limit for the trajectory executed on the command. The maximum interpolation velocity/acceleration/deceleration of the robot TCP that will be reached during the command execution is calculated before launching the movement.

If calculated maximum interpolation velocity/acceleration/deceleration is higher than the maximum value, an error occurs for the Group and all movement commands associated with this Group are aborted.

The pre-check is enabled only in case of the robot TCP is selected. If other tool is selected, the check is disabled.

■ Every motion cycle check

During the command execution, the instant velocity and the instant acceleration of the active TCP is calculated and compared against the maximum values.

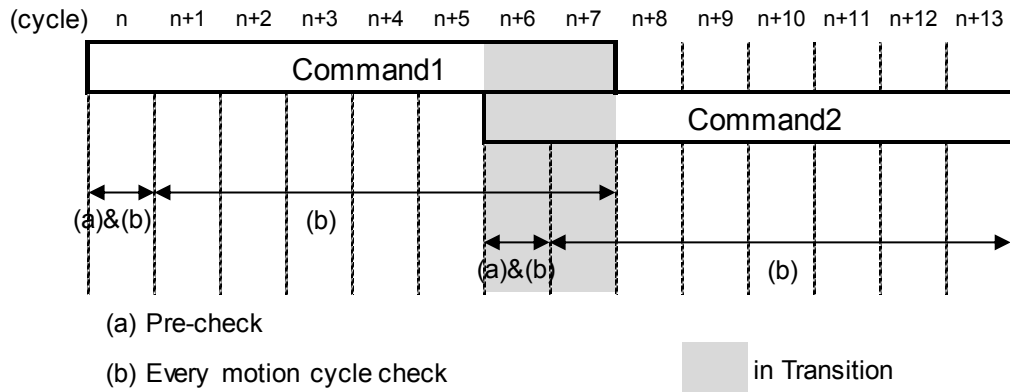
If calculated instant velocity/acceleration is higher than the maximum value, an error occurs for the Group and all movement commands associated with this Group are aborted.

Exception) At the first cycle of the command execution only instant velocity is checked.

■ Check during Transition

During the transition between two commands, the every motion cycle check is performed in each command.

Checking of maximum interpolation velocity/acceleration during transition is shown in the follow chart:



3.3.11 Multi-Execution (Buffered)

Multi-execution with Buffered is supported in Robot Ver. 1.01 or later.

The behavior is based on multi-execution of Motion Control Instructions.

3.3.12 Multi-Execution (Blending)

Multi-execution with Blending is supported in Robot Ver. 1.01 or later.

No robotics movements can be blended without using transitions. To blend robotics instructions transitions modes are needed in order to have a smooth transition required to control the robot.

You can specify the TransitionMode input variable to the robot control instruction to select a transition method to use between instructions.

3.3.12.1 Transition with given start remain time

When TransitionMode is set to Transition with given start remain time, the buffered instruction is superimposed on the current instruction. The superimposed area is specified by T_T which is set in the transition parameters.

The transition starts T_T milliseconds before the current instruction finishes, and the transition ends T_T milliseconds after the transition starts.



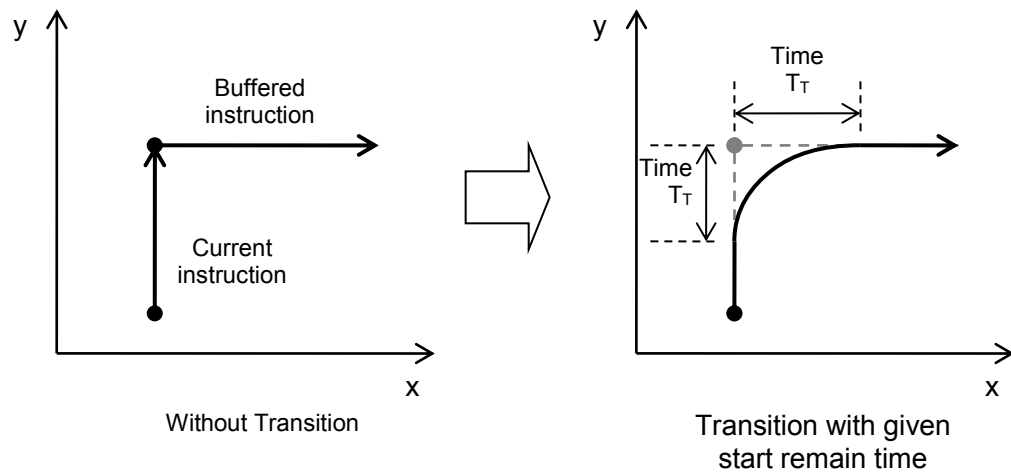
Precautions for Correct Use

- If T_T is not dividable by the controller cycle time, the time before transition will be extended to achieve whole number of controller cycles.
For example, when $T_T=5\text{ms}$ and controller cycle= 2ms , the time before transition will last 6ms instead of 5ms.
- When the remaining duration of the current instruction is smaller than T_T , the transition starts immediately and an observation occurs.
- When the duration of the buffered instruction is equal or smaller than T_T , an error occurs.
- The transition overlaps execution of 2 commands, then resulting movement, velocity, acceleration and deceleration are different from when executing single command.
Velocity error or acceleration error may occur when using Transition even if they did not occur without Transition.
- When MC_SyncLinearConveyor is executed as the buffered command and all conditions described below are true, an error occurs:
 - The direction of the current command and it of the buffered command are opposite on the Z coordinate or the rotational axis.
In Addition, in case of rotational axis, Direction input of the buffered command is ‘_mcPositiveDirection’ or ‘_mcNegativeDirection’.
 - The movement on the coordinate which fulfills previous condition finishes before the end of transition.

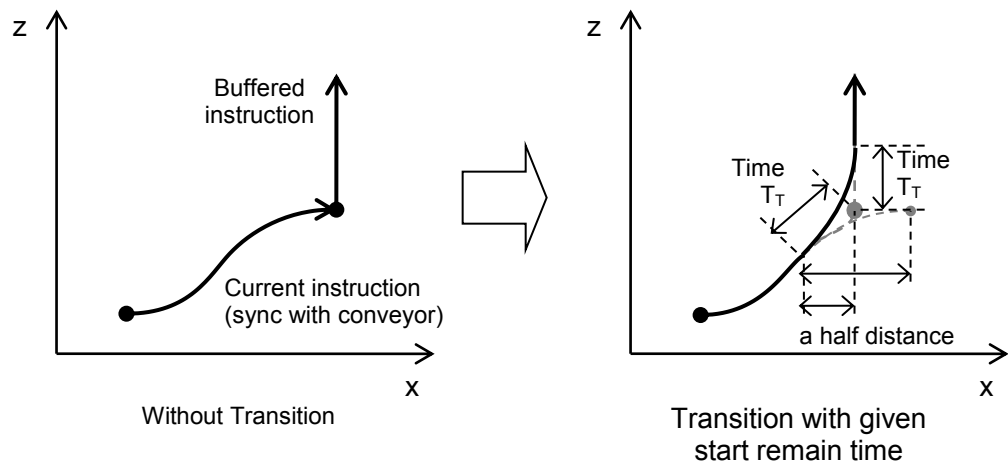
- If Direction Input is set to '_mcNoDirection' when MC_SyncLinearConveyor is executed as the buffered command with Transition, only the target rotation around Z axis of Delta-3R is guaranteed but it might rotate to different directions while moving to the target. Pay attention to that, because also an "overshoot" can happen i.e. rotation around Z axis can rotate over the target rotation and back in some situations

The trajectory of TCP is showed in following figures:

- Two Time-specified Absolute Positioning Commands:



- On the coordinate which is synchronizing with the conveyor before transition:
 The velocity profile of conveyor direction for current instruction is changed to decelerating profile, and the buffered instruction is superimposed on it. The area of decelerating profile is half as big as $T_T \cdot V_s$ (initial velocity of transition).
 This decelerating profile is performed with an appropriate jerk in order to avoid jump of acceleration.



3.3.12.2 Transition with given start height

When TransitionMode is set to Transition with given height, the buffered instruction is superimposed on the current instruction. The superimposed area is specified by T_H which is set in the transition parameters. Transition starts when the height on coordinate Z is higher than T_H . The transition ends at same timing as when the current command without the transition finishes.



Precautions for Correct Use

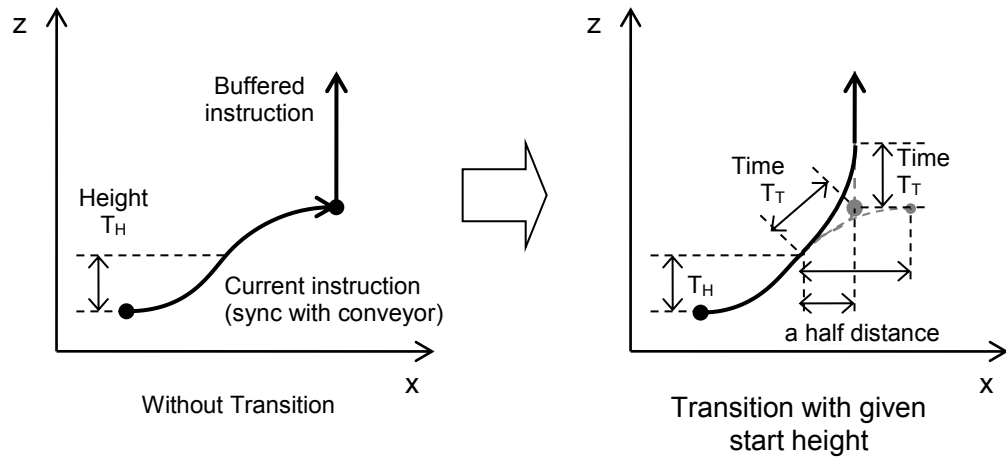
- When the passed height of the current instruction is higher than T_H , the transition starts immediately and an observation occurs.
- When the duration of the buffered instruction is equal or smaller than resulting T_T , an error occurs.
- The transition overlaps execution of 2 commands, then resulting movement, velocity, acceleration and deceleration are different from when executing single command. Velocity error or acceleration error may occur when using Transition even if they did not occur without Transition.
- When MC_SyncLinearConveyor is executed as the buffered command and all conditions described below are true, an error occurs:
 - The direction of the current command and it of the buffered command are opposite on the Z coordinate or the rotational axis.
In Addition, in case of rotational axis, Direction input of the buffered command is '_mcPositiveDirection' or '_mcNegativeDirection'.
 - The movement on the coordinate which fulfills previous condition finishes before the end of transition.
- If Direction Input is set to '_mcNoDirection' when MC_SyncLinearConveyor is executed as the buffered command with Transition, only the target rotation around Z axis of Delta-3R is guaranteed but it might rotate to different directions while moving to the target. Pay attention to that, because also an "overshoot" can happen i.e. rotation around Z axis can rotate over the target rotation and back in some situations

The trajectory of TCP is showed in following figure:

- On the coordinate which is synchronizing with the conveyor before transition:

The velocity profile of conveyor direction for current instruction is changed to decelerating profile, and the buffered instruction is superimposed on it. The area of decelerating profile is half as big as $T_T \cdot V_s$ (initial velocity of transition).

This decelerating profile is performed with an appropriate jerk in order to avoid jump of acceleration.



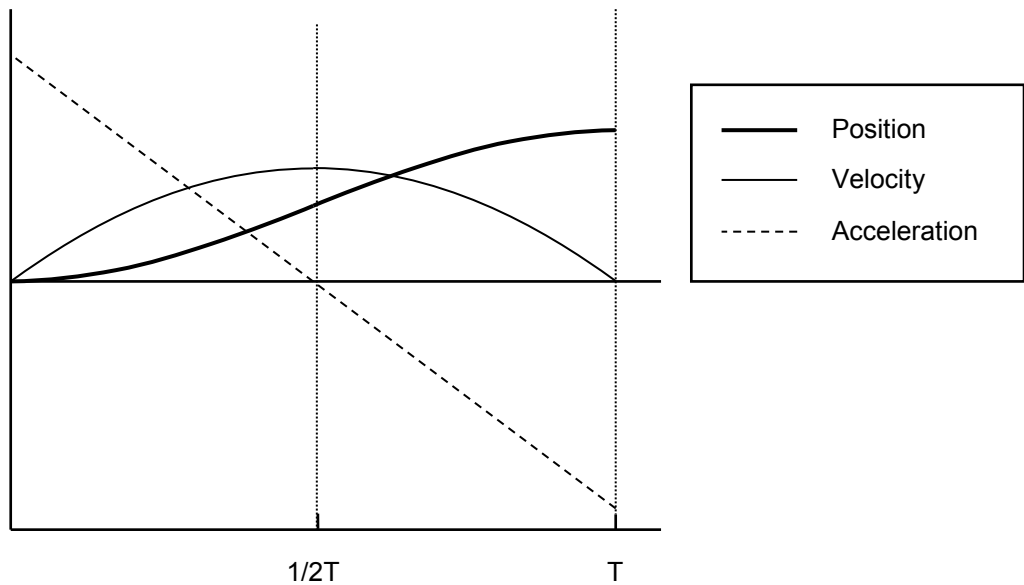
3.3.13 Trajectory types of time-specified move

With a movement in the specified time, the trajectory type can be selected.

3.3.13.1 Polynomial function (3rd degree)

It is a polynomial function of 3rd degree with start/end velocity and acceleration are 0.

In Robot Ver. 1.00, only this type can be selected.



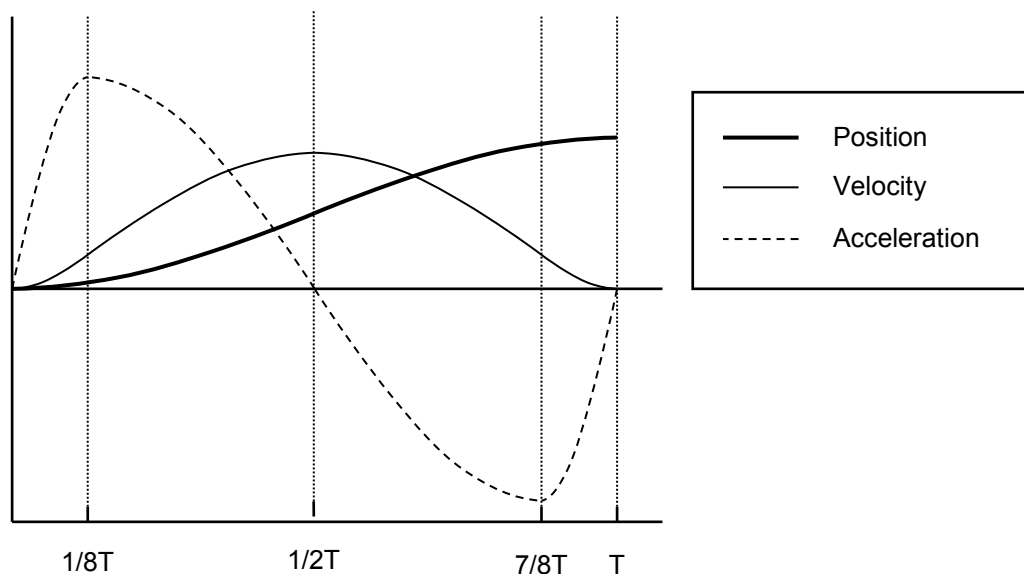
3.3.13.2 Modified Sine

Modified sine is based on a combination of sinusoidal trajectories. It allows controlling acceleration and belongs to Universal CAM Curve family.

This trajectory type is recommended to be used with Transitions enabled.

Used for high-velocity, medium-load equipment. Used to reduce the effect on the motor for load variations.

This type can be selected in Robot Ver. 1.01 or later.

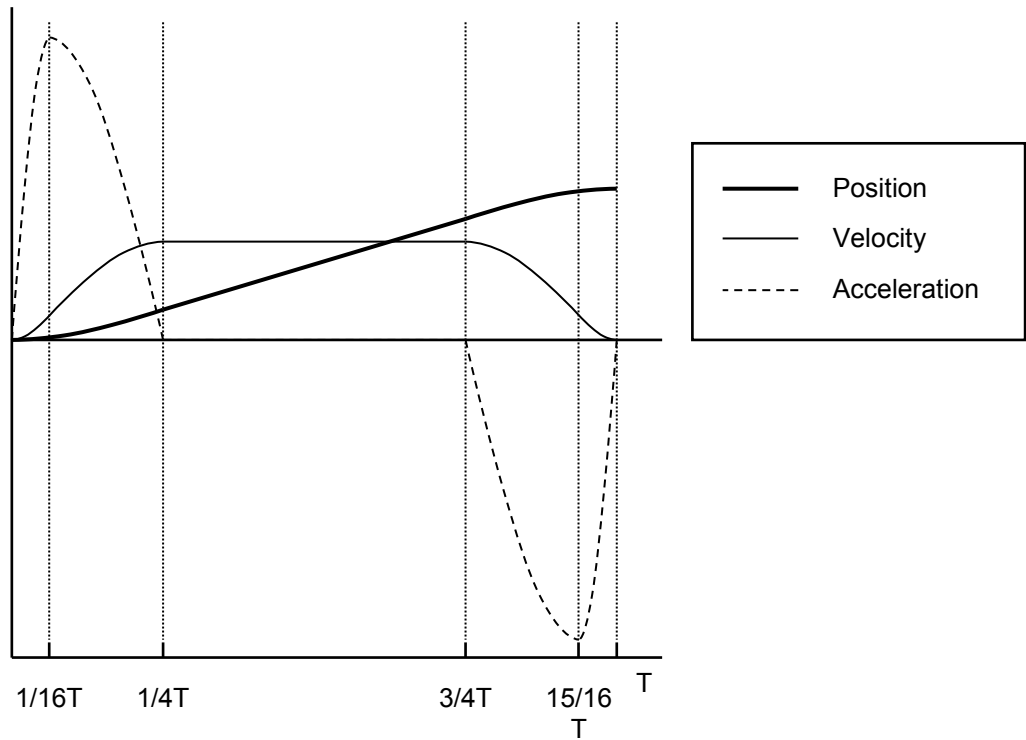


3.3.13.3 Modified Constant Velocity

Modified constant velocity is based on combination of sinusoidal and linear trajectories. It has a constant velocity during one half of trajectory time. It belongs to Universal CAM Curve family.

Used for medium-velocity, high-load equipment.

This type can be selected in Robot Ver. 1.01 or later.



4 Robot Function Settings

In order to use the robot function NJ501-4 □□□, you will need to enable the Robot Option SysmacStudio. After this, do the setting of various parameters by 2 steps as below.

- Set an axes group for the robot (Setting on Sysmac Studio)
- Set link length and other parameters for the robot (Setting on the user program)

This chapter describes this procedure. The details of the procedure and details of the parameters to set, Please refer the Related Manuals.

4.1 Sysmac Studio Robot Option

4.1.1 Procedures of Registering Robot Option

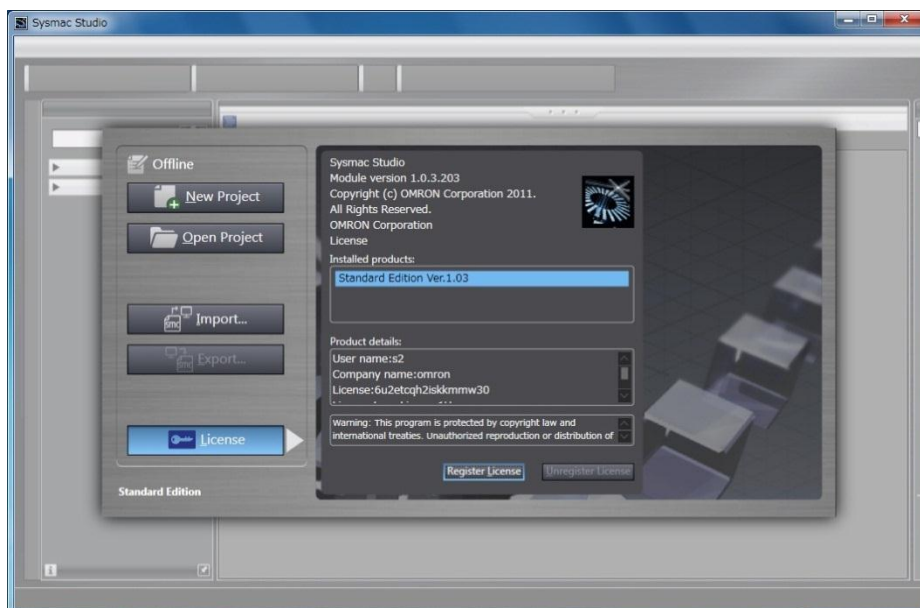
After following procedure have done to Sysmac Studio, Robot option will be available for use.

The project that robot instructions are used is only available in SysmacStudio this Robot option has been enabled. In addition, even if this robot option disabled, the project that robot instructions are used do not erase.

1) Install Sysmac Studio with Standard Edition. Refer Sysmac Studio Version 1 Operation Manual(Cat. No. W504) for this procedure.

Note: If Sysmac Studio is Trial Edition, you cannot register Robot Option.

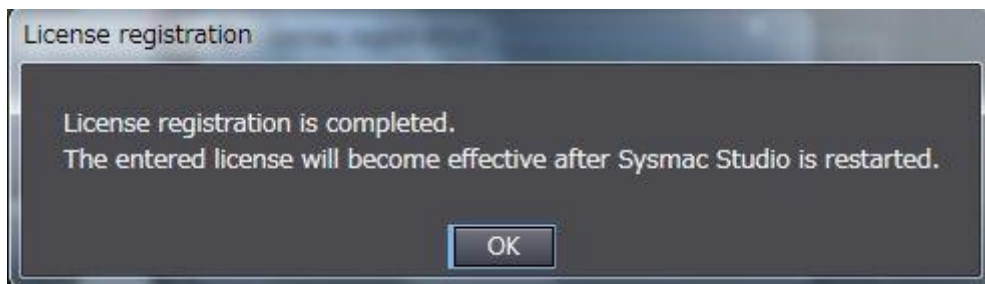
2) Launch Sysmac Studio, press [License] button in left side of start window and press [Register License] button.



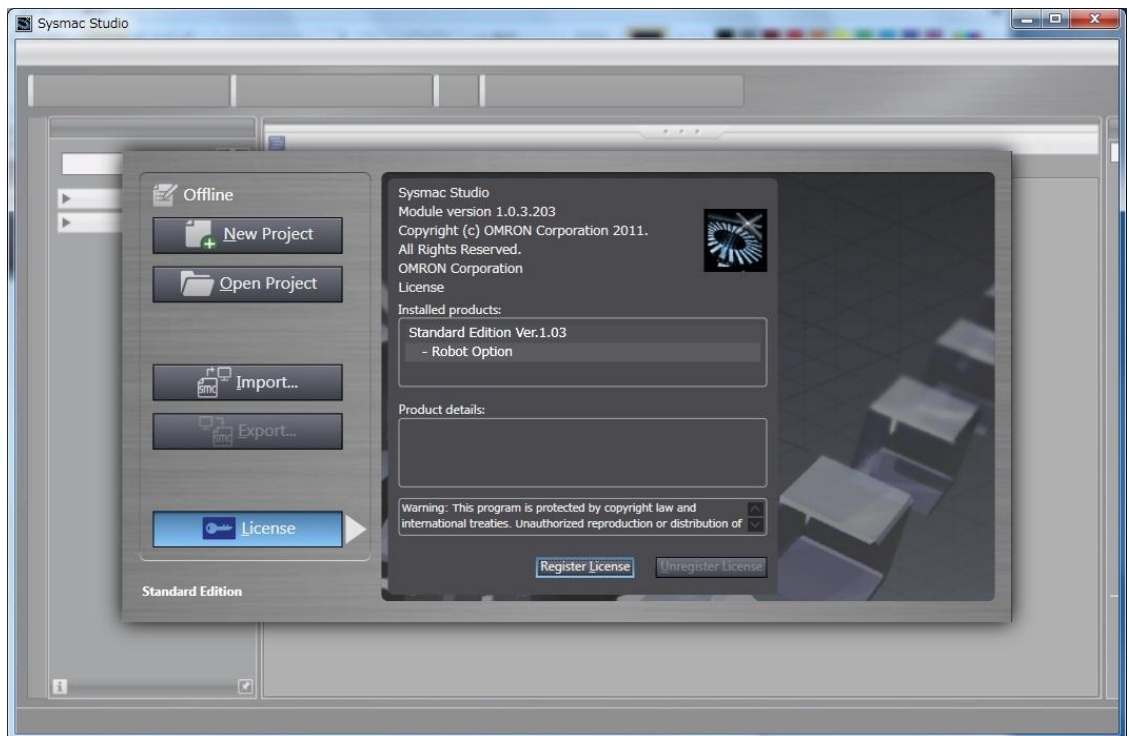
3) Enter the license number of Robot option and press [Register] button.



4) The following message is shown after succeeding to register the license.



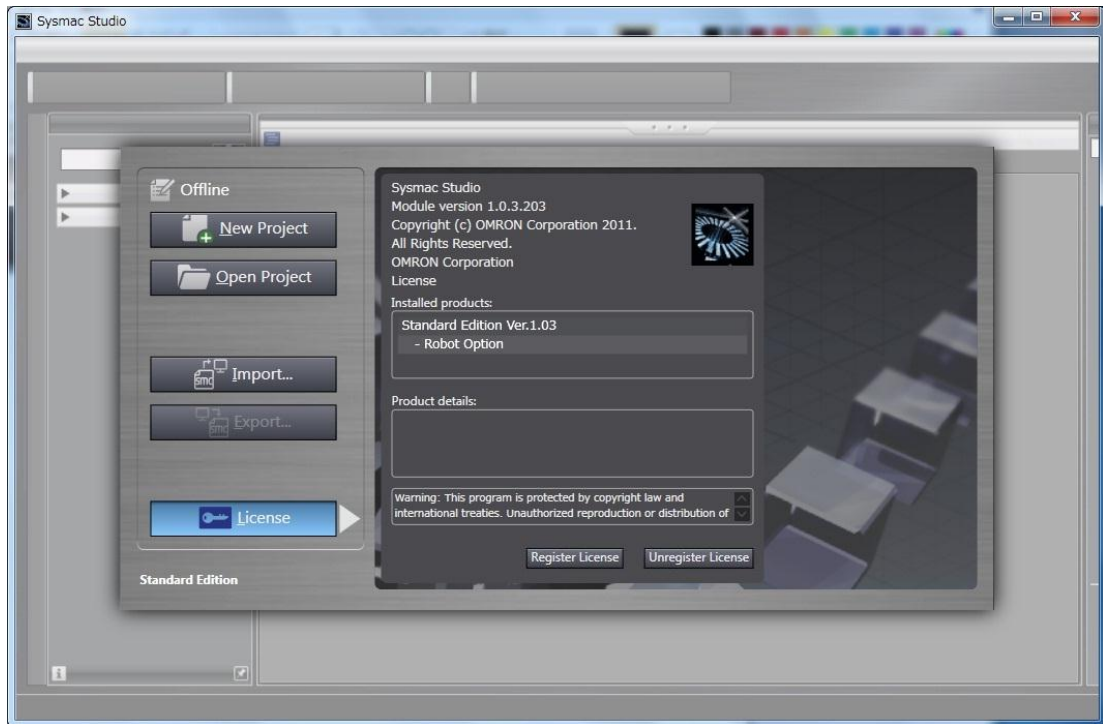
5) Confirm that Robot Option is displayed like following window and restart Sysmac Studio.



4.1.2 Procedures of Unregistering Robot Option

When you will move the license of Robot Option to another PC, Please unregister Robot Option with following steps.

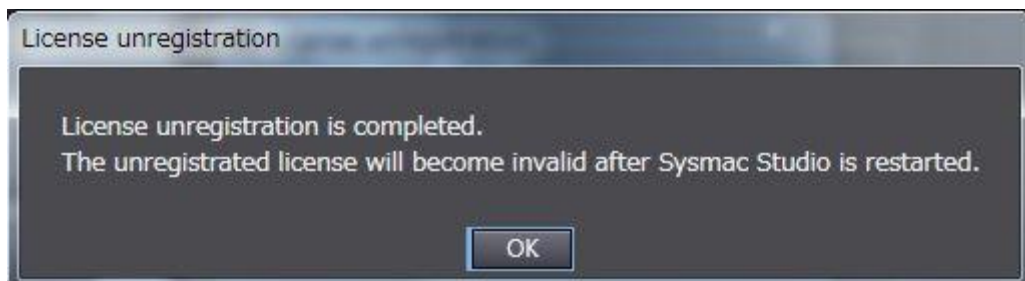
1) Launch Sysmac Studio, press [License] button in left side of start window and press [Unregister License] button.



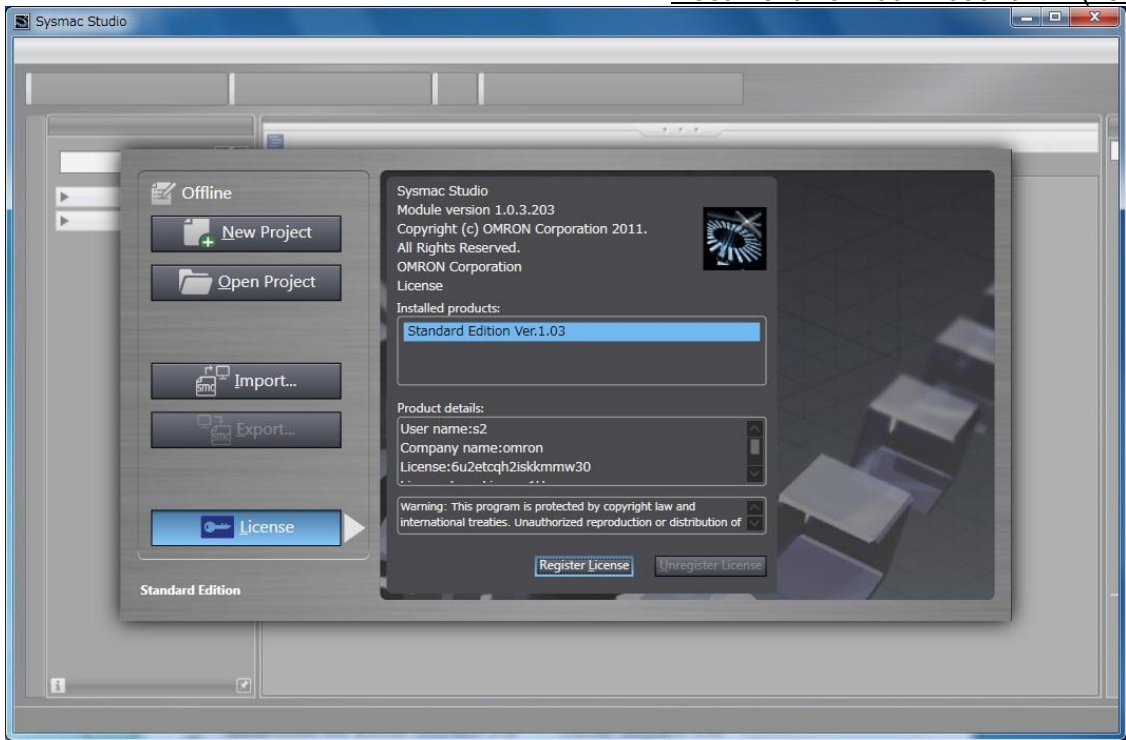
2) Enter the license number of Robot option and press [Unregister] button.



3) The following message is shown after succeeding to register the license.



4) Confirm that Robot Option is not displayed like following window and restart Sysmac Studio.



- Unit Conversion Settings

Unit of Display: Select degree Unit
 Command pulse count per motor rotation: Set to design value
 Work travel distance per motor rotation: Set to design value

- Operation Settings

< Maximum velocity/acceleration/deceleration: >
 Maximum Velocity*1: Set to design value
 Maximum Jog Velocity: Any value
 Maximum Acceleration: Set to design value
 Maximum Deceleration: Set to design value
 Acceleration/Deceleration Over: Any value
 Operation Selection at Reversing: Any value

*1 : In the case over this value, when move the robot, an error occurs and stop immediately. Other parameters are not used in the robot.

< Torque > Any value

< Observation > Any value

- Other Operation Settings Any values

- Limit Settings

Set in the range from -180 to 180 degrees.

- Homing Settings

Homing method: Select Zero position preset

- Position Count Settings

Count mode: Select Linear mode

Encoder type: Select Absolute encoder (ABS)

When using MC_SyncLinearConveyor instruction (described later) to synchronize with a conveyor, you need to setup the conveyor axis. In that case, please select the Rotary Mode of the Count Mode.

The axes group setting for the robot is completed by the above procedure. Make settings for the Controller as required, and then save the project and transfer it to the Controller.

4.3 Setting of Link Length and Other Parameters for Robot (Setting on User Program)

To move the robot, you will need to the following settings by using the user program.

- While the homes of all axes are defined, execute MC_SetKinTransform instruction to set the link length and other parameters and workspace for the robot.

5 Homing

Define the machine home of the robot.

For example, the procedure for Delta3 is following:

- 1) Move and fix each motors of the robot to their machine home.
Refer to the user's manual of the robot.
- 2) With SysmacStudio, reset the multi-rotation data of the absolute encoder to 0 using the absolute encoder setting function.
- 3) With SysmacStudio, set the current position to 0 by homing of the MC Test Run function with 'Zero position preset' homing method.

By doing so, the compensation value for the absolute encoder to set the current position to 0 is stored in the non-volatile memory of the NJ-series CPU Unit.

Refer to the NJ-series CPU Unit Motion Control User's Manual(Cat. No. W507) for the operating procedure of the MC Test Run function.

Instead of the homing of the MC Test Run function, it is possible to use the MC_Home instruction in the user program.

- 4) Repeat the same operation for all axes of the robot, and all axes are homed.

6 Wiring/Settings by Sysmac Studio

Next, the wiring of each robot axis and electronic gear settings of each axis are checked.

- Wiring check:
 - Are the limit sensor and other items wired correctly?
 - Does each axis move downward when a positive-direction command value is given to the axis? (Refer to Section 3.2.1.2 (1).)
- Electronic gear setting check:
 - Can it move to the specified angle (e.g. 90 degree)?

Use the MC Test Run function of Sysmac Studio or user program for the check.

When a moving frame is mounted, remove it to allow each axis to move freely.

Refer to the NJ-series CPU Unit Motion Control User's Manual(Cat. No. W507) for the operating procedure of the MC Text Run function.

When the axes move as expected, turn OFF the Servo for all axes and mount the moving frame.

7 Instructions

7.1 Outline

7.1.1 Instruction List

The following instructions can be used with NJ501-4*00. The robot instructions are classified as axes group instructions.

Category	Motion Control Instruction		Version
	Instruction	Name	
Robot Instruction	MC_SetKinTransform	Set Kinematics Transform	
	MC_DefineCoordSystem	Define Coordinate System	
	MC_GroupMon	Monitor Axes Group	
	MC_MoveTimeAbsolute	Time-specified Absolute Positioning	
	MC_SyncLinearConveyor	Start Conveyor Synchronization	
	MC_SyncOut	End Synchronization	
	MC_InverseKin	Inverse Kinematics	
	MC_DefineToolTransform	Define Tool Transform	Robot Ver. 1.01 or later
	MC_RobotJog	Robot Jog	Robot Ver. 1.01 or later

7.1.2 Unavailable Instructions

The following table shows if axes group instructions can be executed or not, when a kinematics is set to the axes group.

Group Instruction	After Kinematics setting	Before Kinematics setting
MC_GroupEnable	O	O
MC_GroupDisable	O	O
MC_GroupReset	O	O
MC_GroupStop	O	O
MC_GroupImmediateStop	O	O
MC_GroupReadPosition	O	O
MC_GroupSyncMoveAbsolute	O	O
MC_MoveLinear	X	O
MC_MoveLinearAbsolute	X	O
MC_MoveLinearRelative	X	O
MC_MoveCircular2D	X	O
MC_ChangeAxesInGroup	X	O
MC_GroupSetOverride	X	O
MC_SetKinTransform	O	O
MC_DefineCoodSystem	O	X
MC_DefineToolTransform	O	X
MC_GroupMon	O	X
MC_InverseKin	O	X
MC_MoveTimeAbsolute	O	X
MC_SyncLinearConveyor	O	X
MC_SyncOut	O	X
MC_RobotJog	O	X

O: The instruction can be executed

X: An error occurs when the instruction is executed



Precautions for Correct Use

- When run the other instructions except MC_GroupSetOverride, an error occur and immediately stop the axis. Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) Section 7.2. for details.
- MC_WriteAxisParameter can not be executed when a kinematics is set to the axes group.

7.1.3 Starting and Stopping a Robot

When moving a robot, use the MC_MoveTimeAbsolute, MC_SyncLinearConveyor, MC_SyncOut, or MC_GroupSyncMoveAbsolute instruction.

When stopping a running robot, use the MC_SyncOut, MC_GroupStop, or MC_GroupImmediateStop instruction.

Please note that the robot is stopped immediately regardless of the input variable settings when the MC_GroupStop instruction is used to stop a robot.

Also, please note that the robot is stopped immediately when any one of the following occur during robot operation.

- Error of Inside of MC Function Module
- Error of EtherCAT Master Function Module
- Error of Built-in EtherCAT communications port hardware
- Error of EtherCAT slaves

7.1.4 Axes Group Variables

During moving axes in robotics instructions, the following variables will be zero and therefore they can't be used for monitoring purpose: MC_Group□□□.Cmd.Vel, MC_Group□□□.Cmd.AccDec □□□ 000~063

When a robot is stopped due to an error or a stop instruction, the commanded value is output to MC_Group□□□.Cmd.Vel

7.2 How to specify a target position

7.2.1 Position input variable

Target position is specified by Position input variable (X, Y, Z, Rx, Ry, Rz). Position details depending on robot kinematics:

Kinematics type	Position					
	X: Position [0]	Y: Position [1]	Z: Position [2]	Rx: Position [3]	Ry: Position [4]	Rz: Position [5]
Delta 3	O[mm]	O[mm]	O[mm]	X	X	X
Delta 3 R	O[mm]	O[mm]	O[mm]	X	X	O[deg]
Delta 2	O[mm]	X	O[mm]	X	X	X

O: the parameter is enabled

X: the parameter is filtered: the value is ignored.

7.2.2 Control of rotation

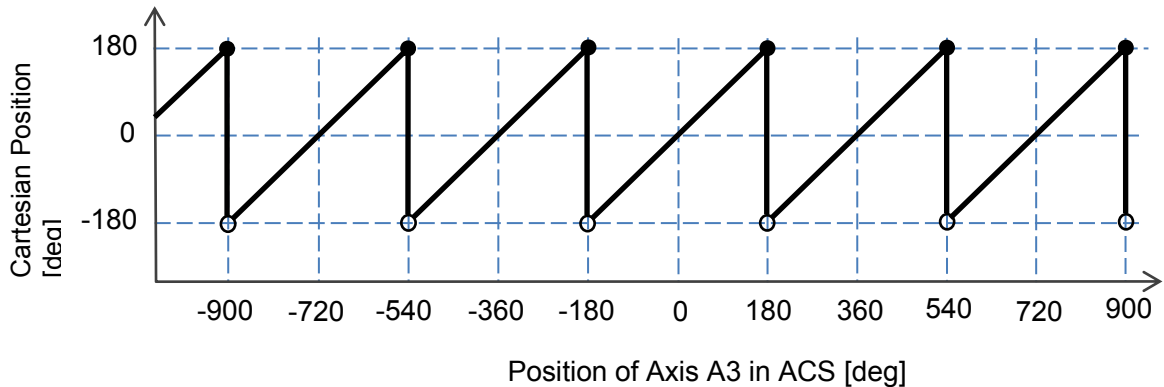
The control of rotation is specified by the rotation around Z and rotational direction.

This functionality is enabled only when Delta3R Kinematics is attached.

7.2.2.1 Rotation around Z

The rotation around Z is specified in the range from -180[deg] to 180[deg].

The lower limit (-180[deg]) is not reachable, therefore the kinematics range of rotation around Z (Rz) is (-180, 180].



The position of Axis A3 has a finite range, and its range is same as the range of the axis which count mode is linear in the Motion Control Function Module.

If this range is exceeded for robot operations, home position is lost an error is raised for the AxesGroup. To avoid exceeding the axis range and lose home position, user can set software limits of the axis.

7.2.2.2 Direction

Direction input specifies the direction of rotational axis in Cartesian coordinate system.

Direction	Input value
Positive Direction	_mcPositiveDirection
Negative Direction	_mcNegativeDirection
No Direction specified	_mcNoDirection

(1) `_mcPositiveDirection`

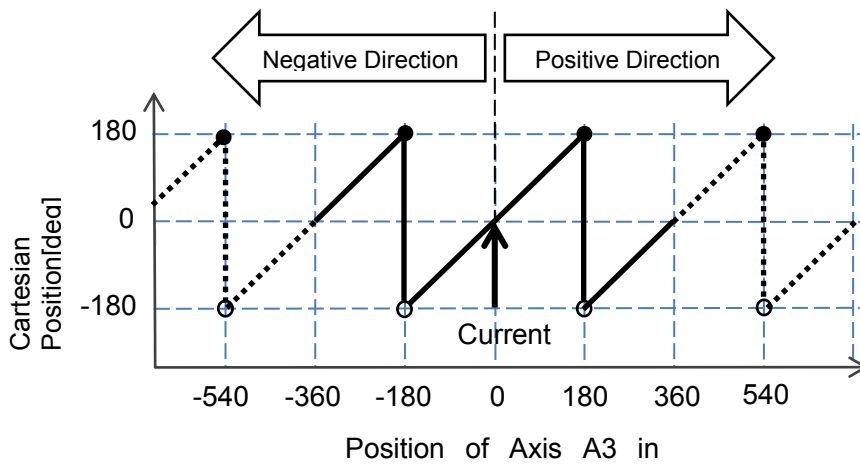
Rotation axis is set to move in the positive direction (counter clockwise) so that the robot achieves the target position set in the Position input.

If target position(Rz) is less than current orientation around Z, the rotational axis will move over 180[deg].

(2) `_mcNegativeDirection`

Rotation axis is set to move in the negative direction (clockwise) so that the robot achieves the target position set in the Position input.

If target position(Rz) is greater than current orientation around Z, the rotational axis will move over 180[deg].



(3) `_mcNoDirection`

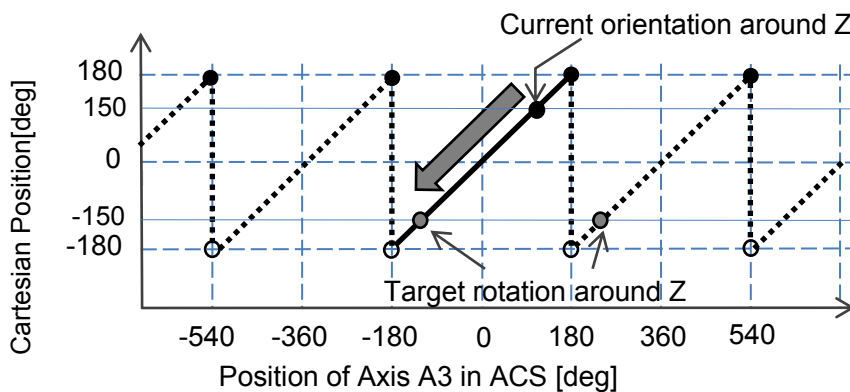
Robot will move the rotation axis so that it achieves the target position keeping the rotation around Z position in the same turn of its kinematics range.

Example) case that the rotational axis will move over 180[deg]

Current orientation around Z: 150 [deg]

Target rotation around Z: -150 [deg]

Direction: `_mcNoDirection`

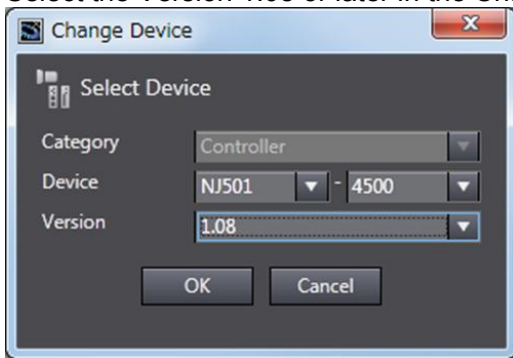


7.3 When using with Robot Ver. 1.01 a project which was created in Robot Ver. 1.00

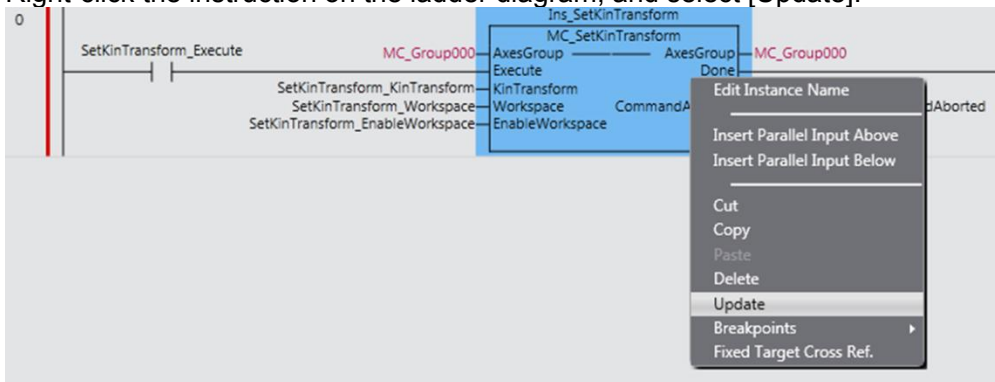
To load a project which is created in Robot Ver. 1.00 and change/modify with functions of Robot Ver. 1.01, operate following procedure.

- It is required only when robot instructions are used in ladder diagrams.
- Following robot instructions require the procedure. Other instructions do not.
 MC_SetKinTransform, MC_GroupMon, MC_MoveTimeAbsolute, MC_SyncLinearConveyor, MC_InverseKin

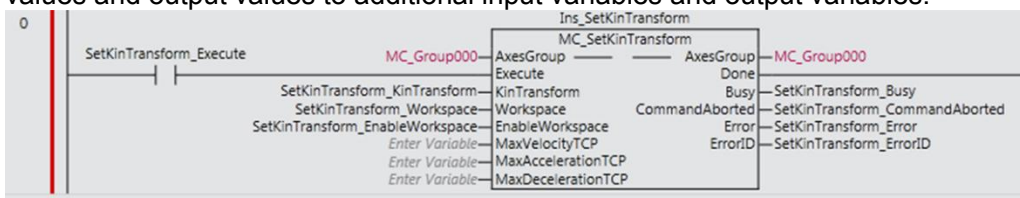
- (1) Load the project in SysmacStudio Ver. 1.09 or later.
- (2) Right-click the Controller Icon and select **Change Device** from the menu. Or, select the **Change Device** from the Controller Menu.
- (3) Select the Version 1.08 or later in the Change Device Dialog Box.



- (4) Right-click the instruction on the ladder diagram, and select [Update].



- (5) The instruction on the ladder diagram is updated to the specification of Robot Ver. 1.01. Set input values and output values to additional input variables and output variables.



- (6) Repeat operations (4) and (5) for all instructions which have to be modified.

7.4 Details

This section describes the robot instructions and the MC_GroupSyncMoveAbsolute instruction that can be used for robots. Refer to the NJ-series Motion Control Instructions Reference Manual(Cat. No. W508) for other instructions.

MC_SetKinTransform <Robot Ver 1.00>

The MC_SetKinTransform instruction sets robot control kinematics and workspace check for an axes group.

Instruction	Name	FB/ FUN	Graphic expression	ST expression
MC_SetKinTransform	Set Kinematics Transform	FB	<p>MC_SetKinTransform_instance</p>	<pre>MC_SetKinTransform_instance(AxesGroup:=parameter, Execute:=parameter, KinTransform:= parameter, Workspace:= parameter, EnableWorkspace:=parameter, Done =>parameter, Busy =>parameter, Error =>parameter, ErrorID =>parameter);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
Execute	Execute	BOOL	TRUE or FALSE	FALSE	The instruction is executed when <i>Execute</i> changes to TRUE.
KinTransform	Kinematics Transform	_sMC_KIN_REF	---	---	Specify the kinematics type, robot's link length and other items.
Workspace	Workspace Parameters	_sMC_WORKSPACE_REF	---	---	Specify the range of workspace.
EnableWorkspace	Enable Workspace	BOOL	TRUE or FALSE	TRUE	Specify whether to enable (TRUE) or disable (FALSE) the workspace function.

sMC_KIN_REF

Member variable	Meaning	Data type	Valid range	Function
KinType	Kinematics Type Selection	_eMC_KIN_TYPE	100:_mcDelta3Type1	Specify the kinematics type to set. Only the 100: <i>Delta3</i> is supported.
KinParam	Kinematics Parameters	ARRAY [0..7] OF LREAL	---	Specify the arm length and other items for the robot.
ExpansionParam	Expansion Parameters	ARRAY [0..7] OF LREAL	---	Specify the values for adjusting the rotation of the machine coordinate system (MCS).

sMC_WORKSPACE_REF

Member variable	Meaning	Data type	Valid range	Function
WorkspaceType	Workspace Type Selection	_eMC_WORKSPACE_TYPE	100:_mcDelta3Workspace	Specify the workspace type to set. Only the 100: <i>Delta3</i> is supported.
WorkspaceParam	Workspace Parameters	ARRAY [0..7] OF LREAL	---	Set the Volume of workspace.

● Output Variables

Name	Meaning	Data type	Valid range	Description
Done	Done	BOOL	TRUE or FALSE	TRUE when the setting is completed.
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
CommandAborted	Command Aborted	BOOL	TRUE or FALSE	TRUE when the instruction is aborted.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#0000 indicates normal execution.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
Done	When transformation is set successfully.	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one control period when <i>Execute</i> is FALSE.
Busy	When <i>Execute</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Done</i> changes to TRUE. When <i>Error</i> changes to TRUE.
CommandAborted	When the axis group is in an error state	<ul style="list-style-type: none"> When „Execute“ is TRUE and changes to FALSE One period after the „Execute“ became FALSE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	When the error is cleared.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
AxesGroup	Axes Group	_sGROUP_REF	---	Specify the axes group.

■ Function

- This instruction is used to set the kinematics and workspace check parameters for a specified axes group.
 - 1) For Kinematics: Kinematics type, robot's link length, and home adjustment parameter
 - 2) For workspace check: Workspace type and workspace volume



Precautions for Correct Use

- When using MC_GroupSyncMoveAbsolute, the Workspace is not checked.
Please pay attention to that.

- For Delta3, the following values are set for the variables defined by the _sMC_KIN_REF and _sMC_WORKSPACE_REF data types.

_sMC_KIN_REF

KinType

Name	Data type	Valid range	Description
KinType	_eMC_KIN_TYPE	100:_mcDelta3Type1	Only the 100: <i>Delta3</i> is supported.

KinParam

Name	Data type	Valid range	Description
KinParam[0]	LREAL	Positive number	Radius of the fixed frame <i>Rf</i> [mm] Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) Section 3.3.1 for details.
KinParam[1]	LREAL	Positive number	Link 1 <i>Lf</i> [mm]
KinParam[2]	LREAL	Positive number	Radius of the moving frame <i>Rm</i> [mm]
KinParam[3]	LREAL	Positive number	Link 2 <i>Lm</i> [mm]
KinParam[4]- [7]	LREAL	0	Reserved

ExpansionParam

Name	Data type	Valid range	Description
ExpansionParam[0]	LREAL	0 to 360	Rotation of Z axis θ [deg] Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) Section 3.3.1 for details.
ExpansionParam[1]-[7]	LREAL	0	Reserved

_sMC_WORKSPACE_REF

WorkspaceType

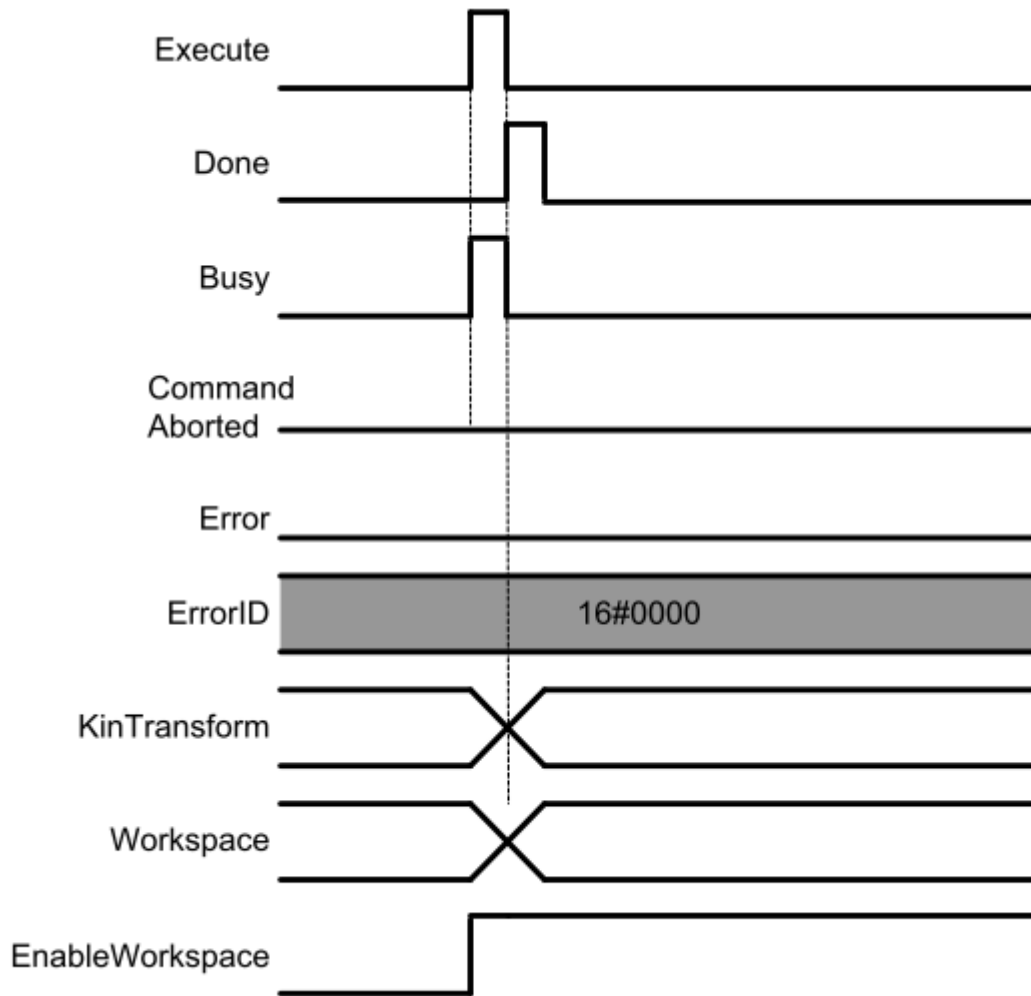
Name	Data type	Valid range	Description
WorkspaceType	_eMC_WORKSPACE_TYPE	100: _mcDelta3Workspace	Only the 100: <i>Delta3</i> is supported.

WorkspaceParam

Name	Data type	Valid range	Description
WorkspaceParam[0]	LREAL	Negative number or 0	Top surface position of cylinder <i>Zu</i> [mm] Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) <i>Section 3.3.2</i> for details.
WorkspaceParam[1]	LREAL	Positive number	Radius of the cylinder <i>Rcy</i> [mm]
WorkspaceParam[2]	LREAL	Positive number	Height of the cylinder <i>Hcy</i> [mm]
WorkspaceParam[3]	LREAL	Positive number	Radius of the frustum cone <i>Rco</i> [mm]
WorkspaceParam[4]	LREAL	Positive number	Height of the frustum cone <i>Hco</i> [mm]
WorkspaceParam[5]-[7]	LREAL	0	Reserved

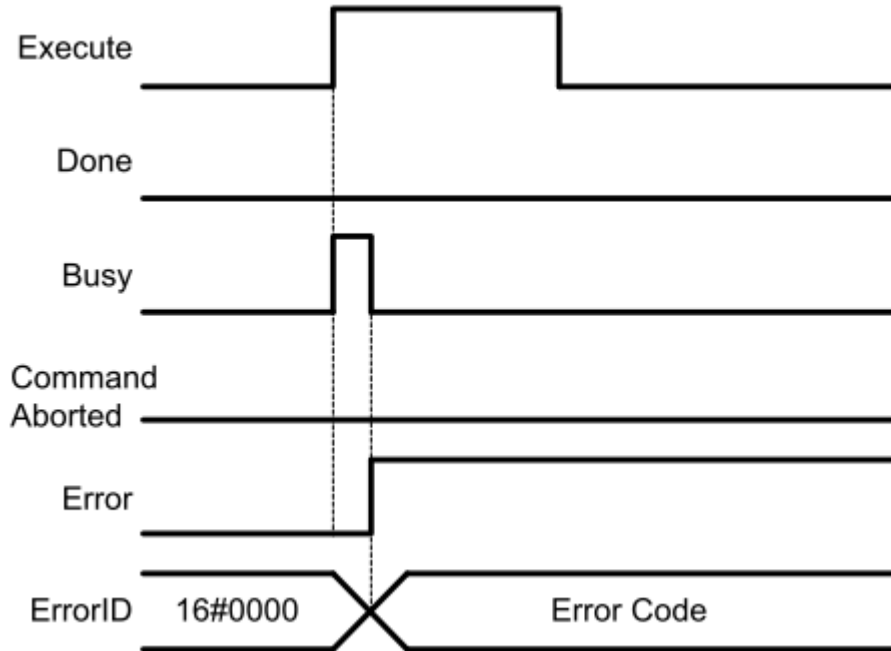
- When the power supply to the Controller is turned OFF, the settings of kinematics and workspace that you set in this instruction will be initialized. If use the robot function, it have to reset the kinematics and the workspace parameters by this instruction each time when the power is turned on.
- The setting value will be retained even after switching to program mode of the CPU Unit.
- Following robot instructions be able to use, after one control period of MC_SetKinTransform instruction is done(Done = TRUE).
 - MC_DefineCoordSystem, MC_GroupMon, MC_MoveTimeAbsolute,
 - MC_SyncLinearConveyor, MC_SyncOut, MC_InverseKin
- Even if kinematics parameters are set by this instruction, you can move each axes by using Axis instructions during the group axis is disabled.
- Even if Disable workspace check (EnableWorkspace = FALSE input variable) is set, the parameters for the workspace check function (parameters of the input variables defined in _sMC_WORKSPACE_REF) are set.
- If Kinematics Initialization Error (error code: 571 F hex) has occurred, the pre-registered Kinematics parameters are removed. For this reason, this error occurs, please reset the kinematics parameters.
- This instruction can be used when all of the following conditions are satisfied.
 - The number of axes registered in the axes group must be the same as the number of axes of the kinematics (robot).
 - Axis display units that are registered in the group axis must be degree.
 - That the number of axes that registered in kinematics(robot) are the same number of axes that are registered in the group axis.
 - For all axes registered in the axes group, the count mode must be set to linear mode.
 - For all axes registered in the axes group, the home must have been defined.
 - The axes group must be disabled.

• Timing Charts



- Errors

If an error occurs during instruction execution, *Error* will change to TRUE. You can find out the cause of the error by referring to the value output by *ErrorID*.



MC_DefineCoordSystem

The MC_DefineCoordSystem instruction sets a user coordinate system for a specified robot (axes group).

Instruction	Name	FB/FUN	Graphic expression	ST expression
MC_DefineCoordSystem	Define Coordinate System	FB	<p>MC_DefineCoordSystem_instance</p>	<pre>MC_DefineCoordSystem_instance(AxesGroup:=parameter, Execute:=parameter, CoordTransform:= parameter, Done =>parameter, Busy =>parameter, Error =>parameter, ErrorID =>parameter);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
Execute	Execute	BOOL	TRUE or FALSE	FALSE	The instruction is executed when <i>Execute</i> changes to TRUE.
CoordTransform	Coordinate System Transform	_sMC_COORD_REF	---	---	Specify the coordinate system to set.

sMC_COORD_REF

Member Variable	Meaning	Data type	Valid range	Function
CSID	Coordinate System ID	UINT	0 to 15	Specify the ID of user coordinate system to set.
Pose	Pose	ARRAY [0..5] OF LREAL	---	Specify the pose of user coordinate system based on MCS.

● Output Variables

Name	Meaning	Data type	Valid range	Description
Done	Done	BOOL	TRUE or FALSE	TRUE when the setting is completed.
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
CommandAborted	Command Aborted	BOOL	TRUE or FALSE	TRUE when the instruction is aborted.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#0000 indicates normal execution.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
Done	When the setting is successfully completed.	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one period when <i>Execute</i> is FALSE.
Busy	When <i>Execute</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Done</i> changes to TRUE. When <i>Error</i> changes to TRUE.
CommandAborted	<ul style="list-style-type: none"> When this instruction is canceled due to an error. When this instruction is executed while there is an error. 	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one period when <i>Execute</i> is FALSE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	When the error is cleared.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
AxesGroup	Axes Group	_sGROUP_REF	---	Specify the axes group.

■ **Function**

- This instruction is used to set a user coordinate system (*CS/D* = 0 to 15) for a specified robot (axes group).
- Up to 16 user coordinate systems can be set for each robot.
- When the user coordinate system is successfully registered, it can be referred to by *MC_GroupMon*, *MC_MoveTimeAbsolute*, and *MC_SyncLinearConveyor* instructions after that.
- When setting a user coordinate system, the following values are set for the variables defined by the *_sMC_COORD_REF* data type.

_sMC_COORD_REF

Name	Data type	Valid range	Description
Pose[0]	LREAL	Positive number, negative number, or 0	Position of home in the user coordinate system(UCS) that is seen from X axis home in machine coordinate system(MCS) Tx[mm]
Pose[1]	LREAL	Positive number, negative number, or 0	Position of home in the user coordinate system(UCS) that is seen from Y axis home in machine coordinate system(MCS) Ty[mm]
Pose[2]	LREAL	Positive number, negative number, or 0	Position of home in the user coordinate system(UCS) that is seen from Z axis home in machine coordinate system(MCS) Tz[mm]
Pose[3]	LREAL	-180 to 180	Rotation around the X axis of user coordinate system (UCS) Rx [deg]
Pose[4]	LREAL	-90 to 90	Rotation around the Y axis of user coordinate system (UCS) Ry [deg]
Pose[5]	LREAL	-180 to 180	Rotation around the Z axis of user coordinate system (UCS) Rz [deg]

- The setting value will be retained even after switching to program mode of the CPU unit.
- This instruction can be used when the following condition is satisfied.

The kinematics must have been set by the *MC_SetKinTransform* instruction.

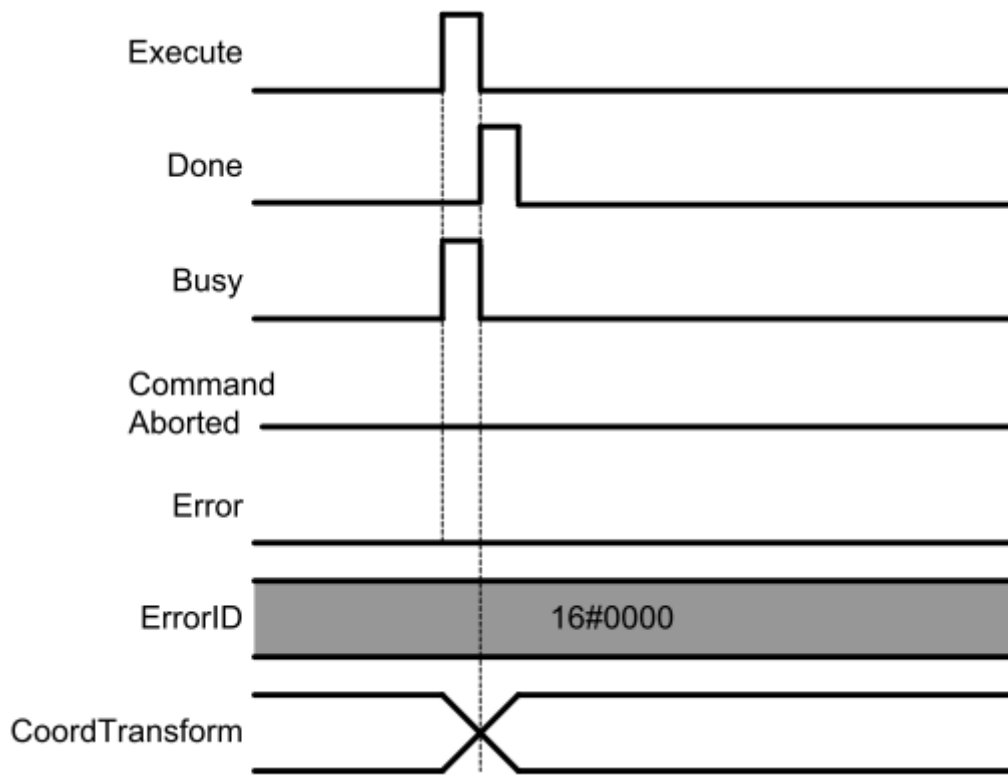


Precautions for Correct Use

- Each UCS is defined against a valid MCS.

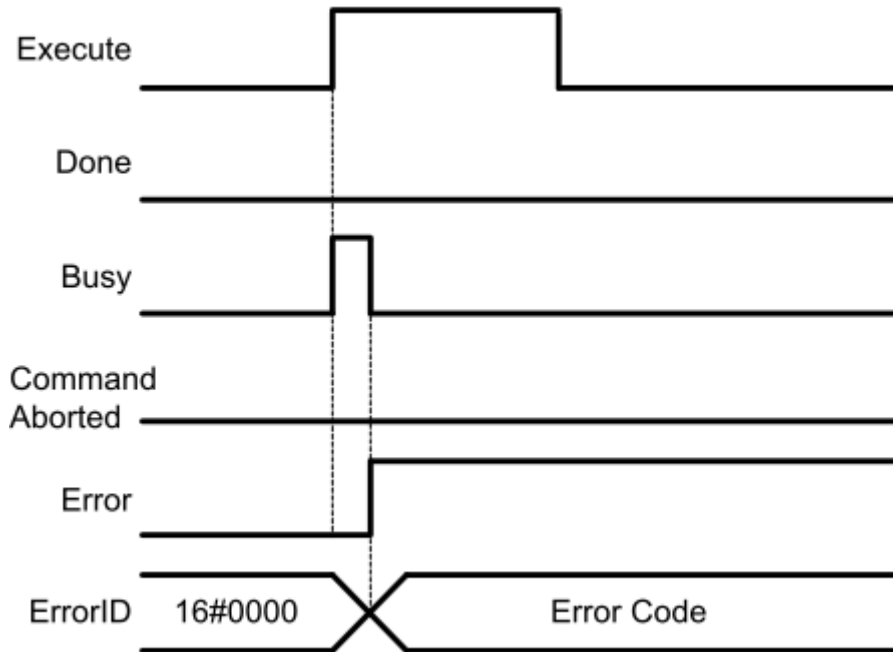
It means that each UCS of one robot must be defined again by executing *MC_DefineCoordSystem* for each UCS, if the *MC_SetKinTransform* was executed again on that robot.

• Timing Charts



- Errors

If an error occurs during instruction execution, *Error* will change to TRUE. You can find out the cause of the error by referring to the value output by *ErrorID*.



MC_GroupMon <Robot Ver 1.00>

The MC_GroupMon instruction reads the current position and current velocity of a robot.

Instruction	Name	FB/FUN	Graphic expression	ST expression
MC_GroupMon	Monitor Axes Group	FB	<p style="text-align: center;">MC_GroupMon instance</p>	<pre>MC_GroupMon (AxesGroup:=variable_name, Enable:=variable_name, CoordSystem:= variable_name, CSID:= variable_name, AxesGroup=>variable_name, Valid=> variable_name, Busy=> variable_name, Error=> variable_name, ErrorID=> variable_name Position=> variable_name, Velocity=> variable_name,);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
Enable	Enable	BOOL	TRUE or FALSE	FALSE	Reading the current values is started when <i>Enable</i> changes to TRUE.
CoordSystem	Coordinate System	_eMC_RBT_COORD_SYSTEM	1: _mcRBT_MCS 3: _mcRBT_UCS	1	Specify the coordinate system. 1: Machine coordinate system (MCS) 3: User coordinate system (UCS)
CSID	Coordinate System ID	UINT	0 to 15	0	Specify the ID of coordinate system when a user coordinate system is specified.

● Output Variables

Name	Meaning	Data type	Valid range	Description
Valid	Enabled	BOOL	TRUE or FALSE	When this value is TRUE, the following current values (<i>Position</i> and <i>Velocity</i>) are valid. The current position is updated every period.
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#0000 indicates normal execution.
Position	Current Position	_sMC_POSITION_REF	---	Shows command current position and actual current position.
Velocity	Current Velocity	_sMC_VELOCITY_REF	---	Shows actual current velocity of each axis and TCP.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

sMC_POSITION_REF

Member variable	Meaning	Data type	Valid range	Function
CommandPosition	Command Current Position	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	Shows the command current position of TCP on the specified coordinate system.
ActualPosition	Actual Current Position	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	Shows the actual current position of TCP on the specified coordinate system.

sMC_VELOCITY_REF

Member variable	Meaning	Data type	Valid range	Function
ActualVelocityTCP	Actual TCP Current Velocity	LREAL	Positive number or 0	Shows the actual current velocity of TCP. The unit is [mm/s].
ActualVelocity	Actual Current Velocity	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	Shows the actual current velocity of each axis when viewing the actual velocity of TCP on the specified coordinate system. The unit is [mm/s].

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
Valid	When reading the current values is completed successfully.	<ul style="list-style-type: none"> When <i>Error</i> changes to TRUE. When <i>Enable</i> changes to TRUE.
Busy	When <i>Enable</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Enable</i> changes to FALSE. When <i>Error</i> changes to TRUE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	When the error is cleared.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
AxesGroup	Axes Group	_sGROUP_REF	--	Specify the axes group.

■ Function

- This instruction is used to read the current position (command current position and actual current position) and the current velocity (actual TCP current velocity and actual current velocity) of a specified robot (axes group) on a specified coordinate system.
- Command/Actual Current position and Actual TCP Current Velocity/Actual Current Velocity on the specified coordinate system are calculated by using direct kinematics from the command current position and actual current position. In this reason, the output variables(Position/Velocity) of this instruction includes the calculation error. This calculation error is affected on the length of the robot link, encoder resolution, and reduction ratio.
- Command current position will be updated every control period when the group is TRUE and output variable is TRUE. The command current position value is the latest value when the axis group was Enabled and output variable Valid was TRUE.
- Actual current position, Actual TCP Current Velocity and Actual Current Velocity are updated every control period, even group is Enabled or Disabled when the output variable Valid is TRUE.
- For Delta3, the following values are read out to the variables defined by the _sMC_POSITION_REF and _sMC_VELOCITY_REF data types.

_sMC_POSITION_REF

CommandPosition

Name	Data type	Valid range	Description
CommandPosition[0]	LREAL	Negative number, positive number, or 0	Command current position of X axis [mm]
CommandPosition[1]	LREAL	Negative number, positive number, or 0	Command current position of Y axis [mm]
CommandPosition[2]	LREAL	Negative number, positive number, or 0	Command current position of Z axis [mm]
CommandPosition[3]-[5]	LREAL	0	Reserved

ActualPosition

Name	Data type	Valid range	Description
ActualPosition[0]	LREAL	Negative number, positive number, or 0	Actual current position of X axis [mm]
ActualPosition[1]	LREAL	Negative number, positive number, or 0	Actual current position of Y axis [mm]
ActualPosition[2]	LREAL	Negative number, positive number, or 0	Actual current position of Z axis [mm]
ActualPosition[3]-[5]	LREAL	0	Reserved

_sMC_VELOCITY_REF

ActualVelocityTCP

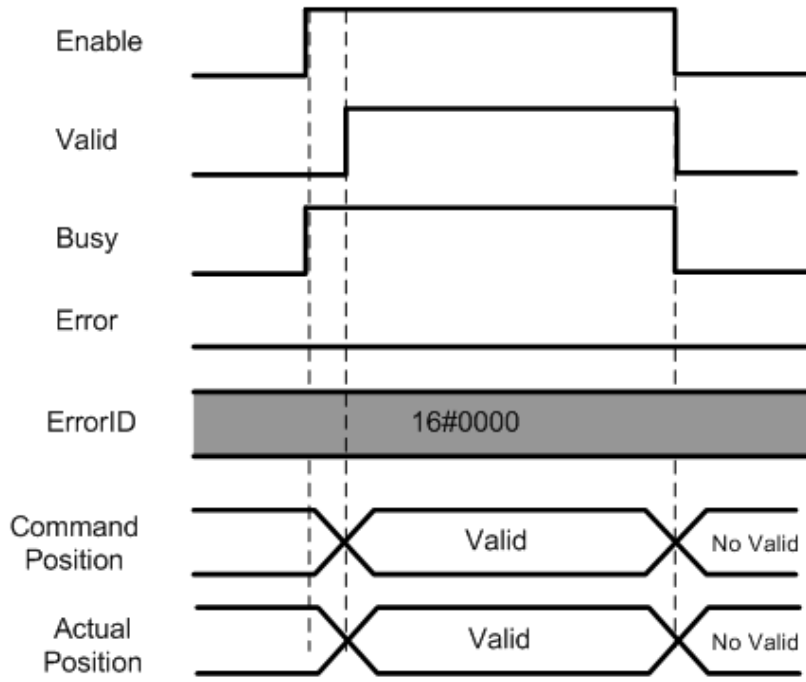
Name	Data type	Valid range	Description
ActualVelocityTCP	LREAL	Positive number or 0	Actual current velocity of the TCP. TCP velocity = $\text{SQRT}(V_x^2 + V_y^2 + V_z^2)$ [mm/s]

ActualVelocity

Name	Data type	Valid range	Description
AcutalVelocity[0]	LREAL	Negative number, positive number, or 0	Actual current velocity of X axis (Vx) [mm/s]
AcutalVelocity[1]	LREAL	Negative number, positive number, or 0	Actual current velocity of Y axis (Vy) [mm/s]
AcutalVelocity[2]	LREAL	Negative number, positive number, or 0	Actual current velocity of Z axis (Vz) [mm/s]
AcutalVelocity[3]-[5]	LREAL	0	Reserved

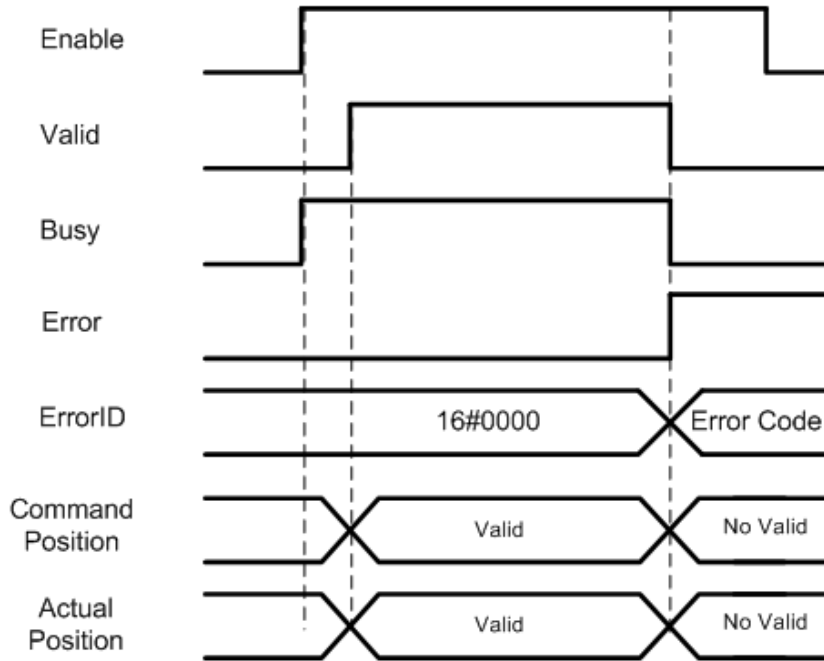
- When the Valid output has changed from TRUE to FALSE, the value of the Position and Velocity Output variables are retained.
- This instruction can be used when all of the following conditions are satisfied.
 - The kinematics must have been set by the MC_SetKinTransform instruction.
 - For all axes registered in the axes group, the home must have been defined.
 - When a user coordinate system is used, the coordinate system ID, which is specified by the MC_DefineCoordSystem instruction, must be set.

• Timing Charts



● Errors

If an error occurs during instruction execution, *Error* will change to TRUE. You can find out the cause of the error by referring to the value output by *ErrorID*.



MC_MoveTimeAbsolute <Robot Ver 1.00>

The MC_MoveTimeAbsolute instruction creates command values to reach a specified target position in a specified time.

Instruction	Name	FB/FUN	Graphic expression	ST expression
MC_MoveTimeAbsolute	Time-specified Absolute Positioning	FB	<p>MC_MoveTimeAbsolute_instance</p>	<pre>MC_MoveTimeAbsolute (AxesGroup :=variable_name, Execute :=variable_name, Position :=variable_name, TrajData :=variable_name, CoordSystem :=variable_name, CSID :=variable_name, BufferMode :=variable_name, TransitionMode :=variable_name, TransitionParameter :=variable_name, AxesGroup =>variable_name, Done =>variable_name, Busy => variable_name, Active =>variable_name, CommandAborted =>variable_name, Error =>variable_name, ErrorID =>variable_name,);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
Execute	Execute	BOOL	TRUE or FALSE	FALSE	The instruction is executed when <i>Execute</i> changes to TRUE.
Position	Target Position	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	0	Specify the target position. For Delta3, the unit is [mm].
TrajData	Trajectory Data	_sMC_MOVE_TRAJ_REF	---	---	Specify the parameters for creating command current positions up to the target position.
CoordSystem	Coordinate System	_eMC_RBT_COORD_SYSTEM	1: _mcRBT_MCS 3: _mcRBT_UCS	1	Specify the coordinate system of the target position. 1: Machine coordinate system (MCS) 3: User coordinate system (UCS)
CSID	Coordinate System ID	UINT	0 to 15	0	Specify the ID of coordinate system when a user coordinate system is specified.
BufferMode	Buffer Mode Selection	_eMC_BUFFER_MODE	0: _mcAborting	0	Specify the behavior when executing more than one motion control instruction. 0: Aborting
TransitionMode	Transition Mode	_eMC_TRANSITION_MODE	0: _mcTMNone	0	Specify the path of motion. 0: Transition disabled
TransitionParameter	Transition Parameters	ARRAY [0..7] OF LREAL	Negative number, positive number, or 0	0	Set the parameters for transition. (Reserved)

sMC_MOVE_TRAJ_REF

Member variable	Meaning	Data type	Valid range	Function
MoveTrajType	Movement Trajectory Type Selection	_eMC_MOVE_TRAJ_TYPE	0:_mcPolynomial3	Specify the trajectory type to create command current position of each axis. Only the third-order polynomial is supported in Robot Ver 1.00. 0: Third-order polynomial
TrajTime	Trajectory Target Time	UINT	Positive number	Specify the time to create command current positions from current position to target position. The unit is [ms].
MaxVelocity	Velocity Error Detection Value	LREAL	Positive number	Specify the velocity at which a velocity error (too fast) is detected when creating a command current position. The unit is [mm/s].
MaxAcceleration	Acceleration Error Detection Value	LREAL	Positive number	Specify the acceleration rate at which an acceleration error (too much acceleration/deceleration) is detected when creating a command current position. The unit is [mm/s ²].

● Output Variables

Name	Meaning	Data type	Valid range	Description
Done	Done	BOOL	TRUE or FALSE	TRUE when the instruction is completed.
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
Active	Controlling	BOOL	TRUE or FALSE	TRUE when the axes group is being controlled.
CommandAborted	Command Aborted	BOOL	TRUE or FALSE	TRUE when the instruction is aborted.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#0000 indicates normal execution.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
Done	When the specified target time is reached.	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one control period when <i>Execute</i> is FALSE.
Busy	When <i>Execute</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Done</i> changes to TRUE. When <i>Error</i> changes to TRUE. When <i>CommandAborted</i> changes to TRUE.
Active	When the axis starts moving	<ul style="list-style-type: none"> When <i>Done</i> changes to TRUE. When <i>Error</i> changes to TRUE. When <i>CommandAborted</i> changes to TRUE.
CommandAborted	<ul style="list-style-type: none"> When this instruction is aborted because another motion control instruction was executed with the Buffer Mode set to <i>Aborting</i>. When this instruction is canceled due to an error. When this instruction is executed while there is an error. When this instruction is started during MC_GroupStop instruction execution. 	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one control period when <i>Execute</i> is FALSE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	When the error is cleared.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
AxesGroup	Axes Group	_sGROUP_REF	---	Specify the axes group.

■ Function

- This instruction is used to move a robot (axes group) from a current to the target position, which is specified by the Position input variable.
- This instruction is completed when the specified target time is reached. No in-position check is performed.
- The details are described below.

Precautions for Safe Use

- After the adjustment of mechanical home of the robot, please move the robot.
- Do not use robot at out of the workspace, without workspace check function. If you use may cause damage to the robot.

● Specifying the Target Position

- The target position (*Position*) is the position on a machine coordinate system (MCS) or user coordinate system (UCS). For Delta3, the target position is set as shown below.

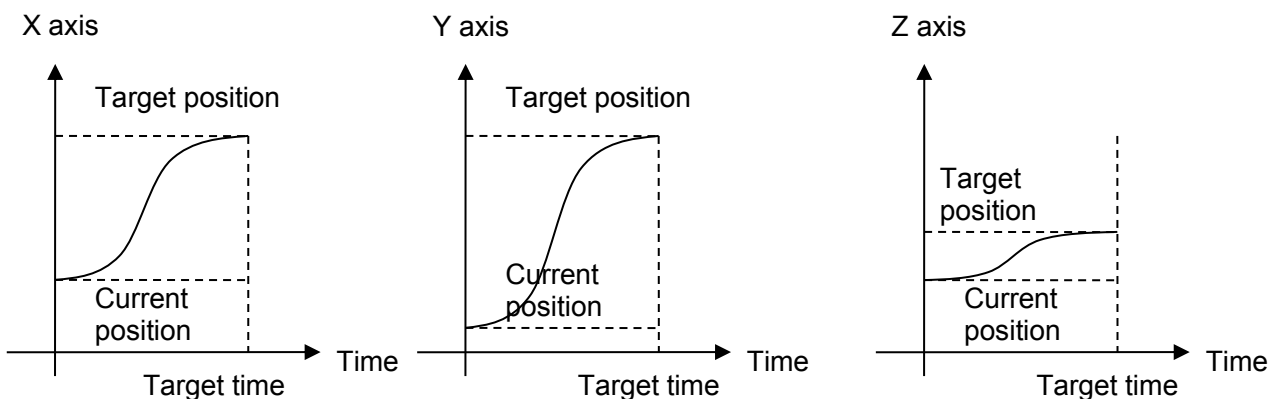
Position

Name	Data type	Valid range	Description
Position[0]	LREAL	Negative number, positive number, or 0	Target position of X axis [mm]
Position[1]	LREAL	Negative number, positive number, or 0	Target position of Y axis [mm]
Position[2]	LREAL	Negative number, positive number, or 0	Target position of Z axis [mm]
Position[3]-[5]	LREAL	0	Reserved

- When using an user coordinate system (UCS), specify its ID number in the Coordinate System ID (*CSID*).
- If the workspace check function is enabled and it detects an error, the axis Group is stopped. Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) section エラー! 参照元が見つかりません。 the details of workspace check function.

● Specifying the Trajectory

- The parameters to create the trajectory up to the target position are specified by *MoveTrajType*, *TrajTime*, *MaxVelocity*, and *MaxAcceleration* of the Trajectory Data (*TrajData*).
- Select the third-order polynomial in Movement Trajectory Type Selection (*MoveTrajType*).



- If any calculated commanded position is outside of the workspace, when the workspace check function is enabled, an error will be raised and the axes stopped.
- When calculating a commanded position, velocity and acceleration of the movement are calculated and checked against the command maximums. If exceeding them, the following errors will be raised: Velocity Error Detection Value (MaxVelocity) and Acceleration Error Detection Value (MaxAcceleration).

● Other Specifications

- Only *Aborting* is supported for *BufferMode*.
- Only *Transition disabled* is supported for *TransitionMode*.
- The *TransitionParameter* is not used. (Reserved)
- This instruction can be used when all of the following conditions are satisfied.

The kinematics must have been set by the MC_SetKinTransform instruction.

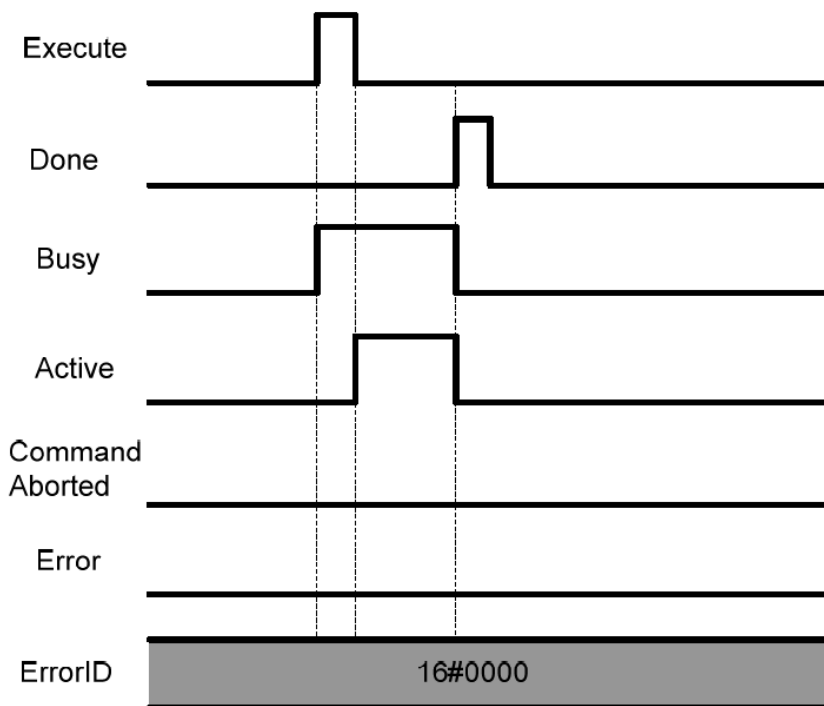
For all axes registered in the axes group, the home must have been defined.

For all axes registered in the axes group, must be turned ON.

For all axes registered in the axes group, limit inputs must be OFF.

The axes group must be enabled.

● Timing Charts



● Re-execution of Motion Control Instructions

- This instruction cannot be re-executed.

A Motion Control Instruction Re-execution Disabled error (error code: 543B hex) occurs if re-execution is attempted, and all axes that are being moved by this instruction stop.

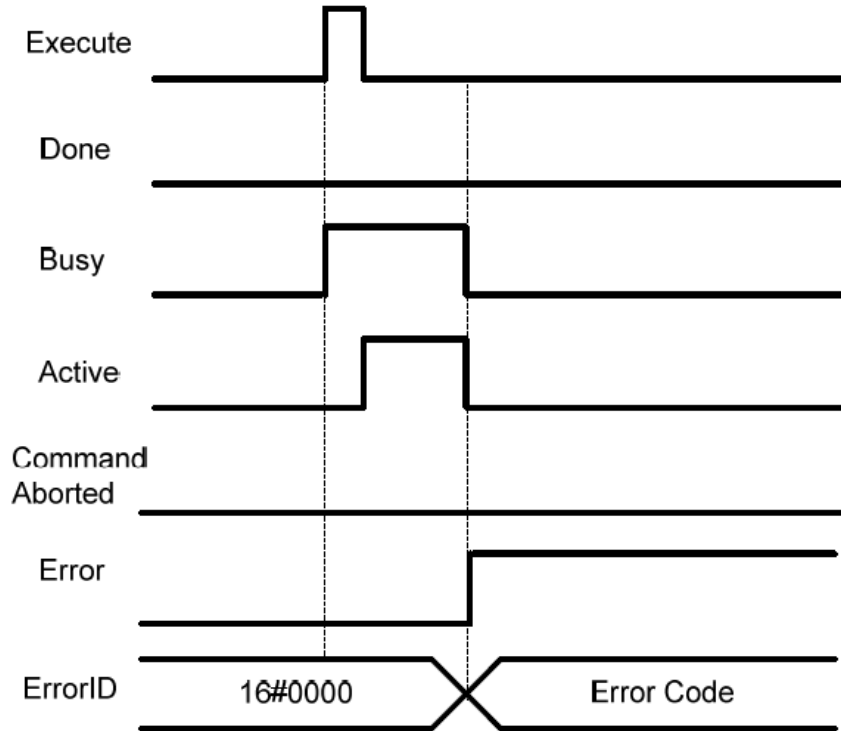
● Multi-execution of Motion Control Instructions

- A restriction applies to the instructions that can be used while this instruction is in execution.
- For details on multi-execution of motion control instructions, refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Multi-execution of Motion Control Instructions* of this document.

● Errors

If an error occurs during instruction execution, *Error* will change to TRUE.

You can find out the cause of the error by referring to the value output by *ErrorID* (Error Code).



MC_SyncLinearConveyor <Robot Ver 1.00>

The MC_SyncLinearConveyor instruction starts creation of command values to catch up with a specified workpiece on a conveyor.

Instruction	Name	FB/ FUN	Graphic expression	ST expression
MC_SyncLinearConveyor	Start Conveyor Synchronization	FB	<p>MC_SyncLinearConveyor _instance</p>	<pre>MC_SyncLinearConveyor (AxesGroup:=variable_name, ConveyorAxis:=variable_name, Execute :=variable_name, InitWorkpiecePosition :=variable_name, TrajData :=variable_name, CoordSystem:=variable_name, CSID :=variable_name, BufferMode :=variable_name, TransitionMode :=variable_name, AxesGroup =>variable_name, ConveyorAxis :=variable_name, InSync =>variable_name, Busy => variable_name, Active =>variable_name, CommandAborted =>variable_name, Error =>variable_name, ErrorID =>variable_name, TrajStatus => variable_name,);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
Execute	Execute	BOOL	TRUE or FALSE	FALSE	The instruction is executed when <i>Execute</i> changes to TRUE.
InitWorkpiecePosition	Initial Workpiece Position	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	0	Specify the workpiece position (target position) at the time when the synchronization with conveyor is started. The unit is [mm] for InitWorkpiecePosition[0]-[2] and [deg] for InitWorkpiecePosition[3]-[5].
TrajData	Trajectory Data	_sMC_SYNC_TRAJ_REF	---	---	Specify the parameters for creating command current positions up to the target position.
CoordSystem	Coordinate System	_eMC_RBT_COORD_SYSTEM	1: _mcRBT_MCS 3: _mcRBT_UCS	1	Specify the coordinate system of the target position. 1: Machine coordinate system (MCS) 3: User coordinate system (UCS)
CSID	Coordinate System ID	UINT	0 to 15	0	Specify the ID of coordinate system when a user coordinate system is specified.
BufferMode	Buffer Mode Selection	_eMC_BUFFER_MODE	0: _mcAborting	0	Specify the behavior when executing more than one motion control instruction. 0: Aborting
TransitionMode	Transition Mode	_eMC_TRANSITION_MODE	0: _mcTMNone	0	Specify the path of motion. 0: Transition disabled
TransitionParameter	Transition Parameters	ARRAY [0..7] OF LREAL	Negative number, positive number, or 0	0	Set the parameters for transition. (Reserved)

sMC_SYNC_TRAJ_REF

Member variable	Meaning	Data type	Valid range	Function
SyncTrajType	Synchronization Trajectory Type Selection	_eMC_SYNC_TRAJ_TYPE	0:_mcFlatTraj	Specify the trajectory type to create command current position for synchronization with the conveyor. Only the third-order polynomial is currently supported. 0: Third-order polynomial
TrajTime	Trajectory Target Time	ARRAY [0..7] OF UINT	---	Specify the time to create command current positions from current position to target position. The unit is [ms].
TrajTransition	Trajectory Transition Ratio	ARRAY [0..7] OF UINT	---	Specify the time ratio to start the movement from current operation to next operation within this command. The unit is [%].
TrajDistance	Trajectory Distance	ARRAY [0..7] OF LREAL	Negative number, positive number, or 0	Specify the distance for upward/downward movement from current position to target position. The unit is [mm].
MaxVelocity	Velocity Error Detection Value	LREAL	Positive number	Specify the velocity at which a velocity error (too fast) is detected when creating a command current position. The unit is [mm/s].
MaxAcceleration	Acceleration Error Detection Value	LREAL	Positive number	Specify the acceleration rate at which an acceleration error (too much acceleration/deceleration) is detected when creating a command current position. The unit is [mm/s ²].

● Output Variables

Name	Meaning	Data type	Valid range	Description
InSync	In Sync	BOOL	TRUE or FALSE	TRUE when the TCP of robot catches up with the workpiece on the conveyor.
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
Active	Controlling	BOOL	TRUE or FALSE	TRUE when the axes group is being controlled.
CommandAborted	Command Aborted	BOOL	TRUE or FALSE	TRUE when the instruction is aborted.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#0000 indicates normal execution.
TrajStatus	Trajectory Status	_sMC_SYNC_TRAJ_ST ATUS	---	Outputs the status (remaining time for creating command current position and phase) of the trajectory.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

sMC_SYNC_TRAJ_STATUS

Member variable	Meaning	Data type	Valid range	Function
RemainTime	Remaining Time	UINT	Positive number or 0	Outputs the remaining time for creating command current position. The unit is [ms].
Phase	Phase	UINT	0 to 6	Outputs which position on trajectory the command position refers to.

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
InSync	If the command current position and actual current position of the workpiece are matched, after the Target Time Trajectory ($T1 + T2 + (T3 - T3 * Trans2)$) (later describe) has elapsed. If the difference between command current position(X, Y) and actual current position of the workpiece (X,Y) of each axis is as below, it is considered as match. ·When control period is 1ms : 0.1[mm] ·When control period is 2ms : 0.2[mm] ·When control period is 4ms : 0.4[mm]	<ul style="list-style-type: none"> When <i>Error</i> changes to TRUE. When <i>CommandAborted</i> changes to TRUE.
Busy	When <i>Execute</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Error</i> changes to TRUE. When <i>CommandAborted</i> changes to TRUE.
Active	When the axis starts moving.	<ul style="list-style-type: none"> When <i>Error</i> changes to TRUE. When <i>CommandAborted</i> changes to TRUE.
CommandAborted	<ul style="list-style-type: none"> When this instruction is aborted because another motion control instruction was executed with the Buffer Mode set to <i>Aborting</i>. When this instruction is canceled due to an error. When this instruction is executed while there is an error. When this instruction is started during MC_GroupStop instruction execution. 	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one period when <i>Execute</i> is FALSE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	When the error is cleared.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
AxesGroup	Axes Group	sGROUP_REF	---	Specify the axes group.
ConveyorAxis	Conveyor Axis	sAxis_REF	---	Specify the axis.

■ Function

- This instruction is used for a specified robot (axes group) to catch up with a workpiece specified by Initial Workpiece Position (*InitWorkpiecePosition*) from the current position.
- The *InSync* output variable changes to TRUE when the robot catches up with the specified workpiece (i.e. when X axis and Y axis match). Then, the synchronization is maintained.
- Normally, the MC_SyncOut (End Synchronization) instruction is used to release the synchronization.
- If processing of the primary periodic task has not been finished within one control period, there is possibility that Velocity Error Detected or Acceleration Error Detected occurs. Please do settings and programming so that the processing of the primary periodic task does not exceed one control period.
- At the moment of starting this instruction, MaxVelocity and MaxAcceleration are checked with the trajectory that the workpiece is stopping. Therefore, it could happen that the velocity error or the acceleration error is detected after Phase 2, due to velocity of the conveyor.
- The details are described below.



Precautions for Safe Use

- After the adjustment of mechanical home of the robot, please move the robot.
- Do not use robot at out of the workspace, without workspace check function. If you use may cause damage to the robot.
- If you want to stop the movement of the running in the instruction MC_SyncLinearConveyor, please use the instruction MC_SyncOut.
- Due to the way the maximum interpolation velocity of this command is calculated, it could happen that in the Phase 2 or in Phase 4 the velocity error (too fast) or the acceleration error

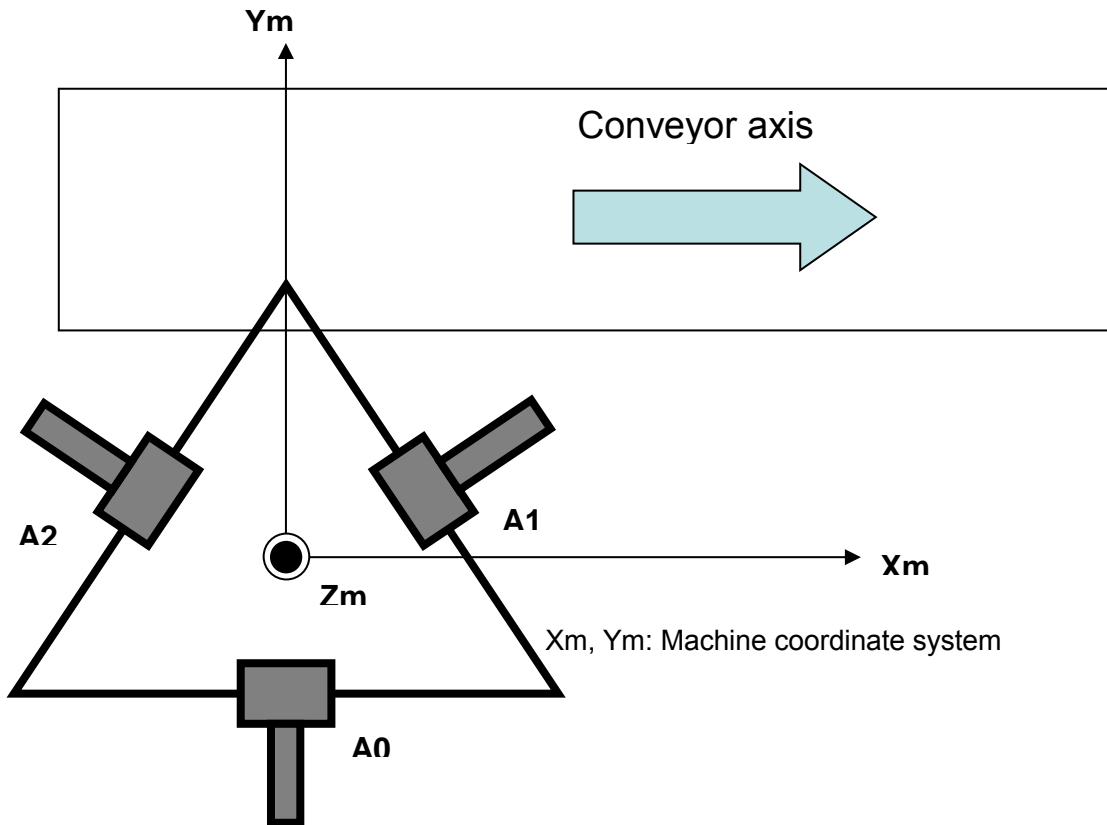
(too much acceleration/deceleration) is detected, even if those error was not detected at the moment of starting this instruction (pre-check). This is due to the superposition of the Z profile with the XY profiles.

- Setting Coordinate System and Conveyor Axis

- When catching up with a workpiece specified by Initial Workpiece Position (*InitWrokpiecePosition*), the X axis of coordinate system, which is specified by the Coordinate System (*CoordSystem*), needs to be the same direction as the conveyor axis.
- The procedure to specify the machine coordinate system (MCS) and user coordinate system (UCS) is shown below.

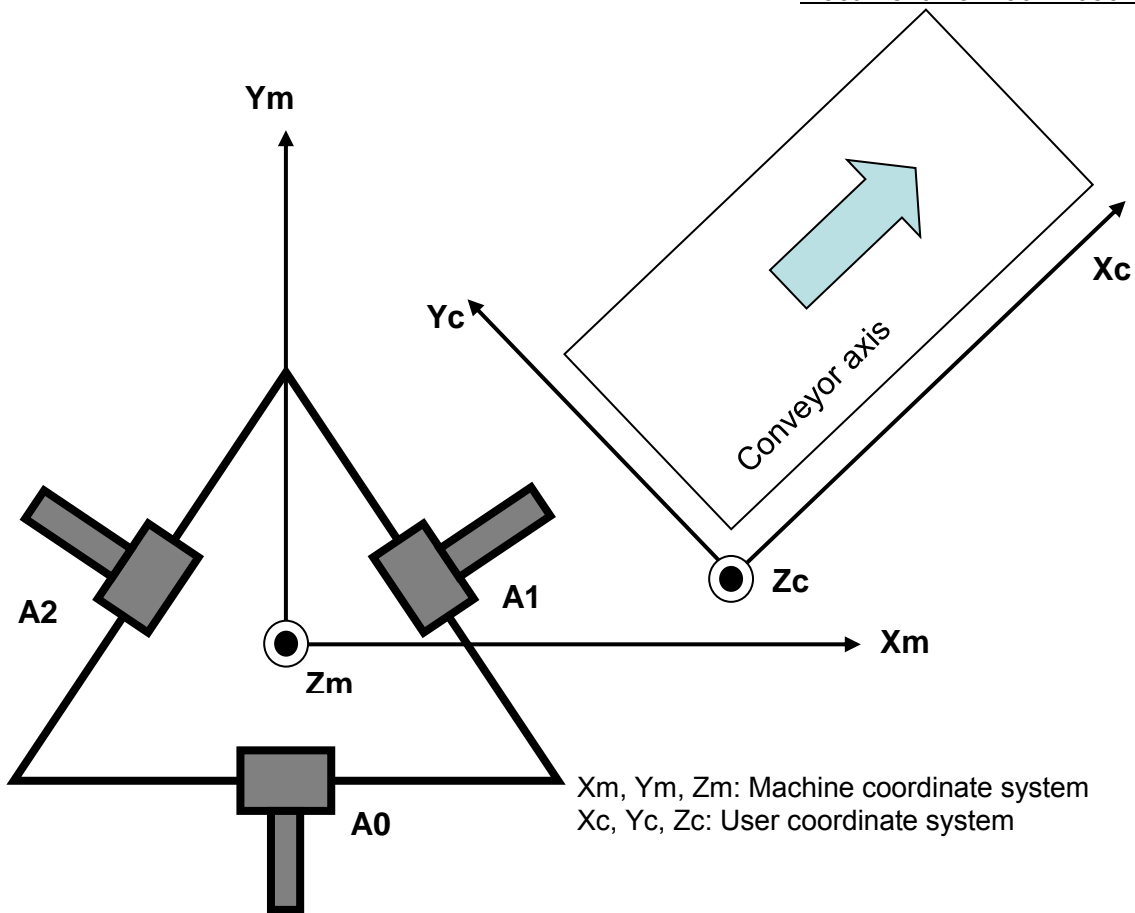
[When specifying the machine coordinate system (MCS)]

Align the directions of X_m and conveyor axis as shown in the following figure.



[When specifying the user coordinate system (UCS)]

Align the directions of X_c and conveyor axis as shown in the following figure.



• Specifying the Position of Workpiece to Catch up with

- The Initial Workpiece Position (*InitWorkpiecePosition*) of the workpiece to catch up with is a position on the machine coordinate system (MCS) or user coordinate system (UCS). For Delta3, the position is specified as shown below.

InitWorkpiecePosition

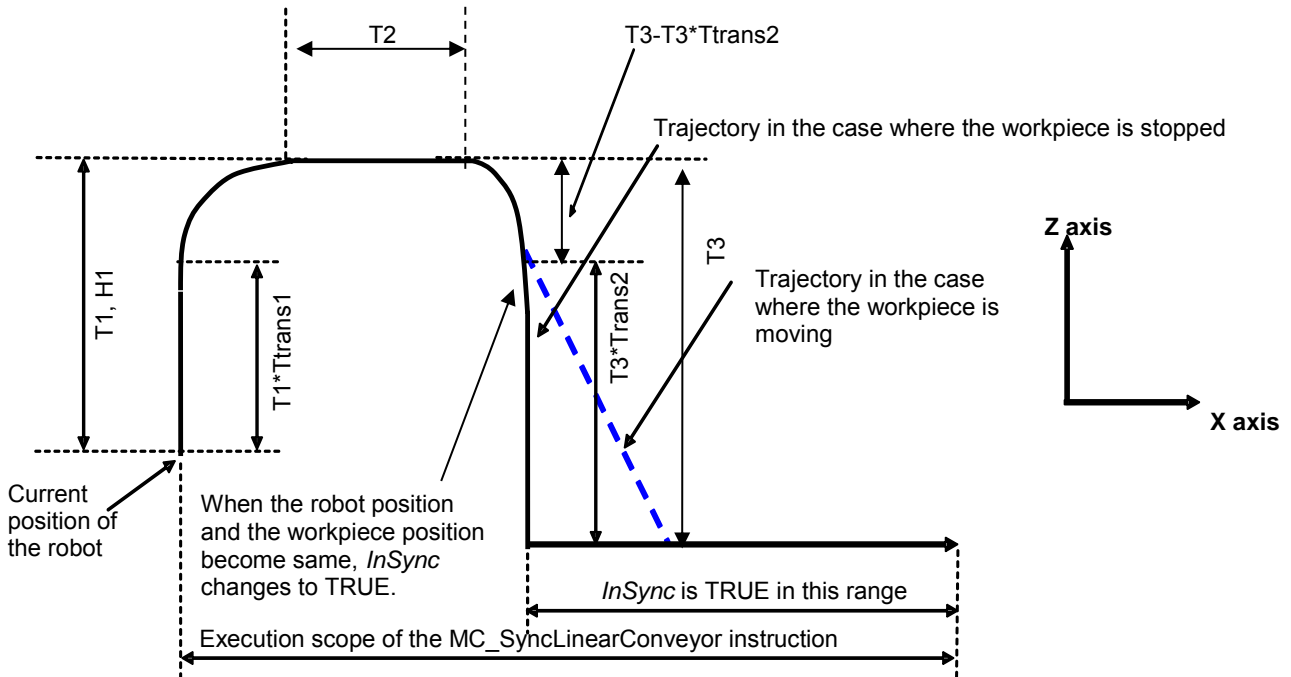
Name	Data type	Valid range	Description
InitWorkpiecePosition[0]	LREAL	Negative position, positive position, or 0	Current position of X axis of the workpiece [mm]
InitWorkpiecePosition[1]	LREAL	Negative position, positive position, or 0	Current position of Y axis of the workpiece [mm]
InitWorkpiecePosition[2]	LREAL	Negative position, positive position, or 0	Current position of Z axis of the workpiece [mm]
InitWorkpiecePosition[3]-[5]	LREAL	0	Reserved

- When using an user coordinate system (UCS), specify its ID number in the Coordinate System ID (*CSID*).
- When this instruction is successfully started, the above workpiece position is updated along with the conveyor axis movement and command position is created toward the updated workpiece position.
- If the initial workpiece position is outside the workspace when the workspace function is enabled, it is considered as an error and the axes are stopped.
- When the current position is outside the workspace, the axes are also stopped.

● Specifying the Trajectory

- The parameters to create the trajectory to catch up with the workpiece are specified by *SyncTrajType*, *TrajTime*, *TrajTransition*, *TrajDistance*, *MaxVelocity*, and *MaxAcceleration* of the Trajectory Data (*TrajData*).
- Only *Third-order polynomial* is supported for Synchronization Trajectory Type Selection (*SyncTrajType*) in Robot Ver 1.00.
- The following figure shows the relationship between specified values and trajectory in the flat type.

[Case 1: $H1 > 0$ and $T1 > 0$]



<Terms> T1,T2,T3: Trajectory Target Time [ms]

Ttrans1, Ttrans2: Trajectory Transition Ratio [%]

H1: Trajectory Distance Z-axis direction in Machine coordinate system (MCS) [mm]

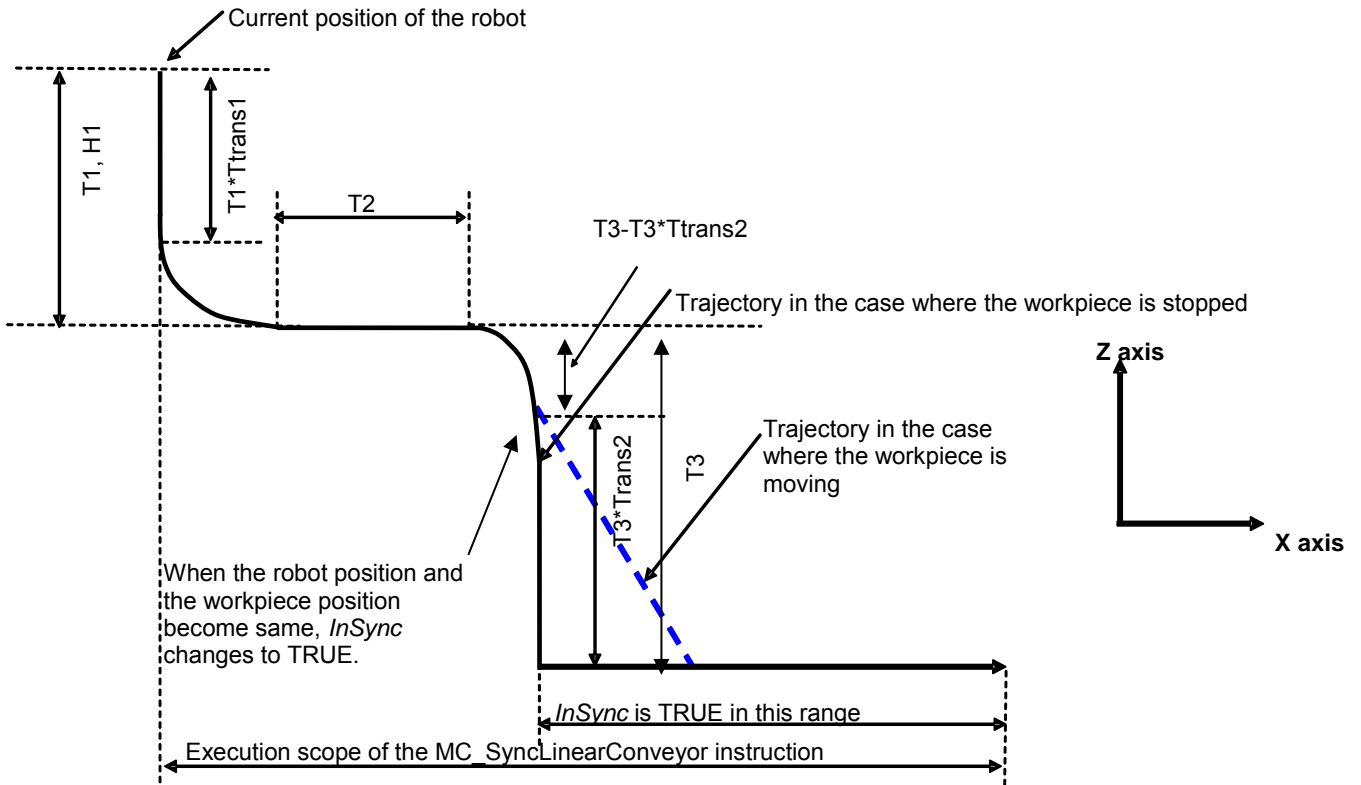
- The command position to move the robot from the current position to “current position + $H1$ ” (Z axis) in $T1$ is created.
- The command position to follow the workpiece after $T1 * Ttrans1$ has elapsed is created. The robot does not stop at this time.
- The command position for moving the robot downward in the Z axis direction after $(T1 + T2)$ has elapsed is created. The robot does not stop at this time.
- After $(T1 + T2 + (T3 - T3 * Ttrans2))$ has elapsed, the difference between command position (X,Y) and actual current position of the workpiece (X,Y) is compared. When each difference is the following range or less, InSync changes to TRUE. If it is exceeded, it is considered as an error and the axes are stopped. At this time, Synchronization Disabled error (error code 6702hex) occurs.

When control period is 1ms : 0.1[mm]

When control period is 2ms : 0.2[mm]

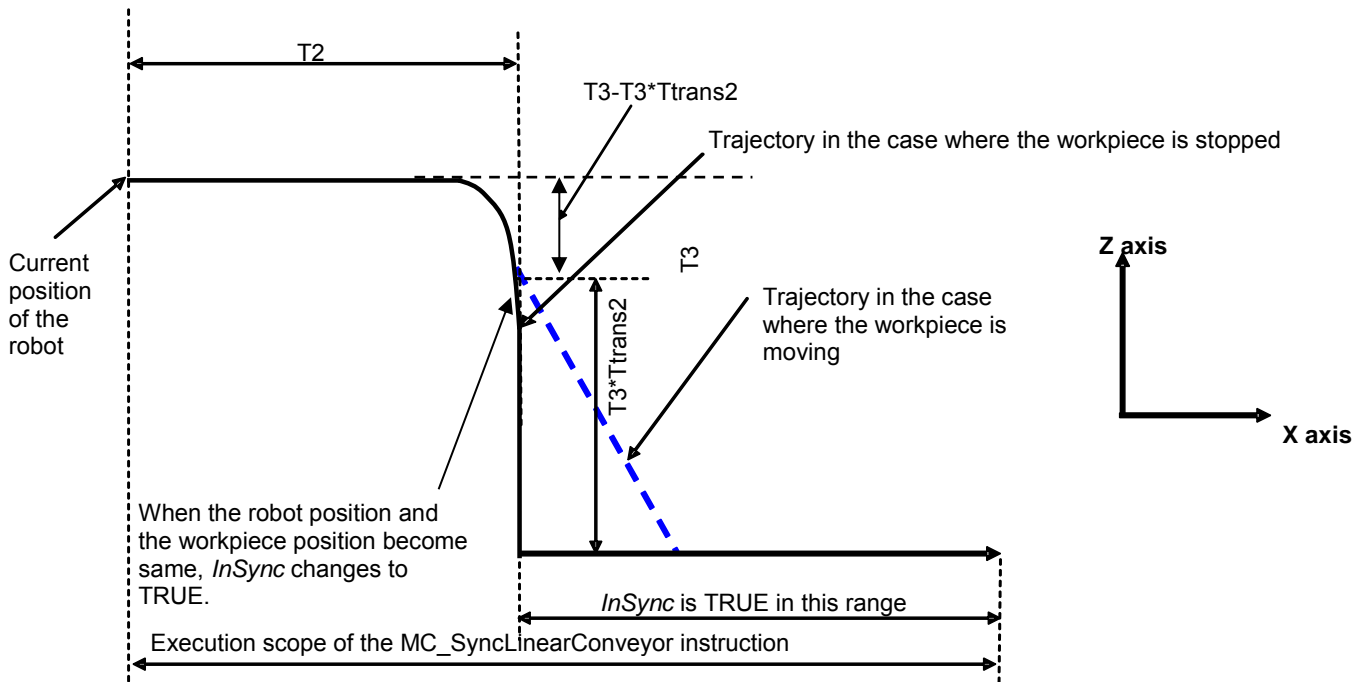
When control period is 4ms : 0.4[mm]

[Case 2: $H1 < 0$ and $T1 > 0$]



- When $H1 < 0$, the command position for moving the robot downward in the Z axis direction is created.
- In other cases, same as Case 1.

[Case 3: $T1 = 0$]



- When $T1 = 0$, initial command position for moving the robot upward/downward in the Z axis direction is not created.
- In other cases, same as Case 1.

- The following values are set for Trajectory Target Time (*TrajTime*), Trajectory Transition Ratio (*TrajTransition*), and Trajectory Distance (*TrajDistance*).

TrajTime

Name	Data type	Valid range	Description
TrajTime[0]	UINT	Positive number or 0	T1 [ms]
TrajTime[1]	UINT	Positive number	T2 [ms]
TrajTime[2]	UINT	Positive number	T3 [ms]
TrajTime[3]-[7]	UINT	0	Reserved

TrajTransition

Name	Data type	Valid range	Description
TrajTransition[0]	UINT	50 to 100	Ttrans1 [%]
TrajTransition[1]	UINT	50 to 100	Ttrans2 [%]
TrajTransition[2]-[7]	UINT	0	Reserved

TrajDistance

Name	Data type	Valid range	Description
TrajDistance[0]	LREAL	Negative number, positive number, or 0	Trajectory Distance Z axis direction in Machine coordinate system (MCS) H1 [mm]
TrajDistance[1]-[7]	LREAL	0	Reserved

- If the workspace check function is enabled and it detects an error, the axis Group is stopped. Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) section エラー! 参照元が見つかりません。 the details of workspace check function.
- When creating the command position, the command velocity and command acceleration are also created that considered velocity of conveyor in every control period. And it is checked whether or not the Velocity Error Detection Value (MaxVelocity) and Acceleration Error Detection Value (MaxAcceleration) are exceeded .

- Other Specifications

- Only *Aborting* is supported for *BufferMode*.
- Only *Transition disabled* is supported for *TransitionMode*.
- The *TransitionParameter* is not used. (Reserved)
- This instruction can be used when all of the following conditions are satisfied.

The kinematics must have been set by the MC_SetKinTransform instruction.

For all axes registered in the axes group, the home must have been defined.

For all axes registered in the axes group, must be turned ON.

For all axes registered in the axes group, limit inputs must be OFF.

The axes group must be enabled.

● Outputting Status during Command Position Creation

- While the command position is created, each status is output to the applicable member of Trajectory Status (*_sMC_SYNC_TRAJ_STATUS*). During phase 5 and 6, robot and workpiece Synchronize and Insync will change to TRUE.

RemainTime: Shows the remaining time of Trajectory Target Time ($T1 + T2 + T3$). This is found by subtraction from $T1 + T2 + T3$.

Phase: Phases 0 to 6.

Shows which trajectory is being calculated.

0: This instruction is not being executed.

1: $t \leq T_{trans1} * T1$

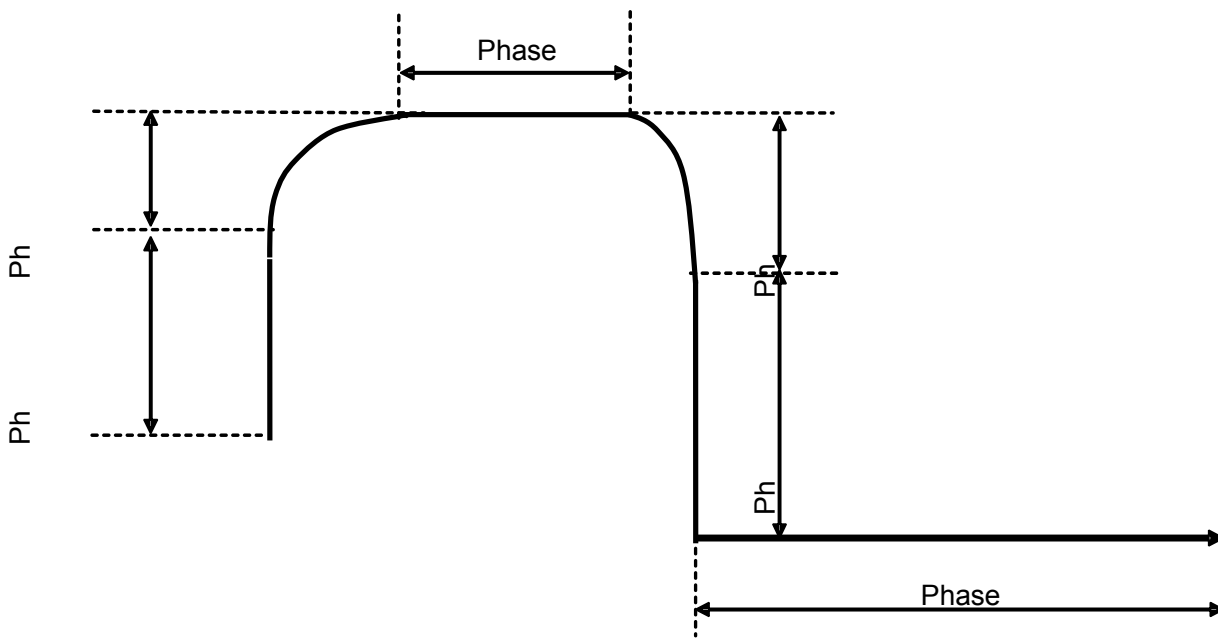
2: $T_{trans1} * T1 < t \leq T1$

3: $T1 < t \leq T1 + T2$

4: $T1 + T2 < t \leq T1 + T2 + (T3 - T3 * T_{trans2})$

5: $T1 + T2 + (T3 - T3 * T_{trans2}) < t \leq (T1 + T2 + T3)$

6: $(T1 + T2 + T3) < t$



When setting Trajectory Transition Ratio parameters (T_{trans1} and T_{trans2}) to 100%, control period of phase 2 and 4 is not to be 0. At least, One control period is secured as below.

Phase

0: This instruction is not being executed.

1: $t \leq T1$

2: $T1 < t \leq T1 + \text{One control period}$

3: $T1 + \text{One control period} < t \leq T1 + T2$

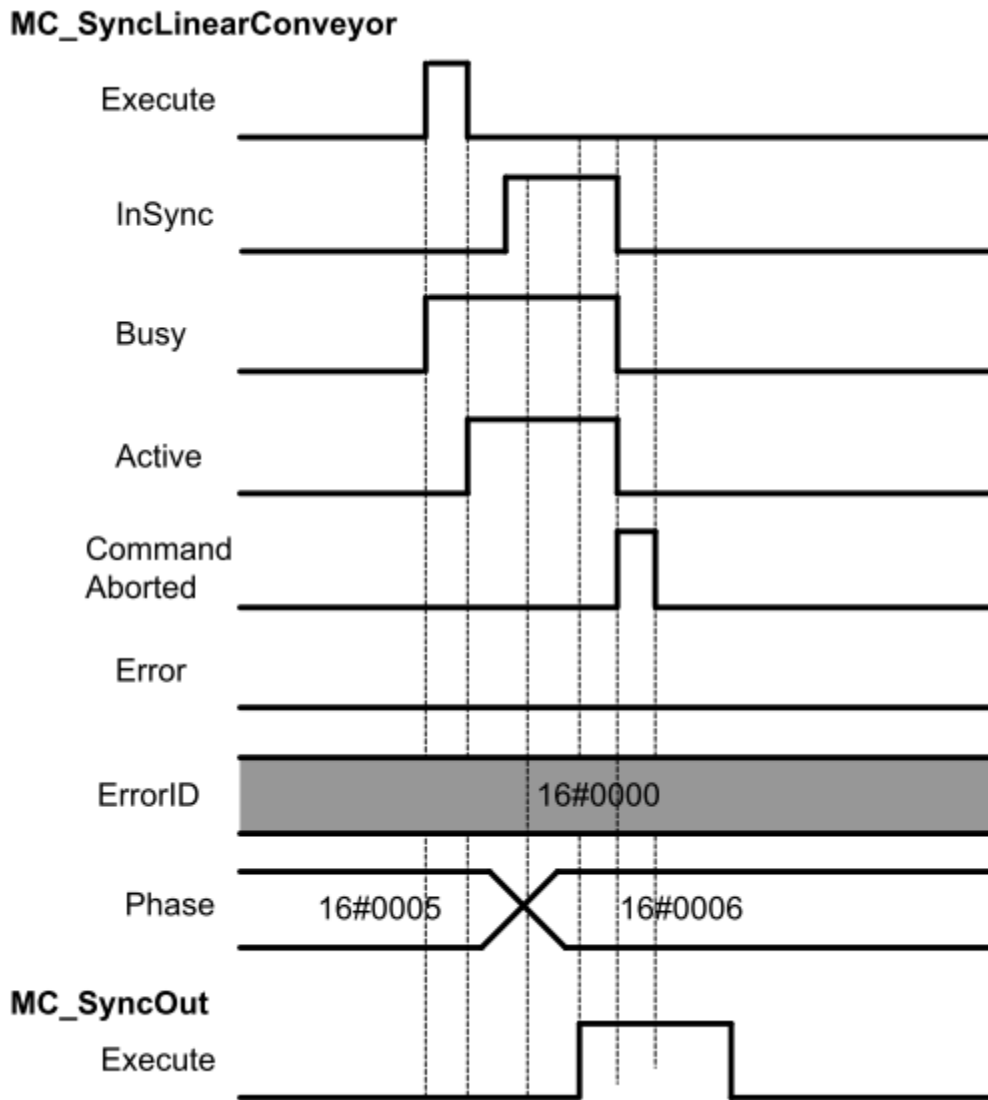
4: $T1 + T2 < t \leq T1 + T2 + \text{One control period}$

5: $T1 + T2 + \text{One control period} < t \leq T1 + T2 + T3$

6: $T1 + T2 + T3 < t$

● Timing Charts

The timing chart of this instruction is shown below. The one for stopping this instruction by the MC_SyncOut instruction is also shown below.



● Re-execution of Motion Control Instructions

- This instruction cannot be re-executed.

A Motion Control Instruction Re-execution Disabled error (error code: 543B hex) occurs if re-execution is attempted, and all axes that are being moved by this instruction stop.

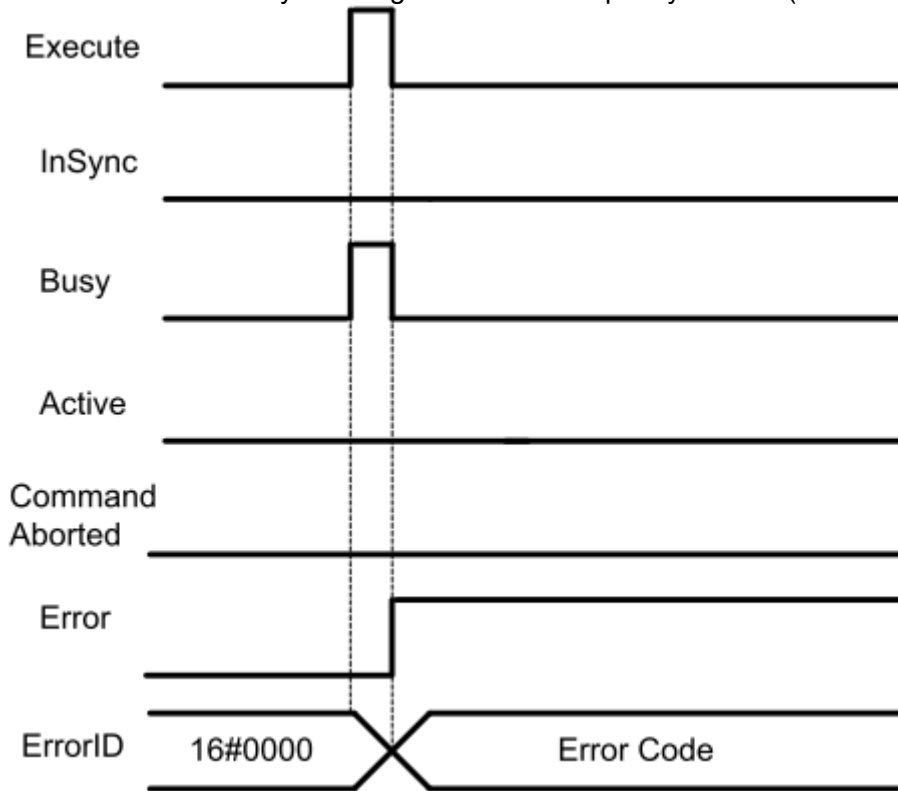
● Multi-execution of Motion Control Instructions

- A restriction applies to the instructions that can be used while this instruction is in execution.
- For details on multi-execution of motion control instructions, refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Multi-execution of Motion Control Instructions* of this document.

● Errors

If an error occurs during instruction execution, *Error* will change to TRUE.

You can find out the cause of the error by referring to the value output by *ErrorID* (Error Code).



MC_SyncOut

The MC_SyncOut instruction ends the synchronization with the conveyor.

Instruction	Name	FB/FUN	Graphic expression	ST expression
MC_SyncOut	End Synchronization	FB	<p style="text-align: center;">MC_SyncOut_instance</p>	<pre>MC_SyncOut (AxesGroup :=variable_name, Execute :=variable_name, TrajData :=variable_name, Done =>variable_name, Busy =>variable_name, CommandAborted =>variable_name, Error =>variable_name, ErrorID =>variable_name,);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
Execute	Execute	BOOL	TRUE or FALSE	FALSE	The instruction is executed when <i>Execute</i> changes to TRUE.
TrajData	Trajectory Data	<code>_sMC_SYNCOUT_TRAJ_REF</code>	---	---	Specify the parameters for releasing the synchronization.

`sMC_SYNCOUT_TRAJ_REF`

Member variable	Meaning	Data type	Valid range	Function
SyncStopType	Synchronization Stop Type Selection	<code>_eMC_SYNC_STOP_TYPE</code>	0: <code>_mcSyncStop</code> 1: <code>_mcRBT_ImmediateStop</code>	Specify the trajectory type to create the command position for releasing the synchronization with conveyor. 0: Synchronization Stop 1: Immediate Stop
TrajTime	Trajectory Target Time	ARRAY [0..7] OF UINT	---	Specify the target time for synchronization stop. The unit is [ms].
TrajTransition	Trajectory Transition Ratio	ARRAY [0..7] OF UINT	0	Reserved
TrajDistance	Trajectory Distance	ARRAY [0..7] OF LREAL	Negative number, positive number, or 0	Specify the distance for upward/downward movement from current position to target position. The unit is [mm].
MaxVelocity	Velocity Error Detection Value	LREAL	Positive number	Specify the velocity at which a velocity error (too fast) is detected when creating a command position. The unit is [mm/s].
MaxAcceleration	Acceleration Error Detection Value	LREAL	Positive number	Specify the acceleration rate at which an acceleration error (too much acceleration/deceleration) is detected when creating a command position. The unit is [mm/s ²].

● Output Variables

Name	Meaning	Data type	Valid range	Description
Done	Done	BOOL	TRUE or FALSE	TRUE when the instruction is completed.
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
CommandAborted	Command Aborted	BOOL	TRUE or FALSE	TRUE when the instruction is aborted.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#0000 indicates normal execution.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
Done	When the specified target time is reached.	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one control period when <i>Execute</i> is FALSE.
Busy	When <i>Execute</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Done</i> changes to TRUE. When <i>Error</i> changes to TRUE. When <i>CommandAborted</i> changes to TRUE.
Active	When the axis starts moving	<ul style="list-style-type: none"> When <i>Done</i> changes to TRUE. When <i>Error</i> changes to TRUE. When <i>CommandAborted</i> changes to TRUE.
CommandAborted	<ul style="list-style-type: none"> When this instruction is aborted because another motion control instruction was executed with the Buffer Mode set to <i>Aborting</i>. When this instruction is canceled due to an error. When this instruction is executed while there is an error. When this instruction is started during MC_GroupStop instruction execution. 	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one control period when <i>Execute</i> is FALSE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	When the error is cleared.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
AxesGroup	Axes Group	_sGROUP_REF	--	Specify the axes group.

■ Function

- This instruction is used to release the synchronization of the robot (axes group) in synchronized motion by the MC_SyncLinearConveyor instruction.
- This instruction is completed when the specified target time is reached. No in-position check is performed.
- The details are described below.

**Precautions for Safe Use**

- After the adjustment of mechanical home of the robot, please move the robot.
- Do not use robot at out of the workspace, without workspace check function. If you use may cause damage to the robot.
- If you want to stop the movement of the running in the instruction MC_SyncLinearConveyor, please use the instruction MC_SyncOut. Other instructions are not smooth deceleration at the stop.
- In Robot Ver. 1.01 or later, the velocity error (too fast) or the acceleration error (too much acceleration/deceleration) are not detected when MaxVelocity or MaxAcceleration are set to 0 or they are default.

Set correct values to MaxVelocity and MaxAcceleration in order to avoid unexpected velocity and acceleration/deceleration.

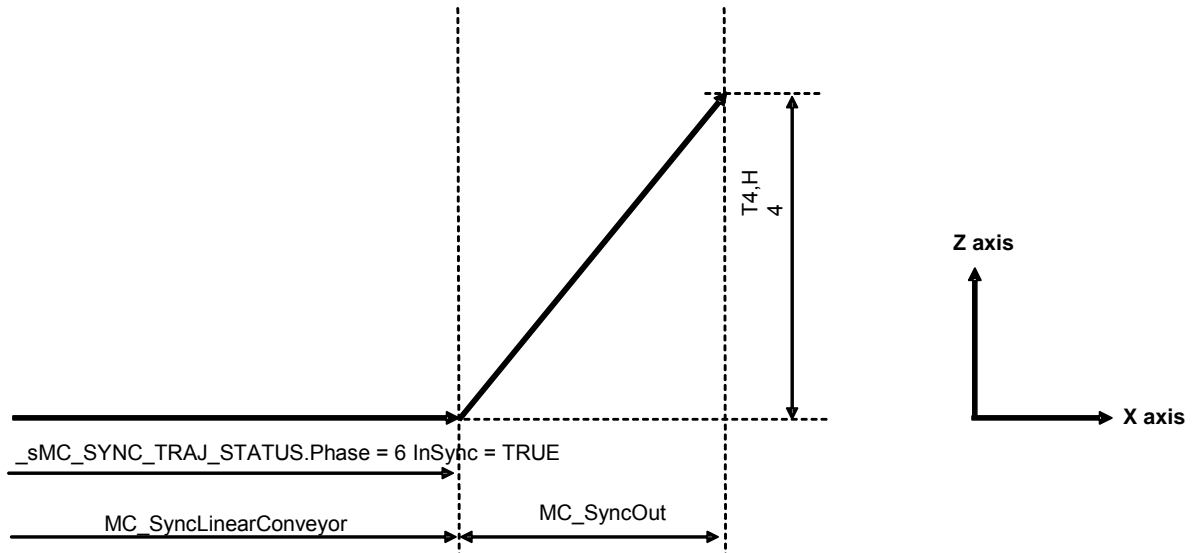
- In Robot Ver. 1.01 or later, the velocity error or the acceleration error are not detected at the moment of starting this instruction when ToolID is set other than 0.

Please pay attention to that.

● Specifying the Trajectory

- The parameters to create the trajectory for releasing the synchronization are specified by *SyncStopType*, *TrajTime*, *TrajTransition*, *TrajDistance*, *MaxVelocity*, and *MaxAcceleration* of the Trajectory Data (*TrajData*).

- Two stop types, immediate stop and synchronization stop, are supported for Synchronization Stop Type Selection (*SyncStopType*).
- If the immediate stop is selected when this instruction is executed, the operation of the MC_SyncLinearConveyor instruction is stopped and also the creation of command position is stopped after that.
- The details of the stop operation and setting parameters for synchronization stop are described below.



<Terms>

_sMC_SYNC_TRAJ_STATUS.Phase = 6 : Phase of the trajectory indicates the 6 status.

T4: Trajectory Target Time [ms]

H4: Trajectory Distance in Machine coordinate system (MCS) Z axis direction [mm]

- The command position to move the robot from the current position to “current position + H4” (Z axis) in T4 is created.
- When switching from the MC_SyncLinearConveyor operation to the MC_SyncOut operation, a command position with continuous velocity is created.
- The following values are set for Trajectory Target Time (*TrajTime*), Trajectory Transition Ratio (*TrajTransition*), and Trajectory Distance (*TrajDistance*).

TrajTime

Name	Data type	Valid range	Description
TrajTime[0]	UINT	Positive number or 0	T4 [ms]
TrajTime[1]-[7]	UINT	0	Reserved

TrajTransition (Not used)

Name	Data type	Valid range	Description
TrajTransition[0]-[7]	UINT	0	Reserved

TrajDistance

Name	Data type	Valid range	Description
TrajDistance[0]	LREAL	Positive number or 0	Trajectory Distance Z axis direction in Machine coordinate system (MCS). H4 [mm]
TrajDistance[1]-[7]	LREAL	0	Reserved

- If the workspace check function is enabled, perform the workspace check, and when an error detected the axis Group is stopped. Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) section 3.2.2 the details of workspace check function.
- When calculating a commanded position, velocity and acceleration of the movement are calculated and checked against the command maximums. If exceeding them, the following errors will be raised: Velocity Error Detection Value (MaxVelocity) and Acceleration Error Detection Value (MaxAcceleration).
- When SyncStopType is set to _mcSyncStop and H4 is set to 0, in Robot Ver. 1.00, the operation is stopped immediately regardless of T4. In Robot Ver. 1.01 or later, the operation will stop after synchronized moving in T4.

- Other Specifications

- This instruction can be used when all of the following conditions are satisfied.

The kinematics must have been set by the MC_SetKinTransform instruction.

For all axes registered in the axes group, the home must have been defined.

For all axes registered in the axes group, must be turned ON.

For all axes registered in the axes group, limit inputs must be OFF.

The axes group must be enabled.

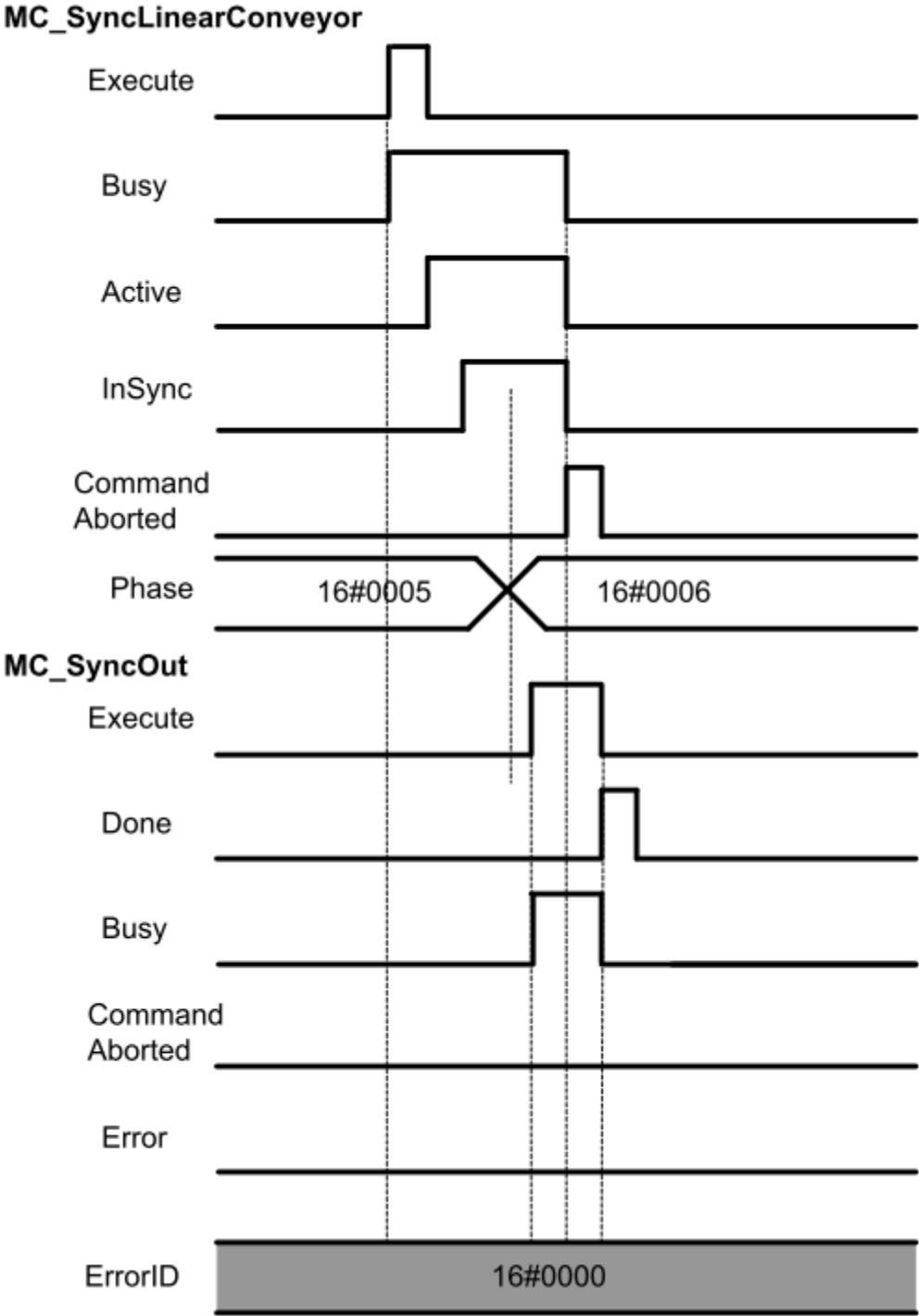
<When SyncStopType is Selected "Synchronization Stop">

Trajectory Status must be Phase 6. (Checked at instruction only at execution timing)

<When SyncStopType is Selected "Immediate Stop">

MC_SyncLinearConveyor instruction must be during execution.

- Timing Charts



● Re-execution of Motion Control Instructions

- This instruction cannot be re-executed.

A Motion Control Instruction Re-execution Disabled error (error code: 543B hex) occurs if re-execution is attempted, and all axes that are being moved by this instruction stop.

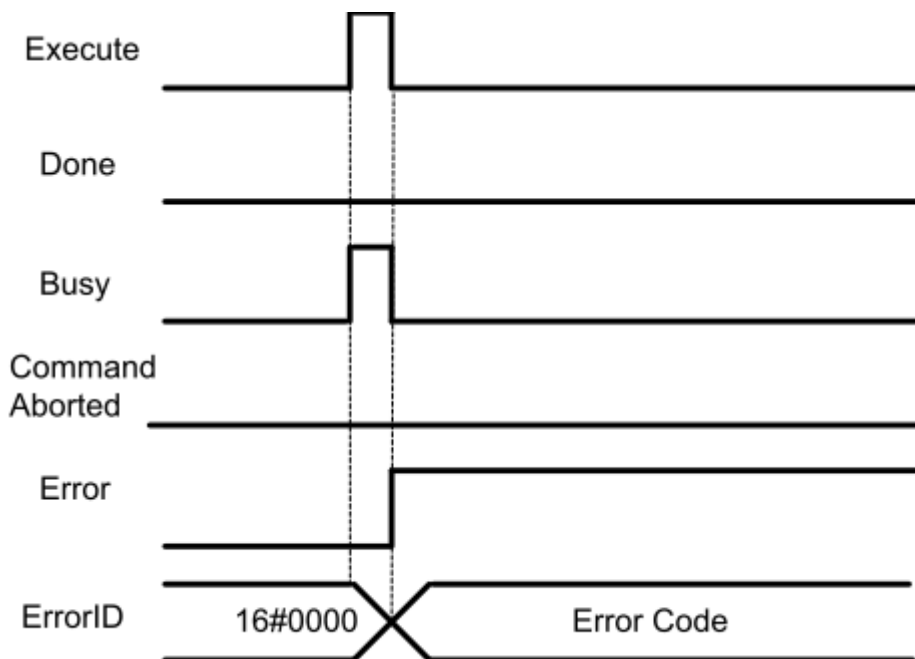
● Multi-execution of Motion Control Instructions

- A restriction applies to the instructions that can be used while this instruction is in execution. For details on multi-execution of motion control instructions, refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Multi-execution of Motion Control Instructions* of this document.

● Errors

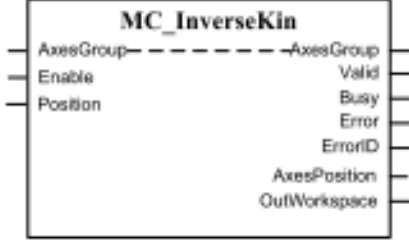
If an error occurs during instruction execution, *Error* will change to TRUE.

You can find out the cause of the error by referring to the value output by *ErrorID* (Error Code).



MC_InverseKin <Robot Ver 1.00>

The MC_InverseKin instruction performs inverse kinematics.

Instruction	Name	FB/ FUN	Graphic expression	ST expression
MC_InverseKin	Inverse Kinematics	FB	<p style="text-align: center;">MC_InverseKin_instance</p> 	<pre>MC_InverseKin(AxesGroup:=variable_name, Enable:=variable_name, Position:= variable_name, AxesGroup=>variable_name, Valid=> variable_name, Busy=> variable_name, Error=> variable_name, ErrorID=> variable_name AxesPosition=> variable_name OutWorkspace=> variable_name,);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
Enable	Enable	BOOL	TRUE or FALSE	FALSE	The target position is transformed to axis target positions of each axis when <i>Enable</i> changes to TRUE.
Position	Target Position	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	0	Specify the target position on the machine coordinate system (MCS). For Delta3, the unit is [mm].

● Output Variables

Name	Meaning	Data type	Valid range	Description
Valid	Enabled	BOOL	TRUE or FALSE	The Axis Target Position (<i>AxesPosition</i>) below is enabled when <i>Valid</i> is TRUE. The axis target position is updated every period.
AxesPosition	Axis Target Position	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	Shows the axis target position of each axis that is found by transforming the target position on the machine coordinate system (MCS). For Delta3, the unit is [deg].
OutWorkspace	Outside Workspace	BOOL	TRUE or FALSE	TRUE when the target position is outside the workspace.
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#000 indicates normal execution.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
Valid	When the kinematics transform is successfully completed.	<ul style="list-style-type: none"> When <i>Error</i> changes to TRUE. When <i>Enable</i> is FALSE.
Busy	When <i>Enable</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Enable</i> changes to FALSE. When <i>Error</i> changes to TRUE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	<ul style="list-style-type: none"> When the error is cleared.
OutWorkspace	When the target position is outside the workspace.	<ul style="list-style-type: none"> When the target position is inside the workspace. When <i>Enable</i> is FALSE. When <i>Error</i> changes to TRUE.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
AxesGroup	Axes Group	_sGROUP_REF	--	Specify the axes group.

■ Function

- This instruction is used to transform the target position (*Position*) on the machine coordinate system (MCS) of the robot to the target position on the axis coordinate system (ACS) of each axis.
- For Delta3, the settings of the specified target position (*Position*) and the transformed axis target position (*AxesPosition*) are shown below.

Position

Name	Data type	Valid range	Description
Position[0]	LREAL	Negative number, positive number, or 0	Target position of X axis on the machine coordinate system (MCS) [mm]
Position[1]	LREAL	Negative number, positive number, or 0	Target position of Y axis on the machine coordinate system (MCS) [mm]
Position[2]	LREAL	Negative number, positive number, or 0	Target position of Z axis on the machine coordinate system (MCS) [mm]
Position[3]-[5]	LREAL	0	Reserved

AxesPosition

Name	Data type	Valid range	Description
AxesPosition[0]	LREAL	Negative number, positive number, or 0	Target position of A0 axis on the axis coordinate system (ACS) [deg]
AxesPosition[1]	LREAL	Negative number, positive number, or 0	Target position of A1 axis on the axis coordinate system (ACS) [deg]
AxesPosition[2]	LREAL	Negative number, positive number, or 0	Target position of A2 axis on the axis coordinate system (ACS) [deg]
AxesPosition[3]-[5]	LREAL	0	Reserved

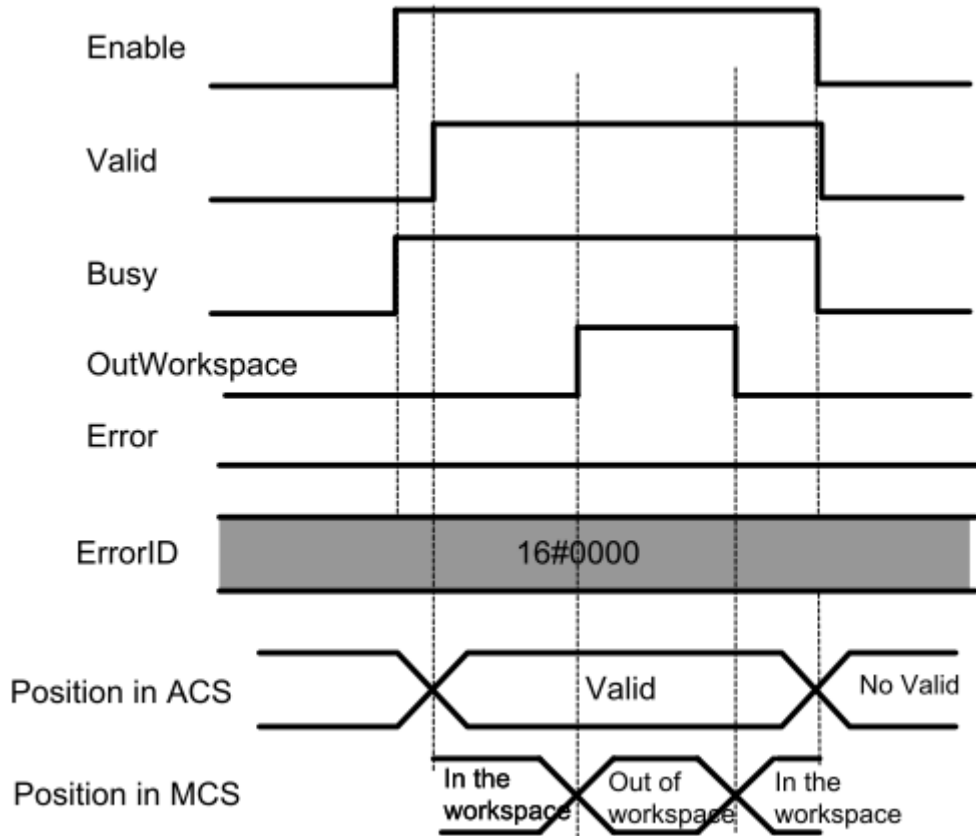
- If the target position is outside the workspace regardless of whether the workspace function is enabled or disabled, Outside Workspace (*OutWorkspace*) changes to TRUE and Enabled (*Valid*) changes to FALSE.
- If the transform target position to the target position in the Actual Coordinate System(ACS) is successfully completed at that time, the Enabled (*Valid*) changes to TRUE.

- Other Specifications

- This instruction can be used when the following condition is satisfied.

The kinematics must have been set by the MC_SetKinTransform instruction.

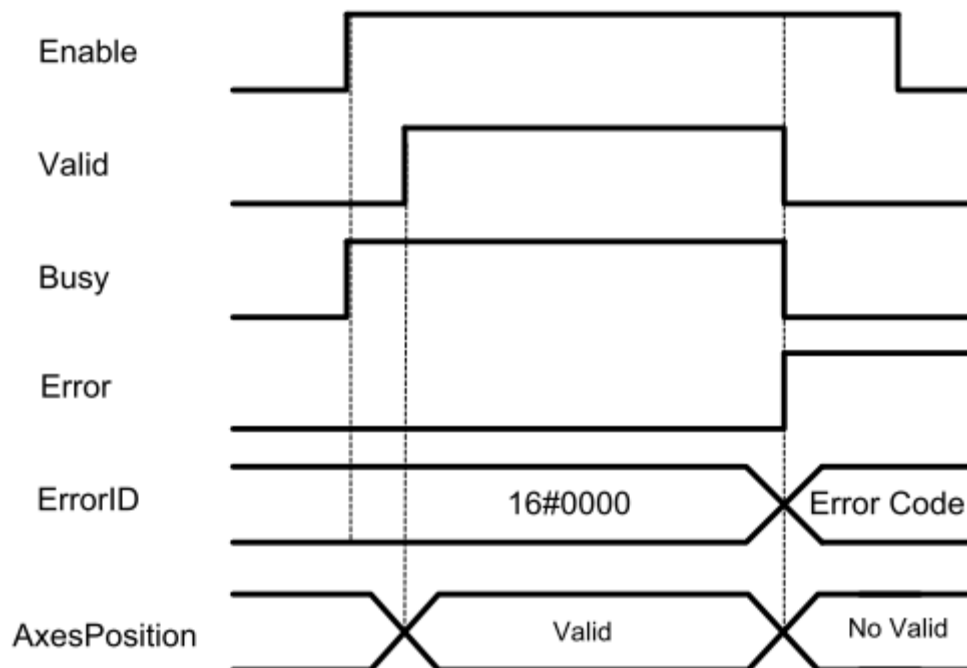
● Timing Charts



● Errors

If an error occurs during instruction execution, *Error* will change to TRUE.

You can find out the cause of the error by referring to the value output by *ErrorID* (Error Code).



MC_GroupSyncMoveAbsolute

The MC_GroupSyncMoveAbsolute instruction outputs the target positions on the axis coordinate system (ACS) set for each axis to the Servo Drive every primary period.

■ Function

- This instruction can be used regardless of whether or not the kinematics is set for the axes group.
- Refer to the NJ-series Motion Control Instructions Reference Manual (Cat.No.W508) for details of this instruction.
- It is possible to move the robot along with the desired trajectory by combining this instruction with the MC_InverseKin (Inverse Kinematics) instruction. The outline is shown below.

MC_Power	Turn ON the Servo of each axis of the robot.
MC_SetKinTransform	Set a kinematics for the axes group.
:	
MC_GroupEnable	Enable the axes group.
:	
:	Execute the steps after this every cycle.
(Calculation of trajectory)	Create a target position (Xi, Yi, Zi) on the machine coordinate system (MCS).
:	
MC_InverseKin	Transform the target position (Xi, Yi, Zi) to the target value (A0i, A1i, A2i) on the axis coordinate system (ACS).
MC_GroupSyncMoveAbsolute	Output the target position (A0i, A1i, A2i) to the Servo Drive.

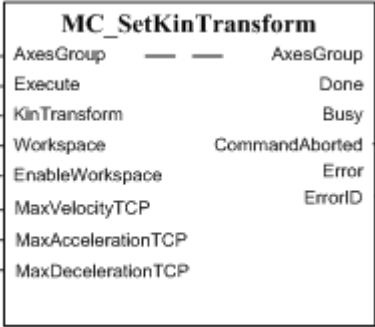


Precautions for Correct Use

- When using this instruction, those are not checked that the Workspace, MaxVelocityTCP, MaxAccelerationTCP and MaxDecelerationTCP which are set by MC_SetKinTransform. Please pay attention to that.

MC_SetKinTransform <Robot Ver 1.01>

The MC_SetKinTransform instruction sets robot control kinematics and workspace check for an axes group.

Instruction	Name	FB/ FUN	Graphic expression	ST expression
MC_SetKinTransform	Set Kinematics Transform	FB	<p>MC_SetKinTransform_instance</p> 	<pre>MC_SetKinTransform_instance(AxesGroup:=parameter, Execute:=parameter, KinTransform:= parameter, Workspace:= parameter, EnableWorkspace:=parameter, MaxVelocityTCP:=parameter, MaxAccelerationTCP:=parameter, Done =>parameter, Busy =>parameter, Error =>parameter, ErrorID =>parameter);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
Execute	Execute	BOOL	TRUE or FALSE	FALSE	The instruction is executed when <i>Execute</i> changes to TRUE.
KinTransform	Kinematics Transform	_sMC_KIN_REF	---	---	Specify the kinematics type, robot's link length and other items.
Workspace	Workspace Parameters	_sMC_WORKSPACE_REF	---	---	Specify the range of workspace.
EnableWorkspace	Enable Workspace	BOOL	TRUE or FALSE	TRUE	Specify whether to enable (TRUE) or disable (FALSE) the workspace function.
MaxVelocityTCP	Maximum interpolation Velocity of the TCP of the Robot	LREAL	Positive number or 0	0 ¹	Maximum interpolation velocity of the default robot TCP [mm/s]
MaxAccelerationTCP	Maximum interpolation Acceleration of the TCP of the Robot	LREAL	Positive number or 0	0 ²	Maximum interpolation acceleration of the default robot TCP [mm/s ²]
MaxDecelerationTCP	Maximum interpolation Deceleration of the TCP of the Robot	LREAL	Positive number or 0	0 ³	Maximum interpolation deceleration of the default robot TCP [mm/s ²]

*1 When MaxVelocityTCP is zero, Maximum interpolation velocity is not detected.

*2 When MaxAccelerationTCP is zero, Maximum interpolation acceleration is not detected.

*3 When MaxDecelerationTCP is zero, Maximum interpolation deceleration is not detected.

sMC_KIN_REF

Member variable	Meaning	Data type	Valid range	Function
KinType	Kinematics Type Selection	_eMC_KIN_TYPE	100:_mcDelta3Type1 101:_mcDelta3Type2 102:_mcDelta2Type1	Specify the kinematics type to set. 100: Delta3 101: Delta3R 102: Delta2
KinParam	Kinematics Parameters	ARRAY [0..7] OF LREAL	---	Specify the arm length and other items for the robot.
ExpansionParam	Expansion Parameters	ARRAY [0..7] OF LREAL	---	Specify the values for adjusting the rotation of the machine coordinate system (MCS).

sMC_WORKSPACE_REF

Member variable	Meaning	Data type	Valid range	Function
WorkspaceType	Workspace Type Selection	_eMC_WORKSPACE_TYPE	100: _mcDelta3Workspace 101: _mcDelta2Workspace	Specify the workspace type to set. 100: Delta3/Delta3R 101: Delta2
WorkspaceParam	Workspace Parameters	ARRAY [0..7] OF LREAL	---	Set the Volume of workspace.

● Output Variables

Name	Meaning	Data type	Valid range	Description
Done	Done	BOOL	TRUE or FALSE	TRUE when the setting is completed.
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
CommandAborted	Command Aborted	BOOL	TRUE or FALSE	TRUE when the instruction is aborted.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#0000 indicates normal execution.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
Done	When transformation is set successfully.	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one control period when <i>Execute</i> is FALSE.
Busy	When <i>Execute</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Done</i> changes to TRUE. When <i>Error</i> changes to TRUE.
CommandAborted	When the axis group is in an error state	<ul style="list-style-type: none"> When „Execute“ is TRUE and changes to FALSE One period after the „Execute“ became FALSE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	When the error is cleared.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
AxesGroup	Axes Group	_sGROUP_REF	---	Specify the axes group.

■ Function

—This instruction is used to setup robot's kinematics, workspace and the background check of maximum interpolation velocity, acceleration and deceleration:

- 1) For Kinematics: Kinematics type, robot's link length, and home adjustment parameter
- 2) For workspace check: Workspace type and workspace volume and enabling/disabling the check
- 3) For the Background check of TCP0:

Maximum interpolation velocity, maximum acceleration and maximum deceleration.



Precautions for Correct Use

- When using MC_GroupSyncMoveAbsolute, those are not checked that the Workspace, MaxVelocityTCP, MaxAccelerationTCP and MaxDecelerationTCP which are set by this instruction. Please pay attention to that.

- For Delta3, the following values are set for the variables defined by the `_sMC_KIN_REF` and `_sMC_WORKSPACE_REF` data types.

`_sMC_KIN_REF`

KinType

Name	Data type	Valid range	Description
KinType	<code>_eMC_KIN_TYPE</code>	100: <code>_mcDelta3Type1</code> 101: <code>_mcDelta3Type2</code> 102: <code>_mcDelta2Type1</code>	100: Delta3

KinParam

Name	Data type	Valid range	Description
KinParam[0]	LREAL	Positive number	Radius of the fixed frame R_f [mm] Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) <i>Section 3.3.1</i> for details.
KinParam[1]	LREAL	Positive number	Link 1 L_f [mm]
KinParam[2]	LREAL	Positive number	Radius of the moving frame R_m [mm]
KinParam[3]	LREAL	Positive number	Link 2 L_m [mm]
KinParam[4]- [7]	LREAL	0	Reserved

ExpansionParam

Name	Data type	Valid range	Description
ExpansionParam[0]	LREAL	0 to 360	Rotation of Z axis θ [deg] Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) <i>Section 3.3.1</i> for details.
ExpansionParam[1]-[7]	LREAL	0	Reserved

`_sMC_WORKSPACE_REF`

WorkspaceType

Name	Data type	Valid range	Description
WorkspaceType	<code>_eMC_WORKSPACE_TYPE</code>	100: <code>_mcDelta3Workspace</code> 101: <code>_mcDelta2Workspace</code>	100: Delta3/Delta3R

WorkspaceParam

Name	Data type	Valid range	Description
WorkspaceParam[0]	LREAL	Negative number or 0	Top surface position of cylinder Z_u [mm] Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) <i>Section 3.3.2</i> for details.
WorkspaceParam[1]	LREAL	Positive number	Radius of the cylinder R_{cy} [mm]
WorkspaceParam[2]	LREAL	Positive number	Height of the cylinder H_{cy} [mm]
WorkspaceParam[3]	LREAL	Positive number	Radius of the frustum cone R_{co} [mm]
WorkspaceParam[4]	LREAL	Positive number	Height of the frustum cone H_{co} [mm]
WorkspaceParam[5]-[7]	LREAL	0	Reserved

- For Delta3R, the description above is also valid, but the kinematics type must be changed to

“101: `_mcDelta3Type2`.”

- For Delta2, the following values are set for the variables defined by the `_sMC_KIN_REF` and `_sMC_WORKSPACE_REF` data types.

`_sMC_KIN_REF`

KinType

Name	Data type	Valid range	Description
KinType	<code>_eMC_KIN_TYPE</code>	100: <code>_mcDelta3Type1</code> 101: <code>_mcDelta3Type2</code> 102: <code>_mcDelta2Type1</code>	102: Delta2

KinParam

Name	Data type	Valid range	Description
KinParam[0]	LREAL	Positive number	Radius of the fixed frame R_f [mm] Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) Section 3.3.1 for details.
KinParam[1]	LREAL	Positive number	Link 1 L_f [mm]
KinParam[2]	LREAL	Positive number	Radius of the moving frame R_m [mm]
KinParam[3]	LREAL	Positive number	Link 2 L_m [mm]
KinParam[4]-[7]	LREAL	0	Reserved

ExpansionParam

For Delta2, this parameter cannot be used.

Name	Data type	Valid range	Description
ExpansionParam[0]-[7]	LREAL	0	Reserved

`_sMC_WORKSPACE_REF`

WorkspaceType

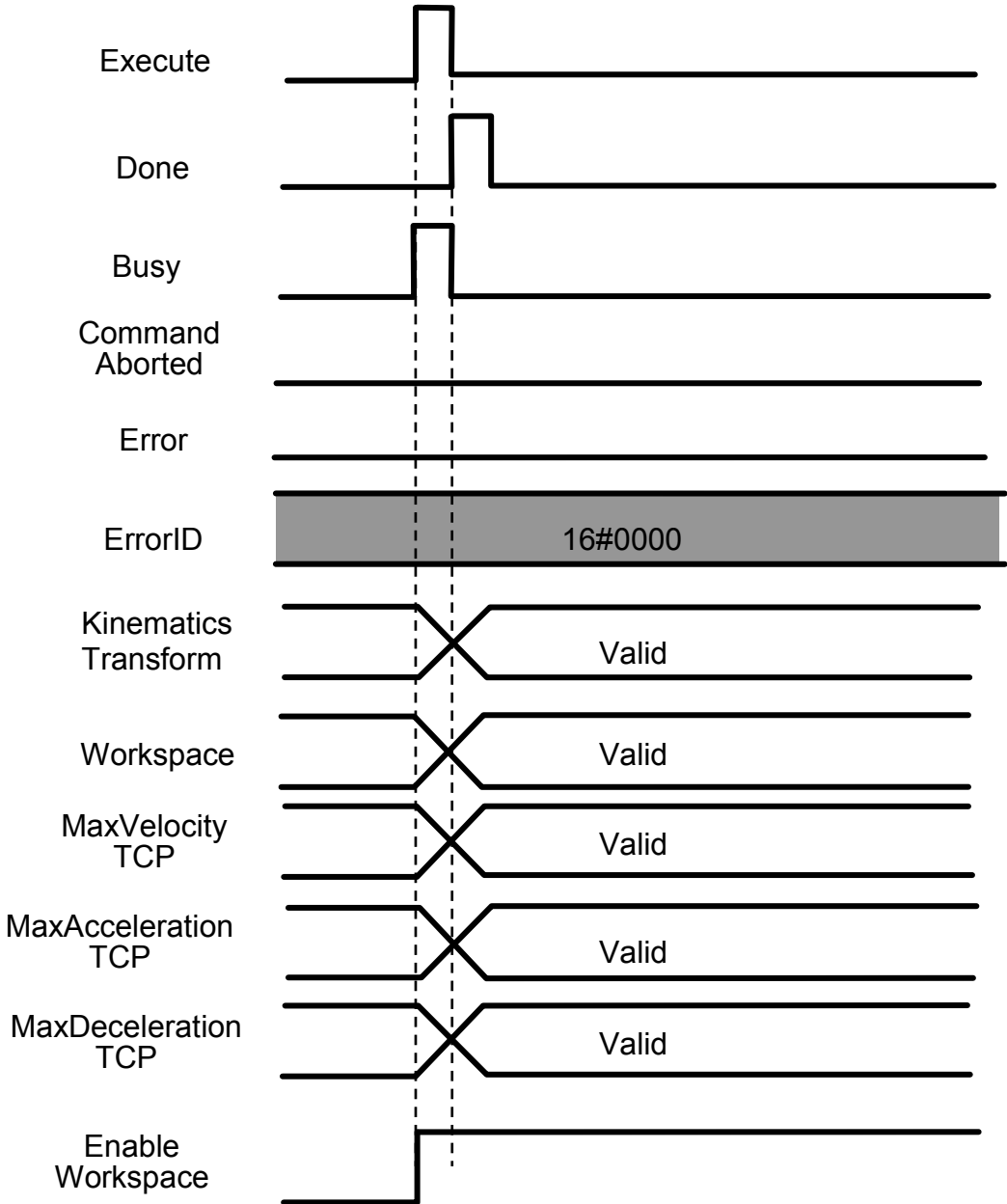
Name	Data type	Valid range	Description
WorkspaceType	<code>_eMC_WORKSPACE_TYPE</code>	100: <code>_mcDelta3Workspace</code> 101: <code>_mcDelta2Workspace</code>	101: Delta2

WorkspaceParam

Name	Data type	Valid range	Description
WorkspaceParam[0]	LREAL	Negative number or 0	Z coordinate of the workspace volume top side Z_u [mm] Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) Section 3.3.2 for details.
WorkspaceParam[1]	LREAL	Positive number	Half length of the top side of the trapezoid R_{top} [mm]
WorkspaceParam[2]	LREAL	Positive number	Half length of the bottom side of the trapezoid R_{bottom} [mm]
WorkspaceParam[3]	LREAL	Positive number or 0	Height of the trapezoid H [mm]
WorkspaceParam[4]-[7]	LREAL	0	Reserved

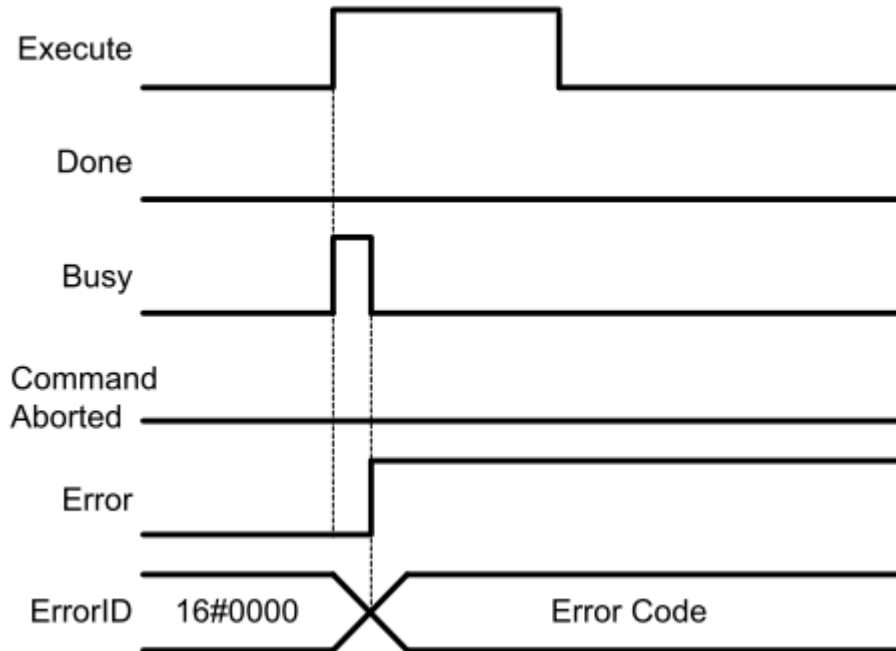
- When the power supply to the Controller is turned OFF, the settings of kinematics and workspace and maximum interpolation velocity and maximum interpolation acceleration and maximum interpolation deceleration that you set in this instruction will be initialized. If use the robot function, it have to reset the kinematics and the workspace parameters by this instruction each time when the power is turned on.
- The setting value will be retained even after switching to program mode of the CPU Unit.
- Following robot instructions be able to use, after one control period of MC_SetKinTransform instruction is done(Done = TRUE).
 - MC_DefineCoordSystem, MC_GroupMon, MC_MoveTimeAbsolute,
 - MC_SyncLinearConveyor, MC_SyncOut, MC_InverseKin, MC_DefineToolTransform, MC_RobotJog
- Even if kinematics parameters are set by this instruction, you can move each axes by using Axis instructions during the group axis is disabled.
- Even if Disable workspace check (EnableWorkspace = FALSE input variable) is set, the parameters for the workspace check function (parameters of the input variables defined in _sMC_WORKSPACE_REF) are set.
- If Kinematics Initialization Error (error code: 571 F hex) has occurred, the pre-registered Kinematics parameters are removed. For this reason, if this error occurs, please reset the kinematics parameters.
- This instruction can be used when all of the following conditions are satisfied.
 - The number of axes registered in the axes group must be the same as the number of axes of the kinematics (robot).
 - Axis display units that are registered in the group axis must be degree.
 - For all axes registered in the axes group, the count mode must be set to linear mode.
 - For all axes registered in the axes group, the home must have been defined.
 - The axes group must be disabled.

• Timing Charts




- Errors

If an error occurs during instruction execution, *Error* will change to TRUE. You can find out the cause of the error by referring to the value output by *ErrorID*.



MC_DefineToolTransform <Robot Ver 1.01>

The MC_DefineTooltransform instruction sets a robot tool for a specified robot (axes group).

Instruction	Name	FB/ FUN	Graphic expression	ST expression
MC_DefineToolTransform	Define Tool System	FB	<p>MC_DefineToolTransform_instance</p> 	<pre>MC_DefineToolTransform_instance(AxesGroup:=parameter, Execute:=parameter, ToolTransform:= parameter, Done =>parameter, Busy =>parameter, Error =>parameter, ErrorID =>parameter);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
Execute	Execute	BOOL	TRUE or FALSE	FALSE	The instruction is executed when <i>Execute</i> changes to TRUE.
ToolTransform	Tool Coordinate System Transformation	_sMC_COORD_REF	---	---	Specify the tool coordinate system to set.

sMC_TOOL_COORD_REF

Member Variable	Meaning	Data type	Valid range	Function
ToolID	Robot Tool ID	UINT	1 to 16	Specify the ID of Robot tool to set.
Pose	Tool Coordinate System Transformation definition	ARRAY [0..5] OF LREAL	---	Specify the pose of tool coordinate system based on TCP.

● Output Variables

Name	Meaning	Data type	Valid range	Description
Done	Done	BOOL	TRUE or FALSE	TRUE when the setting is completed.
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
CommandAborted	Command Aborted	BOOL	TRUE or FALSE	TRUE when the instruction is aborted.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#0000 indicates normal execution.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
Done	When the setting is successfully completed.	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one period when <i>Execute</i> is FALSE.
Busy	When <i>Execute</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Done</i> changes to TRUE. When <i>Error</i> changes to TRUE.
CommandAborted	<ul style="list-style-type: none"> When this instruction is canceled due to an error. When this instruction is executed while there is an error. 	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one period when <i>Execute</i> is FALSE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	When the error is cleared.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
------	---------	-----------	-------------	-------------

AxesGroup	Axes Group	_sGROUP_REF	---	Specify the axes group.
-----------	------------	-------------	-----	-------------------------

■ Function

- This instruction is used to set a user coordinate system (*Tool ID* = 1 to 16) for a specified robot (axes group).
- Up to 16 tool coordinate systems can be set for each robot.
- When the tool coordinate system is successfully registered, it can be referred to by MC_GroupMon, MC_MoveTimeAbsolute, and MC_SyncLinearConveyor instructions after that.
- When setting a user coordinate system, the following values are set for the variables defined by the _sMC_TOOL_COORD_REF data type.

_sMC_TOOL_COORD_REF

Name	Data type	Valid range	Description
Pose[0]	LREAL	Positive number, negative number, or 0	Position of home in the tool coordinate system(UCS) that is seen from X axis home in tool center point(TCP) Tx[mm]
Pose[1]	LREAL	Positive number, negative number, or 0	Position of home in the tool coordinate system(UCS) that is seen from Y axis home in tool center point(TCP) Ty[mm]
Pose[2]	LREAL	Positive number, negative number, or 0	Position of home in the tool coordinate system(UCS) that is seen from Z axis home in tool center point(TCP) Tz[mm]
Pose[3]	LREAL	-180 to 180	Rotation around the X axis of tool coordinate system (TCS) Rx [deg]
Pose[4]	LREAL	-90 to 90	Rotation around the Y axis of tool coordinate system (TCS) Ry [deg]
Pose[5]	LREAL	-180 to 180	Rotation around the Z axis of tool coordinate system (TCS) Rz [deg]

- The setting value will be retained even after switching to program mode of the CPU unit.
- This instruction can be used when the following condition is satisfied.

The kinematics must have been set by the MC_SetKinTransform instruction.

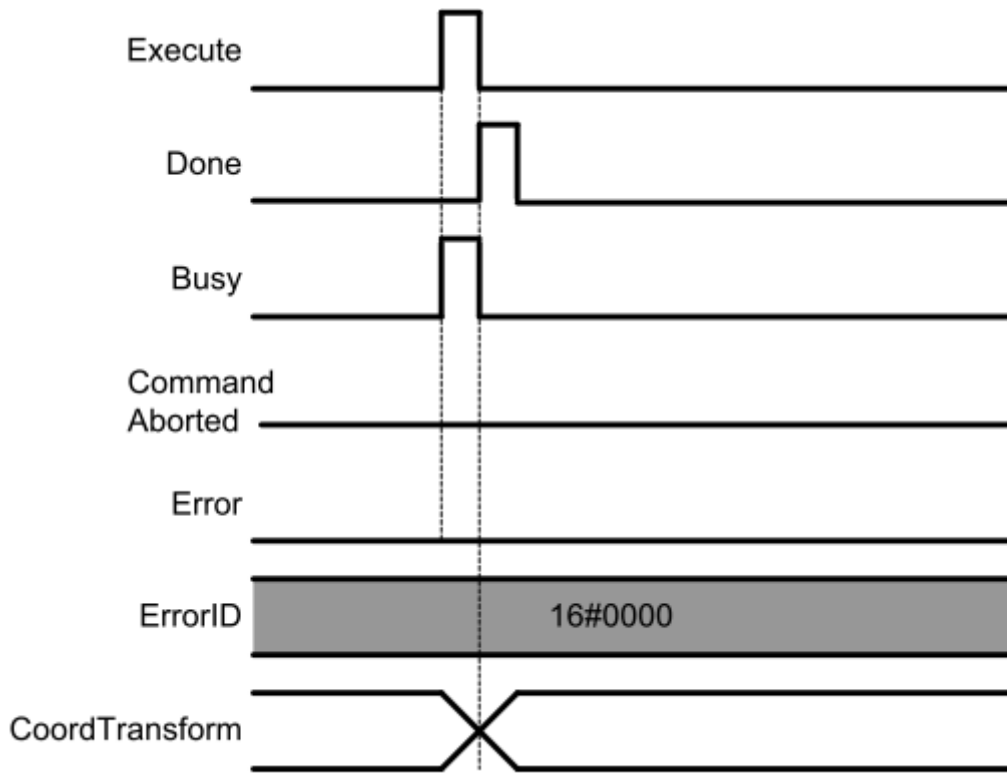


Precautions for Correct Use

- Each Robot Tool is defined against a valid MCS.

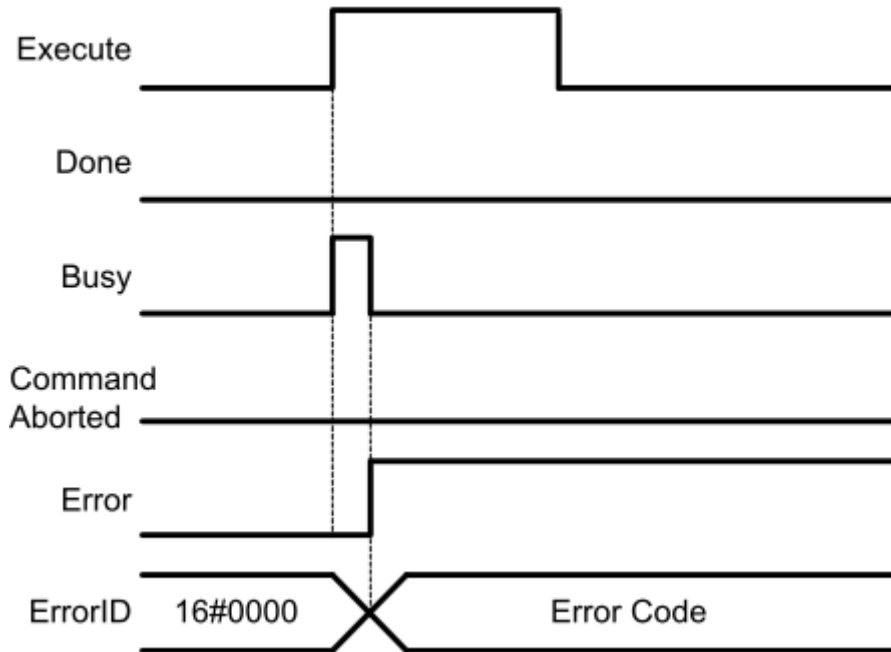
It means that each Robot Tool of one robot must be defined again by executing MC_DefineToolTransform for each Robot Tool, if the MC_SetKinTransform was executed again on that robot.

• Timing Charts



- Errors

If an error occurs during instruction execution, *Error* will change to TRUE. You can find out the cause of the error by referring to the value output by *ErrorID*.



MC_GroupMon <Robot Ver 1.01>

The MC_GroupMon instruction reads the current position and current velocity of a robot.

Instruction	Name	FB/ FUN	Graphic expression	ST expression
MC_GroupMon	Monitor Axes Group	FB	<p style="text-align: center;">MC_GroupMon_instance</p>	<pre>MC_GroupMon (AxesGroup:=variable_name, Enable:=variable_name, CoordSystem:= variable_name, CSID:= variable_name, ToolID:= variable_name, AxesGroup=>variable_name, Valid=> variable_name, Busy=> variable_name, Error=> variable_name, ErrorID=> variable_name Position=> variable_name, Velocity=> variable_name,);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
Enable	Enable	BOOL	TRUE or FALSE	FALSE	Reading the current values is started when <i>Enable</i> changes to TRUE.
CoordSystem	Coordinate System	_eMC_RBT_COORD_SYSTEM	1: _mcRBT_MCS 3: _mcRBT_UCS	1	Specify the coordinate system. 1: Machine coordinate system (MCS) 3: User coordinate system (UCS)
CSID	Coordinate System ID	UINT	0 to 15	0	Specify the ID of coordinate system when a user coordinate system is specified.
ToolID	Robot Tool ID	UINT	0 to 16	0	Specify the selected Robot Tool for monitoring

● Output Variables

Name	Meaning	Data type	Valid range	Description
Valid	Enabled	BOOL	TRUE or FALSE	When this value is TRUE, the following current values (<i>Position</i> and <i>Velocity</i>) are valid. The current position is updated every period.
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#0000 indicates normal execution.
Position	Current Position	_sMC_POSITION_REF	---	Shows command current position and actual current position.
Velocity	Current Velocity	_sMC_VELOCITY_REF	---	Shows actual current velocity of each axis and TCP.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

sMC_POSITION_REF

Member variable	Meaning	Data type	Valid range	Function
CommandPosition	Command Current Position	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	Shows the command current position of TCP on the specified coordinate system.
ActualPosition	Actual Current Position	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	Shows the actual current position of TCP on the specified coordinate system.

sMC_VELOCITY_REF

Member variable	Meaning	Data type	Valid range	Function
ActualVelocityTCP	Actual TCP Current Velocity	LREAL	Positive number or 0	Shows the actual current velocity of TCP. The unit is [mm/s].
ActualVelocity	Actual Current Velocity	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	Shows the actual current velocity of each axis when viewing the actual velocity of TCP on the specified coordinate system. The unit is [mm/s].

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
Valid	When reading the current values is completed successfully.	<ul style="list-style-type: none"> When <i>Error</i> changes to TRUE. When <i>Enable</i> changes to TRUE.
Busy	When <i>Enable</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Enable</i> changes to FALSE. When <i>Error</i> changes to TRUE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	When the error is cleared.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
AxesGroup	Axes Group	_sGROUP_REF	--	Specify the axes group.

■ Function

- This instruction is used to read the current position (command current position and actual current position) and the current velocity (actual TCP current velocity and actual current velocity) of a specified robot (axes group) on a specified coordinate system.
- Command/Actual Current position and Actual TCP Current Velocity/Actual Current Velocity on the specified coordinate system are calculated by using direct kinematics from the command current position and actual current position. In this reason, the output variables(Position/Velocity) of this instruction includes the calculation error. This calculation error is affected on the length of the robot link, encoder resolution, and reduction ratio.
- Command current position will be updated every control period when the group is TRUE and output variable is TRUE. The command current position value is the latest value when the axis group was Enabled and output variable Valid was TRUE.
- Actual current position, Actual TCP Current Velocity and Actual Current Velocity are updated every control period, even group is Enabled or Disabled when the output variable Valid is TRUE.
- For Delta3 and Delta2, the following values are read out to the variables defined by the `_sMC_POSITION_REF` and `_sMC_VELOCITY_REF` data types.

`_sMC_POSITION_REF`

CommandPosition

Name	Data type	Valid range	Description
CommandPosition[0]	LREAL	Negative number, positive number, or 0	Command current position of X axis [mm]
CommandPosition[1]	LREAL	Negative number, positive number, or 0	Command current position of Y axis [mm]
CommandPosition[2]	LREAL	Negative number, positive number, or 0	Command current position of Z axis [mm]
CommandPosition[3]-[5]	LREAL	0	Reserved

ActualPosition

Name	Data type	Valid range	Description
------	-----------	-------------	-------------

ActualPosition[0]	LREAL	Negative number, positive number, or 0	Actual current position of X axis [mm]
ActualPosition[1]	LREAL	Negative number, positive number, or 0	Actual current position of Y axis [mm]
ActualPosition[2]	LREAL	Negative number, positive number, or 0	Actual current position of Z axis [mm]
ActualPosition[3]-[5]	LREAL	0	Reserved

_sMC_VELOCITY_REF

ActualVelocityTCP

Name	Data type	Valid range	Description
ActualVelocityTCP	LREAL	Positive number or 0	Actual current velocity of the TCP. TCP velocity = $\text{SQRT}(V_x^2 + V_y^2 + V_z^2)$ [mm/s]

ActualVelocity

Name	Data type	Valid range	Description
ActualVelocity[0]	LREAL	Negative number, positive number, or 0	Actual current velocity of X axis (V_x) [mm/s]
ActualVelocity[1]	LREAL	Negative number, positive number, or 0	Actual current velocity of Y axis (V_y) [mm/s]
ActualVelocity[2]	LREAL	Negative number, positive number, or 0	Actual current velocity of Z axis (V_z) [mm/s]
ActualVelocity[3]-[5]	LREAL	0	Reserved

- For Delta3R, the following values are read out to the variables defined by the `_sMC_POSITION_REF` and `_sMC_VELOCITY_REF` data types.

`_sMC_POSITION_REF`

CommandPosition

Name	Data type	Valid range	Description
CommandPosition[0]	LREAL	Negative number, positive number, or 0	Command current position of X axis [mm]
CommandPosition[1]	LREAL	Negative number, positive number, or 0	Command current position of Y axis [mm]
CommandPosition[2]	LREAL	Negative number, positive number, or 0	Command current position of Z axis [mm]
CommandPosition[3]	LREAL	Negative number, positive number, or 0	Command current position of robot rotation around X axis [deg]
CommandPosition[4]	LREAL	Negative number, positive number, or 0	Command current position of robot rotation around Y axis [deg]
CommandPosition[5]	LREAL	Negative number, positive number, or 0	Command current position of robot rotation around Z axis [deg]

ActualPosition

Name	Data type	Valid range	Description
ActualPosition[0]	LREAL	Negative number, positive number, or 0	Actual current position of X axis [mm]
ActualPosition[1]	LREAL	Negative number, positive number, or 0	Actual current position of Y axis [mm]
ActualPosition[2]	LREAL	Negative number, positive number, or 0	Actual current position of Z axis [mm]
ActualPosition[3]	LREAL	Negative number, positive number, or 0	Actual current position of robot rotation around X axis [deg]
ActualPosition[4]	LREAL	Negative number, positive number, or 0	Actual current position of robot rotation around Y axis [deg]
ActualPosition[5]	LREAL	Negative number, positive number, or 0	Actual current position of robot rotation around Z axis [deg]

`_sMC_VELOCITY_REF`

ActualVelocityTCP

Name	Data type	Valid range	Description
ActualVelocityTCP	LREAL	Positive number or 0	Actual current velocity of the TCP. TCP velocity = $\text{SQRT}(V_x^2 + V_y^2 + V_z^2)$ [mm/s]

ActualVelocity

Name	Data type	Valid range	Description
ActualVelocity[0]	LREAL	Negative number, positive number, or 0	Actual current velocity of X axis (V_x) [mm/s]
ActualVelocity[1]	LREAL	Negative number, positive number, or 0	Actual current velocity of Y axis (V_y) [mm/s]
ActualVelocity[2]	LREAL	Negative number, positive number, or 0	Actual current velocity of Z axis (V_z) [mm/s]
ActualVelocity[3]	LREAL	Negative number, positive number, or 0	Actual current velocity of robot rotation around X axis (VR_x) [deg/s]
ActualVelocity[4]	LREAL	Negative number, positive number, or 0	Actual current velocity of robot rotation around Y axis (VR_y) [deg/s]
ActualVelocity[5]	LREAL	Negative number, positive number, or 0	Actual current velocity of robot rotation around Z axis (VR_z) [deg/s]

- When the Valid output has changed from TRUE to FALSE, the value of the Position and Velocity Output variables are retained.

- This instruction can be used when all of the following conditions are satisfied.

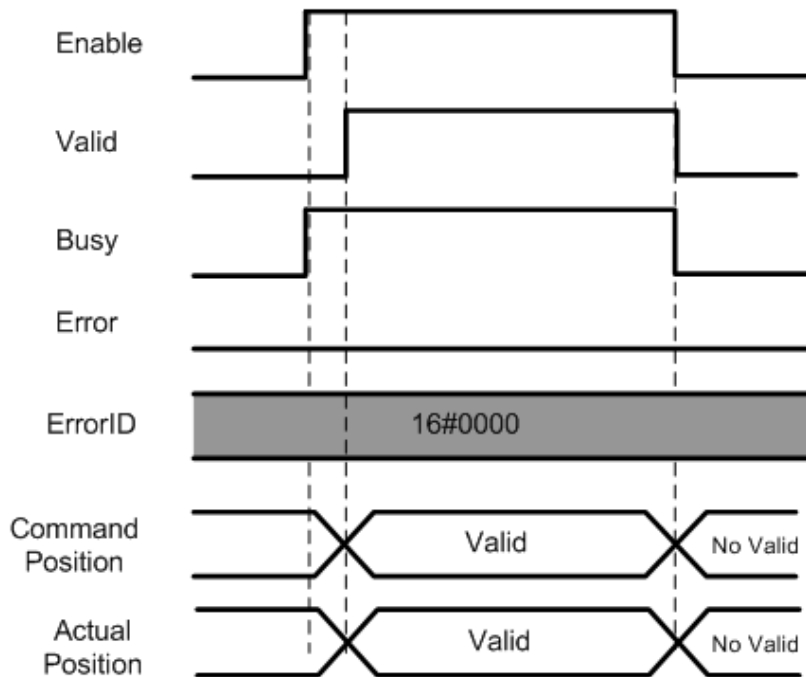
The kinematics must have been set by the MC_SetKinTransform instruction.

For all axes registered in the axes group, the home must have been defined.

When a user coordinate system is used, the coordinate system ID, which is specified by the MC_DefineCoordSystem instruction, must be set.

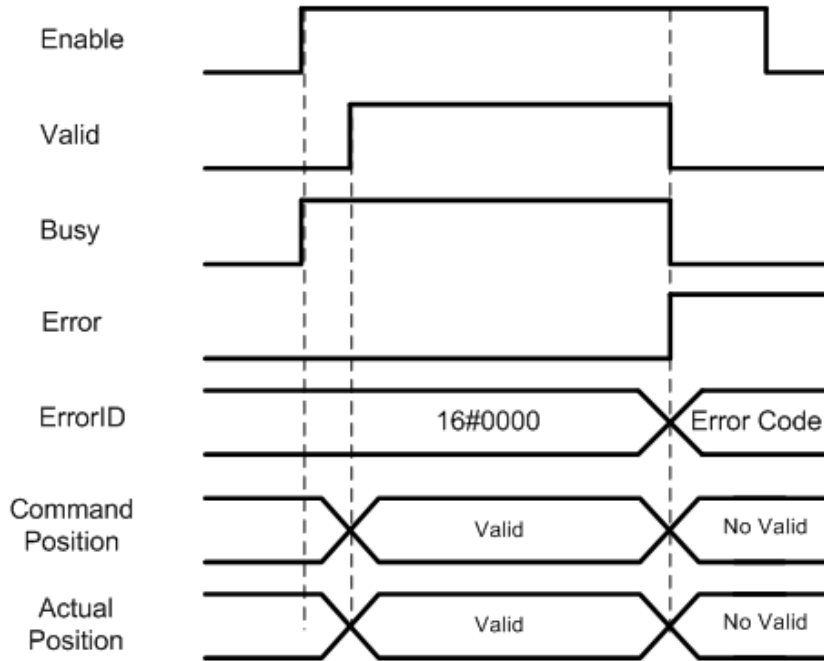
When a Robot tool is used, the tool ID, which is specified by the MC_DefineToolTransform instruction, must be set.

● Timing Charts




● Errors

If an error occurs during instruction execution, *Error* will change to TRUE. You can find out the cause of the error by referring to the value output by *ErrorID*.



MC_MoveTimeAbsolute <Robot Ver 1.01>

The MC_MoveTimeAbsolute instruction creates command values to reach a specified target position in a specified time.

Instruction	Name	FB/FUN	Graphic expression	ST expression
MC_MoveTimeAbsolute	Time-specified Absolute Positioning	FB	<p>MC_MoveTimeAbsolute_instance</p>  <p>The graphic expression shows a rectangular block labeled 'MC_MoveTimeAbsolute'. On the left side, there are input lines for: AxesGroup, Execute, Position, Direction, TrajData, CoordSystem, CSID, ToolID, BufferMode, TransitionMode, and TransitionParameter. On the right side, there are output lines for: Done, Busy, Active, CommandAborted, Error, and ErrorID. A dashed line connects the two AxesGroup inputs.</p>	<pre>MC_MoveTimeAbsolute (AxesGroup :=variable_name, Execute :=variable_name, Position :=variable_name, Direction :=variable_name, TrajData :=variable_name, CoordSystem :=variable_name, CSID :=variable_name, ToolID :=variable_name, BufferMode :=variable_name, TransitionMode :=variable_name, TransitionParameter :=variable_name, AxesGroup =>variable_name, Done =>variable_name, Busy => variable_name, Active =>variable_name, CommandAborted =>variable_name, Error =>variable_name, ErrorID =>variable_name,);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
Execute	Execute	BOOL	TRUE or FALSE	FALSE	The instruction is executed when <i>Execute</i> changes to TRUE.
Position	Target Position	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	0	Specify the target position. For Delta3, the unit is [mm].
Direction	Rotation Direction	ARRAY [0..3] OF _eMC_DIRECTION	0: _mcPositiveDirection 2: _mcNegativeDirection 4: _mcNoDirection	4	Specify the direction of rotation of the rotation axis. 0: Positive direction 2: Negative direction 4: No direction
TrajData	Trajectory Data	_sMC_MOVE_TRAJ_REF	---	---	Specify the parameters for creating command current positions up to the target position.
CoordSystem	Coordinate System	_eMC_RBT_COORD_SYSTEM	1: _mcRBT_MCS 3: _mcRBT_UCS	1	Specify the coordinate system of the target position. 1: Machine coordinate system (MCS) 3: User coordinate system (UCS)
CSID	Coordinate System ID	UINT	0 to 15	0	Specify the ID of coordinate system when a user coordinate system is specified.
ToolID	Robot Tool ID	UINT	0 to 16	0	Specify the selected Robot Tool to command.
BufferMode	Buffer Mode Selection	_eMC_BUFFER_MODE	0: _mcAborting 1: _mcBuffered 2: _mcBlending Low 3: _mcBlendingPrevious 4: _mcBlendingNext 5: _mcBlendingHigh	0	Specify the behavior when executing more than one motion control instruction. 0: Aborting 1: Buffered 2: Blending Low 3: BlendingPrevious 4: BlendingNext 5: BlendingHigh
TransitionMode	Transition Mode	_eMC_TRANSITION_MODE	0: _mcTMNone 11: _mcTMStartHeight 12: _mcTMStartRe	0	Specify the path of motion. 0: Transition disabled 11 Start Height 12: Start Remaining Time

TransitionParameter	Transition Parameters	ARRAY [0..7] OF LREAL	mainingTime Negative number, positive number, or 0	0	Set the parameters for transition. (Reserved)
---------------------	-----------------------	--------------------------	---	---	--

sMC_MOVE_TRAJ_REF

Member variable	Meaning	Data type	Valid range	Function
MoveTrajType	Movement Trajectory Type Selection	_eMC_MOVE_TRAJ_TYPE	0: _mcPolynomial3 1: _mcRBT_ModifiedSine	Specify the trajectory type to create command current position of each axis. 0: Third-order polynomial 1: Modified sine
TrajTime	Trajectory Target Time	UINT	Positive number	Specify the time to create command current positions from current position to target position. The unit is [ms].
MaxVelocity	Velocity Error Detection Value	LREAL	Positive number	Specify the velocity at which a velocity error (too fast) is detected when creating a command current position of active tool. The unit is [mm/s].
MaxAcceleration	Acceleration Error Detection Value	LREAL	Positive number	Specify the acceleration rate at which an acceleration error (too much acceleration/deceleration) is detected when creating a command current position of active tool. The unit is [mm/s ²].

● Output Variables

Name	Meaning	Data type	Valid range	Description
Done	Done	BOOL	TRUE or FALSE	TRUE when the instruction is completed.
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
Active	Controlling	BOOL	TRUE or FALSE	TRUE when the axes group is being controlled.
CommandAborted	Command Aborted	BOOL	TRUE or FALSE	TRUE when the instruction is aborted.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#0000 indicates normal execution.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
Done	When the specified target time is reached.	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one control period when <i>Execute</i> is FALSE.
Busy	When <i>Execute</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Done</i> changes to TRUE. When <i>Error</i> changes to TRUE. When <i>CommandAborted</i> changes to TRUE.
Active	When the axis starts moving	<ul style="list-style-type: none"> When <i>Done</i> changes to TRUE. When <i>Error</i> changes to TRUE. When <i>CommandAborted</i> changes to TRUE.
CommandAborted	<ul style="list-style-type: none"> When this instruction is aborted because another motion control instruction was executed with the Buffer Mode set to <i>Aborting</i>. When this instruction is canceled due to an error. When this instruction is executed while there is an error. When this instruction is started during MC_GroupStop instruction execution. 	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one control period when <i>Execute</i> is FALSE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	When the error is cleared.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
AxesGroup	Axes Group	_sGROUP_REF	--	Specify the axes group.

■ Function

- This instruction is used to move a robot (axes group) from a current to the target position, which is specified by the Position input variable.
- This instruction is completed when the specified target time is reached. No in-position check is performed.
- The details are described below.



Precautions for Safe Use

- After the adjustment of mechanical home of the robot, please move the robot.
 - Do not use robot at out of the workspace, without workspace check function. If you use may cause damage to the robot.
 - In Robot Ver. 1.01 or later, the velocity error (too fast) or the acceleration error (too much acceleration/deceleration) are not detected when MaxVelocity or MaxAcceleration are set to 0 or they are default. Set correct values to MaxVelocity and MaxAcceleration in order to avoid unexpected velocity and acceleration/deceleration.
 - A velocity error or an acceleration error are not detected before starting this instruction when ToolID is set other than 0.
- Specifying the Target Position
 - The target position (*Position*) is the position on a machine coordinate system (MCS) or user coordinate system (UCS). For Delta3 and Delta2, the target position is set as shown below.

Position

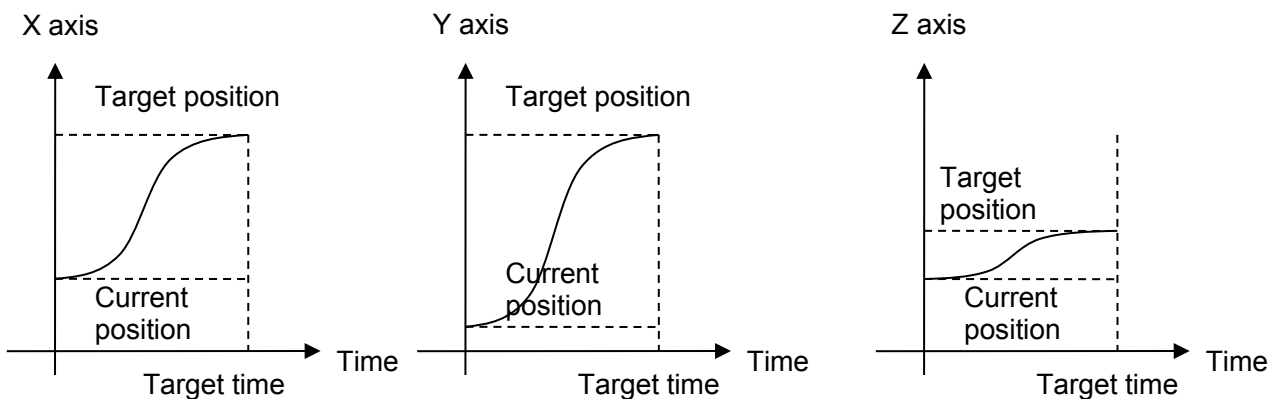
Name	Data type	Valid range	Description
Position[0]	LREAL	Negative number, positive number, or 0	Target position of X axis [mm]
Position[1]	LREAL	Negative number, positive number, or 0	Target position of Y axis [mm]
Position[2]	LREAL	Negative number, positive number, or 0	Target position of Z axis [mm]
Position[3]	LREAL	Negative number, positive number, or 0	Target position of robot rotation around X axis [deg]
Position[4]	LREAL	Negative number, positive number, or 0	Target position of robot rotation around Y axis [deg]
Position[5]	LREAL	Negative number, positive number, or 0	Target position of robot rotation around Z axis [deg]

Position array details depending on kinematics type (KinType):

Name	KinType	Position					
		[0]	[1]	[2]	[3]	[4]	[5]
Delta 3 Robot	100	O	O	O	X	X	X
Delta 3 Robot with Rotation	101	O	O	O	X	X	O
Delta 2 Robot	102	O	X	O	X	X	X

O Used; X Not Used

- When using an user coordinate system (UCS), specify its ID number in the Coordinate System ID (CSID).
- If the workspace check function is enabled and it detects an error, the axis Group is stopped. Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) section エラー! 参照元が見つかりません。 the details of workspace check function.
- Specifying the Trajectory
 - The parameters to create the trajectory up to the target position are specified by *MoveTrajType*, *TrajTime*, *MaxVelocity*, and *MaxAcceleration* of the Trajectory Data (*TrajData*).
 - Select the third-degree polynomial or the modified sine (recommended) in Movement Trajectory Type Selection. (*MoveTrajType*).



- If any calculated commanded position is outside of the workspace, when the workspace check function is enabled, an error will be raised and the axes stopped.
- When calculating a commanded position, velocity and acceleration of the movement are calculated and checked against the command maximums. If exceeding them, the following errors will be raised: Velocity Error Detection Value (MaxVelocity) and Acceleration Error Detection Value (MaxAcceleration).

- **BufferMode (Buffer Mode Selection)**

BufferMode specifies how to join the axis motions for this interpolation instruction and the previous interpolation instruction.

- There are the following six settings.

Buffer Mode Selection		Description
Aborting		Aborts the instruction being executed and switches to this instruction. If the direction of axis motion is reversed by switching instructions, the motion will be reversed according to the Operation Selection at Reversing axis parameter.
Buffered		Buffers this instruction and executes it automatically after the current instruction is completed.
Blending		Starts the buffered instruction at the velocity (transit velocity) at which the current instruction reaches the target position. The operation of the current instruction is changed so that the axes reach the target position at the transit velocity. There are four methods to specify the transit velocity. These are described below.
	Blending low	The lower of the target velocities of the current instruction and the buffered instruction is used as the transit velocity.
	Blending previous	The target velocity of the current instruction is used as the transit velocity.
	Blending next	The target velocity of the buffered instruction is used as the transit velocity.
	Blending high	The higher of the target velocities of the current instruction and the buffered instruction is used as the transit velocity.

For details on *BufferMode* (Buffer Mode Selection), refer to the *NJ-series CPU Unit Motion Control User's Manual* (Cat. No. W507).

- **TransitionMode**

- *TransitionMode* specifies how to join the motions for this interpolation instruction and the previous interpolation instruction.

- *Transition disabled, StartHeight, and StartRemainingTime* are supported for *TransitionMode*.

- If *BufferMode* (Buffer Mode Selection) is set to *Blending*, *TransitionMode* is enabled.

- An error will occur if you do not set *TransitionMode* to *_mcTMNone* (Transition Disabled) when blending is not used.

- For Start Height, The *TransitionParameter* is set as the following.

Name	Data type	Valid range	Description
TransitionParameter[0]	LREAL	Negative number, positive number, or 0	Height of the transition
TransitionParameter [1]-[7]	LREAL	0	Reserved

- For Start Remaining Time, The *TransitionParameter* is set as the following.

Name	Data type	Valid range	Description
TransitionParameter[0]	LREAL	Negative number, positive number, or 0	Duration of the transition
TransitionParameter [1]-[7]	LREAL	0	Reserved

- **Other Specifications**

- This instruction can be used when all of the following conditions are satisfied.

The kinematics must have been set by the *MC_SetKinTransform* instruction.

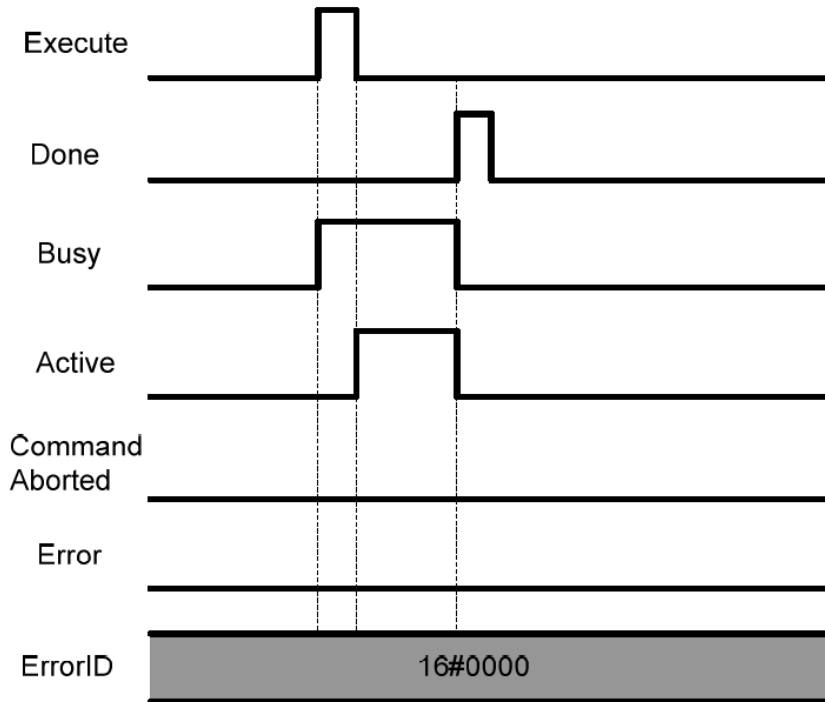
For all axes registered in the axes group, the home must have been defined.

For all axes registered in the axes group, must be turned ON.

For all axes registered in the axes group, limit inputs must be OFF.

The axes group must be enabled.

● Timing Charts



● Re-execution of Motion Control Instructions

- This instruction cannot be re-executed.

A Motion Control Instruction Re-execution Disabled error (error code: 543B hex) occurs if re-execution is attempted, and all axes that are being moved by this instruction stop.

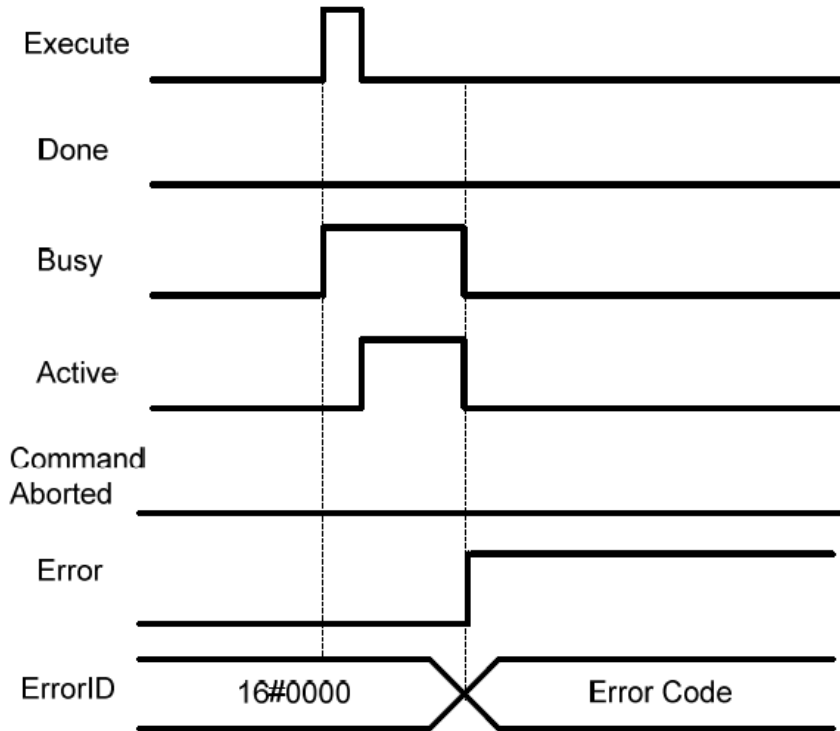
● Multi-execution of Motion Control Instructions

- A restriction applies to the instructions that can be used while this instruction is in execution.
- For details on multi-execution of motion control instructions, refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Multi-execution of Motion Control Instructions* of this document.

● Errors

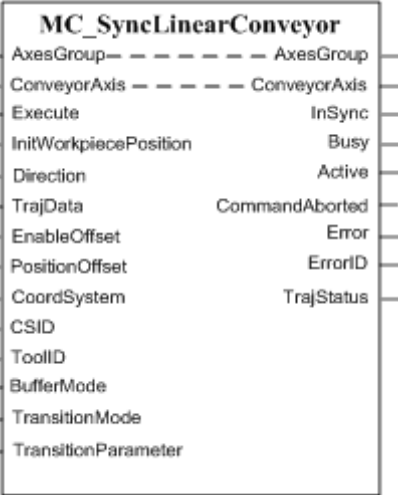
If an error occurs during instruction execution, *Error* will change to TRUE.

You can find out the cause of the error by referring to the value output by *ErrorID* (Error Code).



MC_SyncLinearConveyor <Robot Ver 1.01>

The MC_SyncLinearConveyor instruction starts creation of command values to catch up with a specified workpiece on a conveyor.

Instruction	Name	FB/FUN	Graphic expression	ST expression
MC_SyncLinearConveyor	Start Conveyor Synchronization	FB	<p>MC_SyncLinearConveyor_instance</p>  <p>The diagram shows a rectangular block titled 'MC_SyncLinearConveyor' with various input and output ports. Inputs on the left include AxesGroup, ConveyorAxis, Execute, InitWorkpiecePosition, Direction, TrajData, EnableOffset, PositionOffset, CoordSystem, CSID, ToolID, BufferMode, TransitionMode, and TransitionParameter. Outputs on the right include InSync, Busy, Active, CommandAborted, Error, ErrorID, and TrajStatus.</p>	<pre>MC_SyncLinearConveyor (AxesGroup:=variable_name, ConveyorAxis:=variable_name, Execute :=variable_name, InitWorkpiecePosition :=variable_name, Direction :=variable_name, TrajData :=variable_name, EnableOffset :=variable_name, PositionOffset :=variable_name, CoordSystem:=variable_name, CSID :=variable_name, ToolID := variable_name, BufferMode :=variable_name, TransitionMode :=variable_name, AxesGroup =>variable_name, ConveyorAxis :=variable_name, InSync =>variable_name, Busy => variable_name, Active =>variable_name, CommandAborted =>variable_name, Error =>variable_name, ErrorID =>variable_name, TrajStatus => variable_name,);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
Execute	Execute	BOOL	TRUE or FALSE	FALSE	The instruction is executed when <i>Execute</i> changes to TRUE.
InitWorkpiecePosition	Initial Workpiece Position	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	0	Specify the workpiece position (target position) at the time when the synchronization with conveyor is started. The unit is [mm] for InitWorkpiecePosition[0]-[2] and [deg] for InitWorkpiecePosition[3]-[5].
Direction	Rotation Direction	ARRAY [0..3] OF _eMC_DIRECTION	0: _mcPositiveDirection 2: _mcNegativeDirection 4: _mcNoDirection	4	Specify the direction of rotation of the rotation axis. 0: Positive direction 2: Negative direction 4: No direction
TrajData	Trajectory Data	_sMC_SYNC_TRAJ_REF	---	---	Specify the parameters for creating command current positions up to the target position.
EnableOffset	Enable Offset	BOOL	TRUE or FALSE	FALSE	The offset positions are enabled when <i>Enable</i> is TRUE.
PositionOffset	Mixed Position Offset	ARRAY[0..5] OF LREAL	Negative number, positive number, or 0	0	Specified the offset positions.
CoordSystem	Coordinate System	_eMC_RBT_COORD_SYSTEM	1: _mcRBT_MCS 3: _mcRBT_UCS	1	Specify the coordinate system of the target position. 1: Machine coordinate system (MCS) 3: User coordinate system (UCS)
CSID	Coordinate System ID	UINT	0 to 15	0	Specify the ID of coordinate system when a user coordinate system is specified.
ToolID	Robot Tool ID	UINT	0 to 16	0	Specify the selected Robot Tool to command.
BufferMode	Buffer Mode	_eMC_BUFFER_MODE	0: _mcAborting	0	Specify the behavior when

	Selection				executing more than one motion control instruction. 0: Aborting
TransitionMode	Transition Mode	_eMC_TRANSITION_MODE	0: _mcTMNone	0	Specify the path of motion. 0: Transition disabled
TransitionParameter	Transition Parameters	ARRAY [0..7] OF LREAL	Negative number, positive number, or 0	0	Set the parameters for transition. (Reserved)

SMC_SYNC_TRAJ_REF

Member variable	Meaning	Data type	Valid range	Function
SyncTrajType	Synchronization Trajectory Type Selection	_eMC_SYNC_TRAJ_TYPE	0: _mcFlatTraj 2: _mcFlatTrajModSine 3: _mcFlatTrajModConstantVel	Specify the trajectory type to create command current position for synchronization with the conveyor. 0: Third-order polynomial 2: Modified sine 3: Modified constant velocity
TrajTime	Trajectory Target Time	ARRAY [0..7] OF UINT	---	Specify the time to create command current positions from current position to target position. The unit is [ms].
TrajTransition	Trajectory Transition Ratio	ARRAY [0..7] OF UINT	---	Specify the time ratio to start the movement from current operation to next operation within this command. The unit is [%].
TrajDistance	Trajectory Distance	ARRAY [0..7] OF LREAL	Negative number, positive number, or 0	Specify the distance for upward/downward movement from current position to target position. The unit is [mm].
MaxVelocity	Velocity Error Detection Value	LREAL	Positive number or 0	Specify the velocity at which a velocity error (too fast) is detected when creating a command current position of active tool. The unit is [mm/s].
MaxAcceleration	Acceleration Error Detection Value	LREAL	Positive number or 0	Specify the acceleration rate at which an acceleration error (too much acceleration/deceleration) is detected when creating a command current position of active tool. The unit is [mm/s ²].

● Output Variables

Name	Meaning	Data type	Valid range	Description
InSync	In Sync	BOOL	TRUE or FALSE	TRUE when the TCP of robot catches up with the workpiece on the conveyor.
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
Active	Controlling	BOOL	TRUE or FALSE	TRUE when the axes group is being controlled.
CommandAborted	Command Aborted	BOOL	TRUE or FALSE	TRUE when the instruction is aborted.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#0000 indicates normal execution.
TrajStatus	Trajectory Status	_sMC_SYNC_TRAJ_STATUS	---	Outputs the status (remaining time for creating command current position and phase) of the trajectory.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

SMC_SYNC_TRAJ_STATUS

Member variable	Meaning	Data type	Valid range	Function
RemainTime	Remaining Time	UINT	Positive number or 0	Outputs the remaining time for creating command current position. The unit is [ms].
Phase	Phase	UINT	0 to 6	Outputs which position on trajectory the command position refers to.

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
InSync	If the command current position and actual current position of the workpiece are matched, after the Target Time Trajectory ($T1 + T2 + (T3 - T3 * Trans2)$) (later describe) has elapsed. If the difference between command current position(X, Y) and actual current position of the workpiece (X,Y) of each axis is as below, it is considered as match. ·When control period is 1ms : 0.1[mm] ·When control period is 2ms : 0.2[mm] ·When control period is 4ms : 0.4[mm]	<ul style="list-style-type: none"> When <i>Error</i> changes to TRUE. When <i>CommandAborted</i> changes to TRUE.
Busy	When <i>Execute</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Error</i> changes to TRUE. When <i>CommandAborted</i> changes to TRUE.
Active	When the axis starts moving.	<ul style="list-style-type: none"> When <i>Error</i> changes to TRUE. When <i>CommandAborted</i> changes to TRUE.
CommandAborted	<ul style="list-style-type: none"> When this instruction is aborted because another motion control instruction was executed with the Buffer Mode set to <i>Aborting</i>. When this instruction is canceled due to an error. When this instruction is executed while there is an error. When this instruction is started during MC_GroupStop instruction execution. 	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one period when <i>Execute</i> is FALSE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	When the error is cleared.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
AxesGroup	Axes Group	sGROUP_REF	---	Specify the axes group.
ConveyorAxis	Conveyor Axis	sAxis_REF	---	Specify the axis.

■ Function

- This instruction is used for a specified robot (axes group) to catch up with a workpiece specified by Initial Workpiece Position (*InitWorkpiecePosition*) from the current position.
- The *InSync* output variable changes to TRUE when the robot catches up with the specified workpiece (i.e. when X axis and Y axis match). Then, the synchronization is maintained.
- Normally, the MC_SyncOut (End Synchronization) instruction is used to release the synchronization.
- If processing of the primary periodic task has not been finished within one control period, there is possibility that Velocity Error or Acceleration Error occurs. Please do settings and programming so that the processing of the primary periodic task does not exceed one control period.
- At the moment of starting this instruction, maximum interpolation velocity and maximum interpolation acceleration are checked against a trajectory that is planned based on stopping workpiece. Therefore, it could happen that the velocity error or the acceleration error is detected after Phase 2, due to velocity of the conveyor.
- The details are described below.



Precautions for Safe Use

- After the adjustment of mechanical home of the robot, please move the robot.
- Do not use robot at out of the workspace, without workspace check function. If you use may cause damage to the robot.
- If you want to stop the movement of the running in the instruction MC_SyncLinearConveyor, please use the instruction MC_SyncOut.
- Due to the way the maximum interpolation velocity of this command is calculated, it could

happen that in the Phase 2 or in Phase 4 the velocity error (too fast) or the acceleration error (too much acceleration/deceleration) is detected, even if those error was not detected at the moment of starting this instruction (pre-check). This is due to the superposition of the Z profile with the XY profiles.

- In Robot Ver. 1.01 or later, the velocity error (too fast) or the acceleration error (too much acceleration/deceleration) are not detected when MaxVelocity or MaxAcceleration are set to 0 or they are default. Set correct values to MaxVelocity and MaxAcceleration in order to avoid unexpected velocity and acceleration/deceleration.

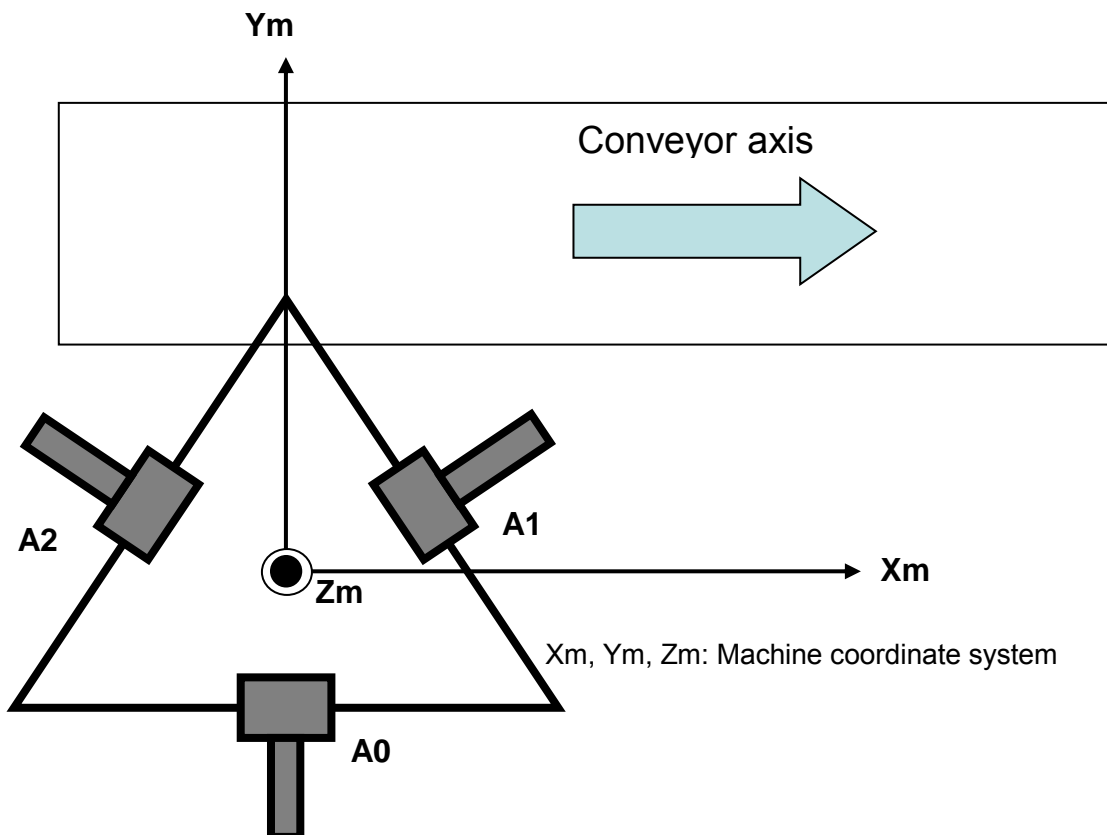
- A velocity error or an acceleration error are not detected before starting this instruction when ToolID is set other than 0.

● Setting Coordinate System and Conveyor Axis

- When catching up with a workpiece specified by Initial Workpiece Position (*InitWrokpiecePosition*), the X axis of coordinate system, which is specified by the Coordinate System (*CoordSystem*), needs to be the same direction as the conveyor axis.
- The procedure to specify the machine coordinate system (MCS) and user coordinate system (UCS) is shown below.

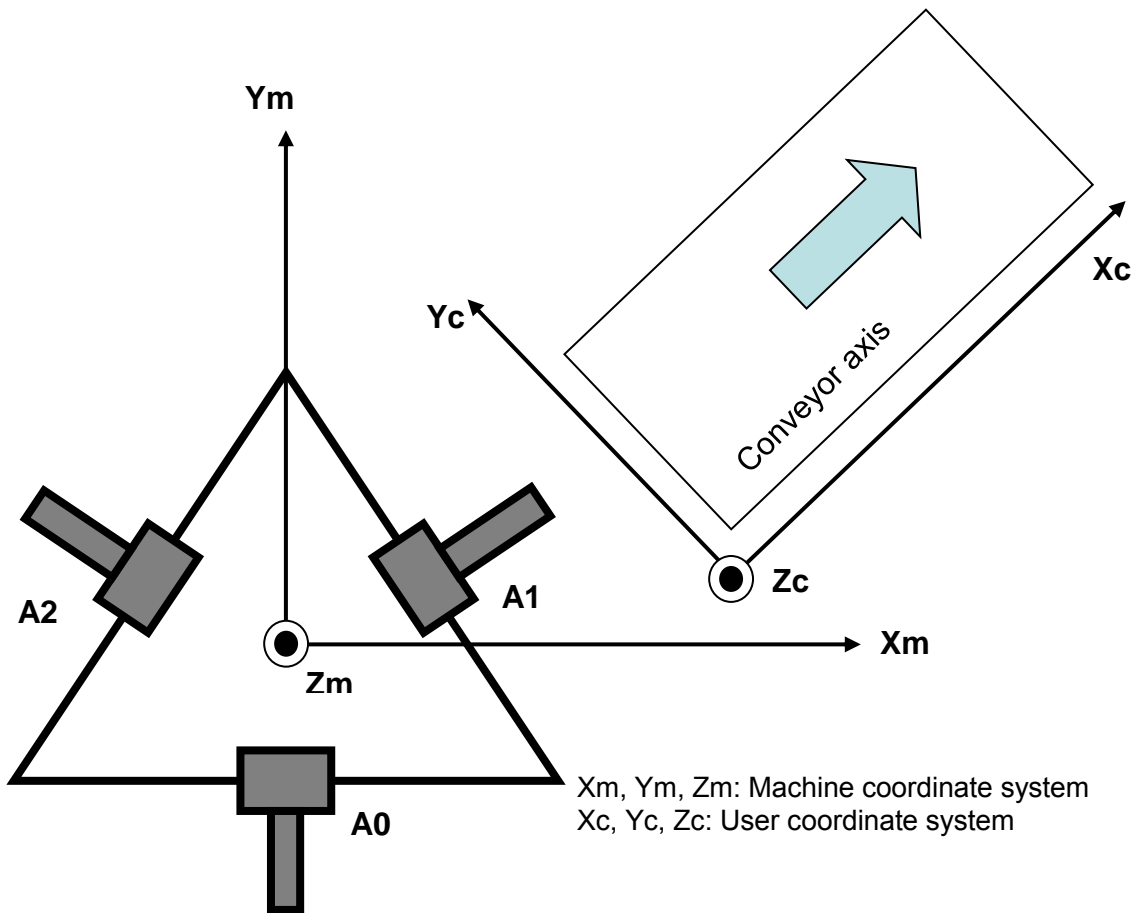
[When specifying the machine coordinate system (MCS)]

Align the directions of X_m and conveyor axis as shown in the following figure.



[When specifying the user coordinate system (UCS)]

Align the directions of Xc and conveyor axis as shown in the following figure.



• Specifying the Position of Workpiece to Catch up with

- The Initial Workpiece Position (*InitWorkpiecePosition*) of the workpiece to catch up with is a position on the machine coordinate system (MCS) or user coordinate system (UCS).

According to kinematics type (KinType), the position is specified as shown below.

InitWorkpiecePosition

Name	Data type	Valid range	Description
InitWorkpiecePosition[0]	LREAL	Negative position, positive position, or 0	Current position of X axis of the workpiece [mm]
InitWorkpiecePosition[1]	LREAL	Negative position, positive position, or 0	Current position of Y axis of the workpiece [mm]
InitWorkpiecePosition[2]	LREAL	Negative position, positive position, or 0	Current position of Z axis of the workpiece [mm]
InitWorkpiecePosition[3]	LREAL	Negative position, positive position, or 0	Current position of rotation around X axis of the workpiece [deg]
InitWorkpiecePosition[4]	LREAL	Negative position, positive position, or 0	Current position of rotation around Y axis of the workpiece [deg]
InitWorkpiecePosition[5]	LREAL	Negative position, positive position, or 0	Current position of rotation around Z axis of the workpiece [deg]

Position array details depending on kinematics type (KinType):

Name	KinType	Position					
		[0]	[1]	[2]	[3]	[4]	[5]
Delta 3	100	O	O	O	X	X	X
Delta 3R	101	O	O	O	X	X	O
Delta 2	102	O	X	O	X	X	X

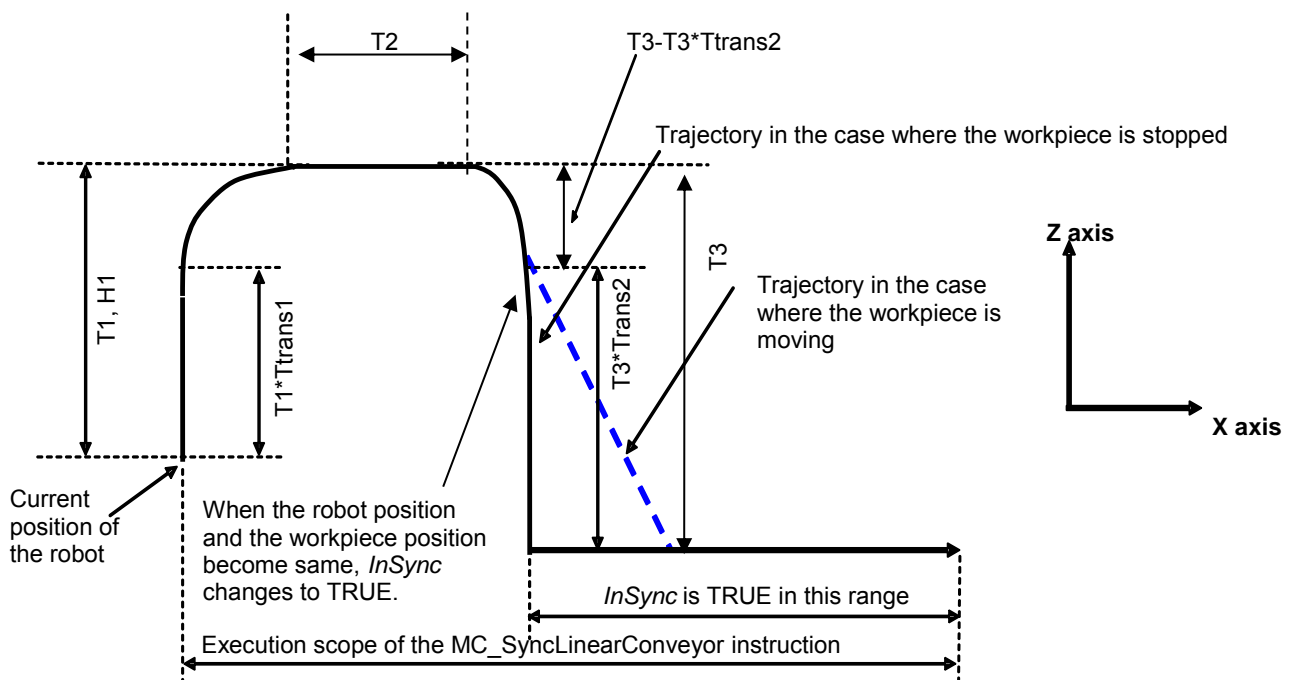
O Used; X Not Used

- When using an user coordinate system (UCS), specify its ID number in the Coordinate System ID (CSID).
- When this instruction is successfully started, the above workpiece position is updated along with the conveyor axis movement and command position is created toward the updated workpiece position.
- If the initial workpiece position is outside the workspace when the workspace function is enabled, it is considered as an error and the axes are stopped.
- When the current position is outside the workspace, the axes are also stopped.

● Specifying the Trajectory

- The parameters to create the trajectory to catch up with the workpiece are specified by *SyncTrajType*, *TrajTime*, *TrajTransition*, *TrajDistance*, *MaxVelocity*, and *MaxAcceleration* of the Trajectory Data (*TrajData*).
- Only *Flat* is supported for Synchronization Trajectory Type Selection (*SyncTrajType*) currently.
- The following figure shows the relationship between specified values and trajectory in the flat type.

[Case 1: $H1 > 0$ and $T1 > 0$]



<Terms> T1,T2,T3: Trajectory Target Time [ms]

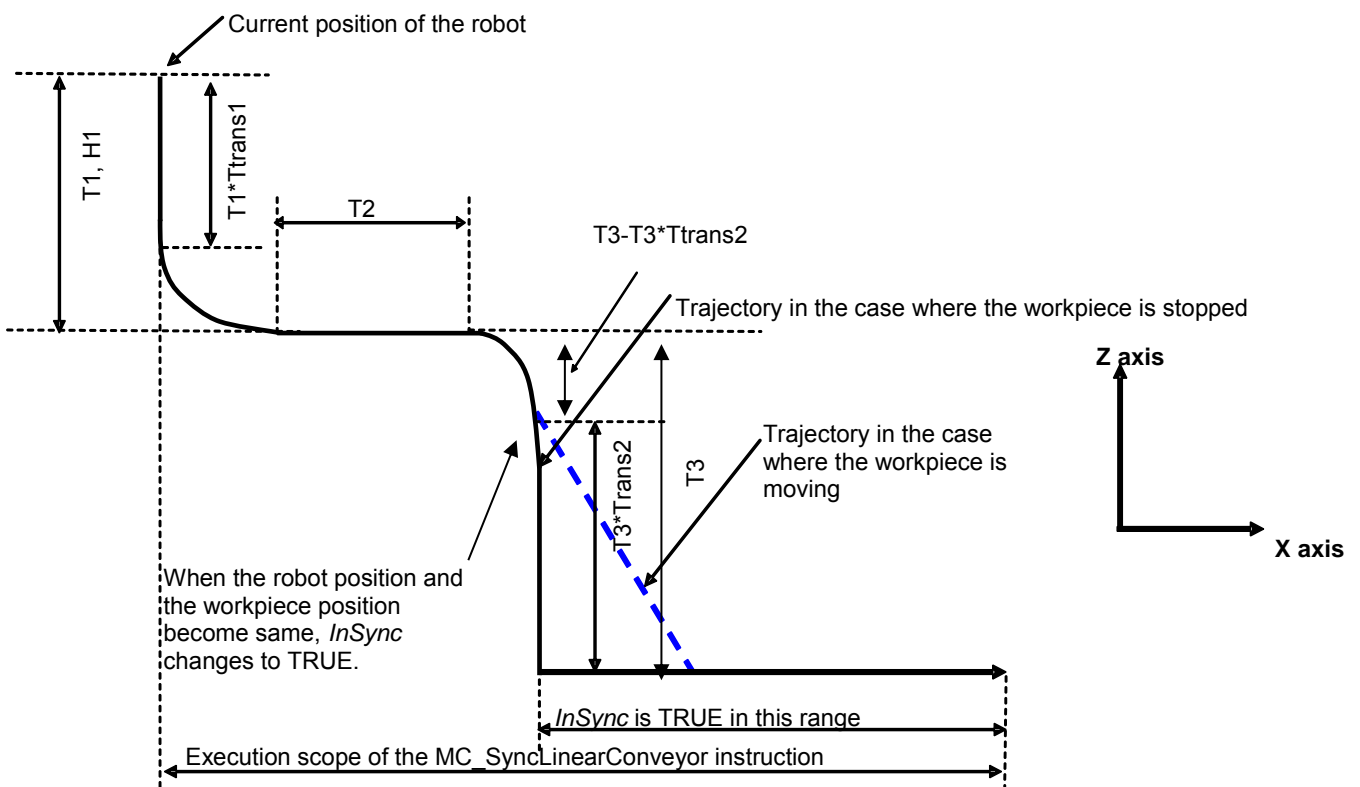
Ttrans1, Ttrans2: Trajectory Transition Ratio [%]

H1: Trajectory Distance Z-axis direction in Machine coordinate system (MCS) [mm]

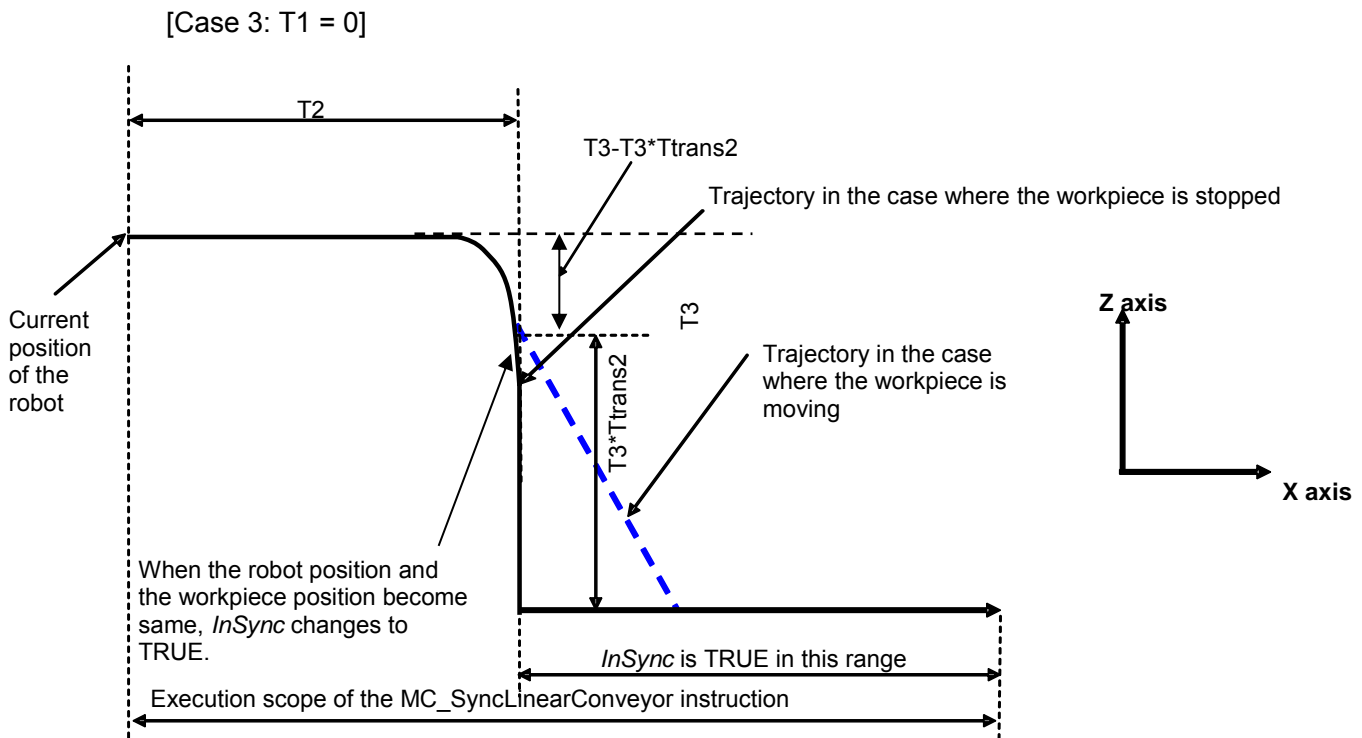
- The command position to move the robot from the current position to “current position + $H1$ ” (Z axis) in $T1$ is created.
- The command position to follow the workpiece after $T1 * Ttrans1$ has elapsed is created.
The robot does not stop at this time.
- The command position for moving the robot downward in the Z axis direction after $(T1 + T2)$ has elapsed is created. The robot does not stop at this time.
- After $(T1 + T2 + (T3 - T3 * Ttrans2))$ has elapsed, the difference between command position (X,Y) and actual current position of the workpiece (X,Y) is compared. When each difference is the following range or less, *InSync* changes to TRUE. If it is exceeded, it is considered as an error and the axes are stopped. At this time, Synchronization Disabled error (error code 6702hex) occurs.

- When control period is 1ms : 0.1[mm]
- When control period is 2ms : 0.2[mm]
- When control period is 4ms : 0.4[mm]

[Case 2: $H1 < 0$ and $T1 > 0$]



- When $H1 < 0$, the command position for moving the robot downward in the Z axis direction is created.
- In other cases, same as Case 1.



- When $T1 = 0$, initial command position for moving the robot upward/downward in the Z axis direction is not created.
- In other cases, same as Case 1.

- The following values are set for Trajectory Target Time (*TrajTime*), Trajectory Transition Ratio (*TrajTransition*), and Trajectory Distance (*TrajDistance*).

TrajTime

Name	Data type	Valid range	Description
TrajTime[0]	UINT	Positive number or 0	T1 [ms]
TrajTime[1]	UINT	Positive number	T2 [ms]
TrajTime[2]	UINT	Positive number	T3 [ms]
TrajTime[3]-[7]	UINT	0	Reserved

TrajTransition

Name	Data type	Valid range	Description
TrajTransition[0]	UINT	1 to 100	Ttrans1 [%]
TrajTransition[1]	UINT	1 to 100	Ttrans2 [%]
TrajTransition[2]	UINT	0 to 4	Rotation transition1 (Initial phase)
TrajTransition[3]	UINT	0 to 4	Rotation transition2 (Final phase)
TrajTransition[4]-[7]	UINT	0	Reserved

TrajDistance

Name	Data type	Valid range	Description
TrajDistance[0]	LREAL	Negative number, positive number, or 0	Trajectory Distance Z axis direction in Machine coordinate system (MCS) H1 [mm]
TrajDistance[1]-[7]	LREAL	0	Reserved

- If the workspace check function is enabled and it detects an error, the axis Group is stopped. Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) section エラー! 参照元が見つかりません。 the details of workspace check function.
- When creating the command position, the command velocity and command acceleration are also created that considered velocity of conveyor in every control period. And it is checked whether or not the Velocity Error Detection Value (MaxVelocity) and Acceleration Error Detection Value (MaxAcceleration) are exceeded .
- For enabling rotation of the wrist axis when using Delta3R kinematics, the TrajTransition[2] and TrajTransition[3] must be set different than zero. If both are set to zero, the rotation is disabled. If one is set to zero and another is not, an error will occur.

● Offset Positions

The function adds an offset to the robot position and orientation when it is synchronized with a conveyor. Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) Section 3.3.6.1 for details.

● BufferMode (Buffer Mode Selection)

BufferMode specifies how to join the axis motions for this interpolation instruction and the previous interpolation instruction.

- There are the following six settings.

Buffer Mode Selection		Description
Aborting		Aborts the instruction being executed and switches to this instruction. If the direction of axis motion is reversed by switching instructions, the motion will be reversed according to the Operation Selection at Reversing axis parameter.
Buffered		Buffers this instruction and executes it automatically after the current instruction is completed.
Blending		Starts the buffered instruction at the velocity (transit velocity) at which the current instruction reaches the target position. The operation of the current instruction is changed so that the axes reach the target position at the transit velocity. There are four methods to specify the transit velocity. These are described below.
	Blending low	The lower of the target velocities of the current instruction and the buffered instruction is used as the transit velocity.
	Blending previous	The target velocity of the current instruction is used as the transit velocity.
	Blending next	The target velocity of the buffered instruction is used as the transit velocity.
	Blending high	The higher of the target velocities of the current instruction and the buffered instruction is used as the transit velocity.

For details on BufferMode (Buffer Mode Selection), refer to the NJ-series CPU Unit Motion Control User's Manual (Cat. No. W507).

● TransitionMode

- TransitionMode specifies how to join the motions for this interpolation instruction and the previous interpolation instruction.
- Transition disabled, StartHeight, and StartRemainingTime are supported for TransitionMode.
- If BufferMode (Buffer Mode Selection) is set to Blending, TransitionMode is enabled.
- An error will occur if you do not set TransitionMode to _mcTMNone (Transition Disabled) when blending is not used.
- For Start Height, The TransitionParameter is set as the following.

Name	Data type	Valid range	Description
TransitionParameter[0]	LREAL	Negative number, positive number, or 0	Height of the transition
TransitionParameter [1]-[7]	LREAL	0	Reserved

- For Start Remaining Time, The *TransitionParameter* is set as the following.

Name	Data type	Valid range	Description
TransitionParameter[0]	LREAL	Negative number, positive number, or 0	Duration of the transition
TransitionParameter [1]-[7]	LREAL	0	Reserved

- Other Specifications

- This instruction can be used when all of the following conditions are satisfied.

The kinematics must have been set by the MC_SetKinTransform instruction.

For all axes registered in the axes group, the home must have been defined.

For all axes registered in the axes group, must be turned ON.

For all axes registered in the axes group, limit inputs must be OFF.

The axes group must be enabled.

● Outputting Status during Command Position Creation

-While the command position is created, each status is output to the applicable member of Trajectory Status (*_sMC_SYNC_TRAJ_STATUS*). During phase 5 and 6, robot and workpiece Synchronize and Insync will change to TRUE.

RemainTime: Shows the remaining time of Trajectory Target Time ($T1 + T2 + T3$). This is found by subtraction from $T1 + T2 + T3$.

Phase: Phases 0 to 6.

Shows which trajectory is being calculated.

0: This instruction is not being executed.

1: $t \leq T_{trans1} * T1$

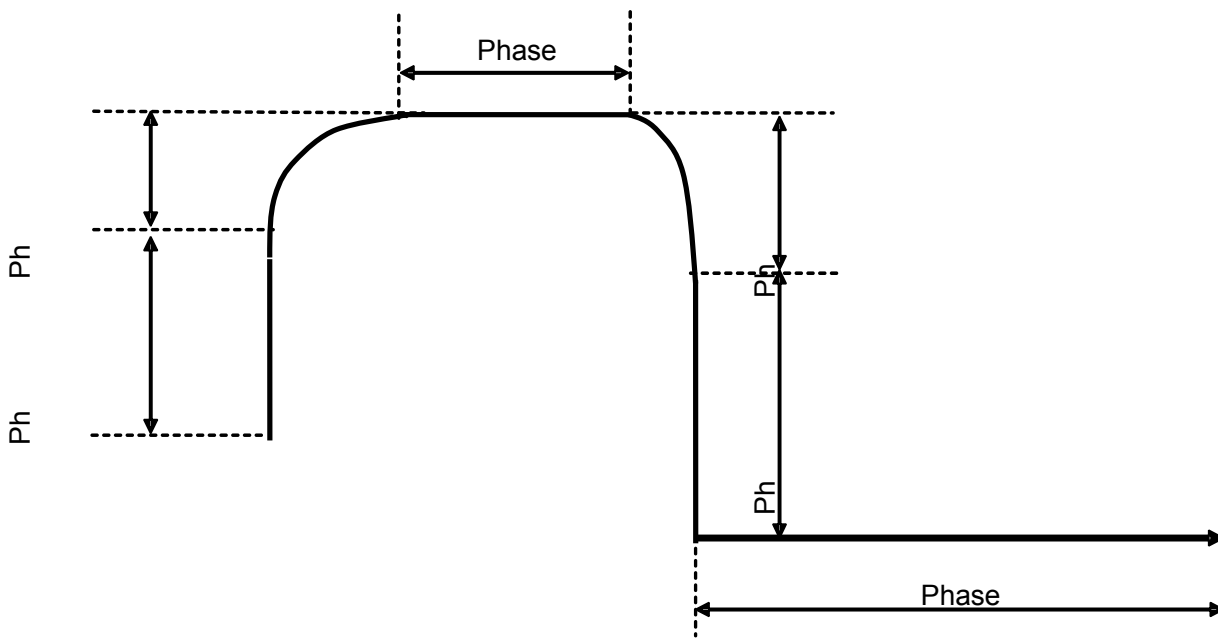
2: $T_{trans1} * T1 < t \leq T1$

3: $T1 < t \leq T1 + T2$

4: $T1 + T2 < t \leq T1 + T2 + (T3 - T3 * T_{trans2})$

5: $T1 + T2 + (T3 - T3 * T_{trans2}) < t \leq (T1 + T2 + T3)$

6: $(T1 + T2 + T3) < t$



When setting Trajectory Transition Ratio parameters (T_{trans1} and T_{trans2}) to 100%, control period of phase 2 and 4 is not to be 0. At least, One control period is secured as below.

Phase

0: This instruction is not being executed.

1: $t \leq T1$

2: $T1 < t \leq T1 + \text{One control period}$

3: $T1 + \text{One control period} < t \leq T1 + T2$

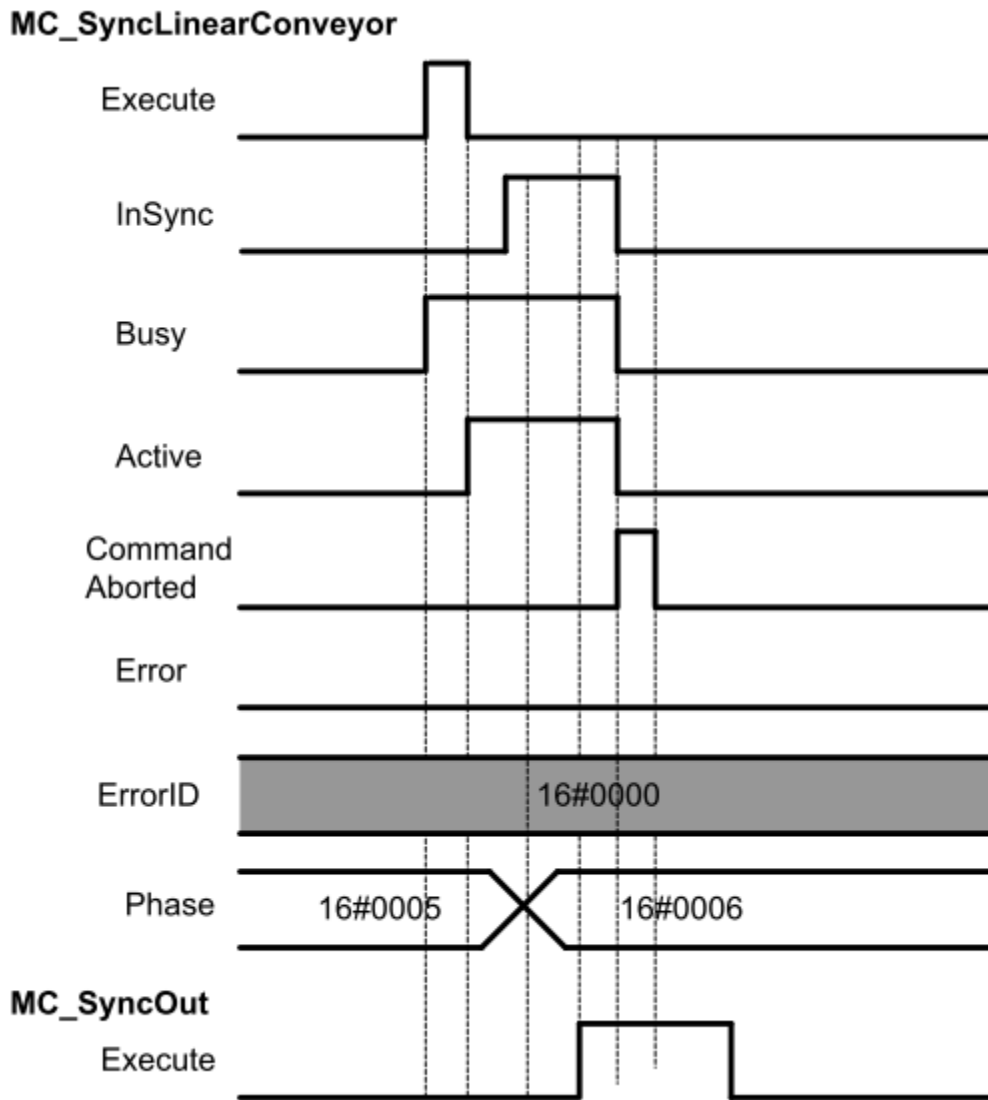
4: $T1 + T2 < t \leq T1 + T2 + \text{One control period}$

5: $T1 + T2 + \text{One control period} < t \leq T1 + T2 + T3$

6: $T1 + T2 + T3 < t$

● Timing Charts

The timing chart of this instruction is shown below. The one for stopping this instruction by the MC_SyncOut instruction is also shown below.



● Re-execution of Motion Control Instructions

- This instruction cannot be re-executed.

A Motion Control Instruction Re-execution Disabled error (error code: 543B hex) occurs if re-execution is attempted, and all axes that are being moved by this instruction stop.

● Multi-execution of Motion Control Instructions

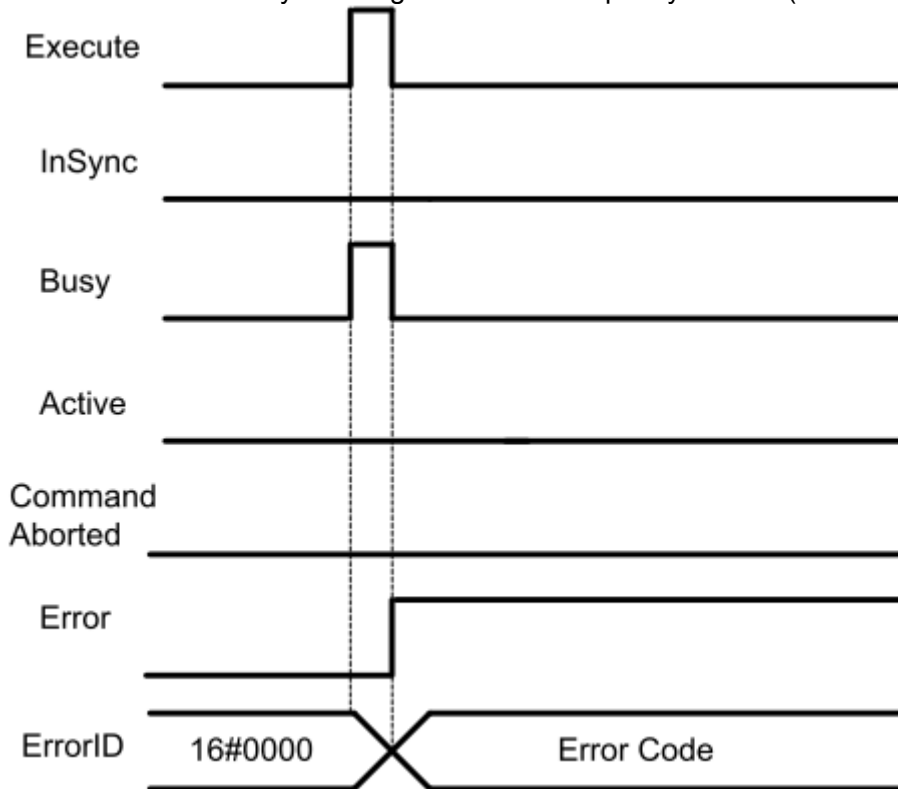
- A restriction applies to the instructions that can be used while this instruction is in execution.

- For details on multi-execution of motion control instructions, refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Multi-execution of Motion Control Instructions* of this document.

- Errors

If an error occurs during instruction execution, *Error* will change to TRUE.

You can find out the cause of the error by referring to the value output by *ErrorID* (Error Code).



MC_InverseKin <Robot Ver 1.01>

The MC_InverseKin instruction performs inverse kinematics.

Instruction	Name	FB/ FUN	Graphic expression	ST expression
MC_InverseKin	Inverse Kinematics	FB	<p style="text-align: center;">MC_InverseKin_instance</p>	<pre>MC_InverseKin(AxesGroup:=variable_name, Enable:=variable_name, Position:= variable_name, ToolID:= variable_name, AxesGroup=>variable_name, Valid=> variable_name, Busy=> variable_name, Error=> variable_name, ErrorID=> variable_name AxesPosition=> variable_name OutWorkspace=> variable_name,);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
Enable	Enable	BOOL	TRUE or FALSE	FALSE	The target position is transformed to axis target positions of each axis when <i>Enable</i> changes to TRUE.
Position	Target Position	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	0	Specify the target position on the machine coordinate system (MCS). For Delta3, the unit is [mm].
ToolID	Robot Tool ID	UINT	0 to 16	0	Specify the selected Robot Tool to command.

● Output Variables

Name	Meaning	Data type	Valid range	Description
Valid	Enabled	BOOL	TRUE or FALSE	The Axis Target Position (<i>AxesPosition</i>) below is enabled when <i>Valid</i> is TRUE. The axis target position is updated every period.
AxesPosition	Axis Target Position	ARRAY [0..5] OF LREAL	Negative number, positive number, or 0	Shows the axis target position of each axis that is found by transforming the target position on the machine coordinate system (MCS). For Delta3, the unit is [deg].
OutWorkspace	Outside Workspace	BOOL	TRUE or FALSE	TRUE when the target position is outside the workspace.
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#0000 indicates normal execution.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
Valid	When the kinematics transform is successfully completed.	<ul style="list-style-type: none"> When <i>Error</i> changes to TRUE. When <i>Enable</i> is FALSE.
Busy	When <i>Enable</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Enable</i> changes to FALSE. When <i>Error</i> changes to TRUE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	<ul style="list-style-type: none"> When the error is cleared.
OutWorkspace	When the target position is outside the workspace.	<ul style="list-style-type: none"> When the target position is inside the workspace. When <i>Enable</i> is FALSE. When <i>Error</i> changes to TRUE.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
AxesGroup	Axes Group	_sGROUP_REF	--	Specify the axes group.

■ Function

- This instruction is used to transform the target position (*Position*) on the machine coordinate system (MCS) of the robot to the target position on the axis coordinate system (ACS) of each axis.
- According to kinematics type (KinType), the settings of the specified target position (*Position*) and the transformed axis target position (*AxesPosition*) are shown below.

Position

Name	Data type	Valid range	Description
Position[0]	LREAL	Negative number, positive number, or 0	Target position of X axis on the machine coordinate system (MCS) [mm]
Position[1]	LREAL	Negative number, positive number, or 0	Target position of Y axis on the machine coordinate system (MCS) [mm]
Position[2]	LREAL	Negative number, positive number, or 0	Target position of Z axis on the machine coordinate system (MCS) [mm]
Position[3]	LREAL	Negative number, positive number, or 0	Target position of rotation around X axis on the machine coordinate system (MCS) [deg]
Position[4]	LREAL	Negative number, positive number, or 0	Target position of rotation around Y axis on the machine coordinate system (MCS) [deg]
Position[5]	LREAL	Negative number, positive number, or 0	Target position of rotation around Z axis on the machine coordinate system (MCS) [deg]

Position array details depending on kinematics type (KinType):

Name	KinType	Position					
		[0]	[1]	[2]	[3]	[4]	[5]
Delta 3	100	O	O	O	X	X	X
Delta 3R	101	O	O	O	X	X	O
Delta 2	102	O	X	O	X	X	X

O Used; X Not Used

AxesPosition

Name	Data type	Valid range	Description
AxesPosition[0]	LREAL	Negative number, positive number, or 0	Target position of A0 axis on the axis coordinate system (ACS) [deg]
AxesPosition[1]	LREAL	Negative number, positive number, or 0	Target position of A1 axis on the axis coordinate system (ACS) [deg]
AxesPosition[2]	LREAL	Negative number, positive number, or 0	Target position of A2 axis on the axis coordinate system (ACS) [deg]
AxesPosition[3]	LREAL	Negative number, positive number, or 0	Target position of A3 axis on the axis coordinate system (ACS) [deg]
AxesPosition[4]-[5]	LREAL	0	Reserved

AxesPosition array details depending on kinematics type (KinType):

Name	KinType	Axes Position					
		[0]	[1]	[2]	[3]	[4]	[5]
Delta 3	100	O	O	O	X	X	X
Delta 3R	101	O	O	O	O	X	X
Delta 2	102	O	O	X	X	X	X

O Used; X Not Used

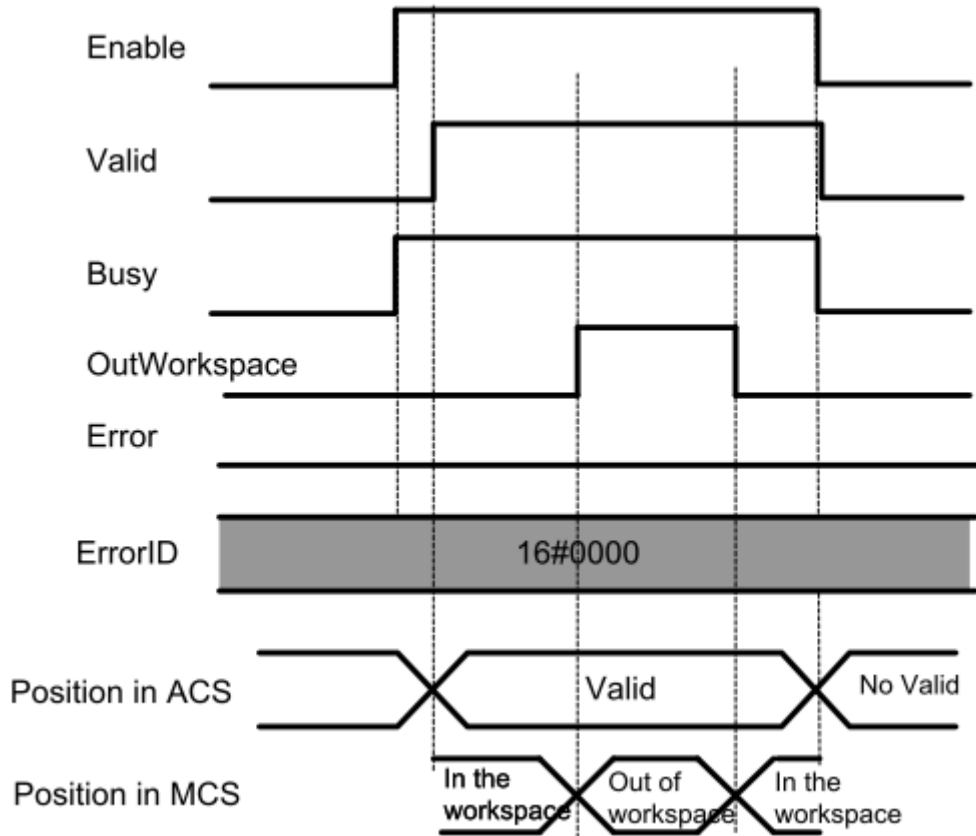
- If the target position is outside the workspace regardless of whether the workspace function is enabled or disabled, *Outside Workspace (OutWorkspace)* changes to TRUE and *Enabled (Valid)* changes to FALSE.
- If the transform target position to the target position in the Actual Coordinate System(ACS) is successfully completed at that time, the *Enabled (Valid)* changes to TRUE.

- Other Specifications

- This instruction can be used when the following condition is satisfied.

The kinematics must have been set by the MC_SetKinTransform instruction.

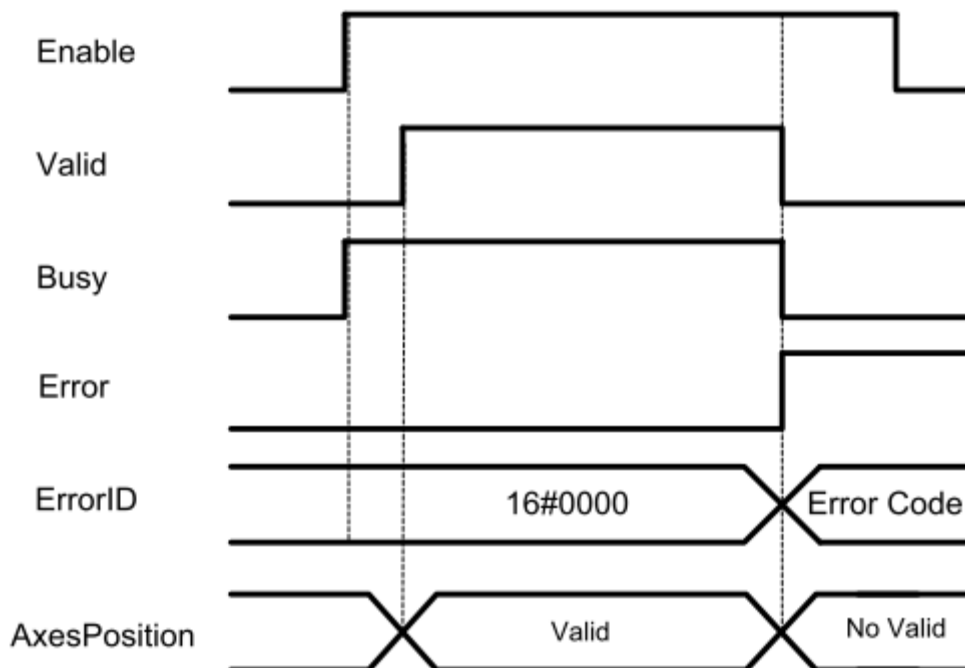
● Timing Charts



● Errors

If an error occurs during instruction execution, *Error* will change to TRUE.

You can find out the cause of the error by referring to the value output by *ErrorID* (Error Code).



MC_RobotJog <Robot Ver 1.01>

The MC_RobotJog instruction jogs a robot defined by an Axes Group according the selected target velocity and Acceleration.

Instruction	Name	FB/ FUN	Graphic expression	ST expression
MC_RobotJog	Robot Jog	FB	<p style="text-align: center;">MC_RobotJog_instance</p>	<pre>MC_RobotJog_instance (AxesGroup :=variable_name, PositiveEnable_X :=variable_name, NegativeEnable_X :=variable_name, PositiveEnable_Y :=variable_name, NegativeEnable_Y :=variable_name, PositiveEnable_Z :=variable_name, NegativeEnable_Z :=variable_name, PositiveEnable_RX :=variable_name, NegativeEnable_RX :=variable_name, PositiveEnable_RY :=variable_name, NegativeEnable_RY :=variable_name, PositiveEnable_RZ :=variable_name, NegativeEnable_RZ :=variable_name, Velocity :=variable_name, Acceleration :=variable_name, Deceleration :=variable_name, JogMode :=variable_name, CSID :=variable_name, ToolID :=variable_name, AxesGroup =>variable_name, Busy =>variable_name, CommandAborted =>variable_name, Error =>variable_name, ErrorID =>variable_name,);</pre>

■ Variables

● Input Variables

Name	Meaning	Data type	Valid range	Default	Description
PositiveEnable_X	Positive Direction Enable of X Axis	BOOL	TRUE,FALSE	FALSE	When this variable changes to TRUE, the axis starts moving in the positive direction of X axis. When it changes to FALSE, the axis stops moving.
NegativeEnable_X	Negative Direction Enable of X Axis	BOOL	TRUE,FALSE	FALSE	When this variable changes to TRUE, the axis starts moving in the negative direction of X axis. When it changes to FALSE, the axis stops moving.
PositiveEnable_Y	Positive Direction Enable of Y Axis	BOOL	TRUE,FALSE	FALSE	When this variable changes to TRUE, the axis starts moving in the positive direction of Y axis. When it changes to FALSE, the axis stops moving.
NegativeEnable_Y	Negative Direction Enable of Y Axis	BOOL	TRUE,FALSE	FALSE	When this variable changes to TRUE, the axis starts moving in the negative direction of Y axis. When it changes to FALSE, the axis stops moving.
PositiveEnable_Z	Positive Direction Enable of Z Axis	BOOL	TRUE,FALSE	FALSE	When this variable changes to TRUE, the axis starts moving in the positive direction of Z axis. When it changes to FALSE, the axis stops moving.
NegativeEnable_Z	Negative Direction Enable of Z Axis	BOOL	TRUE,FALSE	FALSE	When this variable changes to TRUE, the axis starts moving in the negative direction of Z axis. When it changes to FALSE, the axis stops moving.
PositiveEnable_RX	Positive Direction Enable of Rotation around X Axis	BOOL	TRUE,FALSE	FALSE	When this variable changes to TRUE, the axis starts moving in the positive direction of rotation around X axis. When it changes to FALSE, the axis stops moving.
NegativeEnable_RX	Negative Direction Enable of Rotation Around X Axis	BOOL	TRUE,FALSE	FALSE	When this variable changes to TRUE, the axis starts moving in the negative direction of rotation around X axis. When it changes to FALSE, the axis stops moving.
PositiveEnable_RY	Positive Direction	BOOL	TRUE,FALSE	FALSE	When this variable changes to TRUE,

	Enable of Rotation around X Axis				the axis starts moving in the positive direction of rotation around Y axis. When it changes to FALSE, the axis stops moving.
NegativeEnable_R Y	Negative Direction Enable of Rotation Around X Axis	BOOL	TRUE,FALSE	FALSE	When this variable changes to TRUE, the axis starts moving in the negative direction of rotation around Y axis. When it changes to FALSE, the axis stops moving.
PositiveEnable_RZ	Positive Direction Enable of Rotation around X Axis	BOOL	TRUE,FALSE	FALSE	When this variable changes to TRUE, the axis starts moving in the positive direction of rotation around Z axis. When it changes to FALSE, the axis stops moving.
NegativeEnable_R Z	Negative Direction Enable of Rotation Around X Axis	BOOL	TRUE,FALSE	FALSE	When this variable changes to TRUE, the axis starts moving in the negative direction of rotation around Z axis. When it changes to FALSE, the axis stops moving.
Velocity	Target Velocity	ARRAY [0..1] OF LREAL	Negative number, positive number, or 0	0	Specify the target velocity. The unit is command units/s.
Acceleration	Acceleration Rate	ARRAY [0..1] OF LREAL	Negative number, positive number, or 0	0	Specify the acceleration rate. The unit is command units/s ² .
Deceleration	Deceleration Rate	ARRAY [0..1] OF LREAL	Negative number, positive number, or 0	0	Specify the deceleration rate. The unit is command units/s ² .
JogMode	Jog Mode	_eMC_JOG_MODE	1: _mcMachineMode 2: _mcUserMode 3: _mcToolMode	1	Specify the jog mode. 1: Selected Tool moved in MCS. 2: Selected Tool moved in selected UCS. 3: Selected Tool moved in its TCS
CSID	Coordinate System ID	UINT	0 to 15	0	Specify the ID of coordinate system when a user coordinate system is specified.
ToolID	Robot Tool ID	UINT	0 to 16	0	Specify the selected Robot Tool to command.

● Output Variables

Name	Meaning	Data type	Valid range	Description
Busy	Executing	BOOL	TRUE or FALSE	TRUE when the instruction is acknowledged.
CommandAborted	Command Aborted	BOOL	TRUE or FALSE	TRUE when the instruction is aborted.
Error	Error	BOOL	TRUE or FALSE	TRUE while there is an error.
ErrorID	Error Code	WORD		Contains the error code when an error occurs. A value of 16#0000 indicates normal execution.

Refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Chapter 9 Troubleshooting*.

Output Variable Update Timing

Name	Timing for changing to TRUE	Timing for changing to FALSE
Busy	When <i>Enable</i> changes to TRUE.	<ul style="list-style-type: none"> When <i>Enable</i> changes to FALSE. When <i>Error</i> changes to TRUE.
CommandAborted	<ul style="list-style-type: none"> When this instruction is aborted because another motion control instruction was executed with the Buffer Mode set to <i>Aborting</i>. When this instruction is canceled due to an error. When this instruction is executed while there is an error. When this instruction is started during MC_GroupStop instruction execution. 	<ul style="list-style-type: none"> When <i>Execute</i> is TRUE and changes to FALSE. After one control period when <i>Execute</i> is FALSE.
Error	When there is an error in the execution conditions or input parameters for the instruction.	When the error is cleared.

● In-Out Variables

Name	Meaning	Data type	Valid range	Description
AxesGroup	Axes Group	_sGROUP_REF	--	Specify the axes group.

■ Function

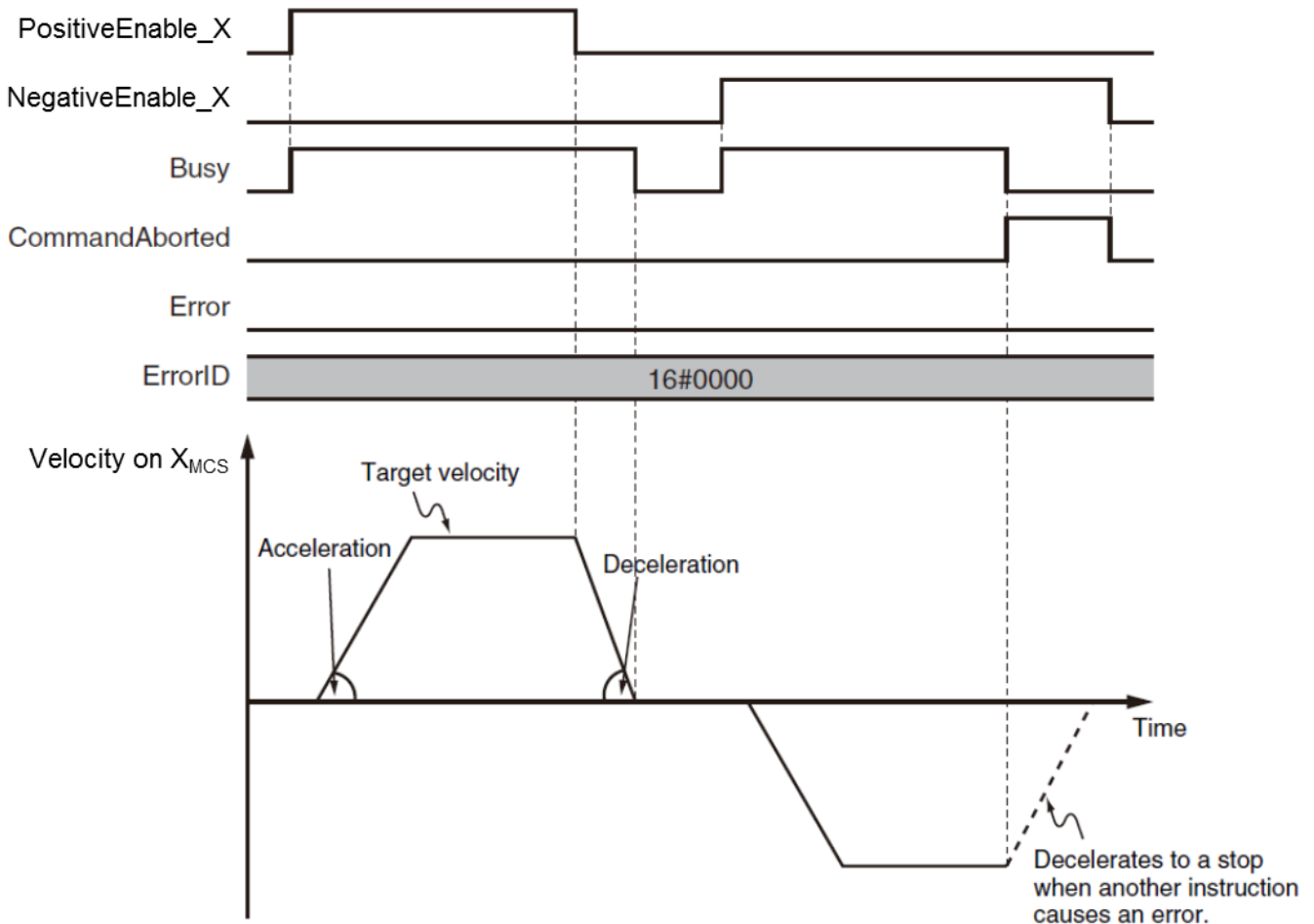
- The MC_RobotJog instruction performs jogging of the AxesGroup with kinematics attached according to the specified *Velocity* (Target Velocity).
- To jog in the positive direction of each axis, change PositiveEnable (Positive Direction Enable) of each axis to TRUE. To jog in the negative direction of each axis, change NegativeEnable (Negative Direction Enable) of each axis to TRUE.
- If PositiveEnable (Positive Direction Enable) and NegativeEnable (Negative Direction Enable) of same axis are changed to TRUE at the same time, PositiveEnable (Positive Direction Enable) of selected axis takes priority. As a result, that axis will jog in the positive direction.
- Robot can be moved in the required direction along the Cartesian coordinates X, Y or Z and around the Cartesian coordinates; Rx, Ry or Rz.
- The point of the robot to be commanded and the coordinate system to base the movement is defined by the Jog Mode. Detailed information of the robot jog modes is in 3.3.9
- The jogging is performed according the specified velocity, acceleration and deceleration. Trajectory follows a trapezoidal profile for each component with the specified acceleration, target velocity and deceleration.
- Each Enable inputs details depending on kinematics type (KinType):

Name	KinType	Positive/NegativeEnable					
		X	Y	Z	RX	RY	RZ
Delta 3 Robot	100	O	O	O	X	X	X
Delta 3 Robot with Rotation	101	O	O	O	X	X	O
Delta 2 Robot	102	O	X	O	X	X	X

O Used; X Not Used

- The jogging is performed according the specified velocity, acceleration and deceleration. Trajectory follows a
- Due to a kinematics ambiguity, if the axis position (axis belongs to commanded AxesGroup) is incremented 180° or more in 1 motion cycle, the algorithm cannot calculate correct position. In this case kinematics calculation error will occur and MC_RobotJog instruction will be aborted.

● Timing Charts



● Re-execution of Motion Control Instructions

- Re-execution with Enable in the same direction

Valid for each component (X, Y, Z, Rx, Ry, Rz):

If you change PositiveEnable (Positive Direction Enable) or NegativeEnable (Negative Direction Enable) to TRUE when it is FALSE and the robot is decelerating, the robot will begin to accelerate towards the target velocity. If you change the Velocity (Target Velocity), Acceleration (Acceleration Rate), or Deceleration (Deceleration Rate) at this time, the new value of the input parameter is used in operation.

The robot is not stopped, and Busy (Executing) does not change to FALSE.

- Re-execution with Enable in the same direction

Valid for each component (X, Y, Z, Rx, Ry, Rz):

If you change NegativeEnable (Negative Direction Enable) to TRUE when PositiveEnable (Positive Direction Enable) is TRUE and the robot is jogging in the positive direction, the robot will reverse its direction and start jogging in the negative direction for the selected component. When this happens, you can jog the robot with the input variables for when NegativeEnable (Negative Direction Enable) changes to TRUE. The input variables are Velocity (Target Velocity), Acceleration (Acceleration Rate), and Deceleration (Deceleration Rate).

The deceleration rate before the robot direction is reversed and the acceleration rate after it is reversed follow the input variables for when NegativeEnable (Negative Direction Enabled) changes to TRUE. When

NegativeEnable (Negative Direction Enable) is TRUE and the robot is jogging in the negative direction, the same operation occurs when PositiveEnable (Positive Direction Enable) changes to TRUE.

If NegativeEnable (Negative Direction Enable) changes to TRUE while PositiveEnable (Positive Direction Enable) is TRUE, the robot starts jogging in the negative direction. In this case, the robot will not jog in the positive direction even if NegativeEnable (Negative Direction Enable) changes to FALSE.

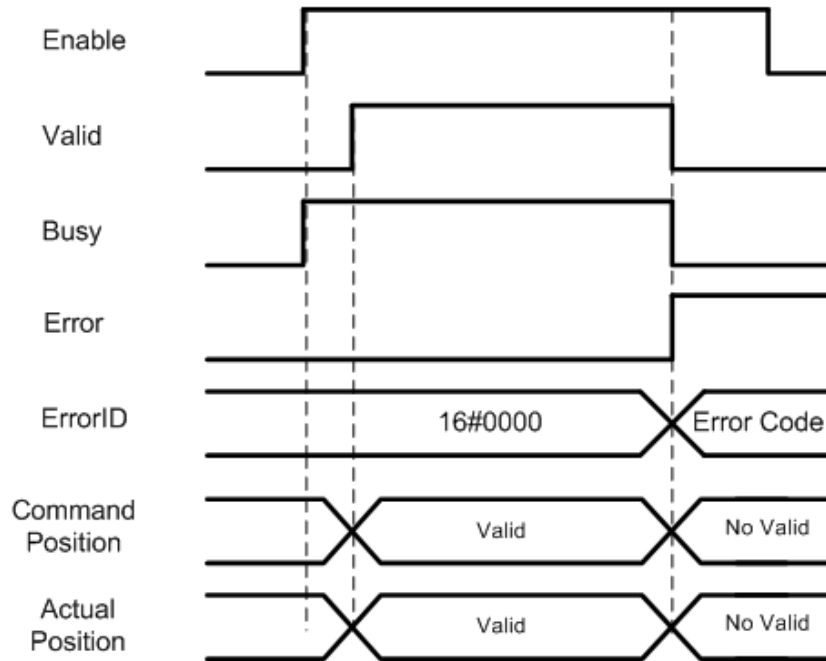
To jog the robot in the positive direction, change PositiveEnable (Positive Direction Enable) to FALSE, and then back to TRUE again. The same operation applies to the opposite case.

● Multi-execution of Motion Control Instructions

- A restriction applies to the instructions that can be used while this instruction is in execution.
- For details on multi-execution of motion control instructions, refer to Robot-supported NJ Series CPU Unit User's Manual (70017-0302347) *Multi-execution of Motion Control Instructions* of this document.

● Errors

If an error occurs during instruction execution, *Error* will change to TRUE. You can find out the cause of the error by referring to the value output by *ErrorID*.



Multi-execution of Motion Control Instructions

Some robot instructions can be executed during execution of another robot instruction, but some robot instructions cannot.

The following table shows whether or not each instruction can be executed during execution of another robot instruction.

MC_GroupSyncMoveAbsolute instruction can be also used to the robot.

■ Aborting

Instruction in execution Instruction to execute	MC_Move Time Absolute	MC_Sync Linear Conveyor (Phase < 6)	MC_Sync Linear Conveyor (Phase = 6)	MC_SyncOut (SyncStop Type=0)	MC_Group SyncMove Absolute	MC_Robot Jog
MC_MoveTimeAbsolute (Aborting selected in BufferMode)	OK1	OK1	OK1	OK1	OK1	OK1
MC_SyncLinear Conveyor (Aborting selected in BufferMode)	OK1	OK1	OK1	OK1	OK1	OK1
MC_SyncOut (SyncStopType =0)	Error	Error	OK	Error	Error	Error
MC_SyncOut (SyncStopType =1)	Error	OK	OK	Error	Error	Error
MC_GroupSyncMove Absolute (Aborting selected in BufferMode)	OK	OK	OK	OK	OK	OK
MC_RobotJog	Error	Error	Error	Error	Error	Error

Description Error: When the instruction is executed, an error is detected and the axes are stopped.

OK: The instruction can be executed. The instruction in execution is aborted.

OK1: The instruction can be executed. The instruction in execution is aborted.

Please note that the command actual position is calculated assuming initial velocity equal to 0.

■ Buffered

Instruction to execute \ Instruction in execution	MC_Move Time Absolute	MC_Sync Linear Conveyor (Phase < 6)	MC_Sync Linear Conveyor (Phase = 6)	MC_SyncOut (SyncStop Type=0)	MC_Group SyncMove Absolute	MC_Robot Jog
MC_MoveTimeAbsolute (Buffered selected in BufferMode)	OK	Error	Error	OK	Error	Error
MC_SyncLinear Conveyor (Buffered selected in BufferMode)	OK	Error	Error	OK	Error	Error

Description Error: When the instruction is executed, an error is detected and the axes are stopped.

OK: The instruction can be executed.

Note) other robot instructions can not be selected Buffered in BufferMode input variable.

■ Blending

Instruction to execute \ Instruction in execution	MC_Move Time Absolute	MC_Sync Linear Conveyor (Phase < 6)	MC_Sync Linear Conveyor (Phase = 6)	MC_SyncOut (SyncStop Type=0)	MC_Group SyncMove Absolute	MC_Robot Jog
MC_MoveTimeAbsolute (Blending selected in BufferMode)	OK1	Error	Error	OK2	Error	Error
MC_SyncLinear Conveyor (Blending selected in BufferMode)	OK1	Error	Error	OK2	Error	Error

Description Error: When the instruction is executed, an error is detected and the axes are stopped.

OK1: The instruction can be executed, only when the transition mode is Transition with given start remain time.

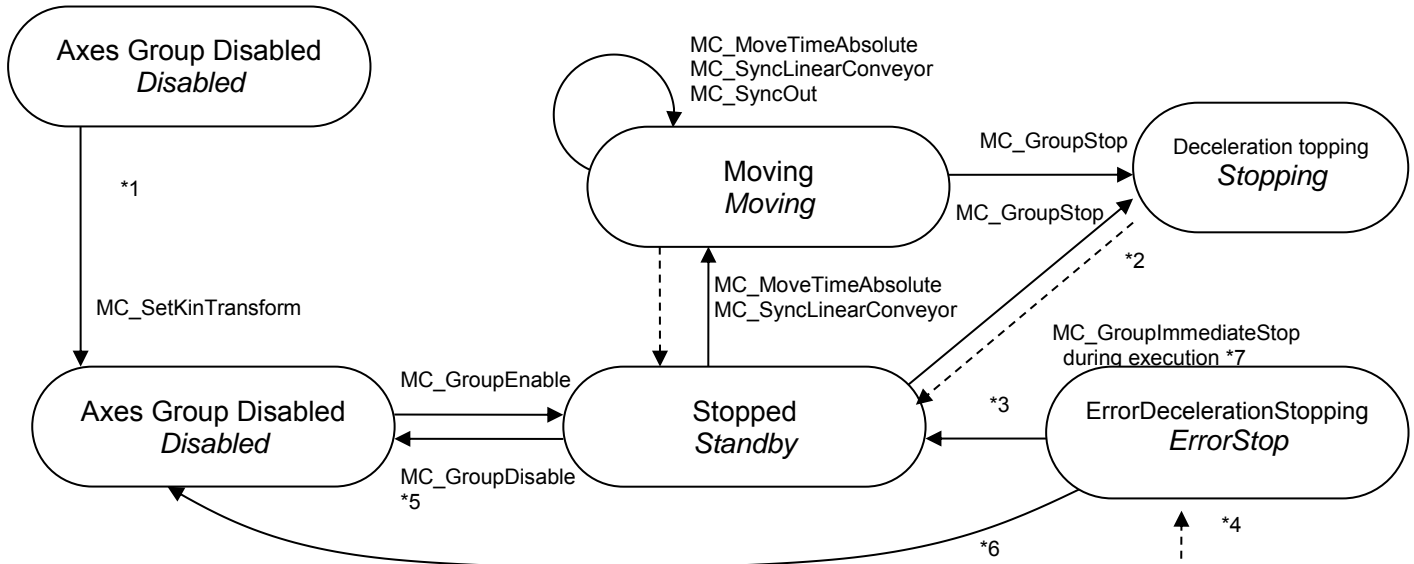
OK2: The instruction can be executed, when the transition mode is Transition with given start remain time or Transition with given start height.

Note) other robot instructions can not be selected Blending in BufferMode input variable.

State Transition of Robot Instructions

This section describes the axes group status (`_MC_GRP[0-31].Status`) at the time when each robot instruction is executed.

The other statuses not mentioned below (such as transition at the time of stopping by the `MC_Stop` instruction) are the same as those of NJ501-1*00.



- *1 Set the kinematics transformation to Axis group. After this procedure, robot instruction can be used.
- *2 Stopping state is entered when the Done output variable from the `MC_GroupStop` instruction is TRUE and the Execute input variable to the same instruction is FALSE.
- *3 The Standby state is entered when the `MC_GroupReset` or `ResetMCErr` Instruction is executed for an enabled axes group.
- *4 The ErrorStop state is entered from any other state. It is even entered if an error occurs when the axes group is disabled.
- *5 The same state is returned to if the `MC_GroupDisable` is executed in ErrorStop state.
- *6 The Disabled state is entered when the `MC_GroupReset` or `ResetMCErr` instruction is executed for a disabled axes group.
- *7 The ErrorStop state is entered from any other all states. When `MC_GroupStop` instruction is executed and during Error Deceleration Stopping.

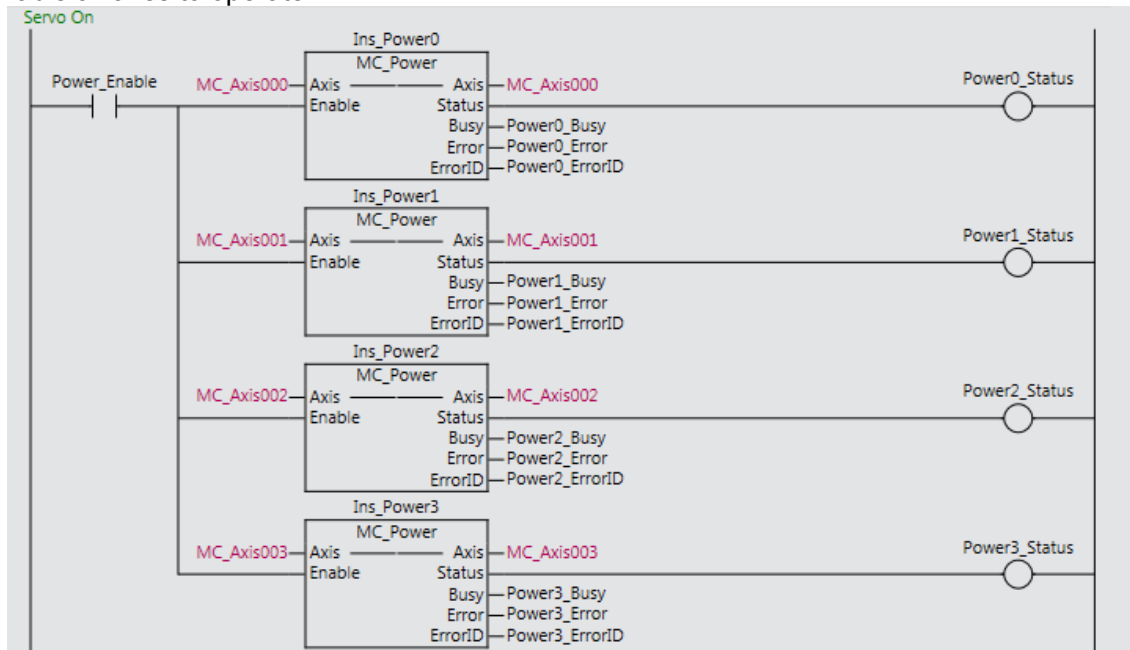
8 Sample Programming

This section shows sample programming for conveyor tracking of a 5-axis machine that has Delta3R and one conveyor axis.

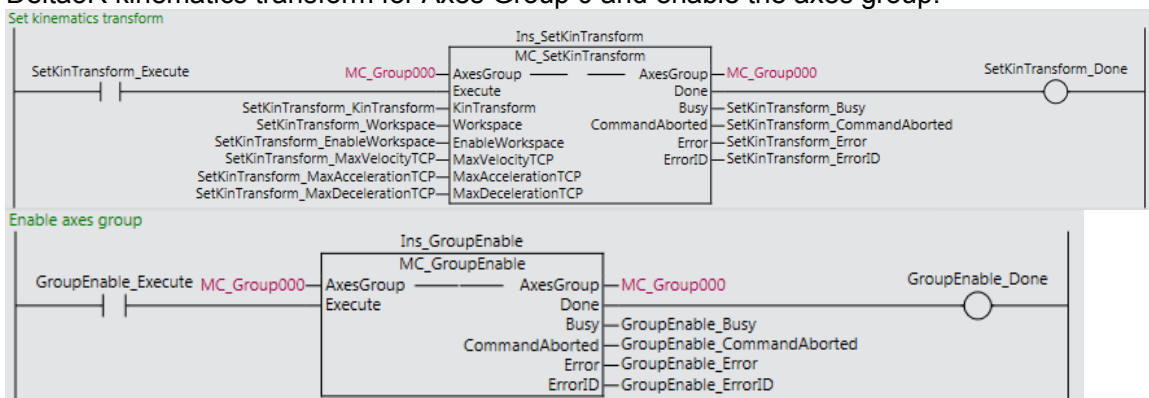
It is assumed that the axis and axes group settings have been completed with Sysmac Studio.

Delta3R	Axes group	MC_Group000
	Logical axes	A0 = MC_Axis000, A1 = MC_Axis001, A2 = MC_Axis002, A3=MC_Axis003
Conveyor axis	MC_Axis004	

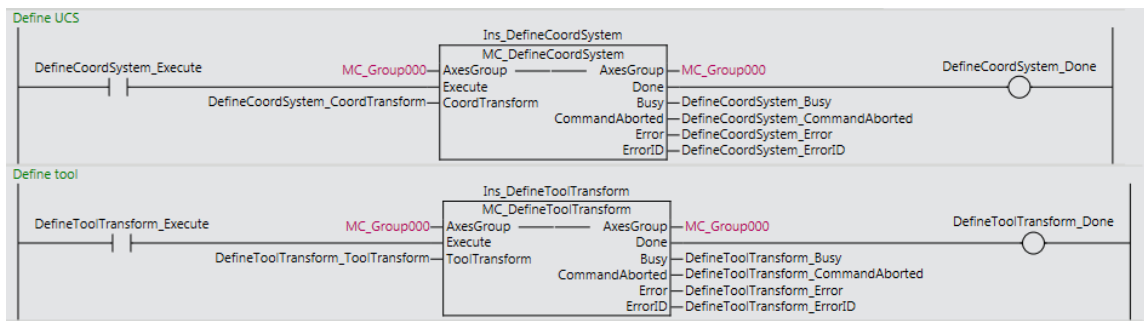
1) Enable all axes to operate.



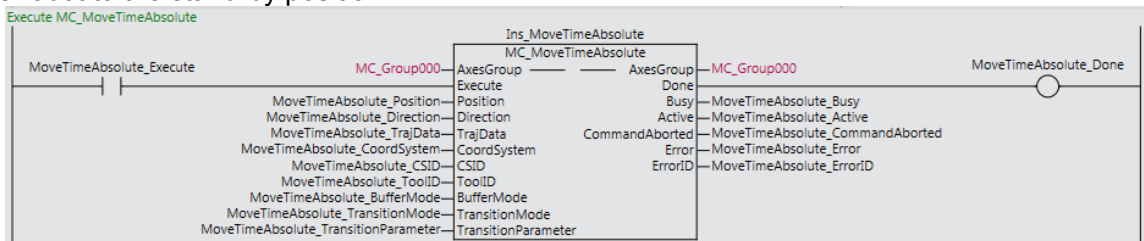
2) Set the Delta3R kinematics transform for Axes Group 0 and enable the axes group.



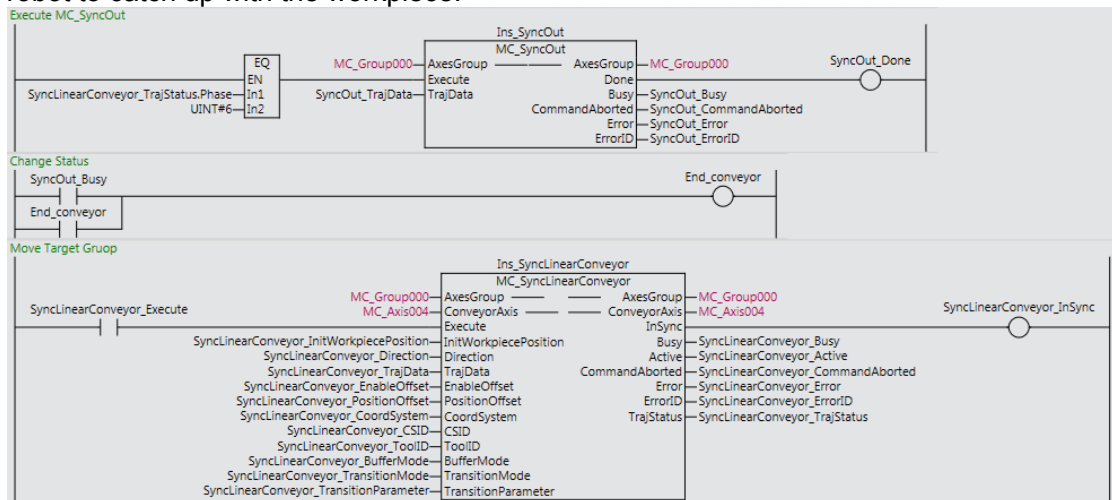
3) Define Robot tools and User Coordinate Systems (if necessary).



4) Move the robot to the stand-by position.



5) Make the robot to catch up with the workpiece.



9 Troubleshooting

9.1 Error Table

This section provides a list of event codes that may occur when a robot instruction is executed or during execution of robot instructions.

The event codes marked with ★ are the new event codes for NJ501-4*00.

The event codes for NJ501-1*00 also occur. When robot instructions can be a new cause of these event codes, items marked with ★ are added in the Meaning and Assumed Cause Columns.

Refer to the manual of NJ501-1*00 for the event codes not directly related to robot instructions (Example: Target Position Positive Software Limit Exceeded for axes).

The upper four digits of the event codes shown in the following table are output as *ErrID* (Error Code).

Event code	Event name	Meaning	Assumed cause	Level				
				Major	Partial	Minor	Observation	Information
542B0000 hex	Buffer Mode Selection Out of Range	The parameter specified for the <i>BufferMode</i> input variable to a motion control instruction is out of range.	<ul style="list-style-type: none"> Instruction input parameter exceeded the valid range of the input variable. 			√		
542C0000 hex	Coordinate System Selection Out of Range	The parameter specified for the <i>CoordSystem</i> input variable to a motion control instruction is out of range.	<ul style="list-style-type: none"> Instruction input parameter exceeded the setting range of the input variable. 			√		
54320000 hex	Transition Mode Selection Out of Range	The parameter specified for the <i>TransitionMode</i> input variable to a motion control instruction is out of range.	<ul style="list-style-type: none"> Instruction input parameter exceeded the valid range of the input variable. (Except for robot instructions) <i>_mcAborting</i> or <i>_mcBuffered</i> was specified for <i>BufferMode</i> and <i>_mcTMCornerSuperimpose</i> was specified for <i>TransitionMode</i>. 			√		
543B0000 hex	Motion Control Instruction Re-execution Disabled	An attempt was made to re-execute a motion control instruction that cannot be re-executed.	<ul style="list-style-type: none"> A motion control instruction that cannot be re-executed was re-executed. 			√		
543E0000 hex	Instruction cannot Be Executed during Multi-axes Coordinated Control	<ul style="list-style-type: none"> An operation instruction was executed for an axis or an axes group that was in a coordinated multi-axes motion. ★An operation instruction was executed while Axes Group was enabled. 	<ul style="list-style-type: none"> An operation instruction was executed for an axis or an axes group that was in a coordinated multi-axes motion. ★Execution of <i>MC_SetKinTransform</i> while axes group was enabled. 			√		
543F0000 hex	Multi-axes Coordinated Control Instruction Executed for Disabled Axes Group	A multi-axes coordinated control instruction was executed for an axes group that was in the Axes Group Disabled state.	<ul style="list-style-type: none"> A multi-axes coordinated control instruction was executed for an axes group that was in the Axes Group Disabled state. ★Any one of the following instructions was executed for an axes group that was in the Axes Group Disabled state. <i>MC_MoveTimeAbsolute</i> <i>MC_SyncLinearConveyor</i> <i>MC_SyncOut</i> 			√		
54410000 hex	Impossible Axis Operation Specified when the Servo is OFF	An operation instruction was executed for an axis for which the Servo is OFF.	<ul style="list-style-type: none"> An operation instruction was executed for an axis for which the Servo is OFF. Home was preset with the <i>MC_Home</i> instruction for an axis for which EtherCAT process data communications are not established. 			√		

Event code	Event name	Meaning	Assumed cause	Level			
				Major	Partial	Minor	Information
54610000 hex	Illegal Axes Group Specification	The axes group specified for the <i>AxesGroup</i> input variable to a motion control instruction does not exist or is not a used group.	<ul style="list-style-type: none"> An axes group does not exist for the variable specified for the <i>AxesGroup</i> input variable to the instruction. The axes group specified for the <i>AxesGroup</i> input variable to the instruction is not specified as a used group. 			√	
54660000 hex	Instruction Execution Error with Undefined Home	High-speed homing or an interpolation instruction was executed when home was undefined.	<ul style="list-style-type: none"> High-speed homing was executed when home was undefined. An interpolation instruction was executed for an axes group that includes an axis with no defined home. 				
★57050000 hex	Instruction Not Supporting Kinematics Transform	An instruction that cannot be used for the axes group to which kinematics transform is set was executed.	<ul style="list-style-type: none"> Any one of the following instructions was executed. MC_MoveLinear MC_MoveLinearAbsolute MC_MoveLinearRelative MC_MoveCircular2D MC_ChangeAxesInGroup 			√	
★57060000 hex	Different Composition between Axes Group and Kinematics	The composition is different between the specified axes group and the specified kinematics.	<ul style="list-style-type: none"> The number of axes in the axes group is different from the number of axes necessary for the specified robot (kinematics type). The count mode of an axis registered in the axes group is not linear mode. (For Delta3) The unit is not "degree". 			√	
★57070000 hex	Kinematics Type Selection Out of Range	The kinematics type specified for the <i>KinTransform</i> input variable to a motion control instruction is out of the setting range.	<ul style="list-style-type: none"> The kinematics type is out of the setting range. 			√	
★57080000 hex	Kinematics Parameter Setting Out of Range	The kinematics parameter specified for the <i>KinTransform</i> input variable to a motion control instruction is out of the setting range.	<ul style="list-style-type: none"> A kinematics parameter is out of setting range. An expansion parameter is out of the setting range. 			√	
★57090000 hex	Workspace Type Selection Out of Range	The workspace type specified for the <i>Workspace</i> input variable to a motion control instruction is out of the setting range.	<ul style="list-style-type: none"> The workspace type is out of the setting range. 			√	
★570A0000 hex	Workspace Parameter Setting Out of Range	The workspace parameter specified for the <i>Workspace</i> input variable to a motion control instruction is out of the setting range.	<ul style="list-style-type: none"> The workspace parameter is out of the setting range. 			√	
★570B0000 hex	Invalid Coordinate System Number	The coordinate system number specified for the <i>CSID</i> input variable to a motion control instruction is out of the setting range or not defined.	<ul style="list-style-type: none"> The coordinate system number is out of the setting range. The data type of the coordinate system number is invalid. The specified coordinate system number is not defined by the <i>MC_DefineCoordSystem</i> instruction. 			√	
★570C0000 hex	Coordinate Transform Parameter Setting Out of Range	The <i>Pose</i> coordinate transform parameter specified for <i>CoordTransform</i> input variable to a motion control instruction is out of the setting range.	<ul style="list-style-type: none"> The <i>Pose</i> coordinate transform parameter is out of the setting range. 			√	
★570F0000 hex	Kinematics Calculation Failed	Calculation of inverse kinematics or direct kinematics is not possible.	<ul style="list-style-type: none"> The kinematics parameter setting is not correct. Any one of the axes (A0 to A2) of the robot is stopped at a position where direct kinematics calculation is not possible. The inverse kinematics calculation is not possible for the specified target position on the machine coordinate system. 			√	
★57100000 hex	Kinematics Transform Not Set	The kinematics transform is not set for the specified axes group.	<ul style="list-style-type: none"> The kinematics transform is not set by the <i>MC_SetKinTransform</i> instruction. 			√	
★57120000 hex	Velocity Error Detection Value Setting Out of Range	The velocity error detection value specified for the <i>TrajData</i> input variable to a motion control instruction is out of the setting range.	<ul style="list-style-type: none"> The velocity error detection value is out of the setting range. 			√	

Event code	Event name	Meaning	Assumed cause	Level				
				Major	Partial	Minor	Observation	Information
★57130000 hex	Acceleration Error Detection Value Setting Out of Range	The acceleration error detection value specified for the <i>TrajData</i> input variable to a motion control instruction is out of the setting range.	<ul style="list-style-type: none"> The acceleration error detection value is out of the setting range. 			√		
★57140000 hex	Trajectory Target Time Setting Out of Range	The trajectory target time specified for the <i>TrajData</i> input variable to a motion control instruction is out of the setting range.	<ul style="list-style-type: none"> The trajectory target time is out of the setting range. 			√		
★57150000 hex	Trajectory Type Selection Out of Range	The trajectory type specified for the <i>TrajData</i> input variable to a motion control instruction is out of setting range.	<ul style="list-style-type: none"> The trajectory type is out of the setting range. 			√		
★57160000 hex	Trajectory Transition Setting Out of Range	The trajectory transition specified for the <i>TrajData</i> input variable to a motion control instruction is out of the setting range.	<ul style="list-style-type: none"> The trajectory transition is out of the setting range. 			√		
★57170000 hex	Trajectory Distance Setting Out of Range	The trajectory distance specified for the <i>TrajData</i> input variable to a motion control instruction is out of the setting range.	<ul style="list-style-type: none"> The trajectory distance is out of the setting range. 			√		
★57190000 hex	Initial Workpiece Position Outside Workspace	The workpiece position specified for the <i>InitWorkpiecePosition</i> input variable to a motion control instruction is outside the workspace.	<ul style="list-style-type: none"> The initial workpiece position is outside the workspace. 			√		
★571A0000 hex	Invalid Conveyor Axis Specification	The axis specified for the <i>ConveyorAxis</i> in-out variable of a motion control instruction is not correct.	<ul style="list-style-type: none"> The specified axis is registered in an axes group. An axis does not exist for the variable specified for the Axis input variable to the instruction. The display unit of the conveyor axis is not correct. 			√		
★571B0000 hex	Target Position Outside Workspace	The target position specified for the <i>Position</i> input variable to a motion control instruction is outside the workspace.	<ul style="list-style-type: none"> The target position is outside the workspace. 			√		
★571C0000 hex	Synchronization Release Disabled	The End Synchronization (MC_SyncOut) instruction cannot be executed.	<ul style="list-style-type: none"> The MC_SyncLinearConveyor instruction is not being executed. The MC_SyncLinearConveyor instruction is being executed, but the robot is not in synchronization with the conveyor. (<code>_sMC_SYNC_TRAJ_STATUS.Phase<>6</code>) 			√		
★571E0000 hex	Kinematics Limit Exceeded	The setting number of kinematics transform exceed the limit of setting number.	<ul style="list-style-type: none"> The setting number of kinematics transform exceed 8 by the MC_SetKinTransform instruction.(NJ501-4300/4400/4500) The setting number of kinematics transform exceed 1 by the MC_SetKinTransform instruction.(NJ501-4310) 			√		
★571F0000 hex	Kinematics Initialization Error	The initialization of kinematics failed.	<ul style="list-style-type: none"> Any one of the axes (A0 to A2) of the robot is stopped at a position where direct kinematics calculation is not possible. 			√		
★67000000 hex	Command Position Outside Workspace	The command position is outside the workspace.	<ul style="list-style-type: none"> The specified trajectory data is not correct. (The trajectory passes outside the workspace.) 			√		
★67010000 hex	Current Position Outside Workspace	The current position at the time when the instruction is executed is outside the workspace.	<ul style="list-style-type: none"> The current position at the time when any one of the following instructions is executed is outside the workspace. MC_MoveTimeAbsolute MC_SyncLinearConveyor MC_SyncOut 			√		
★67020000 hex	Workpiece Synchronization Error Limit Exceeded	Failed to catch up with the workpiece on the conveyor.	<ul style="list-style-type: none"> The conveyor position was changed immediately before the robot catches up with the workpiece. Actual current position of the conveyor is not correct. 			√		

Event code	Event name	Meaning	Assumed cause	Level			
				Major	Partial	Minor	Information
★67030000 hex	Velocity Error Detected	The command velocity exceeded the velocity error detection value specified for the <i>TrajData</i> input variable to a motion control instruction.	<ul style="list-style-type: none"> The command velocity in the combination of the parameters (trajectory target time and distance (= target position - command current position)) specified for the <i>TrajData</i> input variable is too fast. The parameter specified for the <i>TrajData</i> input variable is not correct. The velocity error detection value is too low. 			√	
★67040000 hex	Acceleration Error Detected	The command acceleration exceeded the acceleration error detection value specified for the <i>TrajData</i> input variable to a motion control instruction.	<ul style="list-style-type: none"> The command acceleration in the combination of the parameters (trajectory target time and distance (= target position - command current position)) specified for the <i>TrajData</i> input variable is too high. The parameter specified for the <i>TrajData</i> input variable is not correct. The acceleration error detection value is too low. 			√	
★67050000Hex	Command Current Velocity Axis Maximum Velocity Exceeded	The specified command current velocity exceeds the maximum velocity of axis.	<ul style="list-style-type: none"> The robot was moved with a high speed. Then, the specified command current velocity exceeds the maximum velocity of axis 			√	
74300000 hex	Axes Group Composition Axis Error	An error occurred for an axis in an axes group.	<ul style="list-style-type: none"> An error occurred for an axis in an axes group that was in motion. 			√	
★77000000Hex	Conveyor Axis Position Read Error	The MC_SyncLinearConveyor was not executed because an error occurred in the position of the conveyor axis of the MC_SyncLinearConveyor.	<ul style="list-style-type: none"> EtherCAT process data communications are not established for the conveyor axis of the MC_SyncLinearConveyor. The slave of the conveyor axis for the MC_SyncLinearConveyor was disconnected. An Absolute Encoder Current Position Calculation Failed error (0x64580000) was detected for the conveyor axis of a MC_SyncLinearConveyor. 			√	

Event code	Event name	Meaning	Assumed cause	Level			
				Major	Partial	Minor	Information
★57320000Hex	Illegal Robot Tool ID	The Tool Coordinate System Identifier specified as variable for the input ToolID is out of range or has not been defined.	<ul style="list-style-type: none"> Specified wrong value of ToolID input. Specified Tool Coordinate System ID has not been defined in MC_DefineToolTransform. 			√	
★57330000Hex	Illegal Tool Coordinate Transformation parameters	The Tool Coordinate Transformation parameters specified as variable for the input CoordTransform are of wrong values.	<ul style="list-style-type: none"> Specified a Tool Coordinate System Transformation Pose of wrong values. 			√	
★57340000Hex	Not supported Transition Mode Data	The selected TransitionMode is not supported with the selected data of current or buffered instruction.	<ul style="list-style-type: none"> TransitionMode is not possible to implement with the selected parameters (Transition Parameters) TransitionMode is not possible to implement for the selected parameters (Trajectory Data) of the buffered Instruction. 			√	
★57360000Hex	Illegal execution of Offset functionality	The offset functionality cannot be enabled because MC_SyncLinearConveyor is not in Phase 6.	<ul style="list-style-type: none"> EnableOffset is TRUE when MC_SyncLinearConveyor has not reached to Phase6. 			√	
★57370000Hex	Motion Control Instruction Multi-execution Disabled (trajectory type)	On Multi-execution, Trajectory type of buffered instruction is changed.	<ul style="list-style-type: none"> Trajectory type of buffered instruction is difference to current instruction. 			√	
★57390000Hex	Current Instruction does not support TransitionMode	Specified Transition Mode is not supported by the combination of the instructions.	<ul style="list-style-type: none"> The current instruction does not support the selected Transition Mode with the buffered instruction. 			√	
★57440000Hex	Illegal Jog Mode	JogMode input is out of range or is wrong type	<ul style="list-style-type: none"> Specified an invalid JogMode input argument value. 			√	
★57450000Hex	Initial Workpiece Position setting out of range	The InitialWorkpiecePosition value specified is out of range.	<ul style="list-style-type: none"> Specified wrong value of InitWorkpiecePosition input 			√	
★57460000Hex	Illegal Maximum interpolation velocity.	The Maximum Interpolation Velocity value specified is out of range	<ul style="list-style-type: none"> Specified wrong value of Maximum Velocity of the robot 			√	
★57470000Hex	Illegal Maximum interpolation acceleration	The Maximum Interpolation Acceleration value specified is out of range	<ul style="list-style-type: none"> Specified wrong value of Maximum Acceleration of the robot 			√	
★57480000Hex	Illegal Maximum interpolation deceleration	The Maximum Interpolation Deceleration value specified is out of range	<ul style="list-style-type: none"> Specified wrong value of Maximum Deceleration of the robot 			√	
★645A0000Hex	Maximum interpolation velocity over-passed	The commanded velocity of the default TCP overpasses the maximum robot velocity set in MC_SetKinTransform.	<ul style="list-style-type: none"> Wrong specified trajectory data of current instruction. Specified value for MaxVelocityTCP input variable of MC_SetKinTransform is wrong or too low. 			√	
★645B0000Hex	Maximum interpolation acceleration over-passed	The commanded acceleration of the default TCP overpasses the maximum robot acceleration set in MC_SetKinTransform.	<ul style="list-style-type: none"> Wrong specified trajectory data of current instruction. Specified value for MaxAccelerationTCP input variable of MC_SetKinTransform is wrong or too low. 			√	
★645C0000Hex	Maximum interpolation deceleration over-passed	The commanded deceleration of the default TCP overpasses the maximum robot deceleration set in MC_SetKinTransform.	<ul style="list-style-type: none"> Wrong specified trajectory data of current instruction. Specified value for MaxDecelerationTCP input variable of MC_SetKinTransform is wrong or too low. 			√	
★94230000Hex	Started the Transition by correcting TransitionParameters	Started the Transition by correcting TransitionParameters.	<ul style="list-style-type: none"> When Multi-execution, current instruction is already over the set TransitionParameter of buffered instruction. 				√

9.2 Error Code Details

Event name	Buffer Mode Selection Out of Range			Event code	542B0000 hex	
Meaning	The parameter specified for the <i>BufferMode</i> input variable to a motion control instruction is out of range.					
Source	Motion Control Function Module		Source details	Axis / axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	<p>If "axis" is given for the source details, operation is not possible for relevant axis. Relevant axis decelerates to a stop if it is in motion.</p> <p>If "axes group" is given for the source details, operation is not possible for relevant axes group. If relevant axes group is in motion, non-robot axes group decelerates to a stop and robot axes group stops immediately.</p>		
System-defined variables	Variable		Data type		Name	
	_MC_AX[*].MFAultLvl.Active		BOOL		Axis Minor Fault Occurrence	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Instruction input parameter exceeded the valid range of the input variable.		Correct the parameter so that the valid range of the input variable is not exceeded for the relevant instruction.		Set the input parameter to the instruction so that the valid range of the input variable is not exceeded.	
Attached information	None					
Precautions/Remarks	None					

Event name	Coordinate System Selection Out of Range			Event code	542C0000 hex	
Meaning	The parameter specified for the <i>CoordSystem</i> input variable to a motion control instruction is out of range.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	<p>Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion. If the axes group is robot, stops immediately.</p>		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Instruction input parameter exceeded the valid range of the input variable.		Correct the parameter so that the valid range of the input variable is not exceeded for the relevant instruction.		Set the input parameter to the instruction so that the valid range of the input variable is not exceeded.	
Attached information	None					
Precautions/Remarks	None					

Event name	Transition Mode Selection Out of Range			Event code	54320000 hex	
Meaning	The parameter specified for the <i>TransitionMode</i> input variable to a motion control instruction is out of range.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. If relevant axes group is in motion, non-robot axes group decelerates to a stop and robot axes group stops immediately.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Instruction input parameter exceeded the valid range of the input variable.		Correct the parameter so that the valid range of the input variable is not exceeded for the relevant instruction.		Set the input parameter to the instruction so that the valid range of the input variable is not exceeded.	
Cause and correction	_mcAborting or _mcBuffered was specified for <i>BufferMode</i> and other than _mcTMNone was specified for <i>TransitionMode</i> .		If you specify _mcAborting or _mcBuffered for <i>BufferMode</i> , specify _mcTMNone for <i>TransitionMode</i> . If you specify other than _mcTMNone for <i>TransitionMode</i> , specify _mcBlendingLow, _mcBlendingPrevious, _mcBlendingNext, or _mcBlendingHigh for <i>BufferMode</i> .		If you specify _mcAborting or _mcBuffered for <i>BufferMode</i> , specify _mcTMNone for <i>TransitionMode</i> . If you specify other than _mcTMNone for <i>TransitionMode</i> , specify _mcBlendingLow, _mcBlendingPrevious, _mcBlendingNext, or _mcBlendingHigh for <i>BufferMode</i> .	
			None		None	
Attached information	None					
Precautions/Remarks	None					

Event name	Motion Control Instruction Re-execution Disabled			Event code	543B0000 hex	
Meaning	An attempt was made to re-execute a motion control instruction that cannot be re-executed.					
Source	Motion Control Function Module		Source details	MC common / axis / axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	If "MC Common" is given for the source details, operation is not affected. If "axis" is given for the source details, operation is not possible for relevant axis. Relevant axis decelerates to a stop if it is in motion. If "axes group" is given for the source details, operation is not possible for relevant axes group. If relevant axes group is in motion, non-robot axes group decelerates to a stop and robot axes group stops immediately.		
System-defined variables	Variable		Data type		Name	
	_MC_COM.MFaultLvl.Active		BOOL		MC Common Minor Fault Occurrence	
	_MC_AX[*].MFaultLvl.Active		BOOL		Axis Minor Fault Occurrence	
Cause and correction	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
	Assumed cause		Correction		Prevention	
	A motion control instruction that cannot be re-executed was re-executed.		Correct the program so that the <i>Execute</i> input variable does not change to TRUE until the <i>Busy</i> output variable from the instruction changes to FALSE.		When using instructions that cannot be re-executed, include a condition for the <i>Execute</i> input variable so that it does not change to TRUE unless the <i>Busy</i> output variable for the previous instruction is FALSE. Or, stop the instruction before executing it again.	
Attached information	None					
Precautions/Remarks	None					

Event name	Motion Control Instruction Multi-execution Disabled		Event code	543C0000 hex		
Meaning	Multiple functions that cannot be executed simultaneously were executed for the same target (MC common, axis, or ★axes group).					
Source	Motion Control Function Module		Source details	MC common / axis / axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	<p>If "MC Common" is given for the source details, operation is not affected.</p> <p>If "axis" is given for the source details, operation is not possible for relevant axis. Relevant axis decelerates to a stop if it is in motion.</p> <p>If "axes group" is given for the source details, operation is not possible for relevant axes group. If relevant axes group is in motion, non-robot axes group decelerates to a stop and robot axes group stops immediately.</p>		
System-defined variables	Variable		Data type		Name	
	_MC_COM.MFaultLvl.Active		BOOL		MC Common Minor Fault Occurrence	
	_MC_AX[*].MFaultLvl.Active		BOOL		Axis Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Multiple functions that cannot be executed simultaneously were executed for the same target (MC common, axis, or ★axes group).		Check the specifications of multi-execution of instructions for this instruction and do not execute instructions that cannot be executed at the same time.		Check the specifications for multi-execution of instructions for the instruction and do not execute instructions that cannot be executed at the same time.	
Attached information	None					
Precautions/Remarks	None					

Event name	Instruction Cannot Be Executed during Multi-axes Coordinated Control		Event code	543E0000 hex		
Meaning	An instruction that cannot be used while the axes group is enabled was executed.					
Source	Motion Control Function Module		Source details	Axis / axes group	Detection timing	At instruction execution / at multi-execution of instructions
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	<p>If "axis" is given for the source details, operation is not possible for relevant axis. Relevant axis decelerates to a stop if it is in motion.</p> <p>If "axes group" is given for the source details, operation is not possible for relevant axes group. If relevant axes group is in motion, non-robot axes group decelerates to a stop and robot axes group stops immediately.</p>		
System-defined variables	Variable		Data type		Name	
	_MC_AX[*].MFaultLvl.Active		BOOL		Axis Minor Fault Occurrence	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	An operation instruction was executed for an axis that was in a coordinated multi-axes motion.		Correct the program so that axis operation instructions are executed only for axes that are not in coordinated multi-axes motion.		Execute axis operation instructions only for axes that are not in coordinated multi-axes motion.	
	★ The MC_SetKinTransform instruction was executed for an axes group that is enabled.		Correct the program so that the instruction is executed when the axes group is disabled.		Execute the instruction when the axes group is disabled.	
Attached information	None					
Precautions/Remarks	None					

Event name	Multi-axes Coordinated Control Instruction Executed for Disabled Axes Group		Event code	543F0000 hex		
Meaning	A multi-axes coordinated control instruction was executed for an axes group that was in the Axes Group Disabled state.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	A multi-axes coordinated control instruction was executed for an axes group that was in the Axes Group Disabled state.		Correct the program so that the instruction is executed only after changing the axes group to the Axes Group Enabled state. Execute the MC_GroupEnable (Enable Axes Group) instruction to change an axes group to the Axes Group Enabled state.		Execute multi-axes coordinated operation instructions only after enabling the axes group. Execute the MC_GroupEnable (Enable Axes Group) instruction to change an axes group to the Axes Group Enabled state.	
Attached information	None					
Precautions/Remarks	None					

Event name	Impossible Axis Operation Specified when the Servo is OFF		Event code	54410000 hex		
Meaning	An operation instruction was executed for an axis for which the Servo is OFF.					
Source	Motion Control Function Module		Source details	Axis / axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	
Effects	User program	Continues.	Operation	The operation instruction will not start.		
System-defined variables	Variable		Data type		Name	
	_MC_AX[*].MFAultLvl.Active		BOOL		Axis Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	An operation instruction was executed for an axis for which the Servo is OFF.		Correct the program so that the instruction is executed after the Servo is turned ON.		Make sure to execute the axis operation instruction after the Servo is turned ON.	
Cause and correction	Home was preset with the MC_Home instruction for an axis for which EtherCAT process data communications are not established.		If the <i>_EC_PDSlavTbl</i> (Process Data Communicating Slave Table) system-defined variable for the EtherCAT master is FALSE, remove the cause and execute the MC_Home instruction to preset home after <i>_EC_PDSlavTbl</i> changes to TRUE.		If you execute the MC_Home instruction to preset home immediately after you turn ON the power supply to the Controller, download data, reset a slave communications error, disconnect the slave, or reconnect the slave, write the program to make sure that the <i>_EC_PDSlavTbl</i> (Process Data Communicating Slave Table) system-defined variable for the EtherCAT master is TRUE before you execute MC_Home.	
	Attached information 1: Depends on the source details. Axis: 0 Axes group: Number of the logical axis where the error occurred.					
Precautions/Remarks	None					

Event name	Illegal Axes Group Specification		Event code	54610000 hex		
Meaning	The axes group specified for the <i>AxesGroup</i> input variable to a motion control instruction does not exist or is not a used group.					
Source	Motion Control Function Module		Source details	MC Common	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	The relevant instruction is not executed.		
System-defined variables	Variable		Data type		Name	
	MC_COM.MFaultLvl.Active		BOOL		MC Common Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	An axes group does not exist for the variable specified for the <i>AxesGroup</i> input variable to the instruction.		Correct the specification for the instruction so that the specified axes group exists.		Specify a variable that exists when specifying a variable for an input parameter to an instruction.	
	The axes group specified for the <i>AxesGroup</i> input variable to the instruction is not specified as a used group.		Correct the axes group specified by the instruction to a used group.		Set a used axes group for the <i>AxesGroup</i> input variable to the instruction.	
Attached information	None					
Precautions/Remarks	None					

Event name	Instruction Execution Error with Undefined Home		Event code	54660000 hex		
Meaning	High-speed homing or an interpolation instruction ★ or an robot instruction was executed when home was undefined.					
Source	Motion Control Function Module		Source details	Axis / axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	
Effects	User program	Continues.	Operation	If "axis" is given for the source details, operation is not possible for relevant axis. Relevant axis decelerates to a stop if it is in motion. If "axes group" is given for the source details, operation is not possible for relevant axes group. If relevant axes group is in motion, non-robot axes group decelerates to a stop and robot axes group stops immediately.		
System-defined variables	Variable		Data type		Name	
	MC_AX[*].MFaultLvl.Active		BOOL		Axis Minor Fault Occurrence	
	MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	High-speed homing was executed when home was undefined.		Execute the high-speed homing operation only after homing to define home.		Execute the high-speed homing instruction only after home is defined by homing.	
	An interpolation instruction or an robot instruction was executed for an axes group that includes an axis with no defined home.		Perform homing to define home for all axes in the axes group before executing the interpolation instruction.		Perform homing to define home for all axes in the axes group before executing the interpolation instruction.	
Attached information	Attached information: Depends on the source details. Axis: 0 Axes group: Logical axis number					
Precautions/Remarks	If you execute the Set Position instruction after making the Servo Drive ready to operate or performing homing, home will again be undefined. You must make the Servo Drive ready to operate or perform homing again to define home in this case.					

Event name	Instruction Not Supporting Kinematics Transform		Event code	★57050000 hex		
Meaning	An instruction that cannot be used when kinematics transform is set for the axes group was executed.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Any one of the following instructions was executed for an axes group for which a kinematics transform is set. MC_MoveLinear MC_MoveLinearAbsolute MC_MoveLinearRelative MC_MoveCircular2D MC_ChangeAxesInGroup		Correct the program so that the instruction is not executed for the axes group for which kinematics transform is set.		Do not execute the instructions for the axes group for which kinematics transform is set.	
Attached information	None					
Precautions/Remarks	None					

Event name	Different Composition between Axes Group and Kinematics		Event code	★57060000 hex		
Meaning	The composition is different between the specified axes group and the specified kinematics.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The number of axes in the axes group is different from the number of axes necessary for the specified robot (kinematics type).		Change the number of axes in the axes group to the number of axes necessary for the specified robot (kinematics type).		Set the same number of axes in the axes group as the number of axes necessary for the kinematics.	
	The count mode of an axis registered in the axes group is different from the count mode for the specified robot (kinematics type).		Change the count mode of all axes registered in the axes group to necessary count mode for the specified robot (kinematics type).		Set the count mode of all axes registered in the axes group to necessary count mode for the kinematics.	
	The display unit of an axis registered in the axes group is different from the display unit for the specified robot (kinematics type).		Change the display unit of all axes registered in the axes group to necessary display unit for the specified robot (kinematics type).		Set the display unit of all axes registered in the axes group to necessary display unit for the kinematics.	
Attached information	None					
Precautions/Remarks	None					

Event name	Kinematics Type Selection Out of Range		Event code	★57070000 hex		
Meaning	The kinematics type specified for the <i>KinTransform</i> input variable to a motion control instruction is out of the setting range.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The kinematics type is out of setting range.		Correct the kinematics type for the <i>KinTransform</i> input variable.		Set a correct kinematics type for the <i>KinTransform</i> input variable.	
Attached information	None					
Precautions/Remarks	None					

Event name	Kinematics Parameter Setting Out of Range		Event code	★57080000 hex		
Meaning	The kinematics parameter specified for the <i>KinTransform</i> input variable to a motion control instruction is out of the setting range.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	A kinematics parameter is out of the setting range.		Correct the kinematics parameter or the expansion parameter for the <i>KinTransform</i> input variable.		Set a correct kinematics parameter or expansion parameter for the <i>KinTransform</i> input variable.	
	An expansion parameter is out of the setting range.					
Attached information	None					
Precautions/Remarks	None					

Event name	Workspace Type Selection Out of Range		Event code	★57090000 hex		
Meaning	The workspace type specified for the <i>Workspace</i> input variable to a motion control instruction is out of the setting range.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The workspace type is out of the setting range.		Correct the workspace type for the <i>Workspace</i> input variable.		Set a correct workspace type for the <i>Workspace</i> input variable.	
Attached information	None					
Precautions/Remarks	None					

Event name	Workspace Parameter Setting Out of Range		Event code	★570A0000 hex		
Meaning	The workspace parameter specified for the <i>Workspace</i> input variable to a motion control instruction is out of the setting range.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The workspace parameter is out of the setting range.		Correct the workspace parameter for the <i>Workspace</i> input variable.		Set a correct workspace parameter for the <i>Workspace</i> input variable.	
Attached information	None					
Precautions/Remarks	None					

Event name	Invalid Coordinate System Number		Event code	★570B0000 hex		
Meaning	The coordinate system number specified for the <i>CSID</i> input variable to a motion control instruction is out of the setting range or not defined.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The coordinate system number is out of the setting range.		Correct the coordinate system ID for the <i>CSID</i> input variable.		Set a correct coordinate system ID for the <i>CSID</i> input variable.	
	The specified coordinate system number is not defined by the <i>MC_DefineCoordSystem</i> instruction.		Define the coordinate system by the <i>MC_DefineCoordSystem</i> instruction.		Define the coordinate system and use the defined coordinate system.	
Attached information	None					
Precautions/Remarks	None					

Event name	Coordinate Transform Parameter Setting Out of Range		Event code	★570C0000 hex		
Meaning	The <i>Pose</i> coordinate transform parameter specified for <i>CoordTransform</i> input variable to a motion control instruction is out of the setting range.					
Source	Motion Control Function Module		Source details	MC common	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	---	Log category	System
Effects	User program	Continues.	Operation	Not affected.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The <i>Pose</i> coordinate transform parameter is out of the setting range.		Correct the <i>Pose</i> coordinate transform parameter for the <i>CoordTransform</i> input variable.		Set a correct <i>Pose</i> coordinate transform parameter for the <i>CoordTransform</i> input variable.	
Attached information	None					
Precautions/Remarks	None					

Event name	Kinematics Calculation Failed		Event code	★570F0000 hex		
Meaning	Calculation of inverse kinematics or direct kinematics is not possible.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution / during instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The kinematics parameter setting is not correct.		Correct the kinematics parameter.		Set a correct kinematics parameter.	
	Any one of the axes (A0 to A2) of the robot is stopped at a position where direct kinematics calculation is not possible.		After turning OFF the Servo of each axis of the robot, ensure the safety, and then move the robot to the moving range.		Reconsider the program to move the robot within the moving range.	
	The inverse kinematics calculation is not possible for the specified target position on the machine coordinate system.		Set the target position within the moving range of the robot.		Set the target position within the moving range of the robot.	
Attached information	None					
Precautions/Remarks	None					

Event name	Kinematics Transform Not Set		Event code	★57100000 hex		
Meaning	The kinematics transform is not set for the specified axes group.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The kinematics transform is not set for the axes group.		Set the kinematics transform for the axes group using the MC_SetKinTransform instruction.		Execute the instruction after setting the kinematics transform for the axes group.	
Attached information	None					
Precautions/Remarks	None					

Event name	Velocity Error Detection Value Setting Out of Range		Event code	★57120000 hex		
Meaning	The velocity error detection value specified for the <i>TrajData</i> input variable to a motion control instruction is out of the setting range.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The velocity error detection value is out of the setting range.		Correct the velocity error detection value.		Set a correct velocity error detection value.	
Attached information	None					
Precautions/Remarks	None					

Event name	Acceleration Error Detection Value Setting Out of Range		Event code	★57130000 hex		
Meaning	The acceleration error detection value specified for the <i>TrajData</i> input variable to a motion control instruction is out of the setting range.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The acceleration error detection value is out of the setting range.		Correct the acceleration error detection value.		Set a correct acceleration error detection value.	
Attached information	None					
Precautions/Remarks	None					

Event name	Trajectory Target Time Setting Out of Range			Event code	★57140000 hex	
Meaning	The trajectory target time specified for the <i>TrajData</i> input variable to a motion control instruction is out of the setting range.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The trajectory target time is out of the setting range.		Correct the trajectory target time.		Set a correct trajectory target time.	
Attached information	None					
Precautions/Remarks	None					

Event name	Trajectory Type Selection Out of Range			Event code	★57150000 hex	
Meaning	The trajectory type specified for the <i>TrajData</i> input variable to a motion control instruction is out of the setting range.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The trajectory type is out of the setting range.		Correct the trajectory type.		Set a correct trajectory type.	
Attached information	None					
Precautions/Remarks	None					

Event name	Trajectory Transition Setting Out of Range			Event code	★57160000 hex	
Meaning	The trajectory transition specified for the <i>TrajData</i> input variable to a motion control instruction is out of the setting range.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The trajectory transition is out of the setting range.		Correct the trajectory transition.		Set a correct trajectory transition.	
Attached information	None					
Precautions/Remarks	None					

Event name	Trajectory Distance Setting Out of Range		Event code	★57170000 hex		
Meaning	The trajectory distance specified for the <i>TrajData</i> input variable to a motion control instruction is out of the setting range.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The trajectory distance is out of the setting range.		Correct the trajectory distance.		Set a correct trajectory distance.	
Attached information	None					
Precautions/Remarks	None					

Event name	Initial Workpiece Position Outside Workspace		Event code	★57190000 hex		
Meaning	The workpiece position specified for the <i>InitWorkpiecePosition</i> input variable to a motion control instruction is outside the workspace.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The initial workpiece position is outside the workspace.		Move the workpiece into the workspace.		Make the initial position of the workpiece within the range of workspace.	
Attached information	None					
Precautions/Remarks	None					

Event name	Invalid Conveyor Axis Specification		Event code	★571A0000 hex		
Meaning	The axis specified for the <i>ConveyorAxis</i> in-out variable of a motion control instruction is not correct.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The specified axis is registered in an axes group.		Change to a correct axis.		Set a correct axis.	
	The specified axis does not exist.		Change to an existing axis.		Set an existing axis.	
	The display unit of the conveyor axis is not correct.		Correct the display unit of the axis to [mm].		Set a correct unit [mm] for the axis.	
Attached information	None					
Precautions/Remarks	None					

Event name	Target Position Outside Workspace		Event code	★571B0000 hex		
Meaning	The target position specified for the <i>Position</i> input variable to a motion control instruction is outside the workspace.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The target position is outside the workspace.		Correct the target position.		Set a target position within the range of workspace.	
Attached information	None					
Precautions/Remarks	None					

Event name	Synchronization Release Disabled		Event code	★571C0000 hex		
Meaning	The End Synchronization (MC_SyncOut) instruction cannot be executed.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The MC_SyncLinearConveyor instruction is not executed.		Execute the MC_SyncOut instruction when <i>TrajStatus.Phase</i> is 6.		Execute the instruction when the phase of MC_SyncLinearConveyor instruction is 6 (<i>TrajStatus.Phase</i> = 6).	
	The MC_SyncLinearConveyor instruction is being executed, but the robot is not in synchronization with the conveyor (i.e. not <i>TrajStatus.Phase</i> = 6).					
Attached information	None					
Precautions/Remarks	None					

Event name	Kinematics Limit Exceeded		Event code	571E0000 hex		
Meaning	The setting number of kinematics transform exceed the limit of setting number.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The setting number of kinematics transform exceeds the limit by MC_SetKinTransform instruction.		Correct the program so that the setting number of kinematics transform does not exceed the limit.		Set the number of robots 8 and fewer by MC_SetKinTransform instruction. (NJ501-4300/4400/4500) Set the number of robots 1 and fewer by MC_SetKinTransform instruction. (NJ501-4310)	
Attached information	None					
Precautions/Remarks	None					

Event name	Kinematics Initialization Error			Event code	571F0000 hex	
Meaning	The initialization of kinematics failed.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Any one of the axes 0028A0 to A3) of the robot is stopped at a position where direct kinematics calculation is not possible.		Power off each axis of the robot for safety. And, move the each axis inside the working volume of the robot. When this error is detected, the setting parameters of kinematics, workspace check, user coordinate systems and tool coordinate system for the relevant axes group will be cleared. So, please set these parameters again before moving the robot.		Correct the program so that the robot moves inside the working volume of the robot.	
Attached information	None					
Precautions/Remarks	None					

Event name	Command Position Outside Workspace			Event code	★67000000 hex	
Meaning	The command position is outside the workspace.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution / during instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The specified trajectory data is not correct.		Correct the trajectory data so that the robot moves within the range of workspace.		Set the trajectory data so that the command position of the trajectory does not go out of the workspace.	
Attached information	None					
Precautions/Remarks	None					

Event name	Current Position Outside Workspace			Event code	★67010000 hex	
Meaning	The current position at the time when the instruction is executed is outside the workspace.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The current position at the time when any one of the following instructions is executed is outside the workspace. MC_MoveTimeAbsolute MC_SyncLinearConveyor MC_SyncOut		Move the robot into the workspace.		Correct the program so that the robot moves inside the workspace.	
Attached information	None					
Precautions/Remarks	None					

Event name	Workpiece Synchronization Deviation Over		Event code	★67020000 hex		
Meaning	Failed to catch up with the workpiece on the conveyor.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	During instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The conveyor position was changed immediately before the robot catches up with the workpiece.		Make the conveyor velocity constant, or do not change the velocity rapidly.		Make the conveyor velocity constant.	
	Actual position of the conveyor is not correct.		Correct the actual position of the conveyor.		Make the settings so that the current position of the conveyor is updated correctly.	
Attached information	None					
Precautions/Remarks	None					

Event name	Velocity Error Detected		Event code	★67030000 hex		
Meaning	The command velocity exceeded the velocity error detection value specified for the <i>TrajData</i> input variable to a motion control instruction.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution / during instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The command velocity in the combination of the parameters specified for the <i>TrajData</i> input variable is too fast.		Set the parameters so that the command velocity becomes lower than the velocity error detection value.		Set the parameters so that the command velocity of the trajectory becomes lower than the velocity error detection value.	
	The parameter specified for the <i>TrajData</i> input variable is not correct.		Correct the parameter.		Set a correct parameter.	
	The velocity error detection value is too low.		Correct the velocity error detection value.		Set an appropriate velocity error detection value.	
Attached information	None					
Precautions/Remarks	None					

Event name	Acceleration Error Detected		Event code	★67040000 hex		
Meaning	The command acceleration exceeded the acceleration error detection value specified for the <i>TrajData</i> input variable to a motion control instruction.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution / during instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The command acceleration in the combination of the parameters (trajectory target time and distance (= target position - command current position)) specified for the <i>TrajData</i> input variable is too high.		Correct the parameters so that the command acceleration becomes lower.		Set the parameters so that the command acceleration of the trajectory becomes lower than the acceleration error detection value.	
	The parameter specified for the <i>TrajData</i> input variable is not correct.		Correct the parameter.		Set the input parameter to the instruction so that the valid range of the input variable is not exceeded.	
	The acceleration error detection value is too low.		Correct the acceleration error detection value.		Set an appropriate acceleration error detection value.	
Attached information	None					
Precautions/Remarks	None					

Event name	Command Current Velocity Axis Maximum Velocity Exceeded		Event code	★67050000 hex		
Meaning	The specified command current velocity exceeds the maximum velocity of axis.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	During instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The specified command current velocity exceeds the maximum velocity of axis.		Lower the operation velocity of the robot.		Set the input parameter to the instruction so that the maximum velocity of each axis is not exceeded.	
Attached information	Attached information 1: Depends on the source details. Axis: 0 Axes group: Number of the logical axis where the error occurred					
Precautions/Remarks	None					

Event name	Axes Group Composition Axis Error			Event code	74300000 hex	
Meaning	An error occurred for a composition axis in an axes group.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At any time
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Non-robot axes group: Follows the operation set in the axes group stop method if a composition axis stops. In the other cases, relevant axes group stops on the interpolation trajectory. Robot axes group: Relevant axes group stops immediately.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	An error occurred for an axis in an axes group that was in motion.		Check the error code for composition axes of the axes group and remove the cause.		None	
Attached information	None					
Precautions/Remarks	When an axis error occurs, the axes group that includes the axis cannot operate.					

Event name	Conveyor Axis Position Read Error			Event code	★77000000 hex	
Meaning	The MC_SyncLinearConveyor was not executed because an error occurred in the position of the conveyor axis of the MC_SyncLinearConveyor.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution / during instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues.	Operation	Non-robot axes group: Follows the operation set in the axes group stop method if a composition axis stops. In the other cases, relevant axes group stops on the interpolation trajectory. Robot axes group: Relevant axes group stops immediately.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	EtherCAT process data communications are not established for the conveyor axis of the MC_SyncLinearConveyor.		If the _EC_PDslavTbl (Process Data Communicating Slave Table) system-defined variable for the EtherCAT master of the conveyor axis is FALSE, investigate the error in the conveyor axis and remove the cause.		If you execute synchronized instructions after you turn ON the power supply, download data, or reset slave communications error, make sure that the _EC_PDslavTbl (Process Data Communicating Slave Table) system-defined variable for the EtherCAT master is TRUE for the node of the master axis before you execute the synchronized instruction.	
	The slave of the conveyor axis for the MC_SyncLinearConveyor was disconnected.		Check the slave of the conveyor axis and reconnect it if it was disconnected.		Make sure that the slave of the master axis is not disconnected during execution of a synchronized instruction.	
	An Absolute Encoder Current Position Calculation Failed error (0x64580000) was detected for the conveyor axis of a MC_SyncLinearConveyor.		See if an Absolute Encoder Current Position Calculation Failed error (0x64580000) occurred for the conveyor axis and make suitable corrections to restore operation.		Do not use an axis with an Absolute Encoder Current Position Calculation Failed error (64580000 hex) as the master axis in a synchronized instruction.	
Attached information	None					
Precautions/Remarks	None					

Event name	Illegal Robot Tool ID		Event code	★5732000Hex		
Meaning	The Tool Coordinate System Identifier specified as variable for the input ToolID is out of range or has not been defined.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Specified wrong value of ToolID input		Correct to tool ID of the ToolID input variable.		Set a correct tool ID for the ToolID input variable.	
	Specified Tool Coordinate System ID has not been defined in MC_DefineToolTransform.		Define the tool with MC_DefineToolTransform.		Define a tool and select the defined tool.	
Attached information	None					
Precautions/Remarks	None					

Event name	Illegal Tool Coordinate Transformation parameters		Event code	★5733000Hex		
Meaning	The Tool Coordinate Transformation parameters specified as variable for the input CoordTransform are of wrong values.					
Source	Motion Control Function Module		Source details	MC common	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Not affected.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Specified a Tool Coordinate System Transformation Pose of wrong values.		Correct the parameters of the ToolTransform input variable.		Set a correct parameters for the ToolTransform input variable.	
Attached information	None					
Precautions/Remarks	None					

Event name	Not supported Transition Mode Data		Event code	★5734000Hex		
Meaning	The selected TransitionMode is not supported with the selected data of current or buffered instruction.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	<i>TransitionMode</i> is not possible to implement with the selected parameters (Transition Parameters)		Correct the <i>TransitionParameter</i> so that the transition can be implemented.		Set a correct <i>TransitionParameter</i> so that the transition can be implemented.	
	<i>TransitionMode</i> is not possible to implement for the selected parameters (Trajectory Data) of the buffered Instruction.		Correct the Trajectory Data of the buffered command so that the transition can be implemented.		Set a correct Trajectory Data of the buffered command so that the transition can be implemented.	
Attached information	None					
Precautions/Remarks	None					

Event name	Illegal execution of Offset functionality		Event code	★57360000Hex		
Meaning	The offset functionality cannot be enabled because MC_SyncLinearConveyor is not in Phase 6.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution / during instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	EnableOffset is TRUE when MC_SyncLinearConveyor has not reached to Phase6.		Set EnableOffset during synchronization (TrajStatus.Phase=6).		Set EnableOffset during synchronization (TrajStatus.Phase=6).	
Attached information	None					
Precautions/Remarks	None					

Event name	Motion Control Instruction Multi-execution Disabled (trajectory type)		Event code	★57370000Hex		
Meaning	On Multi-execution, Trajectory type of buffered instruction is changed.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At multi-execution of instructions
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Trajectory type of buffered instruction is difference to current instruction.		Set Trajectory type to the same type of current instruction.		Set Trajectory type to the same type of current instruction.	
Attached information	None					
Precautions/Remarks	None					

Event name	Current Instruction does not support TransitionMode		Event code	★57390000Hex		
Meaning	Specified Transition Mode is not supported by the combination of the instructions.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At multi-execution of instructions
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	The current instruction does not support the selected TransitionMode with the buffered instruction.		Correct the TransitionMode.		Set a correct TransitionMode.	
Attached information	None					
Precautions/Remarks	None					

Event name	Illegal Jog Mode		Event code	★57440000Hex		
Meaning	JogMode input is out of range or is wrong type					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Specified an invalid <i>JogMode</i> input argument value		Correct the value of the <i>JogMode</i> input variable.		Set a correct value for the <i>JogMode</i> input variable.	
Attached information	None					
Precautions/Remarks	None					

Event name	Initial Workpiece Position setting out of range		Event code	★57450000Hex		
Meaning	The <i>InitialWorkpiecePosition</i> value specified is out of range.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Specified wrong value of <i>InitWorkpiecePosition</i> input		Correct the position of the <i>InitWorkpiecePosition</i> input variable.		Correct the program so that the value of <i>InitWorkpiecePosition</i> is in range.	
Attached information	--					
Precautions/Remarks	None					

Event name	Illegal Maximum interpolation velocity.		Event code	★57460000Hex		
Meaning	The Maximum Interpolation Velocity value specified is out of range					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Specified wrong value of Maximum Velocity of the robot		Correct the value of the <i>MaxVelocityTCP</i> input variable.		Set a correct value for the <i>MaxVelocityTCP</i> input variable.	
Attached information	None					
Precautions/Remarks	None					

Event name	Illegal Maximum interpolation acceleration		Event code	★57470000Hex		
Meaning	The Maximum Interpolation Acceleration value specified is out of range					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Specified wrong value of Maximum Acceleration of the robot		Correct the value of the <i>MaxAccelerationTCP</i> input variable.		Set a correct value for the <i>MaxAccelerationTCP</i> input variable.	
Attached information	None					
Precautions/Remarks	None					

Event name	Illegal Maximum interpolation deceleration		Event code	★57480000Hex		
Meaning	The Maximum Interpolation Deceleration value specified is out of range					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Specified wrong value of Maximum Deceleration of the robot		Correct the value of the <i>MaxDecelerationTCP</i> input variable.		Set a correct value for the <i>MaxDecelerationTCP</i> input variable.	
Attached information	None					
Precautions/Remarks	None					

Event name	Maximum interpolation velocity over-passed		Event code	★645A0000Hex		
Meaning	The commanded velocity of the default TCP overpasses the maximum robot velocity set in MC_SetKinTransform.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	During instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFaultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Wrong specified trajectory data of current instruction.		Correct the trajectory data so that the command velocity is lower than the <i>MaxVelocityTCP</i> input variable.		Set a correct trajectory data so that the command velocity is lower than the <i>MaxVelocityTCP</i> input variable.	
	Specified value for <i>MaxVelocityTCP</i> input variable of MC_SetKinTransform is wrong or too low.		Correct the maximum velocity of the <i>MaxVelocityTCP</i> input variable.		Set a correct value for the <i>MaxVelocityTCP</i> input variable.	
Attached information	None					
Precautions/Remarks	None					

Event name	Maximum interpolation acceleration over-passed		Event code	★645B0000Hex		
Meaning	The commanded acceleration of the default TCP overpasses the maximum robot acceleration set in MC_SetKinTransform.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	During instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Wrong specified trajectory data of current instruction.		Correct the trajectory data so that the command acceleration is lower than the <i>MaxAccelerationTCP</i> input variable.		Set a correct trajectory data so that the command acceleration is lower than the <i>MaxAccelerationTCP</i> input variable.	
	Specified value for <i>MaxAccelerationTCP</i> input variable of MC_SetKinTransform is wrong or too low.		Correct the maximum velocity of the <i>MaxAccelerationTCP</i> input variable.		Set a correct value for the <i>MaxAccelerationTCP</i> input variable.	
Attached information	None					
Precautions/Remarks	None					

Event name	Maximum interpolation deceleration over-passed		Event code	★645C0000Hex		
Meaning	The commanded deceleration of the default TCP overpasses the maximum robot deceleration set in MC_SetKinTransform.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	During instruction execution
Error attributes	Level	Minor fault	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Operation is not possible for relevant axes group. Relevant axes group stops immediately if it is in motion.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].MFAultLvl.Active		BOOL		Axes Group Minor Fault Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	Wrong specified trajectory data of current instruction.		Correct the trajectory data so that the command deceleration is lower than the <i>MaxDecelerationTCP</i> input variable.		Set a correct trajectory data so that the command deceleration is lower than the <i>MaxDecelerationTCP</i> input variable.	
	Specified value for <i>MaxDecelerationTCP</i> input variable of MC_SetKinTransform is wrong or too low.		Correct the maximum velocity of the <i>MaxDecelerationTCP</i> input variable.		Set a correct value for the <i>MaxDecelerationTCP</i> input variable.	
Attached information	None					
Precautions/Remarks	None					

Event name	Started the Transition by correcting TransitionParameters		Event code	★94230000Hex		
Meaning	Started the Transition by correcting TransitionParameters.					
Source	Motion Control Function Module		Source details	Axes group	Detection timing	At multi-execution of instructions
Error attributes	Level	--	Recovery	Error reset	Log category	System
Effects	User program	Continues	Operation	Not affected.		
System-defined variables	Variable		Data type		Name	
	_MC_GRP[*].Obsr.Active		BOOL		Axes Group Observation Occurrence	
Cause and correction	Assumed cause		Correction		Prevention	
	When Multi-execution, current instruction is already over the set <i>TransitionParameter</i> of buffered instruction.		Correct <i>TransitionParameter</i> as less than the time or distance of current instruction.		Set a correct <i>TransitionParameter</i> as less than the time or distance of current instruction.	
			Correct the timing of multi-execution as the remaining time or distance of current instruction is greater than the <i>TransitionParameter</i> .		Multi-execute at the timing as the remaining time or distance of current instruction is greater than the <i>TransitionParameter</i> .	
Attached information	None					
Precautions/Remarks	None					

Terms and Conditions of Sale

1. **Offer; Acceptance.** These terms and conditions (these "**Terms**") are deemed part of all quotes, agreements, purchase orders, acknowledgments, price lists, catalogs, manuals, brochures and other documents, whether electronic or in writing, relating to the sale of products or services (collectively, the "**Products**") by Omron Electronics LLC and its subsidiary companies ("**Omron**"). Omron objects to any terms or conditions proposed in Buyer's purchase order or other documents which are inconsistent with, or in addition to, these Terms.
2. **Prices; Payment Terms.** All prices stated are current, subject to change without notice by Omron. Omron reserves the right to increase or decrease prices on any unshipped portions of outstanding orders. Payments for Products are due net 30 days unless otherwise stated in the invoice.
3. **Discounts.** Cash discounts, if any, will apply only on the net amount of invoices sent to Buyer after deducting transportation charges, taxes and duties, and will be allowed only if (i) the invoice is paid according to Omron's payment terms and (ii) Buyer has no past due amounts.
4. **Interest.** Omron, at its option, may charge Buyer 1-1/2% interest per month or the maximum legal rate, whichever is less, on any balance not paid within the stated terms.
5. **Orders.** Omron will accept no order less than \$200 net billing.
6. **Governmental Approvals.** Buyer shall be responsible for, and shall bear all costs involved in, obtaining any government approvals required for the importation or sale of the Products.
7. **Taxes.** All taxes, duties and other governmental charges (other than general real property and income taxes), including any interest or penalties thereon, imposed directly or indirectly on Omron or required to be collected directly or indirectly by Omron for the manufacture, production, sale, delivery, importation, consumption or use of the Products sold hereunder (including customs duties and sales, excise, use, turnover and license taxes) shall be charged to and remitted by Buyer to Omron.
8. **Financial.** If the financial position of Buyer at any time becomes unsatisfactory to Omron, Omron reserves the right to stop shipments or require satisfactory security or payment in advance. If Buyer fails to make payment or otherwise comply with these Terms or any related agreement, Omron may (without liability and in addition to other remedies) cancel any unshipped portion of Products sold hereunder and stop any Products in transit until Buyer pays all amounts, including amounts payable hereunder, whether or not then due, which are owing to it by Buyer. Buyer shall in any event remain liable for all unpaid accounts.
9. **Cancellation; Etc.** Orders are not subject to rescheduling or cancellation unless Buyer indemnifies Omron against all related costs or expenses.
10. **Force Majeure.** Omron shall not be liable for any delay or failure in delivery resulting from causes beyond its control, including earthquakes, fires, floods, strikes or other labor disputes, shortage of labor or materials, accidents to machinery, acts of sabotage, riots, delay in or lack of transportation or the requirements of any government authority.
11. **Shipping; Delivery.** Unless otherwise expressly agreed in writing by Omron:
 - a. Shipments shall be by a carrier selected by Omron; Omron will not drop ship except in "break down" situations.
 - b. Such carrier shall act as the agent of Buyer and delivery to such carrier shall constitute delivery to Buyer;
 - c. All sales and shipments of Products shall be FOB shipping point (unless otherwise stated in writing by Omron), at which point title and risk of loss shall pass from Omron to Buyer; provided that Omron shall retain a security interest in the Products until the full purchase price is paid;
 - d. Delivery and shipping dates are estimates only; and
 - e. Omron will package Products as it deems proper for protection against normal handling and extra charges apply to special conditions.
12. **Claims.** Any claim by Buyer against Omron for shortage or damage to the Products occurring before delivery to the carrier must be presented in writing to Omron within 30 days of receipt of shipment and include the original transportation bill signed by the carrier noting that the carrier received the Products from Omron in the condition claimed.
13. **Warranties.** (a) **Exclusive Warranty.** Omron's exclusive warranty is that the Products will be free from defects in materials and workmanship for a period of twelve months from the date of sale by Omron (or such other period expressed in writing by Omron). Omron disclaims all other warranties, express or implied. (b) **Limitations.** OMRON MAKES NO WARRANTY OR REPRESENTATION, EXPRESS OR IMPLIED, ABOUT NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE OF THE PRODUCTS. BUYER ACKNOWLEDGES THAT IT ALONE HAS DETERMINED THAT THE PRODUCTS WILL SUITABLY MEET THE REQUIREMENTS OF THEIR INTENDED USE. Omron further disclaims all warranties and responsibility of any type for claims or expenses based on infringement by the Products or otherwise of any intellectual property right. (c) **Buyer Remedy.** Omron's sole obligation hereunder shall be, at Omron's election, to (i) replace (in the form originally shipped with Buyer responsible for labor charges for removal or replacement thereof) the non-complying Product, (ii) repair the non-complying Product, or (iii) repay or credit Buyer an amount equal to the purchase price of the non-complying Product; provided that in no event shall Omron be responsible for warranty, repair, indemnity or any other claims or expenses regarding the Products unless Omron's analysis confirms that the Products were properly handled, stored, installed and maintained and not subject to contamination, abuse, misuse or inappropriate modification. Return of any Products by Buyer must be approved in writing by Omron before shipment. Omron Companies shall not be liable for the suitability or unsuitability or the results from the use of Products in combination with any electrical or electronic components, circuits, system assemblies or any other materials or substances or environments. Any advice, recommendations or information given orally or in writing, are not to be construed as an amendment or addition to the above warranty. See <http://www.omron247.com> or contact your Omron representative for published information.
14. **Limitation on Liability; Etc.** OMRON COMPANIES SHALL NOT BE LIABLE FOR SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, LOSS OF PROFITS OR PRODUCTION OR COMMERCIAL LOSS IN ANY WAY CONNECTED WITH THE PRODUCTS, WHETHER SUCH CLAIM IS BASED IN CONTRACT, WARRANTY, NEGLIGENCE OR STRICT LIABILITY. Further, in no event shall liability of Omron Companies exceed the individual price of the Product on which liability is asserted.
15. **Indemnities.** Buyer shall indemnify and hold harmless Omron Companies and their employees from and against all liabilities, losses, claims, costs and expenses (including attorney's fees and expenses) related to any claim, investigation, litigation or proceeding (whether or not Omron is a party) which arises or is alleged to arise from Buyer's acts or omissions under these Terms or in any way with respect to the Products. Without limiting the foregoing, Buyer (at its own expense) shall indemnify and hold harmless Omron and defend or settle any action brought against such Companies to the extent based on a claim that any Product made to Buyer specifications infringed intellectual property rights of another party.
16. **Property; Confidentiality.** Any intellectual property in the Products is the exclusive property of Omron Companies and Buyer shall not attempt to duplicate it in any way without the written permission of Omron. Notwithstanding any charges to Buyer for engineering or tooling, all engineering and tooling shall remain the exclusive property of Omron. All information and materials supplied by Omron to Buyer relating to the Products are confidential and proprietary, and Buyer shall limit distribution thereof to its trusted employees and strictly prevent disclosure to any third party.
17. **Export Controls.** Buyer shall comply with all applicable laws, regulations and licenses regarding (i) export of products or information; (ii) sale of products to "forbidden" or other proscribed persons; and (iii) disclosure to non-citizens of regulated technology or information.
18. **Miscellaneous.** (a) **Waiver.** No failure or delay by Omron in exercising any right and no course of dealing between Buyer and Omron shall operate as a waiver of rights by Omron. (b) **Assignment.** Buyer may not assign its rights hereunder without Omron's written consent. (c) **Law.** These Terms are governed by the law of the jurisdiction of the home office of the Omron company from which Buyer is purchasing the Products (without regard to conflict of law principles). (d) **Amendment.** These Terms constitute the entire agreement between Buyer and Omron relating to the Products, and no provision may be changed or waived unless in writing signed by the parties. (e) **Severability.** If any provision hereof is rendered ineffective or invalid, such provision shall not invalidate any other provision. (f) **Setoff.** Buyer shall have no right to set off any amounts against the amount owing in respect of this invoice. (g) **Definitions.** As used herein, "including" means "including without limitation"; and "Omron Companies" (or similar words) mean Omron Corporation and any direct or indirect subsidiary or affiliate thereof.

Certain Precautions on Specifications and Use

1. **Suitability of Use.** Omron Companies shall not be responsible for conformity with any standards, codes or regulations which apply to the combination of the Product in the Buyer's application or use of the Product. At Buyer's request, Omron will provide applicable third party certification documents identifying ratings and limitations of use which apply to the Product. This information by itself is not sufficient for a complete determination of the suitability of the Product in combination with the end product, machine, system, or other application or use. Buyer shall be solely responsible for determining appropriateness of the particular Product with respect to Buyer's application, product or system. Buyer shall take application responsibility in all cases but the following is a non-exhaustive list of applications for which particular attention must be given: (i) Outdoor use, uses involving potential chemical contamination or electrical interference, or conditions or uses not described in this document. (ii) Use in consumer products or any use in significant quantities. (iii) Energy control systems, combustion systems, railroad systems, aviation systems, medical equipment, amusement machines, vehicles, safety equipment, and installations subject to separate industry or government regulations. (iv) Systems, machines and equipment that could present a risk to life or property. Please know and observe all prohibitions of use applicable to this Product. NEVER USE THE PRODUCT FOR AN APPLICATION INVOLVING SERIOUS RISK TO LIFE OR PROPERTY OR IN LARGE QUANTITIES WITHOUT ENSURING THAT THE SYSTEM AS A WHOLE HAS BEEN DESIGNED TO ADDRESS THE RISKS, AND THAT THE OMRON'S PRODUCT IS PROPERLY RATED AND INSTALLED FOR THE INTENDED USE WITHIN THE OVERALL EQUIPMENT OR SYSTEM.
2. **Programmable Products.** Omron Companies shall not be responsible for the user's programming of a programmable Product, or any consequence thereof.
3. **Performance Data.** Data presented in Omron Company websites, catalogs and other materials is provided as a guide for the user in determining suitability and does not constitute a warranty. It may represent the result of Omron's test conditions, and the user must correlate it to actual application requirements. Actual performance is subject to the Omron's Warranty and Limitations of Liability.
4. **Change in Specifications.** Product specifications and accessories may be changed at any time based on improvements and other reasons. It is our practice to change part numbers when published ratings or features are changed, or when significant construction changes are made. However, some specifications of the Product may be changed without any notice. When in doubt, special part numbers may be assigned to fix or establish key specifications for your application. Please consult with your Omron's representative at any time to confirm actual specifications of purchased Product.
5. **Errors and Omissions.** Information presented by Omron Companies has been checked and is believed to be accurate; however, no responsibility is assumed for clerical, typographical or proofreading errors or omissions.

OMRON AUTOMATION AND SAFETY • THE AMERICAS HEADQUARTERS • Chicago, IL USA • 847.843.7900 • 800.556.6766 • www.omron247.com

OMRON CANADA, INC. • HEAD OFFICE

Toronto, ON, Canada • 416.286.6465 • 866.986.6766 • www.omron247.com

OMRON ELECTRONICS DE MEXICO • HEAD OFFICE

México DF • 52.55.59.01.43.00 • 01-800-226-6766 • mela@omron.com

OMRON ELECTRONICS DE MEXICO • SALES OFFICE

Apodaca, N.L. • 52.81.11.56.99.20 • 01-800-226-6766 • mela@omron.com

OMRON ELETRÔNICA DO BRASIL LTDA • HEAD OFFICE

São Paulo, SP, Brasil • 55.11.2101.6300 • www.omron.com.br

OMRON ARGENTINA • SALES OFFICE

Cono Sur • 54.11.4783.5300

OMRON CHILE • SALES OFFICE

Santiago • 56.9.9917.3920

OTHER OMRON LATIN AMERICA SALES

54.11.4783.5300

OMRON EUROPE B.V. • Wegalaan 67-69, NL-2132 JD, Hoofddorp, The Netherlands. • +31 (0) 23 568 13 00 • www.industrial.omron.eu

Authorized Distributor:

Automation Control Systems

- Machine Automation Controllers (MAC) • Programmable Controllers (PLC)
- Operator interfaces (HMI) • Distributed I/O • Software

Drives & Motion Controls

- Servo & AC Drives • Motion Controllers & Encoders

Temperature & Process Controllers

- Single and Multi-loop Controllers

Sensors & Vision

- Proximity Sensors • Photoelectric Sensors • Fiber-Optic Sensors
- Amplified Photomicrosensors • Measurement Sensors
- Ultrasonic Sensors • Vision Sensors

Industrial Components

- RFID/Code Readers • Relays • Pushbuttons & Indicators
- Limit and Basic Switches • Timers • Counters • Metering Devices
- Power Supplies

Safety

- Laser Scanners • Safety Mats • Edges and Bumpers • Programmable Safety Controllers • Light Curtains • Safety Relays • Safety Interlock Switches