COSIMIR® · Movemaster Command

MOVEMASTER COMMAND

Overview

COSIMIR® · Movemaster Command

Overview of commands

This section gives an overview of a wealth of commands provided the MOVEMASTER command method. Commands are classified into seven types to the following. You should use appropriate commands according to your application.

(1) Position and motion control commands

These commands are concerned with definition of position and coordinates as well as assignment of interpolation, speed, timer, tool, palette, etc.

(2) Program control commands

These commands are concerned with conditional branching, repetitive operation, interrupting of signals, starting and counter operation, etc.

(3) Hand control commands

These commands are concerned with opening/closing action of For the optional motor-operated hand, you can control gripping force by parameter setting.

(4) I/O control commands

These commands are concerned with input/output control of external I/O. Both single and parallel bits can be handled enabling the logical operation in the internal register.

(5) Operation, substitute, exchange command

These commands are concerned with mathematical operations.

(6) Communication command (Using RS232C)

These commands transfer internal information of the robot such counters, positions, program list, input/output signal status, parameters to personal computer.

(7) Other command

These commands are concerned with setting of parameters, selecting of programs, resetting of alarms, and describing of comments.

COSIMIR® · Movemaster Command

Explanation of command

All commands are explained in alphabetical order to the following.

(1) A viewpoint of command explanation

This shows the meaning of title described in the explanation of each command.

*** Symbol**

The command that has * mark can not be described in program with line number. (If done, it will become alarm.) You should execute the commands directly by personal computer or teaching box.

Function

Shows the brief function of command.

Input Format

Shows the argument of the command entry.

< > indicates the command parameter.[ ] indicates the parameter that can be omitted.

Term

Shows the meaning and limited range of command argument.

Explanation

Explains detailed function and precautions.

Relating Parameters

Shows the parameters that have relationship with the command.

Sample program

Gives a typical program using MOVEMASTER command BASIC language.

(2) Line

Line consists of line number and a command sentence. The length of one line contains 120 characters in maximum.

(3) Position number

The range of position numbers is 1-999. When programs are different from each other, positions from 1-900 are memorized as individual data even if they have the number. On the other hand, positions from 901-999 are memorized as common.

Further, you can appoint position number indirectly by the value of counter.

An example:

10 SC 21 ,10;Sets value 10 in counter 21.

20 MO @21;Moves to position 10, the value of counter

Relating Commands: CF, HE, MA, MC, MO, MR, MRA, MS, MT, MTS, PC, PL, PR, PX and SF.

(4) Counter number

The range of counter number is 1-99. When program is different from each other, counters from 1-90 are memorized as individual data even if they have the number. On the other hand, counters from 91-99 are memorized as common. Further, you can appoint number indirectly by the value of counter.

An example:

10 SC 21 ,10;Sets value10 in counter 21.

20 IC •—21;Adds one to counter 10, the value of 21.

Relating Commands CL, CP, CR, DC, IC, OC and SC.

(5) Character string numbers

By using a character string number, character strings (max. 120 alphanumeric or symbol characters) can be communicated to the external device over the serial channel.

The character string number can be specified from $1 $99. If the program differs between $1 and $90, these will be registered as separate data even if the number the same. On the other hand, $91 to $99 are common all programs. Depending on the counter number value (details), the character string number cannot be specified indirectly.

An example:

10 SC $1, "ABC";Character string "ABC" is set in character string number 1

20 CP $1 ;The details of character string number 1 are

; in the character string register

30 CL $2 ;The details of the character string register are set in

; character string number 2.

Relating commands CL, CP, CR, INP, LG, NE, EQ, SM.

(6) Direct execution

If you transfer command to the controller without line number by personal computer or teaching box, it will executed instantaneously. (Not to be programmed.) Especially hold with care in case executing moving command. Conditional branching commands are only in the program with line number and not be executed directly. (Example ,s,a,,d,p)

You can execute HLT, PRN and RS-232C communication command even if the program is running.

(7) Internal register

The data taken from external I/O are memorized in so called "internal register", then used for conditional branching, comparison, logical bit operation, counter setting, and etc. When you carry out conditional branching on the basis of counter value, you use this internal register too.

Relating Commands AN, CL, CP, DR, EQ, ID, NE, OR, SM, TB and XO.

(8) Character string register

The character string led in to the external device over the serial channel is stored in the memory called the "character string register". It is then used for making settings of the character string comparison or string number to jump conditions.

Relating commands CL, CP, INP, LG, NE, EQ, SM.

(9) Non-executing commands

These commands are provided for the purpose of compatibility with MOVEMASTER M2 series. If you these commands in MOVEMASTER Super series, no operation and no alarm will occur.

Relating Commands IN, ML, NR, OT, SL, TR and WR.

(10) Internal resister and related command. (Reference)

It shows it as follows about interior resister, counter, input port and flow of treatment of command

Character string register and related commands (Reference)

The flow of the character string register, character and related command process is shown in the manual.

COSIMIR® · Movemaster Command

Position and motion control commands

These commands are concerned with definition of position and coordinates as well as assignment of interpolation, speed, timer, tool, palette, etc.

Command	Function
ADL	(Acceleration/Deceleration)
CF	(Change Figure)
DJ	(Draw Joint)
DP	(Decrement Position)
DS	(Draw Straight)
DW	(Draw)
HE	(Here)
HO	(Home)
IP	(Increment Position)
JRC	(Joint Roll Change)
MA	(Move Approach)
MC	(Move Continuous)
MJ	(Move Joint)
ML	(Move Linear)
MO	(Move)

MP	(Move Position)
MPB	(Move Playback)
MPC	(Move Playback
MR	(Move R)
MRA	(Move R A)
MS	(Move Straight)
MT	(Move Tool)
MTS	(Move Tool Straight)
NT	(Nest)
OG	(Origin)
OVR	(Override)
PA	(Pallet Assign)
PC *	(Position Clear)
PD	(Position Define)
PL	(Position Load)
PT	(Pallet)
PW	(Pulse Wait)
PX	(Position Exchange)
SD	(Speed Define)
SF	(Shift)
SL	(Set Limit)
SP	(Speed)
TI	(Timer)
TL	(Tool)
TLM	(Tool Matrix)
WRM	(Wait Robot Motion)

COSIMIR® · Movemaster Command

Program control commands

These commands are concerned with conditional branching, repetitive operation, interrupting of signals, starting and stopping, counter operation, etc.

Command	Function
CL	(Counter Load)
CP	(Compare Counter)
DA	(Disable Act)

DC	(Decrement Counter)
DL *	(Delete Line)
EA	(Enable Act)
ED	(End)
EQ	(Equal)
GS	(Go Sub)
GT	(Go To)
HLT	(Halt)
IC	(Increment Counter)
LG	(If Larger)
LR *	(Line Read)
NE	(If Not Equal)
NW *	(New)
NX	(Next)
RC	(Repeat Cycle)
RN *	(Run)
RT	(Return)
SC	(Set Counter)
SM	(If Smaller)

COSIMIR® · Movemaster Command

Communication commands (using RS232C)

These commands transfer internal information of the robot such as counters, positions, program list, input/output signal status, parameters to personal computer.

Command	Function
CR	(Counter Read)
DR	(Data Read)
ER *	(Error Read)
LR *	(Line Read)
NR	(Nest Read)
PMR	(Parameter Read)
PR	(Position Read)
QN	(Question Number)
STR *	(Step Read)
VR	(Version Read)

WH	(Where)
WT	(What Tool)
WTM	(What Tool Matrix)

COSIMIR® · Movemaster Command

Hand control commands

These commands are concerned with opening/closing action of For the optional motor-operated hand, you can control gripping force by parameter setting.

Command	Function
GC	(Grip Close)
GF	(Grip Flag)
GO	(Grip Open)
GP	(Grip Pressure)

COSIMIR® · Movemaster Command

Operation, substitute, exchange commands

These commands are concerned with mathematical operations.

Command	Function
ADD	(Add)
AN	(And)
DC	(Decrement Counter)
DIV	(Divide)
IC	(Increment Counter)
MUL	(Multiply)
SC	(Set Counter)
SUB	(Subtract)
XO	(Exclusive Or)

COSIMIR® · Movemaster Command

I/O control commands

These commands are concerned with input/output control of external I/O. Both single and parallel bits can be handled enabling the logical operation in the internal register.

Command	Function
AN	(And)
ID	(Input Direct)
IN	(Input)
OB	(Output Bit)
OC	(Output Counter)
OD	(Output Direct)
OR	(Or)
OT	(Output)
TB	(Test Bit)
TBD	(Test Bit Direct)
XO	(Exclusive Or)

COSIMIR® · Movemaster Command

Other commands

These commands are concerned with setting of parameters, selecting of programs, resetting of alarms, and describing of comments.

Command	Function
INP	(Input)
N *	(Number)
OPN	(Open)
PMW	(Parameter Write)

PRN *	(Print)
RS *	(Reset)
TR *	(Transfer)
WR *	(Write)
'	(Comment)

COSIMIR® · Movemaster Command

Alphabetic command list

Command	Function
ADL	(Acceleration/Deceleration)
ADD	(Add)
ADL	(Acceleration/deceleration)
AN	(And)
CF	(Change Figure)
CL	(Counter Load)
CP	(Compare Counter)
CR	(Counter Read)
DA	(Disable Act)
DC	(Decrement Counter)
DJ	(Draw Joint)
DIV	(Divide)
DL *	(Delete Line)
DP	(Decrement Position)
DR	(Data Read)
DS	(Draw Straight)
DW	(Draw)
EA	(Enable Act)
ED	(End)
EQ	(Equal)
ER *	(Error Read)
GC	(Grip Close)
GF	(Grip Flag)
GO	(Grip Open)
GP	(Grip Pressure)
GS	(Go Sub)
GT	(Go To)

HE	(Here)
HLT	(Halt)
HO	(Home)
IC	(Increment Counter)
ID	(Input Direct)
IN	(Input)
INP	(Input)
IP	(Increment Position)
JRC	(Joint Roll Change)
LG	(If Larger)
LR *	(Line Read)
MA	(Move Approach)
MC	(Move Continuous)
MJ	(Move Joint)
ML	(Move Linear)
MO	(Move)
MP	(Move Position)
MPB	(Move Playback)
MPC	(Move Playback
MR	(Move R)
MRA	(Move R A)
MS	(Move Straight)
MT	(Move Tool)
MTS	(Move Tool Straight)
MUL	(Multiply)
N *	(Number)
NE	(If Not Equal)
NR	(Nest Read)
NT	(Nest)
NW *	(New)
NX	(Next)
OB	(Output Bit)
OC	(Output Counter)
OD	(Output Direct)
OG	(Origin)
OPN	(Open)
OR	(Or)
OT	(Output)
OVR	(Override)
PA	(Pallet Assign)
PC *	(Position Clear)
PD	(Position Define)
PL	(Position Load)

PMR	(Parameter Read)
PMW	(Parameter Write)
PR	(Position Read)
PRN *	(Print)
PT	(Pallet)
PW	(Pulse Wait)
PX	(Position Exchange)
QN	(Question Number)
RC	(Repeat Cycle)
RN *	(Run)
RS *	(Reset)
RT	(Return)
SC	(Set Counter)
SD	(Speed Define)
SF	(Shift)
SL	(Set Limit)
SM	(If Smaller)
SP	(Speed)
STR *	(Step Read)
SUB	(Subtract)
TB	(Test Bit)
TBD	(Test Bit Direct)
TI	(Timer)
TL	(Tool)
TLM	(Tool Matrix)
TR *	(Transfer)
VR	(Version Read)
WH	(Where)
WR *	(Write)
WRML	(Wait Robot Motion)
WTM	(What Tool Matrix)
XO	(Exclusive Or)
'	(Comment)

COSIMIR® · Movemaster Command

ADD (Add)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

Add the direct value or the contents of the specified counter to the internal register value.

Input Format

ADD <Operation data>

Term

<Operation data>

Describes the data to be operated as a numeric value or counter No. with @.
-32768 <= numeric value (decimal) <= 32767
&8000 <= numeric value (hexadecimal) <=
&7FFF
@1 <= Counter No. <= @99

Explanation

(1) Designate the operation data setting as a numeric value or counter No.

When designating with a numeric value, use a decimal hexadecimal value. When using a hexadecimal, add a to the head of the operation data.

When setting with a counter No. add a "@" to the head the counter No.

The contents of the set counter No. will be used as the operation data.

(2) The operation results are stored in the internal register, so operation, comparison and reading, etc., of the operation results can be carried out with the commands.

(Refer to SUB, MUL, DIV, EQ, NE, LG, SM, CL, DR and OR commands)

Sample program (MOVEMASTER Command)

```
10 CP 1          'Stores counter No. 1 value in internal register

20 ADD @2        'Add counter No.2 to the contents internal register
value.
```

30 CL 3 'Sets internal register value in counter No. 3

'(Counter No. 3 = counter No. 1 + counter No. 2)

40 CP 1 'Stores counter No. 1 value in internal register

50 ADD 15 'Add 15 to the contents of the internal register value.

60 CL 4 'Sets internal register value in counter No. 4

'(Counter No. 4 = counter No. 1 + 15)

RV-E-Robot (RV-E2, RV-E3J), RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-E, RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

ADL (Acceleration/deceleration)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Sets the acceleration/deceleration time for the robot additional axis.

Input Format

ADL [<Acceleration time>] [, <Deceleration time>] [, <Valid devices>]

Term

<Acceleration time>

Sets the acceleration time for reaching the motor rated speed.
0 <= Acceleration time <= 2000 (msec)

<Deceleration time>

Sets the deceleration time for reaching the motor rated speed.
0 <= Deceleration time <= 2000 (msec)

<Valid devices>

Specifies the device for which acceleration/deceleration will be valid.
0: Robot (When omitted)
1: First additional axis
2: Second additional axis

Explanation

(1) This is an additional command for the additional system.

(2) Sets the acceleration/deceleration time for each device.

(3) The acceleration time is the time it takes from a stopped position to the motor rated speed.

(4) The deceleration time is the time it rakes from the motor rated speed to a stopped position.

(5) The acceleration time and the deceleration time the same value as that for the SD command. Consequently, the robot,s acceleration/deceleration times can be set with the ADL and SD commands.

(6) With this command, the value of the parameter does not change.

(7) When omitting an argument, all the other settings still valid.

(8) When the power is turned on, it will have the initial value of the parameter AXAD.

(9) When a non-existing device is designated, it will not execute.

Related parameters

AXAD

Sample program (MOVEMASTER Command)

```
10 MO 1                'Move to the position 1
```

20 ADL 100,100,1 'Set the acceleration time for the first additional axis to 0.1sec.

30 MO 2 'Move to the position 2

RV-E-Robot (RV-E2, RV-E3J), RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-E, RV-M1 and RV-M2 series without additional axis.

COSIMIR® · Movemaster Command

AN (And)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.), RV-M2

Function:

ANDs the specified value with the internal register, stores the result to the internal register.

Input Format

AN <operation data>

Term

<Operation data>

Specify the data to be operated.
-32768 <= operation data (decimal) <= 32767
& 8000 <= operation data (hexadecimal) <= &7FFF

Explanation

(1) Specify the data to be operated in decimal or hexadecimal.

Any hexadecimal value must be headed by "&".

(2) The operation result is stored into the internal

register and can be changed, compared or read by relevant commands. (See the EQ, NE, LG, SM, CL and OR commands)

(3) Execution of the AN command after the input commands (ID or IN) allows receiving of only the bits of the parallel input data fetched from the device

Sample program (MOVEMASTER Command)

10 ID 'Fetches data from external input port.

20 AN &000F 'Receives four lower bits only

30 CL 12 'Loads above data into counter 12

40 EQ 8,200 'Jumps to line 200 if above data is equal to 8.

50 ED 'Ends program.

:

200 MO 99 'Moves to position 99.

RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 series.

COSIMIR® · Movemaster Command

CF (Change Figure)

RV-2AJ, RV-E-Robot with 5 axes (RV-E3J, RV-E5NM, etc.)

Function:

Changes the attitude data of the robot at the specified position.

Input Format

CF <Position number> [, [<R/L>] [, [<A/B>]]]

Term

<Position number>

Specify position number changing attitude data in integer value.
1 <= Position number <= 999

<R/L>

Appoint structure flag of the robot to Right Left.
R : Right (Default)
L : Left

<A/B>

Appoint structure flag of the robot to Above or Below.
A :Above (Default)
B :Below

Explanation

(1) Changes the attitude data of the robot at the specified position.

(2) Does not effect the coordinate data of the specified position and the open/close state of the hand.

(3) Even if the position is reachable before executing command, there is case becoming not reachable in the waist and shoulder joint after execution.

(4) You can confirm the attitude of the specified by PR command.

Sample program (MOVEMASTER Command)

```
10 PD 1,-280,-50,970,-10,-20,L,A,        'Defines the position 1.(Left, Above)

20 MO 1                                  'Moves to position 1.

30 CF 1,R,A                              'Changes the attitude of position 1.(Right, Above)

40 MO 1                                  'Moves to position 1.
```

RV-1A, RV-E-Robot with 6 axes(RV-E2, RV-E4N, etc.)

Function:

Changes the attitude data of the robot at the specified position.

Input Format

CF <Position number> [, [<R/L>] [, [<A/B>] [, [<N/F>]]]]

Term

<Position number>

Specify position number changing attitude data in integer value.
1 <= Position number <= 999

<R/L>

Appoint structure flag of the robot to Right Left.
R : Right (Default)
L : Left

<A/B>

Appoint structure flag of the robot to Above or Below.
A :Above (Default)
B :Below

<N/F>

Appoint structure flag of the robot to Non or Flip.
N : Non flip(Default)
F : Flip

Explanation

(1) Changes the attitude data of the robot at the specified position.

(2) Does not effect the coordinate data of the specified position and the open/close state of the hand.

(3) Even if the position is reachable before executing command, there is case becoming not reachable in the waist and shoulder joint after execution.

(4) You can confirm the attitude of the specified by PR command.

Sample program (MOVEMASTER Command)

10 PD1,530,0,470,10,135,-10,R,A,N 'Defines the position 1.

'(Right, Non flip)

20 MO 1 'Moves to position 1.

30 CF 1,R,A 'Changes the attitude of position 1.

'(Right, Flip)

40 MO 1 'Moves to position 1.

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

CL (Counter Load)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

The internal register value is set in the counter with specified number. The character string register details are set in the character string with the specified value.

Input Format

CL <counter number/character string number>

Term

<Counter number>

Specify counter number in numeric value.
1 <= counter number <= 99

<Character string number>

Specify character string number in value which "$" is added to the head.
$1 <= character string number <= $99

Explanation

<When counter number is specified>

(1) Sets the data fetched from the input port to the specified counter. Hence, the CL commands must be executed after the input command. (See ID)

(2) Since the data from the input port is treated as signed, the data set to the counter is signed. (Take between -32768 and 32767.)

(3) Used to specify the number of job sequences and counter value at palletizing from an external device, such as a programmable controller. At this case, any of the logical operation commands (see AN, OR or XO) may be used as necessary.

(4) Execution of the CL command after the CP allows the counter data to be transferred.

(5) The counter value can be changed, compared or by the relevant commands.

(See IC, DC, SC, CP and CR commands.)

<When character string number is specified>

(1) The data lead in from the character string register set in the specified character string. Thus, this must be executed after the character string register setting command (refer to CP, INP commands) is executed.

(2) This is used to set and confirm the work details or work results from an external device such as a sequencer. Use the compare command are required at this time. (Refer to EQ, LG, NE, SM commands.)

(3) The character string register value is set in the specified character string by this command, so if this

command is executed after the CP command, the data can be copied between character strings.

(4) Operation, comparison and reading of the character string are possible by using the related commands.

(See CP, CR, EQ, INP, LG, NE, SC, SM commands.)

Sample program (MOVEMASTER Command)

10 ID 'Fetches data from external input port.

20 CL 25 'Sets above data to counter 25.

30 CP 11 'Sets data of counter 11 to internal register.

40 CL 21 'Sets data of internal register to counter 21.

50 SC $5,"ABC" 'Set character string "ABC" in string number 5

60 CP $5 'Set details of character string 5 in the character 'string register

70 CL $10 'Set the details of the character register in character string number 10

RV-E-Robot (RV-E2, RV-E3J, etc.), RV-M2

Function:

The internal register value is set in the counter with specified number.

Input Format

CL <counter number>

Term

<Counter number>

Specify counter number in numeric value.
1 <= counter number <= 99

Explanation

(1) Sets the data fetched from the input port to the specified counter. Hence, the CL commands must be

executed after the input command. (See ID)

(2) Since the data from the input port is treated as signed, the data set to the counter is signed. (Take between -32768 and 32767.)

(3) Used to specify the number of job sequences and counter value at palletizing from an external device, such as a programmable controller. At this case, any of the logical operation commands (see AN, OR or XO) may be used as necessary.

(4) Execution of the CL command after the CP allows the counter data to be transferred.

(5) The counter value can be changed, compared or by the relevant commands.

(See IC, DC, SC, CP and CR commands.)

Sample program (MOVEMASTER Command)

```
10 ID      'Fetches data from external input port.
20 CL 25   'Sets above data to counter 25.
30 CP 11   'Sets data of counter 11 to internal register.
40 CL 21   'Sets data of internal register to counter 21.
```

RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 series.

COSIMIR® · Movemaster Command

' (Comment)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Allows the programmer to write a comment.

Input Format

'[<string consisting of up to 120 alphanumeric including line number and '(apostrophe)>]

Explanation

(1) You can describe up to 120 alphanumeric characters including line number and ' (apostrophe).

(2) Use it to describe the name and date on the generated program or to mark a subroutine. Comments are helpful to check back the program as the LR (Line Read) command is used.

(3) The system ignores comments as it processes its commands.

(4) If the number of characters exceeds the limit, the whole excess is ignored.

Sample program (MOVEMASTER Command)

```
10 '* * * * * * * * * * * * * * * * * * * * * * * * * * * * *

20 'Sample Program

30 'Date: 93-10-01

40 'Programmed by Mitsubishi

50 '* * * * * * * * * * * * * * * * * * * * * * * * * * * * *

60 NT
```

RV-M2, RV-M1

Function:

Allows the programmer to write a comment.

Input Format

'[<string consisting of up to 7 alphanumeric

Explanation

(1) This command allows the programmer to write a comment consisting of up to 7 alphanumeric characters following the ' (apostrophe).

(2) Use it to describe the name and date on the generated program or to mark a subroutine. Comments are helpful to check back the program as the LR (Line Read) command is used.

(3) The system ignores comments as it processes its commands.

(4) If the number of characters exceeds 7, the whole excess is ignored.

Sample program (MOVEMASTER Command)

10 'DATE

20 '890311

COSIMIR® · Movemaster Command

CP (Compare Counter)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

The value of the specified counter is set in the internal register. The details of the specified character string set in the character string register.

Input Format

CP <counter number/character string number>

Term

<Counter number>

Specify counter number in numeric value.
1 <= counter number <= 99

<Character string number>

Specify character string number in value which "$" is added to the head.
$1 <= character string number <= $99

Explanation

<When counter number is specified>

(1) To be executed before a conditional jump command (see EQ, NE, LG and SM commands) is executed if the value in the specified counter is used for the jump. Conditional branching command carries out jump on basis of value of internal register set by the CP command.

(2) Even if the value of the specified counter has after execution of the CP command, the value of register remains intact. Accordingly when you carry out conditional branching by value of counter, need to out this command after the counter value has changed.

(3) The input instruction (see ID) uses the same internal register, meaning that the old contents of the internal register are lost when any input command is executed.

(4) The contents of the counter can be changed or read by the relevant commands.

(See SC, IC, INP, DC, CR, CL, AN, OR and XO commands.)

<When character string number is specified>

(1) When jumping the conditions due to the details of specified character string, this command must be executed before the conditions jump command (refer EQ, NE, LG and SM commands). The conditions jump command will execute comparison jumping based on value of the character string register set by the CP command.

(2) Even if the details of the specified character string change after this command is executed, the character string register value will not be affected. Thus, when executing condition jump according to the details of character string, this command must be executed after the details of the character string change.

1. Operation, comparison and reading of the string are possible by using the related commands.

(See CP, CR, EQ, INP, LG, NE, SC, SM commands.)

Sample program (MOVEMASTER Command)

```
100 IC 21             'Add 1 to the contents of counter 21
```

110 CP 21 'Sets the value of counter 21 to the internal register.

120 EQ 255,500 'If the contents of the internal equal 255,

'the program jumps to line number 500.

130 GT 100 'The program jumps to line number 100.

500 SC 21,0 'Sets value 0 to counter 21.

600 SC $5,"OK" 'Set character string "OK" in string number 5

610 CP $5 'Set details of character string number 5 in the character string register

620 EQ $10,800 'If the contents of the character register equal character string number 10, the program jumps to line number 800.

:

800 GT 100 'The program jumps to line number 100.

RV-E-Robot (RV-E2, RV-E3J, etc.), RV-M2, RV-M1

Function:

The value of the specified counter is set in the internal register.

Input Format

CP <counter number>

Term

<Counter number>

Specify counter number in numeric value.
1 <= counter number <= 99

Explanation

(1) To be executed before a conditional jump command (see EQ, NE, LG and SM commands) is executed if the value in the specified counter is used for the jump.

Conditional branching command carries out jump on the basis of value of internal register set by the CP command.

(2) Even if the value of the specified counter has after execution of the CP command, the value of register remains intact. Accordingly when you carry out conditional branching by value of counter, need to carry out this command after the counter value has changed.

(3) The input instruction (see ID) uses the same internal register, meaning that the old contents of the internal register are lost when any input command is executed.

(4) The contents of the counter can be changed or read by the relevant commands.

(See SC, IC, INP, DC, CR, CL, AN, OR and XO commands.)

Sample program (MOVEMASTER Command)

100 IC 21 'Add 1 to the contents of counter 21

110 CP 21 'Sets the value of counter 21 to the internal register.

120EQ 255,500 'If the contents of the internal equal 255, the program jumps to line number 500.

130 GT 100 'The program jumps to line number

500 SC 21,0 'Sets value 0 to counter 21.

510 GT100 'The program jumps to line number

COSIMIR® · Movemaster Command

CR (Counter Read)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

The details of the specified counter or character string are read out. (Using RS-232-C)

Input Format

CR <counter number/character string number>

Term

<Counter number>

Specify counter number in numeric value.
1 <= counter number <= 99

<Character string number>

Specify character string number in
value which "$" is added to the head.
$1 <= character string number <= $99

Explanation

(1) Outputs the contents of the specified counter from RS-232C port.

(2) The output format is in ASCII coded decimal.

(3) Because the terminator of the output data is return (Hex. 0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(4) If an undefined counter is read, the initial value of is returned. (+0 is returned in the case of "0")

(5) If a non-set character string is read, the final code's carriage return (CR: hexadecimal 0D) will be returned.

(6) The contents of the counter are battery backed the power is switched off.

Sample program (BASIC)

<When counter number is specified>

```
10 OPEN "COM1:E83" AS #1                    'Opens
RS-232C communication file by the BASIC.

20 INPUT "Counter number" ;N                'Enter
counter number.

30 INPUT "Counter data "; D                 'Enter
counter data.
```

40 PRINT #1, "SC "+STR$(N)+","+STR$(D) 'Specified data is input into the counter.

50 PRINT #1,"CR"+STR$(N) 'Transfers the data to the personal computer.

60 LINE INPUT #1,A$ 'Saves the received data to A$.

70 PRINT A$ 'Displays data to the screen.

80 END 'Program ends.

RUN 'Run the BASIC program.

Counter number 1

Counter data 100

+100

<When character string number is specified>

10 OPEN "COM1 :E83" AS #1 'Opens the RS-232C communication file by the BASIC.

20 INPUT "Character string number";N 'Enter the character string number.

30 INPUT "Character string data ";J$ 'Enter the character string data.

40 PRINT #1, "SC $"+STR$(N)+","+CHR$(&H22)
+J$+CHR$(&H22)

'Specified is input into the counter.

50 PRINT #1,"CR $"+STR$(N) 'Transfers the data to the personal computer.

60 LINE INPUT #1,A$ 'Saves the received data to A$.

70 PRINT A$ 'Displays data to the screen.

80 END 'Program ends.

RUN 'Run the program.

Counter number 1

Counter data 100

+100

RV-E-Robot (RV-E2, RV-E3J, etc.), RV-M2, RV-M1

Function:

The details of the specified counter are read out. RS-232-C)

Input Format

CR <Counter number>

Term

<Counter number>

Specify counter number in numeric value.
1 <= counter number <= 99

Explanation

(1) Outputs the contents of the specified counter from RS-232C port.

(2) The output format is in ASCII coded decimal.

(3) Because the terminator of the output data is return (Hex. 0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(4) If an undefined counter is read, the initial value of is returned. (+0 is returned in the case of "0")

(5) If a non-set character string is read, the final code's carriage return (CR: hexadecimal 0D) will be returned.

(6) The contents of the counter are battery backed the power is switched off.

Sample program (BASIC)

10 OPEN "COM1:E83" AS #1 'Opens the

232C communication file by the BASIC.

20 INPUT "Counter number" ;N 'Enter the counter number.

30 INPUT "Counter data "; D 'Enter the counter data.

40 PRINT #1, "SC "+STR$(N)+","+STR$(D) 'Specified data is input into the counter.

50 PRINT #1,"CR"+STR$(N) 'Transfers the data to the personal computer.

60 LINE INPUT #1,A$ 'Saves the received data to A$.

70 PRINT A$ 'Displays the data to the screen.

80 END 'Program

RUN 'Run the BASIC program.

Counter number 1

Counter data 100

+100

COSIMIR® · Movemaster Command

DA (Disable Act)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Disables an interrupt of the specified bit through the external input port.

Input Format

DA <input bit number>

Term

<Input bit number>

Specify the bit number to be disabled.
0 <= input bit number <= 32767
0 - 8999: Input signal interrupt
9003 - 32767: Input signal interrupt
0 - 299: General input
900 - 903: Hand input
9000: Alarm interrupt
9001: Emergency stop interrupt
9002: Communication interrupt

Explanation

(1) Clears the interruptible state of the bit defined by interrupt enable commands (see the EA commands).

(2) After the DA command has been executed, no interrupt is executed by the specified bit during execution. Note that execution of the DA command not affect the interruptible states of the other bits.

(3) To inhibit repeated interrupts by a level signal, the DA command must be executed at the beginning line to which the program jumps after the interrupt has taken place

Sample program (MOVEMASTER Command)

see EA command.

RV-M2

Function:

Disables an interrupt of the specified bit through the external input port.

Input Format

DA <input bit number>

Term

<Input bit number>

Specify the bit number to be disabled.
0 <= input bit number <= 17
16 - 17: Hand check inputs

Explanation

(1) Clears the interruptible state of the bit defined by interrupt enable commands (see the EA commands).

(2) After the DA command has been executed, no interrupt is executed by the specified bit during execution. Note that execution of the DA command not affect the interruptible states of the other bits.

(3) To inhibit repeated interrupts by a level signal, the DA command must be executed at the beginning line to which the program jumps after the interrupt has taken place

Sample program (MOVEMASTER Command)

see EA command.

RV-M1

Function:

Disables an interrupt of the specified bit through the external input port.

Input Format

DA <input bit number>

Term

<Input bit number>

Specify the bit number to be disabled.
0 <= input bit number <= 7 (15)
Figure in () is for type A16 or B16 I/O card.

Explanation

(1) Clears the interruptible state of the bit defined by interrupt enable commands (see the EA commands).

(2) After the DA command has been executed, no interrupt is executed by the specified bit during execution. Note that execution of the DA command not affect the interruptible states of the other bits.

(3) To inhibit repeated interrupts by a level signal, the DA command must be executed at the beginning line to which the program jumps after the interrupt has taken place

Sample program (MOVEMASTER Command)

see EA command.

COSIMIR® · Movemaster Command

DC (Decrement Counter)

Function:

Subtracts 1 from the value in the specified counter.

Input Format

DC <Counter number>

Term

<Counter number>

Specify counter number in numeric value.
1 <= counter number <= 99

Explanation

(1) Alarm occurs if the counter value becomes smaller than -32768.

(2) Used to count the number of workpieces and job sequences and to set the number of grid points in

(3) The contents of the counter can be changed, compared or read by the relevant command. (See SC, CP, CR, CL, AN, OR and XO commands.)

Sample program (MOVEMASTER Command)

```
10 SC 21,15    'Sets value 15 to the counter 21.

20 DC 21       'Subtracts 1 from the value in the counter 21.
```

COSIMIR® · Movemaster Command

DIV (Div)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

Divide the internal register value by the direct value or the contents of the specified counter.

Input Format

DIV <Operation data>

Term

<Operation data>

Describes the data to be operated as a numeric value or counter No. with @.
-32768 <= numeric value (decimal) <= 32767
&8000 <= numeric value (hexadecimal) <=
&7FFF
@1 <= Counter No. <= @99

Explanation

(1) Designate the operation data setting as a numeric value or counter No.

When designating with a numeric value, use a decimal hexadecimal value. When using a hexadecimal, add a to the head of the operation data.

When setting with a counter No. add a "@" to the head the counter No.

The contents of the set counter No. will be used as the operation data.

(2) The operation results are stored in the internal register, so operation, comparison and reading, etc., of the operation results can be carried out with the commands.

(Refer to SUB, ADD, MUL, EQ, NE, LG, SM, CL and DR, commands)

Sample program (MOVEMASTER Command)

10 CP1 'Stores counter No. 1 value in internal register

20 DIV @2 'Divides the internal register value by the counter No. 2.

30 CL 3 'Sets internal register value in counter No.

'(Counter No. 3 = counter No. 1/ counter 2)

40 CP 1 'Stores counter No. 1 value in internal register

50 DIV 15 'Divide the internal register value by 15.

60 CL 4 'Sets internal register value in counter No.

'(Counter No. 4 = counter No. 1 / 15)

RV-E-Robot (RV-E2, RV-E3J), RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 series.

COSIMIR® · Movemaster Command

DJ (Draw Joint)

RV-2AJ, RV-E-Robot with 5 axes (RV-E3J, RV-E5NM, etc.)

Function:

Rotates the specified joint by the specified angle from the current position. (Joint interpolation)

Input Format

DJ <joint number>, <turning angle>

Term

<Joint number>

Specify joint number that you want to
1: Waist joint
2: Shoulder joint
3: Elbow joint
4: Pitch joint
5: Roll joint

<Rotating angle>

Specify joint angle that you want to turn.

Explanation

(1) The least input increment of the turning angle is degree, e.g., specify 15.02 for 15.02 degree.

(2) The open/close state of the hand does not change before and after the movement. Alarm occurs before joint motion if any turning angle entry exceeds the robot's operational space.

Sample program (MOVEMASTER Command)

10 MO 1 'Moves to position 1.

20 DJ 1,10 'Turns the waist joint 10 degrees in the positive direction.

RV-1A, RV-E-Robot with 6 axes(RV-E2, RV-E4N, etc.)

Function:

Rotates the specified joint by the specified angle from the current position. (Joint interpolation)

Input Format

DJ <joint number>, <turning angle>

Term

<Joint number>

Specify joint number that you want to
1:Waist joint
2:Shoulder joint
3:Elbow joint
4:Twist joint
5:Pitch joint
6:Roll joint

<Rotating angle>

Specify joint angle that you want to turn.

Explanation

(1) The least input increment of the turning angle is degree, e.g., specify 15.02 for 15.02 degree.

(2) The open/close state of the hand does not change before and after the movement. Alarm occurs before joint motion if any turning angle entry exceeds the robot's operational space.

Sample program (MOVEMASTER Command)

10 MO 1 'Moves to position 1.

20 DJ1,10 'Turns the waist joint 10 degrees in the direction.

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

DL * (Delete Line)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Deletes commands of the specified line or step in the program.

Input Format

DL [<Line number (a)>] [,<Line number (b)>] [, [<step(a)>][<step(b)>]]]]

Term

<Line number(a)>

Specify the top line number that you want delete in the program.

<Line number(b)>

Specify the last line number that you want delete in the program.

<step(a)>

Specify the top step number that you want delete in the program.

<step(b)>

Specify the last step number that you want delete in the program.
1 <= Line number (a), (b) <= 9999
1 <= Step (a), (b) <= 9999
Line number (a) <= Line number (b)
Step (a) <= Step (b)

Explanation

(1) Deletes commands from step (a) to step (b), then deletes commands from line (a) to line (b). (Step (b) line (b) are included too.)

(2) When you omitted line (b), only line (a) is deleted.

(3) When you omitted step (b), only step (a) is deleted.

(4) An alarm will occur if the size relation of (a) and (b) is incorrect.

* This command can only be executed directly. It be used in a program.

Sample program (MOVEMASTER Command)

100 MO10 'Moves to position 10.

110 MO12 'Moves to position 12.

120 MO15 'Moves to position 15.

130 MO17 'Moves to position 17.

140 MO20 'Moves to position 20.

DL 110 'Deletes line number 110.

DL 120,140 'Deletes line number from 120 to 140.

RV-M2

Function:

Deletes commands of the specified line or step in the program.

Input Format

DL <Line number (a)> [,<Line number (b)>]

Term

<Line number(a)>

Specify the top line number that you want to delete in the program.

<Line number(b)>

Specify the last line number that you want to delete in the program.
1 <= Line number (a), (b) <= 3584
Line number (a) <= Line number (b)

Explanation

(1) Deletes commands from line (a) up to and including line (b).

(2) If line number (b) is omitted, the contents of line number (a) only are deleted.

* This command can only be executed directly. It cannot be used in a program.

Sample program (MOVEMASTER Command)

100 MO 10 'Moves to position 10.

110 MO 12 'Moves to position 12.

120 MO15 'Moves to position 15.

130 MO17 'Moves to position 17.

140 MO 20 'Moves to position 20.

DL 110 'Deletes line number 110.

DL 120,140 'Deletes line number from 120 to 140.

RV-M1

Function:

Deletes commands of the specified line or step in the program.

Input Format

DL <Line number (a)> [,<Line number (b)>]

Term

<Line number(a)>

Specify the top line number that you want to delete in the program.

<Line number(b)>

Specify the last line number that you want to delete in the program.
1 <= Line number (a), (b) <= 2048
Line number (a) <= Line number (b)

Explanation

(1) Deletes commands from line (a) up to and including line (b).

(2) If line number (b) is omitted, the contents of line number (a) only are deleted.

* This command can only be executed directly. It cannot be used in a program.

Sample program (MOVEMASTER Command)

100 MO 10 'Moves to position 10.

110 MO 12 'Moves to position 12.

120 MO 15 'Moves to position 15.

130 MO 17 'Moves to position 17.

140 MO 20 'Moves to position 20.

DL 110 'Deletes line number 110.

DL 120,140 'Deletes line number from 120 to 140.

COSIMIR® · Movemaster Command

DP (Decrement Position)

Function:

Moves the robot to a predefined position with a number smaller than the current one. (Joint interpolation)

Input Format

DP

Term

Explanation

(1) Moves the robot to a predefined position with a position number smaller than, and closest to, the one. (See command IP)

(2) Alarm takes place if there is no predefined position which is smaller in position number than the current position.

(3) Even if an alarm occurs, current position number is still maintained.

(4) The additional axis will move if the position additional axis data.

Sample program (MOVEMASTER Command)

100 MO 3 'Moves to position 3.

110 MO 4 'Moves to position 4.

120 MO 5 'Moves to position 5.

130 DP 'Moves to position 4.

COSIMIR® · Movemaster Command

DR (Data Read)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Reads the values of the internal register, hand check state, and general output state. (Using RS-232C)

Input Format

DR <Output bit number>

Term0

<Output bit number>

0 <= Output bit number <= 32767
(0 for default)

Explanation

(1) Outputs the values of the internal register, hand check state, and general output state through the RS-232C. Allows the external input data and hand open/close state to be read when executed after the input command (ID).

(2) Hand check state read by the DR command corresponds to the current state.

(3) When you specify the output bit number, you can read the general output state of 16 bits-width from the specified bit number.

(4) The output format is in ASCII coded hexadecimal, which is headed by "&H" and delimited by "," (Hex. 2C).

The output format:

Value of hand check state and the internal register, general output

The first column next to "& H" responds to the hand check input and the following 4 columns responds to the value of internal register, and the remaining 4 columns headed by "&H" responds to the current general output.

(5) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by a personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

Sample program (BASIC)

10 OPEN "COM1 :E83"AS#1 'Opens the RS-232C communication file by the personal computer.

20 PRINT #1,"ID" 'Sets the value of port to the internal register.

50 PRINT #1,"DR" 'Reads the value of internal register and hand check input.

60 LINE INPUT #1,A$ 'Saves the data to

70 PRINT "Data is ";A$ 'Displays the value of A$.

80 END 'Ends

RUN 'Run the BASIC

Data is &H10FB2, &H30BA 'Displays data.

'Value of hand check input:1 (Hexadecimal)

'Value of the internal register: 0FB2 (Hexadecimal)

'Value of general output:30BA (Hexadecimal)

RV-M2

Function:

Reads the values of the internal register and the hand check state (Using RS-232C).

Input Format

DR

Explanation

(1) Outputs the values of the internal register and the hand check state through the RS-232C. Allows the external input data and hand open/close state to be when executed after the input command (ID or IN).

(2) Hand check state read by the DR command corresponds to the current state.

(3) The output format is in ASCII coded hexadecimal, which is headed by "&H" and delimited by "," (Hex. 2C).

The output format:

Value of hand check state and the internal register

The first column next to "& H" responds to the hand check input and the following 4 columns responds to value of internal register.

(4) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

Sample program (BASIC)

10 OPEN " COM1 :9600, E, 7, 2"AS#1 'Opens the RS-232C communication file by the personal computer.

20 PRINT #1,"ID" 'Sets the value input port to the internal register.

50 PRINT #1,"DR" 'Reads the of internal register and hand check input.

60 LINE INPUT #1,A$ 'Saves the to A$.

70 PRINT "Data is ";A$ 'Displays the value of A$.

80 END 'Ends

RUN 'Run the BASIC program.

Data is &H10FB2 'Displays

'Value of hand check input:1 (Hexadecimal)

'Value of the internal register: 0FB2 (Hexadecimal)

RV-M1

Function:

Reads the values of the internal register (Using RS-

Input Format

DR

Explanation

(1) Outputs the values of the internal register through the RS-232C. Allows the external input data to be read when executed after the input command (ID or IN).

(2) The output format is in ASCII coded hexadecimal, which is headed by "&H" and delimited by "," (Hex. 2C).

The output format:

Value of the internal register.

The 2 or 4 columns next to "& H" respond to the value the internal register.

(3) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

Sample program (BASIC)

10 OPEN " COM1 :9600, E, 7, 2"AS#1 'Opens the RS- communication file by the personal computer.

20 PRINT #1,"ID" 'Sets the value of input port to the internal register.

50 PRINT #1,"DR" 'Reads the value internal register and hand check input.

60 LINE INPUT #1,A$ 'Saves the data A$.

70 PRINT "Data is ";A$ 'Displays the of A$.

80 END 'Ends

RUN 'Run the BASIC program.

Data is &HFF 'Displays data.

'Value of the internal register: FF (Hexadecimal)

COSIMIR® · Movemaster Command

DS (Draw Straight)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Moves the end of the hand to a position away from the current position by the distance specified in X,Y and Z directions. (Linear interpolation)

Input Format

DS [<travel distance in X>], [<travel distance in Y>], [<travel distance in Z>]

Term

<Travel distance in X>

Specify the amount that you want to move X direction from the current position.

<Travel distance in Y>

Specify the amount that you want to move Y direction from the current position.

<Travel distance in Z>

Specify the amount that you want to move Z direction from the current position. (Zero traveling for default of each axis.)

Explanation

(1) The least input increment for the distance is

(e.g., specify 20.01 for 20.01mm)

(2) The attitude of the hand, including the open/close status of the gripper, remains the same before and after the movement.

(3) Alarm occurs before or during movement if the destination or travel path exceeds the operating space the robot. Especially the roll joint tends to exceed the operating space since it is controlled to remain the same orientation during the movement.

(4) Moving speed during linear interpolation is decided the SP or SD commands. (Hand tip at constant speed.)

(5) The hand tip is decided by the tool length at that time. (see TL command).

Sample program (MOVEMASTER Command)

10 DS 100,0,0 'Moves to X axis direction by 100 mm.

20 DS 0,100,0 'Moves to Y axis direction by 100 mm.

30 DS -100,0,0 'Moves to X axis direction by -100 mm.

40 DS 0,-100,0 'Moves to Y axis direction by -100 mm.

In the above example, the hand tip moves through the four corners of a square by linear

RV-M2

Function:

Moves the end of the hand to a position away from the current position by the distance specified in X,Y and Z directions. (Linear interpolation)

Input Format

DS [<travel distance in X>], [<travel distance in Y>], [<travel distance in Z>]

Term

<Travel distance in X>

Specify the amount that you want to move in
X direction from the current position.

<Travel distance in Y>

Specify the amount that you want to move
Y direction from the current position.

<Travel distance in Z>

Specify the amount that you want to move
Z direction from the current position.
(Zero traveling for default of each axis.)

Explanation

(1) The least input increment for the distance is 0.1mm (e.g., specify 20.1 for 20.1mm)

(2) The attitude of the hand, including the open/close status of the gripper, remains the same before and the movement.

(3) Alarm occurs before or during movement if the destination or travel path exceeds the operating space the robot. Especially the roll joint tends to exceed the operating space since it is controlled to remain the orientation during the movement.

(4) Moving speed during linear interpolation is decided the SP or SD commands. (Hand tip at constant speed.)

(5) The hand tip is decided by the tool length at that time. (see TL command).

Sample program (MOVEMASTER Command)

```
10 DS 100,0,0      'Moves to X axis direction by 100

20 DS 0,100,0      'Moves to Y axis direction by 100

30 DS -100,0,0     'Moves to X axis direction by -100

40 DS 0,-100,0     'Moves to Y axis direction by -100
```

In the above example, the hand tip moves through the four corners of a square by linear.

RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 series.

COSIMIR® · Movemaster Command

DW (Draw)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Moves the end of the hand to a position away from the current position by the distance specified in X,Y and Z directions. (Joint interpolation)

Input Format

DW [<travel distance in X>], [<travel distance in Y>], [<travel distance in Z>]

Term

<Travel distance in X>

Specify the amount that you want to move X direction from the current position.

<Travel distance in Y>

Specify the amount that you want to move Y direction from the current position.

<Travel distance in Z>

Specify the amount that you want to move Z direction from the current position.
(Zero traveling for default of each axis.)

Explanation

(1) The least input increment for the distance is (e.g., specify 20.01 for 20.01mm)

(2) The attitude of the hand, including the open/close status of the gripper, remains the same before and

the movement. Alarm occurs before the movement if destination exceeds the operating space of the robot.

(3) The moving path draws a curve in the case of large traveling path because of the linear interpolation.

(4) The hand tip is decided by the tool length at that time. (See TL command).

Sample program (MOVEMASTER Command)

```
10 DW 200,0,0       'Moves to X axis direction by 200
20 DW 0,200,0       'Moves to Y axis direction by 200
30 DW -200,0,0      'Moves to X axis direction by -200
40 DW 0,-200,0      'Moves to Y axis direction by -200
```

In the above example, the hand tip moves through the four corners of a square by joint interpolation and returns the start point finally.

RV-M2, RV-M1

Function:

Moves the end of the hand to a position away from the current position by the distance specified in X,Y and Z directions. (Joint interpolation)

Input Format

DW [<travel distance in X>], [<travel distance in Y>], [<travel distance in Z>]

Term

<Travel distance in X>

Specify the amount that you want to move X direction from the current position.

<Travel distance in Y>

Specify the amount that you want to move Y direction from the current position.

<Travel distance in Z>

Specify the amount that you want to move Z direction from the current position.

(Zero traveling for default of each axis.)

Explanation

(1) The least input increment for the distance is 0.1mm (e.g., specify 20.1 for 20.1mm)

(2) The attitude of the hand, including the open/close status of the gripper, remains the same before and the movement. Alarm occurs before the movement if destination exceeds the operating space of the robot.

(3) The moving path draws a curve in the case of large traveling path because of the linear interpolation.

(4) The hand tip is decided by the tool length at that time. (See TL command).

Sample program (MOVEMASTER Command)

```
10 DW 20,0,0      'Moves to X axis direction by 200 mm

20 DW 0,20,0      'Moves to Y axis direction by 200 mm

30 DW -20,0,0     'Moves to X axis direction by -200

40 DW 0,-20,0     'Moves to Y axis direction by -200
```

In the above example, the hand tip moves through the four corners of a square by joint interpolation and returns the start point finally.

COSIMIR® · Movemaster Command

EA (Enable Act)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Enables the interrupt motion by the specified bit of the external input signal.

Input Format

EA [<+/->] <input bit number>, <line number> [, [<branching approach>]]

Term

<+/->

+ : If the <input bit number> of the external input port turns ON, the program jumps to <line number>. (Default)
- :If the <input bit number> of the external input port turns OFF, the program jumps to the <line number>.

<Input bit number>

Specify the bit number of external input signal that you want to assign for interrupt signal
0 <= Input bit number <= 32767
0 - 8999: Input signal interrupt
(0 - 299: general input)
(900 - 903: hand input)
9003 - 32767: Input signal interrupt
- Enables the interrupt motion by the input signal.
- The EA command allows eight input bits to be registered as interrupt signals concurrently.
- When the interrupt signal is input, the is decelerated to a stop and the program jumps to the <line number>.
-Once the interrupt has taken place, the remains stopping until the DA command is executed at the beginning line to which the program jumps or the interrupt signal turns off.
- The signal input except for the specified does not occur any interrupt.
9000 : Alarm interrupt
When alarm occurred, executes the program from the specified line number.
9001 : Emergency stop interrupt
When emergency stop occurred in the operation panel, teaching box, or external signal, then executes the program from the specified line number.
9002 : Communication interrupt
When data received through the RS-232C executes the program from the specified number.

<Line number>

Specify the line number to which the
jumps by the interrupt signal.
1 <= Line number <= 9999

<Branching approach>

Specify a jump or subroutine calling.
0: Jump (See GT command) (Default)
1: Subroutine calling (See GS command)

Explanation

(1) Causes an interrupt to be executed by an external signal while the program is running. When the interrupt signal is input after execution of the EA command, the robot is decelerated to a stop and the program jumps the specified line number.

(2) The EA command allows 8 signals, giving the to the larger number, to be registered as the interrupt signal at the same time.

(3) Once this command has been executed, the condition is maintained till the interrupt disable command (DA), the end command (ED), or the reset command (RS) is carried out.

(4) If the specified line number has no command, alarm will occur in the program execution.

Sample program (MOVEMASTER Command)

100 EA +5,500 'Sets that the jumps to the line 500

'in case of bit 5 ON.

110 MO 1 'Moves to position 1.

120 ED 'Ends program.

500 D A5 'Interrupt disable

510 MO 2 'Moves to position 2.

520 GT 110 'Jumps to line 110.

In the above example, line 100 causes an interrupt to

enabled and line 110 moves the robot to position 1.
the specified signal is input during this motion, the
is decelerated to a stop and then the program jumps to
line 500, where the interrupt is disabled. Line 510
the robot to position 2 and line 520 causes the program
to jump to line 110. The robot moves to position 1

RV-M2

Function:

Enables the interrupt motion by the specified bit of the external input signal.

Input Format

EA <+/-> <Input bit number>, <Line number>

Term

<+/->

+ : If the <input bit number> of the external input port turns ON, the program jumps to <line number>. (Default)
- :If the <input bit number> of the external input port turns OFF, the program jumps to the <line number>.

<Input bit number>

Specify the bit number of external input signal that you want to assign for interrupt signal
0 <= Input bit number <= 17
0 - 17: Input signal interrupt
(0 - 15: general input)
(16 - 17: hand input)

<Line number>

Specify the line number to which the jumps by the interrupt signal.
1 <= Line number <= 3584

Explanation

(1) Causes an interrupt to be executed by an external signal while the program is running. When the interrupt signal is input after execution of the EA command, the robot is decelerated to a stop and the program jumps the specified line number.

(2) Two or more bits may be specified at one time. If there are more than one inputs received, the bit with highest bit number has precedence over the other bits.

(3) Once this command has been executed, the effective condition is maintained till the interrupt disable command (DA), the end command (ED), or the reset command (RS) is carried out.

(4) If the specified line number has no command, alarm will occur in the program execution.

(5) The EA command is invalid during jog operation the teaching box and during movement effected by the move joint instruction (MJ) or nest instruction (NT).

Sample program (MOVEMASTER Command)

100 EA +5,500 'Sets that the program jumps to the line 500

'in case of bit 5 turning ON.

110 MO 1 'Moves to position 1.

120 ED 'Ends program.

500 DA 5 'Interrupt disable

510 MO 2 'Moves to position 2.

520 GT 110 'Jumps to line number 110.

In the above example, line 100 causes an interrupt to enabled and line 110 moves the robot to position 1. the specified signal is input during this motion, the is decelerated to a stop and then the program jumps to line 500, where the interrupt is disabled. Line 510 the robot to position 2 and line 520 causes the program to jump to line 110. The robot moves to position 1

RV-M1

Function:

Enables the interrupt motion by the specified bit of the external input signal.

Input Format

EA <+/-> <Input bit number>, <Line number>

Term

<+/->

+ : If the <input bit number> of the external input port turns ON, the program jumps to <line number>. (Default)
- :If the <input bit number> of the external input port turns OFF, the program jumps to the <line number>.

<Input bit number>

Specify the bit number of external input signal that you want to assign for interrupt signal
0 <= Input bit number <= 7 (15)
Figure in () is for A16 or B16 type I/O card

<Line number>

Specify the line number to which the jumps by the interrupt signal.
1 <= Line number <= 2048

Explanation

(1) Causes an interrupt to be executed by an external signal while the program is running. When the interrupt signal is input after execution of the EA command, the robot is brought to an immediate stop and the program jumps to the specified line number. Avoid using the command while the robot is moving at high speed, as mechanical failure could result. (Set speed below SP5)

(2)Two or more bits may be specified at one time. If there are more than one inputs received, the bit with highest bit number has precedence over the other bits.

(3) Once this command has been executed, the condition is maintained till the interrupt disable command (DA), the end command (ED), or the reset command (RS) is carried out.

(4) If the specified line number has no command, alarm will occur in the program execution.

(5) The EA command is invalid during jog operation the teaching box and during movement effected by the move joint instruction (MJ) or nest instruction (NT).

Sample program (MOVEMASTER Command)

100 EA +5,500 'Sets that the program jumps to line 500

'in case of bit 5 turning ON.

110 MO 1 'Moves to position 1.

120 ED 'Ends program.

500 DA 5 'Interrupt disable

510 MO 2 'Moves to position 2.

520 GT 110 'Jumps to line number 110.

In the above example, line 100 causes an interrupt to be enabled and line 110 moves the robot to position 1. the specified signal is input during this motion, the is decelerated to a stop and then the program jumps to line 500, where the interrupt is disabled. Line 510 the robot to position 2 and line 520 causes the program to jump to line 110. The robot moves to position 1

COSIMIR® · Movemaster Command

ED (End)

Function:

Ends the program.

Input Format

ED

Explanation

(1) Marks the end of a program. The program ends the ED command is executed. In the case of a program program subroutine, however, returns to the calling program .

(2) Required at the end of a program unless the commands are directly executed from the personal computer.

Example for RV-E type robots

Program can call other program from inside the using GS command.

Arrows shows the sequential order of program

In the above example of <Program to program calling>, the program is executed to line 50 then jumps to the program 10. After executing to line 30 of program 10, program returns line 60 of program 1. Program ends the ED command of program 1 is executed.

Sample program (MOVEMASTER Command)

100 SP 3 'Set speed 3

110 MO 3 'Moves to position 3.

120 MO 5 'Moves to position 5.

130 ED 'Ends the program.

COSIMIR® · Movemaster Command

EQ (Equal)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

This compares the value of the internal register with a specified value. If they are the same, the program will jump to the specified line number. The character register and the details of the specified character are compared, and if the values are the same, the program will jump to the specified line number.

Input Format

EQ <compared value/character string number>, <branching line number>

Term

<Compared value>

Specify the value that the internal register compares contents with.
-32768 <= compared value (decimal) <=
&8000 <= compared value (hexadecimal) <= &7FFF
@1 <= counter number <= @99

<Character string number>

Specify character string number in value which "$" is added to the head.
$1 <= character string number <= $99

<Branching line number>

Specify the line number to which the jumps when the comparison result is equal.
1 <= branching line number <= 9999

Explanation

<When counter number is specified>

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value equals to the value (i.e., when the condition is met), the program jumps to the specified line. Otherwise (i.e., when the condition is not met), the program continues in sequence.

(3) A value can be loaded into the internal register by executing the input command (See ID) for the external input data or by executing the compare counter command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed "&".

(5) Alarm occurs at a jump if the specified line number does not exist.

<When character string number is specified>

(1) The conditions will jump depending on the details

the character string register.

(2) If the details of the character string register are to the details of the character string number (when the conditions are established), the program will jump to specified line number. If the details are not equal (when conditions are not established), the next line will be executed.

(3) By executing an INP command, the data input from external device will be set in the character string register. The details of the character string number will be set by executing a CP command. Thus, when executing condition jumping, one of these commands must be executed first.

(4) If the specified line number is not registered, an alarm will occur during jumping.

Sample program (MOVEMASTER Command)

100 ID 'Fetches data from external port.

110 EQ 100,130 'Jumps to line 130 if the data equals 100.

120 ED 'Ends the program if above condition is not met.

130 SC $5,"OK" 'Set character string "OK" in character string number 5

140 CP $5 'Set details of character string number 5 in the character string register

150 EQ $10,200 'Jumps to line 200 if the data equals character string number 10.

:

200 MO 7 'Moves to position 7.

RV-E-Robot (RV-E2, RV-E3J, etc.)

Function:

Causes a jump to occur if the contents of the internal register equal a specified value.

Input Format

EQ <compared value>, <branching line number>

Term

<Compared value>

Specify the value that the internal register compares contents with.
-32768 <= compared value (decimal) <=
&8000 <= compared value (hexadecimal) <=
&7FFF

<Branching line number>

Specify the line number to which the
jumps when the comparison result is equal.
1 <= branching line number <= 9999

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value equals to the value (i.e., when the condition is met), the program jumps to the specified line. Otherwise (i.e., when the condition is not met), the program continues in sequence.

(3) A value can be loaded into the internal register by executing the input command (See ID) for the external input data or by executing the compare counter command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed "&".

(5) Alarm occurs at a jump if the specified line number does not exist.

Sample program (MOVEMASTER Command)

100 ID 'Fetches data from external input

110 EQ 100,130 'Jumps to line 130 if the data equals 100.

120 ED 'Ends the program if above condition

not met.

130 MO 7 'Moves to position 7.

RV-M2

Function:

Causes a jump to occur if the contents of the internal register equal a specified value.

Input Format

EQ <compared value>, <branching line number>

Term

<Compared value>

Specify the value that the internal register compares contents with.
-32768 <= compared value (decimal) <=
&8000 <= compared value (hexadecimal) <=
&7FFF

<Branching line number>

Specify the line number to which the
jumps when the comparison result is equal.
1 <= branching line number <= 3584

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value equals to the value (i.e., when the condition is met), the program jumps to the specified line. Otherwise (i.e., when the condition is not met), the program continues in sequence.

(3) A value can be loaded into the internal register by executing the input command (See ID) for the external input data or by executing the compare counter command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed

"&".

(5) Alarm occurs at a jump if the specified line number does not exist.

Sample program (MOVEMASTER Command)

100 ID 'Fetches data from external input

110 EQ 100,130 'Jumps to line 130 if the data equals 100.

120 ED 'Ends the program if above condition not met.

130 MO 7 'Moves to position 7.

RV-M1

Function:

Causes a jump to occur if the contents of the internal register equal a specified value.

Input Format

EQ <compared value>, <branching line number>

Term

<Compared value>

Specify the value that the internal register compares contents with.
-32768 <= compared value (decimal) <=
&8000 <= compared value (hexadecimal) <=
&7FFF

<Branching line number>

Specify the line number to which the jumps when the comparison result is equal.
1 <= branching line number <= 2048

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value equals to the value (i.e., when the condition is met), the program

jumps to the specified line. Otherwise (i.e., when the condition is not met), the program continues in sequence.

(3) A value can be loaded into the internal register by executing the input command (See ID) for the external input data or by executing the compare counter command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed "&".

(5) Alarm occurs at a jump if the specified line number does not exist.

Sample program (MOVEMASTER Command)

```
100 ID              'Fetches data from external input
port.

110 EQ 100,130      'Jumps to line 130 if the data
100.

120 ED              'Ends the program if above
is not met.

130 MO 7            'Moves to position 7.
```


COSIMIR® · Movemaster Command

ER * (Error Read)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

Reads the current error status and alarm history contents. (Using RS-232C)

Input Format

ER [<alarm history number>]

Term

<Alarm history number>

Specify the number of alarm history.
1 <= alarm history number <= 128 (If shows the current alarm)

Explanation

(1) Outputs the alarm condition of the robot from the 232C port.

(2) Specify the alarm history number on the basis of current alarm (0). When you specified 1, outputs the alarm information former than the current alarm by 1.

(3) The ER command outputs the former alarm history the ASCII format as below.

The output format:

Alarm history number, alarm number, year, month, hour, minute, second.

(4) If the alarm history number is omitted, the ER command outputs the current alarm with following code.

0; No alarm

1: Serious alarm (Alarm 0100-1900)

2: Operation alarm (Alarm 2300-8900)

(5) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(6) Useful to transfer a sequence of data from the personal computer to the robot while checking for an alarm.

* This command can only be executed directly. It be used in a program.

1. Sample program (BASIC)

10 OPEN "COM1 :E83"AS#1 'Opens the RS-232C communication file

'from the personal computer by BASIC.

20 PRINT#1, "MO1" 'Moves to position 1.

30 GOSUB 100 'Calls the subroutine line 100 in BASIC.

40 PRINT #1, "MO2" 'Moves to position 2.

50 GOSUB 100 'Calls the subroutine line 100 in BASIC.

60 END 'Ends

100 PRINT #1, "ER" 'Reads the current alarm.

110 LINE INPUT #1,A$ 'Saves the received data to A$.

120 IF A$="0"THEN RETURN 'If there is no alarm, returns subroutine.

130 PRINT "Alarm level is "; A$ 'Displays the data on the personal computer screen.

140 END 'Ends

RUN 'Run the BASIC

Alarm level is 2 'Displays alarm level.

(2: Operation alarm)

2. Sample program (BASIC)

10 OPEN "COM1:E83"AS#1 'Opens the RS-232C communication file

'from the personal computer by BASIC.

20 INPUT "History number is "; N 'Enters the alarm history number by the personal computer.

30 PRINT #1,"ER"+STR$(N) 'Reads the alarm information.

40 LINE INPUT #1,A$ 'Saves the received data to A$.

50 PRINT "Alarm information ";A$ 'Displays the data on the personal computer screen.

60 END 'Ends

RUN 'Run the BASIC program.

History number is 1

Alarm information 1,3800,93,10,21,11,34,20

RV-E-Robot (RV-E2, RV-E3J, etc.)

Function:

Reads the current error status and alarm history contents. (Using RS-232C)

Input Format

ER [<alarm history number>]

Term

<Alarm history number>

Specify the number of alarm history.
1 <= alarm history number <= 16 (If omitted, shows the current alarm)

Explanation

(1) Outputs the alarm condition of the robot from the 232C port.

(2) Specify the alarm history number on the basis of current alarm (0). When you specified 1, outputs the alarm information former than the current alarm by 1.

(3) The ER command outputs the former alarm history the ASCII format as below.

The output format:

Alarm history number, alarm number, year, month, hour, minute, second.

(4) If the alarm history number is omitted, the ER command outputs the current alarm with following code.

0: No alarm

1: Serious alarm (Alarm 0100-1900)

2: Operation alarm (Alarm 2300-8900)

(5) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by a personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(6) Useful to transfer a sequence of data from the personal computer to the robot while checking for an alarm.

* This command can only be executed directly. It cannot be used in a program.

1. Sample program (BASIC)

```
10 OPEN "COM1 :E83"AS#1          'Opens the RS-232C
communication file

                                 'from the personal
computer by BASIC.

20 PRINT#1, "MO1"                'Moves to position 1.

30 GOSUB 100                     'Calls the subroutine
line 100 in BASIC.

40 PRINT #1, "MO2"               'Moves to position 2.

50 GOSUB 100                     'Calls the subroutine
line 100 in BASIC.

60 END                           'Ends

100 PRINT #1, "ER"               'Reads the current
alarm.

110 LINE INPUT #1,A$             'Saves the received
data to A$.

120 IF A$="0"THEN RETURN         'If there is no alarm,
returns subroutine.

130 PRINT "Alarm level is "; A$  'Displays the data on
the personal computer screen.

140 END                          'Ends
```

RUN 'Run the BASIC

Alarm level is 2 'Displays alarm level.

(2: Operation alarm)

2. Sample program (BASIC)

10 OPEN "COM1:E83"AS#1 'Opens the RS-232C communication file

'from the personal computer by BASIC.

20 INPUT "History number is "; N 'Enters the alarm number by the personal computer.

30 PRINT #1,"ER"+STR$(N) 'Reads the alarm information.

40 LINE INPUT #1,A$ 'Saves the received data to A$.

50 PRINT "Alarm information ";A$ 'Displays the data on the personal computer screen.

60 END 'Ends

RUN 'Run the BASIC

History number is 1

Alarm information 1,3800,93,10,21,11,34,20

RV-M2, RV-M1

Function:

Reads the current error status and alarm history contents. (Using RS-232C)

Input Format

ER

Explanation

(1) Outputs the alarm condition of the robot from the 232C port.

(2) The ER command outputs the current alarm with following ASCII code.

0: No alarm

1: Serious alarm

2: Operation alarm

(3) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by a personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(4) Useful to transfer a sequence of data from the personal computer to the robot while checking for an alarm.

* This command can only be executed directly. It cannot be used in a program.

1. Sample program (BASIC)

10 OPEN "COM1 :9600, E, 7, 2"AS#1 'Opens the RS-232C communication file

'from the computer by BASIC.

20 PRINT#1, "MO1" 'Moves to position 1.

30 GOSUB 100 'Calls the subroutine line 100 in BASIC.

40 PRINT #1, "MO2" 'Moves to position 2.

50 GOSUB 100 'Calls the subroutine line 100 in BASIC.

60 END 'Ends

100 PRINT #1, "ER" 'Reads the current alarm.

110 LINE INPUT #1,A$ 'Saves the received data to A$.

120 IF A$="0"THEN RETURN 'If there is no alarm, returns subroutine.

130 PRINT "Alarm level is "; A$ 'Displays the data on the personal computer screen.

140 END 'Ends

RUN 'Run the BASIC
program.

Alarm level is 2 'Displays alarm
level.

(2: Operation alarm)

COSIMIR® · Movemaster Command

GC (Grip Close)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

Close the grip of hand.

Input Format

GC [<hand number>]

Term

<Hand number>

Specify the hand number opening grip.
0: Hand 1 (Default)
1: Hand 2
2: Hand 3
3: Hand 4

Explanation

(1) **<Motor-operated hand>** Causes the grip of the hand to be closed by the gripping force waveform defined by the GP command. The motor-operated hand corresponds to only hand 1.

(2) **<Pneumatic hand>** Causes the solenoid valve to be energized to close the hand (or to attract the

workpiece). Refer to the OB command about output bits pattern in this case.

(3) A certain period of time is required when the robot positioned and grasp the workpiece. Therefore a time delay before and after the GC command may be necessary. The GC commands still contains the timer specified by "starting gripping force retention time" in the GP command.

Relating Parameters

The open or close state of the hand at the execution of hand command and at the power activation can be changed by the following parameter (see the for the RV-E2 and RV-E3J and the RV-EN series). To find the change operation, refer to your manual.

Parameter GCD:

Hand 1 forward/reverse, hand 1 default,

Hand 2 forward/reverse, hand 2 default

Contents of each value are

(1) Hand 1 forward/reverse, hand2 forward/reverse: direction of hand open/close.(Default set is 0.)

Motor-operated hand

Hand 1 (2,3,4) forward/reverse setting	0	1
Open/close direction	Normal open/close	Reverse open/close

(2) Hand 1 default, hand 2 default: Set status of output signal 900 and 901 at power on. If hand 2 then output signal are 902 and 903.(Default set is 1.)

Pneumatic hand

Hand 1 (2,3,4) default setting	0	1	2	3
Output bit 900 (902, 904, 906)	0	1	0	1
Output bit 901 (903, 905, 907)	0	0	1	1

Sample program (MOVEMASTER Command)

10 MO 10,O 'Moves to position 10 with hand

20 TI 5 'Sets 0.5 second timer.

30 GC 'Closes hand to grasp workpiece.

40 TI 5 'Sets 0.5 second timer.

50 MO 15,C 'Moves to position 15 with hand closed.

RV-E-Robot (RV-E2, RV-E3J, etc.)

Function:

Close the grip of hand.

Input Format

GC [<hand number>]

Term

<Hand number>

Specify the hand number opening grip.
0: Hand 1 (Default)
1: Hand 2

Explanation

(1) **<Motor-operated hand>** Causes the grip of the hand to be closed by the gripping force waveform defined by the GP command. The motor-operated hand corresponds to only hand 1.

(2) **<Pneumatic hand>** Causes the solenoid valve to be energized to close the hand (or to attract the workpiece). Refer to the OB command about output bits pattern in this case.

(3) A certain period of time is required when the robot positioned and grasp the workpiece. Therefore a time delay before and after the GC command may be necessary. The GC commands still contains the timer specified by "starting gripping force retention time" in the GP command.

Relating Parameters

The open or close state of the hand at the execution of hand command and at the power activation can be changed by the following parameter (see the for the RV-E2 and RV-E3J and the RV-EN series).To find the change operation, refer to your manual.

Parameter GCD:

Hand 1 forward/reverse, hand 1 default,

Hand 2 forward/reverse, hand 2 default

Contents of each value are

(1) Hand 1 forward/reverse, hand2 forward/reverse: direction of hand open/close.(Default set is 0.)

Motor-operated hand

Hand 1 (2) forward/reverse setting	0	1
Open/close direction	Normal open/close	Reverse open/close

(2) Hand 1 default, hand 2 default: Set status of output signal 900 and 901 at power on. If hand 2 then output signal are 902 and 903.(Default set is 1.)

Pneumatic hand

Hand 1 (2) default setting	0	1	2	3
Output bit 900 (902)	0	1	0	1
Output bit 901 (903)	0	0	1	1

Sample program (MOVEMASTER Command)

```
10 MO 10,O      'Moves to position 10 with hand

20 TI 5         'Sets 0.5 second timer.

30 GC           'Closes hand to grasp workpiece.

40 TI 5         'Sets 0.5 second timer.

50 MO 15,C      'Moves to position 15 with hand closed.
```

RV-M2

Function:

Close the grip of hand.

Input Format

GC

Explanation

(1) **<Motor-operated hand>** Causes the grip of the hand to be closed by the gripping force waveform defined by the GP command. The "retained gripping force" is only valid among the GP command parameters if the GC command is used repeatedly.

(2) **<Pneumatic hand>** Causes the solenoid valve to be energized to close the hand (or to attract the workpiece). Refer to the OB command about output bits pattern in this case.

(3) A certain period of time is required when the robot positioned and grasp the workpiece. Therefore a time delay before and after the GC command may be necessary. The GC commands still contains the timer specified by "starting gripping force retention time" in the GP command.

(4) The hand open/close condition depends on the of bit 3 of switch SW1 in the drive unit side door. (Ordinary open/close operation in the lower position. Revers open/close operation in the upper position.)

Sample program (MOVEMASTER Command)

```
10 MO 10,O    'Moves to position 10 with hand opened.

20 TI5        'Sets 0.5 second timer.

30 GC         'Closes hand to grasp workpiece.

40 TI 5       'Sets 0.5 second timer.

50 MO 15,C    'Moves to position 15 with hand closed.
```

RV-M1

Function:

Close the grip of hand.

Input Format

GC

Explanation

(1) **<Motor-operated hand>** Causes the grip of the hand to be closed by the gripping force waveform defined by the GP command. The "retained gripping force" is only valid among the GP command parameters if the GC command is used repeatedly.

(2) **<Pneumatic hand>** Causes the solenoid valve to be energized to close the hand (or to attract the workpiece). Refer to the OB command about output bits pattern in this case.

(3) A certain period of time is required when the robot positioned and grasp the workpiece. Therefore a time delay before and after the GC command may be necessary. The GC commands still contains the timer specified by "starting gripping force retention time" in the GP command.

Sample program (MOVEMASTER Command)

```
10 MO 10,O     'Moves to position 10 with hand

20 TI5          'Sets 0.5 second timer.

30 GC           'Closes hand to grasp workpiece.

40 TI 5         'Sets 0.5 second timer.

50 MO 15,C     'Moves to position 15 with hand closed.
```


COSIMIR® · Movemaster Command

GF (Grip Flag)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Defines the open/close state of the grip of the hand (used with the PD command).

Input Format

GF <switch>

Term

<Switch>

Specify open or close state of the hand in 0
1.
0: Open
1: Close

Explanation

(1) Defines the open or close state of the hand grip with the PD command which defines the coordinates of the specified position. The PD command takes precedence if the hand open/close state has been specified by the PD command.

Sample program (MOVEMASTER Command)

10 GF 0 'Sets the grip flag to open.

20 PD 10,50,320,70,50,40,30,R 'Defines position 10 with hand opened.

RV-M2

Function:

Defines the open/close state of the grip of the hand (used with the PD command).

Input Format

GF <switch>

Term

<Switch>

Specify open or close state of the hand in 0
1.
0: Open
1: Close

Explanation

(1) Defines the open or close state of the hand grip with the PD command which defines the coordinates of the specified position. The PD command takes

precedence if the hand open/close state has been specified by the PD command.

(2) The grip is open when the switch is set to 0 and closed when the switch is 1. The switch is set to 0 (grip open) when the power is turned on. The hand state depends on the setting of bit 3 of switch SW1 in drive unit side door. (Ordinary open/close operation in the lower position. Reverse open/close operation in the upper position.)

(3) Once made, the switch setting is valid until a new value is set.

Sample program (MOVEMASTER Command)

10 GF 0 'Sets the grip flag to open.

20 PD 10,0,550,450,-50,0 'Defines position 10 with opened.

RV-M1

Function:

Defines the open/close state of the grip of the hand (used with the PD command).

Input Format

GF <switch>

Term

<Switch>

Specify open or close state of the hand in 0 1.
0: Open
1: Close

Explanation

(1) Defines the open or close state of the hand grip with the PD command which defines the coordinates of the specified position. The PD command takes precedence if the hand open/close state has been specified by the PD command.

(2) The grip is open when the switch is set to 0 and closed when the switch is 1. The switch is set to 0 (grip open) when the power is turned on.

(3) Once made, the switch setting is valid until a new value is set.

Sample program (MOVEMASTER Command)

10 GF 0 'Sets the grip flag to

20 PD 10,0,550,450,-50,0 'Defines position 10 with hand opened.

COSIMIR® · Movemaster Command

GO (Grip Open)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

Opens the grip of the hand.

Input Format

GO [<hand number>]

Term

<Hand number>

Specify the hand number opening grip.
0: Hand 1 (Default)
1: Hand 2
2: Hand 3
3: Hand 4

Explanation

(1) **<Motor-operated hand>** Causes the grip of the hand to be opened by the gripping force waveform defined the GP command. The motor-operated hand corresponds to only hand 1.

(2) **<Pneumatic hand>** Causes the solenoid valve to be energized to open the hand (or to release the workpiece). Refer to the OB command about output bits pattern in this case.

(3) A certain period of time is required when the robot positioned and releases the workpiece. Therefore a time delay before and after the GO command may be necessary. The GO commands still contains the timer specified by "starting gripping force retention time" in the GP command.

Relating Parameters

The open or close state of the hand at the execution of hand command and at the power activation can be changed by the following parameter (see the for the RV-E2 and RV-E3J and the RV-EN series). To find the change operation, refer to your manual.

Parameter GCD:

Hand 1 forward/reverse, hand 1 default,

Hand 2 forward/reverse, hand 2 default

Contents of each value are

(1) Hand 1 forward/reverse, hand2 forward/reverse: direction of hand open/close.(Default set is 0.)

Motor-operated hand

Hand 1 (2) forward/reverse setting	0	1
Open/close direction	Normal open/close	Reverse open/close

(2) Hand 1 default, hand 2 default: Set status of output signal 900 and 901 at power on. If hand 2 then output signal are 902 and 903.(Default set is 1.)

Pneumatic hand

Hand 1 (2,3,4) default setting	0	1	2	3
Output bit 900 (902, 904, 906)	0	1	0	1
Output bit 901 (903, 905, 907)	0	0	1	1

Sample program (MOVEMASTER Command)

```
10 MO 10,C          'Moves to position 10 with hand
```

closed.

20 TI 5 'Sets 0.5 second timer.

30 GO 'Opens hand to release workpiece.

40 TI 5 'Sets 0.5 second timer.

50 MO 15,O 'Moves to position 15 with hand opened.

RV-E-Robot (RV-E2, RV-E3J, etc.)

Function:

Opens the grip of the hand.

Input Format

GO [<hand number>]

Term

<Hand number>

Specify the hand number opening grip.
0: Hand 1 (Default)
1: Hand 2

Explanation

(1) **<Motor-operated hand>** Causes the grip of the hand to be opened by the gripping force waveform defined the GP command. The motor-operated hand corresponds to only hand 1.

(2) **<Pneumatic hand>** Causes the solenoid valve to be energized to open the hand (or to release the workpiece). Refer to the OB command about output bits pattern in this case.

(3) A certain period of time is required when the robot positioned and releases the workpiece. Therefore a delay before and after the GO command may be necessary. The GO commands still contains the timer specified by "starting gripping force retention time" in the GP command.

Relating Parameters

The open or close state of the hand at the execution of hand command and at the power activation can be changed by the following parameter (see the for the RV-E2 and RV-E3J and the RV-EN series). To find the change operation, refer to your manual.

Parameter GCD:

Hand 1 forward/reverse, hand 1 default,

Hand 2 forward/reverse, hand 2 default

Contents of each value are

(1) Hand 1 forward/reverse, hand 2 forward/reverse: direction of hand open/close.(Default set is 0.)

Motor-operated hand

Hand 1 (2) forward/reverse setting	0	1
Open/close direction	Normal open/close	Reverse open/close

(2) Hand 1 default, hand 2 default: Set status of output signal 900 and 901 at power on. If hand 2 then output signal are 902 and 903.(Default set is 1.)

Pneumatic hand

Hand 1 (2) default setting	0	1	2	3
Output bit 900 (902)	0	1	0	1
Output bit 901 (903)	0	0	1	1

Sample program (MOVEMASTER Command)

```
10 MO 10,C        'Moves to position 10 with hand closed.
20 TI 5           'Sets 0.5 second timer.
30 GO             'Opens hand to release workpiece.
40 TI 5           'Sets 0.5 second timer.
50 MO 15,O        'Moves to position 15 with hand opened.
```

RV-M2

Function:

Opens the grip of the hand.

Input Format

GO

Term

Explanation

(1) **<Motor-operated hand>** Causes the grip of the hand to be opened by the gripping force waveform defined the GP command. The "retained gripping force" is only valid among the GP command parameters if the GC command is used repeatedly

(2) **<Pneumatic hand>** Causes the solenoid valve to be energized to open the hand (or to release the workpiece). Refer to the OB command about output bits pattern in this case.

(3) A certain period of time is required when the robot positioned and releases the workpiece. Therefore a delay before and after the GO command may be necessary. The GO commands still contains the timer specified by "starting gripping force retention time" in the GP command.

(4) The hand open/close condition depends on the of bit 3 of switch SW1 in the drive unit side door. (Ordinary open/close operation in the lower position. Revers open/close operation in the upper position.)

Sample program (MOVEMASTER Command)

10 MO 10,C 'Moves to position 10 with hand closed.

20 TI 5 'Sets 0.5 second timer.

30 GO 'Opens hand to release workpiece.

40 TI 5 'Sets 0.5 second timer.

50 MO 15,O 'Moves to position 15 with hand opened.

RV-M1

Function:

Opens the grip of the hand.

Input Format

GO

Term

Explanation

(1) **<Motor-operated hand>** Causes the grip of the hand to be opened by the gripping force waveform defined the GP command. The "retained gripping force" is only valid among the GP command parameters if the GC command is used repeatedly

(2) **<Pneumatic hand>** Causes the solenoid valve to be energized to open the hand (or to release the workpiece). Refer to the OB command about output bits pattern in this case.

(3) A certain period of time is required when the robot positioned and releases the workpiece. Therefore a delay before and after the GO command may be necessary. The GO commands still contains the timer specified by "starting gripping force retention time" in the GP command.

Sample program (MOVEMASTER Command)

```
10 MO 10,C          'Moves to position 10 with hand closed.

20 TI 5             'Sets 0.5 second timer.

30 GO               'Opens hand to release workpiece.

40 TI 5             'Sets 0.5 second timer.

50 MO 15,O          'Moves to position 15 with hand opened.
```


COSIMIR® · Movemaster Command

GP (Grip Pressure)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.), RV-M2

Function:

Defines the gripping force to be applied when the operated hand is closed and opened.

Input Format

GP <Starting gripping force>, <Retained gripping force>, <Starting gripping force retention time>

Term

<Starting gripping force>

Specify necessary gripping force as integer value to activate hand open or close
0 <= Starting gripping force <= 63

<Retained gripping force>

Specify necessary gripping force as integer value to maintain hand open or close.
0 <= Retained gripping force <= 63

< Starting gripping force retention time >

Specify time continuing starting gripping force as value
0 <= Starting gripping force retention time <= 99 (9,9 seconds)

Explanation

(1) Sets the gripping force of the motor-operate hand (option) as it changes with time (See commands GO and GC).

(2) The starting and retained gripping forces are 63 at their maximum and 0 at their minimum. The starting gripping force retention time is the parameter times seconds (max. 9.9 seconds). Define the parameters optimum for the workpiece to be held. The parameter setting, once made, remains effective until a new is set and also effective in other programs.

(3) The default parameter values are "GP 63,63,3".

(4) When the pneumatic hand is used, the parameter, starting and retained gripping forces, are ineffective.

(5) The robot motion stops during starting gripping retention time.

Sample program (MOVEMASTER Command)

```
10 GP 10, 6, 10       'Set grip pressure
20 GC                 'Closes hand in above settings
```

RV-M1

Function:

Defines the gripping force to be applied when the operated hand is closed and opened.

Input Format

GP <Starting gripping force>, <Retained gripping force>, <Starting gripping force retention time>

Term

<Starting gripping force>

Specify necessary gripping force as integer value to activate hand open or close
0 <= Starting gripping force <= 15

<Retained gripping force>

Specify necessary gripping force as integer value to maintain hand open or close.
0 <= Retained gripping force <= 15

< Starting gripping force retention time >

Specify time continuing starting gripping as integer value
0 <= Starting gripping force retention time
99 (9,9 seconds)

Explanation

(1) Sets the gripping force of the motor-operate hand (option) as it changes with time (See commands GO and GC).

(2) The starting and retained gripping forces are 15 at their maximum and 0 at their minimum. The starting gripping force retention time is the parameter times seconds (max. 9.9 seconds). Define the parameters optimum for the workpiece to be held. The parameter setting, once made, remains effective until a new is set and also effective in other programs.

(3) The default parameter values are "GP 10,10,3".

(4) When the pneumatic hand is used, the parameter, starting and retained gripping forces, are ineffective.

(5) The robot motion stops during starting gripping force retention time.

Sample program (MOVEMASTER Command)

10 GP 10, 6, 10	'Set grip pressure
20 GC	'Closes hand in above settings

COSIMIR® · Movemaster Command

GS (Go Sub)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Carries out subroutine beginning with the specified line number.

Input Format

GS [<line number>] [, [<program name>]]

Term

<Line number>

Specify line number of subroutine in integer value.
1 <= line number <= 9999

<Program name>

Specify program name of subroutine in value or characters. (Less than 8 characters)
1 <= program name <= 8 (characters)
Possible letter used:Digits (0-9)

Characters (A - Z)
Symbols (! @ # $ % ^ & ( ) _ | { } -)
Impossible letter used: * + , . / : ; = ? [\ ] '
Special specification:When you specified only numeric value, the program name is handled as number.
Need to enclose program name with " " in the case of character used.

Explanation

(1) Allows the program to jump to the specified line of the specified program and execute subroutine. The program returns to the main program after executing subroutine. When you specified program number, to the main program by ED command and when you specified only line number, returns by RT command.

(2) Use the RT command to terminate the subroutine existing in the same program. Use the ED command to terminate the subroutine existing in other program.

(3) If the specified line or the specified program does exist, alarm occurs at the time of GS execution.

(4) When you omitted line number, executes the specified program from the top line.

(5) When you omitted line number and program name, nothing occurs .

(6) To call subroutines in other subroutines is called "Nesting". Up to 9 Nesting levels are possible.

Example on the GS command

In the above example of <The same program call>, executes the program from line 10 to 30, then calls the subroutine of line 200. When the RT command is executed in the subroutine, the program returns to the main program and continues from line 40. The program ends when the ED command is executed.

Program can call other program from inside the using GS command. In the above example of <Program program call>, executes the program from line 20 to then calls the program 30. Executes the program from line 40 to 100 and returns to the main program, i.e., 60 of program 20. The program ends when the ED command is executed.

Sample program (MOVEMASTER Command)

10 GS 100 'Carry out subroutine beginning with line number 100.

:

90 ED 'Ends program.

100 MO 11 'Moves to position 11.

110 MO 12 'Moves to position 12.Subroutine

120 MO 13 'Moves to position 13.

130 RT 'Ends subroutine

RV-M2

Function:

Carries out subroutine beginning with the specified line number.

Input Format

GS <Line number>

Term

<Line number>

Specify line number of subroutine in integer value.
1 <= line number <= 3584

Explanation

(1) Allows the program to jump to the specified line of the specified program and execute subroutine. The program returns to the main program after executing subroutine. If the specified line does not exist, alarm (error mode II) occurs at the time of GS execution.

(2) Subroutines are written and stored separately from the main program and must be terminated by the RT command.

(3) To call subroutines in other subroutines is called "nesting". Up to 9 nesting levels are possible.

Sample program (MOVEMASTER Command)

10 GS 100 'Carry out subroutine beginning with line number 100.

:

90 ED 'Ends program.

100 MO 11 'Moves to position 11.

110 MO 12 'Moves to position 12.Subroutine

120 MO 13 'Moves to position 13.

130 RT 'Ends subroutine

RV-M1

Function:

Carries out subroutine beginning with the specified line number.

Input Format

GS <Line number>

Term

<Line number>

Specify line number of subroutine in integer value.
1 <= line number <= 2048

Explanation

(1) Allows the program to jump to the specified line of the specified program and execute subroutine. The program returns to the main program after executing subroutine. If the specified line does not exist, alarm (error mode II) occurs at the time of GS execution.

(2) Subroutines are written and stored separately from the main program and must be terminated by the RT command.

(3) To call subroutines in other subroutines is called "nesting". Up to 9 nesting levels are possible.

Sample program (MOVEMASTER Command)

10 GS 100 'Carry out subroutine beginning with line number 100.

:

90 ED	'Ends program.
100 MO 11	'Moves to position 11.
110 MO 12	'Moves to position 12.Subroutine
120 MO 13	'Moves to position 13.
130 RT	'Ends subroutine

COSIMIR® · Movemaster Command

GT (Go To)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Jumps to the specified line number unconditionally.

Input Format

GT <Line number>

Term

<Line number>

Specify the line number to which the jumps.
1 <= line number <= 9999

Explanation

(1) Causes the program to jump to the specified line number.

(2) If the specified line number does not exist, alarm occurs at the time of GT execution.

Sample program (MOVEMASTER Command)

10 MO 1	'Moves to position 1.

20 GT 100 'Jumps to line 100 unconditionally.

:

100 MO 12 'Moves to position 12.

110 MO 15 'Moves to position 15.

:

RV-M2

Function:

Jumps to the specified line number unconditionally.

Input Format

GT <Line number>

Term

<Line number>

Specify the line number to which the program jumps.
1 <= line number <= 3584

Explanation

(1) Causes the program to jump to the specified line number.

(2) If the specified line number does not exist, alarm occurs at the time of GT execution.

Sample program (MOVEMASTER Command)

10 MO 1 'Moves to position 1.

20 GT 100 'Jumps to line 100 unconditionally.

:

100 MO 12 'Moves to position 12.

110 MO 15 'Moves to position 15.

:

RV-M1

Function:

Jumps to the specified line number unconditionally.

Input Format

GT <Line number>

Term

<Line number>

Specify the line number to which the program jumps.
1 <= line number <= 2048

Explanation

(1) Causes the program to jump to the specified line number.

(2) If the specified line number does not exist, alarm occurs at the time of GT execution.

Sample program (MOVEMASTER Command)

```
10 MO 1      'Moves to position 1.
20 GT 100    'Jumps to line 100 unconditionally.
:
100 MO 12    'Moves to position 12.
110 MO 15    'Moves to position 15.
:
```


COSIMIR® · Movemaster Command

HE (Here)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Defines the current coordinates as the specified

Input Format

HE <Position number>

Term

<Position number>

Specify the position number to be
0 <= position number <= 999
Registers the current position to the user-defined origin in case of zero.

Explanation

(1)The coordinates of the current position are on the basis of the currently set tool length (see the TL command). In the initial condition, the tool length is mm, a point away from the hand mounting surface toward the end of the hand.

(2) If a single number is assigned to two different positions, the one defined last takes precedence with former cleared.

(3) The open/close position of the hand and the flag data are also stored as the position data.

(4) Alarm occurs if the HE command is executed before the origin setting.

(5) When you specified zero position number, current position data in joint coordinates are defined to user-defined origin parameter UOG (see the parameters for the RV-E2 and RV-E3J and the RV-EN series). In this you must change the permission parameter HOE to permit the origin setting at first. After that, return the permission parameter to the former value. (Does not permit the setting.) The above operation is effective by direct command execution.

Sample program (MOVEMASTER Command)

```
10 MO 10        'Moves to position 10.

20 DW 10,0,0    'Moves to +X direction by 10 mm

30 HE 11        'Defines above location as position 11.
```

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Registers the current coordinates in a designated position.

Input Format

HE <Position number> [, <Input format>]

Term

<Position number>

The registration destination position No. Is expressed as an integer.
0 <= Position number <= 999
However, when if 0 is used, enter the user-defined origin.

<Input format>

Specifies the input format of the position for an additional axis.
1: For registering both the robot,s and the additional axis, current positions. (When omitting)
2: For registering only the robot,s current position.
Note that if the position No. Is 0, or during the standard system, this will be invalid.

Explanation

(1) When 0 is designated as the position number, the additional axis position data will not be registered in user-defined origin parameter, UOG.

(2) If you omit the input format, or if you enter 0, the current position of the robot and the additional axis be registered. If you designate 1 for the input format, only the robot,s current position will be entered.

(3) When there is a movement command immediately before this command, it will be executed even though the additional axis is moving. In that event, a position along the moving path will be registered.

(4) In the event that you designate a command to start executing before the additional axis finishes, when the HE command is issued, it might register a position the moving path of the additional axis (Refer to WRM command).

Related parameters

UOG and HOE

Related language

WRM

Sample program (MOVEMASTER Command)

10 MO 10 'Moves to position 10.

20 ML 100 'Move the first additional axis 100 (mm deg.)

30 HE 11 'Enter the above position as the position (the additional axis data is also entered)

RV-M2

Function:

Defines the current coordinates as the specified

Input Format

HE <Position number>

Term

<Position number>

Specify the position number to be
1 <= position number <= 999
Registers the current position to the user-defined origin in case of zero.

Explanation

(1)The coordinates of the current position are on the basis of the currently set tool length (see the TL command). In the initial condition, the tool length is mm, a point away from the hand mounting surface toward the end of the hand.

(2) If a single number is assigned to two different positions, the one defined last takes precedence with former cleared.

(3) The open/close position of the hand and the flag data are also stored as the position data.

(4) Error mode II may occur if the HE command is executed when any of the axes of the robot is

near the boundary of its operational space. In such cases avoid such attitude of the robot.

(5) Alarm (error mode II) occurs if the HE command is executed before the origin setting.

Sample program (MOVEMASTER Command)

10 MO 10 'Moves to position 10.

20 DW 10,0,0 'Moves to +X direction by 10 mm

30 HE 11 'Defines above location as position 11.

RV-M1

Function:

Defines the current coordinates as the specified

Input Format

HE <Position number>

Term

<Position number>

Specify the position number to be
1 <= position number <= 629
Registers the current position to the user-defined origin in case of zero.

Explanation

(1)The coordinates of the current position are on the basis of the currently set tool length (see the TL command). In the initial condition, the tool length is mm, a point away from the hand mounting surface toward the end of the hand.

(2) If a single number is assigned to two different positions, the one defined last takes precedence with former cleared.

(3) The open/close position of the hand and the flag data are also stored as the position data.

(4) Error mode II may occur if the HE command is executed when any of the axes of the robot is near the boundary of its operational space. In such avoid such attitude of the robot.

(5) Alarm (error mode II) occurs if the HE command is executed before the origin setting.

Sample program (MOVEMASTER Command)

```
10 MO 10        'Moves to position 10.
20 DW 10,0,0    'Moves to +X direction by 10 mm
30 HE 11        'Defines above location as position 11.
```


COSIMIR® · Movemaster Command

HLT (HALT)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Interrupts the motion of the robot and the operation of the program.

Input Format

HLT

Explanation

(1) Interrupts the operation of the program and decelerates the robot to a stop. (It becomes the same condition that the external stop signal is input or the STOP switch of the controller front panel is pushed.)

(2) To restart the program, push the START switch, the starting signaling, or execute the RN command. Program restarts from the next line of HLT command.

(3) If the HLT command is directly executed from the personal computer during program running, the is interrupted and the robot stops with deceleration.

(4) The robot does not stop by the HLT command, however, during the execution of the direct motion

command.

Sample program (MOVEMASTER Command)

10 MO 1 'Moves to position 1.

20 HLT 'Stops

30 MO 2 'Moves to position 2.

40 ED 'Ends program.

The program restarts with START switch from line 30.

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Interrupts the movement and the operation of the program of the robot and the additional axis.

Input Format

HLT

Explanation

(1) If there is a movement command preceding this command, the additional axis may not reach its position, since this command will be executed even if additional axis is still moving.

(2) If the next command is designated to execute the next operation before the additional axis has operation, when the HLT command is executed, the moving additional axis will come to stop (Refer to the WRM command).

(3) The additional axis will come to a gradual stop.

Related language

WRM

Sample program (MOVEMASTER Command)

10 MO 1 'Moves to position 1.(Additional axis also moves)

20 HLT 'Stops (Additional axis also stops)

30 MO 2 'Moves to position 2.

40 ED 'Ends program.

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

HO (Home)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Defines the current location and the attitude as origin point.

Input Format

HO [<origin setting approach>]

Term

<Origin setting approach>

Specify the method to set origin in integer value.
0: Mechanical stopper origin
1: Jig origin
2: User-defined origin

Explanation

(1) Establishes the reference position for origin setting.

(2) If you have replaced the robot or changed the combination of robot and controller, you must carry origin setting again using this command. There is approach using teaching box to execute origin setting. See your manual for details.

(3) Change the parameter HOE (see the parameters for

the RV-E2 and RV-E3J and the RV-EN series) to permit origin setting at first, then execute the HO command directly. When the origin setting is completed, return parameter to the former value, otherwise the program can not be started. To find the change operation, refer to your manual.

Relating Parameters

Permits the origin setting from the command (HO).

Parameter name HOE : Origin setting permission parameter

0: Does not permit the use of HO command. (Default)

1: Permits the use of HO command.

Sample program (BASIC)

10 OPEN "COM1 :E83"AS#1 'Opens the RS-232C communication file

'from the personal computer in BASIC.

20 PRINT #1, "HO" 'Executes the "HO" command from the personal computer.

30 END 'Ends

RUN 'Run the BASIC program.

RV-M2, RV-M1

Function:

Establishes the reference position in the Cartesian coordinate system.

Input Format

HO

Explanation

(1) Establishes the reference position for origin setting. Execute the HO command after matching the match marks on each axis of the robot using the teaching box.

(2) If you have replaced the robot or changed the combination of robot and controller, you must carry

origin setting again using this command. There is approach using teaching box to execute origin setting. See your manual for details.

(3) The HO command cannot be executed when bit 4 of the switch SW1 in the drive unit side door is in the lower (OFF) position.

Sample program

```
10 LPRINT "HO"
```


COSIMIR® · Movemaster Command

IC (Increment Counter)

Function:

Adds 1 to the value of the specified counter.

Input Format

IC <Counter number>

Term

<Counter number>

Specify counter number in numeric value.
1 <= counter number <= 99

Explanation

(1) Alarm occurs if the counter value exceeds 32767.

(2) Used to count the number of workpieces and job sequence and to set the number of grid point in the pallet.

(3) The contents of the counter can be changed, compared, or read by the relevant command. (See SC, DC, CP, CR, CL, AN, OR and XO).commands.)

Sample program (MOVEMASTER Command)

```
10 SC 21,15     'Sets value 15 to counter 21.

20 IC 21        'Add 1 to the contents of counter 21.
```


COSIMIR® · Movemaster Command

ID (Input Direct)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Fetches data unconditionally from the external input hand check input.

Input Format

ID [<input bit number>]

Term

<Input bit number>

Specify the bit number of input port in integer value. Fetches data of 16 bits width including the specified 0 <= input bit number <= 32767 (0 for default)

Explanation

(1) Fetches signals from the external equipment, e.g., programmable controller, unconditionally. The data the hand check input can be fetched by specifying the 900th number to the input bit number.

(2) The fetched data is loaded into the internal register and is subsequently used for comparison, bit test, etc. (See EQ, NE, LG, SM and TB commands.)

Sample program (MOVEMASTER Command)

```
100 ID          'Fetches the input data into the register
                'for comparison.
```

110 EQ 100,130 'If the input data equals 100, then jumps to line number 130.

120 ED 'Else ends program.

130 MO 1 'Moves to position 1.

140 ID 100 'Fetches the input data into the register

'for comparison.(Input signals 100 to 115.)

150 TB +0,180 'If the input bit 100 is ON, then jumps line 180.

160 TB+5,200 'If the input bit 105 is ON, then jumps line 200.

170 ED 'Else ends program.

180 MO 2 'Moves to position 2.

190 ED 'Ends program.

200 MO 3 'Moves to position 3.

210 ED 'Ends program.

RV-M2

Function:

Fetches data unconditionally from the external input and hand check input.

Input Format

ID

Explanation

(1) Fetches signals from the external equipment, e.g., programmable controller, unconditionally. The data the hand check input can be fetched by specifying the 900th number to the input bit number.

(2) The fetched data is loaded into the internal register and is subsequently used for comparison, bit test, etc. (See EQ, NE, LG, SM and TB commands.)

Sample program (MOVEMASTER Command)

100 ID 'Fetches the input data into the internal register

'for comparison.

110 EQ 100,130 'If the input data equals 100, then jumps to line number 130.

120 ED 'Else ends program.

130 MO 7 'Moves to position 7.

RV-M1

Function:

Fetches data unconditionally from the external input port.

Input Format

ID

Explanation

(1) Fetches signals from the external equipment, e.g., programmable controller, unconditionally. The data the hand check input can be fetched by specifying the 900th number to the input bit number.

(2) The fetched data is loaded into the internal register and is subsequently used for comparison, bit test, etc. (See EQ, NE, LG, SM and TB commands.)

Sample program (MOVEMASTER Command)

100 ID 'Fetches the input data into the internal register

'for comparison.

110 EQ 100,130 'If the input data equals 100, then jumps to line number 130.

120 ED 'Else ends program.

130 MO 7 'Moves to position 7.

COSIMIR® · Movemaster Command

IN (Input)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function

No operation. Prepared to keep the compatibility with the MOVEMASTER M2 series. Only registration is

Input Format

IN [<input bit number>]

Term

Explanation

This command is prepared for keeping the with the MOVEMASTER M2 series and has no effect on program.

RV-M2

Function:

Fetches data synchronously from the external input and hand check input (using the control signal lines.)

Input Format

IN

Explanation

(1) Causes a signal (parallel data from the external device, e. g. a programmable controller, to be fetched synchronously from the input port, and simultaneously, data to be fetched from the hand check input. At this time, the control signals (!STB and !BUSY signals) must have been connected to the external device.

(2) The external input port data is loaded into the internal comparison register and is subsequently used comparison, bit test, etc. (See the EQ, NE, LG, SM and commands.)

(3) For information on connections refer to your

Sample program (MOVEMASTER Command)

100 IN 'Fetches data from external input

110 EQ 100,130 'Jumps to line number 130 if input equals 100.

120 ED 'Ends program if above condition is met.

130 MO7 'Moves to position 7.

RV-M1

Function:

Fetches data synchronously from the external input (using the control signal lines.)

Input Format

IN

Explanation

(1) Causes a signal (parallel data from the external device, e. g. a programmable controller, to be fetched synchronously from the input port. At this time, the control signals (!STB and !BUSY signals or STB and BUSY signals) must have been connected to the external device.

(2) The external input port data is loaded into the internal comparison register and is subsequently used comparison, bit test, etc. (See the EQ, NE, LG, SM and commands.)

(3) For information on connections refer to your

Sample program (MOVEMASTER Command)

100 IN 'Fetches data from external input port.

110 EQ 100,130 'Jumps to line number 130 if input data equals 100.

120 ED 'Ends program if above condition is not met.

130 MO 7 'Moves to position 7.

COSIMIR® · Movemaster Command

INP (Input)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

The specified counter value, the coordinate value of position number or the data of the specified character string is received according to the PRN command. RS-232-C)

Input Format

INP <channel number>, <counter number/position number/character string number> [, [<contents selection>]]

Term

<Channel number>

Specify the channel number opened by the OPN command.
0 <= channel number <= 2

<Counter number>

Specify counter number.
1 <= counter number <= 99

<Position number>

Specify position number.
1 <= position number <= 999

<Character string number>

Specify character string number in value which $" is added to the head.
$1 <= character string number <= $99

<Contents selection>

Select either counter or position or character string number corresponding to <Counter number/ position number/character string number>.
0: Counter number (Default)
1: Position number
2: character string number

Explanation

(1) This command receives the specified counter value, the coordinate value of the position number or the of the specified character string is received according the PRN command through the RS-232-C port.

(2) The OPN command must be executed first to open RS-232-C channel.

(3) If the counter number is omitted, the data will be read into the internal register. If the character string number is omitted, the data will be read into the character string register. If the position number is omitted, an alarm will occur during execution.

(4) The data is sent from an external device such as a personal computer using the PRM command. The robot program will stop while the data is being red.

(5) The PRM command can be executed before the INP command while the program is running. In that case, sent PRM command data will be registered once, and then will be led into the specified counter, position or character string when the INP command is executed. A max. of 256 characters can be registered in the robot. the PRM command is executed in succession and the number of registered characters exceeds 256
the robot will be set to the "L" level based on the RS-C ER (DRT) and RS (RTS) signal lines (DR (DSR) and CS (CTS) signal lines on the personal computer side). Temporarily stop the data transmission from the computer during this time.

(6) If there is an error in the data sent by the PRN command, an alarm will occur when the INP command executed.

Sample program (MOVEMASTER Command)

10 OPN 2,1 'Opens the RS-232C port.

20 INP 2,1,0 'Reads the data of counter 1 from the RS-232C port.

30 INP 2,5,1 'Reads the data of position 5 from the RS-232C port.

40 IC 1 'Adds 1 to the contents of counter 1.

50 MO 5 'Moves to position 5.

60 OPN 1,1 'Opens the RS-232C port.

70 INP 1,$10,2 'Reads the data of character string 10 from the RS-232C port.

RV-E-Robot (RV-E2, RV-E3J, etc.)

Function:

Reads the data of counter or position transmitted by PRN command. (Using RS-232-C)

Input Format

INP <channel number>, <counter number/position number> [, [<contents selection>]]

Term

<Channel number>

Specify the channel number opened by the OPN command.
0 <= channel number <= 2

<Counter number>

Specify counter number.
1 <= counter number <= 99

<Position number>

Specify position number.
1 <= position number <= 999

<Contents selection>

Select either counter or position or string number corresponding to <Counter number/ position number/character string number>.
0: Counter number (Default)
1: Position number

Explanation

(1) This command receives the specified counter value the coordinate value of the position number received according to the PRN command through the RS-232-C port.

(2) The OPN command must be executed first to open RS-232-C channel.

(3) If the counter number is omitted, the data will be read into the internal register. If the position number is omitted, an alarm will occur during execution.

(4) The data is sent from an external device such as a personal computer using the PRM command. The robot program will stop while the data is being red.

(5) The PRM command can be executed before the INP command while the program is running. In that case, sent PRM command data will be registered once, and then will be led into the specified counter when the command is executed.

(6) If there is an error in the data sent by the PRN command, an alarm will occur when the INP command executed.

Sample program (MOVEMASTER Command)

```
10 OPN 2,1        'Opens the RS-232C port.

20 INP 2,1,0      'Reads the data of counter 1 from the
RS-232C port.

30 INP 2,5,1      'Reads the data of position 5 from the
RS-232C port.

40 IC 1           'Adds 1 to the contents of counter 1.

50 MO 5           'Moves to position 5.
```

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

IP (Increment Position)

Function:

Moves the robot to a predefined position with a number greater than the current one. (Joint interpolation)

Input Format

IP

Explanation

(1) Moves the robot to a predefined position with a position number greater than, and closest to, the one. (See the DP command.)

(2) Alarm occurs if there is no predefined position is greater in position number than the current position.

(3) Even if an alarm occurs, the current position still remains unchanged.

(4) The additional axis will also move if the position includes additional axis data.

Sample program (MOVEMASTER Command)

```
10 MO 5    'Moves to position 5.
20 MO 4    'Moves to position 4.
30 MO 3    'Moves to position 3.
40 IP      'Moves to position 4.
50 IP      'Moves to position 5.
```


COSIMIR® · Movemaster Command

JRC (Joint Roll Change)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Overwrites the current position by adding +/-360 to the joint position of the R-axis. This is done when want to use shortcut control of the R-axis, or when you want to use endless control.

Input Format

JRC <[+] 1/-1>

Term

<+1> Adds 360 degrees to the current joint position on the R-axis.

<-1> Subtracts 360 degrees to the current joint position on the R-axis.

Explanation

(1) When you use this command, you need to change operation range of the R-axis by +/- 720 degrees beforehand. Set the -J6 setting to -720 on the joint operation range parameter JAR (see the parameters for the RV-E2 and RV-E3J and the RV-EN series), and +J6 to 720. For information on how to change parameters, to your manual. If you set a value that is more than +/- 720 degrees, it could result in position slips.

(2) Even though the current coordinate value changes, the robot won't move.

(3) If the joint coordinate value on the R-axis exceeds 720 degrees, the current coordinate value won't and the out-of-bounds alarm will be generated.

(4) If you are using the RV-E4N/E4NM/E4NC, and you want to move the R-axis by more than +/- 180 degrees with this command, the move up to +/- 180 degrees should be by joint interpolation commands. If you use linear interpolation commands or circular interpolation commands, when you execute the program, alarms will be generated. If you are using RV-E5NJ/E5NJM/E5NJC, you don't have such a restriction.

(5) If you use this command while doing continuous movements without acceleration/deceleration

it will not be able to do continuous movements with acceleration/deceleration. (Refer to CNT of the SP command)

(6) When the controller power is a disconnected state, if you move the R axis by more than +/- 7 turns, the next time the power source is supplied, the position of R's will slip. In this case, it is necessary to reset the origin for only the R-axis.

Refer to your manual in how to reset the origin.

Sample program (MOVEMASTER Command)

```
10 MO 1      'Move to position 1 (R-axis is -150 degrees)

20 MO 2      'Move to position 2 (R-axis is +170

30 JRC -1    'Subtract 360 degrees from the current position

             'on the R-axis(R-axis is -190 degrees)

40 MO 1      'Move to position 1
```

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

LG (If Larger)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

This command compares the value of the internal register with a specified value. If larger, the program jump. The character string register and the numbers of characters in a specified character string are
If the character string register is larger, the program

jump.

Input Format

LG <compared value/character string number>,
<branching line number>

Term

<Compared value>

Specify the value compared with the register.
-32768 <= Compared value (decimal) <=
& 8000 <= Compared value (hexadecimal) <=
& 7FFF
@1 <= counter number <= @99

<Character string number>

Specify character string number in value which "$" is added to the head.
$1 <= character string number <= $99

<Branching line number>

Specify the line number to which the jumps when the value of the internal is larger than compared value.
1 <= branching line number <= 9999

Explanation

<When compared value is specified>

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value is larger than the compared value (i.e., when the condition is met), the program jumps to the specified line. Otherwise (i.e., when the condition is not met), the program continues sequence.

(3) A value can be loaded into the internal register by executing the input command (See ID) for the external input data or by executing the compare counter command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed by "&".

<When character string number is specified>

(1) The conditions will jump depending on the data from an external source or the number of characters in specified character string.

(2) If the number of characters in the character string register is larger than the number of characters in a specified character string (when the conditions are established), the program will jump to the specified number. If the number is smaller (when conditions are not established), the next line will be executed. If the specified line number is not registered, an alarm will occur when jumping.

(3) By executing an INP command, the data input from external device will be set in the character string register. The details of the character string number be set by executing a CP command. Thus, when executing condition jumping, one of these commands must be executed first.

Sample program (MOVEMASTER Command)

```
100 ID              'Fetches the data from the external
input port.

110 LG 100,130      'If the input data is larger than 100,
jumps to line 130.

120 ED              'Else program ends.

130 MO 1            'Moves to position 1.

140 OPN 1,1         'Opens the RS-232C port.

150 INP 1, ,2       'Reads the data of character string
register

                    'from the RS-232C port.

160 LG $5,200       'Jumps to line 200 if the data length
large

                    'than character string number 5.

:
```

200 ED 'Ends program.

RV-E-Robot (RV-E2, RV-E3J, etc.)

Function:

This command compares the value of the internal register with a specified value. If larger, the program jump.

Input Format

LG <compared value>, <branching line number>

Term

<Compared value>

Specify the value compared with the register.
-32768 <= Compared value (decimal) <=
& 8000 <= Compared value (hexadecimal) <=
& 7FFF

<Branching line number>

Specify the line number to which the jumps when the value of the internal is larger than compared value.
1 <= branching line number <= 9999

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value is larger than the compared value (i.e., when the condition is met), the program jumps to the specified line. Otherwise (i.e., when the condition is not met), the program continues sequence.

(3) A value can be loaded into the internal register by executing the input command (See ID) for the external input data or by executing the compare counter command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed

"&".

Sample program (MOVEMASTER Command)

100 ID 'Fetches the data from the external input port.

110 LG 100,130 'If the input data is larger than 100, jumps to line 130.

120 ED 'Else program ends.

130 MO 1 'Moves to position 1.

RV-M2

Function:

This command compares the value of the internal register with a specified value. If larger, the program jump.

Input Format

LG <compared value>, <branching line number>

Term

<Compared value>

Specify the value compared with the register.
-32768 <= Compared value (decimal) <=
& 8000 <= Compared value (hexadecimal) <=
& 7FFF

<Branching line number>

Specify the line number to which the jumps when the value of the internal is larger than compared value.
1 <= branching line number <= 3584

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value is larger than the compared value (i.e., when the condition is met), the program jumps to the specified line. Otherwise (i.e.,

when the condition is not met), the program continues sequence.

(3) A value can be loaded into the internal register by executing the input command (See ID and IN) for the external input data or by executing the compare command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed "&".

Sample program (MOVEMASTER Command)

100 ID 'Fetches the data from the external input port.

110 LG 100,130 'If the input data is larger than 100, jumps to line 130.

120 ED 'Else program ends.

130 MO 1 'Moves to position 1.

RV-M1

Function:

This command compares the value of the internal register with a specified value. If larger, the program jump.

Input Format

LG <compared value>, <branching line number>

Term

<Compared value>

Specify the value compared with the register.
-32767 <= Compared value (decimal) <=
& 8001 <= Compared value (hexadecimal) <=
& 7FFF

<Branching line number>

Specify the line number to which the jumps when the value of the internal

is larger than compared value.
1 <= branching line number <= 2048

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value is larger than the compared value (i.e., when the condition is met), the program jumps to the specified line. Otherwise (i.e., when the condition is not met), the program continues sequence.

(3) A value can be loaded into the internal register by executing the input command (See ID and IN) for the external input data or by executing the compare command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed "&".

Sample program (MOVEMASTER Command)

100 ID 'Fetches the data from the external input port.

110 LG 100,130 'If the input data is larger than 100, jumps to line 130.

120 ED 'Else program ends.

130 MO 1 'Moves to position 1.

COSIMIR® · Movemaster Command

LR * (Line Read)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Reads the program of the specified line number. (Using RS-232C)

Input Format

LR [<Line number>]

Term

< Line number >

Specify the line number to be read
0 <= Line number <= 9999
((If omitted, reads the current line number stopping)

Explanation

(1) Outputs the program of the specified line number the current stopping line number) from the RS-232C

(2) The output format is ASCII coded as follows;

- If you specify the line number, => Program content is read.

- If you omit the line number (or specify zero) => stopping line number is read.

(3) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(4) The hexadecimal 0D is read out when the specified line has not been defined.

(5) If an alarm takes place, you can confirm the line number in which the alarm occurs by executing the LR command without line number.

* This command can only be executed directly. It be used in a program.

Sample program (BASIC)

```
10 OPEN "COM1 :E83"AS#1            'Opens the RS-232C
communication file

                                  'from the personal
computer in BASIC.
```

20 INPUT "Start line = ";S 'Enter the top line number that you want to read.

30 INPUT "End line = ";E 'Enter the last line number that you want to read.

40 FOR I=S TO E 'Repeatedly

50 PRINT #1; "LR" + STR$ (I) 'Transmit "LR" + "line number" to the controller side.

60 LINE INPUT #1,A$ 'Saves the received data to A $.

70 IF A$="" THEN 90 'If there is no data, jumps to line 90.

80 PRINT I ; :PRINT A$ 'Displays the data on the personal computer screen.

90 NEXT 'Repeats and jumps to line 40.

100 END

RUN 'Run the BASIC

Start line = 1 'Insert the start line (1)

End line = 5 'Insert the end line (5)

1 NT

2 MO 6

3 MO 4

4 MO 3

5 ED

RV-M2

Function:

Reads the program of the specified line number. (Using RS-232C)

Input Format

LR <Line number>

Term

< Line number >

Specify the line number to be read
0 <= Line number <= 3584

Explanation

(1) Outputs the program of the specified line number the current stopping line number) from the RS-232C

(2) The output format is ASCII coded.

(3) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(4) The hexadecimal 0D is read out when the specified line has not been defined.

(5) The parameter entered in hexadecimal using "&" is output in decimal equivalent (e.g. "OD &FF" is into "OD 255").

* This command can only be executed directly. It be used in a program.

Sample program (BASIC)

```
10 OPEN "COM1 :9600, E, 7, 2"AS#1    'Opens the RS-
communication file

                                     'from the personal
computer in BASIC.

20 INPUT "Start line = ";S            'Enter the top
number that you want to read.

30 INPUT "End line = ";E              'Enter the last
number that you want to read.

40 FOR I=S TO E                       'Repeatedly

50 PRINT #1; "LR" + STR$ (I)          'Transmit "LR" +
"line number" to the controller side.

60 LINE INPUT #1,A$                   'Saves the
data to A $.
```

70 IF A$="" THEN 90 'If there is no data, jumps to line 90.

80 PRINT I ; :PRINT A$ 'Displays the data on the personal computer screen.

90 NEXT 'Repeats and jumps to line 40.

100 END

RUN 'Run the BASIC program.

Start line = 1 'Insert the start (1)

End line = 5 'Insert the end line (5)

1 NT

2 MO 6

3 MO 4

4 MO 3

5 ED

RV-M1

Function:

Reads the program of the specified line number. (Using RS-232C)

Input Format

LR <Line number>

Term

< Line number >

Specify the line number to be read
0 <= Line number <= 2048

Explanation

(1) Outputs the program of the specified line number the current stopping line number) from the RS-232C

(2) The output format is ASCII coded.

(3) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by a personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(4) The hexadecimal 0D is read out when the specified line has not been defined.

(5) The parameter entered in hexadecimal using "&" is output in decimal equivalent (e.g. "OD &FF" is converted into "OD 255").

* This command can only be executed directly. It cannot be used in a program.

Sample program (BASIC)

```
10 OPEN "COM1 :9600, E, 7, 2"AS#1    'Opens the RS-
communication file

                                     'from the personal
computer in BASIC.

20 INPUT "Start line = ";S            'Enter the top
number that you want to read.

30 INPUT "End line = ";E              'Enter the last
number that you want to read.

40 FOR I=S TO E                       'Repeatedly

50 PRINT #1; "LR" + STR$ (I)          'Transmit "LR" +
"line number" to the controller side.

60 LINE INPUT #1,A$                   'Saves the
data to A $.

70IF A$="" THEN 90                    'If there is no
jumps to line 90.

80 PRINT I ; :PRINT A$                'Displays the data
on the personal computer screen.

90 NEXT                               'Repeats and
to line 40.

100 END

RUN                                   'Run the BASIC
```

program.

Start line = 1 'Insert the start (1)

End line = 5 'Insert the end line (5)

1 NT

2 MO 6

3 MO 4

4 MO 3

5 ED

COSIMIR® · Movemaster Command

MA (Move Approach)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.), RV-M2

Function:

Moves the hand tip to the added position. (Linear interpolation)

Input Format

MA <position number(a)>, <position number(b)> [, [<O/C>]]

Term

<Position number (a)>

Specify the position number to be added. (The reference position)

<Position number (b)>

Specify the position number to add. (The increments position)
1 <= position number (a) (b) <= 999

<O/C>

Specify open or close state of the hand.
O: Hand open
C: Hand close

Explanation

(1) Moves to the added position, i.e., the coordinates of positions (a) an destination, although positions (a) and (b) remain unchanged after execu the SE command.)

(2) If the open/close state of the hand has been specified, the robot mo control command. If it has not been specified, the hand state in positior

(3) If the calculating results exceed the robot's operational space, alarm moves.

(4) Alarm also takes place if positions (a) and (b) have not been defined.

(5) The position of the hand tip is decided by the tool length currently e

Sample program (MOVEMASTER Command)

(1) 6-axis type

```
10 HE 1                  'Sets the current coordinates to position 1.
20 PD 5,0,0,30,0,0,0     'Defines the Z coordinate of position 5 as 30 r
30 MA 1,5,O              'Moves to the position that only Z direction a
                         'to coordinate value of position 1 by 30 mm wi
```

(2) 5-axis type

```
10 HE 1                  'Sets the current coordinates to position 1.
20 PD 5,0,0,30,0,0       'Defines the Z coordinate of position 5 as 30
30 MA 1,5,O              'Moves to the position that only Z direction a
                         'to coordinate value of position 1 by 30 mm wi
                         'Coordinates values of position 1 and position 5
```

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Moves to a position determined by adding the previously defined coordin the coordinate values of a designated position. (Joint interpolation)

Input Format

MA <position number(a)>, <position number(b)> [, [<O/C>]]

Term

<Position number (a)>

Specify the position number to be added. (The reference position)

<Position number (b)>

Specify the position number to add. (The increments position)
1 <= position number (a) (b) <= 999

<O/C>

Specify open or close state of the hand.
O: Hand open
C: Hand close

Explanation

(1) The additional axis will also move if the additional position includes

(2) The table below shows how the existence/non-existence of position determines the positions after the MA command is executed.

(3) The original position (a) will not change.

		Pattern 1	Pattern 2
Position (a)	**Robot**	Exists	Exists
Position (a)	**Additional axis**	Doesn,t exist	Exists
Position (b)	**Robot**	Exists	Exists
Position (b)	**Additional axis**	Exists	Doesn,t e
MA (a),(b)	**Robot**	(a) <- (a) + (b)	(a) <- (a)
MA (a),(b)	**Additional axis**	(a) <- (b)	(a) <- (a)

Sample program (MOVEMASTER Command)

(1) 6-axis type

10 HE 1 'Sets the current coordinates to position 1.

20 PD 5,0,0,0,0,0,0,30,0 'Set the additional axis components of posi

30 MA 1,5,O 'Open the hand and move it to a position t
30 [mm or deg.] added to its additional axis components.

(2) 5-axis type

10 HE1 'Sets the current coordinates to position 1.

20 PD 5,0,0,0,0,0,30,0 'Set the additional axis components of posit

30 MA 1,5,O 'Open the hand and move it to a position th
30 [mm or deg.] added to its additional axis components.

Note: Coordinates values of position 1 and position 5 do not change.

RV-M1

Function:

Moves the hand tip to the added position. (Linear interpolation)

Input Format

MA <position number(a)>, <position number(b)> [, [<O/C>]]

Term

<Position number (a)>

Specify the position number to be added. (The reference position)

<Position number (b)>

Specify the position number to add. (The increments position)
1 <= position number (a) (b) <= 629

<O/C>

Specify open or close state of the hand.
O: Hand open
C: Hand close

Explanation

(1) Moves to the added position, i.e., the coordinates of positions (a) an destination, although positions (a) and (b) remain unchanged after exec the SE command.)

(2) If the open/close state of the hand has been specified, the robot mo control command. If it has not been specified, the hand state in positior

(3) If the calculating results exceed the robot's operational space, alarm moves.

(4) Alarm also takes place if positions (a) and (b) have not been defined.

(5) The position of the hand tip is decided by the tool length currently e

Sample program (MOVEMASTER Command)

10 HE 1	'Sets the current coordinates to position 1.
20 PD 5,0,0,30,0,0	'Defines the Z coordinate of position 5 as 30 mm.
30 MA 1,5,O	'Moves to the position that only Z direction added
	'to coordinate value of position 1 by 30 mm with th
	'Coordinates values of position 1 and position 5 do

COSIMIR® · Movemaster Command

MC (Move Continuous)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.), RV-M2

Function:

Moves the robot continuously through the predefined intermediate points between two specified position numbers. (Linear interpolation)

Input Format

MC <position number(a)>, <position number(b)> [, [<O/C>]]

Term

<Position number (a)>

Specify the top position number moving continuous.

<Position number (b)>

Specify the last position number moving continuous.
1 <= position number (a) (b) <= 999
| Position number (a) - position number (b)| <= 99

<O/C>

Specify open or close state of the hand.
(If omitted, the hand data of each position is valid.)
O: Hand open
C: Hand close

Explanation

(1) Moves the robot along the series of positions via (a) (b) without acceleration and deceleration. (Linear interpolation)

(2) Depending on whether position number of (a) is greater than that of (b), or vice versa, the robot moves through the intermediate points in descending or ascending order. The robot decelerates to a stop as it reaches the end position.

(3) When the hand open/close setting has been done, hand control is executed before the movement.

(4) Since the robot does not accelerate or decelerate during motion, alarm may occur when the path involves great change in direction of any of the joints at high speed.

(5) The speed of travel during linear interpolation is determined by the SP or SD command. (Hand tip at constant speed)

(6) Alarm occurs if specified positions (a) and (b) have not been defined or if the difference between the position numbers (a) and (b) exceeds 99.

(7) Alarm also takes place during movement if the movement path goes beyond the robot's operational space.

Sample program (MOVEMASTER Command)

10 SP 10 'Sets speed to 10.

20 MO 1 'Moves to position 1 in joint

30 MC 5,9 'Moves continuously from position 5 to 9 in linear interpolation.

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Moves the robot cont inuously through the predefined intermediate points between two specified position numbers (a) (b) without acceleration and deceleration. (Linear interpolation)

Input Format

MC <position number(a)>, <position number(b)> [, [<O/C>]]

Term

<Position number (a)>

Specify the top position number moving continuous.

<Position number (b)>

Specify the last position number moving continuous.
1 <= position number (a) (b) <= 999
| Position number (a) - position number (b)| <= 99

<O/C>

Specify open or close state of the hand.
(If omitted, the hand data of each position valid.)
O: Hand open
C: Hand close

Explanation

(1) The additional axis will move it the position additional axis data.

(2) If there are additional axis movement components the destination position while the additional axis is still moving, after making a gradual stop, the next interpolation will be executed.

(3) When the robot moves in response to the movement of the additional axis (travel axis, etc.) the locus of the tip of the hand may not be straight.

Sample program (MOVEMASTER Command)

10 SP 10 'Sets speed to 10.

20 MO 1 'Moves to position 1 in joint interpolation.

30 MC 5,9 'Moves continuously from position 5 to 9 in linear interpolation.

RV-M1

Function:

Moves the robot continuously through the predefined intermediate points between two specified position numbers. (Linear interpolation)

Input Format

MC <position number(a)>, <position number(b)>

Term

<Position number (a)>

Specify the top position number moving continuous.

<Position number (b)>

Specify the last position number moving continuous.
1 <= position number (a) (b) <= 999
| Position number (a) - position number (b)| <= 99

Explanation

(1) Moves the robot along the series of positions via (a) (b) without acceleration and deceleration. (Linear interpolation)

(2) Depending on whether position number of (a) is greater than that of (b), or vice versa, the robot moves through the intermediate points in descending or ascending order. The robot decelerates to a stop as it reaches the end position.

(3) When the hand open/close setting has been done, hand control is executed before the movement.

(4) Since the robot does not accelerate or decelerate during its movement through the intermediate points, avoid movement paths that involve a great change in orientation of any of the axes of motion; otherwise, resultant oscillations adversely affect the robot mechanical. (The maximum speed of travel using command MC is equivalent to SP4.).

(5) The speed of travel during linear interpolation is determined by the SP or SD command. (Hand tip at constant speed)

(6) Alarm occurs if specified positions (a) and (b) have not been defined or if the difference between the position numbers (a) and (b) exceeds 99.

(7) Alarm also takes place during movement if the movement path goes beyond the robot's operational space.

Sample program (MOVEMASTER Command)

```
10 SP 10        'Sets speed to 10.

20 MO 1         'Moves to position 1 in joint

30 MC 5,9       'Moves continuously from position 5 to 9
in linear interpolation.
```


COSIMIR® · Movemaster Command

MJ (Move Joint)

RV-2AJ, RV-E-Robot with 5 axes (RV-E3J, RV-E5NM, etc.)

Function:

Turns each joint the specified angle from the current position. (Joint interpolation)

Input Format

MJ [<waist joint angle>], [<shoulder joint angle>], [<elbow joint angle>], [<pitch joint angle>], [<roll joint angle>]

Term

<Each joint angle>

Specify relative amount of each joint from the current position.

Explanation

(1) The least increment of the turning angle is 0.01 degree, e.g., specify 15.02 for 15.02 degree.

(2) The open/close state of the hand does not change before and after the movement. Alarm occurs before joint motion if any turning angle entry exceeds the robot's operational space.

(3) The default turning angle is 0.

(4) For the positive and negative directions of each of motion refer to your manual.

Sample program (MOVEMASTER Command)

10 MJ 90,0,0,0,0 'Turns the waist joint + 90

20 MJ 0,-30,0,0,0 'Turns the shoulder joint - 30 degrees.

RV-1A, RV-E-Robot with 6 axes(RV-E2, RV-E4N, etc.)

Function:

Turns each joint the specified angle from the current position. (Joint interpolation)

Input Format

MJ [<waist joint angle>], [<shoulder joint angle>], [<elbow joint angle>], [<twist joint angle>], [<pitch angle>], [<roll joint angle>]

Term

<Each joint angle>

Specify relative amount of each joint from the current position.

Explanation

(1) The least increment of the turning angle is 0.01 degree, e.g., specify 15.02 for 15.02 degree.

(2) The open/close state of the hand does not change before and after the movement. Alarm occurs before joint motion if any turning angle entry exceeds the robot's operational space.

(3) The default turning angle is 0.

(4) For the positive and negative directions of each joint of motion refer to your manual.

Sample program (MOVEMASTER Command)

10 MJ 90,0,0,0,0,0 'Turns the waist joint + 90 degrees.

20 MJ 0,-30,0,0,0,0 'Turns the shoulder joint - 30 degrees.

30 MJ 0,0,0,20,0,0 'Turns the elbow joint + 20 degrees.

RV-M1, RV-M2

Function:

Turns each joint the specified angle from the current position. (Joint interpolation)

Input Format

MJ [<waist joint angle>], [<shoulder joint angle>], [<elbow joint angle>], [<pitch joint angle>], [<roll joint angle>]

Term

<Each joint angle>

Specify relative amount of each joint from the current position.

Explanation

(1) The least increment of the turning angle is 0.1 degree, e.g., specify 15.2 for 15.2 degree.

(2) The open/close state of the hand does not change before and after the movement. Alarm occurs before joint motion if any turning angle entry exceeds the robot's operational space.

(3) The default turning angle is 0.

(4) For the positive and negative directions of each of motion refer to your manual.

Sample program (MOVEMASTER Command)

10 MJ 90,0,0,0,0 'Turns the waist joint + 90

20 MJ 0,-30,0,0,0 'Turns the shoulder joint - 30
degrees.

COSIMIR® · Movemaster Command

ML

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function

No operation. Prepared to keep the compatibility with the MOVEMASTER

Input Format

ML

Explanation

This command is prepared for keeping the compatibility with the MOVEM

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function

Moves the additional axis as an independent unit.

Input Format

ML [<Movement amount of First additional axis>] [, <Movement amount o

Terms

< Movement amount of First additional axis >	Specifies the movement amo (0 when omitted. The unit is
< Movement amount of Second additional axis >	Specifies the movement amo axis. (0 when omitted. The unit is

Explanation

(1) Moves the additional axis from the current position only by the speci

(2) If omitted, the movement amount is 0.

(3) The unit for the movement amount can be specified in the paramete

(4) An alarm will occur if the destination position exceeds the operation

(5) Nonexistent movement amounts for the additional axis are ignored.

(6) If the next movement command following this command does not co
movement command will be executed even though the additional axis m

Related parameters

AXUN

Related language

WRM

Sample program (MOVEMASTER Command)

10 MO 1 'Move to position 1

20 ML 100 'Move the additional axis 1 to 100 [mm or deg.] in the + dir

30 MO 2 'Move to position 2

40 ML 'Do not move the additional axis

RV-M2 with additional axis

Function

Allows the Locomotive to be operated independently.

Input Format

ML <Travel distance (mm)>

Sample Input

ML +105.3

Explanation

(1) The least input increment of the travel distance is 0.1mm. (Example

(2) Allows the Locomotive to move independently along the specified dis

direction and a negative value for a travel in the negative direction.

(3) Error mode II is caused if the specified position exceeds the operating

(4) When this command is used, the Locomotive coordinate data changes
coordinates data (which indicates the current robot position, e.g. X, Y, Z

Sample Program

10 LPRINT "MO 1" 'Move to position 1.

20 LPRINT "ML 100" 'Move the Locomotive 100mm in the positive di

30 LPRINT "DW 0,50,0" 'Move the hand end 50mm in the Y-axis positiv

40 LPRINT "ML -100" 'Move the Locomotive 100mm in the negative c

COSIMIR® · Movemaster Command

MO (Move)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.), RV-M2

Function:

Moves the hand tip to the specified position. (Joint interpolation)

Input Format

MO <position number> [, [<O/C>]]

Term

<Position number>

Specify the destination position number in integer value.
1 <= position number <= 999

<O/C>

Specify open or close state of the hand. (If omitted, the hand state of the position is valid)

O: Hand open
C: Hand close

Explanation

(1) Moves the tip of hand to the coordinates of the specified position by joint interpolation. The hand tip decided by the tool length currently established. (see command)

(2) If open/close state of the hand has been specified, the robot moves after executing the hand control command.

(3) If it has not been specified, the definition of the specified position is executed.

(4) Alarm takes place if the specified position has not been predefined or the movement exceeds the robot's operational space.

Sample program (MOVEMASTER Command)

```
10 SP 10        'Sets speed to 10.

20 MO 20,C      'Moves to position 20 with hand closed.

30 MO 30,O      'Moves to position 30 with hand opened.
```

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Moves the hand tip to the specified position. (Joint interpolation)

Input Format

MO <position number> [, [<O/C>]]

Term

<Position number>

Specify the destination position number in integer value.
1 <= position number <= 999

<O/C>

Specify open or close state of the hand. (If omitted, the hand state of the position is

valid)
O: Hand open
C: Hand close

Explanation

(1) If the position includes additional axis data, the additional axis will also move.

Sample program (MOVEMASTER Command)

10 ADL 100,100,1 'Set the acceleration time for additional axis 1 to 0.1 sec.

20 MO 20,C 'Moves to position 20 with hand closed.

30 MO 30,O 'Moves to position 30 with hand opened.

RV-M1

Function:

Moves the hand tip to the specified position. (Joint interpolation)

Input Format

MO <position number> [, [<O/C>]]

Term

<Position number>

Specify the destination position number in integer value.
1 <= position number <= 629

<O/C>

Specify open or close state of the hand. (If omitted, the hand state of the position is valid)
O: Hand open
C: Hand close

Explanation

(1) Moves the tip of hand to the coordinates of the specified position by joint interpolation. The hand tip is decided by the

tool length currently established. (see TL command)

(2) If open/close state of the hand has been specified, the robot moves after executing the hand control command.

(3) If it has not been specified, the of the specified position is executed.

(4) Alarm takes place if the specified has not been predefined or the movement exceeds the robot's operational space.

Sample program (MOVEMASTER Command)

10 SP 3 'Sets speed to 3.

20 MO 20,C 'Moves to position 20 with closed.

30 MO 30,O 'Moves to position 30 with opened.

COSIMIR® · Movemaster Command

MP (Move Position)

RV-2AJ, RV-E-Robot with 5 axes (RV-E3J, RV-E5NM, etc.)

Function:

Moves the tip of hand to a position whose coordinates (position and angle) have been specified. (Joint interpolation)

Input Format

MP [< X coordinate value>], [< Y coordinate value>], [< coordinate value>], [<A turning angle>], [<B turning angle>] [, [<R/L>] [, [<A/B>]]]

Term

<X, Y, Z coordinate>

Specify the position in XYZ coordinates (mm) of the robot. (Zero for default)

<A, B turning angle>

Specify the turning angle of roll and pitch joints in XYZ coordinates (degree) of the robot. (Zero for default)

<R/L>

Specify the structure flag of the robot.
or Left)
R: Right (Default)
L: Left

<A/B>

Specify the structure flag of the robot.
or Below))
A: Above(Default)
B: Below

Explanation

(1) The least increment of the coordinate value is 0.01 mm or 0.01 degree.

(2) If the structure flag has not been specified, Right Above flag is selected.

(3) If the specified value exceeds the robot's space, alarm occurs at the execution of the MO command.

(4) The open or close state of the hand remains the before and after the movement.

(5) The position of hand tip is decided by the tool currently established.

Sample program (MOVEMASTER Command)

10 MP 400,0,300,0,0 'Moves to the specified coordinates.

20 MP 200,200,500,0,0,R 'Moves to the specified coordinates.

'(Structure flags are also specified.)

RV-1A, RV-E-Robot with 6 axes(RV-E2, RV-E4N, etc.)

Function:

Moves the tip of hand to a position whose coordinates (position and angle) have been specified. (Joint interpolation)

Input Format

MP [<X coordinate value>], [<Y coordinate value>], [<Z coordinate value>], [<A turning angle A>], [<B turning angle >], [<C turning angle >], [, [<R/L>] [, [<A/B>] [, [<N/F>]]]]

Term

<X, Y, Z coordinate>

Specify the position in XYZ coordinates (mm) of the robot. (Zero for default)

<A, B, C turning angle>

Specify the turning angle of roll and pitch joints in XYZ coordinates (degree) of the robot. (Zero for default)

<R/L>

Specify the structure flag of the robot.
or Left)
R: Right (Default)
L: Left

<A/B>

Specify the structure flag of the robot.
or Below))
A: Above(Default)
B: Below

<N/F>

Specify the structure flag of the robot. (Non flip or Flip)
N: Non flip (Default)
F: Flip

Explanation

(1) The least increment of the coordinate value is 0.01

mm or 0.01 degree.

(2) If the structure flag has not been specified, Right Above flag is selected.

(3) If the specified value exceeds the robot's operational space, alarm occurs at the execution of the MO command.

(4) The open or close state of the hand remains the before and after the movement.

(5) The position of hand tip is decided by the tool length currently established.

Sample program (MOVEMASTER Command)

10 MP 400,0,300,0,0,0 'Moves to the coordinates.

20 MP 200,200,500,0,0,0,R,A,N 'Moves to the coordinates.

'(Structure flags are also specified.)

RV-E-Robot with 5 axes (RV-E3J, RV-E5NM, etc.) and with additional axis

Function:

Moves the tip of hand to a position whose coordinates (position and angle) have been specified. (Joint interpolation)

Input Format

MP [< X coordinate value>], [< Y coordinate value>], [< coordinate value>], [<A turning angle>], [<B turning angle>] [, <First additional axis> [, <Second additional axis>]] [, [<R/L>] [, [<A/B>]]]

Term

<X, Y, Z coordinate>

Specify the position in XYZ coordinates (mm) of the robot. (Zero for default)

<A, B turning angle>

Specify the turning angle of roll and pitch joints in XYZ coordinates (degree) of the robot. (Zero for default)

<First additional axis>

Specifies the moving position for First additional axis (If omitted, value is the current position)
The unit is determined by the parameter, AXUN.

<Second additional axis>

Specifies the moving position for Second additional axis (If omitted, value is the current position)

<R/L>

Specify the structure flag of the robot.
or Left)
R: Right (Default)
L: Left

<A/B>

Specify the structure flag of the robot.
or Below))
A: Above(Default)
B: Below

Explanation

(1) The additional axis will also move if the position includes additional axis data.

(2) If the additional axis is omitted, only the robot will move.

(3) The coordinate values of nonexistent additional will be ignored.

(4) The unit for the additional axis is specified in the parameter, AXUN.

Related parameters:

AXUN

Sample program (MOVEMASTER Command)

```
10 MP 400,0,300,0,0,100,0        'Moves to the specified
```

coordinate values (First additional axis moves 100 [mm deg.]).

RV-E-Robot with 6 axes(RV-E2, RV-E4N, etc.) and with additional axis

Function:

Moves the tip of hand to a position whose coordinates (position and angle) have been specified. (Joint interpolation)

Input Format

MP [<X coordinate value>], [<Y coordinate value>], [<Z coordinate value>], [<A turning angle A>], [<B turning angle >], [<C turning angle >], [, <First additional axis> <Second additional axis>]] [, [<R/L>] [, [<A/B>] [, [<N/F>]]]]

Term

<X, Y, Z coordinate>

Specify the position in XYZ coordinates (mm) of the robot. (Zero for default)

<A, B, C turning angle>

Specify the turning angle of roll and pitch joints in XYZ coordinates (degree) of the robot. (Zero for default)

<First additional axis>

Specifies the moving position for First additional axis (If omitted, value is the current position)
The unit is determined by the parameter, AXUN.

<Second additional axis>

Specifies the moving position for Second additional axis (If omitted, value is the current position)

<R/L>

Specify the structure flag of the robot.
or Left)
R: Right (Default)

L: Left

<A/B>

Specify the structure flag of the robot.
or Below))
A: Above(Default)
B: Below

<N/F>

Specify the structure flag of the robot. (Non flip or Flip)
N: Non flip (Default)
F: Flip

Explanation

(1) The additional axis will also move if the position includes additional axis data.

(2) If the additional axis is omitted, only the robot will move.

(3) The coordinate values of nonexistent additional will be ignored.

(4) The unit for the additional axis is specified in the parameter, AXUN.

Related parameters:

AXUN

Sample program (MOVEMASTER Command)

10 MP 400,0,300,0,0,0,100,0 'Moves to the coordinate values (First additional axis moves 100 [mm deg.]).

RV-M2, RV-M1

Function:

Moves the tip of hand to a position whose coordinates (position and angle) have been specified. (Joint interpolation)

Input Format

MP [< X coordinate value>], [< Y coordinate value>], [< coordinate value>], [<Pitch angle>], [<Roll angle>]

Term

<X, Y, Z coordinate>

Specify the position in XYZ coordinates (mm) of the robot. (Zero for default)

<Roll/Pitch angle>

Specify the turning angle of roll and pitch joints in XYZ coordinates (degree) of the robot. (Zero for default)

Explanation

(1) The least increment of the coordinate value is 0.1 or 0.1 degree.

(2) If the structure flag has not been specified, Right Above flag is selected.

(3) If the specified value exceeds the robot's space, alarm occurs at the execution of the MO command.

(4) The open or close state of the hand remains the before and after the movement.

(5) The position of hand tip is decided by the tool currently established.

Sample program (MOVEMASTER Command)

```
10 MP 400,0,300,0,0           'Moves to the specified coordinates.

20 MP 200,200,500,0,0,R       'Moves to the specified coordinates.

                              '(Structure flags are also specified.)
```

RV-M2 with additional axis

Function

Moves the hand end to a position whose coordinates (position and angle) have been specified. (Articulated interpolation)

Input Format

MP [< X-axis coordinate value>], [< Y-axis coordinate value>], [< Z-axis coordinate value>], [<Pitch angle>], [<Roll angle>], [<Locomotive coordinate>]

Sample Input

MP 0, 380, 300, -70, -40, 520

Explanation

(1) The least input increment of the coordinate values 0.1 mm or 0.1 degree(e.g. specify 20.1 for 20.1mm).

(2) Error mode II occurs if the specified coordinates exceeds the robot,s operational space.

(3) Any coordinate value defaults to 0.

(4) The open/close state of the hand remains before and after the movement.

(5) The position of the hand end depends on the tool length. (See command TL.)

Sample Program

```
10 LPRINT "PD 1, 0, 550, 450, -50, 0, 280"

20 LPRINT "MO 1"

30 LPRINT "MP 0, 550, 430, -50, 0, 280"
```

* In the above example, the hand end first moves to position 1, then goes 20mm down in the Z-axis according to line number 30, with the open/close state of the hand remaining unchanged.

COSIMIR® · Movemaster Command

MPB (Move Playback)

RV-1A, RV-2AJ

Function: No operation.

No operation. This command isn't available for these robottypes.

RV-E-Robot with 5 axes (RV-E3J, RV-E5NM, etc.)

Function:

Moves to the specified position with specified interpolation, specified speed, specified timer, and specified input and output signal.

Input Format

MPB [<speed>], [<timer>], [<output ON>], [<output [<input ON>], [<input OFF>] [, [<interpolation>], [<X coordinate>], [<Y coordinate>], [<Z coordinate>], [<A turning angle>], [<B turning angle>],[, [<R/L>] [, [, [<O/C>]]

Term

<Speed>

Specify the interpolation speed to the destination position.
0 <= speed <= 32767(Joint interpolation: %, Linear interpolation: mm/s)

<Timer>

Set timer at the destination position after the movement.
0 <= timer <= 255

<Output ON>

Set the output signal that turns ON.
0 <= output ON (Hexadecimal) <= & FFFF
1: Setting, 0: Not setting

<Output OFF>

Set the output signal that turns OFF.
0 <= output OFF (Hexadecimal) <= & FFFF
1: Setting, 0: Not setting

<Input ON>

Set the input waiting signal that turns ON.
0 <= input ON (Hexadecimal) <= & FFFF

1: Setting,0: Not setting

<Input OFF>

Set the input waiting signal that turns OFF.
0 <= input OFF (Hexadecimal) <= & FFFF
1: Setting,0: Not setting

<Interpolation>

Specify the interpolation mode to the destination position.
Joint interpolation: 0 (Default)
Linear interpolation: 1
Circular interpolation: 2

<X, Y, Z coordinate>

Specify the location (mm) in XYZ coordinates the robot.
(Zero for default)

<A, B turning angle >

Specify the turning angle around roll(A), pitch (B) axes in XYZ coordinates (degree) of the robot. (0 for default)

<R/L>

Specify the structure flag of the robot. (Right Left)
R: Right (Default)
L: Left

<A/B>

Specify the structure flag of the robot. (Above or Below)
A: Above (Default)
B: Below

<O/C>

Specify open or close state of hand 1.
O: Hand 1 open (Default)
C: Hand 1 close

Explanation

(1) The least increment of the coordinate value is 0.01 or 0.01 degree (e.g., specify 20.01 for 20.01 mm).

(2) Alarm occurs if the specified coordinates exceed the robot's operational space.

(3) The default structure flags are R (Right), A (Above), N (Non Flip).

(4) The default coordinate value is zero.

(5) If you set any condition of speed, timer, or input/output in the teaching playback method, the MPB command is generated at each step .

(6) The input and output setting must be hexadecimal headed with "&".

(7) If the input/output signal is not set, the bit corresponding on the teaching box's screen display will be "-".

(8) The order that the condition data specified with this command is executed is as follows:

1)Interpolation method 2)Speed 3)Movement to target position 4)Timer 5)Signal output (including hand control) Signal input wait

RV-E-Robot with 6 axes(RV-E2, RV-E4N, etc.)

Function:

Moves to the specified position with specified interpolation, specified speed, specified timer, and specified input and output signal.

Input Format

MPB [<speed>], [<timer>], [<output ON>], [<output [<input ON>], [<input OFF>] [, [<interpolation>], [<X coordinate>], [<Y coordinate>], [<Z coordinate>], [<A turning angle>], [<B turning angle>], [<C turning angle>] [<R/L>] [, [<A/B>] [, [<N/F>]]]] [, [<O/C>]]

Term

<Speed>

Specify the interpolation speed to the destination position.
0 <= speed <= 32767(Joint interpolation: %, Linear interpolation: mm/s)

<Timer>

Set timer at the destination position after the movement.
0 <= timer <= 255

<Output ON>

Set the output signal that turns ON.
0 <= output ON (Hexadecimal) <= & FFFF
1: Setting, 0: Not setting

<Output OFF>

Set the output signal that turns OFF.
0 <= output OFF (Hexadecimal) <= & FFFF
1: Setting,0: Not setting

<Input ON>

Set the input waiting signal that turns ON.
0 <= input ON (Hexadecimal) <= & FFFF
1: Setting, 0: Not setting

<Input OFF>

Set the input waiting signal that turns OFF.
0 <= input OFF (Hexadecimal) <= & FFFF
1: Setting,0: Not setting

<Interpolation>

Specify the interpolation mode to the destination position.
Joint interpolation: 0 (Default)
Linear interpolation: 1
Circular interpolation: 2

<X, Y, Z coordinate>

Specify the location (mm) in XYZ coordinates the robot.
(Zero for default)

<A, B,C turning angle >

Specify the turning angle around X (A), Y (B), (C) axes in XYZ coordinates (degree) of the robot. (0 for default)

<R/L>

Specify the structure flag of the robot. (Right Left)

R: Right (Default)
L: Left

<A/B>

Specify the structure flag of the robot. (Above or Below)
A: Above (Default)
B: Below

<N/F>

Specify the structure flag of the robot. (Non or Flip)
N: Non flip (Default)
F: Flip

<O/C>

Specify open or close state of hand 1.
O: Hand 1 open (Default)
C: Hand 1 close

Explanation

(1) The least increment of the coordinate value is 0.01 or 0.01 degree (e.g., specify 20.01 for 20.01 mm).

(2) Alarm occurs if the specified coordinates exceed the robot's operational space.

(3) The default structure flags are R (Right), A (Above), (Non Flip).

(4) The default coordinate value is zero.

(5) If you set any condition of speed, timer, or input/output in the teaching playback method, the MPB command is generated at each step .

(6) The input and output setting must be hexadecimal headed with "&".

(7) If the input/output signal is not set, the bit corresponding on the teaching box's screen display will "-".

(8) The order that the condition data specified with this command is executed is as follows:

1)Interpolation method 2)Speed 3)Movement to target position 4)Timer 5)Signal output (including hand control)

Signal input wait

RV-E-Robot with 5 axes (RV-E3J, RV-E5NM, etc.) and with additional axis

Function:

Moves to the specified position with a specified interpolation method, speed, timer, and input/output signal.

Input Format

MPB [<speed>], [<timer>], [<output ON>], [<output [<input ON>], [<input OFF>] [, [<interpolation>], [<X coordinate>], [<Y coordinate>], [<Z coordinate>], [<A turning angle>], [<B turning angle>] [, <First additional axis> [, <Second additional axis>]] [, [<R/L>] [, [<A/B>]]] [<O/C>]]

Term

<Speed>

Specify the interpolation speed to the destination position.
0 <= speed <= 32767(Joint interpolation: %, Linear interpolation: mm/s)

<Timer>

Set timer at the destination position after the movement.
0 <= timer <= 255

<Output ON>

Set the output signal that turns ON.
0 <= output ON (Hexadecimal) <= & FFFF
1: Setting, 0: Not setting

<Output OFF>

Set the output signal that turns OFF.
0 <= output OFF (Hexadecimal) <= & FFFF
1: Setting, 0: Not setting

<Input ON>

Set the input waiting signal that turns ON.
0 <= input ON (Hexadecimal) <= & FFFF
1: Setting, 0: Not setting

<Input OFF>

Set the input waiting signal that turns OFF.
0 <= input OFF (Hexadecimal) <= & FFFF
1: Setting, 0: Not setting

<Interpolation>

Specify the interpolation mode to the destination position.
Joint interpolation: 0 (Default)
Linear interpolation: 1
Circular interpolation: 2

<X, Y, Z coordinate>

Specify the location (mm) in XYZ coordinates the robot.
(Zero for default)

<A, B turning angle >

Specify the turning angle around roll(A), pitch (B) axes in XYZ coordinates (degree) of the robot. (0 for default)

<First additional axis>

Specifies the moving position for First additional axis (If omitted, value is the position)
The unit is determined by the parameter, AXUN.

<Second additional axis>

Specifies the moving position for Second additional axis (If omitted, value is the position)
The unit is determined by the parameter, AXUN.

<R/L>

Specify the structure flag of the robot. (Right Left)
R: Right (Default)
L: Left

<A/B>

Specify the structure flag of the robot. (Above

or Below)
A: Above (Default)
B: Below

<O/C>

Specify open or close state of hand 1.
O: Hand 1 open (Default)
C: Hand 1 close

Explanation

(1) The additional axis will also move if the position includes additional axis data.

(2) If the additional axis is omitted, only the robot will move.

(3) The coordinate values of nonexistent additional axes will be ignored.

(4) The unit for the additional axis is specified in the parameter, AXUN.

(5) The linear interpolation is specified for the <interpolation>, the speed of the additional axis will be converted from the speed of the joint interpolation (%) (the control of the robot will continue to function as usual). The relationship between linear interpolation and joint interpolation is shown below:

Linear interpolation speed	**Joint interpolation speed [%]**	**Linear interpolation speed**	**Joint interpolation speed [%]**
1-2	0.4	106-123	19.0
3	0.6	124-144	22.2
4-5	0.8	145-168	25.9
6-7	1.1	169-193	29.8
8-9	1.5	194-222	34.2
10-13	2.0	223-264	40.7
14-17	2.7	265-307	47.3
18-23	3.7	308-350	53.9
24-31	4.9	351-393	60.5
32-42	6.5	394-436	67.1
43-53	8.2	437-478	73.7
54-63	9.7	479-521	80.2
64-75	11.6	522-564	86.8
76-89	13.7	565-607	93.4

90-105	16.2	608-32767	100.0

Related parameters:

AXUN

RV-E-Robot with 6 axes(RV-E2, RV-E4N, etc.) and with additional axis

Function:

Moves to the specified position with a specified interpolation method, speed, timer, and input/output signal.

Input Format

MPB [<speed>], [<timer>], [<output ON>], [<output [<input ON>], [<input OFF>] [, [<interpolation>], [<X coordinate>], [<Y coordinate>], [<Z coordinate>], [<A turning angle>], [<B turning angle>], [<C turning angle>] <First additional axis> [, <Second additional axis>]] [, [<R/L>] [, [<A/B>] [, [<N/F>]]]] [, [<O/C>]]

Term

<Speed>

Specify the interpolation speed to the destination position.
0 <= speed <= 32767(Joint interpolation: %, Linear interpolation: mm/s)

<Timer>

Set timer at the destination position after the movement.
0 <= timer <= 255

<Output ON>

Set the output signal that turns ON.
0 <= output ON (Hexadecimal) <= & FFFF
1: Setting, 0: Not setting

<Output OFF>

Set the output signal that turns OFF.
0 <= output OFF (Hexadecimal) <= & FFFF
1: Setting, 0: Not setting

<Input ON>

Set the input waiting signal that turns ON.
0 <= input ON (Hexadecimal) <= & FFFF
1: Setting, 0: Not setting

<Input OFF>

Set the input waiting signal that turns OFF.
0 <= input OFF (Hexadecimal) <= & FFFF
1: Setting, 0: Not setting

<Interpolation>

Specify the interpolation mode to the destination position.
Joint interpolation: 0 (Default)
Linear interpolation: 1
Circular interpolation: 2

<X, Y, Z coordinate>

Specify the location (mm) in XYZ coordinates the robot.
(Zero for default)

<A, B,C turning angle >

Specify the turning angle around X (A), Y (B), (C) axes in XYZ coordinates (degree) of the robot. (0 for default)

<First additional axis>

Specifies the moving position for First additional axis (If omitted, value is the position)
The unit is determined by the parameter, AXUN.

<Second additional axis>

Specifies the moving position for Second additional axis (If omitted, value is the position)
The unit is determined by the parameter, AXUN.

<R/L>

Specify the structure flag of the robot. (Right Left)

R: Right (Default)
L: Left

<A/B>

Specify the structure flag of the robot. (Above or Below)
A: Above (Default)
B: Below

<N/F>

Specify the structure flag of the robot. (Non or Flip)
N: Non flip (Default)
F: Flip

<O/C>

Specify open or close state of hand 1.
O: Hand 1 open (Default)
C: Hand 1 close

Explanation

(1) The additional axis will also move if the position includes additional axis data.

(2) If the additional axis is omitted, only the robot will move.

(3) The coordinate values of nonexistent additional axes will be ignored.

(4) The unit for the additional axis is specified in the parameter, AXUN.

(5) The linear interpolation is specified for the <interpolation>, the speed of the additional axis will be converted from the speed of the joint interpolation (%) (the control of the robot will continue to function as usual). The relationship between linear interpolation and joint interpolation is shown below:

Linear interpolation speed	Joint interpolation speed [%]	Linear interpolation speed	Joint interpolation speed [%]
1-2	0.4	106-123	19.0
3	0.6	124-144	22.2
4-5	0.8	145-168	25.9
6-7	1.1	169-193	29.8

8-9	1.5	194-222	34.2
10-13	2.0	223-264	40.7
14-17	2.7	265-307	47.3
18-23	3.7	308-350	53.9
24-31	4.9	351-393	60.5
32-42	6.5	394-436	67.1
43-53	8.2	437-478	73.7
54-63	9.7	479-521	80.2
64-75	11.6	522-564	86.8
76-89	13.7	565-607	93.4
90-105	16.2	608-32767	100.0

Related parameters:

AXUN

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

MPC (Move Playback Continuous)

RV-1A, RV-2AJ

Function: No operation.

No operation. This command isn't available for these robottypes.

RV-E-Robot with 5 axes (RV-E3J, RV-E5NM, etc.)

Function:

Moves to the specified position with specified interpolation.

Input Format

MPC [<interpolation>], [<X coordinate value>], [<Y coordinate value>], [<Z coordinate value>], [<A turning angle>], [<B turning angle>] [, [<R/L>] [, [<A/B>]]] [, [<O/C>]]

Term

<Interpolation>

Specify the interpolation mode to the destination position.
0: Joint interpolation (Default)
1: Linear interpolation
2: Circular interpolation

<X, Y, Z coordinate>

Specify the location (mm) in XYZ of the robot. (0 for default)

<A, B turning angle >

Specify the turning angle around roll(A), (B) axes in XYZ coordinates (degree) of the robot. (0 for default)

<R/L>

Specify the structure flag of the robot.
or Left)
R: Right (Default) L: Left

<A/B>

Specify the structure flag of the robot.
or Below)
A: Above (Default) B: Below

<O/C>

Specify open or close state of the hand 1.
O: Hand open (Default) C: Hand close

Explanation

(1) The least increment of the coordinate value is 0.01 mm or 0.01 degree (e.g. specify 20.01 for 20.01 mm).

(2) Alarm occurs if the specified coordinates exceed robot's operational space.

(3) The default coordinate value is zero.

(4) Use the MPB command If the setting of speed, timer, or input/output are needed.

(5) If you do not set any condition of speed, timer, or input/output in the teaching playback method, the command is generated at each step.

(6) The order that the condition data specified with command is executed is as follows.

1)Interpolation method 2)Speed 3)Movement to target position 4)Timer 5)Signal output (including hand 6)Signal input wait

RV-E-Robot with 6 axes(RV-E2, RV-E4N, etc.)

Function:

Moves to the specified position with specified interpolation.

Input Format

MPC [<interpolation>], [<X coordinate value>], [<Y coordinate value>], [<Z coordinate value>], [<A turning angle>], [<B turning angle>], [<C turning angle>], [, [<R/L>] [, [<A/B>] [, [<N/F>]]]] [, [<O/C>]]

Term

<Interpolation>

Specify the interpolation mode to the destination position.
0: Joint interpolation (Default)
1: Linear interpolation
2: Circular interpolation

<X, Y, Z coordinate>

Specify the location (mm) in XYZ of the robot. (0 for default)

<A, B,C turning angle >

Specify the turning angle around X (A), Y Z (C) axes in XYZ coordinates (degree) of the robot. (0 for default)

<R/L>

Specify the structure flag of the robot. (Right or Left)
R: Right (Default) L: Left

<A/B>

Specify the structure flag of the robot. or Below)
A: Above (Default) B: Below

<N/F>

Specify the structure flag of the robot. (Nonflip or Flip)
N: Non flip (Default) F: Flip

<O/C>

Specify open or close state of the hand 1.
O: Hand open (Default) C: Hand close

Explanation

(1) The least increment of the coordinate value is 0.01 mm or 0.01 degree (e.g. specify 20.01 for 20.01 mm).

(2) Alarm occurs if the specified coordinates exceed robot's operational space.

(3) The default coordinate value is zero.

(4) Use the MPB command If the setting of speed, timer, or input/output are needed.

(5) If you do not set any condition of speed, timer, or input/output in the teaching playback method, the command is generated at each step.

(6) The order that the condition data specified with command is executed is as follows.

1)Interpolation method 2)Speed 3)Movement to target position 4)Timer 5)Signal output (including hand 6)Signal input wait

RV-E-Robot with 5 axes (RV-E3J, RV-E5NM, etc.) and with additional axis

Function:

Moves to the specified position with specified interpolation.

Input Format

MPC [<interpolation>], [<X coordinate value>], [<Y coordinate value>], [<Z coordinate value>], [<A turning angle>], [<B turning angle>] [, <First additional axis> [, <Second additional axis>]] [, [<R/L>] [, [<A/B>]]] [, [<O/C>]]

Term

<Interpolation>

Specify the interpolation mode to the destination position.
0: Joint interpolation (Default)
1: Linear interpolation
2: Circular interpolation

<X, Y, Z coordinate>

Specify the location (mm) in XYZ coordinates of the robot. (0 for default)

<A, B turning angle >

Specify the turning angle around roll(A), pitch(B) axes XYZ coordinates (degree) of the robot. (0 for default)

<First additional axis>

Specifies the moving position for First additional axis omitted, value is the current position)
The unit is determined by the parameter, AXUN.

<Second additional axis>

Specifies the moving position for Second additional axis (If omitted, value is the current position)
The unit is determined by the additional axes AXUN.

<R/L>

Specify the structure flag of the robot. (Right or Left)
R: Right (Default) L: Left

<A/B>

Specify the structure flag of the robot. (Above or
A: Above (Default) B: Below

<O/C>

Specify open or close state of the hand 1.
O: Hand open (Default) C: Hand close

Explanation

(1) The additional axis will also move if the position includes additional axis data.

(2) If the additional axis is omitted, only the robot will move.

(3) The coordinate values of nonexistent additional will be ignored.

(4) The unit for the additional axis is specified in the additional axes parameter AXUN.

(5) The linear interpolation is specified for the <interpolation>, the speed of the additional axis will be converted from the speed of the joint interpolation (%) (the control of the robot will continue to function as usual). The relationship between linear interpolation joint interpolation is shown in the explanation for the MPB command.

Related parameters:

additional axes parameter AXUN

RV-E-Robot with 6 axes(RV-E2, RV-E4N, etc.) and with additional axis

Function:

Moves to the specified position with specified interpolation.

Input Format

MPC [<interpolation>], [<X coordinate value>], [<Y coordinate value>], [<Z coordinate value>], [<A turning angle>], [<B turning angle>], [<C turning angle>] [, additional axis> [, <Second additional axis>]] [, [<R/L>] [<A/B>] [, [<N/F>]]]] [, [<O/C>]]

Term

<Interpolation>

Specify the interpolation mode to the destination position.
0: Joint interpolation (Default)

1: Linear interpolation
2: Circular interpolation

<X, Y, Z coordinate>

Specify the location (mm) in XYZ
of the robot. (0 for default)

<A, B,C turning angle >

Specify the turning angle around X (A), Y
Z (C) axes in XYZ coordinates (degree) of the
robot. (0 for default)

<First additional axis>

Specifies the moving position for First
additional axis (If omitted, value is the
current position)
The unit is determined by the additional
parameter AXUN.

<Second additional axis>

Specifies the moving position for Second
additional axis (If omitted, value is the
current position)
The unit is determined by the additional
parameter AXUN.

<R/L>

Specify the structure flag of the robot.
or Left)
R: Right (Default) L: Left

<A/B>

Specify the structure flag of the robot.
or Below)
A: Above (Default) B: Below

<N/F>

Specify the structure flag of the robot.
(Nonflip or Flip)
N: Non flip (Default) F: Flip

<O/C>

Specify open or close state of the hand 1.
O: Hand open (Default) C: Hand close

Explanation

(1) The additional axis will also move if the position includes additional axis data.

(2) If the additional axis is omitted, only the robot will move.

(3) The coordinate values of nonexistent additional will be ignored.

(4) The unit for the additional axis is specified in the additional axes parameter AXUN

(5) The linear interpolation is specified for the <interpolation>, the speed of the additional axis will be converted from the speed of the joint interpolation (%) (the control of the robot will continue to function as usual). The relationship between linear interpolation joint interpolation is shown in the explanation for the MPB command.

Related parameters:

additional axes parameter AXUN

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

MR (Move R)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Moves the tip of hand through the predefined intermediate positions in circular interpolation.

Input Format

MR <position number (a) > , <position number (b)>, <position number (c)> [, [< O/C >]]

Term

<Position number>

Specify the positions on the circle.
1 <= position number <= 999

<O/C>

Specify open or close state of the hand. (If omitted, the hand state of the position is valid.)
O: Hand open
C: Hand close

Explanation

(1) Moves the tip of hand through specified positions (a) via (b) to (c) drawing an arc.

(2) The moving speed of circular interpolation is by the SP or SD command. (The tip of hand at constant speed.) Since the locus accuracy depends on the speed circular interpolation, set the moving speed lower if need high accuracy.

(3) The open or close state of the hand does not before and after the movement.

(4) If the starting position (a) is different from the current position, the robot moves to the starting by linear interpolation.

(5) If the circular interpolation is interrupted by the signal and restarted by the start signal, the robot the remaining arc. If the tip of hand is kept away from the stopping position by JOG operation in the above the robot moves to the stopping position by joint interpolation then moves the remaining arc.

(6) Alarm takes place if the specified position has not been predefined or exceeds the robot's operational space. The robot moves by linear interpolation if three positions (a), (b) and (c) are located on a straight line

if two of three positions are the same.

(7) If the moving direction of each joint changes greatly at the beginning of circular interpolation, alarm may occur. Set speed lower or set timer at the beginning in this case.

(8) The drawing direction and the locus of the arc on the order of the specified positions.

Sample program (MOVEMASTER Command)

10 SP 8 'Set speed to 8.

20 MO 1 'Moves to position 1.

30 MR 10,20,30 'Moves to position 10 by linear interpolation.

'Moves the arc determined by 10, 20, 30

'by circular interpolation.

40 MS 3 'Moves to position 3 by linear interpolation.

50 ED 'Ends program.

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Using circular interpolation, moves the tip of the hand through previously specified intermediate positions.

Input Format

MR <position number (a) > , <position number (b)>, <position number (c)> [, [< O/C >]]

Term

<Position number>

Specify the positions on the circle.
1 <= position number <= 999

<O/C>

Specify open or close state of the hand. (If omitted, the hand state of the position is

valid.)
O: Hand open
C: Hand close

Explanation

(1) Moves the tip of the hand through specified to make a circular arc beginning with (a) the start going through (b) the transit point, and finishing at (c) the end point.

(2) The additional axis also moves to the start point if end point has movement components for the additional axis.

(3) The additional axis will not move to the transit from the start point if there are movement components for the additional axis at the transit point. (It will move directly to the end point.)

(4) When the robot moves in response to the additional axis movements (travel axis, etc.), the tip of the locus the hand tip will not from a circular arc.

Sample program (MOVEMASTER Command)

10 SP 8 'Set speed to 8.

20 MO 1 'Moves to position 1.

30 MR 10,20,30 'Moves to position 10 by linear interpolation.

'Moves the arc determined by position 10, 20, 30

'by circular interpolation.

40 MS 3 'Moves to position 3 by linear interpolation.

50 ED 'Ends program.

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

MRA (Move R A)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Moves to the specified position in circular

Input Format

MRA <Position number> [, [<O/C>]]

Term

<Position number>

Specify the destination position.
1 <= position number <= 999

<O/C>

Specify open or close state of the hand. (If omitted, the hand state of the position is valid.)
O: Hand open
C: Hand close

Explanation

(1) Moves the tip of hand on the arc which is defined by the former and the latter positions of the MRA commands. The tip of hand is decided by the tool currently established. (see TL command).

(2) If the open or close state of the hand has been specified, the robot moves after executing the hand control.

(3) Alarm takes place if the specified position has not been predefined.

(4) If the MRA command does not continue more than three, it becomes similar to the MC command. The following command, however, can be executed the MRA command.

SD, SP, TI, OVR, OB, OC, OD, GC and GO

(5) If the execution of the MRA command is interrupted and the tip of hand is kept away from the stopping position by JOG operation, the robot moves, when restarted, to the stopping position by linear then moves the remaining arc.

Sample program (MOVEMASTER Command)

10 MRA 1,O 'Defines the arc with positions 1, 2,

'Moves to position 1 by linear interpolation.

20 MRA 2,O 'Moves to position 2 by circular interpolation.

30 MRA 3,C 'Moves to position 3 by circular interpolation.

40 TI 3 'Timer 0.3 second.

50 MRA 4 'Moves to position 4 by circular interpolation.

60 MRA 5 'Moves to position 5 by circular interpolation.

70 ED 'Ends program.

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Moves to the specified position in circular

Input Format

MRA <Position number> [, [<O/C>]]

Term

<Position number>

Specify the destination position.
1 <= position number <= 999

<O/C>

Specify open or close state of the hand. (If omitted, the hand state of the position is

valid.)
O: Hand open
C: Hand close

Explanation

(1) If there are less than 3 consecutive MRA commands, they will be executed identically to MC commands (movement by linear interpolation). However, the following commands can be described between MRA commands: (They will be by circular interpolation) SD, SP, TI, OVR, OC, OD, GC and GO, ADL

(2) The additional axis also moves if movement components exist for the additional axis.

(3) When the robot moves in response to the additional axis movements (travel axis, etc.), the tip of the locus the hand tip will not form a circular arc.

Sample program (MOVEMASTER Command)

10 MRA 1,O 'Generation of a circle by combining positions 1,2, and 3.

'Move to position 1 by linear interpolation.

20 MRA 2,O 'Move to position 2 by circular interpolation.

30 MRA 3,C 'Move to position 3 by circular interpolation.

40 TI 3 'Timer 0.3 second.

50 MRA 4 'Move to position 4 by circular interpolation.

60 MRA 5 'Move to position 5 by circular interpolation.

70 ED 'End program.

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

MS (Move Straight)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.), RV-M2

Function:

Moves the tip of hand to the specified position. (Linear interpolation)

Input Format

MS <Position number> [, [<O/C>]]

Term

<Position number>

Specify the destination position number in integer value.
1 <= position number <= 999

<O/C>

Specify open or close state of hand. (If omitted, the hand state of the position is valid.)
O: Hand open
C: Hand close

Explanation

(1) Moves the tip of hand to the specified position by linear interpolation. The tip of the hand is decided by tool length currently established. (See the TL

(2) Alarm occurs before or during movement if the destination or movement path goes beyond the robot's operational space.

(3) If the open or close state of the hand has been specified, the robot moves after executing the hand control.

(4) The moving speed is decided by the SP or SD

commands. (The tip of hand at constant speed.)

(5) Use the MC command to move continuously through several positions by linear interpolation.

Sample program (MOVEMASTER Command)

```
10 SP 15     'Sets speed to 15.
20 MO 1      'Moves to position 1 by joint interpolation.
30 MS 5      'Moves to position 5 by linear
40 MS 6      'Moves to position 6 by linear
50 MS 7      'Moves to position 7 by linear
60 MS 8      'Moves to position 8 by linear
70 MS 5      'Moves to position 5 by linear
```

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Moves the tip of hand to the specified position. (Linear interpolation)

Input Format

MS <Position number> [, [<O/C>]]

Term

<Position number>

Specify the destination position number in integer value.
1 <= position number <= 999

<O/C>

Specify open or close state of hand. (If omitted, the hand state of the position is valid.)
O: Hand open
C: Hand close

Explanation

(1) The additional axis also moves if movement components exist for the additional axis.

(2) When the robot moves in response to the additional axis movements (travel axis, etc.), the tip of the locus the hand tip will not be straight.

Sample program (MOVEMASTER Command)

```
10 SP 15     'Sets speed to 15.
20 MO 1      'Moves to position 1 by joint
30 MS 5      'Moves to position 5 by linear
40 MS 6      'Moves to position 6 by linear
50 MS 7      'Moves to position 7 by linear
60 MS 8      'Moves to position 8 by linear
70 MS 9      'Moves to position 9 by linear
```

RV-M1

Function:

Moves the robot to a specified position number through the specified number of intermediate points on a line. (Linear interpolation)

Input Format

MS <Position number>, <No. of intermediate points> [, [<O/C>]]

Term

<Position number>

Specify the destination position number in integer value.
1 <= position number <= 629

< No. of intermediate points >

Specify the number of intermediate points
1 <= No. of intermediate points <= 99

<O/C>

Specify open or close state of hand. (If omitted, the hand state of the position is valid.)
O: Hand open

C: Hand close

Explanation

(1) The number of intermediate points between the current position and the specified position number is calculated by equally dividing the distance of travel position angle (pitch/roll angle) between the two positions (the number of divisions equaling the number intermediate points plus 1). Acceleration and deceleration during the movement are omitted.

(2) The more intermediate points, the smoother the straight line for the movement path, but the more time required for calculating before the robot starts moving (99 intermediate points require about 2.4 seconds). It therefore recommended that the number of points is specified according to the distance of travel the required path accuracy.

(3) If any of the intermediate points specified excess robots operational space, the robot stops in mid- and error mode II is caused. The hand end velocity movement changes in accordance with the attitude.

(4) If the open or close state of the hand has been specified, the robot moves after executing the hand control.

(5) Moves the tip of hand to the specified position by linear interpolation. The tip of the hand is decided by tool length currently established. (See the TL

(6) Error mode II occurs before or during movement if specified position has not been predefined.

(7) The moving speed is decided by the SP or SD commands. (The tip of hand at constant speed.)

(8) Use the MC command to move continuously through several positions by linear interpolation.

(9) Certain positions of the robot may cause In such cases, keep the speed low. (The speed should set to SP5 or lower. 50 mm/second max.)

Sample program (MOVEMASTER Command)

10 SP 15 'Sets speed to 15.

20 MO 1 'Moves to position 1 by joint interpolation.

30 MS 5, 3, C 'Moves to position 5 by linear interpolation with 3 intermediate points

80 ED 'End of program

COSIMIR® · Movemaster Command

MT (Move Tool)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Moves the tip of hand to a position away from the specified position by the distance as specified in the direction. (Joint interpolation)

Input Format

MT <position number>, [<travel distance>] [, [<O/C>]]

Term

<Position number>

Specify the destination position number in integer value.
1 <= position number <= 999

<Travel distance>

Specify the distance in tool direction from specified position to the destination point. (Zero for default)
-3276.80 <= travel distance <= 3276.70

<O/C>

Specify open or close state of the hand. (If omitted, the hand state of the position is valid.)
O: Hand open
C: Hand close

Explanation

(1) The least increment of the distance is 0.01 mm.

(2) When the distance is positive, the tip of hand advances in the tool direction. When the distance is negative, the tip of hand retracts in the tool direction.

(3) If the open or close state of the hand has been specified, the robot moves after executing the hand control.

(4) Alarm occurs when the MT command is executed if the specified position has not been predefined or if the destination exceeds the robot's operational space.

Sample program (MOVEMASTER Command)

10 MT 1,-100;Moves to the point away from the position by 100 mm.

20 MS 1;Moves to position 1.

30 MT 1,-100;Moves to the point away from the position by 100 mm.

RV-M2

Function:

Moves the tip of hand to a position away from the specified position by the distance as specified in the direction. (Joint interpolation)

Input Format

MT <position number>, [<travel distance>] [, [<O/C>]]

Term

<Position number>

Specify the destination position number in integer value.
1 <= position number <= 999

<Travel distance>

Specify the distance in tool direction from specified position to the destination point.
(Zero for default)
-3276.80 <= travel distance <= 3276.70

<O/C>

Specify open or close state of the hand. (If omitted, the hand state of the position is valid.)
O: Hand open
C: Hand close

Explanation

(1) The least increment of the distance is 0.1 mm.

(2) When the distance is positive, the tip of hand advances in the tool direction. When the distance is negative, the tip of hand retracts in the tool direction.

(3) If the open or close state of the hand has been specified, the robot moves after executing the hand control.

(4) Alarm occurs when the MT command is executed if the specified position has not been predefined or if the destination exceeds the robot's operational space.

Sample program (MOVEMASTER Command)

10 MT 1,-100 'Moves to the point away from the position 1 by 100 mm.

20 MS 1 'Moves to position 1.

30 MT 1,-100 'Moves to the point away from the position 1 by 100 mm.

RV-M1

Function:

Moves the tip of hand to a position away from the specified position by the distance as specified in the direction. (Joint interpolation)

Input Format

MT <position number>, [<travel distance>] [, [<O/C>]]

Term

<Position number>

Specify the destination position number in integer value.
1 <= position number <= 629

<Travel distance>

Specify the distance in tool direction from specified position to the destination point. (Zero for default)
-3276.80 <= travel distance <= 3276.70

<O/C>

Specify open or close state of the hand. (If omitted, the hand state of the position is valid.)
O: Hand open
C: Hand close

Explanation

(1) The least increment of the distance is 0.1 mm.

(2) When the distance is positive, the tip of hand advances in the tool direction. When the distance is negative, the tip of hand retracts in the tool direction.

(3) If the open or close state of the hand has been specified, the robot moves after executing the hand control.

(4) Alarm occurs when the MT command is executed if the specified position has not been predefined or if the destination exceeds the robot's operational space.

Sample program (MOVEMASTER Command)

10 MT 1,-100 'Moves to the point away from the position 1 by 100 mm.

20 MS 1 'Moves to position 1.

30 MT 1,-100 'Moves to the point away from the position 1 by 100 mm.

COSIMIR® · Movemaster Command

MTS (Move Tool Straight)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Moves the tip of hand to a position away from the specified position by the distance as specified in the direction. (Linear interpolation)

Input Format

MTS < position number>, [<travel distance>] [, [<O/C>]]

Term

<Position number>

Specify the destination position number in integer value.
1 <= position number <= 999

<Travel distance>

Specify the distance in tool direction from specified position to the destination point. (Zero for default)
-3276.80 <= travel distance <= 3276.70

<O/C>

Specify open or close state of the hand. (If omitted, the hand state of the position is valid.)
O: Hand open
C: Hand close

Explanation

(1) The least increment of the distance is 0.01 mm.

(2) When the distance is positive, the tip of hand advances in the tool direction. When the distance is negative, the tip of hand retracts in the tool direction.

(3) If the open or close state of the hand has been specified, the robot moves after executing the hand control.

(4) Alarm occurs when the MT command is executed if the specified position has not been predefined or if the destination exceeds the robot's operational space.

Sample program (MOVEMASTER Command)

10 MTS 1,-100,O 'Moves to the point away from the position 1

'by 100 mm with hand opened.
Moves straight.

20 MS 1 'Moves to position 1.

30 MTS 1,-100,C 'Moves to the point away from the position 1

'by 100 mm with hand closed.
Moves straight.

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

MUL (Multiply)

RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

Multiplies the internal register value and operation and stores the result in the internal register.

Input Format

MUL <Operation data>

Term

<Operation data>

Describes the data to be operated as a numeric value or counter No. with @.
-32768 <= numeric value (decimal) <= 32767
&8000 <= numeric value (hexadecimal) <=

&7FFF
@1 <= Counter No. <= @99

Explanation

(1) Designate the operation data setting as a numeric value or counter No.

When designating with a numeric value, use a decimal hexadecimal value. When using a hexadecimal, add a to the head of the operation data.

When setting with a counter No. add a "@" to the head the counter No.

The contents of the set counter No. will be used as the operation data.

(2) The operation results are stored in the internal register, so operation, comparison and reading, etc., of the operation results can be carried out with the commands.

(Refer to SUB, ADD, DIV, EQ, NE, LG, SM, CL and DR, OR commands)

Sample program (MOVEMASTER Command)

10 CP 1 'Stores counter No. 1 value in internal register

20 MUL @2 'Multiplies internal register value by counter No. 2 value

30 CL 3 'Sets internal register value in counter 3

'(Counter No. 3 = counter No. 1 x No. 2)

40 CP 1 'Stores counter No. 1 value in internal register

50 MUL 5 'Multiplies internal register value by 5

60 CL 4 'Sets internal register value in counter 4

'(Counter No. 4 = counter No. 1 x 5)

RV-E-Robot (RV-E2, RV-E3J), RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-E, RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

N * (Number)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Select the specified program.

Input Format

N <program name>

Term

<Program name>

Specify the robot program name. (Less than characters)
Possible letter used:Digits (0-9)
Characters (A - Z)
Symbols (! @ # $ % ^ & ( ) _ | { } -)
Impossible letter used:* + , . / : ; = ? [\ ] '
Special specification: When you specified numeric value, is handled for number.
Need to enclose program name with " " in case of character used.

Explanation

(1) Select the specified program. The program selected here becomes an object of the implementation, modification and operation. The program selected once remains till other program number is selected afresh. (Even if the power turns OFF, the program number remains unchanged.)

(2) You can confirm the current program number using the QN command from the personal computer.

(3) Program 1 is selected at product line shipping.

(4) The following name is identified as the same.

An example:

Handled as the same.1,01,001,00000001 (Only numeric value)

Handled as not the same.1,1 A, A0 _ 001 (Includes characters)

(5)The letters that controller can indicate to the LED 0-9, A - Z

* This command can only be executed directly. It cannot be used in a program.

Sample program (BASIC)

10 OPEN "COM1 :E83"AS #1 'Opens the RS-232C communication file in BASIC.

20 PRINT #1, "N10" 'Selects the program

30 PRINT #1, "10 MO 1" 'Implementation of program. (Line 10)

40 PRINT #1, "20 MS 2" 'Implementation of program. (Line 20)

50 PRINT #1, "30 ED" 'Implementation of program. (Line 30)

60 END 'Ends

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

NE (If Not Equal)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

This compares the value of the internal register with a specified value. If not equal, the program will jump. character string register and details of a specified character string are compared. If not equal, the will jump.

Input Format

NE <compared value/character string number>, <branching line number>

Term

<Compared value>

Specify the value that the internal register compares contents with.
-32768 <= compared value (decimal) <=
& 8000 <= compared value (hexadecimal) <=
&7FFF

<Character string number>

Specify character string number in value which "$" is added to the head.
$1 <= character string number <= $99

<Branching line number>

Specify the line number to which the jumps when the comparison result is not equal.
1 <= branching line number <= 9999

Explanation

<When compared value is specified>

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value does not equal to the compared value (i.e. when the condition is met), the program jumps to the specified line. Otherwise (i.e. when the condition is not met), the program continues

sequence.

(3) A value can be loaded into the internal register by executing the input command (See ID) for the external input data or by executing the compare counter command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed "&".

<When character string number is specified>

(1) The conditions will jump depending on the data from an external source or the number of characters in specified character string.

(2) If the number of characters in the character string register is not equal to the details of a specified character string (when the conditions are established), the program will jump to the specified line number. If not (when conditions are not established), the next line will be executed.

(3) By executing an INP command, the data input from external device will be set in the character string register. The details of the character string number be set by executing a CP command. Thus, when executing condition jumping, one of these commands must be executed first.

(4) If the specified line number is not registered, an alarm will occur when jumping.

Sample program (MOVEMASTER Command)

10 ID 'Fetches data from external input port.

20 NE 80,100 'Jumps to line 100 if the data does equal 80.

30 ED 'Ends the program if above condition not met.

100 MO 7 'Moves to position 7.

110 OPN 1,1 'Opens the RS-232C port.

120 INP 1, ,2 'Reads the data of character string

register

'from the RS-232C port.

130 NE $2,200 'Jumps to line 200 if the data not equals

'character string number 2.

:

200 ED 'Ends program.

RV-E-Robot (RV-E2, RV-E3J)

Function:

This compares the value of the internal register with a specified value. If not equal, the program will jump.

Input Format

NE <compared value>, <branching line number>

Term

<Compared value>

Specify the value that the internal register compares contents with.
-32768 <= compared value (decimal) <=
& 8000 <= compared value (hexadecimal) <=
&7FFF

<Branching line number>

Specify the line number to which the jumps when the comparison result is not equal.
1 <= branching line number <= 9999

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value does not equal to the compared value (i.e. when the condition is met), the program jumps to the specified line. Otherwise (i.e. when the condition is not met), the program continues sequence.

(3) A value can be loaded into the internal register by executing the input command (See ID) for the external input data or by executing the compare counter command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed "&".

Sample program (MOVEMASTER Command)

10 ID 'Fetches data from external input port.

20 NE 80,100 'Jumps to line 100 if the data does not equal 80.

30 ED 'Ends the program if above condition is not met.

100 MO 7 'Moves to position 7.

RV-M2

Function:

This compares the value of the internal register with a specified value. If not equal, the program will jump.

Input Format

NE <compared value>, <branching line number>

Term

<Compared value>

Specify the value that the internal register compares contents with.
-32768 <= compared value (decimal) <=
& 8000 <= compared value (hexadecimal) <=
&7FFF

<Branching line number>

Specify the line number to which the jumps when the comparison result is not equal.
1 <= branching line number <= 3584

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value does not equal to the compared value (i.e. when the condition is met), the program jumps to the specified line. Otherwise (i.e. when the condition is not met), the program continues sequence.

(3) A value can be loaded into the internal register by executing the input command (See ID and IN) for the external input data or by executing the compare command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed "&".

Sample program (MOVEMASTER Command)

10 ID 'Fetches data from external input port.

20 NE 80,100 'Jumps to line 100 if the data does not equal 80.

30 ED 'Ends the program if above condition is not met.

100 MO 7 'Moves to position 7.

RV-M1

Function:

This compares the value of the internal register with a specified value. If not equal, the program will jump.

Input Format

NE <compared value>, <branching line number>

Term

<Compared value>

Specify the value that the internal register compares contents with.
-32767 <= compared value (decimal) <=
& 8001 <= compared value (hexadecimal) <=

&7FFF

<Branching line number>

Specify the line number to which the jumps when the comparison result is not equal.
1 <= branching line number <= 2048

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value does not equal to the compared value (i.e. when the condition is met), the program jumps to the specified line. Otherwise (i.e. when the condition is not met), the program continues sequence.

(3) A value can be loaded into the internal register by executing the input command (See ID and IN) for the external input data or by executing the compare command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed "&".

Sample program (MOVEMASTER Command)

10 ID 'Fetches data from external input port.

20 NE 80,100 'Jumps to line 100 if the data does not equal 80.

30 ED 'Ends the program if above condition is not met.

100 MO 7 'Moves to position 7.

COSIMIR® · Movemaster Command

NR (Nest Read)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function

No operation. Prepared to keep the compatibility with the MOVEMASTER M2 series. Only registration is

Input Format

NR [<Axis number>]

Explanation

This command is prepared for keeping the
with the MOVEMASTER M2 series and has no effect on program.

RV-M2

Function

Reads the number of limit pulses of any axis after setting (using RS232C)

Input Format

NR <Axis number>

Term

<Axis number>

Specifies the axis number
1 <= axis number <= 5

Explanation

(1) Outputs the number of limit pulses of the specified axis from the RS232C port after origin setting. The pulses are required to check the limit switch
state of each axis during maintenance and inspection.

(2) The output format is ASCII coded:

Output format:

Number of limit pulses for the specified axis 0D (hexadecimal)

(3) The limit pulse checking operation may also be

performed from the teaching box. See your manual for more information.

(4) the number of limit pulses indicated is the result of the most recent origin setting. Therefore, the number of limit pulses read after setting the origin again may be slightly different from the previous value.

(5) Error mode II occurs if the axis exceeds the specified range.

Sample program (BASIC)

```
10 OPEN "COM1 :9600, E, 7, 2"AS#1

20 INPUT "GELENKNUMMER =" ; J

30 PRINT #1"NR" + STR$(J)              'Read number of
limit pulses

40 LINE INPUT #1, A$

50 PRINT A$

60 END

RUN

250                                    'Number of limit
pulses
```

RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 series.

COSIMIR® · Movemaster Command

NT (Nest)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Carry out origin setting. (The robot moves to the user-defined origin.)

Input Format

NT

Explanation

(1) The moving sequence of each joint is fixed beforehand. Origin setting of the shoulder, the elbow the twist joint is first executed, which is followed by of the waist, the pitch, and the roll joint.

Note.: 5-axis robots don't have the twist joint.

(2) If the arm can interfere with the object surrounding the robot, use the teaching box to move it to a safe location before origin setting.

(3) You can change the moving sequence of the origin setting by the parameter UNG, and also can change the attitude by the parameter UOG (see the parameters for the RV-E2 and RV-E3J and the RV-EN series). The parameter UOG can be set with the teaching box operation, too. (See "HE" command.)

(4) The NT command is not required in the usual operation.

Relating Parameters

The sequential order of user origin setting can be changed by the following parameter.

Parameter UNG:Sequential order

2,1,1,1,2,2 (Default)

The attitude of user origin setting can be changed by following parameter.

Parameter UOG:User origin attitude (degree)

Default:

6-axis robots (e. g. RV-E4N/E4NM/E4NC): -160.00,-45.00,50.00,-160.00,-120.00,-200.00

5-axis robots (e. g. RV-E5NJ/E5NJM/E5NJC) : -160.00,-45.00,0.00,0.00,-120.00,-200.00

To find the change operation, refer to your manual.

Sample program (MOVEMASTER Command)

10 NT 'Executes origin setting.

20 MO 1 'Moves to position 1.

30 ED 'Ends program.

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Carry out origin setting. (The robot moves to the user-defined origin.)

Input Format

NT [< Process type>]

Terms:

< Process type >

Sets the processing type of the origin designation.
0: Robot moves to the user-defined origin (If omitted)
1: Origin designation for the First additional axis is executed.
2: Origin designation for the Second axis is executed.

Explanation

(1) This command will be unprocessed if the additional axis is using the absolute value method.

(2) If the additional axis is using the incremental every time the power is turned ON, and every time the emergency stop input is canceled, use this to designate the origin. (You can also use T/B,s origin designation or I/O ,s OGE command.)

Relating Parameters

Use the parameters and additional axes parameters below for determining the sequence of user-defined origin designations:

Parameter UNG:

Sequential order for origin designation

2,1,1,1,2,2 (Default value)

Use the parameter below for changing the position of user defined origin designations:

Parameter UOG:

Specified origin

6-axis robots (e. g. RV-E4N/E4NM/E4NC):

-160.00,-45.00,50.00,-160.00,-120.00,-200.00 (Default values)

5-axis robots (e. g. RV-E5NJ/E5NJM/E5NJC) :

-160.00,-45.00,0.00,0.00,-120.00,-200.00 (Default

Parameter AXOP:

Zero point position for the additional axis

0.00, 0.00 (Default value)

Relating Language

HE

Sample program (MOVEMASTER Command)

10 NT 'Execute the origin designation

20 NT 1 'Execute the origin designation for the first additional axis 1

30 MO 1 'Moves to position 1.

40 ED 'Ends program.

RV-M2

Function:

Carry out origin setting. (The robot moves to the user-defined origin.)

Input Format

NT

Explanation

(1) Used to return the robot to mechanical origin immediately after the power is switched on. Must be executed before teaching and before a program is executed.

(2) The functionality of the NT command depends on position of bit 4 of switch SW1 in the drive unit side Usually set bit 4 to the lower position. See your manual for further information.

(3) The moving sequence of each joint is fixed beforehand. Origin setting of the J2, J3 and J4 axes is first executed, which is followed by that of the J1 and joint. If the arm can interfere with the object surrounding the robot, use the teaching box to move it a safe location before origin setting.

(4) When the hand holds a workpiece, care must be to prevent personal injury because the hand opens as soon as the origin setting operation is initiated. The open/close condition depends on the setting of bit 3 of switch SW1 in the drive unit side door. (Set to the position for ordinary open/close operation and to the upper position for reverse open/close operation.)

(5) Ordinarily, origin setting may be executed only after the power is switched on. If, after origin setting, the robot has been stopped by emergency stop during re-execution, it returns to the power-on state the origin setting operation must therefore be again.

Sample program (MOVEMASTER Command)

```
10 NT        'Executes origin setting.

20 MO 1      'Moves to position 1.

30 ED        'Ends program.
```

RV-M1

Function:

Carry out origin setting. (The robot moves to the user-defined origin.)

Input Format

NT

Explanation

(1) this command causes the robot to return to the origin, which must be performed immediately after the power is turned ON. Execution of this command is necessary before any move command can be executed. Origin setting is performed automatically by the limit switches and phase-Z of encoders provided in each

(2) The functionality of the NT command depends on position of bit 4 of switch SW1 in the drive unit side Usually set bit 4 to the lower position. See your manual for further information.

(3) The moving sequence of each joint is fixed beforehand. Origin setting of the J2, J3 and J4 axes is first executed, which is followed by that of the J1 and joint. If the arm can interfere with the object surrounding the robot, use the teaching box to move it a safe location before origin setting.

(4) When the hand holds a workpiece, care must be to prevent personal injury because the hand opens as soon as the origin setting operation is initiated.

(5) Do not touch the limit switches and robot body until origin setting completes.

Sample program (MOVEMASTER Command)

```
10 NT      'Executes origin setting.

20 MO 1    'Moves to position 1.

30 ED      'Ends program.
```


COSIMIR® · Movemaster Command

NW * (New)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Deletes the specified program and position data.

Input Format

NW

Explanation

(1) Deletes all positions and counters of the specified program. Common positions (901-999) and common counters (91-99), however, are not deleted.

(2) Origin setting, internal register, tool length, speed setting, pallet setting, and hand setting remain unchanged even if the NW command is executed.

(3) The NW command can not be executed in the with line number. (Only direct execution is possible.)

* This command can only be executed directly. It be used in a program.

Sample program (BASIC)

10 OPEN "COM1 :E83"AS#1 'Opens the RS-232C communication file

'from the personal computer in BASIC.

30 PRINT #1, "NW" 'Transmit command

60 END 'Ends program.

RV-M2, RV-M1

Function:

Deletes all program and position data.

Input Format

NW

Explanation

(1) Deletes all programs and position data stored in the drive unit RAM.

(2) the command does not, however, delete the reference position data in the Cartesian coordinate system.

(3) The NW command can not be executed in the
with line number. (Only direct execution is possible.)

* This command can only be executed directly. It cannot be used in a program.

Sample program (BASIC)

10 LPRINT ,"NW"   ' Delete all programs and position

COSIMIR® · Movemaster Command

NX (Next)

Function:

Specifies the range of a loop in a program executed by the RC command.

Input Format

NX

Explanation

(1) Used in combination with the RC command to
the range of a loop in a program executed by the RC command.

(2) Alarm occurs if there is no corresponding "RC" command specified..

Sample program

See the RC command.

COSIMIR® · Movemaster Command

OB (Output Bit)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Sets the output state of the specified bit through an external output por

Input Format

OB [<+/->] <output bit number>

Term

<+/->

Set ON or OFF state of the specified bit.
+:Bit ON
-: Bit OFF

<Bit number>

Specify the bit number of external output.
0 <= bit number <= 32767

Explanation

(1) Set "+" to switch on the specified bit and "-" to switch off the specifie

(2) All bits other than the specified one are not affected by this commar output state of the specified bit is retained until a new setting is made t or OD command.

(3) If you specify the special bit in the parameters OT1-OT3 (see the for the RV-E2 and RV-E3J and the RV-EN series)of the external output, ala occurs because of the conflict of signals.

(4) For the pneumatic hand, you can specify open or close state of hand by the OB command. Refer to the table below. (Ordinary, use the GC or command.) For the motor-operated hand, these settings are not possible

Notice: GR1-GR4 shows connector number of hand output cable in the rc

Hand	Open/Close	GR1	GR2	GR3	GR4
		Output bit 900	Output bit 901	Output bit 902	Output 903
Hand	Open (GO 0)	ON	OFF		
	Close (GC 0)	OFF	ON		
Hand	Open (GO 0)			ON	OFF
	Close (GC 1)			OFF	ON

Sample program (MOVEMASTER Command)

10 OD &FFFF 'Turns the bits (0-15) of external output into ON entirel

20 OB -10 'Turns only bit 10 to OFF.

30 ED 'Ends program.

RV-M2

Function:

Sets the output state of the specified bit through an external output por

Input Format

OB <+/-> <output bit number>

Term

<+/->

Set ON or OFF state of the specified bit.
+:Bit ON
-: Bit OFF

<Bit number>

Specify the bit number of external output.
0 <= bit number <= 15

Explanation

(1) Set "+" to switch on the specified bit and "-" to switch off the specifie

(2) All bits other than the specified one are not affected by this commar output state of the specified bit is retained until a new setting is made t or OD command.

(3) If no bit number is specified, it defaults to bit 0.

Sample program (MOVEMASTER Command)

10 OD &FFFF 'Turns the bits (0-15) of external output into ON ent

20 OB -10 'Turns only bit 10 to OFF.

30 ED 'Ends program.

RV-M1

Function:

Sets the output state of the specified bit through an external output por

Input Format

OB <+/-> <output bit number>

Term

<+/->

Set ON or OFF state of the specified bit.
+:Bit ON
-: Bit OFF

<Bit number>

Specify the bit number of external output.
0 <= bit number <= 7 (15)
Figure in () is for type A16 or B16 I/O card.

Explanation

(1) Set "+" to switch on the specified bit and "-" to switch off the specifie

(2) All bits other than the specified one are not affected by this comman output state of the specified bit is retained until a new setting is made b OD or OT command.

(3) If no bit number is specified, it defaults to bit 0.

Sample program (MOVEMASTER Command)

10 OD &FF	'Turns the bits (0-8) of external output into ON entirely.
20 OB -4	'Turns only bit 4 to OFF.
30 ED	'Ends program.

COSIMIR® · Movemaster Command

OC (Output Counter)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Outputs the specified counter value unconditionally through the output port.

Input Format

OC <counter number> [, [<output bit>] [, [<bit width>]]]

Term

<Counter number>

Specify the counter number to be output.
1 <= counter number <= 99

<Output bit number>

Specify the reference bit number of output data
0 <= bit number <= 32767 (0 for default)

<Bit width>

Specify bit width of output data.
1 <= bit width <= 16 (16 for default)

Explanation

(1) Outputs the specified counter value unconditionally through the output port. The output data retains after that.

(2) Even if the OC command is executed, the value of specified counter and the internal register remain

(3) You can specify the range of output signal by the bit width of the OC command.

Sample program (BASIC)

```
10 OPEN "COM1 :E83"AS#1            'Opens the RS-232C communication file
                                   'from the personal computer in BASIC.
20 PRINT #1,"SC 5,&0008"           'Set value 8 to counter 5
30 PRINT #1,"OC5"                  'Outputs the value of the
                                   'unconditionally to output port.
```

40 END 'Ends program.

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

OD (Output Direct)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Outputs the specified data unconditionally through the output port.

Input Format

OD <output data> [, [<output bit number>] [, [<bit width>]]]

Term

<Output data>

Specified output data.
-32768 <= output data (decimal) <= 32767
& 8000 <= output data (hexadecimal) <= & 7FFF

<Output bit number>

Specify the reference bit number of output data.
0 <= bit number <= 32767 (0 for default)

<Bit width>

Specify the bit width of output data.
1 <= bit width <= 16 (16 for default)

Explanation

(1) Outputs a signal (parallel data) unconditionally through the output port to external equipment such as programmable controller. The output data retains after that.

(2) Output data is defined either in decimal or hexadecimal. The data defined in hexadecimal must be headed by "&".

(3) For information on connections, see your manual.

(4) You can specify the range of output signal by the bit width of the OD command..

(5) If you specify the special bit in the parameters OT1-OT3 (see the parameters for the RV-E2 and RV-E3J and the RV-EN series) of the external output, alarm occurs because of the conflict of signals. (If you want to reset output signal with factory default setting, execute "OD 0,0,13 " causing the output bit from 0 to 12 to OFF.)

Sample program (MOVEMASTER Command)

10 OD &FFFF 'Sets the output port of 16 bits from bit 0 to ON.

20 OD &FFFF,10 'Sets the output port of 16 bits width from bit 10 to ON.

30 OD &FFFF,10,15 'Sets the output port of 15 bits from bit 10 to ON.

40 ED 'Ends program.

RV-M2

Function:

Outputs the specified data unconditionally through the output port.

Input Format

OD <output data>

Term

<Output data>

Specified output data.

-32768 <= output data (decimal) <= 32767
& 8000 <= output data (hexadecimal) <= &
7FFF

Explanation

(1) Outputs a signal (parallel data) unconditionally through the output port to external equipment such as programmable controller. The output data retains after that.

(2) Output data is defined either in decimal or hexadecimal. The data defined in hexadecimal must be headed by "&".

(3) For information on connections, see your manual.

Sample program (MOVEMASTER Command)

10 OD &FFFF 'Sets all external output port bits (16 bits) to ON.

40 ED 'Ends program.

RV-M1

Function:

Outputs the specified data unconditionally through the output port.

Input Format

OD <output data>

Term

<Output data>

Specified output data.
0 (-32767) <= output data (decimal) <= 255 (+32767)
&0 (& 8000) <= output data (hexadecimal)
&FF (& 7FFF)
Figure in () is for type A16 or B16 I/O card.

Explanation

(1) Outputs a signal (parallel data) unconditionally through the output port to external equipment such as programmable controller. The output data retains after that.

(2) Output data is defined either in decimal or hexadecimal. The data defined in hexadecimal must be headed by "&".

(3) For information on connections, see your manual.

Sample program (MOVEMASTER Command)

10 OD &FF 'Sets all external output port bits (8 bits) to ON.

40 ED 'Ends program.

COSIMIR® · Movemaster Command

OG (Origin)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Moves to the user-defined origin. (Joint interpolation)

Input Format

OG

Explanation

(1) Moves to the user-defined origin specified by the parameter UOG by joint interpolation.

(2) The attitude of the robot, after executing the OG command, is the same as the attitude after executing NT command. The parameter UOG (see the parameters for the RV-E2 and RV-E3J and the RV-EN series), which defines the moving sequence of origin setting, does not effect the OG command.

(3) The parameter UOG can be set with the teaching operation, too. See your manual for further

Relating Parameters

The attitude of user origin setting can be changed by following parameter.

Parameter UOG

User origin attitude

Default 6-axis: -160.00, -45.00, 50.00, -160.00, -200.00

Default 5-axis: -160.00, -45.00, 0.00, 0.00, -120.00, -200.00

To find the change operation, refer to your manual.

Sample program (MOVEMASTER Command)

10 NT 'Executes the origin setting.

20 MO 2 'Moves to position 2.

30 OG 'Moves to origin.

40 ED 'Ends program.

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Moves to the user-defined origin. (Joint interpolation)

Input Format

OG

Explanation

Moves to the user-defined origin specified in the parameter UOG, using joint interpolation. The axis will move to the origin specified in the additional axes parameter AXOP.

Relating Parameters

Use the parameter below for changing the position of user defined origin designations:

Parameter name UOG

Specified origin

Default 6-axis: -160.00, -45.00, 50.00, -160.00, -200.00 (Default values)

Default 5-axis: -160.00, -45.00, 0.00, 0.00, -120.00, -200.00 (Default values)

Parameter name AXOP

Origin position for the additional axis
0.00, 0.00 (Default value)

Relating Language

NT

Sample program (MOVEMASTER Command)

10 NT 'Executes the robot,s origin setting.

20 MO 2 'Move to position 2.

30 OG 'Move all axes to origin position.

40 ED 'End program.

RV-M2

Function:

Moves to the mechanical reference position for origin setting. (Articulated interpolation)

Input Format

OG

Explanation

(1) Moves the robot to its mechanical origin which has been defined by the NT command or the (NST) (ENT) operation from the teaching box.

(2) The attitude of the robot, after executing the OG command, is the same as the attitude after executing NT command.

(3) Error mode II occurs if the OG command is executed before origin setting

Sample program (MOVEMASTER Command)

10 NT 'Executes the origin setting.

20 MO 2 'Moves to position 2.

30 OG 'Moves to origin.

RV-M1

Function:

Moves to the reference position in the Cartesian coordinate system.

Input Format

OG

Explanation

(1) Moves the robot to its reference position in the Cartesian coordinate system established by the HO command or the (P.S) (0) operation from the teaching box.

(2) If the reference position is yet to be defined, this command moves the robot to a position as determined tentative data stored in the system ROM.

(3) Error mode II occurs if the OG command is executed before origin setting

Sample program (MOVEMASTER Command)

10 NT 'Executes the origin setting.

20 MO 2 'Moves to position 2.

30 OG 'Moves to origin.

COSIMIR® · Movemaster Command

OPN (Open)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

Opens communication channel and specify device.

Input Format

OPN <channel number>, <device number>

Term

<Channel number>

Specify input/output channel number.
0 <= channel number <= 2

<Device number>

Specify input/output device number.
1: Standard RS-232C
2: Standard RS-422

Explanation

(1) The corresponding relation of the channel number input/output devices is specified, and the channel is opened.

(2) Up to three channels can be opened simultaneously. However, multiple <channel numbers> cannot be set in the same <input/output device number>.

(3) The counter, position and character string data can be read in with the INP command.

Sample program (MOVEMASTER Command)

```
10 OPN 1,1        'The standard RS-232-C is opened.

                  '(Channel number 1 is assigned.)

20 INP 1,1,1      'The position data is read from RS-232-

30 INP 1,1,0      'The counter data is read from RS-232-

40 INP 1,$1,2     'The character string data is read from
RS-232-C.
```

RV-E-Robot (RV-E2, RV-E3J)

Opens communication channel and specify device.

Input Format

OPN <channel number>, <device number>

Term

<Channel number>

Specify input/output channel number.
1 <= channel number <= 4

<Device number>

Specify input/output device number.
0: RS-422
1: RS-232C
2: RS-232C-1 (option)
3: RS-232C-2 (option)

Explanation

(1) The corresponding relation of the channel number input/output devices is specified, and the channel is opened.

(2) Up to four channels can be opened simultaneously. However, multiple <channel numbers> cannot be set in the same <input/output device number>.

(3) The counter, position and character string data can be read in with the INP command.

Sample program (MOVEMASTER Command)

```
10 OPN 1,1        'The standard RS-232-C is opened.

                  '(Channel number 1 is assigned.)

20 INP 1,1,1      'The position data is read from RS-232-

30 INP 1,1,0      'The counter data is read from RS-232-
```

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

OR (Or)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.), RV-M2

Function:

ORs the specified data and the internal register data.

Input Format

OR <operation data>

Term

<Operation data >

Specify the data to be operated.
-32768 <= operation data (decimal) <= 32767
& 8000 <= operation data (hexadecimal) <=
&7FFF
@1 <= Counter number <= @99

Explanation

(1) Specify the data to be operated in decimal or hexadecimal. Any hexadecimal value must be headed "&".

(2) The operation result is stored into the internal register and can be changed, compared or read by relevant commands.

(See the ADD, SUB, MUL, DIV, EQ, NE, LG, SM, CL, DR, and XO commands)

(3) Execution of the OR command after the input commands (ID or IN) allows to be set to the required of the parallel input data fetched from the external device.

Sample program (MOVEMASTER Command)

```
10 ID                'Fetches data from external input port.

20 OR &FFF0          'Sets 1 to all bits except lower 4 bits.

30 EQ &FFFF,100      'If the above data are all bit 1, jumps to line 100.
```

40 ED 'Ends program.

:

100 MO 10 'Moves to position 10.

RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 series.

COSIMIR® · Movemaster Command

OT (Output)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function

No operation. Prepared to keep the compatibility with the MOVEMASTER M2 series. Only registration is

Input Format

OT <Output data>

Explanation

This command is prepared for keeping the
with the MOVEMASTER M2 series and has no effect on
program.

RV-M2

Function:

Outputs the specified data synchronously through the output port (using the control signal lines).

Input Format

OT <Output data>

Term

<Output data>

Specified output data.
-32768 <= output data (decimal) <= 32767
& 8000 <= output data (hexadecimal) <= &
7FFF

Explanation

(1) Outputs a signal (parallel data) synchronously
the output port to external equipment such as a
programmable controller. At this time, the control
lines (RDY and Ack signals) must have been connected
the external equipment. The data output to the
equipment is retained..

(2) Output data is defined either in decimal or
hexadecimal. The data defined in hexadecimal must be
headed by "&".

(3) For information on connections, see your manual.

Sample program (MOVEMASTER Command)

10 LPRINT "OT &FFFF"   ' Sets all external output port
bits (16 bits) to ON.

RV-M1

Function:

Outputs the specified data synchronously through the
output port (using the control signal lines).

Input Format

OT <Output data>

Term

<Output data>

Specified output data.
0 (-32768) <= output data (decimal) <= 255
(+32767)
&0 (& 8000) <= output data (hexadecimal)
&FF (& 7FFF)
Figure in () is for type A16 or B16 I/O card.

Explanation

(1) Outputs a signal (parallel data) synchronously the output port to external equipment such as a programmable controller. At this time, the control lines (!RDY and !ACK signals or RDY and Ack signals) have been connected to the external equipment. The data output to the external equipment is retained..

(2) Output data is defined either in decimal or hexadecimal. The data defined in hexadecimal must be headed by "&".

(3) For information on connections, see your manual.

Sample program (MOVEMASTER Command)

```
10 LPRINT "OT &FF"        'Sets all external output port
(8 bits) to ON.
```


COSIMIR® · Movemaster Command

OVR (Override)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Specify program override.

Input Format

OVR <specified override>

Term

<Specified override>

Specify override value. (%)
1 <= specified override <= 200

Explanation

(1) Specifies the ratio of working speed of the robot.

(2) The OVR command is effective for every mode, i.e. joint interpolation, linear interpolation and

circular interpolation.

(3) The actual speed in the program eventually becomes the following.

- Joint interpolation speed = playback override X OVR command setting value X SP command setting value.

- Linear interpolation speed = playback override X OVR command setting value X SP or SD command setting value.

- Here, the playback override can be specified by means of the starting display of teaching box or the external input signal. The override specified by the OVR is called program override.

(4) The initial value of program override is 100 %.

(5) The override value once specified in the program is effective till new value is set or the program ends.

(6) Alarm occurs at the execution of the OVR command the value 0 is set to the specified override.

(7) As the acceleration and deceleration distance required for movement are preset, when the specified speed and acceleration/deceleration are set, if the movement distance is small, the set speed may not be reached.

Sample program (MOVEMASTER Command)

```
10 SP 30     'Sets working speed 30 (100 %).

20 OVR 80    'Sets override 80 %.

30 MO 2      'Moves to position 2.

40 ED        'Ends program.
```

If the playback override is specified to 50 % in the example, the actual override is as follows;

Joint interpolation speed = 50 X 80 X 100 (%) = 40 (%)

The robot moves to position 2 with the speed of 40 % of maximum value.

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Specify program override.

Input Format

OVR <specified override> [, <Valid devices>]

Terms

<Specified override>

Specify override value. (%)
1 <= specified override <= 200

<Valid devices>

Specifies the devices where the specified override will valid.
0: Robot (When omitted)
1: First additional axis
2: Second additional axis

Explanation

(1) This command determines the ratio of the speed between the robot and the additional axis.

(2) The actual speed used in the program will be as follows:

- Robot
Joint interpolation speed
= Playback override x Set value of the OVR command x Set value of (SP).

Linear interpolation speed
= Playback override x Set value of the OVR command x Set value of (SP,SD).

- Additional axis
Speed of additional axis
= Playback override x Set value of the OVR command x Set value of (SP).

Here, the playback override can be specified by means the starting display of the teaching box or the external input signal. The override specified by the OVR is called a "program override„.

(3) If a non-existing device is specified , no processing will be done.

Sample program (MOVEMASTER Command)

10 SP 30 'Sets working speed 30 (100 %).

20 OVR 80,1 'Sets the override in the First axis to 80%

30 MO 1 'Move to position 1.

40 ED 'End program.

Note: If you set the playback override to 50% in the above example, the real specified override will be as follows:
Real specified override = 50/100 x 80/100 x 100 (%) = (%)

First additional axis will move to position 1 at 40% of maximum program execution speed.

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

PA (Pallet Assign)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.), RV-M2

Function:

Defines the number of grid points in the column and directions for the specified pallet.

Input Format

PA <pallet number>, <number of column grid points>, <number of row grid points>

Term

<Pallet number>

Specify number of pallet using.
1 <= pallet number <= 9

<Number of column grid points >

Set grid points of column of pallet.
1 <= number of column grid points <= 32767

<Number of row grid points>

Set grid points of row of pallet.
1 <= number of row grid points <= 32767

Explanation

(1) The PA command must be executed before the calculation command (see the PT command) is

(2) The number of grid points is equivalent to that of actual workpieces arranged on the pallet. For example, with a pallet holding 15 workpieces (3 x 5), the of column and row grid points are 3 and 5,

(3) The column and row directions are decided by the directions of the terminating positions, respectively. the PT command)

Sample program (MOVEMASTER Command)

10 PA 5,20,30 'Defines the pallet 5 as the pallet holding 20 x 30 grid points.

20 SC 51,15 'Sets value 15 to counter 51. points)

30 SC 52,25 'Sets value 25 to counter 52. (row points)

40 PT 5 'Sets the calculated coordinates of grid point to position 5.

50 MO 5 'Moves to position 5. (The grid position)

60 ED 'Ends program.

RV-M1

Function:

Defines the number of grid points in the column and directions for the specified pallet.

Input Format

PA <pallet number>, <number of column grid points>, <number of row grid points>

Term

<Pallet number>

Specify number of pallet using.
1 <= pallet number <= 9

<Number of column grid points >

Set grid points of column of pallet.
1 <= number of column grid points <= 255

<Number of row grid points>

Set grid points of row of pallet.
1 <= number of row grid points <= 255

Explanation

(1) The PA command must be executed before the calculation command (see the PT command) is

(2) The number of grid points is equivalent to that of actual workpieces arranged on the pallet. For example, with a pallet holding 15 workpieces (3 x 5), the of column and row grid points are 3 and 5,

(3) The column and row directions are decided by the directions of the terminating positions, respectively. the PT command)

Sample program (MOVEMASTER Command)

```
10 PA 5,20,30       'Defines the pallet 5 as the pallet
holding 20 x 30 grid points.

20 SC 51,15         'Sets value 15 to counter 51.
points)

30 SC 52,25         'Sets value 25 to counter 52. (row
points)

40 PT 5             'Sets the calculated coordinates
of grid point to position 5.

50 MO 5             'Moves to position 5. (The grid
position)
```

```
60 ED              'Ends program.
```


COSIMIR® · Movemaster Command

PC * (Position Clear)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Clears the data of the specified position(s).

Input Format

PC <position number (a)> [, [<position number (b)>]]

Term

<Position number >

Specify position number deleting.
1 <= position number (a), (b) <= 999
Position number (a) <= position number (b)

Explanation

(1) Deletes all position data between positions (a) and (b). (Position (b) included)

(2) If the position number (a) is greater than the number (b), alarm occur.

* This command can only be executed directly. It be used in a program.

Sample program (BASIC)

```
10 OPEN "COM1 :E83"AS#1        'Opens the RS-232C
communication file

                               'from the personal
computer in BASIC.

20 PRINT #1, "MO 10"           'Moves to position 10.
```

```
30 PRINT #1, "MO 11"          'Moves to position 11.

40 PRINT #1, "MO 12"          'Moves to position 12.

50 PRINT #1, "PC 11"          'Delete the position 11.

60 PRINT 31, "DP"             'Moves to position 10.

70 END                        'Ends program.
```

RV-M2

Function:

Clears the data of the specified position(s).

Input Format

PC <position number (a)> [, [<position number (b)>]]

Term

<Position number >

Specify position number dleting.
1 <= position number (a), (b) <= 999
Position number (a) <= position number (b)

Explanation

(1) Deletes all position data between positions (a) and (b). (Position (b) included)

(2) If the position (b) is not specified, only the data of position (a) is cleared.

(3) Note that the reference position data for origin setting is cleared if any position number is not or 0 is defined, with bit 4 of switch SW1 in the drive side door located in the upper position. At this time, robot must be set to the origin setting reference once again by the HO command or by the (P.S) (0) operation from the teaching box.

* This command can only be executed directly. It be used in a program.

Sample program (BASIC)

```
10 OPEN "COM1 :9600, e, 7, 2"AS#1    'Opens the RS-
communication file
```

'from the personal
computer in BASIC.

20 PRINT #1, "MO 10" 'Moves to position
10.

30 PRINT #1, "MO 11" 'Moves to position
11.

40 PRINT #1, "MO 12" 'Moves to position
12.

50 PRINT #1, "PC 11" 'Delete the
11.

60 PRINT 31, "DP" 'Moves to position
10.

70 END 'Ends program.

RV-M1

Function:

Clears the data of the specified position(s).

Input Format

PC <position number (a)> [, [<position number (b)>]]

Term

<Position number >

Specify position number deleting.
1 <= position number (a), (b) <= 629
Position number (a) <= position number (b)

Explanation

(1) Deletes all position data between positions (a) and (b). (Position (b) included)

(2) If the position (b) is not specified, only the data of position (a) is cleared.

(3) Note that the reference position data for origin setting is cleared if any position number is not
or 0 is defined, with bit 4 of switch SW1 in the drive side door located in the upper position. At this time, robot must be set to the origin setting reference once again by the HO command or by the (P.S) (0)

operation from the teaching box.

* This command can only be executed directly. It cannot be used in a program.

Sample program (BASIC)

10 OPEN "COM1 :9600, e, 7, 2"AS#1 'Opens the RS-232C communication file

'from the computer in BASIC.

20 PRINT #1, "MO 10" 'Moves to position 10.

30 PRINT #1, "MO 11" 'Moves to position 11.

40 PRINT #1, "MO 12" 'Moves to position 12.

50 PRINT #1, "PC 11" 'Delete the position 11.

60 PRINT 31, "DP" 'Moves to position 10.

70 END 'Ends program.

COSIMIR® · Movemaster Command

PD (Position Define)

RV-2AJ, RV-E-Robot with 5 axes (RV-E3J, RV-E5NM, etc.)

Function:

Defines the coordinates (location and angle) of the specified position.

Input Format

PD <Position number>, [<X coordinate value>], [<Y coordinate value>], [<Z coordinate value>], [<A turning

angle>], [<B turning angle>] [, [<R/L>] [, [<A/B>]]] [,
[<O/C>]]

Term

<Position number>

Specify position number defining.
1 <= position number <= 999

<X, Y, Z coordinate>

Specify the location (mm) in XYZ
of the robot. (0 for default)

<A, B turning angle>

Specify the turning angle around roll(A),
(B) axes in XYZ coordinates (degree) of the
robot. (0 for default)

<R/L>

Specify the structure flag of the robot.
or Left)
R: Right (default)
L: Left

<A/B>

Specify the structure flag of the robot.
or Below)
A: Above (Default)
B: Below

<O/C>

Specify open or close state of hand 1.
O: Hand 1 open
C: Hand 1 close

Explanation

(1) The least increment of the coordinate value is 0.01 mm or 0.01 degree (e.g. specify 20.01 for 20.01 mm).

(2) Alarm does not occur even if the specified coordinates exceed the robot's operational space. The command, combined with the SE and the MA command, can define the amount for moving.

Sample program (MOVEMASTER Command)

10 PD 10,50,320,70,40,30,R,A,O 'Defines the location and angle of position 10.

40 MO 10 'Moves to position 10.

50 ED 'Ends program.

RV-1A, RV-E-Robot with 6 axes(RV-E2, RV-E4N, etc.)

Function:

Defines the coordinates (location and angle) of the specified position.

Input Format

PD <Position number>, [<X coordinate value>], [<Y coordinate value>], [<Z coordinate value>], [<A turning angle>], [<B turning angle>] , [<C turning angle>], [, [<R/L>] [, [<A/B>] [, [<N/F>]]]] [, [<O/C>]]

Term

<Position number>

Specify position number defining.
1 <= position number <= 999

<X, Y, Z coordinate>

Specify the location (mm) in XYZ
of the robot. (0 for default)

<A, B,C turning angle>

Specify the turning angle (degree) of roll (A)
and pitch (B)and twist (C) joints in XYZ
coordinates of the robot. (0 for default)

<R/L>

Specify the structure flag of the robot.
or Left)
R: Right (default)
L: Left

<A/B>

Specify the structure flag of the robot.
or Below)
A: Above (Default)
B: Below

<N/F>

Specify the structure flag of the robot. (Non flip or Flip)
N: Non flip (Default)
F: Flip

<O/C>

Specify open or close state of hand 1.
O: Hand 1 open
C: Hand 1 close

Explanation

(1) The least increment of the coordinate value is 0.01 mm or 0.01 degree (e.g. specify 20.01 for 20.01 mm).

(2) Alarm does not occur even if the specified coordinates exceed the robot's operational space. The command, combined with the SE and the MA command, can define the amount for moving.

Sample program (MOVEMASTER Command)

10 PD 10,50,320,70,50,40,30,R,A,N,O 'Defines the location and angle of position 10.

40 MO 10 'Moves to 10.

50 ED 'Ends program.

RV-E-Robot with 5 axes (RV-E3J, RV-E5NM, etc.) and with additional axis

Function:

Sets the coordinate values of a position for a specified number to a specified value (location and posture)

Input Format

PD <Position number>, [<X coordinate value>], [<Y coordinate value>], [<Z coordinate value>], [<A turning angle>], [<B turning angle>] [, <First additional axis> [, <Second additional axis>]] [, [<R/L>] [, [<A/B>]]] [, [<O/C>]]

Term

<Position number>

Specify position number defining.
1 <= position number <= 999

<X, Y, Z coordinate>

Specify the location (mm) in XYZ
of the robot. (0 for default)

<A, B turning angle>

Specify the turning angle around roll(A),
(B) axes in XYZ coordinates (degree) of the
robot. (0 for default)

<First additional axis>

Specifies the moving position for First
additional axis (If omitted, coordinate value
not specified)
The unit is determined by the additional
parameter AXUN.

<Second additional axis>

Specifies the moving position for Second
additional axis (If omitted, coordinate value
not specified)
The unit is determined by the additional
parameter AXUN.

<R/L>

Specify the structure flag of the robot.
or Left)
R: Right (default)
L: Left

<A/B>

Specify the structure flag of the robot.
or Below)
A: Above (Default)
B: Below

<O/C>

Specify open or close state of hand 1.
O: Hand 1 open
C: Hand 1 close

Explanation

(1) If the coordinate values of an additional axis are omitted, it will not be defined.

(2) The unit for the additional axis is specified in the additional axes parameter AXUN.

Related Parameters

additional axes parameter AXUN

Sample program (MOVEMASTER Command)

10 PD 10,50,320,70,50,40,100,0 'Set the position additional axis in position 10.

40 MO 10 'Moves to position

50 ED 'Ends program.

RV-E-Robot with 6 axes(RV-E2, RV-E4N, etc.) and with additional axis

Function:

Sets the coordinate values of a position for a specified number to a specified value (location and posture)

Input Format

PD <Position number>, [<X coordinate value>], [<Y coordinate value>], [<Z coordinate value>], [<A turning angle>], [<B turning angle>] , [<C turning angle>] [, additional axis> [, <Second additional axis>]] [, [<R/L>] [<A/B>] [, [<N/F>]]]] [, [<O/C>]]

Term

<Position number>

Specify position number defining.
1 <= position number <= 999

<X, Y, Z coordinate>

Specify the location (mm) in XYZ of the robot. (0 for default)

<A, B,C turning angle>

Specify the turning angle (degree) of roll (A) and pitch (B)and twist (C) joints in XYZ coordinates of the robot. (0 for default)

<First additional axis>

Specifies the moving position for First additional axis (If omitted, coordinate value not specified)
The unit is determined by the additional parameter AXUN.

<Second additional axis>

Specifies the moving position for Second additional axis (If omitted, coordinate value not specified)
The unit is determined by the additional parameter AXUN.

<R/L>

Specify the structure flag of the robot.
or Left)
R: Right (default)
L: Left

<A/B>

Specify the structure flag of the robot.
or Below)
A: Above (Default)
B: Below

<N/F>

Specify the structure flag of the robot. (Non flip or Flip)
N: Non flip (Default)
F: Flip

<O/C>

Specify open or close state of hand 1.
O: Hand 1 open
C: Hand 1 close

Explanation

(1) If the coordinate values of an additional axis are omitted, it will not be defined.
(2) The unit for the additional axis is specified in the additional axes parameter AXUN.

Related Parameters

additional axes parameter AXUN

Sample program (MOVEMASTER Command)

10 PD 10,50,320,70,50,40,30,100,0 'Set the position additional axis in position 10.

40 MO 10 'Moves to position

50 ED 'Ends program.

RV-M2

Function:

Defines the coordinates (location and angle) of the specified position.

Input Format

PD <Position number>, [<X coordinate value>], [<Y coordinate value>], [<Z coordinate value>], [<Pitch angle>], [<Roll angle>] [, [<O/C>]]

Term

<Position number>

Specify position number defining.
1 <= position number <= 999

<X, Y, Z coordinate>

Specify the location (mm) in XYZ of the robot. (0 for default)

<Roll/Pitch angle>

Specify the turning angle around roll(A), (B) axes in XYZ coordinates (degree) of the robot. (0 for default)

<O/C>

Specify open or close state of hand 1.
O: Hand 1 open
C: Hand 1 close

Explanation

(1) The least increment of the coordinate value is 0.1 or 0.1 degree (e.g. specify 20.1 for 20.1 mm).

(2) Alarm does not occur even if the specified coordinates exceed the robot's operational space. The command, combined with the SF and the MA command, can define the amount for moving.

(3) Any coordinate value defaults to 0.

(4) When the open/close state of the hand (O: Open, Close) is not specified, it is defined by the grip flag GF command.

(5) The PD command can only define a robot position when the coordinates of the end of the hand as determined by the tool command are located ahead of the Z-axis (i. e. the direction the robot faces).

Sample program (MOVEMASTER Command)

10 PD 10,50,320,70,50,40,C 'Defines the location and angle of position 10.

40 MO 10 'Moves to position 10.

50 ED 'Ends program.

RV-M2 with additional axis

Function:

Defines the coordinates (position and angle) of the specified position.

Input Format

PD <Position number>, [<X-axis coordinate value>], [<Y-axis coordinate value>], [<Z-axis coordinate value>], [<Pitch angle>], [<Roll angle>], [<Locomotive coordinate>], [<O/C>]

Where, 1 <= Position number <= 999

Sample Input

PD 10, 0, 380, 300, -70, -40, 150, C

Explanation

(1) The least input increment of the coordinate values 0.1mm or 0.1 degree (e.g. specify 20.1 for 20.1 mm).

(2) No error occurs if the defined coordinates exceed robot,s operational space. This allows you to define a

position representing an incremental movement when command is used in combination with the other command, e.g. SF , MA

(3) Any coordinate value defaults to 0.

(4) When the open/close state of the hand (O: Open, Close) has not been specified, it is defined by the grip flag GF command.

(5) The PD command can only define a robot position when the coordinates of the end of the hand as determined by the tool command are located ahead of the Z-axis (i. e. the direction the robot faces).

Sample Program

100 LPRINT "PD 10, 0, 550, 430, -50, -40, 300, C"

110 LPRINT "PD 20, 0, 0, 20, 0, 0, 0"

120 LPRINT "SF 10, 20"

130 LPRINT "MO 10"

* In the above example, line number 100 defines 10, at which the hand is closed, while line number 110 defines position 20. Line number 120 then redefines position 10 as a position being shifted 20mm in the Z- as determined by position 20. Line number 130 moves robot to the new position 10.

RV-M1

Function:

Defines the coordinates (location and angle) of the specified position.

Input Format

PD <Position number>, [<X coordinate value>], [<Y coordinate value>], [<Z coordinate value>], [<Pitch angle>], [<Roll angle>] [, [<O/C>]]

Term

<Position number>

Specify position number defining.
1 <= position number <= 629

<X, Y, Z coordinate>

Specify the location (mm) in XYZ of the robot. (0 for default)

<Roll/Pitch angle>

Specify the turning angle around roll(A), (B) axes in XYZ coordinates (degree) of the robot. (0 for default)

<O/C>

Specify open or close state of hand 1.
O: Hand 1 open
C: Hand 1 close

Explanation

(1) The least increment of the coordinate value is 0.1 or 0.1 degree (e.g. specify 20.01 for 20.01 mm).

(2) Alarm does not occur even if the specified coordinates exceed the robot's operational space. The command, combined with the SF and the MA command, can define the amount for moving.

(3) Any coordinate value defaults to 0.

(4) When the open/close state of the hand (O: Open, Close) is not specified, it is defined by the grip flag GF command.

(5) The PD command can only define a robot position when the coordinates of the end of the hand as determined by the tool command are located ahead of the Z-axis (i. e. the direction the robot faces).

Sample program (MOVEMASTER Command)

```
10 PD 10,50,320,70,50,40,C       'Defines the location
angle of position 10.

40 MO 10                         'Moves to position 10.

50 ED                            'Ends program.
```

COSIMIR® · Movemaster Command

PL (Position Load)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.), RV-M2

Function:

Replaces position (a) by position (b).

Input Format

PL <position number (a)>, <position number (b)>

Term

<Position number (a) >

Specify the position number. (Destination)
1 <= position number (a) <= 999

<Position number (b)>

Specify the position number. (Source)
1 <= position number (b) <= 999

Explanation

(1) After execution, the PL command causes the coordinates of position equivalent to those of position (b) and the old coordinates of position (a

(2) After execution, the PL command also assigns the hand state at posit position (a).

(3) Alarm occurs if position (b) is not defined.

(4) A new position is created if the position (a) is not defined.

Sample program (MOVEMASTER Command)

```
10 HE 2      'Sets the current coordinates and hand state to position 2.

20 PL 3,2    'Replaces the coordinates of position 3 by position 2.

30 ED        'Ends program.
```

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Substitute the coordinate values of a specified position number (a) with
that have been entered in another postition (b)

Input Format

PL <position number (a)>, <position number (b)>

Term

<Position number (a) >

Specify the position number being substituted. (Destination)
1 <= position number (a) <= 999

<Position number (b)>

Specify the position number of the substitute . (Source)
1 <= position number (b) <= 999

Explanation

(1) The table below shows how the existence/nonexistence of position d
additional axis determines the contents position number (a) after the PL
executed.

		Pattern 1	Pattern 2	P
Position (a)	**Robot**	Exists	Exists	E
Position (a)	**Additional axis**	Doesn,t exist	Exists	E
Position (b)	**Robot**	Exists	Exists	E
Position (b)	**Additional axis**	Exists	Doesn,t exist	E
PL (a),(b)	**Robot**	(a) <- (b)	(a) <- (b)	(
PL (a),(b)	**Additional axis**	(a) <- (b)	Doesn,t exist	(

Sample program (MOVEMASTER Command)

10 PD 3, 50,.320, 70, 50, 40, 30 '**6 axis:** Set position 3
(without position data for the additional axis)

10 PD 3, 50,.320, 70, 50, 40 '**5 axis:** Set position 3
(without position data for the additional axis)

20 HE 2 'Set the current position in position
(including the position data for the additional axis)

30 PL 3,2 'Copy position 2 onto position 3
(Corresponds to "Pattern 1" in the table above.)

40 ED 'End program.

RV-M1

Function:

Replaces position (a) by position (b).

Input Format

PL <position number (a)>, <position number (b)>

Term

<Position number (a) >

Specify the position number. (Destination)
1 <= position number (a) <= 629

<Position number (b)>

Specify the position number. (Source)
1 <= position number (b) <= 629

Explanation

(1) After execution, the PL command causes the coordinates of position equivalent to those of position (b) and the old coordinates of position (a

(2) After execution, the PL command also assigns the hand state at posit position (a).

(3) Alarm occurs if position (b) is not defined.

(4) A new position is created if the position (a) is not defined.

Sample program (MOVEMASTER Command)

```
10 HE 2      'Sets the current coordinates and hand state to position 2.

20 PL 3,2    'Replaces the coordinates of position 3 by position 2.

30 ED        'Ends program.
```

PMR (Parameter Read)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Reads contents of parameter specified.

Input Format

PMR ["<parameter name>"]

Term

<Parameter name >

Specify parameter name.
Only defined parameter names are valid. (Defined for default)

Explanation

(1) Outputs the specified parameters (see the for the RV-E2 and RV-E3J and the RV-EN series) from RS-232C port.

The output format:

Parameter name, contents

(2) When you omitted parameter name, a parameter defined next is output. (Parameters are defined in alphabetical order.)If the last parameter is read out, only carriage return (Hex. 0D) is output.

(3) When you specify a parameter that does not exist, contents of next parameter specifying in alphabetical order is output.

Sample program (BASIC)

```
10 OPEN "COM1 :E83" AS#1                'Opens the RS-232C communication file
                                        'from the personal computer in BASIC.
20 INPUT "Parameter name is" ;J$        'Inputs the parameter name.
30 PRINT #1,"PMR"+CHR$(&H22)+J$+CHR$(&H22)
```

'Transmits the PMR command followed

'by the parameter name.

40 LINE INPUT #1,A$ 'Saves the received data to A $.

50 PRINT A$ 'Displays the contents of A $.

60 END 'Ends program.

RUN 'Run the program.

Parameter name is ? ADL 'Enter the parameter name.

ADL,0.20,0.20 'Outputs the contents of ADL

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

PMW (Parameter Write)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Renews the contents of the specified parameter.

Input Format

PMW "<parameter name>", "<parameter contents>"

Term

<Parameter name>

Specify the parameter name changing contents.

<Parameter contents>

Specify the contents that you want to to.

Explanation

(1) Outputs the contents of the specified parameter through the RS-232C port.

(2) When you specified the parameter that does not exist, no operation is executed.

(3) For information on parameters, see the list of all parameters (for the RV-E2 and RV-E3J and the RV-EN series)

(4) The specified parameter becomes effective after having turned off and on the power supply. Give attention that even if you have changed the contents the parameter, the old contents is still effective until power supply is turned on again.

Sample program (MOVEMASTER Command)

PMW "ADL" , "0.40,0.40";Sets the contents of parameter ADL 0.40,0.40

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

PR (Position Read)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Reads the coordinates of the specified position and the open/close state of the hand. (Using RS-232C)

Input Format

PR [<position number>]

Term

<Position number>

Specify the position number that you want read.
0 <= position number <= 999
(If omitted, the current position number is valid.)

Explanation

(1) Outputs the coordinates of the specified position the open/close state of the hand through the RS-232C port. If the position number is omitted or equals 0, the current position number is output .

(2) The data is ASCII coded as follows; The least increment is 0.01 mm or 0.01 degree.

Output format

6-axis type:X, Y, Z coordinate value, A, B,C turning degree, R/L, A/B,N/F,O/C

5-axis type:X, Y, Z coordinate value, A, B turning angle degree, R/L, A/B,O/C

(3) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(4) If you read the position that is not used in the program and not defined yet, "zero" is output with each coordinate.

6-axis type:0,0,0,0,0,0

5-axis type:0,0,0,0,0

If you read the position that is already used in the program but not defined yet, "0.00" is output with each coordinate.

6-axis type:0.00,0.00,0.00,0.00,0.00,0.00

5-axis type:0.00,0.00,0.00,0.00,0.00

(5) If you specify "zero" to the position number or omit you can identify the current position number when an alarm occurs in executing moving command.

Sample program (BASIC)

(1) 6 -axis type

10 OPEN "COM1 :E83"AS#1 'Opens the RS-232C communication file

'from the personal computer in BASIC.

20 INPUT "Position number is"; P 'Inputs the position number that you want to read

30 PRINT #1,"PR"+STR$(P) 'Transmits the PR command followed by the position number.

40 LINE INPUT #1,A$ 'Saves the received data to A$.

50 PRINT A$ 'Displays the to the screen.

60 END 'Ends program.

RUN 'Run the program.

Position number is ? 15 'Enter the position number.

+10.00,+380.00,300.00,-70.00,50.00,40.00,R,A,N,C 'Outputs the contents of the position.

(2)5-axis type

10 OPEN "COM1 :E83"AS#1 'Opens the RS-232C communication file

'from the personal computer in BASIC.

20 INPUT "Position number is"; P 'Inputs the position

number that you want to read

30 PRINT #1,"PR"+STR$(P) 'Transmits the PR command followed by the position number.

40 LINE INPUT #1,A$ 'Saves the received data to A$.

50 PRINT A$ 'Displays the contents to the screen.

60 END 'Ends program.

RUN 'Run the program.

Position number is ? 15 'Enter the position number.

+10.00,+380.00,300.00,50.00,40.00,R,A,C 'Outputs the contents of the position.

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Reads the open/close state of the hand, and the and coordinate values of a specified position. (For the 232-C)

Input Format

PR [<position number>]

Term

<Position number>

Specify the position number that you want read.
0 <= position number <= 999
(If omitted, the current position number is valid.)

Explanation

(1) The output configuration is as shown below, and is output in ASCII code entirely.

The least increment for the output is 0.01mm or 0.01 deg.

Output configuration:

X,Y,Z, A,B,C, First additional axis, Second additional R/L, A/B, N/F, O/C (**6 axis**)

X,Y,Z, A,B, First additional axis, Second additional axis, R/L, A/B, O/C (**5 axis**)

(2) An example of an output configuration is as follows:

for 6 axis:

- First additional axis and Second additional axis are existent or undefined:
X,Y,Z, A,B,C, R,A,N,O

- First additional axis exists, Second additional axis is non-existent or undefined:
X,Y,Z, A,B,C, First additional axis, +0.00, R,A,N,O

- First additional axis and Second additional axis exist:
X,Y,Z, A,B,C, First additional axis, Second additional R,A,N,O

- First additional axis is non-existent or undefined, Second additional axis exists:
X,Y,Z, A,B,C, + 0.00 , Second additional axis, R,A,N,O

for 5 axis:

- First additional axis and Second additional axis are existent or undefined:
X,Y,Z, A,B, R,A,O

- First additional axis exists, Second additional axis is non-existent or undefined:
X,Y,Z, A,B, First additional axis, +0.00, R,A,O

- First additional axis and Second additional axis exist:
X,Y,Z, A,B, First additional axis, Second additional axis, R,A,O

- First additional axis is non-existent or undefined, Second additional axis exists:
X,Y,Z, A,B, + 0.00 , Second additional axis, R,A,O

Sample program (BASIC)

(1) 6 -axis type

10 OPEN "COM1 :E83"AS#1 'Opens the RS-232C communication file

'from the

computer in BASIC.

20 INPUT "Position number is"; P 'Inputs the position number that you want to read

30 PRINT #1,"PR"+STR$(P) 'Transmits the command followed by the position number.

40 LINE INPUT #1,A$ 'Saves the received data to A$.

50 PRINT A$ 'Displays the contents to the screen.

60 END 'Ends program.

RUN 'Run the program.

Position number is ? 15 'Enter the number.

+10.00,+380.00,300.00,-70.00,50.00,40.00,+100.00,+200.00,R,A,N,C 'Outputs the contents of the position.

(2)5-axis type

10 OPEN "COM1 :E83"AS#1 'Opens the RS-232C communication file

'from the personal computer in BASIC.

20 INPUT "Position number is"; P 'Inputs the position number that you want to read

30 PRINT #1,"PR"+STR$(P) 'Transmits the command followed by the position number.

40 LINE INPUT #1,A$ 'Saves the received data to A$.

50 PRINT A$ 'Displays the contents to the screen.

60 END 'Ends program.

RUN 'Run the

Position number is ? 15 'Enter the position number.

+10.00,+380.00,300.00,-
70.00,50.00,100.00,200.00,R,A,C 'Outputs the
of the position.

RV-M2

Function:

Reads the coordinates of the specified position and the open/close state of the hand. (Using RS-232C)

Input Format

PR <Position number>

Term

<Position number>

Specify the position number that you want read.
0 <= position number <= 999

Explanation

(1) Outputs the coordinates of the specified position the open/close state of the hand through the RS-232C port. If the position number is omitted or equals 0, the current position number is output .

(2) The data is ASCII coded as follows; The least increment is 0.1 mm or 0.1 degree.

Output format

X, Y, Z coordinate value, A, B turning angle degree,

(3) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(4) If you read the position that is not used in the program and not defined yet, "zero" is output with each coordinate.

0,0,0,0,0,0

Sample program (BASIC)

10 OPEN "COM1 :9600, E, 7, 2"AS#1 'Opens the RS-

232C communication file

'from the personal computer in BASIC.

20 INPUT "Position number is"; P 'Inputs the position number that you want to read

30 PRINT #1,"PR"+STR$(P) 'Transmits the command followed by the ;position number.

40 LINE INPUT #1,A$ 'Saves the received data to A$.

50 PRINT A$ 'Displays the contents to the screen.

60 END 'Ends program.

RUN 'Run the program.

Position number is ? 15 'Enter the number.

+10.0,+380.0,+300.0,-50.0,-40.0,C 'Outputs the contents of the position.

RV-M2 with additional axis

Function:

Reads the coordinates of the specified position and the open/close state of the hand. (Using RS-232C)

Input Format

PR <Position number>

Where, 1 <= Position number <=999

Sample Input

PR 5

Explanation

(1) This command causes the coordinates of a specified position and the open/close state of the hand to be output from the RS232C port.

(2) The data is output in ASCII as indicated below. The least output increment is 0.1mm or 0.1degree (e.g.

is displayed for 20.1mm).
Output format: X-axis coordinate, Y-axis coordinate, Z-axis coordinate, pitch angle, roll angle, Locomotive coordinate, 0 or C 0D (Hex)
The hand open/close state is then output as 0 for open and C for close.

(3) The delimiter of the data is a comma (,: Hex. 2C) the terminator is a carriage return (CR, Hex. 0D). If the data is to be received by a personal computer, it is necessary to handle the entire data string up to hex. 0D. The BASIC equivalent to this is the LINE INPUT

(4) When an undefined position is read, 0,s are returned (0, 0, 0, 0, 0, 0).

Sample Program (BASIC)

10 OPEN "COM1 :9600, E, 7, 2"AS#1 'Opens the RS-communication file

'from the personal
in BASIC.

20 INPUT "Position number is"; P 'Inputs the position number that you want to read

30 PRINT #1,"PR"+STR$(P) 'Transmits the PR command followed by the ;position number.

40 LINE INPUT #1,A$ 'Saves the data to A$.

50 PRINT A$ 'Displays the contents to the screen.

60 END 'Ends program.

RUN 'Run the program.

Position number is ? 15 'Enter the position number.

+10.0,+380.0,+300.0,-70.0,-40.0, 250,C 'Outputs the contents of the position.

RV-M1

Function:

Reads the coordinates of the specified position. (Using RS-232C)

Input Format

PR <Position number>

Term

<Position number>

Specify the position number that you want read.
0 <= position number <= 629

Explanation

(1) Outputs the coordinates of the specified position through the RS-232C port. If the position number is omitted or equals 0, only the current position number output .

(2) The data is ASCII coded as follows; The least increment is 0.1 mm or 0.1 degree.

Output format

X, Y, Z coordinate value, pitch angle, roll angle

(3) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(4) If you read the position that is not used in the program and not defined yet, "zero" is output with each coordinate.

0,0,0,0,0,0

Sample program (BASIC)

10 OPEN "COM1 :9600, E, 7, 2"AS#1 'Opens the RS-232C communication file

'from the computer in BASIC.

20 INPUT "Position number is"; P 'Inputs the position number that you want to read

30 PRINT #1,"PR"+STR$(P) 'Transmits the PR command followed by the ;position number.

```
40 LINE INPUT #1,A$                        'Saves the
received data to A$.

50 PRINT A$                                'Displays the
contents to the screen.

60 END                                     'Ends program.

RUN                                        'Run the

Position number is ? 15                    'Enter the
position number.

+10.0,+380.0,+300.0,-50.0,-40.0            'Outputs the
contents of the position.
```


COSIMIR® · Movemaster Command

PRN * (Print)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

The counter number setting value, position number coordinate value or the character string number data is transmitted from the personal computer in regard to INP command. (Using RS-232-C)

Input Format

PRN <counter value> | <position coordinates>| "<character string data>"

Term

<Counter value>

Specify the counter value setting to a counter.
-32768 <= Counter value (decimal) <= 32767
& 8000 <= Counter value (hexadecimal) <= & 7FFF

<Position coordinates>

Specify the coordinates value setting to a position. Specify the following coordinates similar to the PD command.
(See the PD command)
(1) 6-axis type
<X, Y, Z coordinates>, <A, B, C turning
<R/L>, <A/B>, <N/F>, <O/C>
(2) 5-axis type
<X, Y, Z coordinates>, <A, B turning angle>,
<R/L>, <A/B>,<O/C>

<Character string data>

Specify the character string to be set.
1 <= Character string data <= 120 (number characters)
Usable characters:
numerals (0 to 9),
alphabetic characters(A to Z),
symbols (!@#, etc.)
Unusable characters: *+,./;:[]\'

Explanation

(1) Transmits the setting value of counter, the coordinates value of position or the character strings from the personal computer through the RS-232C port responding to the INP command in the program.

(2) The robot becomes wait condition in the INP command till the date is transmitted from personal computer by executing the PRN command.

(3) You can execute the PRN command prior to the INP command during the execution of the program.

(4) When transmitting character string data, enclose character string in double quotations (").

* This command can only be executed directly. It be used in a program.

Sample program

<Personal computer program>

10 OPEN "COM1 :E83"AS#1 'Opens the RS-communication file

'from the personal computer in BASIC.

20 INPUT "Counter data is" ;J 'Enter the setting value of counter

'from the personal computer.

30 PRINT #1,"PRN"+STR$ (J) 'Transmits the setting value of counter.

40 PRINT #1,"PRN 100,0,0,0,0,0" 'Transmits the coordinates value of position.

50 END 'Ends program.

<Robot program>

10 OPN 2,1 'Opens the RS-232C port.

20 IN 2,1,0 'Reads the data from RS-232C port to counter 1.

30 IN 'Add 1 to counter 1.

50 MO 5 'Moves to position 5.

RV-E-Robot (RV-E2, RV-E3J)

Function:

The counter number setting value or position number coordinate value is transmitted from the personal computer in regard to the INP command. (Using RS-C)

Input Format

PRN <counter value> | <position coordinates>

Term

<Counter value>

Specify the counter value setting to a counter.
-32768 <= Counter value (decimal) <= 32767
& 8000 <= Counter value (hexadecimal) <= & 7FFF

<Position coordinates>

Specify the coordinates value setting to a position. Specify the following coordinates

similar to the PD command.
(See the PD command)
(1) 6-axis type
<X, Y, Z coordinates>, <A, B, C turning
<R/L>, <A/B>, <N/F>, <O/C>
(2) 5-axis type
<X, Y, Z coordinates>, <A, B turning angle>,
<R/L>, <A/B>,<O/C>

Explanation

(1) Transmits the setting value of counter or the coordinates value of position from the personal through the RS-232C port responding to the INP in the program.

(2) The robot becomes wait condition in the INP command till the date is transmitted from personal computer by executing the PRN command.

(3) You can execute the PRN command prior to the INP command during the execution of the program.

* This command can only be executed directly. It be used in a program.

Sample program

<Personal computer program>

10 OPEN "COM1 :E83"AS#1 'Opens the RS-232C communication file

'from the personal computer in BASIC.

20 INPUT "Counter data is" ;J 'Enter the setting of counter

'from the personal computer.

30 PRINT #1,"PRN"+STR$ (J) 'Transmits the value of counter.

40 PRINT #1,"PRN 100,0,0,0,0,0" 'Transmits the coordinates value of position.

50 END 'Ends program.

<Robot program>

10 OPN 2,1 'Opens the RS-232C port.

20 IN 2,1,0 'Reads the data from RS-232C port to counter 1.

30 IN2,5,1 'Reads the data from RS-232C port to counter 5.

30 IC1 'Add 1 to counter 1.

50 MO 5 'Moves to position 5.

RV-E-Robot (RV-E2, RV-E3J) with additional axis

Function:

Transmits either the setting value of the counter or the coordinate values of a position in regard to the INP command. (For the RS-232-C)

Input Format

PRN <counter value> | <position coordinates>

Term

<Counter value>

Specify the counter value setting to a counter.
-32768 <= Counter value (decimal) <= 32767
& 8000 <= Counter value (hexadecimal) <= & 7FFF

<Position coordinates>

Specify the coordinates value setting to a position. Specify the following coordinates similar to the PD command.
(See the PD command)
(1) 6-axis type
<X, Y, Z coordinates>, <A, B, C turning <First additional axis>, <Second additional axis>, <R/L>, <A/B>, <N/F>, <O/C>
(2) 5-axis type
<X, Y, Z coordinates>, <A, B turning angle>, <First additional axis>, <Second additional axis>, <R/L>, <A/B>,<O/C>

Explanation

(1) If the coordinate values of the additional axis are omitted for the additional axis system, nothing will be specified.

Sample program

<Personal computer program>

10 OPEN "COM1 :E83"AS#1 'Opens the RS-232C communication file

'from the personal computer in BASIC.

20 INPUT "Counter data is" ;J 'Enter the setting value of counter

'from the personal computer.

30 PRINT #1,"PRN"+STR$ (J) 'Transmits the setting value of counter.

40 PRINT #1,"PRN 100,0,0,0,0,0,100,0,R,A,N" '**(6-axis)** Transmits the coordinates value of position.

40 PRINT #1,"PRN 100,0,0,0,0,100,0,R,A" '**(5-axis)** Transmits the coordinates value of position.

50 END 'Ends program.

<Robot program>

10 OPN 2,1 'Opens the RS-232C port.

20 INP 2,1,0 'Load data from the standard RS- into counter 1

30 INP 2,5,1 'Load data from the standard RS- into counter 5

40 IC 1 'Add 1 to counter 1

50 MO 5 'Moves to position 5.

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

PT (Pallet)

Function:

Calculates the coordinates of a grid point on the specified pallet and set coordinates value to the specified position.

Input Format

PT <pallet number>

Term

<Pallet number>

Specify the number of pallet using.
1 <= pallet number <= 9

Explanation

(1) Calculates the coordinates of a grid point on the specified pallet and coordinates to the position which number is corresponding to the specifi number. Before the PT command is executed, the pallet definition comr must be executed for the pallet to be used. After the PT command has t executed, the position data previously defined for the target position is

(2) In order for the PT command to be executed, the pallet positions (gr at four corners of the pallet) must be properly defined which identify a pallet and the pallet counters (column and row) be properly set that spe particular grid point on the pallet. If the pallet positions and pallet coun properly defined, therefore, execution of the PT command allows the of a grid point to be defined as the position number equivalent to the pa number. The following is a listing of a combination of pallet positions an counters corresponding to each pallet number.

(3) Alarm occurs if the pallet position have not been defined and the pal counters have not been set or have been set to have values exceeding tl defined by the PA command. Alarm does not occur, however, even when coordinates obtained for the grid point exceed the robot's operational sp

Pallet number	1	2	3	4	5	6	7	8
Pallet reference point	10	20	30	40	50	60	70	80
Pallet column	11	21	31	41	51	61	71	81

terminating position								
Pallet row terminating position	12	22	32	42	52	62	72	82
Pallet corner position opposite to reference	13	23	33	43	53	63	73	83
Pallet column counter	11	21	31	41	51	61	71	81
Pallet row counter	12	22	32	42	52	62	72	82
Pallet grid position	1	2	3	4	5	6	7	8

(4) The open or close state of the hand at the target grid point is the sam that in the pallet reference position.

(5) When executing the PT command, the tool length of the hand to be us be properly defined by the TL command. The robot must be taught throug pallet positions (four corners) using the predefined correct tool length.

(6) Pay attention when you use pallet nine because the counters (91,92) c nine are common counters among programs and other program may use th counters.

Sample program (Movemaster command)

Suppose there is a pallet on which a total of 24 workpieces are arranged column direction and 6 in the row direction. Then have the system comp coordinates of the workpiece placed in the grid position (2,4), i.e. the se grid in the column direction and the fourth grid in the row direction, and robot hand to reach that position. Assume that pallet 7 is used.

You must teach the positions at four corners (70,71,72,73) in advance.

```
10 TL 200        'Sets the tool length equivalent to the hand using.

20 PA 7,4,6      'Sets the pallet number and the grid points of column a

30 SC 71,2       'Sets the number of grid point in column direction.

40 SC 72,4       'Sets the number of grid point in row direction.

50 PT 7          'Allows the coordinates of the target grid point to be se
position 7.

60 MO 7          'Moves to position 7.

70 ED            'Ends program.
```

COSIMIR® · Movemaster Command

PW (Pulse Wait)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Waits for in-position of servomotor about all joints till becomes within the specified value.

Input Format

PW <positioning pulse>

Term

<Positioning pulse >

Specify the judgment pulse number of in-position.
1 <= positioning pulse <= 10000

Explanation

(1) Waits for in-position of servomotor about all joints it becomes within the specified value.

(2) If you need the positioning of high accuracy when chucking a workpiece at a position, set small value for positioning pulse. If you need the positioning of low accuracy, set large value. If you set small value to the positioning pulse, the robot waits for positioning having the same effect as the TI command is executed.

(3) If you set small value to the positioning pulse when the handling workpiece is relatively heavy or the robot moving at high speed, it may take a longer time to position than usual.

(4) Initial value of positioning pulse is 10000 pulse.

(5) When the setting value exceeds the above limit, alarm occurs.

Sample program (MOVEMASTER Command)

10 PW 100 'Waits for the positioning pulse becoming within 100 pulses.

20 MO 1 'Moves to position 1.

30 GC 'Closes hand.

40 ED 'Ends program.

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

PX (Position Exchange)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.), RV-M2

Function:

Exchanges the coordinates of the specified position for those of another specified position.

Input Format

PX <position number (a) >, <position number (b) >

Term

<Position number>

Specify the position number exchanging.
1 <= position number (a), (b) <= 999

Explanation

(1) After the PX command is executed, the coordinates position (a) are exchanged for those of position (b)

(2) The open or close state of the hand at position (a) also exchanged for that at position (b).

(3) Alarm occurs if positions (a) and (b) have not been predefined.

Sample program (MOVEMASTER Command)

```
10 HE 2                    'Define the current position
position 2

20 MJ 20,30,10,0,0,0       'Drives each joint by the
specified amount.

30 GO                      'Opens the hand.

40 HE 3                    'Define the current position
position 3.

50 PX 2,3                  'Exchanges the coordinates
value of

                          'position 2 for those of
3.

60 ED                      'Ends program.
```

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Exchanges each coordinate value of a specified position with those of another specified position.

Input Format

PX <position number (a) >, <position number (b) >

Term

<Position number>

Specify the position number exchanging.
1 <= position number (a), (b) <= 999

Explanation

(1) The position data for the additional axis are also exchanged.

(2) Position data for additional axis will be exchanged even if one side does not include such data.

Sample program (MOVEMASTER Command)

10 PD 3,50,320,70,50,40,30 '**(6-axis)** Set position 3 (without additional axis data)

10 PD 3,50,320,70,50,40 '**(5-axis)** Set position 3 (without additional axis data)

20 HE 2 'Set position 2 with the current position

'(Including the position for the additional axis)

30 PX 2,3 'Exchange the coordinate values, posture, and additional axis

'position data of position 2 with those of position 3.

'The position data for the additional axis will disappear in position 2.

60 ED 'End program

RV-M1

Function:

Exchanges the coordinates of the specified position for those of another specified position.

Input Format

PX <position number (a) >, <position number (a) >

Term

<Position number>

Specify the position number exchanging.
1 <= position number (a), (b) <= 629

Explanation

(1) After the PX command is executed, the coordinates position (a) are exchanged for those of position (b)

(2) The open or close state of the hand at position (a) also exchanged for that at position (b).

(3) Alarm occurs if positions (a) and (b) have not been predefined.

Sample program (MOVEMASTER Command)

10 HE 2 'Define the current position to

position 2

20 MJ 20,30,10,0,0,0 'Drives each joint by the specified amount.

30 GO 'Opens the hand.

40 HE 3 'Define the current position to position 3.

50 PX 2,3 'Exchanges the coordinates value of
'position 2 for those of position 3.

60 ED 'Ends program.

COSIMIR® · Movemaster Command

QN (Question Number)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Reads the program name or the program information.

Input Format

QN [<Program name>]

Term

<Program name>

Specify the robot program name to be read.
(Max. 8 characters)
Possible letter used:Digits (0-9)
Characters (A - Z)
Symbols ((! @ # $ % ^ & ( ) _ | { } -)
Impossible letters used:* + , . / : ; = ? [\ ] '
Special specification:
When you specified only numeric value, the program name is handled for number.
Need to enclose program name with " " in

case of character used.

Explanation

(1) Outputs the selected program name or the selected program information through the RS-232C port. If you omitted the program name, the selected program is turned over and if you specified it, the information about the program is turned over.

(2) The output format is ASCII coded as follows;

- The program number format: "N" followed by name"

- The program information format: Used number of used number of positions, used number of counters

(3) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

Sample program (BASIC)

10 OPEN "COM1 :E83"AS#1 'Opens the 232C communication file

'from the personal computer in BASIC.

30 PRINT #1,"QN" 'Transmits QN command.

30 LINE INPUT #1,A$ 'Saves the received data to A$.

40 PRINT "Selected program is " : A $ 'Displays the contents of A$ to the screen.

60 END 'Ends.

RUN 'Run the program.

Selected program is N10 'Outputs the program name.

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

RC (Repeat Cycle)

Function:

Repeats the loop specified by the NX command the specified number of times.

Input Format

RC <number of repeated cycles>

Term

<Number of repeated cycles>

Specify the number of times repeating.
1 <= number of repeated cycles (decimal) <= 32767
& 0001 <= number of repeated cycles (hexadecimal) <= & 7FFF

Explanation

(1) Used with the NX command to cause the loop specified by the NX command to be executed the specified number of times and causes the line number following NX to be subsequently executed.

(2) To incorporate another loop (between RC and NX) the existing loop is called "nesting". Up to 9 nesting are possible.

Sample program (MOVEMASTER Command)

10 MO 1 'Moves to position 1.

20 RC 3 'Repeats loop delimited by NX three times

30 MO 2 'Moves to position 2.

```
40 MO 3    'Moves to position 3. Loop.
50 MO 4    'Moves to position 4.
60 NX      'Delimits the loop.
70 MO 5    'Moves to position 5.
80 ED      'Ends program.
```


COSIMIR® · Movemaster Command

RN * (Run)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Executes the specified part of commands in a program.

Input Format

RN [<start line number> [, <end line number> [, program name]]]

Term

<Start line number>

Specify the line number beginning.
1 <= start line number <= 9999 (The top line for default)

<End line number>

Specify the line number ending.
1 <= end line number <= 9999
(The last line or ED command line for

<Program name>

Specify the program name. (Max. 8 characters)
Possible letter used:Digits (0-9)
Characters (A - Z)

Symbols (! @ # $ % ^ & ( ) _ | { } -)
Impossible letter used:* + , . / : ; = ? [\ ] '
Special specification:
When you specified only numeric value, the program name is handled for number.
Need to enclose program name with " " in the case of characters used.

Explanation

(1) Runs the program starting with the specified line and ending with the line one ahead the specified ending line.

(2) If the program is to continue, restart with the line.

(3) If the teaching box is connected, the line number being executed is shown on its display.

(4) You can select the program by appointing the <program name>. In this case, the specified program becomes the target program and the program starts the specified line.

(5) If you describe the RN command with line number program, no operation is executed.

(6) When the RN command is executed, the contents of the counter remain intact (not initialized).

(7) The following name is identified as the same.

An example:

Handled as the same. 1,01,001,00000001 (Only numeric value)

Handled as not the same.1,1 A, A0 _ 001 (Includes characters)

(8) The letters that controller can indicate to the LED 0-9, A •` Z (simplified).

* This command can only be executed directly. It be used in a program.

Sample program (BASIC)

10 OPEN "COM1 :E83"AS#1 'Opens the RS-232C communication file

'from the personal computer in BASIC.

20 PRINT #1,"RN 100,,2" 'Executes the program 2 from line 100.

30 END 'Ends program.

RV-M2

Function:

Executes the specified part of commands in a program.

Input Format

RN [<start line number> [, <end line number>]]

Term

<Start line number>

Specify the line number beginning.
1 <= start line number <= 3584 (The top line for default)

<End line number>

Specify the line number ending.
1 <= end line number <= 3584
(The last line or ED command line for

Explanation

(1) Runs the program starting with the specified line and ending with the line one ahead the specified ending line.

(2) If the program is to continue, restart with the line.

(3) If the teaching box is connected, the line number being executed is shown on its display.

(4) If the starting line number is not specified, the program starts with the first line.

(5) When the RN command is executed, the contents of the counter remain intact (not initialized).

* This command can only be executed directly. It be used in a program.

Sample program (BASIC)

```
10 LPRINT "100 MO 10"

20 LPRINT "110 MO 12"

30 LPRINT "120 GC"

40 LPRINT "130 MO 17"

50 LPRINT "140 ED"

60 LPRINT "RN 100"          'Runs program starting with
line number 100
```

RV-M1

Function:

RN [<start line number> [, <end line number>]]

Term

<Start line number>

Specify the line number beginning.
1 <= start line number <= 2048 (The top line for default)

<End line number>

Specify the line number ending.
1 <= end line number <= 2048
(The last line or ED command line for

Explanation

(1) Runs the program starting with the specified line and ending with the line one ahead the specified ending line.

(2) If the program is to continue, restart with the line.

(3) If the teaching box is connected, the line number being executed is shown on its display.

(4) If the starting line number is not specified, the program starts with the first line.

(5) When the RN command is executed, the contents of the counter remain intact (not initialized).

* This command can only be executed directly. It cannot be used in a program.

Sample program (BASIC)

10 LPRINT "100 MO 10"

20 LPRINT "110 MO 12"

30 LPRINT "120 GC"

40 LPRINT "130 MO 17"

50 LPRINT "140 ED"

60 LPRINT "RN 100" 'Runs program starting with line number 100

COSIMIR® · Movemaster Command

RS * (Reset)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Resets the program and error condition.

Input Format

RS [<reset number>]

Term

<Reset number>

Specify the contents of reset.
Reset contents
0: Cancels alarm and resets program. (Default)
1: Makes all counters undefined condition.
2: Resets the battery timer.
3: Deletes all programs and all positions. (The same as the NW command.)
4: Resets the origin setting condition.

Explanation

(1) Resets alarm condition in alarm mode, switching servo from OFF to ON and causing the program to to its beginning.

(2) If any of the axes has exceeded its software limit, alarm cannot be reset.

(3) The outputs remain unchanged by resetting any alarms.

* This command can only be executed directly. It be used in a program.

Sample program (BASIC)

10 OPEN "COM1 :E83"AS#1 'Opens the RS-232C communication file from personal computer in BASIC.

20 PRINT #1,"MO 1000" 'Alarm occurs because of wrong value.

30 PRINT #1,"MO 1000" 'Cancels the alarm.

40 END 'Ends program.

RV-M2

Function:

Resets the program and error condition.

Input Format

RS

Explanation

(1) Resets an error in error mode I (hardware error), switching servo from OFF to ON and causing the to return to its beginning. If any of the axes has exceeded its software limit, the alarm cannot be reset.

(2) Also resets an error condition in error mode II (software error), causing the error to be reset and the program to return to its beginning.

(3) The counter values and outputs remain unchanged resetting any error.

* This command can only be executed directly. It

be used in a program.

Sample program (BASIC)

10 LPRINT "MO 1000" 'Causes error mode II

'(defined value greater than

20 LPRINT "RS" 'Resets error

RV-M1

Function:

Resets the program and error condition.

Input Format

RS

Explanation

(1) this command resets the program causing it to to its beginning.

(2) Also resets an error condition in error mode II (software error), causing the error to be reset and the program to return to its beginning. Note, however, that error mode I (hardware error) cannot be reset by this command. To reset error mode I, turn power OFF.

(3) The counter values and outputs remain unchanged resetting any error.

* This command can only be executed directly. It be used in a program.

Sample program (BASIC)

10 LPRINT "MO 1000" 'Causes error mode II

'(defined value greater than

20 LPRINT "RS" 'Resets error

RT (Return)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Completes a subroutine and returns to the main

Input Format

RT [<Line number>]

Term

<Line number>

Specify the line number to jump.
1 <= line number <= 9999
(If omitted, returns to the next line of the command.)

Explanation

(1) Completes the subroutine called by the GS and returns to the main program.

(2) Alarm occurs if the corresponding "GS" command is not specified.

(3) If you specify the line number, the program jumps the specified line number after returning to the main routine.

Sample program

see the GS command.

RV-M2, RV-M1

Function:

Completes a subroutine and returns to the main

Input Format

RT

Explanation

(1) Completes the subroutine called by the GS

and returns to the main program.

(2) Alarm occurs if the corresponding "GS" command is not specified.

Sample program

see the GS command.

COSIMIR® · Movemaster Command

SC (Set Counter)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

A specified value is set in the specified counter or character string.

Input Format

SC <counter number/character string number>,
set value/character string set value>]

Term

<Counter number>

Specify the number of counter setting.
1 <= counter number <= 99

<Character string number>

Specify character string number in value which "$" is added to the head.
$1 <= character string number <= $99

<Counter set value>

Specify the value of counter setting.(0 for default)
-32768 <= set value (decimal) <= 32767
& 8000 <= set value (hexadecimal) <= &

<Character string set value>

Specify the character string to be set.
Usable characters:numerals (0 to 9),
alphabetic characters(A to Z),
symbols (!@#, etc.)
Unusable characters: "
Number of characters:
Within 120 characters including line number
and SC command.

Explanation

<When counter number is specified>

(1) All counters are factory-set to zero.

(2) Used to count the number of workpieces and job sequences and to set the number of grid points in the pallet.

(3) The contents of the counter can be changed, compared or read by the relevant command. (Refer to the IC, INP, DC, CP, CR, CL, AN, OR and XO commands.)

(4) The counter set value remains unchanged when the RS, NW or ED command is executed.

(5) The contents of the counter are battery backed the power is switched off.

<When a character string number is specified>

(1)Enclose the set character string with double quotations ("").

Example: When setting the character string ABC, set "ABC".

(2) If the set character string is omitted, the details of the character string number will be blank. Thus, the details of the character string number can be deleted.

(3) Operation, comparison and reading of the character string are possible with the related commands. (Refer the CP, CR, CL, EQ, NE, LG, SM, INP commands.)

(4) The value of the set character string will not even if the RS, NW or ED command is executed. The value will be held by the battery even when the power turned OFF.

Sample program (MOVEMASTER Command)

10 SC 21,10 'Set value 10 to counter 21.

20 IC21 'Add 1 to counter 21.

30 CP 21 'Set value of counter 21 to the internal register.

40 DR 'Outputs the value of the internal register through RS-232C port.

50 SC $5,"OK" 'Set character string "OK" in string number 5

60 CP $5 'Set details of character string number 5 in the

'character string register

70 EQ $10,200 'Jumps to line 200 if the data equals character string number 10.

RV-E-Robot (RV-E2, RV-E3J), RV-M2

Function:

A specified value is set in the specified counter.

Input Format

SC <counter number>, [<counter set value>]

Term

<Counter number>

Specify the number of counter setting.
1 <= counter number <= 99

<Counter set value>

Specify the value of counter setting.(0 for default)
-32768 <= set value (decimal) <= 32767
& 8000 <= set value (hexadecimal) <= & 7FFF

Explanation

(1) All counters are factory-set to zero.

(2) Used to count the number of workpieces and job sequences and to set the number of grid points in the

pallet.

(3) The contents of the counter can be changed, compared or read by the relevant command. (Refer to the IC, INP, DC, CP, CR, CL, AN, OR and XO commands.)

(4) The counter set value remains unchanged when the RS, NW or ED command is executed.

(5) The contents of the counter are battery backed the power is switched off.

Sample program (MOVEMASTER Command)

10 SC 21,10 'Set value 10 to counter 21.

20 IC 21 'Add 1 to counter 21.

30 CP 21 'Set value of counter 21 to the internal register.

40 DR 'Outputs the value of the internal register through RS-232C port.

50 ED 'End of program

RV-M1

Function:

A specified value is set in the specified counter.

Input Format

SC <counter number>, [<counter set value>]

Term

<Counter number>

Specify the number of counter setting.
1 <= counter number <= 99

<Counter set value>

Specify the value of counter setting.(0 for default)
-32768 <= set value (decimal) <= 32767
& 8000 <= set value (hexadecimal) <= &

Explanation

(1) All counters are factory-set to zero.

(2) Used to count the number of workpieces and job sequences and to set the number of grid points in the pallet.

(3) The contents of the counter can be changed, compared or read by the relevant command. (Refer to the IC, INP, DC, CP, CR, CL, AN, OR and XO commands.)

(4) The counter set value remains unchanged when the RS, NW or ED command is executed.

Sample program (MOVEMASTER Command)

10 SC 21,10 'Set value 10 to counter 21.

20 IC21 'Add 1 to counter 21.

30 CP 21 'Set value of counter 21 to the internal register.

40 DR 'Outputs the value of the internal register through RS-232C port.

50 ED 'End of program

COSIMIR® · Movemaster Command

SD (Speed Define)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Defines the moving velocity, first order time constant, acceleration/deceleration time, and continuous path setting.

Input Format

SD <moving speed> [, <first order time constant> [, <acceleration time>, <deceleration time> [, <CNT setting>]]]

Term

<Moving speed>

Set moving speed at linear or
circular interpolation.
0.01 <= moving speed <= 650.00
(mm/sec)

<First order time constant>

1 <= first order time constant <=
300 (millisecond)

<Acceleration time>

Set acceleration time to the
maximum speed.
0 <= acceleration time <= 2000
(millisecond)

<Deceleration time>

Set deceleration time from the
maximum speed.
0 <= deceleration time <= 2000
(millisecond)

<CNT setting>

Specify the enable or disable
of continuous path mode.
0: Disable 1: Enable

Explanation

(1) The least input increment of the moving speed is mm/second or 0.01 degree/second (e.g., specify 20.05 for 20.05 mm/second).

The least input increment of the first order time is 1 millisecond.

(2) Allows the moving speed (or angular speed) of the of hand for linear or circular interpolation to be in smaller increments than the SP command.

(3) Setting a large value to the first order time makes the robot operation slower and smoother.

(4) The speed set by the SD command is effective until new value is set by the SD or SP command. If you omit

the first order time constant and the acceleration/deceleration time, the predefined values continue to be effective.

(5) During linear or circular interpolation, a certain moving speed of the SD command may cause alarm in excess of the maximum speed of the corresponding
In this case, set the speed to a lower value.

(6) At power on, the moving speed is initialized to SP
H (63.3 mm/second).

(7) By enabling the CNT setting, the robot moves continuously without acceleration and deceleration until the SD or SP command disables the CNT setting. (Path motion) However, the robot accelerates and decelerates at a starting and at a stopping point as well as when a timer or an input command is executed during the path motion.

(8) The acceleration time is the maximum time for the robot to reach to the maximum speed. Accordingly,
the moving speed does not reach to the maximum
the actual acceleration time becomes smaller than the specified value. (The situation is the same for the deceleration time.)

(9) As the acceleration and deceleration distance required for movement are preset, when the specified speed and acceleration/deceleration are set, if the movement distance is small, the set speed may not be reached.

(10) If the acceleration/deceleration time is less than 200msec, an overspeed or overload alarm may occur. Depending on the load conditions, the life of the mechanical parts may be shortened, so keep the time at 200msec or more when possible.

CAUTION

CNT is valid in the default state. If the motion speed, acceleration or deceleration is changed in this state, the robot motion locus will change, so take care to prevent collisions with the peripheral devices. If the robot might collide with the peripheral devices, invalidate CNT or shorten the acceleration/deceleration time.

Sample program (MOVEMASTER Command)

```
10 SP 15        'Set the moving speed to 15.
```

```
20 MS 1     'Moves to position 1 by linear interpolation.
(SP 15)

30 SD 100   'Set the moving speed to 100mm/sec.

40 MS 2     'Moves to position 2 by linear interpolation.
(100 mm/sec)

50 MO 3     'Moves to position 3 by joint interpolation.
(SP 15)

60 MS 4     'Moves to position 4 by linear interpolation.
(100 mm/sec)

70 ED       'Ends program.
```

RV-M2

Function:

Defines the moving velocity and first order time of the end of the hand for linear interpolation.

Input Format

SD <moving speed> [, <first order time constant>]

Term

<Moving speed>

Set moving speed at linear or circular interpolation.
0 <= moving speed <= 500.0 (mm/sec)

<First order time constant>

1 <= first order time constant <= 300 (millisecond)
(Default: 80 milliseconds)

Explanation

(1) The least input increment of the moving speed is mm/second or 0.1 degree/second (e.g., specify 20.5 20.5 mm/second).

The least input increment of the first order time is 1 millisecond.

(2) Allows the moving speed (or angular speed) of the of hand for linear or circular interpolation to be

in smaller increments than the SP command.

(3) Setting a large value to the first order time makes the robot operation slower and smoother. When the robot turns a corner by the MC command, the SD command causes the path to be nearer to the inside. When the first order time constant is not specified, it defaults to 80 msec.

(4) The speed set by the SD command is effective until new value is set by the SD or SP command. If you omit the first order time constant the predefined values continue to be effective.

(5) During linear or circular interpolation, a certain moving speed of the SD command may cause error II in excess of the maximum speed of the corresponding joint. In this case, set the speed to a lower value.

(6) The moving velocity of the articulated interpolation command is not determined by the value of the SD command but by that of the SP command.

(7) At power on, the moving speed is initialized to SP 8, H.

Sample program (MOVEMASTER Command)

```
10 SP 15     'Set the moving speed to 15.

20 MS 1      'Moves to position 1 by linear
(SP 15)

30 SD 100    'Set the moving speed to 100mm/sec.

40 MS 2      'Moves to position 2 by linear
(100 mm/sec)

50 MO 3      'Moves to position 3 by joint
(SP 15)

60 MS 4      'Moves to position 4 by linear
(100 mm/sec)

70 ED        'Ends program.
```

RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 series.

COSIMIR® · Movemaster Command

SF (Shift)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.), RV-M2

Function:

Adds each coordinate value of position (b) to each coordinate value of p
defines it again as a new position.

Input Format

SF <position number (a)>, <position number (b)>

Term

<Position number>

Specify the position number.
1 <= position number (a), (b) <= 999

Explanation

(1) The hand open or close state of position (a), as well as the structural
N/F), is not affected by the SF command.

(2) Alarm occurs if the position (a) and/or (b) have not been predefined

(3) Does not effect any robot motion.

Sample program (MOVEMASTER Command) 5-axis type

10 PD 20,0,0,20,0,0,0	'Set the location and the attitude of position 20.
20 HE 10	'Set the current position to position 10.
30 SF 10,20	'To the Z-coordinate of position 10, the
	'Z-coordinate of position 20 (20 mm) is added.
40 MO 10	'Moves to position 10.
50 ED	'Ends program.

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Adds each of the coordinate values of position (a) to corresponding coor position (b) and reenters the data.

Input Format

SF <position number (a)>, <position number (b)>

Term

<Position number>

Specify the position numbers needed for the addition.
1 <= position number (a), (b) <= 999

Explanation

(1) The table below shows how the existence/nonexistence of position d additional axis determines the contents of position number (a) after the executed.

		Pattern 1	Pattern 2	P
Position (a)	**Robot**	Exists	Exists	E
Position (a)	**Additional axis**	Doesn,t exist	Exists	E
Position (b)	**Robot**	Exists	Exists	E
Position (b)	**Additional axis**	Exists	Doesn,t exist	E
SF (a),(b)	**Robot**	(a) <- (a) + (b)	(a) <- (a) + (b)	(
SF (a),(b)	**Additional axis**	(a) <- (b)	(a) <- (a)	(

Sample program (MOVEMASTER Command) 6-axis type

```
10 PD 20,0,0,20,0,0,0       'Set position 20 (Without position data for the

20 HE 10                     'Set the current position to position 10.

30 SF 10,20                  'Move position 10 only by the amount of posit
                            '(Z direction 20mm)

                            'Re-register the coordinate values to the shifte
                            '(Corresponds to "Pattern 2„ in the table above

40 MO 10                     'Moves to position 10 (Z direction 20 mm).

50 ED                        'Ends program.
```

RV-M1

Function:

Adds each coordinate value of position (b) to each coordinate value of p
defines it again as a new position.

Input Format

SF <position number (a)>, <position number (b)>

Term

<Position number>

Specify the position number.
1 <= position number (a), (b) <= 629

Explanation

(1) The hand open or close state of position (a), as well as the structural
N/F), is not affected by the SF command.

(2) Alarm occurs if the position (a) and/or (b) have not been predefined

(3) Does not effect any robot motion.

Sample program (MOVEMASTER Command)

```
10 PD 20,0,0,20,0,0,0     'Set the location and the attitude of position 2(

20 HE 10                   'Set the current position to position 10.

30 SF 10,20                'To the Z-coordinate of position 10, the

                          'Z-coordinate of position 20 (20 mm) is added.

40 MO 10                  'Moves to position 10.

50 ED                     'Ends program.
```


COSIMIR® · Movemaster Command

SL (Set Limit)

RV-M2 with additional axis

Function

Limits the moving range of the Locomotive.

Input Format

SL <Travel range (mm)>

Where, 0 < travel range (mm) <= 800, 1200, 2000 (mm)
The maximum value (800, 1200 or 2000mm) depends on the Locomotive unit used.

Sample Input

SL 600

Explanation

(1) The least input increment of the distance traveled 0.1mm. (Example: Specify 600.1 for 600.1mm.)

(2) Changing the parameter allows the operating range be limited by the software. Note that the operating is referenced from the origin.

(3) Error mode II occurs it the defined value is outside the moving range (800mm, 1200mm or 2000mm) of the Locomotive unit used.

Sample Program

10 LPRINT "100 NT"

20 LPRINT "110 SL 500"

30 LPRINT "120 ML 80"

40 LPRINT "130 GT 120"

* The above program example returns the robot to the origin, defines the Locomotive operating range and repeats an independent operation (80mm positive movement) of the Locomotive. Error mode II occurs the destination position exceeds the above operating range.

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function

No operation. Prepared to keep the compatibility with the MOVEMASTER M2 series. Only registration is

Input Format

SL

Explanation

This command is prepared for keeping the
with the MOVEMASTER M2 series and has no effect on
program.

COSIMIR® · Movemaster Command

SM (If Smaller)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

This compares the value of the internal register with a specified value. If smaller, the program will jump. The character string register and the numbers of characters a specified character string are compared. If the character string register is smaller, the program will jump.

Input Format

SM <compared value/character string number>,
<branching line number>

Term

<Compared value>

Specify the value compared with the register.
-32768 <= Compared value (decimal) <=

& 8000 <= Compared value (hexadecimal) <=
& 7FFF

<Character string number>

Specify character string number in value which "$" is added to the head.

$1 <= character string number <= $99

<Branching line number>

Specify the line number to which the jumps.
1 <= branching line number <= 9999

Explanation

<When counter number is specified>

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value is smaller than the compared value (i.e., when the condition is met), the program jumps to the specified line. Otherwise (i.e., when the condition is not met), the program continues sequence. Alarm occurs at a jump if the specified line number does not exist.

(3) A value can be loaded into the internal register by executing the input command (See ID) for the external input data or by executing the compare counter command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed "&".

<When character string number is specified>

(1)The conditions will jump depending on the data from an external source or the number of characters in specified character string.

(2) If the number of characters in the character string register is smaller than the number of characters in a specified character string (when the conditions are established), the program will jump to the specified number. If the number is larger (when conditions are established), the next line will be executed. If the specified line number is not registered, an alarm will occur when jumping.

(3) By executing an INP command, the data input from external device will be set in the character string register. The details of the character string number

be set by executing a CP command. Thus, when executing condition jumping, one of these commands must be executed first.

Sample program (MOVEMASTER Command)

10 ID 'Fetches the data from the external port.

20 SM 10,100 'If the input data is smaller than 10, jumps to line 100.

30 MS 1 'Moves to position 1 by linear interpolation.

40 ED 'Ends program.

100 MO 10 'Moves to position 10.

140 OPN 1,1 'Opens the RS-232C port.

150 INP 1, ,2 'Reads the data of character string register from the RS-232C port.

160 SM $5,200 'Jumps to line 200 if the data smaller than character string number 10.

:

200 MO 2 'Moves to position 2.

RV-E-Robot (RV-E2, RV-E3J)

Function:

This compares the value of the internal register with a specified value. If smaller, the program will jump.

Input Format

SM <compared value>, <branching line number>

Term

<Compared value>

Specify the value compared with the register.
-32768 <= Compared value (decimal) <=

& 8000 <= Compared value (hexadecimal) <= & 7FFF

<Branching line number>

Specify the line number to which the jumps.
1 <= branching line number <= 9999

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value is smaller than the compared value (i.e., when the condition is met), the program jumps to the specified line. Otherwise (i.e., when the condition is not met), the program continues sequence. Alarm occurs at a jump if the specified line number does not exist.

(3) A value can be loaded into the internal register by executing the input command (See ID) for the external input data or by executing the compare counter command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed "&".

Sample program (MOVEMASTER Command)

10 ID 'Fetches the data from the external port.

20 SM 10,100 'If the input data is smaller than 10, jumps to line 100.

30 MS 1 'Moves to position 1 by linear interpolation.

40 ED 'Ends program.

100 MO 10 'Moves to position 10.

RV-M2

Function:

This compares the value of the internal register with a specified value. If smaller, the program will jump.

Input Format

SM <compared value>, <branching line number>

Term

<Compared value>

Specify the value compared with the register.
-32768 <= Compared value (decimal) <=

& 8000 <= Compared value (hexadecimal) <=
& 7FFF

<Branching line number>

Specify the line number to which the jumps.
1 <= branching line number <= 3584

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) If the internal register value is smaller than the compared value (i.e., when the condition is met), the program jumps to the specified line. Otherwise (i.e., when the condition is not met), the program continues sequence. Error mode II occurs at a jump if the line number does not exist.

(3) A value can be loaded into the internal register by executing the input command (See ID and IN) for the external input data or by executing the compare command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4) The compared value may be defined either in or hexadecimal. A hexadecimal value must be headed "&".

Sample program (MOVEMASTER Command)

10 ID 'Fetches the data from the external input port.

20 SM 10,100 'If the input data is smaller than 10, jumps to line 100.

:

40 ED 'Ends program.

100 MO 10 'Moves to position 10.

RV-M1

Function:

This compares the value of the internal register with a specified value. If smaller, the program will jump.

Input Format

SM <compared value>, <branching line number>

Term

<Compared value>

Specify the value compared with the register.
-32768 <= Compared value (decimal) <=

& 8000 <= Compared value (hexadecimal) <=
& 7FFF

<Branching line number>

Specify the line number to which the jumps.
1 <= branching line number <= 2048

Explanation

(1)Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2)If the internal register value is smaller than the compared value (i.e., when the condition is met), the program jumps to the specified line. Otherwise (i.e., when the condition is not met), the program continues sequence. Error mode II occurs at a jump if the line number does not exist.

(3)A value can be loaded into the internal register by executing the input command (See ID and IN) for the external input data or by executing the compare command (See CP) for the counter data. Accordingly when you carry out conditional branching, need to execute either of the above commands beforehand.

(4)The compared value may be defined either in decimal or hexadecimal. A hexadecimal value must be headed by "&".

Sample program (MOVEMASTER Command)

10 ID 'Fetches the data from the external input port.

20 SM 10,100 'If the input data is smaller than 10, jumps to line 100.

:

40 ED 'Ends program.

100 MO 10 'Moves to position 10.

COSIMIR® · Movemaster Command

SP (Speed)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Sets the operating speed, acceleration or deceleration time and the continuous path setting.

Input Format

SP <speed level> [, <H/L> [, <CNT setting>]]

Term

<Speed level>

Set moving speed.
0 <= speed level <= 30

<H/L>

Set acceleration/deceleration level.

H: High acceleration/deceleration (Max. 0.2 second)
L: Low acceleration/deceleration (Max. 0.4 second)

<CNT setting>

Specify the enable/disable state of the continuous path mode.
0: Disable 1: Enable

Explanation

(1) Sets the operating speed in 31 steps and acceleration/deceleration time upon start and stop in 2 levels.

(2) The speed level is predetermined as a ratio to the maximum rpm of each joint for joint interpolation and a ratio to the maximum speed of the tip of hand (650 mm/second) for linear interpolation.

(3) The acceleration/deceleration time may be from among H or L. The acceleration time is the maximum time for the robot to reach to the maximum speed. Therefore, when the moving speed does not to the maximum speed, the actual acceleration time becomes smaller than the specified value.

(4) The acceleration/deceleration distance required for movement is predetermined according to the specified speed and the set speed may not be reached if the distance of travel is small.

(5) For linear interpolation, the tip of hand, by the tool command, is moved at constant speed. In case, alarm may result from any of the joints its maximum speed. If the motion of the position angle (6-axis type: A,B,C angle, 5-axis type: A,B angle) is greater than the motion of the distance (X, Y, Z), the robot moves in reference to the position angular speed.

The SD command allows the speed to be defined in smaller increments.

(6) Once set, the speed and acceleration/deceleration time remain valid until new values are set. The default values are SP 12, H. The most recent acceleration/deceleration time remains valid when it is not specified.

(7) By enabling the CNT setting, the robot moves

continuously without acceleration and deceleration until the SD or SP command disables the CNT setting. (Path motion) However, the robot accelerates and decelerates at a starting and at a stopping point as well as when a timer or a input command is executed during the path motion.

(8) The default state when the power is turned ON is set to SP12, H.

(9) The speed and acceleration/deceleration set once is valid until set again. If H/L is omitted, the previous setting value will continue to be valid. (If the SD command is executed, that setting value will be valid.

The relation between the speed level and the moving speed:

SP	**Joint interpolation [%]**	**Linear interpolation [mm/s]**
0	0,1	0,2
1	0,4	2,7
2	0,6	3,8
3	0,8	5,3
4	1,1	7,3
5	1,5	9,8
6	2,0	13,3
7	2,7	17,8
8	3,7	23,8
9	4,9	31,7
10	6,5	42,4
11	8,2	53,1
12	**9,7**	**63,3**
13	11,6	75,3
14	13,7	89,2
15	16,2	105,2
16	19,0	123,7
17	22,2	144,5
18	25,9	168,1
19	29,8	193,8
20	34,2	222,0
21	40,7	264,8
22	47,3	307,6
23	53,9	350,4
24	60,5	393,2

25	67,1	436,0
26	73,7	478,8
27	80,2	521,6
28	86,8	564,4
29	93,4	607,2
30	100,0	650,0

Note:

The robot moves in reference to the position angular speed in the case of the amount (6-axis type: A, B, C turning angle, 5-axis type: A,B turning angle) >= the amount (X, Y, Z distance) in linear and circular interpolation. (The angular speed in degree/second is equivalent to the distance speed in mm/second divided by the value 2.12.)

The **fat printed line** in the table above is the initial when the power is turned on.

CAUTION

CNT is valid in the default state. If the motion speed, acceleration or deceleration is changed in this state, the robot motion locus will change, so take care to prevent collisions with the peripheral devices. If the robot might collide with the peripheral devices, invalidate CNT or shorten the acceleration/deceleration time.

Sample program (MOVEMASTER Command)

```
10 SP 8      'Sets the moving speed to 8.
20 MO 5      'Moves to position 5 by joint interpolation.
30 SP 10     'Sets the moving speed to 10.
40 MS 7      'Moves to position 7 by linear interpolation.
50 ED        'Ends program.
```

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Sets the operating speed of the robot and the axis, and the acceleration/deceleration time.

Input Format

SP <speed level> [, <H/L> [, <CNT setting>]]

Term

<Speed level>

Set moving speed.
0 <= speed level <= 30

<H/L>

Set acceleration/deceleration level.
H: High acceleration/deceleration (Max. 0.2 second)
L: Low acceleration/deceleration (Max. 0.4 second)

<CNT setting>

Specify the enable/disable state for the CNT setting.
0: Disable 1 and higher: Enable

Explanation

(1) The specification of the speed level is expressed as ratio of the maximum speed of each axis, in the case joint interpolation, and as a ratio of the maximum (650mm/sec) of hand tip, in the case for linear interpolation. For the additional axis, it is expressed as ratio of the maximum speed of each axis.

(2) The specification for acceleration/deceleration is valid for the additional axis.

(3) By validating the CNT setting, you can omit acceleration/deceleration between consecutive movement commands until next invalidated. (Path motion) However, the robot accelerates and at start points and end points, as well as when like a timer or input wait signal is inserted between movement commands. Also, acceleration/deceleration executed even when the additional axis is moving.

Sample program (MOVEMASTER Command)

```
10 SP 8      'Sets the moving speed to 8.

20 MO 5      'Moves to position 5 by joint interpolation.

30 SP 10     'Sets the moving speed to 10.
```

40 MS 7 'Moves to position 7 by linear interpolation.

50 ED 'Ends program.

RV-M2

Function:

Sets the operating speed and the acceleration/deceleration time for the robot.

Input Format

SP <speed level> [, <H/L>]

Term

<Speed level>

Set moving speed.

0 <= speed level <= 20

<H/L>

Set acceleration/deceleration level.
H: High acceleration/deceleration (Max. 0.2 second)
L: Low acceleration/deceleration (Max. 0.4 second)

Explanation

(1) Sets the operating speed in 21 steps and acceleration/deceleration time upon start and stop in 2 levels.

(2) The speed level is predetermined as a ratio to the maximum rpm of each joint for joint interpolation and a ratio to the maximum speed of the tip of hand (222 mm/second) or to the maximum position angular (222 mm/second) for linear interpolation.

(3) The acceleration/deceleration time may be from among H, max. 0.2 seconds, or L, max. 0.4 (for articulated interpolation only). The acceleration time is the maximum time for the robot to reach to the maximum speed. Therefore, when the moving speed not reach to the maximum speed, the actual time becomes smaller than the specified value.

(4) The acceleration/deceleration distance required for

movement is predetermined according to the specified speed and the set speed may not be reached if the distance of travel is small.

(5) For linear interpolation, the tip of hand, determined by the tool command, is moved at constant speed. In case, error mode II may result from any of the joints exceeding its maximum speed. If the motion of the position angle (wrist pitch, roll) is greater than the motion of the distance (X, Y, Z), the robot moves in reference to the position angular speed. (The speed of the end of the hand is made lower.)

The SD command allows the speed to be defined in smaller increments.

(6) Once set, the speed and acceleration/deceleration time remain valid until new values are set. The default values are SP 8, H. The most recent acceleration/deceleration time remains valid when it is not specified.

(7) If the speed parameter is not specified, it defaults 0.

Relationship between speed parameters and velocity:

SP	**Joint interpolation [%]**	**Velocity [mm/s]**
0	0,1	0,2
1	1,2	2,7
2	1,7	3,8
3	2,4	5,3
4	3,3	7,3
5	4,4	9,8
6	6,0	13,3
7	8,0	17,8
8	**10,7**	**23,8**
9	14,3	31,7
10	19,0	42,2
11	23,9	53,1
12	28,5	63,3
13	33,9	75,3
14	40,2	89,2
15	47,4	105,2
16	55,7	123,7
17	65,1	144,5
18	75,7	168,1

20	100,0	222,0

Note:

The robot moves according to the angular velocity (°/sec.) if the motion of the position angle (wrist roll) >= that of the distance (X, Y, Z)

The **fat printed line** in the table above is the initial when the power is turned on.

Sample program (MOVEMASTER Command)

```
10 SP 8     'Sets the moving speed to 8.

20 MO 5     'Moves to position 5 by joint interpolation.

30 SP 10    'Sets the moving speed to 10.

40 MS 7     'Moves to position 7 by linear interpolation.

50 ED       'Ends program.
```

RV-M1

Function:

Sets the operating speed and the acceleration/deceleration time for the robot.

Input Format

SP <speed level> [, <H/L>]

Term

<Speed level>

Set moving speed.
0 <= speed level <= 9

<H/L>

Set acceleration/deceleration level.
H: High acceleration/deceleration
L: Low acceleration/deceleration

Explanation

(1) This command sets the operating velocity and the acceleration/deceleration time upon starting and

stopping. The velocity is variable in 10 steps, 9 being maximum speed and 0 the minimum. Acceleration/deceleration time may be selected from among H or L. The acceleration time is 0.35 seconds for and 0.5 seconds for L, while deceleration time is 0.4 seconds for H and 0.6 seconds for L. When H is selected, the acceleration and deceleration are constant from SP0 to SP9. When L is selected, the acceleration and deceleration times are constant from SP0 to SP9.

(2) When two or more axes of motion are involved, this command sets the operating velocity of the joint having the greatest number of motor pulses.

(4) The acceleration/deceleration distance required for movement is predetermined according to the specified speed and the set speed may not be reached if the distance of travel is small.

(5) Error mode I may be caused if a high speed and H time are set to effect a backward motion or when the robots load capacity is large. In such cases, set a low speed and L time.

(6) Once set, the speed and acceleration/deceleration time remain valid until new values are set. The default values are SP 4, L. The most recent acceleration/deceleration time remains valid when it is not specified.

(7) If the speed parameter is not specified, it defaults to 0.

Sample program (MOVEMASTER Command)

10 SP 3	'Sets the moving speed to 8.
20 MO 5	'Moves to position 5.
30 SP 6, L	'Sets the moving speed to 10 and time to
40 MO 7	'Moves to position 7.
50 ED	'Ends program.

COSIMIR® · Movemaster Command

STR * (Step Read)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Reads the contents of the specified step number, or stopping step number. (Using RS-232C)

Input Format

STR [<step number>]

Term

<Step number>

Specify the step number reading.
0 <= step number <= 9999

Explanation

(1) Outputs the contents of the specified step number, the stopping step number through the RS-232C port.

(2) The output format is ASCII coded as follows;

<In the case of teaching playback method>

: (colon), program command

<In the case of Movemaster command method>

Line number, program command

(3) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(4) If you specify the undefined step number, the hexadecimal 0D is returned over.

(5) If the step number is not specified or zero is specified, the current stopping line number is read.

(6) In the above case, the command STR allows you to confirm the step number by a personal computer when alarm occurs.

* This command can only be executed directly. It

be used in a program.

Sample program (BASIC) 6-axis type

10 OPEN "COM1:E83"AS#1 'Opens the RS-232C communication file

'from the personal computer in BASIC.

20 INPUT "Reading step is " : J $ 'Enters the step that you want to read.

30 PRINT #1,"STR"+J$ 'Transmits "STR" + number" to the controller.

40 LINE INPUT #1,A$ 'Saves the received to A$.

50 PRINT A$ 'Displays the data on screen.

60 END 'Ends program.

RUN 'Run the BASIC

Reading step is ? 2 'Enters the step

:MPC 0,227.85,371.92,581.68,-60.71,102.83,30.85,R,A,N,C

'Outputs the contents of the step.

Sample program (BASIC) 5-axis type

10 OPEN "COM1:E83"AS#1 'Opens the RS-232C communication file

'from the personal computer in BASIC.

20 INPUT "Reading step is " : J $ 'Enters the step number that you want to read.

30 PRINT #1,"STR"+J$ 'Transmits "STR" + "step number" to the controller.

40 LINE INPUT #1,A$ 'Saves the received data to A$.

50 PRINT A$ 'Displays the data

the screen.

60 END 'Ends program.

RUN 'Run the BASIC program.

Reading step is ? 2 'Enters the step number.

:MPC 0,227.85,371.92,581.68,102.83,30.85,R,A,C

'Outputs the contents of the step.

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

SUB (Subtract)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

Subtracts the direct value or the contents of the specified counter from the internal register value.

Input Format

SUB <Operation data>

Term

<Operation data>

Describes the data to be operated as a numeric value or counter No. with @.
-32768 <= numeric value (decimal) <= 32767
&8000 <= numeric value (hexadecimal) <=

&7FFF
@1 <= Counter No. <= @99

Explanation

(1) Designate the operation data setting as a numeric value or counter No.

When designating with a numeric value, use a decimal hexadecimal value. When using a hexadecimal, add a to the head of the operation data.

When setting with a counter No. add a "@" to the head the counter No.

The contents of the set counter No. will be used as the operation data.

(2) The operation results are stored in the internal register, so operation, comparison and reading, etc., of the operation results can be carried out with the commands.

(Refer to ADD, MUL, DIV, EQ, NE, LG, SM, CL and DR, OR commands)

Sample program (MOVEMASTER Command)

10 CP 1 ' Stores counter No. 1 value in internal register

20 SUB @2 ' Subtracts counter No. 2 value from register value

30 CL 3 ' Sets internal register value in counter No.

' (Counter No. 3 = counter No. 1 - counter No. 2)

40 CP 1 ' Stores counter No. 1 value in internal register

50 SUB 5 ' Subtracts 15 from internal register value

60 CL 4 ' Sets internal register value in counter No.

' (Counter No. 4 = counter No. 1-5)

RV-E-Robot (RV-E2, RV-E3J), RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-E, RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

TB (Test Bit)

RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Causes a jump to occur in accordance with the bit value in the internal register.

Input Format

TB [<+/->] <bit number>, <branching line number>

Term

<+/->

Set the condition that compares bit.
+: The bit is ON.
-:The bit is OFF.

<Bit number>

Specify the bit number of the internal register.
0 <= bit number <= 15

<Branching line number>

Specify the line number to which the jump.
1 <= branching line number <= 9999

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) The program jumps to the specified line number if

specified bit in the internal register is on or off (i.e., when the condition is met). Otherwise (i.e., when the condition is not met), the program continues in sequence.

(3) A value can be loaded into the internal register by executing the input command (see ID) for the external input data or by executing the compare counter command (see CP) for the counter data. It is therefore necessary to execute either of the above commands beforehand so that a conditional jump can occur.

(4) Alarm occurs at a jump if the specified line number not predefined

Sample program (MOVEMASTER Command)

```
10 ID           'Fetches data from the external input port.

20 TB +1,100    'If the bit number 1 of input data is ON,

                'then jumps to line number 100.

30 MS 1         'Moves to position 1 by linear interpolation.

40 ED           'Ends program.

:

100 MO 10       'Moves to position 10.
```

RV-M2

Function:

Causes a jump to occur in accordance with the bit value in the internal register.

Input Format

TB [<+/->] <bit number>, <branching line number>

Term

<+/->

Set the condition that compares bit.
+: The bit is ON.
-:The bit is OFF.

<Bit number>

Specify the bit number of the internal register.
0 <= bit number <= 17

<Branching line number>

Specify the line number to which the jump.
1 <= branching line number <= 3584

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) The program jumps to the specified line number if specified bit in the internal register is on or off (i.e., when the condition is met). Otherwise (i.e., when the condition is not met), the program continues in sequence.

(3) A value can be loaded into the internal register by executing the input command (see ID and IN) for the external input data or by executing the compare command (see CP) for the counter data. It is therefore necessary to execute either of the above commands beforehand so that a conditional jump can occur.

(4) Alarm occurs at a jump if the specified line number not predefined

Sample program (MOVEMASTER Command)

```
10 ID          'Fetches data from the external input port.

20 TB +1,100   'If the bit number 1 of input data is ON,

               'then jumps to line number 100.

30 MS 1        'Moves to position 1 by linear interpolation.

40 ED          'Ends program.

:

100 MO 10      'Moves to position 10.
```

RV-M1

Function:

Causes a jump to occur in accordance with the bit value in the internal register.

Input Format

TB [<+/->] <bit number>, <branching line number>

Term

<+/->

Set the condition that compares bit.
+: The bit is ON.
-:The bit is OFF.

<Bit number>

Specify the bit number of the internal register.
0 <= bit number <= 7 (15)
Figure in () is for type A16 or B16 I/O card

<Branching line number>

Specify the line number to which the jump.
1 <= branching line number <= 2048

Explanation

(1) Causes a jump to occur conditionally in accordance with the external input data or the internal counter value.

(2) The program jumps to the specified line number if specified bit in the internal register is on or off (i.e., when the condition is met). Otherwise (i.e., when the condition is not met), the program continues in sequence.

(3) A value can be loaded into the internal register by executing the input command (see ID and IN) for the external input data or by executing the compare command (see CP) for the counter data. It is therefore necessary to execute either of the above commands beforehand so that a conditional jump can occur.

(4) Alarm occurs at a jump if the specified line number

not predefined

Sample program (MOVEMASTER Command)

10 ID 'Fetches data from the external input port.

20 TB +1,100 'If the bit number 1 of input data is ON,

'then jumps to line number 100.

30 MS 1 'Moves to position 1 by linear interpolation.

40 ED 'Ends program.

:

100 MO 10 'Moves to position 10.

COSIMIR® · Movemaster Command

TBD (Test Bit Direct)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Causes a jump to occur in accordance with the bit value in the external input.

Input Format

TBD [<+/->] <input bit number>, <branching line

Term

<+/->

Set the condition that compares bit.
+: The bit is ON
-: The bit is OFF

<Input bit number>

Specify the bit number of general input.
0 <= input bit number <= 32767

<Branching line number>

Specify the line number to which the jumps.
1 <= branching line number <= 9999

Explanation

(1)Causes a jump to occur conditionally in accordance with the external input data directly.

(2) The program jumps to the specified line number if specified bit in the external input is on or off (i.e., the condition is met). Otherwise (i.e., when the condition is not met), the program continues in

(3)It is not necessary to execute the input command beforehand, and the internal register remains intact the execution of TBD command.

(4)Alarm occurs if the specified line number is not predefined.

Sample program (MOVEMASTER Command)

10 TBD +19,100 'If the bit 19 of the external input is ON,

'then jumps to line number 100.

20 MS 1 'Moves to position 1 by linear interpolation.

30 ED 'Ends program.

:

100 MO 10 'Moves to position 10.

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

TI (Timer)

Function:

Halts the motion for the specified length of time.

Input Format

TI <timer counter>

Term

<Timer counter>

Set the period of timer.
0 <= timer counter <= 32767(0.1 sec)

Explanation

(1)Causes the robot to halt its motion for the specified counter value X 0.1 second. (Max. 3276.7 seconds)

(2)Used to introduce a time delay before and after the hand is opened or closed for gripping a workpiece.

(3)The default value is zero.

Sample program (MOVEMASTER Command)

```
10 MO 1,O    'Moves to position 1.
20 TI 5      'Wait for 0.5 second.
30 GC        'Closes hand.
40 TI 10     'Wait for 1.0 second.
50 MO 2      'Moves to position 2.
60 ED        'Ends program.
```

COSIMIR® · Movemaster Command

TL (Tool)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Establishes the distance between the hand mounting surface and the tip of hand.

Input Format

TL [<tool length>]

Term

<Tool length>

Set the distance from the hand mounting surface to the tip of hand.
0 <= tool length <= 300.00 (mm) (0 for default)

Explanation

(1) The least input increment of the tool length is 0.01 mm (e.g., specify 200.05 for 200.05 mm).

(2) Once established, the tool length remains valid new value is set (battery backed when the power is switched off). When the tool length has been changed, the current position is also changed accordingly, which, however, does not involve any robot motion. (Initial length is 123 mm.)

(3) Since the point defined by the TL command is the basis for calculation of the current position, XYZ and commands involving the XYZ coordinates system, accurate tool length must be established according to tool being used.

(4) Whenever a program is to be run, the same tool length as that established during teaching must be set the beginning of the program.

(5) The current value can be read from the RS-232-C by executing the WT command.

CAUTION

When changing the tool length, accurately input the value so that the setting value is not mistaken.

CAUTION

If the original teaching point is moved to after the tool length, the robot movement posture will change. Take special care to prevent interference the periphery.

CAUTION

Even if the tool length is input accurately, the precision may not be achieved due to the actual dimension precision and the installation posture precision, etc.

CAUTION

When using the 5-axis robot, always describe "0" for the axes other than the Z axis. If other values are set, vibration or an alarm may occur during the operation.

Sample program (MOVEMASTER Command)

10 TL 120 'Sets the tool length to 120 mm.

20 HE 1 'Define the current position to position

30 TL 100 'Changes the tool length to 100 mm.

40 MO 1 'Moves to position 1 advancing 20 mm in the tool direction.

50 ED 'Ends program.

RV-M2

Function:

Establishes the distance between the hand mounting surface and the tip of hand.

Input Format

TL [<tool length>]

Term

<Tool length>

Set the distance from the hand mounting surface to the tip of hand.
0 <= tool length <= 300.00 (mm) (0 for default)

Explanation

(1)The least input increment of the tool length is 0.1 (e.g., specify 200.5 for 200.5 mm).

(2)Once established, the tool length remains valid until new value is set (battery backed when the power is switched off). When the tool length has been changed, the current position is also changed accordingly, which, however, does not involve any robot motion. (Initial length is 123 mm.)

(3)Since the point defined by the TL command is the basis for calculation of the current position, XYZ and commands involving the XYZ coordinates system, accurate tool length must be established according to tool being used.

(4)Whenever a program is to be run, the same tool as that established during teaching must be set at the beginning of the program.

(5)The current value can be read from the RS-232-C by executing the WT command.

Sample program (MOVEMASTER Command)

10 TL 120 'Sets the tool length to 120 mm.

20 HE 1 'Define the current position to position

30 TL 100 'Changes the tool length to 100 mm.

40 MO 1 'Moves to position 1 advancing 20 mm in the tool direction.

50 ED 'Ends program.

RV-M1

Function:

Establishes the distance between the hand mounting surface and the tip of hand.

Input Format

TL [<tool length>]

Term

<Tool length>

Set the distance from the hand mounting surface to the tip of hand.
0 <= tool length <= 300.00 (mm) (0 for default)

Explanation

(1)The least input increment of the tool length is 0.1 (e.g., specify 200.5 for 200.5 mm).

(2)Once established, the tool length remains valid until new value is set (battery backed when the power is switched off). When the tool length has been changed, the current position is also changed accordingly, which, however, does not involve any robot motion. (Initial length is 107 mm.)

(3)Since the point defined by the TL command is the basis for calculation of the current position, XYZ and commands involving the XYZ coordinates system, accurate tool length must be established according to tool being used.

(4)Whenever a program is to be run, the same tool as that established during teaching must be set at the beginning of the program.

Sample program (MOVEMASTER Command)

10 TL 120 'Sets the tool length to 120 mm.

20 HE 1 'Define the current position to position 1.

30 TL 100 'Changes the tool length to 100 mm.

40 MO 1 'Moves to position 1 advancing 20 mm in tool direction.

50 ED 'Ends program.

COSIMIR® · Movemaster Command

TLM (Tool Matrix)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

The tool conversion is carried out.

Input Format

TLM <Tool length X>, <Tool length Y>, <Tool length Z>, <A rotation>, <B rotation>, <C rotation>

Term

<Tool length X>

Describe the X axis direction tool length at the tool coordinate system. Unit: mm

<Tool length Y>

Describe the Y axis direction tool length at the tool coordinate system. Unit: mm

<Tool length Z>

Describe the Z axis direction tool length at the tool coordinate system. Unit: mm

<A rotation>

Describe the X axis rotation angle at the tool coordinate system. Unit: degrees

<B rotation>

Describe the Y axis rotation angle at the tool coordinate system. Unit: degrees

<C rotation>

Describe the Z axis rotation angle at the tool coordinate system. Unit: degrees

Explanation

(1) The minimum setting unit for the tool length is 0.01mm or 0.01 degrees.

(2) The set tool length will keep the previous value until reset. This value is also held by the battery when the power is turned OFF.

(3) When the tool length is changed, the current will change accordingly. However, the robot will not operate according to this.

(4) The point on the tool coordinate system set with this command will be the control target for the position data operation, Cartesian system operation command and orthogonal jog operation. Thus, set an accurate value that matches the tool (hand, etc.) being used.

(5) Only the <Tool length Z> is valid for the 5-axis robot.

(6) The current setting value can be read from RS-232-C by executing the WTM command.

CAUTION

When changing the tool length, accurately input the value so that the setting value is not mistaken.

CAUTION

If the original teaching point is moved to after the tool length, the robot movement posture will change. Take special care to prevent interference the periphery.

CAUTION

Even if the tool length is input accurately, the precision may not be achieved due to the actual dimension precision and the installation posture precision, etc.

Sample program (MOVEMASTER Command)

10 TLM 0,0,100,0,0,0 'Change tool length to Z=100mm

20 HE 1 'Register current position in position 1

20 TLM -50,0,100,0,0,0 'Change tool length to X=-50mm

30 MS 1 'Move to position 1 (Move to position +50mm away

'from the current position in tool coordinate

'systems X axis direction.)

RV-E-Robot (RV-E2, RV-E3J), RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-E, RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

TR * (Transfer)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function: No operation

No operation. Prepared to keep the compatibility with the MOVEMASTER M2 series. Only registration is

Input Format

TR

Explanation

This command is prepared for keeping the with the MOVEMASTER M2 series and has no effect on program.

RV-M2, RV-M1

Function:

Transfers the program and position data stored in to the drive unit RAM.

Input Format

TR

Explanation

(1) This command causes the contents of the EPROM installed in the user socket inside the drive unit side to be transferred to the RAM. Once the command has been executed, the old program and position data in RAM are all cleared.

(2) The command execution indicator LED (EXECUTE) lights up instantaneously when the data is being transferred.

(3) When bit 4 of SW 1 located inside the drive unit side door is in the upper position (ON), the command also causes the Cartesian coordinate system reference position data in EPROM to be read into RAM. This of data does not take place when bit 4 is in the lower position (OFF) and the position data in RAM remains valid.

* This command can only be executed directly. It cannot be used in a program.

Sample program

10 LPRINT "TR"

COSIMIR® · Movemaster Command

VR (Version Read)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Reads the software version of the system ROM. (Using 232C)

Input Format

VR

Explanation

(1) Outputs the software version of the system ROM mounted in the controller through the RS-232C port.

(2) The output format is ASCII coded. (Example: "RV-E2 Ver. A1")

The denotation of the software version is the following.

(3)Because the terminator of the output data is carriage return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by a personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

Sample program (BASIC) for the RV-E2

```
10 OPEN "COM1 :E83"AS#1                'Opens the RS-232C communication file
                                       'from the personal computer in BASIC.
20 PRINT #1,"VR"                       'Transmits the command.
30 LINE INPUT #1,A$                    'Saves the received data to A$.
50 PRINT "Software version is " : A $; 'Displays the on the screen.
60 END                                 'Ends program.
RUN                                    'Run the
Software version is RV-E2 Ver. A1      'Outputs the version name.
```

RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

WH (Where)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Reads the coordinates of the current position and the open or close state of the hand. (Using RS-232C)

Input Format

WH

Explanation

(1)Causes the coordinates of the current position of the tip of hand, as determined by the tool length (see the command), and the hand open or close state to be through the RS-232C port.

(2)The output format is ASCII coded as indicated
The least output increment is 0.01 mm or 0.01 degree. (e.g., 20.01 for 20.01 mm)

Output format:

6-axis type: X, Y, Z coordinate value, A, B, C turning angle, R/L, A/B, N/F, O/C (structural flag)

5-axis type: X, Y, Z coordinate value, A, B turning R/L, A/B, O/C (structural flag)

(3)Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

Sample program (BASIC) 6-axis type

(1)-axis type

10OPEN "COM1 :E83";Opens the RS-232C communication file

;from the personal computer in BASIC.

20PRINT #1,"WH";Transmits the command "WH".

40LINE INPUT #1,A$;Saves the received data to A$.

50 PRINT "Current coordinates are " : A $;Displays the contents of A$ on the screen.

60END;Ends.

RUN;Run the program.

Current coordinates are +10.00,;Displays the value of current position.

+380.00,300.00,-70.00,50.00,+40.00, R, A, N, C

Sample program (BASIC) 5-axis type

10 OPEN "COM1 :E83" 'Opens RS-232C communication file

'from the personal computer in BASIC.

20 PRINT #1,"WH" 'Transmits the command "WH".

40 LINE INPUT #1,A$ 'Saves the received data to A$.

50 PRINT "Current coordinates are " : A $ 'Displays the contents of A$ on the screen.

60 END 'Ends.

RUN 'Run the program.

Current coordinates are +10.00, 'Displays the value of the current position.

+380.00,300.00,+50.00,+40.00, R, A, C

RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Loads the open/close state of the hand, and the and coordinate values of a specified position.

Input Format

WH

Explanation

(1) The output configuration is as shown below, and is output in ASCII code entirely.

The least increment for the output is 0.01mm or

(Example: 20.01mm is displayed as 20.01)

Output configuration:

X,Y,Z A,B,C, First additional axis, Second additional R/L, A/B, N/F, O/C (**6-axis-type**)

X,Y,Z A,B, First additional axis, Second additional axis, R/L, A/B, O/C (**5-axis-type**)

(2) An example of an output configuration is as follows:

For 6-axis-type

- First additional axis and Second additional axis are existent or undefined:

X, Y, Z, A, B, C, R, A, N, O

- First additional axis exists, Second additional axis is non-existent or undefined:

X, Y, Z, A, B, C, First additional axis, +0.00, R, A, N, O

- First additional axis and Second additional axis exist:

X, Y, Z, A, B, C, First additional axis, Second additional axis, R, A, N, O

- First additional axis is non-existent or undefined, Second additional axis exists:

X, Y, Z, A, B, C, +0.00, Second additional axis, R, A, N,

For 5-axis-type

- First additional axis and Second additional axis are existent or undefined:

X, Y, Z, A, B, R, A, O

- First additional axis exists, Second additional axis is non-existent or undefined:

X, Y, Z, A, B, First additional axis, +0.00, R, A, O

- First additional axis and Second additional axis exist:

X, Y, Z, A, B, First additional axis, Second additional R, A, O

- First additional axis is non-existent or undefined, Second additional axis exists:

X, Y, Z, A, B, +0.00, Second additional axis, R, A, O

(3) If there is a movement command immediately preceding this command, the position will be along the path of the movement, since this command will be executed even if the additional axis is still moving.

(4) If the next command is specified to be executed before the additional axis has completed operation, the WH command is executed, a position along the path of the additional axis may be loaded. (Refer to the WRM command)

Related language:

WRM

Sample program (BASIC) 6-axis type

10 OPEN "COM1 :E83" 'Opens RS-232C communication file

'from the personal computer in BASIC.

20 PRINT #1,"WH" 'Transmits the command "WH".

40 LINE INPUT #1,A$ 'Saves the received data to A$.

50 PRINT "Current coordinates are " : A $ 'Displays the contents of A$ on the screen.

60 END 'Ends.

RUN 'Run the program.

Current coordinates are +10.00, 'Displays the value of the current position.

+380.00,300.00,-70.00,50.00,+40.00, +100, 200, R, A, C

Sample program (BASIC) 5-axis type

10 OPEN "COM1 :E83" 'Opens RS-232C communication file

'from the personal computer in BASIC.

20 PRINT #1,"WH" 'Transmits the command "WH".

40 LINE INPUT #1,A$ 'Saves the received data to A$.

50 PRINT "Current coordinates are " : A $ 'Displays the contents of A$ on the screen.

60 END 'Ends.

RUN 'Run the program.

Current coordinates are +10.00, 'Displays the value of the current position.

+380.00,300.00,+70.00,+50.00, 100, 200, R, A, C

RV-M2, RV-M1

Function:

Reads the coordinates of the current position and the open or close state of the hand. (Using RS-232C)

Input Format

WH

Explanation

(1) Causes the coordinates of the current position of tip of hand, as determined by the tool length (see the command), and the hand open or close state to be through the RS-232C port.

(2)The output format is ASCII coded as indicated
The least output increment is 0.1 mm or 0.1 degree. (e.g., 20.1 for 20.1 mm)

Output format:

5-axis type: X, Y, Z coordinate value, pitch angle, roll angle, O/C (structural flag)

(3)Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

Sample program (BASIC)

10 OPEN "COM1 :9600, E, 7, 2"AS#1 'Opens the 232C communication file

'from the personal computer in BASIC.

20 PRINT #1,"WH" 'Transmits command "WH".

40 LINE INPUT #1,A$ 'Saves the received data to A$.

50 PRINT "Current coordinates are " : A $ 'Displays the contents of A$ on the screen.

60 END 'Ends.

RUN 'Run the program.

Current coordinates are +10.00, 'Displays value of the current position.

+380.00,300.00,+50.00,+40.00, C

RV-M2 with additional axis

Function:

Reads the coordinates of the current position and the open or close state of the hand. (Using RS-232C)

Input Format

WH

Explanation

(1) This command causes the coordinates of the position of the hand end, as determined by the tool length (see the TL command) currently being to be output from the RS232C port.

(2) The data is output in ASCII as indicated below. The least output increment is 0.1 mm or 0.1 degree. (e.g., 20.1 for 20.1 mm)

Output format:

X, Y, Z coordinate value, pitch angle, roll angle, Locomotive coordinate, O or C 0D (Hex)

(3) The delimiter of the data is a comma (,: Hex. 2C) the terminator is a carriage return (CR, Hex. 0D). If the data is to be received by a personal computer, it is necessary to handle the entire data string up to hex. The BASIC equivalent to this is the LINE INPUT# statement.

Sample Program (BASIC)

10 OPEN "COM1 :9600, E, 7, 2"AS#1 'Opens the 232C communication file

'from the personal computer in BASIC.

20 PRINT #1,"WH" 'Transmits command "WH".

40 LINE INPUT #1,A$ 'Saves the received data to A$.

50 PRINT "Current coordinates are " ; A $ 'Displays the contents of A$ on the screen.

60 END 'Ends.

RUN 'Run the program.

Current coordinates are +10.00, 'Displays value of the current position.

+380.00,300.00,+50.00,+40.00, 200, C

COSIMIR® · Movemaster Command

WR * (Write)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function

No operation. Prepared to keep the compatibility with the MOVEMASTER M2 series. Only registration is

Input Format

WR

Explanation

This command is prepared for keeping the
with the MOVEMASTER M2 series and has no effect on program.

RV-M2

Function:

Writes the generated program and position data into EPROM.

Input Format

WR

Explanation

(1) This command causes the program and position data generated in the drive unit RAM to be written into the EPROM installed in the user socket inside the drive unit side door. The destination EPROM must have previously been erased before executing the command.

(2) While the data is being written into the EPROM, the command execution indicator LED (EXECUTE) on the unit front panel stays lit. It is switched off as soon as data has been written (which takes about 3 minutes 40 seconds).

(3) Error mode II is caused if the EPROM has not been erased or a write error occurs.

* This command can only be executed directly. It
be used in a program.

Sample program

10 LPRINT "TR"

RV-M1

Function:

Writes the generated program and position data into EPROM.

Input Format

WR

Explanation

(1) This command causes the program and position data generated in the drive unit RAM to be written into the EPROM installed in the user socket inside the drive unit side door. The destination EPROM must have previously been erased before executing the command.

(2) While the data is being written into the EPROM, the command execution indicator LED (EXECUTE) on the unit front panel stays lit. It is switched off as soon as data has been written (which takes about 100 seconds).

(3) Error mode II is caused if the EPROM has not been erased or a write error occurs.

* This command can only be executed directly. It
be used in a program.

Sample program

10 LPRINT "TR"

COSIMIR® · Movemaster Command

WRM * (Wait Robot Motion)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.) with additional axis

Function:

Specifies the movement of the robot when the
axis is moving.

Input Format

WRM [<Movement specification>]

Term

<Movement specification>

Specifies execution/wait for the interpolation of the command when the additional axis is moving.
0: Wait (When omitted)
1: Execute

Explanation

(1) Once the movement specification is set to 1, thereafter, whenever interpolation begins towards a position that is defined only for the robot,
will start even though the additional axis is still moving

(2) Once the movement specification is set to 0, thereafter, whenever interpolation begins towards a position that is defined only for the robot, if the additional axis is still moving, interpolation will not until it has stopped.

(3) The initial status when the power is turned ON is movement specification: 0.

(4) If the next objective position for interpolation is a position that is defined for both the robot and the additional axis, even though the movement
is set to 1, the next interpolation will not be executed until the movement of the additional axis has stopped.

(5) If the Continue function is used for the movement the additional axis, set WRM to 0 (default value). If
is set to 1, the Continue function for the additional axis will not work.

Sample program (MOVEMASTER Command)

```
10 WRM 0          'During interpolation for the robot

                  'if the additional axis is moving, wait

20 MO 1           'Move to position 1 by joint
interpolation

                  '(additional axis also moves)

30 MO 2           'Move to position 2 by joint
interpolation
```

'for robot only and wait for movement

'of the additional axis stop

40 WRM 1 'If using interpolation for robot only,

'execute it even if the additional axis is still moving

50 MO 3 'Move to position 3 by joint

'(additional axis also moves)

60 MO 4 'Even if the additional axis is still moving,

'move to position 4 by joint for robot only.

70 MO 3 'Move to position 3 by joint

'(additional axis also moves)

80 OB +0 'Execute, even if the additional axis is still moving

RV-E-Robot (RV-E2, RV-E3J), RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-E, RV-M1 and RV-M2 series without additional axis.

COSIMIR® · Movemaster Command

WT (What Tool)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.)

Function:

Reads the tool length currently being established. RS-232C)

Input Format

WT

Explanation

(1) Causes the tool length currently being established the TL command) to be output through the RS-232C

(2) The data is output in ASCII coded decimal. The least output increment is 0.01 mm (e.g., 105.07 is displayed for 105.07 mm).

(3) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(4) All robot motions are based on the established tool length. If a wrong tool length has been defined, the may interfere with a surrounding object. When the tool length is unknown, therefore, check the tool length the WT command before starting the robot.

Sample program (BASIC)

```
10 OPEN "COM1 :E83"AS#1        'Opens the RS-232C
communication file

                               'from the personal
computer in BASIC.

20 PRINT #1,"WT"               'Transmits the command
"WT".

40 LINE INPUT #1,A$            'Saves the received data
to A $.

50 PRINT "TOOL=" A$            'Displays the contents of
A$ on the screen.

60 END                         'Ends.

RUN                            'Run the program.

TOOL=105.7                     'Outputs the tool length.
```

RV-M2

Function:

Reads the tool length currently being established. (Using RS-232C)

Input Format

WT

Explanation

(1) Causes the tool length currently being established the TL command) to be output through the RS-232C

(2) The data is output in ASCII coded decimal. The least output increment is 0.1 mm (e.g., 105.7 is displayed 105.7 mm).

(3) Because the terminator of the output data is return (Hex.0D), it is necessary to handle serial data strings up to hexadecimal 0D in receiving a message by personal computer. "LINE INPUT #" statement is equivalent to this in BASIC.

(4) All robot motions are based on the established tool length. If a wrong tool length has been defined, the may interfere with a surrounding object. When the tool length is unknown, therefore, check the tool length the WT command before starting the robot.

Sample program (BASIC)

```
10 OPEN "COM1 :9600, E, 7, 2"AS#1        'Opens the RS-
232C communication file

                                         'from the
computer in BASIC.

20 PRINT #1,"WT"                          'Transmits the
command "WT".

40 LINE INPUT #1,A$                       'Saves the
received data to A $.

50 PRINT "TOOL=" A$                       'Displays the
contents of A$ on the screen.

60 END                                   'Ends.

RUN                                      'Run the

TOOL=105.7                                'Outputs the
length.
```

RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-M1 series.

COSIMIR® · Movemaster Command

WTM (What Tool Matrix)

RV-1A, RV-2AJ, RV-EN-Robot (RV-E4N, RV-E5NJ, etc.)

Function:

The currently set tool length (X, Y, Z< A, B, C) is read using RS-232-C.

Input Format

WTM

Explanation

(1) This command outputs the current tool length set with the TLM command (refer to the TLM command) the RS-232-C port.

(2) The output format is a decimal ASCII format. The minimum output unit is 0.01mm.

(Example: When the data is 105.07mm, 105.07 will display.)

(3) The data end code (terminator) is the carriage (CR: hexadecimal OD), so when receiving the data with personal computer, etc., the series of data up to the hexadecimal OD must be handled as the data train. the general BASIC language, "LINE INPUT#" is equivalent to this.

(4) The robot movement is all carried out according to the tool length set at that time. If an incorrect tool length is set, the robot could interfere with peripheral objects. Thus, if the tool length is unknown, confirm

tool length with this command before operating the robot.

Sample program (BASIC)

10 OPEN "COM1 :E83"AS#1 'Open RS-232-C with personal computer BASIC.

20 PRINT #1,"WTM" 'Transmit command "WTM".

30 LINE INPUT #1,A$ 'Store received data in A$.

40 PRINT "TLM=";A$ 'Output contents of A$ screen.

50 END 'End.

RUN 'Execute program.

TLM=50.00,30.00.100.00,0,0,0 'Output (display) tool length.

RV-E-Robot (RV-E2, RV-E3J), RV-M2, RV-M1

Function: No operation

No operation. This command is not available for the MOVEMASTER RV-E, RV-M1 and RV-M2 series.

COSIMIR® · Movemaster Command

XO (Exclusive Or)

RV-1A, RV-2AJ, RV-E-Robot (RV-E2, RV-E4N, etc.), RV-M2

Function:

EXCLUSIVE ORs the specified data and the internal register data.

Input Format

XO <operation data>

Term

<Operation data>

Specify the data to be operated.
-32768 <= operation data (decimal) <= 32767
& 8000 <= operation data (hexadecimal) <= 7FFF
@ <= counter number <= @99

Explanation

(1) Specify the data to be operated in decimal or hexadecimal. Any hexadecimal value must be headed "&".

(2) The operation result is stored into the internal register and can be changed, compared or read by relevant commands. (See the EQ, NE, LG, SM, CL, DR, and OR commands)

(3) Execution of the XO command after the input command (ID or IN) allows the required bits of the parallel input data fetched from the external device to be flipped to their opposite settings.

Sample program (MOVEMASTER Command)

10 ID 'Fetches data from the external input port.

20 AN &000F 'Fetches only lower 4 bits.

30 XO &000F 'Flips data of 4 lower bits to their opposite settings.

40 CL 21 'Sets above data to counter 21.

50 EQ 10,200 'If the above data equals 10, then to line 200.

60 ED 'Ends program.

:

200 MO 99 'Moves to position 99.

RV-M1

Function: No operation

No operation. This command is not available for the

MOVEMASTER RV-M1 series.

COSIMIR® · Movemaster Command

Overview of the parameters for RV-E2 and RV-E3J

For parameters of additional axes configurations see the overview of additional parameters

Parameter	**Para-meter name**	**Array No.**	**Contents**	**Default value**	**Remarl**
Tool coordinates	XTL	6 real No.	Sets initial value of tool coordinate system (X, Y, Z, A, B, C). Only Z coordinate is changeable for 5-axis robot. Unit: mm, mm, mm, deg, deg, deg	0.0,0.0,123.0, 0.0,0.0,0.0	
Base coordinates	XBS	6 real No.	Sets initial value of base coordinate system (X, Y, Z, A, B, C). Only X, Y, Z coordinates are changeable for 5-axis robot. Unit: mm, mm, mm, deg, deg, deg	0.0, 0.0, 0.0, 0.0, 0.0, 0.0	
Perpendicular area limit	PAR	6 real No.	Sets overrun limits of XYZ coordinate system for minus and plus direction (-X, +X, -Y,	- - -10000,10000	Default setting not exc the lim
Joint area limit	JAR	12 real No.	Sets overrun limits of each joint coordinate system for minus and plus direction (-J1, +J1, -J2, +J2, -J3, +J3, -J4, +J4, -J5, +J5, -J6, +J6) Unit: deg.	RV-E3J: -160.0, -45.0, 135.0, 0.0, 135.0,	

				-160.0, -120.0, -200.0, 200.0 RV-E2: -160.0, -45, 135.00, 50.0, 170.00, -160.0, -120.0, -200.0, 200.0	
User-defined area	UAR	6 real No.	Defines the area where the robot outputs a	1.0, 0.0, 1.0, 0.0, 1.0, 0.0	Default setting out of t setting
Battery application warning time	ALB	1 integer	It sets the time period for battery warning alarm. (Warning alarm does not occur in the case of a negative number.) Unit: Time	8760	
Automatic program execution	ATP	1 string	Sets program name to be executed automatically when powered on.	" " (No setting)	
Continue Function	CTN	1 integer	Sets whether the robot continues to run from the last execution environment when powered on. (Execution step, Program inner variable, I/O status, etc.) 0: Disable, 1: Enable	0	
ON/OFF of buzzer	BZR	1 integer	Sets ON/OFF of buzzer sound. 0: OFF, 1: ON	1	
Arm length	ARM	6 real No.	It sets the arm length of the robot.	350.0, 250.0, 250.0, 85.0, 0.0, 0.0	

Automatic operation speed	SPI	1 integer	Sets initial level of automatic operation speed. (Value of Movemaster command SP command.)	12
External override	EOV	2 real No.	Sets initial override of automatic operation. (External override, Program override) Unit: %	100.0, 100.0
Jog speed control	JGJ	3 real No.	Sets jog and step operation speed for joint interpolation mode. (Inching, low speed, high speed) Unit: deg, %, %	0.1, 1.0, 20.0
Jog speed control	JGP	3 real No.	Sets jog and step operation speed for linear and circular interpolation mode. (Inching, low speed, high speed) Unit: mm, mm/s, mm/s	0.1, 1.5, 100.0
Continuous motion control	CNT	1 real No.	Sets whether robot accelerates and decelerates at taught position when moving. 0: Disable, 1: Enable	0
Accel. and decel. Period	ADL	2 real No.	Sets acceleration and deceleration periods for reaching to maximum speed. Unit: seconds	0.2, 0.2
Delay time constant	TSR	1 real No.	Sets first-order delay constant. Unit: milliseconds	20.0
Pulse width	PWI	1 real No.	Sets accuracy of positioning. Unit: pulse	10000

Priority order of control	CPR	5 integer	It sets the priority order of periphal devices. (Controller, special I/O, teaching box, standard RS-232C, optional RS-232C)	0, 10, 30, 20, 20	
Grip control	GCD	1 integer	Sets the hand open/close normal reverse status. The 1st hand normal/reverse, the 1st hand initial condition, 2nd hand normal/reverse, the 2nd hand initial condition. Normal/reverse: 0 means normal, 1 means reverse Initial state: 0-3	0, 1, 0, 1	See comma GC, GO
The origin point setting permission	HOE	1 integer	Sets enable/disable of origin point setting. 0: Disable, 1: Enable (Robot cannot execute program when set to 1	0	It beco effecti just aft setting
The user-defined origin point	UOG	6 real No.	Sets user-defined origin point. (W, S, E, T, P, R) Unit: deg.	RV-E3J: -160.0, -45.0, 0.0, 0.0, -120.0, - RV-E2: -160.0, -45.0, 50.0, -160.0, -120.0, -	
Joint order returning to origin point	UNG	6 real No.	Sets joint motion order to return to the origin point.	2, 1, 1, 1, 2, 2	It beco effecti just aft setting
Wrist angle coordinate selection	RCD	1 integer	Selects wrist angle coordinate system for control and display.	0	RV-E5N RV-E5N

			0: General angle (Maintains posture), 1: Joint angle (Varies posture)		RV-E5N only
Stop input type selection	INB	1 integer	Sets stop input for A type contact or B type 0: A type (Normally open), 1: B type (Normally closed)	0	
Parallel I/O input setting	IN1 IN2 IN3	20 strings 20 strings 20 strings	Sets function for input board. Sets function for first I/O interface board. Sets function for second I/O interface board. Sets function for third I/O interface board.	PI0,,,,PI1,,,, ,,,, ,,,, ,STA,STP ,RST	
Parallel I/O output setting	OT1 OT2 OT3	16 strings 16 strings 16 strings	Sets function for output board. Sets function for first I/O interface board. Sets function for second I/O interface board. Sets function for third I/O interface board.	,,,, ,,,, ,,,, ,RUN,WAI ,ERR	
Program mode	PST	1 integer	Sets whether the external program No. is read into the robot at startup. 0: No (Present program) 1: Yes (Appointed program)	0	
The mechanical stopper	MOG	6 real No.	It sets the mechanical stopper origin position. (W, S, E, tee, P, R)	RV-E3J: -168.0, -50.0, -5.0, 0.0, -123.0, 0.0	

RV-E2:

-168.0, -50.0,

47.0, -163.0,

-123.3, 0.0

COSIMIR® · Movemaster Command

Overview of the parameters for additional axes of E2 and RV-E3J

Parameter	**Para-meter name**	**Array No.**	**Contents**	**Default**	**Remark**
Servo amplifier type	AXAP	Integer 2	Sets the additional axis servo amplifier that will be used (First additional axis, Second additional axis) -1: Other 0: MR-H(Incremental method) Mitsubishi 1: MR-H(Absolute value method) Mitsubishi 2: MR-J Mitsubishi 3: SGD Yazkawa 4: NPSA-G NEC 5: MSD Matsushita Electric	0,0	

			Industrial. Ltd. For -1 (Other) setting, the options for the additional axis will not function.		
Servomotor type	AXMT	Integer 2	Sets the type of machine for the additional axis servomotor to be used. (First additional axis, Second additional axis) Unit: rpm	0,0	Note 1)
Encoder resolution	AXEN	Integer 2	Sets the multiplier for the encoder's pulse rate (1 to 32767) Unit: rpm	2048,2048	Note 2)
Motor rated speed	AXTR	Integer 2	Sets the servomotor's rated speed (1 to 32767) (First additional axis, Second additional axis) Unit: rpm	3000,3000	Note 2)
Motor max. speed	AXMR	Integer 2	Sets the servomotor's max. speed. (First additional axis, Second additional axis) Unit: rpm	4000,4000	Note 2)
Gear ratio	AXEG	Integer 2	Sets the servo amplifier's electronic gear ratio. (First additional axis, Second additional axis) 1: Servo amplifier electronic gear ratio 1:1 4: Servo amplifier electronic gear ratio 4:1	4,4	Note 2)
Command pulse frequency	AXCL	Integer 2	Sets the max. command pulse frequency from the additional axis interface board.	1,1	Note 2)

			(First additional axis, Second additional axis) 0: 400 kHz 1: 200 kHz		
Origin designation method	AXOM	Integer 2	Sets the origin method (First additional axis, Second additional axis) 0: Dog-type 1 1: Dog-type 2 2: Data set type	0,0	Note 3) Note 6)
Origin designation speed	AXOS	Integer 2	Sets the movement speed (1 - 100) for origin designation. (First additional axis, Second additional axis) Unit: %	10,10	Note 3)
Origin designation direction	AXOD	Integer 2	Sets the movement direction for origin designation. (First additional axis, Second additional axis) 0: Command value - direction 1: Command value + direction	0,0	Note 3)
Origin position	AXOP	Integer 2	Sets all the origin position's position data the additional axis (First additional axis, Second additional axis)	0.00, 0.00	
Initial origin designation execution existence	AXOG	Integer 2	Indicates whether the origin setting and the first origin designation have been executed after the additional axis system is structured.	0,0	Note 8)
Additional	AXUN	Integer	Sets all the command	0,0	

axis unit		2	units for the additional axis. (First additional axis, Second additional axis) 0: deg 1: mm		
Total speed ratio numerator	AXGN	Integer 2	Sets the numerator (1 to 32767) for the percentage of the load movement amount and motor speed (First additional axis, Second additional axis)	1,1	
Total speed ration denominator	AXGD	Integer 2	Sets the denominator (1 to 32767) for the percentage of the load movement amount and motor speed (First additional axis, Second additional axis)	10,10	
Additional axis operation range	AXLM	Real Number 4	Sets the operation range for all the additional Unit: Determined by	-80000.00, 80000.00, -80000.00, 80000.00	Note 4)
The + direction of the motor's direction	AXCD	Integer 2	Sets the direction of the motor's rotation when the command value (First additional axis, Second additional axis) 0: Motor turns in a clockwise (CW) direction 1: Motor turns in a counter- clockwise (CCW) direction	1,1	Note 5)
Additional axis jog speed	AXJR	Real Number 3	Sets the jog speed. (Fixed quantity, Low, High) Unit: deg., %,%	0.10, 1.00, 20.00	
Additional axis	AXAD	Real Number	Sets the acceleration / deceleration time (>= 0)	0.20. 0.20, 0.20, 0.20	

acceleration/ deceleration time		4	for stand alone additional axis operations. (First additional axis increases, First additional axis decreases, Second additional axis increases, Second additional axis decreases) Unit: sec.		
Brake timer time	AXBT	Integer 2	Sets the brake timer time (0 to 32767) for the Servo ON/OFF. (First additional axis, Second additional axis) Unit: msec.	300,300	Note 7)

Note 1: Please refer to section 7.2.2 "Servomotor type" in your manual.

Note 2: This parameter becomes valid when AXMT = -1.

Note 3: Refer to section 6.2 "ORIGIN SET (Origin setting) / ORIGIN (Origin in your manual.

Note 4: When setting the operation range for the servo amplifier, make it smal than that value. Set it so that the lower limit < upper limit.

Note 5: Viewed from the side of the servomotor, the clockwise is the CW direct and counterclockwise is the CCW direction.

Note 6: If you set the parameter AXAP and specify the incremental method for amplifier, and specify the data set method for the parameter AXOM, you will error.

Note 7: Refer to section 7.2.4 "Brake timer time" in your manual.

Note 8: If you change the additional axis system (change the amplifier or motor change the dog position, motor installation position, or origin designation met change the value to 0 before you execute the origin designation or the origin