Part 3

Fics-III Programming

Table of Contents

1: *Fics*-III Programming

This part describes ***Fics*-III** programming. Examples of display screens shown in this paragraph are for ***Fics*-RT1** terminal. Display screens for ***WinFics*** software and ***Fics*-TPC1** terminal have different layout, but the parameters and their setting process are the same. Before beginning programming, make sure thatsystemparamet ers defined in part 2 are correctly set.

***Fics*-III** program is a collection of step data, describing the sequence in automatic operation. A program is identified by a 3-digit program number, considered as program's name. There is no limit to the number of steps in a specific program, but the total number of programs in a system is limited to 999 and the total number of steps is limited to 2000 steps or 3000 steps, depending on the installed system memory. In either case, the following inequality should hold.

> Sum of total number of steps in each program + number of programs
>
> < = Maximum number of steps (2000 or 3000)

1-1: Main Program and Sub-Programs

Programs are classified as main program and as sub-programs. Program numbers range from 001-999. By default, numbers 100-999 are assigned to sub-programs. On how to assign sub-program numbers, refer to 6-5-1-1: Setting range of sub-programs, part 2 of this manual.

The program selected in **DATA** mode of the ***Fics*-RT1** programming menu, or the program after automatic execution is called "Current Program", and its program number is called "Current Program Number". Notice that:

- Number 001 always refers to main program even if it is assigned as sub-program number.
- Automatic execution always starts from the main program.
- A program which is to be CALLed from other programs should be a sub-program.
- Number 999 is a special program used for homing.
- Number 998 is a special program used for system task, which is automatically executed when the power is turned on.

1-2: Setting Program Number (Program Name)

Selecting the [MANU]-[DATA] menu, allows user to enter a program number for the program to be created.

- The program number that was used last time is automatically chosen. If this is the program to be edited, press <ENT> key.
- If the selected program number does not exist, it is registered as a new program. In this case, since there is no data for the program, 'END' will appear in step 0001.
- If the selected program exists, the program will be listed from step 0001 and the program can be edited.

-DATA- <ENT><CLR>
PGM=100
Enter PROGRAM Number

Setting Program Number

1-3: Setting Step Number

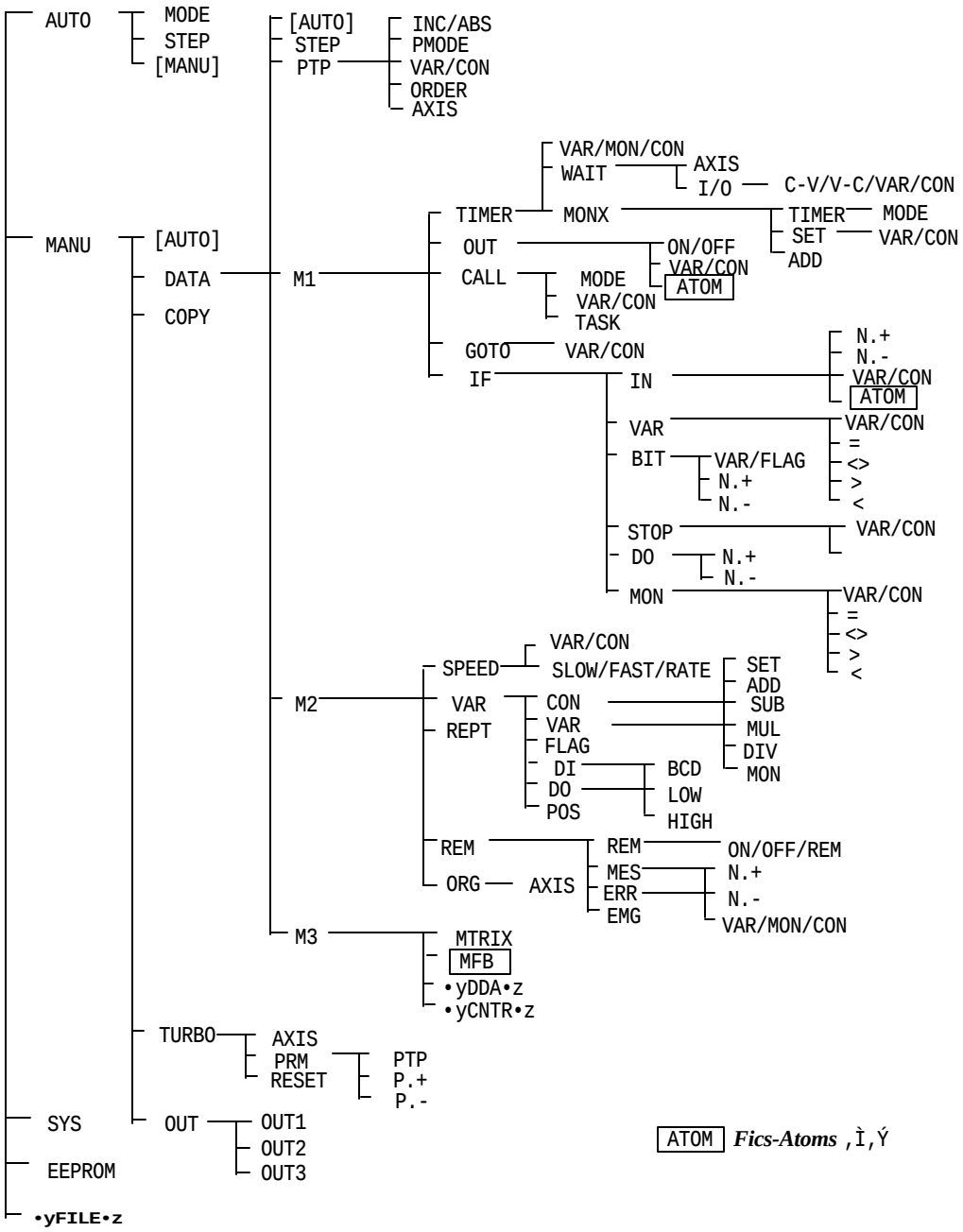
By selecting the [STEP] menu from **AUTO** or **DATA** mode, the screen appears as shown in the figure on the right, and step number can be specified. After setting, press <ENT> key to view the step data.

Step number is automatically switched to 0001 in **AUTO** mode after homing, but it will not change when mode is changed from **DATA** to **AUTO**.

-AUTO- <ENT><CLR>
PGM=100
N=0001
Enter STEP Number

Setting Step Number

1-4: Command Tree



2: Automatic Operation Mode

In automatic operation mode, programs written in **DATA** mode can be executed. There are three different program execution options in **AUTO** mode, The options in "**AUTO**" can be cyclically changed by pressing the [MODE] menu.

AUTO	executes continuously the program beginning with the displayed step number.
STEP	executes only the displayed step.
DEMO	executes the program from the displayed step ignoring I/O commands.

-AUTO-MODE STEP DATA
PGM=100 X=+xxxx.xxmm
N=0001 Y=-yyyy.yymm
ABS . [XYZ(W)]

Automatic Operation Menu

2-1: Switch to AUTO Mode from Other Modes

Selecting the **AUTO** menu leads to automatic operation mode. However, if the following conditions are not satisfied,

1. Homing is completed.
2. Homing is not completed, but all axes are set to home not needed.
3. Current program is the main program.

the system does not switch to **AUTO** mode. Instead, there will be an error message such as "ORIGIN ERROR" or "NOT MAIN PROGRAM".

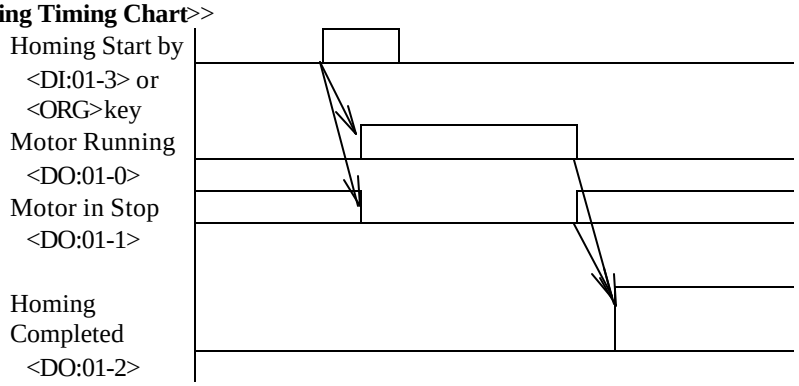
2-2: Switch to AUTO Mode from Homing

In the initial screen, if the <ORG> key is pressed, or external input <DI:01-3> is ON, homing starts with the following message shown on the screen.

Origin Processing

If current program is the main program, after homing is completed, the system is switched to automatic operation mode with step number being set to 0001.

<<**Homing Timing Chart**>>



*"Homing Completed" signal turns on once homing is completed. It switches to off when emergency stop switch is pressed, overrun error occurred, etc.

<<**All Axes Simultaneous Homing and Program Homing**>>

***Fics*-III** allows user to home all axes simultaneously, or define any homing sequences.The latter is called "program homing". On how to choose the homing sequence refer to 6-2-1: Setting homing parameter, part 2 of this manual.

If program homing is chosen, it is necessary to define the homing sequence in program 999. In this case, homing can be performed in any sequence (for example, home X axis first, then Y axis).

2-3: Operation Keys in AUTO mode (Homing, Step Change, Start Execution)

In addition to selections from a menu, it is possible to use the following keys.

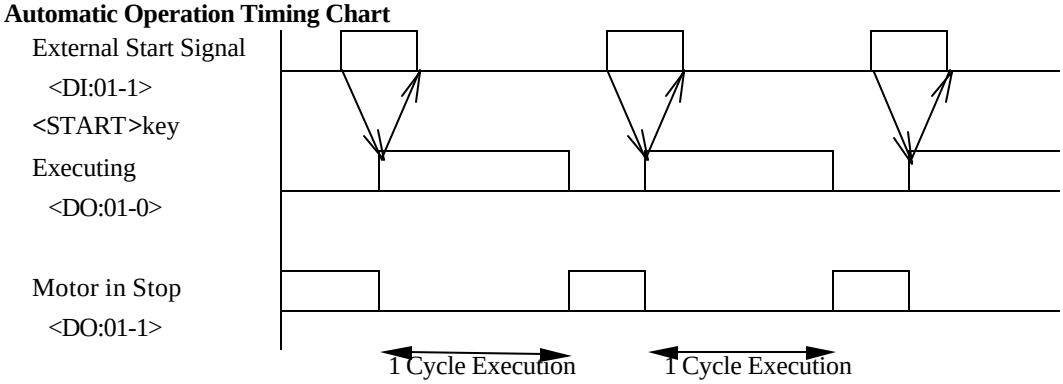
1. <ORG> key (or external input <DI:01-3>): Starts homing. After homing is completed, the "ORG" lamp on handheld terminal is turned on and system automatically switches to step 0001 of the current main program.
2. <ENT> key: Displays program data of the next step.
3. <+/-> key: Displays program data of the previous step.
4. <START> key (or external input <DI:01-1>): Starts automatic execution.

2-4: Automatic Execution of Programs

In automatic operation mode, if the <START> key is pressed or external input <DI:01-1> is ON, the program will start running from the displayed step.

If the <STOP> key is pressed, or external input <DI:01-2> is ON during automatic operation, execution of the program stops after completion of the current step. After stop, the next step is displayed. If the <START> key is pressed again at this point, the automatic execution starts again.

Automatic Operation Timing Chart



2-5: Continuation of Pallet

When **AUTO** mode is selected, the pallet program will continue to execute if it was not completed in the last automatic operation. If it is necessary to execute the pallet program from the beginning each time, insert the "PALLET CLEAR" command at the top of the program. Refer to 5-3-3: CALL command for details.

2-6: Error Handling

During automatic execution, if an error occurs, the program will stop execution and an error message is displayed. In case of emergency stop or overrun, motor stops within a period equal to half of the set deceleration time, and motor becomes free (motor position is not controlled). For error message, refer to 4-2: Alarm List, part 1 of this manual.

2-7: Error Stop in Each Unit of *Fics-Atoms* Series

In a control system using ***Fics*** series controllers, if any error occurs in any drive (***Atom*** series), all other drives will also stop operation. In multi-unit configuration, however, errors occurring in one unit do not affect the operation of other units. This is because stopping all units once error occurs in one unit could lead to a potential accident. From this point of view, user can set, with the aid of system variables, the scope of an emergency stop due to errors (error stop) to be valid for each unit or for the whole system.

When the error stop is set for each unit, units with no errors will not experience emergency stop, but will stop after the execution of the current step or after the completion of PTP. Automatic execution can be re-started when all units have stopped and errors have been reset.

When errors occurr, error messages are displayed immediately, but error reset is disabled untill all units have stopped. For this reason, in host mode, the controller (***Fics-Atoms***) sends R2 command indicating that an error had occurred, but the error clearance command R4 will not be received until all operation of units stops.

During the period when an error occurs and all related units stop, an emergency stop of other unrelated operating units can be activated by an emergency stop input.

In the host mode, when the MA/MR command is being executed, the message of successful operation will not be sent until errors are reset. Before errors are reset, if errors occur in other units, the controller (***Fics-Atoms*** series) will stop all axes in those units, and report an error message after errors are reset for those units.

In the host mode, when errors occur, the stop command (CAN) becomes invalid.

<< ***Fics*** System Variables >>

[SYS61] = 0: All axes stop when drive error occurs.

[SYS61] != 0: Only the unit stops when error occurs in any drive of the unit.

2-8: Error Reset

Errors are reset by pressing <CLR> key or by turning on the external input <DI:01-4>. Since the following errors are treated as emergency stops,

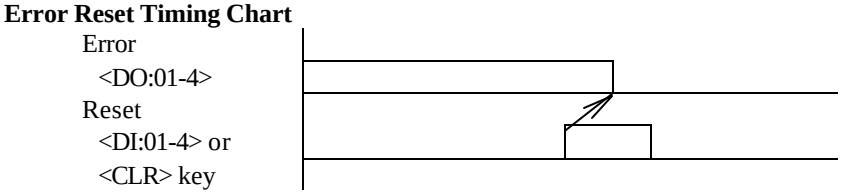
EMERGENCY MOTOR ERROR(08) MOTOR ERROR(14) COMM ERR(STNxx) (***Fics-Atoms***),

it is necessary to execute homing before an automatic operation. For other errors, automatic operation can be re-started.

For all errors, except TIME OVER error, when errors are reset in **AUTO/DEMO** mode, the current program is set to the last main program, and step number is set to 0001. In **STEP** mode, both program and step numbers are not changed.

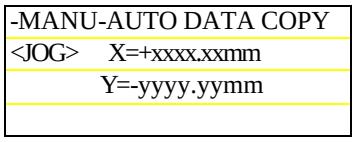
For TIME OVER error, neither program number nor step number changes in both **AUTO/DEMO** modes.

Error Reset Timing Chart



3: Manual Mode

In manual mode, JOG operation and ***Fics*-III** programming are allowed.
By selecting the [DATA] menu, mode is switched to programming mode. Details are stated below.

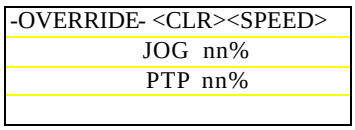


Manual Mode Menu

3-1: Keys Used in Manual Mode(Homing, JOG Operation)

In addition to the menu selection, the following keys can be used.

1. <ORG> key (or external input <DI:01-3>): Start homing.
2. JOG keys (<-X, X->, <-Y, Y->): Manual (JOG) operation.
3. <MODE> key: Assigns PTP and JOG speed override from 0 to 99%.



Setting the Rate of Override

- When the [AUTO] menu is selected, if homing is not completed, or current program is a sub-program, the initial screen is displayed.
- Programming by teaching requires the completion of homing.
- The rate of override is set to 100% (shown on screen as "00%") at power-on, which means PTP and JOG speeds are set as the speeds specified by system parameters. Once the rate is changed, it is valid untill power-off.

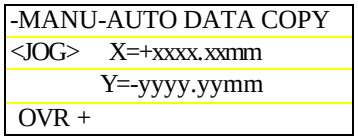
3-2: JOG Operation

JOG operation is carried out by pressing JOG keys (<-X,X->,<-Y,Y-> etc.). There are three different kinds of move for JOG operation. In all cases, coordinate values on the LCD screen indicate current position.

1. One-Shot Move: When any JOG key is pressed momentarily, the corrensponding axis moves a certain distance (One shot) specified by the "One Shot" parameter.
2. Low-Speed JOG Move: When any JOG key is pressed and held for less than 1.5 seconds, the corresponding axis keeps moving at the speed specified by SPEED parameter in the specified direction.
3. High-Speed JOG Move: When any JOG key is pressed and held for longer than 1.5 seconds, the corresponding axis moves at a higher speed in the specified direction. The speed is specified by SPEED [FAST] parameter.

3-3: Overrun in Manual Mode

In manual mode, if an overrun occurs, the system decelerates and stops, and the screen will be as shown in the figure on the right. Axis does not move in the overrun direction but can move in the opposite direction. The overrun in manual mode, however, is not an error.



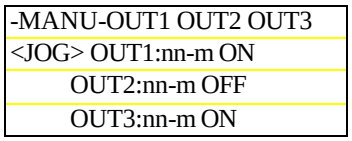
Manual Overrun Display

3-4: Axis Switching for Systems with More Than 4 Axes

The number of axes that can be displayed on the screen is limited, but by pressing JOG keys (<-X,X->,<-Y,Y-> etc.), the desired axis is displayed on the screen automatically, and JOG operation is enabled for the axis.

3-5: Manual ON/OFF Control of Outputs

By selecting the [OUT] menu in manual mode, the screen shown in the figure on the right is displayed. The "Channel-Bit" specified in 'nn-m' field is turned ON/OFF by pressing function keys <F1>-<F3> which corresponds to [OUT1]-[OUT3], respectively.



Manual ON/OFF of Outputs

- Here "nn" stands for channel number, "m" stands for bit number. When "nn" is "00", the bit number specified is ignored.
- Channel number and bit number remain valid until they are altered the next time.

3-6: ON/OFF Control of Outputs by Digital Inputs

In manual and programming modes, digital inputs (DI) can control digital outputs (DO). Inputs and outputs interact in the following manner:

ON/OFF:	Digital output turns ON/OFF if digital input turns ON/OFF.
ON:	Digital output ON when digital input ON.
OFF:	Digital output OFF when digital input ON.

A maximum of 15 digital inputs can be used this way. Refer to part 2 of this manual for details.

4: Programming Mode

Selecting the [MANU]-[DATA] menu enables user to edit programs for positioning and sequential control. In this mode, it is also possible to teach positions by JOG. Before editing program, it is necessary to assign program number.

4-1: Creating New Program

When new program is created, first step of the program is always "END". By selecting desired step number, program step can be created by entering data in input fields. To select desired field, move cursor by **"=>"** or "<=" keys. When cursor is blinking, the field is accepting input data. If all input fields are filled, press <ENT> key to end data input for the step. The step number will then automatically increase by 1. Repeating this process allows user to write a complete program.

4-2: Editing Programs

To modify step data, insert new steps, or change parameters in an existing program, choose one of the following actions to display desired step.

1. Select [STEP] menu.
2. Increase the step number by 1 using <ENT> key.
   if displayed data is the final step, step number returns to 0001
3. Decrease step number by 1 using <+/-> key.
   if displayed data is step 0001, step number goes to the final step

4-3: Inserting Step Data

In **DATA** mode, pressing <INS> key inserts a "NOP" command before the current step. Step numbers that follow are all automatically increased by 1.

Program steps before inserion ("*" indicates current step on display):

```
  N=0002    GOTO  STEP 0005
* N=0003    X=10.00mm  Y=10.00mm
  N=0004    IF  DI:01-1  IS  1
            THEN  GOTO  STEP 0003
  N=0005    CALL  100
```

Program steps after <INS> key is pressed:

```
  N=0002    GOTO  STEP 0006          ; step number is automatically changed
* N=0003    NOP                      ; insert a step
  N=0004    X=10.00mm  Y=10.00mm
  N=0005    IF  DI:01-1  IS  1
            THEN  GOTO  STEP 0004    ; step number is automatically changed
  N=0006    CALL  100
```

4-4: Automatic Correction of Branched Step Number on Insertion or Deletion

When inserting or deleting program steps, all step numbers that follow are automatically changed. All branch commands and conditional branch commands are automatically corrected to match the change. The above program is one of the examples for automatic correction of step numbers on insertion.

4-5: Deleting Program Steps

In **DATA** mode, and when field is not waiting for inputting data, pressing <DEL> key deletes current program step on display. To avoid malfunction, the following message is displayed for confirmation.

DEL OK ? YES DEL

- When this message is displayed, press <DEL> key to delete the program step. Press any keys other than <DEL> key to cancel the deletion.
- The number of steps to be deleted can be specified by numerical keys. By entering 3 digits decimal numbers, specified steps in program are deleted. To delete all steps following current step, enter a number greater than the total number of program steps.
- The last step (with "END" displayed) other than N 0001 cannot be deleted. When attempted to delete it, NUMBER ERROR occurs.

4-6: Deleting Programs

To delete a program, display step N=0001, enter a number greated than the total number of program steps , and press <ENT> key. **PGM=001 program cannot be deleted.**

5: Inputting and Editing Program Steps

The following table lists programming commands for positioning and I/O control. The context in [ ] indicates basic menu for menu selection.

Command	Menu	Function
PTP	[PTP]	Point to point control data input
TIMER	[M1]	Timer command, WAIT command, and Monitor Variable command
OUT	[M1]	DO output
CALL	[M1]	CALL command or PALLET command
GOTO	[M1]	Unconditional branch
IF	[M1]	Conditional branch command
SPEED	[M2]	Setting speed for PTP control
VAR	[M2]	Variable operation and FLAG operation
REPT	[M2]	Repeating command
REM	[M2]	Comments
ORG	[M2]	Homing
MATRIX	[M3]	Matrix command
MFB	[M3]	MFB control (***Fics-Atoms*** series)
DDA	[M3]	Linear, circular and arc interpolation commands

5-1: Screen of Program Step

An example of program step screen is shown in the figure on the right.

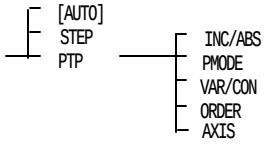
```
-DATA-AUTO STEP PTP
PGM=100 X=+xxxx.xxmm
 N=0001 Y=+yyyy.yymm
ABS .   [XYZ(W)]
```

Screen of Program Step

5-2: Entering and Editing Position Data

For positioning control, it is necessary to specify coordinates of a target point. Selecting the [PTP] menu leads to the following screen.

```
X=. . . . . mm
Y=. . . . . mm
```



When data input field is filled with "…", it means that positioning control of the axis is inactive. Positioning control is activated once the data input field is filled with target coordinate data set by directly keyed-in digits or by teaching when the cursor is blinking. To deactivate positioning control again, move the cursor to the axis and press repeatedly the [VAR]/[CON] key (<F3> in [PTP] menu). A more convenient way of doing this is to delete the step, and select [PTP] menu again.

```
-DATA-INC PMODE VAR
PGM=100 X=-xxxx.xxmm
 N=0001 Y=+yyyy.yymm
ABS .   [XYZ(W)]
```

Entering and Editing Position Data

The figure on the right shows an example where 2 axes are activated for positioning control. Data of target position contains the following attributes:

1. ABS/INC: Data can be either expressed in absolute [ABS] or relative [INC] coordinate. Status of ABS or INC is part of the step data and can be changed by menu selection.
2. PTP Running Mode [PMODE] : Specify either "next step is not executed untill execution of current step is completed"("."), or "next step is executed while current step is being executed"("->")

In the case of "->", if there is positioning control command for the axis which is not moving, the command is executed. If there is positioning control command for the axis which is still moving, the command is executed after positioning control is completed.

FLAG00 has special meaning, which indicates whether PTP control is completed (FLAG00=0) or is running (FLAG00=1). This flag makes it easy to check whether PTP control is completed by the "IF" command in the steps following PTP commands.

3. [CON]/[VAR]: Specifies whether to use constant or variable for coodinate data. VAR00 has a special meaning for deactivating motion to the specified axes, i.e., positioning control does not apply to these axes. Also it should be noted that in the case of variable specification, the unit of variables is treated as "**pulse"** even if the coordinate system is set as "**mm"**.
   Specification by constants: X=±xxxx.xxmm
   Specification by variables: X=VARnn
4. [ORDER]:
   Specifies moving order for systems with more than 3 axes.
   By default, the axes are in alphabetical order. By selecting the [ORDER] menu, the order of axes changes as follows. For example, a system with 3, 4 axes can be ordered as

[XYZ(W)]->[XY->Z]->[Z->XY]->[XYZ(W)]

A system with 5, 6 axes can be ordered as

[XYZ(W)]->[XY->Z]->[Z->XY]->[ALL]->[XYZ(W)]

"[XY->Z]", for example, means that only after the positioning of X and Y axes is completed, the positioning of Z axis is executed, irrespectively of [PMODE].

5. [AXIS]: For a system with more than 5 axes, select XYZ axes or WUV axes.

<XYZ(W)>-> <WU(V)>-> <XYZ(W)>

6. Simultaneous Positioning of Multi-axis: For system with 5, 6 axes, under normal circumstances, one PTP command controls either XYZ or WUV axes simultaneously. To PTP control 5 (or 6) axes at the same time, select [ALL] from the [ORDER] menu and specify PTP command for WUV axes in the next program step.

5-2-1: Other Keys for Inputting Data

In addition to menu selection, the following keys can be used for data input.

1. JOG keys (<-X, X->, <-Y, Y->): Manual (JOG) operation.
2. <START> key: Start PTP control to position on display.
3. <⊙> key: read current physical position as coordinate data.
   To read current physical position into an axis which is deactivated (data field filled with …), move cursor to the axis field before pressing <⊙> key.

5-2-2: Teaching Positions

On some occasions, specifying PTP target position by teaching can be very effective. The following operations must be performed after homing, except if homing is set to NEEDLESS (not needed) (Note: position teaching applies only to absolute coordinate).

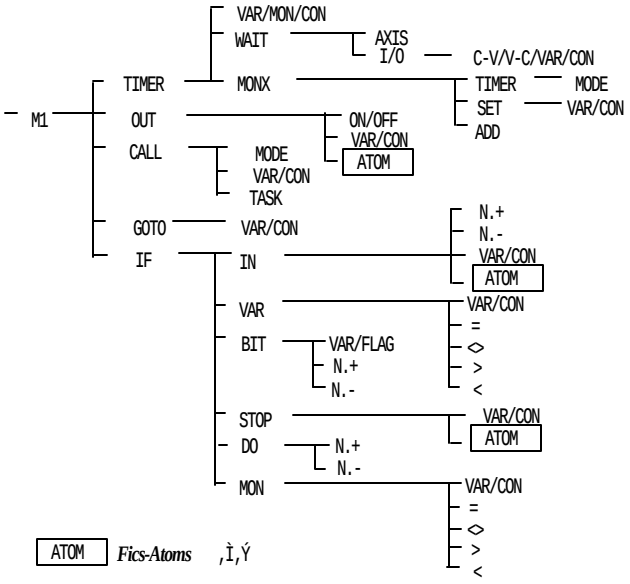
1. Key-in approximate numeric data in PTP command.
2. Press <START> key to move to coordinate specified in 1.
3. Press JOG keys to move desired axis to desired position.
4. Press <⊙> key to read current coordinates into data input field.
5. Press <ENT> key to register current data as step data.

The first two steps can be omitted if not necessary.

5-3: M1 Menu

In **[M1]** menu, there are 5 commands:

TIMER	Timer command
OUT	DO output command
CALL	CALL or PALLET commands
GOTO	Unconditional branch command
IF	Conditional branch command



5-3-1: Timer Command

The [TIMER] menu allows user to insert unconditional waiting command. Waiting time can be specified by constants [CON], variables [VAR], or Monitor Variables from [CON],[VAR],[MON] menus.

Constants:	TIMER=tt.tts
Variables:	TIMER=VARnn
Monitor Variables:	TIMER=MONnn

"nn" stands for Variable number (1~96)
(values are in 10msec unit)

- VAR96 is used by ***Fics***-III for PALLET program (Matrix). Do not use VAR96.

-DATA- VAR WAIT MONX
PGM=100
N=0001 TIMER=tt.tts

Timer Commad Menu

-DATA-AXIS I/O <CLR>
PGM=001 WAIT
N=0001 AXIS(******)

Waiting for Completion of Positioning

-DATA-AXIS I/O <CLR>
PGM=001 WAIT unit=n
N=0001 AXIS(******)

Fics-Atoms Multi-unit

5-3-1-1: Checking and Waiting for the Completion of Positioning

Selecting the [WAIT] menu from the [TIMER] menu allows user to check completion of positioning of axes. When the specified axes are still moving, next program step is not executed. This function can be used as interlock between axes in multi-unit system configuration.

5-3-1-2: I/O Wait Branch Command

Selecting the [I/O] menu allows user to check completion of positioning of axes. Timer and input port can be specified by constants or variables from [C-V],[V-C],[VAR],[CON] menus.

If specified conditions are satisfied within specified time, next step is executed. Otherwise, jump to specified step after properly setting the following monitor variables. This is not treated as time-over error.

MON94: Current step number (=1 if N=0001)
MON95: Current step number+1
MON96: Error number (2 digit number <nn>)

```
-DATA- C-V
PGM=100 WAIT xxxs
 N=0001 IF DI:nn-m=b
 F.<nn> GOTO xxxx
```

I/O Wait Branch Command

5-3-1-3: Monitor Variables

There are 3 commands in the [TIMER]-[MONX] menu.

1. **Timer Control Command**: Selecting START (timer start) and STOP (timer stop) by [MODE] key to control Monitor Variables MON80~MON89.
2. **Monitor Variable Substitution Command**: Substituting constants or variables into Monitor Variables. If substituted with constants, the same value can be substituted into consecutive multiple Monitor Variables by specifying the number of variables with "mm".
3. **Addition of Monitor Variables**: Addition of Monitor Variables.

```
-DATA- MODE
PGM=001 TIMER
 N=0001   START MON8x
```

Timer Control Command

```
-DATA- VAR
PGM=001 MONmm=
 N=0001   +0000000000
              mm
```

Monitor Variable

```
-DATA-
PGM=001 MONnn=MONmm+
 N=0001   +0000000000
```

Addition of MON

5-3-2: OUT Command

From the [OUT] menu, up to 6 bits of specified channel can be turned ON/OFF if specified channel number is not "0". Specifically, from [ON/OFF] menu (<F1> key), the bit with cursor can be set to ON/OFF. "xx" represents channel number, "b" represents bit number.

Selecting [VAR]/[CON] menu (<F2> key) allows user to specify channel number and bit number by either contants or variables. If variables are not used, it is possible to specify 6 outputs. Press <ENT> key to switch between first screen and second screen. If variables are used, only 3 outputs can be specified[Vmm refers to the variable specifying channel number and Vnn refers to the variable specifying bit number]

The [ATOM] menu (<F3> key) is used to specify ON/OFF control of digital outputs on ***Atom*-SRA/SLA** series drives. In this case, "xx" denotes station number, and "b" denotes bit number. Station number corresponds to motor axis.
"Atom" is displayed when specifying digital outputs of *the* ***Atom*-SRA/SLA** drives.

```
-DATA-ON/OF VAR
PGM=100 OUT xx-b:OFF
 N=0001 (1)    xx-b:OFF
              xx-b:OFF
```

[CON] OUT menu

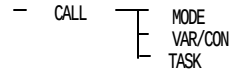
```
-DATA-ON/OF CON
PGM=100   Vmm-Vnn:OFF
 N=0001    Vmm-Vnn:OFF
    OUT   Vmm-Vnn:OFF
```

[VAR] OUT menu

```
-DATA-ON/OF VAR ATOM
PGM=100 OUT xx-b:OFF
 N=0001 (1)    xx-b:OFF
 ATOM          xx-b:OFF
```

Atom OUT Command

5-3-3: CALL Command



The following commands can be used by selecting the [CALL] menu. These CALL modes are switched by selecting the [MODE] menu.

1. CALL: Switching to execute sub-program. After all sub-program steps are executed, control returns to the next step.
2. PALLET: Switch to run specified program. Only one step of specified program is run, and control returns to the next step. Whenever PALLET program is called, one step is executed. In the program to be called by PALLET program, only one MATRIX command, or more than one PTP command can be used. As shown in the following example, the program contains only two steps if MATRIX command is used.
   <<**Matrix Command Example**>> One point in a MATRIX command is executed by one PALLET call.
   N=0001 MATRIX …
   N=0002 END
   <<**PTP Command Example**>> One step is executed by one PALLET call.
   N=0001 PTP X=+xxxx.xxmm Y=+yyyy.yymm
   N=0002 PTP X=+xxxx.xxmm Y=+yyyy.yymm
   N=0003 PTP X=+xxxx.xxmm Y=+yyyy.yymm
   …
   N=000n END
3. PALLET CHECK: Checking whether all steps have been executed by PALLET program.
   When all steps of PALLET program have been executed, or PALLET program was not executed at all, VAR96 is set to "0" and control returns to the next step following current step. When a step in PALLET program is running, step number of the next step in PALLET program is memorized in VAR96, and control returns to the next step.
   N=0010 PALLET xxx CHECK [Check completion of PALLET program]
   N=0011 GOTO STEP 100 [All steps of PALLET program are executed]
   N=0012 .... [PALLET program is running]
4. PALLET CLEAR: Initializing execution status of PALLET program. If PALLET command is called again, PALLET program will be executed from the first step. It should be noted that in this case the MATRIX command in PALLET program is also initialized.

5-3-3-1: Restrictions on Programs Called by CALL Command

A program to be called must be a sub-program within the range set by the initializing parameters. Refer to 6-5-1: Initialization of program area, part 2 of this manual.

CALL command can be used recursively. Maximum nesting depth for CALL command is 10 and up to 10 PALLET commands can be used in a program.

5-3-3-2: Assignment of CALL Program Number by Variable

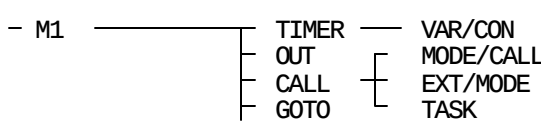
Program number can be determined by variables while running. Select the [CALL] menu to use fixed program number; select the [VAR] menu to assign program number by variables.

-DATA -MODE VAR TASK
PGM=001
N=0001 CALL xxx

CALL Command Menu

Use variable	CALL Vxx
Call fixed number	CALL xxx

5-3-4: Task Command



***Fics*-III** supports multi-task function, a very important feature for advanced motion system programming. Multi-task function allows for simultaneous and independent servo positioning control and I/O sequence control. In ***Fics*-III**, main program can accommodate up to 8 tasks in which all ***Fics*-III** commands except interpolations (DDA, ORG) and CALL commands can be executed. There is a special task called *system task* which is executed automatically at power-on.

5-3-4-1: Task Program Specification

(1). Task Start

Task program is a sub-program and is identified by a three digits program number. Task programs can be started from any program by invoking TASK xxx START command. Once task is started, it runs in the background. Up to 8 task programs can be started simultaneously, but once task program is running, it cannot be restarted.

(2). Task Stop and Termination

Task programs keep running even if the execution of the main program is suspended. The execution of task programs stops in either of the following cases:

1. When the final step has been executed.
2. When a mode is switched to modes other than **AUTO** mode.

If the final step of task program is the GOTO command, task program keeps running as long as the mode is the automatic operation mode.

When time-over error occurs, all task programs stop and restart by error reset.

When other errors occur, all tasks terminate.

For error information associated with task programs, refer to 8-7: Errors associated with task programs.

(3). Commands executable in task programs

TIMER	FLAG
OUT	REPEAT
GOTO	REM
VAR	PTP
NOP	SPEED
IF (IN, VAR, FLAG)	

(4). Commands not executable in task program

- CALL (CALL, PALLET, TASK)
- MATRIX
- ORG [executable in ***Fics-Atoms*** series]
- IF (stop command input for PTP operation)
- Other optional commands

5-3-4-2: Communication between Main Program and Task Program

Task programs, running in the background, communicate with the main program using FLAGs. For instance, in the following example, FLAG10 is used to make the task program wait for the completion of the P TP operation (step N=100) in the main program.

MAIN PROGRAM (PGM=010)		**TASK PROGRAM** (PGM=100)	
N=0031	FLAG10=0		
N=0032	TASK 100 START		
	…	N=0025	IF FLAG10 IS 0
N=0100	PTP X= ******		THEN GOTO 25
	Y= ******		
N=0101			

5-3-4-3: System Task

PGM=998 is a special task program called system task. Some important remarks about system task are listed below.

1. At power-on, system task starts automatically.
2. In **DATA** mode and **SYS** mode, the execution of system tasks is suspended.
3. System tasks are suspended by emergency stop, but resume execution from step 1, and keep running even during error processing.
4. PTP command is not executable in system tasks.
5. SPEED command is not executable in system tasks.

5-3-4-4: Output of *Fics* Status

The running status of the system task PGM=998 can be outputted to DO by setting system variables [SYS01] and [SYS02]. The default is bit 5 of <DO:01>, so [SYS01]=1, [SYS02]=5.

<DO:01> 0 Program is running
1 Program stops
2 Homing is completed
3 Overrun
4 Alarm ON
5 System task is running

-DATA-VAR
PGM=001
N=0001 GOTO xxxx

Branch Command

5-3-5: Branch Command

Use GOTO command to switch the program execution unconditionally to a specified step. The step number to jump to can also be specified by variables from the [VAR] menu. In I/O wait branch command, the step number stored in MON variables can be inserted into Variables and used in conjunction with GOTO command.

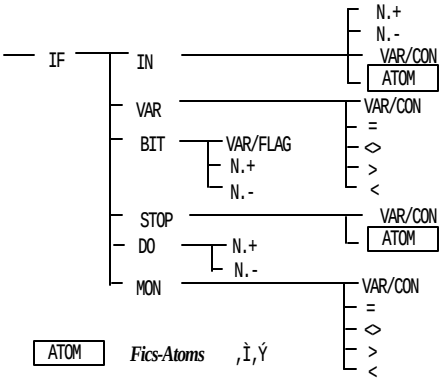
-DATA-CON
PGM=001
N=0001 GOTO VARnn

Branch Command with Variables

5-3-6: Conditional Branch Command

There are 6 types of conditional branch commands in the [IF] menu.

Command	Function
IN	Conditional branch commands by digital inputs.
VAR	Conditional branch commands by comparing variables.
BIT	Conditional branch commands by comparing FLAGs.
STOP	PTP stop command by DI input.
DO	Conditional branch commands by DO output states.
MON	Conditional branch commands by comparing Monitor Variables.



5-3-6-1: Conditional Branch Command by DI input

By selecting the [IN] menu, a branch command, waiting for a specified bit to turn ON/OFF for a specified time, can be inserted. If conditions are not satisfied when time is over, TIME OVER error occurs. If there is a STOP signal during execution of this command, the command will be executed again at next start.

When WAIT time is set to 0, if conditions are satisfied, jump to the specified step. Otherwise, go to the next step. It is also possible to specify the input channel and bit number by variables using the [VAR]/[CON] menu.

It should be noted that in **DEMO** mode, bit condition is always considered to be true without actually checking the bit state.

```
-DATA- N.+ N.- VAR
PGM=100 IF DI:nn-m=b
 N=0001   T.GOTO xxxx
<00>      F.WAIT xxxs
```

Conditional Branch by DI

```
-DATA- N.+ N.- CON
PGM=100 IF Vmm:Vnn=0
 N=0001   T.GOTO xxxx
<00>      F.WAIT xxxs
```

Conditional Branch by VAR

User can specify any message number to be displayed when TIME OVER error occurs. As shown in the figure on the right, message number should be specified in "<>". By default, <00> will display "TIME OVER" error. Error messages can be edited from ***WinFics***.

[ATOM] menu (<F2> key) is only available for ***Fics-Atoms*** with which the digital input of ***Atom*** drives can be specified. In this case, "xx" represents station number, "b" represents bit number. Station number corresponds to motor axis. The maximum number of bit is different depending on the type of ***Atom*** drives.

5-3-6-2: Branch Command by Comparing Variables

By selecting the [VAR] menu, a variable is compared with a constant or other variable. If conditions are met, specified program step is executed. Otherwise, go to the next step. Select the [VAR]/[CON] menu to compare with a constant [CON] or a variable [VAR].

```
-DATA-VAR    =    <>
PGM=100 IF VARnn =
 N=0001   xxxxxxxxx
          T.GOTOxxxx
```

Branch Command Menu [CON]

The following comparison conditions can be selected from the menu.

=	Equal
<>	Not equal
>	Left is larger than right
<	Left is smaller than right

```
-DATA-CON    =    <>
PGM=100 IF VARnn =
 N=0001          VARmm
          T.GOTOxxxx
```

Branch Command with [VAR]

5-3-6-3: Branch Command by FLAG Status

By selecting the [BIT] menu, status of FLAGs can be checked. If conditions are satisfied, go to specified program step. Otherwise, go to the next step.

Do not use FLAG00 which is used to monitor whether PTP operation is completed (FLAG00=0), or still running (FLAG00=1).

-DATA-VAR N.+ N.-
PGM=100 IF FLAGnn=x
N=0001 T.GOTO xxxx
<00> F.WAIT xxxs

Branch Command by FLAG

In the FLAG branch command menu, by selecting the [VAR] menu, specified bit of variables can be checked. If conditions are satisfied, go to specified step. Otherwise, go to the next step.

nn: Variable Number (01~96)

mm: Bit Number (00~31)

31 corresponds to MSB and 0 corresponds to LSB.

-DATA- FLAG N.+ N.-
PGM=100 IF Vnn_mm=b
N=0001 T.GOTO xxxx
<00> F.WAIT xxxs

Brach Command by Variable Bit Check

5-3-6-4: PTP Stop Command Activated by Input

Select the [STOP] menu to gradually stop positioning control when specified input bit is ON (1) or OFF (0). This command is also used in the case where positioning is running in the mode "->" (the next step starts before the completion of current positioning).

This gradual stop occurs before target position is reached, but is different from emergency stop and overrun. Coordinate data for current position is memorized.

From the [VAR]/[CON] menu, it is possible to specify input channel and bit numbers by variables.

-DATA- VAR
PGM=100 IF DI:nn-m=b
N=0001 T.STOP MOVE

PTP Stop Command by Input

-DATA- CON
PGM=100 IF Vnn:Vmm=b
N=0001 T.STOP MOVE

PTP Stop Command by Variables

<<**Program Example**>> When <DI:03-0> is ON, stop positioning control. It is necessary to wait for <DI:03-0> to turn ON at step 0002.

```
0001 PTP -> INC [XYZ]      +0100.00 +0200.00 +003000
0002 IF      DI:03-0   IS  0 THEN  GOTO  0002 ELSE  WAIT  000sec
0003 IF      DI:03-0   IS  1 THEN  STOP  MOVING
0004 END
```

For ***Fics-Atoms*** controller, the [ATOM] menu is added.

By selecting the [ATOM] menu, "ATOM" is displayed at the bottom line, and general DI of Atom series can be used for gradual stop.

This command is only valid if it is located right next to the step of PTP command in the mode ("->") which is the mode where the next step is executed without waiting for the completion of positioning of the current step.

In other cases, the step is skipped.

The "nn" in DI:nn-m does not have any meaning. All axes (station number) specified in the last PTP command are affected by this command. "m" refers to general DI bit number of Atom series, "b" is either ON(1) or OFF(0).

<<**Program Example**>>

Stop positioning when <DI1> on ***Atom*** drives turns ON.

```
0001 PTP -> INC [XYZ]      +0100.00 +0200.00 +003000
0002 IF ATOM DI:00-1  IS  1 THEN  STOP  MOVING
0003
```

-DATA- VAR ATOM
PGM=100 IF DI:nn-m=b
N=0001 T.STOP MOVE
ATOM

PTP Stop Command by Input

-DATA-CON ATOM
PGM=100 IF Vnn:Vmm=b
N=0001 T.STOP MOVE
ATOM

PTP Stop Command by Variables

5-3-6-5: Branch Command by DO Output Status

By selecting the [DO] menu, DO output status can be checked. If conditions are satisfied go to specified step. Otherwise, go to the next step.

-DATA- N.+ N.-
PGM=100 IF DO:nn-m=b
N=0001 T.GOTO xxxx
<00> F.WAIT xxxs

Branch Command by DO Output

5-3-5-6: Branch Command by Comparing Monitor Variables

By selecting the [MON] menu, Monitor variable is compared with constant or another Monitor variable. If conditions are met, go to specified program step. Otherwise, go to the next step.

-DATA-CON = <>
PGM=100 IF MONnn =
N=0001 VARmm
T.GOTOxxxx

Monitor Variable Branch Command [VAR]

Select the [VAR]/[CON] menu to compare with constant [CON] or variable [VAR].

Comparison conditions "=", "<>", ">", or "<" can be selected from the menu.

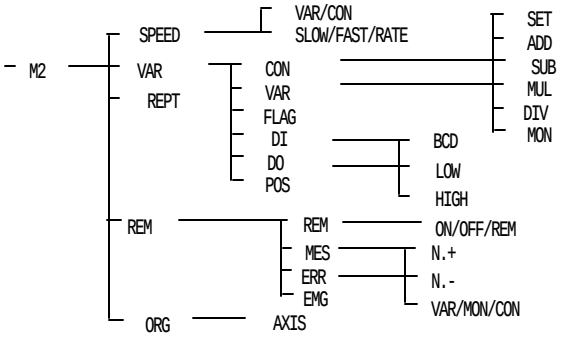
-DATA-VAR = <>
PGM=100 IF MONnn =
N=0001 XXXXXXXXXX
T.GOTOxxxx

Monitor Variable Branch Command

5-4: M2 Menu

There are 5 commands in the [M2] menu.

SPEED	Change PTP speed command
VAR	Substitution of variabls, FLAGs, and Arithmetic command
REPT	Repetition command
REM	Remark command
ORG	Homing command



5-4-1: Speed Command

In program, the SPEED command changes maximum speed for all PTP commands which follow. If the specified speed is greater than [MAX] specified by the SPEED system parameter, speed is set to [MAX]. If there is no SPEED command in program, [MAX] is set as the PTP speed.

Associated with this command, the following 2 menus are available, which makes it possible to set any PTP speed while programs is being executed.

The [VAR]/[CON] menu: Choose whether to specify speed with fixed constant[CON], or with variable [VAR]. If specified with "0" or "VAR00", it is ignored.

If specified with [CON] and [FAST], the screen looks like the figure on the right.

The [SLOW]/[Rate]/[FAST] menu: Change unit of speed.

-DATA- VAR SLOW
PGM=100 SPEED(FAST)
N=0001 X=xxxxmm/sec
Y=yyyymm/sec

Speed Command with [CON]

[FAST]	Specify the same unit as the maximum speed set by system parameter.
[SLOW]	For mm or pulse coordinate, specify unit with 1/100 of [FAST]. For rotational coordinates, the unit is taken as rpm.
[Rate]	Percentage of maximum speed specified by system parameter. Rate takes value in 0~100%. If specified with 0 or VAR00, it is ignored.

-DATA- VAR FAST
PGM=001 SPEED(Rate)
N=0001 X:nnn%
Y:nnn%

Speed Command with Rate

<<**Actual Motor Speed**>>

The actual motor speed is equal to (MAX speed set by system parameter)*(system override%)*(override command value%)

- If specified by variables, the screen for low speed looks like the figure on the right.
- If specified by variables, the units of speed are as follows.
  mm coordinate: FAST:1mm/sec, SLOW:0.01mm/sec
  pulse coordinate: FAST:100pps, SLOW:1pps
  angular/tilt coordinate: FAST: 1rpm, SLOW:1rpm

-DATA- CON Rate
PGM=100 SPEED(SLOW)
N=0001 X=VARnn
Y=VARmm

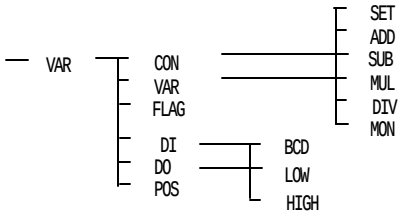
Speed Command(low speed) [VAR]

5-4-2: Assignment and Arithmetic of Variables

From the [VAR] menu, the following operations can be performed: arithmetic of variables with constants ([VAR]-[CON] menu), variables with variables ([VAR]-[VAR] menu), and FLAG settings.

In what follows, the "mm" and "nn" following VAR denote variable numbers (1~96).

If the calculated result exceeds 4 bytes(beyond -2,147,483,648 ~ 2,147,483,647) error occurs.



5-4-2-1: Arithmetic of Variables with Constants

The [VAR]-[CON] menu enables user to assign variables with constants, assign variables with Monitor Variables, and do calculations of variables with constants.

Function	Menu	Arithmetic
Assignment of Constants	[SET]	VARnn = ±xxxxxxxxx
Addition	[ADD]	VARnn = VARmm + ±xxxxxxxxx
Subtraction	[SUB]	VARnn = VARmm – ±xxxxxxxxx
Multiplication	[MUL]	VARnn = VARmm * ±xxxxxxxxx
Division	[DIV]	VARnn = VARmm ÷ ±xxxxxxxxx
Substitution of Monitor Variables	[MON]	VARnn = MONmm

Assignment of same constant to multiple variables

For assignment of constants only, it is possible to assign the same constant to multiple consecutively numbered variables by specifying the number of variables with "mm".

If mm=0, or 1, only the specified variable is assigned with values.

5-4-2-2: Arithmetic of Variables with Variables

The [VAR]-[VAR] menu enables user to assign variables with variables, assign variables with Monitor Variables, and perform calculations on variables with variables.

-DATA- SET ADD SUB
PGM=100 VARnn=VARmm
N=0001 +VARll

Addition of VAR with VAR

5-4-2-3: FLAG Setting

The [VAR]-[FLAG] menu enables user to assign data to FLAGs. In what follows, "nn" denotes FLAG number (1~96), "x" is either 0 or 1, "mm" denotes the number of consecutive FLAGs.

By setting "mm", it is possible to assign the same value to multiple consecutive FLAGs.

If mm=0, or 1, only one FLAG is assigned with values.

-DATA- <ENT><CLR>
PGM=100 FLAGnn = x
N=0001 mm

FLAG Setting Menu

5-4-2-4: Assignment of Variables with Input Channel

The [VAR]-[DI] menu allows user to assign variables with data from digital input channel.

Selectable sub-menus are listed below in [].

1. BCD 2 digit input [BCD]: VARnn = BCD DI:xx
2. Lower 4 bits input [LOW]: VARnn = LOW DI:xx
3. Upper 4 bits input [HIGH]: VARnn = HIGH DI:xx

In case of BCD input, data beyond 0~9 is considered as error.

In case of LOW and HIGH inputs, data between 0~15 is assigned to variables.

-DATA- BCD LOW HIGH
PGM=100 VARnn =
N=0001 BCD DI:xx

Assign Variables with DI

(DI:xx) 0 1
1 2 Lower 4 bits
2 4 (BCD)
3 8
4 1
5 2 Upper 4 bits
6 4 (BCD)
7 8

5-4-2-5: Output Variables to DO

The [VAR]-[DO] menu allows the user to output data to DO.
Sub-menus are listed below in [].

1. BCD 2 digit Output [BCD] :DO:xx = BCD VARnn
2. Lower 4 bits Output [LOW]:DO:xx = LOW VARnn
3. Upper 4 bits Output [HIGH]:DO:xx = HIGH VARnn

In case of BCD output, if the value of a variable is greater than 100, error occurs.

Bit position of BCD, LOW, HIGH are the same as in the previous item.

```
-DATA-     <ENT><CLR>
PGM=100 POS
 N=0001   X:VARnn
          Y:VARmm
```

Read Coordinates to Variables

```
-DATA- BCD LOW HIGH
PGM=100 DO:xx =
 N=0001   BCD   VARnn
```

Output Variables to DO

5-4-2-6: Assignment of Variables with Coordinates

The [VAR]-[POS] menu allows user to assign variables with current coordinates. Use variables numbered 1~96 for the desired axes. If VAR00 is used, the coordinates of the axes are ignored.

5-4-3: Repetition Command

The [REPT] menu allows user to execute "nn" times program steps between current step and specified step. Specified step number must be smaller than the step number of the current step.

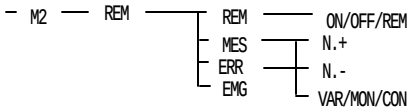
With automatic execution in **STEP** mode, this command is ignored.

```
-DATA-     <ENT><CLR>
PGM=100 REPEAT
 N=0010 nn TIMES TO
        STEP xxxx
```

Repetition Command Menu

5-4-4: Remark Command

The [REM]-[REM] menu allows user to insert remarks between program steps. Remarks do not affect the execution of programs in automatic operation, but they do have special meaning in the following cases by selecting sub-menu [ON][OFF][REM].



No	Case	Function
1	DISPLAY ON	Display program steps after current step in automatic operation.
2	DISPLAY OFF	Do not display program steps after current step in automatic operation.
3	DISPLAY REM	Display only REM command in automatic operation.

```
-DATA- ON
PGM=100 REMARK
N=0001 1234567890123
```

Remark Menu

Since message display uses CPU time, if 1 or 3 is not set explicitly, system assumes case 2.

Alphanumeric characters can be used in remarks. "." (PERIOD) means blank.

To enter a letter displayed on top of a numeric key, press that key and <SHIFT> key simultaneously. For instance, to enter "A", press the "A" key and <SHIFT> key.

5-4-5: Message Command

Message Command:

The [M2]-[REM]-[MESSAGE] menu enables user to display any message on the screen during automatic execution of programs.

```
MESSAGE No.(nn)
```

- Display specified message number
- If number=00, message is erased.
- The next step is executed after display.
- Can be used in task programs, except system task.

```
-DATA- N.+  N.-  VAR
PGM=001  MESSAGE
 N=0001     No.=<nn>
```

Message Command

Error Command:

By selecting the [M2]-[REM]-[ERROR] menu, it is possible to stop automatic execution of programs after displaying any message on screen.

```
** ERROR No.(nn)
```

- Display specified message number.
- After display, the system state is the same as if STOP wasinputted.
- If number=00, message is ignored
- Cannot be used in task programs.

```
-DATA- N.+  N.- MON
PGM=001  ERROR
 N=0001     No.=VARnn
```

Error Commmad

Emergency Command:

The [M2]-[REM]-[EMG] allows user to emergency stop automatic execution of programs.

```
** EMG ERROR No(nn)
```

- Send emergency stop command to all drives.
- Display error message corresponding to message number saved in MON96.
- If data in MON96 is 00, message is ignored.
- Cannot be used in task programs.

```
-DATA- N.+  N.-
PGM=001
 N=0001   EMG ERROR
```

Emergency Commmad

5-4-6: Homing Command

The [ORG] menu allows user to execute origin-return in automatic execution of programs. It should be noted that even if home offset is set by the system parameters, the system does not move to the offset position. If it is necessary to do so, use PTP command with target position being specified by 0 in absolute coordinate.

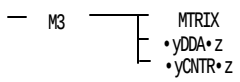
```
-DATA-AXIS
PGM=100 ORIGIN
 N=0001  ( ****** )
```

Homing Command

From the [AXIS] menu, select how to home.

In the [AXIS] menu, the character on cursor changes between axis name and '*'. Axes marked with '*' do not perform homing. If there is more than 1 axis name displayed, homing is performed for all these axes simultaneously.

5-5: Matrix Programming



The [M3]-[MTRIX] menu allows user to input matrix data. Matrix data is formatted data which can simply express more coordinates by specifying only total data numbers and pitches (distance between two points) in X and Y directions, respectively. Matrix data is usually used in PALLET programs.

Three dimensional palletizing commands are also available.

5-5-1: Selection of Matrix Type

There are 7 types of matrix data which can be selected from the [MATRIX] menu. To select specific matrix type, move cursor and press <ENT> key.

-DATA- <1-7><CLR>
PGM=100 1:G81 2:G82
N=0001 3:G83 4:G84
5:G85 6:G86 7:G87

Matrix Type Selection

5-5-2: Input of Matrix Data

Once matrix data type is selected, type number like G81 (type 1 selected) is shown, and user can enter total data number and pitch data.

-DATA- <CLR>
PGM=100 X=+xxxx.xxmm
N=0001 Y=-yyyy.yymm
G87INC Xn=Yn=nn

Matrix Data Input Menu

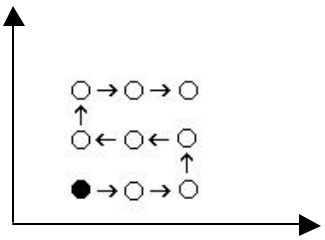
G87 is special case because it is designed for diagonal movement, so it requires same amount of data for both X and Y axes.

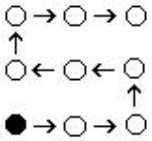
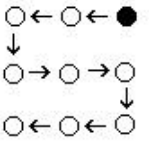
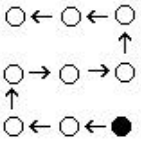
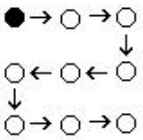
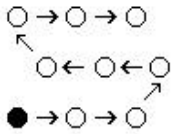
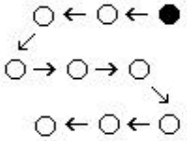
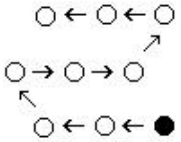
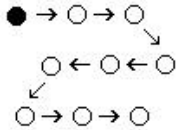
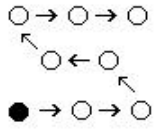
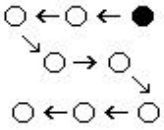
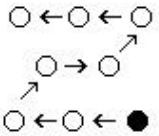
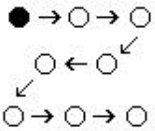
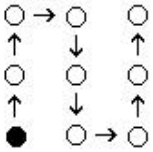
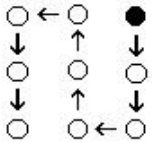
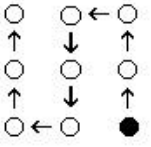
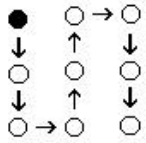
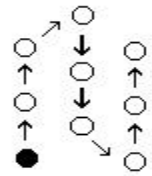
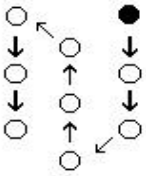
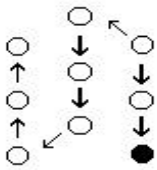
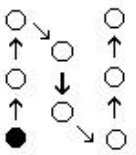
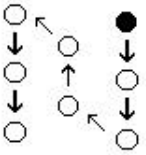
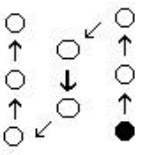
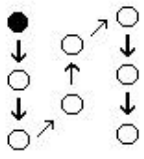
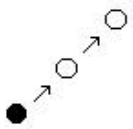
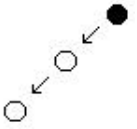
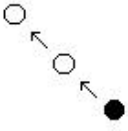
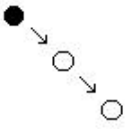
-DATA- <CLR>
PGM=100 X=+xxxx.xxmm
N=0001 Y=-yyyy.yymm
G81INC Xn=nn Ym=mm

5-5-3: Matrix Types

All the listed matrix types are referred to the following coordinate system. The example below shows
G81 Xn=4 (X Plus) Ym=3 (Y Plus)

●: Start Point
○: Mesh Point



Matrix Type	X Plus Y Plus	X Minus Y Minus	X Minus Y Plus	X Plus Y Minus
[**G81**] Xn=3 Ym=3				
[G82] Xn=3 Ym=3				
[**G83**] Xn=3 Ym=3				
[**G84**] Xn=3 Ym=3				
[**G85**] Xn=3 Ym=3				
[**G86**] Xn=3 Ym=3				
[**G87**] Xn=3 Ym=3				

- In G82, G83, G85, and G86, when movement changes direction (X or Y axis), the distance between mesh points is half of the pitch.

5-5-4: Examples of Palletizing Applications (1)

With use of the MATRIX command, programming of palletizing applications is greatly simplified. Typical palletizing application program consists of a main program and a sub-program called "pallet program". In the main program, "PALLET" command is used to call pallet program. It should be noted, however, that:

*One call of pallet program moves only one position along palletizing path.

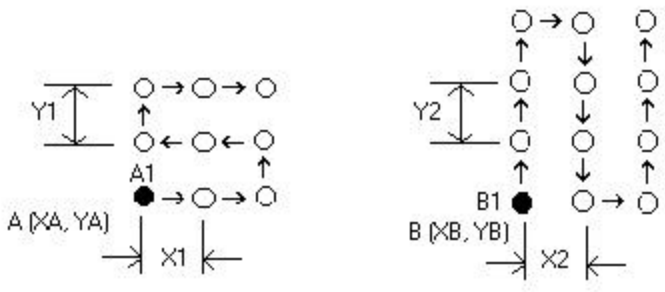
*The first time pallet program is called, the current position is memorized as the first position.

*On the second call, system moves to the second position.

*On the n-th call, system moves to the n-th point.

The following is an example showing how to pick and place objects from A1-A12 at location A to B1-B12 at location B. It is assumed that the grip can be opened(0) and closed(1) by the ON/OFF of <DO:01-7>.

<<Location A>> <<Location B>>



PGM=010 Main Program

N=	Command	Remark
0001	PTP X=XA, Y=YA	Move to A1
0002	PALLET 100	Memorize A1 in pallet program 100
0003	OUT 01-7:ON	Pick up object at location A
0004	PTP X=XB, Y=YB	Move to B1
0005	PALLET 200	Memorize B1 in pallet program 200
0006	OUT 01-7:OFF	Place object at location B
0007	PALLET 100	Move to An in pallet program 100
0008	OUT 01-7:ON	Pick up object at location A
0009	PALLET 200	Move to Bn in pallet program 200
0010	OUT 01-7:OFF	Place object at location B
0011	PALLET 100 CHECK	Check the completion of pallet program 100
0012	GOTO STEP 0014	If pallet program 100 is finished go to step 14
0013	GOTO STEP 0007	Otherwise repeat until all objects are handled
0014	END	End of program

PGM=100 PALLET program at location A

N=	Command	Remark
0001	MATRIX G81 X=X1 Y=Y1 Xn=4 Yn=3	Pallet type = G81, 4 rows and 3 columns
0002	END	End of program

PGM=200 PALLET program at location B

N=	Command	Remark
0001	MATRIX G84 X=X2 Y=Y2 Xn=3 Yn=4	Pallet type = G84, 3 rows and 4 columns
0002	END	End of program

5-5-5: Examples of Palletizing Applications (2)

The following is another example showing sequential positioning control to each point in matrix data. Since current matrix position and execution status of the pallet program are saved in memory, even if power is turned off, the program begins positioning from the next point when power is turned on.

PGM=020 Main Program

N=	Command	Remark
0001	PALLET 300 CHECK	Check pallet program execution status
0002	GOTO STEP 0004	If the first time pallet program is executed, go to step 0004
0003	GOTO STEP 0005	If not the first time pallet program is called, go to step 0005
0004	PTP ABS X=XA, Y=YA	Initial position of pallet
0005	PALLET 300	Call pallet program
0006	GOTO STEP 0001	Repeat
0007	END	End of program

PGM=300 PALLET Program

N=	Command	Remark
0001	MATRIX G81 X=X2 Y=Y2 Xn=nn Yn=mm	Pallet type = G81, nn rows and mm columns
0002	END	End of program

5-6: Commands Related to *Atom*-MFB

***Atom*-MFB** is an analog input/output module which can be directly connected to DYNASERVO controllers via RS485 communication interface.

The [M3]-[MFB] menu allows user to access control commands of ***Atom*-MFB** for digital inputs/outputs and analog inputs/outputs. A maximum of 4 ***Atom*-MFB** can be connected to one ***Fics-Atoms*** series controller.

```
-DATA- TYPE        <CLR>
PGM=001 ATOM-DI
 N=0001      STATION=02
                CH=1 VARnn
```

MFB Command Menu

MFB Commands:

M3 ─┬─ MTRIX
    └─ MFB

Control inputs/outputs of ***Atom*-MFB.**
From [TYPE] menu, select digital or analog input/output.

Command	Function
ATOM-DI	Digital input
ATOM-DO	Digital output
ATOM-AI	Analog input
ATOM-AO	Analog output
STATION	Station number of ***Atom*-MFB** (Max station number of ***Atom*** drives + 1)
CH=1	Channel number (1~4)
VARnn	Assign variables with the input value of ***Atom*-MFB**. Output variables to ***Atom*-MFB**.

6: Copying Programs

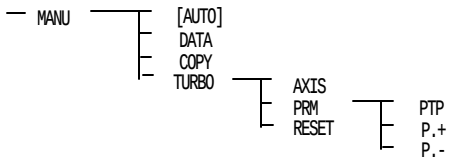
MANU ─┬─ [AUTO]
      ├─ DATA
      └─ COPY

The [MANU]-[COPY] allows user to copy programs.

```
-COPY-    <0-9><CLR>
 COPY <PROGRAM=XXX>
 NEW   PROGRAM=YYY
```

Copy of Programs

7: Tuning Servo Drives



The [MANU]-[TURBO] menu enables user to tune servo parameters (GPID) by creating an automatic shuttle movement between any specified two positions. It is usually required to keep motor running in order to tune drive parameters, therefore this is very useful function.

For the ***Fics-Atoms*** controllers**,** tuning of servo drive can be conducted from ***Fics*-RT1** terminal. Available menus are listed below.

Selecting axes: Use [AXIS] menu to select servo drive to be tuned.

Setting servo parameters: Select [PRM] menu to set various servo parameters. For ***Fics*-PDS/3,** no sub-menu is displayed.

Setting reference position: Select [RESET] menu to determine reference position of shuttle move.

Execution of automatic operation after completion of servo tuning requires completion of homing operation.

```
-Turbo-AXIS PRM RSET
 X=-xxxxxxxxx W=ttt
 V=xxxx rpm   xxxxms
  CURVE=0(0:L 1:S)
```

Servo Tuning Menu
(***Fics-Atoms*** Controllers)

7-1: Setting PTP Data for Automatic Shuttle Move

Name	Description
Starting position	Current position (When taught with arrow keys, the taught position is starting position and data here represents target position)
Reference position	Stop position when <STOP> key is pressed. Axis position when [TURBO] menu is selected, or position when [RESET] menu is selected.
Target position	Displayed coordinates under axis name. Automatic shuttle movement takes place between reference position and target position. Coordinate data can be taught or keyed-in.
Maximum speed	Running at specified maximum speed. Units are [rpm] for ***Fics-Atoms*** series, and [kpps] for pulse type controllers.
Deceleration time	Operated at specified accel/decel times. Acceleration time is the same as deceleration time.
Speed curve	Trapezoid (CURVE=0) or S-curve (CURVE=1)
WAIT time	Stop time at reference position and target position (unit:10msec).

7-2: Parameter Setting under Automatic Shuttle Movement

```
-Turbo-PTP  P.+  P.-
G=08 P=08 I=08 D=08
g=04 p=02 i=04 d=02
```

Servo Parameter Setting Menu

The Automatic shuttle movement starts with <START> key if all data mentioned above is set. To change the G, P, I, and D parameters on the fly, move cursor to desired field and press <F2> or <F3> key.

To increase parameter value: Select [P.+] menu (<F2> key).

To decrease parameter value: Select [P.-] menu (<F3> key).

7-3: Servo Drive Tuning Parameters

The following is a list of parameters for tuning. For details refer to ***Atom*-SRA/SLA** Servo Drive User's Manual.

Parameter	Description	Range	Default Value
G	Scaling gain of the servo feedback loop	(0~15)	8
P	Proportional gain of the feedback loop	(0~15)	8
I	Integral gain of the feedback loop	(0~15)	8
D	Differential gain of the feedback loop	(0~15)	8
g	Range of G	(1,2,4,8,16)	4
p	Range of P	(1,2,4,8,16)	2
i	Range of I	(1,2,4,8,16)	4
d	Range of D	(1,2,4,8,16)	2

* the 16 in g,p,i,d is displayed with 'H'

8: Errors and Troubleshooting

Understanding cause of errors is crucial for troubleshooting. To reduce downtime and to restore system to operating status as quickly as possible, it is recommended that messeges, error numbers and all possible reasons related to the errors be recorded.

8-1: What to Do When Errors Occur?

Availability of the following information will help determine cause of errors. Record and safeguard this information for repair or maintenance purposes.

Items	Contents
When?	▪ year, month, day, hour ▪ how many hours after power-on ▪ power failure? ▪ errors in system adjustment? ▪ errors in system running?(from when system started?)
What situation?	▪ in continuous running? ▪ which mode? ▪ in manual operation? ▪ when inputting data? ▪ motor was rotating?
What happened?	▪ alarm activated ▪ screen faded away? ▪ power interrupted?
Status?	▪ contents of parameters ▪ contents of programs ▪ position of movable parts ▪ sound from motor ▪ states of external machines
Reoccurence?	▪ reoccurence?(if so, in which condition?) ▪ occurred before?(if so, what's the cause at that time)

8-2: Errors with Alarms

When errors occur, some error messages appear on the bottom line of the screen, and system waits for users response. Moreover, error occurrence is signaled to bits<DO:01-4>(alarm ON). Errors can be reset by pressing <CLR> key if ***Fics*-RT1** is used. Error reset signal <DI:01-4>(error reset) has the same effect.

8-3: Possible Problems and Their Solutions

In the following cases, no alarm is activated.

	Problem	Solutions
1	**No display on LCD (or CRT)**	▪ Check power supply and cable. ▪ Check signal cable.
2	**Motor position deviates from pre-set position.**	▪ Check wiring of motor and encoder. ▪ Check ground cable of controller. ▪ Check voltage of power supply. ▪ Check if there is an instantaneous power down. ▪ Check if there is a strong electrical noise source nearby. ▪ Check if there are some back-lash or looseness in power-transfer units. (such as bolts of coupling and gear, etc.)
3	**Continuous vibration of motor .**	▪ Tune feedback gains. ▪ Check if there is looseness or back-lash in power transfer units.
4	**Motor does not rotate**. (The coordinates on screen do not change.)	▪ Check movement of motor by JOG. Referring to 6.5.2 Initializing Parameters, part 2 of this manual, initialize parameters and switch power-on again. In normal state, coordinates on screen should change by pressing the "X->" and "<-X" arrow keys, even if controller and servo drive are disconnected.
5	**Motor does not rotate.** (Despite that coordinates on the screen are changing.)	Output pulses from controller are not sent to servo drive. Since the motor interface is isolated by photo-couplers, it is necessary to provide external power supply to the motor interface of controller. Because each axis is separated, power should be supplied to all axes respectively. This power supply should be independent of the main DC5V power for CPU board.
6	**Motor rotates only in one direction.**	Check if pulse command type of the controller and of the servo drive are consistent. There are two types of pulse signals: **PULSE/DIR**: Pulse signals and direction signals are used as commands. (also called the 1-pulse type) **CW/CCW**: Clockwise and counterclockwise pulses are used as commands. (also called the 2-pulse type)
7	**Motor rotates in opposite direction.**	Refer to 6.2.4 PULSE Parameters, and 6.2.3 JOG-KEY Parameters, part 2 of this manual.

8-4: Errors at Power-on

Message	Cause & Solutions	Notes(References)
PGM Check ERROR	Incorrect status in program area. It occurs either when an old version program has been loaded or when the memory has been damaged	PROGRAM ERROR may occur even if memory status is forced to be correct. Refer to 6-5:Initialization, part 2 of this manual.
PRM Check ERROR	Incorrect status in parameter area. It occurs either when an old version program has been loaded or when the memory has been damaged.	PARAM ERROR may occur even if the memory status is forced to be correct. Refer to 6-5 : Initialization, part 2 of this manual.
PROGRAM ERROR	Incorrect status in program area. It is necessary to initialize the program.	Refer to 6-5 : Initialization, part 2 of this manual.
PARAM ERROR	Incorrect status in parameter area. It is necessary to initialize the parameter.	Refer to 6-5 : Initialization, part 2 of this manual.
BATTERY CHECK	Low battery for memory backup. Change battery.	An alarm sign is blinking on bits <DO:01-4>.
ROM Check ERROR	ROM and hardware are mis-matched. ROM area for optional software is checked for verification of hardware matching.	Refer to 6-5-4 : Version Information, part 2 of this manual.

8-5: Errors during Program Editing

Message	Cause & Solutions	Notes(References)
NUMBER ERROR	Step number specified does not exist.	Check program.
PGM AREA OVER	Over-flow of program memory. Delete unnecessary programs.	Refer to 4-6: Deleting program.

8-6: Errors in Jog or Automatic Operation

When the error mentioned below occurs, homing is necessary, because the position of drive becomes uncertain.

Message	Cause	Notes(References)
PGM=999 stopped	Homing sequence program PGM=999 has been interrupted.	Homing has not been completed.
EMERGENCY	Emergency stop signal is ON.	Refer to the note below .
OVER-RUN	Overrun signal is ON.	In the case of manual operation, the drive has reached mechanical-limit position. In the case of automatic operation, check the program(position data)

Notes:

- In the ***Fics*** series, it is assumed, by default, that "1" corresponds to sensor ON and "0" corresponds to sensor OFF. The logic relation of each I/O bit can be changed. Refer to 6-3: Setting bit mask of DI, part 2 of this manual.
- Errors of emergency stop or errors related to motor are detected except in the **SYS** mode.
- When such errors are displayed on the initial screen, by pressing the <CLR> key, the screen is automatically changed to the **SYS** mode to check input bit and to invert it logically.
  When error is reset, the screen is not changed to the **SYS** mode by error reset signal <DI:01-4> of external input
- Emergency stop input or overrun slows down (at half of deceleration time) motor rotation rapidly to a stop, and motor becomes free.

MOTOR ERROR	Alarm signal is generated by servo drive.
NOT READY X	Servo drive is not ready. Check if controller and drive are connected correctly. X refers to the X axis.

○ The main causes of motor errors are:

- problems with motor itself
- problems with servo drive
- problems with cable between motor and servo drive.

The causes and solution are as follows:

NO	Cause	Solutions
1	Motor overload	▪ Reduce mechanical load ▪ Lower speed and/or increase acceleration/deceleration parameters ▪ Check mechanical parts after power is turned off (e.g., gear burning-in, ballscrew burning-in, brake and clutch pulling-in) ▪ Replace motor
2	Problems with insulation of motor	▪ Clean or replace motor
3	Problems with motor encoder	▪ Replace motor
4	Magnetic degradation of motor	▪ Replace motor
5	Mismatched servo tuning	▪ Tune the parameters of servo drive
6	Damage of servo drive	▪ Replace servo drive
7	Looseness in motor electrical connectors	▪ Check insertion parts of connectors
8	Cable problems between motor and servo drive	▪ Check connector parts. ▪ Repair wiring

8-6: Errors Occurring in - AUTO - Mode

Message	Cause	Solutions (Reference)
ORIGIN ERROR	Homing has not been completed	Execute homing
TIME OVER	Conditions are not satisfied in specified time. Check input signals.	Refer to 3-1:Checking input, part 2 of this manual.
GOTO STEP ERROR	Step No. used in GOTO command does not exist.	
STACK OVERFLOW	Depth of nested CALL command invocation is over ten, or number of PALLET programs is over ten.	
RECURSIVE CALL	Program is being called by itself.	
CALL NO. ERROR	Program Number in CALL command does not exist or it is not a sub-program.	Refer to 5-3-3-1:Restrictions in Program called by CALL Command
NOT SUB PROGRAM	Program No. to be executed is a main program.	Same as above.
CH NO. ERROR	Incorrect channel No. in OUT command or unconditional branch command .	
PGM. EMPTY	No executable steps.	
BARRIER ERR	Target coordinates exceed specified software limit.	Check program (position data)
VAR/FLAG NO. ERR	Number of variables or flags is out of range . (Variable 1~96)(Flag 1~96)	
PGM NOT DEFINED	Program to be executed does not exist	
VAR POS OVERFLOW	Values of the variables used as coordinates are beyond the range of 3bytes(-838860~8388607)	
VAR CALC ERR	Variable calculation errors such as overflow or divided-by-zero.	
PALLET EXEC ERR	Command other than PTP and MATRIX exists in PALLET program	
PALLET NO ERR	Specified program number in PALLET command does not exist.	
BCD CODE error	Incorrect BCD code of the value for assignment to DI.	Refer to 3-1:Checking input, part 2 of this manual.
ILLEGAL COMMAND	Command to be executed is installed as an option.	

8-7: Errors Associated with Task Programs

Error Classification	Error Message	Cause & Solution
Errors which occur when TASK command is running	PGM RUNNING(TASK)	Specified program is being executed as a task program.
	SAME PGM(TASK)	Specified program is the program itself or a program called by itself.
	NO PGM(TASK)	Specified program does not exist
	NO STEP(TASK)	Executable STEP does not exist in specified program
	WRONG STEP(TASK) P:ppp N:nnnn	There is a non-executable step in specified program nnnn: the step number containing the non-executable command.
	TABLE OVER(TASK)	More than 8 tasks
Error which occurs when executing CALL command	SAME PGM(TASK)	Specified program is being executed currently as a task program
Time-Over error	DI nn-b:s P:ppp N:nnnn FLAG nn:s P:ppp N:nnnn	nn: Channel number or flag number b: Bit number s: 0 or 1 ppp: Task program number with error nnnn: Step number with error
Error which occurs when task program is running	Usual Error Message P:ppp N:nnnn	ppp: Task program number with error nnnn: Step number with error

8-8: Other Errors

Cause	Message	Reference
Specified program No. in printing list does not exist.	PGM NOT DEFINED	Refer to 4: Listing, part 2.
Homing program (PGM=999) is undefined.	PGM999 NOT FOUND	Refer to 6-2-1:Homing Parameters, part 2.
Incorrect data have been received in communication with ***WinFics.***	COMMUNICATE ERR	
Program No. to be executed is a sub-program.	NOT MAIN PROGRAM	
Non-executable command exists in program for emergency stop.	WRONG STEP(EMG)	Refer to 6-5-1-4: Setting startup and emergency programs, part 2.
<START> input in the initial screen and JOG mode.	Not Ready	
An error occurred during writing to EEPROM.	EEPROM WRITE ERR	

Appendix: System Resources Tables

The following tables list system resources which are very important for system programming. Some of the resources have already been used or reserved by ***Fics*-III** for special purposes and must not be accessed by users.

1. ***Fics*-series controller on-board DI/DO table
2. Program numbers table
3. System variables table
4. Variables table
5. Monitor variables table
6. FLAGs table

Table of *Fics*-Series Controller On-Board DI/DO

DI (Channel-bit)			
01-0	Emergency stop	03-0	
01-1	Start	03-1	
01-2	Stop	03-2	
01-3	Homing with external DI	03-3	
01-4	Error reset	03-4	
01-5		03-5	
01-6		03-6	
01-7		03-7	
02-0			
02-1			
02-2			
02-3			
02-4			
02-5			
02-6			
02-7			

DO (Channel-bit)			
01-0	Program is running	02-0	
01-1	Program stopped	02-1	
01-2	Homing is completed	02-2	
01-3	Overrun	02-3	
01-4	Alarm-on	02-4	
01-5		02-5	
01-6		02-6	
01-7		02-7	

The number of inputs/outputs and channels is different for every controller in ***Fics*-series.

Table of Program Numbers

No	Description	No.	Description
000	Must not be used		
001	Main program		
		998	System task program
		999	Homing sequence program

By default, numbers 100-999 are assigned to sub-programs.

Table of System Variables

Name(Channel/Bit)	Description	Remark
SYS01/SYS02	DO output channel/bit when running task 998	
SYS03/SYS04	Atom ready input channel/bit	
SYS05/SYS06		
SYS07/SYS08		
SYS09/SYS10		
SYS11/SYS12		
SYS13/SYS14		
SYS15/SYS16		
SYS17/SYS18	Temporary stop execution channel/bit	
SYS19/SYS20		
SYS21/SYS22	Automatic/Manual mode switching channel/bit	
SYS23/SYS24	Demo operation channel/bit	
SYS25/SYS26	Cycle operation channel/bit	
SYS27/SYS28	Step operation channel/bit	
SYS29/SYS30		
SYS31/SYS32		
SYS33/SYS34		
SYS35/SYS36		
SYS37/SYS38		
SYS39/SYS40		
SYS41/SYS42		
SYS43/SYS44	Manual I/O ignore channel/bit	
SYS45/SYS46	DO output channel/bit in automatic operation mode	
SYS47/SYS48		
SYS49/SYS50		
SYS51/SYS52		
SYS53/SYS54		
SYS55/SYS56		
SYS57/SYS58		
SYS59/SYS60		
SYS61		
SYS62		
SYS63		
SYS64		
SYS65		
SYS66		
SYS67		
SYS68		
SYS69		
…		
SYS71/SYS72	JOG channel/bit by external DI	customization
SYS95/SYS96		

Table of Variables

Name	Description	Name	Description
VAR00	(system use: special variable)	VAR51	
VAR01		VAR52	
VAR02		VAR53	
VAR03		VAR54	
VAR04		VAR55	
VAR05		VAR56	
VAR06		VAR57	
VAR07		VAR58	
VAR08		VAR59	
VAR09		VAR60	
VAR10		VAR61	
VAR11		VAR62	
VAR12		VAR63	
VAR13		VAR64	
VAR14		VAR65	
VAR15		VAR66	
VAR16		VAR67	
VAR17		VAR68	
VAR18		VAR69	
VAR19		VAR70	
VAR20		VAR71	
VAR21		VAR72	
VAR22		VAR73	
VAR23		VAR74	
VAR24		VAR75	
VAR25		VAR76	
VAR26		VAR77	
VAR27		VAR78	
VAR28		VAR79	
VAR29		VAR80	
VAR30		VAR81	
VAR31		VAR82	
VAR32		VAR83	
VAR33		VAR84	
VAR34		VAR85	
VAR35		VAR86	
VAR36		VAR87	
VAR37		VAR88	
VAR38		VAR89	
VAR39		VAR90	
VAR40		VAR91	
VAR41		VAR92	
VAR42		VAR93	
VAR43		VAR94	
VAR44		VAR95	
VAR45		VAR96	(system use)
VAR46			
VAR47			
VAR48			
VAR49			
VAR50			

Table of Monitor Variables

Name	Description	Name	Description
MON01		MON51	
MON02		MON52	
MON03		MON53	
MON04		MON54	
MON05		MON55	
MON06		MON56	
MON07		MON57	
MON08		MON58	
MON09		MON59	
MON10		MON60	
MON11		MON61	
MON12		MON62	
MON13		MON63	
MON14		MON64	
MON15		MON65	
MON16		MON66	
MON17		MON67	
MON18		MON68	
MON19		MON69	
MON20		MON70	
MON21		MON71	
MON22		MON72	
MON23		MON73	
MON24		MON74	
MON25		MON75	
MON26		MON76	
MON27		MON77	
MON28		MON78	
MON29		MON79	
MON30		MON80	(system use: power-on timer * 1sec)
MON31		MON81	(system use: automatic operation timer *1sec)
MON32		MON82	(system use: 1 cycle duct time * 1sec)
MON33		MON83	(system use: unused)
MON34		MON84	(system use: decrement timer * 10msec)
MON35		MON85	(system use: decrement timer * 10msec)
MON36		MON86	(system use: decrement timer * 10msec)
MON37		MON87	(system use: increment timer * 10msec)
MON38		MON88	(system use: increment timer * 10msec)
MON39		MON89	(system use: increment timer * 10msec)
MON40		MON90	(reserved)
MON41		MON91	(reserved)
MON42		MON92	(reserved)
MON43		MON93	(reserved)
MON44		MON94	(system use: for I/O wait command)
MON45		MON95	(system use: for I/O wait command)
MON46		MON96	(system use: for I/O wait command)
MON47			
MON48			
MON49			
MON50			

Table of FLAGS

Name	Description	Name	Description
FLAG00	(system use: PTP is running)	FLAG50	
FLAG01	(system use: X-axis is running)	FLAG51	
FLAG02	(system use: Y-axis is running)	FLAG52	
FLAG03	(system use: Z-axis is running)	FLAG53	
FLAG04	(system use: W-axis is running)	FLAG54	
FLAG05	(system use: U-axis is running)	FLAG55	
FLAG06	(system use: V-axis is running)	FLAG56	
FLAG07	(reserved)	FLAG57	
FLAG08	(reserved)	FLAG58	
FLAG09		FLAG59	
FLAG10		FLAG60	
FLAG11		FLAG61	
FLAG12		FLAG62	
FLAG13		FLAG63	
FLAG14		FLAG64	
FLAG15		FLAG65	
FLAG16		FLAG66	
FLAG17		FLAG67	
FLAG18		FLAG68	
FLAG19		FLAG69	
FLAG20		FLAG70	
FLAG21		FLAG71	
FLAG22		FLAG72	
FLAG23		FLAG73	
FLAG24		FLAG74	
FLAG25		FLAG75	
FLAG26		FLAG76	
FLAG27		FLAG77	
FLAG28		FLAG78	
FLAG29		FLAG79	
FLAG30		FLAG80	
FLAG31		FLAG81	
FLAG32		FLAG82	
FLAG33		FLAG83	
FLAG34		FLAG84	
FLAG35		FLAG85	
FLAG36		FLAG86	
FLAG37		FLAG87	
FLAG38		FLAG88	
FLAG39		FLAG89	
FLAG40		FLAG90	
FLAG41		FLAG91	
FLAG42		FLAG92	
FLAG43		FLAG93	
FLAG44		FLAG94	
FLAG45		FLAG95	
FLAG46		FLAG96	
FLAG47			
FLAG48			
FLAG49			