



# **FD CONTROLLER INSTRUCTION MANUAL User Task**

**5th edition**

- Before attempting to operate the robot, please read through this operating manual carefully, and comply with all the safety-related items and instructions in the text.
- The installation, operation and maintenance of this robot should be undertaken only by those individuals who have attended one of our robot course.
- When using this robot, observe the law related with industrial robot and with safety issues in each country.
- This operating manual must be given without fail to the individual who will be actually operating the robot.
- Please direct any queries about parts of this operating manual which may not be completely clear or any inquiries concerning the after-sale service of this robot to any of the service centers listed on the back cover.

**NACHI-FUJIKOSHI CORP.**



# Table of Contents

## Chapter 1 Outline

1.1 Outline ..... 1-1

1.2 Priority..... 1-3

## Chapter 2 Creation of User Task Program

2.1 Creation of Robot Language Program ..... 2-1

    2.1.1 Commands unavailable for user tasks ..... 2-1

    2.1.2 Copying program file created ..... 2-2

2.2 Making Use of Variable Names ..... 2-3

    2.2.1 Variable definition file..... 2-4

    2.2.2 Functions ..... 2-4

2.3 Commands Available for User Tasks..... 2-5

## Chapter 3 Playback of User Task Program

3.1 Starting Methods of User Task Program..... 3-1

    3.1.1 Starting when turning ON power supply ..... 3-1

    3.1.2 Starting from user task monitor ..... 3-1

    3.1.3 Starting from soft key ..... 3-3

    3.1.4 Starting from function..... 3-4

## Chapter 4 User Screen

4.1 User Screen..... 4-1

    4.1.1 How to display the User screen..... 4-2

4.2 User Screen Draw Commands..... 4-3

    4.2.1 User screen display..... 4-3

    4.2.2 User screen title setting..... 4-4

    4.2.3 Character string display..... 4-5

    4.2.4 Drawing figures..... 4-6

    4.2.5 Drawing software keys ..... 4-10

    4.2.6 Drawing a message box..... 4-15

    4.2.7 Getting the information of the key operation ..... 4-16

    4.2.8 Turning ON/OFF the user task screen ..... 4-18

## Chapter 5 Example of User Task Program

5.1 Display of Error Numbers of All Units ..... 5-1

5.2 Initialization of Variables ..... 5-2

5.3 Conversion from signal to variable ..... 5-3

5.4 Control of Motion Range..... 5-4

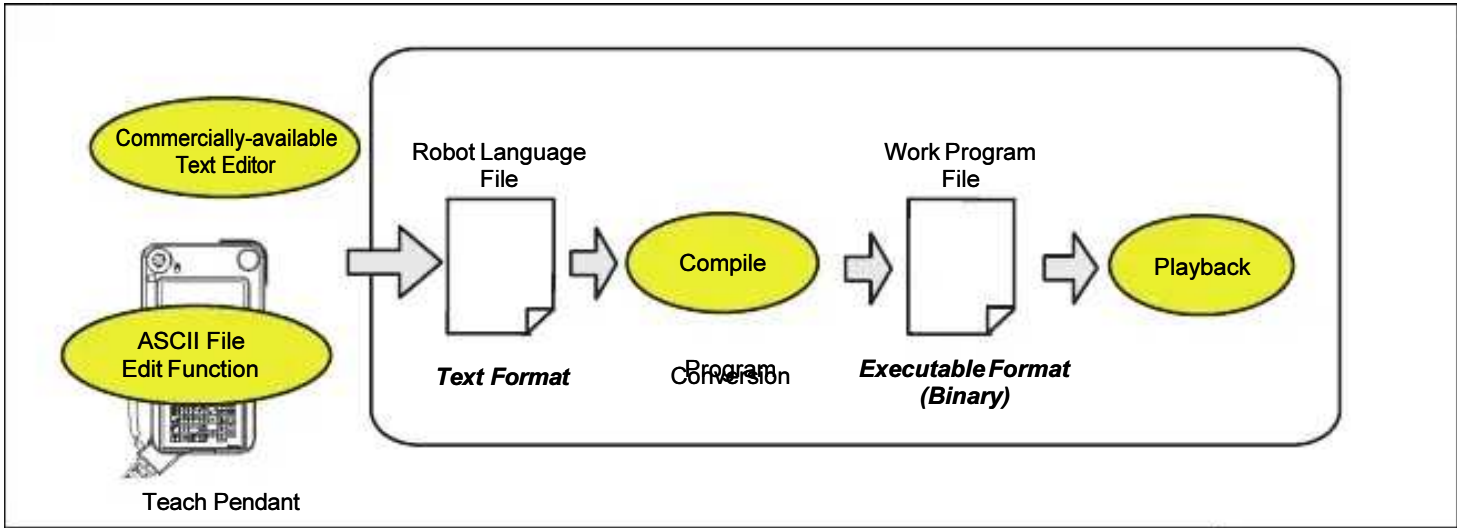
5.5 Displaying a user monitor screen ..... 5-5

# Chapter 1      Outline

## 1.1 Outline

The term “user task” represents the function of running a program created in the robot language involving no robot motions in parallel with a robot motion program.  
This function performs a time-consuming calculation in parallel with robot motions, thus enabling reduction of cycle time. In addition, the function makes it possible to arrange windows on the Teach Pendant screen and display a variety of statuses.

The robot language program is a text file (ASCII text file), which can be created with a commercially-available text editor. This program can be also edited using the teach pendant (ASCII file edit function).  
A program created in the robot language are compiled into a program in an executable format and then played back. The robot language format and the executable format can be converted bi-directionally (compile / decompile).



From creation through playback of user task program

Up to 16 USER TASKs can be performed in parallel. The each USER TASK can be managed by the number of 1 to 16 and be monitored by USER TASK monitor window. (In case of FDV4.70 or before, the maximum number is 4).



User Task Monitor

**i**  
IMPORTANT

After FDV04.71, the maximum number of the USER TASKs that can be used at the same time becomes from 4 to 16.  
In case of FDV4.70 or less, the maximum number is 4.  
In case of FDV4.71 or more, the maximum number is 16.

**i**  
IMPORTANT

The playback of user task program may have adverse influence on robot motions. To avoid that, play back the program after checking for loads on the User Task Monitor.



There are some commands in the robot language that cannot be used for user tasks. Make sure of such command referring to information in the instruction manual “Command Reference”.

## 1.2 Priority

The user task is available for priority setting. The priority setting can be individually made from the User Task Monitor to the task numbers 1 to 16 (In case of FDV04.07 or less, 4). Furthermore, the priority setting makes it possible to strike a balance between a load applied to this controller system and a playback speed through the operation of the user task.

Make priority setting in six levels from 1: lowest priority to 6: highest priority. (In case of FDV4.48 or before, the highest level is 5) Since Priority Level 1 has low severity to other processing, setting Level 1 to a user task that is always performed in the background can reduce influence on the playback of a robot program. Setting priority to Level 6 enables high-speed processing, thus making it possible to reduce cycle time in case the robot operates after the completion of computing.

Load levels are displayed on the User Task Monitor with the operator class set to **EXPERT** or higher. After the robot program and the user task program are played back, ensure on the User Task Monitor that there are no problems with the load levels

11 User Task Monitor					
	Prog.	Priority	Comment	Status	Error
1	900	4	Usertask program	Start	
2	0	3		Stop	
3	0	3		Stop	
4	0	3		Stop	
5	0	3		Stop	
6	0	3		Stop	
7	0	3		Stop	
8	0	3		Stop	
9	0	3		Stop	
10	0	3		Stop	
11	0	3		Stop	
12	0	3		Stop	
13	0	3		Stop	
14	0	3		Stop	
15	0	1		Stop	
16	0	1		Stop	
Load level		Priority:1(Low)–6(High)			

User Task Monitor (Load levels)

Display of load levels

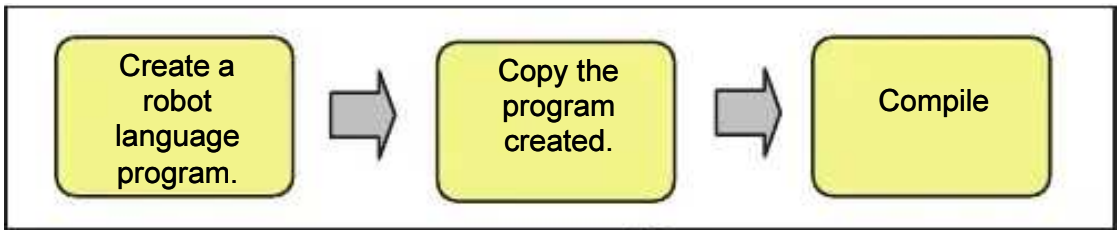
Load level	Description
Blue	A load level, which has no influence on robot motions.
Yellow	A load level, which is likely to have influence on robot motions.
Red	A load, which is highly likely to have influence on robot motions. At this load level, take actions such as giving lower priority or entering the PAUSE command in the user task program.

NOTE



# Chapter 2 Creation of User Task Program

Create a user task program in accordance with the procedure shown below, like the creation of normal robot language program.



Creation of user task program

## 2.1 Creation of Robot Language Program

Create a robot language program as shown right. Describe robot language used for user tasks using the same grammar as normal robot language. For the grammar of robot language, refer to the instruction manual “Robot Language”.

'Example of user task program

L1% = 0

\*START

WAITOUT O50

L10! = V1! - L1!

V3! = L10!/V2!

L1! = V1!

RESET O50

GOTO \*START

END

File name rules	USERTASK-A.*** (** represents a program number.) *User task programs are numbered 0 to 999.
Editing method	Refer to information in the instruction manual of personal computer or text editor in use.
File size	64 KB (65536 bytes) at maximum
Precautions for sentences, lines, and characters	254 characters per line and 999 lines per program are describable. No case sensitivity is available for alphabetic characters. Use one-byte characters for anything other than comments and string variables. It's no problem to describe the blank line.(It is ignored when compiled)
Others	Use SHIFT_JIS for two-byte characters and CR+LF for linefeed codes.

### 2.1.1 Commands unavailable for user tasks

For user task programs, there are some commands that are available in normal robot languages but not for user tasks.  
Commands listed below are unavailable for user tasks.

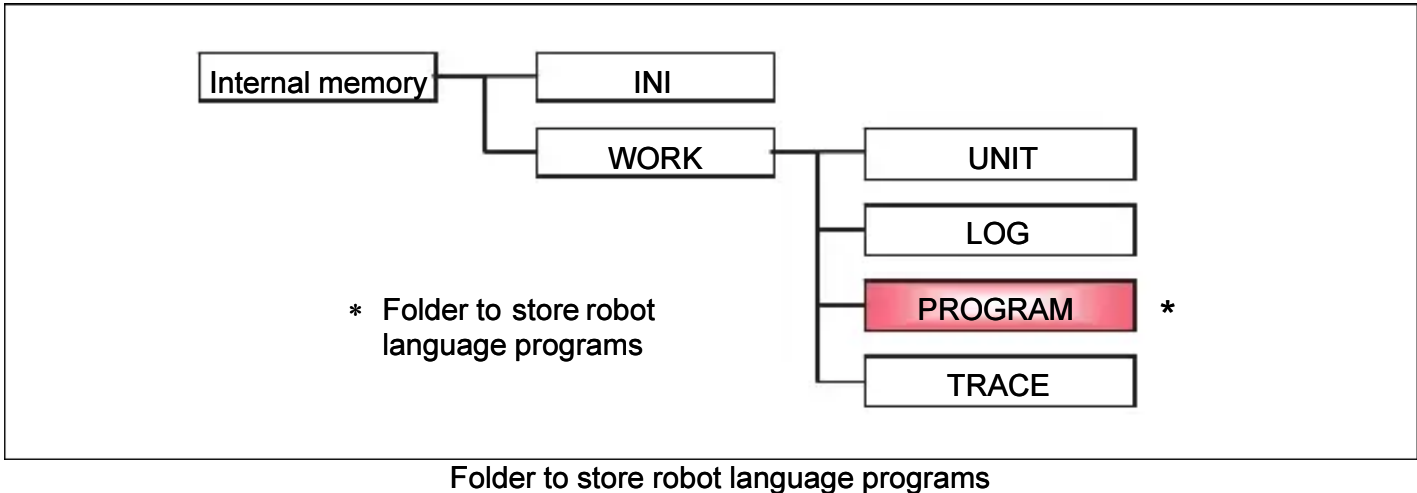
- Commands related to robot motions, including move commands such as MOVEX and posture control commands such as LEFTY
- Commands related to each application, including spot welding, arc welding, and sealing-related commands
- Robot program call commands  
Program call commands to call user tasks are available.
- Part of signal I/O commands

For each command, refer to the instruction manual “Command Reference”.

2.1.2 Copying program file created

In order to download the robot language program created by an external personal computer to this controller, prepare a USB memory or make Ethernet connection.

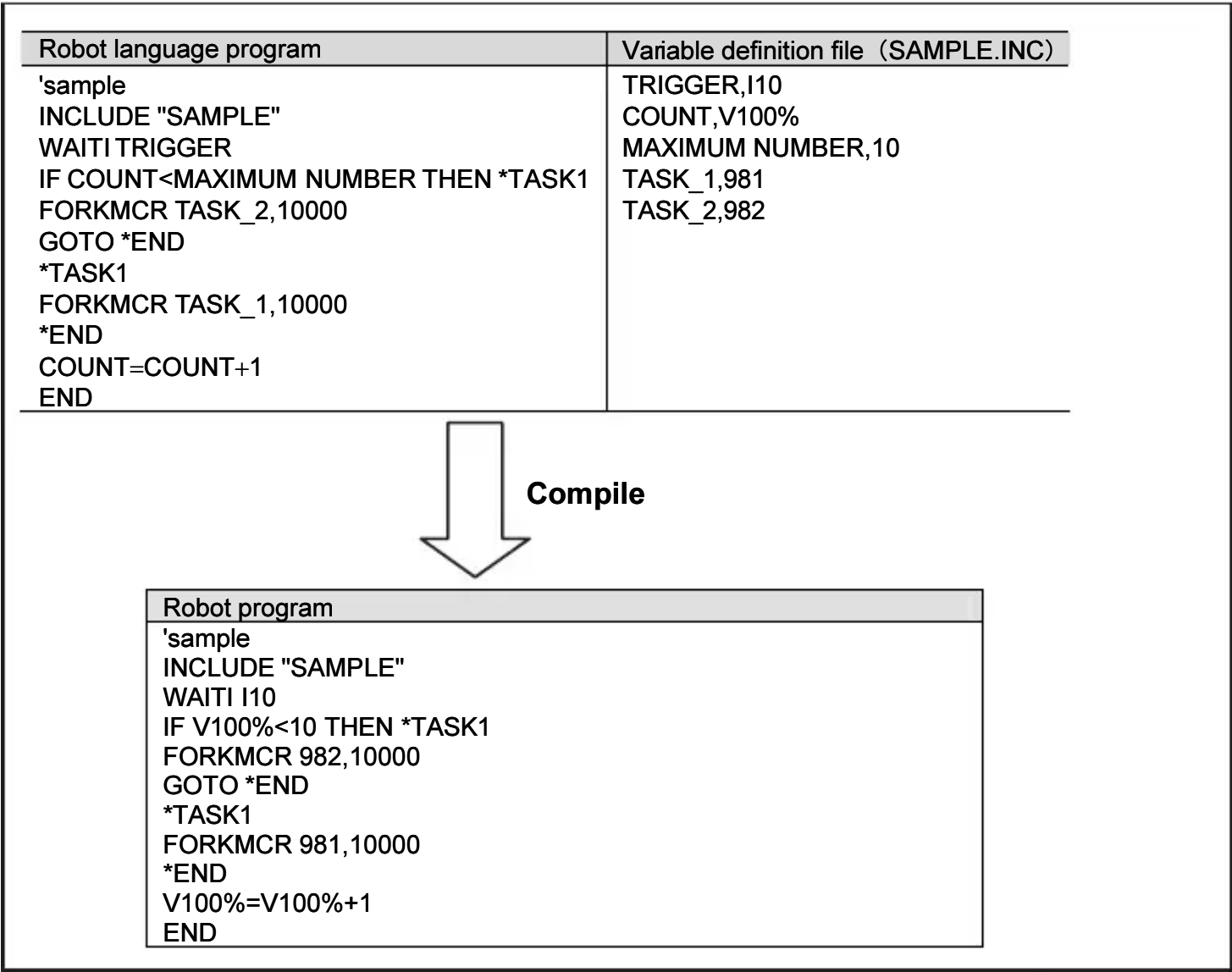
To copy the program from a USB memory, select [Service] → [4 File Manager] → [1 Copy].  
For a destination folder to write the robot language program, specify the PROGRAM folder in which work programs are stored. Robot language programs except for those stored in the PROGRAM folder are not applicable to edition or compilation.  
For the procedure for [File Manager], refer to the instruction manual “Basic Operation”.



## 2.2 Making Use of Variable Names

To create a program, variable numbers and signal numbers can be replaced with character strings using the INCLUDE command. This replacement function makes it possible to upgrade the visibility of program.

To make use of variable names, character strings and signals/variables conversion rules need to be described in the “Variable definition file”. Reading the described variable definition file using the “INCLUDE” command in the robot language program validates the conversion rules from the next line of the “INCLUDE” command.



Character strings defined by the variable definition file are replaced by compiling. Consequently, the teach pendant will display programs with already-replaced character strings.

2.2.1 Variable definition file

Describe the correspondence relation between variable number and character string to the variable definition file.

Variable definition file
TRIGGER,I10
COUNT,V100%
MAXIMUM NUMBER,100
TASK_1,981
TASK_2,982

Describe the variable definition file in CSV (comma separated value) format. Describe a conversion character string in the first column and a signal or variable number in the second column, and ignore the third and later columns.

The variable definition file has restrictions listed in the table below.

Restriction	Description
File name	Be sure to take “inc” as the file name extension, and further save files to the WORK/PROGRAM folder.
Scope	Variable numbers, signal numbers, and numerical characters can be only defined.
Table size	The number of combinations of character string and number defined by the INCLUDE file is 9999 at maximum. To read multiple INCLUDE files in the program, keep the sum of files up to 9999. If the number of combinations exceeds 9999, a compile error will occur.
Line boundary character	“,” (comma), “ ” (en quad), “” (double quotation mark), “'” (single quotation mark) and “¥” (yen mark) cannot be used in any character strings. Furthermore, the same character strings as function names cannot be used as conversion character strings. If any of the characters described above is included, the relevant line will be skipped.
Single-line size	A maximum of 254 characters can be used.
Character code	Japanese can be used for the INCLUDE file. Create the file using SHIFT JIS for two-byte characters and CR+LF for linefeed codes.

2.2.2 Functions

To make use of conversion rules described to the variable definition file and variable names from I/O names, use the INCLUDE or INCLDUEIO function.

Function name	Function
INCLUDE	Used to apply the conversion rules with an extension written to the INC file. Make setting of file name as the function parameter.
INCLUDEIO	Used to establish conversion rules from the I/O name. Variable names without set I/O name or with any line boundary character in the I/O name are not added to the conversion rules.



Using the INCLUDE function more than once makes it possible to apply the conversion rules described to the INC file. However, if the number of conversion rules exceeds 9999, a compile error will occur.

## 2.3 Commands Available for User Tasks

User tasks are limited in available commands. In addition, commands dedicated to user tasks are available. Commands available for user tasks are listed below.

Commands available for user tasks

Command name	Function
RETURN	Used to return to the next step of the GOSUB command after executing this command.
SET	Used to turn ON general-purpose output signals (O1 to O2048).
RESET	Used to turn OFF general-purpose output signals (O1 to O2048).
CALLP	Used to call a different user task program.
JMPP	Used to jump to a different user task program.
GOTO	Used to jump to a specifying line.
GOSUB	Used to jump to a specified line. This command makes it possible to return the line with the RETURN command.
END	Used to finish the playback of the program.
REM	Used to put comments in the program.
PRINT	Used to output character string data.
WAITI	Used to wait until general-purpose input signal is input.
NOP	Used to perform no processing.
*	Used to make setting of label as the reference address of line jump command.
IF	Used to execute conditional branching.
ON GOTO	Used to execute branching in accordance with integer variable values.
FOR	Used to repeatedly execute commands described between the FOR and NEXT commands.
NEXT	
WINDOW	Used to control the User screen.
TITLE	Used to change titles of the User screen.
CLS	Used to delete the display of the User screen.
LOCATE	Used to change the positions of character strings that are displayed on the User screen with the PRINT command.
GLINE	Used to draw a straight line on the User screen.
GBOX	Used to draw a rectangle on the User screen.
BARC	Used to draw an arc on the User screen.
GPAINT	Used to fill in framed areas on the User screen.
GSETP	Used to make one-dot drawing on the User screen.
COLOR	Used to make setting of color used for command to make a drawing.
BGCOLOR	Used to make setting of background color.
CLR	Used to clear local variables.
EXIT	Used to exit the user task program.
PAUSE	Used to make user task pause for a given period of time.
GARC	Used to draw an ellipse on the User screen.
LETRE	Used to assign a value to the shift register.
ADDVI	Used to add a value to obtain integer variable.
ADDVF	Used to add a value to obtain real variable.
SUBVI	Used to subtract a value to find integer variable.

Command name	Function
SUBVF	Used to subtract a value to find real variable.
MULVI	Used to multiply a value to find integer variable.
MULVF	Used to multiply a value to find real variable.
DIVVI	Used to divide a value to find integer variable.
DIVVF	Used to divide a value to find real variable.
TIMER	Used to obtain a lapse of time after turning ON power supply on the millisecond time scale.
SQR	Used to find the square root.
SIN	Used to find sin(f).
COS	Used to find cos(f).
TAN	Used to find tan(f).
ATN	Used to find atan(f).
ATN2	Used to find atan(f1/f2).
ABS	Used to find an absolute value.
MIN	Used to find smaller one of two numerical values.
MAX	Used to find larger one of two numerical values.
FORKMCR	Used to start a user task.
CALLMCR	Used to start a user task and wait for its completion.
WAITOUT	Used to wait for any general-purpose output signal to turn ON.
INCLUDE	Used to read conversion rules from the INC file.
INCLUDEIO	Used to read conversion rules from signal name.
GFONT	Used to change the size of letters displayed on the user window by PRINT command.
GSOFTKEY	Display a software-key on the user window.
GMSGBOX	Display a message box on the user window.
GETKEY	Get a key-cord inputted on the user window.
CPRIMCR	Used to set the priority of user task. ✕available on version FDV4.35 or later
AOUT	Used to output the analog voltage of TCP (tool center point) velocity and or so.

# Chapter 3 Playback of User Task Program

## 3.1 Starting Methods of User Task Program

The following four methods are available for starting the user task program. Playback condition varies with their starting method.

Starting methods and playback condition

Starting method		Playback Condition
1	Starting when turning ON power supply	Executing the END command will return the program to Step 0
2	Starting from monitor	
3	Starting from function	Executing the END command will finish playing back the user task program
4	Starting from soft key	

**POINT** “Starting when turning ON power supply” and “Starting from monitor” will start playing back the user task program continuously. Consequently, it will not end by the END command. To end the user task program by 1 cycle, use the EXIT command instead of the END command.

**IMPORTANT** After FDV04.71, the maximum number of the USER TASKs that can be used at the same time becomes from 4 to 16.  
In case of FDV4.70 or less, the maximum number is 4.  
In case of FDV4.71 or more, the maximum number is 16.

### 3.1.1 Starting when turning ON power supply

To start a user task when turning ON power supply, create a user task program for Program Number 0. If the user task program for Program Number 0 exists when turning ON the power supply, this controller will select Task 1 to start the user task program.

### 3.1.2 Starting from user task monitor

Select a Task Number on the user task monitor to start the user task.

- 1
- Select Monitor [37 User Task].  
>> The [User Task Monitor] will appear as shown below.

User Task Monitor					
	Prog	Priority	Comment	Status	Error
1	900	4	Usertask program	Stop	
2	500	3	test	Start	
3	0	3		Stop	
4	0	3		Stop	
5	0	3		Stop	
6	0	3		Stop	
7	0	3		Stop	
8	0	3		Stop	
9	0	3		Stop	
10	0	3		Stop	
11	0	3		Stop	
12	0	3		Stop	

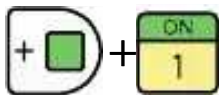


- 2
- Press [EDIT] key to set the system to Edit mode.



3 Enter a program number and priority, and then press <Complete> key.

11 User Task Monitor					
	Prog	Priority	Comment	Status	Error
1	900	4	Usertask program	Stop	
2	500	3	test	Start	
3	555	3	PROG	Stop	
4	0	3		Stop	
5	0	3		Stop	
6	0	3		Stop	
7	0	3		Stop	
8	0	3		Stop	
9	0	3		Stop	
10	0	3		Stop	
11	0	3		Stop	
12	0	3		Stop	



4 Put the cursor on the Status display corresponding to a task number that you want to start, and then press [ENABLE] and [ON] keys. The relevant user task will start up.

>> While in edit mode, no user task can start up. Be sure to push the <Complete> key and exit edit mode, and then start the user task.

11 User Task Monitor					
	Prog	Priority	Comment	Status	Error
1	900	4	Usertask program	Stop	
2	500	3	test	Start	
3	555	3	PROG	Stop	
4	0	3		Enable+ON Start	
5	0	3		Stop	
6	0	3		Stop	
7	0	3		Stop	
8	0	3		Stop	
9	0	3		Stop	
10	0	3		Stop	
11	0	3		Stop	
12	0	3		Stop	



### 3.1.3 Starting from soft key

Assigning the User Task Start button to the soft key will make it possible to start the user task. Assigning the user task to the soft key will make it possible to display status and turn ON/OFF signal with a single touch of the button.

When starting the user task using the soft key, if the user task program does not start up due to any of the following reasons, corresponding information will appear on the screen.

- No user task program exists.
- User tasks 1 to 16 are all in use. (In case of FDV4.70 or less, 1 to 4.)

#### 1 Select <Constant Setting> - [7 T/P Key], then assign following soft key 3006 : Push button (user Task)

For detail operation, please refer to the instruction manual “Basic Operation” “7 Useful Functions” “7.8 Customizing the Softkey”

- At “user task program”, set the user task program number to be started when pressing this soft key.
- Type a title of the button in the box, and then press the [ENABLE] and [Edit] keys. The software keyboard will appear.



#### 2 Press the assigned key to start the user task program.



Example of soft key assigning

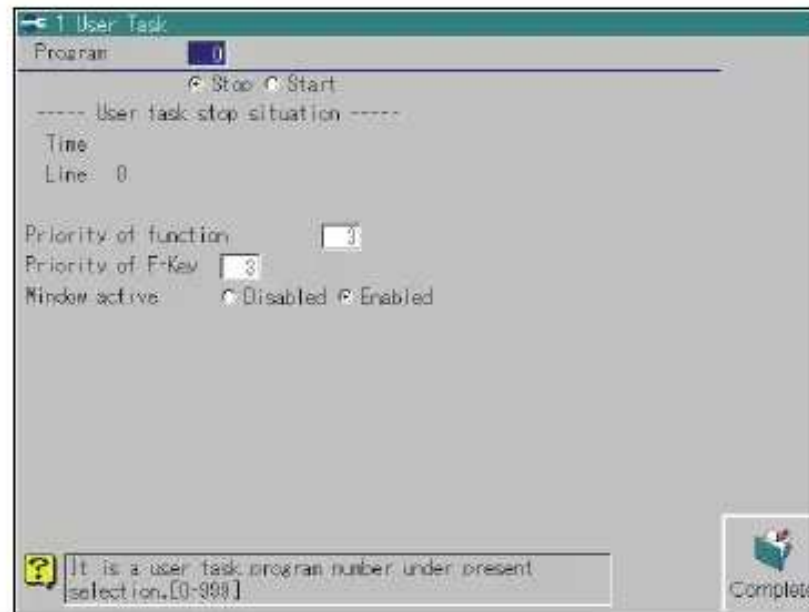
### 3.1.4 Starting from function

Using any of four functions, i.e., F670 FORKMCR, FN671 CALLMCR, FN672 FORKMCR TM, and FN673 FORKMCR DST makes it possible to start the user task in robot program.

#### Priority setting

For priorities for starting the user task from function, do not use priorities displayed on the monitor window, but make priority setting from the Service menu. After the user task starts up, the priority setting can be changed on the monitor window.

**Select <Service> → [12 User Task] → [1 User Task].**  
 >> Following screen will appear.



**2 Put the cursor on “Priority of function” and type a number of priority.**  
 Priority number is 1 to 6. Larger value means higher priority.

#### POINT

In case of the system software version FDV04.48 or before, the available priority is from 1 to 5.



**3 Press the <Complete> key.**



#### IMPORTANT

Setting to a higher priority will enable processing at a higher speed, but may affect the robot work program, thus resulting in variations in robot motions. In addition, the Teach Pendant processing may become slow. To avoid that, display the user task monitor while in playback mode to ensure that there is no problem with load levels.

#### POINT

When starting the user task from the function, the smallest user task which is stopped can be used.

#### POINT

After FDV04.71, the maximum number of the USER TASKs that can be used at the same time becomes from 4 to 16.

In case of FDV4.70 or less, the maximum number is 4.

In case of FDV4.71 or more, the maximum number is 16.

If the maximum number of USER TASKs are already running, it is impossible to start a new USER TASK using application command (FN).

The same program can be started in one time from several user tasks.

FN670 FORKMCR

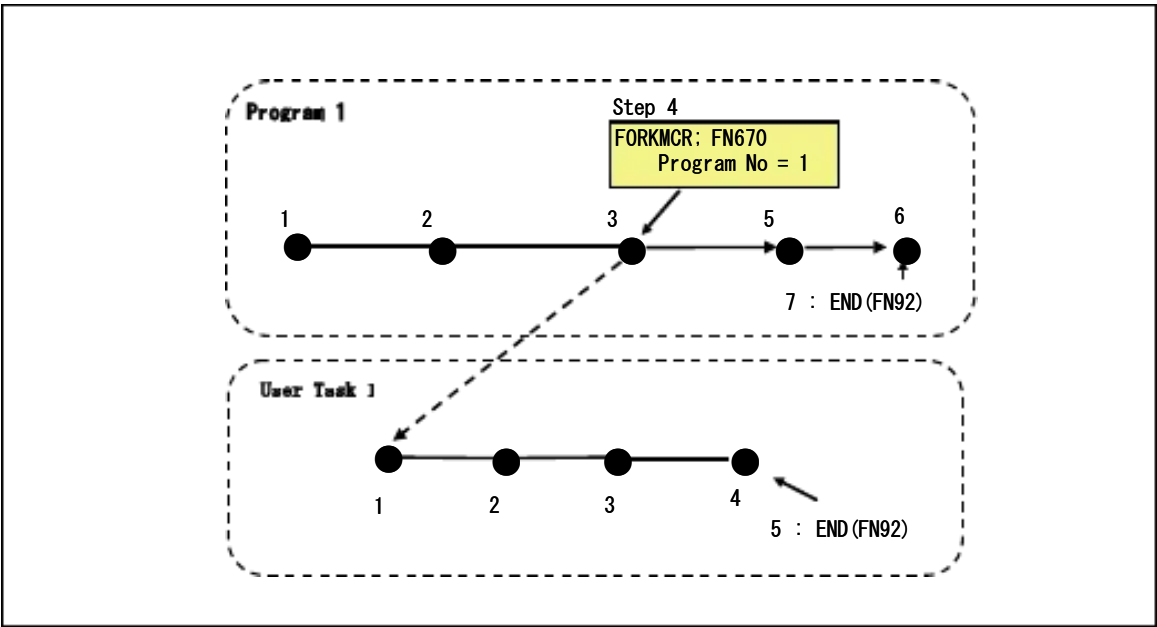
Command name	FORKMCR
FN code	670
Title name	Fork User Task Program
General description	This command is used to start the specified user task program.

■ General description

When this function command is executed, the specified user task program is started. Started user task program finishes its execution after one cycle.

■ Example of operation

The started user task program is recorded in the step 4. After the program go to the step 4, the user task program is started in the play back mode. When user task can not be started, it jumps to the step which shelter step assigned. If it is started successfully, robot program and user task program are run at the same time.



■ Parameter

Parameter No. 1	Program No.	To specify the number of the started user task program.(0-999)
Parameter No. 2	Shelter step	The number of the shelter step should be specified when the specified user task program is not started. (1 - 10000) When 10000 is specified as the shelter step number, an alarm results immediately with no escape operation performed, and the robot can be stopped.

FN671 CALLMCR

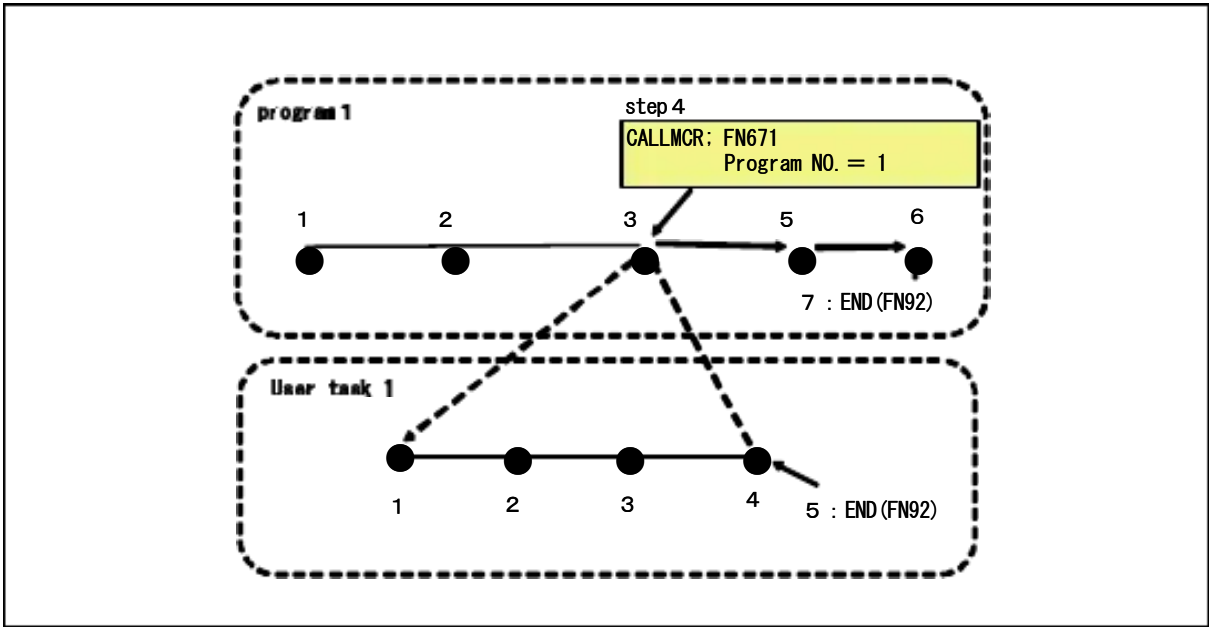
Command name	CALLMCR
FN code	671
Title name	Call User Task Program
General description	This command is used to call the specified user task program.

General description

When the user task program is called, the present program will be stopped and the user task program will be started.  
The source program will be started again after the user task program over.

Example of operation

In step 4, record CALLMCR: User Task Program Call (FN671) and "1" as the program number.  
When this is played back, the program1 will be stopped after arriving at step 4 and the user task program will be started from step 1 to step 4.  
When the playback of user task program 1 is completed (in the status established by executing the END command), the robot returns to step 5 in the source program 1.



Parameter

Parameter No. 1	Program No.	To specify the number of the started user task program.(0-999)
Parameter No. 2	Shelter step	The number of the shelter step should be specified when the specified user task program is not started. (1 to 10000) When 10000 is specified as the shelter step number, an alarm results immediately with no escape operation performed, and the robot can be stopped.

FN672 FORKMCRTM

Command name	FORKMCRTM
FN code	672
Title name	Fork User Task Program (time)
General description	This command is used to start the specified user task program. Furthermore, the command enables advance execution to be specified.

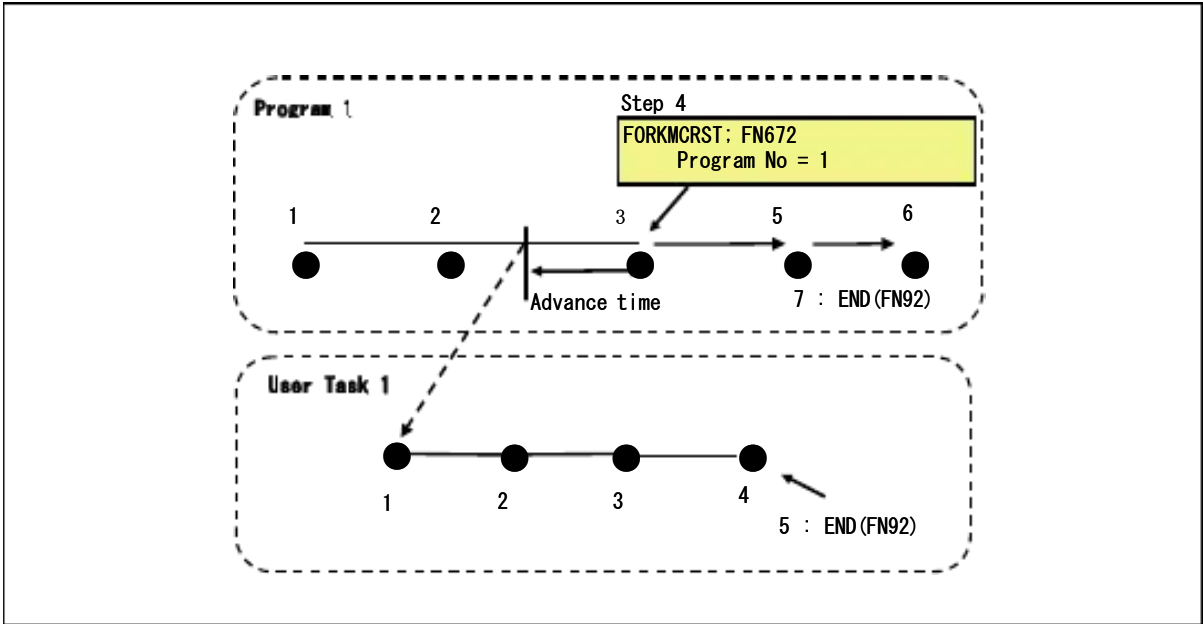
■ General description

When this function command is executed, the specified user task program will be started. Furthermore, the command enables advance execution earlier than at the position reached by the robot in the accuracy range of the move command.

■ Example of operation

In step 4, record FORKMCRTM : User Task Program Start with early execution (FN672) and "1" as the program number. When this is played back, the user task program 1 will be started before specified seconds from arriving at step 4.

When the User Task 1 finished, the program 1 return to the program 1 step 5.



If the locating is not be specified, robot will be practiced in the range of move command accuracy where it reached early.

■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (0-999)
Parameter No. 2	Advance execution time	If "0.0" is specified as the time, the command is executed at the time which coincides with the recorded point. If the setting time is minus, the command will be executed earlier according to the time which have been set. (-10.0 – 0.0)
Parameter No. 3	Shelter step	The number of the shelter step should be specified when the specified user task program is not started. (1 to 10000) When 10000 is specified as the shelter step number, an alarm results immediately with no escape operation performed, and the robot can be stopped.

FN673 FORKMCRDST

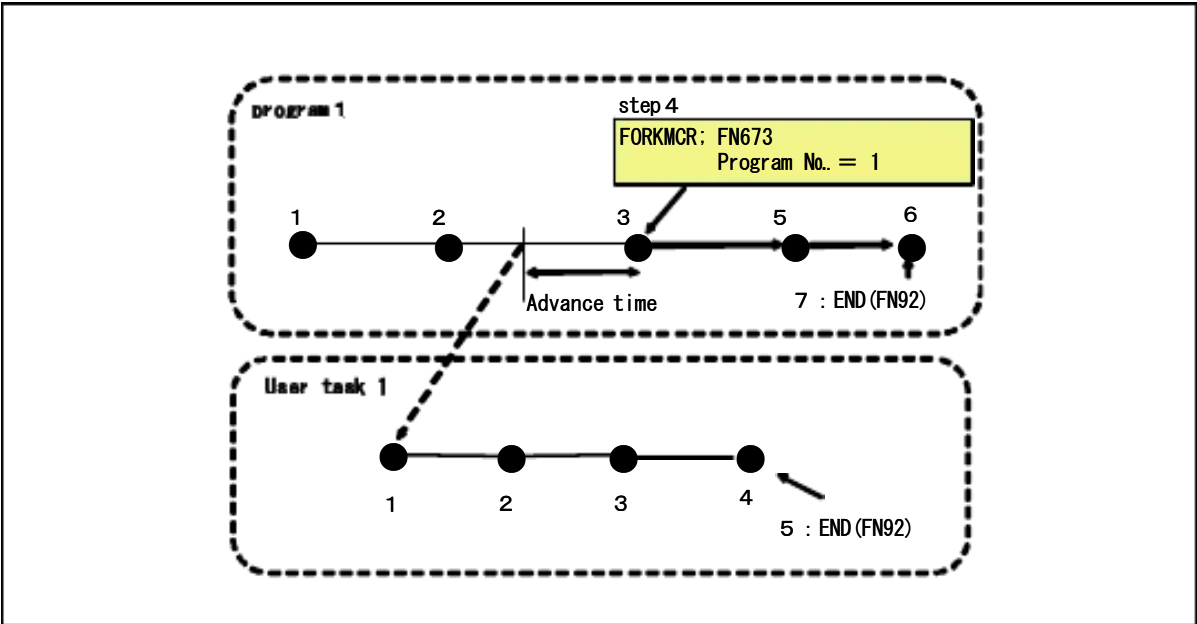
Command name	FORKMCRDST
FN code	673
Title name	Fork User Task Program (distance)
General description	This command is used to start the specified user task program with advancing distance.

General description

When this function command is executed, the specified user task program will be started. Furthermore, the command enables advance execution earlier than at the position reached by the robot in the accuracy range of the move command.

Example of operation

In step 4, record FORKMCRDST : User Task Program Start with early execution (FN673) and "1" as the program number. When this is played back, the user task program 1 will be started before specified seconds from arriving at step 4.



If the locating is not be specified, robot will be practiced in the range of move command accuracy where it reached early.

Parameter

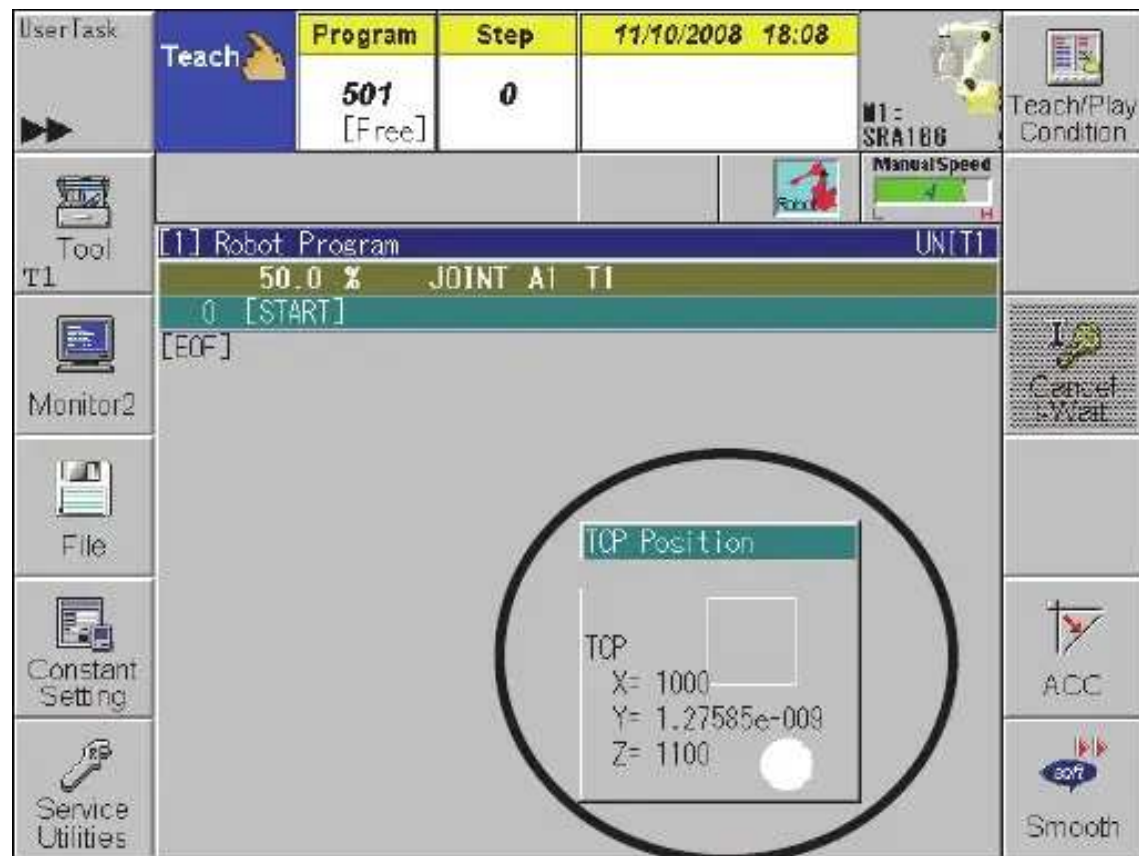
Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (0-999)
Parameter No. 2	Advance execution timing	If "0.0" is specified as the time, the command is executed at the time which coincides with the recorded point. If the setting time is minus, the command will be executed earlier according to the time which have been set. (-10.0 – 0.0)
Parameter No. 3	Shelter step	The number of the shelter step should be specified when the specified user task program is not started. (1 to 10000) When 10000 is specified as the shelter step number, an alarm results immediately with no escape operation performed, and the robot can be stopped.

# Chapter 4 User Screen

## 4.1 User Screen

The User screen is the function of displaying windows from the user task program. Changing the program makes it possible to form and delete windows, display character strings, and display circles without any inhibition.

By creating a user task program used to display information that you want (e.g. TCP position) in advance and assigning the program to the soft key, necessary information can be displayed in a desired format.



Example of User screen display

4.1.1 How to display the User screen

- Following two methods are prepared to display the user screen.
- 1. Always displaying the user screen. (But screen disappears after some key operation.)
  - 2. Opening the user screen from the monitor by pressing [Close / Select Screen] key. (Soft key and key operation is available.)

- 1** Open <Service Utilities> - [12 User Task]→[1 User Task]  
» Following screen will appear.



- 2** Align the cursor to “Window Active”, and press [ENABLE] and the [Left] or [Right] cursor key at the same time to make the selection  
Disabled ; To display the user screen always.  
Enabled ; To display the user screen by pressing [Close / Select Screen] key or by pressing [ENABLE] and [I/F] key at the same time.



- 4** Upon completion of the settings, press <Complete> key.



If soft key or message box or key operation is needed in the user screen, “Window active” must be set to “Enabled”.



## 4.2 User Screen Draw Commands

### 4.2.1 User screen display

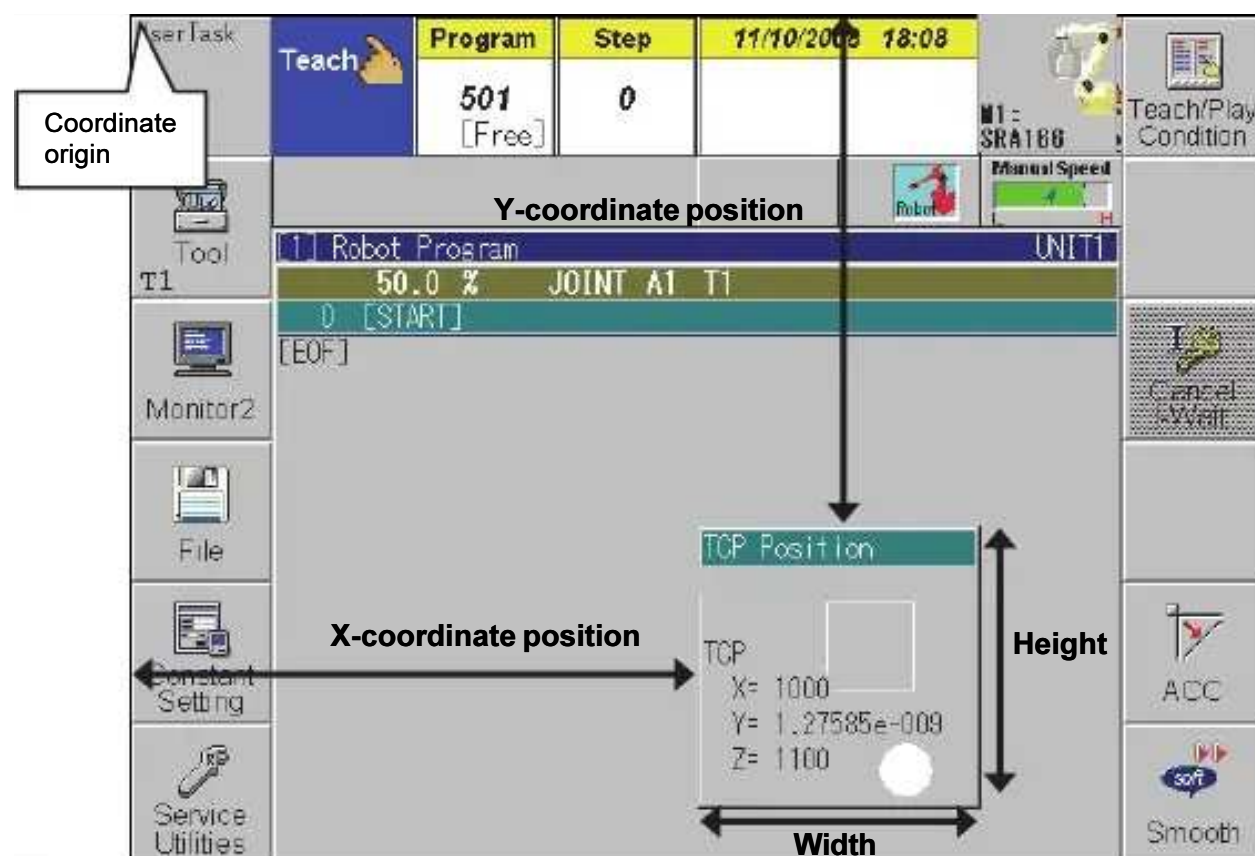
The User screen is displayed in any position. To display the User screen, use the WINDOW command. To change the background color of the User screen, use the BGCOLOR command.

**WINDOW:** X-coordinate position, Y-coordinate position, User screen width, and User screen height

Parameter	Description
X-coordinate position	Used to specify a distance from the left of the Teach Pendant screen to the left side of the User screen in units of dot.
Y-coordinate position	Used to specify a distance from the top of the Teach Pendant screen to the upper hem of the User screen in units of dot.
User screen width	Used to specify the horizontal width of the User screen.
User screen height	Used to specify the height of the User screen including the title bar.

Four user tasks can be played back at one time. One user task can display one user screen. In addition, setting the width and height of the User screen to "0" makes it possible to close the User screen.

All character strings and figures that are described later are displayed on the windows formed in this Section.



Definition of each value

POINT

X coordinate position and Y coordinate position can be aligned every 16 dots. For example, when X coordinate position is designated 30 and Y coordinate is designated 20, X coordinate position is displayed at 16 and Y coordinate position is displayed at 16.

**BGCOLOR “Back ground color”**

This command is used to set the back ground color of user screen.

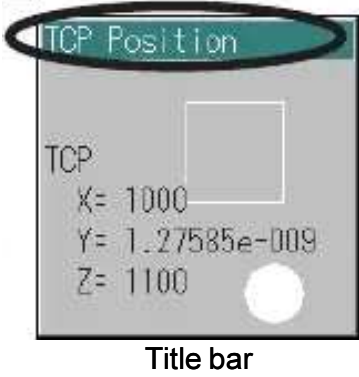
Parameter	Description																																		
Back ground color	Back ground color of user screen is set by 0 to 15. <table><tr><th>No.</th><th>Color</th></tr><tr><td>0</td><td>Black</td></tr><tr><td>1</td><td>Gray</td></tr><tr><td>2</td><td>Dark Blue</td></tr><tr><td>3</td><td>Blue</td></tr><tr><td>4</td><td>Dark Green</td></tr><tr><td>5</td><td>Green</td></tr><tr><td>6</td><td>Dark Light Blue</td></tr><tr><td>7</td><td>Light Blue</td></tr><tr><td>8</td><td>Dark Red</td></tr><tr><td>9</td><td>Red</td></tr><tr><td>10</td><td>Purple</td></tr><tr><td>11</td><td>Pink</td></tr><tr><td>12</td><td>Dark Yellow</td></tr><tr><td>13</td><td>Yellow</td></tr><tr><td>14</td><td>Light Gray</td></tr><tr><td>15</td><td>White</td></tr></table>	No.	Color	0	Black	1	Gray	2	Dark Blue	3	Blue	4	Dark Green	5	Green	6	Dark Light Blue	7	Light Blue	8	Dark Red	9	Red	10	Purple	11	Pink	12	Dark Yellow	13	Yellow	14	Light Gray	15	White
No.	Color																																		
0	Black																																		
1	Gray																																		
2	Dark Blue																																		
3	Blue																																		
4	Dark Green																																		
5	Green																																		
6	Dark Light Blue																																		
7	Light Blue																																		
8	Dark Red																																		
9	Red																																		
10	Purple																																		
11	Pink																																		
12	Dark Yellow																																		
13	Yellow																																		
14	Light Gray																																		
15	White																																		

**4.2.2 User screen title setting**

Make setting of a character string in the title bar on the User screen. To make setting of title character string, use the TITLE command.

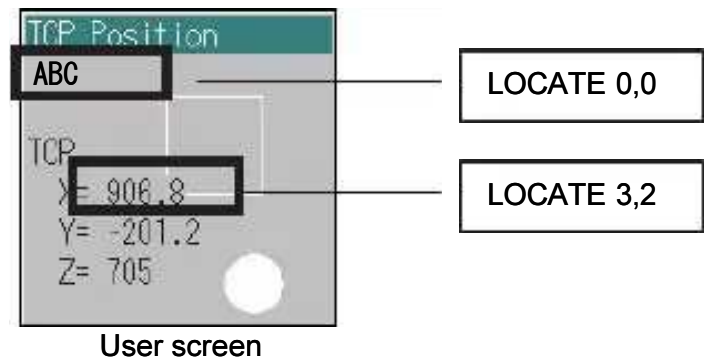
**TITLE: “Title character string”**

Parameter	Description
Title character string	Used to make setting of character string displayed in the title bar on the User screen. Character string variables can be accepted as the character string.



4.2.3 Character string display

To display character strings like those show below, use the LOCATE command, PRINT command, and COLOR command.



LOCATE: Line and Digit

This command is used to make setting of the display positions of character strings to be displayed. Use the Line parameter to specify longitudinal display positions and the Digit parameter to specify lateral display positions.

Parameter	Description
Line	Used to make setting of longitudinal display positions taking the upper left of the screen (not including the title bar) as the origin.
Digit	Used to make setting of lateral display positions taking the upper left of the screen (not including the title bar) as the origin. Adding one digit shifts the display positions of character strings for a one-byte character to the right.



Line is every 20 dots, and Digit is every 8 dots.

PRINT: Output port and “Display character string”

This command is used to display character strings in the positions specified by the LOCATE command. If the LOCATE command specifies no display position, the first character string will be displayed in the position of “0” digit in “0” line, and the second or later character strings will be displayed in the next line of previously-output character string.

Parameter	Description
Output port	Used to make setting of output destination of character strings. To use this parameter on the User screen, set it to “#0”.
Display character string	Used to specify character strings to be displayed.

COLOR: Color number

This command is used to make setting of colors of characters to be output. In addition to colors of characters, the command makes setting of colors of lines, circles, rectangles, and others.

Parameter	Description
Color number	Used to make setting of colors in 0 to15 color numbers. The correspondence between the number and color is the same as that for the BGCOLOR command.

GFONT: Height, Width, Bold, Italic, Underline

This command is used to set the height, width, bold, italic, underline of character.

Parameter	Description
Height	Used to set the height of character. When set to “0”, it means current height.
Width	Used to set the width of character. When set to “0”, it means current width.
Bold	0: Disabled 1: Enabled (Bold)
Italic	0: Disabled 1: Enabled (Italic)
Underline	0: Disabled 1: Enabled (Underline)

### 4.2.4 Drawing figures

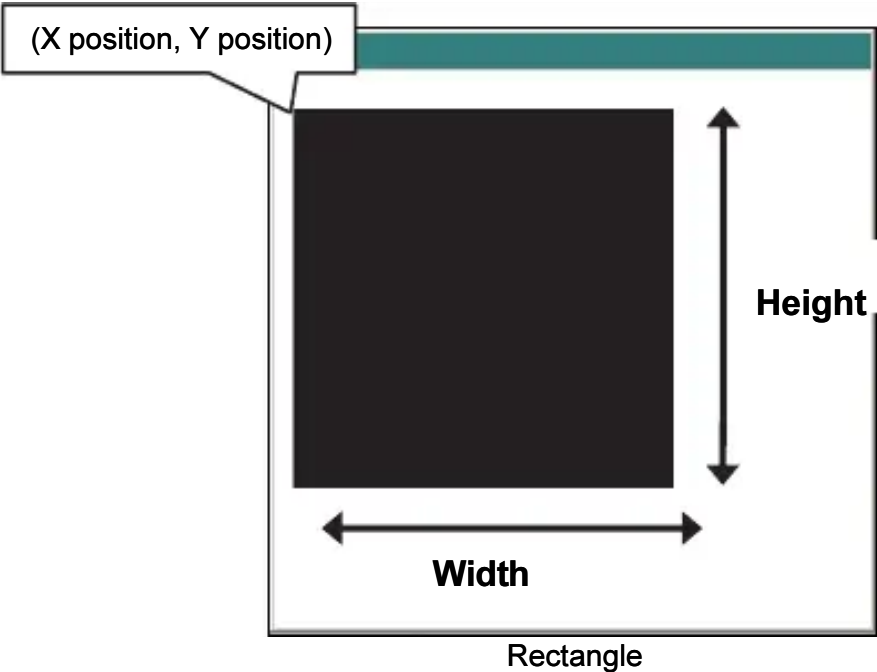
To display figures on the User screen, use the commands shown below.

- Drawing rectangle

**GBOX: X position, Y position, Width, Height, and Fill**

This command is used to display a rectangle on the User screen. If the width and height of rectangle substantially runs off the User screen, the User screen will display only the portion of the rectangle that can be displayed on it.

Parameter	Description
X position	Used to specify a distance from the left of the User screen to the left side of the rectangle in units of dot.
Y position	Used to specify a distance from the top of the User screen (not including the title bar) to the upper hem of the rectangle in units of dot.
Width	Used to specify the horizontal width of the rectangle.
Height	Used to specify the height of the rectangle.
Fill	Used to select whether or not to fill the rectangle. (0: Not fill / 1: Fill)

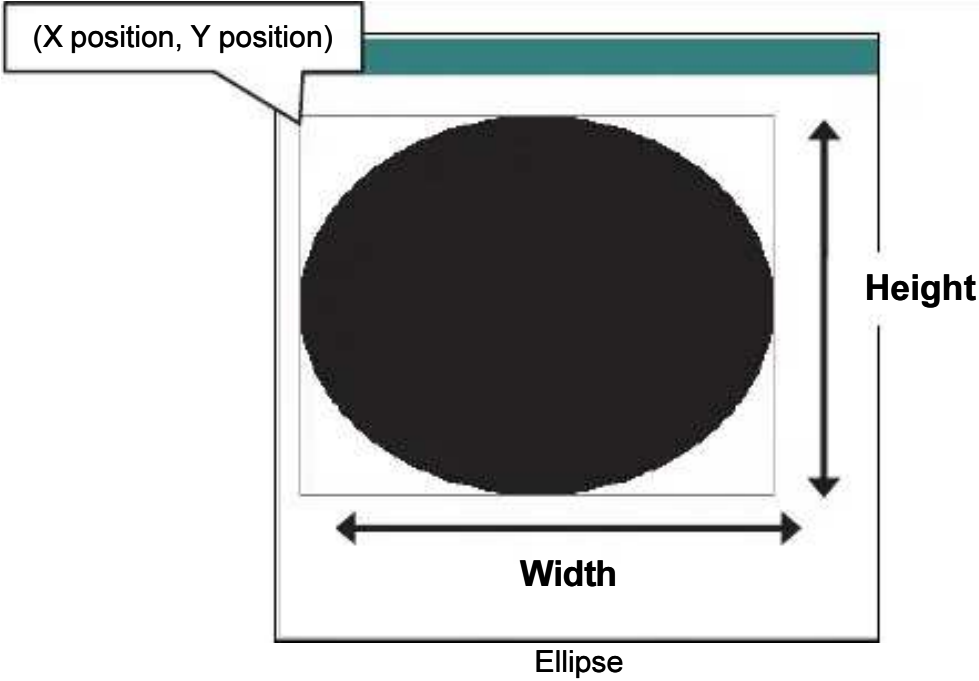


● Drawing ellipse

**GARC: X position, Y position, Width, Height, and Fill**

This command is used to display an ellipse, which is inscribed in a rectangle drawn by the X position, Y position, Width, and Height parameters.

Parameter	Description
X position	Used to specify a distance from the left of the User screen to the left side of the rectangle in units of dot.
Y position	Used to specify a distance from the top of the User screen (not including the title bar) to the upper hem of the rectangle in units of dot.
Width	Used to specify the horizontal width of the rectangle.
Height	Used to specify the height of the rectangle.
Fill	Used to select whether or not to fill the rectangle. (0: Not fill / 1: Fill)

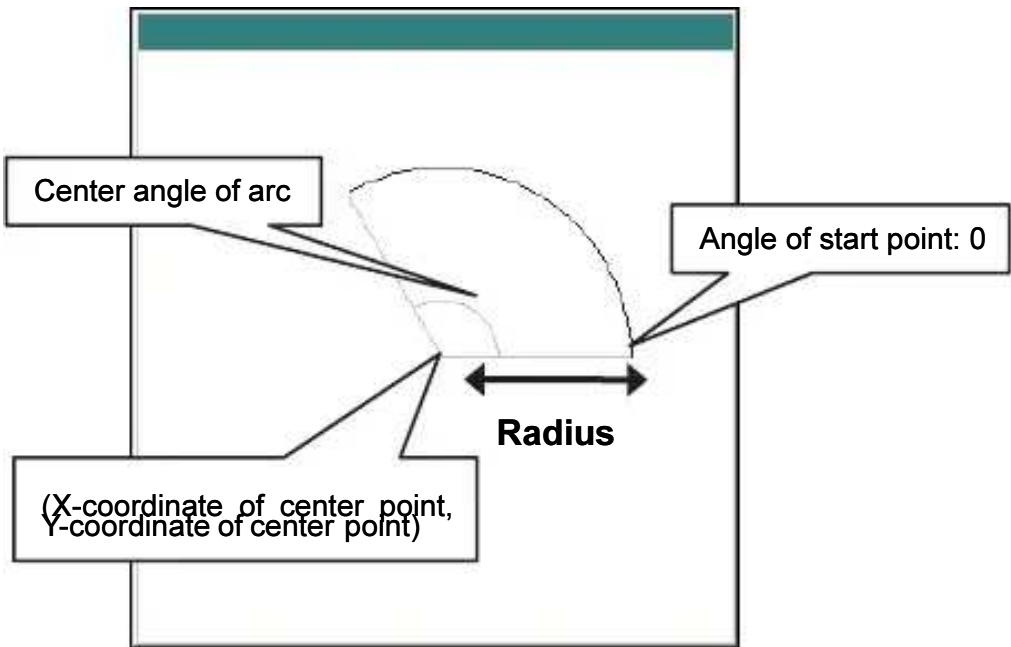


● Drawing arc

**BARC:** X-coordinate of center point, Y-coordinate of center point, Radius, Angle of start point, and center angle of arc

This command is used to display an arc on the User screen.

Parameter	Description
X-coordinate of center point	Used to specify the X-coordinate of arc center taking the upper left of the User screen (not including the title bar) as the origin.
Y-coordinate of center point	Used to specify the Y-coordinate of arc center taking the upper left of the User screen (not including the title bar) as the origin.
Radius	Used to specify the radius of arc.
Angle of start point	Used to specify the start point of arc. The arc is displayed in a counterclockwise direction from the start point.
Center angle of arc	Used to specify the center angle of arc.



● Fill

**GPAINT:** X-coordinate, Y-coordinate, and Boundary color

This command is used to fill a figure drawn on the User screen in colors of numbers set with the COLOR command. Assuming that points specified by the X-coordinate and Y-coordinate parameters are enclosed by the boundary color, the GPAINT command fills up to areas of boundary color.

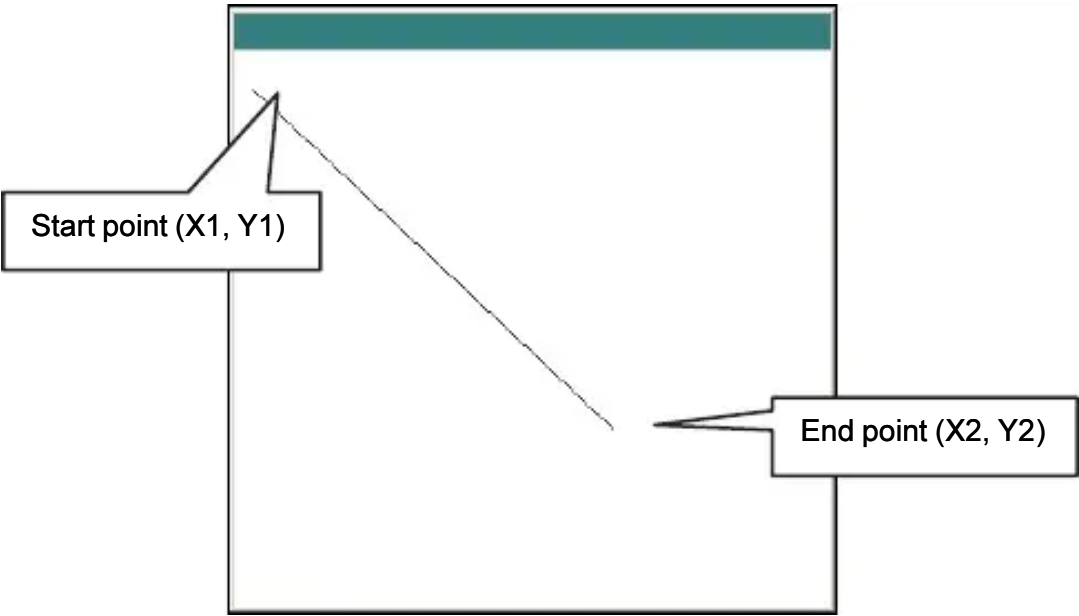
Parameter	Description
X-coordinate	Used to specify the X-coordinate of a point to start fill.
Y-coordinate	Used to specify the Y-coordinate of a point to start fill.
Boundary color	Used to specify a boundary color of fill.

● Drawing line

**GLINE:** X-coordinate of start point, Y-coordinate of start point, X-coordinate of end point, and Y-coordinate of end point

This command is used to draw a line from a start point to an end point on the User screen.

Parameter	Description
X-coordinate of start point	Used to specify the X-coordinate of start point of a line taking the upper left of the User screen (not including the title bar) as the origin.
Y-coordinate of start point	Used to specify the Y-coordinate of start point of a line taking the upper left of the User screen (not including the title bar) as the origin.
X-coordinate of end point	Used to specify the X-coordinate of end point of a line taking the upper left of the User screen (not including the title bar) as the origin.
Y-coordinate of end point	Used to specify the Y-coordinate of end point of a line taking the upper left of the User screen (not including the title bar) as the origin.



● Drawing dots

**GSETP:** X-coordinate and Y-coordinate

This command is used to draw dots in colors specified by the COLOR command in positions specified by the X-coordinate and Y-coordinate on the User screen. Use this command to draw a complicated figure that cannot be drawn in a circle or rectangle.

Parameter	Description
X-coordinate	Use to specify an X-coordinate value taking the upper left of the User screen (not including the title bar) as the origin.
Y-coordinate	Use to specify a Y-coordinate value taking the upper left of the User screen (not including the title bar) as the origin.

● Clearing the User screen

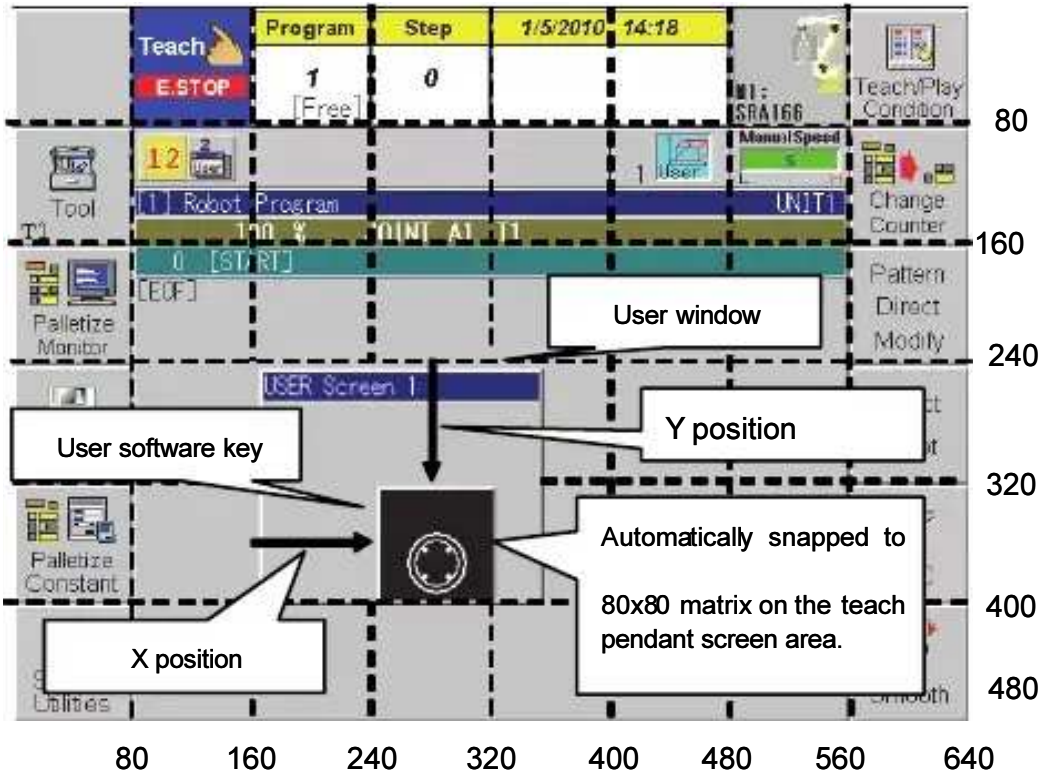
**CLS**

This command is used to clear characters and figures displayed on the User screen. Clearing the screen will put the User screen into the state in which it is filled by the color specified by the BGCOLOR command.

4.2.5 Drawing software keys

To display software keys on the User screen.

Following a file of "AC00USK.INC"(user software key definition file) in which the software key definition is written, software -keys can be displayed on the user screen via this command. This file can be created/edited in the [Service Utilities] [12 User Task] [2 User Task Softkey] screen.  
The actual coordinates (x,y) of the displayed User software key is adjusted automatically to a multiple number of 80.  
The function of the software key displayed on the user screen is the same with that of F-key and the touch screen. Pressing the software key on the user screen does not have effect to the playback of the user task.



**GSOFTKEY** X position, Y position, software-key number

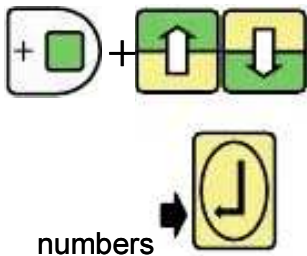
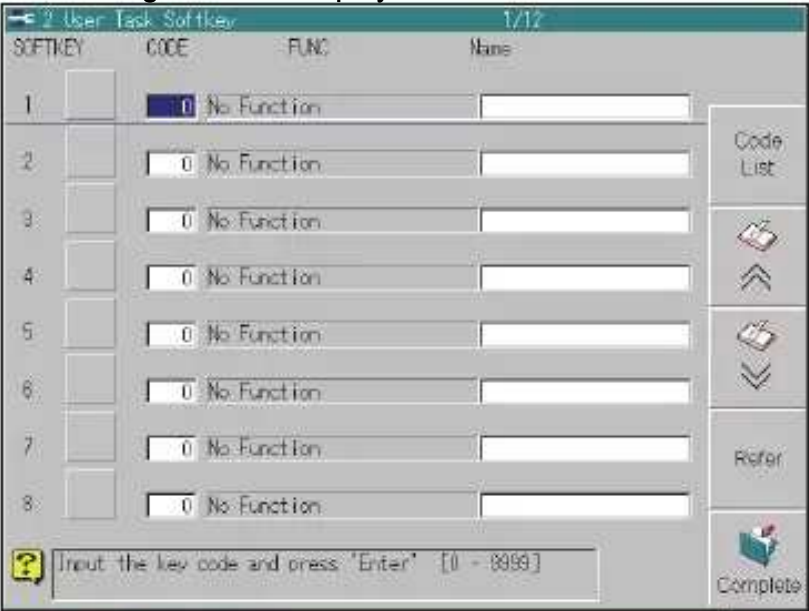
Parameter	Description
X position	Designate the distance from the left edge of the user screen to the left edge of the software-key. (unit : pixels)
Y position	Designate the distance from the upper edge of the user screen to the upper edge of the software-key. (unit : pixels)
Software-key number	Designate the software-key number to be displayed on the screen.



Software key setting

It is necessary to make a setting for the software-key beforehand.

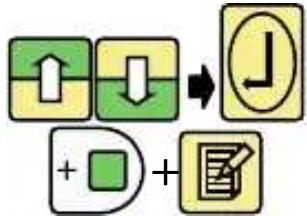
- 1 Select <Service Utilities> - [12 User Task]→[2 User Task Softkey].  
»Following screen is displayed.



- 2 It is possible to switch the list of software key 1 to software key 96 with [Enable] + [Up] / [Down] cursor keys.
- 3 Set the cursor to "CODE" and input the type number (lamp/push button/function key etc.) of the operation key to place on the position.  
Concerning the type numbers, refer to "List of operation switch keys".  
"0" stands for "NO FUNCTION" (Only blank key is displayed)  
»When the type number is set, the function title is displayed on the right side.



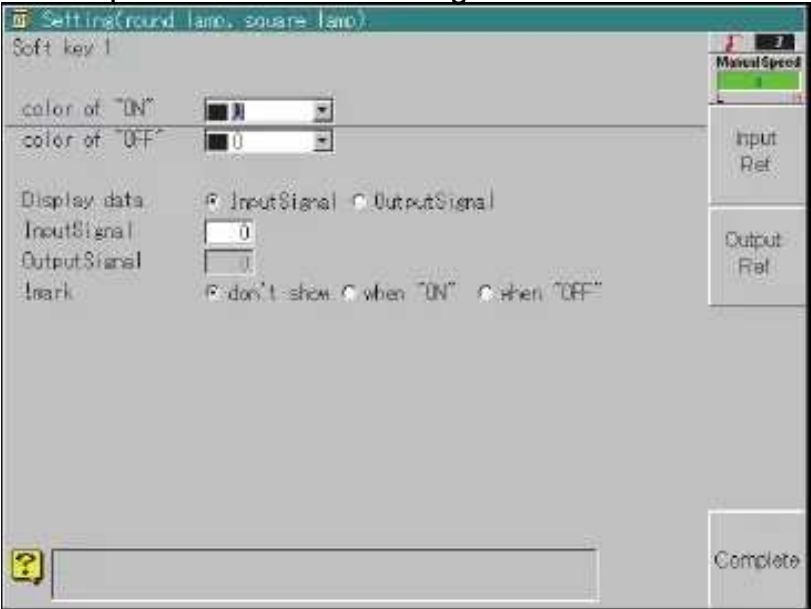
- 4 If the code number to be inputted is not clear, press <Code List> key.  
»The code list is displayed on the screen.



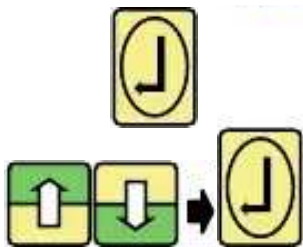
- Select a desired item with cursor key up/down and press [Enter] to enter the code into the edit box. To return to the previous screen, press [RESET R] key.
- 5 Set the cursor to "NAME" and input the name to be displayed.  
A software keyboard will be started with [ENABLE] + [EDIT] keys.  
»The length of the name which can be registered is up to 34 bytes.



- 6 Next, set the data for the details other than the names and font size.  
Align the cursor with the target line, and press <Refer> key.  
>>The operation switch detail setting screen such as the one shown below now appears.



On this screen, set the color, input/output signals and other detail data.  
Above example presents a round lamp is shown. The contents of the setting screen will differ depending on the type of operation switch concerned.



- 7 Align the cursor with "Color", and press the [Enter] key.  
Sixteen color options are displayed. Select the desired color using the [Up] or [Down] cursor key, and then press the [Enter] key.  
In the case of a round lamp, for example, select one color to indicate ON and another color to indicate OFF.

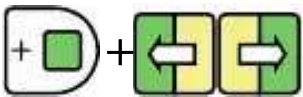
List of colors

Color no.	Color name	Color no.	Color name
0	Black	8	Medium gray
1	Dark red	9	Red
2	Dark green	10	Green
3	Dark yellow	11	Yellow
4	Dark blue	12	Blue
5	Dark magenta	13	Magenta
6	Dark cyan	14	Cyan
7	Light gray	15	White



- 8 Next, set the input and output signals.  
Align the cursor with "Input Signal" or "Output Signal", and input the number of the signal.

As a general rule, allocate the logical input signals for the lamp, display etc. and allocate the logical output signals for the push button, selector switches and other interface panel outputs.  
With the interface panel inputs, first specify the input sources using "Display data", and then set the numbers of the logical input (or output) signals.



- 9 Align the cursor with the last "!" mark," and specify the ! mark display method.  
If, for instance, an ON position status is a normal operation status, by indicating the "!" mark for the reverse OFF position, the operator can identify at a glance which switches have settings that differ from the normal status.

Settings	Details
don't show	The ! mark is not shown.
when "ON"	The ! mark appears when the input/output signal concerned is in the ON status.
when "OFF"	The ! mark appears when the input/output signal concerned is in the OFF status.



Mark shown for an interface panel input



- 10

Upon completion of the settings, press <Complete> key.

» Operation returns to the list display screen in step 2.

The contents set on the detailed setting screen have already been reflected, but at this point they have not yet been saved in a file.







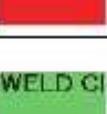






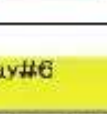
- 11

In the same procedures, make settings for the keys to be used.
- 12

Upon completion of the settings, press <Complete> key.

The settings are saved in the user software key definition file "AC00USK.INC".





List of operation switch keys

Code	Description of function	External appearance	Outline
3000	Round lamp		ON/OFF display lamp These lamps display the ON/OFF statuses of input signals or output signals.
3001	Square lamp		ON/OFF display lamp These lamps display the ON/OFF statuses of input signals or output signals.
3002	Push Buttons (alternation type)		Push button switch (operation holding type) These buttons turn the output signals ON or OFF when operated by the operator.
3003	Push Button (momentary type)		Push button switch (operation non-holding type) These buttons turn the output signals ON or OFF when operated by the operator.
3004	Flash Push Button (alternation type)		Push button switch (operation holding type) + square lamp These buttons turn the output signals ON or OFF when operated by the operator, and display the ON or OFF statuses of the input signals or output signals.
3005	Flash Push Button (momentary type)		Push button switch (operation non-holding type) + square lamp These buttons turn the output signals ON or OFF when operated by the operator, and display the ON or OFF statuses of the input signals or output signals.
3010	Selector switch (ON/OFF)		Output signal ON/OFF selector switch These buttons turn the output signals ON or OFF when operated by the operator.
3011	Contacts switch		2 output signal ON/OFF selector switch These switches turn the output signals ON only when their contacts are selected. These buttons turn the output signals ON or OFF when operated by the operator.
3020	Displays (3 digits)		This displays input signals or variable integers. Display range: 0 to 999 These displays indicate the statuses of the input signals, output signals or variable integers using 3 digits.
3021	Displays (6 digits)		This displays input signals or variable integers. Display range: 0 to 999999 These displays indicate the statuses of the input signals, output signals or variable integers using 6 digits. (They use two spaces side by side.)
3030	Input 3digits		This reflects the values input by the operator in the output signals. Input range: 0 to 999 These buttons turn the output signals ON or OFF when operated by the operator.
3031	Input 6digits		This reflects the values input by the operator in the output signals. Input range: 0 to 999999 These buttons turn the output signals ON or OFF when operated by the operator. (They use two spaces side by side.)



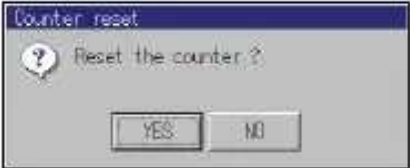


4.2.6 Drawing a message box

This command displays a message box on the User screen.

**GMSGBOX DisplayForm, Icon, Title, Text, Default, Button1, Button2, Button3, Button4**

Parameter	Description
DisplayForm	Button type setting for the message box.  0 : [OK] button is displayed. 1 : [OK] and [CANCEL] are displayed. 2 : [YES] and [NO] are displayed. 3 : [YES], [NO], and [CANCEL] are displayed. 4 : The parameters of from Button1 to Button4 are used for the text of the button.
Icon	Icon type setting for the message box. 0 : No Icon is displayed.  <div>1 :  2 :  3 :  4 : </div>
Title	The string of the title bar is specified. The sting can be specified using string variable.
Text	The string of the text is specified. The sting can be specified using string variable.
Default	The default button selection is specified. (from 1 to 4)
Button1	The string displayed on button 1 is specified when "4" is designated for DisplayForm. The string can be specified using string variable.
Button2	The string displayed on button 2 is specified when "4" is designated for DisplayForm. The string can be specified using string variable.
Button3	The string displayed on button 3 is specified when "4" is designated for DisplayForm. The string can be specified using string variable.
Button4	The string displayed on button 4 is specified when "4" is designated for DisplayForm. The string can be specified using string variable.

Example of message box

<div>DisplayForm = 0 Icon = 0</div> 	<div>DisplayForm = 1 Icon = 1</div> 	<div>DisplayForm = 2 Icon = 2</div> 
<div>DisplayForm = 3 Icon = 3</div> 	<div>DisplayForm = 4 Icon = 4</div> 	

4.2.7 Getting the information of the key operation

This is a command to get the key-code when the operation keys are pressed on the user screen.  
This command is used for the operation on the message dialog window.

String variable = GETKEY\$("Get key method" type number)

Parameter	Description
Get key method	Designate the key code getting method.  0 : The key-codes inputted from the timing when the user screen is activated and to the timing when the GETKEY\$ is executed. 1: Wait until any key or touch panel is pressed and get the pressed key cord. 2: Wait until the numeric keys, [Enter] key, [R RESET] key, or [BS] key is pressed, and get the pressed key-code. 3: Wait until the cursor keys(Up/Down/Right/Left), [Enter] key, or [RESET R] key is pressed, and get the pressed key-code. 4: Wait until the button on the message box is pressed and get the number of the pressed button.

POINT

By changing the key-cord getting method depending on the screen type, it becomes easier to identify the key-cord. (The redundant process for the unnecessary key codes becomes unnecessary)

- In case of numeric number input screen, use "2".
- In case of menu selection screen, use "3".
- In case of message box, use "4".

In case of a message box;

```
L1$ = "RESET CONFIRMATION"           ' TITLE
L2$ = "Palletize counter will be reset. Are you sure?" ' TEXT
MSGBOX 2, 2, L1$,L2$, 2,"","","","" ' [YES] / [NO] confirmation
L10$ = GETKEY$(4)                     ' Wait for the button is pressed
L10% = ORD(L10$)                     ' Converting the key-code to a number

IF (&H01 = L10%)                     ' [YES] was selected

: Reset process

ENDIF
```

Key cord list

Key operation	Key cord		
	Pressed	Pressed with ENABLE key	Released
F1	&H96	&H86	—
F2	&H95	&H85	—
F3	&H94	&H84	—
F4	&H93	&H83	—
F5	&H92	&H82	—
F6	&H91	&H81	—
F7	&H9C	&H8C	—
F8	&H9B	&H8B	—
F9	&H9A	&H8A	—
F10	&H99	&H89	—
F11	&H98	&H88	—
F12	&H97	&H87	—
ENABLE	&H1B	—	&HDB
UNIT / MECHANISM SYNCHRONIZE	&H2C &H51	&H8D &H59	—
INTERP / COORD (COORDINATE)	&H67	&HD9	—
CHECK SPEED / TEACH SPEED	&H68	&HDA	—



STOP / CONT (CONTINUOUS)	&H69	&HDC	—
CLOSE / SCREEN SELECT	&H80	&H90	—
X-	&H73	—	&HF3
X+	&H74	—	&HF4
Y-	&H75	—	&HF5
Y+	&H76	—	&HF6
Z-	&H77	—	&HF7
Z+	&H78	—	&HF8
RX-	&H79	—	&HF9
RX+	&H7A	—	&HFA
RY-	&H7B	—	&HFB
RY+	&H7C	—	&HFC
RZ-	&H7D	—	&HFD
RZ+	&H7E	—	&HFE
CHECK GO	&H57	&H71	&HD7
CHECK BACK	&H58	&H72	&HD8
OUT	&H46	&H46	—
IN	&H4D	—	—
SPD	&H49	&H9E	—
ACC	&H54	&H54	—
TIMER	&H21	&H25	—
0	&H30	&H2b	—
1	&H31	&H11	—
2	&H32	&H12	—
3	&H33	&H13	—
4	&H34	&H14	—
5	&H35	&H15	—
6	&H36	&H16	—
7	&H37	&H17	—
8	&H38	&H18	—
9	&H39	&H19	—
.	&H2E	&H2D	—
O.WRITE / REC	&H3B	&H0B	—
INS (INSERT)	&H0F	&H3D	—
CLAMP ARC	&H3A	&H4A	—
MOD POSITION	&H0E	&H3C	—
HELP	&H3F	—	—
DEL (DELETE)	&H10	&H3E	—
RESET R	&H52	&H5A	—
PROG / STEP	&H53	&H50	—
ENTER	&H0D	&H0A	—
BS	&H08	&H09	—
FN	&H22	&H26	—
EDIT	&H23	&H27	—
I/F	&H24	&H28	—
Cursor UP	&H41	&H61	—
Cursor DOWN	&H42	&H62	—
Cursor RIGHT	&H43	&H63	—
Cursor LEFT	&H44	&H64	—
Touch screen (NOTE 1)	&HCE	—	&HCF
Message box button 1	&H01	—	—
Message box button 2	&H02	—	—
Message box button 3	&H03	—	—
Message box button 4	&H04	—	—

(NOTE 1)  
Key cord of touch screen

When pressing the touch screen, data of 6 bytes that show the pressed position will follow the key code of &HCE. Therefore, when not using &HCE key code, the following 6 bytes should be ignored.

POINT

When designating from "2" to "4" for Get key method, because the "&HCE" and the following 6 bytes data will be ignored automatically, it becomes easier to detect the key cord.  
When releasing the touch screen, position data will not attached. A key cord of &HCF will return.

[Key cords when the touch screen is pressed]  
[&HCE] [X position (digit of 100's place)] [X position (digit of 10's place)]  
[X position (digit of 1's place)] [Y Position (digit of 100's place)]  
[Y Position (digit of 10's place)] [Y position (digit of 1's place)]

4.2.8 Turning ON/OFF the user task screen

It is possible to turn ON/OFF (display / non-display) the user screen by using this command.  
The parameter for this command is 0 or 1.  
(A variable or condition equation that has value of 0 or 1 can be used also for the parameter)

GACTIVE ON/OFF

Parameter	Description
ON/OFF (1/0)	0: The user screen is turned OFF. 1: The user screen is turned ON.

POINT

- To switch the user screen display/non-display status, it is necessary to set "**Enabled**" for the item of "**Window active**" in the menu of <Service Utilities> -[12 User Task] [1 User Task].

- The [CLOSE / SELECT SCREEN] key has the same function.





# Chapter 5 Example of User Task Program

## 5.1 Display of Error Numbers of All Units

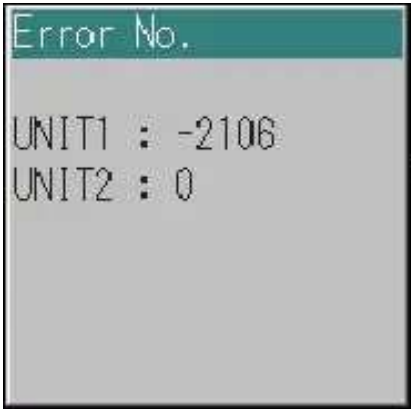
Numbers of errors that are currently occurring on each unit are displayed. The following case indicates that the UNIT 1 has Error Number 2106 (No program is recorded or selected), and that the UNIT 2 has no errors.

```
'UNIT Error
WINDOW 320,300,160,160
TITLE "ERROR NUMBER"

LOCATE 2,0
FOR L10%=1 TO SYSTEM%(1) STEP 1
L1%=SYSTEM%(140+L10%)
L1$=STR$(L1%);
L2$=SYSTEM$(100+L10%)+": "+L1$+" "
PRINT #0,L2$
NEXT
END
```

← Display on the Window.

← Obtain error numbers for the number of units to display them.



Error display

For details of general function such as STR\$, system functions such as SYSTEM\$, and flow control statements such as FOR, refer to the instruction manual “Robot Language”.

## 5.2 Initialization of Variables

This is a program used to clear all variables, which enables the unit to immediately return to its initial state by assigning F-key or else to variables. This program clears all variables to “0” (character string variables to character string having no characters). In addition, the program can be easily changed so as to only initialize necessary variables with any value.

'ALL CLR FOR L1%=1 TO 200 STEP 1 V[L1%]=0 NEXT FOR L1%=1 TO 50 STEP 1 V\$[L1%]="" NEXT EXIT	<div>← For the initialization of integer and real variables</div> <div>← For the initialization of character string variables</div>
--	---

# 5.3 Conversion from signal to variable

To convert from a one-byte signal to an integer, use the GETSIGB function to convert into an integer value. If any variable has multiple bytes or irregular data structure, process the data by user task to enable flexible response to external signals.

As an example, assume a vision device that transmits offset value taking a two-byte fixed-point value (decimal with decimal figures in lower-order two digits and sign in the most significant bit). Furthermore, assume that signals are input in the offset volume of X in the range of I2001 to I2016, the offset volume of Y in the range of I2017 to I2032, and the offset volume of Z in the range of I2033 to 2048.

**Robot program**

Obtain shift values to conduct shift operation.

```
'SHIFT
. . .
CALLMCR( 100, 10000 )
. . .
SHIFTA(3,V1! , V2! , V3! )
MOVEX
MOVEX
SHIFTA(3,0,0,0)
. . .
END
```

← Call a program that assigns the offset value to V1!, V2!, and V3!.

**User task program (USERTASK-A.100)**

Obtain shift values from signals.

```
'OFFSET
X
L1% = GETSIGB( 251 , 0 )
L2% = GETSIGB( 252 , 0 ) AND &H7F
L3% = L1% + L2% * 256
L1! = L3%-2*L3%*INP(2016)
V1! = L1! / 100
'Y
L1% = GETSIGB( 253 , 0 )
L2% = GETSIGB( 254 , 0 ) AND &H7F
L3% = L1% + L2% * 256
L1! = L3%-2*L3%*INP(2032)
V2! = L1! / 100
'Z
L1% = GETSIGB( 255 , 0 )
L2% = GETSIGB( 256 , 0 ) AND &H7F
L3% = L1% + L2% * 256
L1! = L3%-2*L3%*INP(2048)
V3! = L1! / 100
END
```

← Obtain the 8-bit status of signal as integer values.

← When the most significant signal turns ON, highlight the sign.

## 5.4 Control of Motion Range

This is a program that uses the SYSTEM function to obtain the angle of the axis J1 and, if the angle is located outside the specified motion range, not only displays the warning on the screen but also transmit the signal to external device. (The accuracy varies with the robot operating speed or user task cycle.)  
Changing the SYSTEM function argument enables control using the robot's end-point positions (x-, y-, and z-coordinate). For operation of the SYSTEM function, refer to information in the "AX CONTROLLER OPERATING MANUAL - ROBOT LANGUAGE".

### User task program (USERTASK-A.000)

<pre>'RANGE CONTROL INCLUDE "sample_54" IF FLAG=1 THEN *CHECK FLAG=1 LOWER LIMIT OF MOTION RANGE = J1 REFERENCE ANGLE - 20 UPPER LIMIT OF MOTION RANGE = J1 REFERENCE ANGLE + 20 *CHECK J1 AXIS ANGLE=SYSTEM!(SYS_M1J1) IF LOWER LIMIT OF MOTION RANGE&gt;RADDEG(J1 AXIS ANGLE) THEN *STOP IF UPPER LIMIT OF MOTION RANGE&lt;RADDEG(J1 AXIS ANGLE) THEN *STOP WINDOW 0,0,0,0 END *STOP SET STOP REQUEST WINDOW 100,100,132,132 COLOR 15 BGCOLOR 9 TITLE "WARNING" LOCATE 1,4 PRINT #0,"ROBOT" LOCATE 3,4 PRINT #0,"STOP!!" PAUSE 100 END</pre>	<p>← Perform calculations of upper and lower limits of the motion range only for the first motion.</p> <p>← Obtain the angle of the axis J1 of Mechanism 1 in units of radian.</p> <p>← Describe processing to be performed if the robot exceeds this motion range after this character string.</p>
--	---

### Variable definition file (sample\_54.inc)

```
J1 REFERENCE ANGLE,0
SYS_M1J1,100,MECHANISM1 J1JOINT ANGLE (RAD)
J1 AXIS ANGLE,V1!
FLAG,V100%,1 FOR SECOND WEEK AND LATER

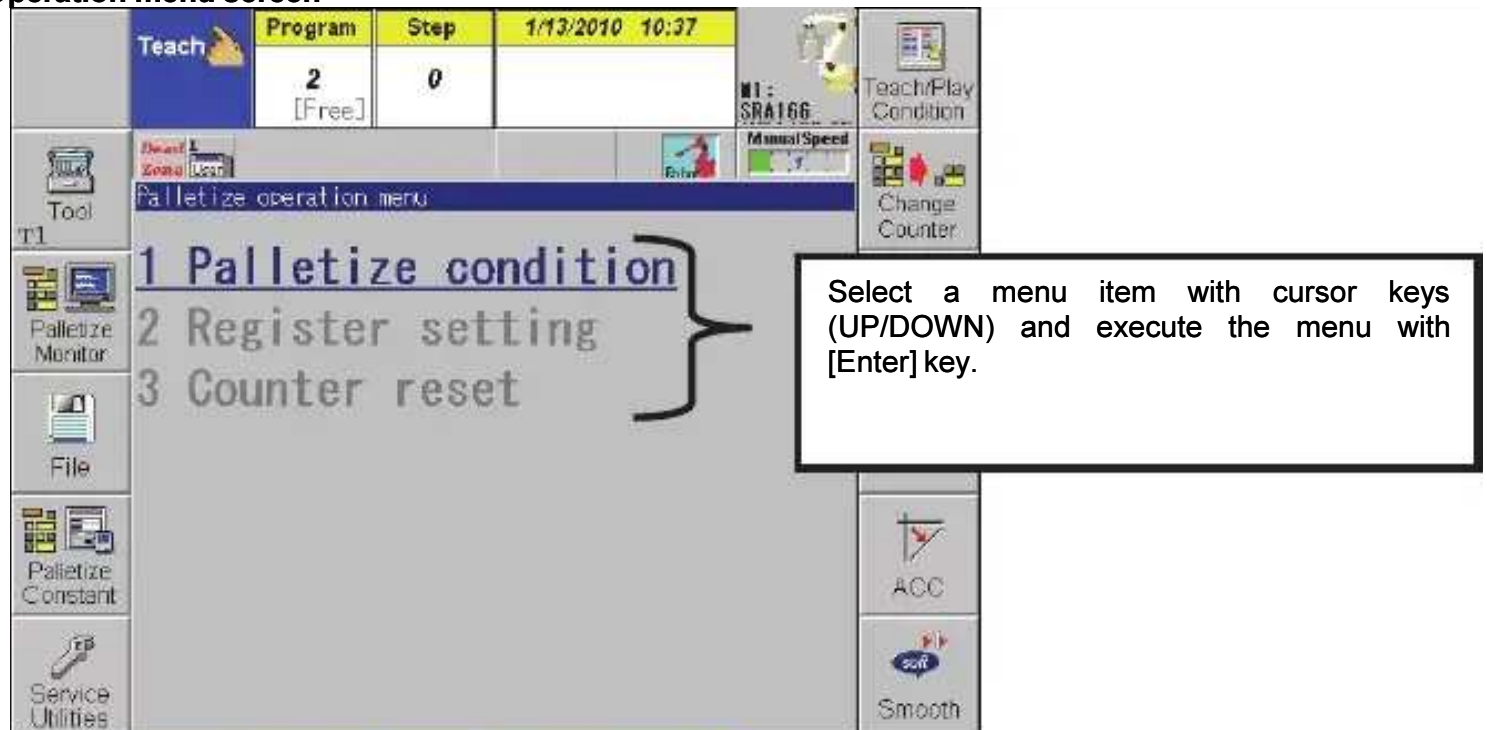
LOWER LIMIT OF MOTION RANGE,V10!
UPPER LIMIT OF MOTION RANGE,V11!
STOP REQUEST,O10
```

## 5.5 Displaying a user monitor screen

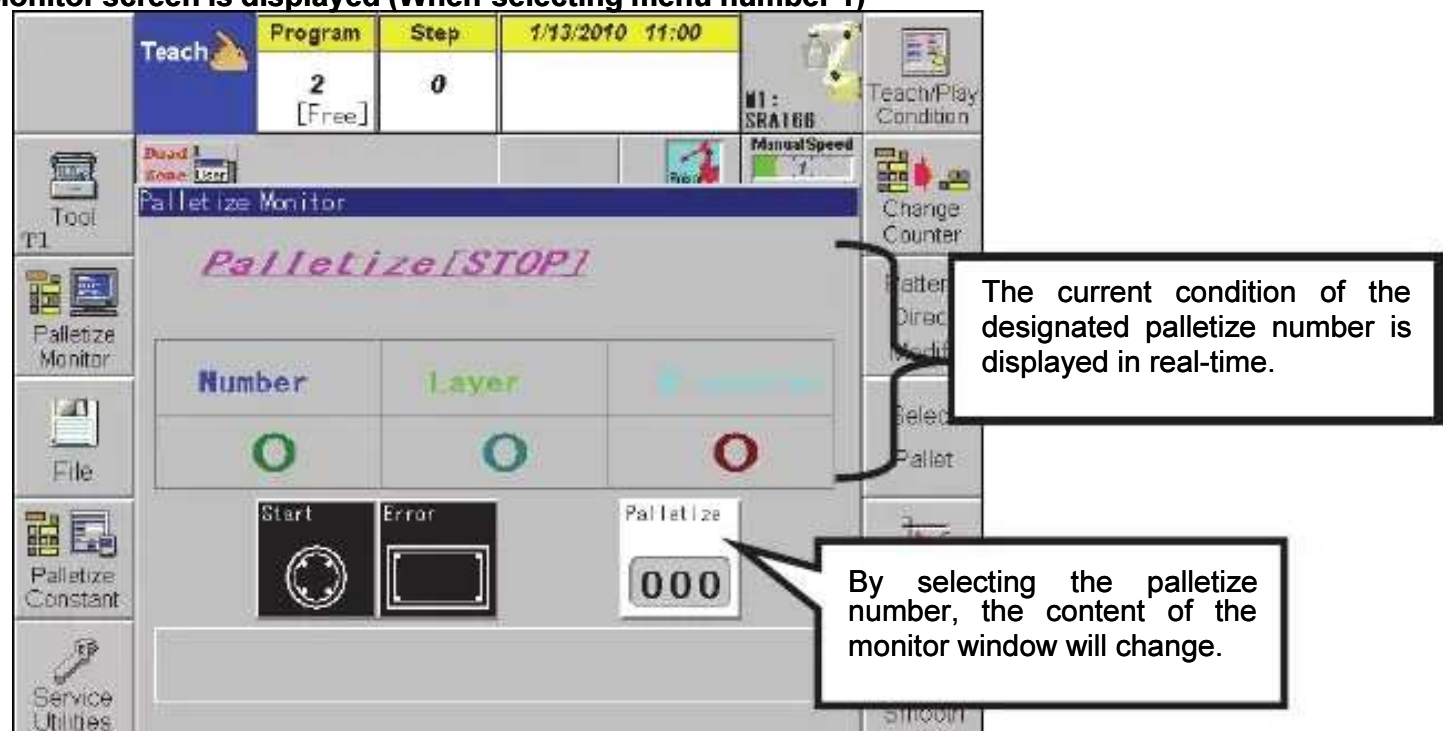
In this example, a menu screen for palletizing application is made.

Like this example, it is possible to execute a process or display a screen suitable for the respective menu items. By customizing menu items or screens using user task function, it becomes easier to operate the controller.

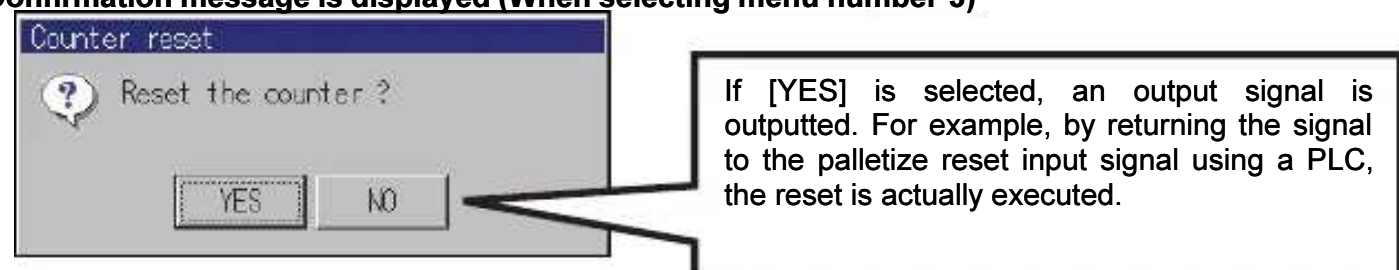
### Operation menu screen



### Monitor screen is displayed (When selecting menu number 1)



### Confirmation message is displayed (When selecting menu number 3)



Menu display program (USERTASK-A.000)

```
' Palletize operation
*INIT
L1$  = "1 Palletize condition"
L2$  = "2 Register setting"
L3$  = "3 Counter reset"

' ---- Create a Window ----
WINDOW 80,120,480,380
CLS

TITLE "Palletize operation menu"

' ---- Menu display ----
L100% = 0
*MENU
LOCATE 1,0
FOR L1%=0 TO 2 STEP 1
IF (L100% = L1%)
COLOR 2
GFONT 32,16,1,0,1
ELSE
COLOR 1
GFONT 32,16,1,0,0
ENDIF
PRINT #0,L$[L1%+1]
NEXT

' ---- Wait for key input ----

L10$ = GETKEY$(3)
L10% = ORD(L10$)

' ---- Cursor UP ----
IF (&H41 = L10%)
IF (0 < L100%)
L100% = L100% - 1
ENDIF

' ---- Cursor DOWN ----
ELSEIF (&H42 = L10%)
IF (2 > L100%)
L100% = L100% + 1
ENDIF

' ---- [Enter] key ----
ELSEIF (&H0D = L10%)
CALLP L100%+1

GOTO *INIT
ENDIF
GOTO *MENU
END
```

Creating a Window

Displaying menu items

Get the key input information and change the menu selection status. When [Enter] key is pressed, the respective program number for the selected menu is called.

**Monitor display program (USERTASK-A.001)**

```

' ---- Palletize user screen ----
' ---- Initialize the local variables ----
L1$ = "Palletize Monitor"
L2$ = "  Number"
L3$ = "  Layer"
L4$ = "W count"
L11$ = "Palletize "
L12$ = "De-palletize "
L13$ = "[RUN]"
L14$ = "[STOP]"

' ---- Initialize the window ----
CLS
TITLE L1$

' ---- Draw the item frames ----
FOR L150%=0 TO 1 STEP 1
FOR L151%=0 TO 2 STEP 1
GBOX 10+149*L151%, 80+49*L150%, 150, 50, 0
NEXT
NEXT

' ---- Display the items ----
GFONT 18,11,1,0,0
FOR L150%=0 TO 2 STEP 1
COLOR L150%*2+3
LOCATE 5,5+19*L150%
PRINT #0, L$[L150%+2]
NEXT

' ---- Display the software keys ----
FOR L150% = 1 TO 6 STEP 1
GSOFTKEY 80*L150%, 240, L150%
NEXT

' ---- Create the guide message area ----
COLOR 15
GBOX 10, 270, 450, 50, 0
'

' ---- Refresh the display ----
*REALTIME_DISP

' ---- Get the palletize information ----
PALLET3_GETREG V1%,V100%,1,16

' ---- Display the current status ----
IF (0=V4%)
L20$ = L11$
ELSE
L20$ = L12$
ENDIF
L20$ = L20$+" [ "
IF (1=V3%)
L20$ = L20$+L13$
ELSE
L20$ = L20$+L14$
ENDIF
L20$ = L20$+" ]"
LOCATE 1,5
COLOR 11
GFONT 20,16,1,1,1
PRINT #0, L20$

' ---- Information display ----
FOR L150%=0 TO 2 STEP 1

```

Initialize the variable for display

Erase the screen for menu display and set the new title.

Make frames for item display

Display the items

Display the software keys

Create a guide message area

Get the palletize information

Display the palletize operation status.

5.5 Displaying a user monitor screen

```
COLOR L150%*2+4
LOCATE 7,5+19*L150%
IF (0=L150%)
L20$ = STR$(V2%)
ELSEIF (1=L150%)
L20$ = STR$(V5%)
ELSE
L20$ = STR$(V6%)
ENDIF
L106%=LEN(L20$)
IF (2 > L106%)

L21$= " " + L20$
ELSE
L21$= L20$
ENDIF
GFONT 32,32,1,0,0
PRINT #0, L21$
NEXT
```

} Display the palletize status.

```
' ---- Get key input information ([Reset R] key)----
L25$ = GETKEY$(0)
L25% = ORD(L25$)
IF (&H52 <> L25%)
GOTO *REALTIME_DISP
ENDIF
END
```

} If [Reset R] key is pressed, exit the program and return to the menu screen.

Message display program (USERTASK-A.003)


```
' Message box (Confirmation display)
L1$ = "Counter reset"
L2$ = "Reset the counter?"
GMSGBOX 2, 2, L1$,L2$, 2,"", "", "", ""
L10$ = GETKEY$(4)
L10% = ORD(L10$)
' Output signal
IF (1 = L10%)
SET O21
PAUSE 100
RESET O21
ENDIF
```

} Display a message box

} Get the key input information and if [YES] is selected, output a signal as a pulse signal.





		<a href="http://www.nachi-fujikoshi.co.jp/">http://www.nachi-fujikoshi.co.jp/</a>	
Japan Main Office	Phone: +81-3-5568-5245	Fax: +81-3-5568-5236	Shiodome Sumitomo Bldg. 17F, 1-9-2 Higashi-Shinbashi Minato-ku, TOKYO, 105-0021 JAPAN
<b>Nachi Robotic Systems Inc. (NRS)</b>		<a href="http://www.nachirobotics.com/">http://www.nachirobotics.com/</a>	
North America Headquarters	Phone: 248-305-6545	Fax: 248-305-6542	42775 W. 9 Mile Rd. Novi, Michigan 48375, U.S.A
Indiana Service Center	Phone: 248-305-6545	Fax: 248-305-6542	Greenwood, Indiana
Ohio Service Center	Phone: 248-305-6545	Fax: 248-305-6542	Cincinnati, Ohio
South Carolina Service Center	Phone: 248-305-6545	Fax: 248-305-6542	Greenville, South Carolina
Canada Branch Office	Phone: 905-760-9542	Fax: 905-760-9477	89 Courtland Ave., Unit No.2, Concord, Ontario, L4K 3T4, CANADA
Mexico Branch Office	Phone :+52-555312-6556	Fax: +52-55-5312-7248	Urbina No.54, Parque Industrial Naucalpan, Naucalpan de Juarez, Estado de Mexico C.P. 53489, MEXICO
<b>NACHI EUROPE GmbH</b>		<a href="http://www.nachi.de/">http://www.nachi.de/</a>	
Central Office Germany	Phone: +49-2151-65046-0	Fax: +49-2151-65046-90	Bischofstrasse 99, 47809, Krefeld, GERMANY
U.K. branch	Phone: +44-0121-423-5000	Fax: +44-0121-421-7520	Unit 3, 92, Kettles Wood Drive, Woodgate Business Park, Birmingham B32 3DB, U.K.
Czech branch	Phone: + 420-255-734-000	Fax: +420-255-734-001	Obchodni 132, 251 01 Cestlice, PRAGUE-EAST CZECH REPUBLIC
Turkey branch	Phone: + 90-(0)216-688-4457	Fax: +90-(0)216-688-4458	Ataturk Mah. Mustafa Kemal Cad. No:10/1A 34758 Atasehir / Istanbul - TURKEY
<b>NACHI AUSTRALIA PTY. LTD.</b>		<a href="http://www.nachi.com.au/">http://www.nachi.com.au/</a>	
Robotic Division & Victoria office	Phone: +61-(0)3-9796-4144	Fax: +61-(0)3-9796-3899	38, Melverton Drive, Hallam, Victoria 3803, , AUSTRALIA
Sydney office	Phone: +61-(0)2-9898-1511	Fax: +61-(0)2-9898-1678	Unit 1, 23-29 South Street, Rydalmere, N.S.W, 2116, AUSTRALIA
Brisbane office	Phone: +61-(0)7-3272-4714	Fax: +61-(0)7-3272-5324	7/96 Gardens Dr, Willawong, QLD 4110, , AUSTRALIA
<b>NACHI SHANGHAI CO., LTD.</b>		<a href="http://www.nachi.com.cn/">http://www.nachi.com.cn/</a>	
Shanghai office	Phone: +86-(0)21-6915-2200	Fax: +86-(0)21-6915-2200	11F Royal Wealth Centre, No.7 Lane 98 Danba Road Putuo District, Shanghai 200062, China
<b>NACHI KOREA</b>		<a href="http://www.nachi-korea.co.kr/">http://www.nachi-korea.co.kr/</a>	
Seoul office	Phone: +82-(0)2-469-2254	Fax: +82-(0)2-469-2264	2F Dongsan Bldg. 276-4, Sungsu 2GA-3DONG, Sungdong-ku, Seoul 133-123, KOREA

Copyright NACHI-FUJIKOSHI CORP.

**Robot Division**

1-1-1, FUJIKOSHIHONMACHI, TOYAMA CITY, JAPAN 930-8511  
Phone +81-76-423-5137  
Fax +81-76-493-5252

NACHI-FUJIKOSHI CORP. holds all rights of this document. No part of this manual may be photocopied or reproduced in any form without prior written consent from NACHI-FUJIKOSHI CORP. Contents of this document may be modified without notice. Any missing page or erratic pagination in this document will be replaced.

In case that an end user uses this product for military purpose or production of weapon, this product may be liable for the subject of export restriction stipulated in the Foreign Exchange and Foreign Trade Control Law. Please go through careful investigation and necessary formalities for export.

Original manual is written in Japanese.