SOFTWARE

KR C2 / KR C3

User Programming

KUKA System Software (KSS)
Release 5.2

Issued: 12 Apr 2006 Version: 01

KUKA Roboter GmbH
Zugspitzstrasse 140
D-86165 Augsburg

Other functions not described in this documentation may be operable in the controller. The user has no claim to these functions, however, in the case of a replacement or service work.
We have checked the content of this documentation for conformity with the hardware and software described. Nevertheless, discrepancies cannot be precluded, for which reason we are not able to guarantee total conformity. The information in this documentation is checked on a regular basis, however, and necessary corrections will be incorporated in subsequent editions.
Subject to technical alterations without an effect on the function.

Contents

1 Program editing

Before creating a robot program for the first time, you should familiarize yourself with the KUKA file manager “Navigator”.

If you wish to create a new program, you must first create a so-called “skeleton program”. If, on the other hand, you wish to modify an existing program, you simply need to select it or load it in the editor.

1.1 Creating and opening a program

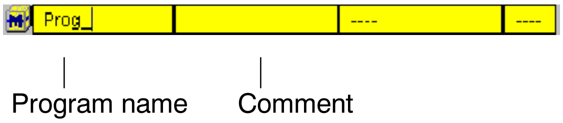
Block functions (e.g. Copy) are also available at expert level. More detailed information can be found in the **Programming Handbook** in the documentation **[Expert Programming]**.

1.1.1 Creating a new program

With the Navigator active, press the softkey “New”.

If a program has been selected or there is a program in the editor, you must first toggle to the file selection window. No new program can be created until this is done.

Enter the desired program name (max. 24 characters) and the corresponding comment in the input line.

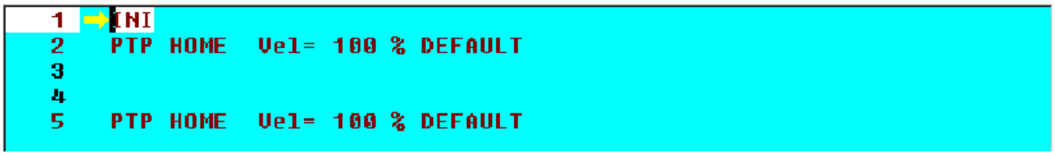


Where possible, create user programs in the directory “R1\Program” in order to ensure that they are automatically saved via the menu item “File > Archive > Applications”.

1.1.2 Selecting an existing program

The softkey “Select” is available for the purposes of program execution.

In this way the program can also to be tested at the same time as being created. The desired program is displayed in the program window.



```
INI
PTP HOME  Vel= 100 % DEFAULT


PTP HOME  Vel= 100 % DEFAULT
```

1.1.3 Duplicating an existing program

The softkey “Duplicate” creates a copy of a program.

1.1.4 Saving a program to floppy disk

The softkey “Archive” allows you to save important data to floppy disk. With Navigator, the user can see what is stored in the archives.

A request for confirmation is generated which must be answered before the saving process is carried out.

> The softkey “Archive” corresponds to the menu command “Archive > Current selection”.
>
> **[File > Archive > Current selection]**

1.1.5 Deleting a program

If you want to delete a program, it must not be currently selected or being edited. You might first have to deselect the program or close the editor. The program is permanently deleted after a request for confirmation.

1.1.6 Opening an existing program in the editor (Edit)

If you wish to make modifications to an existing program, you can either select the program or load it into the editor. While you are editing one program in the editor, another (selected) program can simultaneously be executed.

The command “Open” is provided in the softkey bar for loading files into the editor.

The desired program is displayed in the program window. The assignment of the menu key, softkey and status key bars changes at the same time, in order to make functions available which are necessary for programming the robot.

If a program has already been selected, the option “Open” is no longer available in the softkey bar. Use the corresponding menu command instead.

[File > Open > File/Folder]

The following options are available for processing two programs simultaneously:

Select program then edit another program

Select the desired program, toggle back to the Navigator and load the next program into the editor via the menu “File -> Open”.

A selected program cannot be edited at the same time.

Load program into the editor then select another program

Load the desired program into the editor, toggle to the Navigator and select the next program for editing using the softkey “Select”.

A program in the editor cannot be selected.

1.1.7 Printing a program

[File > Print > Current selection]

If you have connected a printer to your control computer or network, you can print out program listings or the logbook.

The print function is only available if a printer has been configured in the operating system. More detailed information on this can be found in the “Windows” online help file.

1.1.7.1 Print current selection

The files that have been selected using the arrow keys and the space bar are sent to the printer. See chapter “Operator control”.

1.1.7.2 Print logbook

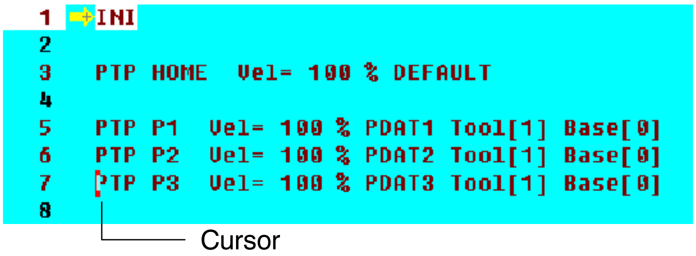
This function sends the logbook file, which can be viewed via the menu “Display > Diagnosis > Logbook > Display”, to the printer.

Detailed information about the logbook can be found in the **Control cabinet** User Manuel.

1.2 Working with a selected program

1.2.1 Deleting a selected program

Use the arrow keys to move the cursor to the line you want to delete.



```
INI

PTP HOME  Vel= 100 % DEFAULT

PTP P1  Vel= 100 % PDAT1 Tool[1] Base[0]
PTP P2  Vel= 100 % PDAT2 Tool[1] Base[0]
PTP P3  Vel= 100 % PDAT3 Tool[1] Base[0]

```

[Program > Delete]

Confirm the request for confirmation that now appears.

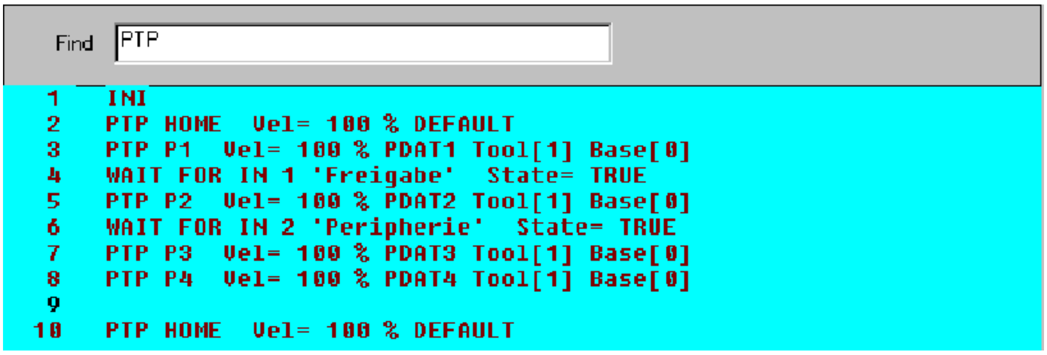
1.2.2 Searching the program for a character string

Here you can search through the program for a character string of your choice.

[Program > Find]

Alternatively, you can use the keyboard shortcut “**CTRL**”+“**F**”.

This function searches through the program for a character string entered by the user. Once the command has been selected, the search form is displayed in the program window:



Find PTP

```
INI
PTP HOME  Vel= 100 % DEFAULT
PTP P1  Vel= 100 % PDAT1 Tool[1] Base[0]
WAIT FOR IN 1 'Freigabe'  State= TRUE
PTP P2  Vel= 100 % PDAT2 Tool[1] Base[0]
WAIT FOR IN 2 'Peripherie'  State= TRUE
PTP P3  Vel= 100 % PDAT3 Tool[1] Base[0]
PTP P4  Vel= 100 % PDAT4 Tool[1] Base[0]

PTP HOME  Vel= 100 % DEFAULT
```

The search is started by pressing the softkey “Find” or the Enter key and commences at the position where the edit cursor is located. If the search program finds the character string entered, the corresponding line is selected.

The character string entered remains displayed in the search form as a suggestion. You can now search further through the program by pressing the softkey “Find” or the Enter key, or search for a new string.

If the term being searched for is not found, a corresponding text is displayed in the message window:

How the search within the program is conducted depends on the display mode in the program window. The expert must switch the option “Detail view” on if FOLDS are also to be included in the search.

Alternatively, you can terminate the search function by pressing the softkey “Cancel” or the Escape key. The edit cursor is then located in the line in which the last character string was found.

If the search function is called again, the last string to be entered is shown in the form as a suggestion.

1.2.3 Cancel program

This command saves the changes made to the program and closes the program window.

[Program > Cancel program]

The softkey “Deselect” is only available in the softkey bar if no program window is open.

1.2.4 Reset program

[Program > Reset program]

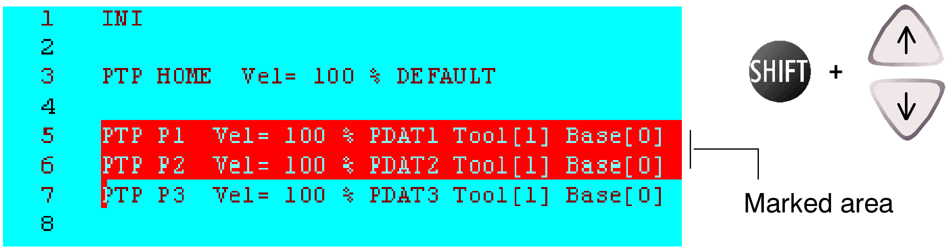
A program that has been stopped or interrupted can be returned to the initial state it had when it was selected.

The yellow block pointer then jumps to the first line of the indicated program. The selected program can subsequently be restarted.

1.3 Working with the program editor

1.3.1 Deleting a program line in the editor

Use the arrow keys to move the cursor to the line you want to delete or select an area.



```
INI

PTP HOME  Vel= 100 % DEFAULT

PTP P1  Vel= 100 % PDAT1 Tool[1] Base[0]
PTP P2  Vel= 100 % PDAT2 Tool[1] Base[0]
PTP P3  Vel= 100 % PDAT3 Tool[1] Base[0]

```

[Program > Delete]

Confirm the request for confirmation that now appears.

1.3.2 Searching for a character string in the editor

See section 1.2.2.

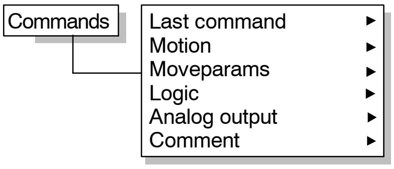
1.3.3 Close editor

Following a request for confirmation, this command saves to the hard disk the changes made in the editor. The editor is then closed.

[Program > Close]

2 Program commands

This chapter is intended to provide an overview of the fundamentals of the program commands available, as well as their functions and the procedure for programming.



Overview	
Last command	Instruction for entering the last command executed
Motion	Enables the programming of PTP, LIN and CIRC motions
Moveparams	Programming of the torque monitoring
Logic	Programming of logic commands and wait times, path-dependent switching and pulse functions, setting or polling inputs and outputs
Analog output	Setting of analog outputs under program control
Comment	Used to insert comments in programs

The “Commands” menu is only available if a program has been selected or loaded in the editor.

Changes in a program are saved to the hard disk in the following situations:

- An inline form is closed by pressing the softkey “OK” or the Enter key;
- Point coordinates are saved by pressing the softkey “Touch Up”;
- A line is inserted, deleted or modified in the program or editor;
- A line is exited in the editor by means of the arrow keys “↑” or “↓”;
- A program is canceled or the editor is closed.

2.1 Last command

This instruction allows you to enter the last command executed.

[Commands > Last command]

This command is also offered in the softkey bar.

2.2 Motion

To move a robot tool to a point under program control, a corresponding motion instruction must be programmed. This instruction contains the type of motion and the velocity, the definition of the end point – for circular paths also a midpoint – and other settings depending on the type of motion.

Please note:

If one or more of the robot axes hits its end stop without being braked and at a velocity in excess of the manual velocity set by the manufacturer, the buffer concerned must be replaced immediately.

If this happens to axis 1 of a wall-mounted robot, its rotating column must be replaced.

Types of motion

The following types of motion are available for programming motions:

Standard motions	
PTP (Point-to-point)	The tool is moved as quickly as possible along a curved path in space to an end point
LIN (Linear)	The tool is guided at a defined velocity along a straight line
CIRC (Circular)	Motion of the tool at a defined velocity along a circular path

In sequences of motion instructions, there are two alternatives for selecting how movements between individual points are executed:

Motion between the individual points	
Exact positioning	The motion stops exactly at the programmed point.
Approximate positioning (Cont)	A smooth transition can be made from one motion to the next without exact positioning at the end point.

Programming

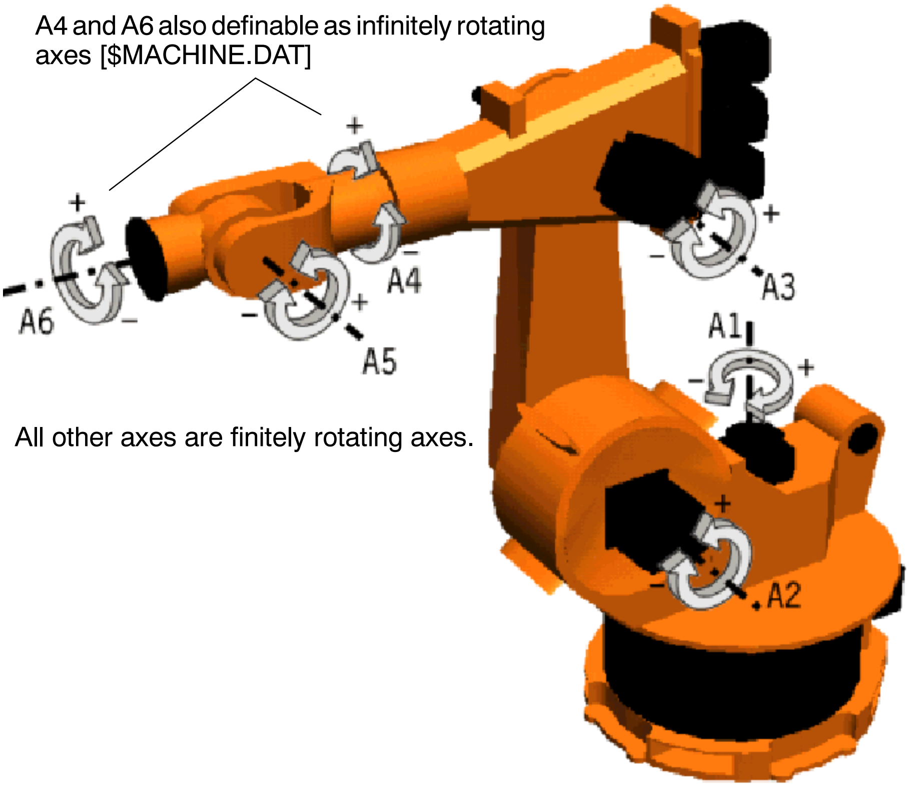
Pay attention to the position of the edit cursor. The next program line created by you will be inserted as a new line after the cursor.

[Commands > Motion > PTP]

You can now make your selection from the motion instructions (PTP, LIN or CIRC) offered.

Infinitely rotating axes

All the robot axes (A1 ... A6) are defined by the manufacturer as finitely rotating axes (i.e. with software limit switches). For certain applications, however, axes A4 and A6 can be configured as infinitely rotating axes.

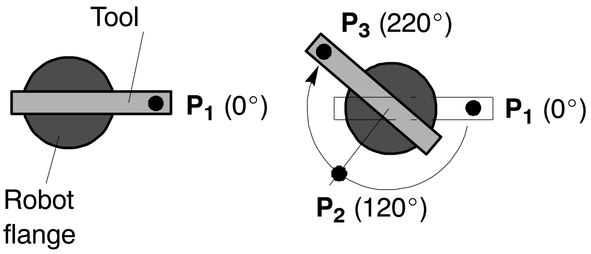

A4 and A6 also definable as infinitely rotating axes [$MACHINE.DAT]
+
−
A6
+
−
A4
+
−
A5
+
−
A3
A1
−
+
+
−
A2
All other axes are finitely rotating axes.

If the machine data have been altered to define the axes A4 and/or A6 as infinitely rotating axes, please bear in mind that each rotation is executed along the shortest path.

This can cause problems if a tool with supply lines (for example, a welding gun) is installed on the robot.

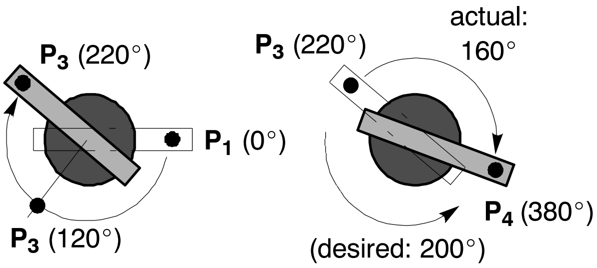
In the following example, two motion instructions (P1 – P2 and P2 – P3) are programmed and the coordinates saved.

The first motion instruction in this example causes axis A6 to turn 120° from P1 (0°) to P2 (120°). The second motion command rotates A6 a further 100° from P2 (120°) to P3 (220°).



The third motion instruction is intended to move the tool 200° in the opposite direction, by rotating axis A6, from P3 to P4, i.e. to the position 20° from the starting position P1.

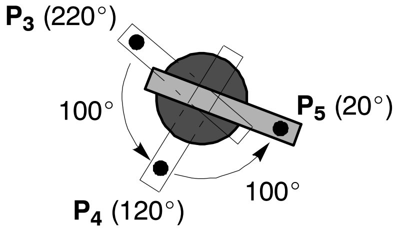
When the program is executed, however, axis A6 will turn along the shorter path from P3 (220°) to P4 (380°), covering just 160°.



This will inevitably result in any supply lines from the robot to the tool being “wound” round the wrist.

It is therefore necessary for the second (return) motion to be split up into two motion instructions.

In this example, two motions (P3-P4 and P4-P5) have been programmed, each with an angle of rotation of 100°. This ensures that the tool moves to the end point in the correct direction when the program is executed.



2.2.1 Point-to-point motions (PTP)

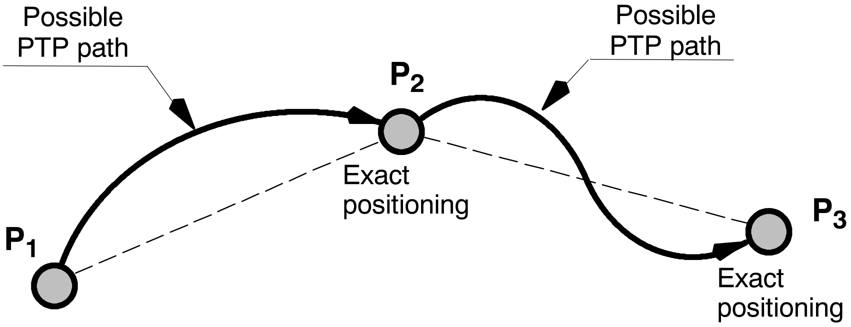
Here, the robot system is positioned using the quickest route between two points. Since the motion starts and ends in all of the axes at the same time, the axes have to be synchronized. The path taken by the robot cannot, therefore, be predicted exactly.

When this instruction is used, the motion follows a path defined by the robot. In order to incorporate dynamic effects and avoid possible collisions, this path must first be executed at reduced program override (POV) near obstacles.

Point names may not begin with "POINT" as this is a keyword used by KRL (KUKA Robot Language).

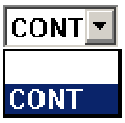
PTP motions with exact positioning

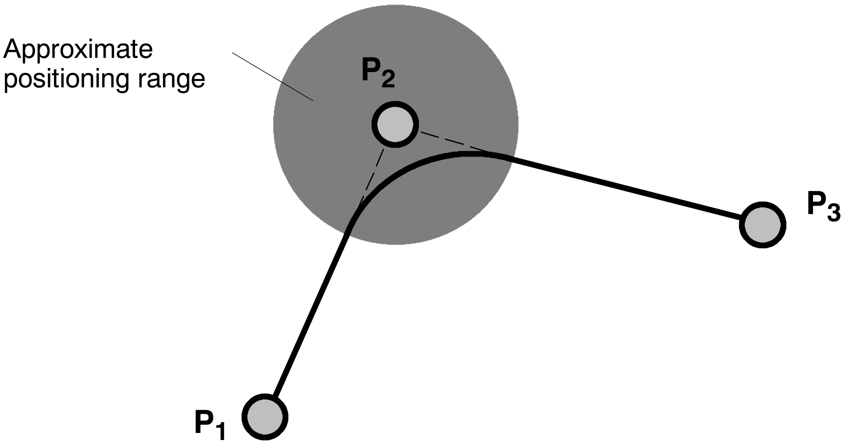
In the case of PTP motions with exact positioning, the robot stops exactly at each end point.



PTP motion with approximate positioning

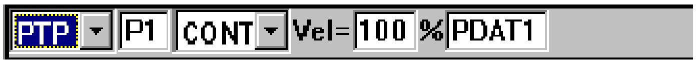
During approximate positioning, the controller monitors a so-called approximate positioning range around the end point. This, in the example below, is the point P$_2$. When the TCP enters this area, the robot motion immediately proceeds towards the end point of the next motion command.





Programming a PTP motion

When the option “PTP” is selected from the menu “Motion”, the inline form for entering the values required for executing this instruction is opened in the program window.



Box name	Function	Range of values
PTP	Type of motion	PTP, LIN, CIRC
P1	Point name	Max. 23 characters
Tool	Tool number	Nullframe, Tool_Data[1]...[16]
Base	Workpiece number	Nullframe, Base_Data[1]...[16], EX_AX_DATA[1]...[6]
External TCP	Robot guides tool/workpiece	True, False
CONT	Approximate positioning ON	“ ”, Cont
Vel=100%	Velocity	1 to 100% of the maximum value (default: 100%)
PDAT1	Motion parameters	
Acceleration	Acceleration	0 ... 100%
Approximation distance *1	Start or end of the approximate positioning range	0 ... 100%
*1 Only available if “CONT” has been switched on		

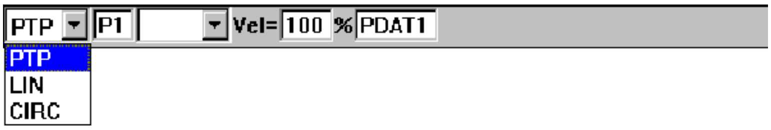
Remember that a high payload/supplementary load, high acceleration and short paths increase the loading of the drive motors unnecessarily.

If the program window is in focus, the various input windows can be selected using the “↓” and “↑” arrow keys. The window that is currently selected is highlighted by a color background.

The program window can be activated by repeatedly pressing the “Window selection” key until the entire window is highlighted in color.

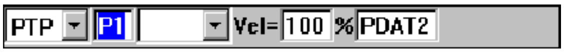
It is possible to abort programming of the PTP motion at any time by pressing the softkey “Cmd. Abort” or the “ESC” key. The command will **not** be inserted in this case.

The type of motion is set using the softkey “LIN/CIRC”.



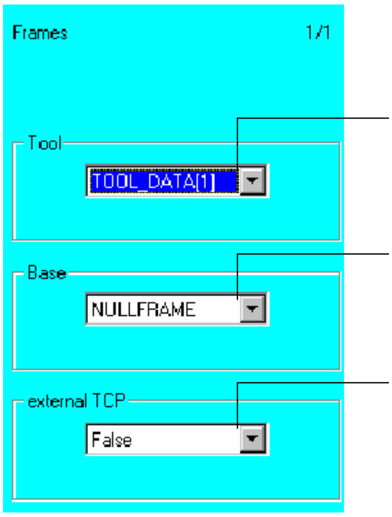
Alternatively, the cursor can be positioned in the first input box. The assignment of the status key changes, thus allowing you to toggle between the different types of motion.

Move the cursor to the next input box, here “P1”.



PTP | P1 | Vel= 100 % | PDAT2

A parameter list is opened for entering data relating to the workpiece and tool. Activate this parameter list using the “Window selection” key.



Frames 1/1

Tool: TOOL_DATA[1]

Base: NULLFRAME

external TCP: False

Select here from the available tools.

Select here one of the workpiece coordinate systems (BASE).

Tell the controller here whether the robot is guiding the tool or the workpiece:

Robot guiding tool: external TCP = False

Robot guiding workpiece: external TCP = True

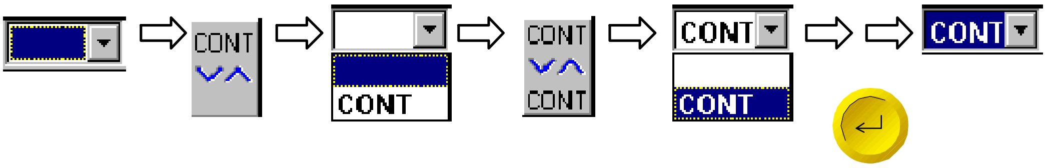
Move the cursor to the next input box in the inline form. The assignment of the status key bar changes. The approximate positioning function can be switched on or off using this status key.

Inline form “Exact positioning”.

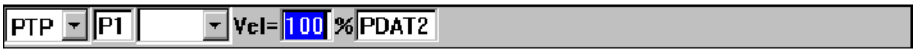
Press the status key on the bottom right-hand side. The menu in the inline form is opened.

Using the status key, choose between “Exact positioning” and “Approximate positioning” (CONT).

Press the Enter key and the menu is closed.



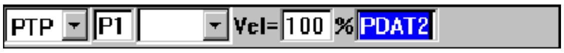
Move the cursor to the box “Vel=”.



PTP | P1 | Vel= 100 % | PDAT2

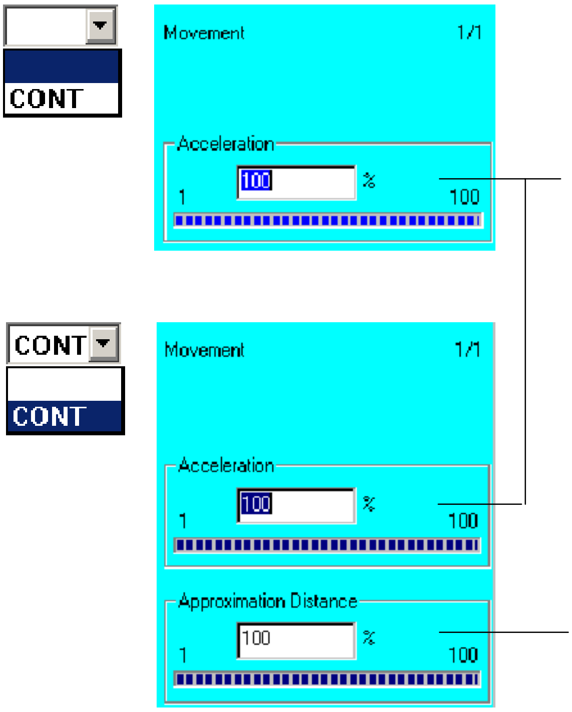
Here you can specify what percentage of the maximum possible velocity the robot should use later when executing the motion. You can either enter the value using the keyboard or alter it using the status key.

Move the cursor to the next input box, here “PDAT2”.



PTP | P1 | Vel= 100 % | PDAT2

A parameter list is opened, in which data must be entered characterizing the motion in greater detail. Activate this status window using the “Window selection” key. You can either enter the values in the input boxes using the keyboard or alter them using the status key.



Here you can reduce the acceleration to be used in the motion.

Here you can change the approximate positioning range to be used in the motion.

You can use the softkey “Logic” at any time to insert a so-called logic command into your program, irrespective of the input box that is currently in focus. This function is not available if an existing motion command is being modified.

Detailed information on logic commands can be found in the section **[Logic]**.

You can use the softkey “Comment” at any time to insert a comment line into your program, irrespective of the input box that is currently in focus. This function is not available if an existing motion command is being modified.

Detailed information on comments in robot programs can be found in the section **[Comment]**.

If the softkey “Suggest” is pressed, the program finds the lowest standard point name available in the local data list and enters this name in the open inline form. i.e. if points P1 and P3 are occupied, P2 will be suggested. This function is not available if an existing motion command is being modified.

This softkey is used to save the coordinates of the current robot position as the end point following a request for confirmation.

The softkey “Touch Up” allows you, at any moment, to save the current robot coordinates for the program line in which the cursor is positioned. You thus have the possibility, for example, to program a sequence of motion instructions and then define the exact end point coordinates later.

The coordinates of the programmed point are saved in the data list.

Press the softkey “Cmd OK” or the Enter key. The motion is saved and the program line is inserted into the program.

If the position of the end point has not yet been taught (Touch Up), the current position of the robot is automatically saved.

2.2.2 Linear motions (LIN)

With a linear motion, the robot axes are coordinated in such a way that the TCP or workpiece reference point is moved along a straight line to the end point.

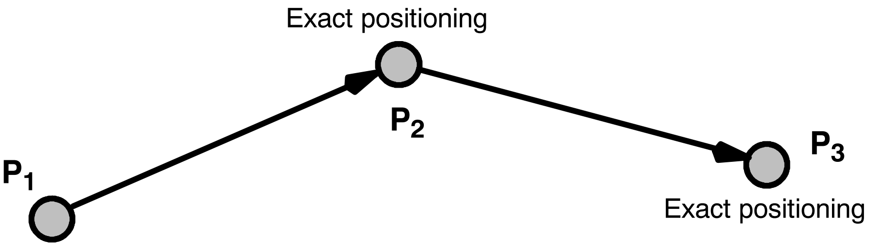
Linear motions are used whenever the robot has to follow an exact path to a point at a defined velocity.

Only the reference point follows the programmed path. The actual tool or workpiece is able to change its orientation during the motion in accordance with the settings that have been made.

Point names may not begin with "POINT" as this is a keyword used by KRL (KUKA Robot Language).

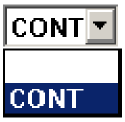
LIN motions with exact positioning

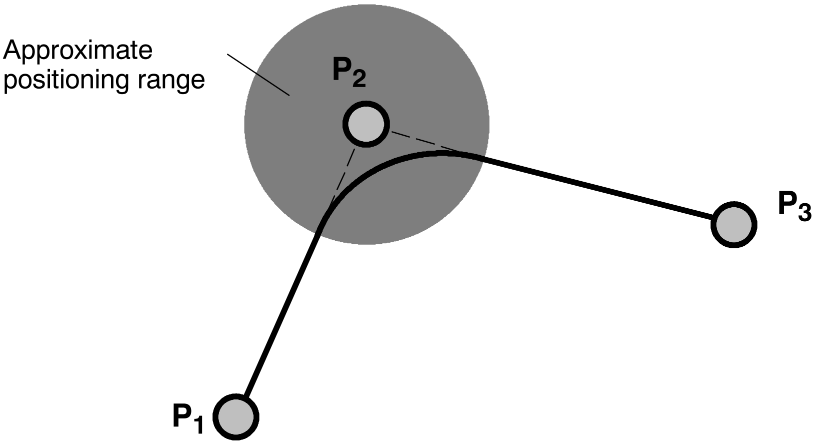
In the case of LIN motions with exact positioning, the robot stops exactly at each end point.



LIN motion with approximate positioning

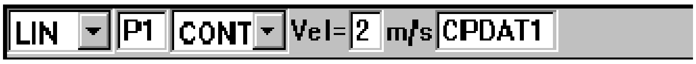
During approximate positioning, the controller monitors a so-called approximate positioning range around the end point. This, in the example below, is the point P_2. When the TCP enters this area, the robot motion immediately proceeds towards the end point of the next motion command.





Programming a LIN motion

When the option “LIN” is selected from the menu “Motion”, the inline form for entering the values required for executing this instruction is opened in the program window.



LIN P1 CONT Vel= 2 m/s CPDAT1

Box name	Function	Range of values
LIN	Type of motion	PTP, LIN, CIRC
P1	Point name	Max. 23 characters
Tool	Tool number	Nullframe, Tool_Data[1]...[16]
Base	Workpiece number	Nullframe, Base_Data[1]...[16]
External TCP	Robot guides tool/workpiece	True, False
CONT	Approximate positioning ON	“ ”, Cont
Vel=2m/s	Velocity	0.001 ... 2 m/s (default: 2 m/s)
CPDAT1	Motion parameters	
Acceleration	Acceleration	0 ... 100%
Approximation distance *1	Start or end of the approximate positioning range	0 ... 300 mm
Orientation control	The type of orientation control	Standard, WristPTP, ConstantOrientation
*1 Only available if “CONT” has been switched on		

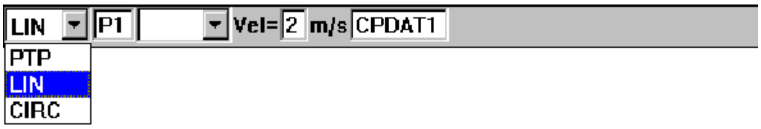
Remember that a high payload/supplementary load, high acceleration and short paths increase the loading of the drive motors unnecessarily.

Depending on the length of the path, the degree of acceleration and the approximation distance, it is possible that the programmed velocity will not be reached.

If the program window is in focus, the various input windows can be selected using the “↓” and “↑” arrow keys. The window that is currently selected is highlighted by a color background. The program window can be activated by repeatedly pressing the “Window selection” key until the entire window is highlighted in color.

It is possible to abort programming of the LIN motion at any time by pressing the softkey “Cmd. Abort” or the “ESC” key. The command will **not** be saved in this case.

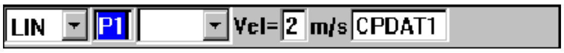
The type of motion is toggled using the softkey “CIRC/PTP”.



LIN P1 Vel= 2 m/s CPDAT1

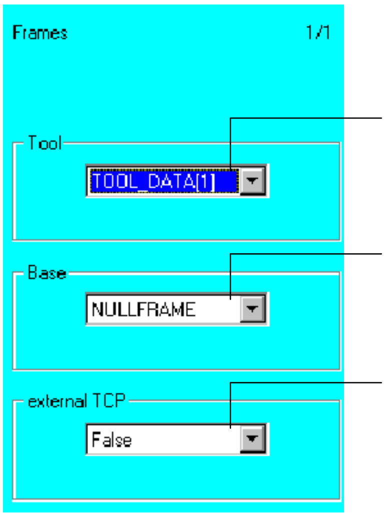
Alternatively, the cursor can be positioned in the first input box. The assignment of the status key changes, thus allowing you to toggle between the different types of motion.

Move the cursor to the next input box, here “P1”.



LIN P1 Vel= 2 m/s CPDAT1

A status window is opened for entering data relating to the workpiece and tool. Activate this status window using the “Window selection” key.



Frames 1/1

Tool: TOOL_DATA[1]

Base: NULLFRAME

external TCP: False

Select here from the 16 available tools.

Select here one of the 16 saved workpiece coordinate systems (BASE).

Tell the controller here whether the robot is guiding the tool or the workpiece:

Robot guiding tool: external TCP = False

Robot guiding workpiece: external TCP = True

Now activate the program window again using the “Window selection” key.

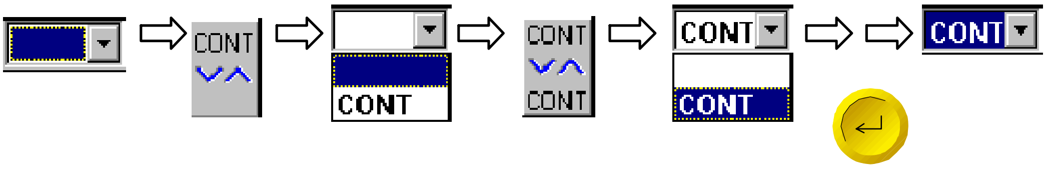
CONT Move the cursor to the next input box. The assignment of the status key changes. The approximate positioning function can be switched on or off using this status key.

Inline form “Exact positioning”.

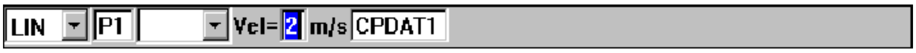
Press the status key on the bottom right-hand side. The menu in the inline form is opened.

Using the status key, choose between “Exact positioning” and “Approximate positioning” (CONT).

Press the Enter key and the menu is closed.



Move the cursor to the box “Vel=”.

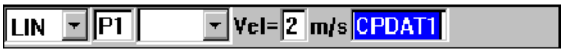


LIN P1 Vel= 2 m/s CPDAT1

Here you can specify the velocity at which the robot should execute the motion. You can either enter the value using the keyboard or alter it using the status key.

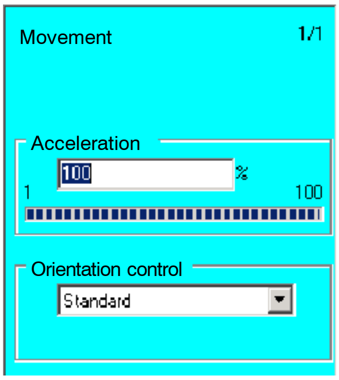
Depending on the length of the path, the degree of acceleration and the approximation distance, it is possible that the programmed velocity will not be reached.

Move the cursor to the next input box, here “CPDAT1”.


LIN P1 Vel= 2 m/s CPDAT1

A parameter list is opened, in which data must be entered characterizing the motion in greater detail. Activate this parameter list using the “Window selection” key. You can either enter the values in the input boxes using the keyboard or alter them using the status key.

CONT


Movement 1/1

Acceleration: 100 % (1 – 100)

Orientation control: Standard

Acceleration

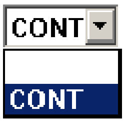
Here you can reduce the acceleration to be used in the motion.

Approximation distance

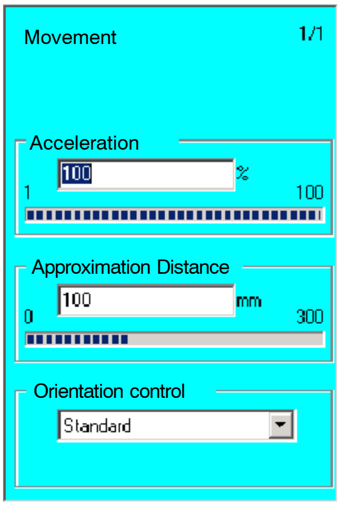
Here you can change the approximate positioning range to be used in the motion.

Orientation control

This option determines whether and how the orientation of the tool is used during the robot motion.


CONT

CONT


Movement 1/1

Acceleration: 100 % (1 – 100)

Approximation Distance: 100 mm (0 – 300)

Orientation control: Standard

Standard: The orientation changes continuously from the start point to the end point. The new orientation is achieved by rotating and pivoting the tool direction. It is possible that problems may occur (e.g. the command acceleration is exceeded) when the robot moves through the extended wrist axis position (wrist singularity).

WristPTP: Here also, the orientation changes continuously from the start point to the end point. The new orientation is achieved by moving the wrist axes in axis-specific mode. There is no possibility of wrist singularities in this case.

ConstantOrientation: The orientation programmed for the start point is also retained for the end point

More detailed information on the subject “Orientation control” may be found in the Programming Handbook, in the main chapter **[Expert Programming]**, chapter **[Motion programming]**.

You can use the softkey “Logic” at any time to insert a so-called logic command into your program, irrespective of the input box that is currently in focus.

Detailed information on logic commands can be found in the section **[Logic]**.

If the softkey “Suggest” is pressed, the program finds the lowest standard point name available in the local data list and enters this name in the open inline form. i.e. if points P1 and P3 are occupied, “P2” will be suggested.

This softkey is used to save the coordinates of the current robot position as the end point following a request for confirmation.

The softkey “Touch Up” allows you, at any moment, to save the current robot coordinates for the program line in which the edit cursor is positioned. You thus have the possibility, for example, to program a sequence of motion instructions and then define the exact end point coordinates later.

The coordinates of the programmed point are saved in the data list.

Now press the softkey “Cmd OK” or the Enter key. The motion is saved.

If the position of the end point has not yet been taught (Touch Up), the current position of the robot is automatically saved.

2.2.3 Circular motions (CIRC)

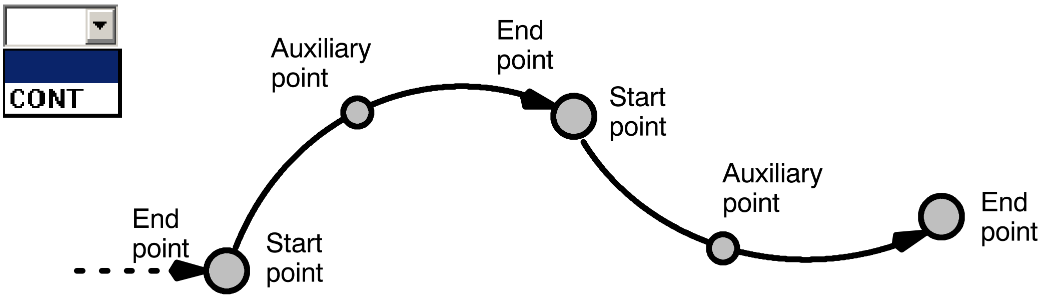
Here, the TCP or workpiece reference point moves to the end point along an arc. The path is defined using start, auxiliary and end points. The end point of a motion instruction serves as the start point for the subsequent motion. The orientation changes uniformly over the whole length of the path.

CIRC motions are used when processing operations are to be executed at a preset velocity along a circular path.

> The start, auxiliary and end points lie in a plane in space. In order for the controller to be able to determine this plane as accurately as possible, these three points should be as far apart as possible. Only the reference point follows the programmed path. The actual tool is able to change its orientation during the motion.
>
> Point names may not begin with “POINT” as this is a keyword reserved by KRL (KUKA Robot Language).

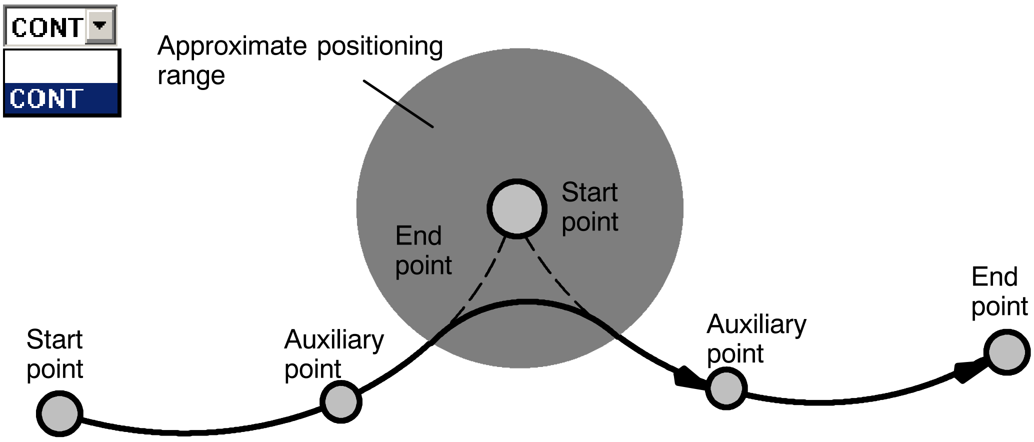
CIRC motion with exact positioning

In the case of CIRC motions with exact positioning, the robot stops exactly at the end point.



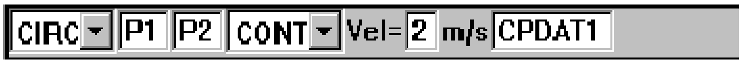
CIRC motion with approximate positioning

During approximate positioning, the controller monitors a so-called approximate positioning range around the end point. When the TCP enters this area, the robot motion immediately proceeds towards the end point of the next motion command.



Programming a CIRC motion

When the option “CIRC” is selected from the menu “Motion”, the inline form for entering the values required for executing this instruction is opened in the program window.



CIRC ▾ P1 P2 CONT ▾ Vel= 2 m/s CPDAT1

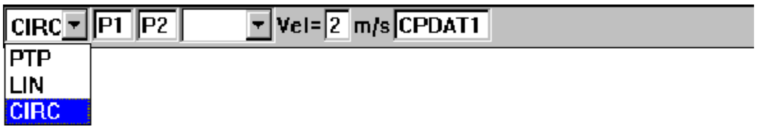
Box name	Function	Range of values
CIRC	Type of motion	PTP, LIN, CIRC
P1	Auxiliary point name	Max. 23 characters
P2	Point name	Max. 23 characters
Tool	Tool number	Nullframe, Tool_Data[1]...[16]
Base	Workpiece number	Nullframe, Base_Data[1]...[16]
External TCP	Robot guides tool/workpiece	True, False
CONT	Approximate positioning ON	“ ”, Cont
Vel=2m/s	Speed	0.001 ... 2 m/s (default: 2 m/s)
PDAT1	Motion parameters	
Acceleration	Acceleration	0 ... 100%
Approximation distance *1	Start or end of the approximate positioning range	0 ... 300 mm
Orientation control	The type of orientation control	Standard, WristPTP, ConstantOrientation
*1 Only available if “CONT” has been switched on		

If the program window is in focus, the various input windows can be selected using the “↓” and “↑” arrow keys. The window that is currently selected is highlighted by a color background.

The program window can be activated by repeatedly pressing the “Window selection” key until the entire window is highlighted in color.

It is possible to abort programming of the CIRC motion at any time by pressing the softkey “Cmd. Abort” or the “ESC” key. The command will **not** be inserted in this case.

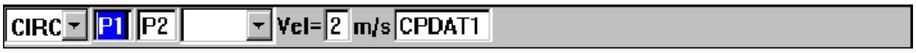
The type of motion is toggled using the softkey “PTP/LIN”.



CIRC ▾ P1 P2 ▾ Vel= 2 m/s CPDAT1

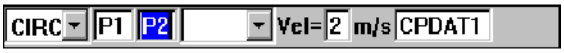
Alternatively, the cursor can be positioned in the first input box. The assignment of the status key changes, thus allowing you to toggle between the different types of motion.

Move the cursor to the next input box, here “P1”.



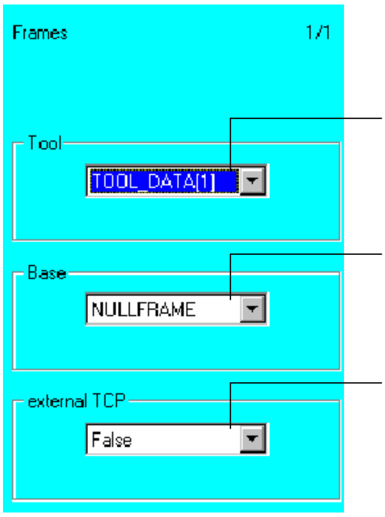
CIRC P1 P2 Vel= 2 m/s CPDAT1

Move the cursor to the next input box, here “P2”.



CIRC P1 P2 Vel= 2 m/s CPDAT1

A status window is opened for entering data relating to the workpiece and tool. Activate this status window using the “Window selection” key.



Frames 1/1

Tool: TOOL_DATA[1]

Base: NULLFRAME

external TCP: False

Select here from the 16 available tools.

Select here one of the 32 saved workpiece coordinate systems (BASE).

Tell the controller here whether the robot is guiding the tool or the workpiece:

Robot guiding tool: external TCP = False

Robot guiding workpiece: external TCP = True

Now activate the program window again using the “Window selection” key.

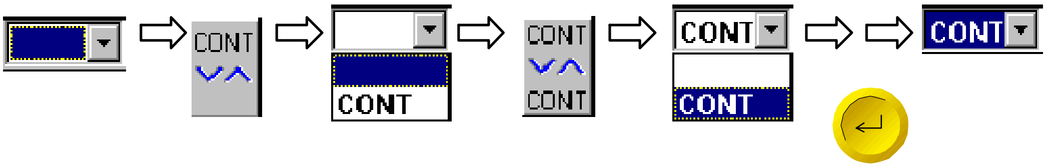
Move the cursor to the input box for approximate positioning. The approximate positioning function can now be switched on or off by means of the corresponding status key.

Inline form “Exact positioning”.

Press the status key on the bottom right-hand side. The menu in the inline form is opened.

Using the status key, choose between “Exact positioning” and “Approximate positioning” (CONT).

Press the Enter key and the menu is closed.



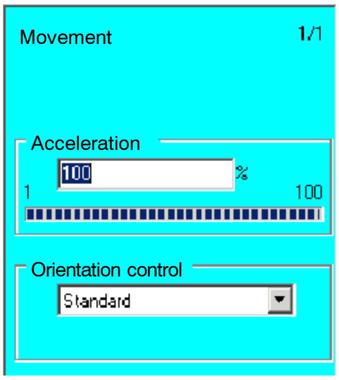
Move the cursor to the box “Vel=”.

Here you can specify the velocity at which the robot should execute the motion. You can either enter the value using the keyboard or alter it using the status key.

Move the cursor to the next input box, “CPDAT1”.

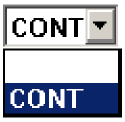
A parameter list is opened, in which data must be entered characterizing the motion in greater detail. Activate this parameter list using the “Window selection” key. You can either enter the values in the input boxes using the keyboard or alter them using the status key.

CONT

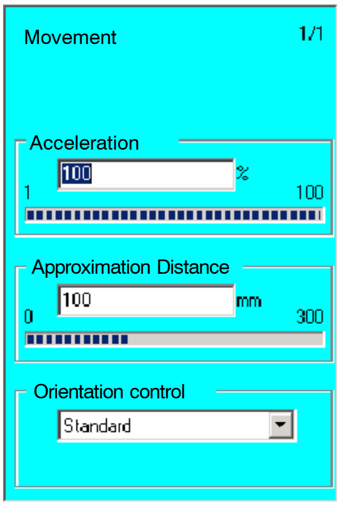

Movement 1/1

Acceleration: 100 % (1 – 100)

Orientation control: Standard


CONT

CONT


Movement 1/1

Acceleration: 100 % (1 – 100)

Approximation Distance: 100 mm (0 – 300)

Orientation control: Standard

Acceleration
Here you can reduce the acceleration to be used in the motion.

Approximation distance
Here you can change the approximate positioning range to be used in the motion.

Orientation control
This option determines whether and how the orientation of the tool is used during the robot motion.

Standard: The orientation changes continuously from the start point to the end point. The new orientation is achieved by rotating and pivoting the tool direction. It is possible that problems may occur (e.g. the command acceleration is exceeded) when the robot moves through the extended wrist axis position (wrist singularity).

WristPTP: Here also, the orientation changes continuously from the start point to the end point. The new orientation is achieved by moving the wrist axes in axis-specific mode. There is no possibility of wrist singularities in this case.

ConstantOrientation: The orientation programmed for the start point is also retained for the end point

If the softkey “Suggest” is pressed, the program finds the lowest standard point name available in the local data list and enters this name in the open inline form. i.e. if points P1 and P3 are occupied, P2 will be suggested.

This softkey is used to save the coordinates of the current robot position as the auxiliary point following a request for confirmation.

The coordinates of the auxiliary point must be saved manually before conclusion of the motion command; they are not automatically saved. If you want to define the point coordinates later, “Teach Aux” can be carried out first in any robot position.

This softkey is used to save the coordinates of the current robot position as the end point following a request for confirmation.

The softkey “Touch Up” allows you, at any moment, to save the current robot coordinates for the program line in which the edit cursor is positioned. You thus have the possibility, for example, to program a sequence of motion instructions and then define the exact end point coordinates later.

The coordinates of the programmed point are saved in the data list.

Finally, press the softkey “Cmd OK” or the Enter key. The motion is saved.

If the position of the end point has not yet been taught (Touch Up), the current robot coordinates are automatically saved.

Please note:

A full circle must be made up of at least two segments to prevent the circle from tilting in the plane.

2.3 Motion parameters

This function allows the monitoring tunnel for collision monitoring to be changed.

[Commands > Moveparams > Torque monitoring]

Further information about torque monitoring (TorqMon) may be found in the Programming Handbook, in the main chapter **[Expert Programming]**, chapter **[Motion programming]**.

2.4 Logic

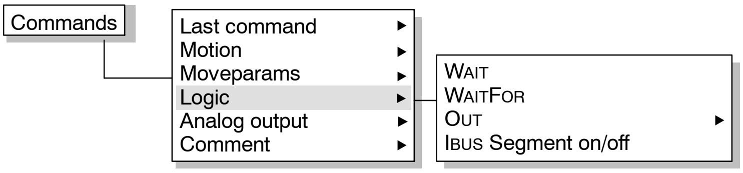
Inputs and outputs of the controller enable the robot system to communicate with its peripheral environment. Specific commands are provided for polling inputs and setting outputs.

It is also possible for wait times to be programmed in order to ensure that operations are completed before the robot starts moving again.

Programming

To program a so-called logic command, you must select a program or load it in the editor.

More detailed information on creating and altering programs can be found in the chapter **[Program editing]**.



You can now make your selection from the logic instructions offered.

2.4.1 Wait functions

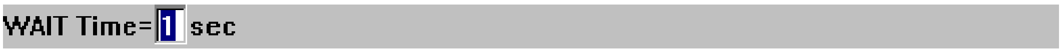
2.4.1.1 Time-dependent wait function (WAIT)

The WAIT function can be used to specify defined wait times.

The selection is made via

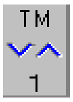
- the menu option “WAIT” or
- the softkey “Logic”.

The following inline form is opened:



WAIT Time=1 sec

Enter the desired wait time using the numeric keypad.



You can alter the default wait time in increments of ± 0.05 seconds using this status key.

Once this softkey has been pressed, the WAIT command is now fully programmed and saved.

Cancel command / functions

If other keys in the softkey bar are pressed, programming of the WAIT function is canceled and the following programming options are offered:

Cmd. Abort:	Only cancellation of the programming (also possible using the "ESC" key). The command is **not** saved.
Motion:	Motion commands
IBUS:	Coupling/decoupling of an Interbus segment
Comment:	Comment lines
WAIT FOR:	Signal-dependent wait function

2.4.1.2 Signal-dependent wait function (WAIT FOR)

The WAIT FOR function allows the programming of a signal-dependent wait for a specified system variable or user-defined variable.

These variables, called operands, can be linked together within a single command by means of logic operators and corresponding use of brackets.

The wait condition can be programmed, for example, in the following general form:

WAIT FOR (IN1 OR IN2 OR IN3) AND (NOT OUT1 OR OUT 2) OR NOT (FLAG1)

The following operators are available:

AND
OR
EXOR (exclusive OR)

They can be called by means of their corresponding softkeys.

The **NOT command** can also be entered by means of a separate softkey.

More detailed information on the use of operators can be found in the **Programming Handbook** in the documentation **Expert Programming**, chapter **[Variables and declarations]**.

The following lists are available for entering operands:

System list: The system variables $IN, $OUT, $CYCFLAG, $TIMER_FLAG and $FLAG are available as standard in a window in the inline form. If long text names already exist, these are displayed together with the system variable, e.g.:

WAIT FOR (OUT 144 **"adhesive_valve"**).

User list: Further operands may be obtained from a list created beforehand by the user. In this case, it is possible, for example, to use more extensive expressions, such as:

WAIT FOR (**USER_INT >=100**) EXOR (**USER_BOOL**),

You can toggle between the two operand lists by pressing the softkeys "System List" and "User List".

This allows, for example, the programming of mixed expressions, such as:

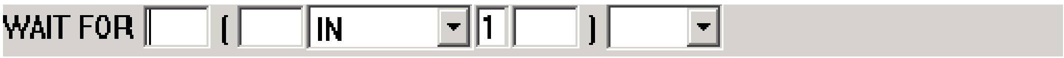
WAIT FOR (**USER_FIELD[COUNT-1]<>USER_INT**) AND (**NOT IN 17 "pressure_valve"**)

Selection

When the WAIT FOR command is selected, via

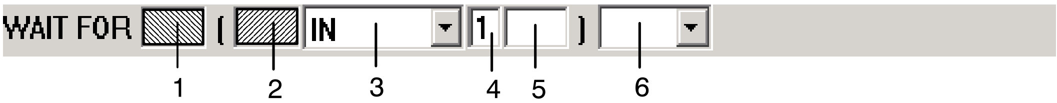
- the menu option “WAIT FOR” or
- the softkeys “Logic -> WAIT FOR”,

the following inline form is opened:



Entry

Corresponding parameters in the inline form are entered using the status key at the bottom right-hand side of the screen or by means of the softkey bar, the assignment of which adapts according to the position of the cursor.



In particular, the double function of boxes 1 and 2 must be taken into consideration. Here it is possible not only to define the status of the operand, but also to set the brackets accordingly. This is dependent on the position of the cursor in the respective box.

Box	Function	Values	Remarks
1	Operand status	“ ” NOT	Softkey: AndTerm, OrTerm, ExOrTerm Insertion of an external logic operation. Softkey: Not Negation of the Boolean expression.
2	Operand status	“ ” NOT	Softkey: AndOp, OrOp, ExOrOp Insertion of an internal logic operation. Softkey: Not Negation of the Boolean expression.
3	Operand name	IN OUT CYCFLAG TIMER FLAG User variable	System list: $IN[1] ... $IN[1024] $OUT[1] ... $OUT[1024] $CYCFLAG[1] ... $CYCFLAG[32] $TIMER_FLAG[1] ... $TIMER_FLAG[16] $FLAG[1] ... $FLAG[1024] User list: It is possible to enter a user-defined name.
4	Operand no.	1 ... 1024	
5	Long text name	“ ” Existing name	By pressing the softkey “Long text”, the long text name can be programmed in Expert mode with the system list activated.
6	Approximate positioning	“ ” CONT	Execution with advance run stop. Execution without advance run stop.

Once this softkey has been pressed, the command is now fully programmed and saved.

Positioning of brackets

In the inline form you can choose between two different types of bracketing:

- External logic operation: The operator is situated between the bracketed expressions. Example: WAIT FOR (IN1) **AND** (IN2)
- Internal logic operation: The operator is situated inside a bracketed expression. Example: WAIT FOR (IN1 **AND** IN2)

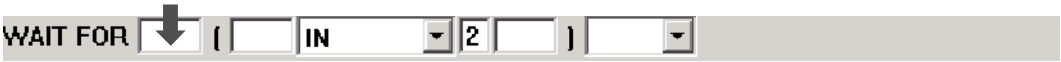
Mixed forms, such as WAIT FOR (IN1 **AND** IN2) **OR** (IN3) are possible.

A maximum of 12 operands can be linked in a form.

The logic operations cannot be nested. There is only **one** level of bracketing!

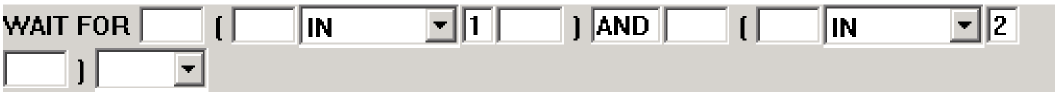
<u>Insertion of an external logic operation</u>: WAIT FOR **(IN1) AND** (IN2)

In the inline form illustrated, position the cursor in box 1.

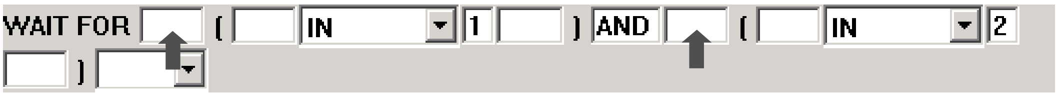


The new expression is inserted at the cursor position.

Pressing the softkey “AndTerm + ” expands the inline form as follows:

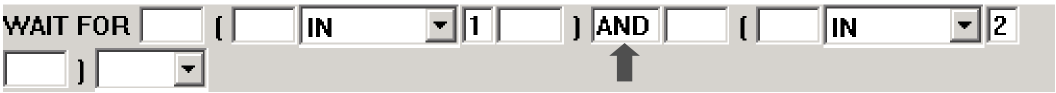


If the cursor is moved to the positions indicated by the arrows, further external expressions can be inserted.



<u>Deletion of an external logic operation / change of operator</u>:

If the cursor is situated in the operator box AND,



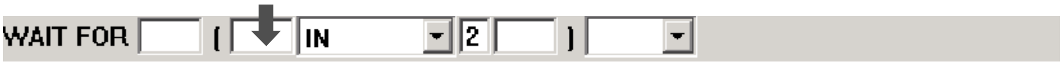
the following options are displayed in the softkey bar:

ToggleOp: Change of operator

Operator-: Deletion of logical operation

Insertion of an internal logic operation: WAIT FOR (**IN1 AND** IN2)

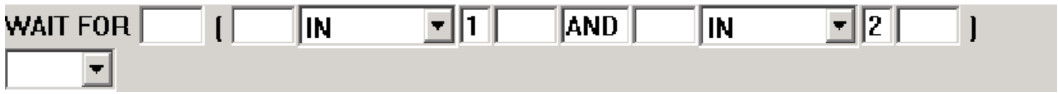
In the inline form, position the cursor in box 2.



WAIT FOR (IN 2)

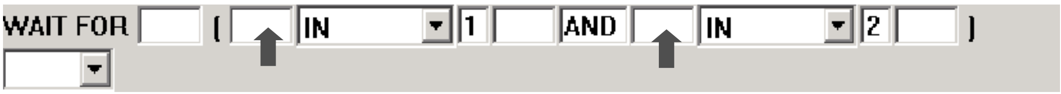
The new expression is inserted at the cursor position.

AndOp+: Pressing the softkey expands the inline form as follows:



WAIT FOR (IN 1 AND IN 2)

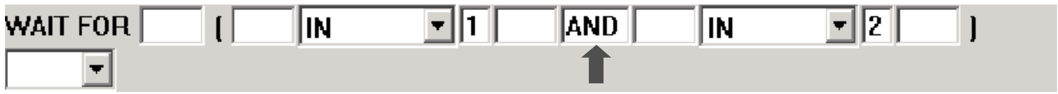
If the cursor is moved to the positions indicated by the arrows, further internal expressions can be inserted.



WAIT FOR (IN 1 AND IN 2)

Deletion of an internal logic operation / change of operator:

If the cursor is situated in the operator box AND,



WAIT FOR (IN 1 AND IN 2)

the following options are displayed in the softkey bar:

ToggleOp: Change of operator

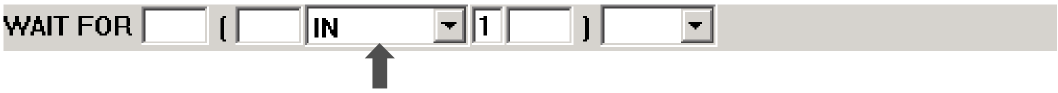
Operator-: Deletion of logical operation

User list

The user list allows you to use application-specific operands directly. These may include variables and other types of data object (functions, etc.) defined by the user in KRL.

The precondition is that these variables and/or functions have been defined and initialized in the control program.

Once the cursor is positioned in box 3 in the inline form,

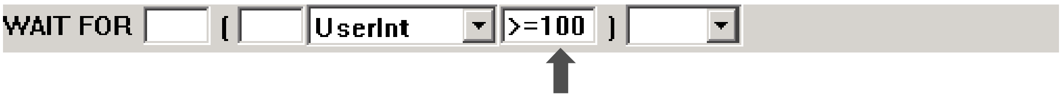


WAIT FOR (IN 1)

and the softkey is pressed,

User List: you can select the desired variable.

The input situation changes, as illustrated in this example:



WAIT FOR (UserInt >=100)

The numeric input box for the system list and the input box for the long text name are no longer present. Now, however, you can program complex relationships in the box indicated (see arrow) in which text can be entered freely.

The result of the user-specific variables together with the freely programmable box must always give a Boolean expression.

No user list available:

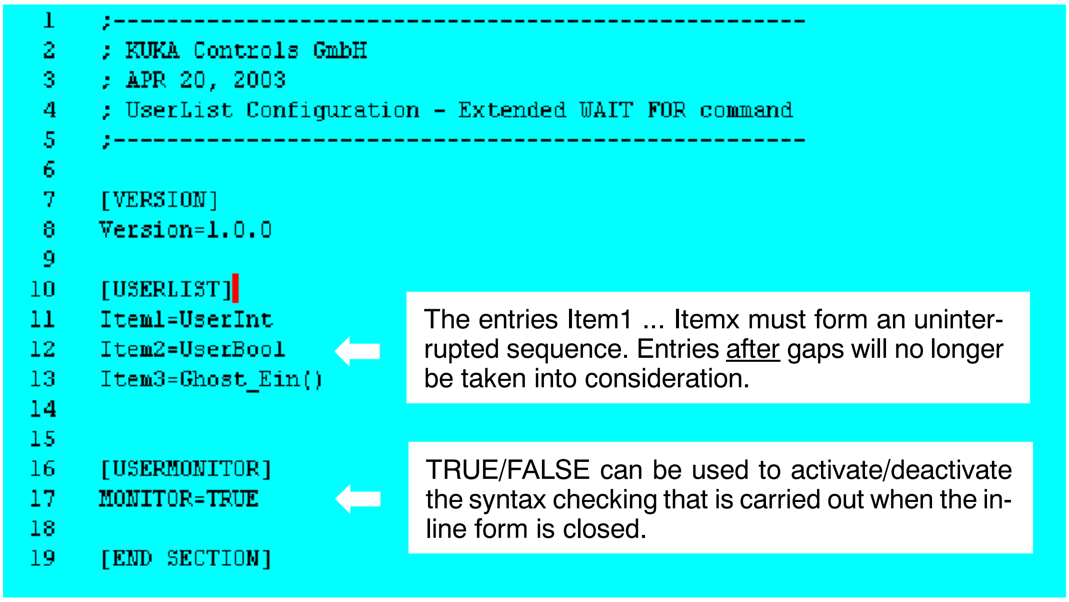
If it is not possible to access the user list, a message appears. In this case, the corresponding user list must be created.

Create user list:

The user-specific list must be created in C:\KRC\ROBOTER\INIT\WAIT_USERLIST.INI.

> More detailed information can be found in the **Programming Handbook** in the documentation **Expert Programming**, chapter **[Variables and declarations]**.

Example of the entry of variables in the file WAIT_UserList.ini:



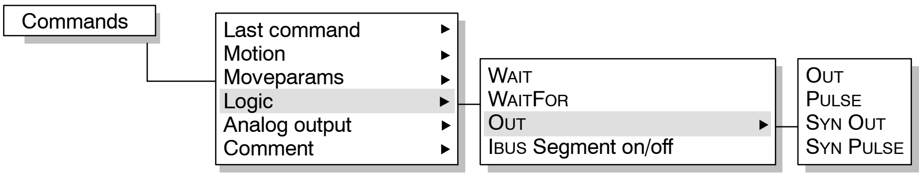
```
;----------------------------------------------------
; KUKA Controls GmbH
; APR 20, 2003
; UserList Configuration - Extended WAIT FOR command
;----------------------------------------------------

[VERSION]
Version=1.0.0

[USERLIST]
Item1=UserInt
Item2=UserBool
Item3=Ghost_Ein()


[USERMONITOR]
MONITOR=TRUE

[END SECTION]
```

The entries Item1 ... Itemx must form an uninterrupted sequence. Entries after gaps will no longer be taken into consideration.

TRUE/FALSE can be used to activate/deactivate the syntax checking that is carried out when the inline form is closed.

2.4.2 Switching functions

After selecting the menu item "OUT", a submenu with further options is opened.

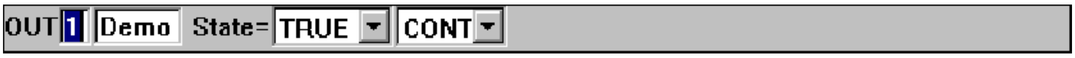


This menu offers you various switching functions, which are described in detail below.

2.4.2.1 Simple switching function (OUT)

This function sets a selected output to TRUE or FALSE.

When the menu option "OUT" is selected, an inline form is opened in the programming window for entering the parameters that are required for executing the simple switching function.



Box name	Function	Range of values
1	Output	1 ... 1024
Demo	Long text	Name of the input or output
TRUE	State	TRUE, FALSE
CONT	Approximate positioning	" ", CONT

It is possible to abort programming of the switching function at any time by pressing the softkey "Cmd. Abort" or the "ESC" key. The command will **not** be saved in this case.

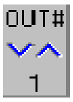
If the program window is in focus, the various input windows can be selected using the "↓" and "↑" arrow keys. The window that is currently selected is highlighted by a color background.

The program window can be activated by repeatedly pressing the "Window selection" key until the entire window is highlighted in color.

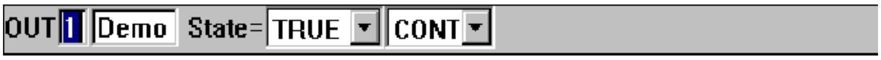
Pressing the softkey "PULSE" switches immediately to programming of the simple pulse function. This cancels programming of the simple switching function.

Pressing the softkey "SYNOUT" switches immediately to programming of a path-dependent switching function. This cancels programming of the simple switching function.

The long text belonging to the corresponding output can be modified by pressing the "Longtext" softkey. This option is not available below the user group "Expert".

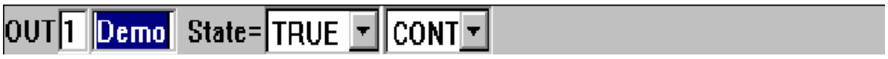


Move the cursor to the input box “OUT”. The assignment of the status key at the bottom right of the display changes. Using the numeric keypad, specify the output that is to be set. It is also possible to change the value shown in the input box using the status key.

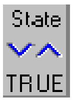


OUT 1 Demo State= TRUE CONT

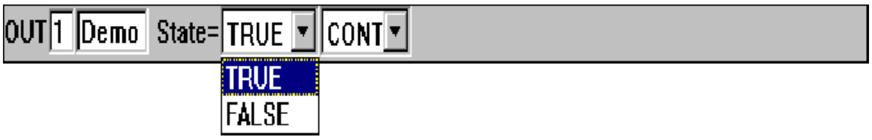
The long text of the output can be modified in the next input box. To do so, the user group “Expert” must be active. Press the “Longtext” softkey and enter the desired name.



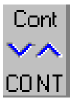
OUT 1 Demo State= TRUE CONT



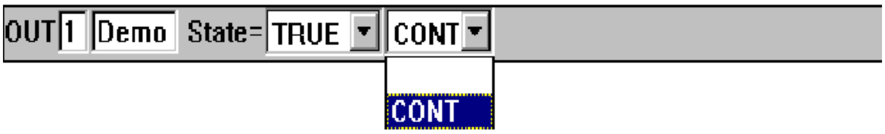
Move the cursor to the input box “State”. The assignment of the status key at the bottom right of the display changes. With this status key, you can specify the state to which the selected output is to be set.



OUT 1 Demo State= TRUE CONT



Move the cursor to the input box “CONT”. The assignment of the status key at the bottom right of the display changes. If you set “CONT”, the output is set with the advance run pointer. If “CONT” is not selected, the output is set with the main run pointer, i.e. as soon as the advance run pointer reaches this function, it is stopped until the main run pointer has also reached this function.



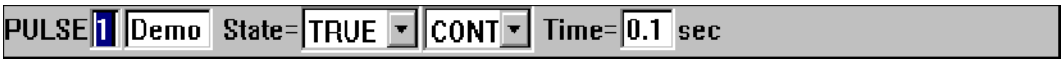
OUT 1 Demo State= TRUE CONT

Now press the softkey “Cmd OK” or the Enter key. The command is now fully programmed and inserted.

2.4.2.2 Simple pulse function (PULSE)

This function sets a selected output to TRUE/FALSE for the duration of a defined pulse time.

When the menu option “PULSE” is selected, an inline form is opened in the programming window for entering the parameters that are required for executing the simple pulse function.



PULSE 1 Demo State= TRUE CONT Time= 0.1 sec

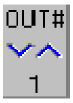
Box name	Function	Range of values
1	Output	1...1024
Demo	Long text	Name of the input or output
TRUE	State	TRUE, FALSE
CONT	Approximate positioning	“ ”, CONT
0.1	Length of the pulse	0.1 ... 3 s

It is possible to abort programming of the pulse function at any time by pressing the softkey “Cmd. Abort” or the “ESC” key. The command will **not** be saved in this case.

Pressing the softkey “OUT” switches immediately to programming of the simple switching function. This cancels programming of the simple pulse function.

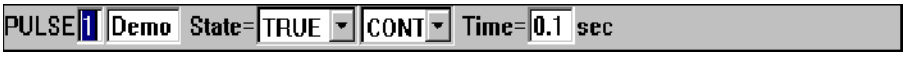
Pressing the softkey “SYNPULSE” switches immediately to programming of the path-dependent pulse function. This cancels programming of the simple pulse function.

The long text belonging to the corresponding output can be modified by pressing the “Longtext” softkey. This option is not available below the user group “Expert”.



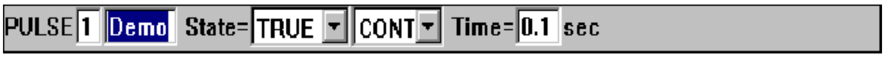
OUT#
1

Move the cursor to the input box “PULSE”. The assignment of the status key at the bottom right of the display changes. Using the numeric keypad, specify the output that is to be set. It is also possible to change the value shown in the input box using the status key.

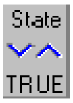


PULSE 1 Demo State= TRUE CONT Time= 0.1 sec

The long text of the output can be modified in the next input box. To do so, the user group “Expert” must be active. Press the “Longtext” softkey and enter the desired name.

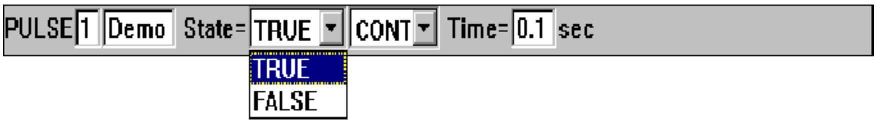


PULSE 1 Demo State= TRUE CONT Time= 0.1 sec



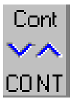
State
TRUE

Move the cursor to the input box “State”. The assignment of the status key at the bottom right of the display changes. With this status key, you can specify the state to which the selected output is to be set.



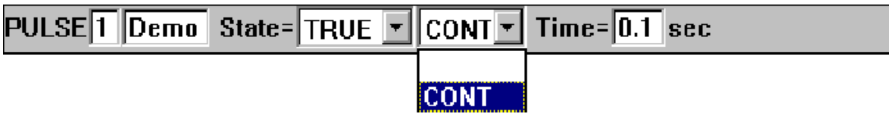
PULSE 1 Demo State= TRUE CONT Time= 0.1 sec

TRUE
FALSE

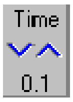


Cont
CONT

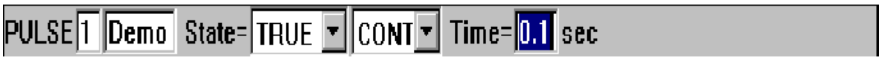
Move the cursor to the input box “CONT”. The assignment of the status key at the bottom right of the display changes. If you set “CONT”, the function is executed when reached by the advance run pointer. If “CONT” is not selected, the function is executed when the main run pointer arrives, i.e. as soon as the advance run pointer reaches this function, it is stopped until the main run pointer has also reached this function.



PULSE 1 Demo State= TRUE CONT Time= 0.1 sec



Move the cursor to the input box “Time”. The assignment of the status key at the bottom right of the display changes. You can set the length of the pulse to between 0.1 and 3 seconds using this status key. The increment is 0.1 seconds.

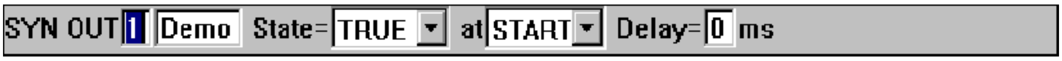


PULSE 1 Demo State= TRUE CONT Time= 0.1 sec

Now press the softkey “Cmd OK” or the Enter key. The command is now fully programmed and inserted.

2.4.2.3 Path-dependent switching function (SYN OUT)

After selecting the menu option “SYN OUT”, the inline form for entering the parameters that are required for executing the path-dependent switching function is opened in the program window.



SYN OUT 1 Demo State= TRUE at START Delay= 0 ms

Box name	Function	Range of values
1	Output	1...1024
Demo	Long text	Name of the input or output
TRUE	State	TRUE, FALSE
START	Time at which the switching function is executed	START, END, PATH
	Distance from the end point at which the output is set *1)	-2000 ... 2000 mm
0	Delay of the switching action	-1000 ... 1000 ms
*1) Only available if “PATH” is selected as the start option		

It is possible to abort programming of the switching function at any time by pressing the softkey “Cmd. Abort” or the “ESC” key. The command will **not** be inserted in this case.

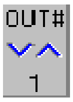
If the program window is in focus, the various input windows can be selected using the “↓” and “↑” arrow keys. The window that is currently selected is highlighted by a color background.

The program window can be activated by repeatedly pressing the “Window selection” key until the entire window is highlighted in color.

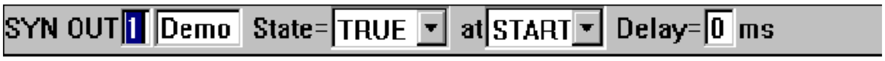
Pressing the softkey “SYNPULSE” switches immediately to programming of the path-dependent pulse function. Programming of the path-dependent switching function is canceled first.

Pressing the softkey “OUT” switches immediately to programming of the simple switching function. Programming of the path-dependent switching function is canceled first.

The long text belonging to the corresponding output can be modified by pressing the “Longtext” softkey. This option is not available below the user group “Expert”.

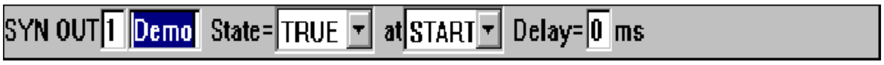


Move the cursor to the input box “SYN OUT”. Using the numeric keypad, specify the output that is to be set. It is also possible to change the value shown in the input box using the status key.

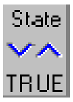


SYN OUT 1 Demo State= TRUE at START Delay= 0 ms

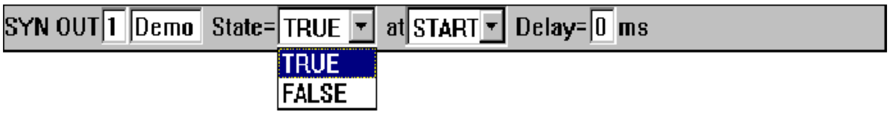
The long text of the output can be modified in the next input box. To do so, the user group “Expert” must be active. Press the “Longtext” softkey and enter the desired name.



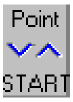
SYN OUT 1 Demo State= TRUE at START Delay= 0 ms



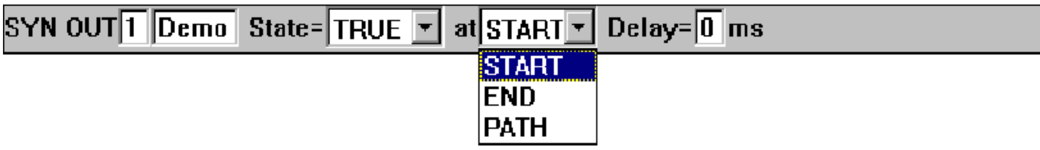
Move the cursor to the input box “State”. The assignment of the status key at the bottom right of the display changes. With this status key, you can specify the state to which the selected output is to be set.



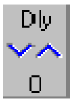
SYN OUT 1 Demo State= TRUE at START Delay= 0 ms



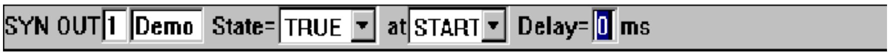
Move the cursor to the input box “at”. The assignment of the status key at the bottom right of the display changes. With this status key, you can specify the point of the path segment at which the selected output is to be set.



SYN OUT 1 Demo State= TRUE at START Delay= 0 ms

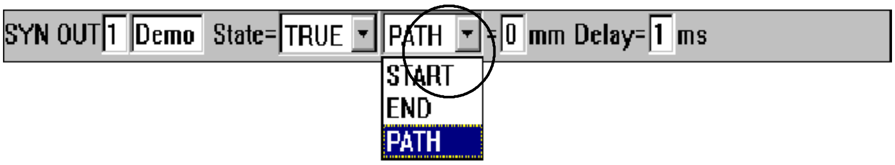


Move the cursor to the input box “Delay”. The assignment of the status key at the bottom right of the display changes. With this status key, you can shift the time at which the selected output is to be set, by any amount between the start and end points of the path segment. The increment is 1 ms. You can also enter this value using the numeric keypad.



SYN OUT 1 Demo State= TRUE at START Delay= 0 ms

If, on the other hand, you want to set the output in relation to the path, you must select the option “PATH” in the input box “at”.



SYN OUT 1 Demo State= TRUE PATH = 0 mm Delay= 1 ms

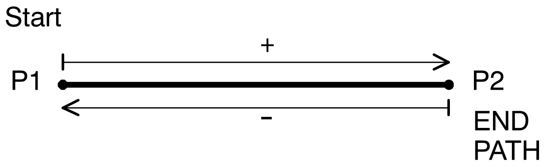
An additional input box then appears in the inline form in which you can specify the distance from the end point at which the output should be set. If, for example, you want the output to be set between the start point and the end point, you must enter a value with a negative sign.

The current value of the variable $DIST_NEXT can be displayed and altered by means of the menu function “Monitor – Variable – Single”.

The motion commands (type: LIN) before and after the SYN OUT function are important for determining locally the start and end points and the points along the path.

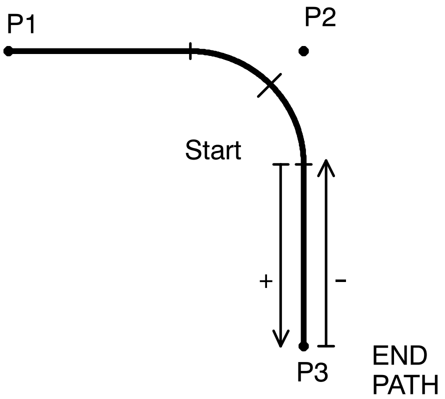
- Case 1: the start point and end point are exact positioning points

```
Program:
LIN P1
SYN OUT
LIN P2
```



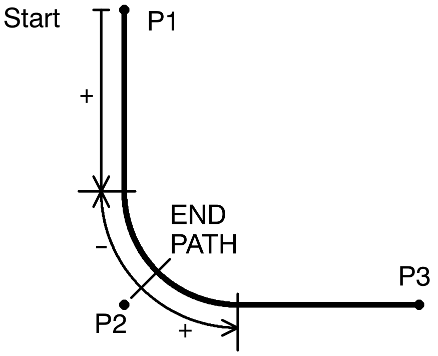
- Case 2: the start point is approximated, the end point is an exact positioning point

```
Program:
LIN P1
LIN P2 CONT
SYN OUT
LIN P3
```



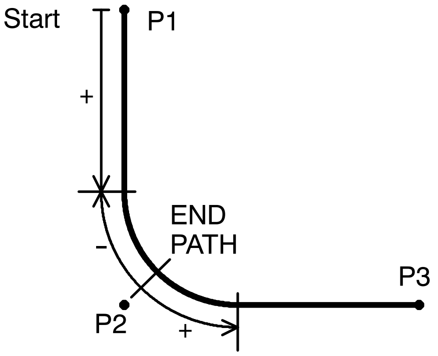
- Case 3: the end point is approximated, the start point is an exact positioning point

```
Program:
LIN P1
SYN OUT
LIN P2 CONT
LIN P3
```



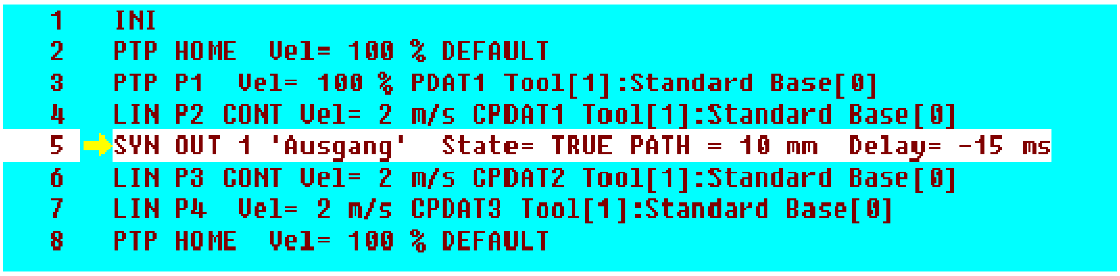
- Case 4: the start point and end point are approximated

```
Program:
LIN P1
SYN OUT
LIN P2 CONT
LIN P3
```



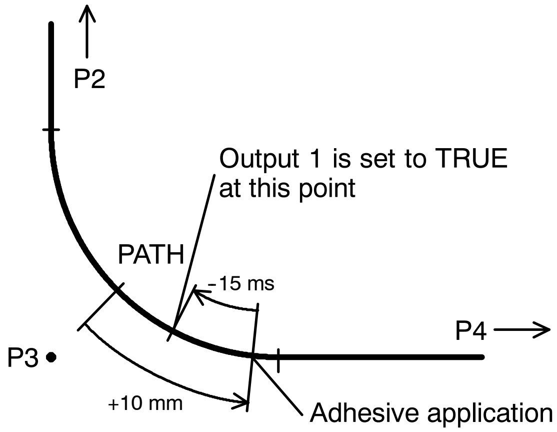
Example adhesive application:

An adhesive gun is to start applying adhesive at a certain point on the workpiece. For this purpose the command SYN OUT is set. When output 1= TRUE, the adhesive gun is activated. PATH=10 mm defines the point at which the adhesive application is to begin. The adhesive gun itself, however, must be activated approximately 15 ms earlier, as a certain advance time is required before the adhesive begins to flow. The delay time is thus set to -15 ms.



```
INI
PTP HOME  Vel= 100 % DEFAULT
PTP P1  Vel= 100 % PDAT1 Tool[1]:Standard Base[0]
LIN P2 CONT Vel= 2 m/s CPDAT1 Tool[1]:Standard Base[0]
SYN OUT 1 'Ausgang'  State= TRUE PATH = 10 mm  Delay= -15 ms
LIN P3 CONT Vel= 2 m/s CPDAT2 Tool[1]:Standard Base[0]
LIN P4  Vel= 2 m/s CPDAT3 Tool[1]:Standard Base[0]
PTP HOME  Vel= 100 % DEFAULT
```

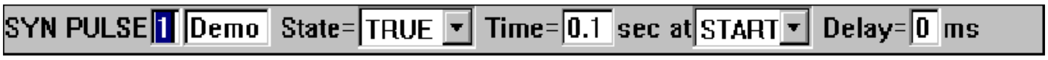
Extract from the illustration relating to “case 4”



Now press the softkey “Cmd OK” or the Enter key. The command is now fully programmed and inserted.

2.4.2.4 Path-dependent pulse function (SYN PULSE)

When the menu option “SYN PULSE” is selected, an inline form is opened in the programming window for entering the parameters that are required for executing the path-dependent pulse function.



SYN PULSE 1 Demo State= TRUE Time= 0.1 sec at START Delay= 0 ms

The assignment of the softkey bar changes at the same time:

Box name	Function	Range of values
1	Output	1...1024
Demo	Long text	Name of the input or output
TRUE	State	TRUE, FALSE
0.1	Pulse duration	0.1 ... 3 s
START	Time at which the pulse function is executed	START, END, PATH
	Distance from the end point at which the output is set *1)	-2000 ... 2000 mm
0	Delay of the pulse function	-1000 ... 1000 ms
*1) Only available if “PATH” is selected as the start option		

It is possible to abort programming of the pulse function at any time by pressing the softkey “Cmd. Abort” or the “ESC” key. The command will **not** be saved in this case.

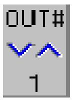
If the program window is in focus, the various input windows can be selected using the “↓” and “↑” arrow keys. The window that is currently selected is highlighted by a color background.

The program window can be activated by repeatedly pressing the “Window selection” key until the entire window is highlighted in color.

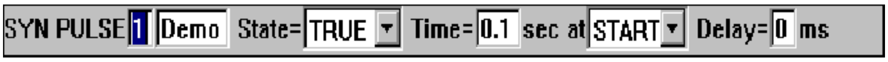
Pressing the softkey “SYNOUT” switches immediately to programming of the path-dependent switching function. This cancels programming of the path-dependent pulse function.

Pressing the softkey “PULSE” switches immediately to programming of the simple pulse function. This cancels programming of the path-dependent pulse function.

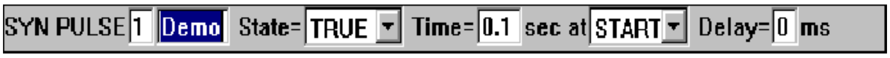
The long text belonging to the corresponding output can be modified by pressing the “Longtext” softkey. This option is not available below the user group “Expert”.

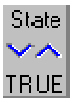


Move the cursor to the input box “SYN PULSE”. The assignment of the status key at the bottom right of the display changes. Using the numeric keypad, specify the output that is to be set. It is also possible to change the value shown in the input box using the status key.

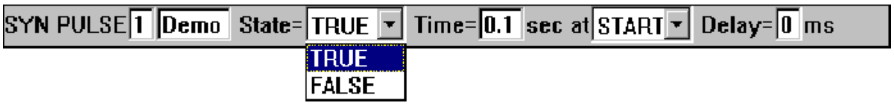

SYN PULSE 1 Demo State= TRUE Time= 0.1 sec at START Delay= 0 ms

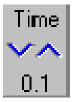
The long text of the output can be modified in the next input box. To do so, the user group “Expert” must be active. Press the “Longtext” softkey and enter the desired name.


SYN PULSE 1 Demo State= TRUE Time= 0.1 sec at START Delay= 0 ms

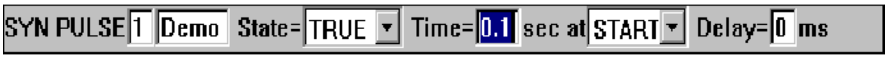


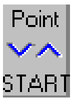
Move the cursor to the input box “State”. The assignment of the status key at the bottom right of the display changes. With this status key, you can switch between the states to which the selected output can be set.


SYN PULSE 1 Demo State= TRUE Time= 0.1 sec at START Delay= 0 ms
TRUE
FALSE

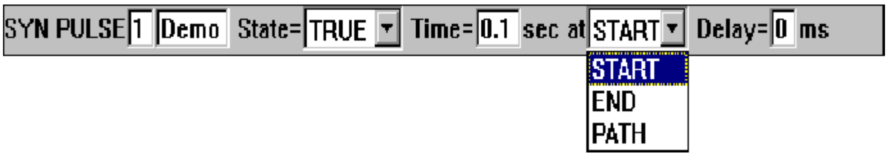


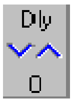
Move the cursor to the input box “Time”. The assignment of the status key at the bottom right of the display changes. You can set the length of the pulse within the range of 0.1 to 3 seconds using this status key. The increment is 0.1 seconds. You can also enter the value using the numeric keypad.


SYN PULSE 1 Demo State= TRUE Time= 0.1 sec at START Delay= 0 ms

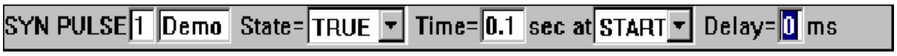


Move the cursor to the input box “at”. The assignment of the status key at the bottom right of the display changes. With this status key, you can specify the point of the path segment at which the selected output is to be set.

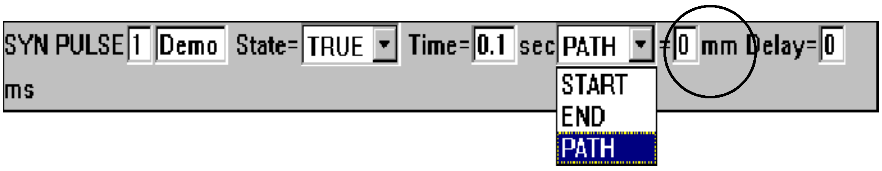

SYN PULSE 1 Demo State= TRUE Time= 0.1 sec at START Delay= 0 ms



Move the cursor to the input box “Delay”. The assignment of the status key at the bottom right of the display changes. With this status key, you can shift the time at which the selected output is to be set, by any amount between the start and end points of the path segment. The increment is 1 ms. You can also enter the value using the numeric keypad.


SYN PULSE 1 Demo State= TRUE Time= 0.1 sec at START Delay= 0 ms

If, on the other hand, you want to set the output in relation to the path, you must select the option “PATH” in the input box “at”.


SYN PULSE 1 Demo State= TRUE Time= 0.1 sec PATH = 0 mm Delay= 0 ms

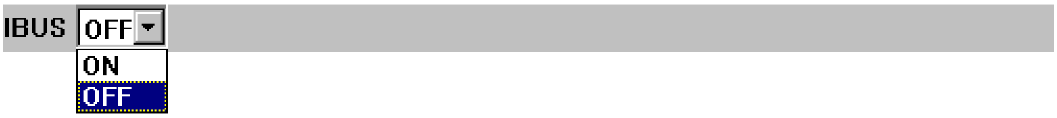
An additional input box then appears in the inline form in which you can specify the distance from the end point at which the output should be set. If, for example, you want the output to be set between the start point and the end point, you must enter a negative value.

The motion commands (type: LIN and CIRC) before and after the SYN PULSE function are important for determining locally the start and end points and the points along the path. Compare cases 1 to 4 for the previous function SYN OUT!

Now press the softkey “Cmd OK” or the Enter key. The command is now fully programmed and inserted.

2.4.3 Coupling and decoupling an Interbus segment (IBUS Segment on/off)

When the menu option “IBUS Segment on/off” is selected, the inline form for programming the coupling and decoupling of the INTERBUS segment is opened in the programming window.



IBUS OFF

Box name	Function	Range of values
OFF	Coupling and decoupling an IBus segment	ON, OFF

It is possible to abort programming of this function at any time by pressing the softkey “Cmd Abort” or the “ESC” key. The command will **not** be saved in this case.

If the program window is in focus, the various input windows can be selected using the “↓” and “↑” arrow keys. The window that is currently selected is highlighted by a color background.

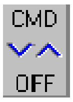
The program window can be activated by repeatedly pressing the “Window selection” key until the entire window is highlighted in color.

Pressing the softkey “Motion” switches immediately to programming of motion instructions. Programming of the function is thereby canceled.

Pressing the softkey “WAIT” switches immediately to programming of the wait functions. Programming of the function is thereby canceled.

Pressing the softkey “Comment” switches immediately to programming of a comment line. Programming of the function is thereby canceled.

Pressing the softkey “WAIT FOR” switches immediately to programming of the wait functions. Programming of the function is thereby canceled.



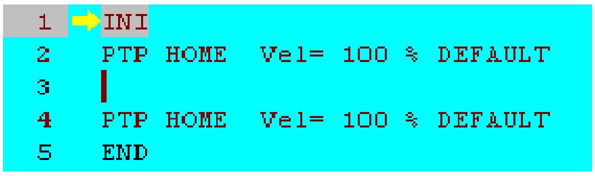
By pressing the status key “CMD”, you can switch between coupling and decoupling of the INTERBUS.

Now press the softkey “Cmd OK” or the Enter key. The command is now fully programmed and inserted.

2.5 Analog output

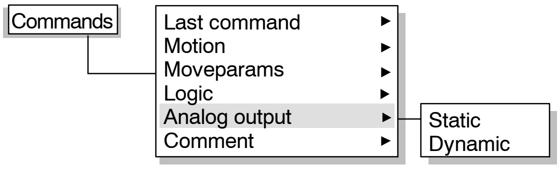
The eight analog outputs of the robot controller are set under program control using this function.

Programming



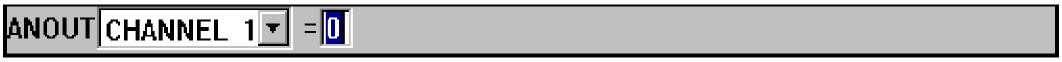
```
INI
PTP HOME  Vel= 100 % DEFAULT

PTP HOME  Vel= 100 % DEFAULT
END
```

Pay attention to the position of the edit cursor. The next program line created by you will be inserted as a new line after the cursor.



2.5.1 Static

This option is used to set an analog output with a fixed value. After selecting the menu option, the inline form for entering the necessary values is opened in the program window.



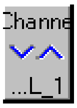
ANOUT CHANNEL 1 = 0

Box name	Function	Range of values
CHANNEL 1	Analog output	1 ... 32
0	Output voltage	0 ... 1

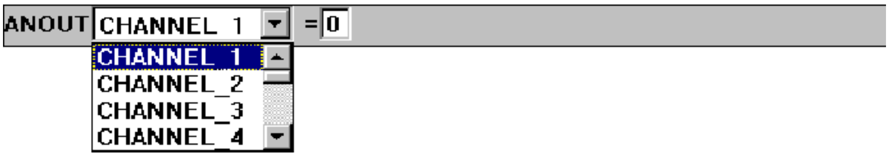
If the program window is in focus, the various input windows can be selected using the "↓" and "↑" arrow keys. The window that is currently selected is highlighted by a color background.

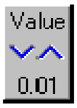
The program window can be activated by repeatedly pressing the "Window selection" key until the entire window is highlighted in color.

It is possible to abort programming of the function at any time by pressing the softkey "Cmd. Abort" or the "ESC" key. The command will **not** be inserted in this case.

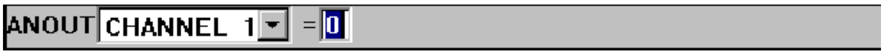


Move the cursor to the left-hand input box. The assignment of the status key at the bottom right of the display changes. Here you can specify which of the 32 analog outputs you want to set.


ANOUT CHANNEL_1 =0



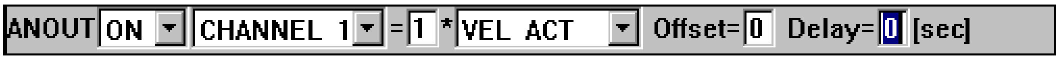
Move the cursor to the next input box, here “0”. The assignment of the status key at the bottom right of the display changes. Using the numeric keypad, enter the value the analog output should take. Here, it is also possible to change the value shown in the input box using the status key. The increment is 10 mV.


ANOUT CHANNEL_1 =0

Finally, press the softkey “Cmd OK” or the Enter key. The command line is then inserted into the program.

2.5.2 Dynamic

This option is used to set an analog output depending on the velocity or technology. After selecting the menu option, the inline form for entering the necessary values is opened in the program window.

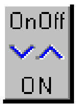

ANOUT ON CHANNEL_1 =1 * VEL_ACT Offset=0 Delay=0 [sec]

Box name	Function	Range of values
ON	Switching simulation mode on and off	ON, OFF
CHANNEL 1	Analog output	1 ... 32
1	Multiplier	0 ... 10
VEL ACT	Velocity or technology parameters	VEL ACT, TECHVAL1 ... TECHVAL6
0	Offset voltage	-1 ... 1
0	Delay	-0.2 ... 0.5 s

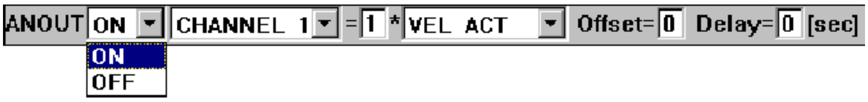
If the program window is in focus, the various input windows can be selected using the “↓” and “↑” arrow keys. The window that is currently selected is highlighted by a color background.

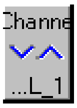
The program window can be activated by repeatedly pressing the “Window selection” key until the entire window is highlighted in color.

It is possible to abort programming of the function at any time by pressing the softkey “Cmd. Abort” or the “ESC” key. The command will **not** be inserted in this case.

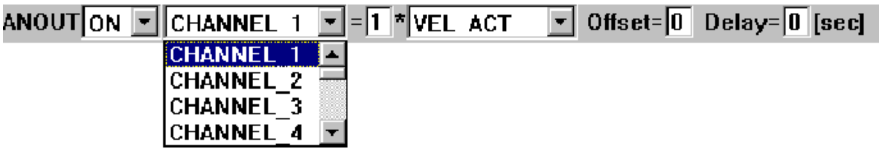


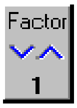
Move the cursor to the left-hand input box. The assignment of the status key at the bottom right of the display changes. Here you can specify whether you wish to switch the analog output on or off.


ANOUT ON CHANNEL 1 = 1 * VEL ACT Offset= 0 Delay= 0 [sec]

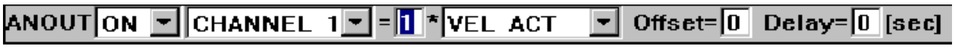


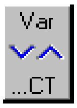
Move the cursor to the next input box, here “Channel 1”. The assignment of the status key at the bottom right of the display changes. Here you can specify which of the eight analog outputs you want to set or reset.


ANOUT ON CHANNEL 1 = 1 * VEL ACT Offset= 0 Delay= 0 [sec]

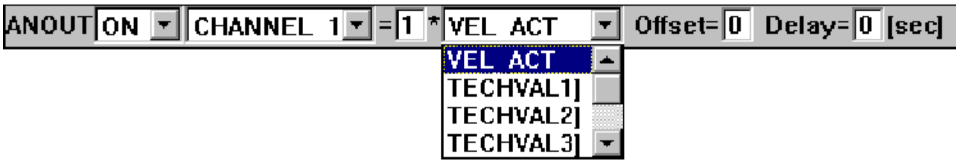


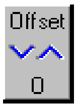
Move the cursor to the next input box, here “1”. The assignment of the status key at the bottom right of the display changes. Using the numeric keypad, enter the factor by which the corresponding velocity/technology parameters are to be multiplied. Here, it is also possible to change the value shown in the input box using the status key. The increment is 0.05.


ANOUT ON CHANNEL 1 = 1 * VEL ACT Offset= 0 Delay= 0 [sec]

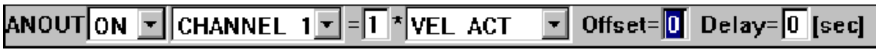


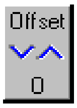
Move the cursor to the next input box, here “VEL_ACT”. The assignment of the status key at the bottom right of the display changes. Here you can specify the velocitytechnology parameters with which the selected analog output is to be combined.


ANOUT ON CHANNEL 1 = 1 * VEL ACT Offset= 0 Delay= 0 [sec]

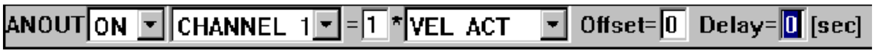


Move the cursor to the next input box, here “Offset”. The assignment of the status key at the bottom right of the display changes. Using the numeric keypad, enter the value of the offset voltage for the selected analog output. Here, it is also possible to change the value shown in the input box using the status key. The increment is 100 mV.


ANOUT ON CHANNEL 1 = 1 * VEL ACT Offset= 0 Delay= 0 [sec]



Move the cursor to the next input box, here “Delay”. The assignment of the status key at the bottom right of the display changes. Using the numeric keypad, enter a delay value. Here, it is also possible to change the value shown in the input box using the status key. The increment is 1/100 seconds.

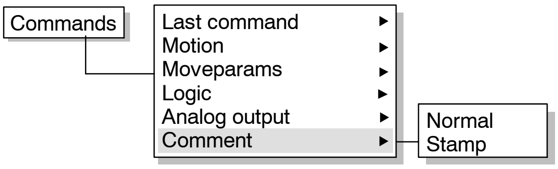

ANOUT ON CHANNEL 1 = 1 * VEL ACT Offset= 0 Delay= 0 [sec]

Finally, press the softkey “Cmd OK” or the Enter key. The command line is then inserted into the program.

2.6 Comment

For the purpose of giving your programs a clearer structure, comment lines should be inserted containing texts explaining the program modules.

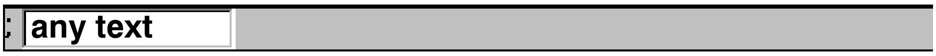
To do so, place the edit cursor one line above the position where you want to insert the comment line.



You can now select one of the comment types “Normal” or “Stamp”.

2.6.1 Normal

Using the command “Normal”, you can insert a comment line containing any text into your program. A semicolon (;) will automatically be inserted at the beginning of the comment line.



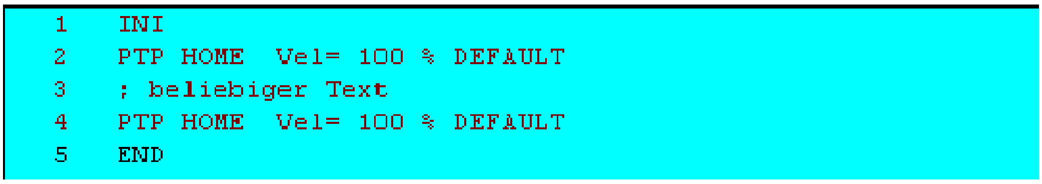
; any text

Press the softkey “Cmd. Abort” to discard your entries and close the inline form without inserting a comment line into the program listing.

You can switch to the inline form for the stamp directly by using the softkey “Stamp”.

If you would like to enter a different text, you can delete the old entry by means of the softkey “New text”.

Use the softkey “Cmd OK” to terminate your entry, close the inline form and insert the comment line.

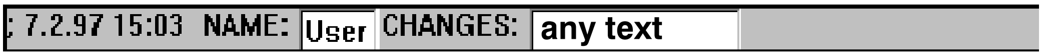


```
INI
PTP HOME  Vel= 100 % DEFAULT
; beliebiger Text
PTP HOME  Vel= 100 % DEFAULT
END
```

When the “Comment” function is selected again, the string entered previously is offered as a default text in the inline form.

2.6.2 Stamp

You can also insert a stamp in your program. In this instance, it is a comment line with the time and date added. A semicolon (;) is automatically inserted at the beginning of the line in this case as well.



; 7.2.97 15:03 NAME: User CHANGES: any text

Use this type of comment to document changes to programs.

You can enter your name in the first input box after NAME. Enter an abbreviation for the change that has been made in the second box after CHANGES.

Press the softkey “Cmd. Abort” to discard your entries and close the inline form without inserting a comment line into the program listing.

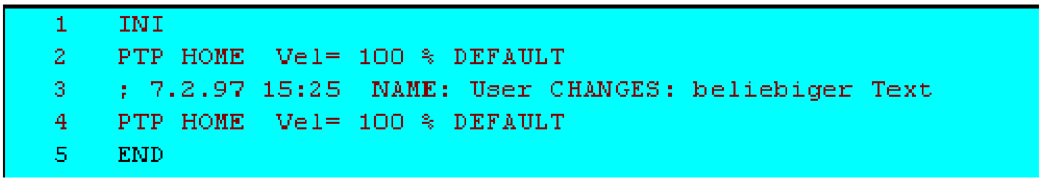
You can switch to the inline form for a normal comment directly by using the softkey “Normal”.

By pressing the softkey “New time”, the new system time and date are entered into the inline form.

If you would like to enter a different name, the previous entry can be deleted by pressing the softkey “New time”.

If you want to enter something else after CHANGES, press the softkey “New text”. This deletes the previous entry and a new text can now be entered.

Use the softkey “Cmd OK” to terminate your entry, close the inline form and insert the comment line.



```
INI
PTP HOME  Vel= 100 % DEFAULT
; 7.2.97 15:25  NAME: User CHANGES: beliebiger Text
PTP HOME  Vel= 100 % DEFAULT
END
```

When the “Comment” function is selected again, the string entered previously is offered as a default text in the inline form.

2.6.3 Deleting a comment

Use the arrow keys “↑” or “↓” to move the edit cursor to the line you want to delete.

[Program > Delete]