Engineer^IT
Control Builder F

Engineering Manual
Process Station - Rack System

Notice

Information provided in this manual is subject to change without prior notice and represents no obligation on the part of ABB Automation Products.

The industrial standards and regulations (e.g. DIN, VDE, VDI, etc.) applicable in the Federal Republic of Germany are used. Outside the Federal Republic of Germany, the relevant national specifications, standards and regulations must be observed.

The designations used and the products shown/mentioned in this manual have not been specifically marked regarding existing industrial property rights.

Should you find any mistakes in this manual, please make a copy of the appropriate page(s) and send it/them to us with your comments. Any suggestions which may help to improve comprehension or clarity will also be gratefully accepted.

Please send your suggestions to:

Product Management Dept., DEAPR/LMS-Hannover, Fax: +49 (0)511 6782 701

Engineering Manual
Process Station – Rack System

Gross Automation, 1725 South Johnson Road, New Berlin, WI 53146, www.ssacsales.com, 800-349-5827

Contents

Notes on this manual

To grant direct access to information, we have used different types of scripts and symbols.

Script	Meaning
Italics	Representation for (selectable) menu items or parameters.
SMALL CAPITALS	Inputs to be made via the keyboard, also via virtual keys.
boldface	Highlights important information, also as an orientation hint.

Symbols	Meaning
	Cross-reference to another Freelance manual.
	Selection with mouse. The various instruction steps are separated by arrows. **Example**: → *Edit* → *Insert Below* → **...** . In this example, the menu item *Edit* is to be selected, followed by the menu item *Insert Below*.
	Operating alternative with the mouse.
	Information on operation with the keyboard, inasmuch as it differs from the Windows Standard. Example: Select module → ALT → E → D → **...** . Having selected the module via the ARROW KEYS, the keys ALT, E and D must be pressed successively. If two keys are to be pressed simultaneously: **...** → SHIFT + INSERT → **...** .
	Alternative keyboard operation
	Hints
	Warnings and special hints, must be observed!

1 General

Within the hardware structure the resources defined in the project tree are allocated to the hardware actually required. A scalable Freelance 2000 system essentially consists of the process and operator stations. These stations are allocated to resources according to IEC 61131-3 and serve as a structuring level in the project tree for allocation of the application program parts and displays to the hardware actually required.

The D-PS resource stands for a **process station**. This accepts the I/O modules for process control, and is essentially determined by the CPU module. The CPU module processes all programs of this resource, assigns a station number to the process station, and is the communication module to the system and the I/O units. The process station consists of the central unit and up to four I/O units. The central unit is always the station with the CPU module. The I/O units do not need an own CPU module: they can accept other I/O modules so that their numbers can be expanded (from 8 for the central unit) to a maximum of 44 for a process station. All units need a link module for feeding in the power supply and an identical rack for holding the modules.

A single software tool suffices for making all necessary configuration and commissioning actions from a PC. Graphical user interfaces are available for diagnosing.

Efficient graphic editors allow you to configure your automation tasks in accordance with IEC 6-1131-3 and then download them into the process station in online mode.

In order to facilitate the visualization of your process, powerful visualization packages like DigiVis are available for your process stations.

1.1 System overview

The system view shows the graphic view of the hardware structure.
The system bus **(Ethernet)** links the individual stations with each other. It transmits data between the process stations, the operator stations and the engineering station via coaxial or fiber-optic cables.

The station bus **(CAN)** links the main rack with other racks or I/O units.

The **engineering station** is a PC or laptop with MS-Windows NT 4.0. It is used by the operator for system configuration, documentation and commissioning. After this has been done, it can be disconnected and used for other purposes.

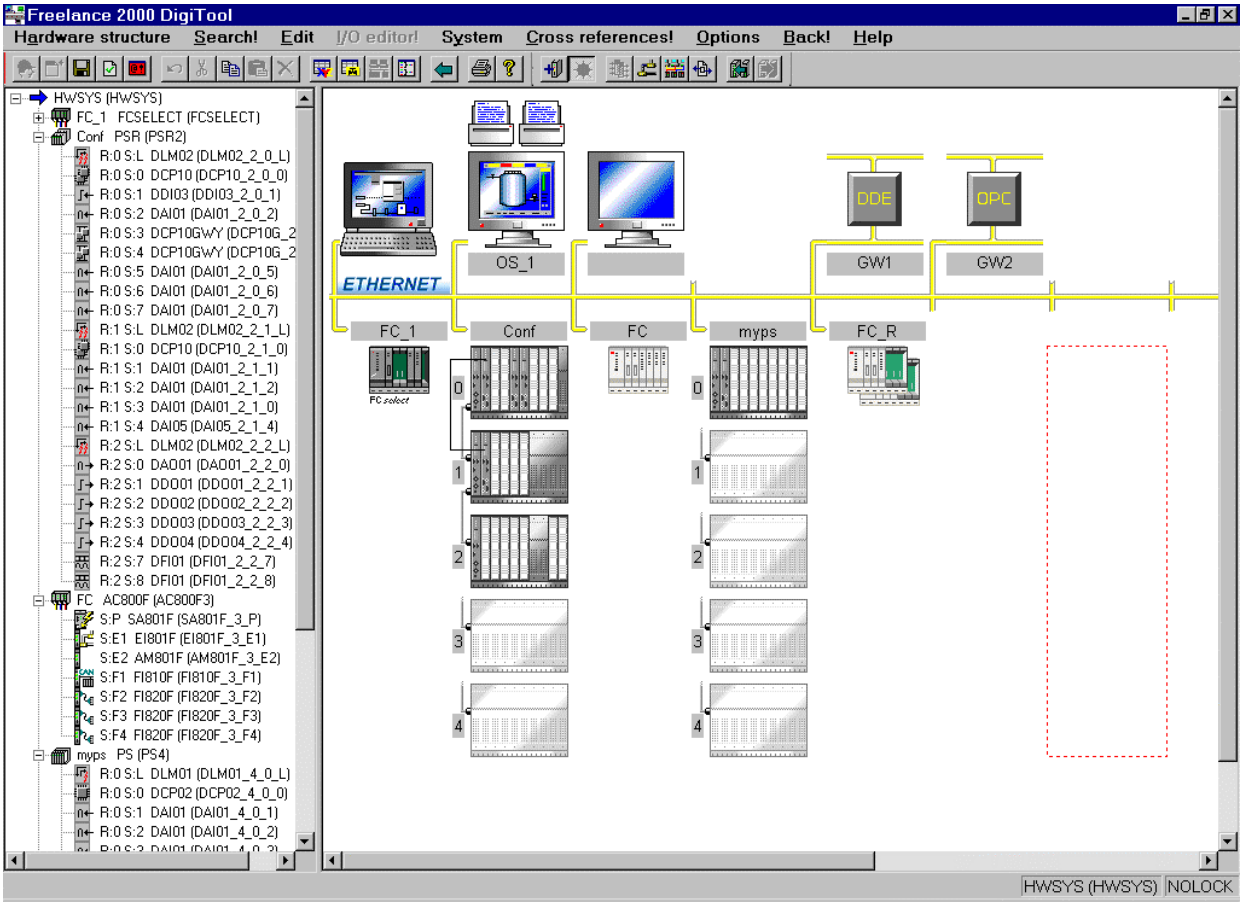
The **operator station** is based on a PC with MS-Windows NT 4.0. It is usable for configuration and commissioning as well, provided that the engineering software has been installed here.

A scalable Freelance system consists of operator stations and process stations/FieldControllers. The process stations consist of a central unit and up to four I/O units.

Up to two printers can be connected to each PC. The engineering station does not need to be activated separately. It is only used for displaying the network address.

Specific positions are assigned to the possible stations, either directly by double-clicking a position in the graphic view, or in a dialog when they are inserted in the tree structure.

→ *System* → *Hardware structure* → *System object (HWSYS)* → *Edit* → *Insert*



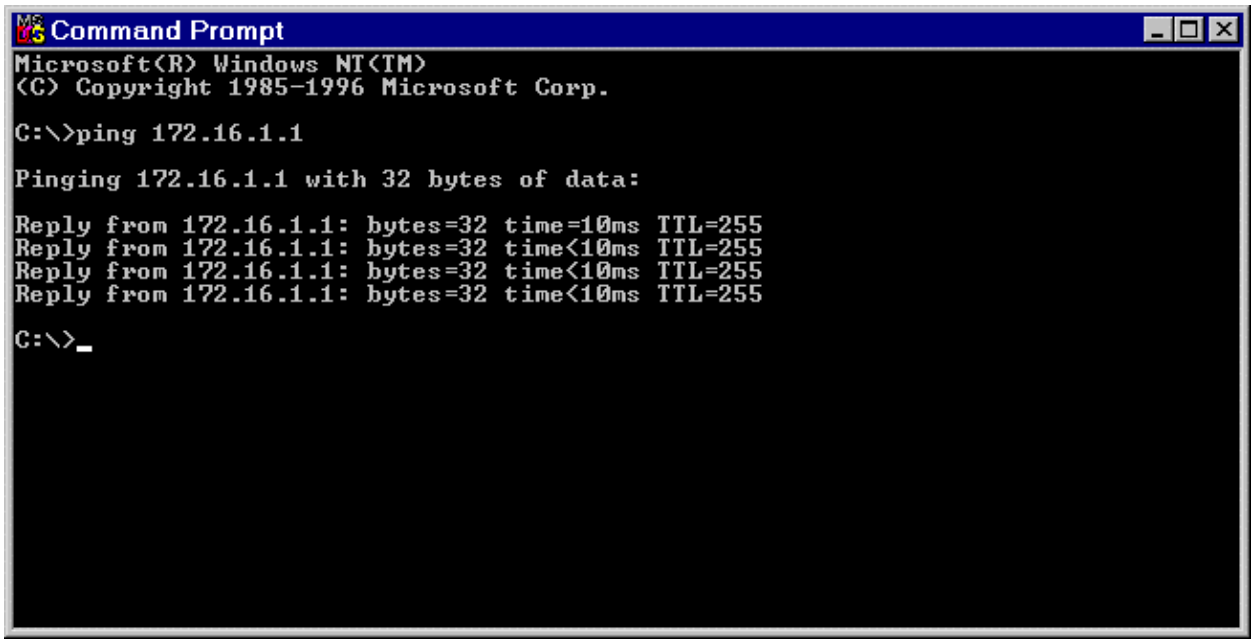
ra015us.bmp

2 Operating System and EPROM Download

2.1 Boot EPROM Download

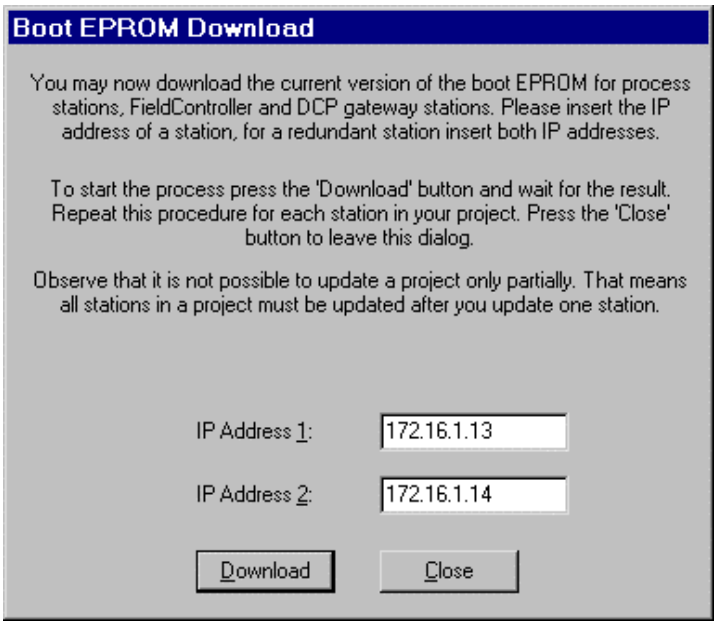
The Boot EPROM on the DCP02/PCP10 must have the same version number as the engineering software. The Boot EPROM can be updated either in the installation phase from DigiTool, or at a later time from DigiTool Configure. The prerequisite is that the TCP/IP protocol is installed on the engineering station, and that there is an Ethernet connection to the process station.

Prior to loading the Boot EPROM to a CPU you should check that the network connection between the engineering PC and the process station works properly. For this purpose, a "ping" is entered in the command prompt window under Windows NT and then transmitted to the process station. A positive response from the CPU (Reply from 172.16.1.5: bytes=32 time=10ms TTL=255) indicates that there is a network connection to the process station.



ra002us.bmp

→ *Start* → *Programs* → Command prompt c:\> ping 172.16.1.5



ra001us.bmp

DOWNLOAD The current version of the Boot EPROM is downloaded to the process station with the given IP address(es).

CLOSE Terminates the EPROM update and completes the installation of the DigiTool version or the download of the Boot EPROMs.

If you have more than one process station in your project, repeat the steps described above for each of them. If you are using a redundant process station, enter both addresses (IP addresses 1 an 2). The Boot EPROM will then be loaded for both of the CPUs.

Taskbar → *Start* → *Programs* → *Freelance 2000 Tools V6.1* → *DigiTool Configure*

2.2 Operating system download

The operating system is downloaded into the station via the system bus by actuating **BOOTSTRAP**.

Prerequisites: Make sure that no operating system resides yet in the process station (RUN-STOP LED off). Otherwise, you have to **initialize all** prior to downloading the system.

The operating system bootstrap is normally executed automatically when the project is loaded for the first time by using the **Load whole station** option. However, it can also be performed upon a separate initialization action, independently of loading the user program. Downloading the operating system into the secondary CPU is not necessary, this is done automatically by the primary CPU.

3 Configuring the D-PS or D-PS/RED Resource in the Project Tree

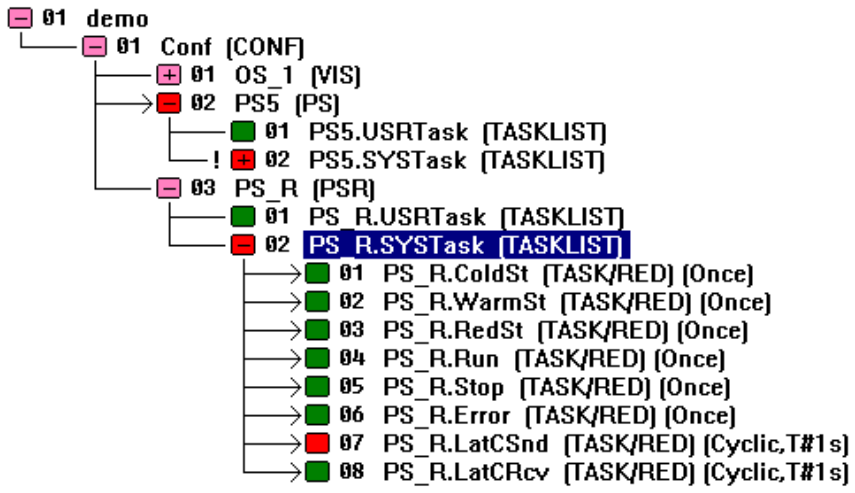
In the project tree the individual software components of a project are shown in a clear overview. The individual elements or objects, usually referred to as the **project objects**, are implemented according to the IEC 61131-3 standard. Data processing of process data actually takes place in the **D-PS resources** (**process stations**). Process operation and monitoring are realized in the **D-LS** resources (**operator stations**) or **D-GS resources** (**gateway stations**). Data from other systems are integrated by the **OPC server** resource.

The resources in the project tree have to be assigned to the hardware objects in the hardware manager, in order to define which program is to be executed by which process, operator or gateway stations.

The process station resource (**D-PS**) stands for a Freelance 2000 process station, on which the **resource** and the tasks and programs assigned to it are processed.

The short name **D-PS** or **D-PS/RED** in the project tree indicates that no allocation to a physical station has been made yet in the hardware manager. After the assignment has been made, the allocated station type PS (or PSR for a redundant process station) is indicated.

The illustration below shows a redundant and a non-redundant PS in the tree.



ra003us.bmp

3.1 Inserting a D-PS or D-PS/RED resource

Select the target position in the project tree

→ *Insert above*
Inserts a new process station **above** the selected object.

→ *Insert below*
Inserts a new process station **below** the selected object

→ *Insert next level*
Inserts a new process station one level lower

The target position can be on the next lower level under the configuration element, or on the configuration level itself (with *Insert next level* only).

→ Position cursor on target position, click with left mouse button → OK

3.2 Inserting a task or redundant task

In the process station, the actual programs are run in the tasks. The programs are processed either with program lists or with the sequential function charts.
A redundant task has so-called redundancy data. These are data of the process image and of the function blocks. The data sets are matched with each other after each cycle. This ensures that a redundancy toggle can take place at any point in time.
So that the data of a redundant task are redundancy-capable, all variables of a redundant task must be written **in process image mode**.
Not only the redundant task (TASK/RED), but also tasks of the previous type (TASK) can be configured under the redundant resources. Thus only those functions of the user program which also have actually to be redundant need to be configured redundantly within a redundant resource.

4 Processing and Error Handling

4.1 Automatic error handling at task level

Error handling at task level is described in the flow chart (**see page 15**).

If an error is detected in a user program, the error task is executed once. The error task has the highest priority (priority 100) in a resource and is used for handling errors in user programs with user programs. If an unrecoverable error is detected, the error-producing task changes over to the unrunnable state. It the error is recoverable, the error-producing task can remain in the running state, provided that automatic error correction has been activated for the D-PS resource. Due to its high priority it cannot be interrupted by other tasks. However, the execution of the error task can be deactivated.

Error handling at task level is switched on by default, but can be deactivated in the resource configuration (see the **Engineering Manual, System Configuration, Project Tree**). When deactivated an error causes the task to change to the unrunnable state, even if the error is recoverable. The error appears in text form in the task header, and the object number of the faulty project object is also displayed.

If automatic error handling is active and a recoverable error is detected, the affected task remains in the running state. In this case, the error condition is ascertainable only be evaluation of the system variables.

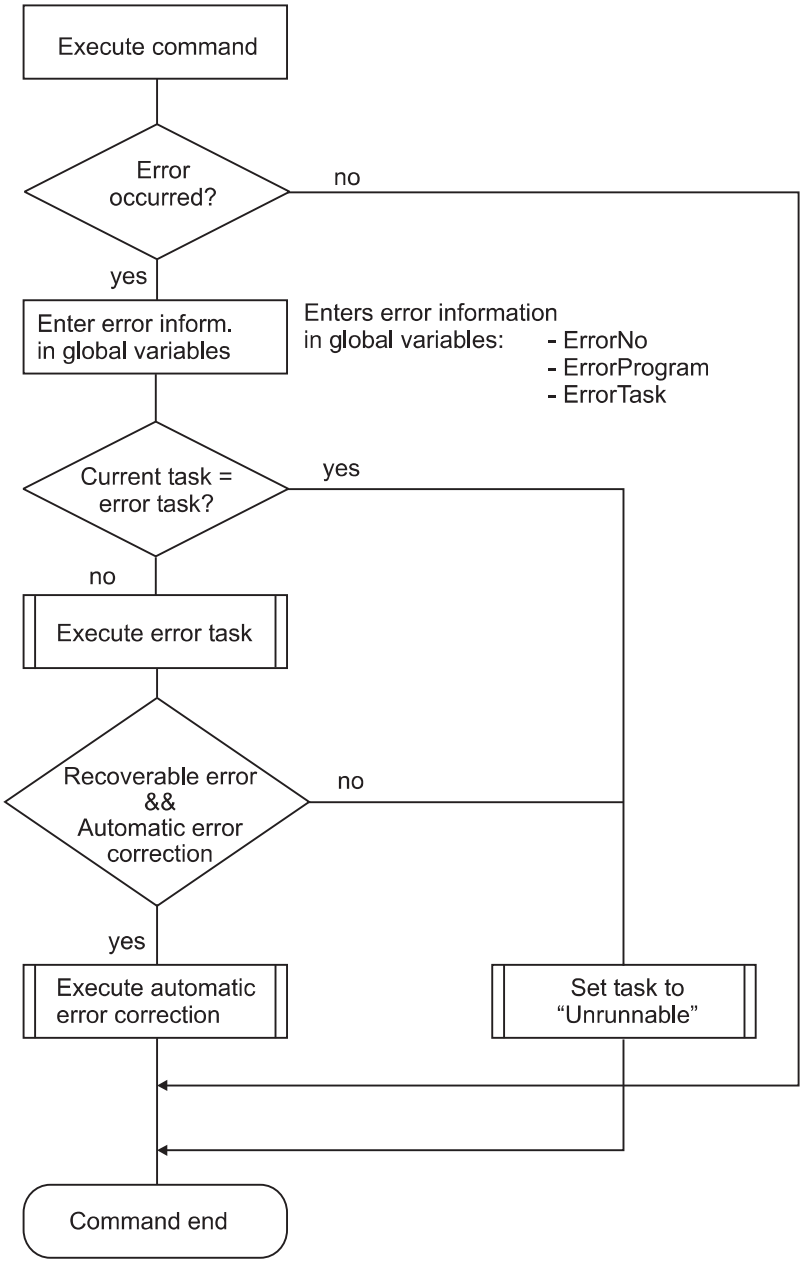
Each runtime error detected during execution of a command results in an entry recording the cause and location of the error in system variables specially defined for error handling. The cause of the error, e.g. 4 for UNIT Div by 0, is saved in the variable "ErrorNo". The variable "ErrorProgra" is written with the object number of the program or function block that caused the error. Furthermore, the object number of the affected task is saved in the variable "ErrorTask", but only displayed in the task header if error handling is deactivated. The affected object can then be localized in the object list with its name, state, type and position in the project tree. If the affected task, the one which triggered the error, is not the error-handling task, the error handling task itself is now triggered. Error handling specific to the cause of the error and/or its location can be undertaken in this task, with the aid of the system variables.
If an error should occur in the error-handling task itself, the error-handling task is signaled as **unrunnable**.

See also **Engineering Manual, IEC 61131-3 Programming, Variables, System Variables**.

4.2 Task error messages

Error #	Error text	Description
1	INT/DINT overflow	Signed integer or time operation caused overflow.
2	INT/DINT underflow	Signed integer or time operation caused underflow.
3	INT/DINT div by 0	Signed integer division by zero.
4	UINT/DINT div by 0	Unsigned integer division by zero
5	INT/DINT save overflow	Signed 16-bit-integer operation caused overflow on storage.
6	REAL overflow	Real operation caused overflow.
7	REAL underflow	Real operation caused underflow.
8	REAL div by 0	Real division by zero.
9	REAL invalid value	Real operation with undefined value
10	DT overflow	Operation with DT and TIME caused overflow.
11	DT underflow	Operation with DT and TIME or with DT and DT caused underflow
12	UINT/UDINT overflow	Unsigned integer operation caused overflow.
13	UINT/UDINT underflow	Unsigned integer operation caused underflow.
128	Program exec. error	Unloaded object (e.g. program, FB, etc.) to be executed.
129	FB error integer	A function block from the function block library has triggered an integer overflow or underflow, the error can only be detected following execution of the block and, thus, no recovery is possible.
130	FB error real	Function block from function block library triggered REAL overflow or underflow, error can only be detected after execution of the block and, thus, no recovery is possible.
131	Process image read error	Unrecoverable error occurred on reading process image.
132	Process image write error	Unrecoverable error occurred on writing process image

4.3 Task error characteristics



Error messages

Recoverable errors		Unrecoverable errors
1 INT_OVERFLOW	8 FLOAT_DIV_BY_ZERO	128 PROGRAM_EXECUTION_ERROR
2 INT-UNDERFLOW	9 FLOAT_NAN_VALUE	129 INT_FB_ERROR
3 INT_DIV_BY_ZERO	10 DT_OVERFLOW	130 FLOAT_FB_ERROR
4 UINT_DIV-BY_ZERO	11 DT_UNDERFLOW	131 PROZABB_READ_ERROR
5 INT_STORE_OVERFLOW	12 UINT_OVERLOW	132 PROZABB_WRITE_ERROR
6 FLOAT_OVERFLOW	13 UINT_UNDERFLOW	
7 FLOAT_UNDERFLOW		

di1503us.eps

4.4 Startup and shutdown performance of the modules

A signal for connection monitoring is permanently exchanged between I/O modules and the CPU module of the process station. If this signal does not appear for more than 250 ms (for 500 kbit/s) and/or 1,25 s (for 100 kbit/s) both sides detect the interruption of the connection. The CPU module sends a system message to the operator station and the output modules adopt the safety values. Dependent on the requests of the process the safety values can configured as "Hold the last value" or to a particular value/state.

A cold process station start always leads to output of the safety values.

For "Load whole station" and at resource Initialization/Overall reset, the output levels of the output modules go to zero current.

4.5 Power fail signal (PF) and no power fail

PF < 15 ms	While PF signal is active, the CPU module of the process station does not communicate actual values; I/O modules maintain the last values.
PF > 15 ms	I/O modules maintain the last values until they have recognized the interrupted connection with the CPU module; they then adopt the configured safety values. Safety values are maintained until the CPU module has executed a warm start and sends new values to the I/O modules.

4.6 Power fail signal (PF) and power fail

I/O modules maintain the last values until they have recognized the connection interruption and then adopt safety values until the CPU module sends actual values after restart. In the case of power fail of the I/O module, output levels adopt zero voltage and/or zero current.

After voltage return I/O modules only change their output level when the CPU module communicates actual values. Analog output modules are an exception. They hold the last values as long as external voltage supply does not fail. Outputs are only deenergized when the CPU module is powered again.

5 Configuring the Process Station in the Hardware Structure

Within the hardware structure the resources defined in the project tree are allocated to the hardware actually required. The D-PS resource stands for a process station in the hardware structure.

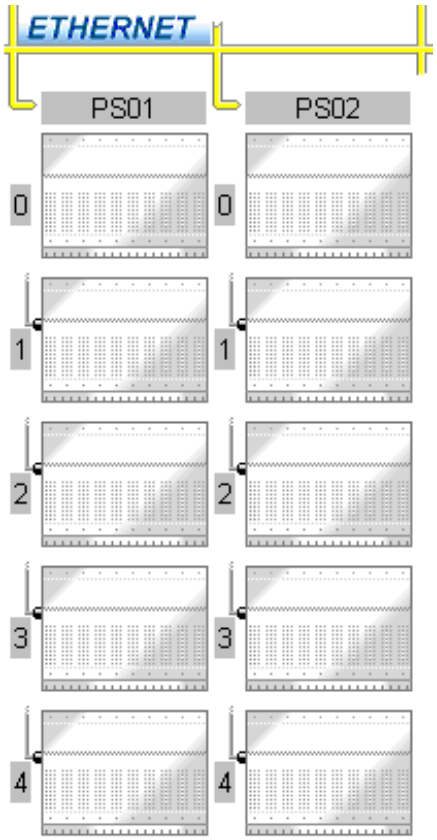
The rack-based process station consists of a basic unit (main rack) and the extension units (extension racks). Each rack accommodates various modules. A CPU module is required for every process station, and a link module DLM is needed for every rack. The modules must be plugged in special slots.
The I/O modules can be arranged in all racks as required. They can be plugged in any slots, except for those reserved for the DCP and DLM modules.

The allocation to the physical stations is made in the hardware manager. You can select here the process stations (PS) for rack-based systems and FieldControllers for connection of fieldbus modules. A distinction is made between redundant and non-redundant process stations. Already when inserting a resource you can thus define whether or not the tasks and programs assigned to it are to be processed in redundant or non-redundant mode.

A redundant process station of type D-PS/RED consists of two DCP10s linked via a redundancy link. In the project tree, they appear as one process station resource. When loading the process station, one of the two process stations becomes a primary and one the secondary station. Note that only the primary stations is shown in the project tree.

5.1 Inserting process stations

Process stations can be inserted in the system in either the tree view or system view. After *Insert* the object will appear in both the tree view and the system view. The object is shown in the respective station position, in graphical form. Double-click on the gray field in the header of the inserted process station to make the assignment of the resource.



ra005us.bmp

→ Select system object (HWSYS) in the tree view → *Edit* → *Insert*
→ Select station type → Select position

→ Select station position in system view → Double-click

5.2 Allocating a resource

When a station is allocated, it is automatically activated or a search operation is carried out in the DigiTool "Commissioning" mode to find the station on the system bus.

In this way, the parts of the user program configured in the project tree are allocated to the stations.

→ Left mouse click on the gray text field of the station
→ *Edit* → *Resource allocation*
→ Select the corresponding resource from the list

→ Double-click on the gray text field of the station
→ Select the corresponding resource from the list

5.3 Inserting modules

Modules can be inserted in the process station in either the tree view or the station view. After *Insert*, the object appears in both the tree view and the station view. The object is shown in the corresponding slot. In the station view, the modules can be equipped by double-clicking.


→ Select process station in the tree view → *Edit* → *Insert*
→ Select module type → Select a free slot


→ Select slot (L, 0 - 8) in the station view → Double click

The following modules are available:

DAI01	Analog input module 16 x 0/4...20mA, Ri=50 ohms
DAI02	Analog input module 16 x 0...10V DC
DAI03	Analog input module 16 x 0/4...20mA, Ri=250 ohms
DAI04	Analog input module 8 x Pt100 / mV
DAI05	Analog input module 16 x 0/4...20mA, MU supply
DAO01	Analog output module 16 x 0/4...20mA
DCO01	Communication module 4 x RS 485/RS 422 or RS 232
DCP02	CPU module
DCP02GWY	DCP gateway DCP for the Maestro UX gateway
DCP10	CPU module (capable of redundant operation)
DCP10GWY	DCP gateway DCP for the Maestro UX gateway
DDI01	Digital input module 32 x 24V DC
DDI02	Digital input module 16 x 24...60V AC/DC
DDI03	Digital input module 16 x 90...230V AC
DDI04	Digital input module 28 x 24 V DC NAMUR initiators
DDI05	Digital input module 32 x115/230 V AC
DDO01	Digital output module 32 x 24 V DC, 0.5 A
DDO02	Digital output module 16 x 24...230 V AC/DC
DDO03	Digital output module 16 x 24...60 V AC/DC, read back
DDO04	Digital output module 16 x 115...230 V AC, read back
DFI01	Frequency input module 4 x f <=45kHz
DLM01	Link module for power supply
DLM02	Link module with redundant power supply