Albatros

3.1.9

Operator's manual

Tecnologie e Prodotti per l'Automazione

Table of Contents

1 Introduction

1.1 How to use this manual

This user's guide describes the functions of *Albatros s'* numeric control.
Thanks to the structure of the manual, getting to know the system and learning how to use it will be an easy task for the operator.

The main subjects of each section of the guide are:

- *Albatros s*' windows and tools.
- description of the typical architecture of a system, such as *Albatros* .
- how to display the devices and operate on them with manual and diagnostic functionality, using the Synoptic window.
- how to display Technological Parameters, Geometrical Parameters and Tools Parameter and how to modify them if necessary.
- how to display the Devices and operate on them with manual and diagnostic functionality.

To avoid overcharging this guide, for further information concerning the use of the *mouse*, *menus* and *toolbar* and all the current operative functions of Windows, we refer the reader to Windows Operative System manuals.

1.2 Work windows

There are various types of work windows, depending on the kind of operation required, and more than one can be kept open at the same time.
The types of windows are the following:

Window	Description
Main	Albatros's main window. It allows you to call functions and contains all the other windows whose content depends on the specific application they represent.
Synoptic	it contains a graphic representation of the machine, or of parts of the machine, and allows you to operate on them.
Technological Parameter	it enables to display and modify the technological parameters and the machine parameters.
Tools Parameter	it enables to display and modify the tools parameters.
Diagnostic	it enables to display the state of the devices and, if possible, to operate on them.
System Errors	window containing the list of the most recent system errors. It is also possible to display cycle errors and messages.

2 System composition

2.1 Access rights to the system

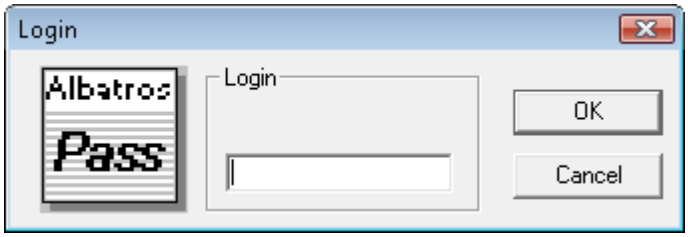
Albatros has four access levels to the system:

- ***User**:* is the level with most access restrictions. It does not allow you to modify any of the devices settings. It is the level used for machining and normal machine operations. When the system is booted this access level is automatically enabled .
- ***Service**:* is the level used for ordinary maintenance of the machine. The operator should be able to modify certain configuration parameters, without altering the structure of the machine.
- ***Manufacturer**:* is the level used to configure plants and machines. At this level almost any kind of modification is possible. It is used by developers.
- ***Tpa**:* is the highest access level of the system. Its function is to protect access to particularly delicate settings whose modification requires a detailed knowledge of Albatros. This level is very rarely used and the access password must be requested directly at T.P.A. S.p.a.

To access the system at a higher level than User, or to return to User level after introducing changes at a higher level, the corresponding password must be introduced.

To recall the login window, press **[CTRL+ *]**(asterisk). Alternatively, click on the icon at the right end of the Windows' "**applications bar**" with the right mouse button to view a menu in which the **Change pass level** command appears.

The window you are opening looks like this:



Login window

Now introduce the password and press the **[OK]** key to confirm. The letters composing the word will be visualised as "*" characters, to avoid anyone reading the password typed in.
By typing in the password you have logged into the corresponding access level. To have a confirmation of the level accessed, select the heading ***About* Albatros** from the ***Help*** menu.

If the introduced password is not correct the error message "Warning! Wrong Password!" will appear.

2.2 Multilingual support

Albatros supports the display of text in multiple languages.
Albatros does not currently support some languages such as the oriental languages, which use a special set of characters or don't use the left->right direction typical of the western languages.

Change language
The language may be changed at any access level of the system. To select a different language press **[CTRL + /]** or select the icon from Windows "**applications bar**". In the now opening window select the language required and click on **[OK] .**
Change of language is not made immediately, but at the following restart of Albatros.

2.3 Typical architecture of the system

Because many aspects of graphic representation and the structure of basic data of the Machine depend greatly on the kind of Machine, this Manual provides by way of example a description of the composition of a typical system, as well as some general information.

The detailed information, the diagrams and the graphics of the real system obviously depend on the specific application, and are consequently prepared by the Manufacturer of the Machine Tool.

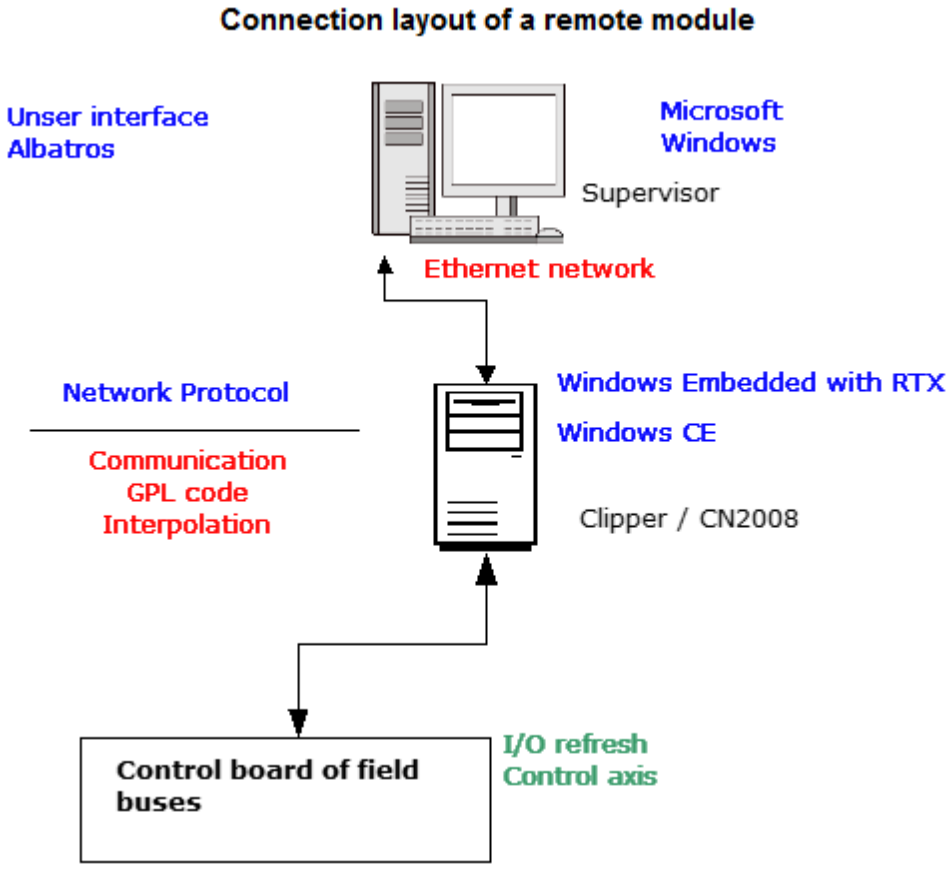
The Albatros numeric control system is composed by a supervisor PC, showing the Operator-Machine interface, and a number of modules (ranging between 1 and 16) for the piloting and control of all operative resources of the Machine Tool or Plant.

So, you can have two kind or plants:

Monomodule consisting of one module connected directly to the PC bus.

Multimodule consisting of a minimum of 1 and a maximum of 16 modules, usually used for applications on Plants or Lines with a number of machines; the PC unit in this case is physically separate from the modules, which can be located in various points of the Line or Plant.

In both architectures the modules are composed by one or more axis cards for the direct control of the Machine Axes and the logic management of the Input/Output system.
In the monomodule version, the axis cards are installed directly in the Supervisor PC, while in the multimodule version they are installed in an industrial PC (with or without screen and keyboard) connected to the Supervisor PC via Ethernet. The following figure shows the diagram of the connection between the Supervisor PC and the remote module (Clipper). The main activities of the single components are also described.



Connection diagram of a remote module

Intelligent remote devices pilot I/O devices and axes (TRS-AX remote) directly on the machine. These devices read the Digital Input (ON/ OFF) or Analog Input channels, refresh the Digital or Analog Output channels and are connected to the Modules by means of a Greenbus (serial bus RS485 - 1 Mbaud) and CAN bus and PowerLink II and Ethercat optionally. The profile machining of Albatros is protected by a USB hardware key, configured by T.P.A. S.p.A.

2.4 Organisation and logic configuration

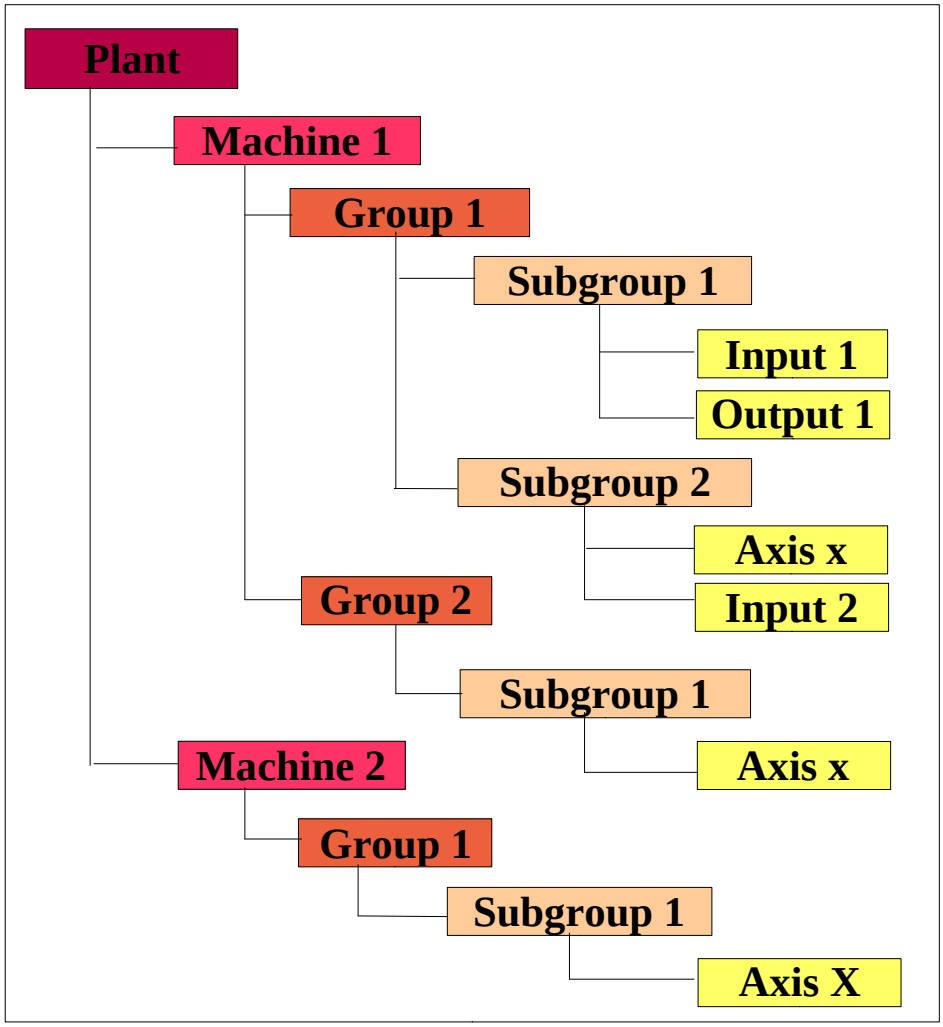
In the Albatros system the descriptive structure of the plant or single machine tool is organised in a technological file with a hierarchical structure.

This approach allows you, if necessary, to maintain the modular structure of the machine as far as the configuration data and access modalities are concerned, by classifying it in terms of dynamic association of various modules, aggregates and devices that may be enabled or disabled according to the required setting.

Following this logical structure, in the most general and complex case the higher hierarchical level will be composed by:

1. **Plant** simply a set of machines. It represents the operational parts managed by the Numeric Control. The plant is always present, even in the case of a single machine and it is not necessary to mention it explicitly.
2. **Machine** from a "logic" point of view it is defined as a set of devices (axes, timer etc.) and control cyclics, corresponding to a GPL language code that applies the control algorithms of the machine itself. Generally the machine is provided with a large number of devices which are organised into groups.
3. **Groups** are "containers" which allows you to organize the components of the machine following a logical criteria. For example we could define an "axes" group containing all the axes of the machine, the limit switches, the cyclic performing the axes homing etc.
4. **Subgroups** indicate a further subdivision of a group. For example, the "axes group" could be divided into "digital axes" and "step-by-step axes".
5. **Devices** are the lowest level of the hierarchy. They are a logic representation of the electrical and mechanical components of the machine and are independent of the hardware below.

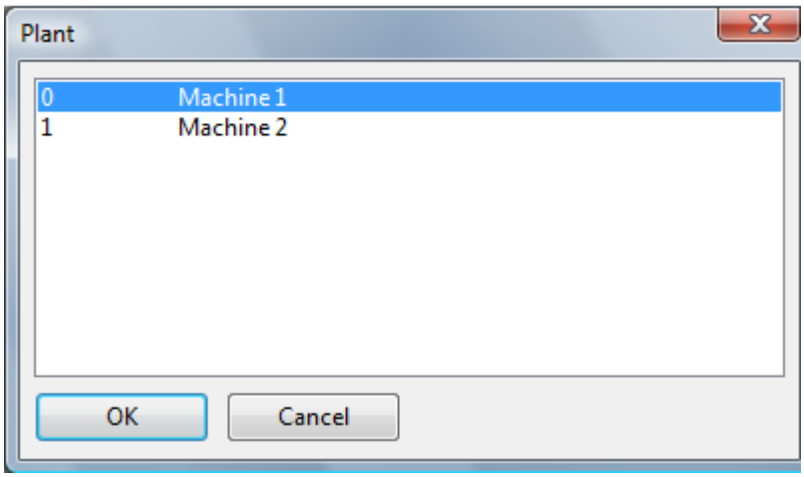
The following figure schematises the structure of a hypothetical plant composed by two machines:



Example of hierarchical structure of a plant.

NOTE: The Groups do not necessarily have to be divided into Subgroups, they can be made up directly of Devices.

In the case of a plant with more than one machine, to access determined functions, such as Diagnostic, System configuration and Technological Parameter, it is necessary to select the machine whose data we are interested in, through the following dialog window



Window for machine selection

2.5 Devices

The devices can be grouped into two categories: physical devices and logic devices. In the system all the devices are identified by a name describing their function.

Physical devices

By physical devices, we intend all those parts which act on the electrical or pneumatic parts of the machine or verify their state. These are:

Symbol	*Device*	*Function*
	*Digital input*	it verifies the state, "on" or "off", of a device. For example, the safety switch of a door.
	*Digital output*	it enables or disables a device, setting it on "on" or "off". It is used, for example, to pilot solenoid valves.
	*Analog input*	it measures the voltage of input power in the corresponding terminal. For example the power generated by a tachimetric dynamo.
	*Analog output*	it defines the output voltage of the corresponding terminal. It can be used, for example, to pilot an inverter.
	*Input port*	it is composed by 8 digital input channels.
	*Outport port*	it is composed by 8 digital output channels.
	*Input Nibble*	it is composed by 4 digital input channels.
	*Output Nibble*	it is composed by 4 digital output channels.
	*Axis*	it controls the movement of electrical axes. It is possible to control various kinds of axes: analog controlled, digitally controlled, step-by-step motors, counting axes (only encoder reading) and frequency/direction controlled axes.

Logic devices

Logic devices are parts which act exclusively within the operating programs and do not have a physical counterpart:

Symbol	*Device*	*Function*
	*Timer*	time counting device. The measurement unit is the second. Resolution:4 ms. It can only indicate positive numbers, displaying a maximum time span of 8.589.934 seconds (with Real Time at 250 Hz). The amount is recorded in the non-volatile memory of the axis card.
	*Counter*	operation counting device. It may display any number between -2.147.483.648 and +2.147.483.647. The amount is recorded in the non-volatile memory of the axes card.
	*Flag bit*	off/on indicator.
	*Flag switch*	special flags that can be connected to certain buttons on the tool bar, as the Start flag, for example.
	*Flag Port*	it is composed of 8 flag bit channels.
	Function	the basic unit of the GPL code which, together with other functions, forms a program. It is the logic control device for other devices
	*Variable*	GPL code *integer* type global variable.
	*Variable*	GPL code *char* type global variable.
	*Variable*	GPL code *float* type global variable.
	*Variable*	GPL code *double* type global variable.
	Variable	GPL code *string* type global variable.
	*Variable*	GPL code *array* type global variable.
	*Variable*	GPL code *matrix* type global variable

3 Synoptic View

3.1 Using the Synoptic View

During machine functioning it is possible to open the *Synoptic View* window to verify the state of the most important devices.

Synoptic views display the same information contained in the diagnostic window. However, while in the latter the information is displayed in a tree structure (which includes all the devices present on the machine), synoptics illustrate the information graphically (displaying, for example, an image of the machine and setting the position of the axes next to the axes themselves). Synoptics also allow you to select the most significant information, grouping the remaining information in secondary screen pages, to be recalled by the user when necessary.

3.2 How to operate on the Synoptic View

The Operator can select the various pages composing the synoptic, for diagnostic purposes, by double clicking with the mouse on one of the areas of the machine. The different areas are delimited by a hatched rectangle, and are also known as "hot spots".

To select a "hot spot", a device or an axis, simply move the mouse pointer into the image of the required object. The name of the selected device appears immediately in the Status Bar.

The appearance of the mouse pointer changes according to the selected object, to indicate what kind of operation is allowed on that specific object.
The pointer will appear as a:

 *magnifying glass*	if it's a "hot spot"
 *hand*	if it's an output device
 *text cursor*	if it's a set-value box

3.3 How to act on Devices

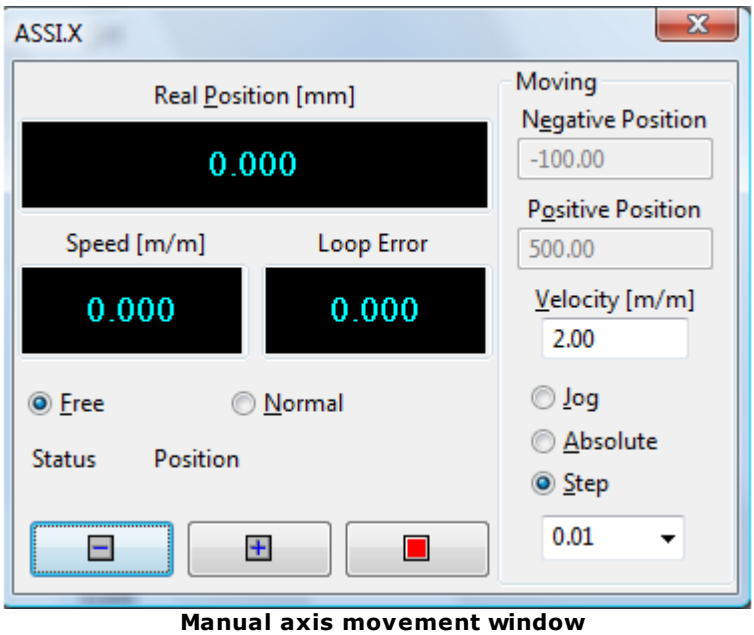
To act on a device, point the mouse on the required device, and complete action as described below (actions vary according to the type of device).

Representation mode	**Action**	**Device**
Device icon	point and *click*	Digital output Flag switch Flag bit
Set-Value box	point, *click*and set value	Analog output Output port Flag port Axis position Timer Counter
Select-value box	point and *click*	Output Nibble

3.4 Manual Axes Movement

To access the manual axis movement function, it is necessary to have the required access rights. Access rights are assigned by the manufacturer of the machine.

To interact with an axis, simply *double click* with the mouse on the positions display field of the required axis. The dialog window in the figure below will open. In the case of Virtual, Stepping motor and Count axes, the dialog window contains less data. For example, in the case of a Count axis, only the Real Position and Speed values are displayed.



Manual axis movement window

The window is composed of two areas, whose contents are described below.

Visualization area

- Three cells displaying axis *Real Position [mm]*, its *Speed [m/min]* and the *Loop Error* or tracking error.
- Two select buttons which indicate axis *Status* (*Free* = open loop, for ex. because of a system error, *Normal* = closed loop, corresponding to normal position control status). It is also possible to set the status by using these buttons.
- During movement, the signaling of axis *Status* (ex. Acceleration).
- Two buttons to select negative ⊟ or positive ⊞ direction axis movement.
- The ■ button**,** to Stop axis movement at any moment, during movement in Absolute or Step mode.

Movement area

- Two cells to set a *Negative Position* and a *Positive Position*, which will be used in *Absolute* mode.
- One cell to set the *Speed* imposed on the axis during manual movements.
- Three select buttons to choose what kind of movement to apply: *Jog*, *Absolute* position or *Step*.
- One cell to set the *Step* value to be used in *Step* mode.

To move an axis, set the parameters described above as required. Select movement mode and press the ⊞ button (to move the axis in positive direction) or the ⊟ button (to move the axis in negative direction).
In *Jog* mode the axis will keep on moving as long as the ⊞ or ⊟ minus button is kept pressed.
In *Step* mode, the axis will move as far as indicated in the "Step" cell each time the ⊞ or ⊟ button is pressed.
In *Absolute* mode, the axis reaches directly the position indicated in the Positive position or Negative position cell.

It is also possible to use the keyboard "+" (or Ctrl+P), "-" (or Ctrl+M) and "Space bar", instead of the ⊞ , ⊟ and ■ buttons.

4 Technological and Tools parameters

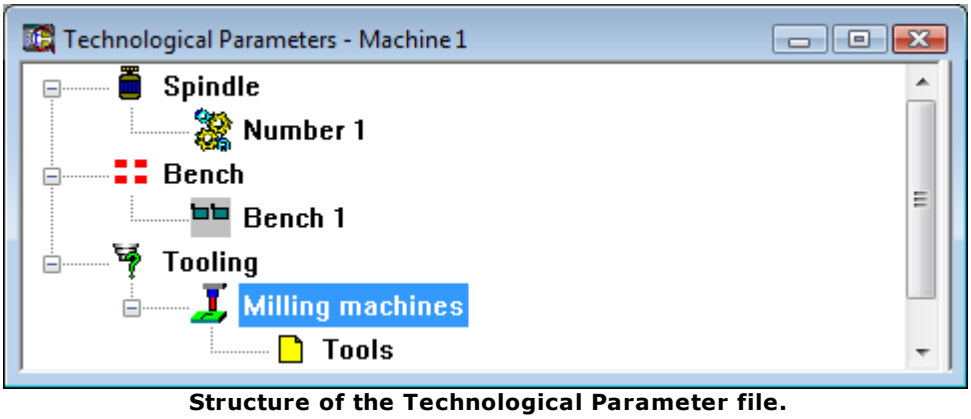
4.1 Technological Parameter window

The Technological Parameter file allows you to record the geometrical and technical information of a machine. The numeric control needs this information to handle machine functioning correctly.
To open: menu ***File->Open Technolgical Parameters***.

Technological Parameters are usually organized in Groups / Subgroups (Normally the groups and subgroups of the Technological Parameters are independent of the groups and subgroups in which the machine devices are divided). The display modes are defined by the Machine Manufacturer and depend on the specific application.

The values listed in the file are usually set by the manufacturer in the Machine calibration phase and can not be modified by the User, if not exceptionally. Therefore, some data may be protected by Password to avoid accidental modifications which could affect correct machine functioning.

The Technological Parameter window displays in a tree structure all the Groups and Subgroups of parameters that compose the file, as shown in the following figure.



Structure of the Technological Parameter file.

The window contains Groups, displayed in a tree structure, with their relative Subgroups of parameters. The tree structure can be expanded or collapsed using the ⊞ and ⊟ buttons found at each node. The +, -, Right/Left arrows and **Enter** keys and the **Space Bar** can also be used to open and close parts of the tree.

How to operate on Technological Parameters
Once the required Group / SubGroup tree is opened, it is possible to access the page containing the data. The data can be listed in a table, or in text or selection cells, depending on the type of data and how the Manufacturer set the data.
If any data is modified, it is necessary to press **[OK]** to make the changes permanent.

How to associate datatable rows to axes defined in the machine
To do that, you need to configure in the DataTable in PaStudio as follows:
- the suffix of the matrix name must be ":axis" (for ex: "MxConf:axis"), regardless of it is a group either a global or a library matrix;
- in the first Column you have to indicate that the character number must be greater of 20. In this column write the name of the axis, like in GPL.
In Albatros you have to enter in the first Column the names of the axes, whose parameters you wish to set. When you perform the "Cnc->Initialise" command, Albatros replaces the axis name with its logic address. From GPL through the instruction DEVICEID you retrieve the logic address of an axis, through which you can make a search into the parameter matrix.

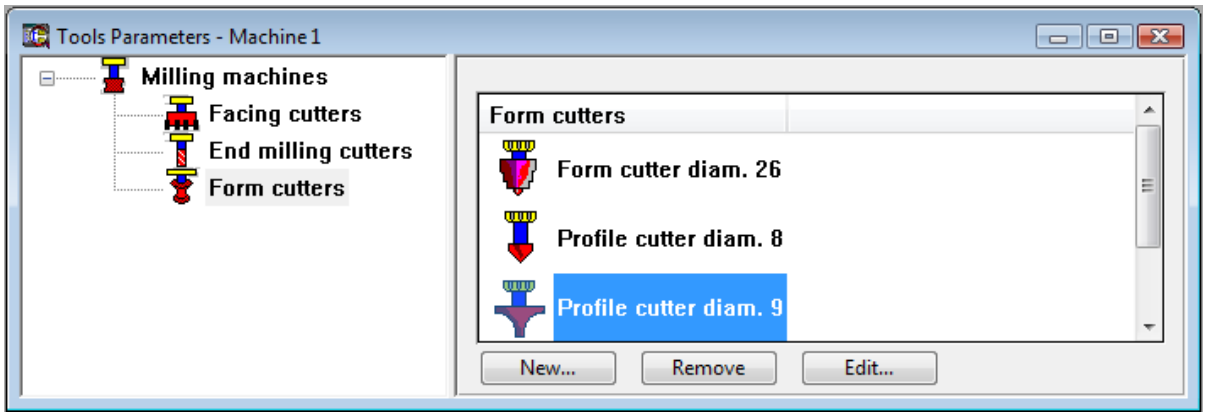
Tooling
Tooling is an unusual kind of machine data. Typically, any information concerning the set of tools the machine is equipped with (tooling) is saved in the Technological Parameter file. However, any information concerning the tools themselves is saved in the Tools Parameter file. For this reason, to define the tooling of a machine it is necessary to combine the information contained in the two files. If the system provides for this situation it will be possible to recall information contained in the Tools Parameter file from the Technological Parameters file. Usually the connection is implemented by means

of a button with a similar icon to the one below.

Selecting the icon and double clicking on it with the mouse's left hand button will open a window containing the list of tools defined in the Tools Parameter file, allowing you to select the required tool. When this has been done, the icon button changes, displaying the icon that represents the specified tool. It is also possible to display the tool data by double clicking on the icon with the right hand button of the mouse.

4.2 Tools Parameter Window

The window of Tools Parameters can be opened in Menu ***File->Open Tools Parameters***.
Tools parameters, determined by the Manufacturer on the basis of the operations performed by the machine, are usually organised as shown in the figure below:



Example of a Tools Parameter Window.

The Tools Parameter window is divided in two areas:

- the *left-hand area* contains the Groups, with the corresponding Subgroups of tools, displayed in a tree structure. The tree structure can be expanded or collapsed using the ⊞ and ⊟ buttons placed at each node. For example we could have a Milling Cutters Group composed by Subgroups of cutters with different characteristics, such as profile milling cutters, traverse milling cutters etc. Each one of these subgroups is associated to one or more tools whose characteristics are assigned in a dialog window defined by the manufacturer. The tools contained in each subgroup are displayed in the right-hand side of the screen.
- the *right-hand area*, that takes the name of the selected Subgroup, contains the list of Tools belonging to the Subgroup. The tools defined in this area do not necessarily exist on the machine. The association between the tool and the position on the machine (tooling) is normally done in the technological parameter file.

How to operate on Tools Parameters
Tools are added, modified and deleted from the file by means of buttons located in the lower section of the window:

[New]	enables to add a new tool to the Subgroup. It opens the "New Tool" dialog window, in which the following data can be inserted: - *Description:* a message that identifies the tool. The description can be chosen from the ones already in the list, if it has not already been assigned to another tool, or a new description can be written. - *Image:* an icon that identifies the tool. It can be chosen from the ones already in the list, or it can be called from a folder using the **[Image]** button. The tool is inserted in the list following the alphabetical order of the descriptions.
[Remove]	allows you to remove a tool from the Subgroup, although it is subject to confirmation; the description of the tool is not deleted and remains available for another tool.
[Edit]	allows you to replace the *description* or *image* of the selected tool, through the same dialog window described in the **[New]** command.

5 Diagnostic

5.1 The Diagnostic window

The Diagnostic window can be opened during machine execution to allow the operator to keep machine functioning under control, by monitoring the logic state of the I/O digital signals, analog I/O data, counters and timers data and axes movement.
Depending on the access rights conceded by the manufacturer, it may also be possible to modify the state of the devices.

If permitted by the access level, it is possible in real time:

- to display the state (ON/OFF) of all the digital Input and Output signals.
- to able and disable the digital Output signals.
- to display the voltage (ranging between +/-10V) of the Analog inputs.
- to assign a voltage (ranging between +/-10V) to all the Analog outputs.
- to move an axis in Manual by selecting the speed, the Pitch or the final absolute Position, and display real position, speed and loop error.
- to display and modify the global variables.

In the next paragraphs the devices and global variables will be described in detail, together with their graphic representation.

NOTE: In the diagnostic window only the devices enbled for the current access level are displayed.

5.2 Diagnostic window composition

It is possible to access the devices through the "Groups / Subgroups" structure, already described in the chapter System composition, which are then displayed in a tree structure.
At the head of the structure we find the machine, indicated by the icon:

 , followed by its Name and a Comment.

The structure can be expanded or collapsed by clicking on the ⊞ or ⊟ buttons. The tree can also be opened and closed by clicking on: **+**, **-**, left/right arrow key.

When a Group is opened, the following items are displayed in the tree:

- the "Devices List" of the Group, indicated by the icon 
- the Subgroups composing the Group, if any.

When a Subgroup is opened, the devices composing the subgroup are also displayed.

5.3 Representation of the Devices

The following information is shown with all the devices displayed.

- a graphic symbol;
- its State or current value;
- its Name;
- a Comment.

The list below contains the graphic representations of the devices, the type of device and the value displayed in real-time.

The state of digital inputs, digital outputs and flags is represented graphically by a LED which changes colour depending on whether the input is enabled or disabled.

In the case of Ports and Nibbles, that is, a number of lines (8 or 4 respectively) represented at the same time, a row of LEDs will be shown, where the first line of the group is indicated by the right hand LED and the last one by the left hand LED.

Device	Symbol	State	Real-time display
Digital input			state: Enabled = GREEN, Disabled = GREY
Digital output			state: Enabled = RED, Disabled = GREY
Analog input		22.000	current value
Analog output		22.000	current numeric value in Volts
Input port			state of each line (as Digital input). State: Enabled = GREEN, Disabled = GREY
Output port			state of each line (as Digital output). State: Enabled = RED, Disabled = GREY
Input Nibble			state of each line (as Digital input). State: Enabled = GREEN, Disabled = GREY
Output Nibble			state of each line (as Digital output). State: Enabled = RED, Disabled = GREY
Axis		100.000	current absolute position
Timer		12.000	current value in seconds
Counter		58	current numeric value
Flag bit			state: Enabled = YELLOW, Disabled = GREY
Flag switch			state (as Flag bit). State: Enabled = YELLOW, Disabled = GREY
Flag port			state of each line (as Flag bit). State: Enabled = YELLOW, Disabled = GREY
Global variable		2	GPL code integer type global variable
Global variable		127	GPL code char type global variable
Global variable		50.00000000	GPL code float type global variable
Global variable		200.0000000	GPL code double type global variable
Global variable		Area	GPL code string type global variable
Global variable		[256]	GPL code array type global variable
Global variable		[10][3]	GPL code matrix type global variable

5.4 Interacting with Devices

It is possible to interact with devices to read their state or modify their value, for diagnostic purposes. However, this is not possible for some types of devices, such as input devices and other devices protected by the Manufacturer. Should the Operator try to operate on these devices, a message will notify him.

When the device has been selected, double click on it with the mouse, or press **Enter**, or the **Space Bar**, to access the window that allows you to change the state or the value of the device.

If the device concerned is a **Digital output** or a **Flag bit**, no window will appear, but the state of the device will be automatically changed. If the output is functioning correctly, the LED indicating its state will change colour.

If the device concerned is an **Output port** or an **Output Nibble**, point the mouse on the LED corresponding to the required output and *double click* on it to change its state.
The same applies to **Flag switches** and **Flag ports**.

As far as **Analog outputs**, **Timers** and **Counters** are concerned, a dialog window is displayed, showing the current value and enabling to set immediately the new value we want applied to the device.

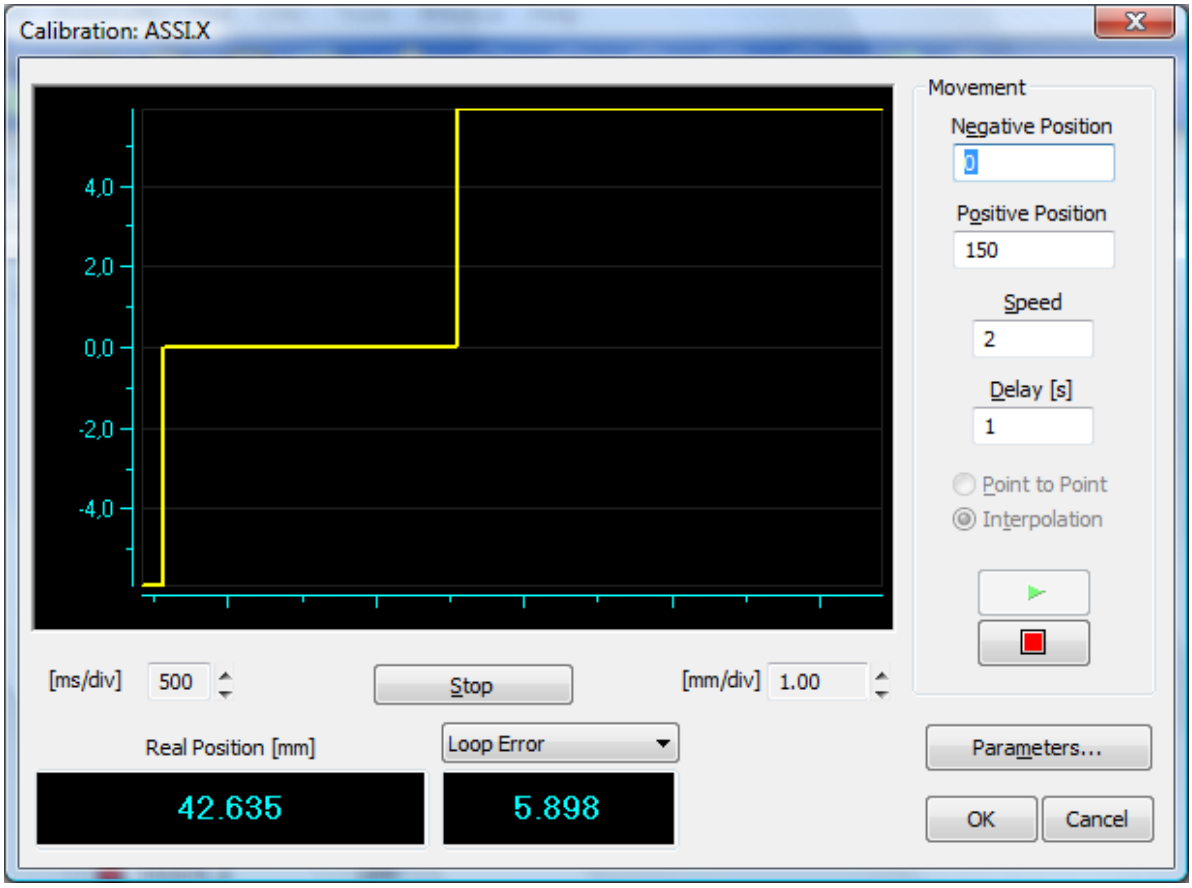
Axes interaction modes are described in the Manual Axes movement paragraph.

5.5 List of navigation keys to navigate through a tree structure

Key	Description
Up arrow Down arrow	moves the selection to the immediately previous row or to the following one
Right arrow	expands the selected branch to an extra level and, if already expanded, moves the selection on the next branch
Left arrow	collapses the selected branch and, if already collapsed, transfers the selection on the previous branch
+	expands the selected branch to one level
-	collapses the selected branch
*	expands all the levels of the selected branch

5.6 Axis calibration control board

The axis calibration control board allows you to modify axis configuration parameters and, at the same time, to move the axes and see its behaviour displayed on a virtual oscilloscope.
The axis calibration board is only accessible as from the "Manufacturer" access level. The calibration board is accessed in diagnostic or manual mode by double clicking on the axis to be calibrated while keeping the "shift" key pressed.
The calibration control board shown in the following figure will be displayed:



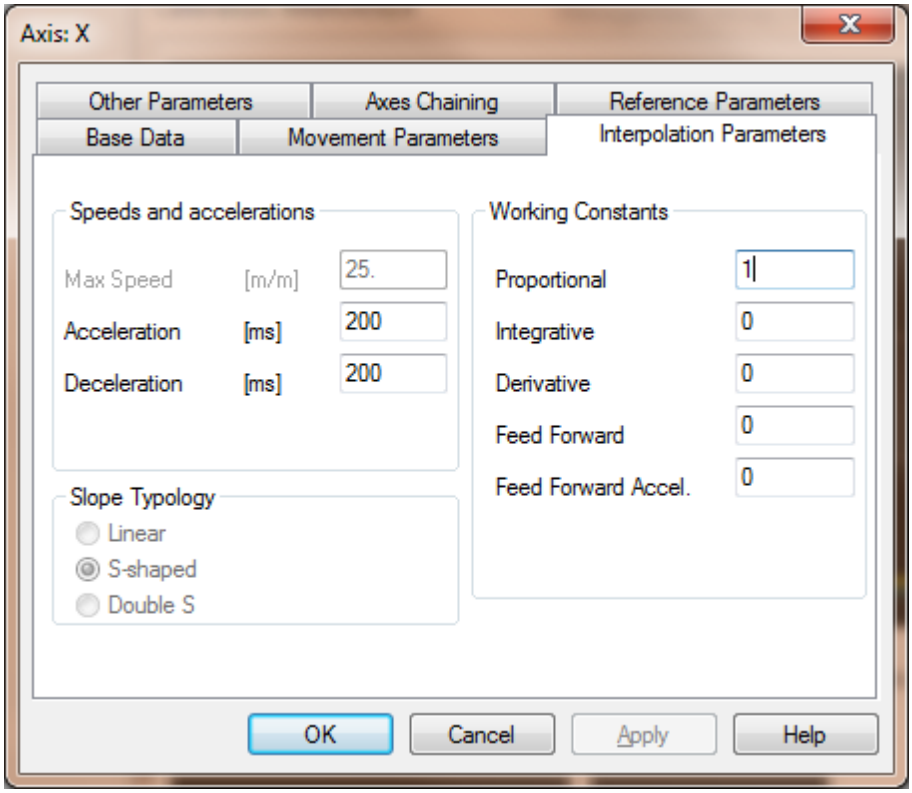
Axis calibration control board window

To verify axis behaviour as parameters change, the axis is moved continuously between two limit positions called **Positive Position** and **Negative Position**. As well as these parameters, axis movement **Speed** will also have to be set. In the early stages of calibration we suggest using a low speed value. A **Delay**, to be applied between movements, can also be fixed.

The oscilloscope window will display the axis loop error graph or one of the other axis' values. It is

possible, as with bench oscilloscopes, to scale the graph to adapt it to the size of the window and to examine it in detail. By means of a mouse or control keys and buttons you can examine the last calibration minute again, display one or two cursors to measure and check on the sampled data, enlarge an area of the graph to analyse the details of the sampled data, change the offset and the scale both in the x and y-coordinates. Moreover graph scrolling can be interrupted by pressing the **Stop** button, to allow a careful study of the graph without having to stop the axis.
Besides the graph, two boxes showing (on the left) the real position and (on the right) the size displayed graphically. This can be set using the combo box situated above the display box.

To change the parameters of the axes, press the **[Parameters...]** button, which activates the window shown in figure 6.2 where most axis parameters can be edited directly. Once a change has been made to one or more parameters it can be activated by pressing the **[Apply]** button. For such changes to be saved in the Configuration press the **[Ok]** button.



Parameter definition window for axis calibration

The principal parameters to be operated on are the following:

- **Proportional** coefficient
- **Integral** coefficient
- **Derivative** coefficient
- **Feed Forward** percentage of current speed provided directly by operation (independent of loop error)
- **Feed Forward Accel.** percentage of speed reference provided directly by operation during axis acceleration and deceleration phases (in addition to feed forward)
- **Acceleration** time of acceleration ramp
- **Deceleration** time of deceleration ramp

Axis Calibration
Axis calibration is a delicate operation to be carried out with great care and caution.

Through the "CalibSampleTime" option in the [Albatros] unit in Tpa.ini you can modify the data sampling time of an axis for the calibration window. The value in milliseconds and cannot be less than the frequency value of the control axis or less than 100.

Before calibrating the axes from the control board, set all the parameters in configuration and set the full-scale value for drive speed. At maximum speed Albatros reaches a voltage of 9 Volts.

To avoid damaging the machine by setting incorrect parameters, it is advisable to set a low speed, for

example the equivalent of 10% of axis max speed. This will avoid excessively violent reactions of the axes, even when the gain is set too high.

Normally the machine is calibrated first for point to point movements and then for interpolation movements.

The first step, if it hasn't been done in configuration, is setting the acceleration and deceleration times. The longer the time, less will be the acceleration to which the axis is submitted.

The second step is setting a minimum gain, that allows axis movement. This is necessary to verify the correct drive calibration. Albatros is set to provide a reference of 9 volts when it reaches the maximum speed set in axis configuration. For example, if the axis is moved at a speed corresponding to 10% of maximum speed, and if the drive is calibrated correctly, the reference power should be 1'0% of maximum power, that is 0.9 volts. If this reference voltage is not obtained, the drive's full-scale value must be modified.

When the drive has been calibrated, we begin to increase the position loop gain, a little at a time and with great caution. Each time the position loop is increased we must check that this has not caused conditions of excessive deflection or instability. In this phase, speed must be kept at 10% of max speed, or less, at all times. Moreover, it is always advisable to analyze the obtained speed profile carefully with the virtual oscilloscope, enlarging the image as much as possible to highlight the details.

When stable and ready axis performance has been obtained, movement speed can be gradually increased. Check axis behaviour each time the speed is modified. The value of the gain must also be modified if it is not satisfactory. Gain and speed must never be increased abruptly, as apparently stable calibration conditions at a low speed may not be as stable at a higher speed.

When the optimal value of the Gain has been determined, if necessary, Integrative and Derivative coefficients and then the Feed Forward may be gradually increased to reduce the loop error, bringing it within acceptably precise values. The feed forward allows you to eliminate the loop error almost completely during movement, but not during acceleration and deceleration. To further reduce the loop error in these phases, Feed Forward Acceleration can be increased. Normally, even very low values in this parameter are sufficient for satisfactory results.

As far as axis calibration for interpolation movements is concerned, the same values set for point to point movement can be used, although the other axes of the machine must be taken into account. It is particularly important to balance the axes' loop errors to obtain maximum precision during interpolation movements. This means that once the axis with the greatest loop error (at equal speed) has been identified, the calibration of the others must be adapted (limitedly to the interpolation parameters) to obtain identical loop errors.

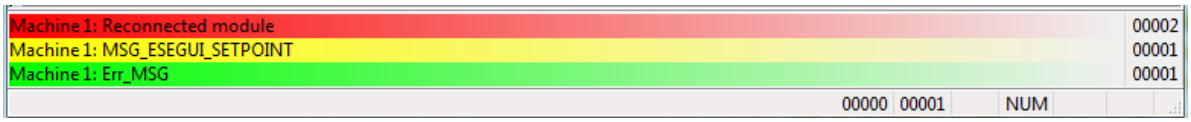
6 System Errors

6.1 Introduction to System Errors

By *System Errors* we mean all the errors that the *Albatros* system is able to detect automatically, both during program execution phases and during maintenance operations and plant diagnostic.

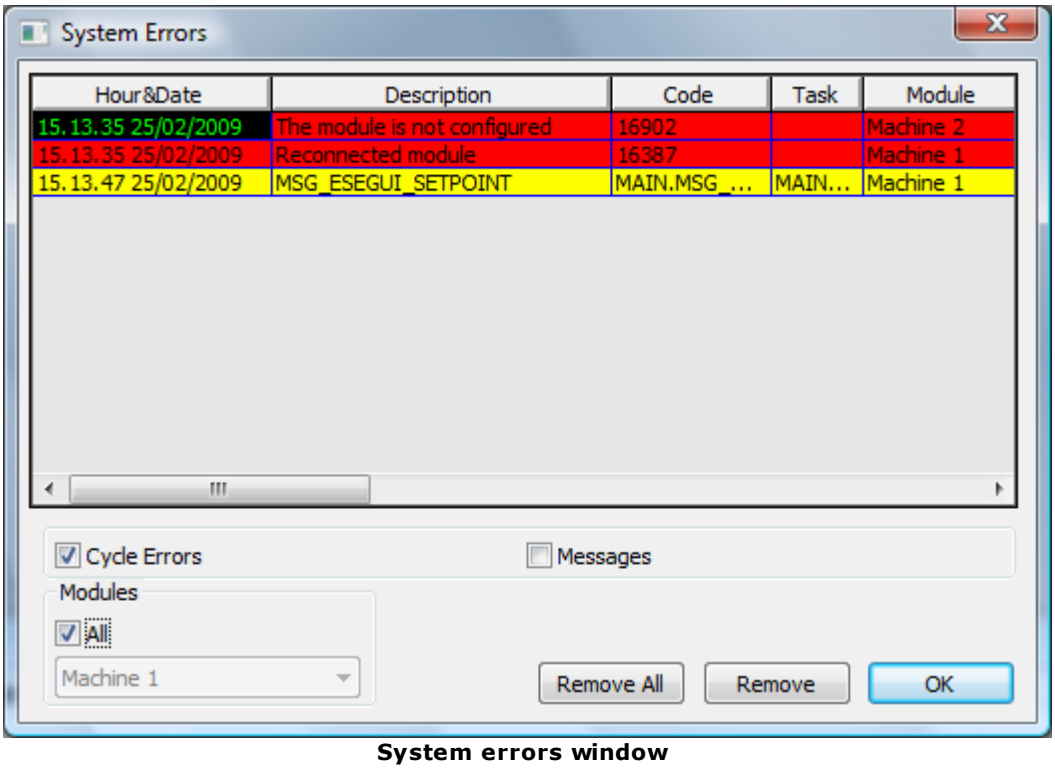
These include all kinds of errors, ranging from axis management problems to program execution problems.

System errors can be dealt with directly inside the machining program, by means of the ONERRSYS instruction. If this is not possible, program execution of the module where the error occurred is terminated.
The most recent system error is displayed in the Error Bar, together with the last cycle error and the last message.
System errors are indicated in red.
Cycle errors are indicated in yellow. These are errors occurring during program execution, which, however, usually allows execution to continue as soon as the error has been cleared.
Messages are indicated in green. These may be alert messages generated in anomalous situations during program execution or notifications of help request from the operator. They do not interrupt execution of the program itself.



Errors Bar

All the errors are saved in a file, creating an error data base. Any errors generated as from booting are displayed in a window opened either by double clicking on the *Errors bar* with the mouse or using the three commands in the ***View*** menu. Additional information about system errors is also displayed in this window.



System errors window

The window is divided into areas, as following: The **top** part of the window lists the following information:

- *Hour&Date* hour and date in which the error occurred.
- *Description* description of the error.
- *Code* number of error message.
- *Task* name of the task that generated the error (not indicated in Errors Bar).

By *double clicking* on one of these columns, the information is ordered according to the content of the column.
The **bottom** part contains the cells:

- *Cycle Errors* if this cell is enabled, cycle errors are also displayed .
- *Messages* if this cell is enabled, messages are also displayed.
- *All* if enabled, it lists all the messages, sent by any module of the system, concerning the type of information displayed.
- *Module Name Cell* shows the name of the module whose information is being displayed. It also allows you to select, in the case of multi-module systems, the module whose information is required.

The control **buttons** are:

- **[Remove All]** clears from memory all the information displayed, without deleting it from the file.
- **[Remove]** clears current information from memory, without deleting it from the file.
- **[OK]** closes window.

6.2 Errors generated by axes control

6.2.1 1 AxisName: incorrect encoder connection

Cause:
The difference between the theoretical position of a still axis, and the real position of the axis, exceeded 1024 encoder steps.
This often happens during axis commissioning, when the encoder phases are reversed. During normal functioning it occurs when an axis is moved manually, with the drive off, without setting the axis on FREE, or when the axis is subject to overshoot in the arrival phase, due to inaccurate calibration.
When this error occurs, the reference signal is set to zero and the axis is set on FREE.

Solution:
During axis commissioning, check the connection of the axis' encoder phases (if necessary enable the encoder phases inversion option in axis configuration).
Verify axis calibration using the specific Diagnostic mode.

6.2.2 2 AxisName: not ended movement

Cause:
When the move has concluded, 5 seconds after the end of theoretical movement, the gap between the theoretical position and the real position of the axis exceedes the window indicated in Configuration.
This could be simply because the drive is off or disabled or it could be due to inaccurate drive offset regulation.
However, it could also be due to mechanical backlash on the axis or an excessively low axis position loop gain.

Solution:
Check that the drive is on and enabled. Verify axis calibration and adjust the drive offset of the concerned axis.

6.2.3 3 AxisName: servoerror

Cause
In any type of movement, the difference between the theoretical position and the real position of the axis exceeded the maximum error indicated in Configuration or set with the instruction SETMAXER.
Normally this is due to the incorrect setting of the position loop gain or of the full-scale value of operation speed. It could also depend on excessive axis inertia.

Solution:
Verify the gain setting and the full-scale value of drive speed.
Check that the encoder and the motor/drive group are functioning correctly.

Check for any mechanical block.

6.2.4 4 AxisName: limit switch positive

Cause:
The theoretical position of the axis exceeded the positive position limit indicated in Configuration or set using the SETLIMPOS instruction.

Solution:
Correct in the program the position exceeding the positive position limit or set new axis position limits .

6.2.5 5 AxisName: limit switch negative

Cause:
The theoretical position of the axis exceeded the negative position limit indicated in Configuration or set using the SETLIMPOS instruction.

Solution:
Correct the position exceeding the negative position limit in the program or set new axis position limits .

6.2.6 6 AxisName: error writing SLM data

Cause:
The axis card does not communicate properly with the drive. Normally, this communication problem is due to the fact that the drive is switched off or the cables are not properly connected or are damaged. It is also possible that external power supply (+24VDC) of the AlbSLM card is absent.

Solution:
Check that the drive is switched on and is functioning correctly. Verify card power supply, verify connection cables. If the problem persists, please contact the constructor.

6.2.7 7 AxisName: error reading SLM data

Cause:
The axis card does not communicate properly with the drive. Normally, this communication problem is due to the fact that the drive is switched off or the cables are not properly connected or are damaged. It is also possible that external power supply (+24VDC) of the AlbSLM card is absent.

Solution:
Check that the drive is switched on and is functioning correctly, verify card power supply, verify connection cables. If the problem persists, please contact the constructor.

6.2.8 8 AxisName: error executing SLM command

Cause:
The axis card does not communicate properly with the drive. Normally, this communication problem is due to the fact that the drive is switched off or the cables are not properly connected or are damaged. It is also possible that external power supply (+24VDC) of the AlbSLM card is absent.

Solution:
Check that the drive is switched on and is functioning correctly, verify card power supply, verify connection cables. If the problem persists, please contact the constructor.

6.2.9 9 AxisName: communication error

Cause:
The axis card (AlbNt o ALbSLM) is not communicating correctly.

Solution:
Contact T.P.A. S.p.A.

6.2.10 10 AxisName: the Real-Time execution is faster than the profile construction

Cause
The realtime execution of the movement profile goes faster than the gpl generation of the profile itself. The lookahead is emptied faster than its filling. The error might be due to two generally simultaneous causes:

- the interpolation speed rate is too high with respect to the space dimensions to be covered.
- The spaces to be covered are too short.

Solution:
Verify that the set interpolation rate speed is not too high with respect to the space dimension to cover; furthermore, verify that the interpolation spaces to cover are not too shorts.

6.3 Errors generated by remote I/O

6.3.1 262 Test failed on dual-port memory of remote IO

Cause:
An error was generated during axis card initialization tests. Namely, the GreenBus transmitter (i296 microcontroller) Dual Port Memory test failed. This could be due to the incorrect configuration of the card's I/O and IRQ addresses or to conflict with other peripherals in the system. It could also be the consequence of a damaged axis card.

Solution:
Check the card configuration, and that there are no conflicts with other peripherals. Qualified technicians can test the Hardware of the i296 microcontroller Dual Port Memory. If the problem persists, please contact the constructor.

6.3.2 2049 Receiver number: incorrect configuration

Cause:
The remote receiver received a different I/O expansions configuration from the one detected on the field. This can happen, if the remote is not equal to the one chosen in the hardware configuration of Albatros. For example, the remote receiver is an Albre16 and in Albatros a remote ALbre24 (GreenBus v3.0) or another TRS-IO with a wrong TRS-IO-E expansion number (GreenBus v4.0) has been configured.

Solution:
Verify Hardware configuration.

6.3.3 2050 Receiver number: disconnected

Cause:
The remote receiver does not respond to the transmitter's commands.

Solution:
Verify the receiver's power supply and the serial connection.

6.3.4 2051 Receiver number: reconnected

Cause:
The connection between the transmitter and the receiver has been restored.

6.3.5 2052 Receiver number: error in reading Output not connected number OutputNumber

Cause:
The indicated digital output is in protection or in short circuit state, however, it is not in the state expected by control. The output is not associated to any logic device in Virtual-Phisical Configuration, which indicates an incongruity between Configuration and the real cabling of the machine.

Solution:
Verify Virtual-Physical Configuration. Remove short circuit or verify that the applied load does not exceed maximum limits (consult technical documentation).

6.3.6 2053 Receiver number: error on the expansion module number ModuleNumber

Cause:
Error related to Albrem modules and their relative AlbIN and AlbOUT expansions. The configuration read on the field during functioning does not correspond to initialisation configuration. Generally, this is due to an interruption of the connection between Albrem and the expansion modules.

Solution:
Verify connection between Albrem and expansions.

6.3.7 2054 Receiver number: wrong type

Cause:
During remotes initialization, a different receiver from the one specified in configuration has been detected at a certain address.

Solution:
Verify that the Hardware Configuration agrees with the remote module setting.

6.3.8 2055 Receiver number: initialized

Cause:
The receiver has reconnected to the transmitter after an interruption caused by a power fail.

6.3.9 2056 Receiver number: +24 V DC power fail

Cause:
Field power (+24 V DC) of a receiver module is not active or not working.

Solution:
Check the +24 V DC power.

6.3.10 2057 GreenBus power fail

Cause:
The field bus power supply, connecting the I/O modules with the control, is not working properly. This power supply should have a nominal value of +12 Vcc and it is supplied by the control.

Solution:
Check the presence of the GreenBus power line, check the GreenBus cables. Switch-off and switch back on. If necessary, replace the control board.

6.3.11 2058 Receiver number: error reading DeviceType DeviceName

Cause:
The state of the specified output does not correspond to the set state. This could be due to a shortcircuit, to overload protection or simply to the lack of power. The specified output can be a digital output, an analog output, an axis control output. The kind of output is specified in the error view.

Solution:
If it is a digital output, verify the +24V power supply (field side), remove the possible short circuit or the excessive output adsorption (see technical documentation). If it is an analog output or a axis control output, verify the presence and the value of the voltage set at the output (tester or oscilloscope), remove the possible short circuit or the excessive output adsorption (see technical documentation).

6.3.12 2059 Test failed on dualport memory of transmitter

Cause:
An error was generated during axis card initialization tests. Namely, initialization of the GreenBus transmitter (i296 microcontroller) failed. This could be due to the incorrect configuration of the card's I/O and IRQ addresses or to conflict with other peripherals in the system. It could also be the consequence of a damaged axis card.

Solution:
Check the card configuration, check that there are no conflicts with other peripherals. If a remote module is used, retransmit the firmware to the module. Qualified technicians can test the Hardware of the i296 microcontroller Dual Port Memory. If the problem persists, contact the manufacturer.

6.3.13 2060 Error initializing transmitter

Cause:
An error was generated during axis card initialization tests. Namely, firmware transmission to the GreenBus transmitter (i296 microcontroller) failed. This could be due to the incorrect configuration of the card's I/O and IRQ addresses or to conflict with other peripherals in the system. It could also be the consequence of a damaged axis card.

Solution:
Check the card configuration, check that there are no conflicts with other peripherals. If a remote module is used, retransmit the firmware to the module. Qualified technicians can test the Hardware of the i296 microcontroller Dual Port Memory. If the problem persists contact the manufacturer.

6.3.14 2061 Error transmitting firmware to transmitter

Cause:
An error was generated during axis card initialization tests. Namely, transmission of the remote I/O module configuration to the GreenBus transmitter (i296 microcontroller) failed.

Solution:
Verify hardware configuration, if a remote module is used, retransmit the firmware to the module. Qualified technicians can carry out a Hardware test on the i296 microcontroller RAM. If the problem persists contact the manufacturer.

6.3.15 2062 Error transmitting configuration to transmitter

Cause:
An error was generated during axis card initialization tests. Namely, remote I/O modules initialization failed.

Solution:
Verify the hardware configuration, if a remote module is used, retransmit the firmware to the module. Qualified technicians can carry out a Hardware test on the i296 microcontroller RAM. If the problem persists contact the manufacturer.

6.3.16 2063 Error transmitting configuration to receiver

Cause:
Error detected during initialization of a remote module.

Solution:
Verify the hardware configuration. Qualified technicians can carry out a Hardware test on the remote module. If the problem persists contact the technical support service.

6.3.17 2064 Receiver number: firmware version not compatible

Cause:
Remote receiver has a firmware version not compatible with the controller's firmware.

Solution:
Check installation of controller. If the problem persists, please contact the technical support service.

6.3.18 2065 Receiver number: error in an asynchronous communication

Cause:
There was an error or a non-response during the communication of a command with the remote (GreenBus v.4.0).

Solution:
Check the connections and the GreenBus power supply. If the problem persists, please contact the technical support service.

6.3.19 2066 Receiver number: generic error

Cause:
There was a generic error while communicating an event or an alarm from a remote (GreenBus v4.0)

Solution:
Check the connections and the GreenBus power supply. If the problem persists, please contact the technical support service.

6.3.20 2067 Receiver number: error during the transmission of the configuration

Cause:
A communication error while transmitting some configuration data to a remote (GreenBus v.4.0) occurred.

Solution:
Check the connections and the GreenBus power supply. Switch-off and switch back on. If the problem persists, please contact the technical support service.

6.3.21 2068 Receiver number: internal error no. errornumber

Cause:
An internal error on the indicated remote has occurred.

Solution:
Contact T.P.A. S.p.A.

6.3.22 2069 Receiver number: power supply error +24 Vcc bench number

Cause:
Field power supply (+24 Vcc) of a output group connected to the same feed clamp is not active or does not work correctly.

Solution:
Check the +24 Vcc power.

6.3.23 4610 Plug number: error reading DeviceType DeviceName

Cause:
The state of the specified output does not correspond to the set state. This could be due to a shortcircuit, to overload protection or simply to the lack of field side power (+24V) in the AlblO32 card (+24V).

Solution:
Verify that the +24V power supply is functioning correctly, if necessary remove the shortcircuit or verify that the load does not exceed limits (consult technical documentation).

6.4 Errors generated by Mechatrolink II

6.4.1 2308 Plug PlugNumber: inizialization failed. A parameter configuration setting was not correct

Cause:
In virtual physical Configuration any axis (logical device) was not connected to the board with Mechatrolink II bus (physical device).

Solution:
Check the connections in Virtual physical Configuration

6.4.2 2341 Plug PlugNumber: the number of servo drives exceeds the maximum allowed

Cause:
A number of servodrives, exceeding the configuration set, was connected to a Mechatrolink II bus.

Solution:
Check the Axis Control Frequency value in system configuration.

In the table below the right values to be set according the number of servodrives controlled by the board:

Board	Frequency Axis Control (Hz)	Maximum Number of servodrives
AlbMech	1000	8
AlbMech	<=500	16
DualMech Mono	1000	8
DualMech Mono	500	20
DualMech Mono	250	30
DualMech	1000	16
DualMech	500	40
DualMech	250	60

6.4.3 2342 Plug PlugNumber: the hardware address of the Servo servo drive exceeds the maximum value allowed

Cause:
An axis (logical device), whose hardware address (physical device) is higher than the number of the servodrives, that can be controlled by the board, has been connected to a Mechatrolink II bus.

Solution:
Check in system configuration the Axis Control Frequency value. In the table below the right values to be set according the number of servodrives controlled by the board:

Board	Axis Frequency Control (Hz)	Maximum Number of servodrives
AlbMech	1000	8
AlbMech	<=500	16
DualMech Mono	1000	8
DualMech Mono	500	20
DualMech Mono	250	30
DualMech	1000	16
DualMech	500	40
DualMech	250	60

Check in Physical-Virtual Configuration the connection between logical device and physical device. For example, if the maximum number of servodrives is 8, so the connection between logical and physical device must included among the first 8 axes (from Ax1 to Ax8).

6.4.4 2349 Plug PlugNumber: servodrive SERVO is not connected

Cause:
Physical connection to servodrive *PlugNumber* is interrupted.

Solution:
Check servodrive and Mechatrolink II bus cablings.

6.5 Errors generated by the CanBUS control

6.5.1 2761 Plug CAN number: node number: not connected

Cause:
The Can node shown seems currently not to be plugged to field bus, that makes reference to the board shown, although it is included in the configuration.

6.5.2 2762 Node number: replugged

Cause:
The Can node shown seems to be just plugged to field bus, that makes reference to the board shown.

6.5.3 2763 Node number: error missing transmission

Cause:
Error inside the indicated board. Data transmission to the indicated Can node failed.

Solution:
Contact T.P.A. S.p.A.

6.5.4 2764 Node number: error missing answer

Cause:
Data reception from the node Can failed.

Solution:
Check connection and power supply of the indicated Can device. Check wiring of the whole Can line. Check line connection to the numeric control. Check coherence in the protocol settings of the indicated device Can in respect of transmitter settings in the numeric control (baud rate, address, specific settings of the adopted protocol)

6.5.5 2765 Node number: inizialized

Cause:
The CAN node shown has been plugged to the field bus, and then it has been properly initialized

6.5.6 2766 Fault condition on CAN interface

Cause:
An internal power supply failure of the CAN Interface device is reported.

Solution:
Contact T.P.A. S.p.A.

6.5.7 2767 CAN open state loss

Cause:
Because of a serious problem CAN transmitter is not operational anymore.

Solution:
Contact T.P.A. S.p.A.

6.5.8 2768 Node number: error - PDO has not been received

Cause:
CAN node has not received the expected PDO.

Solution:
Check PDO node transmission settings.

6.5.9 2769 Node number: error receiving a non-configured node

Cause:
On the CAN network has been detected a node, that was not described in the file CANBUS.DEF.

Solution:
Check the node hardware address and the address declared in the CANBUS.DEF file.

6.5.10 2770 Node number: wrong configuration

Cause:
In the file CANBUS.DEF RPDO and TPDO data descriptions are wrong.

Solution:
Correct CANBUS.DEF file.

6.5.11 2771 Node number: SDO communication error

Cause:
CAN node has not responded in an asynchronous communication (SDO).

Solution:
Check connection node status. If the problem persists, contact T.P.A. S.p.A.

6.5.12 2772 Timeout on querying nodes CAN cycle

Cause:
A timeout error on querying nodes CAN cycle occurred

Solution:
In the file CANBUS.DEF change the value of the set sampling time.

6.5.13 3073 Node number: emergency error no.

Cause:
CANopen device has detected an error situation of the node, specified by the displayed code. It is a matter of error situations pertaining to a single node and in keeping with the standard CiA DS301-

EMERGENCY protocol.

Solution:
Make reference to the node documentation

6.5.14 3074 Node number: internal CAN error no.

Cause:
An internal error on the indicated node has occurred.

Solution:
Contact T.P.A. S.p.A.

6.5.15 3088 CAN Board number: Node NUMNODO: SDO communication error n. NUMERRORE - description

Cause:
In a ReadDictionary or WriteDictionary instruction, one or more requests for SDO read/write failed. The failure of the instructions can be caused, for example, by the read request of a CANOpen object that is not implemented in the device to which we are reffering to; or it can be related to the write, in a CANOpen register, of a not compatible data with the object type
(for example: writing attempt of a string into an object whose type is integer). The provided error code complies with the DS402 specifications and the textual description is also provided, in addition to the numerical code.

Solution:
check the parameters correctness of BAUDRATE, TIME, etc. set in the file CANBUS.DEF and the parameters of possible Readdictionary and/or Writedictionary instructions which are in the GPL code.

6.6 Errors generated by bus Powerlink control

6.6.1 2817 PowwerLink board number: none

Cause:
Missing board for the communication on PowerLink bus.

Solution:
Check that in the PC one of the communication boards, supported by T.P.A. is inserted and that the address on the board is correct.

6.6.2 2818 PowerLink board number: Type of CN unknown

Cause:
A CN has been defined with an unrecognised type.

Solution:
Check the definitions in the EPLBUS.DEF file: the two types of recognised CN are those defines as SERVO or as IO.

6.6.3 2819 PowerLink board number: wrong MN number

Cause:
A communication board on a PowerLink board could not be used.

Solution:

Check the definitions in the EPLBUS.DEF file

6.6.4 2820 PowerLink board number: wrong CN number

Cause:
One of the devices configured on the PowerLink bus could not be used.

Solution:
Check the definitions in the EPLBUS.DEF file

6.6.5 2822 PowerLink board number: Multiplexing setup error

Cause:
Multiplexing setup of one of the devices configured on PowerLink bus could not be used.

Solution:
Check the definitions in the EPLBUS.DEF file, in particular the MPX fields.

6.6.6 2823 PowerLink board number: Board initialisation error

Cause:
A communication board on PowerLink bus could not be initialised.

Solution:
Check that the address set on the board is the same as the address indicated in the EPLBUS.DEF configuration file.
Check that the model board is included into the controlled models and that it works properly.

6.6.7 2824 PowerLink board number: Wrong setting of the master board

Cause:
Configuration of more than one master board

Solution:
Check that in the EPLSBUS.DEF one only Master board is defined.

6.6.8 2825 PowerLink board number: Bus control frequency not correct

Cause:
Frequency of PowerLink control bus is not coherent with the programmed control frequency of the axes.

Solution:
Control frequency of the PowerLink bus is 1000Hz. The frequency corresponds to the MPX highest value and it should not exceed the frequency value of the control axis set in the configuration.

6.6.9 2826 PowerLink board number: Device connected to a board not configured

Cause:
One of the logical devices, mapped on PowerLink buses, is connected to a board, that is not present.

Solution:
Check the definitions in the EPLBUS.DEF file

6.6.10 2827 PowerLink board number: Wrong Board Number

Cause:
Error in the configuration of a board for the communication on PowerLink bus.

Solution:
Check the definitions in the EPLBUS.DEF file, in particular the MN fields.

6.6.11 2828 PowerLink board number: Disabling a board

Cause:
A communication board on PowerLink bus could not be disabled.

Solution:
Check that the address set on the board is the same as the address indicated in the EPLBUS.DEF configuration file.
Check that the model board is included into the controlled models and that it works properly.
Remove the board from the PC.

6.6.12 2829 PowerLink board number: Wrong PDO dimension

Cause:
The dimension of a PDO exceed the value of 1490 bytes.

Solution:
Modify the PDO definition in the EPLSBUS.DEF file, so that its dimension return into the limit.

6.7 Errors generated by bus EtherCAT control

6.7.1 3329 Error in communication socket initialisation

Cause:
The firmware could not communicate with the network card

Solution:
If the board has been configured in a RTX System, check that the ini.files available in the Albatros FW subfolder are properly written. To check the syntax of the files, see "Albatros installation manual".

6.7.2 3330 Error while scanning the EtherCAT network

Cause:
While prescanning the EtherCAT network, the master has not received any answer from some or from all the configured slaves or the configuration does not match the real network EtherCAT available in the field.

Solution:
Check the wiring between EtherCAT master and Slave
Check the descriptions in the ECATBUS.DEF file Hardware Diagnostic window helps you to find the error. Here, the existing nodes are displayed and, if wrongly configured, besides the name of the device found, the name of the expected device is displayed.

6.7.3 3331 Error in the configuration of the transmission mailbox

Cause:
The EtherCAT node has not responded to the command given by the Master. Potential causes: absent communication, faulty node ...

Solution:
Check the wiring and the remote operation.

6.7.4 3332 Error in the configuration of the receive mailbox

Cause:
The EtherCAT node has not responded to the command given by the Master. Potential causes: absent communication, node failure...

Solution:
Check the wiring and the remote operation.

6.7.5 3333 PowerLink board number: Error in the kind of expansions of the NUMNODO node

Cause:
The kind of expansions configured on an EtherCAT node in the ECATBUS.DEF file does not correspond to the kind of the expansion actually present. (For example, in the ECATBUS.DEF file a TRS-CAT has been defined with a TRS-IO-E expansion, while in the system a TRS-CAT is available with a TRS-AN-E expansion).

Solution:
Check that the devices described in the ECATBUS.DEF file correspond to those available.

6.7.6 3334 Error during the configuration of the PDOs

Cause:
The EtherCAT node, for which you tried to configure the PDOs, is not available or has a trouble.

Solution:
Check that the EtherCAT network configuration described in the ECATBUS.DEF file correspond to the physical network configuration.

6.7.7 3335 NUMNODO node in ErrorNumber alarm

Cause:
The indicated node is in an alarm situation.

Solution:
Check the alarm code in the following table

Alarm code	Description
0x0001	Unspecified error
0x0002	No memory
0x0011	Invalid requested state change
0x0012	Unknown requested state
0x0013	Bootstrap not supported
0x0014	No valid firmware
0x0015	Invalid mailbox configuration
0x0016	Invalid mailbox configuration
0x0017	Invalid sync manager configuration
0x0018	No valids inputs available
0x0019	No valid outputs
0x001A	Synchronization error
0x001B	Sync manager watchdog
0x001C	Invalid Sync Manager Types
0x001D	Invalid Output Configuration
0x001E	Invalid Input Configuration
0x001F	Invalid Watchdog Configuration
0x0020	Slave needs cold start
0x0021	Slave needs INIT
0x0022	Slave needs PREOP
0x0023	Slave needs SAFEOP
0x0024	Invalid input mapping
0x0025	Invalid output mapping
0x0026	Inconsistent settings
0x0027	Free-Run not supported
0x0028	Synchronization not supported
0x0029	Free-Run needs 3 buffer mode
0x002A	Background watchdog
0x002B	No valid inputs and outputs
0x002C	Fatal Sync error

0x002D	No Sync error
0x0030	Invalid DC SYNCH Configuration
0x0031	Invalid DC Latch Configuration
0x0032	PLL Error
0x0033	Invalid DC IO Error
0x0034	Invalid DC Timeout Error
0x0035	DC Invalid Sync Cycle Time
0x0036	DC Sync0 Cycle Time
0x0037	DC Sync1 Cycle Time
0x0041	MBX_AOE
0x0042	MBX_EOE
0x0043	MBX_COE
0x0044	MBX_FOE
0x0045	MBX_SOE
0x004F	MBX_VOE
0x0050	EEPROM no access
0x0051	EEPROM error
0x0060	Slave restarted locally

6.7.8 3336 EtherCAT board number: the NUMNODO node expansion number is wrong

Cause:
The configured expansion number on an EtherCAT node in the ECATBUS.DEF file does not correspond to the type of the expansion actually present. (For example, in the ECATBUS.DEF file a TRS-CAT has been defined with a TRS-IO-E expansion, while one only expansion is available in the system).

Solution:
Check that the devices described in the ECATBUS.DEF file correspond to those available.

6.7.9 4400 The node has not not responded to the request

Cause:
In a ReadDictionary or WriteDictionary instruction, performed on EtherCAT nodes, the node did not respond.

Solution:
Check if the the node has declared in the CN parameter of the GPL instruction is an existing node.

6.8 Errors generated by initialization

6.8.1 769 Error in software configuration

Cause
The hardware configuration of the remote module does not correspond to the software configuration specified in the system configuration.

Solution
Verify that the hardware parameters of the remote module and the software parameter correspond.

6.8.2 770 Wrong IRQ number in configuration

Cause:
The IRQ of the axis card has not been set correctly in the Module configuration. Normally a hardware conflict with other peripherals in the system is the cause.

Solution:

Verify in the motherboard BIOS settings that the IRQ used by the axis card is reserved for "Legacy ISA" only. Verify that no other peripherals are using the same IRQ assigned to the axis card. If possible, modify the IRQ of the peripheral in conflict with the axis card, otherwise modify the axis card IRQ.

6.8.3 772 Error reading backup memory

Cause:
An error was generated during axis card initialization tests. Namely, the buffered RAM (Dallas) test failed. This could be due to the incorrect configuration of the card's I/O and IRQ addresses or to conflict with other peripherals in the system. It could also be the consequence of a damaged axis card.

Solution:
Verify the hardware configuration. Qualified technicians may carry out a Hardware test on the i296 microcontroller RAM. Notice that the RAM Hardware test implies clearing all the data saved in it. The buffered RAM contains the values of certain devices, such as counters, timers and axis DAC offsets. Save these values before running the test.
If the problem persists, please contact the manufacturer.

6.8.4 773 Reached maximum number of axes in configuration

Cause:
An attempt to configure more axes than allowed.

Solution:
Reduce the number of axis to be configured. Please, contact T.P.A. S.p.A. for further information.

6.8.5 774 Axes Real-Time not running

Cause:
The axes management firmware was initialized but is not functioning properly. Normally, a hardware conflict with other peripherals in the system is the cause.

Solution:
Check that there is no conflict with other peripherals. Modify the configuration of the peripherals causing the conflict or remove these peripherals from the system.
The cause of a conflict can be an IRQ value assigned to the heading "IRQ line for albnt master" in module configuration different from the IRQ value set on albnt card.

6.8.6 775 Not enough time for GPL execution

Cause:
The execution of a real-time task takes up too much cycle time. This is generated when a Real-time task does not end before the start of the next axis real-time task (for example when an infinite cycle has been created).

Solution:
Change the GPL code so as to reduce the length of the real-time task.

6.8.7 776 Real-Time execution time too long

Cause:
The execution of a real-time task takes up too much cycle time. The execution time is slightly over the maximum allowed.

Solution:
Change the GPL code so as to reduce the length of the real-time task.

6.8.8 777 Watch Dog timeout

Cause:
The firmware is stuck.

Solution:
Contact T.P.A. S.p.A..

6.8.9 778 Main firmware code is blocked

Cause:
The firmware has crashed for more then 5 realtimes.

Solution:
Contact T.P.A. S.p.A..

6.8.10 779 Error in opening communication channel

Cause:
Couldn't open communication channel with cn2004 card.

Solution:
Check configuration of network card used for realtime TCP/IP communication with cn2004 card. If the problem persists, please contact the technical support service.

6.8.11 1025 Plug PlugNumber: don't reply

Cause:
An axis plug was detected during initialization but it does not respond correctly to commands.

Solution:
Qualified technicians can carry out a Hardware test on the axis card. If the problem persists, please contact the constructor.

6.8.12 1026 Plug PlugNumber: error on axes plug firmware transmission

Cause:
An axis plug was detected during initialization but it is not possible to transmit any firmware to it.

Solution:
Qualified technicians can carry out a Hardware test on the axis card. If the problem persists, please contact the manufacturer.

6.8.13 1028 Plug PlugNumber: firmware not present

Cause:
Firmware on board is not compliant with detected plug.

Solution:
Transmit correct version of firmware.

6.8.14 1029 Plug PlugNumber: MAIN blocked

Cause:
The axis card firmware was blocked during normal functioning.

Solution:
Qualified technicians can carry out a Hardware test on the axis card. Please, contact T.P.A. S.p.A.

6.8.15 1030 Plug PlugNumber: axes control blocked

Cause:
The axis card firmware was blocked during normal functioning for more than 5 realtimes

Solution:
Contact T.P.A. S.p.A.

6.8.16 1031 Plug PlugNumber: initialization error

Cause:
An error was generated during axis card initialization tests.

Solution:
Check and fix the causes of the system errors occurred in the moments before the occurrence of the current error.
Then initialize the system.

6.8.17 1032 Plug PlugNumber: test failed in dual-port memory

Cause:
An error was generated during axis card initialization tests. Namely, the i960 microcontroller's Dual Port Memory initialisation failed. Generally this is due to hardware conflict with other peripherals in the system, although it could also be the consequence of a damaged card.

Solution:
Verify the card configuration, check that there are not conflicts with other peripherals. If a remote module is used, retransmit the firmware to the module. Qualified technicians can carry out a Hardware test on the i960 microcontroller Dual Port Memory. If the problem persists, please contact the manufacturer.

6.8.18 1033 Plug PlugNumber: firmware Boot code is not running

Cause:
The i960 microcontroller's booting firmware was initialized but is not functioning correctly. Generally this is due to hardware conflict with other peripherals in the system.

Solution:
Verify the card configuration, check that there are not conflicts with other peripherals. If a remote module is used, retransmit the firmware to the module. Qualified technicians can carry out a Hardware test on the i960 microcontroller Dual Port Memory. If the problem persists, please contact the manufacturer.

6.8.19 1034 Plug PlugNumber: IRQ could not be set

Cause:
IRQ assigning to the AlbSLM axis card failed.

Solution:
Contact T.P.A. S.p.A..

6.8.20 1035 Plug PlugNumber: not present

Cause:
An error was generated during axis card initialization tests. Namely, the card was not detected.

Solution:
Verify that the card is actually in the system and that it is not damaged. Qualified technicians can carry out a Hardware test on the card. If the problem persists, please contact the manufacturer.

6.8.21 1036 Plug PlugNumber: error in software configuration

Cause:
The card hardware configuration does not correspond to the software configuration specified in system configuration. A possible cause, is that an axis was connected, in the Virtual-Physical Configuration, to the 5th, 6th, 7th or 8th connector of a card axis with only four connectors.

Solution:
Verify that the card hardware and software parameters correspond.

6.8.22 1037 Plug PlugNumber: failed opening dualport memory

Cause:
It failed opening the i960 microcontroller Dual Port Memory.

Solution:
Qualified technicians can run a Hardware test on the axis card. Please, contact T.P.A. S.p.A.

6.8.23 1038 Plug PlugNumber: error in ISA bus reading

Cause:
An error occurred during axis card access.

Solution:
Verify that the card is actually present in the system and that it is not damaged. Qualified technicians can carry out a Hardware test on the card. If the problem persists, please contact the constructor.

6.8.24 1039 Plug PlugNumber: Watch Dog timeout

Cause:
The firmware of the axis card *PlugNumber* is stuck.

Solution:
Contact T.P.A. S.p.A..

6.8.25 1040 Plug PlugNumber: +24 VDC power failed

Cause:
There is not any field power (+24 VDC) of the exits or it is not working properly

Solution:
Check the +24 VDC field power.

6.8.26 1041 Plug PlugNumber: +24 VDC SLM power failed

Cause:
Internal power supply of a cn2004 card does not fall within range prescribed by specifications. This power supply is also used as a backup supply of encoders for SLM drives.

Solution:
Check operation and stability of +24 VDC power supply. When using digital SLM drives, check their wiring (for short-circuits). If the problem persists, please contact the technical support service.

6.8.27 1042 Plug PlugNumber: +5 VDC power failed

Cause:
+5 VDC power supply of a cn2004 card does not fall within range prescribed by specifications. This power supply is available to supply external devices, such as encoders.

Solution:
Check wiring of external devices (for short-circuits). If the problem persists, please contact the

technical support service.

6.8.28 1043 Plug PlugNumber: axes power failed

Cause:
Power supply of DACs of a cn2004 card does not fall within range prescribed by specifications.

Solution:
Check wiring of axes (for short-circuits). If the problem persists, please contact the technical support service.

6.8.29 1044 Plug PlugNumber: disconnected

Cause:
The connection between the PC and a cn2004 board was interrupted.
The possible causes are as follows:
- power failure of the cn2004 board
- Ethernet cable disconnection, even if temporary, due to a false contact in the connectors or to damaged cables.
- firmware stop
- CPU reset due to overheating or to EM disturbance

Solution:
Verify that the module is switched on and properly powered. Verify Ethernet cables and connectors. If necessary update the firmware aboard. If the problem persists, please contact the technical support service.

6.8.30 1045 Plug PlugNumber: connected

Cause:
A cn2004 card has been connected to controller after initialization phase. Normally generated following a previous disconnection (error 1044).

6.8.31 1046 Plug PlugNumber: initialized

Cause:
A cn2004 card has been connected to controller after disconnection as a result of power failure.

6.8.32 1047 Plug PlugNumber: software configuration not permitted

Cause:
The device has received a configuration that is not compatible with the hardware in use or enabled. For instance, an axis has been configured whereas actual hardware axis is disabled.

Solution:
Verify that the card hardware and software parameters correspond.

6.8.33 1048 Plug PlugNumber: axis analog output write error

Cause:
Error detected during analogue output writing of an axis owing to a probable malfunction of the axis expansion module.

Solution:
Qualified technicians can carry out a Hardware test on the board. Please, contact the technical support service.

6.8.34 1049 Plug PlugNumber: analog output write error

Cause:
Error detected during analogue output writing.

Solution:
Qualified technicians can carry out a Hardware test on the board. Please, contact the technical support service.

6.8.35 1050 Plug PlugNumber: main power supply error

Cause:
Main power supply of a cn2004 card does not fall within range prescribed by specifications. Error may also be generated when power supply is irregular.

Solution:
Check operation and stability of power supply. If the problem persists, please contact the technical support service.

6.8.36 1051 Plug PlugNumber: firmware Main code is corrupt

Cause:
The cn2004 card is in Safe mode (only boot code active) because hardware has been forced (see card's technical literature).
It appears the flash memory is not enabled. This error generally indicates that a previous attempt to upgrade has failed, leaving the firmware code incomplete or corrupt.

Solution:
Upgrade firmware on the card, then switch off the card, disable Safe mode and switch back on. If the problem persists, please contact the technical support service.

6.8.37 1052 Plug PlugNumber: boot code is running

Cause:
The cn2004 card is in Safe mode (only boot code active) because hardware has been forced.
The card is generally set to Safe mode so that an upgrade of firmware on the card can be forced.
To find out how to enable or disable Safe mode, consult the card's technical literature.

Solution:
If necessary, upgrade firmware on the card, then switch off the card, disable Safe mode and switch back on. If the problem persists,please contact the technical support service.

6.8.38 1053 Plug PlugNumber: axes Watch Dog Expired

Cause:
A serious error while executing the firmware of the axis control board has occurred. Axes are disabled and the SYSOK signal, if any, is lowered. Do not reset the system.

Solution:
Contact T.P.A. S.p.A.

6.8.39 1054 Plug PlugNumber: axes Real-Time not running

Cause:
The cn2004 card has disabled axes and the SYSOK signal, if any, as a result of a serious system error on the ALBESLM plug. Do not reset the system.

Solution:
Contact T.P.A. S.p.A.

6.8.40 1055 Watchdog expired for plug PlugNumber

Cause:
The firmware of the card *PlugNumber* is stuck.

Solution:
Contact T.P.A. S.p.A..

6.8.41 1056 Plug PlugNumber: CAN Interface power failed

Cause:
Power supply of the transmission device on CanBus line in the indicated plug failed. It can depends on a short circuit, on an bus wiring error or on a damaged plug.

Solution:
Check the wiring of the whole Can line. Check the line connection to the numeric control. Remove the short circuit, if any. If the communication cannot be restored, please contact T.P.A. S.p.A.

6.8.42 1057 Plug NumeroPlug: Errore interno numero NumeroErrore

Causa:
Error in the hardware of the remote.

Solution:
Contact T.P.A.

6.9 Errors generated by memory management

6.9.1 1281 Error in the memory allocation on the heap area

Cause:
Available RAM memory is not sufficient to satisfy the requirement, for example, of a global matrix.

Solution:
Reduce the size of the global variables allocated in RAM.

6.9.2 1286 Error handling heap

Cause:
Error in the firmware's memory handling.

Solution:
Contact T.P.A. S.p.A.

6.9.3 1287 Too many heap memory deallocations

Cause:
Error in the firmware's memory handling.

Solution:
Contact T.P.A. S.p.A.

6.9.4 1289 Error creating global variables

Cause:
Too many global variables were defined, or the defined global matrixes are too large.

Solution:
Reduce the number of global variables or the size of the matrixes.

6.9.5 1290 Error in the dimension on non volatile variables

Cause:
Too many non volatile variables were defined, or the defined non volatile matrixes are too large.

Solution:
Reduce the number of non volatile variables or the size of the non volatile matrixes.

6.9.6 1291 Error in the dimension of read only variables

Cause:
Too many read only variables were defined, or the defined read only matrixes are too large.

Solution:
Reduce the number of read only variables or the size of the read only matrixes.

6.10 Errors generated by faults

6.10.1 1559 Breakpoint Trace

Cause:
Serious firmware error.

Solution:
Contact T.P.A. S.p.A.

6.10.2 1569 Invalid microprocessor operating code

Cause:
The microprocessor has encountered an unknown instruction. This could either be due to PC hardware problems or the files containing Albatros's firmware could be damaged.

Solution:
In the case of a local module, check that the files are not damaged and try reinstalling Albatros. In the case of Clipper modules, update the firmware. Run a PC hardware test, especially on the RAM. If the problem persists, please contact T.P.A. S.p.A.

6.10.3 1586 INTEGER value division by zero

Cause:
An attempt to divide an INTEGER by zero.

Solution:
Verify that all the divisions in the GPL functions are correct.

6.10.4 1600 Overflow in a floating point operation

Cause:
The result of an operation between FLOATs is greater than the capacity of the recipient.

± 3,402823E+38	for floats
± 1,79769313486231E+308	for doubles.

Solution:
Verify that the floating point calculations in the GPL functions are correct.

6.10.5 1601 Underflow in a floating point operation

Cause:
The result of an operation between FLOATs is smaller than the capacity of the recipient.

± 1,401298E-45	for floats
± 4,94065645841247E-324	for doubles.

Solution:
Verify that the float calculations in the GPL functions are correct.

6.10.6 1602 Invalid argument in a floating point operation

Cause:
An operand different from float type was used in a float operation.

Solution:
Verify that float calculations in the GPL functions are correct.

6.10.7 1603 Floating point value divided by zero

Cause:
An attempt to divide a float or double by zero. Raised also when a logarithm of zero is executed.

Solution:
Verify that all the divisions in the GPL functions are correct.

6.10.8 1604 Incorrect result in a floating point operation

Cause:
The result of an operation between floats is incorrect.

Solution:
Verify that the float calculations in the GPL functions are correct.

6.10.9 1605 Incorrect value for a floating point data

Cause:
The use of a smaller floating point value than the minimum representable value:

± 1,401298E-45	for floats
± 4,94065645841247E-324	for doubles.

Solution:
Verify that float calculations in the GPL functions are correct.

6.10.10 1728 Attempt to get access to an invalid address

Cause:
The program accessed an invalid memory area.

Solution:
Verify global/local variable congruity, if the problem persists, please report the anomaly.

6.10.11 1736 Data not aligned

Cause:
Serious error of the firmware.

Solution:
Contact T.P.A. S.p.A.

6.10.12 1735 Generic exception

Cause:
An unknown exception occurred.

Solution:
Contact T.P.A. S.p.A.

6.10.13 1801 Temperature alarm

Cause:
Temperature of controller's CPU has exceeded maximum permissible limits.

Solution:
Make sure there is no ventilation problem or anything causing overheating. If the problem persists, please contact the technical support service.

6.10.14 1802 Fan alarm

Cause:
Fan of controller's CPU is not working properly. Problem can lead to CPU overheating in no time.

Solution:
Contact technical support service.

6.10.15 1803 Unstable CPU frequency

Cause:
CPU work frequency is not stable.

Solution:
Contact T.P.A. S.p.A..

6.11 Errors generated by GPL functions

6.11.1 4097 The DeviceType DeviceName device is not configurated

Cause:
A GPL instruction used a non configured device, that is a device with no Virtual-Physical connection. It can be generated by all the instructions in which a device is passed as parameter.

Solution:
Check in the control configuration that all the devices used by the function have a Virtual-Physical connection. Then retransmit configurations to the card.

6.11.2 4098 Global variable VariableName missing

Cause:
A GPL instruction received an undefined global variable as argument.
This usually happens when the control was not correctly initialized.

Solution:
Recompile the whole GPL code and initialize control again.

6.11.3 4099 Function FunctionName missing

Cause:
An absent function was called.
It can occur when the control has not been initialized after modifying the GPL code.

Solution:
Recompile the whole GPL code and initialize control again.

6.11.4 4101 Inconsistent axis AxisName management

Cause:
An illegal state change was performed on an axis. For state changes consult the relative documentation.
The error could be generated by any of the instructions managing the axes, normally it occurs in the following cases:

- if an attempt is made to interpolate, coordinate an axis already occupied in a point-to-point movement (or vice versa).
- if a Chain, SetPFly or SetPZero instruction is executed on an axis in transparent mode.
- if an attempt is made to interpolate, coordinate a slave axis.

Solution:
Check that all axis transfers end with a wait in position instruction, especially if the axes alternate different types of movements (point-to-point, interpolation, etc)

6.11.5 4105 Instruction not executable on step motor axis AxisName

Cause:
An attempt to execute an instruction on an axis which does not support it. For example, an interpolation instruction on a step-by-step axis.

Solution:
Correct the GPL code.

6.11.6 4106 Remote module of the setp-by-step axis AxisName is not connected

Cause:
An attempt to operate on a step-by-step axis that is not connected to the control.

Solution:
Check the connection of the remote controlling the axis.

6.11.7 4107 SYSOK instruction with incorrect arguments

Cause:
A SYSOK instruction with incorrect arguments was executed. Verify whether one or more digital outputs passed as instruction arguments are not correctly configured.

Solution:
Verify the GPL code and the Virtual-Physical configuration.

6.11.8 4108 AxisName: Final quote beyond software limits

Cause:
An attempt to move an axis beyond the limits set in configuration or by the GPL code.

Solution:
Correct the machining program that caused the error. If necessary, correct the GPL code or axis configuration.

6.11.9 4110 Wrong speed

Cause:
An axis was assigned a null or negative speed.

Solution:
Correct GPL code.

6.11.10 4111 Negative Acceleration on axis AxisName

Cause:
An axis was assigned negative acceleration.

Solution:
Correct GPL code.

6.11.11 4112 Negative Deceleration on axis AxisName

Cause:
An axis was assigned negative deceleration.

Solution:
Correct the GPL code.

6.11.12 4113 Axis AxisName: SLM command TimeOut expired

Cause:
An SLM command was not executed within maximum time allowed.

Solution:
Contact T.P.A. S.p.A.

6.11.13 4114 Axis AxisName: reset on Fast Input not effected

Cause:
The Fast Input Reset (on the fly homing) was not completed correctly. This procedure enables to reset to zero the position of a moving axis, the moment the corresponding fast input changes state. If the axis concludes the movement in process with no input switching, the system error is generated. This could be due to the incorrect setting of axis movement parameters or to a cabling problem in the fast input.

Solution:
Check the GPL code implementing on the fly homing, check fast input cabling.

6.11.14 4115 Axis AxisName: zero pulse not found

Cause:
The encoder zero pulse reset was not completed correctly. This procedure enables to reset to zero the position of a moving axis the moment the encoder 's zero pulse is detected. If the axis reaches the pulse search position without detecting the zero pulse, the system error is generated. This could be due to the incorrect setting of axis movement parameters or to a cabling problem in the pulse signal (axis connector C phase).

Solution:
Check the GPL code implementing pulse homing, check axis cabling.

6.11.15 4353 Unknown instruction operational code (Function: FunctionName line:LineNumber)

Cause:
An illegal instruction was detected during the execution of a GPL function. Generally this indicates that the files containing the compiled GPL code are damaged. Verify also whether the control software and firmware were updated without recompiling the GPL code, as the earlier version could contain instructions which are no longer supported by the new one.

Solution:
Recompile the whole GPL code and initialize control. If the problem persists, please contact T.P.A. S.p.A.

6.11.16 4354 Incorrect mathematical operation (Function: FunctionName: LineNumber)

Cause:
A GPL instruction tried executing an incorrect mathematical operation, such as dividing by zero. Or data introduced in the GPL instruction is incongruent. This error is often generated by interpolation movement instructions, as this is the Firmware that performs the most mathematical operations.

Solution:
Check that the data passed to interpolation instructions is correct. If the problem persists, please report the problem to T.P.A. S.p.A..

6.11.17 4355 Incorrect address of matrix or vector (Function: FunctionName line: LineNumber)

Cause:
A GPL instruction tried accessing an array or matrix element exceeding maximum size. For example, it tried accessing element 10 of a 5-element array.
It could be generated by any instruction accepting an array or matrix as a parameter.

Solution:
Verify that all the matrix and array indexes passed to the instructions are within the array and matrix size.

6.11.18 4356 Instruction RET without CALL (Function: FunctionName line: LineNumber)

Cause:
A RET instruction was executed although the stack did not contain the relative return address. Declaring a subprocedure before the exit function FRET instruction, without protecting it with a GOTO to avoid accidental execution, is the most frequent cause. It is also possible that an accidental jump occurred in a subprocedure.

Solution:
Check the GPL program flow. When possible, place subprocedures at the end of the body of the function (after the FRET instruction)

6.11.19 4357 Local variable missing (Function:FunctionName line LineNumber)

Cause:
A GPL instruction tried to access a local variable which has not been allocated.

Solution:
Recompile and retransmit all card functions. If the problem persists, please report the problem.

6.11.20 4358 Jump label missing Function: FunctionName line: LineNumber)

Cause:
A GPL instruction jumped to a non-existing jump label.
It can be generated by GOTO, CALL, FCALL and all IFs.

Solution:
Recompile and retransmit all the functions to the card. If the problem persists, please report the problem.

6.11.21 4359 Incorrect macro argument (Function:FunctionName line: LineNumber)

Cause:
A GPL instruction was passed invalid arguments. It can be generated by any instruction. However, in the great majority of cases, the GPL system tries to correct the situation automatically, by performing automatic type conversions (cast), which may imply wasting time. The error is generated when these

conversions are not possible and especially in the following cases:
- instructions operating on specific devices (SETTIMER, SETCOUNTER) that are given a different type of device.
- instructions operating on bits that are given a floating point number (AND, OR, etc)
- instructions operating on matrixes or arrays that are given a simple variable (SORT, MOVEMAT, etc.)
- instructions that operate on strings that are not given strings.

The error is generated even when the system tries to carry out an instruction in a board that does not manage such an instruction (for example an instruction SENDPDO or an instruction RECEIVEPDO in a board that is not a TMSCan or a TMSCan+ board)

Solution:
Correct the GPL code.

6.11.22 4360 Error in the memory allocation for executing (Function: FunctionName line:LineNumber

Cause:
The GPL function tried to allocate a region of memory for internal use, but did not find available memory.
The error could indicate a temporary situation, due, for example, to an excessive number of tasks in execution at the same time or to excessively large global variables.

Solution:
Check the size of the global and local variables and if possible reduce their size. Verify if too many tasks are in execution at the same time and if necessary reduce them.

6.11.23 4361 Too many tasks enabled (Function: FunctionName line: LineNumber)

Cause:
An attempt to execute more than 256 tasks at the same time.

Solution:
Reduce the number of tasks in execution at the same time.

6.11.24 4362 Incorrect matrix format (Function: FunctionName line: LineNumber)

Cause:
An instruction operating on matrixes has found an invalid format. The instructions that could generate this error are the following:
- MOVEMAT if the format of the source matrix and the destination matrix do not correspond.
- CLEAR if a non-existing row of the matrix is being deleted.
- GETAXIS if the format of the matrix, passed as a parameter, does not correspond to the format expected by the instruction (consult GPL language documentation)

Solution:
Verify the above mentioned instructions in the task that generated the error. Check especially that the matrixes passed to MOVEMAT have the same number of columns of the same type and that the matrix passed to GETAXIS has the right format.

6.11.25 4363 Too many active ONINPUT instruction (Function: FunctionName line:LineNumber)

Cause:
More than 128 OnINput instructions have been activated.

Solution:
Reduce the number of ONINPUT.

6.11.26 4364 Already engaged axis with local reference (Function: FunctionName line:LineNumber)

Cause:
The error concerns the activation of the rototraslate axis terns to execute interpolations on a number of Cartesian axes.
There was an attempt to execute a SETRIFLOC passing to the instruction an axis that was already engaged in a reference axis tern. It can also be generated if a RESRIFLOC is executed on an axis which is not engaged in any axis tern. It is also possible that no reference terns were available (there can be a maximum of 32 terns).

Solution:
Check that the terns passed by the SETRIFLOC have no axes in common.
Check RESRIFLOCs.
Check that the RESRIFLOC is preceded by wait in position instructions.
Remember that the RESRIFLOC is not executed until the interpolation has concluded.

6.11.27 4365 Instruction ONINPUT actived on the same INPUT Function: FunctionName line: LineNumber

Cause:
The same input was passed to an ONINPUT instruction more than once.

Solution:
Check that the same input is not sent as a parameter to two ONINPUTs.

6.11.28 4366 Too many ONFLAG instruction active (Function: FunctionName line: LineNumber)

Cause:
More than 128 OnFlags instructions were activated.

Solution:
Reduce the number of ONFLAGs.

6.11.29 4367 Instruction ONFLAG actived on the same FLAG (Function: FunctionName line:LineNumber)

Cause:
An ONFLAG instruction was passed to the same flag more than once.

Solution:
Check that the same flag is not passed as a parameter to two ONFLAGs.

6.11.30 4368 A ReadOnly variable writing has been attempted (Function: FunctionName line: Line Number)

Cause:
An attempt to write on a readonly variable.
Readonly variables are always global and reside in the command flash. They are indicated as "static" in the global variables editor. If an attempt is made to write on one of these global variables this system error is generated.
The error is also generated if variables residing in the buffered RAM ("non volatile") are used as arguments of certain write instructions.
These instructions are:
- CRF the matrix passed for the milling cutter radius correction must be in RAM.
- COORDIN the variable passed to indicate the row in elaboration must be in RAM.

Solution:
Check all the static and non volatile variables.

6.11.31 4369 Too many master axes active (Function: FunctionName line: LineNumber)

Cause:
An attempt to activate more than four axes as master at the same time.
This error is only generated by the CHAIN instruction.

Solution:
Reduce the number of master axes.

6.11.32 4370 Too many slave axes active (Function: FunctionName line: LineNumber)

Cause:
An attempt to activate more than eight axes as slaves of a single master axis.
This error is only generated by the CHAIN instruction.

Solution:
Reduce the number of slave axes.

6.11.33 4372 Incorrect use of an instruction (Function: FunctionName line LineNumber)

Cause:
An attempt to use a mailbox handling instruction (Sendmail, Waitmail, Endmail, Ifmail) inside a function called by an Errsys, Oninput or Onflag instruction.

Solution:
Clear the instruction that caused the error.

6.11.34 4373 Can't read feed rate (Function: FunctionName line: LineNumber)

Cause:
The Getfeed instruction was used on another card, instead of the ALBNT card or ALBSLM or ALBMEC or CN2004.

Solution:
Clear the instruction GetFeed from the function code. In hardware configuration check that the master board is an ALBNT or an ALBSLM or an ALBMEC or a CN2004 one.

6.11.35 4374 Too many IPC instruction in execution (Function: FunctionName line: LineNumber)

Cause:
The maximum limit of 16 IPC instruction in execution at the same time was exceeded.

Solution:
Modify the GPL code.

6.11.36 4375 Not all axes are connected to the same board (Function: FunctionName line: LineNumber)

Cause:
A FASTREAD instruction was executed, although the axes passed as parameters were not all connected to the same board.

Solution:
Modify the GPL code or the Virtual-Physical configuration as required.

6.11.37 4378 Instruction not enabled (Function: FunctionName line LineNumber)

Cause:
An attempt to use an instruction whose execution was not enabled. Probably, the hardware key is not correctly inserted or is missing.

Solution:
Insert the hardware key correctly. If the problem persists, contact the manufacturer.

6.11.38 4379 Instruction can't be used in function called by events (Function: FunctionName line: LineNumber)

Cause:
An attempt to use an illegal instruction in a function launched by interrupt. The functions launched by interrupt are passed as parameters to ONERRSYS, ONINPUT and ONFLAG instructions.

Solution:
Modify GPL code. Consult the list of instructions which can not be used with interrupt

6.11.39 4380 Too many writing request into backup memory area (Function: FunctionName line: LineNumber)

Cause:
Too many write operations were performed on the buffered memory at the same time (buffered memory is characterised by relatively slow access).

Solution:
Verify the instructions that perform write operations on the variables allocated in the buffered memory: counters, timers, matrixes and variables declared as "non volatile".

6.11.40 4381 Can't use a serial channel not yet open (Function: FunctionName line: LineNumber)

Cause:
An attempt to execute an instruction that operates on the serial port, before executing the COMOPEN instruction for this port.

Solution:
Modify GPL code.

6.11.41 4382 Can't open a serial channel already open (Function: FunctionName line: LineNumber)

Cause:
A COMOPEN instruction was executed on a serial port that has already been opened with the same instruction.

Solution:
Modify GPL code.

6.11.42 4383 Too many auxiliary processes open (Function: FunctionName line: LineNumber)

Cause:
An attempt to open more than 4 auxiliary processes at the same time.

Solution:
Modify GPL code.

6.11.43 4384 Auxiliary process not in execution (Function: FunctionName line: LineNumber)

Causa:
An attempt to access an auxiliary process which is not in execution.

Solution:
Modify GPL code.

6.11.44 4385 Attempt to open an auxiliary process from another task (Function: FunctionName line: LineNumber)

Cause:
An attempt to open an auxiliary process from a different task from the one that started execution. Auxiliary tasks can only be used by the tasks that started their execution.

Solution:
Modify GPL code.

6.11.45 4386 Attempt to use a CanBUS communication port not yet opened (Function: FunctionName line: LineNumber)

Cause:
An attempt to execute an instruction that operates on a CanBUS communication port, before executing the CANOPENDRIVER instruction on the port.

Solution:
Modify GPL code.

6.11.46 4388 Attempt to close a CanBUS communication port not yet opened (Function: FunctionName line: LineNumber)

Cause:
An attempt to close a CanBUS communication port, without having previously executed the CANOPENDRIVER instruction for this port.

Solution:
Modify GPL code.

6.11.47 4387 Error opening CanBUS communication (Function: FunctionName line: LineNumber)

Cause:
The CanBUS card initialization was not completed successfully. This could be due to an incorrect card configuration or to hardware conflicts with other components in the system.

Solution:
Verify the correct configuration of the CanBUS card. Check that there are no hardware conflicts. If the problem persists, contact T.P.A. S.p.A.

6.11.48 4389 Attempt to open a CanBUS communication already opened (Function: FunctionName line: LineNumber)

Cause:
An attempt to open a CanBUS communication port which had already been opened with the CANOPENDRIVER instruction.

Solution:
Modify GPL code.

6.11.49 4390 CanBUS communication error (Function: FunctionName line: LineNumber)

Cause:
An error occurred while using the GPL CANGETOBJECT or CANSETOBJECT functions. This could be due to the incorrect configuration of the CanBUS devices, to the use of incorrect parameters or to cabling problems. The parameters and configuration depend on the specific CanBUS device used, consult the technical documentation provided by the manufacturer.

Solution:
Check that the parameters used to access the devices are correct. Verify the devices' configuration and make sure that the devices' cabling is correct.

6.11.50 4391 Error activating SYSOK (Function: FunctionName line: LineNumber)

Cause:
The SYSOK signal activation was not successfully concluded. This is often due to a malfunctioning Greenbus transmitter on the axis card.

Solution:
Qualified technicians can perform a Hardware test on the i296 microcontroller Dual Port Memory. If the problem persists, please contact the constructor.

6.11.51 4392 Synchronized movement channel not opened (Function: FunctionName line: LineNumber)

Cause:
A synchronized movement instruction has been executed without having previously opened the corresponding channel using the command SYNCROOPEN.

Solution:
Correct the GPL code. Please, contact the machine constructor.

6.11.52 4393 No lines to process (Function: FunctionName line: LineNumber)

Cause:
The processing of a synchronized movement profile has been activated without the crossing points being previously assigned using the command SYNCROMOVE.

Solution:
Correct the GPL code. Please, contact the machine constructor.

6.11.53 4394 Too many cycle errors (Function: FunctionName line: LineNumber)

Cause:
There are more than 2000 cycle errors active.

Solution:
Correct GPL code by limiting the number of warning signals.

6.11.54 4395 Too many messages (Function: FunctionName line: LineNumber)

Cause:
There are ore than 2000 messages active.

Solution:
Correct GPL code by limiting the number of warning signals.

6.11.55 4397 Stack overflow on the function FunctionName (Function: FunctionName line: LineNumber)

Cause:
A GPL function stack exceeded the maximum limit of 2Kbyte.

Solution:
Compile the GPL code again and check in the compiler report the estimated stack space of the function that generated the system error. Then reduce the number of local variables and of parameters passed to the functions (replacing them, for example with global variables). Reduce the number of CALLs.

6.11.56 4398 Stack underflow on the function FunctionName (Function: FunctionName line: LineNumber)

Cause:
It can only occur in the case of a serious Firmware error, such as the incorrect management of function parameters or local variables.

Solution:
Contact T.P.A. S.p.A.

6.11.57 4399 Parameter out of range (Function: FunctionName line: LineNumber)

Cause:
A GPL variable or a device was assigned a value outside the allowed range.

Solution:
Correct and compile the GPL code again.

6.11.58 4865 The definition of the machine for the interpolation (G216 or G217) is missing

Cause
An attempt to move the axes with an ISO interpolation or the configuration indices have been set without defining in advance the configuration matrices and the axes that form the machine.

Solution:
Correct and compile the GPL code again, using the instructions ISOG216 .

6.11.59 4866 The definition of the indeces of the selected machine configuration (M6) is missing

Cause
An attempt to move the axes with an ISO interpolation without defining in advance the indices of the machine's configuration matrices.

Solution:
Correct and compile the GPL code again, using the instruction ISOM6.

6.12 Errors generated by CNCTPA communication driver

6.12.1 16385 Disconnected module

Cause:
The connection between the Supervisor PC and a module was interrupted.
The possible causes are the following:
- power failure of the remote module
- Ethernet cable disconnection, even if temporary, due to a false contact in the connectors or to damaged cables.
- power failure or malfunctioning of the Ethernet hub (if present)
- interruption of the remote module firmware due to damaged configuration files.

- remote module CPU reset due to overheating or to EM disturbance.

Solution:
Verify that the module is switched on. Verify Ethernet cables and connectors. Update the firmware in the remote module. Check that this module has not overheated because of insufficient ventilation and that it is not subject to EM disturbance. If the problem persists, please contact T.P.A. S.p.A.

6.12.2 16386 Connected module

Cause:
A remote module was connected to the Supervisor PC after Albatros' initialisation phase. Albatros tries to connect all the modules indicated in the System Configuration during booting, which lasts approximately 4 seconds. Any module connected later generates a system error.

6.12.3 16387 Reconnected module

Cause:
A remote module was reconnected to the Supervisor PC after being disconnected. This error always follows error 16385: "Disconnected module".

6.12.4 16388 Initialized module

Cause:
A remote module was reinitialized during normal functioning. This implies that the module was disconnected and reconnected to the Supervisor PC beforehand. Therefore this error always follows error 16385: "Disconnected module".
It indicates the module reset due, for example, to power failure.

6.12.5 16389 Module interrupted connection

Cause:
A remote module has closed the connection with Albatros. This may happen when the module does not receive any command or query from the Supervisor PC for a long time. This error shows a problem (overload or deadlock) on the Supervisor PC.

Solution:
Check the Supervisor PC for programs that may cause overloads or deadlocks. Disable the screen saver on the Supervisor PC. If the problem persists, contact the machine constructor.

6.12.6 16641 AlbRtx doesn't answer to the commands

Cause:
An error arose during system initialization. Specifically, firmware does not respond as expected. This fault might be caused by a damaged firmware file.

Solution:
Try to reset system and if necessary install Albatros again. If the problem persists, contact T.P.A. S.p.A.

6.12.7 16642 AlbNet doesn't answer to the commands

Cause:
An error arose during system initialization. Specifically, communication software with remote modules does not respond as expected. This fault might be caused by a damaged software file.

Solution:
Try to reset system and if necessary install Albatros again. If the problem persists, please contact T.P.A. S.p.A.

6.12.8 16643 Operating System doesn't allow use of RTX

Cause:
The Operating System installed on the PC does not permit the use of RTX and consequently doesn't allow correct operation of versions of Albatros requiring its presence.

Solution:
Update the PC Operating System. Check minimum system requirements on the "Installation Manual of Albatros".

6.12.9 16644 Operating System doesn't allow use of AlbNet

Cause:
The Operating System installed on the PC does not allow correct operation of current version of Albatros.

Solution:
Update the PC Operating System. Check minimum system requirements on the "Installation Manual of Albatros".

6.12.10 16645 Error sending firmware code ...

Cause:
An error arose during system initialization. Specifically, transmission of a firmware file to a module failed.

Solution:
Try to reset system. If the problem persists, contact T.P.A. S.p.A.

6.12.11 16646 Could not restart firmware code

Cause:
An error arose during system initialization. Specifically, firmware failed restarting after a previous stop.

Solution:
Try to reset system. If the problem persists, contact T.P.A. S.p.A.

6.12.12 16647 Error while sending CanBUS hardware configuration

Cause:
An error was generated during the transmission to CAN master card of devices configuration. Error occurs when file could’t be registered into remote module or the dowload of file in CAN card flash memory failed. Configuration file is a binary file wich name is CANBUS followed by card number (from 0 to 3) and with extension DBM and it is refistered into module CONFIG folder. It exists a CANBUSn.DBM for every card present in the system.

Solution:
Check that file is complete, that it has a size not higher than a little number of Kbs, that card is running. If the problem persists, contact T.P.A. S.p.A.

6.12.13 16897 RTX not installed

Cause:
The installed version of Albatros requires RTX installed on the PC; however, this has not been detected.

Solution:
Install RTX or install it again, if already loaded. Consult "Installation Manual of Albatros".

6.12.14 16898 User has no Administrator rights

Cause:
Albatros was started up by a user without Administrator rights on the PC. Administrator rights are required for correct operation of Albatros.

Solution:
Close current working session and access system as "Administrator" or as other user with Administrator rights.

6.12.15 16899 Wrong dimension of module RAM

Cause:
RAM dimension detected on remote module is incongruent with expected dimension. This fault is normally caused by a hardware failure.

Solution:
If the problem persists, contact T.P.A. S.p.A.

6.12.16 16900 Incorrect module IP address

Cause:
A remote module has been detected whose IP address does not belong to the supervisor PC subnet. Albatros cannot communicate correctly with the module.

Solution:
Check settings of AlbDHCP service and of LAN board on the supervisor PC. Consult "Installation Manual of Albatros".

6.12.17 16901 Module is already connected to another equipment

Cause:
A remote module appears to be connected to a different supervisor PC. This may be caused by presence on the network of another PC with Albatros running and using the same module. It may also be caused by a failure of the communication software on the module.

Solution:
Check that no other supervisor PC is using the remote module. Reset the module. If the problem persists, please contact T.P.A. S.p.A.

6.12.18 16902 The module is not configured

Cause:
A module appears not to be configured in "System Configuration".

Solution:
Configure the module.

6.12.19 16903 Firewall settings prevent communication

Cause:
A firewall blocking communication between Albatros and remote modules has been detected.

Note: Albatros can identify Windows Xp firewall only and not other firewalls as those included in some antivirus software packages.

Solution:
Modify firewall settings or disable it.

6.12.20 16904 Network card not present or disabled

Cause:
No network card available for connection to remote modules has been found.

Note: the detection of a network card does not grant proper settings and connection.

Solution:
Check the network card and its configuration. If the problem persists, contact the machine constructor

6.12.21 16905 Main firmware code missing

Cause:
Albatros can't find a firmware file on the PC hard disk. The problem may be caused by an accidental file deletion or a incorrect update.

Solution:
Check that files in the FW subfolder of Albatros setup are present and have the right version number. Contact the machine manufacturer.

6.12.22 16906 RTX version incompatible with main firmware code

Cause:
RTX version is not compatible with the installed firmware.

Solution:
Install the right RTX version or update the firmware. Contact the machine constructor.

6.12.23 16907 Operating system version incompatible with the Main code of the firmware

Cause:
Operating system version of the remote moduleis not compatible with the firmware installed.

Solution:
Install in the remote module the correct operating system version or update the firmware.Contact the machine constructor.

6.12.24 17153 PLUGTYPE: firmware code of GreenBus transmitter not found

Cause:
A firmware file could not be found in FW folder. Normally this depends on an accidental erasure or on an incomplete or damaged installation.

Solution:
Reinstall Albatros after executing a backup of the whole system. Contact the machine constructor.

6.12.25 17154 PLUGTYPE: firmware code of GreenBus transmitter damaged

Cause:
File containing firmware of GreenBus transmitter resides in FW folder but appears to be damaged or incomplete.

Solution:
Reinstall Albatros after executing a backup of the whole system. Contact the machine constructor.

6.12.26 17155 PLUGTYPE: error sending bootstrap code of GreenBus transmitter

Cause:
An error occurred during system initialization. Specifically, a firmware file failed being sent to a module.

Solution:
Try to reset system. If the problem persists, contact T.P.A. S.p.A.

6.12.27 17156 PLUGTYPE: error sending main code to GreenBus transmitter

Cause:
An error occurred during system initialization. Specifically, a firmware file failed being sent to a module.

Solution:
Try to reset system. If the problem persists, contact T.P.A. S.p.A.

6.12.28 17157 PLUGTYPE: bootstrap code not found

Cause:
A firmware file could not be found in FW folder. Normally this depends on an accidental erasure or on an incomplete or damaged installation.

Solution:
Install Albatros again after executing a backup of the whole system. Contact the machine constructor.

6.12.29 17158 PLUGTYPE: main code not found

Cause:
A firmware file could not be found in FW folder. Normally this depends on an accidental erasure or on an incomplete or damaged installation.

Solution:
Reinstall Albatros after executing a backup of the whole system. Contact the machine constructor.

6.12.30 17159 PLUGTYPE: error sending bootstrap code

Cause:
An error occurred during system initialization. Specifically, a firmware file failed being sent to a module.

Solution:
Try to reset system. If the problem persists, contact T.P.A. S.p.A.

6.12.31 17160 PLUGTYPE: error sending main code

Cause:
An error occurred during system initialization. Specifically, a firmware file failed being sent to a module.

Solution:
Try to reset system. If the problem persists, contact T.P.A. S.p.A.

6.12.32 17409 Could not send auxiliary executable ...

Cause:
This error may occur while updating firmware on a remote module. It may be caused by a momentary network failure but also by a damaged firmware on the module. This error message may include an error code.

Solution:
Try to switch off and switch back on the remote module and repeat the update procedure. If the problem persists, contact T.P.A. S.p.A.

6.12.33 17410 Could not run auxiliary executable ...

Cause:
An error occurred during system initialization. Specifically, an auxiliary program could not start execution. The error message also reports the auxiliary program name and possibly an error code.

Solution:
Try to reset system. If the problem persists, contact T.P.A. S.p.A.

6.12.34 17665 Communication library not found ...

Cause:
A Albatros system library has not been found. Normally this is caused by an accidental erasure of the file, or by an incomplete or damaged installation. The error message also reports name of the missing library and possibly an error code.

Solution:
Install Albatros again after executing a backup of the whole system. Contact the machine constructor.

6.12.35 17666 Error using communication library ...

Cause:
A system library returned an unexpected error code. Normally this happens when the current library version is not aligned with the rest of the system.

Solution:
Contact T.P.A. S.p.A.

6.12.36 17667 DLLNAME: firmware code could not start running

Cause:
An error occurred during system initialization. Specifically, firmware code could not start. "*DLLNAME*" corresponds to the software component that caused the error.

Solution:
Try to reset system. If the problem persists, Contact T.P.A. S.p.A.

6.12.37 17668 DLLNAME: could not get pointer to shared RAM

Cause:
An error occurred during system initialization. Specifically, the communication channel with firmware could not be opened. "*DLLNAME*" corresponds to the software component that caused the error.

Solution:
Try to reset system. If the problem persists, Contact T.P.A. S.p.A.

6.12.38 17669 DLLNAME: key 'Bin=' missing in TPA.INI

Cause:
Configuration file of Albatros is incorrect. Specifically, key "Bin" is missing, which identifies folder where system components are stored. This error may also be caused by an accidental erasure. "*DLLNAME*" corresponds to the software component that caused the error.

Solution:
Check that file "TPA.INI" exists and contains a valid "Bin" key. Contact the machine constructor.

6.12.39 17921 Could not send NODETPA ...

Cause:
This error may occur while updating firmware on a remote module. It may be caused by a momentary network failure but also by a damaged firmware on the module. This error message may include an error code.

Solution:
Try to switch off and on the remote module and repeat the update procedure. If the problem persists, contact T.P.A. S.p.A.

6.12.40 17922 NODETPA did not restart ...

Cause:
This error may occur while updating firmware on a remote module. It may be caused by a momentary failure but also by a damaged firmware on the module. This error message may include an error code.

Solution:
Try to reset Clipper module and repeat the update procedure. If the problem persists, contact T.P.A. S.p.A.

6.12.41 17923 NODETPA not running ...

Cause:
A remote module has been detected on the network, whose communication software is not running. Normally this is caused by a failure of communication software. The error message may include an error code.

Solution:
Try to switch off and on the remote the module. If the problem persists, contact T.P.A. S.p.A.

6.12.42 18177 NodeTpa tried to access to an invalid address

Cause:
The remote module communication software raised an error. The error message may include an error code.

Solution:
Try to switch off and on the module. If the problem persists, contact T.P.A. S.p.A.

Tecnologie e Prodotti per l'Automazione s.p.a
Via Carducci 221
I - 20099 Sesto S.Giovanni (MI)
Tel. +39 02.36527550
Fax. +39 02.2481008
www.tpaspa.it