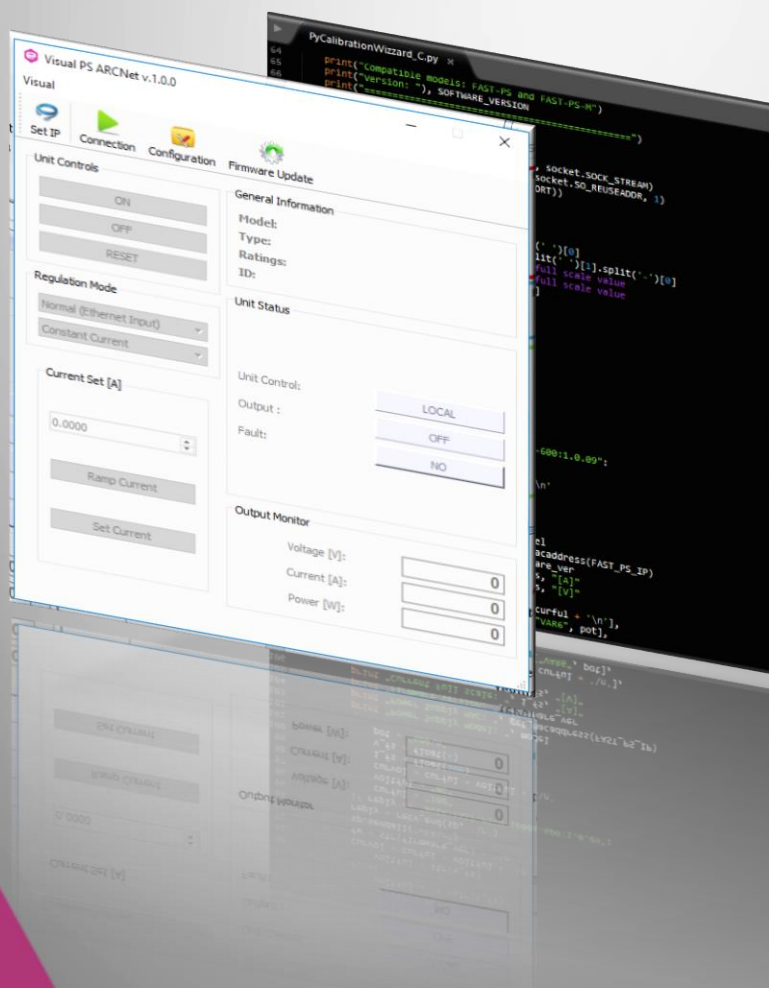


# Power Supply Series



## Remote Control Manual - ANET



MAGNET POWER SUPPLY SYSTEMS

All Rights Reserved  
© CAEN ELS s.r.l.  
Rev. 1.0 – June 2017



**CAEN ELS s.r.l.**  
c/o AREA Science Park  
S.S. 14 km 163,5 – 34149 Basovizza (TS)  
Italy  
Mail: [info@caenels.com](mailto:info@caenels.com)  
Web: [www.caenels.com](http://www.caenels.com)



## **Remote Control Manual – Models**

*This manual covers the following Power Supplies models:*

- **FAST-PS-ANET**



# Table Of Contents

<b>1. OVERVIEW.....</b>	<b>7</b>
<b>2. DEDICATED SOFTWARE.....</b>	<b>8</b>
2.1 CAENELS DEVICE MANAGER.....	8
2.1.1 Searching for connected devices.....	9
2.1.2 Device Configuration.....	9
2.2 CAEN ELS VISUAL PS ARCNET SOFTWARE .....	11
2.2.1 Power Supply IP .....	11
2.2.2 Main windows .....	12
2.2.3 Unit Configuration.....	13
2.2.4 Faults Monitoring and Reset .....	15
2.2.5 Interlocks setting.....	16
2.2.6 Firmware Update.....	17
<b>3. SOFTWARE COMMANDS .....</b>	<b>18</b>
3.1 ETHERNET INTERFACE .....	18
3.2 COMMAND SYNTAX .....	19
3.3 COMMAND REPLIES .....	20
3.4 ERROR TABLE .....	21
3.5 COMMAND TABLE.....	22
3.6 BASIC COMMANDS.....	27
3.6.1 MON Command.....	27
3.6.2 MOFF Command.....	28
3.6.3 VER Command.....	29
3.7 FAULTS MONITORING AND RESET.....	30
3.7.1 MST Command.....	30
3.7.2 FAST-PS-ANET Status Register .....	31
3.7.3 MRESET Command.....	32
3.8 CURRENT AND VOLTAGE SETTING/READING .....	33
3.8.1 UPMODE Command .....	33
3.8.2 MRI Command.....	35
3.8.3 MRV Command.....	36
3.8.4 LOOP Command.....	37
3.8.5 MWI Command.....	38
3.8.6 MWV Command.....	39
3.8.7 MWIR Command.....	40
3.8.8 MSRI Command.....	41
3.8.9 MWVR Command.....	42
3.8.10 MSRV Command.....	43
3.9 GENERIC MONITORING PARAMETERS .....	44
3.9.1 MRT Command.....	44
3.9.2 MRW Command.....	45
3.9.3 MGC Command.....	46
3.9.4 MRID Command.....	47
3.10 CONFIGURATION COMMANDS .....	48
3.10.1 MRG Command .....	50
3.10.2 MWG Command.....	51
3.10.3 Interlock Setting .....	52

3.10.4	PASSWORD Command.....	53
3.10.5	MSAVE Command .....	55



## Document Revisions

Revision	Date	Comment
1.0	June 1 <sup>st</sup> 2017	Official Release



# 1. Overview

In this manual, the user can find all the information related to the dedicated software and programming language.

Chapter 2 regards the dedicated software, while chapter 3 the dedicated programming language. When the power supply is controlled through the dedicated software or through the programming language, possible operations are exactly the same.

The main difference is that the programming language is a ready-to-use high-level (ASCII-based) language which allows the user to perform specific routines for its specific application, so guaranteeing high freedom and flexibility.

## 2. Dedicated Software

The described utilities allow a user-friendly and fast access to the functionalities and configuration of the power module unit.



Two different software packages are available for operation with CAEN ELS power supplies: “*CAENels Device Manager*” and “*Visual*” Software. Both utilities can be downloaded free-of-charge from the CAENels website [www.caenels.com](http://www.caenels.com). An overview of both utilities is given in the next sections.

### 2.1 CAENels Device Manager

The “*CAENels Device manager*” software can be used to detect

- CAEN ELS devices described within this document and connected to the local network;
- Their network configuration;

The “*CAENels Device manager*” is available for Windows and Linux platform. The system requirements are:

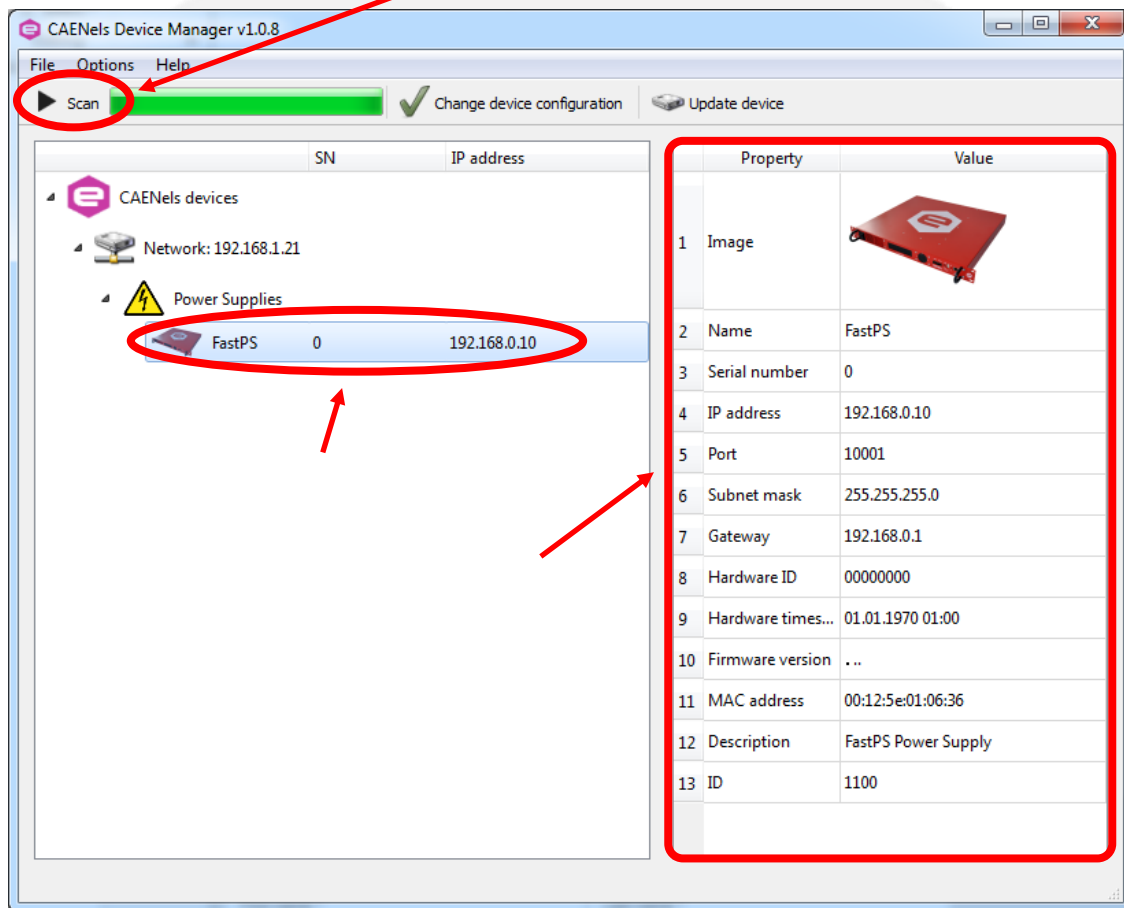
1.  Windows minimum system requirements:
  - Windows® XP or newer
  - Intel® or equivalent processor
  - 70 MB available HD space
  - Ethernet network card
2.  Linux minimum system requirements:
  - Linux kernel 2.2.x or newer
  - Intel® or equivalent processor
  - 70 MB available HD space
  - Ethernet network card



### 2.1.1 Searching for connected devices

The following steps have to be performed in order to carry out a search of all the CAEN ELS units connected to the local network:

- Install the “CAENels Device manager” software;
- Launch the software;
- Perform a scan to discover the connected e.g. FAST-PS device(s) by clicking the “Scan” button as indicated in **Figure 1**. If there are multiple available connections it is possible to select the network/networks to be scanned in the “Selected network interfaces” window available under the “Options” menu. All the information about the selected devices are shown in the right side of the main window.



**Figure 1:** Device Manager - Main interface

Make sure that the firewall is not preventing communication between your computer and the unit(s). The “CAENels Device manager” uses **UDP port 30719** to find the device, so ensure that the UDP traffic is allowed in both directions on this port.

### 2.1.2 Device Configuration

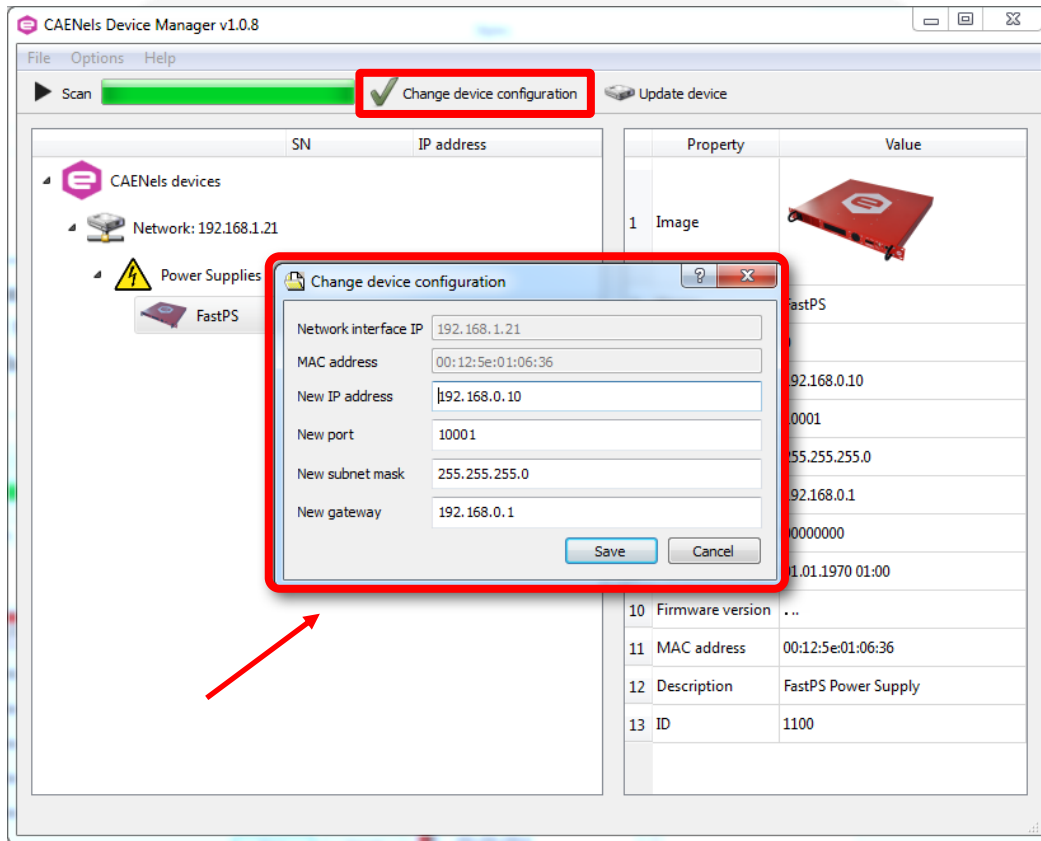


The software allows also to change the Network configuration of the found device(s) in the local network.

In order to change the network configuration of the unit it is necessary to select the desired device and click on the “*Change device configuration*” button in the main window as shown in **Figure 2**. The configurable Network options are:

- Device IP address;
- TCP/IP communication port;
- Subnet mask;
- Gateway.


To apply the changes on the device configuration it is necessary to edit the corresponding fields and then to click on the “*Save*” button. A screenshot of a sample device configuration is shown in the following picture:



**Figure 2:** Device Manager - Change device configuration

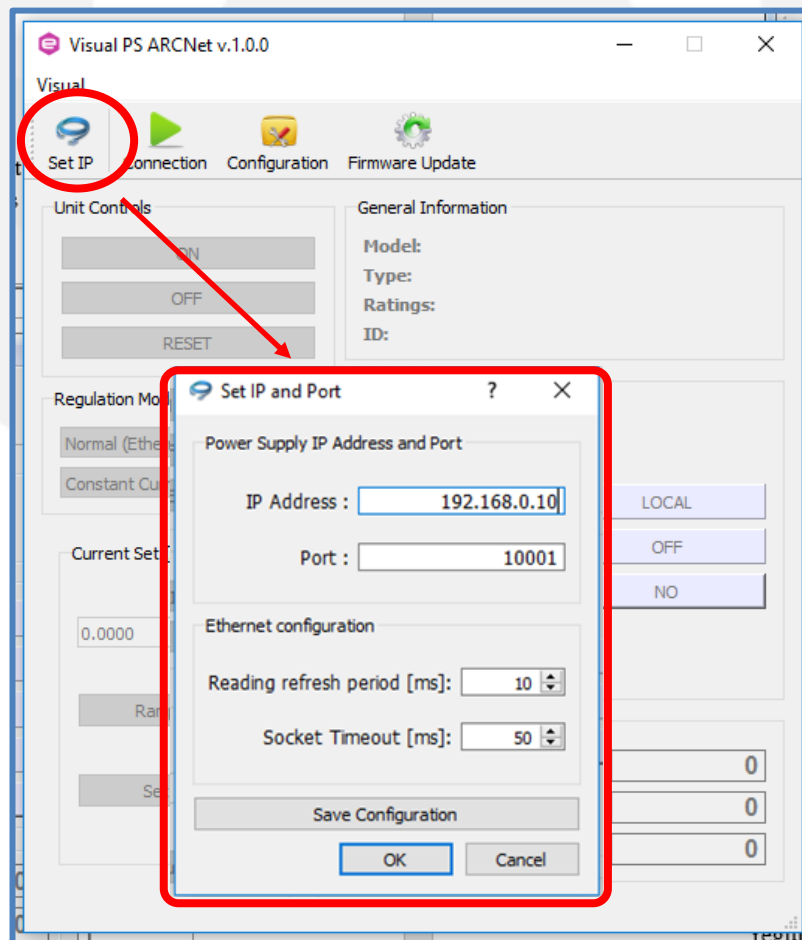
## 2.2 CAEN ELS Visual PS ARCnet Software

The *Visual* software makes it easy to remote control the main features of the CAEN ELS power converters using a Graphic User Interface (GUI). The software is developed using Qt, which is a cross-platform application and UI framework with APIs for C++ programming. The software is available for Windows platform. The system requirements are as follows:

3.  Windows minimum system requirements:
  - Windows® XP or newer
  - Intel® or equivalent processor
  - 30 MB available HD space
  - Ethernet network card

### 2.2.1 Power Supply IP

To communicate with a power unit, it is necessary to set its IP address and its port. This operation can be made by using the module IP address window, accessible by clicking the first “Set IP” icon of the Visual Toolbar.

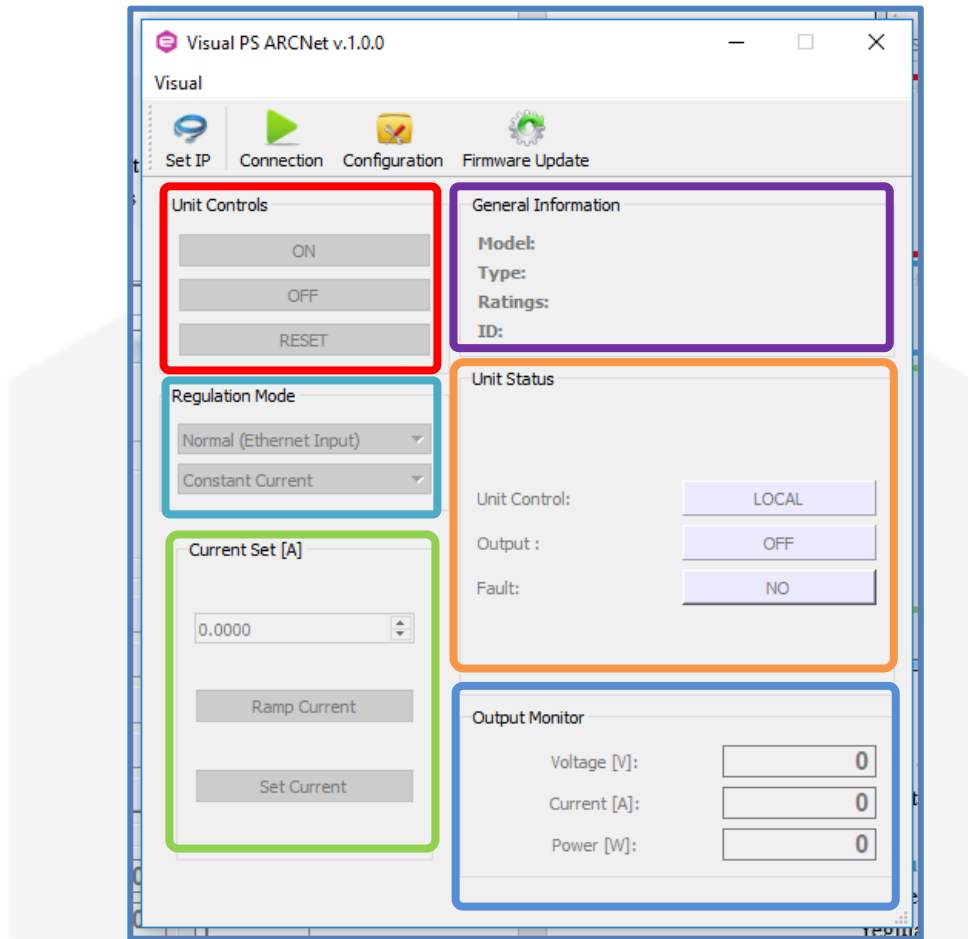


**Figure 3:** Visual PS ARCnet – Set IP

To establish the connection with the module, it is necessary to click on the “Connection” icon. Once the communication is established it is possible to configure the unit and monitor its status.

### 2.2.2 Main windows

The Visual PS ARCnet main window is organized in the following sections:



**Figure 4:** Visual PS ARCnet – Main Window

- **Unit Controls:** allow to Switch ON/OFF the unit and to Reset its status register (in case of a fault).
- **Regulation mode:** allows to select the mode of operation. The available modes are: “Constant Voltage” mode or “Constant Current” mode. The regulation mode can be changed only when the unit is switched OFF.
- **Voltage/Current Set section:** this section allows to set the Voltage (for the “Constant Voltage” mode) or Current (for the “Constant Current” mode) setpoint. To apply a setpoint it is necessary to click on the *Ramp* or *Set* button. The *Ramp* button performs a ramp to the selected setpoint, otherwise the *Set* button applies directly the selected setpoint.

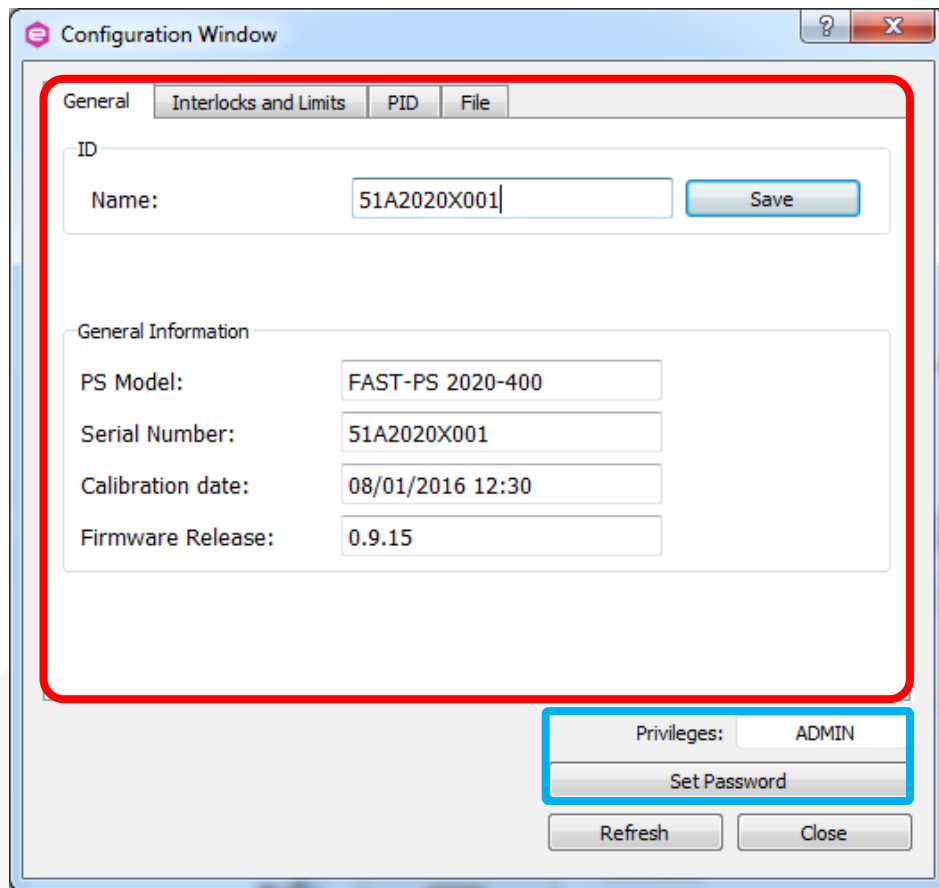
- **General information:** indicates some information regarding the connected unit, like the model and its serial number.
- **Unit status:** indicates some information regarding output status, temperature, current leakage, unit control (local or remote) and fault status. By clicking on the fault status indicator it is possible to visualize detailed fault status windows.
- **Output monitor:** indicates the actual output voltage, current and power.

### 2.2.3 Unit Configuration

To display the configuration Window, click on the Configuration button on the main Toolbar. From this window, it is possible to configure the unit. Several fields are password protected. To have the access to the password protected fields it is necessary to insert the correct password word. For additional information regarding the password protected cells and the memory structure, please refer to section 3.10.

The Configuration Window is divided in the following tabs:

- **General tab:** shows the general information about the unit as the name of the unit, its model, firmware version, serial number and the calibration date.
- **Interlock and Limit tab:** from this tab the user can:
  - display and edit the names, direction, status (enabled or disabled) and intervention time of the available external interlocks,
  - visualize and edit the module limits (which generate a fault condition) as: the maximum allowable temperature, min DC-link voltage, Leakage current limit etc.
- **PID tab:** from this tab it is possible to edit the PID regulators parameters – i.e. proportional, derivative and integral terms.
- **File tab:** in this tab the user can display, edit and save the content of the unit memory.



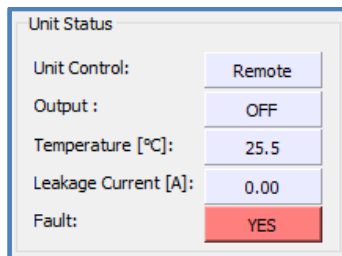
The image shows a 'Configuration Window' with a blue title bar and standard window controls. It features four tabs: 'General' (selected), 'Interlocks and Limits', 'PID', and 'File'. The 'General' tab contains an 'ID' section with a 'Name' field set to '51A2020X001' and a 'Save' button. Below this is a 'General Information' section with four fields: 'PS Model' (FAST-PS 2020-400), 'Serial Number' (51A2020X001), 'Calibration date' (08/01/2016 12:30), and 'Firmware Release' (0.9.15). At the bottom right, there is a 'Privileges' dropdown set to 'ADMIN', a 'Set Password' button, and 'Refresh' and 'Close' buttons. A red rectangle highlights the 'ID' and 'General Information' sections, and a blue rectangle highlights the 'Privileges' and 'Set Password' buttons.

Field	Value
Name	51A2020X001
PS Model	FAST-PS 2020-400
Serial Number	51A2020X001
Calibration date	08/01/2016 12:30
Firmware Release	0.9.15
Privileges	ADMIN

**Figure 5:** Visual – Configuration Window

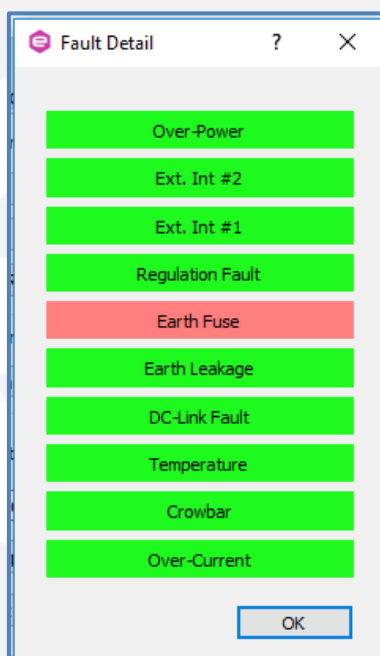
### 2.2.4 Faults Monitoring and Reset

In the case of a fault, the power supply will be turned off and the Unit Status in the Visual PS ARCnet software will show that a fault is present:



**Figure 6:** Visual PS ARCnet - Fault

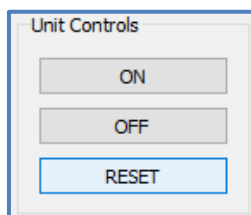
To understand the nature of the fault, please click on the Fault button (red “YES” button in Figure 6); the software will show the fault:



**Figure 7:** Visual PS ARCnet – Fault list

In order to reset the fault and so, to turn the power supply on again:

1. Remove the fault cause (in this case check the earth fuse, eventually replacing it)
2. Click on the RESET button:



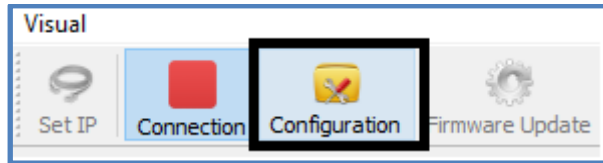
**Figure 8:** Visual PS ARCnet – RESET button

- Now the power supply can be turned on again.

### 2.2.5 Interlocks setting

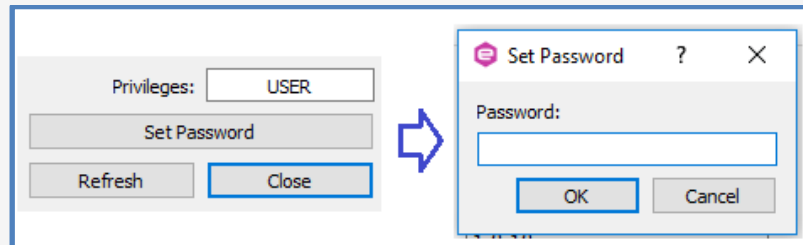
To set interlocks levels please follow this procedure:

- Click on the “Configuration” button:



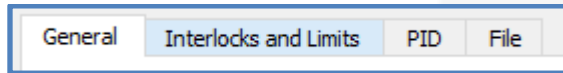
**Figure 9:** Visual PS ARCnet – Configuration button

- If you have “USER” privileges you are not allowed to change interlocks levels, so click on “Set Password” and digit “PS-ADMIN” in the Password field, then press “OK”:



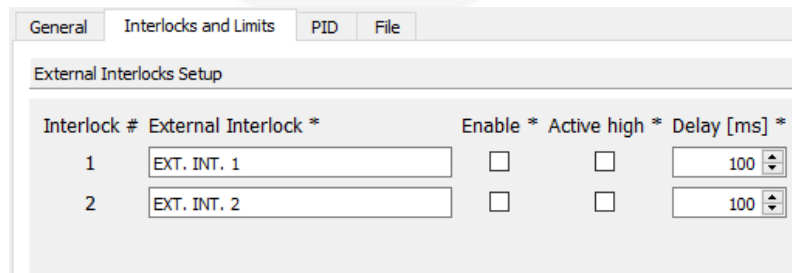
**Figure 10:** Visual PS ARCnet – Set Password

- Now click on “Interlocks and Limits”:



**Figure 11:** Visual – Interlocks and Limits

- It is now possible to enable interlocks (checking the “Enable” box), to set the high or low level and the intervention time (delay). When interlock 1 is enabled at high level, fault will appear when 24V are not applied to it, if it is enabled at low level, fault will appear when 24V are applied to it.



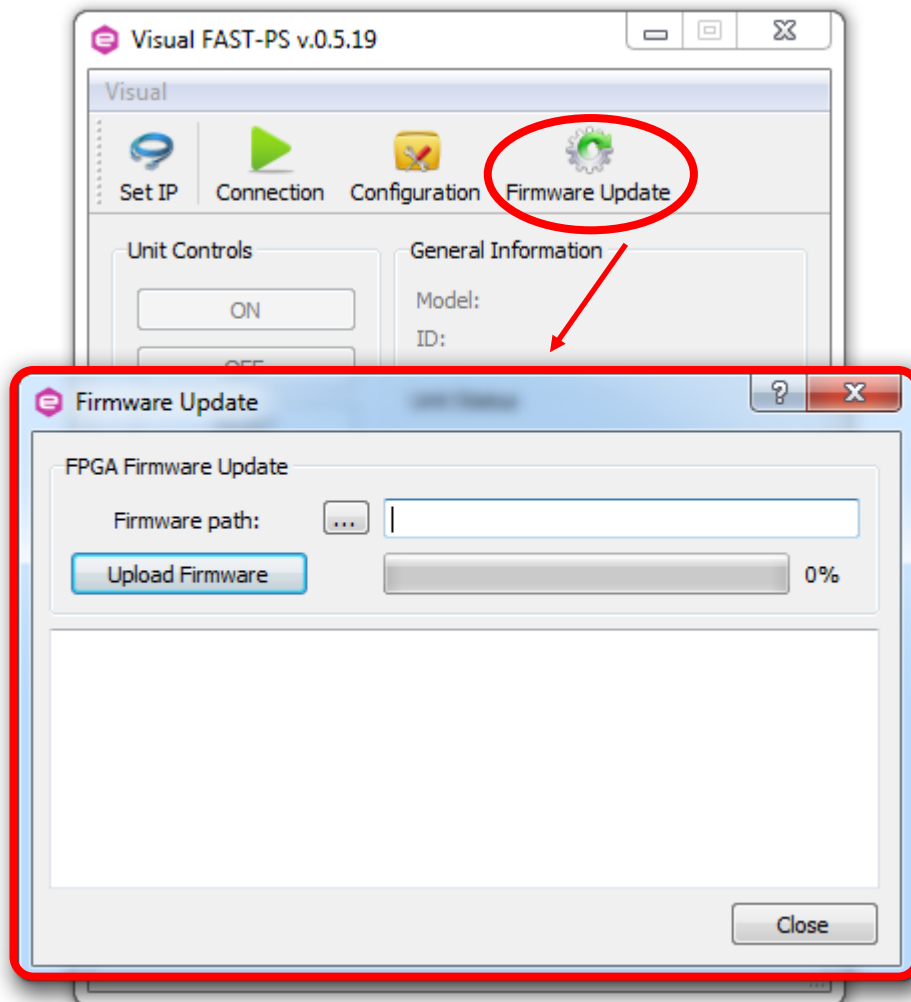
**Figure 12:** Visual PS ARCnet – Interlocks setting

- Reset user privileges by writing a wrong password in the password field and press “OK”.



### 2.2.6 Firmware Update

In order to update the power unit firmware, it is necessary to click on the Firmware Update button on the main toolbar. From the Firmware Update window select the desired firmware file (**.updt** extension) and click on the Upload Firmware button. The update procedure can take several minutes for the firmware update procedure.



**Figure 13:** Visual – Firmware Update

## 3. Software Commands

This chapter describes the base TCP/IP software commands used for the control and configuration of the power module.

### 3.1 Ethernet Interface

The device is shipped with default IP address, subnet mask, and gateway and TCP-IP communication port:

Parameter	Factory Value
IP address	192.168.0.10
Subnet mask	255.255.255.0
Gateway	192.168.0.1
TCP/IP port	10001

**Table 1:** Default Ethernet Settings

## 3.2 Command Syntax

The command syntax used by the protocol is described in the following sections.

Commands must be sent in ASCII format and are composed by a “*command field*” and one, two or none “*parameter field*”, separated by a colon (“:” or “0x3A” in hexadecimal notation). The number of “*parameter fields*” depends on the specific command. Commands are **NOT case sensitive** and therefore the command string can be sent either using uppercase or lowercase characters (conversion to uppercase characters is performed internally). Each command must be terminated with the termination sequence. The protocol supports two termination sequences:

- “*carriage return*” termination char “**|r**” (“0x0D” in hexadecimal notation or commonly CR) or
- “*carriage return\line feed*” sequence “**|r|n**” (“0x0D 0x0A” in hexadecimal notation or commonly CRLF).

### Command Example:

**MWI:20.5580|r** or **MWI:20.5580|r|n**

- “**MWI**” is the command field;
- “**:**” is the parameter’s separation character;
- “**20.5580**” is the first parameter field;
- “**|r**” or “**|r|n**” are the termination sequences of the command.

In the following command description the “**|r**” termination char is used, but it can be always replaced with the termination sequence “**|r|n**”.

Commands are processed one at a time; therefore, **user must wait for a response from the unit before sending the next command.**

All the responses from the FAST-PS-ANET module are in upper case and are terminated with the same “*carriage return\line feed*” sequence (“**|r|n**”, “0x0D 0x0A” in hexadecimal notation or commonly CRLF).

**MWI:10.5875|r|n** →

or:

**MWI:10.5875|r** →

← **#ACK|r|n**

← **#ACK|r|n**

### 3.3 Command Replies

The reply from the module depends on the given command. In general, the command can be grouped in two categories: Write commands and Read commands.

For **write commands** there are two specific replies that indicate that the command has been correctly elaborated or not. Those replies are hereafter presented:

- **AcKnowledge** (“**#AK**”) indicates that the command is valid and it was correctly elaborated by the device:

**#AK|r|n**

- “**#AK**” is the **AcKnowledge** response to a valid command;
  - “**|r|n**” is the termination sequence of the reply.
- **Not AcKnowledge** (“**#NAK**”) indicates that the command is either not valid or that it was not accepted by the device; the “**NAK**” reply is followed by an “*error code*” field, which can be used to determine the cause of the error (see the List of the Error Codes appendix, section 3.4, for a detailed list of all possible error codes):

**#NAK:01|r|n**

- “**#NAK**” is the **Not AcKnowledge** response to an invalid command;
- “**:**” is the parameter’s separation character;
- “**01**” is the error code,
- “**|r|n**” is the termination sequence of the reply.

For **read commands**, the replies are generally formed by an echo string, followed by the corresponding read value. The echo string is preceded by the hash character (“#”) and the echo is separated from the “**:**” separation character.

Some examples are hereafter shown:

**MRI|r** →

← **#MRI:12.8875|r|n**

or:

**MWL:~|r** →

← **#MWL:10.9850|r|n**

or:

**MRG:90|r** →

← **#MRG:90:0x2|r|n**

- the read commands are highlighted in **blue**;
- the echo string is highlighted in **green**;
- the read value is in **purple**;
- the termination char is highlighted in **red**.

For more detailed information about the single command please refer to the specific command section.

### 3.4 Error Table

The list of error codes returned with the **#NAK** reply and their description are hereafter shown:

Error Code #	Description
01	Unknown command
02	Unknown Parameter
03	Index out of range
04	Not Enough Arguments
05	Privilege Level Requirement not met
06	Saving Error on device
07	Invalid password
08	Power supply in fault
09	Power supply already ON
10	Setpoint is out of model limits
11	Setpoint is out of software limits
12	Setpoint is not a number
13	Module is OFF
14	Slew Rate out of limits
15	Device is set in local mode
16	Module is not in waveform mode
17	Module is in waveform mode
18	Device is set in remote mode
19	Module is already in the selected loop mode
20	Module is not in the selected loop mode
99	Unknown error

**Table 2:** NAK Error code table

## 3.5 Command Table

The list of commands used within the communication protocol and the corresponding syntax is hereafter presented as well as a description of each command purpose and any special requirements related to the specific command. The base commands are summarized in Table 3.



Command	Read/ Write	Parameter #1	Parameter #2	Detailed description	Reply value
<b>VER</b>	R	/	/	Return the module model and installed firmware versions	ASCII indicating the module model and firmware version
<b>MON</b>	W	/	/	Turn on the module	“AK” or “NAK”
<b>MOFF</b>	W	/	/	Turn the module OFF	“AK”
<b>LOOP</b>	W R	“I” or “V” “?”	/ /	Set the power module loop mode Query for the power supply loop mode	“AK” or “NAK” Loop mode (“I” or “V”)
<b>UPMODE</b>	W R	Update mode “?”	/ /	Set the new update mode (ASCII) Query for the current update mode	“AK” or “NAK” ASCII indicating the current setpoint
<b>SETFLOAT</b>	W R	“F” or “N” “?”	/ /	Set if the output is floating or not	“AK” or “NAK” Float mode if “F” or not “N”
<b>MST</b>	R	/	/	Read module internal status register	Internal status register (Hex representation)
<b>MRESET</b>	W	/	/	Reset the module status register	“AK” or “NAK”
<b>MRI</b>	R	/	/	Read output current value	ASCII indicating the output read current
<b>MRV</b>	R	/	/	Read output voltage value	ASCII indicating the output read voltage
<b>MRW</b>	R	/	/	Read output power	ASCII indicating the output read power

Command	Read/ Write	Parameter #1	Parameter #2	Detailed description	Reply value
<b>MRIA</b>	R	/	/	Read instantaneous output current value	ASCII indicating the output read current
<b>MRVA</b>	R	/	/	Read instantaneous output voltage value	ASCII indicating the output read voltage
<b>MRWA</b>	R	/	/	Read instantaneous output power	ASCII indicating the output read power
<b>MRIO</b>	R	/	/	Read output current offset	ASCII indicating the output read current
<b>MRVO</b>	R	/	/	Read output voltage offset	ASCII indicating the output read voltage
<b>MRT</b>	R	/	/	Read MOSFET Heatsink Temperature [°C]	ASCII indicating the temperature value
<b>MRP</b>	R	/	/	Read DC-Link Voltage	ASCII indicating the DC-Link voltage
<b>MGC</b>	R	/	/	Read Earth Leakage current	ASCII indicating the Earth leakage current
<b>MWV</b>	W R	V Setpoint “?”	/	Set the new voltage setpoint (ASCII) Query for the last applied setpoint	“AK” or “NAK” ASCII indicating the voltage setpoint
<b>MWVR</b>	W R	V Setpoint “?”	/	Go to the given setpoint with a <u>ramp</u> (ASCII) Query for the last accepted final ramp setpoint	“AK” or “NAK” ASCII indicating the voltage setpoint
<b>MWI</b>	W R	I Setpoint “?”	/	Set the new current setpoint (ASCII) Query for the last applied current setpoint	“AK” or “NAK” ASCII indicating the current setpoint



Command	Read/ Write	Parameter #1	Parameter #2	Detailed description	Reply value
<b>MWIR</b>	W R	I Setpoint “?”	/ /	Go to the given setpoint with a <u>ramp</u> (ASCII) Query for the last accepted final ramp setpoint	“AK” or “NAK” ASCII indicating the current setpoint
<b>MSRI</b>	W R	I Ramp Slew rate “?”	/ /	Set the I ramp slew rate [A/s] (ASCII) Query for the I ramp slew-rate	“AK” or “NAK” ASCII indicating the I ramp slew-rate
<b>MSRV</b>	W R	I Ramp Slew rate “?”	/ /	Set the I ramp slew rate [V/s] (ASCII) Query for the I ramp slew-rate	“AK” or “NAK” ASCII indicating the I ramp slew-rate
<b>MPLST</b>	R	/	/	Read Current, Voltage and Status simultaneously (mean values)	ASCII indicating the outputs readings and internal status register (Hex representation)
<b>MPLSTA</b>	R	/	/	Read Current, Voltage and Status simultaneously (instantaneous value)	ASCII indicating the outputs readings and internal status register (Hex representation)
<b>MRW</b>	R	/	/	Read estimated active output power value [W]	ASCII indicating the active output power value
<b>MRID</b>	R	/	/	Read module identification	Module identification (ASCII)
<b>HELP</b>	R	/	/	Print this command List	Command List
<b>HWRESET</b>	W			Reboot power supply software	“AK” or “NAK”
<b>PASSWORD</b>	W R	Password word “?”	/	Set the password word (ASCII) Query for the actual user privileges	“AK” or “NAK” User privileges (ASCII representation)
<b>MRG</b>	R	Parameter field #		Read the given parameter field	Field content (ASCII)
<b>MWG</b>	W	Parameter field #	Cell content (ASCII)	Write to the given parameter field	“AK” or “NAK”

Command	Read/ Write	Parameter #1	Parameter #2	Detailed description	Reply value
<b>MSAVE</b>	W	/	/	Save the used parameter in the non-volatile memory	“AK” or “NAK”

**Table 3:** Commands overview table

## 3.6 Basic Commands

In the following section are described the basic commands that allows to control the power unit and to monitor its status.

### 3.6.1 MON Command

The **MON** (Module ON) command is intended to turn ON the module output driver, thus enabling the output current terminals and allowing the power supply to regulate and feed current or voltage to the connected load.

After the reception of an “MON” command, the power supply automatically sets output current to 0A or 0V (depending if the module is set in constant current or constant voltage mode).

Replies from the power converter to a **MON** command are in the form “#AK|r|n” – when the command is correctly executed - or “#NAK:xx|r|n”, when the command cannot be executed, with “xx” indicating the error code. The complete list of the error codes is shown in the

Error Table. Sending a **MON** command when the module output is already enabled generates a non-acknowledgment response.

#### Examples:

*MON command example:*

MON|r → #AK|r|n

*MON command example when the module is already enabled (09 code):*

MON|r → #NAK:09|r|n

### 3.6.2 MOFF Command

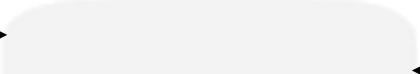
The **MOFF** (Module OFF) command is intended to turn OFF the module output driver, thus disabling the output terminals.

The **MOFF** command automatically sets output current to 0A or 0V with a ramp before disabling the output drivers. This is done in order to avoid output overshoots (especially in constant current regulation mode). The slew-rate of the ramp is factory defined.

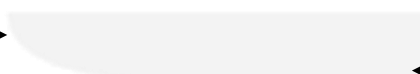
Replies from the unit to a **MON** command are in the form “**#AK|r|n**” – when the command is correctly executed - or “**#NAK:xx|r|n**”, when the command cannot be executed, with “**xx**” indicating the error code.

#### Examples:

*MOFF command example:*

**MOFF|r** →  ← **#AK|r|n**

*MOFF command example when the module is in local mode:*

**MOFF|r** →  ← **#NAK:15|r|n**

### 3.6.3 VER Command

The **VER** command returns the information regarding the model and the current installed firmware version.

The response to the **VER** command is in the following format:

**#VER:ps\_model:fw\_version|r|n**

where “**#VER**” is the echo string, “**ps\_model**” is the power converter model and “**fw\_version**” is the current firmware version. The echo, model and firmware information are separated by “:” character and the string is terminated with the standard “**|r|n**” character sequence.

**Example:**

*VER command example:*

**VER|r** → **#VER:FAST-PS 2020-400:0.9.01|r|n**

## 3.7 Faults Monitoring and Reset

This section describes how to behave in the case of a fault (as described in the “User Manual”), and so how to reset the power supply (section 0), and how to monitor the fault presence (section 3.7.1).

### 3.7.1 MST Command

The **MST** command returns the value of the power supply internal status. The response to the **MST** command is in the following format:

**#MST:status\_reg|r|n**

where “**#MST**” is the echo string and “**status\_reg**” is the hexadecimal representation of the internal status register. The internal status register has 32 bits and so its representation is composed by 8 hexadecimal values.

The **MST** command, being a reading command, returns a response in any module condition.

**Example:**

*MST command example:*

**MST|r** → **#MST: 08000002|r|n**

In this example the value 08000002 (hex) has the binary representation:

**0000 1000 0000 0000 0000 0000 0000 0010**

Which means that the 2<sup>th</sup> bit and the 27<sup>th</sup> bit are set. One can refer to the status register (next pages) to see to what these bits are referred to. In the case of a FAST-PS this situation is referred to a fault condition (2<sup>nd</sup> bit), in particular due to the 2<sup>nd</sup> interlock (27<sup>th</sup> bit).

### 3.7.2 FAST-PS-ANET Status Register

The following table shows the FAST-PS internal status register structure:

Bit #	Bit name	Description
#31	/	reserved
#30	/	reserved
#29	OVP	Over Power condition
#28	/	reserved
#27	Ext. Interlock #2	External interlock 2 has tripped
#26	Ext. Interlock #1	External interlock 1 has tripped
#25	Excessive Ripple	Module is having excessive ripple
#24	Regulation Fault	Modules has experienced a regulation fault
#23	Earth Fuse	Earth fuse is blown
#22	Earth Leakage	Earth current leakage fault
#21	DC-Link Fault	DC-Link under-voltage condition
#20	OVT	Over Temperature condition
#19	/	reserved
#18	Crowbar	Crowbar protection intervention
#17	Input OVC	Input Over Current
#16	/	reserved
#15	/	reserved
#14	/	reserved
#13	Waveform	Waveform is in execution
#12	Ramping	Module is ramping current or voltage
#11	/	reserved
#10	/	reserved
#9	/	reserved
#8	/	reserved
#7 - #6	Update mode [2 bits]	Normal [00], Analog input [11]
#5	Regulation mode	C.C. [0] or C.V. [1] output regulation mode
#4	/	reserved
#3 - #2	Control Mode [2 bits]	Indicates the mode of operation of the unit (Remote [00], Local [01])
#1	Fault condition	This bit is set if the module has experienced a fault condition
#0	ON/OFF	This bit is set when the module is enabled and correctly regulating the output

**Table 4:** FAST-PS Status Register structure

Before sending the **MRESET** command to the power supply it is important to remove the fault presence, or the power supply will instantly get a new fault issue. After an **MRESET** command with reply in the form “#AK|r|n”, the power supply can be turned on again.

**Error Table, section 3.4).**

*MRESET command example:*

$\#AK|r|n$

← #NAK:15|r|n



## 3.8 Current and Voltage Setting/Reading

In this section, specific commands to set or to read the value of the current and/or voltage are discussed.

### 3.8.1 UPMODE Command

First of all, in order to update the setpoint through remote control, the operation mode needs to be in **NORMAL** mode. **UPMODE** command can be used in order to select the update mode of the set-point. There are two possible mode of operation:

- **NORMAL** – in this mode of operation the power unit works in the standard update mode. The direct set-point or ramp commands are received using the standard Ethernet communication and they are applied immediately when the command is received. The analog input signal is ignored.
- **ANALOG** – in this mode of operation the power unit receives the set-point from its analog input. The analog inputs goes from -10 V up to +10 V. The setting command from the local or remote interface are not accepted.

Note: this update mode is available only on units that have the analog input. This option has to be communicated at the order of the unit.

To set the update mode of operation the following commands has to be used:

**UPMODE:mode|r|n**

where “mode” is a string indicating the mode of operation:

- “**NORMAL**” for normal update mode,
- “**ANALOG**” for analog input update mode.

Replies from the unit to a **UPMODE** set are in the form “#AK|r|n” – when the command is correctly executed - or “#NAK:xx|r|n”, when the command cannot be executed (“xx” is the error code). The analog update mode of operation can be set only when the module is turned OFF.

To read the current used loop mode of operation the query command: “**UPMODE:?**” has to be used. The response to the “**UPMODE:?**” query command is in the following format:

**#UPMODE:mode |r|n**

where “#UPMODE” is the echo string, “mode” is a single character indicating the loop mode (“normal” for normal update mode and “analog” for analog input update mode).

The update mode is also visible in the status register (bits #7, #6). For further information, please see the MST command

**Examples:**

*UPMODE example to set the update mode to analog:*

UPMODE:ANALOG|r→

←#AK|r|n

*UPMODE query example when the module is in normal mode:*

UPMODE:?|r→

←#UPDMODE: NORMAL|r|n

### 3.8.2 MRI Command

The **MRI** command returns the readback value of the power supply actual output current.

The readback current value is represented with 6-digit precision. Replies from the power supply to this command are in the following format:

**#MRI:current\_value|r|n**

where “#MRI” is the echo string, “current\_value” is the output current value readback in Ampere [A].

**Example:**

*MRI command example:*

**MRI|r** → **#MRI:22.123456|r|n**

### 3.8.3 MRV Command

The **MRV** command returns the readback value of the power supply actual output voltage.

The voltage readback value is represented with 6-digit precision. Replies from the power supply to this command are in the following format:

**#MRV:voltage\_value|r|n**

where “**#MRV**” is the echo string, “**voltage\_value**” is the output voltage value readback in Volts [V].

**Example:**

*MRV command example:*

**MRV|r** → **#MRV:10.123456|r|n**

### 3.8.4 LOOP Command

The **LOOP** command can be used in order to select the mode of loop control of the FAST-PS unit. There are two possible modes of operation:

- Constant Current (c.c.),
- Constant Voltage (c.v.).

To set the mode of operation the following commands has to be used:

**LOOP:mode|r|n**

where “**mode**” is a single char indicating the mode of operation:

- “**I**” for Constant Current (c.c.) mode and
- “**V**” for Constant Voltage (c.c.) mode.

Replies from the unit to a **LOOP** set are in the form “**#AK|r|n**” – when the command is correctly executed - or “**#NAK:xx|r|n**”, when the command cannot be executed (“**xx**” is the error code). The two modes of operation can be changed only when the module is turned OFF.

To read the current used loop mode of operation the query command: “**LOOP:?**” has to be used. The response to the “**LOOP:?**” query command is in the following format:

**#LOOP:mode |r|n**

where “**#LOOP**” is the echo string, “**mode**” is a single character indicating the loop mode (“**I**” for constant current mode and “**V**” for constant voltage mode).

#### Examples:

*LOOP set example to set the constant current mode:*

**LOOP:I|r** → **#AK|r|n**

*LOOP set example when the module is ON:*

**LOOP:V|r** → **#NAK:09|r|n**

*LOOP query example when the module is in constant voltage (c.v.) mode:*

**LOOP:~|r** → **#LOOP:V|r|n**

### 3.8.5 MWI Command

The **MWI** command can be used to set the output current value when the module is in the constant current mode (see **LOOP** Command). This command is usually needed when running feedback-related applications and for small changes in the output current.

The use of this command is alternative to the

MWIR Command (ramping current command), which is advised for regular use.

This command has the following format:

**MWI:current\_setpoint|r|n**

where “**current\_setpoint**” is the desired current set-point expressed in Ampere [A].

Replies from the module to a **MWI** set are in the form “**#AK|r|n**” – when the command is correctly executed - or “**#NAK:xx|r|n**”, when the command cannot be executed (“**xx**” is the error code).

To read last applied current setpoint the query command: “**MWI:?**” has to be used. The response to this query command is in the following format:

**#MWI:current\_setpoint|r|n**

where “**#MWI**” is the echo string, “**current\_setpoint**” is the last applied current setpoint expressed in Ampere [A].

#### Examples:

*MWI set example, with current setpoint +1.52 A:*

**MWI:1.52|r** → **#AK|r|n**

*MWI set example when the module is OFF:*

**MWI:1.52|r** → **#NAK:13|r|n**

*MWI query example:*

**MWI:?|r** → **#MWI:1.52|r|n**

### 3.8.6 MWV Command

The **MWV** command can be used to set the output voltage value when the constant voltage mode is used (see **LOOP** Command). The use of this command is alternative to the

MWVR Command (ramping voltage command).

This command has the following format:

**MWV:voltage\_setpoint|r|n**

where “**voltage\_setpoint**” is the desired voltage set-point expressed in Volts [V].

Replies from the unit to a **MWV** set are in the form “**#AK|r|n**” – when the command is correctly executed - or “**#NAK:xx|r|n**”, when the command cannot be executed (“**xx**” is the error code).

To read last applied voltage setpoint the query command: “**MWV:?**” has to be used. The response to this query command is in the following format:

**#MWV:voltage\_setpoint |r|n**

where “**#MWV**” is the echo string, “**voltage\_setpoint**” is the last applied voltage setpoint expressed in Volts [V].

#### Examples:

*MWV set example, with voltage setpoint +10.525 V:*

**MWV:10.525|r** → **#AK|r|n**

*MWV set example when the module is OFF:*

**MWV:10.525|r** → **#NAK:13|r|n**

*MWV query example:*

**MWV:?** → **#MWV:10.525|r|n**

### 3.8.7 MWIR Command

The **MWIR** command can be used to perform a ramp to the given current setpoint. This command can be used when the constant current mode is selected (see **LOOP** Command).

The use of this command is alternative to the **MWI** Command. The difference between the **MWI** command and the **MWIR** command is that the first one generates a direct change in output current characterized by the PID regulator parameters (the command is ideally suited for small output current changes and feedback purposes) while the second one makes the power supply go from the previous to the actual current value performing a ramp, defined by a slew-rate in [A/s].

The default value of the slew-rate is stored in the parameter table and it can be read and modified using the **MSRI** command (section 3.8.8).

To dynamically change the current slew-rate value it is possible using the **MSRI** Command. This command has the following format:

**MWIR:final\_ramp\_setpoint|r|n**

where “**final\_ramp\_setpoint**” is the final current value expressed in Ampere [A] to which the power unit will ramp with the defined slew-rate.

Replies from the power unit to a **MWIR** set are in the form “**#AK|r|n**” – when the command is correctly executed - or “**#NAK:xx|r|n**”, when the command cannot be executed (“**xx**” is the error code).

To read the selected final ramp setpoint, the query command: “**MWIR:?**” has to be used. The response to this query command is in the following format:

**#MWIR:final\_ramp\_setpoint|r|n**

where “**#MWIR**” is the echo string and “**final\_ramp\_setpoint**” is the final ramp setpoint expressed in Ampere [A].

#### Examples:

*MWIR set example, with final ramp setpoint +10.5 A:*

**MWIR:10.5|r** →

← **#AK|r|n**

*MWIR set example when the module is OFF:*

**MWIR:10.5|r** →

← **#NAK:13|r|n**

*MWIR query example:*

**MWIR:?** →

← **#MWIR:10.5|r|n**



### 3.8.8 MSRI Command

The **MSRI** command can be used to dynamically change the value of the current ramp slew-rate. The default slew-rate, used at start-up of the unit, is the value stored in the parameters table.

This command has the following format:

**MSRI:***slew\_rate***|r|n**

where “*slew\_rate*” is slew-rate for the current ramp expressed in Ampere per second [A/s].

Replies from the power converter to a **MSRI** set are in the form “**#AK**|r|n” – when the command is correctly executed - or “**#NAK:xx**|r|n”, when the command cannot be executed (“xx” is the error code).

To read the current used slew-rate for the current ramp, the query command: “**MSRI:?**” has to be used. The response to this query command is in the following format:

**#MSRI:***slew\_rate***|r|n**

where “**#MSRI**” is the echo string and “*slew\_rate*” is the slew-rate value used for the current ramp expressed in Ampere per second [A/s].

#### Examples:

*MSRI example, to set the current slew-rate to 10 A/s:*

**MSRI:10**|r|n → ← **#AK**|r|n

*MSRI set example when the unit is in local mode:*

**MSRI:10**|r|n → ← **#NAK:15**|r|n

*MSRI query example:*

**MSRI:?**|r|n → ← **#MSRI:10**|r|n

### 3.8.9 MWVR Command

The **MWVR** command can be used to perform a ramp to the given voltage setpoint. This command can be used, when the constant voltage mode is selected (see **LOOP** Command).

The use of this command is alternative to the **MWV** Command. The difference between the **MWV** command and the **MWVR** command is that the first one generates a direct change in output voltage characterized by the PID regulator parameters while the second one makes the power supply go from the previous to the actual current value performing a ramp, defined by a slew-rate in [V/s].

To dynamically change the slew-rate value it is possible to use the **MSRV** Command (section 3.8.10).

This command has the following format:

**MWVR:final\_ramp\_setpoint|r|n**

where “**final\_ramp\_setpoint**” is the final voltage value expressed in Volts [V] to which the power unit will ramp with the defined slew-rate.

Replies from the FAST-PS to a **MWVR** set are in the form “**#AK|r|n**” – when the command is correctly executed - or “**#NAK:xx|r|n**”, when the command cannot be executed (“**xx**” is the error code).

To read the selected final ramp setpoint, the query command: “**MWVR:?**” has to be used. The response to this query command is in the following format:

**#MWVR:final\_ramp\_setpoint|r|n**

where “**#MWVR**” is the echo string and “**final\_ramp\_setpoint**” is the final ramp setpoint expressed in Volts [V].

#### Examples:

*MWVR set example, with final ramp setpoint +15.2 A:*

**MWVR:15.2|r** → **#AK|r|n**

*MWVR set example when the module is OFF:*

**MWVR:15.2|r** → **#NAK:13|r|n**

*MWVR query example:*

**MWVR:?** → **#MWVR:15.2|r|n**

### 3.8.10 MSRV Command

The **MSRV** command can be used to dynamically change the value of the voltage ramp slew-rate. The default slew-rate, used at start-up of the unit, is the value stored in the parameters table.

This command has the following format:

**MSRV:slew\_rate|r|n**

where “**slew\_rate**” is slew-rate for the voltage ramp expressed in Volts per second [V/s].

Replies from the unit to a **MSRV** set are in the form “**#AK|r|n**” – when the command is correctly executed - or “**#NAK:xx|r|n**”, when the command cannot be executed (“**xx**” is the error code).

To read the current used slew-rate for the voltage ramp, the query command: “**MSRV:?**” has to be used. The response to this query command is in the following format:

**#MSRV:slew\_rate|r|n**

where “**#MSRV**” is the echo string and “**slew\_rate**” is the slew-rate value used for the voltage ramp expressed in Volts per second [V/s].

#### Examples:

*MSRV example, to set the current slew-rate to 10 V/s:*

**MSRV:10|r** →

← **#AK|r|n**

*MSRV set example when the unit is in local mode:*

**MSRV:10|r** →

← **#NAK:15|r|n**

*MSRV query example:*

**MSRV:?:r** →

← **#MSRV:10|r|n**

## 3.9 Generic Monitoring Parameters

This section describes some generic parameters that can be monitored through dedicated instructions (e.g. MOSFET temperature, estimated active power applied to the connected load and so on).

### 3.9.1 MRT Command

The **MRT** command returns the value of the temperature directly measured on the output stage MOSFET heatsink.

The response to the **MRT** command is in the following format:

**#MRT:temperature|r|n**

where “**#MRT**” is the echo string and “**temperature**” is the temperature value expressed in Celsius [°C]. The **MRT** command, being a reading command, returns a response in any module condition.

**Example:**

*MRT command example:*

**MRT|r** →

← **#MRT:37.4|r|n**

### 3.9.2 MRW Command

The **MRW** command returns the actual value of the estimated active power applied to the connected load.

The response to the **MRW** command is in the following format:

**#MRW:active\_power|r|n**

where “**#MRW**” is the echo string and “**active\_power**” is the output active power readback expressed in Watts [W], estimated as the product of the output voltage and output current readbacks. The **MRW** command, being a reading command, returns a response in any module condition.

*Example:*

*MRW command example:*

**MRW|r** → **#MRW:100.4542|r|n**

### 3.9.3 MGC Command

The **MGC** command returns the readback value of the actual leakage current of the unit.

The response to the **MGC** command is in the following format:

**#MGC:***leakage\_current***|r|n**

where “**#MGC**” is the echo string and “*leakage\_current*” is the earth leakage current, expressed in Ampere [A].

When a leakage fault condition is tripped, the **MGC** command will return the value of the max leakage current that tripped the fault. To return to the normal behavior of the read command, the module status has to be reset (see MRESET command).

The **MGC** command, being a reading command, returns a response in any module condition.

**Example:**

*MGC command example for a 60mA ground leakage current:*

**MGC|r** →

← **#MGC:0.06|r|n**

### 3.9.4 MRID Command

The **MRID** command returns the module identification name string. This description is useful in case that there are numerous units installed and it is possible to give a description for each unit (for example the name of the load on which the unit is connected). This information is also displayed on the local display.

The response to the **MRID** command is in the following format:

**#MRID:***fast\_ps\_identification***|r|n**

where “**#MRID**” is the echo string and “*fast\_ps\_identification*” is the module identification string. The identification string is stored in the parameters table and so it is possible to change it using the MWG command.

**Example:**

*MRID example with the module identification “SkewMag1.3”:*

**MRID****|r** → **#MRID:SKEWMAG1.3|r|n**

### 3.10 Configuration Commands

In the following section are described the software commands that allow to read, set and store the working parameters of the power supply unit. **MRG** Command and **MWG** Command allow to read or modify the working parameters. The write-access to several parameters is password protected and certain parameters are read only and so it is not possible to modify them. To change the password privileges, use the **PASSWORD** Command. In order to save the parameter on the on-board non-volatile memory, the **MSAVE** Command has to be used.

The complete list of the configuration parameters, their field index and the access privileges are hereafter shown:

Parameter #	Access Privileges	Parameter Name
#0	Read Only	Firmware ID
#1	Read Only	PS Model
#2	Read Only	Serial Number
#3	Read Only	MAC Ethernet
#4	Read Only	MAC SFP #1
#5	Read Only	MAC SFP #2
#6 - #8	/	Reserved
#9	Read Only	Calibration date
#10	Read Only	Current Calibration Parameter a
#11	Read Only	Current Calibration Parameter b
#12	Read Only	Current Calibration Parameter c
#13	Read Only	Current Calibration Parameter d
#14	Read Only	Voltage Calibration Parameter a
#15	Read Only	Voltage Calibration Parameter b
#16	Read Only	Voltage Calibration Parameter c
#17	Read Only	Voltage Calibration Parameter d
#18	Read Only	DC Link Calibration Parameter a
#19	Read Only	DC Link Calibration Parameter b
#20	Read Only	AC Link Calibration Parameter a
#21	Read Only	AC Link Calibration Parameter b
#22	Read Only	Current Leakage Calibration Parameter a
#23	Read Only	Current Leakage Calibration Parameter b
#24	Read Only	Analog Input Calibration Parameter a
#25	Read Only	Analog Input Calibration Parameter b
#26	Read Only	Analog Input Calibration Parameter c
#27	Read Only	Analog Input Calibration Parameter d
#28 - #29	/	Reserved
#30	User	Module Identification
#31	User	Default Current Slew Rate [A/s]
#32	User	Default Voltage Slew Rate V [V/s]



Parameter #	Access Privileges	Parameter Name
#33 - #39	/	Reserved
#40	User	PID I: Kp_v
#41	User	PID I: Ki_v
#42	User	PID I: Kd_v
#43	User	PID I: Kp_i
#44	User	PID I: Ki_i
#45	User	PID I: Kd_i
#46	User	PID I: Upper Limit Acc_v
#47	User	PID I: Lower Limit Acc_v
#48 - #59	/	Reserved
#60	User	PID V: Kp_i
#61	User	PID V: Ki_i
#62	User	PID V: Kd_i
#63	User	PID V: Kp_v
#64	User	PID V: Ki_v
#65	User	PID V: Kd_v
#66	User	PID V: Upper Limit Acc_i
#67	User	PID V: Lower Limit Acc_i
#68 - #77	/	Reserved
#78	Admin	Min Current Setpoint [A]
#79	Admin	Min Voltage Setpoint [V]
#80	Admin	Max Current Setpoint [A]
#81	Admin	Max Voltage Setpoint [V]
#82	Admin	Max Mosfet Temperature
#83	Admin	Min DC-link Threshold
#84	Admin	Earth Leakage Limit
#85	/	Reserved
#86	Admin	Current Regulation Fault Limit [A]
#87	Admin	Voltage Regulation Fault Limit [A]
#88	Admin	Regulation Fault Intervention Time [s]
#89	/	Reserved
#90	Admin	Interlock Enable Mask
#91	Admin	Interlock Activation State
#92	Admin	Interlock #1 intervention time [ms]
#93	Admin	Interlock #1 name
#94	Admin	Interlock #2 intervention time [ms]
#95	Admin	Interlock #2 name
#96 - #99	/	Reserved

Table 5: Parameters table for FAST-PS-ANET

### 3.10.1 MRG Command

The **MRG** command returns the value stored in the given parameter number. The correct form for the reading request is as follow:

**MRG:parameter\_index|r|n**

where “**parameter\_index**” is the index of the parameter to be read. The response to the **MRG** command is in the following format:

**#MRG:parameter\_index:parameter\_value|r|n**

where “**#MRG**” is the echo string, “**parameter\_index**” is the parameter’s index and “**parameter\_value**” is the parameter caption. The unit replies with “**#NAK:xx|r|n**”, when the command cannot be executed (“**xx**” is the error code) – for example if the given parameter is out of the permitted range.

#### Examples:

*MRG example of the FAST-PS Model (parameter #1):*

**MRG:1|r** → **#MRG:1:FAST-PS 2020-400|r|n**

*MRG example of read a not valid parameter’s index (parameter # -1):*

**MRG:-1|r** → **#NAK:03|r|n**

### 3.10.2 MWG Command

The **MWG** command lets users write a desired value in the given parameters index.

**MWG:***parameter\_index*:*parameter\_value*|*r*|*n*

where “*parameter\_index*” is the parameter’s index and “*parameter\_value*” is the content to be written.

Replies from the power unit to a **MWG** write are in the form “**#AK**|*r*|*n*” – when the command is correctly executed - or “**#NAK:xx**|*r*|*n*”, when the command cannot be executed (“*xx*” is the error code).

After a **MWG** command the values are immediately applied, but they are not stored in the internal memory. To store the modified parameters in the non-volatile internal memory it is necessary to use the **MSAVE Command**.

#### Examples:

*MWG example of the Module ID (parameter #30)*

**MWG:30:MAGNET A**|*r* →

← **#AK**|*r*|*n*

*MWG write example to the read-only field #1 (PS Model):*

**MWG:1:MAGNET A**|*r* →

← **#NAK:05**|*r*|*n*

### 3.10.3 Interlock Setting

The power supply external interlock can be enabled or disabled by writing to the corresponding Interlock Enable/Disable Mask field of the advanced configuration parameters (field #90), using the MWG command. The value to be written is in ASCII format, representing the corresponding bit mask, as shown in the following table:

Bit Mask	Interlock number	ASCII string
0001	Interlock #1	0x1
0010	Interlock #2	0x2
0100	Interlock #3	0x4
1000	Interlock #4	0x8

**Table 6:** Interlock Mask Parameter

The module allows to set also the interlock intervention time (how long an interlock signal needs to be at its activation level before tripping and thus generating a fault condition). The Intervention time parameters are stored in:

- field #92 for Interlock #1;
- field #94 for Interlock #2;

The value to be set is in ASCII format, representing the intervention time in milliseconds. The minimum settable value is 0 (immediate generating of fault condition) and the maximum value is 10.000 ms (corresponding to 10 seconds).

**Example:** if interlock #1 needs to have an interlock intervention time of 750 ms, the following command has to be sent to the power unit: “MWG:92:750\r”. This setting has no effect if the interlock is disabled.

#### Interlock Identification Name

Units also allows associating a name to the interlocks in order to read from the remote interface or to display on the local display the interlock condition name. The Intervention names are stored in:

- field #93 for Interlock #1;
- field #95 for Interlock #2;

The value to be set is in ASCII format, representing the interlock name.

**Example:** if the interlock #1 is associated to the cabinet door open, the following command can be sent to the power unit: “MWG:93:Cabinet door\r”. This setting has no effect if the interlock is disabled.

### 3.10.4 PASSWORD Command

The **PASSWORD** command can be used to unlock or lock the access to the protected parameter fields.

Several parameters are protected in order not to let inexperienced users to change some power supply parameters that might compromise the correct operation of the module. See the **Parameters table** for further details regarding the password-protected cells (parameters with *User* access privileges are not password protected; parameters with *Admin* access privileges are password protected; parameters with *Read Only* access privileges cannot be modified).

The correct format for this command is as follows:

**PASSWORD:***password\_word***|r|n**

where “*password\_word*” is the password to lock or unlock the protected parameter fields, that can be:

- “**PS-ADMIN**” to receive the *Admin* access privileges and unlock the protected parameter fields;
- “**LOCK**” to return to *User* access privileges and lock the protected parameters fields.

Replies from the unit to a **PASSWORD** command are in the form “**#AK**|r|n” – when the command is accepted - or “**#NAK:xx**|r|n”, when the command is not accepted (“xx” is the error code). When a wrong password word is received, the unit replies with a “**#NAK:07**|r|n” (error code 07 – invalid password) and locks the protected parameter fields.

To read the current privileges level the following query command can be used: “**PASSWORD:?**”. The response to this query command is in the following format:

**#PASSWORD:***privileges\_level***|r|n**

where “**#PASSWORD**” is the echo string and “*privileges\_level*” is the string indicating the privileges level.

The privileges level “**ADMIN**” indicates that the user is able to modify the protected parameter fields, otherwise “**USER**” indicates that the user is able to modify only the not protected parameter fields.

*The password to unlock password-protected cells is:*

**PS-ADMIN**

**Examples:**

*PASSWORD example of correct password word (unlock the protected cells):*

PASSWORD:PS-ADMIN|r →

← #AK |r|n

*PASSWORD example of correct password word (lock the protected cells):*

PASSWORD:LOCK|r →

← #AK |r|n

*PASSWORD example of wrong password word:*

PASSWORD:CAENELS|r →

← #NAK:07 |r|n

*PASSWORD access level query:*

PASSWORD:~|r →

← #PASSWORD:ADMIN|r|n

### 3.10.5 MSAVE Command

The **MSAVE** command can be used store the parameter fields in the non-volatile internal memory. If the parameter fields are not saved, they will be lost at power-off of the power supply.

Replies from the power modules to a **MSAVE** are in the form “**#AK|r|n**” – when the command is correctly executed - or “**#NAK:xx|r|n**”, when the command cannot be executed (“**xx**” is the error code).

**Example:**

*MSAVE example:*

The diagram illustrates the MSAVE command and its response. On the left, the command **MSAVE|r** is shown in blue and red text, with a black arrow pointing to the right. On the right, the response **#AK|r|n** is shown in purple and red text, with a black arrow pointing to the left. The background features a large, light gray stylized 'e' logo.