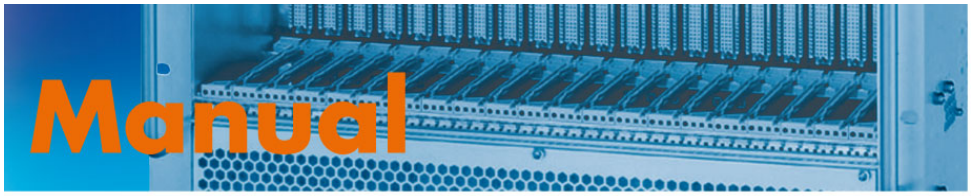


ELMA
Your Solution Partner

Sysmon Online*PRO* manual

24.08.09 / 024-154 Rev. 3

Table of content

1. Product description and functions 2
1.1. Voltage monitoring 3
1.2. Temperature monitoring 3
1.3. Fan monitoring and control 4
1.3.1. Temperature / speed characteristic 4
1.4. Digital inputs and outputs 5
1.4.1. Defining logic level 5
1.4.2. Manual setting / clearing of outputs 5
1.4.3. Controlling outputs through internal / external events 6
1.5. Generating the ACFAIL and SYSRESET VME signals 7
2. Ethernet interface 8
3. RS232 serial interface 8
4. WEB interface 9
4.1. Escape codes for the web page 9
4.2. Example and HTML code 10
5. Command Line Interface (CLI) 12
5.1. Log-in 12
5.2. Syntax directory 13
5.2.1. General syntax conventions 13
5.2.2. saveenv - command (save environment) 13
5.2.3. lanconfig - command (lan configuration) 14
5.2.4. voltage - command 15
5.2.5. port - command 16
5.2.6. temp - command (temperature) 17
5.2.7. fan - command 18
5.2.8. define - commands 19
5.2.8.1. define activevoltage 19
5.2.8.2. define voltagemask 19
5.2.8.3. define voltagereset 19
5.2.8.4. define powerfail 20
5.2.8.5. define inputmask 20
5.2.8.6. define vme 20
5.2.8.7. define inputled 21
5.2.8.8. define debouncetime 21
5.2.8.9. define tempdebouncetime 21
5.2.8.10. define startupdelay 22
5.3. logout - command 22
5.4. uptime - command 22
5.5. xmodem - command 23
5.6. passw - command 23
5.7. reboot - command 23
5.8. info – command 24
5.9. errors – command 24
6. Pin assignments of the connectors 26
6.1. Description of the signals 28
6.2. Layout / position of the connectors 29
7. Technical data / Dimensions 31
8. Order codes, product variants and accessories 32
9. Add-on module: FanCon Online*PRO* 33
9.1. Fancon OnlinePRO pin assignment 33
9.2. FanCon OnlinePRO technical data / Dimensions 33

1. Product description and functions

The Sysmon Online*PRO* (SOP) is a platform-independent system monitoring unit which monitors - and, if necessary, controls - the internal parameters of a System Platform such as voltages, temperatures, digital inputs and fan speeds. The unit uses a 16-bit microcontroller with an integrated 12-bit A/D converter and suitable peripheral circuitry for this.

These parameters of the System Platform are measured or read in and analyzed at regular intervals. If a parameter value exceeds or falls below a user-defined limit, the SOP detects this as an “Event”. In principle, fixed pre-defined actions are carried out when the corresponding events occur. These actions are described later. The user can also use these “Events” to set digital outputs.

Any limit infringements (high or low) occurring for voltages V1 to V4, temperature and fan minimum speed are signaled via the front-panel LEDs. If indication of more than 4 voltages is required, the ELMA LED display (p/n 024-927) can be connected to the SOP as an accessory and installed at any position in the system.

The measured values are retrievable at any time via the RS232 serial interface and via Telnet. In addition, limits and system parameters can be changed at any time with the unit in service. As a result, the SOP – and hence the connected system - can be controlled and monitored online via any computer with an Internet connection.

To complete the information spectrum, the integrated web page, which the user can design himself, is also continuously updated with the measured values.

When the SOP is used in VME systems, it takes care of the generation of the "ACFAIL2" and "SYSRESET2" VME signals.

The measured values are evaluable at any time via the RS232 serial interface and via Telnet. In addition, limits and system parameters can be changed at any time with the unit in service.

1.1. Voltage monitoring

- Monitoring of up to 8 voltages (standard: 5 x pos. and 3 x neg.)
- Upper and lower limits can be individually configured for each voltage
- Monitoring of individual voltages can be disabled
- Status LED on the front panel (024-874 and 024-875 only) for voltages V1 - V4 (+5V, +3.3V, +12V, -12V)
- A limit infringement (high or low) of one of the monitored voltages is an internal event (global error – syntax directory 5.2.8. define - commands) that can be further processed by the port command (syntax directory 5.2.5. port - command).
- Optionally, a SYSRESET for VME systems can be initiated in the event of a fault.

- Monitored voltage ranges of the individual inputs:
  V1= 0V to +8V V2= 0V to +8V
  V3= 0V to +14V V4= -14V to 0V
  V5= 0V to +8V V6= -8V to 0V
  V7= -26V to 0V V8= 0V to +26V

- Commands for the voltage monitoring:

  voltage (syntax directory 5.2.4. voltage - command)
  define activevoltage (syntax directory 5.2.8.1. define activevoltage)
  define voltagemask (syntax directory 5.2.8.2. define voltagemask)
  define voltagereset (syntax directory 5.2.8.3. define voltagereset)

1.2. Temperature monitoring

- Monitoring of up to 6 analog temperature sensors (10 kohm NTC thermistors with ß=3950)
- Freely-configurable upper and lower limits
- Temperature range from –20°C to +100°C
- Temperature measurement accuracy +/-3°C (max.)
- Status LED on the front panel for temperature error indication (024-874 and 024-875 only)
- Monitoring of a further 8 digital temperature sensors via the I^2C bus.
- In the event of an error, the yellow temp-fail LED on the front panel is activated and the fans are set to maximum speed until the temp-fail condition is cleared.
- A limit infringement (high or low) of one of the monitored temperatures is an internal event (global error) that can be further processed by the **port** command (syntax directory 5.2.5. port - command).

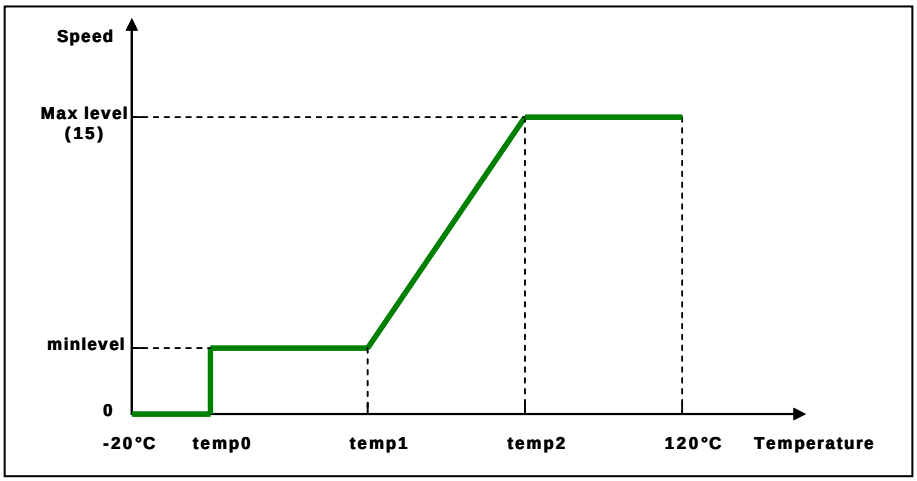
- Commands for the temperature monitoring:

  temp (syntax directory 5.2.6. temp - command (temperature))

1.3. Fan monitoring and control

- Monitoring and control of up to 12 fans. The fans require a tachometer signal output as well as a PWM signal input
- Configurable minimum speed (>700rpm)
- Speed control via PWM signal (frequency configurable from 20Hz to 10KHz)
- Control of non-PWM fans possible with FanCon Online*PRO* add-on module (section 9. Add-on module: FanCon OnlinePRO)
- LED on the front panel for indication of fan errors
- Soft start – to minimize the starting current
- A limit infringement of the minimum speed (**minspeed**) of one of the monitored fans is an internal event (global error) that can be further processed by the **port** command (syntax directory 5.2.5. port - command).
- In the event of an error, the yellow fan-fail LED on the front panel is activated and the fans are set to maximum speed until the fan-fail condition is cleared.
- Freely-configurable temperature / speed characteristic by setting 4 parameters (minlevel, temp0, temp1, temp2). Setting are valid (uniform) for all temperature sensor.

1.3.1. Temperature / speed characteristic



The fan speed is controlled as a function of the temperature. The speed range of the fan is divided into 15 steps. The graph gives rise to 4 control regions:

1. -20°C to temp0: Fans stopped
2. temp0 to temp1: Fans run all the time at a minimum level (minlevel) specified in advance
3. temp1 to temp2: Fans vary their speed proportional to the temperature
4. temp2 to 120°C: Fans run at full speed at level 15

- Command for the fan control and for setting the parameters:

fan (syntax directory 5.2.7. fan - command)

1.4. Digital inputs and outputs

- 16 digital inputs grouped in two 8-bit ports (port 1 + port 2)
- 16 outputs grouped in two 8-bit ports (port A + port B)
- Logic level: 5V TTL. Outputs can carry 20mA with respect to GND
- Each individual input/output can be declared as active low or active high (section 1.4.1. Defining logic level)
- Manual setting and clearing of outputs (section 1.4.2. Manual setting / clearing of outputs)
- Outputs can be set independently of internal and external events.
  (section 1.4.3. Controlling outputs through internal / external events)

1.4.1. Defining logic level

In order to be able to work independently of actual voltage levels, the states of the inputs and outputs are considered as logic levels. For this, you first have to define whether the signals are active low or active high. The "**port ... level**" command is used for this. (syntax directory 5.2.5. port - command)

- Command to define logic levels:

  port...level (syntax directory 5.2.5. port - command)

	1	2	3	4	5	6	7	8	Hex Value
Port x active level	0	0	1	0	1	0	1	1	2B

1.4.2. Manual setting / clearing of outputs

The states of the outputs can be changed with the "**port ... set**" or "**port ... clr**" command. The addressed output is enabled with "set". The voltage then applied at the output is determined by the previously specified definition of its logic level (section 1.4.1. Defining logic level).

Note: this function can be disabled in ADMIN access by setting the appropriate bits in the mask of the output (section 1.4.3. Controlling outputs through internal / external events)

- Commands to enable / disable outputs:

  port...set (syntax directory 5.2.5. port - command)
  port...clr (syntax directory 5.2.5. port - command)

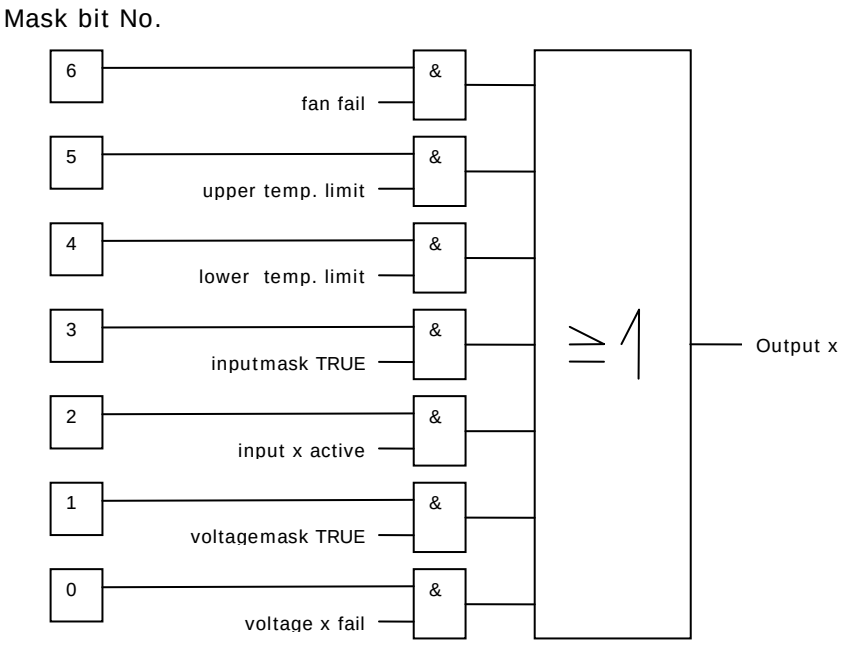
1.4.3. Controlling outputs through internal / external events

To control outputs dependent on internal events (e.g. temperature errors, fan errors) or external events (enabled inputs), a bitmask can be created for each individual output. If the “OR” logical operator of the individual bits of this mask = 1 (TRUE), the output is enabled. The voltage then applied at the output is determined by the previously specified definition of its logic level (Section 1.4.1. Defining logic level). The output is reset again as soon as the “OR” logical operator = 0 (FALSE)

Mask structure for output No. “x”:

Bit No.	Description
7	1= Enable “port…set” and “port…clr” for output “x” (Section 0)
6	Event “fan fail”
5	Event “upper temperature limit exceeded”
4	Event “lower temperature limit infringed”
3	Event “inputmask TRUE” (Section 0)
2	Event “input x active” (corresponding input active)
1	Event “voltagemask TRUE” (Section 0)
0	Event “voltage x out of limits”

Logic plan:



- Command to set the masks of the outputs:

  port…mask (syntax directory 5.2.5. port - command)

1.5. Generating the ACFAIL and SYSRESET VME signals

The SOP generates the ACFAIL and SYSRESET signals in accordance with VME specification ANSI/VITA 1-1994. For correct performance of this function, the SOP must be suitably set up via the "**define vme**" (syntax directory 5.2.6. temp - command (temperature)) and "**define powerfail**" (syntax directory 5.2.8.4. define powerfail) commands. The signals can be picked up at the "Voltage1" connector.Timing diagram from VME specification ANSI/VITA 1-1994:

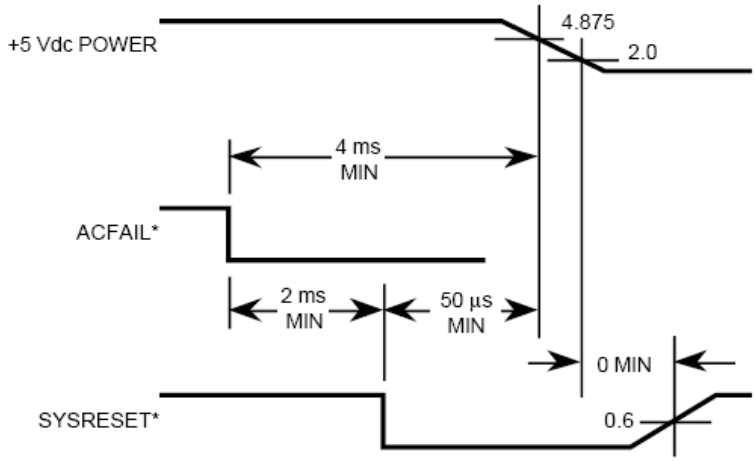


Figure 5 - 4 Power Monitor Power Failure Timing

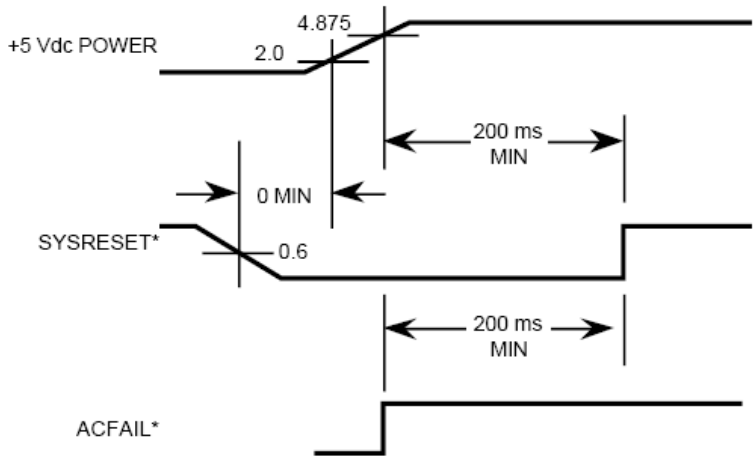


Figure 5 - 5 Power Monitor System Restart Timing

2. Ethernet interface

The integrated 10Mbps Ethernet interface allows the SOP to be linked to any existing network. The interface supports HTTP and TELNET protocols via TCP/IP. So all monitored system parameters can be displayed via a standard browser (HTTP protocol). You have full access to the commands of the Command Line Interface (CLI) via TELNET, enabling you to read parameters and configure the Sysmon. The use of standard protocols avoids the need for special software or drivers and so achieves platform-independence. The TCP/IP protocol supports 10 simultaneous connections – the maximum packet size is limited to 1k.

On Windows systems, we recommend the use of “Hyperterminal” or “PuTTY” as TELNET client.

! The factory default setting for the Sysmon IP address is 193.155.166.51 - the IP address must be adjusted via the **lanconfig** command (syntax directory 5.2.3. lanconfig - command (lan configuration)) for use in your network.

Terminal settings:

- Local echo: *off*
- Local line editing: *off*
- Backspace key: *Control-H*

! Use a crossed network cable if you connect the Sysmon direct to your PC.

3. RS232 serial interface

The SOP provides an RS232 serial interface over which the commands of the Command Line Interface (CLI) can be sent.

! Use a NULL MODEM cable for direct connection to the serial port of a PC.

On Windows systems, we recommend the use of “Hyperterminal” as the terminal program.

Terminal settings:

- 9600 bits per second
- data bits: 8
- parity: none
- stop bit: 1

In addition, the “**xmodem**” CLI command can be used via the serial interface for file transfer. (syntax directory 5.5. xmodem - command)

4. WEB interface

For remote polling using a standard browser, the Sysmon provides a web page that can be created by the user and saved via the "**xmodem web**" command (syntax directory 5.5. xmodem - command). The maximum file size of the HTML file is 10 Kbytes. Only HTML files can be saved. If images are used, these must be filed externally.

The following values can be displayed via the web page:

- Voltage values V1-V8
- Status colors (red/green/gray) of voltages V1-V8
- Temperature values T1-T14
- Status colors (yellow/gray) of temperatures T1-T14
- Fan speeds F1-F12
- Status colors (yellow/gray) of fans F1-F12

In the created HTML files, suitable Escape Codes (section 4.1. Escape codes for the web page) must be inserted at the positions where the aforementioned values / colors are to be displayed. The parser integrated on the Sysmon continuously replaces these codes with the current values.

4.1. Escape codes for the web page

The Sysmon continuously replaces the following escape codes with the current values in the saved HTML file.

Value of voltage No. x (Voltage x value): **~VxV**
Status color of voltage No. x (Voltage x color): **~VxC**
Valid values for x: **1,2,3,4,5,6,7,8**

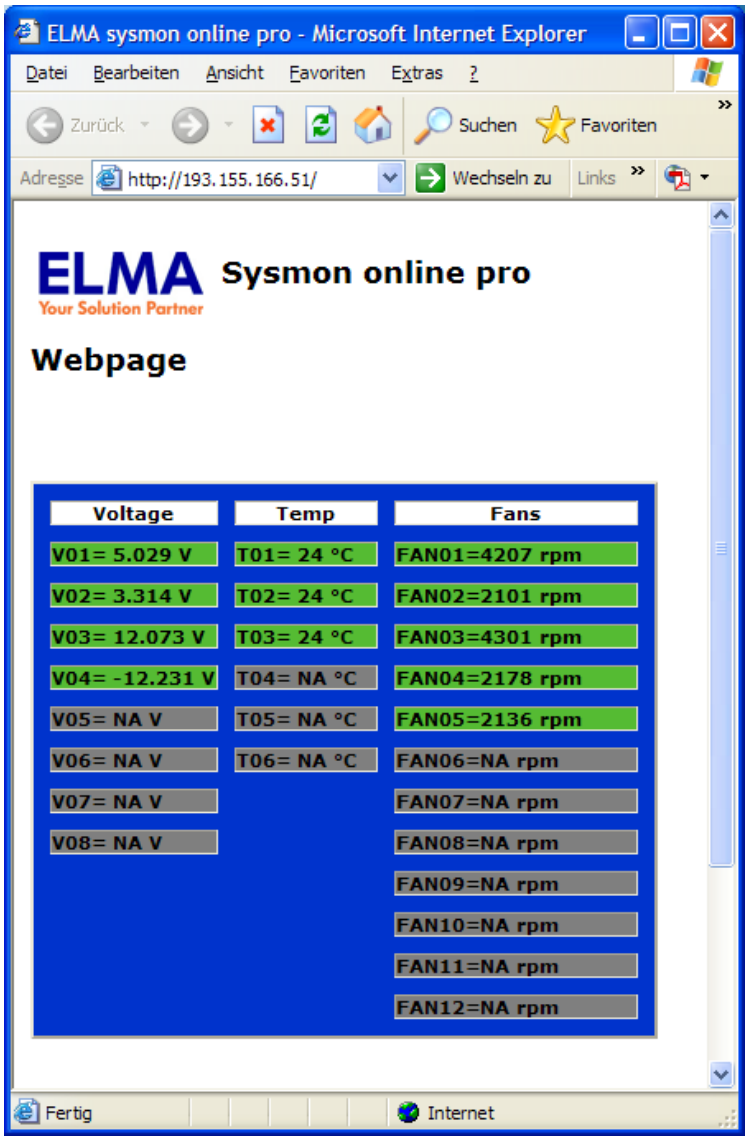
Value of temperature No. x (Temperature x value): **~TxV**
Status color of temperature No. x (Temperature x color): **~TxC**
Valid values for x: **1,2,3,4,5,6,7,8,9,a,b,c,d,e (hex value)**

Speed of fan No. x (Fan x speed): **~FxV**
Status color of fan No. x (Fan x color): **~FxC**
Valid values for x: **1,2,3,4,5,6,7,8,9,a,b,c (hex value)**

- Command for transmission of the web page:

  xmodem web (syntax directory 5.5. xmodem - command)

4.2. Example and HTML code



HTML-Code :

(Note: Pictures can't be stored on the SOP. Pictures have to be stored external with the adequate URL implemented in the HTML code!).

```
<table width="95%" border="2" cellspacing="10" cellpadding="0" bgcolor="#0033CC">
      <tr>
        <td width="107" bgcolor="#FFFFFF">
          <div align="center"><b>Voltage</b></div>
        </td>
        <td width="92" bgcolor="#FFFFFF">
          <div align="center"><b>Temp</b></div>
        </td>
        <td width="154" bgcolor="#FFFFFF">
          <div align="center"><b>Fans</b></div>
```

```
  </td>
 </tr>
 <tr>
  <td width="107" bgcolor="~V1C"><b>V01= ~V1V V</b></td>
  <td width="92" bgcolor="~T1C"><b>T01= ~T1V &deg;C</b></td>
  <td width="154" bgcolor="~F1C"><b>FAN01=~F1V rpm</b></td>
 </tr>
 <tr>
  <td width="107" bgcolor="~V2C"><b>V02= ~V2V V</b></td>
  <td width="92" bgcolor="~T2C"><b>T02= ~T2V &deg;C</b></td>
  <td width="154" bgcolor="~F2C"><b>FAN02=~F2V rpm</b></td>
 </tr>
 <tr>
  <td width="107" bgcolor="~V3C"><b>V03= ~V3V V</b></td>
  <td width="92" bgcolor="~T3C"><b>T03= ~T3V &deg;C</b></td>
  <td width="154" bgcolor="~F3C"><b>FAN03=~F3V rpm</b></td>
 </tr>
 <tr>
  <td width="107" bgcolor="~V4C"><b>V04= ~V4V V</b></td>
  <td width="92" bgcolor="~T4C"><b>T04= ~T4V &deg;C</b></td>
  <td width="154" bgcolor="~F4C"><b>FAN04=~F4V rpm</b></td>
 </tr>
 <tr>
  <td width="107" bgcolor="~V5C"><b>V05= ~V5V V</b></td>
  <td width="92" bgcolor="~T5C"><b>T05= ~T5V &deg;C</b></td>
  <td width="154" bgcolor="~F5C"><b>FAN05=~F5V rpm</b></td>
 </tr>
 <tr>
  <td width="107" bgcolor="~V6C"><b>V06= ~V6V V</b></td>
  <td width="92" bgcolor="~T6C"><b>T06= ~T6V &deg;C</b></td>
  <td width="154" bgcolor="~F6C"><b>FAN06=~F6V rpm</b></td>
 </tr>
 <tr>
  <td width="107" bgcolor="~V7C"><b>V07= ~V7V V</b></td>
  <td width="92"><b></b></td>
  <td width="154" bgcolor="~F7C"><b>FAN07=~F7V rpm</b></td>
 </tr>
 <tr>
  <td width="107" bgcolor="~V8C"><b>V08= ~V8V V</b></td>
  <td width="92"><b></b></td>
  <td width="154" bgcolor="~F8C"><b>FAN08=~F8V rpm</b></td>
 </tr>
 <tr>
  <td width="107"><b></b></td>
  <td width="92"><b></b></td>
  <td width="154" bgcolor="~F9C"><b>FAN09=~F9V rpm</b></td>
 </tr>
 <tr>
  <td width="107"><b></b></td>
  <td width="92"><b></b></td>
  <td width="154" bgcolor="~FaC"><b>FAN10=~FaV rpm</b></td>
 </tr>
 <tr>
  <td width="107"><b></b></td>
  <td width="92"><b></b></td>
  <td width="154" bgcolor="~FbC"><b>FAN11=~FbV rpm</b></td>
 </tr>
 <tr>
  <td width="107" height="2"><b></b></td>
  <td width="92" height="2"><b></b></td>
  <td width="154" bgcolor="~FcC" height="2"><b>FAN12=~FcV rpm</b></td>
 </tr>
</table>
```

5. Command Line Interface (CLI)

The Command Line Interface (short-form: CLI) is available via both Telnet (section 2. Ethernet interface) and the RS232 serial interface (section 3. RS232 serial interface).

The user can read or newly configure and save system parameters via the CLI. Access is divided into 2 profiles and is password-protected.

“user” profile:

System parameters can only be read in this profile – the exception to this write-protect is the **lanconfig** command (syntax directory 5.2.3. lanconfig - command (lan configuration)) for setting the IP, subnet and gateway addresses. If the outputs have been suitably configured in the admin profile (section 1.4.3. Controlling outputs through internal / external events), the user can also manually set and clear individual outputs (section 1.4.2. Manual setting / clearing of outputs).

“admin” profile:

! Full access to all system parameters. All available CLI commands can be executed. To avoid possible damage or malfunctions, the access data for this profile must only be known to trained personnel with appropriate knowledge and competence relating to the system in which the SOP is used!

5.1. Log-in

As soon as you have established a connection via Telnet or RS232, you will be prompted to log in. You might have to press ENTER once to get to the login field.
The measured values are evaluable at any time via the RS232 serial interface and via Telnet. In addition, limits and system parameters can be changed at any time with the unit in service.

Default access settings:

login: user
password: USER

login: admin
password: ADMIN

5.2. Syntax directory

5.2.1. General syntax conventions

Command [parameter1 | parameter2 | parameter3 [*value*]]

[] = optional

A command on its own with no entry of other parameters returns all available current values associated with the command.

If ONE parameter is entered, but no value, the SOP only returns the current value available to this parameter.

If the command line contains a *value*, this is assigned to the corresponding parameter and saved temporarily in the RAM. The change is active immediately. If the value is finally to be saved in the FLASH ROM and is also to be valid after the reboot, the environment variables must be saved with the "**savenenv"** command (syntax directory 5.2.2. saveenv - command (save environment)). Changes not confirmed with saveenv are lost after a reboot.

5.2.2. saveenv - command (save environment)

Syntax:

saveenv

Function:

Saves all environment variables temporarily held in the RAM permanently in the FLASH ROM.

5.2.3. lanconfig - command (lan configuration)

Syntax:

lanconfig [ip | mask| gate [*address***]]**

Functions:

Readout or setting of network parameters.

- **no parameter** – return of IP, mask and gateway addresses of the LAN interface
- **ip** – IP address of the Sysmon
- **mask** – network mask
- **gateway** – standard gateway
- ***address*** – IP address of the SOP (factory default IP setting is 193.155.166.51)

! After a new address is set, the change must be saved with **saveenv** and the SOP restarted. Either with the **reboot** command (syntax directory 5.2.7. fan - command) or by the reset key.

Examples:

- Readout of all network parameters

```
%>lanconfig
IP=193.155.166.51
Mask=255.255.255.0
Gateway=193.155.166.100
```

- Readout of the IP address

```
%>lanconfig ip
IP=193.155.166.51
```

- Changing the IP address

```
%>lanconfig ip 196.100.100.1
IP=196.100.100.1
```

5.2.4. voltage - command

Syntax:

voltage [sysmon |min | max] [*voltagenumber***[***value***]]**

Functions:

Readout of the voltage values or setting the individual limits for the voltage monitoring

- **sysmon** = output of the SOP operating voltage (no other parameters permissible)
- **min** = lower limit
- **max** = upper limit
- ***voltagenumber*** = No. of the analog inputs 1-8
- ***value*** = limit

Examples:

- Readout of all voltage values. Any voltages not connected are depicted with the value “NA”.

```
%>voltage
V1=4.958 V2=3.304 V3=12.132 V4=NA V5=NA V6=NA V7=NA V8=NA
```

- Readout of the upper limit of voltage V1 (+5V)

```
%>voltage min 1
V1=4.749
```

- Setting the upper limit of voltage V1 to 5.5V

```
%>voltage max 1 5.5
V1=5.499
```

5.2.5. port - command

Syntax:

port [*port_name* **[level** [*value*]] **| mask** *output* [*value*] **| set** *output* **| clr** *output*]]

Functions:

Readout and change of the port states, the logic levels and access to port masks.

- ***port_name*** = 1,2,A,B (1,2 = inputs ; A,B = outputs)
- **level** = readout or setting of the logic level. An 8-bit value is expected with the subsequent entry of *value*. Each bit represents the corresponding input or output.
- **mask *output*** = only for port A,B (outputs). Readout or setting the mask for the addressed output. An 8-bit value is expected with the subsequent entry of *value*. Each bit represents the corresponding input or output.
- **set *output*** = only for port A,B (outputs). Enable addressed output
- **clr *output*** = only for port A,B (outputs). Disable addressed output.
- ***value*** = 8-bit value in HEX notation (0x00 to 0xff) or decimal (0 to 255).
- ***output*** = entry of output A or B

Please refer to Section 1.4.3. Controlling outputs through internal / external events for the bit structure for masks.

Examples:

- Output of the states of all ports

```
%>port
   Port1=0xFE
   Port2=0xFF
   PortA=0xFD
   PortB=0xFF
```

- Set inputs 1-4 of port 1 to “active low” (or “high when inactive”) and inputs 5-8 to “active high” (or “low when inactive”)

```
%>port 1 level 0x0f
 Level_Port0=0x0F
```

- Set the mask for output 1 of port A such that this output is enabled as soon as voltage No. 1 is out of limits. The output for **port set** and **port clr** must be enabled at the same time

```
%>port a mask 1 0x81
 Mask PortA.1=0x81
```

- Enable output 1 of port A

```
%>port a set 1
 Done...
```

5.2.6. temp - command (temperature)

Syntax:

temp [[min | max | digitalno | analogno] [*value*]]

Output of the current temperatures and limits or setting of temperature limits and number of sensors.
Setting are valid (uniform) for all temperature sensor.

- **min** = lower limit
- **max** = upper limit
- **analogno**= number of analog sensors
- **digitalno**= number of digital I²C sensors
- ***value*** = limit in °C or value for the number of sensors

Examples:

- Output of all available current temperature readings

```
%>temp
 A: T1=24 T2=24 T3=24
```

- Output of the lower temperature limit

```
%>temp min
 Temp min =-10
```

- Setting the upper limit to 70°C

```
%>temp max 70
 Temp max =70
```

- Setting the number of analog sensors to 4

```
%>temp analogno 4
 Analog temp sensors:4
```

5.2.7. fan - command

Syntax:

fan [minspeed |minlevel | temp0 | temp1 | temp2 | pwmfreq | level | fanno [*value*]]

Functions:

Output or configuration of the fan parameters.

- **minspeed** = readout/setting of the minimum speed
- **minlevel** = readout/setting of the minlevel value
- **temp0** = readout/setting of the temp0 value
- **temp1** = readout/setting of the temp1 value
- **temp2** = readout/setting of the temp2 value
- **pwmfreq** = readout/setting of the PWM frequency
- **level** = readout of the current level that the fans are running at
- **fanno** = readout/setting of the number of fans
- *value* = associated value for the individual parameters

The meaning of the parameters is explained in detail in section 1.3. Fan monitoring and control.

Examples:

- Set the number of fans to 3

```
%>fan fanno 3
 Fan number=3
```

- Readout of the current fan speeds

```
%>fan
 F1=4956 F2=4932 F3=0
```

- Set the minimum speed to 700 (a FANFAIL is generated below this speed).

```
%>fan minspeed 700
 Min speed=700
```

- Readout of the minlevel parameter

```
%>fan minlevel
 Min level=3
```

- Setting the temp0 parameter to 0°C (the fans stop below this value)

```
%>fan temp0 0
 Fan temp0=0
```

5.2.8. define - commands

5.2.8.1. define activevoltage

Bit	8	7	6	5	4	3	2	1
V	V8	V7	V6	V5	V4	V3	V2	V1

Syntax:

define activevoltage [*value***]**

Example:
0x0F = V1 to V4 will be monitored

Function:

Output or setting of the voltages to be monitored

- ***value*** = 8-bit value in HEX notation (0x00 to 0xff) or decimal (0 to 255). Each bit represents the particular voltage.

5.2.8.2. define voltagemask

Syntax:

define voltagemask [*value***]**

Function:

Output or setting of the mask for monitoring a number of voltages and generation of an internal event.

- ***value*** = 8-bit value in HEX notation (0x00 to 0xff) or decimal (0 to 255). Each bit represents the particular voltage.

If all the voltage numbers set accordingly are out of limits, the result of this mask is TRUE and can be further processed with the **port** command (section 1.4.3. Controlling outputs through internal / external events)

5.2.8.3. define voltagereset

Syntax:

define voltagereset [*value***]**

Function:

Output or setting of the mask for monitoring a number of voltages that are to initiate a SYSRESET.

- ***value*** = 8-bit value in HEX notation (0x00 to 0xff) or decimal (0 to 255). Each bit represents the particular voltage.

If all the voltage numbers set accordingly are out of limits, the result of this mask is TRUE and a SYSRESET is initiated.

5.2.8.4. define powerfail

Syntax:

define powerfail [*value***]**

Function:

Output or setting of the logic level of the connected powerfail signal at the POWER connector. If powerfail is active and the VME flag is at "1", the VME timing as described in section 1.4.3. Controlling outputs through internal / external events is triggered.

- ***value*** = decimal value 0,1 (0 = active low ; 1 = active high)

5.2.8.5. define inputmask

Syntax:

define inputmask [*value***]**

Function:

Output or setting of the mask for monitoring a number of inputs and generation of an internal event.

- ***value*** = 16-bit value in HEX notation (0x0000 to 0xffff) or decimal (0 to 65535). Each bit represents the particular input.

  If all the inputs set accordingly are active, the result of this mask is TRUE and can be further processed with the **port** command (section 1.4.3. Controlling outputs through internal / external events)

5.2.8.6. define vme

Syntax:

define vme [*value***]**

Function:

Output or setting of the flag indicating whether the SOP is connected to a VME system and hence whether the VME timing as described in section 1.5. Generating the ACFAIL and SYSRESET VME signals should be carried out.

- ***value*** = decimal value 0,1 (0 = no VME System ; 1 = VME System)

5.2.8.7. define inputled

Syntax:

define inputled [*value***]**

Function:

Setting of the mask witch leads to an activated InputLED on the LED display 024-927.

- ***value*** = 16-bit value in HEX notation (0x0000 to 0xffff) or decimal (0 to 65535). Each bit represents the particular input.

  If one of the inputs set accordingly are active, the result of this mask is TRUE and the InputLED is activated.

5.2.8.8. define debouncetime

Syntax:

define debouncetime [*value***]**

Function:

Setting of the debounce time for the inputs.

- ***value*** = decimal (0 to 15) 0=0ms ; 1=100ms ; 15=1.5s

Example:

- Setting the debounce time to 500ms

```
%>define debouncetime 5
  Debounce time=5
```

5.2.8.9. define tempdebouncetime

Syntax:

define tempdebouncetime [*value***]**

Function:

Setting of the debounce time for the temperature measurement.

- ***value*** = decimal (0 to 30) 0=0ms ; 1=100ms ; 30=3s

Example:

- Setting the debounce time to 2s

```
%>define tempdebouncetime 20
  Temp debounce time=20
```

5.2.8.10. define startupdelay

Syntax:

define startupdelay [*value***]**

Function:

Setting of a timeout at start-up before the measurements starts.

- ***value*** = decimal (0 to 15) 0=0ms ; 1=100ms ; 15= 1.5s

Example:

- Setting the delay time to 300ms

```
%>define startupdelay 3
  Startup delay=3
```

5.3. logout - command

Syntax:

logout

Function:

End current session.

5.4. uptime - command

Syntax:

uptime

Function:

Indication of the time since the last reboot.

5.5. xmodem - command

Syntax:

xmodem [config | web] → **only via RS232!**

Function:

Transfer of files with XMODEM protocol – only available via RS232.

- **config** – start of the transfer of a binary configuration file
- **web –** start of the transfer of an HTML file

After the command is entered, the SOP goes into data receive mode and waits for the data to be sent in XMODEM protocol. The SOP reports this with continuous C's on the terminal. You can then start the file transfer immediately with your terminal program and select XMODEM as the protocol.

Example:

```
%>xmodem web
%>CCCC
```

5.6. passw - command

Syntax:

passw

Function:

Changing the password in the current profile.

5.7. reboot - command

Syntax:

reboot

Function:

Restart Sysmon (RESET)

5.8. info – command

Syntax:

info

Function:

Returns MAC-Adress, Serial No. and Firmware version

Example:

```
%>info
 MAC=00:14:DB:00:04:C1
 Serial No=10505681
 Version=3.10
```

5.9. errors – command

Syntax:

errors [clr]

Function:

Output or deletion of the stored error statuses (global errors)
and input events (input errors).

At each event (Fan Fail, Voltage Fail, etc.) and on activation of inputs, the SOP saves these events and files them in the **global errors** and **input errors** registers. Even if the error or event is no longer current, it stays saved in the register until the error memory is cleared again with **errors clr**.

- **without parameter** – return of the value in 16-bit HEX format (0x0000 to 0xffff)
- **clr** – clear global errors and input errors

“global errors” register structure:

Bit	Description
0	1 = Voltage No.1 out of limits
1	1 = Voltage No.2 out of limits
2	1 = Voltage No.3 out of limits
3	1 = Voltage No.4 out of limits
4	1 = Voltage No.5 out of limits
5	1 = Voltage No.6 out of limits
6	1 = Voltage No.7 out of limits
7	1 = Voltage No.8 out of limits
8	1 = Fan error (FANFAIL) has occurred
9	1 = Lower temperature limit infringed
10	1 = Upper temperature limit exceeded
11	1 = PowerFail Signal has been detected
12	Internal
13	Internal
14	Internal
15	Internal

“input errors” register structure:

Bit	Description
0	1 = Input No.1 was active
1	1 = Input No.2 was active
2	1 = Input No.3 was active
3	1 = Input No.4 was active
4	1 = Input No.5 was active
5	1 = Input No.6 was active
6	1 = Input No.7 was active
7	1 = Input No.8 was active
8	1 = Input No.9 was active
9	1 = Input No.10 was active
10	1 = Input No.11 was active
11	1 = Input No.12 was active
12	1 = Input No.13 was active
13	1 = Input No.14 was active
14	1 = Input No.15 was active
15	1 = Input No.16 was active

Example:

- Example of the stored error statuses / input events

```
%>errors
 Global errors=0x0000
 Input  errors=0x0000
```

- Clearing the stored error statuses / input events

```
%>errors clr
 Permanent errors cleared...
```

6. Pin assignments of the connectors

Power

Pin	Signal
1	+12V
2	+12V
3	GND
4	GND
5	+5V
6	+3.3V_OUT
7	PWRFAIL

Voltage 1

Pin	Signal
1	SYSRESET#
2	ACFAIL#
3	V4 -12V
4	GND
5	V3 +12V
6	GND
7	V2 +3.3V
8	GND
9	V1 +5V
10	GND

Voltage 2

Pin	Signal
1	SYSFAIL#
2	NC
3	V7N
4	GND
5	V8P
6	GND
7	V6N
8	GND
9	V5P
10	GND

FCON

Pin	Signal
A1	GND
A2	PRST#
A3	Sysfail#
A4	FAL#
A5	GND
A6	GND
B1	GND
B2	V4_-12V
B3	V3_+12V
B4	V2_+3,3
B5	V1_+5V
B6	NC

Fan

Pin	Signal
1	
3	
5	RSV
7	
9	TACH_10
11	PWM_2
13	TACH_8
15	
17	PWM_1
19	TACH_4
21	TACH_3
23	
25	TACH_1

Pin	Signal
2	
4	
6	PWM_3
8	TACH_11
10	TACH_9
12	
14	TACH_7
16	TACH_6
18	TACH_5
20	
22	PWM_0
24	TACH_2
26	TACH_0

In 1/2

Pin	Signal
1	IN1.1
3	IN1.3
5	IN1.4
7	GND
9	IN1.7
11	IN2.1
13	IN2.2
15	GND
17	IN2.5
19	IN2.7

Pin	Signal
2	IN1.2
4	+5V
6	IN1.5
8	IN1.6
10	IN1.8
12	+5V
14	IN2.3
16	IN2.4
18	IN2.6
20	IN2.8

Temp 1

Pin	Signal
1	Temp3+
2	Temp3-
3	Temp2+
4	Temp2-
5	Temp1+
6	Temp1-

Temp 2

Pin	Signal
1	Temp6+
2	Temp6-
3	Temp5+
4	Temp5-
5	Temp4+
6	Temp4-

Out A/B

Pin	Signal
1	OUTA.1
3	OUTA.3
5	OUTA.4
7	GND
9	OUTA.7
11	OUTB.1
13	OUTB.2
15	GND
17	OUTB.5
19	OUTB.7

Pin	Signal
2	OUTA.2
4	+5V
6	OUTA.5
8	OUTA.6
10	OUTA.8
12	+5V
14	OUTB.3
16	OUTB.4
18	OUTB.6
20	OUTB.8

I2C_2

Pin	Signal
1	SDA_2
2	SCL_2
3	RST
4	INT_2
5	+5V
6	GND

Digital Temp

Pin	Signal
1	+5V
2	SDA_3
3	GND
4	SCL_3

1P1

	Z	A	B	C	D	E	F	
25	GND	E0TP	E0RP	GND			GND	**25**
24	GND	E0TM	E0RM				GND	**24**
23	GND	GND	GND		GND	GND	GND	**23**
22	GND						GND	**22**
21	GND		SDA_3	SCL_3	INT_3	GND	GND	**21**
20	GND	RS232_TX				REF2.5	GND	**20**
19	GND		REF2.5	TEMP6		TEMP5	GND	**19**
18	GND			TEMP4			GND	**18**
17	GND	RS232_RX		TEMP3			GND	**17**
16	GND			TEMP2			GND	**16**
15	GND			TEMP1			GND	**15**
14								**14**
13								**13**
12								**12**
11	GND	SDA_2	SCL_2	INT_2	GND		GND	**11**
10	GND	TACH_0	TACH_1	TACH_2			GND	**10**
9	GND	PWM_0	TACH_3	TACH_4	+5V		GND	**9**
8	GND	PWM_1	TACH_5	TACH_6			GND	**8**
7	GND	PWM_2	TACH_7	TACH_8	+5V	V3 +12V	GND	**7**
6	GND	PWM_3	TACH_9	TACH_10		V4 -12V	GND	**6**
5	GND	TACH_11	SDA_1	SCL_1	+5V	V5_VIO	GND	**5**
4	GND	GND	SDA_0	SCL_0		V2 +3.3V	GND	**4**
3	GND				+5V	V1 +5V	GND	**3**
2	GND		GND	GND	PWRFAIL	GND	GND	**2**
1	GND	SYSRESET#	ACFAIL#		RST		GND	**1**

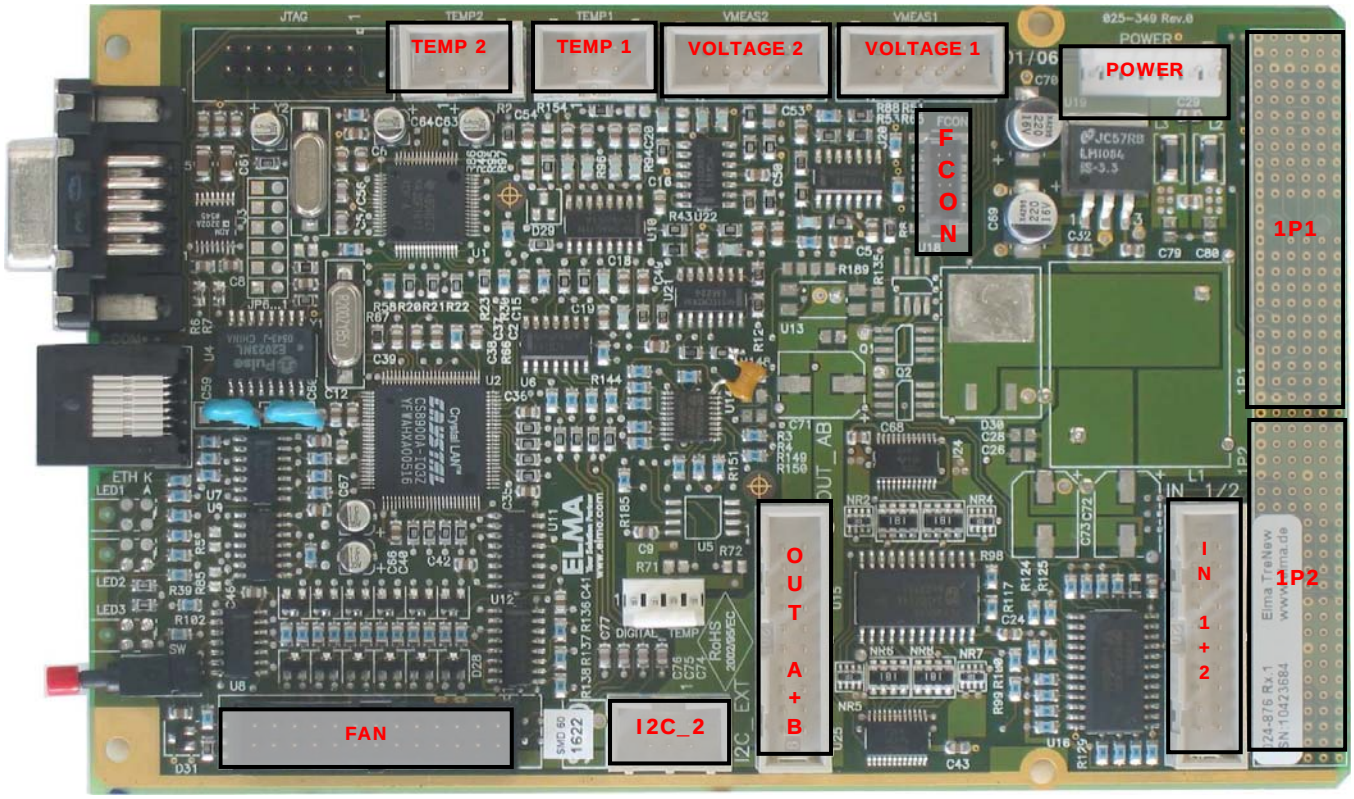
1P2

	Z	A	B	C	D	E	F	
22	GND						GND	**22**
21	GND			+5V			GND	**21**
20	GND						GND	**20**
19	GND			GND			GND	**19**
18	GND						GND	**18**
17	GND						GND	**17**
16	GND						GND	**16**
15	GND				OUTB.2	OUTB.1	GND	**15**
14	GND			+5V	OUTA.8	OUTA.7	GND	**14**
13	GND	OUTB.8	OUTB.7		OUTA.6	OUTA.5	GND	**13**
12	GND	OUTB.6	OUTB.5		OUTA.4	OUTA.3	GND	**12**
11	GND	OUTB.4	OUTB.3	GND	OUTA.2	OUTA.1	GND	**11**
10	GND						GND	**10**
9	GND			+5V			GND	**9**
8	GND			GND			GND	**8**
7	GND						GND	**7**
6	GND						GND	**6**
5	GND						GND	**5**
4	GND			GND		IN2.8	GND	**4**
3	GND	IN2.5	IN2.2	IN1.7	IN1.4	IN1.1	GND	**3**
2	GND	IN2.6	IN2.3	IN1.8	IN1.5	IN1.2	GND	**2**
1	GND	IN2.7	IN2.4	IN2.1	IN1.6	IN1.3	GND	**1**

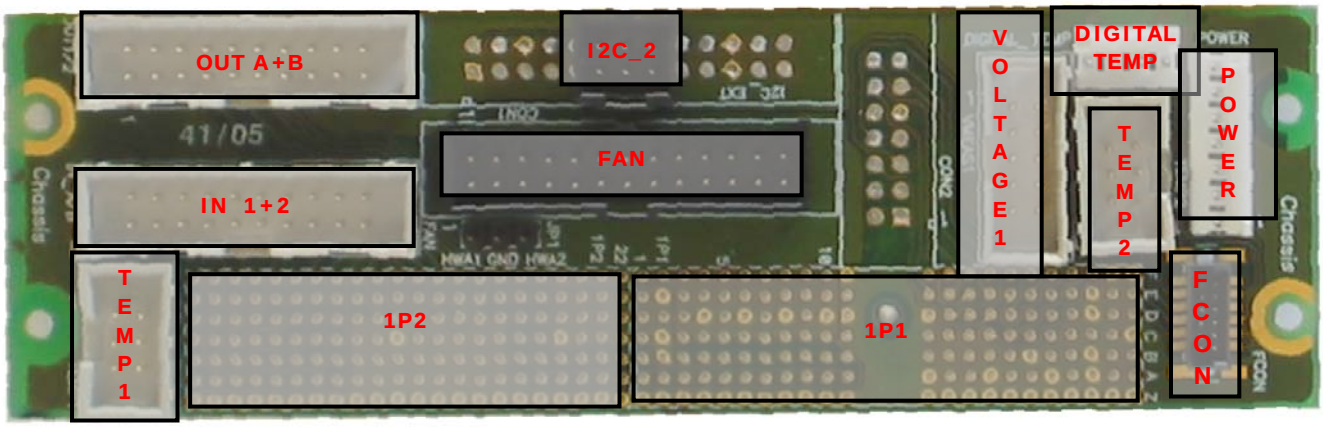
6.1. Description of the signals

Signal	Description
E0TP	Ethernet TX+
E0TM	Ethernet TX-
E0RP	Ethernet RX+
E0RM	Ethernet RX-
RS232_TX	RS232 Transmit
RS232_RX	RS232 Receive
TACH_0..11	TACH 1..11 inputs
PWM_0..3	PWM 0..3 outputs
TEMP 1..6	Analog temp sensors. The sensor must be connected between TEMP x and REF2.5
REF2.5	Temperature sensors reference. 2.5V (Temp x-)
SDA_0	I2C_0 data line. Not used
SCL_0	I2C_0 clock line. Not used
SDA_1	I2C_1 data line. Not used
SCL_1	I2C_1 clock line. Not used
SDA_2	I2C_2 data line. Used by the external LED display
SCL_2	I2C_2 clock line. Used by the external LED display
INT_2	I2C_2 interrupt line
SDA_3	I2C_3 data line. Used to connect I2C temperature sensors
SCL_3	I2C_3 clock line. Used to connect I2C temperature sensors
INT_3	I2C_3 interrupt line
PWRFAIL	Power fail. Input from power supply
RST	Used to connect an external reset button
ACFAIL#	VME – ACFAIL# ; CPCI – FAL#
SYSRESET#	VME – SYSRESET# ; CPCI – PRST#
V1 +5V	Voltage 1 monitor input
V2 +3.3V	Voltage 2 monitor input
V3 +12V	Voltage 3 monitor input
V4 -12V	Voltage 4 monitor input
V5 VIO	Voltage 5 monitor input
IN1.1..8	Input port 1
IN2.1..8	Input port 2
OUTA.1..8	Output port A
OUTB.1..8	Output port B
+5V	Monitor board power input. +5V
GND	Ground
+3.3V_OUT	+3.3 Voltage generated locally on the monitor board. Do not connect it with 3.3V on BP

6.2. Layout / position of the connectors

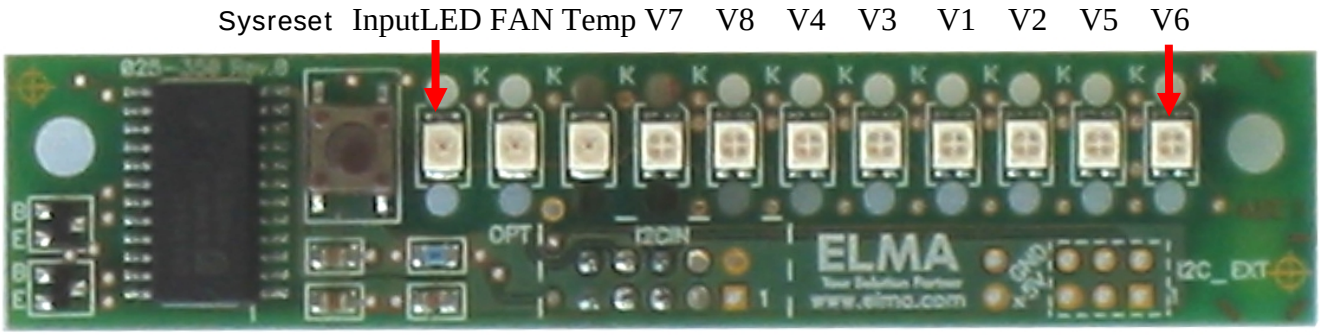


024-876: Assembly variants can differ from the illustration



Adapter card for 024-875: Assembly variants can differ from the illustration

024-874 and 024-875: Assembly variants can differ from the illustration

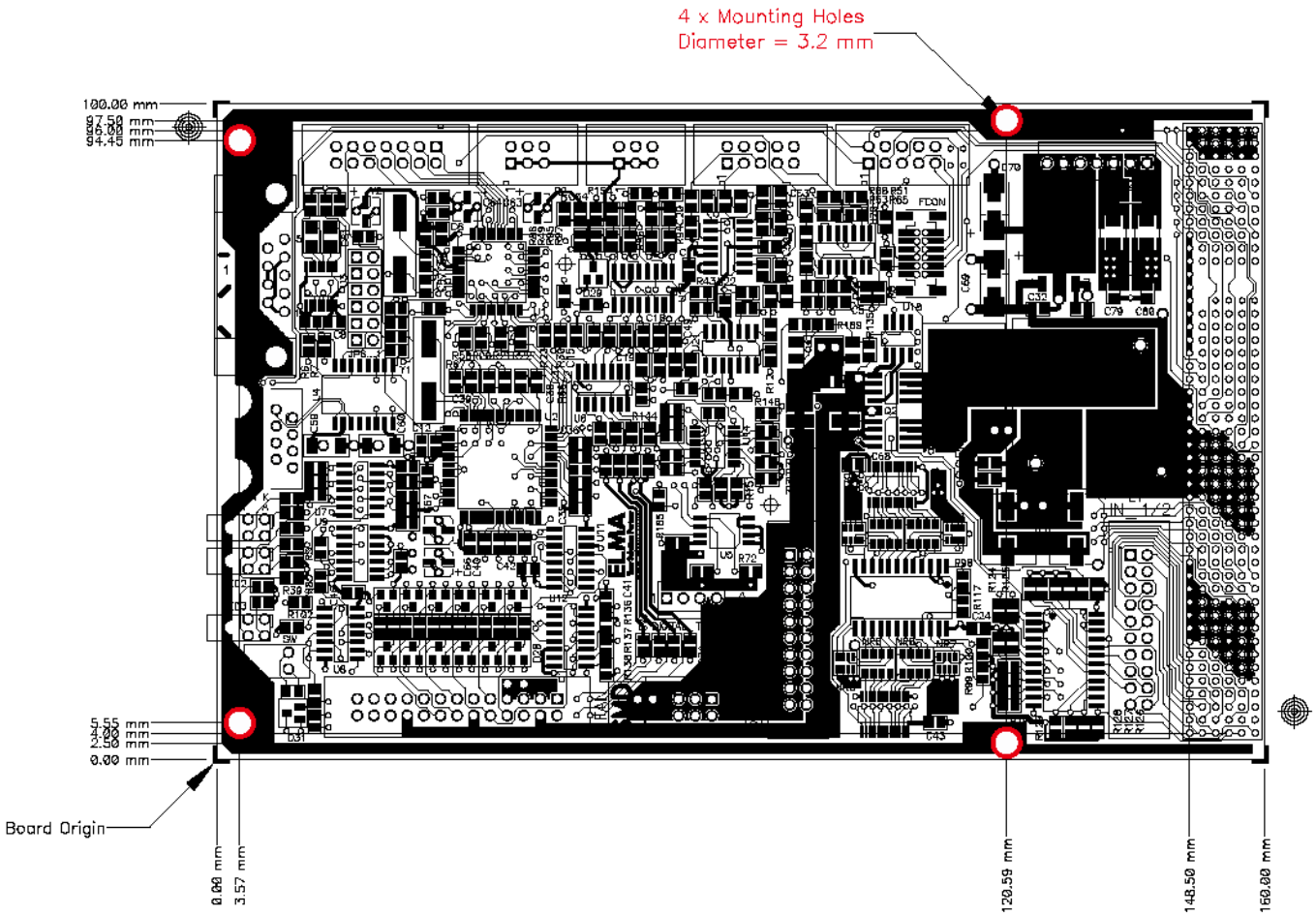


024-927: LED Display

7. Technical data / Dimensions

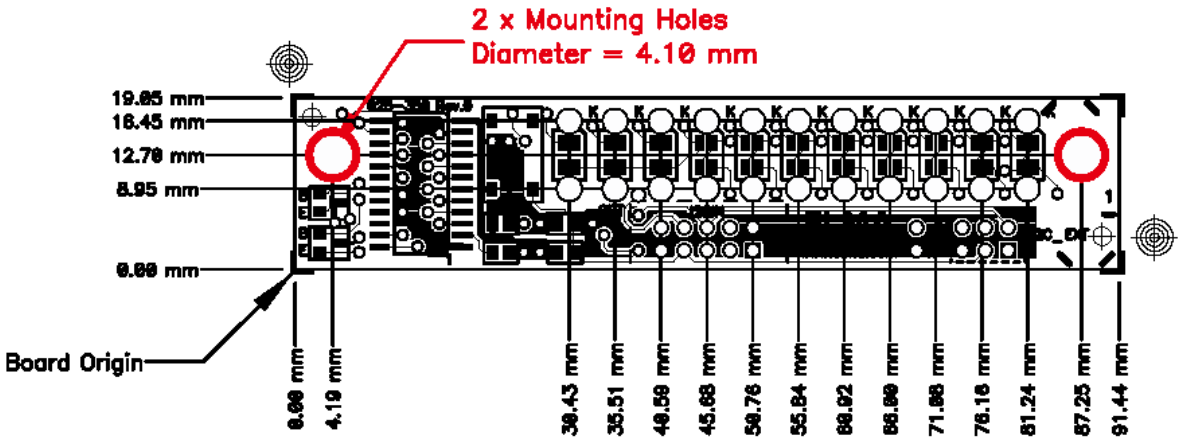
Power supply	+5V DC
Current consumption	500mA max.
Operating temperature	0°C to +70°C
Storage temperature	-40°C to +85°C
Physical dimensions	100.00 x 160.00 mm
Fan tacho signal requirement	open collector (NO TTL allowed!)
Load of PWM Signal	short-circuit-proof to GND. Internal pull-up resistors. **No external pull-up resistor allowed!**
PWM signal level	3.3V TTL
Signal level - inputs	5V TTL
Signal level - outputs	5V TTL
Max. loading - outputs	IOH: -32mA IOL: 64mA

VIEW FROM THE TOP (SMDs) SIDE



Dimensions SysMon Online*PRO*

VIEW FROM THE TOP (SMDs) SIDE



024-927: Dimensions LED Display

8. Order codes, product variants and accessories

Part number	Description
024-874	Plug-in version of the SysMon Online*PRO* with front panel. (a suitable matching connector for the SOP must be available on the backplane!)
024-875	Plug-in version (complete with 8 HP adaptor card) of the SysMon Online*PRO* with front panel.
024-876	Panel-mounting version of the SysMon Online*PRO* without front panel.
024-927	LED display for 024-876 complete with 0.8 m cable and matching connector (front plate not included in delivery).
024-915	Cable set consisting of: all matching connectors and terminals, cable and 3 analog temperature sensors for SysMon Online*PRO*.
025-092	FanCon Online*PRO* Module for controlling fans without a PWM signal input.

9. Add-on module: FanCon Online*PRO*

To connect fans with no PWM signal input, i.e. where the speed has to be controlled by varying the supply voltage, the FanCon Online*PRO* add-on module (Art. No. 025-092) can be connected between the Sysmon Online*PRO* and the fans. A PWM output of the SOP is connected to the Fancon Online*PRO* here. This generates the appropriate supply voltage for the fans from the PWM signal.

! However, the tacho signals of the fans are routed to the FAN connector of the SOP even when the Fancon Online*PRO* is used.

Please bear in mind the max. loading given in the technical data in section 9.2. FanCon OnlinePRO technical data / Dimensions! Also. a number of Fancon modules can be operated on the SOP, if necessary.

9.1. Fancon Online*PRO* pin assignment

X1

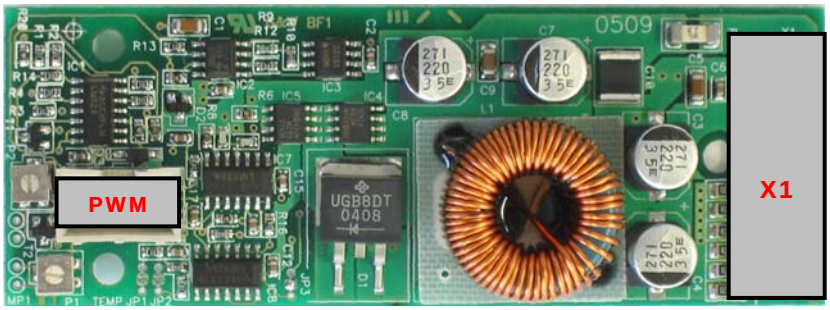
Pin	Signal
1	NC
2	-Fan3
3	-Fan2
4	-Fan1
5	+Fan3
6	+Fan2
7	+Fan1
8	Supply voltage GND
9	Supply voltage +12V

PWM

Pin	Signal
1	PWM in
2	GND
3	NC
4	NC
5	NC
6	NC

9.2. FanCon Online*PRO* technical data / Dimensions

Supply voltage	+12V DC
Current consumption (no-load operation)	100mA
Max. current consumption	4A
Number of fans (expandable through parallel connection)	3
Temperature range for speed regulation (fans)	+0°C to +50°C
Operating temperature	+0°C to +70°C
Storage temperature	-40°C to +85°C
Dimensions	104,00 x 38,00 mm
MTBF	143.673 hours



025-092: Assembly variants can differ from the illustration