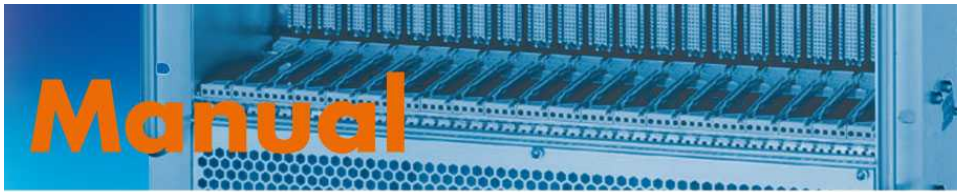


Sysmon Monitor Online*PRO* Handbuch 04.11.11 / 024-154 Rev. 6

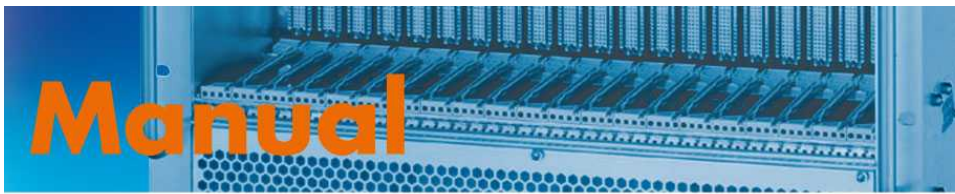
Software Version: 4.4

INHALT

- 1. Produktbeschreibung und Funktionen..... 3**
 - 1.1. Spannungsüberwachung..... 4
 - 1.2. Temperaturüberwachung..... 4
 - 1.3. Lüfterüberwachung und –regelung..... 5
 - 1.3.1. Temperatur / Drehzahlverhalten 5
 - 1.4. Digitale Ein- und Ausgänge 6
 - 1.4.1. Logische Pegel definieren..... 6
 - 1.4.2. Ausgänge manuell setzen / rücksetzen 6
 - 1.4.3. Ausgänge durch interne / externe Events steuern..... 7
 - 1.5. Generierung der VME-Signale ACFAIL und SYSRESET..... 8
- 2. Ethernet Interface..... 9**
- 3. Serielles RS232 Interface..... 9**
- 4. WEB Interface..... 10**
 - 4.1. Escape Codes für die Webseite 10
 - 4.2. Beispiel und HTML-Code 11
- 5. Command Line Interface (CLI)..... 13**
 - 5.1. Log-in..... 13
 - 5.2. Syntaxverzeichnis 14
 - 5.2.1. Generelle Syntaxkonventionen 14
 - 5.2.2. saveenv - Befehl (save environment)..... 14
 - 5.2.3. lanconfig - Befehl (lan configuration) 15
 - 5.2.4. voltage – Befehl 16
 - 5.2.5. port – Befehl..... 17
 - 5.2.6. temp - Befehl (temperature) 18
 - 5.2.7. fan – Befehl..... 19
 - 5.2.8. define – Befehle 20
 - 5.2.8.1. define activevoltage 20
 - 5.2.8.2. define voltagemask 20
 - 5.2.8.3. define voltagereset 20
 - 5.2.8.4. define powerfail..... 21
 - 5.2.8.5. define inputmask 21
 - 5.2.8.6. define vme..... 21
 - 5.2.8.7. define inputled 22
 - 5.2.8.8. define debouncetime 22
 - 5.2.8.9. define tempdebouncetime 22
 - 5.2.8.10. define startupdelay..... 23
 - 5.2.8.11. define tcptimeout 23
 - 5.2.8.12. define fanppr 24
 - 5.3. logout - Befehl..... 24
 - 5.4. uptime - Befehl 24
 - 5.5. xmodem - Befehl..... 25
 - 5.6. passw - Befehl 25
 - 5.7. reboot - Befehl..... 25
 - 5.8. info – Befehl 26
 - 5.9. watchdog - Befehl 26
 - 5.10. errors – Befehl 26
- 6. Pinbelegungen der Steckverbinder..... 28**
 - 6.1. Beschreibung der Signale 30
 - 6.2. Layout / Position der Steckverbinder 31
- 7. Technische Daten / Abmessungen 33**



8. Bestellschlüssel, Produktvarianten und Zubehör..... 34
9. Zusatzmodul: FanCon OnlinePRO 35
9.1. Pinbelegung Fancon OnlinePRO 35
9.2. Technische Daten / Abmessungen FanCon OnlinePRO 35



1. Produktbeschreibung und Funktionen

Der Sysmon Online*PRO* (SOP) ist eine plattformunabhängige Systemüberwachungseinheit, welche die Parameter eines Grundsystems wie Spannungen, Temperaturen, digitale Eingänge und Lüfterdrehzahlen überwacht und ggf. steuert. Hierzu dient ein 16-bit Microcontroller mit einem integrierten 12-bit A/D Wandler und entsprechender peripherer Beschaltung.

In regelmäßigem Zyklus werden diese Parameter gemessen bzw. eingelesen und ausgewertet. Wird ein vom Anwender definierter Grenzwert über- oder unterschritten wird dies vom SOP als „Event“ erkannt. Grundsätzlich werden bei den entsprechenden Events fest vordefinierte Aktionen ausgeführt die später beschrieben werden. Diese „Events“ können zusätzlich vom Anwender zum Setzen von digitalen Ausgängen verwendet werden.

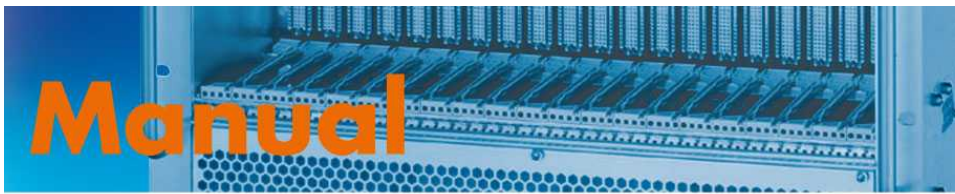
Aufgetretene Grenzwertüber- oder unterschreitungen für die Spannungen V1 bis V4, die Temperatur und die Lüftermindestdrehzahl werden über die Frontplatten LED's signalisiert. Ist die Anzeige von mehr als 4 Spannungen erwünscht, kann das ELMA LED Display 024-927 als Zubehör an den SOP angeschlossen und an beliebiger Stelle im System montiert werden.

Die gemessenen Werte sind jederzeit über die serielle RS232 Schnittstelle und über Telnet abrufbar. Grenzwerte und Systemparameter können über diese Schnittstellen ebenfalls jederzeit im Betrieb verändert werden. Dadurch kann der SOP – und damit das angeschlossene System - online über jeden Rechner mit Internetanbindung gesteuert und überwacht werden.

Auch die integrierte Webseite, die vom Anwender selbst gestaltet werden kann wird mit den gemessenen Werten fortlaufend aktualisiert und rundet damit das Informationsspektrum ab.

Bei der Verwendung in VME-Systemen übernimmt der SOP die Erzeugung VME Signale "ACFAIL2" und "SYSRESET2".

Die Grenzwerte des Grundsystems sowie die verschiedenen Parameter können jederzeit, auch im laufenden Betrieb, eingestellt werden.



1.1. Spannungsüberwachung

- Überwachung von bis zu 8 Spannungen (Standard: 5 x pos. und 3 x neg.)
- Obere und untere Grenzwerte sind für jede Spannung einzeln konfigurierbar
- Die Überwachung von einzelnen Spannungen kann deaktiviert werden
- Status LED auf der Frontplatte (nur für 024-874 und 024-875) für die Spannungen V1 - V4 (+5V, +3.3V, +12V, -12V)
- Die Grenzwertüber- bzw. unterschreitung einer der überwachten Spannungen ist ein interner Event (global error – siehe 5.2.8. define – Befehle), der durch den port Befehl (siehe 5.2.5. port – Befehl) weiter verarbeitet werden kann.
- Optional kann ein SYSRESET für VME Systeme im Fehlerfall ausgelöst werden

- Überwachte Spannungsbereiche der jeweiligen Eingänge:

V1= 0V..+8V V2= 0V..+8V
V3= 0V..+14V V4= -14V.. 0V
V5= 0V..+8V V6= -8V.. 0V
V7= -26V.. 0V V8= 0V..+26V

- Befehle für die Spannungsüberwachung:

voltage (siehe 5.2.4. voltage – Befehl)

define activevoltage (siehe 5.2.8.1. define activevoltage)

define voltagemask (siehe 5.2.8.2. define voltagemask)

define voltagereset (siehe 5.2.8.3. define voltagereset)

1.2. Temperaturüberwachung

- Überwachung von bis zu 6 analogen Temperatursensoren (10K Ohm NTC Thermistoren mit $\beta=3950$)
- Obere und untere Grenzwerte frei konfigurierbar
- Temperaturmessbereich von $-20^{\circ}\text{C}..+100^{\circ}\text{C}$
- Genauigkeit der Temperaturmessung $\pm 3^{\circ}\text{C}$ (max.)
- Status LED auf der Frontplatte zur Temperaturfehleranzeige (nur für 024-874 und 024-875)
- Überwachung von weiteren 8 digitalen Temperatursensoren über den I²C Bus.
- Im Fehlerfall wird die gelbe temp-fail LED an der Frontplatte aktiviert und die Maximaldrehzahl der Lüfter so lange gesetzt bis kein temp-fail mehr vorliegt.
- Die Grenzwertüber- bzw. unterschreitung einer der überwachten Temperaturen ist ein interner Event (global error), der durch den **port** Befehl (siehe 5.2.5. port – Befehl) weiter verarbeitet werden kann.

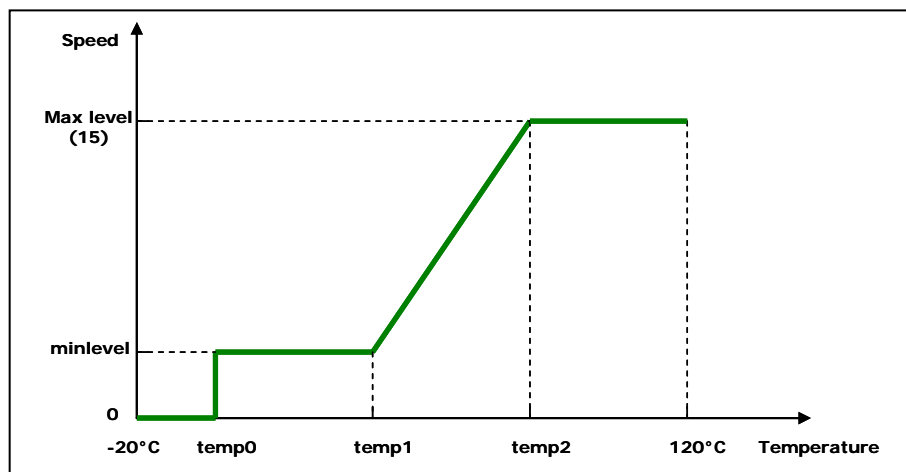
- Befehl für die Temperaturüberwachung:

temp (siehe 5.2.6. temp - Befehl (temperature))

1.3. Lüfterüberwachung und –regelung

- Überwachung und Steuerung von bis zu 12 Lüftern. Die Lüfter benötigen einen Tacho-Signalausgang sowie einen PWM-Signaleingang
- Minimale Drehzahl konfigurierbar (>700rpm)
- Drehzahlregelung über PWM Signal (Frequenz konfigurierbar von 20Hz..10KHz)
- Ansteuerung von Non-PWM Lüftern mit Zusatzmodul FanCon OnlinePRO möglich (siehe 9. Zusatzmodul: FanCon OnlinePRO)
- LED auf der Frontplatte zur Anzeige von Lüfterfehlern
- Softstart – zur Minimierung des Anlaufstroms
- Die Unterschreitung der Minstdrehzahl (**minspeed**) einer der überwachten Lüfter ist ein interner Event (global error), der durch den **port** Befehl (siehe 5.2.5. port – Befehl) weiter verarbeitet werden kann.
- Im Fehlerfall wird die gelbe fan-fail LED an der Frontplatte aktiviert und die Maximaldrehzahl der Lüfter so lange gesetzt bis kein fan-fail mehr vorliegt.
- Frei konfigurierbares Temperatur / Drehzahlverhalten über die Einstellung von 4 Parametern (minlevel, temp0, temp1, temp2). Einstellungen sind dabei einheitlich für alle Temperatursensoren.

1.3.1. Temperatur / Drehzahlverhalten

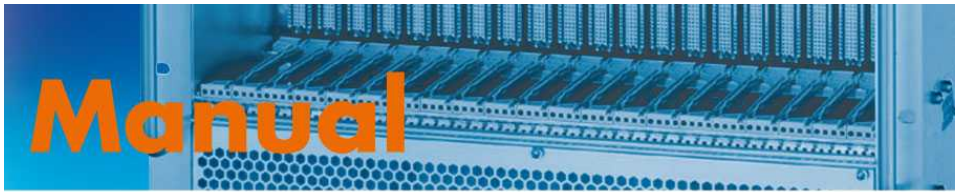


Die Lüfterdrehzahl wird in Abhängigkeit der Temperatur geregelt. Der Drehzahlbereich der Lüfter wird in 15 Stufen eingeteilt. Durch das Diagramm ergeben sich 4 Regelungsbereiche:

1. -20°C..temp0: Lüfter stehen
2. temp0..temp1: Lüfter laufen konstant auf zuvor festgelegtem Min. Pegel (minlevel)
3. temp1..temp2: Lüfter verändern ihre Drehzahl proportional zur Temperatur
4. temp2..120°C: Lüfter laufen mit voller Geschwindigkeit auf Level 15

- Befehl für die Lüfterregelung und zur Einstellung der Parameter:

fan (siehe 5.2.7. fan – Befehl)



1.4. Digitale Ein- und Ausgänge

- 16 digitale Eingänge gruppiert in zwei 8 Bit Ports (Port 1 + Port 2)
- 16 Ausgänge gruppiert in zwei 8 Bit Ports (Port A + Port B)
- Logikpegel: 5V TTL. Ausgänge mit 20mA gegen GND belastbar
- Jeder einzelne Ein/Ausgang kann als low- oder high-aktiv deklariert werden (siehe 1.4.1. Logische Pegel definieren)
- Manuelles setzen und rücksetzen von Ausgängen (siehe 1.4.2. Ausgänge manuell setzen / rücksetzen)
- Ausgänge können abhängig von internen und externen Events automatisch gesetzt werden. (siehe 1.4.3. Ausgänge durch interne / externe Events steuern)

1.4.1. Logische Pegel definieren

Um losgelöst von Spannungspegeln arbeiten zu können werden die Zustände der Ein- und Ausgänge als logische Pegel betrachtet. Hierzu muss vorab definiert werden ob die Signale Low- oder Highaktiv sind. Hierzu wird der Befehl „**port ... level**“ verwendet. (siehe 5.2.5. port – Befehl)

- Befehl um logische Pegel zu definieren:
port...level (siehe 5.2.5. port – Befehl)

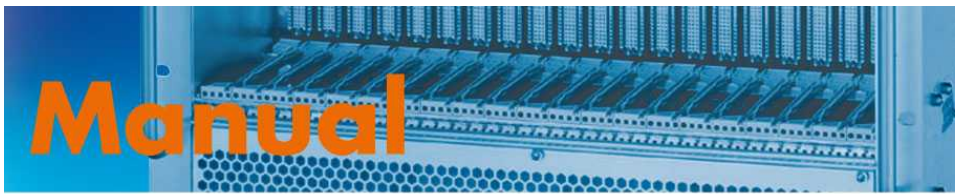
	1	2	3	4	5	6	7	8	Hex Value
Port x active level	0	0	1	0	1	0	1	1	2B

1.4.2. Ausgänge manuell setzen / rücksetzen

Mit dem Befehl „**port ... set**“ bzw. „**port ... clr**“ können die Zustände der Ausgänge verändert werden. Mit „set“ wird der angesprochene Ausgang aktiviert – welcher Spannungspegel dann am Ausgang anliegt bestimmt die zuvor festgelegte Definition seines logischen Pegels (siehe 1.4.1. Logische Pegel definieren).

Anmerkung: Diese Funktion kann im ADMIN Zugang durch setzen des entsprechenden Bits in der Maske des Ausganges deaktiviert werden (siehe 1.4.3. Ausgänge durch interne / externe Events steuern)

- Befehle um Ausgänge zu aktivieren / deaktivieren:
port...set (siehe 5.2.5. port – Befehl)
port...clr (siehe 5.2.5. port – Befehl)



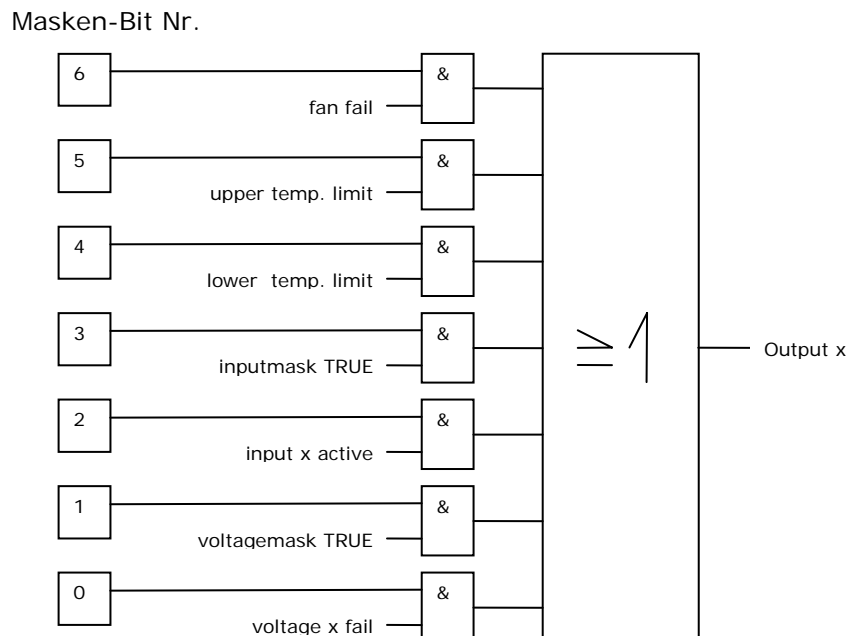
1.4.3. Ausgänge durch interne / externe Events steuern

Um Ausgänge in Abhängigkeit von internen Events (z.B. Temperaturfehler, Lüfterfehler) oder externen Events (aktivierte Eingänge) zu steuern kann eine Bitmaske für jeden einzelnen Ausgang erstellt werden. Ist die „ODER“ Verknüpfung der einzelnen Bits dieser Maske = 1 (TRUE) wird der Ausgang aktiviert. Welcher Spannungspegel dann am Ausgang anliegt bestimmt die zuvor festgelegte Definition seines logischen Pegels (siehe 1.4.1. Logische Pegel definieren). Sobald die „ODER“ Verknüpfung =0 (FALSE) ist, wird der Ausgang wieder zurückgesetzt.

Maskenaufbau für Ausgang Nr. „x“:

Bit Nr.	Beschreibung
7	1= Freigabe „port...set“ und „port...clr“ für den Ausgang „x“ (siehe 1.4.2. Ausgänge manuell setzen / rücksetzen)
6	Event „Lüfterfehler“
5	Event „Oberes Temperaturlimit überschritten“
4	Event „Unteres Temperaturlimit unterschritten“
3	Event „inputmask TRUE“ (siehe 5.2.8.5. define inputmask)
2	Event „Eingang x aktiv“ (Korrespondierender Eingang aktiv)
1	Event „voltage mask TRUE“ (siehe 5.2.8.2. define voltage mask)
0	Event „Spannung x außerhalb der Grenzwerte“

Logikplan:



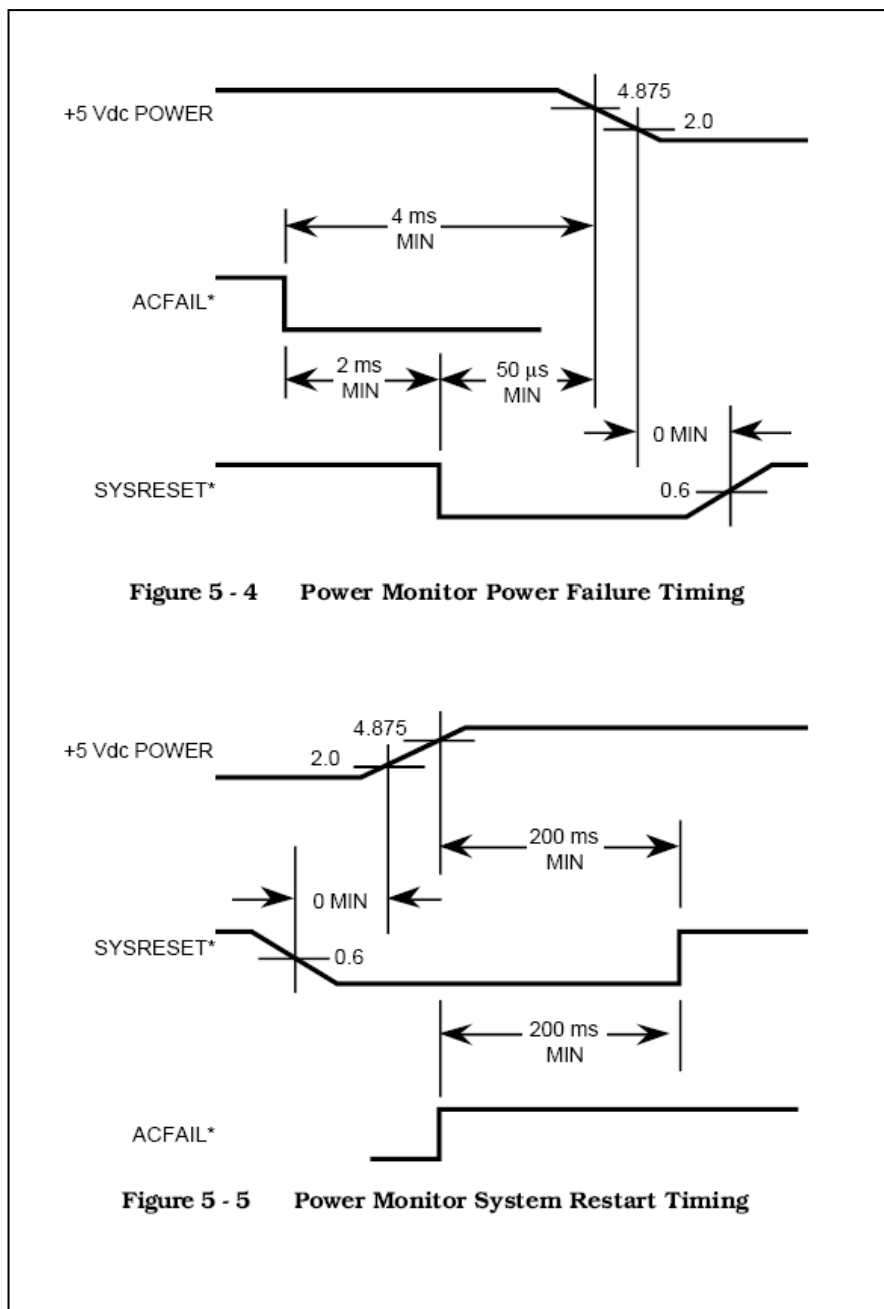
- Befehl um die Masken der Ausgänge zu setzen:

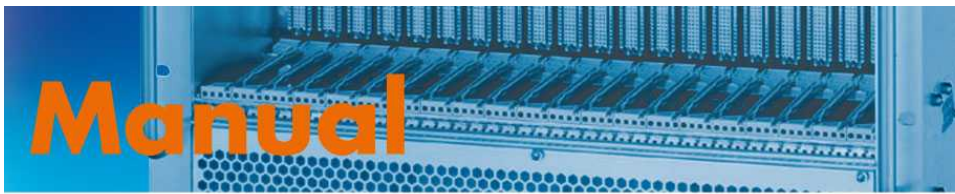
port...mask (siehe 5.2.5. port – Befehl)

1.5. Generierung der VME-Signale ACFAIL und SYSRESET

Der SOP generiert die Signale ACFAIL und SYSRESET gemäß VME Spezifikation ANSI/VITA 1-1994. Damit diese Funktion korrekt ausgeführt wird, muss der SOP über die Befehle „**define vme**“ (siehe 5.2.6. temp - Befehl (temperature)) und „**define powerfail**“ (siehe 5.2.8.4. define powerfail) entsprechend eingerichtet werden. Die Signale können am Steckverbinder „Voltage1“ abgegriffen werden.

Timing Diagramm aus der VME-Spezifikation ANSI/VITA 1-1994:





2. Ethernet Interface

Durch das integrierte 10Mbps Ethernet Interface kann der SOP an jedes existierende Netzwerk eingebunden werden. Das Interface unterstützt die Protokolle HTTP und TELNET über TCP/IP. Somit können alle überwachten Systemparameter über einen Standard Browser angezeigt werden (HTTP-Protokoll). Über TELNET hat man vollen Zugriff auf die Befehle des Command Line Interface (CLI) und kann somit Parameter lesen und den Sysmon konfigurieren. Durch den Einsatz dieser Standard-Protokolle entfällt die Verwendung von spezieller Software oder Treiber und man erreicht damit eine Plattformunabhängigkeit. Das TCP/IP Protokoll unterstützt 10 gleichzeitige Verbindungen – die maximale Paketgröße ist auf 1k begrenzt.

Auf Windows-Systemen empfehlen wir als TELNET-Client den Einsatz von „Hyperterminal“ oder „PuTTY“.



Im Auslieferungszustand lautet die IP des Sysmon 193.155.166.51 - Die IP Adresse muss für den Einsatz in Ihrem Netzwerk über den Befehl **lanconfig** (siehe 5.2.3. lanconfig - Befehl (lan configuration)) angepasst werden.

Terminal Einstellungen:

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



Verwenden Sie ein gekreuztes Netzkabel, wenn Sie den Sysmon direkt an einen PC anschließen!

3. Serielles RS232 Interface

Der SOP bietet ein serielles RS232 Interface über das die Befehle des Command Line Interface (CLI) gesendet werden können.



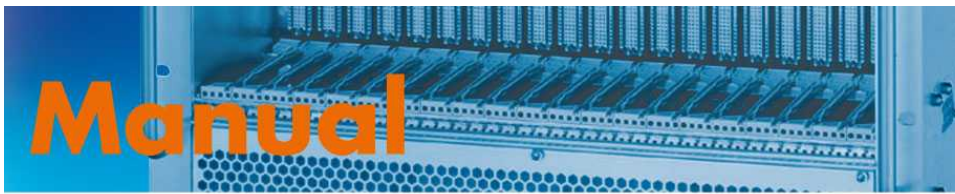
Verwenden Sie ein NULL-MODEM Kabel für den direkten Anschluss an die serielle Schnittstelle eines PC's.

Auf Windows-Systemen empfehlen wir den Einsatz von „Hyperterminal“ als Terminal Programm.

Terminal Einstellungen:

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

Über die serielle Schnittstelle kann zusätzlich der CLI-Befehl **xmodem** zum Dateitransfer verwendet werden. (siehe 5.5. xmodem - Befehl)



4. WEB Interface

Der Sysmon bietet zur Fernabfrage mittels eines Standard Browsers eine Webseite, die vom Anwender selbst erstellt und über den Befehl „**xmodem web**“ (siehe 5.5. xmodem - Befehl) gespeichert werden kann. Die maximale Dateigröße der HTML Datei beträgt 10kbyte. Es ist nur das Speichern einer HTML Datei möglich. Werden Bilder verwendet, müssen diese extern abgelegt werden.

Folgende Werte können über die Webseite angezeigt werden:

- Spannungswerte V1-V8
- Statusfarben (rot/grün/grau) der Spannungen V1-V8
- Temperaturwerte T1-T14
- Statusfarben (gelb/grau) der Temperaturen T1-T14
- Lüfterdrehzahlen F1-F12
- Statusfarben (gelb/grau) der Lüfter F1-F12

In der erstellten HTML Datei müssen an den Stellen, an denen die o.g. Werte / Farben dargestellt werden sollen entsprechende Escape Codes (siehe 4.1. Escape Codes für die Webseite) eingesetzt werden. Der auf dem Sysmon integrierte Parser ersetzt fortlaufend diese Codes durch die aktuellen Werte.

4.1. Escape Codes für die Webseite

Nachfolgende Escape Codes werden vom Sysmon in der gespeicherten HTML Datei fortlaufend durch die aktuellen Werte ersetzt:

Wert der Spannung Nr. x (Voltage x value): **~VxV**

Statusfarbe der Spannung Nr. x (Voltage x color): **~VxC**

Gültige Werte für x: **1,2,3,4,5,6,7,8**

Wert der Temperatur Nr. x (Temperature x value): **~TxV**

Statusfarbe der Temperatur NR. x (Temperature x color): **~TxC**

Gültige Werte für x: **1,2,3,4,5,6,7,8,9,a,b,c,d,e (Hex Wert)**

Drehzahl des Lüfters Nr. x (Fan x speed): **~FxV**

Statusfarbe des Lüfters Nr. x (Fan x color): **~FxC**

Gültige Werte für x: **1,2,3,4,5,6,7,8,9,a,b,c (Hex Wert)**

- Befehl zur Übertragung der Webseite:

xmodem web (siehe 5.5. xmodem - Befehl)

4.2. Beispiel und HTML-Code



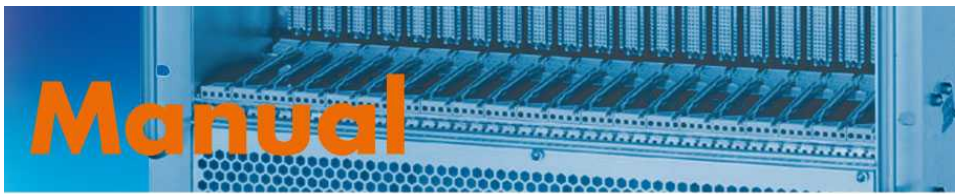
HTML-Code :

(Wichtig: Es können KEINE Bilder auf dem SOP gespeichert werden. Wenn Sie Bilder verwenden möchten müssen diese extern abgelegt und die URL im HTML Code entsprechende eingegeben werden!).

```
<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)

Das Command Line Interface (kurz: CLI) ist sowohl über Telnet (siehe 2. Ethernet Interface) als auch über die serielle RS232 Schnittstelle (siehe 3. Serielles RS232 Interface) verfügbar.

Über das CLI kann der Anwender Systemparameter lesen oder neu konfigurieren und speichern. Der Zugriff ist aufgeteilt in 2 Profile und ist passwortgeschützt.

„user“ Profil:

Systemparameter können in diesem Profil nur gelesen werden – ausgenommen vom Schreibschutz ist der Befehl **lanconfig** (siehe 5.2.3. lanconfig - Befehl (lan configuration)) zum Einstellen der IP-, Subnet- und Gateway- Adresse. Wurden im admin-Profil die Ausgänge entsprechend konfiguriert (siehe 1.4.3. Ausgänge durch interne / externe Events steuern), kann der user einzelne Ausgänge auch manuell setzen und rücksetzen (siehe 1.4.2. Ausgänge manuell setzen / rücksetzen).

„admin“ Profil:

! Vollzugriff auf alle Systemparameter. Alle verfügbaren Befehle des CLI können ausgeführt werden. Die Zugangsdaten für dieses Profil dürfen nur geschultem Personal mit entsprechenden Kenntnissen und Kompetenzen für das System in das der SOP verbaut wurde bekannt sein um Schaden oder Fehlfunktionen zu vermeiden!

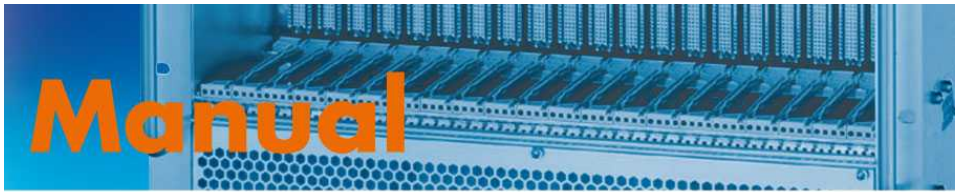
5.1. Log-in

Sobald Sie eine Verbindung über Telnet oder RS232 hergestellt haben, werden Sie zum login aufgefordert. Evtl. müssen Sie einmalig ENTER drücken um zum login zu gelangen. Die gemessenen Werte können jederzeit durch RS232 Schnittstelle oder Telnet ausgelesen werden. Weiterhin können die Grenzwerte des Grundsystems sowie die verschiedenen Parameter jederzeit, auch im laufenden Betrieb, eingestellt werden.

Voreingestellte Zugänge:

login: user
password: USER

login: admin
password: ADMIN



5.2. Syntaxverzeichnis

5.2.1. Generelle Syntaxkonventionen

Befehl [parameter1 | parameter2 | parameter3 [Wert]]

[] = optional

Ein Befehl alleine ohne Angabe von weiteren Parametern gibt alle zum Befehl zugehörigen und zur Verfügung stehenden aktuellen Werte zurück.

Wird EIN Parameter angegeben, jedoch kein Wert gibt der SOP nur den aktuellen Wert der zu diesem Parameter zur Verfügung steht zurück.

Ist ein *Wert* in der Befehlszeile vorhanden, wird dieser dem entsprechenden Parameter zugeordnet und temporär im RAM gespeichert. Die Änderung ist sofort aktiv. Soll der Wert endgültig in das Flash-ROM gespeichert werden und auch nach dem reboot gültig sein, müssen die Umgebungsvariablen mit dem Befehl „**saveenv**“ (siehe 5.2.2. saveenv - Befehl (save environment)) gespeichert werden. Änderungen ohne saveenv gehen nach einem reboot verloren.

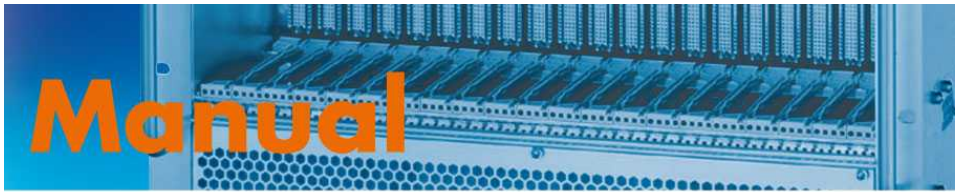
5.2.2. saveenv - Befehl (save environment)

Syntax:

saveenv

Funktion:

Speichert alle im RAM vorhandenen Umgebungsvariablen dauerhaft ins Flash-ROM.



5.2.3. lanconfig - Befehl (lan configuration)

Syntax:

lanconfig [ip | mask| gate [address]]

Funktionen:

Auslesen oder setzen von Netzwerk Parametern.

- **kein parameter** – Rückgabe von IP, Mask und Gateway Adressen des LAN interface
- **ip** – IP Adresse des Sysmons
- **mask** – Netzwerk Maske
- **gateway** – Standard Gateway
- **address** – IP Adresse des SOP (IP bei Auslieferung ist 193.155.166.51)

! Nach dem Setzen einer neuen Adresse muss mit **saveenv** die Änderung gespeichert und der SOP neu gestartet werden. Entweder mit dem Befehl **reboot** (siehe 5.2.7. fan – Befehl) oder per Reset-Taster.

Beispiele:

- Auslesen aller Netzwerk Parameter

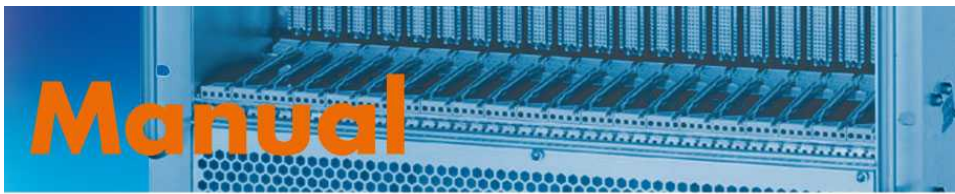
```
%>lanconfig  
IP=193.155.166.51  
Mask=255.255.255.0  
Gateway=193.155.166.100
```

- Auslesen der IP Adresse

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

- Ändern der IP Adresse

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



5.2.4. voltage – Befehl

Syntax:

```
voltage [sysmon | min | max] [voltage number[value]]
```

Funktionen:

Auslesen der Spannungswerte oder setzen der einzelnen Grenzwerte zur Spannungsüberwachung.

- **sysmon** = Ausgabe der SOP Betriebsspannung (keine weiteren Parameter zulässig)
- **min** = unterer Grenzwert
- **max** = oberer Grenzwert
- **voltage number** = Nr. des Analogeingangs 1-8
- **value** = Grenzwert

Beispiele:

- Auslesen aller Spannungswerte. Nicht angeschlossene Spannungen werden mit dem Wert „NA“ dargestellt.

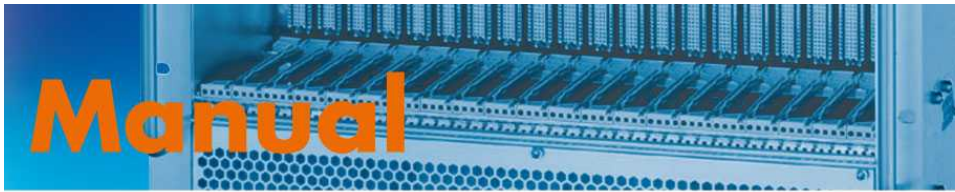
```
%>voltage  
V1=4.958 V2=3.304 V3=12.132 V4=NA V5=NA V6=NA V7=NA V8=NA
```

- Auslesen des unteren Grenzwerts der Spannung V1 (+5V)

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

- Setzen des oberen Grenzwerts der Spannung V1 auf 5.5V

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

5.2.5. port – Befehl

Syntax:

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

Funktionen:

Auslesen bzw. ändern der Port-Zustände, der logischen Pegel und Zugriff auf Port-Masken.

- **port_name** = 1,2,A,B (1,2 = Eingänge ; A,B = Ausgänge)
- **level** = Auslesen oder setzen der logischen Pegel. Bei nachfolgender Angabe von *value* wird ein 8-bit Wert erwartet. Jedes bit steht für den entsprechenden Ein- bzw. Ausgang
- **mask output** = Nur für Port A,B (Ausgänge). Auslesen oder setzen der Maske für den angesprochenen Ausgang (*output*) Bei nachfolgender Angabe von *value* wird ein 8-bit Wert erwartet. Jedes bit steht für den entsprechenden Ein- bzw. Ausgang.
- **set output** = Nur für Port A,B (Ausgänge). Angesprochenen Ausgang (*output*) aktivieren.
- **clr output** = Nur für Port A,B (Ausgänge). Angesprochenen Ausgang (*output*) deaktivieren.
- **value** = 8-bit Wert in HEX Schreibweise (0x00 bis 0xff) oder dezimal (0 bis 255).
- **output** = Angabe des Ausgangs A oder B

Die Bitstruktur für Masken finden Sie in Kapitel 1.4.3. Ausgänge durch interne / externe Events steuern.

Beispiele:

- Ausgabe der Zustände aller Ports

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

- Setze die Eingänge 1-4 des Ports 1 auf "Low Aktiv" (bzw. "high wenn inaktiv") und Eingänge 5-8 auf „High Aktiv“ (bzw. „low wenn inaktiv“)

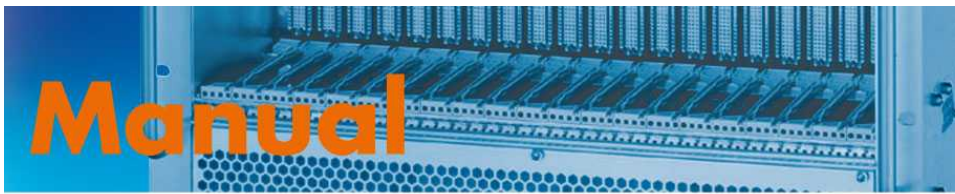
```
%>port 1 level 0x0f
Level_Port0=0x0F
```

- Maske für Ausgang 1 des Port A so setzen, dass dieser Ausgang aktiviert wird, sobald Spannung Nr.1 außerhalb der Grenzwerte ist. Gleichzeitig soll der Ausgang für **port set** und **port clr** freigegeben werden.

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

- Aktiviere Ausgang 1 des Port A

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



5.2.6. temp - Befehl (temperature)

Syntax:

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

Ausgabe der aktuellen Temperaturwerte bzw. Grenzwerte oder setzen von Temperaturgrenzwerten und Sensoranzahlen.

Einstellungen sind dabei einheitlich für alle Temperatursensoren.

- **min**= unterer Grenzwert
- **max**= oberer Grenzwert
- **analogno**= Anzahl analoger Sensoren
- **digitalno**= Anzahl digitaler I²C Sensoren
- **value** = Grenzwert in °C oder Wert für die Anzahl der Sensoren

Beispiele:

- Ausgabe aller verfügbaren aktuellen Temperaturmesswerte

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

- Ausgabe des unteren Temperaturgrenzwerts

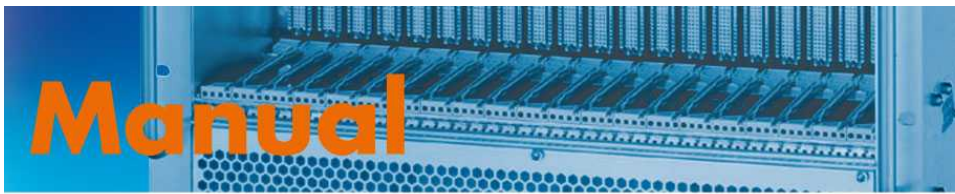
```
%>temp min  
Temp min =-10
```

- Setzen des oberen Grenzwerts auf 70°C

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

- Anzahl der analogen Sensoren auf 4 einstellen

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



5.2.7. fan – Befehl

Syntax:

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

Funktionen:

Ausgabe oder Konfiguration der Lüfterparameter.

- **minspeed** = Auslesen/setzen der Mindestdrehzahl
- **minlevel** = Auslesen/setzen des minlevel Werts
- **temp0** = Auslesen/setzen des Werts temp0
- **temp1** = Auslesen/setzen des Werts temp1
- **temp2** = Auslesen/setzen des Werts temp2
- **pwmfreq** = Auslesen/setzen der PWM Frequenz
- **level** = Auslesen des aktuellen Levels auf dem die Lüfter laufen
- **fanno** = Auslesen/setzen der Lüfteranzahl
- **boostlevel** = Auslesen/setzen des aktuellen Start-Levels der Lüfter
- **hysteresis** = Auslesen/setzen der Hysterese für die Temperaturregelung (Wert: 0-20)
- *value* = zugehöriger Wert für die einzelnen Parameter

Die Bedeutung der Parameter werden in Kapitel 1.3. Lüfterüberwachung und –regelung ausführlich erklärt.

Beispiele:

- Setze die Anzahl der Lüfter auf 3

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

- Auslesen der aktuellen Lüfterdrehzahlen

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

- Setze die Mindestdrehzahl auf 700 (Unter dieser Drehzahl wird ein FANFAIL erzeugt).

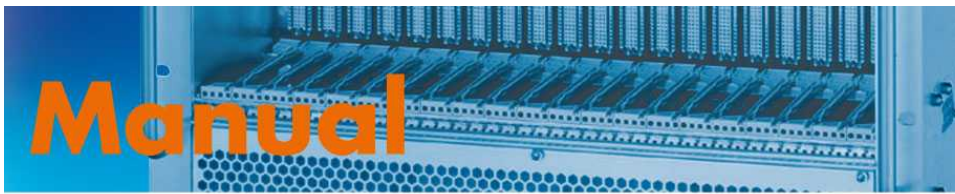
```
%>fan minspeed 700  
Min speed=700
```

- Auslesen des Parameters minlevel

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

- Setzen des Parameters temp0 auf 0°C (Unter diesem Wert stoppen die Lüfter)

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



5.2.8. define – Befehle

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*]

Beispiel:

0x0F = V1 bis V4 werden überwacht

Funktion:

Ausgabe oder setzen der zu überwachenden Spannungen

- **value** = 8-bit Wert in HEX Schreibweise (0x00 bis 0xff) oder dezimal (0 bis 255). Jedes Bit steht für die jeweilige Spannung.

5.2.8.2. define voltagemask

Syntax:

define voltagemask [*value*]

Funktion:

Ausgabe oder setzen der Maske zur Überwachung mehrerer Spannungen und Erzeugung eines internen Events.

- **value** = 8-bit Wert in HEX Schreibweise (0x00 bis 0xff) oder dezimal (0 bis 255). Jedes Bit steht für die jeweilige Spannung.

Sind die alle entsprechend gesetzten Spannungsnummern außerhalb der Grenzwerte ist das Ergebnis dieser Maske TRUE und kann mit dem **port** Befehl weiter verarbeitet werden (siehe 1.4.3. Ausgänge durch interne / externe Events steuern)

5.2.8.3. define voltagereset

Syntax:

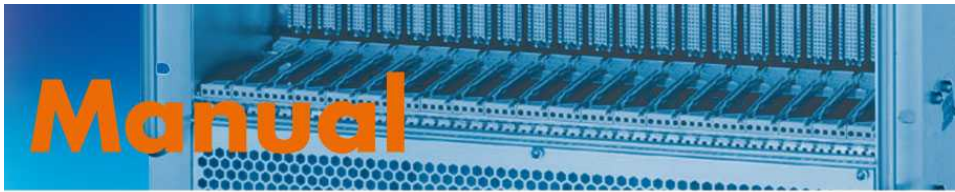
define voltagereset [*Wert*]

Funktion:

Ausgabe oder setzen der Maske zur Überwachung mehrerer Spannungen welche einen SYSRESET auslösen soll.

- **value** = 8-bit Wert in HEX Schreibweise (0x00 bis 0xff) oder dezimal (0 bis 255). Jedes Bit steht für die jeweilige Spannung.

Sind die alle entsprechend gesetzten Spannungsnummern außerhalb der Grenzwerte ist das Ergebnis dieser Maske TRUE und es wird ein SYSRESET ausgelöst.



5.2.8.4. define powerfail

Syntax:

define powerfail [*value*]

Funktion:

Ausgabe oder setzen des logischen Pegels vom angeschlossenen Signal powerfail am POWER Steckverbinder. Ist Powerfail aktiv und das VME flag auf „1“ wird das VME Timing wie in Kapitel 1.4.3. Ausgänge durch interne / externe Events steuern beschrieben ausgelöst.

- **value** = Dezimalwert 0,1 (0 = low aktiv ; 1 = high aktiv)

5.2.8.5. define inputmask

Syntax:

define inputmask [*value*]

Funktion:

Ausgabe oder setzen der Maske zur Überwachung mehrerer Eingänge und Erzeugung eines internen Events.

- **value** = 16-bit Wert in HEX Schreibweise (0x0000 bis 0xffff) oder dezimal (0 bis 65535). Jedes Bit steht für den jeweiligen Eingang.

Sind die alle entsprechend gesetzten Eingänge aktiv ist das Ergebnis dieser Maske TRUE und kann mit dem **port** Befehl weiter verarbeitet werden (siehe 1.4.3. Ausgänge durch interne / externe Events steuern)

5.2.8.6. define vme

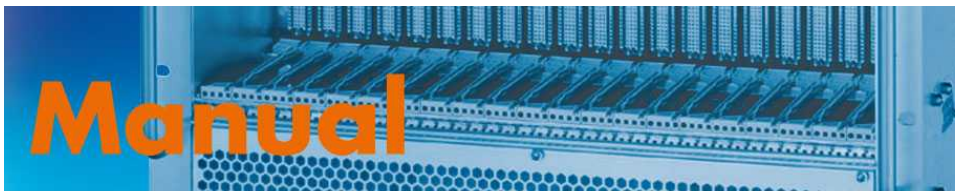
Syntax:

define vme [*value*]

Funktion:

Ausgabe oder setzen des Flags ob der SOP an ein VME System angeschlossen ist und somit das VME Timing wie in Kapitel 1.5. Generierung der VME-Signale ACFAIL und SYSRESET beschrieben durchführen soll.

- **value** = Dezimalwert 0,1 (0 = kein VME System ; 1 = VME System)



5.2.8.7. define inputled

Syntax:

define inputled [*value*]

Funktion:

Setzen der Maske über die Eingänge die zur Aktivierung der InputLED auf dem LED-Display 024-927 führt.

- **value** = 16-bit Wert in HEX Schreibweise (0x0000 bis 0xffff) oder dezimal (0 bis 65535). Jedes Bit steht für den jeweiligen Eingang.

Ist einer entsprechend gesetzten Eingänge aktiv ist das Ergebnis dieser Maske TRUE und die InputLED wird aktiviert.

5.2.8.8. define debouncetime

Syntax:

define debouncetime [*value*]

Funktion:

Setzen der Entprellzeit der Eingänge

- **value** = dezimal (0 bis 15) 0=0ms ; 1=100ms ; 15=1.5s

Beispiel:

- Setzen der Entprellzeit auf 500ms

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

5.2.8.9. define tempdebouncetime

Syntax:

define tempdebouncetime [*value*]

Funktion:

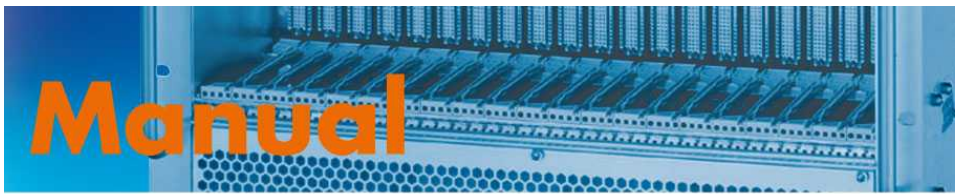
Setzen der Entprellzeit für die Temperaturmessung

- **value** = dezimal (0 bis 30) 0=0ms ; 1=100ms ; 30=3s

Beispiel:

- Setzen der Entprellzeit auf 2s

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



5.2.8.10. define startupdelay

Syntax:

define startupdelay [*value*]

Funktion:

Setzen eines Timeouts nach dem Start bis der Systemmonitor mit Überwachung beginnt.

- **value** = dezimal (0 bis 15) 0=0ms ; 1=100ms ; 15= 1.5s

Beispiel:

- Setzen der Verzögerung auf 300ms

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

5.2.8.11. define tcptimeout

Syntax:

define tcptimeout [*value*]

Funktion:

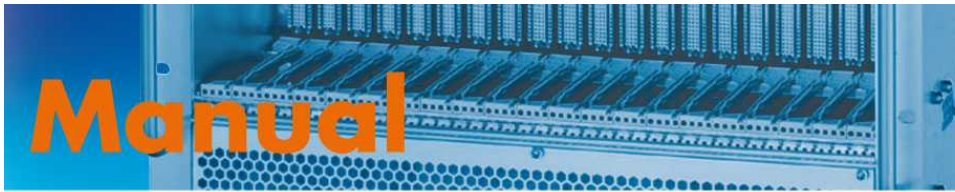
Setzen eines Timeouts nach dem eine TCP Verbindung (im Leerlauf) für eine neue Verbindung freigemacht wird.

- **value** = dezimal (0 bis 36000) 1=0,1s ; 36000= 1h
Ist der Wert 0, dann werden keine neuen Verbindungen erlaubt, falls alle TCP Verbindungen belegt sind.

Beispiel:

- Setzen des Timeouts auf 30min

```
%>define tcptimeout 18000  
TCP timeout=18000
```



5.2.8.12. define fanppr

Syntax:

define fanppr [*value*]

Funktion:

Einstellen der Anzahl Tachoimpulse pro Umdrehung

- **value** = dezimal (2 , 4) 2=2 Impulse pro Umdrehung ; 4=4 Impulse pro Umdrehung

5.3. logout - Befehl

Syntax:

logout

Funktion:

Aktuelle Sitzung beenden.

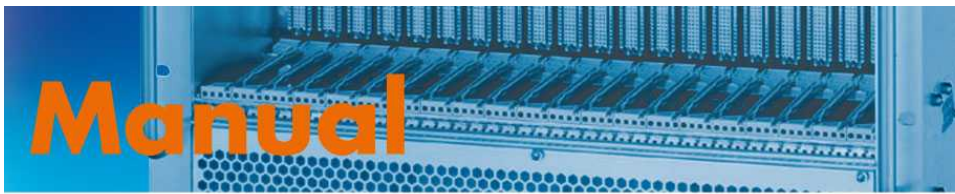
5.4. uptime - Befehl

Syntax:

uptime

Funktion:

Anzeige der Zeit seit dem letzten reboot.



5.5. xmodem - Befehl

Syntax:



xmodem [config | web] → **Nur über RS232!**

Funktion:

Übertragen von Dateien mit XMODEM Protokoll – nur über RS232 verfügbar.

- **config** – Start der Übertragung einer binären Konfigurationsdatei
- **web** – Start der Übertragung einer HTML Datei

Nach Eingabe des Befehls geht der SOP auf Datenempfang und wartet auf das Senden der Daten im XMODEM Protokoll. Dies meldet der SOP mit fortlaufenden C's auf dem Terminal. Starten Sie dann umgehend mit ihrem Terminalprogramm den Dateitransfer und wählen sie als Protokoll XMODEM aus.

Beispiel:

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

5.6. passw - Befehl

Syntax:

passw

Funktion:

Ändern des Passworts im aktuellen Profil.

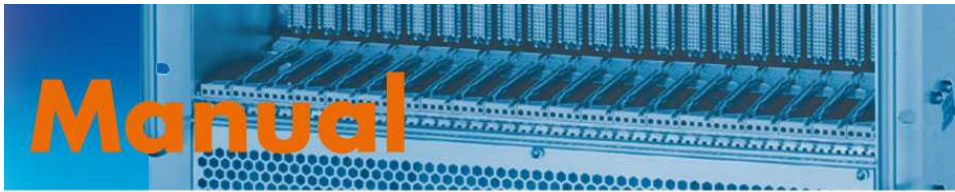
5.7. reboot - Befehl

Syntax:

reboot

Funktion:

Sysmon neu starten (RESET)



5.8. info – Befehl

Syntax:

info

Funktion:

Ausgabe der MAC-Adresse, Seriennummer und der Firmware Version

Beispiel:

```
%>info
MAC=00:14:DB:00:04:C1
Serial No=10505681
Version=4.2
```

5.9. watchdog - Befehl

Syntax:

watchdog [*value*]

Funktion:

Mit diesem Befehl kann der Watchdog ein- bzw. ausgeschaltet werden.

- *value* = on / off

Beispiel:

- Watchdog einschalten

```
%>watchdog on
Watchdog ON
```

5.10. errors – Befehl

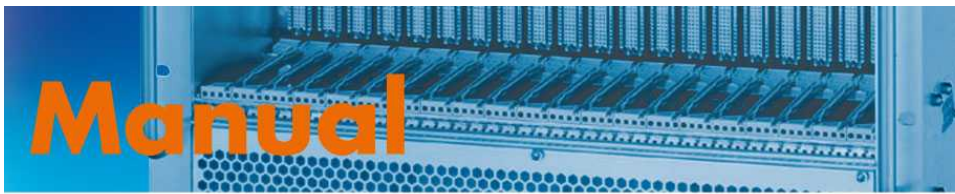
Syntax:

errors [*clr*]

Funktion:

Ausgabe oder löschen der gespeicherten Fehlerstati (global errors)
und Eingangsevents (input errors)

Der SOP speichert bei jedem Event (Fan Fail, Voltage Fail, usw.) und bei Aktivierung von
Eingängen diese Events und legt sie in den Registern **global errors** und **input errors** ab.



Wenn der Fehler bzw. Event nicht mehr aktuell ist, bleibt er dennoch in diesen Registern gespeichert bis mit **errors clr** der Fehlerspeicher wieder gelöscht wird.

- **ohne parameter** – Rückgabe der Werte im 16-bit HEX Format (0x0000 bis 0xffff)
- **clr** – global errors und input errors löschen

Aufbau Register „global errors“:

Bit	Beschreibung
0	1 = Spannung Nr.1 außerhalb der Grenzwerte
1	1 = Spannung Nr.2 außerhalb der Grenzwerte
2	1 = Spannung Nr.3 außerhalb der Grenzwerte
3	1 = Spannung Nr.4 außerhalb der Grenzwerte
4	1 = Spannung Nr.5 außerhalb der Grenzwerte
5	1 = Spannung Nr.6 außerhalb der Grenzwerte
6	1 = Spannung Nr.7 außerhalb der Grenzwerte
7	1 = Spannung Nr.8 außerhalb der Grenzwerte
8	1 = Lüfterfehler (FANFAIL) lag vor
9	1 = Unterer Temperaturgrenzwert unterschritten
10	1 = Oberer Temperaturgrenzwert überschritten
11	1 = PowerFail Signal wurde erkannt
12	Intern
13	Intern
14	Intern
15	Intern

Aufbau Register „input errors“:

Bit	Beschreibung
0	1 = Eingang Nr.1 war aktiv
1	1 = Eingang Nr.2 war aktiv
2	1 = Eingang Nr.3 war aktiv
3	1 = Eingang Nr.4 war aktiv
4	1 = Eingang Nr.5 war aktiv
5	1 = Eingang Nr.6 war aktiv
6	1 = Eingang Nr.7 war aktiv
7	1 = Eingang Nr.8 war aktiv
8	1 = Eingang Nr.9 war aktiv
9	1 = Eingang Nr.10 war aktiv
10	1 = Eingang Nr.11 war aktiv
11	1 = Eingang Nr.12 war aktiv
12	1 = Eingang Nr.13 war aktiv
13	1 = Eingang Nr.14 war aktiv
14	1 = Eingang Nr.15 war aktiv
15	1 = Eingang Nr.16 war aktiv

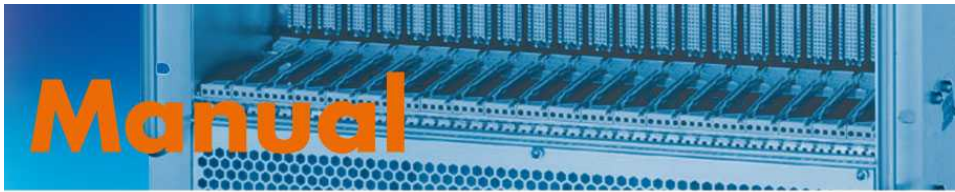
Beispiel:

- Ausgabe der gespeicherten Fehlerstati / Eingangsevents

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

- Löschen der gespeicherten Fehlerstatu / Eingangsevents

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



6. Pinbelegungen der Steckverbinder

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

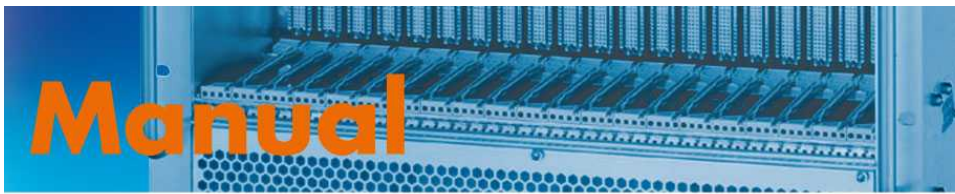
Pin	Signal
1	+5V
2	SDA_3
3	GND
4	SCL_3

1P1

	Z	A	B	C	D	E	F	
25	GND	E0TP	EORP	GND			GND	25
24	GND	E0TM	EORM				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

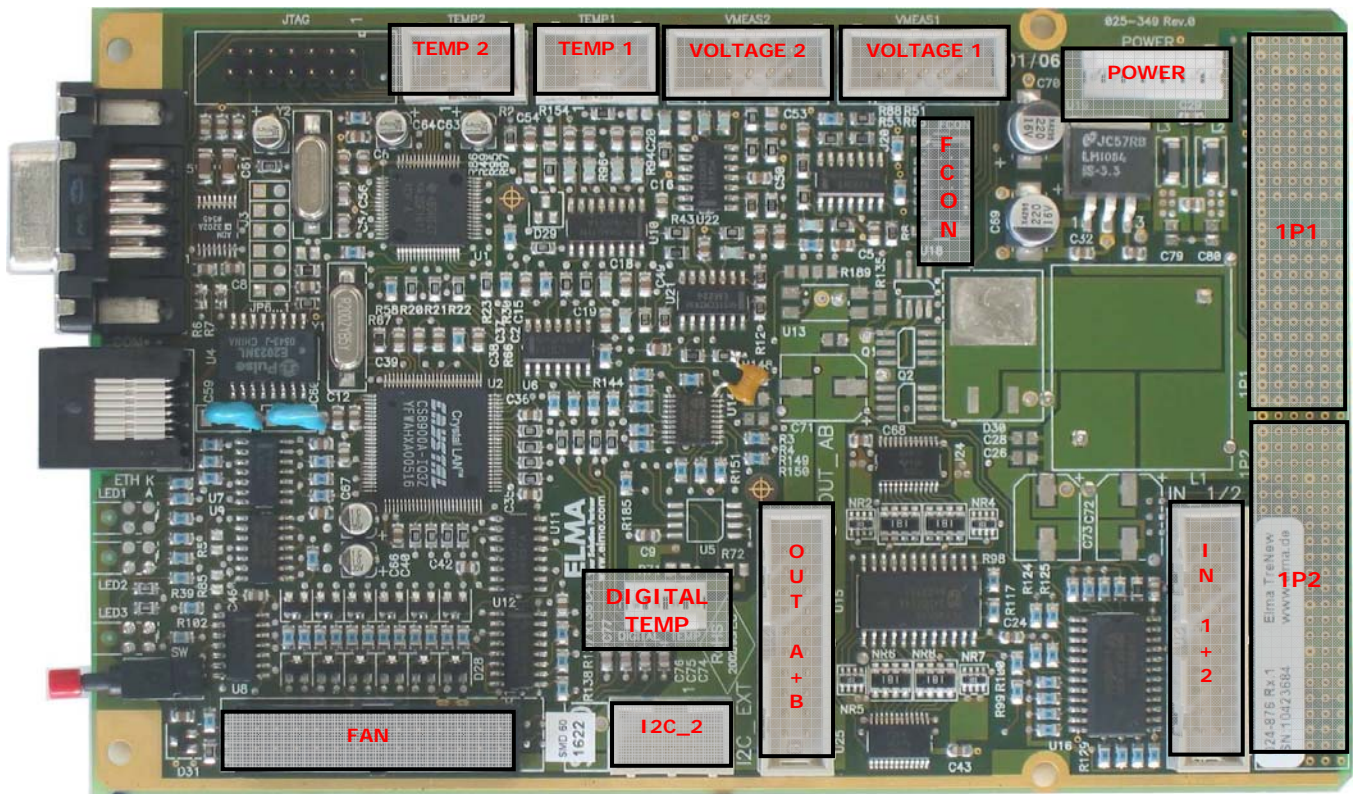
	Z	A	B	C	D	E	F	
22	GND						GND	22
21	GND			+5V			GND	21
20	GND						GND	20
19	GND			GND			GND	19
18	GND						GND	18
17	GND						GND	17
16	GND						GND	16
15	GND				OUTB.2	OUTB.1	GND	15
14	GND			+5V	OUTA.8	OUTA.7	GND	14
13	GND	OUTB.8	OUTB.7		OUTA.6	OUTA.5	GND	13
12	GND	OUTB.6	OUTB.5		OUTA.4	OUTA.3	GND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



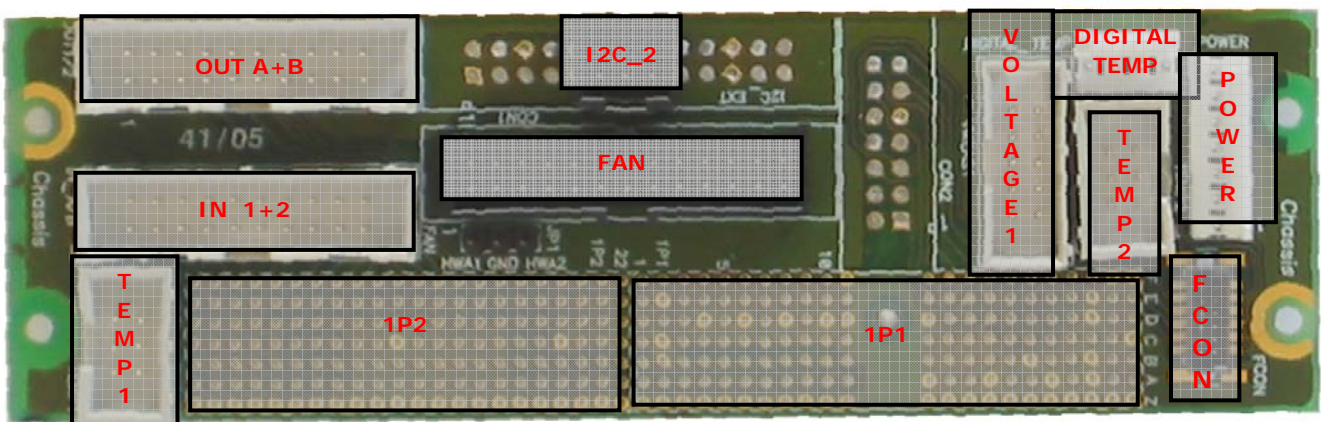
6.1. Beschreibung der Signale

Signal	Beschreibung
EOTP	Ethernet TX+
EOTM	Ethernet TX-
EORP	Ethernet RX+
EORM	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 der Steckverbinder



024-876: Bestückungsvarianten können von Abbildung abweichen

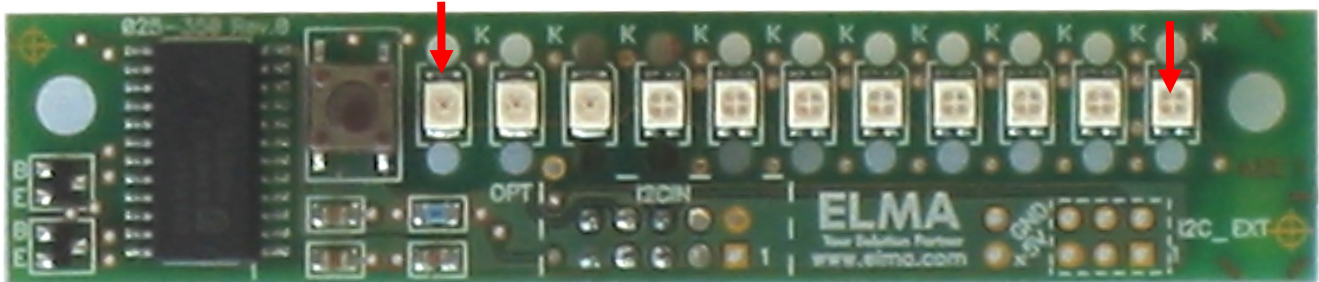


Adapter card für 024-875: Bestückungsvarianten können von Abbildung abweichen



024-874: Bestückungsvarianten können von Abbildung abweichen

Sysreset InputLED FAN Temp V7 V8 V4 V3 V1 V2 V5 V6



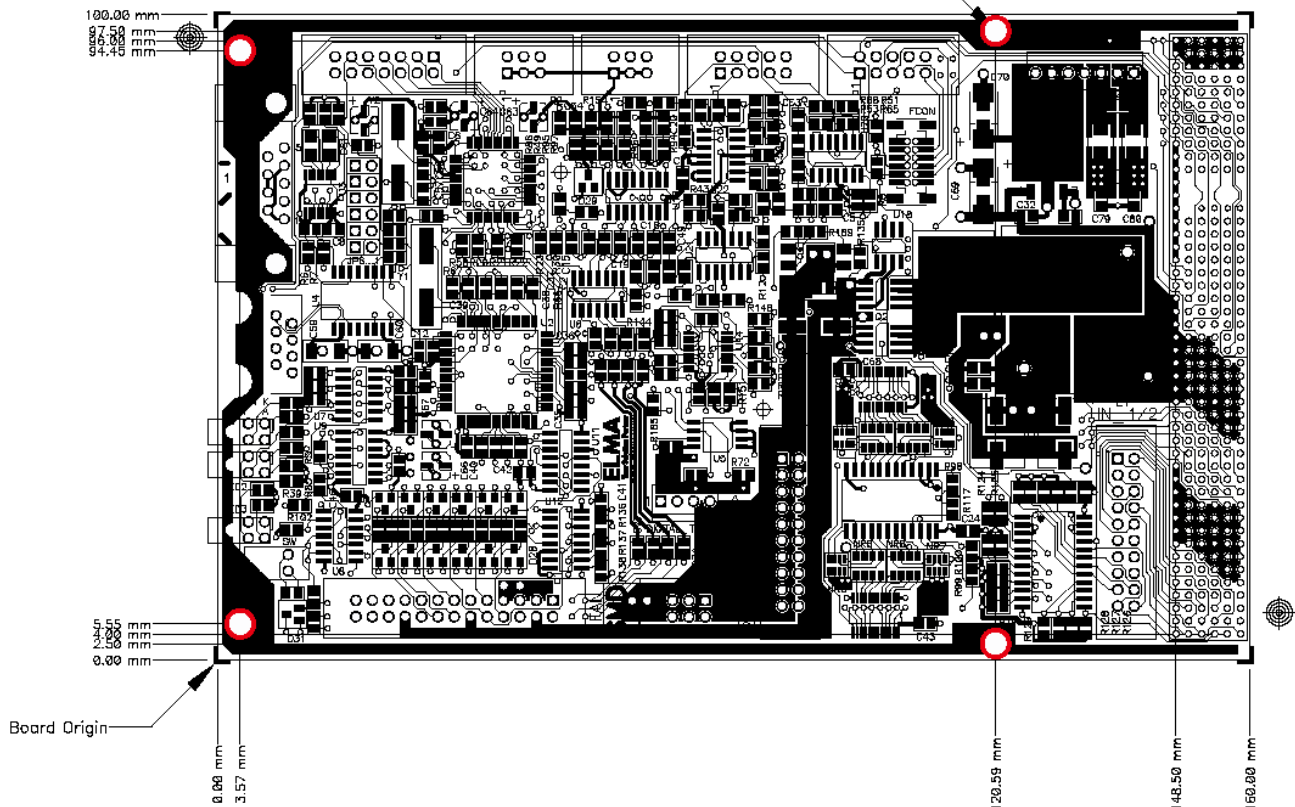
024-927: LED Display

7. Technische Daten / Abmessungen

Versorgungsspannung	+5V DC
Stromaufnahme	500mA max.
Betriebstemperatur	-20°C to +70°C
Lagertemperatur	-40°C to +85°C
Mechanische Abmessungen	100,00 x 160,00 mm
Anforderung an Lüfter-Tachosignale	Open collector (Kein TTL zugelassen !)
Pegel PWM Signal	3.3V TTL
Belastbarkeit PWM Signal	Kurzschlussfest nach GND. Interne Pull-up Widerstände. Keine externen Pull-up Widerstände zugelassen !
Pegel Eingänge	5V TTL
Pegel Ausgänge	5V TTL
Belastbarkeit Ausgänge	IOH: -32mA IOL: 64mA

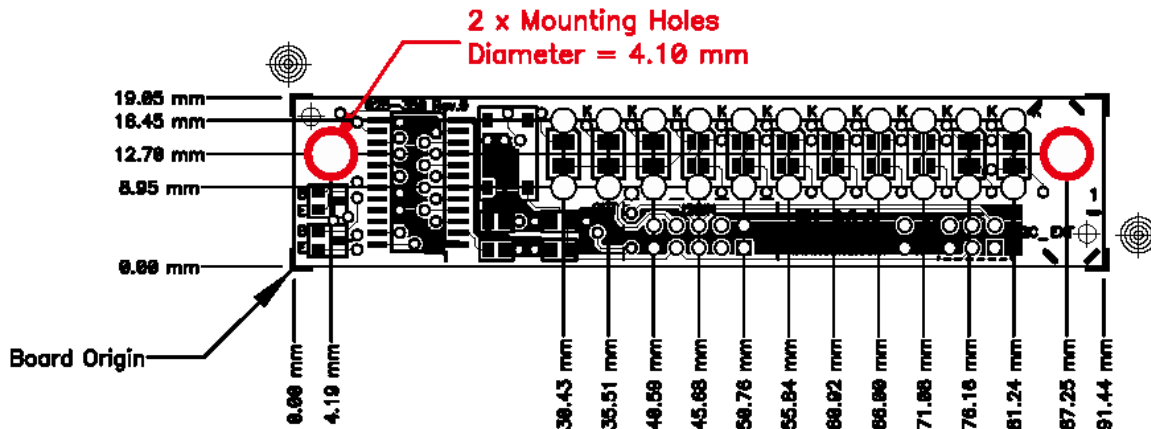
VIEW FROM THE TOP (SMDs) SIDE

4 x Mounting Holes
Diameter = 3.2 mm



Abmessungen SysMon OnlinePRO

VIEW FROM THE TOP (SMDs) SIDE



024-927: Abmessungen LED Display

8. Bestellschlüssel, Produktvarianten und Zubehör

Artikelnummer	Beschreibung
024-874	Steckbare Version des SysMon Online <i>PRO</i> mit Frontplatte. (Ein entsprechender Gegenstecker für den SOP muss auf der Backplane vorhanden sein!)
024-875	Steckbare Version (inkl. 8 TE Adapter Karte) des SysMon Online <i>PRO</i> mit Frontplatte.
024-876	Montierbare Version des SysMon Online <i>PRO</i> ohne Frontplatte.
024-927	LED Display für 024-876 inkl. 0.8 m Kabel und Gegenstecker (Frontplatte ist nicht im Lieferumfang enthalten).
024-915	Kabelsatz bestehend aus: allen Gegensteckern und Kontakten, Kabel und 3 analogen Temperatursensoren für SysMon Online <i>PRO</i>
025-092	FanCon Online <i>PRO</i> Modul zur Ansteuerung von Lüftern ohne PWM Signaleingang.

9. Zusatzmodul: FanCon OnlinePRO

Zum Anschluss von Lüftern ohne PWM Signaleingang – deren Drehzahl also durch Veränderung der Versorgungsspannung geregelt werden muss – kann das Zusatzmodul FanCon OnlinePRO (Art. Nr. 025-092) zwischen Sysmon OnlinePRO und den Lüftern geschaltet werden. Hierzu wird ein PWM-Ausgang des SOP an den Fancon OnlinePRO angeschlossen. Dieser generiert aus dem PWM Signal die entsprechende Versorgungsspannung für die Lüfter.



Die Tachosignale der Lüfter werden jedoch auch bei Verwendung des Fancon OnlinePRO an den FAN-Steckverbinder des SOP geführt!

Beachten Sie bitte die max. Belastung in den technischen Daten in Kapitel 9.2. Technische Daten / Abmessungen FanCon OnlinePRO! Es können auch mehrere FanCon's am SOP betrieben werden, falls notwendig.

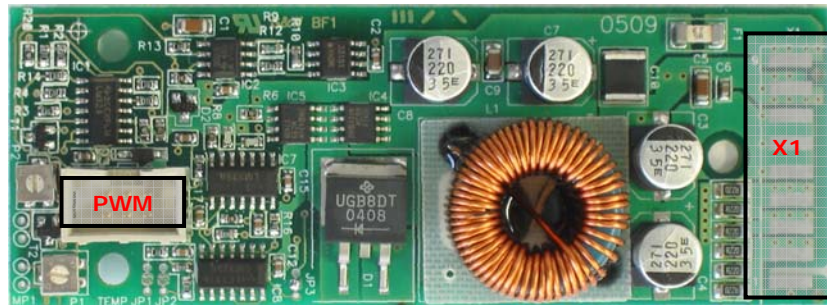
9.1. Pinbelegung Fancon OnlinePRO

X1	
Pin	Signal
1	offen
2	-Fan3
3	-Fan2
4	-Fan1
5	+Fan3
6	+Fan2
7	+Fan1
8	Versorgung GND
9	Versorgung +12V

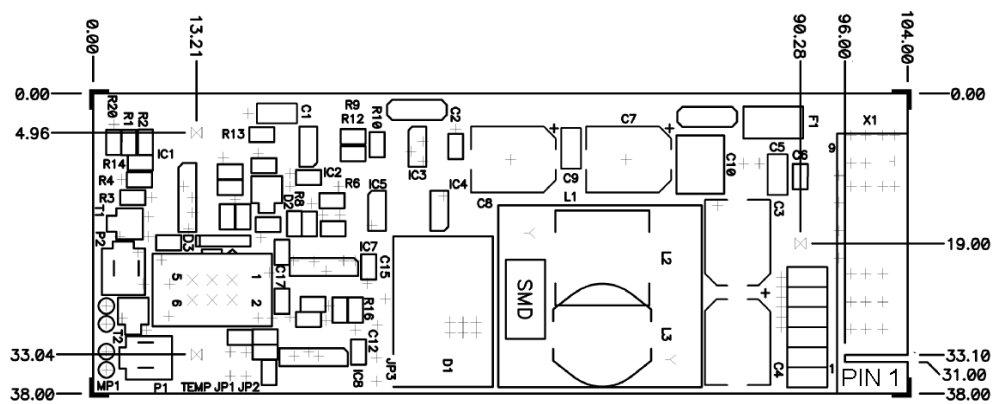
PWM	
Pin	Signal
1	PWM in
2	GND
3	offen
4	offen
5	offen
6	offen

9.2. Technische Daten / Abmessungen FanCon OnlinePRO

Versorgungsspannung	+12V DC
Stromverbrauch (im Leerlauf)	100mA
max. Stromaufnahme	4A
Anzahl Lüfteranschluss (erweiterbar durch Parallelschaltung)	3
Drehzahlregelung der Lüfter im Temperaturbereich	+0°C bis +50°C
Betriebstemperatur	+0°C bis +70°C
Lagertemperatur	-40°C bis +85°C
Abmessungen	104,00 x 38,00 mm
MTBF	143.673 Stunden



025-092: Bestückungsvarianten können von Abbildung abweichen



Irrtümer und technische Änderungen vorbehalten. Copyright ELMA Electronic GmbH.