



Application Note

Interpreting Alarms on the CPC7301 – ISM

March 29, 2006

Performance Technologies
205 Indigo Creek Drive
Rochester, NY 14626
585.256.0200
support@pt.com

www.pt.com

© 2005 Performance Technologies, Inc.
All Rights Reserved.



Disclaimer

This document presents information for users of Performance Technologies Inc. products. Although the information contained within this document is considered accurate and characteristic of the subject product, Performance Technologies Inc. reserves the right to make changes to this document and any products described herein to improve reliability, functionality, or design. Performance Technologies Inc. does not assume any liability arising out of the application or use of any product or circuit described herein. No part of this document may be copied or reproduced in any form or by any means without the prior permission of Performance Technologies Inc.

Trademarks

This document is the sole property of Performance Technologies Inc. All other brands or names are trademarks of their respective holders.

© Copyright 2006 Performance Technologies, Inc.

All rights reserved. Printed in the United States of America.

March 29, 2006



Contents

Chapter 1: Overview 5

Introduction.....	5
Objective.....	5
Alarm LEDs.....	5
Understanding and Interpreting SEL Entries.....	6
Critical Alarms.....	7
Major Alarms.....	8
Minor Alarms.....	9
Other Considerations.....	10





Chapter 1: Overview

Introduction

There are three alarm LEDs on the front panel of the CPC7301 Intelligent Shelf Manager: major, minor, and critical. There are also multiple sensors within a typical system across the many slot components, such as power supplies, single board computers, fans, etc. These sensors monitor key elements of a chassis: temperature, voltage, and board presence, among others.

The LEDs indicate the presence of any event-triggered alarms that are configurable by the user based on sensor feedback. By default, several of these event-triggered alarms are already configured. These alarms, in conjunction with other system management tools, can be used to determine system health. These alarms/tools may also be used to set up varying levels of monitoring and control over the system depending upon the application.

Objective

This document will outline the steps necessary to determine the cause of an alarm LED on the CPC7301 Intelligent Shelf Manager (ISM).

Alarm LEDs

There are three levels of event severity supported on the CPC7301: minor, major and critical. When various events occur, the associated alarm LED on the front plate of the CPC7301 is lit. When an event, or alarm condition, is deasserted, the LED remains lit. The LED will remain lit until the alarm is cleared. This can be accomplished by using the CLI command 'alarm clear', by using SNMP, or by using the "Clear" button in the "Alarm" section of the main page of the GUI.

The ACO button toggles the ACO state. When ACO is activated, the active alarm LEDs blink and all of the alarm relays are deactivated. This button does not actually clear alarms.

By default, the CPC7301 is configured to set alarms on several conditions. This is based on the Platform Event Filter (PEF) configuration. This document assumes the PEF configuration is factory default and thus, the alarms are set accordingly. If you have modified your PEF to set alarms for additional conditions or to not set alarms for the default conditions, you will need to take this into consideration when interpreting your SEL (for more information, see “Understanding and Interpreting SEL Entries”). While you may not be able to use these exact commands to perform this task, the commands documented here might still be useful to you.

Table 1 outlines the causes for each of these alarm LEDs to be lit, given the default PEF configuration mentioned above.

Table 1 – Events Associated with Alarms

Alarm	Description
Minor	Upper non-critical threshold asserted
	Lower non-critical threshold asserted
	Power supply degrade asserted
Major	Upper critical threshold asserted
	Lower critical threshold asserted
	Fan tray removed
	Power supply fail asserted
Critical	Upper Non-recoverable threshold asserted
	Lower Non-recoverable threshold asserted

Understanding and Interpreting SEL Entries

In order to determine what is causing the alarm on your CPC7301, the System Event Log (SEL) must be interpreted. This document assumes that you clear the events in the SEL associated with each alarm, as each alarm is handled. If not, then you may end up with multiple SEL entries that could have caused the alarm, making it difficult to interpret which caused the alarm.

To interpret the SEL, use the CLI “sel” command in conjunction with the “grep” function as outlined in the following sections. But first, let’s look at a sample SEL entry and its key components.

0x0373: Event: at Jan 12 18:43:34 2004; from:(0xfc,0,0); sensor:(0x01,2); event: 0x1(deasserted): "Upper Non-Critical", Threshold: 0x27, Reading: 0x1c

<i>0x0373</i>	Event ID
<i>Jan 12 18:43:34 2004</i>	Time and date stamp
<i>0xfc</i>	IPMB Address generating the event
<i>2</i>	Sensor Number generating the event
<i>0x1(deasserted)</i>	Event direction –either asserted or deasserted

"Upper Non-Critical", Threshold Event type and class – the type of trigger for the event such as Lower Critical or Upper Non-Critical, and the class such as discrete or threshold.

For more detail on the SEL record format, see section 26 of the *Intelligent Platform Management Interface Specification Revision 1.5*.

Critical Alarms

Critical alarms are caused by either an upper or lower non-recoverable threshold being crossed.

For critical alarms, perform the following command:

```
# clia sel | grep "Non-Recoverable" | grep "(asserted"
```

You will see at least one SEL entry similar to this:

```
0x003D: Event: at Jan 16 20:06:54 2006; from:(0xfe,0,0); sensor:(0x01,2); event:  
0x1(asserted): "Upper Non-Recoverable", Threshold: 0x16, Reading: 0x1f
```

The information from this SEL entry shows that an upper non-recoverable threshold was asserted. The rest of the information takes some interpretation, but is easier than it would initially seem. The two key pieces of information embedded into the SEL entry are what device the event came from and which sensor is involved. The information that is highlighted in bold in the entry above will provide this information.

As shown in the sample SEL entry in the previous section, the portion of the entry “from:(0xfe...,” shows that the event is being generated from the device with IPMB address 0xfe. Using the “*FRU IPMB Controller Addresses*” table in the *CPC7301 Software Manual*, the device at 0xfe is determined to be ISM B.

The portion of the SEL entry, “sensor:(0x01, 2),” shows that it is sensor number 2 on the ISM B which is generating the event. Referring to the “*Local ISM Sensor Number*” table in the *CPC7301 Software Manual*, sensor 2 is determined to be the local temperature.

We have now determined that this particular major alarm was being caused by the upper non-recoverable threshold being asserted for the local temperature on ISM B. In this manner, all alarms can be interpreted.

There are three tables in the *CPC7301 Software Manual* that are useful when interpreting the SEL:

- FRU IPMB Controller Addresses
- Local ISM Sensor Numbers
- Active ISM Sensor Numbers

If the event is being generated from a device other than the ISM, such as an SBC or line card, a listing of sensors for that device will also be needed. For Performance Technologies products, the sensors are being added to the manuals or you may contact Technical Support for assistance.

Major Alarms

For major alarms, you will need to perform a few commands to narrow down the source.

First, try:

```
# clia sel | grep "Critical" | grep "(asserted"
```

If an upper critical threshold or lower critical threshold assertion is the cause of the alarm, you will see an entry similar to this:

```
0x0127: Event: at Jan 20 13:28:11 2006; from:(0xfe,0,0); sensor:(0x02,5); event:
0x1(asserted): "Lower Critical", Threshold: 0xb7, Reading: 0x00
```

Using the method shown previously, you can determine that this alarm was set by a lower critical threshold being asserted on backplane +12V on ISM B.

If an upper or lower critical threshold assertion did not cause the alarm, or you suspect more than one cause for the alarm, run the following command:

```
# clia sel | grep "Fail" | grep "(asserted"
```

If a power supply fail signal being asserted is the cause of the alarm, you will see an entry similar to this:

```
0x00DB: Event: at Jan 19 17:15:17 2006; from:(0x20,0,0); sensor:(0xe1,196); even
t:0x6f(asserted): Power Supply # 5, FRU # 27, "Fail"
```

The information in this SEL entry shows that power supply #5 asserted its fail signal. Looking at the IPMB address and the sensor number, you can see that this was generated by the active ISM on the state of power supply 5. This is a case where the information stated in plain English in the SEL entry is fairly redundant for the information interpreted from the IPMB address and sensor numbers.

If an upper or lower critical threshold has not been asserted, nor a power supply fail been asserted, then that leaves fan tray removal as the only cause for a major alarm. The ISM supports up to four fan tray FRU IDs. Depending on your chassis, you may have fewer than four fan trays. The FRU IDs for fan trays are outlined in Table 2 below.

Table 2 – Fan Tray FRU IDs

Fan tray 1	31
Fan tray 2	32
Fan tray 3	33
Fan tray 4	34

The following command may need to be repeated, based on the number of fan trays installed in your chassis, until you determine which fan tray has been removed.

```
# clia sel | grep "FRU <x>" | grep ">M0"
```

where *x* is the FRU ID of the fan tray.

If the removal of a fan tray is the cause of the major alarm, the above command will produce an output similar to this:

```
0x007C: Event: at Jan 18 14:00:59 2006; from:(0x20,0,0); sensor:(0xf0,34); event:0x6f(asserted): HotSwap: FRU 33 M1->M0, Cause=0x0
```

The above SEL entry might not be too obvious. You might determine that a hot swap event occurred. What else? We know that it is coming from the active ISM (IPMB address 0x20) and that it is relevant to fan tray 3 (sensor 34). The information “FRU 33” also reinforces that this is fan tray 3 that has been removed. How do we know that it has been removed? The information, “M1->M0” indicates the hot swap state change of the fan tray. M0 indicates the fan tray is not installed. The hot swap states are defined in the *PICMG 3.0 AdvancedTCA Specification’s Shelf Management* section.

Minor Alarms

Minor alarms are caused by one of two conditions: an upper or lower non-critical threshold being asserted, or a power supply degrade signal being asserted. To determine the cause of a minor alarm, you might start by looking for a power supply degrade assertion:

```
# clia sel | grep "Degrade" | grep "(asserted"
```

If a power supply degrade signal being asserted is the cause of the alarm, you will receive output to the above command similar to this:

```
0x00DD: Event: at Jan 19 17:15:21 2006; from:(0x20,0,0); sensor:(0xe1,196); event:0x6f(asserted): Power Supply # 5, FRU # 27, "Degrade"
```

This SEL entry indicates that power supply #5 has a degrade signal asserted. Interpretation of the IPMB address and sensor number reinforce this information.

If this is not the case, try the following command:

```
# clia sel | grep "Non-Critical" | grep "(asserted"
```

If a non-critical threshold was asserted, you will see output from the above command similar to this:

```
0x003B: Event: at Jan 16 20:06:41 2006; from:(0xfe,0,0); sensor:(0x01,2); event:  
0x1(asserted): "Upper Non-Critical", Threshold: 0x11, Reading: 0x1f
```

This SEL entry indicates that an upper non-critical threshold has been asserted. Interpretation of the IPMB address and sensor number indicate this is for the local temperature on ISM B.

Other Considerations

This document has covered how to determine the conditions which have set one or more alarm LEDs on your CPC7301. What we have looked at is the initial triggering of the alarm, by the assertion of one or more of the conditions outlined in Table 1. This may or may not be enough information for you to determine what the trouble is with your system. If it is not, then you will want to look at the SEL entries with more context.

All of the commands listed here use the “grep” command to find the assertion of a condition. You may need to look at the context of the SEL to see if there was a corresponding deassertion. Did the temperature of the ISM come back down? Was it a momentary spike? Is a voltage threshold being asserted and deasserted continuously indicating a fluctuation in the voltage? To do this, you can use similar commands as outlined in this document, using the event index to examine more closely a particular event and its surrounding events.

The command would look like this:

```
# clia sel | grep -cn "<event ID>"
```

where *event ID* is the event ID from the SEL entry and *n* is the desired number of lines of context to be displayed.