



Version 2.0

User Manual

by Jamie Green

**Rebound! designed by
Amar Singh**

**Rebound! software by
Richard Elmore**

**Critical mass
Jonathan Feinstein**

Service requirements

In the event of equipment malfunction, all repairs should be performed by our Company or an authorized agent. It is the responsibility of users requiring service to report the need for service to our Company or to one of our authorized agents. Service can be obtained at Sophisticated Circuits, Inc., PO Box 727, Bothell, WA 98041-0727, 425-485-7979.

FCC statement

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Shielded cables and I/O cords must be used for this equipment to comply with the relevant FCC regulations.

Changes or modifications not expressly approved in writing by Sophisticated Circuits, Inc. may void the user's authority to operate this equipment.

PowerKey Rebound! PKRB-M



Tested To Comply
With FCC Standards

FOR HOME OR OFFICE USE

Industry Canada statement

This Class B digital apparatus meets all requirements of the Canadian Interference-Causing Equipment Regulations (ICES-003, Class B).

Cet appareil numérique de la Class B respecte toutes les exigences du Règlement sur le matériel brouilleur du Canada (ICES-003, Class B).

Europe-EU declaration of conformity

Rebound! conforms to the following European Directives and Standards: Application of Council Directives: 73/23/EEC, 89/336/EEC. Standards to which Conformity is declared: EN60950, EN55022, EN50082-1.

Copyright notice

The PowerKey Rebound! documentation, program, product design and design concepts are copyright ©2000 Sophisticated Circuits, Inc. All rights reserved. Your rights are subject to the copyright laws of the United States of America. Under the copyright laws, this manual may not be copied, in whole or part, including translation to another language or format, without the express written consent of Sophisticated Circuits, Inc.

Trademarks

PowerKey and PowerKey Rebound! are registered trademarks, and Kick-off! and monoLog are trademarks of Sophisticated Circuits, Inc.

Apple, the Apple logo, AppleScript, Mac OS, Mac, Macintosh and PowerBook are registered trademarks, and Balloon Help, Finder and iBook are trademarks of Apple Computer, Inc.

Frontier is a trademark of UserLand software.

Contents

Introduction

System requirements	4
-------------------------------	---

Installation

Hardware installation	5
Software installation	6

Using Rebound!

The Rebound! control panel	8
The “System Crashes” panel	9
The “Application Crashes” panel	11
The “Other” panel	14

Advanced topics

How Rebound! works	
Detecting system crashes	16
Detecting application crashes.	16
Testing Rebound!	
Testing system crash detection.	18
Testing application crash detection.	18
Using Rebound! with AppleScript	
Application properties	19
AppTimer elements	22

Troubleshooting	23
----------------------------------	-----------

How to contact us	25
------------------------------------	-----------

Introduction

Thank you for purchasing PowerKey Rebound!®

Rebound! continuously monitors your Mac® system to make sure it is running normally. If the system or an application crashes, Rebound! restarts it automatically. Many popular server applications integrate Rebound! support to give additional protection. Once you install Rebound!, you can sleep well, knowing your server will stay up and running!

System requirements

Rebound! works with any desktop or server Power Macintosh® computer equipped with a ADB port. It also works with later 68040 Macs that respond to the keyboard “command-control-powerkey” restart command.



For older ADB Mac models which do not respond to this command consistently, we recommend PowerKey® Pro with the Server Restart Option™. PowerKey Pro has the ability to turn off its outlets, effectively “pulling the plug” to restart your server.



For newer Macs with a USB port, try our newest product, Kick-off™. It offers the same proven crash-detection technology as Rebound! and PowerKey Pro, and connects to your USB port.

Rebound! uses the same patents-pending technology as our PowerKey® Pro family of smart power strips to monitor your computer. If it crashes, Rebound! automatically sends the “command-control-powerkey” signal to restart your server.

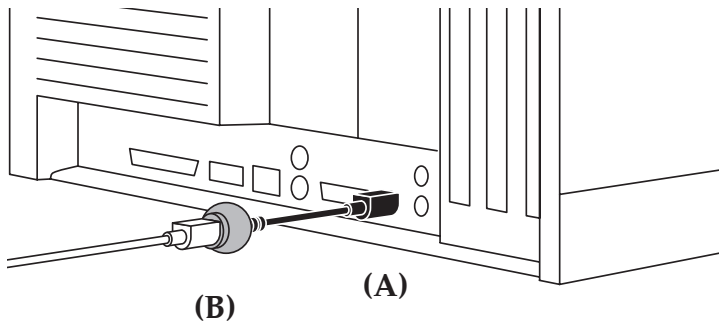
The Rebound! software requires Mac® OS 8.5.1 or later. It supports Balloon Help™, Apple Help, Apple Events and AppleScript®.

Installation

PowerKey Rebound! detects crashes using both hardware and software, working in combination. The software gives the hardware the feedback it needs to determine whether or not the system has crashed, so you must install both to use Rebound!

Hardware installation

The Rebound! hardware connects to your computer's Apple Desktop Bus ("ADB") port (as labelled with the symbol to the right) to communicate with the software. It has a pass-through connector allowing you to connect your keyboard (and possibly other devices) in a "chain." Refer to the diagram below:



1. Shut down your computer normally.
2. Disconnect your keyboard cable from the ADB port on the back of your computer.
3. Connect Rebound!'s short cable to the ADB port on the back of your computer (A).

4. Connect your keyboard cable into Rebound! (B).

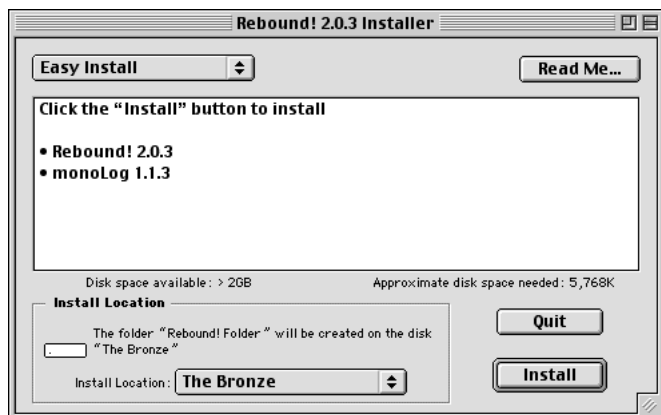


A keyboard is not required for Rebound! to work. If you are using Rebound! on a server with no keyboard, you may skip this step.

5. Start up your computer normally.

Software installation

6. Insert the Rebound! CD into your CD drive, and double-click the icon labelled “Installer.” The Installer window will open.



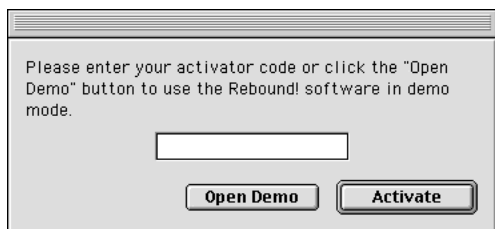
7. Select a location for the Rebound! manual and supplemental files from the “Install Location” pop-up menu. (The Rebound! Extension and control panel will be installed in the active system folder.)
8. Click Install. The Installer will ask if it can quit any open applications before it begins. Make sure you don’t have any unsaved documents, then click Continue.

When the installation is complete, the Installer will restart your computer.



You can also select “Custom Install” from the pop-up menu to install optional components. See the “Read Me—Rebound!” document for more information on these components.

9. After your computer has started up, select “Control Panels” from your Apple menu, and open “Rebound!”. A dialog box will appear.



10. Type your activator code into the field in this dialog box. This code can be found in your Rebound! packaging, or if you purchased a Rebound! 2.0 upgrade, in the e-mail order confirmation.

Rebound! starts watching for system crashes as soon as you enter your activator code. The next section shows you how Rebound! works, and how you can change its settings.

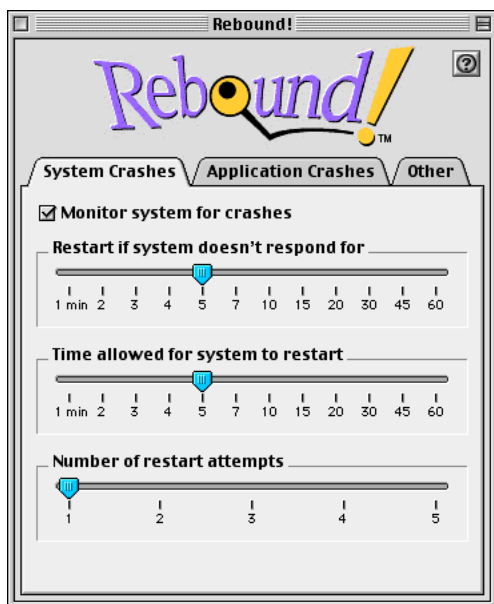
Using Rebound!

As soon as you finish installing Rebound!, it will begin monitoring your system for crashes. Rebound! is designed to work unobtrusively in the background: you will never see any sign that it is working until your system crashes!

The Rebound! control panel

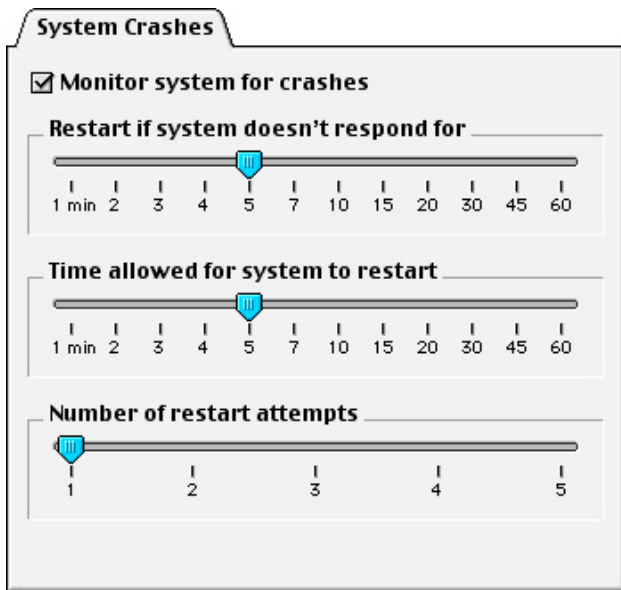
Rebound!'s default configuration should work in most systems, but you can customize its settings using the Rebound! control panel. This control panel is installed in the Control Panels folder, and can be accessed from the Apple menu.

When you open the control panel, a window appears with three tabs. Click on a tab to reveal its configuration panel.



The “System Crashes” panel

When you click on the “System Crashes” tab, a panel appears with controls for customizing Rebound!’s system crash detection.



The screenshot shows the 'System Crashes' panel with the following settings:

- ☒ **Monitor system for crashes**
- Restart if system doesn't respond for**: A slider set to 5 minutes. The scale ranges from 1 min to 60 minutes with increments of 1, 2, 3, 4, 5, 7, 10, 15, 20, 30, 45, and 60.
- Time allowed for system to restart**: A slider set to 5 minutes. The scale ranges from 1 min to 60 minutes with increments of 1, 2, 3, 4, 5, 7, 10, 15, 20, 30, 45, and 60.
- Number of restart attempts**: A slider set to 1. The scale ranges from 1 to 5 with increments of 1.

Monitor system for crashes

If this box is checked, Rebound! will monitor your system and attempt to restart it when a crash is detected. Uncheck this box to turn Rebound!’s system crash detection off.



It is a good idea to uncheck this box before doing any unusually processor-intensive task such as a long calculation. See “Restart if system doesn’t respond” below for more information.

Restart if system doesn’t respond

This slider lets you set how long the Rebound! hardware will wait before deciding that the system has crashed.

If an application runs certain long uninterruptible tasks, the normal communication between the Rebound! software and hardware can be temporarily stopped. To keep Rebound! from misinterpreting these pauses as crashes, you can set this slider to a higher waiting time.



Be careful not to set this waiting time too low. The lower the waiting time, the more likely a long task will be misinterpreted as a crash.

Time allowed for system to restart

If the *Number of restart attempts* setting (see below) is higher than '1', this slider lets you set how long the Rebound! hardware will wait to reestablish communication with the software after a restart.

Normally, the Rebound! software reestablishes communication as the system finishes restarting. If the system crashes again during restart, the Rebound! software won't be able to do this. This slider lets you set how long the Rebound! hardware will wait before it decides something is wrong and tries to restart the computer again.

Number of restart attempts

This slider lets you set the number of times the Rebound! hardware will attempt to restart the computer if it crashes again during the restart process.

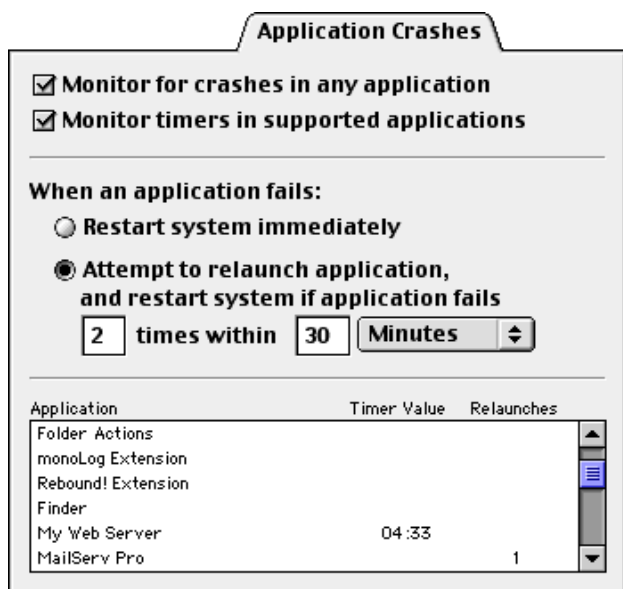
This feature works in conjunction with the *Time allowed for system to restart* slider above. After detecting a crash and restarting the system, the Rebound! hardware will wait for the Rebound! software to reestablish communication, signifying a successful restart. If this doesn't happen within the time allowed, the hardware can try again up to four more times.



If this slider is set higher than '1' and you disable the Rebound! Extension, you may experience unwanted restarts. Be sure to set this slider to '1' before disabling the Rebound! software.

The “Application Crashes” panel

When you click on the “Application Crashes” tab, a panel appears with controls for customizing Rebound!’s application crash detection. This panel is divided into three sections. The first lets you enable application crash detection, the second lets you configure crash recovery, and the third shows you the current system status.



The screenshot shows the 'Application Crashes' panel with the following settings:

- ☒ Monitor for crashes in any application
- ☒ Monitor timers in supported applications

When an application fails:

- ☐ Restart system immediately
- ☒ Attempt to relaunch application, and restart system if application fails

2 times within 30 Minutes

Application	Timer Value	Relaunches
Folder Actions		
monoLog Extension		
Rebound! Extension		
Finder		
My Web Server	04:33	
MailServ Pro		1

Monitor for crashes in any application

If this box is checked, Rebound! will monitor all running applications for crashes. This will detect any time an application disappears, causing an “application has unexpectedly quit” error message to appear.

Monitor timers in supported applications

If this box is checked, Rebound! will monitor applications which include Rebound! support. This will let Rebound!

respond to failures which don't necessarily cause the application to completely crash.



In order for Rebound! to monitor an application's timer, the application must have support built in. Many popular server applications already include Rebound! support, and more are added regularly. A current list can be found at our web site, www.sophisticated.com.

When an application fails

This section of the "Application Crashes" panel contains a pair of radio buttons that lets you choose how Rebound! will attempt to recover from crashes.

If you select *Restart system immediately*, Rebound! will restart the computer when an application crashes. It will first attempt to initiate a Finder restart, and the Finder will attempt to quit all open applications. If an application fails to quit, the Rebound! hardware will then perform a hard restart, after a delay specified by the *Restart if system doesn't respond* slider in the "System Crashes" panel.

If you select *Attempt to relaunch application*, Rebound! will first attempt to relaunch the application which crashed, before it resorts to restarting the system.

Application crashes can potentially leave the system in an unstable state. If the application keeps crashing after being relaunched, it probably means that something else is wrong with the system. In this situation, Rebound! can "fall back" and restart the entire system. You can configure when Rebound! will resort to restarting the system by typing values into the two fields below this radio button.

The first value is the number of crashes that will cause Rebound! to restart the system. (A value of '1' here is equivalent to selecting the *Restart system immediately* button.)

The second value (and associated pop-up menu) is the interval in which these repeated crashes must occur. If an application crashes once every few days, Rebound! will

simply relaunch it, but if it crashes repeatedly over a short period, Rebound! will restart the system.

Application status

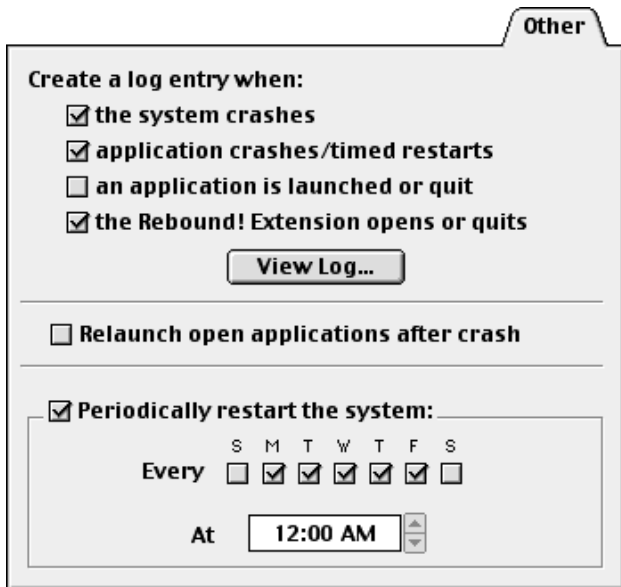
The third section of the “Application Crashes” panel contains a list of currently running applications and their statuses. (The list shows all running applications, including “invisible” applications that are part of Mac OS, in the order they were launched.)

The second column shows the current value of the timer for each application which supports Rebound!’s application timers. This timer counts down each second until the next time it is updated by the application. See “Detecting application crashes” on page 16 for more information.

The third column shows the number of times each application has been relaunched recently (i.e. within the relaunching interval described above). This alerts you to applications which may be crashing frequently.

The “Other” panel

When you click the “Other” tab, a panel appears with controls for customizing all of Rebound!’s extra features.



The screenshot shows the 'Other' panel of the Rebound! settings window. The panel has a tab labeled 'Other' at the top right. It contains the following options:

- Create a log entry when:**
 - ☒ the system crashes
 - ☒ application crashes/timed restarts
 - ☐ an application is launched or quit
 - ☒ the Rebound! Extension opens or quits
- View Log...** (button)
- ☐ Relaunch open applications after crash
- ☒ Periodically restart the system:
 - Every: S ☐ M ☒ T ☒ W ☒ T ☒ F ☒ S ☐
 - At: 12:00 AM (with up/down arrows)

Create a log entry when...

Rebound! supports our new “monoLog” software to make a log of system activity. These check boxes let you configure what types of activities Rebound! will add to the log. Click the *View Log...* button to open monoLog.



You can also use monoLog to notify you when important activities occur. See the monoLog User Manual for more information.

Relaunch open applications after crash

If this box is checked, Rebound! will relaunch every application that was running before a crash when it restarts the system. This “bookmarking” feature is more flexible than

Mac OS' "Startup Items" folder, since it returns your computer to the state it was in before a crash.



The bookmarking feature will relaunch applications, but it will not open documents that may have been open before the crash.



For security reasons, the bookmarking feature is disabled when running in Mac OS 9 multiple user mode.

Periodically restart the system

If this box is checked, Rebound! will automatically restart the system at the same time of day on any day(s) of the week you specify.

As with application crashes, Rebound! will first attempt to initiate a Finder restart, and the Finder will attempt to quit all open applications. If an application fails to quit, the Rebound! hardware will then perform a hard restart, after a delay specified by the *Restart if system doesn't respond* slider in the "System Crashes" panel.



Some applications suffer from slow "memory leaks" or fragmentation of memory. Over time these can accumulate and reduce the performance and stability of your server. Periodically restarting your computer during times of low usage can reduce the chance of a crash occurring during times of high usage.

Advanced Topics

How Rebound! Works

Detecting system crashes

The Rebound! hardware monitors your system by periodically communicating with the Rebound! software. Every few seconds, the software resets a “system timer” in the hardware, which then counts down each second. As long as the system is running normally, the software will keep resetting this timer, and it will never reach zero.

If the system crashes, the Rebound! software will stop running, and won’t be able to reset the system timer. When the timer reaches zero, the hardware will attempt to restart the computer. To do this, it sends a “command-control-power-key” signal, just as if you’d typed it on a keyboard.

Detecting application crashes

Rebound! includes two types of application crash detection. The first will detect any crash which causes an “application has unexpectedly quit” message to appear. To do this, the Rebound! software watches as applications are opened and quit. If one quits abnormally, the Rebound! software will notice and can respond as you specify in the “Application Crashes” panel.

The second type of application crash detection monitors applications which include Rebound! support. This lets Rebound! respond to failures which don’t necessarily cause the application to quit.

This type of detection works in a similar manner to the system timers described in “Detecting system crashes” above. An application with Rebound! support periodically resets an “application timer” in the Rebound! software. If the application fails, its application timer will count down to zero. When it reaches zero, the Rebound! software will respond.



You can see the application timers working in the “application status” section of the Rebound! control panel.

A list of applications which support application timers can be found on our web site at www.sophisticated.com, or check with your application’s developer to see if they support Rebound!

Testing Rebound!

Testing system crash detection

To simulate a system crash, hold down the command key and press the power key. A small window will appear with nothing but a '>' symbol in it. This window is the Mac's built-in mini-debugger, and while it is open, no other software can run. The Rebound! software will not be able to update Rebound!'s system timer, and in a few minutes Rebound! will restart the computer.



There are also several test applications available which purposely cause a variety of system crashes. We offer one called "Crash Test," which is available in the Rebound! installer and on our web site, www.sophisticated.com.



Some crash utilities purposely quit all applications before crashing. These utilities have less real-world validity (real crashes aren't so polite), and actually disable some crash detection tools, including Rebound! The Rebound! software will display a warning dialog if another application attempts to make it quit.

Testing application timers

To simulate a failure in an application with support for Rebound!'s application timers, bring that application to the foreground and type command-option-escape. You will be asked if you wish to force the application to quit; click *Force Quit*. Since the application wasn't able to quit normally, its Application Timer will continue counting down. Rebound! will think the application has crashed and respond as you specified in the "Application Crashes" panel.



This test will not simulate a full application crash in an application without support for Rebound!'s application timers. For these applications, Rebound! only responds to "application has unexpectedly quit" messages.

Using Rebound! with AppleScript

Advanced users can control all of Rebound!’s features with AppleScript or any other OSA-compliant scripting language (such as UserLand Frontier™). To work with Rebound!’s settings, place a **tell block** in your script as follows:

```
tell application "Rebound! Extension"  
    . . .  
end tell
```

Application properties

The Rebound! Extension contains several properties which can be read or modified. These properties correspond to the settings in the Rebound! control panel.

version	Version of the Rebound! software
---------	----------------------------------

“System Crashes” panel properties

sysTimerEnabled	Set to true or false to enable or disable system crash detection
-----------------	--

crashDetectTimeout	Value of the <i>Restart if system doesn’t respond</i> slider, in seconds
--------------------	--

restartInterval	Value of the <i>Time allowed for system to restart</i> slider, in seconds
-----------------	---

restartCount	Value of the <i>Number of restart attempts</i> slider
--------------	---

“Application Crashes” panel properties

appCrashDetectEnabled	Set to true or false to enable or disable application crash detection
appTimersEnabled	Set to true or false to enable or disable application timer monitoring
appRelaunchCount	Value of the <i>Attempt to relaunch application</i> count field (set to 0 to enable the <i>Restart immediately</i> button)
appRelaunchWindow	Value of the <i>Attempt to relaunch application</i> interval field, in seconds

“Other” panel properties

sysCrashRestartPriority	Priority of system crash log messages
appCrashPriority	Priority of application crash/timed
appTimerExpirePriority	restart log messages (timed restart
appRelaunchPriority	messages use the same priority as
appCrashRestartPriority	application crash restart messages)
appLaunchPriority	Priority of application launch/quit
appQuitPriority	log messages
appForceQuitPriority	
daemonLaunchPriority	Priority of “Rebound! Extension
daemonQuitPriority	started” and “...quit” log mes-
daemonQuitAEPriority	sages



See the monoLog User Manual for a description of priority levels of log messages. To disable any of these log messages, set its priority value to -1.

bookmarkingEnabled Set to true or false to enable or disable relaunching of open applications after crashes

rebootDays Days to periodically restart the server (set to 0 to disable periodic restarting)



rebootDays is a “bitmask” field: each day of the week has a value (Sunday=1, Monday=2, Tuesday=4, Wednesday=8, Thursday=16, Friday=32 and Saturday=64). To set the days for restarting, add the values for the desired restarting days. For example, to restart on Monday, Thursday and Saturday, set rebootDays to $2+16+64=82$.

rebootTime Time of day to periodically restart the server, in seconds after midnight

Using the properties

You can examine and modify these values by using AppleScript’s **get** and **set** commands. For example, to change the value of the *Restart if system doesn’t respond* slider to ten minutes, execute the following command inside your tell block:

set crashDetectTimeout to 600



It is possible to set values between or outside the marks on the control panel’s sliders, but they will snap back the next time the user modifies them from the Rebound! control panel.

AppTimer elements

Rebound!'s application crash detection works by creating **appTimers** for each application being monitored. Each application sets its appTimer periodically to keep it from reaching zero. If an appTimer does reach zero, Rebound! assumes the application has crashed and restarts the system.

You can use Rebound!'s application crash detection to monitor custom scripts, databases or applications you have written. To do this, your application should periodically send a **tickleAppTimer** AppleScript command:

tickleAppTimer 300

The Rebound! software will automatically create a new appTimer for your application if necessary, and repeated commands will update that timer.



You can also set your appTimer with a simple Apple Event, bypassing AppleScript. Use Event Class 'PKPr' and Event ID 'Tick', and put the desired value as a long integer into the direct object parameter.

Applications will usually not need to view appTimers themselves, but if you do wish to do this, you can deal with them like any other element of an application. AppTimers have **name** and **value** properties. You cannot set the properties of an appTimer, but you can get them. To set your appTimer, use the tickleAppTimer command described above.

get name of every appTimer
get value of appTimer "My Application"



For development information on supporting Rebound!, see the PowerKey Software Developer's Kit in the Rebound! installer and on our web site at www.sophisticated.com.

Troubleshooting

Rebound! doesn't restart my computer after the system crashes.

Some crashes which appear to be system crashes are actually caused by applications. Though the system may be unresponsive, other processes are still running normally. See “Detecting Application Crashes” on page 16 for more information.

I disabled my Rebound! software, but my computer keeps restarting every few minutes.

If the *Number of restart attempts* slider is set higher than ‘1’, the Rebound! hardware will wait for the Rebound! software to reestablish communication at startup. Since the software has been disabled, this won’t occur, and the Rebound! hardware will restart the computer again.

To avoid this problem, always set this slider to ‘1’ before disabling the software.

Rebound! doesn't recover after an application crashes.

It is possible for applications to fail in ways that do not cause actual crashes. Our “application timer” technology gives Rebound! another method of detecting application failures. An application must be specifically written to support Rebound!’s application timers. Contact your application’s developer to see if they offer Rebound! support.



Rebound! uses the same software as our Rebound! product, so if your application supports Rebound!’s application timers, it automatically supports Rebound! as well.

If you are running an application which includes support for Rebound!’s application timers, make sure its timer is being set. Open the Rebound! control panel and click on the

"Application Crashes" tab. The timer should be shown in the list at the bottom of the panel. If it does not appear, check your application's user manual to configure this support.

Rebound! restarts my computer when I run some applications or tasks.

Some long operations will take control of the computer, and not give time to any other processes, including the Rebound! Extension. The Rebound! unit interprets this as a crash. Increase the value of the *Restart if system doesn't respond* slider.

Some applications, including some disk utilities and games, will take control of the computer for very long times. If you need to run one of these programs, you should temporarily uncheck the *Monitor system for crashes* box.



This can be done automatically with AppleScript. See "Using Rebound! with AppleScript" on page 20 for more information.

An error message appears at startup every time Rebound! restarts my computer, saying the computer was shut down improperly.

This is a standard message that appears in Mac OS after a crash occurs. You can disable this message in the General Controls control panel. Uncheck the *Warn me if computer was shut down improperly* box.

My log shows that Rebound! restarted due to a system crash immediately after startup.

Rebound! can't add an entry to your log while the computer is crashed, so it adds one after it restarts the computer. The monoLog software must be running before Rebound! can add its entry, so the "monoLog Extension started" entry will appear before the crash (or any other) entry.

How to Contact Us

Mailing:

Sophisticated Circuits, Inc.
PO Box 727
Bothell, WA 98041-0727

Shipping:

Sophisticated Circuits, Inc.
19501 144th Ave NE Ste D1000
Woodinville, WA 98072

Phone:

Voice: 425-485-7979
Sales: 800-769-3773
Fax: 425-485-7172

Internet:

Web: <http://www.sophisticated.com/>

E-mail:

info@sophisticated.com,
sales@sophisticated.com,
support@sophisticated.com