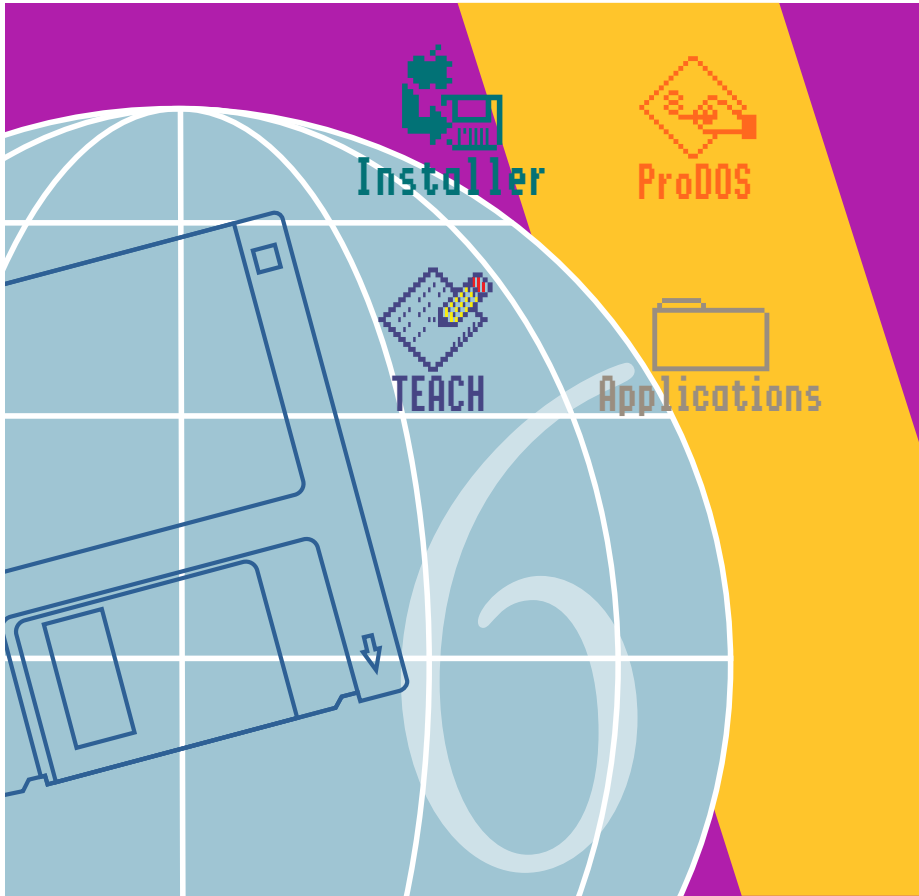




# *Sound Control Panel 2.0*

*Ver. 1.1*



# System 6 IIGS

Apple® IIGS® System Software 6.0 — Release Notes  
Golden Master 3 Release — March 5, 1992  
Sound Control Panel 2.0  
Version 1.1

PDF by [khaibitgfx@outlook.com](mailto:khaibitgfx@outlook.com)  
September 6, 2019

Apple, the Apple logo, GS/OS®, and IIGS® are registered trademarks of Apple Computer, Inc.  
© Apple Computer, Inc., 1992  
20525 Mariani Avenue  
Cupertino, CA 95014-6299  
(408) 996-1010



[khaibitgfx@outlook.com](mailto:khaibitgfx@outlook.com)

# Contents

Introduction & Overview / 1
Overview of the Sound Control Panel / 1
Files related to the Sound Control Panel / 2
Human Interface DetailsDesign / 2
Opening the Sound Window / 2
The Sound Window / 2
The Scroll bars / 2
The Check Box / 2
The Event Menu / 3
The Sound Menu / 3
Copy Sound to Clipboard / 4
Notes / 4
Help / 4
Implementation Details / 5
BootCDEV / 5
Contents of the Event Menu / 5
Contents of the Sound Menu / 5
rSoundSample Format / 5
Sound's Request Procedure / 5
Sound Chains into BELLVECTOR / 6
Battery RAM locations / 7
Limitations / 7

# Introduction & Overview

This document describes Sound Control Panel 2.0, part of Apple IIGS System Software 6.0.

## Overview of the Sound Control Panel

### Features:

- The Pitch scroll bar lets you adjust the pitch of the standard beep. (The setting is stored in Battery RAM.)
- The Volume scroll bar lets you adjust the system volume. (The setting is stored in Battery RAM.)
- The “Give visual indication of sounds” checkbox tells the system whether to blink the border along with sounds. (The setting is stored in Battery RAM.)
- Event and Sound pop-up menus let you choose what sound to play for each of twenty-two different events. (The various events occur when something calls `SysBeep2` in the Miscellaneous tools. Many parts of the system do this, as does third-party software that takes advantage of System 6.0.)
- Each event can be set to any sound in your Sounds folder, or to Standard Beep, Silence, or Not assigned.
- Sounds are stored as named `rSoundSample` resources in the `*:System:Sounds` folder.

## Files related to the Sound Control Panel

Sound in `*:System:CDevs`—this is the Sound Control Panel itself.

Sound.Settings in `*:System:Sounds`—this stores your assignments from events to sounds.

Sounds are read from all extended files in `*:System:Sounds` that contain named `rSoundSample` resources (type \$8024; see Apple IIGS Technical Note #76, Miscellaneous Resource Formats). This is the same sound format HyperCard IIGS uses.

## Human Interface Details Design

### Opening the Sound Window

To open the Sound Control Panel, first open the Control Panels NDA by choosing “Control Panels” from the Apple menu (or type Command-Shift-Esc). Scroll to the Sound icon and double-click it (or type “so” return).

If you're in the Finder, you may instead double-click the Sound file directly from the CDevs folder. (You can open the CDevs folder by holding down Option while choosing Control Panels, or by typing Option-Command-Shift-Esc).

### The Sound Window

**The Sound window contains two scroll bars, one check box, and two pop-up menus, as described below.**

#### *The Scroll bars*

The Pitch scroll bar adjusts the pitch of the Standard Beep (there are fifteen levels).

When you adjust the Pitch scroll bar, you hear the Standard Beep, regardless of the sound showing in the Sounds menu.

The Volume scroll bar adjusts the system volume (there are fifteen levels).

When you adjust the Volume scroll bar, the System Beep plays (this may or may not be the Standard Beep—it depends how you've set it with the pop-up menus).

#### *The Check Box*

There is a “Give visual indication of sounds” checkbox. When the box is checked, the system blinks the screen border when playing certain sounds.

You can toggle the check box by clicking in the box or in the text, or by typing G (or Command-G).

**The system beep and the following SysBeep2 events can have a visual indication:**

\$0000..\$000C	impossible operations, etc)
\$0030	disk request
\$0033	bad disk
\$0050..\$0059	AlertWindow
\$0Exx	ErrorWindow
\$0Fxx	reserved

*The Event Menu*

The Event menu lists a subset of the defined SysBeep2 codes (see the System 6.0 Toolbox documentation, Miscellaneous tools chapter, under SysBeep2).

When you choose an Event, the Sound menu automatically shows you what sound is currently assigned to that event.

---

<b>Code</b>	<b>Name</b>	<b>Code</b>	<b>Name</b>
---	System Beep		
\$0050	Attention	\$000A	Input field full
\$0033	Bad disk	\$0053	Note alert
\$0008	Bad keypress	\$0052	Stop alert
\$0009	Bad input value	\$0031	System startup
\$0004	Can't click there	\$0015	System shutdown
\$0054	Caution alert	\$0005	Task completed
\$0014	Disk ejected	\$000C	Task failed
\$0013	Disk inserted	\$000B	Task impossible
\$0030	Disk request	\$0041	Whoosh closed
\$0043	Empty trash	\$0040	<b>Whoosh open</b>
\$0042	Fill trash	\$0100	<b>You Have Mail</b>

---

*The Sound Menu*

Choosing an item from the Sound menu associates the current Event with that sound, and immediately plays the sound as a sample. (To hear the current sound again, just press Return or Enter.)

The Sound menu lists “Not assigned”, “Silence”, and “Standard Beep,” plus any sounds found in the Sounds folder. (If no sounds are present, the dividing line under Standard Beep does not appear.)

Only the sounds that presently have events mapped to them are kept in RAM; others are loaded from disk as needed.

(Note that sounds that stop being mapped to events remain in RAM until you reboot.)

### *Copy Sound to Clipboard*

Choose Cut or Copy from the application's Edit menu to copy the current sound (as shown in the Sound menu) to the clipboard (you can't copy Not assigned, Silence, or Standard Beep). This puts both the sound and its name onto the clipboard (as a `sampledSoundScrap, $0002`, and a `textScrap, $0000`).

### *Notes*

The pop-up menus are “type 2” so that there is white space for items to scroll into, as needed. (Select an item at the top of a menu, and then pop it up again. There will be white space above the item. With type 1 menus, you would only be able to see a few items at a time.)

If the Sound window (or any other System window) is in front, Command-W always closes it (the only exception is if you have GSBug installed and the CapsLock key is down; this is a GSBug feature which causes the current application to handle the keypress—if the application supports Command-W for Close, it will work anyway).

You can double-click on the Sound file (like any other Control Panel) from the Finder. The Sound window will appear (but the Control Panels window will not).

### *Help*

**To read the help, select the Sound icon in the Control Panels window, and click Help (or type Command-H). The help reads:**

- The Sound Control Panel lets you change the pitch of the standard beep and the volume of all sounds.
- To assign a sound to an event, first choose the event from the Event menu, and then choose a sound from the Sound menu. Sounds are stored in the Sounds folder inside the System folder.



# Implementation Details

## BootCDEV

When Sound receives the `BootCDEV` message, it installs a Request Procedure (see System 6.0 Toolbox documentation, Tool Locator chapter), scans the `*:System:Sounds` folder for sounds, and loads the Sound Settings file if present.

If the request procedure can't be installed (usually because you have two copies of the Sound Control Panel in your system!), Sound instructs the Control Panel to “X” out the Sound icon on the boot screen.

## Contents of the Event Menu

The “System Beep” item is hard-coded, but all other events come from the `rTaggedStrings ID=1` resources in the Sound file. (If somebody writes an editor for this standard resource type, power users will be able to easily customize their Event menu.)

(The `rTaggedStrings ID=1` resource is locked and fixed and should remain that way.)

## Contents of the Sound Menu

The “Not assigned”, “Silence”, and “Standard Beep” items are hard-coded. All other items are the resource names of `rSoundSample` resources found in files in the `*:System:Sounds` folder.

Sound scans for sounds at boot time, and whenever the Sound window opens.

## rSoundSample Format

Just like HyperCard IIGs, the Sound Control Panel plays sampled sounds at the pitch indicated by the `relPitch` field, ignoring the sample-frequency field.

(See the HyperCard IIGs Technical Notes for more information.)

## Sound's Request Procedure

Here are the request codes the Sound Control Panel's request procedure can accept:

`$0001 systemSaysBeep`

This is the one `SysBeep2` uses. If Sound successfully plays a sound mapped to the given event, it accepts the request. The low word of `dataIn` is the event code; bit 31 of `dataIn` disables playing the sound (but it is still accepted normally).

#### \$0004 `srqGetrSoundSample`

This is a request to return an `rSoundSample` handle corresponding to a given name. `dataIn` is a pointer to the Pascal string name. If Sound can provide such a handle, it accepts the request and fills in the `dataOut` buffer, which is formatted as follows:

`dataOut`:

+000: `recvCount` (required by `SendRequest`)

+002: sound handle (not guaranteed to be locked)

+006: flags word (bit 15 = caller must dispose of the handle when done, 14-0=reserved)

#### \$0005 `srqSynchronize`

`dataIn` and `dataOut` are reserved and should be zero. Sound waits for any pending sound to finish playing; then accepts the request (whether it had to wait for anything or not).

#### \$0006 `srqPlayrSoundSample`

`dataIn` is the handle of an `rSoundSample` resource. If Sound can play the sound, it begins playing and accepts the request; otherwise it rejects the request. `dataOut` is reserved.

There are special values for `dataIn`: If the value is `$0000xxxx`, it's not a sound handle. `$00000001` is a request to play Silence, and `$00000002` is a request to play the Standard Beep.

\$8000 (sent by name to "Apple~SoundCP~")

\$8001 (sent by name to "Apple~SoundCP~")

\$8003 (sent by name to "Apple~SoundCP~")

\$8100 (sent by name to "Apple~SoundCP~")

These are used internally for the Sound Control Panel to communicate with its request procedure—nobody else should call them. (The request procedure stays in RAM all the time, but the main Control Panel code does not, so they can't just communicate with direct calls or link-time shared globals.)

## Sound Chains into BELLVECTOR

For remapping the System Beep, Sound chains into BELLVECTOR, using `GetVector($1B)` and `SetVector($1B)`. Entry and exit is in 8-bit native mode. On exit, B, D, and X are always preserved, and Y is always zero (as required by the Apple IIGS Firmware Reference).

If the System Beep has been reassigned with Sound, then it plays the remapped sound and returns with the carry clear.

If System Beep has not been reassigned, it passes control to the previous value of BELLVECTOR.

## Battery RAM locations

- Pitch: location \$1F (range 0 to 14).
- Volume: location \$1E (range 0 to 14).
- Visual indication of sounds: bit 0 of location \$5E (the bit is set when visual indication of sounds is *off*).

## Limitations

Sound Control Panel 2.0 accepts `srqPlayrSoundSample` only if the Sound Tools are available and a page of bank zero space is allocatable. However, separate utilities can accept `srqPlayrSoundSample` to play sounds under difficult conditions.