Producing an Alert Sound Catching the User's Attention

You can produce a **system alert sound** to catch the user's attention by calling the **SysBeep** procedure. The **SysBeep** procedure is a **Sound Manager** routine that plays the system alert sound, selected by the user in the **Sound** control panel. Here's an example of calling **SysBeep**:

```
if (myErr != noErr) SysBeep(30);
```

You must supply a parameter when you call the **SysBeep** procedure, even though the **Sound Manager** ignores that parameter in most cases. All system alert sounds are stored as format 1 'snd ' resources in the System file and are played by the **Sound Manager**. There is, however, one exception to this rule: if the user has selected the Simple Beep as the system alert sound on a Macintosh Plus or Macintosh SE, then the beep is generated by code stored in ROM rather than by the **Sound Manager**, and the duration parameter is interpreted in ticks.

If an application has an open sound channel, your call to the **SysBeep** procedure may not generate any sound because only one synthesizer can be active at any time. For example, if an application is using the square-wave synthesizer to play sounds, **SysBeep** may fail to generate the system alert sound because that sound uses the sampled sound synthesizer. Even if the user has selected the Simple Beep as the system alert sound (which uses the square-wave synthesizer), the alert sound will not be heard if some application already has an open channel linked to the square-wave synthesizer. This is because only one channel at a time can be linked to the square-wave synthesizer.

In cases like these where the system alert sound cannot be created, the Operating System flashes the menu bar. Applications using any of the sound synthesizers should dispose of their channels as soon as they have finished making a sound so that the system alert sound can be played. Once again, Macintosh computers without the Apple Sound Chip (that is, the Macintosh Plus and Macintosh SE) operate differently: when the Simple Beep is selected, the beep is heard because the Operating System bypasses the **Sound Manager** to make the sound.

The **SysBeep** procedure cannot be called at interrupt time because doing so may cause the **Sound Manager** to attempt to allocate memory and load a resource.

> **Note:** If your primary use of the **SysBeep** procedure is to alert the user of important or abnormal occurrences, it may be better to use the **Notification Manager**.

The new **Sound Manager** includes two functions-**SndGetSysBeepState** and **SndSetSysBeepState**-that allow you to determine and alter the status of the system alert sound. Currently, two states are defined:

sysBeepDisable	system alert sound disabled
sysBeepEnable	system alert sound enabled

When the system alert sound is disabled, the **Sound Manager** effectively ignores all calls to **SysBeep**. No sound is created and the menu bar will not flash. Also, no resources are loaded into memory. Even when the system alert

sound is enabled, it is possible that the system alert sound will not be played; for example, the speaker volume may be set to 0, or playing the requested system alert sound might require too much CPU time. In these cases, the menu bar will flash. By default, the system alert sound is enabled.