Getting Started with Dreamcast R10

and Mastering a GD-ROM

Table of Contents

Preface

Getting Started with Dreamcast

This document contains material for programmers who are getting started with the Dreamcast development system.

To get set up as quickly as possible, follow the procedures detailed in each section, in order. Start with "Installing the Hardware and Software" and work through each section. This guide concludes with a a brief walkthrough showing how to compile, debug, and execute a sample program.

1. Hardware and Software Requirements

Programming Environment

- IBM PC compatible running Windows 95, 98, or NT 4.0 (Not Windows 2000)
- P200 or greater
- 32 MB of memory
- 500-600 MB of free hard drive space (minimum)
- Adaptec 2940U PCI-to-Ultra SCSI card dedicated to Dreamcast
- A second VGA monitor and/or composite TV monitor
- CD-ROM drive

Supported Art Packages

- SoftImage v.3.51 & 3.7 for SGI or Windows NT
- Alias for SGI v.7.5
- 3D Studio Max R2
- Lightwave 3D

2. Dreamcast System Components

Before you start using this guide, make sure that your Dreamcast development system includes all of the following components:

- Set 5.24 development mini-tower
- 50-pin external SCSI cable
- 2 Active SCSI terminators
- Dreamcast control pad
- Stereo AV cable (composite video + stereo audio)
- Katana Software Development Kit CD-ROM, Release 10

3. Installing the Hardware and Drivers

Sega recommends Adaptec 2940U (PCI-to-Ultra SCSI) or 2940UW (PCI-to-Wide Ultra SCSI) controllers for use with the Dreamcast development system. These controllers have been tested and verified with both the Cross Products' Debug Adapter (DA) and GD-ROM Emulator (GD-M).

Note: When you purchase a 2940UW, make sure the package includes a 68-pin to 50-pin external adapter.

Do not connect the Dreamcast development system to a SCSI controller serving other SCSI devices, such as the Windows boot drive.

During the development process, the debug adapter may lock up or crash, contributing to timing problems and possibly causing data loss.

Although other PCI SCSI controllers may function with the KatanaDA, problems in the GD-ROM file system have been reported when other controllers were used. Cards known to have failed to operate properly include Diamond Fireport 20/40s and older Adaptec ISA 1540/1542s.

The following procedure assumes the use of an Adaptec 2940U controller.

Before installing the Dreamcast development system, make sure your computer has been properly shut down and that the power cord has been disconnected from the wall outlet.

Note: If a dedicated SCSI controller has already been installed for a Set 4 or early Set 5 development system, skip to step 4.

1) Remove the outer case from the computer as described in the owner's manual.

2) Locate an open PCI slot and remove the protective plate covering the opening. If you are installing a second SCSI controller, make a note of the card location dedicated to the Dreamcast development system. (This card may need to be swapped later).

3) Install the SCSI controller according to the instructions provided by Adaptec. Fasten the card in place.

4) Connect one end of the supplied 50-pin SCSI cable to the port on the back of the Set 5.24 labeled `SCSI-A` and the other end to the dedicated SCSI controller card (use the 68-pin to 50-pin adapter if required).

5) Attach the two active-SCSI terminators to the ports labeled `SCSI-B` and For `GD-WRITER`.

6) Connect the Dreamcast controller to the leftmost port (Port A) on the front of the development system.

7) Connect a TV monitor to the composite video and audio jacks on the back of the Set 5.24 using the AV cable provided (or substitute an S-Video cable if available). Optionally connect a VGA monitor to the `RGB Analog (VGA)` port on the back of the development system.

8) Plug the Set 5.24 unit into the wall outlet and press the large power button on the front left corner of the development system to boot it.

9) Wait approximately 15 seconds while the Dreamcast development system completes the power-on sequence. When the hard drive finishes mounting, plug the PC's power cord back into the wall outlet and boot the system.

10) As the PC boots, watch the SCSI controller's status information and verify that the following devices mount:

```
ID 3: CPL KATANA DA
ID 4: CPL GD-M
```

If you have previously installed a development system, skip to step 21.

11) On some PCs with two SCSI controllers, the Windows boot drive may need to be located on the first controller. If no boot device is found after the Katana development system mounts, shut down both computers, and swap the SCSI controller cards. Repeat the installation procedure described above.

Note: Do not attempt to modify the default SCSI controller settings unless absolutely necessary.

---

12) Windows will auto-recognize the Adaptec 2940U card and load the appropriate software. Follow the instructions provided by Adaptec to install the latest `AHA-2940U/AHA-2940UW` SCSI controller drivers.

13) As the system boots, a message should appear stating that an `unknown device` has been found. When the `Update Device Driver Wizard` opens, select `Next` to search for a device driver.

14) The hardware wizard should return with the message `Unable to locate a driver for this device`. At this point, select `Finish`.

No device driver is necessary for the Katana DA.

15) Once the O/S has finished loading, go to the Windows Control Panel and double-click the `System` icon.

16) Click on the `Device Manager` tab and select `View devices by connection`.

17) Expand the Plug and Play BIOS category followed by the `PCI bus` device type to reveal the `Adaptec AHA-2940U/UHA-2940UW PCI SCSI Controller`.

Note: If you installed a second 2940U controller, verify that only two devices are present. If a SCSI controller was swapped, multiple 'false' cards may be listed. In this case, highlight all the conflicting devices and select `Remove`. Repeat steps 12 through 17.

---

18) The `Adaptec AHA-2940U/UHA-2940UW PCI SCSI Controller` should show the `CPL KATANA DA` and `CPL GD-M` units as mounted (see step 10).

Select the `Close` button on the Device Manager control panel to return to the desktop.

19) The Dreamcast development system usually contains a 4 GB hard drive divided into two FAT-16 formatted partitions (2 GB each). If you open `My Computer`, the Dreamcast's internal hard drive partitions should show up as two `Removable Disk` drives. You can read and write files on these drives just as you can on any removable media.

   A bug currently exists in Windows NT preventing the second partition from being seen. If you need the entire 4 GB storage space, the drive can be re-formatted as NTFS or (for Windows 95) FAT-32. For more details, see the section titled "Formatting the Emulation Drive" in the GD-Workshop User Guide distributed with this release.

20) Before continuing with the installation procedure, verify that Windows is running the latest ASPI drivers from Adaptec. Locate the following files in the `Windows\System` folder:

   ```
   C:\WINDOWS\SYSTEM\WINASPI.DLL
   C:\WINDOWS\SYSTEM\WNASPI32.DLL
   C:\WINDOWS\SYSTEM\APIENUM.VXD
   C:\WINDOWS\SYSTEM\IOSUBSYS\APIX.VXD
   ```

   Right click on each file and select `Properties`. Click the `Version` tab and verify that the `File version` number is 4.57 (1008).

   The most recent ASPI drivers are available on the Adaptec web site:

   ```
   http://www.adaptec.com/support/faqs/aspilayer.html
   ```

   Look for the file `Aspi32.exe`.

Note: The default Windows 98 ASPI drivers do not work with the Dreamcast development system. Install the Adaptec drivers to overcome this problem.

---

21) Locate the four DIP switches on the front of the Dreamcast development system. Make certain all four DIP switches are in the OFF (down) position for default operation.

Note: The left two DIP switches determine the standard video output mode reported to software:

Switch 1	Switch 2	Meaning
OFF	OFF	Stereo AV cable, S-video, or RF output
ON	OFF	RGB (NTSC or PAL)
OFF	ON	Undefined
ON	ON	VGA cable

Note: The rightmost DIP switch determines whether the Dreamcast development system boots in self-check mode. This mode outputs to VGA display only. A MIDI loopback cable is required to test the sound system.

Switch 4	Meaning
OFF	Boot to flash memory (default)
ON	Boot to EPROM (run self-check)

22)   Make certain the slide switch on the front of the Dreamcast development system is set to the LEFT.

Note: Setting this switch to the right enables writing to the Dreamcast Boot ROM (flash memory). The Boot ROM contains code to load games, play audio CDs, transfer files to save game cards, etc. This switch has no effect on the Cross Products DA (Debug Adapter) and GD-M (GD-ROM Emulator) flash ROMs.

Your development system may become unbootable if you accidentally write to flash memory when this switch is to the right. Don't try to reflash your system's Boot ROM unless instructed to do so by Sega.

23)   Set the rotary switch on the front panel of the Dreamcast development system to the country for which you're developing:

Code	Country or Region
0	Japan (NTSC)
1	Not used
2	Not used
3	Not used
4	North America (NTSC)
5	Not used
6	Brazil (PAL-M)
7	Argentina (PAL-N)
8	Not used
9	Europe (PAL)
A	Not used
B	Not used
C	Not used
D	Not used
E	Not used
F	Not used

Note: The country code determines PAL/NTSC output. In a production system, this code will also be checked by the boot loader to make certain that the title was published for the country in which it is running.

24) After the DIP switch settings have been verified, install the Dreamcast development software, and continue the hardware setup by following the procedure labeled "Testing the Development System."

4. Installing the Development Software

To install the Katana Software Development Kit (SDK), open the "Katana SDK, Release 10" CD-ROM and double-click the `Setup.exe` icon. Follow the instructions contained within the installation program. Using this method, you may select the entire SDK or `Custom Install` only a few basic components. The full software release requires approximately 500-600 MB of drive space. When the installation completes, the DBFlash utility starts automatically. (Chapter 5, "Flashing the DA and GD-M Firmware and the BOOT ROM," tells how to use DBFlash.)

Note: To un-install the SDK, open the Windows `Add/Remove Programs` Control Panel, select `Katana Development Software`, and click `Add/Remove`.

To avoid confusion, install the current Katana SDK in a different location from the previous release.

5. Flashing the DA and GD-M Firmware and the BOOT ROM

Whenever a new version of the CodeScape debugger or GD-ROM Workshop is released, the Flash ROM files on the Dreamcast development system must be updated. This chapter explains how to re-flash these ROMs.

1)   When DBFlash completes its initialization, the installer autoamtically runs the DBFlash utility.

2)   A four-pane DB-Flash window then opens (see Figure 5.1). You can use the three top panes in the window to flash DA Firmware, GD-M Firmware, and Boot ROM.

Figure 5.1 *DB-Flash window.*

Note: If an error occurs when you try to open the DBFlash window, the problem is most likely caused by old or incompatible ASPI drivers. (For more details, see Step 20 in Chapter 3 "Installing the Hardware and Drivers" for more information.

Warning: Never attempt to flash the system when you suspect that the DBFlash program may be having problems.

3) Referring back to Figure 5.1, note the three text boxes labeled `Current Version` numbers that appear in the top three panes of the DB-Flash window. As you complete the steps that follow, DB-Flash Will use these text boxes to match the latest release of the Dreamcast Development firmware with the version numbers of your system's current DA ROM, your system's current GD-M Firmware, and your system's current Boot ROM.

Also notice the check boxes labeled Reflash that appear in the window's top three panes. If a check mark appears inside any of these check boxes during the steps that follow, that means that the corresponding ROM in your system needs to be updated. If no check marks appear during the steps that follow, flashing will not be necessary.

Warning: You can reflash your system's DA and GD-M firmware yourself, provided you carefully follow the steps outlined in this section. However, If you discover that your Boot ROM needs to be updated, contact Sega Developer Support for instructions. Don't try to flash your Boot ROM without contacting Sega! If you do, you could easily damage your Dreamcast Development System.

4) To start the job of reflashing your system's firmware, click the `Browse` button in the top (DA Firmware) pane of the DB-Flash window.

DBFlash responds by opening a file chooser dialog.

5) Using the file chooser, navigate to the `Genie.fsh` Flash ROM file, which resides in the `\Utl\Dev\Codescape` folder. Then double-click on the file's icon to select it.

6) If your DA Firmware needs to be updated, two things now happen: A check mark appears inside the Reflash check box in the top (DA Firmware) pane of the DB-Flash window, and the version number of the newest DA Firmware release appears inside the text box just the right of the updated check box. If these two changes occur, you will know that your systems' DA Firmware needs to be reflashed.

But it is not time to do that yet. Instead, for now, just move on to Step 7.

7) Navigate to the `gmX_X_Xx.fsh` Flash ROM file (where X's stand for version numbers). This file resides in the `\Utl\Dev\GDWorkshop` folder. When you have located it, double-click on its icon to select it.

8) If your GD-M Firmware needs to be updated, two things now happen: A check mark appears inside the Reflash check box in the second (GDM Firmware) pane of the DB-Flash window, and the version number of the newest GD-M Firmware release appears inside the text box just the right of the updated check box. If these two changes occur, you will know that you need to reflash your systems' GD-M Firmware.

9) If neither your DA Firmware nor your GD-M Firmware needs to be updated, the DB-Flash window now looks like the one shown in Figure 5.2 on page 15.

The version numbers of your system's DA and GD-M Firmware match the corresponding version numbers of the latest Dreamcast firmware release, so you can now close the DB-Flash window without performing any flashing operations.

Figure 5.2 *What the DB-Flash window looks like when no reflashing is required.*

10) Although the DB-Flash window also contains a Boot ROM pane, you should not use it without first contacting Sega Developer support.

   As mentioned previously, attempting to reflash your system's Boot ROM by yourself can easily damage your system.

11) If the preceding steps revealed that your DA Firmware needs updating, or that your GD-M Firmware needs to be updated (or both), click the Reflash button in the lower right corner of the DBFlash window. When the flashing process begins, DBFlash displays a status bar at the bottom of the DBFlash window to inform you that the operation has started.

12) It takes about 25 seconds to reflash your system's Firmware. Then DBFlash displays a Success dialog like the one shown in Figure 5.3.
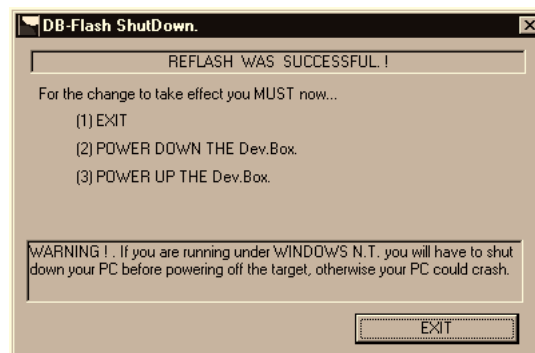

Figure 5.3 *The DB-Flash "Success" dialog.*

13) After noting the instructions displayed in the bulleted list in the DB-Flash window, close the window by clicking the Exit DB-Flash button.

14) Shut down your PC by choosing `Shut Down` from the Windows Start menu. If Windows does not turn your PC off by shutting down its power, click your PC's On/Off switch to power it down manually.

15) Manually turn off the Dreamcast dev box by clicking its power button.

16) Wait at least 30 seconds, and then turn your dev box back on again. This wait period is important! Waiting more than 30 seconds won't hurt; waiting less than 30 seconds might.

17) Reboot your PC and verify that the DA and GD-M devices mount, as described in Step 10 of "Installing the Hardware and Drivers."

Note: In some cases, Windows may report after a reboot that an `unknown device` has been found. When the `Update Device Driver Wizard` opens, select `Next` to search for a device driver. When the message `Unable to locate a driver for this device` appears, select `Finish`. No device drivers are necessary for the DA or GD-M devices.

18) After your Flash ROMs have been updated, you can test the functionality of the development system by running a utility called `DACheck`.

DACheck is located in the `\Utl\Dev\DACheck` folder. From the DACheck main window, click the `Test DA` and `Test GD-M` buttons.

If each component tests OK, the corresponding status light will be green. A red light indicates failure.

Note: At any time during the preceding series of operations, if a problem occurs or contact is lost with the Debug Adapter, shut down Windows and turn off the Dreamcast development system. Again, be sure to wait at least 30 seconds before restarting the Set 5.

Recovery from a bad Flash ROM

If contact is lost with the Debug Adapter due to an error in the flashing process, the system may be recovered using the following procedure:

1) After shutting down both Windows and the Set 5.24 unit for at least 30 seconds, restart the system and start DACheck.

2) Click the `Exit` button. A `DACheck Reset Options` dialog box will appear.

3) Check the box labeled `Set Factory Defaults` and click `Exit DACheck` to continue.

4) After a brief reset period, DACheck will close. Shut down both the Set 5.24 and PC and wait at least 10 seconds before rebooting.

Note: If the previous operations do not work, try to boot into self-check mode by following the instructions given in Step 4 of Section 8, `Testing the Development System`.

5) Re-launch DACheck. The system status should have returned to normal.

If the DA indicator light is still red, repeat the above procedure.

Do not try to re-flash the Set 5.24 when the light is red unless the Flash ROM appears permanently bad. In these cases, revert to the last known working version of the Flash ROM.

6. Using the CodeScape Debugger

When the Dreamcast development system is completely installed and has been updated with the latest firmware, you can begin compiling, debugging, and executing code. To download and run Shinobi executables on the SH-4, use Cross Products' CodeScape debugger.

The quickest way to verify that the hardware is functioning properly is to load and run a pre-compiled sample program. The following procedure tells how to use the basic features of CodeScape to execute the `QUIKTST2` sample.

Note: The `QUIKTST2` sample is provided as a pre-compiled `.elf` file in the R10 SDK. If you wish to test other samples, you must compile them before proceeding with this exercise. (For details, see Chapter 9, "Compiling the Samples.")

---

1) Before you load CodeScape, you must make sure that a file system is present. To obtain that assurance, load GDWorkshop and locate the Control & Log window, a child window that appears in the bottom part of the GDWorkshop window. In the Control & Log window, locate the emulation mode button, shown below. The yellow circle inside the emulation mode button represents a disk, and the gray circle above the yellow circle represents emulation mode. Set the switch next to the two circles to point downward, toward the yellow circle (the disk), as shown in the illustration.

2) Start CodeScape by double-clicking the `CodeScape.exe` icon in the `\Utl\Dev\CodeScape` folder. CodeScape will display a dialog box warning that a reflash has occurred. This is not really an error, so press `OK` to continue.

3) Choose `Configure DA Settings` from CodeScape's Debug menu to find out what mode CodeScape is running in. For general debugging purposes, you should run Shinobi code in CPU Mode. `OS Mode` is similar to `CPU Mode`, but after a reset executes the Dreamcast boot ROM and attempts to launch a bootable GD-ROM disk (or emulated project).

4) The first time you run CodeScape, it may display a message warning you that it is `Unable to restore toolbar state`. If you see such a message, click `OK` to continue.

5) From the `File` menu, choose `Load Program File`.

---

6) When the Load Program File dialog opens, click the `Browse` button to bring up a standard file dialog. Then navigate to the `QUIKTST2.elf` sample executable, which resides in `\Sample\Kamui2\QUIKTST2\Exe`.

7) Leave the `Load Options` and `Reset Options` items at their default settings. Check the `Enable Run Options` box, and select `Run to main`.

8) Click `OK` to load `QUIKTST2.elf`. At this point, make sure that a TV set or a VGA monitor is plugged into the back of the Dreamcast development system and that the appropriate DIP switch settings for video output are set. (For the correct DIP switch settings, see Step 21 in Chapter 3 "Installing the Hardware and Drivers.")

9) From CodeScape's `Window` menu, select `New Window` (Ctrl-N). CodeScape then opens a blank window with `No type`.

10) From the `Region` menu, pull down `Type` and select `Source`. You can modify the window type by clicking the desired toolbar icon.

11) `QUIKTST2.elf` contains hard-coded paths to `\Sample\Kamui2\QUIKTST2\Exe`. If the SDK has been installed in a different directory, Select `Edit Source Path` from CodeScape's `Project` menu.

12) Click the `Browse` button to bring up a standard file dialog or use an MS-DOS window to enter the path to the `QUIKTST2` source folder.

Note: When the Program Counter (PC) is located in a section of code with no source or debugging information, the following message is displayed in the source window:

```
**** No debug info. available for address <addr> ****
```

---

13) The source window now displays the following from `NjLoop.c`:

```
=> int main (void)    /* Arrow (=>) represents PC. */
  {
   KMSYSTEMCONFIGSTRUCT kmsc;                    /* Kamui 2 system...
   KMVERSIONINFO       VersionInfo;             /* Kamui 2 version ...
   KMSURFACEDESC       PrimarySurfaceDesc;      /* Primary framebuf...
  ...
  }
```

14) From the `Debug` menu, pull down `Execution` and select `Run` (F9). A spinning QUIKTST2 is now displayed on the TV/monitor.

15) To expedite the process of loading and running the `QUIKTST2` sample, save the current session by selecting `Session Save` from the `File` menu. This records directory paths, window states, break points, etc.

You may want to continue testing the Dreamcast development system by running additional sample programs that exercise the Maple controller box and GD-ROM file system. Read through the following two sections for information on how to do this.

7. Using GD Workshop

To learn how to burn a GD-ROM, see Chapter 11 "Burning a GD-ROM."

The Dreamcast system requires the use of a special kind of disk that resembles an ordinary CD-ROM but is designed to different specifications.

This special variety of disk stores more information than an ordinary CD-ROM. It is called a gigabyte disk, or GD-ROM.

A GD-ROM has a playing time of approximately 120 minutes, and holds Approximately 1 gigabyte of data (In comparison, an ordinary CD-ROM has a playing time of about 79 minutes, and stores about 650 MB of data.)

If you look closely at a Dreamcast GD-ROM, you can see that it is divided into two different physical areas — a single-density area that holds just 4 minutes of data, and a high-density area, which can hold approximately 112 minutes of data. The inner ring of a GD-ROM is its single-density area, while its high- density area is its larger outer ring.

The single-density area of a GD-ROM has the same layout as a conventional CD-ROM, except that it has a capacity of only about 35 MB of data. This area is divided into two tracks: a data track (Track 1) and an audio track (Track 2).

Track 1 contains data that is visible if the user places the GD-ROM into a CD-ROM drive on a PC. On Track 2, you must store a `Warning.DA` file that you can find in the `\Utl\Dev\GDWorkshop` directory. This file contains a warning message that alerts the user when an attempt is made to play a Dreamcast GD-ROM on an ordinary CD-ROM drive. You'll learn how to copy the "Warning.DA" message to Track in the next section "Getting Started with the GD-ROM Workshop."

When you record data on a GD-ROM, it is placed on the disk in the form of tracks. A GD-ROM has two kinds of tracks — Mode1 tracks (that's `Mode-1`, not `Model`) and CDDA (CD Digital Audio) tracks. Mode1 tracks are used for data storage, and CDDA tracks are used for storage of raw audio data.

Hence, Track 1 of the low-density area of a GD-ROM is a Mode1 (data) track, while Track 2 is a CDDA track.

The high-density area of a GD-ROM contains one Mode1 (data) track, followed by up to 95 CDDA tracks, followed in turn by another Mode1 track. The single- density area contains just two tracks: a Mode1 track and a CDDA track.
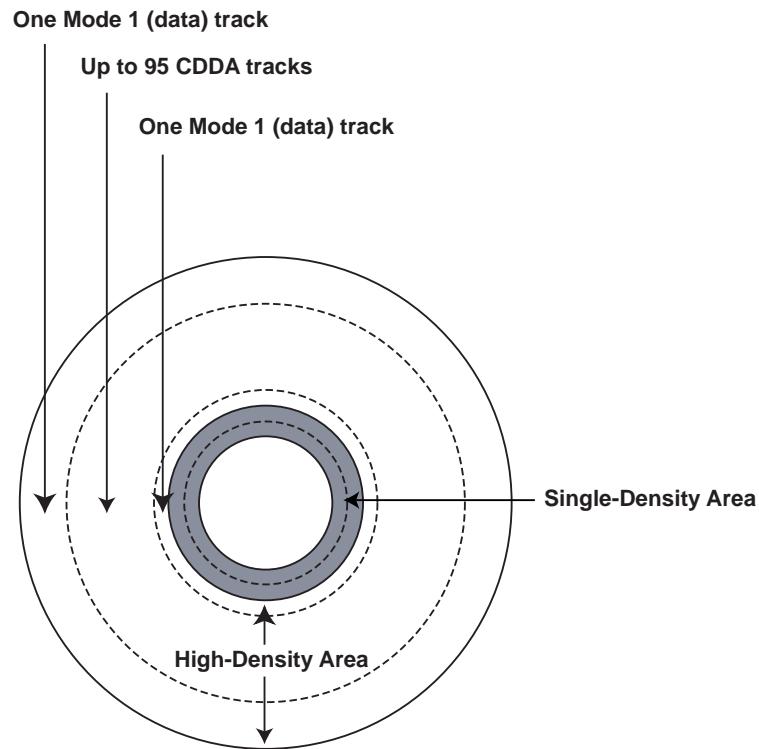
Figure 5.4 *Layout of the GD-ROM.(not drawn to scale) shows the layout of a GD-ROM disk.*

Getting Started with the GD-ROM Workshop

The GD-ROM Workshop (GDWorkshop) utility allows you to lay out and burn GD-ROMs. It also controls the emulation of GD-ROMs on the Dreamcast Development system's hard drive.

Use the following procedure to (1) set up the GD-M emulation environment and (2) test it by executing a sample program that accesses the file system.

1) Start GDWorkshop by double-clicking the `GDWorkshop.exe` icon in the `\Utl\Dev\GDWorkshop` folder.

2) The first time you run GDWorkshop, you may see a message warning you that `Workshop was unable to determine your emulation configuration`, and stating that `you will be taken to the configuration screen`. If such a message appears, click `OK` to continue.

3) Open the Configuration menu and choose `Configure drive letters`. GD-Workshop responds by opening a `Configure drive letters` window.

4) When the `Configure drive letters` window opens, it lists the available SCSI controllers on your system, along with any attached devices that match the characteristics of the GD-M Emulator. For example:
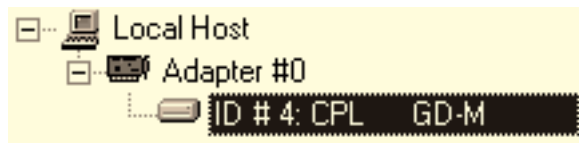
Figure 5.5 *The Configure drive letters window*

5) Highlight the `CPL GD-M` device. On the right side of the display is a pull-down menu containing drive letter assignments. Select an appropriate drive name and click `Add`. Repeat this process for the second drive partition (if present). When you are finished, the lower portion of the window labeled `Current drive selections` should list:

```
E: 0,4 CPL GD-M  (assuming drive E: is the first available)
F: 0,4 CPL GD-M
```

Note: Be careful not to select other removable drives, such as a `Zip` drive, as a Kamui 2 emulation device.

---

6) When you are finished assigning drive letters, press `OK` to exit. Re-launch the GDWorkshop application to continue.

7) From the `File` menu, select `New` (Ctrl-N) and supply a name for your project. For this example, use the title `Samples`. This action opens the GDWorkshop window, which displays a pre-built GD-ROM disk structure in a panel on the left and a number of user-modifiable tab menus in a panel on the right.

   In the tree display shown in the left panel of the GDWorkshop window, `Session 1` represents the single-density area of the GD-ROM, and `Session 2` represents the high-density area of the disk.

8) Leaving GDWorkshop running in the background, use the Windows Explorer to locate the folder `\Sample\Kamui2\quiktst2\gd_root`. Open this directory and select all the files inside it. These are user data files. In the next step, they will be copied into one of the two Mode1 tracks in the high-density area (`Session 2`) of the GD-ROM.

9) Drag and drop the selected files onto the track labeled `05 - MODE1`, at the bottom of the GD-ROM tree. This action opens a file transfer window that displays a progress bar for each file as it is copied to the tree that represents the GD-ROM.

10) When you have finished dragging and dropping, the root directory of Track 5 will contain several `.pvr` files used by the `QUIKTST2` demo, and the tree shown in the GDWorkshop window will have a structure something like this:

```
GD-ROM
   +-- Session
       +-- ISO Level 2
           +-- 01 - Mode 1
           +-- 02 - CDDA
   +-- Session
       +-- ISO Level 2
           +-- 03 - Mode 1
           +-- 04 - CDDA
           +-- 05 - Mode 1
```

Notice that Session 1, which represents the single-density area of the GD-ROM, now has its allotted two tracks (a Mode1 track and a CDDA track), which the high-density area of the disk now has its minimum allotment of one Mode1 track, one or more CDDA tracks, and another Mode1 track.

11) Some files now need to be added to the first two tracks on the disk (those in Session 1). Highlight Track 2 (02 - CDDA) and right-click the mouse to bring up an Options menu. Then choose `Insert | File`. GDWorkshop responds by opening a file chooser dialog.

12) From the file chooser, select `Warning.DA` located in the `\Utl\Dev\GDWorkshop` directory.

13) Drag and drop the `Warning.DA` file onto Track 2 of Session 1 (02 - CDDA).

14) Drag and drop the same file `Warning.DA` — onto Track 1 of Session 1 (01 - MODE1), and into Track 3 (03 - MODE1) and Track 4 (04 -CDDA)

Note: The `Warning.DA` file that is copied to Tracks 1 and 2 in the preceding steps contains seconds of 44.1 kHz, 16-bit, stereo audio intended to warn the user that the disk is designed for a Dreamcast game system and is not meant to be played on an ordinary CD player.

Track 2 is the only track on which the `Warning.DA` file is required. However, `Warning.DA` is also copied to Tracks 1, 3, 4, and 5 in this exercise because a four seconds of data must be placed on each of those tracks as padding, and the `Warning.DA` file just happens to contain four seconds of audio data, making it an ideal file to place on those tracks, strictly from the point of view of the amount of data it contains. If you have a file of your own that contains four seconds of data, you can place it on Tracks 1, 3, 4, and 5 in place of the `Warning.DA` file. Just remember that you must place the `Warning.DA` file supplied by Sega on Track 2.

15) For reasons you will learn about in Chapter 10, "Burning a GD-ROM," Track 3 requires some special treatment. After you copy the `Warning-DA` file (or some other four-second file) onto Track 2, you must pad, or buffer the rest of the track with zero data. Here's how to do that:

1. Highlight Track 3 in Session 2 in the tree structure shown in Figure 7.4.

2. Select the `Track` tab in the GDWorkshop window.

3. Check the `Pad Track` check box.

4. Click the `Apply` button at the bottom of the GDWorkshop window. (If you don't see it, you may have to scroll down until it appears.)

16) Locate the four icons in the task bar of the `Control & Log` window, which appears in the bottom part of the GDWorkshop window (Figure 5.6). The rightmost icon in the Control & Log window's task bar (the one made up of two circles and a line) shows a switch that can point to either one of two circles: a yellow circle, which represents a disk, and a gray circle, which represents emulation mode. Point the switch up - away from the yellow circle and toward the emulation mode circle - as shown in the illustration.
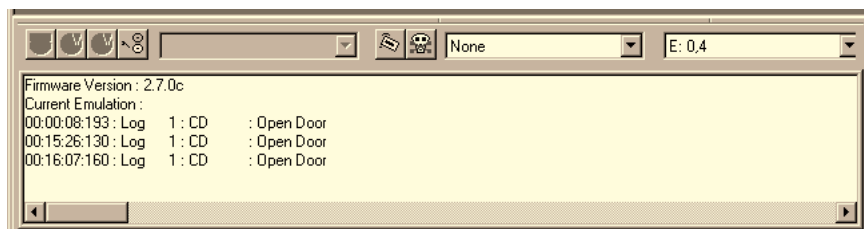

Figure 5.6 *The Control & Log window*

17) The leftmost icon in Control & Log window represents the state of a `virtual "Drive Door"`. When this virtual door is open, you can add and delete files in a GD-ROM project. When the door is closed, the disk inside it is said to be in `emulation mode`. To begin a process of emulating the `Samples` disk, click the `Drive Door` icon (F9).

Note:	Changes to the project do not take effect until the drawer is opened and closed again.

18) After the virtual drive door is shut, GDWorkshop prints out several messages to the log window indicating that the file system is being prepared. When it has finished mounting, the text `Close Door` is displayed. At this point, you can minimize the GDWorkshop window.

Note:	The disk continues to emulate until either the power is turned off on the Set 5.24 or the virtual drive door is opened. GDWorkshop can be quit without halting the emulation.

19) Launch the CodeScape Debugger. Follow the procedure detailed in the previous section for loading and running an executable file. Load the `QUIKTST2.elf` example located in `\Sample\Kamui2\QUIKTST2\Exe`.

20) As the `QUIKTST2` sample is executed within CodeScape, texture files are read from the GD-ROM file system. If GDWorkshop is still running in the background, status messages are printed to the log window listing the `.pvr` files as they are loaded.

If any error occurs during the emulation process, perform a hard reset of the Dreamcast development system (using CodeScape, select `Reset, Hard Reset` from the `File` menu). Wait a few seconds, then open and close the `Drive Door` in GDWorkshop.

To learn more about GDWorkshop and GD-ROM disks, see Chapter, "Burning a GD-ROM."

8. Testing the Dreamcast Development System

To perform a hardware self-test of the Dreamcast development system, use the following procedure:

1) Launch the DACheck utility, then select the `Exit` button. A `DACheck Reset Options` dialog box will appear.

2) Select the `OS Mode` radio button and click `Exit DACheck` to continue.

3) With default DIP switch settings (all OFF), the Dreamcast boot ROM will execute and display the consumer interface to load games, play audio CD's, transfer files to save game cards, etc. The output will be to a composite (television) monitor.

4) To enter self-check mode, flip the fourth DIP switch on the front of the unit ON and press the `Reset` button. The output of the self-check is VGA only.

5) The self-check takes several minutes to complete. As each component passes, its test description line turns green and an `OK` is appended. If a test fails, the line is colored red.

Note: To test the full functionality of the audio system, a MIDI loopback cable is required. If not present, the `AICA & ARM check.` will report failure.

9. Compiling the Samples

There are three ways to compile and link an individual sample program:

- Using an MS-DOS console window.
- Using Metrowerks CodeWarrior with the Hitachi compiler.
- sUsing Metrowerks CodeWarrior with the GNU compiler.

To compile a sample program using an MS-DOS window, follow these steps:

1) From Windows, navigate to the directory containing the sample you want to compile. For example, to compile the Teapot program, go to `C:\Katana\Kamui2\QUIKTST2` (assuming that's the location of the Teapot source files on your system).

2) Run the GNUMAKE utility by executing the command

        GMAKE

Note: Although you use the GMAKE command in this step, Dreamcast programs are actually compiled using the Hitachi compiler.)

---

3) At this point, you are compiling the QUIKTST2 program. During compilation, two new folders are created: an Exe folder and an Obj folder. (The object files and some intermediate files are deposited in the Obj folder. The `.elf` file (the executable file format for Dreamcast) is placed in the Exe folder.)

4) In this example, we use a default makefile. You can customize this makefile setting several different flags. To edit the makefile for customization, open the file in an editor and locate a group of four lines that look like this:

```
OUTPUT          = ELF
OPTIMIZE        = TRUE
DEBUG           = TRUE
#LIST           = TRUE
#COMPILER       = SHC
```

You can use each of these lines to set or clear a flag.

OPTIMIZE: The OPTIMIZE flag optimizes for speed (TRUE = optimizations on, FALSE = optimizations off). Turning optimizations off helps you in the debugging process by generating lines linearly instead of moving them around to make code as compact as possible.)

DEBUG: The DEBUG flag specifies whether you want debugging symbols embedded in your code. Turn this flag on for debugging, off for the release version of your game.

LIST: For every file the compiler processes, the compiler outputs a .lst file, which contains the assembly-language code generated by the compiler. This file is placed in the Obj folder. The linker outputs a .map file that shows all the modules which have been linked into your application, and where they have been placed in memory.

OUTPUT: Specifies the output file type. The file type can be `BIN` (for `1ST_READ.BIN`), `LIB` (library), or `ELF` for a debug executable.

10. Burning a GD-ROM

Note: To understand the material presented in this section, you should first read Chapter 7, "Using GD-Workshop."

As noted in Chapter 7 "Using GD-ROM Workshop," Tracks 1 and 2 of the single- density area of a GD-ROM must contain 4 seconds of data each. You may recall that in Chapter 7, you were instructed to copy a file named `Warning.DA` onto these two tracks.

You may also recall that although the specific contents of the `Warning.DA` file were not significant in this operation, the fact that the file contains four seconds of data is significant. That's because the `Warning.DA` file contains just enough information to provide these two tracks with something they require: padding, or a buffered area, that is at least four seconds long.

Tracks 1 and 2 of Session 1 are not the only tracks on the GD-ROM that require a buffered area at least four seconds long. As mentioned earlier, every track that you place on the GD-ROM must contain at least four seconds of data.

Track 3

It's particularly important to pad Track 3 (the first Mode1 track of the disk's high-density area) by filling it with zero data. That's because the `1ST_READ.BIN` file — your actual game executable— is always placed on Track 5, which must always lie outside the disk's 100-minute mark.

Track 3 is used only for padding. To pad it, you fill it with zero data beginning at its first sector. When you do that, GD-Workshop forces any data files on the track to occupy the end of the track. GD-Workshop also calculates the size of the track, and arranges the data on it in such a way that your game's executable always winds up on the outside edge of the disk.

To pad Track 3, follow these steps:

1) Highlight Track 3 in Session 2 in the tree structure shown in Figure 5.

2) Select the `Track` tab in the GDWorkshop window.

3) Check the `Pad Track` check box.

4) Click the `Apply` button at the bottom of the GDWorkshop window. (If you don't see it, you may have to scroll down until it appears.)

Track 4

At least one CDDA track (raw digital audio) is required on Track 4 of Session 2.

Even if your game doesn't use digital audio, you still must place at least four seconds of data on Track 4. You can add new CDDA tracks by following these steps:

1) Right-click the Session icon in the GDWorkshop window's tree display. A popup menu then appears.

2) Select `Insert | Track | Digital Audio`.

When you add a new CDDA (digital audio) track, GD-Workshop renumbers your last Mode1 track. When you have finished adding tracks, your GD-ROM will have, from the inside of the disk moving outward, a single-density-area track, a Mode1 track, up to 95 CDDA tracks, and a final Mode1 track (which stores your game data).

As mentioned previously, the first Mode1 track on the GD-ROM is just padding to make sure that later Mode1 tracks and CDDA tracks are out as far out as possible toward the outside of the disk.

Compiling a Bootable File (a 1ST_READ.BIN file)

When you have placed all the necessary tracks on your disk, you're ready to start the job of GD-mastering; that is, the process of creating a bootable GD-ROM. To show you how to do that, we'll use the `QUIKTST2` example.

These are the steps to follow:

1) From Windows, locate the `\Sample\QUIKTST2` folder.

2) In a text editor, open the appropriate makefile for editing.

3) Navigate to bottom of makefile and find the `OUTPUT` option.

4) Change the `OUTPUT` option in the makefile to `BIN`.

5) Recompile the sample program (see Chapter 9 if you need more instructions).

Track 5

Track 5 is designed to store only game data, including all remaining data files and the `1ST_READ.BIN` file — which, as you have seen, is your game's actual executable.

To create a game executable in bin format, start by opening the makefile in an editor and changing the comments in the makefile as shown in the following code fragments.

When you open the makefile, notice that the `OPTIMIZE` flag is set to 1, and that the `DEBUG` flag is set to `TRUE`. Also note that the `LIST` flag is set to `FALSE`.

To edit the makefile, set the `OUTPUT` flag to `BIN`, and set the `DEBUG` flag to `FALSE`.

Note: When these changes are made, you must recompile your executable. When you recompile the sample file, you'll create a file named `samplename.bin` file in the Exe directory.

Creating the IST_READ.BIN File

To create a `1st_READ.BIN` file — that is, an executable file for your game—Follow these steps:

1) Locate the file, `QUIKTST2`.bin inside the `\Sample\Kamui2\QUIKTST2` directory.

2) Change the name of the `QUIKTST2`.bin file to `1ST_READ.BIN`.

3) Insert `1ST_READ.BIN` into Track 5 of the GD-ROM project.

Note: The last file added to a track is physically located at the end of the track. This is where `1ST_READ.BIN` should be. If you need to reorder the tracks on your disk, you can do that by choosing the `List View` option from GD Workshop's `View` menu. When the left panel of the GDWorkshop window displays a list view, you can drag and drop files shown in the list view to change their order. When you finish reordering your files, you can use the View menu to switch the left panel of the GDWorkshop window back to a tree view.

---

4) Create a file named IP.BIN file. To do this:

1. Navigate to the `\Utl\Dev\IPmaker` directory.

2. Execute `ipmaker.exe`.

4. When the IPMaker window opens, find the check boxes that let you specify whether your game will be used in the U.S., in Europe, or in Asia. For now, check all three boxes.

5. Fill out the other edit fields in the window, such as your game title, your development company, and your publishing company. When you're Satisfied that you have filled out all those fields correctly, click the `Apply` button at the bottom of the window. That action creates a pair of files named ip.bin and ip0000.bin in the `\Utl\Dev\IPmaker` directory. These files are the initial program files that the Dreamcast unit will load.

6. At the bottom of the window, there's a box that lets you select a filename. The filename `1ST_Read.bin` appears there by default. Leave the default setting alone; that's the name you'll want to use.

7. There's also a `Peripherals` tab in the IPMaker window. If your game supports peripherals other than the standard Dreamcast controller, click the `Peripherals` tab and check all boxes that apply to the controllers you plan to use.

Note: If you want your game to support VGA output, check the VGA box, which by default is unchecked.

---

8. Close the IPMaker window.

Making a Disk Bootable

When you have finished working with the `lst_READ.BIN` file and have created An `IP.BIN` file, a few more steps are needed to make your GD-ROM bootable:

1) Open GD Workshop.

2) Copy these three files into Track 1 of Session 1:

```
BIBLIOGR.TXT
COPYRIGH.TXT
ABSTRACT.TXT
```

`BIBLIOGR.TXT` is a file in which you can place a bibliography — that is, a list of reference materials that you used to create your game.

The `COPYRIGH.TXT` file contains important copyright information. In the `ABSTRACT.TXT` file, you can place a description of your game. Near the release of Dreamcast, more details about what these files should contain will be available. Meanwhile, you can create empty files with these names and copy them into Track 1 of Session 1 (01 - MODE1).

3) Select the track tab and pad the track by following these steps:

1. Select the `Disk` tab. You'll then see two different header boxes equipped with `Browse` buttons. The first box says, `System area file (single)`. The second says, `System area file (high)`. Select the browse button on `System area file (single)`.

2. Navigate to the `\Utl\Dev\Ipmaker` directory.

3. Select `ip0000.bin` and click `OK`.

4. Select `System area file (high)`, and then select the `Browse` button.

5. Navigate again to the `\Utl\Dev\Ipmaker` directory.

6. Select the `ip.bin` file.

7. Click the `Disk` tab and then click the `Apply` button (you may have to scroll to see it).

A Final Check

When you complete the preceding steps successfully, you will have a bootable disk. Then only one task remains: to test it to make sure that it actually is bootable.

To do that, follow these steps:

1) Check to make sure that your dev station is in OS mode. (For instructions on how to do that, see Step 2 in Chapter 6, "Using the CodeScape Debugger.")

2) When you have verified that your dev station is in OS mode, close the virtual drawer in GD-Workshop by clicking the button in the lower left part of the window. Then physically reset the dev station. If you see a message asking if you want to reload the program file, answer `No`.

Note: If there are errors on the disk or it isn't set up properly, you'll see error message boxes when you click the drawer. These messages will tell you what the errors are.

3) When you have reset your dev station, the files to be written to your bootable disk will be displayed in the GD-Workshop log window.

Note: There is file that you can add to your last Mode1 track. It's a 256x256 file named "`texture 0gdtex.pvr.`" It allows you to customize what the texture will be on the picture of the GD that you see spinning around on your last Mode1 track. If you like, you can change that default texture to your company's logo, or whatever else you think would be cool to display there.

Two Ways to Burn a Disk

If everything has worked properly up to now, it's time to burn a disk. Currently, there are two ways to do that:

- With a GD-Writer connected to a dev station. (Currently, this is the only way to burn a disk that can be submitted to Sega DTS for processing.)
- With a GD-Writer connected to a PC. (Connecting GD-Writers to PCs can be a convenient and relatively inexpensive method for making copies of a disk for in-house distribution to members of a development team.)

10.0.1 Burning a Disk with a GD-Writer and a Dev Station

To burn a disk using Method 1 — a GD-Writer connected to a dev station — here are the steps to follow:

1) On the back of the GD-Writer, locate the SCSI dial switch and make sure it's set to 5.
2) Connect the GD-Writer to the GD-Writer SCSI connector on the back of the Set to 5.
3) Shut down your dev station.
4) Shut down your PC.
5) Turn on the following devices, in this order:

    1. The GD-Writer.

    2. Your dev station.

    3. Your PC.

6) When all those devices are up and running again, start GD-Workshop.
7) From the Configuration menu, select `SCSI Bus`.
8) When the SCSI Bus Configuration window opens, select the GD-Writer.
9) Click the button that says, `Add as CDR`.
10) Load the QUIKTST2 project.
11) From the `Emulator` menu, select `Test Write`.

    The test write process takes about an hour. During this process, GD-Writer it actually goes through the disk-burning process, but without actually turning on the laser. Carrying out this procedure ensures that the GD-Writer and the dev station are communicating properly.

12) When your test write is complete, select `Write` the emulator menu. This action starts your actual burn. A real burn also takes about an hour.

When your burn is complete, it's time to test it. To do that, follow these steps:

1)   In GD-Workshop, locate the rightmost icon in the Control & Log window's task bar (the one made up of two circles and a line). That icon represents a switch that can point to either one of two disks. Verify that the switch is set in `GD-ROM` mode, not in `GE Emulator` mode.

     GD Emulator mode allows you to boot from the GD-ROM in the Set 5 rather than from the virtual drive.

2)   Insert the GD-ROM into the Set 5's GD-ROM drive.

3)   Physically reset the dev station.

4)   Your disk now boots. If it does not, here are some things to check:

   • Make sure you have set the date or time in your boot ROM. If you don't do that, your disk won't boot.

   • By default, the dev station is in auto-boot mode. If for some Reason this setting has been changed, you can put the dev station back into auto-book mode by setting the `Plan disk manually` option in the Dreamcast opening screen.

   • If you have created your `IP.BIN` file correctly and `1ST_READ.BIN` file is in the correct location on the GD-ROM, you'll see the SEGA copyright screen when you boot your disk. If your `IP.BIN` file has been created incorrectly or if your `1ST_READ.BIN` file is not in the correct position on your GD-ROM, you won't see the SEGA copyright screen; instead, you'll see the main Dreamcast opening screen and a warning message. If that happens, make sure that you have created your `IP.BIN` file correctly and that your `1ST_READ.BIN` file is in the correct location on the disk.

If you have checked all these items and your system crashes or you see a black screen when you try to run your game, perform the following checks:

   • If you see a SEGA logo but your game does not run, verify that you created your 1ST_READ.BIN file correctly. (For instructions on how to perform that check, see the subsection titled `Creating the 1st_READ.BIN File` on page 31 of this chapter.

   • Check to see whether you have inadvertently left any debugging calls in your game's source code. (If you have done that, your game will work correctly on the Dreamcast dev station, but not on the Dreamcast console.)

   • Check to see whether the stack has exceeded its maximum size of 64K.

10.0.2 Burning a Disk with GD-Writers Connected to PCs

When you want to burn GD-ROMs for in-house use by members of a development team, you can connect one or more GD-writers to PCs and then follow these steps:

1) Create an emulation project and test it using GDWorkshop (see "A Final Check" on page 32).

2) In GDWorkshop, choose "Save CRI Script…" from the File menu.

Note: If you aren't certain at this point that you have everything set up correctly, you can test your setup by executing the `crfdc.exe` command with the options "wenv=test `lang=e -wm=imagefile -bld-`*scrfile* " (where *scrfile* is the CRI script) before you perform Step 3.

---

3) Open the "`\Utl\Dev\CDCraft`" directory and execute the `crfdc.exe` command with the options "`lang=e -bld-`*scrfile*" (where *scrfile* is the CRI script). This operation will generate a disk image file.Transfer the disk image file to any number of PCs connected to GD-Writers. (When making this connection, make sure that the SCSI ID is *not* 0.

4) To burn a disk, execute the `crfdc.exe` command with the options "`lang=e -wm=imagefile -bld-`*scrfile*." (These options are the same as those used in Step 3, with the `-wm=imagefile` option added.)

Note: If you wish, you can use the `-num=n` option to repeat the burn process. Currently, however, you can use only one GD-Writer per PC.

10.0.3 Generating an Uncompressed File

The procedure described above generates a compressed image file. CDCraft creates a compact image file because it knows the disk is to be padded, but it doesn't put the actual filler data in the file.

To generate an uncompressed image file that is suitable for submission to Sega for pressing, you must create a full 1GB image file. To do this, add the "`-benv-imfl`" option to the "`-bld=`*scrfile*" command mentioned in Step 3 of "Burning a Disk with GD-Writers Connected to PCs" on page 35.

In addition to creating a full image, you will also need to generate a text-format TOC file that Sega can use to verify the contents of your GD-ROM. To create a TOC file, run CDCraft with the option "`-tck=myimagefile`."