APPENDIX D

Cisco 4000 Series Virtual Configuration Register

This appendix describes the Cisco 4000 series virtual configuration register, the factory-default settings, and the procedures for changing those settings.

Virtual Configuration Register Settings

The Cisco 4000 series has a 16-bit virtual register, which is written into the nonvolatile random access memory (NVRAM). Use the processor configuration register information contained in this appendix to do the following:

- Set and display the configuration register value
- Force the system into the bootstrap program
- Select a boot source and default boot filename
- Enable or disable the Break function
- Control broadcast addresses
- Set the console terminal baud rate
- Load operating software from ROM
- Enable booting from a Trivial File Transfer Protocol (TFTP) server

Table D-1 lists the meaning of each of the virtual configuration memory bits.

Table D-1 Virtual Configuration Bit Meanings

Bit No.	Hex	Meaning
00–03	0x0000–0x000F	Boot field (see Table D-2)
06	0x0040	Causes system software to ignore nonvolatile memory contents
07	0x0080	OEM bit enabled
08	0x0100	Break disabled
10	0x0400	IP broadcast with all zeros
11–12	0x0800–0x1000	Console line speed
13	0x2000	Boots default ROM software if network boot fails
14	0x4000	IP broadcasts do not have net numbers
15	0x8000	Enables diagnostic messages and ignores NVRAM contents

Changing Configuration Register Settings

Some common reasons to modify the value of the virtual configuration register follow:

- Recover a lost password
- Change the console baud rate
- Enable or disable the Break function
- Manually boot the operating system using the **b** command at the ROM monitor prompt
- Force the router to boot automatically its system image in Flash memory, or boot as per any **boot system** commands that are stored in its configuration file in NVRAM

Note If the router finds no **boot system** commands, it uses the configuration register value to form a filename from which to boot a default system image stored on a network server. (See Table D-3.)

Take the following steps to change the configuration register while running the Cisco IOS software:

Step 1 Enter the **enable** command and your password to enter the privileged level, as follows:

```
router> enable
Password:
router#
```

Step 2 At the privileged-level system prompt (router #), enter the command **configure terminal.** You will be prompted as shown in the following example:

```
router# configure term
Enter configuration commands, one per line.
Edit with DELETE, CTRL/W, and CTRL/U; end with CTRL/Z
```

Step 3 To set the contents of the configuration register, enter the **config-register** *value* configuration command where *value* is a hexadecimal number preceded by 0x (see Table D-3), as in the following:

config-register 0x*value*

(The virtual configuration register is stored in nonvolatile memory.)

Step 4 Press **CTRL-Z** to exit the configuration mode. The new value settings will be saved to memory; however, the new settings do not take effect until the system software is reloaded by rebooting the router.

Step 5 Enter the **show version** EXEC command to display the configuration register value currently in effect and the value that will be used at the next reload. The value will be displayed on the last line of the screen display as in the following example:

```
Configuration register is 0x142 (will be 0x102 at next reload)
```

Step 6 Reboot the router. The new value takes effect. Configuration register changes take effect only when the server restarts, for example, when you switch the power OFF and ON or when you issue a **reload** command from the console.

Configuring the Boot Field

The lowest four bits of the processor configuration register (bits 3, 2, 1, and 0) form the boot field. (See Table D-2.)

Table D-2 Explanation of Boot Field (Configuration Register Bits 00–03)

Boot Field	Meaning
00	Stays at the system bootstrap prompt (ROM monitor) on a reload or power cycle
01	Boots the boot helper image as a system image
02-F	Specifies a default netboot filename Enables default booting from system Flash memory Enables boot system commands that override the default netboot filename[1]

1. Values of the boot field are 2–15 in the form cisco<*n*>-*processor_name*, where 2 ≤ n ≤ 15.

The boot field specifies a number in binary. If you set the boot field value to 0, you must have console port access to boot the operating system manually. Boot the operating system manually by entering the **b** command at the bootstrap prompt as follows:

```
> b [tftp] flash filename
```

Definitions of the various command options follow:

b—Boots the default system software from ROM

b flash—Boots the first file in Flash memory

b *filename [host]*—Boots over the network using TFTP

b flash *[filename]*—Boots the file (*filename*) from Flash memory

For more information about the **b** [**tftp**] **flash** *filename* command, see the appropriate Cisco IOS software publications.

If you set the boot field value to a value of 2 through F, and there is a valid system boot command stored in the configuration file, the router boots the system software as directed by that value. (See Table D-3.) If you set the boot field to any other bit pattern, the router uses the resulting number to form a default boot filename for booting from a network (TFTP) server.

If there are no boot commands in the configuration file, the router attempts to boot the first file in system Flash memory. If no file is found in system Flash memory, the router attempts to netboot a default file whose name is derived from the value of the boot field (for example: cisco2-4500). If the attempt to boot from a network (TFTP) server fails, the boot helper image in boot Flash will boot up.

If boot commands are in the configuration file, the router software processes each boot command in sequence until the process is successful or the end of the list is reached. If the end of the list is reached without a file being successfully booted, the router will retry the netboot commands up to six times unless the boot default ROM software if netboot fails bit (bit 13 of the virtual configuration register) is set. If bit 13 is set, the system boots the boot helper image found in boot Flash memory without any retries.

In the following example, the virtual configuration register is set to boot the router automatically from Flash memory and to ignore Break at the next reboot of the router:

```
router# configure terminal
Enter configuration commands, one per line.
Edit with DELETE, CTRL/W, and CTRL/U; end with CTRL/Z
config-register 0x102
Ctrl-Z
router#
```

The server creates a default boot filename as part of the automatic configuration processes. To form the boot filename, the server starts with *cisco* and links the octal equivalent of the boot field number, a dash, and the processor-type name. Table D-3 lists the default boot filenames or actions for the processor.

Note A **boot system** configuration command in the router configuration in NVRAM overrides the default netboot filename.

Table D-3 Default Boot Filenames

Action/File Name	Bit 3	Bit 2	Bit 1	Bit 0
bootstrap mode	0	0	0	0
ROM software	0	0	0	1
cisco2-4000 or cisco2-4500	0	0	1	0
cisco3-4000 or cisco3-4500	0	0	1	1
cisco4-4000 or cisco4-4500	0	1	0	0
cisco5-4000 or cisco5-4500	0	1	0	1
cisco6-4000 or cisco6-4500	0	1	1	0
cisco7-4000 or cisco7-4500	0	1	1	1
cisco10-4000 or cisco10-4500	1	0	0	0
cisco11-4000 or cisco11-4500	1	0	0	1
cisco12-4000 or cisco12-4500	1	0	1	0
cisco13-4000 or cisco13-4500	1	0	1	1
cisco14-4000 or cisco14-4500	1	1	0	0
cisco15-4000 or cisco15-4500	1	1	0	1
cisco16-4000 or cisco16-4500	1	1	1	0
cisco17-4000 or cisco17-4500	1	1	1	1

Reviewers, any additions to above table?

Bit 8 controls the console Break key. Setting bit 8 (the factory default) causes the processor to ignore the console Break key. Clearing bit 8 causes the processor to interpret Break as a command to force the system into the bootstrap monitor, halting normal operation. A Break can be sent in the first 60 seconds while the system reboots, regardless of the configuration settings.

Bit 10 controls the host portion of the Internet broadcast address. Setting bit 10 causes the processor to use all zeros; clearing bit 10 (the factory default) causes the processor to use all ones. Bit 10 interacts with bit 14, which controls the network and subnet portions of the broadcast address. Table D-4 shows the combined effect of bits 10 and 14.

Table D-4 Configuration Register Settings for Broadcast Address Destination

Bit 14	Bit 10	Address (<net> <host>)
Off	Off	<ones> <ones>
Off	On	<zeros> <zeros>
On	On	<net> <zeros>
On	Off	<net> <ones>

Bit 13 determines the server response to a bootload failure. Setting bit 13 causes the server to load operating software from ROM after five unsuccessful attempts to load a boot file from the network. Clearing bit 13 causes the server to continue indefinitely to attempt loading a boot file from the network. By factory default, bit 13 is cleared to 0.

Bits 11 and 12 in the configuration register determine the baud rate of the console terminal. Table D-5 shows the bit settings for the four available baud rates. (The factory-default baud rate is 9600.)

Table D-5 System Console Terminal Baud Rate Settings

Baud	Bit 12	Bit 11
9600	0	0
4800	0	1
1200	1	0
2400	1	1

Enabling Booting from Flash Memory

To enable booting from Flash memory, set bits 3, 2, 1, and 0 to a value between 2 through 15. To specify a filename to boot, enter the system software configuration **boot system flash** *filename* command in the configuration file.

To enter the configuration mode while in the system software image, enter the **configure** command at the enable prompt as follows:

```
Gateway# configure
Configuring from terminal, memory, or network [terminal]? term
Enter configuration commands, one per line.
Edit with DELETE, CTRL/W, and CTRL/U; end with CTRL/Z
boot system flash filename
```

To disable Break and enable the **boot system flash** command, enter the **config-register** command with a value as follows:

```
config-reg 0x102
CTRL/Z
```

If you set the configuration register value to 0x102, as in the example, it is not necessary to enter the **boot system flash** command unless there is more than one image in Flash memory.