

B

The ROM Monitor and NMI Mini-Monitor

Most of the commands in the ROM monitor and NMI mini-monitor are discussed in the *NeXTSTEP Network and System Administration* manual; some other useful commands are covered in Chapter 9 of this book, "Building, Loading, and Debugging Loadable Kernel Servers." However, a few commands are of interest to only a few hardware or device-driver developers. These rarely used commands are described in this appendix.

ROM Monitor Commands

Most ROM monitor commands display the current value of a register or parameter, and prompt for a new value with a question mark. To enter a new value, type the value and press the Return key. To leave the current value unchanged and skip to the next one (if any), just press the Return key. Type a period to exit the command and leave the current value unchanged.

Numeric values are usually in hexadecimal (base 16) notation if they represent a memory address or data value. The **R** (radix) command can be used to change the input base/radix.

The values of register bit fields are displayed symbolically between angle brackets < > in addition to the numeric value (for example, "2700<trace=0,s,ipl=7>"). For more information on 68040 registers, see Motorola's *MC68040 User's Manual*.

Open Address Register

The command

a [*n*]

accesses the contents of address registers a0 through a7 of the 680x0 processor.

Example:

```
NeXT> a
a0: 00001234? 5678           Change a0 from 0x1234 to 0x5678.
a1: 00000000?              Don't change a1.
a2: 00000001? .           Type a period to exit.
```

```
NeXT> a0
a0: 00001234? 5678           Change a0 from 0x1234 to 0x5678.
```

Open Data Register

The command

d [*n*]

accesses the contents of data registers d0 through d7 of the 680x0 processor.

Example:

```
NeXT> d
d0: 00001234? 5678          Change d0 from 0x1234 to 0x5678.
d1: 00000000?             Don't change d1.
d2: 00000001? .           Type a period to exit.

NeXT> d0
d0: 00001234? 5678          Change d0 from 0x1234 to 0x5678.
NeXT>
```

Open Processor Register

The command

r [*regname*]

accesses the contents of the 680x0 processor registers. Possible values for *regname* are:

pc	Program counter
sr	Status register
usp	User stack pointer
isp	Interrupt stack pointer
msh	Master stack pointer
vbr	Vector base register
sfc	Source function code
dfc	Destination function code
cacr	Cache control register
caar	Cache address register (68030 only)

If you don't specify *regname*, all the processor registers are opened in the order listed above.

Open System Register

The command

s [*systemreg*]

accesses the contents of the system registers. Possible values for *systemreg* are:

intrstat	Interrupt status register
intrmask	Interrupt mask register
scr1	System control register #1
scr2	System control register #2

If you don't specify *systemreg*, all of the system registers are opened in the order they're listed

above.

Examine Memory Locations

The command

```
e [lwb] [addrlist] [format]
```

lets you examine particular locations in memory.

<i>[lwb]</i>	Specify l , w , or b to select long, word, or byte length. The default is long.
<i>[addrlist]</i>	This argument specifies the starting address or list of addresses to cyclically examine. If you don't specify a value for <i>addrlist</i> , the command uses the most recent value of <i>addrlist</i> .
<i>[format]</i>	This argument controls how the value is displayed. It can be any of the standard format types supported by the C language printf() library routine. The default, %x , displays a number in hexadecimal.

To have access to hardware addresses, you must get into the ROM monitor just after the system is powered on, before it has booted. Otherwise, you'll read or write to virtual addresses, not physical addresses. To enter the ROM monitor, do the following:

1. If your computer is on, press the Power key to turn it off.

Locate the ~ key on your keyboard. (You'll need to press it along with the Command and left Alternate keys soon after you turn the power back on.)

2. Turn the computer back on by pressing the Power key.
3. *Within three seconds* of the appearance of the message ^aLoading from disk^o or ^aLoading from network,^o hold down the Command and left Alternate keys and press the ~ key. (If you're using a keyboard that has two Command keys, don't hold down the Alternate key; just hold down one of the Command keys and press the ~ key.)

You should now be in the ROM monitor. You can use the **e** command to look at addresses on the CPU board.

Example 1:

```
NeXT> e 4000000      Examine the long at memory location 0x4000000.  
4000000: 0? 12345678  See that its value is 0x0; deposit the value 0x12345678.  
4000004: 0? .       Type a period to exit.
```

Example 2:

```
NeXT> e 4000000 4000000  Repeatedly examine memory location 0x4000000.  
4000000: 0? 12345678  See that its value is 0x0; deposit the value 0x12345678.  
4000000: 12345678? .  See that its value is 0x12345678; type a period to exit.
```

Open Function Code

The command

```
S [fcode]
```

lets you inspect or modify the 680x0 function code (address space) used with the **e** (examine)

command. The default code is 5 (supervisor data space).

Possible values for *fcode* are:

0	Undefined, reserved
1	User data space (UD)
2	User program space (UP)
3	Undefined, reserved
4	Undefined, reserved
5	Supervisor data space (SD)
6	Supervisor program space (SP)
7	CPU space

Example:

```
NeXT> s
Function code 5 (SD)           Current function code is 5.
NeXT> s 6
Function code 6 (SP)          Change function code to 6.
```

Set Input Radix

The command

R [*radix*]

lets you set the input radix. The default value for the input radix is 16. Any numbers you enter are interpreted in the base of the input radix (for example, input radix 16 means numbers are interpreted as base 16). If you don't specify *radix*, the current default input radix is displayed.

NMI Mini-Monitor Commands

The following NMI mini-monitor commands aren't currently useful for developers outside of NeXT.

Set or Examine Any Kernel Flag

The command

flag=[*value*]

lets you examine or modify internal system flags (most internal system flags are of interest only to system developers at NeXT).

Example:

```
nmi> debug=
kernel flags debug = 0x0       The debug flag has the value 0.
nmi> debug=2
kernel flag = 0x2             Change the value of the debug flag to 2.
```

Set the IP Address

The command

kd_ipaddr=*value*

lets you set the IP address used by the computer for kernel debugging. It's useful only to system developers at NeXT.